



PHẦN VI

*Microsoft* **ACCESS**

**2007**



## Bài 1. Cơ bản về Access 2007

---

### 1. Khởi động, tìm hiểu giao diện

Bài viết này bàn về các cải tiến trên giao diện người dùng của Microsoft Office Access 2007 (UI). Giao diện người dùng mới của Microsoft Office Access 2007 là kết quả của việc nghiên cứu kỹ lưỡng và kiểm tra rộng rãi, nó được thiết kế để làm cho việc tìm kiếm những lệnh mà bạn cần trở nên dễ dàng hơn.

Bạn sẽ tìm hiểu về các yếu tố mới trên giao diện người dùng - những gì nó cung cấp và làm thế nào để làm việc với chúng. Bạn cũng sẽ tìm hiểu làm thế nào để kích hoạt tính năng tabbed tài liệu trong một cơ sở dữ liệu, và làm thế nào để định hướng những đối tượng giao diện mới.

Giao diện người sử dụng mới trong Microsoft Access 2007 bao gồm một số các yếu tố xác định cách thức bạn tương tác với các sản phẩm. Những yếu tố mới này được chọn để giúp bạn làm chủ Access, và giúp bạn tìm thấy nhanh hơn những lệnh mà bạn cần. Cách thiết kế mới cũng giúp bạn dễ dàng khám phá các tính năng khác có thể còn vẫn ẩn dưới lớp thanh công cụ và trình đơn. Bạn cũng sẽ bắt đầu nhanh hơn nhờ vào trang Getting Started trong Microsoft Office Access, trong đó cung cấp cho bạn khả năng tiếp cận nhanh hơn khi bạn chưa có nhiều kiến thức, bao gồm một bộ các Template được thiết kế chuyên nghiệp.

Yếu tố quan trọng nhất trên giao diện mới được gọi là yếu tố Ribbon, nó là một phần của giao diện Microsoft Office fluent. Các Ribbon trải dài phía trên cùng của cửa sổ chương trình và chứa các nhóm lệnh. Microsoft Office fluent Ribbon cung cấp các lệnh, nhóm lệnh thay thế cho Menu và thanh công cụ cổ điển. Trên Ribbon là các Tab kết hợp những lệnh có cùng ý nghĩa. Ribbon chính các Tab chính mà bạn thường thấy là Home, Create, External Data, and Database. Mỗi Tab lại chứa các nhóm lệnh có liên quan đến nhau.

Các thành phần giao diện mới trong Microsoft Access 2007 là:

- Getting Started with Microsoft Office Access: Trang này được hiển thị khi bạn bắt đầu với Access.

- Microsoft Office Fluent Ribbon: Là khu vực ở phía trên cùng của cửa sổ chương trình, nơi bạn có thể chọn các lệnh.

Tab Command: Những lệnh có cùng ý nghĩa được kết hợp với nhau.

+ Nội dung lệnh Tab: Một thẻ lệnh xuất hiện tùy thuộc vào bối cảnh, những đối tượng mà bạn đang làm việc hoặc công việc mà bạn đang thực hiện.

+ Gallery: Là điều khiển hiển thị một sự lựa chọn trực quan qua đó bạn có thể nhìn thấy những kết quả mà bạn thực hiện.

+ Quick Access Toolbar: Một thanh công cụ chuẩn trên Ribbon cho phép người dùng Click chuột chọn tới những lệnh cần sử dụng như Save, Undo, ....

- Navigation Pane: Nằm ở khu vực bên trái cửa sổ hiển thị các đối tượng trong Database của bạn. Pane Navigation thay thế của sổ DataBase mà bạn thường thấy ở các phiên bản cũ của Access.

- Tabbed documents: Các đối tượng như Table, Query, Form, Report, Page, và các Macro của bạn được hiển thị trong Tabbed Document.

- Status bar: Nằm ở dưới cùng của cửa sổ chương trình sẽ hiển thị các thông tin trạng thái về hoạt động, sự kiện mà bạn vừa thao tác, ngoài ra nó còn chứa các nút lệnh cho phép bạn thay đổi chế độ hiển thị đối tượng hiện thời.

#### 1.1. Khởi động Microsoft Access 2007

Để khởi động ứng dụng bạn có thể làm theo các cách sau:

- Từ Desktop Double Click vào biểu tượng Microsoft Office Access 2007 (nếu có).
- Từ thanh Start Click chọn All Programs/Microsoft Office/Microsoft Office Access 2007.

- Cách thứ 3 bạn Click chọn Start/Run... Hộp thoại xuất hiện Click chọn nút Browse. Trong hộp thoại này bạn trở tới đường dẫn:

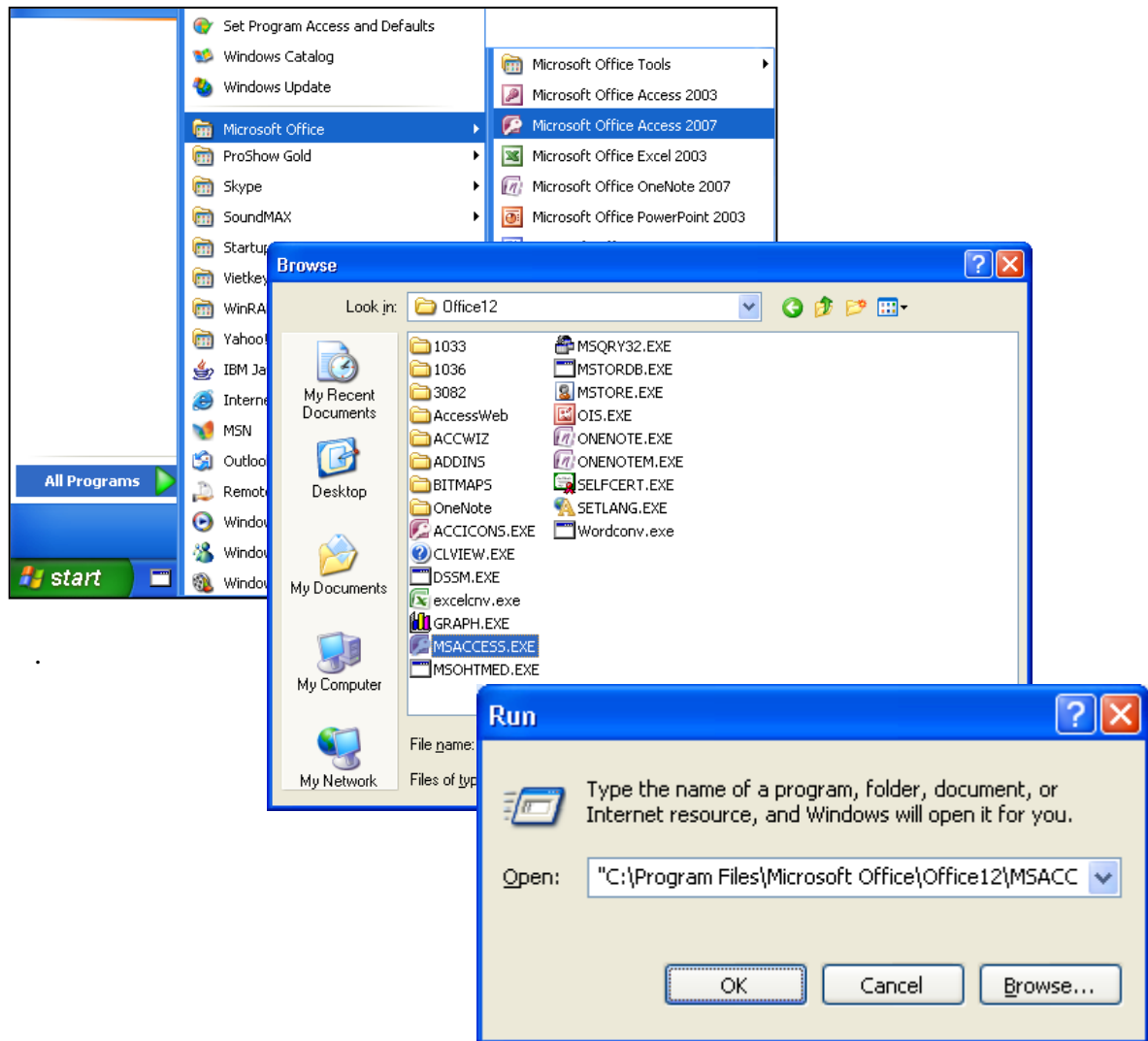
- "C:\Program Files\Microsoft Office\ Office12\ MSACCESS.EXE"

- Từ thanh Start Click chọn All Programs/Microsoft Office/Microsoft Office Access 2007.

- Cách thứ 3 bạn Click chọn Start/Run... Hộp thoại xuất hiện Click chọn nút Browse. Trong hộp thoại này bạn trở tới đường dẫn:

"C:\Program Files\Microsoft Office\ Office12\ MSACCESS.EXE"

Rồi Click Open




Cuối cùng Click chọn Ok để kết thúc.

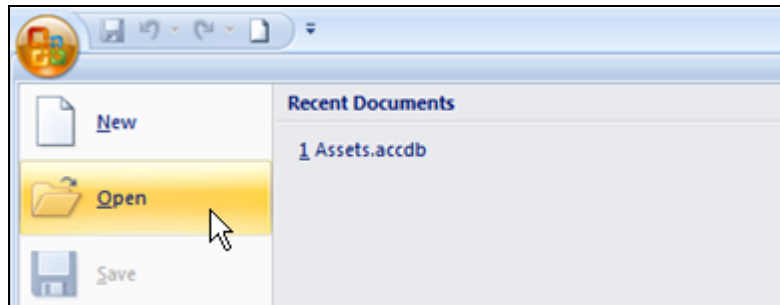
Bằng một trong 3 cách trên bạn màn hình làm việc Microsoft Access 2007 sẽ xuất hiện như hình



## 1.2. Getting Started with Microsoft Office Access

Khi bạn bắt đầu với Microsoft Access 2007, trang Getting Started with Microsoft Office Access sẽ xuất hiện như hình dưới đây. Trang này cho thấy bạn có thể làm gì để bắt đầu với Microsoft Access 2007.

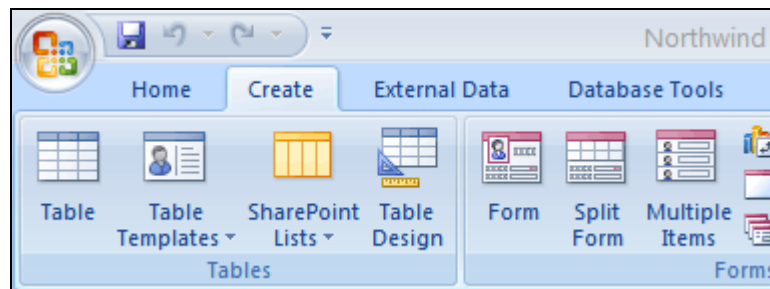
Ví dụ, bạn có thể tạo mới một DataBase trống, Tạo một DataBase từ Template, hoặc mở một DataBase đã tồn tại (nếu bạn đã mở một vài DataBase trước đó). Bạn cũng có thể truy cập trực tiếp tới Microsoft Office Online để học thêm về về hệ thống Microsoft Office 2007 và Microsoft Access 2007, hoặc bạn cũng có thể Click chọn nút Office Button  và sử dụng menu Open để mở một DataBase đã tồn tại trên đĩa cứng.



## 1.3. Ribbon

Office Fluent Ribbon là một sự thay thế chính cho các menu và thanh công cụ và cung cấp các lệnh chính trên giao diện của Microsoft Access 2007. Một trong những lợi ích chính của Ribbon là củng cố tập chung tất cả các yêu cầu về một điểm mà trước đây nó nằm trên Menu, ToolBar, Task Pane và các thành phần giao diện khác. Với Ribbon bạn chỉ phải quan tâm tới một nơi ở đó chứa tất cả các lệnh mà bạn cần để làm việc với Microsoft Access 2007.

Khi bạn mở một Cơ sở dữ liệu, Ribbon sẽ xuất hiện trên đỉnh của cửa sổ chính trong Office Access 2007.



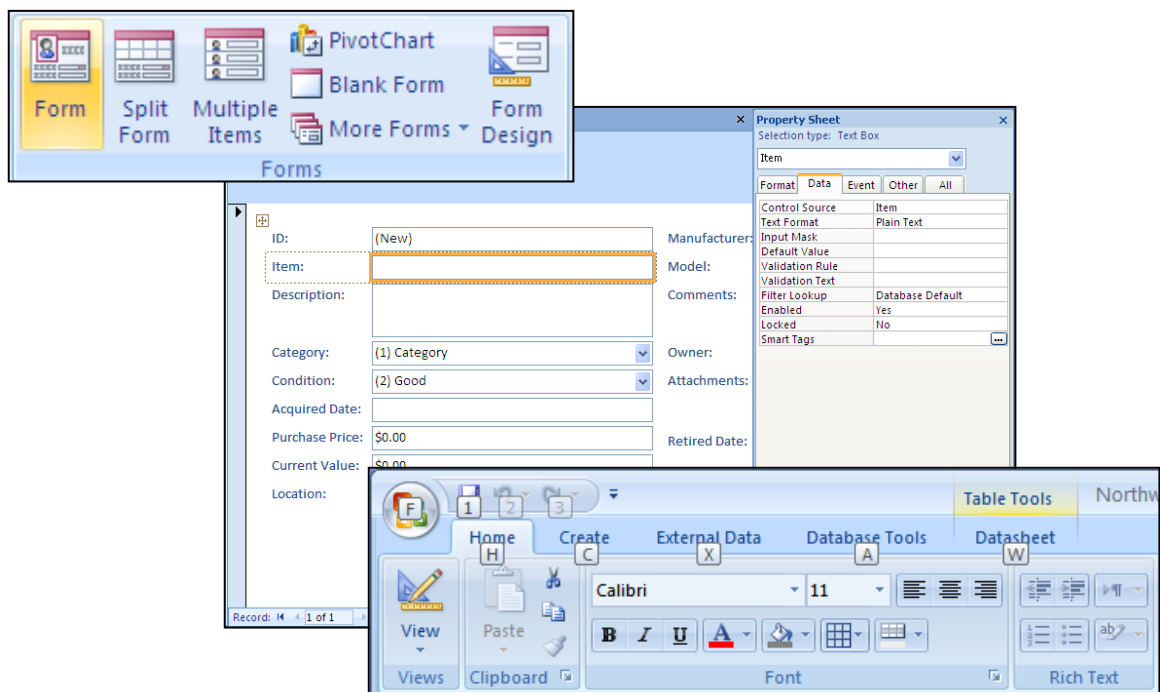
Các Ribbon chứa một loạt các Tab lệnh có chứa lệnh. Trong Microsoft Access 2007, Các Tab lệnh chính là Home, Create, External Data, và Database Tools. Mỗi tab chứa các nhóm lệnh liên quan.

Những lệnh trên Ribbon đưa vào “tài khoản” của đối tượng hiện đang hoạt động. Ví dụ, nếu bạn có một bảng mở ở Datasheet View và bạn Click chuột vào Form trên Tab Create, trong nhóm Forms.

Microsoft Access 2007 sẽ tạo ra một Form mới dựa trên Table đang hoạt động. Và tên trường sẽ được gắn vào thuộc tính Control Source của từng Control trên Form.

Bạn có thể sử dụng các phím tắt với Ribbon. Tất cả các phím tắt ở phiên bản trước đều có thể sử dụng trên phiên bản này. Keyboard Access System thay thế các menu ở những phiên bản cũ. Đây là hệ thống sử dụng các chỉ số nhỏ xuất hiện trên giao diện Ribbon những phím nào có thể truy cập được các chức năng.

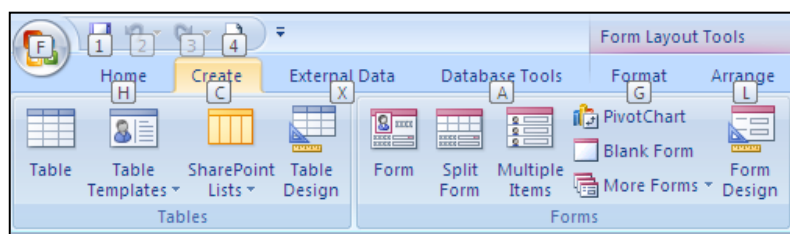
Khi bạn đã chọn một tab lệnh, bạn có thể duyệt qua các lệnh có sẵn trong tab đó.



### Chọn Tab lệnh

**Bước 1:** Khởi động Microsoft Access 2007

**Bước 2:** Click chuột chọn tab cần sử dụng hoặc Click phím ALT khi đó trên màn hình sẽ xuất hiện như hình dưới đây:



Bạn muốn chọn nút nào thì nhấn phím tương ứng như hướng dẫn trên màn hình.

Có rất nhiều cách khác nhau để thực hiện một lệnh. Nhưng nhanh và trực tiếp là cách sử dụng các tổ hợp phím nóng. Nếu bạn đã từng sử dụng các phím nóng này trong những phiên bản trước của Access thì bạn sẽ rất dễ dàng khi làm việc với Microsoft Access 2007.

### ***Thực hiện một lệnh***

***Bước 1:*** Khởi động Access

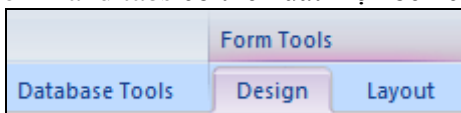
***Bước 2:*** Click chọn Tab thích hợp tương ứng với lệnh cần sử dụng. Bảng dưới đây là một số lệnh chung nhất trong các Tab trên Ribbon, tuy nhiên các Tab lệnh cũng sẽ thay đổi tùy thuộc vào những gì bạn đang làm việc.

<b>Tab lệnh</b>	<b>Những điều bạn có thể làm</b>
Home	Chọn chế độ hiển thị Sao chép và dán từ Clipboard Đặt lại Font chữ hiện tại Đặt lại Alignment cho nội dung Định dạng lại nội dung cho trường Memo Làm việc với bản ghi (Refresh, New, Save, Delete, Totals, Spelling). Sắp xếp và lọc bản ghi Tìm kiếm bản ghi.
Create	Tạo một Table mới Tạo một bảng mới sử dụng Template Tạo mới một bảng trống ở chế độ Design view. Tạo mới một Form dựa trên Table hoặc Query đang hoạt động. Tạo mới một bảng hoặc một biểu đồ pivot. Tạo mới một báo cáo dựa trên một bảng hoặc một Query đang hoạt động. Tạo mới một Query, Macro, Module hoặc một Class Module
External Data	Import hoặc liên kết với dữ liệu bên ngoài. Kết xuất dữ liệu Thu thập và cập nhật dữ liệu từ Email Làm việc với danh sách offline SharePoint. Di chuyển một số hay toàn bộ các phần của cơ sở dữ liệu tới một SharePoint site mới hoặc đã tồn tại trước đó.
Database Tools	Chạy Visual Basic editor hoặc chạy macro. Tạo và hiển thị mối liên kết giữa các bảng. Ẩn, hiện đối tượng phụ thuộc hoặc thuộc tính của Sheet Chạy Database Documenter hoặc phân check hiệu suất Di chuyển dữ liệu tới Microsoft SQL Server hoặc tới cơ sở dữ liệu khác. Chạy Linked Table Manager. Manage Access add-ins. Tạo vào sửa các Module Visual Basic for Applications (VBA).

**Bước 3:** Click chuột chọn tab cần sử dụng hoặc Click phím ALT khi đó trên màn hình sẽ xuất hiện các chỉ dẫn, bạn muốn chọn nút nào thì nhấn phím tương ứng như hướng dẫn trên màn hình.

### **Contextual command tabs**

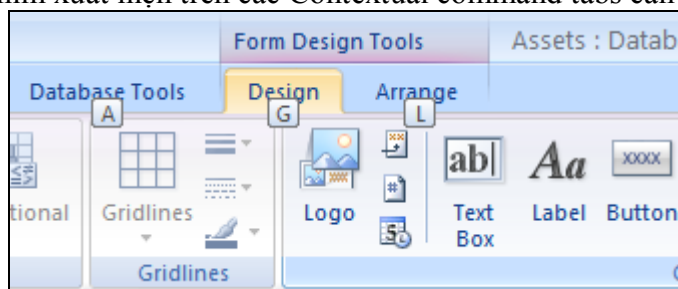
Ngoài những Tab lệnh chuẩn, Microsoft Access 2007 còn sử dụng một thành phần giao diện mới trong Office Professional 2007 được gọi là contextual command tab. Tùy thuộc vào bối cảnh của bạn (có nghĩa là, đối tượng mà bạn đang làm việc và những gì bạn đang làm), một hoặc nhiều Contextual command tabs có thể xuất hiện bên cạnh các Tab lệnh chuẩn.



Khởi động một contextual command tab

**Bước 1:** Click phím Alt khi này các ký tự đại diện sẽ xuất hiện trên từng chức năng.

**Bước 2:** Nhấn phím xuất hiện trên các Contextual command tabs cần chọn.



Các contextual command tabs chứa lệnh và các tính năng mà bạn cần phải làm việc trong một bối cảnh cụ thể. Ví dụ, khi bạn mở một bảng trong thiết kế xem, các lệnh trên contextual command tab chỉ áp dụng khi bạn làm việc với bảng ở chế độ đó. Một ví dụ khác, khi bạn mở một bảng ở chế độ Design View, một contextual command tabs tên là Design xuất hiện tiếp theo là Tab Advanced Tools. Khi bạn Click chọn Tab Design, Ribbon hiển thị những lệnh có tác dụng chỉ trên những đối tượng ở chế độ Design View.

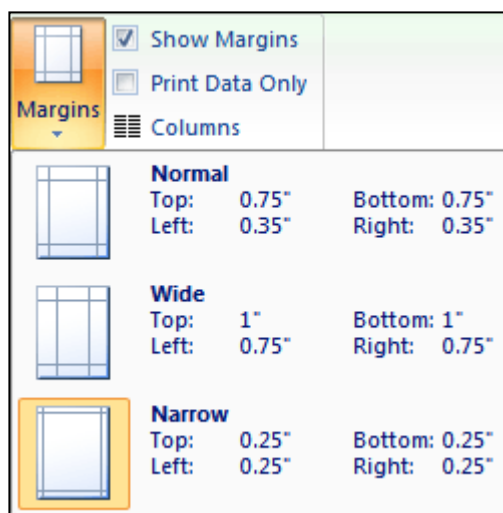
### **Galleries**

Một sự đổi mới khác và là một phần quan trọng trong giao diện mới của Microsoft Access 2007 và các Control mới được gọi là gallery. Các gallery Control được thiết kế để làm việc với Ribbon và hướng sự chú ý của bạn tới những kết quả sẵn có. Thay vì chỉ hiển thị lệnh các gallery Control cho thấy kết quả của việc sử dụng các lệnh đó. Mục đích là để cung cấp cách nhìn trực quan giúp bạn có thể duyệt qua và xem những gì Microsoft Access 2007 có thể làm. Và như vậy bạn sẽ chỉ quan tâm tới kết quả thay vì phải suy nghĩ xem lệnh sẽ thực hiện những gì.

Galleries có hình dạng và kích cỡ khác nhau. Các kết quả được bố trí trên một bảng đồ xuống cho phép người dùng lựa chọn

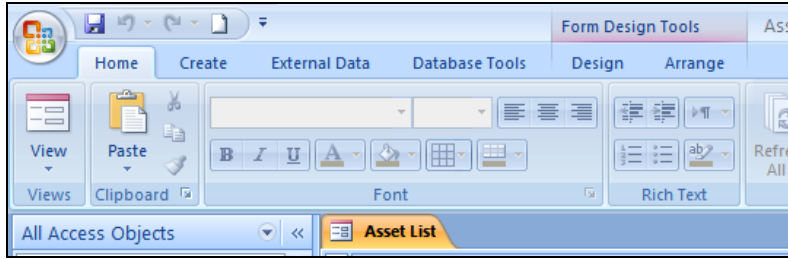
### **Ẩn hiện các Ribbon**

Đôi khi, bạn có thể cần thêm một chút không gian, khoảng trống cho khu vực làm việc của bạn. Bạn có thể làm cho các Ribbon thu nhỏ lại trên màn hình chỉ còn các Bar và các Tab lệnh. Để làm được điều này bạn chỉ cần Double Click chuột vào Ribbon, khi cần chúng hiện ra bạn hãy làm tương tự như vậy một lần nữa.

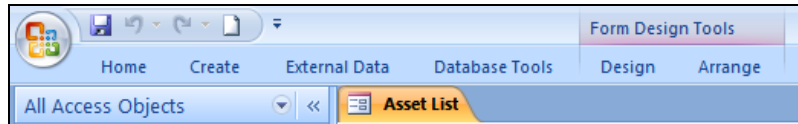




## Ban đầu

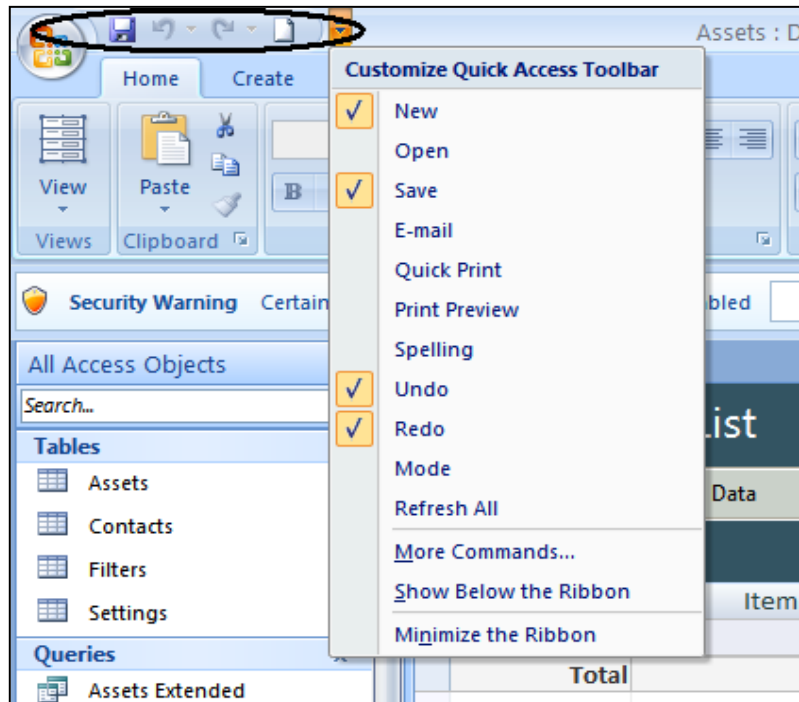


Sau khi làm ẩn các Ribbon



### 1.4. Thanh công cụ Quick Access

Là thanh công cụ nhỏ mặc định nằm kề bên Ribbon, bạn Click chuột trực tiếp vào lệnh cần sử dụng. Các lệnh mặc định ở thanh công cụ là những lệnh bạn thường sử dụng như Save, Undo, Redo. Tuy nhiên, bạn có thể tùy chỉnh thanh công cụ Quick Access, bạn cũng có thể bổ xung các lệnh khác lên đó. Bạn cũng có thể thay đổi vị trí, kích thước của thanh công cụ. Khi bạn thay đổi vị trí của thanh công cụ này xuống phía dưới của Ribbon kích thước nó sẽ được kéo rộng.



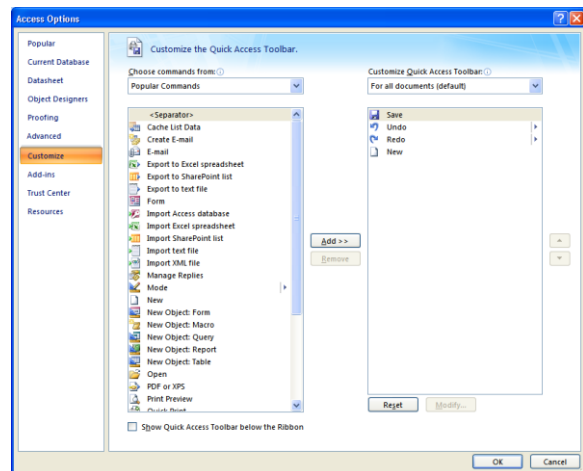
### Tùy biến thanh công cụ Quick Access

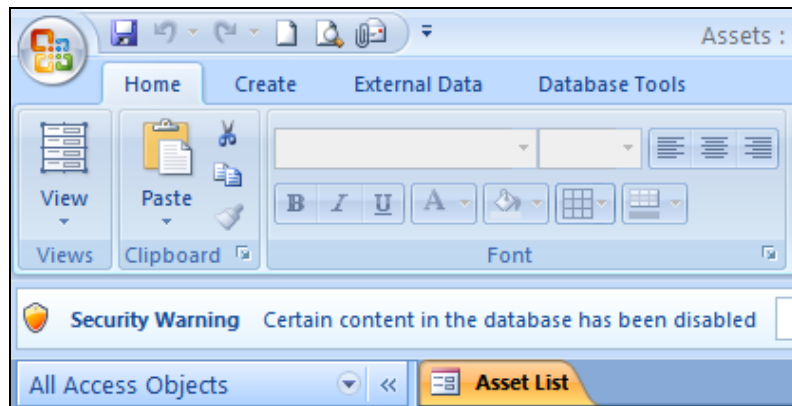
Click chuột vào hộp thoại đổ xuống bên cạnh thanh công cụ

**Bước 1:** Trong hộp thoại đổ xuống Customize Quick Access Toolbar Click chọn một số lệnh thường chú được liệt kê sẵn như Open, E-Mail, Quick Print, ... hoặc Click chọn More Commands, khi này hộp thoại Access Option xuất hiện.

**Bước 2:** Trong hộp thoại này bạn chọn lệnh cần thêm vào ToolBar rồi Click Add, hoặc làm ngược lại để loại bớt lệnh trong ToolBar.

**Bước 3:** Cuối cùng Click Ok để hoàn tất.



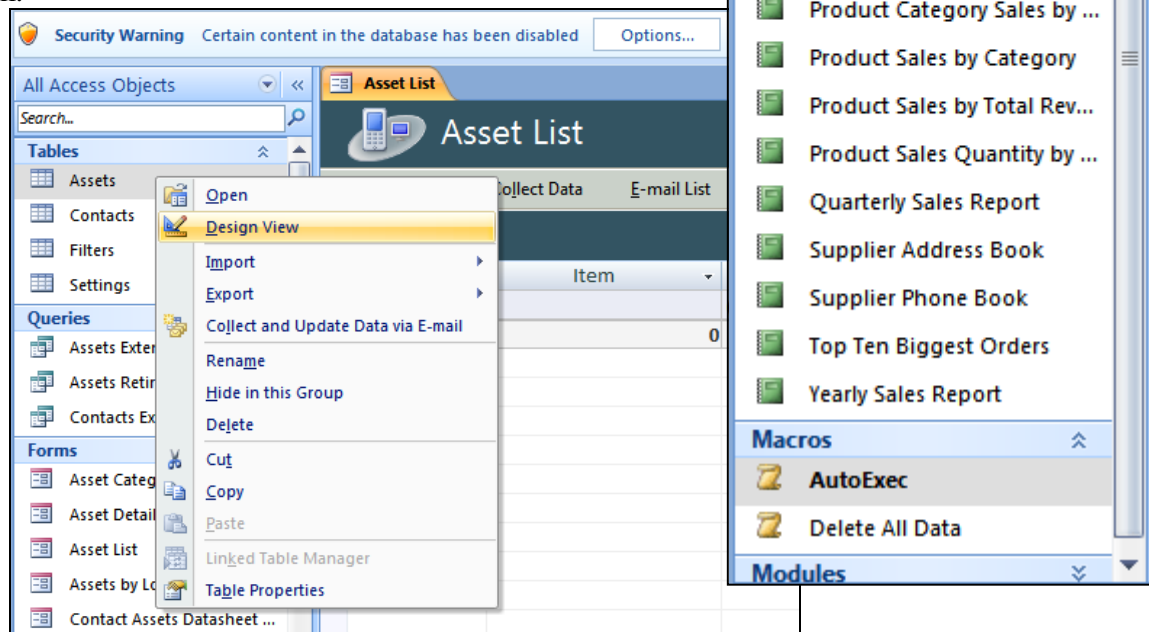


Sau khi thay đổi bạn sẽ có được ToolBar như ý muốn.

### 1.5. Navigation Pane

Khi bạn tạo mới hoặc mở một DataBase, tên các đối tượng trong DataBase của bạn sẽ nằm trong Navigation Pane. Các đối tượng này bao gồm các Table, Form, Report, Page, Macro, Module. Navigation Pane thay thế cho cửa DataBase được sử dụng trong các phiên bản trước của Access. Nếu như trước đây bạn sử dụng cửa sổ DataBase để làm việc thì bây giờ trong Microsoft Access 2007 bạn sẽ sử dụng Pane Navigation để làm những việc đó. Ví dụ nếu bạn muốn thêm một trong vào bảng ở chế độ DataSheet View, bạn mở bảng từ Navigation Pane.

Để mở hoặc thực thi một lệnh cho một đối tượng trên DataBase bạn Click phải chuột vào đối tượng đó, một Menu tương ứng sẽ xuất hiện bạn chọn lệnh cần thực hiện.



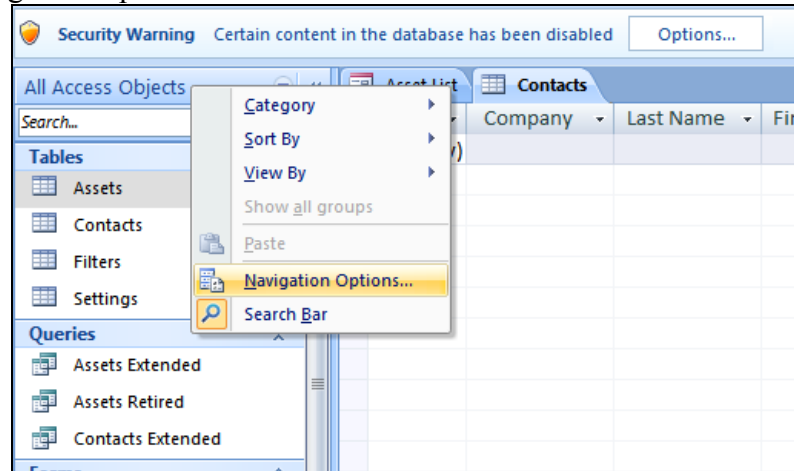
Nội dung của các Context Menu này tùy thuộc vào ngữ cảnh mà bạn đang làm việc.

Dưới đây là các cách mở một đối tượng trong DataBase ví dụ như Table, Form hoặc Report.

Cách 1: Từ Navigation Pane chọn đối tượng cần thao tác rồi Click phím Enter, hoặc Double Click chuột.

Cách 2: Từ Navigation Pane chọn đối tượng cần thao tác rồi Click phải chuột, trong Menu đổ xuống chọn lệnh cần thực hiện.

Chú ý: Bạn có thể đặt tùy chọn mở một đối tượng chỉ bằng một động tác Click chuột trong cửa sổ Navigation Option.

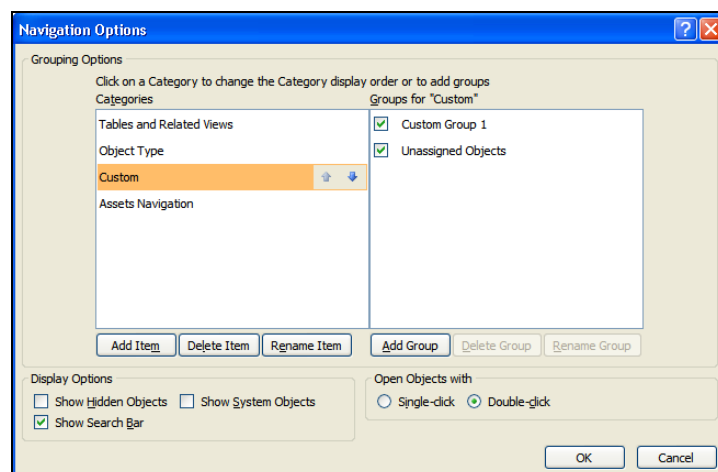
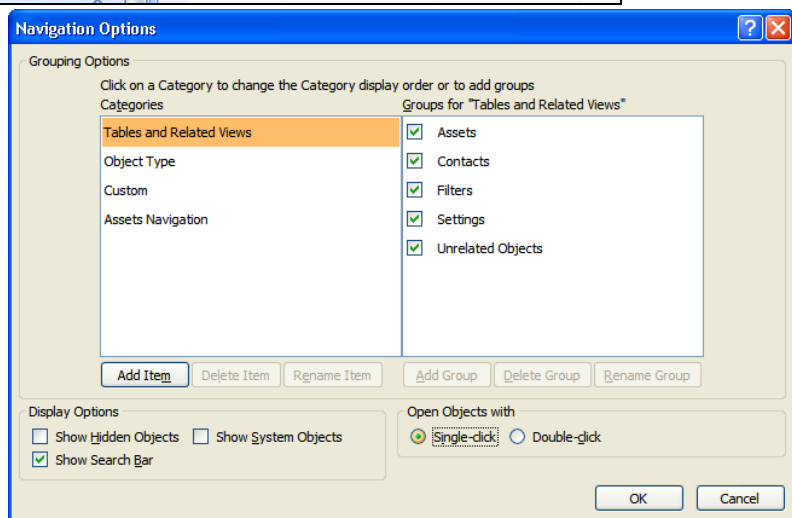


Để bật hộp thoại này bạn Click phải chuột lên phần chữ All Access Objects trên Navigation Option, chọn Navigation Options...

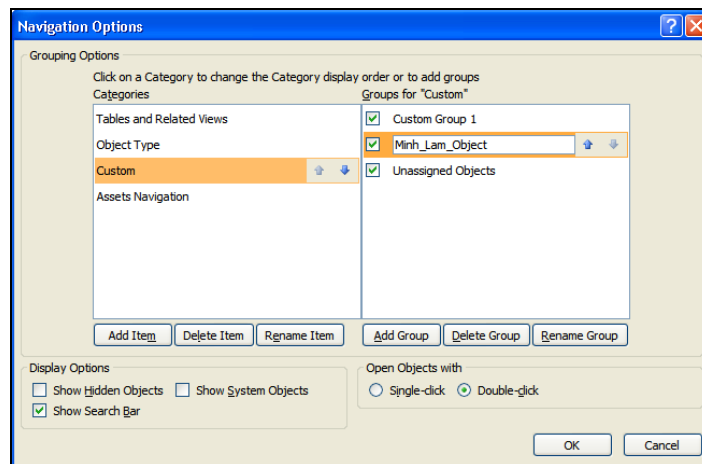
Trong mục Open Objects With bạn check chọn Single-click. Cuối cùng Click Ok để hoàn tất.

Navigation Pane chia các đối tượng trong DataBase của bạn thành các mục khác nhau, và mỗi danh mục đó chứa một nhóm các đối tượng. Một số mục như bạn thấy khi lần đầu tiên làm việc với Pane này đã được tạo sẵn, bạn hoàn toàn có thể thêm vào các nhóm của riêng mình.

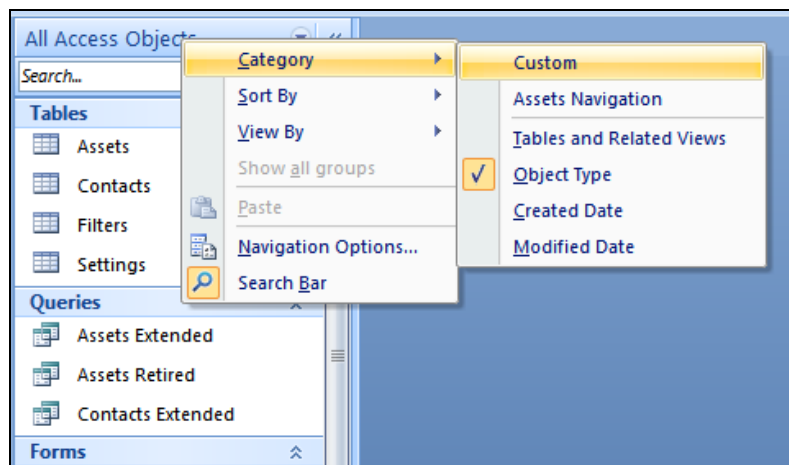
Đầu tiên bạn mở hộp thoại Navigation Option như ở phần trước.



Ở mục Categories bạn chọn Custom rồi Click Add Group sau đó Click vào tên của nhóm mới.



Click Enter để xác nhận rồi Click Ok để hoàn tất, như vậy bạn đã có một nhóm mới trong Navigation Pane, bạn có thể chứa các đối tượng mới mà mình tạo ra. Vì mặc định Pane Navigation đang sắp xếp theo Objects Type nên bạn không thể thấy được nhóm mới mà mình vừa tạo ra, bạn cần phải sắp xếp lại theo Custom.



Sau khi chọn kiểu sắp xếp này Pane Navigation sẽ như hình dưới đây:

Mặc định Navigation Pane xuất hiện khi bạn mở một DataBase trong Microsoft Access 2007, cũng như các DataBase được tạo ở phiên bản trước. Tuy nhiên bạn có thể thiết lập lại chế độ cho Pane Navigation. Dưới đây là các bước thực hiện.

#### **Hiện thị hoặc ẩn Navigation Pane**

Click nút phía trên bên góc phải của Pane Navigation

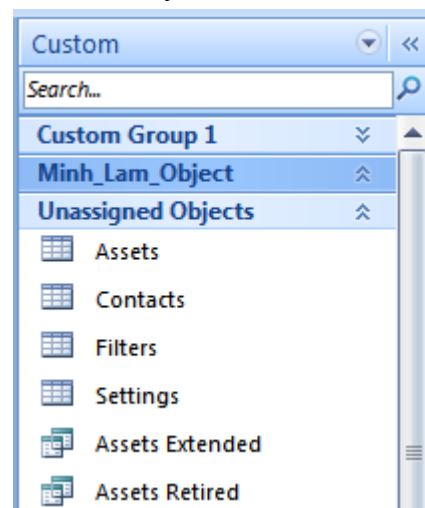


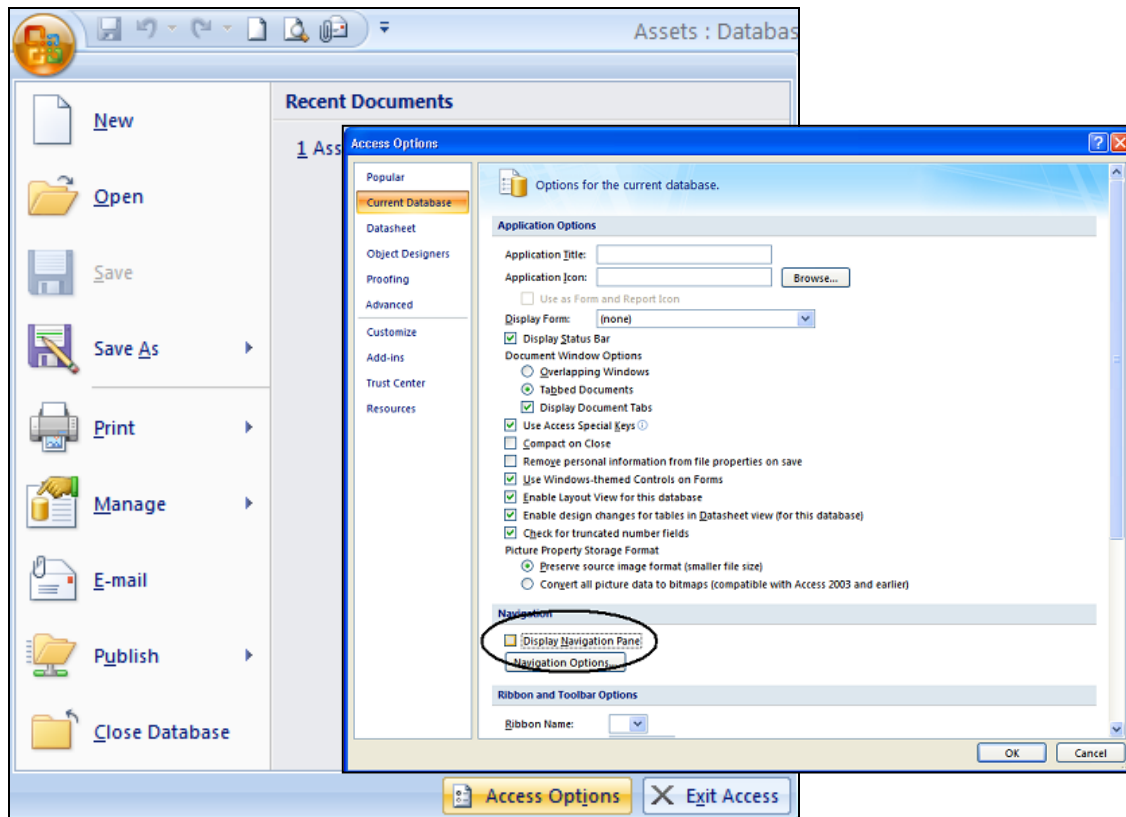
hoặc nhấn phím F11.

**Bước 1:** Click chuột vào nút Microsoft Office Button



sau đó chọn Access Option.



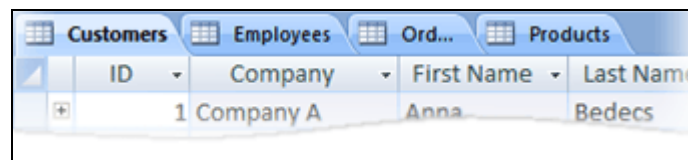


**Bước 2:** Bên phía trái Pane Click chọn Current DataBase.

**Bước 3:** Trong hộp thoại này phía dưới mục Navigation bạn bỏ check mục Display Navigation Pane nếu không muốn Pane này xuất hiện khi mở DataBase và ngược lại nếu nó xuất hiện mặc định. Cuối cùng Click Ok để hoàn tất.

### 1.6. Tab document

Bắt đầu với Microsoft Access 2007 bạn có thể hiển thị các đối tượng trong cơ sở dữ liệu của mình trên các Tab Document thay vì nằm chổng chéo trên các cửa sổ. Với Tab Document sẽ giúp bạn thuận lợi hơn khi phải làm việc thường xuyên với các đối tượng, tuy nhiên bạn có thể sử dụng hoặc không sử dụng giao diện Tab Document bằng cách thiết lập trong Access Option. Sau đó bạn phải đóng và mở lại DataBase để thiết lập mới có hiệu lực.



### ***Ẩn hiện chế độ Tab Document.***

**Bước 1:** Click chọn nút Microsoft Office Button , sau đó chọn Access Options.

Hộp thoại Access Option xuất hiện.

**Bước 2:** Bên phía trái Pane Click chọn Current DataBase.

**Bước 3:** Trong phần Application Options, dưới mục Document Window Options bạn check hoặc bỏ check chọn để tắt bật chế độ này

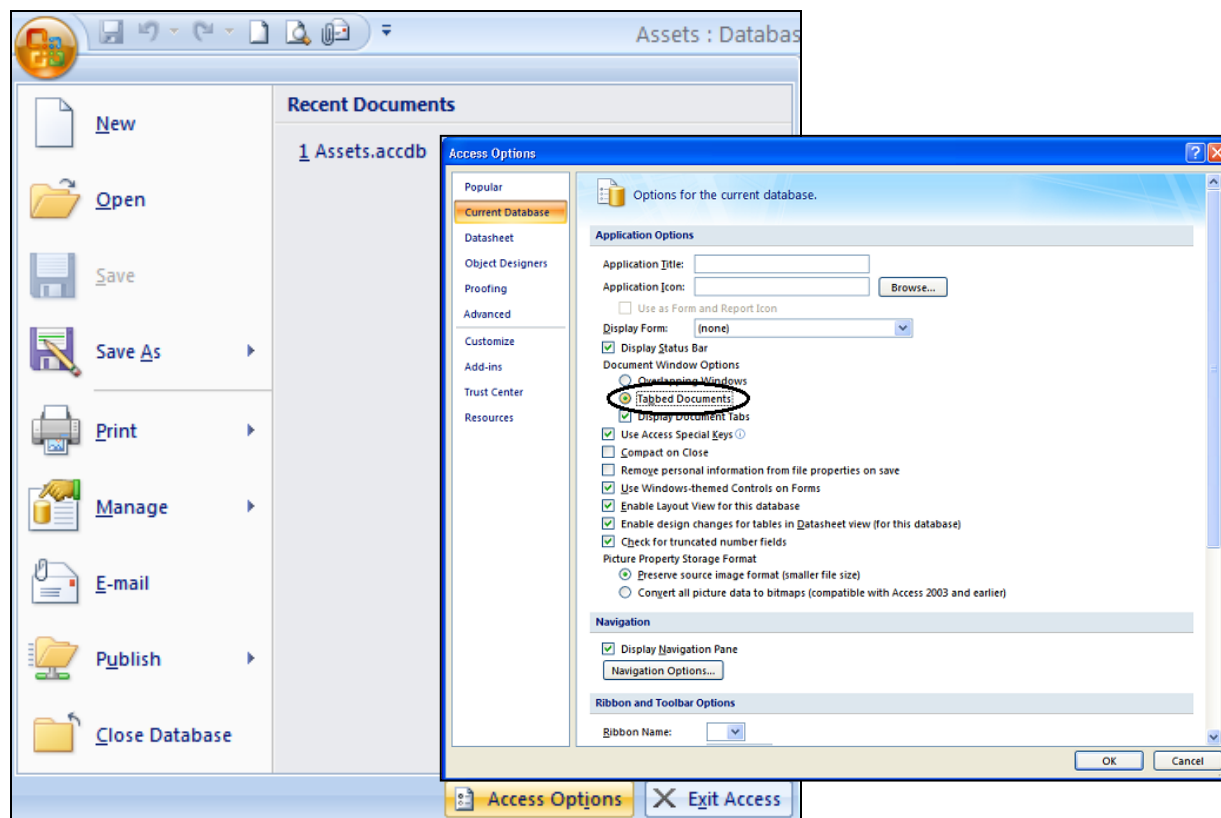
**Bước 4:** Click Ok để hoàn tất.

**Chú ý:** Thiết lập Display Document Tabs không phải là tùy chọn toàn cục nó chỉ có tác dụng trên DataBase mà bạn đang thao tác. Vì vậy nếu bạn muốn thiết lập cho các DataBase khác bạn phải thực hiện lại từ đầu.

Sau khi thay đổi thiết lập Display Document Tabs. Bạn cần phải đóng rồi mở lại DataBase của mình để thấy những thay đổi có tác dụng.

Những DataBase mới được tạo bởi Microsoft Access 2007 được gán mặc định chế độ Tab Document.

Những DataBase được tạo từ các phiên bản trước của Access sẽ được gán mặc định chế độ các cửa sổ riêng rẽ.



## 1.7. Status bar

Cũng giống như những phiên bản trước của Access, với Microsoft Access 2007, bạn có thể hiển thị thanh trạng thái dưới đáy của cửa sổ hiện thời. Nó tiếp tục là một thành phần chuẩn trong ứng dụng giúp người dùng có thể thấy các thông báo trạng thái, thuộc tính, gợi ý, ... với Microsoft Access 2007, thanh trạng ngoài những chức năng kể trên nó còn đảm nhiệm hai chức năng chuẩn khác mà bạn có thể tìm thấy trong tất cả các ứng dụng của bộ Office 2007 là View (chọn chế độ hiển thị) và Zoom (phóng to, thu nhỏ).

Bạn có thể dễ dàng chuyển đổi giữa các chế độ hiển thị cửa sổ Window bằng cách sử dụng các điều khiển sẵn có trên thanh Status Bar.

Nếu bạn đang hiển thị một đối tượng có cung cấp khả năng Zoom, bạn có thể điều chỉnh các mức Zoom để phóng to hoặc thu nhỏ đối tượng, hay có thể sử dụng thanh trượt trên thanh trạng thái

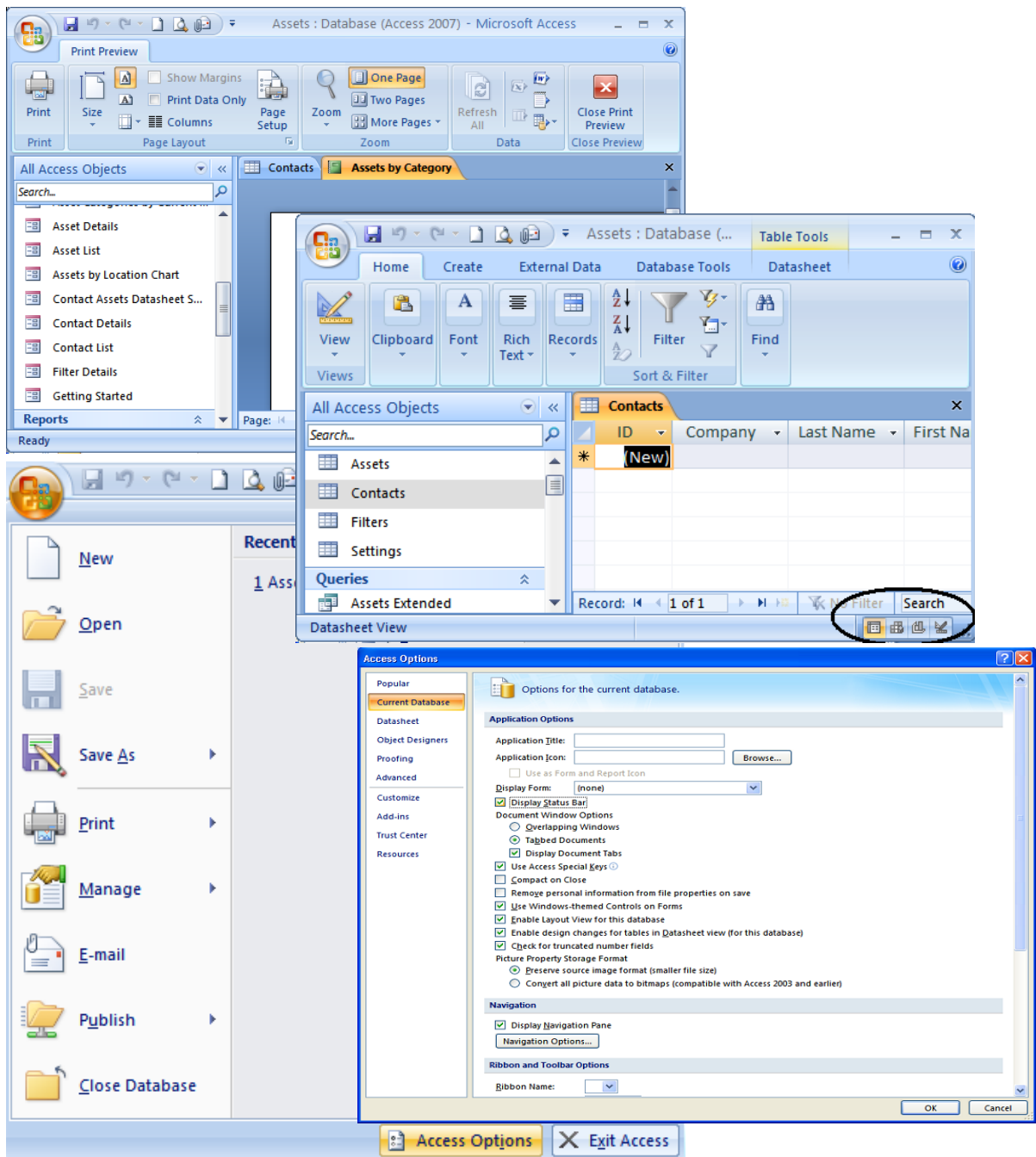
Thanh trạng thái có thể tắt bật dễ dàng với tùy chỉnh trong hộp thoại Access Option.

Dưới đây là các bước hiển thị hoặc ẩn đi thanh trạng thái.

**Bước 1:** Click chọn nút Microsoft Office Button , sau đó chọn Access Options  
Hộp thoại Access Option xuất hiện.

**Bước 2:** Bên phía trái Pane Click chọn Current DataBase.

**Bước 3:** Dưới mục Application Options, chọn hoặc hủy chọn Display Status Bar để tắt hoặc bật Status Bar.



**Bước 4:** Cuối cùng Click Ok để hoàn tất.

### 1.8. Mini toolbar

Một trong những thao tác phổ biến trong các chương trình Office Professional là định dạng văn bản. Trong các phiên bản trước đây của Access, để định dạng văn bản chương trình luôn yêu cầu sử dụng một trình đơn hoặc một



thanh công cụ hiển thị các định dạng. Với Microsoft Access 2007, bạn có thể định dạng văn bản một cách dễ dàng hơn bằng cách sử dụng Mini Toolbar. Khi bạn chọn cho các định dạng văn bản, các Mini Toolbar tự động xuất hiện trên văn bản được lựa chọn. Nếu bạn di chuyển con chuột trở lại gần các nút lệnh trên thanh Mini Toolbar chúng sẽ sáng lên và cho phép bạn có thể sử dụng nó để áp dụng các định dạng như Bold, Italic, Font Size, Color ... cho văn bản.

### 1.9. Tạo Database đầu tiên

#### **Tạo một DataBase mới rỗng**

**Bước 1:** Bắt đầu với Microsoft Access 2007 từ Menu Start hoặc từ Shortcut. Trang Getting Started with Microsoft Office Access sẽ xuất hiện.

**Bước 2:** Trên trang Getting Started with Microsoft Office Access dưới mục New Blank DataBase Click chọn Blank DataBase.

**Bước 3:** Trên Pane Blank DataBase, trong hộp File Name gõ vào tên tệp tin .accdb

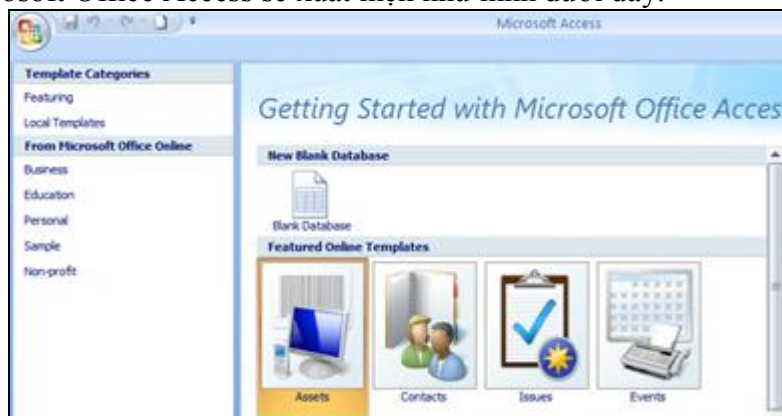
**Bước 4:** Click Create.

Sau 4 bước trên một DataBase với tên tệp tin là Minh\_Lam.accdb sẽ được tạo và một bảng mới được mở trong Datasheet view.

Microsoft Access 2007 cung cấp một số Template với một số sản phẩm, và bạn có thể tải về nhiều hơn từ Microsoft Office Online. Vậy Template là gì? Trong bối cảnh Microsoft Office Access 2007, đó là một DataBase được thiết lập hoàn chỉnh từ trước với các Table, Form, và Report chuyên nghiệp. Các Template đem đến cho bạn một sự khởi đầu nhanh hơn khi bạn bắt đầu tạo lập một DataBase mới.

#### **Tạo DataBase mới từ Template**

**Bước 1:** Khởi động Microsoft Access 2007 từ thanh Start hoặc từ Shortcut. Trang Getting Started with Microsoft Office Access sẽ xuất hiện như hình dưới đây.



**Bước 2:** Trên trang Getting Started with Microsoft Office Access dưới mục Featured Online Templates, Click chọn một mẫu Template, giả sử ở đây chúng ta chọn Assets.

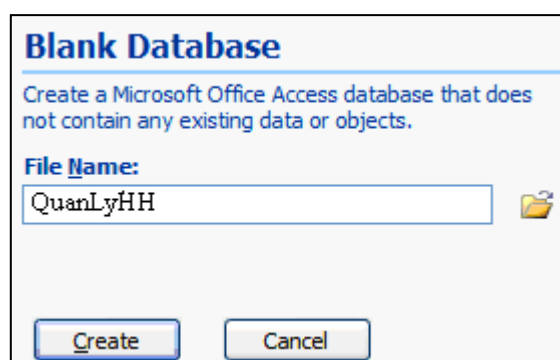
**Bước 3:** Trong hộp File Name, gõ vào tên file với phần mở rộng là **.accdb**

**Bước 4:** Tùy chọn, Click chọn “Create and link your database to a Windows SharePoint Services site” nếu bạn muốn liên kết tới trang web Windows SharePoint Services.

**Bước 5:** Click chọn Create hoặc Download.

Access tạo một DataBase mới từ Template và mở nó.

Bạn có thể Download thêm các mẫu Template trực tiếp từ Office Online.

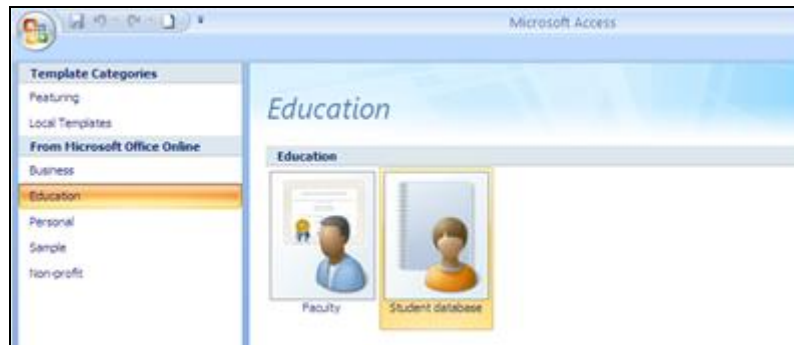




### **Tạo DataBase từ Microsoft Office Online Template**

**Bước 1:** Khởi động Access từ menu Start hoặc từ Shortcut. Trang Getting Started with Microsoft Office Access xuất hiện.

**Bước 2:** Trên trang Getting Started with Microsoft Office Access, trong ô Template Categories bên phía trái, Click vào một loại và sau đó, khi bản mẫu xuất hiện trong danh mục, Click chọn một loại cần tạo.



**Bước 3:** Trong mục File Name nhập vào tên File DataBase.

**Bước 4:** Check vào mục Create and link your database to a Windows SharePoint Services site nếu muốn kết nối DataBase của bạn tới Windows SharePoint Services.

**Bước 5:** Cuối cùng Click Download để tải về máy.

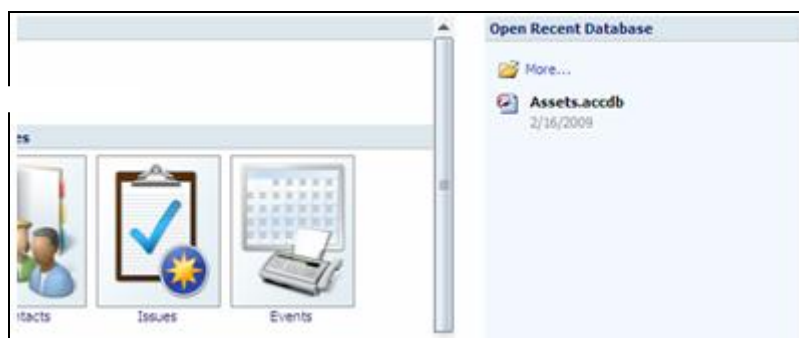
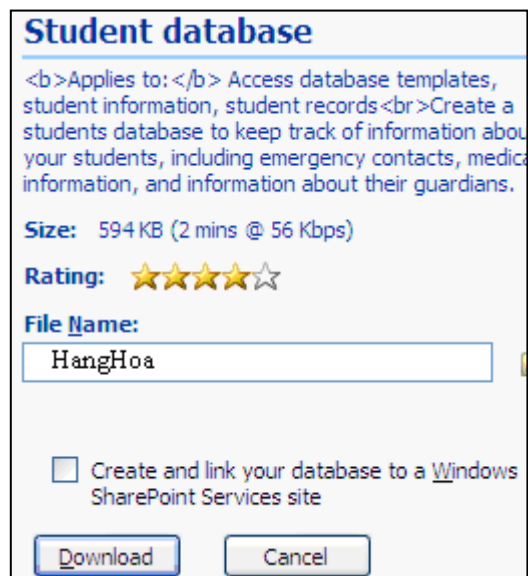
Access tự động Download mẫu về máy tính của bạn và tạo một DataBase mới dựa trên Template, sau đó mở DataBase này.

Khi bạn ở (hoặc tạo và mở) một DataBase, Access thêm tên File và đường dẫn của DataBase đó vào danh sách văn bản được sử dụng. Đây là danh sách được hiển thị trên trang Getting Started with Microsoft Office Access nên bạn có thể dễ dàng mở lại những DataBase bạn đã sử dụng gần đây nhất một cách dễ dàng.

### **Mở một DataBase sử dụng gần đây nhất**

**Bước 1:** Khởi động Microsoft Access 2007

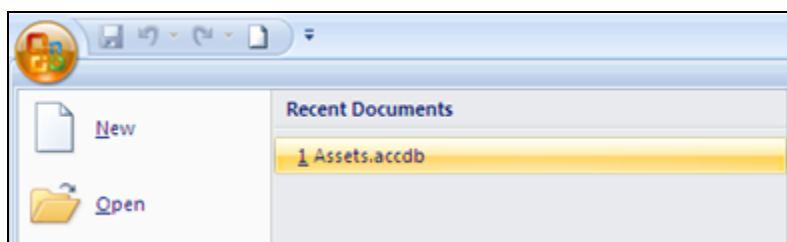
**Bước 2:** Trên trang Getting Started with Microsoft Office Access phía dưới mục Open Recent DataBase Click chọn DataBase mà bạn muốn mở.



## **Mở một DataBase sử dụng nút Microsoft Office Access**

**Bước 1:** Khởi động Access

**Bước 2:** Click chọn Microsoft Office Button , sau đó Click chọn DataBase mà bạn cần mở, nếu như nó xuất hiện ở Pane bên phải.



Hoặc Click chọn Open, hộp thoại xuất hiện chọn đến DataBase cần mở rồi Click Open. Như vậy bạn đã mở được một cơ sở dữ liệu có sẵn.

## **2. Bắt đầu làm việc với cơ sở dữ liệu**

### **2.1. Lần đầu tiên sử dụng Access**

Office Access 2007 với những chức năng mới đã làm cho quá trình tạo ra một cơ sở dữ liệu dễ dàng hơn bao giờ hết. Những người dùng đã tạo ra những cơ sở dữ liệu trong Access ở các phiên bản trước sẽ đánh giá rất cao những cải tiến mới này và nhờ chúng mà quá trình tạo ra các đối tượng được tăng tốc đáng kể.



Khi bạn bắt đầu với Microsoft Access 2007, màn hình đầu tiên xuất hiện là trang Getting Started with Microsoft Office Access (trừ trường hợp bạn khởi động Access bằng cách Double Click chuột vào một file Database Access cụ thể nào đó, khi đó Access sẽ khởi động tùy vào từng trường hợp cụ thể).

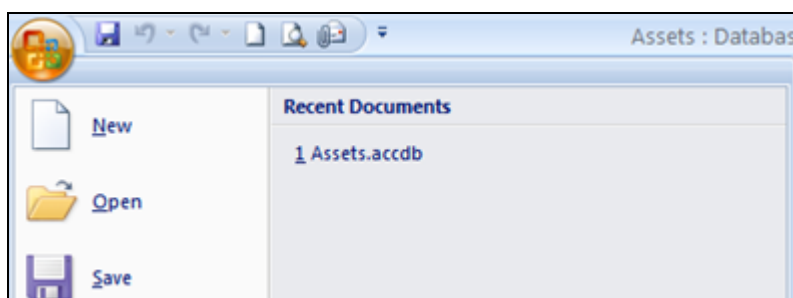
Trang Getting Started with Microsoft Office Access là điểm khởi đầu để giúp bạn có thể tạo ra một DataBase mới, mở một DataBase đã tồn tại, hoặc hiển thị nội dung đặc trưng từ Microsoft Office Online.

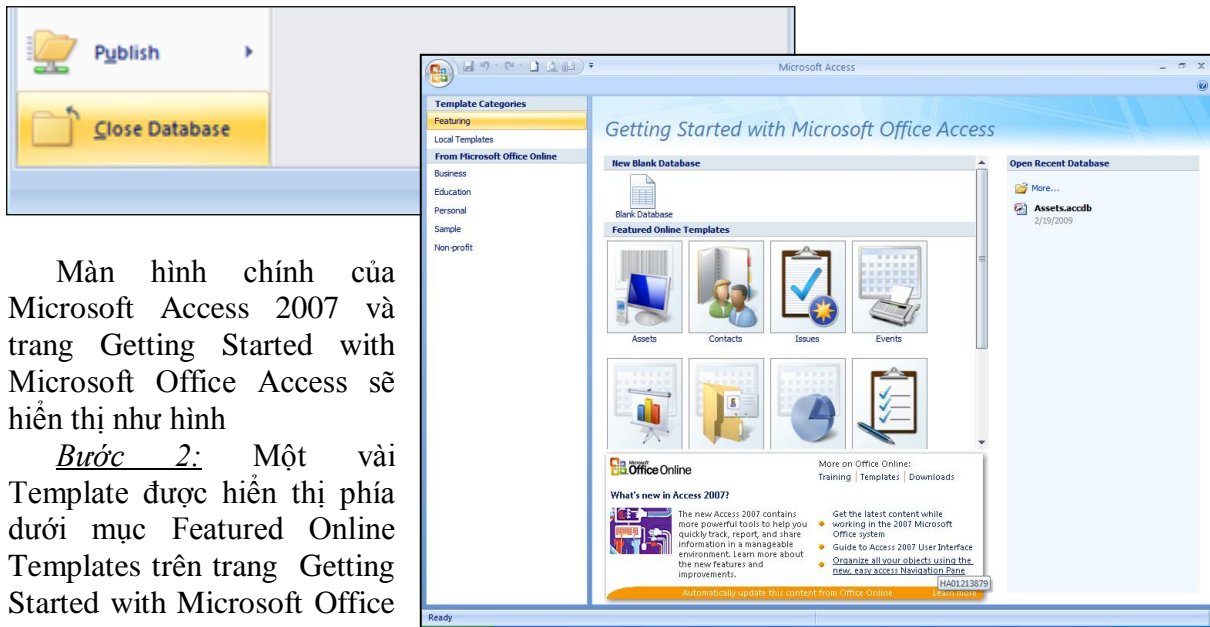
### **2.2. Tạo một DataBase sử dụng Template.**

Access cung cấp cho bạn rất nhiều mẫu Template mà bạn có thể sử dụng để đẩy nhanh quá trình tạo ra những Database của riêng mình. Một Template là một DataBase đã sẵn sàng sử dụng mà nội dung của nó bao gồm các Table, Query, Form, Report cần cho một công việc cụ thể.

Ví dụ, có những mẫu Template mà bạn có thể sử dụng để quản lý danh bạ, hoặc theo dõi chi phí, ... Các Database Template có thể sử dụng với những gì nó đã có hoặc bạn có thể tùy biến cho chúng phù hợp hơn với nhu cầu mà bạn cần.

**Bước 1:** Khởi động Access nếu bạn chưa mở. Trong trường hợp bạn đang mở một DataBase bạn Click chuột vào nút Microsoft Office Button  và chọn Close Database  để có thể hiển thị trang Getting Started with Microsoft Office Access.

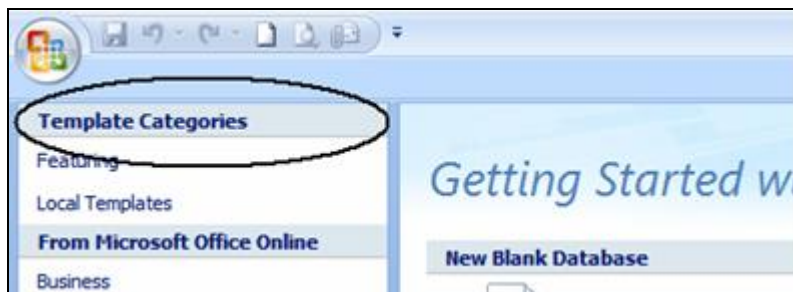




Màn hình chính của Microsoft Access 2007 và trang Getting Started with Microsoft Office Access sẽ hiển thị như hình

**Bước 2:** Một vài Template được hiển thị phía dưới mục Featured Online Templates trên trang Getting Started with Microsoft Office Access

Để có nhiều hơn bạn Click chuột vào Template Categories ở bên trái của cửa sổ Access.




Ngoài ra bạn có thể tải về từ Website của Microsoft Office.

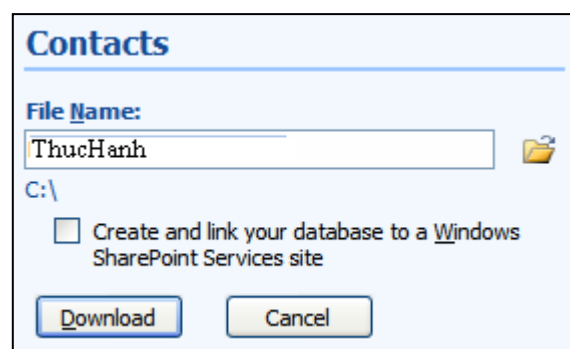


**Bước 3:** Click chuột vào Template mà bạn muốn sử dụng.

**Bước 4:** Trong Pane bên phải cửa sổ Access, Access yêu cầu bạn nhập vào tên cho DataBase trong hộp File Name.

Bạn cũng có thể sửa lại đường dẫn mặc định thành đường dẫn khác bằng cách Click vào nút  (Browse for a location ...).

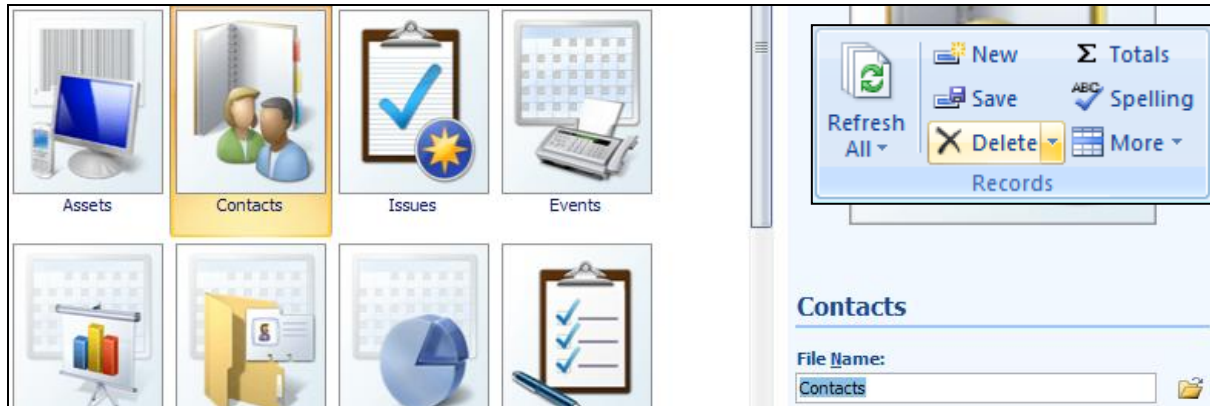
Tùy chọn Create your database and link it to a Microsoft Windows SharePoint Services site cho phép bạn liên kết trực tiếp DataBase



tới Website của Microsoft Windows SharePoint Services.

**Bước 5:** Click Create (hoặc Download, cho trường hợp bạn tải các Template online) để hoàn tất

Với các bước trên Access sẽ tạo và mở DataBase. Một Form được mở và bạn đã có thể bắt đầu nhập liệu cho DataBase.





**Chú ý:** Nếu nội dung Template có dữ liệu mẫu bạn có thể xóa chúng đi theo các bước dưới đây:

- Click chuột vào dòng đầu tiên của bản ghi muốn xóa.
- Trong Tab Home tìm tới phần Record Click chọn Delete.

**Bước 6:** Để bắt đầu nhập dữ liệu, Click chuột vào dòng trống đầu tiên trên Form và sau đó có thể bắt đầu nhập dữ liệu. Sau khi đã nhập vào một vào bản ghi bạn có thể sử dụng Pane Navigation để xem liệu có một Form hoặc một Report khác mà bạn muốn sử dụng.

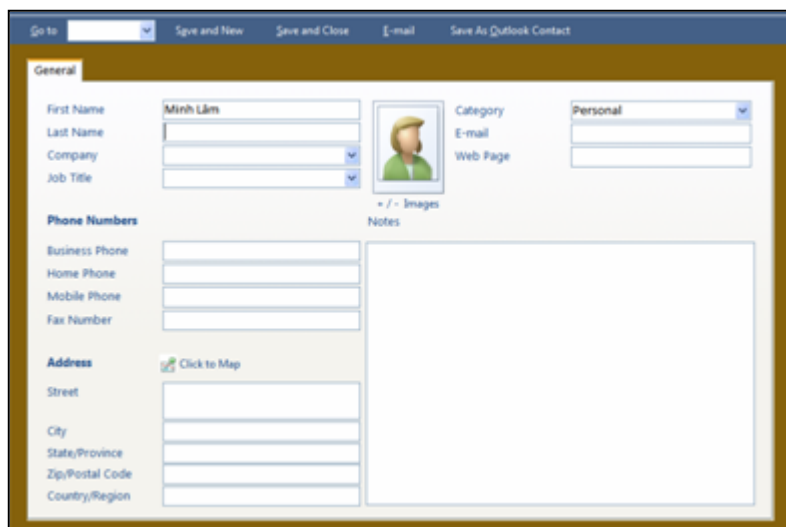
### 2.3. Tải các Template từ Microsoft Office Online

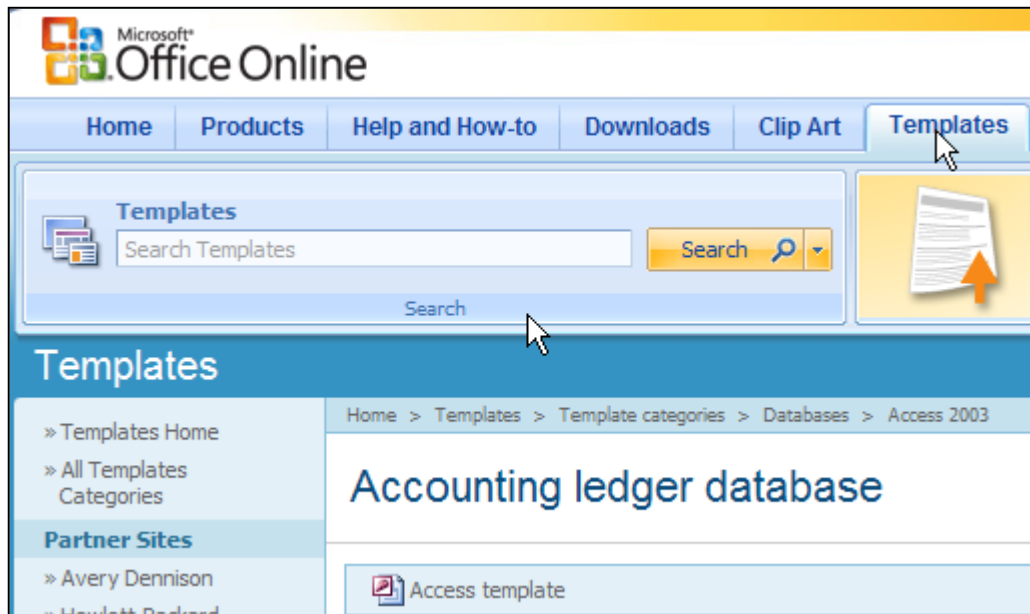
Nếu bạn không thể tìm thấy một mẫu phù hợp với yêu cầu của mình trên trang Getting Started with Microsoft Office Access, bạn có thể khám phá trên Web Office Online ở đó sẽ có rất nhiều lựa chọn cho bạn.

**Bước 1:** Khởi động Access nếu bạn chưa mở. Trong trường hợp bạn đang mở một DataBase bạn Click chuột vào nút Microsoft Office Button  và chọn Close Database  để có thể hiển thị trang Getting Started with Microsoft Office Access.

**Bước 2:** Gần phía cuối của trang Getting Started with Microsoft Office Access và dưới phần More on Office Online, Click chọn Templates.

Website Office Online sẽ xuất hiện hiện trong một trình duyệt mới.





**Bước 3:** Sử dụng các công cụ tìm kiếm trên trang web của Office Online để tìm và tải về các mẫu mà bạn muốn.

**Bước 4:** Nếu bạn tải về một mẫu mới, DataBase đó sẽ được lưu trữ ở một trong số các thư mục sau đây:

- Với Windows Vista: c:\Users\user name\Documents
- Với Microsoft Windows Server 2003 hoặc Microsoft Windows XP: c:\Documents and Settings\user name\My Documents.

Sau đó khi cần làm việc với các cơ sở dữ liệu này, bạn có thể sử dụng Access hoặc Windows Explorer để mở nó ra ở đường dẫn đã lưu trước đó.

#### 2.4. Tạo DataBase từ đầu

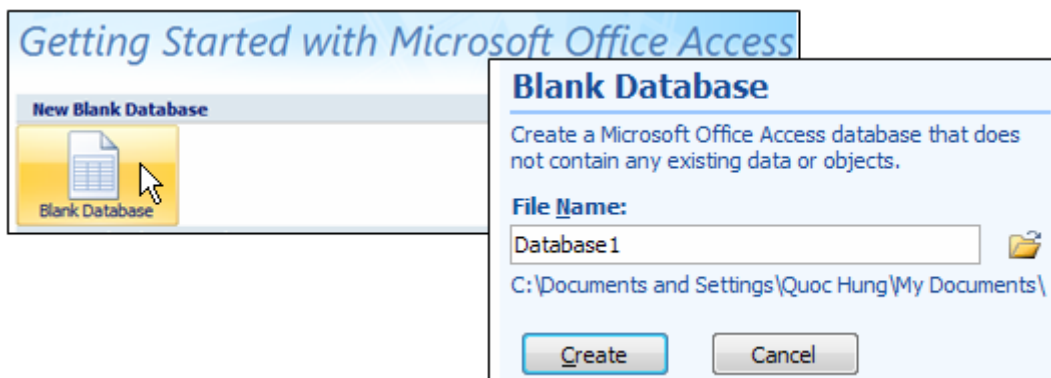
Nếu không có một Template nào phù hợp với yêu cầu của bạn, hoặc bạn có dữ liệu trong một chương trình khác mà bạn muốn Import vào Access, bạn hãy chắc chắn rằng nó là tốt hơn là việc ra một cơ sở dữ liệu từ đầu. Đối với hầu hết các ứng dụng, điều này thường liên quan đến một trong hai vấn đề sau đây:

- Tạo các bảng mới, sau đó nhập dữ liệu, hoặc Import dữ liệu vào các bảng đó.
- Import dữ liệu từ một nguồn khác và các bảng sẽ được tạo ra trong quá trình đó.

#### 2.5. Tạo một DataBase rỗng

**Bước 1:** Khởi động Access

**Bước 2:** Trên trang Getting Started with Microsoft Office Access Click chọn Blank Database.




**Bước 3:** Trong Pane Blank DataBase hãy nhập vào tên cho DataBase trong mục File Name, nếu bạn không điền phần mở rộng cho File thì Access sẽ tự động điền thêm.

Mặc định chương trình sẽ lưu File tại đường dẫn:

- Windows Vista: c:\Users\user name\Documents

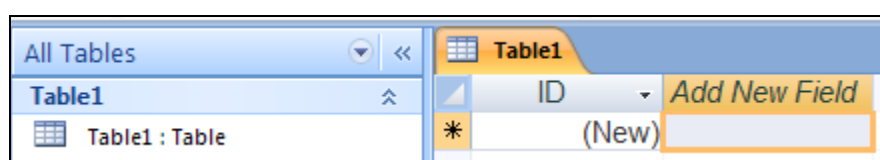
- Microsoft Windows Server 2003 hoặc Microsoft Windows XP: c:\Documents and Settings\user name\My Documents

Để thay đổi đường dẫn chứa tập tin, Click chọn nút Browse  bên cạnh vào ô File Name, một hộp thoại sẽ xuất hiện cho phép bạn chọn đường dẫn mới, Click Ok sau khi đã chọn xong.

**Bước 4:** Click Create để hoàn tất.

Sau các bước trên Access sẽ tạo ra một DataBase mới và mở sẵn nó với một bảng trắng ở chế độ DataSheet View cho bạn thiết kế.


**Bước 5:** Vị trí con trỏ chuột của Access đặt tới ô trắng đầu tiên trên cột Add New Field.



Để thêm dữ liệu bạn có thể nhập vào hoặc Copy dữ liệu từ những nguồn khác.

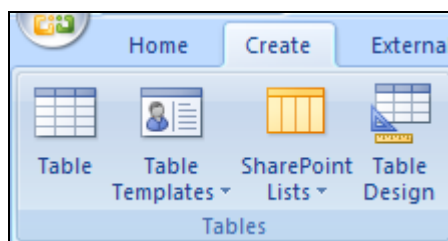
**Chú ý:** Việc nhập dữ liệu ở chế độ Datasheet View được thiết kế tương tự như nhập dữ liệu trong Wooksheet Excel. Những hạn chế chủ yếu là dữ liệu nhập vào phải được nhập ở các dòng, cột liền nhau, bắt đầu ở góc trên bên trái của Datasheet. Bạn không nên cố gắng để định dạng dữ liệu nhập vào có các hàng hoặc cột trắng, bởi vì làm như vậy sẽ gây lãng phí bộ nhớ nhớ trong các bảng.

Cấu trúc của bảng dữ liệu được tạo trong khi bạn nhập dữ liệu vào nó. Tại một thời điểm bất kỳ bạn hoàn toàn có thể thêm một cột mới vào bảng, khi đó một trường mới sẽ được định nghĩa. Access thiết lập kiểu dữ liệu cho trường mới dựa vào kiểu của dữ liệu mà bạn nhập vào. Ví dụ, nếu bạn có một cột mà trong đó bạn chỉ nhập giá trị ngày tháng, Access sẽ thiết lập kiểu dữ liệu của trường đó là Data/Time. Sau này giả sử bạn cố gắng nhập vào trường này một giá trị không phải ngày tháng (chẳng hạn như là tên hoặc một số điện thoại nào đó), Access sẽ hiển thị thông báo rằng giá trị của bạn không hợp lệ. Vì vậy, khi có thể, bạn nên phân check lại bảng dữ liệu của mình sao cho mỗi trường chỉ chứa cùng một loại dữ liệu, cho dù đó là văn bản, ngày tháng, số điện thoại, .... Điều này làm cho việc xây dựng các Query, Form, Report (lấy dữ liệu từ những bảng này) sau này sẽ trở nên dễ dàng và thuận tiện hơn rất nhiều.

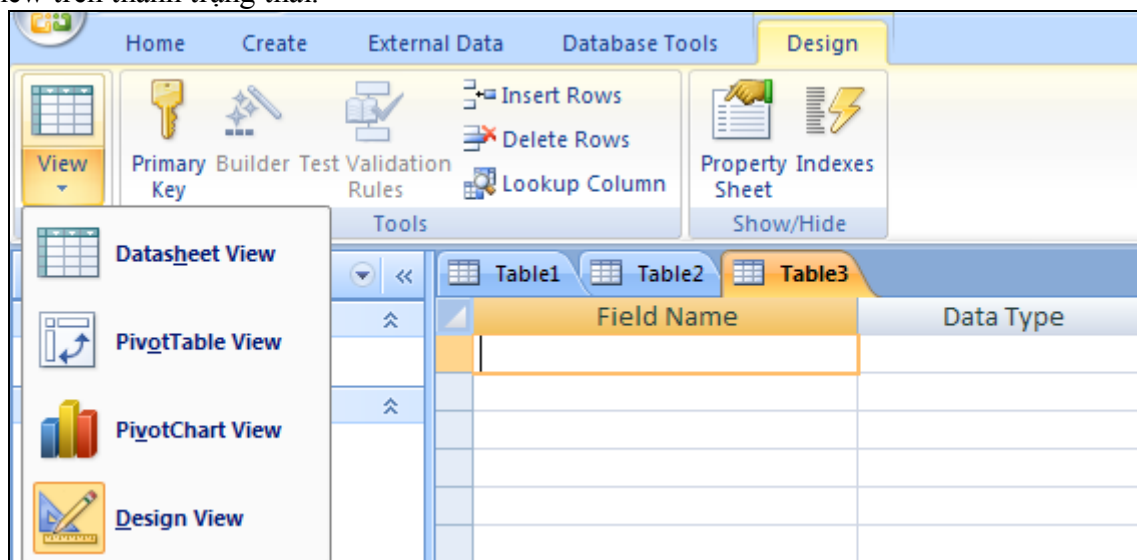
Nếu bạn không quan tâm đến việc nhập dữ liệu vào thời điểm này, hãy Click vào nút Close 

## 2.6. Thêm một bảng vào Database

Bạn có thể thêm một bảng mới vào Database đã tồn tại bằng cách sử dụng công cụ trong nhóm Tables trên Tab Create.




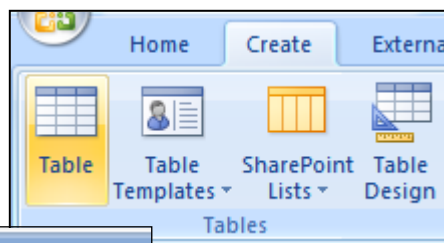
Click chuột vào nút Table để tạo một bảng trắng mới ở chế độ Datasheet View. Sau đó bạn hoàn toàn có thể nhập ngay dữ liệu vào bảng và Access sẽ tạo ra cấu trúc bảng cho bạn khi bạn thoát ra, hoặc bạn có thể sử dụng Design View để tạo ra cấu trúc bảng trước sau đó chuyển về chế độ Datasheet View để nhập dữ liệu. Bất kể bạn bắt đầu làm việc với bảng ở chế độ nào bạn đều có thể chuyển sang những chế độ hiển thị khác bằng cách sử dụng nút View trên thanh trạng thái.



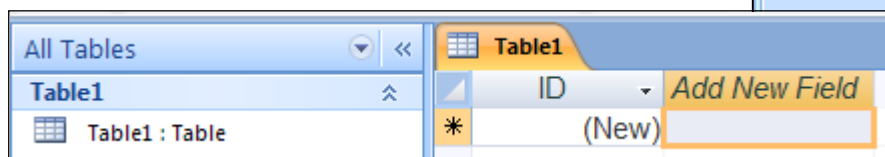
***Thêm một bảng, bắt đầu ở chế độ Datasheet view :***

Trong Datasheet View, bạn có thể nhập ngay dữ liệu, và Access sẽ xây dựng cấu trúc cho bảng dựa trên những dữ liệu đó. Tên trường được đặt lần lượt (Field1, Field2, ...), và các kiểu dữ liệu được Access tự động thiết lập tùy thuộc vào loại dữ liệu mà bạn nhập vào các trường.

**Bước 1:** Trên Tab Create trong nhóm Tables Click chọn Table .

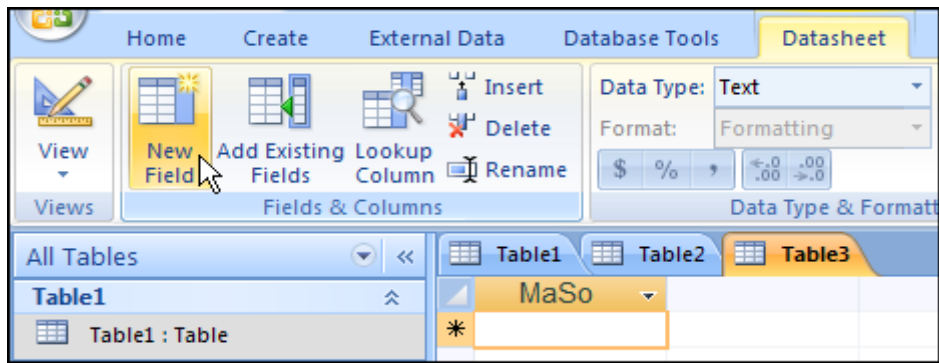


**Bước 2:** Sau bước này Access sẽ tạo ra một bảng và đặt vị trí con trỏ tại ô trắng đầu tiên của cột Add New Field.

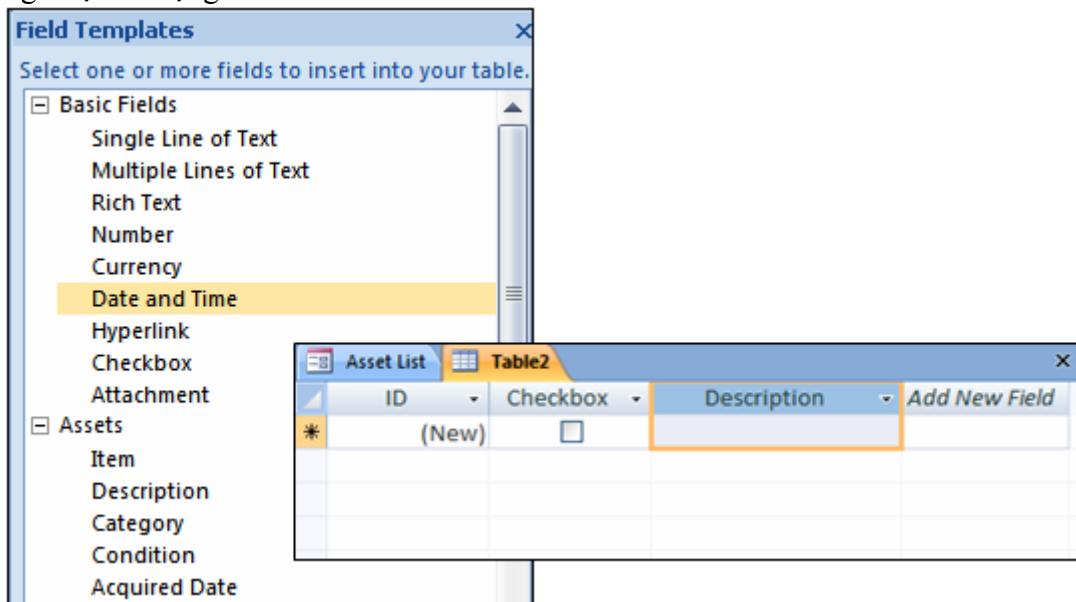


**Chú ý:** Nếu bạn không nhìn thấy cột có tên là Add New Field, có thể bạn vẫn đang ở chế độ Design View chứ không phải DataSheet View. Để chuyển qua chế độ hiển thị này bạn Double Click chuột vào bảng đó trên thanh Pane Navigation. Access sẽ nhắc bạn nhập vào tên cho bảng và chuyển sang chế độ hiển thị Datasheet view.

**Bước 3:** Trên Tab DataSheet, trong nhóm Field & Columns Click chọn New Field .



Access hiển thị Pane Field Templates, trong pane này chứa danh sách các kiểu trường thường được sử dụng.



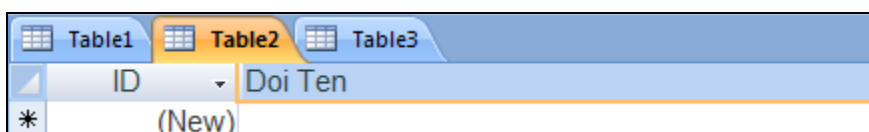
Nếu bạn kéo thả một trong số chúng lên DataSheet của bạn, Access sẽ tự động thêm trường đó vào bảng với tên là tên của kiểu dữ liệu đối với các trường cơ bản và tên như trường gốc nếu nó được lấy từ các bảng khác. Ngoài ra Access cũng thiết lập những thuộc tính phù hợp cho từng trường. Bạn cần phải kéo và thả trường vào phần chứa dữ liệu trên DataSheet.

Như hình ảnh trên các bạn có thể thấy trường có tên Checkbox được lấy từ Basic Field và trường Description được lấy từ một bảng có sẵn trong cơ sở dữ liệu.

**Bước 4:** Để nhập liệu vào bảng, bắt đầu gõ từ ô trắng đầu tiên hoặc bạn cũng có thể dán dữ liệu từ một nguồn dữ liệu khác.

**Chú ý:**

- Để đổi tên một cột bạn Double Click chuột vào phần tiêu đề của cột, sau đó gõ vào tên mới.




Tên trường bạn vừa đặt sẽ là nội dung hiển thị trường đó trên Pane Field List.

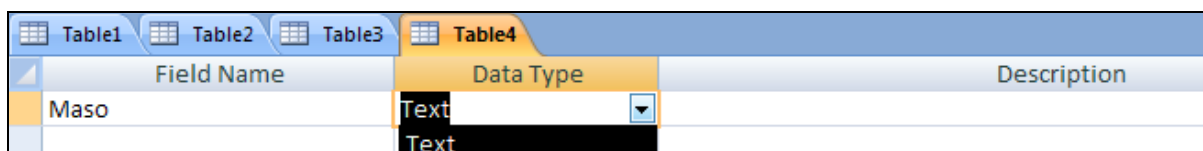


- Để di chuyển một cột, bạn hãy chọn nó bằng cách Click chuột lên phần tiêu đề của cột đó và kéo thả đến một vị trí bất kỳ trên DataSheet. Bạn cũng có thể chọn nhiều cột một lúc và di chuyển chúng đến vị trí mới.

**Chèn thêm một bảng bắt đầu ở chế độ Design View:** Ở chế độ Design View, đầu tiên bạn tạo một cấu trúc bảng mới, sau đó bạn chuyển sang chế độ DataSheet để nhập liệu, hoặc bạn có thể nhập dữ liệu vào bảng bằng những phương thức khác như dán vào hoặc chèn thêm.

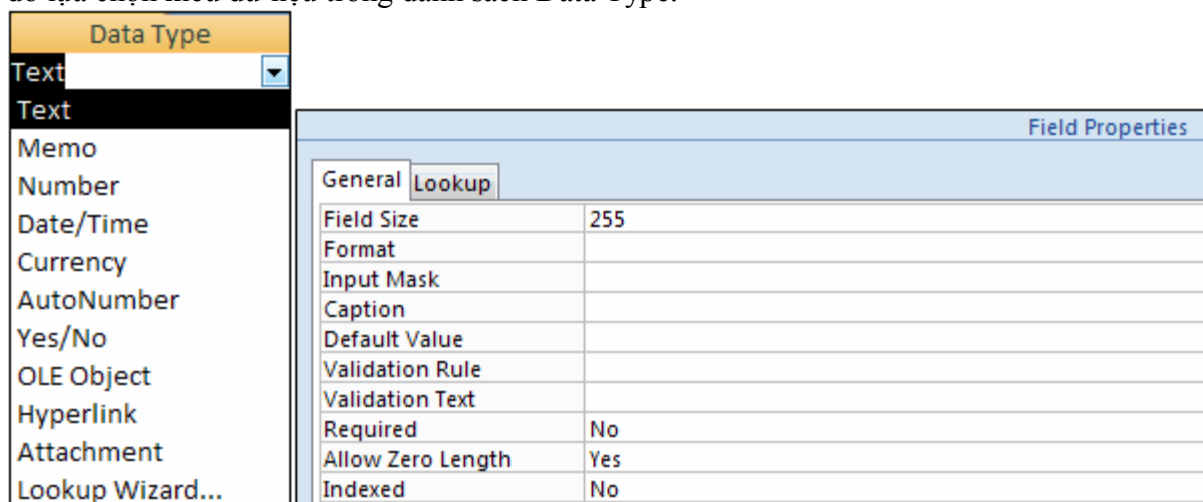
*Bước 1:* Trên Tab Create trong nhóm Tables Click chọn Table Design 

Sau bước này trên màn hình xuất hiện một bảng mới ở chế độ Design.



Field Name	Data Type	Description
Maso	Text	
	Text	

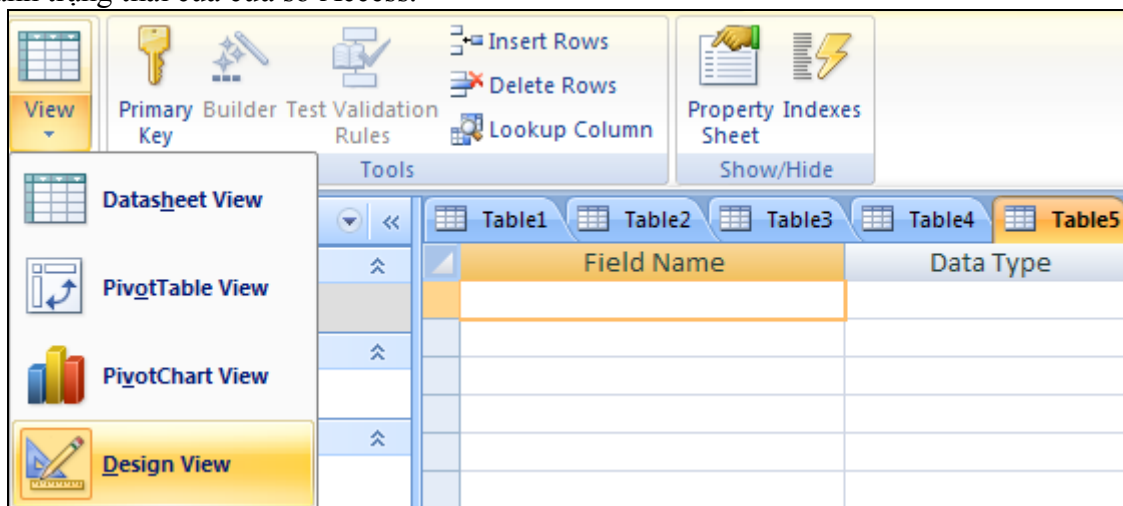
*Bước 2:* Đối với mỗi trường trong bảng, bạn nhập tên trường trong cột Field Name, sau đó lựa chọn kiểu dữ liệu trong danh sách Data Type.



Field Properties	
General	Lookup
Field Size	255
Format	
Input Mask	
Caption	
Default Value	
Validation Rule	
Validation Text	
Required	No
Allow Zero Length	Yes
Indexed	No

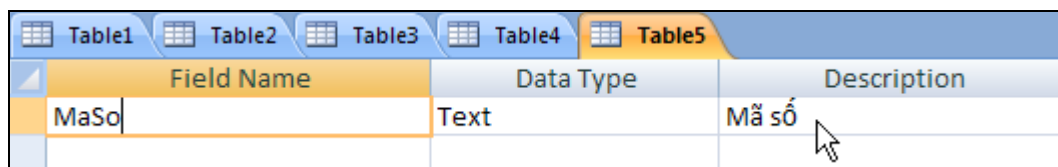
*Chú ý:*

Nếu bạn không nhìn thấy cột Field Name và cột Data Type, có thể bạn đang ở chế độ DataSheet thay vì chế độ hiển thị Design View, hãy Click chuột vào nút Design View trên thanh trạng thái của cửa sổ Access.




Access sẽ đưa ra thông báo yêu cầu bạn nhập tên mới cho bảng và sau đó tự động chuyển về chế độ Design View.

**Bước 3:** Nếu muốn, bạn có thể nhập vào thông tin để diễn giải cho các trường ở cột Description. Những thông tin này sẽ hiển thị trên thanh trạng thái khi bạn nhập dữ liệu vào trường đó. Và bạn cũng có thể sử dụng nội dung trên thanh trạng thái cho bất cứ điều khiển nào trên Form hoặc Report mà bạn tạo ra bằng cách kéo thả trường từ Pane Field List.



Field Name	Data Type	Description
MaSo	Text	Mã số

**Bước 4:** Sau khi đã thêm vào tất cả các trường cần thiết, bạn hãy lưu bảng lại bằng cách Click chọn nút Microsoft Office Button  rồi chọn Save hoặc Click tổ hợp phím Ctrl+S.

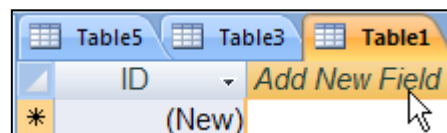
**Bước 5:** Sau bước này bạn có thể nhập dữ liệu vào bảng bất cứ khi nào bạn chuyển sang chế độ hiển thị Datasheet View.



MaSo	HoSV	TenSV
SV001	Hàng Phi	

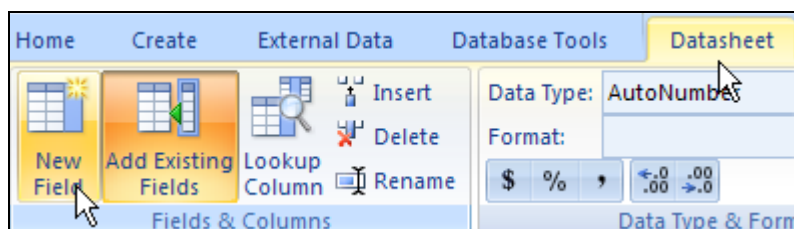
Ở chế độ này bạn di chuyển tới ô đầu tiên và nhập lần lượt dữ liệu cho từng bản ghi. Ngoài ra bạn cũng có thể dán dữ liệu vào bảng từ một nguồn dữ liệu khác.

Nếu sau khi đã nhập một số dữ liệu vào bảng, bạn lại chợt nhớ cần phải thêm một hoặc một số trường nữa mới phù hợp. Thay vì việc chuyển về chế độ Design View như lúc ban đầu thiết kế bạn gõ thẳng dữ liệu vào cột Add New Field ở chế độ hiển thị Datasheet View.



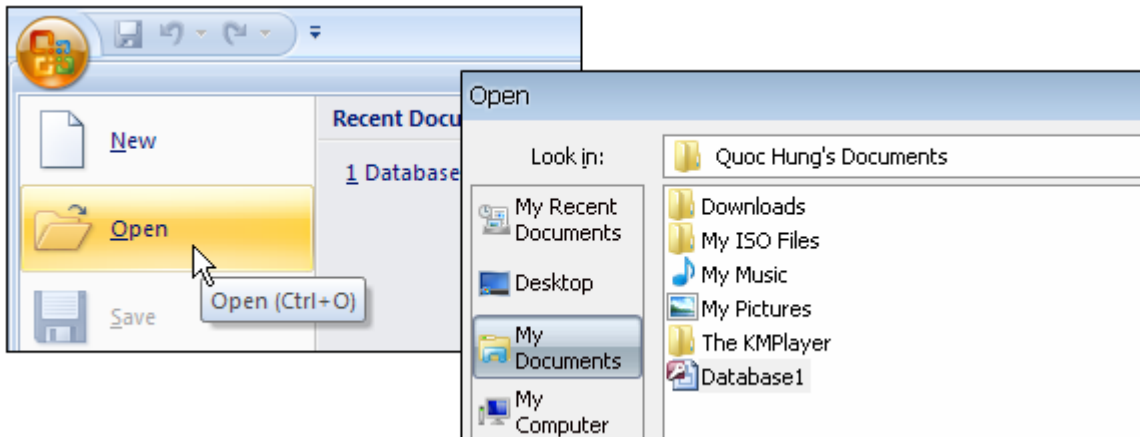
ID	Add New Field
(New)	

Hoặc bạn cũng có thể thêm một trường mới bằng cách Click chọn nút lệnh New Field trong nhóm Fields & Column của Tab Datasheet.



## 2.7. Mở một Database đã tồn tại

**Bước 1:** Click chọn nút Microsoft Office Button , sau đó chọn Open.

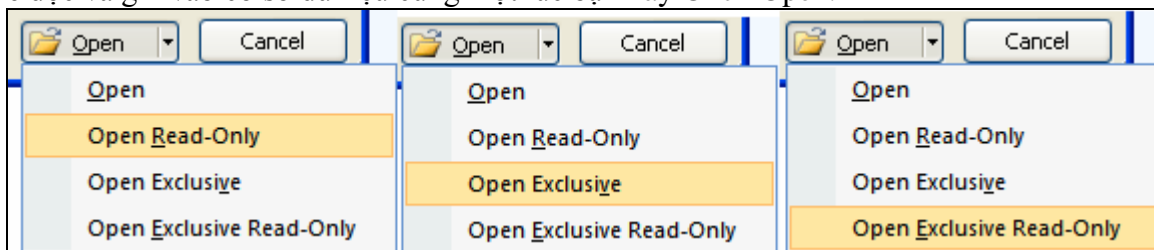


**Bước 2:** Tìm tới thư mục có chứa Database rồi Click chọn DataBase cần mở.

**Bước 3:** Khi đã tìm thấy và Click chọn Database cần mở bạn có thể làm theo một trong các cách sau đây:

- Mở Database ở chế độ mặc định của chương trình: Bạn Double Click chuột vào Database đó.

- Mở Database cho phép chia sẻ nhiều người dùng (chia sẻ nhiều người dùng: Là một Database cho phép nhiều hơn một người sử dụng có thể truy cập vào sửa đổi, nhập liệu tại cùng một thời điểm) trong một môi trường: Cho phép cả bạn và những người dùng khác có thể đọc và ghi vào cơ sở dữ liệu cùng một lúc bạn hãy Click Open.



- Mở Database ở chế độ Read-only (chỉ đọc): Bạn có thể xem dữ liệu nhưng không thể chỉnh sửa được nó hãy Click vào mũi tên chỉ xuống dọc theo nút Open/Open Read-only.

- Mở Database được phép Exclusive (độc quyền): Không có một người nào khác có thể mở Database này trong khi bạn đang mở nó. Để mở theo chế độ này bạn Click mũi tên dọc theo nút Open chọn Open Exclusive.

- Mở Database ở chế độ Exclusive Read-only (Độc quyền và chỉ đọc): Ở chế độ này khi bạn mở một Database bạn chỉ có thể đọc dữ liệu và những người khác thì không thể mở lại nó. Mở ở chế độ này bạn chọn mũi tên dọc theo nút Open/Open Exclusive Read-only.

## 2.8. Nếu không tìm thấy Database cần mở

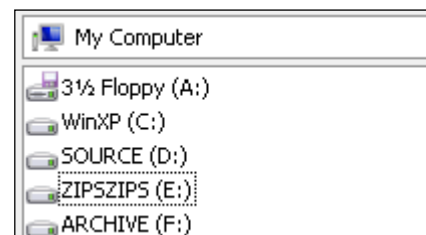
**Bước 1:** Trong hộp thoại Open, Click chuột chọn biểu tượng đường tắt My Computer hoặc trong hộp Look in Click chọn My Computer.

**Bước 2:** Trong danh sách ổ đĩa, Click chuột phải vào ổ đĩa mà bạn cho là có chứa Database mà bạn đang tìm kiếm chọn Search.

**Bước 3:** Cửa sổ tìm kiếm của Window xuất hiện. Nhập vào điều kiện tìm kiếm và sau đó nhấn Enter để tìm cơ sở dữ liệu.

**Bước 4:** Nếu tìm thấy, để mở bạn hãy Double Click chuột vào dữ liệu đó trong hộp thoại Search Results.

**Bước 5:** Vì tìm kiếm đã được bắt đầu từ hộp thoại Open, nên bạn phải Click chuột vào Cancel để tắt hộp thoại Open trước khi mở Database.



**Search by any or all of the criteria below.**

All or part of the file name:

A word or phrase in the file:

Look in:

**When was it modified?**

**What size is it?**

**More advanced options**

*Chú ý:* Bạn có thể mở trực tiếp một file dữ liệu ngoài định dạng một tệp tin (như là dBASE, Paradox, Microsoft Exchange, or Excel). Bạn cũng có thể trực tiếp mở bất cứ dữ liệu nguồn ODBC nào, như là Microsoft SQL Server hoặc Microsoft FoxPro. Access sẽ tự động tạo ra một cơ sở dữ liệu mới trong cùng một thư mục như là các tệp tin dữ liệu, và thêm những liên kết cho mỗi bảng trong cơ sở dữ liệu bên ngoài.

## 2.9. Mở Database làm việc gần đây nhất

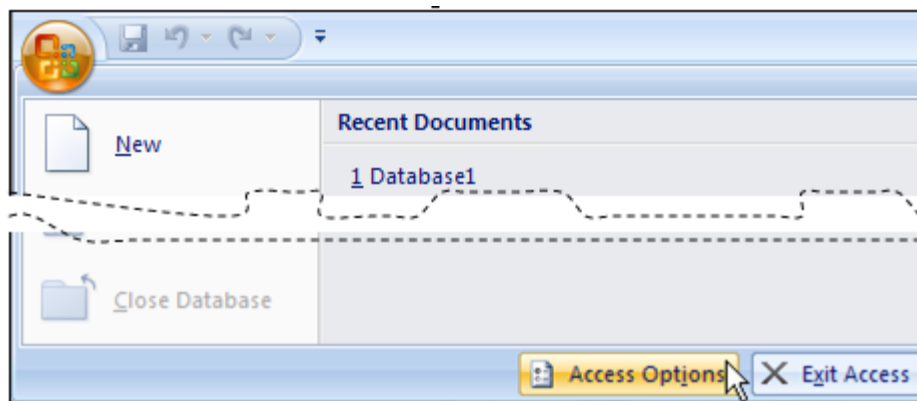
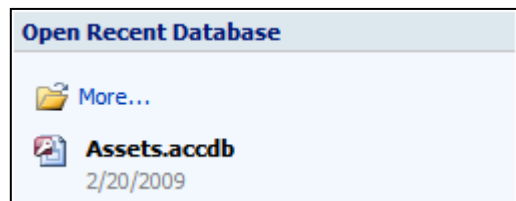
Bạn hãy Click chuột vào tên một File trong danh sách Open Recent Database trên trang Getting Started with Microsoft Office Access.

Để mở một trong những Database mà bạn làm gần đây nhất, bạn hãy Click chuột vào tên của Database đó trong danh sách Open Recent Database

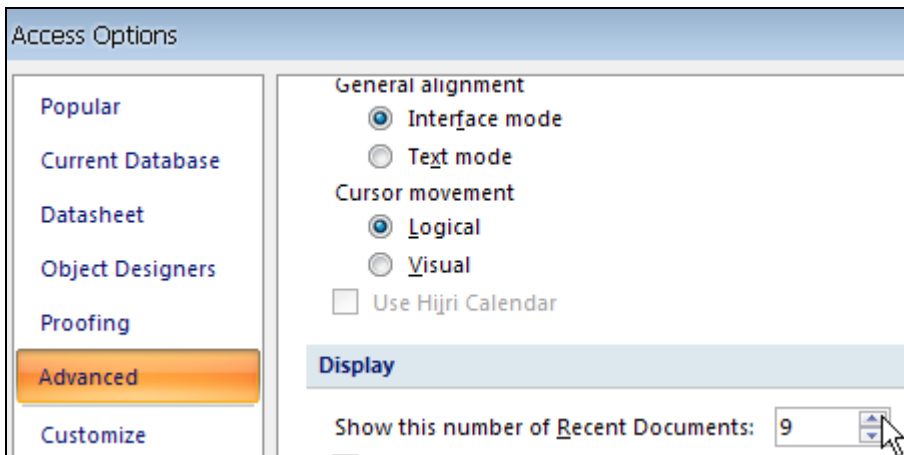
trên trang Getting Started with Microsoft Office Access. Access sẽ mở Database và thiết lập những tùy chọn như lần cuối cùng bạn làm việc với nó. Nếu trong danh sách các file được sử dụng trong thời gian gần đây không được hiển thị, bạn hãy Click vào nút Microsoft Office




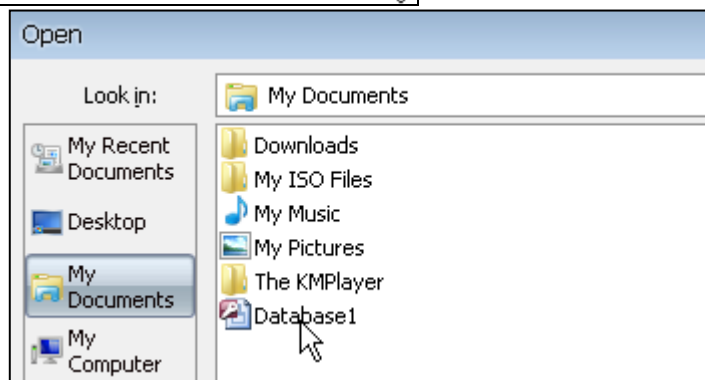
và sau đó Click chọn **Access Options**



Hộp thoại Access Options xuất hiện, Click vào mục Advanced, trong cửa sổ ở giữa hộp thoại dưới mục Display chọn vào hộp TextBox Recently used file list nhập vào đó số số file cần lưu lại trong danh sách các tệp tin đã mở tối đa là 9.



Nếu bạn đang mở một Database bằng cách Click chọn nút Microsoft Office Button  chọn nút Open. Bạn có thể hiển thị đường tắt tới danh sách Database mà bạn đã mở trước đó bằng cách Click chuột vào mục My Recent Documents trong hộp thoại Open.

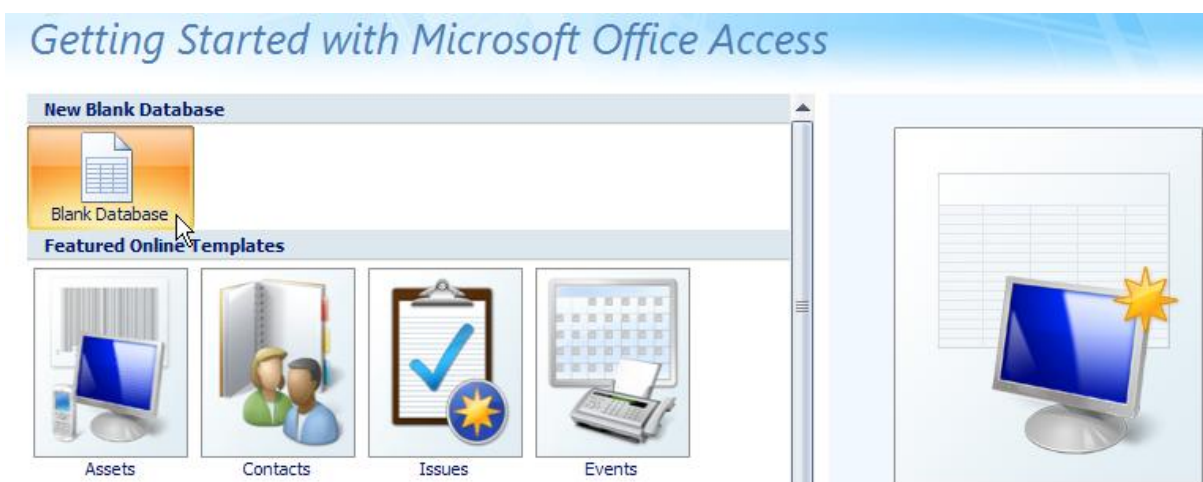


### 3. Có gì mới trong Microsoft Office Access 2007

Microsoft Office Access 2007 cung cấp một bộ công cụ mạnh mẽ giúp bạn nhanh chóng bắt đầu theo dõi, báo cáo, và chia sẻ thông tin. Bạn có thể nhanh chóng tạo ra sức hấp dẫn và các chức năng theo dõi các ứng dụng tùy biến của một trong nhiều bộ Templates dựng sẵn, chuyển đổi một cơ sở dữ liệu hiện có, hoặc tạo ra một cơ sở dữ liệu mới, và bạn có thể làm như vậy mà không cần phải trang bị quá nhiều kiến thức về cơ sở dữ liệu.

#### 3.1. Để bắt đầu nhanh hơn

Microsoft Office Access 2007 cung cấp các Templates mà bạn có thể dễ dàng tạo ra cơ sở dữ liệu, bảng biểu, và các trường. Ngoài ra còn có những công cụ hiển thị và thiết kế mới giúp bạn trong quá trình tạo Cơ sở dữ liệu mới và làm việc với dữ liệu.



**Cơ sở dữ liệu mẫu để xây dựng hoàn chỉnh các ứng dụng**

Microsoft Access 2007 cung cấp một bộ cơ sở dữ liệu bao gồm các mẫu thiết kế chuyên nghiệp giúp bạn có thể theo dõi địa chỉ liên lạc, nhiệm vụ, sự kiện, học sinh, tài sản, và nhiều kiểu dữ liệu khác. Bạn có thể sử dụng ngay lập tức hoặc cải tiến và tinh chỉnh chúng cho phù hợp với yêu cầu quản lý của mình.

Trên trang Getting Started with Microsoft Office Access xuất hiện khi khởi động ứng dụng bạn có thể dễ dàng tạo một DataBase mới bằng cách Click chọn Blank DataBase.

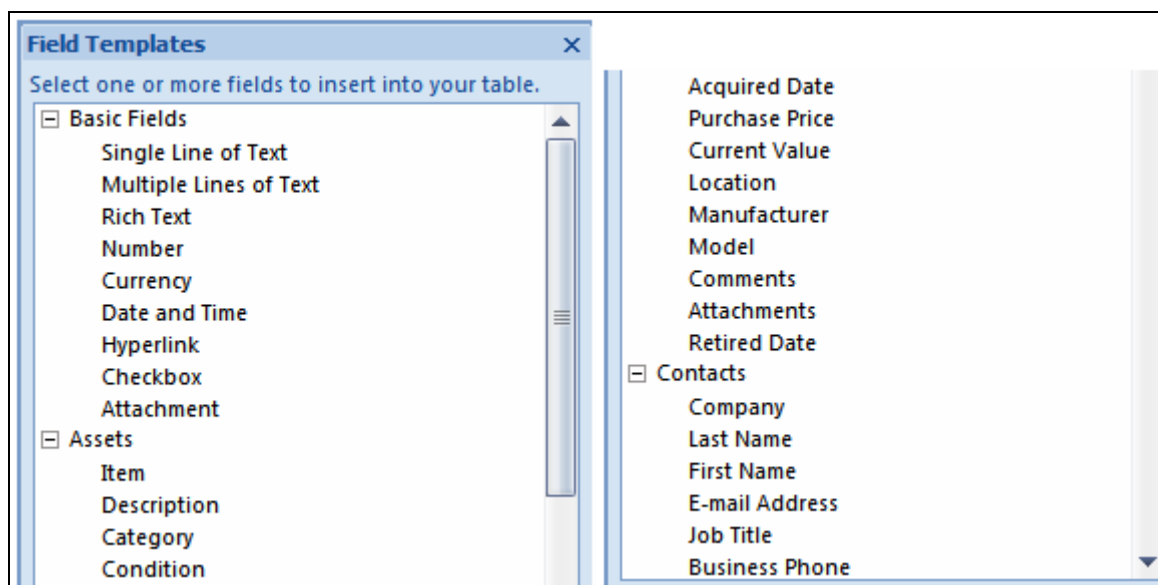
Hoặc nhanh chóng bắt đầu tạo ra cơ sở dữ liệu của bạn bằng cách mở một trong những bản mẫu nằm trong mục Featured online Templates.

Mỗi DataBase Templates là một ứng dụng tương đối hoàn chỉnh với một bài toán chung. Chúng bao gồm các Table, Form, Report, Query, Macro và các mối quan hệ giữa các bảng. Những DataBase Template này cho phép bạn có thể sử dụng chúng ngay lập tức nếu bạn cho là cấu trúc của nó đáp ứng được yêu cầu quản lý của mình. Nếu vẫn chưa đủ bạn có sử dụng, kế thừa chúng để tạo ra những cơ sở dữ liệu phù hợp hơn yêu cầu mình.

Ngoài các Template mà Microsoft Access 2007 cung cấp bạn cũng có thể kết nối với Microsoft Office Online để có thể tải về các Template khác.

### Field và Table Templates

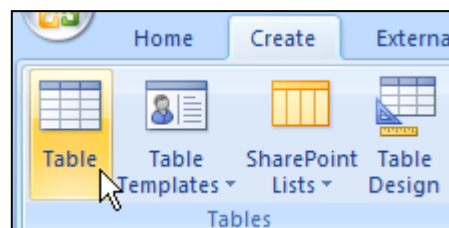
Thay vì việc bỏ rất nhiều thời gian để thiết kế các trường bạn có thể sử dụng lại các trường dựng sẵn (File Templates) với tên, kiểu dữ liệu và các thuộc tính được định nghĩa từ trước. Bạn có thể kéo thả trực tiếp các trường mà bạn cần trong Panel Fields Templates vào một Datasheet. Các trường được tạo ra dựa trên (Based on) các file XML schema (. Xsd) giúp bạn có thể thiết lập, xây dựng thành những định nghĩa chuẩn của riêng mình và chia sẻ sử dụng trong phòng hoặc nhóm làm việc.



Ngoài ra, Microsoft Office Access 2007 còn bao gồm các Table thường được sử dụng trong cơ sở dữ liệu. Ví dụ, bạn có thể sử dụng Contacts table template để thêm Table Contacts vào cơ sở dữ liệu của bạn. Bảng này bao gồm các trường như Last Name, First Name, and Address. Các thuộc tính của những trường này cũng được thiết lập sẵn để bạn có thể sử dụng ngay chúng. Còn có các bảng mẫu có sẵn khác như Tasks, Issues, Events, và Assets.

### Nâng cấp Datasheet view

Với Microsoft Access 2007 bạn có thể dễ dàng tạo một bảng dữ liệu bằng cách Click chọn Tab Create rồi chọn Table.



Microsoft Access 2007 sẽ tự động xác định kiểu dữ liệu phù hợp nhất cho mỗi trường sau khi bạn gõ xong dữ liệu và di chuyển qua nó.

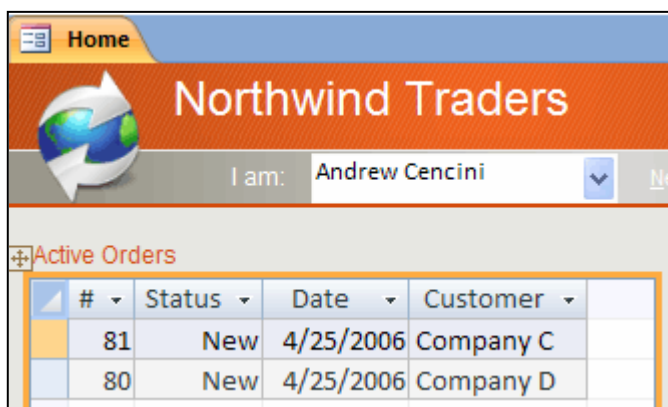
Để thêm một trường mới bạn Click chuột tới cột cuối cùng (Add new field) và thay đổi dữ liệu trên nó.

Và nếu bạn cần thay đổi kiểu dữ liệu hoặc định dạng kiểu dữ liệu hiển thị của một trường mới hay một trường đã có bạn có thể sử dụng các nút chức năng trên thanh Ribbon. Nó là một phần của giao diện người dùng Microsoft Office fluent. Bạn cũng có thể dán một bảng dữ liệu từ Microsoft Excel vào một Datasheet mới trong Microsoft Access 2007, bằng cách tạo trước một cấu trúc các trường tương ứng với bảng dữ liệu mà bạn cần dán vào.

### Layout View mới giúp cải thiện tốc độ thiết kế Forms và Report.

Sử dụng Layout View để thực hiện các thay đổi thiết kế trong khi hiển thị dữ liệu trong một Form hoặc một Report.

Ví dụ, bạn có thể thêm vào một trường trên lưới thiết kế bằng cách kéo thả từ Field list pane, hoặc bạn có thể thay đổi các thuộc bằng cách sử dụng thuộc tính trên Sheet. Layout View cung cấp một cách bố trí mới cho các Control đó là stacked layout (Bố trí dạng ngăn xếp) và tabular layout (Bố trí dạng bảng)-nhóm các Controls mà bạn có thể di chuyển và thay đổi kích thước như một đơn vị để bạn có thể dễ dàng sắp xếp lại các trường, cột, hàng, hoặc toàn bộ Layout. Bạn cũng có thể loại bỏ một trường hoặc thêm vào định dạng một cách dễ dàng trong Layout View. Chế độ Design View cũng tương tự như các phiên bản trước giúp bạn có thể biết thêm chi tiết về các Control, ở chế độ này cũng được hỗ trợ Layout dạng stacked và tabular.



### 3.2. Microsoft Office fluent

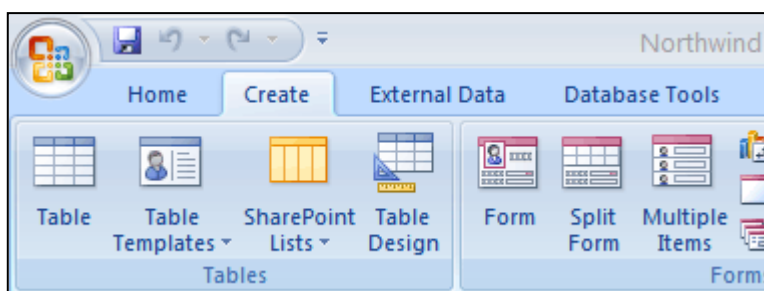
Những kết quả mới theo định hướng giao diện người dùng - Microsoft Office fluent giao diện người sử dụng - giúp bạn dễ dàng làm việc với Microsoft Access 2007. Ở những phiên bản trước Lệnh và các tính năng thường được giấu kín trong các menu, thanh công cụ phức tạp “bí hiểm”. Nhưng bây giờ bạn có thể dễ dàng tìm kiếm công việc theo định hướng sử dụng trên các tab chứa các nhóm lệnh. Nhiều ô hộp thoại được thay thế bằng các lựa chọn hiển thị trên thanh công cụ sẵn sàng cho người dùng sử dụng, ngoài ra nó còn được mô tả trên các Tooltips hoặc hiển thị dưới dạng các ví dụ.

Không có bất cứ một vấn đề gì với các hoạt động mà bạn đang thực hiện trong giao diện người dùng mới - cho dù đó là tạo ra một Report hoặc nhập dữ liệu – Hãy tiếp cận với những công cụ hữu ích nhất mà chúng tôi sẽ trình bày dưới đây chắc chắn các bạn sẽ thành công trong việc nghiên cứu và sử dụng Microsoft Access 2007.

#### Microsoft Office fluent giao diện người sử dụng:

Microsoft Office fluent giao diện người dùng bao gồm các khu vực tiêu chuẩn (standard area) được gọi là Ribbon, trong đó có chứa các nhóm lệnh được tổ chức, sắp xếp theo chức năng.

Sử dụng Office fluent Ribbon để xác định vị trí các



nhóm lệnh có liên quan nhanh hơn. Ví dụ, nếu bạn cần phải tạo một Form hoặc Report, hãy sử dụng một trong những lệnh trên Tab Create. Việc thiết kế bố trí giao diện mới làm cho việc tìm kiếm lệnh, chức năng mà bạn cần trở nên nhanh chóng, thêm vào đó bạn còn có thể khám phá ra những tính năng khác mà trước đây bạn chưa hề biết tới. Các nút lệnh được đặt gần hơn trên giao diện, điều đó có nghĩa là bạn không cần phải tìm kiếm trong menu hay cố nhớ vị trí của chúng ở đâu đó.

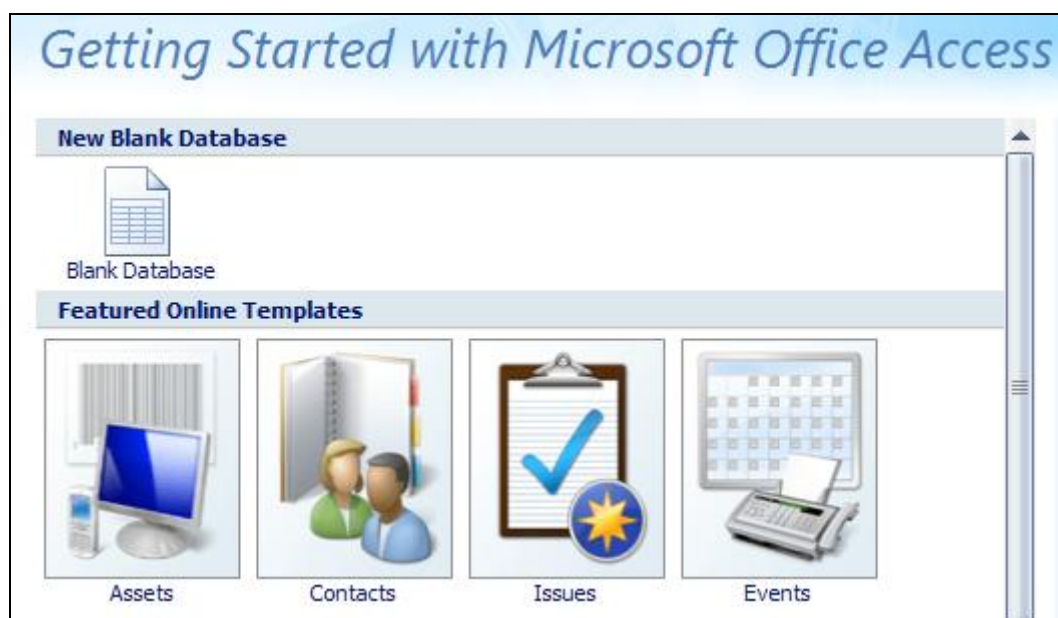
**Phím chức năng của giao diện người dùng Microsoft Office Fluent bao gồm:**

- Tab lệnh: Các Tab chứa những nút lệnh thường được sử dụng cùng nhau, nhờ vậy khi cần sử dụng bạn có thể dễ dàng tìm thấy chúng.

- Nội dung các Tab lệnh: Một Tab lệnh sẽ xuất hiện tùy thuộc vào bối cảnh làm việc của bạn trên màn hình giao diện. Một nội dung tab lệnh sẽ chứa những lệnh có nhiều khả năng để áp dụng cho những gì bạn đang làm việc.

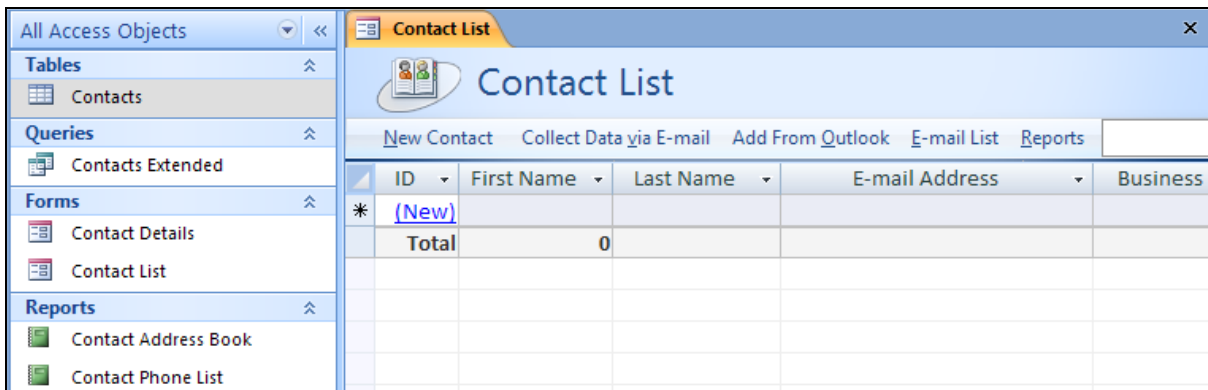
- Toolbar truy cập nhanh: Với một thanh công cụ Standard xuất hiện trên Ribbon, chỉ bằng một cái Click chuột bạn có thể truy cập tới những lệnh cần thiết chẳng hạn như Save Undo, ....

**Getting Started with Microsoft Office Access page:** Trang này cung cấp khả năng truy cập nhanh đến một thư viện, các cơ sở dữ liệu mẫu được thiết kế chuyên nghiệp, ngoài các cơ sở dữ liệu mà bạn đã mở trong thời gian gần đây (nếu bạn đang kết nối vào Internet) và các liên kết đến các bài viết Office Online phổ biến.

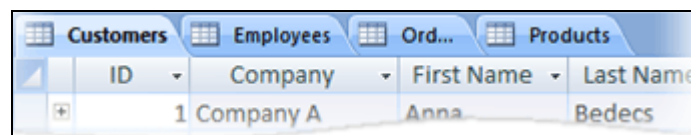


**Navigation Pane:** Pane này liệt kê và cung cấp cách thức truy cập dễ dàng tới tất cả các đối tượng trong cơ sở dữ liệu hiện tại đang mở. Bạn có thể sử dụng Navigation Pane để tổ chức, sắp xếp các Object của bạn theo kiểu (Object Type), ngày tạo (Create Date), ngày sửa đổi (Modified Date), ..., hoặc tùy chỉnh trong các nhóm mà bạn tạo ra. Nếu bạn cần thêm không gian để làm việc với các Form khi thiết kế, bạn có thể dễ dàng thu hẹp Navigation Pane để nó chiếm ít không gian hơn. Pane Navigation được tạo ra để thay thế cửa sổ DataBase đã được sử dụng trong phiên bản trước đây của Access 2007.





**Tabbed objects:** Các Table, queries, forms, reports, macros của bạn được hiển thị như là các Tabbed objects trong Access window.

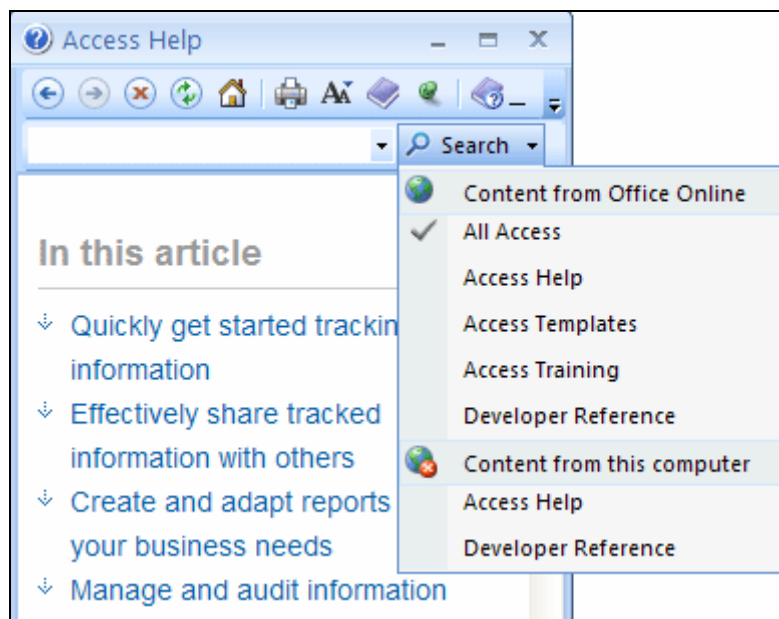


Bằng cách Click chuột vào các đối tượng Tab, bạn có thể dễ dàng chuyển đổi giữa các đối tượng khác nhau.

**Status bar:** Là thanh trạng thái ở dưới cùng của cửa sổ, hiển thị thông tin trạng thái, các nút lệnh cho phép bạn chuyển đổi giữa các lần hiển thị.

**Mini toolbar:** Là một yếu tố, giống như một thanh công cụ, xuất hiện khi một nội dung văn bản được lựa chọn, giúp bạn có thể dễ dàng áp dụng các định dạng như in đậm hoặc nghiêng, hoặc thay đổi font chữ.

**Help window:** Không giống như các phiên bản trước của Access, Microsoft Access 2007 cho phép truy cập cả Access Help và nội dung Developer Reference trên cùng một cửa sổ trợ giúp một cách dễ dàng. Bạn cũng có thể dễ dàng thay đổi phạm vi tìm kiếm của chỉ mình nội dung Developer Reference.

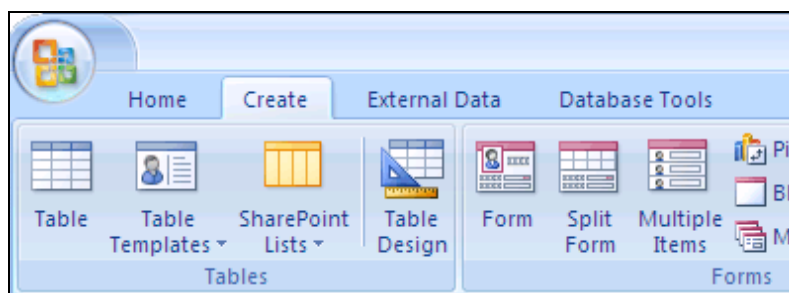


### 3.3. Thêm các công cụ mạnh để tạo đối tượng

Office Access 2007 cung cấp một môi trường trực quan để tạo các Form và các Report, cho phép bạn nhanh chóng tạo ra các Form và các Report. Chúng hiển thị sau khi đã được sắp xếp, lọc và nhóm theo các thông tin về từng đối tượng.

#### Create tab

Sử dụng Tab Create giúp bạn nhanh chóng tạo ra các Form, các Report, Table, Query, macros và Module mới. Nếu bạn chọn một Table hoặc một Query trong Navigation Pane, bạn có thể tạo ra một Form hoặc một báo cáo mới dựa trên các đối tượng đó bằng cách Click chuột vào nút Form hoặc Report. Các Form và các Report mới này được tạo ra gần như ngay lập tức và các thiết kế trên đó cũng được nâng cấp đáng kể. Tự động tạo ra các Form và các Report rất chuyên nghiệp với logo và tiêu đề được định nghĩa sẵn. Ngoài ra, nó còn tự động tạo ra một báo cáo bao gồm cả thông tin về ngày tháng năm, Footer, Totals.

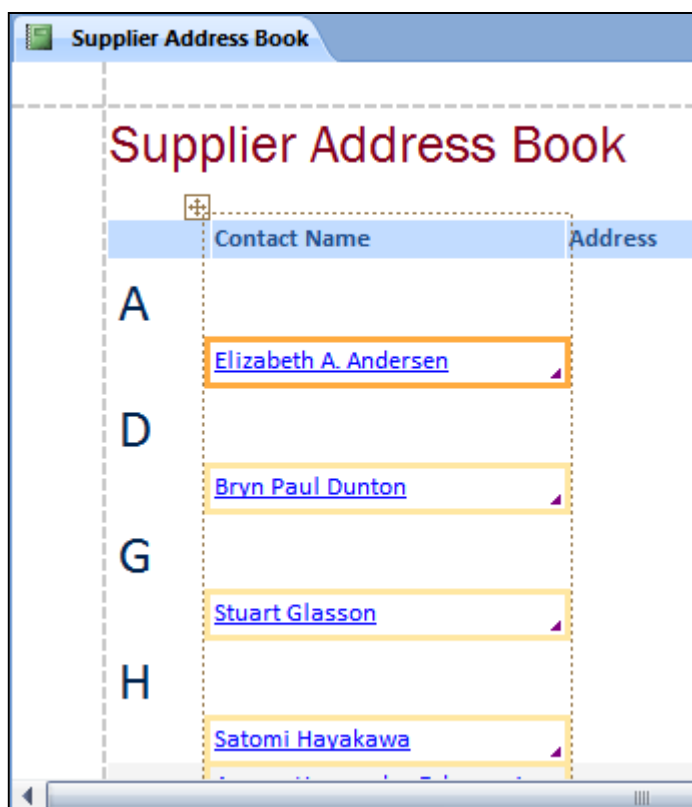


Bạn sẽ thấy rằng các đối tượng được tạo bằng cách này giúp bạn rất nhiều về thời gian và nhanh chóng đưa bạn tiến xa hơn trong quá trình nghiên cứu ứng dụng.

#### Chế độ hiển thị mới: Report view và Layout view

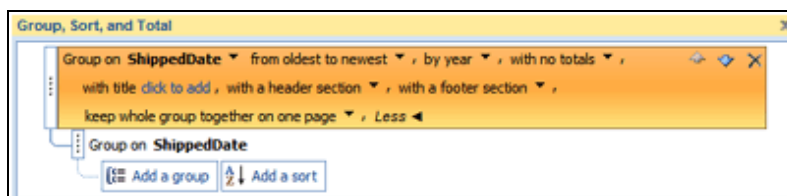
Hai chế độ hiển thị mới này cho phép bạn làm việc với các Form và các Report một cách trực quan hơn, bạn có thể duyệt vẽ một cách chính xác báo cáo mà không cần phải in ra hoặc xem ở chế độ xem trước khi in. Để tham chiếu tới một số bản ghi, bạn sử dụng tính năng lọc, hoặc sử dụng công cụ tìm kiếm theo nội dung văn bản trong bản ghi đó.

Report View cho bạn khả năng duyệt báo cáo nhưng Layout View lại giúp bạn thay đổi thiết kế báo cáo ngay khi bạn đang xem chúng. Bạn có thể sử dụng Layout View để thực hiện thay đổi các thiết kế chung khi đang hiển thị nội dung dữ liệu trên Form hoặc báo cáo. Ví dụ, thêm vào một trường bằng cách kéo một trường đó từ cửa sổ Field List pane, hoặc thay đổi thuộc tính cho đối tượng bằng cách sử dụng Properties Sheet. Layout view với những hỗ trợ mới, stacked layout và tabular layout (Nhóm các Control), cho phép bạn dễ dàng sắp xếp lại các trường, cột, hàng, hoặc toàn bộ Layout. Bạn cũng có thể dễ dàng loại bỏ một trường hoặc thêm vào định dạng trong Layout View.

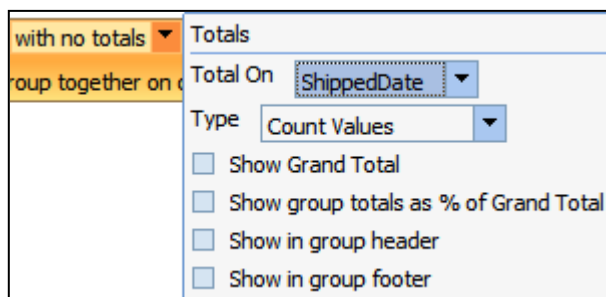


## Group và sort trong các báo cáo

Với Microsoft Access 2007 bạn sẽ có một công cụ rất tốt để Group, Sort, và thêm tổng con trong báo cáo. Bạn sẽ tìm thấy một giao diện người dùng mới, dễ dàng hơn để di chuyển và hiểu được, và khi nó được sử dụng với Layout view mới, bạn sẽ nhìn thấy hiệu quả của các thay đổi ngay lập tức.

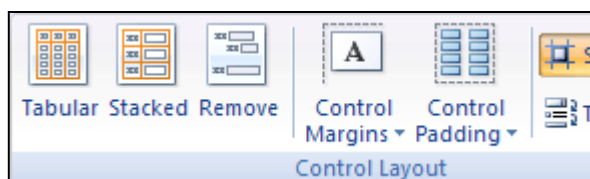


Giả sử bạn cần tạo một báo cáo về doanh số bán hàng theo khu vực. Hãy sử dụng Layout View và Panel Group, Sort, and Total để thêm vào một cấp nhóm, và xem sự thay đổi hiện ra trên báo cáo! Với cách tạo dòng Total mới bạn có thể dễ dàng tạo Total ở nhiều dạng khác nhau trên Header, Footer của báo cáo như kiểu Sum, average, count, maximum, hoặc minimum. Với một Total đơn giản bạn không cần thiết phải tạo ra một trường tính toán mà có thể Click chọn luôn trên công cụ này.



## Điều khiển bố cục trợ giúp việc tạo Form, Report.

Các Form, Report thường bao gồm các thông tin tabuler, như là một cột có chứa tên của khách hàng hay một hàng chứa tất cả các trường dành cho khách hàng đó. Bạn có thể sử dụng Microsoft Access 2007 để nhóm những Control trong một Layout vào với nhau một cách dễ dàng, nó có thể bao gồm cả các Label.



Bạn có thể chọn các điều khiển từ các vùng khác, ví dụ như nhấn trong phần Header hoặc Footer, nó thật sự rất linh hoạt và bạn có thể dễ dàng:

- Di chuyển hoặc thay đổi kích cỡ của một Layout. Ví dụ, di chuyển một cột từ trái qua phải.

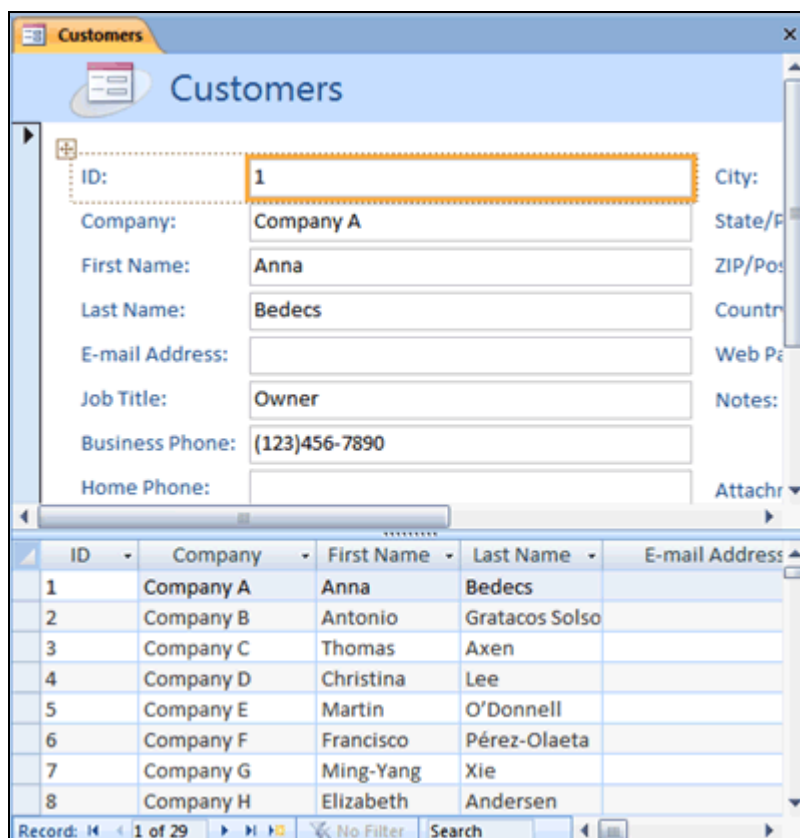
- Định dạng một Layout. Ví dụ, Tên một khách hàng được đặt trong cột in đậm và để nổi.

- Thêm một trường.
- Xóa một trường.

## Split Forms mới hiển thị dữ liệu nhanh hơn

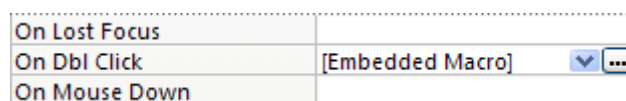
Sử dụng Split Form mới để tạo Form mà kết hợp từ một Datasheet view và một Form View. Bạn có thể thiết lập thuộc tính để chỉ định Access đặt lại vị trí của DataSheet lên trên, xuống dưới hay sang trái sang phải.

	Q2	Q1
	\$1,632.50	\$13,800.00
	\$7,327.50	\$680.00
	\$3,132.00	\$1,551.00
	\$3,520.00	\$1,049.00
	\$3,786.50	\$0.00
	\$3,625.25	\$0.00
	\$2,778.00	\$127.50
	\$620.00	\$1,930.00



### Nhúng Macros vào các Form, Report

Một cách mới và đáng tin cậy, bạn có thể nhúng macros để tránh phải viết mã. Một Macro nhúng được lưu giữ trong một thuộc tính và là một phần của đối tượng chứa nó. Bạn có thể chỉnh sửa những thiết kế của một Macro nhúng mà không phải lo lắng về những điều khiển khác cũng sử dụng Macro này – các Macro nhúng hoàn toàn độc lập. Macros nhúng rất đáng tin cậy bởi vì chúng có khả năng tự động ngăn chặn một số những thao tác không an toàn.

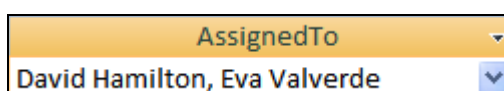


### 3.4. Nhúng kiểu dữ liệu và điều khiển mới

Microsoft Access 2007 cung cấp nhiều kiểu dữ liệu và điều khiển mới cho phép lưu trữ nhiều dạng dữ liệu khác nhau, và bạn có thể nhập chúng dễ dàng hơn.

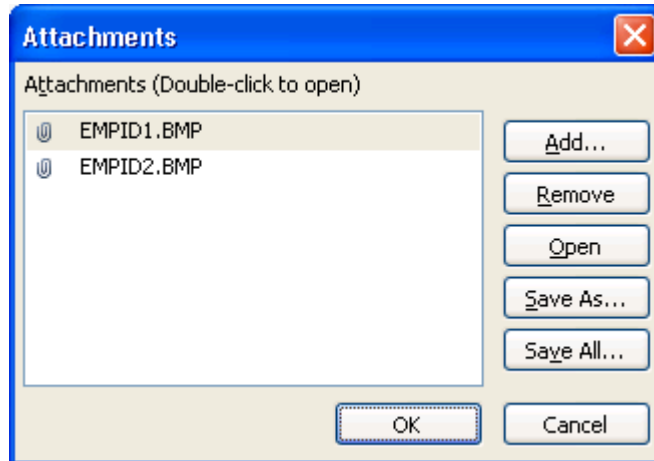
#### Trường đa trị cho các dữ liệu phức tạp

Bạn có thể tạo ra một trường chứa nhiều giá trị, còn được gọi là dạng dữ liệu phức tạp. Giả sử bạn cần phải chỉ định một công việc cho một trong những nhân viên hoặc nhà thầu của mình, nhưng bạn muốn chỉ định nó cho nhiều hơn một người. Trong hầu hết các hệ thống quản lý cơ sở dữ liệu và trong các phiên bản cũ của Access, bạn cần tạo ra rất nhiều các mối quan hệ để làm điều này một cách chính xác. Trong Microsoft Access 2007, phần cứng sẽ thực hiện việc này cho bạn, khi bạn chọn một trường mà bạn thiết lập để chấp nhận nhiều giá trị. Trường đa trị trong Microsoft Access 2007 sẽ rất thích hợp để bạn làm việc này. Office Access 2007 có khả năng tương thích với rất nhiều kiểu dữ liệu.



### Trường Attachment để chứa Files

Kiểu dữ liệu Attachment mới cho phép bạn dễ dàng lưu trữ tất cả các loại tài liệu và tập tin nhị phân trong cơ sở dữ liệu của bạn mà không cần quan tâm tới kích thước của chúng. Microsoft Access 2007 sẽ tự động nén chúng lại đến mức có thể để giảm thiểu không gian lưu trữ. Nếu bạn cần đính kèm một tập tin Microsoft Word 2007 và một bản ghi hoặc lưu trữ một bức ảnh kỹ thuật số vào CSDL của mình? Bằng cách sử dụng trường Attachment sẽ làm cho công việc trở nên hết sức dễ dàng. Thậm chí bạn có thể thêm nhiều tập tin đính kèm chỉ với một bản ghi duy nhất.



### Trường Memo lưu trữ Rich Text và hỗ trợ lịch sử sửa đổi

Với khả năng lưu trữ Rich Text của Microsoft Access 2007, trong một bản ghi số lượng văn bản của bạn sẽ không còn bị giới hạn. Bạn có thể định dạng văn bản với các tùy chọn (như đậm, nghiêng, phông chữ, màu sắc, và những tùy chọn định dạng phổ biến khác) và lưu giữ các văn bản trong cơ sở dữ liệu của mình. Rich-formatted text được lưu giữ trong một trường Memo như một văn bản HTML dựa trên định dạng tương ứng với Rich Text, là một kiểu dữ liệu trong Windows SharePoint Services.

Trường Memo sẽ hữu ích cho việc lưu trữ một số lượng lớn thông tin. Với Microsoft Access 2007, bạn có thể thiết lập thuộc tính Append Only để giữ lại tất cả lịch sử của những lần thay đổi thông tin trên trường đó. Và bạn hoàn toàn có thể xem lại những sự thay đổi này. Tính năng này cũng hỗ trợ theo dõi trong Windows SharePoint Services, nhờ vậy bạn có thể sử dụng Access để xem nội dung lịch sử danh sách SharePoint.

### Chọn này với Calendar

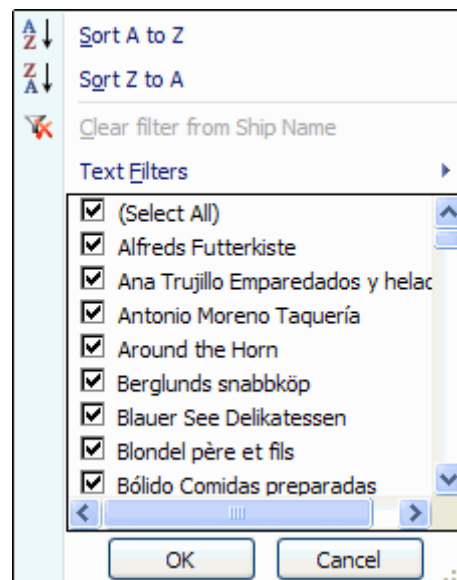
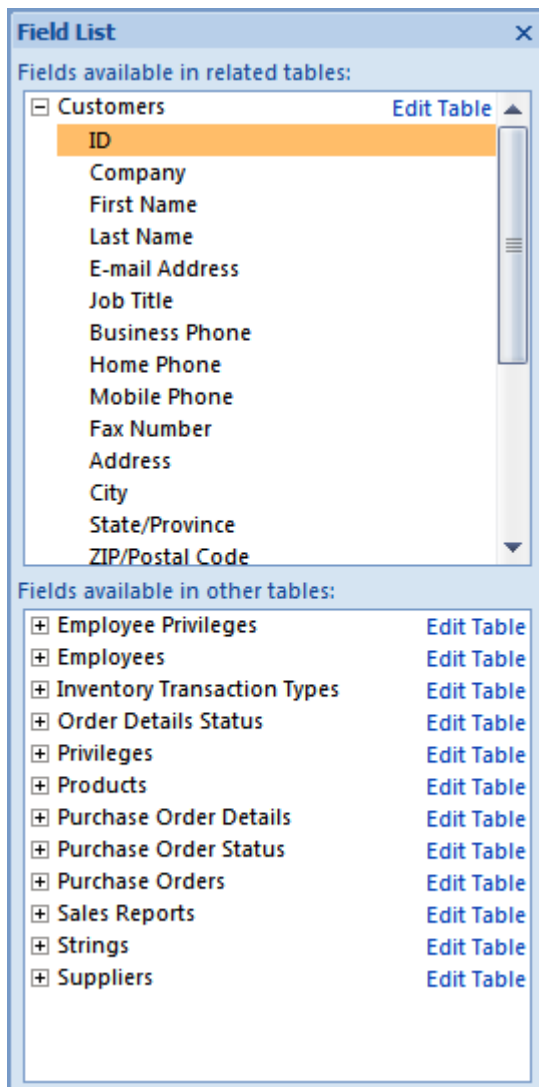
Những trường và những điều có kiểu dữ liệu kiểu Date/Time sẽ tự động cập nhật tính năng mới này: Một điều khiển lịch sẽ được gắn kèm cho phép chọn ngày tháng. Nút lịch tự động xuất hiện và trở chính xác tới ngày tháng hiện tại. Nếu bạn muốn biết thứ 6 tới là ngày bao nhiêu? Bạn chỉ Click chọn và một bảng lịch sẽ tự động xuất hiện cho phép bạn tìm và chọn ngày đó. Bạn cũng có thể tùy chọn tắt bảng lịch này trên trường hoặc điều khiển bằng cách sử dụng một thuộc tính.

### 3.5. Nâng cấp thiết kế và các công cụ phân check

Những công cụ mới trong Microsoft Access 2007 sẽ giúp bạn tạo ra các đối tượng cơ sở dữ liệu nhanh hơn và sau đó phân check dữ liệu một cách dễ dàng hơn.

#### Nâng cao Field List pane

Với Pane Field List bạn có thể kéo thả trường từ một bảng vào trong bản ghi nguồn, các bảng có thể có mối quan hệ với nhau hoặc không. Microsoft Access 2007 đã tạo sẵn các cấu trúc vì vậy nếu một mối quan hệ giữa các bảng là cần thiết thì nó sẽ tự động tạo ra mối qua hệ đó.



### Nâng cao công cụ lọc, sắp xếp

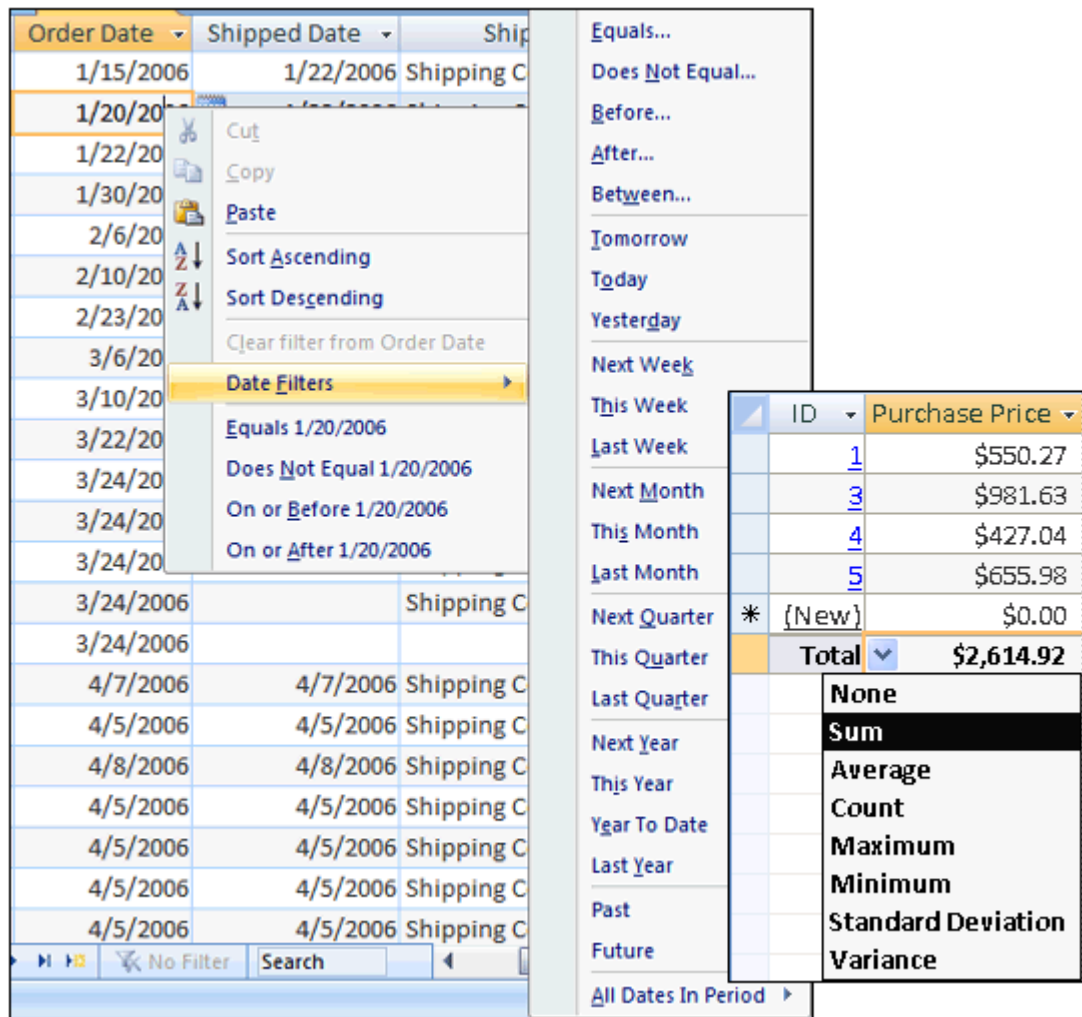
Giả sử bạn cần phải nhanh chóng tìm thấy một giá trị phù hợp hoặc sắp xếp một cột của dữ liệu. Microsoft Access 2007 với chức năng AutoFilter cung cấp khả năng lọc nhanh chóng để bạn có thể tìm thấy các dữ liệu mà bạn cần. Bạn cũng có thể dễ dàng lựa chọn từ các giá trị duy nhất trong một cột, điều này rất có ích cho các tình huống khi bạn không thể gọi lại tên mà bạn cần, hoặc bạn có thể sắp xếp các giá trị bằng cách sử dụng các trình đơn tùy chọn.

Các bộ lọc tùy chọn phổ biến nhất rất dễ dàng nhìn thấy trong trình đơn lệnh, bạn có thể nhanh chóng sử dụng chúng để giới hạn các thông tin dựa trên các dữ liệu nhập vào. Bộ lọc tùy chọn tự động thay đổi dựa trên kiểu dữ liệu mà bạn đang tác tác.

Những tính năng lọc hay sắp xếp mới được thiết kế sẽ giúp bạn có được những kinh nghiệm nhất định làm việc không chỉ với Access 2007 mà cả Excel 2007.

### Totals (Tổng cộng) và màu nền trong DataSheet

Với dòng Total trong Datasheet View mới bạn có thể hiển thị một tổng (sum), hoặc đến bản ghi (count), tính giá trị trung bình (average), ... Sau khi bạn thêm dòng Total, bạn trở vào mũi tên trong các cột và chọn phép tính cần thiết muốn



Datasheet, Report, Forms đều được hỗ trợ màu nền cho các row, bạn có thể thiết lập mặc định màu nền cho chúng. Tạo bóng cho các hàng cũng rất dễ dàng, và bạn có thể chọn bất kỳ màu sắc nào mà mình thích.

Orders		
	Order ID	Employee
+	30	Anne Hellung-Larsen
+	31	Jan Kotas
+	32	Mariya Sergienko
+	33	Michael Neipper
+	34	Anne Hellung-Larsen
+	35	Jan Kotas
+	36	Mariya Sergienko
+	37	Laura Giussani

### 3.6. Tăng cường an ninh

Microsoft Access 2007 tăng cường chức năng bảo mật một cách mạnh mẽ và thông minh với Windows SharePoint giúp bạn quản lý hiệu quả, theo dõi các ứng dụng an toàn hơn trước rất nhiều. Bằng cách lưu trữ dữ liệu theo dõi của bạn trong danh sách ứng dụng trên Windows SharePoint Services, bạn có thể kiểm tra xem xét lại lịch sử, khôi phục lại thông tin bị xóa, phân quyền dữ liệu truy cập.

### 3.7. Cách thức mới để chia sẻ dữ liệu và cộng tác với những người khác

Bằng cách sử dụng các tính năng cộng tác mới của Microsoft Access 2007, bạn có thể thu thập và chia sẻ thông tin với những người khác hiệu quả hơn trong một môi trường bảo mật trên Web.

#### Check hợp với Windows SharePoint Services

Microsoft Access 2007 với Windows SharePoint Services trong một số trường hợp, giúp bạn tận dụng lợi thế của nhiều tính năng cộng tác trong Windows SharePoint Services.

- **Sửa đổi dữ liệu theo dõi:** Với sự giúp đỡ của Windows SharePoint Services, bạn có thể xem lịch sử sửa đổi của một bản ghi. Giả sử bạn cần phải biết ai đã thay đổi nội dung một hàng và nó được thay đổi khi nào? Tất cả có thể tìm thấy bằng cách xem trong lịch sử sửa đổi.

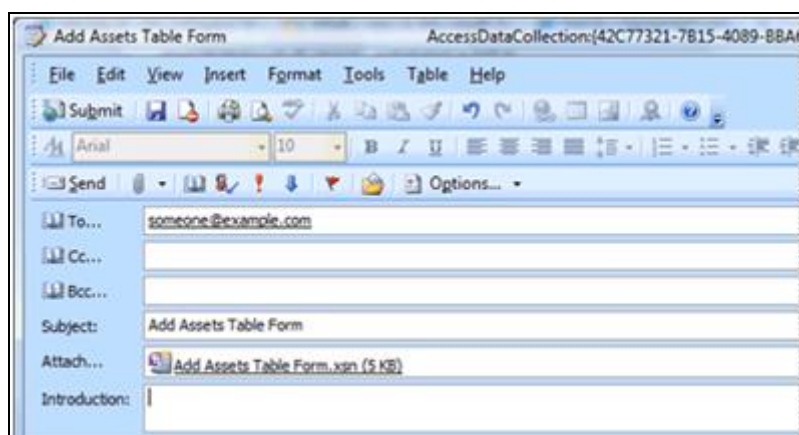
- **Cho phép cài đặt:** Với sự giúp đỡ của Windows SharePoint Services, bạn có thể đặt quyền truy cập cho phép người dùng thêm vào danh sách. Bạn có thể chọn lọc cho phép hoặc từ chối một số người dùng truy cập. Bạn cũng có thể chỉ định giới hạn chế quyền chỉ đọc hoặc Full (mọi thay đổi tác động đều được chấp nhận).

- **Lấy từ Recycle Bin:** Bạn có thể sử dụng Recycle Bin mới trong Windows SharePoint Services để dễ dàng xem các trường đã xóa hoặc khôi phục lại các thông tin đã bị xóa trước đó.

**Mở Form, Report trong Windows SharePoint Services:** Windows SharePoint Services cho phép người dùng có thể mở trực tiếp các danh sách trong Access views (Form, Report, DataSheet) từ một site SharePoint. Khi bạn chọn một Access views, Microsoft Access 2007 mở và gọi tới các Form, Report hay Datasheet đó. Vì vậy nó giúp bạn có thể chạy trên SharePoint site mà không cần khởi động Microsoft Access 2007. Bạn cũng có thể chọn để mở một danh sách trong Access, và nếu cơ sở dữ liệu đó không tồn tại, nó sẽ tự động tạo mới và các Form, Report sẽ được kế thừa trong danh sách của bạn.

#### Thu thập dữ liệu bằng cách sử dụng các biểu mẫu InfoPath và Outlook

Bằng cách sử dụng các tính năng mới Data Collection, Microsoft Access 2007 có thể tự động tạo ra một định dạng Microsoft Office InfoPath 2007 hoặc định dạng HTML và nhúng nó trong phần body của một e-mail. Sau đó bạn có thể gửi chúng đi tới những địa chỉ người nhận từ Contact Outlook của bạn, hoặc để tên của người nhận được lưu giữ trong một trường trong cơ sở dữ liệu Access.



Bạn lựa chọn xem để thu thập những thông tin mới hay cập nhật các thông tin hiện có. Người nhận biểu mẫu sau đó sẽ hoàn thiện và gửi lại. Microsoft Access 2007 tự động nhận dạng các biểu mẫu gửi đến và lưu vào cơ sở dữ liệu.



### **Export sang định dạng PDF và XPS**

Bây giờ bạn có thể xuất dữ liệu một định dạng file .PDF (Portable Document Format) hoặc .XPS (XML Paper Specification) dùng cho việc in ấn, Post bài hay e-mail. Việc xuất một Form, Report, hoặc Datasheet sang file có định dạng .Pdf hay .XPS cho phép bạn dễ dàng phân phối các thông tin mà vẫn giữ lại tất cả nhưng định dạng mà bạn thiết lập. Mà không cần người nhận phải cài đặt Access.

**Portable Document Format:** Một định dạng PDF là một file cố định giúp bảo vệ các định dạng của văn bản và chia sẻ tệp tin. Các định dạng PDF bảo đảm rằng khi các tệp tin được xem trực tuyến hoặc in ra, nó giữ lại chính xác các định dạng mà bạn đã thiết kế, và giữ cho tệp tin không bị sao chép hoặc sửa đổi. Các định dạng PDF cũng hữu ích cho các tài liệu mà sẽ được sao chép bằng cách sử dụng các phương pháp in ấn thương mại.

**XML Paper Specification:** Một XPS là một định dạng tệp tin điện tử, bảo vệ định dạng tài liệu và giúp chia sẻ tệp tin. Các định dạng XPS đảm bảo rằng khi các tệp tin được xem trực tuyến hoặc in ra, nó giữ lại chính xác các định dạng mà bạn đã định sẵn, giúp cho các dữ liệu trong tệp tin không bị sao chép hoặc thay đổi.

Bạn có thể xuất dữ liệu ra một file có định dạng .Pdf hay .XPS từ Microsoft Access 2007 chỉ sau khi bạn cài đặt các add-in.

### **Làm việc với dữ liệu bên ngoài được thực hiện dễ dàng hơn**

Các tính năng mới trong Microsoft Access 2007 làm cho việc Import hay Export trở nên dễ dàng hơn. Bạn có thể lưu lại một hoạt động Import hay Export và sau đó tái sử dụng ở thời gian tiếp theo. Với Import Spreadsheet Wizard sẽ cho phép bạn ghi đè lên kiểu dữ liệu cho bởi Access, và bạn có thể Import, Export, và và liên kết đến các định dạng tệp tin Excel 2007 mới.

### **3.8. Cách tốt hơn để gỡ rối vấn đề**

Microsoft Office Diagnostics là một loạt các xét nghiệm chẩn đoán có thể giúp bạn khám phá lý do tại sao máy tính của bạn có crashing. Các xét nghiệm chẩn đoán có thể giải quyết một số vấn đề trực tiếp và có thể xác định cách thức mà bạn có thể giải quyết các vấn đề. Microsoft Office Diagnostics thay thế cho những tính năng sau đây của Microsoft Office 2003: Dò tìm, sửa chữa, và phục hồi ứng dụng Microsoft Office.

### **3.9. Nâng cao chất lượng các công cụ Proofing**

Sau đây là một số tính năng mới của công cụ kiểm tra chính tả.

Công cụ kiểm tra chính tả đã được làm chắc chắn hơn trong bộ Microsoft Office 2007. Dưới đây là ví dụ về sự thay đổi này:

Một vài tùy chọn kiểm tra toàn cục bây giờ sẽ mang tính toàn cục. Nếu bạn thay đổi một trong các tùy chọn này ở bất cứ ứng dụng nào trong bộ Office nó cũng sẽ thay đổi tương ứng trong các ứng dụng khác của bộ Office.

Ngoài ra để chia sẻ cùng một bộ từ điển, tất cả các chương trình có thể quản lý chúng bằng cách sử dụng cùng một hộp thoại.

## Bài 2. Làm việc với Table

Khi bạn tạo ra một cơ sở dữ liệu, bạn lưu trữ dữ liệu trong các bảng (dựa trên nội dung của các hàng và cột). Ví dụ, bạn có thể tạo một bảng Contacts lưu trữ danh sách gồm tên, địa chỉ, và số điện thoại, hoặc một bảng Products lưu trữ những thông tin về sản phẩm cần quản lý.

Bài viết này với mục đích hướng dẫn bạn làm thế nào để tạo ra một bảng và làm thế nào để thêm các trường rồi thiết lập thuộc tính cho những trường đó. Cũng như hướng dẫn bạn cách tạo ra những khóa chính cho các bảng đó.

Bởi vì có rất nhiều đối tượng khác nhau nhưng lại phụ thuộc lẫn nhau trong một Database. Vì vậy trước khi bắt tay vào thiết kế các bảng bạn nên xem xét một cách kỹ lưỡng các yêu cầu mình, để làm sao không bị dư thừa các bảng dữ liệu không có ích.

### 1. Tổng quát

Mỗi Table là một đối tượng của CSDL mà bạn sử dụng để lưu trữ về một chủ đề cụ thể, chẳng hạn như nhân viên trong công ty, sản phẩm của một nhà máy nào đó, ... Một bảng gồm có nhiều trường và nhiều bản ghi.

Mỗi bản ghi chứa dữ liệu về một chủ thể mà bảng lưu trữ, chẳng hạn như các thông tin về một nhân viên. Một bản ghi thường tương ứng với một dòng trong bảng dữ liệu.

Mỗi trường chứa dữ liệu về một trong những thuộc tính của chủ thể trong bảng, chẳng hạn như tên hay địa chỉ e-mail của một nhân viên. Mỗi trường tương ứng với một cột trong bảng dữ liệu.

Một bản ghi bao gồm giá trị của nhiều trường chẳng hạn như Nhà sách Minh Lâm, nhà sách, nickname@hotmail.com . Giá trị của một trường thường được gọi là một thực thể.

ID	Company	First Name	Last Name
1	Company A	Anna	Bedecs
2	Company B	Antonio	Gratacos Solsona
3	Company C	Thomas	Axen

Trên đây là một bảng được mở ở chế độ Datasheet View

Trong đó:

(1) : Một bản ghi (2): Một trường (3): Giá trị một trường

Như vậy chúng ta có thể tổng kết lại như sau:

Một cơ sở dữ liệu có thể chứa nhiều bảng, mỗi bảng lưu trữ thông tin về một đối tượng, chủ thể khác nhau. Trong mỗi bảng có thể chứa một hoặc nhiều trường với những kiểu dữ liệu cũng khác nhau như kiểu chuỗi, số, ngày tháng, kiểu liên kết, ...

#### 1.1. Thuộc tính của bảng và trường

Bảng và trường có những thuộc tính mà bạn có thể thiết lập để kiểm soát hành vi cũng như đặc điểm của chúng.



Đây là một bảng được mở ở chế độ Design View. Trong đó:

- (1): Thuộc tính của bảng
- (2): Thuộc tính của trường

Trong một Database Access, thuộc tính của bảng ảnh hưởng trực tiếp, toàn diện tới sự thể hiện cũng như hành vi của bảng đó. Ở chế độ Design View, thuộc tính của bảng được thiết lập Pane Property Sheet. Ví dụ, bạn có thể thiết lập thuộc tính Default View cho bảng để mỗi lần hiển thị nó sẽ hiển thị theo chế độ đó.

Một thuộc tính của trường được áp dụng cho một trường cụ thể trong bảng và xác định một trong những tính chất đặc trưng của trường đó. Bạn có thể thiết lập thuộc tính cho trường ở chế độ hiển thị Datasheet View, cũng như ở chế độ Design View bằng cách sử dụng Pane Field Properties.

## 1.2. Kiểu dữ liệu

Mỗi trường có một kiểu dữ liệu nhất định. Một kiểu dữ liệu của một trường cho biết kiểu dữ liệu mà trường đó lưu trữ chẳng hạn như trường dữ liệu kiểu Text sẽ cho phép lưu trữ các ký tự, hay kiểu Date/Time cho phép lưu trữ dữ liệu kiểu ngày tháng, ...

Field Name	Data Type	Description
Comments	Memo	
Attachments	Attachment	

Field Properties

Kiểu dữ liệu của trường thực chất là thuộc tính của trường đó, tuy nhiên nó vẫn khác thuộc tính ở một số điểm sau đây:

- Bạn có thể thiết lập kiểu dữ liệu cho trường trong lưới thiết kế bảng nhưng không thể làm việc đó trong Pane Field Properties.
- Một kiểu dữ liệu của trường xác định những thuộc tính mà trường đó có thể có.

### Chú ý:

Bạn có thể tạo ra một trường mới ở chế độ Datasheet view trong Access bằng cách nhập dữ liệu trực tiếp dữ liệu vào một cột mới. Khi bạn tạo ra một trường mới bằng cách này Access sẽ tự động định nghĩa kiểu dữ liệu cho trường dựa trên những giá trị mà bạn nhập vào cột đó. Nếu như không có một kiểu dữ liệu nào phù hợp cho rất nhiều giá trị trên cột đó

Access sẽ hiểu đó là dữ liệu kiểu Text, nếu cần thiết bạn hoàn toàn có thể thay đổi lại những kiểu dữ liệu này.

Dưới đây là bảng mô tả về kiểu dữ liệu mà Access tự động thiết lập cho trường tương ứng với dữ liệu bạn nhập vào.

Dữ liệu có thể nhập vào	Kiểu dữ liệu mà Access có thể tự định nghĩa
zipszips	Text
<a href="http://www.zipszips.com">http://www.zipszips.com</a>	Hyperlink
1	Number, Long Integer
50,000	Number, Long Integer
50,000.99	Number, Double
50000.389	Number, Double
12/67	Date/Time
December 31, 2006	Date/Time
10:50:23	Date/Time
10:50 am	Date/Time
17:50	Date/Time
\$12.50	Currency
21.75	Number, Double
123.00%	Number, Double
3.46E+03	Number, Double

### 1.3. Quan hệ giữa các bảng

Mặc dù mỗi bảng lưu trữ dữ liệu về một chủ thể nhất định, nhưng xét trong một Database thì các chủ thể này lại có mối quan hệ mật thiết với nhau. Ví dụ, một cơ sở dữ liệu có thể chứa:

Một bảng customers là danh sách khách hàng công ty của bạn.

Một bảng products chứa danh sách những sản phẩm mà công ty bạn sản xuất ra.

Một bảng orders chứa những đơn đặt hàng mà khách hàng mua sản phẩm.

Vì khách hàng trong orders mua sản phẩm của công ty sản xuất ra nên những khách hàng đó phải là khách hàng trong bảng customers, và những sản phẩm cũng phải là những sản phẩm trong bảng Products.

Chính vì những mối liên hệ đó mà bạn cần phải làm thế nào để có thể liên kết các bảng này lại với nhau. Nhờ việc liên kết này mà bạn có thể dễ dàng kiểm soát sự toàn vẹn dữ liệu trong một Database.

Để liên kết các bảng lại với nhau bạn cần phải tạo ra các mối liên kết (quan hệ). Một mối quan hệ là một kết nối Logical giữa hai bảng với nhau thông qua một trường có chung Ví dụ trong bảng Customers sẽ có trường Id xác định tính duy nhất của khách hàng đó và trong bảng Orders cũng có trường Id\_Kh trường này lưu trữ Id của khách hàng mua sản phẩm, hai trường này được gọi là trường liên kết giữa bảng Customers và bảng Order.

### 1.4. Khóa

Những trường mà là một phần của mối quan hệ trong bảng gọi là trường khóa. Một khóa thường bao gồm một hoặc nhiều trường khác nhau. Trong bảng có thể có hai loại khóa **Primary key**:

**Một bảng chỉ có duy nhất một khóa Primary key.** Một khóa Primary key có thể bao gồm một hoặc nhiều trường kết hợp với nhau xác định tính duy nhất của mỗi bản ghi lưu trữ trong bảng.

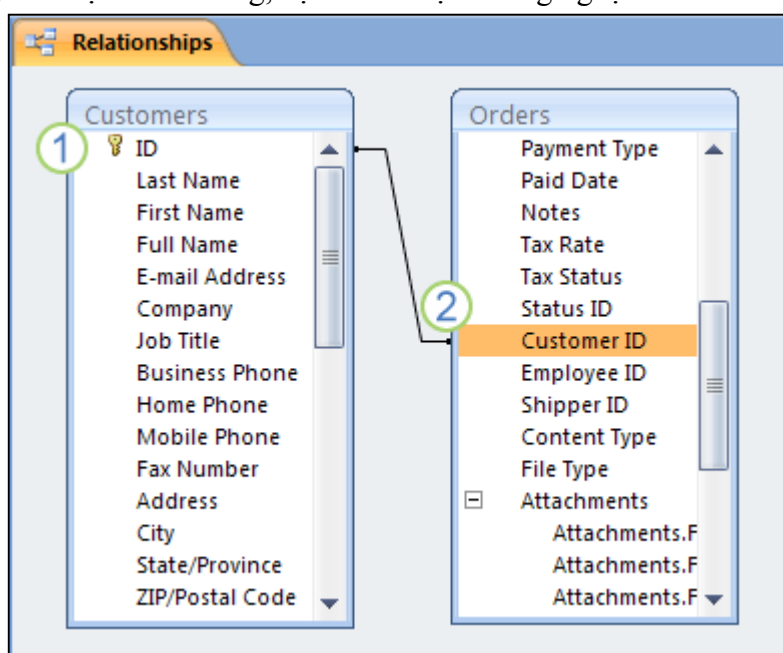
Thông thường, có một số để xác định tính duy nhất của một bản ghi, chẳng hạn như số ID, số sê-ri, hoặc mã số sẽ được chọn làm khóa. Ví dụ, bạn có một bảng Customers mỗi khách hàng được xác định duy nhất thông qua số Id của khách hàng đó. Như vậy trường Id chính là trường khóa của bảng Customers. Khi một chính khóa (Primary key ) chứa nhiều hơn một trường thì nó là sự kết hợp của những trường đã tồn tại, kết hợp với nhau để tạo ra một giá trị duy nhất. Ví dụ, bạn có thể kết hợp các trường Ho, Ten, Ngay\_Sinh để tạo thành một trường khóa cho bảng People.

### Foreign key:

Một bảng có thể có một hoặc nhiều Foreign key. Một Foreign key chứa các giá trị mà tương ứng với những giá trị đó là những giá trị của trường khóa chính ở những bảng khác. Ví dụ, bạn có thể có một bảng Orders, trong đó mỗi đơn hàng có một số Id của khách hàng đặt đơn hàng đó, Id xác định tính duy nhất của một bản ghi trong bảng Customers. Như vậy có thể nói trường Id khách hàng trong bảng đơn hàng là một Foreign key (ngoại khóa)

Sự tương ứng của các giá trị giữa những trường khóa là nên tảng của một mối quan hệ giữa các bảng với nhau. Bạn có thể sử dụng mối quan hệ của bảng để kết hợp dữ liệu từ những bảng liên quan. Ví dụ, giả sử bạn có một bảng Customers và một bảng Orders, mỗi bản ghi được xác định duy nhất bởi trường khóa Id (trường khóa chính).

Để liên kết mỗi đơn hàng với một khách hàng, bạn thêm một trường ngoại khóa vào bảng Orders tương ứng với trường Id trong bảng Customers, và sau đó tạo mối quan hệ giữa hai khóa này. Khi bạn thêm một bản ghi vào Orders, bạn sử dụng giá trị cho trường Id khách hàng được lấy từ bảng Customers. Khi nào bạn cần xem bất kỳ thông tin nào về một đơn đặt hàng của khách hàng, bạn nên sử dụng các mối quan hệ để xác định được những dữ liệu từ bảng Customers tương ứng với các bản ghi trong bảng Orders.



### Mối quan hệ giữa hai bảng, được hiển thị trong cửa sổ Relationships.

- (1): Là trường primary key (khóa chính), xác định bởi biểu tượng khóa bên cạnh tên trường.  
 (2): Là trường foreign key (ngoại khóa)

### 1.5. Lợi ích của việc sử dụng relationships

Việc lưu trữ dữ liệu được phân chia trong những bảng liên quan có những lợi ích sau đây:

**Sự nhất quán:** Vì mỗi nội dung của dữ liệu được ghi chỉ một lần, trong một bảng, có ít khả năng xảy ra sự mâu thuẫn, không minh bạch. Chẳng hạn, bạn lưu trữ tên của một khách hàng chỉ một lần, trong bảng khách hàng, hơn là lưu trữ nó nhiều lần (Và tiềm ẩn những mâu thuẫn) trong bảng chứa dữ liệu đơn hàng.

**Hiệu quả:** Ghi dữ liệu ở một chỗ có nghĩa là bạn sử dụng ít không gian trên đĩa cứng hơn. Thêm vào đó, bảng ít dung lượng sẽ cho phép truy cập, truy vấn dữ liệu nhanh hơn là các bảng lớn. Cuối cùng, nếu bạn không sử dụng những bảng riêng biệt để lưu trữ thông tin về

các chủ thể riêng biệt, bạn sẽ phải theo dõi, quản lý cả những giá trị Null dẫn đến sự dư thừa dữ liệu trong các bảng. Cả hai lý do đó có thể sẽ tiêu phí không gian lưu trữ và ngăn cản quá trình làm việc với dữ liệu của bạn.

Sự hiểu biết: Quá trình thiết kế một Database sẽ dễ hiểu hơn nếu một chủ thể được phân rã hợp lý thành những bảng riêng biệt.

### 1.6. Trước khi bạn bắt đầu

Trước khi bạn bắt đầu tạo ra các bảng trong Microsoft Office Access 2007, có vài sự khác biệt so với những phiên bản trước đó của Access mà bạn cần chú ý.

Bảng và trường Template đã được thay thế cho Table Wizard trong các phiên bản trước, bạn đã sử dụng Table Wizard để tạo ra một Table rất đơn giản với việc chỉ phải trả lời vài câu hỏi. Trong Microsoft Access 2007, Table Wizard sẽ được thay thế bằng Table và các trường Template (mẫu).

Datasheet trong Microsoft Access 2007, bạn có thể tạo ra và sửa đổi các bảng, trường trong khi đang việc làm việc ở chế độ hiển thị Datasheet View

## 2. Tạo mới một bảng

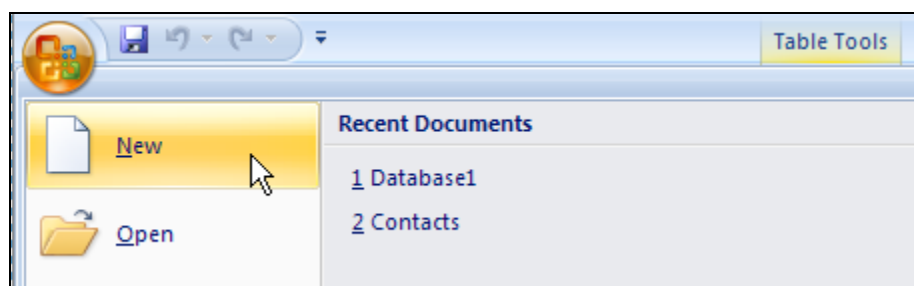
Một cơ sở dữ liệu đơn giản, như danh sách liên lạc, có thể sử dụng chỉ duy nhất một bảng. Tuy nhiên nhiều cơ sở dữ liệu, phải sử dụng rất nhiều bảng.

Khi bạn tạo ra một cơ sở dữ liệu mới, đồng thời bạn cũng đã tạo ra một File trên máy tính, nó đóng vai trò như một Container (nơi chứa) tất cả các đối tượng trong cơ sở dữ liệu, trong đó có các bảng dữ liệu.

Bạn có thể tạo ra một bảng bằng việc tạo ra một cơ sở dữ liệu mới, hoặc cũng có thể tạo ra bằng việc chèn một bảng vào trong một cơ sở dữ liệu đã tồn tại, hay bằng cách Import hoặc liên kết tới các bảng từ những nguồn dữ liệu khác. Khi bạn tạo mới một DataBase xong, một bảng mới chưa có nội dung sẽ được chèn vào đó. Bạn có thể nhập dữ liệu vào cũng như bắt đầu định nghĩa các trường cho bảng.

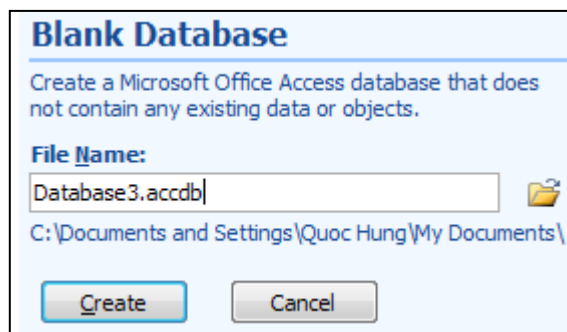
### 2.1. Tạo bảng mới trong Database mới.

Click chuột chọn nút Microsoft Office Button , sau đó click New.



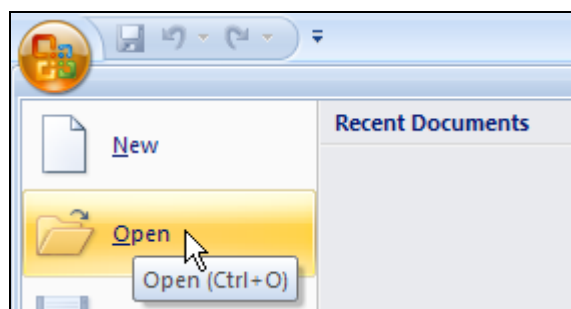
Trong hộp nhập File Name, gõ vào tên cho Database mới. Để thay đổi đường dẫn lưu Database bạn Click chuột chọn biểu tượng Folder để chọn lại. Cuối cùng Click Create.

Một Database mới được mở và một Table mới với tên là Table1 được tạo ra và mở ở chế độ Datasheet View.

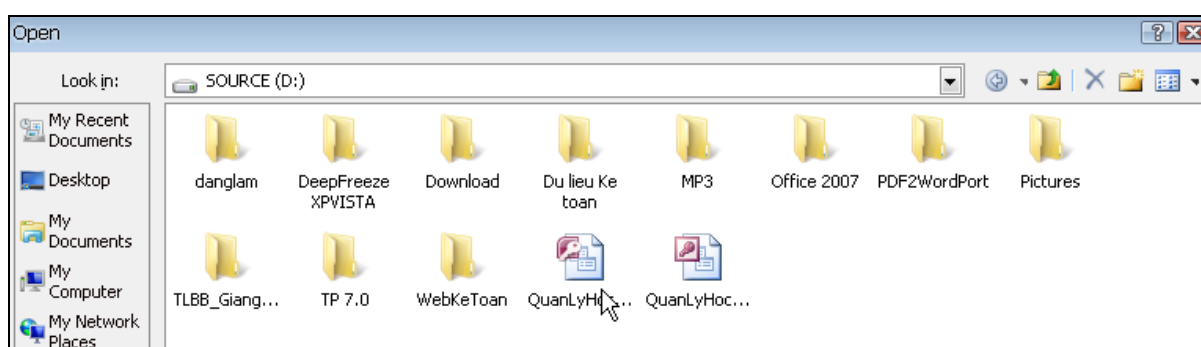


## 2.2. Tạo bảng mới trong Database đã tồn tại

Click chuột chọn nút Microsoft Office Button , sau đó chọn Open.

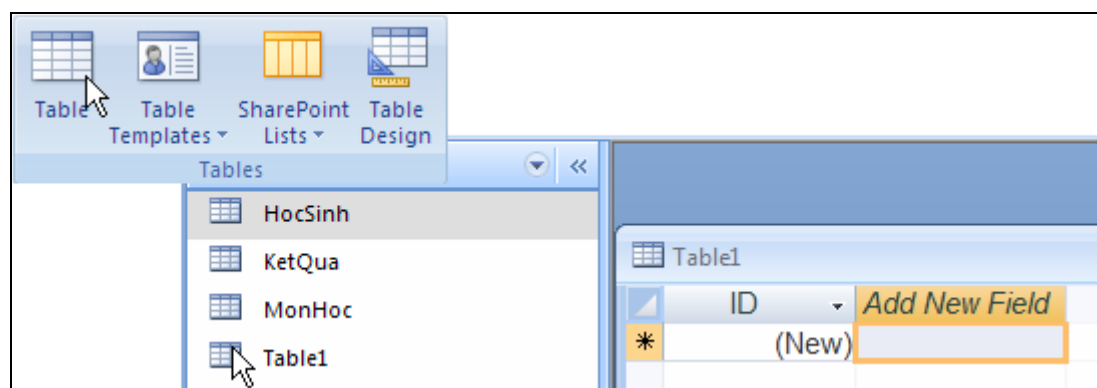


Hộp thoại Open xuất hiện chọn tới Database cần mở rồi Click Open.



Trên Tab Create trong nhóm Table chọn Table.

Một bảng mới được Insert vào Database và bảng này được mở ở chế độ Datasheet View.



## 2.3. Sử dụng bảng Template để tạo bảng

Trong những phiên bản trước đây của Access, bạn sử dụng Table Wizard để tạo nhanh một bảng từ những bảng và trường mẫu. Thay cho chức năng này trong Microsoft Access 2007, bạn có thể sử dụng các bảng và trường Templates.

Một bảng mẫu là một bảng trống không chứa dữ liệu, bạn có thể bắt đầu luôn hoặc chỉnh sửa cho phù hợp với yêu cầu của mình. Microsoft Access 2007 cung cấp những mẫu Table Template dưới đây:

**Contacts:** Là bảng sử dụng để quản lý thông tin liên lạc trong kinh doanh. Như địa chỉ Email, Website, các File đính kèm,...

**Tasks:** Là bảng dùng để theo dõi công việc, bao gồm các trường sử dụng cho những phiếu đính kèm.

**Issues:** Là bảng sử dụng để theo dõi những hoạt động, bao gồm các sử dụng cho những phiếu đính kèm và một trường kiểu Memo lưu trữ lịch sử sửa đổi giá trị trên các trường trong bảng.

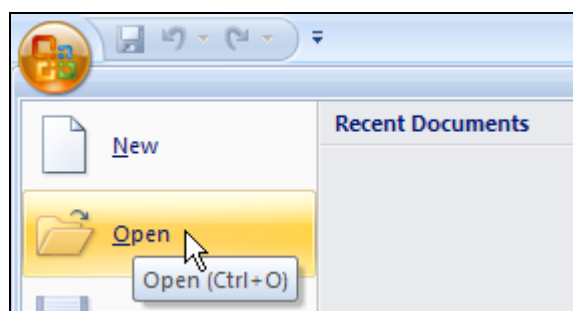
**Events:** Là bảng dùng để quản lý các sự kiện bao gồm một trường Rich Text Memo và một trường cho những phiếu đính kèm.

**Assets:** Là bảng dùng để quản lý tài sản của doanh nghiệp, bao gồm 2 trường tiền tệ để có thể theo dõi được giá trị khấu hao của tài sản.

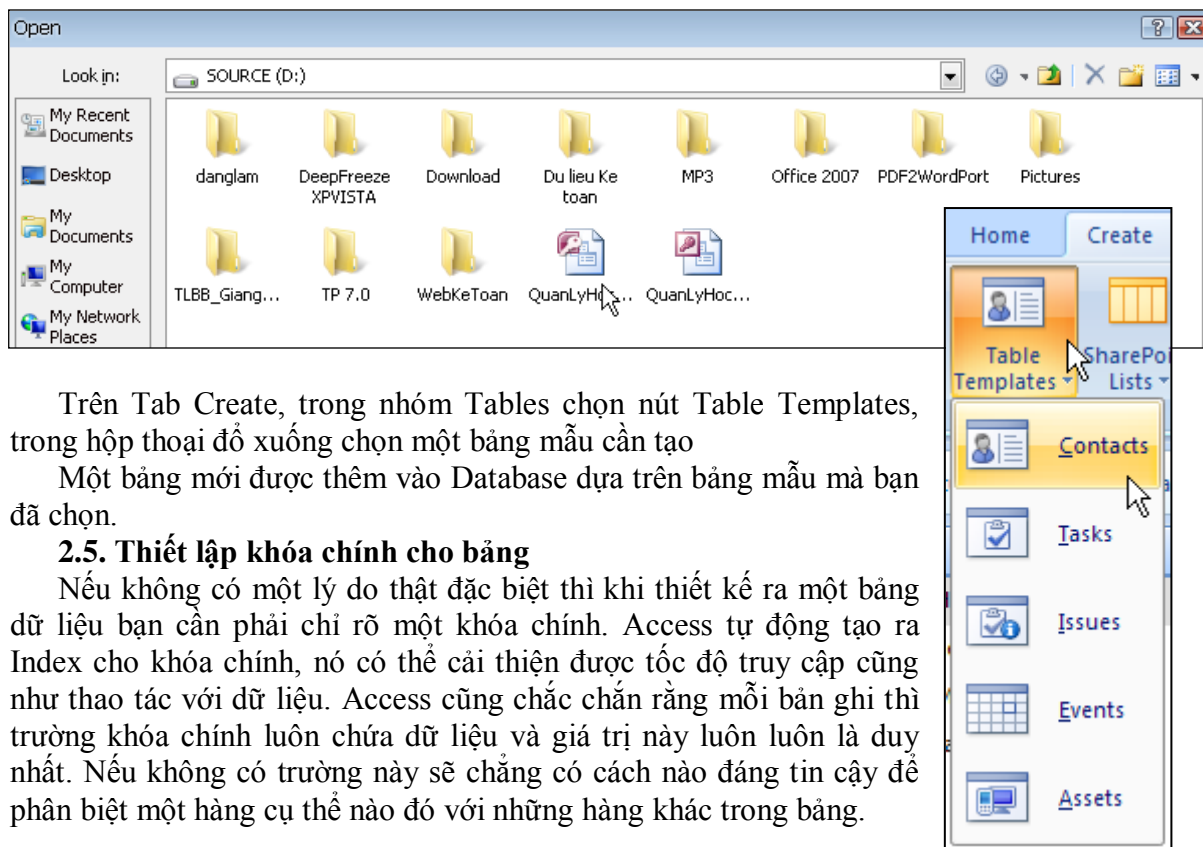
Sau khi bạn tạo ra một bảng bằng cách sử dụng bảng Template, có thể bạn sẽ muốn thêm vào những trường mới bằng cách sử dụng các trường Template. Một trường Template là trường đã được định nghĩa sẵn, bạn có thể thêm vào bất cứ bảng nào ở chế độ hiển thị Datasheet View.

## 2.4. Tạo bảng mới sử dụng Template

Click chọn nút Microsoft Office Button , sau đó chọn Open.



Hộp thoại Open xuất hiện chọn tới Database cần mở rồi Click Open.



Trên Tab Create, trong nhóm Tables chọn nút Table Templates, trong hộp thoại đổ xuống chọn một bảng mẫu cần tạo

Một bảng mới được thêm vào Database dựa trên bảng mẫu mà bạn đã chọn.

## 2.5. Thiết lập khóa chính cho bảng

Nếu không có một lý do thật đặc biệt thì khi thiết kế ra một bảng dữ liệu bạn cần phải chỉ rõ một khóa chính. Access tự động tạo ra Index cho khóa chính, nó có thể cải thiện được tốc độ truy cập cũng như thao tác với dữ liệu. Access cũng chắc chắn rằng mỗi bản ghi thì trường khóa chính luôn chứa dữ liệu và giá trị này luôn luôn là duy nhất. Nếu không có trường này sẽ chẳng có cách nào đáng tin cậy để phân biệt một hàng cụ thể nào đó với những hàng khác trong bảng.



Khi bạn tạo ra một bảng mới trong chế độ hiển thị Datasheet View, Access sẽ tự động tạo ra một khóa chính và định nghĩa nó với tên là ID và kiểu dữ liệu là AutoNumber.

Ở chế độ hiển thị Design View, bạn có thể thay đổi, loại bỏ khóa chính, hoặc thiết lập cho một bảng chưa có khóa chính.

#### **Xác định trường đặt làm khóa chính**

Đôi khi, bạn có thể tự xác định được trường mà mình muốn sử dụng làm khóa chính. Chẳng hạn, bạn có thể có những số ID dành cho mỗi nhân công. Nếu bạn tạo ra một bảng để theo dõi thông tin về những người công nhân, bạn có thể quyết định sử dụng ID làm khóa chính cho bảng đó. Hoặc là sự kết hợp giữa ID của người công nhân với Id phòng ban để tạo nên tính duy nhất, yêu cầu rằng bạn sử dụng cả trường này để tạo nên khóa chính. Một hoặc nhiều trường muốn sử dụng làm trường khóa phải đảm bảo một vài nguyên tắc sau đây:

- Mỗi bản ghi phải có một hoặc nhiều trường kết hợp để tạo ra một giá trị duy nhất cho mỗi bản ghi.

- Một trường hoặc sự kết hợp giữa nhiều trường phải không bao giờ chứa giá trị rỗng (luôn luôn có giá trị)

- Những giá trị không thay đổi.

- Nếu không có dữ liệu thích hợp tồn tại để tạo ra khóa chính bạn có thể tạo ra một trường mới chỉ để làm khóa chính. Khi bạn tạo ra một trường mới sử dụng làm khóa chính bạn hãy thiết lập kiểu dữ liệu cho nó là AutoNumber. Với kiểu dữ liệu này trường sẽ nhận dữ liệu kiểu số, nằm trong dãy số nguyên liên tiếp, và hoàn toàn không trùng lặp. Hơn nữa trường này sẽ tự động tăng khi thêm một bản ghi mới giúp bạn không cần phải quan tâm việc tạo ra dữ liệu duy nhất cho trường khóa chính.

#### **Thiết lập hoặc thay đổi khóa chính**

Chọn bảng có khóa chính mà bạn muốn thiết lập hoặc thay đổi.

Trên Tab Home chọn nhóm Views, Click chọn View trong hộp thoại dropdown này chọn Design View.

Bảng xuất hiện ở chế độ Design View, hãy chọn trường (hoặc những trường) mà bạn muốn sử dụng làm khóa chính.

Field Name	Data Type	
MaHS	Text	Mã học sinh
HoHS	Text	Họ và họ đệm
TenHS	Text	Tên học sinh

Để lựa chọn một trường hãy Click chọn vào cột Row selector (cột đầu tiên màu xanh bên trái mỗi dòng) của trường đó. Nếu muốn chọn nhiều trường bạn hãy giữ phím Ctrl và lần lượt chọn tới các trường cần thiết lập. → Trên Tab Table trong nhóm Tools chọn Primary Key.

Một biểu tượng hình chìa khóa sẽ xuất hiện bên trái của những trường mà bạn vừa thiết lập khóa chính

### **Loại bỏ khóa chính**

Chọn bảng cần loại bỏ khóa chính trong Navigation Pane.

Tên Tab Home tìm tới nhóm Views Click chuột chọn nút View, trong danh sách đổ xuống này chọn Design View.

Click chuột chọn tới trường hoặc danh sách trường đang là những trường khóa chính (cách làm tương tự phần trước)

Trên Tab Design tìm tới nhóm Tools chọn Primary Key.

Biểu tượng chia khóa bên phía trái của các cột này sẽ được loại bỏ.

**Chú ý:** Khi bạn lưu trữ một bảng mới mà không được thiết lập khóa chính, Access sẽ nhắc cần phải tạo mới một trường làm khóa chính. Nếu bạn chọn Yes, Access tạo ra một trường có tên ID và gán cho nó kiểu dữ liệu là AutoNumber để cung cấp một giá trị duy nhất cho mỗi bản ghi. Trong trường hợp bảng đã có sẵn một trường kiểu AutoNumber, Access sẽ chọn luôn trường này làm khóa chính. Nếu bạn chọn No trên hộp thông báo Access sẽ không thêm trường và không thiết lập khóa chính cho bảng.

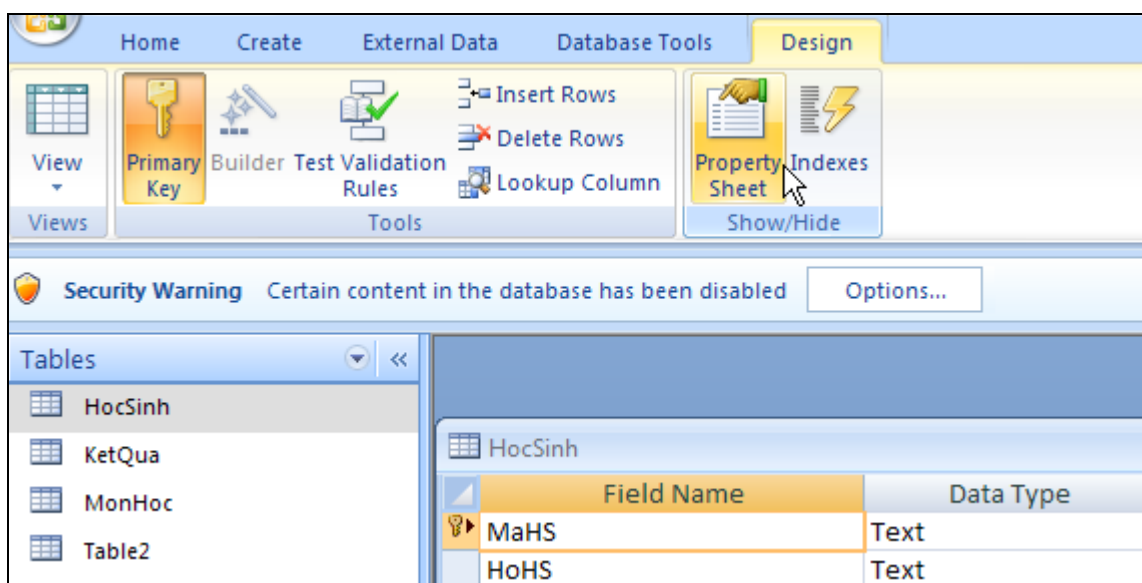
### **2.6. Thiết lập thuộc tính cho bảng**

Ngoài việc thiết lập thuộc tính cho các trường bạn còn có thể thiết lập thuộc tính chung cho bảng hoặc tất cả các bản ghi.

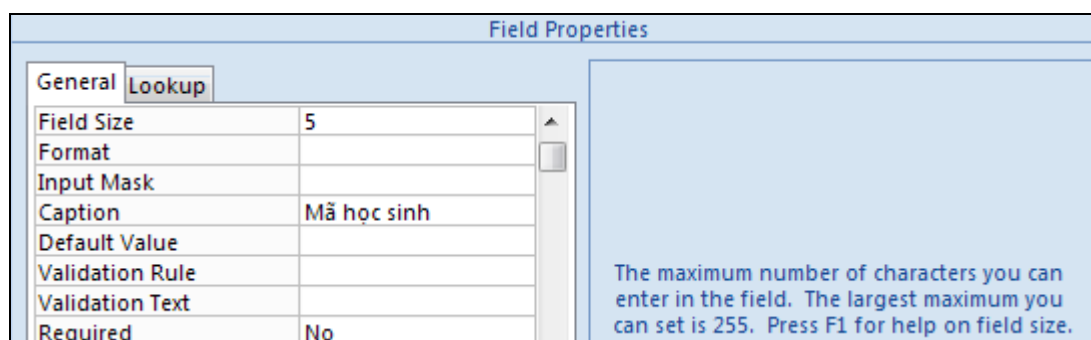
Chọn tới bảng cần thiết lập thuộc tính

Trên Tab Home, tìm tới nhóm Views, Click chọn Views trong hộp thoại đổ xuống bạn chọn Design View.

Trên Tab Design, tìm tới nhóm Show/Hide, Click chọn Property Sheet.



Property Sheet sẽ xuất hiện như hình dưới đây:



Trên Pane Property Sheet Click chọn Tab General.  
 Chọn cột bên trái để tìm tới thuộc tính của bảng và cột bên phải để thiết lập giá trị.

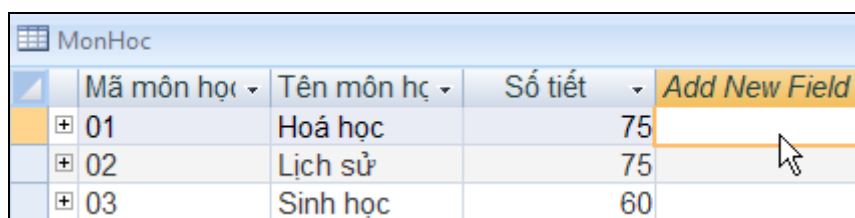
## 2.7 Thêm trường vào bảng

Bạn lưu trữ từng phần dữ liệu mà bạn muốn theo dõi trong một trường. Ví dụ, ở bảng Contacts bạn tạo ra danh sách trường để lưu lần lượt các thông tin như Last Name, First Name, Telephone Number, and Address. Hay trong bảng Procedures là các trường Product Name, Product ID, và Price.

Trước khi tạo ra các trường, dữ liệu được phân chia thành những phần nhỏ. Đó là cách dễ nhất để có thể xử lý dữ liệu được linh hoạt. Ví dụ, thay vì tạo ra một trường Full Name lưu trữ đầy đủ họ tên của một người nào đó, chúng ta nên tạo ra hai trường riêng biệt là First Name và Last Name. Sau đó thì có thể dễ dàng tìm kiếm, sắp xếp theo từng trường hoặc theo cả hai trường.

## 2.8. Thêm trường khi đang nhập dữ liệu

Khi bạn tạo một bảng mới hoặc mở một bảng đã tồn tại ở chế độ hiển thị Datasheet View, bạn có thể thêm một trường vào bảng bằng cách nhập dữ liệu vào cột Add New Field trên Datasheet.



	Mã môn học	Tên môn học	Số tiết	Add New Field
+	01	Hoá học	75	
+	02	Lịch sử	75	
+	03	Sinh học	60	

Bảng dữ liệu được tạo mới hoặc được mở ở chế độ Datasheet View.

(1): Nhập dữ liệu vào cột Add New Field.

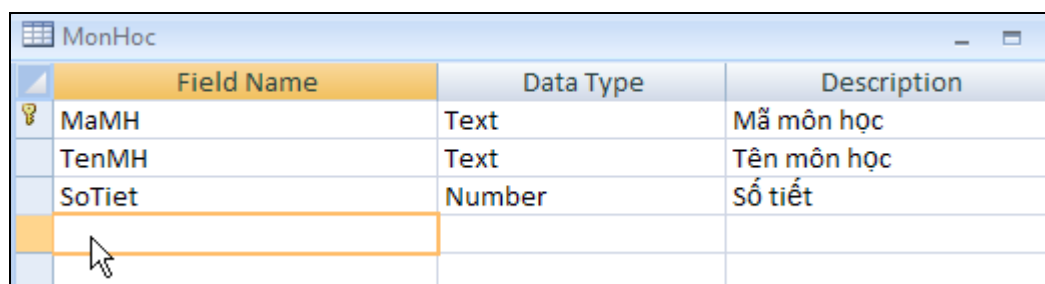
Trên phân tiêu đề của cột Add New Field hãy nhập vào tên trường mà bạn muốn tạo, tên này sẽ mô tả trường và giúp bạn dễ dàng nhận ra nó hơn.

Cuối cùng nhập dữ liệu vào trường đó.

## 2.9 Thêm trường sử dụng trường Template

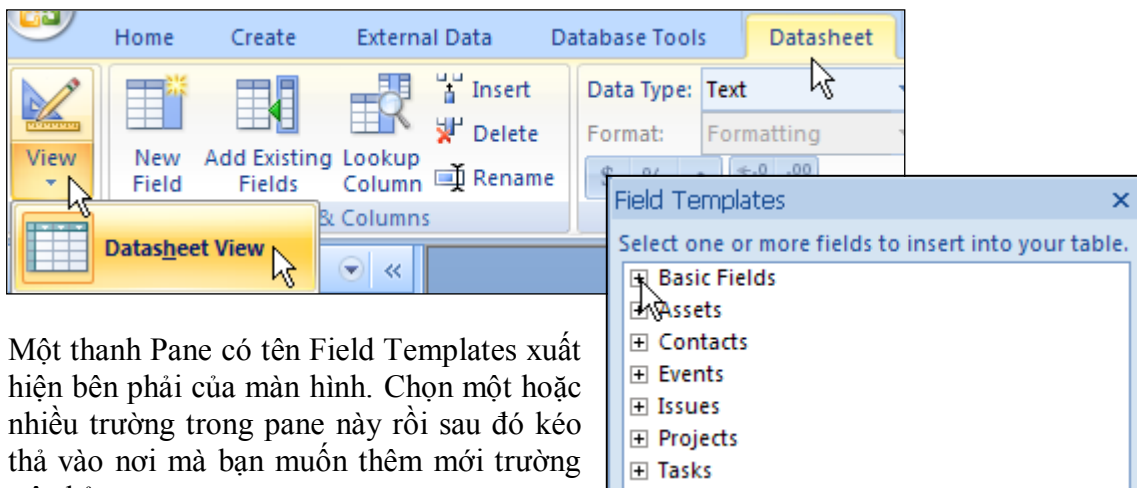
Trong một số trường hợp thật dễ dàng lựa chọn một trường phù hợp với yêu cầu của bạn trong danh sách các trường đã được thiết lập sẵn thay vì việc bạn phải tự tạo ra nó. Bạn có thể sử dụng Field Templates task pane để chọn một trong những trường mẫu. Các trường mẫu này đã được thiết lập sẵn các thuộc tính cần thiết. Ví dụ như tên trường, kiểu dữ liệu, các thuộc tính định dạng, và nhiều thuộc tính khác.

Trên Tab Home, tìm tới nhóm Views, Click chọn View, trên hộp thoại đổ xuống bạn chọn Datasheet View.



Field Name	Data Type	Description
MaMH	Text	Mã môn học
TenMH	Text	Tên môn học
SoTiet	Number	Số tiết

Trên Tab Datasheet, tìm tới nhóm Fields & Columns, Click chọn New Field.

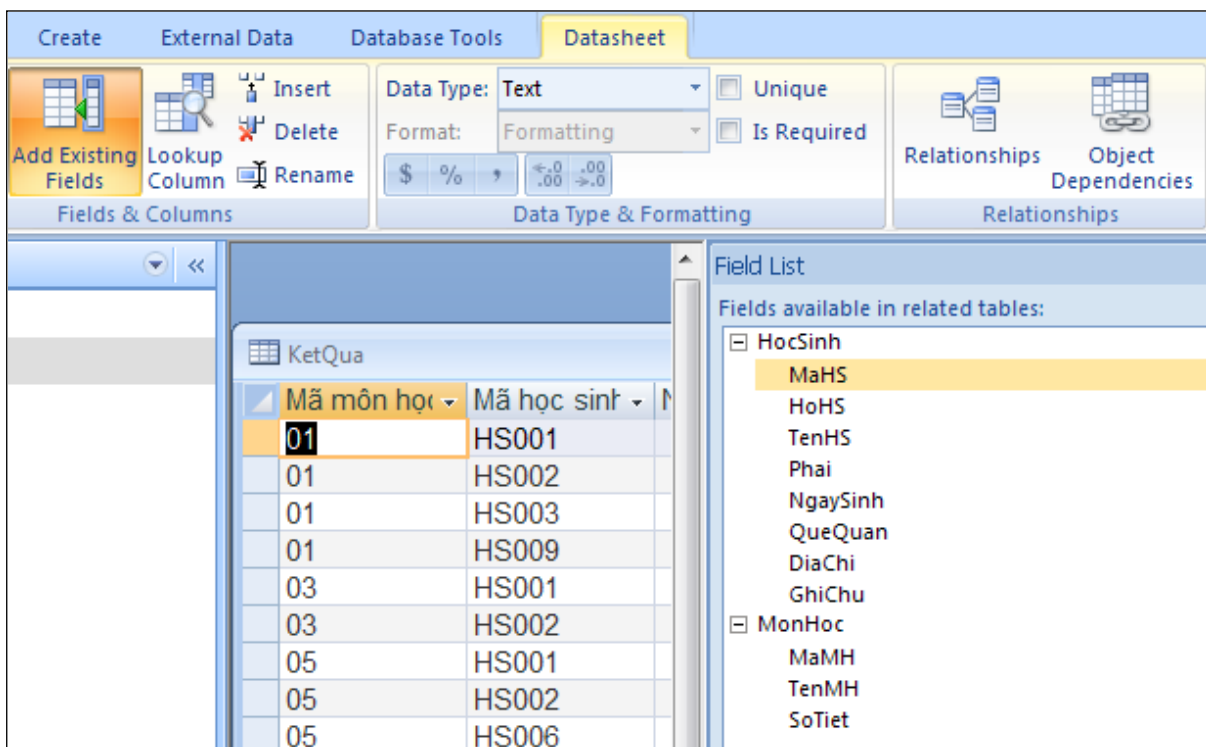


Một thanh Pane có tên Field Templates xuất hiện bên phải của màn hình. Chọn một hoặc nhiều trường trong pane này rồi sau đó kéo thả vào nơi mà bạn muốn thêm mới trường trên bảng.

## 2.10. Thêm trường từ trường của bảng khác

Nếu cơ sở dữ liệu của bạn đã có sẵn các Table, bạn có thể thêm trường vào một bảng mới từ danh sách các trường của một trong những bảng này, chúng được liệt kê trong Pane Field List. Pane Field List liệt kê tất cả các bảng trong cơ sở dữ liệu, chi tiết trong các bảng này là danh sách trường.

Nếu bảng mà bạn thêm trường mới có mối quan hệ với các bảng khác, thì các trường quan hệ sẽ được liệt kê đầu tiên trong Field List Pane



Click chọn nút Microsoft Office Button , tiếp theo chọn Open.

Trong hộp thoại Open, chọn tới Database bạn cần làm việc.

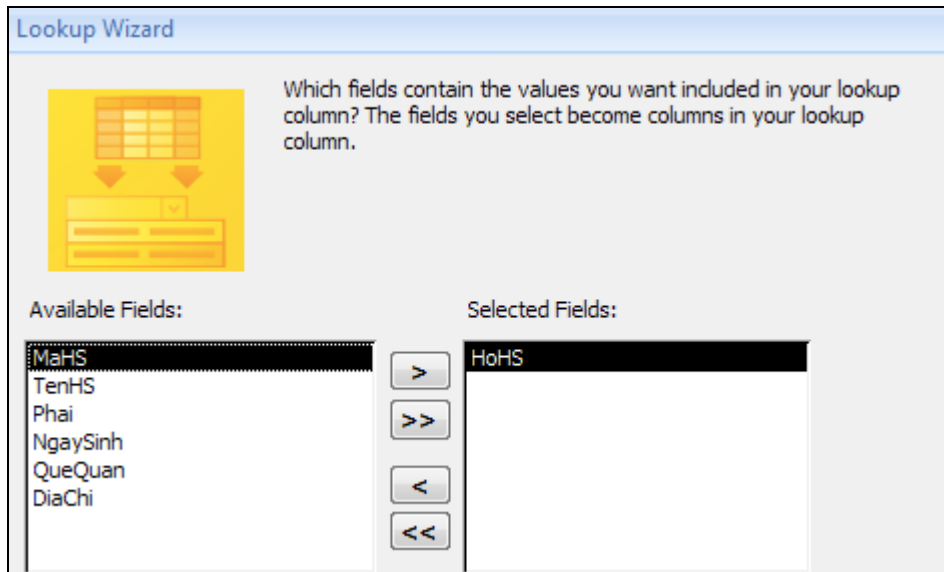
Trong Pane Navigation, Double Click chuột vào bảng cần thêm trường mới.

Bảng được mở ở chế độ Datasheet View.

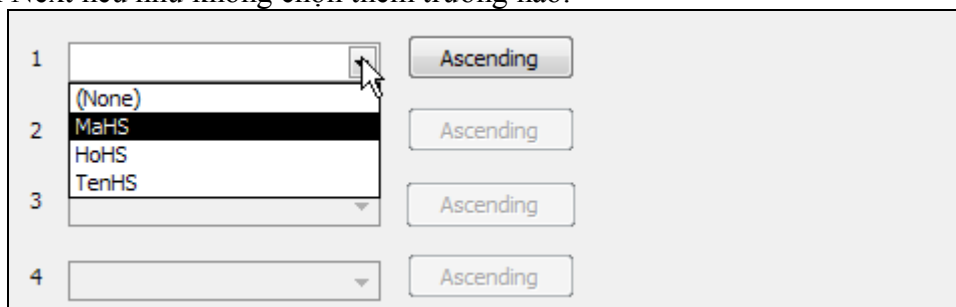
Trên Tab Datasheet, trong nhóm Fields & Columns Click chọn Add Existing Fields  
Pane Field List xuất hiện

Click vào biểu tượng (+) trong danh sách tại bảng cần lấy trường, một danh sách trong bảng đó sẽ xuất hiện, bạn chọn tới trường cần lấy rồi kéo thả đến vị trí cần thêm trường trên bảng mới.

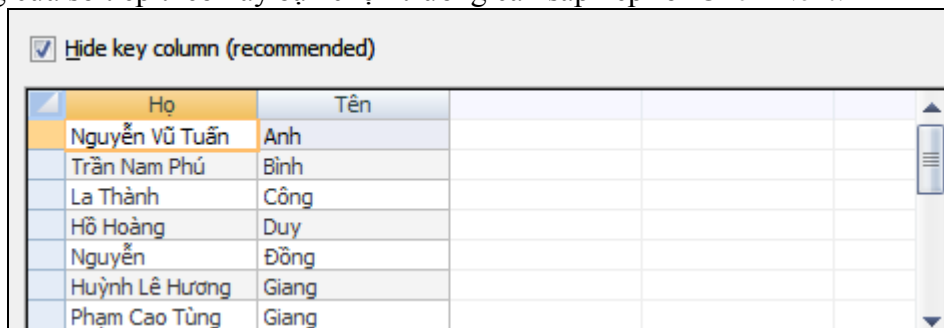
Cửa sổ Lookup Wizard xuất hiện



Click Next nếu như không chọn thêm trường nào.



Trong cửa sổ tiếp theo này bạn chọn trường cần sắp xếp rồi Click Next.



Bạn Click Next để chuyển sang bước tiếp theo

Nhập lại tên cho cột nếu cần thiết, cuối cùng Click Finish để kết thúc.

Như vậy một trường mới sẽ được thêm vào bảng

What label would you like for your lookup column?

HoHS

Do you want to store multiple values for this lookup?

Allow Multiple Values

Mã môn học	Mã học sinh	HoHS	Ngày kiểm t	Điểm
01	HS001		24/11/2009	7.8
01	HS002		15/09/2009	7.0
01	HS003		15/09/2009	6.3
01	HS009		24/11/2009	8.5

## 2.11. Thiết lập thuộc tính cho trường

Sau khi tạo một trường mới, để có thể điều khiển các hành vi, sự kiện diễn ra đối với trường này bạn cần phải thiết lập thuộc tính cho nó.

Ví dụ, việc thiết lập thuộc tính cho trường sẽ giúp bạn:

- Điều khiển được sự hiển thị dữ liệu trên trường đó.
- Giúp ngăn ngừa việc nhập dữ liệu không chính xác.
- Gán giá trị mặc định cho trường.
- Giúp tăng tốc độ tìm kiếm sắp xếp dữ liệu.

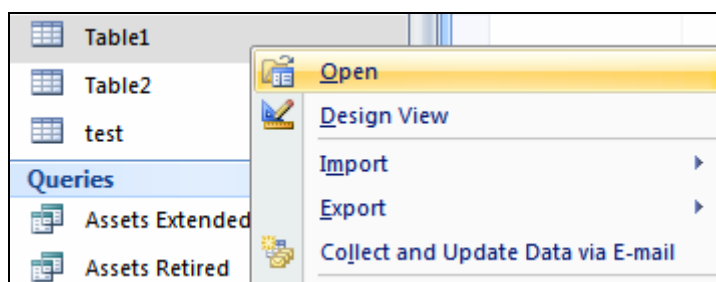
Bạn có thể thiết lập một số thuộc tính sẵn có của trường trong khi đang làm việc ở chế độ Datasheet View. Tuy nhiên để có thể thiết lập đầy các thuộc tính cho trường bạn nên sử dụng chế độ hiển thị Design View.

### \* Ở chế độ Datasheet View

Ở chế độ Datasheet View bạn có thể đổi tên, dữ liệu, định dạng, và một số thuộc tính khác của một trường.

### Mở bảng ở chế độ Datasheet View

Trên Pane Navigation, Click phải chuột vào bảng cần mở, trong menu popup chọn Open.



### Đổi tên trường

Khi bạn thêm một trường bằng cách nhập thẳng dữ liệu vào bảng ở chế độ Datasheet View, Access sẽ tự động định nghĩa một cái tên chung chung cho trường, ví dụ như bắt đầu với trường thứ nhất là Field1, các trường tiếp theo sẽ là Field2, Field3, ... Mặc định Access sử dụng tên của trường như một Label ở bất cứ nơi nào mà trường được hiển thị, ví dụ như tiêu đề cột khi mở bảng ở chế độ Datasheet View. Việc đổi tên trường sẽ giúp trường có thêm những thông tin mô tả về nó, nhờ đó mà giúp cho người dùng dễ dàng hơn trong quá trình sử dụng và chỉnh sửa.

Click phải chuột vào phần tiêu đề cột của cột cần đổi tên, chọn Rename trong popup menu.

Nhập vào tên cần sửa đổi, lưu ý tên trường không được quá 64 ký tự bao gồm cả các khoảng trắng.

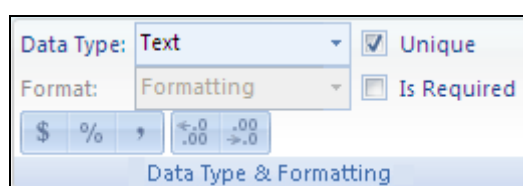
### **Thay đổi kiểu dữ liệu của trường**

Khi bạn tạo ra một trường bằng cách nhập dữ liệu vào trường ở chế độ hiển thị Datasheet View, Access sẽ kiểm tra dữ liệu đó để xác định kiểu dữ liệu thích hợp cho trường. Chẳng hạn, nếu bạn nhập vào giá trị 1/ 1/ 2006, Access sẽ đoán rằng đó là dữ liệu ngày tháng và thiết lập kiểu dữ liệu cho trường này là Date/Time. Nếu như với một trường mà dữ liệu bạn nhập vào cho nó thuộc nhiều kiểu khác nhau, khi đó Access sẽ không thể xác định được trường đó thuộc kiểu gì và nó tự động gán cho trường đó là dữ liệu kiểu Text.

Kiểu dữ liệu của trường sẽ quyết định những thuộc tính mà nó có thể có. Ví dụ bạn chỉ có thể thiết lập thuộc tính Append Only cho trường kiểu Hyperlink và trường kiểu Memo.

Có rất nhiều trường hợp Access hiểu lầm kiểu dữ liệu mà bạn muốn thiết lập và bạn phải thay đổi lại kiểu dữ liệu cho trường đó. Ví dụ, bạn có một trường lưu số phòng trong trong khách sạn, bạn nhập dữ liệu cho trường là 10/2001, với chức năng tự tìm kiếm Access sẽ kiểm tra và gán cho trường này kiểu dữ liệu ngày tháng, nhưng thực tế bạn chỉ muốn lưu dữ liệu trường này kiểu Text. Để sửa lại kiểu dữ liệu cho trường này bạn làm như sau:

- Trên Ribbon Click chọn Tab Datasheet.
- Trong danh sách Data Type của nhóm Data Type & Formatting bạn chọn kiểu dữ liệu mà bạn muốn.



Các kiểu dữ liệu sẵn có trong Access.

<b>Data type</b>	<b>Use to store</b>	<b>Size</b>
Text	Sử dụng cho văn bản hoặc số nhưng không dùng để tính toán.	Tối đa 255 ký tự
Memo	Sử dụng lưu trữ ký tự, văn bản, hoặc các định dạng Rich Text.	Tối đa 01 GB cho ký tự.
Number	Lưu trữ giá trị số nguyên.	1, 2, 4, hoặc 8 bytes, hoặc 16 bytes (ID)
Date/Time	Dữ liệu kiểu ngày tháng	8 bytes.
Currency	Lưu giá trị kiểu tiền tệ	8 bytes.
AutoNumber	Giá trị nguyên được tự động thêm vào khi Insert một bản ghi.	4 bytes, 16 bytes
Yes/No	Giá trị kiểu Logic	1 bit (8 bits = 1 byte).
OLE Object	Lưu trữ đối tượng nhúng.	Tối đa 1 gigabyte.
Attachment	Lưu trữ Pictures, Images, Binary files, Office files.	Có thể lên tới 2 gigabytes
Hyperlink	Lưu trữ các liên kết.	Có thể lên tới 2Gb
Lookup Wizard	Sử dụng khi bạn muốn tạo ra trường Look up (tìm kiếm). hiển thị dưới dạng một Combo	Tương ứng giá trị tìm kiếm trả về

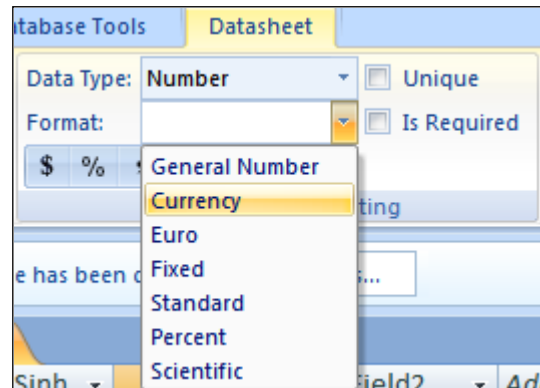
**Chú ý:** Dung lượng tối đa mà Access dành cho một Database là 2 Gigabyte.

### ***Thay đổi định dạng của trường***

Ngoài việc thiết lập kiểu dữ liệu cho một trường mới bạn còn có thể thiết lập thuộc tính Format cho trường đó. Ví dụ, nếu bạn nhập vào 10 giờ 50 phút sáng, Access sẽ thiết lập kiểu dữ liệu là Data/Time và thuộc tính Format là Medium Time. Để thay đổi thuộc tính Format của trường này bạn làm như sau:

Trên Ribbon, Click chọn tab Datasheet.

Trong danh sách Format, trong nhóm Data Type & Formatting, chọn kiểu định dạng mà bạn muốn thiết lập.

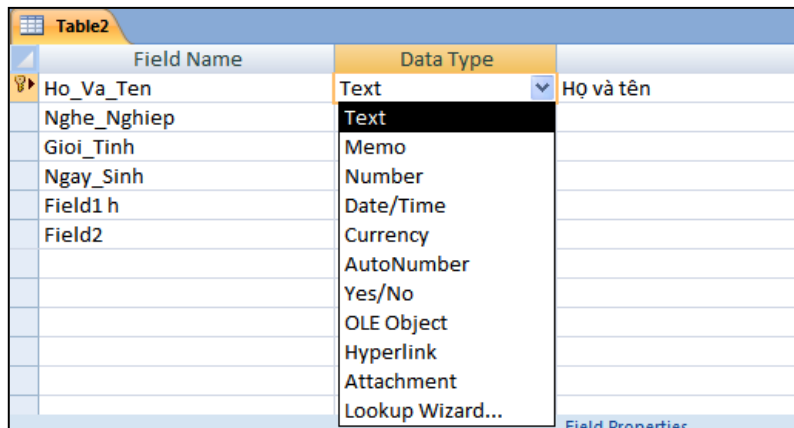


***Chú ý:*** Trong danh sách Format chỉ có những giá trị sẵn có và nó thay đổi tùy thuộc vào kiểu dữ liệu của trường đó.

### ***Ở chế độ Design View***

Khác với chế độ hiển thị Datasheet View, ở chế độ Design View bạn có thể thiết lập cho trường bất cứ thuộc tính nào mà bạn muốn. Ở chế độ này, bạn thiết lập kiểu dữ liệu cho trường trong bảng lưới.

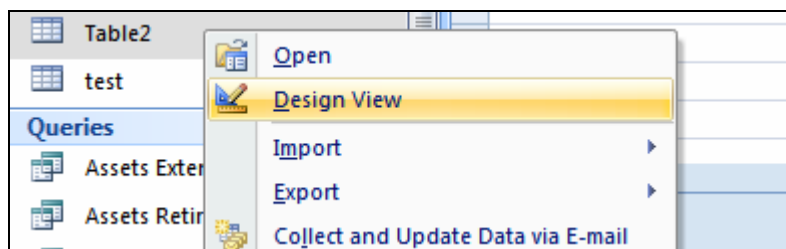
Và thiết lập thuộc tính khác cho trường trên Pane Field Properties.



### ***Mở bảng ở chế độ Design View***

Để thiết lập kiểu dữ liệu cho trường ở chế độ Design View bạn làm như sau:

Trên Pane Navigation, Click phải chuột vào bảng cần mở chọn Design View.



Bảng xuất hiện ở chế độ Design View, trong lưới thiết kế bạn chọn tới trường cần thay đổi kiểu dữ liệu. Trong cột Data Type, lựa chọn lựa chọn kiểu dữ liệu từ danh sách đổ xuống.

### ***Thiết lập các thuộc tính khác cho trường***

Trong bảng lưới thiết kế, chọn trường mà bạn muốn thiết lập thuộc tính. Access sẽ hiển thị những thuộc tính của trường đó trong Pane Field Properties.



Field Name	Data Type	
MaMH	Text	Mã môn học
MaHS	Text	Mã học sinh
HoHS	Text	
NgàyKT	Date/Time	Ngày kiểm tra
Diem	Number	Điểm

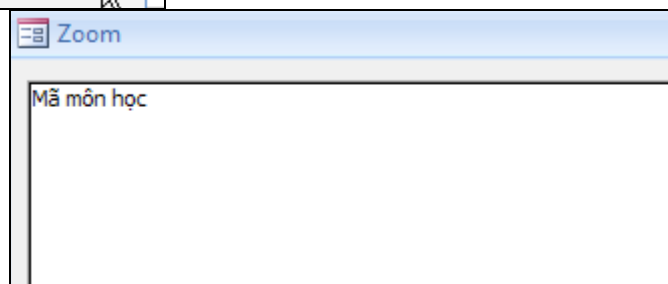
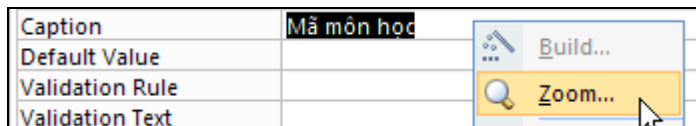
  

Field Properties	
General	Lookup
Field Size	2
Format	
Input Mask	
Caption	Mã môn học
Default Value	
Validation Rule	

**Chú ý:** Kiểu dữ liệu của trường xác định nên những thuộc tính mà trường đó có thể có.


Trong Pane Field Properties, chọn tới những thuộc tính cần thiết lập ở cột thứ nhất và điền lại giá trị ở cột tiếp theo, bạn cũng có thể sử dụng phím F6 để chuyển đổi qua lại giữa lưới danh sách trường phía trên và Pane Field properties.

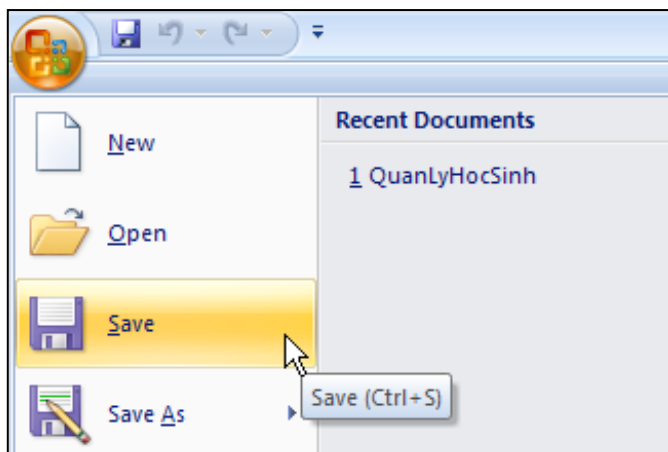
Khi làm việc với những thông tin dạng Text bạn có thể sử dụng phím Shift + f2 để bật hộp thoại Zoom, với hộp thoại này bạn sẽ có nhiều không gian hơn để soạn thảo. Ví dụ trong trường hợp này tôi đặt trỏ chuột vào dòng đầu tiên của cột Description rồi Click Shift + F2 hộp thoại Zoom xuất hiện như hình dưới đây:



### Lưu lại bảng

Sau khi đã tạo, chỉnh sửa bảng, bạn nên lưu lại các thiết kế đó. Khi bạn lưu bảng ở lần đầu tiên, hãy nhập vào tên cho bảng qua hộp thoại thông báo của Access. Tên bảng có ý nghĩa mô tả nội dung dữ liệu mà bảng đó đang chứa. Bạn có thể nhập tên bảng là các ký tự hoặc chữ số và bao gồm cả khoảng trắng nhưng không được quá 64 ký tự.

Để lưu lại bảng bạn có thể Click trực tiếp tổ hợp phím Ctrl + S hoặc từ nút Microsoft Office Button , bạn chọn Save.



## Bài 3. Làm việc với Query

Khi bạn muốn thêm, sửa đổi hoặc xóa dữ liệu trong Database, bạn nên sử dụng các query.

Với việc sử dụng Query, bạn có thể trả lời được rất nhiều câu hỏi cụ thể về dữ liệu mà bạn hiện có, trong khi việc đó sẽ rất khó khăn nếu bạn truy cập thẳng vào các bảng. Bạn có thể sử dụng các Query để lọc dữ liệu, thực hiện những phép tính với dữ liệu hoặc tóm tắt dữ liệu. Bạn cũng có thể sử dụng các Query để tự động quản lý dữ liệu và xem xét các thay đổi trong dữ liệu trước khi cho phép thực hiện những thay đổi đó.

Một Query là một yêu cầu đến cơ sở dữ liệu và kết quả trả về là dữ liệu hoặc hoạt động (Action) diễn ra trên cơ sở dữ liệu đó. Bạn có thể sử dụng một Query (truy vấn) để trả lời một câu hỏi đơn giản, thực hiện phép tính, kết hợp dữ liệu từ các bảng khác nhau, hoặc thậm chí có thể thêm, thay đổi, hoặc xóa các bảng dữ liệu. Các truy vấn mà bạn sử dụng để lấy dữ liệu từ một bảng hoặc để thực hiện các phép tính được gọi là truy vấn lựa chọn. Các truy vấn mà thêm, thay đổi, hoặc xóa dữ liệu được gọi là truy vấn hành động.

Trong phần này chúng tôi sẽ cung cấp cho các bạn những thông tin cơ bản về các truy vấn cùng với nó là các ví dụ về các loại hình khác nhau của các truy vấn.

Bạn cũng có thể sử dụng truy vấn để cung cấp dữ liệu cho một Form hoặc một Report. Một cơ sở dữ liệu được thiết kế tốt thì dữ liệu mà bạn muốn để hiển thị lên Form hoặc Report thường nằm ở một số bảng khác nhau. Bằng cách sử dụng truy vấn, bạn có thể liên kết các dữ liệu mà bạn muốn sử dụng lại với nhau trước khi đưa lên Form hoặc Report.

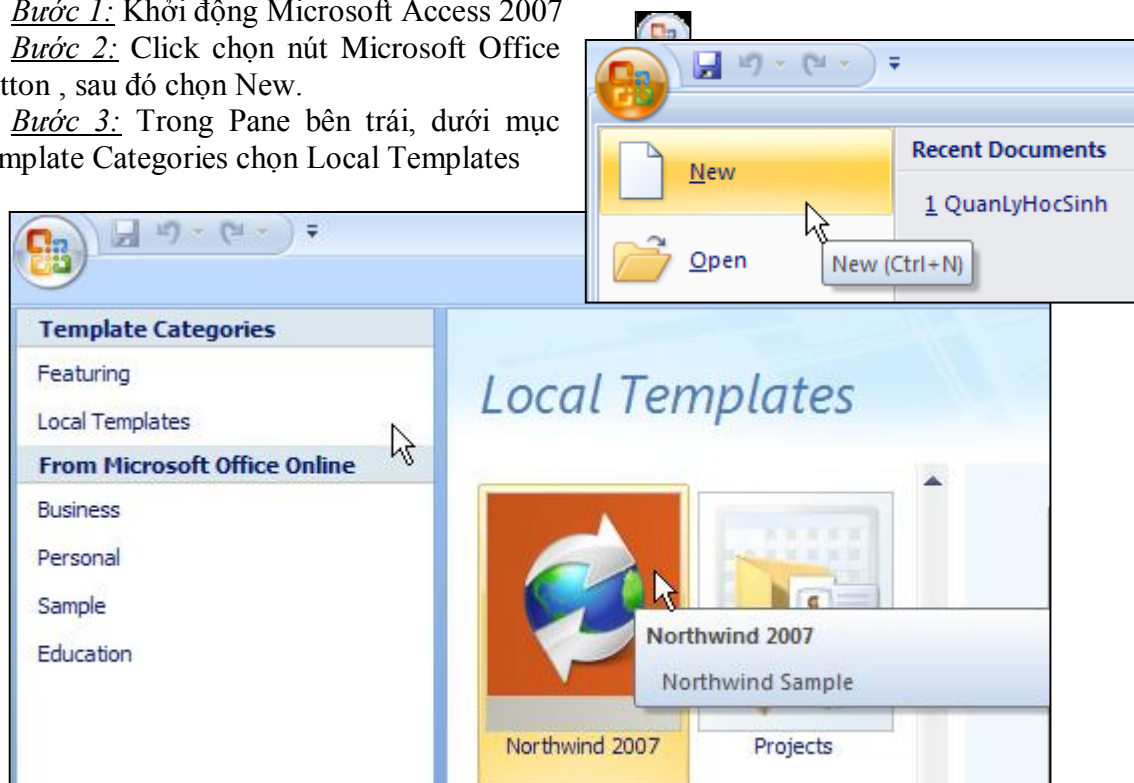
**Chú ý:** Các ví dụ trong bài này sử dụng Database Northwind có sẵn trong Template của Microsoft Access 2007.

Để thiết lập Northwind 2007 bạn làm theo các bước sau:

**Bước 1:** Khởi động Microsoft Access 2007

**Bước 2:** Click chọn nút Microsoft Office Button, sau đó chọn New.

**Bước 3:** Trong Pane bên trái, dưới mục Template Categories chọn Local Templates



**Bước 4:** Trong phần Local Template chọn Northwind 2007 sau đó Click Create.

**Bước 5:** Tiếp theo trang Northwind Traders xuất hiện bạn Click nút Close để đóng cửa sổ này lại.

## 1. Tìm kiếm một nhóm dữ liệu trong bảng - SELECT

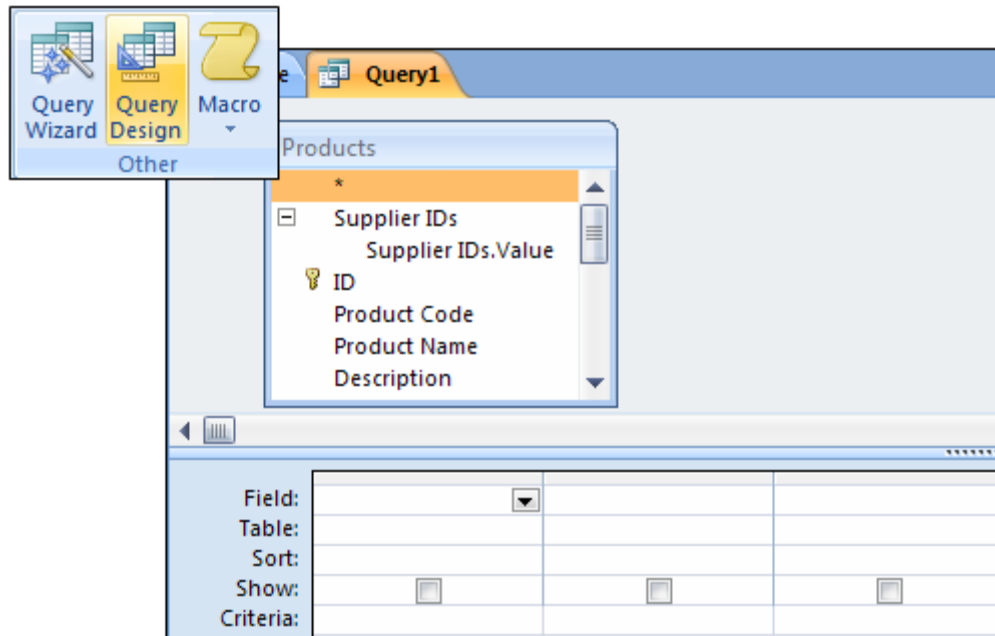
Đôi khi bạn muốn xem xét tất cả dữ liệu từ một bảng, nhưng ở những lần khác, bạn lại chỉ muốn xem dữ liệu ở một số trường, hoặc bạn lại muốn xem một số trường nhưng phải đáp ứng một vài tiêu chí nào đó, ... Với những yêu cầu như vậy bạn có thể sử dụng các truy vấn.

Giả sử bạn muốn xem xét một danh sách các sản phẩm và giá cả của chúng. Bạn có thể tạo một truy vấn mà trả về thông tin sản phẩm và giá cả bằng cách sử dụng thủ tục sau đây:

Bước 1: Mở Northwind 2007.

Bước 2: Trên Tab Create tìm tới nhóm Other Click chọn Query Design.

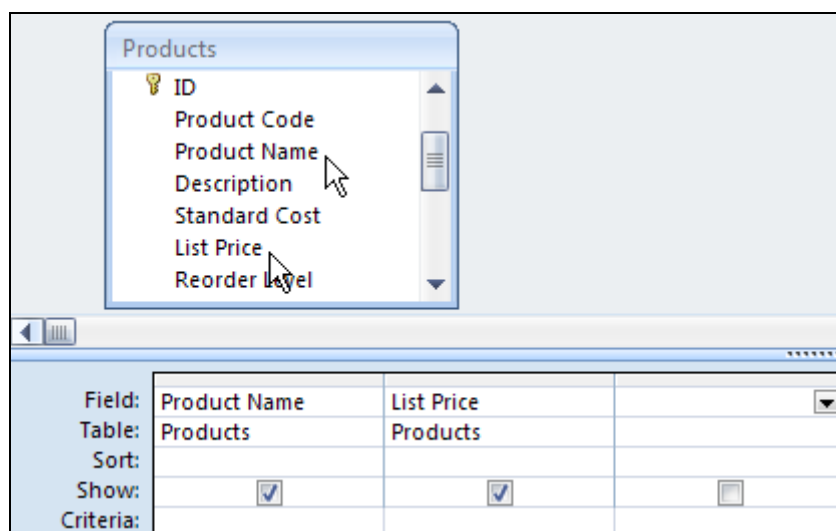
Bước 3: Trên hộp thoại Show Table tìm tới Tab Tables chọn Products rồi Click Add.



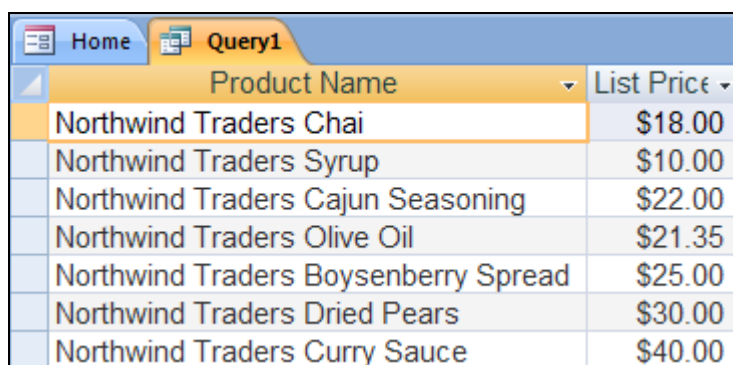
Bước 4: Click Close để đóng hộp thoại Show Table.

Bước 5: Trên màn hình thiết kế Query tìm tới bảng Products, Double Click chuột lên hai trường là Product Name và List Price để thêm trường đó vào lưới thiết kế Query.

Sau khi Click chọn hai trường trong bảng Products kết quả sẽ được như hình dưới đây:



Bước 6: Trên Tab Design trong nhóm Results Click chọn View để xem Kết quả



Product Name	List Price
Northwind Traders Chai	\$18.00
Northwind Traders Syrup	\$10.00
Northwind Traders Cajun Seasoning	\$22.00
Northwind Traders Olive Oil	\$21.35
Northwind Traders Boysenberry Spread	\$25.00
Northwind Traders Dried Pears	\$30.00
Northwind Traders Curry Sauce	\$40.00

## 2. Xem dữ liệu đồng thời từ nhiều bảng

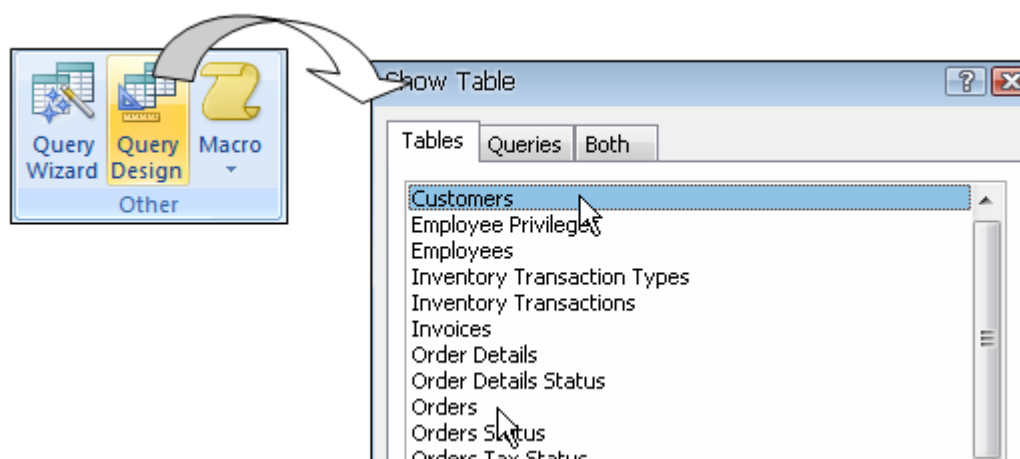
Những bảng biểu được thiết kế tốt thường mang theo các mối quan hệ với các bảng khác. Những mối quan hệ này tồn tại trên cơ sở các bảng có chung trường quan hệ. Khi bạn muốn xem dữ liệu từ nhiều bảng liên quan bạn có thể sử dụng truy vấn lựa chọn.

Giả mà bạn muốn xem xét lại đơn đặt hàng của một khách hàng sống tại thành phố nào đó. Dữ liệu về đơn đặt hàng và các dữ liệu về khách hàng được lưu trữ trong hai bảng trên cùng một cơ sở dữ liệu. Mỗi bảng có một Customer\_Id, và hai bảng này quan hệ với nhau ở dạng một nhiều. Bạn có thể tạo một truy vấn mà kết quả trả về là đơn hàng của một khách ở ở một thành phố cụ thể nào đó, ví dụ như Las Vegas, bằng cách thực hiện các bước dưới đây:

Bước 1: Mở Northwind 2007.

Bước 2: Trên Tab Create, trên nhóm Other, Click chọn Query Design.

Bước 3: Trong hộp thoại Show Table, trên Tab Tables Double Click chuột vào bảng Customers và Orders.

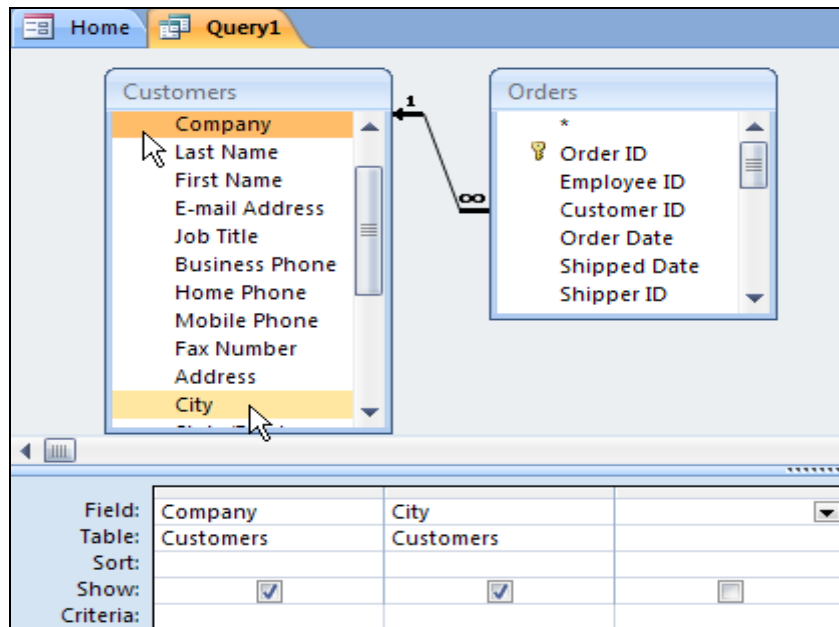


Bước 4: Click nút Close để đóng hộp thoại Show Table

Chú ý: Giữa hai bảng có một đường Line kết nối giữa trường Id của bảng khách hàng Customer Id của bảng Orders. Đây là đường Line hiển thị mối qua hệ giữa hai bảng.

Bước 5: Trong bảng Customers, Double Click chuột vào trường Company và City để thêm những trường đó vào phần lưới thiết kế Query.

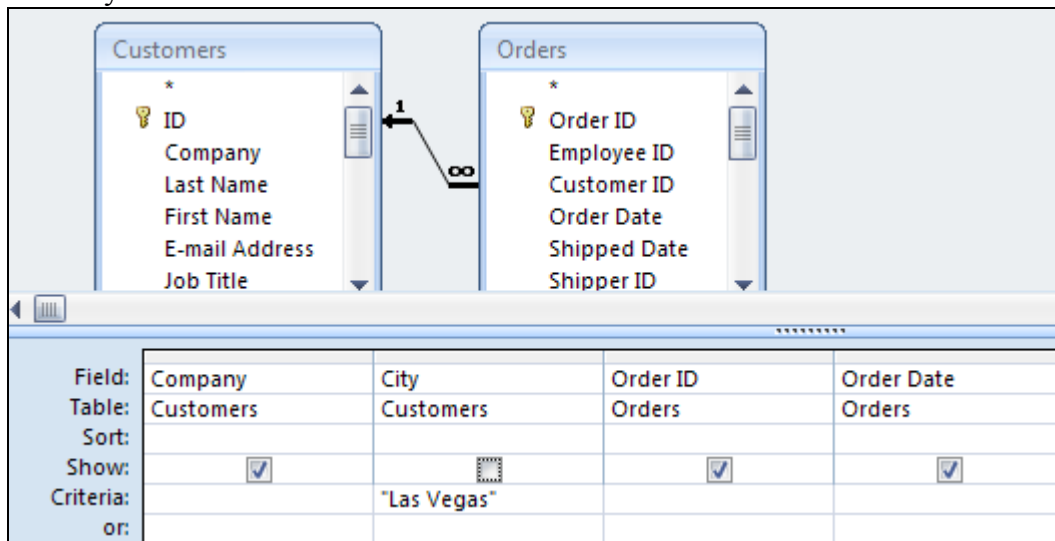
Bước 6: Trên lưới thiết kế Query, trên cột chứa trường City, bỏ check chọn trên hàng Show của cột này.



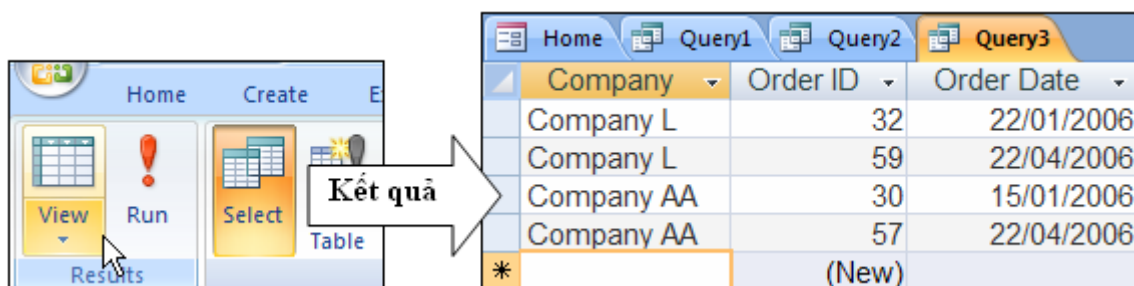
**Bước 7:** Trên dòng Criteria của cột City, bạn gõ vào giá trị là ‘Las Vegas’.

Xóa check chọn ở dòng Show để kết quả trả về của Query không hiển thị thông tin này, và gõ điều kiện ‘Las Vegas’ trên dòng Criteria để kết quả trả về của Query chỉ lấy những khách hàng ở thành phố Las Vegas.

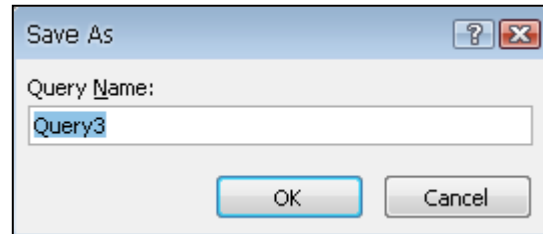
**Bước 8:** Trên bảng Orders Double Click vào trường Order Id và trường Order Date để bổ xung thêm 2 trường này lên lưới thiết kế Query. Kết quả trên lưới sẽ được hiển thị như hình dưới đây:



**Bước 9:** Trên Tab Design tìm tới nhóm Results Click View để xem kết quả



**Bước 10:** Click tổ hợp phím Ctrl + S để lưu lại Query, một hộp thoại thông báo xuất hiện bạn nhập vào tên cho Query là Orders by city rồi Click Ok.



### 3. Sử dụng Tham số với Query - PARAMETER

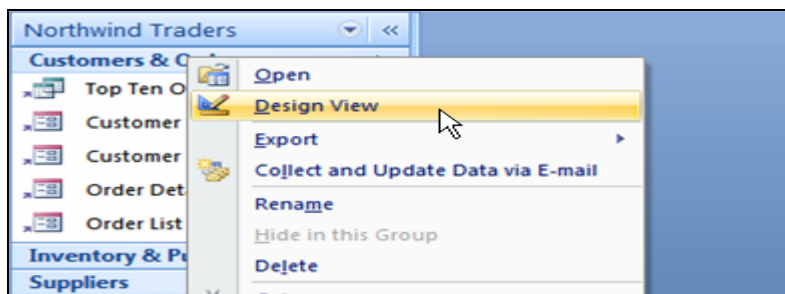
Đôi khi bạn muốn chạy một truy vấn mà chỉ hơi khác với một yêu cầu tìm kiếm hiện tại. Bạn có thể thay đổi truy vấn nguyên thủy để sử dụng điều kiện mới của bạn, nhưng nếu bạn thường xuyên muốn chạy các biến thể của một truy vấn cụ thể, hãy cân nhắc việc sử dụng một tham số truy vấn. Khi bạn chạy một tham số truy vấn, truy vấn sẽ nhắc bạn điền giá trị vào cho trường, nó sẽ sử dụng giá trị mà bạn vừa cung cấp để tạo ra các điều kiện (Criteria) cho truy vấn của bạn.

Trong ví dụ trước, bạn đã tạo ra một truy vấn mà kết quả trả về là những đơn hàng của những khách hàng ở thành phố Las Vegas. Bạn có thể chỉnh sửa truy vấn để thông báo cho người dùng nhập vào thành phố cần tìm, trong khi chạy truy vấn. Dưới đây là các bước tạo một truy vấn như vậy.

**Bước 1:** Mở Database Northwind 2007

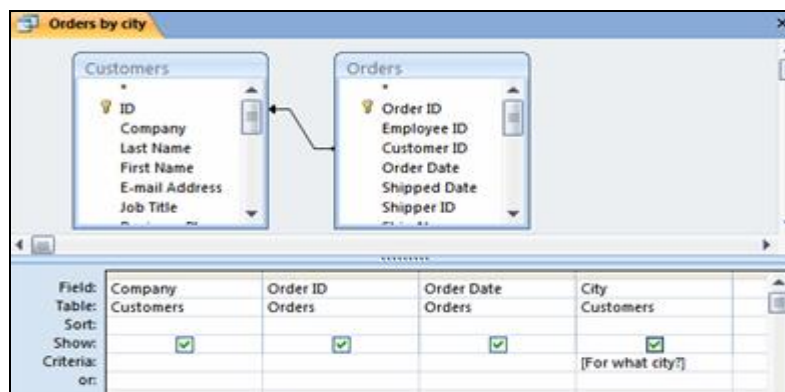
**Bước 2:** Click chọn Shutter Bar để hiển thị Pane Navigation

**Bước 3:** Trong Pane Navigation, Click phải chuột vào Query có tên Orders by city mà bạn đã tạo ở phần trước, menu xuất hiện chọn Design View.



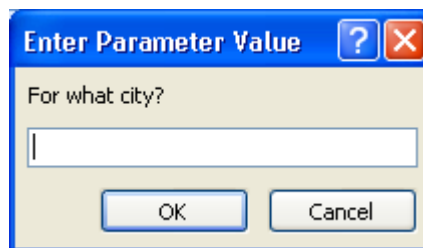
**Bước 4:** Trong lưới thiết kế Query, ở dòng Criteria của cột City xóa bỏ giá trị 'Las Vegas' nhập vào nội dung [For what city?], chuỗi [For what city?] là nội thông báo cho Parameters của bạn. Dấu ngoặc vuông cho thấy rằng bạn muốn truy vấn hiển thị thông báo về Parameter cần nhập vào, và nội dung văn bản (trong trường hợp này: For what city?) Là một câu hỏi hiển thị trên thông báo.

**Bước 5:** Click chọn dấu check trên dòng Show của cột City để hiển thị cột này trên kết quả của Query.



**Bước 6:** Trên Tab Design tìm tới nhóm Results, Click View để chạy thử Query. Access sẽ hiển thị hộp thoại yêu cầu nhập vào thành phố mà bạn muốn xem.

**Bước 7:** Giả sử ở đây bạn nhập New York rồi Click Enter chúng ta sẽ được kết quả như hình dưới đây.



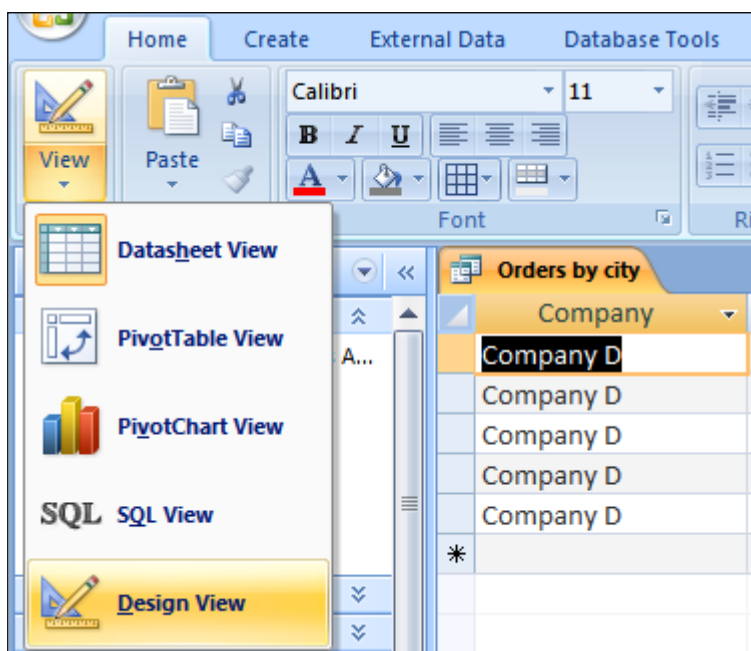
Company	Order ID	Order Date	City
Company D	31	1/20/2006	New York
Company D	34	2/6/2006	New York
Company D	58	4/22/2006	New York
Company D	61	4/7/2006	New York
Company D	80	4/25/2006	New York
	(New)		

**Bước 8:** Nếu như bạn không muốn nhập vào đó giá trị chính xác, để tham số có thể mềm dẻo hơn, bạn có thể sử dụng ký tự đại diện trong chuỗi điều kiện (Criteria). Để làm được điều này bạn thực hiện tiếp các bước dưới đây.

**Bước 9:** Từ Tab Home tìm tới nhóm Views chọn nút View trên nút lệnh đổ xuống bạn chọn Design.

**Bước 10:** Trên lưới thiết kế Query tại dòng Criteria của cột City bạn nhập vào nội dung dưới đây:

Like [For what city?]&"\*"



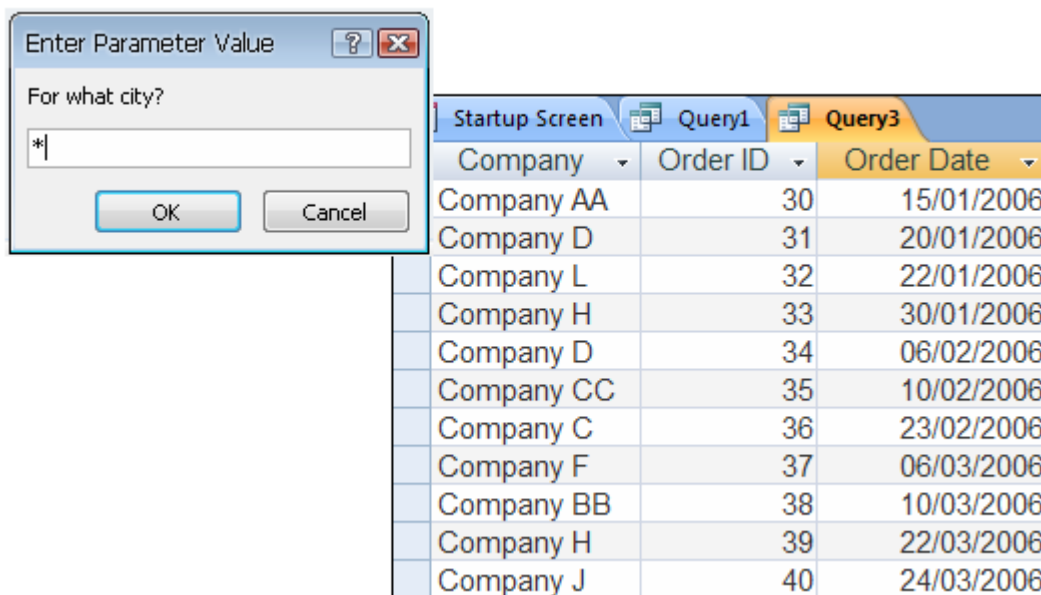
Field:	Company	Order ID	Order Date	City
Table:	Customers	Orders	Orders	Customers
Sort:				
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Criteria:				Like [For what city?] & "*"

Trong nội dung thông báo này, từ khóa Like, toán tử & và dấu (\*) cho phép cho phép người sử dụng gõ vào một giá trị mà có sự kết hợp giữa các ký tự với nhau để trả về nhiều

hơn các kết quả. Ví dụ, nếu người dùng gõ vào ký tự (\*), truy vấn sẽ trả về tất cả các thành phố, nếu người sử dụng các loại L, truy vấn sẽ tìm kiếm và trả lại tất cả các thành phố mà bắt đầu bằng chữ "L" và nếu người dùng gõ \* s \*, truy vấn trả về tất cả các thành phố mà tên chứa chữ "s".

**Bước 11:** Trên Tab Design tìm tới nhóm Result Click chọn Run.

Dưới đây là kết quả khi bạn gõ vào ô Parameter là \*.



#### 4. Thực hiện tính toán trên cơ sở dữ liệu

Thông thường bạn không sử dụng bảng để lưu trữ các trường tính toán bởi vì những giá trị đó có thể được dựa trên rất nhiều thông tin ở các bảng khác nhau trong cùng một cơ sở dữ liệu. Ví dụ, bảng Order Details trong Database Northwind 2007 không lưu giữ giá trị tổng sản phẩm, bởi nó được tính toán từ các trường Quantity, Unit Price và Discount trong bảng này.

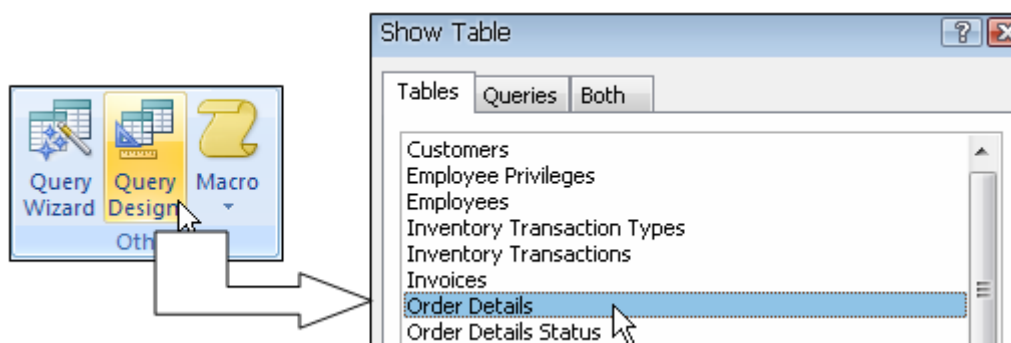
Trong một số trường hợp, các trị tính toán sẽ trở nên không phù hợp, lỗi thời, bởi vì các giá trị tạo nên nó đã thay đổi. Ví dụ, bạn không muốn để lưu lại độ tuổi của ai đó trong một bảng, bởi vì mỗi năm bạn sẽ cần phải cập nhật lại những giá trị đó, thay vào đó, bạn có thể lưu trữ ngày tháng năm sinh của người này, sau đó khi cần sử dụng bạn chỉ cần viết một truy vấn tính ra tuổi của người đó dựa trên dữ liệu ngày tháng năm sinh và ngày hiện tại.

Bạn sử dụng thủ tục sau đây để tính tổng sản phẩm từ bảng Order Details.

**Bước 1:** Mở cơ sở dữ liệu Northwind 2007

**Bước 2:** Trên Tab Create, tìm tới nhóm Other chọn Query Design.

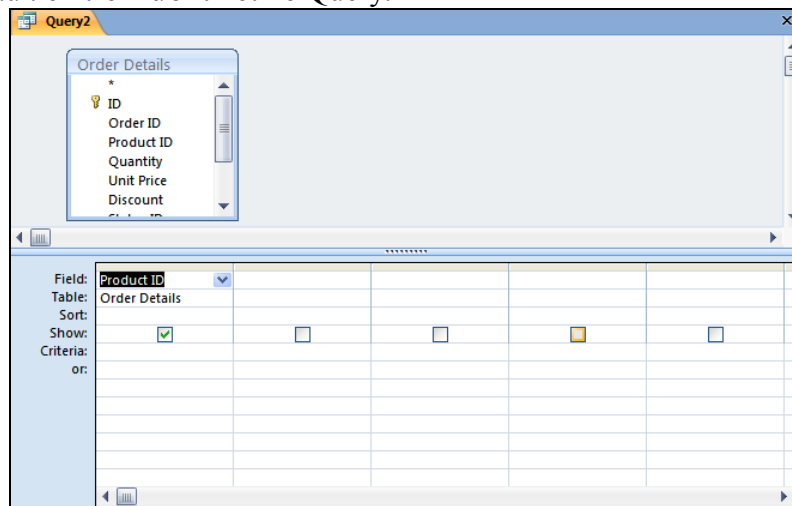
**Bước 3:** Trong hộp thoại Show Table, trong tab Tables, Double Click chọn Order Details.



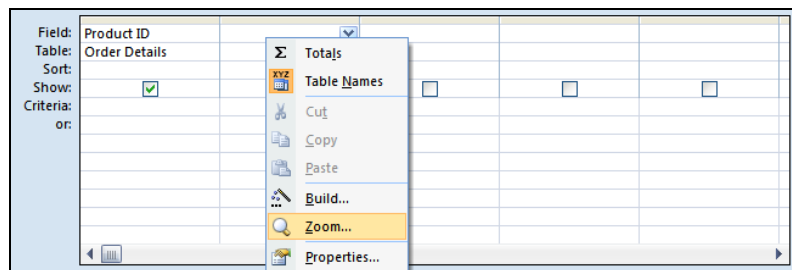
**Bước 4:** Click Close để đóng cửa sổ này



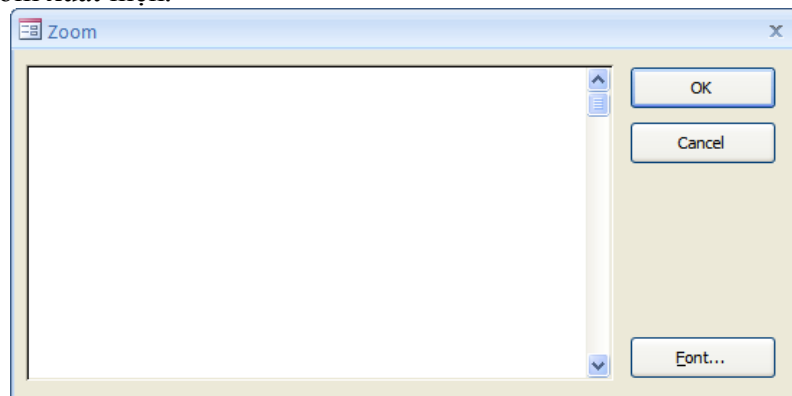
**Bước 5:** Trên bảng Order Details, Double Click chuột vào trường Product ID để thêm trường vào cột đầu tiên trên lưới thiết kế Query.



**Bước 6:** Ở cột thứ 2 trên lưới thiết kế Query, Click phải chuột vào hàng Field rồi chọn Zoom.

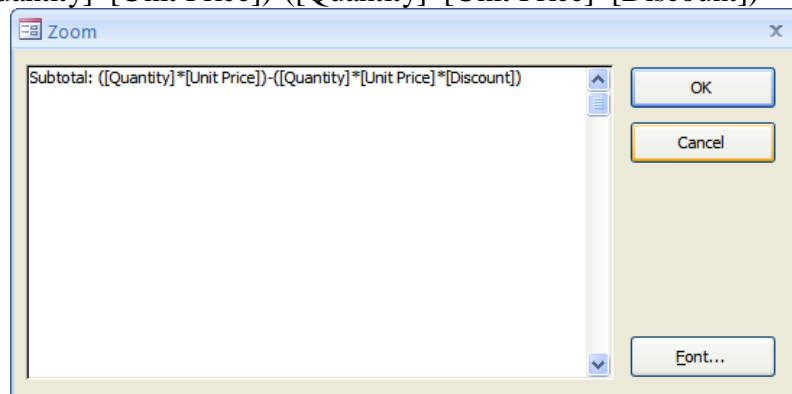


Hộp thoại Zoom xuất hiện.



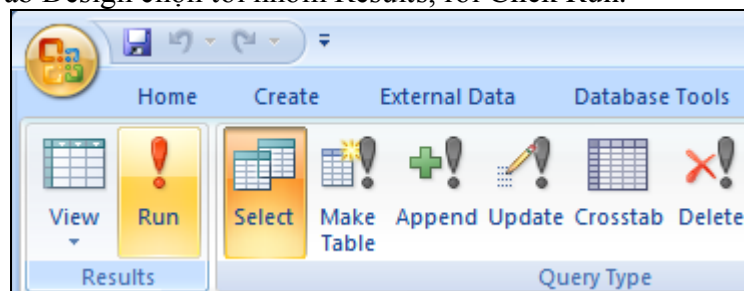
**Bước 7:** Trong hộp thoại Zoom, gõ vào đó nội dung dưới đây.

Subtotal: ([Quantity]\*[Unit Price])-([Quantity]\*[Unit Price]\*[Discount])

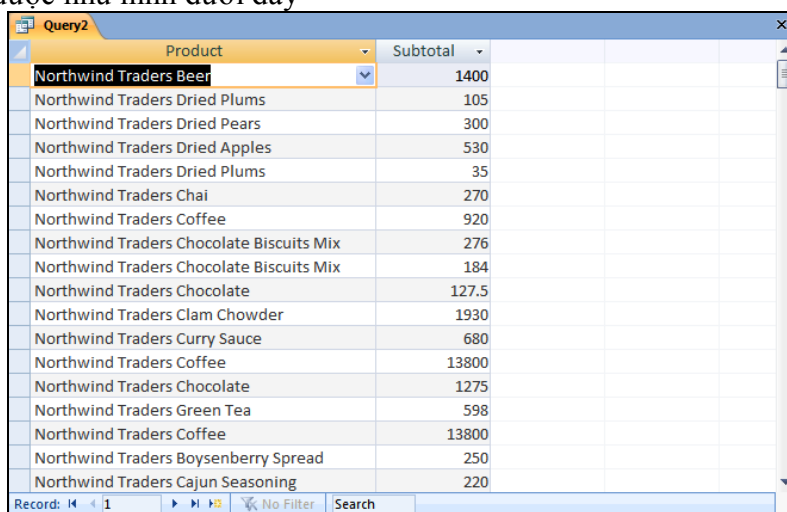


Click Ok để chấp nhận giá trị.

**Bước 8:** Trên Tab Design chọn tới nhóm Results, rồi Click Run.

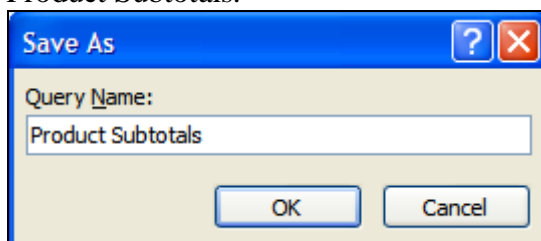


Kết quả sẽ được như hình dưới đây



Product	Subtotal
Northwind Traders Beer	1400
Northwind Traders Dried Plums	105
Northwind Traders Dried Pears	300
Northwind Traders Dried Apples	530
Northwind Traders Dried Plums	35
Northwind Traders Chai	270
Northwind Traders Coffee	920
Northwind Traders Chocolate Biscuits Mix	276
Northwind Traders Chocolate Biscuits Mix	184
Northwind Traders Chocolate	127.5
Northwind Traders Clam Chowder	1930
Northwind Traders Curry Sauce	680
Northwind Traders Coffee	13800
Northwind Traders Chocolate	1275
Northwind Traders Green Tea	598
Northwind Traders Coffee	13800
Northwind Traders Boysenberry Spread	250
Northwind Traders Cajun Seasoning	220

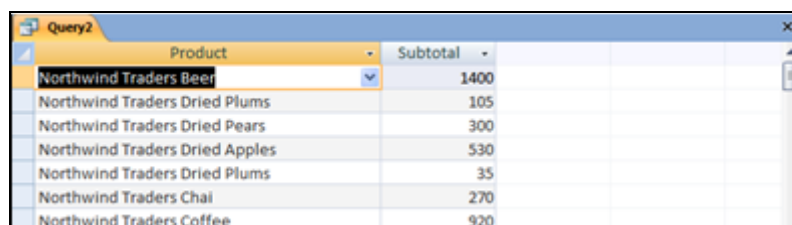
**Bước 9:** Cuối cùng Click Ctrl + S để lưu lại Query, trong hộp thoại nhắc nhập tên bạn nhập vào tên cho Query là: Product Subtotals.



## 5. Tổng hợp hoặc tính tổng dữ liệu - TOTAL

Khi bạn sử dụng bảng để ghi lại các giao dịch hoặc lưu trữ dữ liệu phát sinh, tại một thời điểm nào đó bạn hoàn toàn có thể tổng hợp, thống kê như tính tổng hay trung bình..., Microsoft Office Access 2007 cung cấp một tính năng mới, một dòng tổng cộng, cho phép bạn tổng hợp dữ liệu một cách đơn giản trong bất kỳ Datasheet nào như Form, Query, Report, Hãy thử chức năng này bằng cách thêm một dòng tổng cộng vào Query Product Subtotals mà bạn vừa tạo ở phần trước.

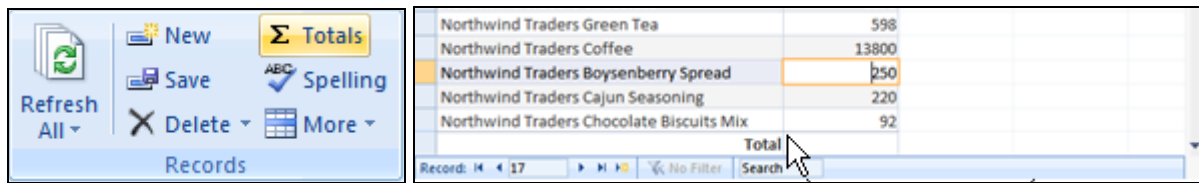
**Bước 1:** Chạy Query Product Subtotals ở chế độ hiển thị Datasheet View.



Product	Subtotal
Northwind Traders Beer	1400
Northwind Traders Dried Plums	105
Northwind Traders Dried Pears	300
Northwind Traders Dried Apples	530
Northwind Traders Dried Plums	35
Northwind Traders Chai	270
Northwind Traders Coffee	920

**Bước 2:** Trên Tab Home, trong nhóm Records Click chọn Totals.

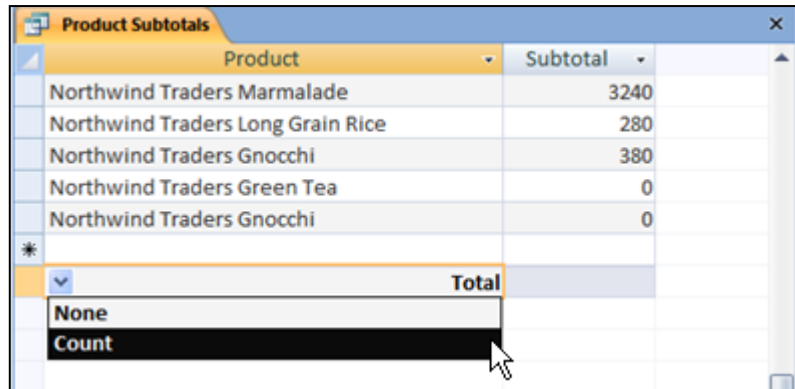
Một dòng mới xuất hiện phía dưới đáy của Datasheet với dòng chữ Total ở cột đầu tiên.



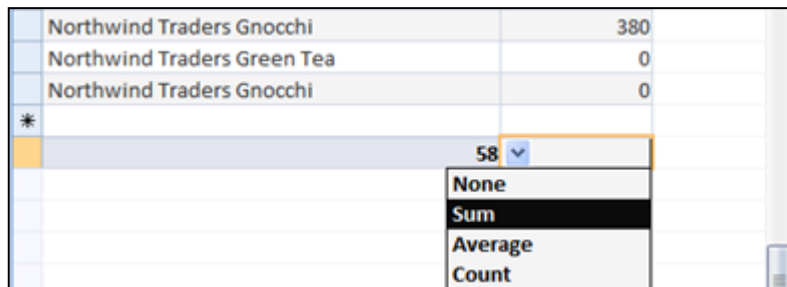
**Bước 3:** Click chọn vào Cell đầu tiên ở dòng cuối cùng của Datasheet (dòng Total).

Vì đây là dữ liệu kiểu Text vì vậy chỉ có hai lựa chọn cho bạn tổng hợp dữ liệu là None và Count, ở đây bạn chọn Count.

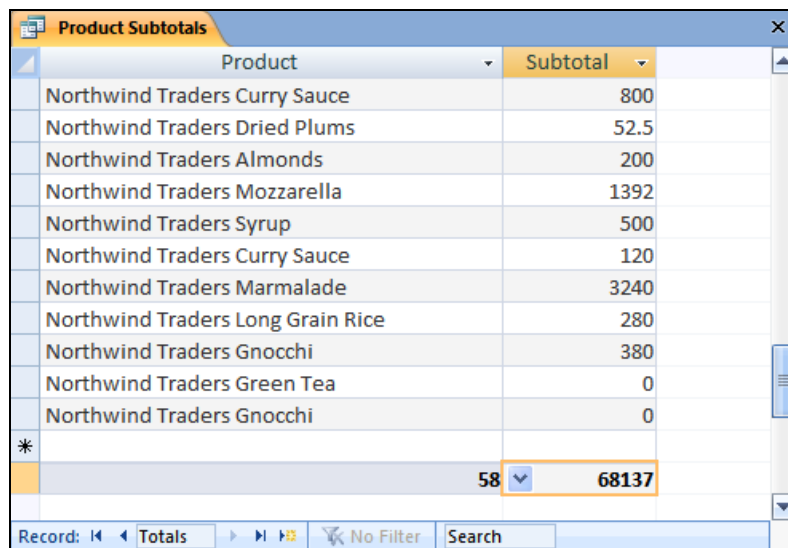
**Bước 4:** Tiếp tục Click chọn ô ở cột thứ 2.



Vì đây là trường số nên bạn có rất nhiều lựa chọn để tổng hợp dữ liệu, trong trường hợp này bạn chọn Sum để tính tổng cột Subtotal.



**Bước 5:** Kết quả cuối cùng Query sẽ hiển thị như hình dưới đây.

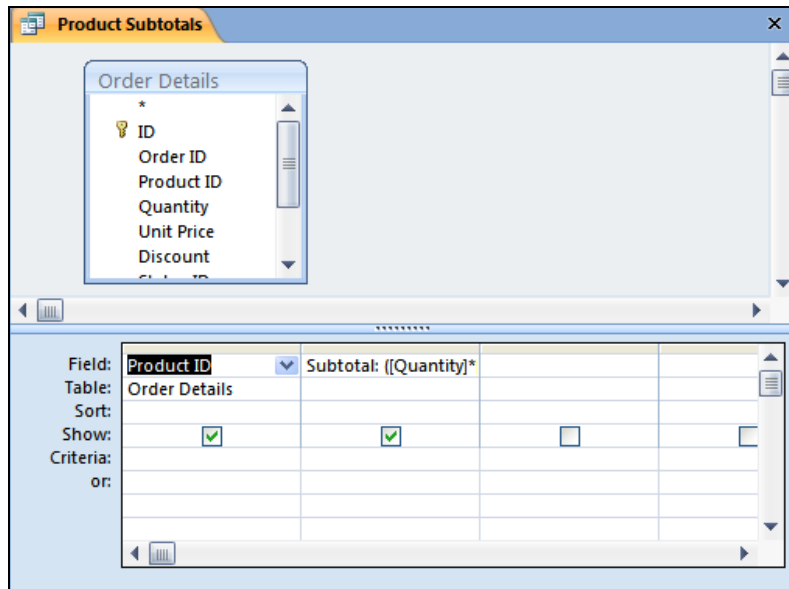


### **Tạo ra một Query tính tổng phức tạp hơn**

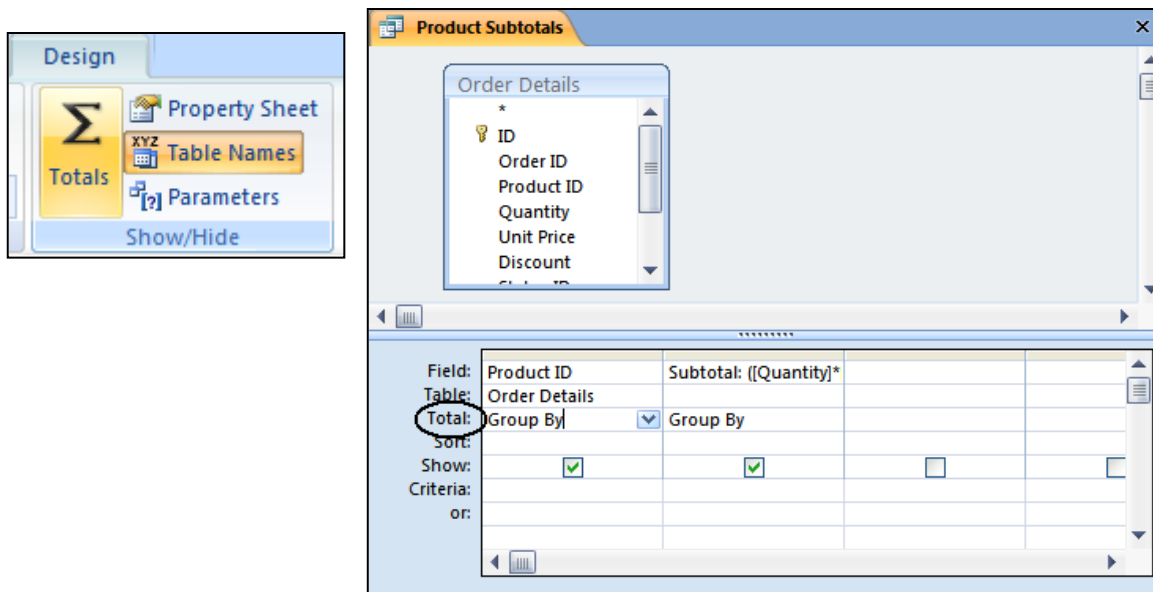
Dòng tổng cộng mà các bạn vừa thực hiện trên Datasheet có thể mạnh riêng của nó truy nhiên đối với những câu hỏi phức tạp hơn bạn nên sử dụng Query tổng cộng. Query tổng cộng bản chất là một Query lựa chọn mà cho phép bạn có thể nhóm hoặc tổng hợp dữ liệu. Ví dụ, giả sử bạn muốn xem tổng số doanh thu trên mỗi sản phẩm. Trong Query tổng cộng bạn có thể sử dụng hàm Sum là một trong những hàm thống kê ( SUM, COUNT, AVG, ...) để xem tổng số doanh thu trên mỗi sản phẩm.

Sử dụng các thủ tục sau đây để sửa query Product Subtotals để nó có tổng hợp theo từng sản phẩm.

**Bước 1:** Chọn tới Query Product Subtotals và mở nó ở chế độ Design View.



**Bước 2:** Từ Tab Design chọn tới nhóm Show/Hide Click chọn Totals.



Một dòng Totals sẽ xuất hiện trên lưới thiết kế Query.

**Chú ý:** Mặc dầu cùng tên nhưng dòng Totals trên lưới thiết kế và dòng Total trên Datasheet không giống nhau.

**Bước 3:** Ở cột thứ 2 trên lưới thiết kế Query, trên dòng Total bạn chọn giá trị Sum trong hộp Combo đổ xuống.

Field:	Product ID	Subtotal: ([Quantity]*	
Table:	Order Details		
Total:	Group By	Sum	
Sort:		Group By	
Show:	<input checked="" type="checkbox"/>	Sum	
Criteria:		Avg	
or:		Min	
		Max	
		Count	
		StDev	
		Var	
		First	
		Last	
		Expression	
		Where	

Product	Subtotal
Northwind Traders Chai	720
Northwind Traders Syrup	500
Northwind Traders Cajun Seasoning	880
Northwind Traders Olive Oil	533.75
Northwind Traders Boysenberry Spread	2500
Northwind Traders Dried Pears	1200
Northwind Traders Curry Sauce	2600
Northwind Traders Fruit Cocktail	1560
Northwind Traders Chocolate Biscuits Mix	782
Northwind Traders Marmalade	3240
Northwind Traders Scones	200
Northwind Traders Beer	6818
Northwind Traders Crab Meat	2208
Northwind Traders Clam Chowder	2798.5
Northwind Traders Coffee	29900
Northwind Traders Chocolate	2550
Northwind Traders Dried Apples	2120
Northwind Traders Long Grain Rice	280
<b>24</b>	<b>68137</b>

Record: 1 of 24 | No Filter | Search

Bước 4: Trên Tab Design tìm tới nhóm Result chọn View để chạy thử Query

Bước 5: Cuối cùng Click Ctrl + S để lưu lại Query

## 6. Tạo một bảng từ nhiều bảng khác. MAKE TABLE

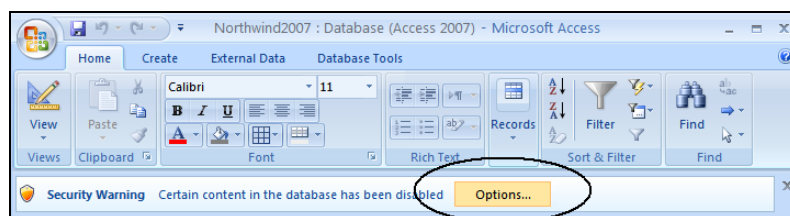
Bạn có thể sử dụng **make-table query** để tạo một bảng mới từ dữ liệu được lưu trữ ở nhiều bảng khác.

Ví dụ, giả sử bạn muốn gửi dữ liệu về đơn hàng cho các đối tác ở Chicago và bạn sử dụng Access để lập báo cáo. Thay vì gửi cho họ tất cả các dữ liệu về đơn đặt hàng, bạn muốn giới hạn những dữ liệu được gửi đi chỉ là những đơn đặt hàng ở Chicago.

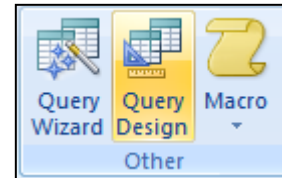
Bạn có thể xây dựng một Query lựa chọn chỉ chứa những đơn hàng ở Chicago, và sau đó sử dụng Query này để tạo một bảng dữ liệu mới. Dưới đây là các bước thực hiện những công việc đó.

Bước 1: Mở cơ sở dữ liệu Northwind 2007

Bước 2: Để chạy một make-table query, bạn cần phải **kích hoạt tính năng enable the database content** nếu trước đó bạn mở nó ở chế độ **Untrusted**. Bạn Click chọn **Options** trên thanh **Message Bar**, xuất hiện phía **dưới Ribbon**.



Hộp thoại Microsoft Office Security Options xuất hiện bạn Click chọn mục Enable this content rồi Click Ok. Cửa sổ Login Dialog xuất hiện, bạn hãy đóng cửa sổ này lại.

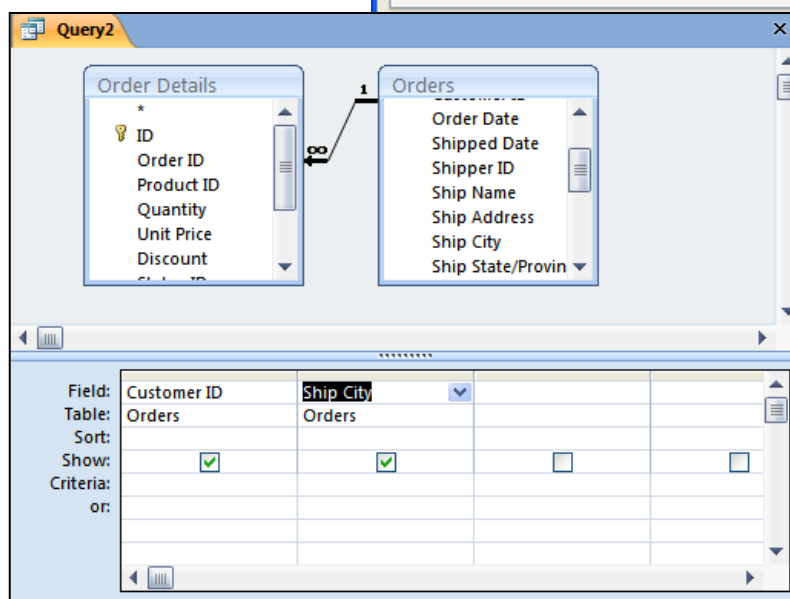
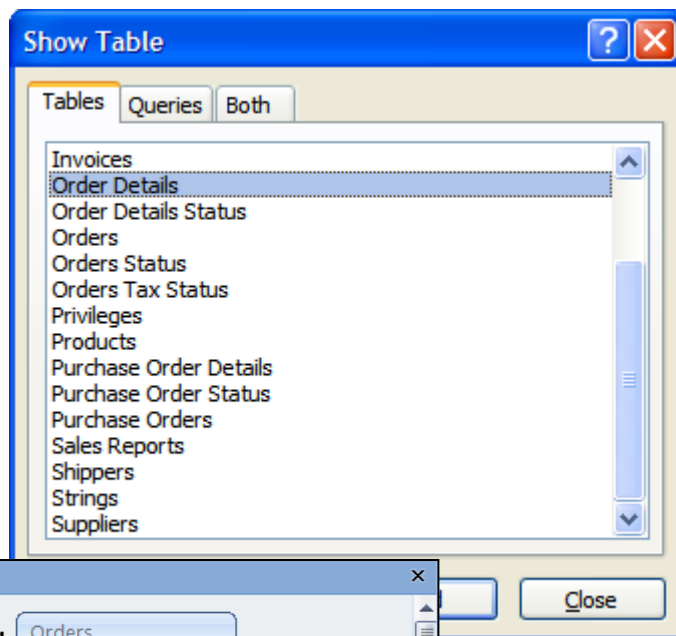


*Bước 3:* Trên Tab Create tìm tới nhóm Other Click chọn Query Design.

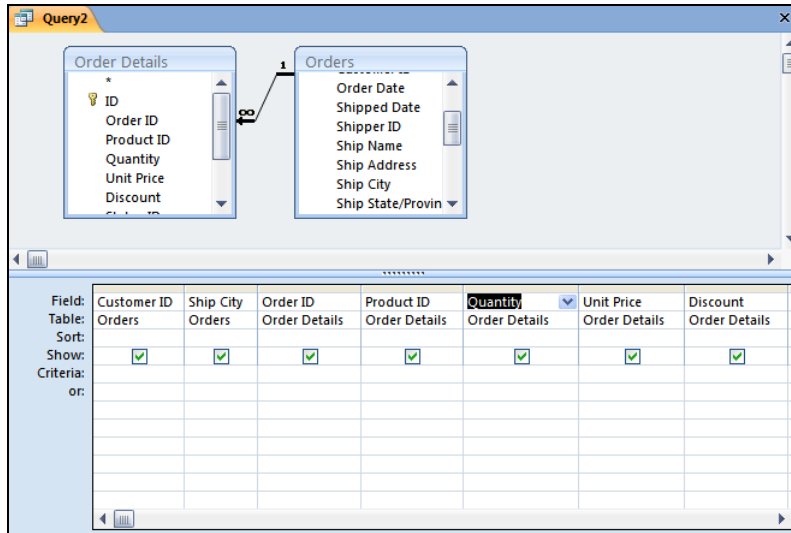
*Bước 4:* Trong hộp thoại Show Table Double Click vào mục Order Details và Orders.

*Bước 5:* Click Close để đóng hộp thoại Show Table.

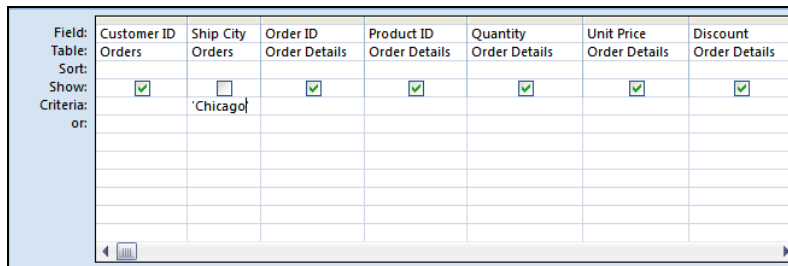
*Bước 6:* Trên bảng Orders bạn Double Click vào lần lượt các trường Customer ID và Ship City để bổ xung những trường này lên lưới thiết kế Query.



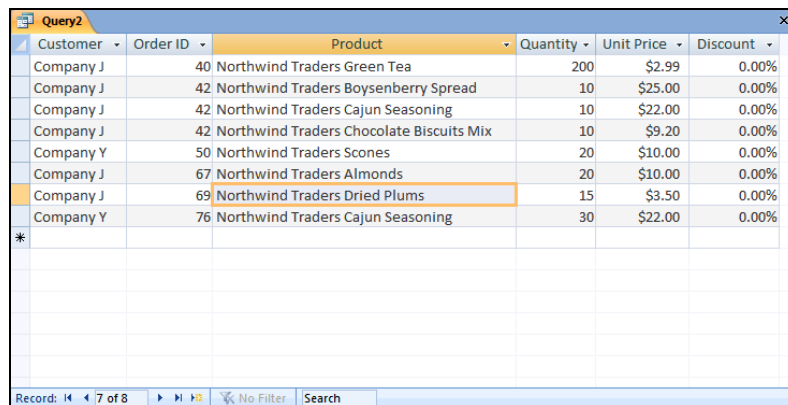
*Bước 7:* Trên bảng Order Details làm tương tự như ở bước 6 với các trường Order ID, Product ID, Quantity, Unit Price và Discount.



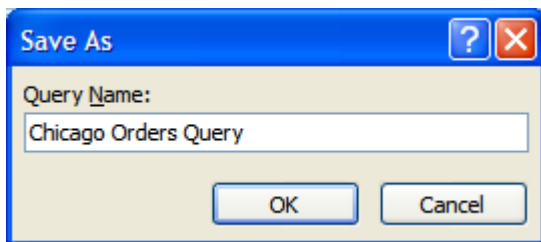
**Bước 8:** Ở cột Ship City trên lưới thiết kế Query, xóa bỏ nút Check ở dòng Show. Trên dòng Criteria bạn gõ vào giá trị 'Chicago'. Chạy thử Query để biết kết quả truy vấn trước khi chúng ta tạo bảng



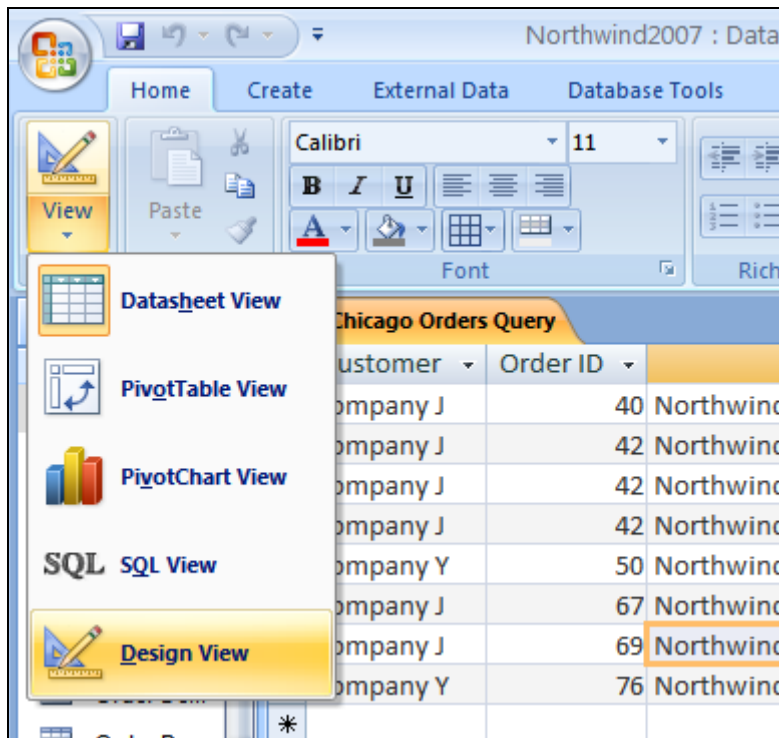
**Bước 9:** Trên Tab Design tìm tới nhóm Results Click chọn View, kết quả Query trả về như hình dưới đây:



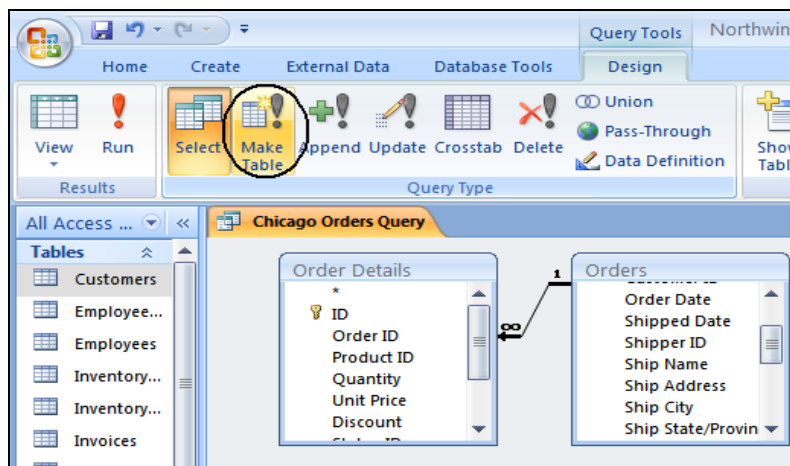
**Bước 10:** Lưu Query lại với tên Chicago Orders Query.



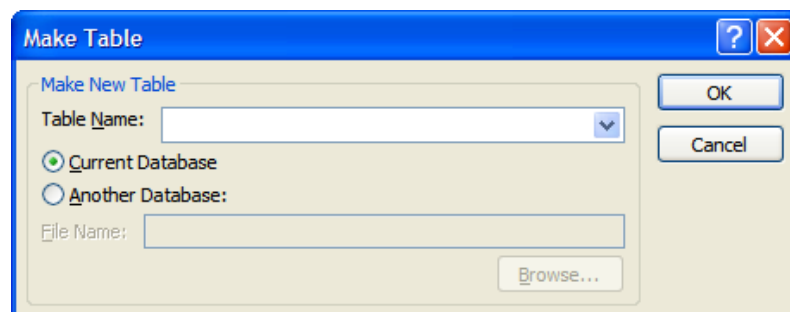
Bước 11: Trên Tab Home tìm tới nhóm Views Click chọn View trong hộp Combo để xuống bạn chọn Design View.



Bước 12: Trên Tab Design tìm tới nhóm Query Type bạn chọn Make Table.

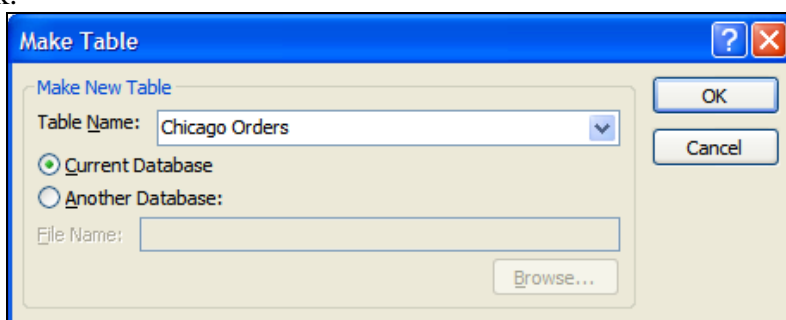


Hộp thoại Make Table xuất hiện như hình dưới đây:



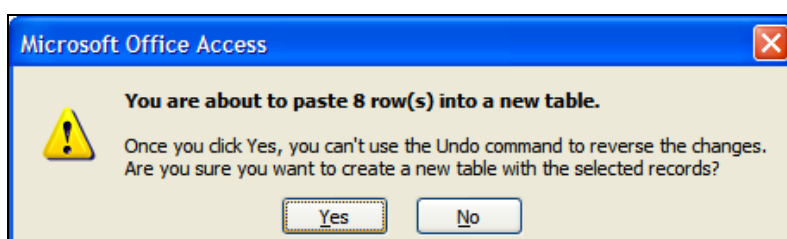


**Bước 13:** Trong hộp thoại này bạn tìm tới ô Table Name gõ vào đó để đặt tên cho bảng là Chicago Orders và Click Ok.

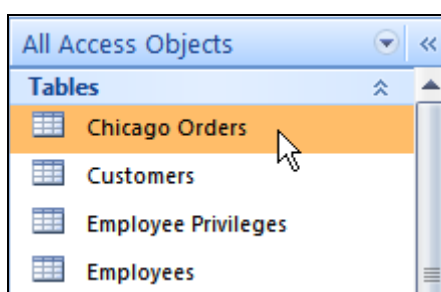


**Bước 14:** Trên Tab Design tìm tới nhóm Results chọn Run.

**Bước 15:** Một hộp thoại xác nhận xuất hiện bạn chọn Yes.



Một bảng mới sẽ được tạo trong Pane Navigation.



## 7. Thêm dữ liệu vào bảng từ nhiều bảng khác. APPEND

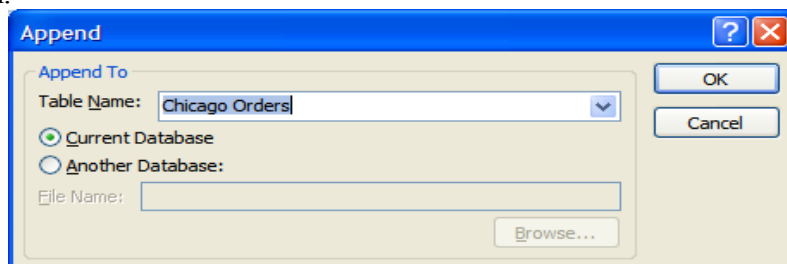
Bạn có thể sử dụng một Append Query để lấy dữ liệu từ một hoặc nhiều bảng vào một bảng khác.

Giả sử bạn tạo ra một bảng và chia sẻ nó với những khách hàng ở Chicago cùng trong hiệp hội, nhưng bạn nhận ra rằng hiệp hội cũng làm việc với những khách hàng ở khu vực Milwaukee. Bạn muốn thêm những dòng dữ liệu chứa những giao dịch với khu vực này trước khi chia sẻ cho hiệp hội.

Bạn có thể thêm dữ liệu về khu vực Milwaukee vào bảng Chicago Orders mà bạn đã tạo ra ở phần trước theo các bước thực hiện dưới đây:

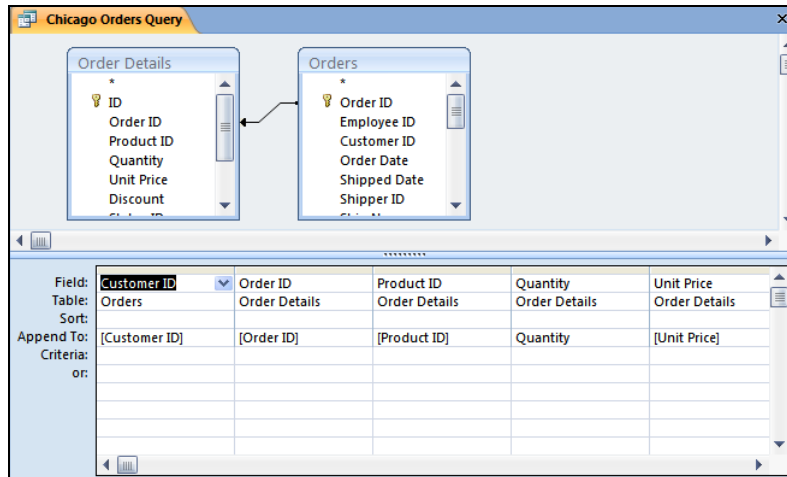
**Bước 1:** Mở Query có tên Chicago orders Query ở chế độ Design.

**Bước 2:** Trên Tab Design tìm tới nhóm Query Type Click chọn Append, hộp thoại Append xuất hiện.

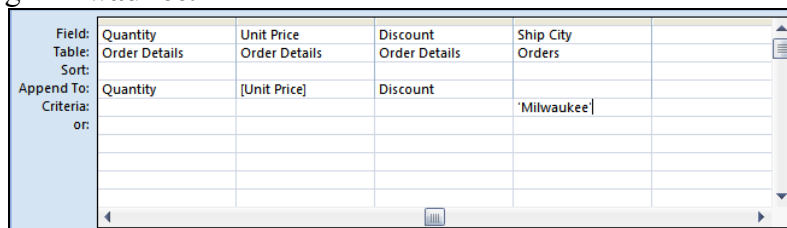


Trong Combo Table Name bạn chọn Chicago Orders sau đó Click Ok.

Khi hộp thoại Append đóng lại trên lưới Design xuất hiện thêm dòng Append và dòng Show biến mất.



**Bước 3:** Trên lưới thiết kế ở dòng Criteria của cột Ship City, bạn xóa giá trị 'Chicago' và thay vào đó bằng 'Milwaukee'.



**Bước 4:** Trên dòng Append To chọn trường thích hợp cho từng cột. Trong ví dụ này các trường đều có tên giống như các cột chỉ trừ trường điều kiện Ship City.

**Bước 5:** Trên Tab Design tìm tới nhóm Result chọn Run để chạy thử Query.

## 8. Thay đổi dữ liệu tự động - UPDATE

Bạn có thể sử dụng một Query Update để thay đổi dữ liệu trong các bảng của mình, và bạn có thể sử dụng các yêu cầu tìm kiếm với Query này để chọn ra những dòng đặc biệt cần được thay đổi. Query Update cung cấp cho bạn một cơ hội để xem xét các dữ liệu cần thiết trước khi thực hiện việc cập nhật.

**Chú ý:** Một hành động của Query khi thực hiện sẽ không thể hoàn tác vì vậy bạn cần phải xem xét để Back up dữ liệu của bất cứ bảng nào mà bạn sẽ cập nhật lại bằng cách sử dụng Query Update.

Trong ví dụ trước, Bạn đã thêm các dòng mới vào bảng Chicago Orders. Trong bảng này, trường Product ID hiển thị số Id của sản phẩm. Để làm cho dữ liệu dễ sử dụng hơn trên báo cáo, bạn có thể sẽ muốn thay thế các ID sản phẩm bằng tên của chúng. Để có thể thay thế các Id sản phẩm, đầu tiên bạn phải đổi kiểu dữ liệu của trường này từ kiểu số sang kiểu Text, để trường Product Id có thể chấp nhận được dữ liệu của trường Product Name.

Dưới đây là các bước thực hiện:

**Bước 1:** Mở bảng Chicago Orsers ở chế độ hiển thị Design.

**Bước 2:** Trên hàng Product ID đổi kiểu dữ liệu từ Number thành kiểu Text.

**Bước 3:** Lưu và đóng bảng Chocago Orders lại

**Bước 4:** Trên Tab Create tìm tới nhóm Other, chọn Query Design.

**Bước 5:** Trong hộp thoại Show Table, Double Click chuột vào 2 bảng Chicago Orders và Products.

**Bước 6:** Click Close để đóng hộp thoại Show Table.

Bước 7: Trên Tab Design tìm tới nhóm Query Type, Click chọn Update. Trên lưới thiết kế, 2 dòng Sort và Show biến mất và thay vào đó là dòng Update To.

Bước 8: Trong bảng Chicago Orders, Double Click chuột vào trường Product Id để bổ xung trường này lên lưới thiết kế Query.

Bước 9: Trên lưới thiết kế, tại dòng Update To của cột Product ID bạn nhập vào nội dung dưới đây:

[Products].[Product Name]

Bước 10: Trên dòng Criteria bạn nhập vào nội dung dưới đây:

[Product ID] Like ([Products].[ID])

Từ khóa Like được sử dụng trong trường hợp này là rất cần thiết bởi vì những trường mà bạn so sánh nội dung khác nhau về kiểu dữ liệu (trường Product ID là kiểu Text bạn vừa đổi, và trường ID là kiểu Number).

Bước 11: Bạn có thể xem xét các giá trị sẽ bị thay đổi khi thực hiện Query Update bằng cách hiển thị Query ở chế độ Datasheet.

Trên Tab Design, tìm tới nhóm Results Click chọn View, trong danh sách đổ xuống bạn chọn Datasheet View.

Kết quả trả về từ Query là một danh sách các Product Id mà bạn sẽ Update.

Bước 12: Trên Tab Design tìm tới nhóm Results Click chọn Run.

Khi bạn mở bảng Chicago Orders, bạn sẽ nhìn thấy các giá trị số trong trường Product ID được thay thế bởi tên các sản phẩm trong bảng Products.

## **9. Xóa dữ liệu tự động - DELETE**

Bạn có thể sử dụng một Query Delete để xóa dữ liệu trong các bảng, và bạn cũng có thể sử dụng một Query Delete với các điều kiện tìm kiếm để có thể xóa những bản ghi đặc biệt. Một Query Delete cung cấp cho bạn cơ hội để xem xét các bản ghi sẽ được xóa bỏ trước khi bạn thực hiện việc xóa dữ liệu.

Giả sử bạn đang chuẩn bị dữ liệu trong bảng Chicago Orders để gửi tới hiệp hội khách hàng ở Chicago, và bạn nhận thấy rằng một số hàng chứa các giá trị rỗng. Bạn muốn loại bỏ các hàng này trước khi gửi bảng dữ liệu đi. Rất đơn giản, bạn có thể mở bảng và tìm tới những hàng đó để xóa một cách thủ công, nhưng bạn sẽ gặp vấn đề lớn khi số lượng hàng cần xóa nhiều lên và nằm rải rác ở các vị trí khác nhau. Khi đó bạn sẽ thấy được tác dụng của Query Delete.

Dưới đây là các bước tạo một Query để xóa các dòng trong bảng Chicago Orders với điều kiện trường Order ID không có giá trị, và dưới đây là các bước thực hiện.

Bước 1: Trên Tab Create tìm tới nhóm Other chọn Query Design.

Bước 2: Trong hộp thoại Show Table, Double Click chuột chọn Chicago Orders.

Bước 3: Click Close để đóng hộp thoại Show Table.

Bước 4: Trên Tab Design, tìm tới nhóm Query Type, Click chọn Delete.

Trên lưới thiết kế, hàng Sort và Show biến mất thay vào đó là hàng Delete.

Bước 5: Trên bảng Chicago Orders Double Click chuột vào trường Order Id để thêm nó vào lưới thiết kế.

Bước 6: Trên lưới thiết kế, ở dòng Criteria của cột Order ID nhập vào đó giá trị Is Null.

Bước 7: Trên Tab Design tìm tới nhóm Result Click chọn Run để chạy thử Query.

## Bài 4. Làm việc với Form

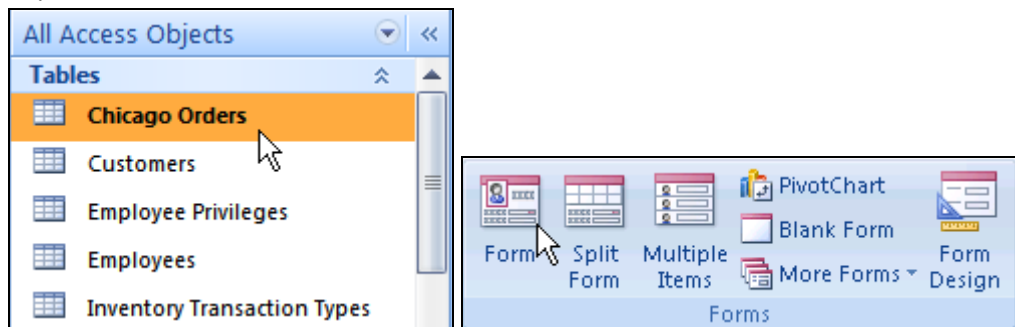
### 1. Tạo Form

#### Tạo Form đơn giản bằng Form

Bạn có thể sử dụng công cụ biểu mẫu (Form) để tạo ra một biểu mẫu chỉ với một lần Click chuột. Khi bạn sử dụng công cụ này, tất cả các trường từ các nguồn dữ liệu sẽ được đặt lên biểu mẫu. Bạn có thể bắt đầu sử dụng biểu mẫu ngay lập tức hoặc bạn cũng có thể chỉnh sửa nó ở chế độ Layout View hoặc Design View để nó phù hợp hơn với nhu cầu sử dụng của mình.

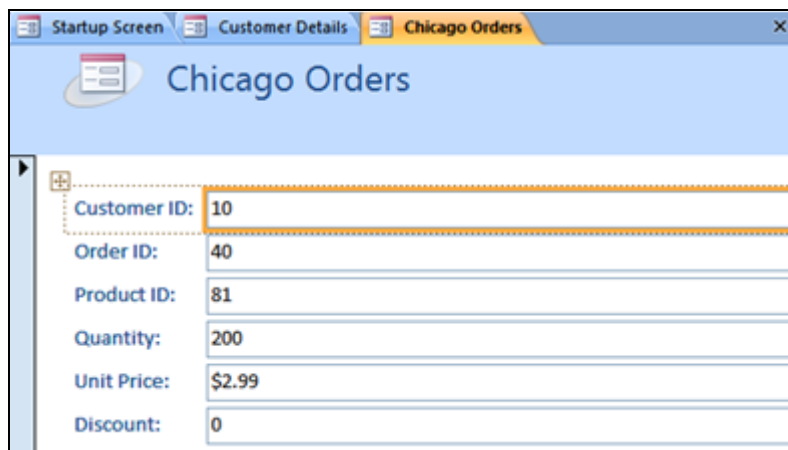
Các bước tạo Form bằng cách sử dụng công cụ Form.

**Bước 1:** Trên Pane Navigation, Click chọn Table hoặc Query chứa dữ liệu mà bạn muốn hiển thị lên Form.



**Bước 2:** Từ Tab Create tìm tới nhóm Forms, Click chọn Form.

Access sẽ tạo ra một Form và hiển thị nó ở chế độ Layout View. Trong Layout View, bạn có thể thực hiện các thay đổi về mặt hình thức ngay cả khi Form đang hiển thị dữ liệu. Ví dụ, bạn có thể điều chỉnh kích thước của các TextBox để chúng phù hợp với những dữ liệu đang hiển thị.



Nếu Access tìm thấy một bảng mà có mối quan hệ một nhiều với một bảng hoặc một Query khác cùng sử dụng để tạo nên Form, Access sẽ thêm một Datasheet nữa vào Form dựa trên mối quan hệ của bảng với một bảng hoặc một Query khác. Ví dụ, nếu bạn tạo một biểu mẫu đơn giản dựa trên các bảng Employees, và có một mối quan hệ một nhiều được xác định giữa bảng Employees và bảng Orders, một Datasheet hiển thị tất cả các bản ghi của bảng Orders mà liên quan đến bản ghi hiện thời của bảng Employee. Bạn có thể xóa các Datasheet trên Form nếu bạn quyết định không cần đến nó. Trong trường hợp có nhiều hơn một bảng có mối quan hệ một nhiều với bảng mà bạn sử dụng để tạo Form, Access cũng sẽ không thêm các Datasheet khác vào biểu mẫu.

## Tạo Split Form bằng công cụ Split Form.

Split Form là một chức năng mới trong Microsoft Office Access 2007 nó cung cấp cho bạn hai chế độ hiển thị dữ liệu cùng một lúc – Form View và Datasheet View.

Hai chế độ hiển thị được kết nối tới cùng một nguồn dữ liệu và đồng bộ với nhau tại mọi thời điểm. Chọn một trường trong phần này cũng chính là chọn trường đó trong phần còn lại của biểu mẫu. Bạn có thể thêm, chỉnh sửa, hoặc xóa dữ liệu từ một trong hai phần hiển thị.

Làm việc với các Split Form sẽ cung cấp cho bạn những lợi ích của cả hai kiểu hiển thị dữ liệu nhưng lại được bố trí trên một biểu mẫu duy nhất. Ví dụ, bạn có thể sử dụng phần Datasheet của biểu mẫu để tìm kiếm các bản ghi và sử dụng phần còn lại để hiển thị chi tiết và chỉnh sửa dữ liệu.

Các bước tạo một Split Form bằng cách sử dụng công cụ Split Form.

ID	Company	First Name	Last Name	E-mail Address
1	Company A	Anna	Bedecs	
2	Company B	Antonio	Gratacos Solso	
3	Company C	Thomas	Axen	
4	Company D	Christina	Lee	
5	Company E	Martin	O'Donnell	
6	Company F	Francisco	Pérez-Olaeta	
7	Company G	Ming-Yang	Xie	
8	Company H	Elizabeth	Andersen	

**Bước 1:** Trên Pane Navigation, Click chọn Table hoặc Query chứa dữ liệu mà bạn muốn tạo Form, tiếp theo bạn mở bảng hoặc Query đó ở chế độ Datasheet View.

**Bước 2:** Trên Tab Create tìm tới nhóm Form chọn Split Form.

Access sẽ tạo ra một Form mới và hiển thị trên Layout View. Trong Layout View, bạn có thể thay đổi các thiết kế cho Form ngay khi nó đang hiển thị dữ liệu.

## Tạo Form sử dụng công cụ Multiple Items.

Khi bạn tạo một Form bằng cách sử dụng công cụ Simple Form, tại một thời điểm Form này chỉ hiển thị được duy nhất một bản ghi. Nếu bạn muốn có một Form có thể hiển thị nhiều bản ghi và có thể tùy chỉnh tiện dụng hơn ở trên Datasheet một lúc bạn có thể sử dụng công cụ Multiple Items.

Dưới đây là các bước thực hiện

**Bước 1:** Trên Pane Navigation Click chọn Table hoặc Query chứa dữ liệu mà bạn muốn hiển thị lên Form.

**Bước 2:** Trên Tab Create tìm tới nhóm Forms Click chọn Multiple Items .

Access sẽ tạo ra một Form và hiển thị nó ở chế độ Layout View. Ở chế độ này bạn có thể thay đổi các thiết kế cho Form ngay khi nó đang hiển thị dữ liệu. Ví dụ như bạn có thể thay đổi độ rộng các TextBox để chứa đủ dữ liệu, ...

Customer ID	Order ID	Product ID	Quantity	Unit Price
10	40	81	200	\$2.99
10	42	6	10	\$25.00
10	42	4	10	\$22.00
10	42	19	10	\$9.20
25	50	21	20	\$10.00
10	67	74	20	\$10.00
10	69	80	15	\$3.50
25	76	4	30	\$22.00

Khi bạn sử dụng công cụ Multiple Items, Form được tạo ra sẽ tương tự như khi chúng ta hiển thị bảng ở chế độ Datasheet. Dữ liệu trên Form được sắp xếp theo các hàng và cột và tại một thời điểm cho phép bạn thấy nhiều hơn một bản ghi. Tuy nhiên, một Form Multiple Items Form cho phép bạn có thể tùy chỉnh nhiều hơn khi hiển thị dữ liệu trong bảng ở chế độ Datasheet, chẳng hạn như khả năng thêm các yếu tố đồ họa vào Form, các nút Button hay nhiều các điều khiển khác.

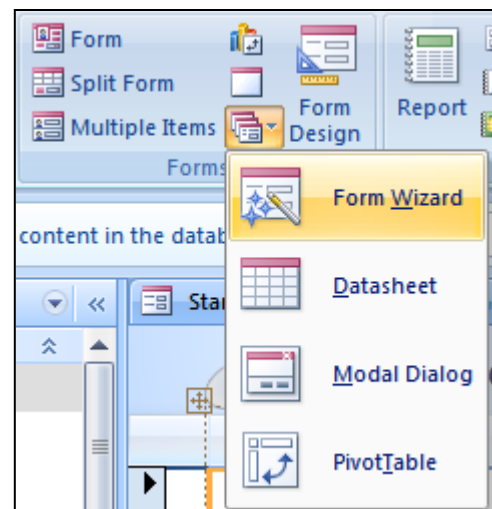
### Tạo Form sử dụng Form Wizard.

Để có thêm lựa chọn về những trường gì xuất hiện trên Form, bạn có thể sử dụng công cụ Form Wizard thay vì các công cụ mà chúng ta đã được tìm hiểu ở các phần trước. Bạn cũng có thể xác định xem dữ liệu như thế nào thì được nhóm, được sắp xếp, và bạn cũng thể sử dụng các trường từ nhiều hơn một bảng hoặc một Query.

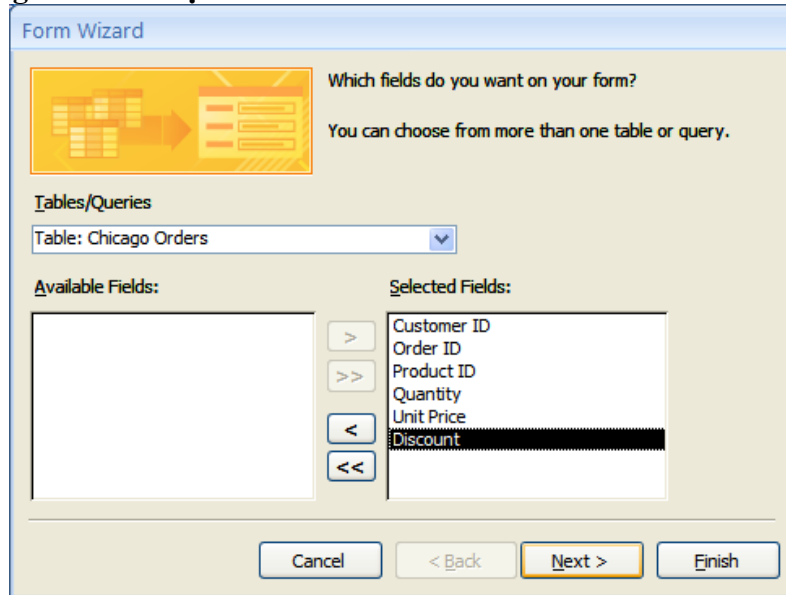
Các bước tạo Form với công cụ Form Wizard.

**Bước 1:** Trên Tab Create, tìm tới nhóm Forms, Click chọn More Form rồi chọn Form Wizards.

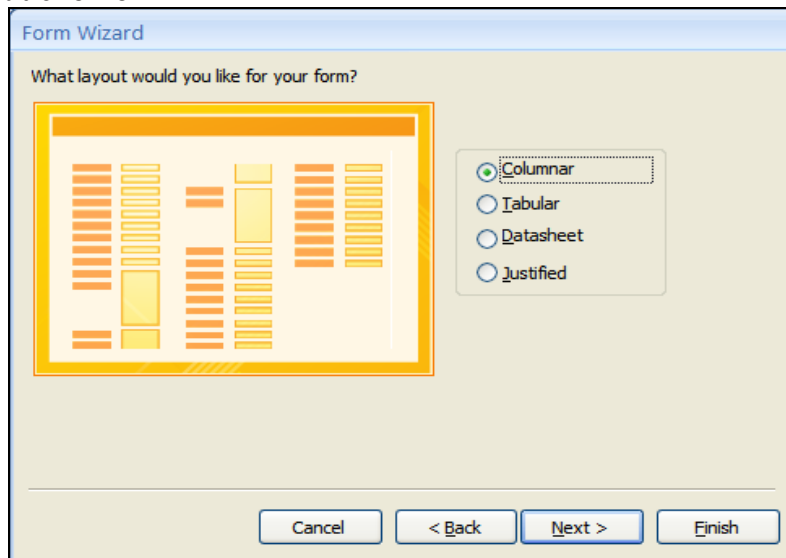
**Bước 2:** Theo các bước hướng dẫn trên các hộp thoại Form Wizards.



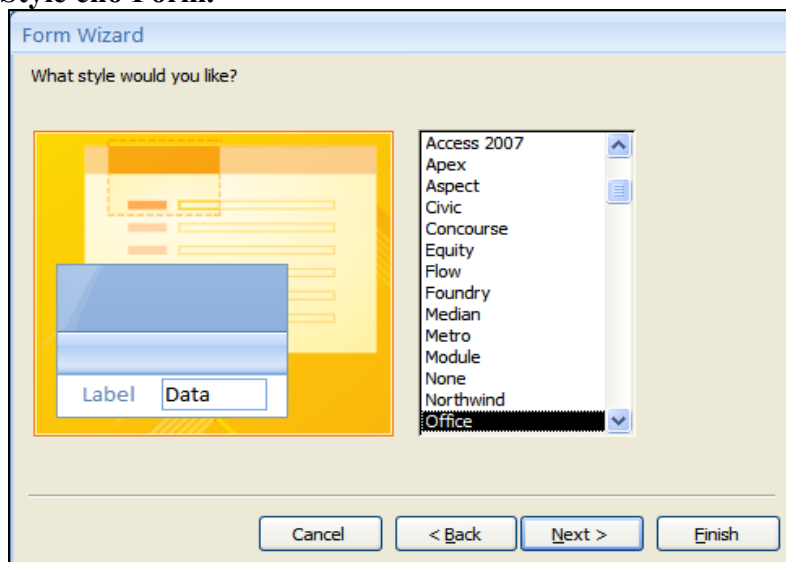
### Chọn trường cần hiển thị lên Form.



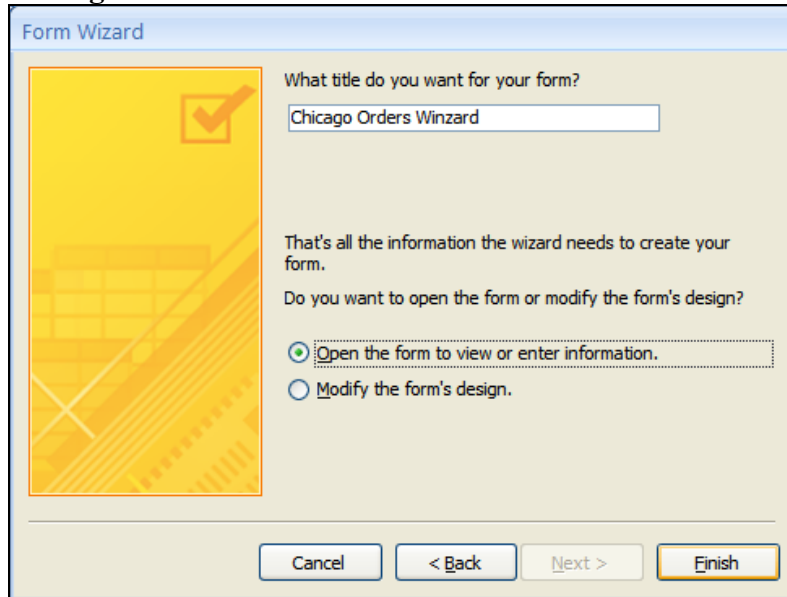
### Chọn Layout cho Form



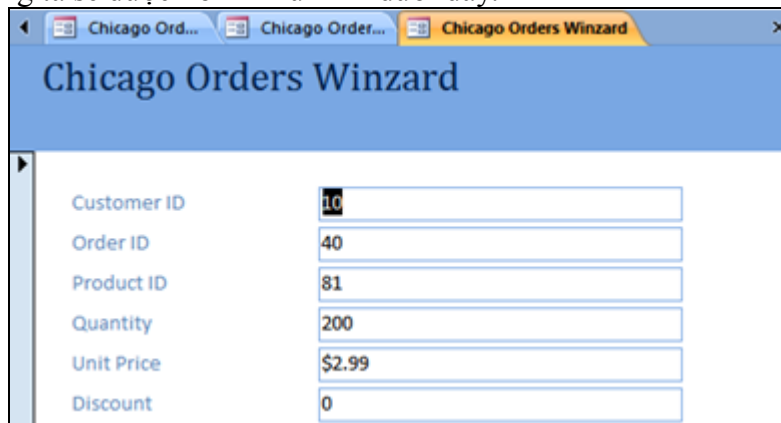
### Chọn kiểu Style cho Form.



**Bước 3: Cuối cùng Click Finish để hoàn tất.**




Kết quả chúng ta sẽ được Form như hình dưới đây.



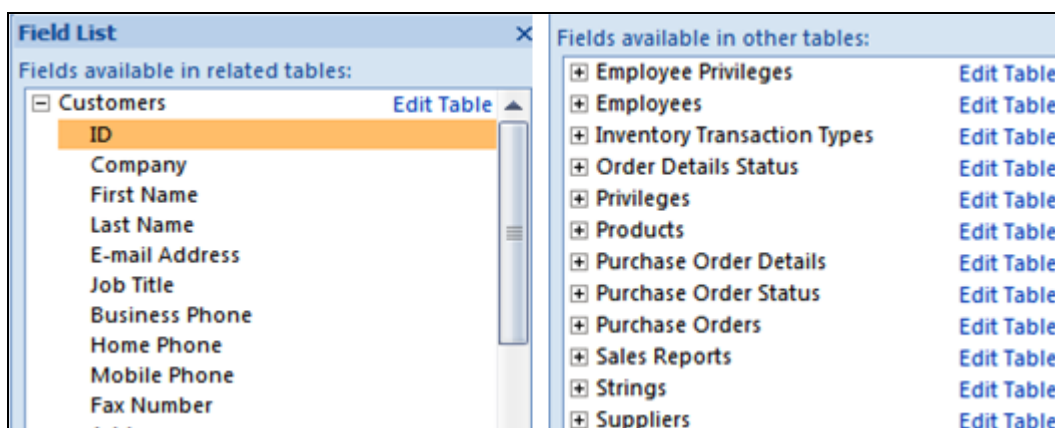
**Tạo Form sử dụng công cụ Blank Form**

Nếu Winzard hoặc một công cụ tạo Form nào trước đây mà chưa đáp ứng được yêu cầu của bạn, bạn có thể sử dụng chức năng Blank Form. Công cụ này là cách nhanh nhất để xây dựng một Form, đặc biệt nếu bạn dự định chỉ hiển thị một vài trường lên Form.

Các bước tạo Form với công cụ Blank Form

**Bước 1:** Trên Tab Create tìm tới nhóm Forms Click chọn Blank Form 

Access sẽ mở một Form trắng ở chế độ Design View, và hiển thị Pane Field List.

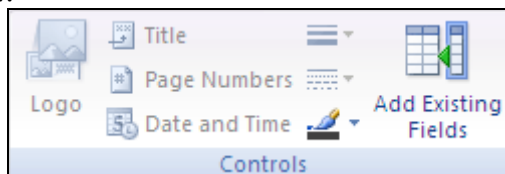





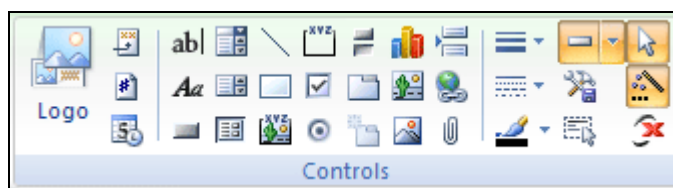
**Bước 2:** Trên Pane Field List, Click chọn vào nút có hình (+) để xem chi tiết các trường trong bảng hoặc Query mà bạn muốn hiển thị lên Form.

**Bước 3:** Để thêm một trường lên Form bạn Click chọn trường rồi kéo thả lên Form. Để thêm nhiều trường cùng một lúc bạn giữ phím Ctrl sau đó check chọn những trường cần thêm rồi kéo thả cùng một lúc lên Form.

**Bước 4:** Sử dụng các công cụ trong nhóm Controls trên Tab Formatting để thêm Logo, tiêu đề, số trang, hoặc ngày tháng, ....



**Bước 5:** Nếu bạn muốn thêm các Control phức tạp hơn lên Form bạn phải chuyển sang chế độ Design View, muốn chuyển về chế độ này bạn Click phải chuột vào Form rồi chọn Design View  sau đó sử dụng các công cụ trong nhóm Controls ở trên Tab Design để thêm chúng lên Form.



## 2. Tìm hiểu về các Controls.

Controls là các đối tượng dùng để hiển thị dữ liệu, thực hiện các sự kiện, và cho phép bạn xem và làm việc với các thông tin đó trên giao diện người dùng, ví dụ như các hình ảnh hay các nhãn thông báo. Control được sử dụng nhiều nhất là TextBox các điều khiển khác có thể kể đến như Label, Check Box, ....

Controls có thể là Control ràng buộc, không ràng buộc hoặc Control tính toán.

- Điều khiển ràng buộc: Một điều khiển có các nguồn dữ liệu là một trường trong một bảng hoặc Query được gọi là Control ràng buộc. Bạn sử dụng các Control (điều khiển) ràng buộc để hiển thị các giá trị của trường trong cơ sở dữ liệu. Những giá trị đó có thể dạng văn bản, dạng số, ngày tháng, hình ảnh, kiểu logic, ... Ví dụ, một TextBox trên Form hiển thị tên của một nhân viên, điều khiển này lấy dữ liệu từ trường Last Name của bảng Employees.

- Điều khiển không ràng buộc: Một Control mà không được gắn với một nguồn dữ liệu (chẳng hạn như một trường hay biểu thức) được gọi là một Control không kiểm soát. Bạn sử dụng các điều khiển này để hiển thị thông tin, các đường Line, hình chữ nhật, hay các hình ảnh. Ví dụ, một Label hiển thị tiêu đề của một Form được gọi là một Control không kiểm soát.

- Điều khiển tính toán: Một điều khiển có nguồn dữ liệu là một biểu thức chứ không phải là một trường, được gọi là điều khiển tính toán. Bạn chỉ rõ giá trị mà bạn muốn sử dụng như là nguồn của dữ liệu trên điều khiển bằng cách định nghĩa một biểu thức. Ví dụ sau đây tính giá của một danh mục sau khi đã trừ đi 25% giảm giá, giá trị còn lại chỉ là 0.75.

$$=[\text{Unit Price}] * 0.75$$

Một biểu thức có thể sử dụng dữ liệu từ một trường trong bảng hoặc Query, hoặc dữ liệu từ các điều khiển khác trên Form.

Khi bạn tạo ra một Form, có lẽ nó hiệu quả nhất để thêm và sắp xếp tất cả các điều khiển ràng buộc đầu tiên, đặc biệt là nếu chúng chiếm phần lớn các điều khiển về form. Sau đó bạn có thể thêm vào điều khiển không ràng buộc và điều khiển tính toán rằng việc thiết kế hoàn toàn bằng cách sử dụng các công cụ trong các nhóm Controls trên tab Design trong Design view.

Bạn tạo ràng buộc để điều khiển một trường xác định các trường từ đó kiểm soát được các dữ liệu. Bạn có thể tạo một điều khiển ràng buộc với trường được lựa chọn bằng cách kéo các kéo từ ô Field List vào form. Ô Field List hiển thị những trường của các bảng hoặc truy vấn. Để hiển thị ô Field List, nhấn ALT + F8, hoặc trên tab Design, trong nhóm Tools, Click chọn Add Existing Field. Khi bạn Double Click chuột vào một trường trong pane Field List, Access sẽ thêm một điều khiển tương ứng với trường đó trên Form.

Ngoài ra, bạn có thể ràng buộc một trường vào một điều khiển bằng cách gõ trực tiếp tên trường vào điều khiển đó, hoặc thiết lập giá trị cho thuộc tính Control Source của điều khiển đó là tên trường của trường dữ liệu. Các thuộc tính xác định các đặc tính của Control như là tên, nguồn dữ liệu, hoặc là định dạng, để mở hộp thoại Property sheet hãy nhấn phím F4.

Sử dụng Pane Field List là cách tốt nhất để tạo một Control ràng buộc vì hai lý do sau đây:

- Access sẽ tự động đính kèm một điều khiển Label có gắn sẵn tên của trường đó, vì vậy bạn không phải gõ các nhãn cho những Control của mình.
- Access sẽ tự động thiết lập rất nhiều thuộc tính phù hợp với giá trị của trường mà bạn đang làm việc (ví dụ như Format, Decimal Places, Input mask)

Nếu bạn đã tạo ra một điều khiển không ràng buộc và bạn lại muốn ràng buộc nó vào một trường nào đó, bạn hãy thiết lập thuộc tính Control Source cho điều kiện này giá trị là tên trường cần ràng buộc giá trị.

### 3. Tìm hiểu về Control Layout.

Điều khiển Layouts cho phép bạn căn chỉnh các điều kiện trên Form theo cả chiều ngang và chiều dọc. Để chúng xuất hiện trên Form theo một định dạng thống nhất. Bạn có thể xem Control Layout như một bảng, ở đó chứa các ô mà nội dung trong mỗi ô đó là các Control. Sau đây là cách thêm, loại bỏ, hoặc sắp xếp các điều kiện trên Control Layout.

Điều khiển Layout có 2 dạng một là dạng bảng và dạng thứ 2 là dạng ngăn xếp (Stacked).

- Trên điều khiển Layout dạng bảng, các điều kiện trong nó được sắp xếp thành các hàng và cột giống như một bảng biểu, với các Label được đặt trên đầu. Điều khiển Layout dạng bảng luôn chia Form thành 2 vùng, một chứa các điều khiển hiển thị ở phía dưới, và một là vùng phía trên chứa các Label.

Last Name	First Name	Business Phone
Cencini	Andrew	(123)456-7890
Freehafer	Nancy	(123)456-7890
Giussani	Laura	(123)456-7890
Hellung-Larsen	Anne	(123)456-7890
Kotas	Jan	(123)456-7890
Neipper	Michael	(123)456-7890
Sergienko	Mariya	(123)456-7890
Thorpe	Steven	(123)456-7890
Zare	Robert	(123)456-7890

Trên điều khiển Layout kiểu ngăn xếp, các điều kiện trên đó được bố trí theo chiều dọc giống như trên một trang giấy, với mỗi Label bên trái của từng điều khiển. Khác với điều khiển Layout kiểu bảng, điều khiển Layout kiểu ngăn xếp luôn chứa các Control ở một khu vực duy nhất trên Form.

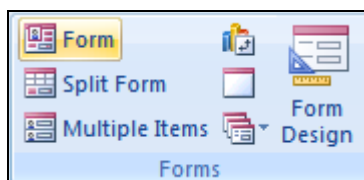
Last Name	Cencini
First Name	Andrew
Business Phone	(123)456-7890
Last Name	Freehafer
First Name	Nancy
Business Phone	(123)456-7890
Last Name	Giussani

Bạn có thể thiết kế nhiều Control Layout ở một trong hai kiểu nói trên, trên cùng một Form. Ví dụ, bạn có thể có một Control Layout dạng bảng để tạo dòng dữ liệu cho từng bản ghi, và một hoặc nhiều Control Layout dạng ngăn xếp ở phía dưới chứa nhiều hơn các thông tin của cùng bản ghi đó.

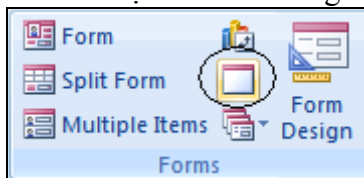
#### Tạo mới một Control Layout

Access tự động tạo ra Control Layout dạng ngăn xếp trong các trường hợp sau đây:

- Tạo một Form mới bằng cách Click chuột vào nút Form trong nhóm Forms trên Tab Create.



- Bạn tạo một Form mới bằng cách Click vào nút Blank Form trong nhóm Forms trên Tab Create sau đó kéo thả các trường cần hiển thị lên Form trong Pane Field List.







Trên một Form đã tồn tại bạn có thể tạo mới Control Layout theo các bước dưới đây:

**Bước 1:** Chọn điều khiển mà bạn muốn bổ sung vào Layout.

**Bước 2:** Nếu bạn muốn thêm nhiều Control khác nhau lên cùng một Layout, bạn giữ phím Shift sau đó check chọn những Control cần thêm.

**Bước 3:** Sau đó có thể làm theo một trong những cách sau đây:

- Trên Tab Arrange, tìm tới nhóm Control Layout Click chọn Tabular  hoặc Stacked .

- Click phải chuột chọn một hoặc nhiều Control, chỉ tới Layout sau đó chọn Tabular  hoặc Stacked .



Bằng một trong 2 cách trên Access sẽ tạo Control Layout và thêm các điều khiển được chọn lên đó.

### **Chuyển một Control Layout từ dạng bảng sang dạng ngăn xếp và ngược lại.**

Để chuyển kiểu của một Control Layout từ dạng này sang dạng khác bạn làm như sau:

- Chọn Control Layout bằng cách Click chọn vào ô Select ở góc trái của Layout. Toàn bộ các Cell trong Layout sẽ được chọn.

- Sau đó bạn làm theo một trong các cách sau:

+ Trên Tab Arrange tìm tới nhóm Control Layout Click chọn kiểu Layout muốn tạo (Tabular  hoặc Stacked .

+ Click phải chuột chọn một hoặc nhiều Control, chỉ tới Layout sau đó chọn Tabular  hoặc Stacked .



Access sẽ tự động sắp xếp lại các Control trong Layout theo dạng mà bạn đã chọn.



### **Chia một Control thành hai dạng.**

Bạn có thể chia một Control Layout thành dạng khác nhau theo các bước dưới đây:

**Bước 1:** Giữ phím Shift sau đó Click chọn các Control mà bạn muốn di chuyển thành một dạng Control Layout mới.

**Bước 2:** Sau đó làm theo một trong hai cách sau đây:

- Trên Tab Arrange tìm tới nhóm Control Layout Click chọn kiểu Layout muốn tạo (Tabular  hoặc Stacked .

- Click phải chuột chọn một hoặc nhiều Control, chỉ tới Layout sau đó chọn Tabular  hoặc Stacked .

Bằng một trong hai cách trên Access sẽ tạo ra một Control Layout mới và thêm lên đó những điều khiển mà bạn vừa chọn.

### Sắp xếp lại các Control trên một Control Layout

Bạn có thể di chuyển một Control trên Control Layout bằng cách kéo thả nó tới vị trí mà bạn muốn. Cũng như là việc di chuyển vị trí các trường trong bảng, Thanh cuộn ngang dọc sẽ cho biết nơi bạn có thể thả nút chuột.

Bạn cũng có thể di chuyển một Control từ một Control layout này sang một Control Layout khác mà chúng có cùng kiểu định dạng. Ví dụ, bạn có thể kéo một Control từ một Control Layout dạng ngăn xếp sang một Control Layout cũng dạng ngăn xếp khác chứ không phải một Control Layout dạng bảng.

### Thêm Control vào một Control Layout

Để thêm một trường mới từ Pane Field List vào một Control Layout đã tồn tại bạn làm như sau:



Kéo thả trường từ Pane Field List vào Control layout. Thanh cuộn ngang, dọc sẽ cho bạn biết nơi bạn sẽ thả giữ nút trên chuột.

Để thêm một Control đã tồn tại vào một Control Layout đã có bạn làm như sau:

**Bước 1:** Chọn Control đầu tiên mà bạn muốn thêm vào Control Layout.

**Bước 2:** Nếu bạn muốn thêm các nhiều Control trên cùng một Layout, bạn giữ phím Shift rồi check chọn lần lượt các Control. Ngoài ra bạn cũng có thể chọn các control trên Layout khác.

**Bước 3:** Kéo thả các trường các trường được chọn vào Layout. Thanh cuộn ngang, dọc sẽ báo cho bạn biết vị trí của trường mà bạn có thể thả giữ chuột.

**Bước 4:** Trên Tab Arrange trong nhóm Control Layout Click chọn kiểu Layout mà bạn muốn đưa các Control vào đó. Nếu bạn muốn thêm các điều khiển vào Layout dạng bảng bạn chọn Tabular , ngược lại bạn chọn Stacked .



Sau bước này Access sẽ tạo ra một Layout mới và thêm các điều Control mà bạn đã chọn lên đó.

### Loại bỏ Control trên một Control Layout.

Việc loại bỏ Control từ một Control Layout cho phép bạn có thể thực hiện ở bất kỳ vị trí nào trên Form mà không bị ảnh hưởng tới các điều khiển khác

Bạn chọn tới Control cần loại bỏ trên Control Layout. Để lựa chọn nhiều Control một lúc bạn giữ phím Shift sau đó check chọn nhưng Control cần loại bỏ. Để Lựa chọn tất cả các Control trên Control Layout đó bạn Click chọn và hộp chọn ở góc trên bên phía trái của Control Layout.

Và tiếp theo hãy làm theo các bước dưới đây:

- Trên Tab Arrange tìm tới nhóm Control Layout Click chọn Remove .
- Click phải chuột vào một trong những điều khiển được chọn trở tới mục Layout chọn Remove .

### 4.4. Tìm Layout View và Design View.

**Layout view:** Layout View là chế độ hiển thị trực quan nhất để bạn có thể sử dụng trong quá trình chỉnh sửa Form, ở chế độ này nó cung cấp cho bạn những công cụ mà gần như có thể thay đổi mọi thứ mà bạn muốn khi làm việc với Form trong Microsoft Office Access 2007. Ở chế độ Layout View, Form đang chạy thực sự, vì vậy bạn có thể nhìn thấy dữ liệu xuất hiện trên Form, nhưng vẫn có thể thay đổi những thiết kế của Form ở chế độ này. Đây là chế độ hiển thị tiện dụng nhất để thiết lập độ rộng của các điều khiển cũng như có thể thực hiện hầu hết các công việc bố trí sự bố cục của Form trên giao diện người dùng.

Một số công việc không thể thực hiện được trong chế độ Layout View mà yêu cầu phải chuyển sang chế độ Design View. Trong những tình huống đó Access sẽ hiển thị thông báo, rằng bạn cần phải chuyển sang chế độ hiển thị Design View để thực hiện những công việc đó.

**Design view:** Chế độ hiển thị Design View cho phép bạn xem chi tiết hơn về cấu trúc Form đang thiết kế. Chế độ này cho phép chúng ta có thể nhìn thấy các phần Header, Detail,


Footer. Form khi hiển thị ở chế độ Design View sẽ không chạy vì vậy bạn không thể nhìn thấy dữ liệu mà nó chứa bên trong khi đang thay đổi thiết kế. Tuy nhiên có một số công việc chính mà bạn có thể thực hiện dễ dàng ở chế độ này thay vì Layout View:

- Có thể thêm nhiều hơn các Control lên Form như các nhãn, hình ảnh, đường Line, hình chữ nhật.
- Chính sửa nguồn dữ liệu cho các TextBox bằng cách sử dụng pane Property sheet.
- Thay đổi kích thước của các phần (sections) trên Form, ví dụ như phần tiêu đề, phần chi tiết.
- Thay đổi một số thuộc tính chính mà không thể thay đổi được ở chế độ Layout View.

## 5. Tinh chỉnh Form

### Làm việc ở chế độ hiển thị Layout View


Sau khi bạn tạo ra một Form, bạn có thể dễ dàng tinh chỉnh các thiết kế của nó khi hiển thị ở chế độ Layout View. Sử dụng dữ liệu thực tế như hướng dẫn của bạn, bạn có thể sắp xếp lại các điều khiển và điều chỉnh kích cỡ của chúng. Cũng như có thể thay thế các Control mới trên Form và thiết lập các thuộc tính cho Form và các điều khiển trên đó.

Để chuyển về chế độ Layout View Click phải chuột vào Form đó trên pane Navigation Pane, tiếp theo chọn layout View .

Access sẽ hiển thị Form ở chế độ Layout View.

Bạn có thể sử dụng Property Sheet để chỉnh sửa thuộc tính cho Form và các điều khiển trong các vùng (Section) trên Form. Muốn thị hộp thoại Property Sheet bạn nhấn phím F4.

Chúng ta cũng có thể sử dụng Pane Field List để thêm một trường từ một bảng hoặc một Query lên Form đang thiết kế. Dưới đây là một trong các cách hiển thị Pane Field List:

- Trên Tab Formatting, trong nhóm Controls chọn Add Existing Fields .
- Click phím Alt + F8


Sau đó bạn có thể kéo thả trực tiếp các trường từ Pane Field List vào Form.

- Để thêm một trường đơn giản, bạn Double Click chuột vào nó hoặc kéo thả nó từ Pane Field List vào một vùng nhất định trên Form nơi mà bạn cần hiển thị.

- Để thêm nhiều hơn một trường, bạn giữ phím Ctrl sau đó dùng chuột check chọn những trường cần muốn thêm. Sau đó kéo thả chúng lên Form.

### Làm việc ở chế độ hiển thị Design View.

Bạn cũng có thể tinh chỉnh những thiết kế của Form khi làm việc với nó ở chế độ Design view. Bạn thêm các Control, các trường lên Form bằng cách bỏ xung chúng lên lưới thiết kế. Hộp thoại Property sheet sẽ cho phép bạn truy cập một số lượng lớn các thuộc tính có thể thiết lập để chỉnh sửa Form.

Để chuyển Form về chế độ Design View, hãy Click phải chuột lên tên Form trong Pane Navigation sau đó chọn Design View .

Access sẽ hiển thị Form ở chế độ Design View

Bạn có thể sử dụng hộp thoại Property Sheet để chỉnh sửa các thuộc tính cho Form cũng như các Control trên Form bằng cách nhấn phím F4.

Bạn có thể sử dụng Pane Field List để thêm một trường từ bảng hoặc Query vào thiết kế của Form. Sau đây là một số cách thực hiện

- Trên Tab Formatting, tìm tới nhóm Controls Click chọn Add Existing Fields .
- Click tổ hợp phím Ctrl + F8

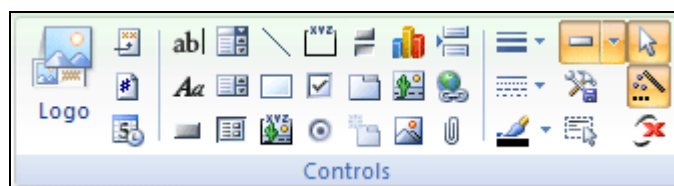
Bạn có thể kéo thả trực tiếp các trường từ Pane Field List lên Form.

- Để thêm một trường, hãy Double Click chuột vào nó hoặc kéo thả từ Pane Field List lên Form.

- Để thêm nhiều hơn một trường, hãy giữ phím Ctrl rồi dùng chuột check chọn những trường cần thiết. Sau đó kéo thả những trường này lên Form.

## 6. Thêm Control lên Form

Một số Control được tạo ra tự động, ví dụ như các Control phụ thuộc được tạo ra khi bạn thêm một trường từ Pane Field List vào Form. Nhiều Control khác cũng có thể được tạo ra ở chế độ Design View bằng cách sử dụng các công cụ trong nhóm Controls trên Tab Design.





*Chú ý:* nhiều công cụ trong nhóm Controls chỉ có thể truy cập khi form của bạn được mở ở chế độ Design.

### Xác định tên của một Control

Để xác định tên của một Control bạn chỉ cần di chuột tới nó, Access sẽ hiển thị tên của Control đó trên Tooltip.

### Sử dụng các công cụ Wizard.

Bạn có thể sử dụng công cụ Wizards để được hỗ trợ trong quá trình tạo các Command Button, list boxes, subforms, combo box và option groups. Trên Tab **Design** tìm tới nhóm Controls, nếu nút **Use Control Wizards** nếu như nó không sáng .

Nếu bạn yêu cầu tạo các Control mà không cần sự trợ giúp của Wizards bạn hãy Click chọn Use Control Wizards để tắt chế độ này .

### Tạo một Control sử dụng các công cụ trong nhóm Controls.

*Bước 1:* Click chọn công cụ tương ứng với kiểu Control mà bạn muốn thêm, ví dụ để tạo một hộp kiểm bạn Click chọn Control TextBox.

*Bước 2:* Click chuột lên lưới thiết kế Form, nơi bạn muốn bắt đầu đặt góc trái của điều khiển đó. Nếu Click một chuột lên Form, Control sẽ được tạo với cỡ mặc định, còn không bạn có thể kéo thả chuột để tùy chỉnh cỡ của Control trên Form.

*Bước 3:* Trong trường hợp lựa chọn công cụ Wizards, khi bạn chọn một Control, chức năng này sẽ hướng dẫn từng bước để bạn có thể tạo ra một Control.

*Bước 4:* Trong trường hợp vị trí của Control trên Form chưa thích hợp, bạn có thể sắp xếp lại hoặc di chuyển chúng đến vị trí thích hợp theo các bước sau:


- Click chọn Control trên Form.
- Kéo thả Control tới vị trí mà bạn muốn.

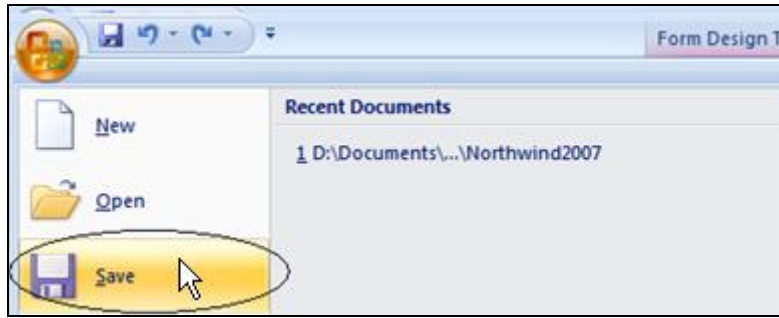
Nếu bạn sử dụng một Control Wizard, công cụ này sẽ lần lượt thực hiện các bước để giúp bạn kiểm soát việc Bind (ràng buộc) dữ liệu từ một trường vào Control đó. Tuy nhiên thủ tục này tạo ra một Control mà không bị ràng buộc bởi các trường dữ liệu. Nếu Control này là dạng Control có thể hiển thị dữ liệu (TextBox, CheckBox,...), bạn phải nhập vào tên trường hoặc biểu thức cho giá trị của thuộc tính Control Source của Control, khi đó nó mới có thể hiển thị được dữ liệu bạn mong muốn. Để hiển thị bằng thuộc tính cho các Control hãy Click phím F4.

## 7. Lưu lại các công việc của bạn

Sau khi lưu lại Form như thiết kế, bạn có thể chạy nó thường xuyên nếu bạn muốn. Các thiết kế vẫn giữ nguyên, nhưng dữ liệu sẽ được hiển thị ở thời điểm bạn chạy Form. Nếu cần thay đổi bạn hoàn toàn có thể chỉnh sửa hoặc tạo mới Form dữ trên những gì mà đã làm trước đó.

### Lưu lại những thiết kế của Form



*Bước 1:* Click chọn nút Microsoft Office Button , sau đó chọn Save.



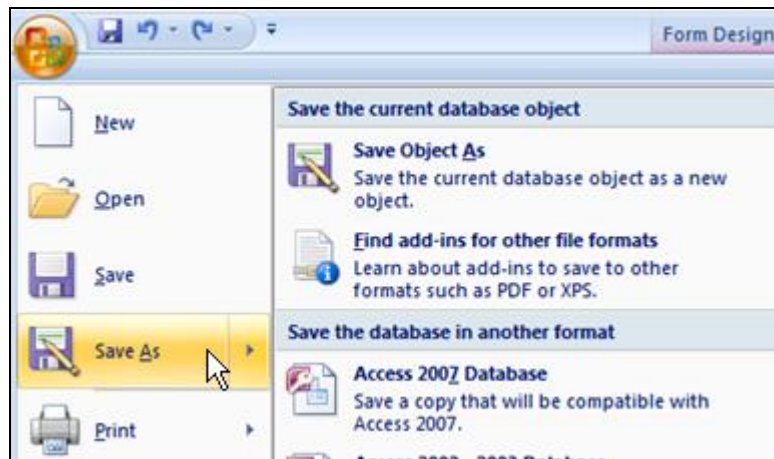
Hoặc chọn nút Save  trên thanh Quick Access Toolbar (Ctrl + S)

Bước 2: nếu Form được lưu lần đầu tiên một hộp thoại sẽ xuất hiện yêu cầu bạn Click vào tên Form trong hộp Click Form Name, cuối cùng Click Ok để hoàn tất.

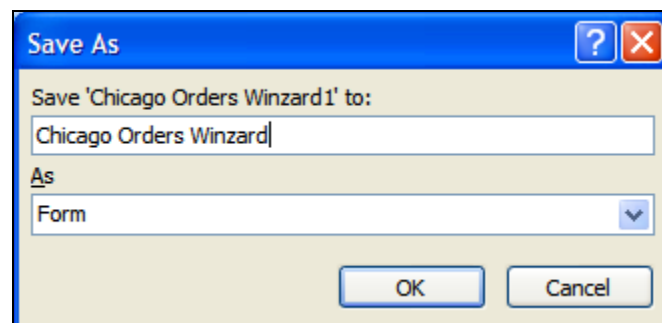
**Lưu lại thiết kế của Form với một tên khác.**

Bước 1: Click chọn nút **Microsoft Office Button** , chọn Save As .

Bước 2: Trong hộp thoại Save As, bạn gõ tên Form mới trong hộp Click Save Form To.



Chọn giá trị là Form trong hộp combo As.



Cuối cùng Click Ok để hoàn tất.

## Bài 5. Làm việc với Report

Với Microsoft Office Access 2007 bạn có thể tạo ra nhiều loại báo cáo khác nhau, từ đơn giản đến phức tạp. Đầu tiên hãy nghĩ về báo cáo dùng để hiển thị những bản ghi từ nguồn dữ liệu. Cho dù là báo cáo với một danh sách đơn giản, hoặc báo cáo nhóm chỉ tiêu doanh thu theo vùng miền, thì việc đầu tiên bạn nên quan tâm là những trường chứa dữ liệu nào cần hiển thị lên báo cáo, và những bảng hay Query nào chứa chúng.

Sau khi bạn chọn được những bản ghi cần thiết, bạn có thể dễ dàng tạo ra một báo cáo bằng công cụ Report Wizard. Report Wizard là một chức năng trong Access, chức năng này sẽ hướng dẫn bạn thông qua các câu hỏi, báo cáo được tạo ra dựa trên những câu trả lời của bạn cho những câu hỏi đó.

### 1. Lựa chọn bản ghi nguồn.

Một báo cáo bao gồm những thông tin được đưa vào từ các bản ghi trong một bảng hoặc một Query. cũng như các thông tin đã được lưu giữ cùng với lúc báo cáo được thiết kế thiết kế, chẳng hạn như nhãn, đề mục, và các hình ảnh. Các bảng hoặc truy vấn cung cấp các dữ liệu này còn được biết đến như là nguồn bản ghi của báo cáo. Nếu những trường mà bạn muốn sử dụng đều nằm trong một bảng thì bạn có thể sử dụng bảng đó như nguồn bản ghi. Nếu những trường sử dụng nằm ở nhiều bảng khác nhau bạn cần bạn sẽ phải sử dụng một hoặc nhiều Query như là nguồn bản ghi. Những Query này có thể đã tồn tại trong cơ sở dữ liệu của bạn, hoặc nếu cần bạn có thể tạo mới những Query để có thể chứa được những dữ liệu mà bạn muốn hiển thị lên báo cáo.

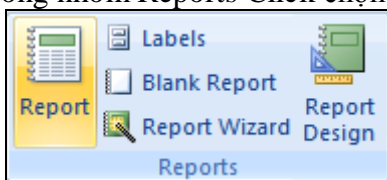
### 2. Tạo báo cáo

#### Sử dụng công cụ Report

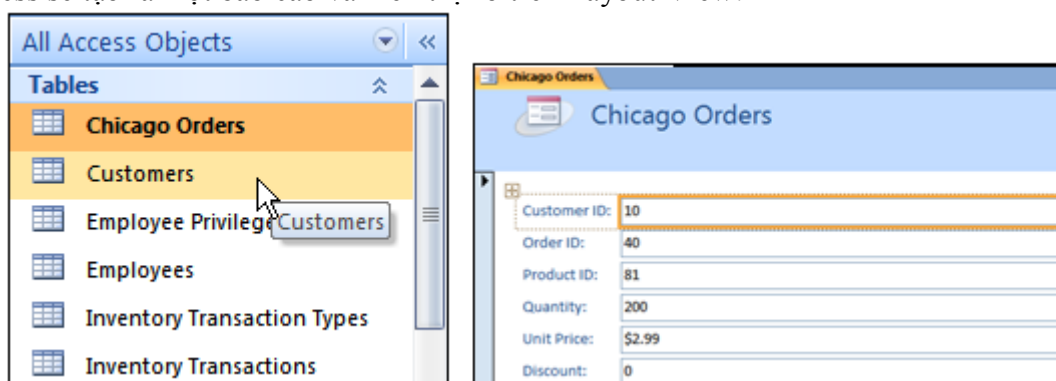
Các công cụ Report cung cấp con đường nhanh nhất để tạo ra một báo cáo. Bởi vì nó tự động tạo ra báo cáo mà không cần lấy thông tin hỏi đáp từ người thiết kế. Báo cáo hiển thị tất cả các trường từ nguồn dữ liệu của một bảng hoặc Query. Tuy các báo cáo sử dụng công cụ Report không hoàn hảo và không thể tạo ra những sản phẩm cuối cùng. Nhưng nó thật sự hữu dụng khi để hiển thị nhanh dữ liệu. Bạn có thể lưu lại báo cáo và chỉnh sửa nó ở chế độ hiển thị Layout View hoặc Design View sao cho có thể phục vụ tốt nhất mục đích của bạn.

**Bước 1:** Trong pane Navigation, Click chọn Table hoặc Query mà bạn muốn tạo báo cáo.

**Bước 2:** Trên Tab Create, trong nhóm Reports Click chọn Report .



Access sẽ tạo ra một báo cáo và hiển thị nó trên Layout View.



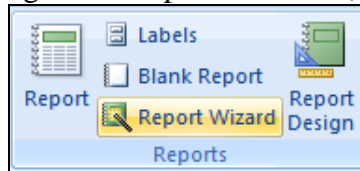


Sau khi xem các báo cáo, bạn có thể lưu và sau đó đóng cả báo cáo và các bảng hoặc Query mà bạn đã sử dụng làm các bản ghi nguồn. Tại một thời điểm nào đó cần mở lại báo cáo, khi đó Access sẽ hiển thị những dữ liệu mới nhất từ nguồn dữ liệu của báo cáo.

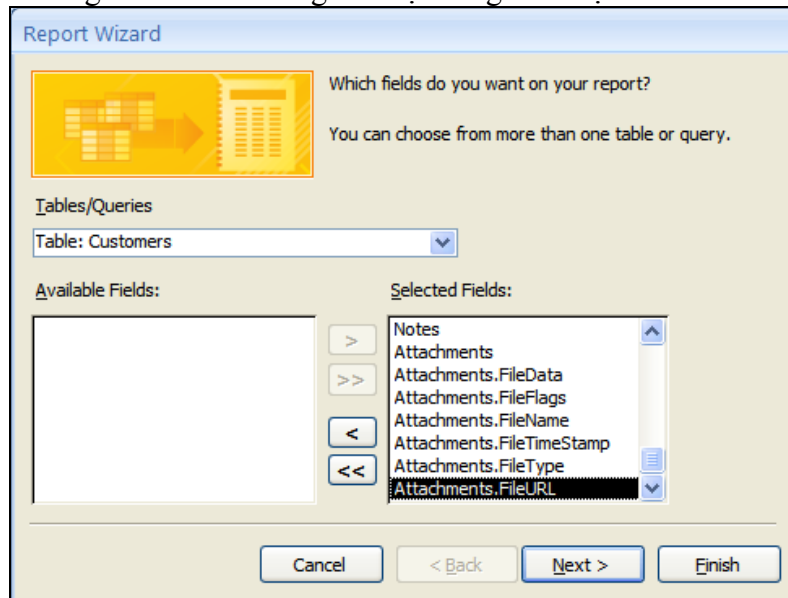
### Sử dụng công cụ Report Wizard.

Bạn sử dụng Report Wizard để có nhiều chọn lựa hơn về những trường sẽ xuất hiện trên báo cáo. Và cũng có thể chỉ ra các trường được nhóm hoặc sắp xếp, ngoài ra có thể sử dụng nhiều hơn một bảng hoặc Query làm dữ liệu nguồn cho báo cáo.

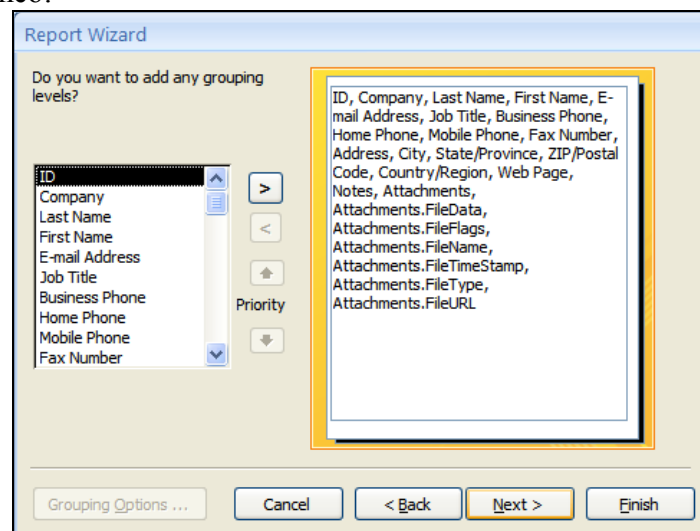
**Bước 2:** Trên Tab Create, trong nhóm Reports Click chọn Report Wizard.



**Bước 3:** Chức năng Wizard sẽ hướng dẫn bạn từng bước tạo ra báo cáo:



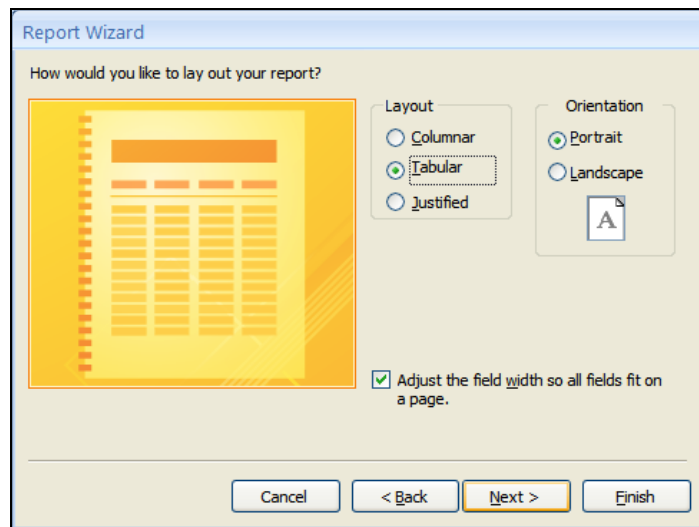
Hộp thoại này cho phép bạn lựa chọn các trường hiển thị trên báo cáo. Click Next để chuyển bước tiếp theo.



Trong cửa sổ này cho phép cho phép bạn chọn trường cần nhóm. Click Next để chuyển bước tiếp theo.



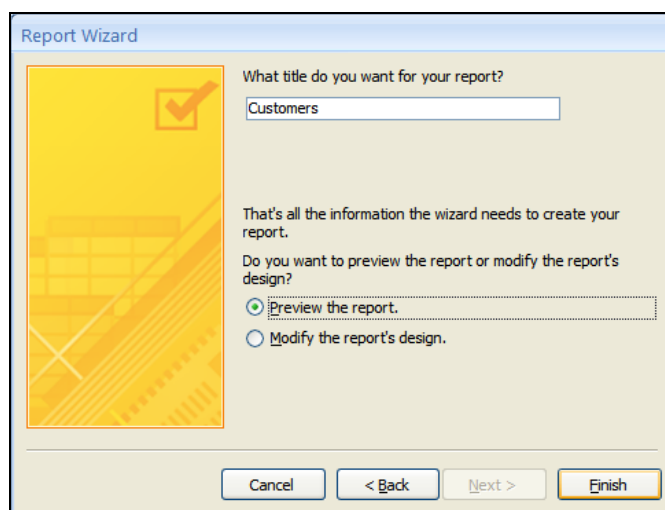
Chọn trường sắp xếp cho báo cáo. Click Next để chuyển bước tiếp theo.



Chọn kiểu bố trí báo cáo trong khung Layout, và chọn khổ giấy ngang hay dọc ở khung Orientation. Click Next để chuyển bước tiếp theo.

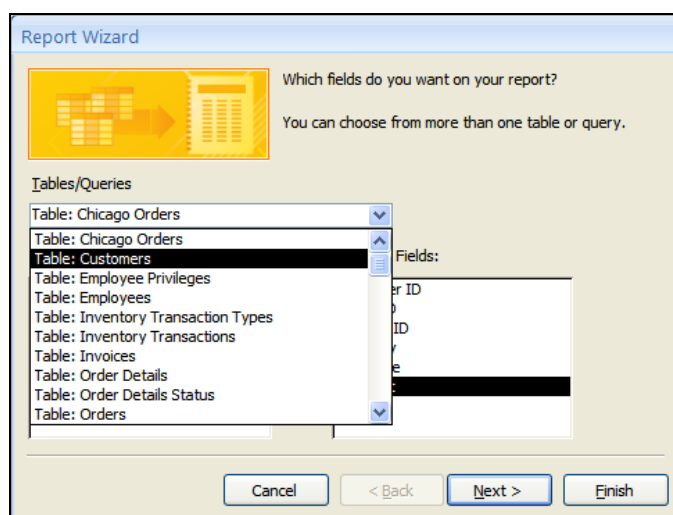


Chọn Style cho báo cáo, Click Next để chuyển bước tiếp theo.



Nhập vào tên của báo cáo trong phần What title do you... Cuối cùng chọn Finish

*Chú ý:* Nếu bạn muốn lấy các trường vào báo cáo từ nhiều bảng hoặc Query, bạn không Click Next hoặc Finish sau khi đã lựa chọn xong trường từ Table hoặc Query đầu tiên trên trang đầu tiên của Report Wizards.



Thay vì chuyển sang bước tiếp theo bạn tiếp tục Click chọn các bảng hoặc Query khác cần lấy dữ liệu trong mục Tables/Queries.

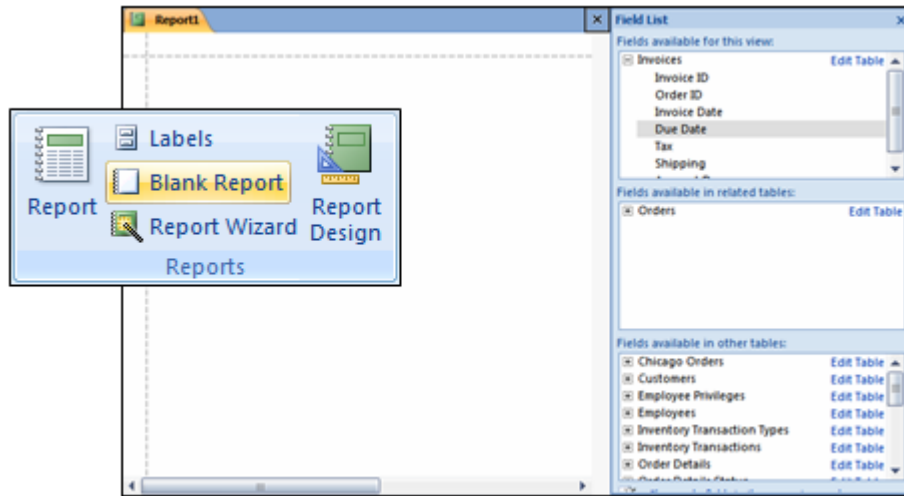
Sau khi đã chọn xong tất cả các trường cần thiết bạn có thể Click Next hoặc Finish để máy tính tiếp tục tạo báo cáo.

### Sử dụng công cụ Blank Report

Nếu không quan tâm đến việc sử dụng công cụ Report hoặc Report Wizards, bạn có thể sử dụng công cụ Blank Report để xây dựng một báo cáo từ đầu. Với công cụ này bạn sẽ nhanh chóng tạo ra một báo cáo, đặc biệt thích hợp khi bạn dự định chỉ thêm vào đó một số trường nhất định. Dưới đây là các bước tạo báo cáo bằng công cụ Blank Report.

**Bước 1:** Trên Tab Create, trong nhóm Reports Click chọn Blank Report.

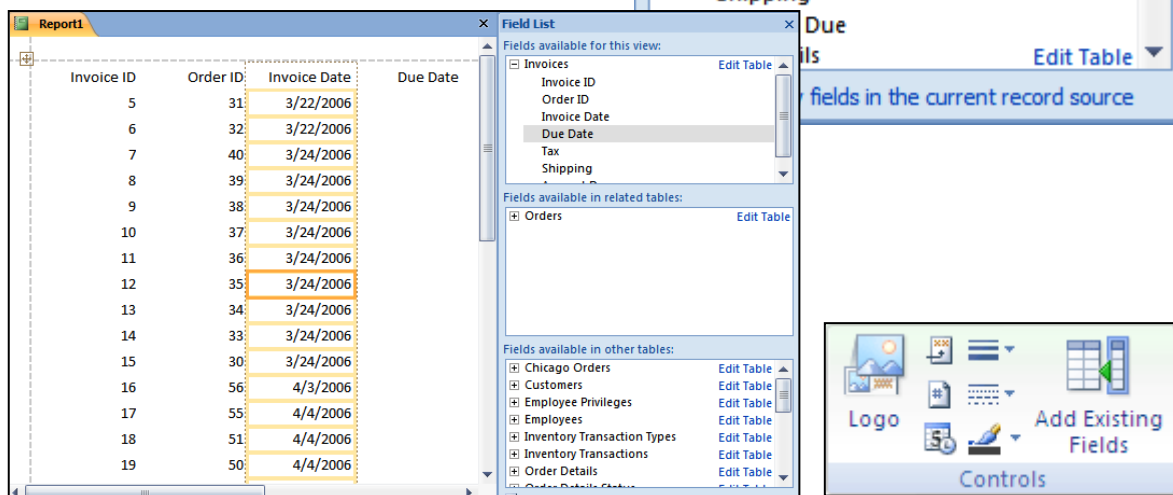
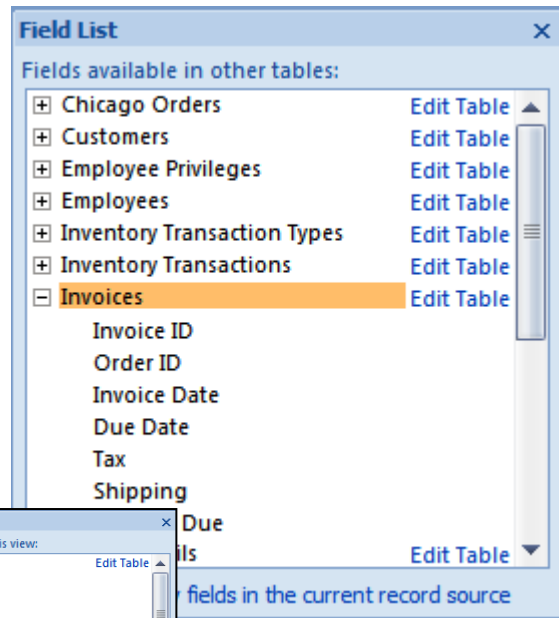
Một Blank Report được hiển thị ở chế độ Design, và Pane Field List xuất hiện bên trái của màn hình chính.



**Bước 2:** Trong Pane Field List Click chuột vào nút có hình (+) vào bảng chứa những trường mà bạn muốn hiển thị dữ liệu trên báo cáo.

**Bước 3:** Click chọn một trường hoặc có thể giữ phím Ctrl rồi Click chọn nhiều trường một lúc rồi kéo thả lên báo cáo.

**Bước 4:** Sử dụng các công cụ trong nhóm Control trên Tab Formatting để thêm Logo, tiêu đề, số trang, ... cho báo cáo.



### 3. Tìm hiểu về Report section

Trong Access, việc thiết kế các báo cáo được chia thành nhiều phần (Section). Bạn hãy hiển thị báo cáo ở chế độ Design View để nhìn thấy các vùng (section) đó. Muốn tạo các báo cáo đáp ứng được yêu cầu, cần phải hiểu rõ từng phần, và cách làm việc với các Section đó. Ví dụ, phần nào có thể lựa chọn để đặt các điều Control tính toán? Và làm thế nào để Access tính toán ra kết quả. Dưới đây là danh sách tóm tắt cả kiểu Section và cách sử dụng chúng.

- **Report Header:** Đây là phần được in ra chỉ một lần khi bắt đầu báo cáo. Sử dụng các tiêu đề báo cáo cho các thông tin mà thông thường có thể xuất hiện trên một trang bìa, như là một biểu tượng, một tiêu đề, hoặc một ngày tháng. Các Report Header được in trước khi in trang đầu tiên

- **Page Header:** Đây là Section sẽ được in trên đầu mỗi trang. Ví dụ, sử dụng Page Header để lặp lại tiêu đề báo cáo trên tất cả các trang.

- **Group Header:** Đây là phần được in khi bắt đầu mỗi nhóm bản ghi. Sử dụng Section này để in ra các tên nhóm. Ví dụ, trong một báo cáo được nhóm theo sản phẩm lúc đó Group Header được sử dụng để in tên các sản phẩm. Khi bạn đặt một Control tính toán mà sử dụng hàm Sum trong Group Header, Access sẽ tính tổng cho từng nhóm.

- **Detail:** Section này in tất cả các dòng có trong nguồn bản ghi. Đây là nơi bạn đặt các Control hiển thị dữ liệu và là phần thân chính của báo cáo.

- **Group Footer:** Đây là phần được in ở cuối mỗi nhóm bản ghi. Sử dụng Section này để in các thông tin tóm tắt cho một nhóm.

- **Page Footer:** Được in ở cuối mỗi trang. Sử dụng phần này để in thông tin về số trang cũng như trang hiện thời của báo cáo.

- **Report Footer:** Đây là phần chỉ được in duy nhất một lần. Sử dụng phần này để in tổng số hoặc các thông tin tóm tắt về toàn bộ báo cáo.

*Chú ý:* Ở chế độ Design, Report Footer xuất hiện dưới phần Page Footer. Tuy nhiên khi báo cáo được in hoặc xem trước khi in, thì Report Footer lại xuất hiện phía trên Page Footer, chỉ cần Group Footer cuối cùng hoặc dòng cuối cùng của phần Detail nằm ở trang cuối.

### 4. Tìm hiểu về các Control

Tương tự như phần “Tìm hiểu về các Controls” – Bài 4

### 5. Tinh chỉnh báo cáo

Tương tự như phần “Tinh chỉnh Form” – Bài 4

### 6. Thêm Control vào báo cáo

Tương tự như phần “Thêm Control lên Form” – Bài 4

### 7. Xem báo cáo

Có nhiều cách để xem báo cáo, cách được lựa chọn phụ thuộc vào những gì mà bạn muốn thao tác với báo cáo và dữ liệu trên đó.

- Nếu bạn muốn tạm thời thực hiện các thay đổi với dữ liệu sẽ xuất hiện trên báo cáo trước khi bạn in nó, hoặc nếu bạn muốn sao chép dữ liệu từ các báo cáo vào bộ nhớ tạm, thì hãy sử dụng Report View.

- Nếu bạn muốn có thể thay đổi các thiết kế ngay khi báo cáo đang hiển thị dữ liệu bạn hãy sử dụng chế độ Layout View.

- Nếu bạn chỉ đơn giản muốn xem những gì báo cáo sẽ được thấy khi in ra, bạn hãy sử dụng chế độ Print Preview.

#### Xem báo cáo ở chế độ Report View

Report View là chế độ hiển thị mặc định được sử dụng khi bạn Double Click chuột vào một báo cáo trên pane Navigation nếu như nó chưa được mở.

Trong trường hợp báo cáo đang mở ở một chế độ nào đó bạn Click phải chuột vào tên báo trên Pane Navigation chọn Report View.

### Làm việc với dữ liệu ở chế độ Report View

Trong chế độ Report View, bạn có thể lựa chọn nội dung văn bản trên đó và Copy vào bộ nhớ tạm của máy tính. Để lựa chọn toàn bộ các cột của một số hàng, bạn Click giữ chuột vào lề bên trái rồi kéo tới hàng muốn chọn. Sau đó có thể copy những hàng đã chọn đó bằng một trong các cách dưới đây:

- Trên Tab Home, trong nhóm Clipboard, Click chọn Copy .

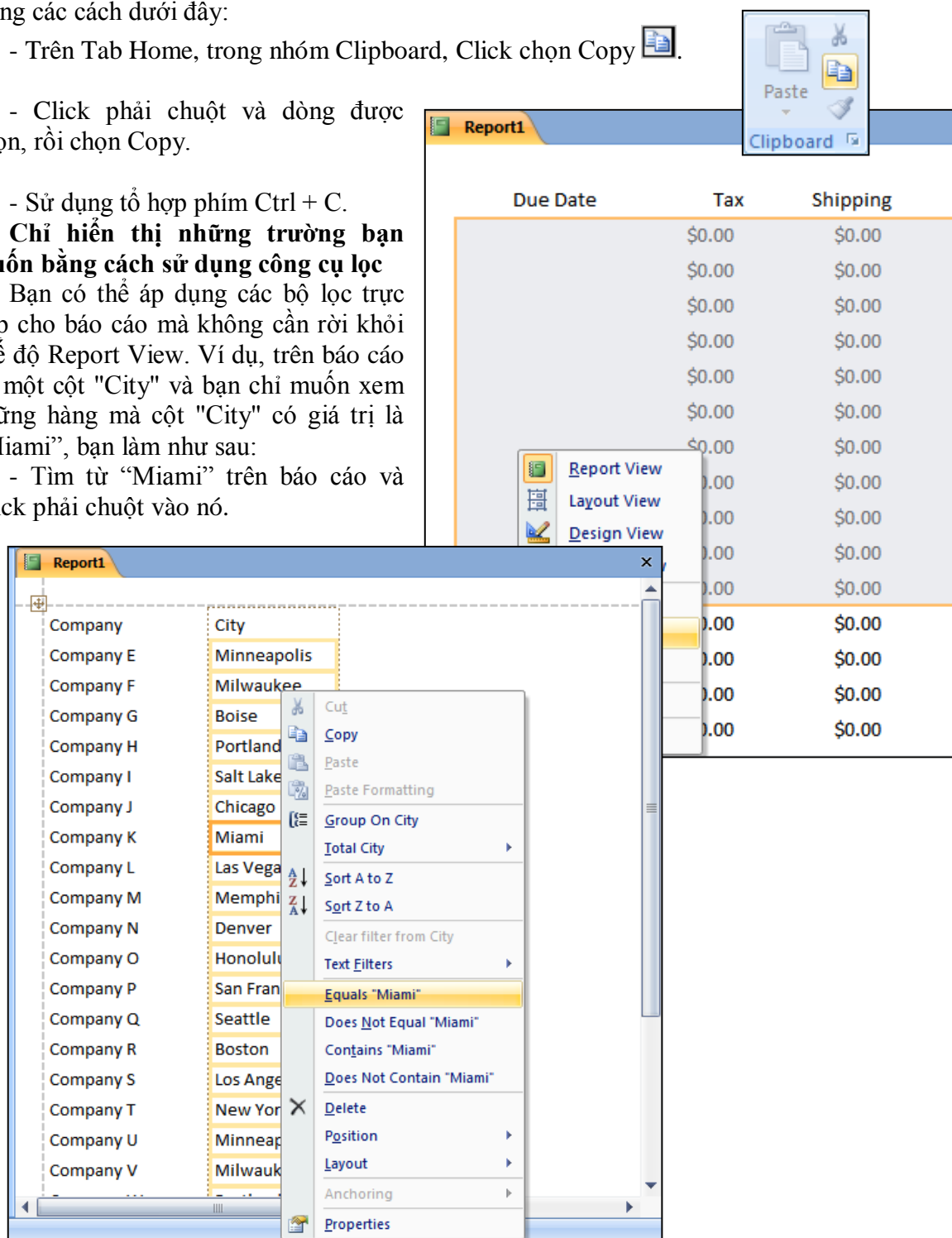
- Click phải chuột và dòng được chọn, rồi chọn Copy.

- Sử dụng tổ hợp phím Ctrl + C.

### Chỉ hiển thị những trường bạn muốn bằng cách sử dụng công cụ lọc

Bạn có thể áp dụng các bộ lọc trực tiếp cho báo cáo mà không cần rời khỏi chế độ Report View. Ví dụ, trên báo cáo có một cột "City" và bạn chỉ muốn xem những hàng mà cột "City" có giá trị là "Miami", bạn làm như sau:

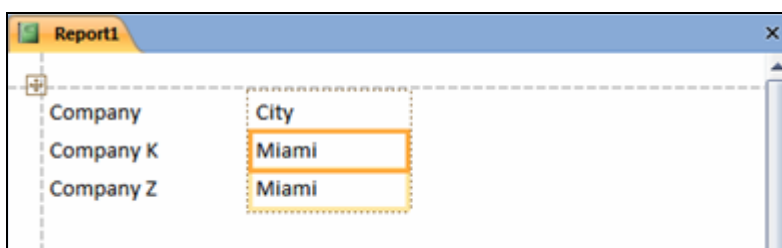
- Tìm từ "Miami" trên báo cáo và Click phải chuột vào nó.



The screenshot shows a report window titled 'Report1' with a table containing columns 'Due Date', 'Tax', and 'Shipping'. The 'Tax' and 'Shipping' columns show values of '\$0.00'. A context menu is open over the 'City' column, listing various actions like 'Cut', 'Copy', 'Paste', and 'Filter'. The 'Filter' section is expanded, and 'Equals Miami' is selected. A 'Clipboard' menu is also visible at the top right of the report window.

- Click chọn Equals "Miami".

Access sẽ tạo và áp dụng bộ lọc, lúc này báo cáo chỉ còn những bản ghi có cột City là Miami.



### Tạo một bộ lọc chi tiết hơn.

**Bước 1:** Click phải chuột vào trường cần lọc.

**Bước 2:** Chọn Common Filters.

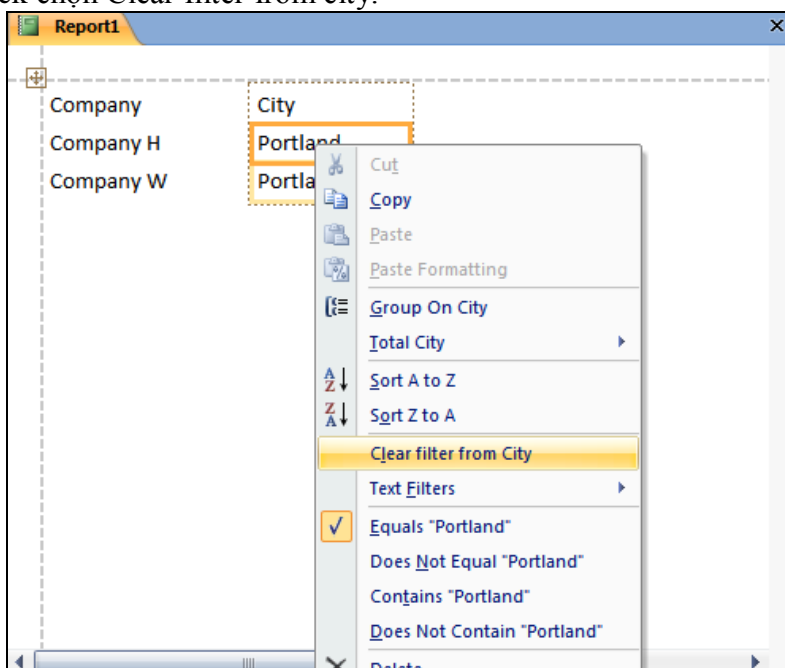
**Bước 3:** Click chọn điều kiện lọc mà bạn muốn.

**Bước 4:** Nhập vào giá trị lọc.

### Hủy bỏ một bộ lọc

**Bước 1:** Click phải chuột vào trường mà bạn muốn loại bỏ bộ lọc.

**Bước 2:** Click chọn Clear filter from city.



**Chú ý:** Nếu bạn áp dụng một bộ lọc cho một báo cáo sau đó lưu và đóng báo cáo lại khi đó bộ lọc sẽ được lưu cùng báo cáo. Tuy nhiên lần sau khi mở lại báo cáo, Access sẽ không sử dụng được bộ lọc. Để sử dụng lại bộ lọc, trên Tab Home tìm tới nhóm Sort & Filter, tiếp theo Click chọn Toggle Filter.

### Xem báo cáo trước khi in sử dụng Print Preview.

Click phải chuột vào báo cáo trong Pane Navigation, sau đó chọn Print Preview.

Các nút trên Navigation có thể giúp bạn xem một cách tuần tự hoặc bất cứ trang nào mà bạn muốn.

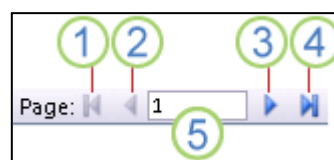
(1). Trở về trang đầu tiên

(2): Trở về trang trước

(3): Chuyển tới trang tiếp theo

(4): Chuyển tới trang cuối cùng

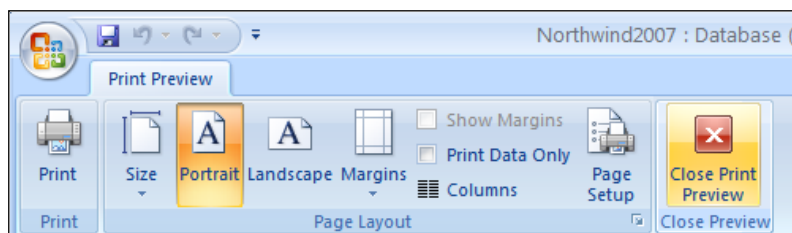
(5): Gõ vào số trang trong hộp Click và nhấn Enter để nhảy tới trang đó.



Ở chế độ Print Preview, bạn có thể phóng to để xem chi tiết hoặc thu nhỏ để xem vị trí của dữ liệu trên trang. Nếu vị trí con trỏ chuột nằm trong báo cáo bạn chỉ cần Click chuột một lần. Để đảo ngược chế độ hiển thị bạn bấm chuột một lần nữa. Bạn cũng có thể sử dụng Control Zoom trên thanh trạng thái của Access để tùy chỉnh cỡ phóng to hay thu nhỏ báo cáo.

Để đóng chế độ Print Preview bạn làm theo một trong các cách sau đây:

- Trên Tab Print Preview chọn nút Close Print Preview.
- Click phải chuột vào báo cáo trên Pane Navigation sau đó chọn Layout View hoặc chọn Design View.







## 8. In báo cáo

Bạn có thể in một báo cáo trong khi nó đang được mở ở bất kỳ chế độ xem nào, hoặc thậm chí trong khi nó đang đóng. Trước khi in, hãy chắc chắn là đã thiết lập các thông số cài đặt cho trang in, như lề, khổ giấy trang in, ... Access lưu lại những cài đặt đó vì vậy chỉ phải thiết lập chúng một lần duy nhất. Sau này bạn cũng có thể thiết lập lại nếu yêu cầu in ấn thay đổi.

### Thay đổi định dạng trang in.



- Mở một báo cáo ở chế độ Print Preview. Bạn có thể thay đổi các thiết lập cho trang in ở bất kỳ chế độ nào, nhưng ở chế độ hiển thị Print Preview là trực quan nhất bởi khi đó có thể nhìn thấy ngay lập tức những thay đổi mà bạn vừa thực hiện.

- Trên Tab Print Preview tìm tới nhóm Page Layout, Click chọn Portrait  hoặc Landscape  để thiết lập các trạng thái định hướng. Size  để thiết lập kích thước cho trang in, Margins  để điều chỉnh lề, ...

- Sau khi thực hiện một thay đổi, hãy sử dụng các nút trên Navigation để xem thử một số trang để chắc chắn rằng bạn đã không tạo ra bất kỳ vấn đề gì về định dạng trên các trang phía sau.

### In báo cáo

**Bước 1:** Mở Report ở bất kỳ chế độ nào mà bạn muốn, hoặc có thể chọn báo cáo trên Navigation Pane.

**Bước 2:** Click chọn nút Microsoft Office Button , sau đó Click Print . Hộp thoại Print xuất hiện

**Bước 3:** Nhập các tùy chọn in báo cáo trong hộp thoại, cuối cùng Click Ok để in báo cáo



## Bài 6. Macro

Là một tập hợp chuỗi các hành động có sẵn của Access. Các hành động này sẽ thực thi một tác vụ nào đó. VD: Mở một Form, xem một báo cáo, Thêm một mẫu tin...

### 1. Tạo mới một Macro

B1: Click vào Tab **Create** → chọn **Macro**

B2: Cửa sổ tạo Macro hiện ra

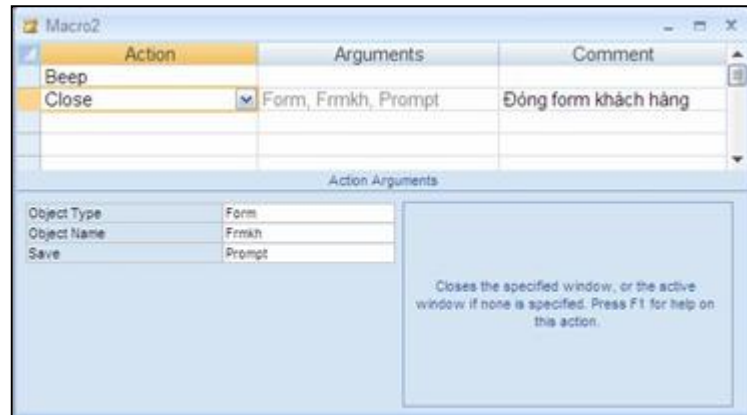
• Chọn hành động cho Macro tại cột **Action**

• Nhập chú thích cho hành động nếu cần tại cột **Comment**

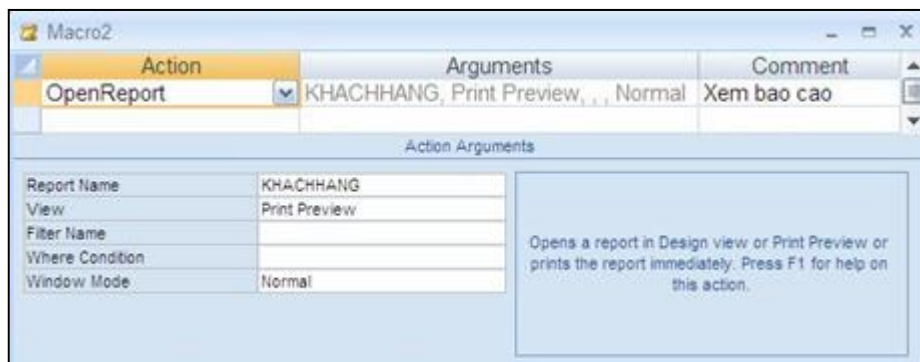
• Chọn tham số cho hành động tại vùng **Action Arguments**

**Arguments**

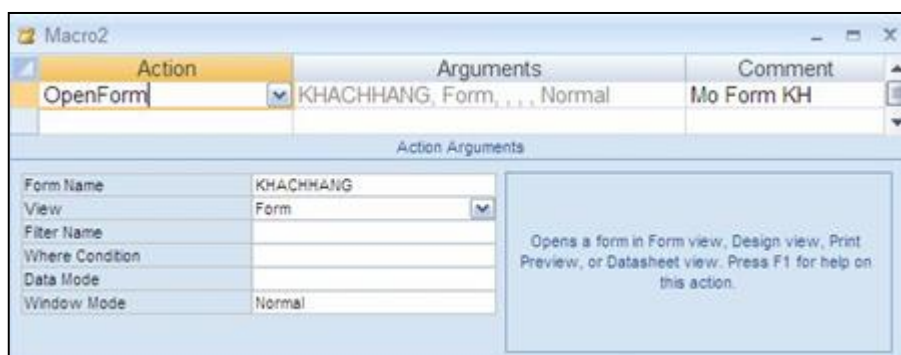
• Lưu Macro



**Macro mở báo cáo khách hàng**



**Macro mở form khách hàng**



### 2. Macro nhóm

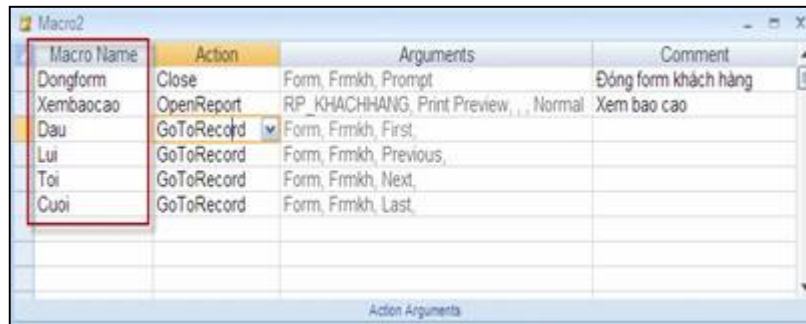
- Là nơi ta có thể tạo một Macro nhưng chứa nhiều Macro con, các Macro này được tạo thành từng nhóm giúp thuận tiện trong quá trình sử dụng

- Macro con trong nhóm sẽ có một tên phân biệt dùng để gọi. Để gọi Macro con trong Macro Group ta dùng cú pháp sau:

**MacroGroupName.MacroName**

**B1:** Click chọn **Tab Create** → chọn nút **Macro**

**B2:** Cửa sổ tạo Macro hiện ra → Click nút **Macro Name**



-Nhập tên Macro tại cột **Macro Name**

-Chọn hành động cho Macro tại cột **Action**

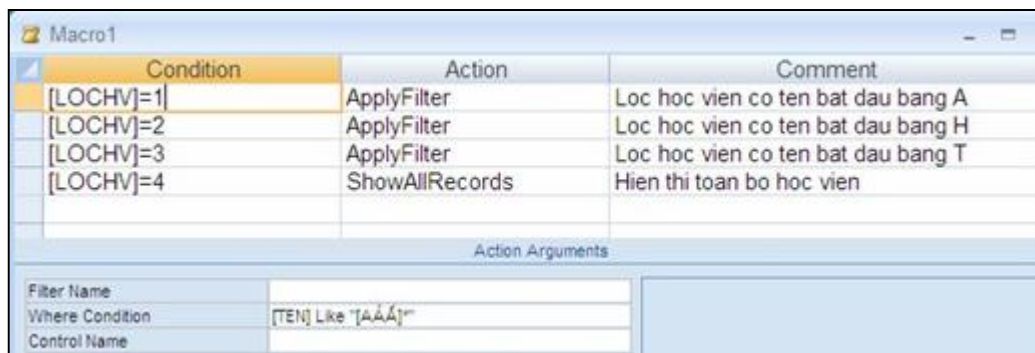
-Nhập chú thích cho hành động nếu cần tại cột **Comment**

-Chọn tham số cho hành động tại vùng **Action Arguments**

-Luu Macro

### 3. Macro điều kiện

Để xem xét một điều kiện nào đó trước khi thực hiện một số hành động trong Macro, ta tạo Macro có điều kiện



**B1:** Click vào **Tab Macro**

**B2:** Tại cửa sổ thiết kế Macro → Chọn nút **Conditions** → cột Conditions sẽ xuất hiện

•Nhập biểu thức điều kiện tại cột Conditions tương ứng với hành động mà ta muốn qui định.

•Chọn hành động tại cột Action

•Nhập chú thích cho hành động tại cột Comment nếu cần

**B3:** Lưu Macro

Form ứng dụng Macro điều kiện để lọc học viên theo vắn



#### 4. Gán Macro cho đối tượng

B1: Tạo đối tượng cần gán Macro (Button, Option group, ...)

B2: Chọn đối tượng → Tại thuộc tính của đối tượng → chọn thẻ Event

B3: Tại sự kiện cần gán đối tượng → click chuột chọn Macro cần gán cho đối tượng

(Vd: Tại nút đóng Form, Sự kiện Onclick ta chọn Macro MCDONGFORM)



HOCVIEN\_Diem

## KET QUA THI

MAHV  MAMH

MAHV	MAMH	DANGKY_MAMH	DIEM
TH01	WORD	WORD	8
*			

Record: 14 1 of 12 No Filter Search

HOCVIEN\_Diem

## KET QUA THI

MAHV:  MAMH:

Form Header

Detail

MAHV	MAHV
MAMH	MONHOC_MAMH
DANGKY_MAMH	DANGKY_MAMH

**Property Sheet**

Selection type: Subform/Subreport

DANGKY Subform\_diem

Format Data Event Other All

Source Object	DANGKY Subform_diem
Link Master Fields	CBOMAHV:CBOMAMH
Link Child Fields	MAHV:MONHOC_MAMH
Filter On Empty Master	Yes
Enabled	Yes
Locked	No



# **Giáo trình Microsoft Access 2000**

# GIỚI THIỆU

Giáo trình này là sự đúc kết kinh nghiệm của tác giả trong giảng dạy các khoá đào tạo về Access cũng như kinh nghiệm sử dụng Access làm hệ quản trị cơ sở dữ liệu và công cụ phát triển phần mềm. Hầu hết nội dung trình bày trong giáo trình đều bám theo những bài tập thực hành cụ thể. Do vậy, sau khi đọc xong và thực hành theo trình tự cuốn sách này, bạn đọc gần như có thể thiết kế và xây dựng được những ứng dụng về quản lý thông tin. Cụ thể ở đây là: Quản lý lương cán bộ một cơ quan và Quản lý bán hàng tại một cửa hàng.

Đối tượng chính của giáo trình là sinh viên, học sinh các trường THCN hệ chuyên CNTT cũng như không chuyên CNTT có học môn Access. Tất nhiên cũng hoàn toàn phù hợp cho những ai quan tâm học hỏi môn học này bởi lẽ các nội dung đều được trình bày rất trực quan, có thứ tự và kết quả rõ ràng.

Thời lượng thiết kế 75 tiết. Trong đó 24 tiết lý thuyết, 46 tiết thực hành và 5 tiết dành cho kiểm tra. Tùy theo mức độ ứng dụng của từng trường, ngành vào môn học này mà mỗi đơn vị sẽ chọn cho mình một quỹ thời gian cũng như các nội dung giảng dạy phù hợp từ giáo trình này.

Nội dung giáo trình gồm một bài mở đầu và 7 chương:

## **Bài mở đầu**

Bài này sẽ trình bày tổng quát về môn học; giới thiệu về Access làm sao hiểu được Access dùng để làm gì? Phân biệt được với một số phần mềm đã được học như Pascal, Foxpro, Word, Excel,... Cuối cùng bạn đọc sẽ hiểu được môi trường làm việc của Access và thực hiện vào ra và quản lý các tệp dữ liệu.

## **Chương 1: Xây dựng cơ sở dữ liệu (CSDL)**

Chương 1 tập trung vào những kiến thức cơ bản giúp có thể xây dựng CSDL Access một cách chắc chắn; đặc biệt thể hiện qua qui trình xây dựng một CSDL tổng kết được nội dung toàn bộ chương học.

## **Chương 2: Truy vấn dữ liệu**

Tập trung vào các kiến thức và kỹ năng giúp xử lý dữ liệu một cách đa dạng thông qua các loại Query.

## **Chương 3: Thiết kế giao diện**

Chương này trình bày các khái niệm cũng như những kỹ thuật, kỹ năng có thể thiết kế được những giao diện phần mềm thông qua việc sử dụng Form và các công cụ đi kèm.

## **Chương 4: In ấn**

Bao gồm các kiến thức cũng như kỹ năng cần thiết để thiết kế các mẫu biểu in ấn đa dạng sử dụng công cụ Report của Access. Hơn nữa kết hợp với Form, sẽ giải quyết được các bài tập mang tính tổng hợp, hoàn thiện.

Nội dung từ chương 1 đến chương 4 mang tính căn bản. Những chương tiếp theo mang tính chuyên ngành và nâng cao hơn.

## **Chương 5: Lập trình VBA căn bản**

Chương này cung cấp những khái niệm căn bản về môi trường, cách làm việc và làm việc trên những bài toán đơn giản sử dụng ngôn ngữ lập trình VBA. Đây là cơ sở để tiếp cận tốt hơn trong chương tiếp theo- lập trình CSDL.

## **Chương 6: Lập trình CSDL**

Trình bày những khái niệm, nguyên tắc, công cụ và những kỹ năng rất căn bản về lập trình CSDL nói chung cũng như lập trình CSDL Access bằng VBA nói riêng. Kết thúc chương này có thể xử lý được CSDL rất đa dạng, thực tế và hoàn toàn có thể dùng Access để phát triển các CSDL thành những sản phẩm phần mềm hoàn chỉnh.

## **Chương 7: Menu & Toolbar**

Cung cấp những kiến thức, những kỹ năng cuối cùng để có thể hoàn thiện một CSDL Access thành phần mềm đóng gói, hoàn chỉnh.

Cuối cùng, chúng tôi xin chân thành cảm ơn sự quan tâm và tạo điều kiện của Sở GD&ĐT Hà Nội; các đồng nghiệp đã tận tình giúp đỡ, đóng góp ý kiến xây dựng để giáo trình được hoàn thiện như bây giờ. Tuy nhiên rất khó tránh khỏi những thiếu sót từ khách quan đến chủ quan. Chúng tôi rất mong nhận được sự góp ý, phê bình để giáo trình dần được hoàn thiện hơn, phục vụ tốt cho việc dạy-học trong nhà trường và các bạn đọc.

*Hà nội, ngày 3 tháng 2 năm 2005*

**Tác giả**

Nguyễn Sơn Hải

Trung tâm Tin học - Bộ Giáo dục và Đào tạo

ĐT: 0913.382.686.

Email: [nshai@moet.edu.vn](mailto:nshai@moet.edu.vn)

*\* Đề nghị ghi rõ xuất xứ khi sử dụng bất kỳ nội dung nào của giáo trình này!*

# BÀI MỞ ĐẦU

Bài mở đầu sẽ trình bày về môi trường làm việc, cũng như cách vào ra phần mềm Access, đó là:

- Giới thiệu Access2000;
- Cách khởi động;
- Tạo tệp Access mới;
- Môi trường làm việc;
- Mở tệp Access đã tồn tại;
- Thoát khỏi Access.



## 1. Giới thiệu Access 2000

Từ cuối những năm 80, hãng Microsoft đã cho ra đời hệ điều hành Windows, đánh dấu một bước ngoặt trong phát triển các ứng dụng phần mềm trên nền Windows (giao diện GUI- Graphical User Interface). Một trong những ứng dụng nổi bật nhất đi kèm lúc đó là bộ phần mềm tin học văn phòng Microsoft Office. Từ đó đến nay, bộ phần mềm này vẫn chiếm thị phần số 1 trên thế giới trong lĩnh vực tin học văn phòng.

Ngoài những ứng dụng về văn phòng quen thuộc phải kể đến như: MS Word - để soạn thảo tài liệu; MS Excel - bảng tính điện tử; MS Powerpoint - để trình chiếu báo cáo; .. còn phải kể đến phần mềm quản trị cơ sở dữ liệu rất nổi tiếng đi kèm: MS Access. Đến nay phiên bản mới nhất là AccessXP. Toàn bộ nội dung giáo trình này, chúng tôi giới thiệu trên Access2000. Về cơ bản, các phiên bản từ Access97 trở lại đây cách sử dụng gần giống nhau. Mỗi phiên bản chỉ khác một số tính năng đặc biệt và một chút về giao diện. Do đó, khi học Access2000, bạn đọc luôn có được những kiến thức cần thiết nhất để tiếp thu những phiên bản Access mới sau này cũng như để nhìn nhận và sử dụng tốt các phiên bản cũ hơn. Chúng tôi khuyên bạn đọc nên sử dụng Access97 trở lên.

Nếu là người chưa từng biết gì về Access, bạn sẽ có một câu hỏi:

### **Access làm được gì, và những ứng dụng của nó trong thực tế?**

- Access là một Hệ quản trị cơ sở dữ liệu quan hệ (RDMS- Relational Database Management System), rất phù hợp cho các bài toán quản lý vừa và nhỏ. Hiệu năng cao và đặc biệt dễ sử dụng- bởi lẽ giao diện sử dụng phần mềm này gần giống hết một số phần mềm khác trong bộ MS Office quen thuộc như : MS Word, MS Excel;
- Hơn nữa, Access còn cung cấp hệ thống công cụ phát triển khá mạnh đi kèm (Development Tools). Công cụ này sẽ giúp các nhà phát triển phần mềm đơn giản trong việc xây dựng trọn gói các dự án phần mềm quản lý qui mô vừa

và nhỏ. Đặc biệt những ai muốn học phát triển phần mềm thì đây là cách dễ học nhất, nhanh nhất giải quyết bài toán này.

- Đến đây có thể khẳng định được 2 ứng dụng chính của Access là :
  1. Dùng để xây dựng hệ cơ sở dữ liệu (chỉ là phần cơ sở dữ liệu, còn phần phát triển thành phần mềm có thể dùng các công cụ khác để làm như : Visual Basic, Visual C, Delphi, .NET,..)
  2. Có thể dùng để xây dựng trọn gói những phần mềm quản lý qui mô vừa và nhỏ.

### **Access2000 có gì mới so với các phiên bản cũ (Access97)?**


- Phải thừa nhận giao diện sử dụng đã có những tiến bộ vượt bậc. Các thao tác sử dụng ít đi, đơn giản hơn và giao diện rất thân thiện;
- Công nghệ truy cập dữ liệu ADO – ActiveX Data Objects hoàn toàn có thể thay thế công nghệ trước đây sử dụng trong Access là DAO – Data Access Objects bởi nhiều lý do cả về cách thức sử dụng lẫn những tối ưu về kỹ thuật. Chúng tôi sẽ giới thiệu kỹ công nghệ này trong Phần 2 cuốn giáo trình;
- Ngôn ngữ lập trình VBA được cải tiến, đặc biệt bản MS Access 2000 Developer còn cung cấp công cụ để đóng gói dự án Access (tức là tạo bộ cài đặt mà khi sử dụng không cần phải cài đặt Access lên máy tính)
- Có khả năng tạo các ứng dụng truy cập cơ sở dữ liệu thông qua giao diện web (web-base). Điều này chưa hề có trong các phiên bản trước đây. Tuy nhiên, khả năng này vẫn còn một số giới hạn, chưa thể thực sự mạnh như các công cụ chuyên nghiệp khác như : ASP, PHP, .NET, ...

Để có thể sử dụng Access, máy tính phải được cài đặt phần mềm này thông qua bộ Microsoft Office 2000. Các bước cài đặt xin tham khảo tài liệu hướng dẫn sử dụng đi kèm đĩa CD phần mềm.

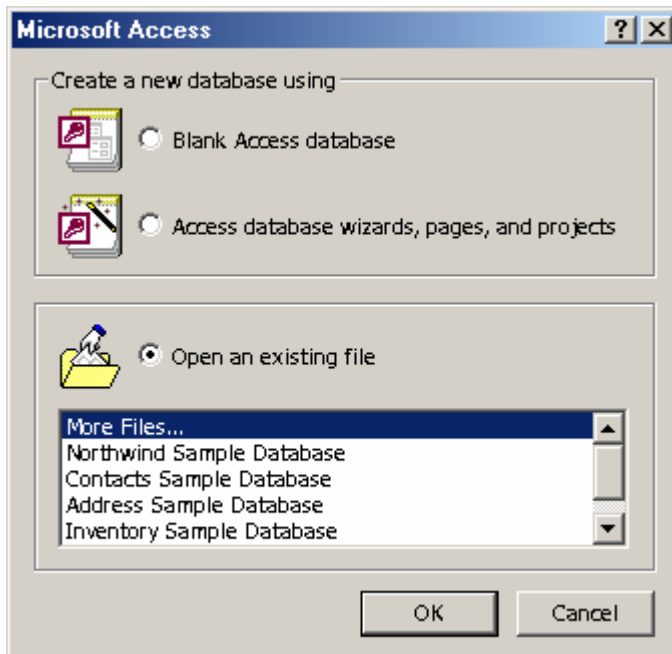
## **2. Khởi động**

Có khởi động Access theo nhiều cách:

- Mở lệnh **Start | Programs | Microsoft Access** của Window;

- Hoặc có thể nhấn kép chuột lên tên tệp có phần mở rộng .MDB với biểu tượng của Microsoft Access 

Sau khi ra lệnh chạy chương trình Access, thông thường màn hình tiếp theo xuất hiện cho phép chọn cách làm việc:



- Chọn **Blank Access database** để bắt đầu tạo một tệp Access mới (tạo mới tệp);
- Hoặc **Access database wizard, page, and project** để tạo một CSDL theo mẫu có sẵn (không trình bày trong giáo trình này);
- Hoặc **Open an existing file** để mở tệp Access đã tồn tại để làm việc tiếp.

### 3. Tạo mới tệp Access

Một dự án Access (Access project) là một hoặc nhiều tệp Access nhằm giải quyết một công việc lớn nào đó. Khuôn khổ giáo trình này chỉ trình bày những dự án có 1 tệp.

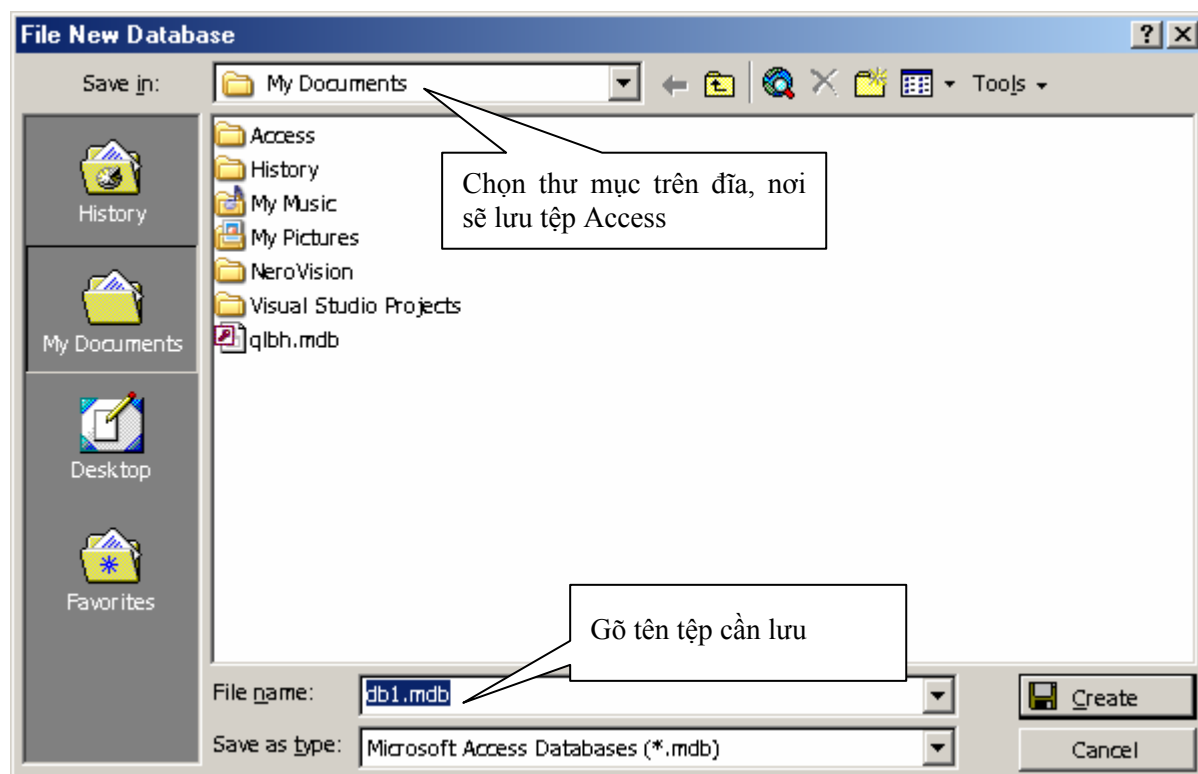
Tệp Access có phần mở rộng \*.MDB (ngoài ra những tệp có phần mở rộng \*.DBE cũng mở được nhưng chỉ có thể thực thi chứ không chỉnh sửa được cấu trúc). Có 5 thành phần chính trên một tệp:


- **Tables** – nơi chứa toàn bộ các bảng dữ liệu;

- **Queries** – nơi chứa toàn bộ các truy vấn dữ liệu đã được thiết kế;
- **Forms** – nơi chứa các mẫu giao diện phần mềm;
- **Reports** – nơi chứa các mẫu báo cáo đã được thiết kế;
- **Macro** – nơi chứa các Macro lệnh phục vụ dự án;
- **Modules** – nơi chứa các khai báo, các thư viện chương trình con phục vụ dự án.

Thông thường mỗi tệp Access cần phải làm việc trên tất cả các thành phần trên.

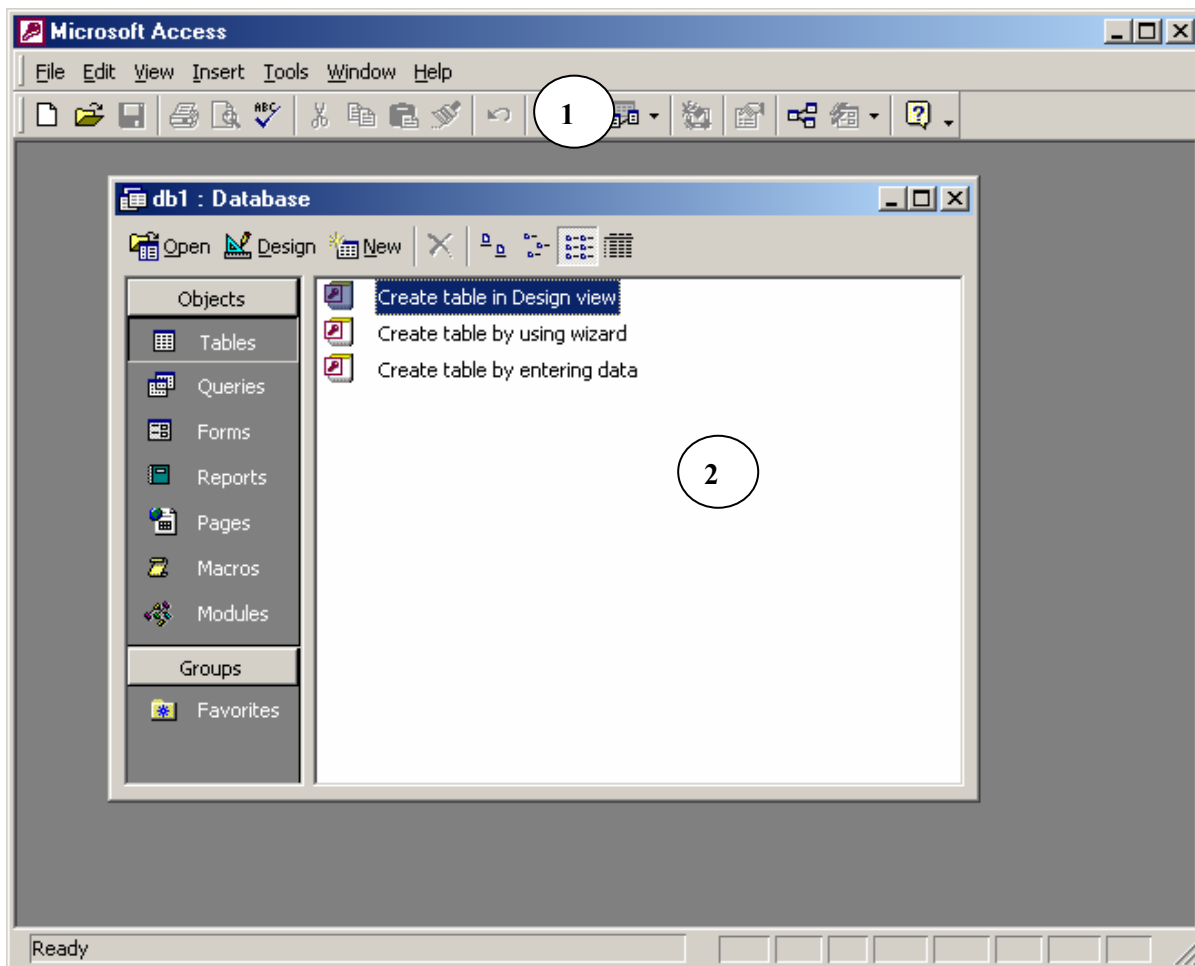
Để bắt đầu tạo một tệp mới, chọn mục **Blank Access database** ở cửa sổ mục 2, tiếp theo nhấn **OK**, hộp thoại sau xuất hiện yêu cầu chọn nơi (thư mục) lưu trữ và đặt tên tệp Access:



Hãy chọn nơi lưu trữ ở hộp **Save in**; gõ vào tên tệp ở mục **File Name**; nhấn  để ghi lại thiết lập. Đến đây đã sẵn sàng sử dụng tệp Access vừa khai báo.

## 4. Môi trường làm việc


Sau khi một tệp Access được mở, môi trường làm việc trên Access xuất hiện với những thành phần như sau:

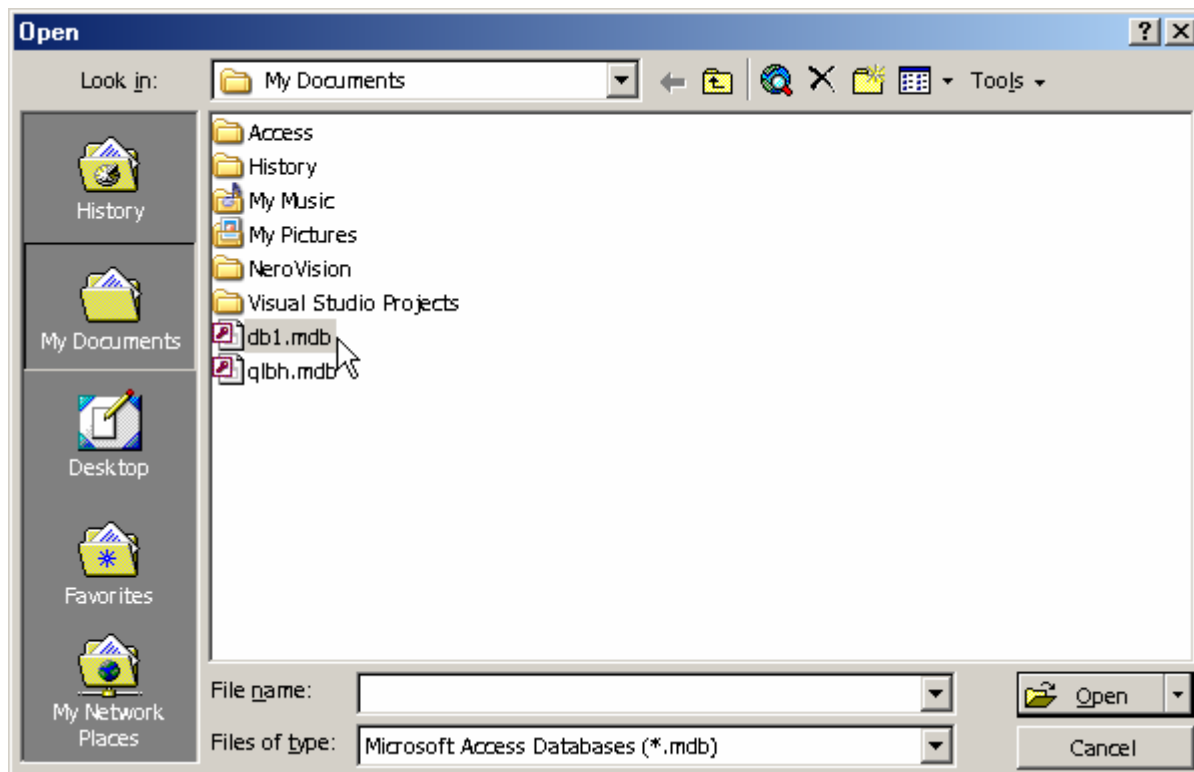


- (1) - hệ thống thực đơn (menu) và các thanh công cụ (Toolbar)- nơi thực hiện các lệnh khi cần;
- (2) - Cửa sổ tệp Access đang làm việc bao gồm 7 phần chính: Tables, Queries, Forms, Reports, Pages, Macros và Modules. Chúng ta sẽ lần lượt học cách làm việc trên từng phần này trong các chương tiếp theo.

## 5. Mở tệp đã tồn tại

Mỗi tệp Access phải tạo mới duy nhất một lần, được mở ra làm việc và ghi lại trong những lần tiếp theo. Để mở một tệp Access đã tồn tại để làm việc, làm như sau:

**Bước 1:** Từ môi trường Access gọi thực đơn: **File | Open** (hoặc nhấn nút **Open**  trên thanh công cụ), hộp thoại **Open** xuất hiện:




**Bước 2:** Tìm đến tệp Access cần mở trên hộp thoại Open bằng cách:

- Tìm đến thư mục – nơi chứa tệp Access cần mở ở hộp **Look in**;
- Tiếp theo chọn tệp Access cần mở trên danh sách và nhấn nút **Open** hoặc Enter.

Đến đây cửa màn hình làm việc Access với tệp vừa mở xuất hiện để tiếp tục làm việc.

## 6. Thoát khỏi Access

Khi không làm việc với Access, hãy ra lệnh thoát khỏi Access bằng một trong các cách:

- Mở thực đơn **File | Exit**;
- Nhấn tổ hợp phím nóng **Alt + F4**;
- Hoặc sử dụng nút Close  trên cửa sổ Access đang mở.

# CHƯƠNG 1

## XÂY DỰNG CƠ SỞ DỮ LIỆU

Xây dựng CSDL là công việc quan trọng đầu tiên trong toàn bộ qui trình phát triển một ứng dụng trên Access. Một CSDL được thiết kế và xây dựng tốt sẽ là những thuận lợi, những tự tin đầu tiên để bước vào một qui trình phát triển ứng dụng; nhưng trái lại sẽ là một thảm họa cho dự án đang phát triển: sẽ thường xuyên gặp phải những khó khăn để phải chỉnh sửa lại CSDL và tồi tệ hơn, dự án có thể phải thực hiện lại từ đầu do việc thiết kế CSDL quá kém.

Trong chương này sẽ trình bày các khái niệm cũng như các kỹ năng để học viên có thể tiếp cận và xây dựng được các hệ CSDL trên Access, cụ thể là:

- Tạo cấu trúc các bảng dữ liệu;
- Thiết lập thuộc tính LookUp;
- Thiết lập quan hệ và các thuộc tính đảm bảo toàn vẹn dữ liệu;
- Nhập dữ liệu cho CSDL;
- Đặc biệt là qui trình xây dựng một CSDL Access.

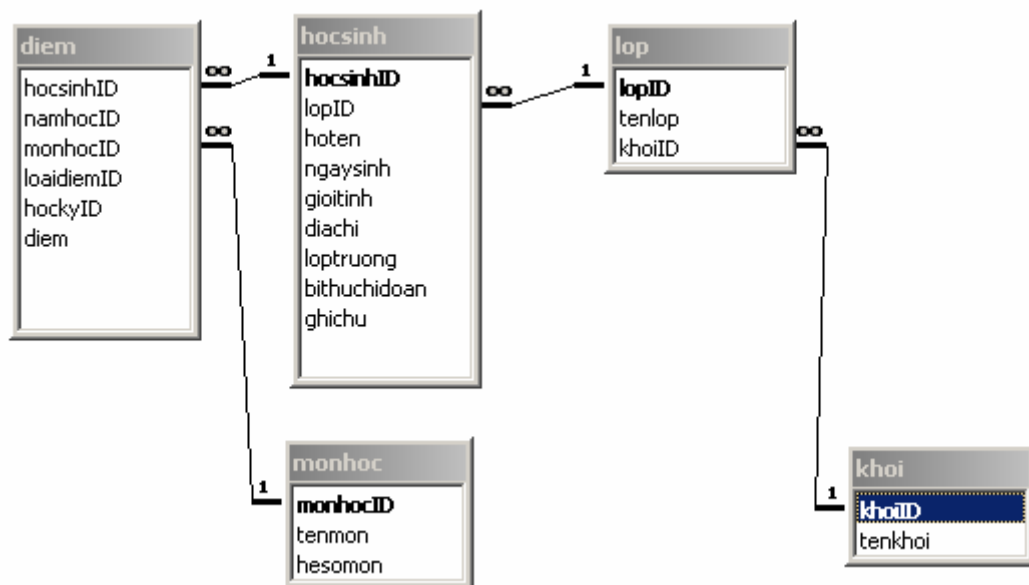
## 1. Các khái niệm về CSDL Access

### 1.1 CSDL Access

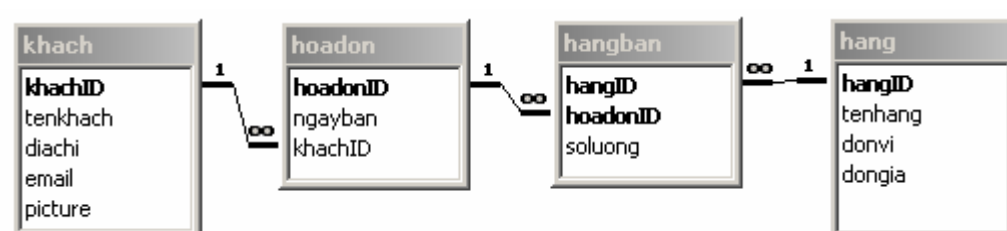
CSDL Access là một đối tượng bao gồm tập hợp các bảng dữ liệu, các kết nối giữa các bảng được thiết kế một cách phù hợp để phục vụ lưu trữ dữ liệu cho một ứng dụng quản lý dữ liệu nào đó.

#### Ví dụ:

CSDL Quản lý học sinh bao gồm tập hợp các bảng dữ liệu: *HOCSINH*, *LOP*, *KHOI*, *MONHOC*, *DIEM* được kết nối với nhau một cách phù hợp phục vụ lưu trữ dữ liệu cho ứng dụng quản lý học sinh một trường học. Toàn bộ cấu trúc CSDL quản lý học sinh trên Access được mô tả như sau:

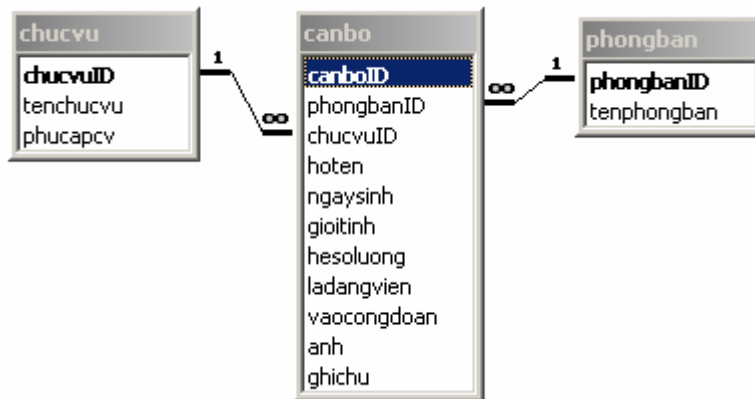


CSDL Quản lý bán hàng bao gồm tập hợp các bảng dữ liệu: *HANG*, *KHACH*, *HOADON*, *HANGBAN* được kết nối với nhau một cách phù hợp, phục vụ ứng dụng quản lý việc bán hàng tại một cửa hàng. Sơ đồ cấu trúc CSDL này như sau:

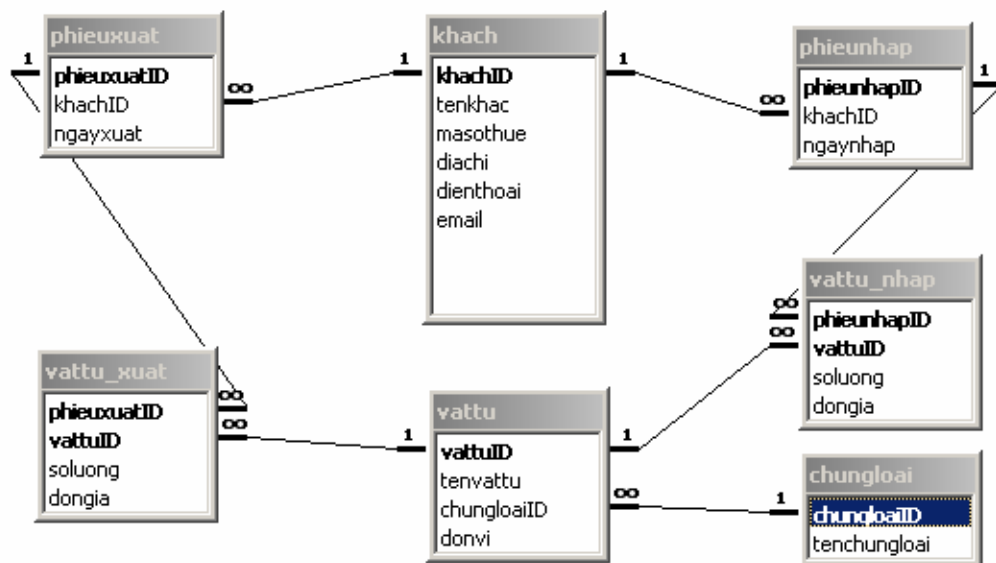




CSDL Quản lý lương cán bộ, bao gồm các bảng: *PHONGBAN*, *CHUCVU* và *CANBO* được kết nối với nhau phục vụ lưu trữ dữ liệu cho ứng dụng quản lý lương cán bộ một cơ quan. Cấu trúc CSDL này được biểu diễn như sau:



CSDL Quản lý việc nhập-xuất vật tư một cửa hàng. Bao gồm các bảng: *VATTU*, *KHACH*, *PHIEUNHAP*, *PHIEUXUAT*, *VATTU\_NHAP*, *VATTU\_XUAT* được biểu diễn như sau:



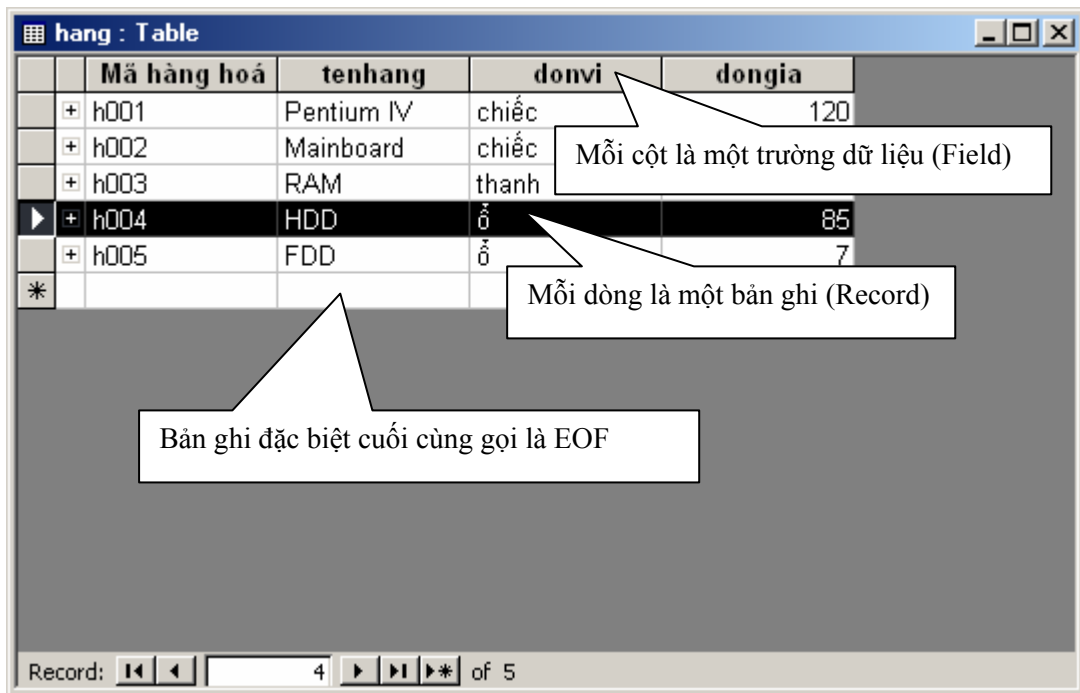
## 1.2 Bảng dữ liệu

Bảng dữ liệu (Tables) là một phần quan trọng nhất của CSDL; Là nơi lưu trữ những dữ liệu tác nghiệp cho ứng dụng. Một CSDL có thể có rất nhiều bảng, các bảng phải được thiết kế sao cho có thể lưu trữ được đầy đủ dữ liệu cần thiết, đảm bảo giảm tối đa tình trạng gây dư thừa dữ liệu (dư thừa dữ liệu được hiểu đơn giản là tình trạng lưu trữ những dữ liệu không cần thiết trên một số bảng. Tác hại của hiện tượng này sẽ gây: sai lệch dữ liệu tác nghiệp và làm tăng dung lượng dữ

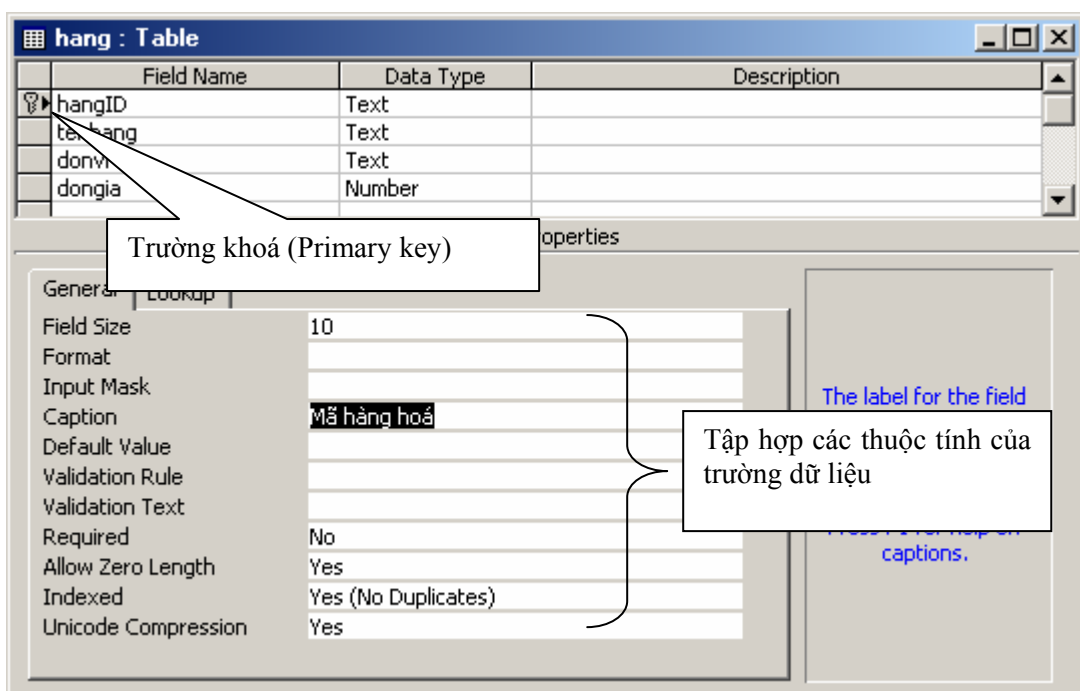
liệu không cần thiết); giảm tối đa dung lượng CSDL có thể, đồng thời tạo môi trường làm việc thuận lợi cho việc phát triển ứng dụng trong các bước tiếp theo.

Một bảng dữ liệu trên Access bao gồm các thành phần: Tên bảng, các trường dữ liệu, trường khoá, tập hợp các thuộc tính cần thiết cho mỗi trường dữ liệu và tập hợp các bản ghi.

Mô tả một bảng dữ liệu trong trạng thái Datasheet (nhập, xem, sử dữ liệu):



Mô tả một bảng dữ liệu trong trạng thái Design view (đang thiết kế cấu trúc):



## Tên bảng

Mỗi bảng có một tên gọi. Tên bảng thường được đặt sau khi tạo xong cấu trúc của bảng, tuy nhiên cũng có thể đổi lại tên bảng trên cửa sổ Database như đổi tên tệp dữ liệu trên cửa sổ Windows Explorer.

Không nên sử dụng dấu cách (Space), các ký tự đặc biệt hoặc chữ tiếng Việt có dấu trong tên bảng.

## Trường dữ liệu (Field)

Mỗi cột dữ liệu của bảng sẽ tương ứng với một trường dữ liệu. Mỗi trường dữ liệu sẽ có một tên gọi và tập hợp các thuộc tính miêu tả trường dữ liệu đó ví dụ như: kiểu dữ liệu, trường khoá, độ lớn, định dạng, ..

Mỗi trường dữ liệu phải được định kiểu dữ liệu. Trong Access, trường dữ liệu có thể nhận một trong các kiểu dữ liệu sau:

TT	Kiểu dữ liệu	Độ lớn	Lưu trữ..
1	Number	Tùy thuộc kiểu cụ thể	Số: số thực, số nguyên theo nhiều kiểu
2	Autonumber	4 bytes	Số nguyên tự động được đánh số.
3	Text	tùy thuộc độ dài xâu	Xâu ký tự
4	Yes/No	1 bytes	Kiểu logic
5	Date/Time	8 bytes	Lưu trữ ngày, giờ
6	Currency	Sing	Lưu trữ dữ liệu kèm ký hiệu tiền tệ
7	Memo	tùy thuộc giá trị	kiểu ghi nhớ
8	Hyperlink	tùy thuộc độ dài xâu	Lưu trữ các siêu liên kết (hyperlink)
9	OLE	tùy thuộc dữ liệu	Âm thanh, hình ảnh, đồ hoạ, ... (Objects)

Không nên sử dụng dấu cách (Space), các ký tự đặc biệt hoặc chữ tiếng Việt có dấu trong tên trường.

### **Bản ghi (Record)**

Mỗi dòng dữ liệu của bảng được gọi một bản ghi. Mỗi bảng có một con trỏ bản ghi. Con trỏ bản ghi đang nằm ở bản ghi nào, người dùng có thể sửa được dữ liệu bản ghi đó. Đặc biệt, bản ghi trắng cuối cùng của mỗi bảng được gọi EOF.

### **Trường khoá (Primary key)**

Trường khoá có tác dụng phân biệt giá trị các bản ghi trong cùng một bảng với nhau. Trường khoá có thể chỉ 01 trường, cũng có thể được tạo từ tập hợp nhiều trường (gọi bộ trường khoá).

### **Ví dụ:**

(1)- bảng *THISINH* của CSDL thi tuyển sinh, trường khoá là *SoBaoDanh*. Vì mỗi thí sinh có thể nhiều trường có giá trị hết nhau, nhưng *SoBaoDanh* thì duy nhất.

(2)- bảng *CANBO* trường *MaCanBo* sẽ là trường khóa vì không thể tồn tại 2 cán bộ nào trong bảng này trùng *MaCanBo*

(3)- bảng *HANGBAN* của CSDL Quản lý bán hàng, 2 trường *hangID* và *hoadonID* là một bộ trường khoá. Vì không thể trên một hoá đơn bán hàng nào có bán một mặt hàng nào đó ghi lặp lại 2 lần.

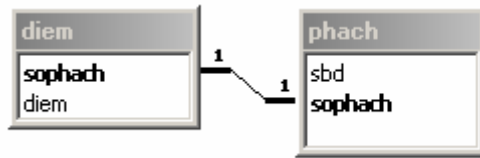
## **1.3 Liên kết các bảng dữ liệu**

Liên kết các bảng dữ liệu là một kỹ thuật trong thiết kế CSDL quan hệ. Chúng là mối liên kết giữa 2 bảng với nhau theo thiết kế cho trước để đảm bảo được mục đích lưu trữ dữ liệu cho ứng dụng.

Trong Access tồn tại 2 kiểu liên kết: liên kết 1-1 và liên kết 1-n (một-nhiều)

**Liên kết 1-1** là: mỗi bản ghi của bảng này sẽ liên kết với duy nhất tới một bản ghi của bảng kia và ngược lại;

Ví dụ liên kết 1-1:



Mô tả dữ liệu 2 bảng này như sau:

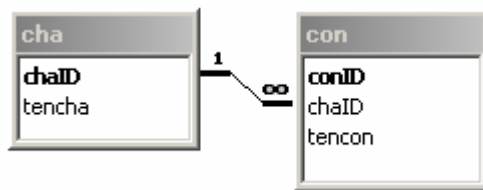
	sbd	sophach
	01	12
	02	02
	02	23
	03	14
	04	15
	05	32
	06	43
	07	08
▶	08	06
	09	10
*		

	sophach	diem
	12	5
	02	44
	23	0
	14	7
	15	7
	32	6
	43	9
	08	4
	06	7
	10	6
▶		0

**Liên kết 1-n** là: mỗi trường của bảng 1 sẽ có thể liên kết với một hoặc nhiều bản ghi của bảng nhiều (n). Ngược lại, mỗi bản ghi của bảng nhiều sẽ liên kết tới duy nhất 1 trường của bảng 1.

Ví dụ liên kết 1-n:



Có thể tham khảo mỗi cha có thể có nhiều con qua 2 bảng sau:

	chaID	tencha
▶	+ 01	Nguyễn Đức Sơn
	+ 02	Phan Anh Quân
	+ 03	Lê Hoàng
*		

	chaID	conID	tencon
▶	01	001	Nguyễn Đức Thìn
	01	002	Nguyễn Thị Thanh
	02	003	Phan Quang
	02	004	Phan Huy Hoàng
	02	005	Phan Thùy Anh
	03	006	Lê Văn Tám
*			

Hoặc xem theo một kiểu khác:

chaID	tencha
01	Nguyễn Đức Sơn
02	Phan Anh Quân
03	Lê Hoàng

conID	tencon
001	Nguyễn Đức Thìn
002	Nguyễn Thị Thanh
003	Phan Quang
004	Phan Huy Hoàng
005	Phan Thủy Anh
006	Lê Văn Tám

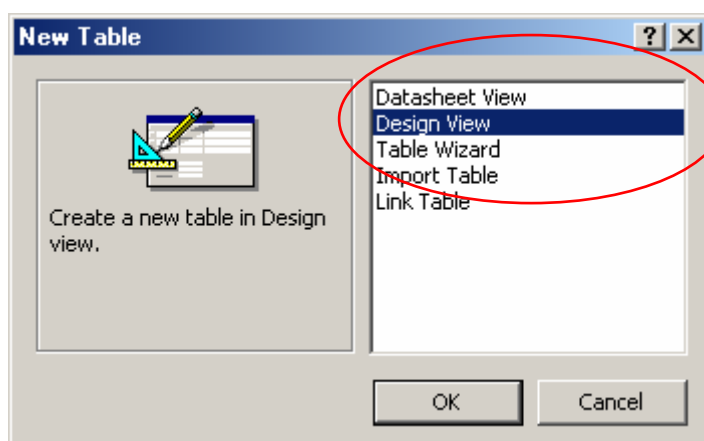
## 2. Xây dựng cấu trúc bảng

Mục này hướng dẫn cách thiết kế cấu trúc một bảng dữ liệu trên CSDL Access. Để có thể làm tốt được công việc này, đề nghị học viên nên tuân thủ theo thứ tự các bước sẽ liệt kê dưới đây.

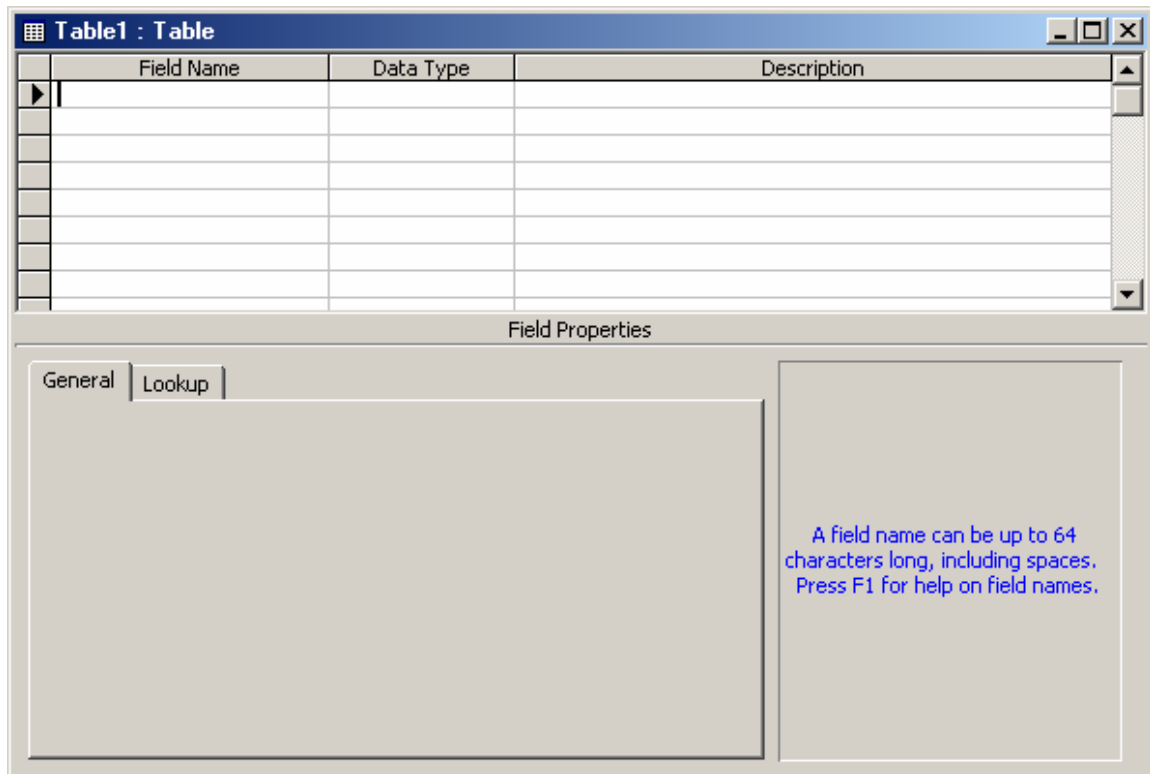
Minh họa này hướng dẫn cách tạo cấu trúc bảng **HANG** bao gồm các trường *hangID*, *tenhang*, *donvi*, *dongia*.

**Bước 1:** Khởi động trình thiết kế cấu trúc bảng ở chế độ Design View

Ở thẻ Tables, nhấn nút **New**, chọn **Design View**, nhấn **OK**



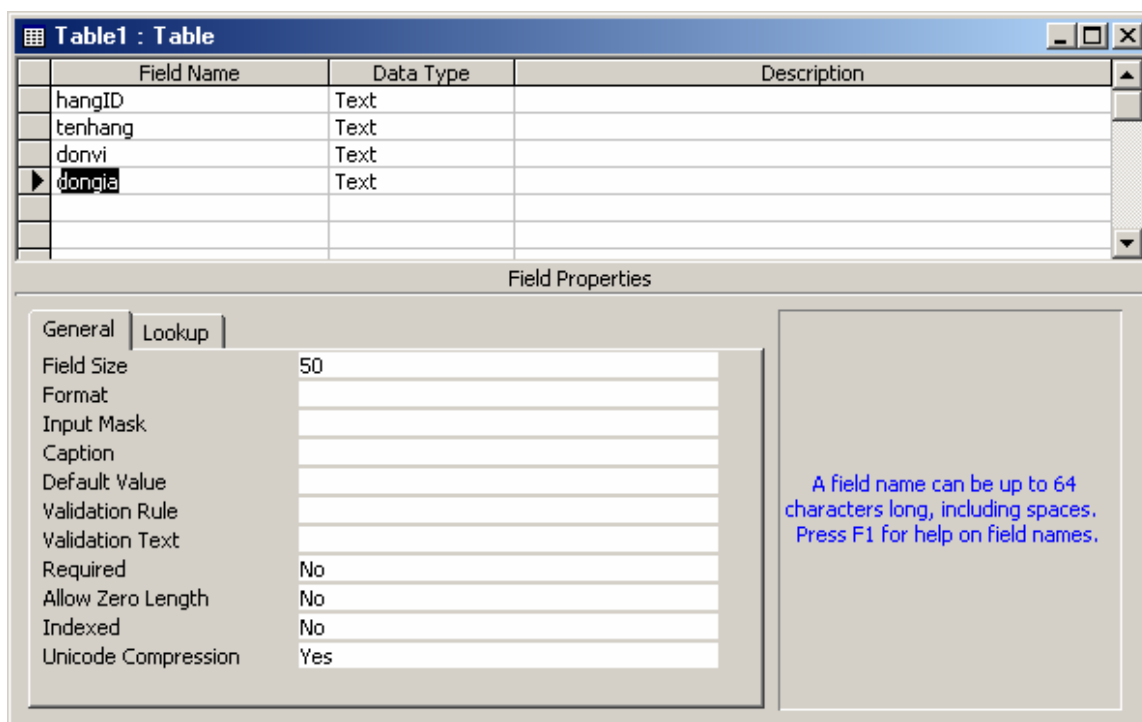
Hoặc nhấn  **Create table in Design view** trên thẻ Tables. Hộp thoại thiết kế cấu trúc một bảng xuất hiện:



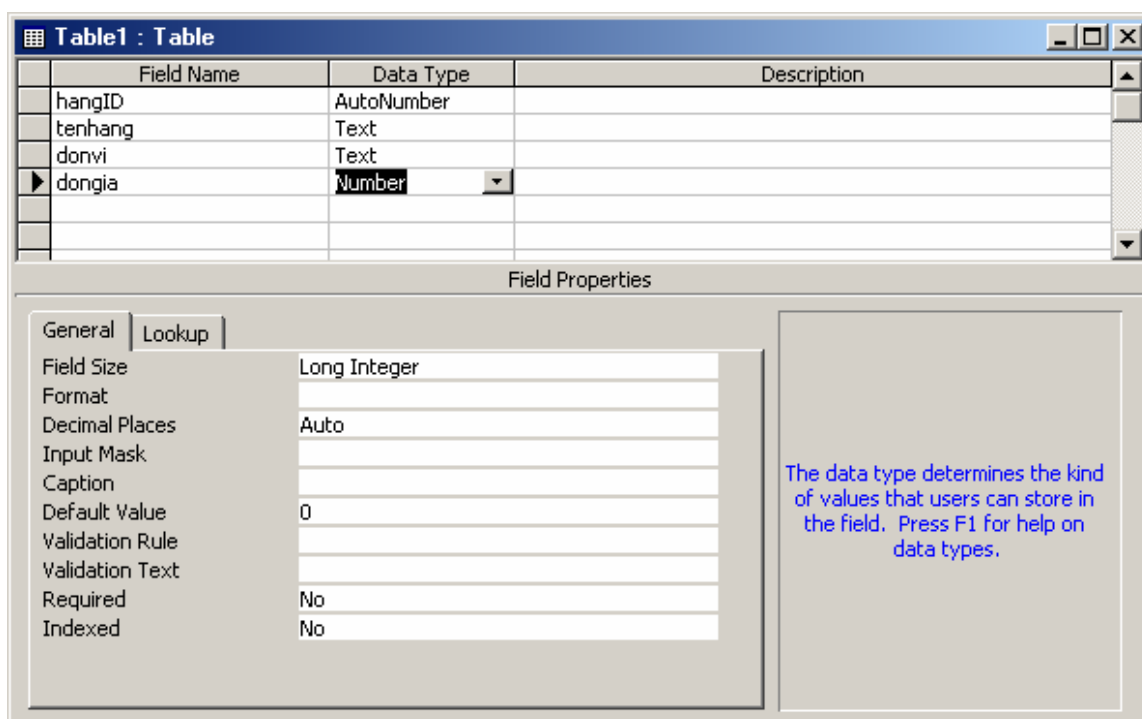
**Bước 2:** Khai báo danh sách tên các trường của bảng: bằng cách gõ danh sách tên các trường lên cột **Field Name** của cửa sổ thiết kế.

**Chú ý:** tên trường không nên chứa dấu cách (space), chữ tiếng Việt có dấu.

Sau khi gõ vào danh sách tên các trường của bảng HANG, hộp thoại thiết kế sẽ có dạng:



**Bước 3:** Khai báo kiểu dữ liệu cho các trường của bảng, bằng cách chọn kiểu dữ liệu cho từng trường ở cột **Data Type** tương ứng. Sau khi chọn xong kiểu dữ liệu cho các trường, hộp thoại thiết kế sẽ có dạng:




Để chọn kiểu dữ liệu, có thể dùng chuột chọn kiểu dữ liệu từ hộp thả; mặt khác cũng có thể nhấn ký tự đầu tiên của kiểu dữ liệu cần chọn mỗi khi định vị đến ô Data Type cần làm việc. Ví dụ:




Nhấn	Sẽ được kiểu
A	<u>A</u> utonomous
N	<u>N</u> umber
T	<u>T</u> ext
Y	<u>Y</u> es/No
D	<u>D</u> ate/Time

Nhấn	Sẽ được kiểu
M	<u>M</u> emo
C	<u>C</u> urrency
H	<u>H</u> yperlink
O	<u>O</u> LE


**Bước 4:** Thiết lập trường khoá cho bảng (những bảng không có trường khoá có thể bỏ qua bước này).

- Chọn các trường muốn thiết lập khoá bằng cách: dùng chuột kết hợp giữ phím **Shift** đánh dấu đầu dòng các trường muốn thiết lập khoá;
- Mở thực đơn **Edit | Primary key** để thiết lập thuộc tính khoá cho các trường vừa chọn. Cũng có thể ra lệnh này bằng cách nhấn nút *Primary key*  trên thanh công cụ.

Sau khi thiết lập khoá, những trường khoá sẽ có biểu tượng như sau:

	Field Name	Data Type
	hangID	Text
	tenhang	Text
	donvi	Text
	dongia	Number

Biểu tượng của trường khoá

**Bước 5:** Lưu lại cấu trúc bảng. Nhấn tổ hợp phím **Alt + S** hoặc nhấn nút  Save trên thanh công cụ, hộp thoại yêu cầu ghi tên cho bảng xuất hiện:

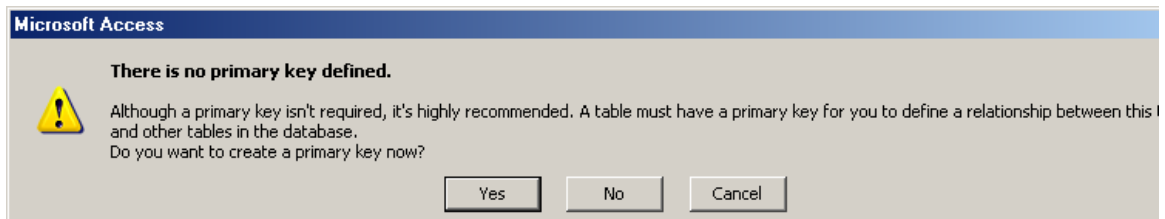


The image shows a 'Save As' dialog box with a text input field containing 'HANG'. The text 'HANG' is highlighted with a red oval. There are 'OK' and 'Cancel' buttons to the right of the input field.

Hãy gõ tên bảng và nhấn **OK**.

#### Đặc biệt:

- Với những bảng không thiết lập trường khoá, trong quá trình ghi lại cấu trúc bảng, máy tính sẽ hỏi:



Nhấn **Yes**- máy tính sẽ tạo thêm một trường mới có tên ID và thiết lập trường này làm khoá. Nếu không muốn như vậy hãy nhấn **No**; nhấn **Cancel** để huỷ lệnh cất.

- Tên bảng không nên chứa dấu cách, các ký tự đặc biệt khác hoặc chữ Việt có dấu.

Mỗi trường dữ liệu được khai báo trong cửa sổ trên đều có thể thiết lập được rất nhiều các thuộc tính tùy thuộc kiểu dữ liệu trường đó đã nhận. Các thuộc tính này có thể thiết lập tại phần *Tập hợp các thuộc tính của các trường* như đã trình bày ở trên. Dưới đây là danh sách một số các thuộc tính hay được sử dụng.

#### Thuộc tính **Field size**

Để thiết lập kích thước dữ liệu. Chỉ áp dụng cho các trường có kiểu dữ liệu Number và Text.

Đối với các trường kiểu số, **Field size** cho biết trường đó nhận giá trị loại số nào.

TT	Loại số	Độ lớn
1	Byte	số nguyên 2 byte
2	Integer	số nguyên 4 byte
3	Long Integer	số nguyên 6 byte
4	Single	số thực 8 byte
5	Double	số thực 12 byte
6	Decimal	số thực 16 byte

Đối với các trường kiểu Text, thuộc tính này cho biết chiều dài tối đa của xâu ký tự. Ví dụ: Với trường *Hoten* thì Field size khoảng 30.

#### Thuộc tính **Format**

Để thiết lập định dạng dữ liệu khi hiển thị. Trường này áp dụng cho hầu hết các kiểu dữ liệu trừ ra kiểu: Memo, OLE, Yes/No

#### Thuộc tính **Input Mark**

Thiết lập mặt nạ nhập dữ liệu cho các trường. Kiểu này có thể áp dụng cho các loại trường kiểu Text, Number, Datetime, Currency.

#### Thuộc tính **Default Value**

Để thiết lập giá trị ngầm định cho trường mỗi khi ra lệnh thêm mới một bản ghi. Ví dụ: Trường số lượng mỗi khi thêm một bản ghi mới, giá trị trường này tự động là 1. Khi đó phải thiết lập thuộc tính **Default Value** của trường này là **1**.

#### Thuộc tính **Caption**

Thiết lập tiêu đề cột mà trường đó hiển thị. Tên trường không nên chứa dấu cách và chữ Việt có dấu, nhưng *Caption* của các trường thì nên gõ bằng tiếng Việt có dấu sao cho dễ đọc và nhận biết. Đặc biệt giá trị thuộc tính Caption nếu có sẽ được sử dụng làm tiêu đề cho các trường tương ứng mỗi khi sử dụng công cụ Form Wizard hay Report Wizard sau này- sẽ rất tiện lợi.

#### Thuộc tính **Validation Rule**

Thiết lập điều kiện kiểm tra tính đúng đắn của dữ liệu khi được nhập vào.

Ví dụ: trường *NgaySinh* của học sinh phải nhập vào những ngày  $\geq 1/1/1980$  chẳng hạn. Khi đó ở thuộc tính *Validation Rule* của trường *Ngaysinh* hãy gõ vào  **$\geq \#1/1/1980\#$**

#### Thuộc tính **Required**

Để yêu cầu phải nhập dữ liệu cho trường này (nếu thiết lập **Yes**) khi bắt đầu một bản ghi mới hoặc không nếu thiết lập **No**.

Còn một số các thuộc tính khác xin mời tham khảo qua phần Help của Access.

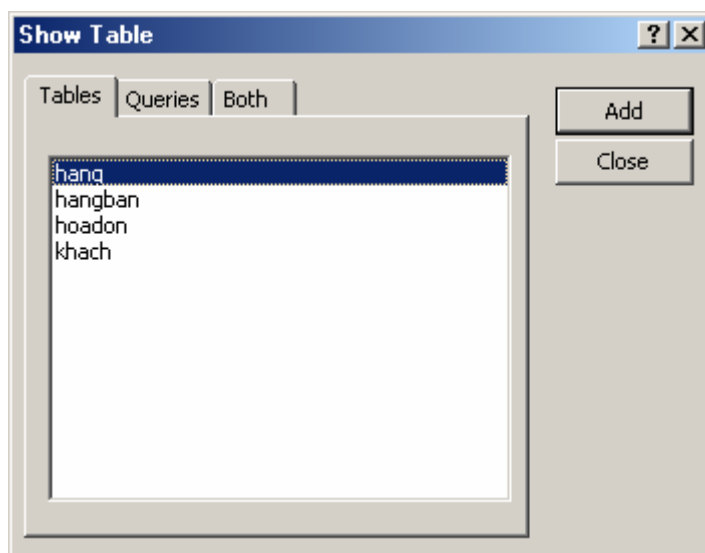
### 3. Thiết lập quan hệ

Một bước quan trọng trong xây dựng CSDL Access là thiết lập quan hệ các bảng trong CSDL. Làm được điều này bạn sẽ gặp được rất nhiều thuận lợi trong quá trình sử dụng các trình Wizard và Design View trong Access sau này.

Dưới đây sẽ là hướng dẫn cách thiết lập quan hệ cho một cặp bảng. Tương tự bạn phải thiết lập toàn bộ các quan hệ có thể trên CSDL.

**Bước 1:** Mở cửa sổ thiết lập quan hệ bởi thực đơn: **Tools | Relationship..**

**Bước 2:** Đưa các bảng (Tables) tham gia thiết lập quan hệ thông qua hộp thoại Show Tables (nếu chưa thấy hộp thoại này dùng thực đơn **Relationship | Show table**):

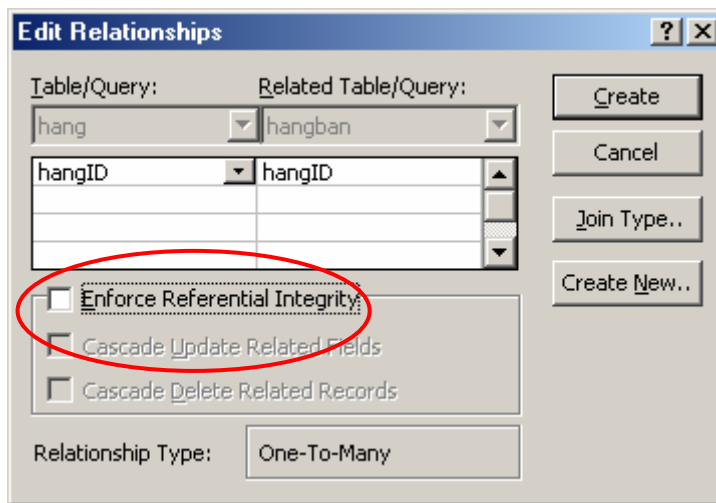


Cách đưa các bảng lên cửa sổ thiết lập quan hệ (Database) như sau:

- Chọn bảng cần tham gia thiết lập quan hệ (thường thì chọn tất cả) ;
- Nhấn nút **Add**;
- Chọn xong toàn bộ nhấn **Close** để đóng cửa sổ.

**Bước 3:** Thực hiện tạo kết nối giữa từng cặp bảng theo thiết kế, cách làm như sau:

Dùng chuột kéo (Drag) trường cần liên kết của bảng này (ví dụ trường hangID của bảng HANG) thả (Drop) lên trường cần liên kết đến của bảng kia (ví dụ trường hangID của bảng HANGBAN). Khi đó hộp thoại **Edit Relationships** xuất hiện:



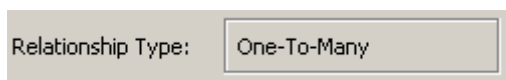
Trong trường hợp muốn thiết lập các thuộc tính đảm bảo toàn vẹn dữ liệu (Enforce Referential Integrity) cho quan hệ hãy thực hiện chọn (checked) 3 mục chọn sau:

**Enforce Referential Integrity** để đồng ý thiết lập các thuộc tính đảm bảo toàn vẹn dữ liệu;

**Cascade Update Related Fields** đảm bảo toàn vẹn dữ liệu khi xoá dữ liệu giữa 2 bảng liên quan. Khi đó, nếu một bản ghi ở bảng có quan hệ 1 bị xoá, toàn bộ các bản ghi có quan hệ với bản ghi hiện tại sẽ được tự động xoá ở bảng có quan hệ nhiều (nếu xoá 1 CHA, toàn bộ các con của cha đó sẽ tự động bị xoá khỏi bảng CON);

**Cascade Delete Related Records** đảm bảo toàn vẹn dữ liệu khi cập nhật dữ liệu giữa 2 bảng liên quan. Khi đó, nếu giá trị trường khoá liên kết ở bảng 1 bị thay đổi, toàn bộ giá trị trường khoá liên kết ở bảng nhiều cũng bị thay đổi theo.

Hộp **Relationship Type**: cho biết kiểu quan hệ giữa 2 bảng đang thiết lập:



- **One – To – One** Kiểu 1-1
- **One – To – Many** Kiểu 1-∞
- **Indeterminate** Không xác định được kiểu liên kết

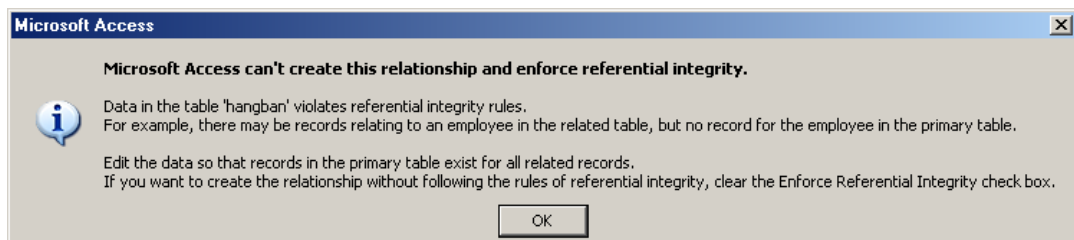
Tùy thuộc vào kiểu khoá của các trường tham gia liên kết mà Access tự xác định ra được kiểu liên kết giữa 2 bảng. Dưới đây là một số kiểu liên kết được Access tự động xác định:

TT	Bảng A	Bảng B	Kiểu liên kết
1	Khoá chính	Khoá chính	1-1
2	Khoá chính	Khoá phụ (hoặc không khoá)	1-n
3	Khoá phụ	Khoá phụ (hoặc không khoá)	Không xác định được kiểu liên kết
4	Không khoá	Không khoá	Không xác định được kiểu liên kết



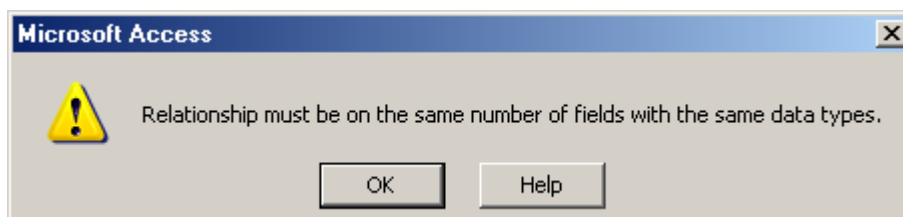
### Chú ý

- Khi hộp Relationships Type chỉ **Indeterminate** có nghĩa là quan hệ đang thiết lập không đúng về cấu trúc khoá của 2 bảng (quan hệ sai);
- Trong trường hợp thiết lập các thuộc tính đảm bảo toàn vẹn dữ liệu nếu gặp phải hộp thoại thông báo lỗi:



Lỗi này do một số dữ liệu đang tồn tại trong bảng quan hệ ∞ không thoả mãn với bảng quan hệ 1. Hiểu đơn giản là: có một số mặt hàng khai báo trong bảng HANGBAN chưa có trong danh mục hàng (bảng HANG) → như vậy là không hợp lý, không thoả mãn các điều kiện về toàn vẹn dữ liệu giữa 2 bảng này.

- Khi gặp phải lỗi sau đây khi kết nối giữa 2 bảng:



Tức là 2 trường tham gia kết nối không cùng kiểu dữ liệu. Ví dụ: một trường kiểu Number, trường kia kiểu Text; hoặc một trường kiểu Text, trường kia kiểu Date/Time - đều bị sai vì không cùng kiểu dữ liệu.

## 4. Nhập dữ liệu

Nhập dữ liệu là yêu cầu bắt buộc đối với bất kỳ một ứng dụng nào. Dữ liệu có thể được nhập vào CSDL bằng nhiều con đường khác nhau. Trong phần này sẽ trình bày cách nhập dữ liệu từ bàn phím vào trực tiếp các bảng dữ liệu trong Access.

### 4.1 Cách nhập dữ liệu

Có 2 bước để có thể nhập trực tiếp dữ liệu vào bảng:

**BƯỚC 1:** Mở bảng để nhập dữ liệu bằng cách: nhấn đúp chuột lên tên bảng cần nhập dữ liệu; hoặc chọn bảng cần nhập dữ liệu rồi nhấn nút Open;

**BƯỚC 2:** Thực hiện nhập dữ liệu vào bảng đang mở bằng bàn phím

	Mã hàng hoá	tenhang	donvi	dongia	
+	h001	Pentium IV	chiếc	120	
+	h002	Mainboard	chiếc	56	
+	h003	RAM	thanh	24	
+	h004	HDD	ổ	85	
▶	+	h005	FDD	ổ	7
*				0	

Record: 5 of 5



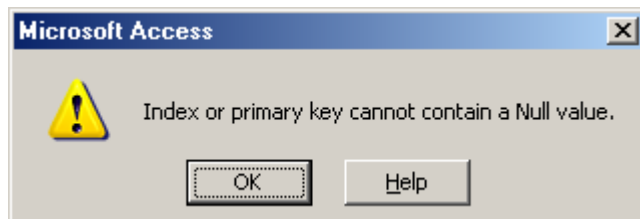
**Chú ý :** Một số lỗi có thể xảy ra khi nhập dữ liệu

#### Lỗi thứ nhất:



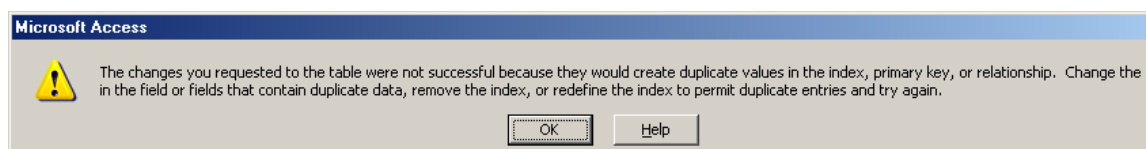
**Lỗi do:** Bạn đã nhập vào giá trị không tương thích với kiểu dữ liệu của trường đã chỉ định. Ví dụ: trường kiểu Numeric mà gõ vào chữ cái; hoặc không gõ đầy đủ các giá trị ngày, tháng, năm cho trường kiểu Date/Time,.. lỗi này sẽ xuất hiện.

**Khắc phục:** hãy nhập lại cho đúng, đủ giá trị các trường đã yêu cầu đến khi không xuất hiện thông báo lỗi.

**Lỗi thứ 2:**

**Lỗi do:** Bạn không nhập giá trị hoặc để trống giá trị trường khoá. Đã là trường khoá luôn yêu cầu phải nhập dữ liệu cho mỗi bản ghi.

**Khắc phục:** phải nhập đầy đủ giá trị cho trường khoá.

**Lỗi thứ 3:**

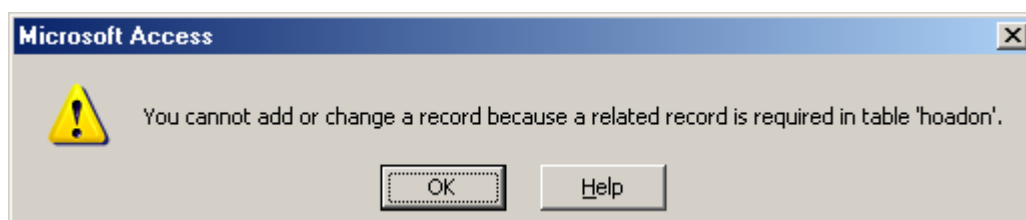
**Lỗi do:** Giá trị trường khoá trùng nhau. Giá trị trường khoá vừa nhập vào đã trùng với giá trị của một bản ghi nào đó trên bảng dữ liệu.

**Khắc phục:** nhập lại giá trị trường khoá khác sao cho vừa đúng, đủ và không bị trùng khoá.

**Lỗi thứ 4:**

**Lỗi do:** Bản ghi vừa nhập dữ liệu đã bỏ trống trường bắt buộc nhập dữ liệu (những trường được thiết lập thuộc tính *Required=Yes*)

**Khắc phục:** Phải nhập đủ dữ liệu cho các trường bắt buộc phải nhập dữ liệu.

**Lỗi thứ 5:**

**Lý do:** Lỗi do thực hiện một thao tác vi phạm các nguyên tắc đảm bảo toàn vẹn dữ liệu. Ví dụ: Bạn đã nhập dữ liệu trên một bảng có quan hệ mà bản ghi đang nhập không thể liên



kết được tới được một bản ghi nào của bảng có quan hệ 1 với nó (nhập một hàng bán mà mã hàng đó chưa có trong bảng danh mục hàng hoá).

**Khắc phục:** Tìm và nhập cho đúng giá trị theo bảng quan hệ 1 tương ứng. Tham khảo cách khắc phục lỗi này ở mục 5: thuộc tính Lookup.


## 4.2 Một số thao tác xử lý dữ liệu trên bảng

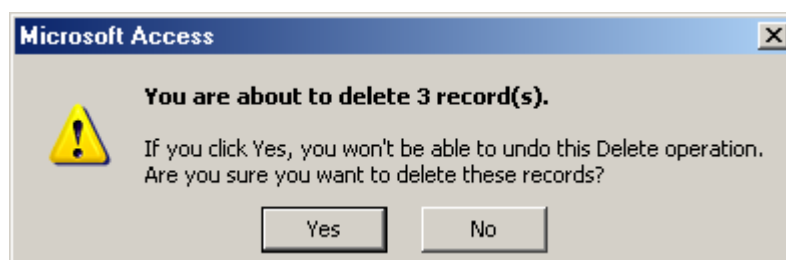
Đứng trước một bảng dữ liệu, có rất nhiều thao tác cần xử lý, ví dụ: sắp xếp bảng, tìm kiếm các bản ghi, xoá bản ghi nào đó,... Nội dung phần này trình bày cách sử dụng một số thao tác đó.

### a. Xoá bản ghi

Xoá bản ghi là thao tác xoá bỏ một số bản ghi ra khỏi bảng. Với bảng dữ liệu đang mở có thể thực hiện 2 bước sau để xoá các bản ghi:

**Bước 1:** Chọn những bản ghi cần xoá. Có thể chọn một hoặc nhiều bản ghi bằng cách dùng chuột đánh dấu đầu dòng những bản ghi cần chọn;

**Bước 2:** Ra lệnh xoá bằng cách: mở thực đơn **Edit | Delete Record** hoặc nhấn nút **Delete Record** trên thanh công cụ  hoặc nhấn phải chuột lên vùng đã chọn, tiếp theo nhấn **Delete Record**. Một hộp thoại xuất hiện để bạn khẳng định một lần nữa việc xoá dữ liệu:



- Chọn **Yes** để đồng ý xoá;
- Nhấn **No** để huỷ lệnh xoá.



### **Chú ý**

Dữ liệu đã đồng ý xoá sẽ không thể phục hồi lại được. Nên phải cân nhắc trước quyết định này cũng như cân nhắc trước các câu hỏi kiểu Yes/No trước khi quyết định.

## b. Sắp xếp dữ liệu

Sắp xếp là việc thay đổi thứ tự hiển thị một bảng dữ liệu theo một trật tự nào đó. Kết quả của việc sắp xếp giúp người dùng có thể quan sát được tốt hơn dữ liệu trên bảng, tất nhiên muốn quan sát bảng dữ liệu theo trường nào phải thực hiện sắp xếp bảng theo dữ liệu trường ấy. Cách sắp xếp dữ liệu trên bảng đang mở như sau:

**Bước 1:** Đặt con trỏ lên trường (cột) muốn sắp xếp;

**Bước 2:** Nhấn nút lệnh sắp xếp trên thanh công cụ:  - sắp xếp tăng dần hoặc  - sắp xếp giảm dần.

Bạn sẽ thu được kết quả như mong muốn.

## c. Lọc dữ liệu

Lọc dữ liệu là việc lọc ra những bản ghi trên bảng có cùng một số giá trị. Kết quả việc lọc dữ liệu sẽ giúp người dùng làm việc một cách hiệu quả trên tập hợp các bản ghi họ mong muốn.

Một ví dụ về tính hiệu quả của việc lọc dữ liệu:

Sắp đến ngày Quốc tế Phụ nữ 8-3. Cơ quan quyết định thưởng mỗi chị em 100,000đ. Đã có cột thưởng trên bảng cùng danh sách tất cả cán bộ cơ quan, làm sao để có thể nhập vào giá trị cột thưởng cho chị em (không nhập cho nam giới) là nhanh nhất.

Nếu cứ lần mò xem ai là nữ rồi nhập 100,000 cho cột thưởng sẽ là rất lâu. Nếu làm theo cách sau:

**Bước 1:** Lọc ra danh sách là các cán bộ nữ;


**Bước 2:** Nhập 100,000 cho tất cả các bản ghi đã lọc. Như vậy chắc chắn những cán bộ đang hiển thị (sau khi đã lọc) đều là chị em phụ nữ, không bỏ sót ai mà cũng không nhầm một người nam nào được lọc vào danh sách thưởng!

Các bước để lọc dữ liệu trên một bảng đang mở như sau:

**BƯỚC 1:** Nhấn phải chuột lên trường cần lọc dữ liệu. Một menu xuất hiện:

**BƯỚC 2:** Thiết lập điều kiện lọc trên trường đang chọn. Có rất nhiều cách để xác định điều kiện lọc:

- Nếu muốn lọc những bản ghi có cùng giá trị của bản ghi đang chọn hãy chọn mục 

- Muốn lọc những bản ghi thoả mãn điều kiện nào đó, hãy gõ điều kiện lên mục:  Ví dụ:

+ Gõ **>=10** - để lọc ra những bản ghi có giá trị trường đang thiết lập lọc lớn hơn hoặc bằng 10;

+ Gõ **<>3** - lọc ra những bản ghi có giá trị trường đang lọc khác 3,...

+ Đặc biệt: giá trị trống, rỗng được miêu tả là **Null**;

+ Lọc ra những người tên Nam gõ như sau **Like '\*Nam'**;

+ Lọc ra giá trị trong khoảng 10 đến 15 gõ như sau: **Between 10 And 15** (tham khảo toán tử Like và Between ở chương sau- Queries).

Muốn huỷ chế độ đặt lọc, nhấn phải chuột lên bảng dữ liệu và chọn mục:



## 5. Thuộc tính LOOKUP

Qua cách nhập dữ liệu cho bảng có quan hệ nhiều trên CSDL ta thấy việc nhập dữ liệu cho trường tham gia liên kết của bảng nhiều đòi hỏi phải có độ chính xác với dữ liệu trên bảng quan hệ 1 (phải nhớ mã để nhập). Trong thực tế với những danh mục lên đến hàng trăm, thậm chí nhiều hơn nữa thì việc nhớ mã để nhập dữ liệu quả là khó khăn: hoặc gõ sai mã, nguy hiểm hơn gõ đúng nhưng nhầm mã. Thuộc tính LOOKUP sẽ giúp giải quyết phần nào việc khó khăn trong nhập dữ liệu trên các bảng quan hệ nhiều như vậy.

Thuộc tính LOOKUP được thiết lập tại trường tham gia liên kết trên bảng có quan hệ nhiều sang trường tham gia liên kết của bảng có quan hệ 1.

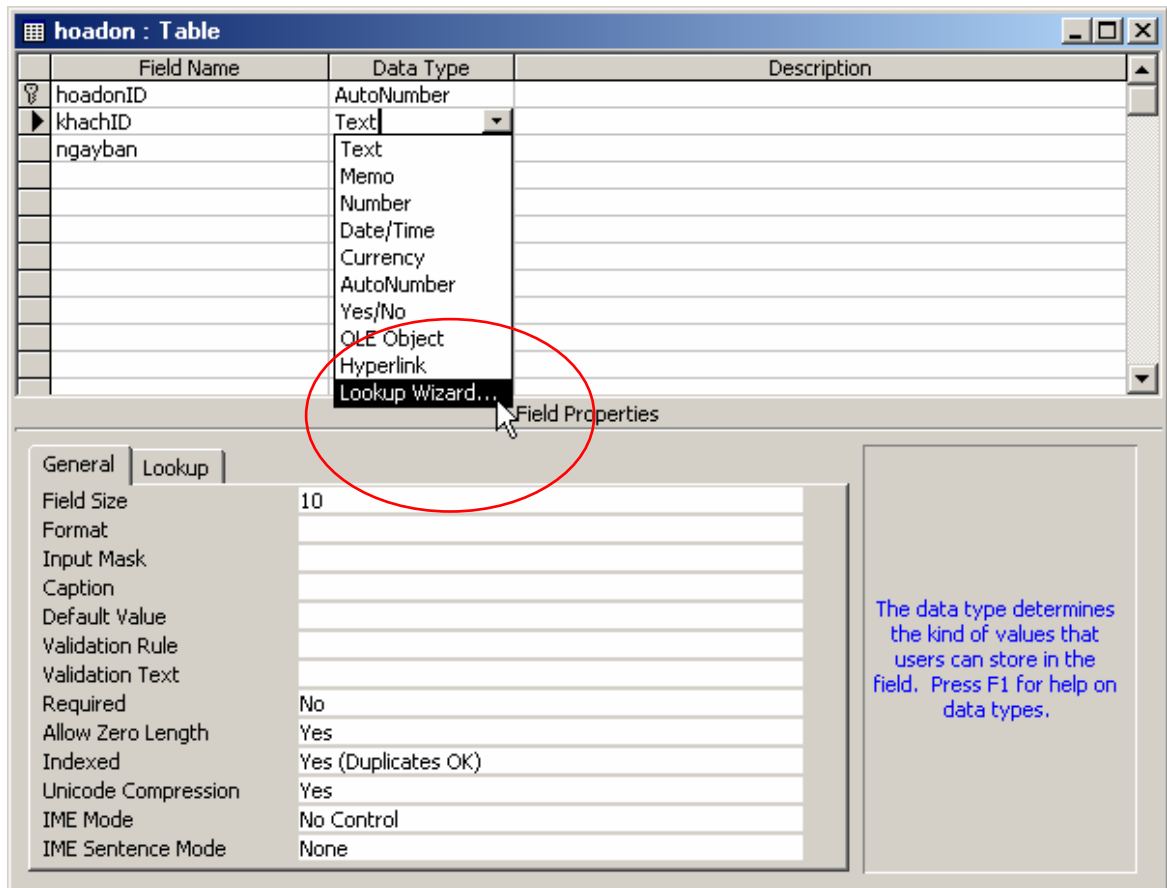
Ví dụ: Với CSDL Quản lý bán hàng thì:

- trường *khachID* của bảng *HOADON* phải thiết lập thuộc tính LOOKUP sang trường *khachID* của bảng *KHACH*;
- trường *hangID* của bảng *HANGBAN* phải thiết lập thuộc tính LOOKUP sang trường *hangID* của bảng *HANG*;
- trường *hoadonID* của bảng *HANGBAN* phải thiết lập thuộc tính LOOKUP sang trường *hoadonID* của bảng *HOADON*.

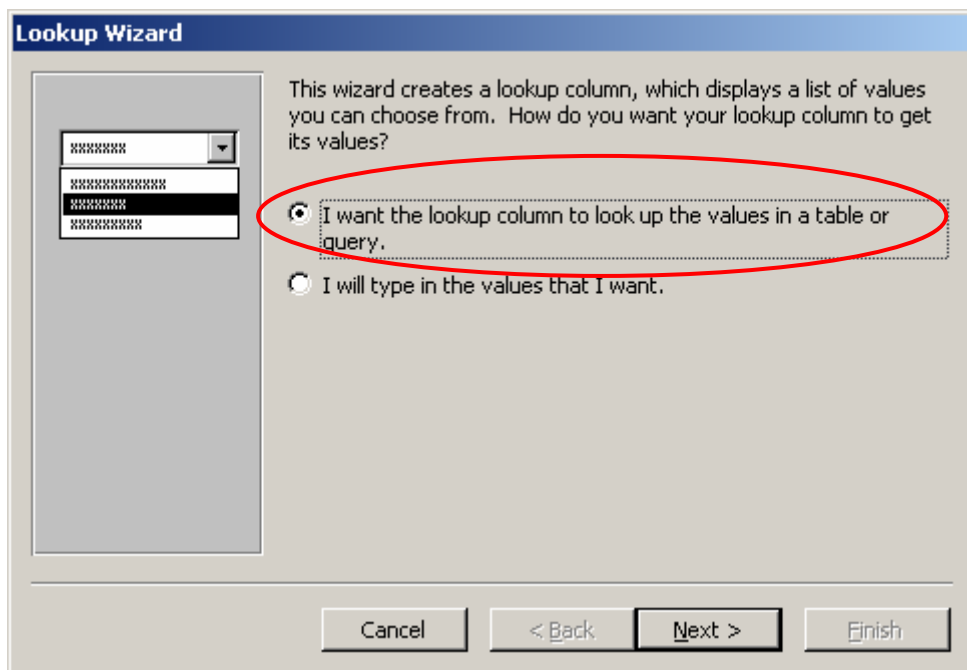
Thông thường, ứng với mỗi quan hệ 1-∞ đã được thiết kế cần phải thiết lập thuộc tính LOOKUP cho trường tham gia liên kết từ bảng quan hệ 1 sang trường tham gia liên kết của bảng quan hệ nhiều. Một trong các cách thiết lập thuộc tính này đơn giản nhất là trình LookUp Wizard của Access. Dưới đây là ví dụ về thiết lập thuộc tính LOOKUP cho trường *khachID* của bảng *HOADON* sang trường *khachID* của bảng *KHACH* trong CSDL quản lý bán hàng:

**Bước 1:** Mở bảng có trường cần thiết lập LOOKUP ra (bảng *HOADON*) ở chế độ Design View bằng cách: chọn tên bảng, nhấn nút **Design**;

**Bước 2:** Kích hoạt trình LookUp Wizard bằng cách: Tại cột Data Type của trường cần thiết lập thuộc tính LOOKUP (trường *khachID*), chọn mục **Lookup Wizard..** từ danh sách thả xuống:



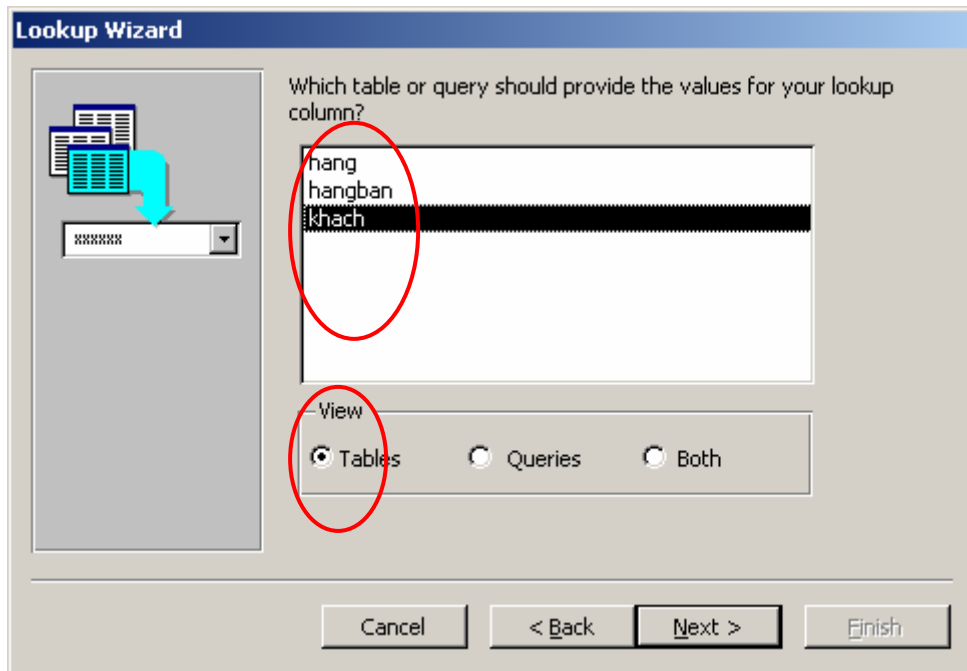
Hộp thoại Lookup Wizard xuất hiện:



Hộp thoại này hỏi *Dữ liệu để đưa vào danh sách chọn lấy từ đâu?* Trong trường hợp này danh sách dữ liệu để chọn lấy từ bảng KHACH, nên chọn mục *I want the lookup column to look up the values in a table or query.*

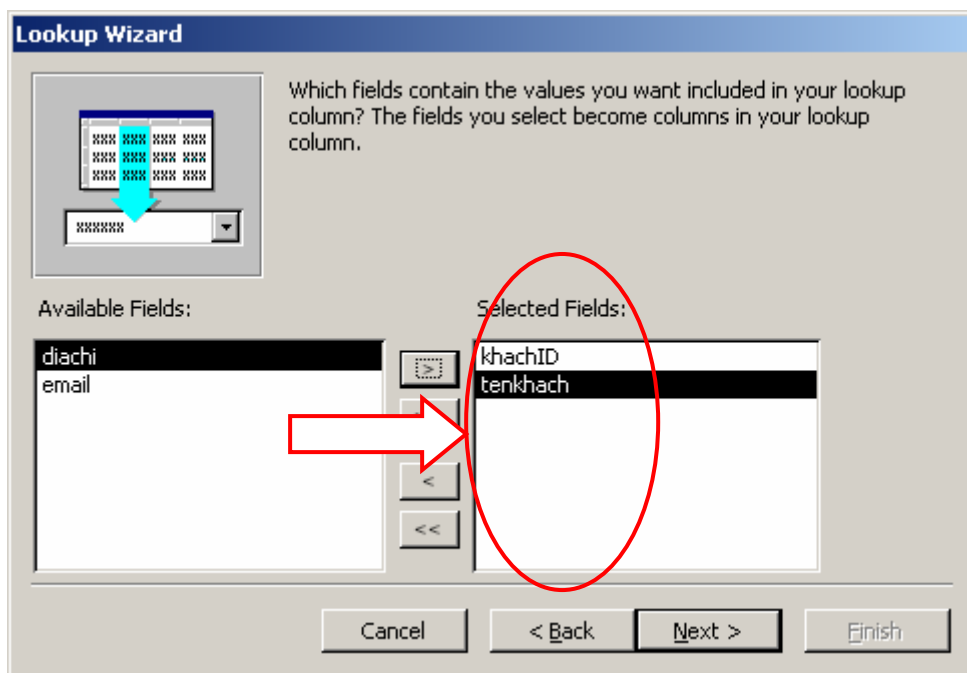
Nhấn **Next** để tiếp tục:

**Bước 3:** Chọn dữ liệu cho danh sách:



Hộp thoại trên trả lời câu hỏi: *Bảng (query) nào chứa dữ liệu cần đưa vào danh sách?*. Trong trường hợp này lấy dữ liệu từ bảng KHACH nên mục **View** chọn *Tables*; và chọn bảng *KHACH*.

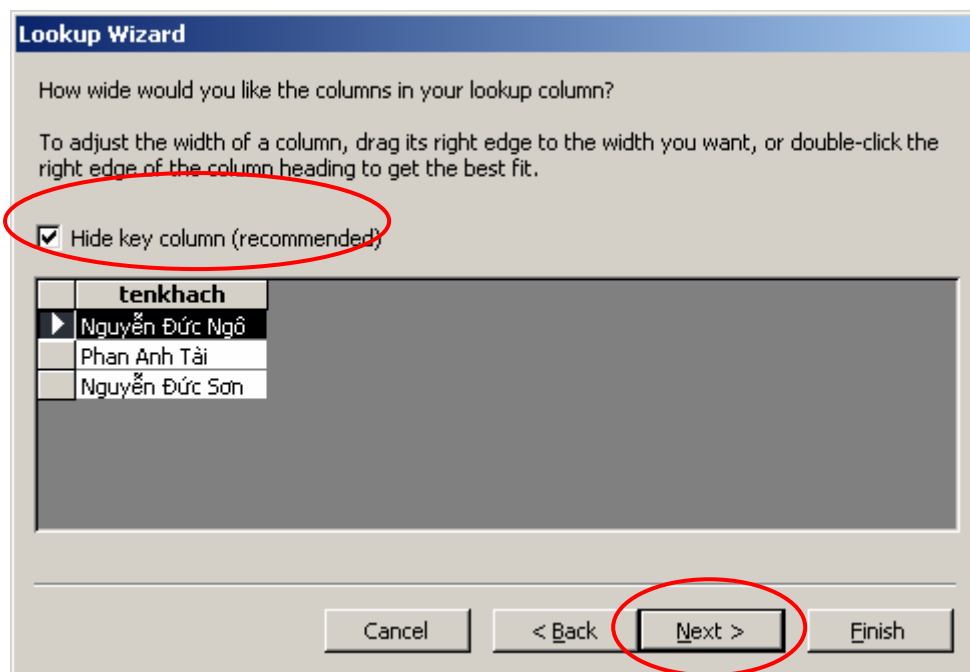
Chọn xong nhấn **Next**, hộp thoại sau xuất hiện:



Hộp thoại này trả lời câu hỏi: *Giá trị những trường nào của bảng (query) đã chọn sẽ được hiển thị trên danh sách?* Hãy dùng các nút >, >>, <, << để đưa 2 trường *khachID* và *tenkhach* từ danh sách **Available Fields:** (danh sách các trường có thể chọn) sang danh sách **Selected Fields:** (danh sách các trường đã chọn).

Lưu ý: Về nguyên tắc chỉ cần đưa trường *khachID* của bảng khác vào danh sách Selected Fields: là đủ, tuy nhiên nên đưa thêm trường *tenkhach* để thuận tiện hơn khi chọn lựa dữ liệu khi nhập sau này.

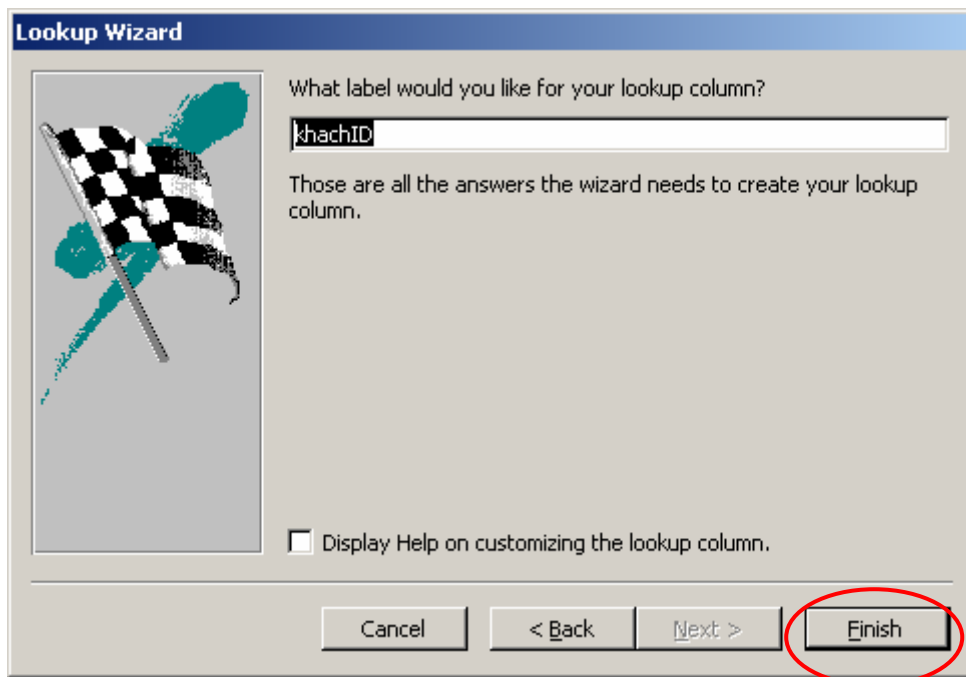
Chọn xong nhấn **Next** để tiếp tục:



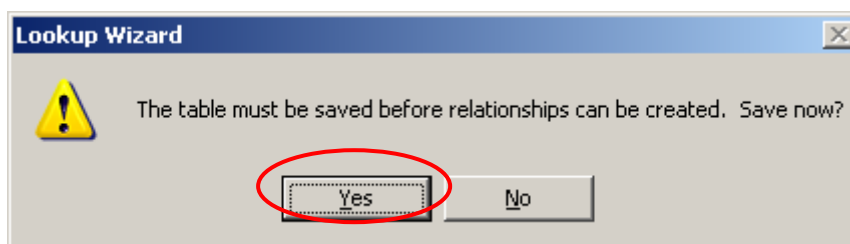
Nếu muốn ẩn mã khách khi chọn dữ liệu hãy chọn (checked) hộp

Hide key column (recommended) nếu không thì bỏ qua;

Nhấn **Next** để tiếp tục



Cuối cùng nhấn **Finish** để kết thúc tiến trình Lookup Wizard. Khi đó hộp thoại yêu cầu ghi lại cấu trúc bảng xuất hiện:



Hãy nhấn **Yes** để đồng ý.

## 6. Quy trình xây dựng CSDL Access

Đến đây chúng tôi có thể khuyến cáo một quy trình tốt để xây dựng một CSDL Access theo thiết kế sẵn có :


**BƯỚC 1** : Lần lượt xây dựng cấu trúc từng bảng dữ liệu trong CSDL. Với mỗi bảng dữ liệu khi khai báo cấu trúc cần giải quyết các công việc sau :

- Khai báo danh sách các trường của bảng ở cột *Field Name*;
- Chọn kiểu dữ liệu phù hợp cho các trường ở cột *Data Type*;
- Thiết lập trường khoá cho bảng;
- Thiết lập một số khác cần thiết cho các trường như : *Field Size*, *Format*, *Input Mark*, *Required*, *Validate Rule*, ...



- Ghi tên bảng

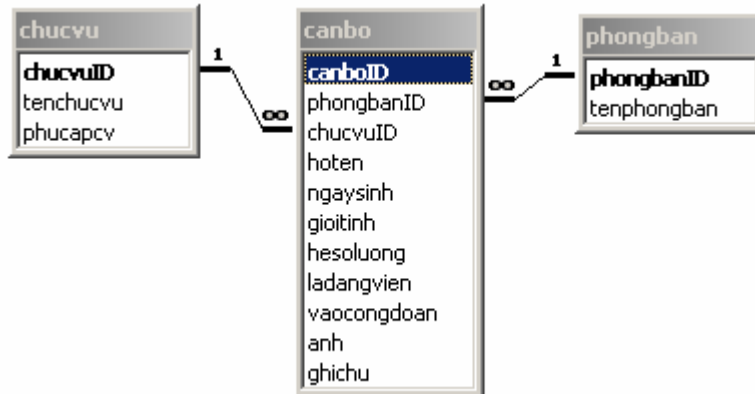
**BƯỚC 2 :** Lần lượt thiết lập thuộc tính LOOKUP cho các trường một cách phù hợp. Mỗi quan hệ trên bảng thiết kế sẽ cần một thao tác thiết lập thuộc tính LOOKUP (sử dụng trình **LookUp Wizard**) từ trường trên bảng quan hệ nhiều sang trường bảng quan hệ một;

**BƯỚC 3 :** Thiết lập các thuộc tính đảm bảo toàn vẹn dữ liệu cần thiết cho các quan hệ tại cửa sổ Relationships (menu **Tool | Relationships..** hoặc nhấn nút  trên thanh công cụ);

**BƯỚC 4 :** Thực hiện nhập dữ liệu cho các bảng nếu cần. Chú ý : bảng có quan hệ 1 phải được nhập dữ liệu trước bảng có quan hệ nhiều.

## Bài tập

1. Xây dựng CSDL Quản lý lương cán bộ một cơ quan có cấu trúc như sau :

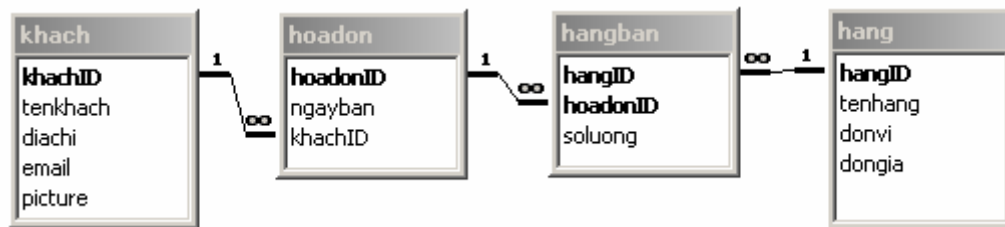


### Yêu cầu :

- Thiết kế cấu trúc các bảng một cách phù hợp : kiểu dữ liệu các trường; trường khoá; thuộc tính Lookup và các thuộc tính khác;
- Thiết lập quan hệ cùng các thuộc tính đảm bảo toàn vẹn dữ liệu cho các quan hệ;
- Nhập dữ liệu như sau :
  - + 4 phòng ban;

- + 5 loại chức vụ;
- + 20 hồ sơ cán bộ.

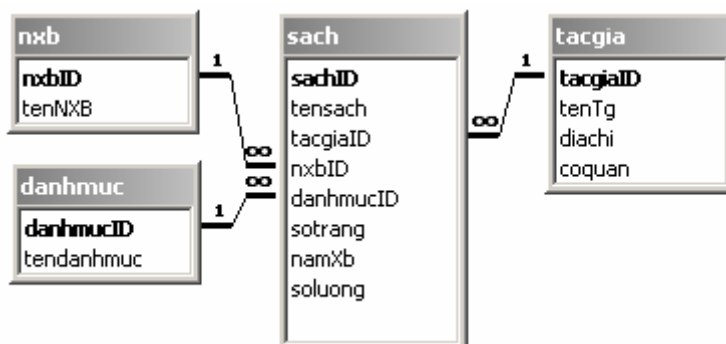
2. Xây dựng CSDL Quản lý việc bán hàng cho một cửa hàng có cấu trúc như sau :



### Yêu cầu :

- Thiết kế cấu trúc các bảng một cách phù hợp : kiểu dữ liệu các trường; trường khoá; thuộc tính Lookup và các thuộc tính khác;
- Thiết lập quan hệ cùng các thuộc tính đảm bảo toàn vẹn dữ liệu cho các quan hệ;
- Nhập dữ liệu như sau :
  - + 5 khách hàng;
  - + 15 danh mục hàng hoá có bán;
  - + lập 10 hoá đơn bán hàng;
  - + với 25 lượt hàng hoá được bán ra.

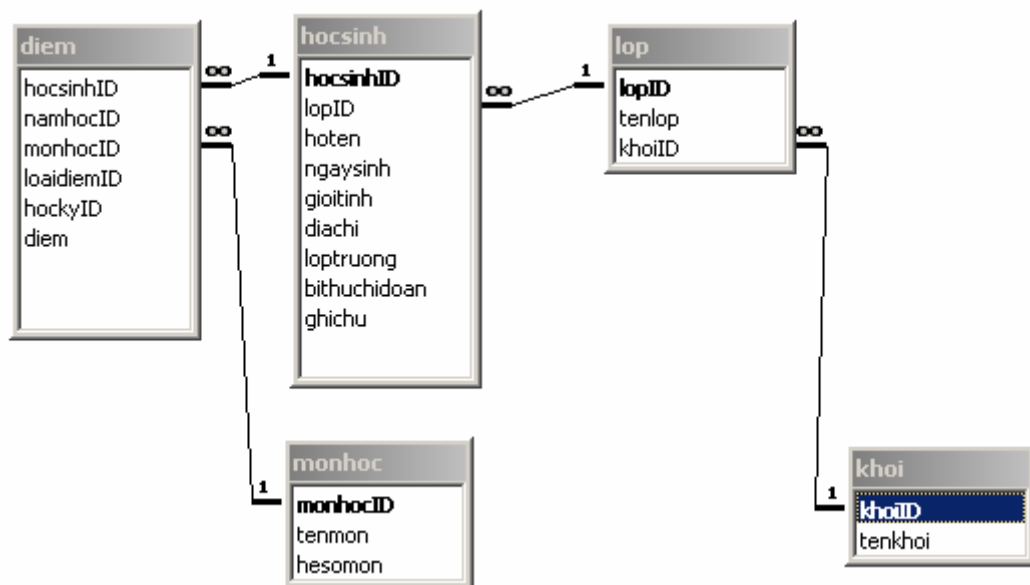
3. Xây dựng CSDL Quản lý sách một thư viện có cấu trúc như sau :



### Yêu cầu :

- Thiết kế cấu trúc các bảng một cách phù hợp : kiểu dữ liệu các trường; trường khoá; thuộc tính Lookup và các thuộc tính khác;
- Thiết lập quan hệ cùng các thuộc tính đảm bảo toàn vẹn dữ liệu cho các quan hệ;
- Nhập dữ liệu như sau :
  - + 5 tác giả;
  - + 3 nhà xuất bản;
  - + 3 danh mục sách;
  - + 20 đầu sách.

4. Xây dựng CSDL Quản lý điểm học sinh trường phổ thông có cấu trúc như sau :



#### Yêu cầu :

- Thiết kế cấu trúc các bảng một cách phù hợp: kiểu dữ liệu các trường; trường khoá; thuộc tính Lookup và các thuộc tính khác;
- Thiết lập quan hệ cùng các thuộc tính đảm bảo toàn vẹn dữ liệu cho các quan hệ;
- Nhập dữ liệu như sau:

- + 10 môn học phổ thông;
- + 3 khối học;
- + 15 lớp chia đều cho 3 khối;
- + 30 học sinh;
- + Và nhập điểm 2 học kỳ cho học sinh của một lớp nào đó.

# CHƯƠNG 2

## TRUY VẤN DỮ LIỆU

Trong chương trước đã giới thiệu những khái niệm cũng như các kỹ năng, trình tự cần thiết để có thể xây dựng tốt một CSDL trên Access. Chương này sẽ cung cấp những khái niệm cũng như những kỹ năng cần thiết để xử lý dữ liệu khi cần. Một trong những công cụ xử lý dữ liệu trực quan, hữu hiệu trên Access là **Query**.

Có rất nhiều dạng yêu cầu xử lý dữ liệu như: trích - lọc - hiển thị dữ liệu; tổng hợp - thống kê; thêm - bớt - cập nhật dữ liệu; ... Vì vậy sẽ tồn tại một số loại Query tương ứng để giải quyết các yêu cầu xử lý dữ liệu trên. Có 7 loại query trong Access :

- Select Query;
- Total Query;
- Crosstab Query;
- Append Query;
- Delete Query;
- Update Query;
- Make table Query.

Bản chất của Query là các câu lệnh SQL (Structured Queries Language- ngôn ngữ truy vấn dữ liệu có cấu trúc)- một ngôn ngữ truy vấn dữ liệu được dùng khá phổ biến trên hầu hết các hệ quản trị CSDL hiện nay. Việc thiết kế một query là gián tiếp tạo ra một câu lệnh xử lý dữ liệu SQL. Việc thi hành query chính là việc thi hành câu lệnh SQL đã tạo ra.

## 1. SELECT queries

### 1.1 Cách tạo

Select query là loại truy vấn dùng trích - lọc - kết xuất dữ liệu từ nhiều nguồn khác nhau từ CSDL ra một bảng kết quả. Ví dụ:

- Đưa ra thông tin chi tiết bảng lương tháng 8;
- Đưa ra danh sách cán bộ là Đảng viên;
- Đưa ra thông tin chi tiết về các hoá đơn bán ra trong ngày hôm nay;
- Đưa ra doanh thu bán hàng của một tháng nào đó;
- ...

Tất cả những yêu cầu dạng như vậy (đưa ra một danh sách kết quả) đều có thể sử dụng **SELECT query** để đáp ứng. Mỗi yêu cầu xử lý dữ liệu cần phải tạo ra một Select query đáp ứng; mỗi query sẽ có một tên gọi (như cách đặt tên bảng dữ liệu); query sau khi đã tạo ra đều có thể chỉnh sửa lại được cấu trúc cũng như nội dung; qui trình để đáp ứng mỗi yêu cầu trên như sau: tạo một query đáp ứng được các yêu cầu → thi hành query đã tạo để thu nhận kết quả.

Tiếp theo minh hoạ qui trình các bước để tạo một select query đáp ứng yêu cầu đơn giản nhất:

Từ CSDL Quản lý lương cán bộ, hãy đưa ra bảng lương cán bộ với những thông tin sau: *canboID*, *hoten*, *ngaysinh*, *tencv*, *luongchinh*, *phucapcv*, *thuclinh*.

Trong đó: **luongchinh = hesoluong \* 290000**

**Thuclinh = luongchinh + phucapcv**

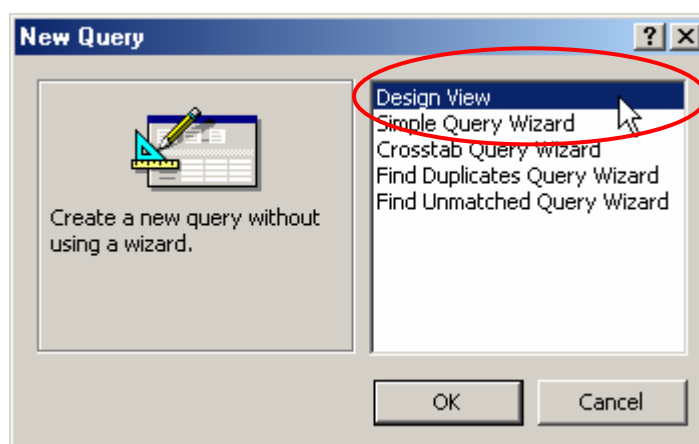
**Phân tích yêu cầu:**

(1). Thông tin yêu cầu đòi hỏi từ nhiều bảng khác nhau (bảng CANBO với các cột: canboID, hoten, ngaysinh; bảng CHUCVU với các cột: ten cv, phucapcv);

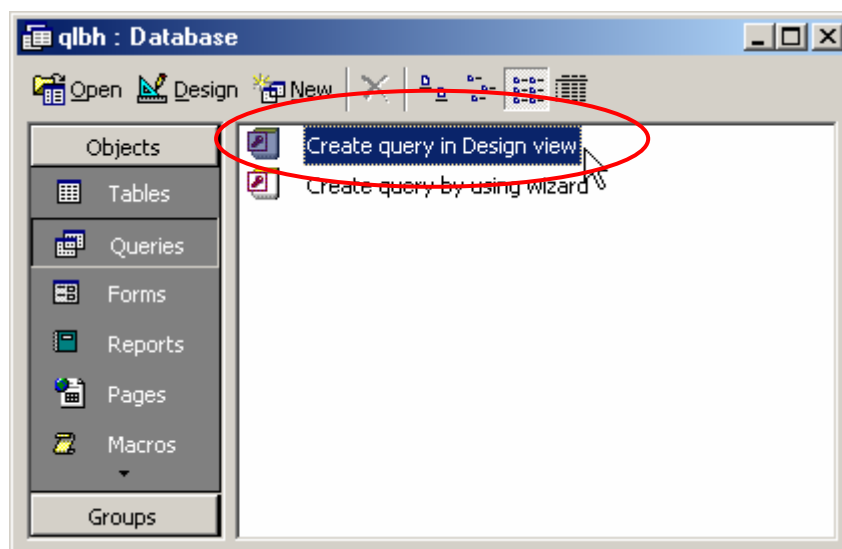
(2). Có những cột thông tin đã có sẵn trên CSDL (5 cột kể trên) nhưng một số cột yêu cầu phải được tính bởi biểu thức: cột luongchinh và cột thuclinh.


### Hướng dẫn cách làm:

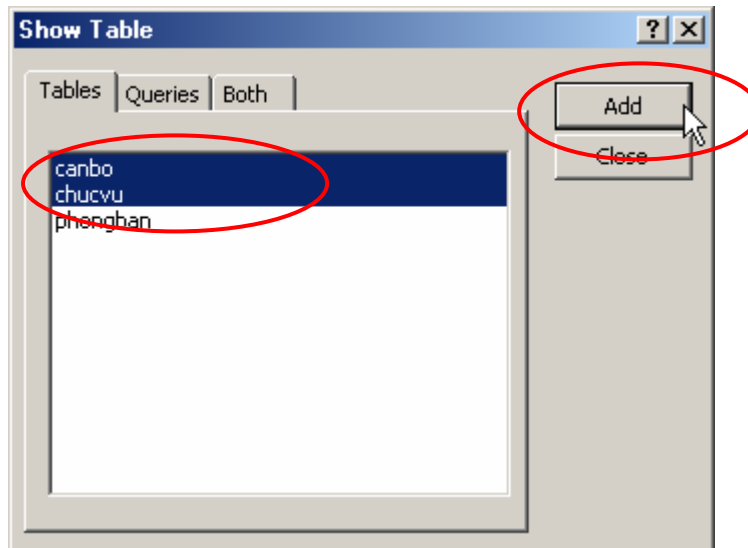
**Bước 1:** Ra lệnh tạo một Select query mới bằng cách: kích hoạt thẻ Queries, nhấn nút **New**, chọn **Design View**, nhấn **Ok** (hình dưới):



Hoặc nhấn đúp chuột lên biểu tượng **Create query in Design view** trên màn hình:

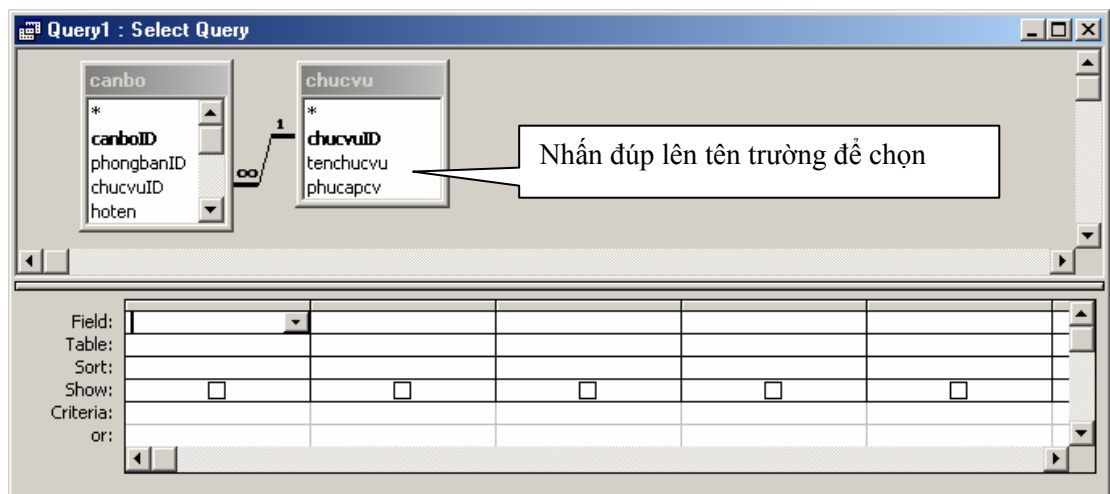


**Bước 2:** Chọn những bảng có chứa dữ liệu liên quan lên màn hình thiết kế query từ cửa sổ **Show Table** (không thấy cửa sổ này nhấn nút Show table  trên thanh công cụ):



Hãy chọn các bảng có liên quan (bảng CANBO và bảng CHUCVU); chọn xong nhấn **Add** và nhấn **Close**. Khi đó màn hình thiết kế query xuất hiện:

**Bước 3:** Khai báo những thông tin cần thiết cho query:



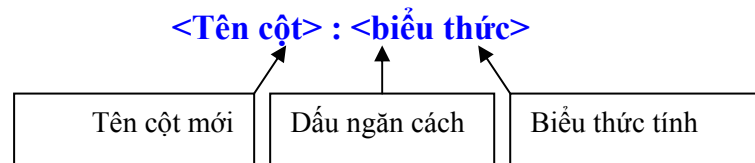
Dòng **Field:** là nơi khai báo danh sách các thông tin (cột dữ liệu) của bảng kết quả.

Có 2 loại thông tin bài toán yêu cầu: thông tin có sẵn từ các trường trên CSDL như *canboID*, *hoten*, *tencv*, *phucapcv* và thông tin phải được tính theo một biểu thức nào đó như: *luongchinh* và *thuclinh*.

Muốn hiển thị trường (field) nào lên query, chỉ việc nhấn đúp chuột lên tên trường đó hoặc dùng chuột kéo tên chúng từ các bảng lên dòng Field. Hãy dùng phương pháp này để đưa 4 trường *canboID*, *hoten*, *ngaysinh* và *phucapcv* lên dòng *Field*.



Với 2 cột *luongchinh* và *thuclinh* phải đưa vào query bằng cách tạo một cột mới với một tên gọi <tên cột> được tính bằng một <biểu thức> trên dòng **Field** theo cú pháp như sau:

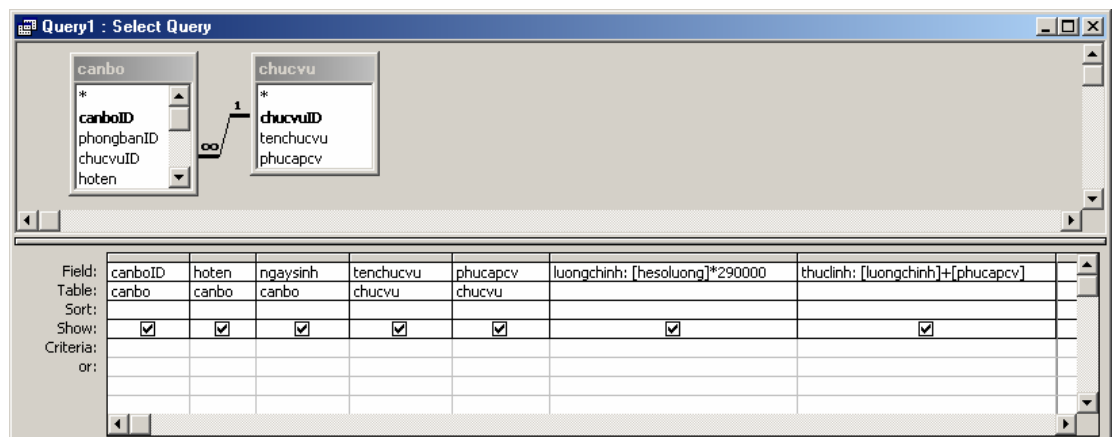


Ví dụ:

**Luongchinh : hesoluong \* 290000** để tạo thêm cột *luongchinh* mới



**Thuclinh : luongchinh + phucapcv** để tạo thêm cột *thuclinh* mới.

Cuối cùng, màn hình thiết kế query như sau:



*Có thể ghi query lại với một tên gọi khi ra lệnh cắt **Alt + S***

**Bước 4:** Sử dụng query vừa tạo. Một query sau khi đã tạo xong có thể:

- (1) Kích hoạt chúng để lấy kết quả bằng cách: nhấn đúp chuột lên tên query. Nếu đang trong chế độ thiết kế (Design view), nhấn nút View  trên thanh công cụ hoặc nhấn phải chuột lên Query đang thiết kế chọn  Bạn sẽ xem được kết quả trả về của query và có thể thao tác bảng dữ liệu này như trên một Table (xem mục 4.2, Phần 1- Tạo CSDL):

canboID	hoten	ngaysinh	tenchucvu	phucapcv	luongchinh	thuclinh
1	Nguyễn Đức Ngô	12/04/1976	Giám đốc	500000	904800	1404800
4	Nguyễn Thu Loan	05/02/1978	Nhân viên	200000	516200	716200
5	Hà Anh Tú	31/12/1975	Nhân viên	200000	516200	716200
6	Phan Quang Sáng	03/09/1976	Nhân viên	200000	696000	896000
3	Nguyễn Hải Hà	13/09/1979	Phó giám đốc	400000	611900	1011900
2	Phan Anh Quân	23/07/1978	Trưởng phòng	300000	611900	911900
7	Nguyễn Ánh Tuyết	02/07/1980	Trưởng phòng	300000	696000	996000

(2) Sửa lại được cấu trúc thiết kế query khi chọn chúng và nhấn nút **Design**.



Bản chất của một SELECT query là câu lệnh SQL có dạng:-----

**SELECT ... FROM ...**

**Một số thiết lập khác cho Query**

Dòng **Sort**: để thiết lập thứ tự sắp xếp dữ liệu trên Query. Muốn sắp xếp dữ liệu cho trường nào, thiết lập thuộc tính Sort cho trường ấy. Có 2 giá trị cho thuộc tính Sort: *Ascending* - sắp xếp tăng dần và *Descending* - sắp xếp giảm dần. Trường nào đứng trước sẽ được thứ tự sắp xếp trước.

Field:	canboID	hoten	ngaysinh
Table:	canbo	canbo	canbo
Sort:	Ascending		Descending
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:			



Bản chất của yêu cầu sắp xếp dữ liệu thể hiện ở mệnh đề:-----

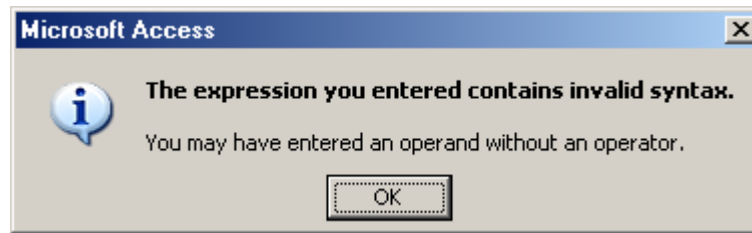
**... ORDER BY ... [ASC] [DESC]** trong câu lệnh SQL

Dòng **Show**: để chỉ định hiển thị hay không hiển thị dữ liệu trường đó ra bảng kết quả. Nếu chọn (checked)- dữ liệu sẽ được hiển thị ra bảng kết quả.

Field:	canboID	hoten	ngaysinh
Table:	canbo	canbo	canbo
Sort:	Descending		
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Criteria:			

**Chú ý:** Có 2 lỗi hay mắc phải:

- Lỗi gõ sai biểu thức:

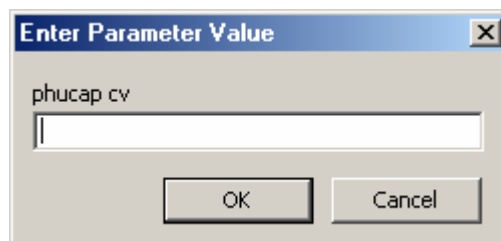


Nguyên nhân gây ra lỗi này là rất nhiều, có thể liệt kê ra đây một số tình huống:

**Tình huống 1:** Có thể một trong số tên các trường bạn gõ trong biểu thức có chứa dấu cách hoặc các ký tự đặc biệt. Ví dụ: trường *hesoluong* của bạn là he so luong thì phải gõ như sau: *[he so luong]* – thêm dấu móc vuông vào 2 đầu của tên trường;

**Tình huống 2:** Sai ký pháp lô gíc của biểu thức: do viết thừa hoặc thiếu các toán tử, toán hạng, hoặc các dấu mở đóng ngoặc không khớp..;

- Gõ không đúng tên trường trong biểu thức:

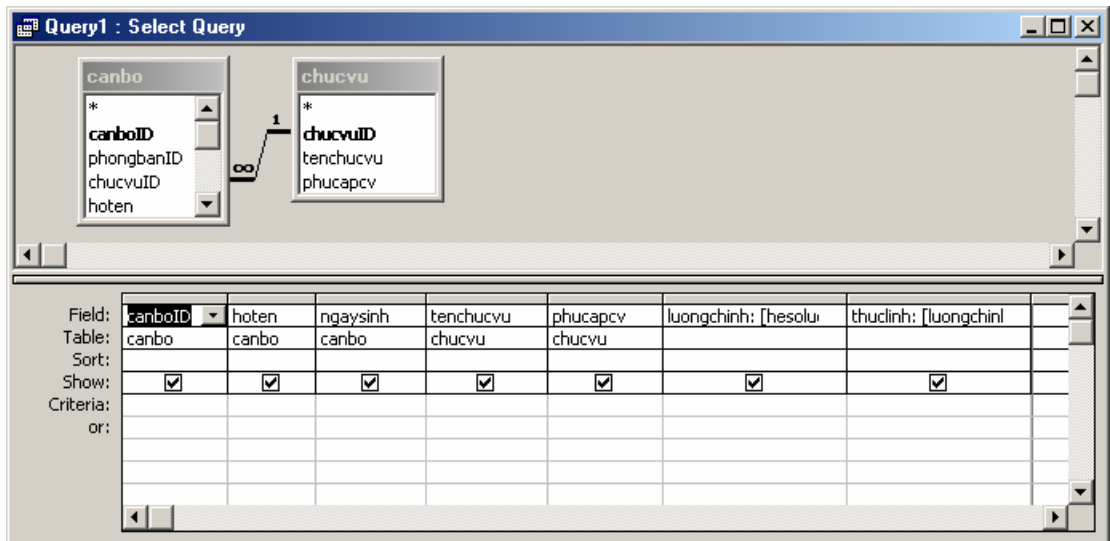


Lỗi này xảy ra khi bạn đã gõ sai tên trường. Tên hiển thị trên hộp thoại (*phucap cv*) máy tính không hiểu, có thể tên đúng của trường này là *phucapcv*. Bạn phải lần tới biểu thức có chứa tên trên và kiểm tra sửa cho đúng với tên trường có trong CSDL. Lỗi này hay xảy ra khi thiết kế bảng đặt tên các trường có chứa dấu cách!

## 1.2 Lọc dữ liệu

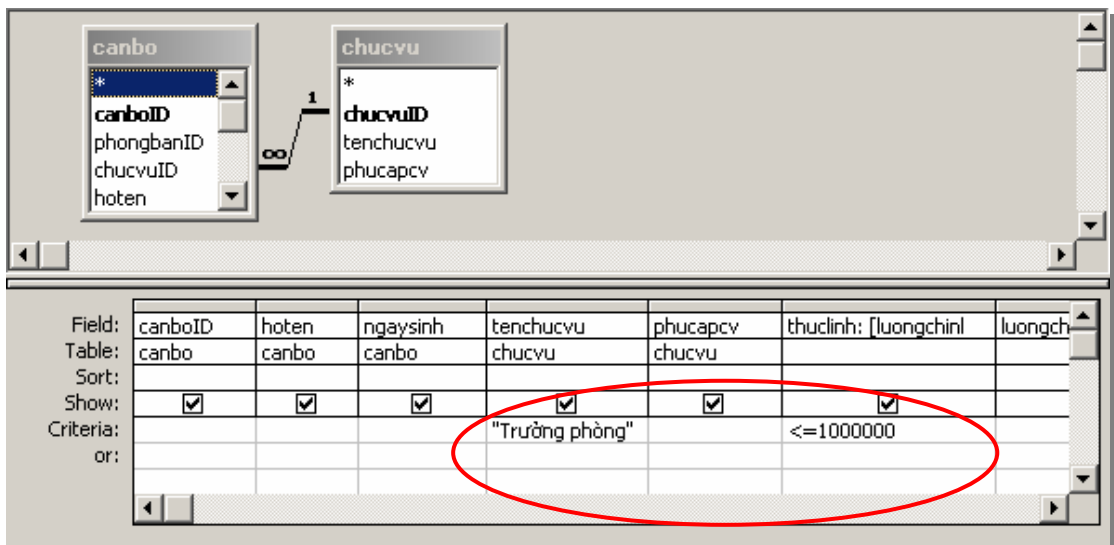
Khác với bảng (Tables), Queries cung cấp một khả năng lọc dữ liệu khá hoàn chỉnh; Có thể lọc ra những dữ liệu theo những điều kiện phức tạp hơn, đặc biệt có thể chấp nhận những giá trị lọc là các tham biến.

Để lọc dữ liệu, bạn phải thiết lập điều kiện đặt lọc lên vùng **Criteria** của queries (trong chế độ đang thiết kế).

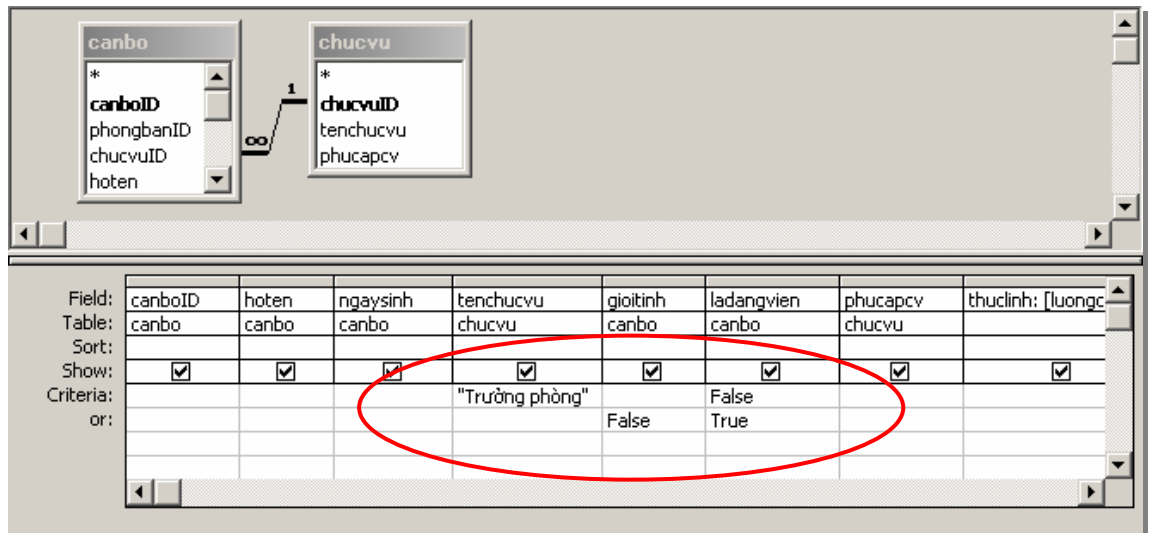


Các điều kiện nằm trên cùng một dòng Criteria sẽ được nối với nhau bởi toán tử **AND** (và); mỗi dòng Criteria sẽ được nối với nhau bởi toán tử **OR** (hoặc). Xét các ví dụ sau thực hiện lọc trên query bảng lương vừa được tạo ra:

**Ví dụ 1:** Lọc ra những cán bộ là *trưởng phòng* có *thực lĩnh*  $\leq 1,000,000$  đ



**Ví dụ 2:** Lọc ra những cán bộ là *Trưởng phòng* mà không phải là *Đảng viên* hoặc những cán bộ *Nữ* là *Đảng viên*:



**Ví dụ 3:** Lọc ra những cán bộ có tên là *Nam*. Chú ý: Tên chỉ là một phần của trường *Hoten*:

canboID	hoten	ngaysinh	tenchucvu
canbo	canbo	canbo	chucvu
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	Like "*Nam"		

Toán tử **LIKE** để biểu diễn những giá trị mang tính tương đối (có thể chỉ giống một phần giá trị thực). Cú pháp biểu diễn toán tử này như sau:

**Like '<giá trị tương đối>'**

Trong <giá trị tương đối> có thể chứa hằng (*Nam*) và các ký tự đại diện. Có 2 ký tự đại diện là:

- Ký tự \* để biểu diễn bất kỳ giá trị nào;
- Ký tự ? để biểu diễn một ký tự bất kỳ.

Một số ví dụ minh họa toán tử LIKE:

- **Like 'Nguyễn\*'** - lọc ra những người họ Nguyễn. 6 ký tự đầu là Nguyễn, các ký tự còn lại là thoải mái;
- **Like '\*Đức\*'** - lọc ra những người có Họ hoặc Đệm hoặc Tên là Đức;
- **Like '\*/\*/1980'** - lọc ra những người sinh năm 1980;
- **Like '\*/11/\*'** - lọc ra những người sinh tháng 11;

- **Like '\*/\*/198?'** - lọc ra những người sinh từ năm 1980 đến năm 1989;
- ...

**Ví dụ 4:** Lọc ra những cán bộ có  $2,000,000 \geq Thuclinh \geq 1,000,000$ :

luongchinh: [hes:	thuclinh: [luongchinh]+[phucapcv]
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Between 1000000 And 2000000	

Toán tử **BETWEEN** để lọc ra các giá trị nằm trong một khoảng nào đó. Thường giải quyết các yêu cầu *lớn hơn hoặc bằng bao nhiêu và nhỏ hơn hoặc bằng bao nhiêu?*

Cú pháp toán tử Between như sau:

**Between <giá trị nhỏ> And <giá trị lớn>**

Cả 4 ví dụ trên đều đặt lọc theo các tham trị (lọc cứng). Trong trường hợp sử dụng các tham biến vào điều kiện lọc, xin tiếp tục theo dõi hướng dẫn dưới đây:

Muốn thiết lập giá trị đặt lọc nào là tham biến bạn làm như sau:

*[Gõ một lời nhắc trong ngoặc vuông]* tại đúng vị trí tham số cần thiết lập.

Ví dụ: Lọc ra danh sách các cán bộ phòng ban nào đó, làm như sau:

hoten	tenphongban	ngaysinh
canbo	phongban	canbo
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
[Gõ tên phòng ban cần xem]		

Khi thi hành query, một hộp thoại yêu cầu gõ vào tham số cần lọc:

Hãy gõ vào Tên phòng ban cần xem. Việc thiết lập tham số kiểu này không giới hạn trên một query.



Bản chất của yêu cầu lọc dữ liệu thể hiện ở mệnh đề:-----

... **WHERE** ... trong câu lệnh SQL

## 2. TOTAL queries

Tổng hợp – là phép xử lý dữ liệu khá phổ biến. Trong Access, query là một trong những công cụ xử lý khá tốt việc này. **Total query** là một điển hình.

Hãy xét một số yêu cầu tổng hợp dữ liệu như sau:

tenphongban	TSCB	Tong LC	tong PCCV	Tong Thuclinh
Kinh doanh	3	2212700	1000000	3212700
Tổng hợp	4	2340300	1100000	3440300

Record: 1 of 2

tenchucvu	tscb
Giám đốc	1
Nhân viên	3
Phó giám đốc	1
Trưởng phòng	2

Record: 1 of 4

hoadonID	ngayban	tenkhach	tongtien
1	12/10/2004	Nguyễn Đức Ngô	893
2	24/10/2004	Nguyễn Đức Ngô	114
3	10/10/2004	Phan Anh Tài	855
4	02/11/2004	Nguyễn Đức Sơn	1394

Record: 3 of 4

	tenhang	ngayban	Tong so
	RAM 256	10/10/2004	12
	CPU P4 2.4	10/10/2004	3
	Hard disk	12/10/2004	9
	Floppy disk	12/10/2004	3
	CPU P4 2.4	12/10/2004	2
	RAM 256	24/10/2004	3
	Floppy disk	02/11/2004	8
	CPU P4 2.4	02/11/2004	10

Tất cả những yêu cầu trên đều có thể được đáp ứng bằng các **TOTAL query**. Dưới đây sẽ hướng dẫn cách tạo **Total query**. Ví dụ: tạo query đưa ra bảng tổng hợp *Tenchucvu*, *Tổng số CB* (query thứ 2);

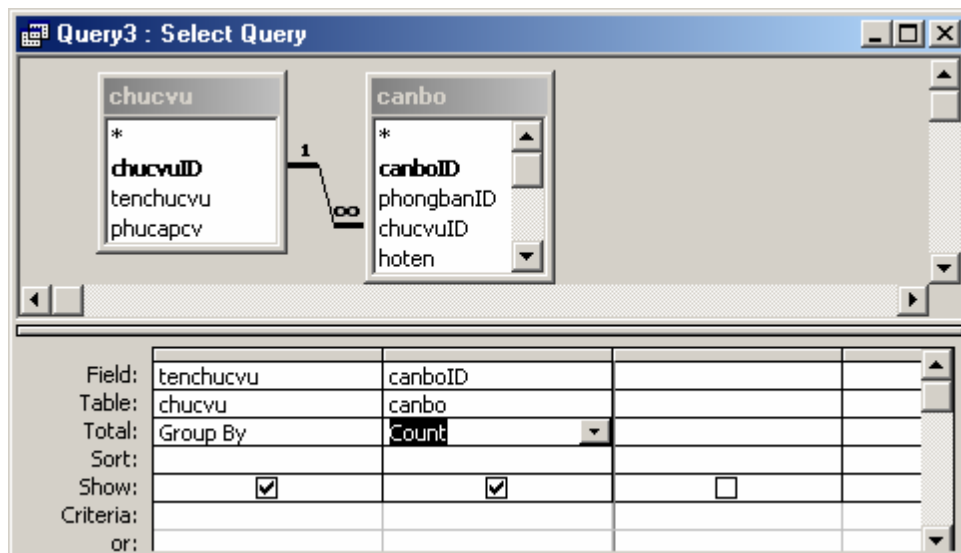
**BƯỚC 1:** Tạo một **select query** với đầy đủ các thông tin liên quan bảng tổng hợp:

Field:	tenchucvu	canboID		
Table:	chucvu	canbo		
Sort:				
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Criteria:				
or:				

**BƯỚC 2:** Tùy chọn **Total query** bằng cách mở thực đơn **View | Total** hoặc nhấn nút **Total**  $\Sigma$  trên thanh công cụ;

**BƯỚC 3:** Thiết lập các tùy chọn **Total** cho các trường một cách phù hợp như sau:





- Vì bài toán yêu cầu tổng hợp thông tin theo từng loại chức vụ, nên **Total** của trường *TENCHUCVU* là **Group By**;
- Trường *CANBOID* dùng để đếm số cán bộ từng chức vụ, nên chọn phép tổng hợp là **Count**.

Đến đây đã tạo xong query và có thể thi hành để thu nhận kết quả.

Ngoài phép Count để đếm, Total query còn cung cấp một số phép tổng hợp khác như:

- Sum** - tính tổng;
- AVG** - tính trung bình cộng;
- Max** - xác định giá trị lớn nhất;
- Min** - xác định giá trị nhỏ nhất



Bản chất của Total query thể hiện ở câu lệnh SQL:-----

**SELECT .. FROM .. GROUP BY ...**

-----

### 3. CROSSTAB queries

Ở mục 2 chúng ta đã được tìm hiểu về **Total query** trong việc đáp ứng các yêu cầu tổng hợp dữ liệu. Trên thực tế còn rất nhiều các yêu cầu tổng hợp khác mà **Total query** không thể đáp ứng được. Nhiều trong số đó như là:

	tenphongban	Giám đốc	Trưởng phòng	Phó giám đốc	Nhân viên
▶	Kinh doanh	1	1		3
	Tổng hợp		1	1	4

Record: 1 of 2

	tenphongban	Ts Nam	Ts Nu
▶	Kinh doanh	1	4
	Tổng hợp	5	1

Record: 1 of 2

	tenhang	Tongsoban	10/10/2004	12/10/2004	24/10/2004
▶	CPU P4 2.4	3	1	2	
	Floppy disk				
	Hard disk				
	RAM 256				1

Record: 1 of 4

Yêu cầu tổng hợp dữ liệu là các tiêu đề dòng và cột!

Tất cả những yêu cầu dạng trên có thể sử dụng **CROSSTAB query** để đáp ứng. Trước khi tìm hiểu cách tạo **Crosstab query**, chúng ta hãy tìm hiểu cấu trúc một **Crosstab**, được minh họa bởi hình sau:

Row heading	Column heading
	Value

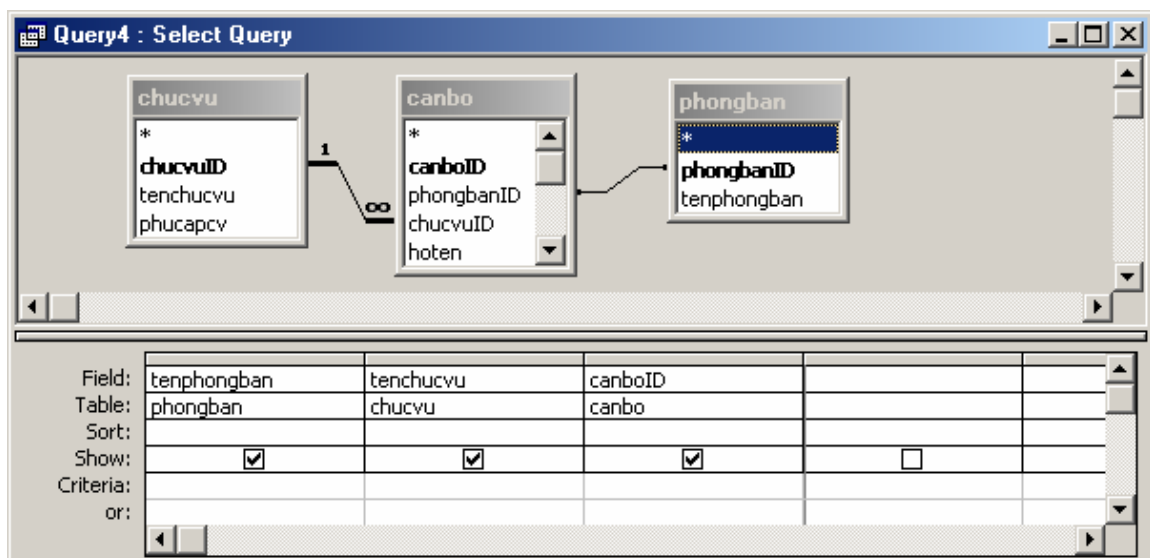
- **Row heading** là tiêu đề các dòng, có chứa các giá trị của các trường nào đó làm tiêu chí thống kê. Mỗi **Crosstab** phải có tối thiểu 1 trường làm Row heading;
- **Column heading** là tiêu đề các cột, có chứa các giá trị của một trường nào đó làm tiêu chí thống kê. Mỗi **Crosstab** chỉ có duy nhất 01 trường làm Column heading;
- **Value** là vùng dữ liệu tổng hợp (là các con số). Chỉ có duy nhất một trường làm **Value**, tương ứng với nó là một phép tổng hợp hoặc: đếm, tính tổng, tính trung bình cộng, max, min,...

Các bước để tạo một **Crosstab query**. Ví dụ tạo query đưa ra được bảng tổng hợp sau:

	tenphongban	Giám đốc	Trưởng phòng	Phó giám đốc	Nhân viên
▶	Kinh doanh	1	1		3
	Tổng hợp		1	1	4

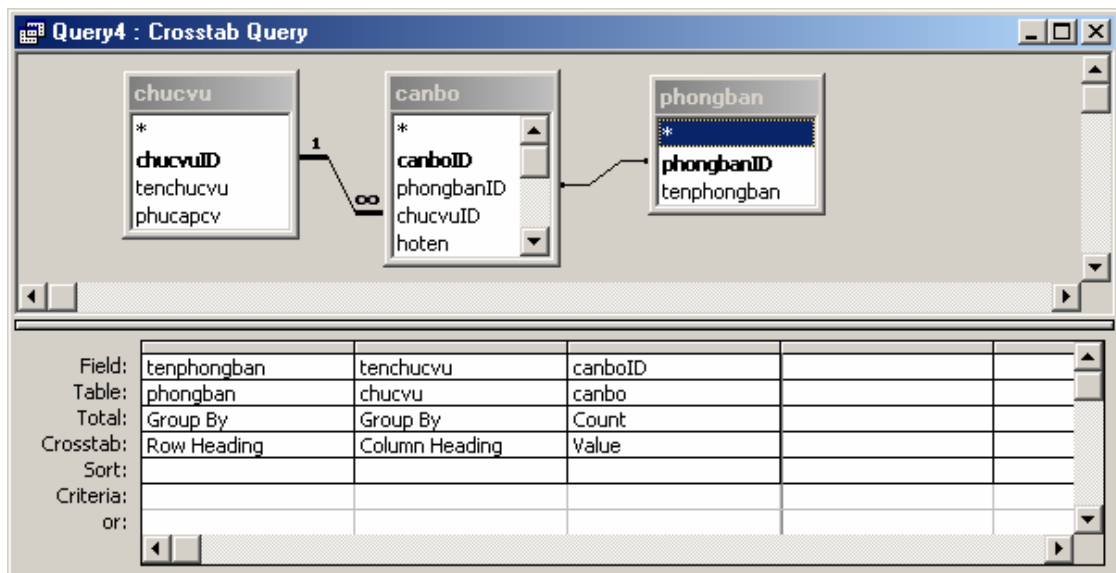
Record: 1 of 2

**Bước 1:** Tạo một **select query** với đầy đủ các trường có liên quan đến phép tổng hợp (không thừa và cũng không thể thiếu một trường nào) như sau:



**Bước 2:** Ra lệnh chọn kiểu **Crosstab query** bằng cách mở thực đơn **Queries | Crosstab Query**;

**Bước 3:** Thiết lập các thuộc tính **Total** cũng như **Crosstab** cho các trường phù hợp như sau:



Cụ thể:

Tên trường	Thuộc tính Total	Thuộc tính Crosstab
Tenphongban	<i>Group By</i>	<i>Row Heading</i>
Tenchucvu	<i>Group By</i>	<i>Column Heading</i>
canboID	<i>Count</i>	<i>Value</i>



Bản chất của Crosstab query thể hiện ở câu lệnh SQL:-----

**TRANSFORM .. SELECT .. FROM .. GROUP BY ... PIVOT ...**

#### 4. MAKE TABLE queries

Select và Total query luôn đưa ra một bảng kết quả- đó là giá trị tức thời, mới nhất được lấy ra từ CSDL tại thời điểm đó. Tại thời điểm khác khi thi hành query đó, rất có thể chúng ta không thu lại được bảng kết quả như thời điểm trước đó. Muốn lưu lại bảng kết quả của một query tại một thời điểm nào đó, trong Access có một cách là đưa dữ liệu kết quả query ra một bảng (Table) để lưu trữ lâu dài bởi một MAKE TABLE query.

Các bước tạo một **Make table** query để lưu trữ kết quả một query ra một bảng dữ liệu được tiến hành theo 2 bước chính:

**Bước 1:** Tạo một **Select query** hoặc một **Total query** đưa ra được bảng kết quả cần lưu trữ;

**Bước 2:** Ra lệnh tạo **Make table** query từ query đang thiết kế và thi hành để thu được bảng kết quả như mong muốn.

Giả sử đã tạo được một Select query đưa ra được bảng kết quả như sau:

	hoten	ngaysinh	tenchucvu	tenphongban	luongchin	congdoanph	dangphi	thuclich
▶	Trần Văn Ngô		Nhân viên	Tổng hợp	1740000	0	20000	1920000
	Nguyễn Đức Ngô	12/04/1976	Giám đốc	Kinh doanh	904800	45240	20000	1339560
	Phan Anh Quân	23/07/1978	Trưởng phòng	Kinh doanh	611900	30595	20000	861305
	Nguyễn Hải Hà	13/09/1979	Phó giám đốc	Tổng hợp	611900	30595	0	981305
	Nguyễn Thu Loan	05/02/1978	Nhân viên	Tổng hợp	516200	25810	20000	670390
	Hà Anh Tú	31/12/1975	Nhân viên	Tổng hợp	516200	0	0	716200
	Phan Quang Sáng	03/09/1976	Nhân viên	Kinh doanh	696000	34800	0	861200
	Nguyễn Ánh Tuyết	02/07/1980	Trưởng phòng	Tổng hợp	696000	34800	20000	941200
	Nguyễn Văn Hoà		Nhân viên	Kinh doanh	580000	29000	0	751000
	Phan Hoài Linh		Nhân viên	Kinh doanh	580000	29000	0	751000
	Tạ Quang Luận		Nhân viên	Tổng hợp	1160000	58000	20000	1282000
	Mai Vũ Hằng		Nhân viên	Tổng hợp	870000	0	0	1070000
*								

Record: 1 of 12

Bài toán đặt ra là: đưa toàn bộ kết quả của query đang hiển thị ra một bảng mới có tên **bangluong**. Cách làm như sau:


**Bước 1:** mở select query đã tạo được ở chế độ **Design View**;

**Bước 2:** ra lệnh tạo Make table query bằng cách mở thực đơn **Queries | Make Table query**, hộp thoại Make table xuất hiện:

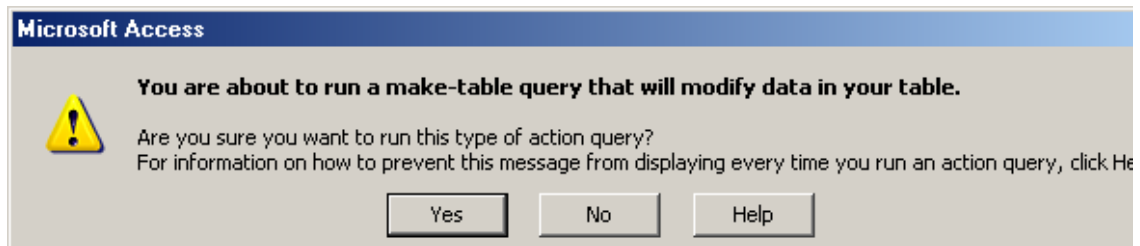
Hãy nhập vào tên bảng dữ liệu cần lưu vào ô **Table Name**: Chú ý:

- Nếu tên bảng nhập vào là mới, **Access** sẽ tự động tạo một bảng mới và sao chép toàn bộ dữ liệu mà query kết xuất được ra bảng này;
- Nếu tên nhập vào trùng một tên bảng đã có sẵn, khi thi hành **Access** sẽ xoá bảng cũ và điền vào dữ liệu mới (cần cân nhắc khi đặt tên bảng trùng tên bảng đã tồn tại).

**Bước 3:** thi hành query để nhận kết quả bằng cách:

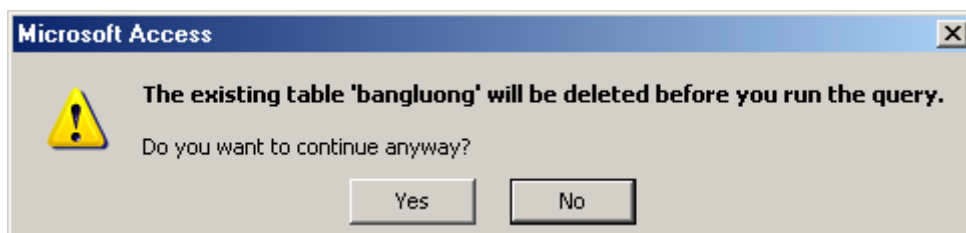
- Nếu query đang ở chế độ thiết kế, nhấn nút Run  trên thanh công cụ;
- Hoặc nháy đúp chuột lên query cần thực hiện.

Khi đó một hộp thoại cảnh báo việc bạn đang ra lệnh thi hành một query có thể làm thay đổi đến dữ liệu trên CSDL:



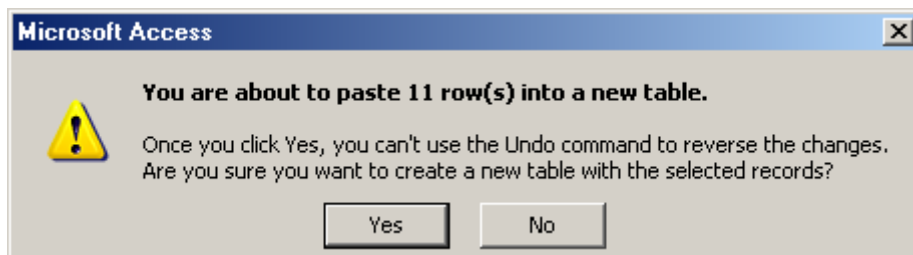
Nhấn **Yes** để tiếp tục hoặc nhấn **No** để huỷ bỏ lệnh.

Nếu trên CSDL đã tồn tại một bảng có cùng tên bảng bạn đã chỉ định cho query này lưu dữ liệu, Access sẽ hỏi bạn: *Đã tồn tại bảng xxx trên CSDL rồi, nó sẽ bị xoá sạch khi query này thi hành, bạn có muốn tiếp tục không?*



Nhấn **Yes** để tiếp tục (tất nhiên bảng dữ liệu đó sẽ bị xoá và thay vào nội dung mới); nhấn **No** để huỷ bỏ.

Cuối cùng một hộp thoại hỏi một lần cuối xem *bạn có đồng ý dán xxx bản ghi vào bảng đã chỉ định hay không?*



Nhấn **Yes** để đồng ý, **No** để huỷ bỏ.

Thi hành xong hãy mở bảng vừa tạo được để kiểm tra kết quả.



Bản chất của Crosstab query thể hiện ở câu lệnh SQL:-----

**SELECT .. INTO .. FROM..**

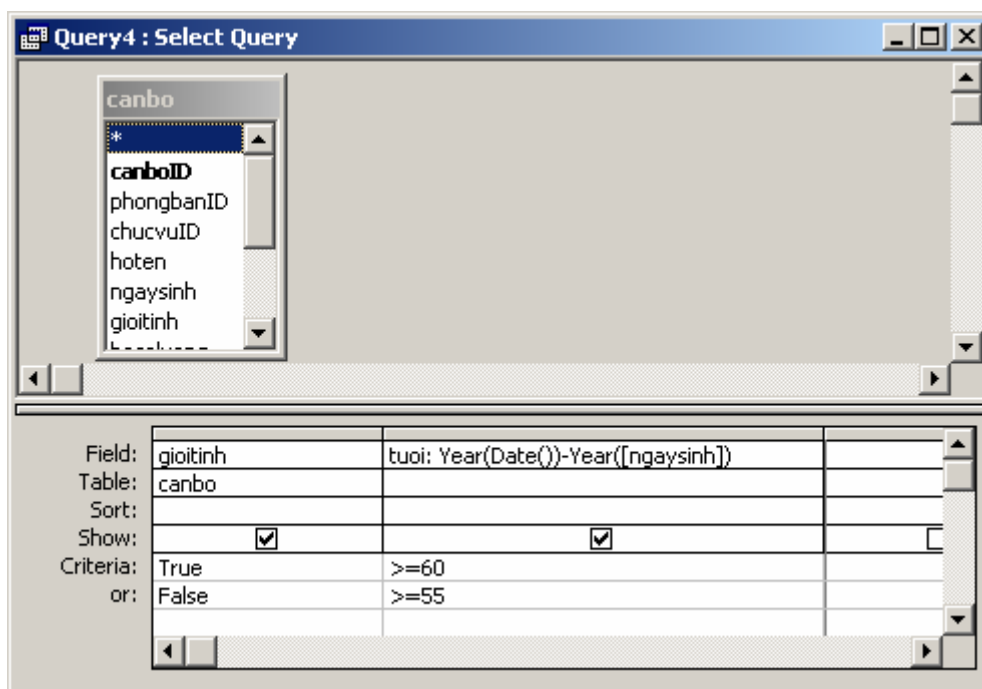
-----

## 5. DELETE queries

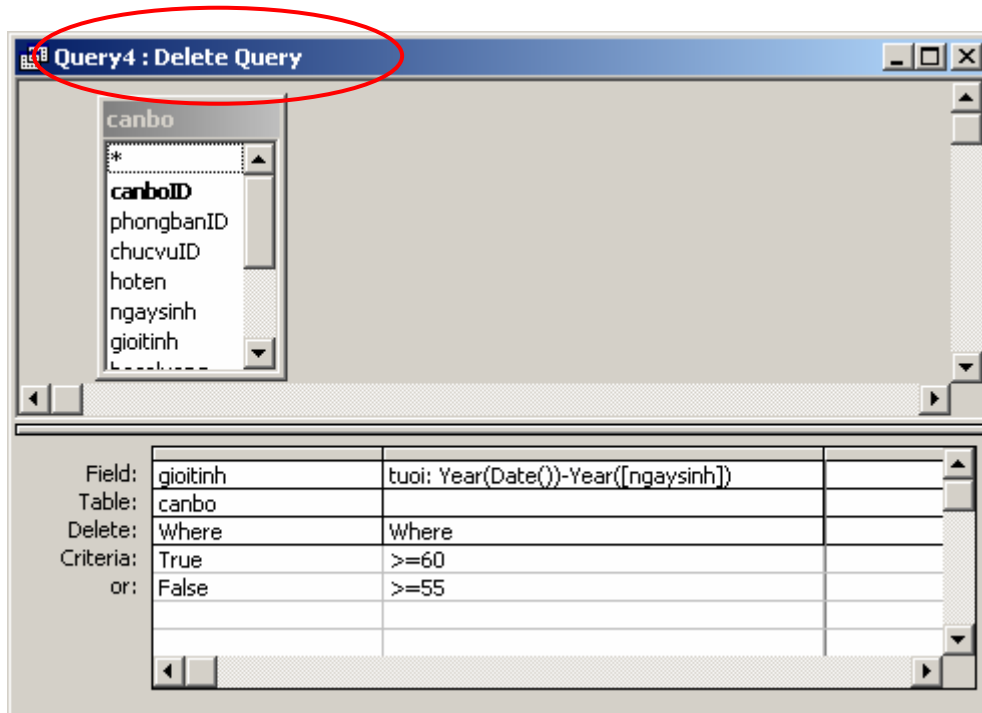
**Delete Query** là một loại **Action Query** (query hành động). Nó có thể gây thay đổi dữ liệu trong CSDL. Trong trường hợp này, Delete query dùng để xoá các bản ghi từ CSDL thoả mãn những điều kiện nào đó.


Dưới đây là hướng dẫn tạo một Delete query xoá đi những cán bộ đến tuổi nghỉ hưu ra khỏi CSDL Quản lý lương cán bộ. Khi query này thi hành, danh sách các cán bộ đến tuổi hưu sẽ bị xoá khỏi bảng *canbo*. Cách tạo query này như sau:

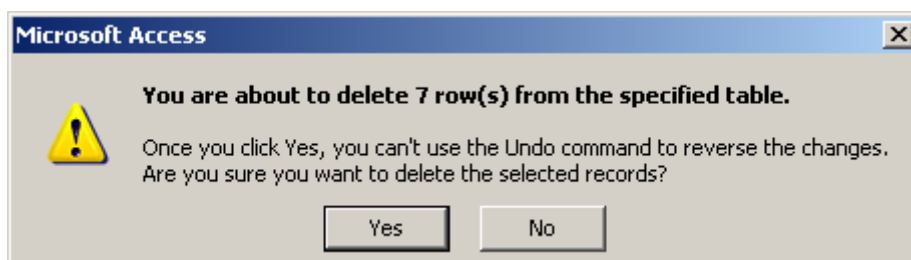
**Bước 1:** Tạo một Select query như sau:



**Bước 2:** đổi query hiện hành thành Delete query bằng cách mở thực đơn **Queries | Delete Query**



Để thi hành việc xoá cán bộ, nhấn nút **Run**  trên thanh công cụ. Khi đó một hộp thoại cảnh báo xuất hiện:



Nhấn **Yes** để đồng ý xoá đi các bản ghi (bản ghi đã xoá không thể phục hồi lại được); nhấn **No** để huỷ bỏ lệnh.

Phải cẩn trọng trước khi quyết định lệnh xoá dữ liệu.



Bản chất của Delete query thể hiện ở câu lệnh SQL:-----

**DELETE ... FROM ... [WHERE]**

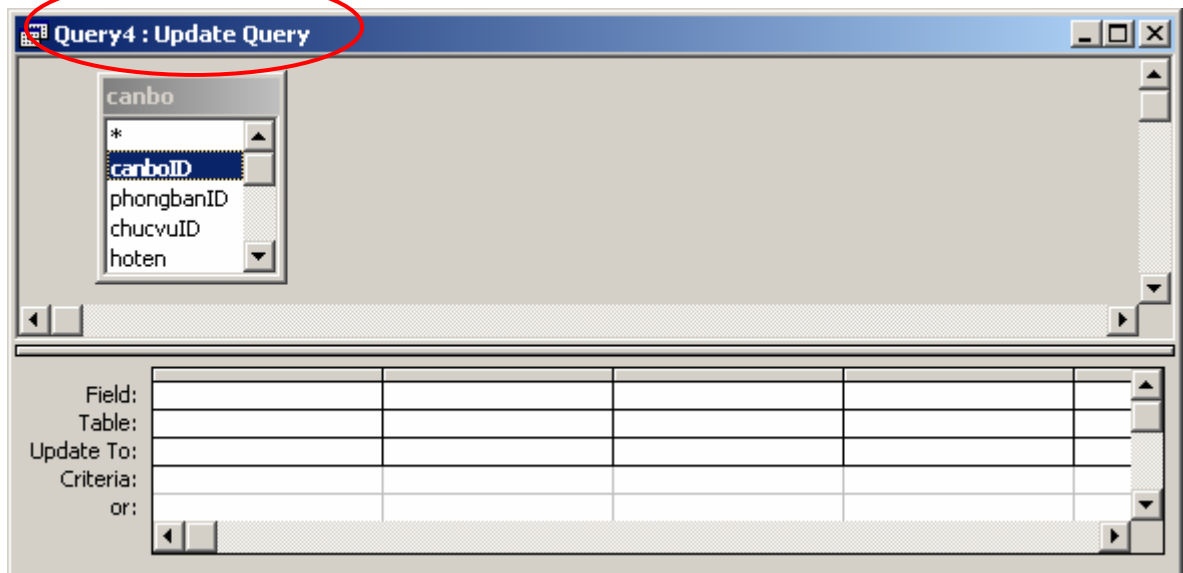
## 6. UPDATE queries

Update query dùng cập nhật dữ liệu một số trường nào đó trong CSDL. Giống với Delete query, Update query là loại query hành động, làm thay đổi nội dung dữ liệu trên CSDL.



Dưới đây là một ví dụ sử dụng **Update query** để tính giá trị cho cột *luongchinh* (lương chính) là một trường mới được thêm vào bảng *canbo*.

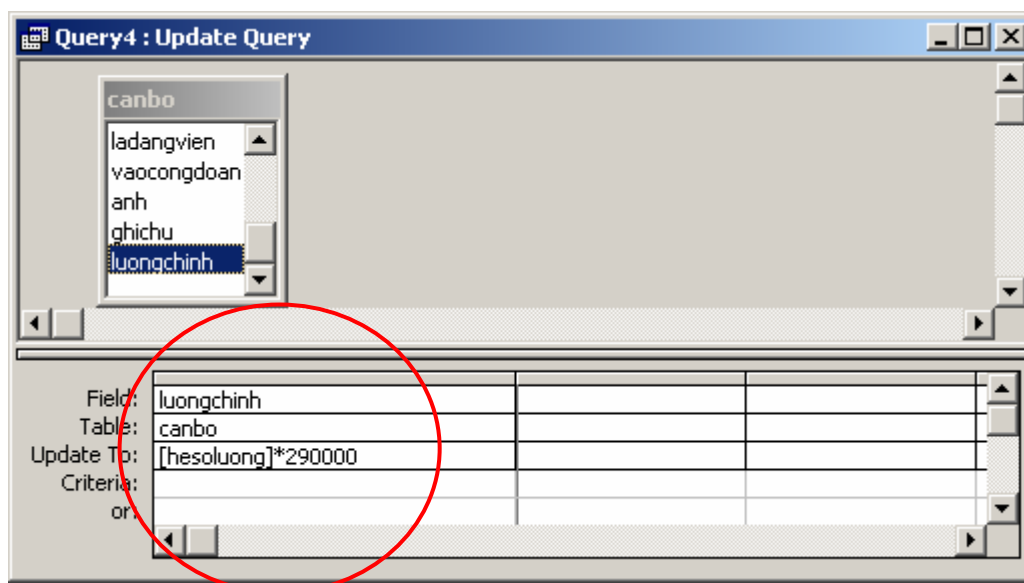
**Bước 1:** Tạo một query và có chứa bảng *canbo* và chuyển thành Update query bằng cách mở thực đơn **Queries | Update query**; hộp thoại thiết kế query như sau:




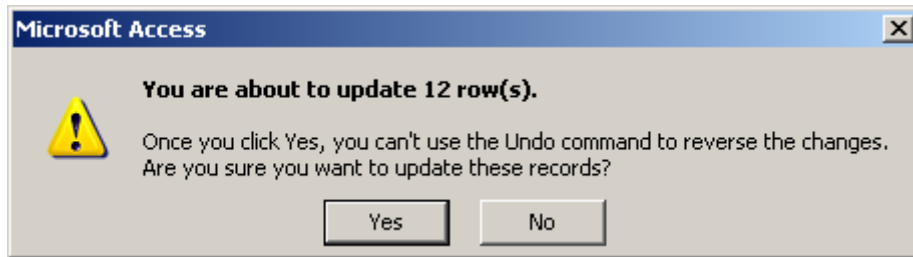
**Bước 2:** Thiết lập các trường cần cập nhật dữ liệu bằng cách:

- Chọn tên trường cần cập nhật dữ liệu ở dòng **Field**;
- Gõ vào biểu thức tính giá trị cho trường đó ở dòng **Update To**;
- Gõ vào điều kiện để tính toán nếu cần ở dòng **Criteria**.

Với yêu cầu trên, **luongchinh = hesoluong \* 290000** của sổ thiết kế query sẽ như sau:



Để thi hành việc xoá cán bộ, nhấn nút *Run*  trên thanh công cụ. Khi đó một hộp thoại cảnh báo xuất hiện:



Nhấn **Yes** để đồng ý cập nhật dữ liệu (dữ liệu sau khi đã cập nhật không thể phục hồi lại được); nhấn **No** để huỷ bỏ lệnh.

Phải cẩn trọng trước khi quyết định lệnh này.



Bản chất của Update query thể hiện ở câu lệnh SQL:-----

**UPDATE ... SET ... [WHERE]**

-----

## Bài tập

### Thực hiện trên CSDL Quản lý lương cán bộ các yêu cầu:

**Bài số 1:** Hãy tính và in ra bảng lương của cơ quan bao gồm các thông tin:

*Hoten, ngaysinh, tenchucvu, tenphongban, luongchinh, phucapcv, dangphi, congdoanphi, thuclinh.*

Trong đó:

**Luongchinh = hesoluong \* 290000**

**Dangphi = 20,000** với những ai là Đảng viên

**Congdoanphi = 5%Luongchinh** với những ai đã vào công đoàn

**Bài số 2:** Hãy đưa ra danh sách các cán bộ là Đảng viên mà chưa vào công đoàn;

**Bài số 3:** Hãy lọc ra danh sách cán bộ của một phòng ban nào đó, tên phòng ban được nhập từ bàn phím khi query được kích hoạt;

**Bài số 4:** Hãy lọc ra danh sách cán bộ là Đảng viên và có thực lĩnh lớn hơn một số nào đó- được nhập từ bàn phím khi query được kích hoạt;

**Bài số 5:** Hãy lọc ra danh sách cán bộ mà năm sinh rơi vào trong một khoảng năm nào đó được nhập vào từ bàn phím. Ví dụ: khi kích hoạt query, Access yêu cầu nhập vào khoảng năm sinh từ năm nào, đến năm nào. Khi đó query sẽ lọc ra danh sách các cán bộ thoả mãn điều kiện đó;

**Bài số 6:** Lọc ra danh sách cán bộ có tên được nhập vào từ bàn phím khi kích hoạt query. Ví dụ: khi query kích hoạt sẽ hỏi: **Gõ vào tên cần xem?** Sau khi gõ vào

một tên (có thể cả họ hoặc cả họ và tên), máy tính sẽ lọc ra danh sách cán bộ thỏa mãn điều kiện;

**Bài số 7:** Lọc ra danh sách cán bộ đến tuổi nghỉ hưu, biết rằng điều kiện nghỉ hưu xác định như sau:

- Là Nam giới, tuổi  $\geq 60$
- Là Nữ giới, tuổi  $\geq 55$

**Bài số 8:** Hãy đưa ra bảng tổng hợp số cán bộ đã vào Đảng hay chưa vào Đảng theo giới tính như sau:

	Giới tính	Chưa vào Đảng	Là Đảng viên
▶	Nam	3	4
	Nữ	3	1

**Bài số 9:** Hãy đưa ra bảng tổng hợp tổng số cán bộ theo từng loại chức vụ của các phòng ban như sau:

	tenphongban	Tổng số CB	Giám đốc	Nhân viên	Phó giám đốc	Trưởng phòng
▶	Kinh doanh	5	1	3		1
	Tổng hợp	6		4	1	1

**Bài số 10:** Đưa ra bảng tổng hợp về chức vụ theo giới tính như sau:

	Chức vụ	Tổng số CB	Nam	Nữ
▶	Giám đốc	1	1	
	Nhân viên	7	4	3
	Phó giám đốc	1	1	
	Trưởng phòng	2	1	1

**Bài số 11:** Đưa ra bảng tổng hợp lương theo từng phòng ban như sau:

tenphongban	TSCB	Tong LC	Tổng CĐP	Tổng Đảng phí	Tổng TL
Kinh doanh	5	3372700	168635	40000	4564065
Tổng hợp	6	4370300	149205	60000	5661095

**Thực hiện trên CSDL Quản lý bán hàng các yêu cầu:**

**Bài số 12:** Tạo query hiển thị chi tiết thông tin từng hàng bán:

ngayban	hoadonID	tenhang	soluong	dongia	thanhtien
12/10/2004	1	Floppy disk	3	8	24
12/10/2004	1	Hard disk	9	67	603
12/10/2004	1	CPU P4 2.4	2	133	266
24/10/2004	2	RAM 256	3	38	114
10/10/2004	3	RAM 256	12	38	456
10/10/2004	3	CPU P4 2.4	3	133	399
12/10/2004	4	Floppy disk	8	8	64
12/10/2004	4	CPU P4 2.4	10	133	1330

**Bài số 13:** Tạo query đưa ra bảng tổng hợp tổng tiền bán được từng ngày của cửa hàng như sau:

ngayban	Tongtien
10/10/2004	855
12/10/2004	2287
24/10/2004	114

**Bài số 14:** Tạo query đưa ra bảng tổng hợp tổng số bán và tổng tiền đã bán ra của từng loại mặt hàng như sau:

tenhang	Tong so ban	Tongtien
CPU P4 2.4	15	1995
Floppy disk	11	88
Hard disk	9	603
RAM 256	15	570

**Bài số 15:** Tạo bảng tính tổng tiền từng hoá đơn bán hàng như sau:

	hoadonID	ngayban	tenkhach	Tongtien
▶	1	12/10/2004	Nguyễn Đức Ngô	893
	2	24/10/2004	Nguyễn Đức Ngô	114
	3	10/10/2004	Phan Anh Tài	855
	4	12/10/2004	Nguyễn Đức Sơn	1394

**Bài số 16:** Lập bảng tổng hợp số lượng bán ra từng ngày của từng mặt hàng như sau:

tenhang	10/10/2004	12/10/2004	24/10/2004
▶ CPU P4 2.4	3	12	
Floppy disk		11	
Hard disk		9	
RAM 256	12		3

**Bài số 17:** Lập bảng tổng hợp số lượng mua từng mặt hàng của các khách hàng như sau:

tenkhach	10/10/2004	12/10/2004	24/10/2004
▶ Nguyễn Đức Ngô		893	114
Nguyễn Đức Sơn		1394	
Phan Anh Tài	855		

# CHƯƠNG 3

## THIẾT KẾ GIAO DIỆN

Chương này trình bày về FORMs - một công cụ thiết kế giao diện người sử dụng phần mềm- bộ phận giao tiếp giữa người dùng với ứng dụng.

Toàn chương sẽ đề cập đến các mảng kiến thức:

- Các khái niệm về Forms;
- Sử dụng Form wizard;
- Sử dụng Form design view;
- Kỹ thuật Sub-form

Sau khi học xong chương này, học viên sẽ:

- Nắm được các khái niệm về Form, về thiết kế giao diện, về lập trình trực quan, hướng đối tượng;
- Qua đó, học viên hoàn toàn có những kỹ năng tốt để có thể tạo ra được các dạng Form nhập dữ liệu; form điều khiển, xử lý dữ liệu theo những yêu cầu của bài toán thực tế đặt ra. Tuy nhiên, để có một khả năng phát triển chuyên nghiệp hơn, học viên cần chuyên sâu đến phần 2 của giáo trình này nói về Lập trình CSDL.

## 1. Khái niệm Forms

Khi sử dụng một ứng dụng, đa phần công việc của người dùng làm trên các hộp thoại (Dialogue), cửa sổ (Windows). Cả 2 thành phần này trong lập trình đều được gọi là **Form**. Với người dùng, **Form** là giao diện để sử dụng phần mềm; còn với những người phát triển phần mềm, **Form** là những cái mà họ phải nghĩ, phải thiết kế và tạo ra sao cho người dùng họ cảm thấy rất thoải mái, phù hợp và dễ dùng.

Có 2 môi trường dùng tạo Form trong Access:

- Sử dụng trình **Form Wizard**. Đây là cách rất đơn giản, nhanh chóng, dễ dùng giúp tạo nhanh một Form. Tuy nhiên Form được tạo ra có nhiều hạn chế, không gần gũi với yêu cầu của người dùng;
- Sử dụng trình **Form Design View** - một công cụ tương đối hoàn chỉnh để tạo ra các form đáp ứng nhiều yêu cầu khác nhau của người sử dụng.

Tiếp theo sẽ giới thiệu cả 2 môi trường này.

## 2. Sử dụng FORM WIZARD

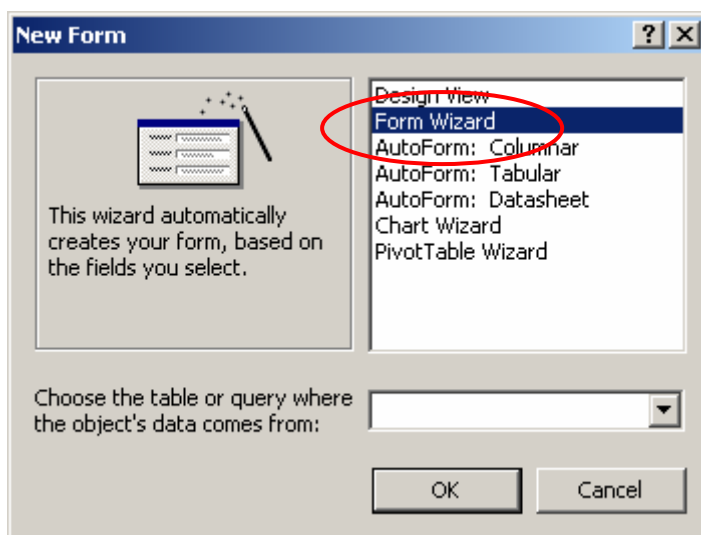
Form wizard là công cụ trên Access giúp người dùng nhanh chóng và dễ dàng tạo ra một Form dùng làm giao diện nhập dữ liệu cho các bảng trên CSDL.

Dưới đây trình bày các bước để tạo một form làm giao diện nhập dữ liệu cho bảng *CANBO* trong CSDL Quản lý lương cán bộ. (bạn phải chắc chắn đã tạo hoàn chỉnh cấu trúc CSDL trên):

**Bước 1:** Kích hoạt trình Form Wizard

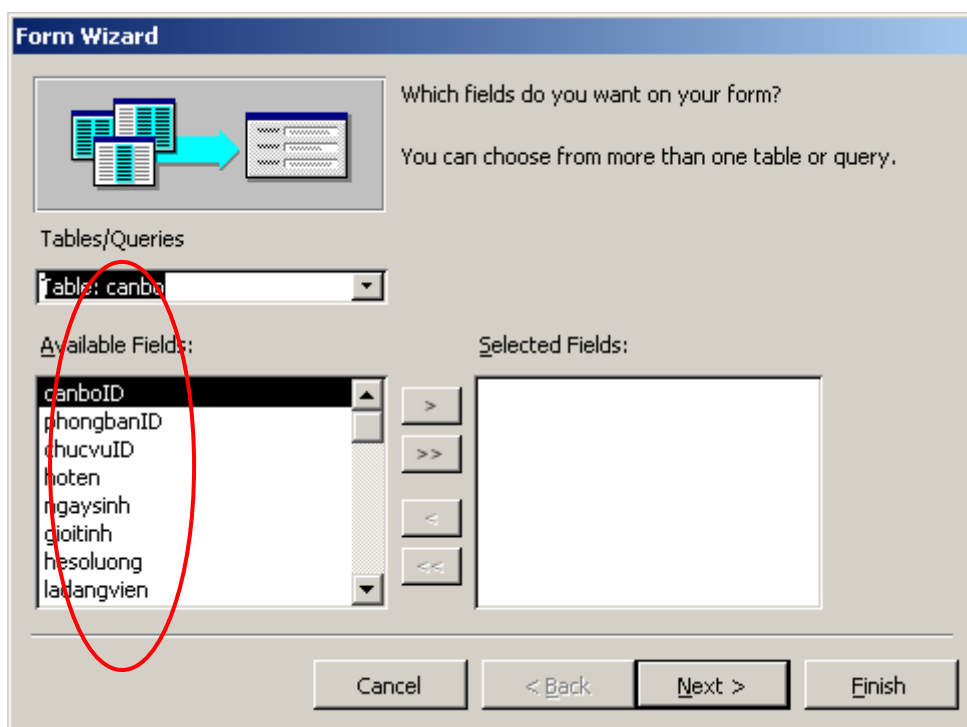
Ở thẻ Forms, nhấn nút **New**, chọn **Form Wizard**, nhấn **OK**;





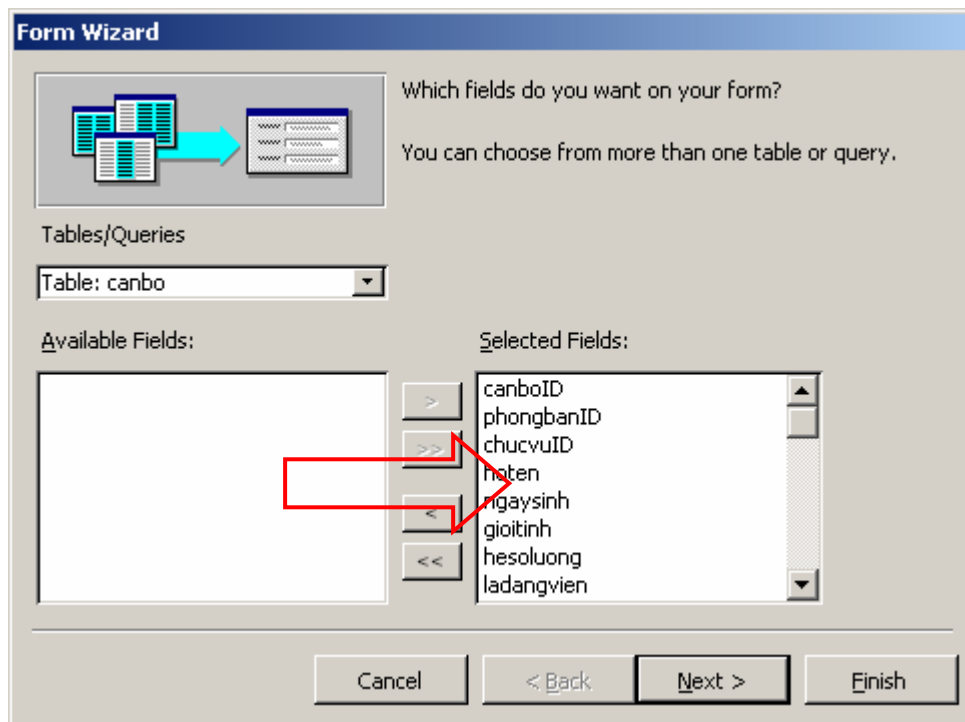
Hoặc có thể nhấn đúp chuột lên Shortcut **Creat Form by using Form wizard**

**Bước 2:** Chọn các thông tin cần đưa lên form. Đây là bước quan trọng nhất để khai báo những trường dữ liệu nào của form sẽ có thể được nhập dữ liệu thông qua form đang tạo. Học viên phải cực kỳ cẩn nhắc các thông tin ở đây: chọn bảng nào và chọn những trường nào cho đúng?



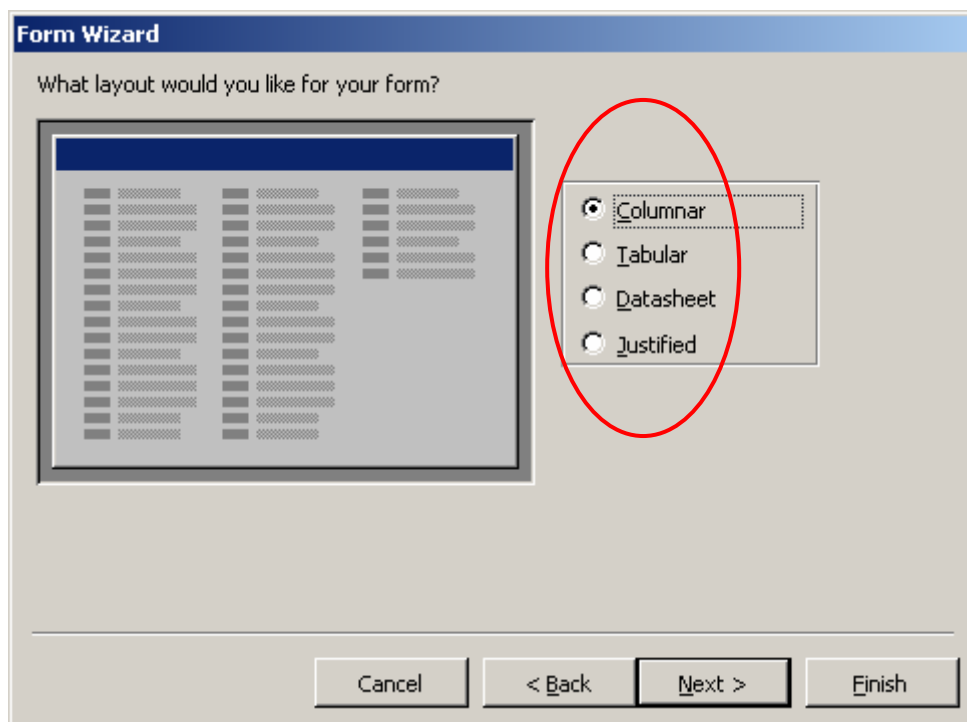
- Chọn bảng - nơi có trường cần đưa lên form để nhập dữ liệu ở hộp **Tables/Queries**;
- Tiếp theo đưa các trường cần nhập dữ liệu lên form từ danh sách **Available Fields**: sang danh sách **Selected Fields**: bằng các nút lệnh: >, >>, <, <<;

- Hãy làm lần lượt đến khi chọn được đủ các trường cần nhập dữ liệu lên form.



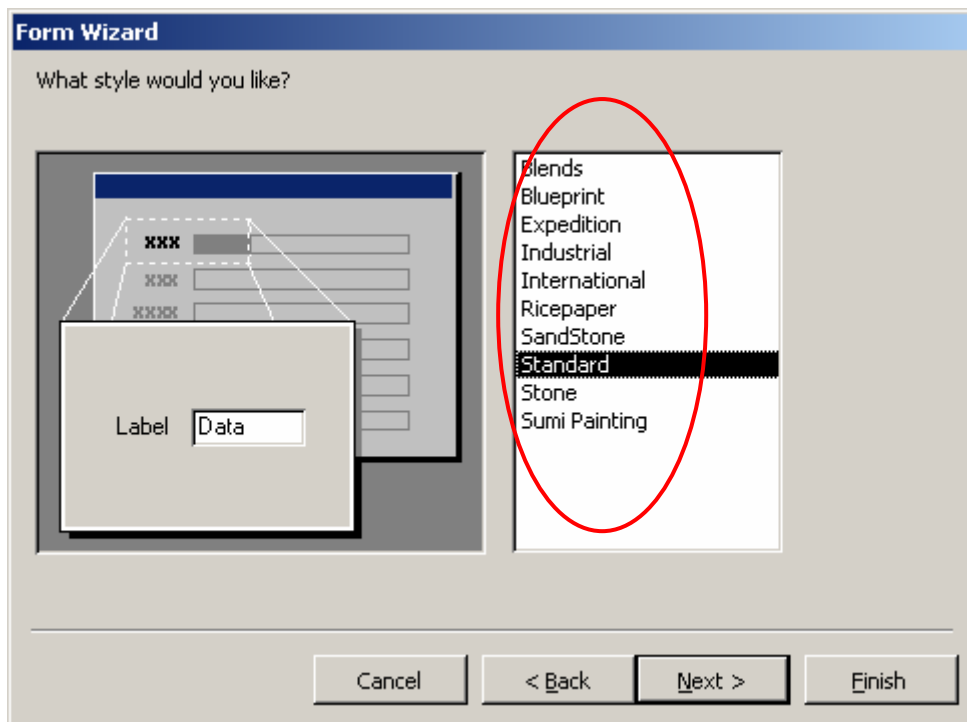
Để tiếp tục nhấn **Next**;

**Bước 3:** Chọn bố cục (Layout) cho form. Có 4 kiểu bố cục có thể thiết lập được cho form là: *Columnar*; *Tabular*; *Datasheet* và *Justified*. Hãy lần lượt nhấn lên từng kiểu bố cục (hình dưới) và xem kết quả minh họa (Preview) ở hộp chữ nhật bên trái hộp thoại để quyết định nên chọn loại nào:



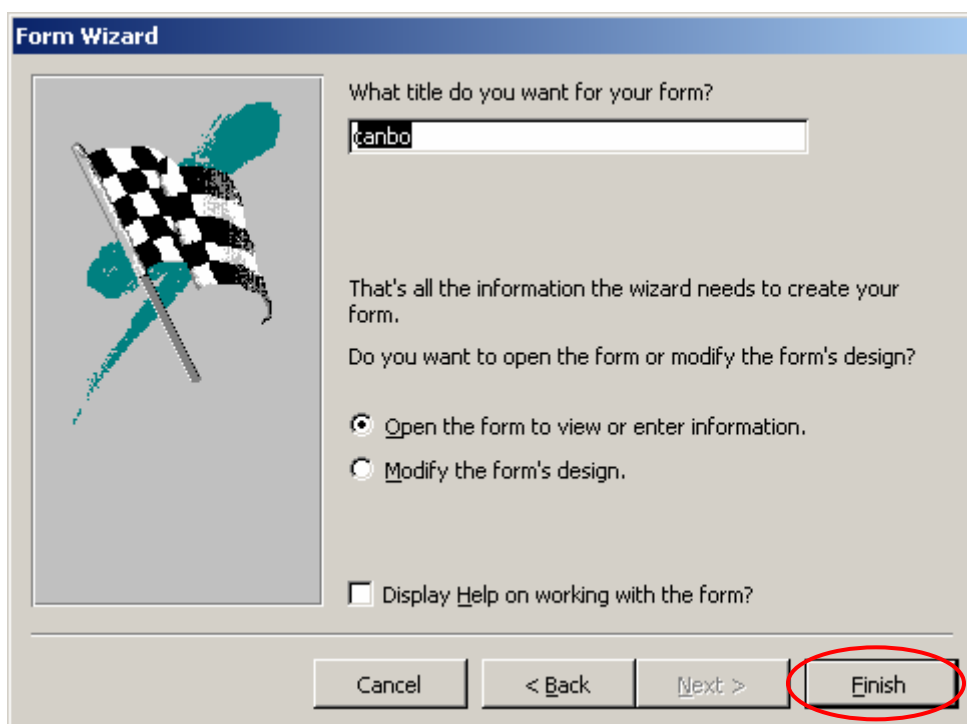
Chọn xong nhấn **Next** để tiếp tục;

**Bước 4:** Chọn kiểu dáng (Style) cho form. Có một danh sách các kiểu dáng để chọn. Hãy thử từng kiểu dáng và chọn cho form một phong cách phù hợp:



Chọn xong nhấn **Next**:

**Bước 5:** Hoàn thiện công việc:




- Bạn có thể gõ vào tên gọi cũng như tiêu đề cho form ở hộp **What title do you want for your form?**
- Có thể tùy chọn kích hoạt ngay form vừa tạo nếu chọn **Open the form to view or enter information** hoặc mở form ra ở chế độ Design view để sửa cấu trúc nếu chọn mục **Modify the form's design**;
- Nhấn **Finish** để hoàn tất toàn bộ công việc.




### Chú ý





Quan trọng nhất là bước 1, 2. Từ bước 3 trở đi có thể bỏ qua bằng cách nhấn **Finish** ngay từ bước 2 nếu đảm bảo toàn bộ các thiết lập từ bước 3, 4, 5 là phù hợp.

### Sử dụng form đã tạo được vào việc nhập dữ liệu:

Chọn form, nhấn nút **Open**. Nếu form đang ở chế độ thiết kế có thể nhấn nút View  trên thanh công cụ. Lúc này có thể sử dụng form để cập nhật dữ liệu:

Tại mỗi thời điểm, form nhập dữ liệu chỉ hiển thị giá trị của một bản ghi. Bạn có thể nhập, sửa trực tiếp các trường của bản ghi hiện tại trên form này. Thanh định hướng (Navigator bar) sẽ giúp xử lý một số thao tác trên form, cách sử dụng như sau:

- Nút  để chuyển đến bản ghi kế sau;

- Nút  để chuyển đến bản ghi kê trước;
- Nút  để chuyển về bản ghi đầu tiên;
- Nút  để chuyển đến bản ghi cuối cùng;
- Nút  để thêm một bản ghi mới

### 3. Sử dụng FORM DESIGN VIEW


Để tạo ra được form mang tính chuyên nghiệp và đáp ứng được sát yêu cầu thực tế bắt buộc phải sử dụng đến **Form Design View**. Nói như vậy không có nghĩa là không dùng Form Wizard để tạo form, mà trong nhiều tính huống vẫn dùng Form Wizard rồi sử dụng đến Form Design View để tiếp tục hoàn thiện yêu cầu.

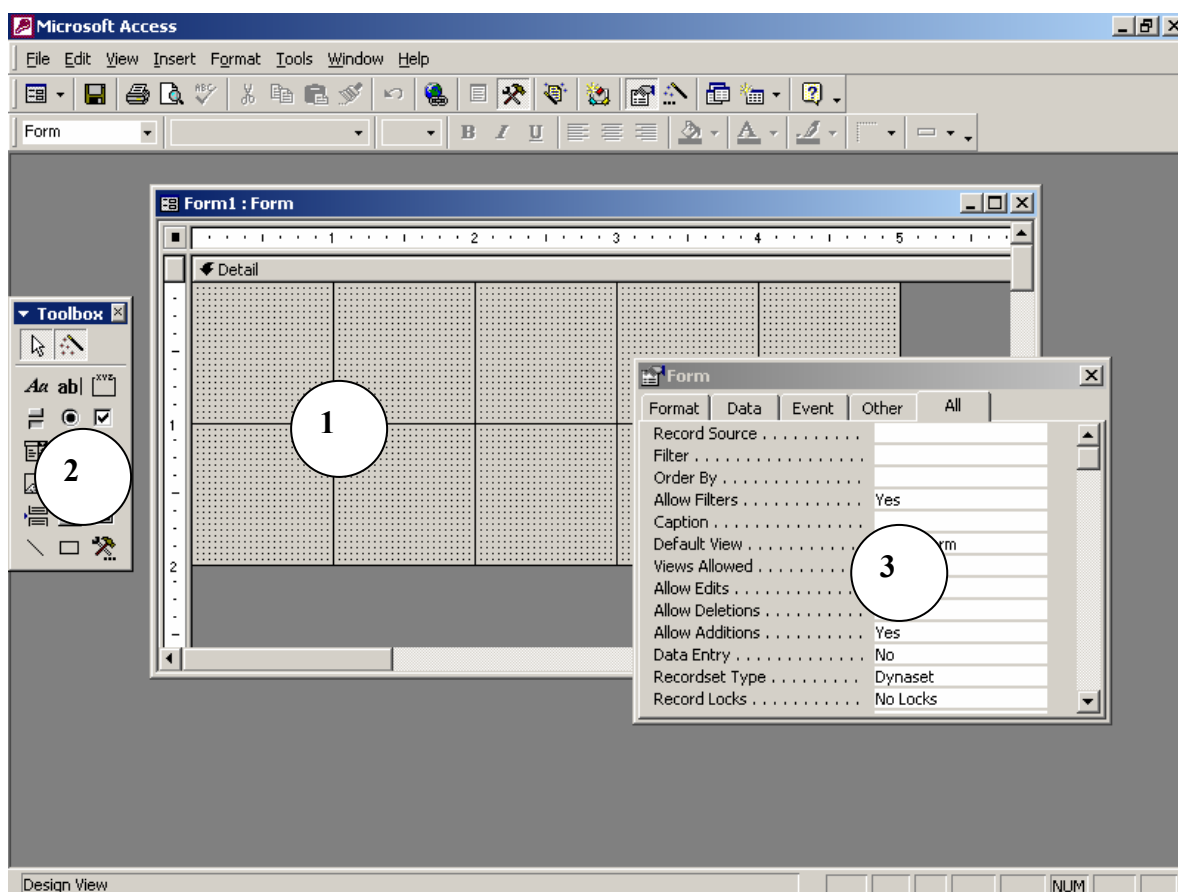
Nội dung mục này trình bày môi trường thiết kế form cũng như cách tạo một số kiểu form từ đơn giản đến phức tạp sử dụng Form Design View.

#### 3.1 Thiết kế Form nhập dữ liệu đơn giản

Chúng ta sẽ tìm hiểu cách thiết kế một form nhập dữ liệu đơn giản thông qua yêu cầu: tạo form dùng làm mẫu nhập dữ liệu cho bảng CANBO.

##### **Bước 1:** Ở thẻ Form, nhấn **New**, chọn **Design View**

Hoặc nhấn biểu tượng  **Create form in Design view** trên cửa sổ QLL Database. Môi trường thiết kế form xuất hiện:



Có 3 thành phần quan trọng để làm việc:

- (1) Cửa sổ **Form** – nơi sẽ thiết kế và xây dựng các thông tin cần thiết theo yêu cầu bài toán. Cấu trúc form gồm 3 phần:

*Form Header* - phần tiêu đề đầu form;

*Form Footer* - phần tiêu đề cuối;

*Detail* - phần thân form..

Toàn bộ các thông tin trên form đều được chứa trong các đối tượng điều khiển (Control), các đối tượng này được lấy từ thanh công cụ Toolbox sẽ trình bày phân tiếp sau.

- (2) Thanh công cụ **Toolbox** – nơi chứa những đối tượng, những công cụ có thể đưa lên form với mục đích thiết kế giao diện và điều khiển dữ liệu theo bài toán. Ví dụ: muốn tạo ô nhập Họ tên trên form có thể dùng đối tượng Textbox, muốn đưa một chú thích (nhãn hiển thị) có thể dùng Label, muốn tạo một nút lệnh có thể dùng CommandButton,...

Mỗi đối tượng sẽ có tập hợp các thuộc tính (Properties) và tập các sự kiện (Events). Thuộc tính để mô tả tính chất cho đối tượng đó, ví dụ như: màu sắc, kích thước, tính chất dữ liệu,.. Sự kiện- nơi có thể gắn các mã lệnh VBA hoặc gắn các Macro lệnh để xử lý những công việc nào đó. Chúng ta sẽ tìm hiểu kỹ việc sử dụng các sự kiện trong phần lập trình VBA.

- (3) Cửa sổ **Properties** – nơi có thể thiết lập các thuộc tính (properties) cho form cũng như các đối tượng trên form;

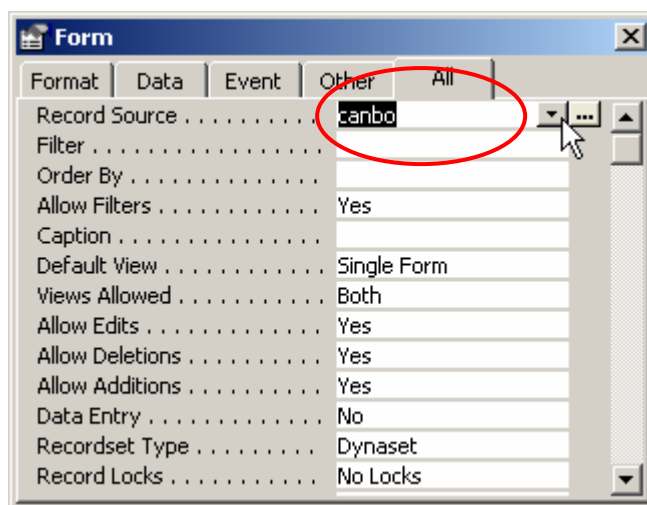
**Bước 2:** Thiết lập nguồn dữ liệu cho form ở thuộc tính Record Source.

Form đang thiết kế là loại để nhập dữ liệu, bước này để xác định nguồn dữ liệu để form làm việc. Vì chỉ để nhập dữ liệu cho bảng CANBO, nên nguồn dữ liệu sẽ là bảng cán bộ. Cách thiết lập thuộc tính này như sau:

- Chọn thuộc tính form bằng cách chọn tên đối tượng *Form* ở hộp chọn **Object** trên thanh công cụ **Formatting**:

Hoặc nhấn chuột lên ô vuông- vị trí giao giữa 2 thước kẻ ngang-dọc của form đang thiết kế. Làm sao khi tiêu đề cửa sổ Properties là *Form* là ok.

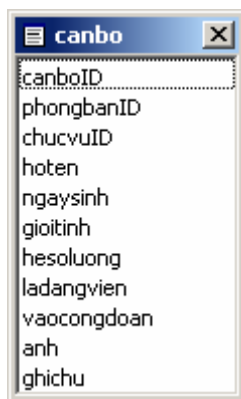
- Thiết lập thuộc tính **Record Source** cho form bằng cách chọn tên bảng *CANBO* ở hộp **Record Source**. Có thể tìm thuộc tính này ở thẻ **Data** - chỉ những thuộc tính liên quan đến dữ liệu; hoặc thẻ **All**- có đầy đủ tất cả các thuộc tính và sự kiện:



**Bước 3:** Mở cửa sổ Field List.

Cửa sổ **Field List** có chứa danh sách các trường trên CSDL có trong nguồn dữ liệu của Form. Nó hỗ trợ việc đưa những trường dữ liệu này lên form để nhập và hiển thị dữ liệu rất tốt. Trong trường hợp này ta dùng để đưa những trường cần nhập dữ liệu từ bảng CANBO lên form.

Nếu chưa thấy cửa sổ này xuất hiện, hãy thực hiện hiển thị nó bằng cách mở thực đơn **View | Field List** hoặc nhấn nút *Field List* trên thanh công cụ chuẩn.



**Bước 4:** Đưa những trường cần nhập dữ liệu từ cửa sổ Field List lên Form đang thiết kế bằng cách: Dùng chuột kéo từng trường muốn thiết kế lên form từ cửa sổ Field List thả lên vị trí hợp lý trên form (với bài này phải kéo toàn bộ các trường lên form). Lúc này cửa sổ thiết kế form có dạng:

Mỗi khi kéo một trường từ Field List lên form, Access sẽ tự động tạo một đối tượng gắn kết tới trường dữ liệu tương ứng, đối tượng này có thể là Textbox, Combobox hay đối tượng khác tùy thuộc vào kiểu dữ liệu của trường tương ứng; và đối tượng Label đi kèm nhằm tạo nhãn chú thích cho trường dữ liệu.



Đến đây đã tạo xong form nhập dữ liệu đơn giản cho một bảng.

### 3.2 Tinh chỉnh cấu trúc Form

Khi thiết kế một form, đòi hỏi lập trình viên không những phải tạo ra được form đáp ứng đầy đủ các yêu cầu về xử lý dữ liệu của bài toán, mà còn phải đáp ứng tính thẩm mỹ. Thế nào là form có thẩm mỹ cao? Câu hỏi này rất khó! Tùy thuộc vào bài toán, vào đối tượng người sử dụng mà thiết kế những kiểu giao diện thân thiện. Thanh công cụ Toolbox và cửa sổ Properties là những công cụ cần thiết để thiết kế giao diện.

#### a. Sửa thuộc tính

Mở form để sửa ở chế độ thiết kế (Design view) bằng cách: chọn form, nhấn nút **Design**; hoặc nhấn nút *Design* trên thanh công cụ. Màn hình thiết kế form xuất hiện:

Tùy từng mục đích, đối tượng làm việc cụ thể mà có các cách làm việc khác nhau. Sau đây là một số ví dụ:

#### Sửa nhãn (Label):

Label là đối tượng tạo ra dòng chữ chú thích trên form. Dòng chữ này (nhãn) phải được nhập trực tiếp từ bàn phím làm sao thật ngắn gọn, xúc tích để người dùng có thể hiểu được bản chất cũng như ý nghĩa của việc bạn đang giải thích.

Giá trị hiển thị trên nhãn chính là giá trị của thuộc tính *Caption*. Sửa nhãn là sửa thuộc tính *Caption* hoặc có thể bấm chuột trực tiếp lên nhãn để sửa giá trị.

### **Thay đổi kích thước đối tượng (Resize):**

Kích thước của đối tượng thường được mô tả ở thuộc tính: *Height* - chiều cao và *With* - chiều rộng. Tuy nhiên bạn hoàn toàn có thể thay đổi kích thước đối tượng một cách trực quan bằng chuột. Cách làm tương tự như việc thay đổi kích thước một đối tượng đồ họa (Graphic) trên Word. Cụ thể như sau:

Khi trỏ chuột (chọn) lên đối tượng cần thay đổi, sẽ xuất hiện 6 điểm trên đối tượng:



Khi đó có thể đặt chuột lên từng điểm (khi nào con trỏ chuột chuyển thành mũi tên 2 đầu) để điều chỉnh độ lớn bằng cách giữ và di chuột.

### **Di chuyển đối tượng:**

Mỗi đối tượng nằm trên form đều được xác định bởi một tọa độ, tọa độ này được thể hiện qua 2 thuộc tính: *Top* - khoảng cách từ tiêu đề form đến đối tượng và *Left* - khoảng cách từ mép form bên trái đến đối tượng. Tuy nhiên bạn nên di chuyển đối tượng bằng cách trực quan sử dụng chuột với thao tác kéo-thả. Chỉ trong những trường hợp đòi hỏi phải di chuyển đối tượng đến các vị trí chính xác mới cần dùng đến 2 thuộc tính *Top* và *Left* để thiết lập.

### **Thay đổi Font chữ:**

Những đối tượng có hiển thị chữ (Text) hoàn toàn có thể thay đổi được phong chữ. Giá trị này được miêu tả ở thuộc tính *Font Name*. Tuy nhiên bạn hoàn toàn sử dụng hộp *Font* trên thanh công cụ *Formatting* để thiết lập nhanh.

### **Thay đổi màu nền:**

Mỗi đối tượng thường có thể thiết lập được màu nền, chúng thể hiện ở thuộc tính *BackColor*. Tuy nhiên hoàn toàn có thể thiết lập màu nền một cách nhanh chóng ở hộp *Fill\Back Color* trên thanh *Formatting*.

### Thay đổi màu chữ:

Mỗi đối tượng thường hiển thị chữ có thể thiết lập được màu chữ, chúng thể hiện ở thuộc tính *ForeColor*. Tuy nhiên hoàn toàn có thể thiết lập màu chữ một cách nhanh chóng ở hộp *Font\Fore Color* trên thanh Formatting

Ngoài ra có thể sử dụng thuộc tính *Format* và *InputMark* để định dạng dữ liệu khi hiển thị cũng như mặt nạ khi nhập dữ liệu. Xin tham khảo cách làm này ở phần các thuộc tính của bảng dữ liệu (Chương 1; mục 2- Xây dựng cấu trúc bảng).

Hãy sử dụng các tính năng định dạng như trên một cách phù hợp, hoàn toàn có thể đưa form ban đầu về dạng dễ nhìn, dễ sử dụng hơn như sau:

The screenshot shows a Microsoft Access form window titled "Form1 : Form". The form is titled "NHẬP HỒ SƠ CÁN BỘ" (Enter Staff Record). It contains the following fields and controls:

- Mã cán bộ: canboID (text box)
- Phòng ban: phongbanID (dropdown menu)
- Họ tên: hoten (text box)
- Chức vụ: chucvuID (dropdown menu)
- Ngày sinh: ngaysinh (text box)
- Hệ số lương: hesoluong (text box)
- Ghi chú: ghichu (text area)
- Three checked checkboxes: "Là nam giới", "Là Đảng viên", and "Vào công đoàn".

### b. Sử dụng Command Button Wizard

Tuy chưa được học một chút gì về lập trình trên Access, nhưng học viên vẫn có thể tạo được một số các nút lệnh đặt trên form điều khiển công việc. Tính năng **Command Button Wizard** giúp làm điều đó.

Nút lệnh (Command Button) thường được dùng để lập trình xử lý các công việc nào đó. Ví dụ: khi nhấn lên nút **In bảng lương** - một bảng lương sẽ được in ra hoặc khi nhấn nút **Đóng Form** - form đang làm việc sẽ được đóng lại. Muốn vậy, người lập trình phải viết các lệnh phía sau nút đó, để làm sao khi nhấn lên nút, các lệnh sẽ

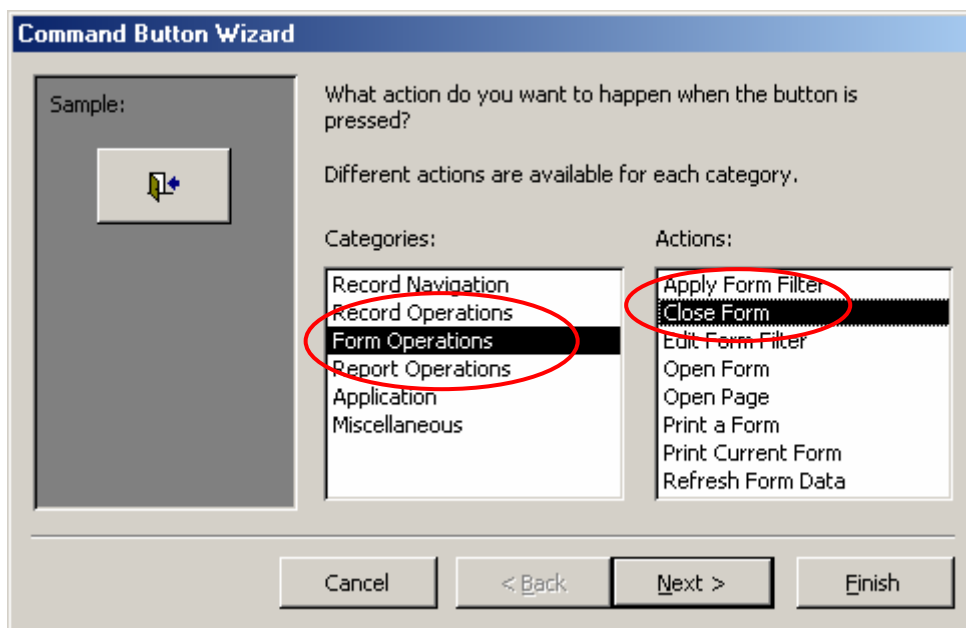
được thi hành để điều khiển công việc như yêu cầu. Control Button Wizard sẽ giúp tạo một số loại nút lệnh mà không cần biết đến lập trình. Dưới đây là các bước sử dụng:

**Ví dụ: Tạo nút** 

Trước khi sử dụng tính năng này, phải đảm bảo nút **Control Wizard** trên thanh công cụ Toolbox đã được nhấn chìm xuống:



**Bước 1:** Dùng chuột đưa đối tượng **Command Button** từ thanh công cụ lên vị trí thích hợp trên Form, hộp thoại sau xuất hiện:



**Bước 2:** Chọn hành động cần làm cho nút lệnh. Quan sát hộp thoại trên có 2 danh sách:

- **Categories:** chứa các nhóm thao tác mà một nút lệnh có thể nhận;
- **Actions:** chứa danh sách các lệnh của mỗi nhóm.

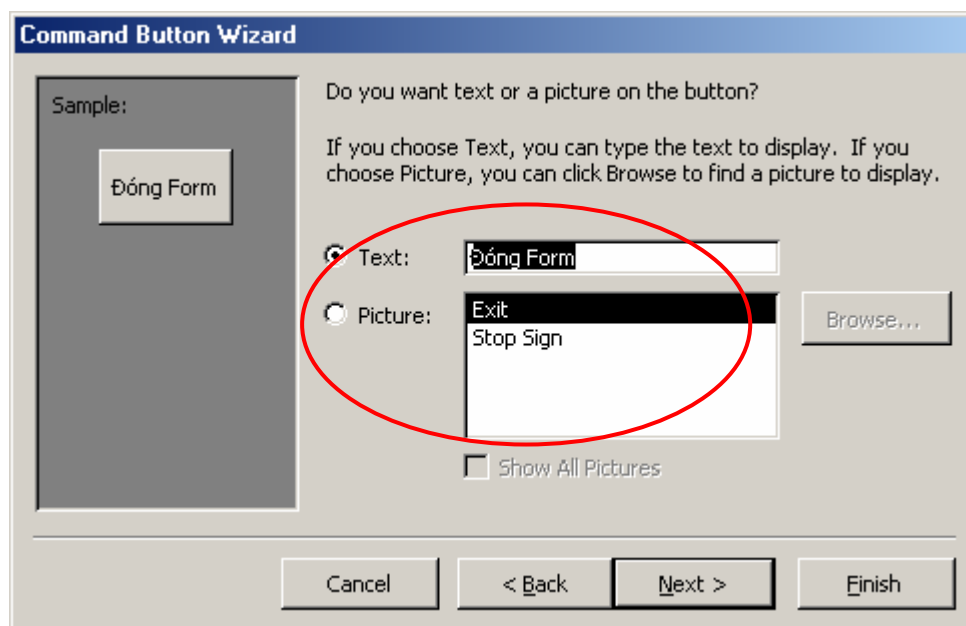
Bảng dưới đây liệt kê danh sách lệnh của từng nhóm:

TT	Nhóm \ Lệnh	Ý nghĩa
1	<b>Record Navigation</b>	<b>Nhóm định hướng bản ghi</b>

1.1	- Goto First Record	- Chuyển về bản ghi đầu
1.2	- Goto Last Record	- Chuyển đến bản ghi cuối cùng
1.3	- Goto Next Record	- Chuyển bản ghi kế sau
1.4	- Goto Previous Record	- Chuyển bản ghi kế trước
2	<b>Record Operations</b>	<b>Các xử lý với bản ghi</b>
2.1	- AddNew Record	- Thêm bản ghi mới
2.2	- Delete Record	- Xoá bản ghi hiện tại
2.3	- Update Record	- Cập nhật bản ghi hiện tại
2.4	- Undo Record	- Phục hồi thay đổi dữ liệu bản ghi
3	<b>Form Operations</b>	<b>Các xử lý với Form</b>
3.1	- Close Form	- Đóng form
3.2	- Open Form	- Mở một form khác
4	<b>Report Operations</b>	<b>Các xử lý với Report</b>
4.1	- Preview report	- Xem trước nội dung (Preview) report
4.2	- Print report	- In report
4.3	- Send report to file	- Xuất report ra một tệp tin ngoài
5	<b>Applications</b>	<b>Xử lý ứng dụng</b>
5.1	- Quit Application	- Thoát khỏi Access
5.2	- Run Application	- Chạy một ứng dụng nào đó (tệp .exe)
6	<b>Miscellaneous</b>	<b>Một số thao tác khác</b>
6.1	- Print table	- In nội dung một bảng
6.2	- Run macro	- Thi hành một Macro
6.3	- Run query	- Thi hành một query

Với yêu cầu đóng form, phải chọn: **Categories** = *Form Actions* và **Actions** = *Close Form*.

Chọn xong nhấn **Next**:

**Bước 3:** Chọn hiển thị cho nút lệnh trên hộp thoại dưới:

- Chọn tùy chọn **Text** nếu muốn nút lệnh hiển thị bằng chữ. Khi đó gõ vào chữ hiển thị trên nút. Hình trên gõ *Đóng Form*;
- Chọn tùy chọn **Picture** nếu muốn thể hiện một hình ảnh lên nút lệnh. Khi đó có thể chọn một hình ảnh trên danh sách.

Bạn có thể quan sát trước được thể hiện của nút lệnh ở hộp Sample bên trái hộp thoại.

- Cuối cùng, nhấn **Finish** để kết thúc việc tạo một nút lệnh.

## 4. Kỹ thuật Sub-form

Kỹ thuật thiết kế form đã trình bày ở các phần trên có thể gọi là Single-form. Sub-form là kỹ thuật thiết kế giao diện rất mạnh, đáp ứng được những yêu cầu xử lý dữ liệu phức tạp mà kỹ thuật Single-form chưa thể đáp ứng. Có thể hiểu Sub-form là việc form này lồng trong form kia (có thể lồng trong nhau nhiều lớp). Form chứa gọi là form mẹ (Main form); form được lồng vào gọi là form con (Sub-form). Việc xử lý dữ liệu trên từng form có thể xử lý độc lập hoặc có quan hệ với nhau tùy theo mục đích công việc.

Sub-form có thể được sử dụng trong các loại form nhập dữ liệu, hoặc sử dụng để hiển thị dữ liệu. Dưới đây minh họa 2 ví dụ tạo form có sử dụng kỹ thuật này.

## Sử dụng Sub-form trong form nhập dữ liệu:

**Bài toán:** Thiết kế form nhập thông tin hoá đơn bán hàng như sau:

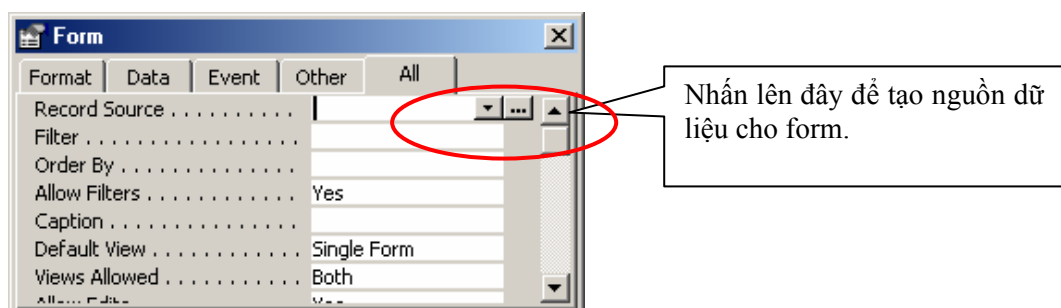
hangID	donvi	dongia	soluong	Thanhtien
Floppy disk	chiếc	8	3	24
Hard disk	ổ	67	9	603
CPU P4 2.4	con	133		266

- Với yêu cầu này, phải tạo một form con cho phép hiển thị và nhập các thông tin về hàng bán (1);
- Phải tạo một form mẹ để nhập thông tin về hoá đơn bán hàng (2);
- Phải kết nối form mẹ với form con thông qua trường hoadonID.

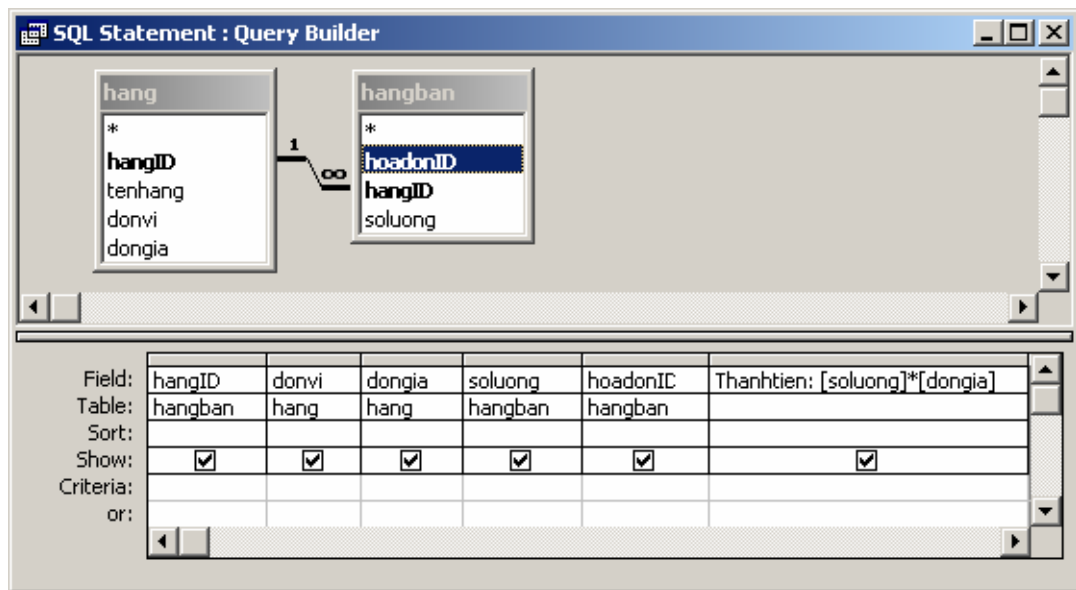
Cách làm như sau (yêu cầu thực hiện theo đúng thứ tự công việc các bước):

### BƯỚC 1: Tạo form con

- Tạo mới form ở chế độ **Design View**;
- Vì form con sẽ hiển thị dữ liệu dạng bảng nên phải thiết lập thuộc tính **Default View** cho form con là *Datasheet*;
- Thiết lập nguồn dữ liệu làm việc cho form con ở thuộc tính **Record Source** bằng cách nhấn nút ... (hình dưới)



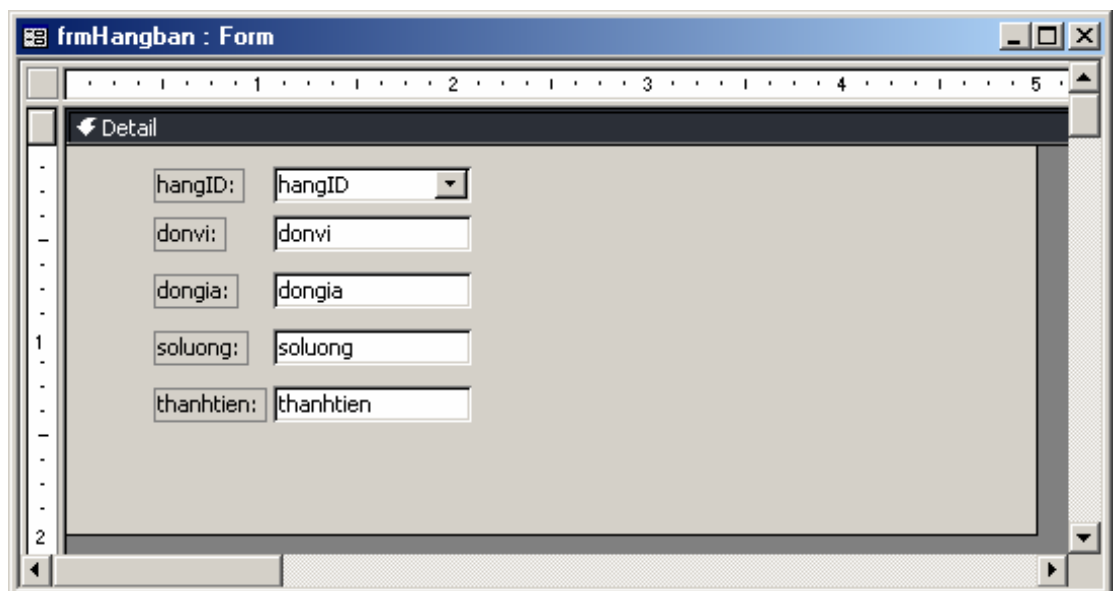
Một màn hình thiết kế query xuất hiện. Hãy thiết lập query này như sau:



(Chú ý phải chọn đúng tên trường của các bảng đã chỉ dẫn)

Tiếp theo đóng màn hình thiết kế query

- Sử dụng cửa sổ **Field List** để đưa các trường cần thiết lên vùng Detail của form bao gồm: *hangID*, *donvi*, *dongia*, *soluong*, *thanhTien*;
- Ghi lại form với một tên gọi (ví dụ: *frmHangban*) và đóng lại, chuẩn bị tạo form mẹ.

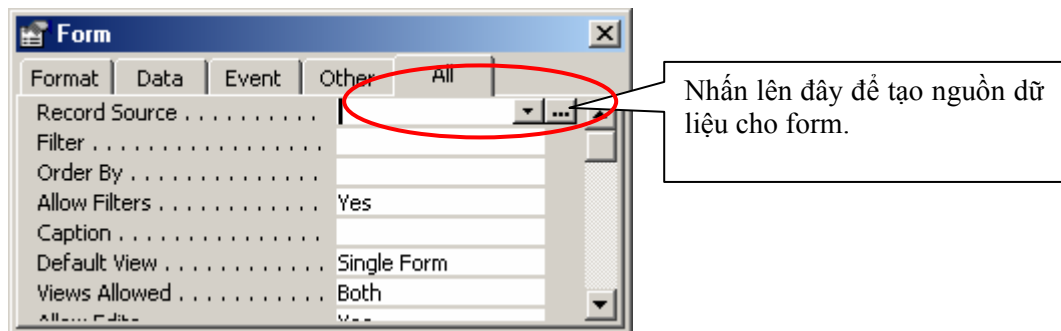


## BƯỚC 2: Tạo form mẹ

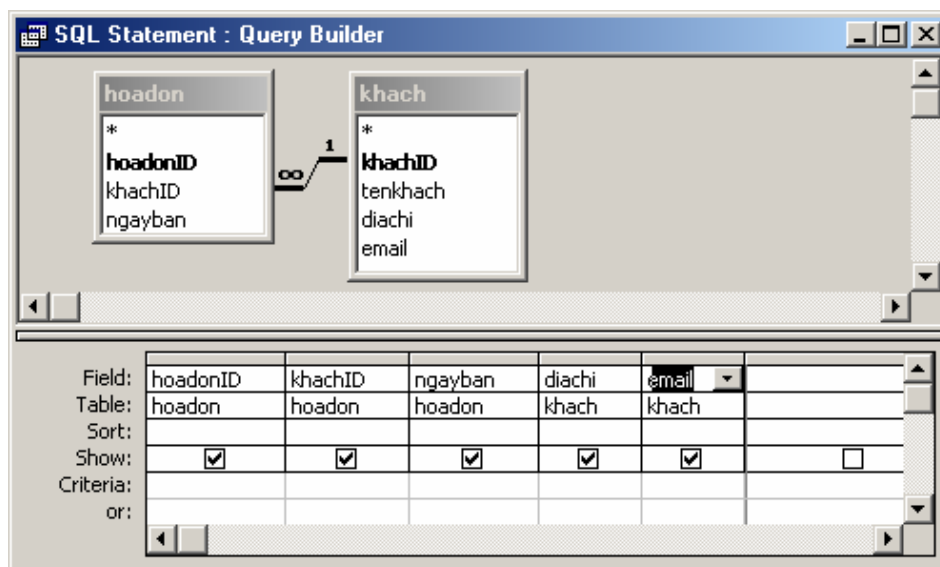
- Tạo mới form ở chế độ **Design View**;



- Thiết lập nguồn dữ liệu làm việc cho form mẹ ở thuộc tính **Record Source** bằng cách nhấn nút ... (hình dưới)



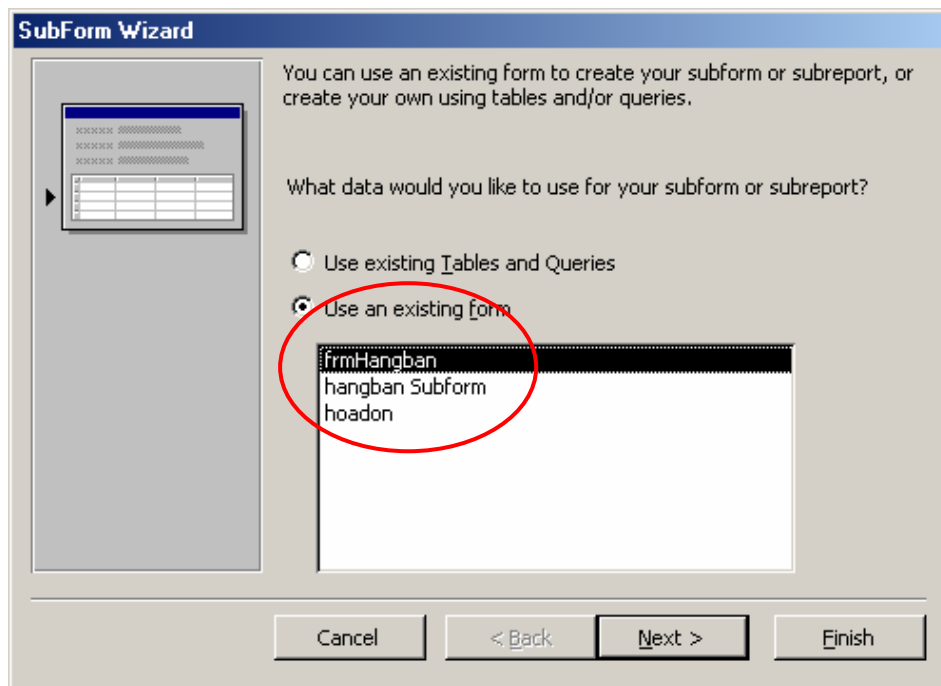
Một màn hình thiết kế query xuất hiện. Hãy thiết lập query này như sau:



*(Chú ý phải chọn đúng tên trường của các bảng đã chỉ dẫn)*

Tiếp theo đóng màn hình thiết kế query

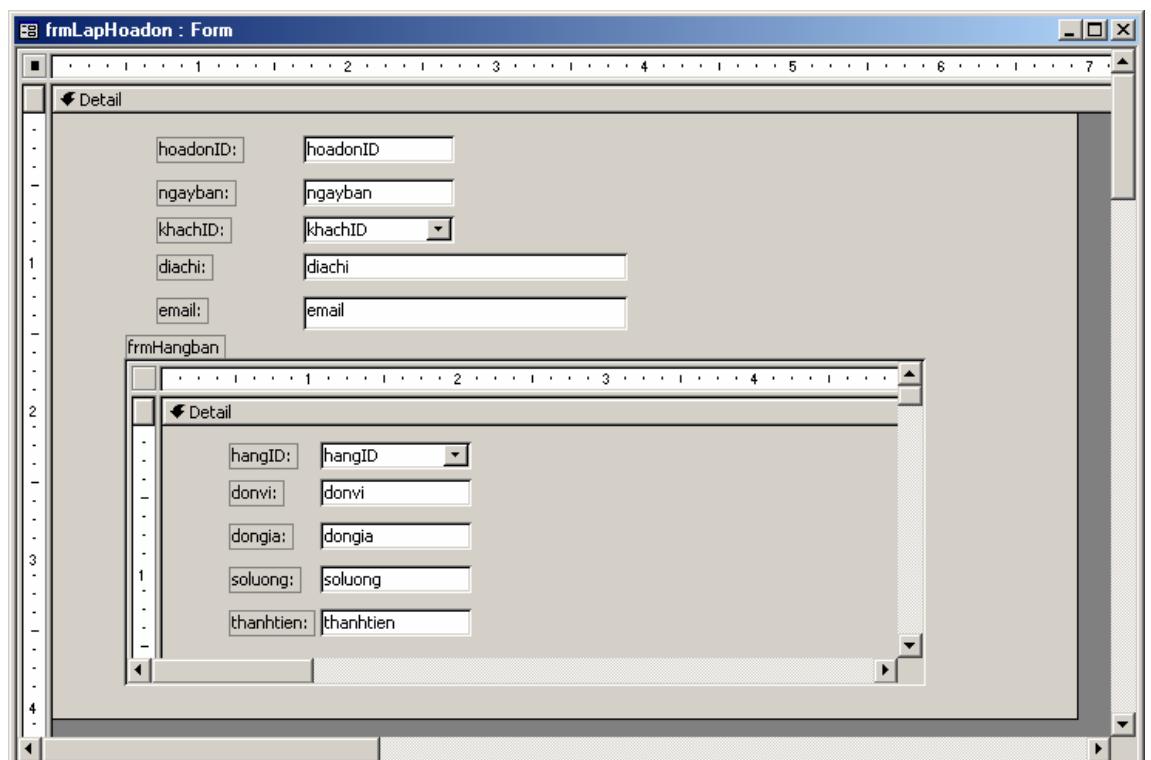
- Sử dụng cửa sổ **Field List** để đưa các trường cần thiết lên vùng Detail của form bao gồm: *hoadonID, khachID, ngayban, diachi, email*;
- Sử dụng đối tượng **Subform\Subreport** trên thanh công cụ Toolbox để đưa form con lên form mẹ. Sau khi đưa một đối tượng Subform\Subreport lên form, một hộp thoại xuất hiện;



Hãy chọn tên form con cần đưa lên ở danh sách **Use an existing form**.  
 Chọn xong nhấn **Finish**.

*(Có thể đưa form con lên form mẹ bằng cách dùng chuột gấp tên form con  
 thả lên form mẹ)*

- Ghi lại form với một tên gọi (ví dụ: *frmLapHoadon*) và đóng lại, kết thúc yêu cầu cơ bản bài toán, thu được kết quả như sau:



Khi sử dụng Form wizard, bạn cũng rất dễ dàng khi tạo form theo yêu cầu trên. Cách làm ở đây là kết hợp giữa form wizard với form Design view. Chúng tôi trình bày thêm cách này để các bạn tham khảo.

**Bước 1:** Tạo form wizard lấy những dữ liệu sau:

Bảng *HOADON* lấy các trường: *hoadonID*, *khachID*, *ngayban*;

Bảng *KHACH* lấy các trường: *diachi*, *email*;

Bảng *HANGBAN* lấy các trường: *hangID*, *soluong*;

Bảng *HANG* lấy các trường: *donvi*, *dongia*.

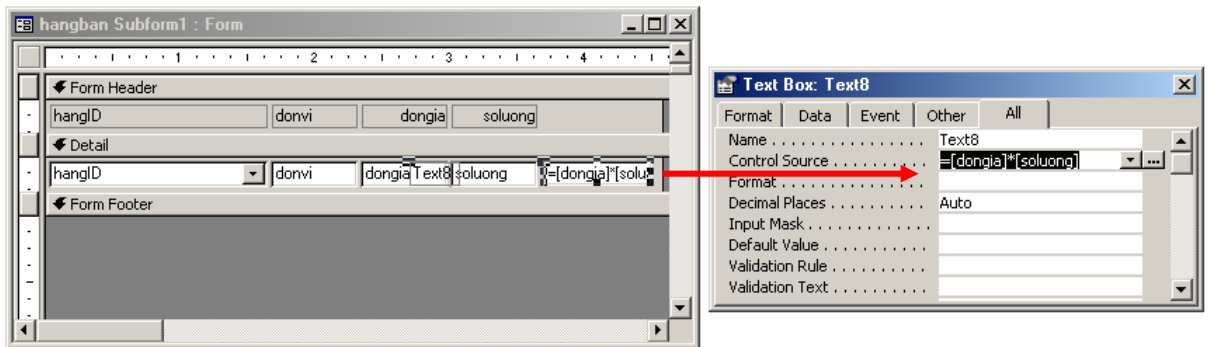
Sau bước này Access sẽ tạo được form theo yêu cầu có dạng:

hangID	donvi	dongia	soluong
Floppy disk	chiếc	8	3
Hard disk	ổ	67	9
CPU P4 2.4	con	133	2

Form này gần đáp ứng yêu cầu đầu bài (chỉ thiếu cột *Thanhtien* trong form con).

**Bước 2:** Sửa form con để đáp ứng bài toán:

- Mở form con ở chế độ Design View (thông thường tên form con này Access sẽ đặt *hangban subform*);
- Thêm một **Textbox** lên vùng Detail của form con và thiết lập thuộc tính **Control Source** cho text box này như sau (= **[dongia]\*[soluong]**):



- Đóng và ghi lại tất cả. Bạn đã tạo được form như yêu cầu.

### Sử dụng Sub-form trong form lọc dữ liệu

**Bài toán:** Thiết kế form lọc ra danh sách cán bộ một phòng ban nào đó như sau:

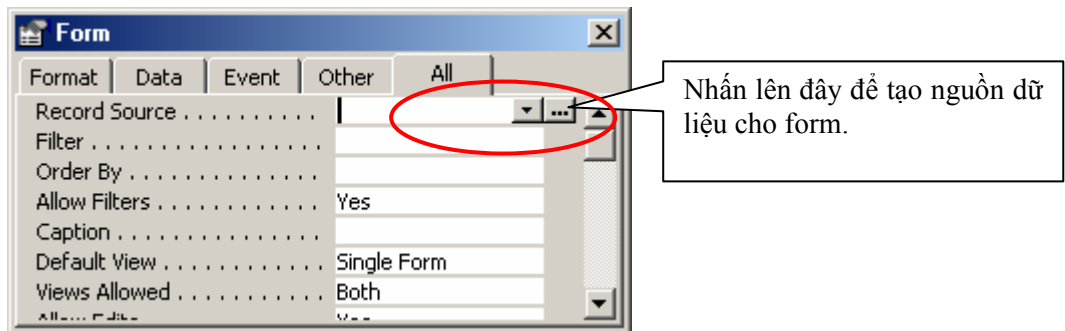
	Họ tên	Ngày sinh	Giới tính	Chức vụ
▶	Nguyễn Đức Ngô	4/12/1976	Nam	Giám đốc
	Phan Quang Sáng	8/8/1976	Nữ	Nhân viên
	Nguyễn Văn Hoà		Nữ	Nhân viên
	Phan Hoài Linh		Nam	Nhân viên
	Phan Anh Quân	7/23/1978	Nữ	Trưởng phòng
*				

- Với yêu cầu này, phải tạo một form con cho phép hiển thị các thông tin về danh sách cán bộ cần in ra (1);
- Phải tạo một form mẹ cho phép nhập các tham số cần lọc (2);
- Phải thiết lập tham số cho form con sao cho mỗi khi chọn một phòng ban trên form mẹ, danh sách cán bộ ở phòng ban đó sẽ tự động hiển thị ra form con.

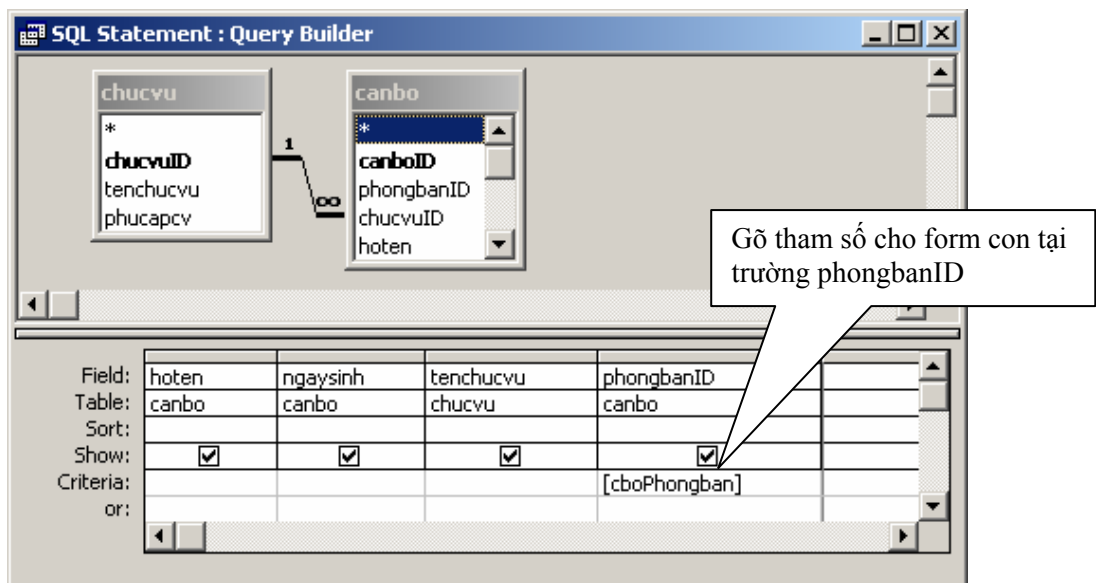
Cách làm như sau (yêu cầu thực hiện theo đúng thứ tự công việc các bước):

**Bước 1:** Tạo form con

- Tạo mới form ở chế độ Design View;
- Vì form con sẽ hiển thị dữ liệu dạng bảng nên phải thiết lập thuộc tính **Default View** cho form con là *Datasheet*;
- Thiết lập nguồn dữ liệu làm việc cho form con ở thuộc tính **Record Source** bằng cách nhấn nút ... (hình dưới)



Một màn hình thiết kế query xuất hiện. Hãy thiết lập query này như sau:



(Chú ý phải chọn đúng tên trường của các bảng đã chỉ dẫn)

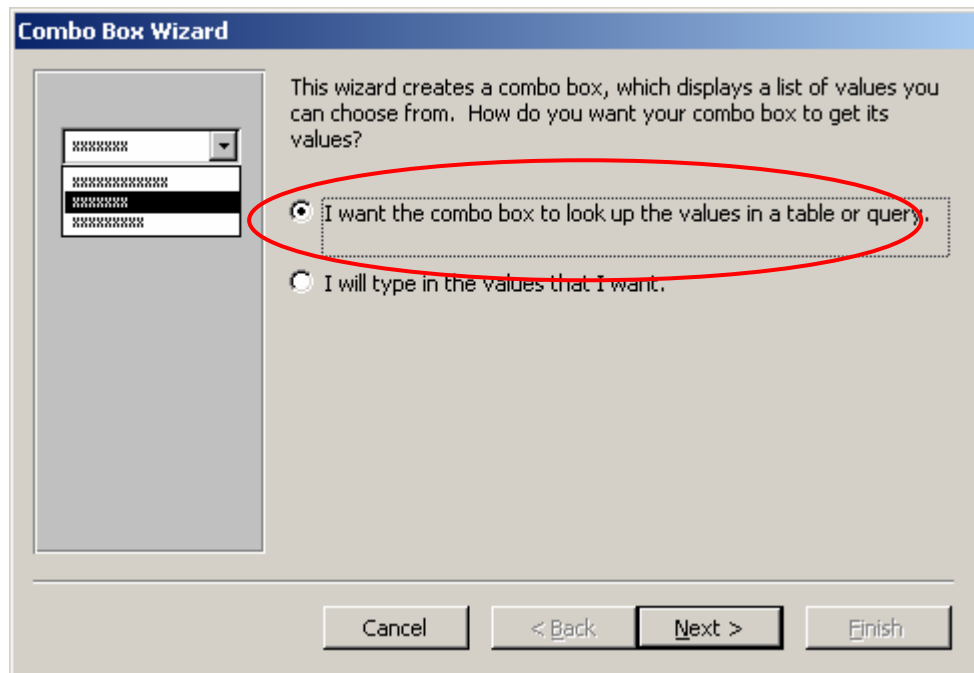
Tiếp theo đóng màn hình thiết kế query

- Sử dụng cửa sổ **Field List** để đưa các trường cần thiết lên vùng Detail của form bao gồm: *hoten*, *gioitinh*, *ngaysinh*, *tenchucvu*;
- Ghi lại form với một tên gọi (ví dụ: *frmDscb\_subform*) và đóng lại, chuẩn bị tạo form mẹ.

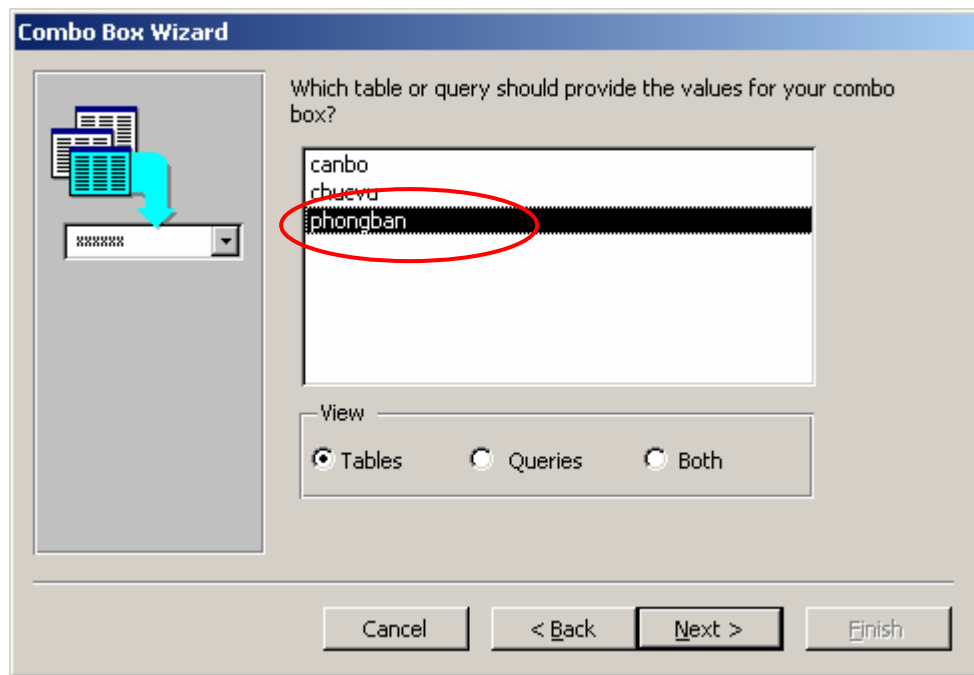
**Bước 2:** Tạo form mẹ

- Tạo mới form ở chế độ **Design View**;
- Tạo hộp **Combo box** chứa danh sách các phòng ban có thể chọn trên form mẹ bằng cách:

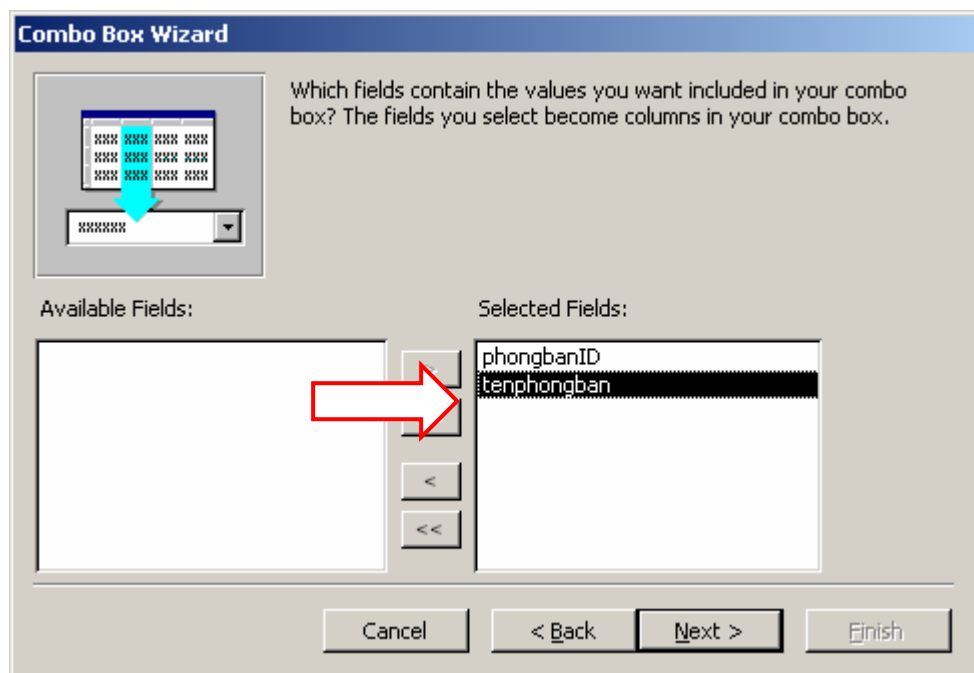
Dùng chuột kéo đối tượng **Combobox** từ thanh công cụ **ToolBox** thả lên form mẹ, hộp thoại sau xuất hiện:



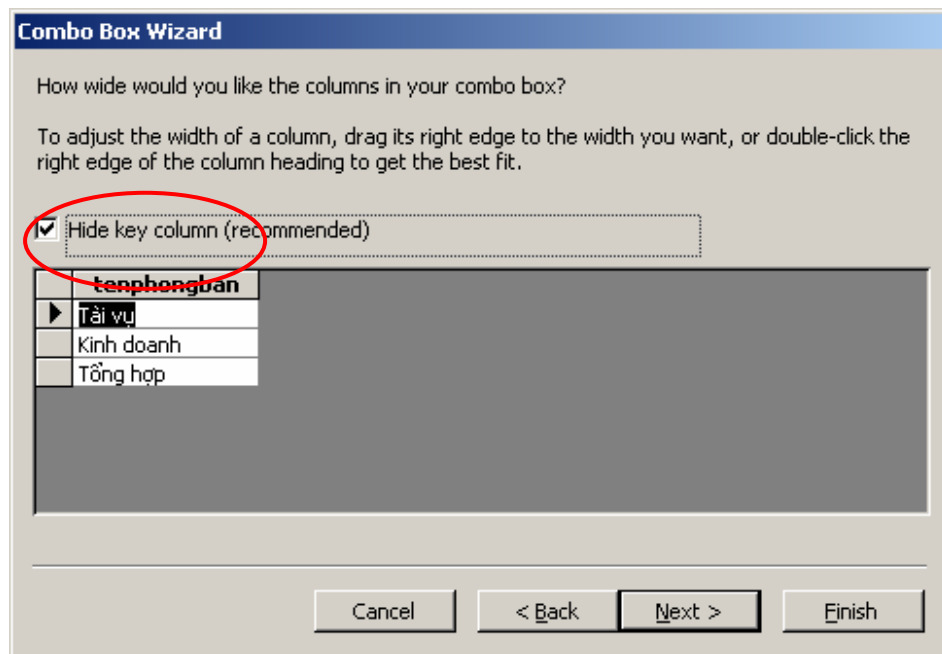
Hãy chọn mục **I want the combo box look up the value in a table or query**. Nhấn **Next** để tiếp tục, hộp thoại sau xuất hiện:



Chọn bảng từ danh sách **Table\Queries** – nơi có chứa danh sách phòng ban cần đưa vào hộp chọn Combobox. Chọn PHONGBAN, nhấn **Next**, hộp thoại tiếp theo xuất hiện:



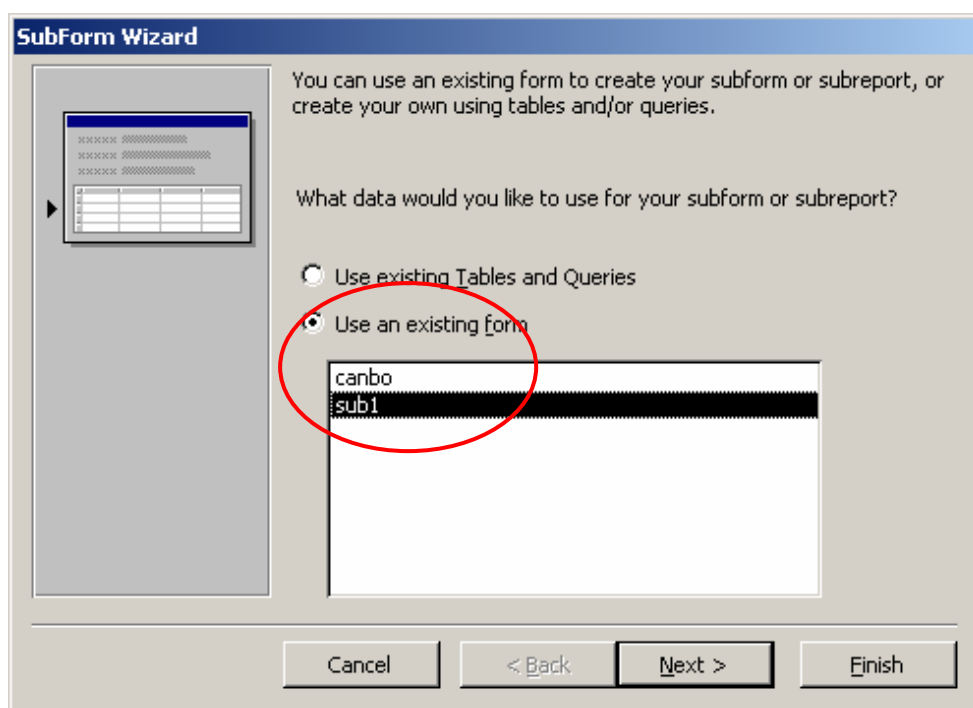
Chọn các trường sẽ hiển thị giá trị lên Combo box từ danh sách **Available Fields:** → **Selected Fields:** (chọn 2 trường *phongbanID* và *Tenphan*). Nhấn **Next** để tiếp tục:



Nên chọn mục **Hide key column (recommended)** để ẩn đi khoá làm cho danh sách thả xuống hộp thả gọn gàng hơn. Cuối cùng nhấn **Finish** để kết thúc việc tạo Combo box.

Thiết lập thuộc tính **Name** cho **Combo box** trùng tên với tham số đã thiết lập ở form con là *cboPhongban*.

- Sử dụng đối tượng **Subform\Subreport** trên thanh công cụ **ToolBox** để đưa form con lên form mẹ. Sau khi đưa một đối tượng **Subform\Subreport** lên form, một hộp thoại xuất hiện;





Hãy chọn tên form con cần đưa lên ở danh sách **Use an existing form**.  
Chọn xong nhấn **Finish**.

- Ghi lại form với một tên gọi (ví dụ: *frmTracuuDanhSachCanBo*) và đóng lại, kết thúc yêu cầu cơ bản bài toán.



#### **Chú ý**

1. Giá trị thuộc tính Name của hộp Combobox trên form mẹ phải trùng với tham số đã đặt ở form con. Nếu chưa trùng nhau phải sửa lại cho khớp: hoặc sửa Name của Combo cho khớp với tham số form con hoặc thao tác ngược lại mở form con sửa tham số cho khớp Name của ô Combobox.

## Bài tập

### Cơ sở Quản lý lương cán bộ

**Bài số 1:** Tạo form cho phép xem danh sách cán bộ từng phòng ban như sau:

Chọn phòng ban: kinh doanh

**Danh sách cán bộ:**

	Họ tên	Ngày sinh	Giới tính	Chức vụ
▶	Nguyễn Đức Ngô	12/04/1976	Nam	Giám đốc
	Phan Quang Sáng	03/09/1976	Nữ	Nhân viên
	Nguyễn Văn Hoà		Nữ	Nhân viên
	Phan Hoài Linh		Nam	Nhân viên
	Phan Anh Quân	23/07/1978	Nữ	Trưởng phòng
*				

Record: 1 of 5

Mỗi khi chọn một phòng ban từ hộp thả

Chọn phòng ban: kinh doanh

danh sách cán bộ phòng ban đó được hiển thị lên **Subform**.

**Bài số 2:** Tạo form cho phép tìm kiếm cán bộ theo tên như sau:

Form: **Tìm cán bộ theo Tên**

Nhập tên cán bộ muốn tìm:

**Danh sách cán bộ tìm thấy:**

	hoten	ngaysinh	tenchucvu	tenphongban
▶	Nguyễn Đức Ngô	12/04/1976	Giám đốc	Kinh doanh
	Trần Văn Ngô		Nhân viên	Tổng hợp
*				

Record:  of 2

Sau khi gõ một tên (có thể là đệm + Tên hoặc đầy đủ họ và tên), danh sách các kết quả tìm thấy sẽ được liệt kê lên Subform.

### Trên CSDL Quản lý bán hàng hãy:

**Bài số 3:** Thiết kế form lập hoá đơn bán hàng theo mẫu:

Form: **Lập hoá đơn bán hàng**

hoadonID:  Ngày bán:

Tên khách:

Địa chỉ:

E-mail:

**Chi tiết hàng bán:**

	hangID	donvi	dongia	soluong	thanhtien:
	Floppy disk	chiếc	8	3	24
	Hard disk	ổ	67	9	603
▶	CPU P4 2.4	con	133	2	266
*					

Record:  of 3

Tổng tiền:

Buttons:

Yêu cầu chi tiết:

- Các nút Tiến, Lùi để định vị hoá đơn cần làm việc;

- Nút Thêm mới để bắt đầu tạo mới một hoá đơn;
- Nút Xoá HĐ để xoá hoá đơn hiện tại. Yêu cầu phải có xác nhận trước khi xoá;
- Nút In HĐ để in chi tiết hoá đơn bán hàng ra report (chức năng này sẽ được hoàn thiện sau khi học xong Chương Report)

**Bài số 4:** Thiết kế form tổng hợp thông tin bán hàng như sau:

Tên hàng:	Đơn vị:	Tổng số bán:
CPU P4 2.4	con	15
Floppy disk	chiếc	11
Hard disk	ổ	9
RAM 256	thanh	15

Sau khi nhập xong giá trị từ ngày và đến ngày, thông tin chi tiết về các mặt hàng bán ra trong phạm vi ngày đó được tổng hợp theo như giao diện trên.

**Bài số 5:** Thiết kế form tổng hợp thông tin bán hàng cho một khách hàng nào đó trong một khoảng ngày nào đó như sau:

**TỔNG HỢP THÔNG TIN BÁN HÀNG**

Chọn khách hàng: Nguyễn Đức Ngô

Từ ngày: 1/1/2004 Đến ngày: 12/1/2004

**Chi tiết hàng đã bán:**

Tên hàng:	Đơn vị:	Tổng số bán:
CPU P4 2.4	con	5
Floppy disk	chiếc	3
Hard disk	ổ	9
RAM 256	thanh	15

Record: 1 of 4

Yêu cầu: Sau khi chọn một khách hàng, nhập vào khoảng ngày cần tổng hợp. Thông tin tổng hợp về từng mặt hàng của khách đó mua sẽ được liệt kê.

**Bài số 6:** Tạo form tổng hợp thông tin hàng đã bán chi tiết theo từng ngày như sau:

**THEO DÕI CHI TIẾT HÀNG BÁN**

Từ ngày: 1/10/2004 Đến ngày: 30/10/2004

**Chi tiết hàng bán:**

tenhang:	10/10/2004:	12/10/2004:	24/10/2004:
CPU P4 2.4	3	12	
Floppy disk		11	
Hard disk		9	
RAM 256	12		3

Record: 1 of 4

**Bài số 7:** Tạo form theo dõi thông tin hoá đơn bán hàng của một khách hàng nào đó:

Chọn khách: Nguyễn Đức Ngô

**Danh sách hoá đơn đã mua:**

	hoadonID:	ngayban:	Tongtien:
▶	1	12/10/2004	893
	2	24/10/2004	114
	3	10/10/2004	855
	4	12/10/2004	1394

Record: 1 of 4

Sau khi chọn tên một khách, thông tin về các hoá đơn mua hàng được hiển thị.

# CHƯƠNG 4

## THIẾT KẾ BÁO CÁO

Report là công cụ dùng thiết kế in ấn dữ liệu hữu hiệu trong Access. Hầu như tất cả những gì bạn muốn in và in theo bố cục như thế nào? Report đều có thể đáp ứng!

Chương này sẽ trình bày từ những khái niệm căn bản về thiết kế in ấn, về Report của Access đến tiếp cận những kỹ thuật in ấn phức tạp như: report có tham số, sub-report, ..

Nội dung cụ thể bao gồm:

- Các khái niệm về Report;
- Sử dụng report wizard;
- Sử dụng report design view;
- Kỹ thuật sub-report;
- Tham số cho report.

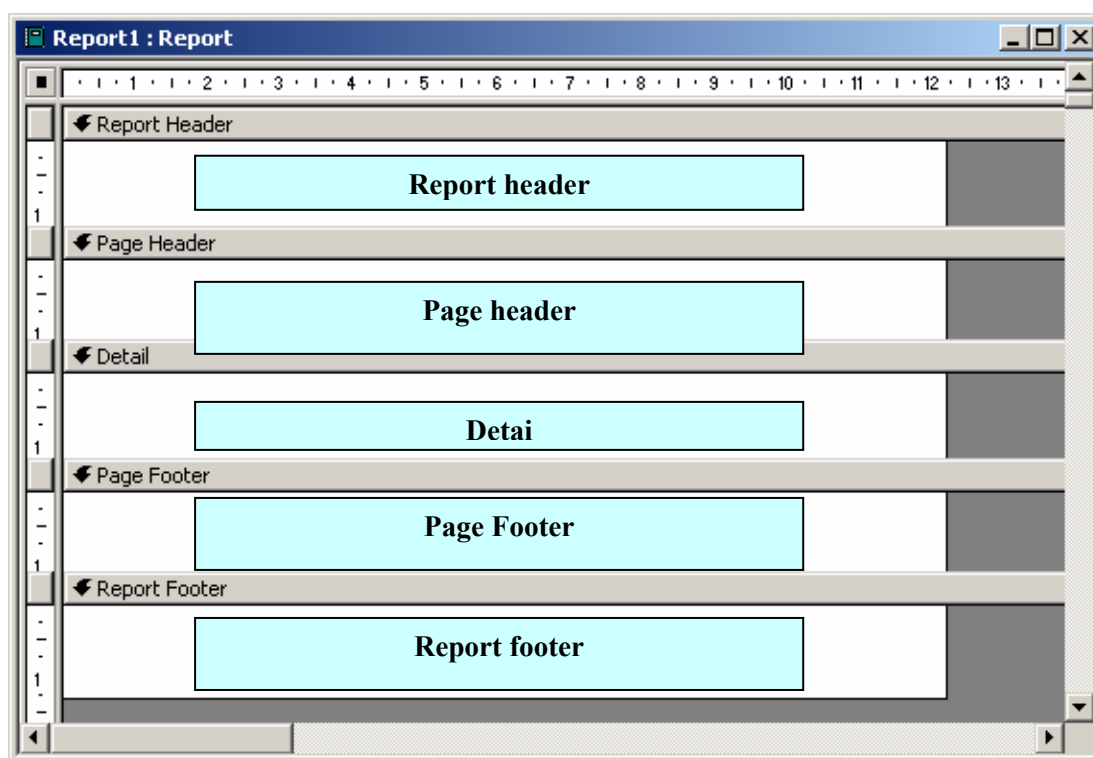
Kết thúc chương, học viên hoàn toàn có thể thiết kế được những mẫu biểu in ấn đơn giản đến những biểu phức tạp phục vụ nhu cầu in ấn đa dạng trong các bài toán thực tế từ CSDL.

## 1. Các khái niệm về Report

Muốn in các báo cáo trong Access bạn có thể sử dụng Report- một công cụ in ấn rất mạnh. Ứng với mỗi mẫu báo cáo có thể thiết kế các thông tin lên một report. Mỗi khi report hiển thị kết quả (preview) là lúc có thể in được nội dung báo cáo ra giấy.

### 1.1 Cấu trúc Report

Cấu trúc một report thông thường gồm 5 phần:



- **Page Header**

Là phần đầu tiên của một trang báo cáo. Giống như khái niệm Page header trong Word và Excel. Phần này có thể có hoặc không có thông tin tùy vào người thiết kế.

- **Page Footer**



Là phần cuối cùng của mỗi trang báo cáo. Giống như khái niệm Page footer trong Word và Excel. Phần này có thể có hoặc không có thông tin tùy vào người thiết kế.

- **Detail**

Là phần thân của report – nơi hiển thị giá trị các bản ghi sẽ in ra. Phần này có thể bị thay đổi, phụ thuộc vào nguồn dữ liệu tại thời điểm sẽ in ra của report. Ví dụ: Danh sách học sinh lớp 10A1 là 45 khi đó Detail report sẽ in ra 45 dòng; nếu chuyển sang in danh sách học sinh lớp 10A2 có 48 học sinh, lúc này Detail report sẽ in ra 48 dòng (tương ứng với số bản ghi của nguồn dữ liệu sẽ in ra).

- **Report Header**

Là phần tiêu đề của trang đầu tiên Report, nằm tiếp theo phần Page header và nằm trên phần Detail. Mỗi Report sẽ chỉ có nhiều nhất 1 Report header. Phần này có thể có hoặc không có thông tin tùy vào người thiết kế.

- **Report Footer**

Là phần tiêu đề cuối của trang cuối cùng Report, nằm tiếp theo phần Detail và phía trước phần Page Footer. Mỗi Report sẽ chỉ có nhiều nhất 1 Report footer. Phần này có thể có hoặc không có thông tin tùy vào người thiết kế.

## **1.2 Môi trường làm việc**

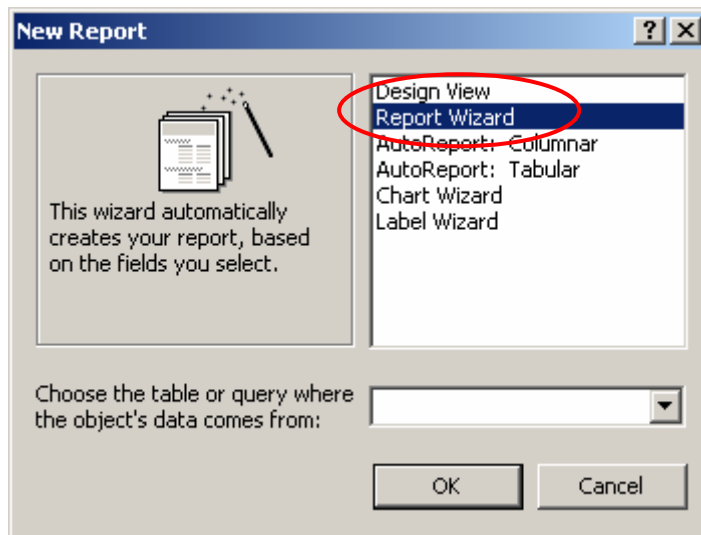
Làm việc với **Report** gần giống làm việc với **Form**. Sự khác nhau cơ bản của Report và Form là: Form có thể hiển thị, tra cứu, cập nhật và điều khiển dữ liệu từ CSDL; còn Report chỉ có thể lập báo cáo và in ra, đặc biệt Report không làm thay đổi giá trị dữ liệu trong CSDL.


Thông thường mỗi report sẽ in dữ liệu của một Table hoặc một Query nào đó. Tức là phải có một nguồn dữ liệu cần in cụ thể (trường hợp đặc biệt Report không có nguồn dữ liệu sẽ nói đến phần cuối)

## 2. Sử dụng Report wizard

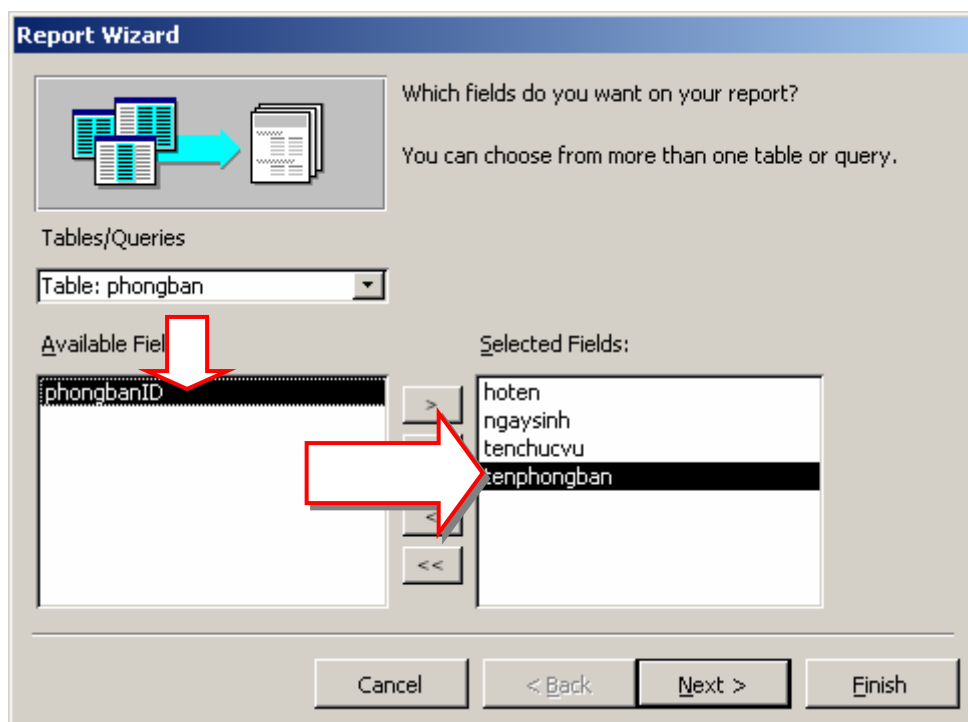
Giống như Form wizard, Report wizard là một công cụ rất đơn giản, dễ dùng để tạo nhanh một Report. Dưới đây hướng dẫn từng bước dùng Report wizard để tạo một report in ra danh sách cán bộ từ CSDL Quản lý lương bao gồm các thông tin: *Hoten, ngaysinh, tenchucvu, Tenphongban*.

**Bước 1:** Ở thẻ Reports, nhấn **New**, chọn *Report wizard*, nhấn **OK**:



Hoặc nhấn đúp chuột lên biểu tượng  **Create report by using wizard**

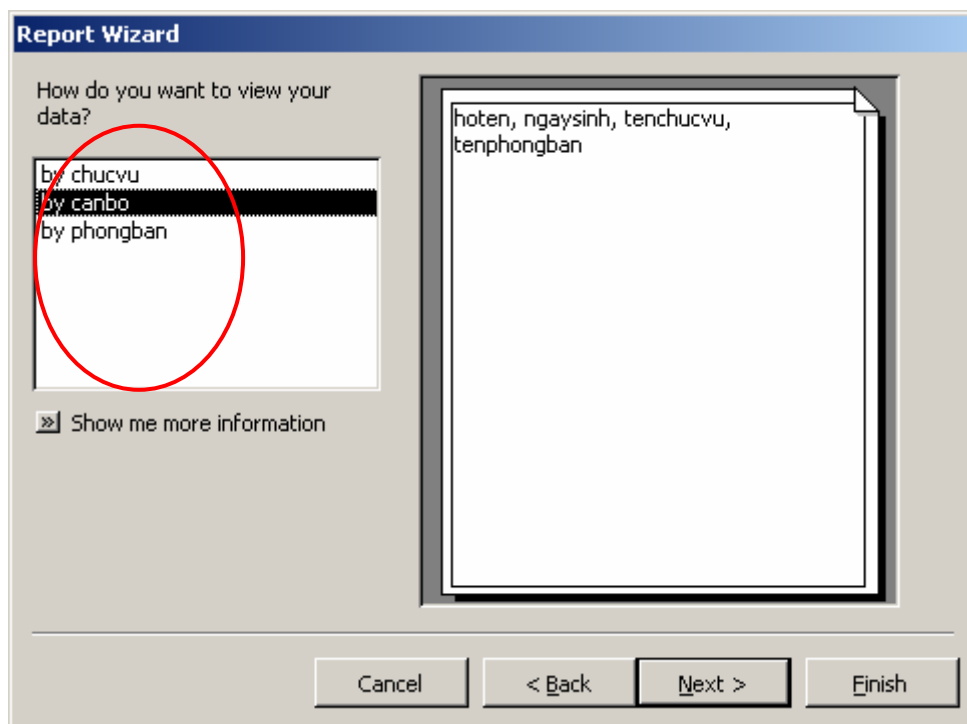
**Bước 2:** Chọn dữ liệu cần in trên hộp thoại sau:



Để chọn dữ liệu cần in lên report, có 2 thao tác phải làm trên hộp thoại này:

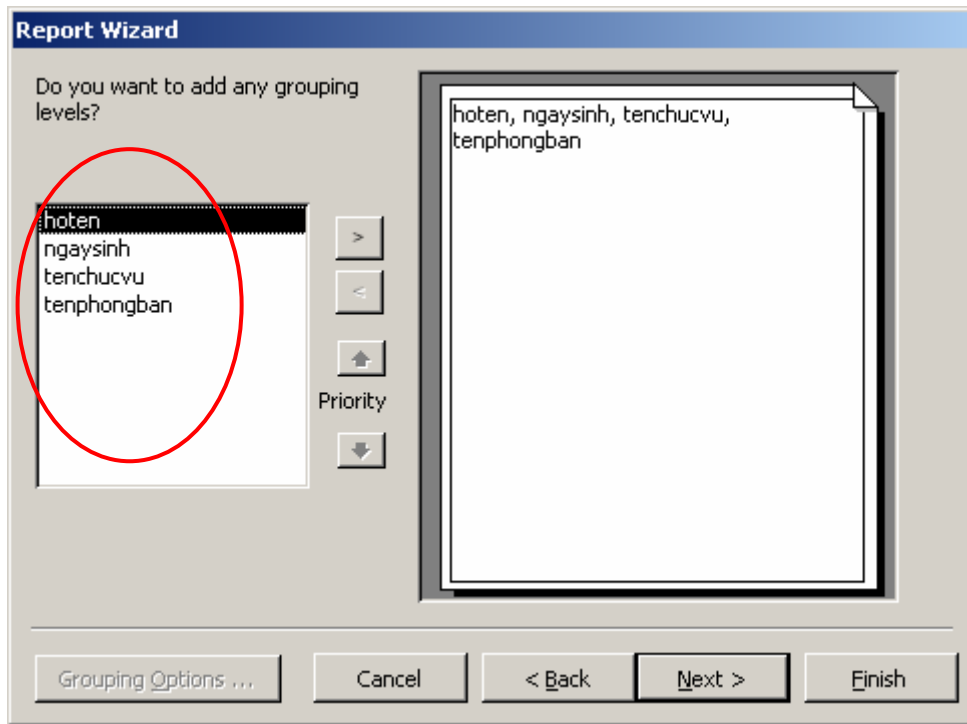
- Chọn bảng hoặc query nơi có chứa trường dữ liệu cần in ra ở hộp **Table/Queries**;
- Sử dụng các nút lệnh >, >>, <, << để đưa các trường cần in từ danh sách **Available Fields**: (danh sách các trường có thể in) sang danh sách **Selected Fields**: (danh sách các trường sẽ được in ra report). Hãy lần lượt thực hiện chọn các trường *hoten*, *ngaysinh* (từ bảng *CANBO*), trường *tenchucvu* (bảng *CHUCVU*) và trường *tenphongban* (bảng *PHONGBAN*). Chọn xong nhấn **Next**:

**Bước 3:** Chọn kiểu cách hiển thị dữ liệu trên report:



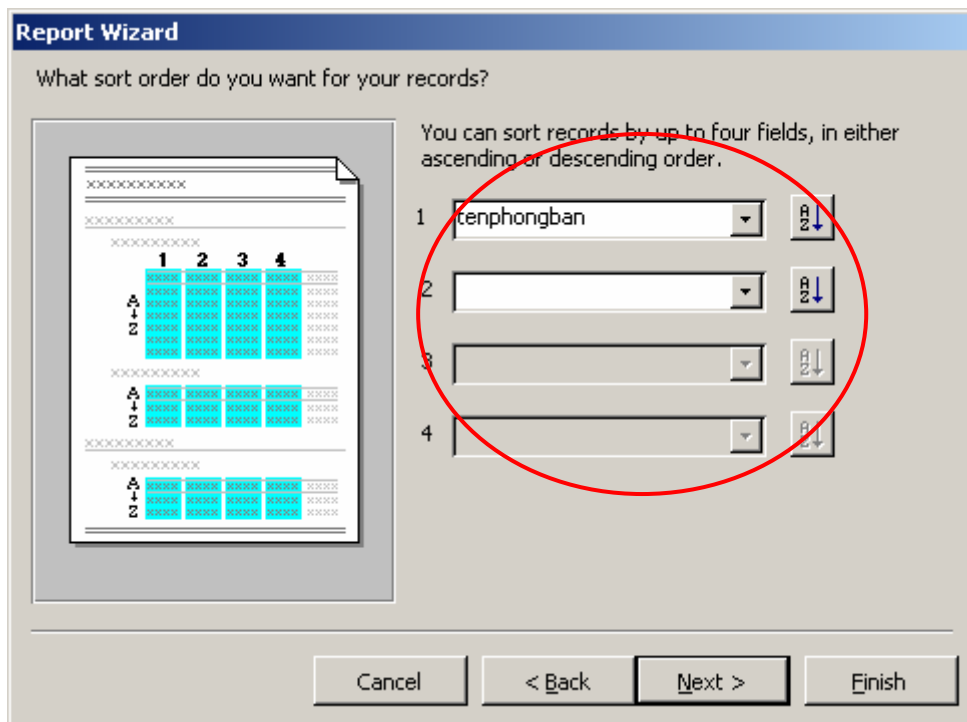
Bạn muốn in dữ liệu theo kiểu nào: Hãy chọn kiểu cần in từ danh sách bên trái hộp thoại. Trong yêu cầu này có thể có 3 kiểu hiển thị dữ liệu trên report (vì dữ liệu được chọn ra từ 3 bảng khác nhau). Vì muốn in một danh sách cán bộ nên chọn kiểu *by CANBO* (có thể xem kiểu hiển thị bên phải hộp thoại). Nhấn **Next** để tiếp tục:

**Bước 4:** Thiết lập các nhóm dữ liệu cần thiết:



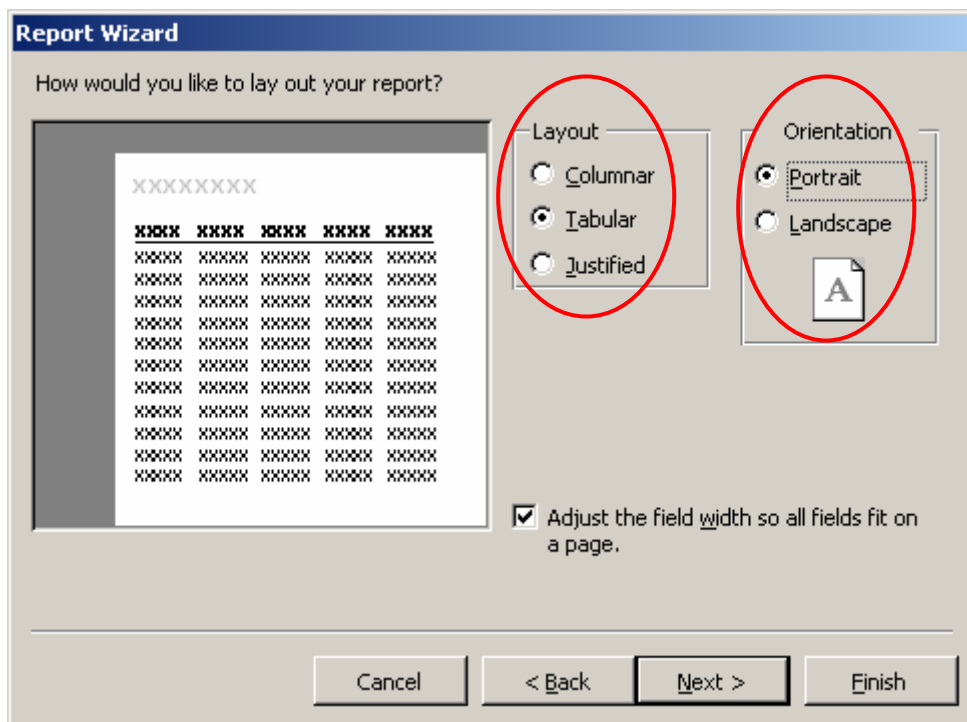
Muốn hiển thị dữ liệu theo nhóm giá trị của trường nào, hãy Add trường đó từ danh sách bên trái hộp thoại sang hộp preview bên phải hộp thoại. Trong bài này chỉ cần hiển thị một danh sách chung nên không cần thiết lập nhóm ở bước này. Nhấn **Next** để tiếp tục:

#### **Bước 5:** Thiết lập sắp xếp dữ liệu trên báo cáo:



Report wizard cho phép tối đa 4 mức ưu tiên sắp xếp dữ liệu được đánh số từ 1 đến 4 (hộp thoại trên). Trường nào thiết lập trước, sẽ được ưu tiên sắp xếp trước. Trong trường hợp giá trị trường đó trùng nhau, Access sẽ chuyển đến các mức tiếp theo để sắp xếp. Kiểu sắp xếp (theo chiều tăng hoặc giảm) có thể thiết lập khi nhấn nút ... bên cạnh. Thiết lập xong nhấn **Next** để tiếp tục:

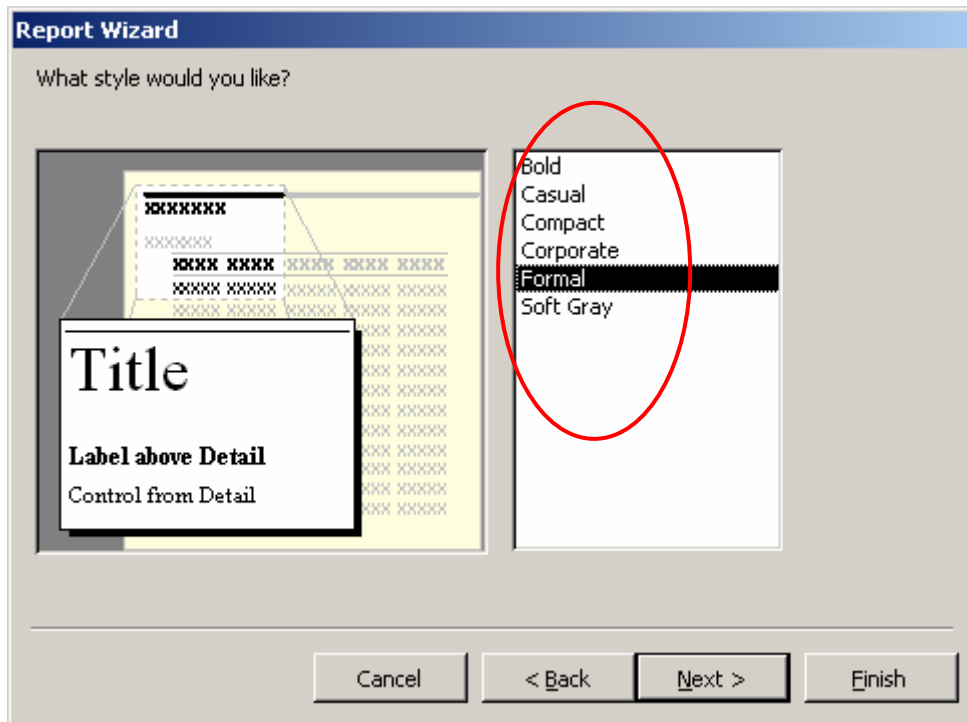
**Bước 6:** Chọn bố cục (Layout) cho Report:



Có 2 thiết lập trong bước này:

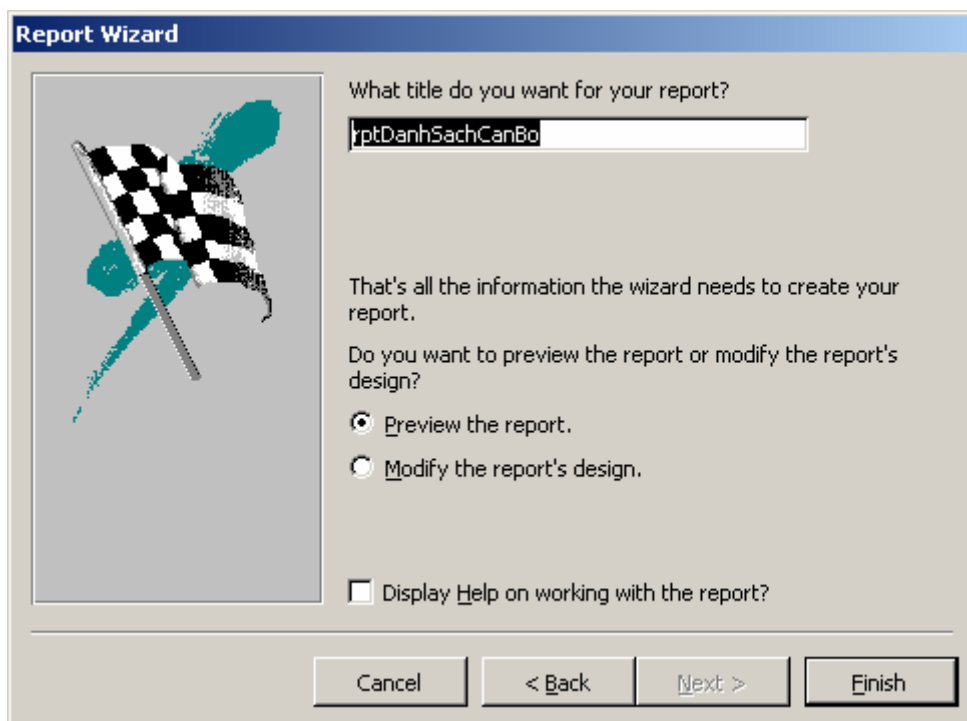
- Chọn bố cục cho report ở hộp **Layout**. Có 3 kiểu Layout: *Columnar*, ***Tabular*** và *Justified*- hãy chọn một kiểu phù hợp (xem hộp preview bên trái để biết trước kết quả);
- Chọn hướng giấy in ở hộp **Orientation**. Có 2 kiểu hướng in là: ***Portrait*** – in theo khổ giấy dọc và *Landscape*- in theo khổ giấy ngang;
- Chọn xong nhấn **Next** để tiếp tục:

**Bước 7:** Chọn mẫu định dạng (Style) cho report:



Hãy chọn một mẫu định dạng từ danh sách bên trái hộp thoại. Chọn xong nhấn **Next** để tiếp tục:

**Bước 8:** Hoàn thiện các thông tin cuối cùng:

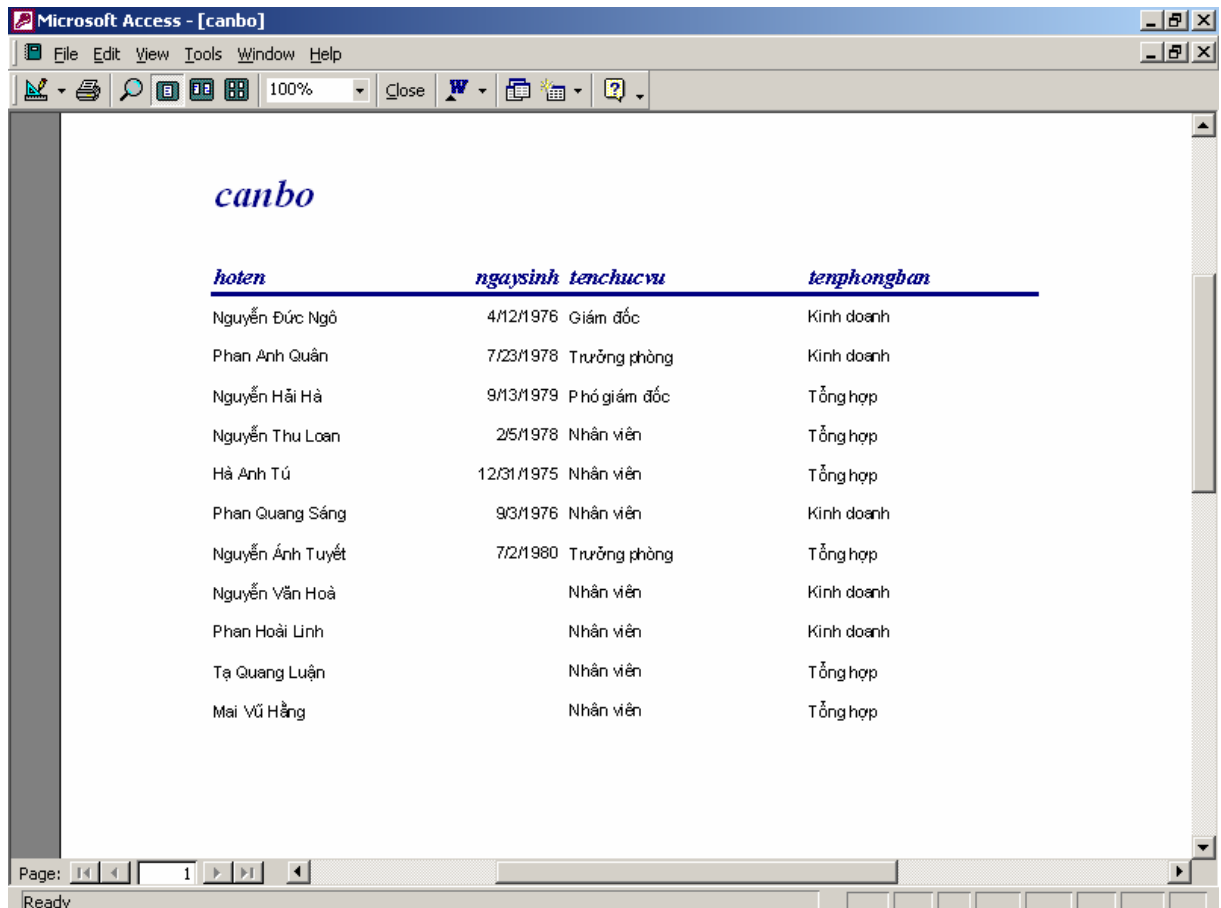


Bao gồm các thông tin sau:

- Gõ vào tiêu đề report cũng như tên report trên hộp **What do you want for your report?**;

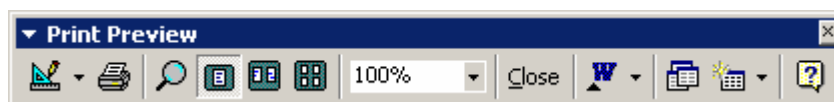
- Chọn **Preview the report** để hiển thị dữ liệu của report ngay sau khi kết thúc; hoặc hiển thị ngay màn hình thiết kế để sửa cấu trúc report khi chọn **Modify the report's design**;
- Cuối cùng nhấn **Finish** để kết thúc các công việc sử dụng report wizard.


Màn hình hiển thị dữ liệu report như sau:




<i>hoten</i>	<i>ngaysinh</i>	<i>tenchucvu</i>	<i>tenphongban</i>
Nguyễn Đức Ngô	4/12/1976	Giám đốc	Kinh doanh
Phan Anh Quân	7/23/1978	Trưởng phòng	Kinh doanh
Nguyễn Hải Hà	9/13/1979	Phó giám đốc	Tổng hợp
Nguyễn Thu Loan	2/5/1978	Nhân viên	Tổng hợp
Hà Anh Tú	12/31/1975	Nhân viên	Tổng hợp
Phan Quang Sáng	9/3/1976	Nhân viên	Kinh doanh
Nguyễn Ánh Tuyết	7/2/1980	Trưởng phòng	Tổng hợp
Nguyễn Văn Hoà		Nhân viên	Kinh doanh
Phan Hoài Linh		Nhân viên	Kinh doanh
Tạ Quang Luận		Nhân viên	Tổng hợp
Mai Vũ Hằng		Nhân viên	Tổng hợp






Từ màn hình này có thể thực hiện rất nhiều các thao tác thông qua thanh công cụ Print Preview như sau:



Nút *Print*  Để in nội dung report ra máy in;

Nút *Design*  Để mở report ra chế độ thiết kế;

Nút *One page*  Để hiển thị report ra màn hình trong từng trang báo cáo;

- Nút *Two page*  Để hiển thị 2 trang báo cáo một trên màn hình;
- Nút *Multi Page*  Để chọn nhiều trang báo cáo có thể hiển thị trên màn hình;
- Nút *Zoom*  Để phóng to, thu nhỏ nội dung report;
- Nút *Office Link*  Để kết xuất (Export) thông tin trên report ra các loại định dạng khác của MS Office như Word, Excel, HTML.
- Nút *Close*  Để đóng màn hình preview report.

Toàn bộ 8 bước sử dụng report wizard liệt kê ở trên, không phải lúc nào cũng cần thực hiện đầy đủ. Với những report đơn giản như trên, chỉ cần thực hiện Bước 1; Bước 2 đã có thể nhấn **Finish** để kết thúc. Tuy nhiên tùy vào yêu cầu in báo cáo mà muốn dừng lại ở bước nào để thiết lập các thông tin cho phù hợp hãy chuyển nhanh đến bước đó.

### 3. Thiết kế report

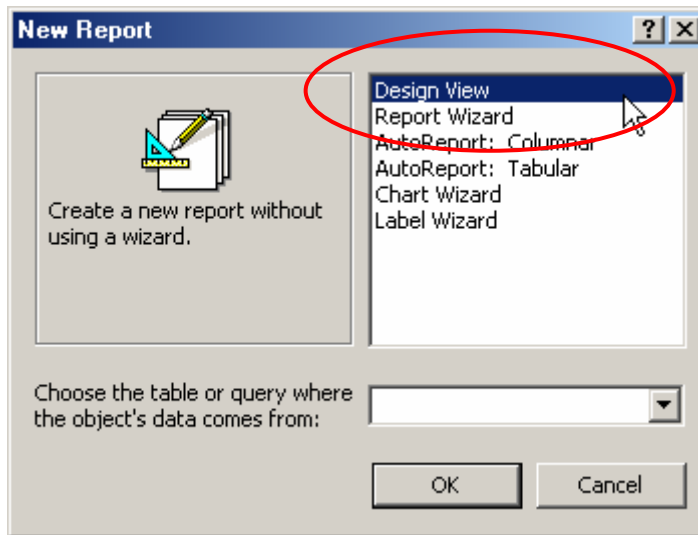
Phần trước đã trình bày các bước dùng Report wizard để có thể thiết kế các mẫu báo cáo in ấn dữ liệu từ CSDL. Đó là cách làm rất dễ dàng và nhanh chóng. Tuy nhiên, các mẫu báo cáo được sinh ra chỉ giới hạn theo một số mẫu mã máy tính đã cung cấp sẵn, không thể in ra được những biểu báo cáo phức tạp theo như những yêu cầu đa dạng của người sử dụng. Report wizard chỉ có thể tạo ra các report in dữ liệu từ các bảng hoặc queries theo cấu trúc đơn giản; trong những trường hợp yêu cầu cấu trúc report phức tạp, hoặc phải in dữ liệu không phải hoàn toàn chỉ từ các bảng và queries hoặc in ấn báo cáo theo các tiêu chí, tham số động nào đó thì không thể đáp ứng được. Trong trường hợp này phải dùng đến Report design view.

Trong chừng mực nào đó, cách làm việc với Report design view gần giống cách làm việc với Form design view. Sau đây là từng bước hướng dẫn sử dụng Report design view để thiết report theo yêu cầu như trên:

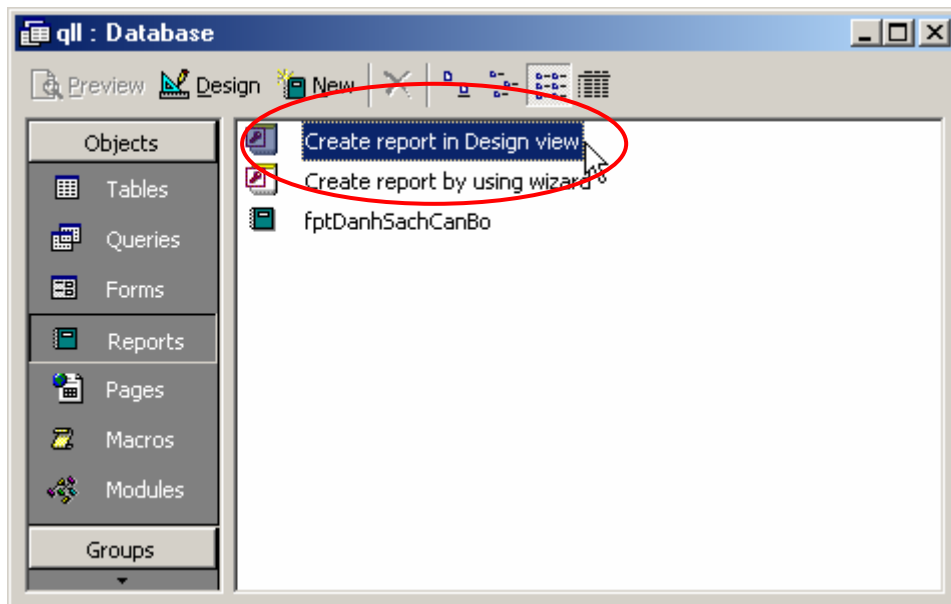
**Bước 1:** Khởi động Report design view:

Từ thẻ Report nhấn nút **New**, chọn **Design view**, nhấn **OK**

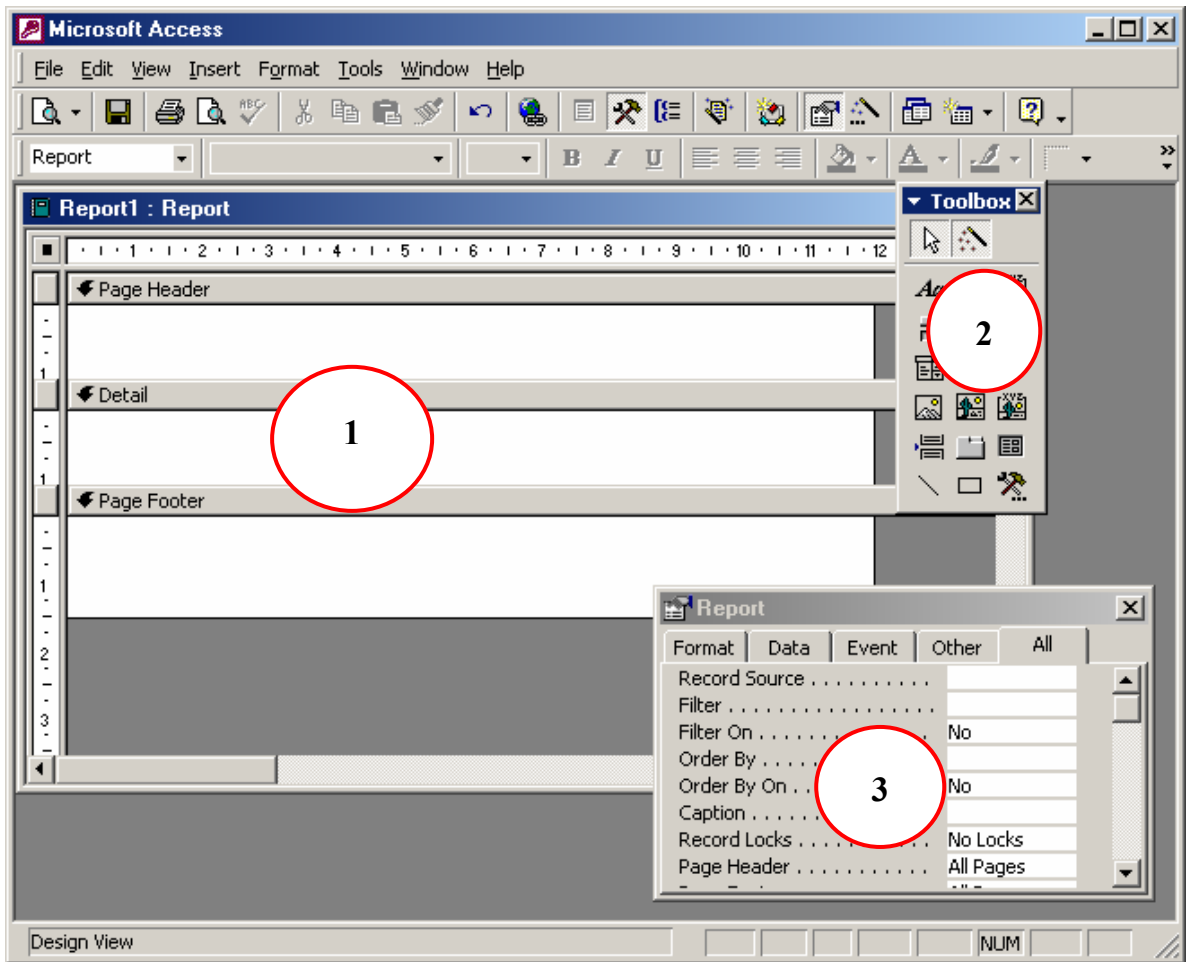




Hoặc nhấn đúp chuột lên biểu tượng *Create report in Design view* trên cửa sổ dự án:



Môi trường làm việc với Report design view xuất hiện:



Có 3 phần quan trọng trên môi trường làm việc này:

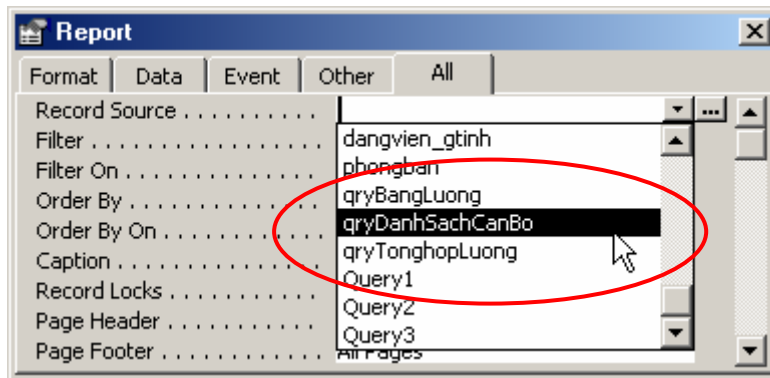
- (1) Cửa sổ thiết kế Report (hình trên tiêu đề cửa sổ này là *Report1: Report*)- nơi để thiết kế nội dung cần in ấn. Nội dung được thiết kế trên cửa sổ này là các đối tượng từ thanh công cụ Toolbox sau khi đã được thiết lập các thuộc tính phù hợp với mục đích;
- (2) Thanh công cụ Toolbox- nơi chứa những đối tượng giúp đưa các thông tin cần thiết lên report. Chức năng và cách làm việc trên thanh công cụ này gần giống với làm việc trên thanh công cụ Toolbox của Form design view;
- (3) Cửa sổ Properties – nơi thiết lập các thuộc tính phù hợp cho các đối tượng trên màn hình thiết kế report.

Chúng ta sẽ được tìm hiểu kỹ hơn các thành phần trên trong các bước tiếp theo khi làm việc cụ thể với chúng.

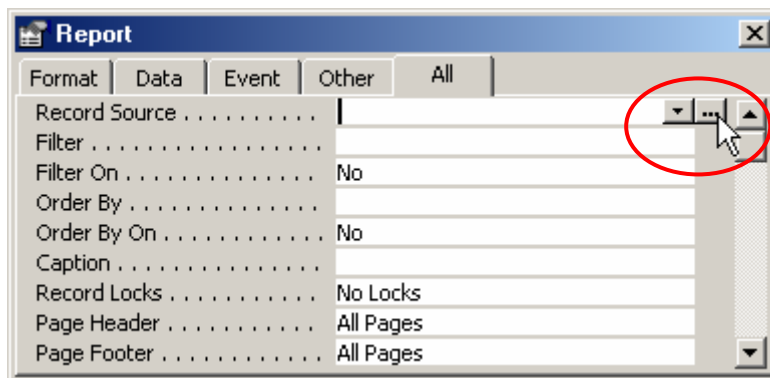
**Bước 2:** Xây dựng nguồn dữ liệu để in ấn cho report.

*Report sẽ in những thông tin gì? Phạm vi như thế nào?* Là các câu hỏi phải được trả lời ở bước này bằng cách thiết lập thuộc tính **Record Source** cho report.

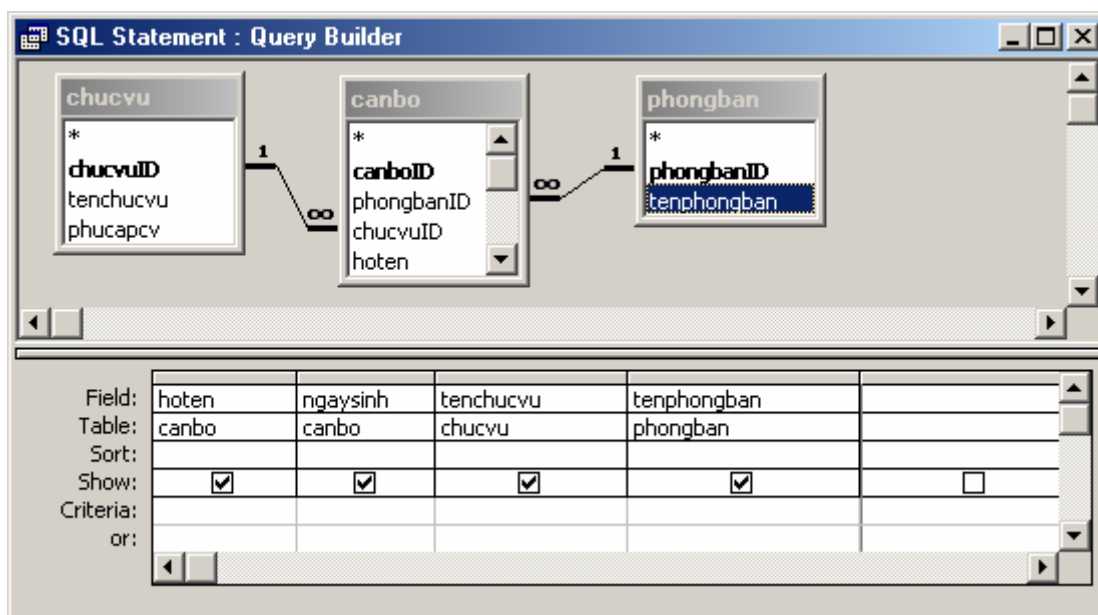
Thông thường, report sẽ in dữ liệu từ một hay nhiều bảng hoặc từ kết quả của một query, tổng quát là từ một query. Query đó có thể được xây dựng sẵn từ danh sách các Queries của dự án (chứa trên thẻ Queries)- khi đó bước này chỉ chọn query cần in tại thuộc tính *Record Source* của report:



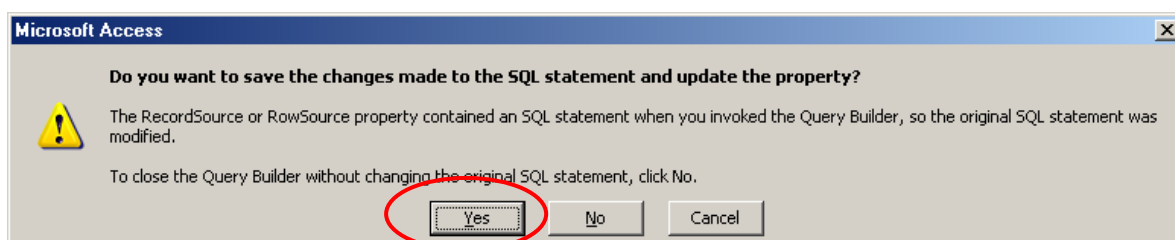
Tuy nhiên query cần in có thể được tạo ra trong chính bản thân report (không hiển thị tên query trên thẻ Queries)- điều này nên làm vì như vậy sẽ đảm bảo sự chắc chắn của report. Khi đó hãy nhấn chuột lên nút ... của thuộc tính *Record Source*:



Một cửa sổ thiết kế query làm nguồn dữ liệu cho report xuất hiện:



Hãy thiết lập các thông tin cần in ra report trên query này. Thiết lập xong nhấn nút đóng cửa sổ thiết kế query (hoặc nhấn phím nóng *Ctrl+W*) và chọn **Yes** trong hộp thoại sau:



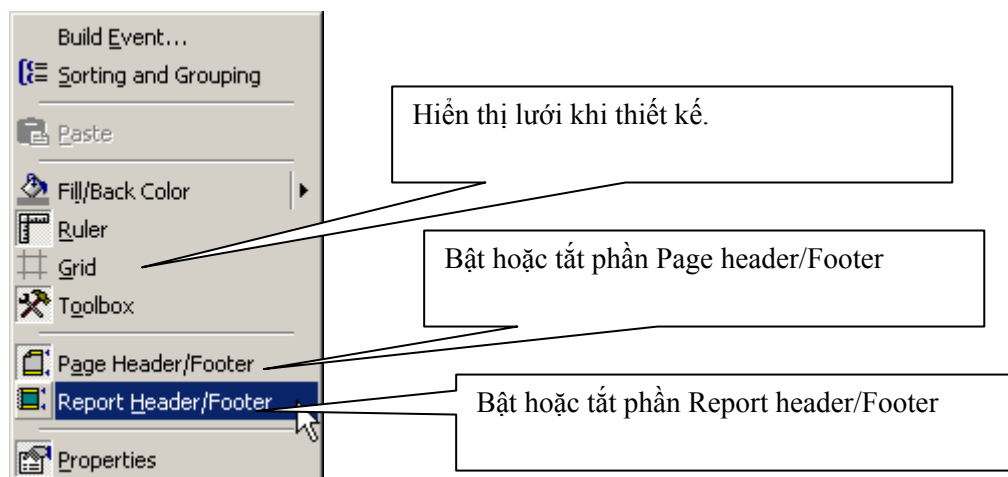
**Bước 3:** Đưa các thông tin lên cửa sổ thiết kế report:

- (1) Có rất nhiều loại thông tin phải đưa lên report. Mỗi khi có ý định đưa một thông tin lên, bạn phải trả lời được 2 câu hỏi:

***Đó là thông tin gì?*** là tiêu đề “*DANH SÁCH CÁN BỘ*”.


***Sẽ đặt thông tin đó lên phần nào của report?*** Đặt lên phần *Page Header*!

- (2) Theo ngầm định, cửa sổ thiết kế report chỉ xuất hiện 3 phần: Page header, Page footer và Detail. Nếu report đang thiết kế yêu cầu có cả Report header và Report footer, hãy hiển thị chúng bằng cách: nhấn phải chuột lên cửa sổ thiết kế report và chọn:




Những phần nào không dùng đến khi thiết kế có thể tắt đi hoặc dùng chuột thu lại diện tích phần đó.

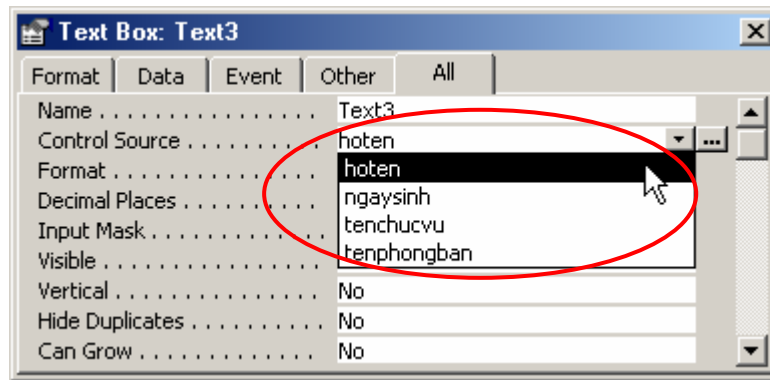
### (3) Sử dụng công cụ **Label**

- Dùng chuột nhấp nút Label  trên thanh công cụ Toolbox thả lên vị trí cần đặt vào trên Report;
- Gõ vào nội dung tiêu đề cần in ra report;
- Có thể sử dụng thanh công cụ Formatting để định dạng hộp dữ liệu này.

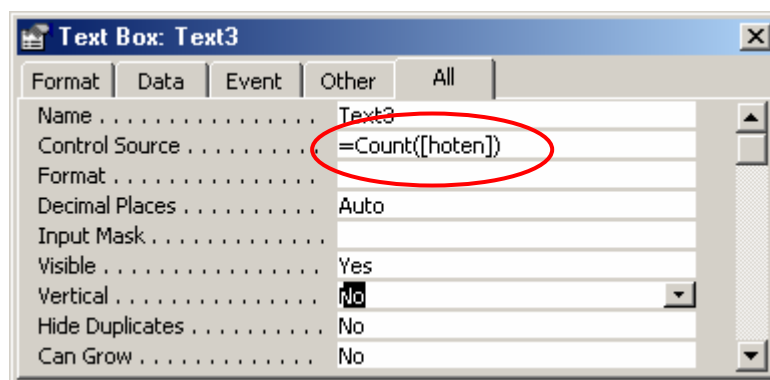
### (4) Sử dụng công cụ **Text box**

Text box là ô dùng hiển thị dữ liệu của một trường dữ liệu (Field) nào đó, hiển thị dữ liệu của một biểu thức (ví dụ tính Tổng tiền chẳng hạn). Cách sử dụng Textbox như sau:

- Dùng chuột nhấp nút Textbox  trên thanh công cụ Toolbox thả lên vị trí cần đặt vào trên Report;
- Gõ vào nguồn dữ liệu sẽ hiển thị lên Textbox ở thuộc tính **Control Source**. Giá trị thuộc tính này có thể:
  - + Hiển thị giá trị một trường: Khi đó hãy chọn trường muốn đưa thông tin vào (hình dưới):



+ Hoặc hiển thị kết quả một biểu thức. Khi đó gõ biểu thức lên thuộc tính này bắt đầu một dấu bằng “=”. Ví dụ:




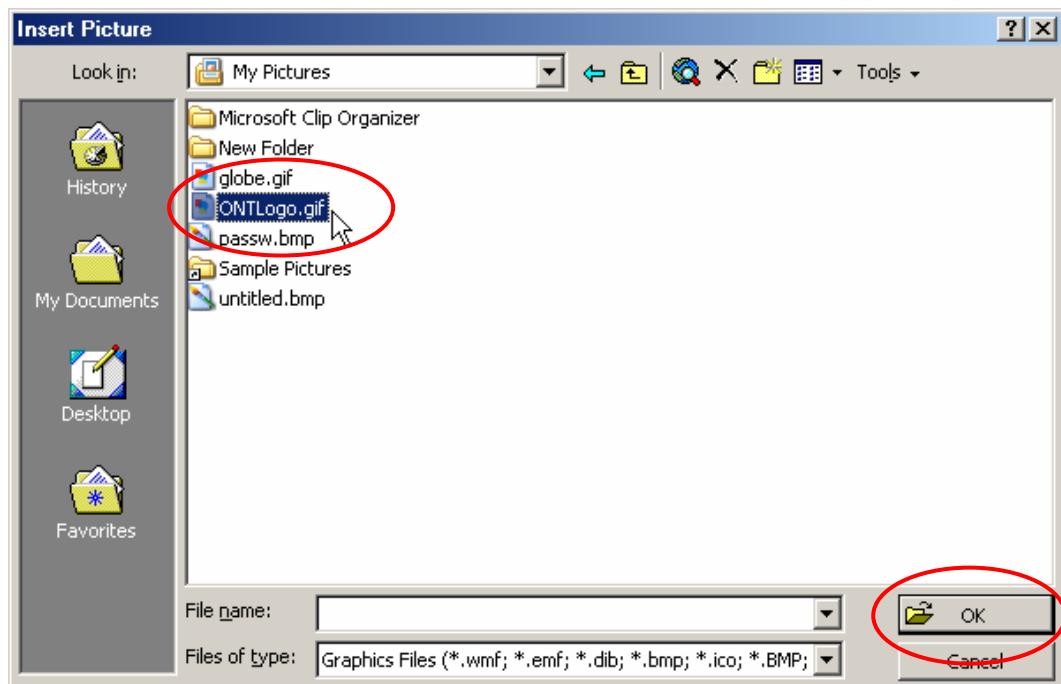
- Có thể sử dụng thanh công cụ Formatting để định dạng hộp dữ liệu này.

#### (5) Sử dụng cửa sổ **Field List**

Danh sách các trường dữ liệu có thể in ra được report thể hiện ở cửa sổ **Field List** (nếu chưa thấy hiển thị mở thực đơn **View | Field List**). Muốn in ra giá trị của trường nào lên vị trí nào của report có thể dùng đối tượng Textbox như giới thiệu ở trên hoặc có thể dùng cửa sổ Field List này bằng cách: dùng chuột kéo trường cần in ra từ cửa sổ Field List thả lên vị trí cần in trên cửa sổ thiết kế report (nên làm theo cách này thay vì dùng Textbox).


#### (6) Sử dụng công cụ **Image**

Công cụ Image  trên thanh công cụ Toolbox giúp đưa ảnh từ các tệp tin ảnh in ra report. Sau khi dùng chuột nhấp nút nút Image từ thanh công cụ đặt lên report, một hộp thoại xuất hiện cho phép tìm đến tệp tin ảnh cần đưa lên report:



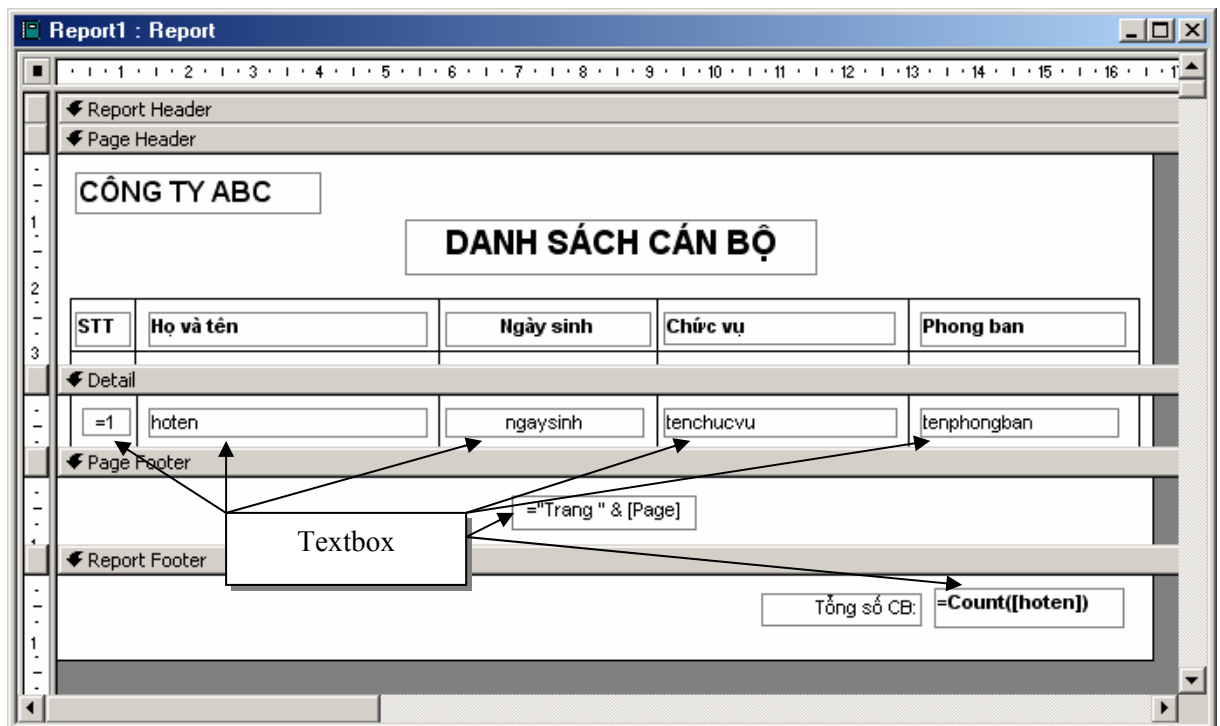
Hãy tìm và chọn tệp ảnh, chọn xong nhấn **OK** để hoàn tất công việc.

#### (7) Sử dụng công cụ **Line**

Công cụ Line  dùng để vẽ các đường thẳng lên Report. Rất hữu hiệu trong việc kẻ bảng biểu.

Sau khi nhấp nút Line trên thanh công cụ, hãy thực hiện kẻ bảng cách di chuột.

Với yêu cầu như trên, bằng cách sử dụng các công cụ như đã hướng dẫn hãy thiết kế một report như sau:



Trong đó:

- Textbox bao gồm các ô như đã đánh mũi tên chỉ dẫn. Trong đó:
  - *hoten, ngaysinh, tenchucvu, tenphongban* được tạo bằng cách kéo từ cửa sổ Field List lên phần Detail;
  - Textbox đếm tổng số cán bộ thiết lập thuộc tính **Control Source** là *=Count([hoten])*, để ở phần Report footer.
  - Textbox để tự động đánh số thứ tự phải thiết lập như sau: **Control Source** là *=1*; thuộc tính **Running sum** là *Over Group*.
- Các hộp chữ còn lại dùng đối tượng Label;
- Toàn bộ bảng biểu sử dụng đối tượng Line. Chú ý các đường kẻ phải được nối khít với nhau. Nếu không khít sẽ tạo ra các khe hở và nét đứt (không liền nét)

#### 4. Report chứa tham số

Report có tham số thực chất là loại Report có khả năng lọc dữ liệu khi in. Ví dụ: bình thường Report in ra danh sách cán bộ ở trên sẽ in danh sách toàn bộ cán bộ trong cơ quan. Bây giờ muốn in danh sách cán bộ một phòng ban nào đó? Lúc này phải cần đến report có tham số.



Có nhiều cách để thiết lập và sử dụng report có tham số như:

- Thiết lập tham số trên Record Source của report;
- Thiết lập tham số trên điều kiện lọc (Where Condition) dùng Macro;
- Thiết lập tham số trong câu lệnh VBA DoCmd;
- ...

Trong phần này, chúng tôi hướng dẫn cách thiết lập tham số trên thuộc tính Record Source của report. Cách dễ làm, không cần am hiểu về lập trình VBA. Cách dùng Macro cũng tốt nhưng không nên dùng. Vì phương pháp lập trình này đã trở nên cứng nhắc. Thực tế, phương pháp dùng câu lệnh DoCmd trên VBA là tốt nhất, nó thể hiện tính linh hoạt và chuyên nghiệp của cách giải quyết vấn đề. Riêng cách này, các bạn có thể tham khảo ở Chương Lập trình CSDL.

Sau đây là cách giải quyết bài toán: In danh sách cán bộ một phòng ban nào đó. Phòng ban cần in được chọn từ Combo box một form như sau:



Sau khi chọn tên một phòng ban từ danh sách, nhấn nút **In danh sách**. Danh sách cán bộ phòng đã chọn sẽ được in ra một report.

### **Cách làm:**

#### **Bước 1:** Tạo Report đáp ứng thông tin cần in

Có thể sử dụng report Wizard hoặc Report Design View để tạo ra một Report in danh sách cán bộ với các thông tin như sau:

The screenshot shows a report window titled "rptDscb : Report". The report layout includes a header section with "CÔNG TY ABC" and "DANH SÁCH CÁN BỘ". Below the header is a text box labeled "Phòng ban:" with the value "tenphongban". The main body of the report is a table with four columns: "STT", "Họ và tên", "Ngày sinh", and "Chức vụ". The table contains one row with the values: "=1", "hoten", "ngaysinh", and "tenchucvu". The footer section contains the text "TRƯỞNG PHÒNG TCCB".

### Bước 2: Tạo form theo yêu cầu bài toán

- Có thể sử dụng Combo Wizard để tạo Combobox lấy ra danh sách các phòng ban từ bảng phongban;
- Có thể sử dụng Command button Wizard để tạo các nút lệnh **Đóng** và **In danh sách**;

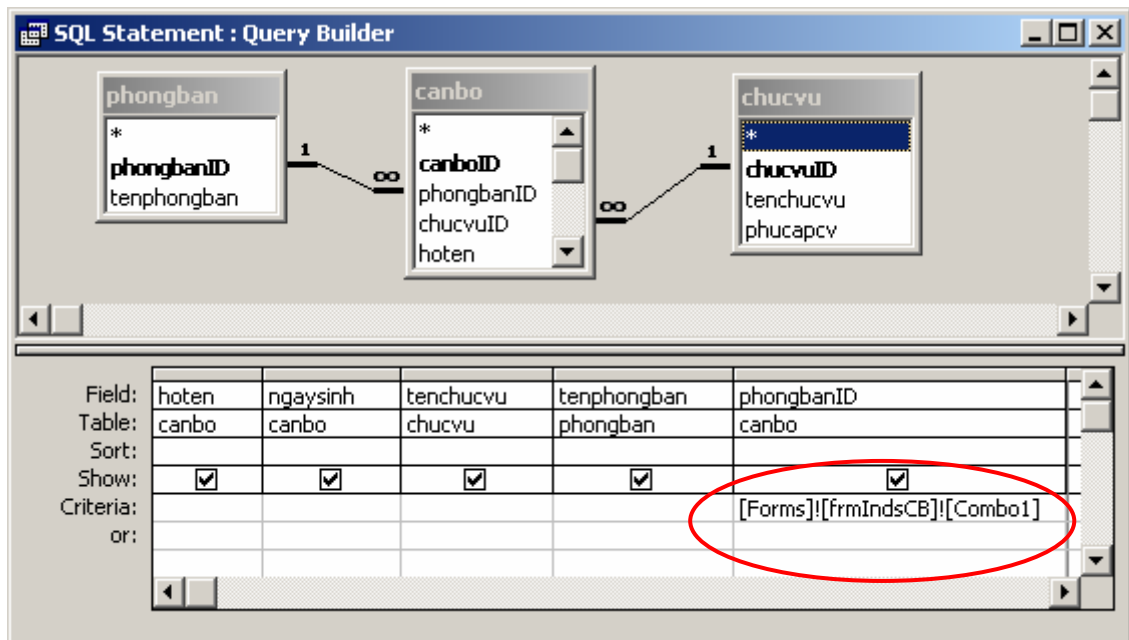
Cuối cùng được form như sau:

The screenshot shows a form window titled "frmIndsCB : Form". The form layout includes a header section with "IN DANH SÁCH CÁN BỘ". Below the header is a text box labeled "Chọn phòng ban:" with a dropdown menu showing "Unbound". Below the dropdown are two buttons: "In danh sách" and "Đóng".

### Bước 3: Thiết lập tham số cho Report

Ở đây trình bày phương pháp thiết lập tham số vào thuộc tính Record Source của Report. Cách làm như sau:

- Mở report đã tạo được ở chế độ Design View (chế độ thiết kế);
- Mở query được thiết lập ở thuộc tính Record Source –nơi tạo nguồn dữ liệu và thiết lập tham số cho trường phongbanID của query như sau:



Trong đó: tham số cho trường phongbanID là tên (Name) của ô Combo box chứa phòng ban được chọn trên form. Cú pháp viết tham chiếu tới một đối tượng trên form như sau:

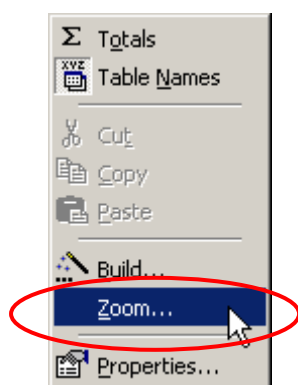
**Forms!<Tên form>!<tên đối tượng>**

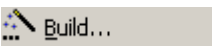

Trong trường hợp này tên ô Combo đó là Combo1 và tên của form (Name của form) là frmIndsCB, cách viết tham số sẽ như sau:

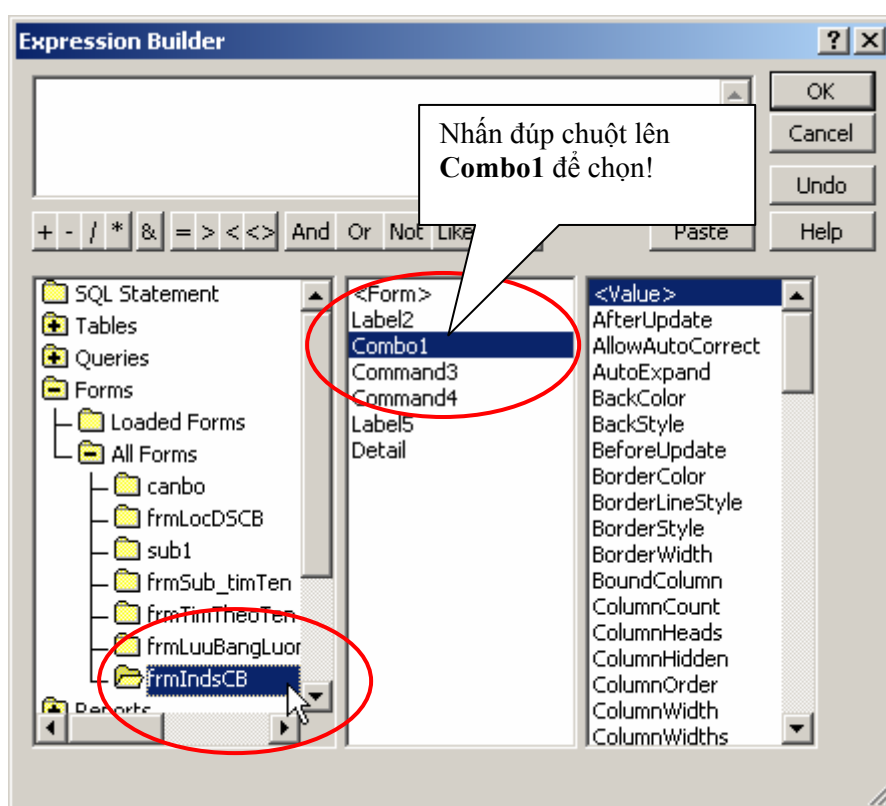
**Forms!frmIndsCB!Combo1**

Nếu cách viết này khó thực hiện, bạn có thể sử dụng tính năng Build Expression có sẵn trên Access để giúp tạo biểu thức này như sau:

Nhấn phải chuột lên ô Criteria của trường *phongbanID*- nơi sẽ gõ vào tham số. Một menu xổ xuống xuất hiện:



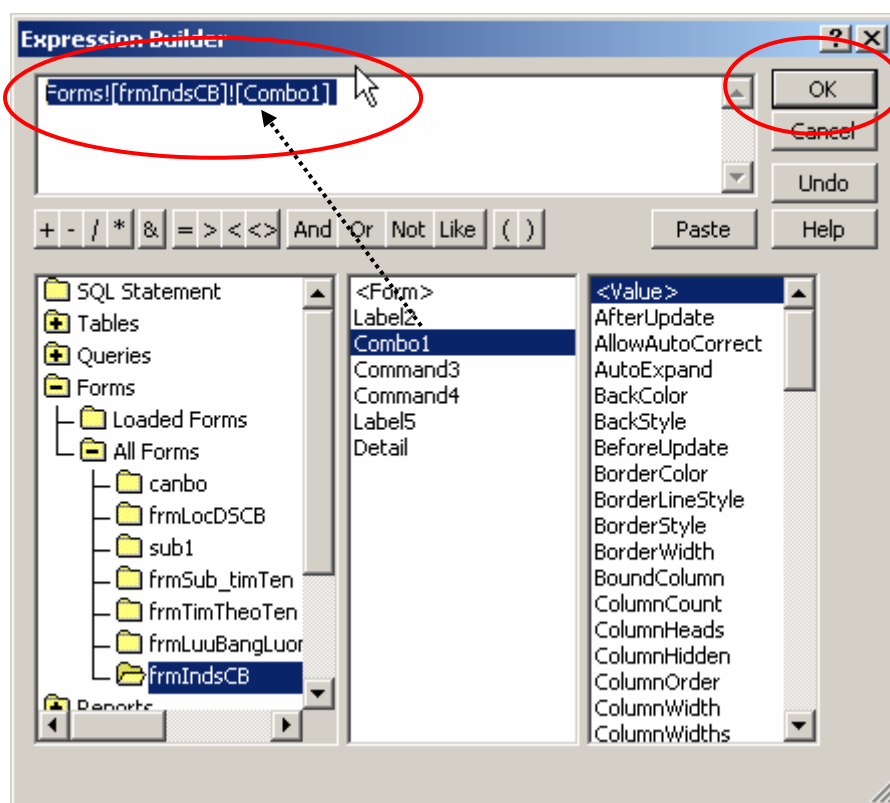
Chọn  hoặc có thể nhấn nút *Build*  trên thanh công cụ Standard, hộp thoại Expression Builder xuất hiện:



Ở cây các đối tượng bên trái hộp thoại, hãy chọn:

**Forms | All Forms | frmIndsCB** – đây là form chứa đối tượng combo box phòng ban cần lọc. Khi đó một danh sách các đối tượng trên form frmIndsCB xuất hiện ở giữa hộp thoại;

Hãy nhấn đúp chuột lên **Combo1** - đối tượng chứa phòng ban cần lọc, được kết quả như hình sau:



Cuối cùng nhấn **OK** để hoàn tất việc tạo tham số cho query bằng hộp thoại Expression Builder.

## Bài tập

Trên CSDL Quản lý lương cán bộ thực hiện các yêu cầu sau:

**Bài số 1:** Thiết kế report in danh sách cán bộ với các thông tin: *canboID*, *hoten*, *ngaysinh*, *gioitinh* (phải ghi rõ là Nam hay Nữ), *tencv*, *tenphan*.

**Bài số 2:** In danh sách cán bộ một phòng ban nào đó như sau:

Sau khi chọn tên một phòng ban, nhấn nút In danh sách cán bộ. danh sách cán bộ phòng ban đã chọn sẽ được in ra một report.

**Bài số 3:** Thiết kế report in bảng lương cán bộ cơ quan, bao gồm các thông tin: *Hoten*, *tencv*, *ngaysinh*, *luongchinh*, *phucapcv*, *dangphi*, *congdoanphi*, *thuclinh*

**Bài số 4:** Thiết kế query in ra bảng tổng hợp cán bộ như sau:

STT	Tên chức vụ	Tổng số cán bộ

**Bài số 5:** Tạo form và report để xem và in bảng lương của các phòng ban như sau:

Chọn phòng ban: kinh doanh In bảng lương

**Bảng lương cán bộ:**

	Họ tên	Ngày sinh	Giới tính	Chức vụ	Lương chính:
▶	Nguyễn Đức Ngô	12/04/1976	Nam	Giám đốc	655200
	Phan Quang Sáng	03/09/1976	Nữ	Nhân viên	504000
	Nguyễn Văn Hoà		Nữ	Nhân viên	420000
	Phan Hoài Linh		Nam	Nhân viên	420000
	Phan Anh Quân	23/07/1978	Nữ	Trưởng phòng	443100
*					

Record: 1 of 5

Sau khi chọn tên một phòng ban, bảng lương các cán bộ phòng đó hiển thị lên sub-form. Nhấn nút In bảng lương, bảng lương riêng phòng ban đó sẽ được in ra một report.

# CHƯƠNG 5

## LẬP TRÌNH VBA CĂN BẢN

Access không những là hệ quản trị CSDL rất mạnh- cụ thể các bạn đã được tìm hiểu rất kỹ ở Chương 1 và Chương 2. Hơn nữa, với những công cụ có sẵn đi kèm như Forms, Report, Macros và đặc biệt là ngôn ngữ lập trình VBA – Access sẽ còn là một công cụ phát triển phần mềm rất mạnh, dễ sử dụng cho các ứng dụng quản lý vừa và nhỏ.

Nội dung chương này sẽ trình bày căn bản về ngôn ngữ lập trình VBA (Visual Basic for Application)- một ngôn ngữ khá quen thuộc đối với những người sử dụng chuyên sâu sản phẩm Microsoft Office. Đây chính là cơ sở quan trọng để các bạn tiếp cận cụ thể chuyên ngành lập trình CSDL sẽ được giới thiệu trong chương tiếp theo.

Qua chương này, học viên sẽ hiểu được môi trường làm việc ngôn ngữ VBA; biết cách sử dụng các cấu trúc lệnh; viết và sử dụng tốt chương trình con; đặc biệt dần làm quen việc lập trình trên các đối tượng ActiveX- sẵn sàng tiếp cận các công cụ lập trình hướng đối tượng trực quan hiện đại như Visual Basic và Visual Basic .NET.

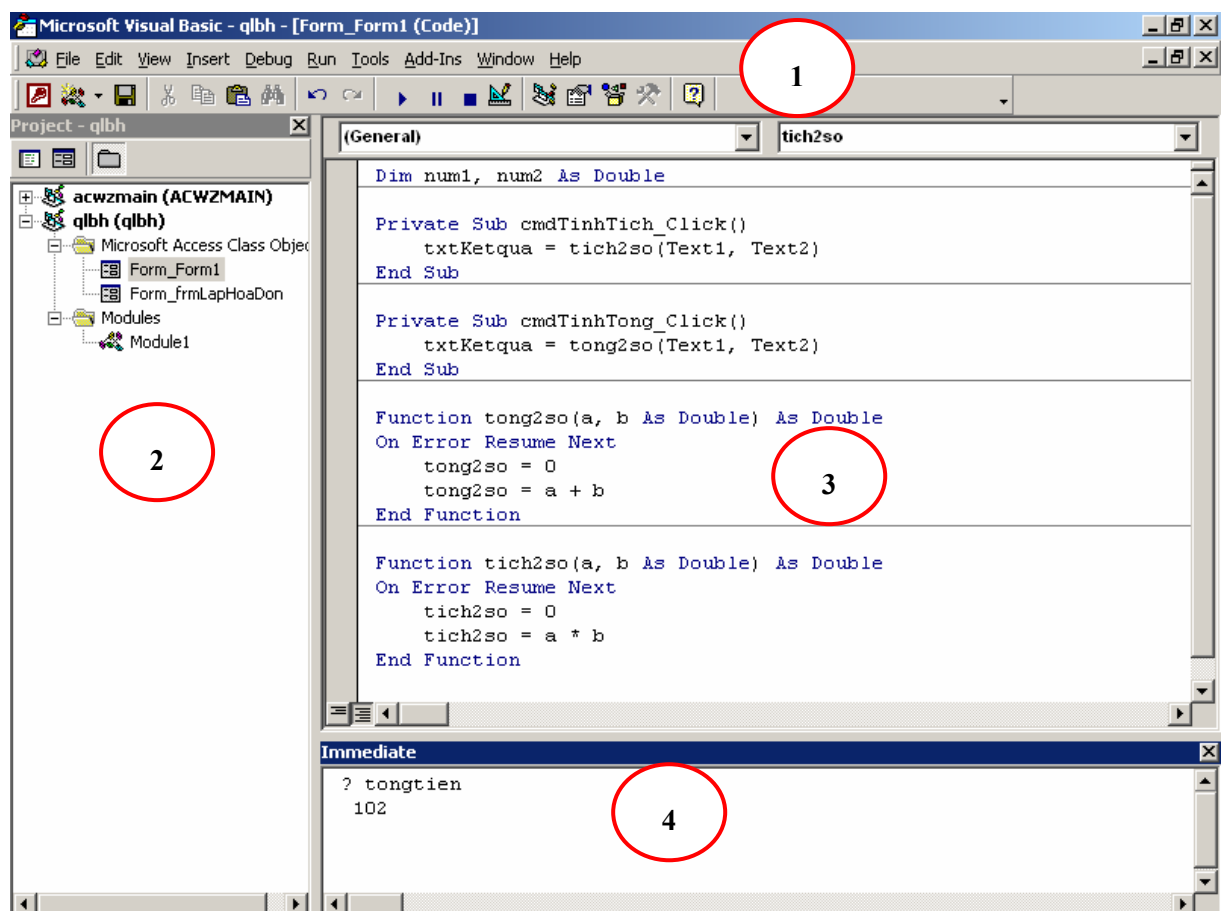


## 1. Môi trường lập trình VBA

Bộ phần mềm tin học văn phòng Microsoft Office phải nói là nổi tiếng nhất thế giới trong lĩnh vực tin học văn phòng. Word cung cấp khả năng chế bản điện tử đẹp đẽ và hiện đại; Excel với khả năng bảng tính điện tử mạnh mẽ; FrontPage với khả năng tạo ra các trang web sống động; Access với khả năng quản trị CSDL;... tất cả các phần mềm đó đã tạo nên sự phổ biến của bộ phần mềm này với hầu hết người dùng máy tính trên toàn thế giới.

Không dừng ở mức ứng dụng có sẵn, bộ phần mềm này còn có một ngôn ngữ lập trình đi kèm VBA – Visual Basic for Application để giúp người dùng có thể tạo ra các tùy biến mạnh hơn, thân thiện hơn với trong công việc của mình. Với Word, Excel bạn hoàn toàn có thể tự tạo ra các macro để tăng tốc độ sử dụng ứng dụng; hơn thế nữa VBA trên Access đã thể hiện tính chuyên nghiệp trong lập trình, phần nào biến được một CSDL đơn giản trở thành những sản phẩm đóng gói thương mại.

Màn hình làm việc ngôn ngữ VBA thường có dạng:



**Trong đó:****(1) Hệ thống thực đơn và thanh công cụ**

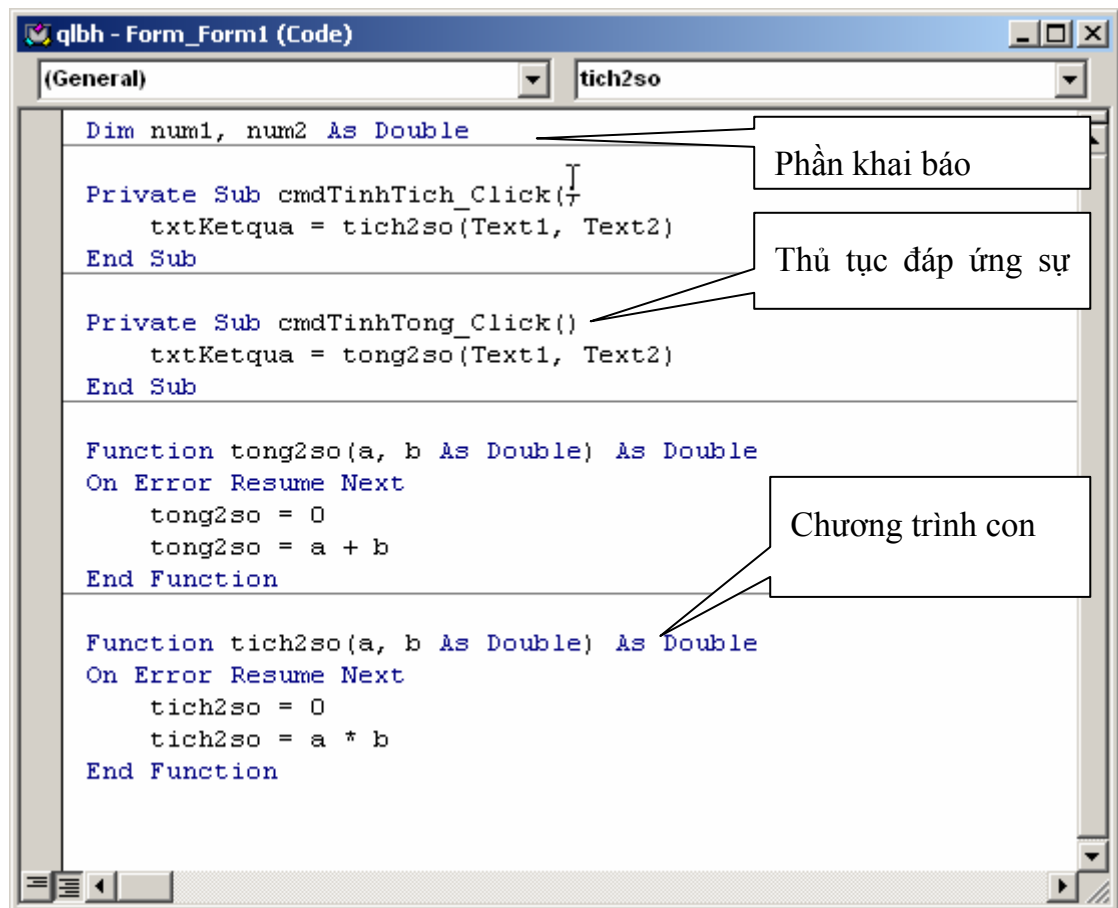
Cũng như bất kỳ môi trường làm việc nào đều có hệ thống thực đơn và thanh công cụ đi kèm. Trên đó có chứa các lệnh để gọi, thi hành hoặc thiết lập các điều khiển cần thiết.

**(2) Cửa sổ Project Explorer;**

Có rất nhiều các thành phần có thể lập trình được bởi VBA như: Forms, Reports, Modules. Cửa sổ Project Explorer là cây phân cấp lớp các đối tượng có chứa mã lệnh VBA, đồng thời giúp lập trình viên dễ dàng trong việc viết (coding) cũng như quản lý các mã lệnh VBA đã viết.

**(3) Cửa sổ viết lệnh;**

Cửa sổ viết lệnh là nơi soạn thảo các dòng lệnh VBA. Mỗi cửa sổ sẽ chứa toàn bộ mã lệnh cho một đối tượng như: Forms, Reports, Modules. Trong mỗi cửa sổ có thể có nhiều phần được viết lệnh, mỗi phần có thể là nội dung một khai báo, một chương trình con, nội dung một thủ tục đáp ứng sự kiện. Ví dụ:



#### (4) Cửa sổ Intermediate

Cửa sổ Intermediate là nơi giúp thi hành trực tiếp một câu lệnh nào đó, rất hữu dụng trong việc gỡ lỗi phần mềm (sẽ quay trở lại vấn đề gỡ rối phần mềm ở cuối chương)

## 2. Các kiểu dữ liệu và khai báo

### 2.1 Các kiểu dữ liệu cơ bản

Cũng như các ngôn ngữ lập trình khác, VBA đều hỗ trợ các kiểu dữ liệu cơ bản. Dưới đây giới thiệu chi tiết về từng kiểu.

#### Boolean

Kiểu lô gíc, tương tự kiểu Boolean trên Pascal. Kiểu này chiếm 2 byte bộ nhớ; chỉ nhận một trong 2 giá trị là: Yes – No hoặc True – False hoặc đôi khi thể hiện

dưới dạng số 0 tương đương với False, True tương ứng với bất kỳ số nào khác 0. Khi lập trình CSDL, kiểu Boolean tương ứng với kiểu Yes/No trong bảng dữ liệu.

### **Byte**

Kiểu số nguyên dương trong phạm vi từ 0..255. Kiểu này chiếm 1 byte bộ nhớ.

### **Integer**

Kiểu nguyên, có giá trị trong khoảng -32768...32767. Kiểu này chiếm 2 bytes bộ nhớ.

### **Long**

Kiểu số nguyên *dài*, có giá trị trong khoảng 2,147,483,648 .. 2,147,483,647. Kiểu này chiếm 4 bytes bộ nhớ.

### **Single**

Kiểu số thực, có giá trị trong khoảng 1.401298E-45 to 3.402823E38. Chiếm 4 bytes bộ nhớ.

### **Double**

Kiểu số thực có độ lớn hơn kiểu Single, có giá trị trong khoảng 4.94065645841247E-324 to 1.79769313486232E308. Chiếm 8 bytes bộ nhớ.

### **Currency**

Kiểu tiền tệ. Bản chất là kiểu số, độ lớn 8 bytes, có giá trị trong khoảng -922,337,203,685,477.5808 to 922,337,203,685,477.5807. Đặc biệt, kiểu này luôn có ký hiệu tiền tệ đi kèm.

## String

Kiểu chuỗi ký tự. Kiểu này tương ứng với kiểu String trong Pascal, tương ứng với kiểu Text trong các trường CSDL Access. Độ lớn tối đa 255 bytes tương đương với khả năng xử lý chuỗi dài 255 ký tự.

## Variant

Variant là kiểu dữ liệu không tường minh. Biến kiểu này có thể nhận bất kỳ một giá trị nào có thể. Ví dụ :

```
Dim a As Variant  
a = 123  
a = "Nguyễn Văn Ngô"
```

Hoàn toàn không có lỗi.

Người ta thường khai báo biến kiểu Variant trong những trường hợp phải xử lý biến đó mềm dẻo. Khi thì biến nhận giá trị kiểu này, khi thì nhận giá trị và xử lý theo kiểu dữ liệu khác.

## Object

Object là một loại biến kiểu Variant, chiếm dung lượng nhớ 4 bytes, dùng để tham chiếu tới một loại đối tượng (Object) nào đó trong khi lập trình. Tất nhiên muốn khai báo biến Object kiểu nào, phải chắc chắn đối tượng đó đã được đăng ký vào thư viện tham chiếu VBA bởi tính năng **Tool | Reference**. Chúng ta sẽ còn trở lại vấn đề này khi lập trình CSDL.

## 2.2 Biến và cách sử dụng biến

### a. Biến – khai báo biến đơn giản

Biến (Variable) là thành phần của một ngôn ngữ lập trình, giúp xử lý dữ liệu một cách linh hoạt và mềm dẻo.

Thông thường trong các ngôn ngữ lập trình, mỗi biến khi tồn tại phải được định kiểu, tức là phải nhận một kiểu dữ liệu xác định. Tuy nhiên trong VBA thì không,

mỗi biến có thể định kiểu (được khai báo trước khi sử dụng) hoặc không định kiểu (không khai báo vẫn sử dụng được). Trong trường hợp này biến đó sẽ tự nhận kiểu giá trị Variant.

Biến có thể được khai báo bất kỳ ở đâu trong phân viết lệnh của VBA. Tất nhiên, biến có hiệu lực như khai báo chỉ bắt đầu từ sau lời khai báo và đảm bảo phạm vi hoạt động như đã qui định. Vì biến trong VBA hoạt động rất mềm dẻo, nên có nhiều cách khai báo biến như:

**Ví dụ 1:** Khai báo biến i kiểu Integer

```
Dim i As Integer
```

**Ví dụ 2:** Khai báo 2 biến i, j kiểu Integer

```
Dim i, j As Integer
```

**Ví dụ 3:** Khai báo biến i kiểu Integer, st kiểu String độ dài 15 ký tự

```
Dim i As Integer, st As String*15
```

**Ví dụ 4:** Khai báo biến i kiểu Variant

```
Dim i As Variant  
'hoặc  
Dim i
```

**Ví dụ 5:** Khai báo biến txt kiểu Textbox

```
Dim txt As TextBox
```

**Ví dụ 6:** Khai báo mảng kiểu String\*30 gồm 46 phần tử

```
Dim Hoten(45) As String * 45
```

**Ví dụ 7:** Khai báo biến mảng 2 chiều  $A(i, j)$  trong đó:  $i = 0..3$  và  $j = 0..4$

```
Dim A(3, 4) As Integer
```

**Ví dụ 8:** Khai báo mảng 3 chiều  $A(i, j, k)$  trong đó:  $i = 1..5$ ;  $j = 4..9$  và  $k = 3..5$

```
Dim A(1 To 5, 4 To 9, 3 To 5) As Double
```

**Ví dụ 9:** Khai báo một mảng động kiểu Variant. Mảng động là mảng không có định chiều dài.

```
Dim MyArray()
```

## b. Phạm vi biến

Như chúng ta đã biết, mỗi biến sau khi được khai báo nó sẽ nhận một kiểu dữ liệu và có một phạm vi hoạt động, tức là lời khai báo biến chỉ có tác dụng trong những vùng đã được chỉ định; ngoài vùng chỉ định đó biến sẽ không có tác dụng, nếu có tác dụng sẽ theo nghĩa khác (biến cục bộ kiểu Variant chẳng hạn).

### **Biến cục bộ:**

Biến cục bộ được khai báo sau từ khoá **Dim**, nó chỉ có tác dụng trong một chương trình con, cục bộ trong một form hoặc một module nào đó. Dưới đây sẽ chỉ ra 3 trường hợp biến cục bộ này:

- Trong một chương trình con, nếu nó được khai báo trong chương trình con đó;
- Trong cả một Form, nếu nó được khai báo trong phần Declarations của Form đó;

- Trong cả một Reports, nếu nó được khai báo trong phần *Decralations* của Report đó;
- Trong cả một Modules, nếu nó được khai báo trong phần *Decralations* của Modules đó;

\* *Biến chỉ có tác dụng sau lệnh khai báo Dim*

### **Biến toàn cục:**

Biến toàn cục được khai báo sau cụm từ khoá **Public**, nó có tác dụng trong toàn bộ chương trình (ở bất kỳ chỗ nào có thể viết lệnh). Loại biến này luôn phải được khai báo tại vùng *Decralations* của một Module nào đó.

### **Ví dụ:**

```
Public Hoten(45) As String * 45
```

Trên một tệp Access, không được phép khai báo trùng tên biến toàn cục. Tuy nhiên tên biến cục bộ vẫn có thể trùng tên biến toàn cục, trong trường hợp đó VBA sẽ ưu tiên sử dụng biến cục bộ trong phạm vi của nó.

## **2.3 Hằng và cách sử dụng hằng**

### **a. Khai báo hằng**

Hằng (Constan) là đại lượng có giá trị xác định và không bị thay đổi trong bất kỳ hoàn cảnh nào. Tương ứng với từng kiểu dữ liệu, sẽ có những hằng tương ứng.

Khai báo hằng số bởi từ khoá **Const**. Sau đây là các ví dụ về khai báo các loại hằng:

**Ví dụ 1:** Hằng a =5 (hằng số)

```
Const a = 5
```

**Ví dụ 2:** Hằng ngày = 24/12/2004 kiểu Date (bao bởi cặp dấu thăng #..#)



```
Const ngay = #24/12/2004#
```

**Ví dụ 3:** Hằng xâu ký tự (bao bởi cặp dấu nháy kép “.”)

```
Const phongban = "Tài vụ"
```

**Ví dụ 4:** Hằng kiểu Logic xác định bởi *True* hoặc *False*

```
Const ok = True
```

## b. Phạm vi hằng

Tương tự như biến, hằng cũng có những phạm vi hoạt động của nó. Hằng được khai báo trong thủ tục nào, hoặc cục bộ trong form, report hoặc module nào sẽ chỉ có tác dụng trong phạm vi đó.

Muốn hằng có phạm vi toàn cục, phải được khai báo sau từ khoá **Public Const**, tại vùng *Declarations* của một module nào đó như sau:

```
Public Const a = 12
```

## 3. Các cấu trúc lệnh VBA

Các cấu trúc lệnh là thành phần cơ bản của mỗi ngôn ngữ lập trình. Thông thường các ngôn ngữ lập trình đều có các cấu trúc lệnh như nhau: lệnh xử lý điều kiện, lệnh lặp biết trước số vòng lặp, lệnh lặp không biết trước số vòng lặp,.. Tuy nhiên cách thể hiện (cú pháp) mỗi cấu trúc lệnh có thể khác nhau tùy thuộc vào mỗi ngôn ngữ lập trình. Hơn nữa, mỗi ngôn ngữ cũng có thể có một số điểm khác biệt, đặc trưng trong mỗi cấu trúc lệnh.

Cũng giống như nhiều ngôn ngữ lập trình hiện đại khác, các cấu trúc lệnh trong VBA đều tuân thủ các nguyên tắc:

- Có cấu trúc: mỗi cấu trúc lệnh đều có từ khoá bắt đầu và một từ khóa báo hiệu kết thúc;
- Thực hiện tuần tự (loại trừ trường hợp đặc biệt thủ tục Goto <Label>);
- Có khả năng lồng nhau;

### 3.1 Cấu trúc IF... END IF

Cấu trúc này thường gọi là *lệnh lựa chọn*. Tức là nếu một điều kiện nào đó xảy ra sẽ là gì, hoặc trái lại có thể làm gì. Trong VBA cú pháp lệnh này như sau:

```
If <điều kiện> Then
    <thủ tục 1>
[ Else
    <thủ tục 2> ]
End If
```

Ý nghĩa lệnh trên là: **nếu** <điều kiện> = True **thì** thực hiện các lệnh trong <thủ tục1>. Trái lại thực hiện các lệnh trong <thủ tục 2>.

Phần trong cặp dấu ngoặc vuông [...] có thể có hoặc không có trong câu lệnh, tùy thuộc vào mục đích xử lý.

**Ví dụ 1:** Kiểm tra và trả lời một số là *chẵn* hay *lẻ*?

```
If so Mod 2 = 0 Then
    MsgBox "Là số chẵn !"
Else
    MsgBox "Là số lẻ !"
End If
```

Cho biết *thang* (số nguyên) rơi vào đầu năm (1..4), giữa năm (5..8) hay cuối năm (9//12)?

```
If thang >=9 Then
    MsgBox "Cuối năm "
Else
    If thang >=5 Then
        MsgBox "Giữa năm "
    Else
        MsgBox "Đầu năm "
    End If
End If
```

### 3.2 Cấu trúc SELECT CASE .. END SELECT

Đây là một loại của cấu trúc lựa chọn. Thông thường hoàn toàn có thể sử dụng If .. End If để thực hiện các xử lý liên quan đến kiểu cấu trúc này, nhưng trong những trường hợp đặc biệt, cấu trúc **Select Case .. End Select** thể hiện được sự tiện dụng vượt trội. Trong VBA cú pháp lệnh này như sau:

```
Select Case <biểu thức>
  Case <giá trị 1>
    <thủ tục 1>
  Case <giá trị 2>
    <thủ tục 2>
  .....
  Case <giá trị n>
    <thủ tục n>
  [Case Else
    <thủ tục n+1>]
End Select
```

**Trong đó:** <Biểu thức> luôn trả về giá trị kiểu vô hướng đếm được như: số nguyên, xâu ký tự, kiểu lô gíc,...

Với cấu trúc này, VBA hoạt động như sau:

(1) Tính giá trị của biểu thức

(2) Kiểm tra <biểu thức> = <giá trị 1> ?

- Nếu đúng thực hiện <thủ tục 1> và kết thúc lệnh, thực hiện lệnh tiếp theo sau từ khoá End Select.

- Nếu sai, thực hiện tiếp việc so sánh <biểu thức> = <giá trị i> tiếp theo và xử lý tương tự qui trình nêu trên.

(3) Trong trường hợp <biểu thức> <> <giá trị i>, i=1..n khi đó có 2 khả năng:

- Nếu có tùy chọn *Case Else* thì VBA sẽ thực hiện <thủ tục n+1>;

- Nếu không có tùy chọn *Case Else*, VBA sẽ không thực hiện bất kỳ thủ tục nào đã liệt kê trong vùng `Select .. End Select` cả mà chuyển tới thực hiện lệnh tiếp theo sau từ khoá `End Select`.

Xét ví dụ sau: Kiểm tra một số nguyên (so) và trả về tên tiếng Anh tháng tương ứng với số nguyên đó (biến thang), ví dụ:

1 - *January*

2 - *February*

...

12 - *December*

>12 - *Không xác định*

Nếu dùng lệnh `If` hoàn toàn có thể đáp ứng được bài toán này, thay vào đó sẽ là một tập hợp 12 lệnh `If .. Else .. End If` như sau:

```
If so = 1 Then
    thang = "January"
Else
    If so = 2 Then
        thang = "Feb"
    Else
        If so = 3 Then
            thang = "Feb"
        Else
            If so = 4 Then
                thang = "Feb"
            Else
                If so = 5 Then
                    thang = "Feb"
                Else
                    If so = 6 Then
                        thang = "Feb"
                    Else
                        If so = 7 Then
                            thang = "Feb"
                        Else
                            If so = 8 Then
                                thang = "Feb"
                            Else
                                If so = 9 Then
                                    thang = "Feb"
                                Else
                                    If so = 10 Then
                                        thang = "Feb"
                                    Else
                                        If so = 11 Then
```

```

thang = "Feb"
Else
  If so = 12 Then
    thang =

  Else
    thang =

  End If
End If
End If
End If
End If
End If
End If
End If
End If
End If
End If
End If
End If
End If
End If

```

Tuy nhiên khi sử dụng **Select Case .. End Select**, cấu trúc sẽ gọn gàng và sáng sủa hơn nhiều. Cụ thể như sau:

```

Select Case so
  Case 1
    thang = "January"
  Case 2
    thang = "January"
  Case 3
    thang = "January"
  Case 4
    thang = "January"
  Case 5
    thang = "January"
  Case 6
    thang = "January"
  Case 7
    thang = "January"
  Case 8
    thang = "January"
  Case 9
    thang = "January"
  Case 10
    thang = "January"
  Case 11
    thang = "January"
  Case 12
    thang = "January"
  Case Else
    thang = "Không xác định"
End Select

```

### 3.3 Cấu trúc FOR ... NEXT

**For... Next** là một cấu trúc lặp biết trước số lần lặp trong VBA, tuy nhiên trong những tình huống đặc biệt, vẫn có thể sử dụng cấu trúc này như cấu trúc không biết trước được số lần lặp.

Cú pháp cấu trúc For...Next như sau:

```
For <biến chạy> = <giá trị 1> To <giá trị 2> [Step <n>]
  <thủ tục>
  [Exit For]
Next
```

#### Trong đó:

- <biến chạy> là biến kiểu vô hướng đếm được, hay dùng nhất là biến kiểu nguyên;
- <giá trị 1>, <giá trị 2> là các giá trị mà biến chạy sẽ nhận và thực hiện dịch chuyển sau mỗi lần lặp. Có thể dịch chuyển đi 1 đơn vị, có thể dịch chuyển đi nhiều đơn vị một lần, có thể dịch chuyển tiến, cũng có thể dịch chuyển lùi- tất cả điều này tùy thuộc vào việc có hay không có tùy chọn [Step <n>];
- Nếu có tùy chọn [Step <n>] biến chạy sẽ dịch n đơn vị sau mỗi lần lặp. Khi đó, nếu n>0 dễ dịch tiến, ngược lại sẽ dịch lùi;
- Mỗi lần lặp, VBA sẽ thực hiện <thủ tục> một lần;
- Trong trường hợp đặc biệt nếu gặp phải lệnh **Exit For** trong vòng lặp, ngay lập tức thoát khỏi lệnh lặp và thực hiện lệnh tiếp ngay sau từ khoá Next. Chính **Exit For** đã làm mất đi tính lặp biết trước được số lần lặp của loại lệnh này.

Tiếp theo là các ví dụ:

**Ví dụ 1:** Tính tổng các số từ 1 đến 50, giá trị được lưu vào biến *tong*.

```
Dim i As Byte
Dim tong As Integer
```

```
tong = 0
For i = 1 To 50
    tong = tong + i
Next
Msgbox tong
```

**Ví dụ 2:** Tính tổng các số chia hết cho 3 từ 1 đến 50, giá trị được lưu vào biến *tong*.

```
Dim i As Byte
Dim tong As Integer

tong = 0
For i = 3 To 50 Step 3
    tong = tong + i
Next
Msgbox tong
```

Lệnh For trong ví dụ này chỉ khác lệnh For ở ví dụ 1 ở chỗ **Step 3**. Vì <giá trị *i*> = 3 là số chia hết cho 3, nên tất cả các giá trị *i* còn lại sẽ chia hết cho 3 (vì  $i = i + 3$ ).

**Ví dụ 3:** Kiểm tra một số nguyên (>2) có phải là nguyên tố hay không?

```
Dim so As Integer
Dim uoc As Integer
Dim nguyento As Boolean

nguyento = True
For uoc = 2 To Int(so / 2)
    If so Mod uoc = 0 Then
        nguyento = False
        Exit For
    End If
Next

If nguyento Then
    MsgBox "là nguyên tố"
Else
    MsgBox "không là nguyên tố !"
End If
```

Giải thuật đơn giản để xác định một số có phải nguyên tố hay không là: xác định xem tất cả các số (*uoc*) có thể trở thành ước của số (*so*) cần kiểm tra. Nếu tìm thấy

một ước thực sự đầu tiên, kết luận ngay không phải số nguyên tố bởi lệnh *nguyento* = *False* và thoát khỏi vòng lặp bằng lệnh **Exit For**; trong trường hợp xét toàn bộ các ước có thể mà không tìm được một số nào là ước thực sự, kết luận đây là số nguyên tố (biến *nguyento* = *True* như giá trị ban đầu)

### 3.4 Cấu trúc WHILE ... WEND

**While ... Wend** là một cấu trúc lặp không biết trước số lần lặp trong VBA. Cấu pháp cấu trúc While...Wend như sau (Wend - viết tắt của cụm từ **While End**):

```
While <điều kiện>
    <thủ tục>
Wend
```

#### Trong đó:

- **While, Wend** là các từ khoá của lệnh lặp;
- Nếu <điều kiện> = *True*, các lệnh trong <thủ tục> sẽ được thực hiện. Thực hiện xong lại quay lên dòng lệnh While để kiểm tra tiếp <điều kiện>;
- Nếu <điều kiện> = *False*, sẽ thoát khỏi vòng lặp và thực hiện lệnh tiếp theo từ khoá **Wend**.

**Chú ý:** Luôn phải chứng minh được rằng, sau một số hữu hạn lần thực hiện <thủ tục>, giá trị của <biểu thức> phải là *False* để thoát khỏi vòng lặp. Trong trường hợp không thể thoát khỏi vòng lặp, có nghĩa người lập trình đã mắc phải *lỗi lặp vô hạn*. Có thể dẫn đến chương trình bị treo.

Các ví dụ:

**Ví dụ 1:** Tính tổng các số chia hết cho 3 trong khoảng từ 1 đến 50

```
Dim i As Byte
Dim tong As Integer

tong = 0
i = 3
While i <= 50
    tong = tong + i
```



```

    i = i + 3
Wend
Msgbox tong

```

**Ví dụ 2:** Ví dụ này thể hiện vòng lặp vô hạn. Lý do có thể là chủ quan, rất đơn giản vì gõ nhầm! Hãy chỉ ra dòng lệnh gõ nhầm và thực hiện sửa cho đúng.

```

Dim i As Byte
Dim tong As Integer

tong = 0
i = 1
While i <= 50
    If i Mod 3 = 0 Then
        tong = tong + i
    End If
    j = i + 1
Wend
Msgbox tong

```

### 3.5 Lệnh DoCmd

Bạn có thể dùng lệnh **DoCmd** để thi hành các công việc thông thường trên Access thông qua môi trường VBA. Ví dụ như: dùng DoCmd để có thể mở form, mở report, query, lọc dữ liệu, thi hành macro xử lý bản ghi, ứng dụng,... Hầu hết các thao tác xử lý trên các đối tượng của Access đều có thể dùng lệnh doCmd để gọi ra thực hiện trong môi trường VBA.

Dưới đây liệt kê một số các phép xử lý của lệnh DoCmd thông dụng:

#### Lệnh đóng một đối tượng

Lệnh này để đóng (Close) hoặc giải phóng đối tượng nào đó ra khỏi bộ nhớ. Hay dùng lệnh này để đóng form đang hoạt động hoặc đóng một report đang preview.

Cú pháp như sau:

**DoCmd.Close** [*ObjectType*], [*ObjectName*], [*SaveOption*]

Trong đó:

**ObjectType** chỉ kiểu đối tượng cần đóng. Cụ thể như sau:

*acForm* Đóng form  
*acReport* Đóng report  
*acQuery* Đóng query  
*acTable* Đóng bảng

**ObjectName** - chỉ tên đối tượng cần đóng;

**SaveOption** - chỉ định tùy chọn ghi lại cấu trúc (nếu có sự thay đổi). Cụ thể:

*SaveNo* Không khi lại  
*SaveYes* Luôn ghi lại  
*SavePromt* Hiện thị hộp thoại nhắc để ghi nếu có sự thay đổi

Ví dụ sau để đóng form frmHoadon, không cần ghi lại cấu trúc nếu có sự thay đổi.

```
DoCmd.Close acForm, "frmHoadon", acSaveNo
```

Đặc biệt, để ra lệnh đóng đối tượng chủ đang mở chỉ cần ra lệnh sau:

```
DoCmd.Close
```

## Lệnh mở form

Là một lệnh hoàn chỉnh để mở và thiết lập môi trường làm việc cho một form. Cú pháp như sau:

```
DoCmd.OpenForm [ObjectName], [ViewMode], [FilterName], [WhereCondition],  
[DataMode], [WindowsMode]
```

Trong đó:

**ObjectName** – tên form muốn mở;

**ViewMode** - chế độ mở. Cụ thể:

*acDesign* Mở form ra chế độ thiết kế  
*acNormal* Mở form ra để thi hành

**FilterName** - Đặt lọc

**WhereCondition** - Giới hạn các bản ghi trong nguồn dữ liệu

**DataMode** - thiết lập chế độ dữ liệu trên form, cụ thể:

**WindowsMode** - thiết lập kiểu cửa sổ form là:

*acDialog* Kiểu hộp thoại  
*acWindowsNormal* Kiểu cửa sổ bình thường

Ví dụ: Dưới đây là lệnh mở form lập hoá đơn bán hàng (frmLapHoaDon), trong đó chỉ hiển thị nội dung của hoá đơn có mã "HD0035"

```
DoCmd.OpenForm "frmLapHoaDon", , , "hoadonID = 'HD0035'"
```

## Lệnh mở report

Là một lệnh hoàn chỉnh để mở và thiết lập môi trường làm việc cho một report. Cú pháp như sau:

**DoCmd.OpenReport** [*ObjectName*], [*ViewMode*], [*FilterName*], [*WhereCondition*], [*DataMode*], [*WindowsMode*]

Trong đó:

**ObjectName** – tên Report muốn mở;

**ViewMode** - chế độ mở. Cụ thể:

*acDesign* Mở Report ra chế độ thiết kế  
*acNormal* Mở Report ra để thi hành

**FilterName** - Đặt lọc

**WhereCondition** - Giới hạn các bản ghi trong nguồn dữ liệu

**DataMode** - thiết lập chế độ dữ liệu trên Report, cụ thể:

**WindowsMode** - thiết lập kiểu cửa sổ Report là:

*acDialog* Kiểu hộp thoại  
*acWindowsNormal* Kiểu cửa sổ bình thường

Ví dụ: Dưới đây là lệnh **Preview report** để in ra hoá đơn bán hàng (rptHoaDon), trong đó chỉ hiển thị nội dung của hoá đơn hiện tại trên một form (ô chứa mã hoá đơn là *txtHoadonID*)

```
DoCmd.OpenReport "rptHoadon", , , "hoadonID = '" + txtHoadonID + "
```

## Lệnh thi hành câu lệnh SQL

Dùng để thi hành một lệnh SQL. Cú pháp như sau:

### DoCmd.RunSQL

Giả sử trên bảng canbo có thêm trường *luongchinh*. Lệnh sau đây sẽ cập nhật giá trị cho trường này thông qua lệnh SQL cập nhật dữ liệu:

```
DoCmd.RunSQL "UPDATE canbo SET luongchinh = hessoluong*290000"
```

Hoặc dưới đây là lệnh xoá bỏ những cán bộ đến tuổi nghỉ hưu ra khỏi CSDL:

```
DoCmd.RunSQL "DELETE * FROM canbo " _
+ " WHERE (Year(Date())-Year([ngaysinh])>=60 AND gioitinh=Yes) "
+ " OR (Year(Date())-Year([ngaysinh])>=55 AND gioitinh=No) "
```

## 4. Chương trình con

Chương trình con là một đơn vị mã lệnh VBA, nó có thể chứa tập hợp các câu lệnh nhằm thao tác, tính toán hoặc điều khiển mục đích hoặc dữ liệu nào đó. Trong VBA có 2 loại chương trình con:

- Chương trình con dạng thủ tục, được khai báo bởi từ khoá Sub;
- Chương trình con dạng hàm, được khai báo bởi từ khoá Function.

Về bản chất, 2 loại chương trình con trên đều như nhau: khai báo, tham số và truyền tham số. Tuy nhiên, điểm khác nhau cơ bản là:

- Function luôn trả về một giá trị kiểu vô hướng chuẩn, ví dụ: hàm Date() - trả về giá trị ngày hiện tại kiểu Date. Trong Access đã sẵn có rất nhiều các hàm tính toán (tham khảo ở trang ...), chúng được gọi là các build-in function. Hơn nữa, người dùng hoàn toàn có thể tạo ra các hàm để sử dụng cho các mục đích riêng loại hàm này gọi là user-define function;
- Còn Sub thì không, nó chỉ thực hiện một số các công việc. Tất nhiên những công việc này hoàn toàn có thể làm thay đổi dữ liệu theo mong muốn trong chương trình. Cũng như Function, Access và VBA sẵn có một thư viện các thủ

tục; hơn nữa người dùng cũng có thể tự tạo thêm những thủ tục mới phục vụ việc xử lý dữ liệu theo mục đích riêng. Đặc biệt, Access còn định nghĩa thủ tục đáp ứng sự kiện. Thủ tục này sẽ được tự động gọi ra khi sự kiện đáp ứng bị ảnh hưởng. Chúng ta sẽ trở lại nội dung này qua các ví dụ lập trình VBA.

Tùy từng tính huống cụ thể sẽ lựa chọn sử dụng Function hoặc Sub.

## 4.1 Chương trình con dạng hàm

Cú pháp

```
Function <tên hàm>([<danh sách các tham số>]) As <kiểu DL hàm>
    <thủ tục>
End Function
```

**Trong đó:**

- **Function, End Function** là các từ khoá bắt buộc khai báo cấu trúc một chương trình con dạng hàm;
- <tên hàm> là tên gọi hàm định khai báo. Tên không được chứa dấu cách (space) và các ký tự đặc biệt;
- <danh sách các tham số> - danh sách các tham số cần thiết cho hàm. Có hay không có danh sách này tùy thuộc vào hàm cần định nghĩa;
- <kiểu DL hàm> - kiểu dữ liệu mà hàm sẽ trả lại. Phần này bắt buộc phải được khai báo với mỗi hàm;
- <thủ tục> - thân chương trình con. Trong đó câu lệnh <tên hàm> = <biểu thức> phải xuất hiện ít nhất một lần trong thủ tục. Câu lệnh này có tác dụng gán giá trị cho hàm.

Nếu không có từ khoá Public trước Function, hàm đó chỉ có tác dụng cục bộ: trong một module, trong một report hoặc trong một form. Khi có từ khoá Public trước Function, hàm sẽ có tác dụng toàn cục. Tức là có thể sử dụng bất kỳ nơi nào trên tệp Access đó. Tất nhiên, tất cả những gì khai báo là Public phải được khai báo trong phần **Declarations** của một Module nào đó.

Các ví dụ:

**Ví dụ 1:** hàm tính tổng 2 số

```

Function Tong2So(a, b As Double) As Double
    Tong2So = a + b
End Function

```

**Ví dụ 2:** hàm kiểm tra một số có phải là nguyên tố hay không?

```

Function laNguyenTo(so As Integer) As Boolean
    Dim uoc As Integer

    laNguyenTo = True
    If so > 2 Then
        For uoc = 2 To Int(Sqr(so))
            If so Mod uoc = 0 Then
                laNguyenTo = False
                Exit For
            End If
        Next
    End If
End Function

```

Ví dụ trên có sử dụng đến:

- hàm *Int(number)* – hàm lấy phần nguyên của một số;
- hàm *Sqr(number)* – hàm lấy căn bậc hai một số

**Ví dụ 3:** hàm tách tên trong chuỗi họ và tên.

Đây là một bài toán gặp phải rất nhiều trong thực tế. Cụ thể bài toán giải quyết vấn đề sau:

Nếu biết họ tên là *Nguyễn Sơn Hải*, hàm sẽ tách ra được tên là *Hải*. Toàn bộ mã lệnh hàm như sau:

```

Function GetTen(hoten As String) As String
    Dim pos As Integer

    pos = 1
    If InStr(pos, Trim(hoten), " ") = 0 Then
        GetTen = hoten
        Exit Function
    End If
    While InStr(pos + 1, Trim(hoten), " ") > 0

```

```

        pos = Instr(pos + 1, Trim(ten), " ")
    Wend

    GetTen = Mid(hoten, pos)
End Function

```

**Ví dụ 4:** Hàm dùng so sánh 2 chuỗi kiểu chữ TCVN3 chúng tôi đưa ra dưới đây là một tham khảo rất tốt. Trong Word, Access cũng như các bảng dữ liệu tiếng Việt có dấu trên máy tính, việc sắp xếp chuỗi ký tự là một bài toán mà người Việt phải giải quyết. Ví dụ, dưới đây là một danh sách trên Word:

STT	Tên
1	Quang
2	Đức
3	Đoàn
4	Băng
5	Bang
6	An
7	Ân

Sau khi sử dụng tính năng sắp xếp (Sort) của Word theo cột Tên theo thứ tự tăng dần, được danh sách kết quả như sau:

STT	Tên
7	Ân
3	Đoàn
2	Đức
6	An
4	Băng

5	Bang
1	Quang

Mà danh sách sắp xếp đúng phải là:

STT	Tên
6	An
7	Ân
5	Bang
4	Bảng
3	Đoàn
2	Đức
1	Quang

Hàm **Mahoa** dưới đây sẽ giúp qui đổi một chuỗi tiếng Việt chuẩn TCVN3 (bộ phong ABC) về dạng không dấu. Muốn sắp xếp hay so sánh vị thứ các chuỗi, hãy so sánh các chuỗi không dấu được chuyển đổi bởi hàm **Mahoa** này.

```
Private Function MahoaTCVN3(Ckt As String)
    Dim kq, kti As String
    Dim vt1, vt2, i As Integer
    Dim Cgoc1, Cma1 As String, Cgoc2, xd, Cma2 As String

    Cgoc1 =
    "àáâãäåæçèéêëìíîïðòóôõöøùúûüýÿ"
    "àáâãäåæçèéêëìíîïðòóôõöøùúûüýÿ"
    Cma1 = "abadafaparazblbnbpbcdbccllblcldlelfla"
    Cgoc2 =
    "AaĂăÂâBbCcDdĐđEeÊêFfGgHhIiJjKkLlMmNnOoÔôƠơPpQqRrSsTtUuƯưVvWwXxYyZz"

    Cma2 =
    "aaabacadaeafagahaiajakalamanaoapaqarasatauavawaxayazbabbbcbdbefb
    gbhbibjkbklbmbnbpqbqbrbsbtbubvbvbwxbvzbzcccbcccdcefcgchcicjcckclcmcn"
```



```

kq = ""
xd = ""

For i = 1 To Len(Ckt)
    kti = Mid(Ckt, i, 1)
    vt1 = InStr(Cgoc1, kti)
    If vt1 <> 0 Then
        kq = kq & Mid(Cma1, 1 + ((vt1 - 1) \ 6) * 2, 2)
        xd = xd & Mid(Cma1, 25 + ((4 + vt1) Mod 6) * 2, 2)
    Else
        vt2 = InStr(Cgoc2, kti)
        If vt2 <> 0 Then
            kq = kq & Mid(Cma2, (vt2) * 2 - 1, 2)
        Else
            kq = kq + kti
        End If
    End If
Next i
MahaTCVN3 = kq & xd
End Function

Function Maha(Ckt As String) As String
    Dim vt1 As Integer
    Dim kq, Ctam As String
    Ckt = Ckt & " "
    kq = ""
    vt1 = InStr(Ckt, " ")

    Do While vt1 <> 0
        Ctam = Trim(Left(Ckt, vt1 - 1))
        Ckt = Right(Ckt, Len(Ckt) - vt1)
        kq = MahaTCVN3(Ctam) & " " & kq
        vt1 = InStr(Ckt, " ")
    Loop
    Maha = kq
End Function

```

## 4.2 Chương trình con dạng thủ tục

Cú pháp

```

[Public] [Private] Sub <tên CTC>([<danh sách các tham số>])
    <thủ tục>
End Sub

```

**Trong đó:**

- **Sub, End Sub** là các từ khoá bắt buộc khai báo cấu trúc một chương trình con dạng thủ tục;
- *<tên CTC>* là tên gọi thủ tục định khai báo. Tên không được chứa dấu cách (space) và các ký tự đặc biệt;
- *<danh sách các tham số>* - danh sách các tham số cần thiết cho thủ tục. Có hay không có danh sách này tùy thuộc vào thủ tục cần tạo
- *<thủ tục>* - thân chương trình con.

Nếu không có từ khoá Public trước Sub, thủ tục đó chỉ có tác dụng cục bộ: trong một module, trong một report hoặc trong một form. Khi có từ khoá Public trước Sub, thủ tục sẽ có tác dụng toàn cục. Tức là có thể sử dụng bất kỳ nơi nào trên tệp Access đó. Tất nhiên, tất cả những gì khai báo là Public phải được khai báo trong phần Declarations của một Module nào đó.

Các ví dụ:

#### **Ví dụ 1:** Thủ tục tính tổng hai số

```
Sub tong2so(a, b As Double)
    tong = a + b
```

*'chú ý: tong- là biến được khai báo toàn cục*  
End Sub

#### **Ví dụ 2:** Cũng là tính tổng, nhưng thủ tục sau đây không có ý nghĩa gì!

```
Sub tong2so(a, b As Double)
    Dim tong As Double
```

```
tong = a + b
```

*'chú ý: tong- là biến được khai báo toàn cục*  
End Sub

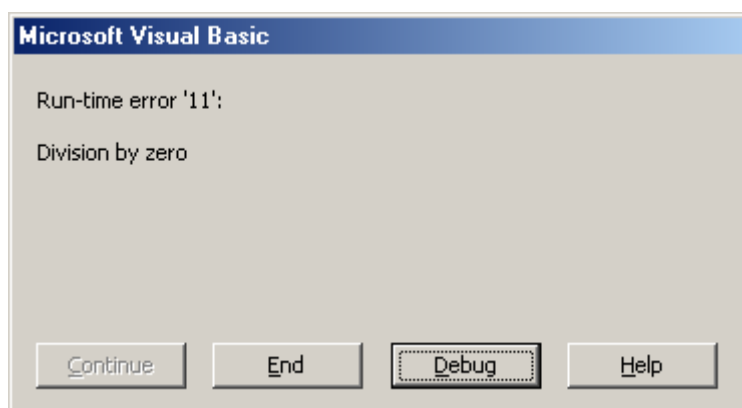
Vì sao? Vì biến *tong* được khai báo cục bộ trong CTC *tong2so*, nên khi CTC này kết thúc, biến *tong* cũng bị giải thoát khỏi bộ nhớ luôn. Không gây ảnh hưởng gì đến dữ liệu cũng như thể hiện của chương trình.

## 5. Kỹ thuật xử lý lỗi

Xử lý lỗi là kỹ thuật rất quan trọng trong lập trình. Đã lập trình thì khó tránh khỏi lỗi (Errors). Có rất nhiều nguyên nhân gây ra lỗi; các nguyên nhân này có thể được lường trước hoặc không được lường trước. Kỹ thuật xử lý lỗi bao gồm các kỹ năng phát hiện và xử lý các tình huống khi chương trình gây lỗi.

### 5.1 Xử lý lỗi

Là việc xử lý khi đang lập trình gặp phải lỗi. Thông thường khi chạy thử chương trình trong lúc đang xây dựng phần mềm nếu gặp phải lỗi, sẽ xuất hiện hộp thoại thông báo lỗi có dạng:




Thông thường một hộp thoại thông báo lỗi gồm 2 thành phần:


- Thành phần báo lỗi bao gồm:

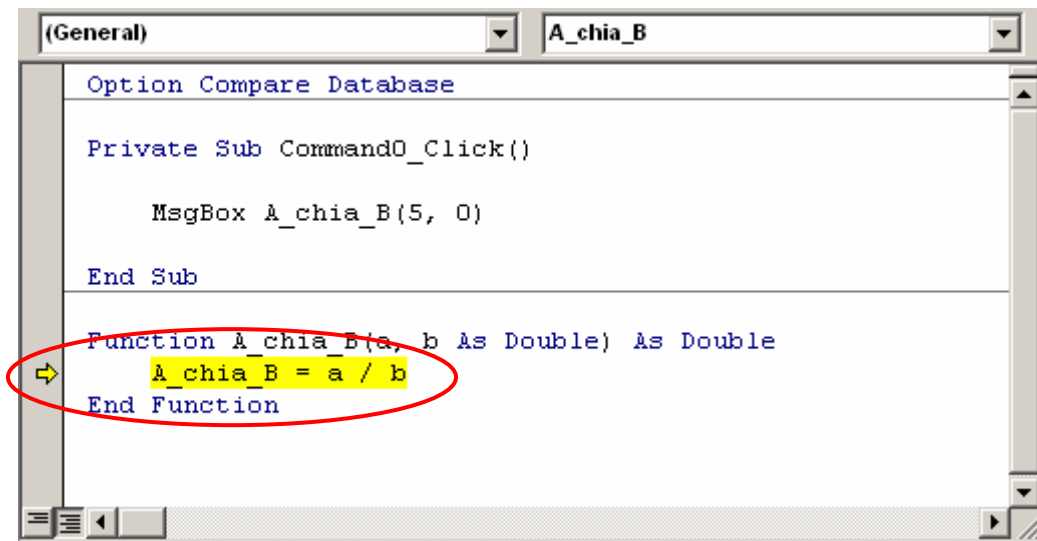
+ **Mã số lỗi** - Mỗi lỗi mà VBA có thể kiểm tra được đều có một mã số, được hiển thị ở dòng thông báo: *Run-time error 'mã số lỗi'*: Ví dụ trên là : *Run-time error '11'*:

+ **Tên lỗi**. Ở ví dụ trên tên lỗi là: *Division by zero* - lỗi sai kiểu dữ liệu.

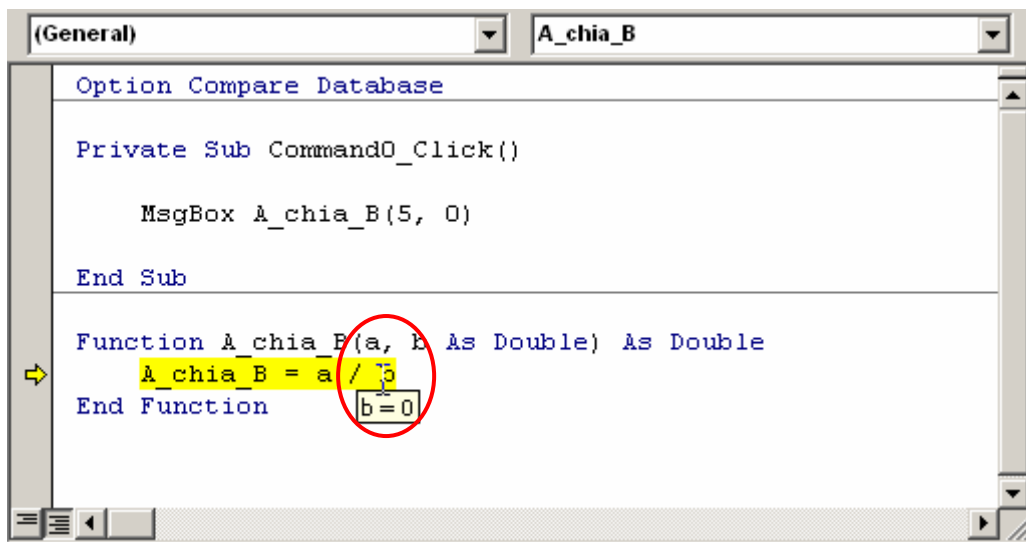
- Thành phần xử lý lỗi gồm 2 nút lệnh:

+ Nút  - để dừng ngay chương trình, chuyển về chế độ **Design** - thiết kế bình thường;

+ Nút  - để dừng chương trình chuyển về chế độ **Break** - sửa lỗi trực tiếp. Khi đó câu lệnh lỗi sẽ được tô bởi màu nền vàng cho phép người lập trình có thể sử dụng mã chương trình:

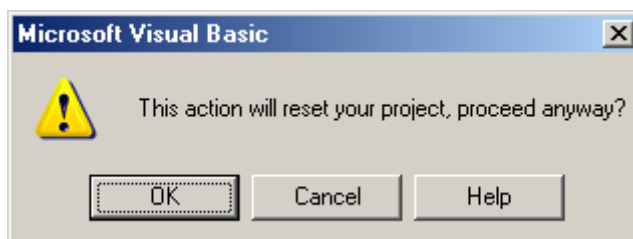


Khi dịch chuột lên một biến nào đó, giá trị biến sẽ được hiển thị dưới dạng **Tool tip**.



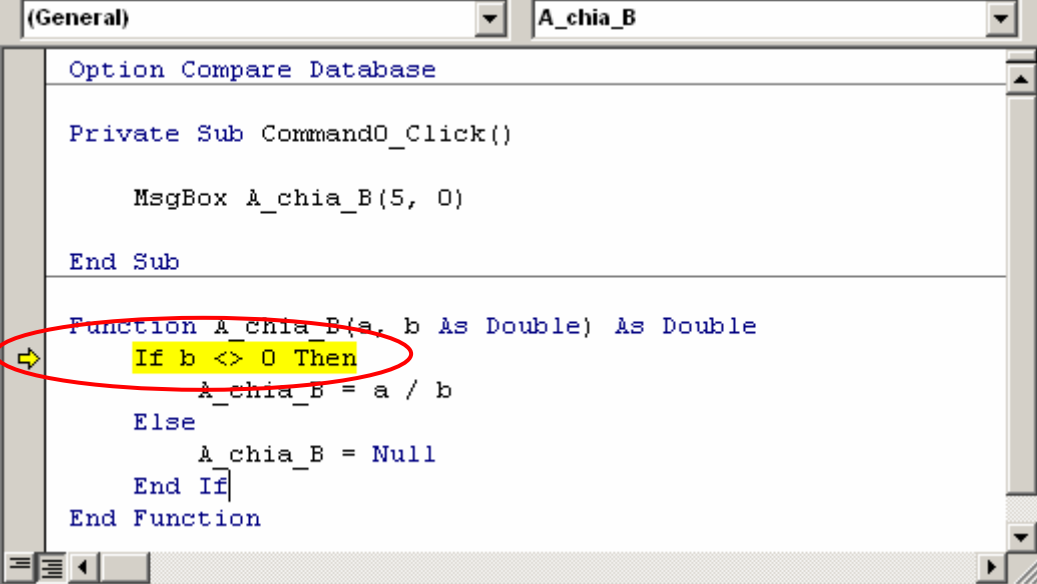
Hình trên khi di chuột lên biến b, giá trị biến b xuất hiện dưới dạng Tool tip (giá trị b = 0).

Sau khi chọn nút **Debug**, bạn hoàn toàn có thể thực hiện sửa mã lệnh trong chương trình. Tuy nhiên, trong một số trường hợp khi sửa mã lệnh VBA sẽ hỏi:



Điều này có nghĩa: việc sửa đổi mã lệnh vừa rồi, VBA yêu cầu phải trở về chế độ thiết kế bình thường nếu nhấn **Ok**; trái lại nhấn **Cancel**- việc thay đổi mã lệnh sẽ không được chấp nhận.

Sau khi thực hiện sửa mã lệnh, bạn có thể yêu cầu VBA thực thi tiếp chương trình.





```
(General) A_chia_B
Option Compare Database

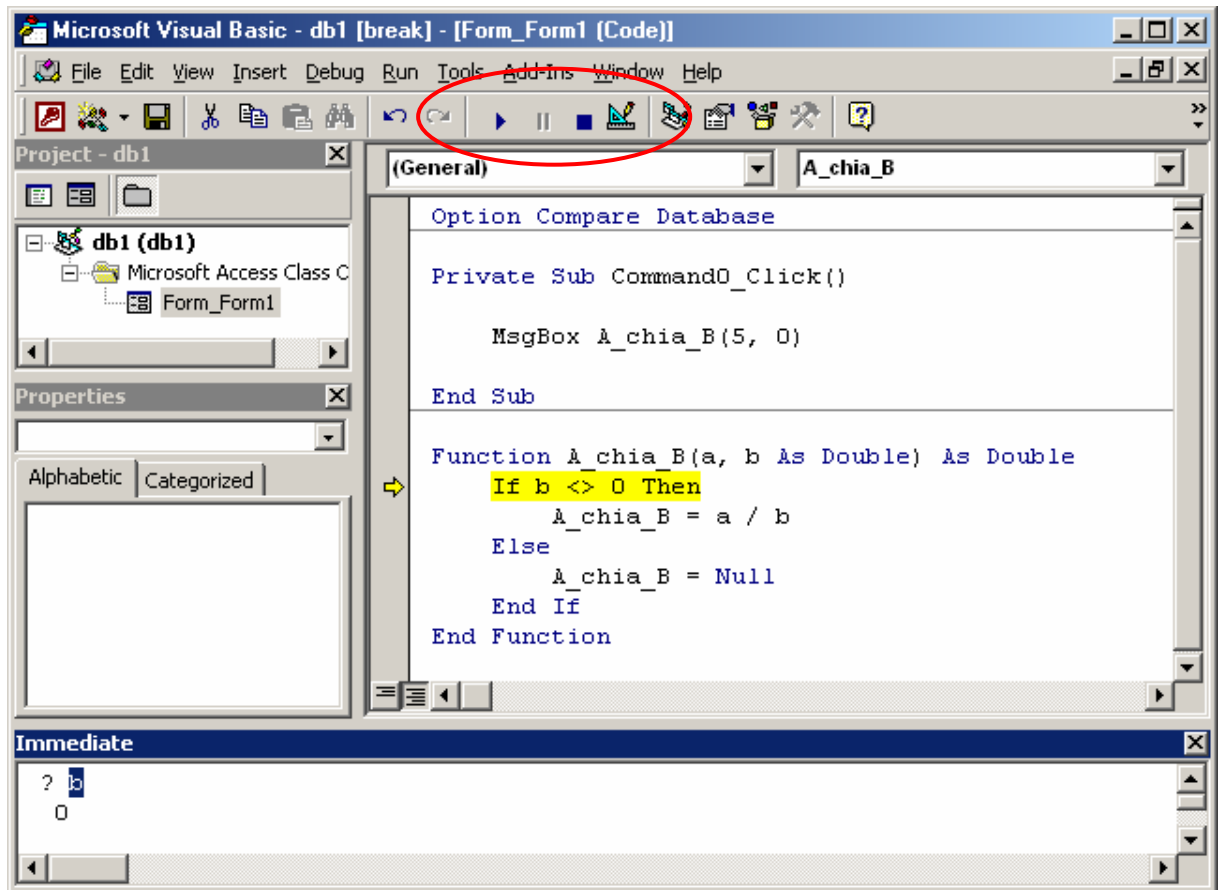
Private Sub Command0_Click()

    MsgBox A_chia_B(5, 0)

End Sub

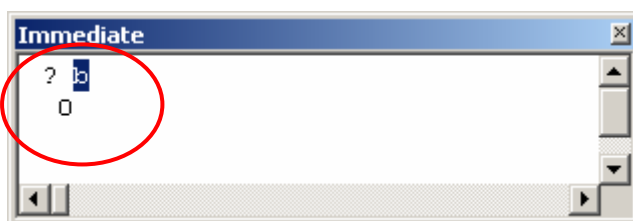
Function A_chia_B(a, b As Double) As Double
    If b <> 0 Then
        A_chia_B = a / b
    Else
        A_chia_B = Null
    End If
End Function
```

Việc thực thi sẽ được tiến hành tiếp tục tại vị trí vệt sáng đang trỏ. Bạn có thể dùng chuột để dịch chuyển vệt sáng về lệnh cần thực thi (chỉ trong cùng một chương trình con). Để thực thi tiếp nhấn phím F5 hoặc nút **Continue**  trên thanh công cụ; hoặc nhấn nút **Stop**  nếu muốn dừng việc sửa mã lệnh trong chế độ *Break*, chuyển về chế độ *Design*.



## Cửa sổ Immediate

Là công cụ hữu hiệu hỗ trợ việc dò lỗi bởi: hộp thoại này cho phép thực thi từng câu lệnh trên chế độ hội thoại.



Giả sử ví dụ trên sau khi gõ lệnh:

```
? b
```

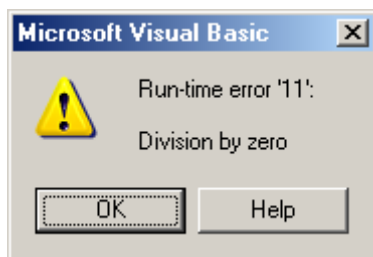
Xem giá trị của biến b. Sau khi nhấn Enter sẽ nhận được kết quả

```
0
```

Hoặc nếu gõ:

? a / b

Sẽ nhận được thông báo lỗi:



Vì  $b = 0$ .

## 5.2 Bẫy lỗi

Mục 5.1 đã trình bày những kỹ năng để xử lý lỗi khi đang soạn thảo chương trình. Các thao tác đó chỉ được thực hiện trong lúc đang xây dựng phần mềm (VBA IDE), do người lập trình xử lý. Khi phần mềm đã được đóng gói để chuyển đến người dùng nếu gặp lỗi, nó sẽ hiển thị một hộp thoại thông báo lỗi (*Error Dialog*) cho biết lý do vắn tắt về lỗi. Sau khi bạn nhấn OK, chương trình sẽ ngừng hoạt động, bị thoát.

Để xử lý lỗi trong tình huống này, có 2 phương pháp bẫy lỗi mà chúng tôi đưa ra dưới đây để tham khảo; hy vọng bạn sẽ chọn lựa được tình huống phù hợp để sử dụng một trong các phương pháp này đảm bảo chương trình viết ra chạy được đúng theo mục đích.

### Sử dụng lệnh **On Error Resume Next**

Khi đó từ chỗ đó trở đi, nếu chương trình gặp lỗi, nó sẽ bỏ qua (ignore) hoàn toàn. Điểm này tiện ở chỗ giúp chương trình EXE của ta tránh gặp lỗi thoát khỏi đột ngột như phân tích ở trên. Nhưng nó cũng bất lợi là khi khách hàng cho hay họ gặp những trường hợp lạ, không giải thích được (vì lỗi đã bị bỏ qua mà không ai để ý), thì ta cũng bí luôn, có thể không biết bắt đầu từ đâu để gỡ lỗi. Do đó, trong lúc gỡ lỗi ta không nên dùng nó, nhưng trước khi giao cho khách hàng bạn nên cân nhắc kỹ có nên sử dụng trong các đoạn mã lệnh hay không.

Ví dụ sử dụng **On Error Resume Next** để bỏ qua lỗi:

```

Function A_chia_B(a, b As Double) As Double
    On Error Resume Next

    A_chia_B = Null

    A_chia_B = a / b

End Function

```

Trong chương trình con trên, nếu  $b = 0$ , lệnh  $A\_chia\_B = a / b$  sẽ gặp phải lỗi. Do có lời khai báo **On Error Resume Next** nên lệnh lỗi này được bỏ qua (không thực hiện). Tức là giá trị hàm là Null.

### Sử dụng câu lệnh **On Error Goto <nhãn>**

Khi một thủ tục được đặt câu lệnh này, nếu gặp phải một lỗi nào đó, VBA sẽ chuyển thẳng việc thực hiện đến <nhãn> đã chỉ định. Thông thường các lệnh tiếp theo của <nhãn> là xử lý các tính huống lỗi.

Sau đây là ví dụ sử dụng phương pháp **On Error Goto <nhãn>** để bắt lỗi:

```

Function A_chia_B(a, b As Double) As Double
    On Error GoTo Loi

    A_chia_B = a / b
    MsgBox " Ok! "

Loi:

    If Err.Number = 11 Then
        MsgBox "Lỗi chia cho 0 !"
    End If

End Function

```

Trong chương trình con trên, trong trường hợp  $b = 0$  câu lệnh  $A\_chia\_B = a / b$  sẽ gây ra lỗi. Theo như khai báo **On Error Goto Loi** ban đầu, VBA sẽ bỏ qua tất cả các lệnh sau lệnh lỗi và chuyển thẳng tới các lệnh sau nhãn **Loi**: Ở đây là lệnh kiểm tra lỗi. Nếu  $Mã\ lỗi = 11 \rightarrow$  kết luận ngay một thông báo lỗi tiếng Việt. *Lỗi chia cho 0!*

Phương pháp này cũng được dùng phổ biến cả trong quá trình xây dựng để phát hiện lỗi, cũng như trong phần mềm đã đóng gói gửi đến khách hàng. Mỗi khi gặp lỗi sẽ được thông báo nguyên nhân gây ra lỗi bằng tiếng Việt (chẳng hạn) mà vẫn không ảnh hưởng đến hoạt động khác của phần mềm.



Trong phương pháp này, người lập trình nên khai thác tối đa đối tượng **Err** - đối tượng mang những thông tin về lỗi đang xảy ra, cụ thể:

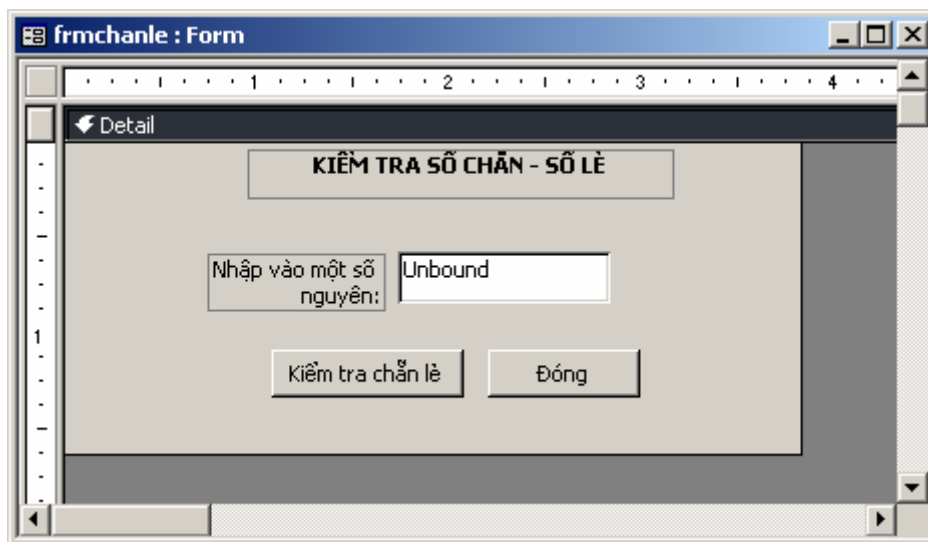
Hành động	Kết quả
Err.Description	Mô tả tên lỗi
Err.Number	Đưa ra mã lỗi
Err.Number	Xoá bỏ các giá trị của đối tượng Err

## 6. Một số ví dụ

Phần này trình bày một số ví dụ sử dụng Form, một số đối tượng điều khiển (Control), các khai báo, các cấu trúc lệnh và những kỹ thuật liên quan để giải quyết một số bài toán thực tế đơn giản.

**Bài toán 1:** Nhập vào một số nguyên và kiểm tra số đó là chẵn hay số lẻ?

Thiết kế form như sau:



Thiết lập thuộc tính cho các đối tượng như sau:

### Form

Caption: *Kiểm tra số chẵn - lẻ*  
 Default view: *Single Form*  
 Scroll bar: *Neither*  
 Record selector: *No*

Navigation Buttons: *No*

Diving line: *No*

### **Ô nhập số cần kiểm tra**

Name: *Text0*

### **Nút Kiểm tra chẵn lẻ**

Name: *cmdChanLe*

Caption: *Kiểm tra chẵn lẻ*

### **Nút Đóng**

Name: *cmdClose*

Caption: *Đóng*

Và mã lệnh cho form như sau:

```
'-----  
'Lệnh cho nút Kiểm tra chẵn lẻ  
'  
Private Sub cmdChanLe_Click()  
    If Text0 Mod 2 = 0 Then  
        MsgBox Text0 + " Là số chẵn !"  
    Else  
        MsgBox Text0 + " Là số lẻ !"  
    End If  
End Sub  
  
'-----  
'Lệnh cho nút Đóng  
'  
Private Sub cmdClose_Click()  
    DoCmd.Close  
End Sub
```

**Bài toán 2:** Nhập vào 2 số nguyên và tính USC và BCS của 2 số đó

Thiết kế form như sau:

Thiết lập thuộc tính cho các đối tượng như sau:

### Form

Caption: *Tim USC và BSC*  
 Default view: *Single Form*  
 Scroll bar: *Neither*  
 Record selector: *No*  
 Navigation Buttons: *No*  
 Diving line: *No*

### Ô nhập số cần kiểm tra

Ô	Thuộc tính Name
A:	txtA
B:	txtB
USC:	txtUSC
BSC:	txtBSC

### Nút Tính toán

Name: *cmdTinhToan*  
 Caption: *Tính toán*

### Nút Đóng

Name: *cmdClose*  
 Caption: *Đóng*

Và mã lệnh cho form như sau:

```

'-----
'Hàm xác định USCNN của 2 số nguyên (thuật toán O-cơ-lít)
'
Function usc(a, b As Integer) As Integer
    Dim a1, b1 As Integer

    a1 = a
    b1 = b
    While a1 <> b1
        If a1 > b1 Then
            a1 = a1 - b1
        Else
            b1 = b1 - a1
        End If
    Wend
    usc = a1

End Function

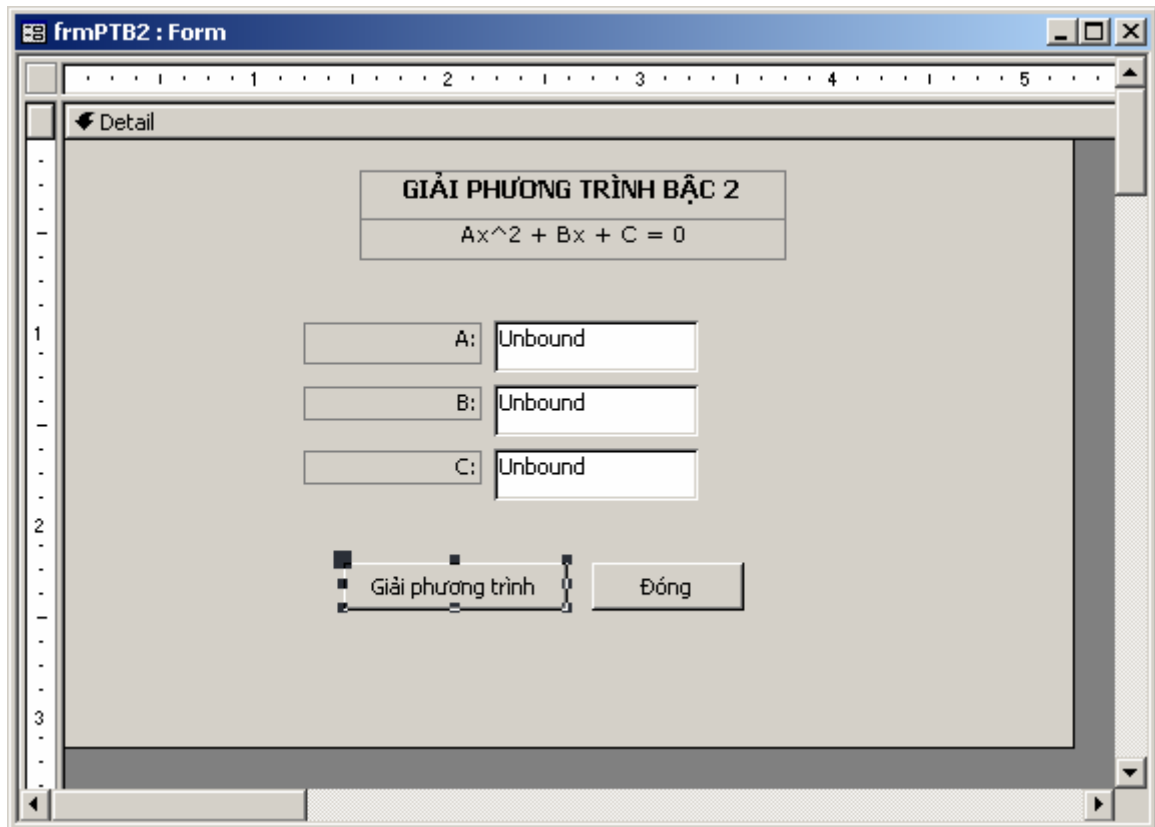
'-----
'Mã lệnh cho nút Tính toán
'
Private Sub cmdTinhToan_Click()
    txtUSC = usc(txtA, txtB)
    txtBSC = txtA * txtB / usc(txtA, txtB)
End Sub

'-----
'Mã lệnh cho nút Đóng
'
Private Sub cmdClose_Click()
    DoCmd.Close
End Sub

```

**Bài toán 3:** Nhập vào 3 hệ số A, B, C của phương trình bậc hai  $Ax^2 + Bx + C = 0$  và cho biết nghiệm phương trình đó:

Thiết kế form như sau:



Thiết lập thuộc tính cho các đối tượng như sau:

### Form

Caption: *Giải phương trình bậc 2*  
 Default view: *Single Form*  
 Scroll bar: *Neither*  
 Record selector: *No*  
 Navigation Buttons: *No*  
 Diving line: *No*

### Ô nhập số cần kiểm tra

Ô	Thuộc tính Name
A:	txtA
B:	txtB
C:	txtC

### Nút Tính toán

Name: *cmdGPTB2*  
 Caption: *Giải phương trình*

### Nút Đóng

Name: *cmdClose*  
 Caption: *Đóng*

Và mã lệnh cho form như sau:

```
'-----  
'Mã lệnh cho nút Giải phương trình  
'  
Private Sub cmdChanLe_Click()  
    Dim delta, x, x1, x2 As Double  
    Dim kqua As String  
  
    delta = txtB * txtB - 4 * txtA * txtC  
    If delta = 0 Then  
        x = -txtB / (2 * txtA)  
        kq = "Nghiem kép: x1 = x2 = " + Trim(Str(x))  
    Else  
        If dleta > 0 Then  
            x1 = (-txtB + Sqr(delta)) / (2 * txtA)  
            x1 = (-txtB - Sqr(delta)) / (2 * txtA)  
            kqua = "Có 2 nghiệm phân biệt:" + Chr(13) _  
                + " X1 = " + Trim(Str(x1)) + Chr(13) _  
                + " X2 = " + Trim(Str(x2))  
        Else  
            kqua = "Phương trình vô nghiệm"  
        End If  
    End If  
  
    MsgBox kqua  
  
End Sub  
  
'-----  
'Lệnh cho nút Đóng  
'  
Private Sub cmdClose_Click()  
    DoCmd.Close  
End Sub
```

# CHƯƠNG 7

## LẬP TRÌNH CƠ SỞ DỮ LIỆU

Access không những là một hệ quản trị CSDL mạnh mà còn cung cấp những công cụ mạnh cho phép phát triển một CSDL đơn thuần thành một sản phẩm đóng gói thương mại. Chúng ta đã được học Queries, Forms, Report – đó là những công cụ khá mạnh và dễ học dùng để xây dựng các truy vấn, biểu mẫu và báo cáo in ấn trong Access. Chương 7 đã được làm quen với ngôn ngữ VBA- một ngôn ngữ lập trình trực quan, hướng đối tượng trong các ứng dụng Microsoft Office nói chung cũng như trên Access. Đó là cơ sở quan trọng để trong chương này chúng ta tìm hiểu những kỹ thuật lập trình CSDL bằng VBA- có thể nói là một mức chuyên sâu tiếp theo những gì đã học VBA căn bản.

Các chủ đề sẽ được bàn đến trong chương này bao gồm:

- Lớp đối tượng truy cập dữ liệu (DAO- Data Access Objects);
- Bài toán tìm kiếm;
- Bài toán đặt lọc;
- Một số bài toán khác.

## 1. Kỹ thuật DAO

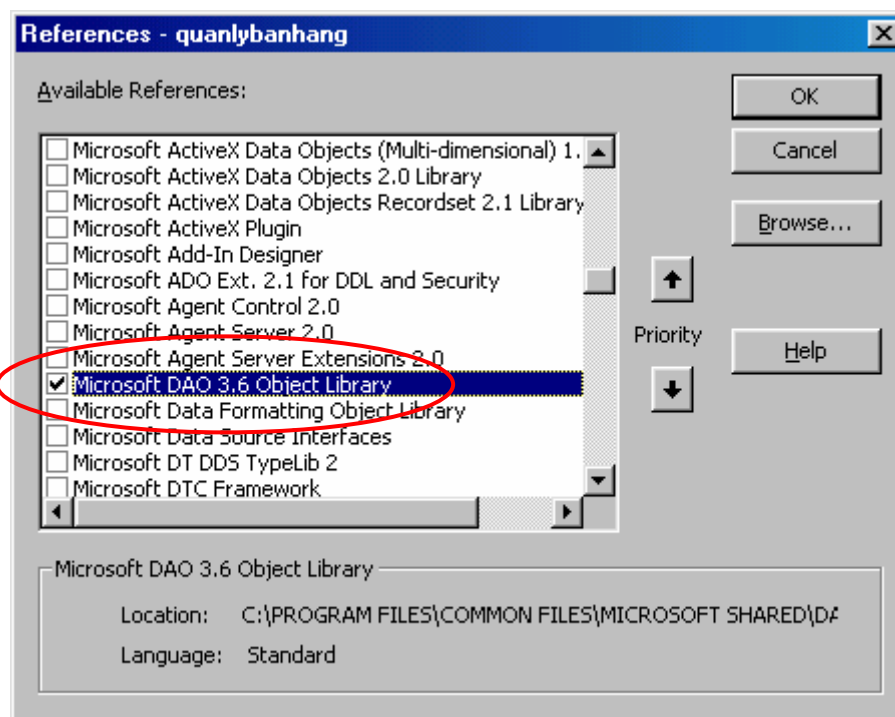
DAO (Data Access Objects – Các đối tượng truy xuất dữ liệu) là tập hợp bao gồm lớp các đối tượng có thể dùng để lập trình truy cập và xử lý dữ liệu trong các hệ CSDL. Ở đây CSDL Access, ngôn ngữ lập trình VBA.

DAO được phát triển khá sớm, gần đây nhất là phiên bản DAO 3.5 và 3.51- nó có thể thực hiện tốt được trên các phiên bản Access từ 97 trở về trước. Với Access 2000, XP phải dùng phiên bản DAO 3.6. Với phiên bản mới này, DAO 3.6 sử dụng nền Microsoft Jet 4.0. Vì vậy, có thể làm việc được trên nền Unicode dễ dàng.

Để nạp thư viện DAO3.6 vào làm việc, hãy thực hiện như sau:

**Bước 1:** Mở cửa sổ lập trình VBA;

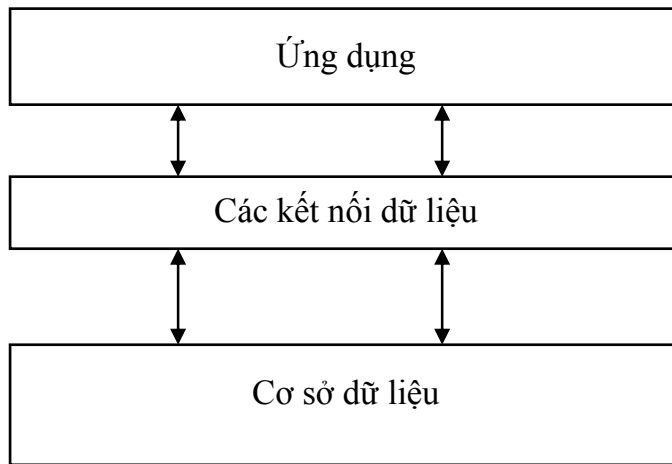
**Bước 2:** Chọn thực đơn **Tools | References ..** Hộp thoại sau xuất hiện:



Hãy chọn (tích) mục **Microsoft DAO 3.6 Object Library** trên danh sách *Available References*; chọn xong, nhấn **OK** để đóng lại.



Trước khi bước vào học lập trình CSDL, các bạn hãy xem cách thức làm việc như thế nào?



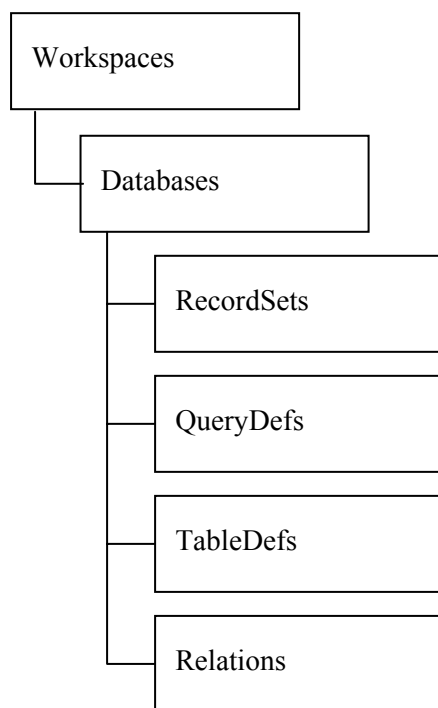
Trong đó:

- Tầng ứng dụng: bao gồm những giao diện người sử dụng cũng như những công cụ đơn giản mà người lập trình có thể dùng để xử lý dữ liệu theo các bài toán;
- Tầng Kết nối dữ liệu: bao gồm tập hợp các công cụ, phương thức để kết nối tới những dữ liệu cần làm việc trong CSDL. Ở đây, tầng kết nối bao gồm các chuẩn Microsoft Jet 4.0 và các lớp đối tượng DAO;
- Tầng Cơ sở dữ liệu: bao gồm các bảng, các query trong cơ sở dữ liệu thực tại.

Như vậy để lập trình trên một CSDL phải sử dụng các đối tượng, các phương thức ở tầng kết nối như là những công cụ để có thể truy cập được vào CSDL tác nghiệp xử lý. Tầng kết nối đó chính là Jet 4.0 và DAO 3.6 mà chúng ta sẽ được tìm hiểu dưới đây.

## 1.1 Lớp đối tượng DAO

Cấu trúc một CSDL bao gồm nhiều thành phần, đòi hỏi lập trình cũng cần có những thành phần tương ứng để làm việc. Lớp các thành phần tương ứng để có thể lập trình được trên toàn bộ cấu trúc CSDL là lớp các đối tượng DAO. Chúng có tên gọi, có những tập thuộc tính, các phương thức làm việc và có quan hệ mật thiết với nhau. Cây phân cấp lớp các đối tượng DAO sau đây thể hiện điều đó:

**Trong đó:**

- **Workspaces** – định nghĩa tập hợp các vùng làm việc. Đây có thể coi là lớp làm việc cao nhất. Về lý thuyết có thể khai báo một vài vùng làm việc (Workspace), nhưng trên thực tế chỉ cần khai báo một vùng làm việc và vùng này luôn được khai báo ngầm định cho CSDL hiện tại. Nên sẽ không cần bàn nhiều đến lớp các WorkSpace này;
- **Databases** - định nghĩa tập hợp các CSDL Access cần làm việc trên một dự án;
- **RecordSets**- định nghĩa các tập hợp bản ghi (Records) cần làm việc;
- **QueryDefs** - định nghĩa tập hợp các Query để làm việc. Querydefs và Recordsets là khả năng truy xuất, xử lý dữ liệu (Data Manipulation) của DAO;
- **TableDefs** - định nghĩa tập hợp các bảng (Table) cần làm việc. Đây là khả năng định nghĩa dữ liệu (Data-Definition Language);
- **Relations** - định nghĩa tập hợp các quan hệ (Relationship) cần làm việc;

Mỗi lớp các đối tượng trên sẽ bao gồm tất cả các đối tượng đối tượng cùng loại trong một đối tượng mẹ đang mở. Ví dụ:

- Databases sẽ bao gồm tất cả các CSDL đang được mở trong vùng làm việc hiện tại;

- RecordSets sẽ bao gồm tập hợp tất cả các Recordset đang được mở trên CSDL hiện tại.

Khi đó, để tham chiếu đến một đối tượng cụ thể cần làm việc, có thể dùng chỉ số (số thứ tự của đối tượng đó trên tập hợp tất cả các đối tượng đó) hoặc dùng tên gọi đối tượng đó để tham chiếu. Ví dụ sau liệt kê tên của tất cả các Recordset đang sử dụng trong CSDL db.

```
Dim db As DAO.Database
'-----
'các câu lệnh tiếp theo ở đây..
'
For i = 0 To db.Recordsets.Count
    MsgBox db.Recordsets(i).Name
Next
```

Để làm việc tới một đối tượng cụ thể, cần phải tham chiếu từ lớp các đối tượng mẹ của nó.

Ví dụ: Để hiển thị giá trị của trường (Field) hoten trên tập hợp các bản ghi (Recordset) rs1 làm như sau:

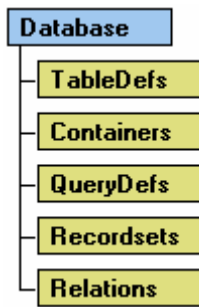
```
MsgBox rs1.Fields("hoten").Value

' hoặc
MsgBox rs1.Fields![hoten].Value
```

## 1.2 Đối tượng Database

Database là đối tượng dùng làm việc với một CSDL (trong trường hợp này có thể hiểu một CSDL như một tệp Access .MDB).

Lớp các đối tượng con của Database được thể hiện qua sơ đồ sau:



## Khai báo

```

Dim db As DAO.Database

' Gán db cho một CSDL cụ thể
Set db = OpenDatabase("C:\Baitap\qlbh.mdb")

'Đặc biệt, lệnh gán db cho CSDL hiện tại như sau:
Set db = CurrentDb
  
```

Khi không làm việc với CSDL nào đó, có thể ra lệnh đóng để giải phóng bộ nhớ bằng cách:

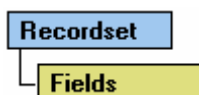
```
db.Close
```

Sau khi lệnh này thực thi, tất cả các đối tượng con của db nếu đang mở sẽ được đóng lại để giải phóng bộ nhớ. Bản thân db cũng được giải phóng bộ nhớ (bằng Nothing), tất nhiên tệp CSDL và dữ liệu vẫn còn nguyên trên đĩa.

### 1.3 Đối tượng RecordSet

Recordset là đối tượng dùng để miêu tả tập hợp các bản ghi của một bảng, của một query hoặc tập các bản ghi kết quả của việc thi hành một câu lệnh SQL nào đó.

Lớp các đối tượng con của Recordset được thể hiện qua sơ đồ sau:



## Khai báo

```
Set rs=db.OpenRecordset (<Name>)
```

### Trong đó:

- **Set rs = db.OpenRecordset** là lệnh để tạo ra tập hợp các bản ghi từ CSDL db gán vào biến kiểu recordset *rs*;
- *<Name>* là một chuỗi ký tự chỉ ra nguồn dữ liệu sẽ trả về cho Recordset. Chuỗi này có thể là tên một bảng, một Query hoặc một câu lệnh SQL;

Mỗi biến Recordset khi làm việc, phải được chỉ ra Database xuất xứ của nó (phải được tham chiếu từ một biến kiểu Database đã được khai báo). Sau đây là các ví dụ:

### Ví dụ 1:

Gán tập hợp các bản ghi từ một bảng vào biến Recordset (ở đây là bảng *canbo*).

```
Dim rs As DAO.Recordset  
  
Set rs = db.OpenRecordset ("canbo")
```

### Ví dụ 2:

Gán tập hợp các bản ghi từ một câu lệnh chọn dữ liệu SQL vào biến Recordset (ở đây là các thông tin *hoten, ngaysinh* của tất cả các cán bộ *nữ* từ bảng *canbo*).

```
Dim rs As DAO.Recordset  
  
Set rs = db.OpenRecordset ("SELECT hoten, ngaysinh FROM canbo  
WHERE gioitinh = False")
```

## Một số thuộc tính của Recordset

### Thuộc tính **Name**

Trả về chuỗi ký tự trong tham số *<name>* của lệnh gọi Recordset. Ví dụ: lệnh sau sẽ cho biết chuỗi ký tự tạo nguồn dữ liệu cho Recordset là gì?

```
MsgBox rs.Name
```

### Thuộc tính **AbsolutePosition**

Cho biết vị trí bản ghi hiện tại (được tính từ 0). Trong trường hợp không có bản ghi nào trên recordset hoặc con trỏ bản ghi đang nằm ở EOF- sẽ không thể lấy được giá trị thuộc tính này. Do vậy để sử dụng thuộc tính này thường phải đi kèm thuộc tính kiểm tra có tồn tại bản ghi nào hay không (`RecordCount > 0`) và con trỏ bản ghi có ở cuối tệp chưa (`EOF = False`).

### Thuộc tính **RecordCount**

Cho biết tổng số bản ghi trả về trên Recordset

### Thuộc tính **EOF**

Cho biết con trỏ bản ghi hiện tại có nằm ở EOF hay không? Nếu có giá trị thuộc tính này là True, trái lại là False.

### Thuộc tính **Fields**

Dùng tham chiếu tới các trường (Fields) trên tập hợp các bản ghi mà Recordset trả về. Thực tế Field cũng là một đối tượng và cũng có bộ thuộc tính và các phương thức của nó. Với Field của Recordset thông thường người ta hay sử dụng thuộc tính *Value*. Nếu không chỉ định thuộc tính cụ thể nào cho Field, VBA vẫn hiểu ngầm định đó là Value.

Ví dụ: Hiện thị giá trị trường *hoten* trong Recordset *rs*

```
Msgbox rs.Fields("hoten").Value  
  
'hoặc  
Msgbox rs.Fields("hoten")
```

## Một số phương thức của Recordset

### Phương thức **Close**

Để đóng Recordset, giải phóng bộ nhớ. Chỉ thực hiện hành động này khi không làm việc với Recordset nào đó.

*Các phương thức di chuyển bản ghi của Recordset*

### Phương thức **MoveFirts**

Để chuyển con trỏ về bản ghi đầu tiên

### Phương thức **MoveLast**

Để di chuyển con trỏ về bản ghi cuối cùng

### Phương thức **MoveNext**

Dịch đến bản ghi kế sau

### Phương thức **MovePrevious**

Dịch đến bản ghi kế trước

### Ví dụ 3:

Ví dụ sau duyệt và hiển thị toàn bộ *Hoten* của bảng *canbo*

```
Dim db As DAO.Database
Dim rs As DAO.Recordset

Set db = CurrentDb
Set rs = db.OpenRecordset ("canbo")

If rs.RecordCount > 0 Then
    rs.MoveFirst
    While rs.EOF = False
        MsgBox rs.Fields ("hoten").Value
        rs.MoveNext
    Wend
End If
```

### Phương thức **AddNew, Update**

Để thêm mới một bản ghi vào Recordset. Qui trình thêm một bản ghi mới như sau:

1. Ra lệnh Addnew
2. Gán giá trị cho các trường của bản ghi mới
3. Ra lệnh Update

Dưới đây là ví dụ thêm mới một hồ sơ cán bộ mới vào bảng *canbo*.

### Ví dụ 4:

```
Dim db As DAO.Database
Dim rs As DAO.Recordset
```

```

Set db = CurrentDb
Set rs = db.OpenRecordset ("canbo")

'-----
'Ra lệnh thêm mới bản ghi
'
rs.AddNew

'-----
'Gán giá trị cho các trường của bản ghi mới
'
rs.Fields("canboID") = "CB00565"
rs.Fields("hoten") = "Nguyễn Sơn Hải"
rs.Fields("ngaysinh") = #2/11/1975#
rs.Fields("gioitinh") = True
rs.Fields("chucvuID") = "CV002"

'-----
'Ra lệnh ghi lại dữ liệu
'
rs.Update

```

### Phương thức **Edit, Update**

Phương thức Edit để sửa dữ liệu một bản ghi nào đó trên recordset. Qui trình để sửa một bản ghi như sau:

1. Định vị tới bản ghi cần sửa trên recordset
2. Ra lệnh Edit
3. Gán giá trị mới cho các trường cần sửa
4. Ra lệnh Update

Dưới đây là ví dụ về sửa hồ sơ cán bộ có mã *CB000565*

#### **Ví dụ 5:**

```

Dim db As DAO.Database
Dim rs As DAO.Recordset

Set db = CurrentDb

'-----
'Định vị tới bản ghi cần sửa
'
Set rs = db.OpenRecordset("SELECT * FROM canbo WHERE
canboID='CB000565'")

If rs.RecordCount > 0 Then

```



```

rs.MoveFirst

'-----
'Ra lệnh sửa bản ghi
'
rs.Edit

'-----
'Thực hiện sửa các trường dữ liệu cần thiết
'
rs.Fields("hoten") = "Nguyễn Văn Hải"
rs.Fields("ngaysinh") = #22/11/1975#

'-----
'Ra lệnh ghi lại dữ liệu vừa sửa
'
rs.Update
End If

```

### Phương thức **Delete**

Để xoá bản ghi hiện tại ra khỏi Recordset. Khi đó bản ghi hiện tại sẽ bị xoá bỏ khỏi CSDL. Cần thận trọng mỗi khi ra lệnh này. Thông thường các lệnh một nút Xoá bản ghi của một mẫu nhập liệu (nhập vào biến Recordset *rs*) như sau:

#### **Ví dụ 6:**

```

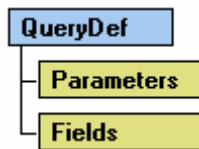
Private Sub cmDelete_Click()
    Dim tbao

    tbao = MsgBox("Đã chắc chắn xoá chưa?", vbYesNo + vbCritical)
    If tbao = vbYes Then
        rs.Delete
        rs.MoveNext
    End If
End Sub

```

## **1.4 Đối tượng QueryDef**

Đối tượng Querydef dùng để tham chiếu tới các Query có sẵn (Buil-in) trên CSDL Access, hoặc cũng có thể lập trình tạo các Query từ các câu lệnh SQL.



Để tạo và kích hoạt một query trên VBA bằng cách thực thi câu lệnh SQL bạn làm theo hướng dẫn sau:

```

'-----
' Khai báo một biến kiểu Database và một biến kiểu QueryDef

Dim db As DAO.Database
Dim qr As DAO.QueryDef

'-----
' Ra lệnh tạo một Query mới, có tên rỗng (chỉ ở trong bộ nhớ)

Set qr = db.CreateQueryDef (<tên query>)

'-----
' Gán chuỗi lệnh SQL vào thuộc tính SQL của query
'
qr.SQL = "Gõ lệnh SQL cần thi hành vào đây"

'-----
' Ra lệnh thi hành query
'
qr.Execute

'-----
' giải phóng bộ nhớ

qr.Close
  
```

#### Trong đó:

- Bắt buộc phải khai báo một biến kiểu QueryDef để làm việc (biến *qr*);
- Phải có một biến Database đã được khai báo sẵn (biến *db*);
- Lệnh **Set qr = db.CreatQueryDef(<tên query>)** để tạo một query mới lên CSDL. <tên query> sẽ được hiển thị trên danh sách trong thẻ Queries trên cửa sổ Database. Nếu <tên query>="", query này sẽ chỉ tồn tại trong bộ nhớ. Tùy thuộc vào mục đích công việc mà có đặt tên query hay không, nếu chỉ đơn thuần tạo một query để xử lý công việc nào đó rồi giải phóng, nên đặt <tên query>="";

- Lệnh **qr.SQL=<câu lệnh SQL>** để gán lệnh SQL cần thực thi vào Query. Tùy thuộc vào câu lệnh SQL này mà query sẽ thực hiện những gì. Ví dụ: **qr.SQL = "DELETE \* FROM canbo"** lệnh này sẽ xoá tất cả các bản ghi trên bảng cán bộ;
- Lệnh **qr.Execute** để thi hành câu lệnh SQL đã được thiết lập. Lệnh này tương đương nhấn nút Run đối với một query trên chế độ thiết kế;
- Lệnh **qr.Close** để đóng query hiện tại và giải phóng bộ nhớ khi không cần làm việc đến nữa.

Sau đây là một số ví dụ về sử dụng biến QueryDef để tạo ra một số loại query xử lý dữ liệu trong môi trường VBA.

**Ví dụ 1:** Tạo DELETE query để xoá danh sách những cán bộ có tuổi lớn hơn 60 ra khỏi bảng *canbo* (cán bộ đã nghỉ hưu)

```
Dim db As DAO.Database
Dim qr As DAO.QueryDef

Set db = CurrentDb

Set qr = db.CreateQueryDef("")

qr.SQL="DELETE * FROM canbo WHERE Year(Date()) - " _
    & " Year(Ngaysinh)>=60"

qr.Execute

qr.Close
```

**Ví dụ 2:** Giả sử đã thêm một trường mới trên bảng cán bộ có tên *luongchinh*. Tạo UPDATE query để tính giá trị cho trường này = **hesoluong \* 290000**.

```
Dim db As DAO.Database
Dim qr As DAO.QueryDef

Set db = CurrentDb

Set qr = db.CreateQueryDef("")

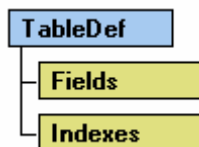
qr.SQL = "UPDATE canbo SET canbo.luongchinh = hesoluong * 290000"

qr.Execute

qr.Close
```

## 1.5 Đối tượng TableDef

Đối tượng TableDef được dùng để tham chiếu tới các bảng dữ liệu (Table) trên CSDL. Thông qua đối tượng này có thể thiết kế, chỉnh sửa được cấu trúc các bảng dữ liệu trong chế độ Run-time của VBA như trên chế độ thiết kế bảng Design View.



### Một số thuộc tính quan trọng của TableDef

Thuộc tính **Name**

Cho biết tên bảng được gán vào biến kiểu TableDef

Thuộc tính **RecordCount**

Cho biết tổng số bản ghi hiện có trên bảng được gán bởi biến TableDef

Thuộc tính **DateCreated**

Cho biết thời gian tạo ra bảng được gán vào biến kiểu TableDef

Thuộc tính **Fields**

Để tham chiếu tới các trường của bảng. Đây là thuộc tính hay được sử dụng nhất đối với TableDef. Thực chất, Field ở đây là một đối tượng, do đó cũng có tập các thuộc tính và phương thức riêng cho thuộc tính này.

Dưới đây là thủ tục hiển thị tên của tất cả các trường trong một bảng nào đó (ngầm định trên một CSDL đã được khai báo và gán biến *db* - kiểu Database).

#### Ví dụ 1:

```
Sub LietKeTenTruong(tenbang As String)
    Dim tbl As DAO.TableDef
```

```

Set tbl = db.TableDefs (tenbang)

For i = 0 To tbl.Fields.Count - 1
    MsgBox tbl.Fields(i).Name
Next

End Sub

```

## Một số phương thức của TableDef

### Phương thức **CreateTableDef**

Để tạo ra một bảng mới bằng VBA. Cú pháp tạo bảng mới như sau:

```

Set tbl = db.CreateTableDef (<Tên bảng mới>)

'-----
'...Các thủ tục tạo trường mới cho bảng
'-----

db.TableDefs.Append tbl

```

#### Trong đó:

- *db* – là biến kiểu Database đã được gán bởi CSDL cần làm việc (bảng mới sẽ được tạo ra trên CSDL này);
- <Tên bảng mới> là tên bảng cần tạo.
- Lệnh **db.TableDefs.Append tbl** là lệnh ghi cấu trúc bảng đang khai báo lên CSDL đã chỉ định.

### Phương thức **CreateField**

Để tạo ra các trường cho một bảng kiểu TableDef nào đó. Để thêm một trường mới lên bảng, sử dụng cú pháp sau:

```
tbl.Fields.Append tbl.CreateField (<tên trường>, <KiểuDL>, <độ lớn>)
```

#### Trong đó:

- <tên trường> - tên trường mới cần tạo;
- <KiểuDL> - là một tùy chọn để khai báo kiểu dữ liệu của trường cần tạo. Kiểu dữ liệu được khai báo theo các hằng số như sau:

**Giá trị: Tương ứng với kiểu**

*dbBoolean* Boolean

*dbByte* Byte

*dbChar* Char

*dbCurrency* Currency

*dbDate* Date/Time

*dbDecimal* Decimal

*dbDouble* Double

*dbFloat* Float

*dbGUID* GUID

*dbInteger* Integer

*dbLong* Long

*dbMemo* Memo

*dbNumeric* Numeric

*dbSingle* Single

*dbText* Text

*dbTime* Time

- <Độ lớn> là một tùy chọn để khai báo độ lớn dữ liệu nếu cần.

Tiếp theo là ví dụ minh họa cách tạo cấu trúc một bảng dữ liệu tổng hợp những hướng dẫn đã trình bày trên.

**Ví dụ 2:**

```
Sub TaoBangMoi()  
On Error GoTo Loi  
    Dim tbl As DAO.TableDef  
  
    Set tbl = db.CreateTableDef("NewTable")  
  
    tbl.Fields.Append tbl.CreateField("ID", dbInteger)  
    tbl.Fields.Append tbl.CreateField("Name", dbText)  
    tbl.Fields.Append tbl.CreateField("Age", dbByte)  
    tbl.Fields.Append tbl.CreateField("DateBirth", dbDate)  
    tbl.Fields.Append tbl.CreateField("Comment", dbMemo)  
  
    db.TableDefs.Append tbl  
  
Exit Sub
```

```

Loi:
    If Err.Number = 3010 Then
        MsgBox "Đã tồn tại bảng có tên " + tbl.Name
    End If
End Sub

```

## 1.6 Đối tượng Relation

Đối tượng Relation dùng để tạo kết nối (Relationship) giữa 2 bảng trong CSDL Access. Dưới đây là một ví dụ tạo kết nối giữa 2 bảng hoadon và khách trong CSDL Quản lý bán hàng.

```

Sub CreatRelationship()
On Error GoTo Loi
    Dim db As DAO.Database
    Dim rls As DAO.Relation

    Set db = CurrentDb

    Set rls = db.CreateRelation("TaoQuanHe", "khach", "hoadon",
dbRelationUpdateCascade)

    rls.Fields.Append rls.CreateField("khachID")
rls.Fields("khachID").ForeignName = "khachID"
db.Relations.Append rls

Loi:
    If Err.Number = 3012 Then
        MsgBox "Đã tồn tại quan hệ này !"
    End If

End Sub

```

Trong trường hợp đã tồn tại kết nối này, một thông báo lỗi tiếng Việt "*Đã tồn tại quan hệ này !*" xuất hiện.

## 2. Bài toán đặt lọc dữ liệu

Đặt lọc là lớp bài toán phổ dụng trong thực tế. Với bài toán này phải có những yêu cầu cụ thể về lọc dữ liệu (điều kiện lọc). Kết quả trả về sẽ là một tập hợp các bản ghi, có thể được kết xuất trên form hoặc được in ra máy in dưới dạng report.

Dưới đây là minh hoạ một bài toán đặt lọc sau khi đã được xây dựng và đang hoạt động:

Chọn khách cần xem: Nguyễn Đức Ngộ

**Các hoá đơn đã mua:**

	hoadonID:	ngayban:	tongtien:
▶	1	12/10/2004	893
	2	24/10/2004	114

Record: 1 of 2

Qui trình xây dựng một bài toán đặt lọc có thể thực hiện như sau:

**Bước 1:** Xây dựng form con - form sẽ chứa những kết quả lọc được;

**Bước 2:** Xây dựng form mẹ - form chứa những thiết lập điều kiện để lọc;

**Bước 3:** Thực hiện lọc ra các bản ghi thoả mãn các điều kiện trên form mẹ và hiển thị kết quả lên form con.

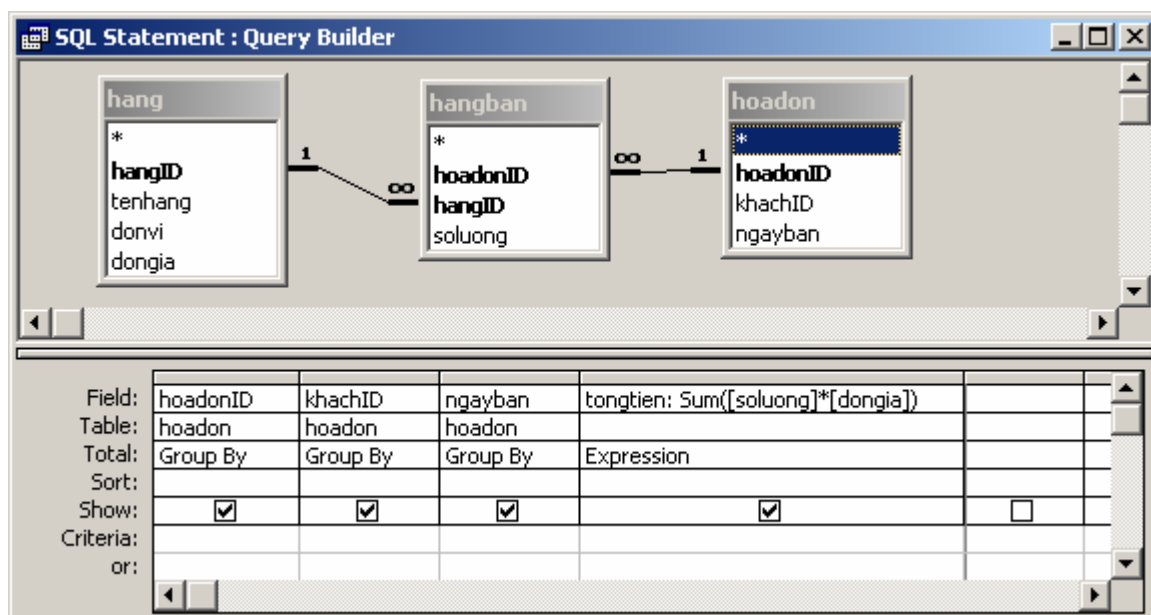
Với bài toán trên, cách giải quyết như sau:

**Bước 1: Xây dựng Form con**

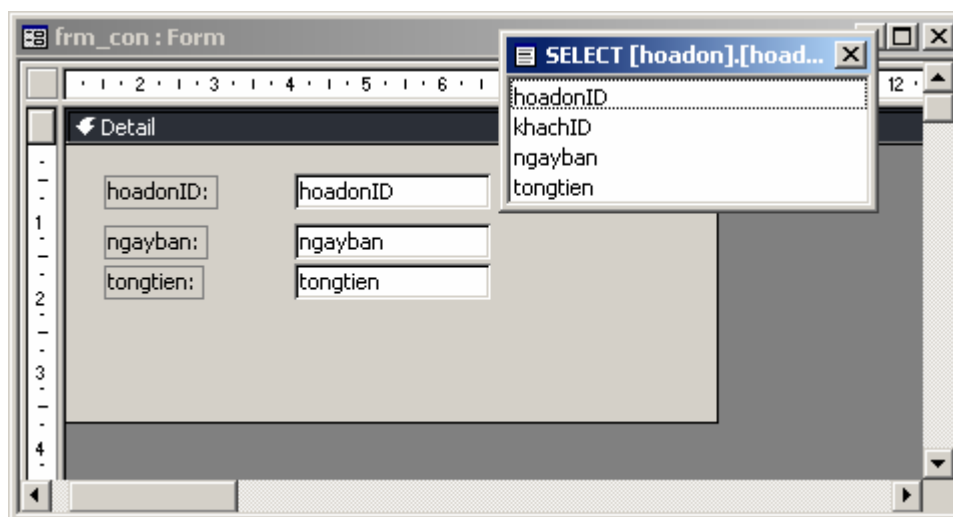
Sử dụng các kỹ năng thông thường để tạo một form con đáp ứng được các kết quả theo như bài toán. Cụ thể từng bước như sau:

- Tạo mới form ở chế độ Design view;
- Thiết lập thuộc tính Record Source cho form là một Total Query như sau:





- Thiết lập thuộc tính Default View cho form con là *Datasheets*;
- Mở cửa sổ FieldList và thiết kế form như sau:



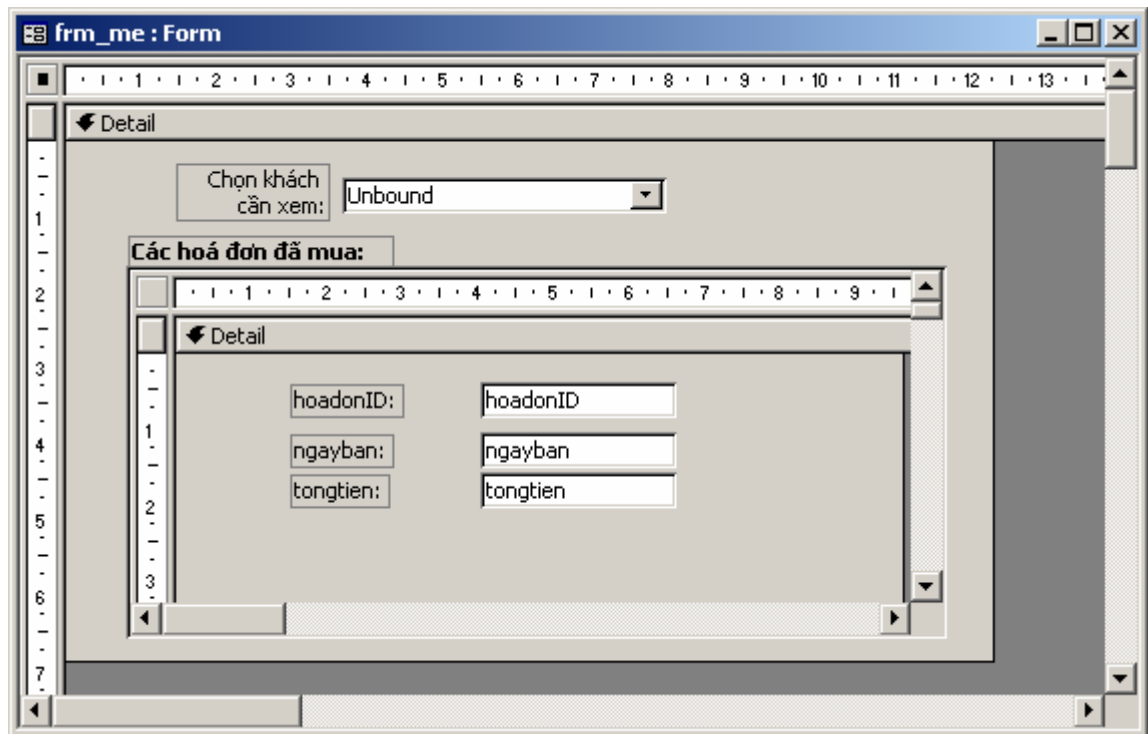
- Đóng thiết kế form con lại và ghi tên là *frm\_formcon*.

## Bước 2: Xây dựng form mẹ

- Tạo mới form ở chế độ Design view;
- Đưa hộp Combobox từ thành công cụ Toolbox lên form mẹ (giả sử tên (Name) của Combo này là *Combo0*). Sử dụng tính năng Combobox Wizard của Access để làm việc này. Mục đích là đưa danh sách các khách hàng từ bảng khách vào hộp Combo để chọn, phục vụ nhu cầu lọc dữ liệu;
- Sử dụng đối tượng Sub-form/Sub-report trên thanh công cụ Toolbox để đưa form con vừa tạo lên form mẹ. Ngầm định tên của sub-form này là

frm\_formcon. Chú ý đến tên gọi này vì phần tiếp sau sẽ sử dụng nó để lập trình.

Cuối cùng, giao diện thiết kế form mẹ như sau:



### Bước 3: Thiết lập lệnh lọc dữ liệu trên form mẹ

Công việc của bước này là làm sao để sau khi chọn tên một khách hàng ở hộp Combobox, danh sách các hoá đơn mua hàng của khách đó sẽ được hiển thị lên form con. Muốn thế, việc lập trình lọc dữ liệu ở đây phải được thực hiện trong thủ tục đáp ứng sự kiện **Combo0\_Click**. Giải thuật sẽ là:

- Tạo một biến Recordset để thi hành câu lệnh SQL đưa ra danh sách kết quả thoả mãn điều kiện đặt lọc. Câu lệnh SQL trong trường hợp này là:

```
"SELECT hoadonID, kháchID, ngayban, Sum([soluong]*[dongia]) " _
+ " AS tongtien FROM " _
+ " hoadon INNER JOIN (hang INNER JOIN hangban ON " _
+ " hang.hangID = hangban.hangID) ON hoadon.hoadonID =" _
+ " hangban.hoadonID WHERE Trim(kháchID)='"+Trim(Combo0)'" _
+ " GROUP BY hoadonID, kháchID, ngayban "
```

- Gán thuộc tính Recordset của form con là biến kiểu recordset vừa tạo ra (chứa kết quả đã lọc);
- Ra lệnh làm tươi dữ liệu cho form con.

Tất nhiên, trước đó phải khai báo một biến kiểu Database toàn cục trong form và định nghĩa nó ở thủ tục Form\_Load()

Toàn bộ mã lệnh cho bài toán trên như sau:

```
Dim db As DAO.Database
```

```
Private Sub Form_Load()
```

```
    Set db = CurrentDb
```

```
End Sub
```

```
Private Sub Combo0_Click()
```

```
    Dim rs As DAO.Recordset
```

```
    Set rs = db.OpenRecordset("SELECT hoadonID, khachID, " _
    + " ngayban, Sum([soluong]*[dongia]) AS tongtien FROM" _
    + " hoadon INNER JOIN (hang INNER JOIN hangban ON " _
    + " hang.hangID = hangban.hangID) ON hoadon.hoadonID =" _
    + " hangban.hoadonID WHERE Trim(khachID)='"&Trim(Combo0) _
    + " GROUP BY hoadonID, khachID, ngayban ")
```

```
    Set frm_formcon.Form.Recordset = rs
```

```
    frm_formcon.Requery
```

```
End Sub
```

Tóm lại, với mỗi bài toán đặt lọc áp dụng theo phương pháp kể trên đều giống nhau về cách làm việc, điểm khác nhau duy nhất là câu lệnh SQL. Có thể kết luận mấu chốt của bài toán này là tạo ra được lệnh SQL đúng để đáp ứng mọi điều kiện đặt lọc từ form mẹ.



### Chú ý

Việc viết ra câu lệnh SQL đúng là rất khó, đòi hỏi lập trình viên phải có nhiều kinh nghiệm. Để làm tốt được việc này, nên thiết kế một query ở chế độ design view; rồi sử dụng tính năng Copy, Paste để dán câu lệnh SQL mà query đã tạo lên nơi soạn thảo lệnh VBA và chỉnh sửa cho phù hợp. Thông thường sửa mệnh đề WHERE của câu lệnh.

# CHƯƠNG 7

## MENU & TOOLBAR

Chương này trình bày những kỹ thuật tạo menu và toolbar nhằm liên kết toàn bộ những gì đã làm được thành một hệ thống phần mềm hoàn chỉnh. Bao gồm các mảng kiến thức:

- Tạo Menu;
- Tạo Toolbar;
- Tạo form chính phần mềm

## 1. Tạo Menu

Menu là đối tượng được thiết kế có cấu trúc nhằm gắn kết các chức năng phần mềm lại để người dùng thuận tiện trong việc khai thác các tính năng cũng như CSDL.

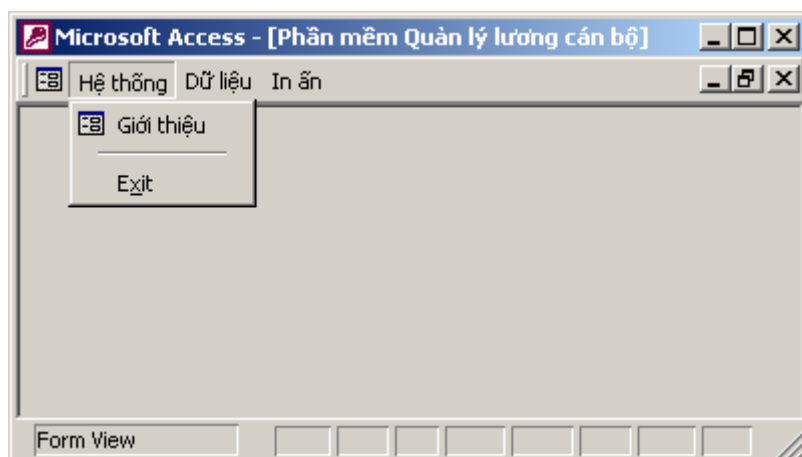
Có nhiều cách tạo menu trong Access, dưới đây chúng tôi trình bày cách đơn giản, dễ tạo và hiệu quả, đó là: sử dụng khả năng Customize của thanh công cụ.

Trước khi tạo menu, bạn phải đảm bảo đã xây dựng xong đầy đủ các tính năng cần thiết của phần mềm; bước này sẽ là cuối cùng, mang tính chất gắn kết các chức năng lại thành phần mềm hoàn chỉnh.

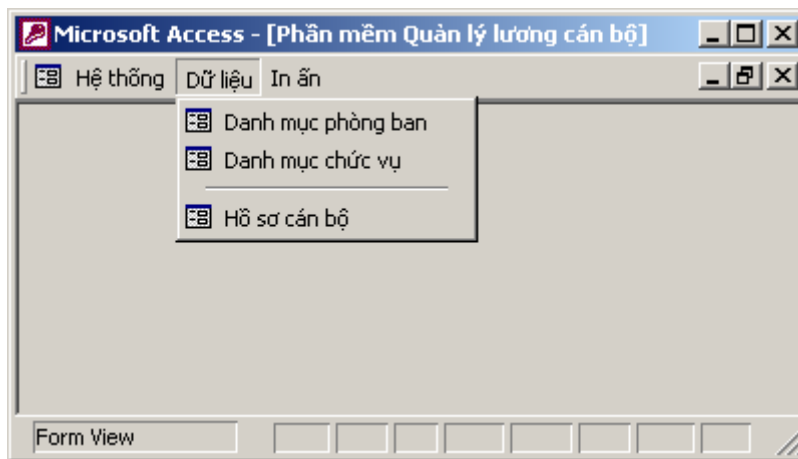
Giả sử trong CSDL Quản lý lương cán bộ đã làm được những việc sau:

1. Đã tạo xong CSDL
2. Đã tạo form nhập dữ liệu cho bảng *chucvu*, form nhập dữ liệu cho bảng *phongban*, form nhập dữ liệu cho bảng *canbo*;
3. Đã tạo xong report để in ra Danh sách chức vụ, Danh sách phòng ban;
4. Đã tạo xong form tìm kiếm cán bộ theo tên, in danh sách cán bộ một phòng ban
5. Đã tạo xong form để giới thiệu phần mềm (frmAbout)

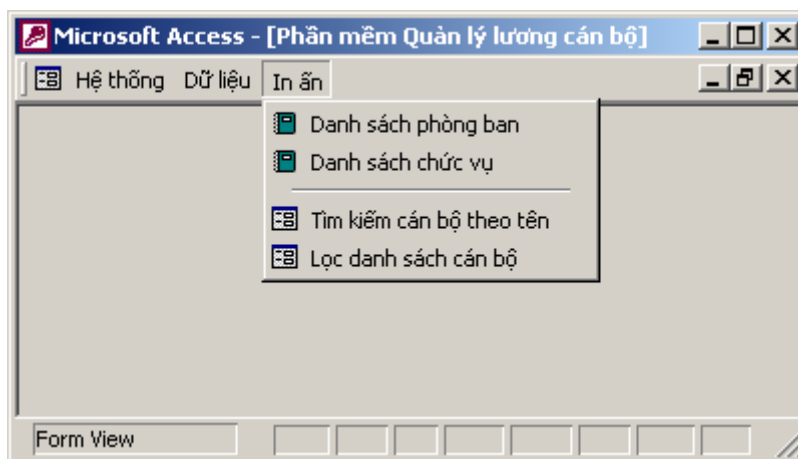
Dưới đây là hướng dẫn để tạo ra hệ thống menu có cấu trúc như sau:



(các mục chọn của menu Hệ thống)



(Các mục chọn của menu Dữ liệu)

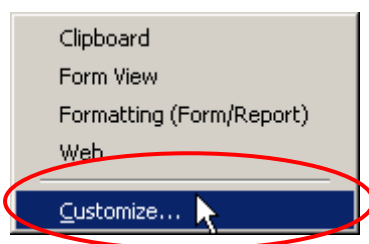


(Các mục chọn của menu In ấn)

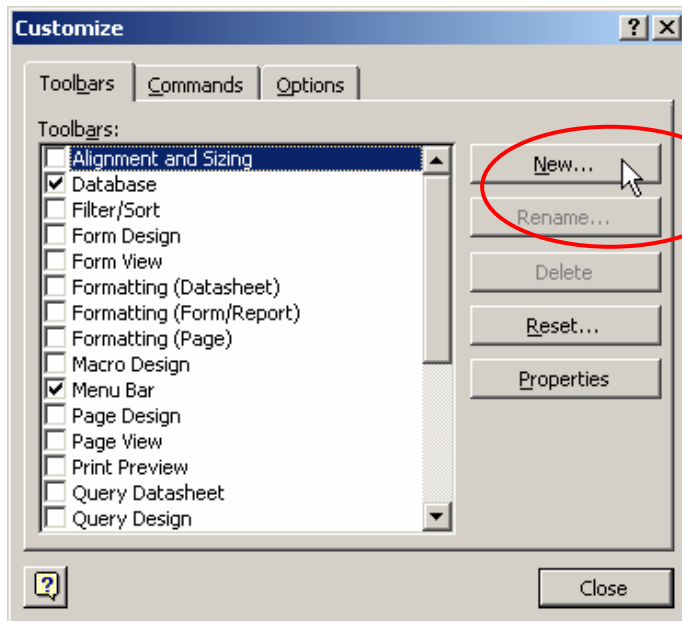
Các bước để tạo hệ thống menu trên như sau:

**Bước 1:** Tạo một Toolbar mới

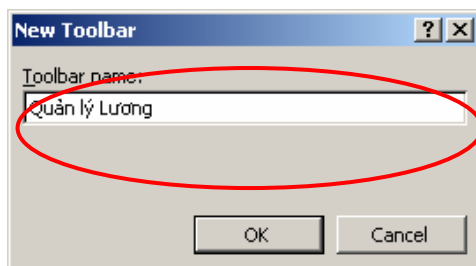
Nhấn chuột phải lên thanh công cụ của Access, chọn Customize



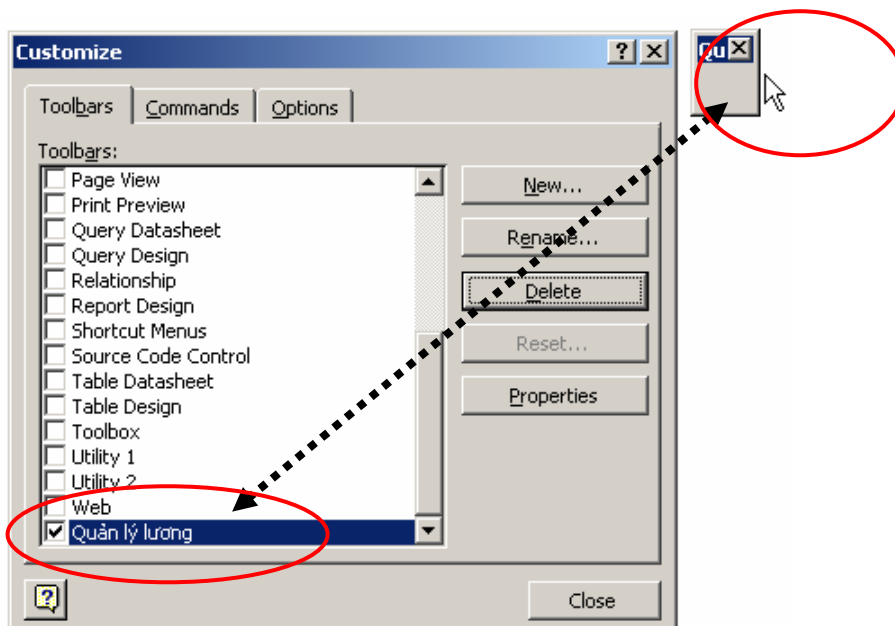
Hộp thoại **Customize** xuất hiện:



Để tạo mới một thanh công cụ, nhấn nút New. Hộp thoại sau xuất hiện, yêu cầu gõ vào tên cho thanh công cụ:

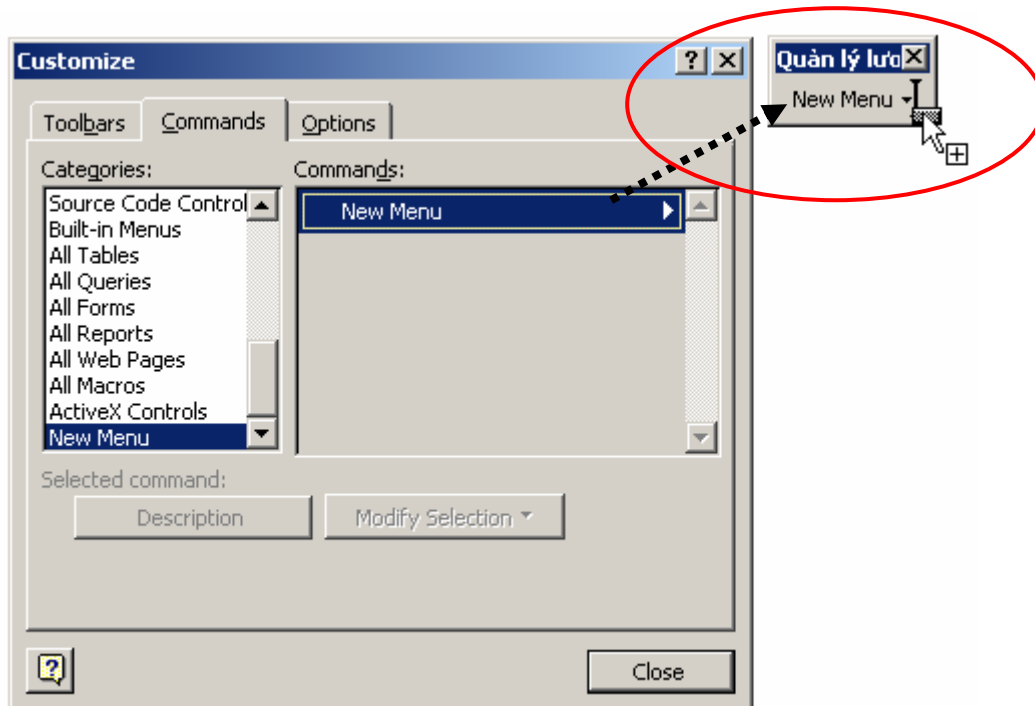


Hãy gõ tên cho thanh công cụ, giả sử *Quản lý lương*. Gõ xong nhấn **OK**. Khi đó mà hình làm việc xuất hiện một thanh công cụ trống. Công việc tiếp theo là xây dựng hệ thống Menu trên thanh công cụ này.



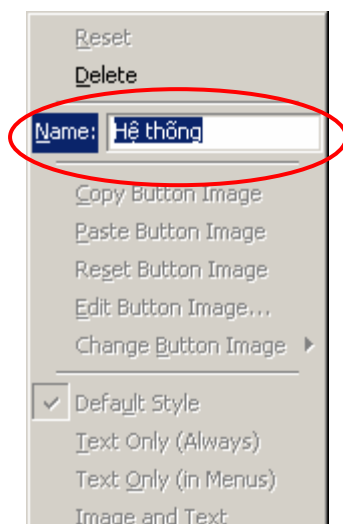
## Bước 2: Xây dựng các mục cấp 1

Các mục cấp 1 là: *Hệ thống*, *Dữ liệu* và *In ấn*. Để xây dựng các mục này, trên hộp thoại Customize mở thẻ **Commands**. Ở danh sách **Categories** chọn **New Menu** và lần lượt kéo mục **New Menu** trên danh sách Commands thả lên Toolbar đang thiết kế:



Hãy lần lượt kéo thả đủ số menu cấp 1 (3 menu cấp 1). Tiếp theo lần lượt thay đổi tiêu đề cho 3 menu này bằng cách:

- Nhấn chuột phải lên menu cần thay đổi;
- Gõ tiêu đề mới vào hộp *Name* (hình dưới):

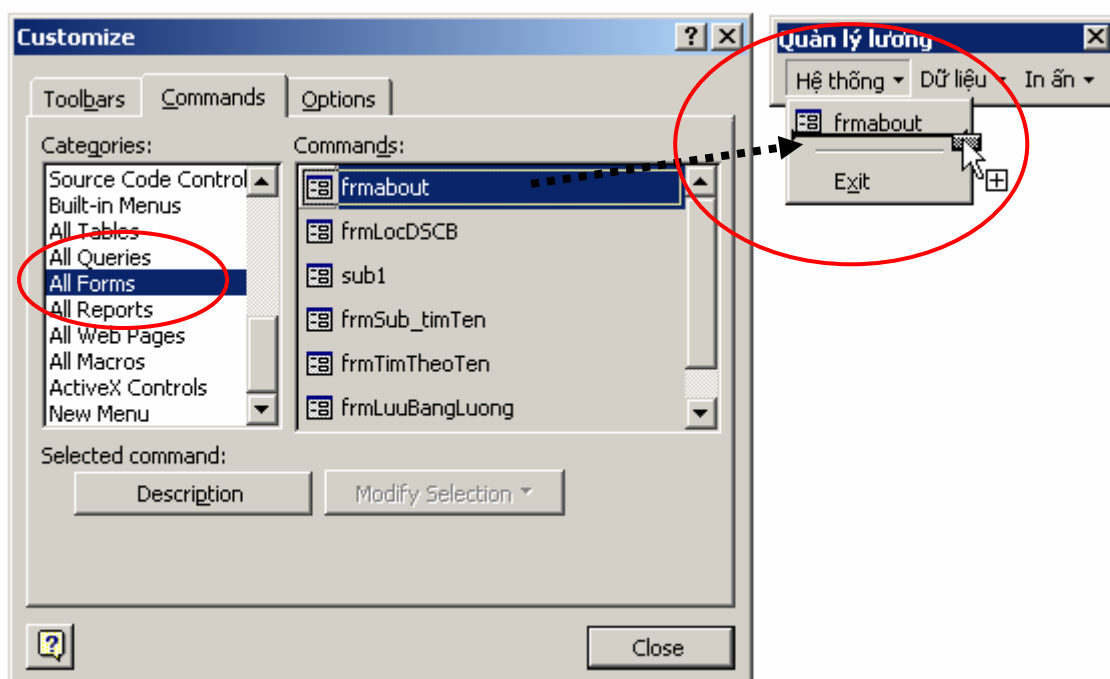




### Bước 3: Xây dựng các mục cấp con

3 mục tạo trên là cấp 1, các mục còn lại nằm trong 3 mục đó đều là mục con, cháu. Bước này sẽ xây dựng toàn bộ các mục con như vậy.

Mỗi mục con có thể là lời gọi một Form, một Report, một bảng,... (gọi tắt là đối tượng) ra để làm việc. Muốn mở đối tượng nào, hãy xác định chúng loại đối tượng đó bên danh sách **Categories**; tiếp theo dùng chuột kéo-thả đối tượng cần đưa lên menu từ danh sách **Commands** lên vị trí xác định trên menu đang thiết kế.



Hình trên minh họa cách đưa lời gọi đến form *frmAbout* lên menu *Hệ thống*.

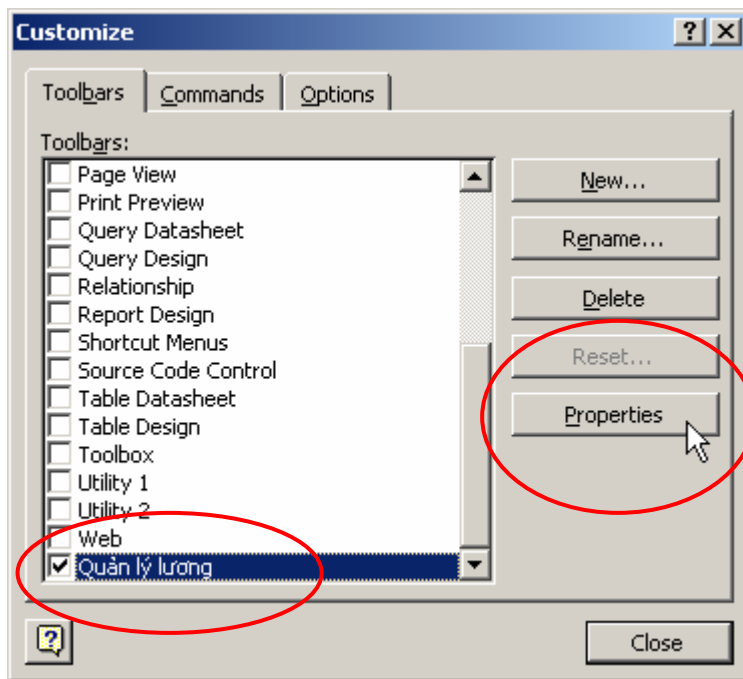
Tương tự hãy đưa các form và report còn lại lên vị trí thích hợp trên menu Quản lý lương. Cách đặt tên cho các menu này thực hiện tương tự như cách đặt tên cho 3 menu mẹ đã trình bày.

Đặc biệt mục *Exit* lấy ở **File | Exit**.

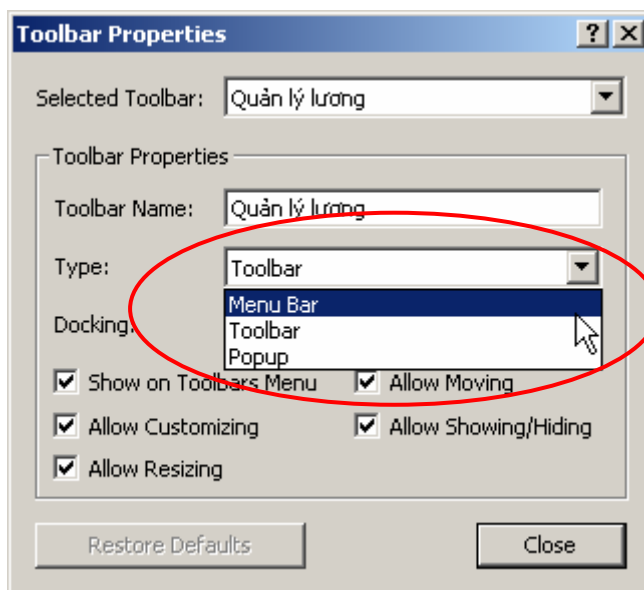
### Bước 4: Quyết định là Menu hay ToolBar

Với các bước làm việc như trên, Access ngầm hiểu là đang tạo một ToolBar. Bước này hướng dẫn cách tùy chọn lại là menu hay toolbar? Cách làm như sau:

Trở về thẻ ToolBars của hộp thoại Customize, chọn tên toolbar cần làm việc (*Quản lý lương*) và nhấn nút **Properties**:



Hộp thoại Toolbar Properties xuất hiện cho phép thiết lập các thuộc tính cho Toolbar đang chọn:



- **Toolbar Name** - để thiết lập tên cho Toolbar;
- **Type** - để xác định kiểu là Toolbar hay Menu?
- **Docking** - để tùy chọn các kiểu khoá Menu: không cho phép thay đổi lại (Can't Change); cho phép thay đổi lại Any Allow);
- **Show on Toolbars Menu** – cho phép hiển thị trên hệ thống thanh công cụ và menu;
- **Allow Customizing** – cho phép định nghĩa lại các mục;

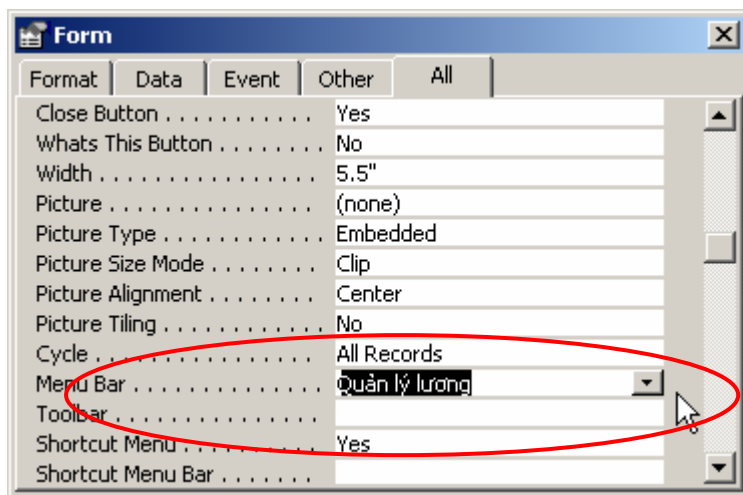
- **Allow Resizing** – cho phép thay đổi kích cỡ hiển thị;
- **Allow Moving** – cho phép di chuyển được;

Thiết lập xong nhấn **Close**.

## 2. Gắn kết Menu, Toolbar

Một Menu hay Toolbar sau khi đã tạo ra, muốn đi kèm với đối tượng nào phải thực hiện gắn kết vào đối tượng đó bằng cách thiết lập thuộc tính Toolbar - nếu muốn gắn Toolbar hoặc Menu Bar – nếu muốn gắn menu bar. Cách làm như sau:

**Bước 1:** Mở đối tượng cần gắn kết Menu hoặc ToolBar (ví dụ một form) ra chế độ Design View:



**Bước 2:** Chọn Menu Bar hoặc ToolBar cần gắn kết bằng cách thiết lập thuộc tính **ToolBar** hoặc **Menu Bar** (hình trên chọn Menu Bar *Quản lý lương*).

## 3. Tạo form chính

Form chính là form chứa hệ thống menu (hoặc toolbar) của phần mềm, nó xuất hiện ngay sau khi khởi động (Start-Up Object). Với mỗi phần mềm đóng gói hầu hết phải tạo form chính. Để từ đây có thể mở đến các chức năng phần mềm cần làm việc.

Trong một tệp Access, form chính là một form được thiết kế ở chế độ Design view; có gắn Menu (hoặc ToolBar) và được thiết lập khởi động đầu tiên mỗi khi tệp này được mở.

Dưới đây là hướng dẫn cách tạo form chính cho CSDL Quản lý lương như thể hiện ở mục 1. Tạo menu

### BƯỚC 1: Tạo frmMain

- Tạo một form mới ở chế độ Design View;
- Đặt tên form này là *frmMain*;
- Thiết lập một số thuộc tính cho form như sau:

Menu Bar *Quản lý lương*

Caption *Quản lý lương*

Record Selector *No*

Navigation Button *No*

Diving Line *No*

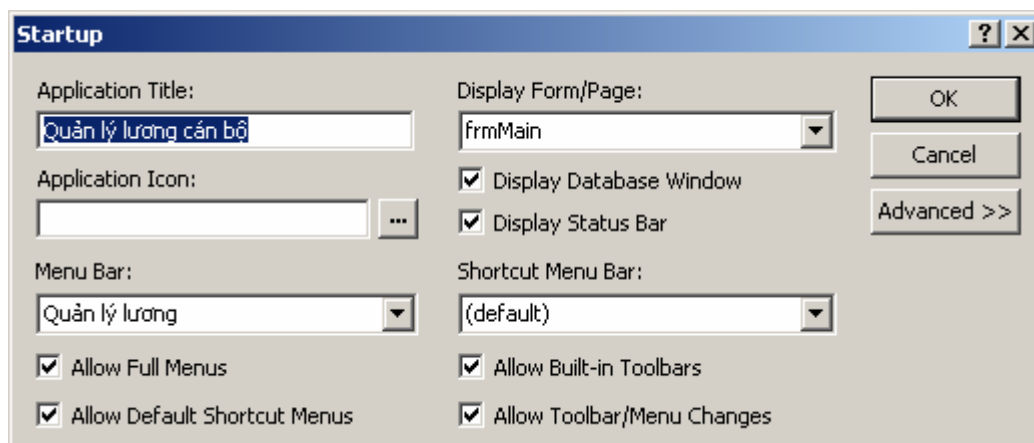
Để mỗi khi form này khởi động sẽ tự động phóng cực đại cửa sổ, hãy mở cửa sổ VBA và viết lệnh cho sự kiện **Form\_Open** như sau:

```
Private Sub Form_Open(Cancel As Integer)
    DoCmd.Maximize
End Sub
```

Ý nghĩa lệnh **DoCmd.Maximize** là phóng to (Maximize) form đang làm việc.

### BƯỚC 2: Thiết lập các thông tin về ứng dụng

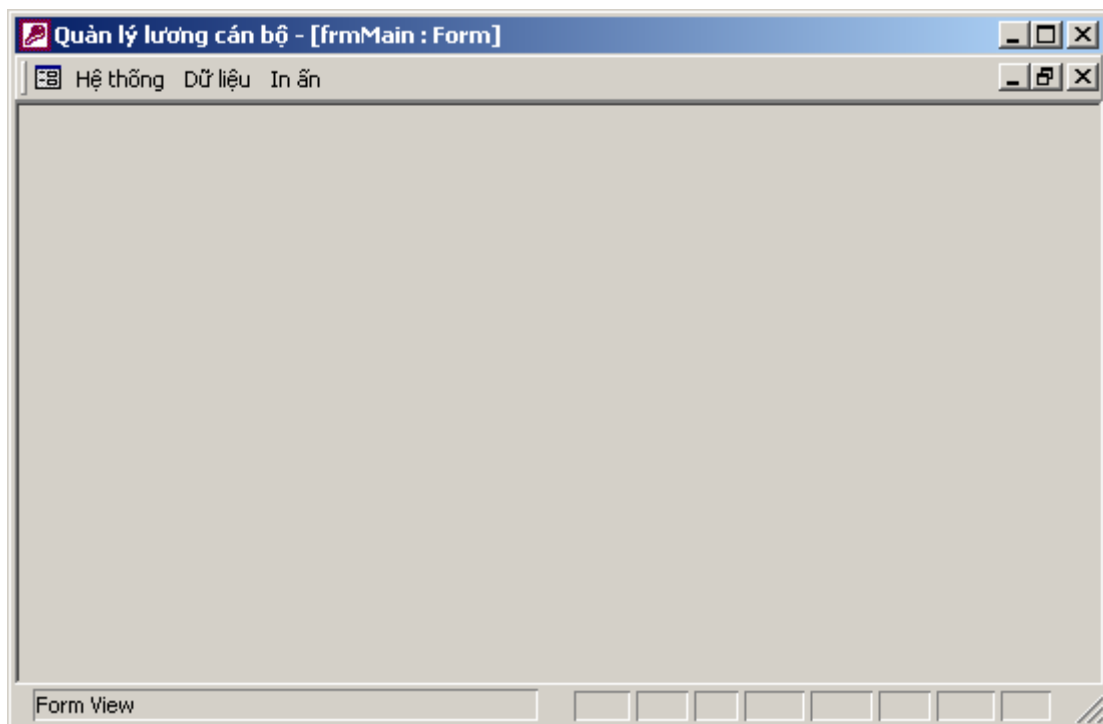
Mở thực đơn **Tools | Startup...**, hộp thoại **Startup** xuất hiện:



- Gõ tiêu đề phần mềm vào ô **Application Title**;

- Chọn form chính ở hộp **Display Form/Page** (chọn *frmMain*);
- Chọn Menu cho ứng dụng ở hộp **Menu Bar** (chọn menu Quản lý lương);
- Hộp **Application Icon** để chọn một biểu tượng cho ứng dụng (loại tệp \*.ico);
- **Display Database Window** Cho phép hiển thị cửa sổ Database hay không?;
- **Display Status Bar** Cho phép hiển thị thanh trạng thái?;
- **Allow Toolbar/Menu Changes** Cho phép thay đổi menu bar và toolbar?;
- ...

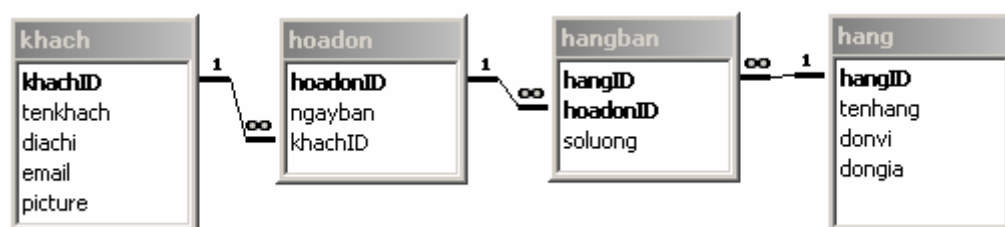
Thiết lập xong nhấn **OK** để đóng lại. Lần sau khi mở tệp CSDL này, màn hình đầu tiên bạn gặp là form *frmMain*:



## Bài tập

### Bài số 1: Xây dựng trên CSDL Quản lý bán hàng

1. Xây dựng CSDL theo cấu trúc như sau :



#### Yêu cầu :

- Thiết kế cấu trúc các bảng một cách phù hợp : kiểu dữ liệu các trường; trường khoá; thuộc tính Lookup và các thuộc tính khác;
- Thiết lập quan hệ cùng các thuộc tính đảm bảo toàn vẹn dữ liệu cho các quan hệ;

2. Tạo các form làm giao diện nhập dữ liệu cho các bảng :

**Lập hoá đơn bán hàng**

hoadonID:  Ngày bán:

Tên khách:

Địa chỉ:

E-mail:

**Chi tiết hàng bán:**

	hangID	donvi	dongia	soluong	thanhtien:
	Floppy disk	chiếc	8	3	24
	Hard disk	ổ	67	9	603
▶	CPU P4 2.4	con	133	2	266
*					

Record:  of 3

Tổng tiền:

Tiến > < Lùi Thêm mới Xoá HĐ In HĐ Thoát

Yêu cầu chi tiết:

- Các nút Tiến, Lùi để định vị hoá đơn cần làm việc;
- Nút Thêm mới để bắt đầu tạo mới một hoá đơn;
- Nút Xoá HĐ để xoá hoá đơn hiện tại. Yêu cầu phải có xác nhận trước khi xoá;
- Nút In HĐ để in chi tiết hoá đơn bán hàng ra report.

3. Thiết kế form tổng hợp thông tin bán hàng theo ngày như sau:

The screenshot shows a form window titled "Tổng hợp thông tin bán hàng..". The main title is "TỔNG HỢP THÔNG TIN BÁN HÀNG". Below the title, there are two date input fields: "Từ ngày: 1/1/2004" and "Đến ngày: 12/1/2004". Underneath, the text "Chi tiết hàng đã bán:" is followed by a table with the following data:

	Tên hàng:	Đơn vị:	Tổng số bán:
▶	CPU P4 2.4	con	15
	Floppy disk	chiếc	11
	Hard disk	ổ	9
	RAM 256	thanh	15

At the bottom of the table area, there is a record navigation bar showing "Record: 1 of 4" with navigation icons for first, previous, next, last, and refresh.

Sau khi nhập xong giá trị từ ngày và đến ngày, thông tin chi tiết về các mặt hàng bán ra trong phạm vi ngày đó được tổng hợp theo như giao diện trên.

4. Thiết kế form tổng hợp thông tin bán hàng cho khách hàng theo ngày như sau:

**TỔNG HỢP THÔNG TIN BÁN HÀNG**

Chọn khách hàng: Nguyễn Đức Ngô

Từ ngày: 1/1/2004 Đến ngày: 12/1/2004

**Chi tiết hàng đã bán:**

	Tên hàng:	Đơn vị:	Tổng số bán:
▶	CPU P4 2.4	con	5
	Floppy disk	chiếc	3
	Hard disk	ổ	9
	RAM 256	thanh	15

Record: 1 of 4

Yêu cầu: Sau khi chọn một khách hàng, nhập vào khoảng ngày cần tổng hợp. Thông tin tổng hợp về từng mặt hàng của khách đó mua sẽ được liệt kê.

5. Tạo form để theo dõi thông tin chi tiết bán hàng theo ngày như sau:

**THEO DÕI CHI TIẾT HÀNG BÁN**

Từ ngày: 1/10/2004 Đến ngày: 30/10/2004

**Chi tiết hàng bán:**

	tenhang:	10/10/2004:	12/10/2004:	24/10/2004:
▶	CPU P4 2.4	3	12	
	Floppy disk		11	
	Hard disk		9	
	RAM 256	12		3

Record: 1 of 4



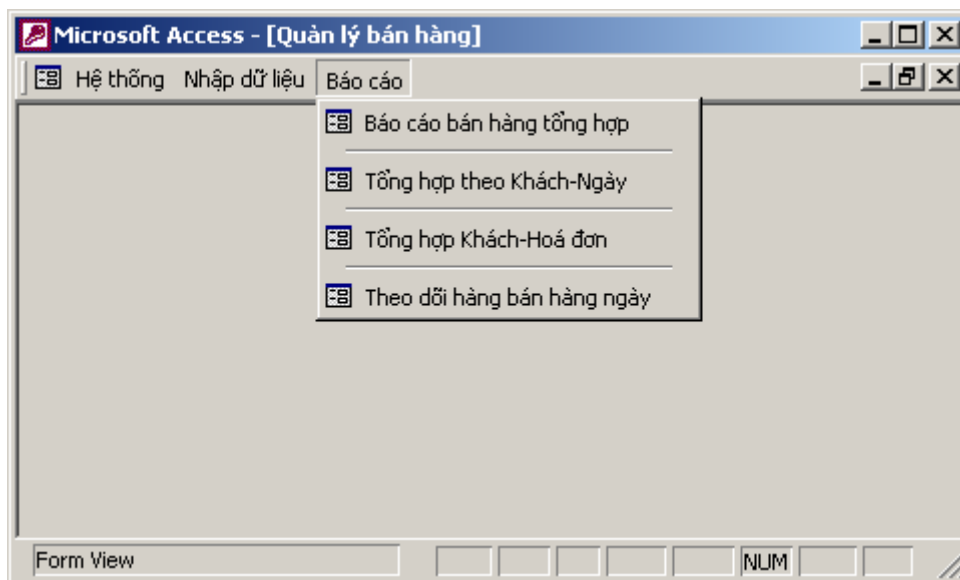
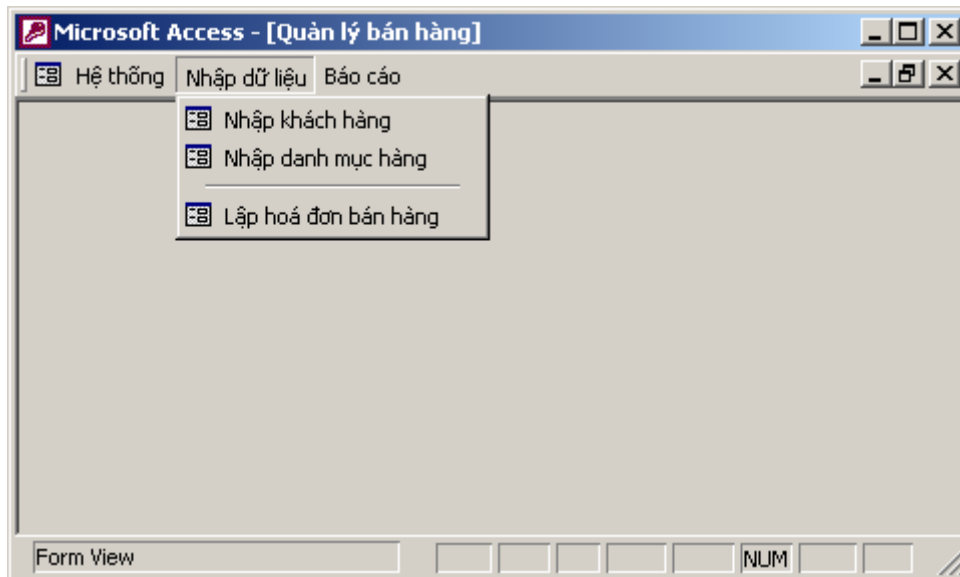
6. Tạo form để theo dõi hoá đơn mua hàng các khách hàng như sau:

	hoadonID:	ngayban:	Tongtien:
▶	1	12/10/2004	893
	2	24/10/2004	114
	3	10/10/2004	855
	4	12/10/2004	1394

Sau khi chọn tên một khách, thông tin về các hoá đơn mua hàng được hiển thị.

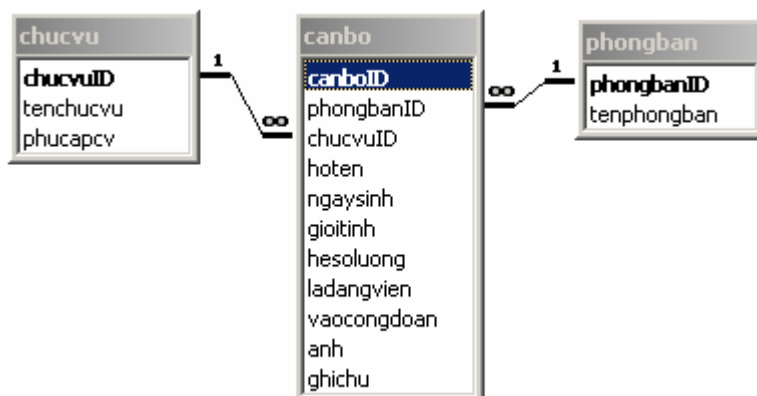
7. Tạo form frmAbout đưa những thông tin giới thiệu về phần mềm này.

8. Thiết kế form chính và menu bar để liên kết toàn bộ các chức năng đã làm được ở trên CSDL Quản lý bán hàng như sau:



**Bài số 2:** Xây dựng trên CSDL Quản lý lương cán bộ

1. Xây dựng CSDL như sau:



**Yêu cầu :**

- Thiết kế cấu trúc các bảng một cách phù hợp : kiểu dữ liệu các trường; trường khoá; thuộc tính Lookup và các thuộc tính khác;
- Thiết lập quan hệ cùng các thuộc tính đảm bảo toàn vẹn dữ liệu cho các quan hệ;

## 2. Thiết kế form làm giao diện nhập dữ liệu cho các bảng:

*frmPhongban* - cập nhật dữ liệu danh sách phòng ban;

*frmChucvu* - cập nhật danh mục chức vụ.

## 3. Thiết kế form để cập nhật hồ sơ cán bộ như sau:

The screenshot shows a form window titled "canbo" with the main heading "QUẢN LÝ HỒ SƠ CÁN BỘ". The form contains the following fields and controls:

- Mã cán bộ: [Text box]
- Họ tên: [Text box containing "Nguyễn Đức Ngô"]
- Ngày sinh: [Text box containing "12/04/1976"]
- Là nam giới
- Phòng ban: [Dropdown menu containing "Kinh doanh"]
- Chức vụ: [Dropdown menu containing "Giám đốc"]
- Hệ số lương: [Text box containing "3.12"]
- Là Đảng viên
- Vào Công đoàn
- Ghi chú: [Text box containing "Thương binh hạng 1/4"]

At the bottom of the form, there are four buttons: "Thêm hồ sơ", "Xoá hồ sơ", "In hồ sơ", and "Đóng". Below the buttons is a record navigation bar showing "Record: 1 of 12" with navigation icons.

3. Tạo form cho phép xem và in danh sách cán bộ từng phòng ban như sau:

Chọn phòng ban:

**Danh sách cán bộ:**

	Họ tên	Ngày sinh	Giới tính	Chức vụ
▶	Nguyễn Đức Ngô	12/04/1976	Nam	Giám đốc
	Phan Quang Sáng	03/09/1976	Nữ	Nhân viên
	Nguyễn Văn Hoà		Nữ	Nhân viên
	Phan Hoài Linh		Nam	Nhân viên
	Phan Anh Quân	23/07/1978	Nữ	Trưởng phòng
*				

Record:       of 5

4. Tạo form cho phép tìm kiếm cán bộ theo tên như sau:

Nhập tên cán bộ muốn tìm:

**Danh sách cán bộ tìm thấy:**

	hoten	ngaysinh	tenchucvu	tenphongban
▶	Nguyễn Đức Ngô	12/04/1976	Giám đốc	Kinh doanh
	Trần Văn Ngô		Nhân viên	Tổng hợp
*				

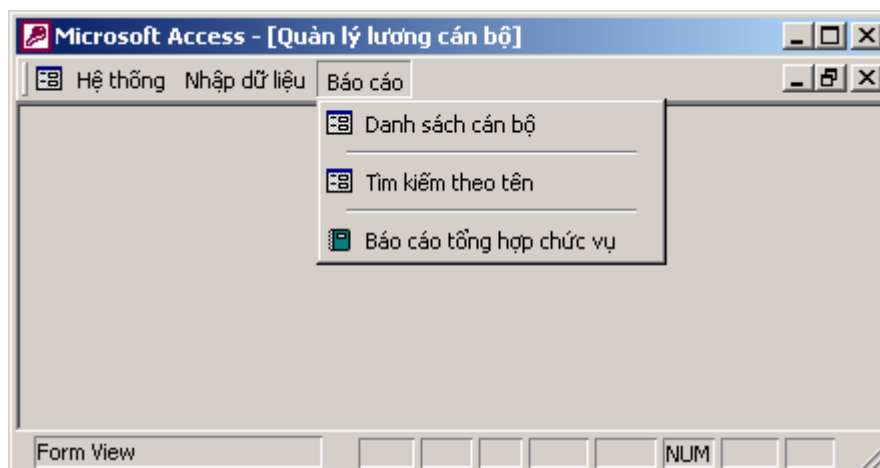
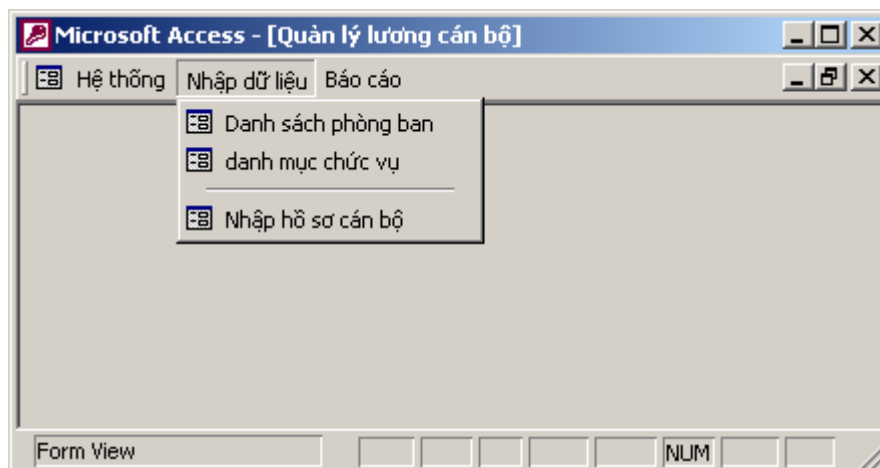
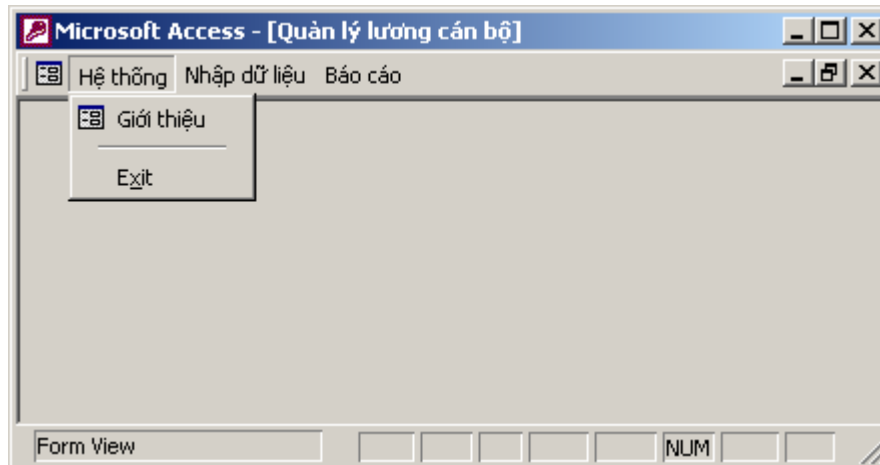
Record:       of 2

Sau khi gõ một tên (có thể là đệm + Tên hoặc đầy đủ họ và tên), danh sách các kết quả tìm thấy sẽ được liệt kê lên Subform.

5. Tạo report in ra báo cáo tổng hợp tổng số cán bộ theo chức vụ của từng phòng ban.

6. Tạo form frmAbout để giới thiệu các thông tin về phần mềm.

7. Tạo form chính cùng hệ thống menu gọi tới tất cả các chức năng đã làm đwocj như sau:



# THUẬT NGỮ TIN HỌC

Dưới đây là danh sách một số các thuật ngữ tin học hay gặp phải trong khi học và sử dụng Microsoft Access được xếp theo thứ tự từ điển để bạn đọc tham khảo.

<b>Append</b>	Chỉ hành động chèn thêm các bản ghi lên các bảng trong cơ sở dữ liệu
<b>Child</b>	Quan hệ con. Thuật ngữ này thường đi với Master – quan hệ cha
<b>Column</b>	Cột trong bảng dữ liệu
<b>Data</b>	Dữ liệu, dữ kiện, số liệu
<b>Data integrity</b>	Tính toàn vẹn dữ liệu. Một yếu tố rất quan trọng khi thiết kế một cơ sở dữ liệu.
<b>Data source</b>	Nguồn cung cấp dữ liệu, có thể là cung cấp cho một Form, cung cấp dữ liệu cho một Report.
<b>Data Table</b>	Bảng dữ liệu- nơi chứa dữ liệu vật lý của cơ sở dữ liệu
<b>Data type</b>	Kiểu dữ liệu. Dữ liệu trong cơ sở dữ liệu phải được định kiểu: kiểu số, kiểu chữ, kiểu lô gíc, ...- đó là các kiểu dữ liệu
<b>Database</b>	Cơ sở dữ liệu. Nơi chứa toàn bộ dữ liệu cho một mục đích quản lý nào đó, ở đó dữ liệu được thiết kế và lưu trữ theo các cấu trúc tối ưu.
<b>Datasheet</b>	Bảng dữ liệu- nơi dữ liệu hiển thị dưới dạng bảng bao gồm các cột và các dòng dữ liệu.
<b>Design View</b>	Môi trường thiết kế, có thể là thiết kế Form, thiết kế report, query,...
<b>Detete</b>	Chỉ hành động xóa các bản ghi ra khỏi cơ sở dữ liệu
<b>EOF</b>	End Of File – một dấu hiệu cho biết con trỏ bản ghi đã nằm ở vị trí cuối cùng của một recorset.

<b>Field</b>	Trường dữ liệu
<b>Field size</b>	Độ lớn của trường dữ liệu
<b>Form</b>	Đối tượng dùng thiết kế giao diện trong phát triển phần mềm. Kết quả của form khi sử dụng sẽ là các cửa sổ, hộp thoại- nơi mà người dùng có thể sử dụng để thao tác trên phần mềm.
<b>Form wizard</b>	Một trình điều khiển giúp đơn giản tạo một form nhập dữ liệu
<b>Formating</b>	Chỉ việc định khuôn dạng dữ liệu khi hiển thị
<b>Group By</b>	Một mệnh đề trong câu lệnh SQL dùng nhóm các bản ghi phục vụ các việc tổng hợp dữ liệu nào đó.
<b>Input mark</b>	Mặt nạ để nhập dữ liệu hay khuôn nhập dữ liệu
<b>Insert</b>	Chỉ hành động chèn thêm các bản ghi lên các bảng trong cơ sở dữ liệu
<b>Master</b>	Quan hệ cha. Thuật ngữ này thường đi với Child – quan hệ con
<b>Order By</b>	Một mệnh đề trong câu lệnh SQL dùng sắp xếp các bản ghi
<b>Query</b>	Một đối tượng dùng truy xuất, xử lý dữ liệu trên các hệ cơ sở dữ liệu
<b>Record</b>	Bản ghi
<b>Recordset</b>	Tập hợp các bản ghi
<b>Relationship</b>	Chỉ quan hệ giữa các bảng trong cơ sở dữ liệu
<b>Report wizard</b>	Một trình điều khiển giúp đơn giản tạo một report để hiển thị và in ấn dữ liệu
<b>Row</b>	Dòng trong một bảng dữ liệu
<b>Select</b>	Chỉ hành động chọn các bản ghi từ cơ sở dữ liệu
<b>SQL</b>	Structured Query Language – ngôn ngữ truy vấn dữ liệu có cấu trúc. Một ngôn ngữ xử lý dữ liệu khá phổ dụng trên các hệ cơ sở dữ liệu. Với Access, Query Design chính là nơi để thiết kế tạo ra các câu lệnh SQL thi hành các phép xử lý dữ liệu.

<b>Sub-form</b>	Form con- một form nằm trong một form khác. Khi đó form chứa sub-form gọi là form mẹ (form cha) thuật ngữ tiếng Anh là Master form.
<b>Table wizard</b>	Một trình điều khiển giúp đơn giản tạo một bảng dữ liệu
<b>Update</b>	Chỉ hành động cập nhật dữ liệu các bản ghi trong cơ sở dữ liệu
<b>Validation</b>	Một khái niệm chỉ sự đúng đắn của dữ liệu
<b>Where</b>	Một mệnh đề trong câu lệnh SQL để thiết lập điều kiện xử lý dữ liệu
<b>Wizard</b>	Là trình điều khiển giúp người lập trình đơn giản trong việc xây dựng ứng dụng. Có rất nhiều trình Wizard trong Acces như: Form wizard, report wizard, table wizard, database wizard, control wizard,...



# TÀI LIỆU THAM KHẢO

1. Microsoft Access 2000 Bible Quick Start; Cary N.Prague, Michael R. Iruin; Pulished by Hungry Minds, Inc, 2002.
2. Microsoft Access 2000 with VBA – Advanced; Al Napier, Phil Judd, H. Albert Napier, Philip J. Judd; Pulished by Kris Oxford, 2000.
3. Mastering Microsoft Office 2000 Professional Edition; Gini Courter and Annette Marquis; Pulished by Oxford Express, 1999.
4. Tài liệu từ Mạng Giáo dục Việt Nam  
<http://www.edu.net.vn/Default.aspx?tabindex=1&tabid=6&mid=18>
5. Thông tin sản phẩm và các tài liệu học tập liên quan tại Website của Công ty Microsoft <http://office.microsoft.com/en-ca/FX010857911033.aspx>.

# MỤC LỤC

<b>GIỚI THIỆU .....</b>	<b>1</b>
<b>BÀI MỞ ĐẦU .....</b>	<b>3</b>
1. Giới thiệu Access 2000 .....	4
2. Khởi động.....	5
3. Tạo mới tệp Access.....	6
4. Môi trường làm việc.....	8
5. Mở tệp đã tồn tại .....	8
6. Thoát khỏi Access.....	9
<b>CHƯƠNG 1: XÂY DỰNG CƠ SỞ DỮ LIỆU .....</b>	<b>10</b>
1. Các khái niệm về CSDL Access .....	11
1.1 CSDL Access .....	11
1.2 Bảng dữ liệu .....	12
1.3 Liên kết các bảng dữ liệu .....	15
2. Xây dựng cấu trúc bảng .....	17
3. Thiết lập quan hệ.....	23
4. Nhập dữ liệu.....	26
4.1 Cách nhập dữ liệu.....	26
4.2 Một số thao tác xử lý dữ liệu trên bảng.....	28
<i>a. Xoá bản ghi.....</i>	<i>28</i>
<i>b. Sắp xếp dữ liệu.....</i>	<i>29</i>
<i>c. Lọc dữ liệu.....</i>	<i>29</i>
5. Thuộc tính LOOKUP .....	30
6. Quy trình xây dựng CSDL Access .....	35
Bài tập .....	36
<b>CHƯƠNG 2: TRUY VẤN DỮ LIỆU .....</b>	<b>40</b>
1. SELECT queries .....	41
1.1 Cách tạo.....	41
1.2 Lọc dữ liệu .....	46
2. TOTAL queries .....	50
3. CROSSTAB queries .....	52
4. MAKE TABLE queries .....	55
5. DELETE queries .....	58
6. UPDATE queries .....	59
Bài tập .....	62

<b>CHƯƠNG 3: THIẾT KẾ GIAO DIỆN.....</b>	<b>66</b>
1. Khái niệm Forms.....	67
2. Sử dụng FORM WIZARD.....	67
3. Sử dụng FORM DESIGN VIEW.....	72
3.1 Thiết kế Form nhập dữ liệu đơn giản.....	72
3.2 Tinh chỉnh cấu trúc Form.....	76
4. Kỹ thuật Sub-form.....	81
Bài tập.....	93
<b>CHƯƠNG 4: THIẾT KẾ BÁO CÁO.....</b>	<b>98</b>
1. Các khái niệm về Report.....	99
1.1 Cấu trúc Report.....	99
1.2 Môi trường làm việc.....	100
2. Sử dụng Report wizard.....	101
3. Thiết kế report.....	107
4. Report chứa tham số.....	115
Bài tập.....	121
<b>CHƯƠNG 5: LẬP TRÌNH VBA CĂN BẢN.....</b>	<b>123</b>
1. Môi trường lập trình VBA.....	124
2. Các kiểu dữ liệu và khai báo.....	126
2.1 Các kiểu dữ liệu cơ bản.....	126
2.2 Biến và cách sử dụng biến.....	128
a. <i>Biến – khai báo biến đơn giản</i> .....	128
b. <i>Phạm vi biến</i> .....	130
2.3 Hằng và cách sử dụng hằng.....	131
a. <i>Khai báo hằng</i> .....	131
b. <i>Phạm vi hằng</i> .....	132
3. Các cấu trúc lệnh VBA.....	132
3.1 Cấu trúc IF... END IF.....	133
3.2 Cấu trúc SELECT CASE .. END SELECT.....	134
3.3 Cấu trúc FOR ... NEXT.....	137
3.4 Cấu trúc WHILE ... WEND.....	139
3.5 Lệnh DoCmd.....	140
4. Chương trình con.....	143
4.1 Chương trình con dạng hàm.....	144
4.2 Chương trình con dạng thủ tục.....	148
5. Kỹ thuật xử lý lỗi.....	150
5.1 Xử lý lỗi.....	150
5.2 Bẫy lỗi.....	154
6. Một số ví dụ.....	156

<b>CHƯƠNG 6: LẬP TRÌNH CƠ SỞ DỮ LIỆU.....</b>	<b>162</b>
1. Kỹ thuật DAO .....	163
1.1 Lớp đối tượng DAO .....	164
1.2 Đối tượng Database.....	166
1.3 Đối tượng RecordSet.....	167
1.4 Đối tượng QueryDef .....	172
1.5 Đối tượng TableDef .....	175
1.6 Đối tượng Relation.....	178
2. Bài toán đặt lọc dữ liệu .....	178
<b>CHƯƠNG 7: MENU &amp; TOOLBAR .....</b>	<b>183</b>
1. Tạo Menu .....	184
2. Gắn kết Menu, Toolbar .....	190
3. Tạo form chính.....	190
Bài tập .....	193
<b>THUẬT NGỮ TIN HỌC .....</b>	<b>202</b>
<b>TÀI LIỆU THAM KHẢO .....</b>	<b>204</b>



**Giáo trình**

**Access 2007**

<b>Bài 1. Cơ bản về Access 2007</b> .....	4
<b>1.1. Khởi động, tìm hiểu giao diện</b> .....	4
1.1.1. Khởi động Microsoft Access 2007.....	4
1.1.2. Getting Started with Microsoft Office Access ..	8
1.1.3. Ribbon .....	10
1.1.4. Thanh công cụ Quick Access .....	19
1.1.5. Navigation Pane .....	21
1.1.6. Tab document.....	30
1.1.7. Status bar .....	33
1.1.8. Mini toolbar.....	36
1.1.9. Tạo Database đầu tiên.....	37
<b>1.2. Bắt đầu làm việc với cơ sở dữ liệu</b> .....	47
1.2.1. Lần đầu tiên sử dụng Access.....	47
1.2.2. Tạo một DataBase sử dụng Template.....	47
1.2.3. Tải các Template từ Microsoft Office Online	55
1.2.4. Tạo DataBase từ đầu.....	58
1.2.5. Tạo một DataBase rỗng .....	58
1.2.6. Thêm một bảng vào Database.....	62
1.2.7. Mở một Database đã tồn tại.....	75
1.2.8. Nếu không tìm thấy Database cần mở.....	77
1.2.9. Mở Database làm việc gần đây nhất.....	79
<b>1.3. Có gì mới trong Microsoft Office Access 2007</b> .....	83
1.3.1. Để bắt đầu nhanh hơn.....	83
1.3.2. Microsoft Office fluent.....	90
1.3.3. Thêm các công cụ mạnh để tạo đối tượng .....	95
1.3.4. Những kiểu dữ liệu và điều khiển mới .....	101
1.3.5. Nâng cấp thiết kế và các công cụ phân tích. ....	103
1.3.6. Tăng cường an ninh .....	108
1.3.8. Tốt hơn cách để gỡ rối vấn đề.....	112
1.3.9. Nâng cao chất lượng các công cụ Proofing..	112
<b>Bài 2. Làm việc với Table</b> .....	113
<b>2.1. Tổng quát</b> .....	113
2.1.1. Thuộc tính của bảng và trường .....	114
2.1.2. Kiểu dữ liệu .....	117

2.1.3. Quan hệ giữa các bảng.....	119
2.1.4. Khóa .....	120
2.1.5. Lợi ích của việc sử dụng relationships.....	122
2.1.6. Trước khi bạn bắt đầu .....	123
2.2. Tạo mới một bảng.....	123
2.3.1. Tạo bảng mới trong Database mới. ....	124
2.3.2. Tạo bảng mới trong Database đã tồn tại.....	126
2.3.3. Sử dụng bảng Template để tạo bảng .....	129
2.3.4. Tạo bảng mới sử Template .....	130
2.3.5. Thiết lập khóa chính cho bảng.....	133
2.3.6. Thiết lập thuộc tính cho bảng .....	138
2.3. Thêm trường vào bảng .....	140
2.4.1. Thêm trường khi đang nhập dữ liệu .....	141
2.4.2. Thêm trường sử dụng trường Template .....	141
2.4.3. Thêm trường từ trường của bảng khác.....	144
2.4.4. Thiết lập thuộc tính cho trường.....	151
Bài 3. Làm việc với Query.....	163
3.1. Tìm kiếm một nhóm dữ liệu trong bảng .....	165
3.2. Xem dữ liệu đồng thời từ nhiều bảng.....	169
3.3. Sử dụng Parameters với Query .....	174
3.4. Thực hiện tính toán trên cơ sở dữ liệu .....	180
3.5. Tổng hợp hoặc tính tổng dữ liệu .....	185
3.6. Tạo một bảng từ nhiều bảng khác.....	192
3.7. Thêm dữ liệu vào bảng từ nhiều bảng khác.....	200
3.8. Thay đổi dữ liệu tự động.....	202
3.9. Xóa dữ liệu tự động.....	204
Bài 4. Làm việc với Form .....	206
4.1. Tạo Form.....	206
4.2. Tìm hiểu về các Controls. ....	218
4.3. Tìm hiểu về Control Layout. ....	220
4.4. Tìm Layout View và Design View. ....	226
4.5. Tinh chỉnh Forn .....	227
4.6. Thêm Control lên Form.....	229
4.7. Lưu lại các công việc của bạn .....	231

<b>Bài 5. Làm việc với Report</b> .....	235
<b>5.1. Lựa chọn bản ghi nguồn.</b> .....	235
<b>5.2. Tạo báo cáo</b> .....	235
<b>5.3. Tìm hiểu về Report section</b> .....	246
<b>5.4. Tìm hiểu về các Control</b> .....	247
<b>5.5. Tinh chỉnh báo cáo</b> .....	247
<b>5.6. Thêm Control vào báo cáo</b> .....	247
<b>5.7. Xem báo cáo</b> .....	247
<b>5.8. In báo cáo</b> .....	253



## **Bài 1. Cơ bản về Access 2007**

### **1.1. Khởi động, tìm hiểu giao diện**

Bài viết này bàn về các cải tiến trên giao diện người dùng của Microsoft Office Access 2007 (UI). Giao diện người dùng mới của Microsoft Office Access 2007 là kết quả của việc nghiên cứu kỹ lưỡng và kiểm tra rộng rãi, nó được thiết kế để làm cho việc tìm kiếm những lệnh mà bạn cần trở nên dễ dàng hơn.

Bạn sẽ tìm hiểu về các yếu tố mới trên giao diện người dùng - những gì nó cung cấp và làm thế nào để làm việc với chúng. Bạn cũng sẽ tìm hiểu làm thế nào để kích hoạt tính năng tabbed tài liệu trong một cơ sở dữ liệu, và làm thế nào để định hướng những đối tượng giao diện mới.

Giao diện người sử dụng mới trong Microsoft Access 2007 bao gồm một số các yếu tố xác định cách thức bạn tương tác với các sản phẩm. Những yếu tố mới này được chọn để giúp bạn làm chủ Access, và giúp bạn tìm thấy nhanh hơn những lệnh mà bạn cần. Cách thiết kế mới cũng giúp bạn dễ dàng khám phá các tính năng khác có thể còn vẫn ẩn dưới lớp thanh công cụ và trình đơn. Bạn cũng sẽ bắt đầu nhanh hơn nhờ vào trang Getting Started trong Microsoft Office Access, trong đó cung cấp cho bạn khả năng tiếp cận nhanh hơn khi bạn chưa có nhiều kiến thức, bao gồm một bộ các Template được thiết kế chuyên nghiệp.

Yếu tố quan trọng nhất trên giao diện mới được gọi là yếu tố Ribbon, nó là một phần của giao diện Microsoft Office fluent. Các Ribbon trải dài phía trên cùng của cửa sổ chương trình và chứa các nhóm lệnh. Microsoft Office fluent Ribbon cung cấp các lệnh, nhóm lệnh thay thế cho Menu và thanh công cụ cổ điển. Trên Ribbon là các Tab kết hợp những lệnh có cùng ý nghĩa. Ribbon chính các Tab chính mà bạn thường

thấy là Home, Create, External Data, and Database. Mỗi Tab lại chứa các nhóm lệnh có liên quan đến nhau.

Các thành phần giao diện mới trong Microsoft Access 2007 là:

- Getting Started with Microsoft Office Access: Trang này được hiển thị khi bạn bắt đầu với Access.

- Microsoft Office Fluent Ribbon: Là khu vực ở phía trên cùng của cửa sổ chương trình, nơi bạn có thể chọn các lệnh.

Tab Command: Những lệnh có cùng ý nghĩa được kết hợp với nhau.

+ Nội dung lệnh Tab: Một thẻ lệnh xuất hiện tùy thuộc vào bối cảnh, những đối tượng mà bạn đang làm việc hoặc công việc mà bạn đang thực hiện.

+ Gallery: Là điều khiển hiển thị một sự lựa chọn trực quan qua đó bạn có thể nhìn thấy những kết quả mà bạn thực hiện.

+ Quick Access Toolbar: Một thanh công cụ chuẩn trên Ribbon cho phép người dùng nhấp chuột chọn tới những lệnh cần sử dụng như Save, Undo, ....

- Navigation Pane: Nằm ở khu vực bên trái cửa sổ hiển thị các đối tượng trong Database của bạn. Pane Navigation thay thế của sổ DataBase mà bạn thường thấy ở các phiên bản cũ của Access.

- Tabbed documents: Các đối tượng như Table, Query, Form, Report, Page, và các Macro của bạn được hiển thị trong Tabbed Document.

- Status bar: Nằm ở dưới cùng của cửa sổ chương trình sẽ hiển thị các thông tin trạng thái về hoạt động, sự kiện mà bạn vừa thao tác, ngoài ra nó còn chứa các nút lệnh cho phép bạn thay đổi chế độ hiển thị đối tượng hiện thời.

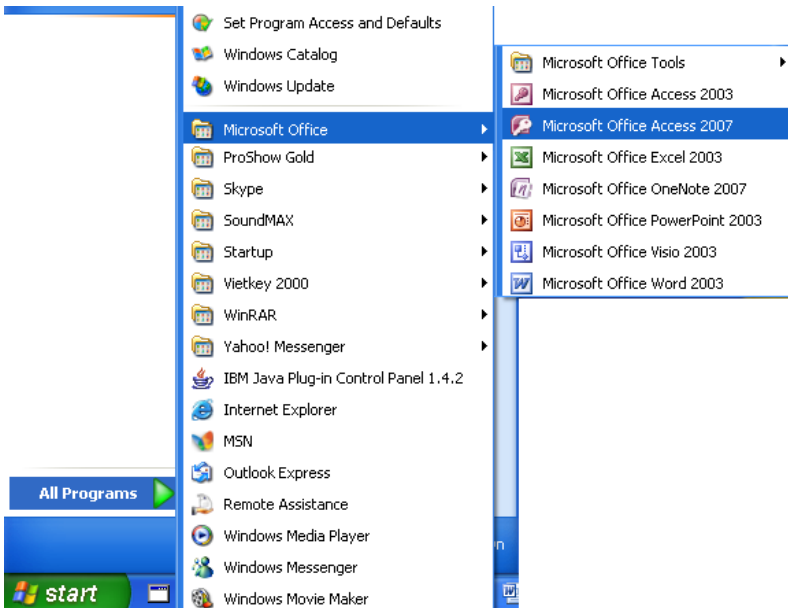
### **1.1.1. Khởi động Microsoft Access 2007**

Để khởi động ứng dụng bạn có thể làm theo các cách sau:

- Từ Desktop nhấp đúp vào biểu tượng Microsoft Office Access 2007 (nếu có).



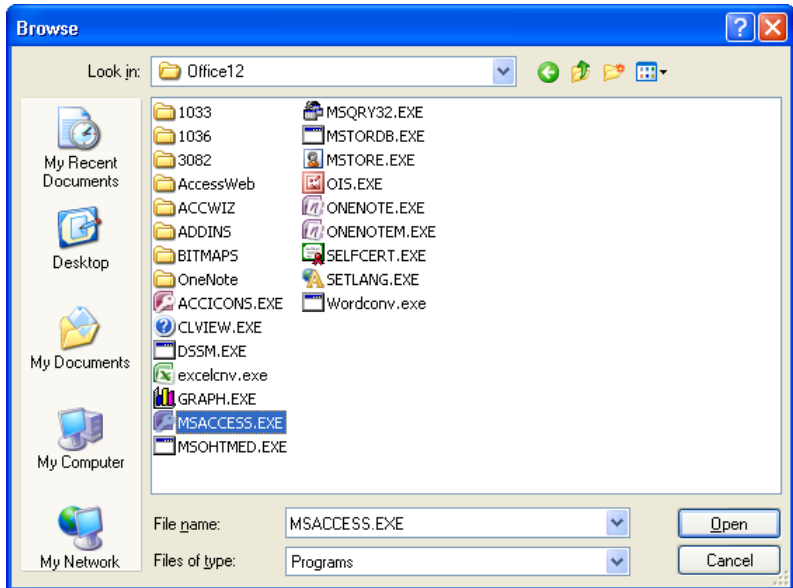
- Từ thanh Start nhấp chọn All Programs/Microsoft Office/Microsoft Office Access 2007.



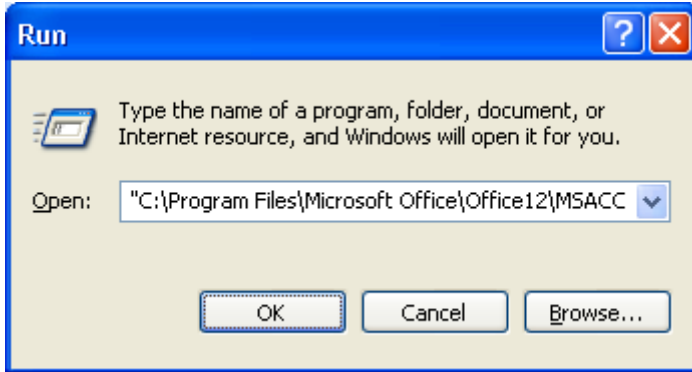
- Cách thứ 3 bạn nhấp chọn Start/Run... Hộp thoại xuất hiện nhấp chọn nút Browse. Trong hộp thoại này bạn trở tới đường dẫn:

"C:\Program Files\Microsoft Office\ Office12\MSACCESS.EXE"

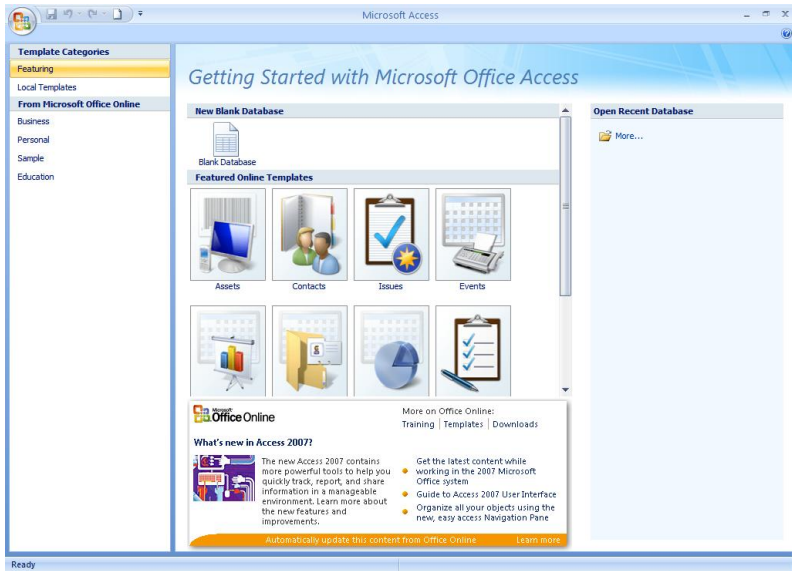
Rồi nhấp Open.



Cuối cùng nhấp chọn Ok để kết thúc.



Bằng một trong 3 cách trên bạn màn hình làm việc Microsoft Access 2007 sẽ xuất hiện như hình dưới đây.




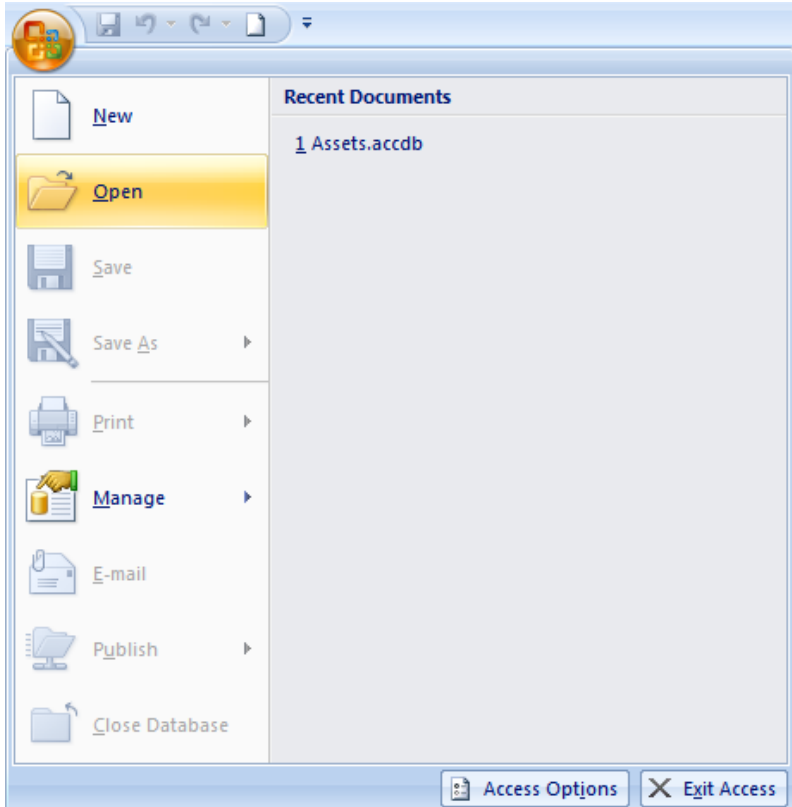
### **1.1.2. Getting Started with Microsoft Office Access**

Khi bạn bắt đầu với Microsoft Access 2007, trang Getting Started with Microsoft Office Access sẽ xuất hiện như hình dưới đây. Trang này cho thấy bạn có thể làm gì để bắt đầu với Microsoft Access 2007.



Ví dụ, bạn có thể tạo mới một DataBase trắng, Tạo một DataBase từ Template, hoặc mở một DataBase đã tồn tại (nếu bạn đã mở một vài DataBase trước đó). Bạn cũng có thể truy cập trực tiếp tới Microsoft Office Online để học thêm về về hệ thống Microsoft Office 2007 và Microsoft Access

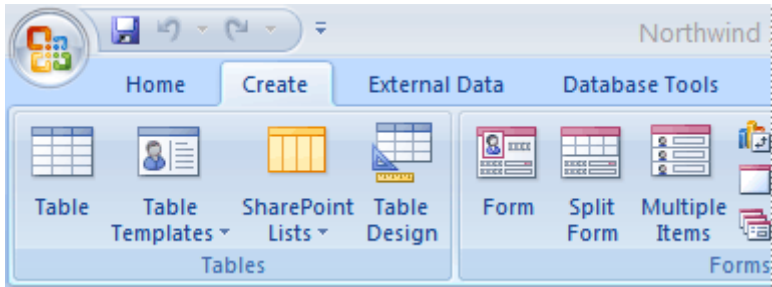
2007, hoặc bạn cũng có thể nhấp chọn nút Office Button  và sử dụng menu Open để mở một DataBase đã tồn tại trên đĩa cứng.



### **1.1.3. Ribbon**

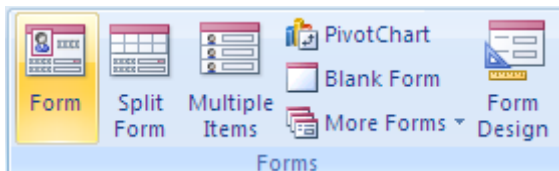
Office Fluent Ribbon là một sự thay thế chính cho các menu và thanh công cụ và cung cấp các lệnh chính trên giao diện của Microsoft Access 2007. Một trong những lợi ích chính của Ribbon là củng cố tập chung tất cả các yêu cầu về một điểm mà trước đây nó nằm trên Menu, ToolBar, Task Pane và các thành phần giao diện khác. Với Ribbon bạn chỉ phải quan tâm tới một nơi ở đó chứa tất cả các lệnh mà bạn cần để làm việc với Microsoft Access 2007.

Khi bạn mở một Cơ sở dữ liệu, Ribbon sẽ xuất hiện trên đỉnh của cửa sổ chính trong Office Access 2007.



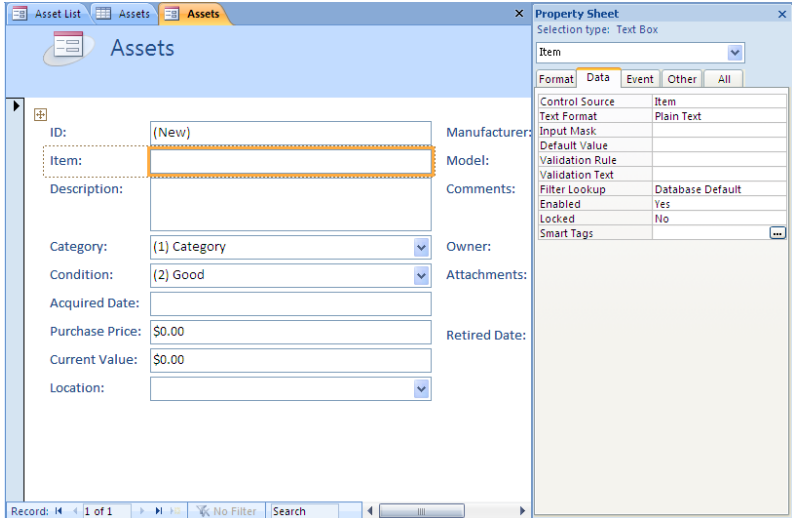
Các Ribbon chứa một loạt các Tab lệnh có chứa lệnh. Trong Microsoft Access 2007, Các Tab lệnh chính là Home, Create, External Data, và Database Tools. Mỗi tab chứa các nhóm lệnh liên quan.

Những lệnh trên Ribbon đưa vào “tài khoản” của đối tượng hiện đang hoạt động. Ví dụ, nếu bạn có một bảng mở ở Datasheet View và bạn nhấp chuột vào Form trên Tab Create, trong nhóm Forms.

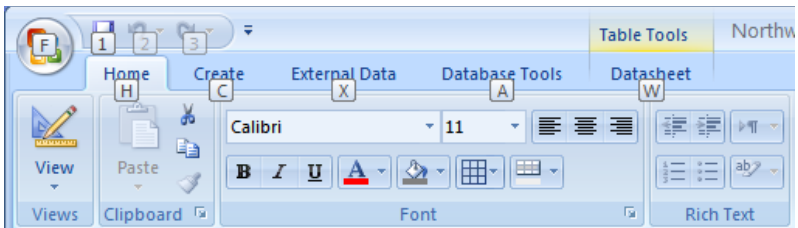


Microsoft Access 2007 sẽ tạo ra một Form mới dựa trên Table đang hoạt động. Và tên trường sẽ được gắn vào thuộc tính Control Source của từng Control trên Form.





Bạn có thể sử dụng các phím tắt với Ribbon. Tất cả các phím tắt ở phiên bản trước đều có thể sử dụng trên phiên bản này. Keyboard Access System thay thế các menu ở những phiên bản cũ. Đây là hệ thống sử dụng các chỉ số nhỏ xuất hiện trên giao diện Ribbon những phím nào có thể truy cập được các chức năng.

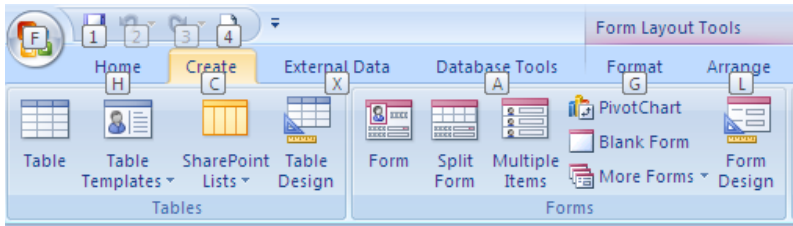


Khi bạn đã chọn một tab lệnh, bạn có thể duyệt qua các lệnh có sẵn trong tab đó.

### ***Chọn Tab lệnh***

***Bước 1:*** Khởi động Microsoft Access 2007

**Bước 2:** Nhấp chuột chọn tab cần sử dụng hoặc nhấp phím ALT khi đó trên màn hình sẽ xuất hiện như hình dưới đây:



Bạn muốn chọn nút nào thì nhấn phím tương ứng như hướng dẫn trên màn hình.

Có rất nhiều cách khác nhau để thực hiện một lệnh. Nhưng nhanh và trực tiếp là cách sử dụng các tổ hợp phím nóng. Nếu bạn đã từng sử dụng các phím nóng này trong những phiên bản trước của Access thì bạn sẽ rất dễ dàng khi làm việc với Microsoft Access 2007.

### ***Thực hiện một lệnh***

**Bước 1:** Khởi động Access

**Bước 2:** Nhấp chọn Tab thích hợp tương ứng với lệnh cần sử dụng. Bảng dưới đây là một số lệnh chung nhất trong các Tab trên Ribbon, tuy nhiên các Tab lệnh cũng sẽ thay đổi tùy thuộc vào những gì bạn đang làm việc.

<b>Tab lệnh</b>	<b>Những điều bạn có thể làm</b>
Home	Chọn chế độ hiển thị
	Sao chép và dán từ Clipboard
	Đặt lại Font chữ hiện tại

Đặt lại Alignment cho nội dung

Định dạng lại nội dung cho trường Memo

Làm việc với bản ghi (Refresh, New, Save, Delete, Totals, Spelling).

Sắp xếp và lọc bản ghi

Tìm kiếm bản ghi.

Create

Tạo một Table mới

Tạo một bảng mới sử dụng Template

Tạo mới một bảng trắng ở chế độ Design view.

Tạo mới một Form dựa trên Table hoặc Query đang hoạt động.

Tạo mới một bảng hoặc một biểu đồ pivot.

Tạo mới một báo cáo dựa trên một bảng hoặc một Query đang hoạt động.

Tạo mới một Query, Macro, Module hoặc một Class Module

External Data Import hoặc liên kết với dữ liệu bên ngoài.

Kết xuất dữ liệu

Thu thập và cập nhật dữ liệu từ Email

Làm việc với danh sách offline SharePoint.

Di chuyển một số hay toàn bộ các phần của cơ sở dữ liệu tới một SharePoint site mới hoặc đã tồn tại trước đó.

Database Tools

Chạy Visual Basic editor hoặc chạy macro.

Tạo và hiển thị mối liên kết giữa các bảng.

Ẩn, hiện đối tượng phụ thuộc hoặc thuộc tính của Sheet

Chạy Database Documenter hoặc phân tích hiệu suất

Di chuyển dữ liệu tới Microsoft SQL Server hoặc tới cơ sở dữ liệu khác.

Chạy Linked Table Manager.

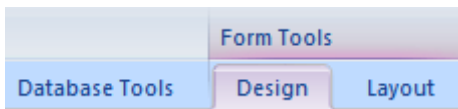
Manage Access add-ins.

Tạo vào sửa các Module Visual Basic for Applications (VBA).

**Bước 3:** Nhấp chuột chọn tab cần sử dụng hoặc nhấp phím ALT khi đó trên màn hình sẽ xuất hiện các chỉ dẫn, bạn muốn chọn nút nào thì nhấn phím tương ứng như hướng dẫn trên màn hình.

### ***Contextual command tabs***

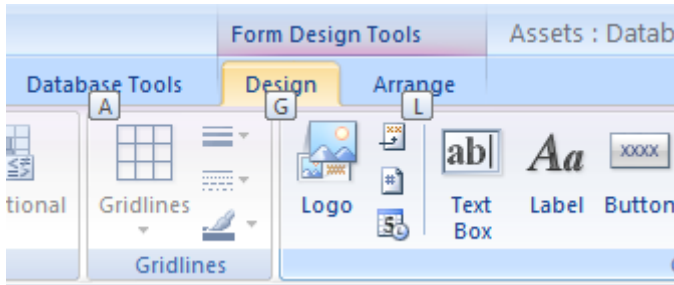
Ngoài những Tab lệnh chuẩn, Microsoft Access 2007 còn sử dụng một thành phần giao diện mới trong Office Professional 2007 được gọi là contextual command tab. Tùy thuộc vào bối cảnh của bạn (có nghĩa là, đối tượng mà bạn đang làm việc và những gì bạn đang làm), một hoặc nhiều Contextual command tabs có thể xuất hiện bên cạnh các Tab lệnh chuẩn.



Khởi động một contextual command tab

**Bước 1:** Nhấp phím Alt khi này các ký tự đại diện sẽ xuất hiện trên từng chức năng.

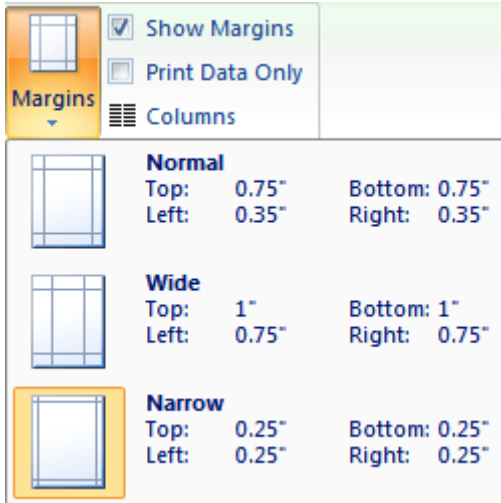
**Bước 2:** Nhấn phím xuất hiện trên các Contextual command tabs cần chọn.



Các contextual command tabs chứa lệnh và các tính năng mà bạn cần phải làm việc trong một bối cảnh cụ thể. Ví dụ, khi bạn mở một bảng trong thiết kế xem, các lệnh trên contextual command tab chỉ áp dụng khi bạn làm việc với bảng ở chế độ đó. Một ví dụ khác, khi bạn mở một bảng ở chế độ Design View, một contextual command tabs tên là Design xuất hiện tiếp theo là Tab Advanced Tools. Khi bạn nhấp chọn Tab Design, Ribbon hiển thị những lệnh có tác dụng chỉ trên những đối tượng ở chế độ Design View.

### ***Galleries***

Một sự đổi mới khác và là một phần quan trọng trong giao diện mới của Microsoft Access 2007 và các Control mới được gọi là gallery. Các gallery Control được thiết kế để làm việc với Ribbon và hướng sự chú ý của bạn tới những kết quả sẵn có. Thay vì chỉ hiển thị lệnh các gallery Control cho thấy kết quả của việc sử dụng các lệnh đó. Mục đích là để cung cấp cách nhìn trực quan giúp bạn có thể duyệt qua và xem những gì Microsoft Access 2007 có thể làm. Và như vậy bạn sẽ chỉ quan tâm tới kết quả thay vì phải suy nghĩ xem lệnh sẽ thực hiện những gì.

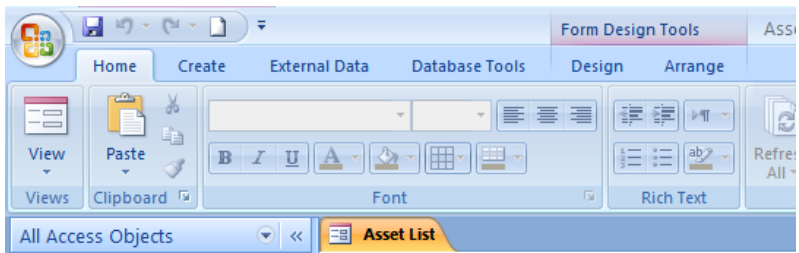


Galleries có hình dạng và kích cỡ khác nhau. Các kết quả được bố trí trên một bảng đồ xuống cho phép người dùng lựa chọn

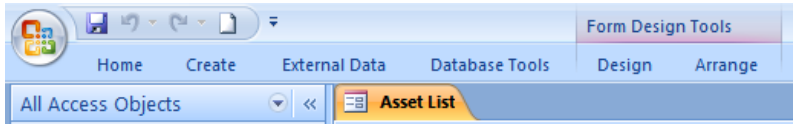
### ***Ẩn hiện các Ribbon***

Đôi khi, bạn có thể cần thêm một chút không gian, khoảng trống cho khu vực làm việc của bạn. Bạn có thể làm cho các Ribbon thu nhỏ lại trên màn hình chỉ còn các Bar và các Tab lệnh. Để làm được điều này bạn chỉ cần nhấp đúp chuột vào Ribbon, khi cần chúng hiện ra bạn hãy làm tương tự như vậy một lần nữa.

Ban đầu



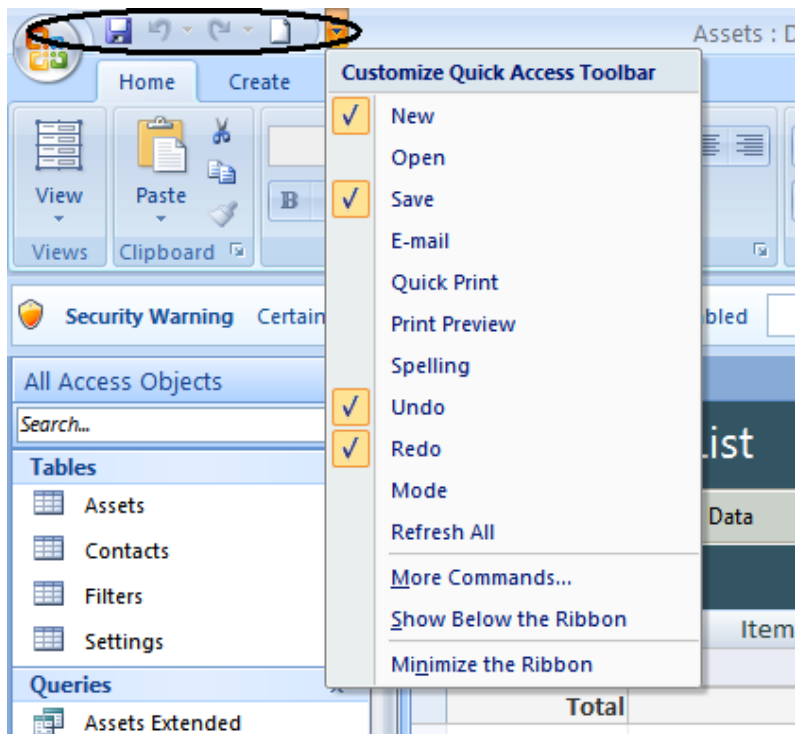
Sau khi làm ẩn các Ribbon



#### **1.1.4. Thanh công cụ Quick Access**

Là thanh công cụ nhỏ mặc định nằm kề bên Ribbon, bạn nhấp chuột trực tiếp vào lệnh cần sử dụng. Các lệnh mặc định ở thanh công cụ là những lệnh bạn thường sử dụng như Save, Undo, Redo. Tuy nhiên, bạn có thể tùy chỉnh thanh công cụ Quick Access, bạn cũng có thể bổ xung các lệnh khác lên đó. Bạn cũng có thể thay đổi vị trí, kích thước của thanh công cụ. Khi bạn thay đổi vị trí của thanh công cụ này xuống phía dưới của Ribbon kích thức nó sẽ được kéo rộng.



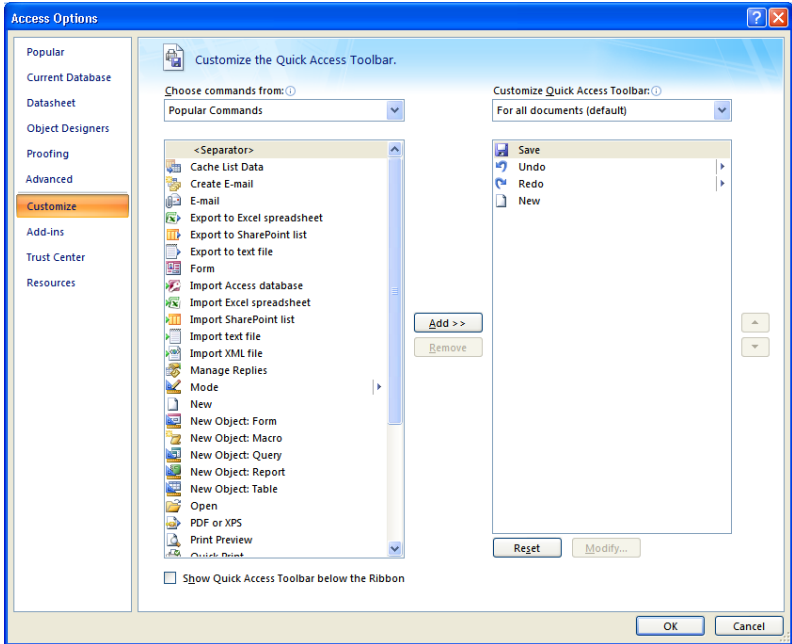


### ***Tùy biến thanh công cụ Quick Access***

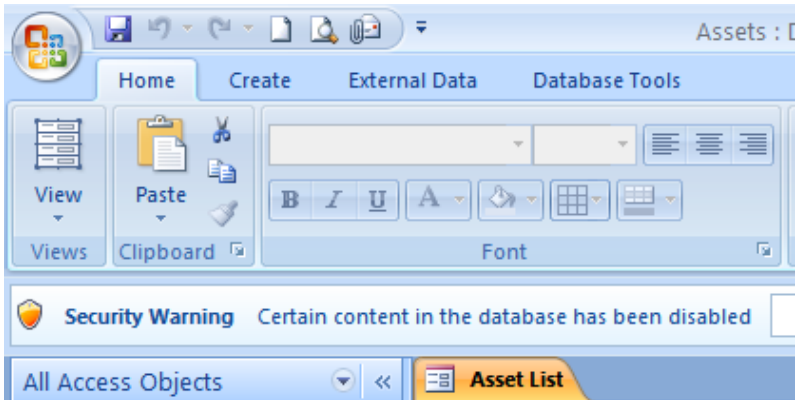
Nhấp chuột vào hộp thoại đổ xuống bên cạnh thanh công cụ

***Bước 1:*** Trong hộp thoại đổ xuống Customize Quick Access Toolbar nhấp chọn một số lệnh thường chú được liệt kê sẵn như Open, E-Mail, Quick Print, ... hoặc nhấp chọn More Commands, khi này hộp thoại Access Option xuất hiện.

***Bước 2:*** Trong hộp thoại này bạn chọn lệnh cần thêm vào ToolBar rồi nhấp Add, hoặc làm ngược lại để loại bớt lệnh trong ToolBar.



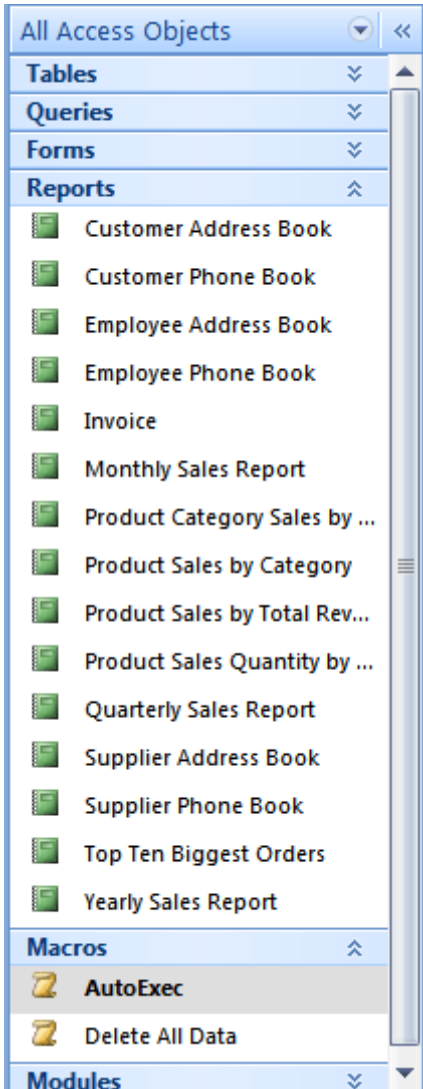
**Bước 3:** Cuối cùng nhấp Ok để hoàn tất.



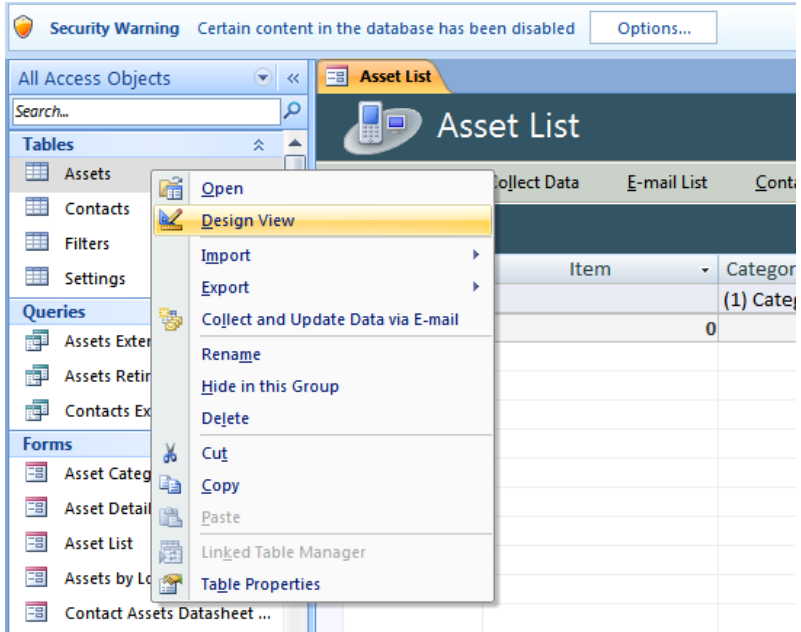
Sau khi thay đổi bạn sẽ có được ToolBar như ý muốn.

### 1.1.5. Navigation Pane

Khi bạn tạo mới hoặc mở một DataBase, tên các đối tượng trong DataBase của bạn sẽ nằm trong Navigation Pane. Các đối tượng này bao gồm các Table, Form, Report, Page, Macro, Module. Navigation Pane thay thế cho cửa DataBase được sử dụng trong các phiên bản trước của Access. Nếu như trước đây bạn sử dụng cửa sổ DataBase để làm việc thì bây giờ trong Microsoft Access 2007 bạn sẽ sử dụng Pane Navigation để làm những việc đó. Ví dụ nếu bạn muốn thêm một trong vào bảng ở chế độ DataSheet View, bạn mở bảng từ Navigation Pane.



Để mở hoặc thực thi một lệnh cho một đối tượng trên DataBase bạn nhấp phải chuột vào đối tượng đó, một Menu tương ứng sẽ xuất hiện bạn chọn lệnh cần thực hiện.



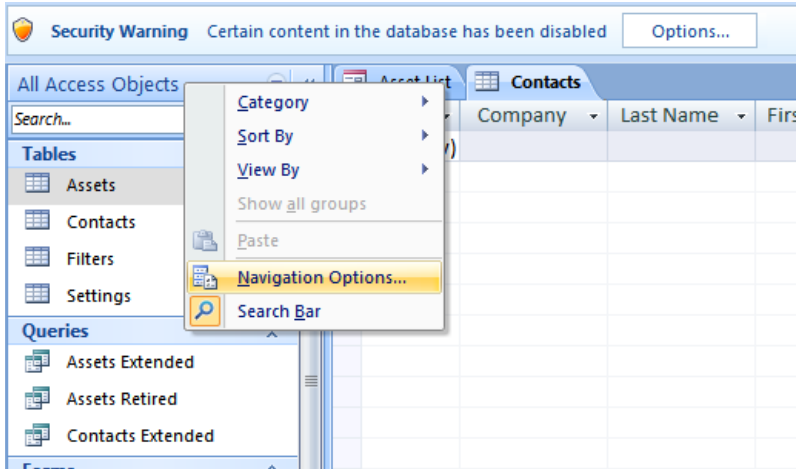
Nội dung của các Context Menu này tùy thuộc vào ngữ cảnh mà bạn đang làm việc.

Dưới đây là các cách mở một đối tượng trong DataBase ví dụ như Table, Form hoặc Report.

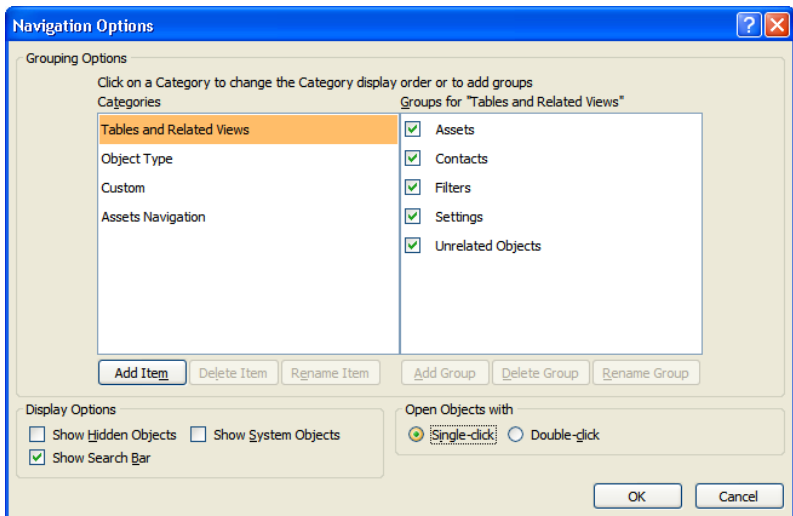
Cách 1: Từ Navigation Pane chọn đối tượng cần thao tác rồi nhấn phím Enter, hoặc nhấp đúp chuột.

Cách 2: Từ Navigation Pane chọn đối tượng cần thao tác rồi nhấp phải chuột, trong Menu đổ xuống chọn lệnh cần thực hiện.

Chú ý: Bạn có thể đặt tùy chọn mở một đối tượng chỉ bằng một động tác nhấp chuột trong cửa sổ Navigation Option.



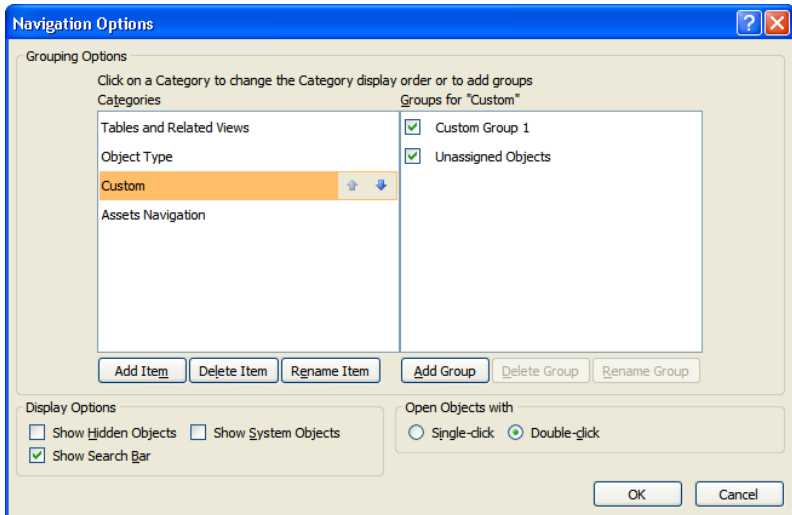
Để bật hộp thoại này bạn nhấp phải chuột lên phần chữ All Access Objects trên Navigation Option, chọn Navigation Options...



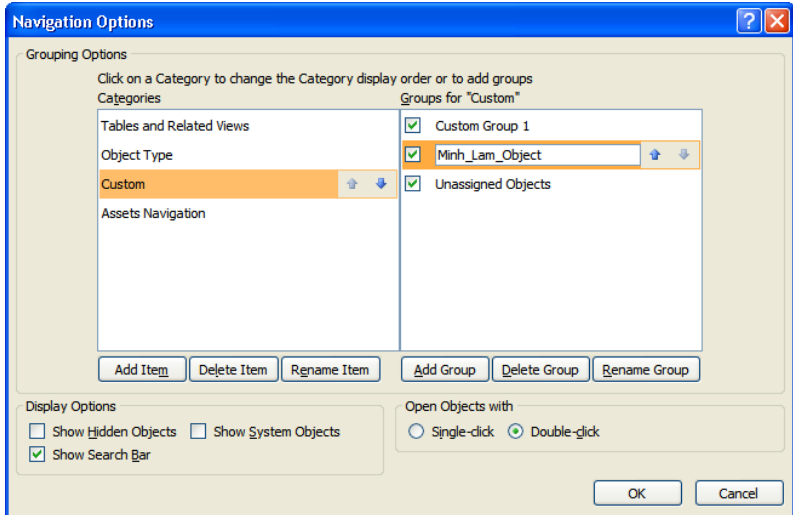
Trong mục Open Objects With bạn tích chọn Single-click. Cuối cùng nhấp Ok để hoàn tất.

Navigation Pane chia các đối tượng trong DataBase của bạn thành các mục khác nhau, và mỗi danh mục đó chứa một nhóm các đối tượng. Một số mục như bạn thấy khi lần đầu tiên làm việc với Pane này đã được tạo sẵn, bạn hoàn toàn có thể thêm vào các nhóm của riêng mình.

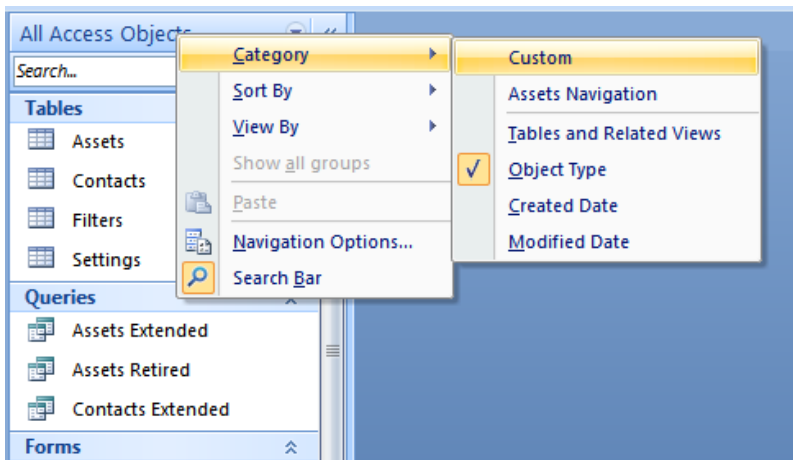
Đầu tiên bạn mở hộp thoại Navigation Option như ở phần trước.



Ở mục Categories bạn chọn Custom rồi nhấp Add Group sau đó nhấp vào tên của nhóm mới.

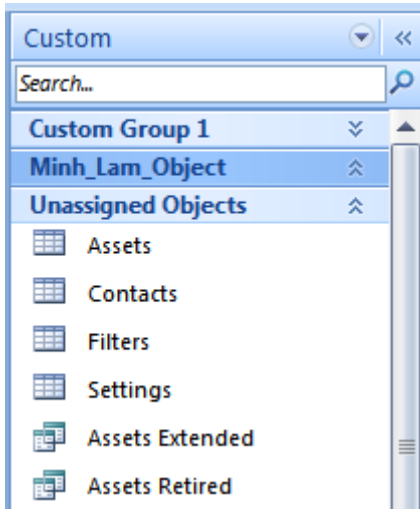


Nhấp Enter để xác nhận rồi nhấp Ok để hoàn tất, như vậy bạn đã có một nhóm mới trong Navigation Pane, bạn có thể chứa các đối tượng mới mà mình tạo ra. Vì mặc định Pane Navigation đang sắp xếp theo Objects Type nên bạn không thể thấy được nhóm mới mà mình vừa tạo ra, bạn cần phải sắp xếp lại theo Custom.





Sau khi chọn kiểu sắp xếp này Pane Navigation sẽ như hình dưới đây:



Mặc định Navigation Pane xuất hiện khi bạn mở một DataBase trong Microsoft Access 2007, cũng như các DataBase được tạo ở phiên bản trước. Tuy nhiên bạn có thể thiết lập lại chế độ cho Pane Navigation. Dưới đây là các bước thực hiện.

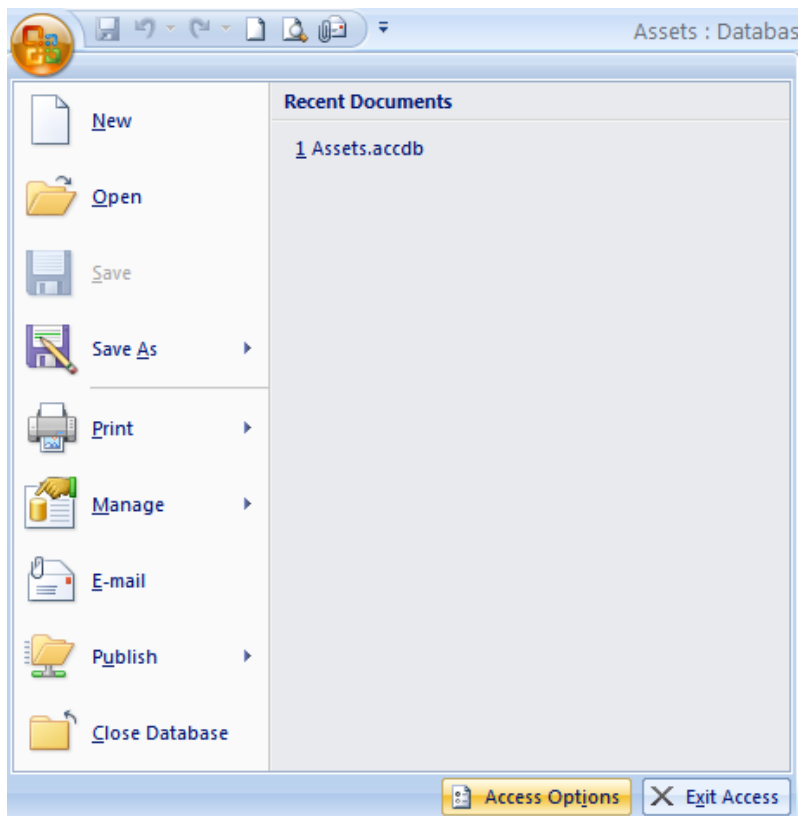
### ***Hiện thị hoặc ẩn Navigation Pane***

Nhấp nút phía trên bên góc phải của Pane Navigation hoặc nhấn phím F11.

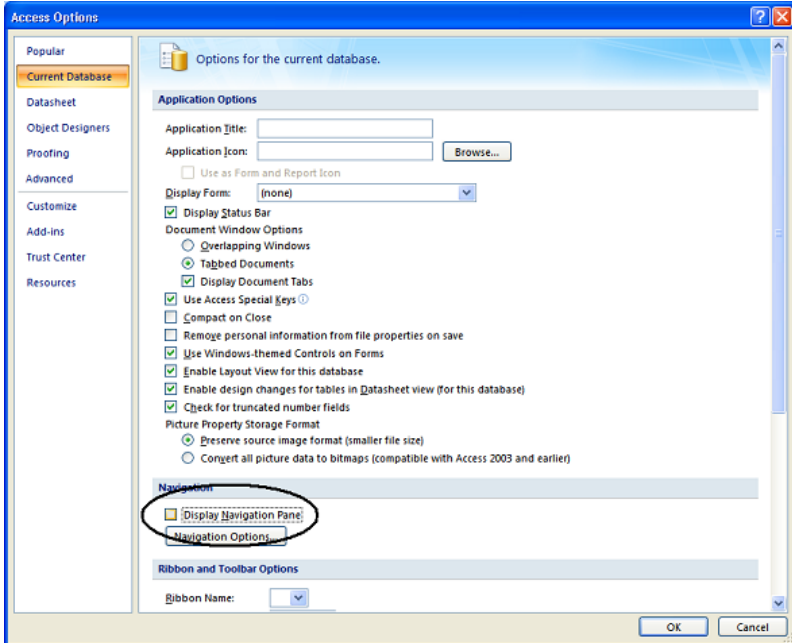


***Bước 1:*** Nhấp chuột vào nút Microsoft Office Button sau đó chọn Access Option.





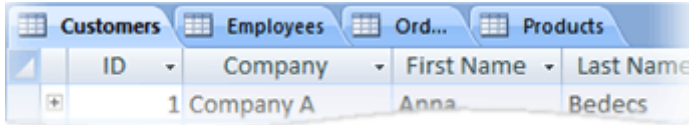
Bước 2: Bên phía trái Pane nhấp chọn Current DataBase.



**Bước 3:** Trong hộp thoại này phía dưới mục Navigation bạn bỏ tích mục Display Navigation Pane nếu không muốn Pane này xuất hiện khi mở DataBase và ngược lại nếu nó xuất hiện mặc định. Cuối cùng nhấp Ok để hoàn tất.

### 1.1.6. Tab document

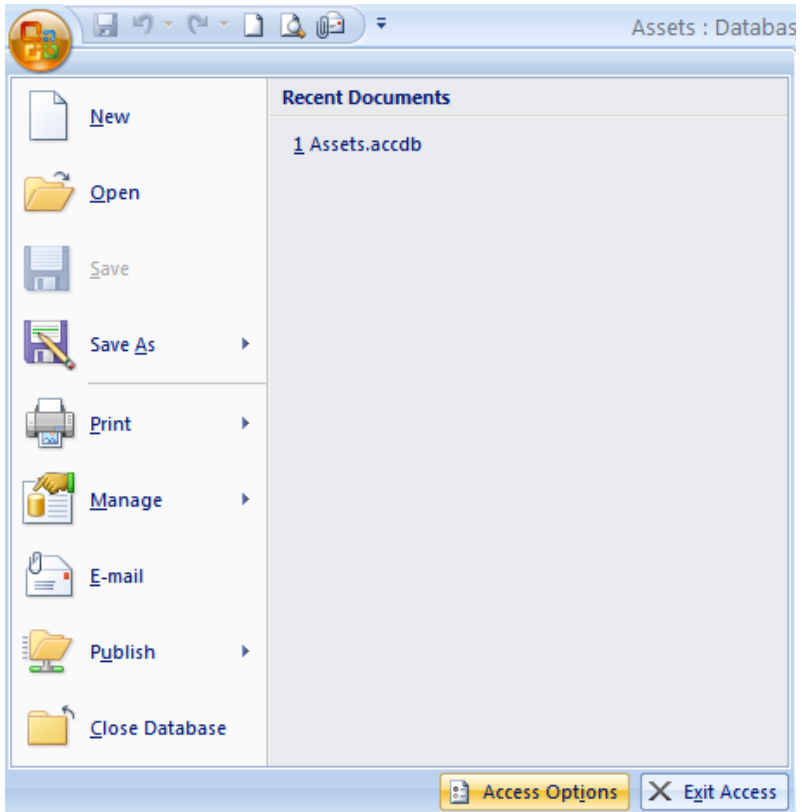
Bắt đầu với Microsoft Access 2007 bạn có thể hiển thị các đối tượng trong cơ sở dữ liệu của mình trên các Tab Document thay vì nằm chồng chéo trên các cửa sổ. Với Tab Document sẽ giúp bạn thuận lợi hơn khi phải làm việc thường xuyên với các đối tượng, tuy nhiên bạn có thể sử dụng hoặc không sử dụng giao diện Tab Document bằng cách thiết lập trong Access Option. Sau đó bạn phải đóng và mở lại DataBase để thiết lập mới có hiệu lực.



ID	Company	First Name	Last Name
1	Company A	Anna	Bedecs

### ***Ẩn hiện chế độ Tab Document.***

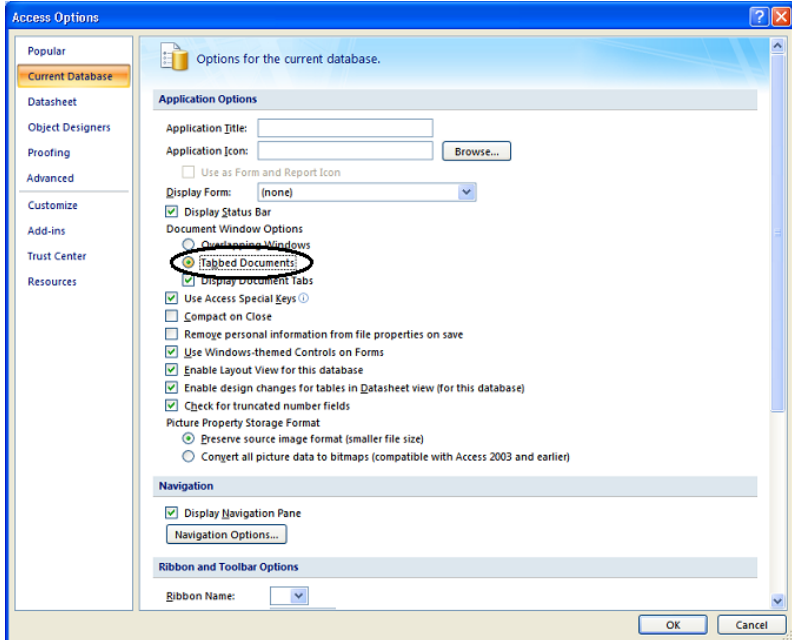
***Bước 1:*** Nhấp chọn nút Microsoft Office Button , sau đó chọn Access Options.



Hộp thoại Access Option xuất hiện.

***Bước 2:*** Bên phía trái Pane nhấp chọn Current DataBase.

**Bước 3:** Trong phần Application Options, dưới mục Document Window Options bạn tích hoặc bỏ tích chọn để tắt bật chế độ này.



**Bước 4:** Nhấp Ok để hoàn tất.

**Chú ý:** Thiết lập Display Document Tabs không phải là tùy chọn toàn cục nó chỉ có tác dụng trên DataBase mà bạn đang thao tác. Vì vậy nếu bạn muốn thiết lập cho các DataBase khác bạn phải thực hiện lại từ đầu.

Sau khi thay đổi thiết lập Display Document Tabs. Bạn cần phải đóng rồi mở lại DataBase của mình để thấy những thay đổi có tác dụng.

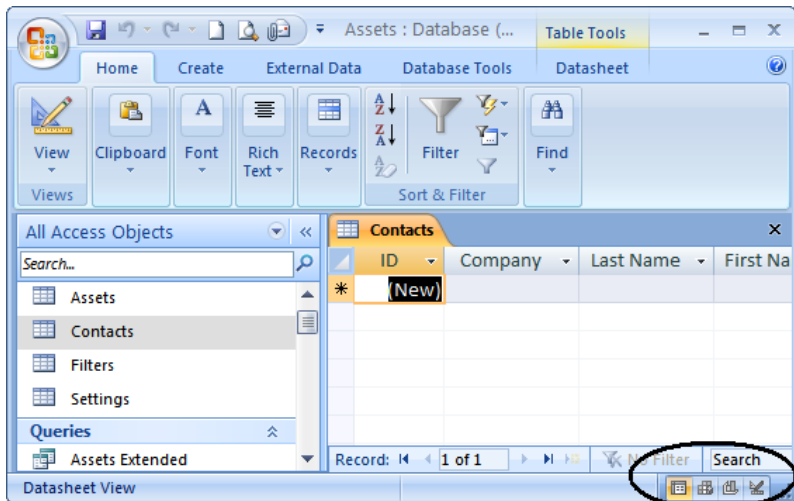
Những DataBase mới được tạo bởi Microsoft Access 2007 được gán mặc định chế độ Tab Document.

Những DataBase được tạo từ các phiên bản trước của Access sẽ được gắn mặc định chế độ các của sổ riêng rẽ.

### **1.1.7. Status bar**

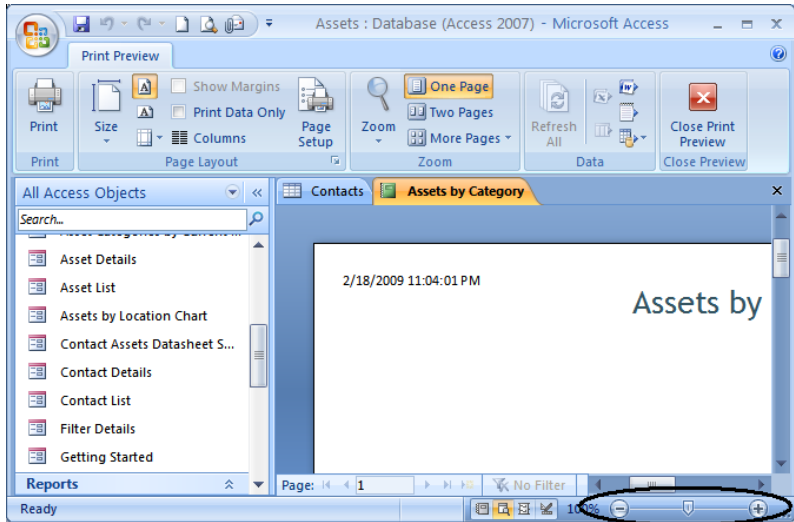
Cũng giống như những phiên bản trước của Access, với Microsoft Access 2007, bạn có thể hiển thị thanh trạng thái dưới đáy của cửa sổ hiện thời. Nó tiếp tục là một thành phần chuẩn trong ứng dụng giúp người dùng có thể thấy các thông báo trạng thái, thuộc tính, gợi ý, ... với Microsoft Access 2007, thanh trạng ngoài những chức năng kể trên nó còn đảm nhiệm hai chức năng chuẩn khác mà bạn có thể tìm thấy trong tất cả các ứng dụng của bộ Office 2007 là View (chọn chế độ hiển thị) và Zoom (phóng to, thu nhỏ).

Bạn có thể dễ dàng chuyển đổi giữa các chế độ hiển thị của sổ Window bằng cách sử dụng các điều khiển sẵn có trên thanh Status Bar.



Nếu bạn đang hiển thị một đối tượng có cung cấp khả năng Zoom, bạn có thể điều chỉnh các mức Zoom để phóng

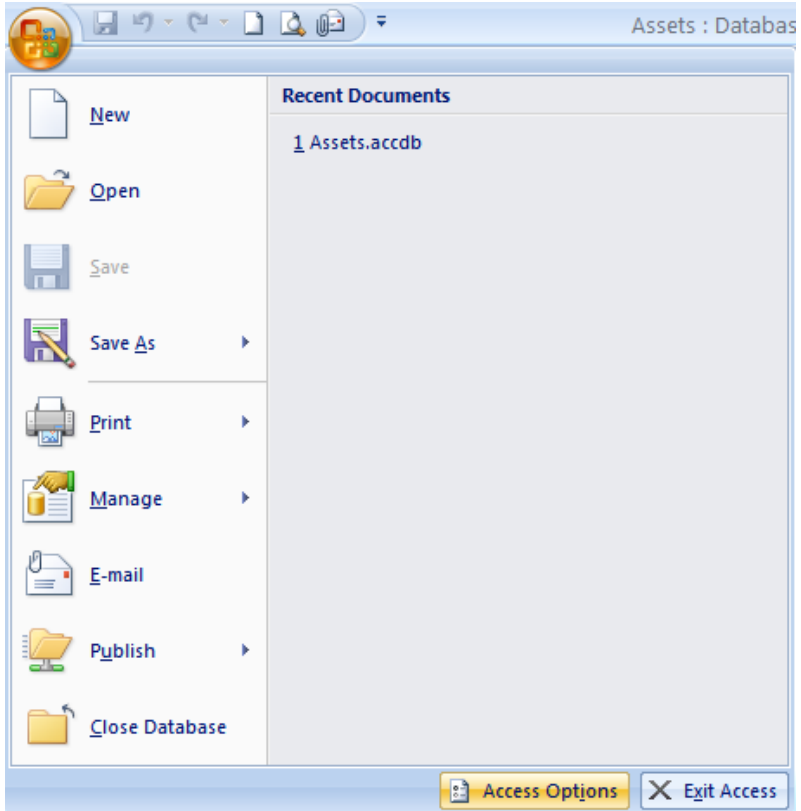
to hoặc thu nhỏ đối tượng, hay có thể sử dụng thanh trượt trên thanh trạng thái



Thanh trạng thái có thể tắt bật dễ dàng với tùy chỉnh trong hộp thoại Access Option.

Dưới đây là các bước hiển thị hoặc ẩn đi thanh trạng thái.

**Bước 1:** Nhấp chọn nút Microsoft Office Button , sau đó chọn Access Options.

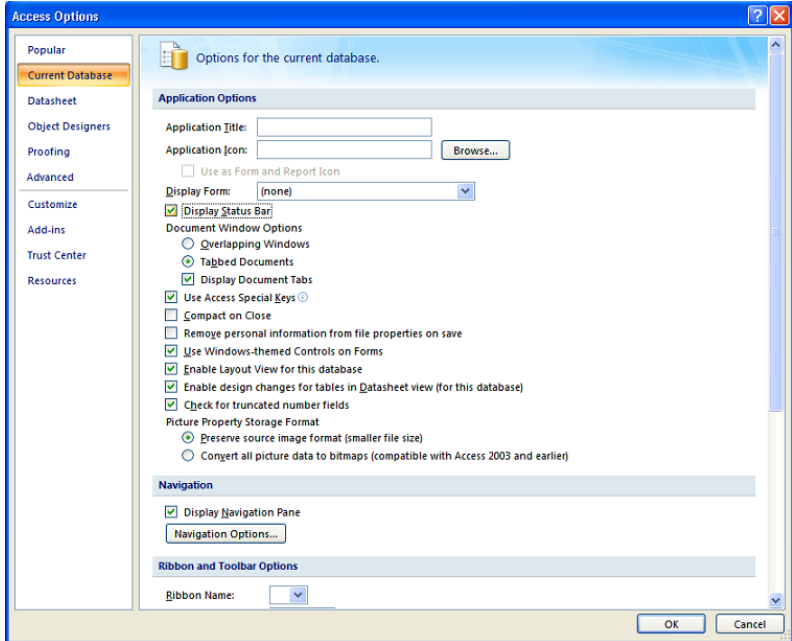


Hộp thoại Access Option xuất hiện.

***Bước 2:*** Bên phía trái Pane nhấp chọn Current DataBase.

***Bước 3:*** Dưới mục Application Options, chọn hoặc hủy chọn Display Status Bar để tắt hoặc bật Status Bar.

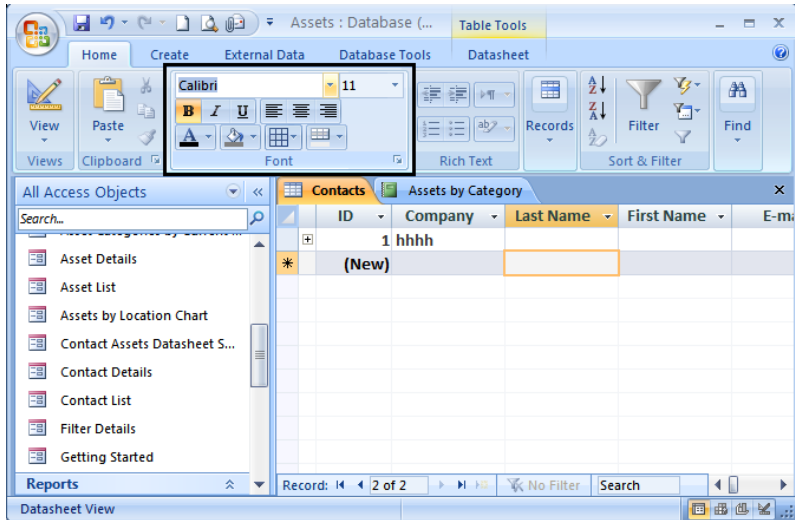




**Bước 4:** Cuối cùng nhấp Ok để hoàn tất.

### 1.1.8. Mini toolbar

Một trong những thao tác phổ biến trong các chương trình Office Professional là định dạng văn bản. Trong các phiên bản trước đây của Access, để định dạng văn bản chương trình luôn yêu cầu sử dụng một trình đơn hoặc một thanh công cụ hiển thị các định dạng. Với Microsoft Access 2007, bạn có thể định dạng văn bản một cách dễ dàng hơn bằng cách sử dụng Mini Toolbar. Khi bạn chọn cho các định dạng văn bản, các Mini Toolbar tự động xuất hiện trên văn bản được lựa chọn. Nếu bạn di chuyển con chuột trở đến gần các nút lệnh trên thanh Mini Toolbar chúng sẽ sáng lên và cho phép bạn có thể sử dụng nó để áp dụng các định dạng như Bold, Italic, Font Size, Color ... cho văn bản.



### 1.1.9. Tạo Database đầu tiên

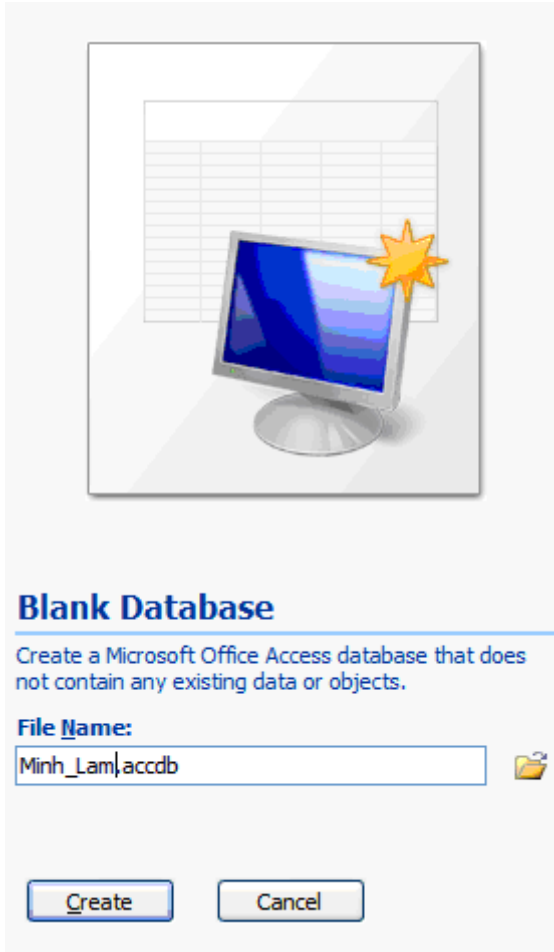
#### *Tạo một DataBase mới rỗng*

Bước 1: Bắt đầu với Microsoft Access 2007 từ Menu Start hoặc từ Shortcut. Trang Getting Started with Microsoft Office Access sẽ xuất hiện.

Bước 2: Trên trang Getting Started with Microsoft Office Access dưới mục New Blank DataBase nhấp chọn Blank DataBase.



Bước 3: Trên Pane Blank DataBase, trong hộp File Name gõ vào tên tệp tin .accdb



Bước 4: Nhấp Create.

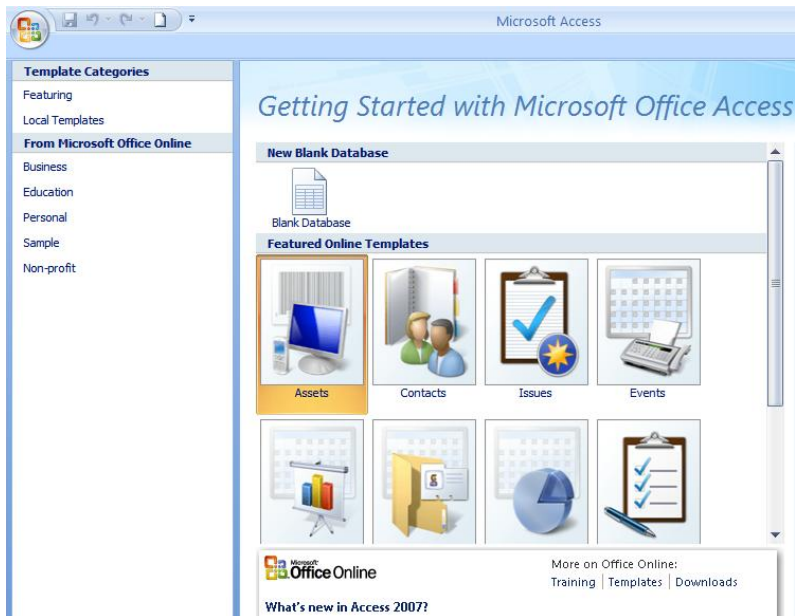
Sau 4 bước trên một DataBase với tên tệp tin là Minh\_Lam.accdb sẽ được tạo và một bảng mới được mở trong Datasheet view.

Microsoft Access 2007 cung cấp một số Template với một số sản phẩm, và bạn có thể tải về nhiều hơn từ Microsoft Office Online. Vậy Template là gì? Trong bối cảnh Microsoft

Office Access 2007, đó là một DataBase được thiết lập hoàn chỉnh từ trước với các Table, Form, và Report chuyên nghiệp. Các Template đem đến cho bạn một sự khởi đầu nhanh hơn khi bạn bắt đầu tạo lập một DataBase mới.

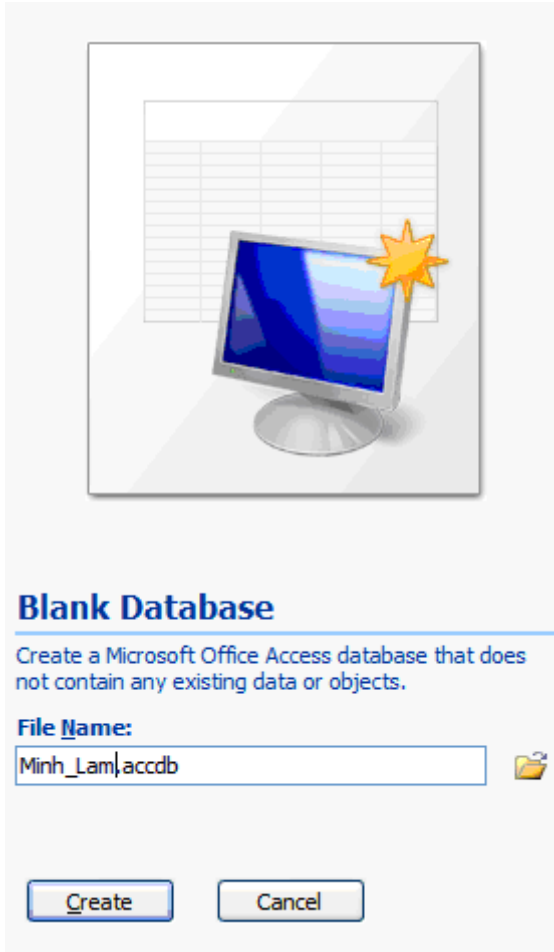
### ***Tạo DataBase mới từ Template***

***Bước 1:*** Khởi động Microsoft Access 2007 từ thanh Start hoặc từ Shortcut. Trang Getting Started with Microsoft Office Access sẽ xuất hiện như hình dưới đây.



***Bước 2:*** Trên trang Getting Started with Microsoft Office Access dưới mục Featured Online Templates, nhấp chọn một mẫu Template, giả sử ở đây chúng ta chọn Assets.

***Bước 3:*** Trong hộp File Name, gõ vào tên file với phần mở rộng là .accdb.



**Bước 4:** Tùy chọn, nhấp chọn “Create and link your database to a Windows SharePoint Services site” nếu bạn muốn liên kết tới trang web Windows SharePoint Services.

**Bước 5:** Nhấp chọn Create hoặc Download.

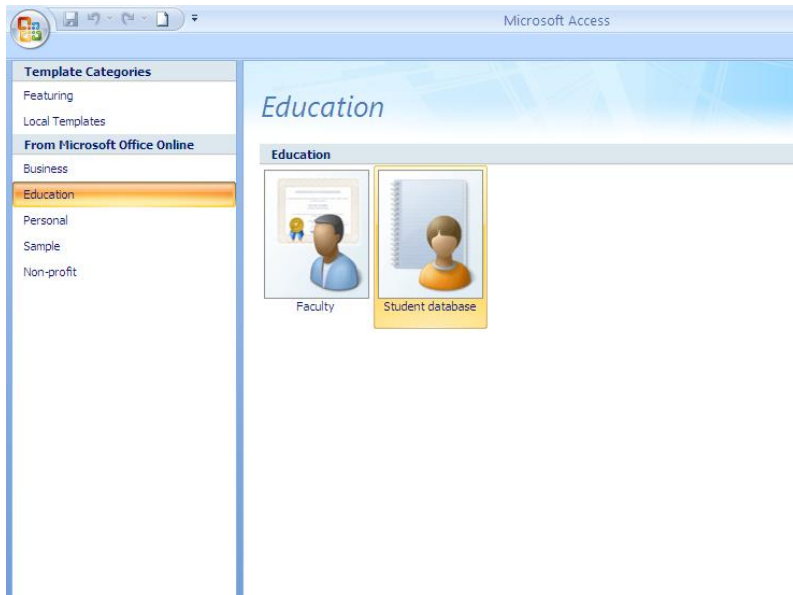
Access tạo một DataBase mới từ Template và mở nó.

Bạn có thể Download thêm các mẫu Template trực tiếp từ Office Online.

### ***Tạo DataBase từ Microsoft Office Online Template***

***Bước 1:*** Khởi động Access từ menu Start hoặc từ Shortcut. Trang Getting Started with Microsoft Office Access xuất hiện.

***Bước 2:*** Trên trang Getting Started with Microsoft Office Access, trong ô Template Categories bên phía trái, nhấp vào một loại và sau đó, khi bản mẫu xuất hiện trong danh mục, nhấp chọn một loại cần tạo.



***Bước 3:*** Trong mục File Name nhập vào tên File DataBase.

Navigation Pane

Student Details

Untitled

Go to [ ] Save and New E-mail Save As Outlook Contact

General Guardian Information Medical Information Attendance

First Name [ ]

Last Name [ ]

Web Page [ ]

E-mail Address [ ]

Special Circumstances [ ] + / -

Phone Numbers Note

Home Phone [ ]

Mobile Phone [ ]

Address Click to Map

Street [ ]

City [ ]

### Education

## Student database

<b>Applies to:</b> Access database templates, student information, student records<br>Create a students database to keep track of information about your students, including emergency contacts, medical information, and information about their guardians.

**Size:** 594 KB (2 mins @ 56 Kbps)

**Rating:** ★★★★★

**File Name:**

Student\_Minh\_Lam.accdb

Create and link your database to a Windows SharePoint Services site

**Download** **Cancel**

**Bước 4:** Check vào mục Create and link your database to a Windows SharePoint Services site nếu muốn kết nối DataBase của bạn tới Windows SharePoint Services.



***Bước 5:*** Cuối cùng nhấp Download để tải về máy.

Access tự động Download mẫu về máy tính của bạn và tạo một DataBase mới dựa trên Template, sau đó mở DataBase này.

Khi bạn ở (hoặc tạo và mở) một DataBase, Access thêm tên File và đường dẫn của DataBase đó vào danh sách văn bản được sử dụng. Đây là danh sách được hiển thị trên trang Getting Started with Microsoft Office Access nên bạn có thể dễ dàng mở lại những DataBase bạn đã sử dụng gần đây nhất một cách dễ dàng.

### ***Mở một DataBase sử dụng gần đây nhất***


***Bước 1:*** Khởi động Microsoft Access 2007

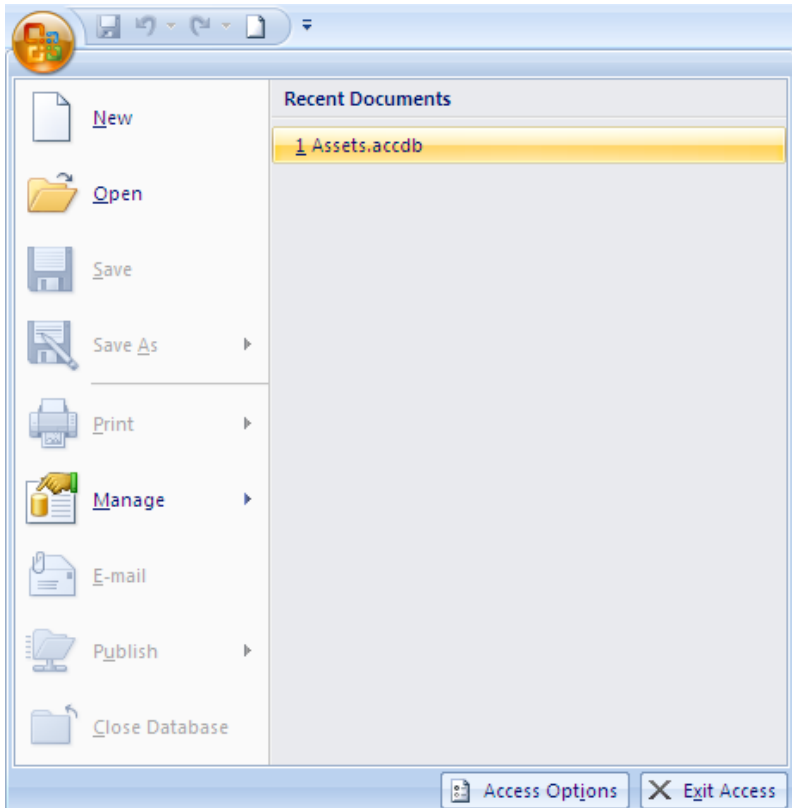
***Bước 2:*** Trên trang Getting Started with Microsoft Office Access phía dưới mục Open Recent DataBase nhấp chọn DataBase mà bạn muốn mở.



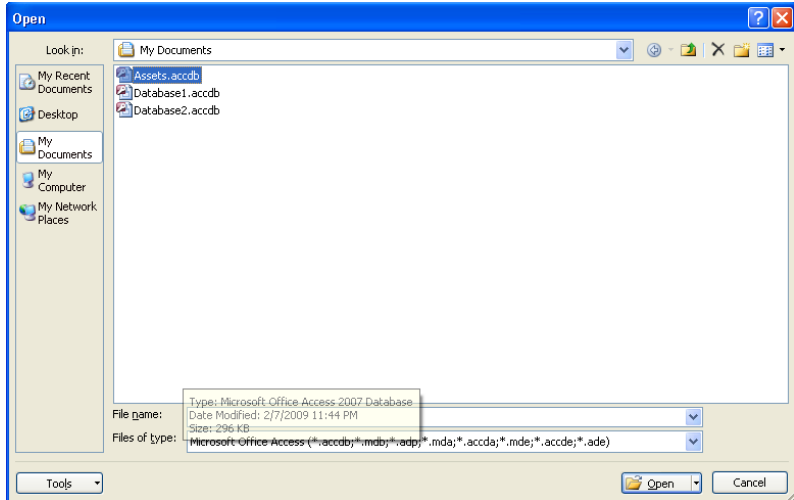
## Mở một DataBase sử dụng nút Microsoft Office Access

**Bước 1:** Khởi động Access

**Bước 2:** Nhấp chọn Microsoft Office Button , sau đó nhấp chọn DataBase mà bạn cần mở, nếu như nó xuất hiện ở Pane bên phải.



Hoặc nhấp chọn Open, hộp thoại xuất hiện chọn đến DataBase cần mở rồi nhấp Open.



Như vậy bạn đã mở được một cơ sở dữ liệu có sẵn.

## **1.2. Bắt đầu làm việc với cơ sở dữ liệu**

### **1.2.1. Lần đầu tiên sử dụng Access**

Office Access 2007 với những chức năng mới đã làm cho quá trình tạo ra một cơ sở dữ liệu dễ dàng hơn bao giờ hết. Những người dùng đã tạo ra những cơ sở dữ liệu trong Access ở các phiên bản trước sẽ đánh giá rất cao những cải tiến mới này và nhờ chúng mà quá trình tạo ra các đối tượng được tăng tốc đáng kể.

Khi bạn bắt đầu với Microsoft Access 2007, màn hình đầu tiên xuất hiện là trang Getting Started with Microsoft Office Access (trừ trường hợp bạn khởi động Access bằng cách nhấp đúp chuột vào một file Database Access cụ thể nào đó, khi đó Access sẽ khởi động tùy vào từng trường hợp cụ thể).

Trang Getting Started with Microsoft Office Access là điểm khởi đầu để giúp bạn có thể tạo ra một DataBase mới, mở một DataBase đã tồn tại, hoặc hiển thị nội dung đặc trưng từ Microsoft Office Online.

### **1.2.2. Tạo một DataBase sử dụng Template.**

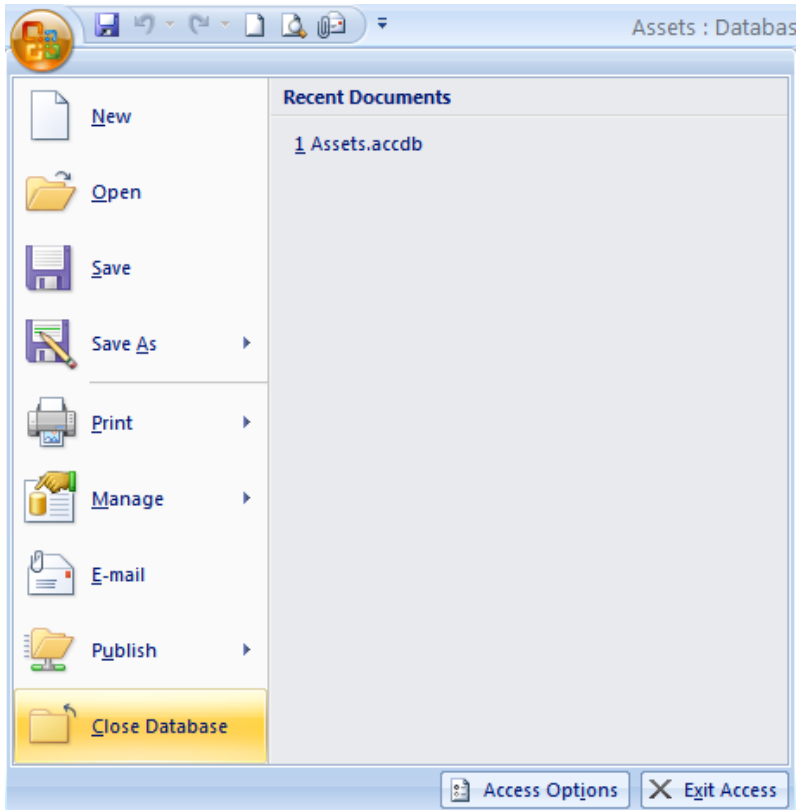
Access cung cấp cho bạn rất nhiều mẫu Template mà bạn có thể sử dụng để đẩy nhanh quá trình tạo ra những những DataBase của riêng mình. Một Template là một DataBase đã sẵn sàng sử dụng mà nội dung của nó bao gồm các Table, Query, Form, Report cần cho một công việc cụ thể.

Ví dụ, có những mẫu Template mà bạn có thể sử dụng để quản lý danh bạ, hoặc theo dõi chi phí, ... Các Database Template có thể sử dụng với những gì nó đã có hoặc bạn có thể tùy biến cho chúng phù hợp hơn với nhu cầu mà bạn cần.

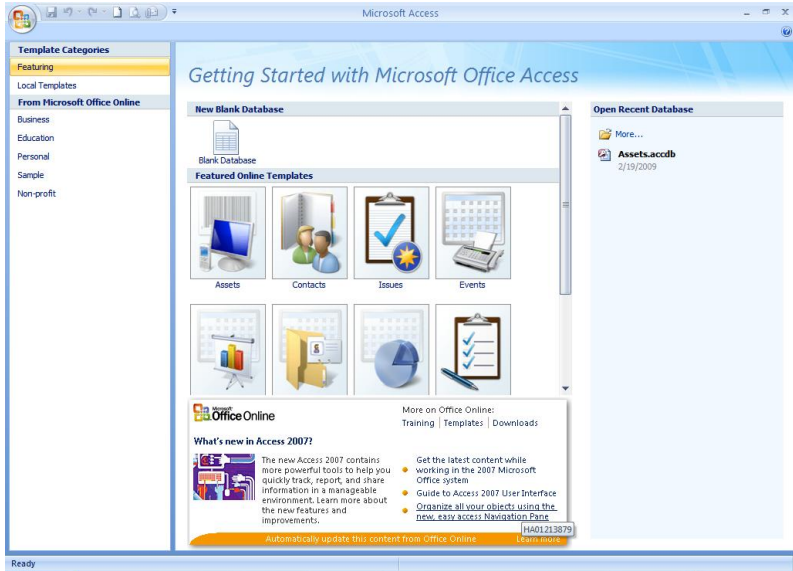
Bước 1: Khởi động Access nếu bạn chưa mở. Trong trường hợp bạn đang mở một DataBase bạn nhấp chuột vào nút

Microsoft Office Button  và chọn Close Database  để

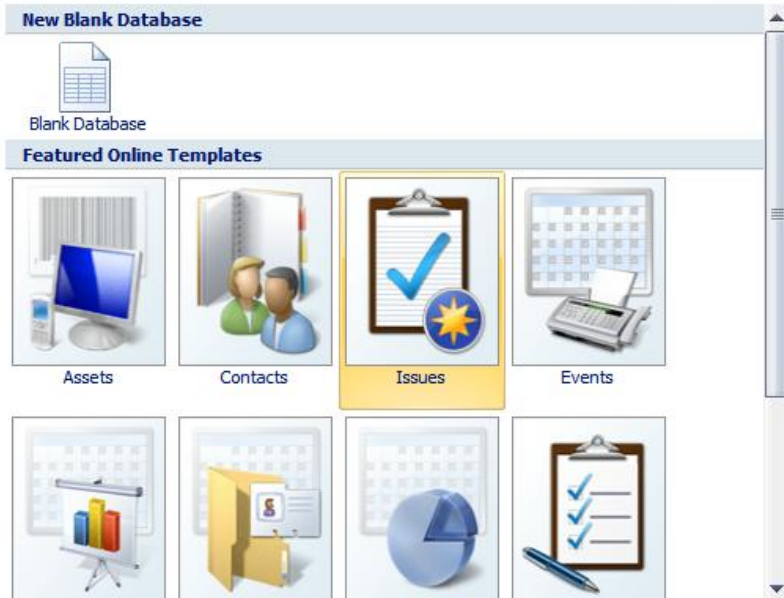
có thể hiển thị trang Getting Started with Microsoft Office Access.



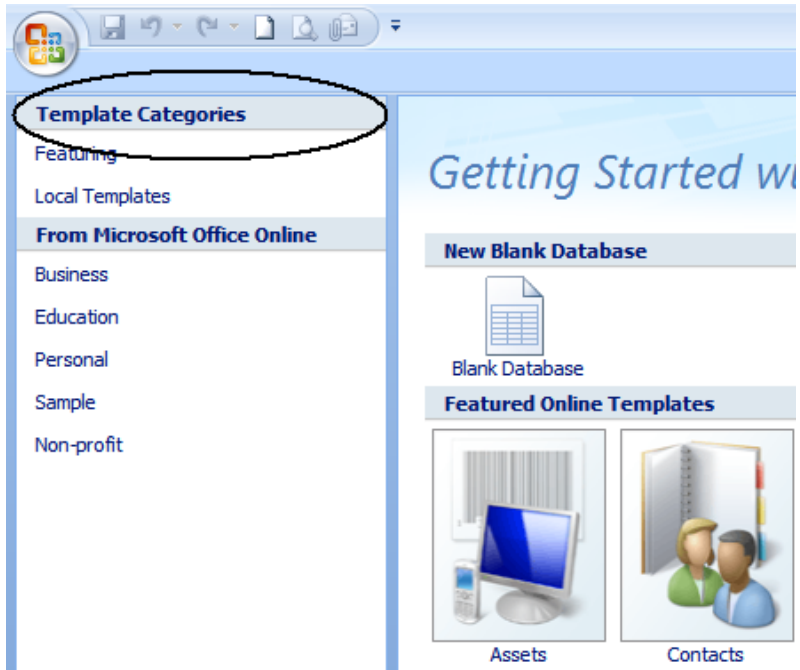
Màn hình chính của Microsoft Access 2007 và trang Getting Started with Microsoft Office Access sẽ hiển thị như hình dưới đây.



**Bước 2:** Một vài Template được hiển thị phía dưới mục Featured Online Templates trên trang Getting Started with Microsoft Office Access



Để có nhiều hơn bạn nhấp chuột vào Template Categories ở bên trái của cửa sổ Access.



Ngoài ra bạn có thể tải về từ Website của Microsoft Office.



Assets      Contacts      Issues      Events

Microsoft Office Online

**What's new in Access 2007?**

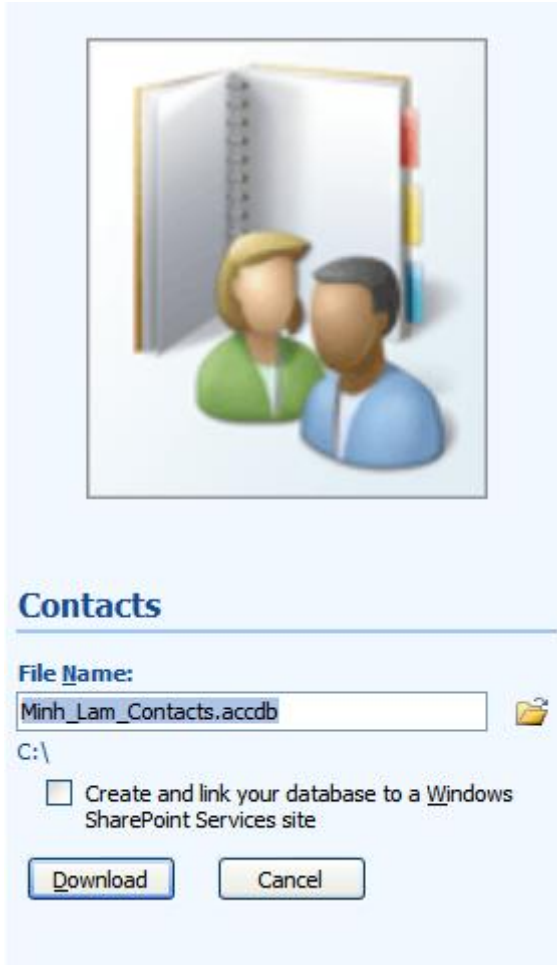
The new Access 2007 contains more powerful tools to help you quickly track, report, and share information in a manageable environment. Learn more about the new features and improvements.


- [Get the latest content while working in the 2007 Microsoft Office system](#)
- [Guide to Access 2007 User Interface](#)
- [Organize all your objects using the new, easy access Navigation Pane](#)

Automatically update this content from Office Online      Learn more

**Bước 3:** Nhấp chuột vào Template mà bạn muốn sử dụng.

**Bước 4:** Trong Pane bên phải cửa sổ Access, Access yêu cầu bạn nhập vào tên cho DataBase trong hộp File Name.

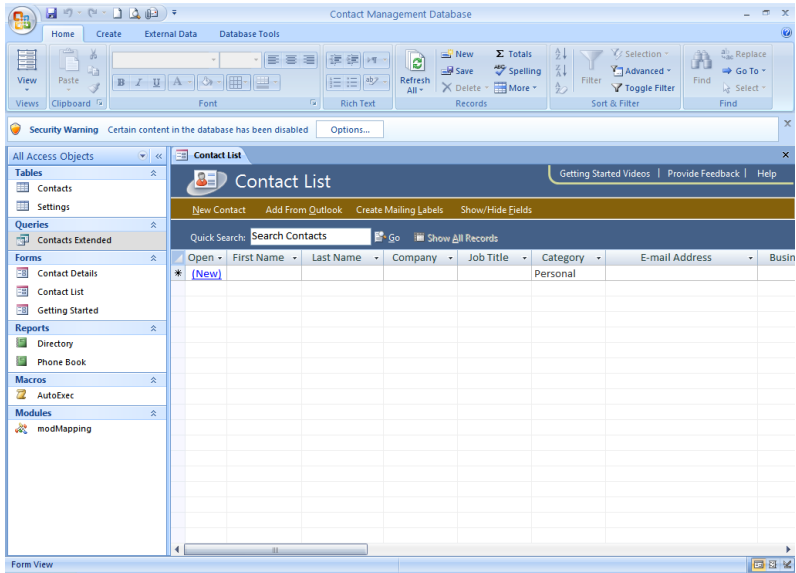


Bạn cũng có thể sửa lại đường dẫn mặc định thành đường dẫn khác bằng cách nhấp vào nút  (Browse for a location ...).

Tùy chọn Create your database and link it to a Microsoft Windows SharePoint Services site cho phép bạn liên kết trực tiếp DataBase tới Website của Microsoft Windows SharePoint Services.

**Bước 5:** Nhấp Create (hoặc Download, cho trường hợp bạn tải các Template online) để hoàn tất

Với các bước trên Access sẽ tạo và mở DataBase. Một Form được mở và bạn đã có thể bắt đầu nhập liệu cho DataBase.



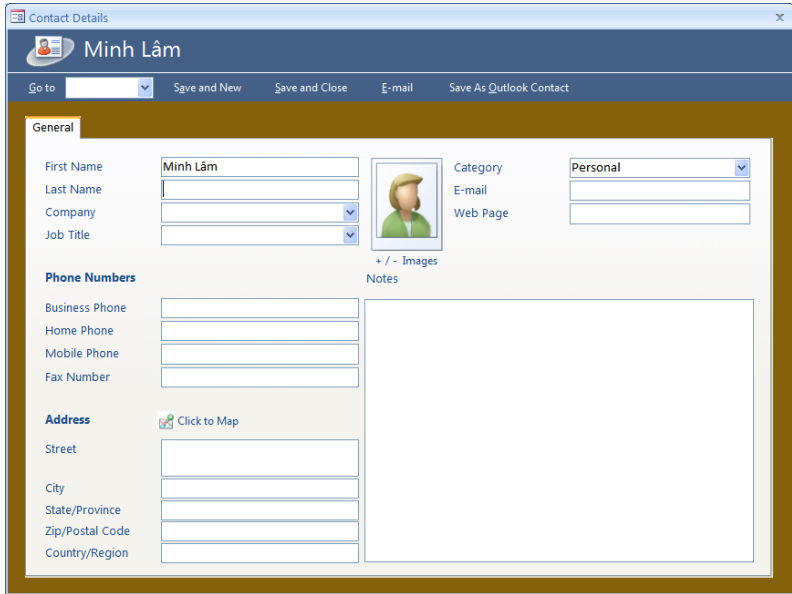
**Chú ý:** Nếu nội dung Template có dữ liệu mẫu bạn có thể xóa chúng đi theo các bước dưới đây:

- Nhấp chuột vào dòng đầu tiên của bản ghi mà bạn muốn xóa.
- Trong Tab Home tìm tới phần Record nhấp chọn Delete.



**Bước 6:** Để bắt đầu nhập dữ liệu, nhấp chuột vào dòng trắng đầu tiên trên Form và sau đó có thể bắt đầu nhập dữ liệu. Sau

khi đã nhập vào một vào bản ghi bạn có thể sử dụng Pane Navigation để xem liệu có một Form hoặc một Report khác mà bạn muốn sử dụng.



The screenshot shows a Microsoft Access window titled 'Contact Details' for a contact named 'Minh Lâm'. The window has a menu bar with options: 'Go to', 'Save and New', 'Save and Close', 'E-mail', and 'Save As Outlook Contact'. The 'General' tab is active, displaying a form with the following fields:

- First Name:** Minh Lâm
- Last Name:** (empty)
- Company:** (empty)
- Job Title:** (empty)
- Category:** Personal
- E-mail:** (empty)
- Web Page:** (empty)

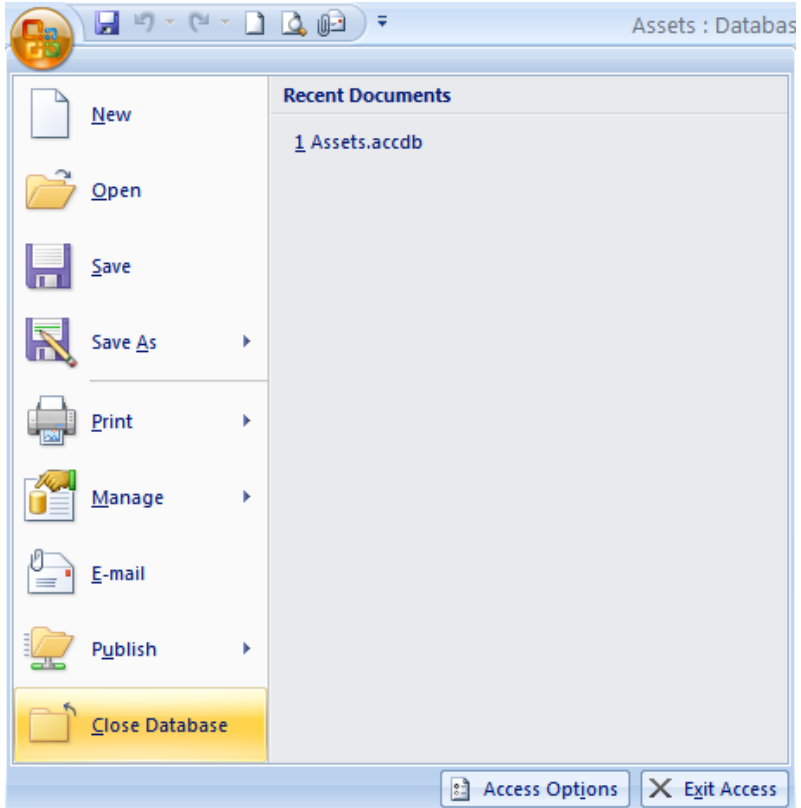
Below these fields are sections for 'Phone Numbers' (Business, Home, Mobile, Fax) and 'Address' (Street, City, State/Province, Zip/Postal Code, Country/Region). There is also a 'Notes' field and a 'Click to Map' button. A small profile picture of a man is visible next to the name.

### 1.2.3. Tải các Template từ Microsoft Office Online

Nếu bạn không thể tìm thấy một mẫu phù hợp với yêu cầu của mình trên trang Getting Started with Microsoft Office Access, bạn có thể khám phá trên Web Office Online ở đó sẽ có rất nhiều lựa chọn cho bạn.

**Bước 1:** Khởi động Access nếu bạn chưa mở. Trong trường hợp bạn đang mở một DataBase bạn nhấp chuột vào nút

Microsoft Office Button  và chọn Close Database  để có thể hiển thị trang Getting Started with Microsoft Office Access.



***Bước 2:*** Gần phía cuối của trang Getting Started with Microsoft Office Access và dưới phần More on Office Online bạn nhập chọn Templates.

The screenshot shows the Microsoft Office Online homepage. At the top, there are four icons representing different Office applications: a presentation, a folder, a pie chart, and a clipboard. Below these is the Microsoft Office Online logo. To the right, there are navigation links for 'Training', 'Templates', and 'Downloads', with 'Templates' circled in red. The main content area is titled 'What's new in Access 2007?' and features a purple graphic of a person at a computer. The text describes the new features of Access 2007, such as improved tracking, reporting, and sharing capabilities. A list of bullet points highlights key features: working in the 2007 Microsoft Office system, the new user interface, and the easy access navigation pane. At the bottom, there is an orange banner that says 'Automatically update this content from Office Online' and a 'Learn more' link.

Website Office Online sẽ xuất hiện hiện trong một trình duyệt mới.

This screenshot shows the Microsoft Office Online website as it appears in a Microsoft Internet Explorer browser window. The browser's address bar shows the URL 'http://office.microsoft.com'. The website's header includes the Microsoft Office Online logo and a sign-in prompt. The main navigation menu includes 'Home', 'Products', 'Help and How-to', 'Downloads', 'Clip Art', 'Templates', and 'Microsoft Office Live'. The 'Templates' section is active, displaying a search bar and several featured templates, including 'Basic calendar (Word)', 'One-page calendar (Excel)', and 'All 2009 calendars'. There are also promotional banners for 'New Templates' and 'Have a heart' (Valentine's Day templates). The browser's status bar at the bottom shows 'Internet'.

Bước 3: Sử dụng các công cụ tìm kiếm trên trang web của Office Online để tìm và tải về các mẫu mà bạn muốn.

Bước 4: Nếu bạn tải về một mẫu mới, DataBase đó sẽ được lưu trữ ở một trong số các thư mục sau đây:

- Với Windows Vista: c:\Users\user name\Documents
- Với Microsoft Windows Server 2003 hoặc Microsoft Windows XP: c:\Documents and Settings\user name\My Documents.

Sau đó khi cần làm việc với các cơ sở dữ liệu này, bạn có thể sử dụng Access hoặc Windows Explorer để mở nó ra ở đường dẫn đã lưu trước đó.

#### **1.2.4. Tạo DataBase từ đầu**

Nếu không có một Template nào phù hợp với yêu cầu của bạn, hoặc bạn có dữ liệu trong một chương trình khác mà bạn muốn Import vào Access, bạn hãy chắc chắn rằng nó là tốt hơn là việc ra một cơ sở dữ liệu từ đầu. Đối với hầu hết các ứng dụng, điều này thường liên quan đến một trong hai vấn đề sau đây:

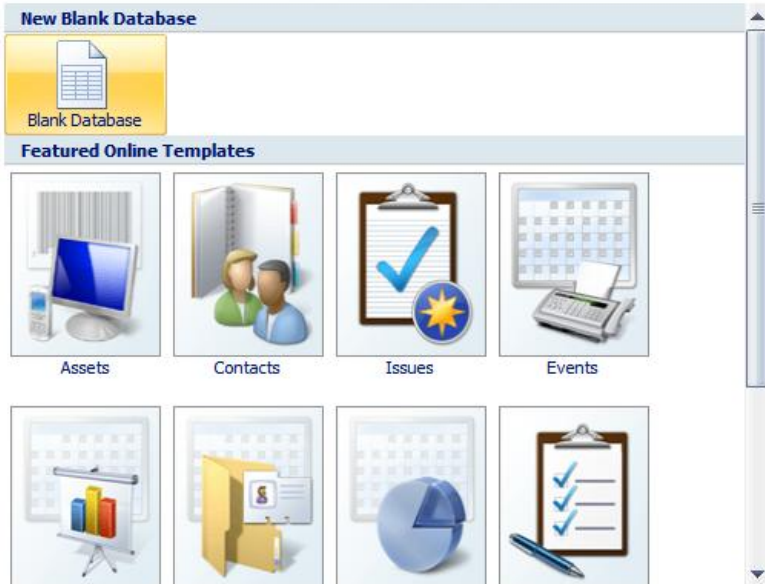
- Tạo các bảng mới, sau đó nhập dữ liệu, và dán dữ liệu hoặc Import dữ liệu vào các bảng đó.
- Import dữ liệu từ một nguồn khác và các bảng sẽ được tạo ra trong quá trình đó.

#### **1.2.5. Tạo một DataBase rỗng**

Bước 1: Khởi động Access

Bước 2: Trên trang Getting Started with Microsoft Office Access nhấp chọn Blank Database.

## Getting Started with Microsoft Office Access




**Bước 3:** Trong Pane Blank DataBase hãy nhập vào tên cho DataBase trong mục File Name, nếu bạn không điền phần mở rộng cho File thì Access sẽ tự động điền thêm.





Mặc định chương trình sẽ lưu File tại đường dẫn:

- Windows Vista: c:\Users\*user name*\Documents
- Microsoft Windows Server 2003 hoặc Microsoft Windows XP: c:\Documents and Settings\*user name*\My Documents

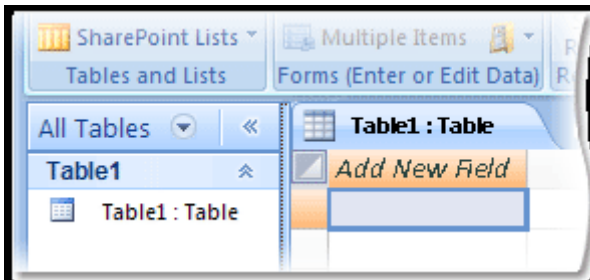
Để thay đổi đường dẫn chứa tập tin, nhấp chọn nút Browse  bên cạnh vào ô File Name, một hộp thoại sẽ xuất hiện

cho phép bạn chọn đường dẫn mới, nhấp Ok sau khi đã chọn xong.

**Bước 4:** Nhấp Create để hoàn tất.

Sau các bước trên Access sẽ tạo ra một DataBase mới và mở sẵn nó với một bảng trống ở chế độ DataSheet View cho bạn thiết kế.

**Bước 5:** Vị trí con trỏ chuột của Access đặt tới ô trống đầu tiên trên cột Add New Field.




Để thêm dữ liệu bạn có thể nhập vào hoặc Copy dữ liệu từ những nguồn khác.

**Chú ý:** Việc nhập dữ liệu ở chế độ Datasheet View được thiết kế tương tự như nhập dữ liệu trong Worksheet Excel. Những hạn chế chủ yếu là dữ liệu nhập vào phải được nhập ở các dòng, cột liền nhau, bắt đầu ở góc trên bên trái của Datasheet. Bạn không nên cố gắng để định dạng dữ liệu nhập vào có các hàng hoặc cột trống, bởi vì làm như vậy sẽ gây lãng phí bộ nhớ trong các bảng.

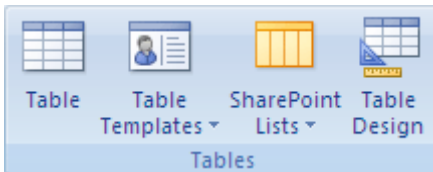
Cấu trúc của bảng dữ liệu được tạo trong khi bạn nhập dữ liệu vào nó. Tại một thời điểm bất kỳ bạn hoàn toàn có thể thêm một cột mới vào bảng, khi đó một trường mới sẽ được định nghĩa. Access thiết lập kiểu dữ liệu cho trường mới dựa vào kiểu của dữ liệu mà bạn nhập vào. Ví dụ, nếu bạn có một cột mà trong đó bạn chỉ nhập giá trị ngày tháng, Access sẽ

thiết lập kiểu dữ liệu của trường đó là Data/Time. Sau này giả sử bạn cố gắng nhập vào trường này một giá trị không phải ngày tháng (chẳng hạn như là tên hoặc một số điện thoại nào đó), Access sẽ hiển thị thông báo rằng giá trị của bạn không hợp lệ. Vì vậy, khi có thể, bạn nên phân tích lại bảng dữ liệu của mình sao cho mỗi trường chỉ chứa cùng một loại dữ liệu, cho dù đó là văn bản, ngày tháng, số điện thoại, .... Điều này làm cho việc xây dựng các Query, Form, Report (lấy dữ liệu từ những bảng này) sau này sẽ trở nên dễ dàng và thuận tiện hơn rất nhiều.

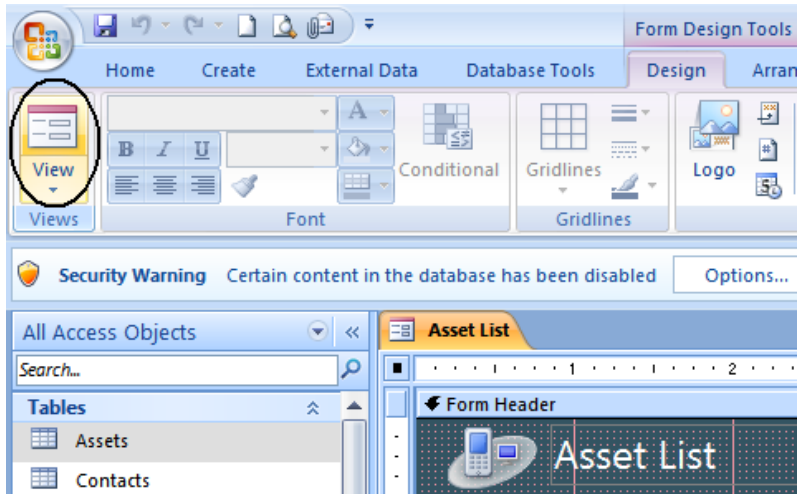
Nếu bạn không quan tâm đến việc nhập dữ liệu vào thời điểm này, hãy nhấp vào nút Close .

### **1.2.6. Thêm một bảng vào Database**

Bạn có thể thêm một bảng mới vào Database đã tồn tại bằng cách sử dụng công cụ trong nhóm Tables trên Tab Create.




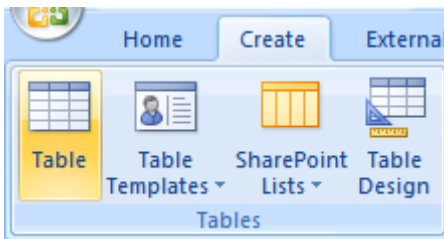
Nhấp chuột vào nút Table để tạo một bảng trống mới ở chế độ Datasheet View. Sau đó bạn hoàn toàn có thể nhập ngay dữ liệu vào bảng và Access sẽ tạo ra cấu trúc bảng cho bạn khi bạn thoát ra, hoặc bạn có thể sử dụng Design View để tạo ra cấu trúc bảng trước sau đó chuyển về chế độ Datasheet View để nhập dữ liệu. Bất kể bạn bắt đầu làm việc với bảng ở chế độ nào bạn đều có thể chuyển sang những chế độ hiển thị khác bằng cách sử dụng nút View trên thanh trạng thái.



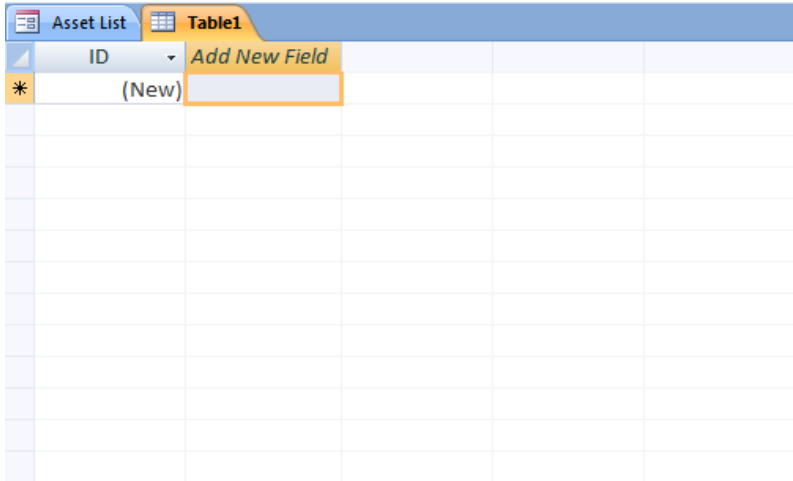
Thêm một bảng, bắt đầu ở chế độ Datasheet view :

Trong Datasheet View, bạn có thể nhập ngay dữ liệu, và Access sẽ xây dựng cấu trúc cho bảng dựa trên những dữ liệu đó. Tên trường được đặt lần lượt (Field1, Field2, ...), và các kiểu dữ liệu được Access tự động thiết lập tùy thuộc vào loại dữ liệu mà bạn nhập vào các trường.


**Bước 1:** Trên Tab Create trong nhóm Tables nhấp chọn Table .

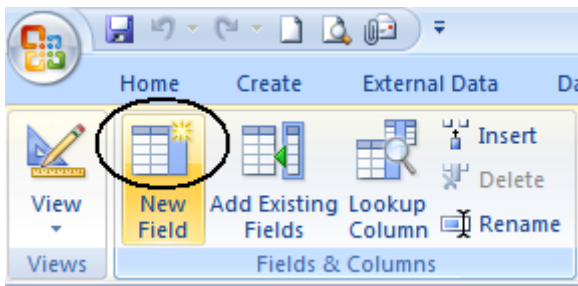


**Bước 2:** Sau bước này Access sẽ tạo ra một bảng và đặt vị trí con trỏ tại ô trống đầu tiên của cột Add New Field.

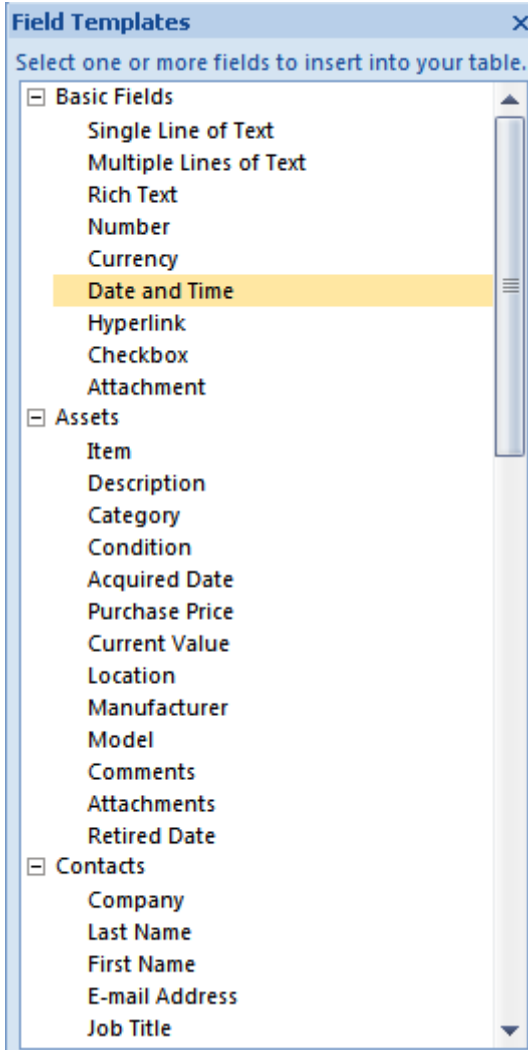


**Chú ý:** Nếu bạn không nhìn thấy cột có tên là Add New Field, có thể bạn vẫn đang ở chế độ Design View chứ không phải Datasheet View. Để chuyển qua chế độ hiển thị này bạn nhấp đúp chuột vào bảng đó trên thanh Pane Navigation. Access sẽ nhắc bạn nhập vào tên cho bảng và chuyển sang chế độ hiển thị Datasheet view.

**Bước 3:** Trên Tab DataSheet, trong nhóm Field & Columns nhấp chọn New Field .

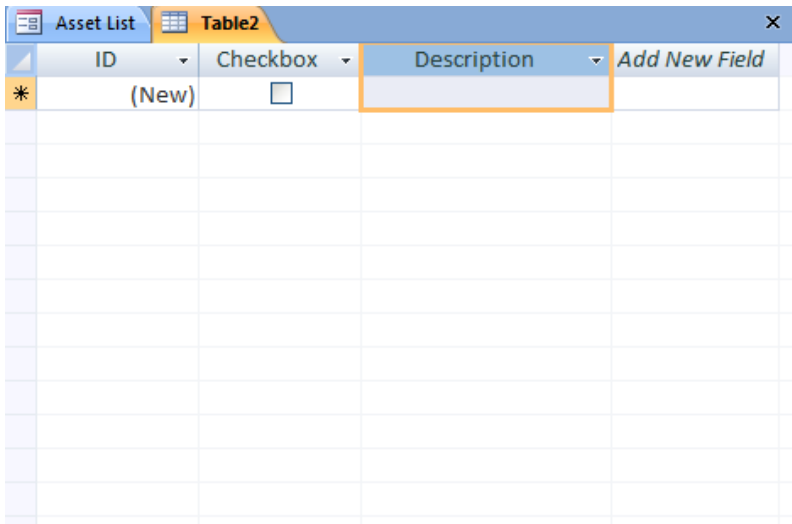


Access hiển thị Pane Field Templates, trong pane này chứa danh sách các kiểu trường thường được sử dụng.



Nếu bạn kéo thả một trong số chúng lên DataSheet của bạn, Access sẽ tự động thêm trường đó vào bảng với tên là tên của kiểu dữ liệu đối với các trường cơ bản và tên như trường gốc nếu nó được lấy từ các bảng khác. Ngoài ra Access cũng

thiết lập những thuộc tính phù hợp cho từng trường. Bạn cần phải kéo và thả trường vào phần chứa dữ liệu trên DataSheet.

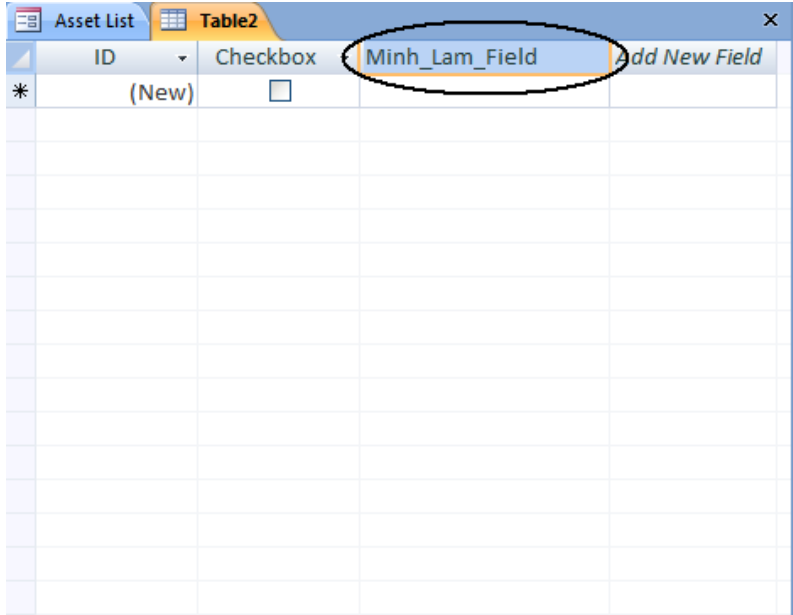


Như hình ảnh trên các bạn có thể thấy trường có tên Checkbox được lấy từ Basic Field và trường Description được lấy từ một bảng có sẵn trong cơ sở dữ liệu.

***Bước 4:*** Để nhập liệu vào bảng, bắt đầu gõ từ ô trống đầu tiên hoặc bạn cũng có thể dán dữ liệu từ một nguồn dữ liệu khác.

***Chú ý:***


- Để đổi tên một cột bạn nhấp đúp chuột vào phần tiêu đề của cột, sau đó gõ vào tên mới.



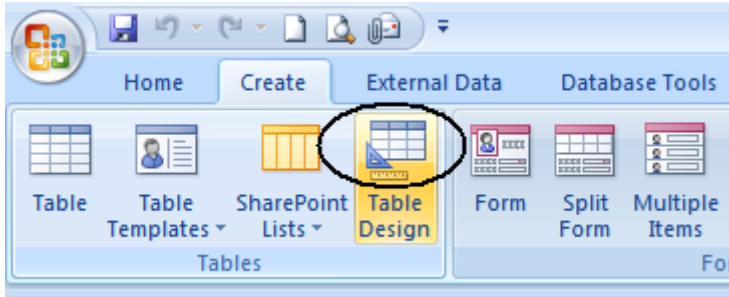
Tên trường bạn vừa đặt sẽ là nội dung hiển thị trường đó trên Pane Field List.

- Để di chuyển một cột, bạn hãy chọn nó bằng cách nhấp chuột lên phần tiêu đề của cột đó và kéo thả đến một vị trí bất kỳ trên DataSheet. Bạn cũng có thể chọn nhiều cột một lúc và di chuyển chúng đến vị trí mới.

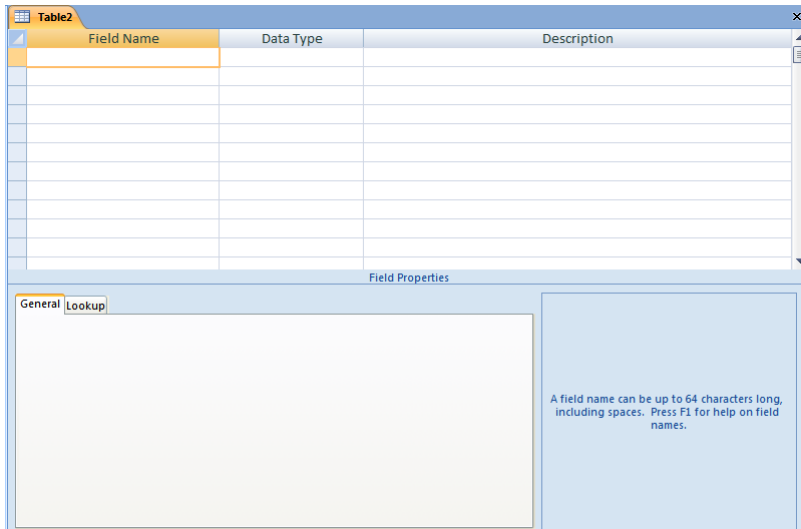
**Chèn thêm một bảng bắt đầu ở chế độ Design View:** Ở chế độ Design View, đầu tiên bạn tạo một cấu trúc bảng mới, sau đó bạn chuyển sang chế độ DataSheet để nhập liệu, hoặc bạn có thể nhập dữ liệu vào bảng bằng những phương thức khác như dán vào hoặc chèn thêm.

Bước 1: Trên Tab Create trong nhóm Tables nhấp chọn Table Design .

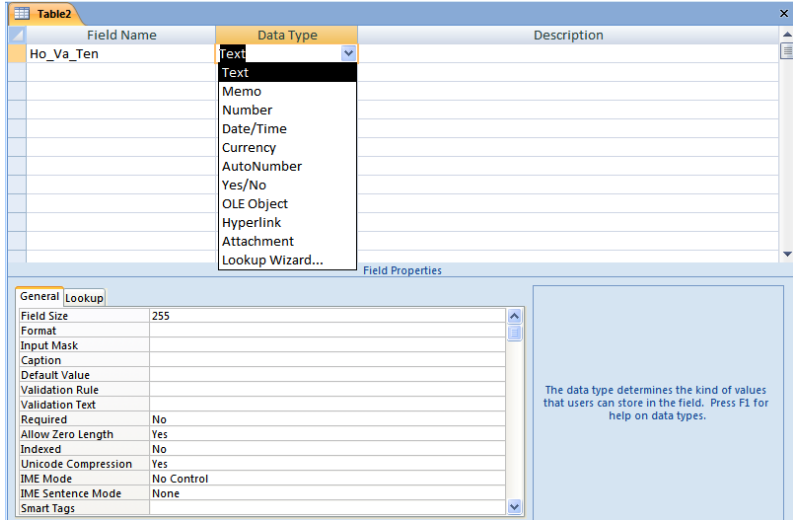




Sau bước này trên màn hình xuất hiện một bảng mới ở chế độ Design.

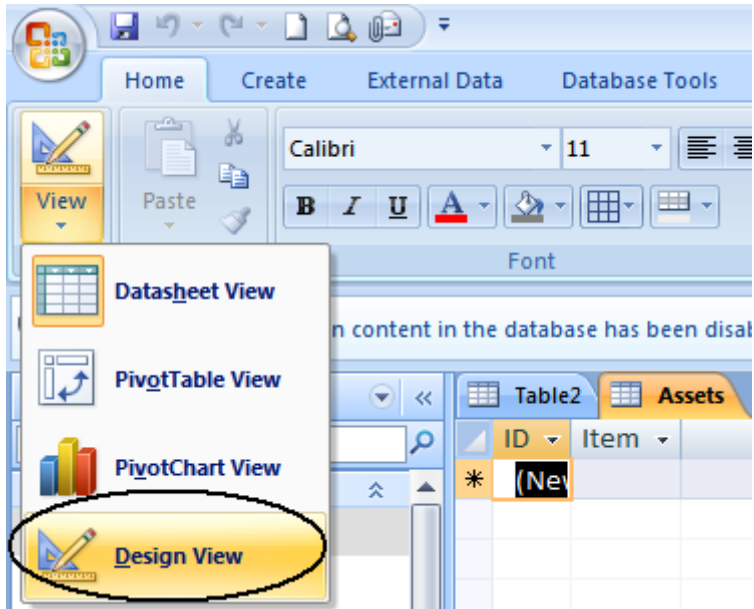


**Bước 2:** Đối với mỗi trường trong bảng, bạn nhập tên trường trong cột Field Name, sau đó lựa chọn kiểu dữ liệu trong danh sách Data Type.



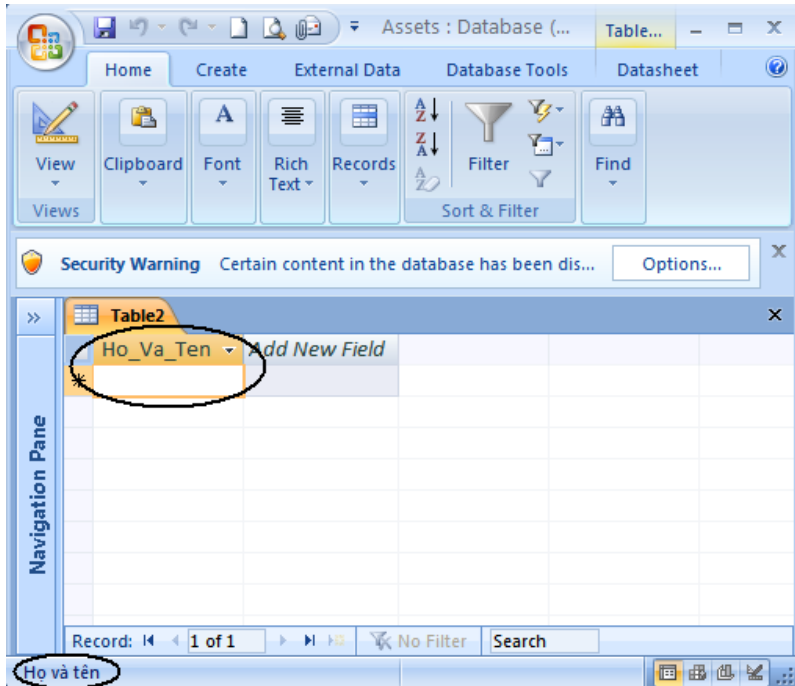
Chú ý:


Nếu bạn không nhìn thấy cột Field Name và cột Data Type, có thể bạn đang ở chế độ DataSheet thay vì chế độ hiển thị Design View, hãy nhấp chuột vào nút Design View trên thanh trạng thái của cửa sổ Access.

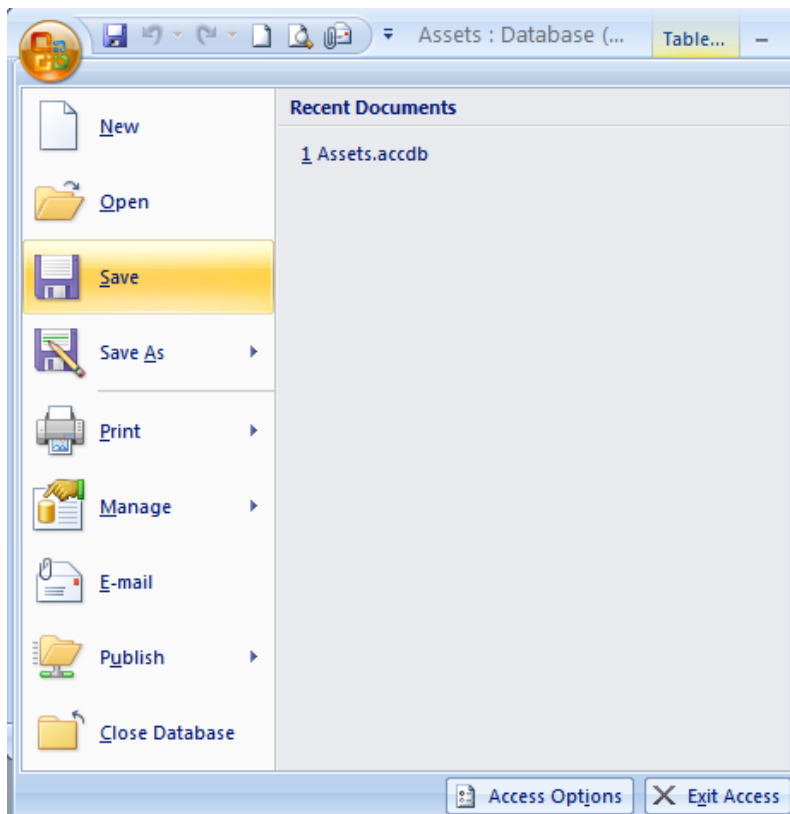


Access sẽ đưa ra thông báo yêu cầu bạn nhập tên mới cho bảng và sau đó tự động chuyển về chế độ Design View.

**Bước 3:** Nếu muốn, bạn có thể nhập vào thông tin để diễn giải cho các trường ở cột Description. Những thông tin này sẽ hiển thị trên thanh trạng thái khi bạn nhập dữ liệu vào trường đó. Và bạn cũng có thể sử dụng nội dung trên thanh trạng thái cho bất cứ điều khiển nào trên Form hoặc Report mà bạn tạo ra bằng cách kéo thả trường từ Pane Field List.



**Bước 4:** Sau khi đã thêm vào tất cả các trường cần thiết, bạn hãy lưu bảng lại bằng cách nhấp chọn nút Microsoft Office Button  rồi chọn Save hoặc nhấp tổ hợp phím Ctrl+S.

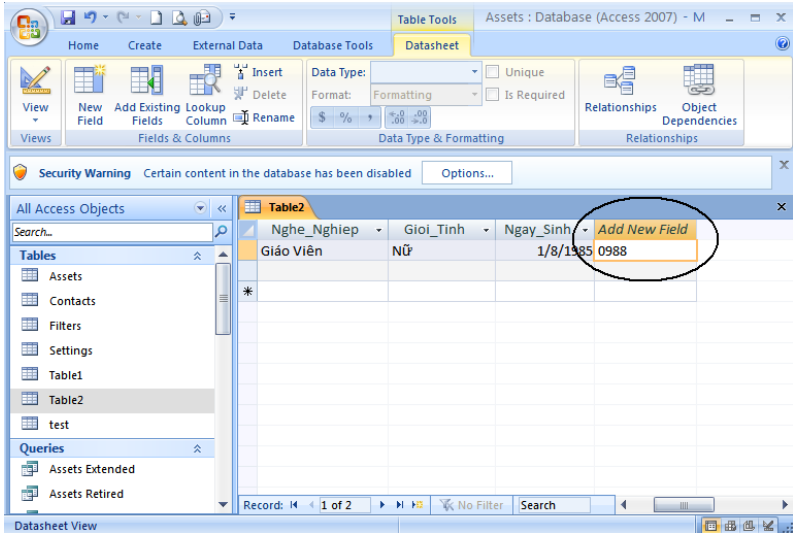


***Bước 5:*** Sau bước này bạn có thể nhập dữ liệu vào bảng bất cứ khi nào bạn chuyển sang chế độ hiển thị Datasheet View.

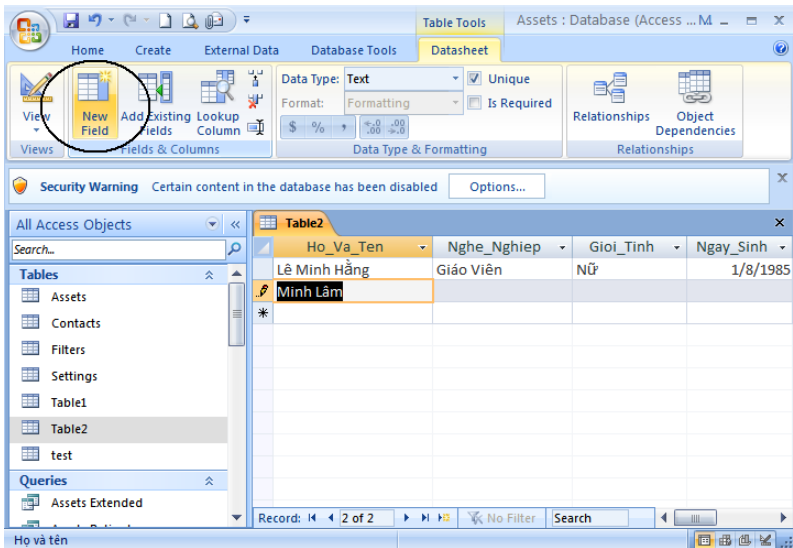
Ho_Va_Ten	Nghề_Nghiệp	Giới_Tính	Ngày_Sinh
Lê Minh Hằng	Giáo Viên	NỮ	1/8/1985
Minh Lâm			
*			

Ở chế độ này bạn di chuyển tới ô đầu tiên và nhập lần lượt dữ liệu cho từng bản ghi. Ngoài ra bạn cũng có thể dán dữ liệu vào bảng từ một nguồn dữ liệu khác.

Nếu sau khi đã nhập một số dữ liệu vào bảng, bạn lại chợt nhớ cần phải thêm một hoặc một số trường nữa mới phù hợp. Thay vì việc chuyển về chế độ Design View như lúc ban đầu thiết kế bạn gõ thẳng dữ liệu vào cột Add New Field ở chế độ hiển thị Datasheet View.

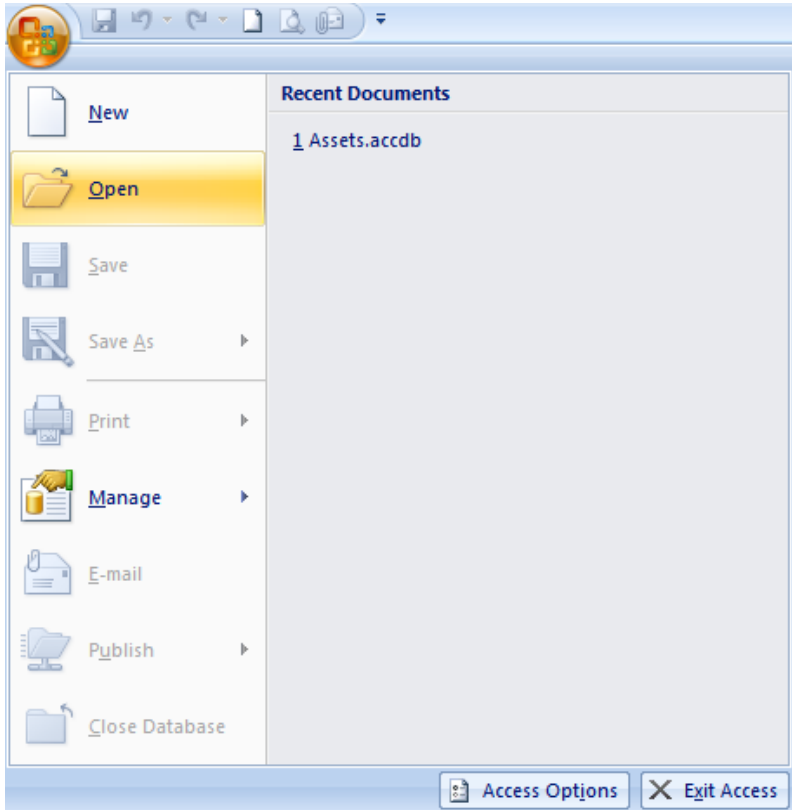


Hoặc bạn cũng có thể thêm một trường mới bằng cách nhấp chọn nút lệnh New Field trong nhóm Fields & Column của Tab Datasheet.



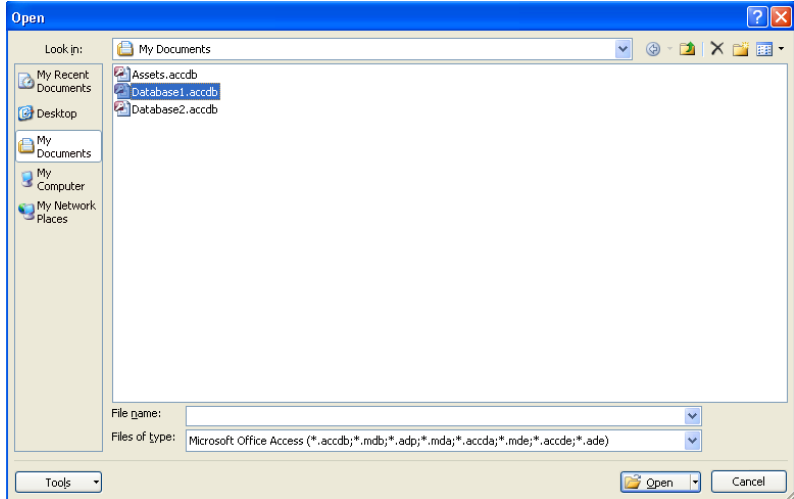
### 1.2.7. Mở một Database đã tồn tại

**Bước 1:** Nhấp chọn nút Microsoft Office Button , sau đó chọn Open.



**Bước 2:** Tìm tới thư mục có chứa Database rồi nhấp chọn DataBase cần mở.



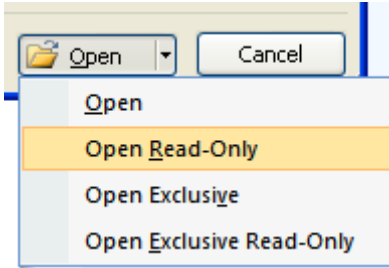


**Bước 3:** Khi đã tìm thấy và nhấp chọn Database cần mở bạn có thể làm theo một trong các cách sau đây:

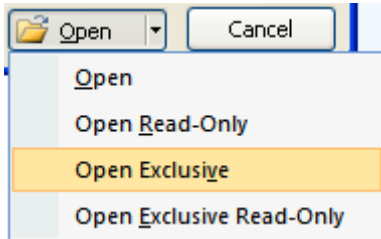
- Mở Database ở chế độ mặc định của chương trình: Bạn nhấp đúp chuột vào Database đó.

- Mở Database cho phép chia sẻ nhiều người dùng (chia sẻ nhiều người dùng: Là một Database cho phép nhiều hơn một người sử dụng có thể truy cập vào sửa đổi, nhập liệu tại cùng một thời điểm) trong một môi trường: Cho phép cả bạn và những người dùng khác có thể đọc và ghi vào cơ sở dữ liệu cùng một lúc bạn hãy nhấp Open.

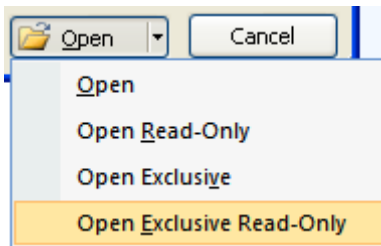
- Mở Database ở chế độ Read-only (chỉ đọc): Bạn có thể xem dữ liệu nhưng không thể chỉnh sửa được nó hãy nhấp chuột vào mũi tên chỉ xuống dọc theo nút Open/Open Read-only.



- Mở Database được phép Exclusive (độc quyền): Không có một người nào khác có thể mở Database này trong khi bạn đang mở nó. Để mở theo chế độ này bạn nhấp mũi tên dọc theo nút Open chọn Open Exclusive.

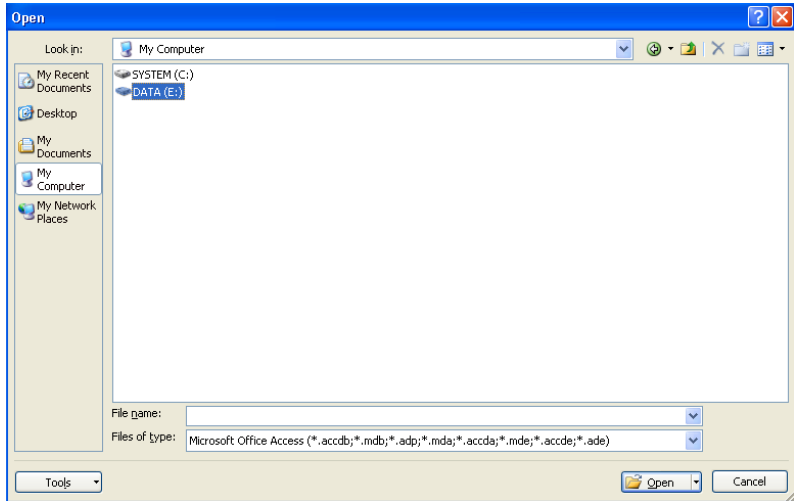


- Mở Database ở chế độ Exclusive Read-only (Độc quyền và chỉ đọc): Ở chế độ này khi bạn mở một Database bạn chỉ có thể đọc dữ liệu và những người khác thì không thể mở lại nó. Mở ở chế độ này bạn chọn mũi tên dọc theo nút Open/Open Exclusive Read-only.



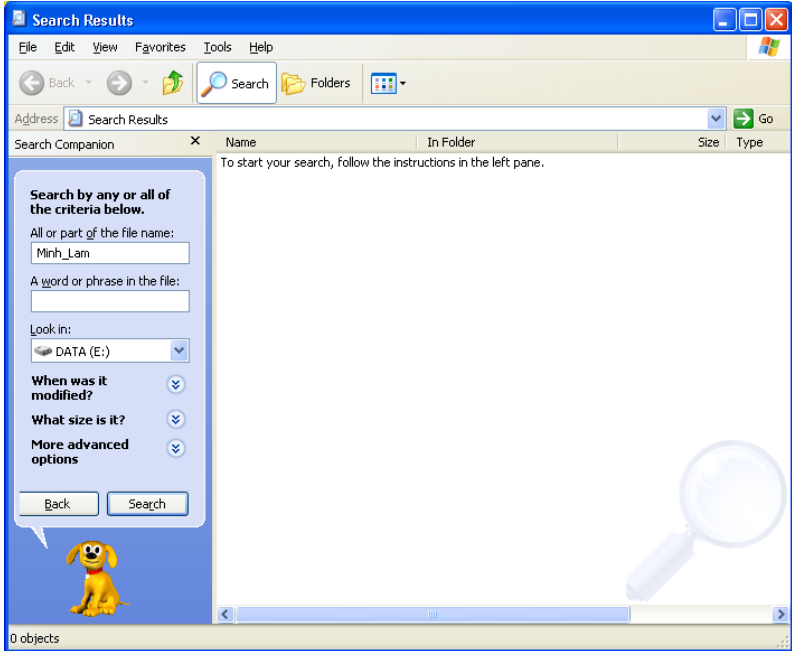
### 1.2.8. Nếu không tìm thấy Database cần mở

**Bước 1:** Trong hộp thoại Open, nhấp chuột chọn biểu tượng đường tắt My Computer hoặc trong hộp Look in nhấp chọn My Computer.



**Bước 2:** Trong danh sách ổ đĩa, nhấp chuột phải vào ổ đĩa mà bạn cho là có chứa Database mà bạn đang tìm kiếm chọn Search.

**Bước 3:** Cửa sổ tìm kiếm của Window xuất hiện. Nhập vào điều kiện tìm kiếm và sau đó nhấn Enter để tìm cơ sở dữ liệu.



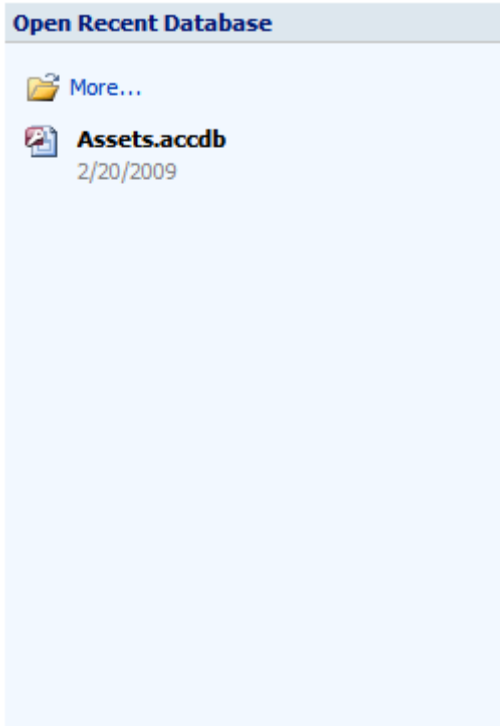
**Bước 4:** Nếu tìm thấy, để mở bạn hãy nhấp đúp chuột vào dữ liệu đó trong hộp thoại Search Results.


**Bước 5:** Vì tìm kiếm đã được bắt đầu từ hộp thoại Open, nên bạn phải nhấp chuột vào Cancel để tắt hộp thoại Open trước khi mở Database.

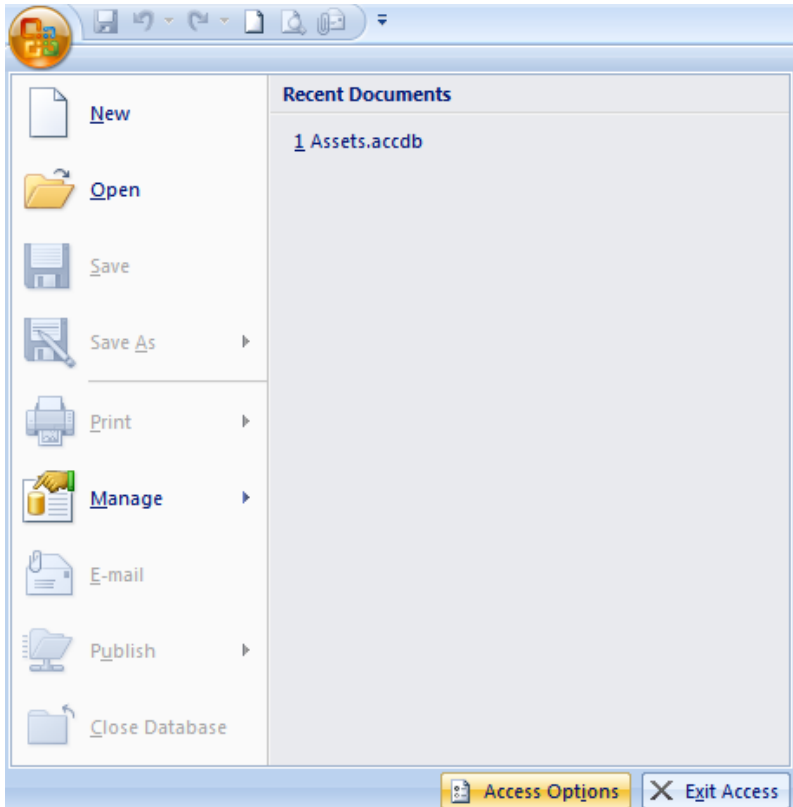
**Chú ý:** Bạn có thể mở trực tiếp một file dữ liệu ngoài định dạng một tệp tin (như là dBASE, Paradox, Microsoft Exchange, or Excel). Bạn cũng có thể trực tiếp mở bất cứ dữ liệu nguồn ODBC nào, như là Microsoft SQL Server hoặc Microsoft FoxPro. Access sẽ tự động tạo ra một cơ sở dữ liệu mới trong cùng một thư mục như là các tệp tin dữ liệu, và thêm những liên kết cho mỗi bảng trong cơ sở dữ liệu bên ngoài.

### 1.2.9. Mở Database làm việc gần đây nhất

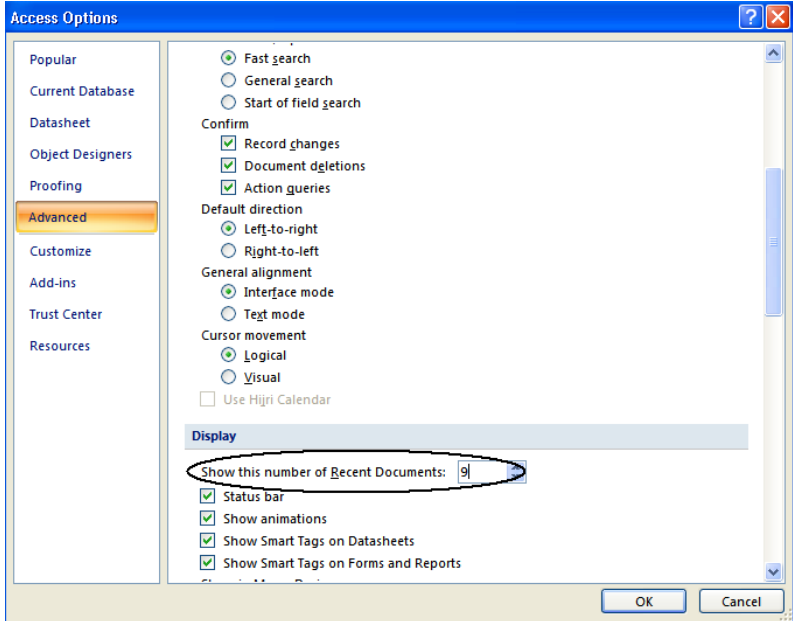
Bạn hãy nhấp chuột vào tên một File trong danh sách Open Recent Database trên trang Getting Started with Microsoft Office Access.




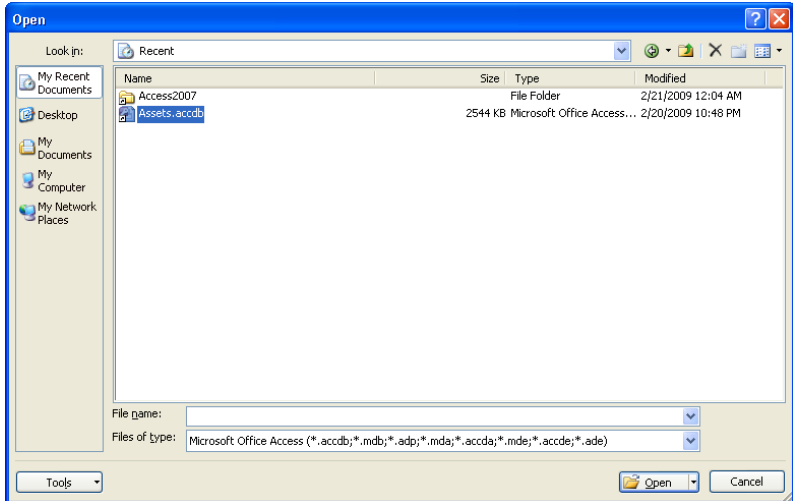
Để mở một trong những Database mà bạn làm gần đây nhất, bạn hãy nhấp chuột vào tên của Database đó trong danh sách Open Recent Database trên trang Getting Started with Microsoft Office Access. Access sẽ mở Database và thiết lập những tùy chọn như lần cuối cùng bạn làm việc với nó. Nếu trong danh sách các file được sử dụng trong thời gian gần đây không được hiển thị, bạn hãy nhấp vào nút Microsoft Office , và sau đó nhấp chọn **Access Options**



Hộp thoại Access Options xuất hiện, nhấp vào mục Advanced, trong cửa sổ ở giữa hộp thoại dưới mục Display chọn vào hộp TextBox Recently used file list nhập vào đó số số file cần lưu lại trong danh sách các tệp tin đã mở tối đa là 9.



Nếu bạn đang mở một Database bằng cách nhấp chọn nút  chọn nút Open. Bạn có thể hiển thị đường tắt tới danh sách Database mà bạn đã mở trước đó bằng cách nhấp chuột vào mục My Recent Documents trong hộp thoại Open.



### 1.3. Có gì mới trong Microsoft Office Access 2007

Microsoft Office Access 2007 cung cấp một bộ công cụ mạnh mẽ giúp bạn nhanh chóng bắt đầu theo dõi, báo cáo, và chia sẻ thông tin. Bạn có thể nhanh chóng tạo ra sức hấp dẫn và các chức năng theo dõi các ứng dụng tùy biến của một trong nhiều bộ Templates dựng sẵn, chuyển đổi một cơ sở dữ liệu hiện có, hoặc tạo ra một cơ sở dữ liệu mới, và bạn có thể làm như vậy mà không cần phải trang bị quá nhiều kiến thức về cơ sở dữ liệu.

#### 1.3.1. Để bắt đầu nhanh hơn

Microsoft Office Access 2007 cung cấp các Templates mà bạn có thể dễ dàng tạo ra cơ sở dữ liệu, bảng biểu, và các trường. Ngoài ra còn có những công cụ hiển thị và thiết kế mới giúp bạn trong quá trình tạo Cơ sở dữ liệu mới và làm việc với dữ liệu.

#### Cơ sở dữ liệu mẫu để xây dựng hoàn chỉnh các ứng dụng

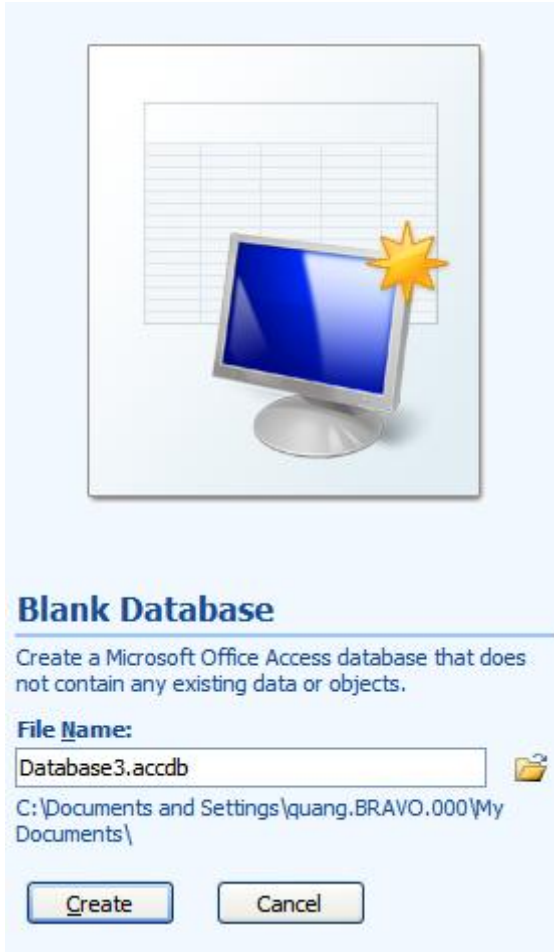
Microsoft Access 2007 cung cấp một bộ cơ sở dữ liệu bao gồm các mẫu thiết kế chuyên nghiệp giúp bạn có thể theo dõi



địa chỉ liên lạc, nhiệm vụ, sự kiện, học sinh, tài sản, và nhiều kiểu dữ liệu khác. Bạn có thể sử dụng ngay lập tức hoặc cải tiến và tinh chỉnh chúng cho phù hợp với yêu cầu quản lý của mình.



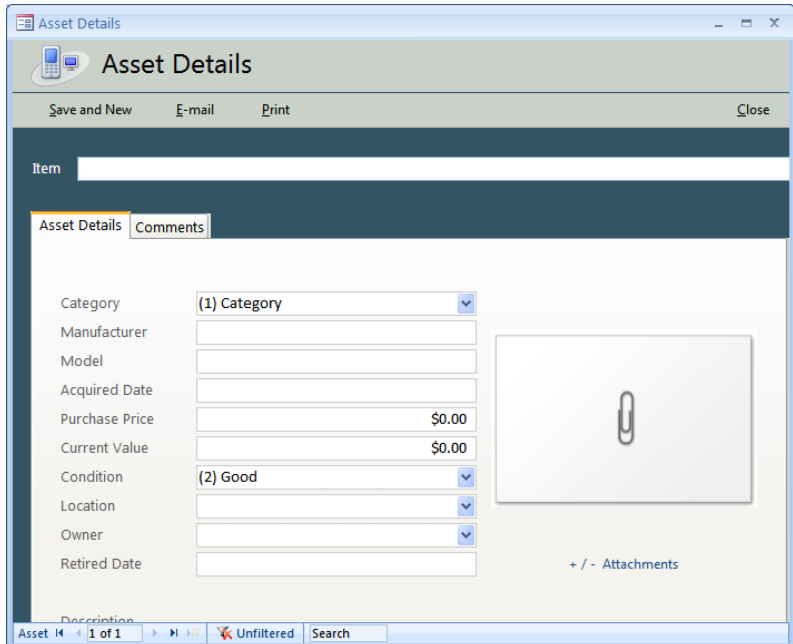
Trên trang Getting Started with Microsoft Office Access xuất hiện khi khởi động ứng dụng bạn có thể dễ dàng tạo một DataBase mới bằng cách nhấp chọn Blank DataBase.



Hoặc nhanh chóng bắt đầu tạo ra cơ sở dữ liệu của bạn bằng cách mở một trong những bản mẫu nằm trong mục Featured online Templates.

Mỗi DataBase Templates là một ứng dụng tương đối hoàn chỉnh với một bài toán chung. Chúng bao gồm các Table, Form, Report, Query, Macro và các mối quan hệ giữa các bảng. Những DataBase Template này cho phép bạn có thể sử

dụng chúng ngay lập tức nếu bạn cho là cấu trúc của nó đáp ứng được yêu cầu quản lý của mình. Nếu vẫn chưa đủ bạn có sử dụng, kế thừa chúng để tạo ra những cơ sở dữ liệu phù hợp hơn yêu cầu mình.

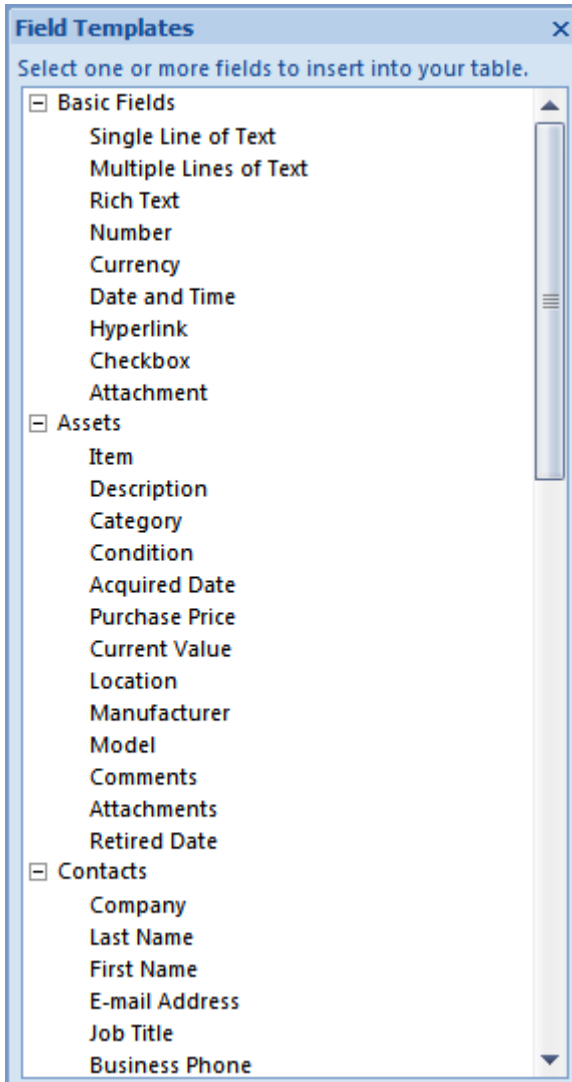


Ngoài các Template mà Microsoft Access 2007 cung cấp bạn cũng có thể kết nối với Microsoft Office Online để có thể tải về các Template khác.

### Field và Table Templates

Thay vì việc bỏ rất nhiều thời gian để thiết kế các trường bạn có thể sử dụng lại các trường dựng sẵn (File Templates) với tên, kiểu dữ liệu và các thuộc tính được định nghĩa từ trước. Bạn có thể kéo thả trực tiếp các trường mà bạn cần trong Panel Fields Templates vào một Datasheet. Các trường được tạo ra dựa trên (Based on) các file XML schema (. Xsd) giúp bạn có thể thiết lập, xây dựng thành những định nghĩa chuẩn

của riêng mình và chia sẻ sử dụng trong phòng hoặc nhóm làm việc.

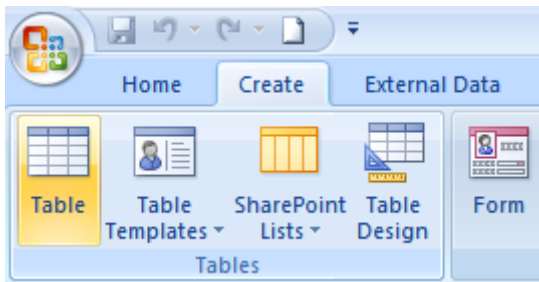


Ngoài ra, Microsoft Office Access 2007 còn bao gồm các Table thường được sử dụng trong cơ sở dữ liệu. Ví dụ, bạn

có thể sử dụng Contacts table template để thêm Table Contacts vào cơ sở dữ liệu của bạn. Bảng này bao gồm các trường như Last Name, First Name, and Address. Các thuộc tính của những trường này cũng được thiết lập sẵn để bạn có thể sử dụng ngay chúng. Còn có các bảng mẫu có sẵn khác như Tasks, Issues, Events, và Assets.

### **Nâng cấp Datasheet view**

Với Microsoft Access 2007 bạn có thể dễ dàng tạo một bảng dữ liệu bằng cách nhấp chọn Tab Create rồi chọn Table.



Microsoft Access 2007 sẽ tự động xác định kiểu dữ liệu phù hợp nhất cho mỗi trường sau khi bạn gõ xong dữ liệu và di chuyển qua nó.

Để thêm một trường mới bạn nhấp chuột tới cột cuối cùng (Add new field) và thay đổi dữ liệu trên nó.

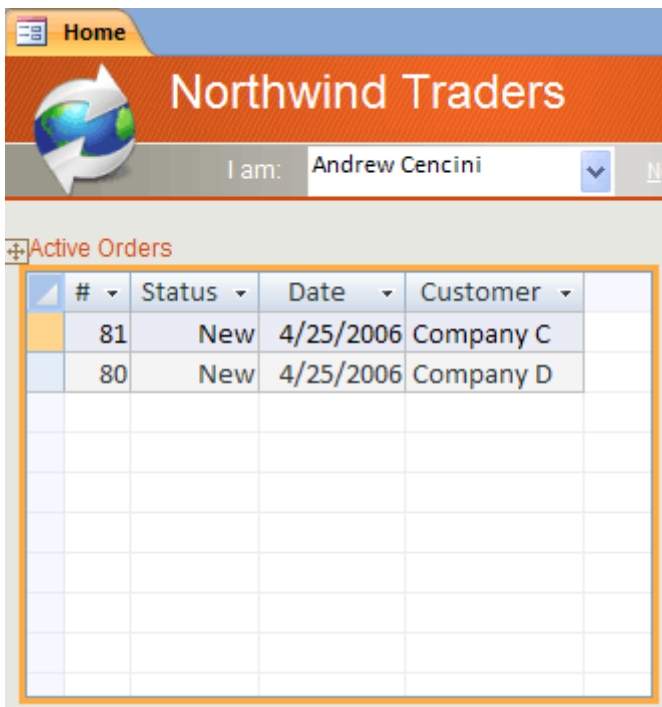
Table1					
	ID	Ho_Dem	Ten	Tuoi	Add New Field
	1	Bùi Bình	Minh	25	
	2				
*	(New)				

Và nếu bạn cần thay đổi kiểu dữ liệu hoặc định dạng kiểu dữ liệu hiển thị của một trường mới hay một trường đã có bạn

có thể sử dụng các nút chức năng trên thanh Ribbon. Nó là một phần của giao diện người dùng Microsoft Office fluent. Bạn cũng có thể dán một bảng dữ liệu từ Microsoft Excel vào một Datasheet mới trong Microsoft Access 2007, bằng cách tạo trước một cấu trúc các trường tương ứng với bảng dữ liệu mà bạn cần dán vào.

### **Layout View mới giúp cải thiện tốc độ thiết kế Forms và Report.**

Sử dụng Layout View để thực hiện các thay đổi thiết kế trong khi hiển thị dữ liệu trong một Form hoặc một Report.



Ví dụ, bạn có thể thêm vào một trường trên lưới thiết kế bằng cách kéo thả từ Field list pane, hoặc bạn có thể thay đổi các thuộc bằng cách sử dụng thuộc tính trên Sheet. Layout View

cung cấp một cách bố trí mới cho các Control đó là stacked layout (Bố trí dạng ngăn xếp) và tabular layout (Bố trí dạng bảng) - nhóm các Controls mà bạn có thể di chuyển và thay đổi kích thước như một đơn vị để bạn có thể dễ dàng sắp xếp lại các trường, cột, hàng, hoặc toàn bộ Layout. Bạn cũng có thể loại bỏ một trường hoặc thêm vào định dạng một cách dễ dàng trong Layout View. Chế độ Design View cũng tương tự như các phiên bản trước giúp bạn có thể biết thêm chi tiết về các Control, ở chế độ này cũng được hỗ trợ Layout dạng stacked và tabular.

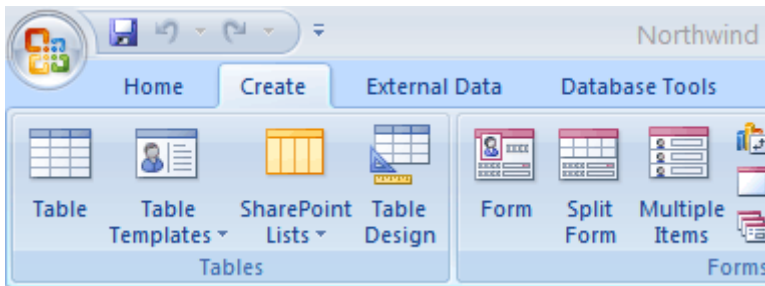
### **1.3.2. Microsoft Office fluent**

Những kết quả mới theo định hướng giao diện người dùng - Microsoft Office fluent giao diện người sử dụng - giúp bạn dễ dàng làm việc với Microsoft Access 2007. Ở những phiên bản trước Lệnh và các tính năng thường được dấu kín trong các menu, thanh công cụ phức tạp “bí hiểm”. Nhưng bây giờ bạn có thể dễ dàng tìm kiếm công việc theo định hướng sử dụng trên các tab chứa các nhóm lệnh. Nhiều ô hộp thoại được thay thế bằng các lựa chọn hiển thị trên thanh công cụ sẵn sàng cho người dùng sử dụng, ngoài ra nó còn được mô tả trên các Tooltips hoặc hiển thị dưới dạng các ví dụ.

Không có bất cứ một vấn đề gì với các hoạt động mà bạn đang thực hiện trong giao diện người dùng mới - cho dù đó là tạo ra một Report hoặc nhập dữ liệu – Hãy tiếp cận với những công cụ hữu ích nhất mà chúng tôi sẽ trình bày dưới đây chắc chắn các bạn sẽ thành công trong việc nghiên cứu và sử dụng Microsoft Access 2007.

**Microsoft Office fluent giao diện người sử dụng:** Microsoft Office fluent giao diện người dùng bao gồm các khu vực tiêu chuẩn (standard area) được gọi là Ribbon, trong đó có chứa các nhóm lệnh được tổ chức, sắp xếp theo chức

năng. Ribbon thay thế các lớp trình đơn và thanh công cụ cổ điển trong các phiên bản Access cũ.



Sử dụng Office fluent Ribbon để xác định vị trí các nhóm lệnh có liên quan nhanh hơn. Ví dụ, nếu bạn cần phải tạo một Form hoặc Report, hãy sử dụng một trong những lệnh trên Tab Create. Việc thiết kế bố trí giao diện mới làm cho việc tìm kiếm lệnh, chức năng mà bạn cần trở nên nhanh chóng, thêm vào đó bạn còn có thể khám phá ra những tính năng khác mà trước đây bạn chưa hề biết tới. Các nút lệnh được đặt gần hơn trên giao diện, điều đó có nghĩa là bạn không cần phải tìm kiếm trong menu hay cố nhớ vị trí của chúng ở đâu đó.

### **Phím chức năng của giao diện người dùng Microsoft Office Fluent bao gồm:**

- Tab lệnh: Các Tab chứa những nút lệnh thường được sử dụng cùng nhau, nhờ vậy khi cần sử dụng bạn có thể dễ dàng tìm thấy chúng.
- Nội dung các Tab lệnh: Một Tab lệnh sẽ xuất hiện tùy thuộc vào bối cảnh làm việc của bạn trên màn hình giao diện. Một nội dung tab lệnh sẽ chứa những lệnh có nhiều khả năng để áp dụng cho những gì bạn đang làm việc.
- Toolbar truy cập nhanh: Với một thanh công cụ Standard xuất hiện trên Ribbon, chỉ bằng một cái nhấp chuột bạn có



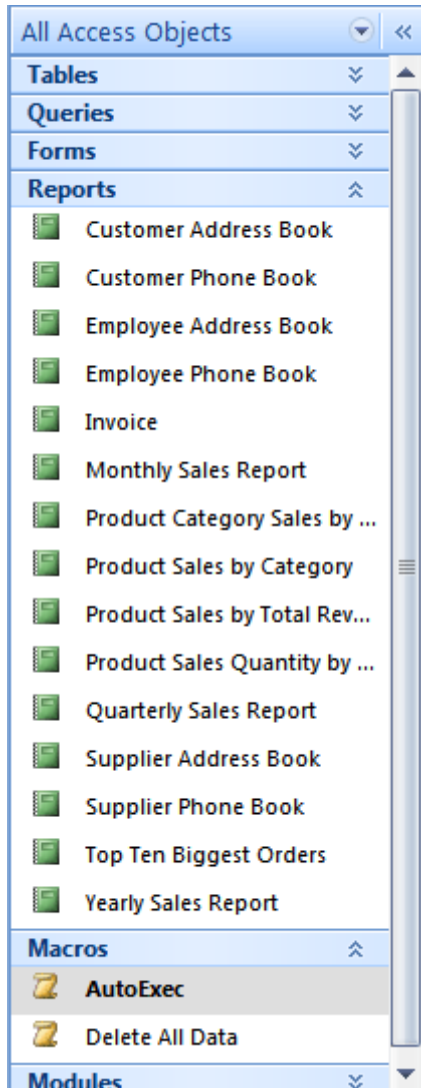
thể truy cập tới những lệnh cần thiết chẳng hạn như Save Undo, ....

**Getting Started with Microsoft Office Access page:** Trang này cung cấp khả năng truy cập nhanh đến một thư viện, các cơ sở dữ liệu mẫu được thiết kế chuyên nghiệp, ngoài các cơ sở dữ liệu mà bạn đã mở trong thời gian gần đây (nếu bạn đang kết nối vào Internet) và các liên kết đến các bài viết Office Online phổ biến.

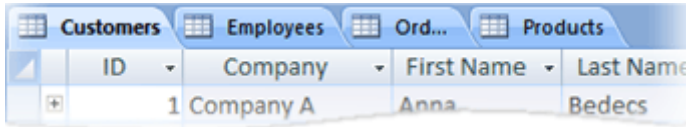


**Navigation Pane:** Pane này liệt kê và cung cấp cách thức truy cập dễ dàng tới tất cả các đối tượng trong cơ sở dữ liệu hiện tại đang mở. Bạn có thể sử dụng Navigation Pane để tổ chức, sắp xếp các Object của bạn theo kiểu (Object Type), ngày tạo (Create Date), ngày sửa đổi (Modified Date), ..., hoặc tùy chỉnh trong các nhóm mà bạn tạo ra. Nếu bạn cần thêm không gian để làm việc với các Form khi thiết kế, bạn có thể dễ dàng thu hẹp Navigation Pane để nó chiếm ít không

gian hơn. Pane Navigation được tạo ra để thay thế cửa sổ DataBase đã được sử dụng trong phiên bản trước đây của Access 2007.



**Tabbed objects:** Các Table, queries, forms, reports, macros của bạn được hiển thị như là các Tabbed objects trong Access window.

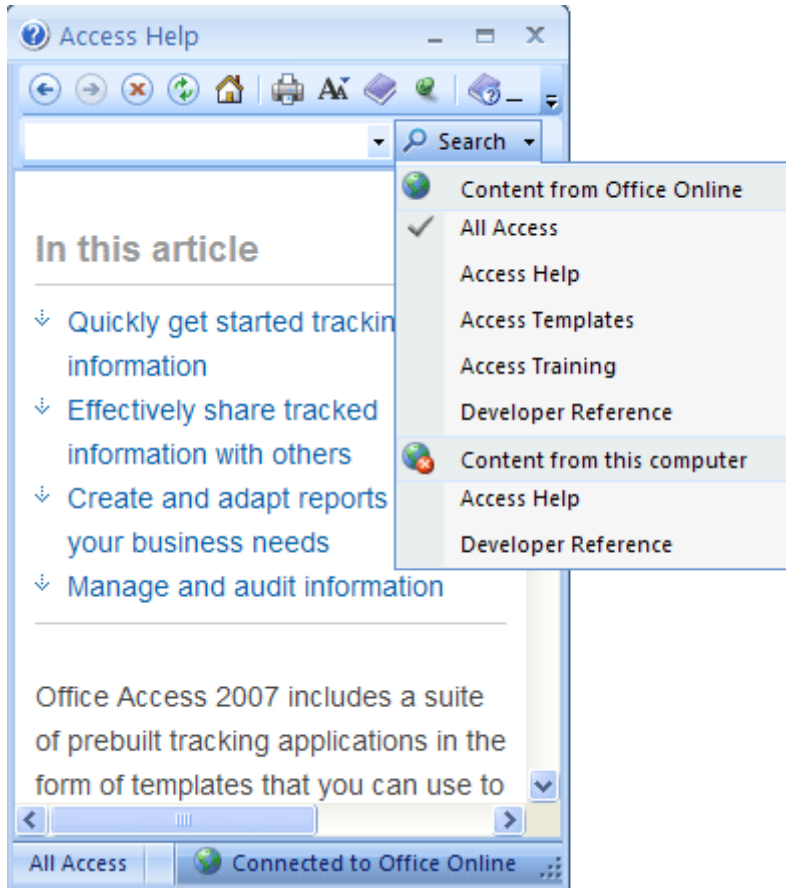


Bằng cách nhấp chuột vào các đối tượng Tab, bạn có thể dễ dàng chuyển đổi giữa các đối tượng khác nhau.

**Status bar:** Là thanh trạng thái ở dưới cùng của cửa sổ, hiển thị thông tin trạng thái, các nút lệnh cho phép bạn chuyển đổi giữa các lần hiển thị.

**Mini toolbar:** Là một yếu tố, giống như một thanh công cụ, xuất hiện khi một nội dung văn bản được lựa chọn, giúp bạn có thể dễ dàng áp dụng các định dạng như in đậm hoặc nghiêng, hoặc thay đổi font chữ.

**Help window:** Không giống như các phiên bản trước của Access, Microsoft Access 2007 cho phép truy cập cả Access Help và nội dung Developer Reference trên cùng một cửa sổ trợ giúp một cách dễ dàng. Bạn cũng có thể dễ dàng thay đổi phạm vi tìm kiếm của chỉ mình nội dung Developer Reference.



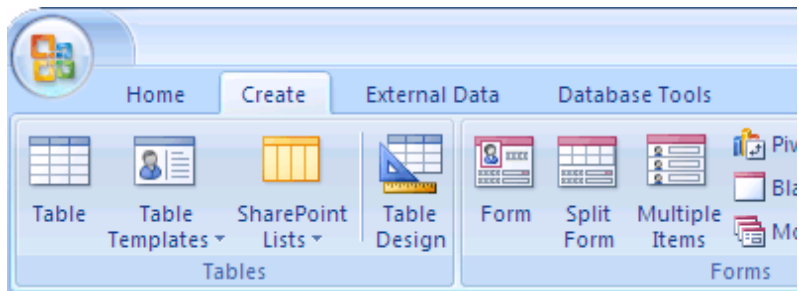
### **1.3.3. Thêm các công cụ mạnh để tạo đối tượng**

Office Access 2007 cung cấp một môi trường trực quan để tạo các Form và các Report, cho phép bạn nhanh chóng tạo ra các Form và các Report. Chúng hiển thị sau khi đã được sắp xếp, lọc và nhóm theo các thông tin về từng đối tượng.

#### **Create tab**

Sử dụng Tab Create giúp bạn nhanh chóng tạo ra các Form, các Report, Table, Query, macros và Module mới. Nếu bạn

chọn một Table hoặc một Query trong Navigation Pane, bạn có thể tạo ra một Form hoặc một báo cáo mới dựa trên các đối tượng đó bằng cách nhấp chuột vào nút Form hoặc Report. Các Form và các Report mới này được tạo ra gần như ngay lập tức và các thiết kế trên đó cũng được nâng cấp đáng kể. Tự động tạo ra các Form và các Report rất chuyên nghiệp với logo và tiêu đề được định nghĩa sẵn. Ngoài ra, nó còn tự động tạo ra một báo cáo bao gồm cả thông tin về ngày tháng năm, Footer, Totals.



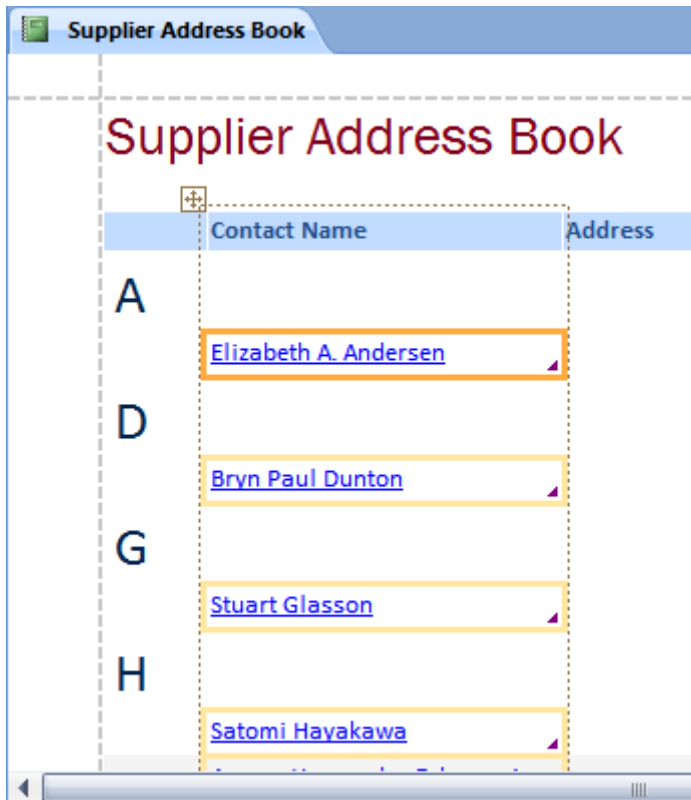
Bạn sẽ thấy rằng các đối tượng được tạo bằng cách này giúp bạn rất nhiều về thời gian và nhanh chóng đưa bạn tiến xa hơn trong quá trình nghiên cứu ứng dụng.

### **Chế độ hiển thị mới: Report view và Layout view**

Hai chế độ hiển thị mới này cho phép bạn làm việc với các Form và các Report một cách trực quan hơn. Bằng cách sử dụng Report Views, bạn có thể duyệt vẽ một cách chính xác báo cáo mà không cần phải in ra hoặc xem ở chế độ xem trước khi in. Để tham chiếu tới một số bản ghi, bạn sử dụng tính năng lọc, hoặc sử dụng công cụ tìm kiếm theo nội dung văn bản trong bản ghi đó.

Report View cho bạn khả năng duyệt báo cáo nhưng Layout View lại giúp bạn thay đổi thiết kế báo cáo ngay khi bạn đang xem chúng. Bạn có thể sử dụng Layout View để thực hiện thay đổi các thiết kế chung khi đang hiển thị nội dung

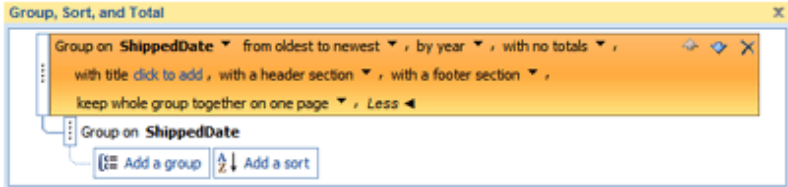
dữ liệu trên Form hoặc báo cáo. Ví dụ, thêm vào một trường bằng cách kéo một trường đó từ cửa sổ Field List pane, hoặc thay đổi thuộc tính cho đối tượng bằng cách sử dụng Properties Sheet. Layout view với những hỗ trợ mới, stacked layout và tabular layout (Nhóm các Control), cho phép bạn dễ dàng sắp xếp lại các trường, cột, hàng, hoặc toàn bộ Layout. Bạn cũng có thể dễ dàng loại bỏ một trường hoặc thêm vào định dạng trong Layout View.



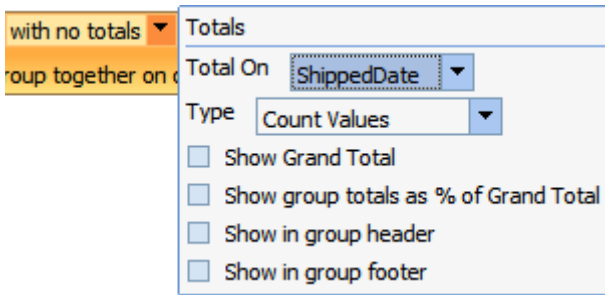
### **Group và sort trong các báo cáo**

Với Microsoft Access 2007 bạn sẽ có một công cụ rất tốt để Group, Sort, và thêm tổng con trong báo cáo. Bạn sẽ tìm thấy

một giao diện người dùng mới, dễ dàng hơn để di chuyển và hiểu được, và khi nó được sử dụng với Layout view mới, bạn sẽ nhìn thấy hiệu quả của các thay đổi ngay lập tức.

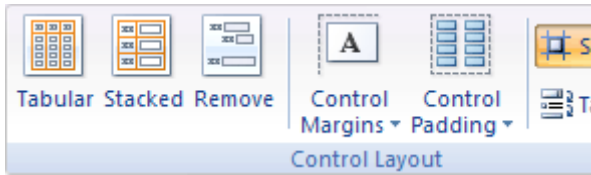


Giả sử bạn cần tạo một báo cáo về doanh số bán hàng theo khu vực. Hãy sử dụng Layout View và Panel Group, Sort, and Total để thêm vào một cấp nhóm, và xem sự thay đổi hiện ra trên báo cáo! Với cách tạo dòng Total mới bạn có thể dễ dàng tạo Total ở nhiều dạng khác nhau trên Header, Footer của báo cáo như kiểu Sum, average, count, maximum, hoặc minimum. Với một Total đơn giản bạn không cần thiết phải tạo ra một trường tính toán mà có thể nhấp chọn luôn trên công cụ này.



### **Điều khiển bố cục trợ giúp việc tạo Form, Report.**

Các Form, Report thường bao gồm các thông tin tabular, như là một cột có chứa tên của khách hàng hay một hàng chứa tất cả các trường dành cho khách hàng đó. Bạn có thể sử dụng Microsoft Access 2007 để nhóm những Control trong một Layout vào với nhau một cách dễ dàng, nó có thể bao gồm cả các Label.



Bạn có thể chọn các điều khiển từ các vùng khác, ví dụ như nhãn trong phần Header hoặc Footer, nó thật sự rất linh hoạt và bạn có thể dễ dàng:

- Di chuyển hoặc thay đổi kích cỡ của một Layout. Ví dụ, di chuyển một cột từ trái qua phải.
- Định dạng một Layout. Ví dụ, Tên một khách hàng được đặt trong cột in đậm và để nổi.
- Thêm một trường.
- Xóa một trường.

## Sales Report

	Q2	Q1
	\$1,632.50	\$13,800.00
	\$7,327.50	\$680.00
	\$3,132.00	\$1,551.00
	\$3,520.00	\$1,049.00
	\$3,786.50	\$0.00
	\$3,625.25	\$0.00
	\$2,778.00	\$127.50
	\$620.00	\$1,930.00

### **Split Forms mới hiển thị dữ liệu nhanh hơn**

Sử dụng Split Form mới để tạo Form mà kết hợp từ một Datasheet view và một Form View. Bạn có thể thiết lập



thuộc tính để chỉ định Access đặt lại vị trí của DataSheet lên trên, xuống dưới hay sang trái sang phải.

The screenshot shows the Microsoft Access interface. At the top, there is a window titled 'Customers' with a tab labeled 'Customers'. Below the title bar, there is a form with the following fields:

- ID: 1 (highlighted with a dashed orange border)
- Company: Company A
- First Name: Anna
- Last Name: Bedecs
- E-mail Address: (empty)
- Job Title: Owner
- Business Phone: (123)456-7890
- Home Phone: (empty)

On the right side of the form, there are labels for 'City:', 'State/P...', 'ZIP/Pos...', 'Country', 'Web Pa...', 'Notes:', and 'Attachr...'.

Below the form is a data table with the following columns: ID, Company, First Name, Last Name, and E-mail Address. The table contains 8 rows of data:

ID	Company	First Name	Last Name	E-mail Address
1	Company A	Anna	Bedecs	
2	Company B	Antonio	Gratacos Solso	
3	Company C	Thomas	Axen	
4	Company D	Christina	Lee	
5	Company E	Martin	O'Donnell	
6	Company F	Francisco	Pérez-Olaeta	
7	Company G	Ming-Yang	Xie	
8	Company H	Elizabeth	Andersen	

At the bottom of the window, there is a status bar showing 'Record: 1 of 29', 'No Filter', and a 'Search' button.

## **Nhúng Macros vào các Form, Report**

Một cách mới và đáng tin cậy, bạn có thể nhúng macros để tránh phải viết mã. Một Macro nhúng được lưu giữ trong một thuộc tính và là một phần của đối tượng chứa nó. Bạn có thể chỉnh sửa những thiết kế của một Macro nhúng mà không phải lo lắng về những điều khiển khác cũng sử dụng Macro này – các Macro nhúng hoàn toàn độc lập. Macros nhúng rất

đáng tin cậy bởi vì chúng có khả năng tự động ngăn chặn một số những thao tác không an toàn.

On Lost Focus	
On Dbl Click	[Embedded Macro] ▼ ...
On Mouse Down	

### 1.3.4. Những kiểu dữ liệu và điều khiển mới

Microsoft Access 2007 cung cấp nhiều kiểu dữ liệu và điều khiển mới cho phép lưu trữ nhiều dạng dữ liệu khác nhau, và bạn có thể nhập chúng dễ dàng hơn.

#### Trường đa trị cho các dữ liệu phức tạp

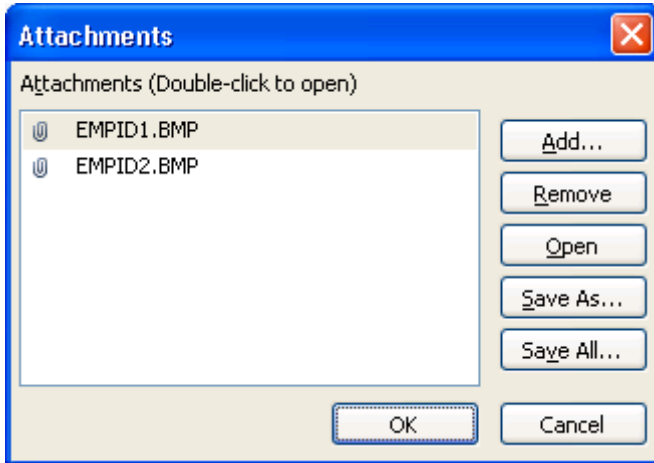
Bạn có thể tạo ra một trường chứa nhiều giá trị, còn được gọi là dạng dữ liệu phức tạp. Giả sử bạn cần phải chỉ định một công việc cho một trong những nhân viên hoặc nhà thầu của mình, nhưng bạn muốn chỉ định nó cho nhiều hơn một người. Trong hầu hết các hệ thống quản lý cơ sở dữ liệu và trong các phiên bản cũ của Access, bạn cần tạo ra rất nhiều các mối quan hệ để làm điều này một cách chính xác. Trong Microsoft Access 2007, phần cứng sẽ thực hiện việc này cho bạn, khi bạn chọn một trường mà bạn thiết lập để chấp nhận nhiều giá trị. Trường đa trị trong Microsoft Access 2007 sẽ rất thích hợp để bạn làm việc này. Office Access 2007 có khả năng tương thích với rất nhiều kiểu dữ liệu.

AssignedTo	▼
David Hamilton, Eva Valverde	▼

#### Trường Attachment để chứa Files

Kiểu dữ liệu Attachment mới cho phép bạn dễ dàng lưu trữ tất cả các loại tài liệu và tập tin nhị phân trong cơ sở dữ liệu của bạn mà không cần quan tâm tới kích thước của chúng. Microsoft Access 2007 sẽ tự động nén chúng lại đến mức có thể để giảm thiểu không gian lưu trữ. Nếu bạn cần đính kèm một tập tin Microsoft Word 2007 và một bản ghi hoặc lưu trữ

một bức ảnh kỹ thuật số vào CSDL của mình? Bằng cách sử dụng trường Attachment sẽ làm cho công việc trở nên hết sức dễ dàng. Thậm chí bạn có thể thêm nhiều tập tin đính kèm chỉ với một bản ghi duy nhất.



### **Trường Memo lưu trữ Rich Text và hỗ trợ lịch sử sửa đổi**

Với khả năng lưu trữ Rich Text của Microsoft Access 2007, trong một bản ghi số lượng văn bản của bạn sẽ không còn bị giới hạn. Bạn có thể định dạng văn bản với các tùy chọn (như đậm, nghiêng, phông chữ, màu sắc, và những tùy chọn định dạng phổ biến khác) và lưu giữ các văn bản trong cơ sở dữ liệu của mình. Rich-formatted text được lưu giữ trong một trường Memo như một văn bản HTML dựa trên định dạng tương ứng với Rich Text, là một kiểu dữ liệu trong Windows SharePoint Services.

Trường Memo sẽ hữu ích cho việc lưu trữ một số lượng lớn thông tin. Với Microsoft Access 2007, bạn có thể thiết lập thuộc tính Append Only để giữ lại tất cả lịch sử của những lần thay đổi thông tin trên trường đó. Và bạn hoàn toàn có thể xem lại những sự thay đổi này. Tính năng này cũng hỗ trợ theo dõi trong Windows SharePoint Services, nhờ vậy

bạn có thể sử dụng Access để xem nội dung lịch sử danh sách SharePoint.

### **Chọn này với Calendar**

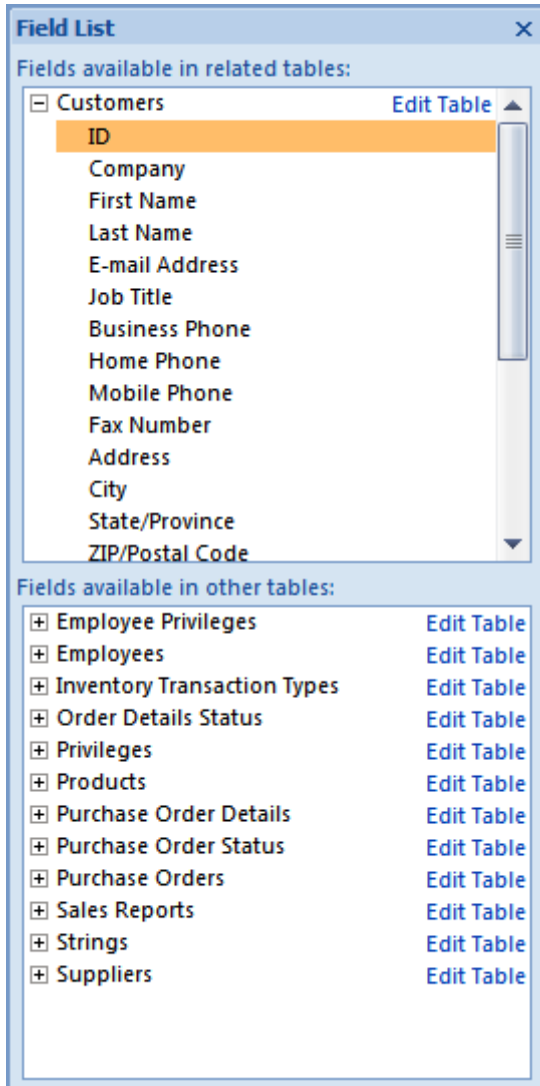
Những trường và những điều có kiểu dữ liệu kiểu Date/Time sẽ tự động cập nhật tính năng mới này: Một điều khiển lịch sẽ được gắn kèm cho phép chọn ngày tháng. Nút lịch tự động xuất hiện và trở chính xác tới ngày tháng hiện tại. Nếu bạn muốn biết thứ 6 tới là ngày bao nhiêu? Bạn chỉ Click chọn và một bảng lịch sẽ tự động xuất hiện cho phép bạn tìm và chọn ngày đó. Bạn cũng có thể tùy chọn tắt bảng lịch này trên trường hoặc điều khiển bằng cách sử dụng một thuộc tính.

### **1.3.5. Nâng cấp thiết kế và các công cụ phân tích**

Những công cụ mới trong Microsoft Access 2007 sẽ giúp bạn tạo ra các đối tượng cơ sở dữ liệu nhanh hơn và sau đó phân tích dữ liệu một cách dễ dàng hơn.

### **Nâng cao Field List pane**

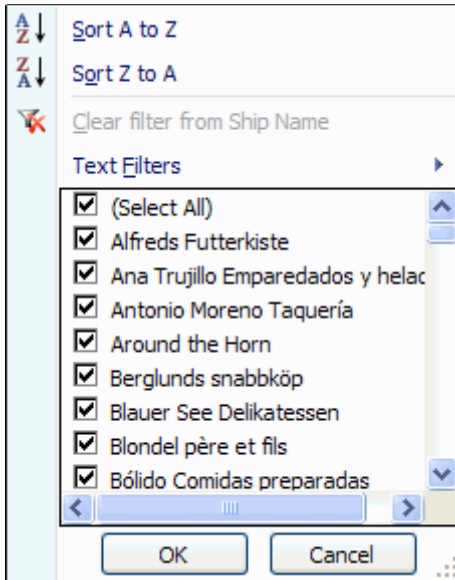
Với Pane Field List bạn có thể kéo thả trường từ một bảng vào trong bản ghi nguồn, các bảng có thể có mối quan hệ với nhau hoặc không. Microsoft Access 2007 đã tạo sẵn các cấu trúc vì vậy nếu một mối quan hệ giữa các bảng là cần thiết thì nó sẽ tự động tạo ra mối qua hệ đó.



## Nâng cao công cụ lọc, sắp xếp

Giả sử bạn cần phải nhanh chóng tìm thấy một giá trị phù hợp hoặc sắp xếp một cột của dữ liệu. Microsoft Access 2007 với chức năng AutoFilt cung cấp khả năng lọc nhanh

chóng để bạn có thể tìm thấy các dữ liệu mà bạn cần. Bạn cũng có thể dễ dàng lựa chọn từ các giá trị duy nhất trong một cột, điều này rất có ích cho các tình huống khi bạn không thể gọi lại tên mà bạn cần, hoặc bạn có thể sắp xếp các giá trị bằng cách sử dụng các trình đơn tùy chọn.



Các bộ lọc tùy chọn phổ biến nhất rất dễ dàng nhìn thấy trong trình đơn lệnh, bạn có thể nhanh chóng sử dụng chúng để giới hạn các thông tin dựa trên các dữ liệu nhập vào. Bộ lọc tùy chọn tự động thay đổi dựa trên kiểu dữ liệu mà bạn đang tác tác.

Order Date	Shipped Date	Ship
1/15/2006	1/22/2006	Shipping C
1/20/2006		
1/22/2006		
1/30/2006		
2/6/2006		
2/10/2006		
2/23/2006		
3/6/2006		
3/10/2006		
3/22/2006		
3/24/2006		
3/24/2006		
3/24/2006		Shipping C
3/24/2006		
4/7/2006	4/7/2006	Shipping C
4/5/2006	4/5/2006	Shipping C
4/8/2006	4/8/2006	Shipping C
4/5/2006	4/5/2006	Shipping C
4/5/2006	4/5/2006	Shipping C
4/5/2006	4/5/2006	Shipping C
4/5/2006	4/5/2006	Shipping C
4/5/2006	4/5/2006	Shipping C

Context menu for Order Date:

- Cut
- Copy
- Paste
- Sort Ascending
- Sort Descending
- Clear filter from Order Date
- Date Filters**
  - Equals 1/20/2006
  - Does Not Equal 1/20/2006
  - On or Before 1/20/2006
  - On or After 1/20/2006

Filter pane:

- Equals...
- Does Not Equal...
- Before...
- After...
- Between...
- Tomorrow
- Today
- Yesterday
- Next Week
- This Week
- Last Week
- Next Month
- This Month
- Last Month
- Next Quarter
- This Quarter
- Last Quarter
- Next Year
- This Year
- Year To Date
- Last Year
- Past
- Future
- All Dates In Period

Những tính năng lọc hay sắp xếp mới được thiết kế sẽ giúp bạn có được những kinh nghiệm nhất định làm việc không chỉ với Access 2007 mà cả Excel 2007.

### **Totals (Tổng cộng) và màu nền trong DataSheet**

Với dòng Total trong Datasheet View mới bạn có thể hiển thị một tổng (sum), hoặc đến bản ghi (count), tính giá trị trung bình (average), ... Sau khi bạn thêm dòng Total, bạn trở vào mũi tên trong các cột và chọn phép tính cần thiết muốn.

ID	Purchase Price
1	\$550.27
3	\$981.63
4	\$427.04
5	\$655.98
* {New}	\$0.00
<b>Total</b>	<b>\$2,614.92</b>

None
<b>Sum</b>
Average
Count
Maximum
Minimum
Standard Deviation
Variance

Datasheet, Report, Forms đều được hỗ trợ màu nền cho các row, bạn có thể thiết lập mặc định màu nền cho chúng. Tạo bóng cho các hàng cũng rất dễ dàng, và bạn có thể chọn bất kỳ màu sắc nào mà mình thích.





	Order ID	Employee
+	30	Anne Hellung-Larsen
+	31	Jan Kotas
+	32	Mariya Sergienko
+	33	Michael Neipper
+	34	Anne Hellung-Larsen
+	35	Jan Kotas
+	36	Mariya Sergienko
+	37	Laura Giussani

### **1.3.6. Tăng cường an ninh**

Microsoft Access 2007 tăng cường chức năng bảo mật một cách mạnh mẽ và thông minh với Windows SharePoint giúp bạn quản lý hiệu quả, theo dõi các ứng dụng an toàn hơn trước rất nhiều. Bằng cách lưu trữ dữ liệu theo dõi của bạn trong danh sách ứng dụng trên Windows SharePoint Services, bạn có thể kiểm tra xem xét lại lịch sử, khôi phục lại thông tin bị xóa, phân quyền dữ liệu truy cập.

### **1.3.7. Cách thức mới để chia sẻ dữ liệu và cộng tác với những người khác**

Bằng cách sử dụng các tính năng cộng tác mới của Microsoft Access 2007, bạn có thể thu thập và chia sẻ thông tin với những người khác hiệu quả hơn trong một môi trường bảo mật trên Web.

### **Tích hợp với Windows SharePoint Services**

Microsoft Access 2007 với Windows SharePoint Services trong một số trường hợp, giúp bạn tận dụng lợi thế của nhiều tính năng cộng tác trong Windows SharePoint Services.

- **Sửa đổi dữ liệu theo dõi:** Với sự giúp đỡ của Windows SharePoint Services, bạn có thể xem lịch sử sửa đổi của một bản ghi. Giả sử bạn cần phải biết ai đã thay đổi nội dung một hàng và nó được thay đổi khi nào? Tất cả có thể tìm thấy bằng cách xem trong lịch sử sửa đổi.

- **Cho phép cài đặt:** Với sự giúp đỡ của Windows SharePoint Services, bạn có thể đặt quyền truy cập cho phép người dùng thêm vào danh sách. Bạn có thể chọn lọc cho phép hoặc từ chối một số người dùng truy cập. Bạn cũng có thể chỉ định giới hạn chế quyền chỉ đọc hoặc Full (mọi thay đổi tác động đều được chấp nhận).

- **Lấy từ Recycle Bin:** Bạn có thể sử dụng Recycle Bin mới trong Windows SharePoint Services để dễ dàng xem các trường đã xóa hoặc khôi phục lại các thông tin đã bị xóa trước đó.

**Mở Form, Report trong Windows SharePoint Services:** Windows SharePoint Services cho phép người dùng có thể mở trực tiếp các danh sách trong Access views (Form, Report, DataSheet) từ một site SharePoint. Khi bạn chọn một Access views, Microsoft Access 2007 mở và gọi tới các Form, Report hay Datasheet đó . Vì vậy nó giúp bạn có thể chạy trên SharePoint site mà không cần khởi động Microsoft Access 2007. Bạn cũng có thể chọn để mở một danh sách trong Access, và nếu cơ sở dữ liệu đó không tồn tại, nó sẽ tự động tạo mới và các Form, Report sẽ được kế thừa trong danh sách của bạn.

**Thu thập dữ liệu bằng cách sử dụng các biểu mẫu InfoPath và Outlook**

Bằng cách sử dụng các tính năng mới Data Collection, Microsoft Access 2007 có thể tự động tạo ra một định dạng Microsoft Office InfoPath 2007 hoặc định dạng HTML và nhúng nó trong phần body của một e-mail. Sau đó bạn có thể

gửi chúng đi tới những địa chỉ người nhận từ Contact Outlook của bạn, hoặc để tên của người nhận được lưu giữ trong một trường trong cơ sở dữ liệu Access.

The screenshot shows an Outlook email window titled "Add Assets Table Form" with the address bar "AccessDataCollection:[42C77321-7B15-4089-BBA...". The email header includes "To: someone@example.com", "Subject: Add Assets Table Form", and an attachment "Add Assets Table Form.xsn (5 KB)". The main body of the email contains the following text and form:

Fill out the form included in this message and send it back to me.

**Add Assets Table Form**

Item:   
Type any combination of numbers and letters up to 50 characters.

Description:   
Type any combination of numbers and letters.

Category:   
Select an item from the drop-down list.

Condition:   
Select an item from the drop-down list.

Insert a row

**Done? Click Submit to send your reply.**  
Note: To successfully reply to this mail you will need to use Microsoft Outlook 2007 and Microsoft InfoPath 2007.

Bạn lựa chọn xem để thu thập những thông tin mới hay cập nhật các thông tin hiện có. Người nhận biểu mẫu sau đó sẽ hoàn thiện và gửi lại. Microsoft Access 2007 tự động nhận dạng các biểu mẫu gửi đến và lưu vào cơ sở dữ liệu.

## **Export sang định dạng PDF và XPS**

Bây giờ bạn có thể xuất dữ liệu một định dạng file .PDF (Portable Document Format) hoặc .XPS (XML Paper Specification) dùng cho việc in ấn, Post bài hay e-mail. Việc xuất một Form, Report, hoặc Datasheet sang file có định dạng .Pdf hay .XPS cho phép bạn dễ dàng phân phối các thông tin mà vẫn giữ lại tất cả nhưng định dạng mà bạn thiết lập. Mà không cần người nhận phải cài đặt Access.

**Portable Document Format:** Một định dạng PDF là một file cố định giúp bảo vệ các định dạng của văn bản và chia sẻ tệp tin. Các định dạng PDF bảo đảm rằng khi các tệp tin được xem trực tuyến hoặc in ra, nó giữ lại chính xác các định dạng mà bạn đã thiết kế, và giữ cho tệp tin không bị sao chép hoặc sửa đổi. Các định dạng PDF cũng hữu ích cho các tài liệu mà sẽ được sao chép bằng cách sử dụng các phương pháp in ấn thương mại.

**XML Paper Specification:** Một XPS là một định dạng tệp tin điện tử, bảo vệ định dạng tài liệu và giúp chia sẻ tệp tin. Các định dạng XPS đảm bảo rằng khi các tệp tin được xem trực tuyến hoặc in ra, nó giữ lại chính xác các định dạng mà bạn đã định sẵn, giúp cho các dữ liệu trong tệp tin không bị sao chép hoặc thay đổi.

Bạn có thể xuất dữ liệu ra một file có định dạng .Pdf hay .XPS từ Microsoft Access 2007 chỉ sau khi bạn cài đặt các add-in.

## **Làm việc với dữ liệu bên ngoài được thực hiện dễ dàng hơn**

Các tính năng mới trong Microsoft Access 2007 làm cho việc Import hay Export trở nên dễ dàng hơn. Bạn có thể lưu lại một hoạt động Import hay Export và sau đó tái sử dụng ở thời gian tiếp theo. Với Import Spreadsheet Wizard sẽ cho

phép bạn ghi đè lên kiểu dữ liệu cho bởi Access, và bạn có thể Import, Export, và liên kết đến các định dạng tập tin Excel 2007 mới.

### **1.3.8. Cách tốt hơn để gỡ rối vấn đề**

Microsoft Office Diagnostics là một loạt các xét nghiệm chẩn đoán có thể giúp bạn khám phá lý do tại sao máy tính của bạn có crashing. Các xét nghiệm chẩn đoán có thể giải quyết một số vấn đề trực tiếp và có thể xác định cách thức mà bạn có thể giải quyết các vấn đề. Microsoft Office Diagnostics thay thế cho những tính năng sau đây của Microsoft Office 2003: Dò tìm, sửa chữa, và phục hồi ứng dụng Microsoft Office.

### **1.3.9. Nâng cao chất lượng các công cụ Proofing**

Sau đây là một số tính năng mới của công cụ kiểm tra chính tả.

Công cụ kiểm tra chính tả đã được làm chắc chắn hơn trong bộ Microsoft Office 2007. Dưới đây là ví dụ về sự thay đổi này:

Một vài tùy chọn kiểm tra toàn cục bây giờ sẽ mang tính toàn cục. Nếu bạn thay đổi một trong các tùy chọn này ở bất cứ ứng dụng nào trong bộ Office nó cũng sẽ thay đổi tương ứng trong các ứng dụng khác của bộ Office.

Ngoài ra để chia sẻ cùng một bộ từ điển, tất cả các chương trình có thể quản lý chúng bằng cách sử dụng cùng một hộp thoại.

## **Bài 2. Làm việc với Table**

Khi bạn tạo ra một cơ sở dữ liệu, bạn lưu trữ dữ liệu trong các bảng (dựa trên nội dung của các hàng và cột). Ví dụ, bạn có thể tạo một bảng Contacts lưu trữ danh sách gồm tên, địa chỉ, và số điện thoại, hoặc một bảng Products lưu trữ những thông tin về sản phẩm cần quản lý.

Bài viết này với mục đích hướng dẫn bạn làm thế nào để tạo ra một bảng và làm thế nào để thêm các trường rồi thiết lập thuộc tính cho những trường đó. Cũng như hướng dẫn bạn cách tạo ra những khóa chính cho các bảng đó.

Bởi vì có rất nhiều đối tượng khác nhau nhưng lại phụ thuộc lẫn nhau trong một Database. Vì vậy trước khi bắt tay vào thiết kế các bảng bạn nên xem xét một cách kỹ lưỡng các yêu cầu mình, để làm sao không bị dư thừa các bảng dữ liệu không có ích.

### **2.1. Tổng quát**

Mỗi Table là một đối tượng của CSDL mà bạn sử dụng để lưu trữ về một chủ đề cụ thể, chẳng hạn như nhân viên trong công ty, sản phẩm của một nhà máy nào đó, ... Một bảng gồm có nhiều trường và nhiều bản ghi.

Mỗi bản ghi chứa dữ liệu về một chủ thể mà bảng lưu trữ, chẳng hạn như các thông tin về một nhân viên. Một bản ghi thường tương ứng với một dòng trong bảng dữ liệu.

Mỗi trường chứa dữ liệu về một trong những thuộc tính của chủ thể trong bảng, chẳng hạn như tên hay địa chỉ e-mail của một nhân viên. Mỗi trường tương ứng với một cột trong bảng dữ liệu.

Một bản ghi bao gồm giá trị của nhiều trường chẳng hạn như Nhà sách Minh Lâm, nhà sách, [minhlam@yahoo.com](mailto:minhlam@yahoo.com). Giá trị của một trường thường được gọi là một thực thể.

ID	Company	First Name	Last Name
1	Company A	Anna	Bedecs
2	Company B	Antonio	Gratacos Solsona
3	Company C	Thomas	Axen

Trên đây là một bảng được mở ở chế độ Datasheet View

Trong đó:

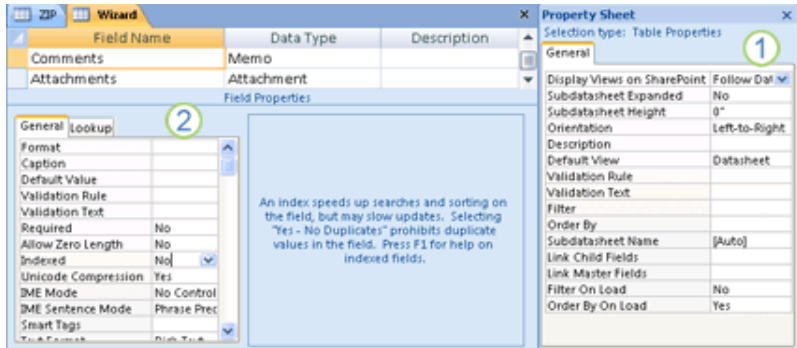
- (1) : Một bản ghi
- (2): Một trường
- (3): Giá trị một trường

Như vậy chúng ta có thể tổng kết lại như sau:

Một cơ sở dữ liệu có thể chứa nhiều bảng, mỗi bảng lưu trữ thông tin về một đối tượng, chủ thể khác nhau. Trong mỗi bảng có thể chứa một hoặc nhiều trường với những kiểu dữ liệu cũng khác nhau như kiểu chuỗi, số, ngày tháng, kiểu liên kết, ...

### 2.1.1. Thuộc tính của bảng và trường

Bảng và trường có những thuộc tính mà bạn có thể thiết lập để kiểm soát hành vi cũng như đặc điểm của chúng.



Đây là một bảng được mở ở chế độ Design View.

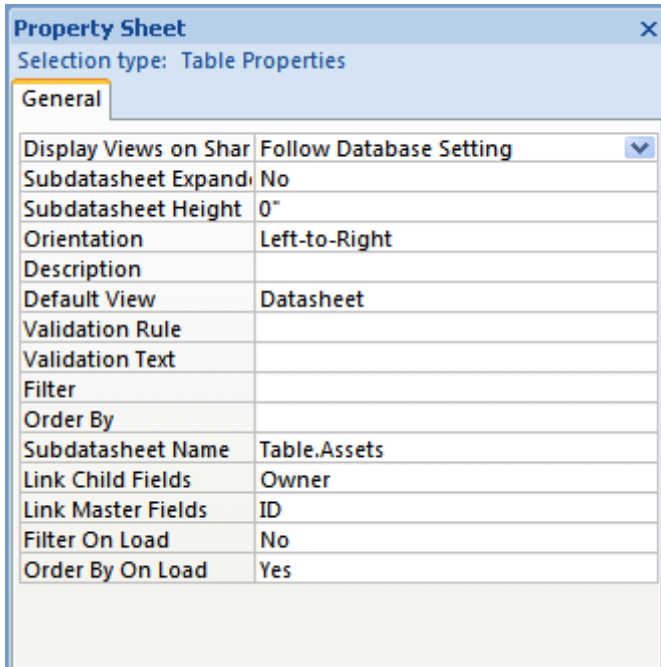
Trong đó:

(1): Thuộc tính của bảng

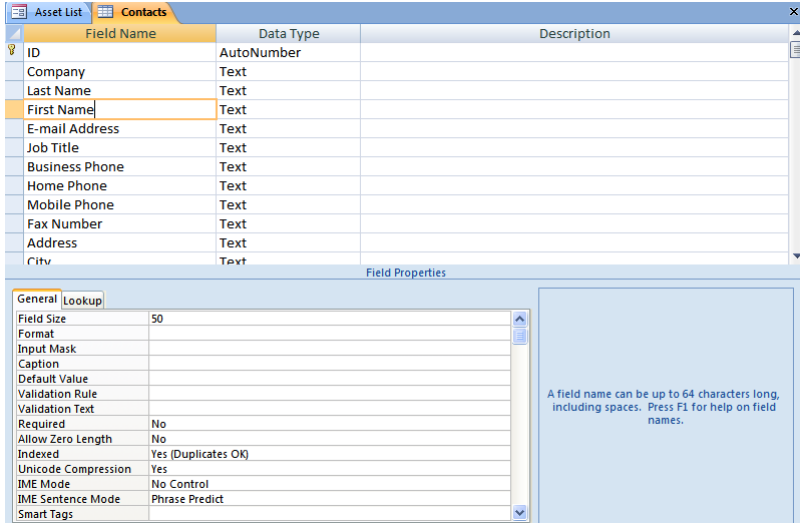
(2): Thuộc tính của trường

Trong một Database Access, thuộc tính của bảng ảnh hưởng trực tiếp, toàn diện tới sự thể hiện cũng như hành vi của bảng đó. Ở chế độ Design View, thuộc tính của bảng được thiết lập Pane Property Sheet. Ví dụ, bạn có thể thiết lập thuộc tính Default View cho bảng để mỗi lần hiển thị nó sẽ hiển thị theo chế độ đó.





Một thuộc tính của trường được áp dụng cho một trường cụ thể trong bảng và xác định một trong những tính chất đặc trưng của trường đó. Bạn có thể thiết lập thuộc tính cho trường ở chế độ hiển thị Datasheet View, cũng như ở chế độ Design View bằng cách sử dụng Pane Field Properties.



### 2.1.2. Kiểu dữ liệu

Mỗi trường có một kiểu dữ liệu nhất định. Một kiểu dữ liệu của một trường cho biết kiểu dữ liệu mà trường đó lưu trữ chẳng hạn như trường dữ liệu kiểu Text sẽ cho phép lưu trữ các ký tự, hay kiểu Date/Time cho phép lưu trữ dữ liệu kiểu ngày tháng, ...

Field Name	Data Type	Description
Comments	Memo	
Attachments	Attachment	

Field Properties

Kiểu dữ liệu của trường thực chất là thuộc tính của trường đó, tuy nhiên nó vẫn khác thuộc tính ở một số điểm sau đây:

- Bạn có thể thiết lập kiểu dữ liệu cho trường trong lưới thiết kế bảng nhưng không thể làm việc đó trong Pane Field Properties.
- Một kiểu dữ liệu của trường xác định những thuộc tính mà trường đó có thể có.

Chú ý:

Bạn có thể tạo ra một trường mới ở chế độ Datasheet view trong Access bằng cách nhập dữ liệu trực tiếp dữ liệu vào một cột mới. Khi bạn tạo ra một trường mới bằng cách này Access sẽ tự động định nghĩa kiểu dữ liệu cho trường dựa trên những giá trị mà bạn nhập vào cột đó. Nếu như không có một kiểu dữ liệu nào phù hợp cho rất nhiều giá trị trên cột đó Access sẽ hiểu đó là dữ liệu kiểu Text, nếu cần thiết bạn hoàn toàn có thể thay đổi lại những kiểu dữ liệu này.

Dưới đây là bảng mô tả về kiểu dữ liệu mà Access tự động thiết lập cho trường tương ứng với dữ liệu bạn nhập vào.

<b>Dữ liệu có thể nhập vào</b>	<b>Kiểu dữ liệu mà Access có thể tự định nghĩa</b>
Minh Lâm	Text
<a href="http://www.minhlam.com">http://www.minhlam.com</a>	Hyperlink
1	Number, Long Integer
50,000	Number, Long Integer
50,000.99	Number, Double
50000.389	Number, Double
12/67	Date/Time
December 31, 2006	Date/Time
10:50:23	Date/Time

10:50 am	Date/Time
17:50	Date/Time
\$12.50	Currency
21.75	Number, Double
123.00%	Number, Double
3.46E+03	Number, Double

### **2.1.3. Quan hệ giữa các bảng**

Mặc dù mỗi bảng lưu trữ dữ liệu về một chủ thể nhất định, nhưng xét trong một Database thì các chủ thể này lại có mối quan hệ mật thiết với nhau. Ví dụ, một cơ sở dữ liệu có thể chứa:

Một bảng customers là danh sách khách hàng công ty của bạn.

Một bảng products chứa danh sách những sản phẩm mà công ty bạn sản xuất ra.

Một bảng orders chứa những đơn đặt hàng mà khách hàng mua sản phẩm.

Vì khách hàng trong orders mua sản phẩm của công ty sản xuất ra nên những khách hàng đó phải là khách hàng trong bảng customers, và những sản phẩm cũng phải là những sản phẩm trong bảng Products.

Chính vì những mối liên hệ đó mà bạn cần phải làm thế nào để có thể liên kết các bảng này lại với nhau. Nhờ việc liên kết này mà bạn có thể dễ dàng kiểm soát sự toàn vẹn dữ liệu trong một Database.

Để liên kết các bảng lại với nhau bạn cần phải tạo ra các mối liên kết (quan hệ). Một mối quan hệ là một kết nối Logical giữa hai bảng với nhau thông qua một trường có chung ở hai bảng.

Ví dụ trong bảng Customers sẽ có trường Id xác định tính duy nhất của khách hàng đó và trong bảng Orders cũng có trường Id\_Kh trường này lưu trữ Id của khách hàng mua sản phẩm, hai trường này được gọi là trường liên kết giữa bảng Customers và bảng Order.

#### **2.1.4. Khóa**

Những trường mà là một phần của mối quan hệ trong bảng gọi là trường khóa. Một khóa thường bao gồm một hoặc nhiều trường khác nhau. Trong bảng có thể có hai loại khóa sau đây:

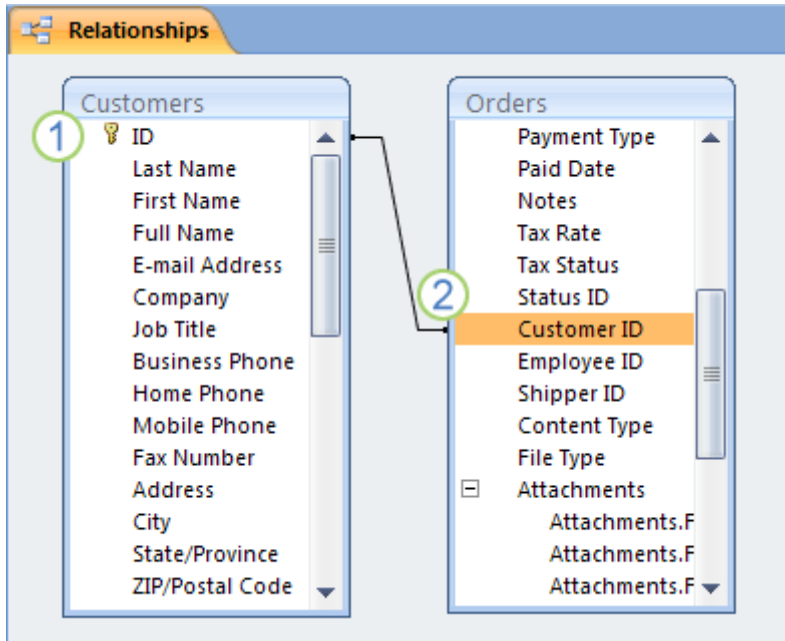
**Primary key:** Một bảng chỉ có duy nhất một khóa Primary key. Một khóa Primary key có thể bao gồm một hoặc nhiều trường kết hợp với nhau xác định tính duy nhất của mỗi bản ghi lưu trữ trong bảng.

Thông thường, có một số để xác định tính duy nhất của một bản ghi, chẳng hạn như số ID, số sê-ri, hoặc mã số sẽ được chọn làm khóa. Ví dụ, bạn có một bảng Customers mỗi khách hàng được xác định duy nhất thông qua số Id của khách hàng đó. Như vậy trường Id chính là trường khóa của bảng Customers. Khi một chính khóa (Primary key ) chứa nhiều hơn một trường thì nó là sự kết hợp của những trường đã tồn tại, kết hợp với nhau để tạo ra một giá trị duy nhất. Ví dụ, bạn có thể kết hợp các trường Ho, Ten, Ngay\_Sinh để tạo thành một trường khóa cho bảng People.

**Foreign key:** Một bảng có thể có một hoặc nhiều Foreign key. Một Foreign key chứa các giá trị mà tương ứng với những giá trị đó là những giá trị của trường khóa chính ở

những bảng khác. Ví dụ, bạn có thể có một bảng Orders, trong đó mỗi đơn hàng có một số Id của khách hàng đặt đơn hàng đó, Id xác định tính duy nhất của một bản ghi trong bảng Customers. Như vậy có thể nói trường Id khách hàng trong bảng đơn hàng là một Foreign key (ngoại khóa). Sự tương ứng của các giá trị giữa những trường khóa là nền tảng của một mối quan hệ giữa các bảng với nhau. Bạn có thể sử dụng mối quan hệ của bảng để kết hợp dữ liệu từ những bảng liên quan. Ví dụ, giả sử bạn có một bảng Customers và một bảng Orders, mỗi bản ghi được xác định duy nhất bởi trường khóa Id (trường khóa chính).

Để liên kết mỗi đơn hàng với một khách hàng, bạn thêm một trường ngoại khóa vào bảng Orders tương ứng với trường Id trong bảng Customers, và sau đó tạo mối quan hệ giữa hai khóa này. Khi bạn thêm một bản ghi vào Orders, bạn sử dụng giá trị cho trường Id khách hàng được lấy từ bảng Customers. Khi nào bạn cần xem bất kỳ thông tin nào về một đơn đặt hàng của khách hàng, bạn nên sử dụng các mối quan hệ để xác định được những dữ liệu từ bảng Customers tương ứng với các bản ghi trong bảng Orders.



Mối quan hệ giữa hai bảng, được hiển thị trong cửa sổ Relationships.

(1): Là trường primary key (khóa chính), xác định bởi biểu tượng khóa bên cạnh tên trường.

(2): Là trường foreign key (ngoại khóa)

### 2.1.5. Lợi ích của việc sử dụng relationships

Việc lưu trữ dữ liệu được phân chia trong những bảng liên quan có những lợi ích sau đây:

Sự nhất quán: Vì mỗi nội dung của dữ liệu được ghi chỉ một lần, trong một bảng, có ít khả năng xảy ra sự mâu thuẫn, không minh bạch. Chẳng hạn, bạn lưu trữ tên của một khách hàng chỉ một lần, trong bảng khách hàng, hơn là lưu trữ nó nhiều lần (Và tiềm ẩn những mâu thuẫn) trong bảng chứa dữ liệu đơn hàng.

Hiệu quả: Ghi dữ liệu ở một chỗ có nghĩa là bạn sử dụng ít không gian trên đĩa cứng hơn. Thêm vào đó, bảng ít dung lượng sẽ cho phép truy cập, truy vấn dữ liệu nhanh hơn là các bảng lớn. Cuối cùng, nếu bạn không sử dụng những bảng riêng biệt để lưu trữ thông tin về các chủ thể riêng biệt, bạn sẽ phải theo dõi, quản lý cả những giá trị Null dẫn đến sự dư thừa dữ liệu trong các bảng. Cả hai lý do đó có thể sẽ tiêu phí không gian lưu trữ và ngăn cản quá trình làm việc với dữ liệu của bạn.

Sự hiểu biết: Quá trình thiết kế một Database sẽ dễ hiểu hơn nếu một chủ thể được phân rã hợp lý thành những bảng riêng biệt.

### **2.1.6. Trước khi bạn bắt đầu**

Trước khi bạn bắt đầu tạo ra các bảng trong Microsoft Office Access 2007, có vài sự khác biệt so với những phiên bản trước đó của Access mà bạn cần chú ý.

Bảng và trường Template đã được thay thế cho Table Wizard trong các phiên bản trước, bạn đã sử dụng Table Wizard để tạo ra một Table rất đơn giản với việc chỉ phải trả lời vài câu hỏi. Trong Microsoft Access 2007, Table Wizard sẽ được thay thế bằng Table và các trường Template (mẫu).

Datasheet trong Microsoft Access 2007, bạn có thể tạo ra và sửa đổi các bảng, trường trong khi đang việc làm việc ở chế độ hiển thị Datasheet View

## **2.2. Tạo mới một bảng**




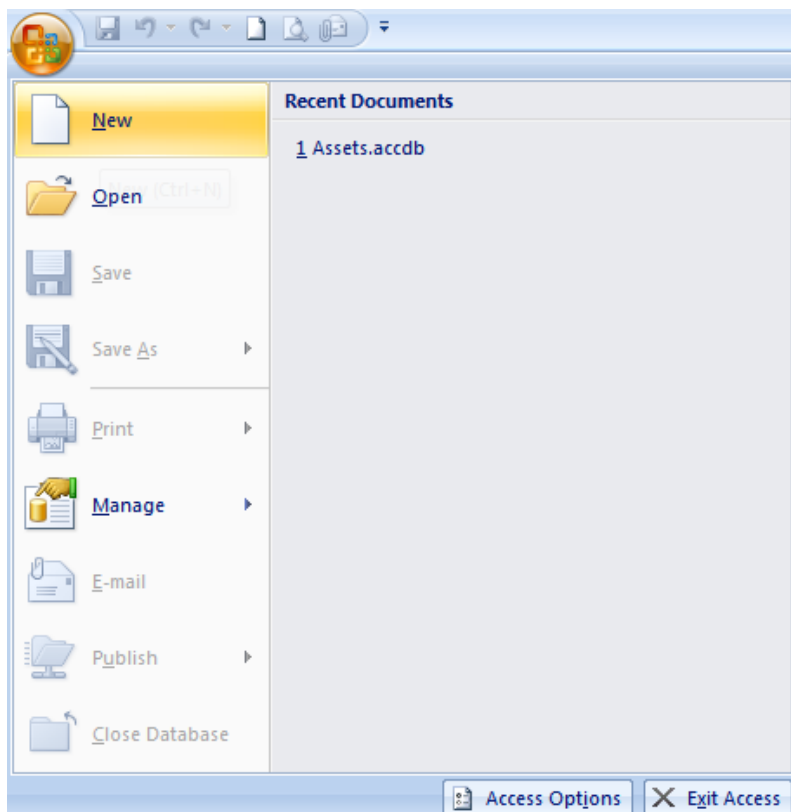
Một cơ sở dữ liệu đơn giản, như danh sách liên lạc, có thể sử dụng chỉ duy nhất một bảng. Tuy nhiên nhiều cơ sở dữ liệu, phải sử dụng rất nhiều bảng.

Khi bạn tạo ra một cơ sở dữ liệu mới, đồng thời bạn cũng đã tạo ra một File trên máy tính, nó đóng vai trò như một Container (nơi chứa) tất cả các đối tượng trong cơ sở dữ liệu, trong đó có các bảng dữ liệu.

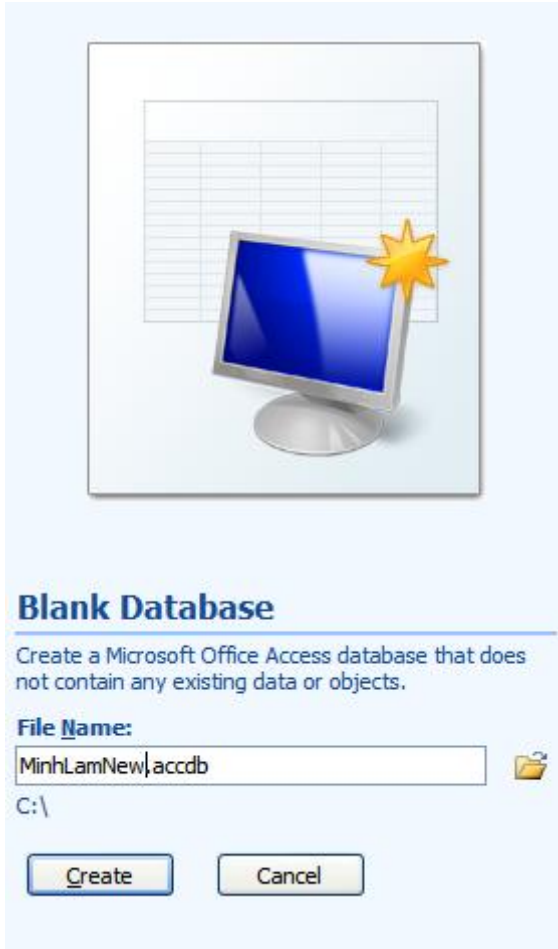
Bạn có thể tạo ra một bảng bằng việc tạo ra một cơ sở dữ liệu mới, hoặc cũng có thể tạo ra bằng việc chèn một bảng vào trong một cơ sở dữ liệu đã tồn tại, hay bằng cách Import hoặc liên kết tới các bảng từ những nguồn dữ liệu khác. Khi bạn tạo mới một DataBase rỗng, một bảng mới chưa có nội dung sẽ được chèn vào đó. Bạn có thể nhập dữ liệu vào cũng như bắt đầu định nghĩa các trường cho bảng.

### **2.3.1. Tạo bảng mới trong Database mới.**

Nhấp chuột chọn nút Microsoft Office Button , sau đó nhấp New.



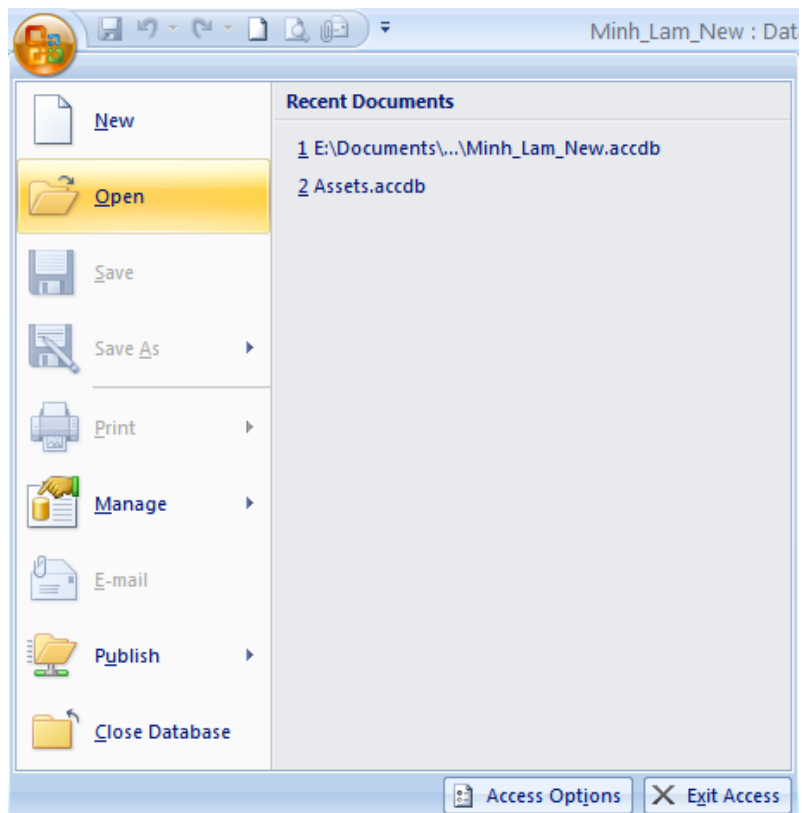
Trong hộp nhập File Name, gõ vào tên cho Database mới. Để thay đổi đường dẫn lưu Database bạn nhấp chuột chọn biểu tượng Folder để chọn lại. Cuối cùng nhấp Create.



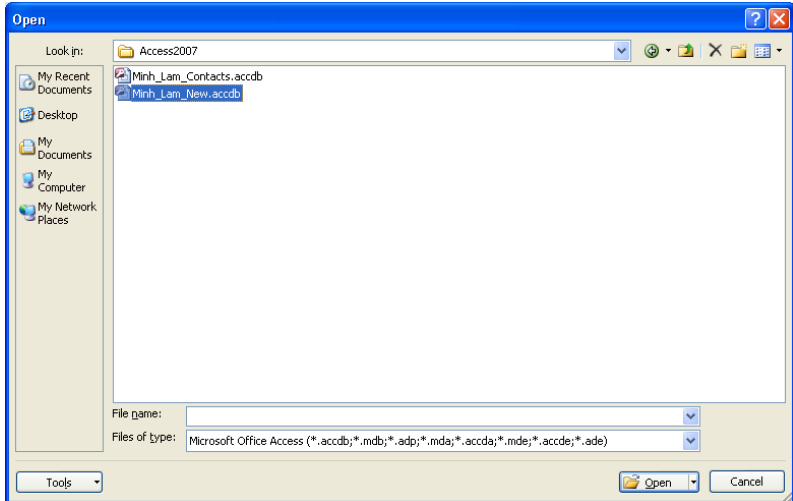
Một Database mới được mở và một Table mới với tên là Table1 được tạo ra và mở ở chế độ Datasheet View.

### 2.3.2. Tạo bảng mới trong Database đã tồn tại

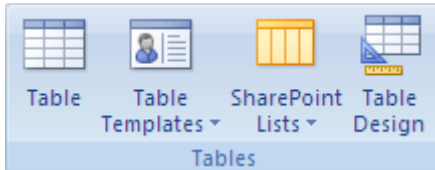
Nhấp chuột chọn nút Microsoft Office Button , sau đó chọn Open.



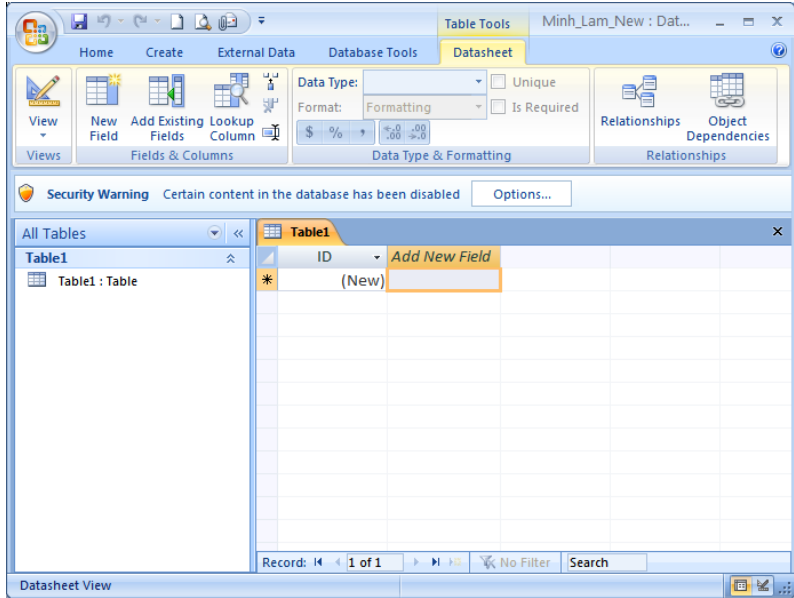
Hộp thoại Open xuất hiện chọn tới Database cần mở rồi nhấp Open.



Trên Tab Create trong nhóm Table chọn Table.



Một bảng mới được Insert vào Database và bảng này được mở ở chế độ Datasheet View.



### 2.3.3. Sử dụng bảng Template để tạo bảng

Trong những phiên bản trước đây của Access, bạn sử dụng Table Wizard để tạo nhanh một bảng từ những bảng và trường mẫu. Thay cho chức năng này trong Microsoft Access 2007, bạn có thể sử dụng các bảng và trường Templates.

Một bảng mẫu là một bảng trống không chứa dữ liệu, bạn có thể bắt đầu luôn hoặc chỉnh sửa cho phù hợp với yêu cầu của mình. Microsoft Access 2007 cung cấp những mẫu Table Template dưới đây:

**Contacts:** Là bảng sử dụng để quản lý thông tin liên lạc trong kinh doanh. Như địa chỉ Email, Website, các File đính kèm,...

**Tasks:** Là bảng dùng để theo dõi công việc, bao gồm các trường sử dụng cho những phiếu đính kèm.

Issues: Là bảng sử dụng để theo dõi những hoạt động, bao gồm các sử dụng cho những phiếu đính kèm và một trường kiểu Memo lưu trữ lịch sử sửa đổi giá trị trên các trường trong bảng.

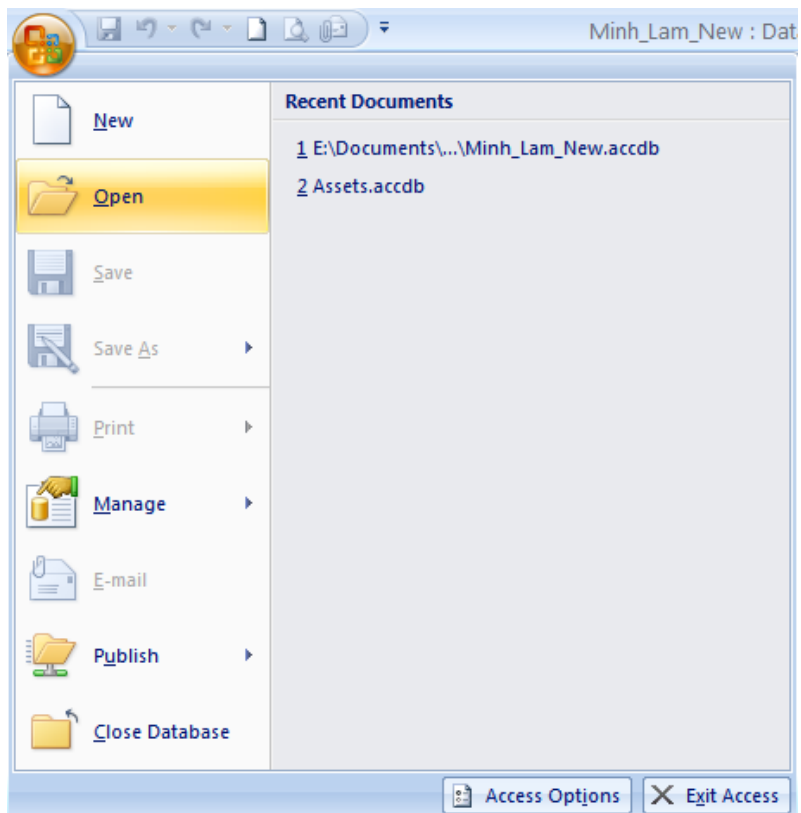
Events: Là bảng dùng để quản lý các sự kiện bao gồm một trường Rich Text Memo và một trường cho những phiếu đính kèm.

Assets: Là bảng dùng để quản lý tài sản của doanh nghiệp, bao gồm 2 trường tiền tệ để có thể theo dõi được giá trị khấu hao của tài sản.

Sau khi bạn tạo ra một bảng bằng cách sử dụng bảng Template, có thể bạn sẽ muốn thêm vào những trường mới bằng cách sử dụng các trường Template. Một trường Template là trường đã được định nghĩa sẵn, bạn có thể thêm vào bất cứ bảng nào ở chế độ hiển thị Datasheet View.

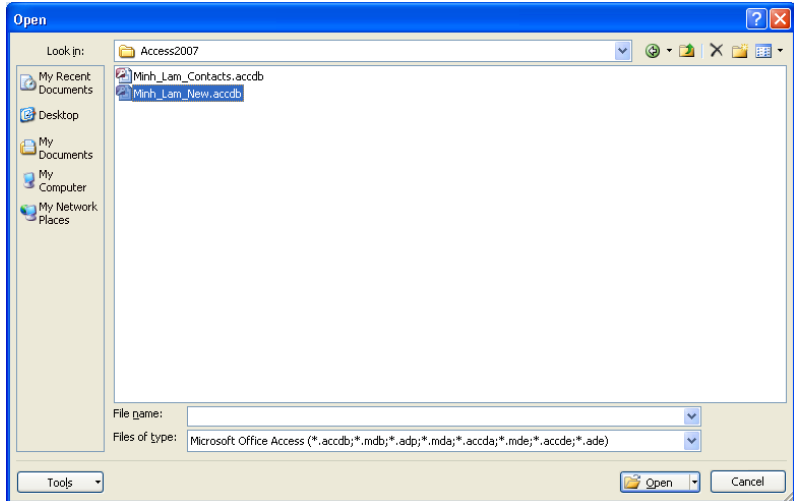
#### **2.3.4. Tạo bảng mới sử Template**

Nhấp chọn nút Microsoft Office Button , sau đó chọn Open.

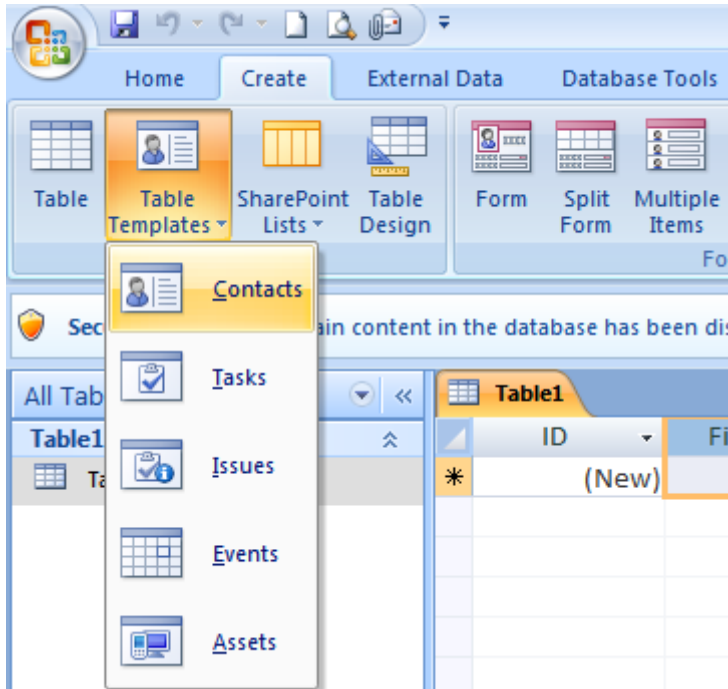


Hộp thoại Open xuất hiện chọn tới Database cần mở rồi nhấp Open.





Trên Tab Create, trong nhóm Tables chọn nút Table Templates, trong hộp thoại đổ xuống chọn một bảng mẫu cần tạo



Một bảng mới được thêm vào Database dựa trên bảng mẫu mà bạn đã chọn.

### 2.3.5. Thiết lập khóa chính cho bảng

Nếu không có một lý do thật đặc biệt thì khi thiết kế ra một bảng dữ liệu bạn cần phải chỉ rõ một khóa chính. Access tự động tạo ra Index cho khóa chính, nó có thể cải thiện được tốc độ truy cập cũng như thao tác với dữ liệu. Access cũng chắc chắn rằng mỗi bản ghi thì trường khóa chính luôn chứa dữ liệu và giá trị này luôn luôn là duy nhất. Nếu không có trường này sẽ chẳng có cách nào đáng tin cậy để phân biệt một hàng cụ thể nào đó với những hàng khác trong bảng.

Khi bạn tạo ra một bảng mới trong chế độ hiển thị Datasheet View, Access sẽ tự động tạo ra một khóa chính và định nghĩa nó với tên là ID và kiểu dữ liệu là AutoNumber.

Ở chế độ hiển thị Design View, bạn có thể thay đổi, loại bỏ khóa chính, hoặc thiết lập cho một bảng chưa có khóa chính.

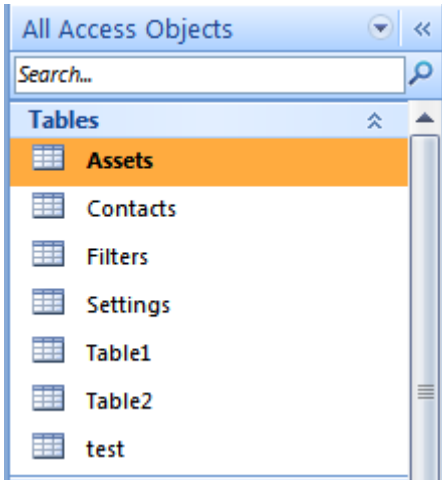
### ***Xác định trường đặt làm khóa chính***

Đôi khi, bạn có thể tự xác định được trường mà mình muốn sử dụng làm khóa chính. Chẳng hạn, bạn có thể có những số ID dành cho mỗi nhân công. Nếu bạn tạo ra một bảng để theo dõi thông tin về những người công nhân, bạn có thể quyết định sử dụng ID làm khóa chính cho bảng đó. Hoặc là sự kết hợp giữa ID của người công nhân với Id phòng ban để tạo nên tính duy nhất. yêu cầu rằng bạn sử dụng cả trường này để tạo nên khóa chính. Một hoặc nhiều trường muốn sử dụng làm trường khóa phải đảm bảo một vài nguyên tắc sau đây:

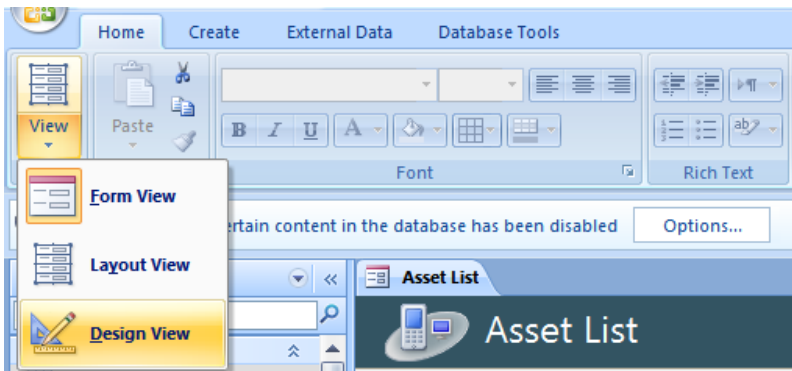
- Mỗi bản ghi phải có một hoặc nhiều trường kết hợp để tạo ra một giá trị duy nhất cho mỗi bản ghi.
- Một trường hoặc sự kết hợp giữa nhiều trường phải không bao giờ chứa giá trị rỗng (luôn luôn có giá trị)
- Những giá trị không thay đổi.
- Nếu không có dữ liệu thích hợp tồn tại để tạo ra khóa chính bạn có thể tạo ra một trường mới chỉ để làm khóa chính. Khi bạn tạo ra một trường mới sử dụng làm khóa chính bạn hãy thiết lập kiểu dữ liệu cho nó là AutoNumber. Với kiểu dữ liệu này trường sẽ nhận dữ liệu kiểu số, nằm trong dãy số nguyên liên tiếp, và hoàn toàn không trùng lặp. Hơn nữa trường này sẽ tự động tăng khi thêm một bản ghi mới giúp bạn không cần phải quan tâm việc tạo ra dữ liệu duy nhất cho trường khóa chính.

### ***Thiết lập hoặc thay đổi khóa chính***

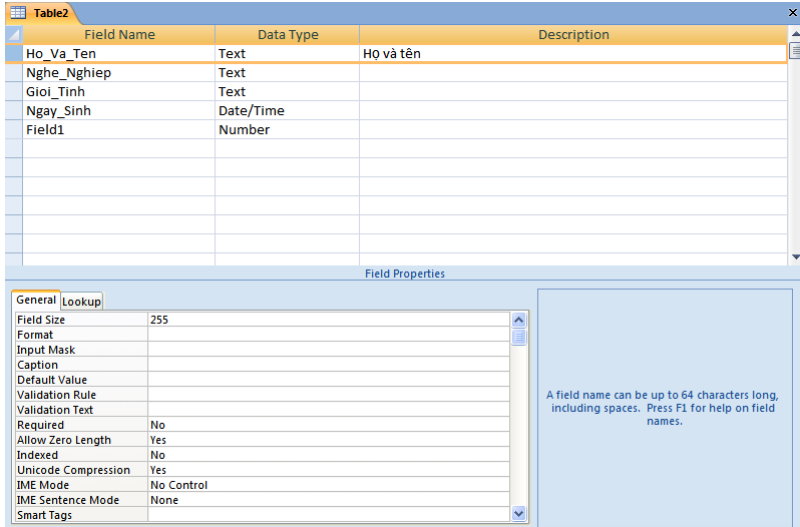
Chọn bảng có khóa chính mà bạn muốn thiết lập hoặc thay đổi.



Trên Tab Home chọn nhóm Views, nhấp chọn View trong hộp thoại dropdown này chọn Design View.

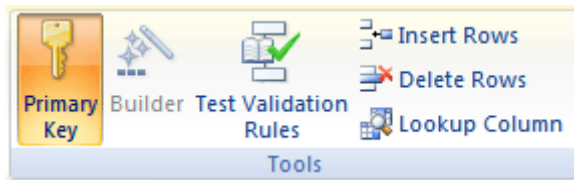


Bảng xuất hiện ở chế độ Design View, hãy chọn trường (hoặc những trường) mà bạn muốn sử dụng làm khóa chính.



Để lựa chọn một trường hãy nhấp chọn vào cột Row selector (cột đầu tiên màu xanh bên trái mỗi dòng) của trường đó. Nếu muốn chọn nhiều trường bạn hãy giữ phím Ctrl và lần lượt chọn tới các trường cần thiết lập.

Trên Tab Table trong nhóm Tools chọn Primary Key.



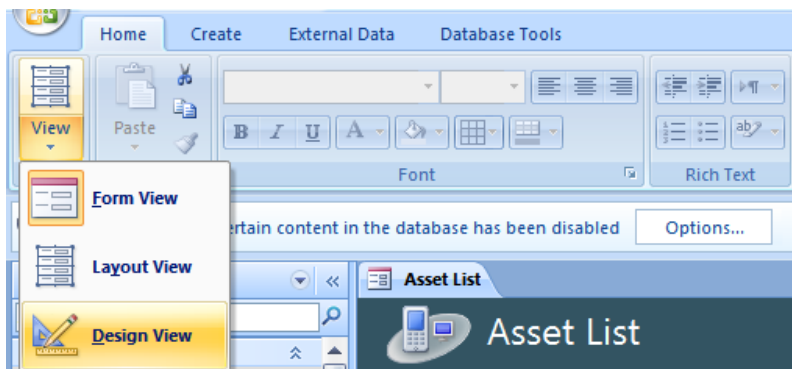
Một biểu tượng hình chìa khóa sẽ xuất hiện bên trái của những trường mà bạn vừa thiết lập khóa chính

	Field Name	Data Type
🔑	Ho_Va_Ten	Text
	Nghe_Nghiep	Text
	Gioi_Tinh	Text
	Ngay_Sinh	Date/Time
	Field1	Number

### ***Loại bỏ khóa chính***

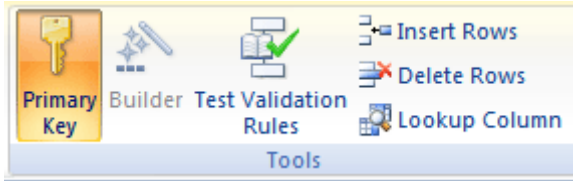
Chọn bảng cần loại bỏ khóa chính trong Navigation Pane.

Tên Tab Home tìm tới nhóm Views nhấp chuột chọn nút View, trong danh sách đổ xuống này chọn Design View.



Nhấp chuột chọn tới trường hoặc danh sách trường đang là những trường khóa chính (cách làm tương tự phần trước)

Trên Tab Design tìm tới nhóm Tools chọn Primary Key.



Biểu tượng chìa khóa bên phía trái của các cột này sẽ được loại bỏ.

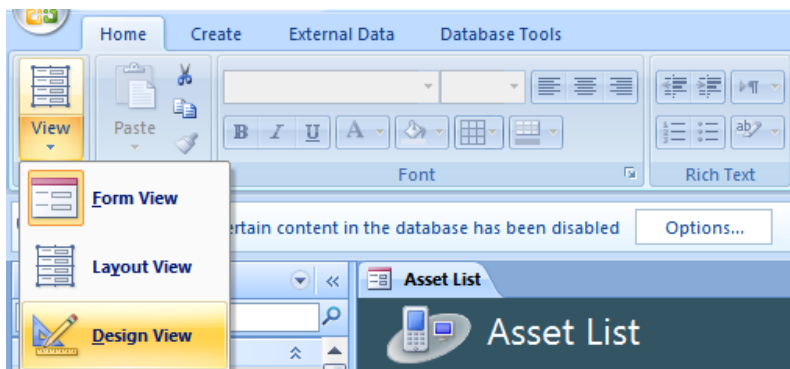
**Chú ý:** Khi bạn lưu trữ một bảng mới mà không được thiết lập khóa chính, Access sẽ nhắc cần phải tạo mới một trường làm khóa chính. Nếu bạn chọn Yes, Access tạo ra một trường có tên ID và gán cho nó kiểu dữ liệu là AutoNumber để cung cấp một giá trị duy nhất cho mỗi bản ghi. Trong trường hợp bảng đã có sẵn một trường kiểu AutoNumber, Access sẽ chọn luôn trường này làm khóa chính. Nếu bạn chọn No trên hộp thông báo Access sẽ không thêm trường và không thiết lập khóa chính cho bảng.

### 2.3.6. Thiết lập thuộc tính cho bảng

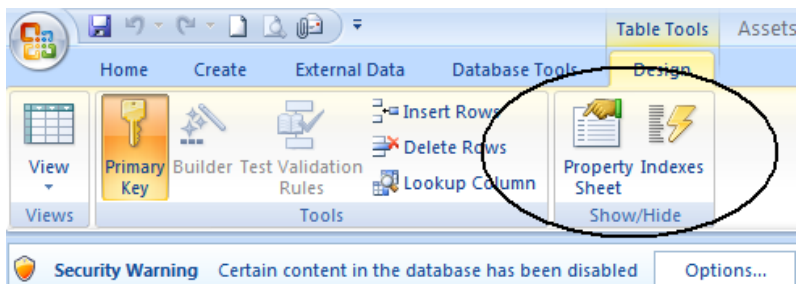
Ngoài việc thiết lập thuộc tính cho các trường bạn còn có thể thiết lập thuộc tính chung cho bảng hoặc tất cả các bản ghi.

Chọn tới bảng cần thiết lập thuộc tính

Trên Tab Home, tìm tới nhóm Views, nhấp chọn Views trong hộp thoại để xuống bạn chọn Design View.

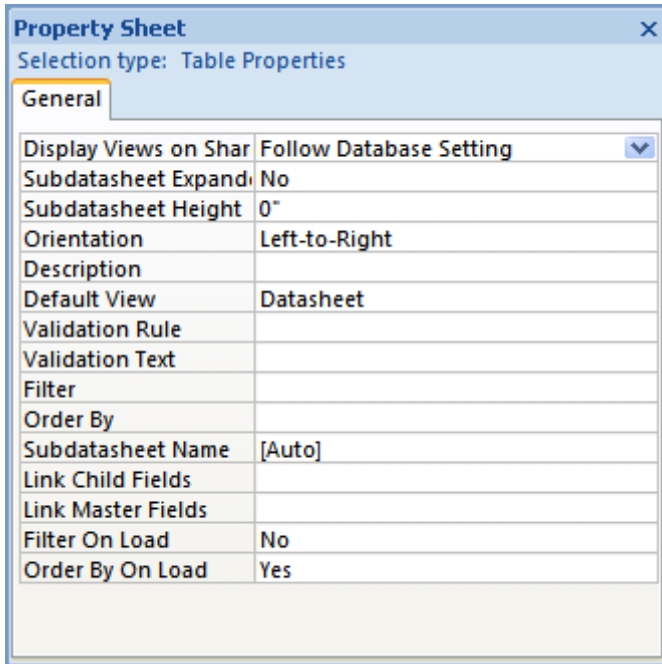


Trên Tab Design, tìm tới nhóm Show/Hide, nhấp chọn Property Sheet.



Property Sheet sẽ xuất hiện như hình dưới đây:





Trên Pane Property Sheet nhấp chọn Tab General.

Chọn cột bên trái để tìm tới thuộc tính của bảng và cột bên phải để thiết lập giá trị.

### 2.3. Thêm trường vào bảng

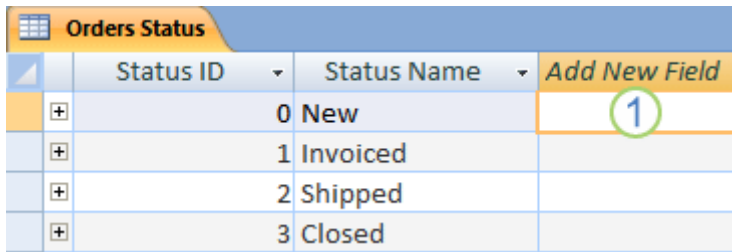
Bạn lưu trữ từng phần dữ liệu mà bạn muốn theo dõi trong một trường. Ví dụ, ở bảng Contacts bạn tạo ra danh sách trường để lưu lần lượt các thông tin như Last Name, First Name, Telephone Number, and Address. Hay trong bảng Procedures là các trường Product Name, Product ID, và Price.

Trước khi tạo ra các trường, dữ liệu được phân chia thành những phần nhỏ. Đó là cách dễ nhất để có thể xử lý dữ liệu được linh hoạt. Ví dụ, thay vì tạo ra một trường Full Name lưu trữ đầy đủ họ tên của một người nào đó, chúng ta nên tạo

ra hai trường riêng biệt là First Name và Last Name. Sau đó thì có thể dễ dàng tìm kiếm, sắp xếp theo từng trường hoặc theo cả hai trường.

### **2.4.1. Thêm trường khi đang nhập dữ liệu**

Khi bạn tạo một bảng mới hoặc mở một bảng đã tồn tại ở chế độ hiển thị Datasheet View, bạn có thể thêm một trường vào bảng bằng cách nhập dữ liệu vào cột Add New Field trên Datasheet.



Orders Status			
	Status ID	Status Name	Add New Field
+	0	New	1
+	1	Invoiced	
+	2	Shipped	
+	3	Closed	

Bảng dữ liệu được tạo mới hoặc được mở ở chế độ Datasheet View.

(1): Nhập dữ liệu vào cột Add New Field.

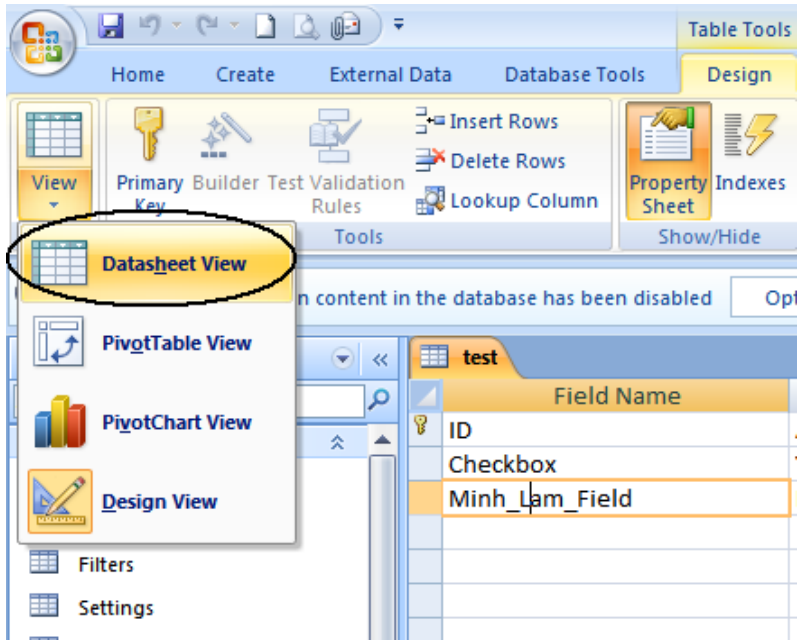
Trên phần tiêu đề của cột Add New Field hãy nhập vào tên trường mà bạn muốn tạo, tên này sẽ mô tả trường và giúp bạn dễ dàng nhận ra nó hơn.

Cuối cùng nhập dữ liệu vào trường đó.

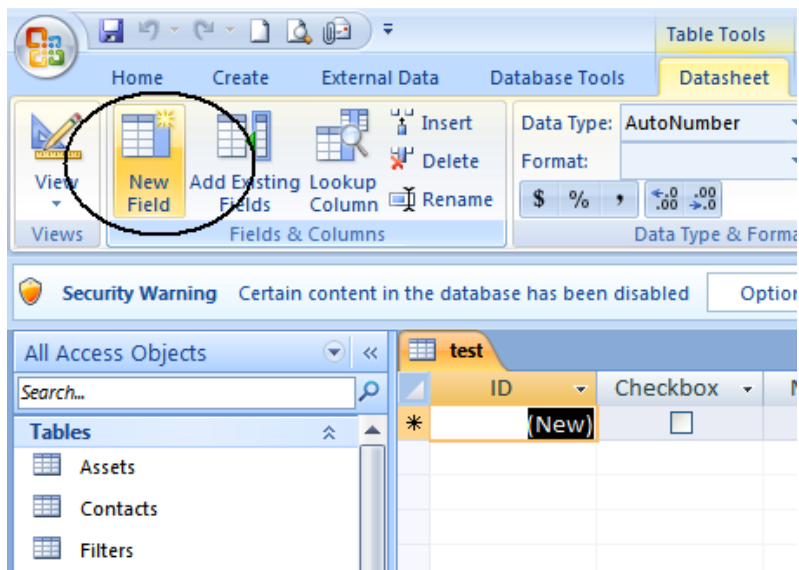
### **2.4.2. Thêm trường sử dụng trường Template**

Trong một số trường hợp thật dễ dàng lựa chọn một trường phù hợp với yêu cầu của bạn trong danh sách các trường đã được thiết lập sẵn thay vì việc bạn phải tự tạo ra nó. Bạn có thể sử dụng Field Templates task pane để chọn một trong những trường mẫu. Các trường mẫu này đã được thiết lập sẵn các thuộc tính cần thiết. Ví dụ như tên trường, kiểu dữ liệu, các thuộc tính định dạng, và nhiều thuộc tính khác.

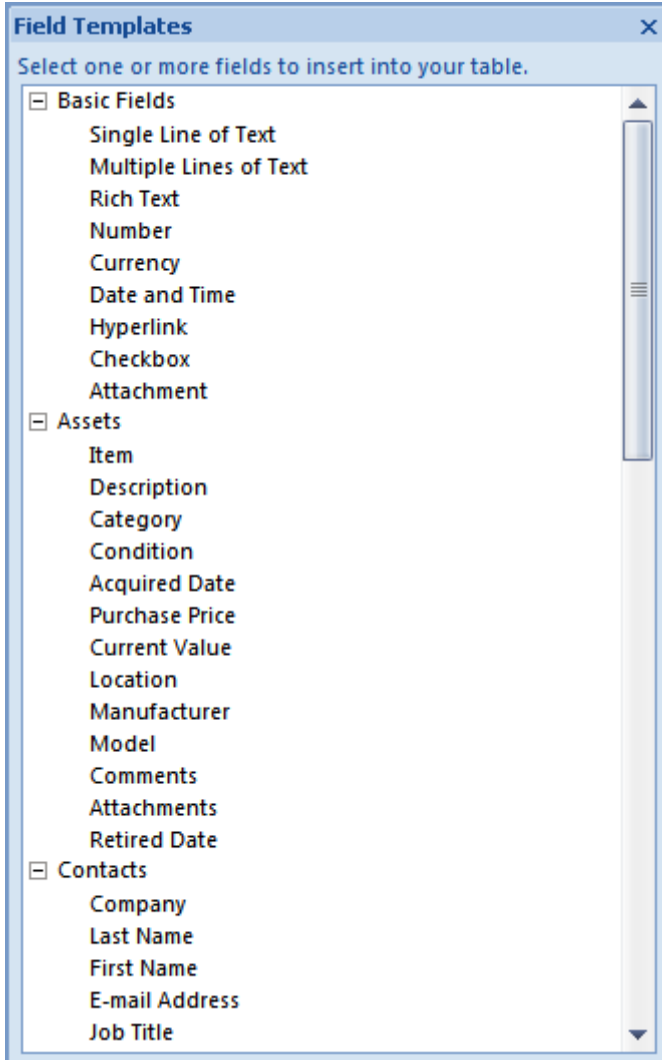
Trên Tab Home, tìm tới nhóm Views, nhấp chọn View, trên hộp thoại đổ xuống bạn chọn Datasheet View.



Trên Tab Datasheet, tìm tới nhóm Fields & Columns, nhấp chọn New Field.



Một thanh Pane có tên Field Templates xuất hiện bên phải của màn hình.

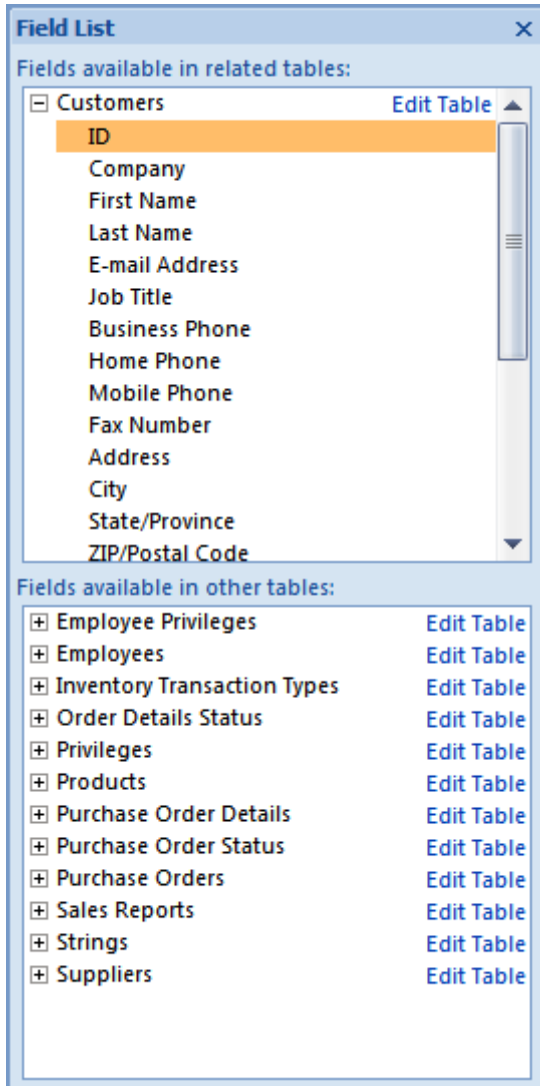



Chọn một hoặc nhiều trường trong pane này rồi sau đó kéo thả vào nơi mà bạn muốn thêm mới trường trên bảng.

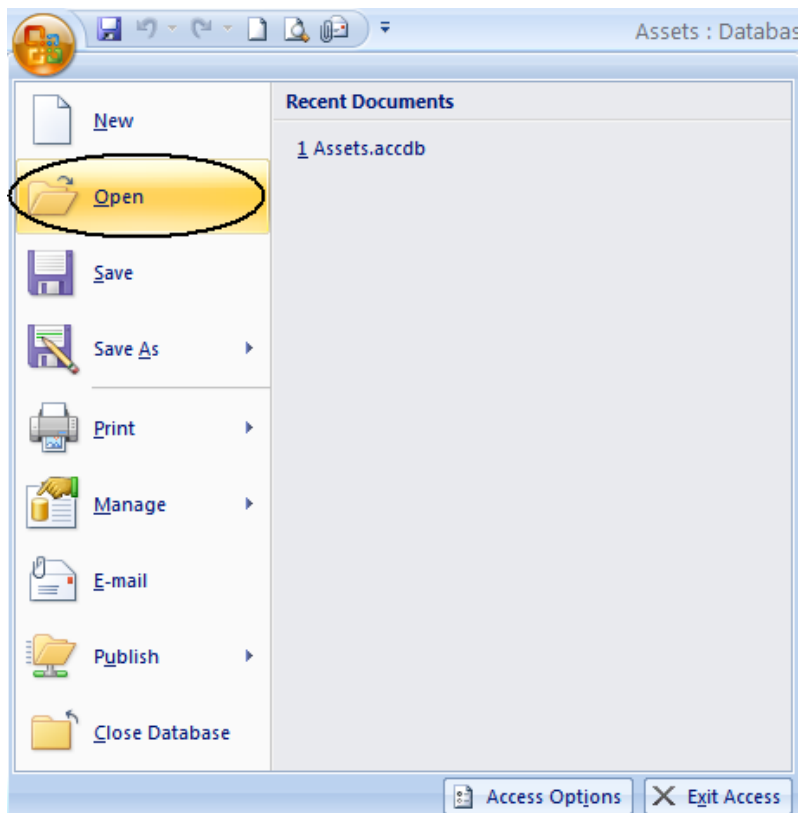
### **2.4.3. Thêm trường từ trường của bảng khác**

Nếu cơ sở dữ liệu của bạn đã có sẵn các Table, bạn có thể thêm trường vào một bảng mới từ danh sách các trường của một trong những bảng này, chúng được liệt kê trong Pane Field List. Pane Field List liệt kê tất cả các bảng trong cơ sở dữ liệu, chi tiết trong các bảng này là danh sách trường.

Nếu bảng mà bạn thêm trường mới có mối quan hệ với các bảng khác, thì các trường quan hệ sẽ được liệt kê đầu tiên trong Field List Pane

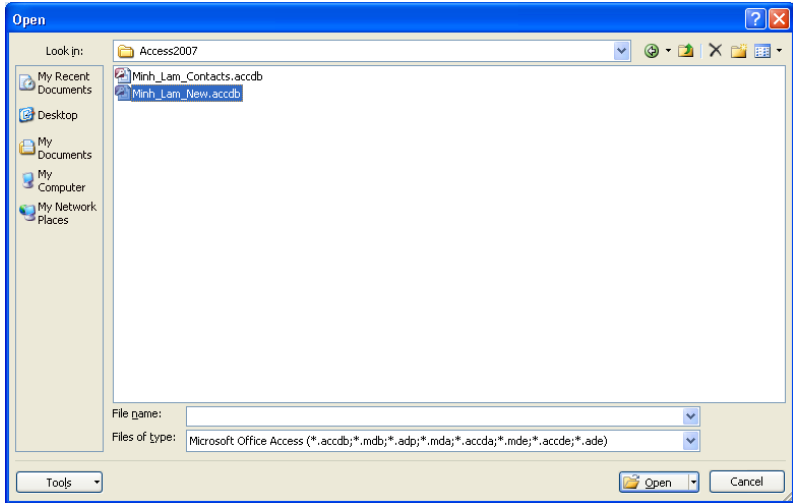


Nhấp chọn nút Microsoft Office Button , tiếp theo chọn Open.



Trong hộp thoại Open, chọn tới Database bạn cần làm việc.

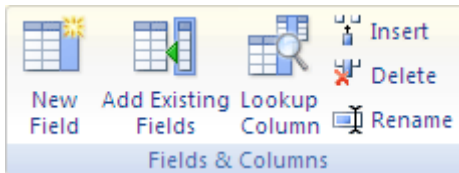




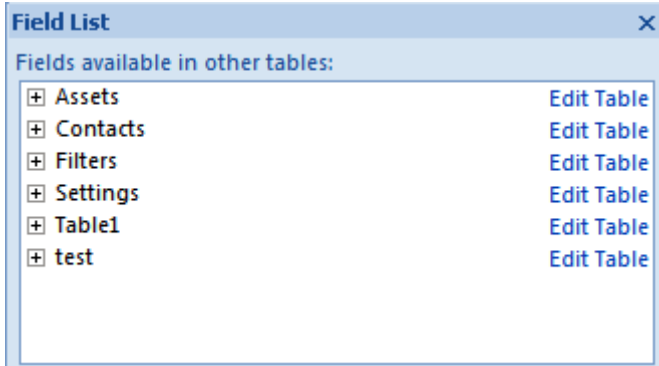
Trong Pane Navigation, nhấp đúp chuột vào bảng cần thêm trường mới.

Bảng được mở ở chế độ Datasheet View.

Trên Tab Datasheet, trong nhóm Fields & Columns nhấp chọn Add Existing Fields

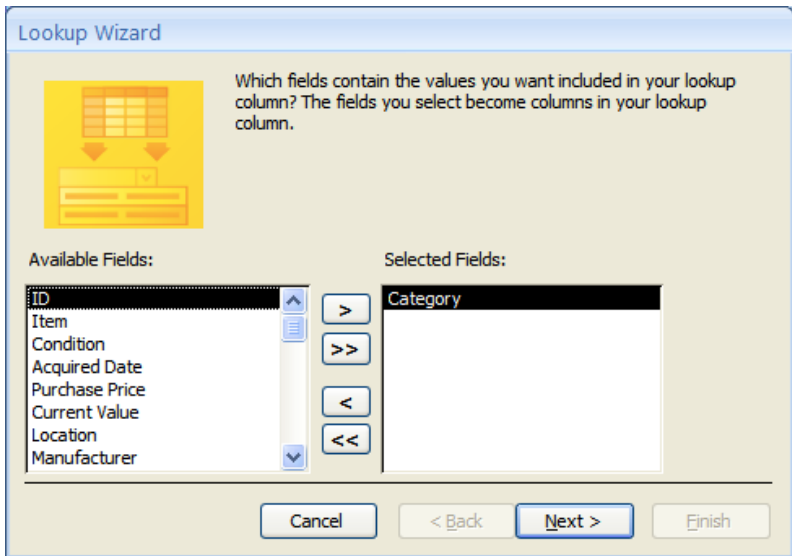


Pane Field List xuất hiện như hình dưới đây:



Nhấp vào biểu tượng (+) trong danh sách tại bảng cần lấy trường, một danh sách trong bảng đó sẽ xuất hiện, bạn chọn tới trường cần lấy rồi kéo thả đến vị trí cần thêm trường trên bảng mới.

Cửa sổ Lookup Wizard xuất hiện



Nhấp Next nếu như không chọn thêm trường nào.

**Lookup Wizard**

What sort order do you want for the items in your list box?

You can sort records by up to four fields, in either ascending or descending order.

1

2

3

4

Trong cửa sổ tiếp theo này bạn chọn trường cần sắp xếp rồi nhấp Next.

**Lookup Wizard**

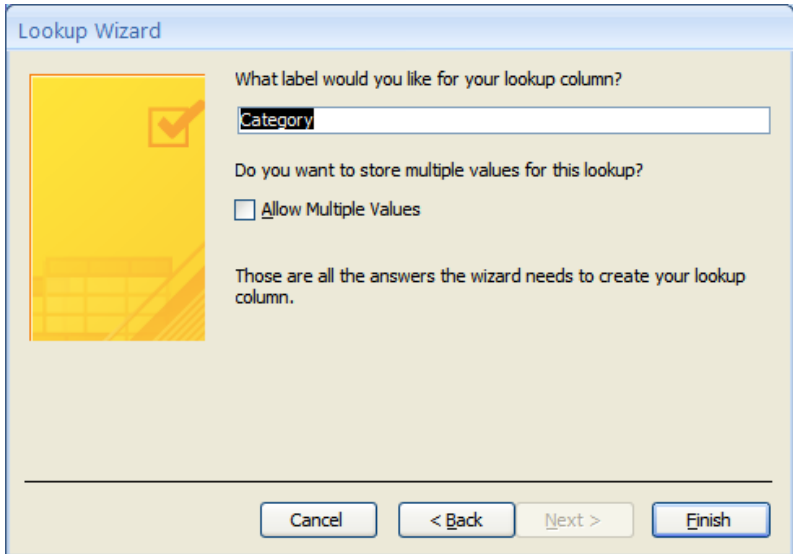
How wide would you like the columns in your lookup column?

To adjust the width of a column, drag its right edge to the width you want, or double-click the right edge of the column heading to get the best fit.

Hide key column (recommended)

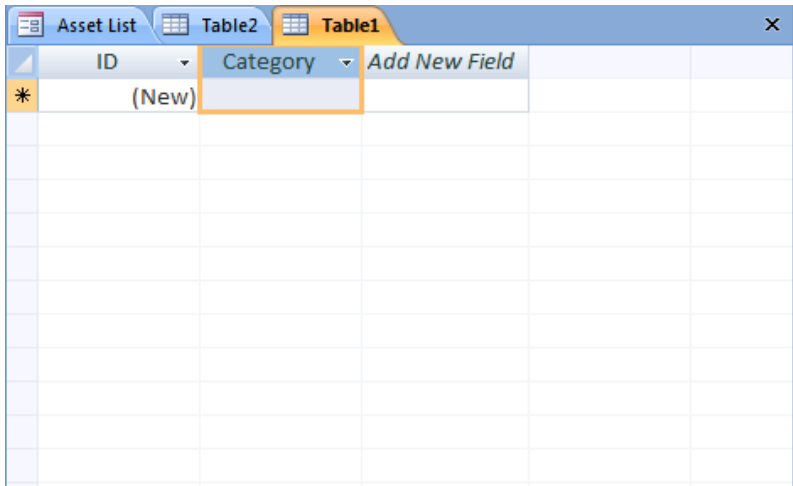
Category				

Bạn nhấp Next để chuyển sang bước tiếp theo



Nhập lại tên cho cột nếu cần thiết, cuối cùng nhấp Finish để kết thúc.

Như vậy một trường mới sẽ được thêm vào bảng



#### **2.4.4. Thiết lập thuộc tính cho trường**

Sau khi tạo một trường mới, để có thể điều khiển các hành vi, sự kiện diễn ra đối với trường này bạn cần phải thiết lập thuộc tính cho nó.

Ví dụ, việc thiết lập thuộc tính cho trường sẽ giúp bạn:

- Điều khiển được sự hiển thị dữ liệu trên trường đó.
- Giúp ngăn ngừa việc nhập dữ liệu không chính xác.
- Gán giá trị mặc định cho trường.
- Giúp tăng tốc độ tìm kiếm sắp xếp dữ liệu.

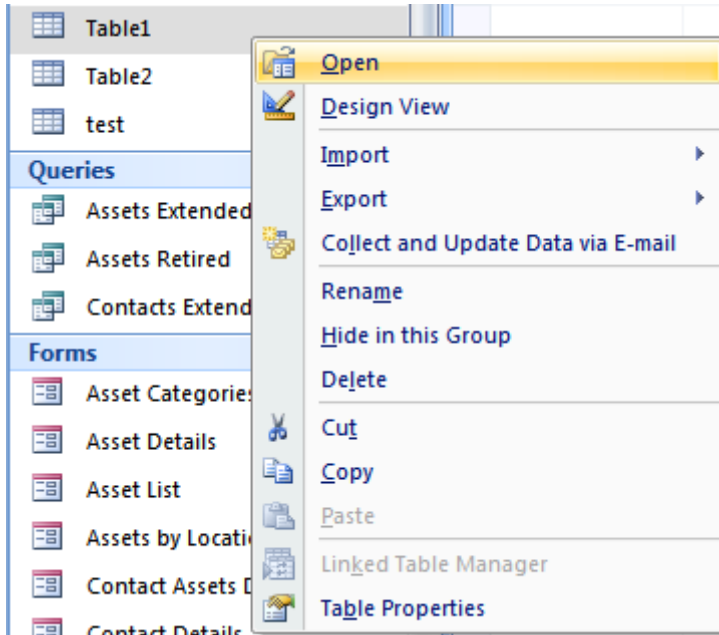
Bạn có thể thiết lập một số thuộc tính sẵn có của trường trong khi đang làm việc ở chế độ Datasheet View. Tuy nhiên để có thể thiết lập đầy các thuộc tính cho trường bạn nên sử dụng chế độ hiển thị Design View.

#### **\* Ở chế độ Datasheet View**

Ở chế độ Datasheet View bạn có thể đổi tên, dữ liệu, định dạng, và một số thuộc tính khác của một trường.

#### ***Mở bảng ở chế độ Datasheet View***

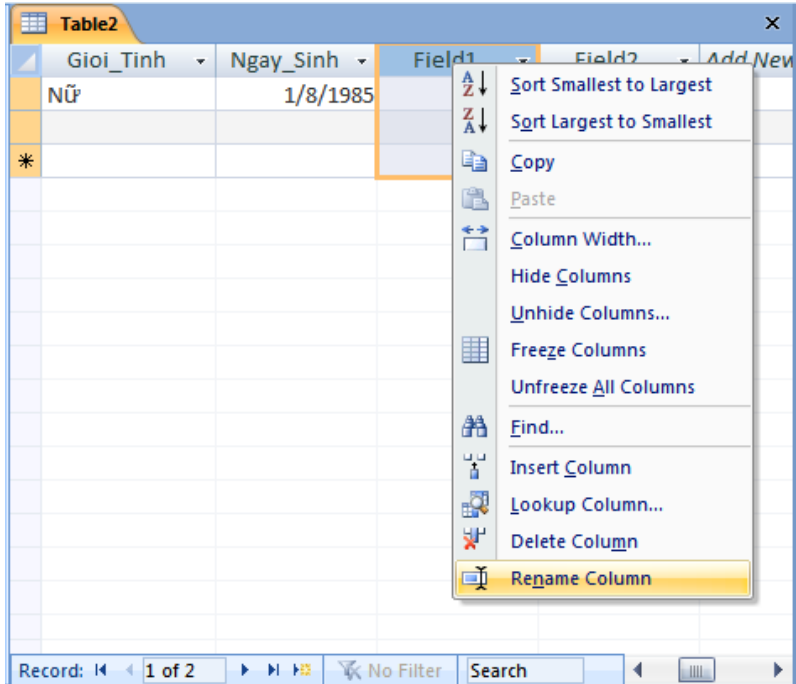
Trên Pane Navigation, nhấp phải chuột vào bảng cần mở, trong menu popup chọn Open.



### ***Đổi tên trường***

Khi bạn thêm một trường bằng cách nhập thẳng dữ liệu vào bảng ở chế độ Datasheet View, Access sẽ tự động định nghĩa một cái tên chung chung cho trường, ví dụ như bắt đầu với trường thứ nhất là Field1, các trường tiếp theo sẽ là Field2, Field3, ... Mặc định Access sử dụng tên của trường như một Label ở bất cứ nơi nào mà trường được hiển thị, ví dụ như tiêu đề cột khi mở bảng ở chế độ Datasheet View. Việc đổi tên trường sẽ giúp trường có thêm những thông tin mô tả về nó, nhờ đó mà giúp cho người dùng dễ dàng hơn trong quá trình sử dụng và chỉnh sửa.

Nhấp phải chuột vào phần tiêu đề cột của cột cần đổi tên, chọn Rename trong popup menu.



Nhập vào tên cần sửa đổi, lưu ý tên trường không được quá 64 ký tự bao gồm cả các khoảng trắng.

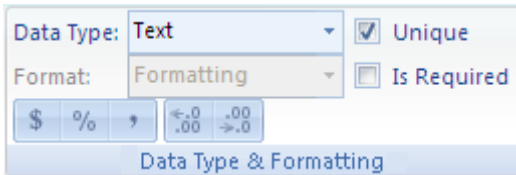
### ***Thay đổi kiểu dữ liệu của trường***

Khi bạn tạo ra một trường bằng cách nhập dữ liệu vào trường ở chế độ hiển thị Datasheet View, Access sẽ kiểm tra dữ liệu đó để xác định kiểu dữ liệu thích hợp cho trường. Chẳng hạn, nếu bạn nhập vào giá trị 1/ 1/ 2006, Access sẽ đoán rằng đó là dữ liệu ngày tháng và thiết lập kiểu dữ liệu cho trường này là Date/Time. Nếu như với một trường mà dữ liệu bạn nhập vào cho nó thuộc nhiều kiểu khác nhau, khi đó Access sẽ không thể xác định được trường đó thuộc kiểu gì và nó tự động gán cho trường đó là dữ liệu kiểu Text.

Kiểu dữ liệu của trường sẽ quyết định những thuộc tính mà nó có thể có. Ví dụ bạn chỉ có thể thiết lập thuộc tính Append Only cho trường kiểu Hyperlink và trường kiểu Memo.

Có rất nhiều trường hợp Access hiểu lầm kiểu dữ liệu mà bạn muốn thiết lập và bạn phải thay đổi lại kiểu dữ liệu cho trường đó. Ví dụ, bạn có một trường lưu số phòng trong khách sạn, bạn nhập dữ liệu cho trường là 10/2001, với chức năng tự tìm kiếm Access sẽ kiểm tra và gán cho trường này kiểu dữ liệu ngày tháng, nhưng thực tế bạn chỉ muốn lưu dữ liệu trường này kiểu Text. Để sửa lại kiểu dữ liệu cho trường này bạn làm như sau:

- Trên Ribbon nhấp chọn Tab Datasheet.
- Trong danh sách Data Type của nhóm Data Type & Formatting bạn chọn kiểu dữ liệu mà bạn muốn.



Các kiểu dữ liệu sẵn có trong Access.

Data type	Use to store	Size
Text	Sử dụng cho văn bản hoặc số nhưng không dùng để tính toán.	Tối đa 255 ký tự
Memo	Sử dụng lưu trữ ký tự, văn bản, hoặc các định dạng Rich	Tối đa 01 GB cho ký tự.



	Text.	
Number	Lưu trữ giá trị số nguyên.	1, 2, 4, hoặc 8 bytes, hoặc 16 bytes khi sử dụng cho trường ID
Date/Time	Dữ liệu kiểu ngày tháng	8 bytes.
Currency	Lưu giá trị kiểu tiền tệ	8 bytes.
AutoNumber	Giá trị nguyên được tự động thêm vào khi Insert một bản ghi.	4 bytes, 16 bytes
Yes/No	Giá trị kiểu Logic	1 bit (8 bits = 1 byte).
OLE Object	Lưu trữ đối tượng nhúng.	Tối đa 1 gigabyte.
Attachment	Lưu trữ Pictures, Images, Binary files, Office files.	Có thể lên tới 2 gigabytes
Hyperlink	Lưu trữ các liên kết.	Có thể lên tới 2Gb

Lookup Wizard	Sử dụng khi bạn muốn tạo ra trường Lookup (tìm kiếm). hiển thị dưới dạng một Combo cho phép chọn dữ liệu từ nguồn được khai báo	Tương ứng giá trị tìm kiếm trả về
---------------	--	-----------------------------------

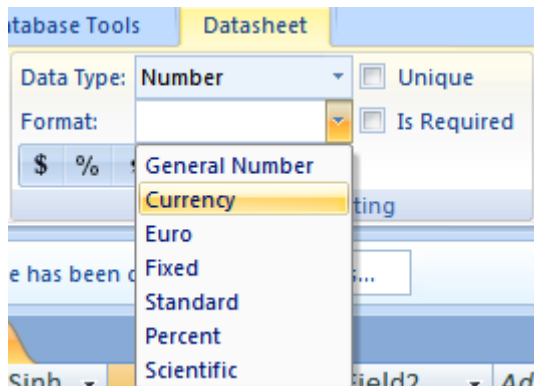
**Chú ý:** Dung lượng tối đa mà Access dành cho một Database là 2 Gigabyte.

### ***Thay đổi định dạng của trường***

Ngoài việc thiết lập kiểu dữ liệu cho một trường mới bạn còn có thể thiết lập thuộc tính Format cho trường đó. Ví dụ, nếu bạn nhập vào 10 giờ 50 phút sáng, Access sẽ thiết lập kiểu dữ liệu là Data/Time và thuộc tính Format là Medium Time. Để thay đổi thuộc tính Format của trường này bạn làm như sau:

Trên Ribbon, nhấp chọn tab Datasheet.

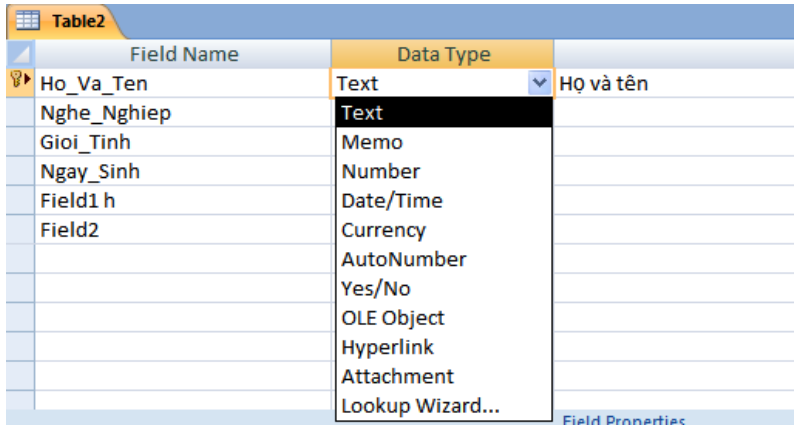
Trong danh sách Format, trong nhóm Data Type & Formatting, chọn kiểu định dạng mà bạn muốn thiết lập.



**Chú ý:** Trong danh sách Format chỉ có những giá trị sẵn có và nó thay đổi tùy thuộc vào kiểu dữ liệu của trường đó.

### **Ở chế độ Design View**

Khác với chế độ hiển thị Datasheet View, ở chế độ Design View bạn có thể thiết lập cho trường bất cứ thuộc tính nào mà bạn muốn. Ở chế độ này, bạn thiết lập kiểu dữ liệu cho trường trong bảng lưới.



The screenshot shows the Design View of a table named 'Table2'. The table has columns for 'Field Name' and 'Data Type'. The 'Data Type' column for the first field, 'Ho\_Va\_Ten', has a dropdown menu open, showing various data types: Text, Memo, Number, Date/Time, Currency, AutoNumber, Yes/No, OLE Object, Hyperlink, Attachment, and Lookup Wizard... The 'Text' option is currently selected. The 'Field Name' column contains 'Ho\_Va\_Ten', 'Nghe\_Nghiep', 'Gioi\_Tinh', 'Ngay\_Sinh', 'Field1 h', and 'Field2'. The 'Data Type' column contains 'Text', 'Text', 'Memo', 'Number', 'Date/Time', 'Currency', 'AutoNumber', 'Yes/No', 'OLE Object', 'Hyperlink', 'Attachment', and 'Lookup Wizard...'. The 'Field Name' column also contains the text 'Họ và tên' next to the first field.

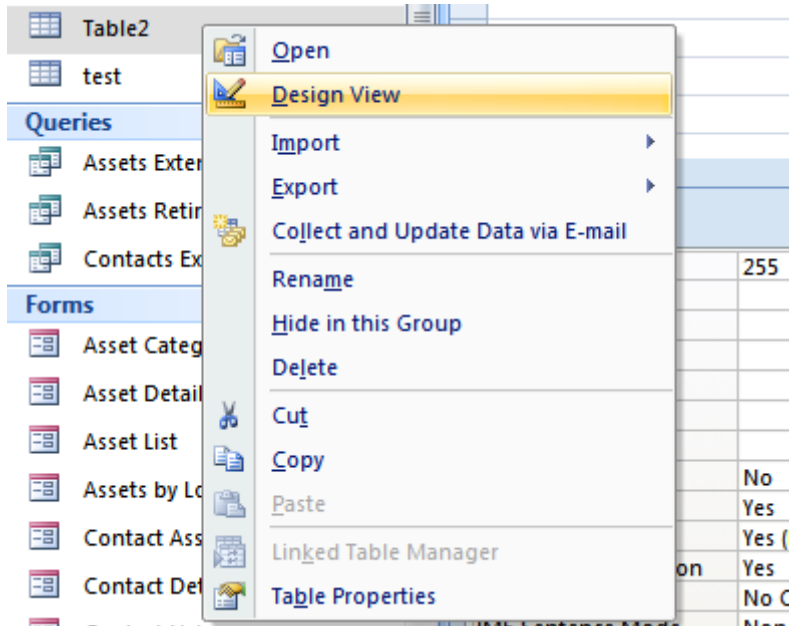
Field Name	Data Type
Ho_Va_Ten	Text
Nghe_Nghiep	Text
Gioi_Tinh	Memo
Ngay_Sinh	Number
Field1 h	Date/Time
Field2	Currency
	AutoNumber
	Yes/No
	OLE Object
	Hyperlink
	Attachment
	Lookup Wizard...

Và thiết lập thuộc tính khác cho trường trên Pane Field Properties.

### **Mở bảng ở chế độ Design View**

Để thiết lập kiểu dữ liệu cho trường ở chế độ Design View bạn làm như sau:

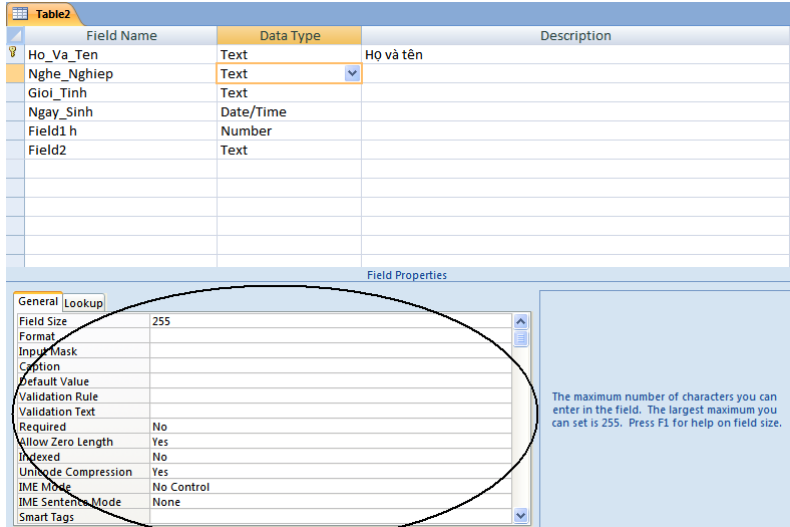
Trên Pane Navigation, nhấp phải chuột vào bảng cần mở chọn Design View.



Bảng xuất hiện ở chế độ Design View, trong lưới thiết kế bạn chọn tới trường cần thay đổi kiểu dữ liệu. Trong cột Data Type, lựa chọn lựa chọn kiểu dữ liệu từ danh sách đổ xuống.

### ***Thiết lập các thuộc tính khác cho trường***

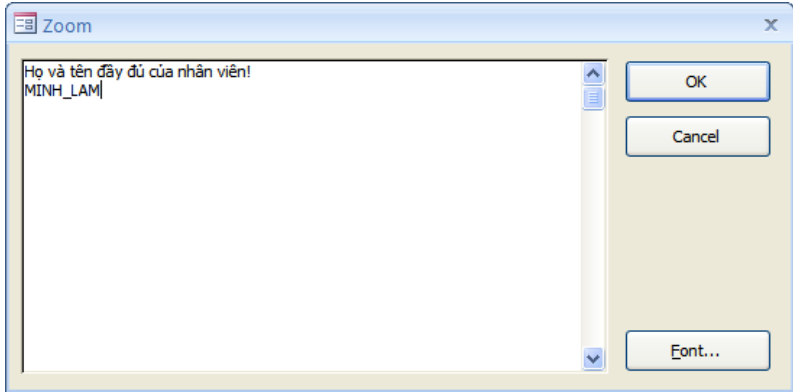
Trong bảng lưới thiết kế, chọn trường mà bạn muốn thiết lập thuộc tính. Access sẽ hiển thị những thuộc tính của trường đó trong Pane Field Properties.



**Chú ý:** Kiểu dữ liệu của trường xác định nên những thuộc tính mà trường đó có thể có.


Trong Pane Field Properties, chọn tới những thuộc tính cần thiết lập ở cột thứ nhất và điền lại giá trị ở cột tiếp theo, bạn cũng có thể sử dụng phím F6 để chuyển đổi qua lại giữa lưới danh sách trường phía trên và Pane Field properties.

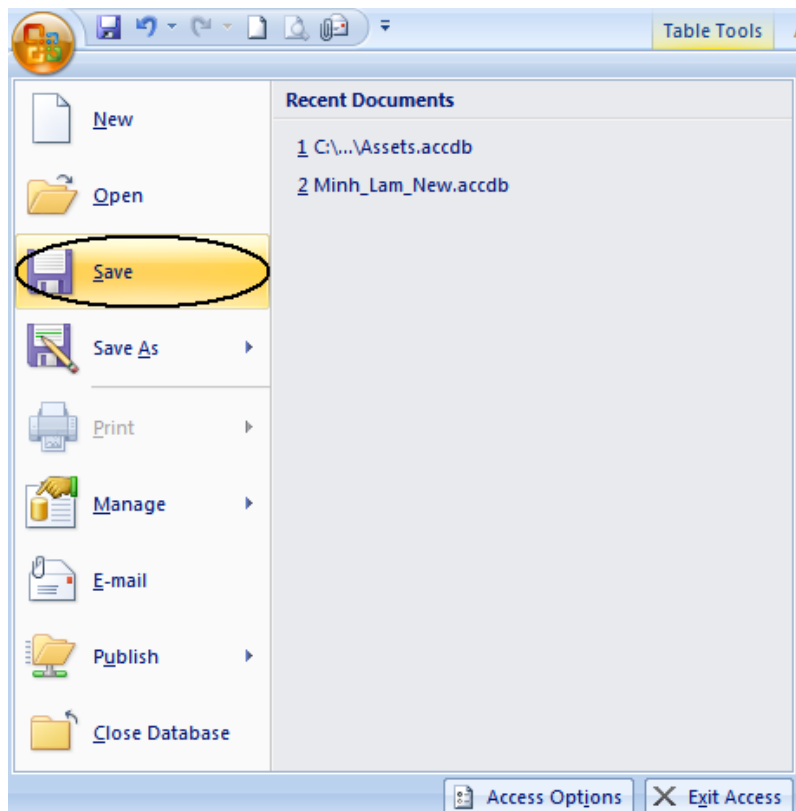
Khi làm việc với những thông tin dạng Text bạn có thể sử dụng phím Shift + f2 để bật hộp thoại Zoom, với hộp thoại này bạn sẽ có nhiều không gian hơn để soạn thảo. Ví dụ trong trường hợp này tôi đặt trỏ chuột vào dòng đầu tiên của cột Description rồi nhấp Shift + F2 hộp thoại Zoom xuất hiện như hình dưới đây:



### Lưu lại bảng

Sau khi đã tạo, chỉnh sửa bảng, bạn nên lưu lại các thiết kế đó. Khi bạn lưu bảng ở lần đầu tiên, hãy nhập vào tên cho bảng qua hộp thoại thông báo của Access. Tên bảng có ý nghĩa mô tả nội dung dữ liệu mà bảng đó đang chứa. Bạn có thể nhập tên bảng là các ký tự hoặc chữ số và bao gồm cả khoảng trắng nhưng không được quá 64 ký tự.

Để lưu lại bảng bạn có thể nhấp trực tiếp tổ hợp phím Ctrl + S hoặc từ nút Microsoft Office Button , bạn chọn Save.



### **Bài 3. Làm việc với Query**

Khi bạn muốn thêm, sửa đổi hoặc xóa dữ liệu trong Database, bạn nên sử dụng các query.

Với việc sử dụng Query, bạn có thể trả lời được rất nhiều câu hỏi cụ thể về dữ liệu mà bạn hiện có, trong khi việc đó sẽ rất khó khăn nếu bạn truy cập thẳng vào các bảng. Bạn có thể sử dụng các Query để lọc dữ liệu, thực hiện những phép tính với dữ liệu hoặc tóm tắt dữ liệu. Bạn cũng có thể sử dụng các Query để tự động quản lý dữ liệu và xem xét các thay đổi trong dữ liệu trước khi cho phép thực hiện những thay đổi đó.

Một Query là một yêu cầu đến cơ sở dữ liệu và kết quả trả về là dữ liệu hoặc hoạt động (Action) diễn ra trên cơ sở dữ liệu đó. Bạn có thể sử dụng một Query (truy vấn) để trả lời một câu hỏi đơn giản, thực hiện phép tính, kết hợp dữ liệu từ các bảng khác nhau, hoặc thậm chí có thể thêm, thay đổi, hoặc xóa các bảng dữ liệu. Các truy vấn mà bạn sử dụng để lấy dữ liệu từ một bảng hoặc để thực hiện các phép tính được gọi là truy vấn lựa chọn. Các truy vấn mà thêm, thay đổi, hoặc xóa dữ liệu được gọi là truy vấn hành động.

Trong phân này chúng tôi sẽ cung cấp cho các bạn những thông tin cơ bản về các truy vấn cùng với nó là các ví dụ về các loại hình khác nhau của các truy vấn.

Bạn cũng có thể sử dụng truy vấn để cung cấp dữ liệu cho một Form hoặc một Report. Một cơ sở dữ liệu được thiết kế tốt thì dữ liệu mà bạn muốn để hiển thị lên Form hoặc Report thường nằm ở một số bảng khác nhau. Bằng cách sử dụng truy vấn, bạn có thể liên kết các dữ liệu mà bạn muốn sử dụng lại với nhau trước khi đưa lên Form hoặc Report.

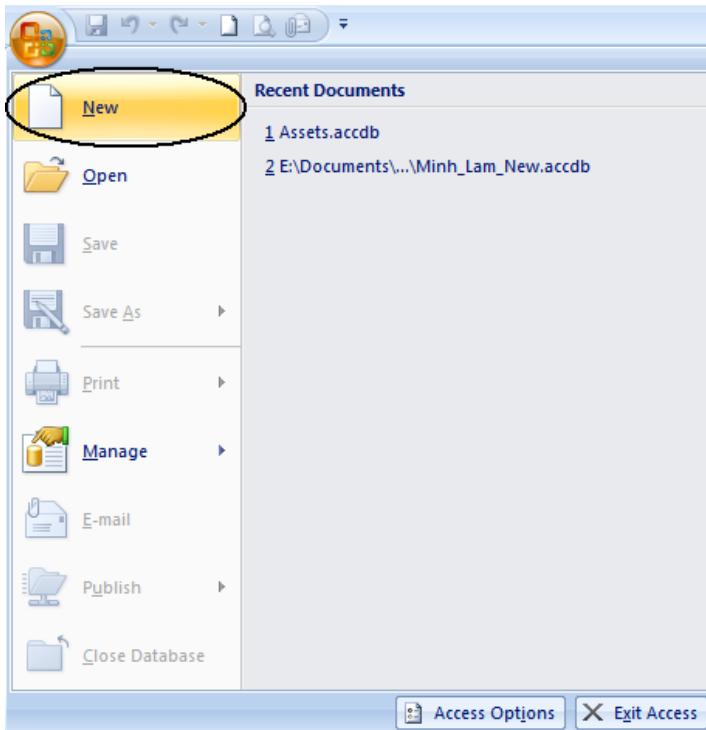
*Chú ý:* Các ví dụ trong bài này sử dụng Database Northwind có sẵn trong Template của Microsoft Access 2007.



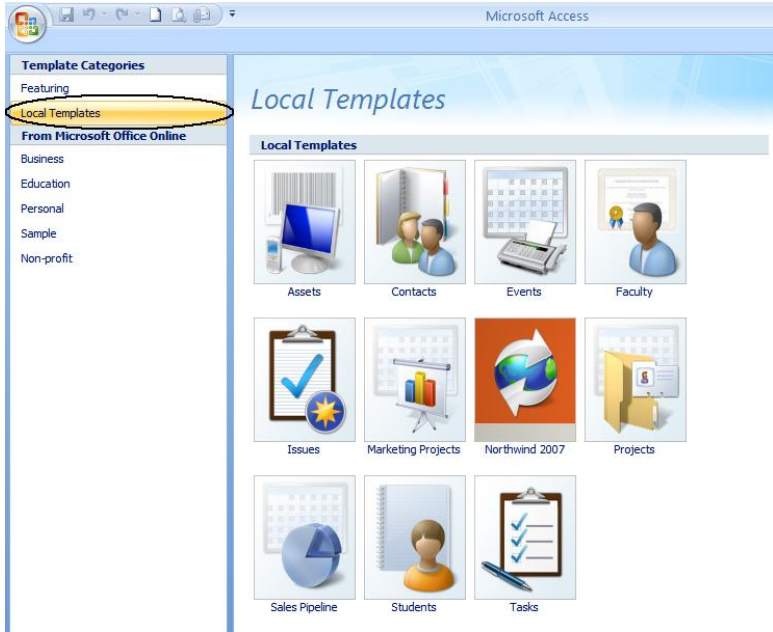
Để thiết lập Northwind 2007 bạn làm theo các bước sau:

**Bước 1:** Khởi động Microsoft Access 2007

**Bước 2:** Nhấp chọn nút Microsoft Office Button , sau đó chọn New.



**Bước 3:** Trong Pane bên trái, dưới mục Template Categories chọn Local Templates



**Bước 4:** Trong phần Local Template chọn Northwind 2007 sau đó nhấp Create.

**Bước 5:** Tiếp theo trang Northwind Traders xuất hiện bạn nhấp nút Close để đóng cửa sổ này lại.

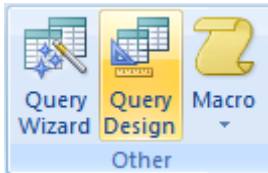
### 3.1. Tìm kiếm một nhóm dữ liệu trong bảng

Đôi khi bạn muốn xem xét tất cả dữ liệu từ một bảng, nhưng ở những lần khác, bạn lại chỉ muốn xem dữ liệu ở một số trường, hoặc bạn lại muốn xem một một số trường nhưng phải đáp ứng một vài tiêu chí nào đó, ... Với những yêu cầu như vậy bạn có thể sử dụng các truy vấn.

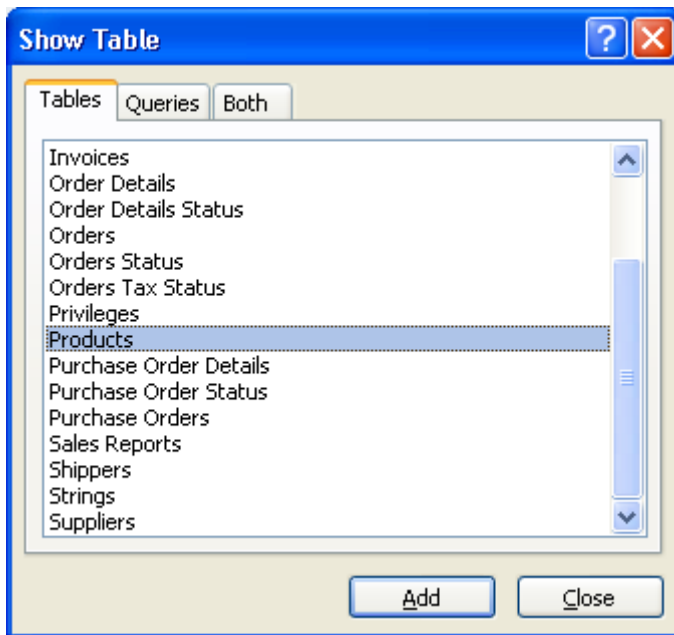
Giả sử bạn muốn xem xét một danh sách các sản phẩm và giá cả của chúng. Bạn có thể tạo một truy vấn mà trả về thông tin sản phẩm và giá cả bằng cách sử dụng thủ tục sau đây:

**Bước 1:** Mở Northwind 2007.

**Bước 2:** Trên Tab Create tìm tới nhóm Other nhấp chọn Query Design.

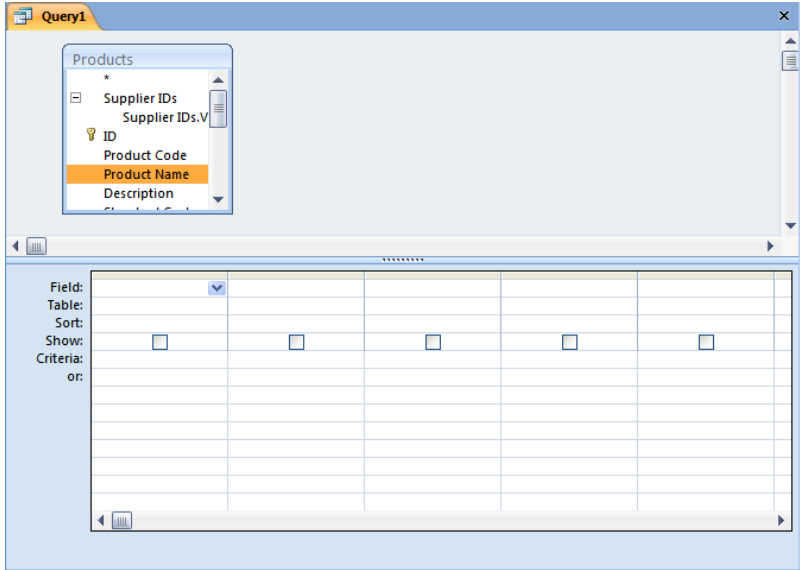


**Bước 3:** Trên hộp thoại Show Table tìm tới Tab Tables chọn Products rồi nhấp Add.

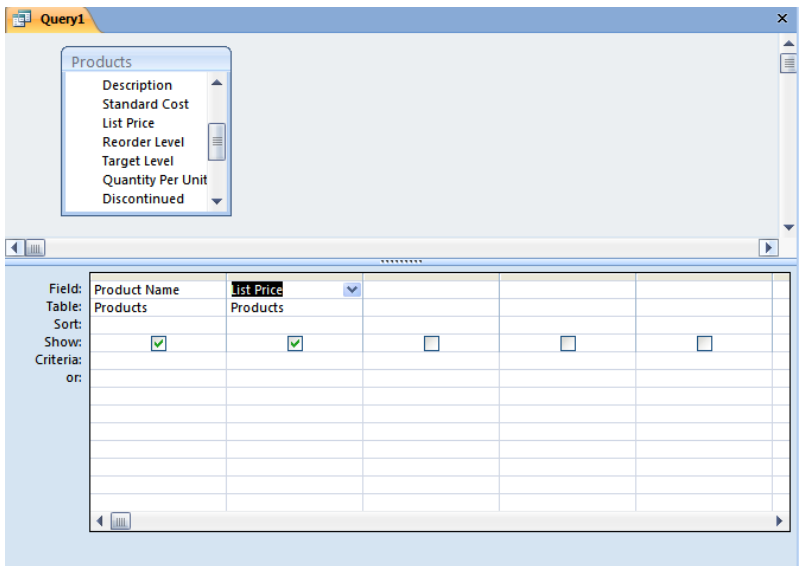


**Bước 4:** Nhấp Close để đóng hộp thoại Show Table.

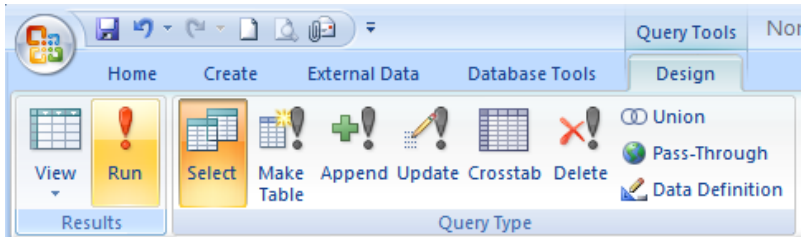
**Bước 5:** Trên màn hình thiết kế Query tìm tới bảng Products, nhấp đúp chuột lên hai trường là Product Name và List Price để thêm trường đó vào lưới thiết kế Query.



Sau khi nhấp chọn hai trường trong bảng Products kết quả sẽ được như hình dưới đây:



***Bước 6:*** Trên Tab Design trong nhóm Results nhấp chọn Run để chạy thử Query.



Kết quả sẽ được như hình dưới đây:



Product Name	List Price
Northwind Traders Chai	\$18.00
Northwind Traders Syrup	\$10.00
Northwind Traders Cajun Seasoning	\$22.00
Northwind Traders Olive Oil	\$21.35
Northwind Traders Boysenberry Spread	\$25.00
Northwind Traders Dried Pears	\$30.00
Northwind Traders Curry Sauce	\$40.00
Northwind Traders Walnuts	\$23.25
Northwind Traders Fruit Cocktail	\$39.00
Northwind Traders Chocolate Biscuits Mix	\$9.20
Northwind Traders Marmalade	\$81.00
Northwind Traders Scones	\$10.00
Northwind Traders Beer	\$14.00
Northwind Traders Crab Meat	\$18.40
Northwind Traders Clam Chowder	\$9.65
Northwind Traders Coffee	\$46.00
Northwind Traders Chocolate	\$12.75
Northwind Traders Dried Apples	\$53.00
Northwind Traders Long Grain Rice	\$7.00
Northwind Traders Gnocchi	\$38.00
Northwind Traders Ravioli	\$19.50
Northwind Traders Hot Pepper Sauce	\$21.05
Northwind Traders Tomato Sauce	\$17.00
Northwind Traders Mozzarella	\$34.80
Northwind Traders Almonds	\$10.00
Northwind Traders Mustard	\$12.00

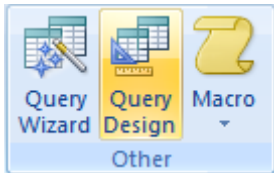
### 3.2. Xem dữ liệu đồng thời từ nhiều bảng

Những bảng biểu được thiết kế tốt thường mang theo các mối quan hệ với các bảng khác. Những mối quan hệ này tồn tại trên cơ sở các bảng có chung trường quan hệ. Khi bạn muốn xem dữ liệu từ nhiều bảng liên quan bạn có thể sử dụng truy vấn lựa chọn.

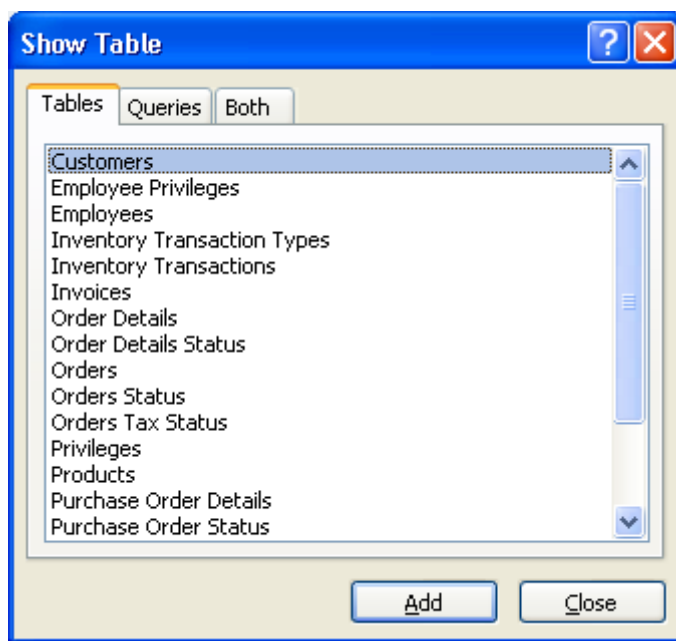
Giả mà bạn muốn xem xét lại đơn đặt hàng của một khách hàng sống tại thành phố nào đó. Dữ liệu về đơn đặt hàng và các dữ liệu về khách hàng được lưu trữ trong hai bảng trên cùng một cơ sở dữ liệu. Mỗi bảng có một Customer\_Id, và hai bảng này quan hệ với nhau ở dạng một nhiều. Bạn có thể tạo một truy vấn mà kết quả trả về là đơn hàng của một khách ở ở một thành phố cụ thể nào đó, ví dụ như Las Vegas, bằng cách thực hiện các bước dưới đây:

Bước 1: Mở Northwind 2007.

Bước 2: Trên Tab Create, trên nhóm Other, nhấp chọn Query Design.

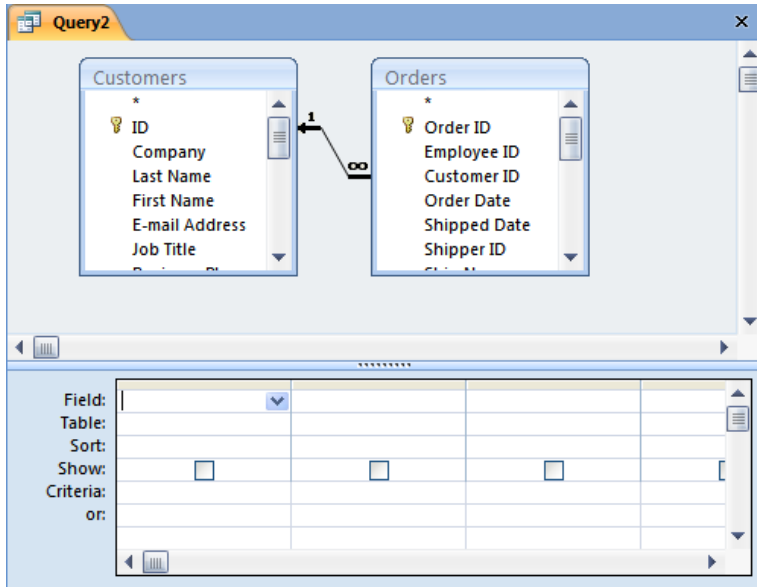


Bước 3: Trong hộp thoại Show Table, trên Tab Tables nhấp đúp chuột vào bảng Customers và Orders.



***Bước 4:*** Nhấp nút Close để đóng hộp thoại Show Table. Kết quả sẽ được như hình dưới đây:

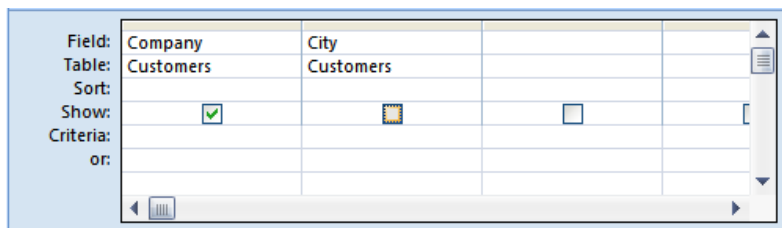




**Chú ý:** Giữa hai bảng có một đường Line kết nối giữa trường Id của bảng khách hàng Customer Id của bảng Orders. Đây là đường Line hiển thị mối qua hệ giữa hai bảng.

**Bước 5:** Trong bảng Customers, nhấp đúp chuột vào trường Company và City để thêm những trường đó vào phần lưới thiết kế Query.

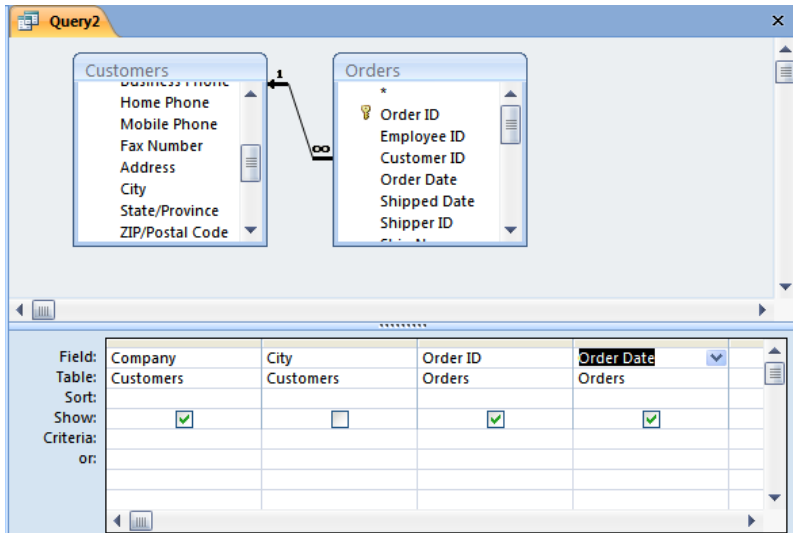
**Bước 6:** Trên lưới thiết kế Query, trên cột chứa trường City, bỏ tích chọn trên hàng Show của cột này.



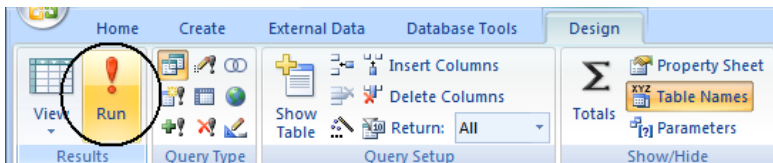
**Bước 7:** Trên dòng Criteria của cột City, bạn gõ vào giá trị là ‘Las Vegas’.

Xóa tích chọn ở dòng Show để kết quả trả về của Query không hiển thị thông tin này, và gỡ điều kiện ‘Las Vegas’ trên dòng Criteria để kết quả trả về của Query chỉ lấy những khách hàng ở thành phố Las Vegas.

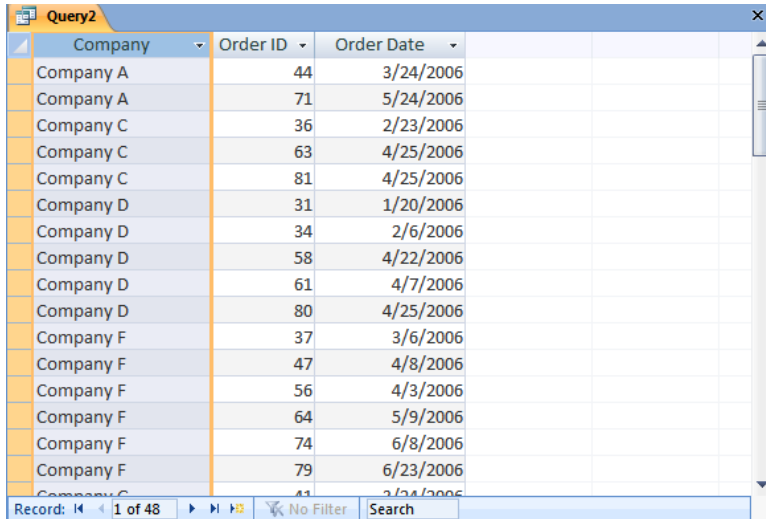
**Bước 8:** Trên bảng Orders nhấp đúp vào trường Order Id và trường Order Date để bổ xung thêm 2 trường này lên lưới thiết kế Query. Kết quả trên lưới sẽ được hiển thị như hình dưới đây:



**Bước 9:** Trên Tab Design tìm tới nhóm Results nhấp Run để chạy báo cáo.

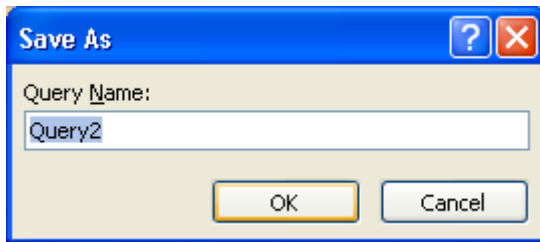


Kết quả Query trả về như hình dưới đây:



Company	Order ID	Order Date
Company A	44	3/24/2006
Company A	71	5/24/2006
Company C	36	2/23/2006
Company C	63	4/25/2006
Company C	81	4/25/2006
Company D	31	1/20/2006
Company D	34	2/6/2006
Company D	58	4/22/2006
Company D	61	4/7/2006
Company D	80	4/25/2006
Company F	37	3/6/2006
Company F	47	4/8/2006
Company F	56	4/3/2006
Company F	64	5/9/2006
Company F	74	6/8/2006
Company F	79	6/23/2006
Company C	41	2/24/2006

**Bước 10:** Nhấp tổ hợp phím Ctrl + S để lưu lại Query, một hộp thoại thông báo xuất hiện bạn nhập vào tên cho Query là Orders by city rồi nhấp Ok.



### 3.3. Sử dụng Parameters với Query

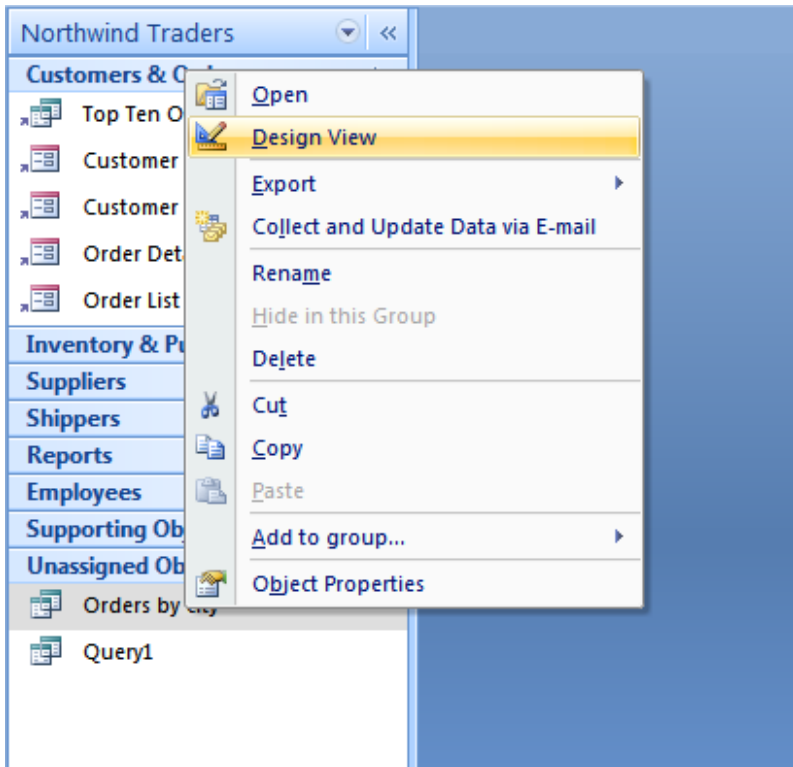
Đôi khi bạn muốn chạy một truy vấn mà chỉ hơi khác với một yêu cầu tìm kiếm hiện tại. Bạn có thể thay đổi truy vấn nguyên thủy để sử dụng điều kiện mới của bạn, nhưng nếu bạn thường xuyên muốn chạy các biến thể của một truy vấn cụ thể, hãy cân nhắc việc sử dụng một tham số truy vấn. Khi bạn chạy một tham số truy vấn, truy vấn sẽ nhắc bạn điền giá trị vào cho trường, nó sẽ sử dụng giá trị mà bạn vừa cung cấp để tạo ra các điều kiện (Criteria) cho truy vấn của bạn.

Trong ví dụ trước, bạn đã tạo ra một truy vấn mà kết quả trả về là những đơn hàng của những khách hàng ở thành phố Las Vegas. Bạn có thể chỉnh sửa truy vấn để thông báo cho người dùng nhập vào thành phố cần tìm, trong khi chạy truy vấn. Dưới đây là các bước tạo một truy vấn như vậy.

**Bước 1:** Mở Database Northwind 2007

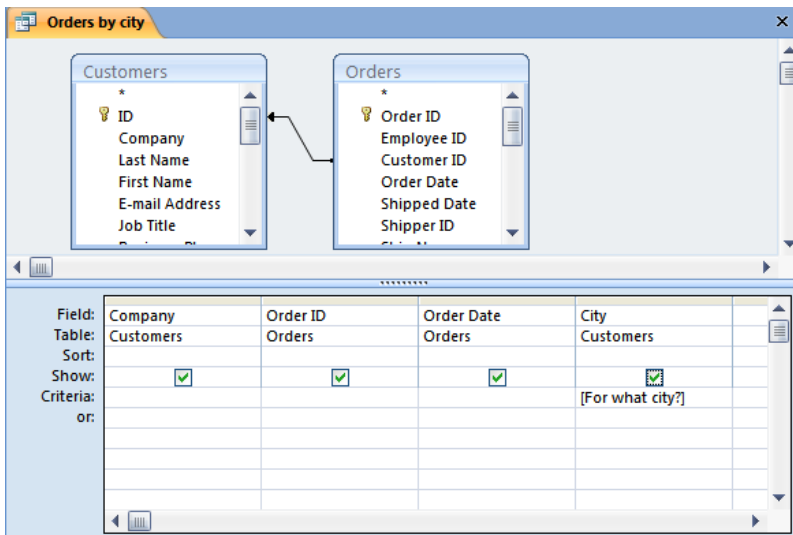
**Bước 2:** Nhấp chọn Shutter Bar để hiển thị Pane Navigation

**Bước 3:** Trong Pane Navigation, nhấp phải chuột vào Query có tên Orders by city mà bạn đã tạo ở phần trước, menu xuất hiện chọn Design View.

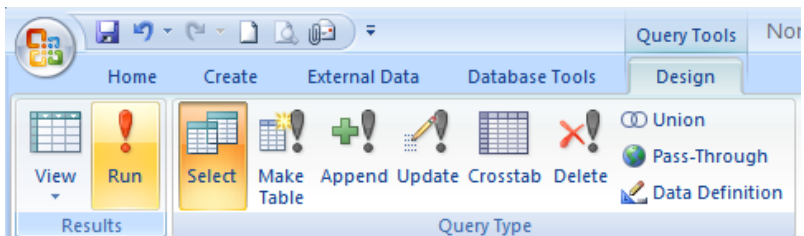


**Bước 4:** Trong lưới thiết kế Query, ở dòng Criteria của cột City xóa bỏ giá trị ‘Las Vegas’ nhập vào nội dung [For what city?], chuỗi [For what city?] là nội thông báo cho Parameters của bạn. Dấu ngoặc vuông cho thấy rằng bạn muốn truy vấn hiển thị thông báo về Parameter cần nhập vào, và nội dung văn bản (trong trường hợp này: For what city?) Là một câu hỏi hiển thị trên thông báo.

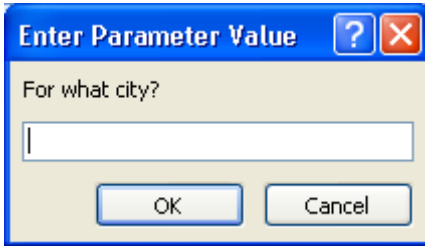
**Bước 5:** Nhấp chọn dấu tích trên dòng Show của cột City để hiển thị cột này trên kết quả của Query.



**Bước 6:** Trên Tab Design tìm tới nhóm Results, nhấp Run để chạy thử Query.



Access sẽ hiển thị hộp thoại yêu cầu nhập vào thành phố mà bạn muốn xem.

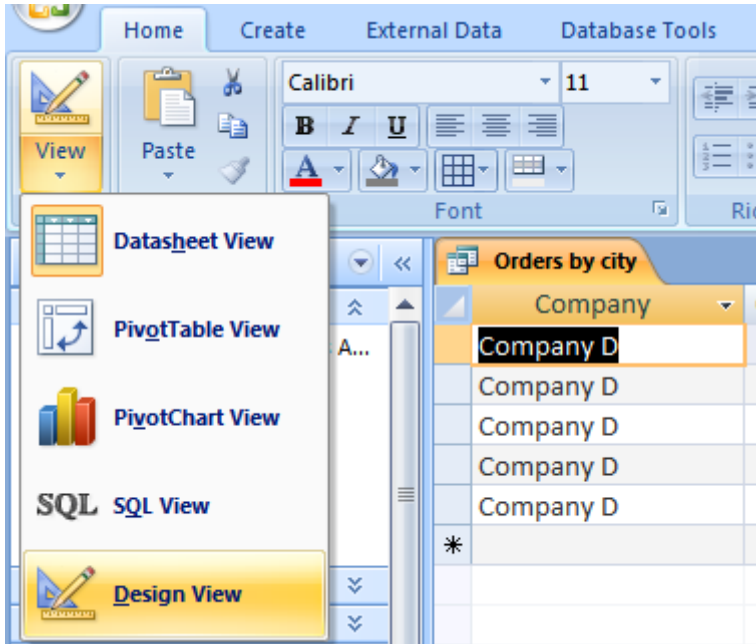


**Bước 7:** Giả sử ở đây bạn nhập New York rồi nhập Enter chúng ta sẽ được kết quả như hình dưới đây.

Company	Order ID	Order Date	City
Company D	31	1/20/2006	New York
Company D	34	2/6/2006	New York
Company D	58	4/22/2006	New York
Company D	61	4/7/2006	New York
Company D	80	4/25/2006	New York
*	(New)		

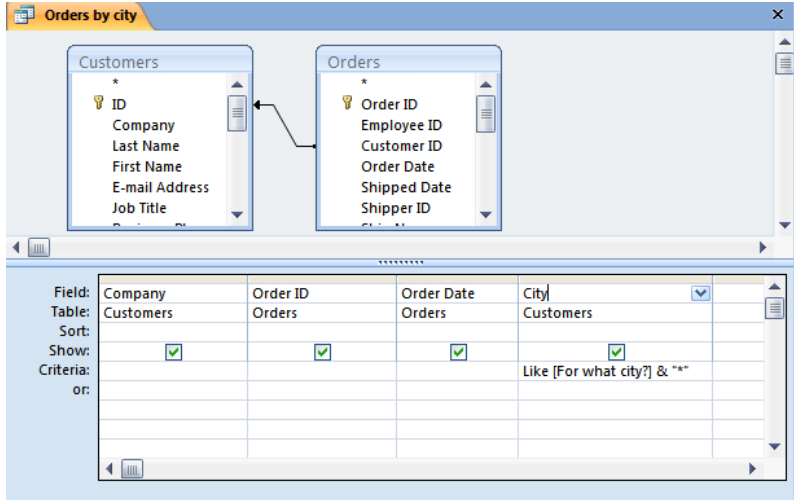
**Bước 8:** Nếu như bạn không muốn nhập vào đó giá trị chính xác, để tham số có thể mềm dẻo hơn, bạn có thể sử dụng ký tự đại diện trong chuỗi điều kiện (Criteria). Để làm được điều này bạn thực hiện tiếp các bước dưới đây.

**Bước 9:** Từ Tab Home tìm tới nhóm Views chọn nút View trên nút lệnh đổ xuống bạn chọn Design.



***Bước 10:*** Trên lưới thiết kế Query tại dòng Criteria của cột City bạn nhập vào nội dung dưới đây:

Like [For what city?]&"\*"



Trong nội dung thông báo này, từ khóa Like, toán tử & và dấu (\*) cho phép cho phép người sử dụng gõ vào một giá trị mà có sự kết hợp giữa các ký tự với nhau để trả về nhiều hơn các kết quả. Ví dụ, nếu người dùng gõ vào ký tự (\*), truy vấn sẽ trả về tất cả các thành phố, nếu người sử dụng các loại L, truy vấn sẽ tìm kiếm và trả lại tất cả các thành phố mà bắt đầu bằng chữ "L" và nếu người dùng gõ \* s \*, truy vấn trả về tất cả các thành phố mà tên chứa chữ "s".

**Bước 11:** Trên Tab Design tìm tới nhóm Result nhấp chọn Run.

Dưới đây là kết quả khi bạn gõ vào ô Parameter là \*.



Company	Order ID	Order Date	City
Company AA	30	1/15/2006	Las Vegas
Company D	31	1/20/2006	New York
Company L	32	1/22/2006	Las Vegas
Company H	33	1/30/2006	Portland
Company D	34	2/6/2006	New York
Company CC	35	2/10/2006	Denver
Company C	36	2/23/2006	Los Angelas
Company F	37	3/6/2006	Milwaukee
Company BB	38	3/10/2006	Memphis
Company H	39	3/22/2006	Portland
Company J	40	3/24/2006	Chicago
Company G	41	3/24/2006	Boise
Company J	42	3/24/2006	Chicago
Company K	43	3/24/2006	Miami
Company A	44	3/24/2006	Seattle
Company BB	45	4/7/2006	Memphis
Company I	46	4/5/2006	Salt Lake City
Company F	47	4/8/2006	Milwaukee
Company H	48	4/5/2006	Portland
Company Y	50	4/5/2006	Chicago

### 3.4. Thực hiện tính toán trên cơ sở dữ liệu

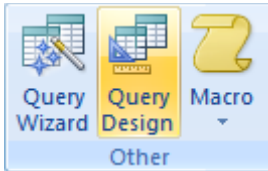
Thông thường bạn không sử dụng bảng để lưu trữ các trường tính toán bởi vì những giá trị đó có thể được dựa trên rất nhiều thông tin ở các bảng khác nhau trong cùng một cơ sở dữ liệu. Ví dụ, bảng Order Details trong Database Northwind 2007 không lưu giữ giá trị tổng con sản phẩm, bởi nó được tính toán từ các trường Quantity, Unit Price và Discount cũng trong bảng này.

Trong một số trường hợp, các trị tính toán sẽ trở nên không phù hợp, lỗi thời (out-of-date), bởi vì các giá trị tạo nên nó đã thay đổi. Ví dụ, bạn không muốn để lưu lại độ tuổi của ai đó trong một bảng, bởi vì mỗi năm bạn sẽ cần phải cập nhật lại những giá trị đó, thay vào đó, bạn có thể lưu trữ ngày tháng năm sinh của người này, sau đó khi cần sử dụng bạn chỉ cần viết một truy vấn tính ra tuổi của người đó dựa trên dữ liệu ngày tháng năm sinh và ngày hiện tại.

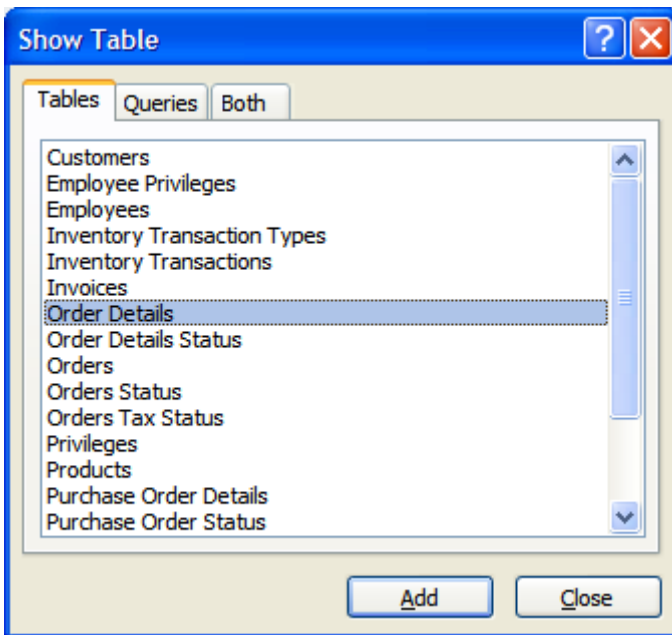
Bạn sử dụng thủ tục sau đây để tính tổng con sản phẩm dựa trên dữ liệu từ bảng Order Details.

**Bước 1:** Mở cơ sở dữ liệu Northwind 2007

**Bước 2:** Trên Tab Create, tìm tới nhóm Other chọn Query Design.

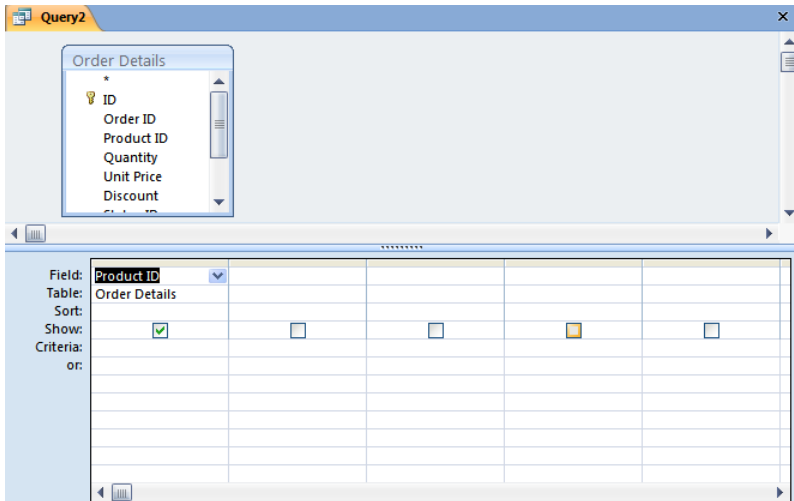


**Bước 3:** Trong hộp thoại Show Table, trong tab Tables, nhấp đúp chuột chọn Order Details.

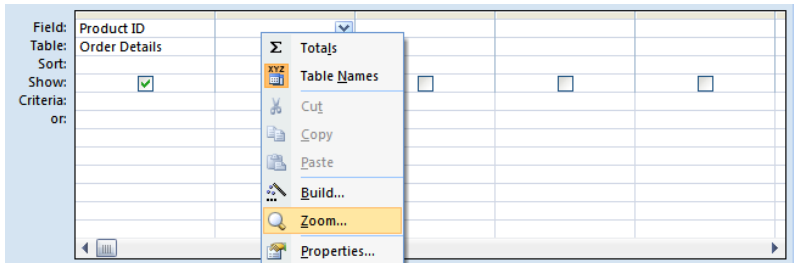


**Bước 4:** Nhấp Close để đóng cửa sổ này

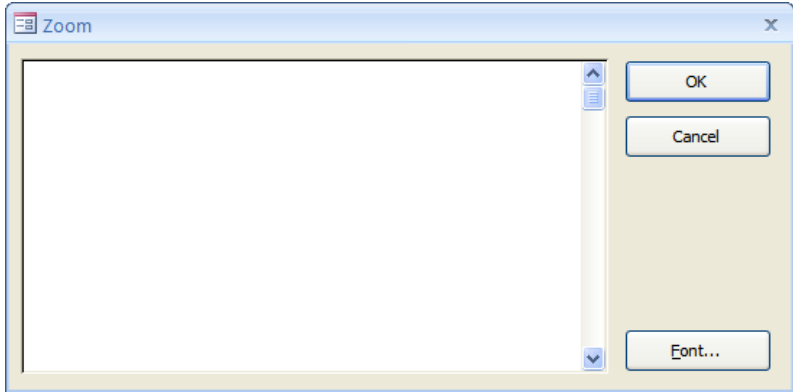
**Bước 5:** Trên bảng Order Details, nhấp đúp chuột vào trường Product ID để thêm trường vào cột đầu tiên trên lưới thiết kế Query.



**Bước 6:** Ở cột thứ 2 trên lưới thiết kế Query, nhấp phải chuột vào hàng Field rồi chọn Zoom.

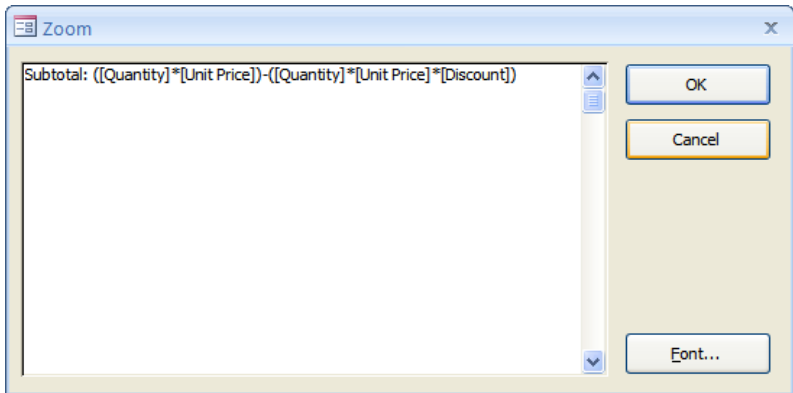


Hộp thoại Zoom xuất hiện.



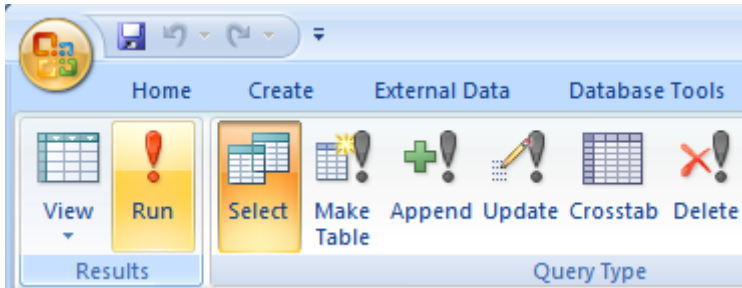
**Bước 7:** Trong hộp thoại Zoom, gõ vào đó nội dung dưới đây.

Subtotal: ([Quantity]\*[Unit Price])-([Quantity]\*[Unit Price]\*[Discount])



Nhấp Ok để chấp nhận giá trị.

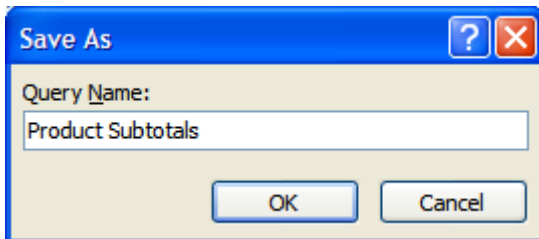
**Bước 8:** Trên Tab Design chọn tới nhóm Results, rồi nhấp Run.



Kết quả sẽ được như hình dưới đây

Product	Subtotal
Northwind Traders Beer	1400
Northwind Traders Dried Plums	105
Northwind Traders Dried Pears	300
Northwind Traders Dried Apples	530
Northwind Traders Dried Plums	35
Northwind Traders Chai	270
Northwind Traders Coffee	920
Northwind Traders Chocolate Biscuits Mix	276
Northwind Traders Chocolate Biscuits Mix	184
Northwind Traders Chocolate	127.5
Northwind Traders Clam Chowder	1930
Northwind Traders Curry Sauce	680
Northwind Traders Coffee	13800
Northwind Traders Chocolate	1275
Northwind Traders Green Tea	598
Northwind Traders Coffee	13800
Northwind Traders Boysenberry Spread	250
Northwind Traders Cajun Seasoning	220

**Bước 9:** Cuối cùng nhập Ctrl + S để lưu lại Query, trong hộp thoại nhắc nhập tên bạn nhập vào tên cho Query là: Product Subtotals.



### 3.5. Tổng hợp hoặc tính tổng dữ liệu

Khi bạn sử dụng bảng để ghi lại các giao dịch hoặc lưu trữ dữ liệu phát sinh, tại một thời điểm nào đó bạn hoàn toàn có thể tổng hợp, thống kê chúng ví như tính tổng hay trung bình, ...

Microsoft Office Access 2007 cung cấp một tính năng mới, một dòng tổng cộng, cho phép bạn tổng hợp dữ liệu một cách đơn giản trong bất kỳ Datasheet nào như Form, Query, Report, ... Hãy thử chức năng này bằng cách thêm một dòng tổng cộng vào Query Product Subtotals mà bạn vừa tạo ở phần trước.

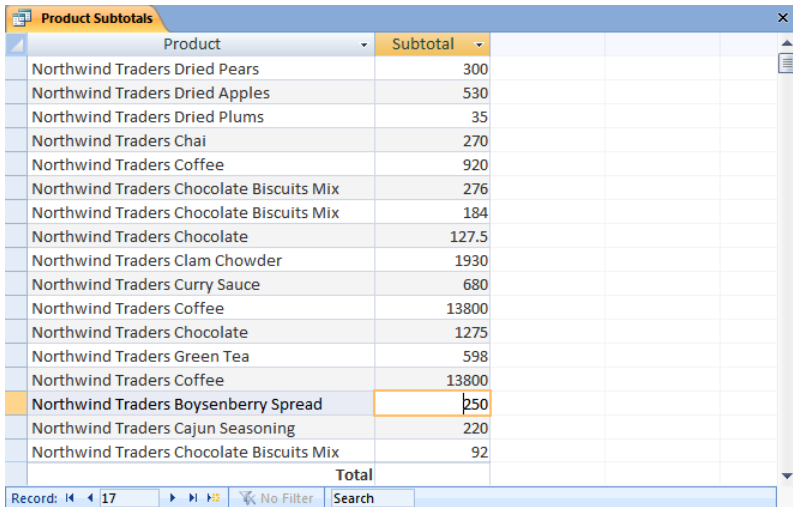
**Bước 1:** Chạy Query Product Subtotals ở chế độ hiển thị Datasheet View.

Product	Subtotal
Northwind Traders Beer	1400
Northwind Traders Dried Plums	105
Northwind Traders Dried Pears	300
Northwind Traders Dried Apples	530
Northwind Traders Dried Plums	35
Northwind Traders Chai	270
Northwind Traders Coffee	920
Northwind Traders Chocolate Biscuits Mix	276
Northwind Traders Chocolate Biscuits Mix	184
Northwind Traders Chocolate	127,5
Northwind Traders Clam Chowder	1930
Northwind Traders Curry Sauce	680
Northwind Traders Coffee	13800
Northwind Traders Chocolate	1275
Northwind Traders Green Tea	598
Northwind Traders Coffee	13800
Northwind Traders Boysenberry Spread	250
Northwind Traders Cajun Seasoning	220

**Bước 2:** Trên Tab Home, trong nhóm Records nhấp chọn Totals.

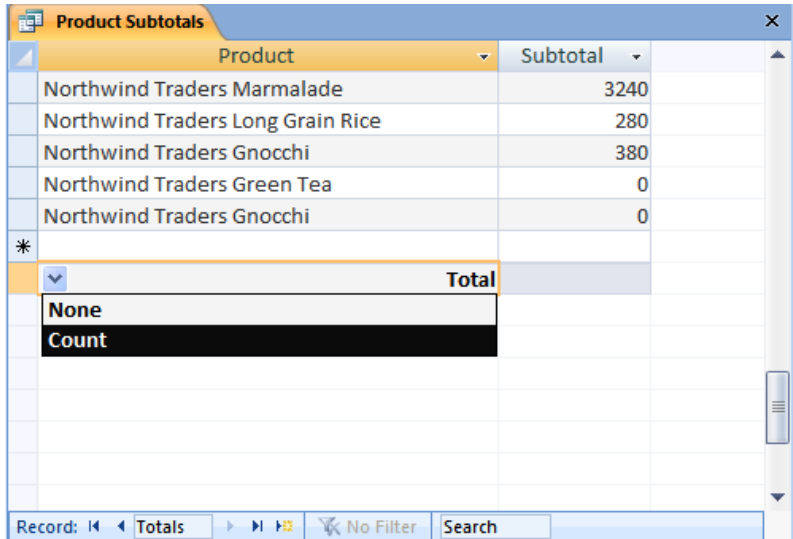


Một dòng mới xuất hiện phía dưới đáy của Datasheet với dòng chữ Total ở cột đầu tiên.



Product	Subtotal
Northwind Traders Dried Pears	300
Northwind Traders Dried Apples	530
Northwind Traders Dried Plums	35
Northwind Traders Chai	270
Northwind Traders Coffee	920
Northwind Traders Chocolate Biscuits Mix	276
Northwind Traders Chocolate Biscuits Mix	184
Northwind Traders Chocolate	127.5
Northwind Traders Clam Chowder	1930
Northwind Traders Curry Sauce	680
Northwind Traders Coffee	13800
Northwind Traders Chocolate	1275
Northwind Traders Green Tea	598
Northwind Traders Coffee	13800
Northwind Traders Boysenberry Spread	250
Northwind Traders Cajun Seasoning	220
Northwind Traders Chocolate Biscuits Mix	92
<b>Total</b>	

**Bước 3:** Nhấp chọn vào Cell đầu tiên ở dòng cuối cùng của Datasheet (dòng Total).



The screenshot shows the 'Product Subtotals' window in Microsoft Access. The window title is 'Product Subtotals'. It contains a table with two columns: 'Product' and 'Subtotal'. The data rows are:

Product	Subtotal
Northwind Traders Marmalade	3240
Northwind Traders Long Grain Rice	280
Northwind Traders Gnocchi	380
Northwind Traders Green Tea	0
Northwind Traders Gnocchi	0
* Total	

A dropdown menu is open for the 'Total' row, showing two options: 'None' and 'Count'. The 'Count' option is currently selected. At the bottom of the window, there is a status bar with the following text: 'Record: < < Totals > > No Filter Search'.

Vì đây là dữ liệu kiểu Text vì vậy chỉ có hai lựa chọn cho bạn tổng hợp dữ liệu là None và Count, ở đây bạn chọn Count.

Bước 4: Tiếp tục nhấp chọn ô ở cột thứ 2.



Product	Subtotal
Northwind Traders Marmalade	3240
Northwind Traders Long Grain Rice	280
Northwind Traders Gnocchi	380
Northwind Traders Green Tea	0
Northwind Traders Gnocchi	0

Vì đây là trường số nên bạn có rất nhiều lựa chọn để tổng hợp dữ liệu, trong trường hợp này bạn chọn Sum để tính tổng cột Subtotal.

**Bước 5:** Kết quả cuối cùng Query sẽ hiển thị như hình dưới đây.

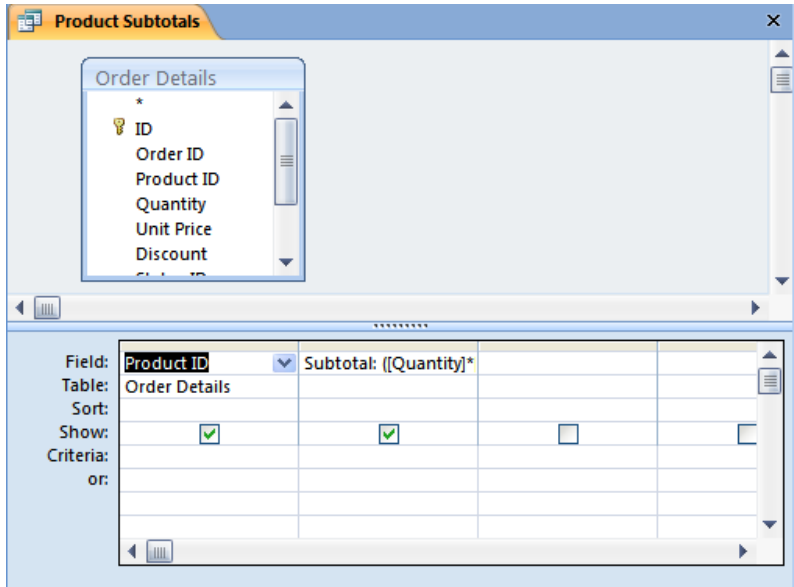
Product	Subtotal
Northwind Traders Curry Sauce	800
Northwind Traders Dried Plums	52.5
Northwind Traders Almonds	200
Northwind Traders Mozzarella	1392
Northwind Traders Syrup	500
Northwind Traders Curry Sauce	120
Northwind Traders Marmalade	3240
Northwind Traders Long Grain Rice	280
Northwind Traders Gnocchi	380
Northwind Traders Green Tea	0
Northwind Traders Gnocchi	0
*	
	68137

### ***Tạo ra một Query tính tổng phức tạp hơn***

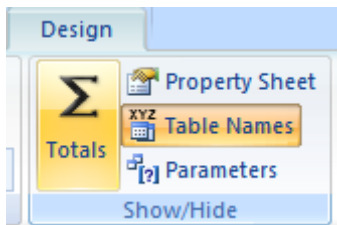
Dòng tổng cộng mà các bạn vừa thực hiện trên Datasheet có thể mạnh riêng của nó truy nhiên đối với những câu hỏi phức tạp hơn bạn nên sử dụng Query tổng cộng. Query tổng cộng bản chất là một Query lựa chọn mà cho phép bạn có thể nhóm hoặc tổng hợp dữ liệu. Ví dụ, giả sử bạn muốn xem tổng số doanh thu trên mỗi sản phẩm. Trong Query tổng cộng bạn có thể sử dụng hàm Sum là một trong những hàm thống kê ( SUM, COUNT, AVG, ...) để xem tổng số doanh thu trên mỗi sản phẩm.

Sử dụng các thủ tục sau đây để sửa query Product Subtotals để nó có tổng hợp theo từng sản phẩm.

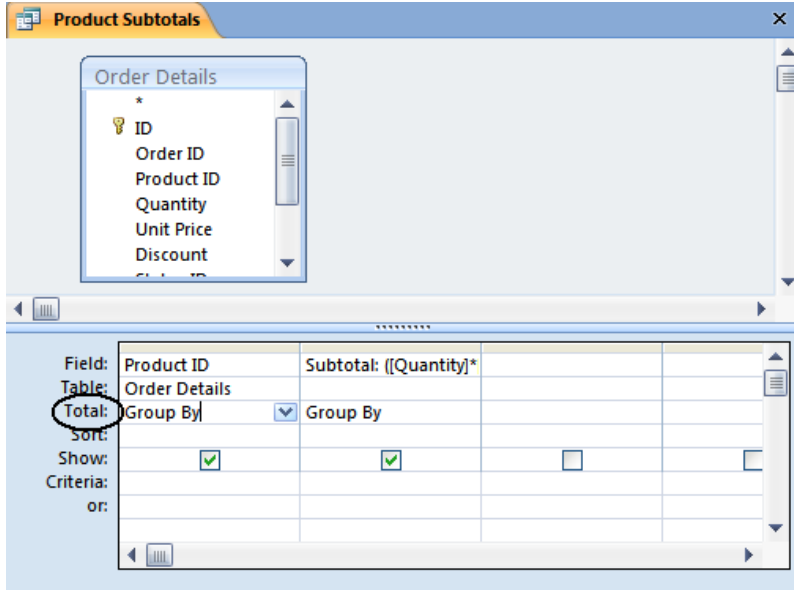
***Bước 1:*** Chọn tới Query Product Subtotals và mở nó ở chế độ Design View.



Bước 2: Từ Tab Design chọn tới nhóm Show/Hide nhấp chọn Totals.

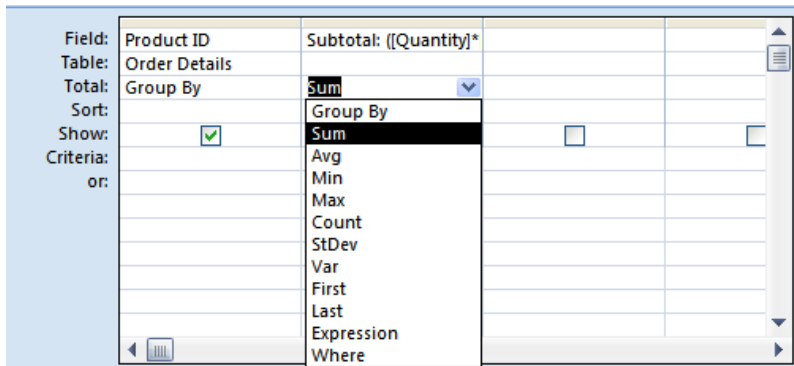


Một dòng Totals sẽ xuất hiện trên lưới thiết kế Query.



**Chú ý:** Mặc dầu cùng tên nhưng dòng Totals trên lưới thiết kế và dòng Total trên Datasheet không giống nhau.

**Bước 3:** Ở cột thứ 2 trên lưới thiết kế Query, trên dòng Total bạn chọn giá trị Sum trong hộp Combo đổ xuống.



**Bước 4:** Trên Tab Design tìm tới nhóm Result chọn Run để chạy thử Query, kết quả sẽ được như hình dưới đây:

Product	Subtotal
Northwind Traders Chia	720
Northwind Traders Syrup	500
Northwind Traders Cajun Seasoning	880
Northwind Traders Olive Oil	533.75
Northwind Traders Boysenberry Spread	2500
Northwind Traders Dried Pears	1200
Northwind Traders Curry Sauce	2600
Northwind Traders Fruit Cocktail	1560
Northwind Traders Chocolate Biscuits Mix	782
Northwind Traders Marmalade	3240
Northwind Traders Scones	200
Northwind Traders Beer	6818
Northwind Traders Crab Meat	2208
Northwind Traders Clam Chowder	2798.5
Northwind Traders Coffee	29900
Northwind Traders Chocolate	2550
Northwind Traders Dried Apples	2120
Northwind Traders Long Grain Rice	280
<b>24</b>	<b>68137</b>

**Bước 5:** Cuối cùng nhấn Ctrl + S để lưu lại Query

### 3.6. Tạo một bảng từ nhiều bảng khác.

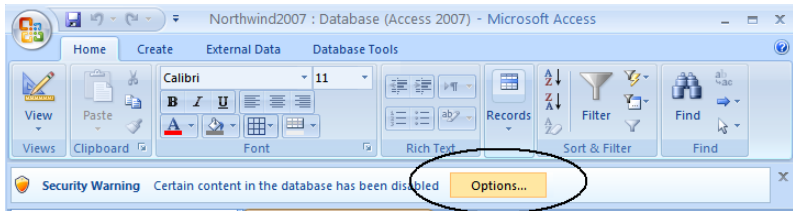
Bạn có thể sử dụng make-table query để tạo một bảng mới từ dữ liệu được lưu trữ ở nhiều bảng khác.

Ví dụ, giả sử bạn muốn gửi dữ liệu về đơn hàng cho các đối tác ở Chicago và bạn sử dụng Access để lập báo cáo. Thay vì gửi cho họ tất cả các dữ liệu về đơn đặt hàng, bạn muốn giới hạn những dữ liệu được gửi đi chỉ là những đơn đặt hàng ở Chicago.

Bạn có thể xây dựng một Query lựa chọn chỉ chứa những đơn hàng ở Chicago, và sau đó sử dụng Query này để tạo một bảng dữ liệu mới. Dưới đây là các bước thực hiện những công việc đó.

**Bước 1:** Mở cơ sở dữ liệu Northwind 2007

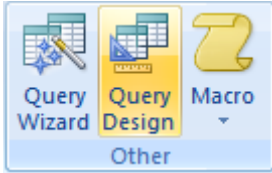
**Bước 2:** Để chạy một make-table query, bạn cần phải kích hoạt tính năng enable the database content nếu trước đó bạn mở nó ở chế độ Untrusted. Bạn nhấp chọn Options trên thanh Message Bar, xuất hiện phía dưới Ribbon.



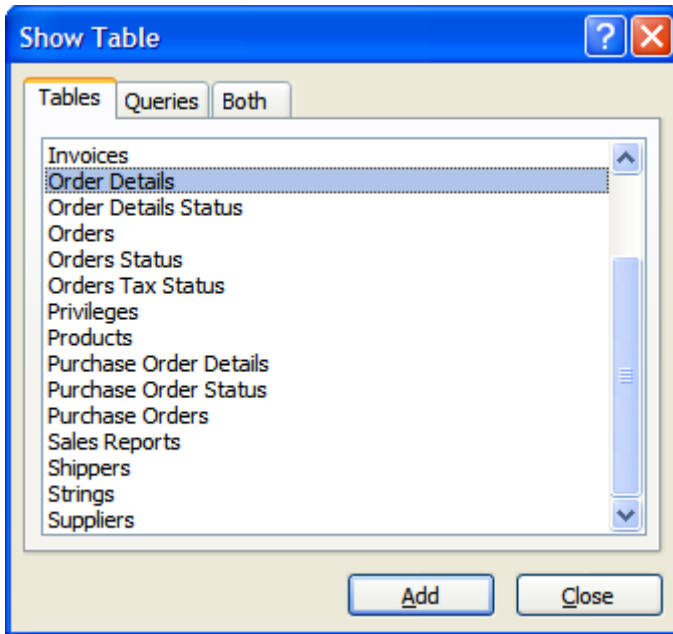
Hộp thoại Microsoft Office Security Options xuất hiện bạn nhấp chọn mục Enable this content rồi nhấp Ok. Cửa sổ Login Dialog xuất hiện, bạn hãy đóng cửa sổ này lại.



**Bước 3:** Trên Tab Create tìm tới nhóm Other nhấp chọn Query Design.

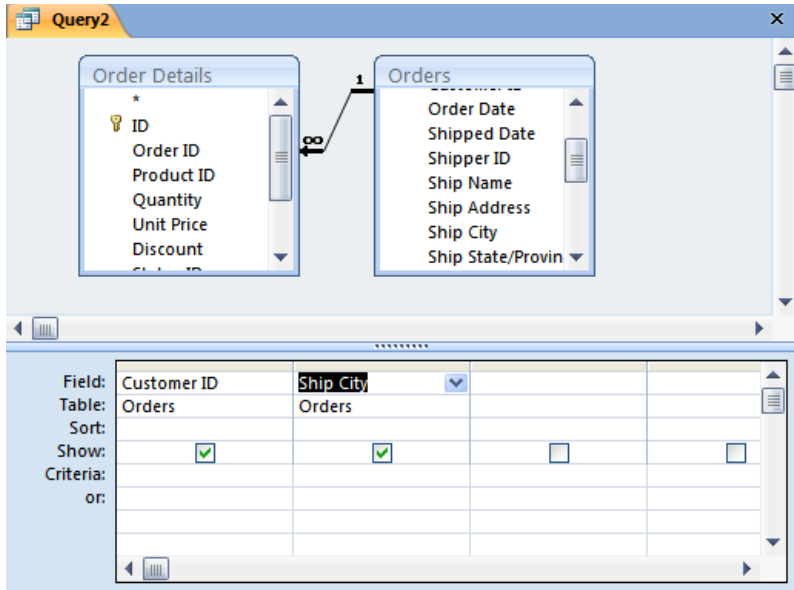


**Bước 4:** Trong hộp thoại Show Table nhập đúp vào mục Order Details và Orders.

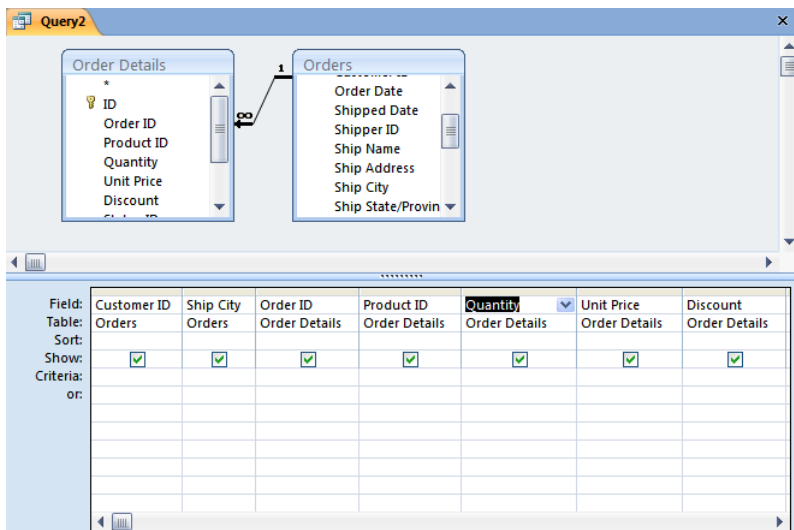


**Bước 5:** nhấp Close để đóng hộp thoại Show Table.

**Bước 6:** Trên bảng Orders bạn nhấp đúp vào lần lượt các trường Customer ID và Ship City để bổ xung những trường này lên lưới thiết kế Query.



**Bước 7:** Trên bảng Order Details làm tương tự như ở bước 6 với các trường Order ID, Product ID, Quantity, Unit Price và Discount.





**Bước 8:** Ở cột Ship City trên lưới thiết kế Query, xóa bỏ nút Check ở dòng Show. Trên dòng Criteria bạn gõ vào giá trị ‘Chicago’. Chạy thử Query để biết kết quả truy vấn trước khi chúng ta tạo bảng

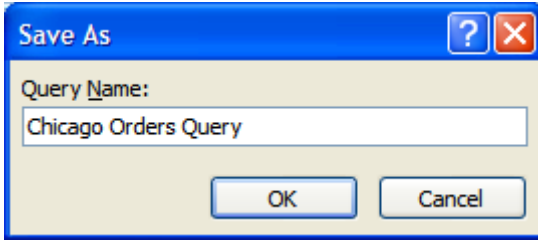
Field:	Customer ID	Ship City	Order ID	Product ID	Quantity	Unit Price	Discount
Table:	Orders	Orders	Order Details	Order Details	Order Details	Order Details	Order Details
Sort:							
Show:	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:		*Chicago					
or:							

**Bước 9:** Trên Tab Design tìm tới nhóm Results nhấp chọn Run, kết quả Query trả về như hình dưới đây:

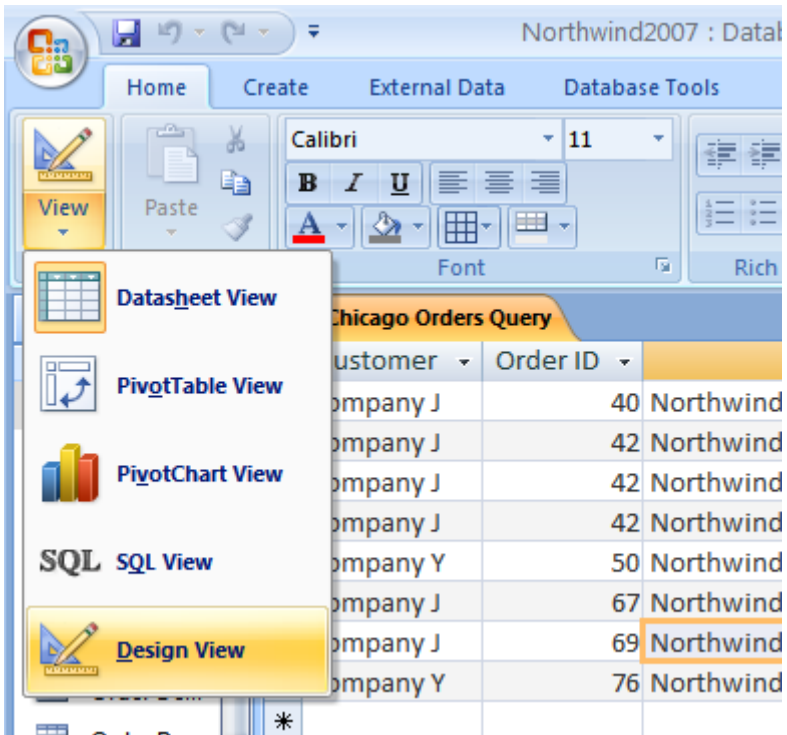
Customer	Order ID	Product	Quantity	Unit Price	Discount
Company J	40	Northwind Traders Green Tea	200	\$2.99	0.00%
Company J	42	Northwind Traders Boysenberry Spread	10	\$25.00	0.00%
Company J	42	Northwind Traders Cajun Seasoning	10	\$22.00	0.00%
Company J	42	Northwind Traders Chocolate Biscuits Mix	10	\$9.20	0.00%
Company Y	50	Northwind Traders Scones	20	\$10.00	0.00%
Company J	67	Northwind Traders Almonds	20	\$10.00	0.00%
Company J	69	Northwind Traders Dried Plums	15	\$3.50	0.00%
Company Y	76	Northwind Traders Cajun Seasoning	30	\$22.00	0.00%
*					

Record: 14 of 8

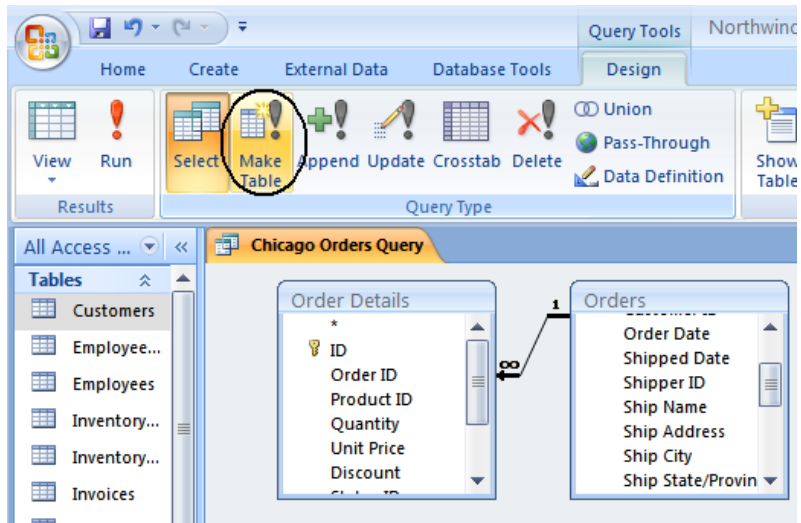
**Bước 10:** Bây giờ hãy lưu Query lại với tên Chicago Orders Query.



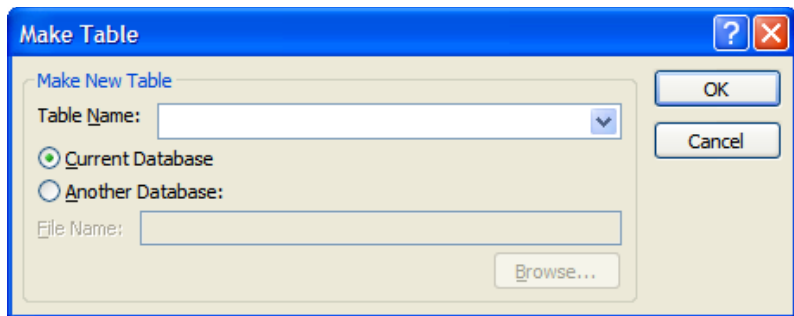
**Bước 11:** Trên Tab Home tìm tới nhóm Views nhấp chọn View trong hộp Combo để xuống bạn chọn Design View.



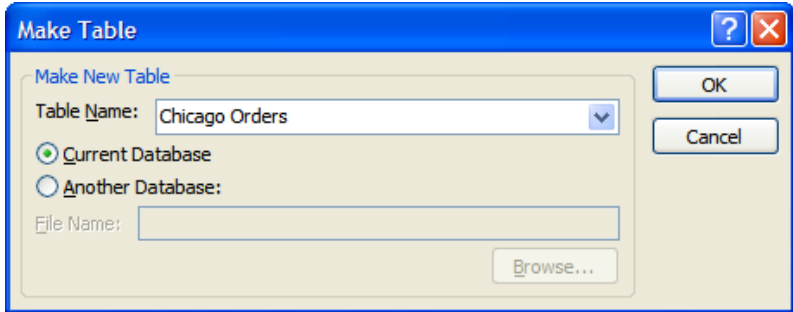
**Bước 12:** Trên Tab Design tìm tới nhóm Query Type bạn chọn Make Table.



Hộp thoại Make Table xuất hiện như hình dưới đây:

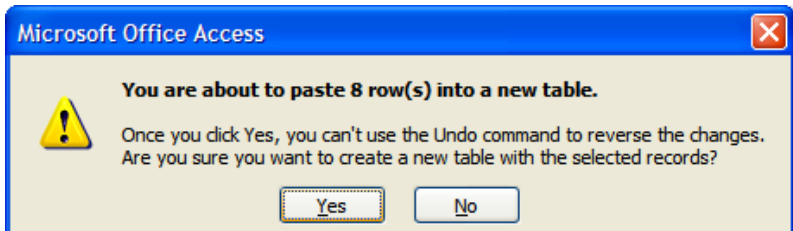


**Bước 13:** Trong hộp thoại này bạn tìm tới ô Table Name gõ vào đó để đặt tên cho bảng là Chicago Orders và nhấp Ok.

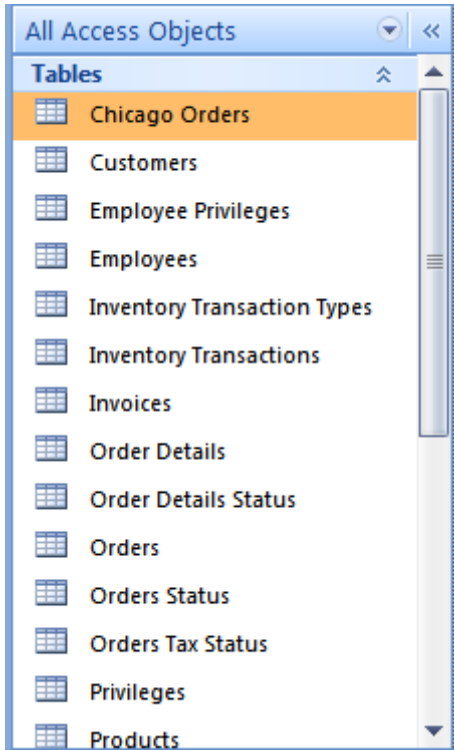


***Bước 14:*** Trên Tab Design tìm tới nhóm Results chọn Run.

***Bước 15:*** Một hộp thoại xác nhận xuất hiện bạn chọn Yes.



Một bảng mới sẽ được tạo trong Pane Navigation.



### 3.7. Thêm dữ liệu vào bảng từ nhiều bảng khác.

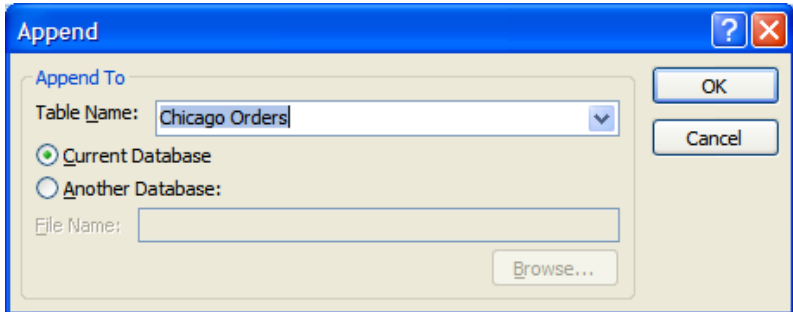
Bạn có thể sử dụng một Append Query để lấy dữ liệu từ một hoặc nhiều bảng vào một bảng khác.

Giả sử bạn tạo ra một bảng và chia sẻ nó với những khách hàng ở Chicago cùng trong hiệp hội, nhưng bạn nhận ra rằng hiệp hội cũng làm việc với những khách hàng ở khu vực Milwaukee. Bạn muốn thêm những dòng dữ liệu chứa những giao dịch với khu vực này trước khi chia sẻ cho hiệp hội.

Bạn có thể thêm dữ liệu về khu vực Milwaukee vào bảng Chicago Orders mà bạn đã tạo ra ở phần trước theo các bước thực hiện dưới đây:

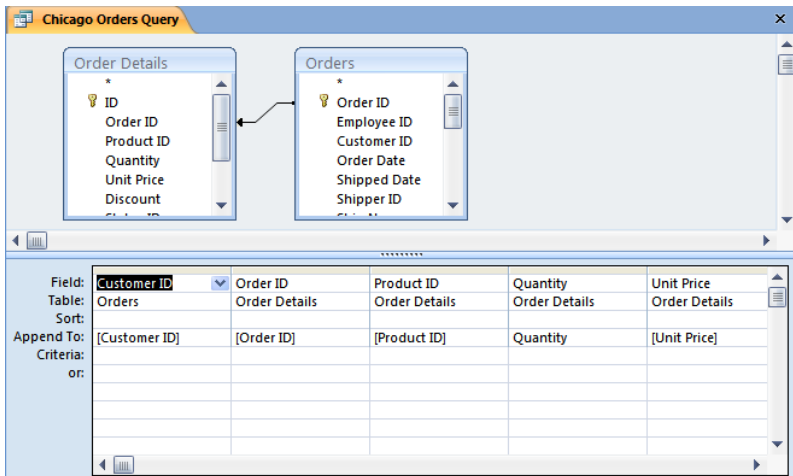
**Bước 1:** Mở Query có tên Chicago orders Query ở chế độ Design.

**Bước 2:** Trên Tab Design tìm tới nhóm Query Type nhập chọn Append, hộp thoại Append xuất hiện.

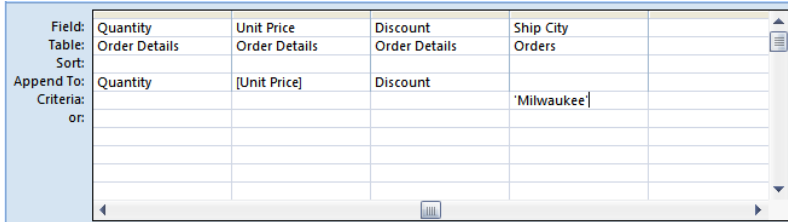


Trong Combo Table Name bạn chọn Chicago Orders sau đó nhập Ok.

Khi hộp thoại Append đóng lại trên lưới Design xuất hiện thêm dòng Append và dòng Show biến mất.



**Bước 3:** Trên lưới thiết kế ở dòng Criteria của cột Ship City, bạn xóa giá trị 'Chicago' và thay vào đó bằng 'Milwaukee'.



Field:	Quantity	Unit Price	Discount	Ship City	
Table:	Order Details	Order Details	Order Details	Orders	
Sort:					
Append To:	Quantity	[Unit Price]	Discount		
Criteria:				*Milwaukee*	
or:					

**Bước 4:** Trên dòng Append To chọn trường thích hợp cho từng cột. Trong ví dụ này các trường đều có tên giống như các cột chỉ trừ trường điều kiện Ship City.

**Bước 5:** Trên Tab Design tìm tới nhóm Result chọn Run để chạy thử Query.

### 3.8. Thay đổi dữ liệu tự động

Bạn có thể sử dụng một Query Update để thay đổi dữ liệu trong các bảng của mình, và bạn có thể sử dụng các yêu cầu tìm kiếm với Query này để chọn ra những dòng đặc biệt cần được thay đổi. Query Update cung cấp cho bạn một cơ hội để xem xét các dữ liệu cần thiết trước khi thực hiện việc cập nhật.

**Chú ý:** Một hành động của Query khi thực hiện sẽ không thể hoàn tác vì vậy bạn cần phải xem xét để Back up dữ liệu của bất cứ bảng nào mà bạn sẽ cập nhật lại bằng cách sử dụng Query Update.

Trong ví dụ trước, Bạn đã thêm các dòng mới vào bảng Chicago Orders. Trong bảng này, trường Product ID hiển thị số Id của sản phẩm. Để làm cho dữ liệu dễ sử dụng hơn trên báo cáo, bạn có thể sẽ muốn thay thế các ID sản phẩm bằng tên của chúng. Để có thể thay thế các Id sản phẩm, đầu tiên bạn phải đổi kiểu dữ liệu của trường này từ kiểu số sang kiểu Text, để trường Product Id có thể chấp nhận được dữ liệu của trường Product Name.

Dưới đây là các bước thực hiện:

Bước 1: Mở bảng Chicago Orsers ở chế độ hiển thị Design.

Bước 2: Trên hàng Product ID đổi kiểu dữ liệu từ Number thành kiểu Text.

Bước 3: Lưu và đóng bảng Chocago Orders lại

Bước 4: Trên Tab Create tìm tới nhóm Other, chọn Query Design.

Bước 5: Trong hộp thoại Show Table, nhấp đúp chuột vào 2 bảng Chicago Orders và Products.

Bước 6: Nhấp Close để đóng hộp thoại Show Table.

Bước 7: Trên Tab Design tìm tới nhóm Query Type, nhấp chọn Update. Trên lưới thiết kế, 2 dòng Sort và Show biến mất và thay vào đó là dòng Update To.

Bước 8: Trong bảng Chicago Orders, nhấp đúp chuột vào trường Product Id để bổ xung trường này lên lưới thiết kế Query.

Bước 9: Trên lưới thiết kế, tại dòng Update To của cột Product ID bạn nhập vào nội dung dưới đây:

[Products].[Product Name]

Bước 10: Trên dòng Criteria bạn nhập vào nội dung dưới đây:

[Product ID] Like ([Products].[ID])

Từ khóa Like được sử dụng trong trường hợp này là rất cần thiết bởi vì những trường mà bạn so sánh nội dung khác nhau về kiểu dữ liệu (trường Product ID là kiểu Text bạn vừa đổi, và trường ID là kiểu Number).

Bước 11: Bạn có thể xem xét các giá trị sẽ bị thay đổi khi thực hiện Query Update bằng cách hiển thị Query ở chế độ Datasheet.



Trên Tab Design, tìm tới nhóm Results nhấp chọn View, trong danh sách dropdown bạn chọn Datasheet View.

Kết quả trả về từ Query là một danh sách các Product Id mà bạn sẽ Update.

*Bước 12:* Trên Tab Design tìm tới nhóm Results nhấp chọn Run.

Khi bạn mở bảng Chicago Orders, bạn sẽ nhìn thấy các giá trị số trong trường Product ID được thay thế bởi tên các sản phẩm trong bảng Products.

### **3.9. Xóa dữ liệu tự động**

Bạn có thể sử dụng một Query Delete để xóa dữ liệu trong các bảng, và bạn cũng có thể sử dụng một Query Delete với các điều kiện tìm kiếm để có thể xóa những bản ghi đặc biệt. Một Query Delete cung cấp cho bạn cơ hội để xem xét các bản ghi sẽ được xóa bỏ trước khi bạn thực hiện việc xóa dữ liệu.

Giả sử bạn đang chuẩn bị dữ liệu trong bảng Chicago Orders để gửi tới hiệp hội khác hàng ở Chicago, và bạn nhận thấy rằng một số hàng chứa các giá trị rỗng. Bạn muốn loại bỏ các hàng này trước khi gửi bảng dữ liệu đi. Rất đơn giản, bạn có thể mở bảng và tìm tới những hàng đó để xóa một cách thủ công, nhưng bạn sẽ gặp vấn đề lớn khi số lượng hàng cần xóa nhiều lên và nằm rải rác ở các vị trí khác nhau. Khi đó bạn sẽ thấy được tác dụng của Query Delete.

Dưới đây là các bước tạo một Query để xóa các dòng trong bảng Chicago Orders với điều kiện trường Order ID không có giá trị, và dưới đây là các bước thực hiện.

*Bước 1:* Trên Tab Create tìm tới nhóm Other chọn Query Design.

Bước 2: Trong hộp thoại Show Table, nhấp đúp chuột chọn Chicago Orders.

Bước 3: Nhấp Close để đóng hộp thoại Show Table.

Bước 4: Trên Tab Design, tìm tới nhóm Query Type, nhấp chọn Delete.

Trên lưới thiết kế, hàng Sort và Show biến mất thay vào đó là hàng Delete.

Bước 5: Trên bảng Chicago Orders nhấp đúp chuột vào trường Order Id để thêm nó vào lưới thiết kế.

Bước 6: Trên lưới thiết kế, ở dòng Criteria của cột Order ID nhập vào đó giá trị Is Null.

Bước 7: Trên Tab Design tìm tới nhóm Result nhấp chọn Run để chạy thử Query.

## **Bài 4. Làm việc với Form**

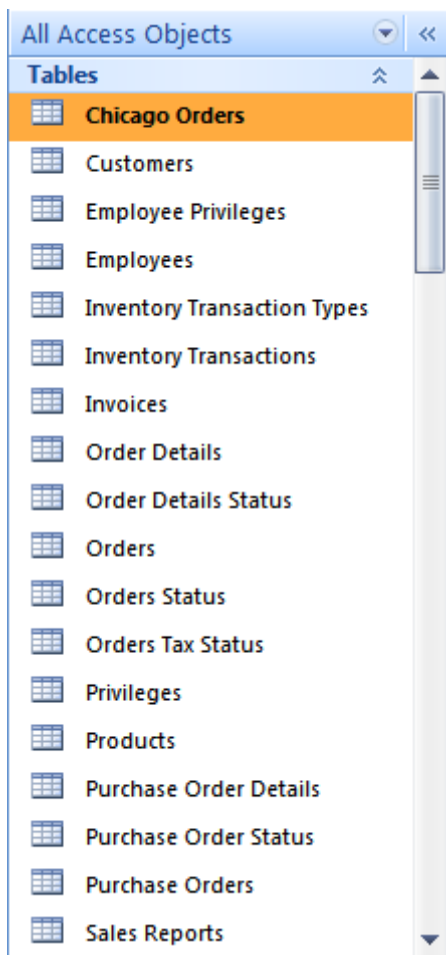
### **4.1. Tạo Form**

#### **Tạo Form đơn giản bằng Form**

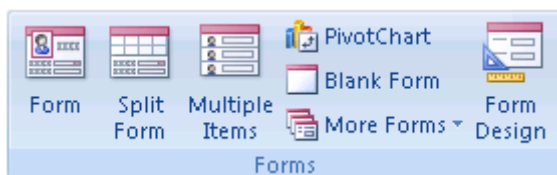
Bạn có thể sử dụng công cụ biểu mẫu (Form) để tạo ra một biểu mẫu chỉ với một lần nhấp chuột. Khi bạn sử dụng công cụ này, tất cả các trường từ các nguồn dữ liệu sẽ được đặt lên biểu mẫu. Bạn có thể bắt đầu sử dụng biểu mẫu ngay lập tức hoặc bạn cũng có thể chỉnh sửa nó ở chế độ Layout View hoặc Design View để nó phù hợp hơn với nhu cầu sử dụng của mình.

Các bước tạo Form bằng cách sử dụng công cụ Form.

*Bước 1:* Trên Pane Navigation, nhấp chọn Table hoặc Query chứa dữ liệu mà bạn muốn hiển thị lên Form.



**Bước 2:** Từ Tab Create tìm tới nhóm Forms, nhấp chọn Form.



Access sẽ tạo ra một Form và hiển thị nó ở chế độ Layout View. Trong Layout View, bạn có thể thực hiện các thay đổi về mặt hình thức ngay cả khi Form đang hiển thị dữ liệu. Ví dụ, bạn có thể điều chỉnh kích thước của các TextBox để chúng phù hợp với những dữ liệu đang hiển thị.

Customer ID:	10
Order ID:	40
Product ID:	81
Quantity:	200
Unit Price:	\$2.99
Discount:	0

Nếu Access tìm thấy một bảng mà có mối quan hệ một nhiều với một bảng hoặc một Query khác cùng sử dụng để tạo nên Form, Access sẽ thêm một Datasheet nữa vào Form dựa trên mối quan hệ của bảng với một bảng hoặc một Query khác. Ví dụ, nếu bạn tạo một biểu mẫu đơn giản dựa trên các bảng Employees, và có một mối quan hệ một nhiều được xác định giữa bảng Employees và bảng Orders, một Datasheet hiển thị tất cả các bản ghi của bảng Orders mà liên quan đến bản ghi hiện thời của bảng Employee. Bạn có thể xóa các Datasheet

trên Form nếu bạn quyết định không cần đến nó. Trong trường hợp có nhiều hơn một bảng có mối quan hệ một nhiều với bảng mà bạn sử dụng để tạo Form, Access cũng sẽ không thêm các Datasheet khác vào biểu mẫu.

### **Tạo Split Form bằng công cụ Split Form.**

Split Form là một chức năng mới trong Microsoft Office Access 2007 nó cung cấp cho bạn hai chế độ hiển thị dữ liệu cùng một lúc – Form View và Datasheet View.

Hai chế độ hiển thị được kết nối tới cùng một nguồn dữ liệu và đồng bộ với nhau tại mọi thời điểm. Chọn một trường trong phần này cũng chính là chọn trường đó trong phần còn lại của biểu mẫu. Bạn có thể thêm, chỉnh sửa, hoặc xóa dữ liệu từ một trong hai phần hiển thị.

Làm việc với các Split Form sẽ cung cấp cho bạn những lợi ích của cả hai kiểu hiển thị dữ liệu nhưng lại được bố trí trên một biểu mẫu duy nhất. Ví dụ, bạn có thể sử dụng phần Datasheet của biểu mẫu để tìm kiếm các bản ghi và sử dụng phần còn lại để hiển thị chi tiết và chỉnh sửa dữ liệu.

Các bước tạo một Split Form bằng cách sử dụng công cụ Split Form.

*Bước 1:* Trên Pane Navigation, nhấp chọn Table hoặc Query chứa dữ liệu mà bạn muốn tạo Form, tiếp theo bạn mở bảng hoặc Query đó ở chế độ Datasheet View.

*Bước 2:* Trên Tab Create tìm tới nhóm Form chọn Split Form.

Access sẽ tạo ra một Form mới và hiển thị trên Layout View. Trong Layout View, bạn có thể thay đổi các thiết kế cho Form ngay khi nó đang hiển thị dữ liệu.

The screenshot shows the 'Customers' form in Microsoft Access. The form is titled 'Customers' and has a tab labeled 'Customers'. The form displays the following fields and values:

- ID: 1
- Company: Company A
- First Name: Anna
- Last Name: Bedecs
- E-mail Address: (empty)
- Job Title: Owner
- Business Phone: (123)456-7890
- Home Phone: (empty)

Below the form is a data table with the following data:

ID	Company	First Name	Last Name	E-mail Address
1	Company A	Anna	Bedecs	
2	Company B	Antonio	Gratacos Solso	
3	Company C	Thomas	Axen	
4	Company D	Christina	Lee	
5	Company E	Martin	O'Donnell	
6	Company F	Francisco	Pérez-Olaeta	
7	Company G	Ming-Yang	Xie	
8	Company H	Elizabeth	Andersen	


The status bar at the bottom of the form shows 'Record: 1 of 29', 'No Filter', and a search box.

### Tạo Form sử dụng công cụ Multiple Items.

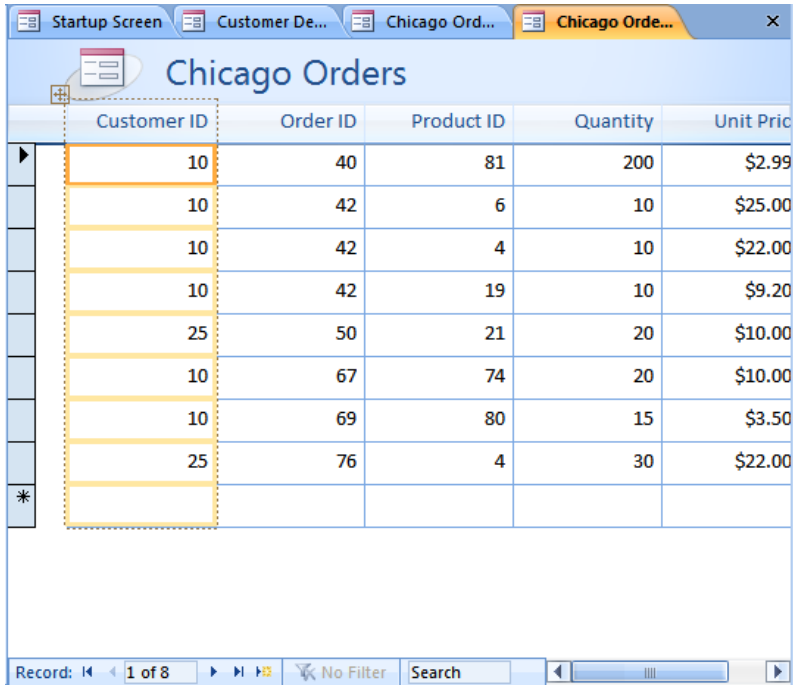
Khi bạn tạo một Form bằng cách sử dụng công cụ Simple Form, tại một thời điểm Form này chỉ hiển thị được duy nhất một bản ghi. Nếu bạn muốn có một Form có thể hiển thị nhiều bản ghi và có thể tùy chỉnh tiện dụng hơn ở trên Datasheet một lúc bạn có thể sử dụng công cụ Multiple Items.

Dưới đây là các bước thực hiện

**Bước 1:** Trên Pane Navigation nhấp chọn Table hoặc Query chứa dữ liệu mà bạn muốn hiển thị lên Form.

**Bước 2:** Trên Tab Create tìm tới nhóm Forms nhấp chọn Multiple Items .

Access sẽ tạo ra một Form và hiển thị nó ở chế độ Layout View. Ở chế độ này bạn có thể thay đổi các thiết kế cho Form ngay khi nó đang hiển thị dữ liệu. Ví dụ như bạn có thể thay đổi độ rộng các TextBox để chứa đủ dữ liệu, ...



Customer ID	Order ID	Product ID	Quantity	Unit Price
10	40	81	200	\$2.99
10	42	6	10	\$25.00
10	42	4	10	\$22.00
10	42	19	10	\$9.20
25	50	21	20	\$10.00
10	67	74	20	\$10.00
10	69	80	15	\$3.50
25	76	4	30	\$22.00
*				

Khi bạn sử dụng công cụ Multiple Items, Form được tạo ra sẽ tương tự như khi chúng ta hiển thị bảng ở chế độ Datasheet. Dữ liệu trên Form được sắp xếp theo các hàng và cột và tại một thời điểm cho phép bạn thấy nhiều hơn một bản ghi. Tuy nhiên, một Form Multiple Items Form cho phép bạn có thể tùy chỉnh nhiều hơn khi hiển thị dữ liệu trong bảng ở chế độ Datasheet, chẳng hạn như khả năng thêm các



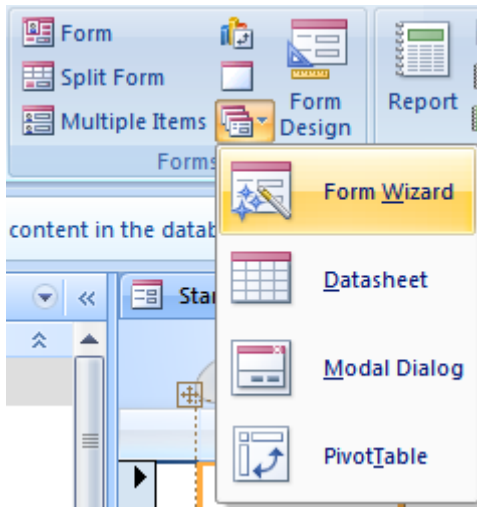
yếu tố đồ họa vào Form, các nút Button hay nhiều các điều khiển khác.

### **Tạo Form sử dụng Form Wizard.**

Để có thêm lựa chọn về những trường gì xuất hiện trên Form, bạn có thể sử dụng công cụ Form Wizard thay vì các công cụ mà chúng ta đã được tìm hiểu ở các phần trước. Bạn cũng có thể xác định xem dữ liệu như thế nào thì được nhóm, được sắp xếp, và bạn cũng thể sử dụng các trường từ nhiều hơn một bảng hoặc một Query.

Các bước tạo Form với công cụ Form Wizard.

Bước 1: Trên Tab Create, tìm tới nhóm Forms, nhấp chọn More Form rồi chọn Form Wizards.



Bước 2: Theo các bước hướng dẫn trên các hộp thoại Form Wizards.

Chọn trường cần hiển thị lên Form.

Form Wizard

Which fields do you want on your form?  
You can choose from more than one table or query.

Tables/Queries  
Table: Chicago Orders

Available Fields:

Selected Fields:


- Customer ID
- Order ID
- Product ID
- Quantity
- Unit Price
- Discount

Cancel < Back Next > Finish

### Chọn Layout cho Form

Form Wizard

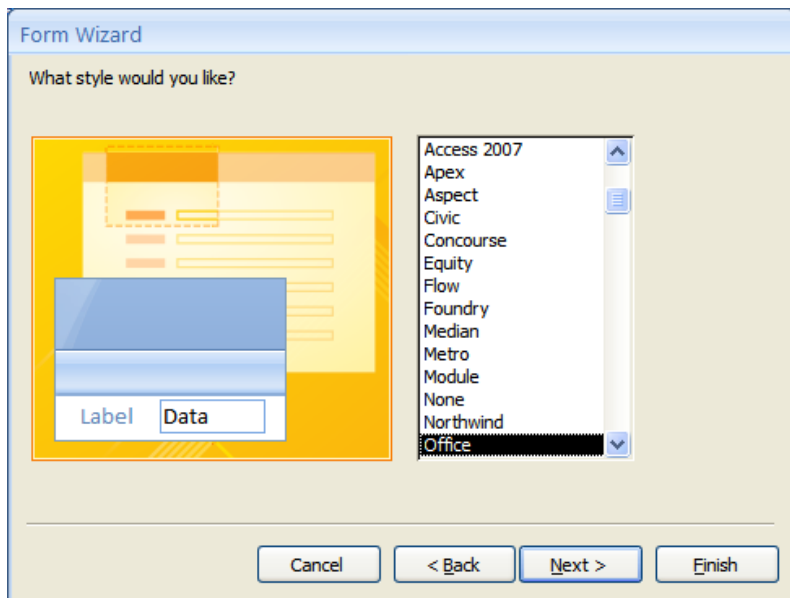
What layout would you like for your form?



Columnar  
 Tabular  
 Datasheet  
 Justified

Cancel < Back Next > Finish

Chọn kiểu Style cho Form.



Bước 3: Cuối cùng nhấp Finish để hoàn tất.

Form Wizard

What title do you want for your form?

Chicago Orders Winzard

That's all the information the wizard needs to create your form.

Do you want to open the form or modify the form's design?

Open the form to view or enter information.

Modify the form's design.

Cancel < Back Next > Finish


Kết quả chúng ta sẽ được Form như hình dưới đây.

Field Name	Value
Customer ID	10
Order ID	40
Product ID	81
Quantity	200
Unit Price	\$2.99
Discount	0

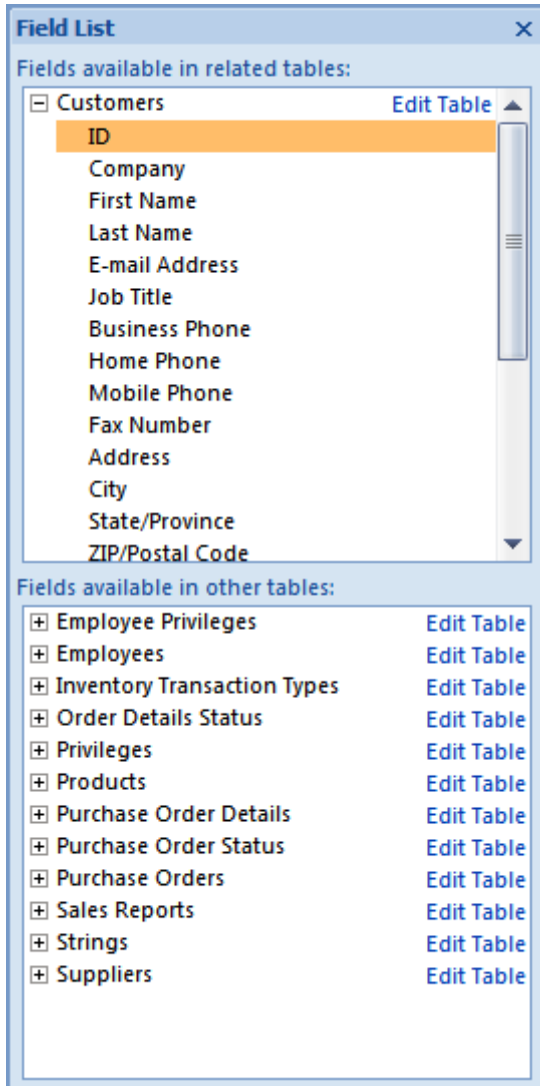
## Tạo Form sử dụng công cụ Blank Form

Nếu Winzard hoặc một công cụ tạo Form nào trước đây mà chưa đáp ứng được yêu cầu của bạn, bạn có thể sử dụng chức năng Blank Form. Công cụ này là cách nhanh nhất để xây dựng một Form, đặc biệt nếu bạn dự định chỉ hiển thị một vài trường lên Form.

Các bước tạo Form với công cụ Blank Form

**Bước 1:** Trên Tab Create tìm tới nhóm Forms nhấp chọn Blank Form .

Access sẽ mở một Form trắng ở chế độ Design View, và hiển thị Pane Field List.

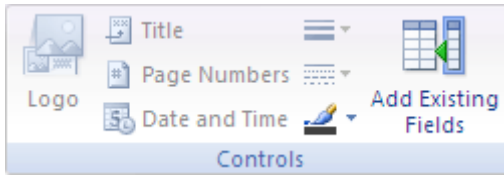



**Bước 2:** Trên Pane Field List, nhấp chọn vào nút có hình (+) để xem chi tiết các trường trong bảng hoặc Query mà bạn muốn hiển thị lên Form.

**Bước 3:** Để thêm một trường lên Form bạn nhấp chọn trường rồi kéo thả lên Form. Để thêm nhiều trường cùng một lúc bạn giữa phím Ctrl sau đó tích chọn những trường cần thêm rồi kéo thả cùng một lúc lên Form.

**Bước 4:** Sử dụng các công cụ trong nhóm Controls trên Tab Formatting để thêm Logo, tiêu đề, số trang, hoặc ngày tháng,

....



**Bước 5:** Nếu bạn muốn thêm các Control phức tạp hơn lên Form bạn phải chuyển sang chế độ Design View, muốn chuyển về chế độ này bạn nhấp phải chuột vào Form rồi chọn Design View , sau đó sử dụng các công cụ trong nhóm Controls ở trên Tab Design để thêm chúng lên Form.



## 4.2. Tìm hiểu về các Controls.

Controls là các đối tượng dùng để hiển thị dữ liệu, thực hiện các sự kiện, và cho phép bạn xem và làm việc với các thông tin đó trên giao diện người dùng, ví dụ như các hình ảnh hay các nhãn thông báo. Control được sử dụng nhiều nhất là TextBox các điều khiển khác có thể kể đến như Label, CheckBox, ....

Controls có thể là Control ràng buộc, không ràng buộc hoặc Control tính toán.

- Điều khiển ràng buộc: Một điều khiển có các nguồn dữ liệu là một trường trong một bảng hoặc Query được gọi là Control ràng buộc. Bạn sử dụng các Control (điều khiển) ràng buộc để hiển thị các giá trị của trường trong cơ sở dữ liệu. Những giá trị đó có thể dạng văn bản, dạng số, ngày tháng, hình ảnh, kiểu logic, ... Ví dụ, một TextBox trên Form hiển thị tên của một nhân viên, điều khiển này lấy dữ liệu từ trường Last Name của bảng Employees.

- Điều khiển không ràng buộc: Một Control mà không được gắn với một nguồn dữ liệu (chẳng hạn như một trường hay biểu thức) được gọi là một Control không kiểm soát. Bạn sử dụng các điều khiển này để hiển thị thông tin, các đường Line, hình chữ nhật, hay các hình ảnh. Ví dụ, một Label hiển thị tiêu đề của một Form được gọi là một Control không kiểm soát.

- Điều khiển tính toán: Một điều khiển có nguồn dữ liệu là một biểu thức chứ không phải là một trường, được gọi là điều khiển tính toán. Bạn chỉ rõ giá trị mà bạn muốn sử dụng như là nguồn của dữ liệu trên điều khiển bằng cách định nghĩa một biểu thức. Ví dụ sau đây tính giá của một danh mục sau khi đã trừ đi 25% giảm giá, giá trị còn lại chỉ là 0.75.

$$=[\text{Unit Price}] * 0.75$$

Một biểu thức có thể sử dụng dữ liệu từ một trường trong bảng hoặc Query, hoặc dữ liệu từ các điều khiển khác trên Form .

Khi bạn tạo ra một Form, có lẽ nó hiệu quả nhất để thêm và sắp xếp tất cả các điều khiển ràng buộc đầu tiên, đặc biệt là nếu chúng chiếm phần lớn các điều khiển về form. Sau đó bạn có thể thêm vào điều khiển không ràng buộc và điều khiển tính toán rằng việc thiết kế hoàn toàn bằng cách sử



dụng các công cụ trong các nhóm Controls trên tab Design trong Design view.

Bạn tạo ràng buộc để điều khiển một trường xác định các trường từ đó kiểm soát được các dữ liệu. Bạn có thể tạo một điều khiển ràng buộc với trường được lựa chọn bằng cách kéo các kéo từ ô Field List vào form. Ô Field List hiển thị những trường của các bảng hoặc truy vấn. Để hiển thị ô Field List, nhấn ALT + F8, hoặc trên tab Design, trong nhóm Tools, nhấp chọn Add Existing Field. Khi bạn nhấp đúp chuột vào một trường trong pane Field List, Access sẽ thêm một điều khiển tương ứng với trường đó trên Form.

Ngoài ra, bạn có thể ràng buộc một trường vào một điều khiển bằng cách gõ trực tiếp tên trường vào điều khiển đó, hoặc thiết lập giá trị cho thuộc tính Control Source của điều khiển đó là tên trường của trường dữ liệu. Các thuộc tính xác định các đặc tính của Control như là tên, nguồn dữ liệu, hoặc là định dạng, để mở hộp thoại Property sheet hãy nhấn phím F4.

Sử dụng Pane Field List là cách tốt nhất để tạo một Control ràng buộc vì hai lý do sau đây:

- Access sẽ tự động đính kèm một điều khiển Label có sẵn tên của trường đó, vì vậy bạn không phải gõ các nhãn cho những Control của mình.
- Access sẽ tự động thiết lập rất nhiều thuộc tính phù hợp với giá trị của trường mà bạn đang làm việc (ví dụ như Format, Decimal Places, Input mask)

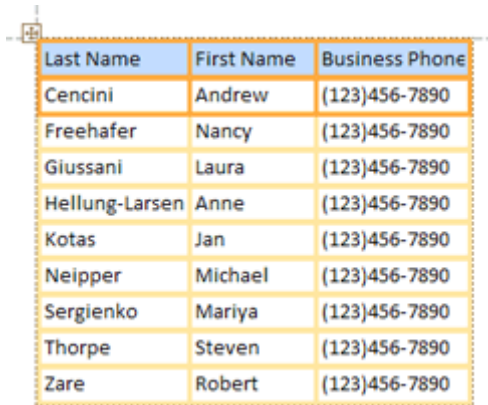
Nếu bạn đã tạo ra một điều khiển không ràng buộc và bạn lại muốn ràng buộc nó vào một trường nào đó, bạn hãy thiết lập thuộc tính Control Source cho điều kiện này giá trị là tên trường cần ràng buộc giá trị.

### **4.3. Tìm hiểu về Control Layout.**

Điều khiển Layouts cho phép bạn căn chỉnh các điều kiện trên Form theo cả chiều ngang và chiều dọc. Để chúng xuất hiện trên Form theo một định dạng thống nhất. Bạn có thể xem Control Layout như một bảng, ở đó chứa các ô mà nội dung trong mỗi ô đó là các Control. Sau đây là cách thêm, loại bỏ, hoặc sắp xếp các điều kiện trên Control Layout.

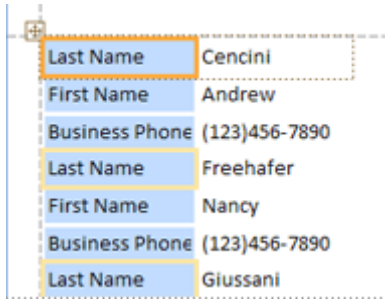
Điều khiển Layout có 2 dạng một là dạng bảng và dạng thứ 2 là dạng ngăn xếp (Stacked).

- Trên điều khiển Layout dạng bảng, các điều khiển trong nó được sắp xếp thành các hàng và cột giống như một bảng biểu, với các Label được đặt trên đầu. Điều khiển Layout dạng bảng luôn chia Form thành 2 vùng, một chứa các điều khiển hiển thị ở phía dưới, và một là vùng phía trên chứa các Label.



Last Name	First Name	Business Phone
Cencini	Andrew	(123)456-7890
Freehafer	Nancy	(123)456-7890
Giussani	Laura	(123)456-7890
Hellung-Larsen	Anne	(123)456-7890
Kotas	Jan	(123)456-7890
Neipper	Michael	(123)456-7890
Sergienko	Mariya	(123)456-7890
Thorpe	Steven	(123)456-7890
Zare	Robert	(123)456-7890

Trên điều khiển Layout kiểu ngăn xếp, các điều khiển trên đó được bố trí theo chiều dọc giống như trên một trang giấy, với mỗi Label bên trái của từng điều khiển. Khác với điều khiển Layout kiểu bảng, điều khiển Layout kiểu ngăn xếp luôn chứa các Control ở một khu vực duy nhất trên Form.



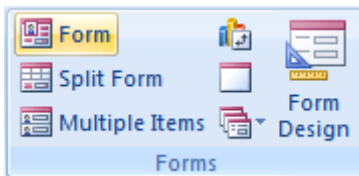
Last Name	Cencini
First Name	Andrew
Business Phone	(123)456-7890
Last Name	Freehafer
First Name	Nancy
Business Phone	(123)456-7890
Last Name	Giussani

Bạn có thể thiết kế nhiều Control Layout ở một trong hai kiểu nói trên, trên cùng một Form. Ví dụ, bạn có thể có một Control Layout dạng bảng để tạo dòng dữ liệu cho từng bản ghi, và một hoặc nhiều Control Layout dạng ngăn xếp ở phía dưới chứa nhiều hơn các thông tin của cùng bản ghi đó.

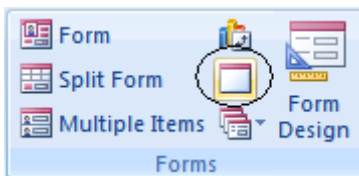
### Tạo mới một Control Layout

Access tự động tạo ra Control Layout dạng ngăn xếp trong các trường hợp sau đây:

- Tạo một Form mới bằng cách nhấp chuột vào nút Form trong nhóm Forms trên Tab Create.



- Bạn tạo một Form mới bằng cách nhấp vào nút Blank Form trong nhóm Forms trên Tab Create sau đó kéo thả các trường cần hiển thị lên Form trong Pane Field List.







Trên một Form đã tồn tại bạn có thể tạo mới Control Layout theo các bước dưới đây:

***Bước 1:*** Chọn điều khiển mà bạn muốn bỏ xung vào Layout.

***Bước 2:*** Nếu bạn muốn thêm nhiều Control khác nhau lên cùng một Layout, bạn giữ phím Shift sau đó tích chọn những Control cần thêm.

***Bước 3:*** Sau đó có thể làm theo một trong những cách sau đây:

- Trên Tab Arrange, tìm tới nhóm Control Layout nhấp chọn Tabular  hoặc Stacked .

- Nhấp phải chuột chọn một hoặc nhiều Control, chỉ tới Layout sau đó chọn Tabular  hoặc Stacked .



Bằng một trong 2 cách trên Access sẽ tạo Control Layout và thêm các điều khiển được chọn lên đó.



### **Chuyển một Control Layout từ dạng bảng sang dạng ngăn xếp và ngược lại.**

Để chuyển kiểu của một Control Layout từ dạng này sang dạng khác bạn làm như sau:

- Chọn Control Layout bằng cách nhấp chọn vào ô Select ở góc trái của Layout. Toàn bộ các Cell trong Layout sẽ được chọn.

- Sau đó bạn làm theo một trong các cách sau:

+ Trên Tab Arrange tìm tới nhóm Control Layout nhấp chọn kiểu Layout muốn tạo (Tabular  hoặc Stacked .

+ Nhấp phải chuột chọn một hoặc nhiều Control, chỉ tới Layout sau đó chọn Tabular  hoặc Stacked .



Access sẽ tự động sắp xếp lại cá Control tron Layout theo dạng mà bạn đã chọn.



### **Chia một Control thành hai dạng.**

Bạn có thể chia một Control Layout thành dạng khác nhau theo các bước dưới đây:

Bước 1: Giữa phím Shift sau đó nhấp chọn các Control mà bạn muốn di chuyển thành một dạng Control Layout mới.

Bước 2: Sau đó làm theo một trong hai cách sau đây:

- Trên Tab Arange tìm tới nhóm Control Layout nhấp chọn kiểu Layout muốn tạo (Tabular  hoặc Stacked .

- Nhấp phải chuột chọn một hoặc nhiều Control, chỉ tới Layout sau đó chọn Tabular  hoặc Stacked .

Bằng một trong hai cách trên Access sẽ tạo ra một Control Layout mới và thêm lên đó những điều khiển mà bạn vừa chọn.

### **Sắp xếp lại các Control trên một Control Layout**

Bạn có thể di chuyển một Control trên Control Layout bằng cách kéo thả nó tới vị trí mà bạn muốn. Cũng như là việc di chuyển vị trí các trường trong bảng, Thanh cuộn ngang dọc sẽ cho biết nơi bạn có thể thả nút chuột.

Bạn cũng có thể di chuyển một Control từ một Control layout này sang một Control Layout khác mà chúng có cùng kiểu định dạng. Ví dụ, bạn có thể kéo một Control từ một Control Layout dạng ngăn xếp sang một Control Layout cũng dạng ngăn xếp khác chứ không phải một Control Layout dạng bảng.

### **Thêm Control vào một Control Layout**

Để thêm một trường mới từ Pane Field List vào một Control Layout đã tồn tại bạn làm như sau:



Kéo thả trường từ Pane Field List vào Control layout. Thanh cuộn ngang, dọc sẽ cho bạn biết nơi bạn sẽ thả giữ nút trên chuột.

Để thêm một Control đã tồn tại vào một Control Layout đã có bạn làm như sau:

Bước 1: Chọn Control đầu tiên mà bạn muốn thêm vào Control Layout.

Bước 2: Nếu bạn muốn thêm các nhiều Control trên cùng một Layout, bạn giữ phím Shift rồi tích chọn lần lượt các Control. Ngoài ra bạn cũng có thể chọn các control trên Layout khác.

Bước 3: Kéo thả các trường các trường được chọn vào Layout. Thanh cuộn ngang, dọc sẽ báo cho bạn biết vị trí của trường mà bạn có thể thả giữ chuột.

Bước 4: Trên Tab Arrange trong nhóm Control Layout nhấp chọn kiểu Layout mà bạn muốn đưa các Control vào đó. Nếu bạn muốn thêm các điều khiển vào Layout dạng bảng bạn chọn Tabular , ngược lại bạn chọn Stacked .

Sau bước này Access sẽ tạo ra một Layout mới và thêm các điều Control mà bạn đã chọn lên đó.


### **Loại bỏ Control trên một Control Layout.**


Việc loại bỏ Control từ một Control Layout cho phép bạn có thể thực hiện ở bất kỳ vị trí nào trên Form mà không bị ảnh hưởng tới các điều khiển khác

Bạn chọn tới Control cần loại bỏ trên Control Layout. Để lựa chọn nhiều Control một lúc bạn giữ phím Shift sau đó tích chọn nhưng Control cần loại bỏ. Để Lựa chọn tất cả các

Control trên Control Layout đó bạn nhấp chọn và hộp chọn ở góc trên bên phía trái của Control Layout.

Và tiếp theo hãy làm theo các bước dưới đây:

- Trên Tab Arrange tìm tới nhóm Control Layout nhấp chọn Remove .

- Nhấp phải chuột vào một trong những điều khiển được chọn trở tới mục Layout chọn Remove .

#### **4.4. Tìm Layout View và Design View.**

**Layout view:** Layout View là chế độ hiển thị trực quan nhất để bạn có thể sử dụng trong quá trình chỉnh sửa Form, ở chế độ này nó cung cấp cho bạn những công cụ mà gần như có thể thay đổi mọi thứ mà bạn muốn khi làm việc với Form trong Microsoft Office Access 2007. Ở chế độ Layout View, Form đang chạy thực sự, vì vậy bạn có thể nhìn thấy dữ liệu xuất hiện trên Form, nhưng vẫn có thể thay đổi những thiết kế của Form ở chế độ này. Đây là chế độ hiển thị tiện dụng nhất để thiết lập độ rộng của các điều khiển cũng như có thể thực hiện hầu hết các công việc bố trí sự bố cục của Form trên giao diện người dùng.

Một số công việc không thể thực hiện được trong chế độ Layout View mà yêu cầu phải chuyển sang chế độ Design View. Trong những tình huống đó Access sẽ hiển thị thông báo, rằng bạn cần phải chuyển sang chế độ hiển thị Design View để thực hiện những công việc đó.

**Design view:** Chế độ hiển thị Design View cho phép bạn xem chi tiết hơn về cấu trúc Form đang thiết kế. Chế độ này cho phép chúng ta có thể nhìn thấy các phần Header, Detail, Footer. Form khi hiển thị ở chế độ Design View sẽ không chạy vì vậy bạn không thể nhìn thấy dữ liệu mà nó chứa bên trong khi đang thay đổi thiết kế. Tuy nhiên có một số công

việc chính mà bạn có thể thực hiện dễ dàng ở chế độ này thay vì Layout View:

- Có thể thêm nhiều hơn các Control lên Form như các nhãn, hình ảnh, đường Line, hình chữ nhật.
- Chỉnh sửa nguồn dữ liệu cho các TextBox bằng cách sử dụng pane Property sheet.
- Thay đổi kích thước của các phần (sections) trên Form, ví dụ như phần tiêu đề, phần chi tiết.
- Thay đổi một số thuộc tính chính mà không thể thay đổi được ở chế độ Layout View.

#### **4.5. Tinh chỉnh Form**

##### **Làm việc ở chế độ hiển thị Layout View**

Sau khi bạn tạo ra một Form, bạn có thể dễ dàng tinh chỉnh các thiết kế của nó khi hiển thị ở chế độ Layout View. Sử dụng dữ liệu thực tế như hướng dẫn của bạn, bạn có thể sắp xếp lại các điều khiển và điều chỉnh kích cỡ của chúng. Cũng như có thể thay thế các Control mới trên Form và thiết lập các thuộc tính cho Form và các điều khiển trên đó.

Để chuyển về chế độ Layout View nhấp phải chuột vào Form đó trên pane Navigation Pane, tiếp theo chọn layout View




Access sẽ hiển thị Form ở chế độ Layout View.

Bạn có thể sử dụng Property Sheet để chỉnh sửa thuộc tính cho Form và các điều khiển trong các vùng (Section) trên Form. Muốn thị hộp thoại Property Sheet bạn nhấn phím F4.

Chúng ta cũng có thể sử dụng Pane Field List để thêm một trường từ một bảng hoặc một Query lên Form đang thiết kế. Dưới đây là một trong các cách hiển thị Pane Field List:



- Trên Tab Formatting, trong nhóm Controls chọn Add Existing Fields .

- Nhấp phím Alt + F8


Sau đó bạn có thể kéo thả trực tiếp các trường từ Pane Field List vào Form.

- Để thêm một trường đơn giản, bạn nhấp đúp chuột vào nó hoặc kéo thả nó từ Pane Field List vào một vùng nhất định trên Form nơi mà bạn cần hiển thị.

- Để thêm nhiều hơn một trường, bạn giữ phím Ctrl sau đó dùng chuột tích chọn những trường cần muốn thêm. Sau đó kéo thả chúng lên Form.

### **Làm việc ở chế độ hiển thị Design View.**


Bạn cũng có thể tinh chỉnh những thiết kế của Form khi làm việc với nó ở chế độ Design view. Bạn thêm các Control, các trường lên Form bằng cách bỏ xung chúng lên lưới thiết kế. Hộp thoại Property sheet sẽ cho phép bạn truy cập một số lượng lớn các thuộc tính có thể thiết lập để chỉnh sửa Form.

Để chuyển Form về chế độ Design View, hãy nhấp phải chuột lên tên Form trong Pane Navigation sau đó chọn Design View .

Access sẽ hiển thị Form ở chế độ Design View

Bạn có thể sử dụng hộp thoại Property Sheet để chỉnh sửa các thuộc tính cho Form cũng như các Control trên Form bằng cách nhấn phím F4.

Bạn có thể sử dụng Pane Field List để thêm một trường từ bảng hoặc Query vào thiết kế của Form. Sau đây là một số cách thực hiện

- Trên Tab Formatting, tìm tới nhóm Controls nhấp chọn Add Existing Fields .

- Nhấp tổ hợp phím Ctrl + F8

Bạn có thể kéo thả trực tiếp các trường từ Pane Field List lên Form.

- Để thêm một trường, hãy nhấp đúp chuột vào nó hoặc kéo thả từ Pane Field List lên Form.

- Để thêm nhiều hơn một trường, hãy giữ phím Ctrl rồi dùng chuột tích chọn những trường cần thiết. Sau đó kéo thả những trường này lên Form.

#### **4.6. Thêm Control lên Form**

Một số Control được tạo ra tự động, ví dụ như các Control phụ thuộc được tạo ra khi bạn thêm một trường từ Pane Field List vào Form. Nhiều Control khác cũng có thể được tạo ra ở chế độ Design View bằng cách sử dụng các công cụ trong nhóm Controls trên Tab Design.




Chú ý: nhiều công cụ trong nhóm Controls chỉ có thể truy cập khi form của bạn được mở ở chế độ Design.


#### **Xác định tên của một Control**

Để xác định tên của một Control bạn chỉ cần di chuột tới nó, Access sẽ hiển thị tên của Control đó trên Tooltip.

#### **Sử dụng các công cụ Wizard.**

Bạn có thể sử dụng công cụ Wizards để được hỗ trợ trong quá trình tạo các Command Button, list boxes, subforms,

combo box và option groups. Trên Tab **Design** tìm tới nhóm Controls, nếu nút **Use Control Wizards** nếu như nó không sáng .

Nếu bạn yêu cầu tạo các Control mà không cần sự trợ giúp của Wizards bạn hãy nhấp chọn Use Control Wizards để tắt chế độ này .

### **Tạo một Control sử dụng các công cụ trong nhóm Controls.**

***Bước 1:*** Nhấp chọn công cụ tương ứng với kiểu Control mà bạn muốn thêm, ví dụ để tạo một hộp kiểm bạn nhấp chọn Control TextBox.

***Bước 2:*** Nhấp chuột lên lưới thiết kế Form, nơi bạn muốn bắt đầu đặt góc trái của điều khiển đó. Nếu nhấp một chuột lên Form, Control sẽ được tạo với cỡ mặc định, còn không bạn có thể kéo thả chuột để tùy chỉnh cỡ của Control trên Form.

***Bước 3:*** Trong trường hợp lựa chọn công cụ Wizards, khi bạn chọn một Control, chức năng này sẽ hướng dẫn từng bước để bạn có thể tạo ra một Control.

***Bước 4:*** Trong trường hợp vị trí của Control trên Form chưa thích hợp, bạn có thể sắp xếp lại hoặc di chuyển chúng đến vị trí thích hợp theo các bước sau:

- Nhấp chọn Control trên Form.
- Kéo thả Control tới vị trí mà bạn muốn.


Nếu bạn sử dụng một Control Wizard, công cụ này sẽ lần lượt thực hiện các bước để giúp bạn kiểm soát việc Bind (ràng buộc) dữ liệu từ một trường vào Control đó. Tuy nhiên thủ tục này tạo ra một Control mà không bị ràng buộc bởi các trường dữ liệu. Nếu Control này là dạng Control có thể hiển thị dữ liệu (TextBox, CheckBox,...), bạn phải nhập vào

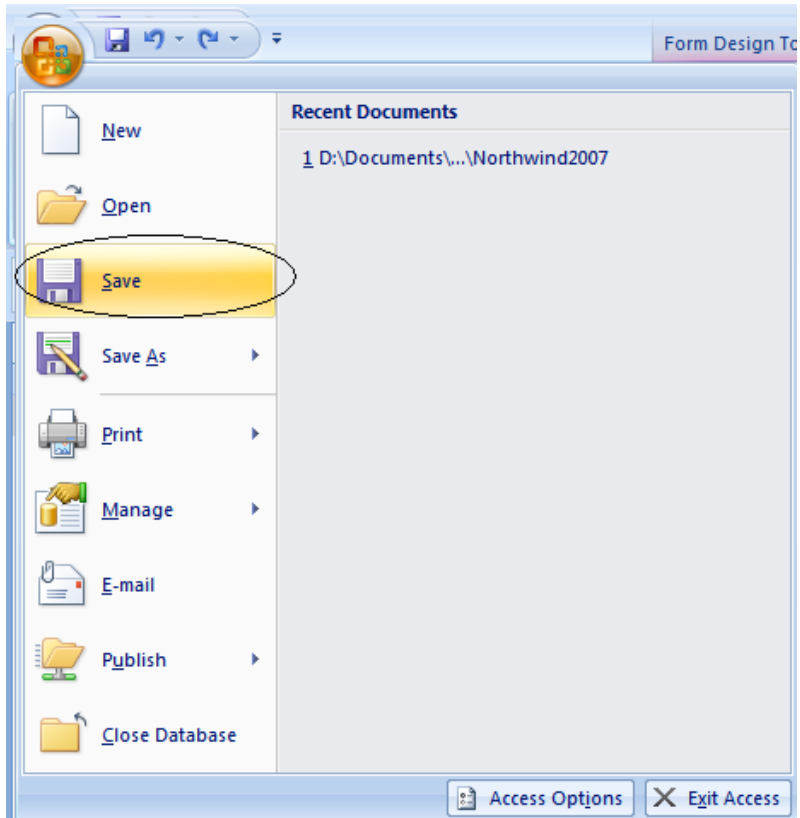
tên trường hoặc biểu thức cho giá trị của thuộc tính Control Source của Control, khi đó nó mới có thể hiển thị được dữ liệu bạn mong muốn. Để hiển thị bảng thuộc tính cho các Control hãy nhấp phím F4.

#### **4.7. Lưu lại các công việc của bạn**

Sau khi lưu lại Form như thiết kế, bạn có thể chạy nó thường xuyên nếu bạn muốn. Các thiết kế vẫn giữ nguyên, nhưng dữ liệu sẽ được hiển thị ở thời điểm bạn chạy Form. Nếu cần thay đổi bạn hoàn toàn có thể chỉnh sửa hoặc tạo mới Form dữ trên những gì mà đã làm trước đó.

#### **Lưu lại những thiết kế của Form**

Bước 1: Nhấp chọn nút Microsoft Office Button , sau đó chọn Save.



Cách khác

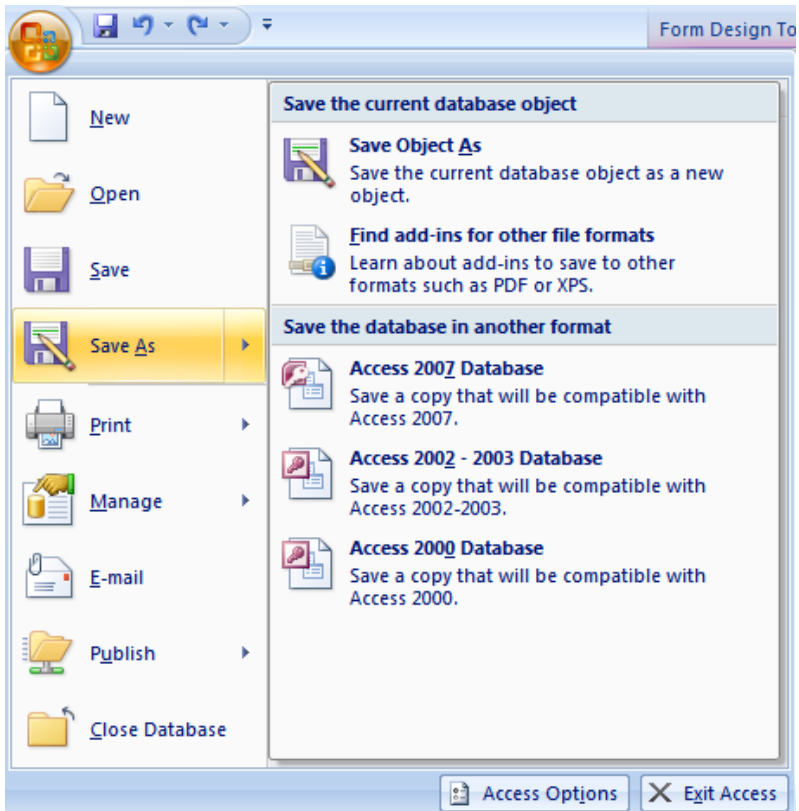
Bạn nhập chọn nút Save  trên thanh Quick Access Toolbar hoặc nhập tổ hợp phím Ctrl + S.

Bước 2: nếu Form được lưu lần đầu tiên một hộp thoại sẽ xuất hiện yêu cầu bạn nhập vào tên Form trong hộp nhập Form Name, cuối cùng nhập Ok để hoàn tất.

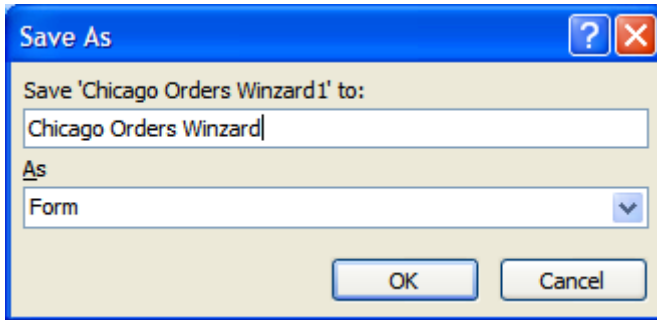
**Lưu lại thiết kế của Form với một tên khác.**

**Bước 1:** Nhấp chọn nút **Microsoft Office Button** , chọn Save As .

**Bước 2:** Trong hộp thoại Save As, bạn gõ tên Form mới trong hộp nhập Save Form To.



Chọn giá trị là Form trong hộp combo As.



Cuối cùng nhập Ok để hoàn tất.

## **Bài 5. Làm việc với Report**

Với Microsoft Office Access 2007 bạn có thể tạo ra nhiều loại báo cáo khác nhau, từ đơn giản đến phức tạp. Đầu tiên hãy nghĩ về báo cáo dùng để hiển thị những bản ghi từ nguồn dữ liệu. Cho dù là báo cáo với một danh sách đơn giản, hoặc báo cáo nhóm chỉ tiêu doanh thu theo vùng miền, thì việc đầu tiên bạn nên quan tâm là những trường chứa dữ liệu nào cần hiển thị lên báo cáo, và những bảng hay Query nào chứa chúng.

Sau khi bạn chọn được những bản ghi cần thiết, bạn có thể dễ dàng tạo ra một báo cáo bằng công cụ Report Wizard. Report Wizard là một chức năng trong Access, chức năng này sẽ hướng dẫn bạn thông qua các câu hỏi, báo cáo được tạo ra dựa trên những câu trả lời của bạn cho những câu hỏi đó.

### **5.1. Lựa chọn bản ghi nguồn.**

Một báo cáo bao gồm những thông tin được đưa vào từ các bản ghi trong một bảng hoặc một Query. cũng như các thông tin đã được lưu giữ cùng với lúc báo cáo được thiết kế thiết kế, chẳng hạn như nhãn, đề mục, và các hình ảnh. Các bảng hoặc truy vấn cung cấp các dữ liệu này còn được biết đến như là nguồn bản ghi của báo cáo. Nếu những trường mà bạn muốn sử dụng đều nằm trong một bảng thì bạn có thể sử dụng bảng đó như nguồn bản ghi. Nếu những trường sử dụng nằm ở nhiều bảng khác nhau bạn cần bạn sẽ phải sử dụng một hoặc nhiều Query như là nguồn bản ghi. Những Query này có thể đã tồn tại trong cơ sở dữ liệu của bạn, hoặc nếu cần bạn có thể tạo mới những Query để có thể chứa được những dữ liệu mà bạn muốn hiển thị lên báo cáo.

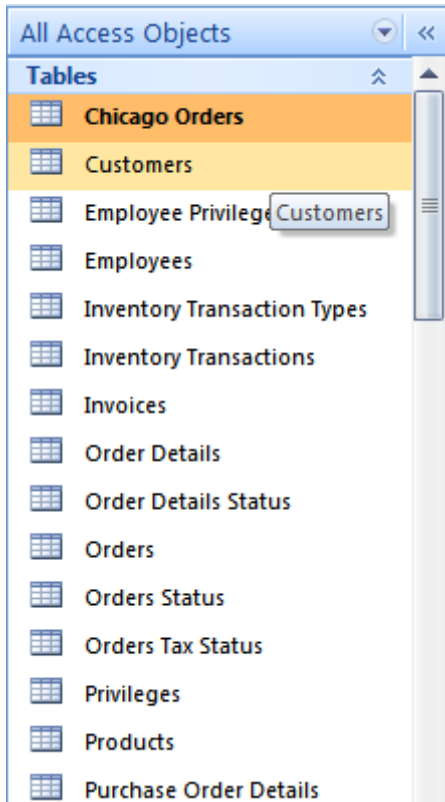
### **5.2. Tạo báo cáo**


#### **Sử dụng công cụ Report**

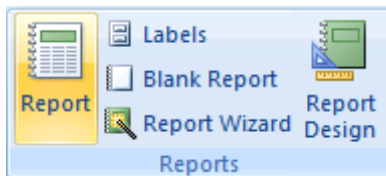


Các công cụ Report cung cấp con đường nhanh nhất để tạo ra một báo cáo. Bởi vì nó tự động tạo ra báo cáo mà không cần lấy thông tin hỏi đáp từ người thiết kế. Báo cáo hiển thị tất cả các trường từ nguồn dữ liệu của một bảng hoặc Query. Tuy các báo cáo sử dụng công cụ Report không hoàn hảo và không thể tạo ra những sản phẩm cuối cùng. Nhưng nó thật sự hữu dụng khi để hiển thị nhanh dữ liệu. Bạn có thể lưu lại báo cáo và chỉnh sửa nó ở chế độ hiển thị Layout View hoặc Design View sao cho có thể phục vụ tốt nhất mục đích của bạn.

*Bước 1:* Trong pane Navigation, nhấp chọn Table hoặc Query mà bạn muốn tạo báo cáo.



Bước 2: Trên Tab Create, trong nhóm Reports nhấp chọn Report .



Access sẽ tạo ra một báo cáo và hiển thị nó trên Layout View.

The screenshot shows a Microsoft Access report titled "Chicago Orders". The report displays a single record with the following data:

Customer ID:	10
Order ID:	40
Product ID:	81
Quantity:	200
Unit Price:	\$2.99
Discount:	0

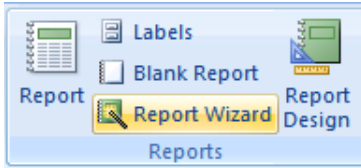
At the bottom of the report, there is a navigation bar showing "Record: 1 of 8" and a search field with the text "No Filter".

Sau khi xem các báo cáo, bạn có thể lưu và sau đó đóng cả báo cáo và các bảng hoặc Query mà bạn đã sử dụng làm các bản ghi nguồn. Tại một thời điểm nào đó cần mở lại báo cáo, khi đó Access sẽ hiển thị những dữ liệu mới nhất từ nguồn dữ liệu của báo cáo.

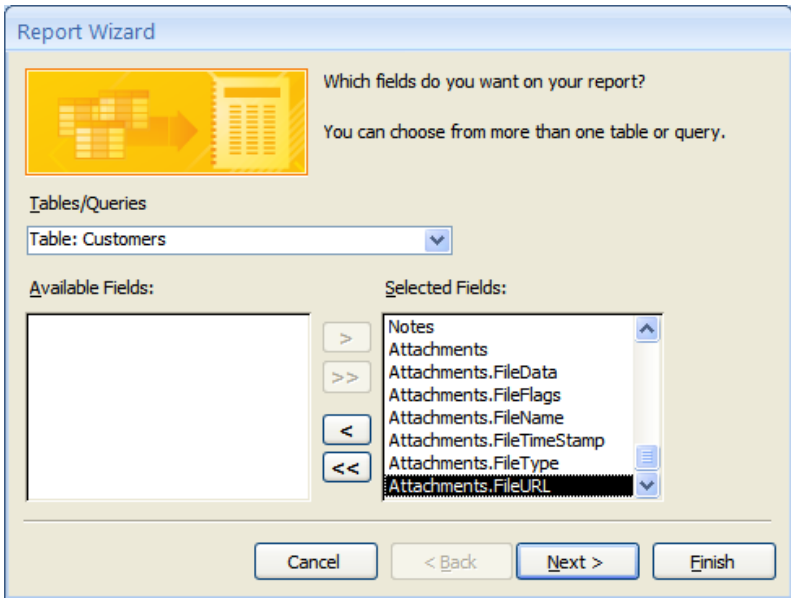
### **Sử dụng công cụ Report Wizard.**

Bạn sử dụng Report Wizard để có nhiều chọn lựa hơn về những trường sẽ xuất hiện trên báo cáo. Và cũng có thể chỉ ra các trường được nhóm hoặc sắp xếp, ngoài ra có thể sử dụng nhiều hơn một bảng hoặc Query làm dữ liệu nguồn cho báo cáo.

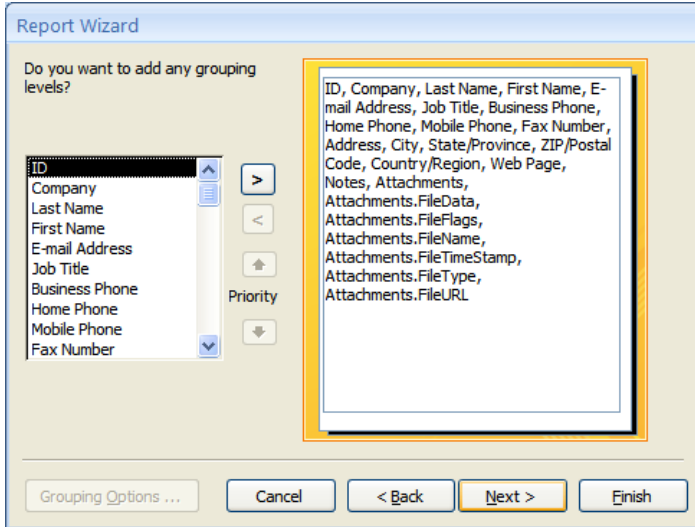
Bước 2: Trên Tab Create, trong nhóm Reports nhấp chọn Report Wizard.



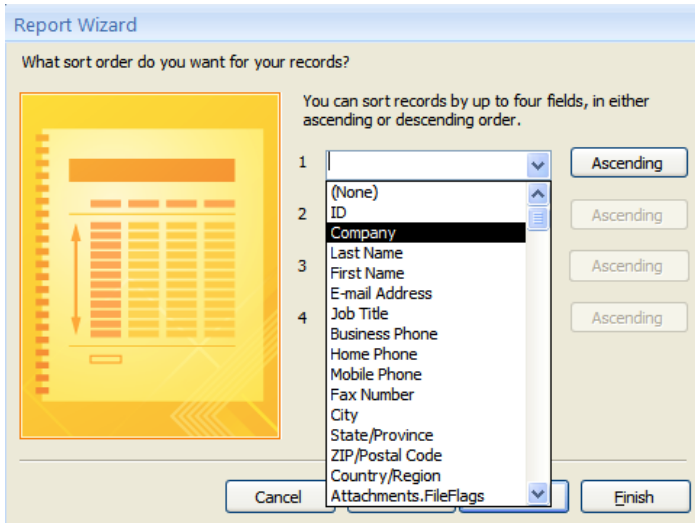
**Bước 3:** Chức năng Wizard sẽ hướng dẫn bạn từng bước tạo ra báo cáo:



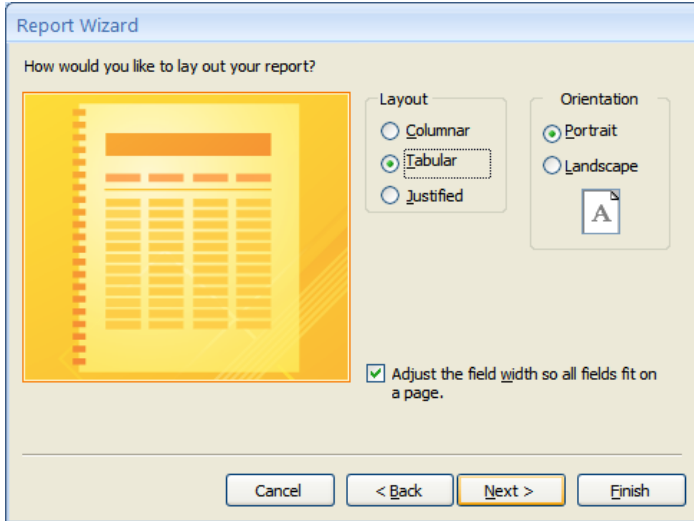
Hộp thoại này cho phép bạn lựa chọn các trường hiển thị trên báo cáo. Nhấp Next để chuyển bước tiếp theo.



Trong cửa sổ này cho phép bạn chọn trường cần nhóm. Nhấp Next để chuyển bước tiếp theo.



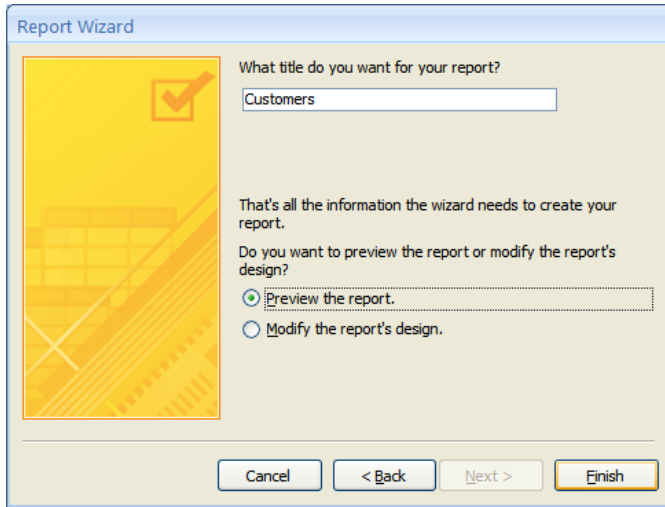
Chọn trường sắp xếp cho báo cáo. Nhấp Next để chuyển bước tiếp theo.



Chọn kiểu bố trí báo cáo trong khung Layout, và chọn khổ giấy ngang hay dọc ở khung Orientation. Nhấp Next để chuyển bước tiếp theo.

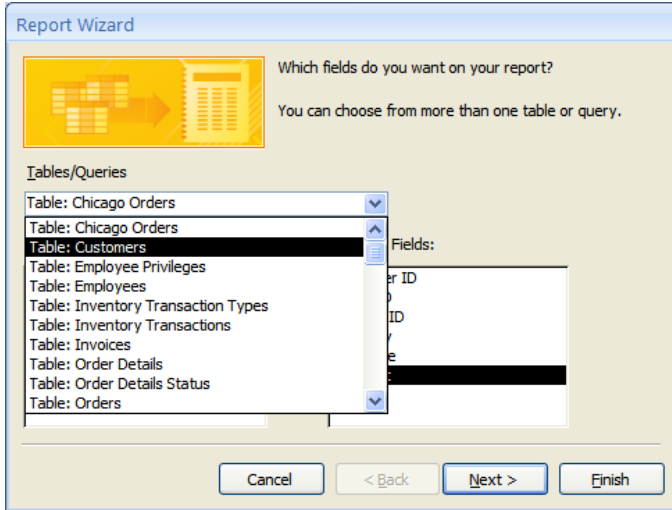


Chọn Style cho báo cáo, nhấp Next để chuyển bước tiếp theo.



Nhập vào tên của báo cáo trong phần What title do you... Cuối cùng chọn Finish để hoàn tất.

***Chú ý:*** Nếu bạn muốn lấy các trường vào báo cáo từ nhiều bảng hoặc Query, bạn không nhấp Next hoặc Finish sau khi đã lựa chọn xong trường từ Table hoặc Query đầu tiên trên trang đầu tiên của Report Wizards.



Thay vì chuyển sang bước tiếp theo bạn tiếp tục nhấp chọn các bảng hoặc Query khác cần lấy dữ liệu trong mục Tables/Queries.

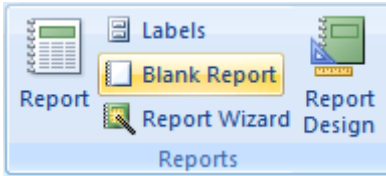
Sau khi đã chọn xong tất cả các trường cần thiết bạn có thể nhấp Next hoặc Finish để máy tính tiếp tục tạo báo cáo.

### Sử dụng công cụ Blank Report

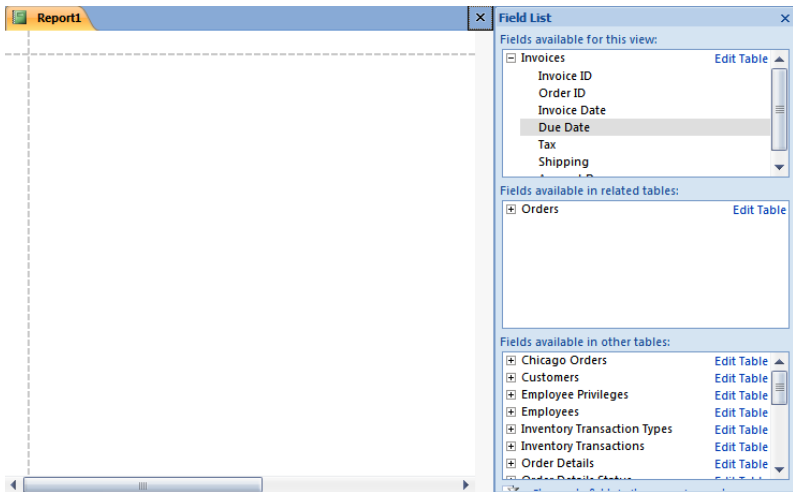
Nếu không quan tâm đến việc sử dụng công cụ Report hoặc Report Wizards, bạn có thể sử dụng công cụ Blank Report để xây dựng một báo cáo từ đầu. Với công cụ này bạn sẽ nhanh chóng tạo ra một báo cáo, đặc biệt thích hợp khi bạn dự định chỉ thêm vào đó một số trường nhất định. Dưới đây là các bước tạo báo cáo bằng công cụ Blank Report.

Bước 1: Trên Tab Create, trong nhóm Reports nhấp chọn Blank Report.

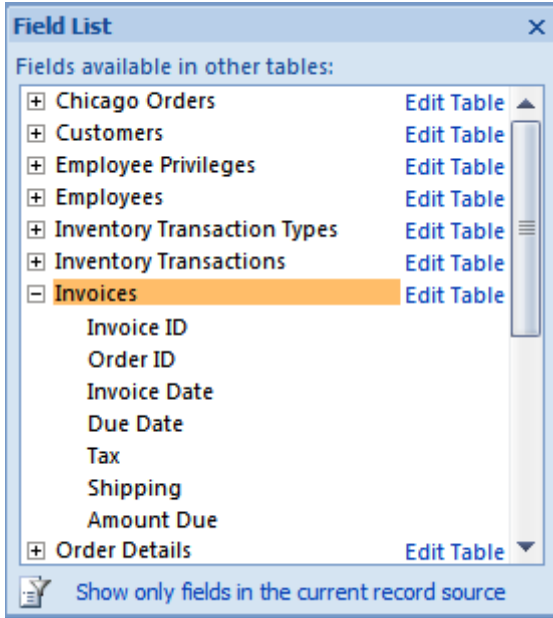




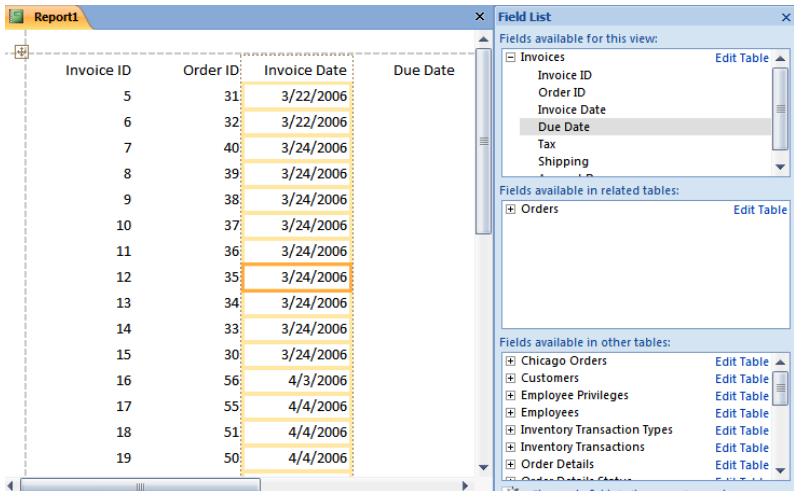
Một Blank Report được hiển thị ở chế độ Design, và Pane Field List xuất hiện bên trái của màn hình chính.



**Bước 2:** Trong Pane Field List nhấp chuột vào nút có hình (+) vào bảng chứa những trường mà bạn muốn hiển thị dữ liệu trên báo cáo.



**Bước 3:** Nhấp chọn một trường hoặc có thể giữ phím Ctrl rồi nhấp chọn nhiều trường một lúc rồi kéo thả lên báo cáo.



**Bước 4:** Sử dụng các công cụ trong nhóm Control trên Tab Formatting để thêm Logo, tiêu đề, số trang, ... cho báo cáo.



### **5.3. Tìm hiểu về Report section**

Trong Access, việc thiết kế các báo cáo được chia thành nhiều phần (Section). Bạn hãy hiển thị báo cáo ở chế độ Design View để nhìn thấy các vùng (section) đó. Muốn tạo các báo cáo đáp ứng được yêu cầu, cần phải hiểu rõ từng phần, và cách làm việc với các Section đó. Ví dụ, phần nào có thể lựa chọn để đặt các điều Control tính toán? Và làm thế nào để Access tính toán ra kết quả. Dưới đây là danh sách tóm tắt cả kiểu Section và cách sử dụng chúng.

- Report Header: Đây là phần được in ra chỉ một lần khi bắt đầu báo cáo. Sử dụng các tiêu đề báo cáo cho các thông tin mà thông thường có thể xuất hiện trên một trang bìa, như là một biểu tượng, một tiêu đề, hoặc một ngày tháng. Các Report Header được in trước khi in trang đầu tiên
- Page Header: Đây là Section sẽ được in trên đầu mỗi trang. Ví dụ, sử dụng Page Header để lặp lại tiêu đề báo cáo trên tất cả các trang.
- Group Header: Đây là phần được in khi bắt đầu mỗi nhóm bản ghi. Sử dụng Section này để in ra các tên nhóm. Ví dụ, trong một báo cáo được nhóm theo sản phẩm lúc đó Group Header được sử dụng để in tên các sản phẩm. Khi bạn đặt một Control tính toán mà sử dụng hàm Sum trong Group Header, Access sẽ tính tổng cho từng nhóm.

- Detail: Section này in tất cả các dòng có trong nguồn bản ghi. Đây là nơi bạn đặt các Control hiển thị dữ liệu và là phần thân chính của báo cáo.
- Group Footer: Đây là phần được in ở cuối mỗi nhóm bản ghi. Sử dụng Section này để in các thông tin tóm tắt cho một nhóm.
- Page Footer: Được in ở cuối mỗi trang. Sử dụng phần này để in thông tin về số trang cũng như trang hiện thời của báo cáo.
- Report Footer: Đây là phần chỉ được in duy nhất một lần. Sử dụng phần này để in tổng số hoặc các thông tin tóm tắt về toàn bộ báo cáo.

Chú ý: Ở chế độ Design, Report Footer xuất hiện dưới phần Page Footer. Tuy nhiên khi báo cáo được in hoặc xem trước khi in, thì Report Footer lại xuất hiện phía trên Page Footer, chỉ cần Goup Footer cuối cùng hoặc dòng cuối cùng của phần Detail nằm ở trang cuối.

#### **5.4. Tìm hiểu về các Control**

Tương tự như phần “Tìm hiểu về các Controls” – Bài 4

#### **5.5. Tinh chỉnh báo cáo**

Tương tự như phần “Tinh chỉnh Form” – Bài 4

#### **5.6. Thêm Control vào báo cáo**

Tương tự như phần “Thêm Control lên Form” – Bài 4

#### **5.7. Xem báo cáo**

Có nhiều cách để xem báo cáo, cách được lựa chọn phụ thuộc vào những gì mà bạn muốn thao tác với báo cáo và dữ liệu trên đó.

- Nếu bạn muốn tạm thời thực hiện các thay đổi với dữ liệu sẽ xuất hiện trên báo cáo trước khi bạn in nó, hoặc nếu bạn muốn sao chép dữ liệu từ các báo cáo vào bộ nhớ tạm, thì hãy sử dụng Report View.
- Nếu bạn muốn có thể thay đổi các thiết kế ngay khi báo cáo đang hiển thị dữ liệu bạn hãy sử dụng chế độ Layout View.
- Nếu bạn chỉ đơn giản muốn xem những gì báo cáo sẽ được thấy khi in ra, bạn hãy sử dụng chế độ Print Preview.

### **Xem báo cáo ở chế độ Report View**

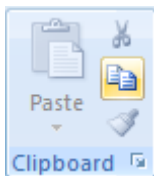
Report View là chế độ hiển thị mặc định được sử dụng khi bạn nhấp đúp chuột vào một báo cáo trên pane Navigation nếu như nó chưa được mở.

Trong trường hợp báo cáo đang mở ở một chế độ nào đó bạn nhấp phải chuột vào tên báo trên Pane Navigation chọn Report View.

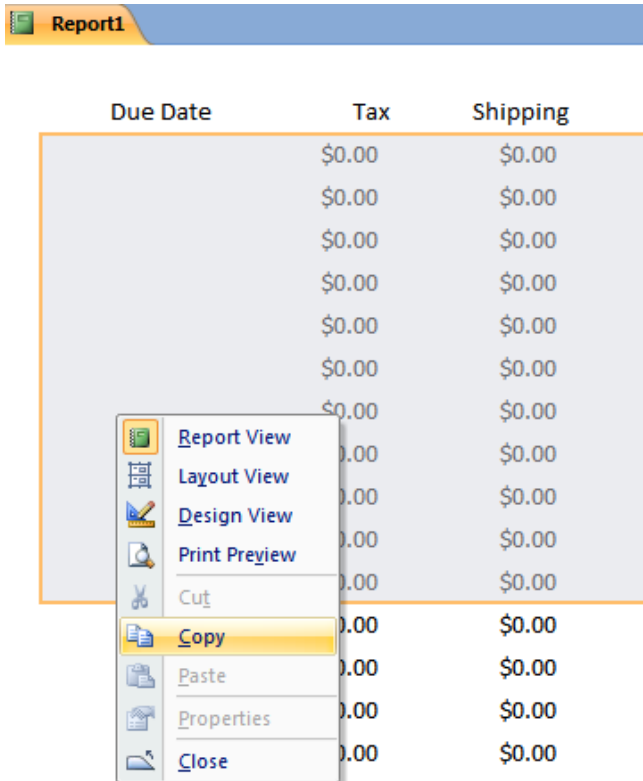
### **Làm việc với dữ liệu ở chế độ Report View**

Trong chế độ Report View, bạn có thể lựa chọn nội dung văn bản trên đó và Copy vào bộ nhớ tạm của máy tính. Để lựa chọn toàn bộ các cột của một số hàng, bạn nhấp giữ chuột vào lề bên trái rồi kéo tới hàng muốn chọn. Sau đó có thể copy những hàng đã chọn đó bằng một trong các cách dưới đây:

- Trên Tab Home, trong nhóm Clipboard, nhấp chọn Copy



- Nhấp phải chuột và dòng được chọn, rồi chọn Copy.

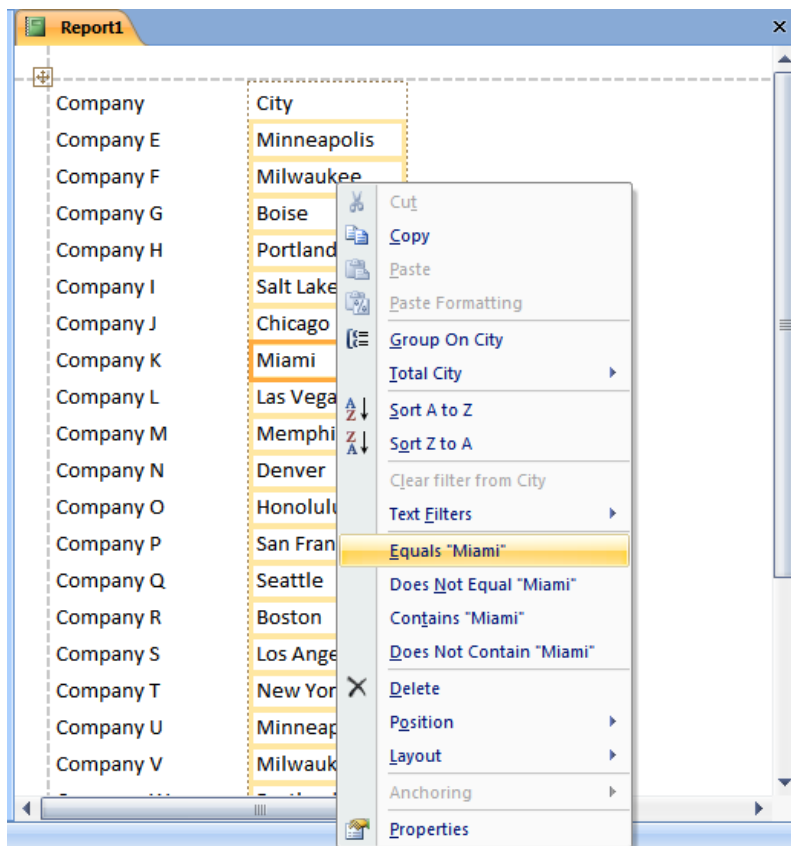


- Sử dụng tổ hợp phím Ctrl + C.

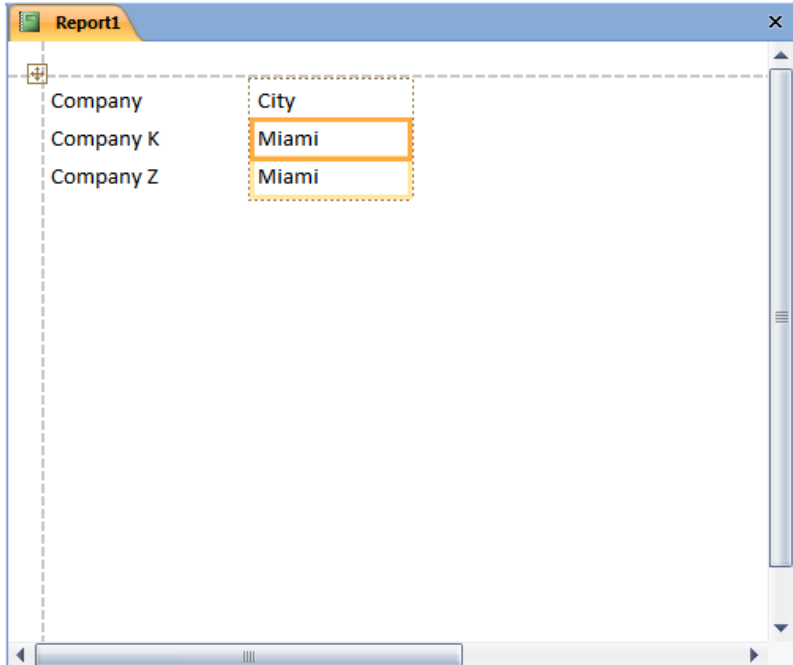
### **Chỉ hiển thị những trường bạn muốn bằng cách sử dụng công cụ lọc**

Bạn có thể áp dụng các bộ lọc trực tiếp cho báo cáo mà không cần rời khỏi chế độ Report View. Ví dụ, trên báo cáo có một cột "City" và bạn chỉ muốn xem những hàng mà cột "City" có giá trị là "Miami", bạn làm như sau:

- Tìm từ "Miami" trên báo cáo và nhấp phải chuột vào nó.
- Nhấp chọn Equals "Miami".



Access sẽ tạo và áp dụng bộ lọc, lúc này báo cáo chỉ còn những bản ghi có cột City là Miami.



### **Tạo một bộ lọc chi tiết hơn.**

Bước 1: Nhấp phải chuột vào trường cần lọc.

Bước 2: Chọn Common Filters.

Bước 3: Nhấp chọn điều kiện lọc mà bạn muốn.

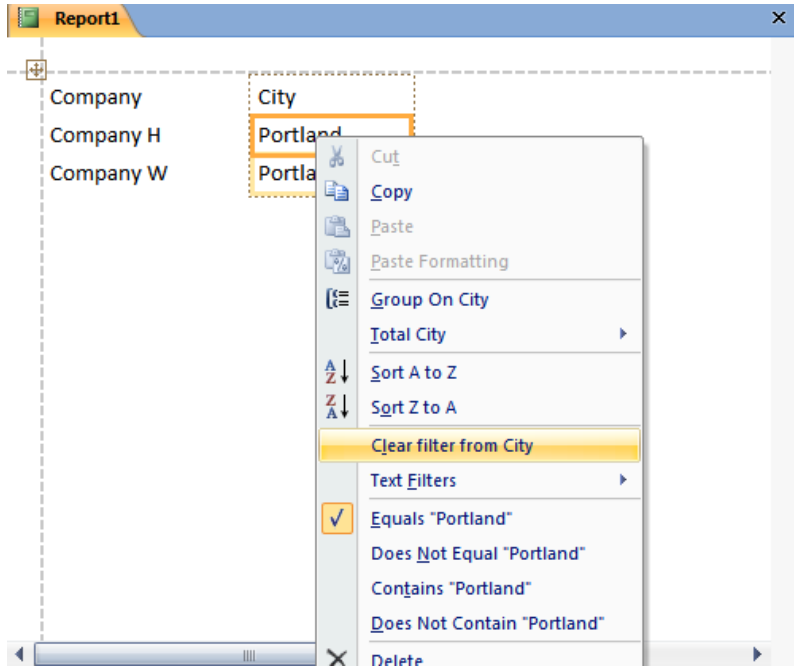
Bước 4: Nhập vào giá trị lọc.

### **Hủy bỏ một bộ lọc**

Bước 1: Nhấp phải chuột vào trường mà bạn muốn loại bỏ bộ lọc.

Bước 2: Nhấp chọn Clear filter from city.



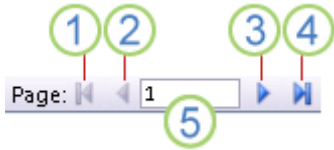


**Chú ý:** Nếu bạn áp dụng một bộ lọc cho một báo cáo sau đó lưu và đóng báo cáo lại khi đó bộ lọc sẽ được lưu cùng báo cáo. Tuy nhiên lần sau khi mở lại báo cáo, Access sẽ không sử dụng được bộ lọc. Để sử dụng lại bộ lọc, trên Tab Home tìm tới nhóm Sort & Filter, tiếp theo nhấp chọn Toggle Filter.

### **Xem báo cáo trước khi in sử dụng Print Preview.**

Nhấp phải chuột vào báo cáo trong Pane Navigation, sau đó chọn Print Preview.

Các nút trên Navigation có thể giúp bạn xem một cách tuần tự hoặc bất cứ trang nào mà bạn muốn.

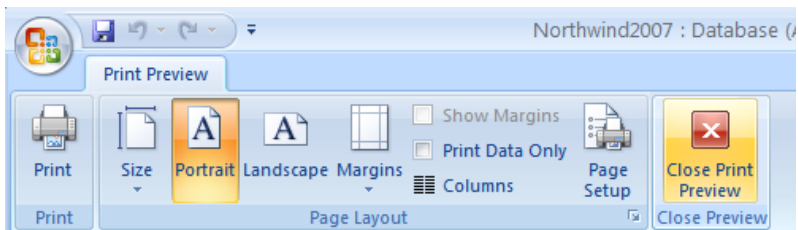


- (1). Trở về trang đầu tiên
- (2): Trở về trang trước
- (3): Chuyển tới trang tiếp theo
- (4): Chuyển tới trang cuối cùng
- (5): Gõ vào số trang trong hộp nhập và nhấn Enter để nhảy tới trang đó.

Ở chế độ Print Preview, bạn có thể phóng to để xem chi tiết hoặc thu nhỏ để xem vị trí của dữ liệu trên trang. Nếu vị trí con trỏ chuột nằm trong báo cáo bạn chỉ cần nhấp chuột một lần. Để đảo ngược chế độ hiển thị bạn bấm chuột một lần nữa. Bạn cũng có thể sử dụng Control Zoom trên thanh trạng thái của Access để tùy chỉnh cỡ phóng to hay thu nhỏ báo cáo.

Để đóng chế độ Print Preview bạn làm theo một trong các cách sau đây:

- Trên Tab Print Preview chọn nút Close Print Preview.







- Nhấp phải chuột vào báo cáo trên Pane Navigation sau đó chọn Layout View hoặc chọn Design View.

## 5.8. In báo cáo

Bạn có thể in một báo cáo trong khi nó đang được mở ở bất kỳ chế độ xem nào, hoặc thậm chí trong khi nó đang đóng. Trước khi in, hãy chắc chắn là đã thiết lập các thông số cài đặt cho trang in, như lề, khổ giấy trang in, ... Access lưu lại những cài đặt đó vì vậy chỉ phải thiết lập chúng một lần duy nhất. Sau này bạn cũng có thể thiết lập lại nếu yêu cầu in ấn thay đổi.

### **Thay đổi định dạng trang in.**



- Mở một báo cáo ở chế độ Print Preview. Bạn có thể thay đổi các thiết lập cho trang in ở bất kỳ chế độ nào, nhưng ở chế độ hiển thị Print Preview là trực quan nhất bởi khi đó có thể nhìn thấy ngay lập tức những thay đổi mà bạn vừa thực hiện.

- Trên Tab Print Preview tìm tới nhóm Page Layout, nhấp chọn Portrait  hoặc Landscape  để thiết lập các trạng thái định hướng. Size  để thiết lập kích thước cho trang in, Margins  để điều chỉnh lề, ...

- Sau khi thực hiện một thay đổi, hãy sử dụng các nút trên Navigation để xem thử một số trang để chắc chắn rằng bạn đã không tạo ra bất kỳ vấn đề gì về định dạng trên các trang phía sau.

### **In báo cáo**

Bước 1: Mở Report ở bất kỳ chế độ nào mà bạn muốn, hoặc có thể chọn báo cáo trên Navigation Pane.

Bước 2: Nhấp chọn nút Microsoft Office Button , sau đó nhấp Print . Hộp thoại Print xuất hiện

Bước 3: Nhập các tùy chọn in báo cáo trong hộp thoại, cuối cùng nhấp Ok để in báo cáo



Mở đầu

Table

Query

Form

Report

Macro

**Giáo viên: Tạ Hùng Thiện, Kỹ thuật viên tin học**  
**TT Bách khoa ASIAN**

**ĐT: 0905288222. Email: thienth@gmail.com**

Nội dung

## **Microsoft Access 2003**

① Mở đầu: Giới thiệu về Microsoft Access

② Bảng (Table)

③ Truy vấn dữ liệu (Query)

④ Làm việc với mẫu biểu (Form) - Command Button

⑤ Báo biểu (Report)

⑥ Macro - Menu

# Chương I: Giới thiệu về Microsoft Access 2003

## I. Giới thiệu

- Access là một hệ quản trị cơ sở dữ liệu mạnh chạy trên môi trường Windows
- Access 2003 là một phần trong bộ chương trình Microsoft Office 2003 được dùng rất rộng rãi hiện nay
- Access cho phép người sử dụng quản lý, bảo trì, khai thác, lưu trữ số liệu một cách có tổ chức trên máy tính
- Với Access, người sử dụng không cần viết từng câu lệnh như Pascal, Foxpro, C... mà chỉ tổ chức dữ liệu và thiết kế các yêu cầu, công việc cần giải quyết



## II. Các khái niệm cơ bản

➤ Cơ sở dữ liệu (Database): là tập hợp các dữ liệu phục vụ cho một mục đích, bài toán cụ thể. Ví dụ như CSDL cho bài toán quản lý lương, bài toán Quản lý sinh viên...

➤ Cơ sở dữ liệu quan hệ (Relational Database): Dữ liệu được tổ chức thành các bảng hai chiều.

\* Chiều dọc: là tập hợp các **đặc điểm** của **đối tượng** cần quản lý (gọi là trường- Field)

Vd: **Sinh viên** (**tên, ngày sinh, lớp, điểm**)

**Mặt hàng** (**tên hàng, số lượng, đơn giá,...**)

\* Chiều ngang: là giá trị của các trường (các bản ghi - Record)

Vd: (**Nguyễn Văn An, 12/5/1970, K98T, 8**)

## II. Các khái niệm cơ bản

- Đối với Access, cơ sở dữ liệu (CSDL) bao gồm:
- **Table** (bảng): là thành phần cơ bản của CSDL, nó cho phép lưu trữ dữ liệu phục vụ công tác quản lý. Các bảng trong một CSDL thường có mối quan hệ với nhau
  - **Query** (truy vấn): là công cụ dùng để tính toán các trường không cần lưu trữ (điểm TB, thành tiền), sắp xếp, tìm kiếm, tổng hợp dữ liệu



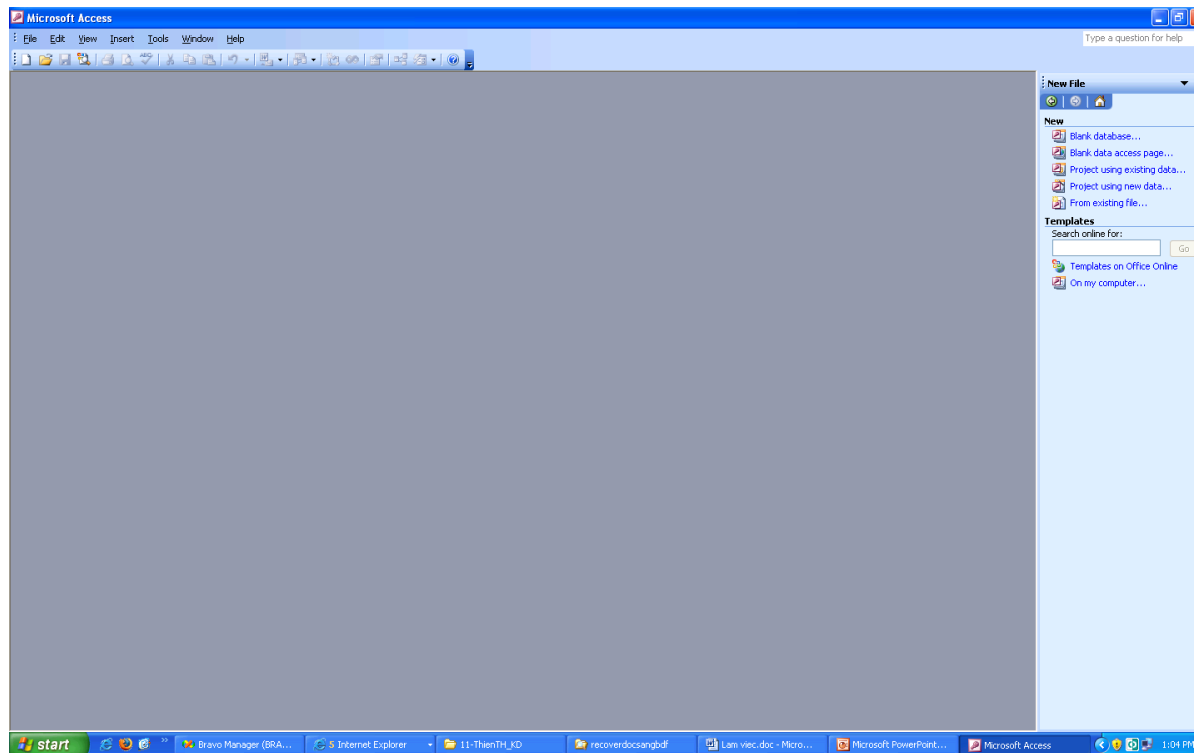
## II. Các khái niệm cơ bản

- Form (Biểu mẫu): cho xây dựng các biểu mẫu nhập số liệu như trong thực tế, giúp NSD thực hiện việc nhập, xuất phong phú, không đơn điệu như nhập xuất trên Table hay Query
- Report (Báo biểu): cho in ấn với các khả năng
  - In dữ liệu dưới dạng bảng biểu
  - Sắp xếp dữ liệu trước khi in
  - In dữ liệu có quan hệ trên một báo cáo
- Macro (tập lệnh): là một tập hợp các lệnh nhằm thực hiện các thao tác thường gặp. Khi gọi một Macro, Access sẽ cho thực hiện một dãy các lệnh tương ứng đã qui định
- Module (đơn thể): là một dạng tự động hoá chuyên sâu hơn Macro. Đó là những hàm riêng của NSD được viết bằng ngôn ngữ Access Basic

➤ Tập tin chương trình do Access tạo ra có đuôi là .MDB

### III. Khởi động, thoát

➤ Khởi động: Chọn Start / Program / Microsoft Access hoặc chọn biểu tượng Microsoft Access (hình chìa khoá) trên màn hình nền hay trên thanh công cụ (nếu có), xuất hiện màn hình sau:

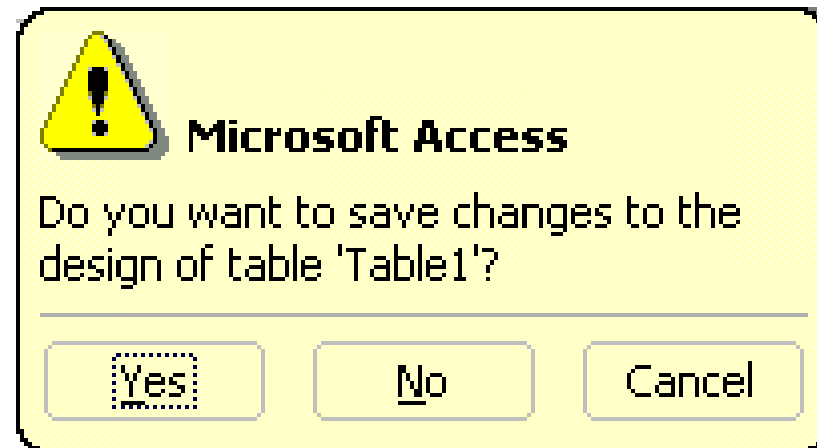


### III. Khởi động, thoát

- Create a new database using
  - Blank Access database: tạo một CSDL mới
  - Access database wizards, pages, and projects: tạo một CSDL mới với các dạng có sẵn của Access
  - Open an Existing Database: mở một CSDL đã có
- Sau khi chọn một trong ba mục trên, chọn OK
- Nếu không thực hiện các chọn lựa trên thì chọn Cancel để vào thẳng màn hình Access

➤ Thoát: chọn lệnh File / Exit (Alt+F4)

Nếu chưa lưu dữ liệu trước khi thoát sẽ có thông báo



## Chương II : Bảng (Table)

Gồm các thao tác:

✓ Tạo bảng mới, nhập dữ liệu vào bảng

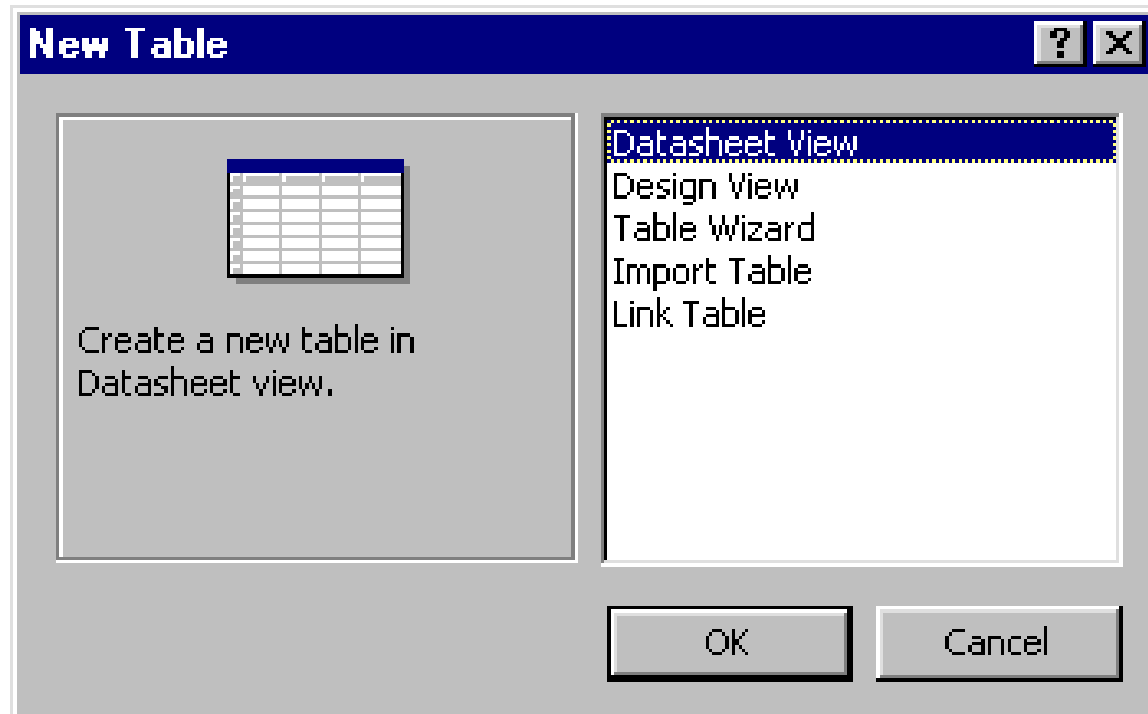
✓ Sử dụng thuộc tính của trường để trình bày dạng dữ liệu , kiểm tra tính hợp lệ của trường khi cập nhật

✓ Thiết lập quan hệ giữa các bảng

## I. Tạo cấu trúc của bảng

- Khởi động Access, trong hộp thoại MS Access, chọn Blank Access Database, OK
- Xuất hiện hộp thoại File New Database
- Gõ tên CSDL, chọn Create, xuất hiện cửa sổ Database

- Từ cửa sổ Database, chọn Table /New, xuất hiện hộp thoại New Table cho phép chọn cách tạo Table:



# I. Tạo cấu trúc của bảng

## 1. Tạo bảng bằng Table Wizard

Cho tạo bảng theo các mẫu có sẵn của Access

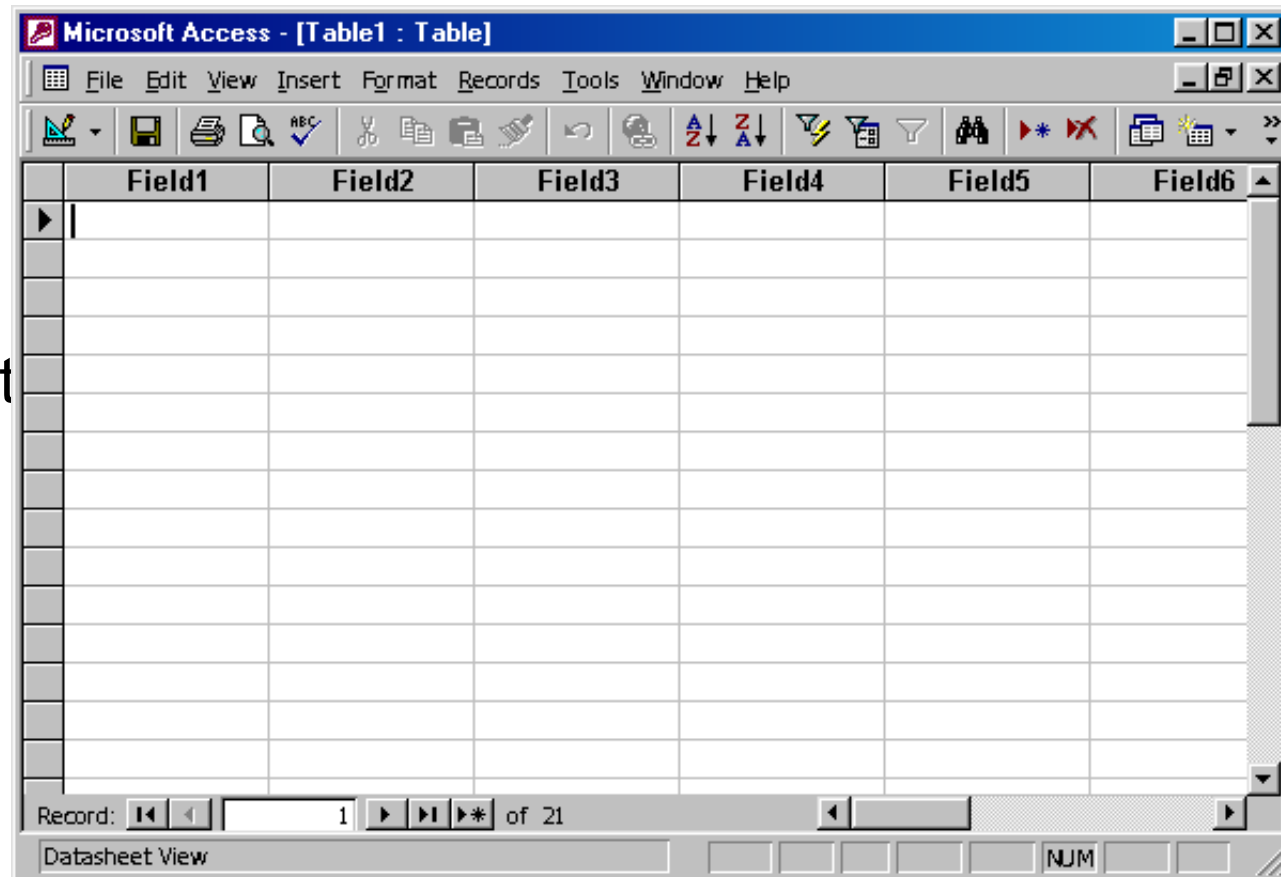
- Bước 1: trong hộp thoại New Table, chọn Table Wizard, OK.
- Bước 2: chọn tên Table, tên trường theo mẫu có sẵn của Access và sửa đổi lại theo yêu cầu của thực tế. Chọn Next để thực hiện các công việc kế tiếp như sửa tên Table,... sau cùng là chọn Finish

# I. Tạo cấu trúc của bảng

## 2. Tạo bảng bằng Datasheet View

Cho tạo bảng theo cách sử dụng một mẫu cho trước, Access dựa vào đó để tạo các Table

- Bước 1:  
trong hộp thoại  
New Table,  
chọn Datasheet  
View, OK. Xuất  
hiện cửa sổ  
MS Access



# I. Tạo cấu trúc của bảng

## 2. Tạo bảng bằng Datasheet View

- Bước 2: Nhập nội dung vào bảng
- Vd: để tạo bảng lưu trữ điểm sinh viên
  - Thay đổi tên trường: đứng trong trường, chọn Format / Rename Column hoặc nhấp phải chuột ở tên Field, chọn Rename Column
  - Nhập dữ liệu vào bảng như trong Excel



Mã SV	Họ lót	Tên	Lớp	Điểm KQ	Xếp loại	Field7
TC001	Trần Ngọc	Châu	Trung cấp	19	Trung bình	
KT002	Lê	Vũ	Kỹ thuật viên	25	Giỏi	
TC002	Phan Thu	Phong	Trung cấp	28	Xuất sắc	



# I. Tạo cấu trúc của bảng

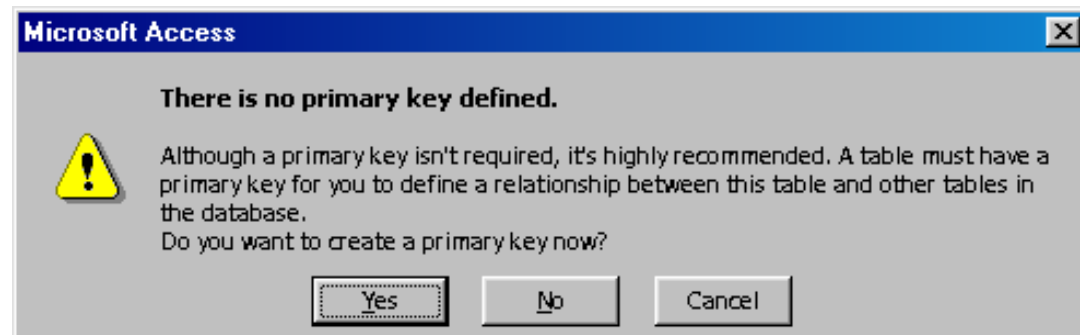
## 2. Tạo bảng bằng Datasheet View

➤ Bước 3: Lưu bảng. Chọn File / Save, xuất hiện hộp thoại

- Table Name: gõ tên bảng, OK



- Access sẽ hỏi có đặt khoá không, bạn chọn No



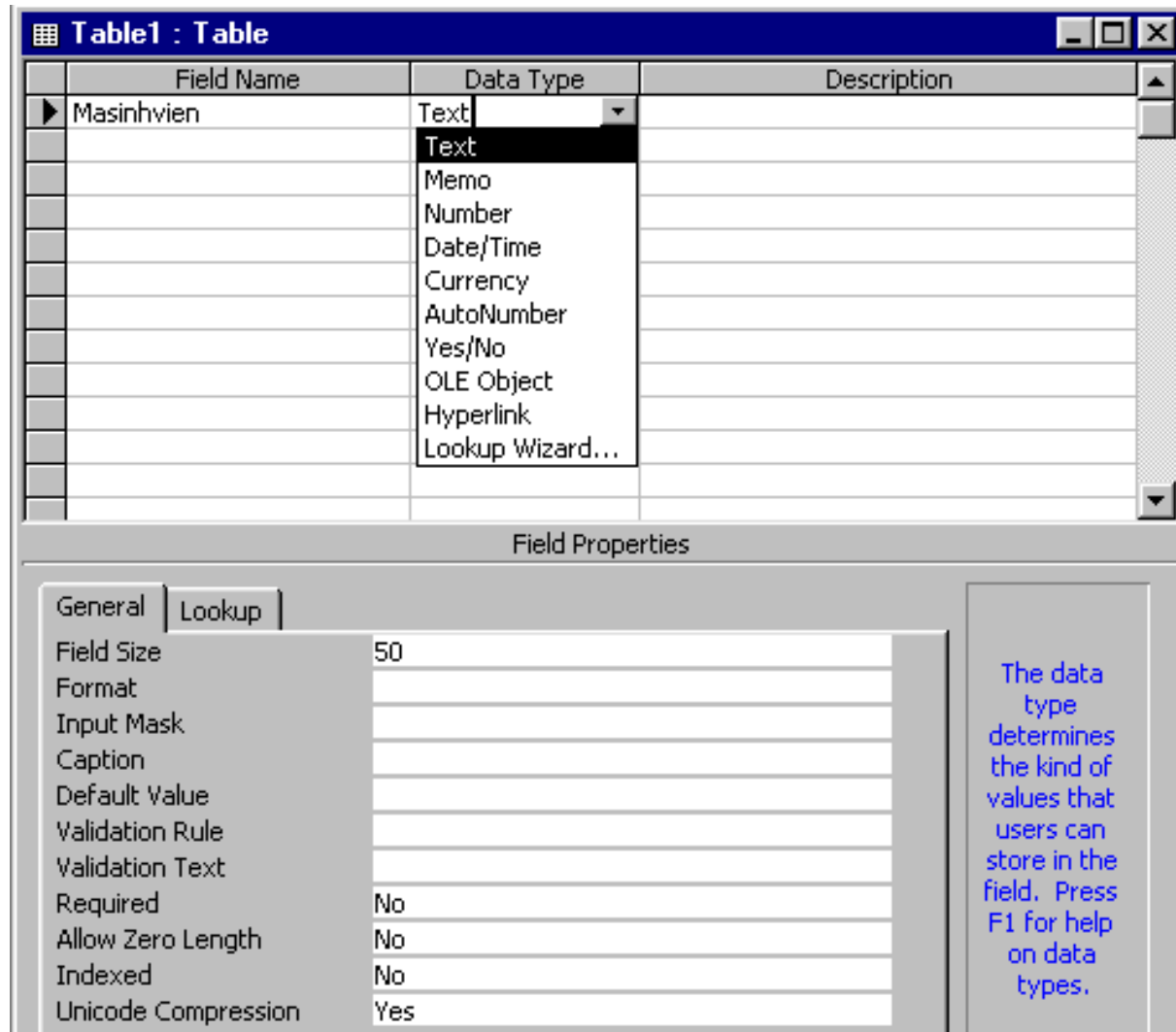
➤ Bước 4: Đóng bảng, chọn File / Close.

# I. Tạo cấu trúc của bảng

## 3. Tạo bảng bằng Design View

Cho tạo bảng theo  
nhu cầu của NSD

- Trong hộp thoại New Table, chọn Design View, OK. Cửa sổ định nghĩa cấu trúc bảng có các phần



# I. Tạo cấu trúc của bảng

## 3. Tạo bảng bằng Design View

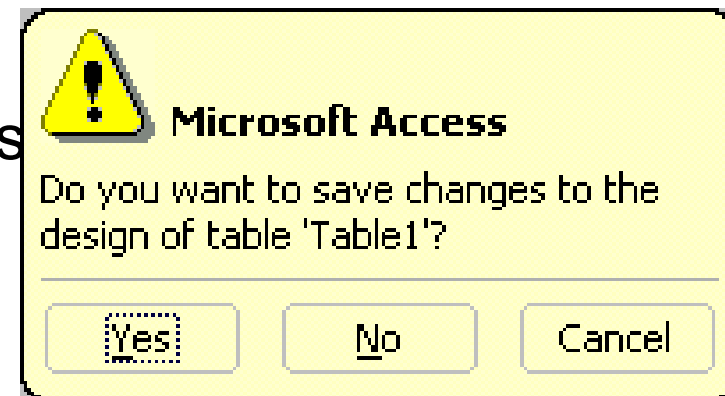
- Bước 1: Khai báo tên trường (Field name): bắt buộc, đặt tùy ý ( $\leq 64$  ký tự), có thể có khoảng trống
- Bước 2: Khai báo kiểu dữ liệu (Data type) tương ứng của trường: bắt buộc. Chọn một trong các kiểu

<i>Kiểu</i>	<i>Ý nghĩa</i>
Text	Chứa tập hợp các ký tự tùy ý, $\leq 255$ ký tự
Memo	Chứa tập hợp các ký tự tùy ý, $\leq 65535$ ký tự
Number	Chứa giá trị số
Date/Time	Chứa giá trị ngày, giờ
Currency	Chứa giá trị tiền tệ
Auto Number	Giá trị số (Access tự tăng 1 đơn vị cho mỗi bản ghi mới)
Yes/No	Chứa giá trị logic (True/False)
Lookup Wizard	Chọn một trị từ danh sách các trị

# I. Tạo cấu trúc của bảng

## 3. Tạo bảng bằng Design View

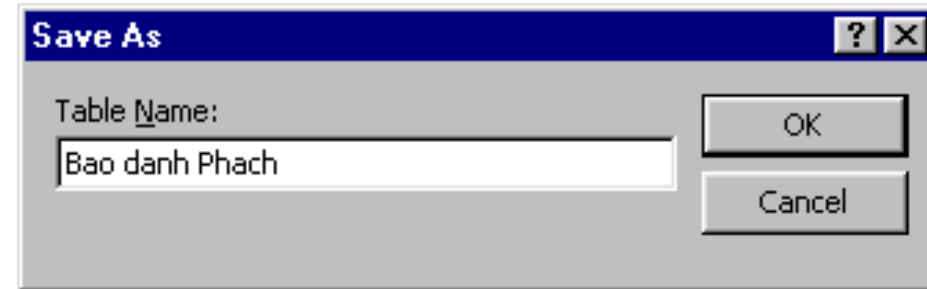
- Bước 3: Field properties: tương ứng với mỗi kiểu dữ liệu, khai báo thêm các thuộc tính, gồm thuộc tính chung (General) và thuộc tính nhập số liệu (Lookup)
- Bước 4: Sau khi thiết kế xong cấu trúc bảng, ghi lên đĩa bằng một trong các cách
  - Chọn lệnh File / Save (Ctrl+S hoặc biểu tượng Save)
  - Hoặc đóng cửa sổ Table, Access sẽ hỏi có lưu không, chọn Yes để lưu.



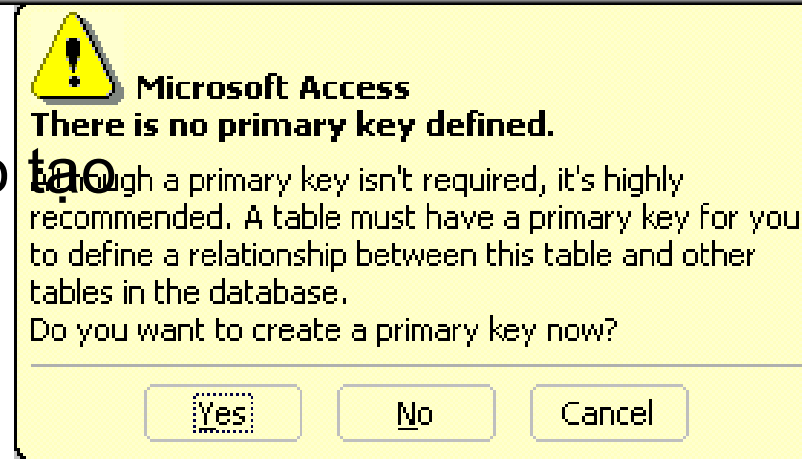
# I. Tạo cấu trúc của bảng

## 3. Tạo bảng bằng Design View

- Gõ tên bảng vào mục  
Table Name



- Nếu chưa tạo khoá chính (Primary Key), Access sẽ hỏi có tạo không



- No: cấu trúc bảng được ghi như thiết kế
  - Yes: Access tự thêm trường có tên ID làm khoá chính
- Bạn nên chọn No

Mở đầu

Table

Query

Form

Report

Macro

### 3. Tạo bảng bằng Design View

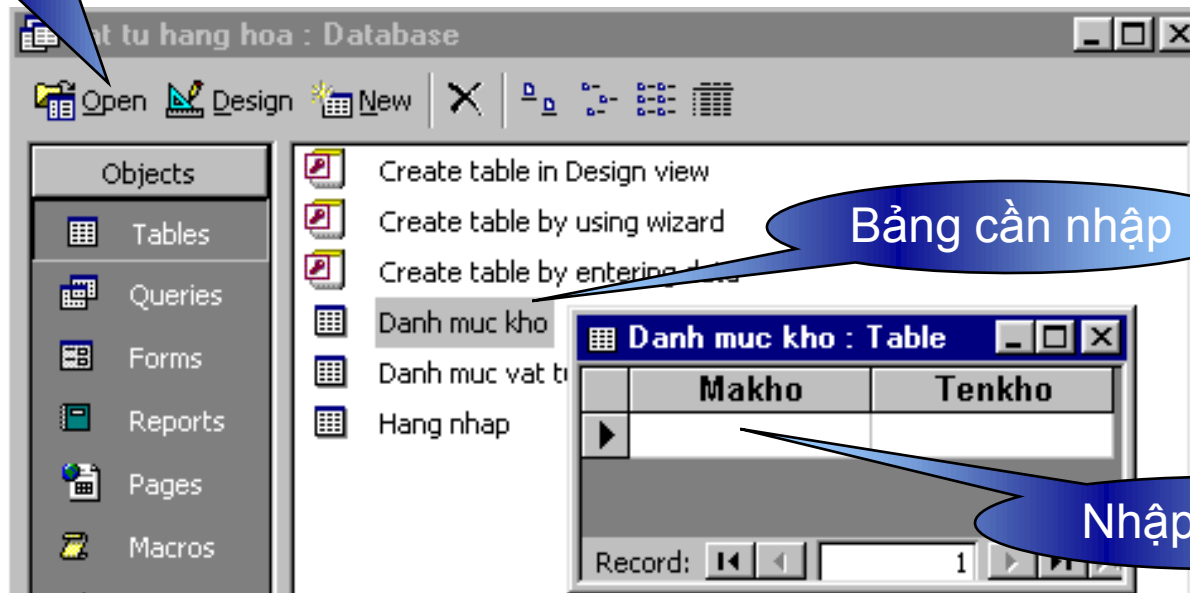
#### ➤ Bước 5: nhập dữ liệu cho bảng

▪ Trong cửa sổ Database, chọn trang Table

▪ Chọn tên bảng cần nhập, chọn Open



Chọn Open



## II. Giới thiệu các thuộc tính của trường

### 1. Công dụng:

- Điều khiển hình thức thể hiện dữ liệu
- Nhập dữ liệu theo mẫu
- Kiểm tra dữ liệu, ngăn cản việc nhập sai
- Tăng tốc độ tìm kiếm

## II. Giới thiệu các thuộc tính của trường

### 2. Tổng quan về các thuộc tính của trường

<i>Thuộc tính</i>	<i>Ý nghĩa</i>
Field Size	Số ký tự của trường Text, hay độ dài của trường Number
Format	Dạng hiển thị dữ liệu kiểu ngày và số
Decimal Place	Số chữ số thập phân trong kiểu Number và Currency
Input Mask	Qui định mặt nạ nhập liệu
Caption	Đặt tiêu đề cho trường (sẽ hiển thị khi nhập liệu thay vì tên trường)
Default Value	Xác định giá trị mặc định của trường
Validation Rule	Qui tắc dữ liệu hợp lệ, dữ liệu phải thoả mãn qui tắc này mới được nhập
Validation Text	Thông báo xuất hiện khi NSD vào dữ liệu vượt quá giới hạn qui định ở mục Validation Rule
Require	Nếu chọn Yes, bắt buộc phải vào dữ liệu cho trường
Allow Zero Length	Nếu chọn Yes cho phép vào chuỗi kí tự rỗng(Text, Memo)
Indexed	Tạo chỉ mục để tăng tốc độ tìm kiếm trên trường này <sup>20</sup>



## II. Giới thiệu các thuộc tính của trường

### 3. Chi tiết về các thuộc tính của trường

#### ➤ Thuộc tính *Field Size*

- Trường Number
- Có thể chọn các kiểu sau

<i>Field Size</i>	<i>Miền giá trị</i>
Integer	-32768 ██████████ 2767
Long Integer	-2147483648 ██████████ 147483648
Single	$-3.4 \cdot 10^{38}$ ██████████ $4 \cdot 10^{38}$
Double	$-1.797 \cdot 10^{308}$ ██████████ $797 \cdot 10^{308}$

#### ➤ Thuộc tính *Format*

- Trường Text

<i>Ký tự</i>	<i>Công dụng</i>
@	Bắt buộc phải nhập kí tự vào văn bản
&	Không bắt buộc phải nhập kí tự vào văn bản
<	Biến chữ hoa thành chữ thường
>	Biến chữ thường thành chữ hoa

## II. Giới thiệu các thuộc tính của trường

### 3. Chi tiết về các thuộc tính của trường

#### ➤ Thuộc tính Format

##### ▪ Trường Number

<i>Dạng</i>	<i>Số</i>	<i>Trình bày</i>
General Number	1234.5	1234.5
Currency	1234.5	\$1234.5
Fixed	1234.5	1234.50
Standard	1234.5	1,234.50
Percent	0.825	82.5%

<i>Dạng</i>	<i>Hiển thị</i>
General Date	4/23/02 10:30:00 AM
Long Date	Tuesday, April 23, 2002
Medium Date	23/April/2002
Short Date	4/23/02
Long Time	10:30:00 AM
Medium Time	10:30 AM
Short Time	10:30

##### ▪ Trường Date / Time: thường chọn kiểu Short Date

## II. Giới thiệu các thuộc tính của trường

### 3. Chi tiết về các thuộc tính của trường

#### ➤ *Thuộc tính Input Mask (mặt nạ nhập liệu)*

##### ▪ Công dụng

- Tạo khuôn dạng nhập liệu cho dễ nhìn. Vd: dấu ngăn cách phần nguyên và thập phân, dấu gạch ngang để phân cách các cụm ký tự của tài khoản
- Kiểm tra tính hợp lệ của ký tự gõ vào. (Vd nếu qui định là chữ số thì Access sẽ không nhận các ký tự không phải là chữ số)
- Tự động biến đổi các ký tự được nhập (Vd từ chữ thường sang chữ hoa)
- Che giấu thông tin đưa vào (Vd khi ta dùng Password)

## II. Giới thiệu các thuộc tính của trường

### 3. Chi tiết về các thuộc tính của trường

#### ➤ Thuộc tính *Input Mask* (mặt nạ nhập liệu)

- Vd1: để nhập số điện thoại dạng Text theo mẫu:

0-511-810425

ta dùng mặt nạ

0-000-000000

- Vd2: để nhập điểm thi chứa hai chữ số phần nguyên, một chữ số phần thập phân ta dùng mặt nạ  
00.0

Ký tự	Công dụng : là vị trí dành cho
0	chữ số 0...9, bắt buộc
9	chữ số 0...9, không bắt buộc
#	chữ số, dấu cộng +, dấu trừ -, dấu cách
L	một chữ cái, bắt buộc
?	chữ hoặc dấu cách, không bắt buộc
A	ký tự chữ hay số, bắt buộc
a	ký tự chữ hay số, không bắt buộc
&	ký tự bất kỳ, bắt buộc
C	ký tự bất kỳ, không bắt buộc

## II. Giới thiệu các thuộc tính của trường

### 3. Chi tiết về các thuộc tính của trường

#### ➤ ***Thuộc tính Default Value***

- Dùng để đặt giá trị mặc định cho trường. Giá trị mặc định này có thể là một hằng hay một hàm của Access.
  - Ví dụ: Date(): ngày hiện hành

#### ➤ ***Thuộc tính Required***

- Nếu chọn Yes, bắt buộc phải nhập dữ liệu vào, nếu không máy sẽ báo lỗi

#### ➤ ***Thuộc tính Validation Rule***

Công dụng: Access tự động kiểm tra tính hợp lệ khi nhập liệu. Muốn chương trình kiểm tra, trong biểu thức này, ta đặt một biểu thức hợp lệ. Ví dụ: >0 (Số lượng phải >0)

## II. Giới thiệu các thuộc tính của trường

### 3. Chi tiết về các thuộc tính của trường

#### ➤ *Thuộc tính Allow Zero Length*

Công dụng: Nếu chọn Yes, sẽ cho phép các trường Text và Memo nhận các chuỗi rỗng

#### ➤ *Thuộc tính Indexed*

- Công dụng: sắp xếp một trường nào đó. Thuộc tính nhận các giá trị sau:

<i>Giá trị</i>	<i>Ý nghĩa</i>
No	Không có chỉ mục
Yes (Duplicate Ok)	Tạo chỉ mục cho phép các giá trị trong trường này trùng lặp
Yes (No Duplicate)	Tạo chỉ mục không cho phép các giá trị trong trường này trùng lặp

### III. Tạo Combo box cho trường (Lookup Wizard)

#### 1. Khái niệm

➤ Là cách tạo trường mà giá trị nhập vào được chọn từ danh sách các trị. Danh sách này được liệt kê trong danh sách đổ xuống (Combo box)

#### 2. Cách tạo: Ví dụ minh họa

➤ Yêu cầu

- Giả sử có Table **Danh mục ngành thi** có cấu trúc và nội dung:
- Ta cần tạo Table **Danh sách thi sinh** có cấu trúc như sau:

	Manganh	Tennganh
	01	Toán học
	02	Tin học
	03	Vật lý
	04	Hoá học
	05	Sinh học
	06	Tài chính
	07	QT Kinh doanh
	08	Luật học
	09	Xã hội học
	10	Văn học
	11	Lịch sử
	12	Địa lý
	*	

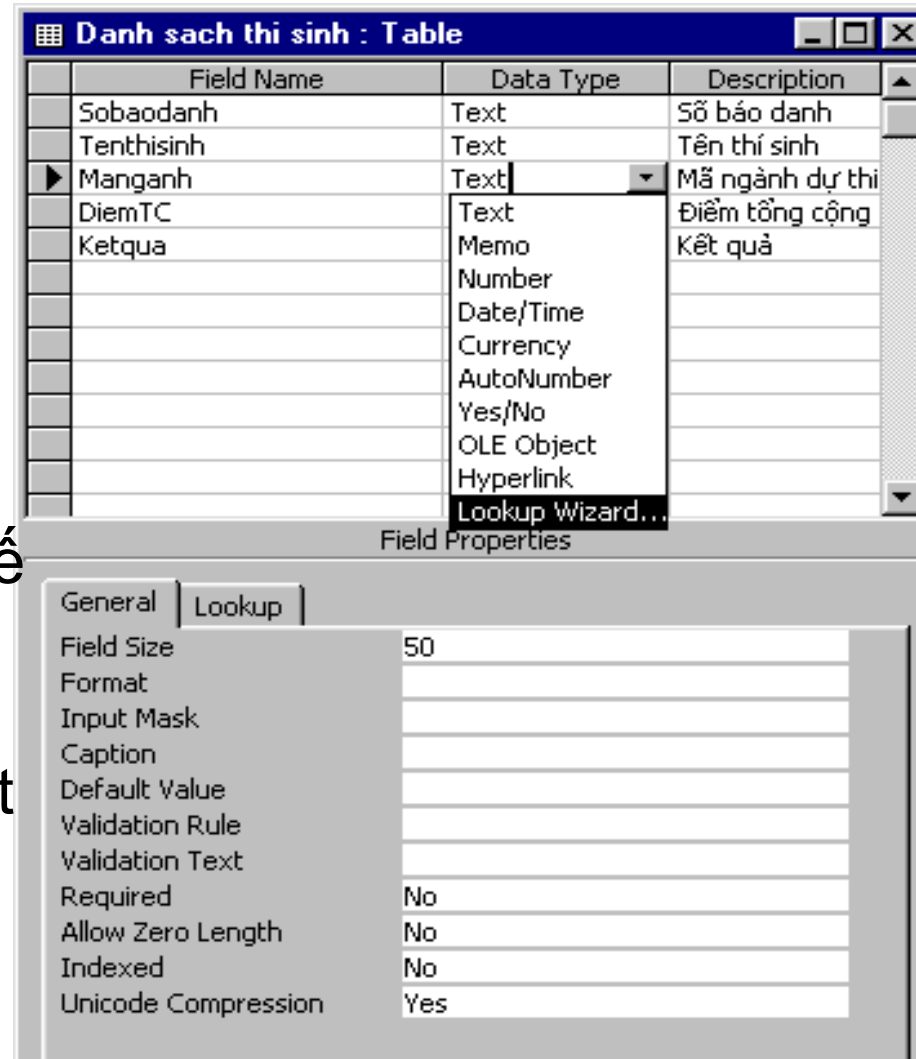
### III. Tạo Combo box cho trường (Lookup Wizard)

#### 2. Cách tạo: Ví dụ minh họa

- Trường Manganh khi nhập liệu, các giá trị được chọn từ danh sách các trị

#### ➤ Cách thực hiện

- Bước 1: Tại cửa sổ thiết kế Table, khi tạo trường Manganh, ở cột Data Type, chọn Lookup Wizard..., xuất hiện hộp thoại Lookup Wizard:



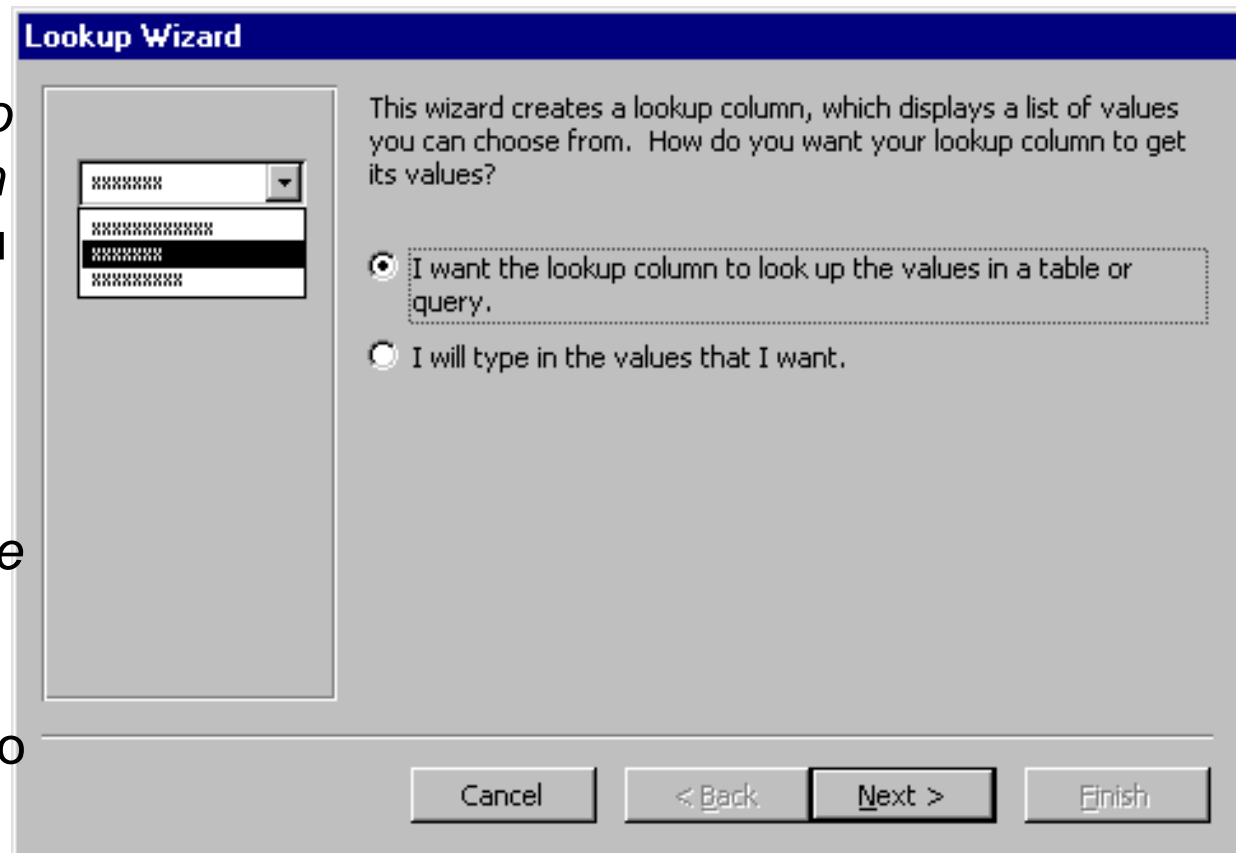


### III. Tạo Combo box cho trường (Lookup Wizard)

#### 2. Cách tạo: Ví dụ minh họa

➤ (1) Chọn *I want the lookup column to look up the values in a table or query*: nếu muốn giá trị trong danh sách được lấy từ table hay query

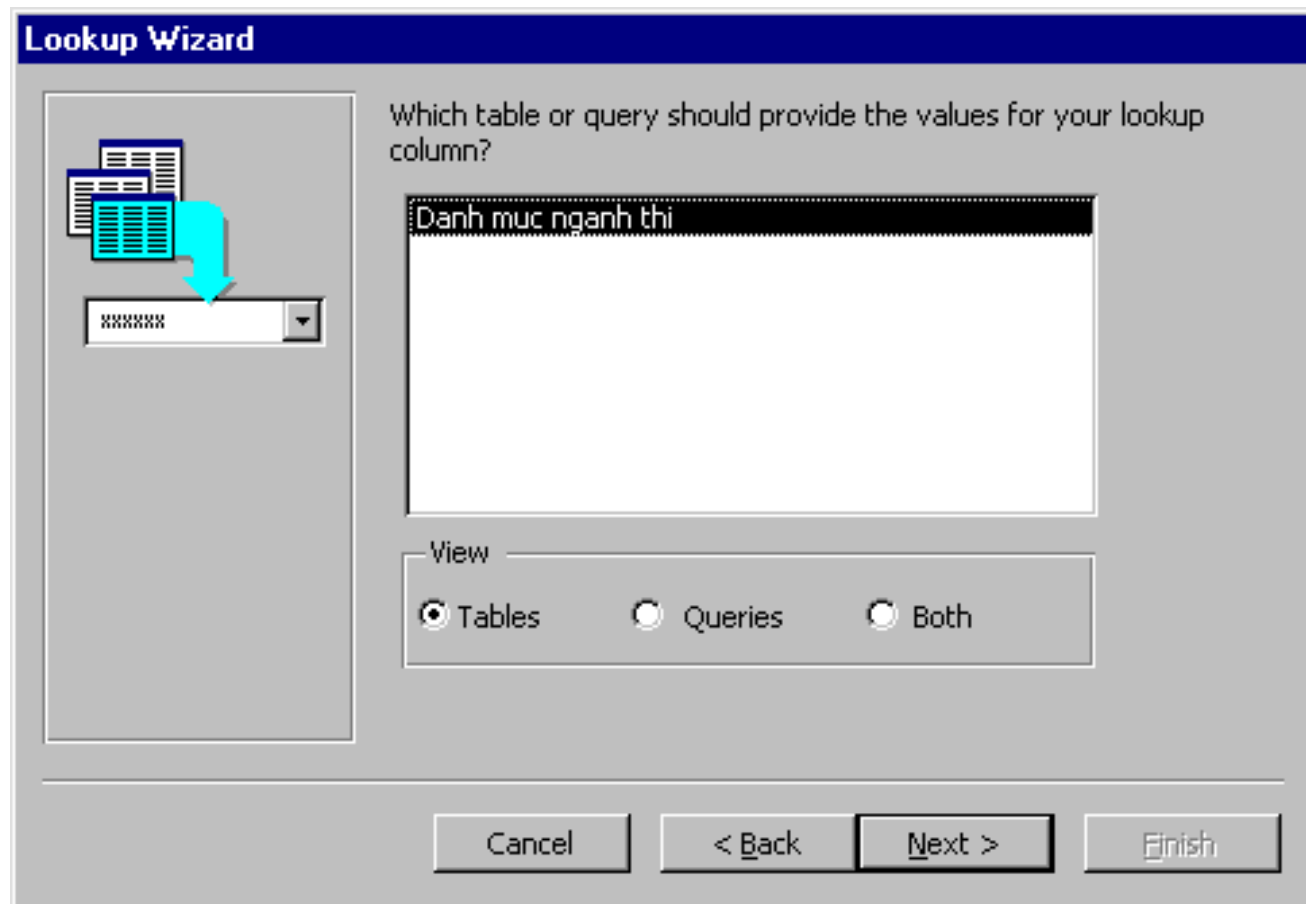
➤ (2) Chọn *I will type in the values that I want*: nếu muốn giá trị trong danh sách do người sử dụng tự đưa vào



☐ Chọn (1), Access sẽ yêu cầu chọn một Table hay Query làm nguồn. Chọn Next, xuất hiện hộp đối thoại:

### III. Tạo Combo box cho trường (Lookup Wizard)

#### 2. Cách tạo: Ví dụ minh họa



- Bước 2: Ở đây, ta chọn Table Danh mục ngành thi, chọn Next, xuất hiện hộp đối thoại:

### III. Tạo Combo box cho trường (Lookup Wizard)

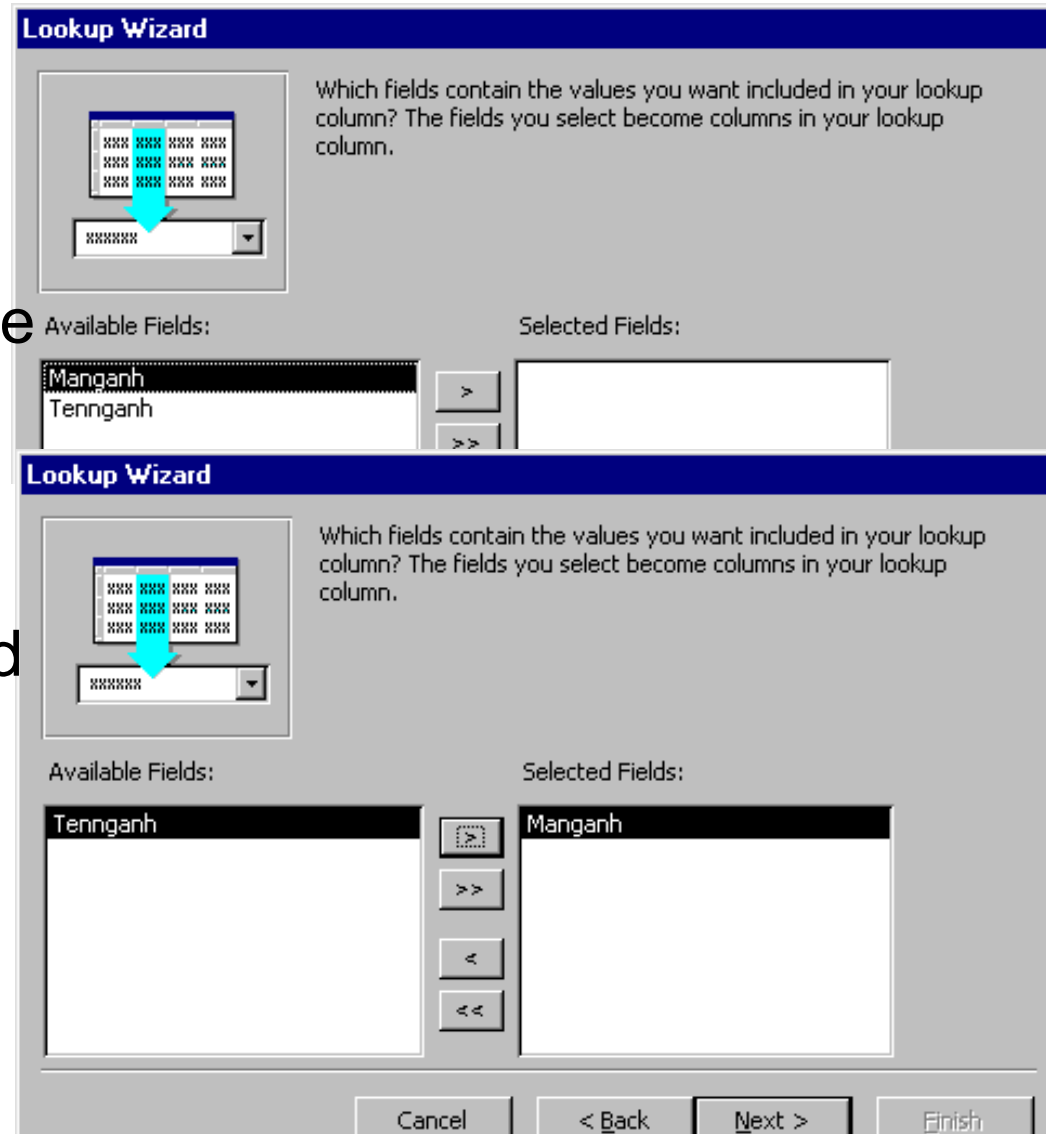
#### 2. Cách tạo: Ví dụ minh họa

#### ■ Bước 3:

- Available Fields: liệt kê các trường có trong Table

- Chọn trường phù hợp rồi kích nút > để đưa sang danh sách Selected Fields. Ở đây ta chọn trường Manganh

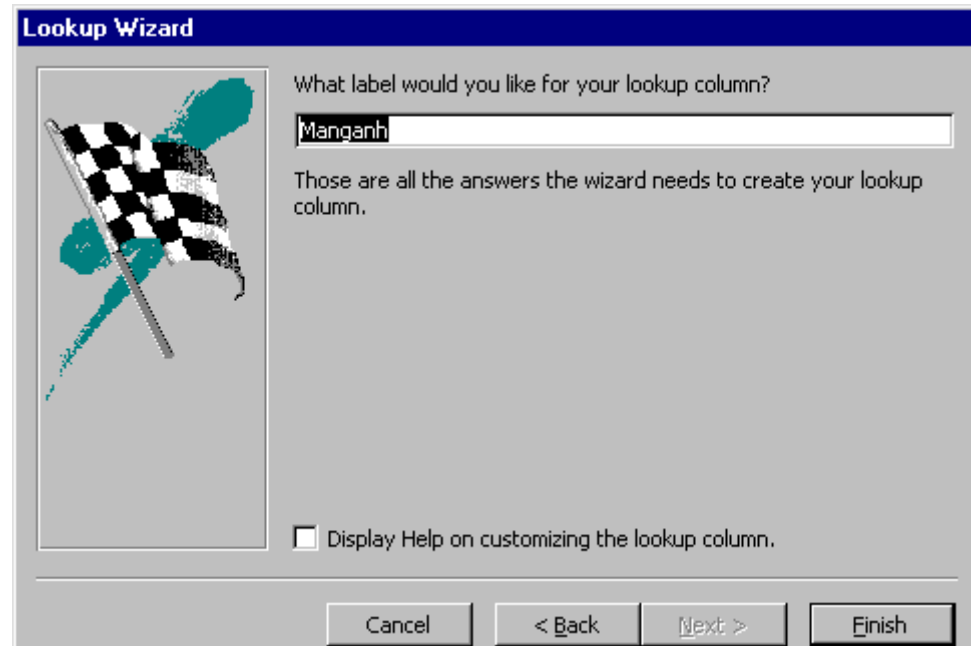
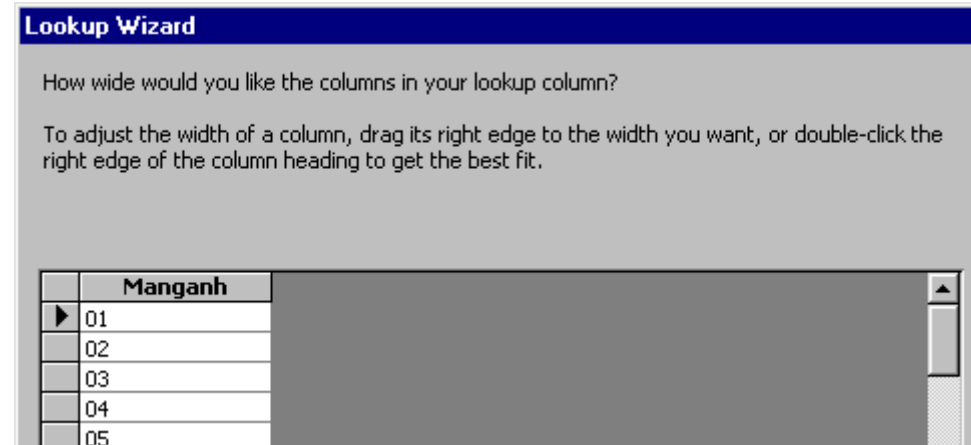
- Chọn Next, xuất hiện hộp đối thoại sau:



### III. Tạo Combo box cho trường (Lookup Wizard)

#### 2. Cách tạo: Ví dụ minh họa

- Bước 4:
  - Chọn Next, xuất hiện hộp đối thoại sau:
  - Chọn Finish để hoàn thành quá trình



### III. Tạo Combo box cho trường (Lookup Wizard)

#### 2. Cách tạo: Ví dụ minh họa

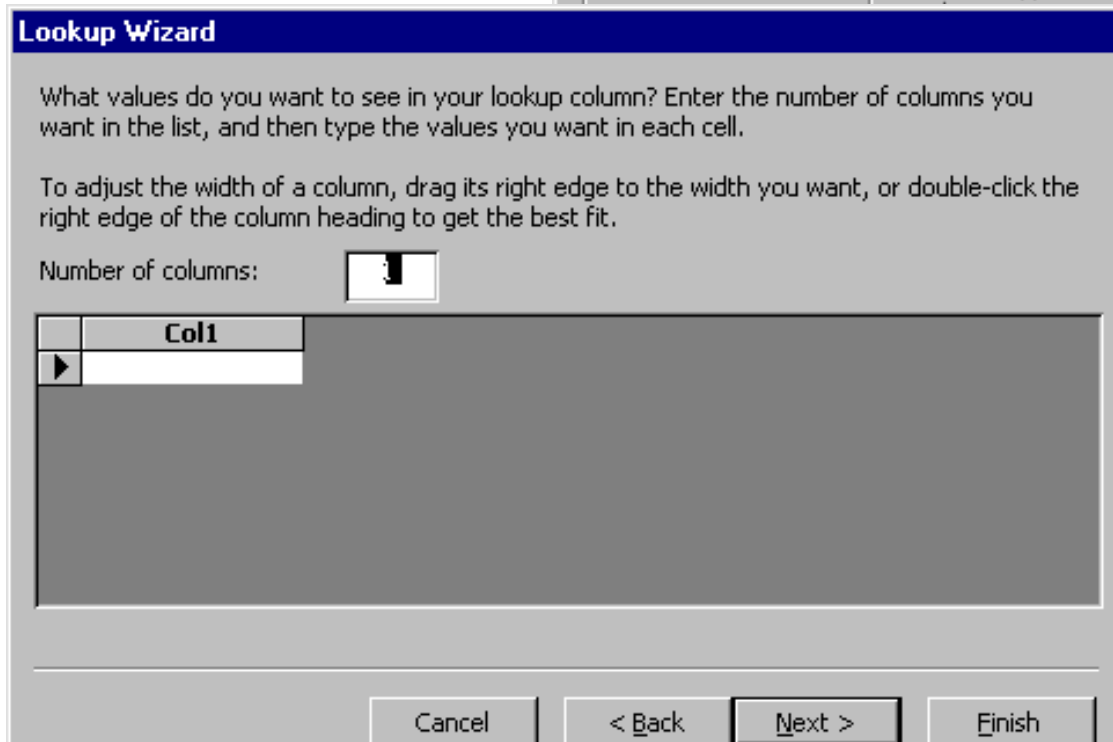
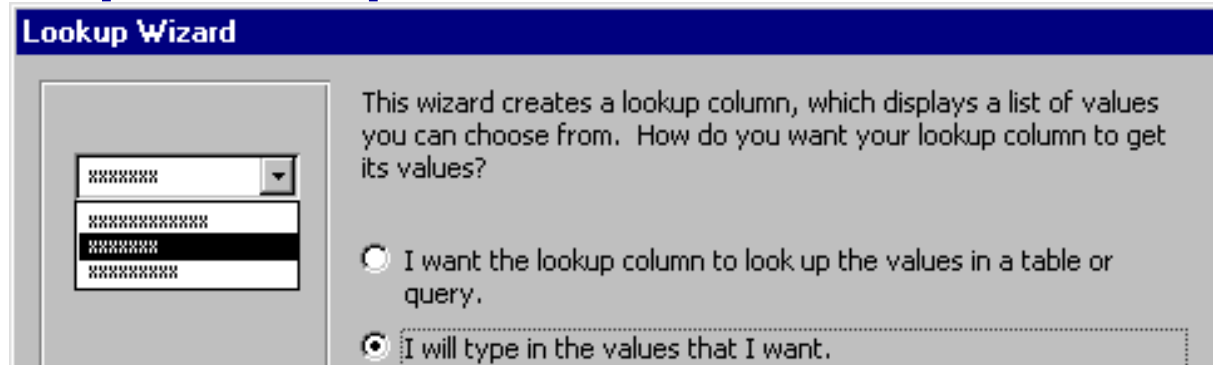
- Mở table Danh sách thi sinh để nhập liệu, bạn sẽ thấy rằng, tại trường Manganh, một danh sách đổ xuống cho bạn chọn mã ngành

Danh sách thi sinh : Table					
	Sobaodanh	Tenthisinh	Manganh	DiemTC	Ketqua
▶	DHDN001	Trần Ngọc Châu		0	
*			01	0	
			02		
			03		
			04		
			05		
			06		
			07		
			08		

## III. Tạo Combo box cho trường (Lookup Wizard)

### 2. Cách tạo: Ví dụ minh họa

☐ Chọn (2) để tự nhập vào các giá trị trong Combo box. Chọn Next, xuất hiện hộp đối thoại:



➤ Lần lượt nhập vào các giá trị trong Col1, (giả sử trong bảng DS Nhan vien) như sau:

### III. Tạo Combo box cho trường (Lookup Wizard)

#### 2. Cách tạo: Ví dụ minh họa

- Chọn Next để sang bước tiếp theo, xuất hiện hộp thoại:
- Chọn Finish để hoàn thành

**Lookup Wizard**


What values do you want to see in your lookup column? Type the values you want in the list, and then type the values you want in the list.

To adjust the width of a column, drag its right edge of the column heading to get the desired width.

Number of columns:

	Col1
	Hành chính
	Kế toán
	Kỹ thuật
	Vật tư
*	

**Lookup Wizard**



What label would you like for your lookup column?

Those are all the answers the wizard needs to create your lookup column.

Display Help on customizing the lookup column.

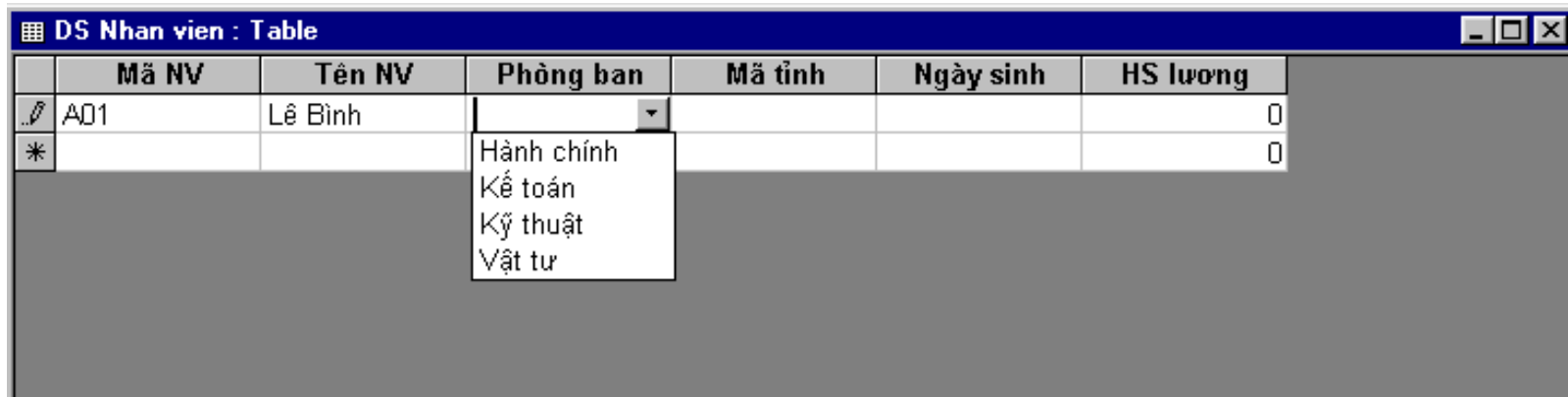
Cancel < Back Next > Finish

Cancel < Back Next > Finish

### III. Tạo Combo box cho trường (Lookup Wizard)

#### 2. Cách tạo: Ví dụ minh họa

- Kết quả khi mở bảng DS Nhan vien để nhập liệu, bạn sẽ thấy rằng, tại trường Phòng ban, một danh sách đổ xuống cho bạn chọn tên phòng ban. Các tên này do ta tự nhập vào trong các bước trên



	Mã NV	Tên NV	Phòng ban	Mã tỉnh	Ngày sinh	HS lương
✎	A01	Lê Bình				0
*			Hành chính Kế toán Kỹ thuật Vật tư			0




## IV. Khoá chính

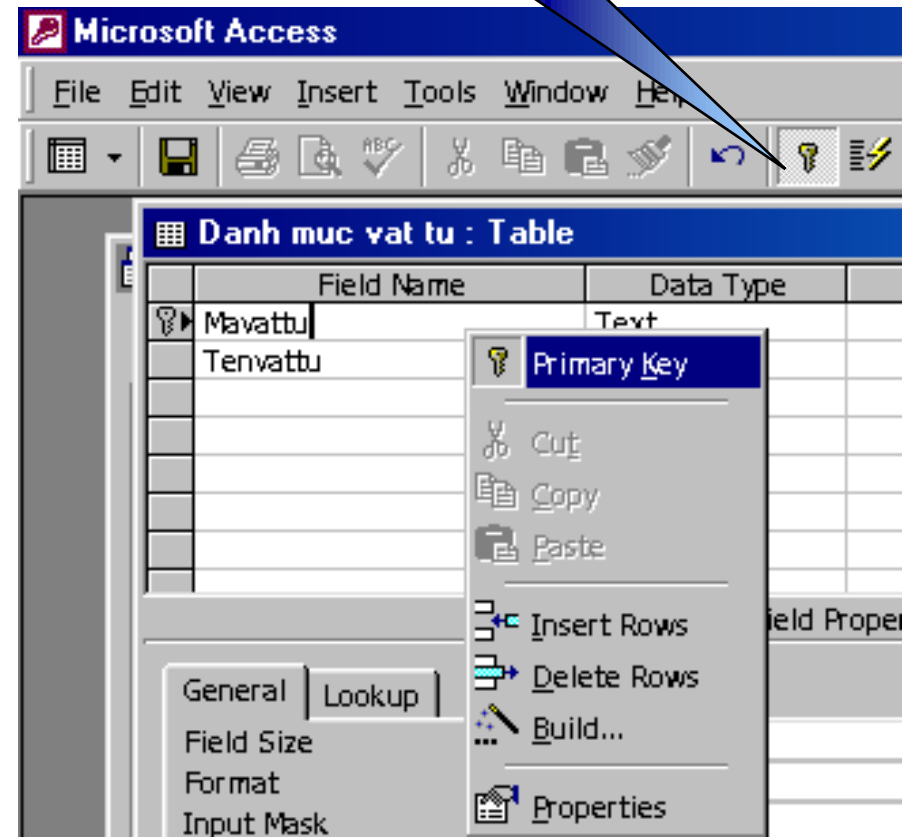
### 1. Định nghĩa

- Khoá chính là một hoặc nhiều trường xác định duy nhất một bản ghi
- Lợi ích của việc tạo khoá chính: Access dùng khoá chính để tạo sự liên kết giữa các bảng

## IV. Khoá chính


### 2. Cách đặt khoá chính:

- Tại cửa sổ thiết kế Table: Nếu mục khoá chính là
  - một trường: đưa con trỏ chuột đến dòng chứa tên trường cần chọn làm mục khoá chính
  - kết hợp của nhiều trường: nhấn giữ phím Shift sau đó chọn tất cả các trường cần chọn làm mục khoá chính
- Chọn lệnh Edit / Primary Key (hoặc nhấn nút phải, chọn Primary Key, hoặc chọn biểu tượng ): trường được chọn sẽ có hình chìa khoá ở đầu



## IV. Khoá chính

### 3. Huỷ khoá chính: Tại cửa sổ thiết kế Table

- Chọn lại lệnh Edit / Primary Key một lần nữa (hoặc nhấn nút phải, chọn Primary Key, hoặc chọn biểu tượng ): hình chìa khoá ở đầu sẽ biến mất

## V. Lập quan hệ giữa các bảng

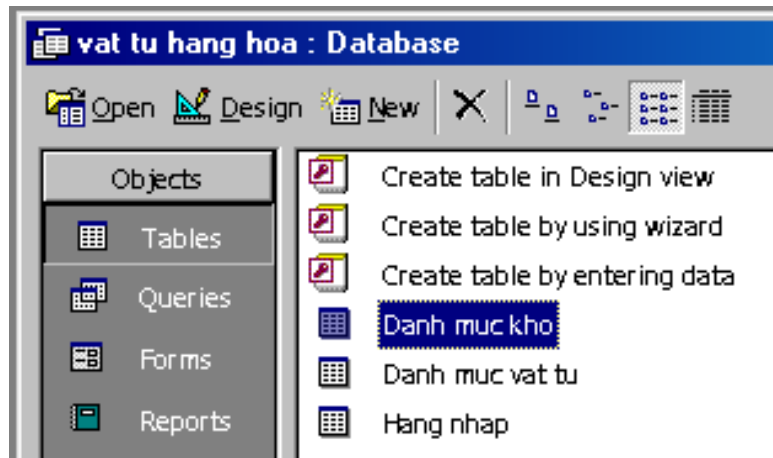
### 1. Công dụng

- Ta thường đặt quan hệ giữa một trường của bảng này với một trường của các bảng khác để liên kết dữ liệu giữa các bảng với nhau (các trường quan hệ cùng tên, cùng kiểu)
- Access dùng quan hệ để đảm bảo những ràng buộc toàn vẹn giữa các bảng liên quan trong các phép: thêm, sửa, xoá mẫu tin

## V. Lập quan hệ giữa các bảng

### 2. Cách tạo

➤ Giả sử trong CSDL Vat tu hang hoa ta có 3 bảng với cấu trúc:



Field Name	Data Type
Makho	Text
tenkho	Text

Field Name	Data Type
Mavattu	Text
Tenvattu	Text
Donvitinh	Text

Field Name	Data Type
Sohoadon	Text
Mavattu	Text
Makho	Text
Soluong	Number
Dongia	Number

➤ Ta sẽ lập quan hệ cho trường **Makho** trong bảng [**Danh muc kho**] và trường **Mavattu** trong bảng [**Danh muc vat tu**] với hai trường **Makho** và **Mavattu** trong bảng [**Hang nhap**]

## V. Lập quan hệ giữa các bảng

### 2. Cách tạo



- Tại cửa sổ Database (1), chọn lệnh Tools / Relationships, xuất hiện màn hình Show Table (Nếu không xuất hiện, chọn lệnh View/Show Table):
- Lần lượt chọn Table cần lập quan hệ, chọn Add. Sau khi chọn xong tất cả Table, chọn Close để đóng hộp thoại

## V. Lập quan hệ giữa các bảng

### 2. Cách tạo

- Trong cửa sổ Relationships, ta thiết lập các quan hệ. Có hai loại quan hệ trong Access
  - Quan hệ một-một: mỗi mẫu tin của bảng A có một mẫu tin tương ứng trong bảng B
  - Quan hệ một-nhiều: mỗi mẫu tin của bảng A có nhiều mẫu tin tương ứng trong bảng B



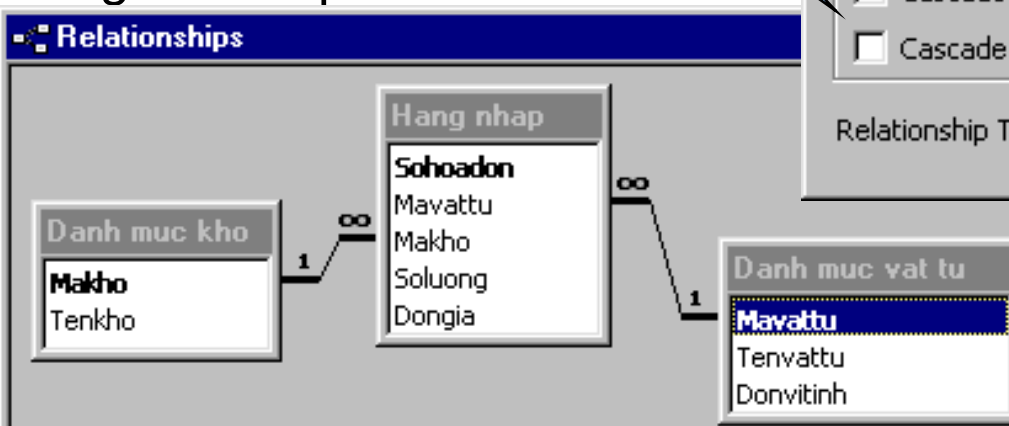
## V. Lập quan hệ giữa các bảng

### 2. Cách tạo

➤ Chọn một trường từ bảng này và kéo sang trường tương ứng của bảng quan hệ. Giả sử ta thiết lập quan hệ cho trường Makho trong bảng [Danh mục kho] và bảng [Hang nhập], xuất hiện hộp thoại Edit Relationships

Chọn một kiểu quan hệ

➤ Chọn nút Create để thiết lập quan hệ, cuối cùng có kết quả sau:





## V. Lập quan hệ giữa các bảng

### 3. Chú ý

➤ Khi đã đánh dấu mục Enforce Referential Integrity thì có thể sử dụng thêm các tùy chọn sau

- Cascade Update Related Fields: khi sửa giá trị của trường khoá trong bảng chính, giá trị tương ứng của các bản ghi trong trường quan hệ sẽ bị sửa theo (Vd: khi thay đổi manhanvien trong bảng [Danh sach nhan vien] thì manhanvien tương ứng trong bảng [Ngoai ngu] cũng thay đổi theo)
- Cascade Delete Related Records: khi xoá một bản ghi trong bảng chính, các bản ghi tương ứng trong bảng quan hệ sẽ xoá (Vd: khi xoá một bản ghi trong bảng [Danh sach nhan vien] thì các bản ghi tương ứng trong bảng [Ngoai ngu] cũng bị xoá theo)




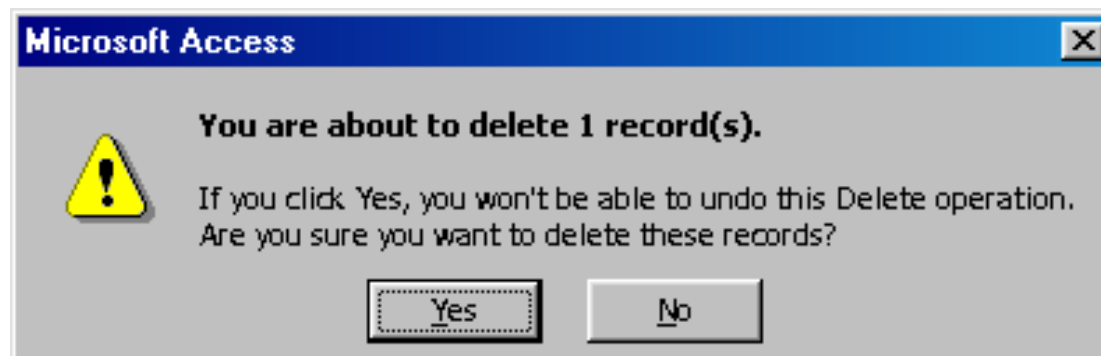
## VI. Các thao tác dữ liệu ở chế độ Datasheet

### 1. Thêm mẫu tin

- Gọi lệnh Insert /New Record (hoặc chọn biểu tượng )



### 2. Xoá mẫu tin

- Chọn các mẫu tin cần xoá
- Gọi lệnh Edit / Delete (nhấn phím Delete hoặc chọn biểu tượng )
- Chú ý: Khi xoá mẫu tin, Access sẽ đưa ra thông báo để NSD xác nhận lại có xoá hay không



## VI. Các thao tác dữ liệu ở chế độ Datasheet

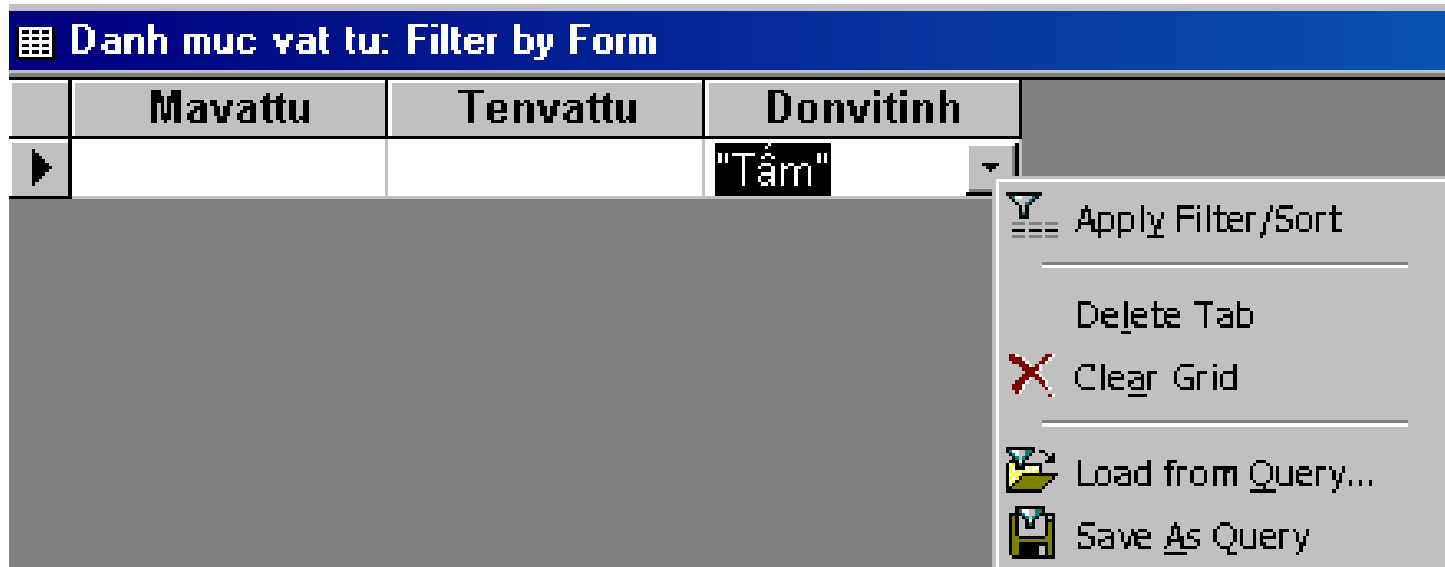
### 3. Sắp xếp mẫu tin

- Chọn trường cần sắp xếp
- Gọi lệnh Records / Sort / Sort Ascending - Sort Descending (hoặc chọn biểu tượng   )

## VI. Các thao tác dữ liệu ở chế độ Datasheet

### 4. Lọc bản ghi

- Chọn lệnh Records / Filter / Filter By Form, xuất hiện hộp thoại Filter By Form như sau:



- Muốn lọc trường nào, ta bấm chuột vào trường đó, chọn giá trị làm điều kiện để lọc
- Nhấp phải chuột, chọn Apply Filter/Sort: Access sẽ liệt kê các bản ghi thoả mãn điều kiện

## Chương III : Truy vấn dữ liệu (Query)

### I. Khái niệm

- Truy vấn là một công cụ mạnh của Access dùng để: tổng hợp, sắp xếp và tìm kiếm dữ liệu
- Truy vấn cho phép NSD thống kê số liệu (truy vấn dữ liệu) theo ý mình, xây dựng các báo cáo tổng hợp dưới nhiều hình thức khác nhau trên dữ liệu gốc là bảng (Table)



**II. Các loại truy vấn thông dụng** : Có nhiều loại truy vấn, nhưng thông dụng nhất là các loại sau:

**1. Select Query (Truy vấn chọn số liệu)**: được sử dụng phổ biến nhất, dùng để:

- Chọn lọc các mẫu tin
- Thêm các trường mới là kết quả thực hiện các phép tính trên các trường của bảng nguồn.
- Đưa vào các điều kiện tìm kiếm, lựa chọn

**2. Update Query (Truy vấn cập nhật)**: dùng để sửa đổi đồng loạt nhiều mẫu tin của một hay nhiều bảng

## II. Các loại truy vấn thông dụng

**3. Delete Query (Truy vấn xoá số liệu):** xoá một nhóm các mẫu tin từ một hay nhiều bảng

**4. Crosstab Query (Truy vấn tham chiếu chéo):** kết nhóm số liệu theo chủng loại và hiển thị số liệu dưới hình thức của một bảng tính kèm theo số liệu tổng hợp ngang và dọc

**5. Append Query (Truy vấn thêm):** Dùng để thêm (nối) các bảng ghi vào phần cuối của một bảng.

**6. Make Table Query (Truy vấn tạo bảng):** cho tạo một bảng mới từ số liệu của Query

### III. Cách tạo truy vấn

➤ Thường dùng hai cách tạo truy vấn ứng với hai lựa chọn:

**in Design view** và **by using wizard**

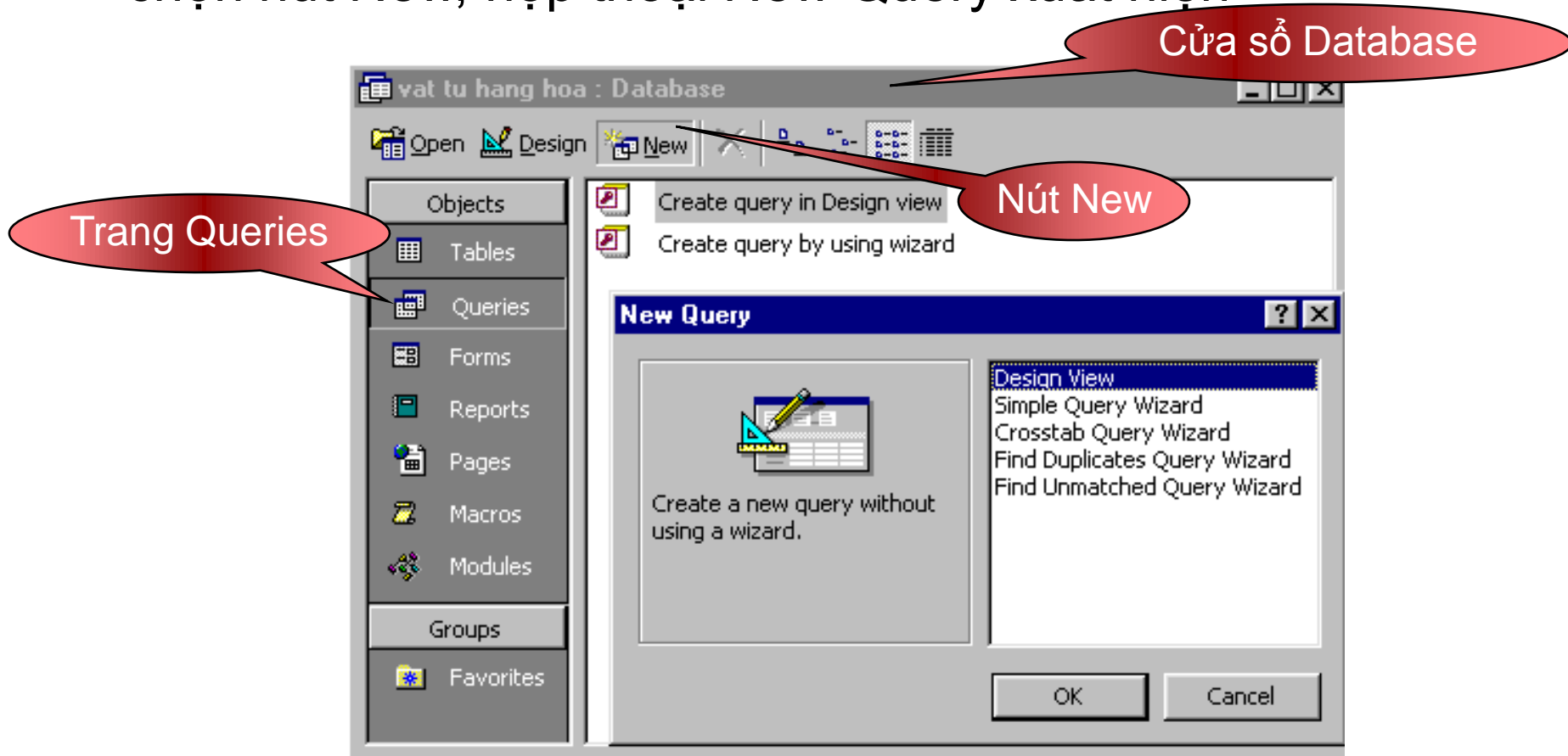
➤ By using wizard: truy vấn được xây dựng bằng công cụ wizard của Access, ta chỉ việc trả lời các câu hỏi theo yêu cầu

➤ In Design view: truy vấn được thiết kế tùy theo mục đích của người sử dụng. Ta thường chọn cách này. Cách thực hiện như sau:



### III. Cách tạo truy vấn

➤ **Bước 1:** Trong cửa sổ Database, chọn trang Queries, chọn nút New, hộp thoại New Query xuất hiện

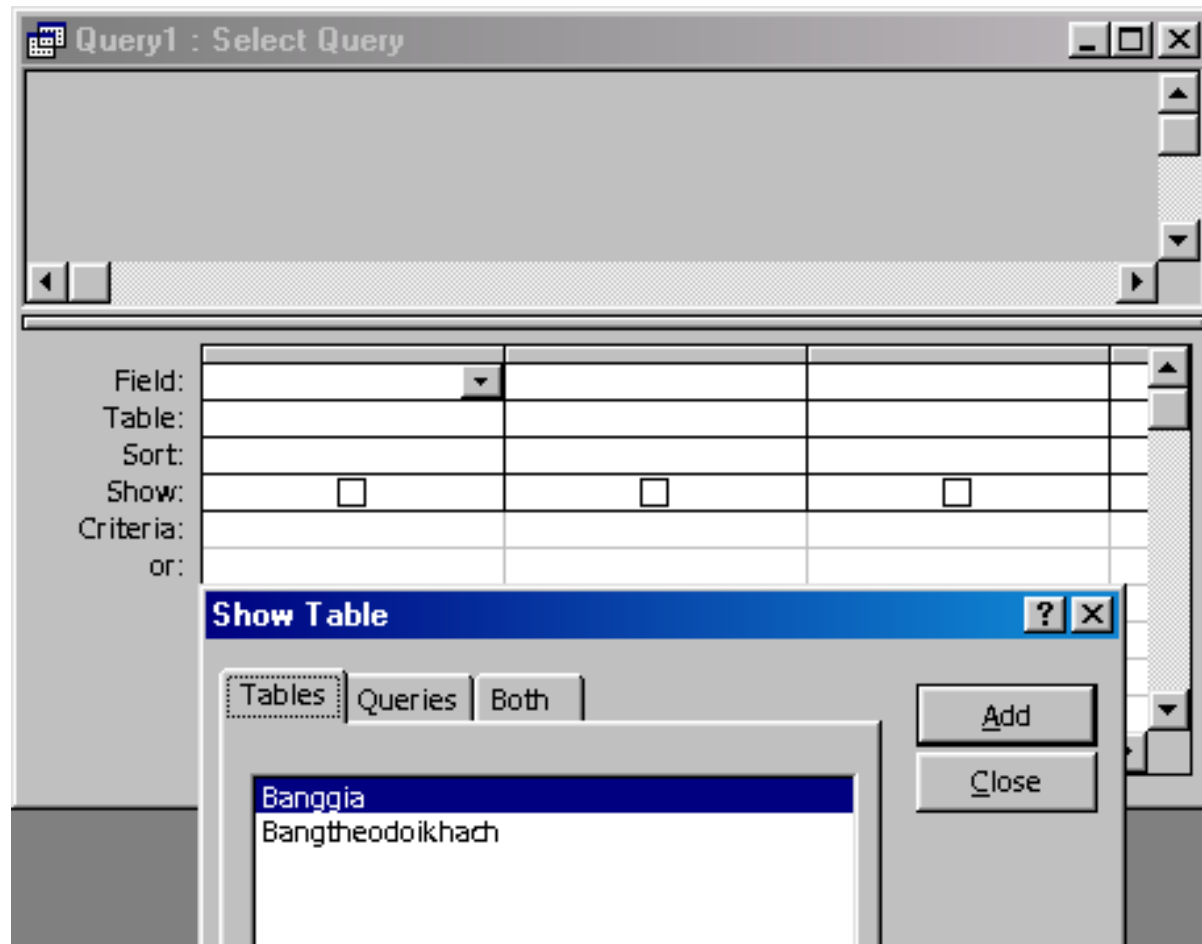


- Chọn Design View, OK, xuất hiện cửa sổ như sau

### III. Cách tạo truy vấn

➤ **Bước 2:** chọn nguồn dữ liệu cho truy vấn

- Cửa sổ Show Table cho chọn các bảng (Tables), các truy vấn (Queries) hoặc chọn cả bảng và truy vấn để làm nguồn dữ liệu cho truy vấn sắp tạo



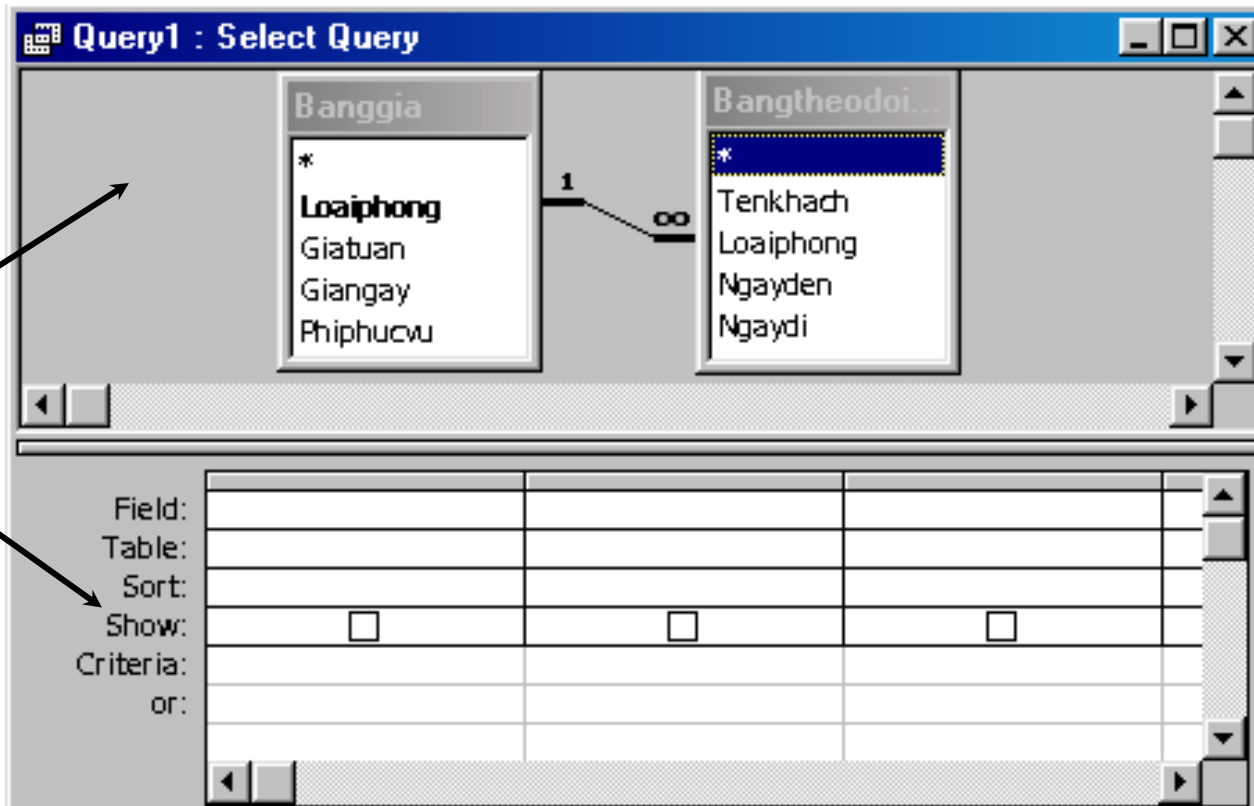
- Chọn nguồn dữ liệu phù hợp, chọn Add. Sau khi chọn xong tất cả, chọn Close để trở về cửa sổ Select Query

### III. Cách tạo truy vấn

#### ✦ Mô tả cửa sổ Select Query

- Phần trên: chứa các bảng và truy vấn đã chọn làm nguồn
- Phần dưới: gọi là lưới thiết kế QBE (Query By Example) dùng chứa các trường của truy vấn mới cần xây dựng

nhấn F6 để  
di chuyển  
giữa 2 phần



### III. Cách tạo truy vấn

#### ✦ Mô tả cửa sổ Select Query

- Field: chứa tên trường hoặc biểu thức làm kết quả truy vấn
- Table: ghi tên nguồn dữ liệu của trường được chọn tại dòng field
- Show: chọn trường có hiển thị hay không
- Sort: chọn cách sắp xếp dữ liệu của trường
- Criteria: biểu thức tiêu chuẩn chọn lọc dữ liệu
- Or: điều kiện hoặc của biểu thức tiêu chuẩn chọn lọc dữ liệu (nếu có)

### III. Cách tạo truy vấn

- **Bước 3:** Thiết kế truy vấn theo yêu cầu

Đưa các trường vào truy vấn: nhấn giữ chuột kéo trường thích hợp thả vào cột của lưới thiết kế, hoặc nhấp đúp tại tên trường

- ✦ **Ghi chú:** Cách thêm trường biểu thức:

Chọn một ô trống trên hàng Field, gõ tên trường biểu thức, dấu : (hai chấm), biểu thức.

Ví dụ: Tongdiem: [Diemtoan]+[Diemly]+[Diemhoa]

- **Bước 4:** Lập tiêu chuẩn lựa chọn (điều kiện tìm kiếm)  
Trong ô Criteria của trường cần lập biểu thức, gõ biểu thức  
Ví dụ: gõ vào L1C (tìm khách ở loại phòng L1C)

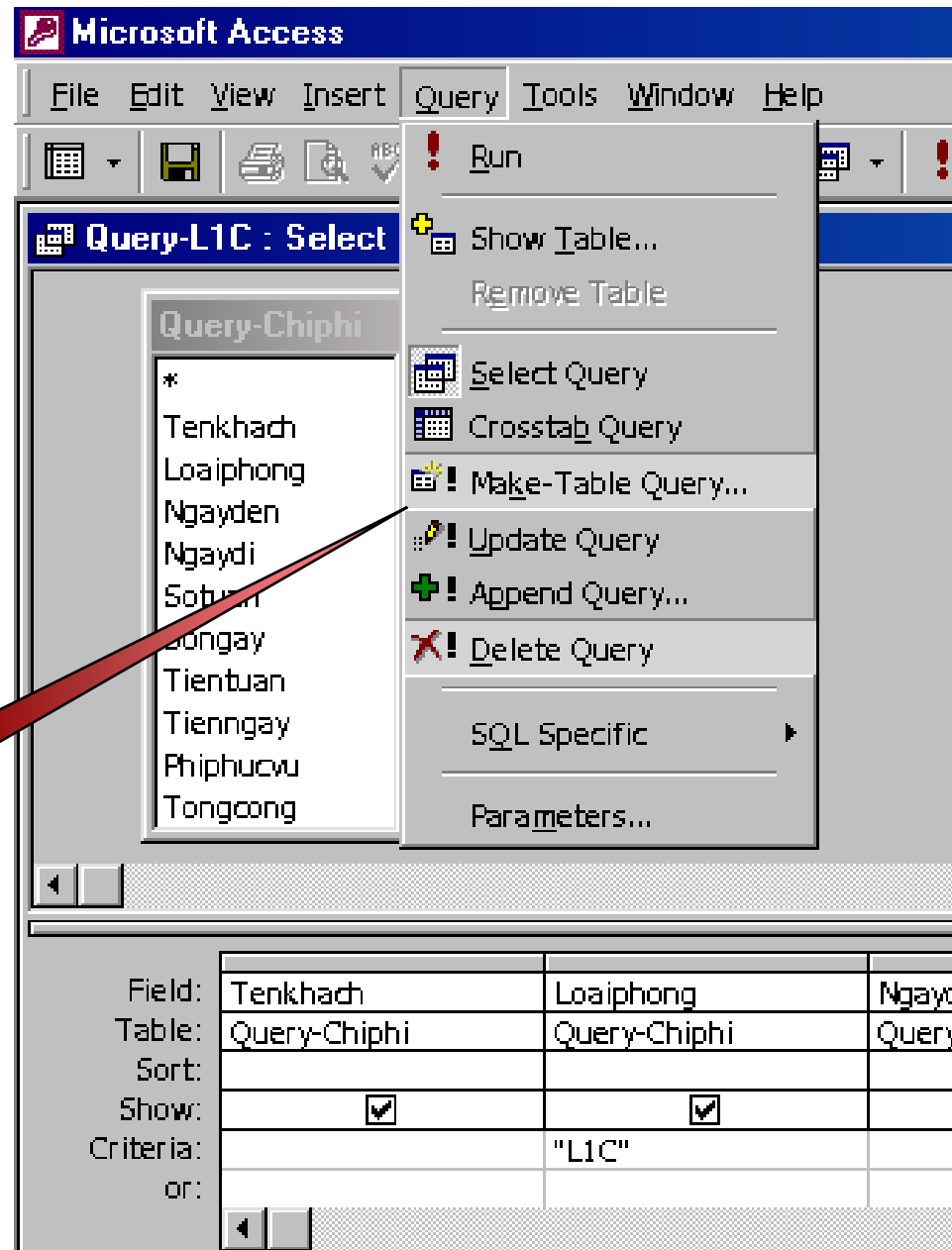
- Lưu truy vấn vừa tạo : Gọi lệnh File / Save

### III. Cách tạo truy vấn

✦ **Ghi chú:** Thay đổi loại truy vấn


Chọn lệnh Query, chọn trong danh sách loại Query phù hợp

Chọn loại Query



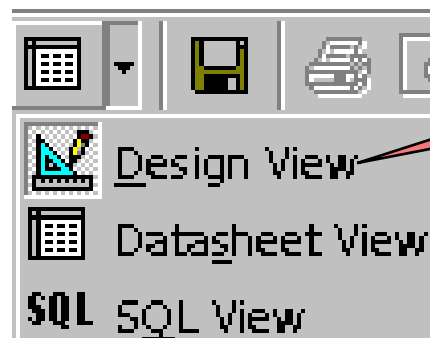
## IV. Thực hiện truy vấn

Trong cửa sổ thiết kế truy vấn

➤ Gọi lệnh Query / Run (hoặc chọn biểu tượng  trên thanh công cụ)

➤ Hoặc chọn biểu tượng View 

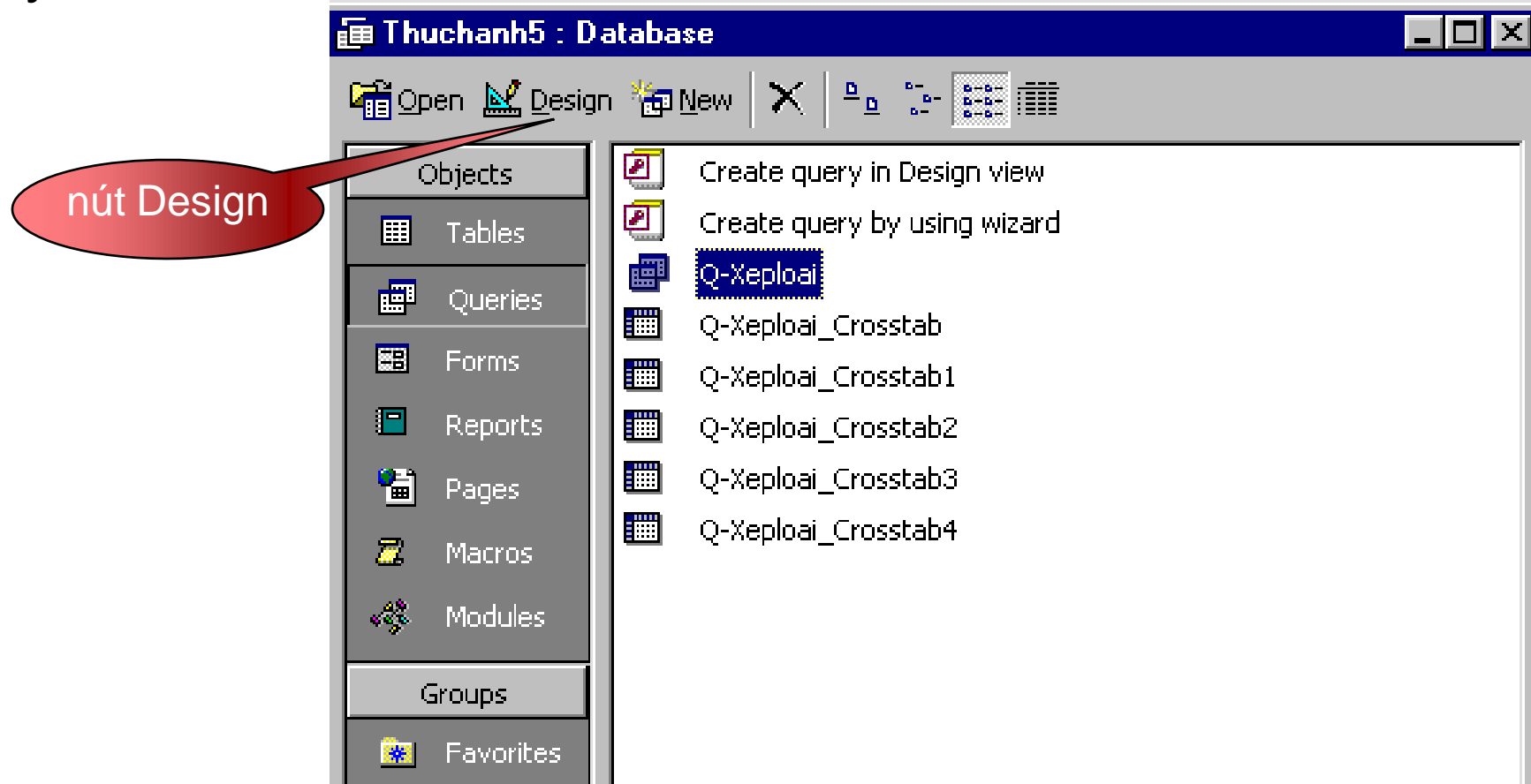
➤ Hoặc chọn Design View



Design View

## V. Hiệu chỉnh truy vấn

Trong cửa sổ Database, chọn tên truy vấn, chọn nút Design để mở cửa sổ thiết kế. Thực hiện hiệu chỉnh theo yêu cầu





## VI. Điều kiện trong truy vấn: khi thiết kế truy vấn và thiết lập biểu thức điều kiện, có thể sử dụng

### ➤ Toán tử toán học

- $+$ ,  $-$ ,  $*$ ,  $/$ , lũy thừa ( $\wedge$ )
- Thương số nguyên (chia lấy phần nguyên):  $/$
- Số dư (chia lấy phần dư): **Mod**

### ➤ Toán tử logic

- Not: cho kết quả ngược lại (Not đúng= sai, Not sai= đúng)
- And (và): cho kết quả đúng khi tất cả điều kiện đúng, nếu một điều kiện nào đó sai thì cho kết quả sai
- Or (hoặc): cho kết quả đúng khi có ít nhất một điều kiện đúng, nếu tất cả điều kiện sai thì cho kết quả sai

## VI. Điều kiện trong truy vấn

### ➤ Toán tử so sánh

<, <=, >, >=, =, <> (không bằng)

### ➤ Toán tử so sánh đặc biệt

#### ▪ **Between ... And ...** (Nằm trong khoảng)

- Vd: [slg] Between 150 And 200

#### ▪ **Like** (gần giống):

- (\*) Tổ hợp bất kỳ - “Anh Ban” Like “\*B\*”
- (?) Ký tự bất kỳ - “Access” Like “???e\*”
- (#) Ký số bất kỳ - 2001 Like “##0#”: True

## VI. Điều kiện trong truy vấn

### ➤ Toán tử so sánh đặc biệt

- **Null, Not null**: kiểm tra dữ liệu đã có hay chưa

- Vd: Để tìm các thí sinh thiếu Ngày sinh, ta đưa giá trị **Null** vào ô Criteria của trường [Namsinh]

- Vd: Để tìm các khách hàng đã có địa chỉ, ta đưa giá trị **Not null** vào ô Criteria của trường [Diachi]

## VI. Điều kiện trong truy vấn

### ➤ Toán tử ghép nối

#### ▪ & (nối chuỗi)

- “Đà” & “ Nẵng”: “Đà Nẵng”
- 123 & 456: 123456
- 100 & “năm cô đơn”: 100 năm cô đơn
- Date() & “Nhâm Ngọ”: 14/6/2002 Nhâm Ngọ
- “Vĩ tuyến” & 90/2 & “ngày đêm”: Vĩ tuyến 45 ngày đêm

#### ▪ + (cộng với)

- 123 + 456 : 579
- “Con thuyền” + “không bến”: Con thuyền không bến
- 1001 + “đêm” : **Type mismatch** (kết quả sai)

## VI. Điều kiện trong truy vấn

### ➤ Hằng trị (constants)

- **True/ False/ Null**: hằng trị đúng/ sai/ rỗng

### ➤ Dấu rào (Delimiter)

- **“...”**: Rào giá trị chuỗi

- Vd: “43 Thanh Thủy - Đà Nẵng”

- **[ ... ]**: Rào tên biến

- Vd: [SoLuong] \* [DonGia]

- **#.../.../...#**: Rào giá trị ngày

- #23/04/2000#

## VI. Điều kiện trong truy vấn

➤ Điều kiện cùng thoả mãn/hoặc thoả mãn

- Các điều kiện viết cùng trên dòng **Criteria** là các điều kiện đồng thời thoả mãn
- Ví dụ: để tìm các khách ở loại phòng "L1C" và đến trong tháng 6, trên dòng Criteria, ta ghi

Field:	Tenkhach	Loaiphong	Ngayden
Table:	Query-Chiphi	Query-Chiphi	Query-Chiphi
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:		"L1C"	Month([Ngayden])=6
or:			

Điều kiện ghi  
trên cùng 1 dòng

Kết quả

	Tenkhach	Loaiphong	Ngayden
	Ha Ngoc Chinh	L1C	12/06/2002

Record: 2 of 2

## VI. Điều kiện trong truy vấn

- Các điều kiện viết trên dòng **Criteria** và dòng **Or** là các điều kiện không cần đồng thời thoả mãn
- Ví dụ: để tìm các khách ở loại phòng “L1C” hoặc đến trong tháng 6, trên dòng Criteria và dòng Or, ta ghi

Field:	Tenkhach	Loaiphong	Ngayden
Table:	Query-Chiphi	Query-Chiphi	Query-Chiphi
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:		"L1C"	
or:			Month([Ngayden])=6

Điều kiện ghi trên dòng Criteria và dòng Or

Kết quả

Tenkhach	Loaiphong	Ngayden
Nguyen Thuan An	L1C	10/05/2002
Ha Ngoc Chinh	L1C	12/06/2002
Nguyen Lan Khue	L1B	26/06/2002

Record: 4 of 4

## VII. Một số hàm thông dụng

### ➤ Hàm xử lý chuỗi

#### ▪ Hàm **LEFT, RIGHT**

**LEFT(Biểu thức chuỗi, n)**

**RIGHT(Biểu thức chuỗi, n)**

- Công dụng: trả về giá trị chuỗi bằng cách lấy n ký tự bên trái (LEFT) hoặc bên phải (RIGHT) của biểu thức chuỗi

- Ví dụ: Trên dòng Criteria ghi:

`RIGHT([Mahoadon],2)="NB"`: tìm các chứng từ có 2 ký tự bên phải của Mã hoá đơn là "NB"

#### ▪ Hàm **LEN(Biểu thức chuỗi)**

- Công dụng: trả về số là chiều dài của biểu thức chuỗi

- Ví dụ: `LEN("TRUNG TAM")` cho kết quả là 8



## VII. Một số hàm thông dụng

### ➤ Hàm thời gian

#### ▪ Hàm **DATE( )**

- Công dụng: trả về ngày hiện tại của hệ thống

#### ▪ Hàm **DAY**(Biểu thức ngày)

#### ▪ Hàm **MONTH**(Biểu thức ngày)

#### ▪ Hàm **YEAR** (Biểu thức ngày)

- Công dụng: trả về trị số là **ngày/tháng/năm** của biểu thức ngày

- Ví dụ: MONTH([Ngayden])=6: Tìm các khách đến trong tháng 6

## VII. Một số hàm thông dụng

### ➤ Hàm logic

- Hàm **IIF(điều kiện, trị 1, trị 2)**

- Công dụng:

Nếu **điều kiện thoả mãn**, hàm nhận **trị 1**

Nếu **điều kiện không thoả mãn**, hàm nhận **trị 2**

- Ví dụ: Ban hành điều kiện cho cột Xếp loại như sau

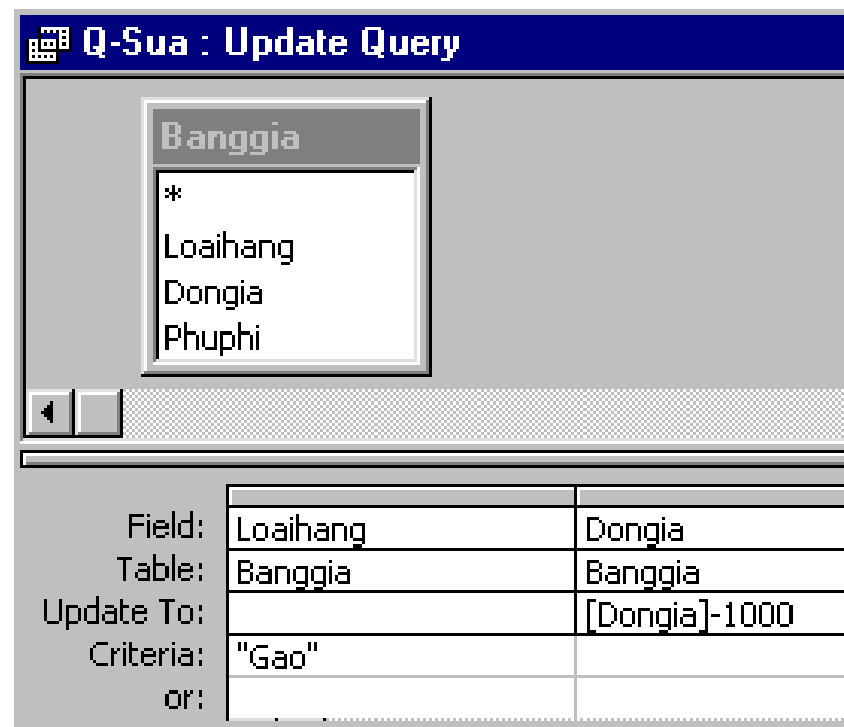
```
IIF([Diem]<5,"Yếu", IIF([Diem]<7,"Trung bình",  
IIF([Diem]<8,"Khá","Giỏi")))
```

## VIII. Một số truy vấn thông dụng

### 1. Truy vấn cập nhật (Update Query)

➤ Các bước xây dựng truy vấn cập nhật

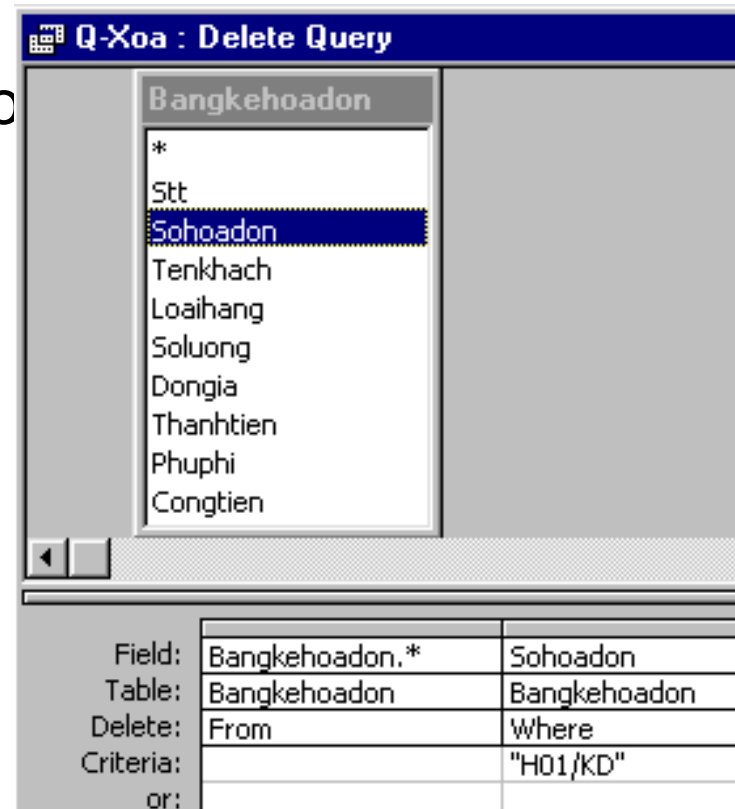
- Chọn nguồn dữ liệu (các bảng hoặc truy vấn khác)
- Chọn lệnh Query / Update Query, hiện thêm hàng **Update to** trong lưới QBE
- Đặt biểu thức vào ô **Update To** của trường cần sửa
- Criteria: ghi điều kiện tương ứng
- Ví dụ: trong hình bên, khi thực hiện Query thì đơn giá mặt hàng Gạo sẽ giảm 1000



## VIII. Một số truy vấn thông dụng

### 2. Truy vấn xoá (Delete Query)

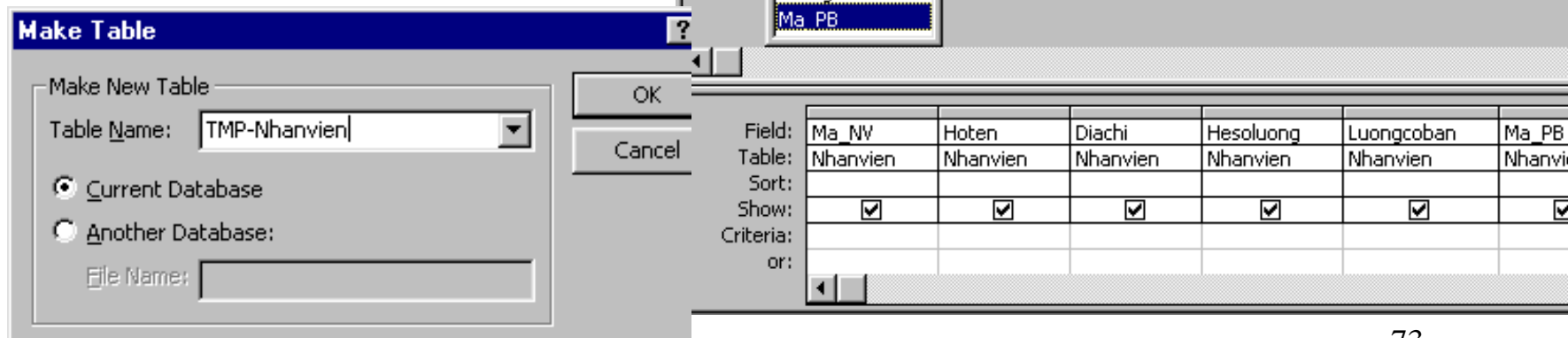
- Chọn nguồn dữ liệu (các bảng hoặc truy vấn khác)
- Chọn lệnh Query / Delete Query, hiện thêm hàng Delete trong lưới QBE
- Kéo dấu \* của bảng muốn xoá vào dòng Field, từ khoá **From** xuất hiện trên dòng Delete
- Kéo trường dùng để lập điều kiện vào dòng Field, từ khoá **Where** xuất hiện trên dòng Delete
- Ví dụ: trong hình bên, khi thực hiện Query thì các mẫu tin có Số hoá đơn là H01/KD sẽ bị xoá



## VIII. Một số truy vấn thông dụng

### 3. Truy vấn tạo bảng (Make Table Query)

- Chọn nguồn dữ liệu (các bảng hoặc truy vấn khác)
- Chọn các trường cần đưa vào bảng và kéo lần lượt vào lưới QBE
- Gọi lệnh Query/ Make Table Query, xuất hiện hộp thoại Make Table như sau:

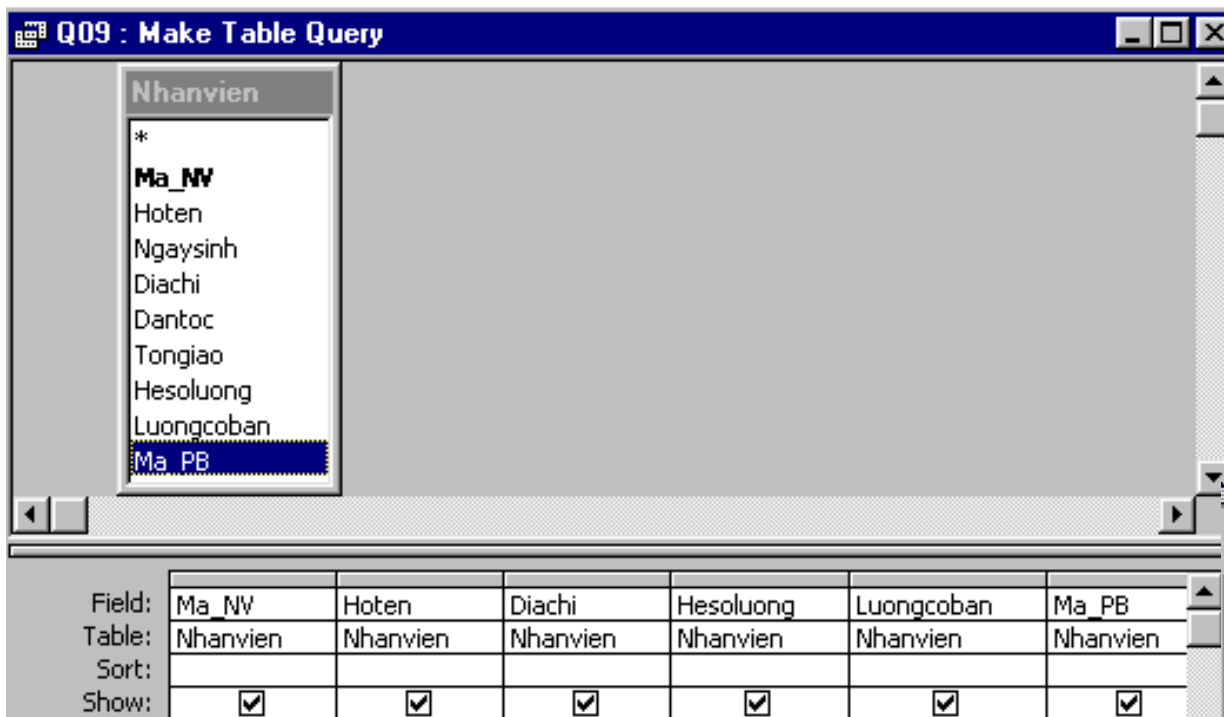


- Table Name: gõ tên cần đặt cho bảng, OK

## VIII. Một số truy vấn thông dụng

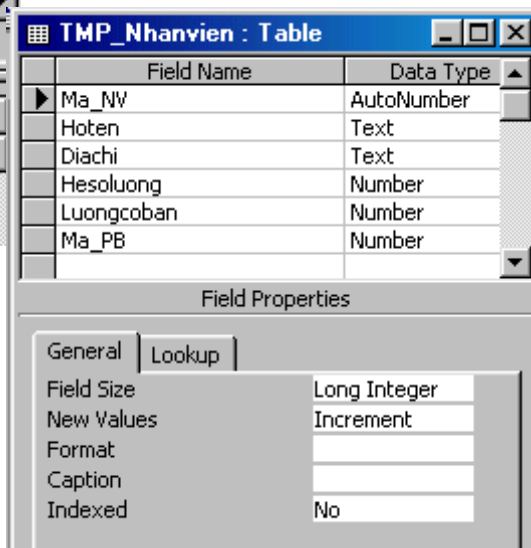
### 3. Truy vấn tạo bảng (Make Table Query)

- Cửa sổ Select Query thay đổi thành Make Table Query



- Thực hiện Query để Access tạo bảng theo yêu cầu

- Mở (cấu trúc) bảng vừa tạo để xem kết quả



## 4. Truy vấn Crosstab

- Công dụng: dùng để thống kê số liệu và trình bày kết quả theo dạng một bảng tính, với hình thức dễ đọc.
- Cách thực hiện: minh họa thông qua ví dụ sau
  - Giả sử trên cơ sở của dữ liệu

Hoten	Phai	Ngaysinh	Noisinh	Khoa	Lop	Diem	Xep loai
Hoang Tung Quan	Nam	03/01/1976	Da Nang	20	Ke toan may	9	Gioi
Tran Thi Hoa	Nu	02/07/1975	Hue	20	KTV Tin hoc	7	Kha
Le Ha Phuong	Nu	30/09/1969	Quy Nhon	20	TC Ke toan	8	Gioi
Nguyen Tien Dung	Nam	04/08/1968	Nha Trang	21	TC Tinhoc	5	Trung binh
Nong Quoc Huy	Nam	20/02/1976	can Tho	20	Ke toan may	6	Trung binh
Nguyen Thi Tuong Vy	Nu	18/09/1969	Da Nang	20	KTV Tin hoc	8	Gioi

- Cần có thống kê số liệu như sau

Lop	Si so	Gioi	Kha	Trung binh	Yeu
Ke toan may	7	5		2	
KTV Tin hoc	5	2	1		2
TC Ke toan	5	1		2	2
TC Tinhoc	5	2	1	2	

## 4. Truy vấn Crosstab

➤ B1: Chọn Queries / New / Crosstab Query Wizard, xuất hiện hộp thoại

▪ Chọn dữ liệu nguồn (từ Tables/ Queries hay cả hai

▪ Chọn Next sang bước tiếp theo

Which table or query contains the fields you want for the the crosstab query results?

To include fields from more than one table, create a query containing all the fields you need and then use this query to make the crosstab query.

View

Tables  Queries  Both

Sample:

	Header1	Header2	Header3
	TOTAL		

Cancel < Back Next > Finish



## 4. Truy vấn Crosstab

➤ B2: Chọn trường làm tiêu đề dòng

▪ Available Fields: chọn tên trường làm tiêu đề dòng (tối đa là 3)

▪ Selected Fields: chứa tên các trường được chọn

▪ Sample: mẫu thể hiện

▪ Chọn Next sang bước tiếp theo

**Crosstab Query Wizard**

Which fields' values do you want as row headings?  
You can select up to three fields.  
Select fields in the order you want information sorted. For example, you could sort and group values by Country and then Region.

Available Fields:

- Hoten
- Phai
- Ngaysinh
- Noisinh
- Khoa
- Diem**
- Xep loai

Selected Fields:

- Lop

Sample:

Lop	Header1	Header2	Header3
Lop1	TOTAL		
Lop2			
Lop3			
Lop4			

Cancel < Back Next > Finish

## 4. Truy vấn Crosstab

### ➤ B3: Chọn trường làm tiêu đề cột

- Chọn Next sang bước tiếp theo

**Crosstab Query Wizard**

Which field's values do you want as column headings?

For example, you would select Employee Name to see each employee's name as a column heading.

Hoten  
Phai  
Ngaysinh  
Noisinh  
Khoa  
Diem  
**Xep loai**

Sample:

Lop	Xep loai1	Xep loai2	Xep loai3
Lop1	TOTAL		
Lop2			
Lop3			
Lop4			

Cancel < Back Next > Finish

## 4. Truy vấn Crosstab

- B4: Chọn trường cần tính tại mỗi giao điểm dòng và cột

- Chọn Next sang bước tiếp theo

**Crosstab Query Wizard**

What number do you want calculated for each column and row intersection?

For example, you could calculate the sum of the field Order Amount for each employee (column) by country and region (row).

Do you want to summarize each row?

Yes, include row sums.

Fields:

Hoten  
Phai  
Ngaysinh  
Noisinh  
Khoa  
Diem

Functions:

Count  
First  
Last  
Max  
Min

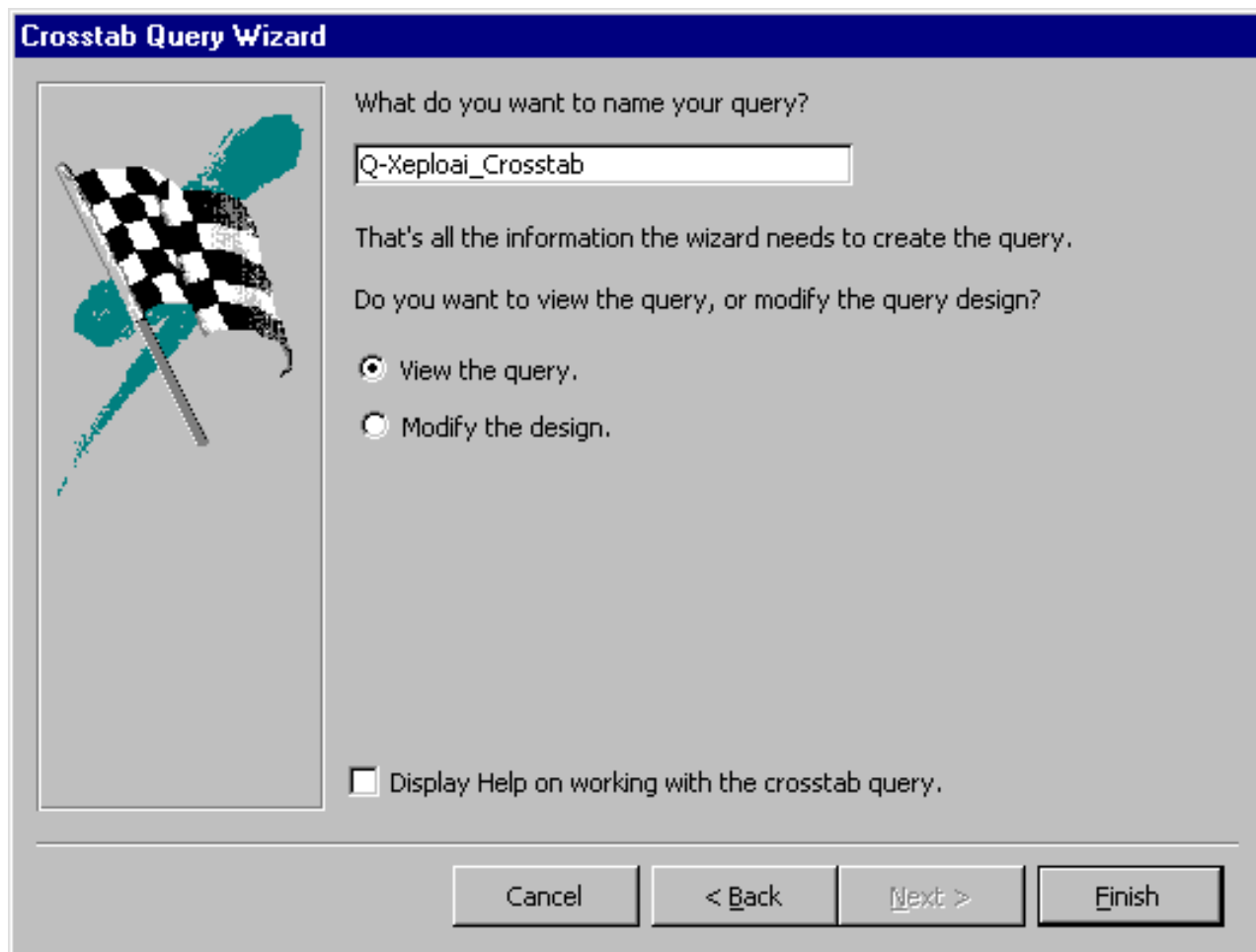
Sample:

Lop	Xep loai1	Xep loai2	Xep loai3
Lop1	Count(Hoten)		
Lop2			
Lop3			
Lop4			

Cancel    < Back    Next >    Finish

## 4. Truy vấn Crosstab

- B5: Gõ tên đặt cho Query, chọn Finish để hoàn thành



- Lưu ý: đối với các Table hoặc Query chứa dữ liệu được chọn làm nguồn, phải có ít nhất là 3 trường

## Chương IV : Form (Mẫu biểu)

### I. Khái niệm

- Form là một công cụ mạnh của Access dùng để thể hiện và cập nhật dữ liệu cho các bảng, tổ chức giao diện chương trình
  
- Nguồn dữ liệu (nếu có) của Form là một bảng hoặc một truy vấn
  - Khi có nguồn dữ liệu: Form dùng để thể hiện và cập nhật dữ liệu cho các trường nguồn
  
  - Khi không có nguồn dữ liệu: Form dùng để tổ chức giao diện chương trình



## II. Tạo Form

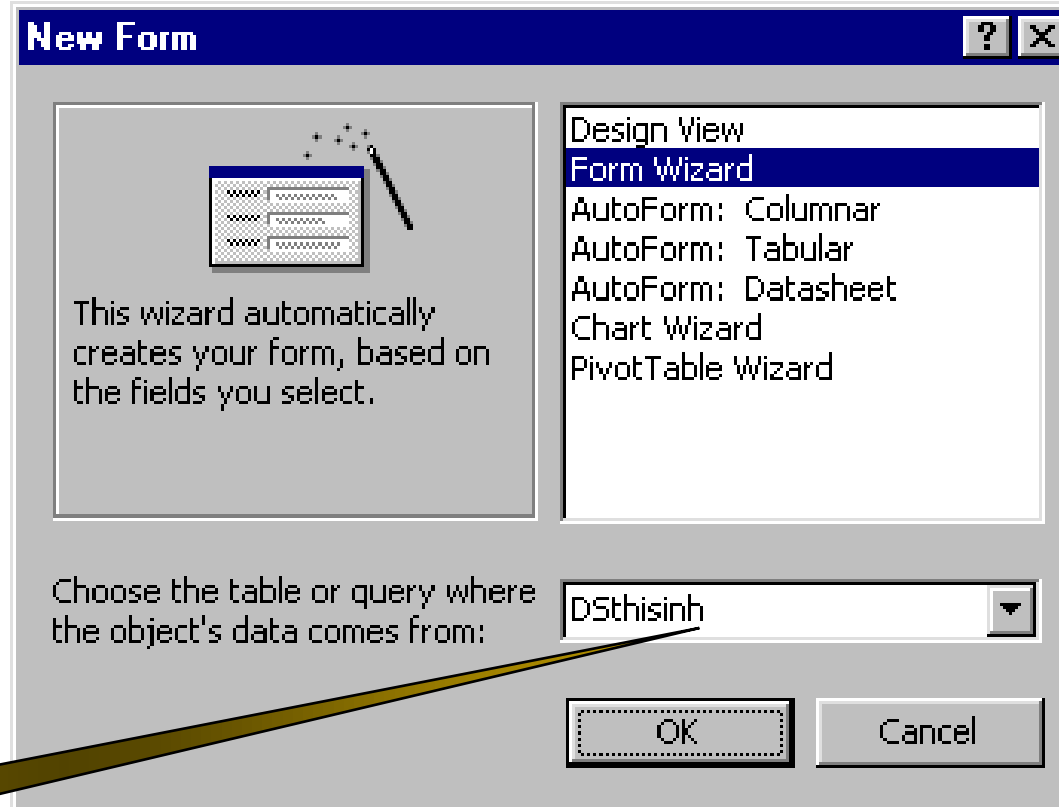
Có thể tạo bằng nhiều cách khác nhau như

- Form Wizard: tạo theo hướng dẫn từng bước của Access
- AutoForm : tự động tạo theo các kiểu cho sẵn (Columnar / Tabular / Datasheet)
- Design View: NSD tự thiết kế theo ý mình

## II. Tạo Form

### 1. Dùng Form Wizard: tạo theo hướng dẫn từng bước của Access

- Từ cửa sổ Database, chọn Forms / New, hộp thoại New Form xuất hiện
- Chọn Form Wizard
- Chọn nguồn dữ liệu cho Form (Table hoặc Query)
- Chọn OK sang bước tiếp theo, xuất hiện hộp thoại **1** a :



Chọn nguồn dữ liệu

## II. Tạo Form

### 1. Dùng Form Wizard:

Form Wizard

Which fields do you want on your form?  
You can choose from more than one table or

Tables/Queries  
Table: DSthisinh

Available Fields:

- Hoten
- DiemToan
- DiemLy
- DiemHoa

Selected Fields:

Cancel < Back

➤ Chọn các Field phù hợp (trong mục Available Fields), bấm nút > hoặc >> để đưa vào Selected Fields(1b)

Form Wizard

Which fields do you want on your form?  
You can choose from more than one t

Tables/Queries  
Table: DSthisinh

Available Fields:

Selected Fields:

- Hoten
- DiemToan
- DiemLy
- DiemHoa

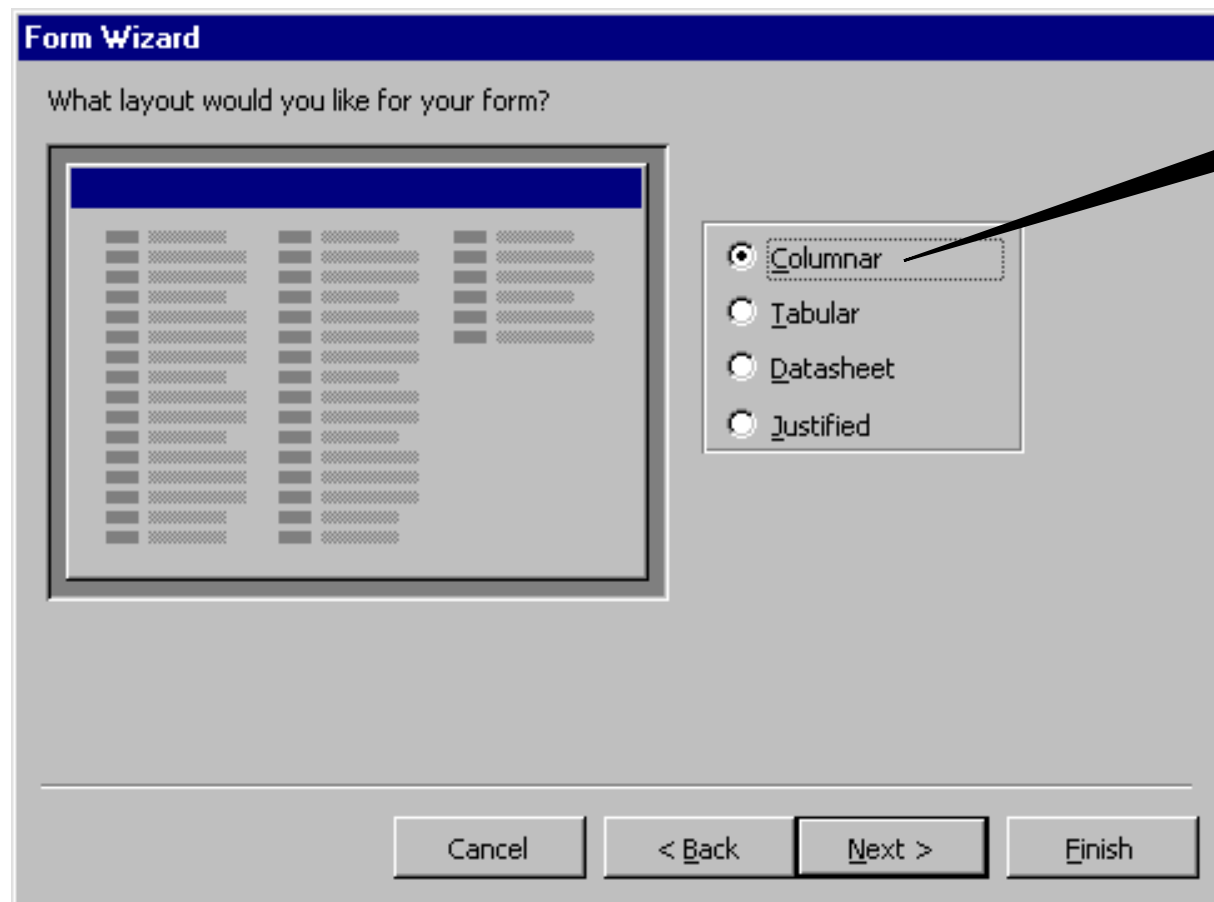
➤ Chọn Next sang bước tiếp theo, xuất hiện hộp thoại 2 :



## II. Tạo Form

### 1. Dùng Form Wizard:

- Chọn kiểu trình bày Form : cột, bảng...

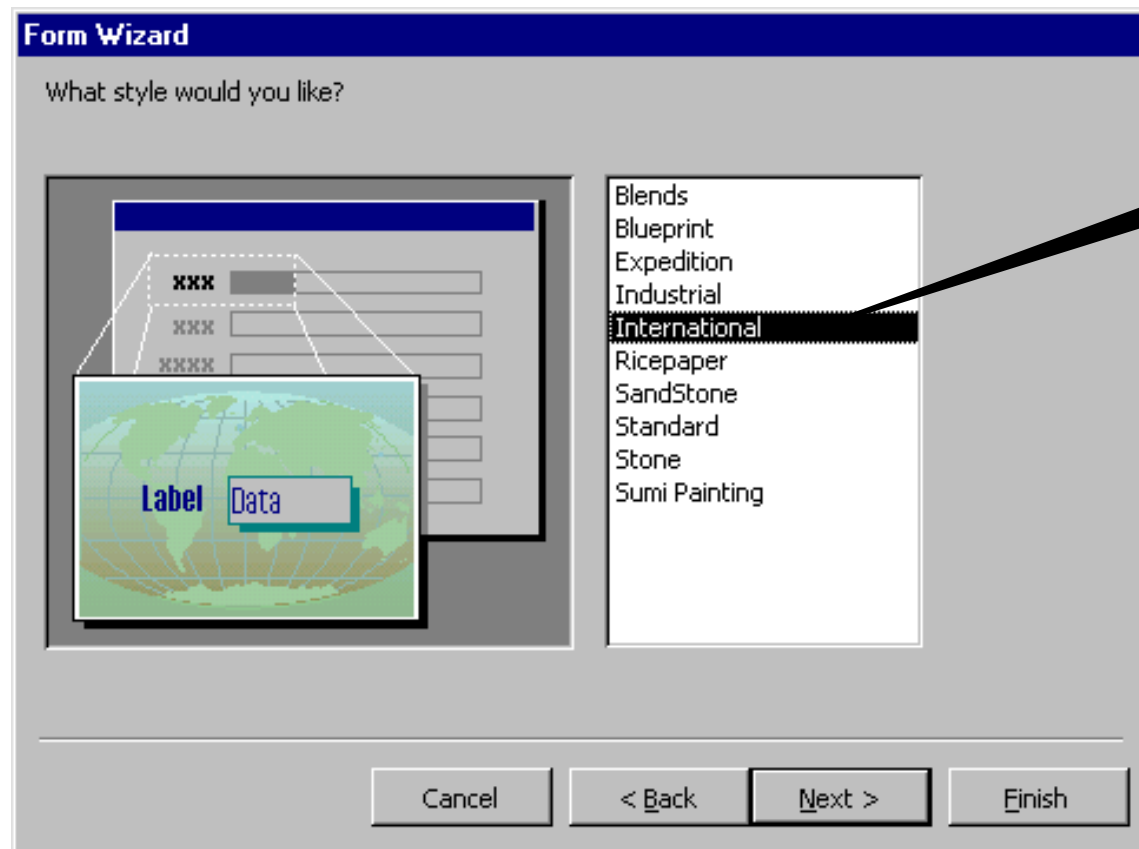


- Chọn Next sang bước tiếp theo, xuất hiện hộp thoại **3** : <sup>85</sup>

## II. Tạo Form

### 1. Dùng Form Wizard:

- Chọn một mẫu trong danh sách các mẫu cho sẵn để áp dụng cho Form



- Chọn Next sang bước tiếp theo, xuất hiện hộp thoại <sup>86</sup>4:

## II. Tạo Form

### 1. Dùng Form Wizard:

#### ➤ Đặt tên cho Form

4

Form Wizard

What title do you want for your form?

DSthisinh

That's all the information the wizard needs to create your form.

Do you want to open the form or modify the form's design?

Open the form to view or enter information.

Modify the form's design.

Display Help on working with the form?

Cancel < Back Next > Finish

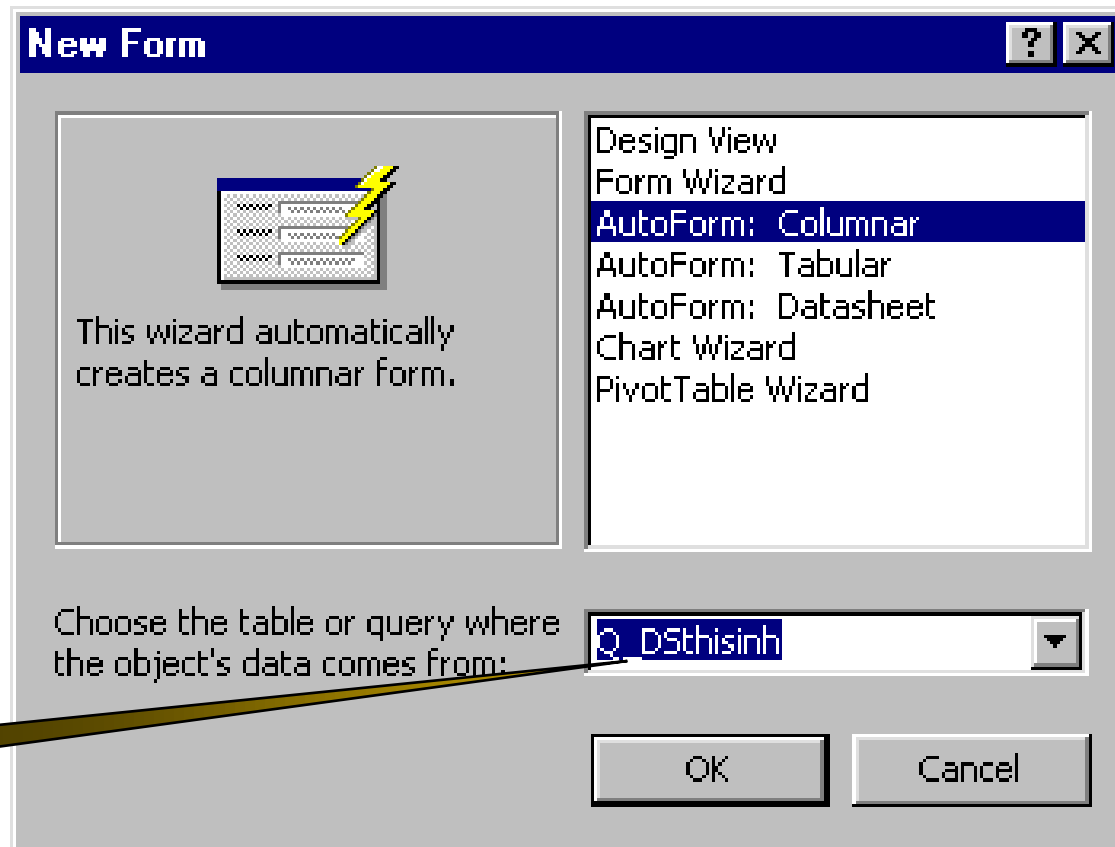
#### ➤ Chọn Finish để hoàn thành quá trình

## II. Tạo Form

### 2. Dùng Auto Form: tạo Form tự động với **tất cả các trường** trong Table hoặc Query

- Từ cửa sổ Database, chọn Forms / New, hộp thoại New Form xuất hiện
- Chọn AutoForm: Columnar
- Chọn nguồn dữ liệu cho Form (Table hoặc Query)
- Chọn OK kết quả sẽ xuất hiện

Chọn nguồn dữ liệu



## II. Tạo Form

### 2. Dùng Auto Form:

- Màn hình nhập liệu như sau

The screenshot shows a Microsoft Access form window titled "Q\_DSthisinh". The form contains the following fields and values:

Hoten	Pham Viet Cam
DiemToan	10
DiemLy	9
DiemHoa	10
DiemTC	29
Xeploai	Gioi

At the bottom of the form is a record navigation bar with the following elements:

- Record: [Navigation icons]
- 1
- [Navigation icons]
- of 8

Về mẫu tin  
phía sau

Về mẫu tin  
cuối cùng

Thêm mẫu  
tin mới

Về mẫu tin đầu  
tiên

Về mẫu tin phía  
trước

Stt mẫu tin  
hiện tại

## II. Tạo Form

### 3. Tự thiết kế Form:

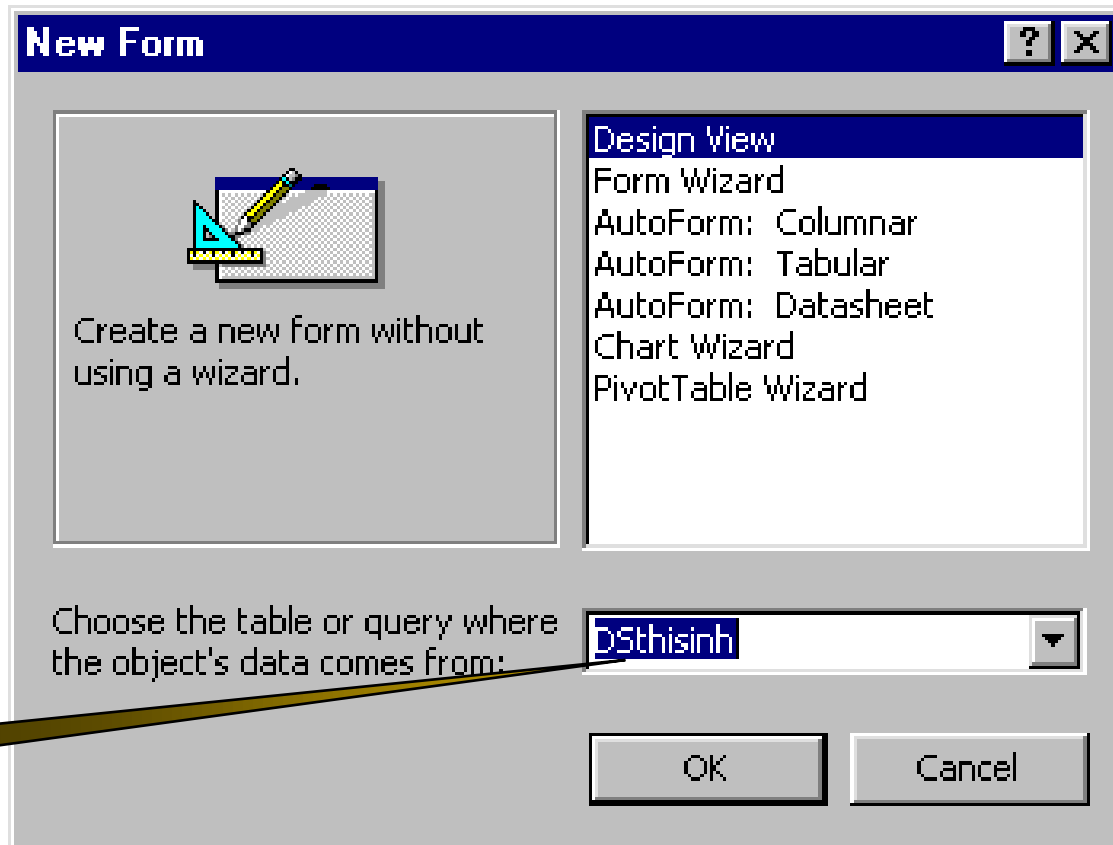
➤ Từ cửa sổ Database, chọn Forms / New, hộp thoại New Form xuất hiện

➤ Chọn Design View

➤ Chọn nguồn dữ liệu cho Form (Table hoặc Query)

➤ Chọn OK, xuất hiện cửa sổ như sau

Chọn nguồn dữ liệu

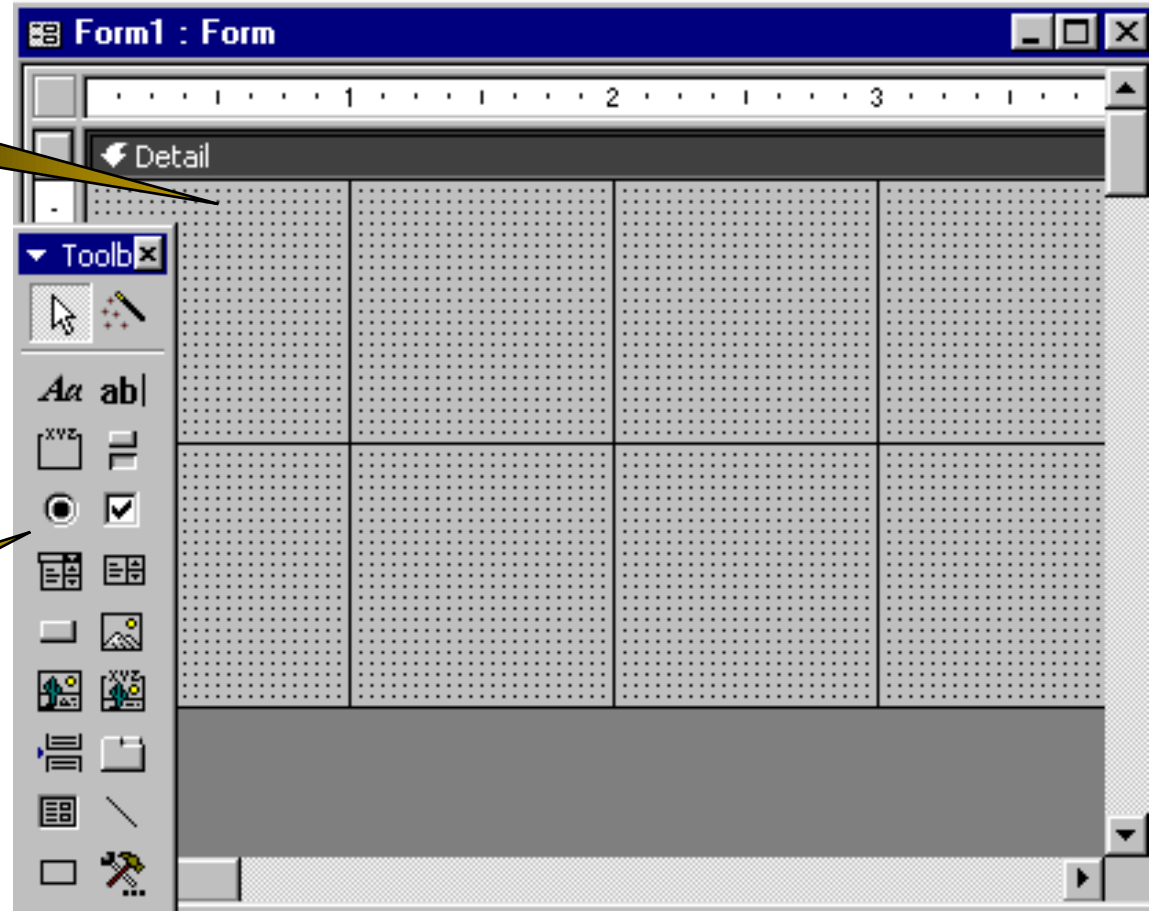


## II. Tạo Form

### 3. Tự thiết kế Form:



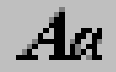






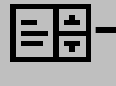










Màn hình thiết kế Form

Thanh công cụ  
Toolbox



Tên gọi và ý nghĩa của các nút chọn trên thanh công cụ Toolbox như sau:

▼ Toolb x

Select Object	←		→		Controls Wizard
Label	←		→		Text Box
Option Group	←		→		Toggle Button
Option Button	←		→		Check Box
Combo Box	←		→		List Box
Command Button	←		→		Image
Unbound Object Frame	←		→		Bound Object Frame
Page Break	←		→		Tab Control
Subform/Subreport	←		→		Line
Rectangle	←		→		More Controls



<b>Select Object</b>	chọn đối tượng cần hiệu chỉnh
<b>Label</b>	tạo nhãn
<b>Option Group</b>	nhóm chọn việc
<b>Option Button</b>	tạo nút chọn một trong nhiều giá trị
<b>Combo Box</b>	chọn giá trị từ danh sách
<b>Command Button</b>	tạo nút lệnh
<b>Unbound Object Frame</b>	tạo một khung hình cố định
<b>Page Break</b>	tạo dấu phân trang
<b>Subform/Subreport</b>	tạo Form con, tạo Report con
<b>Rectangle</b>	tạo hình chữ nhật

<b>Controls Wizard</b>	bật/tắt Controls Wizard
<b>Text Box</b>	hiển thị giá trị/nhập dữ liệu cho trường, kết quả biểu thức
<b>Toggle Button</b>	tạo nút bật tắt
<b>Check Box</b>	hộp đánh dấu để chọn nhiều giá trị cùng lúc
<b>List Box</b>	hộp danh sách, cho chọn 1 giá trị trong danh sách
<b>Image</b>	đối tượng hình ảnh
<b>Bound Object Frame</b>	tạo khung hình không cố định
<b>Tab Control</b>	tạo Tab để điều khiển việc chọn trang
<b>Line</b>	vẽ đường thẳng
<b>More Controls</b>	chọn sử dụng các nút điều khiển từ nhiều chương trình khác

## II. Tạo Form

### 3. Tự thiết kế Form:

- Muốn đưa một công cụ vào trong Form, ta có thể tự thiết kế hoặc sử dụng Control Wizard của Access
  - Dùng Control Wizard
    - Bấm nút Control Wizard (sẽ có màu sáng)
    - Bấm vào công cụ cần chọn để đưa vào Form
  - Tro chuột kéo Field cần chọn vào trong Form tại vị trí thích hợp
    - Khai báo các thông tin cần thiết theo chỉ dẫn của Access
  - Tự thiết kế
    - Bấm vào công cụ cần chọn để đưa vào Form
    - Tro chuột kéo Field cần chọn vào trong Form tại vị trí thích hợp
    - Khai báo các thông tin cần thiết. Nếu muốn sửa đổi các thuộc tính thì nhấp đúp tại công cụ vừa tạo để khai báo lại

## II. Tạo Form

### 4. Hiệu chỉnh Form:

- Trong cửa sổ Database, chọn tên Form cần hiệu chỉnh, chọn nút Design
- Tự sửa đổi như khi thiết kế

### 5. Thực hiện Form

Trong cửa sổ Database, chọn tên Form cần thực hiện, chọn nút Open

### III. Các loại điều khiển

#### 1. Các loại điều khiển

Khi thiết kế Form, thường dùng các loại điều khiển sau:

- Hộp văn bản (Textbox)
- Nhãn (Label)
- Hộp lựa chọn (Combo Box)
- Hộp danh sách (List Box)
- Nút lệnh (Command Button)
- Nhóm lựa chọn (Option Group)

### III. Các loại điều khiển

#### 2. Cách dùng

Có thể dùng các ô điều khiển theo ba cách sau:

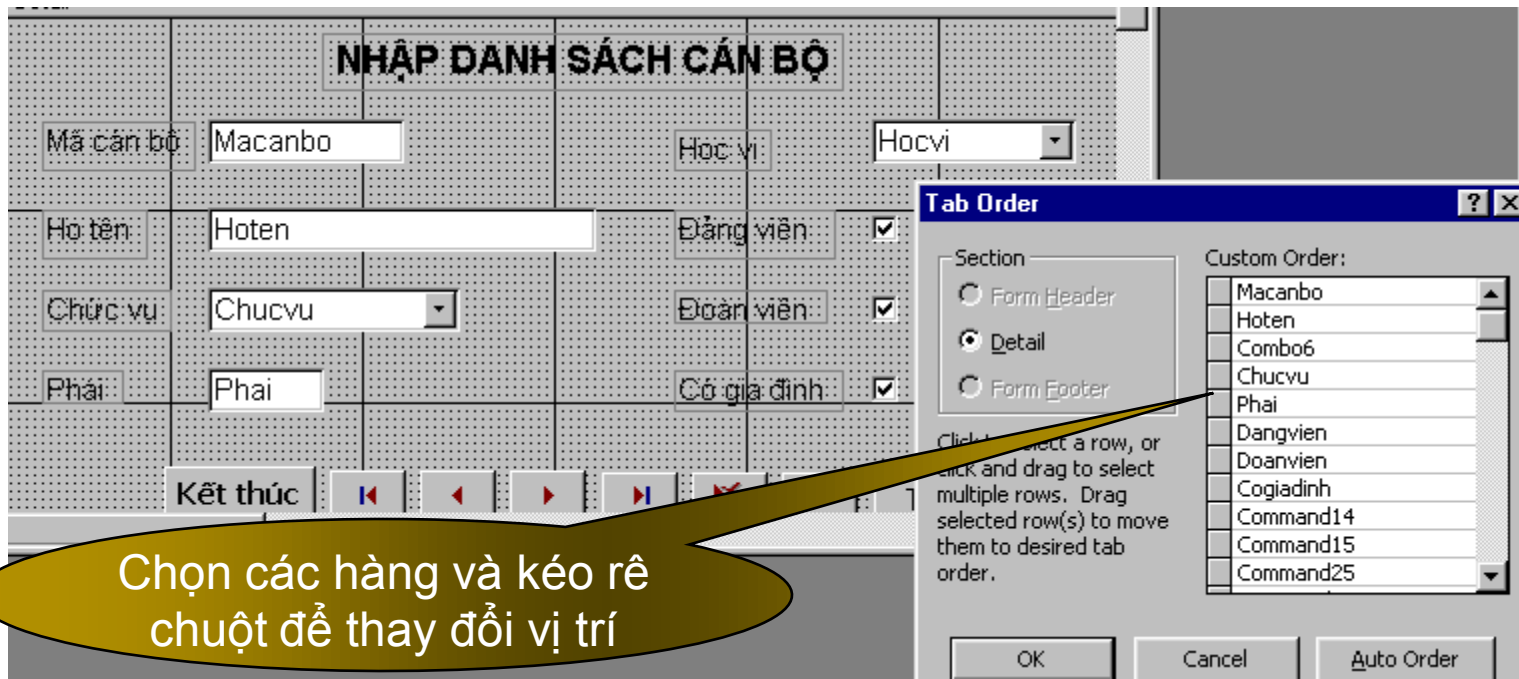
- Bound (buộc với một trường nào đó): nguồn dữ liệu của nó lấy từ Table hay Query. Textbox là loại thông dụng nhất
- Unbound (không buộc vào bất kỳ trường nào): không mang dữ liệu của bất kỳ nguồn nào. Loại này dùng để trình bày. Label là loại thông dụng nhất
- Calculated (tính toán): nguồn dữ liệu là một biểu thức. Giá trị hiển thị ở chế độ Form view là kết quả của biểu thức. Trong biểu thức có thể dùng kết hợp các phép toán (+, -, \*, /, =) với các tên trường. Hàm trả về giá trị hoặc các số. Ví dụ: **Giamoi:0.75\*[Dongia]**

### III. Các loại điều khiển

#### 3. Thay đổi trình tự thực hiện các điều khiển

Khi chạy Form, nếu dùng phím Tab hoặc Shift+Tab thì các điều khiển sẽ được chọn theo trình tự thiết kế. Ta có thể thay đổi bằng cách:

- Mở Form ở chế độ Design.
- Chọn lệnh View / Tab Order, xuất hiện hộp thoại



## IV. Nút lệnh Command Wizard


Để thuận tiện cho người dùng chương trình, khi thiết kế Form phải tạo các nút lệnh cần thiết. Thường dùng Command Control Wizard cho tiện thiết kế

### 1. Cách tạo

➤ Mở Form ở chế độ Design.

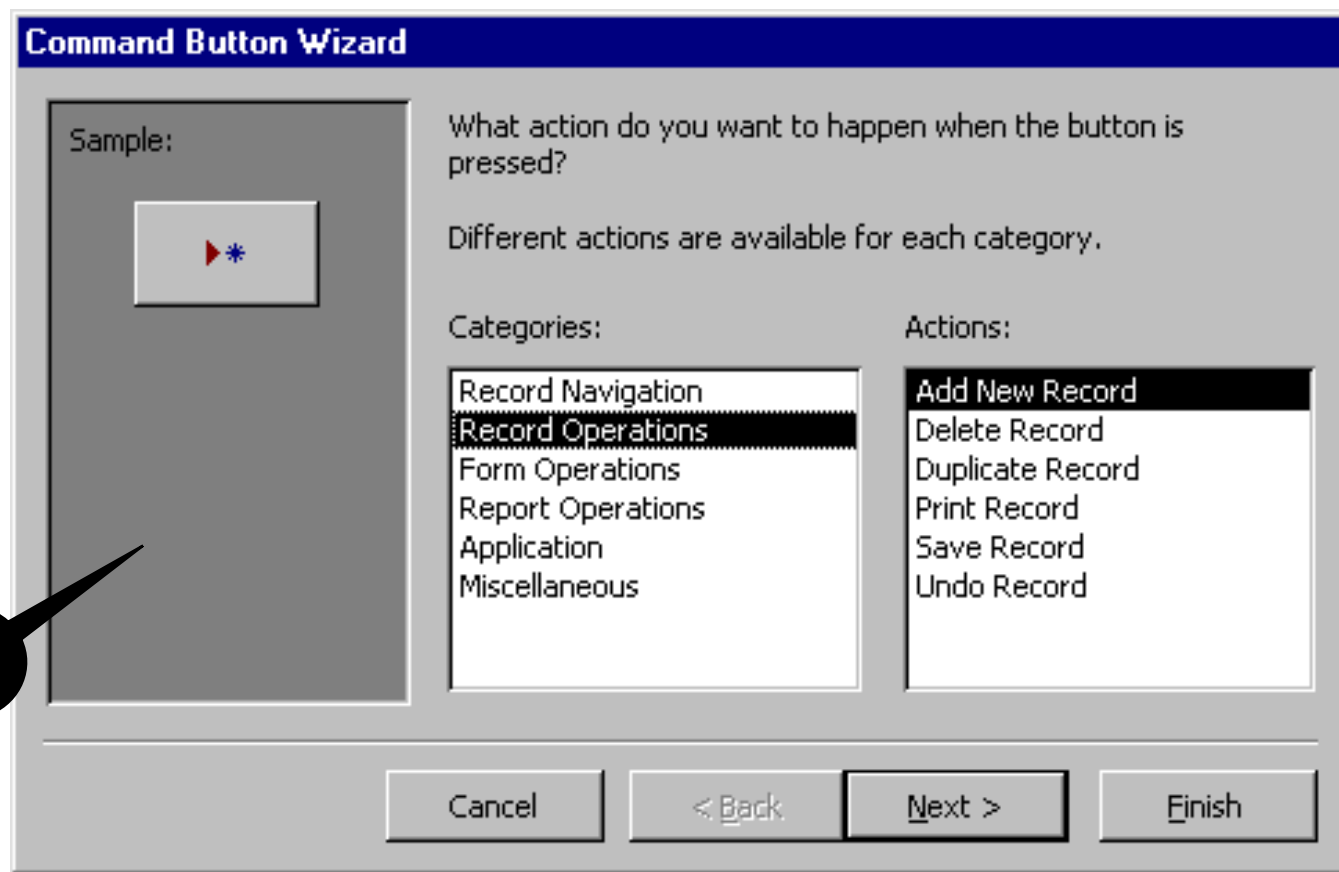
➤ Mở thanh Toolbox và kích hoạt nút Control Wizard



➤ Nhấp chọn biểu tượng Command Button  và kéo chuột trên Form để chỉ định vị trí đặt nút lệnh, xuất hiện hộp đối thoại Command Button Wizard ❶ như sau:

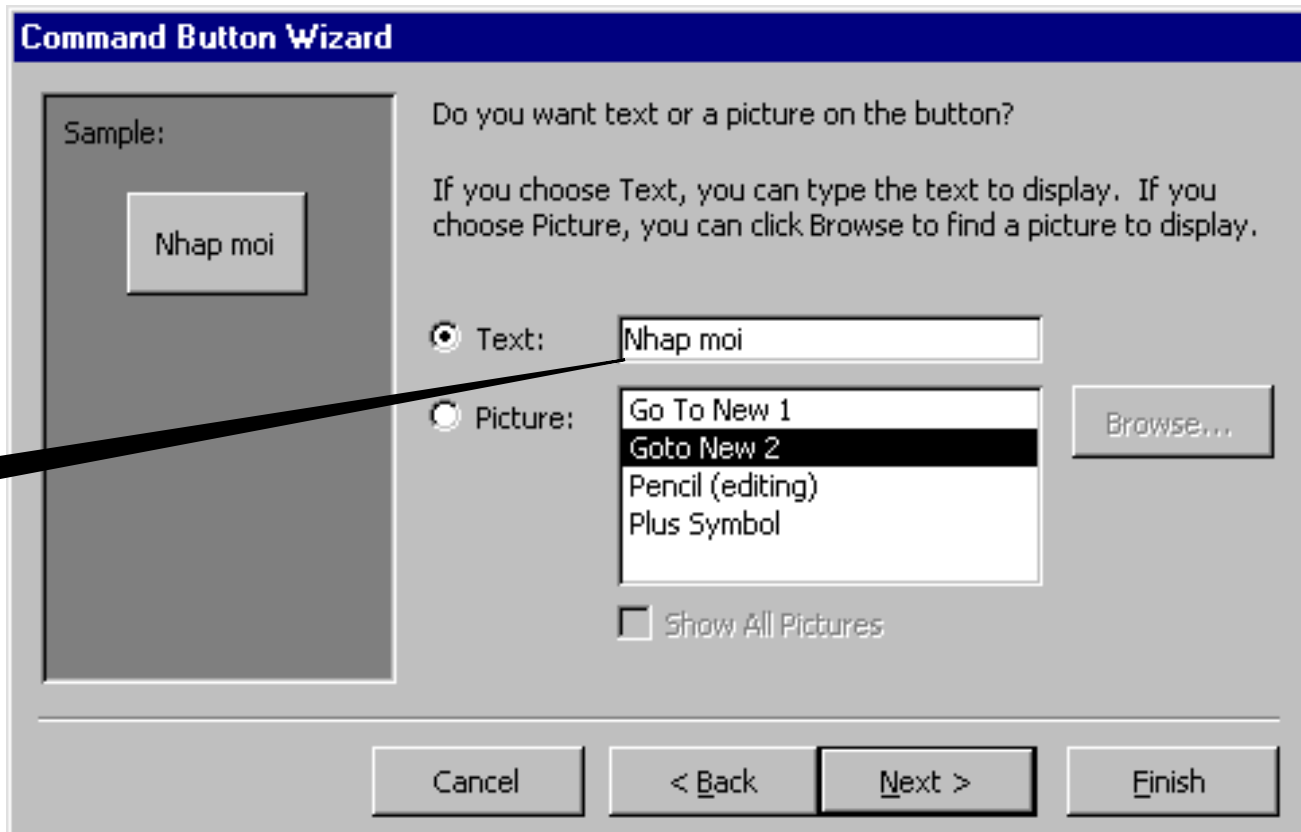


## IV. Nút lệnh Command Wizard



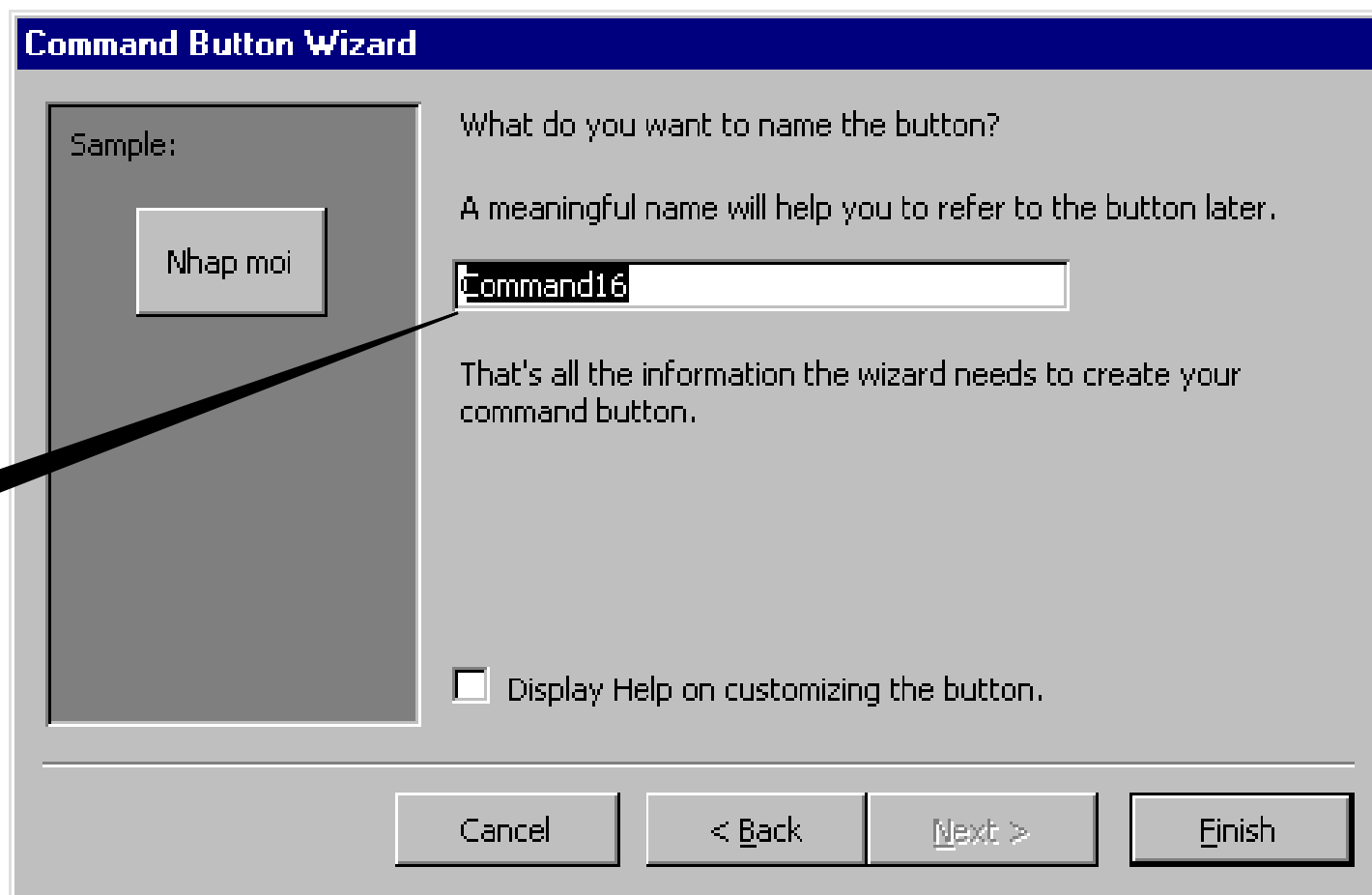
- Categories: chọn nhóm lệnh tùy theo đối tượng
- Actions: chọn lệnh cụ thể
- Chọn Next sang bước tiếp theo, xuất hiện hộp thoại<sup>101</sup> ②

## IV. Nút lệnh Command Wizard



- Text: chọn và nhập tên cho nút lệnh (nếu muốn)
- Picture: chọn hình ảnh cho nút lệnh (không nhập tên)
- Chọn Next sang bước tiếp theo, xuất hiện hộp thoại<sup>102</sup> ③

## IV. Nút lệnh Command Wizard



- Chọn tên cho nút (có thể để mặc nhiên)
- Chọn Finish để hoàn thành

## IV. Nút lệnh Command Wizard

### 2. Danh sách các Category và Action

#### ➤ Record Navigation (điều khiển vị trí mẫu tin)

<b>Find Next</b>	Tìm đến mẫu tin gần nhất
<b>Find Record</b>	Mở hộp thoại tìm mẫu tin
<b>Go to First Record</b>	Đến mẫu tin đầu tiên
<b>Go to Last Record</b>	Về mẫu tin cuối cùng
<b>Go to Next Record</b>	Đến mẫu tin tiếp theo
<b>Go to Previous Record</b>	Về mẫu tin phía trước

## IV. Nút lệnh Command Wizard

### 2. Danh sách các Category và Action

#### ➤ Record Operations (Thao tác với mẫu tin)

<b>Add New Record</b>	Thêm mẫu tin mới
<b>Delete Record</b>	Xoá mẫu tin đang chọn
<b>Duplicate Record</b>	Tạo mẫu tin mới trùng với mẫu tin đang chọn
<b>Print Record</b>	In mẫu tin hiện hành
<b>Save Record</b>	Lưu mẫu tin đang sửa
<b>Undo Record</b>	Phục hồi mẫu tin

## IV. Nút lệnh Command Wizard

### 2. Danh sách các Category và Action

#### ➤ Form Operations (Thao tác với Form)

<b>Apply Form Filter</b>	Áp dụng lọc mẫu tin với tiêu chuẩn vừa tạo
<b>Edit Form Filter</b>	Tạo tiêu chuẩn lọc
<b>Close Form</b>	Đóng Form
<b>Open Form</b>	Mở Form
<b>Print a Form</b>	In Form
<b>Print Current Form</b>	In Form hiện hành
<b>Refresh Form Data</b>	Làm mới lại dữ liệu trong Form

## IV. Nút lệnh Command Wizard

### 2. Danh sách các Category và Action

#### ➤ Report Operations (Thao tác với Report)

<b>Mail Report</b>	Gởi Email Report
<b>Preview Report</b>	Xem Report trước khi in
<b>Print Report</b>	In Report ra giấy
<b>Send Report to File</b>	Chuyển Report thành một tập tin

## IV. Nút lệnh Command Wizard

### 2. Danh sách các Category và Action

#### ➤ Application (Thao tác với ứng dụng)

<b>Quit Application</b>	Thoát ứng dụng
<b>Run Application</b>	Chạy ứng dụng

#### ➤ Miscellaneous (Thao tác với các đối tượng khác)

<b>Print Table</b>	In bảng
<b>Run Macro</b>	Chạy một Macro
<b>Run Query</b>	Thực hiện một Query



## Chương V : Report (Báo cáo-Báo biểu)

### I. Khái niệm

➤ Report là một công cụ thuận tiện để in dữ liệu. Nguồn dữ liệu của Report là các Table có sẵn hoặc số liệu được tổng hợp dưới dạng Query

### II. Các kiểu trình bày Report

Columnar Report (báo biểu dạng cột)

Tabular Report (báo biểu dạng bảng)



## II. Các kiểu trình bày Report

### 1. Columnar Report (báo biểu dạng cột)

➤ Là dạng thức mà các trường số liệu trong Report được trình bày từ trên xuống thành một cột.

#### Báo cáo bán hàng

SoHD	HD001
Mavattu	N002
Tenvattu	Ngoi Mong Rong
Donvitinh	Vien
Soluong	200
Dongia	20,000
Thanhtien	4,000,000.00
SoHD	HD002
Mavattu	T001
Tenvattu	Ton Xi Mang
Donvitinh	Tam
Soluong	100
Dongia	200,000
Thanhtien	20,000,000.00

## II. Các kiểu trình bày Report

### 2. Tabular Report (báo biểu dạng bảng)

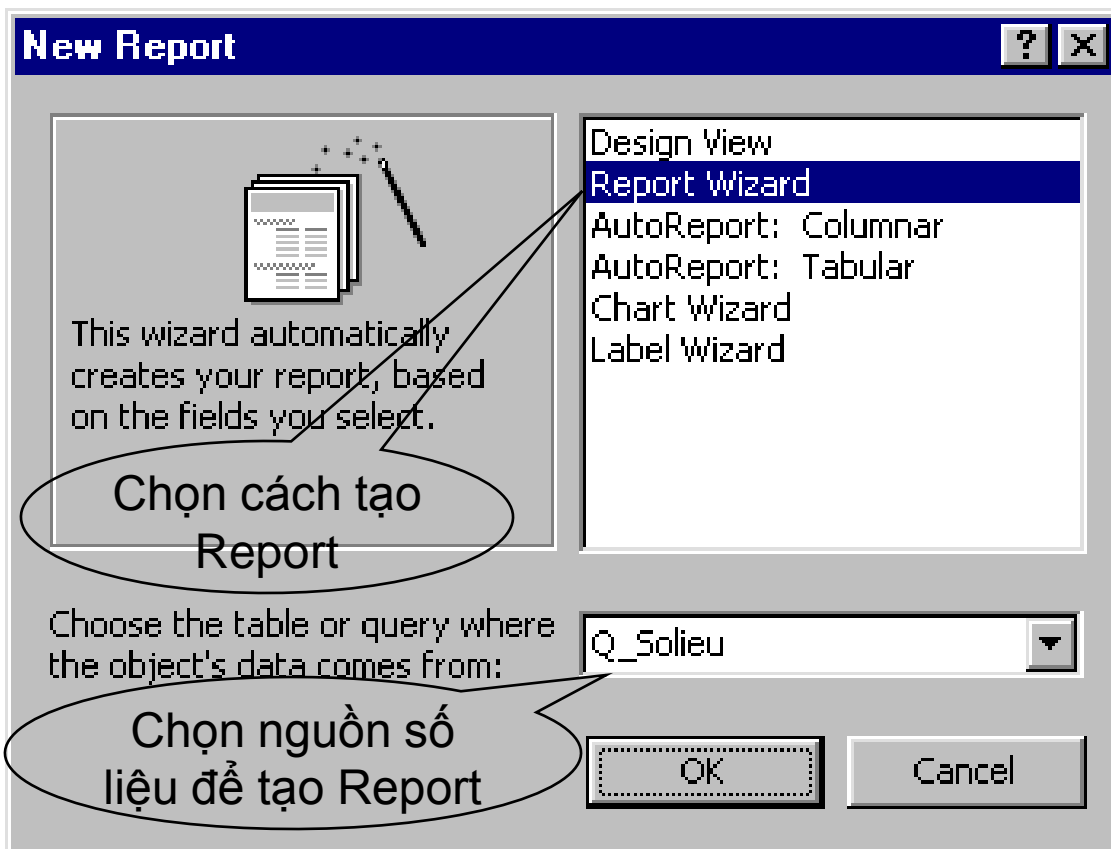
- Mỗi Record thể hiện trên một dòng, các trường được xếp thành cột. Loại này thích hợp cho bảng kê số liệu

#### ***BÁO CÁO BÁN HÀNG***

<i>STT</i>	<i>So HD</i>	<i>Ma vat tu</i>	<i>Ten vat tu</i>	<i>DV Tinh</i>	<i>SLG</i>	<i>Don gia</i>	<i>Thanh tien</i>
1	HD001	N002	Ngoi Mong Rong	Vien	200	20,000	4,000,000
2	HD002	T001	Ton Xi Mang	Tam	100	200,000	20,000,000
3	HD003	N002	Ngoi Mong Rong	Vien	450	15,000	6,750,000
4	HD004	T003	Ton Luon Song	Tam	200	10,000	2,000,000
5	HD005	T001	Ton Xi Mang	Tam	150	25,000	3,750,000
6	HD006	X001	Xi mang Hoang Thach	Tan	400	250,000	100,000,000
7	HD007	TH04	Thep Hoa Khanh	Kg	180	100,000	18,000,000
8	HD008	TH03	Thep Viet Uc	Kg	560	300,000	168,000,000
9	HD009	T003	Ton Luon Song	Tam	480	350,000	168,000,000
10	HD010	X001	Xi mang Hoang Thach	Tan	500	250,000	125,000,000

### III. Cách tạo Report

- Tại cửa sổ Database, chọn Report / New, xuất hiện cửa sổ New Report
- Chọn một trong các cách tạo Report, để đơn giản, ta chọn **Report Wizard**.



- Chọn nguồn số liệu để tạo Report (Table hay Query tùy theo yêu cầu cụ thể)

- Chọn OK, xuất hiện hộp thoại Report Wizard (1a) như sau

### III. Cách tạo Report

- Available Fields: chứa các trường có sẵn trong nguồn

Report Wizard

Which fields do you want on your report?  
You can choose from more than one table or query.

Tables/Queries  
Query: Q\_Solieu

Available Fields:

SoHD	>	
Mavattu	>>	
Tenvattu	>>	
Donv tinh	>	
Soluong	>>	
Dongia	>	
Thanh tien	>>	

Selected

Report Wizard

Which fields do you want on your report?  
You can choose from more than one table or query.

Tables/Queries  
Query: Q\_Solieu

Available Fields:

Selected Fields:

SoHD	>	
Mavattu	>>	
Tenvattu	>>	
Donv tinh	>	
Soluong	>>	
Dongia	>	
Thanh tien	>>	

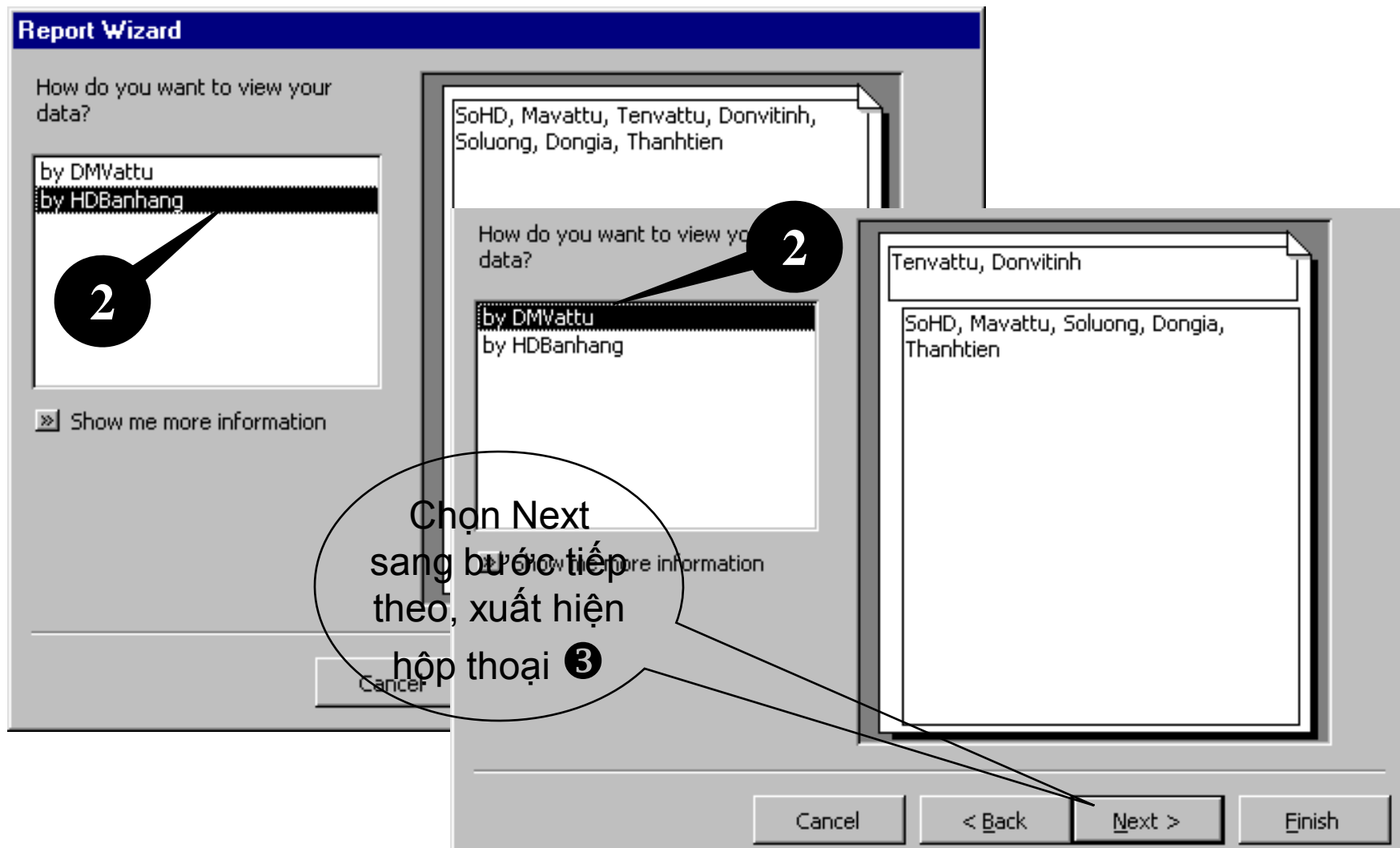
Chọn Next sang bước tiếp theo, xuất hiện hộp thoại 2

Cancel < Back Next >

- Chọn các trường phù hợp và đưa sang mục Selected Fields (1b)

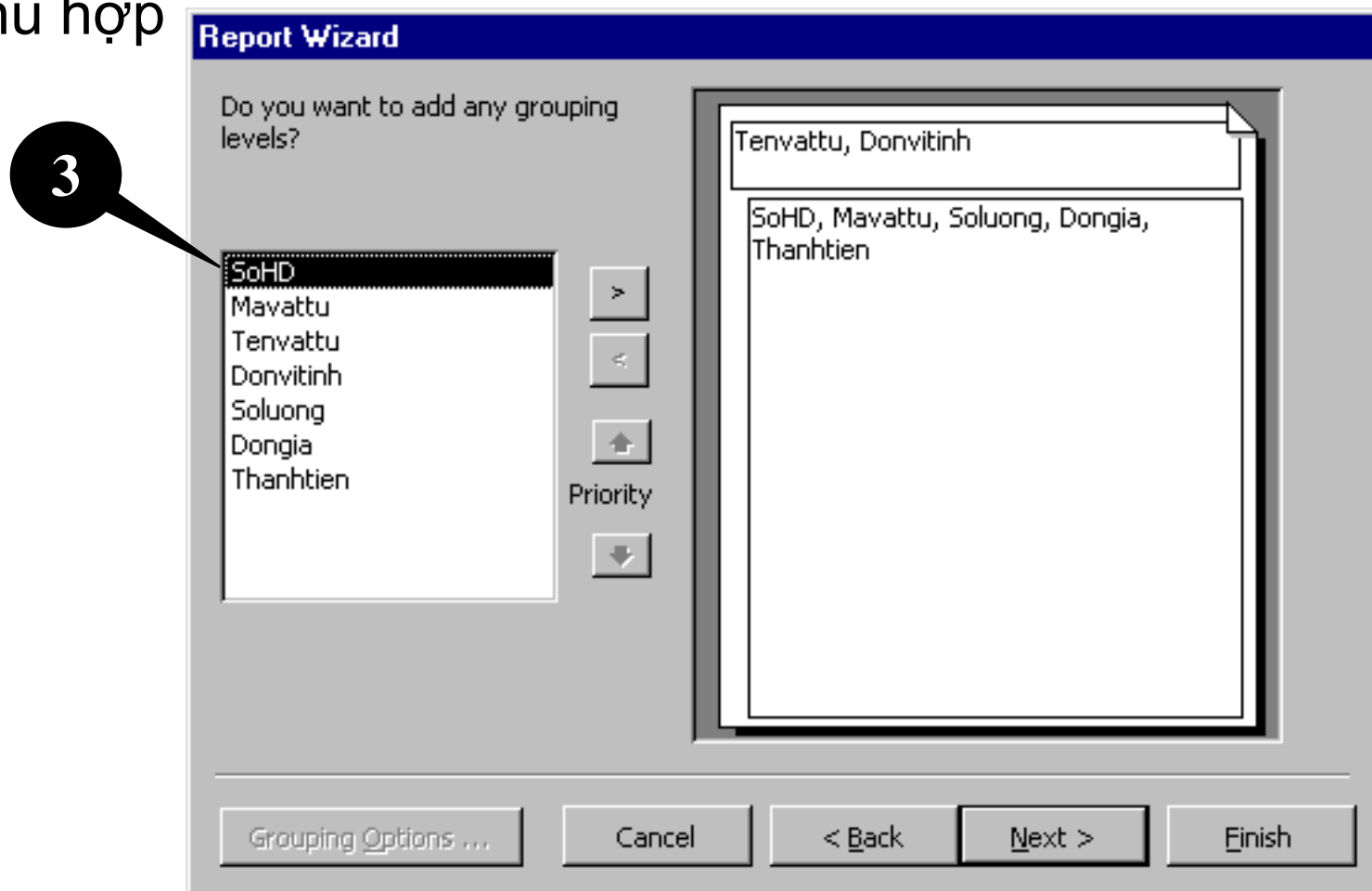
### III. Cách tạo Report

- Hộp thoại này hỏi bạn muốn hiển thị dữ liệu từ đâu, bạn tùy chọn cho phù hợp



### III. Cách tạo Report

- Cho chọn trường cần kết nhóm (nếu muốn), bạn tùy chọn cho phù hợp



- Chọn Next sang bước tiếp theo, xuất hiện hộp thoại<sup>15</sup> ④

### III. Cách tạo Report

- Cho chọn các trường mà danh sách sẽ sắp xếp theo thứ tự ưu tiên, số khoá sắp xếp tối đa là bốn.

4

Report Wizard

What sort order and summary information do you want for detail records?

You can sort records by up to four fields, in either ascending or descending order.

1 Mavattu [Sort Order: A-Z]

2 [Sort Order: A-Z]

3 [Sort Order: A-Z]

4 [Sort Order: A-Z]

Summary Options ...

Cancel < Back Next > Finish

Sắp xếp theo thứ tự tăng hoặc giảm dần

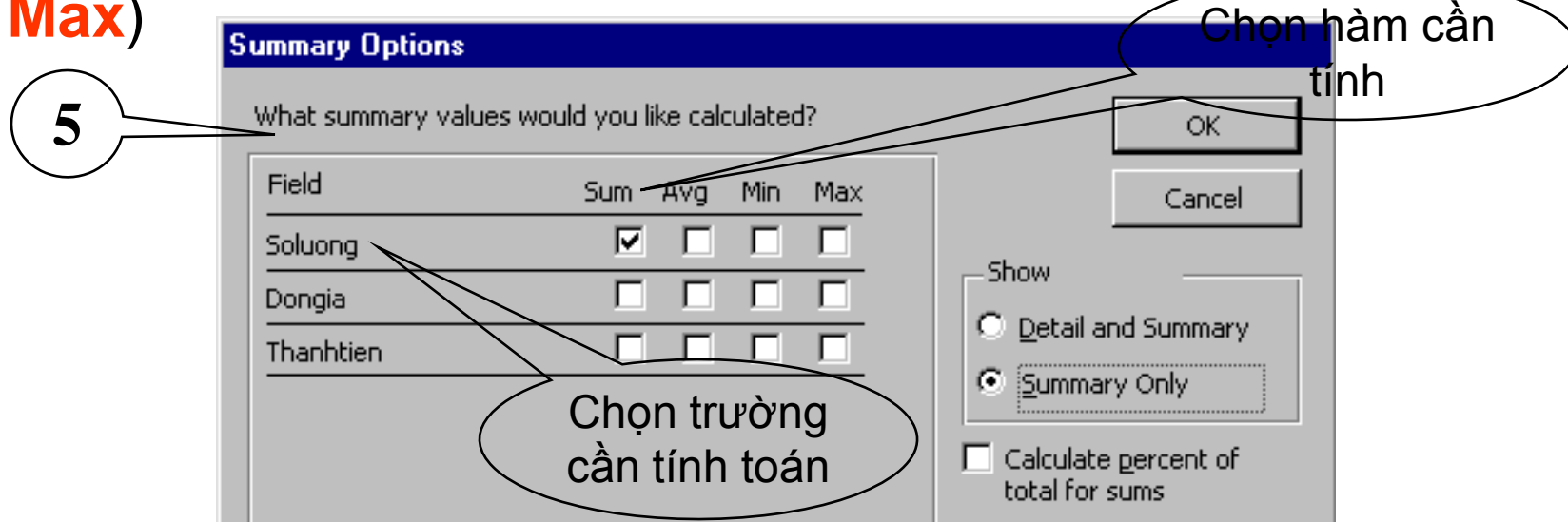
Sắp xếp theo bốn khoá tùy ý

- Chọn Summary Options nếu muốn tính toán, xuất hiện hộp thoại 5



### III. Cách tạo Report

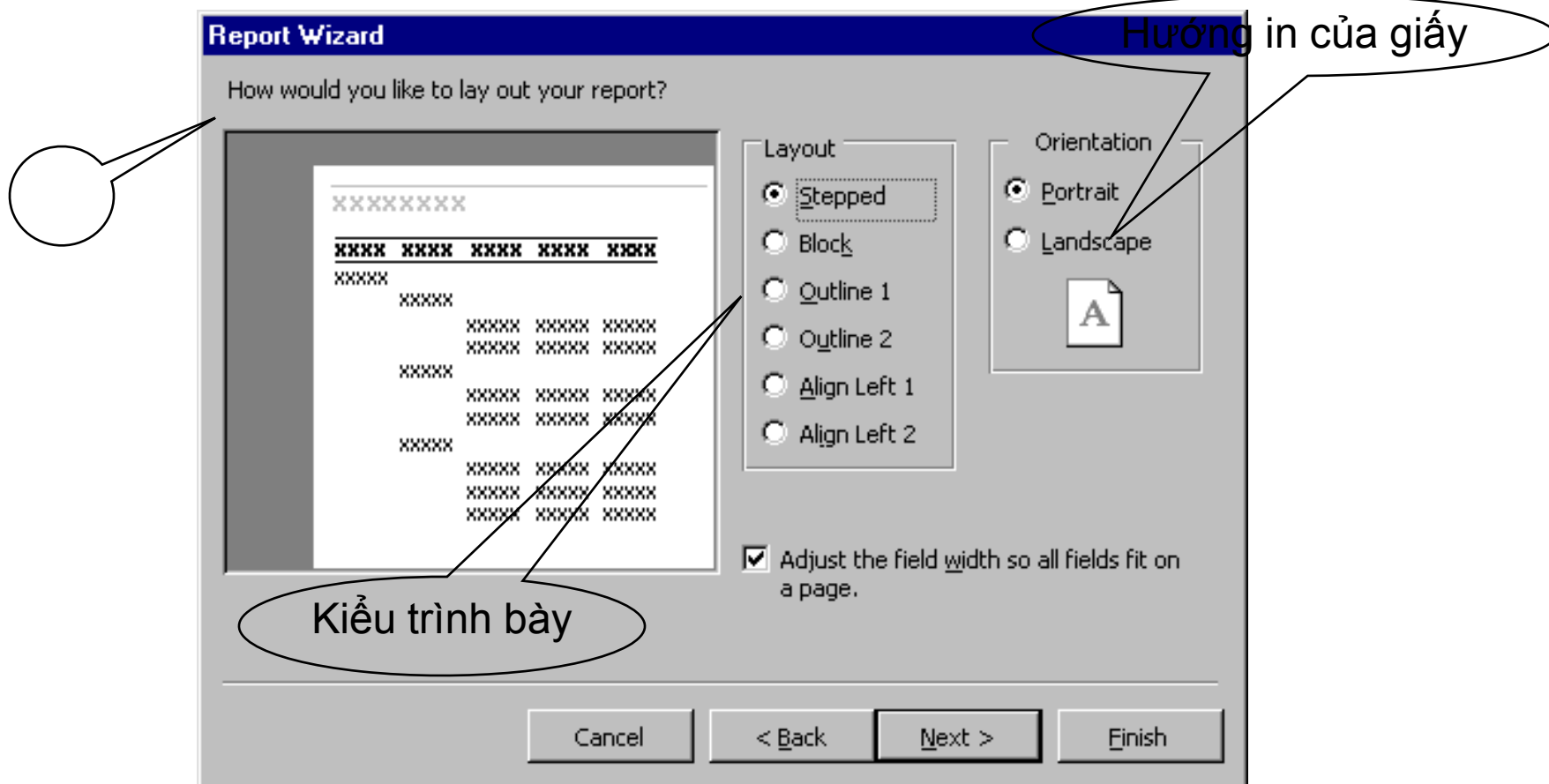
- Chọn các trường cần tính toán, kiểu tính (**Sum, Avg, Min, Max**)



- Show: chọn cách hiển thị: chi tiết và tổng (Detail and Summary) hay chỉ hiện tổng (Summary Only)
- Calculate percent of total for sums: chọn nếu muốn tính % trên tổng số
- Chọn OK, để quay về hộp thoại ④. Tại hộp thoại ④, chọn Next sang bước tiếp theo, xuất hiện hộp thoại ⑥

### III. Cách tạo Report

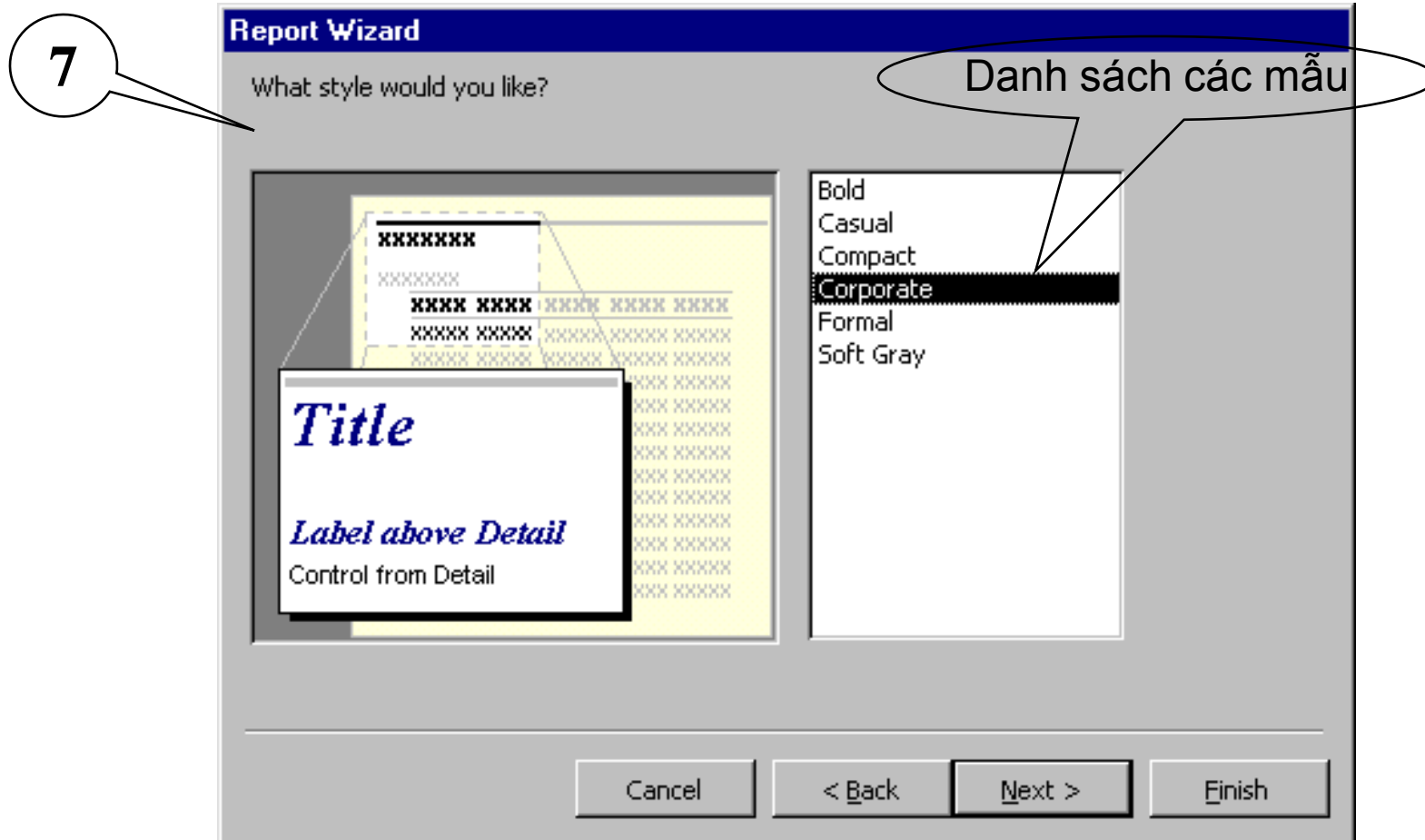
- Layout: chọn kiểu trình bày
- Orientation: chọn hướng in của trang giấy



- Chọn Next sang bước tiếp theo, xuất hiện hộp thoại <sup>118</sup> 7

### III. Cách tạo Report

- Chọn một trong các mẫu trình bày



- Chọn Next sang bước tiếp theo, xuất hiện hộp thoại ⑧

### III. Cách tạo Report

#### ➤ Đặt tên cho Report



➤ Chọn Finish để hoàn thành, kết quả sẽ xuất hiện

➤ Có thể mở cửa sổ Design để trình bày thêm nếu muốn

**Mở đầu****Table****Query****Form****Report****Macro**

### III. Cách tạo Report

Thêm Số thứ tự

#### ***BÁO CÁO BÁN HÀNG***

<i><b>STT</b></i>	<i><b>So HD</b></i>	<i><b>Ma vat tu</b></i>	<i><b>Ten vat tu</b></i>	<i><b>DV Tinh</b></i>	<i><b>SLG</b></i>	<i><b>Don gia</b></i>	<i><b>Thanh tien</b></i>
1	HD001	N002	Ngoi Mong Rong	Vien	200	20,000	4,000,000
2	HD002	T001	Ton Xi Mang	Tam	100	200,000	20,000,000
3	HD003	N002	Ngoi Mong Rong	Vien	450	15,000	6,750,000
4	HD004	T003	Ton Luon Song	Tam	200	10,000	2,000,000
5	HD005	T001	Ton Xi Mang	Tam	150	25,000	3,750,000
6	HD006	X001	Xi mang Hoang Thach	Tan	400	250,000	100,000,000
7	HD007	TH04	Thep Hoa Khanh	Kg	180	100,000	18,000,000
8	HD008	TH03	Thep Viet Uc	Kg	560	300,000	168,000,000
9	HD009	T003	Ton Luon Song	Tam	480	350,000	168,000,000
10	HD010	X001	Xi mang Hoang Thach	Tan	500	250,000	125,000,000
<i><b>Tong cong:</b></i>							61,500,000
<i><b>Trung binh:</b></i>							61,550,000
<i><b>Cao nhất:</b></i>							168,000,000
<i><b>Thap nhất:</b></i>							2,000,000

Kẻ khung

Các tính toán cần thiết

## Chương VI : Macro - Menu

### I. Khái niệm Macro

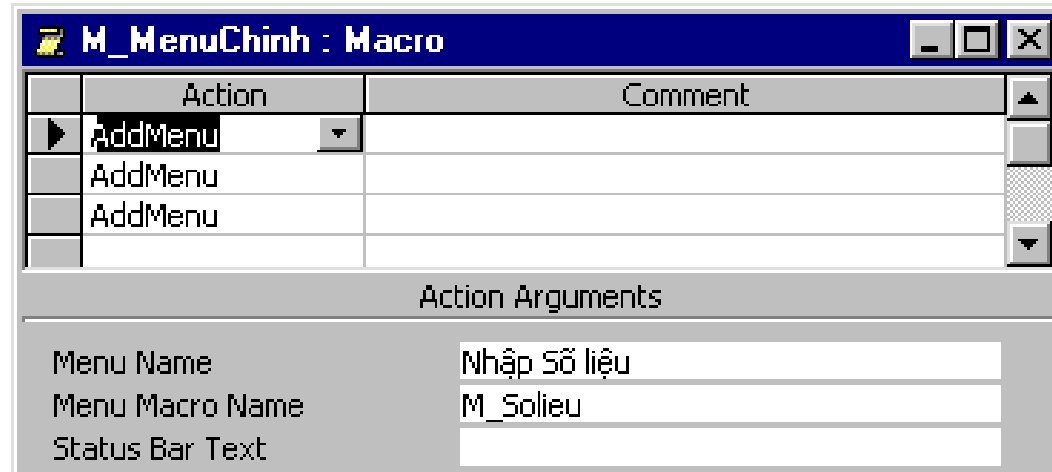
- Macro là một hay một tập hợp các hành động (Action) liên tiếp được định nghĩa và lưu trữ với một tên xác định. Macro cho phép tự động hoá các công việc cần thực hiện
- Có 3 loại Macro
  - Macro kết hợp nhiều hành động: được kết hợp bởi nhiều hành động liên tiếp nhau. Khi Macro được gọi, các hành động sẽ lần lượt tự động thực hiện
  - Macro Group: là tập hợp các Macro có tính năng giống nhau. Để thi hành một Macro trong Macro Group ta chỉ tên nó như sau: **Tên Macro Group. Tên Macro thực hiện**
  - Macro theo điều kiện: là Macro mà các hành động chỉ được thi hành khi thoả mãn điều kiện nào đó



## II. Cách tạo Macro

➤ Trong cửa sổ Database: chọn Macro / New, xuất hiện cửa sổ khai báo Macro như sau

➤ Action: chọn hành động cần thực hiện. Có thể chọn nhiều hành động tương ứng với nhiều dòng



➤ Comment: ghi chú thích cho hành động. Không bắt buộc nhưng giúp NSD dễ dàng khi bảo trì hệ thống vì hiểu được mục đích thiết kế.

➤ Action Arguments: chỉ định các đối số phù hợp cho Action khi cần thiết

## II. Cách tạo Macro

- Lưu ý: Cửa sổ thiết kế Macro chỉ gồm 2 cột. Nếu chọn lệnh View / Macro Names sẽ thêm cột Macro Name phía trước như sau

Cột Macro Name

Macro Name	Action	Comment
M_Timkiem	AddMenu	
Xem Danh sách Cán bộ	OpenReport	

Action Arguments	
Menu Name	Tìm kiếm
Menu Macro Name	M_Timkiem
Status Bar Text	

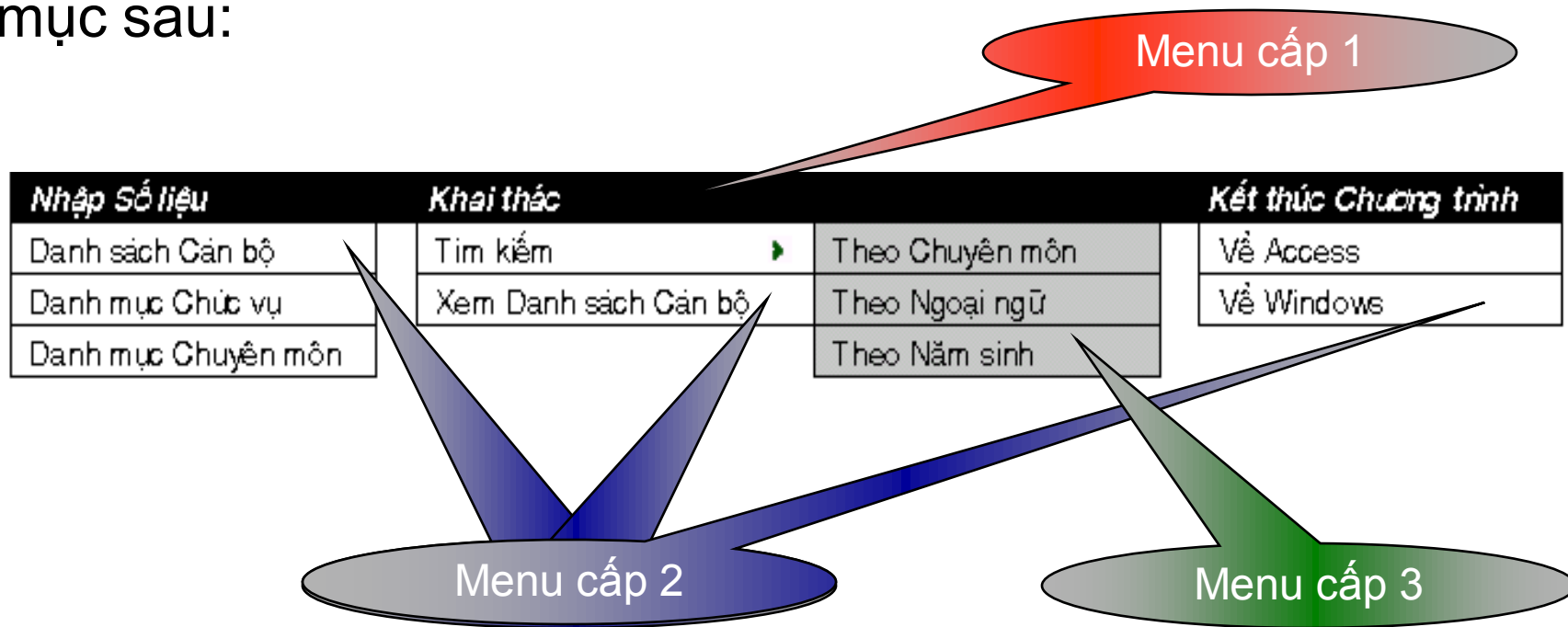
## III. Thực hiện Macro

- Trong cửa sổ Database, chọn tên Macro, chọn Open
- Macro có thể được gọi khi sử dụng Form, Report



## IV. Tạo Menu

- Ta có thể sử dụng Macro để xây dựng hệ thống Menu cho phép lựa chọn công việc dễ dàng. Thông qua Menu, các đối tượng trong Database được liên kết thành một khối thống nhất, thuận tiện cho người sử dụng
- Sau đây trình bày cách tạo một hệ thống Menu gồm các mục sau:



## IV. Tạo Menu

### ➤ Bước 1: Tạo Menu cấp 1

- Trong cửa sổ Database, chọn Macro / New, xuất hiện cửa sổ Macro, khai báo các thông tin như sau:

<i>Action</i>	<i>Menu Name</i>	<i>Menu Macro Name</i>
Addmenu	<b>Nhập Số liệu</b>	M_Solieu
Addmenu	<b>Khai thác</b>	M_Khaithac
Addmenu	<b>Kết thúc Chương trình</b>	M_Ketthuc

Action	Comment
AddMenu	
AddMenu	
AddMenu	

Action Arguments

Menu Name: Nhập Số liệu

Menu Macro Name: M\_Solieu

Status Bar Text:

Menu cấp 1

<b>Nhập Số liệu</b>	<b>Khai thác</b>	<b>Kết thúc Chương trình</b>
Danh sách Cán bộ	Tim kiếm	Về Access
Danh mục Chức vụ	Xem Danh sách Cán bộ	Về Windows
Danh mục Chuyên môn		

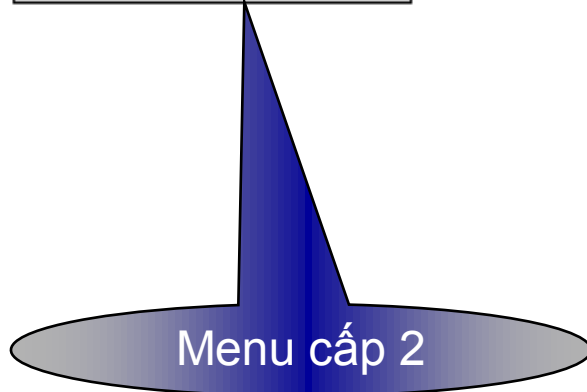
## IV. Tạo Menu

### ➤ Bước 2: Tạo Menu cấp 2

- Chọn Macro / New, chọn View / Macro Names để thêm cột Macro Name, khai báo các thông tin như sau:

<i>Macro Name</i>	<i>Action</i>	<i>Table Name</i>
Danh sách Cán bộ	Open Table	DSCanbo
Danh mục Chức vụ	Open Table	DMChucvu
Danh mục Chuyên môn	Open Table	DMChuyenmon

<i>Nhập Số liệu</i>	<i>Khai thác</i>	<i>Kết thúc Chương trình</i>
Danh sách Cán bộ	Tim kiếm	Về Access
Danh mục Chức vụ	Xem Danh sách Cán bộ	Về Windows
Danh mục Chuyên môn		



M_Solieu : Macro			
	Macro Name	Action	Comment
▶	Danh sách Cán bộ	OpenTable	
	Danh mục Chức vụ	OpenTable	
	Danh mục Chuyên môn	OpenTable	

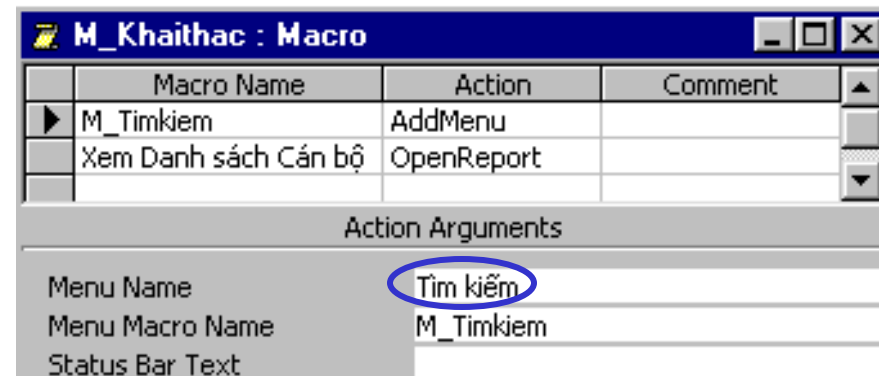
  

Action Arguments	
Table Name	DSCanbo
View	Datasheet
Data Mode	Edit

## IV. Tạo Menu

### ➤ Bước 2: Tạo Menu cấp 2

<i>Macro Name</i>	<i>Action</i>
M_Timkiem	Addmenu
<b>Xem Danh sách Cán bộ</b>	Open Report



- Action Arguments cho Macro **Xem Danh sách Cán bộ**
  - Report Name: R\_DSCanbo
  - View : Print Preview

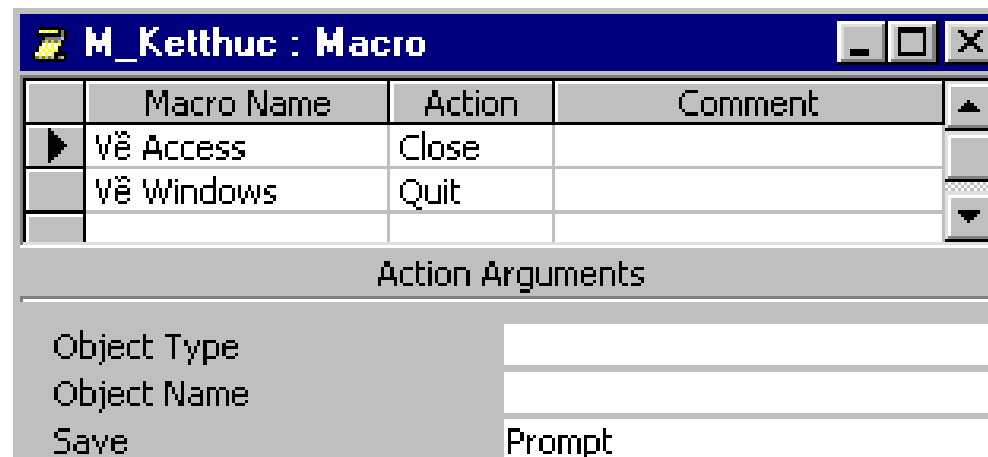
<i>Nhập Số liệu</i>	<i>Khai thác</i>	<i>Kết thúc Chương trình</i>
Danh sách Cán bộ	Tìm kiếm	Theo Chuyên môn
Danh mục Chức vụ	Xem Danh sách Cán bộ	Theo Ngoại ngữ
Danh mục Chuyên môn		Theo Năm sinh
		Về Access
		Về Windows

Menu cấp 2

## IV. Tạo Menu

### ➤ Bước 2: Tạo Menu cấp 2

<i>Macro Name</i>	<i>Action</i>
Về Access	Close
Về Windows	Quit



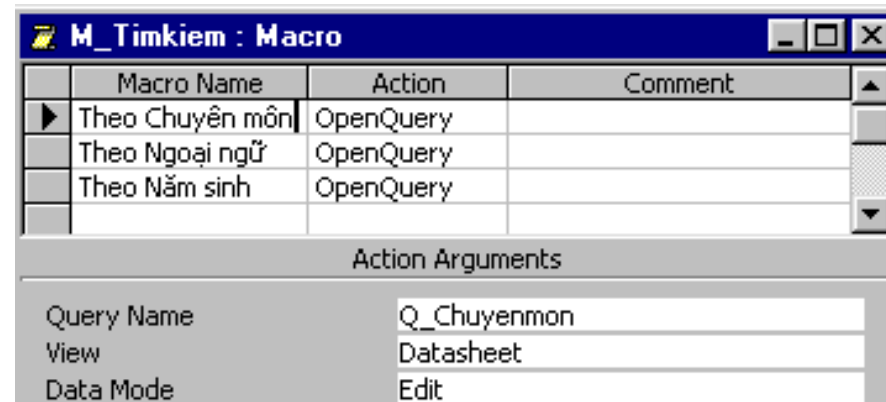
<i>Nhập Số liệu</i>	<i>Khai thác</i>	<i>Kết thúc Chương trình</i>
Danh sách Cán bộ	Tim kiếm ▶	Về Access
Danh mục Chức vụ	Xem Danh sách Cán bộ	Về Windows
Danh mục Chuyên môn	Theo Chuyên môn	
	Theo Ngoại ngữ	
	Theo Năm sinh	

Menu cấp 2

## IV. Tạo Menu

### ➤ Bước 3: Tạo Menu cấp 3

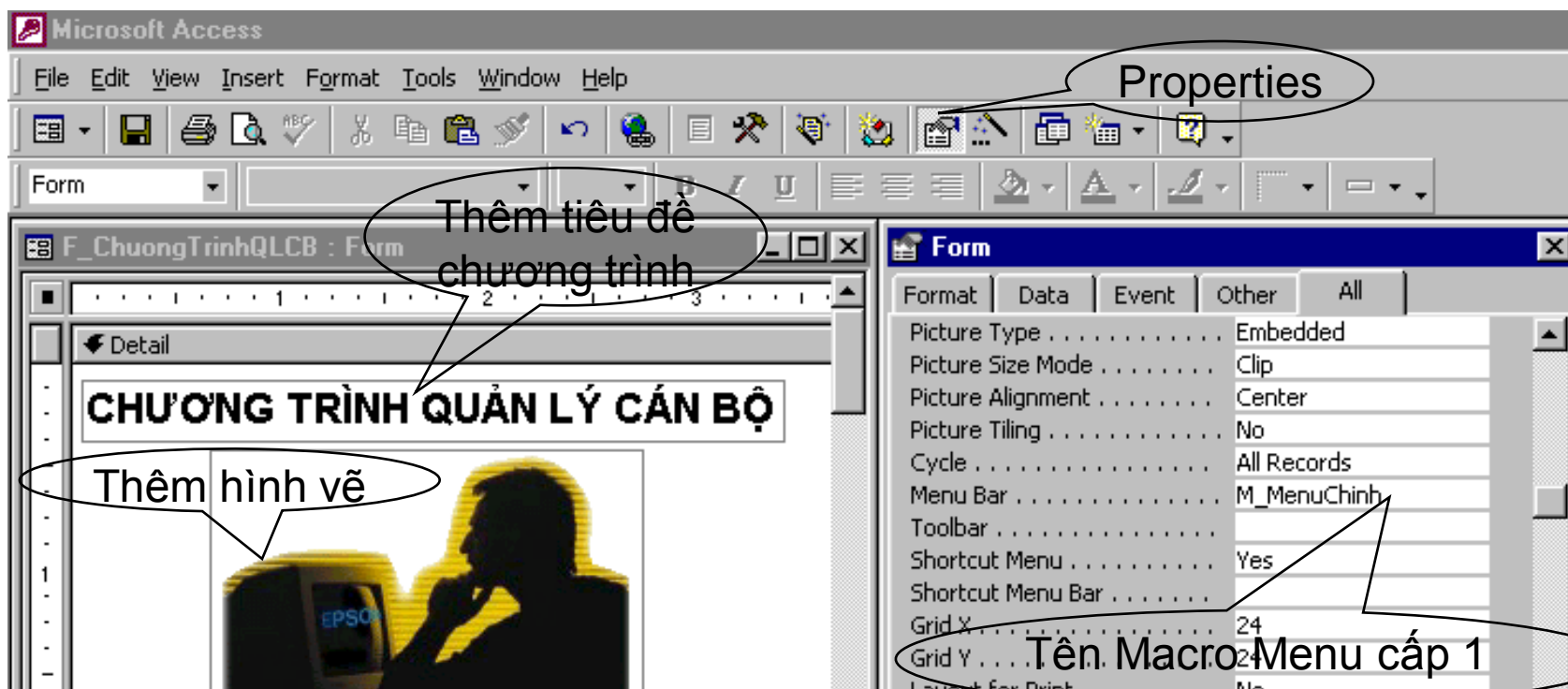
<i>Macro Name</i>	<i>Action</i>	<i>Query Name</i>
Theo Chuyên môn	OpenQuery	Q_Chuyenmon
Theo Ngoại ngữ	OpenQuery	Q_Ngoaingu
Theo Năm sinh	OpenQuery	Q_Namsinh



<i>Nhập Số liệu</i>	<i>Khai thác</i>	<i>Kết thúc Chương trình</i>
Danh sách Cán bộ	Tim kiếm	Về Access
Danh mục Chức vụ	Xem Danh sách Cán bộ	Về Windows
Danh mục Chuyên môn	Theo Chuyên môn	
	Theo Ngoại ngữ	
	Theo Năm sinh	

Menu cấp 3

- Bước 4: Gắn Menu vừa tạo lên một Form (hoặc Report)
- Trong cửa sổ Database, chọn Form / New
- Kích chuột vào biểu tượng Properties trên thanh công cụ để hiện hộp Form



- Menu Bar: ghi tên của Macro dành cho Menu cấp 1

## IV. Tạo Menu

- Trang trí thêm cho Form các thành phần như
  - Tiêu đề chương trình
  - Hình vẽ để minh họa
  - Các trang trí khác (nếu muốn)
  
- Lưu ý: Để tạo Menu, trước khi tạo các Macro, **các nguồn số liệu như Table, Query, Report (nếu có) bạn phải chuẩn bị trước**



Mở đầu

Table

Query

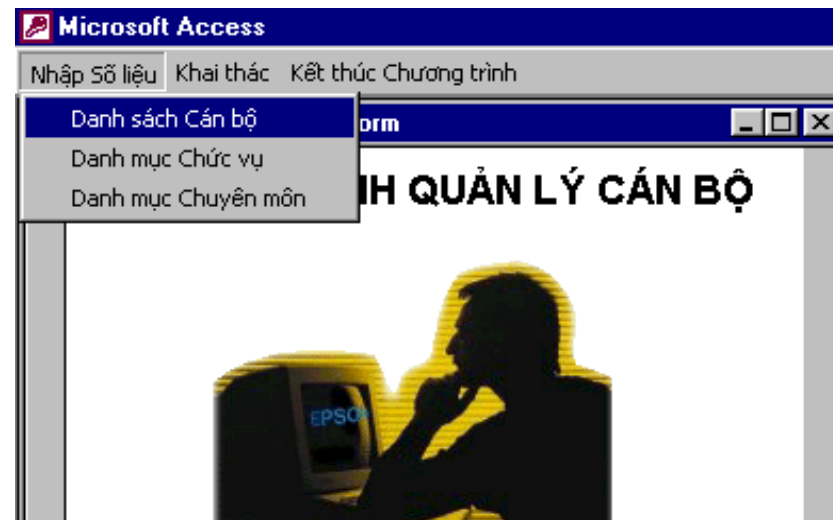
Form

Report

Macro

## V. Sử dụng Menu

➤ Mở Form đã gắn với thực đơn được tạo, có kết quả sau:



# MS-ACCESS 2000

**BÀI 01:(2Tiết)**  
**PTTK CSDL& MS-ACCESS**

**BÀI 02:(4Tiết)** TẠO CSDL,  
TABLE, &RELATIONSHIP

**BÀI 03:(2Tiết)**  
**KHÁI QUÁT QUERY**

**BÀI 04:(6Tiết)**  
**SELECT QUERY**

**BÀI 05: (6Tiết)**  
**QUERY CAO CẤP**

**BÀI 06:(2 Tiết) FORM**  
**THIẾT KẾ VỚI WIZARD**

**BÀI 07:(6Tiết) FORM**  
**THIẾT KẾ DESIGN VIEW**

**BÀI 08:(6Tiết)**  
**MAIN FORM – SUBFORM**

**BÀI 09&10: (6Tiết)**  
**REPORT**

**BÀI 11: (4Tiết)**  
**MACRO&MODUL**

**Ôn Tập (2Tiết)**

**KIỂM TRA (2Tiết)**

# **KHÁI QUÁT VỀ QUERY**

**I. Khái niệm và lý do sử dụng**

**II. Các loại query**

**III. Cách tạo Query**

**IV. Các hình thức hiển thị Query**

**V. Lưu và điều chỉnh Query**

# I. Khái niệm và lý do sử dụng

## Query là gì :

Query là công cụ của Access được dùng để truy vấn dữ liệu trên một hay nhiều Table. Query cũng được dùng như một nguồn dữ liệu khác cho các công cụ khác như Form, Report, hay Query khác.

## Các trường hợp sử dụng Query

- ✓ Lựa chọn các Query cần thiết
- ✓ Lựa chọn các mẫu tin, sắp xếp theo thứ tự nào đó.
- ✓ Tham khảo dữ liệu của nhiều Table
- ✓ Thực hiện phép tính toán.
- ✓ Tạo dữ liệu cho các Form, Report, các Query khác

## II. Các Loại Query

### 1. Select Query (Chọn lọc)

Sử dụng khá phổ biến để trích dữ liệu từ 1 hay nhiều Table.

Ngoài ra còn có thể cho phép tính toán, lọc và kết nhóm.

### 2. CrossTab Query (Tham khảo chéo).

Dùng để tổng hợp dữ liệu theo chủng loại (Tổng số, s.lượng, tb) Loại này hữu hiệu trong việc phân tích dữ liệu.

### 3. Action Query (Hành động)

Dùng để tạo mới hay thực hiện thay đổi cho Table tùy theo công dụng bao gồm : Delete Query, Update Query, Append Query, Make table Query

### 4. Parameter Query (Tham số)

Tạo một Query mà khi khởi động cần một hay nhiều tham số để tạo ra kết quả. Tham số được ghi dưới dạng một tên biến.

### III. Cách tạo Query

- **Dùng Wizard** : Ít dùng vì hạn chế trên một số Query

- **Dùng Query Design**:

Người dùng tự thiết kế thông qua cửa sổ Design View.

Các bước khái quát:

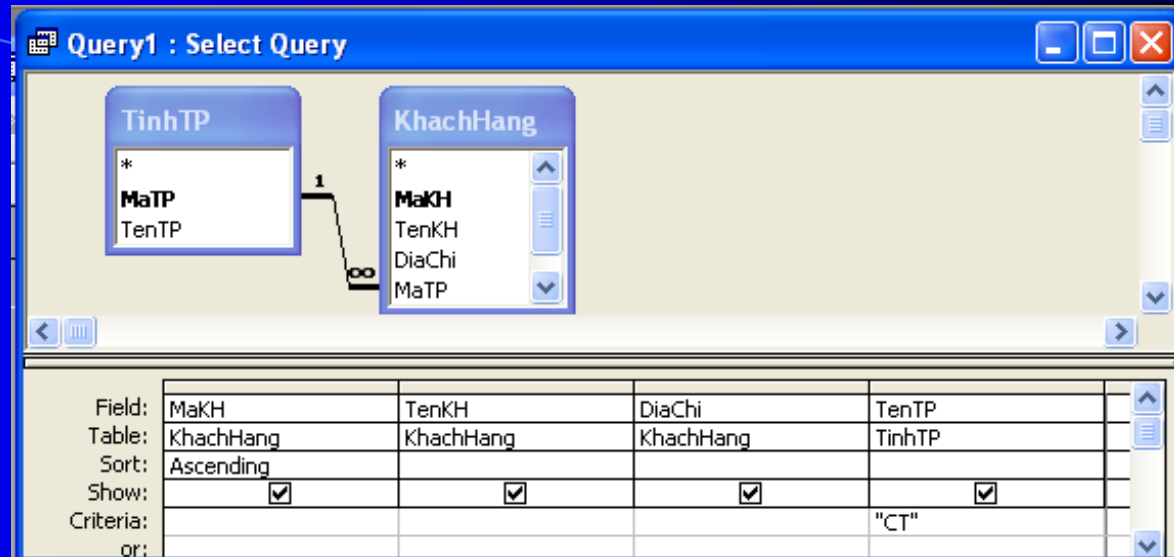
- ✓ Khởi động Design View
- ✓ Chọn các Table, Query liên quan
- ✓ Khai báo các mối quan hệ nếu chưa có
- ✓ Khai báo các tiêu chuẩn lọc nếu cần
- ✓ Chạy thử và lưu lại.

## IV. Các hình thức hiển thị Query

Cửa sổ Query có 3 hình thức hiển thị :

- ✓ **Datasheet View** : Là dạng hiển thị chuẩn khi chạy Query
- ✓ **Design View** : Hiển thị Query theo dạng thiết kế
- ✓ **SQL View** : Hiển thị thiết kế Query dưới dạng dòng lệnh theo ngôn ngữ SQL

# Design View



# SQL View

```
Query32 : Select Query
SELECT NhaCungCap.MaNCC, NhaCungCap.TenNCC, MatHang.MaHang, MatHang.TenHang, MatHang.DonViTinh, MatHang.DonGia
FROM NhaCungCap, MatHang
WHERE (((NhaCungCap.MaNCC)=[MatHang].[MaNCC]) AND ((MatHang.MaNCC)=2 Or (MatHang.MaNCC)=4));
```

# DataSheet View

	MaDDH	NgayDH	MaKH	TenKH	MaNV	HoLot	TenNV
	10144	1/7/1997	RUBIMEX	Công Nghiệp Cao Su	7	Võ Minh	Hải
	10145	1/9/1997	SAMACO	SXKD Dịch Vụ Tổng Hợp	9	Vương Ngọc	Yến
	10148	1/14/1997	VITICO	Hóa Nhựa Vĩnh Tiến	1	Nguyễn Thị Ki	Liên
	10150	1/17/1997	CODACO	Cơ Khí Dân Dụng	4	Hồ Thị	Trang
	10156	1/28/1997	COTEC	Công Nghệ Mới	4	Hồ Thị	Trang
	10157	1/29/1997	DHP	Đại Hưng Phát	4	Hồ Thị	Trang
	10158	1/30/1997	SAMECO	Cơ Khí Sài Gòn	6	Bùi Văn	Bá
	10162	2/6/1997	GIAY32	Giày 32	6	Bùi Văn	Bá
	10163	2/7/1997	TRACODI	Đầu Tư Phát Triển GTVT	4	Hồ Thị	Trang
▶	10166	2/13/1997	VTP	Vạn Thịnh Phát	8	Nguyễn Hồng	Phát

Record: 10 of 10



# V. LƯU VÀ ĐIỀU CHỈNH

**Lưu** : Khi hoàn tất thiết kế hoặc hiệu chỉnh cần lưu lại **Menu File/Save**. nếu không khi đóng Acces sẽ hỏi bạn cần lưu không.

**Hiệu chỉnh**: Thiết kế trong Query : Chọn Query và click chức năng **Design**

**Sao chép Query** : Chọn Query/ Edit/Copy ⇒ Edit/Past

**Xóa** : Chọn Query/ Nhấn Delete / Yes

**BÀI 04**

# **THIẾT KẾ SELECT QUERY**

**I. THỰC HIỆN SELECT QUERY VỚI DESIGN VIEW**

**II. THỰC HIỆN SELECT QUERY SQL VIEW**

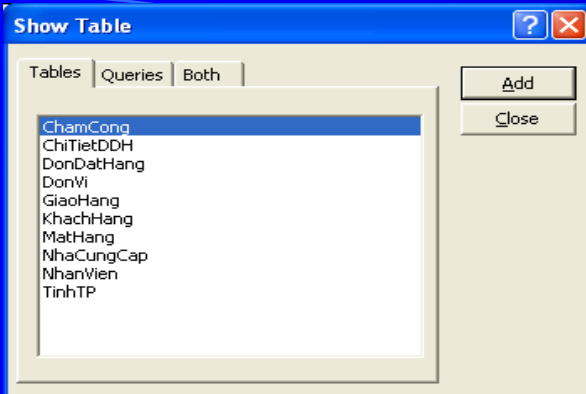
# I. THỰC HIỆN SELECT QUERY VỚI DESIGN VIEW

- **Mở cửa sổ Design View**

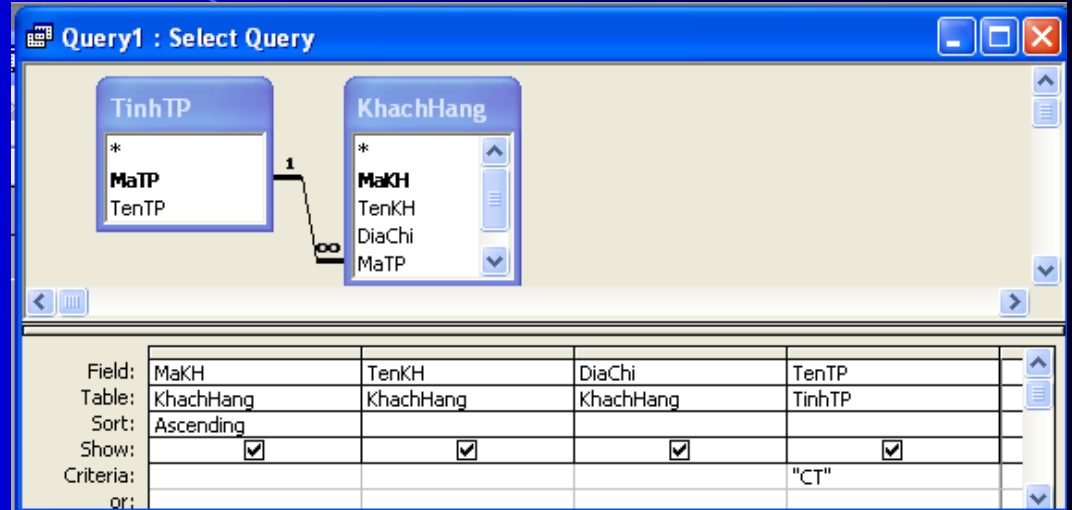
Từ cửa sổ Database : Chọn Query khung bên trái, chọn Create Query in design khung bên phải.

Hay chọn biểu tượng New trên cửa sổ Database chọn Query chọn Design View trong hộp thoại New Query.

Hay chọn lệnh Insert Query



## Chọn các Table tham gia vào Select Query



## Cửa sổ Query có dạng:

- ✓ **Khung trên** chứa các Table, Query tham gia
- ✓ **Khung dưới** chứa các Field đưa vào Query:
  - + Dòng Field : Chứa các Field
  - + Dòng Table chứa tên table của Field được chỉ định.
  - + Dòng Sort : Sắp xếp các mẫu tin
  - + Dòng show : Hiện thị hoặc không hiện thị các filed chỉ định
  - + Dòng Criteria : Dùng khai báo tiêu chuẩn lọc.
  - + Dòng Or khai báo bổ sung điều kiện lọc

## Thêm hay bỏ bớt các Table tham gia vào Select Query.

**Thêm** : Trong Query Design chọn Menu Query>Show Table. Add các Table cần đưa vào

**Bớt** : Chọn Table nhấn Delete (hay Menu Query\Remove Table) để xóa

## Khai báo tiêu chuẩn vấn tin (Criterial)

>234	Trị field lớn hơn 234
BetWeen #2/2/1997# and #1/12/1997#	Trị field từ ngày 2/2/1997 đến 1/12/1997
>="Callahan"	Trị Field có tên "Callahan" trở lên
Like "Sm?th"	Trị Field có tên "Callahan" trở lên

Like "L*ng"	128 <sup>th</sup> , Perth, . . .
Like "* /10/99"	Tất cả ngày của tháng 3
Not "H*"	Không có ký tự đầu = H
Not 2	Trị Field khác 2
In("Canada", "UK")	Là Canada hay UK
Len([TenNV]) > Val(30)	Tên nhân viên có chiều dài hơn 30 kí tự
Right([MaDH],2)="99"	Các đơn hàng có 2 ký tự cuối là 99
Between Date() And DateAdd("m",3,Date())	Thời gian từ hôm nay đến 3 tháng tới
NgàyDH) - Date() < 30	Trước 30 ngày so với ngày hiện tại
Year([NgàyDH])=1998	Đơn đặt hàng trong năm 1998

<code>DatePart("q",[NgàyDH])=4</code>	Đơn đặt hàng thuộc quý 4
<code>DateSerial(Year([NgàyDH]) Month([NgàyDH])+1,1)-1</code>	Hiện thị đơn hàng vào ngày cuối cùng mỗi tháng
<code>Year([NgàyDH])=Year(now()) and Month([NgàyDH])= Month(now())</code>	Hiện thị đơn hàng trong tháng và năm hiện tại

# Dòng Total trong cửa sổ Design View Query

Mặc định dòng total không có , Muốn hiện thị : Menu View\Total

<b>Sum</b>	Tổng giá trị trong một Field
<b>Avg</b>	Trị trung bình của một Field
<b>Min</b>	Trị nhỏ nhất trong một Field
<b>Max</b>	Trị lớn nhất trong một Field
<b>Count</b>	Số các trị trong field không đếm trị Null
<b>StDev</b>	Độ lệch chuẩn của các trị trong một Filed
<b>Var</b>	sự biến thiên của trị trong một Field
<b>Group by</b>	Định nghĩa các nhóm muốn thực hiện tính toán
<b>Expression</b>	Phối hợp nhiều hàm trong một biểu thức
<b>Where</b>	Mà không dùng định nghĩa nhóm Chỉ định tiêu chuẩn của một field



## II. THỰC HIỆN SELECT QUERY SQL VIEW

Từ cửa sổ Design view chọn Menu View\SQL View  
Cú pháp đầy đủ của câu lệnh Select là :

```
SELECT [ ALL | DISTINCT ] [TOP n [PERCENT]  
[WITH TIES]] <Select_list>  
FROM <Table_Source>  
[ WHERE <Search_Condition> ] ]  
[ GROUP BY [ALL] Group by expression [ . . .n ) ]  
[ HAVING <Search_Cndition> ]  
[ ORDER BY {Column name [ASC + Desc ]} [ . . .n ]
```

**Ví dụ 1 :**

```
SELECT *  
FROM NhanVien
```

**Ví dụ 2 :**

```
SELECT N.MaNCC, TenNCC, TenHang, DonViTinh,  
DonGia  
FROM NhaCungCap N, MatHang M  
WHERE N.MaNCC=M.MaNCC And (M.MaNCC=2 Or  
[M.MaNCC=4]);
```

**BÀI 05**

# **QUERY CAO CẤP**

**I. PARAMETER QUERY**

**II. CROSSTAB QUERY**

**III. ACTION QUERY**

# I. PARAMETER QUERY

Parameter Query là query được nhận thông số nhập từ bàn phím để định tiêu chuẩn lọc các mẫu tin cho mỗi lần chạy. nếu thiết kế Query Paramater có yêu cầu trả lời nhiều tiêu chuẩn cho mỗi lần chạy thì các khung đối thoại cho từng tiêu chuẩn khai báo lần lượt hiển thị.

Trong hộp thoại “Enter Parameter value” chỉ có thể nhập 1 tiêu chuẩn và không phải là 1 biểu thức.

# Tạo Parameter Query

+ Tạo hay mở Query chế độ DesignView

+ Tại dòng Criteria của field muốn dùng làm tham số :

Nhập chuỗi ký tự thông báo trong cặp ngoặc vuông (Thông báo này sẽ hiện thị trong hộp “Enter Parameter Value” khi chạy Query. Chuỗi ký tự không được trùng với tên Field.

Có thể dùng Query với thông báo yêu cầu đánh vào vài ký tự đầu tiên của trị muốn tìm

EX: Like [Nhập vào vài ký tự đầu tiên của trị muốn tìm]

& \*

Có thể tạo Query parameter với nhiều thông số cho mỗi field hay nhiều thông số cho một Field (Dùng toán tử Between . . . And . . .)

EX: Between [Nhập ngày bắt đầu] and [Nhập ngày kết thúc]

Minh họa:

The screenshot shows the 'Query Design' view for a query named 'Query47 : Select Query'. The 'DonDatHang' table is selected. The design grid below shows the following configuration:

Field:	MaDDH	NgayDH	NgayGiao	MaNV	MaKH
Table:	DonDatHang	DonDatHang	DonDatHang	DonDatHang	DonDatHang
Sort:					
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:		Between [Tu Ngay] And [Den Ngay]			
or:					

This dialog box prompts for the start date. It contains a text input field with the label 'Tu Ngay' and two buttons: 'OK' and 'Cancel'.

This dialog box prompts for the end date. It contains a text input field with the label 'Den Ngay' and two buttons: 'OK' and 'Cancel'.

```
SELECT MaDDH, NgayDH, NgayGiao, MaNV, MaKH
FROM DonDatHang
WHERE (((NgayDH) Between [Tu Ngay] And
[Den Ngay]));
```

## II. CROSSTAB QUERY

Tạo Crosstab Query để tổng kết dữ liệu theo một dạng bảng tính. Query này có thể biểu diễn số lượng lớn dữ kiện theo dạng thức dễ đọc. ngoài ra có thể là dữ kiện cơ sở cho các báo biểu.

### 1. Các bước tạo CrossTab Query

- + Tạo một Query bằng Design View
- + Bật Table Name : MenuView\Table Name
- + Bật Total : Menu View\ Total.
- + Xác định loại Query: Menu Query\CrosstabQuery

## 2. Khai báo tiêu đề dòng, cột và tổng hợp trên dòng

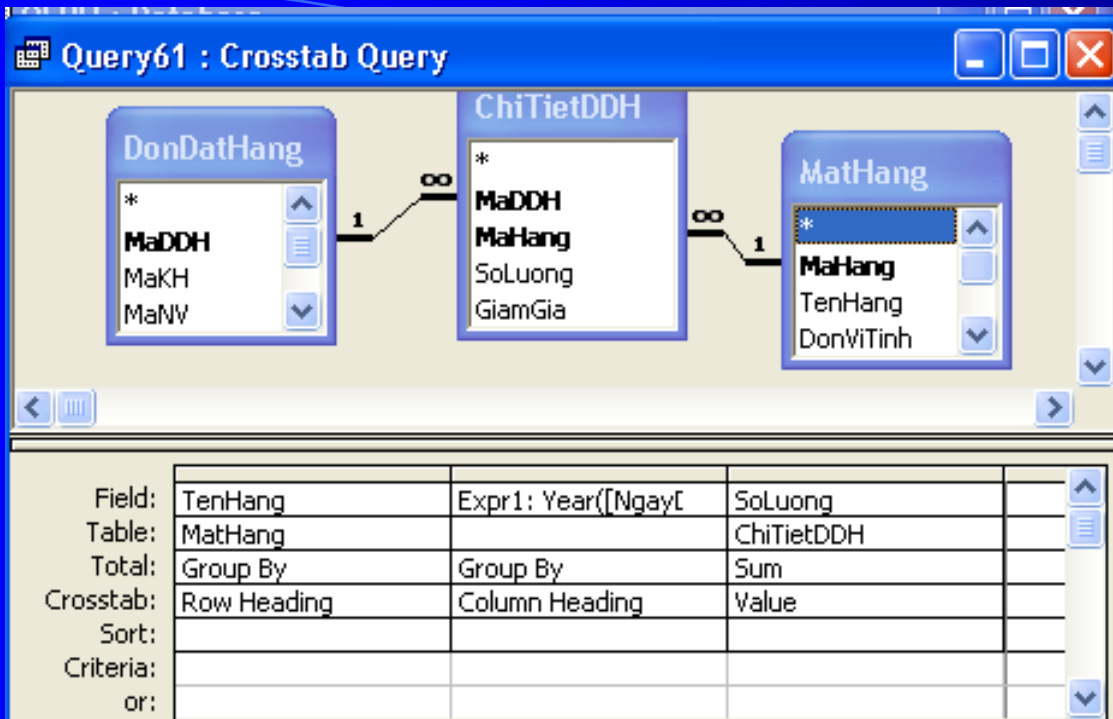
+ Chọn CrossTab của Field làm tiêu đề dòng. của Query rồi chọn mục **Row Heading** (Dòng Total của field này phải khai báo Group by) tối đa 3 field.

+ Chọn CrossTab của Field làm tiêu đề cột. của Query rồi chọn mục **ColumnHeading** (Dòng Total của field này phải khai báo Group by) tối đa 1 field.

+ Chọn CrossTab của Field là tổng kết của Query rồi chọn mục **Value Heading** (Dòng Total của field này phải khai báo các phép tính toán như :Count, Sum)

Chỉ được phép chọn 1 field có dòng CrossTab là Value.





Minh họa:

Query61 : Crosstab Query

TenHang	1997	1998	1999
Bánh Kem	18		6
Bánh Mì	45		
Bơ	40		6
Cá Hộp	24		
Nem	60	9	
Rượu	104		35
Táo	24		
Tiêu	33		

Record: 8 of 8

```

TRANSFORM Sum(ChiTietDDH.SoLuong) AS SumOfSoLuong
SELECT MatHang.TenHang
FROM MatHang INNER JOIN (DonDatHang INNER JOIN
ChiTietDDH ON DonDatHang.MaDDH = ChiTietDDH.MaDDH)
ON MatHang.MaHang = ChiTietDDH.MaHang
GROUP BY MatHang.TenHang
PIVOT Year([NgayDH]);
  
```

# III. SỬ DỤNG ACTION QUERY

Có 4 loại Action Query

**1. Make Table Query**

**2. Delete Query**

**3. Append Query**

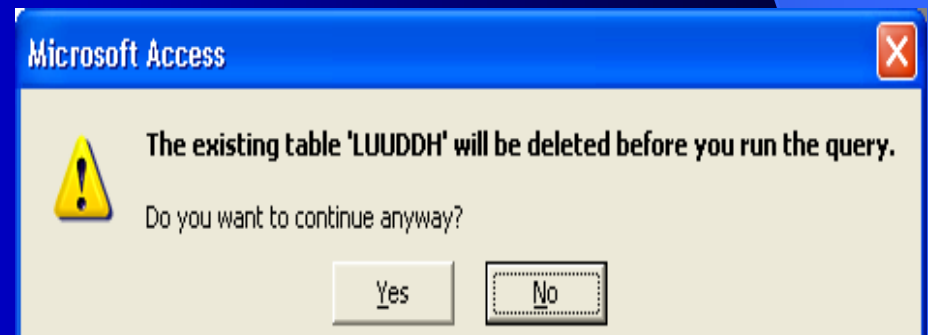
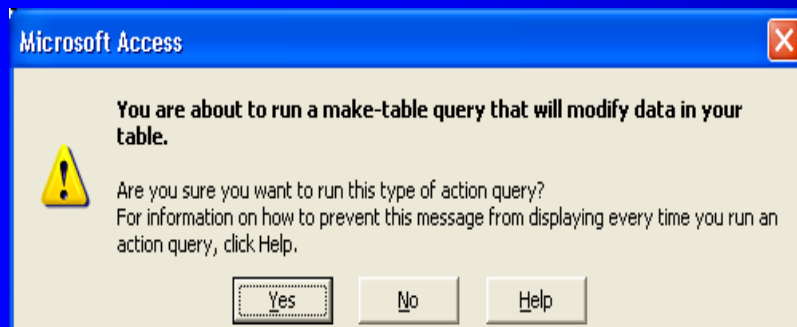
**4. Update Query**

# 1. Male Table Query

Dùng để tạo Table mới từ dữ liệu của các Table khác nhằm tạo bảng lưu cho table.

## Trình tự thực hiện:

- + Tạo Query mới
- + Chọn các field tương ứng
- + Chọn Menu Query/ Make Table Query
- + Đặt tên Table cho khung Table
- + Khai báo nơi lưu trữ dữ liệu: Database hiện tại hay Database khác
- + Hoàn tất : Ghi lưu Query
- + Khi chạy Make Table Query Access thông báo yêu cầu người dùng xác nhận việc thực hiện.

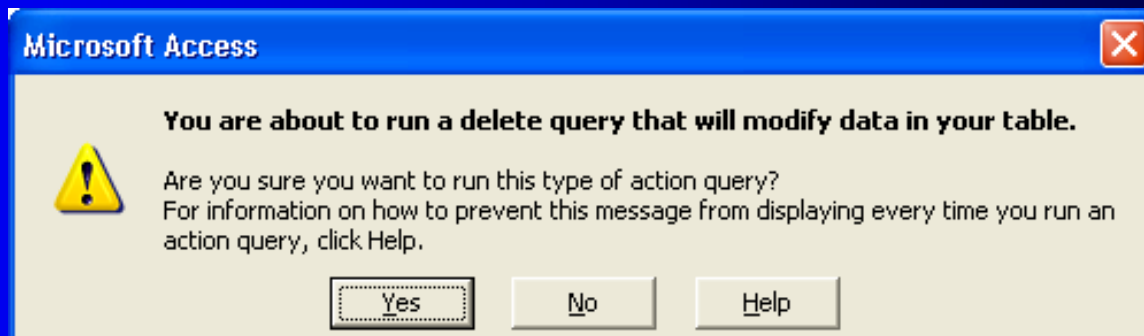


## 2. Delete Query

Dùng để xóa các mẫu tin trong Table.

### Trình tự thực hiện:

- + Tạo Query mới
- + Chọn các Table tham gia
- + Chọn Menu Query/ Delete Query
- + Chọn các field cần xóa ( \* là chọn tất cả)
- + Xác định điều kiện xóa ở dòng Criteria
- + Khi chạy Delete Query Access cảnh báo sẽ không thể phục hồi
- + Khi xóa các mẫu tin bằng delete, chỉ có các mẫu tin bị xóa, table vẫn còn trống. Muốn xóa table phải thực hiện từ cửa sổ Database.

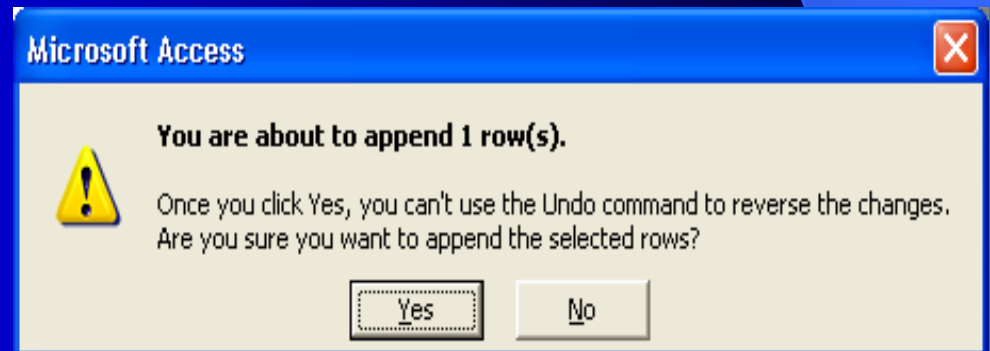
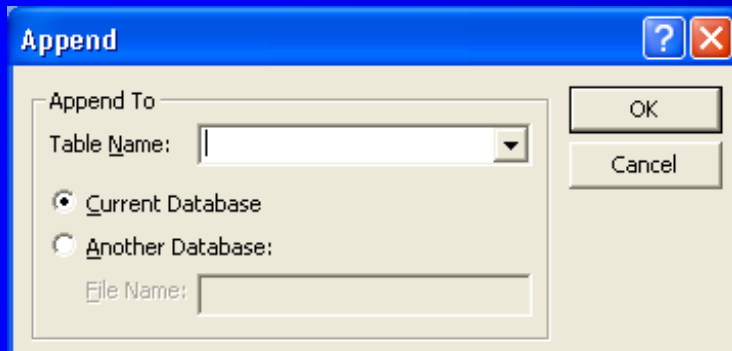


# 3. Append Query

Dùng để nối vào table các mẫu tin lấy từ nhiều table khác.

## Trình tự thực hiện:

- + Tạo Query mới
- + Chọn các Table tham gia
- + Chọn Menu Query/ Append Query
- + Khai báo đích lưu CSDL
- + Xác định Table trong khung Table name
- + Khi chạy Append Query Access sẽ thông báo số record sẽ thêm vào.

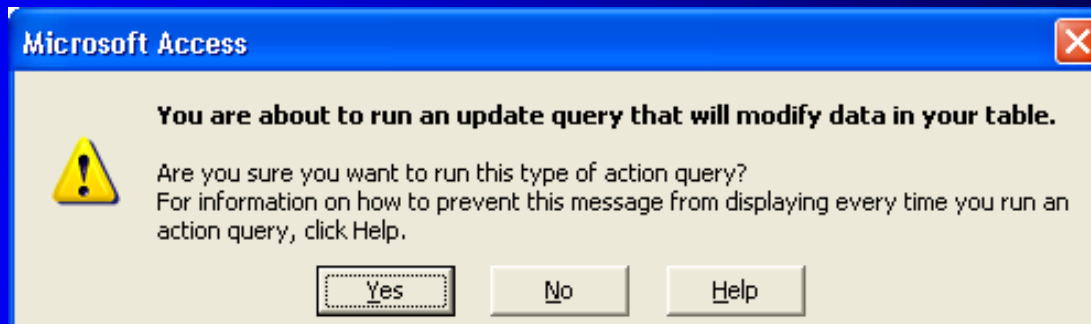


## 4. Update Query

Dùng để cập nhật các mẫu tin trong table.

### Trình tự thực hiện:

- + Tạo Query mới
- + Chọn các Table tham gia
- + Chọn Menu Query/ Update Query
- + Chọn Field muốn cập nhật
- + Khai báo điều kiện tại dòng Criteria
- + Khai báo tại dòng Update To biểu thức theo đó Access cập nhật Field
- + Ghi lưu Update Query
- + Khi chạy Update Query Access sẽ thông báo số record sẽ điều chỉnh



# Giáo trình và bài tập access



## MỤC LỤC

### **TỔNG QUAN CƠ SỞ DỮ LIỆU ACCESS**

#### **Chương 1**

- 1. GIỚI THIỆU MICROSOFT ACCESS**
- 2. KHỞI ĐỘNG VÀ THOÁT KHỎI ACCESS**
- 3. CÁC THAO TÁC TRÊN TẬP TIN CƠ SỞ DỮ LIỆU ACCESS**
- 4. CÁCH SỬ DỤNG CỬA SỔ DATABASE**

#### **Chương 2**

### **BẢNG DỮ LIỆU**

- 1. THIẾT KẾ CƠ SỞ DỮ LIỆU**
- 2. KHÁI NIỆM VỀ BẢNG**
- 3. TẠO BẢNG MỚI TRONG CƠ SỞ DỮ LIỆU**
- 4. THIẾT LẬP KHOÁ CHÍNH (*primary key*)**
- 5. LƯU BẢNG**
- 6. HIỆU CHỈNH BẢNG**
- 7. XEM THÔNG TIN VÀ BỔ SUNG BẢN GHI**
- 8. THIẾT LẬP QUAN HỆ GIỮA CÁC BẢNG**
- 9. SẮP XẾP VÀ LỌC DỮ LIỆU**

#### **Chương 3**

### **TRUY VẤN (QUERY)**

- 1. KHÁI NIỆM TRUY VẤN**
  - 2. CÁC CHẾ ĐỘ HIỂN THỊ TRUY VẤN**
  - 4. THIẾT KẾ TRUY VẤN CHỌN**
  - 5. TRUY VẤN DỰA TRÁN NHIỀU BẢNG DỮ LIỆU**
  - 6. TÍNH TỔNG TRONG TRUY VẤN CHỌN**
- .....



## Nội dung

### Chương 1

# TỔNG QUAN CƠ SỞ DỮ LIỆU ACCESS

---

## 1. GIỚI THIỆU MICROSOFT ACCESS

*Microsoft Access* là một Hệ Quản Trị Cơ Sở Dữ Liệu (QTCSDL) tương tác người sử dụng chạy trong môi trường Windows. Microsoft Access cho chúng ta một công cụ hiệu lực và đầy sức mạnh trong công tác tổ chức, tìm kiếm và biểu diễn thông tin.

*Microsoft Access* cho ta các khả năng thao tác dữ liệu, khả năng liên kết và công cụ truy vấn mạnh mẽ giúp quá trình tìm kiếm thông tin nhanh. Người sử dụng có thể chỉ dùng một truy vấn để làm việc với các dạng cơ sở dữ liệu khác nhau. Ngoài ra, có thể thay đổi truy vấn bất kỳ lúc nào và xem nhiều cách hiển thị dữ liệu khác nhau chỉ cần động tác nhấp chuột.

*Microsoft Access* và khả năng kết xuất dữ liệu cho phép người sử dụng thiết kế những biểu mẫu và báo cáo phức tạp đáp ứng đầy đủ các yêu cầu quản lý, có thể vận động dữ liệu và kết hợp các biểu mẫu và báo cáo trong một tài liệu và trình bày kết quả theo dạng thức chuyên nghiệp.

*Microsoft Access* là một công cụ đầy năng lực để nâng cao hiệu suất công việc. Bằng cách dùng các Wizard của MS Access và các lệnh có sẵn (macro) ta có thể dễ dàng tự động hóa công việc mà không cần lập trình. Đối với những nhu cầu quản lý cao, Access đưa ra ngôn ngữ lập trình Access Basic (Visual Basic For application) một ngôn ngữ lập trình mạnh trên CSDL.

## 2. KHỞI ĐỘNG VÀ THOÁT KHỎI ACCESS

### 2.1. Khởi động ACCESS

Chọn nút Start trên thanh Task bar

Chọn Programs

Chọn Microsoft ACCESS



Khung hội thoại Microsoft ACCESS gồm:

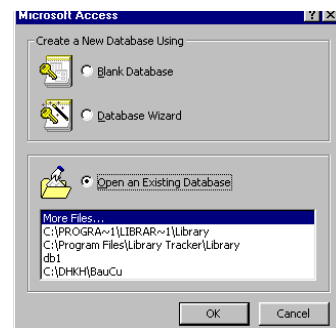
Create a New Database Using : Tạo CSDL ứng dụng mới.

*Blank Database* : Tạo CSDL trống.

*Database Wizard* : Tạo với sự trợ giúp

của Wizard.

Open an Existing Database : Mở một CSDL có sẵn.



## 2.2. Thoát khỏi ACCESS

Chọn File/Exit hoặc nhấn tổ hợp phím ALT+F4

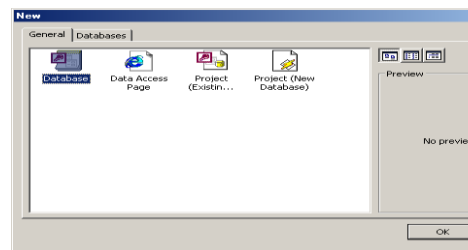
## 3. CÁC THAO TÁC TRÊN TẬP TIN CƠ SỞ DỮ LIỆU ACCESS

### 3.1. Tạo một tập tin CSDL

Thực hiện các thao tác sau:

Chọn File/New hoặc chọn biểu tượng

New trên thanh công cụ



Chọn Database, chọn OK

Trong mục *Save in*: Chọn thư mục cần chứa tên tập tin.

*File name*: Chọn tên tập tin cần tạo

(Phần mở rộng mặc định là MDB)

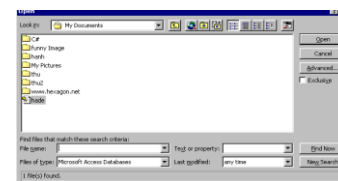
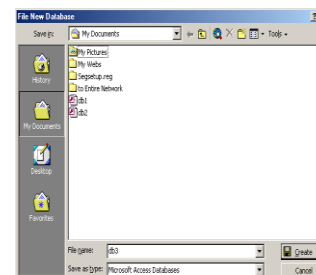
### 3.2. Mở một CSDL đã tồn tại trên đĩa

Chọn File/Open database (Hoặc click biểu tượng Open)

Trong mục *Look in* : Chọn thư mục cần chứa tên tập tin cần mở.

*File name*: Chọn tên tập tin cần mở.

Chọn Open



### 3.3. Đóng một CSDL

Chọn File/Close hoặc ALT+F4

### 3.4. Các thành phần cơ bản của một tập tin CSDL ACCESS

Một tập tin CSDL ACCESS gồm có 6 thành phần cơ bản sau

*Bảng (Tables)* : Là nơi chứa dữ liệu

*Truy vấn (Queries)* : Truy vấn thông tin dựa trên một hoặc nhiều bảng.

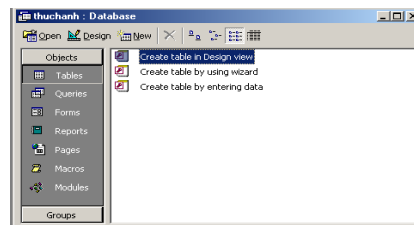
*Biểu mẫu (Forms)* : Các biểu mẫu dùng để nhập dữ liệu hoặc hiển thị dữ liệu.

*Báo cáo (Reports)* : Dùng để in ấn.

*Pages (Trang)* : Tạo trang dữ liệu.

*Macros (Tập lệnh)* : Thực hiện các tập lệnh.

*Modules (Đơn thể)* : Dùng để lập trình Access Basic



#### 4. CÁCH SỬ DỤNG CỬA SỔ DATABASE

Như đã nói ở trên, một CSDL của Access chứa trong nó 7 đối tượng chứ không đơn thuần là bảng dữ liệu. Sau khi tạo mới một CSDL hoặc mở một CSDL có sẵn Access sẽ hiển thị một cửa sổ Database, trên đó hiển thị tên của CSDL đang mở và liệt kê 7 đối tượng mà nó quản lý, mỗi lớp đối tượng đều được phân lớp rõ ràng để tiện theo dõi.

##### 4.1. Tạo một đối tượng mới

Trong cửa sổ Database, chọn tab chứa đối tượng cần tạo (Bảng, Truy vấn, Biểu mẫu, Báo cáo,...) hoặc thực hiện lệnh

*View/Database Object - Table/Query/Form/Report/Pages/Macros/Modules*

Chọn nút New.

##### 4.2. Thực hiện một đối tượng trong CSDL

Trong cửa sổ Database, chọn tab cần thực hiện. Cửa sổ Database liệt kê tên các đối tượng có sẵn, chọn tên đối tượng cần mở.

Chọn nút Open (đối với Bảng, Truy vấn, Biểu mẫu, Trang) hoặc Preview (đối với Báo biểu) hoặc Run (đối với Macro và Module).

##### 4.3. Sửa đổi một đối tượng có sẵn trong CSDL

Trong cửa sổ Database, chọn tab cần thực hiện. Cửa sổ Database liệt kê tên các đối tượng có sẵn, chọn tên đối tượng cần mở, Chọn nút Design.

## Chương 2

# BẢNG DỮ LIỆU

---

Bảng là đối tượng chủ yếu chứa các thông tin cần quản lý, có thể đó chỉ là một vài địa chỉ đơn giản hay cả vài chục nghìn bản ghi chứa đựng thông tin liên quan đến các hoạt động SXKD của một công ty xuất nhập khẩu nào đó. Trước khi ta muốn làm việc với bất kỳ một CSDL nào thì ta phải có thông tin để quản lý, các thông tin đó nằm trong các bảng, nó là cơ sở để cho người sử dụng tạo các đối tượng khác trong CSDL như truy vấn, biểu mẫu, báo biểu...

### 1. THIẾT KẾ CƠ SỞ DỮ LIỆU

Một CSDL được thiết kế tốt cho phép người sử dụng truy cập nhanh chóng đến những thông tin cần tham khảo, giúp tiết kiệm được thời gian truy xuất thông tin. Một CSDL thiết kế tốt giúp người sử dụng rút ra được những kết quả nhanh chóng và chính xác hơn.

Để thiết kế một CSDL tốt chúng ta phải hiểu cách mà một Hệ QTCSDL quản trị các CSDL như thế nào. MS Access hay bất kỳ một Hệ QTCSDL nào có thể cung cấp các thông tin cho chúng ta một cách chính xác và hiệu quả nếu chúng được cung cấp đầy đủ mọi dữ kiện về nhiều đối tượng khác nhau lưu trữ trong các bảng dữ liệu. Ví dụ ta cần một bảng để chứa thông tin về lý lịch của cán bộ, một bảng khác để chứa các đề tài nguyên cứu khoa học của các cán bộ...

Khi bắt tay thiết kế CSDL, chúng ta phải xác định và phân tích các thông tin muốn lưu trữ thành các đối tượng riêng rẽ, sau đó báo cho Hệ QTCSDL biết các đối tượng đó liên quan với nhau như thế nào. Dựa vào các quan hệ đó mà Hệ QTCSDL có thể liên kết các đối tượng và rút ra các số liệu tổng hợp cần thiết.

### CÁC BƯỚC THIẾT KẾ CSDL

**Bước 1:** Xác định mục tiêu khai thác CSDL của chúng ta. Điều này quyết định các loại sự kiện chúng ta sẽ đưa vào MS Access.

**Bước 2:** Xác định các bảng dữ liệu cần thiết. Mỗi đối tượng thông tin sẽ hình thành một bảng trong CSDL của chúng ta.

**Bước 3:** Sau khi đã xác định xong các bảng cần thiết, tiếp đến ta phải chỉ rõ thông tin nào cần quản lý trong mỗi bảng, đó là xác định các trường. Mỗi loại thông tin trong bảng gọi là trường. Mọi mẫu in trong cùng một bảng đều có chung cấu trúc các trường. Ví dụ: Trong lý lịch khoa học cán bộ, những trường (thông tin) cần quản lý là: “HỌ VÀ TÊN”, “CHUYÊN MÔN”, “HỌC VỊ”, “HỌC HÀM”,...

**Bước 4:** Xác định các mối quan hệ giữa các bảng. Nhìn vào mỗi bảng dữ liệu và xem xét dữ liệu trong bảng này liên hệ thế nào với dữ liệu trong bảng khác. Thêm trường hoặc tạo bảng mới để làm rõ mối quan hệ này. Đây là vấn đề hết sức quan trọng, tạo được quan hệ tốt sẽ giúp chúng ta nhanh chóng truy tìm và kết xuất dữ liệu.

**Bước 5:** Tinh chế, hiệu chỉnh lại thiết kế. Phân tích lại thiết kế ban đầu để tìm lỗi, tạo bảng dữ liệu và nhập vào vài bản ghi, thử xem CSDL đó phản ánh thế nào với những yêu cầu truy xuất của chúng ta, có rút được kết quả đúng từ những bảng dữ liệu đó không. Thực hiện các chỉnh sửa thiết kế nếu thấy cần thiết.

## 2. KHÁI NIỆM VỀ BẢNG

**Bảng** là nơi chứa dữ liệu về một đối tượng thông tin nào đó như SINH VIÊN, HÓA ĐƠN,... Mỗi hàng trong bảng gọi là một **bản ghi** (record) chứa các nội dung riêng của đối tượng đó. Mỗi bản ghi của một bảng đều có chung cấu trúc, tức là các **trường** (field). Ví dụ: Cho bảng dưới đây để quản lý lý lịch khoa học cán bộ trong trường đại học, có các trường MACB (Mã cán bộ), TRINHDOVH (Trình độ văn hóa), CHUYENMON (Chuyên môn),...

Trong một CSDL có thể chứa nhiều bảng, thường mỗi bảng lưu trữ nhiều thông tin (dữ liệu) về một đối tượng thông tin nào đó, mỗi một thông tin đều có những kiểu đặc trưng riêng, mà với Access nó sẽ cụ thể thành những kiểu dữ liệu của các trường.

### 3. TẠO BẢNG MỚI TRONG CƠ SỞ DỮ LIỆU

Trong MS Access có hai cách để tạo bảng, một là cách dùng Table Wizard, nhưng các trường ở đây MS Access tự động đặt tên và không có bàn tay can thiệp của người sử dụng. Ở đây, sẽ đưa ra cách tạo mới bảng hoàn toàn do người sử dụng.

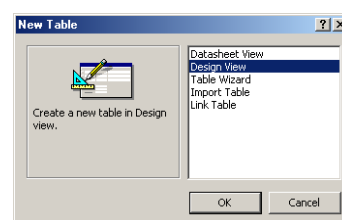
#### 3.1. Tạo bảng không dùng Table Wizard

Trong cửa sổ Database, chọn tab Table (hoặc Lệnh *View/Daatabase object - Table*)

Chọn nút **New**, xuất hiện hộp thoại

*Datasheet View*: Trên màn hình sẽ xuất hiện một bảng trống với các trường (tiêu đề cột) lần lượt Field1, field2

*Design View*: Trên màn hình xuất hiện cửa sổ thiết kế bảng, người sử dụng tự thiết kế bảng.



*Table Wizard*: Thiết kế bảng với sự trợ giúp của MS Access

*Import table*: Nhập các bảng và các đối tượng từ các tập tin khác vào CSDL hiện thời.

*Link table*: Tạo bảng bằng cách nối vào CSDL hiện thời các bảng của CSDL khác.

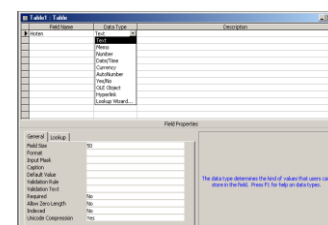
Chọn chức năng Design View, chọn OK.

#### 3.2. Sử dụng Design View

*Field Name*: Tên trường cần đặt (thông tin cần quản lý)

*Data Type*: Kiểu dữ liệu của trường

*Description*: Mô tả trường, phần này chỉ mang ý nghĩa làm rõ thông tin quản lý, có thể bỏ qua trong khi thiết kế bảng.



*Field properties*: Các thuộc tính của trường

Xác định khoá chính của bảng (nếu có)

Xác định thuộc tính của bảng, Lưu bảng dữ liệu

## Đặt tên trường

Tên trường ở đây không nhất thiết phải có độ dài hạn chế và phải sát nhau, mà ta có thể đặt tên trường tùy ý nhưng không vượt quá 64 ký tự kể cả ký tự trắng. Lưu ý rằng, tên trường có thể đặt dài nên nó dễ mô tả được thông tin quản lý, nhưng sẽ khó khăn hơn khi ta dùng các phát biểu SQL và lập trình Access Basic. Do đó khi đặt tên trường ta nên đặt ngắn gọn, dễ gọi nhớ và không chứa ký tự trắng.

## Kiểu dữ liệu

MS Access cung cấp một số kiểu dữ liệu cơ bản sau:

Kiểu dữ liệu	Dữ liệu vào	Kích thước
Text	Văn bản	Tối đa 255 byte
Memo	Văn bản nhiều dòng, trang	Tối đa 64000 bytes
Number	Số	1,2,4 hoặc 8 byte
Date/Time	Ngày giờ	8 byte
Currency	Tiền tệ (Số)	8 byte
Auto number	ACCESS tự động tăng lên một khi một bản ghi được tạo	4 byte
Yes/No	Lý luận (Boolean)	1 bit
OLE Object	Đối tượng của phần mềm khác	Tối đa 1 giga byte
Lookup Wizard		Trường nhận giá trị do người dùng chọn từ 1 bảng khác hoặc 1 danh sách giá trị định trước
Hyper link	Liên kết các URL	

## Quy định thuộc tính, định dạng cho trường

Đặt thuộc tính là một phần không kém quan trọng, nó quyết định đến dữ liệu thực sự lưu giữa trong bảng, kiểm tra độ chính xác dữ liệu khi nhập vào, định dạng

dữ liệu nhập vào ... Mỗi một kiểu dữ liệu sẽ có các thuộc tính và các đặc trưng và khác nhau. Sau đây là các thuộc tính, định dạng của các kiểu dữ liệu.

Để tăng thêm tốc độ xử lý khi nhập dữ liệu cũng như các công việc tìm kiếm sau này thì việc quy định dữ liệu rất quan trọng.

Các trường trong ACCESS có các thuộc tính sau:

### **3.2.1. Field Size**

Quy định kích thước của trường và tùy thuộc vào từng kiểu dữ liệu

*Kiểu Text*: Chúng ta quy định độ dài tối đa của chuỗi.

*Kiểu Number*: Có thể chọn một trong các loại sau:

Byte: 0..255

Integer: -32768..32767

Long Integer: -3147483648.. 3147483647

Single:  $-3,4 \times 10^{38}$  ..  $3,4 \times 10^{38}$  (Tối đa 7 số lẻ)

Double:  $-1.797 \times 10^{308}$  ..  $1.797 \times 10^{308}$  (Tối đa 15 số lẻ)

#### *Decimal Places*

Quy định số chữ số thập phân ( Chỉ sử dụng trong kiểu Single và Double)

Đối với kiểu Currency mặc định decimal places là 2

### **3.2.2. Format**

Quy định dạng hiển thị dữ liệu, tùy thuộc vào từng kiểu dữ liệu.

*Kiểu chuỗi*: Gồm 3 phần

<Phần 1>;<Phần 2>;<Phần 3>

Trong đó:

<Phần 1>: Chuỗi định dạng tương ứng trong trường hợp có chứa văn bản.

<Phần 2>: Chuỗi định dạng tương ứng trong trường hợp không chứa văn bản.

<Phần 3>: Chuỗi định dạng tương ứng trong trường hợp null



**Các ký tự dùng để định dạng chuỗi**

Ký tự	Tác dẽch vuũ du lẽchũng
@	Chuỗi ký tự
>	Đổi tất cả ký tự nhập vào thành in hoa
<	Đổi tất cả ký tự nhập vào thành in thường
“Chuỗi ký tự “	Chuỗi ký tự giữa 2 dấu nháy
\<ký tự>	Ký tự nằm sau dấu \
[black] [White] [red] Hoặc [<số>] Trong đó 0<=số<=56	Màu

**Ví dụ**

Cách định dạng	Dữ liệu	Hiện thị
@ @ @ - @ @ @ @	123456 abcdef	123-456 abc-def
>	Tinhoc	TINHOC
<	TINHOC	Tinhoc
@;”Không có”;”Không biết”	Chuỗi bất kỳ Chuỗi rỗng Giá trị trống (Null)	Hiện thị chuỗi Không có Không biết

**Kiểu Number**

Định dạng do ACCESS cung cấp

Dạng	Dữ liệu	Hiện thị
General Number	1234.5	1234.5
Currency	1234.5	\$1.234.50
Fixed	1234.5	1234

Standard	1234.5	1,234.50
Percent	0.825	82.50%
Scientific	1234.5	1.23E+03

### **Định dạng do người sử dụng**

<Phần 1>;<Phần 2>;<Phần 3>;<Phần 4>

<Phần 1>: Chuỗi định dạng tương ứng trong trường hợp số dương.

<Phần 2>: Chuỗi định dạng tương ứng trong trường hợp số âm.

<Phần 3>: Chuỗi định dạng tương ứng trong trường hợp số bằng zero.

<Phần 4>: Chuỗi định dạng tương ứng trong trường hợp null.

Các ký tự định dạng

<b>Ký tự</b>	<b>Tác dụng</b>
.(Period)	Dấu chấm thập phân
,(comma)	Dấu phân cách ngàn
0	Ký tự số (0-9)
#	Ký tự số hoặc khoảng trắng
\$	Dấu \$
%	Phần trăm

**Ví dụ**

<b>Định dạng</b>	<b>Hiển thị</b>
0;(0);;"Null"	Số dương hiển thị bình thường Số âm được bao giữa 2 dấu ngoặc Số zero bị bỏ trống Null hiện chữ Null
+0.0;-0.0;0.0	Hiển thị dấu + phía trước nếu số dương Hiển thị dấu - phía trước nếu số âm Hiển thị 0.0 nếu âm hoặc Null

*Kiểu Date/Time*

Các kiểu định dạng do ACCESS cung cấp

<b>Dạng</b>	<b>Hiển thị</b>
General date	10/30/99 5:10:30PM
Long date	Friday, may 30 , 1999
Medium date	30-jul-1999
Short date	01/08/99
Long time	6:20:00 PM
Medium time	6:20 PM
Short time	18:20

Các ký tự định dạng

<b>Ký tự</b>	<b>Tác dụng</b>
:	Dấu phân cách giờ
/	Dấu phân cách ngày
d	Ngày trong tháng (1-31)
dd	Ngày trong tháng 01-31)
ddd	Ngày trong tuần (Sun -Sat0
W	Ngày trong tuần (1-7)
WW	Tuần trong năm (1-54)
M	Tháng trong năm (1-12)
MM	Tháng trong năm (01-12)
q	Quý trong năm (1-4)
y	Ngày trong năm (1-366)
yy	Năm (01-99)
h	Giờ (0-23)
n	Phút (0-59)
s	Giây (0-59)

**Ví dụ**

<b>Định dạng</b>	<b>Hiển thị</b>
Ddd,"mmm d",yyyy	Mon,jun 2, 1998
Mm/dd/yyyy	01/02/1998

**Kiểu Yes/No**

Các kiểu định dạng

<b>Định dạng</b>	<b>Tác dụng</b>
Yes/No	Đúng/Sai
True/False	Đúng/Sai
On/Off	Đúng/Sai

**Định dạng do người sử dụng:** Gồm 3 phần

&lt;Phần 1&gt;;&lt;Phần 2&gt;;&lt;Phần 3&gt;

Trong đó: &lt;Phần 1&gt;: Bỏ trống

&lt;Phần 2&gt;: Trường hợp giá trị trường đúng

&lt;Phần 3&gt;: Trường hợp giá trị trường sai

**Ví dụ**

<b>Định dạng</b>	<b>Hiển thị</b>	
	Trường hợp True	Trường hợp False
;"Nam";"Nu"	Nam	Nu
;"Co";"Khong"	Co	Khong

**3.2.3. Input mask (Mặt nạ)**

Thuộc tính này dùng để quy định mặt nạ nhập dữ liệu cho một trường.

Các ký tự định dạng trong input mask

Ký tự	Tác dụng
0	Bắt buộc nhập ký tự số
9	Không bắt buộc nhập, ký tự số
#	Không bắt buộc nhập, số 0-9, khoảng trắng, dấu + và -
L	Bắt buộc nhập, ký tự chữ
?	Không bắt buộc nhập, ký tự chữ hoặc khoảng trắng
a	Bắt buộc nhập, ký tự chữ hoặc số
A	Không bắt buộc nhập, ký tự chữ hoặc số
&	Bắt buộc nhập, ký tự bất kỳ
C	Không bắt buộc nhập ký tự bất kỳ
<	Các ký tự bên phải được đổi thành chữ thường
>	Các ký tự bên phải được đổi thành chữ hoa
!	Dữ liệu được ghi từ phải sang trái
\<Ký tự>	Ký tự theo sau \ sẽ được đưa thẳng vào

**Ví dụ**

Input mask	Dữ liệu nhập vào
(000)000-0000	(054)828-8282
(000)AAA-A	(123)124-E

☞ **Chú ý:** Nếu muốn các ký tự gõ vào quy định thuộc tính input mask là password (Khi nhập dữ liệu vào tại các vị trí đó xuất hiện dấu \*).

**3.2.4. Caption**

Quy định nhãn là một chuỗi ký tự sẽ xuất hiện tại dòng tiêu đề của bảng. Chuỗi ký tự này cũng xuất hiện tại nhãn các cửa các điều khiển trong các biểu mẫu hoặc báo cáo.

**3.2.5. Default value**

Quy định giá trị mặc định cho trường trừ Auto number và OEL Object

**3.2.6. Validation rule và Validation Text**

Quy định quy tắc hợp lệ dữ liệu (Validation rule) để giới hạn giá trị nhập vào cho một trường. Khi giới hạn này bị vi phạm sẽ có câu thông báo ở Validation text.

Các phép toán có thể dùng trong Validation rule

Các phép toán	Phép toán	Tác dụng
Phép so sánh	>, <, >=, <=, =, <>	
Phép toán logic	Or, and, not	Hoặc, và, phủ định
Phép toán về chuỗi	Like	Giống như

**Chú ý:** Nếu hằng trong biểu thức là kiểu ngày thì nên đặt giữa 2 dấu #.

**Ví dụ**

Validation rule	Tác dụng
<>0	Khác số không
Like “*HUE*”	Trong chuỗi phải chứa HUE
<#25/07/76#	Trước ngày 25/07/76
>=#10/10/77# and <=#12/11/77#	Trong khoảng từ 10/10/77 đến 12/12/77

### 3.2.7. Required

Có thể quy định thuộc tính này để bắt buộc hay không bắt buộc nhập dữ liệu cho trường.

Required	Tác dụng
Yes	Bắt buộc nhập dữ liệu
No	Không bắt buộc nhập dữ liệu

### 3.2.8. AllowZeroLength

Thuộc tính này cho phép quy định một trường có kiểu Text hay memo có thể hoặc không có thể có chuỗi có độ dài bằng 0.

**Chú ý:** Cần phân biệt một trường chứa giá trị null (chưa có dữ liệu) và một trường chứa chuỗi có độ dài bằng 0 (Có dữ liệu nhưng chuỗi rỗng “”).

AllowZeroLength	Tác dụng
Yes	Chấp nhận chuỗi rỗng
No	Không chấp nhận chuỗi rỗng

### 3.2.9. Index

Quy định thuộc tính này để tạo chỉ mục trên một trường. Nếu chúng ta lập chỉ mục thì việc tìm kiếm dữ liệu nhanh hơn và tiện hơn.

Index	Tác dụng
Yes( Duplicate OK)	Tạo chỉ mục có trùng lặp
Yes(No Duplicate )	Tạo chỉ mục không trùng lặp
No	Không tạo chỉ mục

### 3.2.10. New value

Thuộc tính này chỉ đối với dữ liệu kiểu auto number, quy định cách thức mà trường tự động điền số khi thêm bản ghi mới vào.

New value	Tác dụng
Increase	Tăng dần
Random	Lấy số ngẫu nhiên

## 4. THIẾT LẬP KHOÁ CHÍNH (*primary key*)

### 4.1. Khái niệm khoá chính

Sức mạnh của một Hệ QTCSDL như Microsoft Access, là khả năng mau chóng truy tìm và rút dữ liệu từ nhiều bảng khác nhau trong CSDL. Để hệ thống có thể làm được điều này một cách hiệu quả, mỗi bảng trong CSDL cần có một trường hoặc một nhóm các trường có thể xác định duy nhất một bản ghi trong số rất nhiều bản ghi đang có trong bảng. Đây thường là một mã nhận diện như Mã nhân viên hay Số Báo Danh của học sinh. Theo thuật ngữ CSDL trường này được gọi là **khóa chính** (*primary key*) của bảng. MS Access dùng trường khóa chính để kết nối dữ liệu nhanh chóng từ nhiều bảng và xuất ra kết quả yêu cầu.

Nếu trong bảng chúng ta đã có một trường sao cho ứng với mỗi trị thuộc trường đó chúng ta xác định duy nhất một bản ghi của bảng, chúng ta có thể dùng trường đó làm trường khóa của bảng. Từ đó cho ta thấy rằng tất cả các trị trong trường khóa chính phải khác nhau. Chẳng hạn đừng dùng tên người làm trường khóa vì tên trường là không duy nhất.

Nếu không tìm được mã nhận diện cho bảng nào đó, chúng ta có thể dùng một trường kiểu **Autonumber** (ví dụ Số Thứ Tự) để làm trường khóa chính.

Khi chọn trường làm khóa chính chúng ta lưu ý mấy điểm sau:

MS Access không chấp nhận các giá trị trùng nhau hay trống (null) trong trường khóa chính.

Chúng ta sẽ dùng các giá trị trong trường khóa chính để truy xuất các bản ghi trong CSDL, do đó các giá trị trong trường này không nên quá dài vì khó nhớ và khó gõ vào.


Kích thước của khóa chính ảnh hưởng đến tốc độ truy xuất CSDL. Để đạt hiệu quả tối ưu, dùng kích thước nhỏ nhất để xác định mọi giá trị cần đưa vào trường.

#### 4.2. Cách đặt khoá chính

Ta có thể tự chọn trường làm khóa chính cho bảng bằng các bước sau đây:

Mở bảng ở chế độ Design View

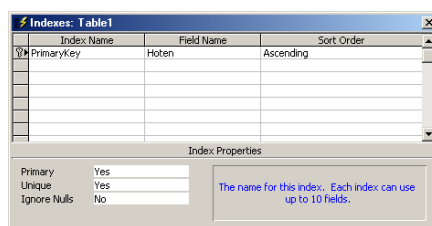
Nhấp chọn trường cần đặt

Thực hiện lệnh *Edit - Primary Key* hoặc nhấp chọn nút  trên thanh công cụ của mục này .

☞ **chú ý:** Không phải mọi trường đều có thể làm khóa chính, mà chỉ có các trường có các kiểu dữ liệu không phải là **Memo** và **OLE Object.**, **Hyper Link**.

Để hủy bỏ khóa chính hoặc các đã thiết lập thì thực hiện lệnh *View - Indexes*, trong hộp thoại này chọn và xóa đi những trường khóa đã thiết lập:





## 5. LƯU BẢNG


Sau khi thiết kế xong, ta tiến hành lưu bảng vào CSDL, có thể thực hiện một trong hai thao tác sau:

Thực hiện lệnh *File - Save*.

Nhấp chọn nút  trên thanh công cụ của mục này (Table Design)

## 6. HIỆU CHỈNH BẢNG

**6.1. Di chuyển trường:** Các thao tác để di chuyển thứ tự các trường

Đưa con trỏ ra đầu trường đến khi con trỏ chuột chuyển thành  thì nhấp chọn.

Đưa con trỏ ra đầu trường vừa chọn, nhấn và kéo đến vị trí mới.

**6.2. Chen trường :** Các thao tác lần lượt như sau

Chọn trường hiện thời là trường sẽ nằm sau trường được chen vào

Thực hiện lệnh *Insert/ Row*

**6.3. Xóa trường:** Các thao tác lần lượt như sau

Chọn trường cần xóa

Thực hiện lệnh *Edit - Delete Rows*

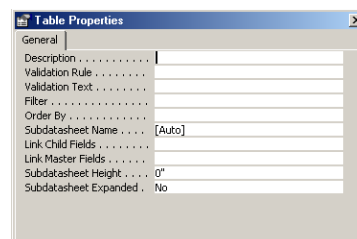
**6.4. Quy định thuộc tính của bảng**

Mở bảng ở chế độ Design View, chọn View/Properties

Description: Dòng mô tả bảng

Validation Rule: Quy tắc hợp lệ dữ liệu cho toàn bảng.

Validation Text: Thông báo lỗi khi dữ liệu không hợp lệ.





## 7. XEM THÔNG TIN VÀ BỔ SUNG BẢN GHI

### 7.1. Xem thông tin ở chế độ datasheet

Muốn xem thông tin trong một bảng chúng ta phải chuyển bảng sang một chế độ hiển thị khác gọi là *Datasheet*. Trong chế độ hiển thị này, mỗi bản ghi được thể hiện trên một hàng ngang, hàng đầu tiên là các tên trường.




Sau đây là các cách để chuyển sang chế độ hiển thị Datasheet:

Trong cửa sổ Database của CSDL đang mở, nhấp chọn tab Table. Trong mục này chọn bảng cần hiển thị rồi chọn nút **Open**, bảng sẽ được mở để bổ sung và chỉnh sửa dữ liệu.


Ta có thể chuyển sang chế độ Datasheet ngay khi đang ở trong chế Design, bằng cách nhấp chọn nút  thì bảng sẽ chuyển sang chế độ Datasheet, để quay trở về chế độ Design, ta nhấp chọn lại nút . Hoặc chọn lệnh *View - /Design View*.

### 7.2. Bổ sung bản ghi cho bảng

Sau khi hoàn thành công việc thiết kế cấu trúc bảng, ta tiến hành nhập dữ liệu, tức là bổ sung các bản ghi, cho bảng. Hiển thị bảng ở chế độ hiển thị Datasheet, mỗi hàng đại diện cho một bản ghi. Ở đây có các ký hiệu chúng ta cần biết công dụng của chúng

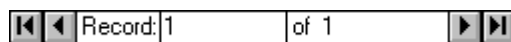
-  Bản ghi hiện thời
-  Bản ghi đang nhập dữ liệu
-  Bản ghi mới

#### 7.2.1. Bổ sung bản ghi mới cho CSDL


Đang đứng tại một bản ghi nào đó (không phải là bản ghi mới), chọn nút  trên thanh công cụ. Hoặc thực hiện lệnh *Record - Go To - New*.

#### 7.2.2. Di chuyển giữa các bản ghi

Ta có thể di chuyển qua lại giữa các bản ghi bằng cách dùng công cụ *Chọn lựa bản ghi* (Record Selector).



### 7.2.3. Nhập dữ liệu cho bản ghi

Khi đang nhập dữ liệu cho một bản ghi nào đó, thì đầu hàng của bản ghi đó xuất hiện biểu tượng .

Tổ hợp phím	Tác dụng
Tab	Sang ô kế tiếp
Shift Tab	Sang ô phía trước
Home	Đến đầu dòng
End	Đến cuối dòng
Ctrl Home	Đến bản ghi đầu tiên
Ctrl End	Đến bản ghi cuối cùng
Shift F2	Zoom

Theo chuẩn, khi nhập dữ liệu thì Access sẽ lấy font mặc định là MS San Serif, điều này có thể giúp cho ta hiển thị được tiếng Việt chỉ khi MS San Serif đó là của VietWare hoặc của ABC.

Để không phụ thuộc vào điều này, ta nên chọn font trước khi tiến hành nhập dữ liệu. Trong chế độ hiển thị Datasheet, thực hiện lệnh *Format - Font...*

Khi nhập dữ liệu là trường cho trường **OLE Object**, ta thực hiện như sau: Lệnh *Edit - Object...*

### 7.2.4. Chọn các bản ghi

Đánh dấu chọn bản ghi:

Chọn Edit/Select Record: Để chọn bản ghi hiện hành

Chọn Edit/ Select all Record để chọn toàn bộ

### 7.2.5. Xóa bản ghi

Chọn các bản ghi cần xóa, sau đó thực hiện lệnh *Edit - Delete* (hoặc nhấn phím DELETE) .

## 8. THIẾT LẬP QUAN HỆ GIỮA CÁC BẢNG

### 8.1. Các loại quan hệ trong cơ sở dữ liệu ACCESS

#### 8.1.1. Quan hệ một - một (1-1)

Trong quan hệ một - một, mỗi bản ghi trong bảng A có tương ứng với một bản ghi trong bảng B và ngược lại mỗi bản ghi trong bảng B có tương ứng duy nhất một bản ghi trong bảng A.

*Ví dụ:* Cho 2 bảng dữ liệu

Bảng **Danh sach**(**Masv**, ten, Ngaysinh, gioitinh) và bảng **Diem thi**(**Masv**, diem)

Ten	Ngaysinh	Gioitinh	Masv		Masv	diem
An	20/10/77	Yes	A001	←→	A001	9
Bình	21/07/80	No	A002	←→	A002	7
Thủy	02/12/77	Yes	A003	←→	A003	9
Lan	03/04/80	No	A004	←→	A004	4
Hồng	12/11/77	No	A005	←→	A005	5

Bảng **Danh sach** và **diem thi** có mối quan hệ **1-1** dựa trên trường **Masv**.

#### 8.1.2. Quan hệ một nhiều (1-∞)

Là mối quan hệ phổ biến nhất trong CSDL, trong quan hệ một nhiều : Một bản ghi trong bảng A sẽ có thể có nhiều bản ghi tương ứng trong bảng B, nhưng ngược lại một bản ghi trong bảng B có duy nhất một bản ghi tương ứng trong bảng A.

**Ví dụ:** Trong một khoa của một trường học nào đó có nhiều sinh viên, nhưng một sinh viên thuộc một khoa nhất định. Ta có 2 bảng dữ liệu như sau:

Bảng Danhsachkhoa(**Makhoa**, tenkhoa, sodthoi)

Bảng danhsachsv(Makhoa, Ten, Quequan, lop)

Tenkhoa	Sodthoi	Makhoa	Makhoa	Ten	Quequan	Lop
CNTT	826767	01	01	Thanh	Huế	K23
TOÁN	878787	02	01	Tùng	Qbình	K24
LÝ	868785	03	02	Thúy	Huế	K25
			02	Hùng	ĐN	K26
			03	Lan	Huế	K25
			03	Hương	ĐN	K26

Bảng Danhsachkhoa và bảng danhsachsv có mối quan hệ **1-∞** dựa trên trường Makhoa.

### 8.3. Quan hệ nhiều nhiều(∞-∞)

Trong quan hệ nhiều nhiều, mỗi bản ghi trong bảng A có thể có không hoặc nhiều bản ghi trong bảng B và ngược lại mỗi bản ghi trong bảng B có thể có không hoặc nhiều bản ghi trong bảng A.

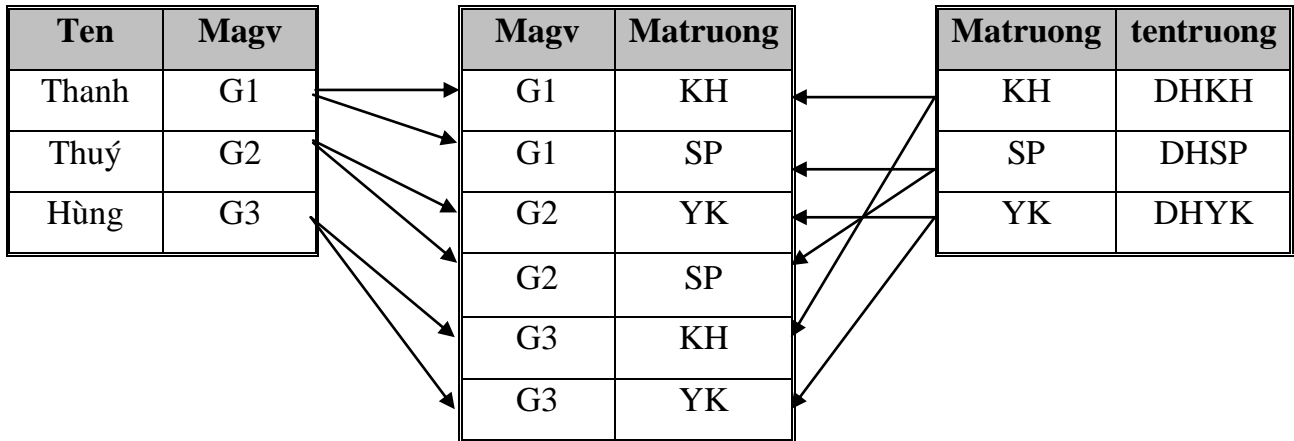
Khi gặp mối quan hệ nhiều- nhiều để không gây nên sự trùng lặp và dư thừa dữ liệu thì người ta tách quan hệ nhiều-nhiều thành 2 quan hệ một-nhiều bằng cách tạo ra một bảng phụ chứa khóa chính của 2 bảng đó.

**Ví dụ:** Một giáo viên có thể dạy cho nhiều trường và một trường có nhiều giáo viên tham gia giảng dạy. Đây là một mối quan hệ nhiều-nhiều

Bảng Danhsachgv(**Magv**,ten)

Bảng Danhsachtruong(**Matruong**, Tentruong)

Tạo ra bảng Phancongday(**Magv**, **matruong**)

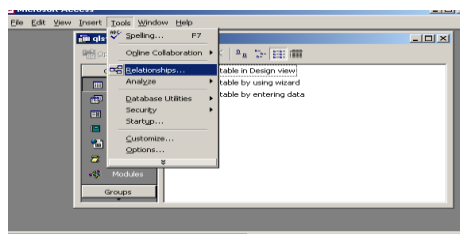


Bảng **Danhsachgv** và bảng **Phancongday** có mối quan hệ **1-∞** dựa trên trường **Magv**.

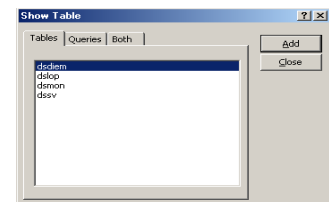
Bảng **Danhsachtruong** và bảng **Phancongday** có mối quan hệ **1-∞** dựa trên trường **Matruong**.

#### 8.4. Thiết lập mối quan hệ giữa các bảng dữ liệu (Relationships)

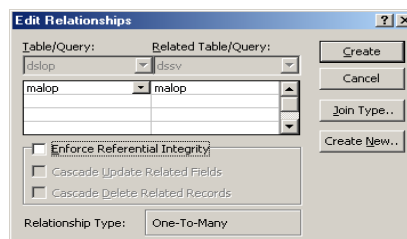
Tại cửa sổ Database, thực hiện lệnh Tools/Relationship



Trong cửa sổ Show bảng cần thiết lập quan hệ, sau đó chọn **Add** và **Close**.



Kéo trường liên kết của bảng quan hệ vào trường của bảng được quan hệ (Table related).



Bật chức năng Enforce Referential Integrity ( Nếu muốn quan hệ này bị ràng buộc tham chiếu toàn vẹn), chọn mỗi quan hệ (one-many) hoặc (one-one).

Chọn nút **Create**.

### **☞ Chú ý**

Quan hệ có tính tham chiếu toàn vẹn sẽ đảm bảo các vấn đề sau:

Khi nhập dữ liệu cho trường tham gia quan hệ ở bên nhiều thì phải tồn tại bên một.

Không thể xoá một bản ghi của bảng bên một nếu trong quan hệ đã tồn tại những bản ghi bên nhiều có quan hệ với bản ghi bên một đó.

Trường hợp vi phạm các quy tắc trên thì sẽ nhận được thông báo lỗi.

#### **8.4.1. Thiết lập thuộc tính tham chiếu toàn vẹn trong quan hệ**

Trong khi chọn mỗi quan hệ giữa các bảng, có 2 thuộc tính tham chiếu toàn vẹn đó là *Cascade update related fields*, *Cascade Delete related records*, có thiết lập 2 thuộc tính này.

Nếu chọn thuộc tính *Cascade update related fields*, khi dữ liệu trên khoá chính của bảng bên một thay đổi thì Access sẽ tự động cập nhật sự thay đổi đó vào các trường tương ứng (có quan hệ) trên các bảng bên Nhiều, hay nói cách khác, dữ liệu ở bảng bên nhiều cũng thay đổi theo.

Nếu chọn thuộc tính *Cascade Delete related records*, khi dữ liệu trên bảng bên một bị xoá thì dữ liệu trên bảng bên nhiều cũng sẽ bị xoá..

#### **8.4.2. Kiểu kết nối (Join type)**

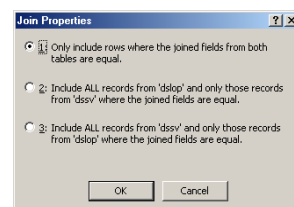
Trong quá trình thiết lập quan hệ giữa các bảng, nếu không chọn nút Create, chọn nút join type để chọn kiểu liên kết

Mục 1: Liên kết nội (Inner join)

Mục 2 và mục 3 là liên kết ngoại (outer join)

#### **8.4.3. Điều chỉnh các mối quan hệ**

Mở cửa sổ quan hệ (Tools/Relationship)



Click chuột phải, chọn edit relationship

#### **8.4.4. Xoá các mối quan hệ**

Mở cửa sổ quan hệ (Tools/Relationship)

Chọn mối quan hệ giữa các bảng, nhấn delete.

### **9. SẮP XẾP VÀ LỌC DỮ LIỆU**

#### **9.1. Một số phép toán và hàm**

##### **9.1.1. Một số phép toán**

Ký tự thay thế:       ? : Thay thế cho một ký tự bất kỳ

Ký tự                   \* : Thay thế cho một dãy các ký tự.

Phép toán   **Like**: Giống như

**In**: Kiểm tra một giá trị có thuộc một tập các giá trị hay không?

**Is Null**: Giá trị của một trường là Null.

**Is not Null**: Giá trị của một trường là không Null.

**Between.....and**: Kiểm tra xem một giá trị có thuộc một "đoạn" nào đó hay không?

##### **9.1.2. Một số hàm**

Hàm **Left\$(<Chuỗi>,<n>)**: Trích bên trái chuỗi n ký tự.

Hàm **Right\$(<Chuỗi>,<n>)**: Trích bên phải chuỗi n ký tự.

Hàm **Ucase(<Chuỗi>)**: Trả lại một chuỗi in hoa.

Hàm **Lcase(<Chuỗi>)**: Trả lại một chuỗi in thường.

Hàm **IIF(<Điều kiện>,<Giá trị 1>,<Giá trị 2>)**: Nếu <Điều kiện> nhận giá trị true thì hàm trả lại <Giá trị 1>, ngược lại hàm trả lại <Giá trị 2>.

#### **9.2. Sắp xếp dữ liệu**

##### **9.2.1. sắp xếp trên một trường**

Đặt con trỏ tại trường cần sắp xếp

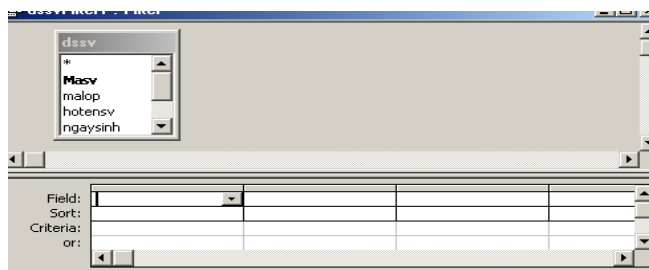
Thực hiện lệnh Records/ Sort/ Sort Ascending (Nếu sắp xếp tăng dần)

/ Sort Descending (Nếu sắp xếp giảm dần)

##### **9.2.2. sắp xếp trên nhiều trường**



Mở bảng trong chế độ Datasheet rồi thực hiện lệnh  
Records/ Filter/ Advanced Filter sort



Trong hàng Field: Chọn các trường cần sắp xếp (Thứ tự ưu tiên từ trái sang phải)

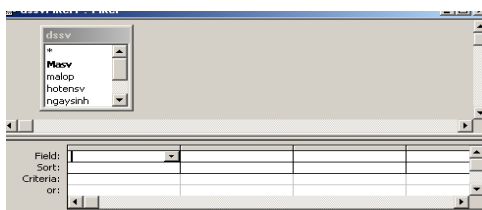
Trong hàng Sort: Chọn tiêu chuẩn sắp xếp.

Trong hàng Criteria: Chọn điều kiện sắp xếp (nếu có).

Xem kết quả.: Chọn Filter/Apply filter.

### 9.3. Lọc dữ liệu

Mở bảng trong chế độ Datasheet rồi thực hiện lệnh  
Records/ Filter/ Advanced Filter sort



Trong hàng Field: Chọn các trường làm tiêu chuẩn lọc dữ liệu

Trong hàng Criterie: Chọn tiêu chuẩn lọc dữ liệu.

Thực hiện lệnh Filter/ Apply filter sort để xem kết quả..

Sức mạnh thực sự của CSDL là khả năng tìm đúng và đầy đủ thông tin mà chúng ta cần biết, trình bày dữ liệu sắp xếp theo ý muốn. Để đáp ứng yêu cầu trên, Access cung cấp một công cụ truy vấn cho phép đặt câu hỏi với dữ liệu đang chứa bên trong các bảng trong CSDL.

### 1. KHÁI NIỆM TRUY VẤN

Truy vấn là một công cụ cho phép đặt câu hỏi với dữ liệu trong bảng dữ liệu trong CSDL.

Loại truy vấn thông dụng nhất là truy vấn chọn (Select Query). Với kiểu truy vấn này chúng ta có thể xem xét dữ liệu trong các bảng, thực hiện phân tích và chỉnh sửa trên dữ liệu đó, có thể xem thông tin từ 1 bảng hoặc có thể thêm nhiều trường từ nhiều bảng khác nhau.

#### *Ví dụ:*

Cho 2 bảng dữ liệu KHOHANG (MAHANG, TENHANG, GIA)  
BANHANG(MAHANG, TENKHACH, SOLUONG, NGAYMUA). Hãy hiển thị những khách hàng mua hàng trong tháng 7 bao gồm các thông tin: MAHANG, TENHANG, GIA, TENKHACH.

Sau khi thực hiện truy vấn, dữ liệu thỏa mãn yêu cầu được rút ra và tập hợp vào một bảng kết quả gọi là **Dynaset** (Dynamic set). Dynaset cũng hoạt động như 1 bảng (Table) nhưng nó không phải là bảng vfa kết quả khi hiển thị có thể cho phép sửa đổi.

Một loại bảng thể hiện kết quả truy vấn khác là **Snapshot**, nó tương tự như dynaset tuy nhiên không thể sửa đổi thông tin ( Như truy vấn Crosstab....).

#### 1.1. Các loại truy vấn trong Access

Select Query : Truy vấn chọn

Crosstab Query : Truy vấn tham khóa chéo (Thể hiện dòng và cột)

Action Query : Truy vấn hành động gồm

Truy vấn tạo bảng (make table Query )

Truy vấn nối (append Query )

Truy vấn cập nhật ( Update Query )

Truy vấn xóa dữ liệu ( Delete Query )

SQL Query : Truy vấn được viết bởi ngôn ngữ SQL.

Pass throught Query : Gởi các lệnh đến một CSDL SQL như Microsoft SQL server.

## 1.2. Sự cần thiết của truy vấn

Khi đứng trước một vấn đề nào đó trong CSDL, nếu sử dụng công cụ truy vấn thì có thể thực hiện được các yêu cầu sau:

Sự lựa chọn các trường cần thiết.

Lựa chọn những bản ghi.

Sắp xếp thứ tự các bản ghi.

Lấy dữ liệu chứa trên nhiều bảng khác nhau trong CSDL.

Thực hiện các phép tính.

Sử dụng truy vấn làm nguồn dữ liệu cho một biểu mẫu (Form), báo cáo (report) hoặc một truy vấn khác (Query ).

Thay đổi dữ liệu trong bảng.

## 2. CÁC CHẾ ĐỘ HIỂN THỊ TRUY VẤN

### 2.1. Cửa sổ thiết kế truy vấn (Design view).

Trong chế độ này, người sử dụng có thể tạo, sửa chữa một truy vấn nào đó. Màn hình truy vấn chứa hai phần, phần thứ nhất chứa các bảng (hoặc truy vấn) tham gia truy vấn, phần thứ hai gọi là vùng lưới QBE (Query By Example).

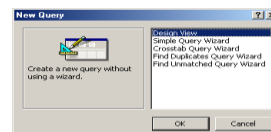
### 2.2. Cửa sổ hiển thị truy vấn (DataSheet view).

Sử dụng chế độ này để xem kết quả.

### 2.3. Cửa sổ lệnh SQL (SQL view).

Sử dụng chế độ này để xem mã lệnh của truy vấn đang tạo

## 3. TẠO TRUY VẤN

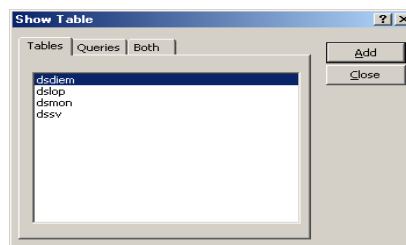


### 3.1. Tạo mới 1 truy vấn

Từ cửa sổ Database, click vào đối tượng Queries.

Chọn nút New.

Chọn Design View, chọn OK



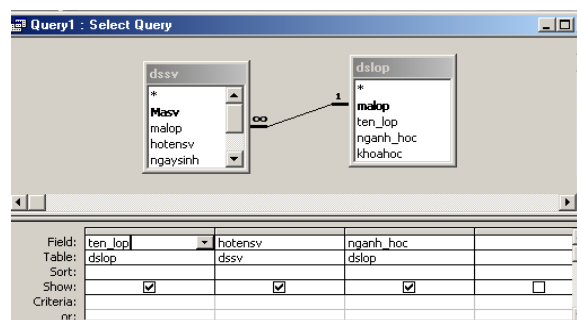
Trong bảng Show Table chọn tables để hiển thị các bảng, chọn các bảng tham gia vào truy vấn và nhấn nút **Add**, sau đó nhấn **Close**. (Nếu chọn Queries thì hiển thị truy vấn, chọn both thì hiển thị cả truy vấn và bảng dữ liệu).

Đưa các trường từ các bảng vào tham gia truy vấn bằng cách kéo các trường và thả vào hàng **Field** trong vùng lưới QBE.

Trong hàng Sort: Sắp xếp dữ liệu (nếu có)

Trong hàng **Criteria** đặt tiêu chuẩn (nếu có)

Lưu truy vấn.



#### *☞ Chú ý*

Mỗi truy vấn có:

Tối đa là 32 bảng tham gia.

Tối đa là 255 trường.

Kích thước tối đa của bảng dữ liệu (do truy vấn tạo ra) là 1 gigabyte.

Số trường dùng làm khóa sắp xếp tối đa là 10.

Số truy vấn lồng nhau tối đa là 50 cấp.

Số ký tự tối đa trong ô của vùng lưới là 1024.

Số ký tự tối đa trong dòng lệnh SQL là 64000.

Số ký tự tối đa trong tham số là 255.

### 3.2. Thay đổi thứ tự, xóa các trường

Các trường trong truy vấn sẽ hiển thị theo thứ tự như xuất hiện trong vùng lưới QBE.

#### 3.2.1. Thay đổi thứ tự của trường

Đưa con trỏ vào thanh chọn sao cho con trỏ biến thành hình mũi tên xuống

Click để chọn trường

Drag để thay đổi vị trí.

### 3.2.2.xoá trường

Đưa con trỏ vào thanh chọn sao cho con trỏ biến thành hình mũi tên xuống

Click để chọn trường

Nhấn phím delete (Nếu muốn xoá tất cả các trường trong vùng lưới QBE: chọn Edit/clear grid)

### 3.3. Thể hiện hoặc che dấu tên bảng trong vùng lưới QBE

Muốn biết tên trường hiện tại trong vùng lưới QBE là của tên bảng nào, tại chế độ Design View người sử dụng thực hiện View/tables name.

### 3.4. Xem kết quả của truy vấn.

Tại cửa sổ Database chọn tên truy vấn rồi chọn Open, hoặc trong khi thiết kế truy vấn thực hiện lệnh View/datasheet View.

### 3.5. Đổi tiêu đề cột trong truy vấn.

Đổi tên tiêu đề cột trong truy vấn mục đích là làm cho bảng kết xuất dễ đọc hơn (Trừ khi đã quy định thuộc tính Caption).

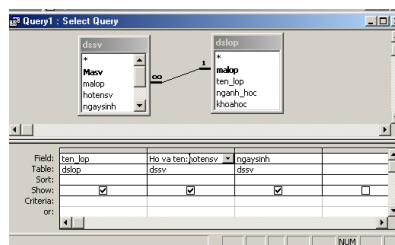
Muốn đổi tên tiêu đề cột thực hiện theo các bước sau:

Mở truy vấn ở chế độ Design View

Chọn vào bên trái ký tự đầu tiên của tên trường trong vùng lưới QBE

Gõ vào tên mới, theo sau là dấu 2 chấm (:).

*Ví dụ:*



ten_lop	Ho va ten	ngaysinh
Tin K23 A	Nguyễn Tèo	10-10-78
Tin K23 B	Trần Bình Chi	05-05-79
Tin K24A	Nguyễn Thanh Thuý	12-10-79
Tin K24A	Lê Mậu Ngọ	08-08-79
Tin K24B	Võ Tông	02-02-79
Tin K25 A	Nguyễn Khánh	10-05-75
Tin K25 A	Nguyễn Tiểu	10-10-78
Tin K25 A	Trần Bạch đặng	20-10-79
Tin K25 B	Lê Thuý Thu	03-03-79
Tin K25 B	Trần Thanh	25-07-77

### 3.6. Định thứ tự sắp xếp

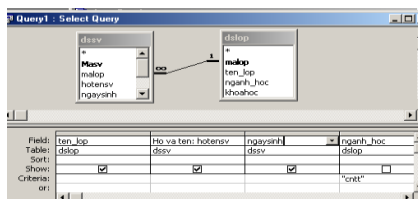
Có thể sử dụng nhanh trong chế độ datasheet View .

Có thể tạo sắp xếp trong khi thiết kế truy vấn bằng cách chọn Ascending (tăng dần) hoặc Descending (giảm dần) trong hàng Sort của vùng lưới QBE .

☞ **Chú ý:** Nếu có nhiều trường định vị sắp xếp thì theo thứ tự ưu tiên từ trái sang phải.

### 3.7. Che dấu hay thể hiện các trường trong Dynaset

Tại hàng Show ứng với trường cần che dấu chúng ta không chọn mặt dù nó vẫn tồn tại, vẫn tham gia truy vấn.



ten_lop	Ho va ten	ngaysinh
Tin K24B	Vũ Tông	02-02-79
Tin K25 A	Nguyễn Khánh	10-05-75
Tin K25 A	Nguyễn Tiểu	10-10-78
Tin K25 A	Trần Bạch đặng	20-10-79
Tin K25 B	Lê Thủy Thu	03-03-79
Tin K25 B	Trần Thanh	25-07-77

### 3.8. Mối quan hệ giữa thuộc tính của trường trong truy vấn và trong bảng dữ liệu

Theo mặc nhiên, các trường trong truy vấn kế thừa tất cả các thuộc tính của trường trong bảng làm nguồn dữ liệu. Nếu không quy định lại trong truy vấn, các trường trong Dynaset hoặc snapshot luôn kế thừa các thuộc tính của bảng làm nguồn dữ liệu. Nếu thay đổi thiết kế trong bảng làm nguồn dữ liệu và thay đổi thuộc tính của các trường thì thuộc tính này cũng được thay đổi trong truy vấn. Tuy nhiên, nếu quy định lại các thuộc tính cho các trường trong truy vấn thì các thuộc tính của các trường trong bảng làm nguồn dữ liệu không thay đổi.

## 4. THIẾT KẾ TRUY VẤN CHỌN

### 4.1. Định nghĩa truy vấn chọn

Truy vấn chọn là loại truy vấn được chọn lựa, rút trích dữ liệu từ các bảng dữ liệu thỏa mãn một hoặc nhiều điều kiện nào đó. Khi thực hiện truy vấn chọn, Access tác động lên dữ liệu và thể hiện các bản ghi thỏa mãn các điều kiện đặt ra trong một bảng kết quả gọi là Recordset.

### 4.2. Lập phép chọn trong truy vấn

#### 4.2.1. Chọn một nhóm các bản ghi thỏa mãn một điều kiện nào đó

Muốn thực hiện các phép chọn trong khi thể hiện truy vấn người ta thường sử dụng các phép toán sau:

Phép toán	Ví dụ	Ý nghĩa
<	<#20/10/99#	Trước ngày 20/10/99
>	>#10/10/98#	Sau ngày 10/10/98
>=	>= #05/05/90#	Sau và trong ngày 05/05/90
<>	<>#01/01/99#	Khác ngày 01/01/99
=	= #10/10/97#	Trong ngày 10/10/97
Between .... and	Between #1/2/97# and #1/7/97#	Từ ngày 1/2/97 đến 1/7/97
...		

**Ví dụ:**

Cho 2 bảng dữ liệu Dslop(**Malop**, Tenlop, Ngành\_hoc, khoaoc)

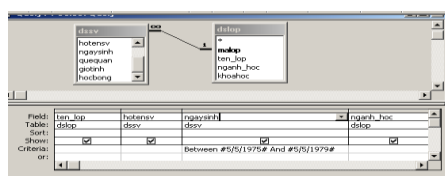
Dssv(Masv, malop, hotensv, ngaysinh, quequan, giotinh, hocbong)

Tạo một truy vấn để hiển thị danh sách những sinh viên có ngaysinh trong khoảng thời gian từ 05/05/75 đến 05/05/79 bao gồm các trường: Tenlop, Hotensv, Ngaysinh, ngành\_hoc.

Tạo truy vấn chọn và đưa 2 bảng dslop và dssv vào tham gia truy vấn

Đưa các trường Tenlop, hotensv, ngaysinh, ngành\_hoc vào vùng lưới QBE

Trong hàng Criteria của trường Ngaysinh: Between #05/05/75# and #05/05/79#



ten_lop	hotensv	ngaysinh	nganh_hoc
Tin K23 A	Nguyễn Téo	10-10-78	Tin hoc
Tin K23 B	Trần Bình Chi	05-05-79	Tin hoc
Tin K24 B	Võ Tông	02-02-79	CNTT
Tin K25 A	Nguyễn Khánh	10-05-75	CNTT
Tin K25 B	Lê Thuý Thu	03-03-79	CNTT
Tin K25 B	Trần Thanh	25-07-77	CNTT
Tin K25 A	Nguyễn Tiểu	10-10-78	CNTT

#### 4.2.2. Ký tự thay thế

Ký tự \* : Thay thế một nhóm ký tự bất kỳ.

Ký tự ? : Thay thế 1 ký tự.

Ký tự [ ] : Thay thế các ký tự trong ngoặc vuông.

Ký tự ! : Phủ định.

Ký tự - : Từ ký tự đến ký tự.

**Ví dụ**

Cho 2 bảng dữ liệu Dslop(**Malop**, Tenlop, Ngành\_hoc, khoaoc)

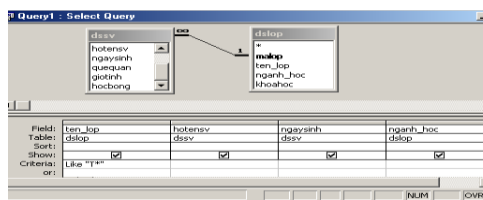
Dssv(Masv, malop, hotensv, ngaysinh, quequan, giotinh, hocbong)

Tạo một truy vấn để hiển thị danh sách những sinh viên có Tenlop bắt đầu là "T" bao gồm các trường: Tenlop, Hotensv, Ngaysinh, ngành\_hoc.

Tạo truy vấn chọn và đưa 2 bảng dslop và dssv vào tham gia truy vấn

Đưa các trường Tenlop, hotensv, ngaysinh, ngành\_hoc vào vùng lưới QBE

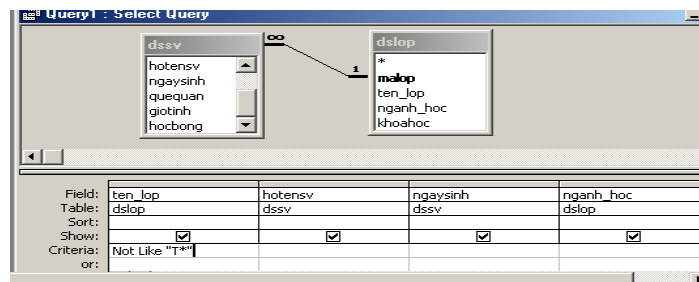
Trong hàng Criteria của trường Tenlop: Like "T\*"



**4.2.3. Chọn các bản ghi không phù hợp với một giá trị nào đó**

Dùng toán tử **Not**

**Ví dụ:** Tạo một truy vấn để hiển thị danh sách những sinh viên có Tenlop không bắt đầu là "T" bao gồm các trường: Tenlop, Hotensv, Ngaysinh, ngành\_hoc.



**4.2.4. Định nhiều tiêu chuẩn trong lựa chọn**

Dùng phép "Và" và phép "Hoặc" trong một trường

Muốn quy định nhiều tiêu chuẩn trong cùng một trường, chúng ta phải sử dụng toán tử AND (và ) cùng toán tử OR (hoặc).

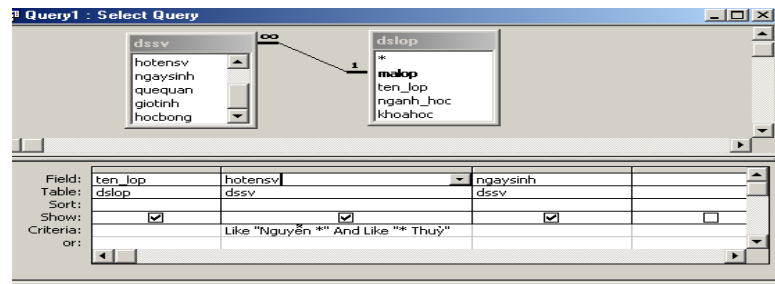
**Ví dụ:** Tạo một truy vấn để hiển thị danh sách những sinh viên có Họ là "Nguyễn" và Tên "Thuỷ" bao gồm các trường: Tenlop, Hotensv, Ngaysinh.

Tạo truy vấn chọn và đưa 2 bảng dslop và dssv vào tham gia truy vấn



Đưa các trường Tenlop, hotensv, ngaysinh vào vùng lưới QBE

Trong hàng Criteria của trường Hotensv : Like "Nguyễn \*" and "\* Thuỷ"



*Dùng phép “Và” và phép “Hoặc” trên nhiều trường*

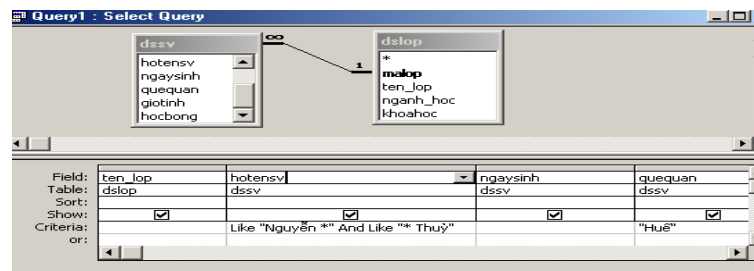
**Ví dụ:** Tạo một truy vấn để hiển thị danh sách những sinh viên có Họ là "**Nguyễn**" và tên "**Thuỷ**" và có quê quán ở "**Huế**" bao gồm các trường: Tenlop, Hotensv, ngaysinh Quequan.

Tạo truy vấn chọn và đưa 2 bảng dslop và dsav vào tham gia truy vấn

Đưa các trường Tenlop, hotensv, ngaysinh, quequan vào vùng lưới QBE

Trong hàng Criteria của trường Hotensv : Like "Nguyễn \*" and "\* Thuỷ"

Quequan : Huế



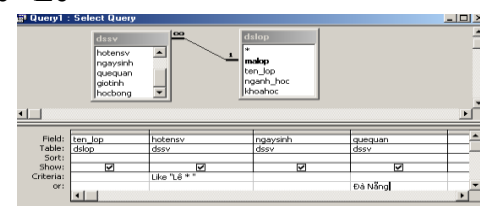
Tạo một truy vấn để hiển thị danh sách những sinh viên có Họ là "**Lê**" hoặc có quê quán ở "**Đà Nẵng**" bao gồm các trường: Tenlop, Hotensv, Ngaysinh, Quequan.

Tạo truy vấn chọn và đưa 2 bảng dslop và dsav vào tham gia truy vấn

Đưa các trường Tenlop, hotensv, ngaysinh , Quequan vào vùng lưới QBE

Trong hàng Criteria của trường Hotensv : Like "Lê \*"

Trong hàng or của trường Quequan : Đà Nẵng



#### 4.2.5. Chọn các bản ghi có chứa có giá trị

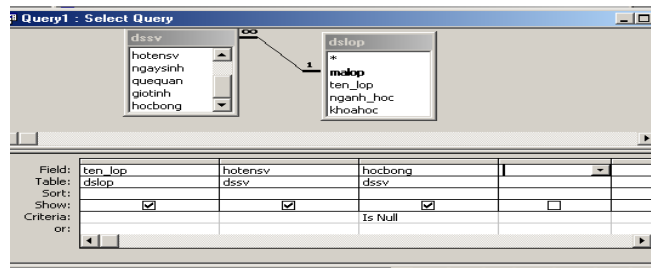
Chúng ta có thể chọn các bản ghi có chứa hoặc không chứa giá trị, chẳng hạn như tìm những sinh viên mà không có số điện thoại nhà ở.

Access cung cấp 2 phép toán

Phép toán	Ý nghĩa
IS NULL	Trường không chứa giá trị
IS NOT NULL	Trường có chứa giá trị

*Ví dụ:*

Tạo một truy vấn để hiển thị danh sách những sinh viên không có học bổng bao gồm các trường: Tenlop, Hotensv, hocbong.

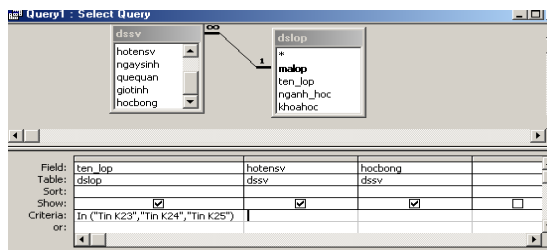


#### 4.2.6. Chọn các bản ghi thuộc danh sách các giá trị nào đó

Chúng ta có thể sử dụng phép toán **IN** (*Danh sách giá trị*)

*Ví dụ*

Tạo một truy vấn để hiển thị danh sách những sinh viên thuộc lớp "Tin K23" hoặc "Tin K24" hoặc "Tin K25" bao gồm các trường: Tenlop, Hotensv, hocbong.



#### 4.2.7. Tham chiếu đến các trường khác

Nếu trong biểu thức chọn của truy vấn, các tính toán tham chiếu đến các trường phải đặt trong dấu [ ], trường hợp tham chiếu đến trường của bảng khác phải chỉ rõ bảng nguồn của nó. [Tên bảng].[Tên trường].

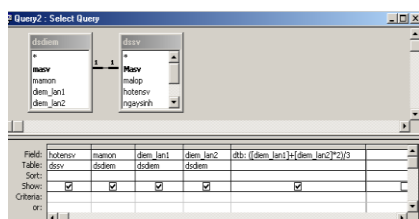
#### 4.2.8. Tạo trường kiểu biểu thức

##### Ví dụ

Cho 2 bảng dữ liệu Dssv( **Masv**, hotensv, ngaysinh, quequan, gioitinh)

Dsdiem( **Masv**, mamon, diem\_lan1, diem\_lan2)

Tạo truy vấn để hiển thị các thông tin: Hotensv, mamon, diem\_lan1, diem\_lan2, dtb, trong đó  $dtb = (diem\_lan1 + diem\_lan2 * 2) / 3$ .



hotensv	mamon	diem_lan1	diem_lan2	dtb
Nguyễn Tèo	01	10	10	10.00
Tần Bình Chi	02	8	3	4.67
Nguyễn Thanh Thuý	01	7	8	7.67
Võ Tung	03	9	8	8.33
Lê Mậu Ngọc	02	3	4	3.67
Nguyễn Khánh	03	6	7	6.67
Lê Thuý Thu	01	4	5	4.67
Tần Thanh	02	6	7	6.67
Nguyễn Tiến	02	5	9	7.67
Tần Bách Giang	03	3	9	7.00

##### ☞ Chú ý

Sau khi thực hiện truy vấn chúng ta không thể thay đổi giá trị trong trường kiểu biểu thức, tuy nhiên nếu thay đổi giá trị trong trường tham gia biểu thức thì kết quả trong trường kiểu biểu thức cũng thay đổi theo.

#### 4.2.9. Chọn giá trị duy nhất

Theo mặc định, access sẽ chọn tất cả các bản ghi thỏa mãn điều kiện, tuy nhiên đôi khi có nhiều giá trị giống nhau được lặp đi lặp lại, do đó để cô đọng dữ liệu thì chúng ta có thể quy định thuộc tính duy nhất trong khi hiển thị

##### Thuộc tính Unique-values

Chọn Yes: Không thể hiện các giá trị trùng nhau

Chọn No: Thể hiện các giá trị trùng nhau

##### Thuộc tính Unique-Records

Chọn Yes: Không thể hiện các bản ghi trùng nhau

Chọn No: Thể hiện các bản ghi trùng nhau

#### 4.2.10. Chọn các giá trị đầu

Khi hiển thị truy vấn đôi khi chúng ta muốn hiển thị một số bản ghi đầu tiên nào đó thoả mãn các điều kiện thì sử dụng thuộc tính **Top values**.

## 5. TRUY VẤN DỰA TRÊN NHIỀU BẢNG DỮ LIỆU

Để tạo truy vấn dựa trên nhiều bảng dữ liệu thì các bảng đó phải được thiết lập mối quan hệ, nếu các bảng không thiết lập mối quan hệ thì khi truy vấn dữ liệu access sẽ cho ra những bộ dữ liệu là tích Đề-Các giữa các bộ dữ liệu trong các bảng.

### 5.1. Liên kết các bảng trong truy vấn

Khi các bảng dữ liệu được thiết lập mối quan hệ thì trường nối với nhau gọi là trường liên kết, trong access phân biệt 3 loại liên kết sau

#### 5.1.1. Liên kết nội (Inner join)

Đây là loại liên kết rất phổ biến nhất giữa 2 bảng dữ liệu. Trong đó dữ liệu khi thể hiện trên Dynaset sẽ gồm những bản ghi mà dữ liệu chứa trong trường liên kết ở hai bảng phải giống nhau hoàn toàn.

#### 5.1.2. Liên kết ngoại (Outer join)

Đây là loại liên kết cho phép dữ liệu thể hiện trên Dynaset của một trong hai bảng tham gia có nội dung trường liên kết không giống nội dung trong trường tương ứng của bảng còn lại. Liên kết ngoại được chia làm hai loại

**Left Outer Join:** Trong kiểu liên kết này, dữ liệu ở bảng bên "1" thể hiện toàn bộ trên Dynaset và chỉ những bản ghi bên bảng "nhiều" có nội dung trong trường liên kết giống trường tương ứng bên bảng "1".

**Right Outer Join:** Trong kiểu liên kết này, dữ liệu ở bảng bên "nhiều" thể hiện toàn bộ trên Dynaset và chỉ những bản ghi bên bảng "1" có nội dung trong trường liên kết giống trường tương ứng bên bảng "nhiều".

#### 5.1.3. Tự liên kết (Self join)

Là kiểu liên kết của một bảng dữ liệu với chính nó. Trong đó một bản ghi trong bảng dữ liệu sẽ liên kết với những bản ghi khác trong bảng dữ liệu đó. Tự liên kết có thể hiểu như là liên kết nội hay liên kết ngoại từ một bảng vào một bảng sao chính nó. Để thực hiện việc tạo tự liên kết chúng ta phải đưa một bảng vào tham gia truy vấn 2 lần.

## 5.2. Tạo liên kết ngoại

Muốn tạo liên kết ngoại giữa 2 bảng dữ liệu ta thực hiện

Tools/Relationships

Double click vào đường liên kết giữa 2 bảng dữ liệu, chọn Join Type



Trong hộp thoại Join Properties chọn mục 2 hoặc mục 3.

## 5.3. Tạo một tự liên kết

Để tạo một tự liên kết chúng ta thực hiện

Tạo truy vấn mới và đưa bảng dữ liệu vào truy vấn 2 lần

Tạo các liên kết

**Ví dụ:**

Cho bảng Dsdiem(Masv, Hoten, Diem\_lan1, Diem\_lan2)

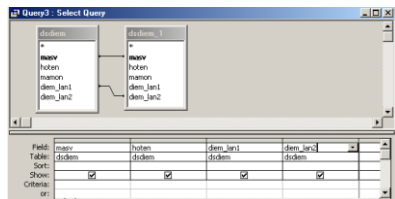
Tạo truy vấn để hiển thị danh sách những sinh viên có điểm thi Lần 1 bằng điểm thi lần 2...

Tạo truy vấn mới và đưa bảng Dsdiem vào tham gia truy vấn 2 lần

Tạo liên kết nội (Inner join) giữa 2 trường Masv

Tạo liên kết nội từ trường Diem\_lan1 vào Diem\_lan2

Đưa các trường vào vùng lưới QBE và xem kết quả



masv	hoten	diem_lan1	diem_lan2
a001	Lan Anh	10	10
a008	Hong Ngu	6	6

Record: 1 of 2

## 5.3. Tự động tìm kiếm dữ liệu (Auto lookup)

Khi nhập dữ liệu vào Dynaset, chức năng tự động tìm kiếm dữ liệu cho phép chúng ta chỉ nhập dữ liệu ở các trường của bảng bên "nhiều" (Ở quan hệ 1-∞) còn Access sẽ tự động tìm kiếm dữ liệu tương ứng trên bảng "1" để hiển thị.

Chức năng Auto Lookup hoạt động trong các truy vấn mà hai bảng tham gia có mối quan hệ 1-∞.

Tạo một truy vấn có sử dụng chức năng Auto Lookup chúng ta thực hiện:

Tạo truy vấn và đưa 2 bảng vào tham gia truy vấn

Đưa trường liên kết của bảng bên nhiều vào vùng lưới QBE

Đưa các trường cần hiển thị dữ liệu của bảng bên "1".

☞ **Chú ý:** Khi nhập dữ liệu chỉ nhập dữ liệu ở các trường của bảng "nhiều"

## 6. TÍNH TỔNG TRONG TRUY VẤN CHỌN

Trong thực tế, chúng ta thường có những câu hỏi đặt ra về việc nhóm dữ liệu nào đó, chẳng hạn trong tháng 10 công ty xăng dầu XYZ bán được bao nhiêu lít xăng, tổng thành tiền bao nhiêu?

Trong Access chúng ta có thể thực hiện một số phép tính lên một nhóm bản ghi bằng cách dùng truy vấn tính tổng

### Một số phép toán thường sử dụng

Phép toán	Ý nghĩa
<i>Sum</i>	Tính tổng các giá trị của một trường
<i>Avg</i>	Tính giá trị trung bình của một trường
<i>Min</i>	Tính giá trị nhỏ nhất của một trường
<i>Max</i>	Tính giá trị lớn nhất của một trường
<i>Count</i>	Đếm số giá trị khác rỗng có trong một trường
<i>First</i>	Giá trị của trường ở bản ghi đầu tiên trong bảng
<i>Last</i>	Giá trị của trường ở bản ghi cuối cùng trong bảng
<i>Where</i>	Giới hạn điều kiện khi tính tổng
<i>Expression</i>	Trường kiểu biểu thức

☞ **Chú ý:** Khi thực hiện truy vấn Total, dữ liệu trong bảng kết quả của nó trình bày không thể chỉnh sửa.

### 6.1. Tạo truy vấn tính tổng

Tạo truy vấn chọn và đưa các bảng vào tham gia truy vấn

Thực hiện lệnh: View/Totals

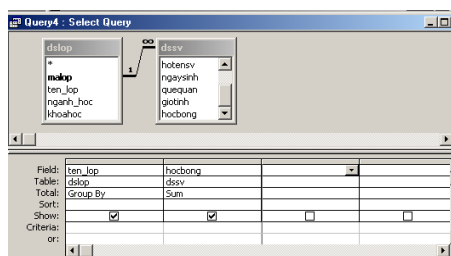
Trong vùng lưới QBE:

Tại hàng      Field chọn các trường

Total chọn các phép toán tương ứng.

Criteria: Chọn điều kiện giới hạn tính tổng (Nếu có)

Lưu và thực hiện truy vấn



### 6.2. Tính tổng của tất cả các bản ghi

Tạo truy vấn chọn.

Đưa các bảng cần thiết vào truy vấn.

Đưa các trường cần thiết vào vùng lưới QBE

Chọn menu View/Totals, dòng Total sẽ xuất hiện trên vùng lưới.

Trong hàng total của mỗi trường chọn phương pháp tính tổng (Sum, Avg, count...).

Vì đang tính tổng của tất cả các bản ghi nên không được phép chọn “Group by” ở bất kỳ trường nào.

Chuyển sang DataSheet View để xem kết quả. (View/ Datasheet View)

**Ví dụ:**

Để quản lý các mặt hàng bán ra trong một cửa hàng người ta sử dụng 2 bảng dữ liệu như sau:

Dshang( **Mahang**, tenhang, dongia)

Dskhach(Mahang, tenkhach, ngaymua, diachi, soluong, thanhtien)

Tạo truy vấn để thống kê xem trong cửa hàng bán bao nhiêu mặt hàng và trung bình đơn giá của mỗi mặt hàng là bao nhiêu?

Tạo truy vấn và đưa bảng Dshang vào tham gia truy vấn

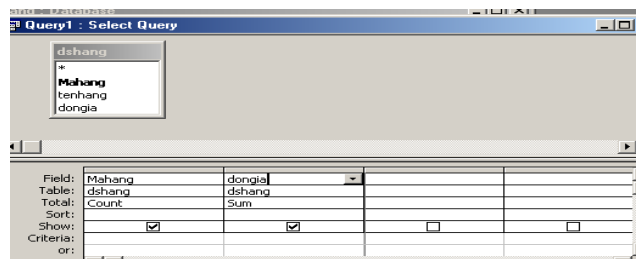
Đưa 2 trường Mahang và dongia vào vùng lưới QBE.

Chọn View/Totals

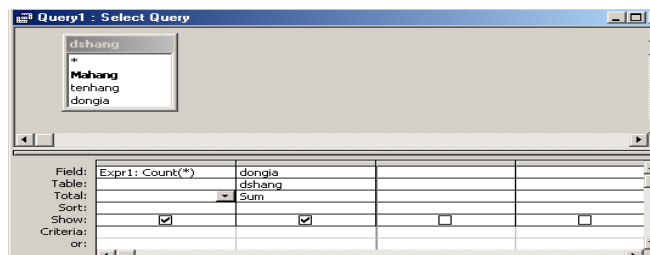
Trong hàng Total của trường Mahang chọn phép toán Count

Trong hàng Total của trường Dongia chọn phép toán Avg

Chọn View/Datasheet View để xem kết quả.



**Chú ý:** Có thể đếm số bản ghi bằng cách dùng Count(\*)



### 6.3. Tính tổng trên từng nhóm bản ghi

Trong nhiều trường hợp chúng ta có thể tính toán trên một nhóm bản ghi nào đó. Chúng ta quy định khi thiết kế truy vấn những trường nào sẽ được tính theo nhóm, trường nào sẽ được tính tổng.

Tạo truy vấn

Đưa các bảng tham gia vào truy vấn

Đưa các trường vào vùng lưới

Chọn View/Totals

Tại hàng total

Chọn “Group by” cho trường làm khóa để nhóm



Chọn các phép toán tính tổng ( Sum,count..) cho các trường còn lại  
Chọn View/Datasheet View để xem kết quả.

**Ví dụ:**

Tạo một truy vấn để tính tổng soluong, thanhtien của mỗi mặt hàng bán được là bao nhiêu?

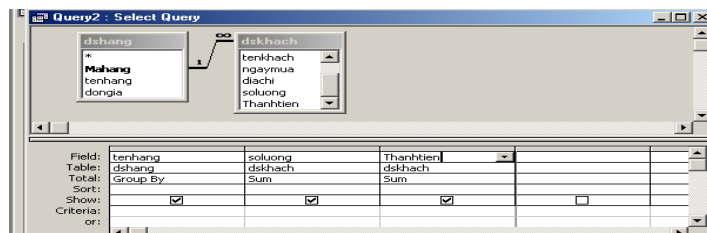
Tạo truy vấn và đưa 2 bảng dshang và dskhach vào tham gia truy vấn

Chọn View/Totals

Đưa các trường Tenhang, soluong, thanhtien vào vùng lưới QBE.

Tại hàng Total của trường Tenhang: Chọn phép toán Group by

Tại hàng Total của trường Soluong, thanhtien: Chọn phép toán SUM.



Chọn View/Datasheet View để xem kết quả.

tenhang	SumOfsoluong	SumOfthanhtien
Dau hoa	35	140000
Xang A83	83	415000
Xang A92	78	468000

Ta có thể thay đổi tiêu đề cột trong khi thực hiện truy vấn tính tổng như sau

tenhang	Tong so luong	Tong thanh tien
Dau hoa	35	140000
Xang A83	83	415000
Xang A92	78	468000

**6.4. Tính tổng trên nhiều nhóm bản ghi**

Access cho phép tính tổng không chỉ trên một mà còn nhiều nhóm bản ghi.

Để làm được điều đó chúng ta chọn “Group by” trên nhiều trường và khi thực hiện

Access sẽ theo thứ tự từ trái sang phải trường bên trái là nhóm mức cao hơn, trường kế tiếp theo là nhóm mức thấp hơn.

**Ví dụ:**

Tạo truy vấn để tính tổng thanh tiền của mỗi mặt hàng bán được theo từng năm nào đó?

Tạo truy vấn và đưa 2 bảng dshang và dskhach vào tham gia truy vấn

Chọn View/Totals

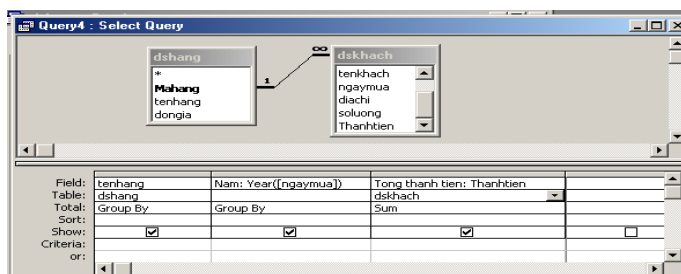
Đưa các trường tenhang, ngaymua, thanh tien vào vùng lưới QBE

Tại hàng Total của trường tenhang chọn phép toán Group by

Tại hàng field của trường ngayban sử dụng hàm year([ngayban]) và tại hàng total chọn phép toán Group by.

Tại hàng Total của trường Thanh tien chọn phép toán Sum.

Thay đổi tiêu đề cột trong truy vấn.



Chọn View/ Datasheet View để xem kết quả

tenhang	Nam	Tong thanh tien
Dau hoa	1997	24000
	1998	88000
	1999	28000
Xang A83	1997	45000
	1998	250000
	1999	120000
XangA92	1997	120000
	1998	168000
	1999	180000

**6.5. Lập biểu thức chọn cho các trường dùng để nhóm khi tính tổng**

Cũng như với những truy vấn khác, chúng ta có thể lập biểu thức chọn cho truy vấn tính tổng theo từng nhóm.

Để thực hiện công việc này, chúng ta lập biểu thức điều kiện ngay hàng Criteria của trường “group by”.

**Ví dụ:**

Tạo truy vấn để tính tổng thanh tiền của mỗi mặt hàng bán được trong năm 1999.

Tạo truy vấn và đưa 2 bảng dshang và dskhach vào tham gia truy vấn

Chọn View/Totals

Đưa các trường tenhang, ngaymua, thanh tien vào vùng lưới QBE

Tại hàng Total của trường tenhang chọn phép toán Group by

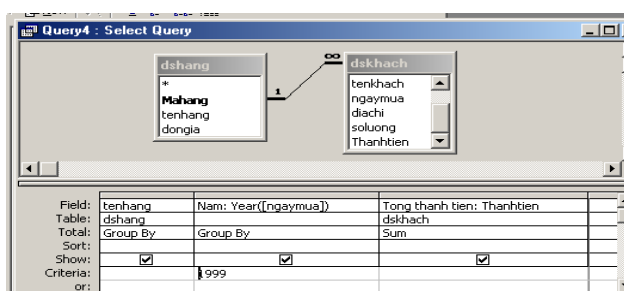
Tại hàng field của trường ngayban sử dụng hàm year([ngayban]) và tại hàng total

chọn phép toán Group by.

Tại hàng Total của trường Thanh tien chọn phép toán Sum.

Tại hàng Criteria của trường ngaymua: gõ vào giá trị 1999

Thay đổi tiêu đề cột trong truy vấn.



Chọn View/ Datasheet View để xem kết quả

tenhang	Nam	Tong thanh tien
Dau hoa	1999	28000
Xang A83	1999	120000
Xang A92	1999	180000

**6.6. Lập biểu thức chọn để giới hạn những bản ghi.**

Trong các phần trước, chúng ta đã biết cách giới hạn các bản ghi trong truy vấn theo một điều kiện nào đó, sự giới hạn này gọi là **giới hạn sau khi tính tổng**.

Bây giờ chúng ta lập biểu thức chọn giới hạn số bản ghi trước khi đưa vào tính tổng trong truy vấn gọi là **giới hạn trước khi tính tổng**.

**Cách tạo**

Tạo truy vấn mới và đưa các bảng tham gia vào truy vấn

Đưa các trường vào vùng lưới QBE.

Chọn menu View/Total

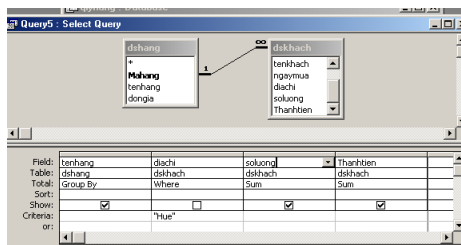
Thiết lập hàng Total thành Where đối với trường chúng ta muốn dùng để đặt biểu thức điều kiện giới hạn số bản ghi trước khi tính tổng.

Gõ biểu thức điều kiện tại hàng Criteria tương ứng.

Chuyển sang chế độ datasheet view để xem kết quả.

**Ví dụ:**

Tạo truy vấn để tính tổng soluong, thanhtien của mỗi mặt hàng bán được đối với khách mua hàng có Queuean ở "Huế"



**Chú ý:** Trong đa số trường hợp, đặt điều kiện lọc trước và sau khi tính tổng có giá trị khác nhau.

**6.7. Dùng truy vấn để cập nhật bản ghi**

Khi truy vấn chỉ dựa trên một bảng, hoặc chỉ bảng có quan hệ 1-1 thì tất cả các trường đều có thể thay đổi, cập nhật. Trong trường hợp có nhiều hơn hai bảng tham gia truy vấn mà có quan hệ 1-∞ thì sẽ phức tạp hơn.

**6.7.1. Khi nào dữ liệu trong trường có thể sửa đổi được**

Bảng sau đây liệt kê các trường hợp khi nào một trường trong kết quả truy vấn hay trong biểu mẫu có thể sửa đổi được.

Loại truy vấn hay trường	Dữ liệu trong trường có cho phép sửa đổi hay không?
Truy vấn dựa trên 1 bảng	Có
Truy vấn dựa trên nhiều bảng có quan hệ 1-1	Có

Truy vấn dựa trên nhiều bảng có quan hệ 1-∞	Thông thường
Truy vấn Tham khảo chéo	Không
Truy vấn tính tổng	Không
Truy vấn với thuộc tính Unique values được thiết lập thành Yes	Không
Truy vấn hội	Không
Truy vấn chuyển nhượng	Không
Trường kiểu biểu thức	Không
Trường trong bản ghi đã bị xoá hoặc bị khoá bởi một người khác trong môi trường nhiều người sử dụng	Không

### **6.7.1. Chỉnh sửa bản ghi trong truy vấn dựa trên hai bảng có quan hệ 1-∞**

Trong truy vấn dựa trên dữ liệu là hai bảng có quan hệ 1-∞, chúng ta có thể sửa đổi tất cả các trường trừ trường liên kết của bảng bên "1". Tuy nhiên có hai trường hợp mà vẫn có thể sửa đổi dữ liệu trong trường liên kết bên "1" là:

Có thể sửa đổi dữ liệu trường liên kết bên bảng "1" trong trường hợp liên kết ngoại và tương ứng của liên kết bên bảng "nhiều" không chứa giá trị.

Có thể sửa đổi dữ liệu trường liên kết bên bảng "1" trong trường hợp đã khai báo thuộc tính tham chiếu toàn vẹn.

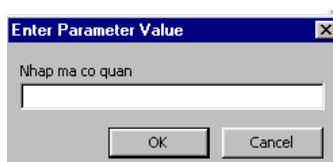
## **7. TRUY VẤN THAM SỐ (Parameter Query)**

### **7.1. Khái niệm**

Nếu thường xuyên chạy cùng một truy vấn, nhưng mỗi lần một tiêu chuẩn khác nhau, thay vì phải thiết kế lại truy vấn sau mỗi lần thực hiện, có thể tiết kiệm thời gian bằng cách tạo truy vấn tham số. Khi thực hiện loại này Access sẽ nhắc nhập điều kiện chọn trong hộp thoại enter parameter Value.

**Ví dụ:**

Giả sử thường xuyên chạy một truy vấn để liệt kê danh sách nhân viên của một cơ quan nào đó có mã cơ quan nhập vào bất kỳ.



**Chú ý:** Nội dung các tham số mà chúng ta nhập vào có thể là hằng ( số, chuỗi, ngày..) nhưng không được biểu thức.

## 7.2. Tạo truy vấn tham số

Tạo truy vấn chọn và đưa các bảng cần thiết vào tham gia truy vấn.

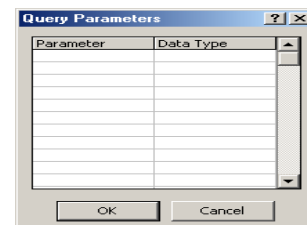
Kéo các trường cần thiết vào vùng lưới QBE.

Tại hàng Criteria gõ vào biểu thức có chứa tham số với chú ý tên tham số phải nằm giữa 2 dấu ngoặc vuông ( [ ] )

Tên tham số cũng là chuỗi nhắc nhở. Access cho phép có khoảng trắng và độ dài tối đa 255 ký tự.

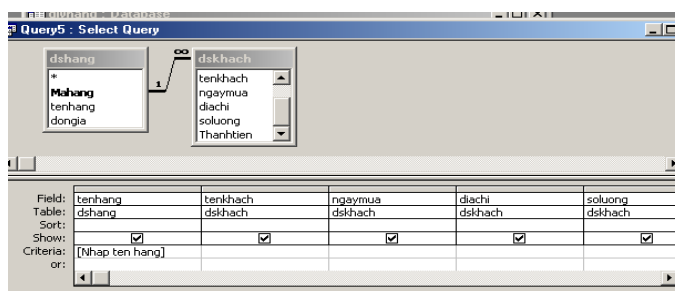
Quy định kiểu dữ liệu cho tham số: Chọn queries/ parameter query.

Trong hộp thoại query parameters: Trong mục Parameter chọn tham số, trong mục Data type chọn kiểu dữ liệu tương ứng.

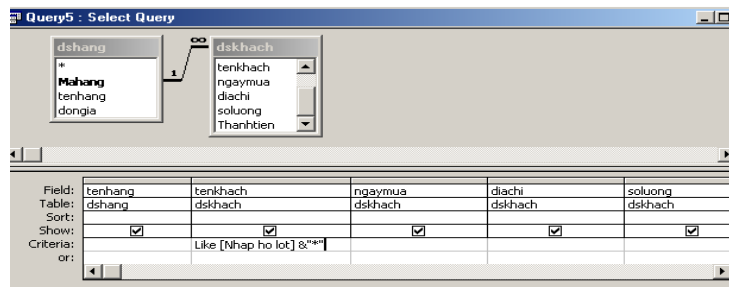


### Ví dụ:

Tạo truy vấn để hiển thị danh sách các khách hàng mua một mặt hàng nào đó (mặt hàng được nhập bất kỳ từ bàn phím).



Tạo truy vấn để hiển thị danh sách các khách hàng mua hàng có họ lót được nhập từ bàn phím.



### 7.3. Truy vấn nhiều tham số

Có thể tạo truy vấn , khi chạy truy vấn nhập nhiều dữ liệu cho điều kiện chọn lựa. Muốn vậy tạo truy vấn nhiều tham số.

**Ví dụ:**

Tạo một truy vấn hiển thị danh sách các khách hàng mua hàng trong khoảng thời gian nào đó (Thời gian được nhập từ bàn phím).

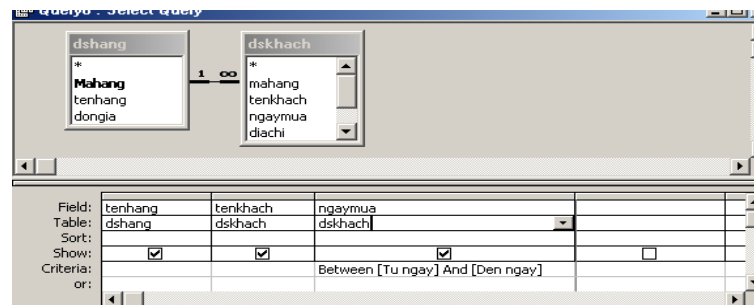
Tạo truy vấn chọn và đưa các 2 bảng dshang và dskhach vào tham gia truy vấn.

Kéo các trường tenhang, tenkhach, ngaymua vào vùng lưới QBE.

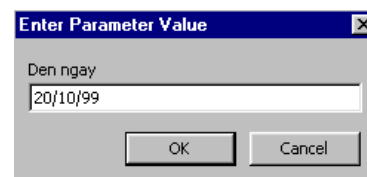
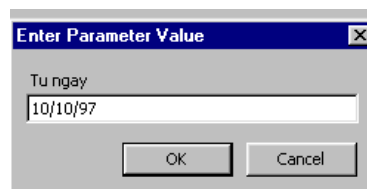
Tại hàng Criteria của trường NGAYSINH chọn:

Between [Từ ngày] and [Đến ngày]

Chọn query/Parameter khai báo kiểu dữ liệu cho 2 tham số là date/time.



Khi chạy truy vấn sẽ cho kết quả sau



### 7.4. Kết hợp giữa truy vấn tham số và truy vấn tính tổng

Trong nhiều bài toán quản lý người ta thường gặp những yêu cầu như: Hãy thống kê xem mỗi loại hàng trong một tháng nào đó bán được với tổng số lượng là bao nhiêu? Tổng thành tiền là bao nhiêu? (Tháng được nhập từ bàn phím). Vì vậy trước hết chúng ta phải thực hiện truy vấn tính tổng xong mới kết hợp truy vấn tham số.

**Ví dụ:**

Hãy tạo một truy vấn để hiển thị tổng thanh tiền của mỗi mặt hàng bán được trong một năm nào đó (Năm được nhập từ bàn phím).

Tạo một truy vấn chọn, đưa bảng Dshang và dskhach vào tham gia truy vấn.

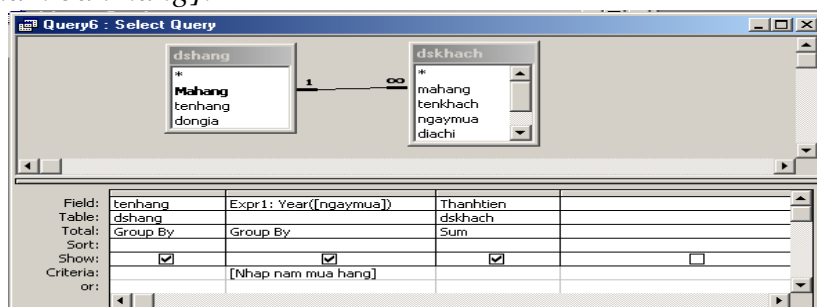
Đưa các trường tenhang, ngayban, thanh tien vào vùng lưới QBE.

Chọn View/Totals

Trong hàng Total: Chọn Group by đối với trường tenhang, và year([ngayban])

Trong hàng Criteria đối với trường Ngaysinh, ta chọn tham số sau:

*[Nhập vào nam bán hàng].*



## 8. TRUY VẤN THAM KHẢO CHÉO (Crosstab query)

### 8.1. Khái niệm

Truy vấn tham khảo chéo là loại truy vấn dùng để tóm lược dữ liệu và trình bày kết quả theo dạng như một bảng tính. Truy vấn tham khảo chéo cũng có thể thống kê một khối lượng dữ liệu lớn và trình bày đơn giản hơn do đó thường sử dụng để so sánh dữ liệu.

### 8.2. Tạo truy vấn tham khảo chéo

Muốn tạo một truy vấn tham khảo chéo chúng ta phải xác định được 3 yếu tố chính: Trường làm tiêu đề cột (Duy nhất 1 trường), trường làm tiêu đề hàng (Có thể nhiều trường), trường tính giá trị (Duy nhất 1 trường).



### **Cách tạo**

Tạo truy vấn chọn và đưa các bảng vào tham gia truy vấn

Đưa các trường vào vùng lưới QBE

Chọn Query/Crosstab

*Quy định trường làm tiêu đề cột*

Tại hàng Total: Bắt buộc chọn phép toán Group by

Tại hàng Crosstab: Chọn Column heading

*Quy định trường làm tiêu đề hàng*

Tại hàng Total: Ít nhất một trong các trường phải chọn phép toán Group by

Tại hàng Crosstab: Chọn Row heading

*Quy định trường tính giá trị*

Tại hàng Total: Chọn phép toán thích hợp

Tại hàng Crosstab: Chọn Value

### **Ví dụ:**

Cho 2 bảng dữ liệu Dstruong(**matruong**, tentruong, sodt)

Danh sach(matruong, hoten, ngaysinh, gioitinh, xeploai)

Tạo một truy vấn Crosstab để phản ánh tổng số lượng sinh viên xếp mỗi loại của trong từng trường bao nhiêu.?

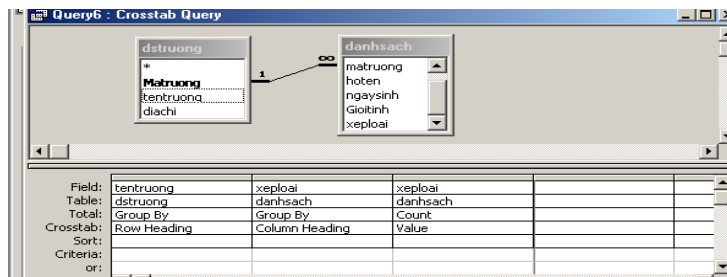
Tạo truy vấn và đưa 2 bảng dữ liệu vào tham gia truy vấn, đưa các trường tentruong và xeploai vào vùng lưới QBE. (Trường Xeploai đưa vào 2 lần)

Chọn Query/ crosstab query

Tại hàng Total của trường tentruong: Chọn phép toán Group by, hàng crosstab: chọn Row heading

Tại hàng Total của trường Xeploai: Chọn phép toán Group by, hàng Crosstab chọn Column heading.

Tại hàng Total của trường Xeploai: Chọn phép toán Count, hàng Crosstab chọn Value.



Chọn View/ Datasheet View để xem kết quả

tentruong	Gioi	Kha	Trung binh	Yeu
DHKH		3	1	
DHNL		1	1	1
DHSP	1	1		1
DHYK		1	1	1

### 8.3. Định dạng cho tiêu đề cột

Với truy vấn Crosstab, chúng ta có thể can thiệp nhiều hơn về cách trình bày tiêu đề cột trong bảng. Chúng ta có thể thay đổi bằng cách đặt lại thuộc tính Column Heading của truy vấn. Thuộc tính này cho phép chúng ta: Chỉ định sắp xếp các tiêu đề cột.

Muốn định dạng tiêu đề cột thực hiện các bước sau:

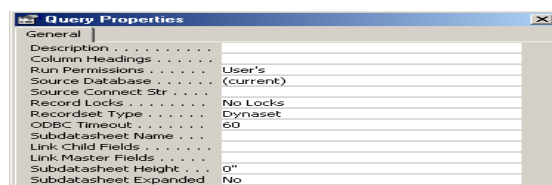
Tạo truy vấn Crosstab.

Chuyển sang chế độ Design View.

Mở bảng thuộc tính truy vấn.

Tại hàng Column Heading : Gõ các tiêu đề cột theo thứ tự mà chúng ta muốn

Các giá trị này phải cách nhau bởi dấu chấm phẩy ( ; )



## 9. TRUY VẤN HÀNH ĐỘNG

### 9.1. Các loại truy vấn hành động

Truy vấn hành động giúp người sử dụng tạo bảng mới hay sửa đổi dữ liệu trong các bảng. Có 4 loại truy vấn hành động:

*Truy vấn tạo bảng (Make table query):* Tạo bảng mới từ một bảng hay nhiều bảng đã tồn tại dữ liệu.

*Truy vấn cập nhật (Update query):* Dùng để cập nhật dữ liệu cho một hoặc nhiều trường trong bảng dữ liệu.

*Truy vấn xoá (Delete query):* Xoá các bản ghi thoả mãn các điều kiện từ một hay nhiều bảng dữ liệu.

*Truy vấn nối (Append query):* Nối một số bản ghi từ một hoặc nhiều bảng dữ liệu vào sau một hoặc nhiều bảng dữ liệu khác.

## 9.2. Truy vấn tạo bảng

Truy vấn tạo bảng sẽ tạo ra một bảng mới bằng cách rút các bản ghi thoả mãn các điều kiện nào đó.

### Cách tạo truy vấn

Để tạo truy vấn tạo bảng chúng ta tạo truy vấn chọn và đưa bảng vào tham gia truy vấn. Đưa các trường vào vùng lưới QBE

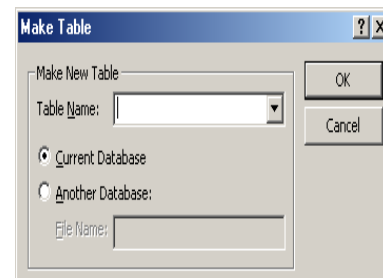
Chọn query/ make table query

Trong mục Table name: Đặt tên mới cho bảng muốn tạo.

Chọn Current Database: CSDL hiện thời

Another Database: Tạo bảng trong CSDL khác.

Chọn các điều kiện (Nếu có).



## 9.3. Truy vấn xoá

Truy vấn xoá giúp chúng ta loại bỏ các bản ghi thoả mãn một số điều kiện nào đó

### Cách tạo truy vấn

Để tạo truy vấn xoá chúng ta tạo truy vấn chọn và đưa bảng vào tham gia truy vấn.

Chọn query/ Delete query

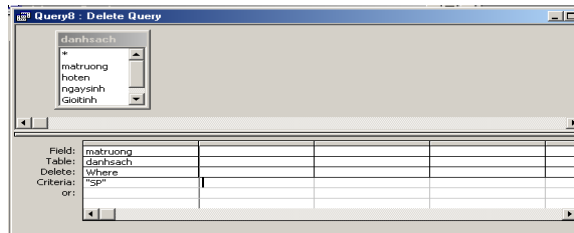
Trong vùng lưới QBE tại hàng Field chọn các trường cần so sánh với điều kiện xoá

Tại hàng Delete: Chọn phép toán Where

Tại hàng Criteria: Chọn điều kiện xoá

### Ví dụ:

Tạo truy vấn để xoá những sinh viên có matruong là "SP"



### 9.3. Truy vấn cập nhật

Truy vấn này dùng để cập nhật giá trị hoặc sửa đổi giá trị của các trường trong bảng dữ liệu.

#### Cách tạo truy vấn

Tạo một truy vấn chọn và đưa bảng vào tham gia truy vấn

Chọn Query/Update query

Tại hàng Field: Chọn trường cần cập nhật dữ liệu

Tại hàng Update to: Chọn Biểu thức cần tính giá trị

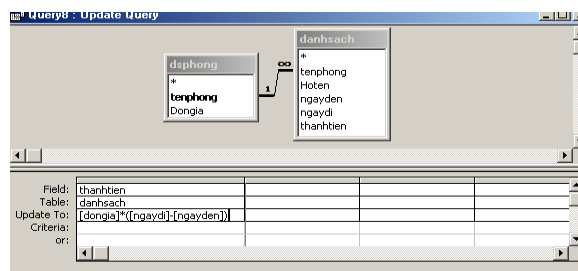
Tại hàng Criteria: Chọn điều kiện (nếu có).

#### Ví dụ:

Cho 2 bảng dữ liệu Dsphong(**tenphong**, dongia)

Dskhachtro (tenphong, ngayden, ngaydi, thanhtien)

Tạo truy vấn để cập nhật giá trị cho trường  $Thanhtien = (ngaydi - ngayden) * dongia$



### 9.4. Truy vấn nối dữ liệu

Truy vấn nối dữ liệu dùng để nối dữ liệu từ một bảng này vào sau một bảng khác.

#### Cách tạo truy vấn

Tạo truy vấn chọn và đưa bảng dữ liệu vào để nối với bảng khác tham gia truy vấn.

Chọn Queries/Append query

Trong mục Table name: Chọn bảng cần nối vào và chọn OK.

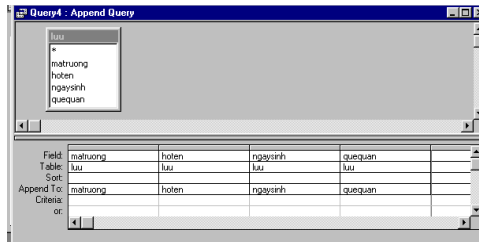
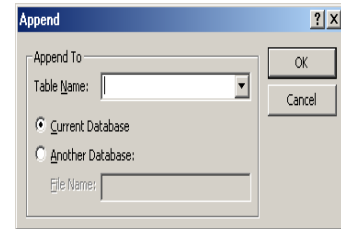
Chọn Current Database: CSDL hiện thời

Another Database: Tạo bảng trong CSDL khác.

Trong vùng lưới QBE của truy vấn tại hàng Field:

Đưa các trường của bảng gốc vào.

Trong hàng Append to: Đưa các trường tương ứng của bảng cần nối và đặt điều kiện nếu cần thiết.



### *☞ Chú ý*

Trong truy vấn nối dữ liệu thì các trường được nối với nhau tương ứng phải cùng kiểu dữ liệu. Nếu các trường tương ứng không có cùng kiểu dữ liệu thì sẽ không được nối. Nếu các trường có Field size không phù hợp thì tùy theo việc nối dữ liệu mà sẽ cắt bớt hoặc thêm vào ký tự trắng.

## 1. GIỚI THIỆU VỀ NGÔN NGỮ SQL

SQL là một ngôn ngữ dùng để truy xuất dữ liệu, cập nhật, thay đổi và quản lý các CSDL quan hệ.

Khi tạo một truy vấn thì ACCESS tự động xây dựng các câu lệnh SQL tương ứng.

Một số truy vấn của ngôn ngữ SQL như sau:

*Truy vấn hội (Union Query)*

*Truy vấn chuyển nhượng (pass through Query)*

*Truy vấn định nghĩa dữ liệu (Data Definition Query)*

*Truy vấn con (Sub Query)*

### ☞ Chú ý

Trong SQL mỗi câu lệnh có thể được viết trên nhiều hàng và kết thúc mỗi câu lệnh là dấu chấm phẩy (;)

## 2. SQL XỬ LÝ TRÊN BẢNG DỮ LIỆU

### 2.1. Tạo bảng mới

#### Cú pháp

```
Create table <Table name>(<Field1> <Type>[(Size)], <Field2>
<Type>[(Size)].....
```

```
[Constraint < Primary key name> primary key <Fieldname key>]
```

```
[Constraint <Index name> Unique <Field name Index>]
```

**Chức năng:** Tạo cấu trúc của một bảng mới.

#### Giải thích chức năng các tham số

*Table name:* Tên bảng cần tạo

*Fieldname1, Fieldname2.....:* Các trường trong bảng cần tạo, tối thiểu 1 trường.

*Type:* Các kiểu dữ liệu tương ứng.

*Size* : Kích thước của trường

*Primary key name* : Tên khoá chính

*Fieldname key* : Trường làm khoá chính

*Index name* : Tên chỉ mục

*Fieldname Index* : Trường làm chỉ mục

*Constraint.....Primary key* : Dùng để thiết lập khoá chính

*Constraint..... Unique:* Thiết lập chỉ mục (Không trùng nhau)

☛ **Chú ý:** Đối với các kiểu dữ liệu khi sử dụng trong SQL

Kiểu dữ liệu chuẩn	Khai báo tương ứng trong SQL
Text	Text(Size)
Byte	Byte
Integer	Short
Long Integer	Long
Single	Single
Double	Double
Date/Time	Datetime
Yes/No	Yesno
OLE Object	OLE Object
Currency, Memo, Counter	Currency, Memo, Counter

**Ví dụ :** Tạo bảng DSTRUONG có cấu trúc sau

Fieldname	Data Type	size
Matruong	Text	2
Tenruong	Text	20
SODT	Text	9

Create table dstruong(matruong text(2), tenruong text(20), Sodt text(9));

**Ví dụ** Tạo bảng DSHOCVIEN có cấu trúc sau

Fieldname	Data Type	size
Mahv	Text	4 (Khoá chính)
Tenhv	Text	30
Ngaysinh	Date/time	

SODT            Text            9 (Lập chỉ mục)

Create table dshocvien(mahv text(4), tenhv text(30), ngaysinh Datetime, sodt text(9),  
Constraint khoa primary key (mahv), Constraint chimuc unique (sodt));

☞ **Chú ý:** Nếu tên trường có ký tự trắng hoặc tên bảng, tên khoá chính, tên chỉ mục thì phải đặt trong cặp dấu [...]

**Ví dụ**

Create table [Bang NV] ([Ma nv] text(2), [ ho va ten] text(30));

**2.2. Thay đổi cấu trúc của bảng**

**2.2.1. Thêm hoặc loại bỏ trường**

**Cú pháp**

*Alter table <Table name>[ add column <fieldname><type>]*

*[Drop column <Field name>]*

*[Add Constraint <Index name> unique <fieldname>]*

*[Drop Constraint <Index name>]*

**Chức năng:** Thay đổi cấu trúc của bảng

Giải thích:

ADD Column...: Thêm trường vào bảng

Drop column...: Loại bỏ trường ra khỏi bảng

Add Constraint.....: Thêm tên chỉ mục

Drop Constraint..... Loại bỏ tên chỉ mục

**Ví dụ:** Giả sử đã tồn tại bảng MATHANG cấu trúc sau

<i>Fieldname</i>	<i>Data Type</i>	<i>size</i>
MAHANG	Text	4
TENHANG	Text	20
GIA	Integer	
MAXN	Text	2
Ngaynhap	Date/time	

Thêm trường SOLUONG có kiểu byte vào bảng MATHANG

Alter Table mathang soluong byte;



**Ví dụ :** Thêm chỉ mục có ten cmx cho trường MAXN

Alter table mathang add constraint cmx unique Maxn

**Ví dụ :** Loại bỏ chỉ mục cmx

Alter table mathang drop constraint cmx

**Ví dụ :** Loại bỏ trường ngaynhap ra khỏi bảng MATHANG

Alter table mathang drop column ngaynhap

### **2.2.2. Loại bỏ chỉ mục**

#### **Cú pháp**

*Drop Index <Index name> on <Table name>*

**Chức năng:** Loại bỏ 1 chỉ mục nào đó.

### **2.3. Xoá bảng**

#### **Cú pháp**

*Drop table <Table name>*

**Chức năng:** Xoá bảng dữ liệu nào đó.

**Ví dụ:** Xoá bảng MATHANG

Drop table MATHANG

## **3. SQL XỬ LÝ TRÊN TRUY VẤN**

### **3.1. Truy vấn chọn (Select query)**

#### **Cú pháp**

*Select <Scope> <Fieldname1> [AS <New name>],.....*

*From <Table name>*

*[Where <Condition>]*

**Chức năng:** Tạo truy vấn chọn

Trong đó:

Scope: Phạm vi (Mặc định là ALL, Top n: Lấy n bản ghi đầu tiên)

Nếu có AS <New name> thì sẽ thay thế tên cho Fieldname tương ứng.

Table name: Tên bảng cần lấy dữ liệu.

Condition: Điều kiện để hạn chế dữ liệu.

**Ví dụ:** Cho bảng dữ liệu DOCGIA sau

Fieldname	Data Type	Size	Description
MADG	Text	2	Mã độc giả (Khoá chính)
MASACH	Text	4	
HOTEN	Text	30	
QUEQUAN	Text	30	
NGAYSINH	Date/time	8	
NGAYMUON	Date/time	8	

Chọn 2 trường MADG và HOTEN

Select MADG, hoten

From docgia;

Chọn 2 trường Masach và hoten mà chỉ những masach bắt đầu là T

Select MADG, hoten

From docgia where masach like "T\*";

Chọn Hoten, quequan, madg cho những độc giả có quê quán ở Huế và đổi tên trường hoten thành Họ và tên

Select hoten AS [Họ và tên], queuqan, madg

From docgia where quequan="Huế";

Chọn những độc giả mượn sách trong tháng 8 hoặc năm 1999.

Select \* from docgia where month([ngaymuon])=8 or year([ngaymuon])=1999;

### 3.2. Truy vấn tính tổng (Total query)

#### Cú pháp

*Select .....from.....[where <Condition>]*

*Group by [Group fieldname]*

*[Having <Group Condition>];*

**Chức năng:** Tạo một truy vấn tính tổng.

**Ví dụ:** Cho bảng dữ liệu BANHANG có cấu trúc

Fieldname	Data type	size
Mahang	text	2
Soluong	integer	

Ngàyban                      Date/time      8

Tạo một truy vấn để thống kê xem mỗi loại hàng bán được với số lượng là bao nhiêu?

```
Select mahang, sum([soluong])
```

```
From banhang
```

```
Group by mahang;
```

Tạo truy vấn để thống kê xem mỗi loại hàng trong tháng 7 bán được với số lượng bao nhiêu? Chỉ hiển thị những loại hàng mà số lượng bán trên 20.

```
Select mahang, sum([soluong])
```

```
From banhang
```

```
where month([ngayban])=7
```

```
Group by mahang
```

```
having sum([soluong])>20;
```

### 3. 3.Truy vấn tham khảo chéo

#### Cú pháp

*Transform <Value Express>*

*Select.....From.....Where.....*

*Group by <Row Heading Field>*

*Pivot <Column heading Field>*

**Chức năng:** Tạo truy vấn tham khảo chéo

#### Ví dụ

```
Transform sum([soluong])
```

```
select Mahang, tenhang, sum([soluong]) from dskhang
```

```
Group by tenhang
```

```
Pivot Mahang;
```

### 3.4. Truy vấn tạo bảng

#### Cú pháp

*Select <Field select> into <New Table name>*

*From <Old Table name>*

*[Where <Condition>]*

**Chức năng:** Tạo một truy vấn tạo bảng

**Ví dụ**

```
Select Mahang, tenhang into Luu
From Dskhang
Where Mahang Like "A*";
```

### 3.5. Truy vấn nối dữ liệu

**Cú pháp**

```
Insert into <append Table name>
Select <field select>
From <Table name>
[Where <Condition>]
```

**Chức năng:** Tạo truy vấn nối dữ liệu

Nếu chỉ thêm 1 bản ghi với các giá trị cụ thể thì ta thực hiện câu lệnh

```
Insert into <Table name and Field list>
values <append values>
```

**Ví dụ:**

```
Insert into luu1(Hoten, quequan)
Values ("Nguyen an", "Hue")
```

### 3.6. Truy vấn cập nhật dữ liệu

**Cú pháp**

```
Update <Update Table name>
Set <Field name>=<Express>
[Where <Condition>]
```

**Chức năng:** Tạo một truy vấn dùng để cập nhật dữ liệu

**Ví dụ:**

```
Update dssv
set [hocbong]=[hocbong]+200000
Where Uutien="1";
```

### 3.7. Truy vấn xoá

#### Cú pháp

```
Delete <Table.*>  
From <Delete Table name>  
[Where <Condition>]
```

**Chức năng:** Dùng để tạo một truy vấn xoá các bản ghi trong bảng theo một hoặc nhiều điều kiện nào đó.

☞ **Chú ý:** nếu mệnh đề From chỉ có 1 bảng duy nhất thì không cần liệt kê các bảng trong mệnh đề DELETE.

#### Ví dụ:

Có 2 bảng dữ liệu DSHS và DSDTHI có quan hệ 1-1 trên trường MAHS. Hãy xoá những học sinh có điểm thi <5 trong bảng DSHS và DSDTHI.

```
Delete DSHS.*  
From DSHS INNER JOIN DSDTHI ON DSHS.MAHS=DSDTHI.MAHS  
Where diem<5.;
```

### 3.8. Tạo mối quan hệ giữa các bảng

Muốn tạo một truy vấn để truy xuất dữ liệu từ 2 hay nhiều bảng thì phải tạo các mối quan hệ giữa các bảng đó.

#### Cú pháp

```
..... From <Table name 1> inner join <Table name 2> ON <Table name 1>.<Field name1>=<Table name 2>.<Field name 2>.....
```

Hoặc

```
.....From <Table name 1> Left join/ Right join <Table name 2> ON <Table name 1>.<Field name1>=<Table name 2>.<Field name 2>.....
```

#### Ví dụ:

Tạo truy vấn gồm : Matruong, tentruong, hoten từ 2 bảng Dstruong và DSHS dựa vào trường liên kết Matruong.

```
Select Matruong, tentruong, hoten From Dstruong inner join dshs on  
Dstruong.matruong=dshs.matruong;
```

**Ví dụ:**

Cho 3 bảng dữ liệu Dstruong( **Matruong**, tentruong)

Dskhoa(Matruong, tenkhoa, **Makhoa**)

DSSV(Makhoa, Hoten, Ngaysinh, quequan)

Tạo một truy vấn để hiển thị danh sách sinh viên thuộc mỗi khoa của mỗi trường

*Select Distinctrow*

*Matruong, tentruong, tenkhoa, hoten*

*From Dstruong inner join (Dskhoa inner join DSSV ON Dskhoa.Makhoa=DSSV.Makhoa)*

*ON Dstruong.Matruong=Dskhoa.Matruong;*

☞ **Chú ý**

Có thể sử dụng liên kết ngoại trái Left join hoặc phải Right join

**3.9. Truy vấn con (Sub query)**

Truy vấn con là một mệnh đề Select.....From.....Wheres được lồng ghép vào một trong các mệnh đề sau:

Select.....From.....Where

Select.....Into.....

Insert.....Into.....

Delete.....

Update.....

**Cú pháp**

*Select.....From.....Where.....*

*<Biểu thức so sánh> ANY|SOME|ALL <Mệnh đề truy vấn con>*

*<Biểu thức tìm kiếm> IN | NOT IN <Mệnh đề truy vấn con>*

*EXIST | NOT EXISTS <Mệnh đề truy vấn con>;*

**Chức năng:** Tạo một truy vấn con

**Giải thích các tham số**

<Biểu thức so sánh>: Là một biểu thức và một phép toán so sánh.

<Biểu thức tìm kiếm>: Là một biểu thức mà tập hợp kết quả của truy vấn con sẽ được tìm kiếm.

<Mệnh đề truy vấn con>: Là dạng mệnh đề ở trong cú pháp và đặt giữa hai dấu ( ).

ANY, SOME: Các bản ghi trong truy vấn chính thoả mãn điều kiện so sánh với bất kỳ hoặc một vài các bản ghi nào truy xuất được từ truy vấn con.

ALL: Các bản ghi trong truy vấn chính thoả mãn với điều kiện so sánh với tất cả bản ghi nào truy xuất được từ truy vấn con.

IN: Các bản ghi trong truy vấn chính mà có tồn tại một vài bản ghi trong truy vấn con có giá trị bằng nó.

NOT IN: Các bản ghi trong truy vấn chính mà không tồn tại một vài bản ghi trong truy vấn con có giá trị bằng nó.

EXISTS (NOT EXISTS): Phép so sánh True/ False để xác định nhận truy vấn con có kết quả là bản ghi nào không.

**Ví dụ:**

Cho 2 bảng dữ liệu KHO(Mahang, Tenhang, Dongia)

NKBAN (Mahang, Hoten, Ngaymua, Giamgia, Dongia)

Tim tất cả những mặt hàng mà đơn giá lớn hơn vài mặt hàng được bán với Giamgia là 20%.

*Select \* From Kho*

*Where dongia > ANY (select dongia From NKBAN Where giamgia=20%);*

Tim những mặt hàng bán ra với giảm giá >=10%.

*Select \* from Kho*

*Where Mahang IN (Select mahang From NKBAN Where giamgia>=0.1);*

**Ví dụ:**

Cho 2 bảng danh sách DSKH(MAKHACH, HOTEN, QUEQUAN, SDT)

DATHANG(MAKHACH, SOLUONG, NGAYDAT)

Tim những người khách đặt hàng trước 10/10/99 bao gồm Hoten, Quequan.

*Select Hoten, quequan From DSKH*

*Where Makhach IN (Select makhach from DATHANG Where NGAYDAT<=#10/10/99#;.*

**3.10. Truy vấn hội (Union Query)**

Dùng để nối (Kết hợp) dữ liệu các trường tương ứng từ 2 hay nhiều bảng hoặc truy vấn vào trường.

**Cú pháp:** *Select .....From.....Where.....*

*UNION | UNION ALL*

*Select.....*

**Chức năng:** Tạo truy vấn hội

**Giải thích:**

UNION: Không muốn các bản ghi trùng nhau hiển thị

UNION ALL: Hiển thị các bản ghi trùng nhau

**Ví dụ:**

Cho 2 bảng dữ liệu DHSVIEN( Hoten, Lop, Matruong, Diachi)

CDHSVIEN( Hoten, Lop, Matruong, Diachi)

Sử dụng truy vấn hội để liệt kê Hoten, Lop, Matruong của sinh viên 2 hệ (Địa học, Cao đẳng).

*Select Hoten, Lop, Matruong from DHSVIEN*

*UNION Select Hoten, Lop, Matruong from DHSVIEN*

Liệt kê Hoten, Lop của những sinh viên 2 hệ và có quê quán ở Huế

*Select Hoten, Lop from DHSVIEN*

*UNION Select Hoten, Lop from DHSVIEN Where Diachi="Huế";*



## Chương 5

# BIỂU MẪU

---

Từ trước đến nay chúng ta vẫn làm việc một cách đơn điệu với các bảng, truy vấn với cách trình bày dữ liệu hiệu quả nhưng không đẹp mắt. Với biểu mẫu (form) trong Access sẽ giúp chúng ta khắc phục điều này. Biểu mẫu trong Access rất linh động, chúng ta có thể dùng biểu mẫu để nhập, xem, hiệu chỉnh dữ liệu. Hoặc là dùng biểu mẫu để tạo ra các bảng chọn công việc làm cho công việc của chúng ta thuận lợi và khoa học hơn. Hoặc dùng biểu mẫu để tạo ra các hộp thoại nhằm thiết lập các tùy chọn cho công việc quản lý của mình.

### 1. KHÁI NIỆM VỀ BIỂU MẪU

Nếu chúng ta đã quen điền các tờ biểu, mẫu trong cuộc sống hàng ngày thì chúng ta có thể hình dung một biểu mẫu trong Access cũng vậy. Một biểu mẫu trong Access định nghĩa một tập dữ liệu chúng ta muốn lấy và từ đó đưa vào CSDL. Cũng vậy biểu mẫu cũng có thể dùng để xem xét dữ liệu hay in ra máy in.

Trong môi trường của Hệ QTCSDL Access chúng ta có thể thiết kế các biểu mẫu có hình thức trình bày đẹp, dễ sử dụng và thể hiện đúng các thông tin cần thiết. Chúng ta có thể đưa vào biểu mẫu các đối tượng như văn bản, hình ảnh, đường vẽ kết hợp với các màu sắc sao cho biểu mẫu của chúng ta đạt được nội dung và hình thức trình bày ưng ý nhất. Hình thức và cách bố trí các đối tượng ra sao trên biểu mẫu hoàn toàn tùy thuộc vào khả năng thẩm mỹ và ng khiếu trình bày của chúng ta.

### 2. TÁC DỤNG VÀ KẾT CẤU CỦA BIỂU MẪU

#### 2.1. Tác dụng của biểu mẫu

Biểu mẫu cung cấp một khả năng thuận lợi để hiển thị dữ liệu. Chúng ta có thể xem mọi thông tin của một bản ghi thay vì ở chế độ Datasheet nghèo nàn trước đây bằng chế độ Form View, một phương cách tiên tiến hơn.

Sử dụng biểu mẫu tăng khả năng nhập dữ liệu, tiết kiệm thời gian và ngăn ngừa các lỗi do đánh sai. Chẳng hạn thay vì gõ vào các giá trị của tất cả các trường chúng ta có thể tạo những danh sách (gọi là combo box) để chọn trên biểu mẫu (đây là phương cách áp dụng rất hiệu quả để tránh đánh sai dữ liệu).

Biểu mẫu cung cấp một hình thức trình bày hết sức tiện nghi để xem, nhập và hiệu chỉnh các bản ghi trong CSDL. Access cung cấp các công cụ thiết kế biểu mẫu hỗ trợ rất đặc lực cho chúng ta trong việc thiết kế những biểu mẫu để sử dụng mà lại có thể tận dụng được các khả năng:

Hình thức thể hiện dữ liệu đẹp, trình bày lôi cuốn với các kiểu font và hiệu ứng đồ họa đặc biệt khác ...

Quen thuộc với người sử dụng vì nó giống các biểu mẫu trên giấy thông thường.

Có thể tính toán được.

Có thể chứa cả biểu đồ.

Có thể hiển thị dữ liệu từ nhiều bảng (hoặc truy vấn)

Tự động hóa một số thao tác phải làm thường xuyên.

## **2.2. Kết cấu của biểu mẫu**

Các thông tin trên biểu mẫu có thể lấy dữ liệu từ một bảng hay truy vấn nào đó, nhưng cũng có thể độc lập đối với cả bảng lẫn truy vấn, chẳng hạn như các đối tượng đồ họa. Dáng vẻ trình bày của biểu mẫu được thực hiện trong quá trình thiết kế.

Tất cả các thông tin thể hiện trên biểu mẫu được chứa trong những đối tượng gọi là *điều khiển* (control). Điều khiển có thể dùng để thể hiện dữ liệu hoặc thực hiện các hàng động hoặc trang trí cho biểu mẫu.

Một số điều khiển được buộc vào với các trường của bảng hay truy vấn, gọi là bảng cơ sở hay truy vấn cơ sở. Do đó chúng ta có thể dùng biểu mẫu để nhập dữ liệu vào

các trường hay lấy dữ liệu từ các trường đó ra để xem. Ví dụ dùng Text box để nhập hay hiển thị chuỗi và số, dùng Object frame để thể hiện hình ảnh.

Một số điều khiển khác trình bày thông tin được lưu trữ trong thiết kế bảng. Ví dụ dùng Label (nhãn) để thể hiện thông tin có tính chất mô tả; đường và các hình khối để tổ chức dữ liệu và làm biểu mẫu có hình thức hấp dẫn hơn.

### 3. TẠO BIỂU MẪU

#### 3.1. Tạo biểu mẫu tự động với Autoform

Access cung cấp chức năng Autoform cho phép chúng ta tạo biểu mẫu dựa trên các bảng hoặc truy vấn đã được xây dựng trước đó.

##### Cách tạo

Trong cửa sổ Database, chọn form, chọn New

Trong mục *Choose the table or query Where*

*the object's data comes from:*

Chọn bảng hoặc truy vấn làm nguồn dữ liệu cho form.

Chọn **Autoform Columnar**: Nếu muốn tạo lập biểu mẫu dạng cột, trong đó mỗi trường trong bảng hay truy vấn là một dòng.

Chọn **Autoform Tabular**: Nếu muốn tạo lập biểu mẫu dạng hàng, trong đó mỗi trường trong bảng hay truy vấn là một cột và một bản ghi trong một dòng.

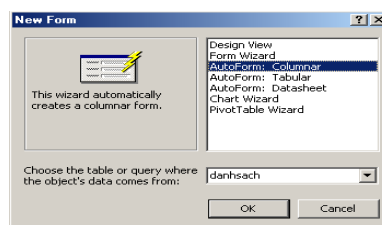
Chọn **Autoform Datasheet**: Nếu muốn tạo lập biểu mẫu theo dạng bảng, trong đó mỗi cột tương ứng một trường và mỗi dòng là một bản ghi.

Chọn OK.

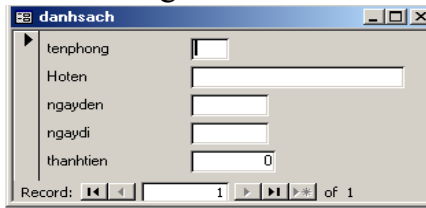
##### Ví dụ

Cho bảng Danhsach( Tenphong, hoten, ngayden, ngaydi, thanhtien)

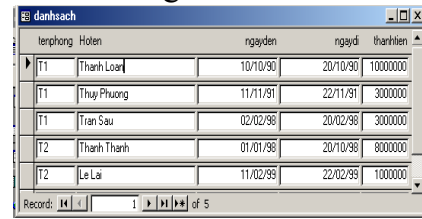
Hãy thiết kế biểu mẫu dựa trên chức năng Autoform sử dụng bảng Danhsach làm nguồn dữ liệu.



Biểu mẫu dạng **Autoform Columnar**



Biểu mẫu dạng **Autoform tabular**



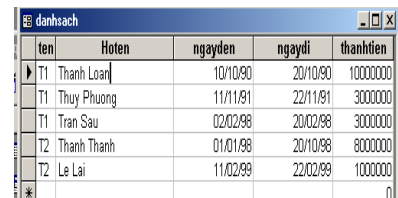
Biểu mẫu dạng **Autoform Datasheet**

**3.2. Tạo biểu mẫu sử dụng Wizard**

Tạo biểu mẫu sử dụng công cụ Autoform

thì Access không cho phép người sử dụng can thiệp

vào quá trình tạo biểu mẫu, chẳng hạn như hạn chế số trường..... thì Form Wizard cho phép người sử dụng can thiệp vào quá trình tạo biểu mẫu.



**Cách tạo**

Trong cửa sổ Database chọn Form, chọn New

Trong mục *Choose the table or query Where the object's data comes from:*

Chọn bảng hoặc truy vấn làm nguồn dữ liệu cho form.

Chọn Form Wizard

Chọn OK

Trong mục Available Field: Chọn các trường đưa vào biểu mẫu, nhấn nút >>

Chọn nút **Next**.

Chọn *Columnar* : Biểu mẫu hiển thị theo dạng cột

*Tabular* : Biểu mẫu hiển thị theo dạng hàng

*Datasheet* : Biểu mẫu hiển thị theo dạng bảng

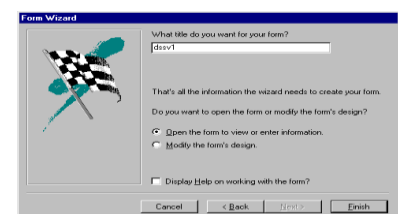
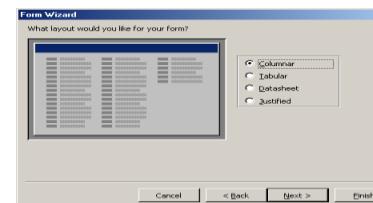
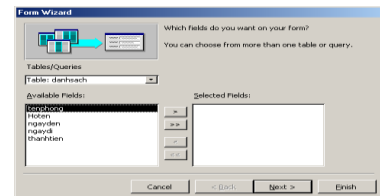
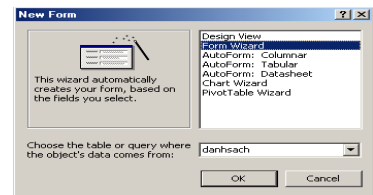
*Justified* : Biểu mẫu hiển bình thường (đều).

Chọn **Next**

Chọn loại biểu mẫu

Chọn **Next**

Đặt tiêu đề cho Form



Chọn *Open the form to view or*

*enter information*: Nếu muốn mở Form sau khi chọn Finish.

Chọn *Modify the form's design*: Nếu muốn form ở dạng thiết kế.

Chọn **Finish** .

Lưu form.

### 3.3. Tạo biểu mẫu không sử dụng Wizard (Do người sử dụng tự thiết kế)

Tạo biểu mẫu sử dụng công cụ Autoform và Form wizard người sử dụng có thể nhanh chóng thiết kế các biểu mẫu nhờ vào các đặc tính hỗ trợ của Access. Nhưng đối với hai cách trên chỉ cung cấp một số hạn chế các phương án xây dựng biểu mẫu mà không thoả mãn yêu cầu của người sử dụng khi muốn thiết kế biểu mẫu theo ý của riêng mình. Do đó người sử dụng phải tự thiết kế một biểu mẫu không cần sự hỗ trợ của Access.

#### Cách tạo

Trong cửa sổ Database chọn Form, chọn New

Chọn bảng dữ liệu hoặc truy vấn làm nguồn

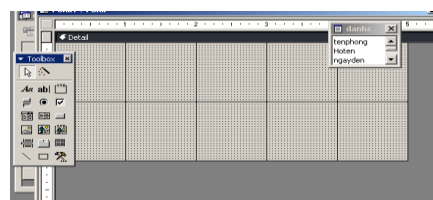
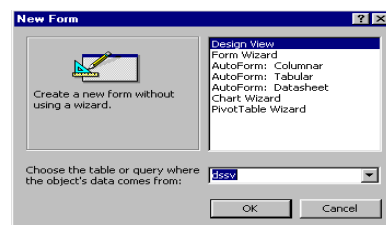
dữ liệu cho form, chọn **OK**

Xây dựng các điều khiển cho biểu mẫu

(Đưa các trường trong bảng dữ liệu vào biểu mẫu).

Thiết lập các thuộc tính cho các điều khiển.

Lưu biểu mẫu.



## 4. CÁC CHẾ ĐỘ HIỂN THỊ VÀ CÁC THÀNH PHẦN CỦA BIỂU MẪU

### 4.1. Các chế độ hiển thị

Có 4 chế độ hiển thị của biểu mẫu

#### 4.1.1. Chế độ Design View

Dùng để tạo biểu mẫu mới hay thay đổi cấu trúc của một biểu mẫu đã tồn tại.

Khi đang ở trong cửa sổ Database: Chọn form /chọn tên form/ chọn Design ( Có

thể click chuột phải rồi chọn Design).

Khi đang ở chế độ Form view: Chọn View/ Form Design

#### **4.1.2. Chế độ Form View**

Dùng để nhập, thay đổi và xem dữ liệu. Trong chế độ Form View người sử dụng có thể xem tất cả các trường của một bản ghi tại một thời điểm.

Khi đang ở trong cửa sổ Database: Chọn form /chọn tên form/ chọn Open ( Có thể click chuột phải rồi chọn Open).

Khi đang ở chế độ Design view: Chọn View/ Form View.

#### **4.1.3. Chế độ hiển thị biểu mẫu dưới dạng bảng (Datasheet View)**

Dùng để nhập, thay đổi và xem dữ liệu trong biểu mẫu theo dạng bảng biểu.

Để mở chế độ hiển thị dạng Datasheet View chúng ta thực hiện như sau:

Khi đang ở chế độ Design View: Chọn View/Datasheet.

Khi đang ở chế độ Form View: Chọn View/ Datasheet View.

#### **4.1.4. Chế độ hiển thị Print Preview**

Dùng để xem biểu mẫu trước khi quyết định in ấn. Trong chế độ Print Preview sẽ duy trì hình dạng trình bày dữ liệu đã được thiết kế trước đó.

Khi đang ở trong cửa sổ Database: Chọn form /chọn tên form/ Chọn File/Print Preview.

### **4.2. Các thành phần biểu mẫu trong chế độ Design View**

Khi muốn thiết kế biểu mẫu thì người sử dụng phải làm việc trong chế độ Design View khi đó biểu mẫu có các thành phần chính sau:

**Thước(Ruler):** Điều chỉnh kích thước của các điều khiển.

**Tiêu đề form (form header):** Sử dụng để trình bày tiêu đề của form, tiêu đề form luôn được trình bày phần trên cùng, đầu tiên của biểu mẫu và trang in biểu mẫu.

**Chân form (Form Footer):** Sử dụng để trình bày chân của form, chân form luôn được trình bày phần dưới cùng, xuất hiện cuối biểu mẫu và trang in biểu mẫu.

**Tiêu đề trang (Page header):** Sử dụng để chứa tiêu đề trang

**Chân trang (Page footer):** Sử dụng để chứa chân trang nhưng xuất hiện phần trước của Form footer trong trang biểu mẫu in.

### *☞ Chú ý*

Page header và Page footer chỉ xuất hiện trong trang biểu mẫu in nên chúng không có những tính chất thông thường như Form header và Form footer.

Chọn View/ Page header/ footer (Nếu 2 thành phần này chưa xuất hiện trên biểu mẫu).

**Chi tiết form (Detail):** Đây là phần rất quan trọng chứa các điều khiển nhằm trình bày các dạng dữ liệu từ các bảng dữ liệu hoặc các truy vấn. Các loại điều khiển có thể là điều khiển buộc, không buộc hoặc tính toán.

## **5. CÁC LOẠI ĐIỀU KHIỂN**

Tất cả thông tin trên biểu mẫu được chứa trong những đối tượng gọi là điều khiển (Control). Điều khiển có thể dùng để thể hiện dữ liệu, thực hiện các hành động hoặc thiết kế biểu mẫu đẹp mắt. Trong ACCESS hệ thống định nghĩa một số loại điều khiển như sau:

*Điều khiển nhãn (Label).*

*Điều khiển hộp văn bản (Text box).*

*Điều khiển nhóm lựa chọn (Option group).*

*Điều khiển loại hộp Combo (Combo box) và hộp danh sách (List Box).*

Ngoài ra còn có một số điều khiển khác như command button.....

Khi tạo lập điều khiển, chúng ta thường xác định hình thức dữ liệu trình bày trong chúng. Có những điều khiển lấy dữ liệu từ các trường trong bảng hay truy vấn, có điều khiển chỉ dùng vào mục đích trang trí, làm tiêu đề, có những điều khiển lấy dữ liệu từ một biểu thức nào đó. Vì vậy người ta phân ra thành ba nhóm điều khiển chính:

Điều khiển buộc ( Bound control)

Điều khiển không buộc ( Unbound control)

Điều khiển tính toán ( Caculated control)

**5.1. Điều khiển bị buộc (bound), không buộc (unbound) và tính toán được (calculated).**

Khi tạo một điều khiển trong biểu mẫu thì phải xác định nó lấy dữ liệu từ nguồn nào để thể hiện.

***Ví dụ***

Tạo một điều khiển loại hộp văn bản (Text box) để hiển thị tên các mặt hàng, chúng ta phải chỉ định cho điều khiển lấy dữ liệu trong trường TEN\_HANG của bảng MAT\_HANG. Hộp văn bản này gọi là bị buộc.

Điều khiển cũng có thể thể hiện những thông tin không có trong CSDL (Không bị buộc).

***Ví dụ:*** Tạo tiêu đề cho biểu mẫu...

**Tóm lại**

Điều khiển bị buộc (Bound Control) là điều khiển mà nguồn dữ liệu của nó lấy từ một trường trong bảng hoặc truy vấn. Trong biểu mẫu dùng điều khiển buộc vào các trường để hiển thị nội dung hoặc cập nhật các trường của CSDL, các giá trị cập nhật có thể là: Văn bản, Date, Number, yes/No, Picture, chart trong đó dạng văn bản là phổ biến nhất.

Điều khiển không bị buộc (Unbound Control) là điều khiển không lấy dữ liệu từ một nguồn nào cả là điều khiển không bị buộc. Dùng điều khiển không buộc để trình bày thông tin không có trong các bảng hay rút được từ truy vấn.

Điều khiển tính toán (Calculated Control) là điều khiển mà nguồn dữ liệu của nó không phải là một trường mà là một biểu thức gọi là điều khiển tính toán (Calculated Control). Chúng ta qui định giá trị xuất hiện trong điều khiển bằng cách lập biểu thức cho nó. Biểu thức này là nguồn dữ liệu của điều khiển. Trong biểu thức có thể dùng các toán tử (+, -, =...) với tên điều khiển.

***Ví dụ:*** Tạo một điều khiển  $THANHTIEN = SOLUONG * DONGIA$

**5.2. Tạo điều khiển loại hộp văn bản ( text box)**

Text box có thể là một điều khiển bị buộc, không buộc hoặc tính toán.

***Tạo hộp văn bản bị buộc (Bound Text box).***

Chúng ta buộc điều khiển Text box vào một trường bằng cách chỉ định điều khiển đó lấy dữ liệu trên trường nào. Chọn trường để buộc vào điều khiển bằng cách Click biểu



tượng Field List để mở danh sách các trường của bảng hay truy vấn làm nền tảng cho biểu mẫu. Theo mặc định như vậy thì hệ thống sẽ tạo một điều khiển loại Text box.

Một cách khác dùng hộp dụng cụ Toolbox để tạo điều khiển và sau đó gõ tên trường muốn buộc vào hộp văn bản.

### ***Mở hộp danh sách trường***

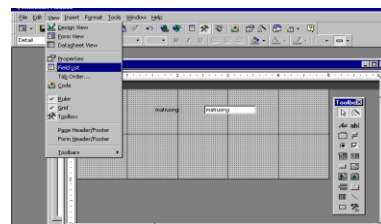
Mở biểu mẫu trong chế độ Design (Tạo biểu mẫu trước).

Trong cửa sổ Database Chọn Form/New (Chọn bảng hoặc truy vấn làm nền cho biểu mẫu).

Chọn View/Field List hoặc click vào biểu tượng Field List trên thanh công cụ).

### ***Tạo một Text Box bị buộc***

Từ danh sách trường chọn một hoặc nhiều trường kéo và đưa vào biểu mẫu.



## **5.3. Tạo một điều khiển khác dùng hộp công cụ**

Dùng hộp công cụ (Toolbox) để tạo những điều khiển không buộc (Unbound control) hoặc để tính toán. Đặc biệt dùng các tính năng của hộp này để tạo những điều khiển bị buộc khác ngoài buộc khác ngoài (Text box). Để bật hộp công cụ ta chọn View/Toolbars.

### ***Tạo điều khiển dùng hộp công cụ***

Click vào biểu tượng công cụ tương ứng với điều khiển muốn tạo.

Tạo điều khiển bị buộc bằng cách chọn một trường trong danh sách trường (Field List) và kéo vào biểu mẫu.

Hoặc tạo điều khiển không buộc hay dùng để tính toán bằng cách click vào một vị trí trên biểu mẫu.

### ***Tạo điều khiển dùng để tính toán***

Nếu muốn trình bày kết quả của một phép toán trong biểu mẫu, khai báo nguồn dữ liệu của điều khiển là một biểu thức. Sau này mỗi lần mở biểu mẫu, ACCESS tính toán lại kết quả của vùng dữ liệu được cập nhật mới nhất từ các bảng.

### ***Tạo điều khiển nhãn (Label Control)***

Muốn trình bày một chuỗi ký tự trên biểu mẫu như để làm tiêu đề... chúng ta dùng loại điều khiển gọi là điều khiển nhãn (label control). Nhãn không thể hiện dữ liệu của một trường hay biểu thức nào, chúng luôn luôn là không bị buộc.

Click vào biểu tượng Label trong Toolbox

Click vị trí muốn đặt nhãn trên biểu mẫu.

***Thay đổi các thuộc tính của điều khiển.***

Sau khi tạo biểu mẫu, biến đổi một số thuộc tính của điều khiển có thể hoàn thiện thêm về thiết kế và hình thức trình bày của số liệu.

Nhấp kép vào điều khiển.

Trong hộp lựa chọn trên đầu bảng thuộc tính, chọn một lớp thích hợp để làm việc.

*All Properties:* Trình bày tất cả các thuộc tính của điều khiển

*Data Properties:* Ấn định các đặc tính thể hiện dữ liệu trong điều khiển như giá trị mặc nhiên, định dạng số.

*Even Properties:* Qui định một tập lệnh (Macro) hay thủ tục (Procedure)..

*Layout Properties:* Định nghĩa các hình thức của điều khiển như cao, rộng.

*Other Properties:* Một số thuộc tính khác như tên điều khiển, thông tin mô tả ở dòng trạng thái.

Click chọn một trong các thuộc tính của bảng để thực hiện.

#### **5.4. Thiết lập một số thuộc tính bổ sung**

##### ***5.4.1. Các thuộc tính hỗ trợ nhập liệu***

Thuộc tính **Default value:** Gán giá trị mặc định vào nội dung trình bày trong điều khiển.

Thuộc tính **ValidationRule** và **ValidationRule Text:** Kiểm tra tính hợp lệ khi nhập dữ liệu cho một điều khiển và thông báo lỗi nếu dữ liệu không hợp lệ

##### ***5.4.2. Các thuộc tính giống lề***

Thuộc tính **General:** Giống hàng văn bản theo lề trái, dữ liệu số và ngày tháng theo lề phải.

Thuộc tính **Left:** Giống hàng văn bản theo lề trái.

Thuộc tính **Center**: Xác lập hàng văn bản ở chính giữa.

Thuộc tính **Center**: Gióng hàng văn bản theo lề phải.

#### **5.4.3. Các thuộc tính màu sắc**

Thuộc tính **BackColor**: Thiết lập màu nền cho điều khiển hay cho biểu mẫu

Thuộc tính **ForeColor**: Thiết lập màu cho hàng chữ trong điều khiển.

Thuộc tính **Bodercolor**: Thiết lập màu cho khung bao quanh trong điều khiển.

#### **5.4.3. Các thuộc tính khung bao**

Thuộc tính **BoderStyle**: Thiết lập loại khung cho điều khiển

Thuộc tính **BoderWidth**: Thiết lập độ dày hay đậm của khung bao.

Thuộc tính **BoderColor**: Thiết lập màu của khung bao.

## **6. NÂNG CẤP BIỂU MẪU**

Access cung cấp nhiều loại điều khiển để làm cho biểu mẫu dễ sử dụng và có nhiều hiệu ứng tốt. Có thể thay thế một Textbox với một List box hay Combo box để chọn từ các giá trị có sẵn thay vì buộc người sử dụng phải nhớ để nhập giá trị vào.

### **6.1. Dùng điều khiển List box và Combo box để tạo danh sách chọn lựa.**

Trong nhiều trường hợp, chọn một danh sách có sẵn thường tiện lợi hơn phải nhớ để gõ vào từ bàn phím. Access cung cấp hai khả năng điều khiển tạo danh sách chọn lựa: List box và Combo box.

List box đơn giản là một danh sách để chọn, combo box tương tự như một text box và một combo box kết hợp vào một điều khiển, có nghĩa là có thể gõ thẳng giá trị vào text box hay chọn từ một danh sách có sẵn.

**Ưu điểm của List box:** Danh sách luôn được thể hiện và người dùng chỉ được phép chọn trong danh sách, do đó dữ liệu nhập luôn luôn là hợp lệ.

**Ưu điểm của Combo box:** Danh sách không được thể hiện cho đến khi người dùng mở hộp điều khiển, do đó ít tốn chỗ trên biểu mẫu hơn.

### **6.2. Tạo List box và Combo box không sử dụng Wizard**

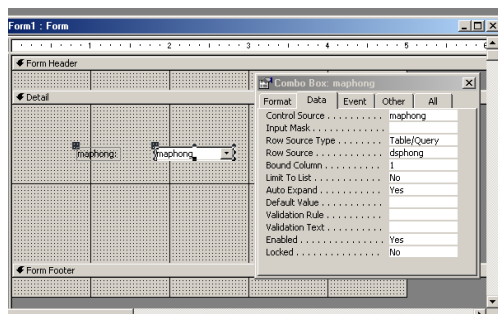
Tắt chức năng Control Wizard trong menu View hoặc trong thanh công cụ.

Click biểu tượng List box hoặc Combo box trong hộp công cụ.

Tạo điều khiển buộc bằng cách click biểu tượng Field list trên thanh công cụ để mở danh sách trường làm nền cho biểu mẫu. Chọn một trường trong danh sách kéo vào biểu mẫu, hoặc tạo điều khiển không buộc bằng cách click biểu mẫu nơi muốn đặt điều khiển.

Lập các thuộc tính của điều khiển để định nghĩa các hàng dùng làm chọn lựa trong danh sách.

<b>Muốn danh sách thể hiện</b>	<b>Lập thuộc tính Row Source Type thành</b>	<b>Lập Row Source thành</b>
Các hàng từ một bảng hay truy vấn	Table/Query (Default)	Tên của bảng hay truy vấn đó
Các hàng lấy từ lệnh Select của SQL	Table/Query (Default)	Một câu lệnh SQL
Một danh sách với các giá trị do người dùng tự đặt	Value List	Danh sách các giá trị đó phân cách nhau bởi dấu chấm phẩy
Tên các trường trong một bảng hoặc truy vấn	Field List	Tên của bảng hoặc truy vấn đó
Các giá trị định nghĩa bởi một hàm trong Access Basic	Tên hàm xây dựng	Để trống



### 6.2.1. Định nghĩa các hàng trong danh sách

Định nghĩa nguồn dữ liệu để hình thành các hàng dùng làm trong danh sách lựa chọn trong List box và Combo box bằng cách lập thuộc tính thích hợp cho RowSource Type và RowSource của điều khiển, hai thuộc tính này phối hợp với nhau để tạo nên các hàng trong danh sách.

Thuộc tính RowSource Type cho biết danh sách sẽ hình thành từ nguồn dữ liệu loại nào, thuộc tính RowSource là thuộc tính xác định nguồn dữ liệu của danh sách.

### ***6.2.2. Danh sách lấy dữ liệu từ bảng hay truy vấn***

Có thể tạo List box hay Combo box lấy dữ liệu từ một hoặc nhiều trường trong bảng ( hoặc truy vấn) để lập thành danh sách chọn lựa cho điều khiển. Thực hiện điều này bằng cách đặt thuộc tính RowSource Type thành Table/Query và RowSource thành tên bảng hay tên truy vấn có chứa dữ liệu để lập thành danh sách đó.

### ***6.2.3. Tạo danh sách nhiều cột***

Có thể tạo danh sách gồm hai hay nhiều cột để có thêm thông tin lựa chọn. Xác định số cột bằng cách lập thuộc tính Columcount và ColumnWidth cho điều khiển. Thuộc tính columcount quy định số cột và nếu trong RowSource là tên một bảng hay truy vấn thì các trường đầu tiên tương ứng số cột của bảng hay truy vấn đó (tính từ trái sang phải) sẽ được đưa vào danh sách. Thuộc tính columnwidth quy định bề rộng mỗi cột tính theo inch hay cm.

### ***6.2.4. Sử dụng danh sách giá trị***

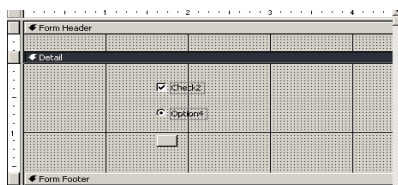
Trường hợp điều khiển chỉ có một số nhỏ chọn lựa và các giá trị này không thay đổi, có thể dùng phương pháp đơn giản là danh sách giá trị. Danh sách này chúng ta tự lập bằng cách đưa các giá trị dùng làm chọn lựa vào thuộc tính RowSource của điều khiển. Sử dụng dấu chấm phẩy để phân cách các mục chọn lựa của danh sách.

## **6.3. Sử dụng hộp kiểm tra (check box), nút chọn lựa (Option button), nút bật tắt (Toggle button).**

Hộp kiểm tra, nút chọn lựa, nút bật tắt thường được sử dụng như các điều khiển độc lập để nhận các chọn lựa Yes/No. Các điều khiển này thực chất chỉ khác nhau về hình thức, do đó chúng ta có thể sử dụng bất cứ nút nào.

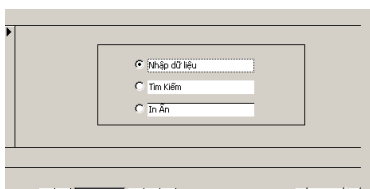
Khi được chọn điều khiển này biểu diễn trị Yes ( hay True), ở trạng thái không chọn thì nhận giá trị No ( hay False).

- Check box** : Được chọn khi có dấu x trong hộp.
- Option button** : Được chọn khi có hạt đậu trong nút.
- Toggle button** : Được chọn khi nó trông như bị nhấn xuống.



#### 6.4. Sử dụng nhóm chọn lựa (Option group)

Chúng ta có thể dùng nhóm chọn lựa để trình bày một số nhỏ các chọn lựa trên biểu mẫu. Nhóm chọn lựa gồm một khung nhóm (Group frame) có gắn nhãn và một số check box, option button hay toggle button.



Nếu nhóm lựa chọn được buộc vào một trường, chỉ có khung nhóm là bị buộc vào trường đó, không phải các thành phần để chọn lựa trong khung. Khung nhóm có thể không bị buộc vào trường nào hoặc chứa một biểu thức, mỗi lần chỉ có thể chọn một thành phần trong nhóm.

Khi sử dụng hộp kiểm tra, nút chọn lựa và nút bật tắt trong nhóm chọn lựa, thuộc tính của chúng khác với như khi được dùng với các điều khiển độc lập. Mỗi phần tử trong nhóm chọn lựa không có thuộc tính Control Source mà thay bằng thuộc tính Option Value. Lập thuộc tính Option Value của mỗi phần tử dùng làm chọn lựa trong nhóm thành một trị số có nghĩa đối với trường mà khung nhóm được buộc vào. Khi một phần tử của nhóm được chọn, Access đưa giá trị lập trong Option value của phần tử đó cho điều khiển Option Group. Đến lượt Option Group trả lại giá trị đó cho trường mà nó bị buộc vào.

## 7. BIỂU MẪU DỰA TRÊN NHIỀU BẢNG

### 7.1. Biểu mẫu phụ (Subform)

Biểu mẫu phụ là phương pháp để đưa thông tin từ nhiều bảng vào một biểu mẫu. Biểu mẫu phụ có nghĩa là một biểu mẫu được lồng trong biểu mẫu khác. Trong Access biểu mẫu chính gọi là Main form, biểu mẫu nằm trong Main form gọi là biểu mẫu phụ (Sub form). Khi dùng biểu mẫu phụ chúng ta dễ nhận thấy mối quan hệ giữa các bản ghi của hai hay nhiều bảng.

Biểu mẫu phụ đặc biệt hữu hiệu khi dùng để hiển thị dữ liệu từ nhiều bảng hay truy vấn có quan hệ một-nhiều với nhau. Biểu mẫu chính đại diện cho bên một, biểu mẫu phụ đại diện cho bên nhiều.

### **7.2. Các loại biểu mẫu phụ**

Khi tạo biểu mẫu phụ chúng ta có thể thiết kế nó thành dạng bảng, hoặc dạng biểu mẫu, hoặc cả hai dạng trên.

*Biểu mẫu dạng bảng:* Là loại dễ tạo nhất và có thể sử dụng như bất kỳ bảng nào khác như sắp xếp....

*Biểu mẫu phụ dạng biểu mẫu:* Cho chúng ta thực sự linh hoạt và mềm dẻo khi thiết kế hơn.

### **7.3. Thiết kế biểu mẫu phụ**

Thông thường chúng ta dùng bảng hay truy vấn làm nguồn dữ liệu cho biểu mẫu chính, một bảng hay truy vấn khác làm nguồn dữ liệu cho biểu mẫu phụ. Nếu dữ liệu trong biểu mẫu chính và biểu mẫu phụ có liên quan với nhau, chúng ta cần đánh giá một số vấn đề sau:

Các bảng hoặc truy vấn có quan hệ một-nhiều với nhau không? Nếu dùng biểu mẫu phụ để thể hiện quan hệ một-nhiều, chúng ta nên dùng bảng bên một đối với bảng chính, bảng bên nhiều đối với bảng phụ.

Các bảng hoặc truy vấn làm nguồn dữ liệu cho biểu mẫu chính/phụ có các trường liên kết không? Access dùng trường kết nối để giới hạn số lượng bản ghi thể hiện trong biểu mẫu phụ.

### **Cách tạo biểu mẫu chính/phụ**

Thiết kế hai biểu mẫu riêng biệt, sau đó kéo biểu mẫu phụ vào biểu mẫu chính.

## Thiết kế biểu mẫu chính

Tạo biểu mẫu chính, dành chỗ trên biểu mẫu này để chứa biểu mẫu phụ.

Lưu và đóng biểu mẫu chính.

### Thiết kế biểu mẫu phụ

Có thể thiết kế biểu mẫu phụ để chỉ thể hiện dữ liệu dưới dạng bảng, biểu mẫu này có cả hai khả năng trên.

Tạo biểu mẫu mới, lập hai thuộc tính ViewAllowed và Default View của biểu mẫu tùy theo yêu cầu sử dụng như sau:

**Biểu mẫu phụ chỉ trình bày dưới dạng bảng:** Đặt các trường trên biểu mẫu theo thứ tự chúng ta muốn chúng xuất hiện trong bảng. Lập cả hai thuộc tính ViewAllowed và Default View thành Datasheet.

**Biểu mẫu phụ chỉ trình bày dữ liệu dưới dạng biểu mẫu:** Sắp đặt các điều khiển như trên. Lập thuộc tính ViewAllowed thành Form và Default View thành Single form hay Continuous form.

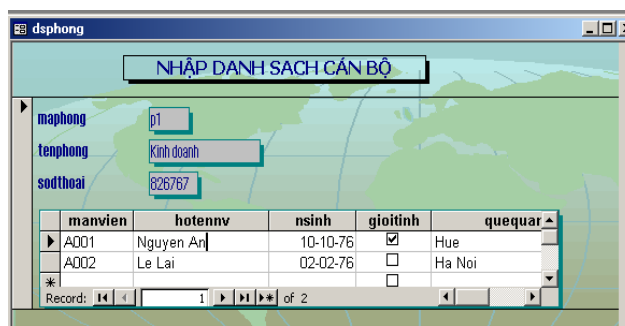
### Đưa biểu mẫu phụ vào biểu mẫu chính

Mở biểu mẫu chính ở chế độ Design View

Chuyển sang cửa sổ Database, nhấn F11

Kéo biểu mẫu phụ từ cửa sổ Database và đặt vào một vị trí trên biểu mẫu chính.

Di chuyển biểu mẫu phụ đến vị trí khác, thay đổi nội dung nhãn hoặc kích thước nếu cần. Chuyển sang chế độ Form View để xem kết quả.



## 7.4. Liên kết biểu mẫu chính và biểu mẫu phụ



Trong chế độ Design View của biểu mẫu chính, mở bảng thuộc tính của điều khiển biểu mẫu phụ.

Lập thuộc tính LinkChildFields thành tên trường nối kết trong biểu mẫu phụ, nếu có nhiều trường nối kết, phân cách nhau bởi dấu phẩy.

Lập thuộc tính LinkMasterFields thành tên trường nối kết hoặc tên điều khiển trong biểu mẫu chính, nếu có nhiều trường nối kết, phân cách nhau bởi dấu phẩy.

## 1. TỔNG QUAN VỀ BÁO CÁO

Báo cáo là phương thức hữu hiệu giúp người sử dụng trình bày dữ liệu dưới dạng đầy đủ và dễ hiểu, nhanh chóng, đẹp mắt để khi in ấn. Người sử dụng có thể tích hợp trong báo cáo các dạng thức trình bày dữ liệu khác nhau như: Hình ảnh, biểu đồ, văn bản.....

Báo cáo được xây dựng trên một nguồn dữ liệu đó là bảng hoặc truy vấn, một câu lệnh SQL hoặc một dạng biểu mẫu nào đó

### 1.1. Các dạng mẫu của báo cáo

**Báo cáo dạng cột (columnar):** báo cáo dạng này sẽ được trình bày theo dạng một cột và kèm theo phần nhãn của mỗi cột dữ liệu bên trái, mỗi dòng tương ứng với một trường dữ liệu.

**Báo cáo dạng hàng (Tabular):** Báo cáo sẽ trình bày dữ liệu theo dạng bảng bao gồm nhiều hàng và nhiều cột.

**Báo cáo dạng nhóm/ Tổng (Group/Total):** Báo cáo dạng này sẽ tổ chức dữ liệu thành các nhóm, mỗi nhóm sẽ trình bày dữ liệu theo dạng Tabular. Người sử dụng có thể nhóm dữ liệu theo cấp và có thể tính toán giá trị tổng cho mỗi nhóm và một giá trị tính tổng cho toàn bộ các nhóm

**Báo cáo dạng biểu đồ ( Chart)**

**Báo cáo dạng nhãn ( Label Report)**

**Báo cáo với báo cáo con**

### 1.2. Các chế độ hiển thị của báo cáo

Báo cáo có thể được trình bày theo 3 chế độ sau

**Report design:** Chế độ thiết kế báo cáo.

**Layout PreView:** Chế độ trình bày dữ liệu trong báo cáo.

**Print PreView:** Chế độ xem hình thức báo cáo trước khi in ấn.

## 2. TẠO BÁO CÁO SỬ DỤNG CÔNG CỤ AUTO REPORT VÀ REPORT WIZARD

### 2.1. Tạo báo cáo sử dụng Auto report

Click biểu tượng Report trong cửa sổ database ( hoặc chọn View/Report)

Chọn New

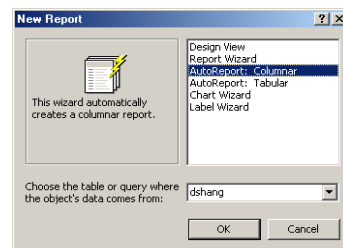
Chọn bảng hoặc Truy vấn làm nguồn dữ liệu báo cáo .

Chọn AutoReport Columnar: Nếu muốn báo cáo hiển thị dạng cột.

AutoReport Tabular: Nếu muốn báo cáo hiển thị dạng hàng

Chọn OK

Lưu Báo cáo.



### 2.2. Tạo báo cáo sử dụng Report Wizard

Click biểu tượng Report trong cửa sổ database ( hoặc chọn View/Report)

Chọn New

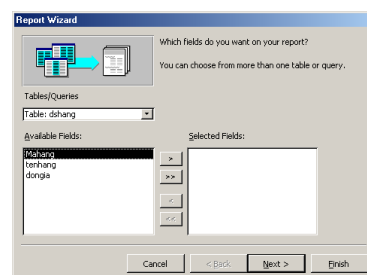
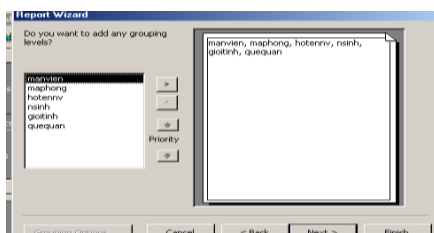
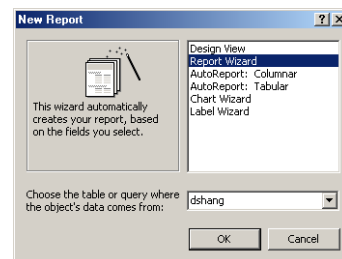
Chọn bảng hoặc Truy vấn làm nguồn dữ liệu báo cáo .

Chọn Report Wizard

Chọn OK

Chọn các trường cần thiết cho báo cáo.

Chọn Next



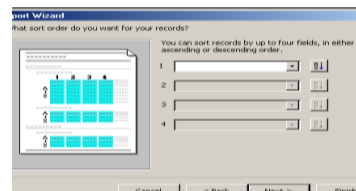
Chọn các trường cần nhóm, chọn Next

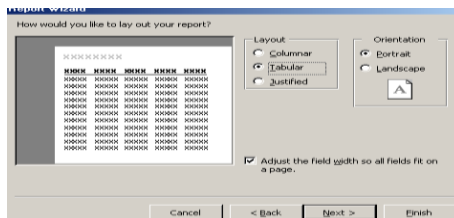
Chọn các trường cần sắp xếp, chọn Next

Chọn dạng thể hiện của Report, chọn Next

Chọn nền thể hiện của Report, chọn Next

Đặt tiêu đề cho Report, chọn Finish





### 3. TẠO ĐIỀU KHIỂN TRONG BÁO CÁO

#### 3.1. Tạo điều khiển Text Box (Hộp văn bản)

Text box có nhiều tính năng, có thể là điều khiển bị buộc, không buộc hay dùng tính toán. Nếu trong báo cáo có cả 3 loại điều khiển này, chúng ta nên ưu tiên điều khiển bị buộc trước.

##### *Tạo một text box bị buộc và không buộc*

Chúng ta buộc điều khiển Text box vào một trường bằng cách chỉ định trường cho text box đó lấy dữ liệu. Text box đó lấy dữ liệu có thể thực hiện bằng cách kéo trường muốn buộc vào điều khiển từ danh sách trường (Field List) vào biểu mẫu đang thiết kế.

Cách khác để tạo Text box là dùng hộp công cụ (Toolbox), sau đó gõ tên trường muốn buộc vào hộp văn bản hoặc bảng thuộc tính của điều khiển.

Dùng danh sách trường là phương pháp tốt nhất để tạo một điều khiển Text box bị buộc vì hai lý do sau:

Điều khiển đó được hệ thống tự động gắn nhãn và nhãn lấy tên trường được kéo làm tiêu đề.

Text box bị buộc đó thừa kế các thiết lập thuộc tính của trường từ bảng hay truy vấn.

Muốn chuyển một điều khiển không buộc thành bị buộc, lập thuộc tính Control Source của điều khiển thành một trường.

##### *Một số thao tác khi thực hiện thiết kế báo cáo.*

Mở báo cáo trong chế độ Design View

Chọn Field List từ menu View (Hoặc click biểu tượng Field List trên thanh công cụ)

Tạo điều khiển Text box bị buộc.

### ***Mở bảng danh sách trường, chọn trường hoặc các trường muốn đặt vào báo cáo***

Chọn một trường, click vào trường đó.

Chọn nhiều trường liên nhau, click trường đầu, giữ phím shift, click trường cuối.

Chọn nhiều trường không liên nhau, giữ phím Ctrl và lần lượt click từng trường.

Chọn tất cả các trường trong danh sách, nhấp kép vào thanh tiêu đề của danh sách trường.

Kéo trường (hoặc các trường) được chọn và đặt vào một vị trí trên mẫu báo cáo.

Click biểu tượng Simple preview trên thanh công cụ để xem kết quả.

### **3.2. Tạo các điều khiển khác dùng hộp công cụ**

Muốn tạo các điều khiển không buộc hay dùng để tính toán, phải dùng công cụ trong Toolbox.

Chọn View/Toolbars để hiển thị thanh công cụ

#### ***Tạo điều khiển dùng Toolbox***

Click công cụ tương ứng loại điều khiển muốn tạo trong báo cáo.

Tạo điều khiển bị buộc bằng cách chọn một trường trong Field list và kéo nó vào trong báo cáo.

### **3.3. Tạo điều khiển dùng tính toán**

Click vào biểu tượng Text box trong hộp công cụ

Click vào một vị trí trên báo cáo. Access tự động gán nhãn cho điều khiển vừa tạo, tiêu đề mặc nhiên thường có dạng “Field0”, có thể thay đổi tiêu đề này theo ý thích.

Đưa con trỏ vào bên trong Text box.

Gõ dấu = , theo sau là biểu thức muốn lập

***Ví dụ:*** =[SOLUONG]\*[DONGIA]

Click vào Sample Preview để xem kết quả.

### **3.4. Tạo điều khiển nhãn**

Nhãn là một điều khiển không buộc, nội dung nhãn không thay đổi từ trang này qua trang khác hay từ bản ghi này qua bản ghi khác.

Click biểu tượng Label trong Toolbox.

Click vào một vị trí trên báo cáo để tạo nhãn. Nhãn được tạo như vậy sẽ có kích thước tự mở rộng khi gõ nội dung vào.

Nếu muốn trình bày văn bản thành nhiều dòng, bấm Ctrl+Enter cuối dòng thứ nhất.

#### 4. CÁC THUỘC TÍNH CỦA ĐIỀU KHIỂN TRONG BÁO CÁO

Thuộc tính xác định các đặc trưng của đối tượng, mỗi điều khiển trong báo cáo cũng có những thuộc tính riêng. Muốn mở bảng thuộc tính của điều khiển, chọn điều khiển đó và click biểu tượng Properties trên thanh công cụ.

Ta xét một số thuộc tính sau:

**Cangrow:** Dùng thuộc tính này để làm cho Text box có thể tự điều chỉnh kích thước theo phương dọc đối với khối dữ liệu chứa trong trường nó bị buộc.(chọn Yes).

**CanShrink:** Khi Text box không có dữ liệu hoặc dữ liệu là chuỗi rỗng Access sẽ chừa trống chỗ đó trên giấy. Điều này có thể làm cho báo cáo quá trống trải nếu có nhiều chỗ như vậy. Chúng ta lập thuộc tính của Text box này thành Yes.

**HideDuplicate:** Dùng thuộc tính này để che Text box khi giá trị trong đó trùng bản ghi trước

#### 5. SẮP XẾP VÀ TẬP HỢP DỮ LIỆU THEO NHÓM

Sắp xếp là phương pháp phổ biến nhằm tổ chức dữ liệu theo một trật tự nào đó để tìm kiếm và phân loại thông tin.

##### 5.1. Sắp xếp dữ liệu

Khi in báo cáo người dùng thường muốn tổ chức các bản ghi theo một trật tự nào đó. Ví dụ in danh sách cán bộ theo thứ tự giảm dần của lương.

*Các bước thực hiện sắp xếp trên báo cáo.*

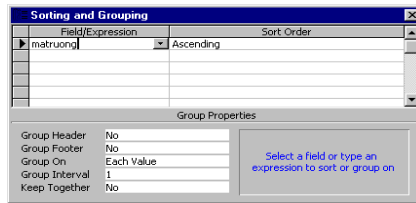
Mở báo cáo ở chế độ Design View.

Chọn Sorting And Grouping trong menu View.

Trong hộp thoại.

*Field/Expression:* Chỉ định sắp xếp theo trường hoặc biểu thức nào đó.

*Sort Order:* Chọn Tăng dần hoặc giảm dần.



## 5.2. Nhóm dữ liệu

Trong nhiều báo cáo, sắp xếp các bản ghi không cũng chưa đủ mà cần phân thành các nhóm. Nhóm là tập hợp các bản ghi cùng với thông tin tóm lược tiêu biểu cho một thể loại thông tin. Một nhóm thường được cấu tạo như sau:

Tiêu đề nhóm (group header), nhóm con (nếu có), các bản ghi chi tiết và chân nhóm (Group footer)

*Tiêu đề nhóm 1*

*Tiêu đề nhóm 2*

*Tiêu đề nhóm 3*

.....

.....

*Tiêu đề nhóm 10*

*Các bản ghi chi tiết*

*Chân nhóm 10*

.....

.....

*Chân nhóm 3*

*Chân nhóm 2*

*Chân nhóm 1*

## Chương 7

## MACRRO

### 1. KHÁI NIỆM

Macro trong MS Access là tập hợp các lệnh (Hành động, hành động.....) được định sẵn nhằm tự động thực hiện chuỗi các tác vụ nào đó mà không cần sự can thiệp từng bước của người sử dụng. Macro có thể liên kết các đối tượng trong tập tin cơ sở dữ liệu (CSDL) như: Table, Query, form, report..... nhằm tạo ra các ứng dụng để khai thác có hiệu quả..

Macro được dùng khi có các hành động nào thường xuyên lặp lại trong MS Access hoặc được dùng khi cần kết hợp các hành động đơn giản nhằm giải quyết một vấn đề nào đó khi xây dựng các ứng dụng. Việc tự động hoá các hành động này bởi macro sẽ được thực hiện một cách nhanh chóng và chính xác.

### 2.TẠO VÀ THI HÀNH MỘT MACRO

#### 2.1.Tạo một macro

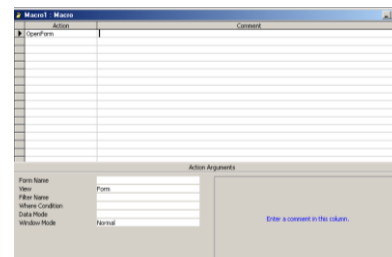
Tại cửa sổ database, chọn đối tượng Macro/ New

*Trong mục Action* : chọn các hành động cho Macro.

*Tong mục Action arguments*: Chọn các tham số

tương ứng cho hành động.

Lưu trữ Macro: File/save.



#### 2.2.Thi hành macro

Tại cửa sổ database: Chọn đối tượng Macro/Run.

### 3. CÁC HÀNH ĐỘNG VÀ CÁC THAM SỐ

Ms Access cung cấp một số hành động để tạo macro, mỗi hành động thực hiện một tác vụ nào đó trên CSDL và tùy thuộc vào tham số của mỗi hành động.

**Open table:** Mở 1 bảng dữ liệu

*Table name:* Tên bảng cần mở

*View:* Chọn dạng thể hiện của bảng (Datasheet/ Design/ Print Preview)



*Data mode:*

Add: Dùng để nhập dữ liệu

Edit: Dùng để thêm, xoá, sửa dữ liệu trong bảng

Read Only: Mở bảng để chỉ xem dữ liệu

**Open Query:** Mở 1 truy vấn

*Query name:* Tên truy vấn cần mở

*View:* Chọn dạng thể hiện của bảng (Datasheet/ Design/ Print Preview)

*Data mode:*

Add: Dùng để nhập dữ liệu

Edit: Dùng để thêm, xoá, sửa dữ liệu trong bảng

Read Only: Mở bảng để chỉ xem dữ liệu

**Open Form:** Mở 1 biểu mẫu

*Form name:* Tên biểu mẫu

*View:* Chọn dạng thể hiện ( Form/ Design/ Print preview/ Datasheet)

*Filter name:* Tên Query lọc các dữ liệu để hiển thị trong form.

*Where condition:* Điều kiện lọc dữ liệu hiển thị trong form.

*Data mode:*

Add : Dùng để nhập dữ liệu

Edit : Dùng để thêm, xoá, sửa dữ liệu trong bảng

Read Only : Mở bảng để chỉ xem dữ liệu

Window mode:

Normal : Dạng cửa sổ form bình thường.

Hidden : Dạng cửa sổ form được ẩn đi.

Icon : Cửa sổ form thu nhỏ thành 1 biểu tượng.

Dialog : Dạng hộp thoại.

**Open Report:** Mở 1 báo cáo

*Report name:* Tên báo cáo

*View:* Chọn kiểu in

Print preview: In ra màn hình.

**Design:** Dạng thiết kế báo cáo

**Print:** In ra máy in

**Filter name:** Tên Query lọc các dữ liệu để hiển thị trong Report.

**Where condition:** Điều kiện lọc dữ liệu hiển thị trong Report.

**Run macro:** Thực hiện một tập lệnh

**Macro name:** Tên macro cần thực hiện

**Repeat count:** Số lần thực hiện macro sẽ lặp lại.

**Repeat Expression:** Biểu thức điều kiện để lặp lại khi thực hiện macro. Macro chỉ dừng khi khi biểu thức điều kiện nhận giá trị False.

**Open module:** Mở cửa sổ soạn thảo thủ tục trong 1 module.

**Module name:** Tên module chứa thủ tục cần mở.

**Procedure name:** Tên thủ tục sẽ mở.

**Run code:** Gọi thực hiện một hàm của Access Basic

**Function name:** Tên hàm cần thực hiện và các đối số của hàm.

**Run App:** Cho thực hiện một ứng dụng nào đó trong môi trường Windows

**Command line:** đường dẫn đến tập tin của một ứng dụng.

**Run SQL:** Cho thực hiện câu lệnh SQL

**SQL Statement:** Nội dung câu lệnh SQL

**Maximize:** Cực đại cửa sổ hiện thời

**Minimize:** Cực tiểu cửa sổ hiện thời thành một biểu tượng.

**Restore:** Phục hồi cửa sổ trở về kích thước cũ.

**Move size:** Di chuyển hoặc thay đổi kích thước cửa sổ hiện thời.

**Right:** Khoảng cách từ góc trên trái của cửa sổ này đến cạnh trái của cửa sổ chứa nó.

**Down:** Khoảng cách từ góc trên trái của cửa sổ này đến cạnh trên của cửa sổ chứa nó.

**Width:** Chiều rộng của cửa sổ này.

**Height:** Chiều cao của cửa sổ này.

**Stop Macro:** Dừng macro đang thực hiện

**Beep:** Phát tiếng kêu bíp

**Hourglass:** Đổi dạng con trỏ thành đồng hồ cát trong khi macro đang chạy

*Hourglass On:* Yes/No (Đổi/ Không đổi)

**Close:** Đóng một cửa sổ đang hoạt động

*Object Type:* Loại cửa sổ của đối tượng cần đóng như Table, Query, form, Report, Macro hoặc Module.

*Object name:* Tên của đối tượng cần đóng.

**Quit:** Thoát khỏi MS Access và trở về Windows

*Option*

Prompt: Hiện thị hộp thoại có lưu trữ không? Nếu đối tượng có thay đổi.

Save all: Lưu trữ tất cả mọi đối tượng.

Exit: Thoát mà không cần lưu trữ.

**Print:** In đối tượng hiện thời

*Print Range:* Phạm vi cần in ấn.

All: In tất cả các đối tượng

Selection: In phần trang được chọn

Pages: In các trang được chọn

*Page from:* Trang bắt đầu in

*Page to:* Trang kết thúc in

*Print Quality:* Chất lượng in

*Copies:* Số bản cần in

*Collate Copies:* Có sắp xếp thứ tự các bản in theo trang.

**Msg Box:** Hiện thị hộp thông báo

*Message:* Câu thông báo cần hiển thị

*Beep:* Yes/ No: Có/ Không phát ra tiếng Bíp khi hiển thị hộp thông báo.

*Type:* Loại hộp thông báo.

*Title:* Tiêu đề của hộp thông báo.

**CancelEvent:** Huỷ bỏ một sự kiện đang thực hiện

**Requery:** Cập nhật dữ liệu cho một đối tượng đang hoạt động bằng cách cập nhật lại dữ liệu nguồn của đối tượng đó.

*Control name:* Tên của đối tượng cần cập nhật dữ liệu (Nếu không chỉ ra thì sẽ cập nhật lại dữ liệu nguồn của chính đối tượng đang hoạt động).

**Select Object:** Chọn đối tượng trong CSDL

*Object Type:* loại đối tượng cần chọn.

*Object name:* Tên đối tượng cần chọn

*In Database Window:* (Yes/No) Xác định MS access có chọn đối tượng trong cửa sổ CSDL không, mặc định là No.

**Set value:** Gán một giá trị cho 1 trường, 1 điều khiển, hoặc một thuộc tính trên một Form hoặc 1 Report.

*Item:* Tên trường, đối tượng hay thuộc tính muốn gán giá trị.

*Expression:* Biểu thức cần gán giá trị cho Item.

☛ **Chú ý:** Nếu tên trường, tên đối tượng, tên thuộc tính ở 1 Form hoặc 1 Report khác thì phải mô tả đầy đủ.

Trong Form khác: [Forms]![Tên Form]![Tên trường/Tên đối tượng]

Trong Report khác: [Reports]![Tên Report]![Tên trường/Tên đối tượng]

Đối với các thuộc tính

[Forms/Reports]![Tên Form/Tên Report]![Tên trường].[Tên thuộc tính]

**Add menu:** Tạo thêm một Drop Down Menu vào một menu bar cho một form hoặc Report.

*Menu name:* Tên của Drop Down Menu muốn thêm vào menu bar.

*Menu macro name:* Tên macro chứa các lệnh về việc tạo menu.

*Status bar:* Thông báo ở thanh trạng thái khi chọn menu này.

**Apply Filter:** Lọc (Truy vấn) các dữ liệu khi xử lý Table, Form, Report.

*Filter name:* Tên của truy vấn lọc dữ liệu.

*Where condition:* Điều kiện lọc dữ liệu.

**FindRecord:** Tìm bản ghi đầu tiên nằm trong phạm vi và thoả mãn điều kiện.

*Find What:* Nội dung dữ liệu cần tìm là một giá trị hoặc một biểu thức, nếu là biểu thức sẽ bắt đầu dấu "=".

*Where:* Qui định cách so sánh giá trị cần tìm với giá trị của trường.

*Any part of field:* Một phần bất kỳ của trường.

*Match Whole field:* Giá trị cần tìm bằng giá trị của trường.

*Start of field:* Giá trị cần tìm là phần đầu của trường.

*Match Case: Yes/No:* Có/Không phân biệt chữ in hoa và chữ in thường.

*Direction:* Quy định hướng tìm

*All:* Tìm toàn bộ

*Up:* Tìm từ bản ghi hiện thời lên phía trên.

*Down:* Tìm từ bản ghi hiện thời lên phía dưới.

*Search As Formatted:* Qui định việc tìm có dựa trên dữ liệu sau khi đã định dạng trong các trường hay không?

*Search in:* Qui định việc tìm trên trường hiện thời hoặc trên tất cả các trường.

*Current Field:* Tìm trên trường hiện thời.

*All Fields:* Tìm tất cả các trường.

*Find First: Yes/No:* Qui định tìm từ bản ghi đầu tiên hay tìm từ bản ghi hiện thời.

**Findnext:** Tìm bản ghi kế tiếp thoả mãn điều kiện tìm kiếm của lệnh FindRecord.

**CopyObject:** Sao chép một đối tượng trong tập tin CSDL hiện thời thành một đối tượng khác của tập tin CSDL khác trong MS Access.

*Destination Database:* Tên tập tin CSDL đích

*New name:* Tên mới của đối tượng sau khi sao chép.

*Source Object Type:* Kiểu của đối tượng nguồn.

*Source Object Name:* Tên của đối tượng nguồn.

**DeleteObject:** Xoá một đối tượng trong tập tin CSDL hiện thời.

*Object Type:* Kiểu của đối tượng.

*Object Name:* Tên của đối tượng

## 4. NHÓM TẬP LỆNH VÀ TẬP LỆNH CÓ ĐIỀU KIỆN

### 4.1. Nhóm tập lệnh

Là Macro chứa các macro con, thay vì tạo ra nhiều macro với nhiều tên khác nhau thì các macro này được gom lại thành một tên chung nhằm giảm bớt số lượng và thuận lợi trong quá trình sử dụng. Tuy nhiên các Macro được nhóm khi chúng có liên quan với nhau.

Có thể có nhiều Macro trong nhóm có cùng hành động, tuy nhiên chúng được phân biệt bởi tên Macro.

Đặt tên cho Macro ta thực hiện:

Tại chế độ thiết kế Macro:

View/Macro name

Đặt tên cho Macro tại cột Macro name.

Cách thực hiện 1 macro trong macro name

<Tên Macro Group>.<Tên Macro cần thực hiện>

## 4.2. Macro có điều kiện

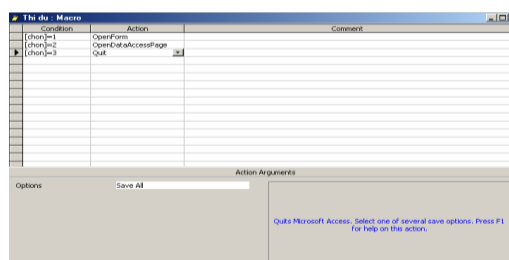
Là macro có chứa các điều kiện thi hành cho mỗi hành động.

### Cách tạo Macro có điều kiện

Tại chế độ thiết kế Macro: Chọn View/Conditions

Tại cột Condition : Đặt điều kiện thi hành cho mỗi hành động.

### Ví dụ



## 4.3. Áp dụng Macro cho form và Report

### 4.3.1. Quy tắc chung khi gọi một đối tượng

Đối với form : *Forms!*[Tên form]![Tên đối tượng]

Đối với Report : *Reports!*[Tên Report]![Tên đối tượng]

### 4.3.2. Các thuộc tính của một đối tượng

Muốn gắn một nút lệnh trên một biểu mẫu hoặc báo cáo với một Macro nào đó vào nút lệnh này ta thực hiện: Click chuột phải vào nút lệnh, chọn Properties và gắn Macro vào các hành động tương ứng.

On Enter: Macro thi hành khi nhấn Enter vào bên trong đối tượng

On Exit: Macro thi hành khi thoát khỏi đối tượng

On Got Focus: Thiết lập nhận biết khi có di chuyển con trỏ đến một form hoặc 1 trường trên form đang mở.

On Click: Macro thi hành khi click vào đối tượng

On Dbl Click: Macro thi hành khi Double click vào đối tượng.

On Mouse Down: Macro thi hành khi ấn và giữ chuột tại đối tượng

On Mouse Move: Macro thi hành khi di chuyển chuột ra khỏi đối tượng

On Mouse Up: Macro thi hành khi nhả chuột ra khỏi đối tượng.

On Key Down: Macro thi hành khi ấn và giữ một phím đối tượng.

On Key Press: Macro thi hành khi ấn một phím đối tượng.

On Key Up: Macro thi hành khi nhả một phím đối tượng.

#### ***4.3.3. Macro tự động thực hiện sau khi mở tập tin CSDL***

Chúng ta có thể tạo ra một Macro mà mỗi khi mở một tập tin CSDL thì Macro này tự động thực hiện.

Để tạo Macro tự động thực hiện ta tiến hành các thao tác sau:

Tạo macro

Lưu trữ Macro với tên AutoExec.

## **5. THIẾT KẾ MENU TRONG ACCESS**

Trong các ứng dụng, thường chúng ta phải tổ chức Menu để cho phép người sử dụng thực hiện các hành động thông qua các chức năng trên Menu này.

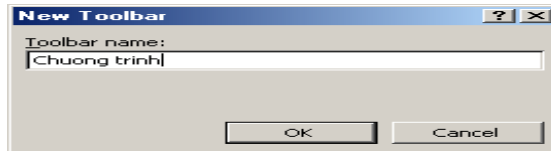
Có 2 cách hiển thị các Menu của người sử dụng (Custom Menu) trong các ứng dụng đó là:

Menu ứng với 1 form xác định nào đó: Menu này chỉ xuất hiện khi truy xuất đến Form này.

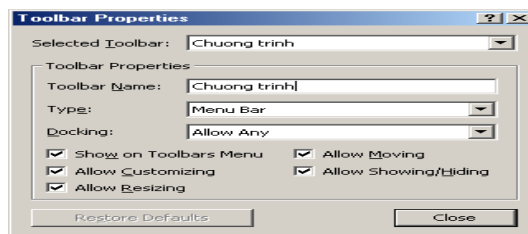
Menu toàn cục (Globally): Menu toàn cục này sẽ xuất hiện trong các ứng dụng và nó chỉ được thay thế khi có 1 menu ứng với form nào đó được mở.

### 5.1. Tạo menu của người sử dụng

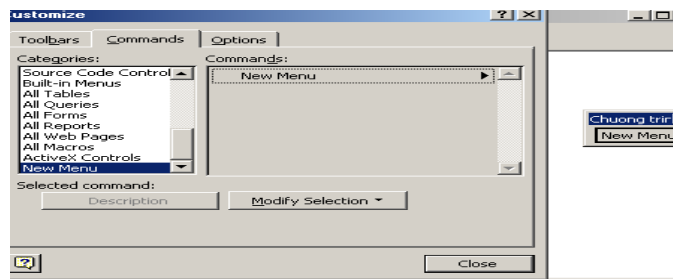
Chọn View/Toolbars/Customize/New/ Đặt tên cho thanh menu



Chọn Properties và chọn Type là Menu bar, close

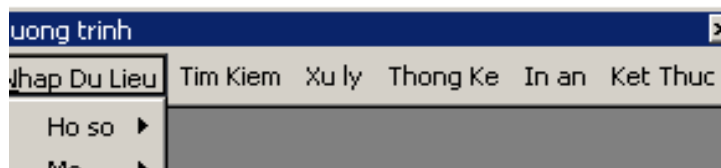


Chọn command, trong mục Categories chọn New menu và kéo sang thanh menu vừa tạo.



Click chuột phải để thay đổi các tiêu đề cho phù hợp trên thanh menu.

Tương tự cho các nhóm khác.



Có thể tạo các chức năng là các hành động như mở bảng, truy vấn, biểu mẫu... bằng cách kéo các biểu tượng này trong mục command vào menu đang tạo.



## 5.2. Tạo menu toàn cục

Nếu muốn tạo một menu toàn cục sẽ thay thế menu có sẵn trong Access khi CSDL này được mở ta thực hiện các bước sau:

Mở hoặc tạo ra một Macro có tên là Autoexec.

Thêm vào hành động: SetValue

Trong mục các tham số (arguments):

Item: Application.menubar

Expression: Tên của Menubar người sử dụng tạo ra và đặt giữa cặp dấu ngoặc kép "".

**BÀI SỐ 1**

Khởi động MSACCESS, tạo một CSDL có tên NHANSU.MDB rồi lần lượt tạo các bảng dữ liệu sau

**Bảng 1: Nhanvien**

STT	Fieldname	Data Type	Fieldsize	Note
1	<b>Manv</b>	Text	4	Khoá chính
2	Holot	Text	20	
3	Ten	Text	10	
4	Ngaysinh	Date/time	8	
5	Gioitinh	Yes/No	1	<b>Yes:Nam, No:Nu</b>
6	Maphong	Text (Lookup)	2	Lấy từ bảng Dsphong
7	MaPXuong	Text (Lookup)	2	Lấy từ bảng dspxuong
8	Diachi	Text	30	
9	Ghichu	Memo		

**Bảng 2: Dsphong**

STT	Fieldname	Data Type	Fieldsize	Note
1	<b>Maphong</b>	Text	2	Khoá chính
2	Tenphong	Text	30	
3	Sodthoai	Text	11	

**Bảng 3: Thunhap\_NV**

STT	Fieldname	Data Type	Fieldsize	Note
1	<b>Manv</b>	Text (Lookup)	4	Khoá chính
2	Luongchinh	Number	Double	
3	Heso	Number	Integer	
4	Phucap	Number	Double	
5	Thue	Number	Double	Thuế
6	Thamnien	Number	Byte	Số năm thâm niên
7	Tongluong	Number	Double	

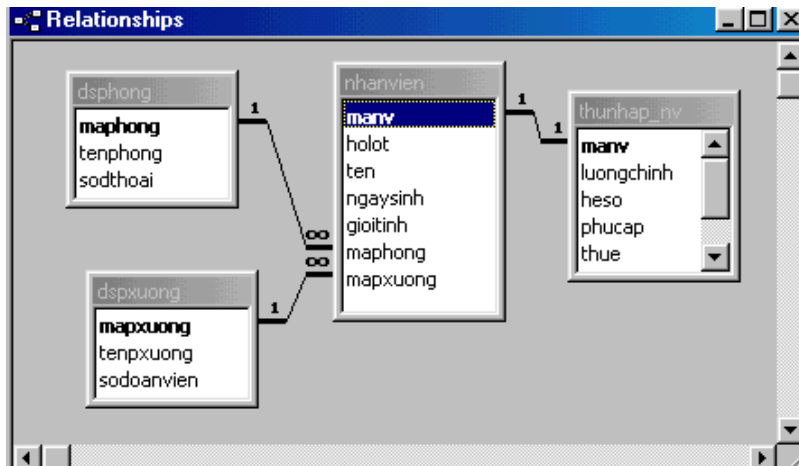
**Bảng 4: DSpxuong**

STT	Fieldname	Data Type	Fieldsize	Note
1	<b>Mapxuong</b>	Text	2	Khoá chính
2	Tenpxuong	Text	20	
3	Sodoanvien	Number	Byte	Số đoàn viên

1. Hãy nhập dữ liệu thích hợp cho các bảng trên.
2. Mở bảng Nhanvien sắp xếp tăng dần theo **manv**, sắp xếp giảm dần theo **ten**.
3. Sắp xếp tăng dần theo **ten**, nếu **ten** trùng nhau thì sắp xếp giảm dần theo **holot** nếu **holot** trùng nhau thì sắp xếp giảm dần theo **maphong**.
4. Lọc và hiển thị danh sách những nhân viên có tên **Thanh**.
5. Lọc và hiển thị những nhân viên có họ **Nguyen** hoặc **Cao** và có tên **anh**.
6. Lọc và hiển thị những nhân viên có **Manv** bắt đầu là **B** và **maphong** là **A1** hoặc **Manv** bắt đầu là **C** và **maphong** là **A2**.
7. Lọc và hiển thị những nhân viên **nam** và có năm sinh **1975**.
8. Lọc và hiển thị những nhân viên **nam** sinh trong thời gian từ **12/12/76** đến **12/12/79**.
9. Lọc và hiển thị những nhân viên **nữ** có **Mapxuong** là **P1** và sinh trong tháng **10** năm **1975** hoặc nhân viên **nam** có năm sinh **1972** đến **1976**.
10. Mở bảng **Thunhap\_nv** để hiển thị những nhân viên có **heso>100** và có **luongchinh<500.000** hoặc có **phucap** từ **100.000** đến **500.000**.

## BÀI SỐ 2

1. Mở CSDL NHANSU.MDB trong bài thực hành số 1 rồi thực hiện các nhiệm vụ sau:
  - ❖ Đặt khoá chính cho trường **Manv** (**nhanvien**), **Maphong**(**dsphong**), **manv**(**thunhap\_nv**) và **mapxuong** (**dspxuong**).
  - ❖ Thiết lập các mối quan hệ giữa các bảng dữ liệu theo sơ đồ sau



❖ Hãy thiết lập các thuộc tính tham chiếu toàn vẹn và các thuộc tính như Format, caption, inputmask..... cho các trường một cách hợp lệ.

Đặt thuộc tính Format của trường **ngaysinh** : dd-mm-yy , Inputmask: 00/00/00

Đặt thuộc tính Inputmask của trường **sodthoai** : (000)000000

2. Tạo một truy vấn **query1** để hiển thị những thông tin sau:

Manv, holot, ten (trong bảng **nhanvien**), Luongchinh, phucap (Trong bảng **Thunhap\_nv**), tenphong, maphong (Trong **dsphong**).

3. Tạo một truy vấn **query 2** để hiển thị những thông tin sau:

Manv, hoten ( Nối holot và ten), maphong, tenphong, mapxuong, tenpxuong, tongthunhap trong đó tongthunhap được tính theo công thức:

$Tongthunhap = luongchinh * heso + phucap - thue$  nếu là nhân viên **nam**.

$Tongthunhap = luongchinh * heso + phucap$  nếu là nhân viên **nữ** .

4. Tạo một truy vấn **query3** để hiển thị những nhân viên có **maphong** là **p1** và **mapxuong** là **x1** sinh trong tháng **7** năm **1976** bao gồm những tin sau:

Hoten, gioitinh, maphong, tenphong, mapxuong, tenpxuong.

5. Tạo một truy vấn **query4** để hiển thị những nhân viên **nữ** và có **maphong** là **p2** hoặc **p3** sinh trong ngày **20** tháng **12** bao gồm những thông tin:

Hoten, maphong, ngaysinh, gioitinh (chú ý giá trị trường gioitinh phải hiển thị nam hoặc nữ).

6. Tạo một truy vấn **query5** để hiển thị những nhân viên **nam** có **tongthunhap**  $\geq 60000$  hoặc thuộc **maphong** là **p2** và không phải họ **Nguyãu**n bao gồm những thông tin: Ten, maphong, tenphong, tongthunhap.
7. Tạo một truy vấn **query6** để hiển thị những nhân viên **nữ** có **maphong** không bắt đầu là **p** hoặc những nhân viên thuộc phân xưởng không có đoàn viên nào bao gồm những thông tin sau: Holot, ten, gioitinh, tenpxuong.
8. Tạo một truy vấn **query7** để hiển thị những nhân viên **nam** tên **Thanh** hoặc **Long** hoặc những nhân viên **nữ** không phải họ **Lê** hoặc **trần** bao gồm những thông tin: Hoten, gioitinh, maphong, tenphong.
9. Tạo một truy vấn **query8** để hiển thị những nhân viên **nam** sinh trong khoảng thời gian từ năm **1973** đến **1980** thuộc phân xưởng có **mapxuong** là **x2** hoặc những nhân viên không có **thuế** sinh trong tháng **4** đến tháng **8** năm **1975** bao gồm những thông tin: Holot, ten, thangsinh, namsinh, mapxuong, tenpxuong.
10. Tạo một truy vấn **query9** để hiển thị những thông tin: Hoten, namsinh, thamnien, luongchinh, trong đó nếu

Luongchinh  $\geq 3000$  và nhân viên **nữ** thì thamnien là **35**.

Luongchinh  $\geq 4000$  và nhân viên **nam** thì thamnien là **30**.

### **BÀI SỐ 3**

Tạo một CSDL có tên QLTV.MDB, rồi lần lượt tạo các bảng dữ liệu sau:

**Bảng 1: Loaisach**

STT	Fieldname	Data Type	Fieldsize	Note
1	<b>Masach</b>	Text	4	Khoá chính
2	Tensach	Text	20	
3	Tentacgia	Text	20	
4	Namxb	Date/time	8	
5	Soluongco	Number	3	Số lượng có
6	Sotrang	Number	5	Số trang sách
7	Manxb	Text	4	Mã nhà xuất bản

**Bảng 1: Docgiamuon**

STT	Fieldname	Data Type	Fieldsize	Note
1	Hoten	Text	30	
2	Quequan	Text	30	
3	<b>Madocgia</b>	Text	4	Khoá chính
4	Masach	Text (Lookup)	4	Lấy dữ liệu từ Loaisach
5	Sluong	Number	1	Số lượng mượn
6	Ngaymuon	Date/time	8	Ngày mượn
7	Ngayhen	Date/time	8	Ngày hẹn trả

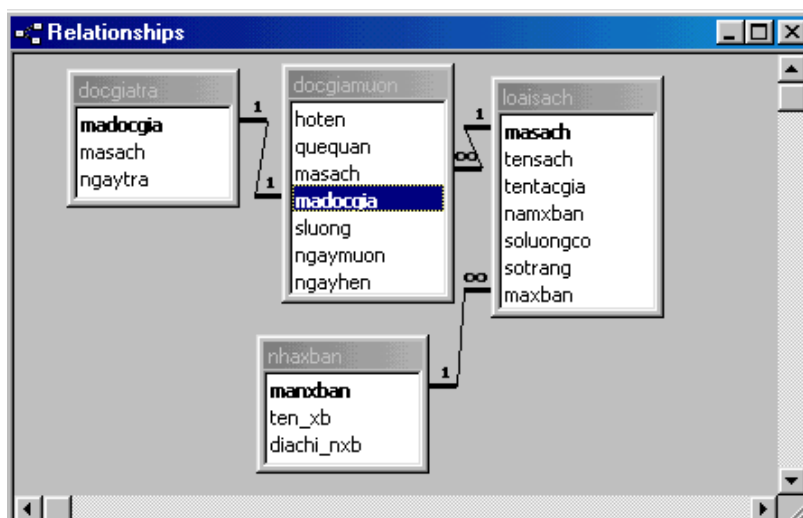
**Bảng 3: Docgiatra**

STT	Fieldname	Data Type	Fieldsize	Note
1	<b>Madocgia</b>	Text	4	Khoá chính
2	Masach	Text	4	
3	Ngaytra	Date/time	8	Ngày trả

**Bảng 4: : Nhaxban**

STT	Fieldname	Data Type	Fieldsize	Note
1	<b>Manxban</b>	Text	4	Khoá chính
2	Ten_xb	Text	30	Tên nhà xuất bản
3	Diachi_nxb	Text	30	Địa chỉ Nhà XB

1. Nhập dữ liệu thích hợp cho các bảng dữ liệu.
2. Đặt khoá chính cho các trường: masach(Loaisach), Madocgia(Docgiamuon), Madocgia(Docgiatra), manxban(Nhaxban).
3. Thiết lập các mối quan hệ giữa các bảng dữ liệu theo sơ đồ sau



4. Tạo một truy vấn **TV1** để hiển thị danh sách độc giả mượn sách trong tháng 10 năm 98 bao gồm những thông tin: Hoten, ngaymuon, tensach, tentacgia.
5. Tạo một truy vấn **TV2** để hiển thị số lượng còn của mỗi loại sách của nhà xuất bản **Kim đồng** bao gồm những thông tin: tensach, nxban, soluongcon.
6. Tạo một truy vấn **TV3** để hiển thị danh sách những độc giả mượn sách quá hạn bao gồm những thông tin: Hoten, Ngaymuon, Ngayhen, songayqua (Số ngày quá hạn).
7. Tạo một truy vấn **TV4** để hiển thị danh sách những độc giả mượn sách quá hạn bao gồm những thông tin Hoten, sluong, mucquahan  
Trong đó: Mucquahan là Mức 1 nếu songayqua<5  
Mucquahan là Mức 2 nếu songayqua<10  
Mucquahan là Mức 3 nếu songayqua>=10
8. Tạo một truy vấn **TV5** để hiển thị danh sách những độc giả trả sách đúng hạn hoặc sớm hạn bao gồm những thông tin: Hoten, masach, tensach.
9. Tạo một truy vấn **TV6** để hiển thị những loại sách có số trang >100 và số lượng còn là 20 của **nhà xuất bản giáo dục** hoặc tên sách có chữ **Tin học** bao gồm Tensach, sotrang, ten\_xb, tentacgia.

## **BÀI SỐ 4**

1. Sử dụng CSDL QLTV.MDB, tạo một truy vấn **Truyvan1** để hiển thị tổng số lượng mượn của từng loại sách trong thư viện.
2. Tạo một truy vấn **Truyvan2** để hiển thị tổng số lượng mượn của từng loại sách trong tháng **12** năm **1998**.
3. Tạo một truy vấn **Truyvan3** để hiển thị tổng số lượng mượn của từng loại sách theo từng tháng trong năm **1999**.
4. Tạo một truy vấn **Truyvan4** để hiển thị tổng số lượng mượn của từng loại sách theo từng tháng của một năm nào đó(Tháng và năm được nhập từ bàn phím).
5. Tạo một truy vấn **Truyvan8** để hiển thị số lượng mượn của từng loại sách trong năm **1999** và có số lượng mượn của mỗi độc giả >2.
6. Tạo một truy vấn **Truyvan9** để hiển thị tổng số loại sách có trong thư viện.

7. Tạo một truy vấn **Truyvan10** để hiển thị tổng số loại sách trong thư viện cho mượn trong năm **1998**.
8. Tạo một truy vấn **Truyvan11** để hiển thị tên của độc giả nào đó mượn sách trong **tháng 1 năm 2001** (Tên được nhập từ bàn phím).
9. Tạo một truy vấn **Truyvan12** để hiển thị tên sách và tên tác giả xuất bản sách trong năm **2002** (Tên sách nhập từ bàn phím, Họ của tác giả nhập từ bàn phím).
10. Tạo một truy vấn tham khảo chéo **Truyvan13** để phản ánh tổng số mỗi loại sách mượn cho mượn trong **tháng 11 năm 2000**.
11. Tạo một truy vấn tham khảo chéo **Truyvan14** để hiển thị tổng số loại sách xuất bản trong năm **1995**.
12. Tạo một truy vấn **Truyvan14** để xoá những sinh viên đã mượn sách quá hạn **5** ngày.

## BÀI SỐ 5

Tạo một CSDL có tên QLSV.MDB, rồi tạo các bảng dữ liệu sau:

**Bảng 1: DSSV**

STT	Fieldname	Data Type	Fieldsize	Note
1	<b>Masv</b>	Text	4	Khoá chính
2	Malop	Text	4	
3	Hotensv	Text	30	
4	Ngaysinh	Date/time	8	
5	Quequan	Text	30	
6	Gioitinh	Yes/No	1	
7	Hocbong	Number	Double	

**Bảng 2: DSDIEM**

STT	Fieldname	Data Type	Fieldsize	Note
1	<b>Masv</b>	Text	4	Khoá chính
2	Mamon	Text	2	Mã môn học
3	Diem_lan1	Number	Double	Điểm thi lần 1
4	Diem_lan2	Number	Double	Điểm thi lần 2



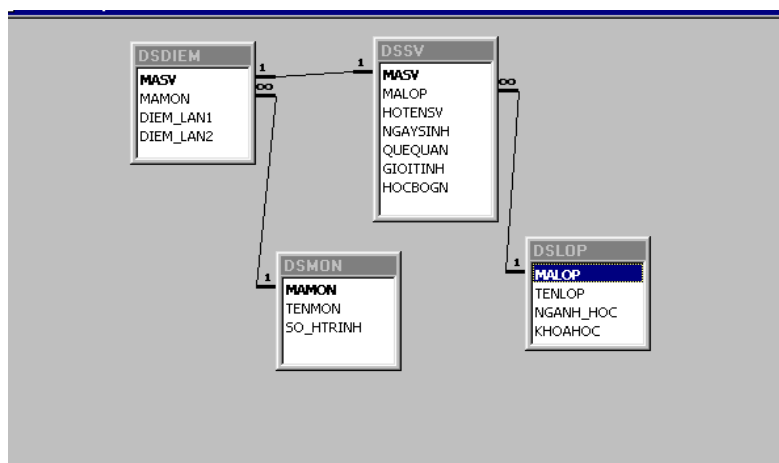
**Bảng 3: DSLOP**

STT	Fieldname	Data Type	Fieldsize	Note
1	<b>Malop</b>	Text	4	Khoá chính
2	Tenlop	Text	20	
3	Nganh_hoc	Text	30	
4	Khoahoc	Text	2	

**Bảng 3: DSMON**

STT	Fieldname	Data Type	Fieldsize	Note
1	<b>Mamon</b>	Text	2	Khoá chính
2	Tenmon	Text	20	
3	So_htrinh	Number	Byte	Số học trình

1. Nhập dữ liệu thích hợp cho các bảng dữ liệu.
2. Đặt khoá chính cho các trường: MASV(DSSV), MASV(DSDIEM), MALOP(DSLOP), Mamon(DSMON).
3. Thiết lập các mối quan hệ giữa các bảng theo sơ đồ sau:



4. Tạo một truy vấn **BT1** để hiển thị tổng số sinh viên của mỗi lớp là bao nhiêu?
5. Tạo một truy vấn tham khảo chéo **BT2** để phản ánh tổng số sinh viên xếp loại **Xsắc**, **Giỏi**, **khá**, **Tb**, **Yếu** của mỗi lớp.
6. Từ bảng dữ liệu DSSV, tạo một truy vấn **BT3** để tạo ra bảng dữ liệu DSLUU lưu trữ những sinh viên có Mã lớp bắt đầu là **T** hoặc **H**.

7. Từ bảng dữ liệu DSSV, tạo một truy vấn **BT4** để tạo ra một bảng dữ liệu DSNAM để lưu trữ những sinh viên **nam** sinh trong năm **1976** bao gồm các thông tin: HOTENV, QUEQUAN, GIOTINH.
8. Từ bảng dữ liệu DSSV, tạo một truy vấn **BT5** để tạo ra một bảng dữ liệu DSNU để lưu trữ những sinh viên **nữ** sinh trong **quý 3** năm **1979** bao gồm các thông tin: HOTENV, QUEQUAN, GIOTINH.
9. Tạo một truy vấn **BT6** để tăng học bổng cho những sinh viên **nữ** thêm **30%**.
10. Tạo một truy vấn **BT7** để nối dữ liệu từ bảng DSNAM vào bảng DSNU.
11. Tạo truy vấn **BT8** để xoá những sinh viên có quê quán ở **Đà Nẵng** và có mã lớp bắt đầu là **B**.

## **BÀI SỐ 6**

1. Sử dụng ngôn ngữ SQL để tạo cấu trúc của các bảng dữ liệu sau:

DSTRUONG ( MATRUONG, TENTRUONG, DIACHI )

DSKHOA ( MATRUONG, MAKHOA, TENKHOA, SODT )

DANHSACH ( MASV, MAKHOA, HOTEN, NGAYSINH, LOP, HOCBONG )

BANGDIEM ( MASV, DTBK1, DTBK2, DTBK3, DTBK4 )

**Trong đó:** Các trường in đậm và gạch chân là khoá chính, kiểu dữ liệu và kích thước của các được mô tả như sau:

### **Bảng DSTRUONG**

*MATRUONG TEXT(2), TENTRUONG TEXT(20), DIACHI TEXT(30)*

### **Bảng DSKHOA**

*MATRUONG TEXT(2), MAKHOA TEXT(4), TENKHOA TEXT(10), SODT TEXT(6).*

Trường SODT được lập chỉ mục.

### **Bảng DANHSACH**

*MASV TEXT(4), MAKHOA TEXT(4), HOTEN TEXT(30), NGAYSINH(DATE/TIME), LOP TEXT(10), HOCBONG (DOUBLE).* Trường MASV được lập chỉ mục.

## **Bảng BANGDIEM**

*MASV TEXT(4), DTBK1 (DOUBLE), DTBK2 (DOUBLE), DTBK3 (DOUBLE), DTBK4 (DOUBLE).*

### **☛ Chú ý**

Sau khi tạo cấu trúc các bảng dữ liệu xong, hãy nhập dữ liệu và thiết lập mối quan hệ giữa các bảng phù hợp. Trường HOCBONG trong bảng DANHSACH chỉ nhập một trong ba giá trị 120000, 180000 hoặc 240000.

### **2. Sử dụng ngôn ngữ SQL để thay đổi cấu trúc của bảng dữ liệu**

- a. Thêm trường GHICHU có kiểu MEMO vào trong bảng DANHSACH
- b. Thêm trường TBCONG có kiểu DOUBLE vào trong bảng BANGDIEM
- c. Thêm trường QUEQUAN có kiểu TEXT và GIOITINH có kiểu YES/NO vào bảng DANHSACH và lập chỉ mục trường QUEQUAN.

### **3. Sử dụng ngôn ngữ SQL để tạo các truy vấn chọn sau**

- a. Chọn MATRUONG, MAKHOA, TENKHOA trong bảng DSKHOA.
- b. Chọn MATRUONG, MAKHOA, SODT trong bảng DSKHOA của những trường có MATRUONG bắt đầu là Q.
- c. Chọn MASV, MAKHOA, HOTEN của những sinh viên sinh trong khoảng thời gian từ 20/10/74 đến 20/10/76 trong bảng DANHSACH.
- d. Chọn MASV, HOTEN, LOP, HOCBONG của những sinh viên có MASV bắt đầu là T và thuộc lớp Tin học hoặc Hoá học hoặc kinh tế trong bảng DANHSACH ( HOTEN đổi thành Họ và tên).
- e. Chọn những sinh viên có tên THANH sinh trong tháng 10/76 hoặc có HOCBONG trong khoảng từ 150000 đến 200000.

### **4. Sử dụng ngôn ngữ SQL để tạo các truy vấn tính tổng sau:**

- a. Tạo một truy vấn để tính tổng HOCBONG của mỗi khoa.
- b. Tạo một truy vấn thống kê xem mỗi khoa số lượng sinh viên là bao nhiêu?
- c. Tạo một truy vấn để tính tổng HOCBONG của mỗi lớp trong mỗi khoa.
- d. Tạo một truy vấn thống kê xem mỗi trường có bao nhiêu khoa?

5. Sử dụng ngôn ngữ SQL để tạo các truy vấn tham số sau:

- a. Tạo một truy vấn tham số thống kê xem mỗi mức HOCBONG mỗi khoa có bao nhiêu sinh viên.
- b. Tạo một truy vấn tham số để thống kê xem số lượng sinh viên sinh trong mỗi tháng của năm 1979 của mỗi khoa là bao nhiêu?

6. Sử dụng ngôn ngữ SQL để tạo các truy vấn tạo bảng sau:

- a. Tạo một bảng DIEMLUU gồm tất cả các trường trong bảng DIEMTHI.
- b. Tạo một bảng DSLUU gồm các trường MASV, MAKHOA, HOTEN từ bảng DANHSACH của những sinh viên sinh trước ngày 20/11/76.
- c. Tạo một bảng DSLUU1 gồm các trường HOTEN, LOP của những sinh viên thuộc lớp **Tin K25A** và sinh năm 1985 hoặc trước năm 1978.

7. Sử dụng ngôn ngữ SQL để tạo các truy vấn nối dữ liệu sau:

- a. Tạo một truy vấn nối dữ liệu từ bảng DIEMLUU vào DIEMTHI nhưng chỉ gồm các trường DTBK1, DTBK2.
- b. Tạo một truy vấn nối dữ liệu từ bảng DIEMLUU vào DIEMTHI nhưng chỉ gồm các trường DTBK1, DTBK2, DTBK3 đối với những bản ghi ghi có  $DTBK3 \geq 8$ .

8. Sử dụng ngôn ngữ SQL để tạo các truy vấn cập nhật dữ liệu sau:

- a. Tính giá trị trường DTBCONG của bảng DIEMTHI theo công thức  $(DTBK1 + DTBK2 + DTBK3 + DTBK4)/4$ .
- b. Tăng HOCBONG thêm 100.000 cho những sinh viên có MASV bắt đầu là A trong bảng DANHSACH.

c. Giảm HOCBONG đi 50.000 cho những sinh viên có năm sinh từ 1975 đến 1978 hoặc những sinh viên có tên NHAN.

**9.** Sử dụng ngôn ngữ SQL để tạo các truy vấn xoá sau:

a. Tạo một truy vấn xoá những sinh viên thuộc khoa toán.

b. Tạo một truy vấn xoá những sinh viên sinh trong khoảng thời gian từ 20/10/74 đến 20/10/76 hoặc có HOCBONG=120.000.

c. Tạo một truy vấn xoá những sinh viên có họ NGUYEN hoặc tên THANH sinh trong tháng 7 năm 1978.

**10.** Sử dụng ngôn ngữ SQL để tạo các truy vấn dựa trên nhiều bảng sau:

a. Tạo truy vấn để hiển thị HOTEN, TENKHOA, LOP, HOCBONG từ 2 bảng dữ liệu DSKHOA và DANHSACH.

b. Tạo một truy vấn để hiển thị TENTRUONG, TENKHOA, HOTEN của những sinh viên sinh trong năm 1980 từ 3 bảng DSTRUONG, DSKHOA, DANHSACH.

c. Tạo một truy vấn để hiển thị TENTRUONG, TENKHOA, HOTEN, DTBK1, DTBK2 từ 4 bảng dữ liệu DSTRUONG, DSKHOA, DANHSACH, DIEMTHI.

**11.** Sử dụng ngôn ngữ SQL để tạo các truy vấn con sau:

a. Tạo một truy vấn để hiển thị HOTEN, NGAYSINH, LOP của những sinh viên có DTBK4 $\geq$ 5.

b. Tạo một truy vấn để hiển thị ít nhất một sinh viên có DTBK2 $\leq$ 4.

**12.** Sử dụng ngôn ngữ SQL để tạo các truy vấn hội sau:

a. Tạo truy vấn hội để hiển thị HOTEN, DTBK1, DTBK2 từ 2 bảng dữ liệu DIEMTHI và DIEMLUU.

b. Tạo truy vấn hội để hiển thị HOTEN, DTBK1, DTBK2, DTBK3 từ 2 bảng dữ liệu DIEMTHI và DIEMLUU nhưng chỉ hiển thị những sinh viên có DTBK3 $\geq$ 7.

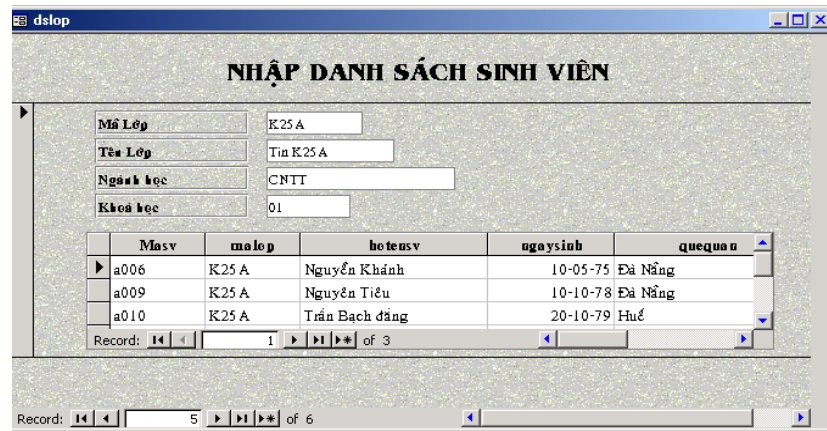
## BÀI SỐ 7

Sử dụng CSDL **QLSV.MDB** trong **bài tập số 5** để thực hiện các yêu cầu sau:

1. Tạo một form có tên **Nhaplop** để nhập dữ liệu cho bảng **dslop** như sau:



2. Tạo một form chính phụ như sau:



3. Tạo một form có tên **Hienthi** để hiển thị danh sách sinh viên như sau :



Mã sv	Mã lớp	Họ và tên	Ngày sinh	Quê quán	Giới tính	Học bổng
a001	K23A	Nguyễn Tèo	10-10-78	Huế	<input checked="" type="checkbox"/>	1000000
a002	K23B	Trần Bình Chi	05-05-79	Đà Nẵng	<input type="checkbox"/>	2000000
a003	K24A	Nguyễn Thanh Thủy	12-10-79	Huế	<input checked="" type="checkbox"/>	2300000
a004	K24B	Võ Tông	02-02-79	Quảng Bình	<input type="checkbox"/>	2200000
a005	K24A	Lê Mậu Ngo	08-08-79	Huế	<input checked="" type="checkbox"/>	1200000
a006	K25A	Nguyễn Khánh	10-05-75	Đà Nẵng	<input checked="" type="checkbox"/>	3000000
a007	K25B	Lê Thủy Thu	03-03-79	Quảng Nam	<input type="checkbox"/>	2000000
a008	K25B	Trần Thanh	25-07-77	Huế	<input checked="" type="checkbox"/>	3000000
a009	K25A	Nguyễn Tiêu	10-10-78	Đà Nẵng	<input type="checkbox"/>	1000000
a010	K25A	Trần Bạch đăng	20-10-79	Huế	<input checked="" type="checkbox"/>	2000000

4. Hãy thiết kế form theo yêu cầu sau:



Hãy gán các chức năng phù hợp với các mục chọn.

5. Hãy thiết kế một **Report** theo yêu cầu sau:

## BÀI SỐ 8

Cho CSDL **QLSVIEN.MBD** gồm các bảng dữ liệu sau:

**DSSVIEN**(MASV, MALOP, HOTENSVIEN, NGAYSINH, GTINH, QQUAN)

**DSMONHOC**(MAMON, TENMON, SOH\_TRINH)

**DSDIEMTHI**(MASV, MAMON, DIEM\_LAN1, DIEM\_LAN2)

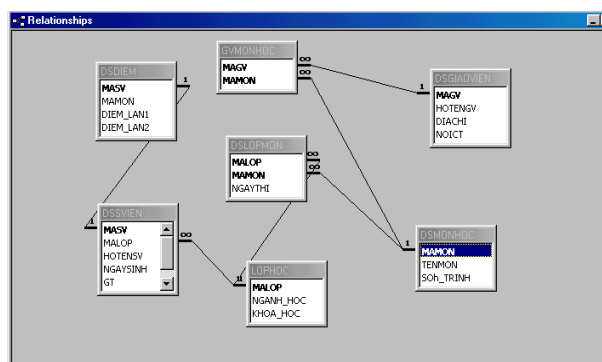
*DSGIAOVIEN*(MAGV, HOTENGVIEI, DIACHI, NOI\_CT)

*DSLOPHOC*(MALOP, NGANH\_HOC, KHOA\_HOC)

*GVMONHOC*(MAGV, MAMON)

*DSLOPMON*(MALOP, MAMON, NGAYTHI)

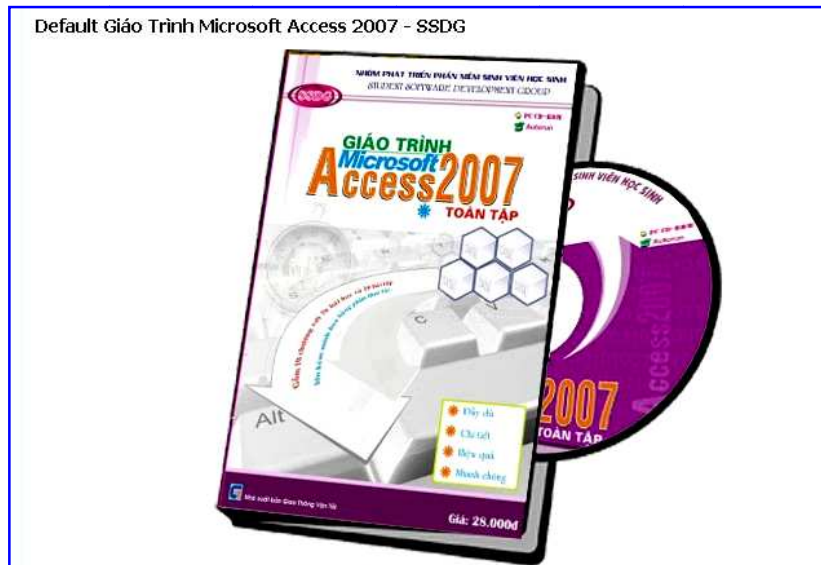
Mối quan hệ giữa các bảng dữ liệu theo sơ đồ sau:



1. Tạo các biểu mẫu để cập nhật dữ liệu cho các bảng trên, chú ý sử dụng combo box hoặc List box đối với các trường cần thiết.
2. Tạo báo cáo để hiển thị danh sách sinh viên trong một lớp học nào đó, bao gồm cả ngành học tương ứng.
3. Tạo báo cáo để hiển thị danh sách sinh viên trong từng lớp học, bao gồm cả ngành học, trong đó các sinh viên cùng một lớp thì tên lớp được nhóm lại với nhau.
4. Tạo báo cáo để hiển thị danh sách giáo viên đã giảng dạy các môn học cho một lớp nào đó.
5. Tạo báo cáo để hiển thị danh sách giáo viên đã tham gia giảng dạy các môn học cho nhiều lớp học, trong đó các giáo viên giảng dạy các môn học cho một lớp học được nhóm lại với nhau.
6. Tạo báo cáo để hiển thị điểm thi các môn học của một sinh viên.
7. Tạo báo cáo để hiển thị điểm thi một môn học của một lớp nào đó.



# Giáo Trình Microsoft Access 2007



**Chương 1: Tổng quan về Microsoft Access**

**Chương 2: Bảng dữ liệu - Table**

**Chương 3: Truy vấn dữ liệu - Query**

**Chương 4: Biểu mẫu- Form**

**Chương 5: Báo cáo - Report**

**Chương 6: Tập lệnh - Macro**

**Chương 7: Tùy biến biểu mẫu hoặc báo cáo**

**Chương 8: Ngôn ngữ Visual Basic for Applications**

**Chương 9: Môi trường ứng dụng nhiều người dùng**

**Chương 10: Bài tập tổng hợp Access**

**Gồm 18 bài tập với hơn 100 câu hỏi:**

- Phan 1: Bài tập bảng
- Phan 2: Bài tập truy vấn
- Phan 3: Bài tập về biểu mẫu
- Phan 4: Bài tập về báo cáo

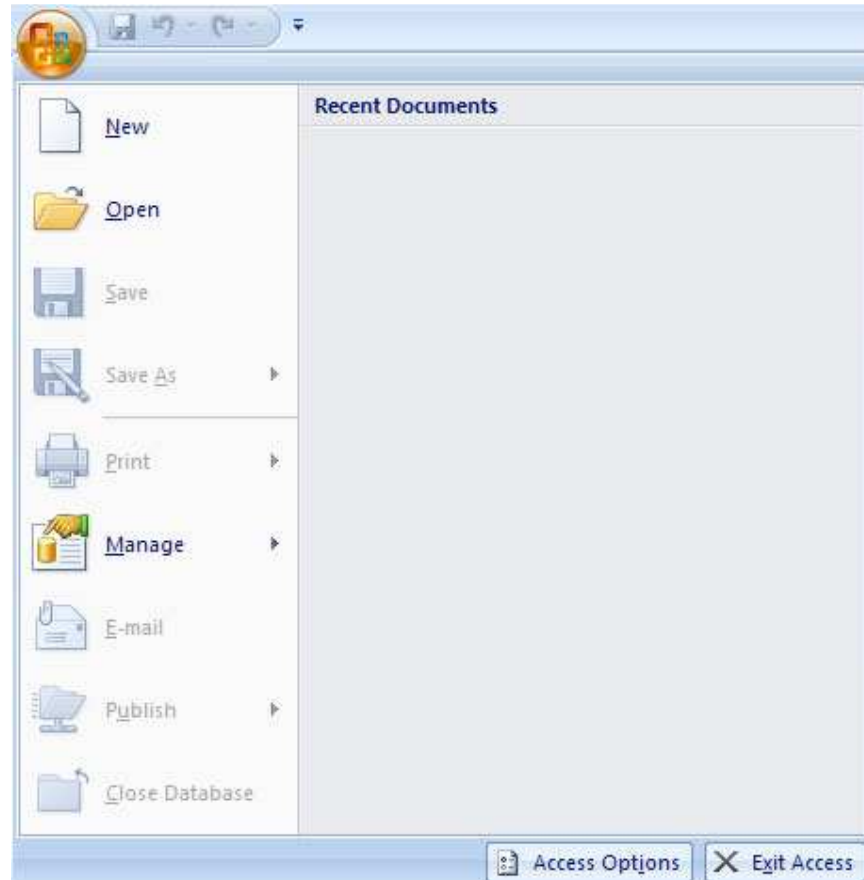
**Phần thư giãn: Với trò chơi ghép hình, tìm cặp, tinh mắt.**

**Mỗi chương đều kèm Bài kiểm tra kiến thức !**

# Giáo Trình Microsoft Access 2007

# MS Access - Bài 1: Bắt đầu với Microsoft Access 2007

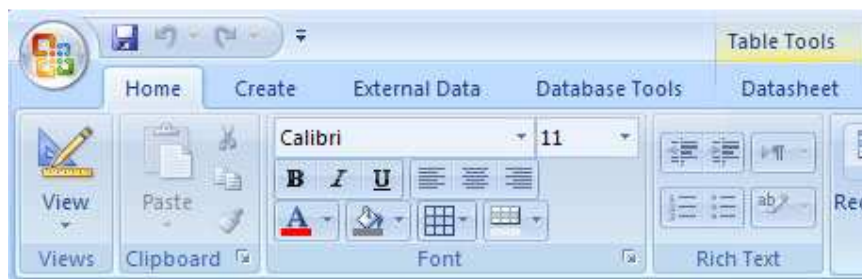
Microsoft Office Button (biểu tượng hình tròn màu vàng ở phía góc trên bên trái) thực thi rất nhiều chức năng mà được đặt ở thực đơn File trong các phiên bản Access cũ. Nút này cho phép bạn tạo một cơ sở dữ liệu mới (*New*), mở một cơ sở dữ liệu đã tồn tại (*Open*), lưu (*Save*) và lưu với tên (*Save as*), in ấn (*Print*), *Send* hay *Close*.



Hình 1: Microsoft Office Button

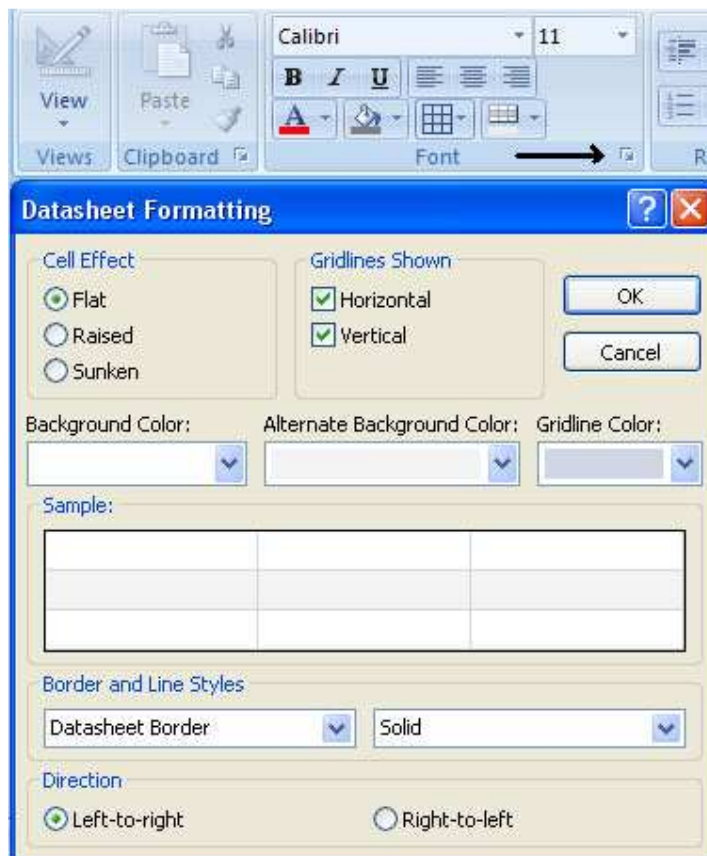
## Ribbon

Ribbon là vùng phía trên của tài liệu. Nó có 5 tab: *Home*, *Create*, *External Data*, *Database Tools*, và *Datasheet*. Mỗi tab được tách thành các nhóm. Các nhóm tập hợp các tính năng thiết kế để thực thi các chức năng mà bạn sẽ sử dụng trong việc phát triển hay chỉnh sửa cơ sở dữ liệu Access.



Hình 2: Các tab trong vùng Ribbon

Các tính năng phổ biến được hiển thị trên vùng Ribbon. Để thấy các tính năng thêm trong mỗi nhóm, kích vào mũi tên phía góc dưới bên phải mỗi nhóm.



Hình 3: Ví dụ tính năng trong nhóm Font

**Home:** Views, Clipboard, Fonts, Rich Text, Records, Sort & Filter, Find.

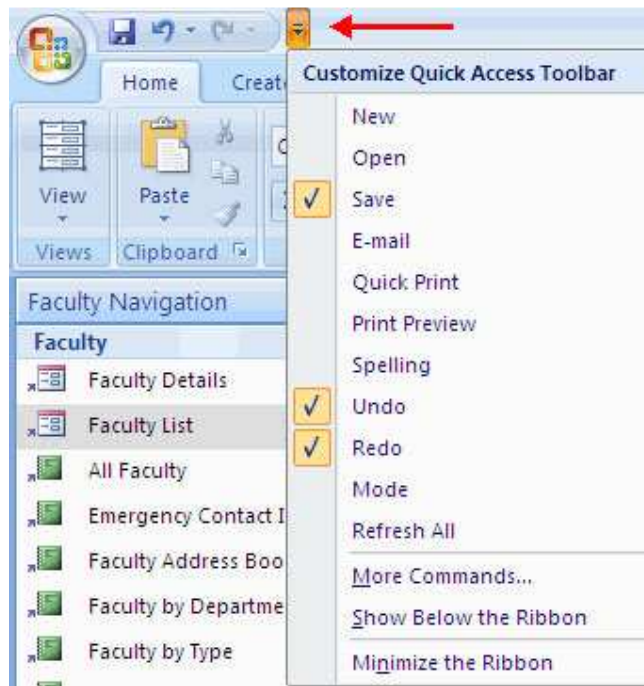
**Create:** Tables, Forms, Reports, Other.

**External Data:** Import, Export, Collect Data, SharePoint Lists.

**Database Tools:** Macro, Show/Hide, Analyze, Move Data, Database Tools.

### Quick Access Toolbar

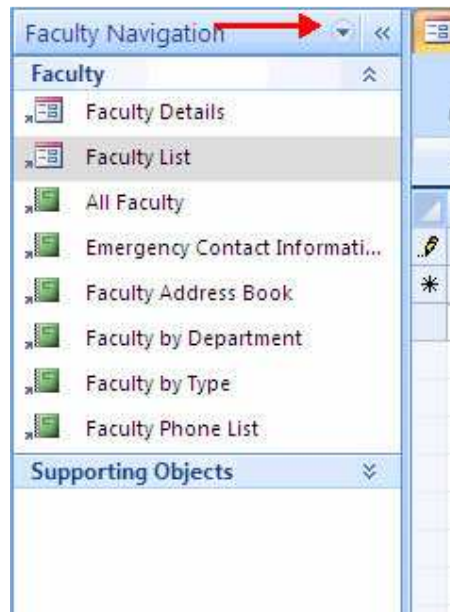
Quick Access Toolbar là thanh công cụ tùy chỉnh chứa các lệnh mà bạn muốn để sử dụng. Bạn có thể đặt Quick access toolbar phía trên hay dưới vùng ribbon. Để thay đổi vị trí của nó, kích vào mũi tên cuối dưới thanh công cụ và chọn **Show Below the Ribbon**.



Hình 4: Quick Access Toolbar

### Vùng Navigation

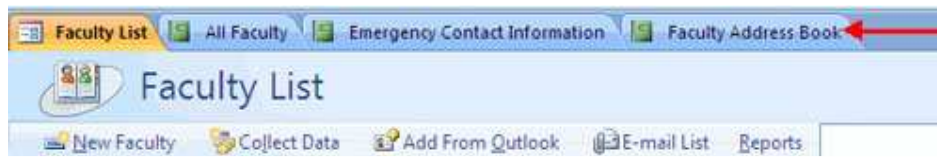
Navigation hiển thị các đối tượng cơ sở dữ liệu như các bảng, biểu mẫu, truy vấn và reports.



Hình 5: Navigation Pane

### Tabbed Document Window Viewing

Tabbed document viewing mở các bảng, biểu mẫu, truy vấn và reports trong cùng cửa sổ. Kích vào các tab để chuyển đổi giữa các cửa sổ.

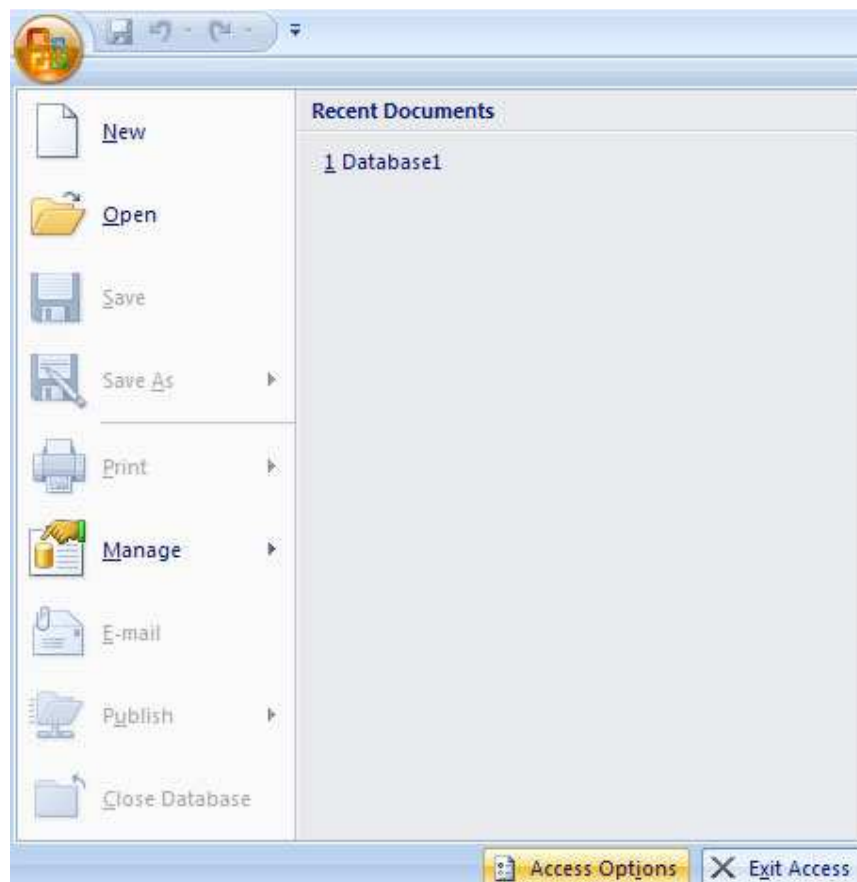


Hình 6: Tabbed Document Window Viewing

## Bài 2: Customize Access

**Access 2007 cung cấp một vùng rộng các tùy chọn có thể tùy chỉnh mà cho phép bạn làm việc với Access tốt hơn.** Để truy cập những tùy chọn này:

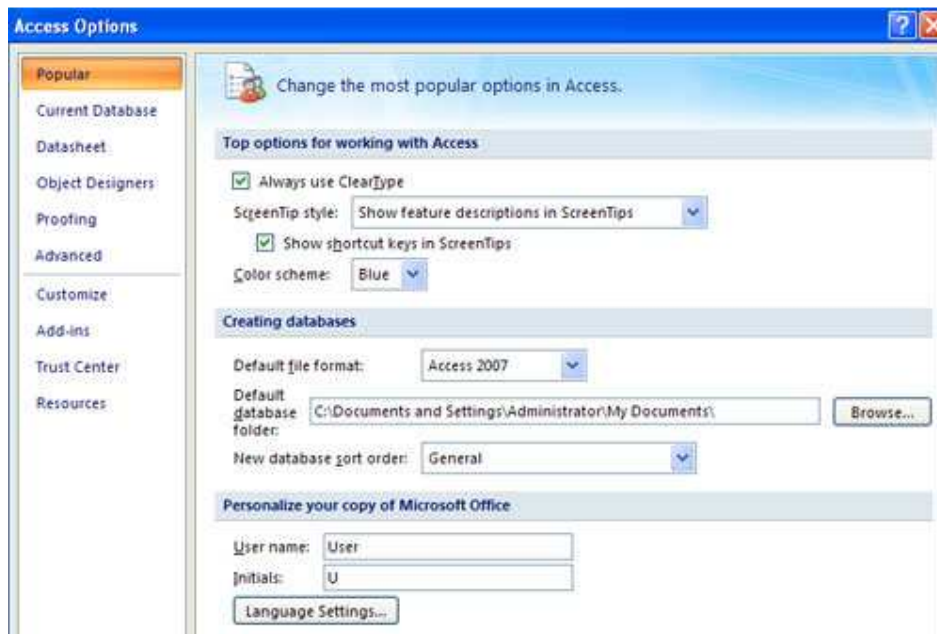
- Kích Office Button
- Chọn Access Options



Hình 1: Chọn Access Options

### Popular

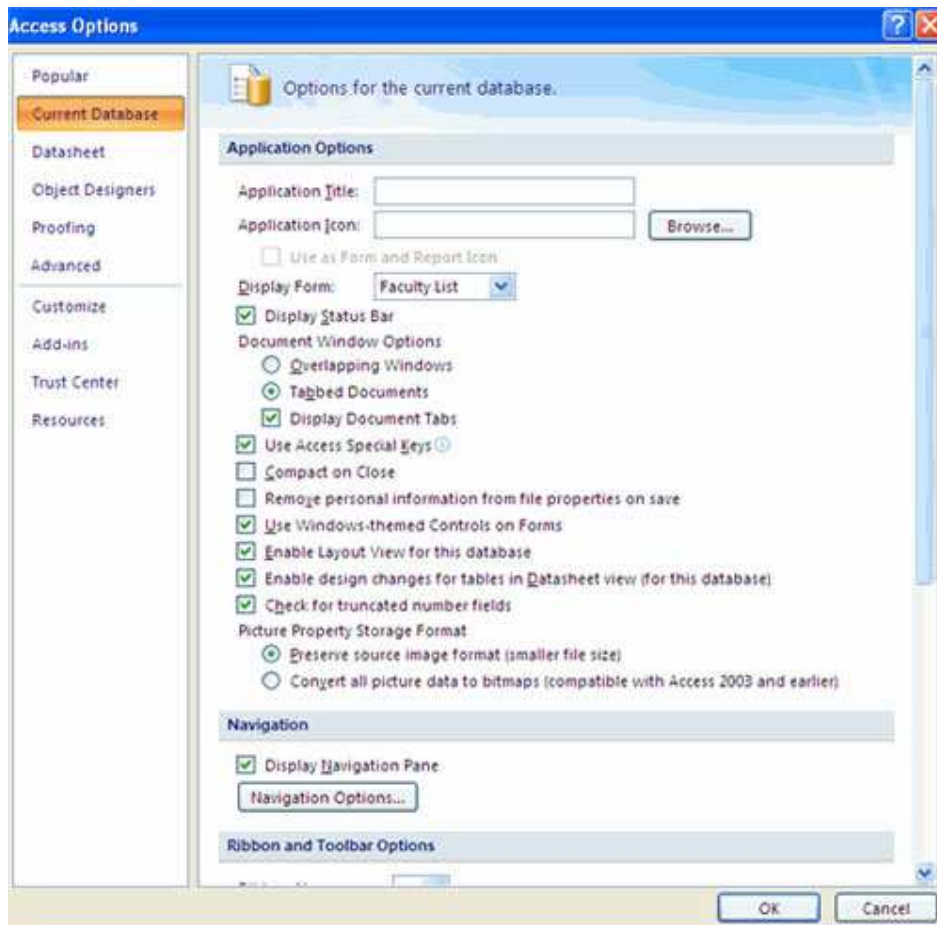
Những tính năng này cho phép bạn xác định môi trường làm việc với việc sử dụng ScreenTips, vị trí và định dạng file cho cơ sở dữ liệu, username.



Hình 2: Thiết lập tính năng Popular

## Current Database

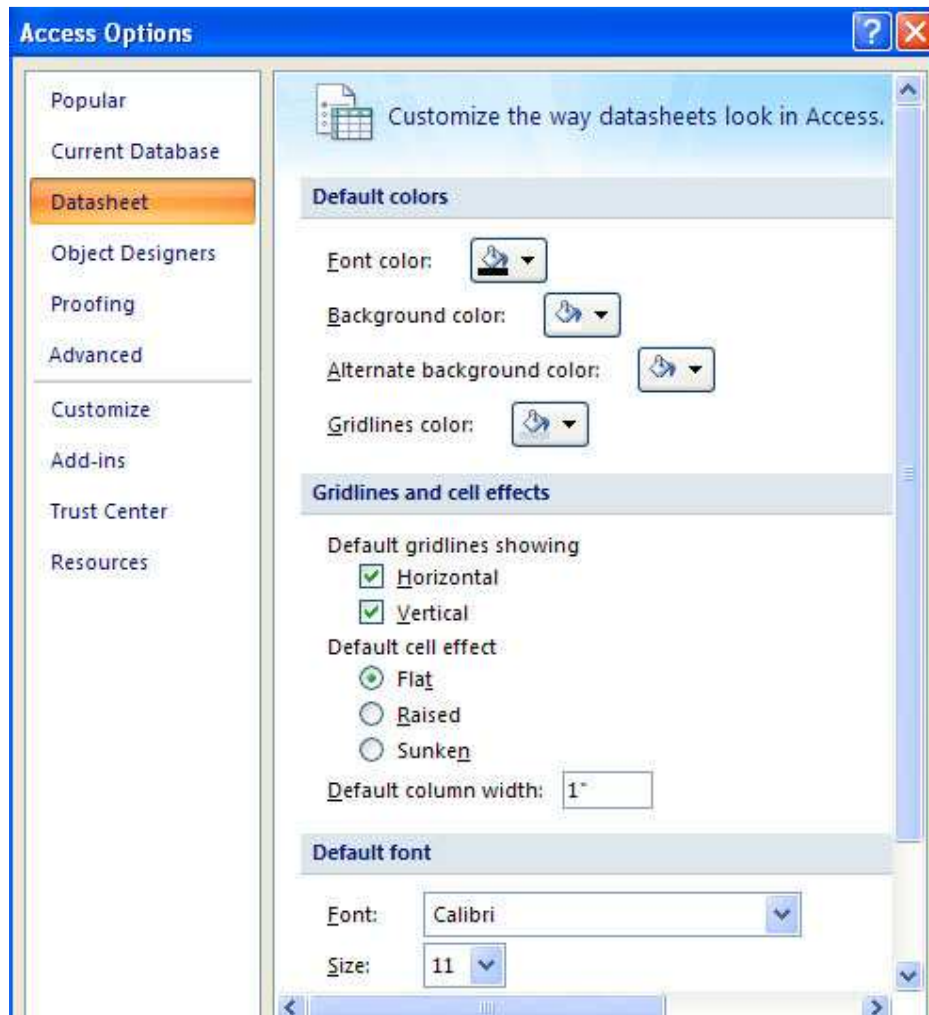
Tính năng này cho phép bạn thiết lập các tùy chọn cho Application, Navigation, Ribbon và Toolbars, AutoCorrect, và Filters.



Hình 3: Thiết lập tính năng Current Database

## Datasheet

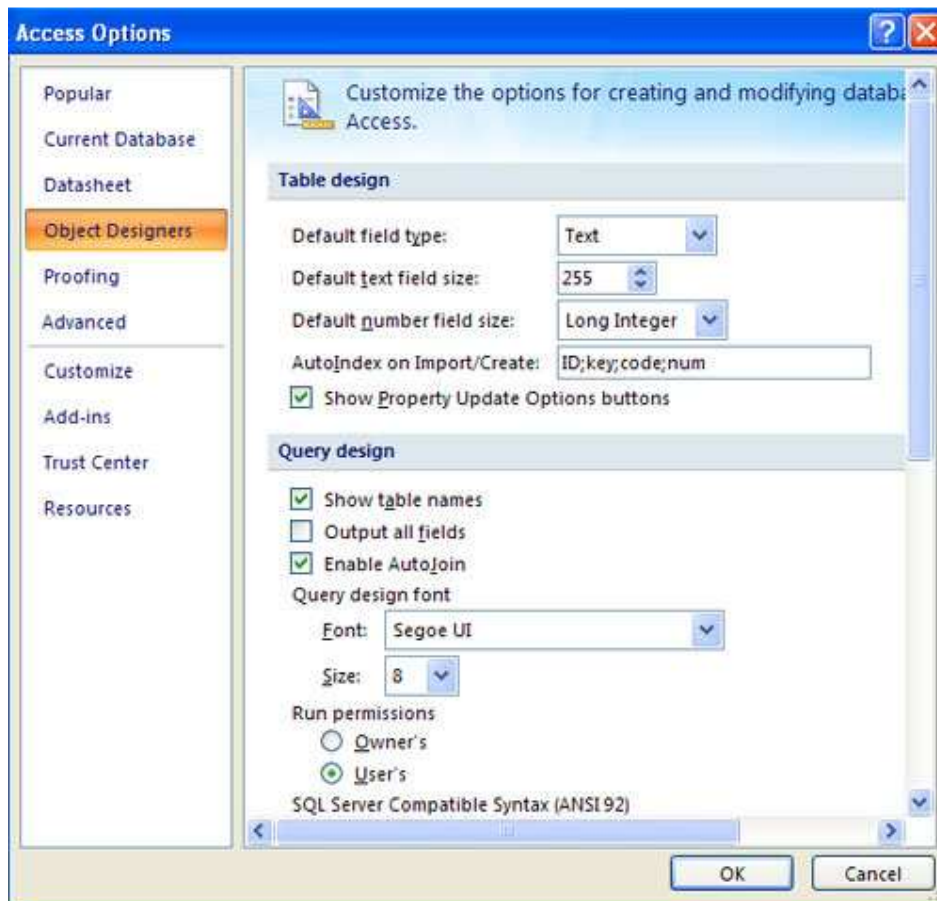
Tính năng này cho phép bạn xác định các tùy chọn riêng trong chế độ datasheet, bao gồm Default colors, Gridlines and cell effects và Default font.



Hình 4: Thiết lập tính năng Datasheet

## Object Designers

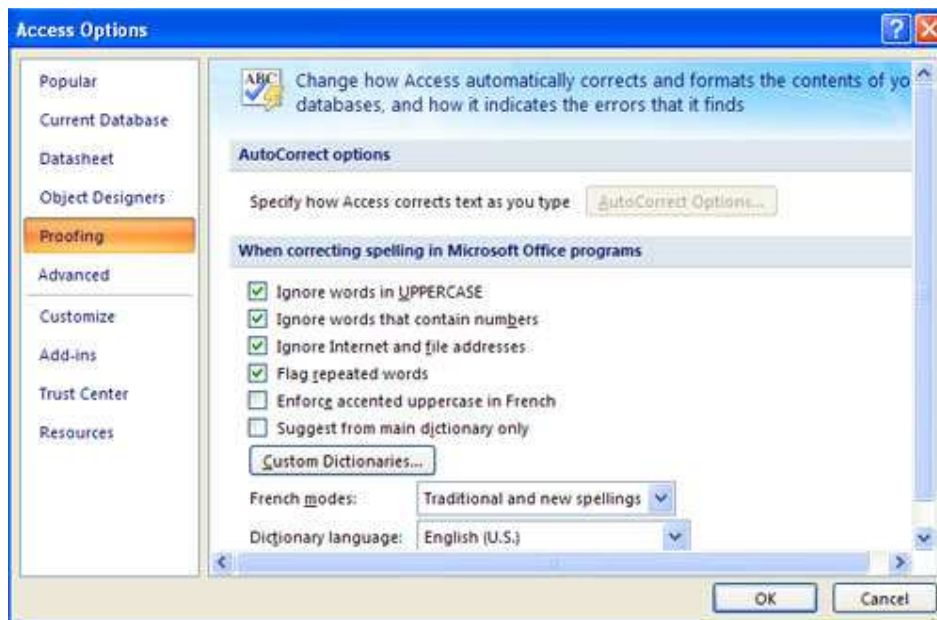
Tính năng này cho phép bạn tùy chỉnh các tùy chọn cho việc tạo và chỉnh sửa các đối tượng cơ sở dữ liệu trong Access bao gồm Table Design, Query Design, Forms/Reports, và Error Checking.



Hình 5: Thiết lập tính năng Object Designers

## Proofing

Tính năng này cho phép bạn thay đổi cách Access tự động chỉnh và định dạng nội dung của cơ sở dữ liệu và cách nó phát hiện lỗi.

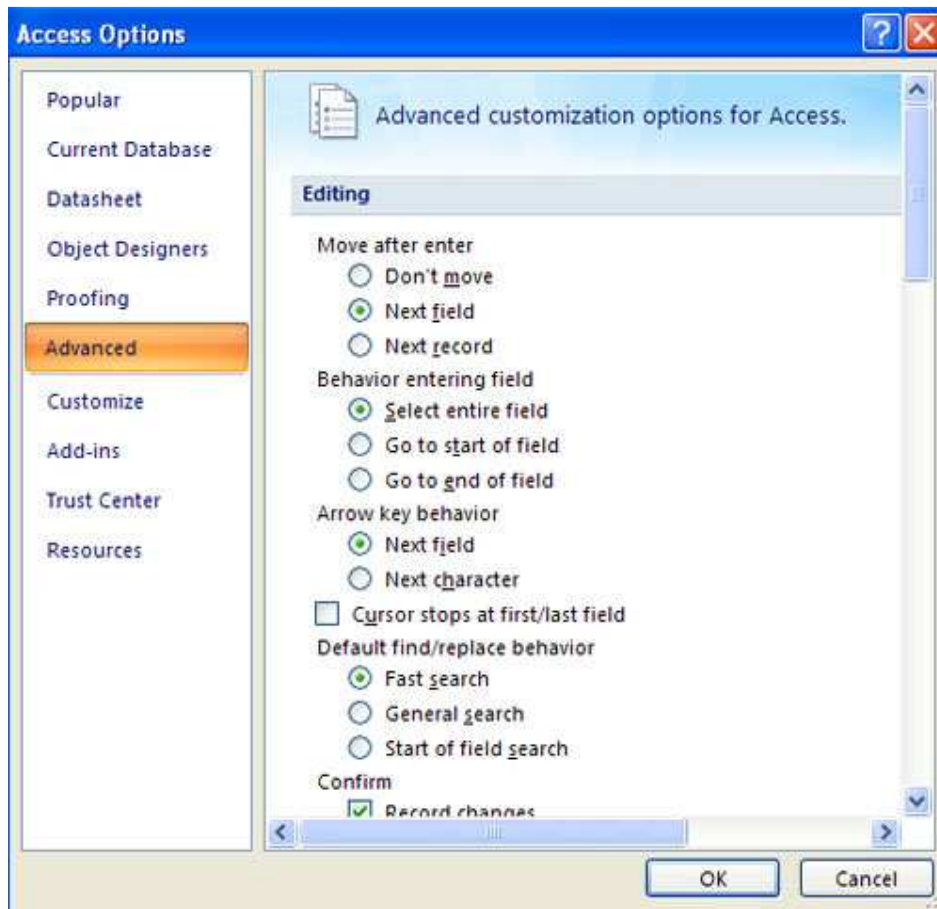


Hình 6: Thiết lập tính năng Proofing



## Advanced

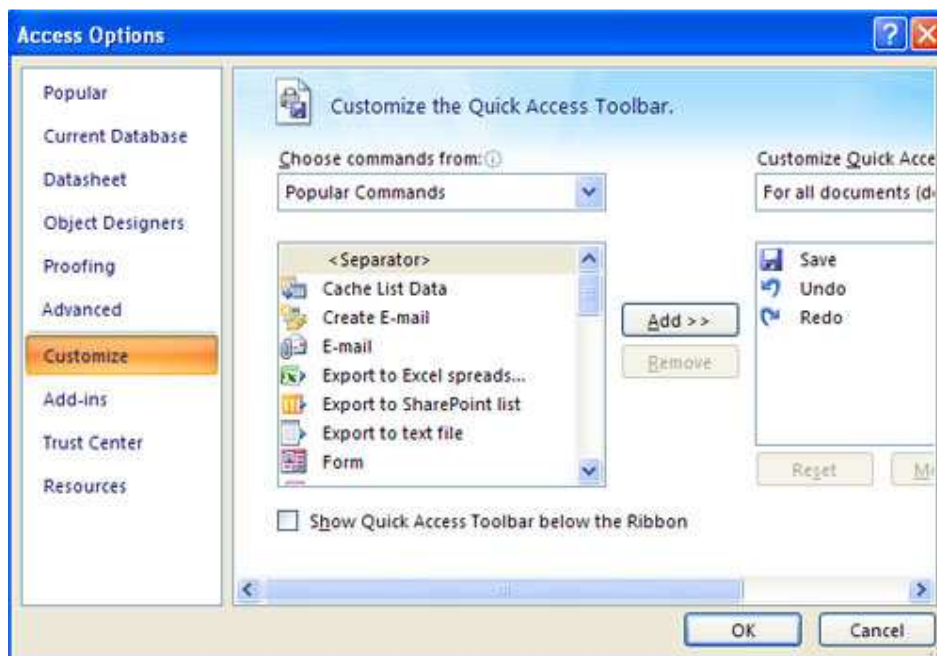
Tính năng này cho phép bạn tùy chỉnh ở mức độ cao cấp trong Access, bao gồm Editing, Display, Printing và các tùy chọn cao cấp khác



Hình 7: Thiết lập tính năng Advanced

## Customize

Customize cho phép bạn thêm các tính năng vào Quick Access Toolbar. Nếu có công cụ cho phép bạn sử dụng thường xuyên, bạn có thể muốn thêm chúng vào Quick Access Toolbar.



Hình 8: Thiết lập tính năng Customize

## Bài 3: Các đối tượng trong cơ sở dữ liệu

Các đối tượng trong cơ sở dữ liệu bao gồm bảng (tables), truy vấn (queries), báo cáo (report), biểu mẫu (forms), ...

### Bảng (Table)

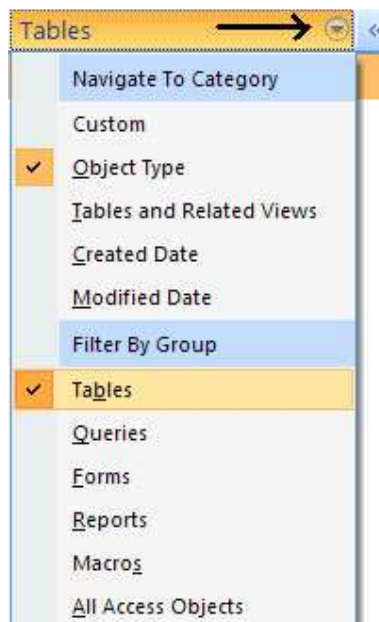
Một bảng là tập hợp các thông tin được đặt trong các dòng và cột. Thông tin về một mục được hiển thị trong một dòng. Cột chứa thông tin cùng loại với mỗi mục. Bảng có dòng đầu cho bạn biết dữ liệu được chứa trong cột là gì.

Makhach	Tenkach	Diachi	Dienthoai	Add New Field
K1	Đại lý Hoàng Minh	88 Thái Hà, Hà Nội	048867567	
K2	Siêu thị Lê Lợi	32 Hàng Bông, Hà Nội	047865432	
K3	Đỗ Ngân Hà	12 Hàng Bài, Hà Nội	046677882	
K4	Cửa hàng Toàn Thắng	76 Cửa Bắc, Hà Nội	045635465	
K5	Đỗ Sơn hai	66 Trần Phú, Hà Nội	048367978	

Hình 1: Ví dụ về thông tin bảng

Để xem dữ liệu trong một bảng:

- Kích vào mũi tên để mở **Navigation pane**.
- Chọn đối tượng **Tables**.
- Sau đó, kích đúp vào tên bảng để mở bảng.



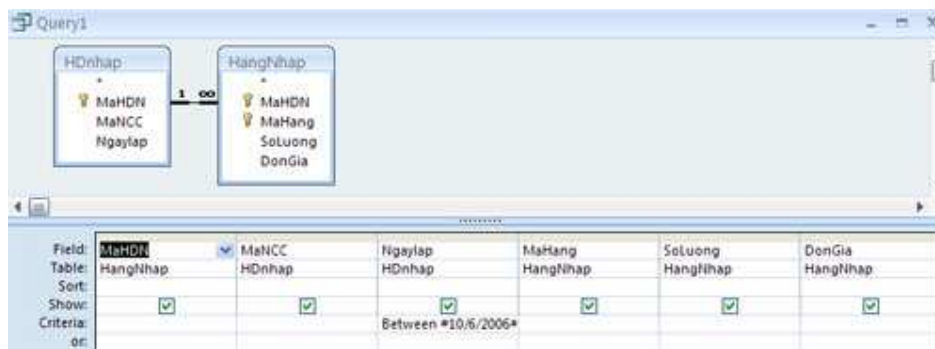
Hình 2: Xem dữ liệu trong bảng

## Truy vấn (Queries)

Các truy vấn lựa chọn các bản ghi từ một hay nhiều bảng trong cơ sở dữ liệu, do vậy mà chúng có thể được xem, phân tích và lưu trữ ở chế độ datasheet. Một truy vấn có thể thực thi các phép toán và hiển thị kết quả. Tập hợp các kết quả của các bản ghi được gọi là **dynaset**, được lưu như một đối tượng cơ sở dữ liệu và có thể sử dụng dễ dàng. Truy vấn sẽ được cập nhật bất cứ khi nào bảng gốc được cập nhật. Nếu kiểu truy vấn là các truy vấn chọn lọc (**select queries**) thì trích dữ liệu từ các bảng dựa trên các giá trị cụ thể, kiểu truy vấn là **find duplicate** thì hiển thị các bản ghi với việc sao chép các giá trị đối với một hoặc nhiều trường cụ thể, và kiểu truy vấn **find unmatched** hiển thị các bản ghi từ một bảng mà không có các giá trị tương ứng với bảng thứ hai.

Để thực thi một truy vấn:

- Kích vào mũi tên để mở Navigation pane.
- Chọn đối tượng **Queries**.
- Kích đúp vào tên truy vấn.



Hình 3: Ví dụ về thiết kế truy vấn

## Recordset

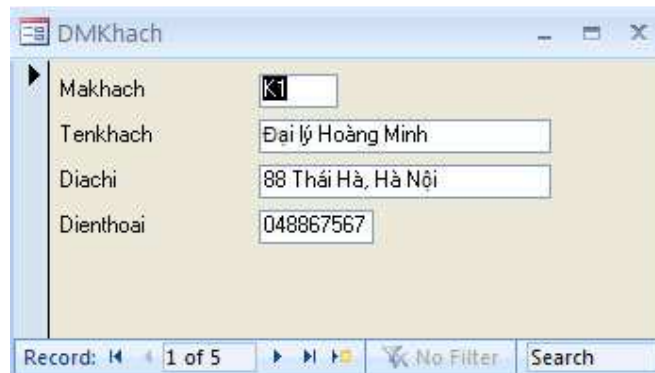
Recordset là một bảng mà hiển thị nhóm các bản ghi từ một bảng hay một kết quả truy vấn.

## Form

Một Form là một giao diện đồ họa được sử dụng để hiển thị và chỉnh sửa dữ liệu. Forms có thể được phát triển từ một bảng hay một truy vấn. Forms gồm các phép toán, đồ họa và các đối tượng như label, textbox, combobox, checkbox, ...

Để xem dữ liệu từ một form:

- Kích mũi tên để mở Navigation pane.
- Chọn đối tượng **Forms**.
- Kích đúp vào tên form.



Hình 4: Thông tin trên form Khách hàng

## Báo cáo (Report)

Một báo cáo là một đầu ra của dữ liệu theo yêu cầu. Các báo cáo có thể thực thi các phép toán và hiển thị kết quả. Các báo cáo được sử dụng để in dữ liệu.

Để xem dữ liệu sử dụng một báo cáo:

- Kích mũi tên để mở Navigation pane.
- Chọn đối tượng **Reports**.
- Kích đúp vào tên báo cáo.

**THỐNG KÊ KHÁCH HÀNG - TÊN HÀNG**

Tenkhach      Đại lý Hoàng Minh

STT	Tenhang:	SoLuong:	Donvi:	Tien:
1	Bias Ha Ni	4	thong	584000

Tổng tiền:                      584000

Hình 5: Ví dụ về báo cáo

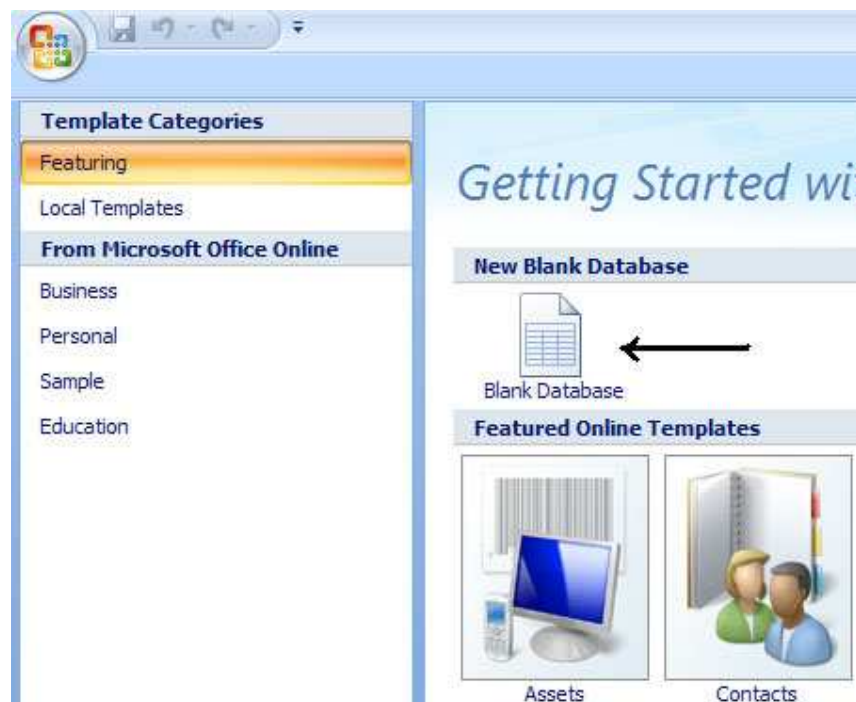
# Bài 4: Tạo cơ sở dữ liệu mới

Bạn có thể tạo một cơ sở dữ liệu mới từ đầu hoặc từ cơ sở dữ liệu wizard.

## Tạo cơ sở dữ liệu mới

Để tạo một cơ sở dữ liệu mới từ đầu:

- Kích chọn **Microsoft Office Button**.
- Chọn **New**.
- Chọn biểu tượng **New Blank Database**.



Hình 1: Tạo cơ sở dữ liệu từ New Blank Database

- Nhập tên cho cơ sở dữ liệu vào phần vùng bên phải ngay sau khi chọn New Blank Database.
- Kích **Create** để tạo.



Hình 2: Bước hoàn thành tạo Database mới

### Các mẫu cơ sở dữ liệu

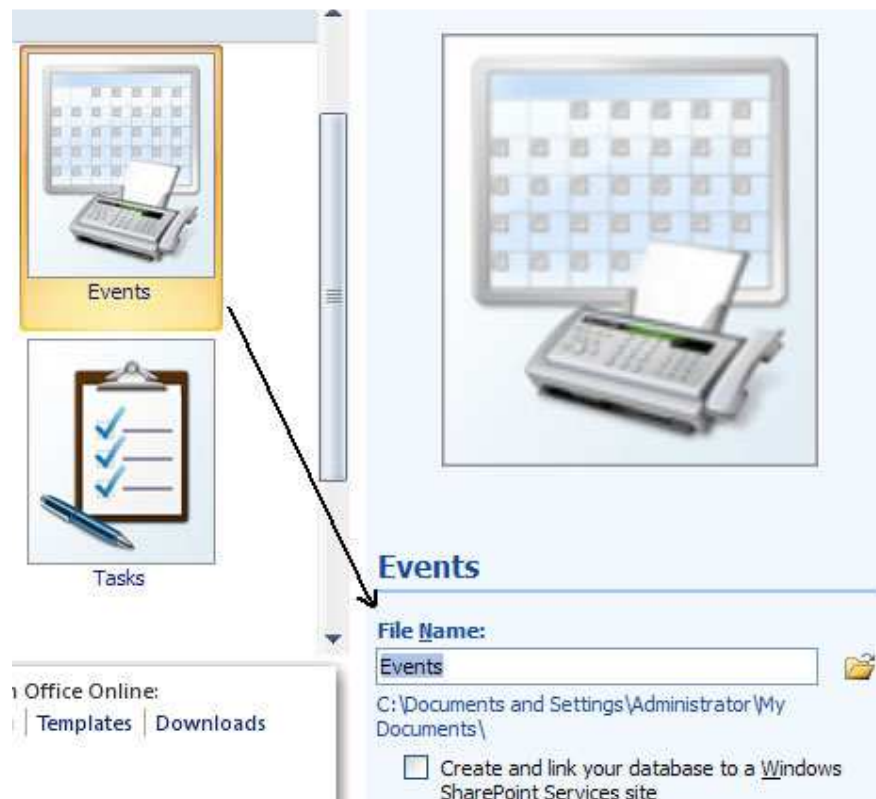
Để tạo một cơ sở dữ liệu mới từ các mẫu cơ sở dữ liệu:

- Kích Microsoft Office Button.
- Kích New.
- Chọn mẫu cơ sở dữ liệu bạn muốn tạo.



Hình 3: Tạo cơ sở dữ liệu từ Templates

- Nhập tên cho cơ sở dữ liệu.
- Kích nút **Download** để tạo một cơ sở dữ liệu mới từ mẫu đó.



Hình 4: Hoàn thành việc tạo Database từ Templates

## Bài 5: Tạo bảng

### Table Views

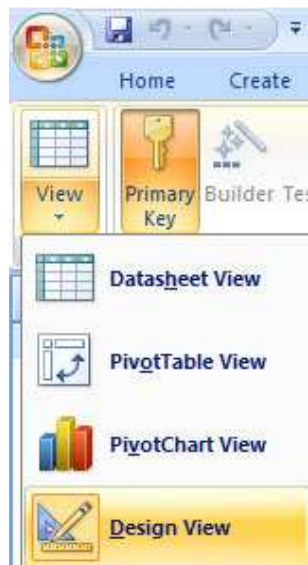
Có hai cách xem bảng trong Access và thêm dữ liệu vào bảng: **Design View** và **Datasheet View**.

Trong chế độ Design View, bạn có thể xem tất cả các trường với các kiểu dữ liệu và sự mô tả về bảng (nếu có).

Field Name	Data Type	Description
<b>ID</b>	AutoNumber	
name	Text	

Để xem bảng ở chế độ **Design View**:

- Kích mũi tên xuống của nút **View**.
- Chọn **Design View**.



Đối với các bảng mà đã được thêm vào cơ sở dữ liệu trước đó thì có thể không xem được (nếu muốn xem thì bạn chọn tên bảng đó, kích chuột phải vào bảng và chọn Design View).

Field Name	Data Type
Makhach	Text
Tenkach	Text
Diachi	Text
Dienthoai	Text

Field Properties

Chế độ Design view cho bảng DMKhach

Trong chế độ **Datasheet View**, bạn có thể hiển thị các bản ghi trong một bảng, mỗi một dòng là một bản ghi. Tiêu đề cột là các trường bạn định nghĩa cho cơ sở dữ liệu.

Để xem ở chế độ Datasheet View:

- Kích vào mũi tên xuống ở nút **View**.
- Chọn **Datasheet View**.

	Makhach	Tenkach	Diachi	Dienthoai	Add New Fie
K1		Đại lý Hoàng Minh	88 Thái Hà, Hà NỘI	048867567	
K2		Siêu thị Lê Lợi	32 Hàng Bông, Hà NỘI	047865432	
K3		ĐỒ Ngân Hà	12 Hàng Bài, Hà NỘI	046677882	
K4		Cửa hàng Toàn Thắng	76 Cửa Bắc, Hà NỘI	045635465	
K5		ĐỒ Sơn hai	66 Trần Phú, Hà NỘI	048367978	

Chế độ Datasheet View cho bảng DMKhach



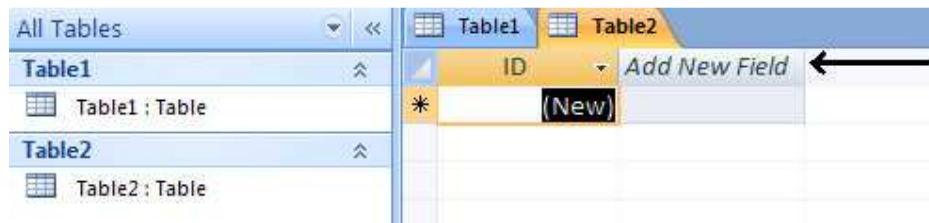
## Thêm các trường mới

Có nhiều cách để thêm các trường mới vào một cơ sở dữ liệu. Các trường mới có thể được thêm trong chế độ Datasheet View hoặc Design View.

Có hai cách để thêm một trường mới trong chế độ Datasheet View: **Add New Field** hoặc **New Field Button**.

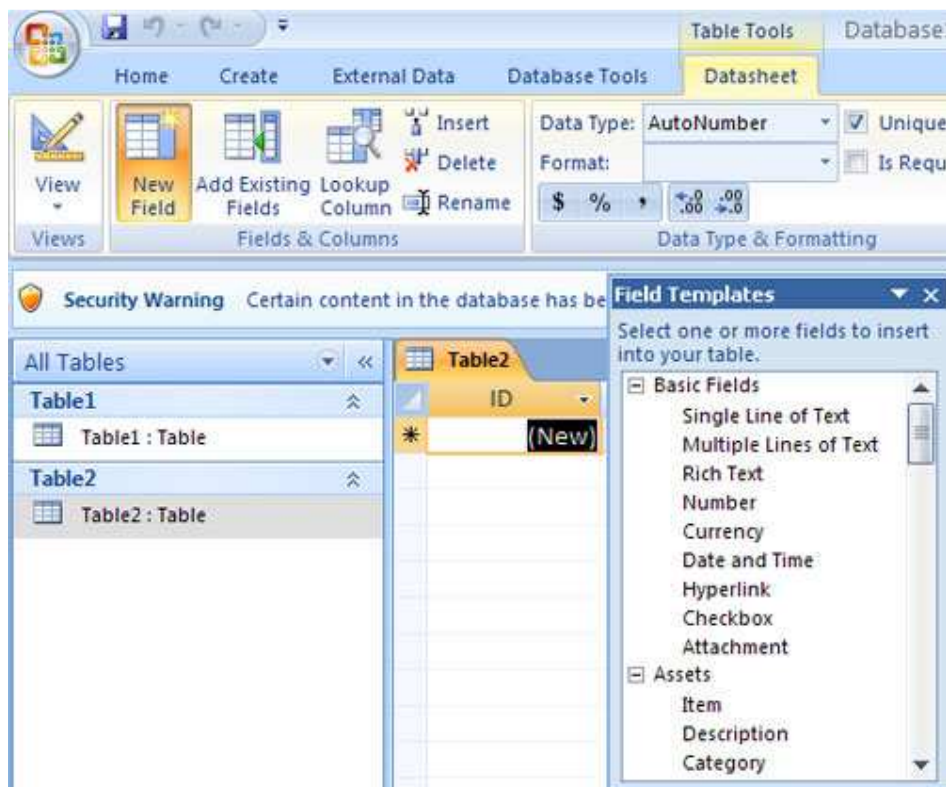
Để thêm một **New Field** trong **Datasheet**:

- Kích cột **Add New Field**



Để thêm một trường mới sử dụng nút **New Field**:

- Kích tab **Datasheet** trên vùng **Ribbon**.
- Chọn nút **New Field**.
- Chọn kiểu trường bạn muốn để thêm từ cửa sổ **Field Templates**.



Để thêm một trường mới trong chế độ **Design View**:

- Kích nút **Design View**.
- Kích vào trường sẵn có tiếp theo.
- Nhập tên trường vào cột **Field Name**.



## Các kiểu dữ liệu

Có nhiều kiểu dữ liệu mà một trường có thể được xác định. Khi tạo một trường mới trong cơ sở dữ liệu, bạn nên đặt kiểu dữ liệu thích hợp cho trường đó.

Bảng sau liệt kê các kiểu dữ liệu cho các trường:

Kiểu dữ liệu	Ý nghĩa
<b>Text</b>	Kiểu văn bản, số, hay chuỗi không quá 255 ký tự
<b>Memo</b>	Tương tự như trường văn bản, có thể chứa kiểu text, số, hay dữ liệu không quá 2GB.
<b>Number</b>	Kiểu số không quá 16bytes
<b>Date/Time</b>	Kiểu ngày và giờ
<b>Currency</b>	Kiểu tiền tệ không quá 8bytes và chính xác tới 4 số thập phân
<b>AutoNumber</b>	Access tạo một số duy nhất cho mỗi bản ghi mới. Thường dùng làm khóa chính cho bảng.
<b>Yes/No</b>	Kiểu logic (đúng/sai)
<b>OLE Object</b>	Kiểu nhúng và kết nối đối tượng (hình ảnh, tài liệu, đồ họa) không quá 2GB
<b>Hyperlink</b>	Địa chỉ siêu liên kết như các địa chỉ Web
<b>Attachment</b>	Đính kèm như ảnh, bảng tính, tài liệu và biểu đồ

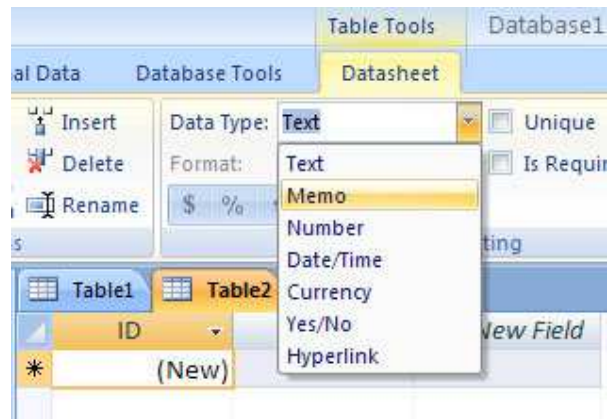
## Chỉnh sửa kiểu dữ liệu trong trường

Khi tạo các bảng, bạn nên xác định các kiểu dữ liệu của bảng sao cho phù hợp với kiểu dữ liệu mà được xác định cho trường.

Để chỉnh sửa kiểu dữ liệu trong **Datasheet View**:

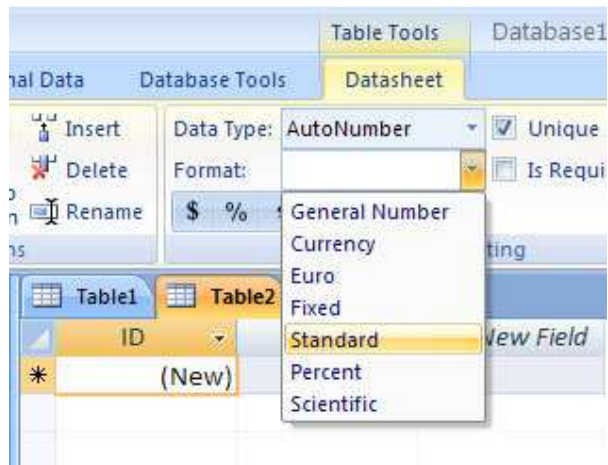
- Kích trường bạn muốn xác định.
- Chọn tab **Datasheet** trên vùng **Ribbon**.
- Kích mũi tên xuống phía bên phải **Data Type**.

- Chọn kiểu dữ liệu mà sẽ xác định cho trường đó.



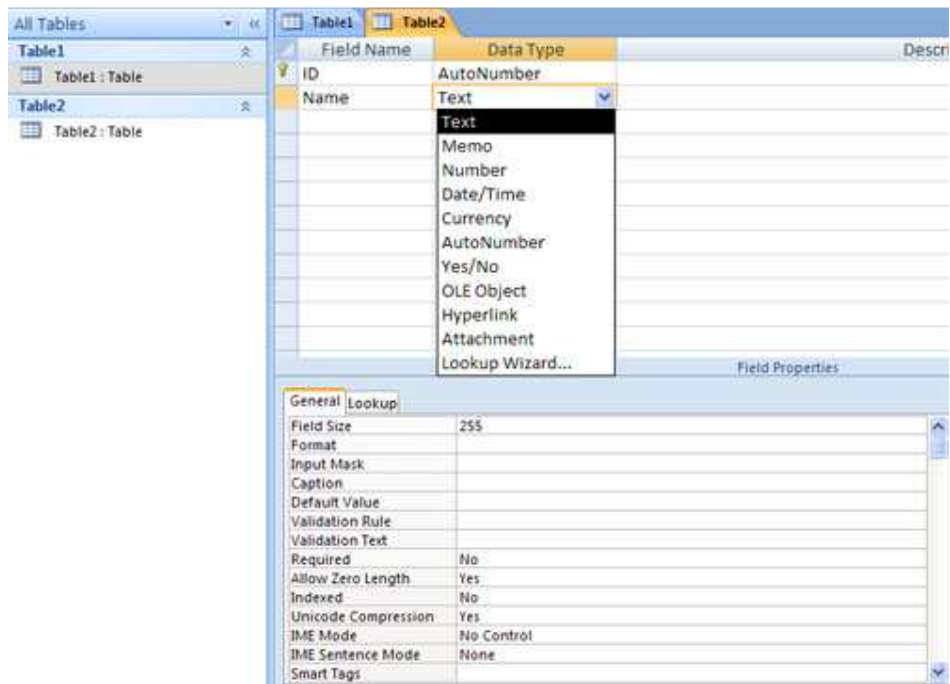
Để chỉnh sửa định dạng của dữ liệu:

- Kích trường bạn muốn định dạng.
- Chọn tab **Datasheet** trên **Ribbon**.
- Chọn mũi tên phía bên phải **Format** và chọn kiểu định dạng phù hợp.



Để chỉnh sửa **Data Type** trong chế độ **Design View**:

- Chọn **Design View**
- Chọn tên trường bạn muốn để định nghĩa hay tạo một trường mới.
- Kích **Data Type**.
- Chọn Data Type thích hợp.
- Định dạng trường trong hộp thoại **Field Properties**.



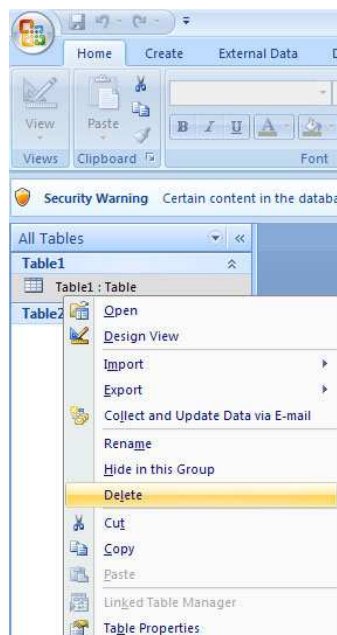
## Bài 6: Quản lý bảng

Trong việc quản lý bảng, chúng ta có thể xóa bảng, đổi tên, và tạo các mô tả về bảng sao cho phù hợp với nội dung chứa trong bảng dữ liệu.

### Xóa bảng

Để xóa bảng:

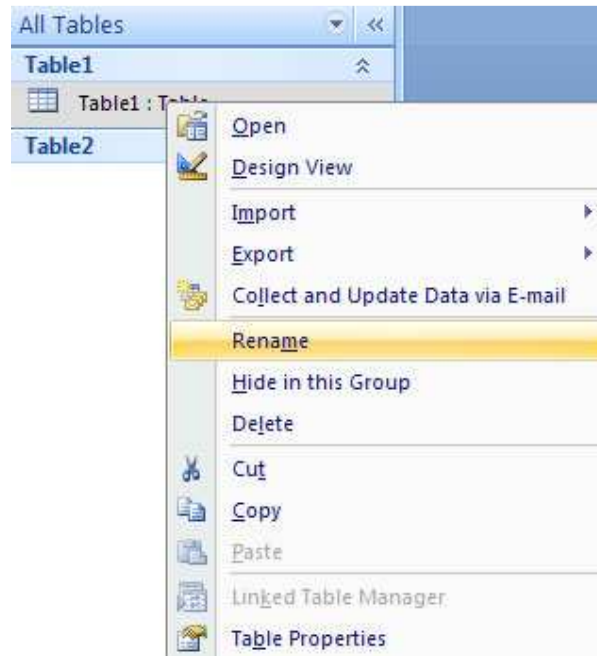
- Mở cơ sở dữ liệu bạn muốn bằng cách kích **Microsoft Office Button** và chọn **Open**.
- Kích phải vào bảng và chọn **Delete**.



## Đổi tên bảng

Để đổi tên một bảng:

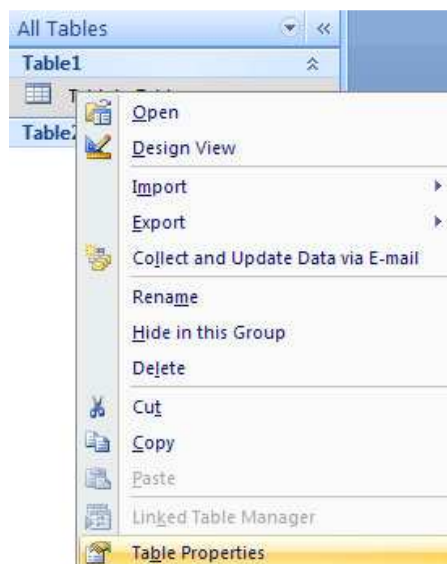
- Mở cơ sở dữ liệu từ Microsoft Office Button và kích Open.
- Kích phải vào bảng và chọn **Rename**.
- Nhập tên mới.



## Thêm phần mô tả cho bảng

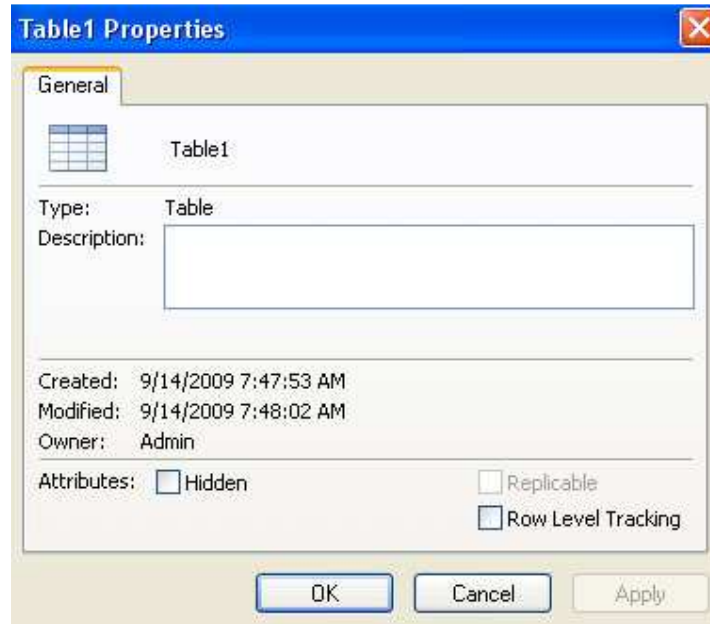
Để thêm phần mô tả cho bảng:

- Mở cơ sở dữ liệu bằng cách kích Microsoft Office Button và chọn Open.
- Kích phải vào bảng và chọn **Table Properties**.



- Kích vào vùng **Description**.

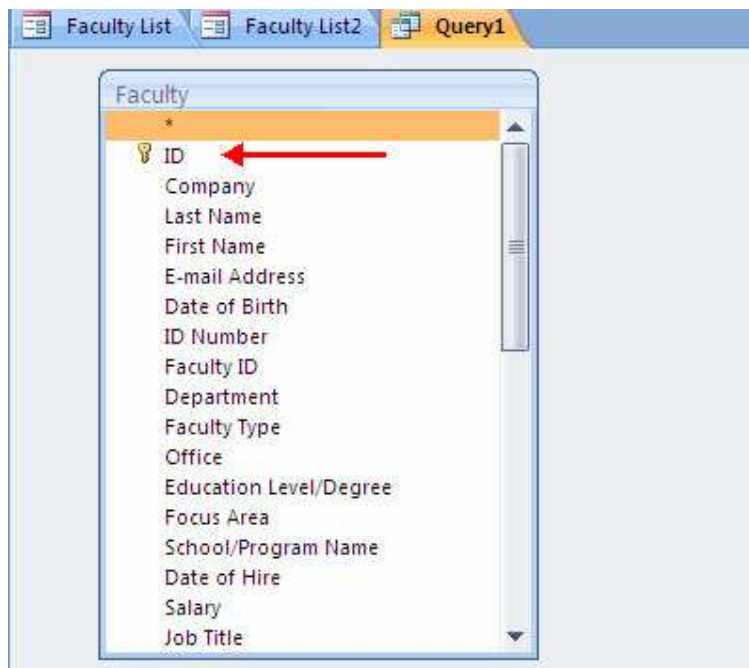
- Nhập phần mô tả.



## Bài 7: Khóa

### Khóa chính

**Khóa chính là một định danh duy nhất cho một bản ghi. Khóa chính không thể cùng tồn tại ở cả hai bản ghi.** Trường này có thể không bao giờ được để trống



### Khóa ghép

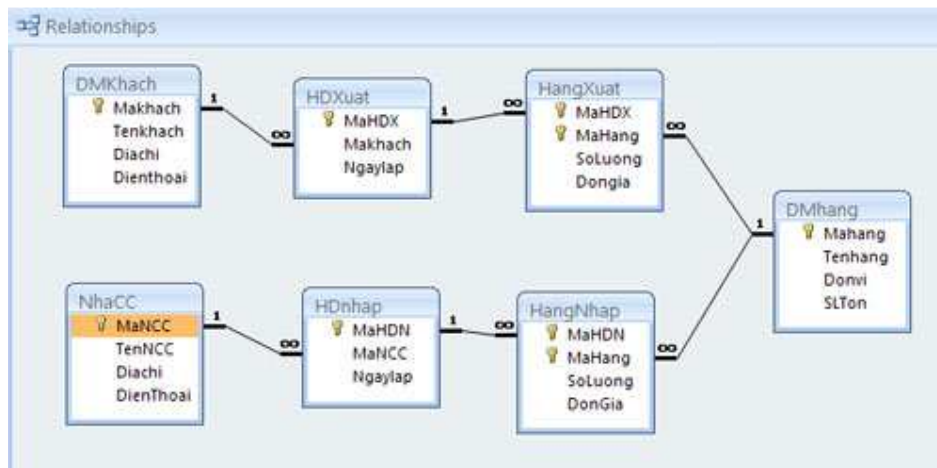
Một khóa ghép là một khóa chính mà bao gồm hai hoặc nhiều trường. Nó cũng có thể được gọi là khóa ràng buộc.

## Khóa ngoại

Một khóa ngoại là một trường hay sự kết hợp các trường được liên quan tới khóa chính của bảng khác.

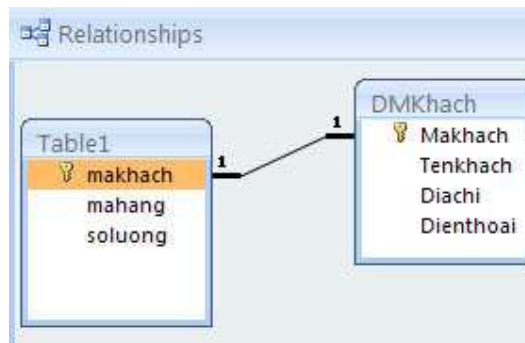
# Bài 8: Thiết lập quan hệ giữa các bảng

**Mỗi quan hệ giữa các bảng là sự liên kết dữ liệu giữa các bảng. Bằng cách xác định mỗi quan hệ bảng, bạn có thể kéo các bản ghi từ các bảng có liên quan dựa trên các trường quan hệ.**



## Quan hệ một-một

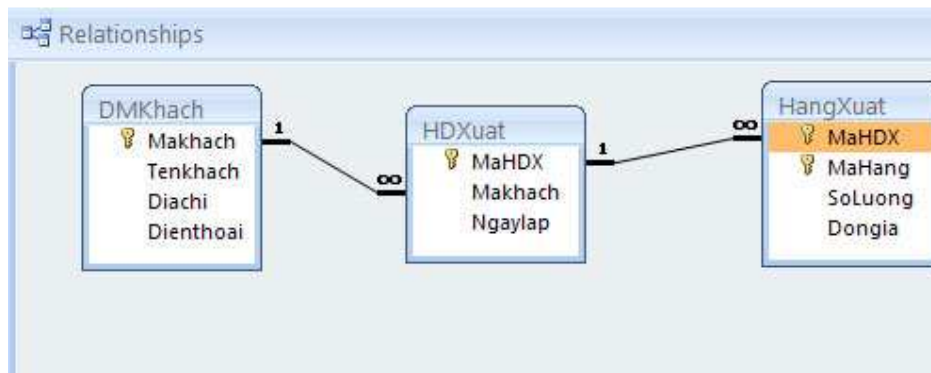
Quan hệ một-một là quan hệ giữa hai bảng mà được thiết lập khi trường quan hệ đều là khóa chính ở cả hai bảng. Mỗi bản ghi trong bảng A chỉ có thể có một bản ghi phù hợp trong bảng B.



Quan hệ một - một

## Quan hệ một - nhiều

Quan hệ một – nhiều là quan hệ giữa hai bảng mà khóa chính trong một bảng có thể được sao chép nhiều lần trong bảng khác hoặc có thể hiểu là một bản ghi trong bảng A chỉ có thể có nhiều bản ghi phù hợp trong bảng B, nhưng một bản ghi trong bảng B chỉ có một bản ghi phù hợp trong bảng A.

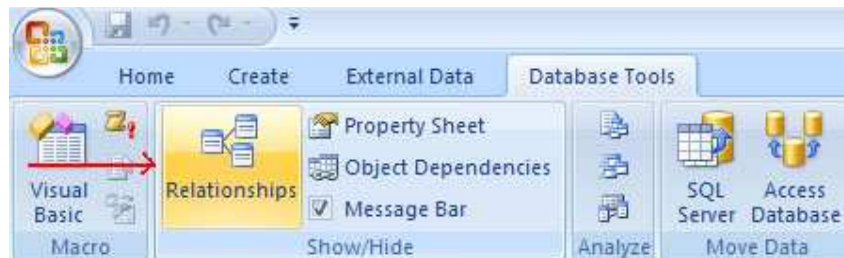


Quan hệ một - nhiều

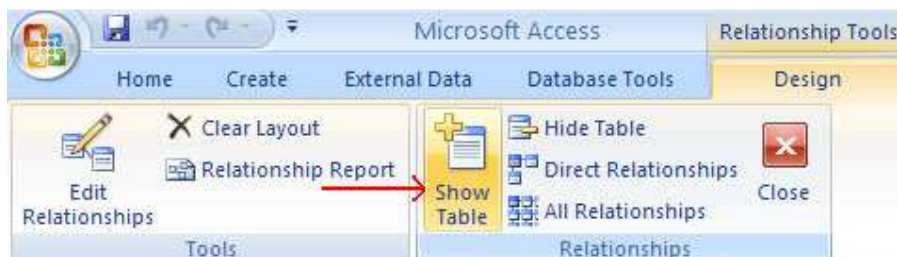
### Tạo mối quan hệ bảng

Để tạo mối quan hệ giữa các bảng:

- Kích vào tab **Database Tools** trên vùng **Ribbon**
- Chọn nút **Relationships**



- Kích vào tab **Design**
- Chọn **Show Table**



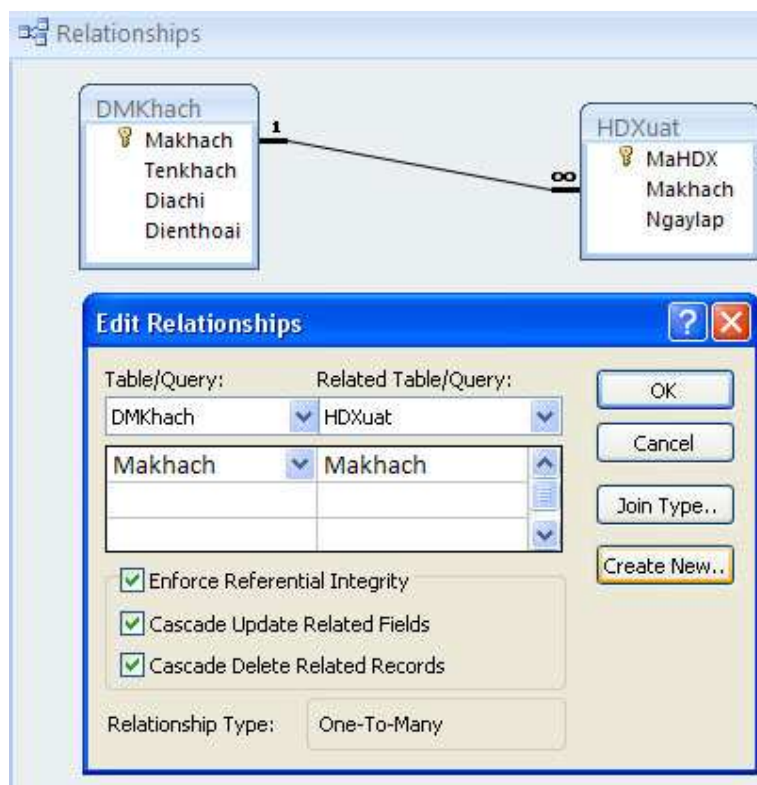
Lựa chọn các bảng muốn tạo quan hệ

- Kích nút **Add** để thêm các bảng
- Kích nút **Close** để tắt cửa sổ Show Table.



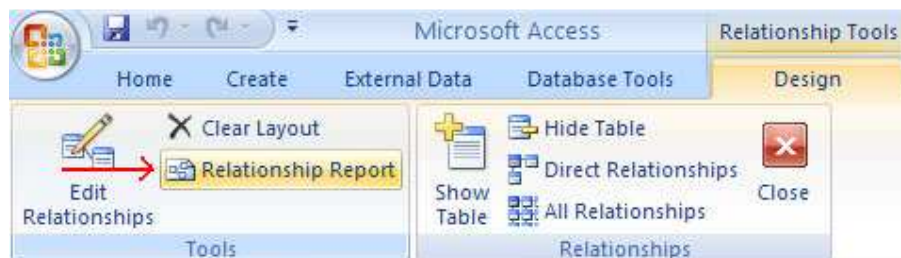


- Chọn trường bạn muốn để tạo một mối quan hệ từ đó
- Kéo một trường ghép với trường quan hệ trong bảng khác
- Kịch **OK**.

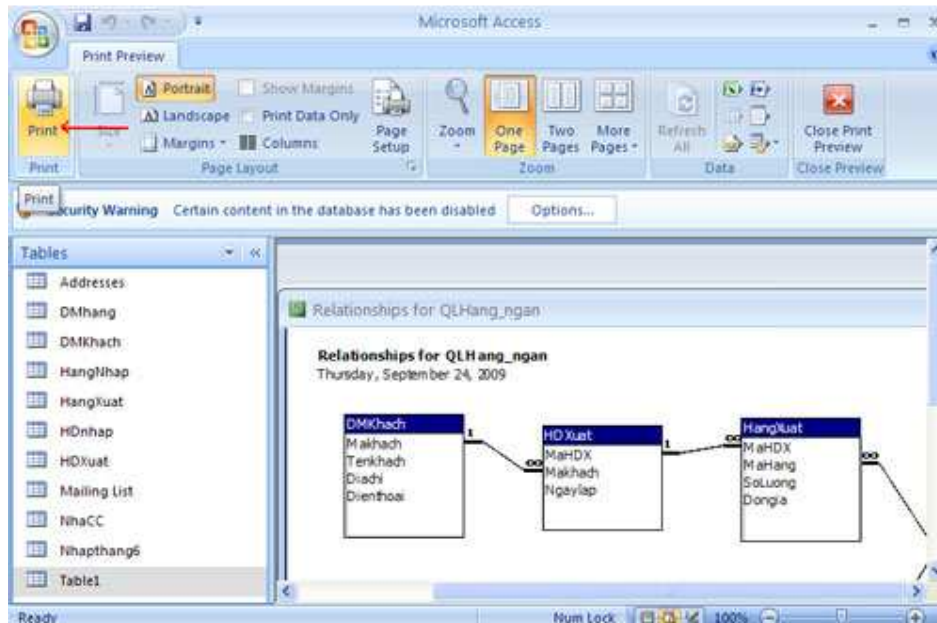


### In mối quan hệ bảng

- Kịch tab **Database Tools**.
- Kịch nút Relationships.
- Kịch nút **Relationship Report** trên tab **Design**.



- Kích nút **Print**.



## Bài 9: Quản lý dữ liệu

**Để dữ liệu hiển thị một cách linh hoạt và đúng theo yêu cầu của bạn.** Trong bài này, chúng tôi xin giới thiệu một số cách cơ bản để quản lý dữ liệu cho cơ sở dữ liệu.

### Thêm bản ghi vào bảng

Để thêm một bản ghi mới vào bảng:

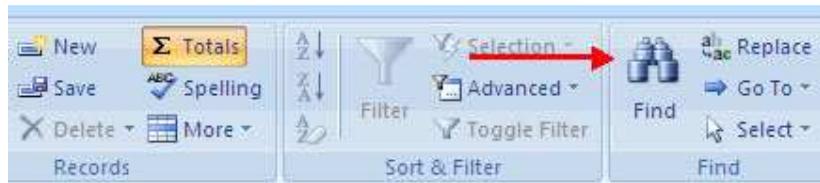
- Mở bảng ở chế độ **Datasheet View**
- Kích vào ô **(New)**
- Nhập bản ghi mới



## Tìm và Thay thế dữ liệu

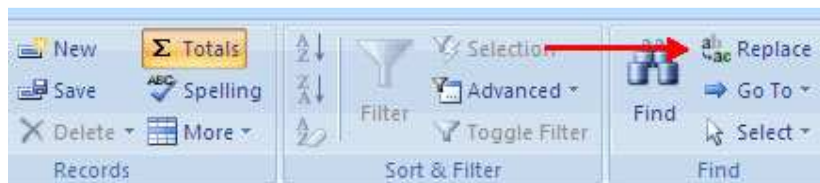
Để tìm dữ liệu:

- Kích nút **Find** trên tab **Home**.

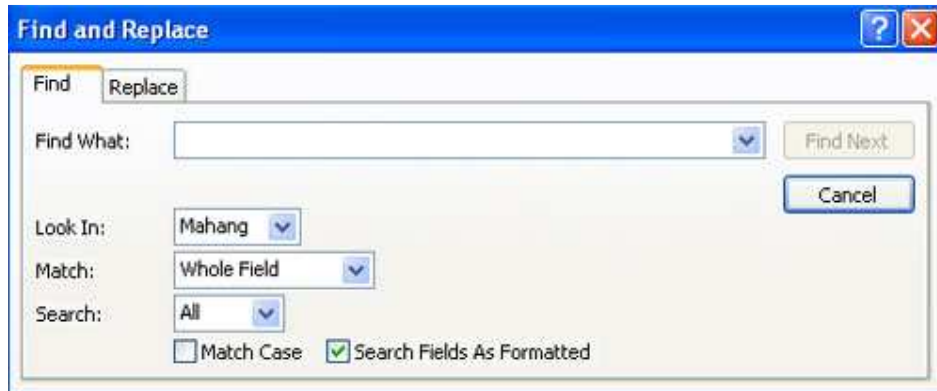


Để tìm và thay thế dữ liệu:

- Kích nút **Replace** trên tab **Home**.



Khi bạn tìm kiếm và thay thế cho dữ liệu, có một số tùy chọn trong hộp thoại Find có thể sử dụng:



Mục <b>Find What</b>	Nhập dữ liệu mà bạn muốn tìm
Danh sách trong mục <b>Look In</b>	Sử dụng để xác định bảng hay cột để tìm kiếm dữ liệu
Danh sách trong mục <b>Match</b>	Giúp bạn giới hạn việc tìm kiếm trong một trường hay phần đầu của một trường
Danh sách trong mục <b>Search</b>	Giúp bạn xác định hướng để tìm kiếm
Hộp chọn <b>Match Case</b>	Sử dụng hộp chọn này để xác định tìm kiếm theo chữ hoa hay chữ thường.

## Totals

Nút Totals cung cấp cho bạn thêm một dòng tính tổng vào cơ sở dữ liệu. Total có thể là **sum** (tổng), **average** (trung bình), **a count** (đếm), **minimum** (cực tiểu), **maximum** (cực đại), hoặc **variance** (khác nhau). Để thiết lập một dòng Total:

- Kích nút **Totals** trên tab **Home**



- Kích mũi tên xuống của ô nơi bạn muốn đặt một Totals
- Chọn Totals phù hợp.

	Mahang	Tenhang	Donvi	SLTon	Add New Field
	H1	Sea try me	hip	10	
	H2	Bias Ha Ni	thong	8	
	H3	Theca law Vine	Cay	9	
	H4	Đường	Kg	6	
	H5	Áo sớ mi dài tay	chic	7	
	H6	Cole	chic	5	
*				0	
	<b>Total</b>				

None

Sum

Average

Count

Maximum

Minimum

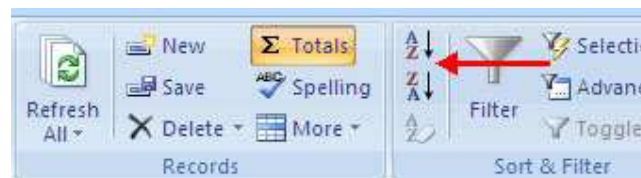
Standard Deviation

Variance

### Sắp xếp các bản ghi

Bạn có thể sắp xếp các bản ghi trong datasheet theo một cột hoặc hai cột liên nhau. Để sắp xếp các bản ghi theo một cột:

- Lựa chọn trường bạn muốn sắp xếp
- Kích vào nút **Sort Ascending** hoặc **Sort Descending**.



Để sắp xếp hai cột:

- Di chuyển các cột sao cho chúng sát nhau

- Lựa chọn các cột để sắp xếp bằng cách ấn phím Shift và kích chuột vào các cột
- Kích nút Sort Ascending hoặc Sort Descending.

	Mahang	Tenang	Donvi	SLTon	Add New Field
	H1	Sea try me	hip	10	
	H2	Bias Ha Ni	thong	8	
	H3	Theca law Vine	Cay	9	
	H4	Đường	Kg	6	
	H5	Áo sơ mi dài tay	chic	7	
	H6	Cole	chic	5	
*	Total			0	

Để xóa bỏ sắp xếp:

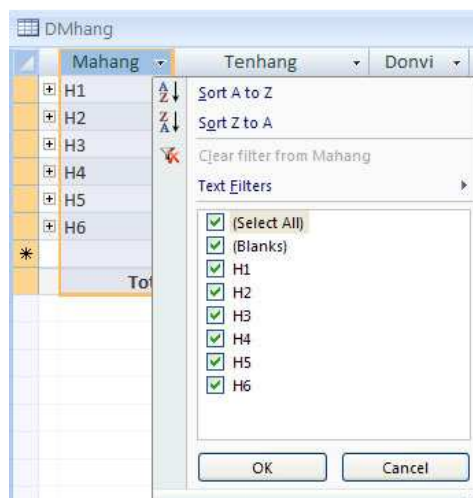
- Kích nút **Clear Soft**



## Filter (lọc)

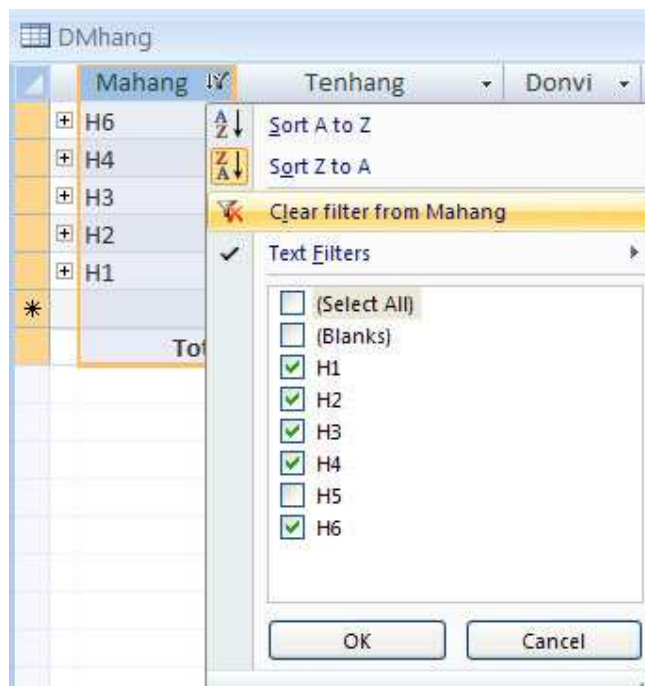
Bạn có thể lọc các bản ghi để hiển thị chúng theo mong muốn. Để lọc theo một cột:

- Mở cơ sở dữ liệu ở chế độ **Datasheet View**
- Kích vào mũi tên hướng xuống trong tên trường
- Chọn tiêu chuẩn lọc thích hợp
- Kích **OK**.



Để gỡ bỏ Filter:

- Kích nút Filter trên nhãn trường
- Kích **Clear Filter**
- Kích **OK**.



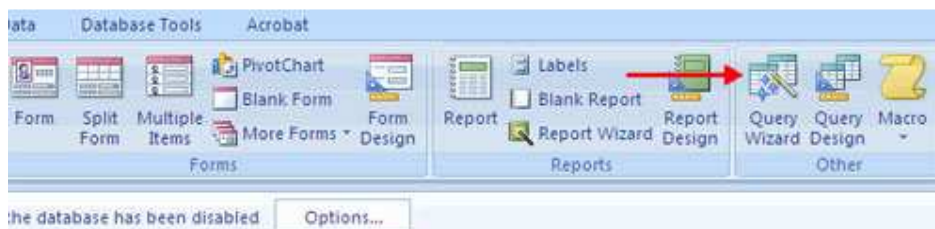
## Bài 10: Truy vấn cơ sở dữ liệu

**Truy vấn cho phép bạn lựa chọn và lọc dữ liệu từ nhiều bảng.** Các truy vấn có thể được lưu và sử dụng bất cứ lúc nào cần thiết.

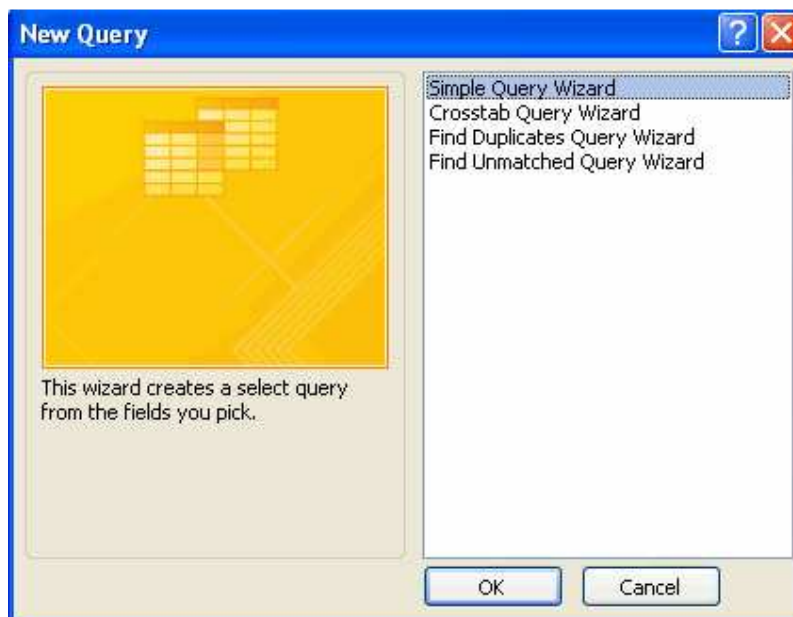
### Truy vấn Wizard


Query Wizard sẽ giới thiệu từng bước để xây dựng một truy vấn. Để thực thi một truy vấn sử dụng Query Wizard:

- Kích tab **Create**
- Kích nút **Query Wizard**



- Chọn kiểu truy vấn bạn muốn thực thi
- Kích **OK**.

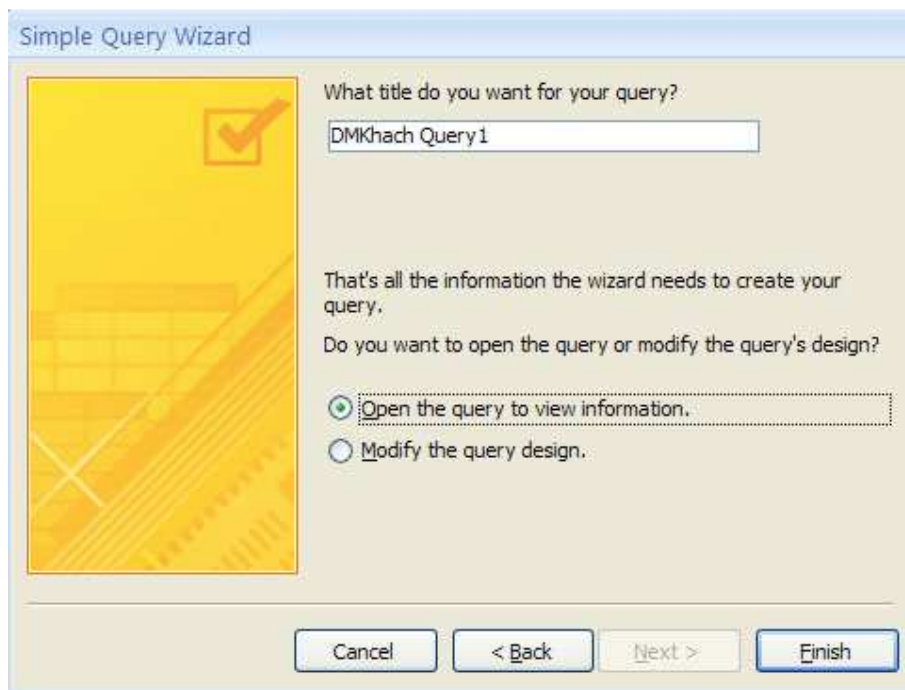


- Chọn các trường bạn muốn có trong mỗi bảng bằng cách chọn trường và kích nút 
- Để lựa chọn các trường từ nhiều bảng khác nhau, kích vào mũi tên xuống của **Table/Queries**
- Kích **Next**.



### Hoàn thành Query wizard

- Nhập tên cho truy vấn
- Kích **Finish**.



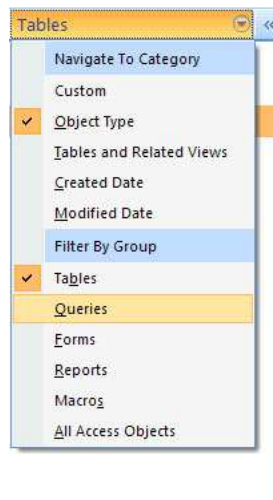
- Kết quả truy vấn sẽ hiển thị như sau:

Makhach	Tenkhach	Diachi	Dienthoai
K1	Đại lý Hoàng Minh	88 Thái Hà, Hà Nội	048867567
K2	Siêu thị Lê Lợi	32 Hàng Bông, Hà Nội	047865432
K3	Đỗ Ngân Hà	12 Hàng Bài, Hà Nội	046677882
K4	Cửa hàng Toàn Thắng	76 Cửa Bắc, Hà Nội	045635465
K5	Đỗ Sơn hai	66 Trần Phú, Hà Nội	048367978
*			

Để chuyển đổi giữa bảng và truy vấn:

- Mở **Navigation Pane**

- Kích đúp vào tên bảng hoặc truy vấn để xem.



**Tính năng thiết kế truy vấn**



Bạn có thể thiết kế một truy vấn với nút **Query Design**. Để thiết kế một truy vấn sử dụng Query Design:

- Kích nút **Query Design** trên tab **Create**.



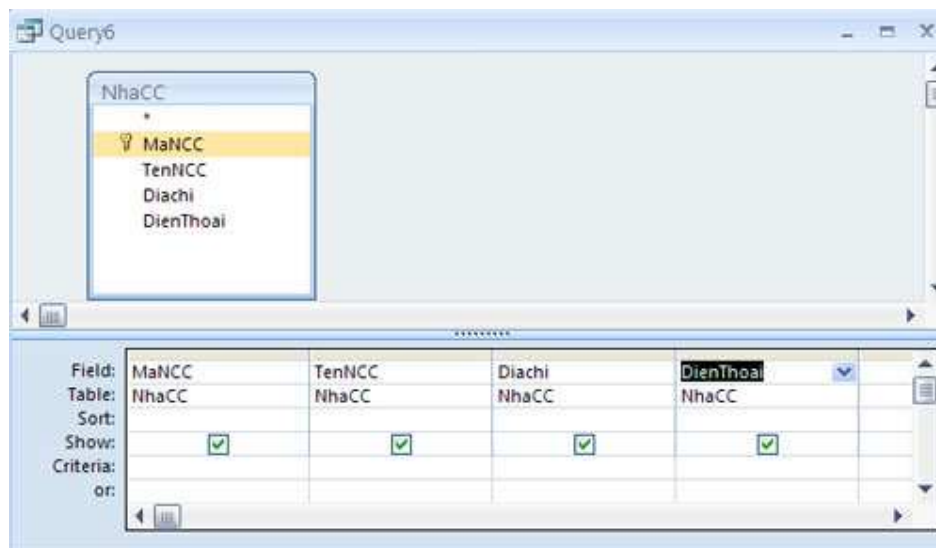
- Hộp thoại **Show Table** xuất hiện cho phép bạn lựa chọn bảng

- Kích **Add** để thêm bảng và kích nút **Close** để đóng hộp thoại Show Table

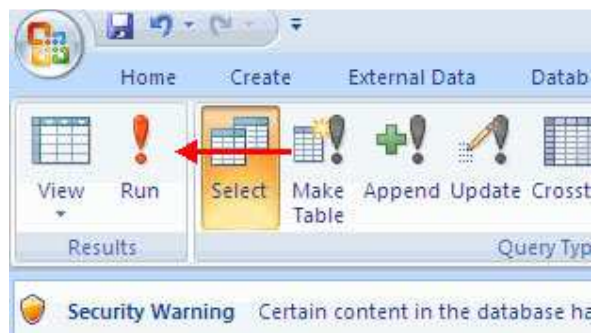


- Kích đúp vào tên trường bạn muốn nó xuất hiện trong lưới thiết kế

- Lặp lại bước trên đối với các trường khác



- Kích nút **Run** để thực thi truy vấn.



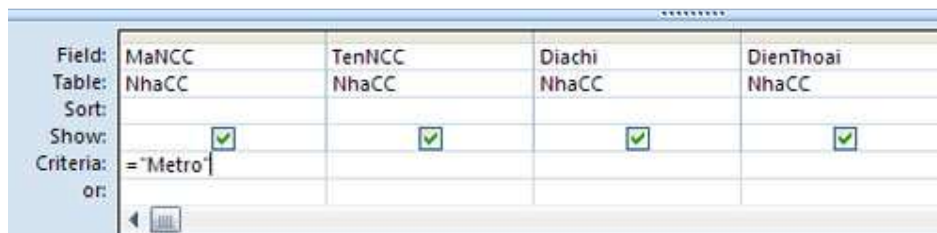
## Thiết kế truy vấn kèm theo các điều kiện

Tiêu chuẩn truy vấn là sự kiểm tìm các điều kiện được sử dụng trong truy vấn để lấy dữ liệu. Bạn có thể thiết lập tiêu chuẩn là một số cụ thể hay chuỗi dữ liệu, hay vùng dữ liệu.

"value"	Chỉ hiển thị bản ghi phù hợp với giá trị chính xác (thay thế từ value bằng từ bạn muốn tìm)
=	Bằng
<	Nhỏ hơn
<=	Nhỏ hơn hoặc bằng
>	Lớn hơn
>=	Lớn hơn hoặc bằng
<>	Khác nhau
Between X And Y	Trong một khoảng (lấy giá trị trong khoảng giữa X và Y)
Is Null	Giá trị Null
And	Chỉ đúng nếu điều kiện tồn tại
Or	Đúng nếu một trong hai điều kiện tồn tại
Not	Đúng nếu 1 trường hợp không đúng

Để tạo điều kiện cho truy vấn:

- Kích vào truy vấn bạn muốn thêm điều kiện
- Nhập điều kiện truy vấn cho trong mục Criteria



- Sau đó kích nút **Run** để thực thi truy vấn.

# Bài 11: Trường tính toán

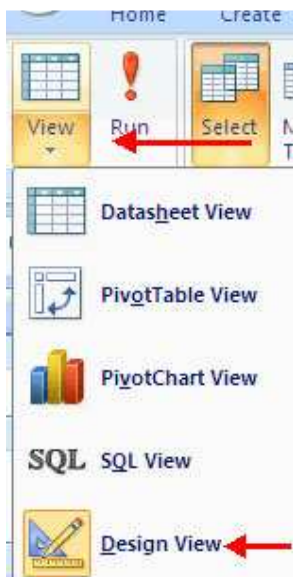
**Trường tính toán là một trường mà nhận được thông tin của nó từ sự tính toán được thực hiện trên các trường khác.** Bạn có thể xây dựng miền kết quả tính toán trong màn hình Query bằng cách sử dụng phép cộng (+), trừ (-), nhân (\*) và chia (/).

## Biểu thức

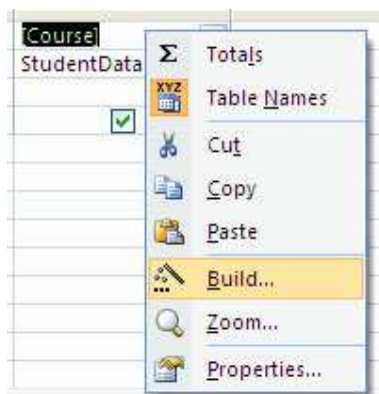
Biểu thức là sự kết hợp các hàm, tên trường, số, văn bản, và các phép toán được liệt kê phía trên.

Để xây dựng một biểu thức để tạo một trường tính toán:

- Mở một truy vấn có sẵn hoặc bắt đầu một truy vấn mới
- Kích vào nút **View**
- Kích chọn chế độ **Design View**.

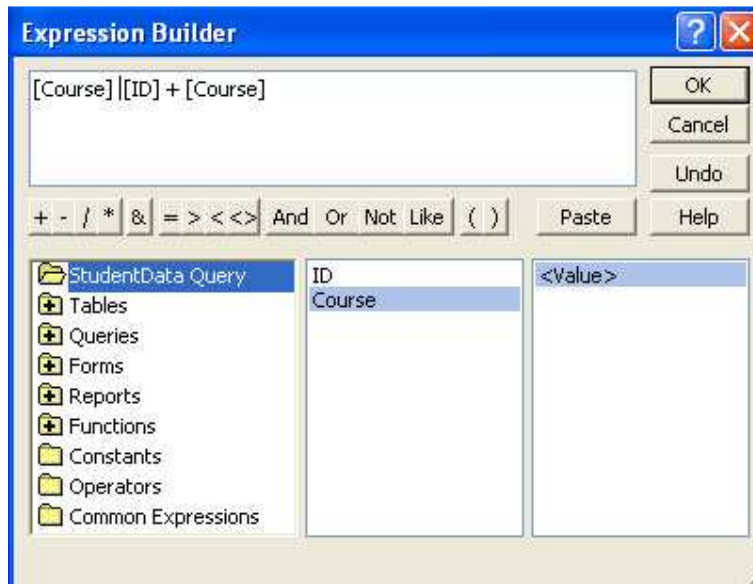


- Trong ô **Query**, kích phải vào trường mà bạn muốn tạo phép toán
- Kích nút **Build**.

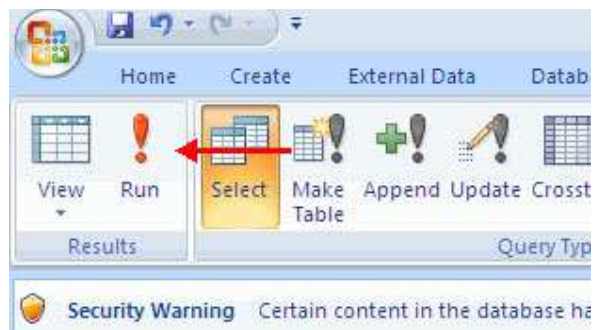


- Chọn bảng mà bạn muốn để xây dựng phép toán từ đó

- Kích đúp vào trường mà bạn muốn gộp nó trong phép toán
- Chọn toán tử mà bạn muốn có trong phép toán
- Kích vào trường thứ hai bạn muốn gộp trong phép toán
- Kích **OK**.



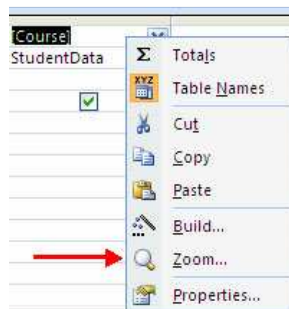
- Kích nút **Run** để thực thi.



## Zoom

Hộp thoại Zoom cho phép bạn xem toàn bộ biểu thức. Để xem hộp thoại Zoom:

- Trong chế độ **Design View**, kích phải vào trường bạn muốn hiển thị.
- Kích **Zoom**.



# Bài 12: Thiết kế Forms

Forms cho phép bạn điều khiển giao diện của màn hình nhập dữ liệu và các báo cáo được tạo ra.

## Form Views

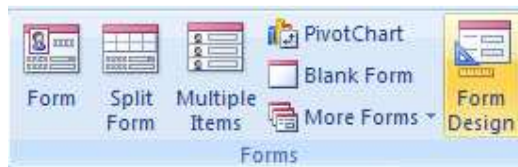
Có 3 cách để xem Forms trong Access:

Design View	Cho phép bạn thiết kế Form bao gồm header, footer và các chi tiết trong Form. Bạn cũng có thể thêm hình ảnh và các điều khiển xuất hiện trên Form.
Form View	Đây là một trang động cho phép người dùng nhập và chỉnh sửa dữ liệu hoặc điều chỉnh dữ liệu trong một lĩnh vực.
Layout View	Cách xem này cho phép bạn thiết kế Form và thao tác dữ liệu.

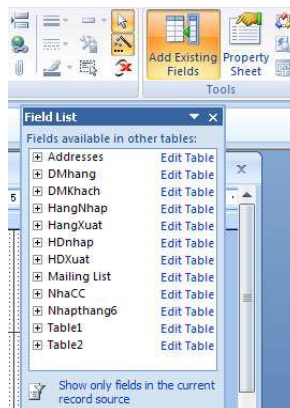
## Tạo form

Bạn có thể tạo một Form từ một bảng hay một truy vấn. Để tạo một Form:

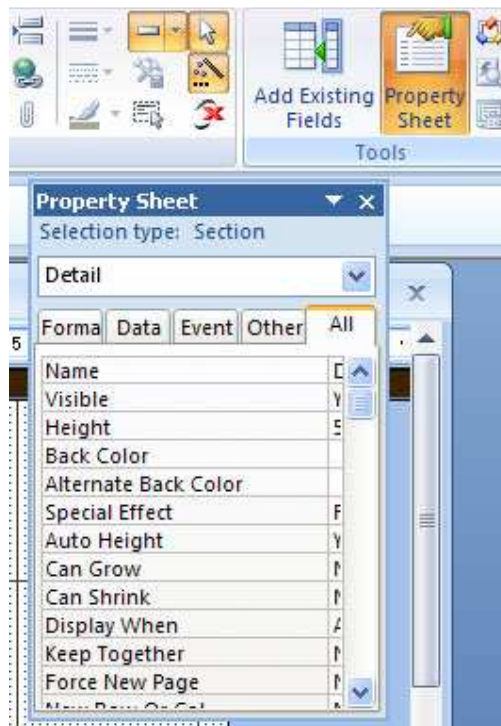
- Kích tab **Create**
- Kích nút **Form Design**



- Kích tab **Format**
- Kích vào nút **Add Existing Fields**
- Trong hộp **Field List** phía bên phải, kích và kéo các trường bạn muốn lên Form

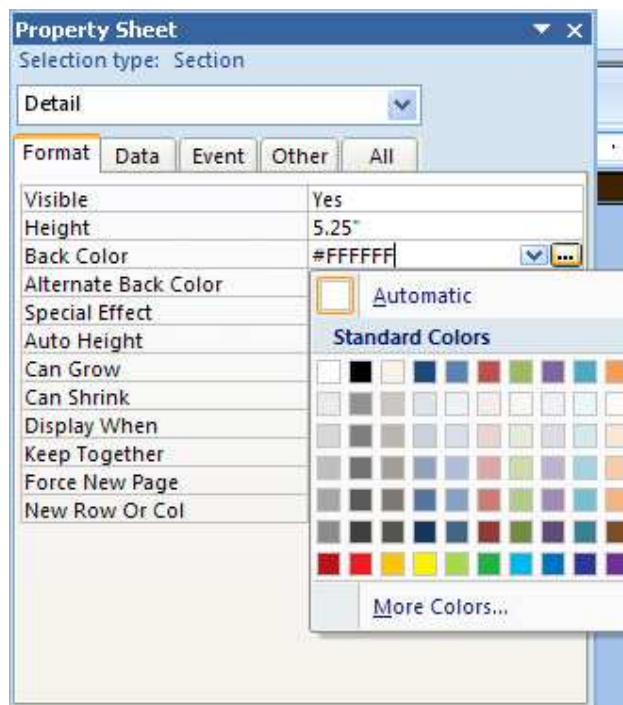


- Để thay đổi màu sắc và font, kích nút **Property Sheet** trên tab **Arrange**



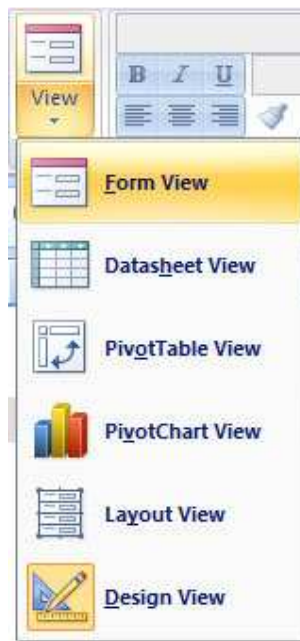
Chỉnh sửa các thuộc tính trong hộp thoại **Property Sheet**

- Chọn **Section** bạn muốn chỉnh sửa
- Chọn thuộc tính bạn muốn chỉnh sửa



Để xem trước Form:

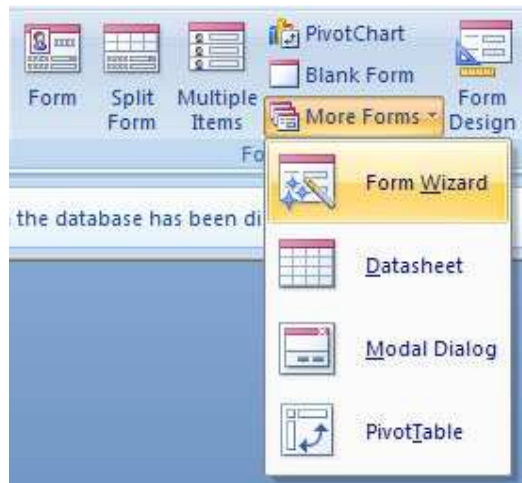
- Kích nút **Views** trên tab Home
- Kích nút **Form View**



## Form Wizard

Bạn có thể tạo Form với sự trợ giúp từ Form Wizard. Để sử dụng Form Wizard:

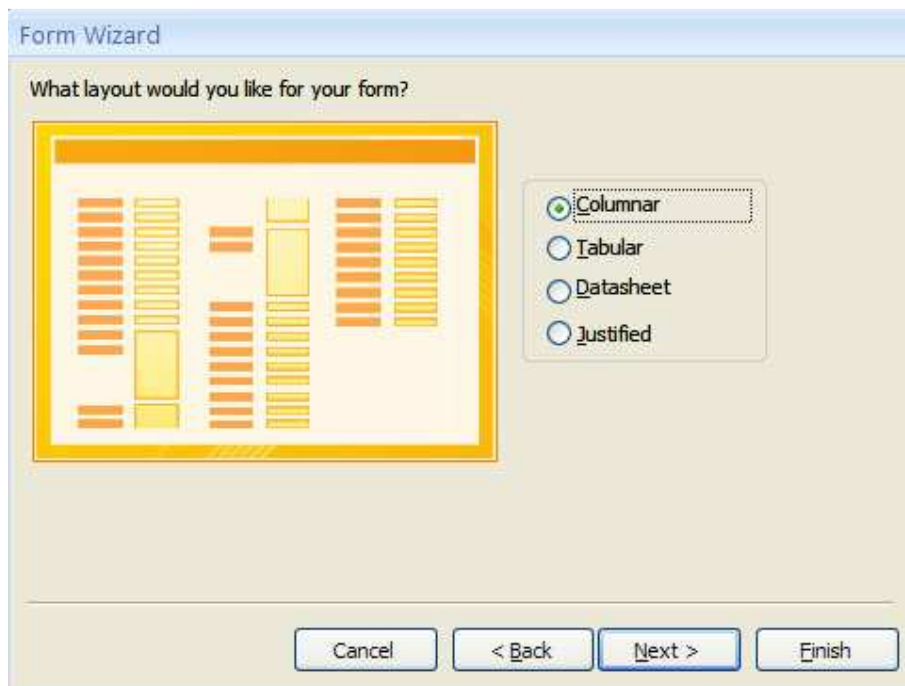
- Trên tab **Create**, kích vào mũi tên xuống chỗ **More Forms**
- Kích **Form Wizard**.



- Chọn **Tables/Queries** mà bạn muốn có trên Form
- Chọn các trường bạn muốn cho lên Forms
- Kích **Next**.

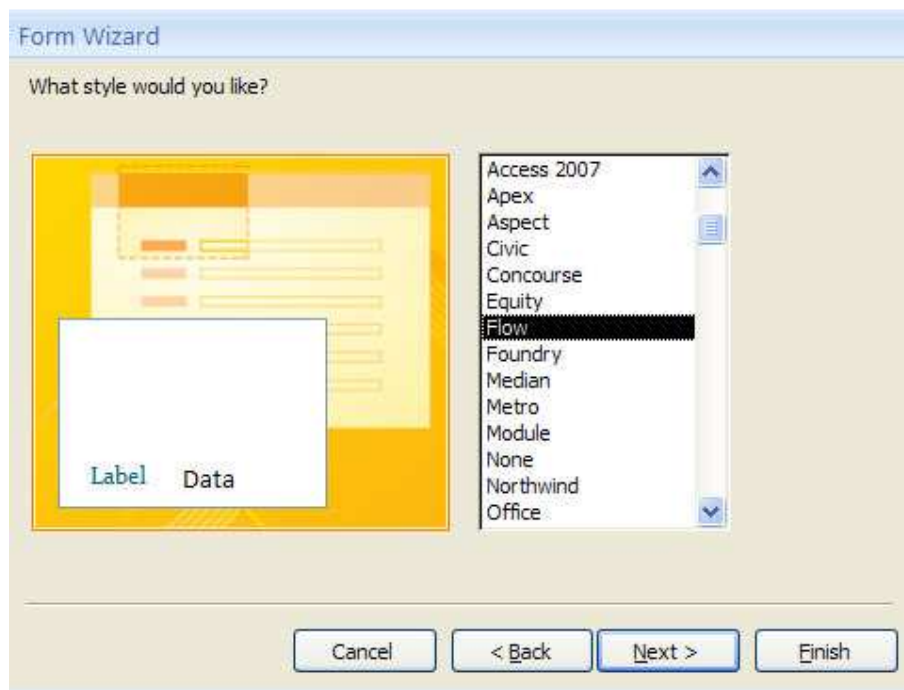


- Chọn cách bố trí cho form
- Kích **Next**.

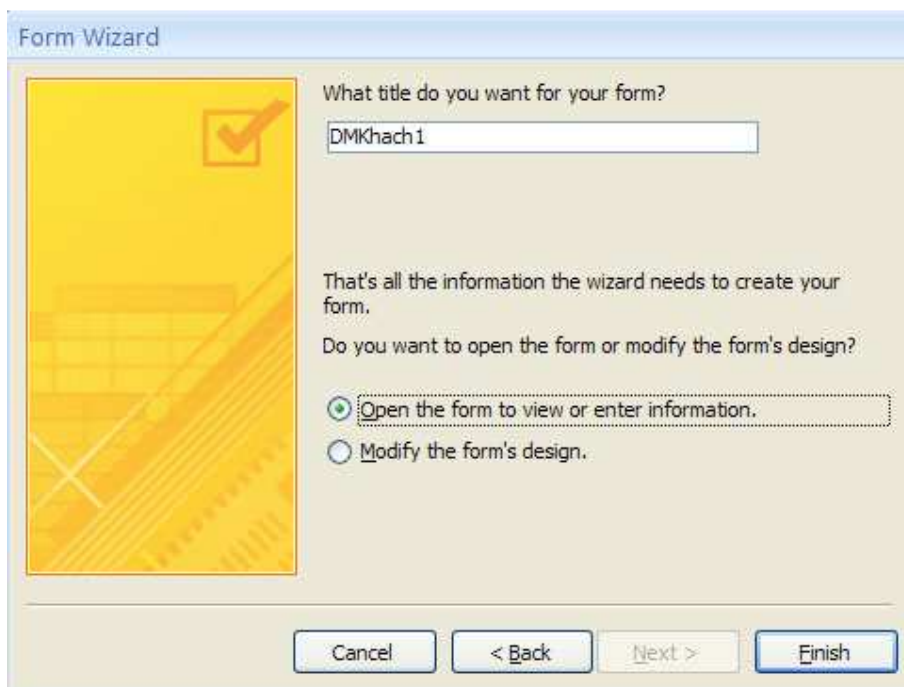


- Chọn kiểu
- Kích **Next**.





- Tạo tiêu đề cho Form
- Chọn nơi bạn muốn để mở Form hay chỉnh sửa thiết kế Form.
- Kích **Finish**.



## Bài 13: Tạo Reports (báo cáo)

**Reports (báo cáo)** là một cách thức để xem và phân tích một số lượng lớn dữ liệu. Bạn có thể sử dụng Report Wizard hay tạo một báo cáo tùy ý mà bạn cần.

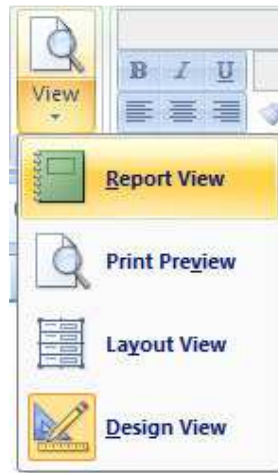
## Report Views

Reports có thể được hiển thị theo 4 chế độ sau:

Design View	Chế độ Design View cung cấp cho bạn với cấu trúc của báo cáo. Bạn có thể thêm, chỉnh sửa hay xóa các thành phần của report. Nhưng bạn không thể thao tác dữ liệu trong nhiều bảng được kết hợp với báo cáo.
Report View	Report View cho phép bạn xem dữ liệu từ bảng nhưng không thay đổi cách bố trí của báo cáo.
Layout View	Chế độ này cho phép bạn xem dữ liệu từ bảng và có thể thêm, chỉnh sửa, xóa các thành phần của báo cáo.
Print Preview	Print Preview cho phép bạn xem báo cáo sẽ trông như thế nào khi được in ra.

Để thay đổi chế độ xem báo cáo:

- Kích vào nút **View** trên tab **Home**



## Tạo một Report

Để tạo một báo cáo trắng:

- Kích nút **Blank Report** trên tab **Create**



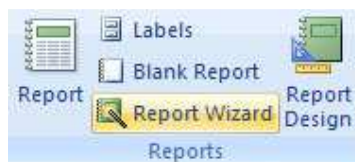
- Kích nút **Add Existing Fields**
- Từ danh sách các trường, kích và kéo các trường lên báo cáo



## Report Wizard

Để tạo một báo cáo sử dụng Report Wizard:

- Trên tab **Create**, kích nút **Report Wizard**



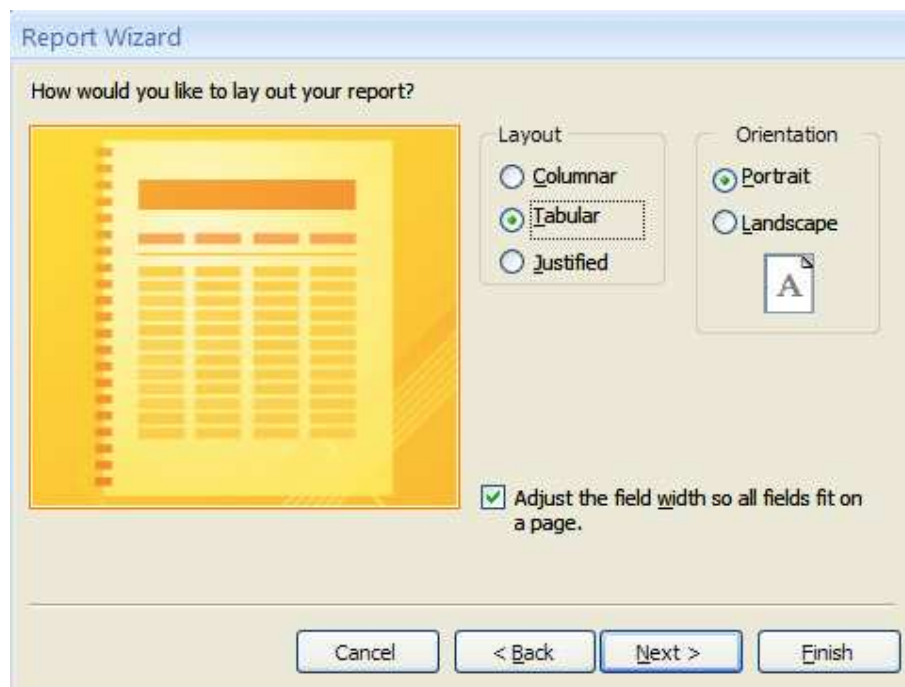
- Chọn **Tables/Queries** mà bạn muốn có trên báo cáo
- Chọn các trường bạn muốn cho lên báo cáo rồi ấn nút > (nếu bạn chọn từng trường) hoặc ấn nút >> (nếu bạn chọn tất cả các trường).
- Kích **Next**



- Chọn sắp xếp theo thứ tự cho báo cáo
- Kích **Next**



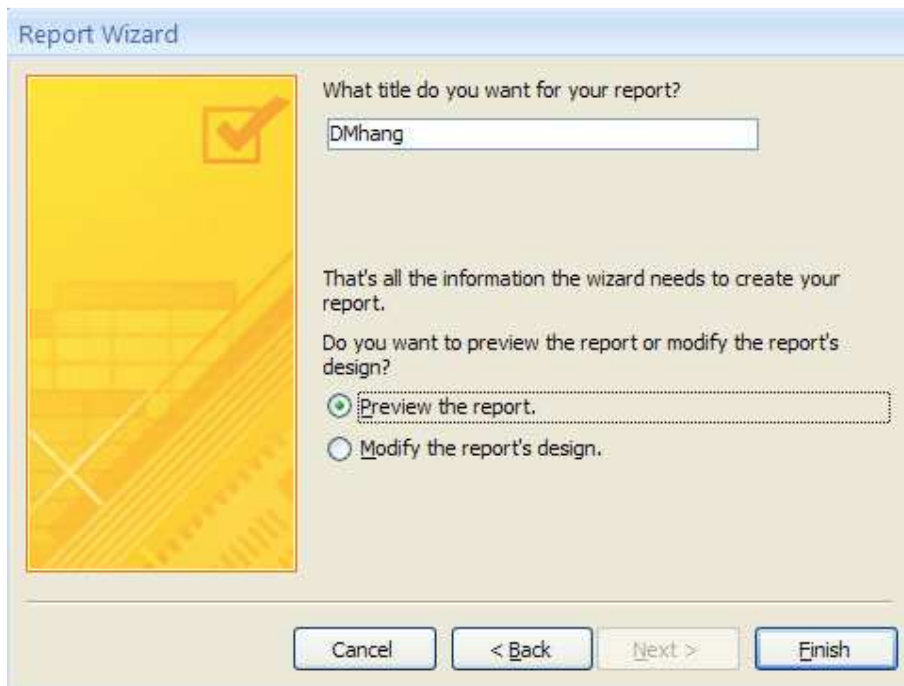
- Chọn cách bố trí (**Layout**) và định hướng quay ngang hay dọc (**Orientation**) cho báo cáo
- Sau đó kích **Next**



- Chọn kiểu cho báo cáo
- Kích **Next**



- Tạo một tiêu đề cho báo cáo
- Chọn cách bạn muốn mở báo cáo để xem (**Preview the report**) hoặc chỉnh sửa báo cáo (**Modify the report's design**)
- Kích **Finish**



### Trường tính toán

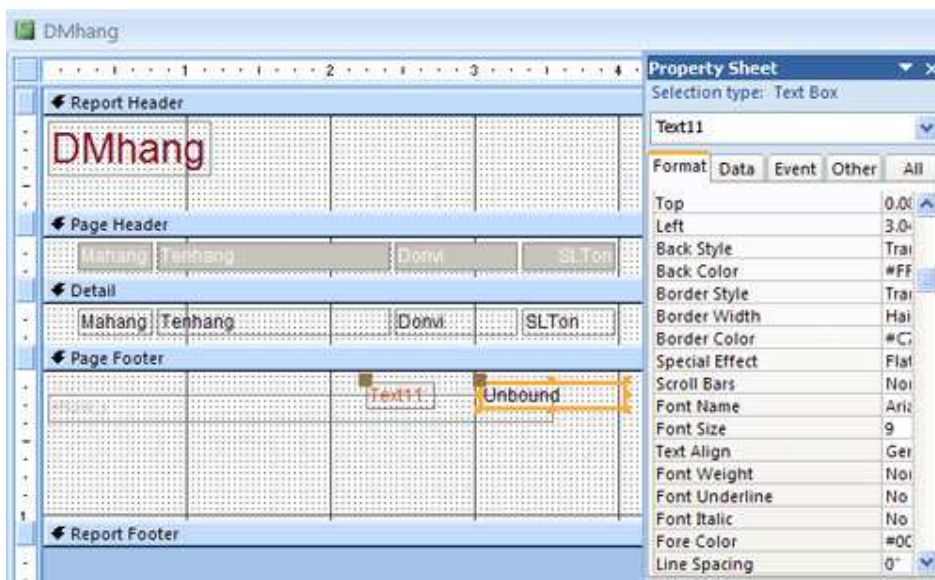
Bạn có thể tạo các báo cáo mà trong đó có cả trường tính toán. Những báo cáo này sẽ hiển thị thông tin mà bạn muốn với một phép toán số học. Để thêm một trường tính toán vào báo cáo:

- Mở Report bạn muốn tạo thêm trường tính toán

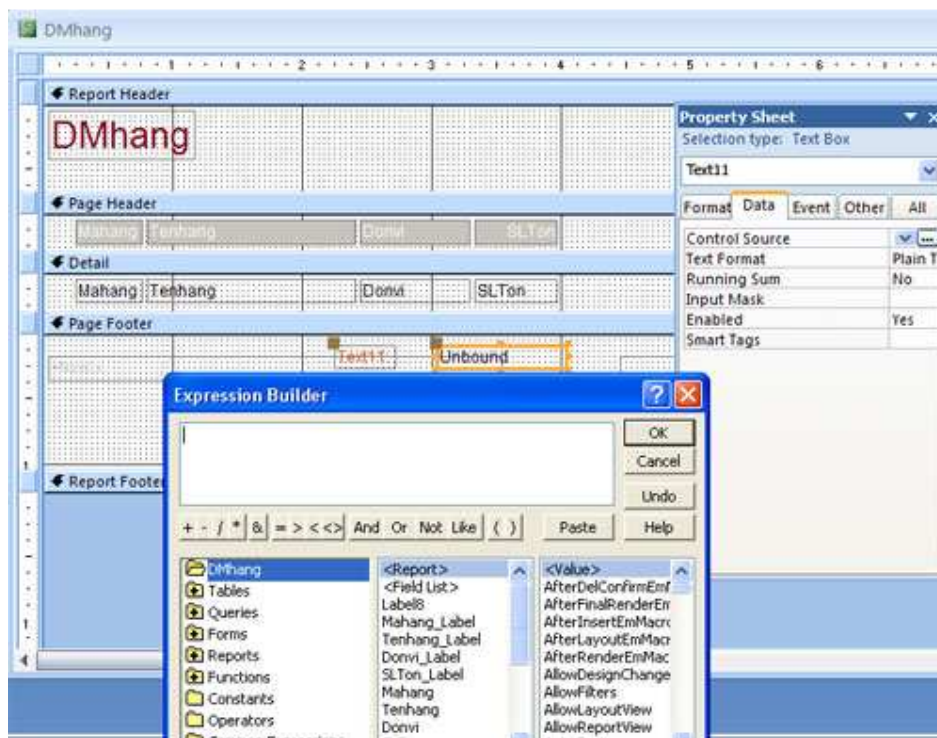
- Kích nút **View**
- Kích **Design View**
- Kích tab **Design**
- Kích nút **Text Box**



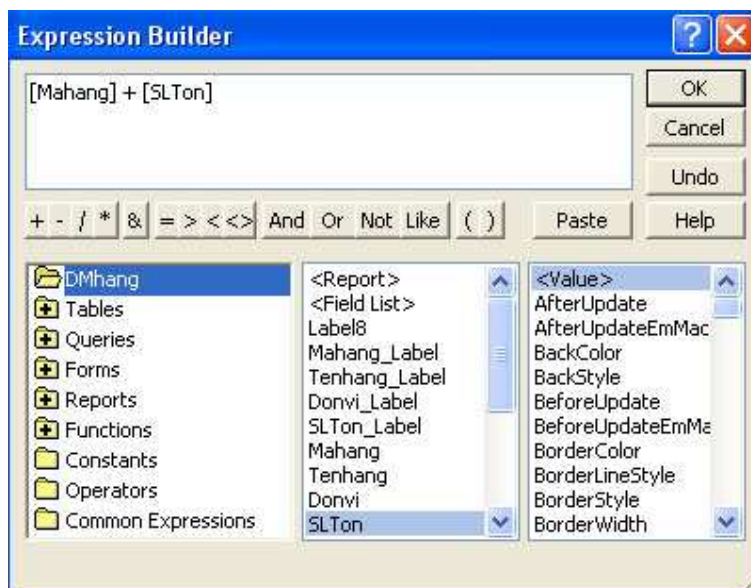
- Kích vào phần trên báo cáo, nơi bạn muốn định vị textbox
- Kích nút **Property Sheet**



- Kích tab **Data**
- Kích vào nút phía bên phải của **Control Source** có ký hiệu ba chấm



- Chèn thêm các trường bạn muốn có trong tính toán
- Kích **OK**.



## Bài 14: In báo cáo

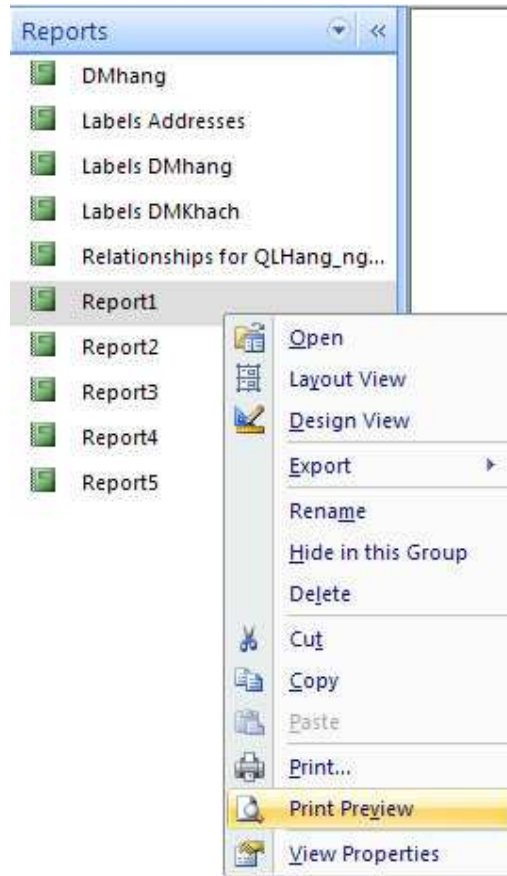
Sau khi bạn tạo một báo cáo, bạn có thể in báo cáo.

### In báo cáo

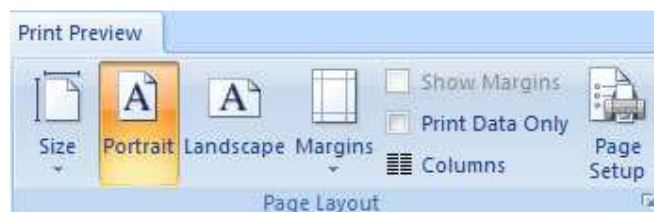
Để in báo cáo:

- Lựa chọn báo cáo bạn muốn in

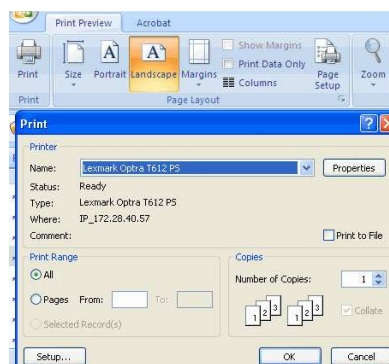
- Kích phải vào tên báo cáo
- Kích **Print Review**



- Chọn bố cục, lề và kích thước giấy phù hợp trong nhóm **Page Layout**



- Sau đó kích **Print**
- Kích **OK**.





# Bài 15: Tạo một bảng từ ban đầu

Ở phần này, bạn đã sẵn sàng để tạo một bảng cho cơ sở dữ liệu mới. Trong chương 2, bạn đã sử dụng Table Wizard để tạo bảng một cách dễ dàng. Cách này không chỉ đưa ra rằng việc tạo bảng bằng tay là rất khó; nhưng nó là cách chắc chắn để hiểu rõ về bảng.

Trong phần còn lại của chương này, bạn sẽ học cách để phát triển bảng không thông qua việc sử dụng Table Wizard. Trong quá trình làm phần này, bạn sẽ sử dụng các thông tin được học ở chương 3, "Tùy biến các thành phần". Bảng này sẽ chứa một danh sách các khách hàng (Nếu bạn không phải là một doanh nghiệp, thì giả sử làm như vậy). Bạn cũng có thể sử dụng Table Wizard để xây dựng bảng, nhưng kết quả gần như không được thỏa mãn tới nhu cầu của bạn như tạo bảng bằng tay.

Để tạo một bảng, bạn cần trải qua các giai đoạn: thiết kế và sắp xếp bảng. Phần sau sẽ miêu tả chi tiết các giai đoạn đó.

## Giai đoạn thiết kế

Bạn ít khi sử dụng máy tính trong suốt quá trình thiết kế. Thay vào đó, chỉ cần một chiếc bút chì và tờ giấy, bạn quyết định thông tin nào có trong cơ sở dữ liệu. Một cách cụ thể, bạn quyết định dữ liệu nào cần để kiểm tra và các đặc điểm (kiểu dữ liệu, các định dạng, ...). Có rất nhiều sách tại các cửa hàng nói về khái niệm và cách thực hành của việc thiết kế cơ sở dữ liệu, tuy nhiên bạn không cần đọc chúng để thiết kế một cơ sở dữ liệu đơn giản. Tất cả những thứ cần là tự hỏi mình (và trả lời) những câu hỏi sau:

- Tôi nên gọi bảng là gì?
- Phần dữ liệu nào mà tôi cần để chứa?
- Kiểu dữ liệu nào là phù hợp nhất cho mỗi phần dữ liệu?
- Không gian lưu trữ bao nhiêu là phù hợp cho mỗi phần dữ liệu?
- Làm cách nào để sắp xếp các phần dữ liệu?
- Có cần một khóa duy nhất (khóa chính) cho dữ liệu?

Có rất nhiều câu hỏi khác bạn có thể tự đặt ra cho mình. Thực tế, bạn có thể tập hợp lại các thông tin chi tiết trước khi ngồi trước máy tính. Nhưng nếu bạn chỉ trả lời được ít trong số những câu hỏi trên, bạn hãy sử dụng cách tạo bảng thứ nhất.

## Mẫu thiết kế bảng

Như đã đề cập trong phần trước, cách tốt nhất để thiết kế bảng là bằng một chiếc bút chì và giấy. Với ý nghĩ đó, bạn có thể tạo *một mẫu thiết kế bảng* để sử dụng như khi làm việc trên bản thiết kế. Hình 1 miêu tả ví dụ như mẫu

Table Design

Table Name:  
Purpose:

Order	Field Name	Type of Data	Max Length

Hình 1: Mẫu thiết kế bảng

Đầu tiên bạn nên xác định tên cho bảng. Do bảng này chứa một danh sách các khách hàng doanh nghiệp, nên sử dụng Business Customers là tên bảng. Viết tên đó ở phía trên mẫu thiết kế.

## Lập kế hoạch các Table Fields

Trên trang tính của giấy, viết tên của các trường bạn thích để sử dụng trong bảng. Không cần chú ý đến thứ tự của chúng ở đó. Đối với các tên trường, bạn có thể gồm tên công ty, địa chỉ (có 2 dòng địa chỉ để dành cho địa chỉ dài và những bộ số), số điện thoại, số fax, tên giao tiếp, ...

Bảng 1 trình bày danh sách các tên trường mà bạn có thể sử dụng cho bảng Business Customers. Nếu chưa làm được, bạn nên nhập mỗi tên trường trong mẫu thiết kế bảng. Thông thường bạn nên nhập tên trường mà bạn đã viết. Tuy nhiên, do cơ sở dữ liệu của bạn ghép nối ví dụ được trình bày trong loạt bài này, sử dụng duy nhất tên trường được liệt kê trong bảng 1.

Tên trường	Ý nghĩa
Company Name	Tên công ty
Address 1	Dòng địa chỉ đầu tiên
Address 2	Dòng địa chỉ thứ hai
City	Tên Thành phố
State	Quận/Huyện
Zip Code	Mã bưu điện
Main Phone	Số điện thoại cho công ty tổng đài
Fax Number	Số Fax
Account Number	Mã số khách hàng
Salesperson	Người quản lý tài khoản
Credit Limit	Số lượng tài khoản có thể mua
Account Opened	Ngày tài khoản được thiết lập
Last Order	Ngày mua cuối cùng
Active	Chỉ rõ nếu đây là tài khoản còn hoạt động hay không hoạt động
Contact	Tên giao dịch của công ty
Contact Phone	Số điện thoại liên hệ
Comments	Những lưu ý cần thiết để ghi lại

Bảng 1: Các tên trường cho bảng Business Customers

## Chọn kiểu dữ liệu

Một kiểu dữ liệu của trường chỉ rõ kiểu giá trị mà trường đó giữ, như giá trị date, text hay currency (tiền tệ). Bước tiếp theo trong việc xác định bảng là quyết định kiểu dữ liệu cho trường. Bảng 2 liệt kê các kiểu dữ liệu trong Access

Kiểu dữ liệu	Ý nghĩa
AutoNumber Số tự động	Access sẽ tự động điền các số theo thứ tự mỗi khi thêm bản ghi.
Currency Kiểu tiền tệ	Số chính xác tới 15 số thập phân phía bên trái dấu thập phân và 4 số phía bên phải.
Date/Time Kiểu ngày tháng	Số tương ứng với ngày hoặc thời gian trong khoảng 100 đến 9999 năm.
Hyperlink	Một địa chỉ liên kết tới các tài liệu, URL hoặc các vị trí trong tài liệu
Memo Kiểu ghi nhớ	Chuỗi văn bản có độ dài lên tới 64000 ký tự
Number Kiểu số	Một giá trị số.
OLE Object	Các đối tượng như đoạn âm thanh hoặc hình ảnh
Text Kiểu văn bản	Gồm nhiều ký tự, độ dài tối đa 255 ký tự.
Yes/No Kiểu logic	Đúng hoặc sai.

Bảng 2: Các kiểu dữ liệu được hỗ trợ bởi Access

Hầu hết các trường trong bảng sẽ sử dụng kiểu dữ liệu là Text. Nhưng đối với một vài trường, bạn sẽ phải chọn kiểu dữ liệu thích hợp. Ví dụ, bạn nên sử dụng kiểu dữ liệu Yes/No cho trường Active, kiểu Date/Time cho trường Date Opened và Last Order, kiểu Memo cho trường Comments. Bảng 3 liệt kê các kiểu dữ liệu bạn nên sử dụng cho mỗi trường trong bảng. Bạn nên viết các kiểu dữ liệu trong cột Type of Data của mẫu thiết kế bảng.

Tên trường	Kiểu dữ liệu
Company Name	Text
Address 1	Text
Address 2	Text
City	Text
State	Text
Zip Code	Text
Main Phone	Text
Fax Number	Text

Account Number	AutoNumber
Salesperson	Text
Credit Limit	Currency
Account Opened	Date/Time
Last Order	Date/Time
Active	Yes/No
Contact	Text
Contact Phone	Text
Comments	Memo

**Bảng 3: Các kiểu dữ liệu cho các trường trong bảng Business Customers**

Chú ý rằng bảng 3 gán kiểu AutoNumber cho trường Account Number. Bởi vì Access tự động tính toán kiểu dữ liệu này với mỗi bản ghi mới nó tự gán mã khách hàng, bạn có thể bỏ qua việc tự động của Access để tự gán số tài khoản khách hàng.

### **Xác định độ dài của trường**

Ở bảng 2, một trường văn bản có thể lên tới 255 ký tự, một trường Memo (kiểu ghi nhớ) có độ dài tối đa 64.000 ký tự. Bước tiếp theo trong việc thiết kế bảng là xác định độ dài cho các trường. Đối với một vài trường, bạn không cần phải làm việc này. Ví dụ, không cần xác định độ dài cho trường Credit Limit cũng như Active bởi vì kiểu dữ liệu của chúng không phụ thuộc độ dài. Tuy nhiên, bạn phải xác định độ dài cho tất cả các trường khác.

Để xác định độ dài lớn nhất của trường một cách thích hợp, bạn cần biết rõ dữ liệu được chứa trong bảng. Ví dụ, bạn cần biết rằng sẽ cần duy trì mã bưu điện của khách hàng là 5 số hay 9 số. Tương tự như vậy, bạn nên biết cần bao nhiêu chỗ cho trường Company Name và Address.

Nếu bạn biết rõ dữ liệu của mình, có thể chọn độ dài phù hợp nhất cho trường. Mặc dù Access luôn cho phép bạn thay đổi độ dài trường, thậm chí sau khi bắt đầu nhập dữ liệu vào trong bảng, bạn nên xem khoảng dữ liệu để tránh làm hỏng và nhân bản làm việc sau này. Bảng 4 liệt kê độ dài tối đa của các trường trong bảng Business Customers.

<b>Tên trường</b>	<b>Độ dài tối đa</b>
Company Name	35
Address 1	40
Address 2	40
City	25
State	2
Zip Code	9
Main Phone	10
Fax Number	10
Account Number	không cần

Salesperson	35
Credit Limit	không cần
Account Opened	không cần
Last Order	không cần
Active	không cần
Contact	35
Contact Phone	10
Comments	không cần

Bảng 4: Độ dài các trường của bảng Business Customers

### Thứ tự các trường

Việc cuối cùng, bạn cần quyết định thứ tự trường xuất hiện trong bảng và chỉ rõ thứ tự trên mẫu thiết kế bảng.

Chú ý, ở đầu quá trình thiết kế bạn nên viết các trường theo thứ tự mà chúng xuất hiện trong đầu bạn. Bởi vì thứ tự các trường (giống như lựa chọn độ dài của trường) là một quá trình cơ bản và cần thiết cho việc tổ chức bảng của bạn, bình thường thì có thể đặt các trường theo thứ tự xuất hiện. Thứ tự trong cột của mẫu thiết kế bảng nên đặt là 1 cho trường đầu tiên, 2 cho trường thứ hai, ...

Tuy nhiên, mục đích của chúng tôi ở đây là sử dụng thứ tự được chỉ định trong bảng 5. Nhập thông tin này vào cột Thứ tự của mẫu thiết kế bảng.

Tên trường	Thứ tự
Company Name	2
Address 1	3
Address 2	4
City	5
State	6
Zip Code	7
Main Phone	8
Fax Number	9
Account Number	1
Salesperson	16
Credit Limit	13
Account Opened	12
Last Order	14
Active	15

Contact	10
Contact Phone	11
Comments	17

Bảng 5: Thứ tự các trường cho bảng Business Customers

Trước khi nhập thông tin ở mẫu thiết kế bảng vào Access, bạn nên quyết định sử dụng một khóa chính cho bảng. Ở chương 2 bạn đã học cách sử dụng khóa chính để sắp xếp các bản ghi và định vị chúng nhanh hơn. Khóa chính tiêu biểu là một giá trị duy nhất (nghĩa là không có bản ghi trùng trong bảng).

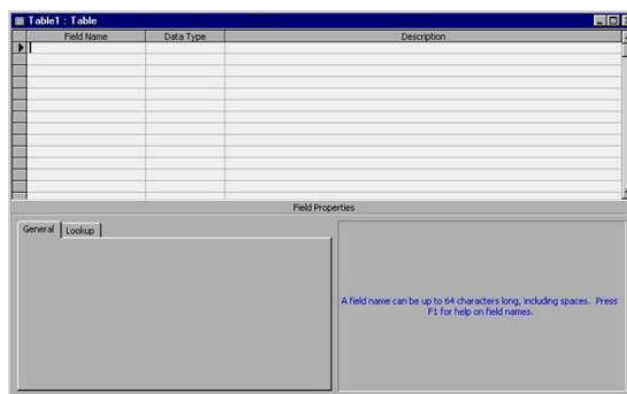
Trong bảng Business Customers, có nhiều trường bạn có thể sử dụng làm khóa. Ví dụ, có thể sử dụng trường Company Name làm khóa chính. Tuy nhiên, cũng có thể có 2 tên công ty trùng tên nhau, và bạn cần có khóa khác là duy nhất. Nếu bạn không thể tìm thấy khóa khác, thì có thể sử dụng trường Company Name và chỉ cho phép có hai tên giống nhau.

Bạn cũng nên sử dụng trường Account Number là khóa. Theo định nghĩa, số này đúng là duy nhất. Lưu ý, Access tự động tạo các mã cho bạn sử dụng trường AutoNumber. Bởi vì Account Number thường là duy nhất, và nó sẽ là một khóa chính hợp lý. Để nhắc nhở chính bạn rằng trường Account Number sẽ là khóa chính, khoanh tròn số trường (1) trong cột Thứ tự của mẫu thiết kế bảng.

### Giai đoạn trình bày

Sau khi hoàn thành việc thiết kế bảng, bạn đã sẵn sàng sử dụng Access. Với mẫu thiết kế bảng trong tay, bạn có thể nhanh chóng và dễ dàng sắp xếp cơ sở dữ liệu. Tất cả những việc cần làm là nhập thông tin theo thứ tự từ mẫu thiết kế bảng tới cửa sổ Design view. Để tạo bảng Business Customers, thực hiện các bước sau:

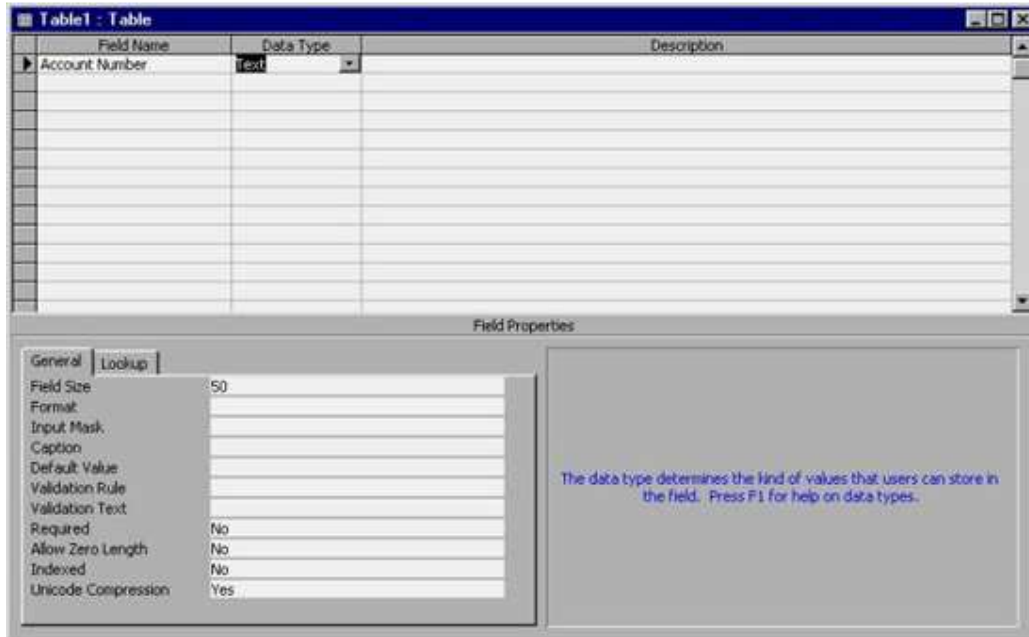
1. Nếu cần thiết, mở Access và vào cơ sở dữ liệu Customer. Đây là cơ sở dữ liệu được tạo ở đầu chương này.
2. Từ cửa sổ Database, chọn nút Tables. Bạn sẽ thấy chưa có bảng nào được liệt kê vì chúng chưa được tạo
3. Kích đúp chuột vào **Create Table in Design View**. Access hiển thị cửa sổ Design view như hình 2.



Hình 2: Cửa sổ Design view.

Ở chương 3, bạn đã học cách sử dụng cửa sổ Design view để thay đổi cách sắp xếp bảng hiện tại. Bây giờ bạn sử dụng nó để tạo một bảng mới. Trong trường hợp này, cửa sổ Design view là hoàn toàn trống. Chú ý là con trỏ nhấp nháy ở dòng đầu tiên trong cột Field Name. Access sẵn sàng cho bạn bố trí bảng.

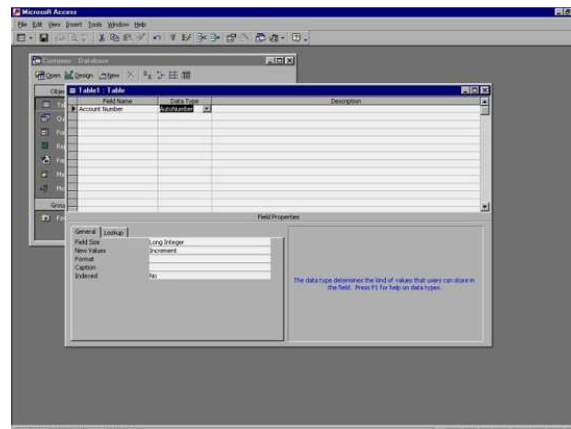
Để bắt đầu, nhập tên trường đầu tiên trong thứ tự bảng: trường Account Number, Nhập **Account Number**, sau đó ấn phím **Enter**. Sau khi Access chấp nhận tên trường, con trỏ di chuyển tới cột Data Type. Cửa sổ Design view bây giờ trông như hình 3. Chú ý rằng Access giả thiết trường này là một trường văn bản và mặc định những thuộc tính cho kiểu dữ liệu Text xuất hiện phía dưới cửa sổ Design view.



Hình 3: Cửa sổ Design view sau khi nhập tên trường đầu tiên.

Trong mẫu thiết kế bảng, bạn chỉ định kiểu dữ liệu trường Account Number là AutoNumber. Để thay đổi kiểu dữ liệu của trường, kích chuột vào mũi tên phía bên phải của ô Data Type. Lần lượt, Access hiển thị một danh sách xổ xuống của kiểu dữ liệu. Kích chuột chọn AutoNumber.

Ô sẽ hiển thị ngay kiểu dữ liệu AutoNumber cho trường Account Number. Cửa sổ Design view giống ở hình 4. Đồng thời, cùng với việc thay đổi kiểu dữ liệu, Access đã thay đổi các thuộc tính của trường.

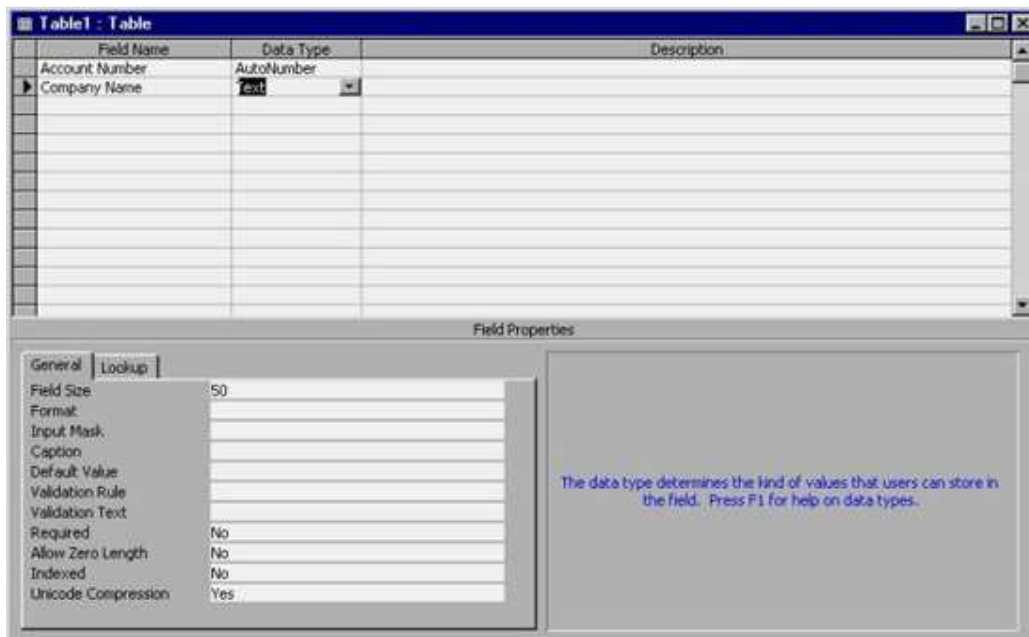


Hình 4: Cửa sổ Design view sau khi thay đổi kiểu dữ liệu

Khi bạn ấn phím ENTER, con trỏ di chuyển tới cột Description. Bạn có thể nhập những lời chú thích cho trường đó. Đối với trường Account Number, có thể gồm một lời chú thích để nhắc nhở bạn hoặc người dùng khác rằng trường này là khóa chính.

Khi ấn phím ENTER một lần nữa, Access chuyển tới dòng tiếp theo trong cột Field Name. Nhập tên trường thứ hai (Company Name) và lựa chọn kiểu dữ liệu. Thực hiện công việc này cho đến khi bạn nhập các thông tin cho trường cuối cùng.

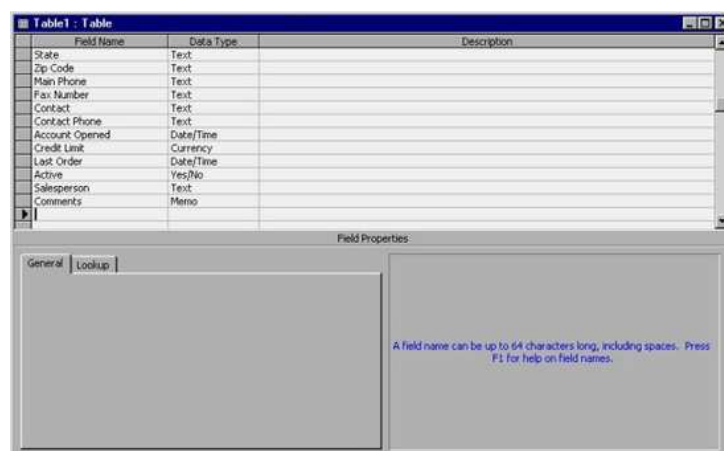
Trước khi chuyển khỏi dòng chứa trường Company Name, hãy nhìn xuống phần các thuộc tính phía dưới cửa sổ Design view (hình 5). Những thuộc tính đó mặc định cho kiểu dữ liệu Text. Chú ý, Access thiết lập thuộc tính Field Size là 50.



Hình 5: Cửa sổ Design view sau khi xác định kiểu dữ liệu cho trường thứ hai.

Khi bạn xác định mẫu thiết kế bảng mà độ dài chỉ cần 35, hãy điều chỉnh thuộc tính Filed Size. Sử dụng chuột để lựa chọn thuộc tính Field Size và thay đổi thành 35. Sau đó dùng chuột trở lại cột Field Name, trong dòng thứ ba nhập tiếp tên trường tiếp theo.

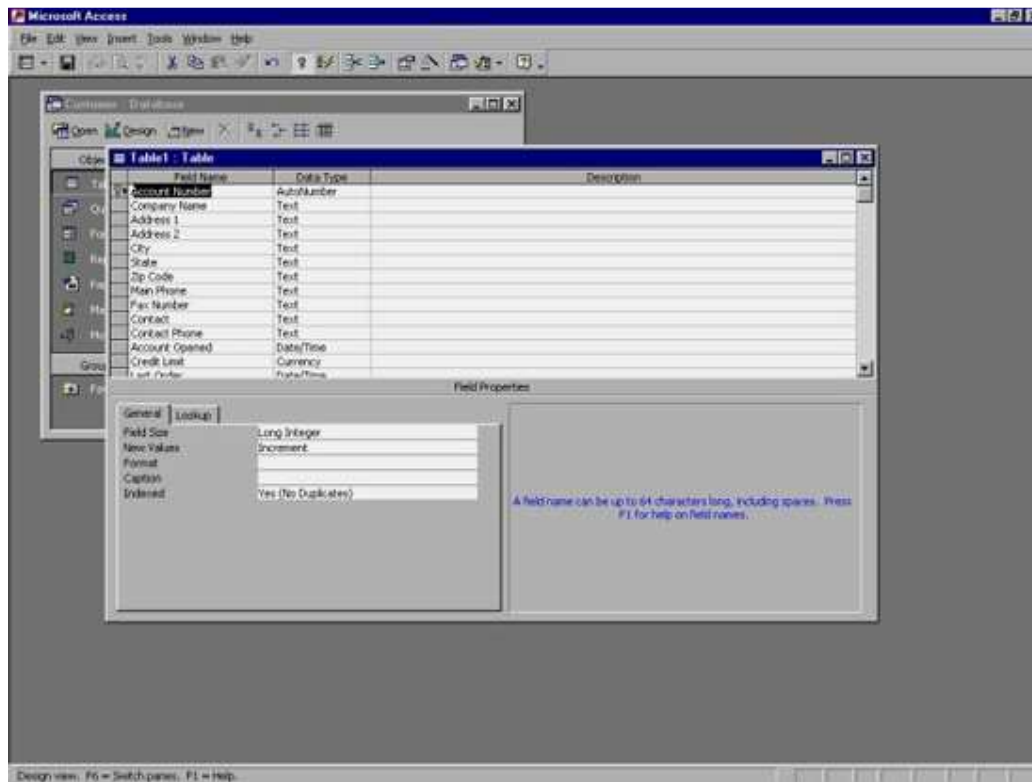
Tiếp tục và nhập tất cả 17 trường cho bảng Business Customers. Khi nhập mỗi trường vào trong bảng, phải chắc là bạn thiết lập thuộc tính DataType và thay đổi Field Size cho hợp lý. Khi đã hoàn thành, cửa sổ Design view xuất hiện như hình 6.



Hình 6: Cửa sổ Design view sau khi nhập trường cuối cùng.

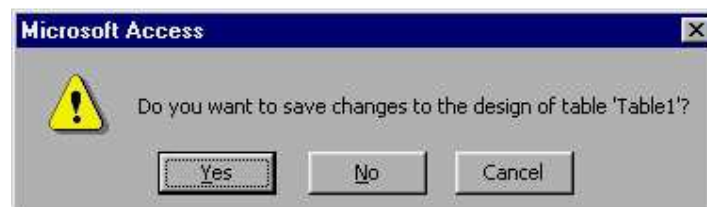


Sau khi hoàn thành tất cả các trường, bạn phải xác định khóa chính. Để làm được điều này, cuộn thiết kế bảng lên trên cho đến khi bạn nhìn thấy trường Account Number trên màn hình. Lựa chọn trường và kích chuột vào công cụ Set Primary Key. Access hiển thị một khóa trong vùng xám phía bên trái dòng Account Number (hình 7).



Hình 7: Cửa sổ Design view sau khi đã hoàn tất.

Bây giờ bạn đã hoàn thành toàn bộ bảng, và muốn lưu nó lại. Để lưu bảng, cách đơn giản là đóng cửa sổ Design view bằng cách kích chuột vào nút Close phía trên góc bên phải cửa sổ. Trước khi đóng, Access hỏi nếu bạn muốn lưu bảng này (hình 8).



Hình 8: Access hỏi xem bạn có muốn lưu bảng không

Để lưu, kích chuột vào nút Yes. Access sẽ lại hỏi bạn xác định tên cho bảng, hình 9.



Hình 9: Đặt tên cho bảng sẽ được lưu.

Nhập tên bạn đã viết phía trên mẫu thiết kế: Business Customers. Nhấn ENTER hoặc kích nút OK để lưu bảng dưới tên đó.

# Bài 16: Kiểm tra cách trình bày của bạn

Sau khi đã định rõ được cách bố trí bảng, bạn cần kiểm tra và cải tiến bảng. Bạn có thể làm điều này một cách dễ dàng bằng cách sử dụng chế độ Datasheet cho bảng. Hiển thị chế độ Datasheet cho bảng Business (nếu bạn cần giúp đỡ, trở lại chương 2). Bạn sẽ thấy một Datasheet trông như hình 1.



Hình 1: Chế độ Datasheet cho bảng Business Customers.

Phải mất một lúc để chờ Datasheet xuất hiện. Nếu các tên trường không xuất hiện đúng như bạn đã đặt, hãy chuyển tới cửa sổ Design view và thực hiện thay đổi tới những thuộc tính Caption của các trường có nhãn không thỏa mãn. (Bạn có thể nhớ lại từ chương 3 rằng Access hiển thị thuộc tính Caption ở phía đầu mỗi cột Datasheet). Để chuyển giữa chế độ Datasheet và Design view, sử dụng biểu tượng ngoài cùng bên trái trên thanh công cụ.

Đó là cách cải tiến lại bảng, bạn nhập thông tin vào chế độ Datasheet. Sau đó, chuyển tới cửa sổ Design view để thực hiện thay đổi. Khi thay đổi hoàn thành, trở lại chế độ Datasheet để tiếp tục nhập thông tin. Tiến trình lặp đi lặp lại giúp bạn nhanh chóng thực hiện bảng đạt yêu cầu tốt nhất.

Với bảng thiết kế đã hoàn thành, nhập một bản ghi. Với bản ghi đầu tiên, sử dụng thông tin trong bảng 1.

Tên trường	Giá trị
Account Number	Access tự động điền
Company Name	Birch Enterprises
Address 1	3456 Maple Lane
Address 2	PO Box 549
City	Kansas City
State	MO
Zip Code	64133
Main Phone	8162375632
Fax Number	8162375633

Contact	Ed Birch
Contact Phone	
Account Opened	3/2/99
Credit Limit	1500
Last Order	7/8/99
Active	Yes (select the check box)
Salesperson	John Davis
Comments	

Bảng 1: Thông tin cho bản ghi đầu tiên.

Khi nhập thông tin, bạn có thể đưa ra một cách để làm dữ liệu nhập dễ dàng hơn. Ví dụ, nhập số điện thoại sẽ dễ hơn nếu bạn thiết lập thuộc tính Input Mask cho trường số điện thoại.

Trở lại cửa sổ Design view và chọn trường số điện thoại đầu tiên (Main Phone). Thiết lập thuộc tính Input Mask (mặt nạ nhập liệu) cho trường này theo các bước sau:

1. Kích chuột vào hộp **Input Mask** ở phía dưới cửa sổ Design view. Bạn sẽ thấy biểu tượng **Builder** (một nút có 3 dấu chấm) xuất hiện phía bên phải hộp.
2. Kích chuột vào biểu tượng **Builder**. Bạn thấy hộp thoại *Input Mask Wizard* xuất hiện. Lựa chọn ở mục **Input Mask** là Phone Number (hình 2)



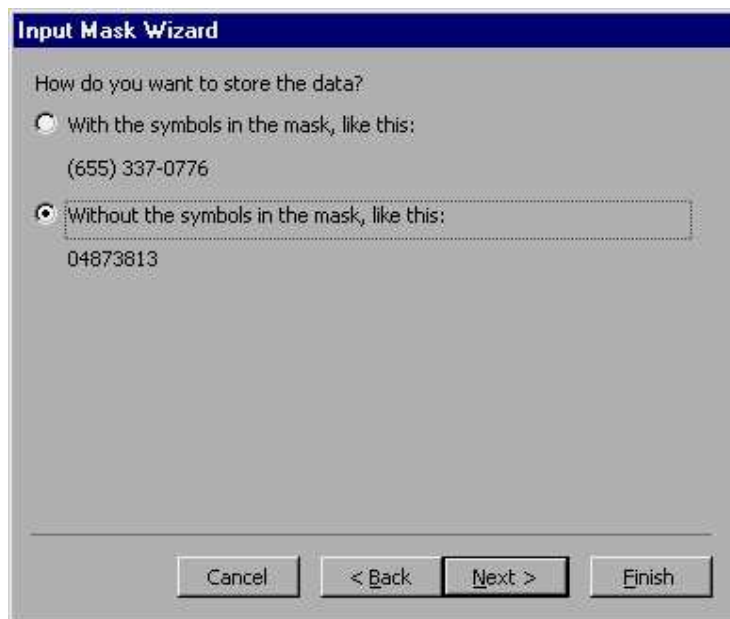
Hình 2: Hộp thoại Input Mask Wizard

Nếu muốn sử dụng mặt nạ nhập liệu cho các số điện thoại, kích chuột vào nút **Next**. Hộp thoại tiếp tục hiển thị như hình 3.



Hình 3: Bạn có thể thay đổi vị trí các ký tự ở hộp thoại này

1. Nếu bạn muốn, hãy thay đổi dấu ngăn cách mà input mask sẽ sử dụng (thường mặc định là dấu gạch chân).
2. Kích chuột vào nút **Next**. Hộp thoại hiển thị như hình 4.



Hình 4: Bạn có thể thay đổi cách Access lưu giữ thông tin ở hộp thoại này.

1. Chỉ rõ cách bạn muốn Access giữ thông tin đã được lưu thông qua mặt nạ nhập liệu. Có thể lưu số điện thoại bằng việc định dạng các ký tự hoặc không (mặc định). Độ dài trường cho số điện thoại thường là 10 ký tự, bạn nên lưu giá trị không dùng định dạng các ký tự.
2. Kích chuột vào nút **Next**. Access hiển thị hộp thoại cuối cùng như hình



Hình 5: Hộp thoại cuối cùng của Input Mask Wizard.

Kích chuột vào nút **Finish**. Sau đó, Access sẽ cập nhật thuộc tính Input Mask trong cửa sổ Design view và nó xuất hiện như sau:

**!(999)000-0000;;\_**

Bạn có thể lặp lại từ bước 1 tới bước 8 cho mỗi trường điện thoại khác nhau. Khi thoát khỏi thuộc tính Input Mask, chú ý rằng Access định dạng lại dấu như sau:

**!\(999")"000\ -0000;;\_**

Access thêm các ký tự bởi vậy mà mặt nạ thích hợp với những đặc điểm bên trong nó.

Khi đã hoàn thành mặt nạ nhập liệu, kích chuột vào công cụ Datasheet view. Đáp lại, Access hỏi bạn nếu muốn lưu sự thay đổi để sắp xếp bảng (hình 6).



Hình 6: Access hỏi bạn nếu muốn lưu sự thay đổi.

Kích chuột vào **Yes** để lưu và trở lại chế độ Datasheet. Bạn có thể nhập bản ghi khác ngay để thấy được kết quả.

## Bài 17: Tóm tắt chương 4

Mặc dù Access cung cấp nhiều công cụ khác nhau (như Wizards) nhưng bạn không cần sử dụng những công cụ này để tạo một cơ sở dữ liệu. Thay vào đó, bạn có thể tạo bằng tay và cải tạo bảng để làm việc tốt hơn. Khi làm việc với Access theo cách này, bạn sẽ có

hiều kiến thức hơn về các tính năng của nó và cách sử dụng chúng.

Trong chương này, bạn đã khai thác bằng tay và cải tiến một cơ sở dữ liệu. Trước khi chuyển sang chương 5, phải chắc chắn bạn hiểu được các nội dung sau:

- Thiết kế một bảng phụ thuộc vào kế hoạch của bạn. Cần làm như vậy để xem xét dữ liệu nào mà bạn quản lý, cũng như các đặc điểm của dữ liệu.
- Sử dụng thiết kế bảng theo mẫu để thiết kế bảng dễ hơn.
- Khi tạo bảng bằng tay, sử dụng cửa sổ Design view để chỉ rõ cách sắp xếp bảng.
- Khi nhập các trường vào cửa sổ Design view, bạn cần thay đổi kiểu dữ liệu và thuộc tính mà Access định vị trước.
- Quá trình cải tiến việc sắp xếp bảng là sự lặp đi lặp lại theo tự nhiên. Nó đòi hỏi làm việc luân phiên ở cả chế độ Datasheet và Design view, lặp đi lặp lại việc chuyển đổi từ một cửa sổ tới cửa sổ khác cho đến khi việc thiết kế hoàn thành.
- Khi làm việc trong chế độ Datasheet, bạn nên tìm nhiều cách để hoàn thiện việc sắp xếp bảng. Chuyển đổi tới cửa sổ Design view để thực hiện thay đổi và trở lại chế độ Datasheet để kiểm tra những thay đổi đó.

**Để bạn nhanh chóng áp dụng thông tin trong bảng và hiển thị nó theo cách có tổ chức hơn, Access cung cấp hai công cụ đặc biệt: sắp xếp và lọc.** Sử dụng những công cụ này, bạn có thể làm việc với thông tin trong mọi cách. Chương này sẽ giải thích cách thực hiện sắp xếp và lọc, cũng như cách sử dụng những công cụ này để làm việc với dữ liệu.

Kết thúc chương này, bạn sẽ hiểu các nội dung sau:

- Sắp xếp là gì?
- Làm thế nào để thực hiện một sắp xếp đơn giản
- Thế nào là một sắp xếp đơn giản?
- Cách để xây dựng một sắp xếp phức tạp.
- Cách loại bỏ các hiệu ứng của một sắp xếp.
- Một bộ lọc là gì?
- Làm cách nào để lọc dữ liệu.
- Cách lưu một bộ lọc.

## **Bài 18: Sắp xếp dữ liệu**

Sắp xếp là quá trình đặt dữ liệu theo đúng thứ tự logic. Chúng ta từng sắp xếp mọi việc hàng ngày. Ví dụ, bạn sắp xếp những hóa đơn riêng lẻ theo ngày tháng do đó có thể quản lý những việc quan trọng trước tiên. Bạn sắp xếp những cuộn băng video theo thứ tự chữ cái hoặc tập hợp các bài hát theo tên nghệ sĩ.

Với Access, bạn có thể thực hiện sắp xếp theo cách đơn giản và phức tạp mà hai phần tiếp theo sẽ đi vào chi tiết các cách sắp xếp đó.

### **Sắp xếp đơn giản**

Trong Access, bạn sẽ sắp xếp các bản ghi trong một bảng dựa vào một hoặc nhiều trường. Ví dụ, bạn phải sắp xếp các bản ghi theo tên họ hoặc tên riêng sao cho tất cả những người bạn của

bạn có tên họ là Smith đã được sắp xếp theo tên riêng.

Sắp xếp các bản ghi theo thứ tự dựa vào nội dung của một hoặc nhiều trường liên nhau. Khi thực hiện một sắp xếp đơn giản, bạn phải định rõ thứ tự muốn Access sắp xếp cho bản ghi:

- **Sắp xếp theo thứ tự tăng dần:** Có nghĩa là Access sẽ sắp đặt các bản ghi theo thứ tự từ thấp đến cao. Theo cách đó, Access sắp đặt các ký tự chữ từ A đến Z và sắp xếp các thông tin theo thứ tự tăng dần của số.

- **Sắp xếp theo thứ tự giảm dần:** Cách này đối lập với sắp xếp theo thứ tự tăng dần, và nó còn được gọi là đảo ngược thứ tự. Access sắp xếp thứ tự từ cao xuống thấp: thông tin theo chữ cái từ Z tới A, theo số từ cao nhất tới thấp nhất.

Ví dụ, giả sử bạn có một bảng về các băng đĩa yêu thích. Nếu muốn thứ tự các bản ghi dựa theo tiêu đề các băng, bạn yêu cầu Access thực hiện một sắp xếp đơn giản. Thực hiện một sắp xếp đơn giản căn cứ vào một trường đơn lẻ, lựa chọn một trường (ô) trong cột mà bạn muốn sắp xếp. Tiếp theo, sử dụng các hành động sau, dựa vào cách bạn muốn sắp xếp như thế nào:

- Để sắp xếp theo thứ tự tăng dần, kích chuột vào công cụ Sort Ascending (A-Z) như hình 1. Bạn có thể chọn tùy chọn Sort từ menu Records và chọn Sort Ascending.

- Để sắp xếp theo thứ tự giảm dần, kích chuột vào công cụ Sort Descending (Z-A) như hình 1. Bạn có thể chọn tùy chọn Sort từ menu Records và chọn Descending Sort.



Hình 1: Công cụ sắp xếp

Khi thực hiện sắp xếp, Access sẽ tổ chức lại bảng sao cho các bản ghi được sắp xếp theo thứ tự mà bạn muốn.

**Chú ý:** Dựa vào số bản ghi có trong bảng, có thể thực hiện sắp xếp theo dữ liệu. Độ dài cần thiết cũng dựa vào tốc độ của máy tính và phần cứng.

Bạn có thể sử dụng tính năng sắp xếp đơn giản để sắp xếp nhiều hơn một trường đơn lẻ. Tuy nhiên, những cột đó phải cạnh nhau, và mỗi trường phải được sắp xếp theo cùng mức (tăng dần hoặc giảm dần). Trong trường hợp khác, bạn không thể sắp xếp một cột theo thứ tự tăng dần và cột khác theo thứ tự giảm dần.

Nếu muốn sắp xếp một cột theo một thứ tự và cột khác theo thứ tự khác, bạn cần sử dụng sắp xếp phức tạp, được đề cập sau chương này. Cũng như nếu cột không sát nhau, bạn phải sử dụng sắp xếp phức tạp.

Như ví dụ sử dụng đặc tính sắp xếp đơn giản với các cột liên nhau, giả sử bạn đã gán các số cho mọi băng đĩa trong tập hợp các video và mỗi bản ghi tương ứng với một hiển thị đơn. (Nhớ rằng, Access có thể sử dụng cùng số băng đĩa đó nhiều hơn một bản ghi. Việc này xảy ra khi một băng video chứa nhiều hơn một bản ghi hiển thị). Bạn muốn sắp xếp bảng theo số băng đĩa và sau đó hiển thị tên như thế nào? Nếu các cột chứa hai trường liên nhau, bạn nên thực hiện các bước sau:

Di chuyển con trỏ chuột sao cho nó ở phía trên cột tên (phía trên chế độ Datasheet) của một

trong những cột mà bạn muốn sắp xếp. Khi làm việc này, con trỏ thay đổi thành mũi tên chéo xuống.

1. Kích chuột trái để đánh dấu cột.
2. Giữ phím **SHIFT** và lặp lại bước 1, 2 với các cột khác. Access sẽ hiển thị tất cả các cột giữa cột đầu tiên và cột thứ hai mà bạn đã chọn.
3. Sử dụng một trong những cách sắp xếp (tăng dần hoặc giảm dần) được miêu tả trong ví dụ trước.

Khi bạn thực hiện một sắp xếp đơn giản sử dụng nhiều cột, Access làm việc từ trái sang phải thông qua bảng. Do vậy, Access sắp xếp cột ngoài cùng bên trái đầu tiên, chuyển dần về phía bên phải và sắp xếp cột tiếp theo,...

### **Di chuyển các cột**

Trước khi thực hiện một sắp xếp đơn giản trên nhiều trường (cột), bạn cần di chuyển các cột sao cho chúng liền nhau. Khi làm việc ở chế độ Datasheet, bạn có thể sử dụng chuột để di chuyển các cột trong bảng. Để làm như vậy, thực hiện các bước sau:

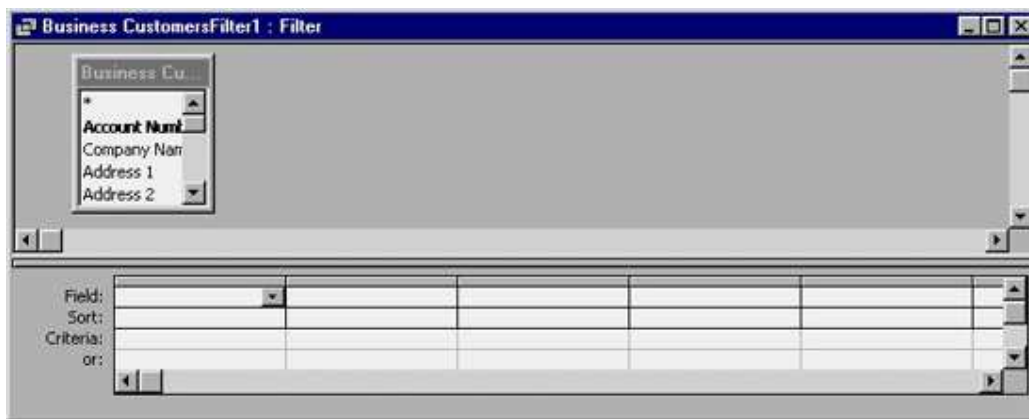
1. Lựa chọn cột bạn muốn di chuyển và để con trỏ chuột lên cột đó
2. Giữ chuột trái. Một hộp nhỏ sẽ xuất hiện xung quanh cuối con trỏ chuột.
3. Kéo cột tới vị trí mới trong bảng. Khi di chuyển chuột sang trái hoặc phải, Access hiển thị dòng giữa các cột để cho biết nơi mà cột đã di chuyển.
4. Khi con trỏ chuột ở phía trên vị trí mà bạn muốn, nhả chuột. Lần lượt như vậy, Access sẽ di chuyển cột tới vị trí mới.

### **Sắp xếp phức tạp**

Ở phần trước, bạn đã học về sắp xếp đơn giản. Sắp xếp phức tạp là bước xa hơn. Bạn có thể sử dụng một sắp xếp phức tạp để sắp xếp nhiều cột, các cột không liền nhau theo thứ tự khác nhau (tăng dần và giảm dần). Ví dụ, nếu bạn có một bảng tên và địa chỉ, có thể sắp xếp bản ghi đầu tiên theo ZIP code (mã bưu điện) và theo tên họ (theo ZIP code). Nếu hai cột không liền nhau (cột Zip code phía bên trái của cột tên cuối cùng), bạn cần sử dụng sắp xếp phức tạp.

Để hoàn thành xong sắp xếp phức tạp, bạn sử dụng cửa sổ Filter. (Bộ lọc được miêu tả chi tiết sau chương này). Để hiển thị cửa sổ, lựa chọn Filter từ thực đơn Records, sau đó chọn Advanced Filter/Sort từ kết quả của trình đơn phụ. Access hiển thị cửa sổ Filter như hình 2.





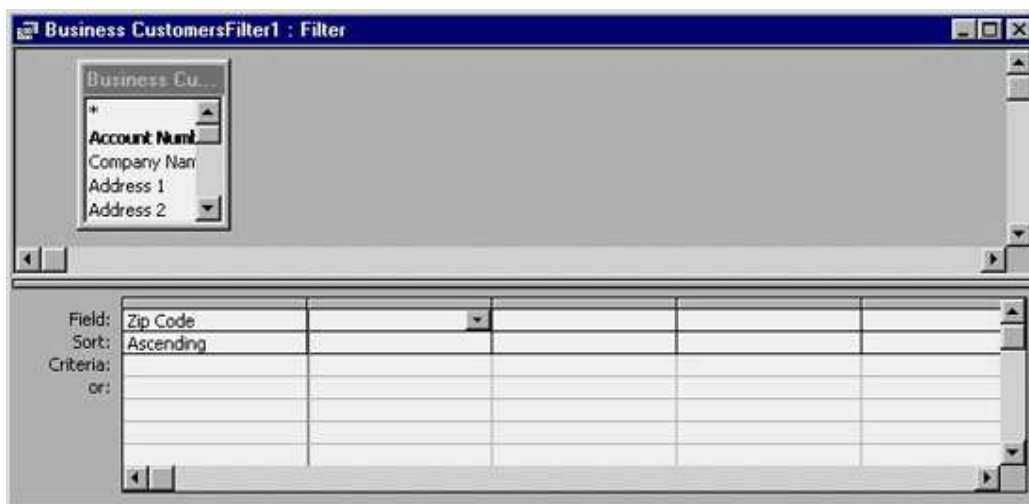
Hình 2: Cửa sổ Filter

Để thực hiện một sắp xếp phức tạp, cần chọn những trường mà bạn muốn để sắp xếp, sau đó chỉ định cách sắp xếp chúng. Ví dụ, giả sử bạn đang làm việc với bảng Susiness Customers (được phát triển trong chương 4, "Tạo cơ sở dữ liệu khác"), và bạn muốn sắp xếp nó trước hết theo Zip Code và sau đó là tên công ty.

Để thực hiện điều này, bạn cần chỉ rõ những trường và thứ tự sắp xếp của chúng trong khung ở phía dưới cửa sổ Filter. Trong cột đầu tiên của cửa sổ Filter, kích chuột vào mũi tên phía bên phải hộp field và chọn trường Zip Code.

Tiếp đó, bạn cần cho biết trường Zip Code được sắp xếp theo thứ tự tăng dần. Để làm được, kích chuột vào dòng Sort ngay phía dưới nơi bạn đặt trường ZipCode, sau đó kích vào mũi tên phía bên phải của ô. Lần lượt như vậy, Access cung cấp một danh sách xổ xuống và bạn có thể chọn Ascending cho thứ tự sắp xếp.

Sau khi chọn thứ tự sắp xếp, cửa sổ Filter sẽ hiển thị như hình 3.

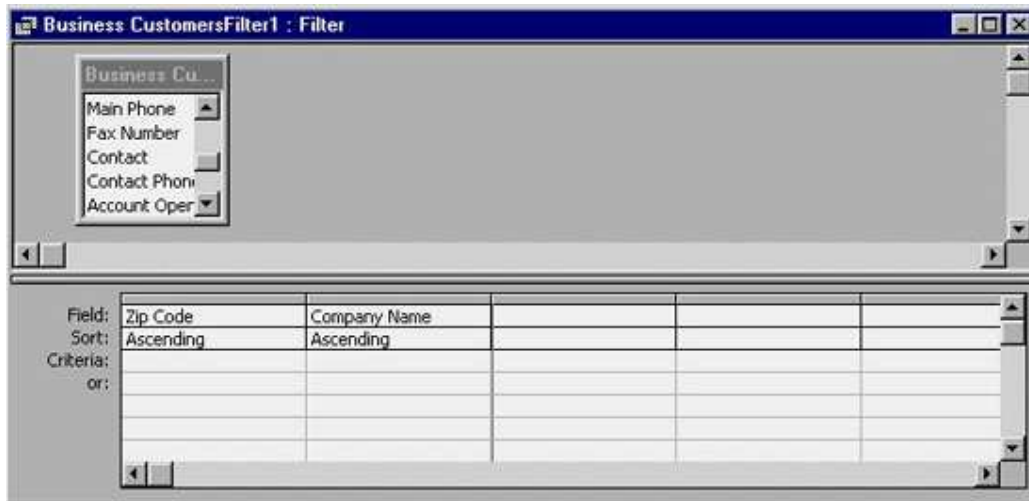


Hình 3: Cửa sổ Filter sau khi chỉ định trường đầu tiên để sắp xếp

Trong cột thứ hai, bạn có thể chỉ định trường thứ hai bạn muốn sắp xếp (Company Name) và cách bạn sắp xếp nó. Nếu sắp xếp bảng theo cách khác, bạn có thể tiếp tục lựa chọn các trường khác và chỉ rõ cách sắp xếp chúng. Khi đã làm xong, bạn cần áp dụng cách sắp xếp cho bảng. Để áp dụng, lựa chọn Apply Filter/Sort từ thực đơn Filter, hoặc kích chuột vào công cụ Apply Filter. Access sẽ hiển thị các bản ghi trong bảng theo thứ tự định sẵn.

Trong cửa sổ Filter, có nhiều cách khác nhau mà bạn có thể thêm các trường để sắp xếp, khác

với các phương thức bạn đã sử dụng (kích chuột vào thanh cuộn của danh sách trường). Ví dụ, nếu bạn kích đúp chuột vào một tên trường trong danh sách phía trên cửa sổ Filter, tên trường sẽ xuất hiện ở cột trống đầu tiên trong dòng Field. Hình 4, nếu bạn kích đúp chuột vào tên trường Main Phone, Access sẽ đặt tên trường vào cột chưa sử dụng.



Hình 4: Bạn có thể kích đúp chuột để lựa chọn trường

Bạn cũng có thể sử dụng chuột để kéo tên trường từ danh sách tới vùng dưới khung. Access cung cấp ba hoặc bốn cách khác để đưa trường bạn muốn sắp xếp. Nếu sau đó bạn muốn xóa một trường, đơn giản là sử dụng chuột để chọn cột chứa trường (trong vùng khung) và ấn phím DEL hoặc chọn Delete Column từ thực đơn Edit.

**Trong phần trước, bạn đã học cách sử dụng cửa sổ Filter để thực hiện sắp xếp phức tạp. Bạn có thể sử dụng cửa sổ Filter để lọc dữ liệu. Một filter cho phép bạn giới hạn các bản ghi để xem.** Việc lọc dữ liệu không làm mất các bản ghi, nó chỉ cho phép xác định những bản ghi nào bạn muốn Access hiển thị.

Khi xác định một filter, bạn chỉ định tiêu chuẩn mà dữ liệu cần có trước khi Access hiển thị nó. Ví dụ, một filter liệt kê các bản ghi về công nhân sống ở New York.

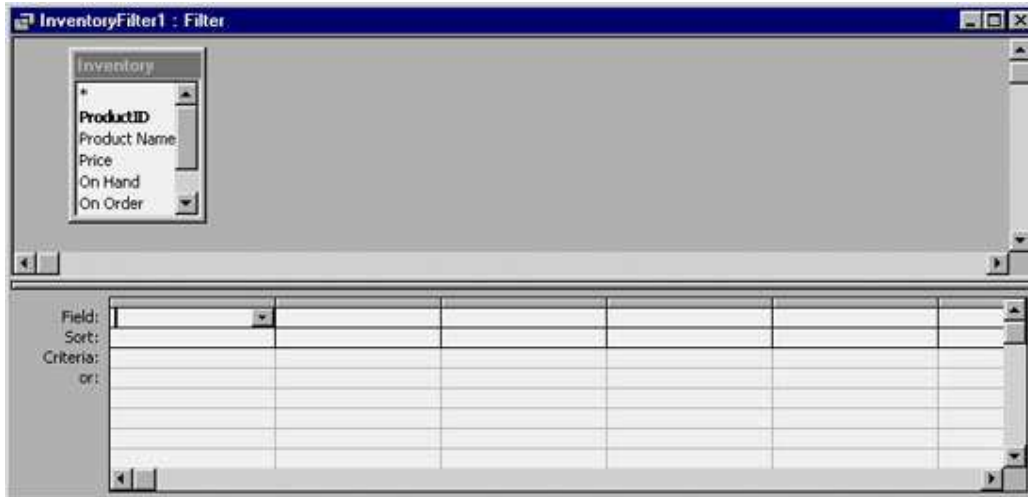
Access chỉ hiển thị những bản ghi phù hợp với tiêu chuẩn filter đưa ra. Thực tế, khi tạo một tập hợp con dữ liệu, bạn có thể làm việc với tập hợp dữ liệu đó như thể nó là một bảng: bạn có thể thêm, xóa các bản ghi và thực hiện các thay đổi như mong muốn. Access sẽ cập nhật bảng gốc để cho thấy mỗi sự thay đổi mà bạn đã thực hiện.

Sử dụng filter để quản lý dữ liệu dễ dàng hơn. Ví dụ, giả sử bạn đang làm việc với một danh sách các khách hàng, và cần thay thế một trong số những người đại diện bán hàng. Lúc này bạn phải cập nhật bảng khách hàng để có một người đại diện mới.

Bạn có thể sử dụng chức năng Replace (được miêu tả ở chương 2, "Học Access trong 1 giờ"), nhưng thay vào đó bạn muốn tạo một filter để có thể xem lại các khách hàng trước khi thật sự muốn thay thế họ. Trong trường hợp này, filter giới hạn các bản ghi mà Access chỉ hiển thị gồm đại diện người bán hàng cũ. Trong hầu hết các trường hợp, đó là cách dễ dàng để làm việc với bảng nhỏ hơn là với toàn bộ bảng khách hàng.

## Tạo một Filter

Để tạo một filter, bạn cần làm việc với cửa sổ Filter. Kích chuột vào tùy chọn **Filter** từ thực đơn **Records**, sau đó chọn **Advanced Filter/Sort** từ trình đơn phụ. Access hiển thị cửa sổ Filter như hình 1.

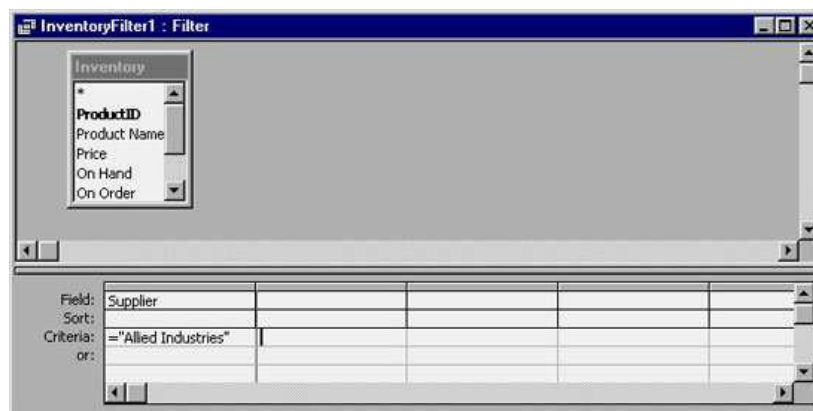


Hình 1: Cửa sổ Filter

Để tạo một filter, tất cả những gì cần làm là đặt một trường và chỉ rõ tiêu chuẩn bạn muốn áp dụng cho trường. Nếu muốn, bạn cũng có thể xác định cách sắp xếp cho trường. (Theo cách đó, bạn vừa sắp xếp và lọc dữ liệu cùng một lúc.)

Ví dụ, giả sử bạn có một bảng chứa các bản ghi về bản kiểm kê (bảng Inventory) được hiển thị trong hình 1. Ở bảng này, một trong các trường được đặt tên là **Supplier**. Bạn sử dụng trường này để theo dõi người cung cấp một mục đặc biệt. Giả sử bạn cần xem mục nào được cung cấp bởi **Allied Industries**. Sử dụng filter, bạn có thể giới hạn các bản ghi mà Access hiển thị tới những bản ghi mà nhà cung cấp là **Allied industries**. Để tạo một filter, bạn thực hiện theo những bước sau:

1. Chọn **Filter** từ thực đơn **Records**, sau đó chọn **Advanced Filter/Sort** từ trình đơn phụ.
2. Trong cột đầu tiên của khung phía dưới cửa sổ, chọn trường **Supplier**.
3. Trong ô **Criteria** của cột, gõ dấu **=**, sau đó là cụm từ **Allied Industries**. Khi nhấn phím ENTER, cửa sổ Filter xuất hiện như hình 2. (Chú ý rằng Access tự động thêm dấu trích dẫn quanh tên nhà cung cấp)



Hình 2: Tạo một filter đơn giản

4. Kích chuột vào **Apply Filter** trên thanh công cụ.

Access áp dụng các quy tắc lựa chọn (được chỉ rõ bằng filter) và hiển thị những bản ghi về Allied Industries. Kết quả đưa ra tập hợp con của bảng kiểm kê sẽ dễ dàng để làm việc hơn là toàn bộ bảng. Trong trường hợp này, filter của bạn chỉ dẫn Access xem xét mỗi bản ghi trong bảng và chỉ hiển thị các bản ghi về trường Supplier có chứa tên Allied Industries.

*Chú ý: Các filter bạn tạo có thể trở nên đơn giản hoặc phức tạp. Bạn chỉ định tiêu chuẩn cho bộ lọc đang sử dụng giống như bạn sử dụng để tạo quy tắc hiệu lực cho các trường (Những Validation rules đã được đề cập ở chương 3, "Tùy biến các thành phần"). Nếu cần trợ giúp để nhớ những biểu thức mà bạn có thể sử dụng, trở lại phần validation rules.*

## Loại bỏ một Filter

Thật không lấy làm lạ khi sử dụng một filter để nhanh chóng xem dữ liệu theo cách đặc biệt, và sau đó muốn loại bỏ bộ lọc để xem tất cả các bản ghi. Access thực hiện việc này rất dễ dàng và bạn có thể hoàn thành thao tác này sử dụng một trong hai cách thức khác nhau.

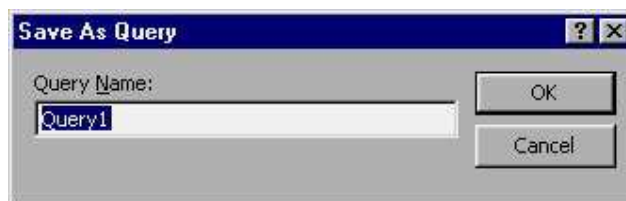
Cách thức thứ nhất (có lẽ là dễ nhất) là kéo thanh công cụ. Khi một filter được hoạt động, trên thanh công cụ có biểu tượng giống như hình cái phễu. Thực tế, có 3 công cụ chứa hình phễu, nhưng chỉ có một công cụ chứa hình phễu mà không có bất kỳ biểu tượng thêm nào khác và nó đang được chọn. Nếu bạn đưa con trỏ chuột lên công cụ đó, bạn sẽ thấy tên của nó là Remove Filter. Kích vào **Remove Filter**, khi đó tất cả các bản ghi sẽ xuất hiện trở lại, filter được gỡ bỏ hoàn toàn.

Phương thức khác của việc gỡ bỏ một filter là sử dụng các thực đơn. Trên thực đơn **Record** là tùy chọn **Remove Filter/Sort**. Nếu bạn chọn tùy chọn này, tất cả các bản ghi được hiển thị trở lại.

## Lưu một filter

Access chỉ giữ lại filter cho đến khi bạn đóng bảng lại. Do vậy, nếu đóng bảng và sau đó quyết định sử dụng filter lại, bạn sẽ cần tạo lại một filter từ scratch. Cách đó có thể gây chán và tốn thời gian.

Để giải quyết vấn đề này, bạn có thể lưu filter như một query (truy vấn). Bạn sẽ học tất cả các truy vấn trong chương 6 "*Sử dụng các truy vấn để xem xét dữ liệu*". Các Filter có liên quan tới truy vấn, mặc dù chúng không có nhiều tính năng hay sự linh hoạt. Khi các filter và truy vấn liên quan, Access cho phép bạn lưu giữ filter dưới tên một truy vấn. Để làm điều này, chắc chắn cửa sổ **Filter** đã hiển thị. Sau đó, chọn **Save As Query** từ thực đơn **File** hoặc kích chuột vào **Save As Query** trên thanh công cụ. Access hiển thị hộp thoại Save As Query như hình 3.



Hình 3: Hộp thoại Save As Query

Nhập tên mà bạn muốn để lưu filter. Tên có thể là bất kỳ tên nào mà bạn muốn. Ví dụ, nếu muốn lưu filter hiển thị các bảng kiểm kê cung cấp bởi Allied Industries, bạn có thể sử dụng tên **Supplied by Allied Industries**. Gõ tên và ấn ENTER hoặc kích chuột vào nút **OK**.

Để sử dụng lại filter, hiển thị cửa sổ Filter và lựa chọn **Load From Query** từ thực đơn **File**. Access hiển thị hộp thoại Applicable Filter như hình 4.



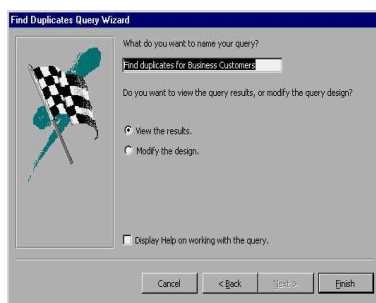
Hình 4: Hộp thoại Applicable Filter

Đánh dấu tên truy vấn mà bạn muốn nhập vào **Filter** và chọn **OK**. Access tải Filter. Bạn có thể sử dụng Filter hoặc thực hiện bất kỳ thay đổi để lọc như bạn muốn.

**Access cho phép bạn sắp xếp, lọc dữ liệu một cách nhanh chóng và dễ dàng. Đó là hai thao tác có liên quan với nhau, chúng có thể giúp bạn quản lý dữ liệu hiệu quả hơn. Chương này đã chỉ cho bạn cách thực hiện cả hai thao tác đó.**

Trước khi chuyển sang chương 6, phải chắc bạn hiểu được các nội dung sau:

- Sắp xếp dữ liệu theo thứ tự xác định.
- Thông tin có thể được lưu giữ theo thứ tự tăng dần (từ nhỏ nhất cho tới lớn nhất) hoặc giảm dần (từ lớn tới nhỏ).
- Bạn có thể sử dụng đặc tính sắp xếp đơn giản để sắp xếp nhanh một bảng dựa trên nội dung của một hay nhiều trường.
- Bạn có thể sử dụng sắp xếp phức tạp để sắp xếp nhiều trường không liên nhau. Và cũng có thể sử dụng nó để sắp xếp các trường theo các thứ tự khác nhau (tăng dần hoặc giảm dần).
- Để tạo một sắp xếp phức tạp, sử dụng cửa sổ Filter.
- Filter (bộ lọc) là một cách để giới hạn tạm thời những bản ghi mà Access hiển thị. Filter tạo ra một bảng con.
- Tạo một filter bằng cách xác định tiêu chuẩn để Access sử dụng khi lựa chọn các bản ghi sẽ hiển thị. Bạn xác định tiêu chuẩn sử dụng cửa sổ Filter.
- Filter sẽ bị mất khi bạn đóng cửa sổ bảng. Bạn có thể lưu và sau đó sử dụng lại filter nếu bạn lưu nó dưới dạng một truy vấn.



Bài 21: Sử dụng Query Wizards Access gồm 4 loại truy vấn Wizard khác nhau, và chỉ có một nơi bạn có thể thấy tất cả các loại truy vấn đó. Chọn Query từ thực đơn Insert, Access hiển thị hộp thoại New Query như hình 1.



Chú ý rằng có 5 tùy chọn được liệt kê trong hộp thoại. Tùy chọn đầu tiên (Design View) không thực sự là một truy vấn, nó cho phép bạn tạo một truy vấn từ hỗn hợp tự tạo. Mục đích của 4 tùy chọn còn lại như sau:

- **Simple Query Wizard.** Tùy chọn này cho phép bạn tạo truy vấn một cách đơn giản nhất. Wizard này giống như khi bạn kích đúp chuột vào Create Query By Using Wizard trong cửa sổ Database (hình 2).



Hình 2: Cửa sổ Database

- **Crosstab Query Wizard.** Tùy chọn này tạo một truy vấn Crosstab đơn giản, đã trình bày ở đầu chương.

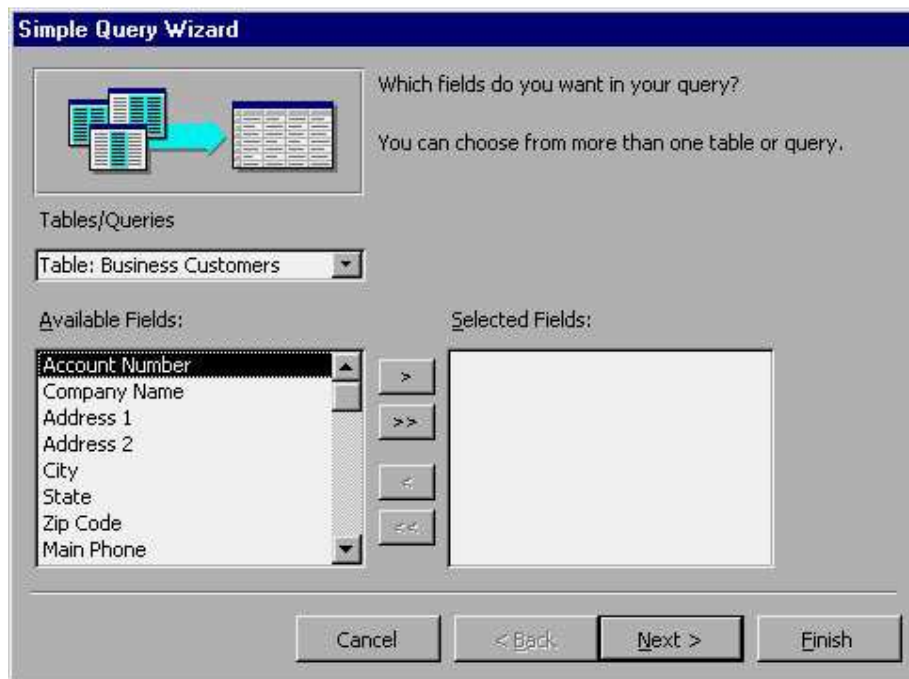
- **Find Duplicates Query.** Tạo một truy vấn lựa chọn hiển thị các bản ghi giống nhau trong một bảng.

- **Find Unmatched Query.** Tạo một truy vấn mà hiển thị các bản ghi không có mối quan hệ với bản ghi ở các bảng khác nhau. (Truy vấn này để sử dụng khi làm việc với cơ sở dữ liệu quan hệ, một khái niệm sẽ được giới thiệu ở chương 13, "Tìm hiểu mối quan hệ dữ liệu").

Truy vấn bạn lựa chọn sẽ quyết định Wizard nào được sử dụng bởi Access. Trong các phần sau đây, bạn sẽ học cách sử dụng ba trong 4 Wizards được đưa ra ở đây.

### Tạo một truy vấn đơn giản

Để tạo một truy vấn đơn giản, chọn **Simple Query Wizard** từ hộp thoại **New Query**, sau đó kích vào nút **OK**. Tương tự, bạn cũng có thể kích đúp chuột vào tùy chọn **Create Query By Using Wizard** sẵn có khi nút Queries được lựa chọn trong cửa sổ Database. Sau đó, Access hiển thị hộp thoại Simple Query Wizard như hình 3.

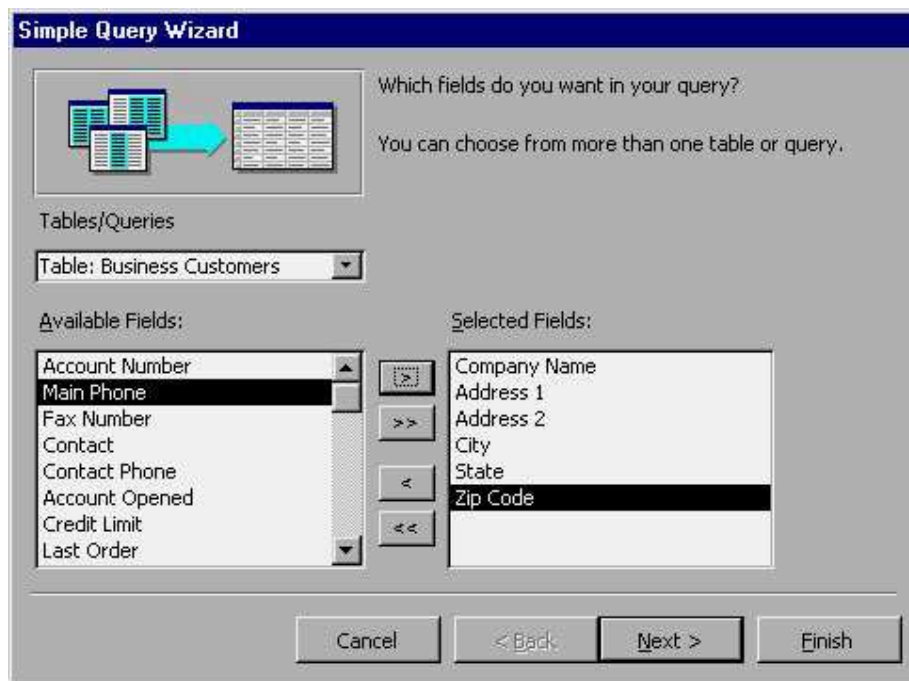


Hình 3: Hộp thoại Simple Query Wizard.

Phía bên trái hộp thoại Simple Query Wizard, bạn có thể sử dụng danh sách xổ xuống liệt kê để xác định những nguồn nào Access nên sử dụng cho truy vấn mới. Chú ý rằng, bạn có thể sử dụng nguồn là các bảng hoặc các truy vấn khác. Khi cơ sở dữ liệu Customers chỉ có duy nhất một bảng (bảng Business Customers), Access mặc định lựa chọn nó cho bạn. Nếu bạn đang làm việc với cơ sở dữ liệu chứa các bảng và các truy vấn khác, bạn có thể sử dụng danh sách xổ xuống để thay đổi những nguồn dữ liệu mà Wizard sử dụng.

Trong phần thấp hơn phía bên trái hộp thoại là danh sách của tất cả các trường được xác định tại bảng hay truy vấn đã được chọn. Nếu bạn thay đổi nguồn dữ liệu trong toàn bộ danh sách (được miêu tả ở đoạn trên) thì các trường sẵn có sẽ thay đổi theo. Bạn có thể sử dụng danh sách và điều khiển phía bên phải danh sách, để cho biết những trường nào bạn muốn được sử dụng vào truy vấn bạn đang tạo ra.

Để minh họa cho ví dụ này, lựa chọn các trường *Company Name*, *Address 1*, *Address 2*, *City*, *State*, và *Zip code*. Phải chắc chắn là sau khi chọn mỗi trường, bạn phải kích chuột vào nút >. Sau đó, Access sẽ hiển thị tên mỗi trường trong danh sách Selected Fields phía dưới bên phải hộp thoại. Khi hoàn thành việc lựa chọn các trường, hộp thoại xuất hiện như hình 4.



Hình 4: Hộp thoại Simple Query Wizard sau khi lựa chọn một số trường.

Với các trường đã được chọn, bạn đã sẵn sàng để tiến hành bước tiếp theo trong Wizard. Kích chuột vào nút **Next**, Access hiển thị hộp thoại như hình 5. Đây là hộp thoại cuối cùng trong Wizard này.



Hình 5: Hộp thoại cuối cùng của Simple Query Wizard.

Tại giao diện này, bạn có thể nhập một tên bạn muốn sử dụng để lưu truy vấn này. Tên được gợi ý đơn giản là tên bảng mà truy vấn đã dựa vào (Business Customers) với từ "Query" được thêm vào. Truy vấn đơn giản bạn đã tạo sử dụng hộp thoại trước có thể được sử dụng cho các nhãn thư, vì nó chỉ chứa thông tin về địa chỉ cho danh sách các khách hàng. Nhập tên Address List, sau đó kích chuột vào nút **Finish**. Access hiển thị kết quả truy vấn đó như hình 6.



Company Name	Address 1	Address 2	City	State	Zip Code
Birch Enterprise	3456 Maple Lane	PO Box 549	Kansas City	MO	64147
Read A Book S	1325 East Main		Parrot	KY	40567
Pocono Hardwa	7500 West Map		Pocono	PA	18347
JP Consolidatec	1821 5th Street		Lubbock	TX	79452
Bridger Law	800 Main Street	PO Box 1281	Alpena	AR	72611
Meadowbrook M	733 North 2nd S		Cameron	AZ	86020
Pete's Casino &	35 Frontage Ro		Crystal	NV	89402
Desert Oasis	5th & Main		Mammoth	CA	93456
Rain Water Con	1805 Palm Aver	PO Box 6035	Olancha	CA	93549
Eden's Ski Supj	1705 Mountain	PO Box 837	Pritchett	CO	81064
Buffalo Distrib	1309 Eagle's Pl		Chugwater	WY	82210
Bob's Retail	326 North Pine		Lost	WY	82224
East Wind Cont	906 South Bloo	PO Box 10127	Catskill	NY	12414
Tipper's Winterv	75 Bymshire Rc		Thetford	VT	05075
Brendon Dairy	1805 Somershir	PO Box 1807	Sharon	CT	06069
Sams Systems	905 West Lynde	PO Box 209	Dillsburg	PA	17019
M&T Distribut	1200 D... ..		Davies	PA	17010

Hình 6: Kết quả của Simple Query Wizard.

### Tạo một truy vấn Crosstab

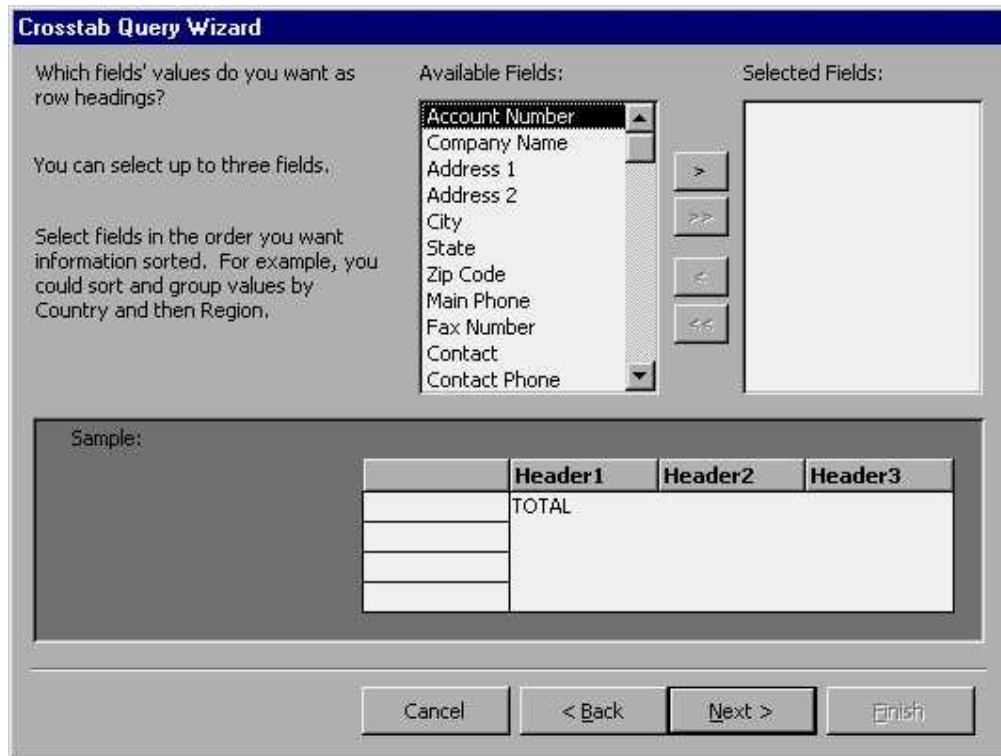
Để tạo một truy vấn Crosstab, đầu tiên mở hộp thoại **New Query** bằng cách chọn Query từ thực đơn **Insert**. Từ hộp thoại đó, chọn **Crosstab Query Wizard** và kích chuột vào nút **OK**. Access hiển thị hộp thoại Crosstab Query Wizard như hình 7.

Hình 7: Hộp thoại Crosstab Query Wizard.

Phía trên cùng hộp thoại Crosstab Query Wizard, Access hỏi bạn lựa chọn bảng nào muốn truy vấn dữ liệu. Vì cơ sở dữ liệu này chỉ có một bảng (Business Customers), Access đã lựa chọn nó cho bạn. Nếu bạn đang làm việc với một cơ sở dữ liệu có chứa các bảng và truy vấn khác, bạn có thể sử dụng vùng View ở giữa hộp thoại để thay đổi nguồn dữ liệu mà Wizard đưa ra.

Phía dưới cùng hộp thoại, Wizard hiển thị cách truy vấn Crosstab đã được hoàn thành. Access

cập nhật trình bày này mỗi lúc bạn thực hiện thay đổi trong khi sử dụng Wizard. Để chuyển sang hộp thoại tiếp theo, kích chuột vào nút **Next** (Hình 8).

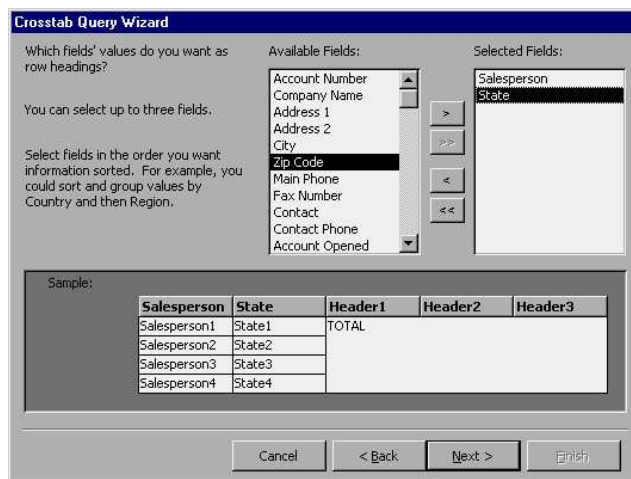


Hình 8: Lựa chọn các dòng.

Truy vấn Crosstab cho phép bạn xác định kết quả hiển thị theo cả hàng và cột. Sử dụng Wizard, bạn nên chỉ ra những trường muốn xuất hiện trong mỗi dòng.

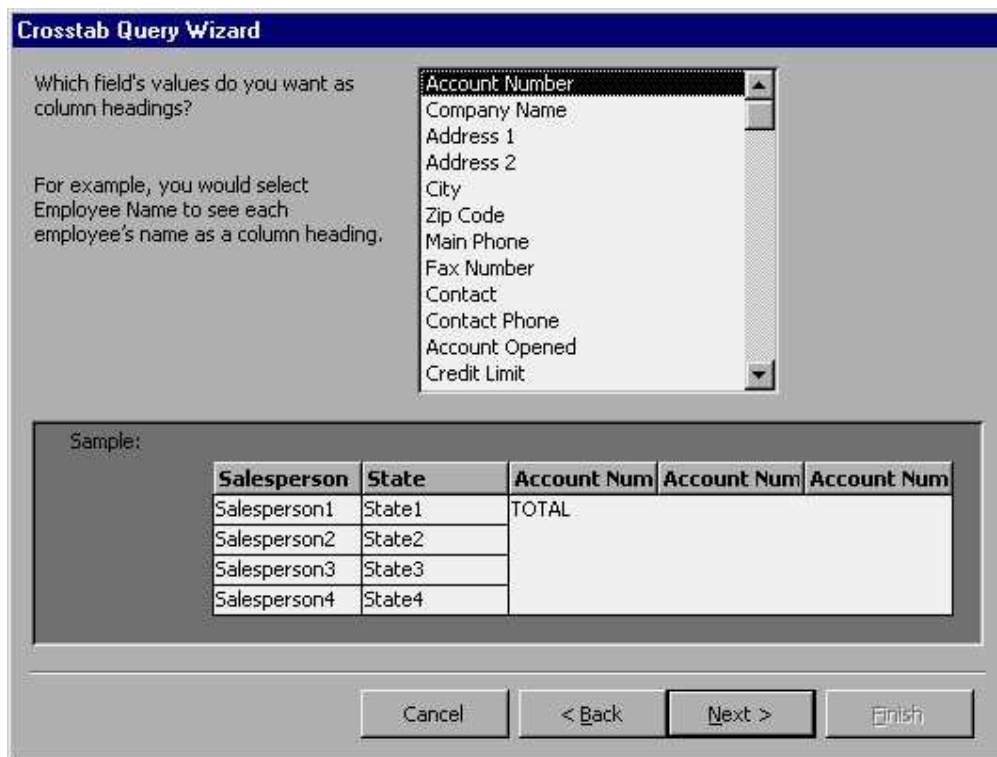
Ví dụ, giả sử bạn muốn lập bảng về số tài khoản còn hoạt động và không còn hoạt động. Trong danh sách Available Fields, bạn nên lựa chọn trường *Salesperson* và kích chuột vào nút >. Chú ý rằng Wizard cập nhật mẫu crosstab ở phía dưới cùng hộp thoại.

Tiếp theo, bạn nên chọn trường *State* và kích chuột vào nút >. Tiếp tục, Wizard cập nhật mẫu crosstab ở phía dưới hộp thoại như hình 8.



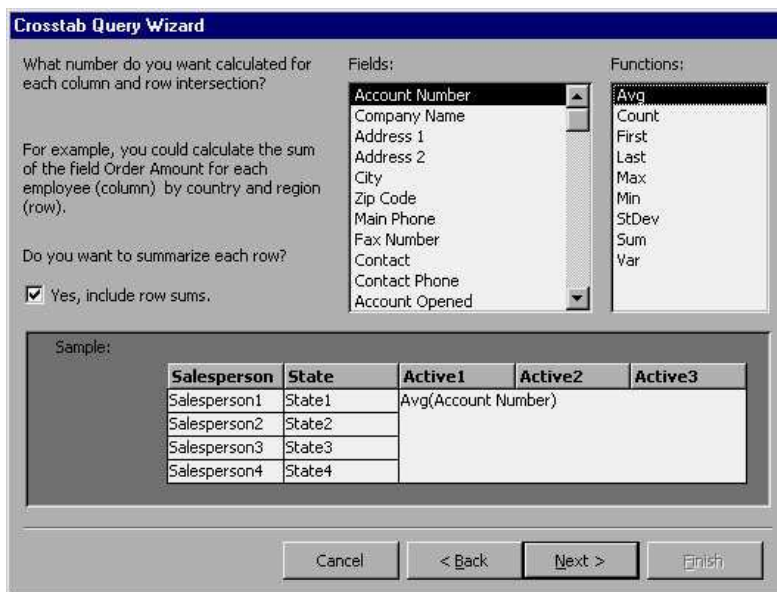
Hình 9: Truy vấn crosstab sau khi lựa chọn tiêu đề hàng

Bạn chỉ thực hiện theo Access để hiển thị những người bán hàng và State phía bên trái của crosstab. Kích chuột vào nút **Next**. Access hiển thị hộp thoại như hình 10.



Hình 10: Lựa chọn cột

Trong hộp thoại này, Access hỏi bạn để lựa chọn những trường mà nó sẽ trình bày thành các cột. Do bạn muốn truy vấn này chỉ cho biết có bao nhiêu tài khoản còn hoạt động hay không đối với mỗi người bán hàng, bạn nên chọn trường *Active*. Sau khi chọn trường *Active*, chú ý rằng Access cập nhật theo mẫu như ở phía dưới hộp thoại. Kích chuột vào nút **Next** để thực hiện tiếp. Access hiển thị hộp thoại như hình 11.

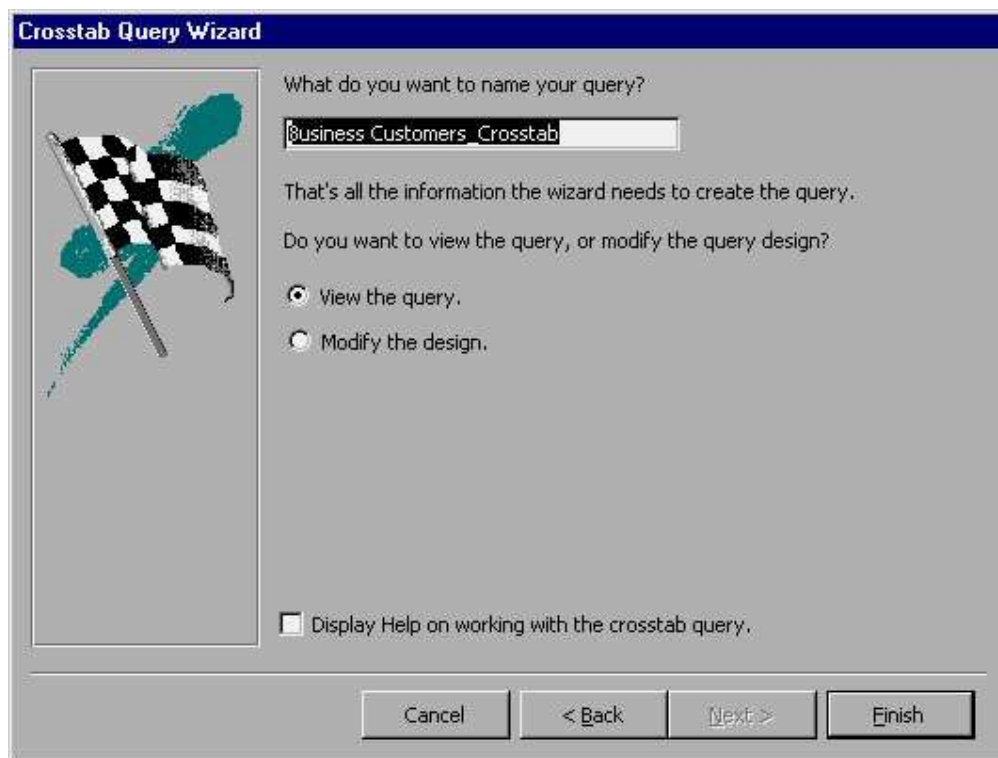


Hình 11: Lựa chọn các thao tác cho truy vấn.

Trong hộp thoại này, Access hỏi bạn để xác định những gì bạn muốn truy vấn tính toán. Tại phía bên phải hộp thoại, hộp Functions liệt kê tất cả các phép tính sẵn có khác nhau. Bạn có thể thực hiện nhiều phép tính trên nhiều trường, tuy nhiên một vài sự kết hợp của các trường và các phép tính không thể thực hiện được. Ví dụ, nếu bạn chọn một trường text, việc thực hiện tính trung bình không thể làm được, bạn chỉ nên thực hiện tính toán trên một trường số hoặc trường tiền tệ.

Để kết thúc truy vấn này, chọn trường *Company Name* và chọn phép tính **Count**. Chú ý rằng thông tin trong vùng mẫu của hộp thoại thay đổi cùng với lựa chọn của bạn. Truy vấn sẽ tính toán cách mà các tài khoản còn hoạt động hay không (có một tên công ty cho mỗi tài khoản) có trong mỗi state cho mỗi người đại diện bán hàng.

Cuối cùng, chú ý rằng có 1 hộp chọn bạn có thể sử dụng để cho biết truy vấn cần cung cấp một tóm tắt cho mỗi dòng. Với một crosstab đơn giản là chỉ có 2 cột (active hoặc inactive) thì một bản tóm tắt có lẽ không cần thiết. Kích chuột vào hộp chọn để tắt bản tóm tắt. Sau khi sự lựa chọn được hoàn thành, bạn có thể kích chuột vào nút **Next**. Access sẽ hiển thị hộp thoại như hình 12.



Hình 12: Access hỏi bạn về tên truy vấn

Ở hộp thoại này, bạn hoàn thành việc tạo truy vấn và cần lưu nó lại. Access hỏi bạn tên để sử dụng cho truy vấn này. Với các tên cho những phần khác của cơ sở dữ liệu, bạn nên sử dụng một vài tên đã miêu tả, như Active Account Tally. Gõ tên và kích chuột vào nút **Finish**. Access sẽ lưu truy vấn và cập nhật danh sách truy vấn trong cửa sổ Database, sau đó mở truy vấn để hiển thị kết quả vừa làm. Hình 13 hiển thị khi truy vấn được hoàn thành.

The screenshot shows a Microsoft Access window with a Crosstab Query titled 'Active Account Tally'. The query results are displayed in a grid with columns for Salesperson, State, and two columns labeled '-1' and '0'. The data is as follows:

Salesperson	State	-1	0
Cathy Sutherland	MI	7	
David Jones	CT	1	1
David Jones	DE	3	
David Jones	NY	1	
David Jones	PA	2	1
David Jones	VT	1	
Edward Holmes	CO		1
Edward Holmes	WY	2	
Jeffrey Thomas	AZ	1	
Jeffrey Thomas	CA	2	1
Jeffrey Thomas	NV	1	
Jenny Goodmar	ID	2	
Jerome Smith	AR	1	1

Hình 13: Kết quả của truy vấn.

Khi xem tại crosstab, một trong những điều đầu tiên mà bạn chú ý rằng các cột có đề là -1 và 0. Nhớ rằng, trường *Active* là một trường có kiểu dữ liệu Yes/No, có nghĩa là Access lưu giữ giá trị trường này như một trong hai số: -1 và 0. Như vậy, những số này cũng hiển thị trong tiêu đề. Sau chương này, bạn sẽ học cách thay đổi tiêu đề bằng cách thay đổi thuộc tính kết hợp với truy vấn.

Nếu bạn xem qua trường hợp lạ này trong phần tiêu đề, bạn có thể thấy rằng, crosstab cung cấp thông tin mà bạn đã yêu cầu. Nó hiển thị tình trạng tài khoản trên active (Yes hoặc -1) và inactive (No hoặc 0) cho mỗi state và rep. Crosstab Query Wizard rất dễ dàng cho bạn tạo truy vấn dữ liệu.

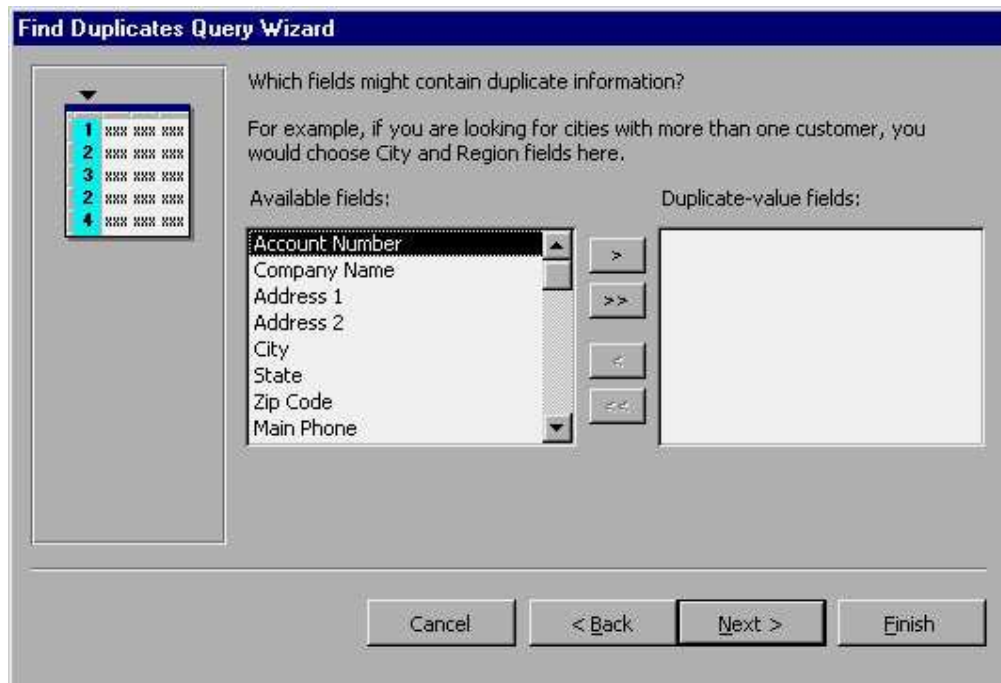
### Tạo một Find Duplicates Query

Bạn cũng có thể sử dụng hộp thoại New Query để tạo một truy vấn Find Duplicates, nó cho phép bạn nhanh chóng tìm kiếm một bảng có các trường trùng nhau. Chọn **Find Duplicates Query Wizard** trong hộp thoại và kích chuột vào nút **OK**. Lần lượt, Access hiển thị hộp thoại Find Duplicates Query Wizard như hình 14.



Hình 14: Hộp thoại Find Duplicates Query Wizard.

Trong hộp thoại này, Access liệt kê các nguồn dữ liệu bạn có thể sử dụng cho truy vấn. Thông thường, bạn sử dụng một bảng như nguồn dữ liệu nhưng với Wizard này bạn cũng có thể sử dụng truy vấn khác nữa. Bởi vì bảng Business Customers là duy nhất trong cơ sở dữ liệu Customers, Access đã lựa chọn bảng này cho bạn. Kích chuột vào nút **Next** để chuyển tới hộp thoại như hình 15.



Hình 15: Chọn một trường để kiểm tra.

Trong hộp thoại này, Access hỏi bạn những trường nào cần kiểm tra giá trị trùng. Mặc dù không có giới hạn về số các trường bạn có thể có trong kiểm tra truy vấn, nhưng bạn cũng chỉ nên chọn những trường cần thiết để chắc chắn là không có sự trùng lặp. Trong trường hợp của bảng Business Customers, chỉ cần kiểm tra trường *Company Name* và *Contact* để xác định nếu có sự trùng lặp.

Đầu tiên, chọn trường *Company Name* và kích chuột vào nút **>**. Tiếp theo, chọn trường *Contact* và kích tiếp vào nút **>**. Access hiển thị hai trường theo thứ tự trong danh sách phía bên phải hộp thoại. Sau khi xác định xong các trường, kích chuột vào nút **Next**. Access hiển thị hộp thoại như hình 16.



Hình 16: Hiển thị thông tin thêm.

Trong hộp thoại này, Access hỏi nếu bạn muốn truy vấn hiển thị thêm thông tin. Khi xem lại các bản ghi trùng, bạn có thể sử dụng thông tin này nhằm giúp việc quyết định bản ghi nào muốn lưu và bản ghi nào muốn xóa. Đối với bảng Business Customers, bạn muốn Access hiển thị các trường *Contact Phone*, *Last Order*, *Salesperson*. Chọn một vài trường (nhớ kích nút > sau khi chọn mỗi trường). Access thêm tên của mỗi trường vào danh sách phía bên phải hộp thoại. Khi hoàn thành, kích chuột vào nút **Next**, Access hiển thị hộp thoại như hình 17.



Hình 17: Kết thúc truy vấn.

Lúc này bạn đã hoàn thành việc tạo truy vấn, Access yêu cầu bạn đặt tên cho truy vấn. Từ gợi ý của Wizard cho tên của truy vấn này, kích chuột vào nút **Finish** để chấp nhận tên đó. Access lưu truy vấn dưới tên này và cập nhật danh sách truy vấn trong cửa sổ Database. Access mở truy vấn để hiển thị kết quả vừa hoàn thành như hình 18.

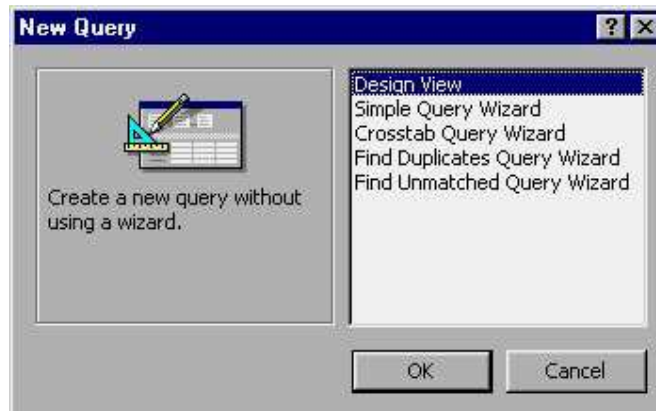
Company Name	Contact	Main Phone	Account Open	Credit Limit	Last Order	Salesperson
Brendon Dairy	Nathan Gehric	(203) 513-2906	4/1/95	\$1,500.00	7/16/99	David Jones
Brendon Dairy	Nathan Gehric	(203) 513-2906	12/15/95	\$1,500.00	3/18/99	David Jones
Bridger Law	Thomas Bridger	(501) 328-1869	2/17/96	\$2,000.00	5/19/99	Jerome Smith
Bridger Law	Thomas Bridger	(501) 328-1869	2/17/96	\$2,000.00	5/19/99	Jerome Smith
Morris Corporat	Greg Crothers	(614) 289-7359	4/8/96	\$1,750.00	8/9/98	John Davis
Morris Corporat	Greg Crothers	(614) 289-7359	4/8/96	\$1,500.00	8/9/98	John Davis
Rain Water Con	Jake Conway	(619) 807-2500	10/16/93	\$1,750.00	3/15/99	Jeffrey Thomas
Rain Water Con	Jake Conway	(619) 807-2500	4/19/99	\$1,500.00	7/2/99	Jeffrey Thomas
Rhymes & Rum	Larry Strong	(302) 218-9465	5/6/94	\$2,000.00	6/27/99	David Jones
Rhymes & Rum	Larry Strong	(302) 218-9465	7/8/95	\$1,500.00	7/8/98	David Jones
Wintergreen Ma	Sam Harrison	(205) 812-9631	4/5/97	\$1,500.00	8/16/99	Luke Jefferson
Wintergreen Ma	Sam Harrison	(205) 812-9631	4/5/97	\$1,500.00	8/16/99	Luke Jefferson
				\$0.00		

Hình 18: Kết quả của truy vấn Find Duplicates.

Bây giờ bạn có thể xem lại các bản ghi trùng lặp và quyết định những việc muốn làm với chúng. Nếu bạn xóa bất kỳ bản ghi nào trong màn hình, Access sẽ xóa bản ghi tương ứng trong bảng phía bên dưới (Business Customers). Với lý do này, truy vấn Find Duplicates rất có ích trong việc quản lý thông tin ở bảng.

# Bài 22: Sử dụng Query Wizards

Access gồm 4 loại truy vấn Wizard khác nhau, và chỉ có một nơi bạn có thể thấy tất cả các loại truy vấn đó. Chọn Query từ thực đơn Insert, Access hiển thị hộp thoại New Query như hình 1.



Hình 1: Hộp thoại New Query.

Chú ý rằng có 5 tùy chọn được liệt kê trong hộp thoại. Tùy chọn đầu tiên (Design View) không thực sự là một truy vấn, nó cho phép bạn tạo một truy vấn từ hỗn hợp tự tạo. Mục đích của 4 tùy chọn còn lại như sau:

- **Simple Query Wizard.** Tùy chọn này cho phép bạn tạo truy vấn một cách đơn giản nhất. Wizard này giống như khi bạn kích đúp chuột vào Create Query By Using Wizard trong cửa sổ Database (hình 2).



Hình 2: Cửa sổ Database

- **Crosstab Query Wizard.** Tùy chọn này tạo một truy vấn Crosstab đơn giản, đã trình bày ở đầu chương.

- **Find Duplicates Query.** Tạo một truy vấn lựa chọn hiển thị các bản ghi giống nhau trong một bảng.

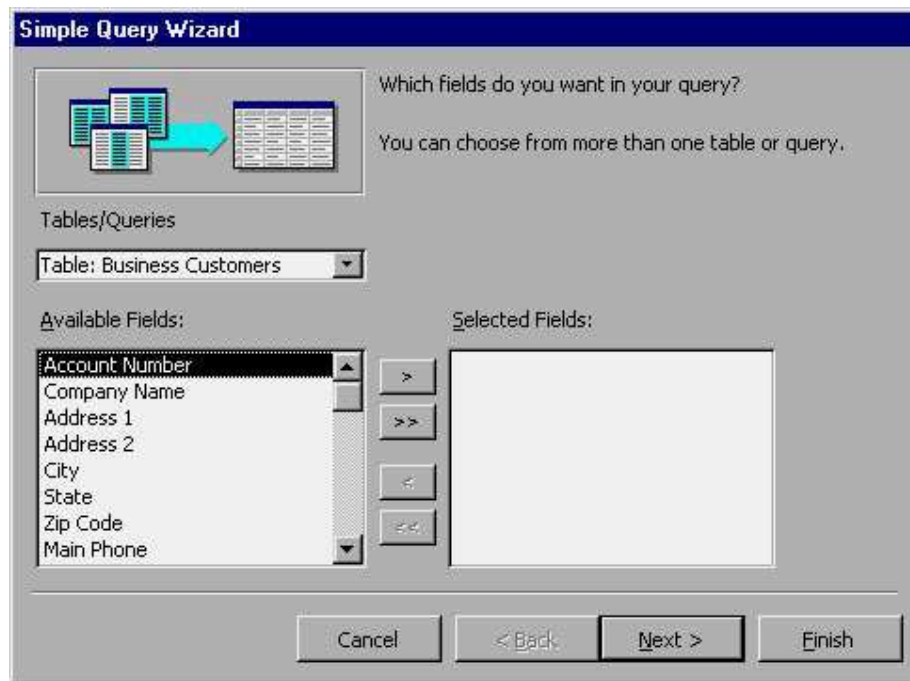


• **Find Unmatched Query.** Tạo một truy vấn mà hiển thị các bản ghi không có mối quan hệ với bản ghi ở các bảng khác nhau. (Truy vấn này để sử dụng khi làm việc với cơ sở dữ liệu quan hệ, một khái niệm sẽ được giới thiệu ở chương 13, “*Tìm hiểu mối quan hệ dữ liệu*”).

Truy vấn bạn lựa chọn sẽ quyết định Wizard nào được sử dụng bởi Access. Trong các phần sau đây, bạn sẽ học cách sử dụng ba trong 4 Wizards được đưa ra ở đây.

### Tạo một truy vấn đơn giản

Để tạo một truy vấn đơn giản, chọn **Simple Query Wizard** từ hộp thoại **New Query**, sau đó kích vào nút **OK**. Tương tự, bạn cũng có thể kích đúp chuột vào tùy chọn **Create Query By Using Wizard** sẵn có khi nút Queries được lựa chọn trong cửa sổ Database. Sau đó, Access hiển thị hộp thoại Simple Query Wizard như hình 3.

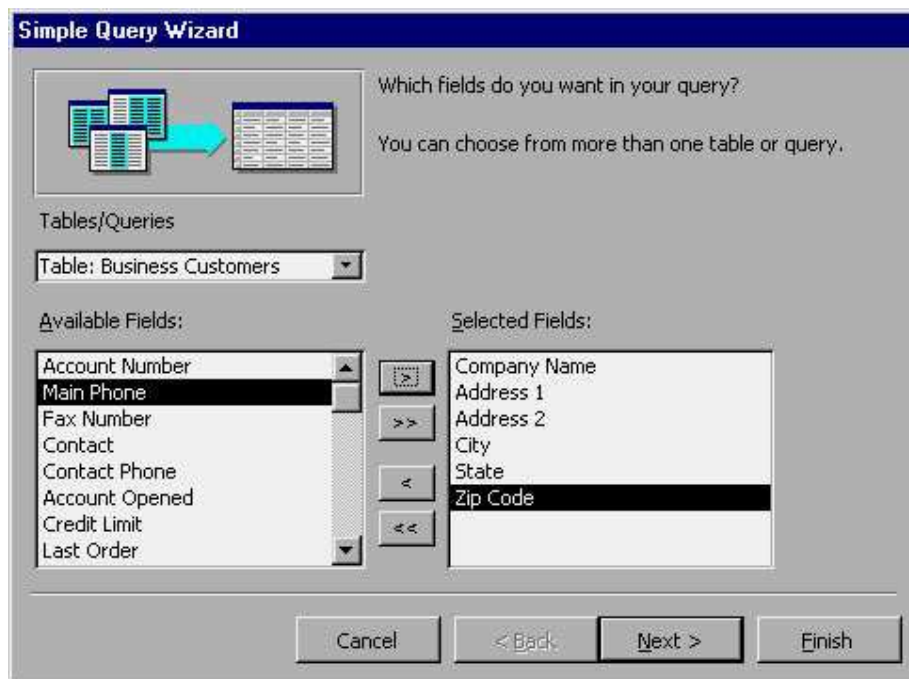


Hình 3: Hộp thoại Simple Query Wizard.

Phía bên trái hộp thoại Simple Query Wizard, bạn có thể sử dụng danh sách xổ xuống liệt kê để xác định những nguồn nào Access nên sử dụng cho truy vấn mới. Chú ý rằng, bạn có thể sử dụng nguồn là các bảng hoặc các truy vấn khác. Khi cơ sở dữ liệu Customers chỉ có duy nhất một bảng (bảng Business Customers), Access mặc định lựa chọn nó cho bạn. Nếu bạn đang làm việc với cơ sở dữ liệu chứa các bảng và các truy vấn khác, bạn có thể sử dụng danh sách xổ xuống để thay đổi những nguồn dữ liệu mà Wizard sử dụng.

Trong phần thấp hơn phía bên trái hộp thoại là danh sách của tất cả các trường được xác định tại bảng hay truy vấn đã được chọn. Nếu bạn thay đổi nguồn dữ liệu trong toàn bộ danh sách (được miêu tả ở đoạn trên) thì các trường sẵn có sẽ thay đổi theo. Bạn có thể sử dụng danh sách và điều khiển phía bên phải danh sách, để cho biết những trường nào bạn muốn được sử dụng vào truy vấn bạn đang tạo ra.

Để minh họa cho ví dụ này, lựa chọn các trường *Company Name*, *Address 1*, *Address 2*, *City*, *State*, và *Zip code*. Phải chắc chắn là sau khi chọn mỗi trường, bạn phải kích chuột vào nút **>**. Sau đó, Access sẽ hiển thị tên mỗi trường trong danh sách Selected Fields phía dưới bên phải hộp thoại. Khi hoàn thành việc lựa chọn các trường, hộp thoại xuất hiện như hình 4.



Hình 4: Hộp thoại Simple Query Wizard sau khi lựa chọn một số trường.

Với các trường đã được chọn, bạn đã sẵn sàng để tiến hành bước tiếp theo trong Wizard. Kích chuột vào nút **Next**, Access hiển thị hộp thoại như hình 5. Đây là hộp thoại cuối cùng trong Wizard này.



Hình 5: Hộp thoại cuối cùng của Simple Query Wizard.

Tại giao diện này, bạn có thể nhập một tên bạn muốn sử dụng để lưu truy vấn này. Tên được gợi ý đơn giản là tên bảng mà truy vấn đã dựa vào (Business Customers) với từ "Query" được thêm vào. Truy vấn đơn giản bạn đã tạo sử dụng hộp thoại trước có thể được sử dụng cho các nhãn thư, vì nó chỉ chứa thông tin về địa chỉ cho danh sách các khách hàng. Nhập tên Address List, sau đó kích chuột vào nút **Finish**. Access hiển thị kết quả truy vấn đó như hình 6.

Company Name	Address 1	Address 2	City	State	Zip Code
Birch Enterprise	3456 Maple Lane	PO Box 549	Kansas City	MO	64147
Read A Book S	1325 East Main		Parrot	KY	40567
Pocono Hardwa	7500 West Map		Pocono	PA	18347
JP Consolidatec	1821 5th Street		Lubbock	TX	79452
Bridger Law	800 Main Street	PO Box 1281	Alpena	AR	72611
Meadowbrook M	733 North 2nd S		Cameron	AZ	86020
Pete's Casino &	35 Frontage Ro		Crystal	NV	89402
Desert Oasis	5th & Main		Mammoth	CA	93456
Rain Water Con	1805 Palm Aver	PO Box 6035	Olancha	CA	93549
Eden's Ski Supj	1705 Mountain	PO Box 837	Pritchett	CO	81064
Buffalo Distrib	1309 Eagle's Pl		Chugwater	WY	82210
Bob's Retail	326 North Pine		Lost	WY	82224
East Wind Cont	906 South Bloo	PO Box 10127	Catskill	NY	12414
Tipper's Winterv	75 Bymshire Rc		Thetford	VT	05075
Brendon Dairy	1805 Somershir	PO Box 1807	Sharon	CT	06069
Sams Systems	905 West Lynde	PO Box 209	Dillsburg	PA	17019
M&T Distribut	1200 D... ..		Davies	PA	17010

Hình 6: Kết quả của Simple Query Wizard.

### Tạo một truy vấn Crosstab

Để tạo một truy vấn Crosstab, đầu tiên mở hộp thoại **New Query** bằng cách chọn Query từ thực đơn **Insert**. Từ hộp thoại đó, chọn **Crosstab Query Wizard** và kích chuột vào nút **OK**. Access hiển thị hộp thoại Crosstab Query Wizard như hình 7.

Which table or query contains the fields you want for the crosstab query results?

To include fields from more than one table, create a query containing all the fields you need and then use this query to make the crosstab query.

View

Tables  Queries  Both

Sample:

	Header1	Header2	Header3
TOTAL			

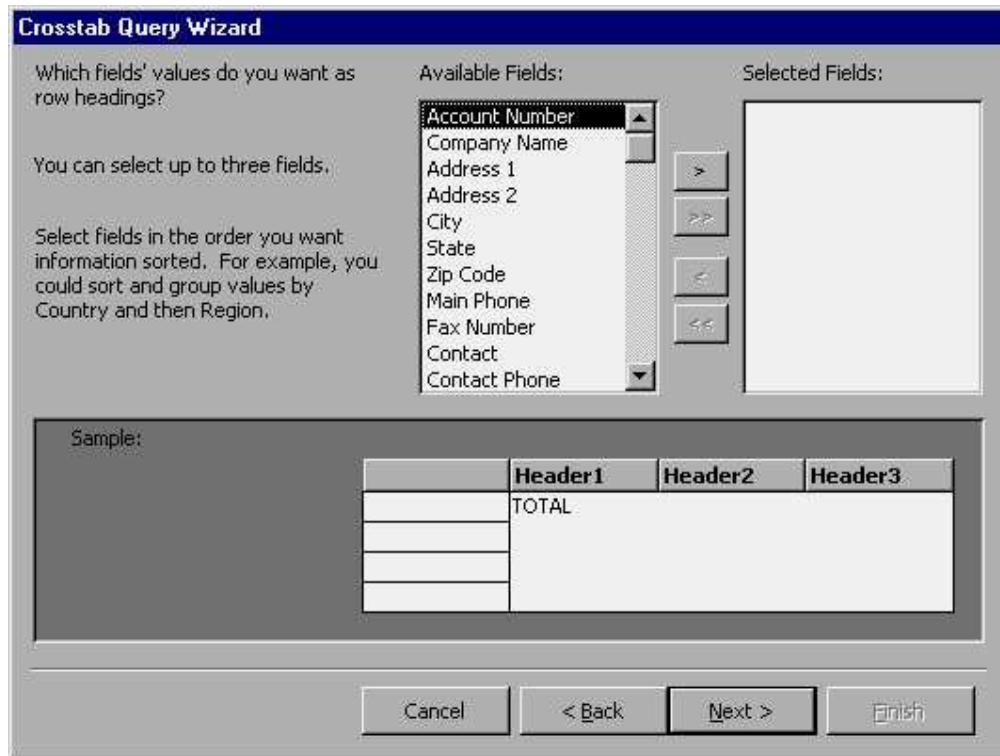
Cancel < Back Next > Finish

Hình 7: Hộp thoại Crosstab Query Wizard.

Phía trên cùng hộp thoại Crosstab Query Wizard, Access hỏi bạn lựa chọn bảng nào muốn truy vấn dữ liệu. Vì cơ sở dữ liệu này chỉ có một bảng (Business Customers), Access đã lựa chọn nó cho bạn. Nếu bạn đang làm việc với một cơ sở dữ liệu có chứa các bảng và truy vấn khác, bạn có thể sử dụng vùng View ở giữa hộp thoại để thay đổi nguồn dữ liệu mà Wizard đưa ra.

Phía dưới cùng hộp thoại, Wizard hiển thị cách truy vấn Crosstab đã được hoàn thành. Access

cập nhật trình bày này mỗi lúc bạn thực hiện thay đổi trong khi sử dụng Wizard. Để chuyển sang hộp thoại tiếp theo, kích chuột vào nút **Next** (Hình 8).

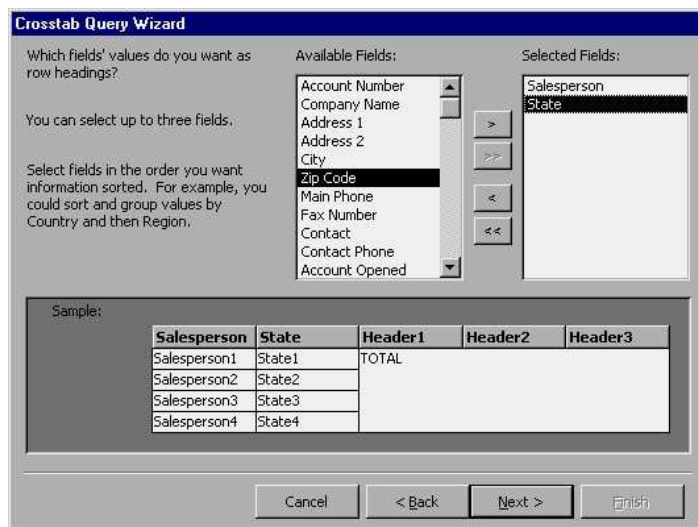


Hình 8: Lựa chọn các dòng.

Truy vấn Crosstab cho phép bạn xác định kết quả hiển thị theo cả hàng và cột. Sử dụng Wizard, bạn nên chỉ ra những trường muốn xuất hiện trong mỗi dòng.

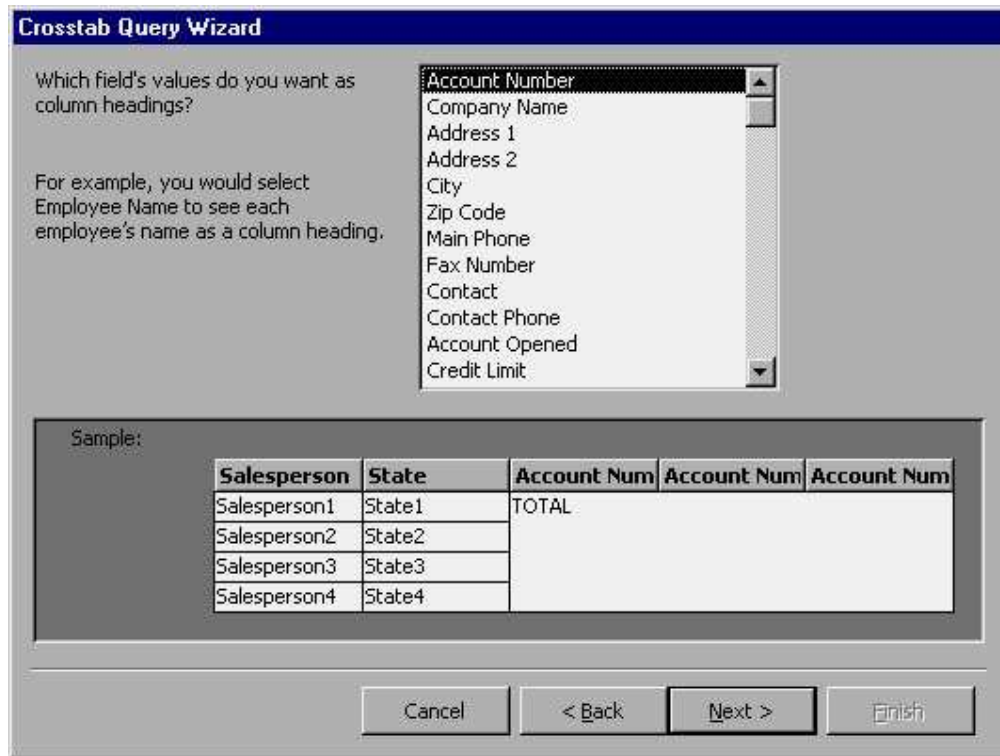
Ví dụ, giả sử bạn muốn lập bảng về số tài khoản còn hoạt động và không còn hoạt động. Trong danh sách Available Fields, bạn nên lựa chọn trường *Salesperson* và kích chuột vào nút >. Chú ý rằng Wizard cập nhật mẫu crosstab ở phía dưới cùng hộp thoại.

Tiếp theo, bạn nên chọn trường *State* và kích chuột vào nút >. Tiếp tục, Wizard cập nhật mẫu crosstab ở phía dưới hộp thoại như hình 8.



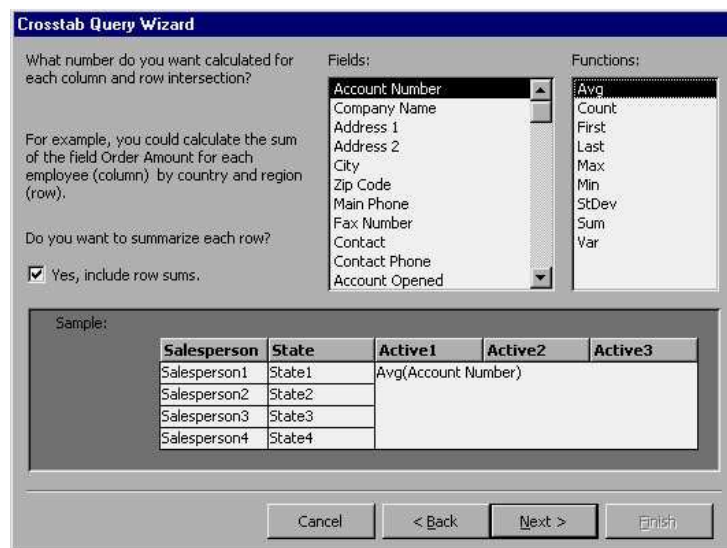
Hình 9: Truy vấn crosstab sau khi lựa chọn tiêu đề hàng

Bạn chỉ thực hiện theo Access để hiển thị những người bán hàng và State phía bên trái của crosstab. Kích chuột vào nút **Next**. Access hiển thị hộp thoại như hình 10.



Hình 10: Lựa chọn cột

Trong hộp thoại này, Access hỏi bạn để lựa chọn những trường mà nó sẽ trình bày thành các cột. Do bạn muốn truy vấn này chỉ cho biết có bao nhiêu tài khoản còn hoạt động hay không đối với mỗi người bán hàng, bạn nên chọn trường *Active*. Sau khi chọn trường *Active*, chú ý rằng Access cập nhật theo mẫu như ở phía dưới hộp thoại. Kích chuột vào nút **Next** để thực hiện tiếp. Access hiển thị hộp thoại như hình 11.



Hình 11: Lựa chọn các thao tác cho truy vấn.

Trong hộp thoại này, Access hỏi bạn để xác định những gì bạn muốn truy vấn tính toán. Tại phía bên phải hộp thoại, hộp Functions liệt kê tất cả các phép tính sẵn có khác nhau. Bạn có thể thực hiện nhiều phép tính trên nhiều trường, tuy nhiên một vài sự kết hợp của các trường và các phép tính không thể thực hiện được. Ví dụ, nếu bạn chọn một trường text, việc thực hiện tính trung

biểu không thể làm được, bạn chỉ nên thực hiện tính toán trên một trường số hoặc trường tiền tệ.

Để kết thúc truy vấn này, chọn trường *Company Name* và chọn phép tính **Count**. Chú ý rằng thông tin trong vùng mẫu của hộp thoại thay đổi cùng với lựa chọn của bạn. Truy vấn sẽ tính toán cách mà các tài khoản còn hoạt động hay không (có một tên công ty cho mỗi tài khoản) có trong mỗi state cho mỗi người đại diện bán hàng.

Cuối cùng, chú ý rằng có 1 hộp chọn bạn có thể sử dụng để cho biết truy vấn cần cung cấp một tóm tắt cho mỗi dòng. Với một crosstab đơn giản là chỉ có 2 cột (active hoặc inactive) thì một bản tóm tắt có lẽ không cần thiết. Kích chuột vào hộp chọn để tắt bản tóm tắt. Sau khi sự lựa chọn được hoàn thành, bạn có thể kích chuột vào nút **Next**. Access sẽ hiển thị hộp thoại như hình 12.



Hình 12: Access hỏi bạn về tên truy vấn

Ở hộp thoại này, bạn hoàn thành việc tạo truy vấn và cần lưu nó lại. Access hỏi bạn tên để sử dụng cho truy vấn này. Với các tên cho những phần khác của cơ sở dữ liệu, bạn nên sử dụng một vài tên đã miêu tả, như Active Account Tally. Gõ tên và kích chuột vào nút **Finish**. Access sẽ lưu truy vấn và cập nhật danh sách truy vấn trong cửa sổ Database, sau đó mở truy vấn để hiển thị kết quả vừa làm. Hình 13 hiển thị khi truy vấn được hoàn thành.

The screenshot shows a Microsoft Access window with a menu bar (File, Edit, View, Insert, Format, Records, Tools, Window, Help) and a toolbar. The main window displays a Crosstab Query titled 'Active Account Tally : Crosstab Query'. The data is presented in a grid with columns for Salesperson, State, and two columns labeled '-1' and '0'. The rows list salespersons and their corresponding states with counts for each category.

Salesperson	State	-1	0
Cathy Sutherland	MI	7	
David Jones	CT	1	1
David Jones	DE	3	
David Jones	NY	1	
David Jones	PA	2	1
David Jones	VT	1	
Edward Holmes	CO		1
Edward Holmes	WY	2	
Jeffrey Thomas	AZ	1	
Jeffrey Thomas	CA	2	1
Jeffrey Thomas	NV	1	
Jenny Goodmar	ID	2	
Jerome Smith	AR	1	1

At the bottom of the window, it indicates 'Record: 14' and '1 of 22'.

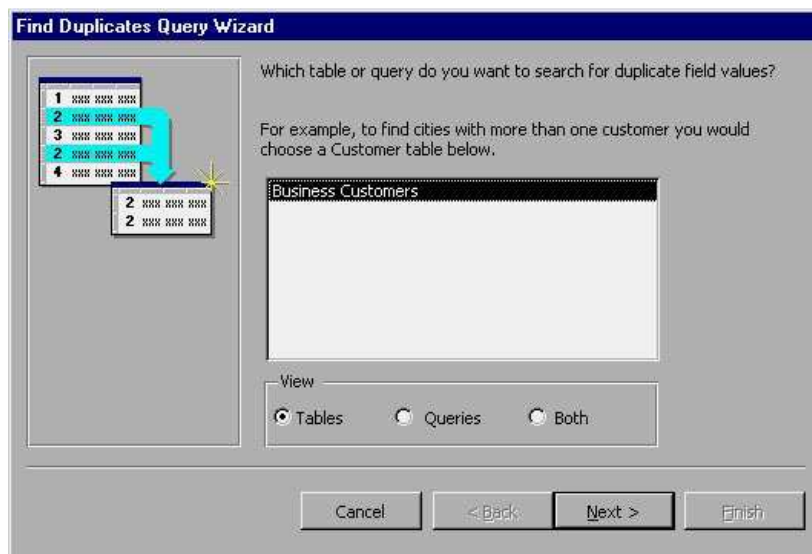
Hình 13: Kết quả của truy vấn.

Khi xem tại crosstab, một trong những điều đầu tiên mà bạn chú ý rằng các cột có đề là -1 và 0. Nhớ rằng, trường *Active* là một trường có kiểu dữ liệu Yes/No, có nghĩa là Access lưu giữ giá trị trường này như một trong hai số: -1 và 0. Như vậy, những số này cũng hiển thị trong tiêu đề. Sau chương này, bạn sẽ học cách thay đổi tiêu đề bằng cách thay đổi thuộc tính kết hợp với truy vấn.

Nếu bạn xem qua trường hợp lạ này trong phần tiêu đề, bạn có thể thấy rằng, crosstab cung cấp thông tin mà bạn đã yêu cầu. Nó hiển thị tình trạng tài khoản trên active (Yes hoặc -1) và inactive (No hoặc 0) cho mỗi state và rep. Crosstab Query Wizard rất dễ dàng cho bạn tạo truy vấn dữ liệu.

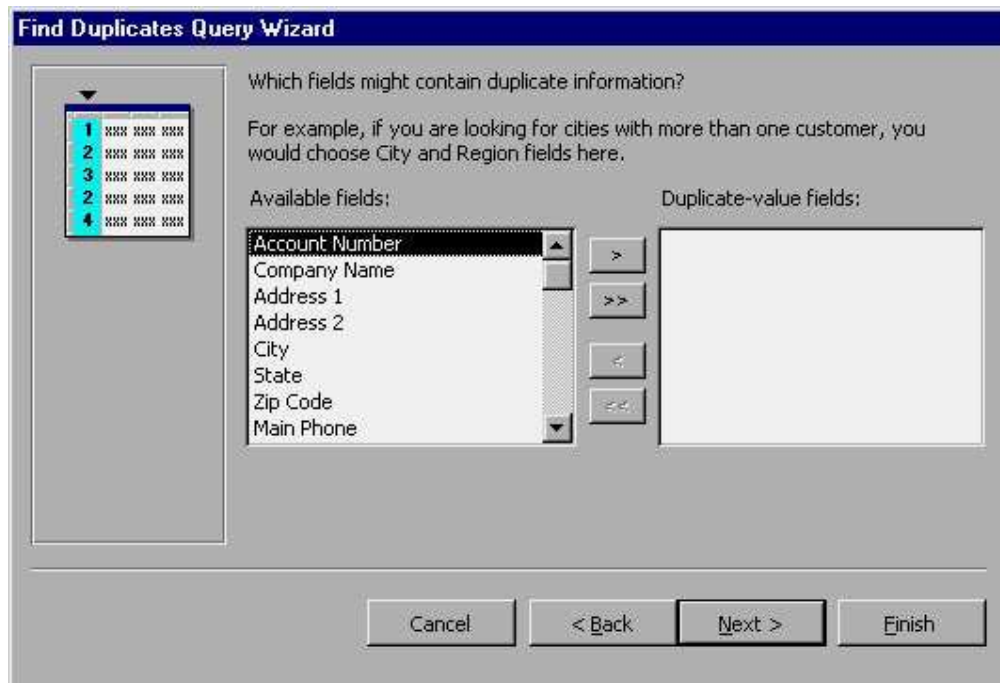
### Tạo một Find Duplicates Query

Bạn cũng có thể sử dụng hộp thoại New Query để tạo một truy vấn Find Duplicates, nó cho phép bạn nhanh chóng tìm kiếm một bảng có các trường trùng nhau. Chọn **Find Duplicates Query Wizard** trong hộp thoại và kích chuột vào nút **OK**. Lần lượt, Access hiển thị hộp thoại Find Duplicates Query Wizard như hình 14.



Hình 14: Hộp thoại Find Duplicates Query Wizard.

Trong hộp thoại này, Access liệt kê các nguồn dữ liệu bạn có thể sử dụng cho truy vấn. Thông thường, bạn sử dụng một bảng như nguồn dữ liệu nhưng với Wizard này bạn cũng có thể sử dụng truy vấn khác nữa. Bởi vì bảng Business Customers là duy nhất trong cơ sở dữ liệu Customers, Access đã lựa chọn bảng này cho bạn. Kích chuột vào nút **Next** để chuyển tới hộp thoại như hình 15.



Hình 15: Chọn một trường để kiểm tra.

Trong hộp thoại này, Access hỏi bạn những trường nào cần kiểm tra giá trị trùng. Mặc dù không có giới hạn về số các trường bạn có thể có trong kiểm tra truy vấn, nhưng bạn cũng chỉ nên chọn những trường cần thiết để chắc chắn là không có sự trùng lặp. Trong trường hợp của bảng Business Customers, chỉ cần kiểm tra trường *Company Name* và *Contact* để xác định nếu có sự trùng lặp.

Đầu tiên, chọn trường *Company Name* và kích chuột vào nút **>**. Tiếp theo, chọn trường *Contact* và kích tiếp vào nút **>**. Access hiển thị hai trường theo thứ tự trong danh sách phía bên phải hộp thoại. Sau khi xác định xong các trường, kích chuột vào nút **Next**. Access hiển thị hộp thoại như hình 16.



Hình 16: Hiển thị thông tin thêm.



Trong hộp thoại này, Access hỏi nếu bạn muốn truy vấn hiển thị thêm thông tin. Khi xem lại các bản ghi trùng, bạn có thể sử dụng thông tin này nhằm giúp việc quyết định bản ghi nào muốn lưu và bản ghi nào muốn xóa. Đối với bảng Business Customers, bạn muốn Access hiển thị các trường *Contact Phone*, *Last Order*, *Salesperson*. Chọn một vài trường (nhớ kích nút > sau khi chọn mỗi trường). Access thêm tên của mỗi trường vào danh sách phía bên phải hộp thoại. Khi hoàn thành, kích chuột vào nút **Next**, Access hiển thị hộp thoại như hình 17.



Hình 17: Kết thúc truy vấn.

Lúc này bạn đã hoàn thành việc tạo truy vấn, Access yêu cầu bạn đặt tên cho truy vấn. Từ gợi ý của Wizard cho tên của truy vấn này, kích chuột vào nút **Finish** để chấp nhận tên đó. Access lưu truy vấn dưới tên này và cập nhật danh sách truy vấn trong cửa sổ Database. Access mở truy vấn để hiển thị kết quả vừa hoàn thành như hình 18.

Company Name	Contact	Main Phone	Account Open	Credit Limit	Last Order	Salesperson
Brendon Dairy	Nathan Gehric	(203) 513-2906	4/1/95	\$1,500.00	7/16/99	David Jones
Brendon Dairy	Nathan Gehric	(203) 513-2906	12/15/95	\$1,500.00	3/18/99	David Jones
Bridger Law	Thomas Bridger	(501) 328-1869	2/17/96	\$2,000.00	5/19/99	Jerome Smith
Bridger Law	Thomas Bridger	(501) 328-1869	2/17/96	\$2,000.00	5/19/99	Jerome Smith
Morris Corporat	Greg Crothers	(614) 289-7359	4/8/96	\$1,750.00	8/9/98	John Davis
Morris Corporat	Greg Crothers	(614) 289-7359	4/8/96	\$1,500.00	8/9/98	John Davis
Rain Water Con	Jake Conway	(619) 807-2500	10/16/93	\$1,750.00	3/15/99	Jeffrey Thomas
Rain Water Con	Jake Conway	(619) 807-2500	4/19/99	\$1,500.00	7/2/99	Jeffrey Thomas
Rhymes & Rum	Larry Strong	(302) 218-9465	5/6/94	\$2,000.00	6/27/99	David Jones
Rhymes & Rum	Larry Strong	(302) 218-9465	7/8/95	\$1,500.00	7/8/98	David Jones
Wintergreen Ma	Sam Harrison	(205) 812-9631	4/5/97	\$1,500.00	8/16/99	Luke Jefferson
Wintergreen Ma	Sam Harrison	(205) 812-9631	4/5/97	\$1,500.00	8/16/99	Luke Jefferson
				\$0.00		

Hình 18: Kết quả của truy vấn Find Duplicates.

Bây giờ bạn có thể xem lại các bản ghi trùng lặp và quyết định những việc muốn làm với chúng. Nếu bạn xóa bất kỳ bản ghi nào trong màn hình, Access sẽ xóa bản ghi tương ứng trong bảng phía bên dưới (Business Customers). Với lý do này, truy vấn Find Duplicates rất có ích trong việc quản lý thông tin ở bảng.

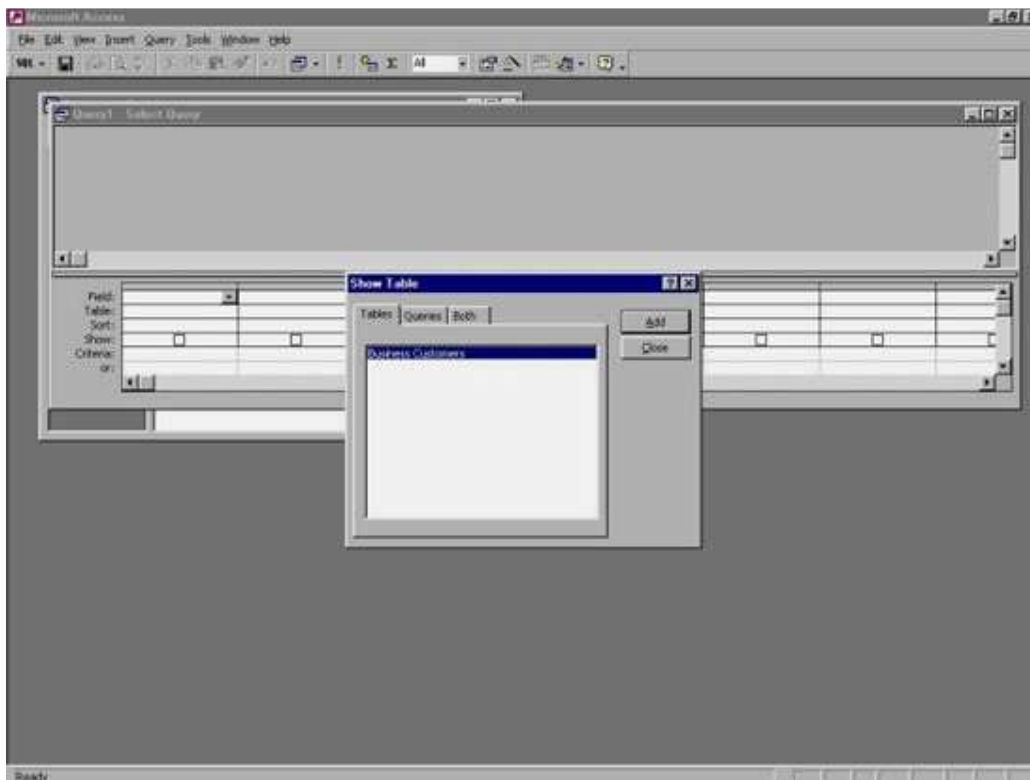
## Bài 23: Thiết kế truy vấn riêng

**Đến lúc này, bạn đã có thể sử dụng Query Wizard để tạo các truy vấn. Nhưng Query Wizard chỉ tạo một vài truy vấn thường dùng. Nếu một trong những truy vấn đó không phù hợp, bạn có thể tạo truy vấn riêng.**

Nhiều người nghĩ rằng tự xây dựng truy vấn riêng là một trong những phần khó nhất khi làm việc với Access. Điều đó có thể đúng, nhưng với một vài hướng dẫn và nhiều lần thực hành, bạn có thể xây dựng bất kỳ truy vấn nào mà bạn nghĩ ra.

Khi tự tạo các truy vấn từ đầu, bạn sẽ được biết các nguyên tắc để có thể áp dụng khi chỉnh sửa truy vấn sẵn có. Với cách như vậy - tạo hay chỉnh sửa - bạn sử dụng cửa sổ thiết kế truy vấn.

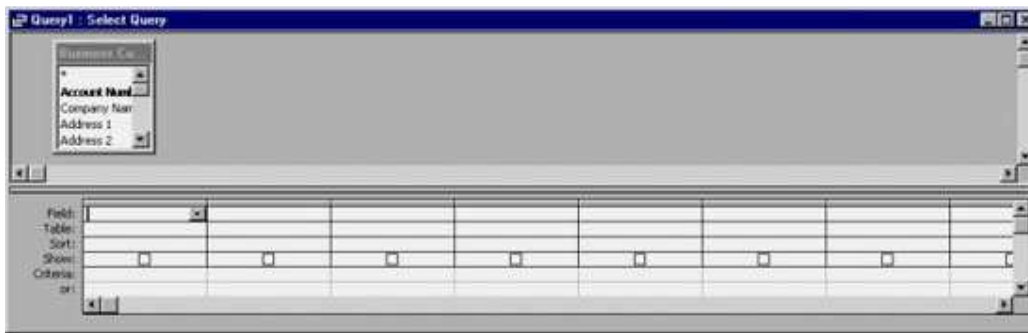
Quá trình sau để tạo một truy vấn, phần lớn dựa vào loại truy vấn mà bạn muốn tạo. Không quan tâm đến loại truy vấn, bạn thường bắt đầu với cùng một cách: chọn nút **Queries** trong cửa sổ Database, kích đúp chuột vào tùy chọn **Create Query in Design View**. Màn hình xuất hiện như hình 1.



Hình 1: Tạo truy vấn

Chú ý rằng cửa sổ thiết kế truy vấn là phần sau. Trong phần đầu, Access hiển thị hộp thoại Show Table đưa ra cho bạn lựa chọn nguồn dữ liệu muốn sử dụng. Như các truy vấn khác, bạn có thể lựa chọn cả bảng hoặc các truy vấn làm nguồn dữ liệu (đều có ở hộp thoại Show Table). Trong trường hợp này, chọn bảng *Business Customers* và kích chuột vào nút **Add**. Access mở một cửa sổ nhỏ cho bảng đó có chứa các trường. Để đóng hộp thoại Show Table, kích chuột vào nút **Close**.

Sau khi đóng hộp thoại Show Table, Access hiển thị cửa sổ thiết kế truy vấn. Ở phần giữa cửa sổ là chỗ trống cho các nguồn dữ liệu đã được mở. Trong trường hợp này, chỗ trống chứa bảng *Business Customers*. Bên dưới là lưới thiết kế OBE (ví dụ cho truy vấn). Cửa sổ thiết kế truy vấn xuất hiện như hình 2.



Hình 2: Cửa sổ thiết kế truy vấn

Trong lưới thiết kế QBE, bạn xác định các tiêu chuẩn cho truy vấn. Những tùy chọn sẵn có trong lưới phụ thuộc vào loại truy vấn bạn tạo.

### Công cụ thiết kế truy vấn

Công cụ thiết kế truy vấn chỉ xuất hiện bên dưới thực đơn bất cứ khi nào bạn sử dụng cửa sổ thiết kế truy vấn. Nhiều công cụ trên thanh công cụ rất quen như đã sử dụng trong các cửa sổ Access khác. Tuy nhiên, có các công cụ khác là chỉ có trên thanh công cụ của truy vấn Design. Bảng 1 hiển thị những công cụ đó và công việc chúng thực hiện.

Công cụ	Tên	Tính năng
	Query Type	Cho phép xác định loại truy vấn bạn tạo
	Run	Thi hành truy vấn nếu cần thiết.
	Show Table	Hiển thị hộp thoại Show Table
	Totals	Hiển thị dòng tổng trong lưới QBE. Nó không sẵn có khi tạo một truy vấn cập nhật hay xóa truy vấn
	Top Values	Tìm giá trị cao nhất trong một truy vấn dựa vào tỷ lệ phần trăm hay số các dòng
	Properties	Hiển thị cửa sổ Properties, cho các trường truy vấn hoặc cho chính các truy vấn
	Build	Hiển thị cửa sổ Builder khi tạo tiêu chuẩn cho một trường.

Bảng 1: Các công cụ thiết kế

Bạn thường sử dụng Query Wizards để tạo các truy vấn bình thường, có nghĩa là sử dụng cửa sổ thiết kế truy vấn cho những truy vấn phổ biến hơn. Do vậy, bạn sẽ thường sử dụng công cụ Query Type khi bắt đầu phần thiết kế để cho biết loại truy vấn bạn tạo. Nếu kích chuột vào mũi tên xuống phía bên phải công cụ, bạn có thể thấy Access cho phép tạo những loại truy vấn sau:

- **Select Query (truy vấn lựa chọn):** Là loại truy vấn được mặc định, cho phép bạn xây dựng một truy vấn mà trả về là số các trường được xác định tiêu chuẩn.
- **Crosstab Query (Truy vấn Crosstab):** Sửa đổi lưới QBE sao cho những tùy chọn của nó đại diện sự cần thiết cho truy vấn.

- **Make-Table Query (Truy vấn tạo bảng):** Áp dụng khi cần tạo ra một bảng mới. Access sử dụng bảng này để lưu trữ kết quả phần chính của truy vấn. Khi chọn công cụ này, Access yêu cầu bạn đặt tên cho bảng và nơi lưu trữ bảng. Lưới QBE cũng tương tự như truy vấn chọn lựa.
- **Update Query (Truy vấn cập nhật):** Dùng để cập nhật nhiều bản ghi trong một bảng. Access thay đổi lưới QBE để đại diện cho sự cần thiết của truy vấn.
- **Append Query (Truy vấn Bổ sung):** Dùng để bổ sung các bản ghi thành một bảng. Khi chọn công cụ này, Access yêu cầu bạn đặt tên bảng và nơi lưu trữ bảng đó. Access thay đổi lưới QBE để phản ánh lại những tiêu chuẩn cho loại truy vấn này.
- **Delete Query (Truy vấn xóa):** Dùng để xóa các bản ghi. Access thay đổi lưới QBE thành dạng đơn giản hơn các truy vấn khác.

Những truy vấn này trông rất quen thuộc, về cơ bản, chúng giống như những truy vấn đã được đề cập ở phần đầu chương này.

### Sử dụng lưới QBE

Bất kỳ bạn chọn loại truy vấn nào, thường sử dụng lưới QBE để thiết kế truy vấn. Bạn có thể áp dụng các mục trong lưới cũng giống như cách dùng ở các lưới thiết kế khác. Ví dụ, Ở chương 5 "*Sắp xếp và phân loại thông tin*", bạn đã biết được cách có thể sử dụng lưới cửa sổ Filter. Lưới QBE cũng thực hiện như vậy, bạn có thể thêm, xóa, và di chuyển các cột. Ngoài ra bạn cũng có thể lựa chọn ô trong lưới và chỉnh sửa chúng.

Ý nghĩa của các dòng trong lưới QBE thay đổi phụ thuộc vào loại truy vấn bạn đang thiết kế. Tuy nhiên, ở dòng đầu tiên luôn luôn chứa tên trường. Mỗi cột ở dòng này đại diện cho trường mà Access sẽ hiển thị trong kết quả truy vấn. Do vậy, nếu bạn muốn hiển thị 5 trường trong kết quả truy vấn, bạn nên sử dụng 5 cột trong lưới QBE. Access hiển thị các trường theo thứ tự bạn đặt trong lưới. Tuy nhiên, bạn có thể thiết lập dòng Show cho mỗi trường quản lý dù Access có hiển thị các trường hay không (Hình 3)



Hình 3: Lưới QBE

Nếu bạn chọn dòng Show cho một cột riêng, Access hiển thị trường đó ra kết quả. Nếu không chọn dòng Show, Access không hiển thị trường. Tại sao bạn không muốn hiển thị trường đó? Đơn giản, nếu bạn chỉ sử dụng một cột để lựa chọn hoặc sắp xếp các bản ghi, bạn không cần hiển thị nó. Ví dụ, nếu bạn chọn các bản ghi dựa vào giá trị của trường là Yes/No (chỉ chọn những trường được thiết lập Yes), như bạn biết rằng mọi giá trị mà kết quả trường sẽ là (-1). Do vậy, đó thực sự không cần để hiển thị trường. Có thể sử dụng dòng Show để ẩn hoặc hiển thị các trường như bạn muốn.

**Chú ý: Dòng Show không áp dụng với các truy vấn Crosstab, Cập nhật, Bổ sung và Xóa.**

Dòng Sort và Criteria làm việc tương tự như đã đề cập về các filter ở chương 5 "*Sắp xếp và lọc thông tin*". Nếu bạn vẫn chưa rõ về vấn đề này, có thể xem lại chương đó.

Cách tốt nhất để khám phá ra cách sử dụng lưới QBE là thực hành. Giả sử bạn muốn tạo một truy vấn sử dụng bảng Business Customers và hiển thị tất cả các tài khoản còn hoạt động với thẻ giới hạn trên 1000\$ của một trong những nhân viên bán hàng. Thêm vào đó, bạn muốn các tài khoản được liệt kê sắp xếp theo thứ tự giảm dần dựa vào thời hạn thẻ của họ (đối với những tài khoản với thời hạn thẻ bằng nhau, bạn muốn chúng được sắp xếp theo thứ tự tên công ty). Để thiết kế truy vấn như vậy, thực hiện các bước sau:

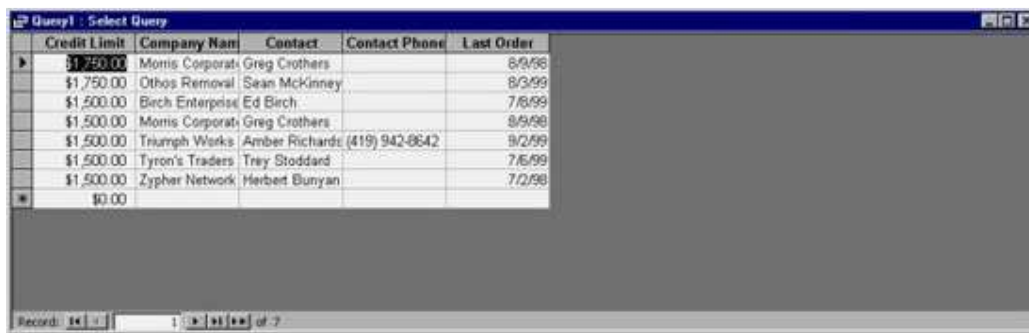
1. Bắt đầu một lưới QBE trống.
2. Sử dụng công cụ Query Type để chắc chắn **Select Query** được thiết kế.
3. Trong danh sách các trường, kích đúp chuột vào tên trường **Active**. Access hiển thị tên trường này ở cột đầu tiên của lưới QBE.
4. Ở dòng **Criteria** đối với cột đầu tiên, nhập phép tính **=Yes**. Điều này có nghĩa Access chỉ ra những bản ghi với trường Active được thiết lập là đúng.
5. Bỏ dấu tích ở dòng **Show** ở cột đầu tiên. Access không hiển thị giá trị của trường.
6. Trong danh sách trường, kích đúp chuột vào tên trường **Salesperson**. Access hiển thị tên trường này ở cột thứ hai trong lưới QBE.
7. Ở dòng **Criteria** đối với cột thứ hai, nhập phép toán **= "John Davis"**. Điều này có nghĩa bạn muốn chỉ những bản ghi có đại diện người bán hàng tên là John Davis.
8. Xóa dấu tích ở dòng **Show** cho cột thứ hai. Access sẽ không hiển thị giá trị của trường.
9. Kích đúp chuột vào tên trường **Credit Limit**. Access hiển thị trên trường này trong cột thứ 3 của lưới QBE.
10. Trong dòng **Sort** ở cột thứ 3, chọn **Descending**.
11. Trong dòng **Criteria** của cột thứ 3, nhập đẳng thức **>1000**. Điều này cho biết bạn chỉ muốn những bản ghi với thẻ giới hạn trên 1000\$.
12. Kích đúp chuột vào tên trường **Company Name**. Access hiển thị trường này ở cột 4 trong lưới QBE.
13. Ở dòng **Sort**, chọn **Ascending**.
14. Kích đúp chuột vào tên trường là **Contact**. Access hiển thị trường này ở cột thứ 5 trong lưới QBE.
15. Kích đúp chuột vào tên trường là **Contact Phone**. Access hiển thị trường này ở cột thứ 6 trong lưới QBE.
16. Kích đúp chuột vào tên trường **Last Order**. Access hiển thị tên trường này ở cột thứ 7 trong lưới QBE.

Lúc này bạn đã hoàn thành xong truy vấn. Cửa sổ thiết kế truy vấn của bạn được đưa ra như hình 4.



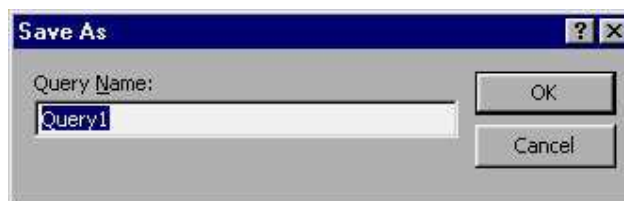
Hình 4: Truy vấn được hoàn thành

Truy vấn này hiển thị 5 trường từ bảng Business Customers, nhưng chỉ đối với những trường có điều kiện ở dòng Criteria. Bạn có thể lưu truy vấn ngay nếu muốn. Tuy nhiên, để thấy kết quả của truy vấn luôn, kích chuột vào công cụ **Run** trên thanh công cụ. Access chạy truy vấn mà không cần lưu nó và hiển thị ra kết quả. Cách này bạn có thể kiểm tra lại để thấy được nếu kết quả như mong đợi. Hình 5 hiển thị kết quả của truy vấn mới.



Hình 5: Kết quả truy vấn.

Khi hoàn thành kiểm tra kết quả truy vấn, bạn có thể sử dụng công cụ **View** (phía bên trái thanh công cụ) để chuyển đổi trở lại cửa sổ thiết kế truy vấn. Sau đó, lưu truy vấn bằng cách đóng cửa sổ thiết kế. Kích chuột vào nút **Yes** và Access đưa ra lời nhắc bạn cần đặt một tên (hình 6).



Hình 6: Bạn có thể lưu truy vấn sử dụng tên bạn muốn.

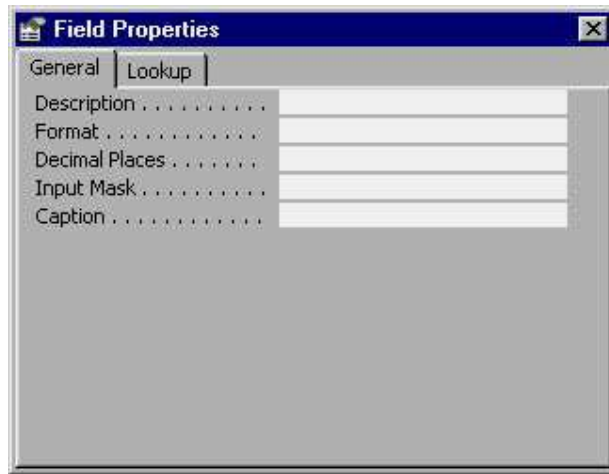
Nhập tên bạn muốn để sử dụng cho truy vấn, khi kích chuột vào nút **OK**, Access lưu truy vấn và cập nhật vào cửa sổ Database để thêm tên mới.

### Thiết lập các thuộc tính

Trong chương 3 "*Tùy biến các thành phần*", bạn đã học về các thuộc tính của trường và cách chúng tác động đến chế độ Datasheet. Các thuộc tính được miêu tả ở chương 3 áp dụng cho các trường trong một bảng nhưng bạn cũng có thể thiết lập các thuộc tính cho trường ở trong một truy vấn. Các thuộc tính truy vấn tác động đến cách Access hiển thị thông tin xuất phát từ truy vấn cũng như cách bạn có thể chỉnh sửa thông tin đó.

Để thấy được các thuộc tính cho một trường cụ thể, lựa chọn trường từ lưới QBE. Sau đó, kích chuột vào công cụ **Properties**. Access hiển thị cửa sổ Properties, xuất hiện các hình thức của

cửa sổ phụ thuộc vào kiểu dữ liệu của trường. Một cửa sổ properties mẫu như hình 7 (Các thuộc tính cho trường Credit Limit)



Hình 7: Cửa sổ Properties cho trường Credit.

Chú ý rằng, có một vài thuộc tính mà bạn thấy trong bảng cửa sổ thiết kế. Tuy nhiên, các thuộc tính được hiển thị có chức năng gần giống với cách mà chúng thực hiện khi làm việc với bảng. (Bạn có thể trở lại chương 3 để xem lại).

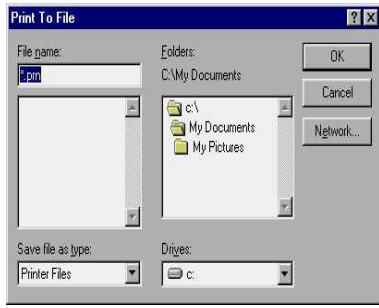
**Truy vấn là những công cụ rất mạnh, bạn có thể sử dụng nó để lựa chọn, thay đổi và thao tác với thông tin trong cơ sở dữ liệu. Từ một nguồn dữ liệu (bảng hay các truy vấn khác), một truy vấn chọn lựa các bản ghi theo các tiêu chuẩn đã được xác định. Access sẽ làm gì với những bản ghi đó, phụ thuộc vào loại truy vấn bạn thiết kế.**

Trong chương này, chúng ta đã tìm hiểu các truy vấn cơ bản, và một số các truy vấn khác. Khi bạn kết hợp thông tin ở chương này với những gì đã học ở các chương trước, bạn sẽ nhận thấy rằng các truy vấn không hề khó hiểu hay phức tạp như bạn nghĩ lúc đầu.

Trước khi chuyển sang chương 7, hãy chắc chắn là bạn hiểu được những nội dung sau:

- Một truy vấn là một cách để bạn đặt câu hỏi cho Access về thông tin trong các bảng.
- Access cho phép bạn tạo năm loại truy vấn: Truy vấn lựa chọn (Select), Truy vấn hành động (Action), Truy vấn Crosstab, Truy vấn tham số (Parameter), và Truy vấn SQL
- Các truy vấn là phần khác của cơ sở dữ liệu, giống như các bảng, báo cáo, và macro. Bạn có thể sử dụng nút Queries trong cửa sổ Database để xem các loại truy vấn trong cơ sở dữ liệu.
- Bạn có thể sử dụng Query Wizard để tạo 4 loại truy vấn khác. Đó là các instructive (trong những gì truy vấn đã hoàn thành) và là nơi bắt đầu cho các truy vấn riêng.
- Bạn có thể sử dụng cửa sổ thiết kế để thiết kế các truy vấn riêng từ sự hỗn tạp. Ở cửa sổ này, cùng với thanh công cụ thiết kế truy vấn, cho phép bạn chỉ định và tạo nhiều loại truy vấn.
- Lưới QBE xuất hiện ở phía dưới của cửa sổ thiết kế truy vấn. Sử dụng lưới, bạn có thể chỉ định những trường Access sử dụng để quyết định nếu các bản ghi được hiển thị, cũng như thêm các bản ghi mà Access sẽ hiển thị trong kết quả truy vấn.
- Khi thiết kế một truy vấn, bạn có thể chỉ định thuộc tính cho các trường trong kết quả truy vấn. Những thuộc tính của trường tương tự như các thuộc tính khi thiết kế cho bảng.

# Bài 25: In bảng



Như đã đề cập, sau khi nhập dữ liệu vào bảng, bạn muốn in một bản sao nội dung bảng. Access thực hiện việc đó một cách rất đơn giản để in dữ liệu trong bảng. Bạn có thể in bảng theo các bước sau đây:

1. Chọn nút **Tables** trong cửa sổ Database.
2. Chọn tên bảng mà bạn muốn in

3. Lựa chọn tùy chọn **Print** từ thực đơn **File** hoặc kích chuột vào nút Print trên thanh công cụ.

Cách khác để in bảng là thực hiện theo các bước sau:

1. Chọn nút **Tables** trong cửa sổ Database
2. Chọn (đánh dấu) tên bảng bạn muốn in.
3. Kích chuột vào nút **Open**. Access hiển thị bảng ở chế độ **Datasheet**; bạn có thể thực hiện các thay đổi cho bảng.
4. Lựa chọn tùy chọn **Print** từ thực đơn **File** hoặc kích chuột vào công cụ Print trên thanh công cụ

Nếu bạn kích chuột vào công cụ Print trên thanh công cụ, Access sẽ in bảng ngay. Nếu chọn tùy chọn Print từ thực đơn File, Access hiển thị hộp thoại Print như hình 1.



Hình 1: Hộp thoại Print

Các phần riêng biệt của hộp thoại Print điều khiển những gì Access in. Nếu muốn điều chỉnh máy in nào mà Access sử dụng để in bảng, sử dụng danh sách **Name** xổ xuống ở phía trên cùng của hộp thoại. Nếu muốn thay đổi các thuộc tính riêng của máy in, kích chuột vào nút **Properties**. Lần lượt, Access hiển thị hộp thoại Properties của máy in (sẽ được miêu tả ngay sau chương này).

Vùng **Print Range** cho phép bạn chọn các phần của bảng mà Access in. Bạn có thể in toàn bộ bảng, một phần của bảng (các dòng hay cột mà bạn chọn) hay một vùng của trang. Ngay sau chương này, bạn sẽ học cách xem trước dữ liệu trên màn hình trước khi in. Sử dụng tùy chọn

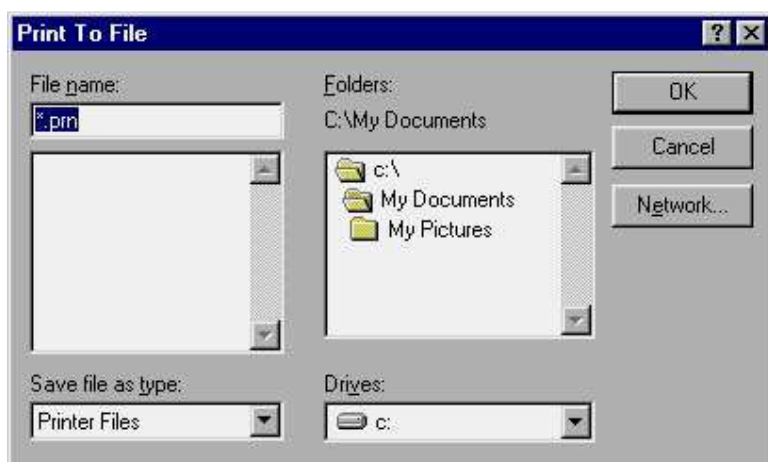


**Print preview**, bạn phải quyết định là in một vùng cụ thể của trang. Tùy chọn **Pages** cho phép bạn xác định một vùng của trang in.

Vùng **Copies** cho phép bạn yêu cầu Access tạo bao nhiêu bản sao của thông tin cần in. Hộp chọn **Collate** chỉ có nghĩa nếu bạn in nhiều hơn một bản sao của bảng. Ví dụ, giả sử bảng của bạn cần in 5 trang. Nếu cần hai bản sao và bạn tích chọn hộp Collate, Access sẽ in tất cả 5 trang bảng (lần 1) và sau đó in lại 5 trang (lần 2). Nếu bạn không chọn hộp Collate Copies, Access in lần lượt hai bản sao của mỗi trang (Trong trường hợp này, bạn sẽ cần phải sắp xếp lại các bản sao sau đó).

Cuối cùng, ở phần giữa phía bên phải hộp thoại Print là hộp chọn **Print to File**. Hộp chọn này cho phép bạn chuyển dữ liệu từ máy in thành một tập tin trên đĩa. In vào đĩa thực sự rất cần thiết. Ví dụ, bạn cần gửi một báo cáo cho một ai đó ở xa. Bạn có thể in nó và fax cho họ hay gửi qua hệ thống thư điện tử. Tuy nhiên, nếu bạn in báo cáo thành một tệp, bạn có thể thay vì gửi cho họ đĩa có chứa tập tin, hoặc gửi thư điện tử in tệp tin cho họ. Nếu báo cáo rất lớn, việc gửi một thư điện tử hay đĩa có thể tiết kiệm rất nhiều chi phí trong việc vận chuyển. Kết quả, họ nhận được thông tin rất nhanh và chi phí rất rẻ. Khi họ nhận tệp tin, họ có thể chuyển nó tới máy in của họ và có một bản sao giống như bản sao bạn đã in cho họ.

Khi đã kết thúc, kích chuột vào nút **OK**. Access sẽ gửi thông tin của bạn tới máy in theo cách của bạn. Nếu bạn lựa chọn hộp **Print to File**, Access sẽ hỏi bạn tên tệp để sử dụng như hình 2.



Hình 2: Hộp thoại Print to File.

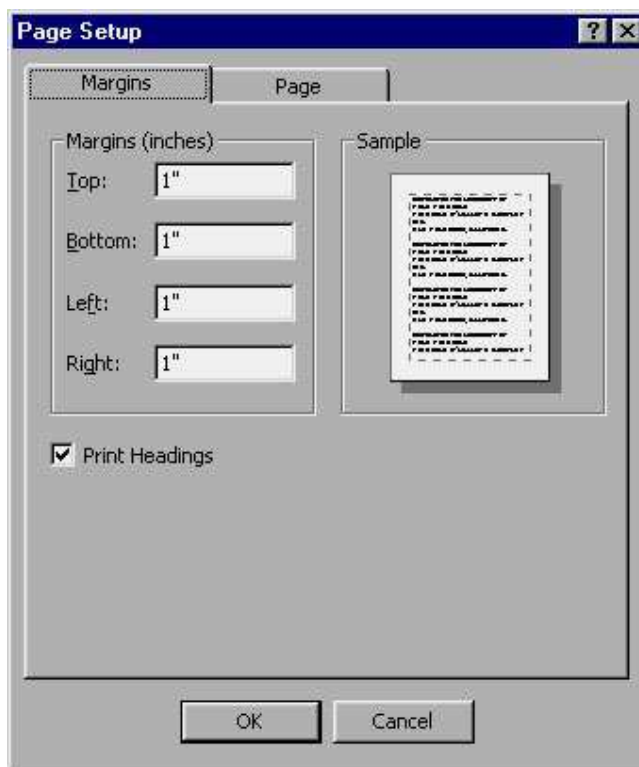
Đây là một hộp thoại điều khiển tệp tin chuẩn. Bạn có thể sử dụng sự điều chỉnh mà nó cung cấp để lựa chọn ổ và thư mục nơi chứa tệp tin. Tất cả việc cần làm là nhập tên tệp muốn và kích chuột vào nút **OK**. Access sẽ gửi đầu ra của máy in vào đĩa tệp tin bạn chỉ định.

Nếu máy tính của bạn có quyền truy cập tới nhiều máy in hay nếu bạn cần kiểm soát dữ liệu in (có lẽ bằng cách thay đổi hướng và lề trang in), bạn có thể sử dụng hộp thoại Print Setup được nói đến ở phần tiếp theo.

## Bài 26: Điều chỉnh trang in

**Giống như nhiều chương trình, Access cung cấp đầy đủ các tính năng của môi trường Window. Access sử dụng các tiện ích in được cung cấp bởi môi trường Windows, có nghĩa là nó có thể sử dụng bất kỳ máy in nào Windows sử dụng.** Bạn không thể thêm hay xóa các máy in trong Access (nó đã có sẵn trong Windows), nhưng có thể điều chỉnh máy in nào Access gửi thông tin và những tính năng máy in mà Access sẽ sử dụng. Để điều chỉnh các thao tác in, sử dụng hộp thoại Page Setup.

Bạn có thể mở hộp thoại Page Setup từ nhiều cách khác nhau trong Access. Bất cứ lúc nào sẵn sàng in, một công cụ hoặc thực đơn tùy chọn cho phép bạn xem hộp thoại Page Setup. Trong phần lớn các trường hợp, bạn sẽ mở hộp thoại bằng cách chọn tùy chọn Page Setup từ thực đơn File. Hộp thoại Page Setup hiển thị như hình 1.



Hình 1: Hộp thoại Page Setup.

Chi tiết của các phần trong hộp thoại:

### **Thiết lập Lề trang in (Printout Margins)**

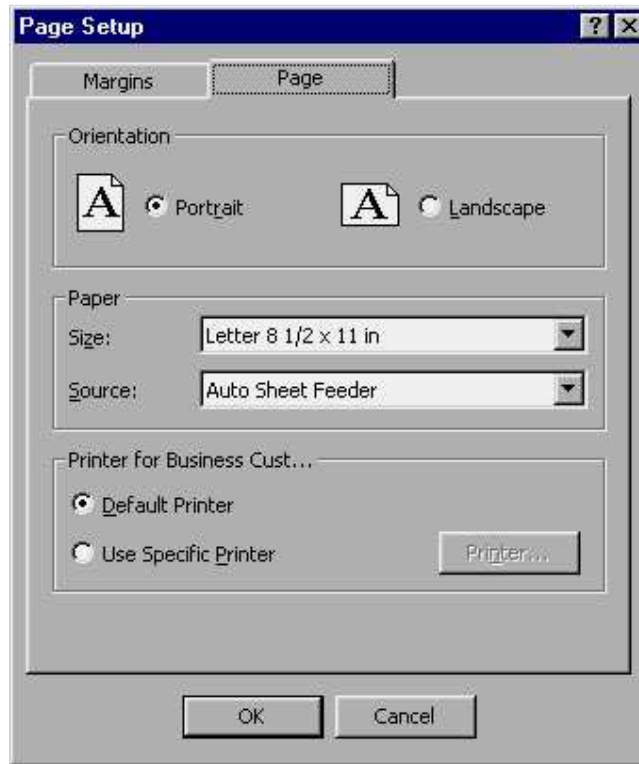
Thẻ đầu tiên trong hộp thoại Page Setup là Margins (hình 1). Dựa vào dữ liệu in, bạn có thể thay đổi mặc định lề trang. Lề trang là khoảng trắng bên trái mỗi trang giấy. Thẻ Margins của hộp thoại Page Setup cho phép bạn điều chỉnh lề bên trái của trang in là bao nhiêu. Để thiết lập một lề trang, chỉ cần thay đổi thiết lập đã có ở một trong 4 margins.

Mặc định các lề là 1 inch trên mỗi trang giấy. Bạn có thể thay đổi các margins này, chỉ cần chọn hộp có margin bạn muốn thay đổi và nhập một giá trị mới.

**Chú ý:** Một vài máy in yêu cầu khoảng cách lề tối thiểu. Ví dụ, một số máy in laser chỉ in những nội dung nằm cách lề giấy 1/4 inch. Trong trường hợp này, Access không cho phép bạn thay đổi các lề ít hơn giới hạn đã định của

## Thiết lập trang

Sử dụng thẻ Page của hộp thoại Page Setup (hình 2) bạn có thể thay đổi cách Access thường đặt thông tin trên trang.



Hình 2: Thẻ Page của hộp thoại Page Setup.

**Chú ý:** Tùy chọn sẵn có trong hộp thoại Page Setup phần lớn phụ thuộc vào loại máy in. Nếu một số tùy chọn không hiện hữu, đừng lo bởi vì máy in bạn đang sử dụng có thể có các tùy chọn mà nhiều người dùng khác không có.

Thiết lập đầu tiên bạn có thể chỉ định trong hộp thoại là định hướng trang giấy. Thực tế, đây là một thuật ngữ sai. Bản chất trang giấy là không được định hướng, đó là hình ảnh tạo bởi Access mà được định hướng trong mỗi tương quan trang giấy. Để chọn định hướng nằm ngang hay dọc, sử dụng thiết lập trong vùng Orientation ở phía trên cùng hộp thoại.

Tài liệu thường in mặc định là để giấy dọc. Điều này có nghĩa khi bạn đọc trang giấy, lề thu hẹp là phía trên và phía dưới của trang. Nếu bạn quay trang 90o về phía bên trái hay bên phải, trang giấy sẽ ở chế độ nằm ngang – lề thu hẹp về phía bên trái và bên phải trang. Chế độ quay ngang có ích khi bạn in các bảng có nhiều cột.

Nếu máy in của bạn hỗ trợ nhiều kích thước giấy, chẳng hạn như các tiêu chuẩn và trang tính hợp lý, bạn chỉ cần chọn cỡ giấy bạn muốn. Để chọn một cỡ giấy, thay đổi thông tin ở vùng Paper của hộp thoại. Dựa vào máy in, bạn có thể thay đổi cả kích thước và nguồn trang giấy (hình 2). Bạn nên chọn kích thước và nguồn giấy mà thực hiện phần lớn ý nghĩa của kết quả bạn muốn tạo.

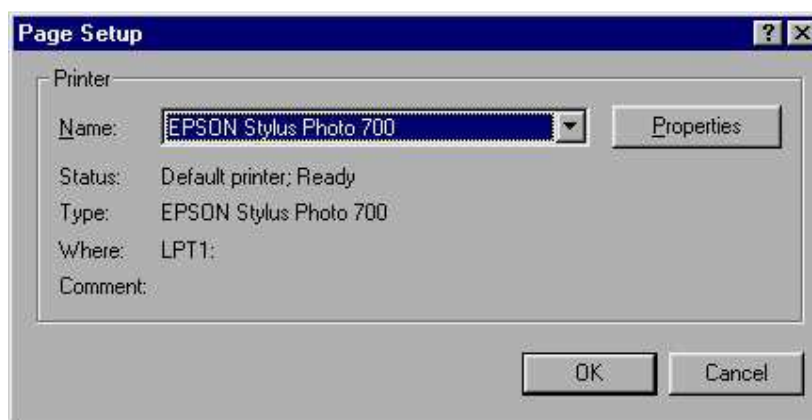
## Thay đổi máy in

Trình điều khiển máy in là phần mềm đặc biệt mà Windows sử dụng để giao tiếp với máy in. Phần lớn các máy in phụ thuộc vào phần mềm đặc biệt của riêng chúng (nói cách khác, là trình điều khiển – driver máy in riêng).

Hiếm có máy tính nào được cài đặt nhiều driver máy in trong cùng một hệ thống. Nhiều người có ít nhất hai máy in, chẳng hạn như máy in mực và máy in laser. Một số khác thì sử dụng máy in trên máy tính cùng mạng. Ví dụ, nếu một người nào đó trong công ty của bạn có một máy in laser phân giải cao, bạn phải có một driver máy in được cài đặt trên máy cho máy in đó nếu không thì không thể truy cập trực tiếp vào máy in. Trong trường hợp đó, bạn có thể xuất thông tin ra thành một tệp tin và gửi tệp tin đó tới máy in khác để in thông tin.

Access cho phép bạn lựa chọn bất kỳ máy in nào có trình điều khiển đã cài đặt trong Windows. Tóm lại, lựa chọn một driver máy in cho biết loại máy in bạn muốn sử dụng. Chọn một driver máy in tại vùng Printer trong thẻ Page của hộp thoại Page Setup. (hình 2).

Thông thường chọn thiết lập Default Printer. Với thiết lập này, Access sử dụng máy in bạn đã thiết lập như máy in mặc định trong Windows. Tuy nhiên, bạn có thể chọn máy in khác bằng cách tích vào tùy chọn Use Specific Printer. Sau đó, nút Printer sẽ được kích hoạt và bạn có thể sử dụng nó để chọn máy in bạn muốn. Khi kích chuột vào nút Printer, Access hiển thị một hộp thoại Page Setup mới cho phép bạn xác định máy in bạn muốn dùng. Hình 3 biểu thị về hộp thoại này.



Hình 3: Hộp thoại Page Setup

Danh sách xổ xuống phía trên cùng hộp thoại hiển thị tên các máy in đã kết nối với máy tính của bạn, cũng như tên các driver máy in bạn đã cài đặt. Không cần để ý tới máy in bạn chọn, Access sẽ nhớ sự chọn lựa của bạn và sử dụng máy in đó cho tới khi bạn chọn máy in khác.

Bạn đã được lưu ý rằng, hộp thoại Print Setup thứ hai này (hình 3) tương tự như phần trên cùng hộp thoại Print (hình 1). Access cho phép bạn xác định máy in nào bạn muốn sử dụng cả hai vị trí. Thay đổi máy in trong một hộp thoại cũng giống như thay đổi máy in trong cả hai vị trí.

**Chú ý:** Thay đổi máy in sẽ ảnh hưởng tới tùy chọn sẵn có trong vùng Page Setup của Access. Thông thường, Access chỉ hiển thị các thiết lập tùy chọn áp dụng cho máy in hiện hành. Không nên lo lắng nếu hộp thoại thiết lập của bạn thay đổi khi máy in thay đổi.

Khi hoàn thành việc xác định loại máy in bạn sử dụng, kích chuột vào nút OK. Sau đó có thể tiếp tục thiết lập các tùy chọn in khác.

## Điều chỉnh các thuộc tính máy in

Có hàng trăm máy in khác nhau trên thị trường. Đó là lý do mà Windows đòi hỏi driver máy in để giao tiếp với một máy in cụ thể. Vấn đề phức tạp là mỗi loại máy in đều có những khả năng khác nhau. Ví dụ, các máy in khác nhau sẽ điều khiển các loại trang giấy khác nhau. Một số máy in có thể in màu. Các máy in khác đưa ra các tùy chọn khác nhau để in đồ họa.

Access cho phép bạn thay đổi các thuộc tính xác định được sử dụng bởi một máy in. Bạn có thể thực hiện điều này bằng cách kích chuột vào nút Properties trong hộp thoại Print (hình 1) hoặc hộp thoại Page Setup (hình 3). Access hiển thị hộp thoại Properties của máy in. Hình thức hiện tại của hộp thoại này hoàn toàn khác nhau, phụ thuộc vào máy in bạn sử dụng. Hình 4 hiển thị ví dụ của hộp thoại Properties cho máy in Epson Stylus Photo 700.



Hình 4: Hộp thoại Properties cho máy in Epson Stylus Photo 700.

Máy in phổ biến này là loại máy in phun mực màu. Do vậy, các điều chỉnh trong hộp thoại Properties được sử dụng để xác định màu nào nên in trên giấy. Các điều chỉnh khác cho phép bạn xác định việc cân bằng giữa chất lượng và tốc độ. Các thẻ khác cho phép bạn chỉ ra các cách khác nhau mà máy in nên sử dụng.

Nhằm thể hiện sự tương phản, hãy nhìn vào hình 5. Nó hiển thị thuộc tính Properties cho máy in HP LaserJet. Chú ý rằng, sự điều chỉnh trong hộp thoại này hoàn toàn khác với sự điều chỉnh trong hình 4. Bởi vì khả năng của hai máy in hoàn toàn khác nhau.



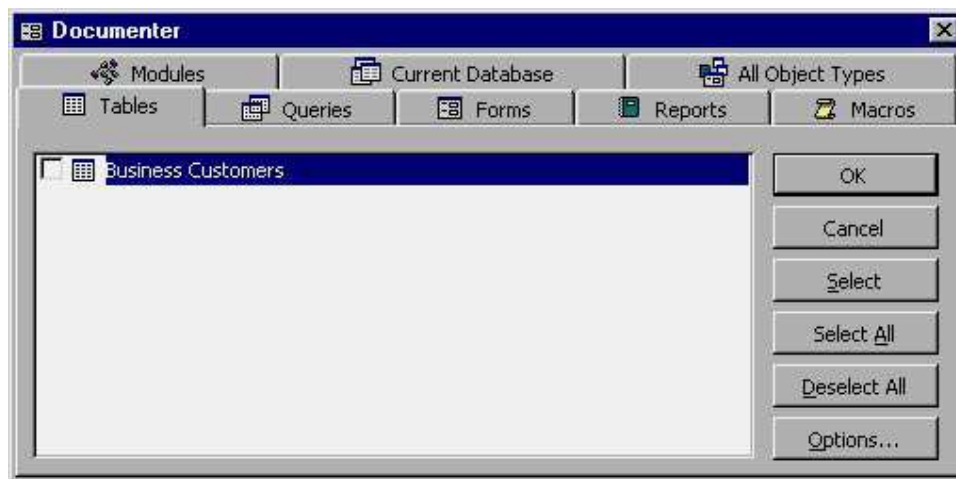
Hình 5: Hộp thoại Properties cho máy in HP LaserJet.

Sự điều chỉnh trong hộp thoại Properties của máy in HP cho phép bạn đặt nhiều hình ảnh trên một trang tính đơn của trang in và điều chỉnh các tính năng của máy in khác.

Phần lớn thời gian của bạn không cần quan tâm với các thiết lập trong hộp thoại Properties của máy in. Thay vào đó, bạn thường sử dụng những thiết lập mặc định được đưa ra bởi Windows. Bạn nên hiểu rằng hộp thoại luôn sẵn có, tuy nhiên, sử dụng nó là cách chính để tận dụng các lợi thế của tính năng máy in. Cách tốt nhất để tìm ra những điều khiển nào thực hiện trong hộp thoại Properties của máy in là thử nghiệm. Bạn không nên sợ làm hỏng máy in, bạn có thể khám phá nhiều điều thú vị và cách sử dụng các tính năng sẵn có trong máy in.

## Bài 27: In ấn thông tin cho các đối tượng

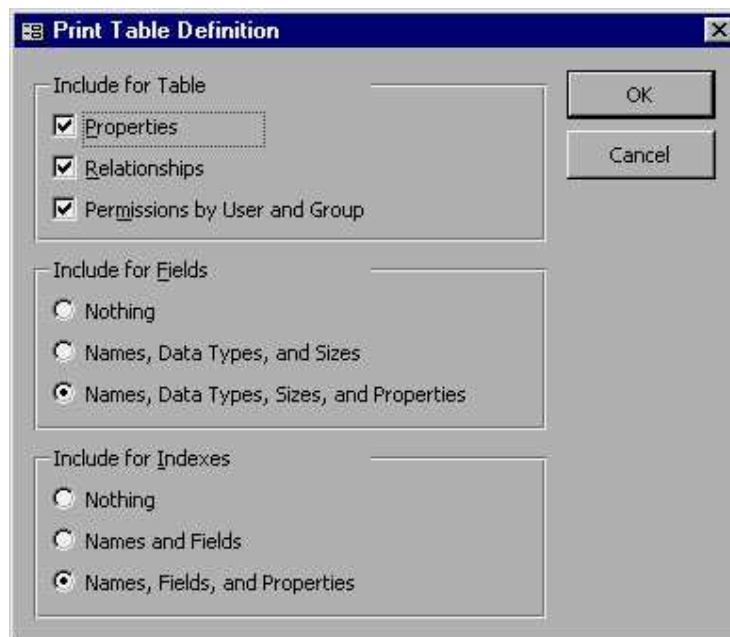
Access cho phép bạn in ấn sự bố trí các bảng hay bất kỳ đối tượng cơ sở dữ liệu bạn đã định rõ (như các truy vấn, biểu mẫu, macro). Thực hiện điều này sử dụng một tính năng mà Access gọi là Documenter. Để in một sắp xếp báo cáo, chọn tùy chọn Analyze từ thực đơn Tools. Access hiển thị một trình đơn phụ và từ đó bạn chọn Documenter. Access hiển thị hộp thoại Documenter như hình 1.



Hình 1: Hộp thoại Documenter

Nếu bạn có một đối tượng được lựa chọn trước khi bạn chọn Documenter, loại đối tượng bạn đã chọn điều chỉnh tab của hộp thoại Documenter mà Access hiển thị. Tuy nhiên, bạn có thể xác định một đối tượng khác, chỉ cần kích chuột vào tab thích hợp. Access hiển thị đối tượng đã định nghĩa phù hợp với thể loại bạn lựa chọn.

Tiếp theo, tích vào các hộp chọn bên cạnh đối tượng hoặc các đối tượng mà bạn muốn có một báo cáo được in. Ví dụ, nếu bạn muốn có một sắp xếp báo cáo cho bảng Business Customers (hình 1), bạn nên chắc chắn hộp chọn phía bên trái đối tượng Business Customers được tích. Sau đó, kích chuột vào nút **Options**. Access hiển thị hộp thoại Print Table Definition như hình 2.



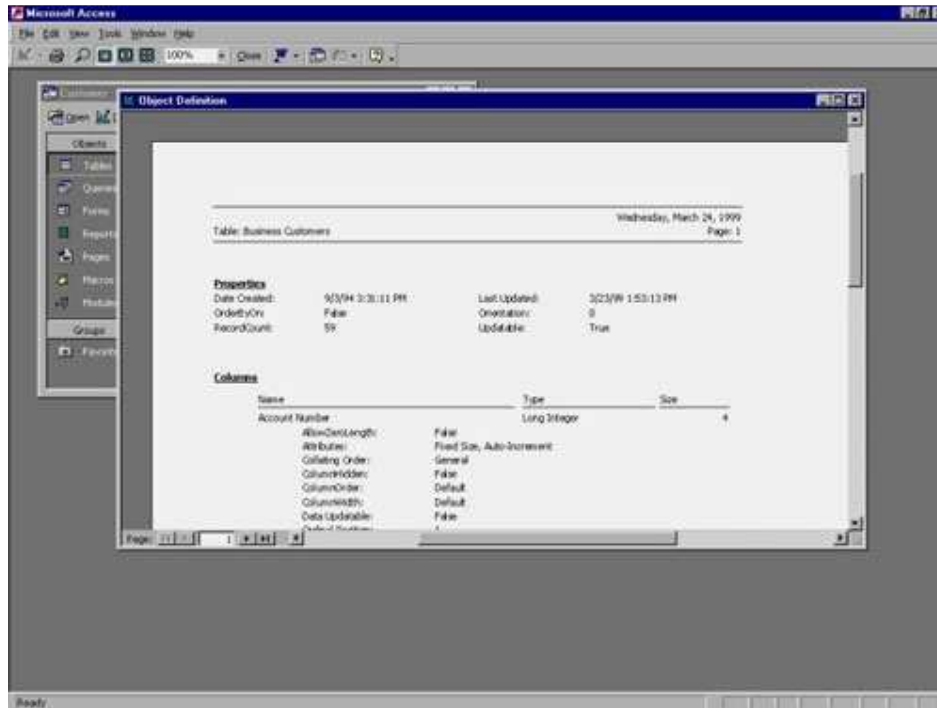
Hình 2: Hộp thoại Print Table Definition.

Hộp thoại Print Table Definition cho phép bạn xác định các phần của bảng định nghĩa mà bạn muốn Access in. Trong phần đầu tiên, xác định bảng thuộc tính bạn muốn Access bao gồm cả dữ liệu in ra. Hai phần tiếp theo cho phép bạn chỉ ra mức độ chi tiết cho các trường và các chỉ mục. (Indexes là một hình thức dưới dạng các phím cho các bảng. Chúng không thường được sử dụng, ngoại trừ các cơ sở dữ liệu chuyên dụng. Bạn tạo các chỉ mục khi thiết kế bảng.) Hãy thực hiện những điều chỉnh nếu cần thiết và sau đó kích chuột vào nút **OK**.

*Chú ý: Nếu bạn chọn in sự sắp xếp của các đối tượng cơ sở dữ liệu khác*

(ngoài bảng), Access thay thế hộp thoại Print Table Definition với một hộp thoại thích hợp cho loại đối tượng được chọn.

Kích chuột vào nút **OK** ở hộp thoại Documenter. Access kiểm tra bảng (hoặc các đối tượng cơ sở dữ liệu bạn đã chọn) và đưa ra một sự xác định in. Sau một thời gian ngắn, Access hiển thị dữ liệu in ra trên màn hình như hình 3.



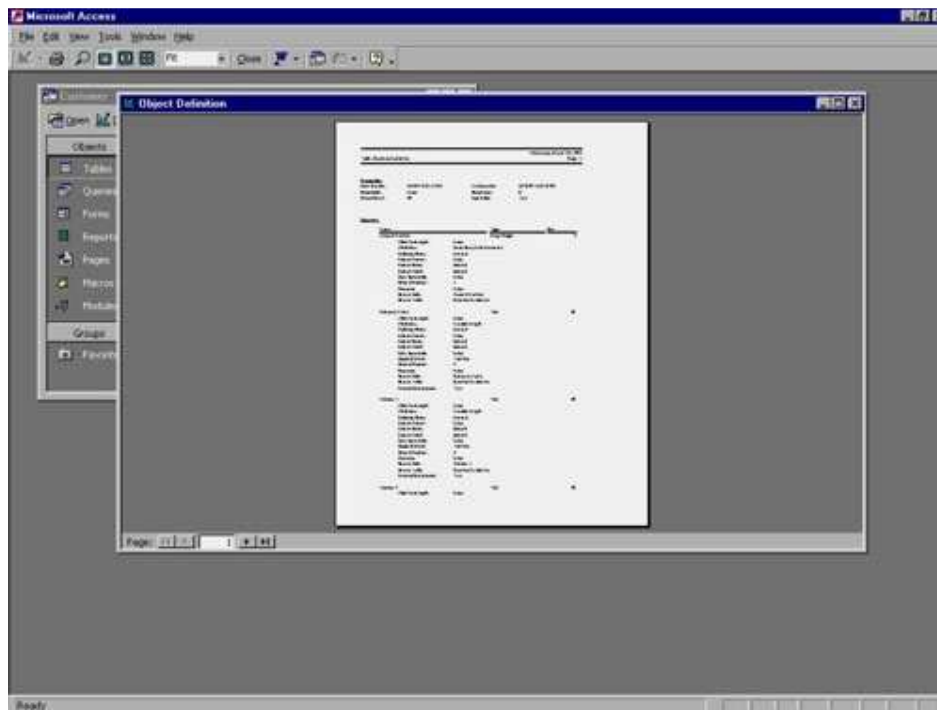
Hình 3: Print Definition cho một bảng.

Trước khi in sự định nghĩa từ máy in, Access cho bạn một cơ hội để xem nó trên màn hình. Quá trình xem này là print preview (xem trước khi in); nó cho phép bạn chắc chắn lại mọi thứ đều ổn trước khi in. Chú ý rằng Access làm giảm bớt số lượng các công cụ trên thanh công cụ ở phía trên cùng của màn hình. Chỉ có một vài công cụ mà bạn thường sử dụng sau đây hiển thị:

- *Print*: Công cụ này hiển thị hộp thoại Print, với mục tiêu của các báo cáo gửi tới máy in.
- *Zoom*: Có hai công cụ zoom. Một công cụ trông giống kính lúp; nó luân phiên giữa việc xem cận cảnh và toàn bộ trang dữ liệu in. Một công cụ khác là công cụ tỷ lệ phần trăm; nó cho phép bạn chỉ ra một tỷ lệ phần trăm cụ thể cho phóng đại.
- *Pages Per Screen*. Loại này có 3 công cụ, được gọi là One Page, Two Pages, và Multiple Pages. 3 công cụ này cho phép bạn xác định có bao nhiêu trang in được hiển thị trên màn hình cùng một lúc.
- *Close Window*. Công cụ này hủy bỏ việc in báo cáo và trở lại cửa sổ Database.

Thông tin thêm về công cụ Zoom, bạn có thể sử dụng chuột để phóng đại hoặc thu nhỏ. Chú ý rằng khi bạn di chuyển nó trên trang in, con trỏ chuột xuất hiện như một kính lúp. Nếu bạn kích chuột, Access hiển thị luân phiên việc xem cận cảnh hoặc toàn bộ trang. Khi bạn thấy báo cáo (hình 3), nó đang ở chế độ cận cảnh. Nếu bạn muốn xem toàn bộ trang, chỉ cần kích chuột lên trang in. Màn hình xuất hiện như hình 4.





Hình 4: Xem toàn bộ trang

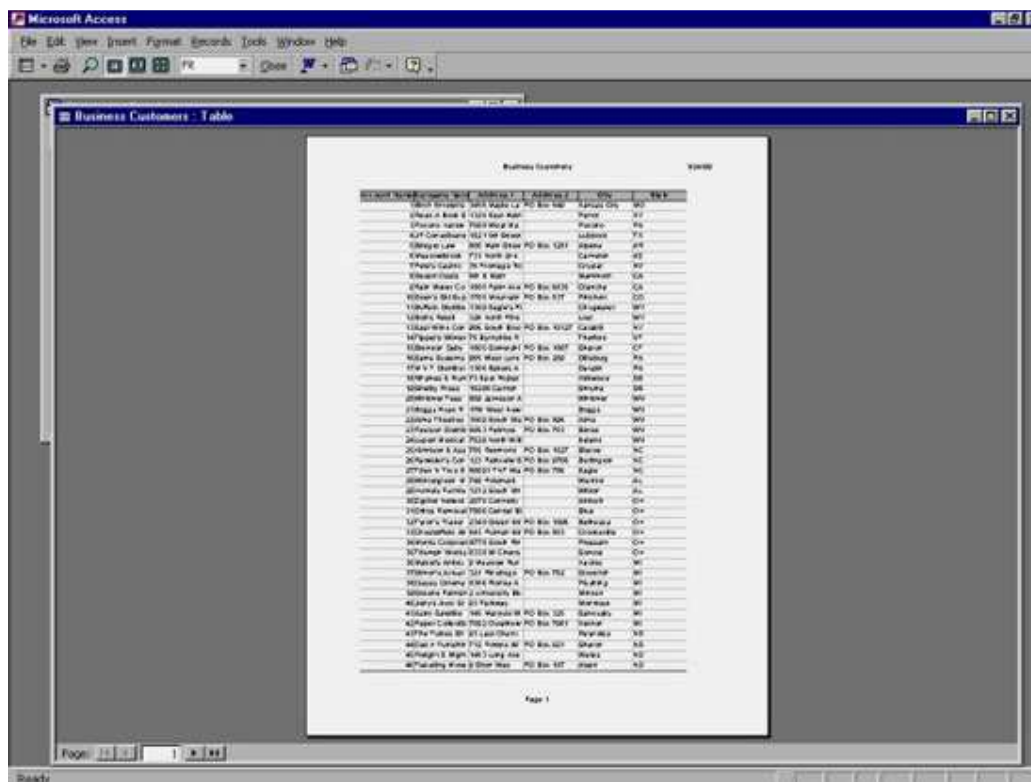
Nếu sử dụng con trỏ chuột để kích chuột lên trang in một lần nữa, bạn sẽ thấy trang được xem cận cảnh tại nơi con trỏ mà bạn kích

Khi bạn đã sẵn sàng sự định nghĩa in, kích chuột vào công cụ **Print**. Lần lượt, Access in báo cáo từ máy in mặc định. Thay vào đó, nếu bạn muốn đóng dữ liệu in, chỉ cần kích chuột vào công cụ **Close**.

## Bài 28: Sử dụng công cụ Print Preview

Ở phần trước, bạn đã làm quen một chút về tính năng của công cụ xem trước khi in trong Access. Công cụ này cho phép bạn xem lại dữ liệu được in trước in ra giấy. Do đó, nó giúp cho bạn nhiều hữu ích hơn.

Bạn có thể chọn tùy chọn Print Preview từ thực đơn File bất kỳ lúc nào muốn. Ví dụ, nếu bạn chọn Print Preview trong khi bạn đang mở một Datasheet, bạn sẽ thấy bảng tương tự như hình 1 khi nó được in.



Hình 1: Chế độ Print Preview của một bảng

Cũng như các công cụ trong phần trước, công cụ Print Preview cho phép bạn xem xét dữ liệu sẽ in ra trước khi nó được đưa tới máy in.

## Bài 29: Tóm tắt chương 7

Access dựa trên môi trường Windows khi đề cập tới in ấn. Chương này đã giới thiệu cho bạn những điều cần bản về in ấn trong Access. Những kiến thức bạn thu được trong chương này sẽ giúp bạn rất nhiều khi làm việc với các chương sau về in ấn các báo cáo.

Trước khi chuyển sang chương 8, phải chắc chắn bạn hiểu được những nội dung sau:

- Access cho phép bạn sử dụng bất kỳ loại máy in nào có thể dùng với Windows. Nhiều lựa chọn in ấn hiện hữu sẽ phụ thuộc vào loại máy in.
- Để thay đổi máy in bạn đang sử dụng, chọn **Page Setup** từ thực đơn **File** hoặc kích chuột vào công cụ **Page setup** hoặc nút **Setup**. Những tùy chọn trong hộp thoại Page Setup cho phép bạn xác định cách bạn muốn thông tin được in ra.
- Bạn có thể sử dụng hộp thoại Page Setup để xác định loại giấy, nơi nó được đặt, cách định hướng giấy và lề trang.
- Access cho phép bạn in nhiều nội dung của bảng một cách dễ dàng. Để thực hiện, chọn hoặc hiển thị bảng. Sau đó, chọn **Print** từ thực đơn **File** hoặc kích chuột vào công cụ **Print**.
- Bạn có thể in một báo cáo dễ dàng mà miêu tả như một đối tượng cơ sở dữ liệu, ví dụ như bảng. Để thực hiện, chọn **Analyze** từ thực đơn **Tools** và **Documenter** từ kết quả trình đơn phụ.

- Print Preview là một tính năng cho phép bạn xem dữ liệu in ra trên màn hình trước khi bạn gửi dữ liệu tới máy in.

Trong chương 8, bạn sẽ học cách có thể tạo một cấu trúc báo cáo đơn giản với Access.

## Thông tin chi tiết về mối quan hệ dữ liệu

### Diagramming a Relationship Biểu đồ là một mối quan hệ

#### Introduction Giới thiệu

In the previous lesson, we saw how to establish a relationship between two tables. Trong bài trước, chúng tôi thấy làm thế nào để thiết lập một mối quan hệ giữa hai bảng. We also saw that the fields that relate both tables must be of the same data type. Chúng tôi cũng thấy rằng các lĩnh vực có liên quan cả hai bảng phải có cùng một kiểu dữ liệu. To make it possible to visually perform data entry on a table, we learned to create a lookup field. Để làm cho nó có thể thực hiện nhập dữ liệu trực quan trên bàn, chúng tôi đã học để tạo ra một lĩnh vực tra cứu. Indeed, this is based on Microsoft Access' high level of support for visual database development. Thật vậy, điều này là dựa trên 'cao cấp của Microsoft Access hỗ trợ cho phát triển cơ sở dữ liệu hình ảnh.

In reality, the only important requirement is that the primary key and the foreign key be of the same data type. Trong thực tế, chỉ yêu cầu quan trọng là khóa chính và ngoại khóa được của cùng một kiểu dữ liệu. In most database scenarios and implementation, the primary key and the foreign key should integer based. Trong kịch bản cơ sở dữ liệu nhất và thực hiện, các khóa chính và ngoại nước quan trọng nên số nguyên dựa. As discussed in the previous lesson, the relationship is typically managed through the long integer data. Như được thảo luận trong bài học trước đó, mối quan hệ thường được quản lý thông qua các dữ liệu số nguyên dài.

With experience, you do not need to create a lookup field on a table. Với kinh nghiệm, bạn không cần phải tạo ra một lĩnh vực tra cứu trên bàn. Besides, since you will usually not let your users use your tables, your primary concern for user interact is to create the combo boxes on forms. Bên cạnh đó, vì bạn sẽ thường không cho phép người dùng của bạn sử dụng bảng của bạn, mối quan tâm chính của bạn cho người sử dụng tương tác là tạo ra các hộp kết hợp về hình thức. To start, you can create the tables and the primary keys as you judge them necessary. Để bắt đầu, bạn có thể tạo các bảng và các từ khóa chính như bạn đánh giá họ cần thiết.

#### ❖ Practical Learning: Creating a Numeric Lookup Field Học tập thực tế: Tạo ra một Field Lookup Numeric

1. Start Microsoft Access and create a Blank Database Bắt đầu Microsoft Access và tạo một cơ sở dữ liệu Blank
2. Set the name of the database as **Bethesda Car Rental2** and click Create Đặt tên của cơ sở dữ liệu như **xe Rental2 Bethesda** và nhấp vào Tạo
3. On the default table, click Add New field and type **Category** Trên bảng mặc định, bấm vào Add New lĩnh vực và loại **Mục**
4. Save the table as **Categories** and switch it to Design View Lưu bảng như là **hạng mục** và chuyển đổi nó để thiết kế Xem
5. Complete the table as follows: Hoàn thành bảng như sau:

Field Name Tên trường	Data Type Loại dữ liệu	Caption Chú thích	Field Size Field Size	Format Định dạng
--------------------------	---------------------------	----------------------	--------------------------	------------------------

<b>CategoryID</b> <b>CategoryID</b> (Primary Key) (Primary Key)	AutoNumber AutoNumber	<b>Category ID Thể loại ID</b>		
<b>Category Thể loại</b>	Text Văn bản		50 50	
<b>Daily Hàng ngày</b>	Number Số		Double Đôi	Fixed Cố định
<b>Weekly Hàng tuần</b>	Number Số		Double Đôi	Fixed Cố định
<b>Monthly Hàng tháng</b>	Number Số		Double Đôi	Fixed Cố định
<b>Weekend Weekend</b>	Number Số		Double Đôi	Fixed Cố định

- Save the table and close it Lưu bảng và đóng nó lại
- On the Ribbon, click Create and, in the Forms section, click Form Design Trên Ribbon, nhấp vào Tạo và, trong phần hình thức, kích Thiết kế Form
- Save the form as Rental Rates and change the following characteristics in the Properties window: Lưu dạng như Giá cho thuê và thay đổi các đặc điểm sau trong cửa sổ Properties:

**Record Source: Cars Record Source: Ô tô**

**Caption: Rental Rates Chú thích: Giá cho thuê**

**Default View: Continuous Forms Mặc định: Các hình thức liên tục**

**Navigation Buttons: No Danh mục chính Buttons: No**

- Design the form as follows (do not try to match the fonts; use whatever font you have on your computer): Thiết kế hình thức như sau (không cố gắng để phù hợp với phông chữ, sử dụng bất cứ font bạn có trên máy tính của bạn):

- Save the form and switch it to Form View Lưu mẫu và chuyển đổi nó vào Form View
- Create the following records: Tạo hồ sơ sau đây:

<b>Category</b>	<b>Daily</b>	<b>Weekly</b>	<b>Monthly</b>	<b>Weekend</b>
-----------------	--------------	---------------	----------------	----------------

<b>Thể loại</b>	<b>Hàng ngày</b>	<b>Hàng tuần</b>	<b>Hàng tháng</b>	<b>Weekend</b>
Economy Kinh tế	34.95 34,95	28.75 28,75	24.95 24,95	24.95 24,95
Compact Compact	38.95 38,95	32.75 32,75	28.95 28,95	28.95 28,95
Standard Standard	45.95 45,95	39.75 39,75	35.95 35,95	34.95 34,95
Full Size Kích thước đầy đủ	50.00 50,00	45.00 45,00	42.55 42,55	38.95 38,95
Mini Van Mini Vãn	55.00 55,00	50.00 50,00	44.95 44,95	42.95 42,95
SUV SUV	56.95 56,95	52.95 52,95	44.95 44,95	42.95 42,95
Truck Xe tải	62.95 62,95	52.75 52,75	46.95 46,95	44.95 44,95
Van Vãn	69.95 69,95	64.75 64,75	52.75 52,75	49.95 49,95

12. Close the form Đóng mẫu

13. To create a new table, on the Ribbon, click Create and, in the Tables section, click the Table Design Để tạo một bảng mới, trên Ribbon, nhấn Create và, trong phần Tables, nhấp vào Thiết kế Bảng

14. Create the table with the following fields: Tạo bảng với các lĩnh vực sau đây:

<b>Field Name</b> <b>Tên trường</b>	<b>Data Type</b> <b>Loại dữ liệu</b>	<b>Caption</b> <b>Chú thích</b>	<b>Field Size</b> <b>Field Size</b>
<b>CarID</b> CarID (Primary Key) (Primary Key)	AutoNumber AutoNumber	<b>Car ID</b> ID <b>xe</b>	
<b>TagNumber</b> <b>TagNumber</b>	Text Văn bản	<b>Tag</b> <b>Number Số</b> <b>Tag</b>	20 20
<b>Make</b> Hã	Text Văn bản		50 50
<b>Model</b> Model	Text Văn bản		50 50
<b>CarYear</b> <b>CarYear</b>	Number Số	<b>Year</b> Năm	Integer Integer
<b>CategoryID</b> <b>CategoryID</b>	Number Số	<b>Category</b> <b>Thể loại</b>	Long Integer Long Integer
<b>Doors</b> Doors	Number Số		Byte Byte
<b>Picture</b> Picture	OLE Object OLE Object		

<b>Condition Điều kiện</b>	Text Văn bản		50 50
<b>Available Available</b>	Yes/No Có / Không		
<b>Notes Ghi chú</b>	Memo Memo		

15. Click Condition and, in the bottom section of the table, click the Lookup tab. Click Điều kiện và, trong phần dưới của bảng, hãy nhấp vào tab Lookup. Change the properties as follows: Thay đổi các thuộc tính như sau:

**Display Control: Combo Box** **Hiển thị Control: Combo Box**

**Row Source Type: Value List** **Row Source Type: Giá trị Danh sách**

**Row Source: "Excellent";"Good Shape";"Needs Repair";"Must be Retired"**

**Row Nguồn: "Tuyệt vời", "Good Shape", "Nhu cầu sửa chữa", "Phải về hưu"**

**Bound Column: 1** **Bound Column: 1**

**Column Count: 1** **Cột Count: 1**

**Column Widths: 1"** **Cột Widths: 1 "**

**List Width: 1"** **Danh sách Chiều rộng: 1 "**

16. Save the table as **Cars** and close the table Lưu bảng như **Ô tô** và đóng bảng  
 17. Create a new table in Design View as follows: Tạo một bảng mới trong thiết kế xem như sau:

<b>Field Name Tên trường</b>	<b>Data Type Loại dữ liệu</b>	<b>Caption Chú thích</b>	<b>Field Size Field Size</b>
<b>EmployeeID</b> <b>EmployeeID</b> (Primary Key) (Primary Key)	AutoNumber AutoNumber	<b>Employee ID</b> <b>Employee ID</b>	
<b>EmployeeNumber</b> <b>EmployeeNumber</b>	Text Văn bản	<b>Employee #</b> <b>Nhân viên #</b>	20 20
<b>FirstName</b> <b>FirstName</b>	Text Văn bản	<b>First Name Tên riêng</b>	50 50
<b>LastName</b> <b>LastName</b>	Text Văn bản	<b>Last Name Tên họ</b>	50 50
<b>Title</b> <b>Tiêu đề</b>	Text Văn bản		100 100
<b>Notes</b> <b>Ghi chú</b>	Memo Memo		

18. Save the table as **Employees** and close it Lưu bảng như là **nhân viên** và đóng nó lại  
 19. Create a form for the Employees table, save it as Employees, and design it as you see fit. Tạo một hình thức cho bảng Nhân viên, lưu nó như là nhân viên, và thiết kế nó như bạn thấy phù hợp. Here is an example: Dưới đây là một ví dụ:

Employees

Form Header

# EMPLOYEES

Detail

Employee ID	EmployeeID
Employee #	EmployeeN
First Name	FirstName
Last Name	LastName
Title	Title
Notes	Notes

Form Footer

Employees

# EMPLOYEES

Employee ID (New)

Employee #

First Name

Last Name

Title

Notes

Record: 1 of 1 Search

20. Save the form and [create the employees](#) Lưu hình thức và [tạo các nhân viên](#)
21. Close the form Đóng mẫu
22. Create a new table in Design View with the following fields: Tạo một bảng mới trong thiết kế Xem với các lĩnh vực sau đây:

Field Name Tên	Data Type Loại	Caption Chú thích	Field Size
----------------	----------------	-------------------	------------

trường	dữ liệu		Field Size
<b>CustomerID</b> <b>CustomerID</b> (Primary Key) (Primary Key)	AutoNumber AutoNumber	<b>Cutomer ID</b> <b>Cutomer ID</b>	
<b>DrvLicNumber</b> <b>DrvLicNumber</b>	Text Văn bản	<b>Driver's License #</b> <b>Bằng lái xe #</b>	50 50
<b>FullName</b> <b>Họ và tên</b>	Text Văn bản	<b>Full Name</b> <b>Họ và tên</b>	100 100
<b>Address</b> <b>Địa chỉ</b>	Text Văn bản		100 100
<b>City</b> <b>Thành phố</b>	Text Văn bản		50 50
<b>State</b> <b>Nhà nước</b>	Text Văn bản		100 100
<b>ZIPCode</b> <b>Mã bưu điện</b>	Text Văn bản	<b>ZIP Code</b> <b>Zip</b>	20 20
<b>Notes</b> <b>Ghi chú</b>	Memo Memo		

23. Save the table as **Customers** and close it Lưu bảng như **Khách hàng** và đóng nó lại
24. Create a form for the Employees table, save it as Employees, and design it as you see fit. Tạo một hình thức cho bảng Nhân viên, lưu nó như là nhân viên, và thiết kế nó như bạn thấy phù hợp. Here is an example: Dưới đây là một ví dụ:



25. Save the form Lưu mẫu
26. [Create some customers](#) before closing the form [Tạo một số khách hàng](#) trước khi đóng cửa dưới hình thức
27. Create a new table in Design View with the following fields: Tạo một bảng mới trong thiết kế Xem với các lĩnh vực sau đây:

Field Name Tên trường	Data Type Loại dữ liệu	Caption Chú thích	Field Size Field Size	Other Properties Các tiết
<b>RentalOrderID</b> <b>RentalOrderID</b> (Primary Key) (Primary Key)	AutoNumber AutoNumber	<b>Rental Order ID Đặt hàng cho thuê ID</b>		
<b>EmployeeID</b> <b>EmployeeID</b>	Number Số	<b>Processed By Xử lý theo</b>	Long Integer Long Integer	
<b>CustomerID</b> <b>CustomerID</b>	Number Số	<b>Processed For Đối với chế biến</b>	Long Integer Long Integer	
<b>CarID CarID</b>	Number Số	<b>Car Xe hơi</b>	Long Integer Long Integer	
<b>CarCondition</b> <b>CarCondition</b>	Text Văn bản	<b>Car Condition Điều kiện xe</b>	50 50	

		<b>hơi</b>		
<b>TankLevel</b> <b>TankLevel</b>	Text Văn bản	<b>Tank Level</b> <b>Cấp Tank</b>	50 50	
<b>MileageStart</b> <b>MileageStart</b>	Number Số	<b>Mileage Start</b> <b>Mileage bắt đầu</b>	Integer Integer	
<b>MileageEnd</b> <b>MileageEnd</b>	Number Số	<b>Mileage End</b> <b>Mileage End</b>	Integer Integer	
<b>TotalMileage</b> <b>TotalMileage</b>	Number Số	<b>Total Mileage</b> <b>Tổng số Mileage</b>	Integer Integer	
<b>StartDate</b> <b>StartDate</b>	Date/Time Ngày / Thời gian	<b>Start Date</b> <b>Ngày bắt đầu</b>		
<b>EndDate</b> <b>EndDate</b>	Date/Time Ngày / Thời gian	<b>End Date</b> <b>Ngày kết thúc</b>		
<b>TotalDays</b> <b>TotalDays</b>	Number Số	<b>Total Days</b> <b>Tổng số ngày</b>	Integer Integer	
<b>RateApplied</b> <b>RateApplied</b>	Number Số	<b>Rate Applied</b> <b>Đánh giá Ứng dụng</b>	Double Đôi	Format: Fixed Định dạng: cố định
<b>TaxRate</b> <b>TaxRate</b>	Number Số	<b>Tax Rate</b> <b>Thuế suất</b>	Double Đôi	Format: Percent Định dạng: Phần trăm Default Value: 7.50 Giá trị mặc định: 7,50
<b>OrderStatus</b> <b>OrderStatus</b>	Text Văn bản	<b>Order Status</b> <b>Tình trạng đặt hàng</b>	50 50	Default Value: "Unknown" Giá trị mặc định: "Không biết"
<b>Notes Ghi chú</b>	Memo Memo			

28. Click TankLevel Click TankLevel and, in the bottom section of the table, click the Lookup tab. và, trong phần dưới của bảng, nhấp vào tab Lookup. Change the properties as follows: Thay đổi các thuộc tính như sau:

**Display Control: Combo Box** **Hiển thị Control: Combo Box**  
**Row Source Type: Value List** **Row Source Type: Giá trị Danh sách**  
**Row Source: "Empty";"1/4 Empty";"Half";"3/4 Full";"Full"** **Row Nguồn: "rỗng", "1 / 4 rỗng", "Một nửa", "3 / 4 toàn", "hoàn toàn"**  
**Bound Column: 1** **Bound Column: 1**  
**Column Count: 1** **Cột Count: 1**  
**Column Widths: 1"** **Cột Widths: 1 "**  
**List Width: 1"** **Danh sách Chiều rộng: 1 "**

29. Click OrderStatus Click OrderStatus and, in the bottom section of the table, click the Lookup tab. và, trong phần dưới của bảng, nhấp vào tab Lookup. Change the properties as follows: Thay đổi các thuộc tính như sau:

**Display Control: Combo Box** **Hiện thị Control: Combo Box**  
**Row Source Type: Value List** **Row Source Type: Giá trị Danh sách**  
**Row Source: "Unknown";"Car On Road";"Order Completed";"Order Reserved"** **Row Nguồn: "Không biết", "xe Trên đường", "đặt hàng đã hoàn thành", "Đặt hàng Reserved"**  
**Bound Column: 1** **Bound Column: 1**  
**Column Count: 1** **Cột Count: 1**  
**Column Widths: 1"** **Cột Widths: 1 "**  
**List Width: 1"** **Danh sách Chiều rộng: 1 "**

30. Save the table as **RentalOrders** and close it Lưu bảng như **RentalOrders** và đóng nó lại

## The Relationships Diagram Các mối quan hệ Sơ đồ

---

Tables and fields relationships can be created and managed in a special window called the Relationships window. Bàn và các lĩnh vực mối quan hệ có thể được tạo ra và quản lý trong một cửa sổ đặc biệt gọi là cửa sổ quan hệ. To display it: Để hiển thị nó:

- If no table is currently opened, on the Ribbon, you can click Database Tools. Nếu không có bảng hiện tại đang mở ra, trên Ribbon, bạn có thể bấm vào cơ sở dữ liệu Công cụ. In the Show/Hide section, click the Relationships button Trong phần Show



Ấn /, nhấn vào nút các mối quan hệ

- If a table is currently displaying, on the Ribbon, you can click the Datasheet tab and, in the Relationships section, click the Relationships button Nếu một bảng hiện đang hiển thị, trên Ribbon, bạn có thể nhấp vào tab Datasheet và, trong phần Quan hệ, nhấn vào nút các mối quan hệ

After clicking one of those: Sau khi nhấp vào một trong những người:

- If no relationship exists among the tables in the current database, the Show Table dialog box would come up, asking you to select the tables whose relationship(s) you want to create. Nếu không có tồn tại mối quan hệ giữa các bảng trong cơ sở dữ liệu hiện tại, các bảng hiển thị hộp thoại sẽ xuất hiện, yêu cầu bạn chọn các bảng có mối quan hệ (s) bạn muốn tạo. To add a table, select it, click Add and click Close Để thêm một bảng, chọn nó, click Add và nhấn Close
- If at least one relationship has been created between two tables, the Relationships window would come up and display that relationship or the already existing relationships Nếu ít nhất một mối quan hệ tại đã được tạo ra giữa hai bảng, cửa sổ các mối quan hệ sẽ đi lên và hiển thị các mối quan hệ đó hoặc các mối quan hệ hiện tại đã được

When you are working on the relationships of your tables, a window with a tab labeled Relationships displays in Microsoft Access. Khi bạn đang làm việc trên các mối quan hệ của các bảng của bạn, một cửa sổ với các tab một mối quan hệ gắn nhãn hiển thị trong Microsoft Access. Also, the Ribbon is equipped with a tab labeled design and that includes two sections: Ngoài ra, các Ribbon được trang bị một thẻ có nhãn thiết kế và bao gồm hai phần:



The Design tab of the Ribbon provides various tools to assist you with creating and managing the relationships. Các tab Thiết kế của Ribbon cung cấp các công cụ khác nhau để giúp quý vị tạo và quản lý các mối quan hệ. For example, if the Show Table dialog box has been closed and if you want to show it: Ví dụ, nếu hiển thị hộp thoại Bảng đã được đóng lại và nếu bạn muốn hiển thị nó:

- You can click the Show Table button on the Ribbon Bạn có thể bấm vào nút Show Table trên Ribbon
- You can right-click the body of the Relationships window and click Show Table... Bạn có thể nhấp chuột phải vào cơ thể của cửa sổ Mối quan hệ và nhấp vào Hiển thị Bảng ...

You can create a relationship only between two tables that are present on the Relationships window. Bạn có thể tạo một mối quan hệ duy nhất giữa hai bảng được trình bày trên cửa sổ Các mối quan hệ. This means that even if a table is part of your database and you want to link it to another table (of your database), if the table has not been added to the Relationships window, you cannot create or manage its relationship to another table. Điều này có nghĩa rằng ngay cả khi bảng là một phần của cơ sở dữ liệu của bạn và bạn muốn liên kết nó với một bảng (trong cơ sở dữ liệu của bạn), nếu bảng đã không được thêm vào cửa sổ Quan hệ, bạn không thể tạo hoặc quản lý mối quan hệ của mình để bàn khác. Of course, there are other ways you can create relationships without using the Relationships window but the Relationships window gives you detailed means of creating and managing relationships. Tất nhiên, có những cách khác bạn có thể tạo mối quan hệ mà không sử dụng cửa sổ các mối quan hệ nhưng cửa sổ mối quan hệ mang lại cho bạn có nghĩa là chi tiết về tạo và quản lý mối quan hệ.

To establish a relationship that does not yet exist between two tables, you can drag the primary key from the parent table to the foreign key of the desired table. Để thiết lập một mối quan hệ đó chưa tồn tại giữa hai bảng, bạn có thể kéo khóa chính từ bảng phụ huynh đến khoá ngoại của bảng mong muốn. If you drag and drop accurately, the relationship would be acknowledged and you can just click Create to make it formal. Nếu bạn kéo và thả một cách chính xác, mối quan hệ này sẽ được công nhận và bạn có thể chỉ cần nhấp vào Tạo để làm cho nó chính thức. If you dropped the primary key on the wrong field, you would have time to select the appropriate fields in the Edit Relationship dialog box. Nếu bạn bỏ khoá chính trên các lĩnh vực sai, bạn sẽ có thời gian để chọn các trường thích hợp trong hộp thoại Edit Mối quan hệ.

After working with the Relationships window, you can close it by clicking its Close button Sau khi làm việc với các cửa sổ Quan hệ, bạn có thể đóng nó bằng cách nhấp vào nút Close của

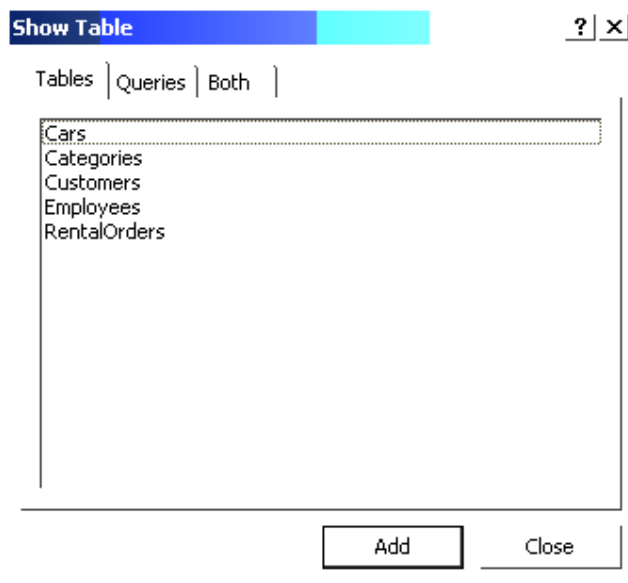


nó . You would be asked to save it in order to keep the relationship(s) created. Bạn sẽ được yêu cầu để lưu nó để giữ mối quan hệ (s) tạo ra.

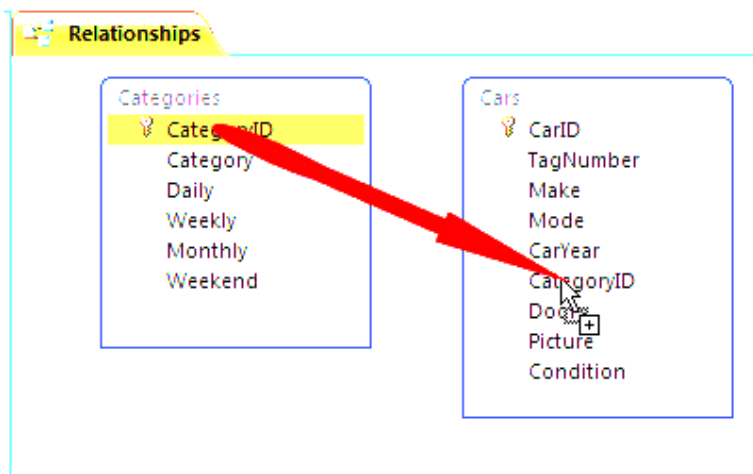
1. The Bethesda Car Renatal3 database should still be opened. Các cơ sở dữ liệu Renatal3 Bethesda xe vẫn còn phải được mở ra.  
On the Ribbon, click Database Tools Trên Ribbon, nhấn Tools Cơ sở dữ liệu
2. In the Show/Hide section, click the Relationships button Trong phần Show Ẩn /, nhấn



vào nút các mối quan hệ. The Show Table property sheet comes up: đi kèm. The Show Bảng sở hữu từ ký:



3. Click Categories once and click the Add button Click vào mục một lần và nhấp vào nút Add
4. Double-click Cars Click đôi vào Ô tô
5. On the Show Table dialog box, click the Close button Trên hộp thoại Show Table, nhấn vào nút Close
6. To add another table as if we forgot it, right-click an empty area in the Relationships window and click Show Table... Để thêm một bảng như nếu chúng ta quên nó, kích chuột phải vào một vùng trống trong cửa sổ Mối quan hệ và nhấp vào Hiển thị Bảng  
...
7. In the Show Table dialog box, click Customers Trong hộp thoại Show Table, click Khách hàng
8. Press and hold Shift Bấm và giữ Shift
9. Click RentalOrders and release Shift Click RentalOrders và phát hành Shift
10. Click Add and click Close Click Add và nhấn Close
11. Drag the CategoryID field from the Categories table and drop it on top of the CategoryID field in the Cars table: Kéo trường CategoryID từ bảng hạng mục và thả nó trên đầu trang của CategoryID trong lĩnh vực Ô tô bảng:

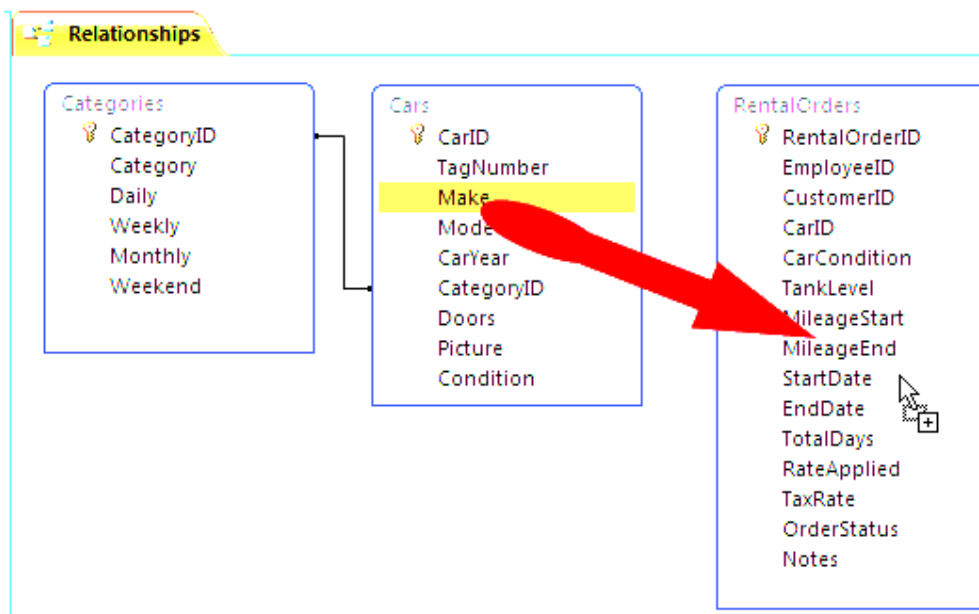


The Edit Relationship dialog box would come up Các hộp thoại Edit Mỗi quan hệ sẽ đi lên

The Edit Relationships dialog box shows the relationship between CategoryID in the Categories table and CategoryID in the Cars table. The relationship type is One-To-Many. There are buttons for Create, Cancel, Join Type.., and Create New..

12. Now you have a line relating these two tables. Bây giờ bạn có một dòng liên quan hai bảng.

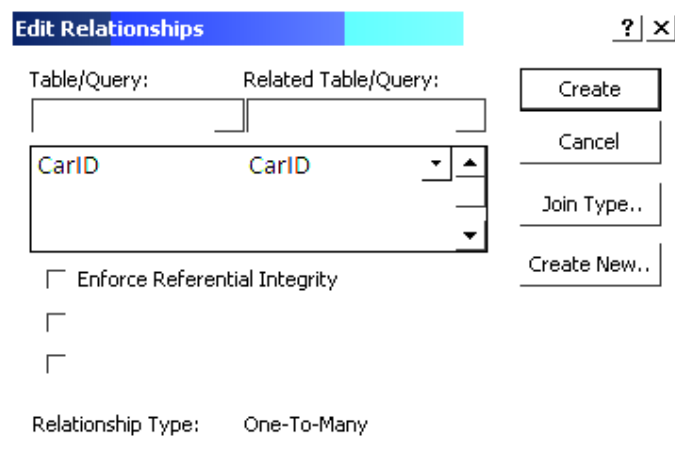
Drag any field from the Cars table and drop it on top of any field in the RentalOrders table if you missed the target Kéo bất kỳ lĩnh vực từ bảng Ô tô và thả nó trên đầu trang của lĩnh vực bất kỳ trong bảng RentalOrders nếu bạn bị mất những mục tiêu



13. Once again, the Edit Relationship dialog box comes up. Một lần nữa, hộp thoại Edit Mỗi quan hệ đi lên.

On the dialog box, under the left Table/Query, click the field that is selected to display its combo box. Trên hộp thoại, dưới bàn trái / truy vấn, hãy nhấp vào trường đó được chọn để hiển thị hộp kết hợp của nó. In that left combo box, select CarID and press Tab Trong hộp kết hợp trái, chọn CarID và nhấn Tab

14. Under Related Table/Query, click the field that is selected and click the arrow of its combo box and select CarID Dưới Related Bảng / nhấp chuột, truy vấn các lĩnh vực đó được chọn và click vào mũi tên của hộp kết hợp của nó và chọn CarID



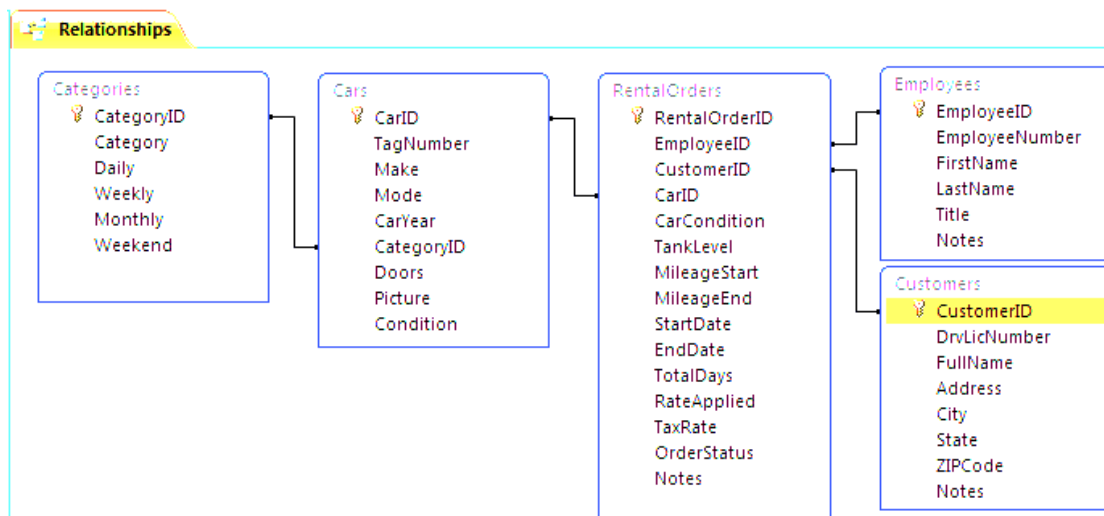
15. Click Create Nhấp vào Tạo

16. In the same way, drag EmployeeID from the Employees table and drop it on EmployeeID in the RentalOrders table Trong cùng một cách, kéo EmployeeID từ bảng nhân viên và thả nó vào EmployeeID trong bảng RentalOrders

17. Click Create Nhấp vào Tạo

18. Drag CustomerID from the Customers table and drop it on CustomerID in the RentalOrders table Kéo CustomerID từ bảng Khách hàng và thả nó vào CustomerID trong bảng RentalOrders

19. Click Create Nhấp vào Tạo



20. On the Relationships section of the Ribbon, click the Close button Về phần các mối



quan hệ nếu Ribbon, nhấp vào nút Close

21. When asked whether you want to save, click Yes Khi được hỏi liệu bạn muốn lưu, click

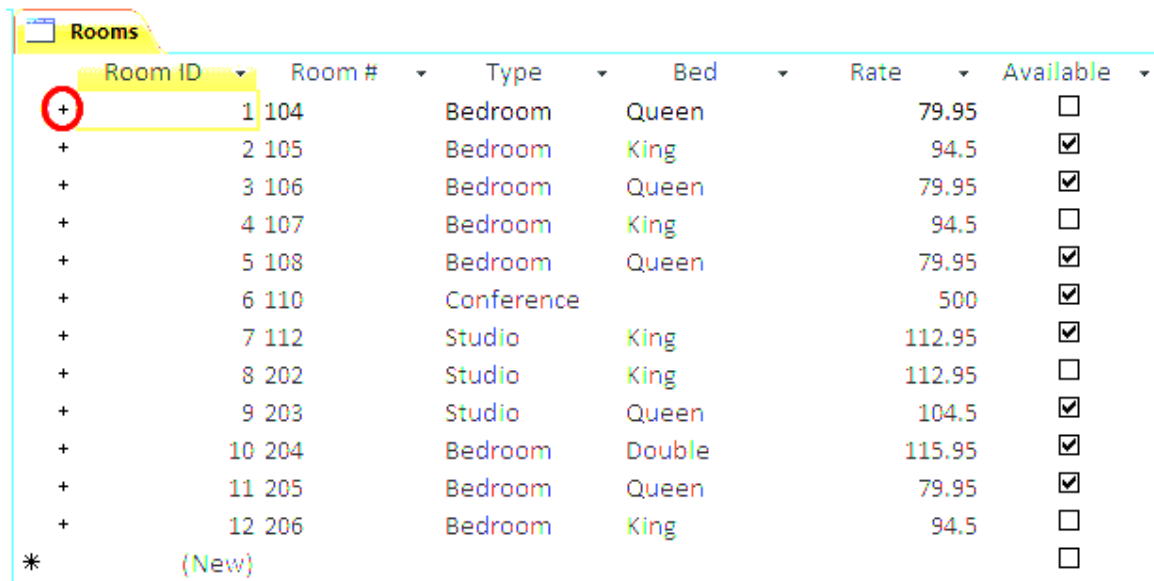
Yes

## A Sub-Datasheet Một tiểu Datasheet

### Introduction Giới thiệu

A sub-datasheet is a means of displaying the dependent records of a parent record on a datasheet. Một phụ Datasheet là một phương tiện hiển thị các bản ghi phụ thuộc của một bản ghi cha mẹ trên một Datasheet. Obviously to make this possible, each record on a table must be configured to have child records. Rõ ràng để làm cho có thể, mỗi bản ghi trên bàn phải được cấu hình để có số liệu trẻ em. In other words, there must be a table containing a foreign key so that the child table can be, or has been, connected to the current table. Nói cách khác, phải có một bảng có chứa một chìa khóa nước ngoài để các bảng con có thể được, hoặc đã được kết nối với bảng hiện hành. For example, imagine that, in a hotel application, various customers have previously rented some rooms and sometimes you want to see the records related to a particular room. Ví dụ, hãy tưởng tượng rằng, trong một ứng dụng khách sạn, khách hàng khác nhau đã từng thuê một số phòng và đôi khi bạn muốn xem các hồ sơ liên quan đến một căn phòng đặc biệt.

Once a relationship has been established between records, when you open the parent table in Datasheet View, each record would appear with a + button to its left: Một khi mối quan hệ đã được thiết lập giữa các hồ sơ, khi bạn mở bảng cha trong Datasheet View, mỗi bản ghi sẽ xuất hiện với một nút + bên trái của nó:



Room ID	Room #	Type	Bed	Rate	Available	
+	1	104	Bedroom	Queen	79.95	<input type="checkbox"/>
+	2	105	Bedroom	King	94.5	<input checked="" type="checkbox"/>
+	3	106	Bedroom	Queen	79.95	<input checked="" type="checkbox"/>
+	4	107	Bedroom	King	94.5	<input type="checkbox"/>
+	5	108	Bedroom	Queen	79.95	<input checked="" type="checkbox"/>
+	6	110	Conference		500	<input checked="" type="checkbox"/>
+	7	112	Studio	King	112.95	<input checked="" type="checkbox"/>
+	8	202	Studio	King	112.95	<input type="checkbox"/>
+	9	203	Studio	Queen	104.5	<input checked="" type="checkbox"/>
+	10	204	Bedroom	Double	115.95	<input checked="" type="checkbox"/>
+	11	205	Bedroom	Queen	79.95	<input checked="" type="checkbox"/>
+	12	206	Bedroom	King	94.5	<input type="checkbox"/>
*	(New)					<input type="checkbox"/>

This means that Microsoft Access is configured to recognize relationships and apply them to show the sub-datasheet. Điều này có nghĩa rằng Microsoft Access được cấu hình để nhận ra mối quan hệ và áp dụng chúng để hiển thị phụ Datasheet.

By its definition, a sub-datasheet allows you to view the related records by clicking the + button. Theo định nghĩa của nó, một phụ Datasheet cho phép bạn xem các hồ sơ liên quan bằng cách nhấn vào nút +. This would expand that record and display its related records. Điều này sẽ mở rộng đó ghi lại và hiển thị hồ sơ liên quan của nó. You can expand just one record by clicking its + button or a few records but clicking the + buttons of the desired records. Bạn có thể mở rộng chỉ là một bản ghi bằng cách nhấn vào nút + của nó hoặc một vài hồ sơ nhưng cách nhấn vào nút + của hồ sơ mong muốn. Here is an example: Dưới đây là một ví dụ:



Room ID	Room #	Type	Bed	Rate	Available	
+ 1	104	Bedroom	Queen	79.95	<input type="checkbox"/>	
- 2	105	Bedroom	King	94.5	<input checked="" type="checkbox"/>	
Occupancy #   Occupancy #   Date Occupied   Processed By   Customer   Rate Applied						
	1	1001	4/5/2008	27049	294209	85.75
	2	1001	4/6/2008	28405	294209	85.75
	4	1001	4/7/2008	70429	294209	85.75
	13	1006	5/2/2008	28405	180204	94.50
	14	1006	5/3/2008	24095	180204	94.50
* (New)						
+ 3	106	Bedroom	Queen	79.95	<input checked="" type="checkbox"/>	
+ 4	107	Bedroom	King	94.5	<input type="checkbox"/>	
- 5	108	Bedroom	Queen	79.95	<input checked="" type="checkbox"/>	
Occupancy #   Occupancy #   Date Occupied   Processed By   Customer   Rate Applied						
	5	1003	4/7/2008	24095	208405	75.85
	6	1003	4/8/2008	28405	208405	75.85
	7	1003	4/9/2008	28405	208405	75.85
	8	1003	4/10/2008	27049	208405	75.85
	9	1003	4/11/2008	24095	208405	75.85
* (New)						
+ 6	110	Conference		500	<input checked="" type="checkbox"/>	
+ 7	112	Studio	King	112.95	<input checked="" type="checkbox"/>	
+ 8	202	Studio	King	112.95	<input type="checkbox"/>	

You can also expand all records. Bạn cũng có thể mở rộng tất cả hồ sơ. To do this, open the table in Datasheet View. Để làm điều này, mở bảng trong Datasheet View. In the Records section of the Home tab of the Ribbon, click More -> Subdatasheet -> Expand All. Trong phần Records của tab Home của Ribbon, nhấn More -> Subdatasheet -> Expand All.

When a record is expanded, it displays a - button to its left. Khi ghi lại một được mở rộng, nó sẽ hiển thị một - nút bên trái của nó. After viewing a record, to collapse it back, you can click its - button. You can do the same for any other record. Sau khi xem một hồ sơ, để thu hẹp nó lại, bạn có thể click - nút của nó. Bạn có thể làm tương tự cho bất cứ kỳ lục khác. You can also collapse all records. Bạn cũng có thể sụp đổ tất cả hồ sơ. To do this, open the table in Datasheet View. Để làm điều này, mở bảng trong Datasheet View. In the Records section of the Home tab of the Ribbon, click More -> Subdatasheet -> Collapse All. Trong phần Records của tab Home của Ribbon, nhấn More -> Subdatasheet -> Collapse All.

### ❖ Practical Learning: Introducing Subdatasheets Học tập thực hành: Giới thiệu Subdatasheets

1. Start a new Blank Database Bắt đầu một cơ sở dữ liệu mới Blank
2. Set the name of the database to **Video Collection3** and click Create Đặt tên của cơ sở dữ liệu để **Video Collection3** và nhấp vào Tạo
3. Close the default table without saving it Đóng bảng mặc định mà không lưu nó
4. To create a new table, on the Ribbon, click Create and, in the Tables section, click the Table Design Để tạo một bảng mới, trên Ribbon, nhấn Create và, trong phần Tables, nhấp vào Thiết kế Bảng
5. Set the name of the first field to **ActorID** and set its **Data Type** to **AutoNumber** Đặt tên trường đầu tiên để **ActorID** và thiết lập **dữ liệu** của nó **Type** để

## AutoNumber

6. While the field is still selected, in the Tools section of the Ribbon, click the **Primary Key** button Trong khi lĩnh vực này vẫn được chọn, trong phần Tools của Ribbon, nhấn



vào nút **Key chính**

7. Set its **Caption** to **Actor ID** Set **Chú thích** của mình cho **nam diễn viên ID**
8. In the upper section of the table, under ActorID, set the name of the second field to **Actor** with the **Data Type** as **Text** Trong phần trên của bảng, dưới ActorID, đặt tên của trường thứ hai để **diễn viên xuất sắc** với các **loại dữ liệu** như **văn bản**
9. Set the last field as Notes with a **Memo** as **Data Type** Đặt trường cuối cùng là cần chú ý với một **Memo** là **loại dữ liệu**

Field Name	Data Type	Description
ActorID	AutoNumber	Automatic Incrementing Number Controlled by the Application
Actor	Text	Actor or actress
Notes	Memo	Comments about the actor or actress

10. To switch the table to the other view, on the Ribbon, click the View button Để chuyển



đổi bảng giao diện khác, trên Ribbon, nhấn vào nút Xem

11. When you are asked whether you want to save the table, click Yes Khi bạn được hỏi bạn có muốn lưu bảng, click Yes
12. Type **Actors** and press Enter. **Diễn viên** Loại hình và nhấn Enter.
13. Press Enter and enter the following names of actors: Nhấn Enter và nhập vào tên sau đây của các diễn viên:

<b>Actor ID</b> <b>Nam diễn viên ID</b>	<b>Actor Diễn viên</b>
1 1	Eddie Murphy Eddie Murphy
2 2	Tom Cruise Tom Cruise
3 3	Ving Rhames Ving Rhames
4 4	Mel Gibson Mel Gibson
5 5	Charlie Sheen Charlie Sheen
6 6	Jaime Pressly Jaime Pressly
7 7	Sheryl Lee Ralph Sheryl Lee Ralph
8 8	Daryl Hannah Daryl Hannah

9 9	Joe Don Baker Joe Don Baker
10 10	Victoria Rowell Victoria Rowell
11 11	Grant Shaud Grant Shaud
12 12	Kevin McCarthy Kevin McCarthy
13 13	Charles S. Dutton Charles S. Dutton
14 14	Mia Kirshner Mia Kirshner
15 15	Lane Smith Lane Smith
16 16	Randy Quaid Randy Quaid
17 17	Philip Seymour Hoffman Philip Seymour Hoffman
18 18	Delroy Lindo Delroy Lindo
19 19	Rene Russo Rene Russo
20 20	Gary Sinise Gary Sinise
21 21	Matthew Broderick Matthew Broderick
22 22	Jean Reno Jean Reno
23 23	Michael Douglas Michael Douglas

14. Close the table Đóng bảng

15. Create a new table in Design View with the following fields: Tạo một bảng mới trong thiết kế Xem với các lĩnh vực sau đây:

Field Name Tên trường	Data Type Loại dữ liệu	Description Mô tả
<b>DirectorID</b> <b>DirectorID</b> (Primary Key) (Primary Key)	AutoNumber AutoNumber	Automatic number Số tự động
<b>Director</b> <b>Giám đốc</b>	Text Văn bản	Name of a director such as "Mark Lynn" or directors as a group of such as "The Hughes Brothers" Tên của một giám đốc như "Mark Lynn" hoặc Giám đốc như là một nhóm như "The Brothers Hughes"
<b>Notes Ghi chú</b>	Memo Memo	Observations about the director or group of directors Quan sát về Giám đốc hoặc nhóm các

		giám đốc
--	--	----------

16. Save the table as **Directors** and switch to Datasheet View Lưu bảng là **Giám đốc** và chuyển sang Datasheet View

17. Enter some directors as follows: Nhập một số giám đốc như sau:

<b>DirectorID</b> <b>DirectorID</b>	<b>Director</b> <b>Giám đốc</b>
1 1	Roland Emmerich Roland Emmerich
2 2	Renny Harlin Renny Harlin
3 3	Tony Scott Tony Scott
4 4	Ron Howard Ron Howard
5 5	Sanjay Leela Bhansali Sanjay Leela Bhansali
6 6	Eddie Murphy Eddie Murphy
7 7	JJ Abrams JJ Abrams
8 8	Jonathan Lynn Jonathan Lynn
9 9	Mel Gibson Mel Gibson
10 10	Steven Spielberg Steven Spielberg
11 11	Joel Gallen Joel Gallen
12 12	Oliver Stone Oliver Stone

18. Create a new table in Design View with the following fields: Tạo một bảng mới trong thiết kế Xem với các lĩnh vực sau đây:

Field Name Tên trường	Data Type Loại dữ liệu	Caption Chú thích
<b>CategoryID</b> <b>CategoryID</b> (Primary Key) (Primary Key)	AutoNumber AutoNumber	Category ID Thẻ loại ID
<b>Category</b> <b>Thẻ loại</b>	Text Văn bản	Category Thẻ loại

<b>Notes Ghi chú</b>	Memo Memo	
----------------------	--------------	--

19. Save the table as **Categories** and switch it to Datasheet View Lưu bảng như là **hạng mục** và chuyển đổi nó để Datasheet View
20. Enter a few categories on the table: Nhập một vài loại trên bàn:

<b>Category ID Thể loại ID</b>	<b>Category Thể loại</b>
	Biography Tiểu sử
	Documentary Phim tài liệu
	Cartoon Cartoon
	Fitness – Aerobic Thể dục - Aerobic
	Comedy – General Hài kịch - Tổng hợp
	Comedy - Parody Hài kịch - Parody
	Comedy – Police Hài kịch - Cảnh sát
	Comedy – Politic Hài kịch - Chính trị
	Comedy – War Hài kịch - Chiến tranh
	Drama – General Drama - Tổng hợp
	Drama – Police Drama - Cảnh sát
	Drama – Politic Drama - Chính trị
	Drama – War Drama - Chiến tranh
	Sci-Fi – General Sci-Fi - Tổng hợp
	Sci-Fi – Comedy Sci-Fi - Hài kịch
	Sci-Fi – Police Sci-Fi - Cảnh sát
	Sci-Fi – Politic Sci-Fi - Chính trị
	Sci-Fi – War Sci-Fi

	- Chiến tranh
	Adventure Phiêu lưu
	Hindu Hindu
	Religious Tôn giáo

## 21. Close the table Đóng bảng

### Creating a Sub-Datasheet Tạo ra một tiểu Datasheet

As we have seen so far, to have a relationship between two tables, you must create a primary key in one table and the corresponding foreign key in another table. Như chúng tôi đã thấy cho đến nay, để có một mối quan hệ giữa hai bảng, bạn phải tạo một khóa chính trong một bảng và các nước ngoài phím tương ứng trong bảng khác. You can then establish a relationship between both tables in the Relationships window. Sau đó bạn có thể thiết lập một mối quan hệ giữa hai bảng trong cửa sổ các mối quan hệ. As stated already, Microsoft Access can take it upon itself to show the related records. Như đã nói rồi, Microsoft Access có thể mất nó khi chính nó để hiển thị các hồ sơ liên quan. You do not have to establish a relation first in order to take advantage of the sub-datasheet effect. Bạn không cần phải thiết lập một mối quan hệ đầu tiên để tận dụng lợi thế của phụ-Datasheet hiệu lực. If you have created two tables, one with a primary key and another with a foreign key, you can create the sub-datasheet yourself. Nếu bạn đã tạo ra hai bảng, một với một khóa chính và khác với một khoá ngoại, bạn có thể tạo các phụ Datasheet chính mình.

To create a non-existing datasheet, open the table with the primary key in Datasheet View. Để tạo một Datasheet không tồn tại, mở bảng với khóa chính trong Datasheet View. In the Records section of the Home tab of the Ribbon, click More, position the mouse on Subdatasheet, and click Subdatasheet... This would open the Insert Subdatasheet dialog box. Trong phần Records của tab Home của Ribbon, nhấn thêm, vị trí chuột vào Subdatasheet, và nhấp vào Subdatasheet ... Điều này sẽ mở hộp thoại Insert Subdatasheet. In the list of tables (or queries), you must click the table (or query) that has the foreign key that relates to the primary key of the current table. Trong danh sách các bảng (hoặc truy vấn), bạn phải bấm vào bảng (hoặc truy vấn) mà có khoá ngoại liên quan đến khoá chính của bảng hiện hành. The names of the primary key and the foreign key would appear in the combo boxes. Tên của các khóa chính và khoá ngoại sẽ xuất hiện trong hộp kết hợp. Once you click OK, Microsoft Access would take care of configuring the subdatasheet. Một khi bạn click OK, Microsoft Access sẽ chăm sóc cấu hình subdatasheet.

If a relationship has been established and a subdatasheet exists in a table but you do not want the subdatasheet to show anymore, you can remove it. Nếu một mối quan hệ đã được thiết lập và subdatasheet một tồn tại trong một bảng nhưng bạn không muốn subdatasheet để hiển thị nữa, bạn có thể gỡ bỏ nó. To delete a subdatasheet, open the table with the primary key in Datasheet View. Để xóa một subdatasheet, mở bảng với khóa chính trong Datasheet View. In the Records section of the Home tab of the Ribbon, click More -> Subdatasheet, and click Remove. Trong phần Records của tab Home của Ribbon, nhấn More -> Subdatasheet, và nhấn Remove.

### ❖ Practical Learning: Creating a Sub-Datasheet Học tập thực tế: Tạo ra một tiểu Datasheet

1. Create a new table in Design View with the following fields: Tạo một bảng mới trong thiết kế Xem với các lĩnh vực sau đây:

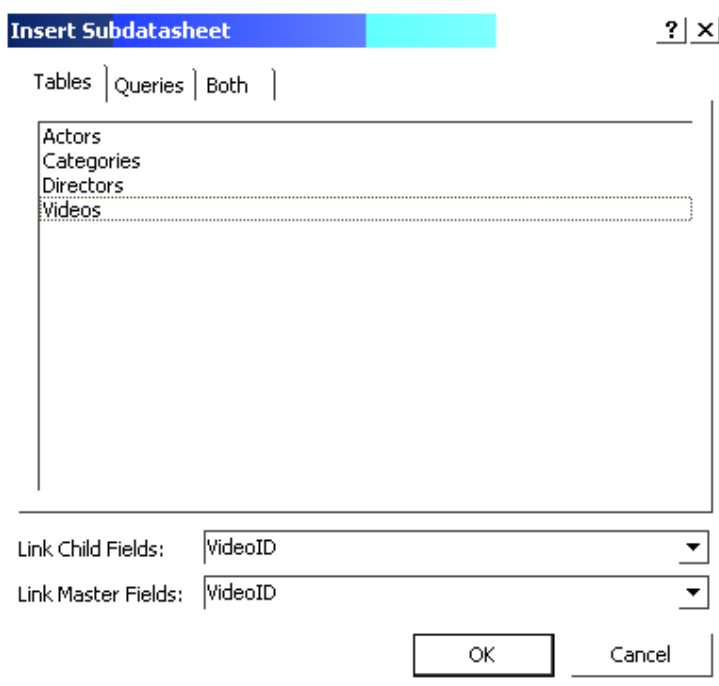
Field Name Tên trường	Data Type Loại dữ liệu	Caption Chú thích	Field Size Field Size
<b>VideoID</b> <b>VideoID</b> (Primary Key) (Primary Key)	AutoNumber AutoNumber	Video ID Video ID	
<b>Title Tiêu đề</b>	Text Văn bản		120 120
<b>DirectorID</b> <b>DirectorID</b>	Number Số	<b>Director</b> <b>Giám</b> <b>độc</b>	Long Integer Long Integer
<b>CopyrightYear</b> <b>CopyrightYear</b>	Number Số	© <b>Year</b> © <b>năm</b>	Integer Integer
<b>Length Chiều dài</b>	Number Số		Integer Integer
<b>CategoryID</b> <b>CategoryID</b>	Number Số	<b>Category</b> <b>Thể loại</b>	Long Integer Long Integer
<b>Rating Đánh giá</b>	Text Văn bản		20 20
<b>Notes Ghi chú</b>	Memo Memo		

- (To get the © character, you can open Microsoft Word, type (c) then select and copy it. Then paste it in the Caption property of the table field) (Để có được nhân vật ©, bạn có thể mở Microsoft Word, kiểu (c) sau đó chọn và sao chép nó. Sau đó dán nó vào tài sản Chú thích của lĩnh vực bảng)
- Save the table as **Videos** and switch it to Datasheet View Lưu bảng như **Video** và chuyển đổi nó để Datasheet View
- Enter a few videos: Nhập một vài video:

<b>Video ID</b> <b>Video ID</b> <b>ID</b>	<b>Title Tiêu đề</b>	<b>Director</b> <b>Giám</b> <b>độc</b>	© <b>Year</b> © <b>năm</b>	<b>Length</b> <b>Chiều</b> <b>dài</b>	<b>Category</b> <b>Thể loại</b>	<b>Rating</b> <b>Đánh</b> <b>giá</b>
1 1	Distinguished Gentleman (The) Thừa các quý ông (The)	8 8		112 112	8 8	R R
2 2	Wall Street Wall Street	12 12	1987 1987	126 126	10 10	R R
3 3	Ransom Chuộc lại	4 4		121 121	11 11	R R
4 4	Not Another Teen Movie Không khác Movie Teen	11 11	2005 2005	100 100	6 6	Unrated Unrated
5 5	Harlem Nights Harlem Nights	6 6	1989 1989	116 116	10 10	R R
6 6	M:i:III M: i: III	7 7	2006	125	19 19	PG-13

			2006	125		PG-13
7 7	Devdas Devdas	5 5		175 175	20 20	Unrated Unrated
8 8	Passion of the Christ (The) Niềm đam mê của Chúa Kitô (The)	9 9	2004 2004		21 21	R R
9 9	Platoon Tiểu đội	12 12	1986 1986	120 120	13 13	R R
10 10	Day After Tomorrow (The) Day After Tomorrow (The)	1 1	2004 2004	123 123		PG-13 PG-13
11 11	Beautiful Mind (A) Beautiful Mind (A)	4 4	2001 2001	135 135		PG-13 PG-13
12 12	Godzilla Godzilla	1 1	1998 1998	139 139		PG-13 PG-13

5. Save the table and close it Lưu bảng và đóng nó lại
6. From the Navigation Pane, double-click Directors: Table to open it Từ Cửa sổ Danh mục chính, nhấp đúp vào Giám đốc: Bảng để mở nó
7. On the Ribbon, click Home. Trên Ribbon, nhấn Home. In the Records section, click More -> Subdatasheet -> Subdatasheet... Trong phần Records, nhấp vào More -> Subdatasheet -> Subdatasheet ...
8. In the Insert Subdatasheet dialog box, make sure the Tables tab is selected. Trong hộp thoại Insert Subdatasheet, hãy chắc chắn tab Bàn được chọn. In the list box, click Videos and notice that DirectorID has been selected in the combo boxes: Trong hộp danh sách, hãy nhấp vào Video và nhận thấy rằng DirectorID đã được chọn trong hộp kết hợp:



9. Click OK Click OK





**Microsoft Office Access did not detect a relationship between the fields you have chosen.**

Do you want a relationship to be created for you now?



10. A message box information informs you and asks whether you want to create a relationship between both tables. Một hộp thông báo thông tin thông báo cho bạn và hỏi bạn có muốn tạo một mối quan hệ giữa hai bảng. Read it and click Yes Đọc nó và nhấn Yes
11. Click a + button and notice the related records. Nhấp vào nút + và thông báo các hồ sơ liên quan. Close the Directors table Đóng bảng Giám đốc

DirectorID	Director	Notes	Add New Field
-	1 Roland Emmerich		
	Video ID	Title	© Year Length
	10	Day After Tomorrow (The)	2004 123
	12	Godzilla	1998 139
*	(New)		
+	2 Renny Harlin		
+	3 Tony Scott		
-	4 Ron Howard		
	Video ID	Title	© Year Length
	3	Ransom	121
	11	Beautiful Mind (A)	2001 135
*	(New)		
+	5 Sanjay Leela Bhansali		
-	6 Eddie Murphy		
	Video ID	Title	© Year Length
	5	Harlem Nights	1989 116
*	(New)		
+	7 J. J. Abrams		
+	8 Jonathan Lynn		
-	9 Mel Gibson		
	Video ID	Title	© Year Length
	8	Passion of the Christ (The)	2004
*	(New)		
+	10 Steven Spielberg		

12. When asked whether you want to save, click Yes Khi được hỏi liệu bạn muốn lưu, click Yes

## Referential Integrity Tham chiếu Liên

### Introduction Giới thiệu

At this time, we know that there is useful functionality to creating relations between tables as they allow the user to select existing information instead of typing it. Vào lúc này, chúng ta biết rằng có hữu ích chức năng để tạo ra các mối quan hệ giữa các bảng khi họ cho phép người dùng chọn thông tin hiện có thay vì gõ nó. This flow of information brings up issues about what happens if data that exists in a parent table gets deleted while such information has been made available to another table. Dòng chảy của thông tin này sẽ trả về các vấn đề

về những gì sẽ xảy ra nếu dữ liệu tồn tại trong một bảng cha sẽ bị xóa khi thông tin đó đã được làm sẵn để bàn khác.

When manipulating data that is in a relationship, it is very important to make sure that data keeps its accuracy from one table or source to the other. Khi thao tác dữ liệu đó là trong mối quan hệ, nó là rất quan trọng để đảm bảo rằng dữ liệu giữ độ chính xác của nó từ một bảng hoặc nguồn khác. To accomplish that goal, some rules must be established to "watch" or monitor the flow of information between two tables. Để thực hiện được mục tiêu đó, một số quy tắc phải được thành lập để "xem" hoặc theo dõi luồng thông tin giữa hai bảng. Data or referential integrity is used to check that two tables are related through one (sometimes more than one) field on each table used as the primary key and the foreign key, data entered in the foreign key of a child table must exist in the parent table, otherwise it would be rejected. Dữ liệu hoặc toàn vẹn tham chiếu được sử dụng để kiểm tra xem hai bảng có liên quan thông qua một (hơn một) trường đôi khi nhiều hơn trên mỗi bảng được sử dụng như là khóa chính và ngoại khóa, dữ liệu nhập vào chính nước ngoài của một bảng con phải tồn tại trong cha mẹ bảng, nếu không nó sẽ bị từ chối. Only two fields of the same data type are used to establish a relationship between two tables, the tables involved in the relationship belong to the same database. Chỉ có hai lĩnh vực của cùng một kiểu dữ liệu được sử dụng để thiết lập một mối quan hệ giữa hai bảng, các bảng liên quan đến mối quan hệ thuộc về cùng một cơ sở dữ liệu.

### Cascades on Related Records Dãy núi Cascade trên Related Records

After creating a legitimate relationship between two tables, you must make sure that when data changes in the parent table, this change is reflected in the child table. Sau khi tạo một mối quan hệ hợp pháp giữa hai bảng, bạn phải chắc chắn rằng khi thay đổi dữ liệu trong bảng cha mẹ, sự thay đổi này được phản ánh trong bảng con. For example, if a bank customer changes her last name after getting married or after a divorce, you should be able to change her name in one object (table) and the related objects, such as the one used to process her transactions would receive the changes without your having to make the change on each object (table). Ví dụ, nếu một ngân hàng thay đổi của khách hàng tên cuối cùng sau khi kết hôn hoặc sau khi ly hôn, bạn sẽ có thể thay đổi tên của mình trong một đối tượng (bảng) và các đối tượng liên quan, chẳng hạn như một trong những sử dụng để quá trình giao dịch của cô sẽ nhận được thay đổi mà không cần phải của bạn để làm cho sự thay đổi trên từng đối tượng (bảng). In the same way, when data is deleted, the objects that are related to it must also have that data deleted. Trong cùng một cách, khi dữ liệu bị xóa, các đối tượng có liên quan đến nó cũng phải có các dữ liệu bị xóa.

To enforce the rules of data integrity, Microsoft Access provides three check boxes in the Edit Relationship dialog box. Để thực thi các quy tắc của toàn vẹn dữ liệu, Microsoft Access cung cấp ba hộp kiểm trong hộp thoại Edit Mối quan hệ. First, if you want Microsoft Access to monitor data flow, you can click the Enforce Referential Integrity check box. Trước tiên, nếu bạn muốn Microsoft Access để theo dõi lưu lượng dữ liệu, bạn có thể bấm vào Enforce toàn vẹn tham chiếu kiểm tra hộp. This would make available two check boxes. Điều này sẽ làm cho các hộp kiểm tra có sẵn hai.

### The Direction of a Relationship Các hướng của một mối quan hệ

#### The One-To-Many Relationship Mối quan hệ One-To-Nhiều

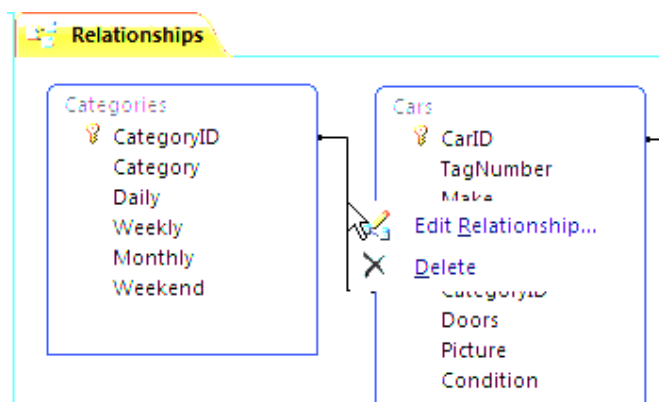
As mentioned already, a relationship between two tables allows one table, the parent, to make its information available to another table (the child). Như đã đề cập đã có, một mối quan hệ giữa hai bảng cho phép một bảng, phụ huynh, để làm cho thông tin của nó có sẵn

cho một bảng (các con). Because in this case the user is asked to select information, it is likely that the same record on a parent table can be tied to various records in the child table. Bởi vì trong trường hợp này người sử dụng được yêu cầu chọn thông tin, có khả năng là cùng một bản ghi trên một bảng cha có thể được gắn với các hồ sơ khác nhau trong bảng con. For example, one customer at a bank can deposit an amount of money today. Ví dụ, một khách hàng tại một ngân hàng có thể đặt cọc một số tiền ngày hôm nay. The same customer can make another deposit tomorrow and even another deposit next month. Cùng một khách hàng có thể thực hiện một khoản tiền gửi vào ngày mai và thậm chí cả tiền gửi khác vào tháng tới. In such a case, the relationship between the tables would show various entries of the same customer's account number in the object (table) used to deposit money but with different transactions. Trong trường hợp này, mối quan hệ giữa các bảng sẽ hiển thị các mục khác nhau của số tài khoản cùng một khách hàng trong đối tượng (bảng) được sử dụng để gửi tiền nhưng với các giao dịch khác nhau. This type of relationship is known as one-to-many because one entry in the parent table can result in many entries in the child table. Loại mối quan hệ được biết đến như một-nhiều bởi vì một trong những mục trong bảng phụ huynh có thể dẫn đến nhiều mục trong bảng con.

To create a one-to-many relationship, check all three referential integrity check boxes and click Create. Để tạo một-to-một trong nhiều mối quan hệ, hãy kiểm tra cả ba hộp kiểm tra tính toàn vẹn tham chiếu và Tạo nhấp chuột. The parent table would have a 1 sign on its side of the joining line. Bảng cha sẽ có một dấu hiệu 1 ở thành của nó của đường gia nhập. The child table would have the infinity symbol on its side of the joining line. Bảng con sẽ có biểu tượng vô cùng của nó về phía của đường gia nhập.

#### ❖ Practical Learning: Managing Referential Integrity Học tập thực tế: Quản lý toàn vẹn tham chiếu

1. Re-open the Bethesda Car Rental2 database that you had created earlier Mở lại Bethesda xe Rental2 cơ sở dữ liệu mà bạn đã tạo ra trước đó
2. On the Ribbon, click Database Tools and, in the Show/Hide section, click Relationships Trên Ribbon, nhấn Tools và cơ sở dữ liệu, trong phần Show Ẩn /, hãy nhấp vào mối quan hệ
3. Right-click the line between Categories and Cars. Kích chuột phải vào dòng giữa mục và Cars. Click Edit Relationship... Nhấp vào Edit Mối quan hệ ...



4. In the Edit Relationship dialog box, click the Enforce Referential Integrity check box Trong hộp thoại Edit Mối quan hệ, hãy nhấp vào hộp kiểm tra tham chiếu Enforce Liên

5. Click the two check boxes under it Nhấp vào hai hộp kiểm tra dưới nó

**Edit Relationships** [?] [X]

Table/Query: Categories    Related Table/Query: Cars

CategoryID    CategoryID

Enforce Referential Integrity  
 Cascade Update Related Fields  
 Cascade Delete Related Records

Relationship Type: One-To-Many

OK  
Cancel  
Join Type...  
Create New...

6. Click OK. Click OK.

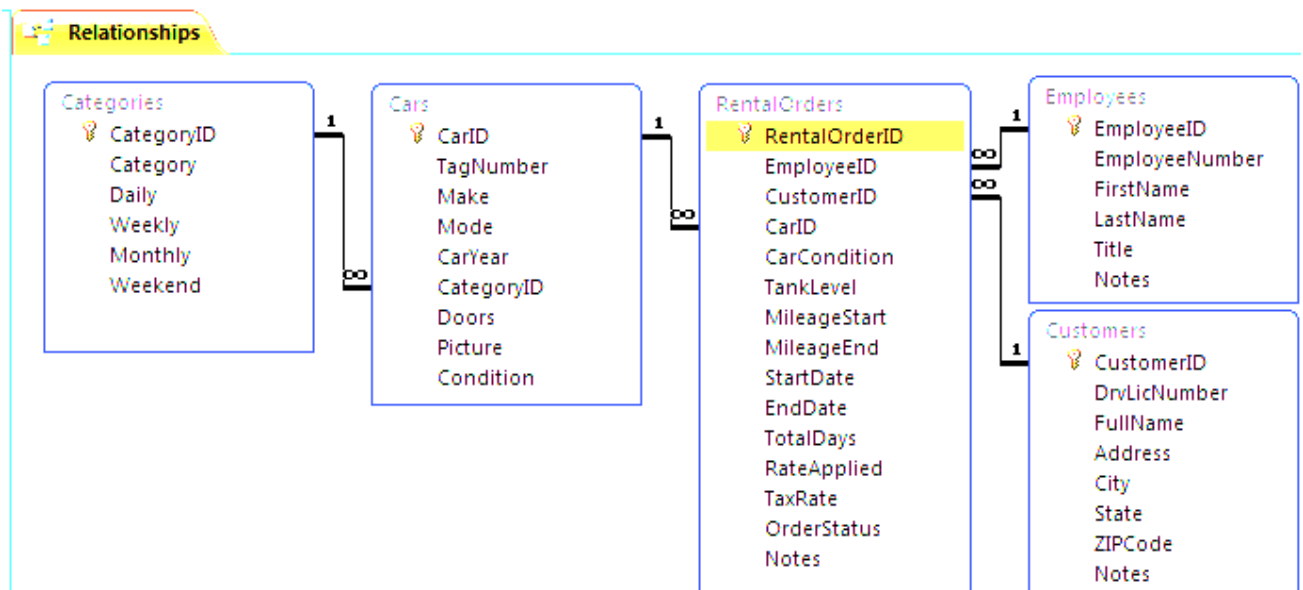
Notice the 1 and the ∞ symbol Thông báo 1 và biểu tượng ∞

7. Double-click the line between Cars and Rental Orders Click đôi vào dòng giữa Ô tô và thuê đơn đặt hàng

8. In the Edit Relationship dialog box, click the Enforce Referential Integrity check box and click the two check boxes under it Trong hộp thoại Edit Mỗi quan hệ, hãy nhấp vào hộp kiểm Enforce tham chiếu Liên và nhấp vào hai hộp kiểm tra dưới nó

9. Click OK Click OK

10. Do the same for the other two remaining lines Làm tương tự cho hai dòng khác còn lại



11. Save and close the Relationships window Lưu và đóng cửa sổ mỗi quan hệ

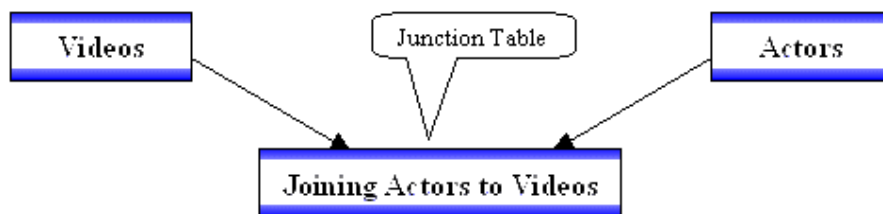
### The Many-to-Many Relationship Mỗi quan hệ nhiều-to-Nhiều

Although one-to-many is the most common type of relationship applied to records of a table, in some databases, you may need to create a relationship in which many records from one table A can have many related records in another table B and vice versa. Mặc dù một-nhiều là loại phổ biến nhất của mỗi quan hệ áp dụng cho các bản ghi của một bảng, trong một số cơ sở dữ liệu, bạn có thể cần để tạo ra một mỗi quan hệ trong đó có nhiều bản ghi từ một bảng A có thể có liên quan nhiều lưu trong một bảng B và ngược ngược lại. This type of relationship is known as many-to-many. Loại mỗi quan hệ được gọi là nhiều-

nhieu. For example, in our Video Collection database: Ví dụ, trong cơ sở dữ liệu Bộ sưu tập Video của chúng tôi:

- It is possible to have one video that has many actors. Có thể có một video có diễn viên nhiều. In this case, if we had created a field to receive actors in the Videos table, we would enter many names of actors in that one field but this type of database would not be professional Trong trường hợp này, nếu chúng ta đã tạo ra một lĩnh vực để nhận được các diễn viên trong bảng Videos, chúng tôi sẽ nhập nhiều tên của các diễn viên trong một lĩnh vực mà cơ sở dữ liệu kiểu này sẽ không được chuyên nghiệp
- At the same time, one actor can have participated in many videos. Đồng thời, diễn viên người ta có thể đã tham gia nhiều video. In this case, if we had created a field to enter the titles of videos in the Actors table, the field would have too many entries, making it unprofessional Trong trường hợp này, nếu chúng ta đã tạo ra một trường để nhập vào các chức danh của video trong bảng Diễn viên, trường sẽ có nhiều mục quá, làm cho nó học trẻ không chuyên

To implement this type of relationship, you can create what is called a junction table. Để thực hiện mối quan hệ kiểu này, bạn có thể tạo ra những gì được gọi là một bảng đường giao nhau. A junction table is a table whose main purpose is to bring together fields from other tables, creating a type of cross relationship for the necessary fields: Một bảng giao nhau là một bảng mà mục đích chính là để mang lại cùng lĩnh vực từ các bảng khác, tạo ra một loại mối quan hệ xuyên cho các lĩnh vực cần thiết:



A junction table contains at least three fields. Một bảng chứa đường giao nhau ít nhất ba lĩnh vực. The first field, almost less engaged, is used as the primary key, the same type of field almost every table has. Trường đầu tiên, hầu như ít tham gia, được sử dụng như là khóa chính, cùng một loại trường hầu như mỗi bảng có. The other fields hold data that would originate from other tables. Các lĩnh vực khác giữ dữ liệu mà có thể bắt nguồn từ các bảng khác.

You can create a junction table either in Datasheet View or in Design View: Bạn có thể tạo một bảng đường giao nhau hoặc trong Datasheet View hoặc trong thiết kế xem bài viết:

- To create a junction table in Datasheet View, you use the Lookup Column menu that would call the Lookup Wizard. Để tạo một bảng giao nhau trong Datasheet View, bạn sử dụng Lookup Cột menu mà có thể gọi các Wizard Lookup. From the Lookup Wizard, select the table that holds the desired field, then select the field itself, and click Finish. Từ Lookup Wizard, chọn bảng chứa các lĩnh vực mong muốn, sau đó chọn lĩnh vực chính nó, và nhấn Finish. After adding the first column, repeat the same steps for each needed field Sau khi thêm các cột đầu tiên, lặp lại các bước tương tự cho từng lĩnh vực cần thiết
- To create a junction table in Design View, set the field's data type to Lookup Wizard and proceed the same way you would for the Lookup Column Để tạo một bảng giao nhau trong thiết kế View, thiết lập kiểu dữ liệu của trường để Lookup Wizard và tiến

hành giống như cách bạn sẽ cho cột Lookup

You can also switch to either view to create a junction table. Bạn cũng có thể chuyển đổi để xem, hoặc để tạo ra một bảng đường giao nhau. In other words, you can create one field in one view and create the other field in the other view. Nói cách khác, bạn có thể tạo một trường trong một xem và tạo ra các lĩnh vực khác trong giao diện khác.

❖ Practical Learning: Creating Junction Tables Học tập thực hành: Tạo bảng Junction

1. Open the Video Collection3 database you started earlier Mở cơ sở dữ liệu Video Collection3 bạn bắt đầu trước đó
2. To create a new table, on the ribbon, click Create and, in the Table section, click Table Design Để tạo một bảng mới, trên ribbon, nhấp vào Tạo và, trong phần Table, hãy nhấp vào Thiết kế Bảng
3. Type **VideoActorID** as the name of the first field Loại **VideoActorID** như tên gọi của trường đầu tiên
4. Set its **Data Type** to **AutoNumber** Đặt **Loại dữ liệu** của nó để **AutoNumber**
5. Right-click the **VideoActorID** name and click **Primary Key** Kích chuột phải vào tên **VideoActorID** và nhấp vào **tiểu học khóa**
6. Complete the table with the following two fields: Hoàn thành bảng với hai lĩnh vực sau đây:

Field Name Tên trường	Data Type Loại dữ liệu	Caption Chú thích	Field Size Field Size
VideoActorID VideoActorID (Primary Key) (Primary Key)	AutoNumber AutoNumber	Video Actor ID Video diễn viên ID	
<b>VideoID</b> <b>VideoID</b>	Number Số	Video Video	Long Integer Long Integer
<b>ActorID</b> <b>ActorID</b>	Number Số	Actor Diễn viên	Long Integer Long Integer

7. Save the table as **VideosAndActors** Lưu bảng như **VideosAndActors**
8. Switch it to Datasheet View Chuyển đổi nó để Datasheet View
9. Enter the records as follows: Nhập các bản ghi như sau:

VideoActorID	VideoID	ActorID
1 1	1 1	1 1
2 2	3 3	4 4
3 3	2 2	23 23
4 4	4 4	6 6
5 5	1 1	7 7
6 6	9 9	5 5
7 7	6 6	2 2
8 8	4 4	14 14

9 9	1 1	9 9
10 10	2 2	8 8
11 11	3 3	19 19
12 12	1 1	10 10
13 13	6 6	17 17
14 14	3 3	20 20
15 15	1 1	11 11
16 16	4 4	16 16
17 17	3 3	18 18
18 18	1 1	15 15
19 19	12 12	21 21
20 20	6 6	3 3
21 21	2 2	5 5
22 22	1 1	13 13
23 23	5 5	1 1
24 24	12 12	22 22
25 25	1 1	12 12

#### 10. Close the table Đóng bảng

### The One-to-One Relationship Mỗi quan hệ One-to-One

A one-to-one relationship is the type of junction between two tables A and B so that one record in table A can have only one corresponding entry in table B and vice versa. A-to-một trong những mối quan hệ là một trong những loại đường giao nhau giữa hai bảng A và B để cho một bản ghi trong bảng A có thể chỉ có một mục tương ứng trong bảng B và ngược lại. Because this is similar to one table of records, this type of relationship is hardly used since you can as well simply create one table. Bởi vì đây là tương tự như một bảng ghi chép, mỗi quan hệ kiểu này là khó sử dụng vì bạn có thể cũng chỉ cần tạo một bảng.

### Data Joins Dữ liệu tham gia

#### Introduction Giới thiệu

When creating relationships among tables, we were selecting the primary key of one table and the foreign key of a dependent table to join them. Khi mỗi quan hệ giữa các bảng tạo, chúng tôi đã chọn khóa chính của một bảng và chính nước ngoài của một bảng phụ thuộc tham gia cùng họ. Once such a relationship is created, you can create a query that combines both tables to create a set of records, also called a record set, that would include either all records or isolate only the records that have entries. Khi đó một mối quan hệ được tạo ra, bạn có thể tạo một truy vấn mà kết hợp cả hai bảng để tạo ra một bộ hồ sơ, còn được gọi là một bộ hồ sơ, mà có thể bao gồm cả tất cả các bản ghi hoặc chỉ cô lập các hồ sơ có mục. For example, imagine you have created a Persons table as follows where the Gender of a record is selected from a lookup field: Ví dụ, hãy tưởng tượng bạn đã tạo một bảng Người như sau, nơi ghi lại giới tính của một được chọn từ một lĩnh vực tra cứu:

PersonID	First Name	Last Name	Gender
1	Anthony	Jeffereys	Male
2	Joan	Scarsdale	Female
3	Willie	Danton	
4	Jason	Aachen	Male
5	Paula	Brookes	Female
6	Rick	Dogley	
7	Deborah	O'Neil	Female
8	Alex	Pearlman	

\* (New)

Suppose you want to create a query that includes the persons of this table and their genders. Giả sử bạn muốn tạo một truy vấn bao gồm những người của bảng này và giới tính của họ. A question that comes is mind is: Do you want to create a list of only people who can be recognized by their gender, or do you want the list to include everybody? Một câu hỏi mà đến là tâm là: Bạn có muốn tạo một danh sách những người duy nhất có thể được công nhận bởi giới tính của họ, hay bạn muốn trong danh sách bao gồm tất cả mọi người? This concept leads to what are referred to as joins of queries. Khái niệm này dẫn đến những gì được gọi là tham gia các câu truy vấn. There are two ways you can get such queries: you can prepare the relationship between two tables to be aware of this type of relationship or you can directly create it when designing a query. Có hai cách bạn có thể nhận được các truy vấn như: bạn có thể chuẩn bị các mối quan hệ giữa hai bảng để được nhận thức của loại hình này mỗi quan hệ hoặc bạn có thể trực tiếp tạo ra nó khi thiết kế một truy vấn.

### Inner Joins Nội tham gia

When building a query, you select fields and ask Microsoft Access to isolate them as being part of the query. Khi xây dựng một truy vấn, bạn chọn các lĩnh vực và yêu cầu Microsoft Access để cô lập chúng như là một phần đang được các truy vấn. Most of the time, you will want only fields that include a type of validation of your choice (a criterion). Hầu hết thời gian, bạn sẽ muốn các lĩnh vực duy nhất mà bao gồm một loại xác nhận của sự lựa chọn của bạn (một tiêu chuẩn). An inner join is the kind of query that presents only fields that have matching entries on both tables of a relationship. Một bên trong tham gia là loại truy vấn mà chỉ trình bày các lĩnh vực có kết hợp các mục trên cả hai bàn của mỗi quan hệ. For example, from the above table, if you want to create a query that includes only persons whose records contain the gender: Ví dụ, từ bảng trên, nếu bạn muốn tạo một truy vấn bao gồm những người chỉ có hồ sơ bao gồm các giới:

First Name	Last Name	Gender
Anthony	Jeffereys	Male
Joan	Scarsdale	Female
Jason	Aachen	Male
Paula	Brookes	Female
Deborah	O'Neil	Female

\* (New)

You would create it as an inner join. Bạn sẽ tạo ra nó như là một tham gia trong. To specify that a relationship is inner join, after creating the relationship, while in the Relationships,



you can display its Edit Relationship dialog box and click Join Type... The Join Properties dialog box presents three options that allow you to define the direction of the relationship between the tables. Để xác định rằng một mối quan hệ bên trong tham gia, sau khi tạo mối quan hệ, trong khi ở các mối quan hệ, bạn có thể hiển thị hộp Edit Mối quan hệ của nó hộp thoại và nhấp Tham gia Type ... Các hộp thoại Properties Tham gia trình bày ba tùy chọn cho phép bạn xác định hướng của mỗi quan hệ giữa các bảng.

To specify an inner join in a SQL statement, you include **INNER JOIN** in the statement. Để chỉ định tham gia nội trong một câu lệnh SQL, bạn bao **gồm, bên trong JOIN** trong bản Tuyên Bố này.

### ❖ Practical Learning: Creating Inner Join Relationships Học tập thực hành: Tạo mối quan hệ Nội Tham gia

---

1. The Video Collection database should still be opened. Cơ sở dữ liệu Video Bộ sưu tập vẫn phải được mở ra.

On the Ribbon, click Database Tools. Trên Ribbon, nhấn Tools Cơ sở dữ liệu. In Show/Hide section, click the Relationships button Trong phần Show Ẩn /, nhấn vào



nút các mối quan hệ

2. If either of the Videos or the Directors table is not displaying, right-click an area in the Relationships window and click Show Tables... Nếu một trong hai của Video hoặc Giám đốc bàn là không được hiển thị, kích chuột phải vào một khu vực trong cửa sổ Mối quan hệ và nhấp vào Hiển thị Bàn ...  
In the list of tables, double-click the missing table(s) (Directors and/or Videos) and click Close. Trong danh sách các bảng, kích đúp vào bảng mất tích (s) (Giám đốc và / hoặc Video) và kích Close.  
Click the joining line between the Directors and Videos Click vào dòng gia nhập giữa Giám đốc và Video
3. In the Tools section of the Ribbon, click Edit Relationships Trong phần Công cụ của Ribbon, nhấp vào Chỉnh sửa các mối quan hệ
4. After making sure that DirectorID is selected in each combo box, click the Enforce Referential Integrity button and click the two check boxes under it Sau khi chắc chắn rằng DirectorID được chọn trong mỗi hộp kết hợp, hãy nhấp vào nút Enforce toàn vẹn tham chiếu và nhấp vào hai hộp kiểm tra dưới nó
5. Click the Join Type button. Nhấp vào nút Join Type.  
In the Join Properties dialog box, click the first radio button (it should be selected already) Trong hộp thoại Properties Tham gia, nhấp vào nút radio đầu tiên (nó cần được lựa chọn đã được)
6. Click OK twice Nhấp OK hai lần

### Outer Joins Ngoài tham gia

---

The queries we have used so far and that were based on related tables allowed us to get only the fields that had entries based on the established relationships. Các truy vấn chúng tôi đã được sử dụng cho đến nay và được dựa trên các bảng liên quan cho phép chúng tôi để có được chỉ các lĩnh vực mà có các mục dựa trên các mối quan hệ được thiết lập. Fields that did not follow the rules were excluded. Những lĩnh vực mà không tuân theo các quy tắc đã được loại trừ. Instead of excluding fields, the SQL allows you to create a query that includes all fields, not just those that follow rules, as long as the records are part of either table. Thay vì không bao gồm các lĩnh vực, SQL cho phép bạn tạo ra một truy vấn bao gồm

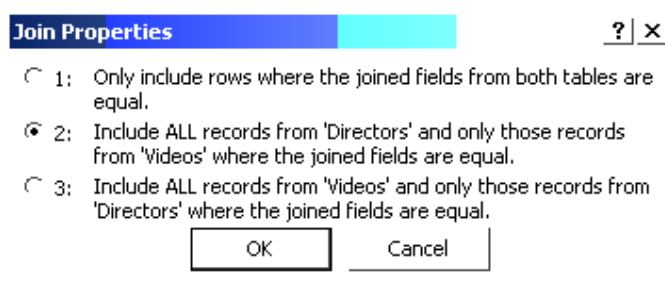
tất cả các lĩnh vực, không chỉ những người mà theo quy định, miễn là các hồ sơ là một phần của bảng, hoặc. Such a query is referred to as outer join. Như một truy vấn được gọi là bên ngoài tham gia. To manage the result of this type of query, the SQL considers the direction of a relationship. Để quản lý kết quả của các loại truy vấn, SQL xem xét theo hướng mỗi quan hệ.

When creating relationships, we learned to drag a primary key from one table, the parent, to a dependent table, the child. Khi tạo mỗi quan hệ, chúng tôi đã học để kéo một khóa chính từ một bảng, phụ huynh, để một bảng phụ thuộc, các trẻ em. In this type of relationship, the table (or query) that holds the origin of the relationship is referred to as the Left table. Trong mỗi quan hệ kiểu này, bảng (hoặc truy vấn) giữ nguồn gốc của mỗi quan hệ này được gọi là bảng còn lại. The other table is referred to as the right table. Các bảng khác được gọi là bảng bên phải. Based on this, there are two types of outer joins: the left join represented in SQL as **LEFT JOIN** and the right join represented by **RIGHT JOIN**. Trên cơ sở này, có hai loại bên ngoài tham gia: bên trái tham gia đại diện trong SQL như **trái JOIN** và quyền tham gia đại diện bởi **RIGHT JOIN**.

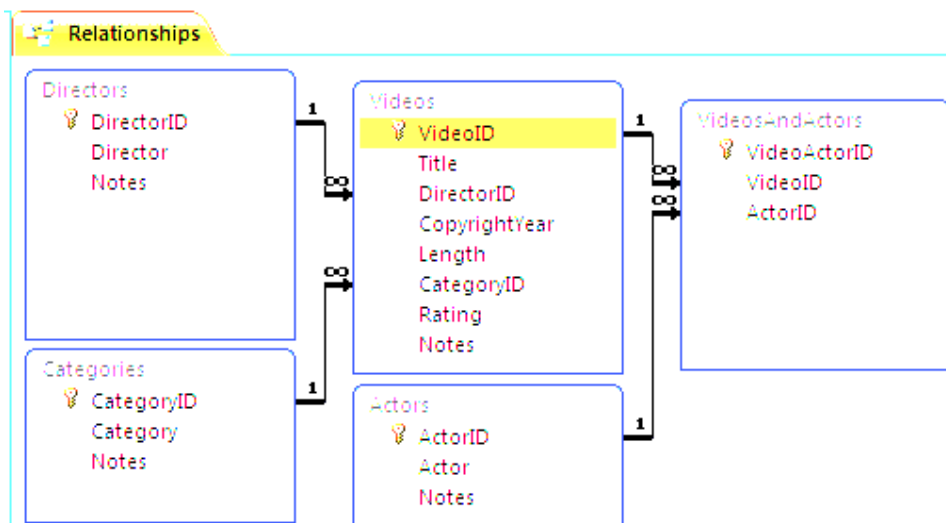
As done with the inner join relationship, the left and right joins can be prepared in the Relationships window on tables. Khi thực hiện với sự tham gia mỗi quan hệ bên trong, bên trái và bên phải tham gia có thể được lập trong cửa sổ các mối quan hệ trên bảng. If the query has already been created and you want to change its direction, you can change it in the Design View of the query. Nếu truy vấn đã được tạo ra và bạn muốn thay đổi hướng của nó, bạn có thể thay đổi nó trong Xem Thiết kế truy vấn.

#### Practical Learning: Creating Outer Join Relationships Học tập thực hành: Tạo mỗi quan hệ ngoài gia

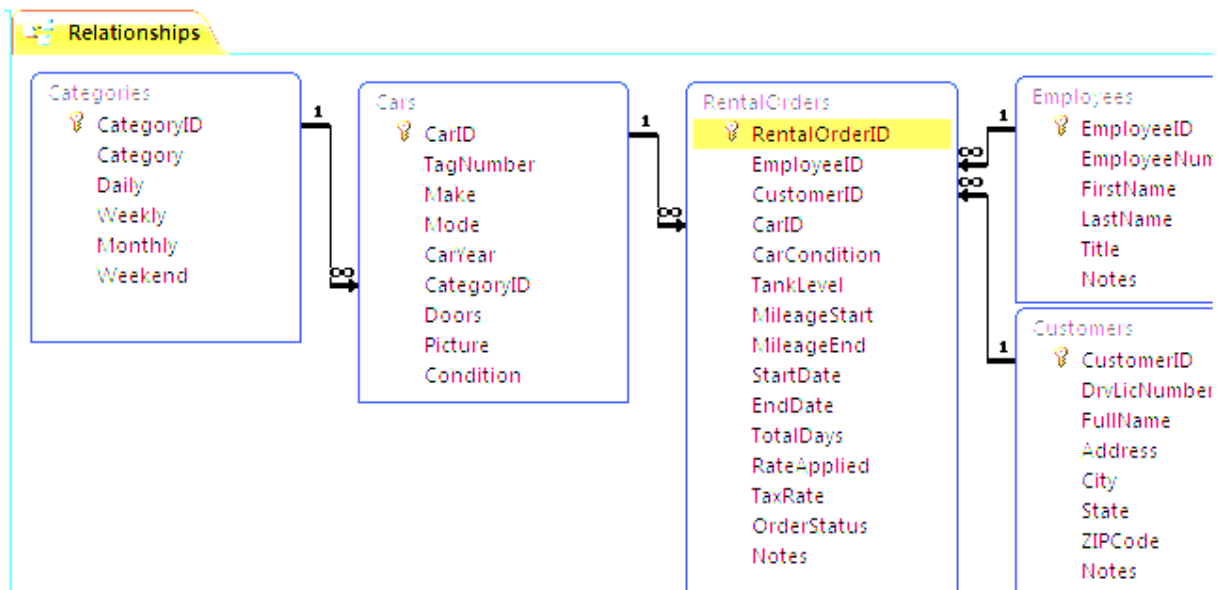
1. The Video Collection3 database should still be opened with the Relationships window displaying. Các cơ sở dữ liệu Collection3 Video vẫn còn phải được mở ra mỗi quan hệ với các cửa sổ hiển thị.  
In the Relationships button, double-click the joining line between Directors and Videos Trong nút Quan hệ, kích đúp vào dòng gia nhập giữa Giám đốc và Video
2. In the Edit Relationship dialog box, click Join Type Trong hộp thoại Edit Mỗi quan hệ, hãy nhấp vào Tham gia Type
3. In the Join Properties dialog box, read all options then click the second radio button Trong hộp thoại Properties Tham gia, đọc tất cả các tùy chọn sau đó bấm vào nút chọn thứ hai



4. Click OK and OK Click OK và OK
5. In the same way, complete the relationships of the Video Collection database (when the relationship does not exist, you must create it) Trong cùng một cách, hoàn chỉnh các mối quan hệ của cơ sở dữ liệu Bộ sưu tập Video (khi các mối quan hệ không tồn tại, bạn phải tạo nó)




6. Save and close the Relationships window Lưu và đóng cửa sổ mối quan hệ
7. Open the Bethesda Car Rental2 database Mở cơ sở dữ liệu Bethesda xe Rental2
8. Open the Relationships window and configure all relationships as follows: Mở cửa sổ Mối quan hệ và cấu hình tất cả các mối quan hệ như sau:



9. Save and close the Relationships window Lưu và đóng cửa sổ mối quan hệ
10. On the Ribbon, click Create and, in the Forms section, click Form Design Trên Ribbon, nhấp vào Tạo và, trong phần hình thức, kích Thiết kế Form
11. Save the form as Cars and change the following characteristics in the Properties window: Lưu hình thức như Ô tô và thay đổi các đặc điểm sau trong cửa sổ Properties:  
**Record Source: Cars Record Source: Ô tô**  
**Caption: Bethesda Car Rental - Cars Chú thích: Bethesda Cho thuê xe - Ô tô**
12. Design the form as you see fit. Thiết kế dạng như bạn thấy phù hợp. Here is an example (no need to match the fonts; use whatever font you have on your computer): Dưới đây là một ví dụ (không cần phải phù hợp với phông chữ, sử dụng bất cứ font bạn có trên máy tính của bạn):

13. Save the form and switch it to Form View Lưu mẫu và chuyển đổi nó vào Form View
14. [Create the cars records](#) and make sure you add the pictures from the resources of these lessons [Tạo ra những chiếc xe hồ sơ](#) và chắc chắn rằng bạn thêm những hình ảnh từ các nguồn tài nguyên của những bài học này
15. Close the form Đóng mẫu

### A Relationship Report Một mối quan hệ báo

After creating a relationship or while working on relationships, you can create a visual report of the result and be able to print it when necessary. Sau khi tạo một mối quan hệ hay trong khi làm việc trên các mối quan hệ, bạn có thể tạo một báo cáo trực quan của kết quả và có thể in nó khi cần thiết. To create the report, in the Tools section of the Ribbon, click the Relationship Report button Để tạo báo cáo, trong mục Tools của Ribbon, nhấn vào nút Báo cáo Mối quan hệ . This action would automatically generate a report with the relationships designed on it. sẽ. hành động này sẽ tự động tạo ra một báo cáo với các mối quan hệ được thiết kế trên nó. To keep the report, you should save it and give it a name. Để giữ cho các báo cáo, bạn nên lưu nó và cho nó một tên. You can then print it as you would print any other report. Sau đó bạn có thể in nó như là bạn sẽ in báo cáo nào khác.

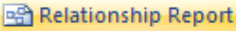
### Practical Learning: Creating a Relationship Report Học tập thực hành: Tạo một Báo cáo Mối quan hệ

1. The Bethesda Car Rental2 database should still be opened. Các cơ sở dữ liệu Rental2 Bethesda xe vẫn còn phải được mở ra.  
On the Ribbon, click Database Tools Trên Ribbon, nhấn Tools Cơ sở dữ liệu

- In the Show/Hide section, click the Relationships button Trong phần Show Ẩn /, nhấn



vào nút các mối quan hệ

- In the Tools section of the Ribbon, click the Relationship Report button Trong phần Công cụ của Ribbon, nhấn vào nút Báo cáo Mối quan hệ 
- When the report has been created, close it Khi báo cáo đã được tạo ra, đóng nó
- When asked whether you want to save, click Yes Khi được hỏi liệu bạn muốn lưu, click Yes
- Type **Bethesda Car Rental Relationships** as the name of the report and click OK Loại **Bethesda Cho thuê xe Các mối quan hệ** như tên gọi của báo cáo và nhấn OK
- In the Navigation Pane, right-click the Bethesda Car Rental Relationships report and click Print... Trong Navigation Pane, kích chuột phải vào Bethesda Cho thuê xe Các mối quan hệ báo cáo và In click ...
- Close the database Đóng cơ sở dữ liệu

### Lesson Summary Tóm tắt bài học

MCAS: Using Microsoft Office Access 2007 Topics MCAS: Sử dụng truy cập vào Microsoft Office 2007 Chủ đề

S2 S2	Define and print table relationships Xác định và các mối quan hệ bảng in
-------	--

### Exercises Các bài tập

#### Watts A loan Watts Một khoản cho vay

- Open the Watts A Loan database. Mở Watts Một Loan cơ sở dữ liệu.  
Open the CustomersTransactions form in Design View. Mở mẫu CustomersTransactions trong thiết kế View. Add a Text Box below the subform. Thêm một hộp văn bản dưới đây subform này. Set its Name to **txtTotalPayments** then set its **Format** to **Currency** with 2 **Decimal Places**. Make it get its value from the **txtTransactions** text box of the **sbfAccountTransactions** form Thiết lập Tên của nó để **txtTotalPayments** sau đó thiết lập **định dạng** của nó để **tiền tệ** với 2 **Decimal Places**. Làm cho nó có được giá trị của nó từ hộp văn bản **txtTransactions** của **sbfAccountTransactions** mẫu  
Add another Text Box below the subform and change its properties as follows: Thêm một văn bản dưới đây subform Box và thay đổi thuộc tính của nó như sau:  
**Name: txtCurrentBalance Tên: txtCurrentBalance**  
**Control Source: =DLookup("LoanAmount", "LoanProcessing", "CustomerID = " & CustomerID) - Nz(txtTotalPayments)** Kiểm soát Nguồn: = **DLookup("LoanAmount", "LoanProcessing", "CustomerID = " & CustomerID) - New Zealand (txtTotalPayments)**  
**Format: Currency Định dạng: tiền tệ**  
**Decimal Places: 2 Decimal Places: 2**  
Save and close the form Lưu và đóng biểu mẫu
- Create a new table in Design View with the following fields: Tạo một bảng mới trong thiết kế Xem với các lĩnh vực sau đây:

Field Name Tên trường	Data Type Loại dữ liệu	Additional Information Thông tin bổ sung
LoanEvaluationID	AutoNumber	Primary Key Tiểu Key
LoanEvaluationID	AutoNumber	Caption: Loan Evaluation

		ID Chú thích: Loan đánh giá ID
LoanAmount LoanAmount	Currency Tiền tệ	Caption: Loan Amount Chú thích: Số tiền vay Default Value: 0 Giá trị mặc định: 0
InterestRate InterestRate	Number Số	Field Size: Double Field Kích thước: Double Format: Percent Định dạng: Phần trăm Caption: Interest Rate Chú thích: Lãi suất Default Value: 0.0875 Giá trị mặc định: 0,0875
NumberOfPeriods NumberOfPeriods	Number Số	Field Size: Integer Field Size: Integer Caption: NumberOfPeriods Chú thích: NumberOfPeriods Default Value: 12 Giá trị mặc định: 12

3. Save the table as LoanEvaluation and close it Lưu bảng như LoanEvaluation và đóng nó lại

4. Create a new form based on the LoanEvaluation table Tạo một hình thức mới dựa trên các bảng LoanEvaluation

Save the form as **LoanEvaluation** Lưu hình thức như **LoanEvaluation**

Add a Text Box in its Detail section and set its properties as follows: Thêm một hộp văn bản trong phần chi tiết của nó và thiết lập các thuộc tính của nó như sau:

**Name: txtPeriodicPayment Tên: txtPeriodicPayment**

**Control Source: Kiểm soát Nguồn:**

**=Abs(Pmt(Nz(CDbI([InterestRate]))/12, = Abs (PMT (NZ (CDbl ([InterestRate])) / 12,**

**Nz(CInt([NumberOfPeriods]),Nz(CDbI([LoanAmount])),0,0)) Nz (CInt ([NumberOfPeriods]), New Zealand (CDbl ([LoanAmount])), 0,0))**

**Format: Currency Định dạng: tiền tệ**

**Decimal Places: 2 Decimal Places: 2**

Use the Command Button Wizard to add a button that can be used to close the form

Sử dụng Command Button Wizard để thêm một nút có thể được dùng để đóng mẫu

Design the form as follows: Thiết kế hình thức như sau:

Disable the Maximize button and make the form Pop Up. Vô hiệu hoá nút Tối đa hoá và làm cho hình thức Pop Up. Don't make it automatically center itself. Không làm cho nó tự động trung tâm chính nó. Before saving the form, position it slightly to the middle-right side of the screen so Microsoft Access would remember that position Trước khi lưu mẫu, vị trí của nó hơi vào phía bên phải giữa màn hình để Microsoft Access sẽ nhớ rằng vị trí

Save and close the form Lưu và đóng biểu mẫu

5. Open the LoanProcessing form in Design View. Mở mẫu LoanProcessing trong thiết

kế View. Using the Command Button Wizard to add a button that, when clicked, would open the LoanEvaluation form (remember that there is no relationship between both forms; therefore, you will Open The Form And Show All The Records). Sử dụng Command Button Wizard để thêm một nút đó, khi nhấn vào, sẽ mở ra hình thức LoanEvaluation (hãy nhớ rằng không có mối quan hệ giữa cả hai hình thức, vì vậy, bạn sẽ mở Mẫu Và Show All The Records). Set the button's Text to Loan Evaluation and its Name to cmdLoanEval. Thiết lập của nút Text để Loan đánh giá và Tên của nó để cmdLoanEval.

Save and close the form Lưu và đóng biểu mẫu

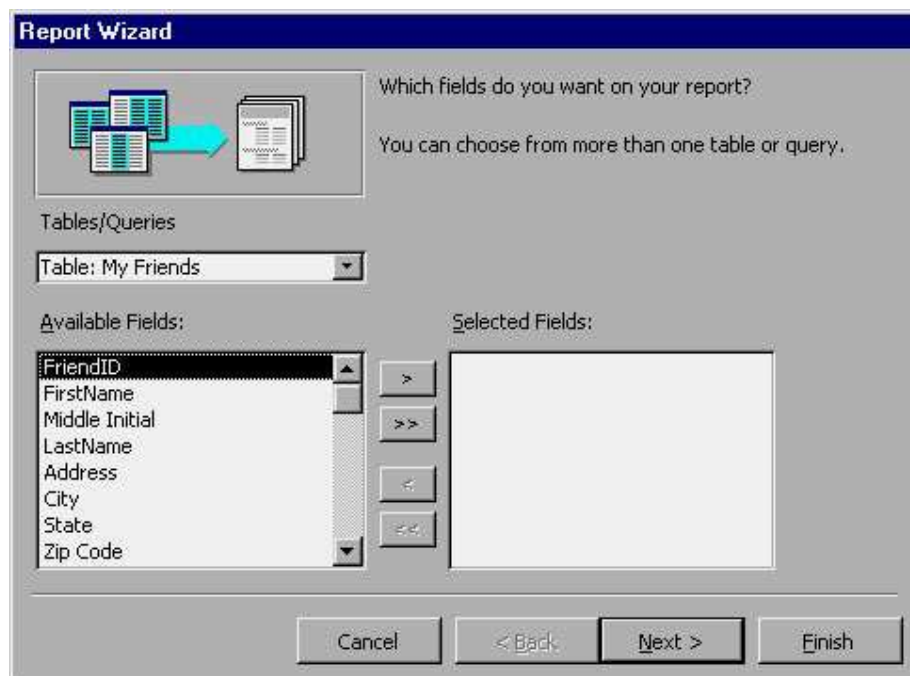
6. Open the LoanProcessing form and use its Loan Evaluation button to open the LoanEvaluation form. Mở và sử dụng hình thức LoanProcessing nút của nó đánh giá cho vay để mở mẫu LoanEvaluation. In the Loan Evaluation form, evaluate a \$1500.00 amount of at 12.50% paid in 28 months. Trong hình thức đánh giá Loan, đánh giá một số tiền \$ 1.500,00 12,50% của lúc thanh toán 28 tháng. After evaluating it, manually create a new personal loan in the Loan Processing form for the amount of \$1500 at 12.50% for 28 payments. Sau khi đánh giá nó, tự tạo ra một cá nhân vay mới trong chế biến hình thức vay với số tiền là \$ 1.500 ở 12,50% cho 28 thanh toán. The loan is processed by the owner, for the 83-457-8 account on April 10th, 2002. Khoản vay được xử lý bởi chủ sở hữu, cho tài khoản 83-457-8 ngày 10 tháng tư 2002. Make the 1st payment due on May 20th of the same year and put a reminder that the payments are due every 22th of the month Thực hiện thanh toán 1 do ngày tháng năm 20 cùng năm và đưa một lời nhắc nhở rằng các khoản thanh toán là do mỗi 22th của tháng

Evaluate other amounts and create loans for the other customers.

Close both forms

7. Open the CustomersTransactions form to see the results

Để bắt đầu với Report Wizard, phải chắc chắn là tab Reports đã được chọn trong cửa sổ Database. Sau đó, kích đúp chuột vào **Create Report By Using Wizard**. Access hiển thị hộp thoại Report Wizard như hình 1.



## Hình 1: Hộp thoại Report Wizard

Phía bên trái hộp thoại Report Wizard, bạn có thể sử dụng danh sách xổ xuống để xác định những nguồn nào mà Access nên dùng cho báo cáo mới này. Hãy nhớ là bạn có thể sử dụng cả dữ liệu nguồn là các bảng và truy vấn. Vì cơ sở dữ liệu *My Friends*

### Chọn lựa thông tin

Tại vị trí phía dưới bên trái hộp thoại là một danh sách của tất cả các trường đã được xác định trong việc chọn bảng hay truy vấn. Nếu bạn thay đổi nguồn dữ liệu từ danh sách, thì các trường hiện hữu cũng sẽ thay đổi. Bạn có thể sử dụng danh sách và các điều khiển ở phía bên phải danh sách, để biết trường nào bạn muốn có trong báo cáo đang tạo. Từ một báo cáo đơn giản này, bạn sử dụng các trường sau đây: chỉ có một bảng (bảng *My Friends*) và Access đã chọn nó cho bạn. Nếu bạn đang làm việc với một cơ sở dữ liệu đã chứa các bảng và truy vấn khác, bạn có thể sử dụng danh sách xổ xuống để thay đổi nguồn dữ liệu

- Last Name
- First Name
- Address
- City
- State
- Zip Code
- Home Phone
- Work Phone

Để cho Access biết bạn muốn sử dụng một trường trong báo cáo, kích chuột vào tên trường (theo thứ tự hiển thị phía dưới đây) và kích chuột vào nút >. Access sẽ chuyển trường từ cột phía bên trái sang cột phía bên phải. Lặp lại quá trình này cho tới khi bạn lựa chọn và chuyển tất cả các trường đã liệt kê ở trên. Sau khi đã hoàn thành, kích chuột vào nút **Next**. Access hiển thị hộp thoại như hình 2.



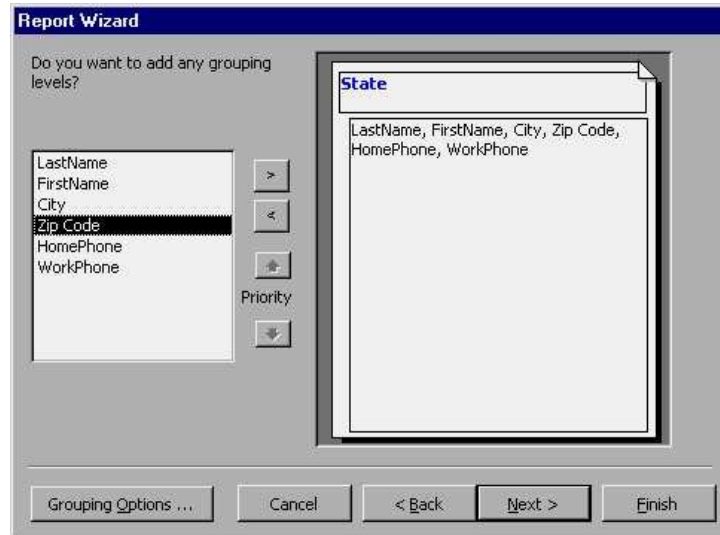
Hình 2: Chọn các trường để gộp nhóm.

### Nhóm các thông tin



Bước tiếp theo là quyết định bạn muốn thông tin được nhóm lại theo thứ tự cụ thể nào. Gộp nhóm thường được thực hiện trong các báo cáo về tài chính, mặc dù nó không có nghĩa là giới hạn các báo cáo tài chính. Ví dụ, nếu bạn tạo một báo cáo hiển thị việc bán hàng cho một công ty, bạn muốn thông tin trong báo cáo được nhóm theo tháng, theo người bán hàng hoặc theo khu vực.

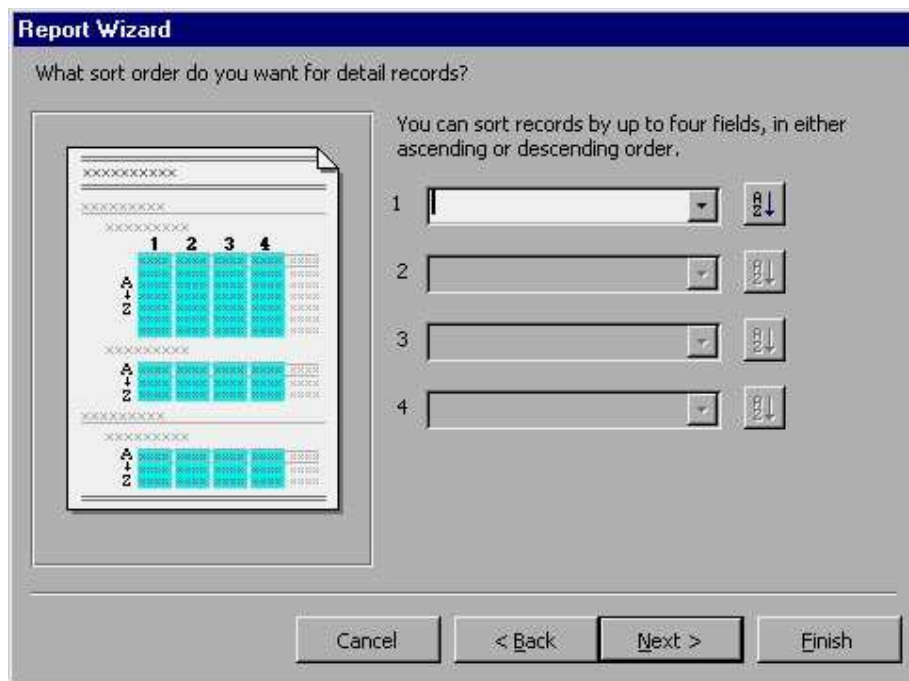
Tạo một báo cáo đơn giản dựa vào bảng My Friends, việc gộp nhóm có thể là không cần thiết. Để xem cách nó làm việc như thế nào, hãy nói rằng bạn có một số lượng lớn bạn bè trong cơ sở dữ liệu và muốn được nhóm lại theo trường *State*. Để thực hiện, chỉ cần chọn trường *State* ở phía bên trái hộp thoại và kích vào nút >. Hộp thoại Report Wizard thay đổi để hiển thị cách gộp nhóm như hình 3.



Hình 3: Gộp nhóm sẽ giúp bạn tổ chức đầu ra trong một báo cáo.

Chú ý rằng trường *State* được chuyển lên trên cùng của mẫu báo cáo và nút **Group Options** được hoạt động. Với một số loại báo cáo, bạn có thể thay đổi các tùy chọn nhóm. Làm như vậy sẽ cho phép bạn xác định cách Access nên phân tích trường mà bạn đang gộp nhóm. Trong trường hợp này, thay đổi tùy chọn gộp nhóm là không cần thiết.

Để tiến hành bước tiếp theo, kích chuột vào nút **Next**. Access hiển thị hộp thoại như hình 4.



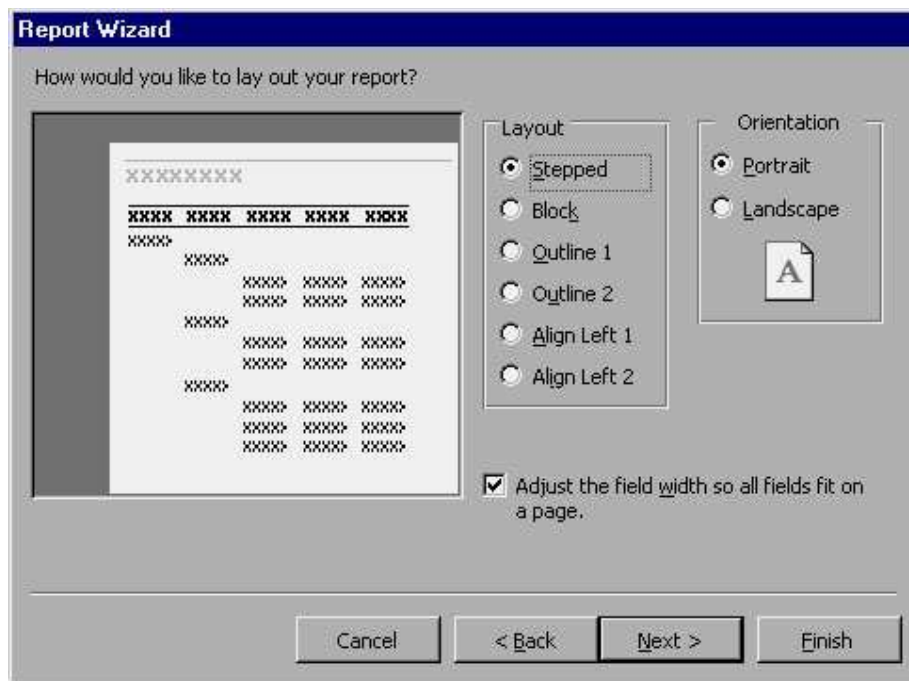
Hình 4: Chọn các trường để sắp xếp theo.

### Sắp xếp thông tin cho báo cáo

Bây giờ, bạn cần quyết định thông tin trong báo cáo được sắp xếp như thế nào. Access cho phép bạn chỉ định tối đa 4 trường có thể sắp xếp. Trong phần lớn các trường hợp bạn chỉ sử dụng một hoặc 2 trường để sắp xếp. Nếu bạn tạo một danh sách bạn bè, và muốn tên của họ được sắp xếp theo thứ tự bảng chữ cái. Bạn nên chọn trường *LastName* và *FirstName*.

Sử dụng danh sách xổ xuống thứ nhất với việc sắp xếp để chọn trường *LastName*. Nếu muốn, bạn có thể kích chuột vào nút sắp xếp phía bên phải trường để cho biết Access nên sắp xếp trường theo thứ tự tăng dần hoặc giảm dần. Và bây giờ, bạn sử dụng danh sách xổ xuống thứ hai để chọn trường *FirstName*.

Khi đã hoàn thành việc chọn các trường để sắp xếp, bạn có thể kích chuột vào nút **Next** để tiến hành bước tiếp theo. Access hiển thị hộp thoại như hình 5.



Hình 5: Xác định layout cho báo cáo.

### Xác định Report Layout

Sử dụng hộp thoại hình 5, bạn có thể biểu thị các tùy chọn khác nhau cho việc bố trí bản báo cáo. Các nút tùy chọn trong hộp thoại điều chỉnh các bố trí khác nhau. Cách tốt nhất để tìm hiểu cách mà bạn muốn dữ liệu xuất hiện là thử các nút khác nhau. Access sẽ thay đổi mẫu báo cáo để biểu thị thông tin.

Chú ý rằng, các điều chỉnh trong hộp thoại không thực hiện với nội dung, mà chỉ áp dụng với cách các báo cáo trình bày nội dung mà bạn đã chọn. Bên cạnh việc chọn một kiểu bố trí, hộp thoại cũng cho phép bạn chọn một sự định hướng cho báo cáo (đó là quay ngang hoặc quay dọc báo cáo). Theo các trường bao gồm trong báo cáo này, bạn chắc chắn chọn kiểu xoay ngang.

Bạn cũng có thể sử dụng hộp thoại này để chỉ ra Access nên giới hạn kích thước của mỗi trường ở đâu ra. Nếu hộp chọn **Adjust**

Khi hoàn thành, kích chuột vào nút **Next** để thực hiện hộp thoại tiếp theo (hình 6). (phía dưới hộp thoại) được chọn, Access có thể cắt bỏ một số dữ liệu. Nếu bạn bỏ qua hộp chọn này, dữ liệu không bị cắt bỏ nhưng nó có thể không nằm hết trên một dòng. Vì mục đích của báo cáo này, chấp nhận sự mặc định là cho phép điều chỉnh độ rộng cho cột. (Nhớ là bạn luôn có thể thay đổi report layout sau đó).



Hình 6: Chọn kiểu báo cáo.

Tại đây, bạn có thêm cơ hội để xác định cách bạn muốn nội dung sẽ xuất hiện như thế nào. Access cung cấp 6 kiểu đã được định nghĩa trước mà xác định các yếu tố báo cáo như font chữ, màu sắc, dòng, ... Bạn có thể chọn lần lượt mỗi kiểu và xem báo cáo của bạn sẽ xuất hiện như thế nào trong hộp thoại. Chọn một kiểu và kích vào nút **Next**. Access hiển thị hộp thoại cuối cùng như hình 7.



Hình 7: Kết thúc tạo báo cáo

Bạn đã gần hoàn thành xong việc định nghĩa báo cáo. Trong hộp thoại cuối, Access hỏi bạn về tiêu đề sử dụng cho báo cáo. Tiêu đề này được dùng trong hai vị trí: phía trên cùng của mỗi trang trong bản báo cáo được in, và khi lưu báo cáo thực trong cơ sở dữ liệu. Chú ý rằng, Wizard cung cấp cùng một tên như bảng đã được sử dụng. Trong trường hợp này thì đó là một tiêu đề phù hợp, do vậy bạn có thể dùng lại nó.

Lúc này bạn đã sẵn sàng để kích chuột vào nút **Finish**. Khi đó, Access sử dụng thuộc tính xem trước khi in để hiển thị cho bạn bản báo cáo sẽ được in ra như thế nào. Đây là cách có thể xem nếu bạn cần thực hiện bất kỳ thay đổi nào. Hình 8 hiển thị báo cáo cuối cùng.

The screenshot shows a window titled 'My Friends' containing a report. The report has a title 'My Friends' and a table with the following data:

State	Last Name	First Name	City	Zip Code	Home Phone	Work
AZ	Lopez	Dan	Holbrook	86029	(602) 223-345	(602)
	Nielson	Edward	Joseph City	86032	(602) 313-006	
	Thompson	Trevor	Cameron	86020	(602) 336-977	(602)
CA	Humphrey	Martha	Los Angeles	90014	(213) 707-209	(213)
CO	Jacobson	Simon	Louisville	80027-9556	(303) 963-846	(303)
	Mathis	Harold	Denver	80221	(303) 852-134	(303)
IL	Brough	Heather	Baylis	62314	(217) 306-294	
	Parker	James	Easton	62633	(618) 556-030	
IN	Johnson	Clyde	Indianapolis	46250	(317) 345-091	
KS	Livingston	Monet	Paola	66071	(913) 312-844	

Hình 8: Báo cáo cuối cùng

Report Wizard đã tạo một báo cáo rất hay. Có vài thay đổi bạn có thể thực hiện (như thay đổi lề, độ rộng các trường) nhưng thay đổi toàn bộ báo cáo thì không nên. Ở chương 14, bạn sẽ học cách sử dụng cửa sổ thiết kế báo cáo và vì vậy bạn có thể tạo những báo cáo riêng từ sự hỗn tạp và thực hiện các thay đổi cho báo cáo.

Khi đã xem xong báo cáo, đóng cửa sổ xem trước khi in bằng cách kích chuột vào công cụ Close trên thanh công cụ hoặc kích vào biểu tượng **Close** (X) ở phía góc trên bên phải cửa sổ. Access chỉ hiển thị cửa sổ Design khi đã hoàn thành. Bạn có thể đóng cửa sổ thiết kế bằng cách kích chuột vào biểu tượng **Close** ở phía góc trên bên phải cửa sổ Design. Lúc này bạn đã hoàn thành bản báo cáo đầu tiên trong Access.

### **NgânNT (LearnAccess)**

**Report Wizard** bạn đã sử dụng ở đâu chương này không phải là phần duy nhất được cung cấp bởi Access, mà còn các phần khác bạn có thể dùng như vậy. Từ thực đơn **Insert**, bạn truy cập Wizard bằng cách chọn **Report**. Sau đó, Access hiển thị hộp thoại **New Report** như hình 1.



Hình 1: Hộp thoại New Report

Chú ý rằng ở phía dưới hộp thoại New Report là nơi xác định nguồn dữ liệu (bảng hoặc truy vấn) sẽ được sử dụng làm cơ sở cho báo cáo. Phía trên hộp thoại là 6 sự chọn lựa đại diện cho các hành động khác nhau mà bạn có thể dùng:

- **Design View.** Cho phép bạn tạo một báo cáo sử dụng cửa sổ thiết kế báo cáo. Phần này sẽ được trình bày chi tiết ở chương 14.
- **Report Wizard.** Tạo một báo cáo sử dụng Report Wizard cơ bản. Đây là cùng một Wizard bạn sử dụng nếu kích đúp chuột vào **Create Report By Using Wizard** trong phần Reports của cửa sổ Database
- **AutoReport: Columnar.** Tạo một báo cáo mà danh sách mỗi trường trong một bản ghi ở trên một dòng ngăn cách. (Loại báo cáo này có thể tiêu tốn một lượng giấy lớn).
- **AutoReport: Tabular.** Tạo một báo cáo tương tự như một bảng tính – một bản ghi trên một dòng, với các trường xuất phát ngang từ trái sang phải trang.
- **Chart Wizard.** Tạo một báo cáo bao gồm một biểu đồ được chuyển hóa từ dữ liệu trong bảng hoặc truy vấn.
- **Label Wizard.** Tạo nhiều loại khác nhau của các labels

Phần lớn những báo cáo trên đây đều sử dụng một Report Wizard khác nhau. Báo cáo của bạn lựa chọn câu trả lời ở đây sẽ quyết định câu hỏi sau đó. Thậm chí kết quả cuối cùng là khác nhau, câu hỏi được hỏi ở phần đưa ra là tương tự nhau. Tất cả bạn cần làm là áp dụng những kỹ năng có được ở chương này và mọi việc sẽ trở nên thuận lợi

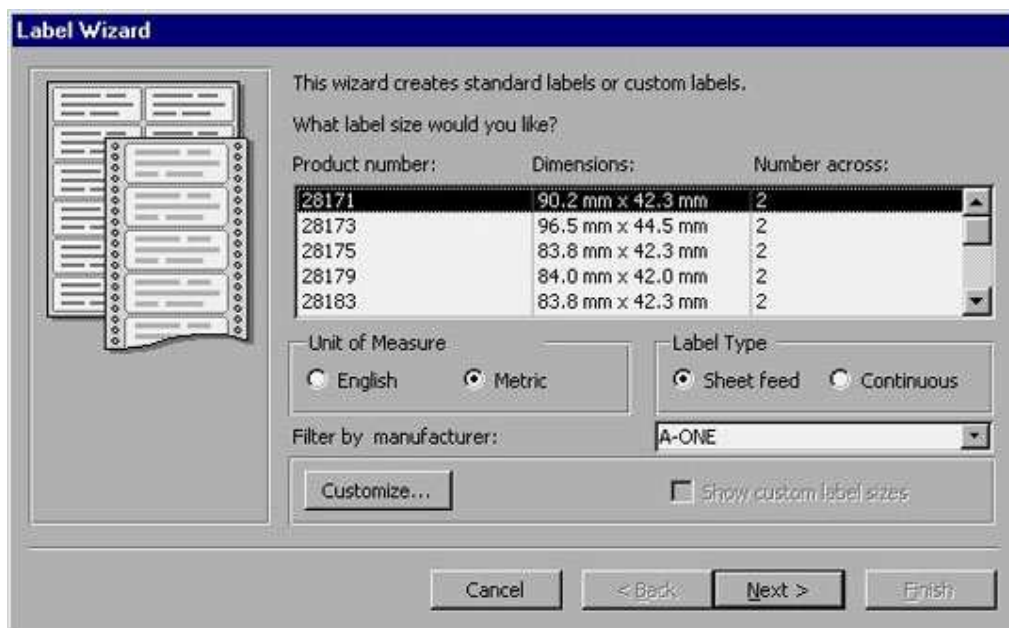
Một trường hợp ngoại lệ là khi bạn muốn tạo các nhãn. Trong phần tiếp theo, bạn học cách sử dụng Label Wizard để tạo các nhãn dựa vào thông tin trong cơ sở dữ liệu.

**Trong phần trước, bạn đã tìm hiểu về Label Wizard mà bạn có thể sử dụng để tạo các nhãn thư tín. Mailing labels cần thiết trong mọi doanh nghiệp.** Giả sử bạn có cơ sở dữ liệu *Customer.mdb*, bạn có thể bắt đầu quy trình tạo nhãn thư tín bằng cách chọn Report từ thực đơn **Insert** hoặc kích nút **Report** ở ngay cửa sổ Database. Hộp thoại New Report xuất hiện như hình 1



Hình 1: Hộp thoại New Report

Sử dụng danh sách xổ xuống phía dưới hộp thoại để chọn bảng *Business Customers* làm nguồn dữ liệu. Khi hoàn thành việc này, chọn **Label Wizard** và kích chuột vào nút **OK**. Access hiển thị bước đầu tiên trong Label Wizard như hình 2



Hình 2: Chọn một loại nhãn

Bước đầu tiên trong việc tạo các label là xác định loại label bạn muốn dùng. Thật may là Access rất dễ dàng để xác định một loại label. Trong khi có rất nhiều các label khác nhau hiện hữu trên thị trường, kỹ nghệ tiêu chuẩn cho các loại label khác nhau. Label Wizard hỗ trợ trên 650 loại label từ 13 hãng sản xuất khác nhau. Rất may là bạn có thể tìm được label cụ thể đã được xác định trước đó bởi Access

Để chọn một loại label, sử dụng 3 điều khiển ở giữa hộp thoại để lọc những label nào được hiển thị. 3 điều khiển đó như sau:

- **Unit of Measure.** Bạn có thể chọn English hoặc Metric. Sự chọn lựa của bạn sẽ quyết định những label nào được hiển thị trong danh sách
- **Label Type.** Bạn có thể chọn Sheet Feed hoặc Continuous. ( Sheet-feed được sử dụng cùng

với máy in khi mà yêu cầu bạn điều khiển mỗi trang tính của label. Continuous được sử dụng khi máy in điều khiển cung cấp chất liệu cho máy). Sự lựa chọn của bạn quyết định những label nào được hiển thị trong danh sách.

- **Filter by Manufacturer.** Sử dụng danh sách xổ xuống này để cho biết ai tạo ra label của bạn. Chỉ các label được sản xuất bởi hãng sản xuất sẽ được liệt kê trong danh sách

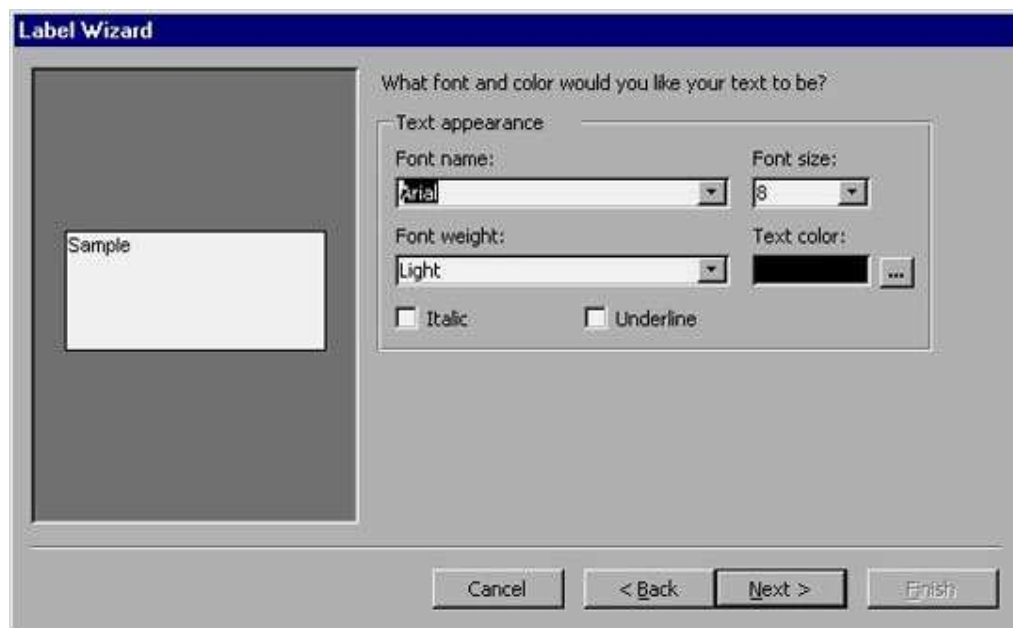
Nếu bạn đang ở US, và không thấy hãng sản xuất được liệt kê, bạn chọn Avery. Phần lớn các hãng sản xuất label có tiêu chuẩn giống như kích thước được hỗ trợ bởi Avery. Nếu không thể tìm thấy nhãn Avery, bạn có thể chuyển tới cột Dimensions trên hộp thoại để tìm hiểu label bạn muốn sử dụng.

**Chú ý:** nếu không thể tìm thấy label bạn muốn hoặc một label tương ứng, bạn có thể kích chuột vào nút Customize để cho biết loại label cụ thể bạn đang sử dụng.

Trong ví dụ này, bạn đang tạo label dạng chuẩn (1" x 4"). Đây là label Avery 5216. Những label này cho phép bạn in 6 dòng trên label và được thiết kế cho máy in laser, đây là một label chuẩn được sản xuất bởi nhiều đại lý khác. Minh họa cho ví dụ này, chọn ở mục Unit of Measure là English, Label type là Sheet Feed và hãng sản xuất là Avery. Sau đó chọn label 5216.

### Chọn Fonts chữ và màu sắc

Kích chuột vào nút Next để cho biết bạn đã hoàn thành việc chọn loại label. Lúc này bạn đã sẵn sàng chọn font chữ và màu sắc. Access hiển thị hộp thoại như hình 3.



Hình 3: Chọn font chữ và màu sắc.

Font và kích thước mà Label Wizard đưa ra là **Font** chữ Arial và kích thước font chữ là 8, sử dụng font weight là light. Đây là một điểm có kích thước nhỏ hơn và bạn nên tăng nó thêm một chút nếu có thể. Dù có thay đổi kích thước font chữ hay không thì đều phụ thuộc vào loại máy in của bạn và các tính năng của nó. Minh họa cho ví dụ này, chúng ta không thay đổi kích thước font chữ, nhưng có thể thay đổi bất kỳ đặc tính khác mà bạn muốn. Ví dụ, bạn có thể thay đổi Font Weight thành **Normal** hoặc **Bold**. Trừ khi bạn có một máy in màu, nếu không thì không có ý nghĩa khi thay đổi **Text Color**.



## Chọn nội dung Label

Sau khi đã xác định font chữ sử dụng, kích chuột vào nút **Next**. Access hiển thị hộp thoại như hình 4.



Hình 4: Trình bày qua về nội dung label

Label Wizard cho phép bạn xác định cách bạn muốn label xuất hiện như thế nào. Phía bên trái hộp thoại, bạn có thể chọn các trường khác nhau từ bảng. Bạn cũng có thể nhập nội dung mà sẽ xuất hiện trên mọi label. Bạn muốn làm điều này thì phải chèn dấu cách, dấu phẩy, dấu gạch chân.

Sẽ chỉ có 6 dòng để làm việc, bạn cần phải dự tính cách muốn label xuất hiện. Label Wizard sẽ sử dụng hai dòng cho dấu cách ở phía trên và phía dưới mỗi label, do đó bạn chỉ còn 4 dòng để làm việc. Bạn biết rằng, tên công ty và địa chỉ sẽ mất 3 dòng. Điều này có nghĩa bạn chỉ còn một dòng. Nó phải có ích nếu bạn được sử dụng dòng trên cùng của label để hiển thị đề mục.

Để tạo một label theo dạng thức này, thực hiện các bước sau:

1. Trong hộp **Prototype**, nhập nội dung *Your account number is*. (Đừng quên thêm một dấu cách vào cuối nội dung. Dấu cách này sẽ ngăn cách nội dung từ trường mục hiện tại)
2. Chọn trường **Account Number** trong danh sách trường, sau đó kích chuột vào nút **>**. Thao tác này sẽ di chuyển số trường mục tới cuối dòng đầu tiên trong label. Bạn đã hoàn thành dòng đầu tiên.
3. Ấn phím ENTER để hoàn thành dòng đầu tiên và di chuyển tới dòng thứ hai của label.
4. Kích chuột vào trường **Company Name** trong danh sách trường, sau đó kích chuột vào nút **>**.
5. Ấn phím ENTER.
6. Kích chuột vào trường **Address 1** trong danh sách trường, sau đó kích vào nút **>**

7. Ấn phím ENTER.

8. Kích chuột vào trường **City** trong danh sách và kích vào nút >.

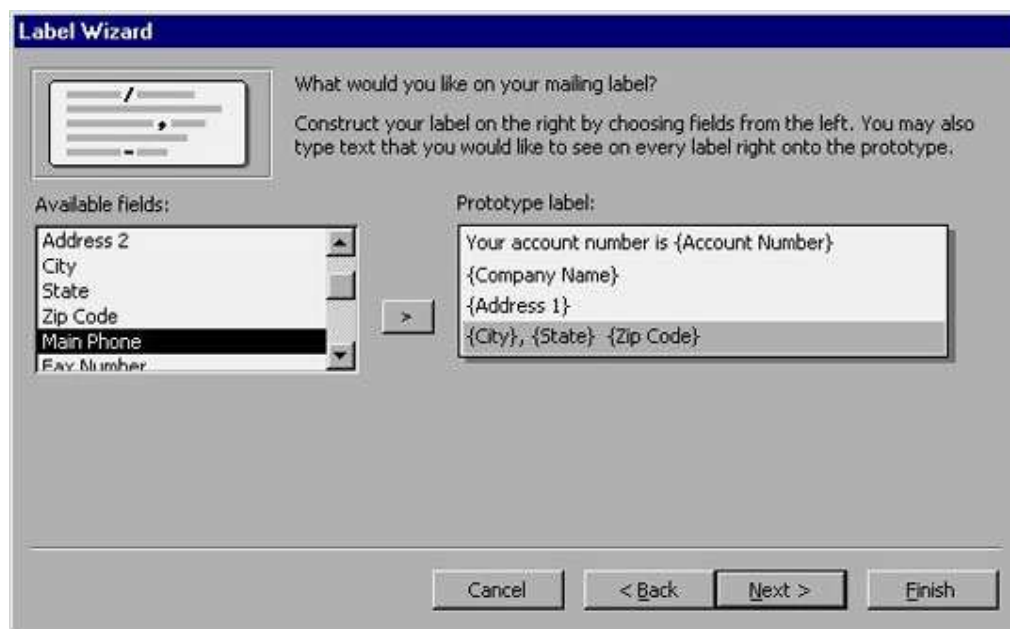
9. Nhập một dấu phẩy

10. Kích chuột vào trường **State** trong danh sách, và kích vào nút >.

11. Nhập dấu cách thứ hai

12. Kích chuột vào trường **Zip Code** trong danh sách và chọn nút >.

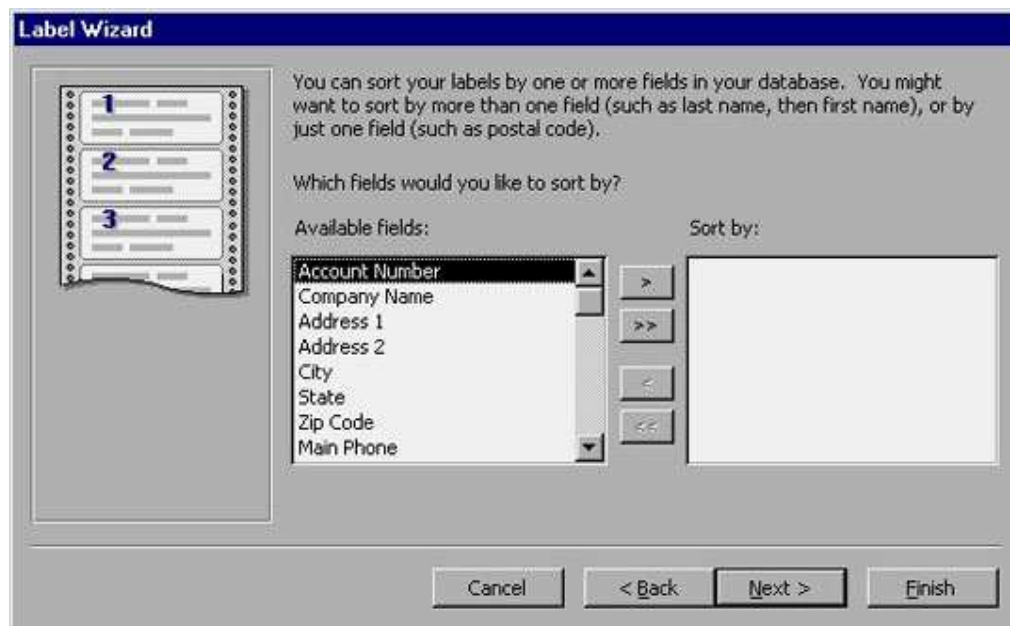
Bạn đã hoàn thành việc thiết kế label, và nó sẽ xuất hiện như hình 5. Hãy dành một chút xem lại label để chắc chắn nó sẽ hiển thị đúng. Nếu không đúng, bạn có thể chọn dòng bạn muốn để thay đổi và ấn phím **DEL** để bỏ dòng. Sau đó có thể thiết lập lại một cách đúng đắn.



Hình 5: Kết thúc label layout.

## Sắp xếp Label

Khi đã vừa ý với label layout, kích chuột vào nút **Next**. Access hiển thị hộp thoại như hình 6.



Hình 6: Chọn sắp xếp theo thứ tự.

Thường thì bạn muốn sắp xếp label theo **Zip code**, nhưng có thể sử dụng bất kỳ trường nào mà bạn muốn. Trong ví dụ này, chúng ta sẽ sắp xếp với hai mức độ: đầu tiên là theo Zip code và sau đó là theo **company name**. Để làm điều này, chọn mỗi trường trong những trường này theo thứ tự, sau đó kích nút >. Cả hai trường sẽ xuất hiện phía bên phải hộp thoại.

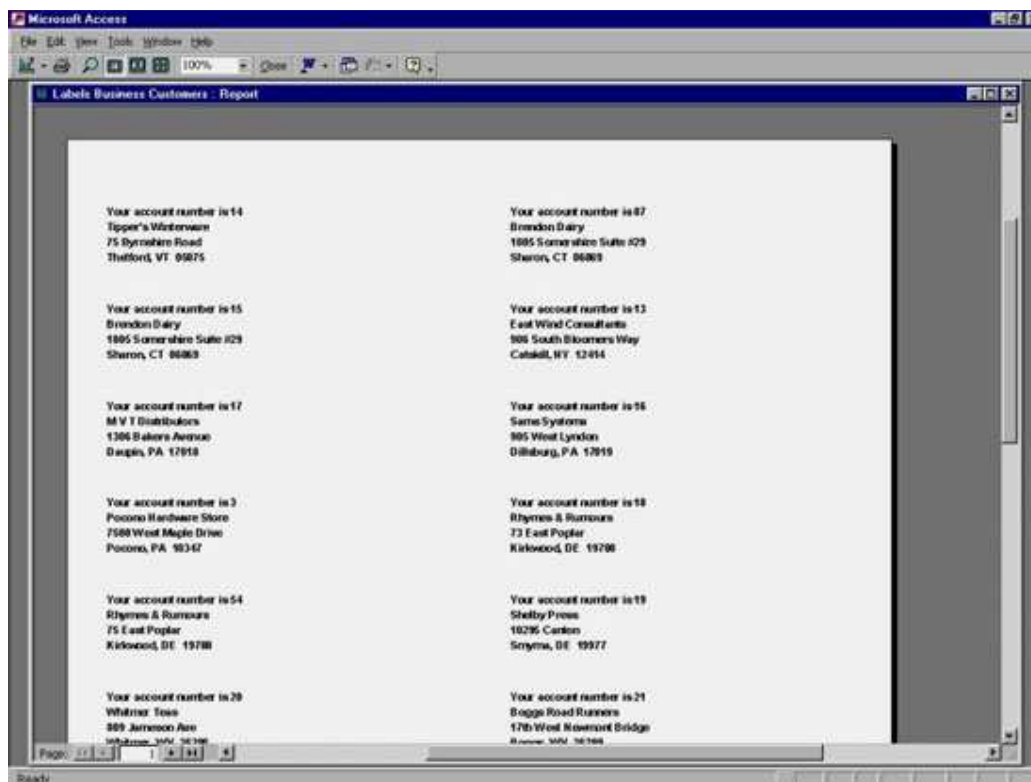
### Kết thúc Label

Bạn đã thực sự hoàn thành label của mình. Kích chuột vào nút **Next**. Access hiển thị hộp thoại như hình 7.



Hình 7: Kết thúc label

Không có gì là bạn cần để thay đổi trong hộp thoại này. Kích chuột vào nút **Finish** và Access hiển thị một chế độ xem trước khi in. Hình 8 hiển thị những gì chúng ta thấy.



Hình 8: Chế độ xem trước khi in cho label

Khi bạn đã hoàn thành xem báo cáo, đóng cửa sổ xem trước khi in.

**Access có một số khả năng báo cáo rất ấn tượng. Thực tế, sự tương tượng của bạn chỉ giới hạn những gì có thể làm với các report. Trong chương này, bạn đã tìm hiểu làm thế nào để tạo hai báo cáo đơn giản với Report Wizards đã xây dựng trong Access.**

Mặc dù những báo cáo này đơn giản nhưng chúng rất hữu ích trong cuộc sống hàng ngày. Quy trình phát triển những báo cáo này đã cho bạn những kỹ năng sử dụng để phát triển những báo cáo khác.

Trước khi chuyển sang chương 9, phải chắc chắn bạn nắm được những nội dung sau:

- Trong Access, một báo cáo nhiều hơn một dữ liệu in ra từ máy tính. Report cung cấp cho bạn hoàn toàn quyền điều khiển trên những gì Access in và cách nó xuất hiện trên trang.
- Bạn có thể tạo các báo cáo sử dụng Report Wizard hoặc cửa sổ thiết kế báo cáo.
- Bạn có thể bắt đầu một Report Wizard từ phần Reports của cửa sổ Database hoặc chọn Reports từ thực đơn Insert.
- Report Wizard có thể được sử dụng để tạo nhiều loại mẫu báo cáo. Những báo cáo này thường đơn giản và gần như là không phải tùy chỉnh như những gì bạn có thể tạo bằng cách sử dụng cửa sổ thiết kế báo cáo.
- Các báo cáo đơn giản và các nhãn thư tín đều rất dễ tạo với Report Wizards.
- Label Wizard hỗ trợ nhiều loại nhãn chuẩn từ các nhà sản xuất, tự động định dạng thông tin để tạo thành nhãn

Nếu bạn muốn biết thêm về việc tạo các báo cáo, tham khảo tới chương 14. Trong chương 9, bạn sẽ bắt đầu bước vào thiết kế các biểu mẫu.

# Bài 35: Tạo Form sử dụng Form Wizards

**Chú ý:** Phần này sử dụng cơ sở dữ liệu Customer như đã nêu trong tuyển tập bài viết này

Với các đối tượng cơ sở dữ liệu khác, Access cung cấp Form Wizards để bạn có thể sử dụng cho việc tạo các biểu mẫu. Để thấy được toàn bộ vùng Form Wizards dùng sẵn, chọn **Form** từ thực đơn **Insert**. Access hiển thị hộp thoại New Form như hình 1.



Hình 1: Hộp thoại New Form

Hộp thoại này trông cũng như hầu hết các hộp thoại mà bạn đã sử dụng để tạo các truy vấn và báo cáo. Thao tác đầu tiên là sử dụng danh sách xổ xuống ở phía dưới hộp thoại để chọn một bảng hay truy vấn dữ liệu mà bạn muốn sử dụng để tạo Form. Bạn nên chọn bảng Business Customers.

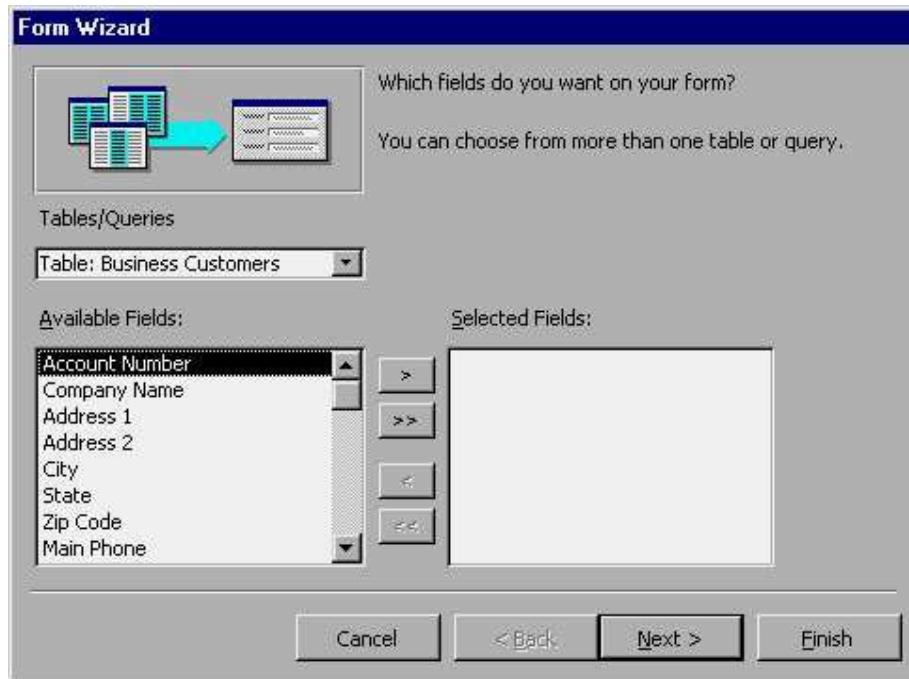
Phía trên của hộp thoại là 7 lựa chọn để xác định các tài nguyên khác nhau mà bạn có thể sử dụng để tạo một Form mới. Các chọn lựa đó là:

- **Design View.** Cho phép bạn tạo một báo cáo sử dụng cửa sổ thiết kế báo cáo. Lựa chọn này sẽ được đề cập chi tiết hơn ở ngay sau chương này.
- **Form Wizard.** Tạo một Form sử dụng Form Wizard cơ bản. Lựa chọn này cũng giống Wizard khi bạn kích đúp chuột vào Create Form By Using Wizard trong vùng Forms của cửa sổ Database.
- **AutoForm: Columnar.** Form sẽ hiển thị tất cả các trường trong bảng theo một cột đơn, với thuộc tính Caption của trường nằm ở phía bên trái
- **AutoForm: Tabular.** Form sẽ hiển thị tất cả các trường trong bảng theo hàng ngang với mỗi bản ghi là một dòng đơn.
- **AutoForm: Datasheet.** Form trông giống khi ở chế độ Datasheet của bảng hay truy vấn.
- **Chart Wizard.** Form sẽ hiển thị một biểu đồ dựa vào dữ liệu bảng.
- **PivotTable Wizard.** Access tạo một Form mà bao gồm một PivoTable. Lựa chọn này có thể giống như trong Excel; cho phép các thông tin được xem dưới một biểu mẫu cô đọng mà trợ giúp cho một số loại phân tích dữ liệu

Sự khác nhau cơ bản giữa những biểu mẫu này là cách bố trí thông tin trên form. Không có sự khác biệt bên trong những gì Form thực hiện hoặc cách tạo nó. Trong những phần sau, bạn sẽ học cách sử dụng Form Wizards để tạo một số loại Forms.

## Tạo một Form cơ bản

Để tạo một Form cơ bản, bạn chọn **Form Wizard** từ hộp thoại New Form và kích chuột vào nút **OK**. (Bạn cũng có thể kích đúp chuột vào tùy chọn **Create Form By Using Wizard** từ mục Forms của cửa sổ Database). Access hiển thị hộp thoại Form Wizard như hình 2.

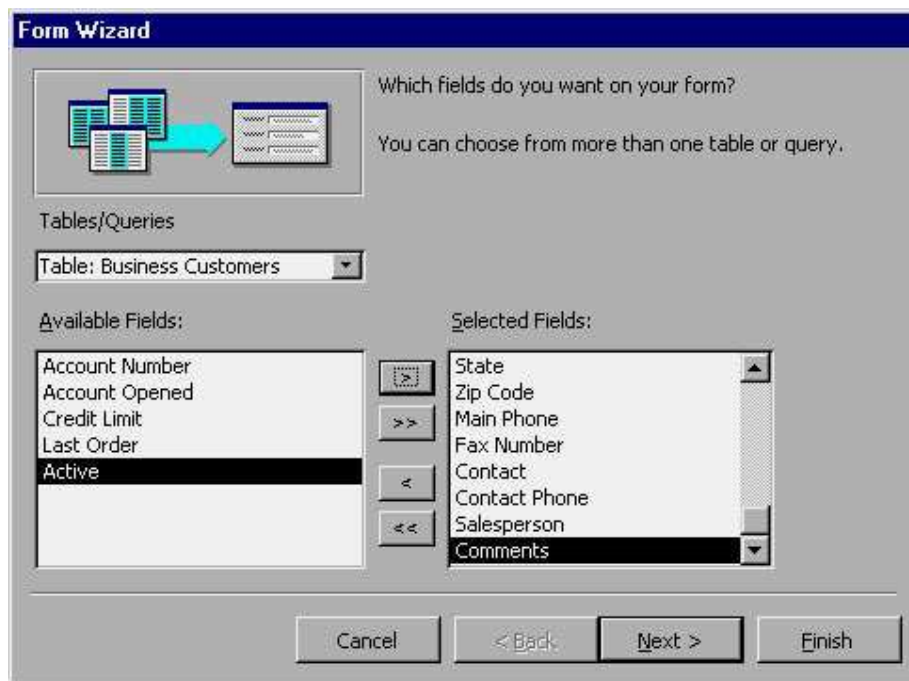


Hình 2: Hộp thoại Form Wizards

Lưu ý rằng hộp thoại bao gồm bảng Business Customers là nguồn dữ liệu cho Form của bạn. (Nhớ là, bạn nên lựa chọn phần này từ hộp thoại New Form). Phía dưới phần bên trái của hộp thoại, bạn có thể thấy các trường khác nhau của bảng Business Customers. Tất cả những gì cần làm là chọn những trường bạn muốn bố trí trên form và sử dụng những điều khiển ngay phía bên phải danh sách để chuyển các trường đó sang vùng Selected Fields.

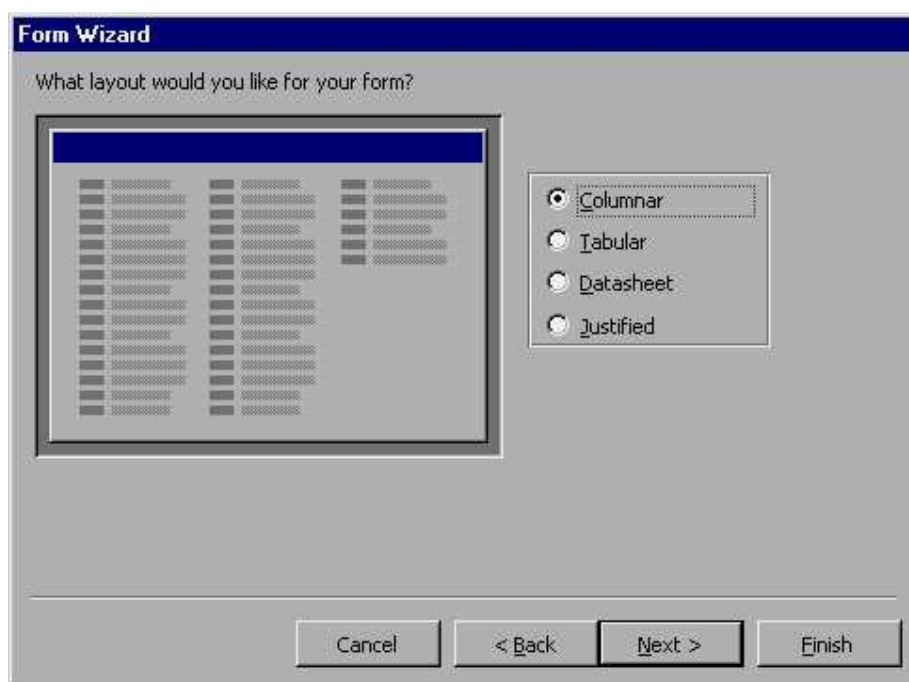
Ví dụ: bạn muốn tạo một Form nhập dữ liệu mà có thể được sử dụng khi thiết lập một tài khoản mới. Chỉ có một vài trường cần thiết cho Form này. Lần lượt, bạn chọn các trường ngoại trừ trường Account Number (trường này đã được tự động chọn), Account Opened, Credit Limit, Last Order và Active (trường này nên được thiết lập tự động).

Chọn được các trường xong, kích chuột vào nút >. Tên các trường xuất hiện ở phía bên phải hộp thoại. Khi đã hoàn thành, hộp thoại xuất hiện như hình 3.



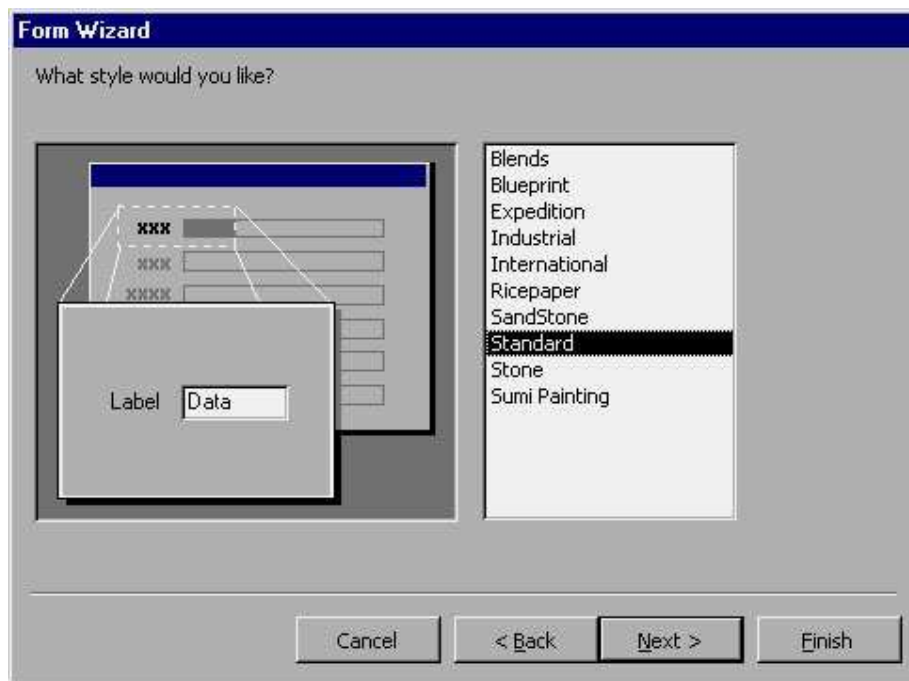
Hình 3: Hộp thoại Form Wizard sau khi chọn các trường.

Khi đã sẵn sàng tiến hành, kích chuột vào nút **Next**. Access hiển thị hộp thoại như hình 4.



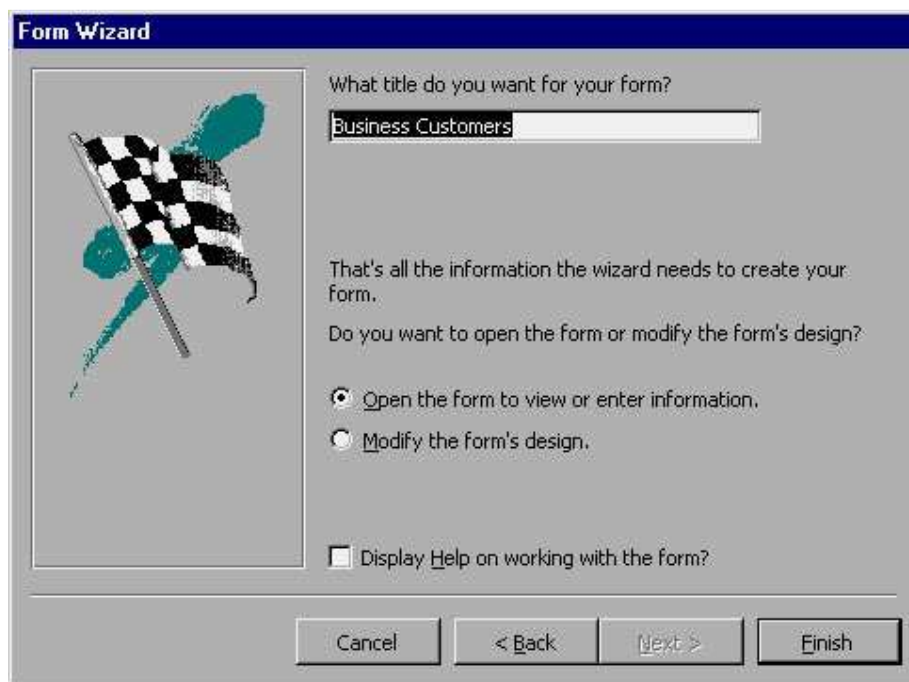
Hình 4: Hộp thoại Form Wizard để chọn cách bố trí.

Ở hộp thoại này đưa ra 4 cách sắp đặt mà bạn có thể chọn: *Columnar* (dạng cột), *Tabular* (dạng bảng), *Datasheet*. Về cơ bản, các cách sắp đặt này cũng giống như những gì bạn tạo với các tùy chọn thứ 3, 4, 5 trong hộp thoại đầu tiên. Khi bạn chọn nút tùy chọn được kết hợp với mỗi loại sắp đặt, chú ý rằng chế độ xem trước Form thay đổi. Điều này cho bạn một ý kiến hay về cách bạn muốn Form xuất hiện như thế nào. Ví dụ: chọn *Columnar*, sau đó kích chuột vào nút **Next**. Access hiển thị hộp thoại Form Wizard tiếp theo như hình 5.



Hình 5. Hộp thoại tiếp theo để chọn kiểu Form.

Hộp thoại này cho bạn 10 dạng Form khác nhau. Khi chọn một kiểu nào đó, bạn có thể thấy kết quả hiện ngay trong hộp thoại. Các kiểu này cung cấp khá nhiều trạng thái khác nhau trong việc trình bày dữ liệu, mà cho thấy nhiều loại dữ liệu có thể được trình bày trong Access. Vì chúng ta đang làm việc với bảng Business Customers nên chọn loại *Standard*. Kích chuột vào nút **Next**, hộp thoại cuối cùng của Form Wizard hiển thị như hình 6.



Hình 6: Kết thúc Form

Đó là những điều cơ bản để tạo Form. Tất cả bạn cần làm lúc này là cung cấp một tiêu đề cho Form. Điều đề này sẽ xuất hiện ở phía trên cùng của form đó và nó sẽ được sử dụng để lưu Form trong cơ sở dữ liệu. Vì tên Form mặc định có thể được chấp nhận, kích chuột vào nút **Finish**. Form Wizard sẽ tạo Form theo các định nghĩa của bạn và hiển thị chúng lên màn hình. Hình 7 hiển thị những gì mà chúng ta đã thực hiện từ đầu bài.



Hình 7: Form hoàn thành

Bạn có thể thấy một Form nhập như thế này sẽ tiện ích hơn Datasheet. Ở phía trên cùng Form, là phần thanh tiêu đề của form mà bạn đã xác định. Phía dưới Form là vùng điều khiển, là nơi bạn có thể định vị từ bản ghi này tới bản ghi kia trong bảng. Bạn cũng có thể xem lại và chỉnh sửa các bản ghi đó, hoặc có thể di chuyển tới bản ghi cuối cùng và thêm bản ghi mới.

**Chú ý:** Form Wizard không phải là luôn thực hiện tốt công việc tạo Forms. Ví dụ: như hình 7, trường hiện số điện thoại quá nhỏ, kết quả hiện số điện thoại bị mất một vài số. Giải thích lý do này, nhiều người sử dụng Wizards để tạo một "điểm khởi đầu" cho việc tạo Forms. Sau đó, họ có thể tùy chỉnh chúng để đáp ứng nhưng nhu cầu cụ thể. Ngay sau chương này, bạn sẽ học cách tạo một biểu mẫu từ đầu. Cũng cùng phương pháp được sử dụng để chỉnh sửa biểu mẫu đã được tạo ra và lưu lại trước đó.

### Tạo một biểu mẫu AutoForm

Nếu bạn xem xét hộp thoại New Form, bạn có thể thấy Access cho phép tạo 3 loại AutoForms: Columnar, Tabular, Datasheet. AutoForms là loại biểu mẫu đơn giản nhất mà bạn có thể tạo. Wizard thực hiện tất cả các quyết định, tất cả bạn cần làm là xác định nguồn dữ liệu là một bảng hay truy vấn được sử dụng trong Form.

Như ví dụ về cách sử dụng một AutoForm – Wizard được hiển thị trong hộp thoại New Form. Hãy chắc chắn rằng Business Customers được chọn là nguồn dữ liệu. Sau đó chọn **AutoForm: Tabular** và kích chuột vào nút **OK**. Lần lượt, Wizard tạo Tabular một biểu mẫu bao gồm tất cả các trường dưới dạng bảng. Bạn không cần phải chọn bất kỳ dạng thức, kiểu mẫu, trường hay các vấn đề khác mà có thể tác động đến cách biểu mẫu xuất hiện. Hình 8 trình bày ví dụ về tạo biểu mẫu AutoForm:

Account Number	Company Name	Address 1	Address 2	City	State	Zip Code	Main Phone
1	Bach Enterprises	3456 Maple Lane	PO Box 549	Kansas City	MO	64147	(816) 237
2	Flood's Book Store	1325 East Main		Pasot	NY	4050	(606) 532
3	Poccon Hardware Store	7500 West Maple Drive		Poccon	PA	18347	(614) 218
4	JP Consolidated	1821 5th Street Suite 27		Lubbock	TX	79402	(915) 614
5	Bridger Law	800 Main Street	PO Box 1281	Alpens	AR	72611	(501) 328
6	Meadowbrook Mall	733 North 2nd Street		Cameron	AZ	00000	(602) 418
7	Pete's Casino & Grill	35 Frontage Road		Crystal	NV	89402	(702) 895
8	Desert Oasis	5th & Main		Mammoth	CA	93456	(619) 821
9	Rain Water Company	1805 Palm Avenue	PO Box 6635	Glancha	CA	93543	(619) 807

Hình 8: Hoàn thành AutoForm

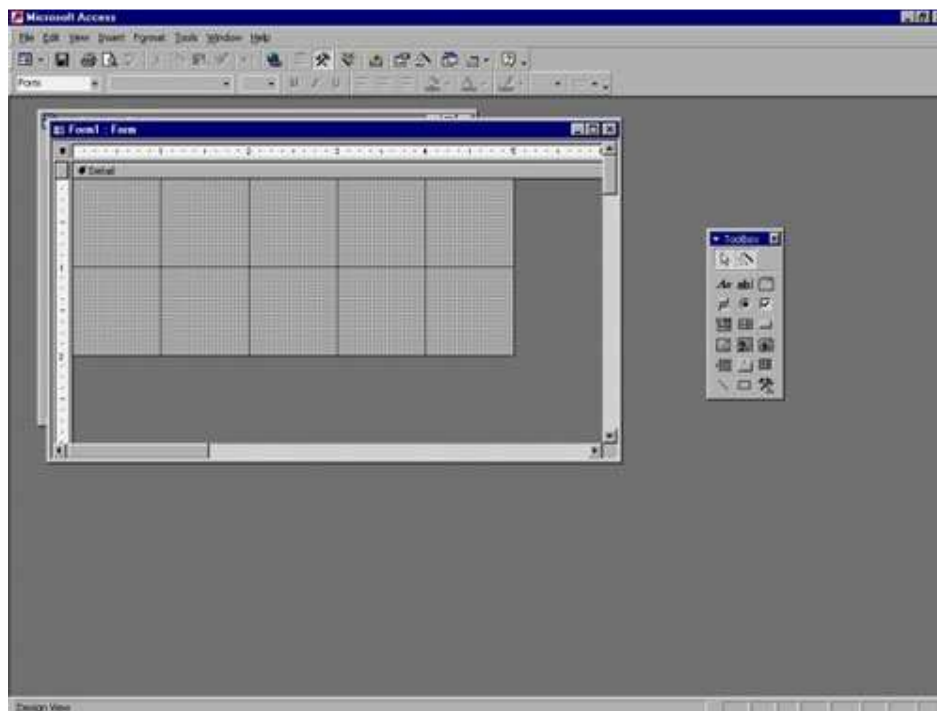
Khi làm việc với biểu mẫu, bạn có thể đóng nó bất kỳ lúc nào, Access sẽ hỏi bạn có muốn lưu lại Form. Nếu kích vào **Yes**, sẽ xuất hiện hộp thoại yêu cầu bạn đặt tên. Ví dụ bạn đặt tên cho Form đó là AutoForm Example, sau khi hoàn thành, kích chuột vào nút **OK**. Access sẽ lưu Form và cập nhật vào cửa sổ Database.

## Bài 36: Định nghĩa các Forms riêng

Khi các biểu mẫu Wizard không đáp ứng được các nhu cầu của bạn, bạn có thể sử dụng cửa sổ Design để tạo một biểu mẫu riêng hay chỉnh sửa biểu mẫu có sẵn. Trong khi cửa sổ thiết kế biểu mẫu cung cấp cho bạn các điều khiển linh động hơn là làm việc với Wizards, nhưng sử dụng cửa sổ thiết kế Form thì phức tạp hơn so với sử dụng bằng Wizards.

**Chú ý:** Phần này sử dụng cơ sở dữ liệu MyFriend.mdb mà bạn đã sử dụng trong các chương trước.

Để tạo một Form riêng, kích chuột vào nút **Forms** trong cửa sổ **Database**. Sau đó, kích đúp chuột vào tùy chọn **Create Form in Design View**. Access hiển thị cửa sổ Design như hình 1.



Hình 1: Cửa sổ thiết kế Form

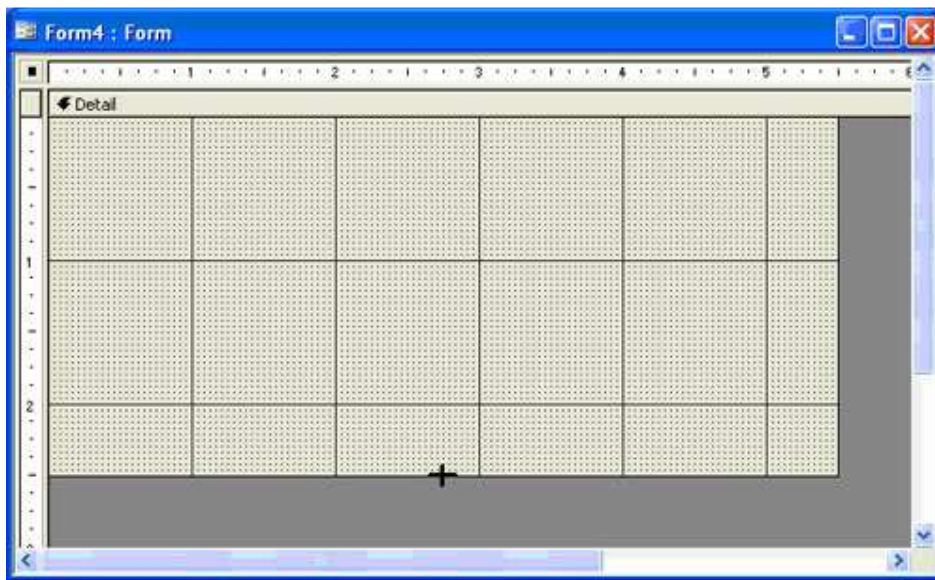
Có một số mục bạn nên chú ý trong cửa sổ này. Đầu tiên, trên thanh công cụ đã có một số công cụ mới. Bạn sẽ tìm hiểu mục đích của mỗi công cụ này ngay sau chương này. Thứ hai, đó là hộp công cụ thiết kế. Bạn sẽ sử dụng hộp công cụ (được miêu tả một cách ngắn gọn) cùng với việc tạo form.

Phần chính của cửa sổ thiết kế là lưới thiết kế. Lưới thiết kế này sẽ trở thành form bạn định thiết kế. Ý tưởng để thiết kế Form là đặt các đối tượng lên lưới thiết kế. Các đối tượng có thể là các trường, các nhãn (label), các nút (button), các điều khiển (control), và các mục khác nữa, ... Bạn sẽ tìm hiểu cách đặt các đối tượng lên lưới thiết kế ngay sau chương này. Những đối tượng này có thuộc tính sẽ xác định cách chúng xuất hiện và thực hiện như thế nào.

### Chỉnh sửa các Form đã có sẵn

Bạn có thể chỉnh sửa các Form đã có sẵn bằng cách sử dụng cùng một phương pháp để tạo một form mới. Lựa chọn tên **Forms** trong cửa sổ **Database**, sau đó kích chuột vào nút **Design**. Nhiều người thường sử dụng Forms Wizard để tạo Form cơ bản và sau đó sử dụng cửa sổ thiết kế để thay đổi Form cho phù hợp.

Chú ý rằng lưới thiết kế Form không lớn, nó có chiều cao 2 inches chiều rộng 5 inches. Nếu bạn muốn tăng kích thước của lưới (trường hợp này là tăng kích thước của Form bạn đang tạo), di chuyển con trỏ chuột lên phía trên đường viền ngoài của lưới. Tiếp đó, giữ chuột và kéo đường viền tới một vị trí mới. Khi bạn nhả chuột, lưới thiết kế đã được mở rộng. Bạn có thể thấy minh họa như ở hình 2.



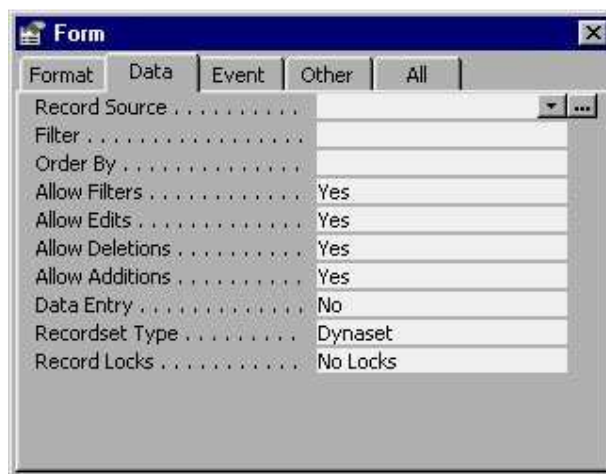
Hình 2: Kích thước lưới thiết kế

**Chú ý:** Bạn có thể phóng to hoặc thu nhỏ kích thước lưới thiết kế Form bất kỳ lúc nào muốn. Bạn sẽ cần phóng to lưới khi thêm các đối tượng mới, và trước khi thu nhỏ kích thước lưới thiết kế, bạn phải gỡ bỏ các đối tượng đã xuất hiện trong lưới thiết kế mà bạn muốn loại.

### Gắn kèm nguồn dữ liệu

Nếu bạn không chỉnh sửa Form đã có, có khả năng Form bạn đang tạo chưa được định nghĩa nguồn dữ liệu. Thao tác đầu tiên trong quá trình tạo Form mới là xác định nguồn dữ liệu. Để xác định nguồn dữ liệu chúng ta thực hiện các bước sau:

1. Kích chuột vào công cụ **Properties** trên thanh công cụ. Hay chọn Properties từ thực đơn **View**. Access hiển thị hộp thoại Properties cho Form
2. Phải chắc chắn tab Data đã được chọn. Access hiển thị hộp thoại Properties như hình 3.



Hình 3: Hộp thoại Properties cho Form mới.

3. Xem xét thuộc tính **Record Source**. Nếu có một vài cái trong thuộc tính đó, thì bạn đã có một nguồn dữ liệu được định nghĩa cho Form. Trong trường hợp này, bạn có thể bỏ qua bước tiếp theo.














4. Chọn thuộc tính Record Source và sử dụng mỗi tên trở xuống ở phía bên phải thuộc tính để chọn bảng My Friends như nguồn dữ liệu. (Hãy nhớ, My Friends là tên bảng được tạo trong cơ sở dữ liệu My Friends).








5. Đóng hộp thoại Properties bằng cách kích chuột vào nút **Close** ở góc phía trên bên phải của hộp thoại.

**Chú ý:** Record Source là một thuộc tính của Form. Ngay sau chương này bạn sẽ biết rõ hơn về cách thay đổi các thuộc tính cho Form.

## Công cụ thiết kế Form

Khi lưới thiết kế Form xuất hiện thì hộp công cụ thiết kế cũng xuất hiện. (Trong hình 1, hộp công cụ được đặt phía bên phải màn hình). Hộp công cụ này chứa các đối tượng mà bạn có thể đặt lên Form. Những đối tượng này thường được gọi là các điều khiển. Bạn có thể ẩn hoặc hiện hộp công cụ này bằng cách kích chuột vào công cụ **Toolbox** trên thanh công cụ hoặc chọn **Toolbox** từ thực đơn **View**. Hộp công cụ chứa 20 công cụ được miêu tả chi tiết trong bảng sau:

Công cụ	Hình minh họa	Mô tả
Select Objects		Được sử dụng để chọn các đối tượng đã có trên Form
Control Wizards		Khi chọn công cụ này, Access sử dụng Wizards để trợ giúp bạn tạo các điều khiển trên Form
Label		Đặt văn bản trên Form
Text Box		Được sử dụng để người dùng nhập hoặc hiển thị giá trị hiện tại, như nội dung của một trường
Option Group		Nhóm các nút, các tùy chọn và các hộp chọn với nhau. Bất kỳ những đối tượng bên trong một tùy chọn nhóm là được chọn lọc (chỉ có thể được chọn một lần trong nhóm).
Toggle Button		Một nút mở hoặc tắt. Khi được chọn, nó sẽ xem xét giá trị True (đúng)
Option Button		Công cụ này còn được gọi là nút radio. Xuất hiện như một vòng tròn, khi được chọn, nó sẽ được lấp kín.
Check Box		Xuất hiện như một hình vuông. Khi được chọn, nó có dấu tích trong đó.
Combo Box		Một hộp danh sách với một textbox phía trên của nó
List Box		Một hộp chứa một chuỗi các giá trị mà người dùng có thể chọn.
Command Button		Thực thi một macro hay một module cơ bản
Image		Chèn ảnh vào trong Form.
Unbound Object Frame		Liên kết và hiển thị ảnh, đồ thị hay các đối tượng OLE khác. Đối tượng không được lưu trữ trong bảng hay

		truy vấn.
Bound Object Frame		Cũng giống như một đối tượng khung, không kể đối tượng được lưu trữ trong một bảng hay truy vấn.
Page Break		Chia một Form thành hai phần
Tab Control		Được sử dụng để thêm một tab vào Form, giống như các hộp thoại chứa các tab
Subform/Subreport		Thêm Form hay Report khác từ một Form, Report đã có. Thường dùng để hiển thị thông tin liên quan trong các bảng.
Line		Vẽ một đường thẳng
Rectangle		Thêm một hình chữ nhật hay hình vuông
More Controls		Cho phép thêm các điều khiển ngoài các điều khiển ở hộp công cụ

**Bảng 1: Các công cụ trong hộp công cụ (từ trên xuống dưới và từ trái sang phải).**

**Chú ý:** Bạn có thể sử dụng chuột để di chuyển hộp công cụ hoặc thay đổi kích thước của nó. Bằng cách trỏ vào thanh tiêu đề của hộp công cụ và sử dụng chuột để kéo nó tới một vị trí mới hay, trỏ vào đường viền của hộp công cụ và sử dụng chuột để kéo nó tới một kích thước mới.

Để đặt một điều khiển lên Form, kích chuột vào công cụ bạn muốn và sử dụng chuột để chỉ ra nơi bạn muốn đặt điều khiển đó lên Form. Ví dụ, nếu bạn muốn đặt một Text box, đầu tiên kích chuột vào công cụ Text Box. Sau đó, sử dụng con trỏ chuột cho biết nơi bạn muốn đặt như một góc của hộp. Khi kích và giữ chuột, Access giữ góc của hộp. Khi bạn kéo, Text box sẽ lớn hơn. Khi text box đã có một kích thước phù hợp, nhả chuột. Text box sẽ được đặt tại vị trí mà bạn chọn trên Form. Bạn có thể đặt bất kỳ điều khiển khác với cách tương tự như vậy.

Mỗi một lần đặt một đối tượng lên Form, bạn có thể di chuyển nó hoặc thay đổi kích thước bằng cách sử dụng chuột. Với mọi đối tượng, khi được chọn nó sẽ được bao quanh bởi điều khiển. Hình vuông nhỏ này bao quanh điều khiển. Bạn có thể sử dụng chuột để kéo những điều khiển này tới một vị trí mới. Nếu đơn giản chỉ trỏ tới cạnh của đối tượng và giữ chuột, bạn có thể kéo toàn bộ đối tượng tới một vị trí mới trên Form. Khi nhả chuột, đối tượng sẽ ở tại vị trí đó.

**Chú ý:** Bạn có thể di chuyển một đối tượng bởi vì con trỏ chuột thay đổi thành hình bàn tay. Các đối tượng khác, hình bàn tay xuất hiện khi bạn trỏ qua cạnh của đối tượng (không phải nơi một điều khiển được đặt). Với các đối tượng khác, hình bàn tay xuất hiện khi con trỏ chuột nằm ở giữa đối tượng. Chỉ cần di chuyển chuột quanh đối tượng cho đến khi hình bàn tay xuất hiện và sau đó, sử dụng chuột để kéo đối tượng tới một vị trí mới.

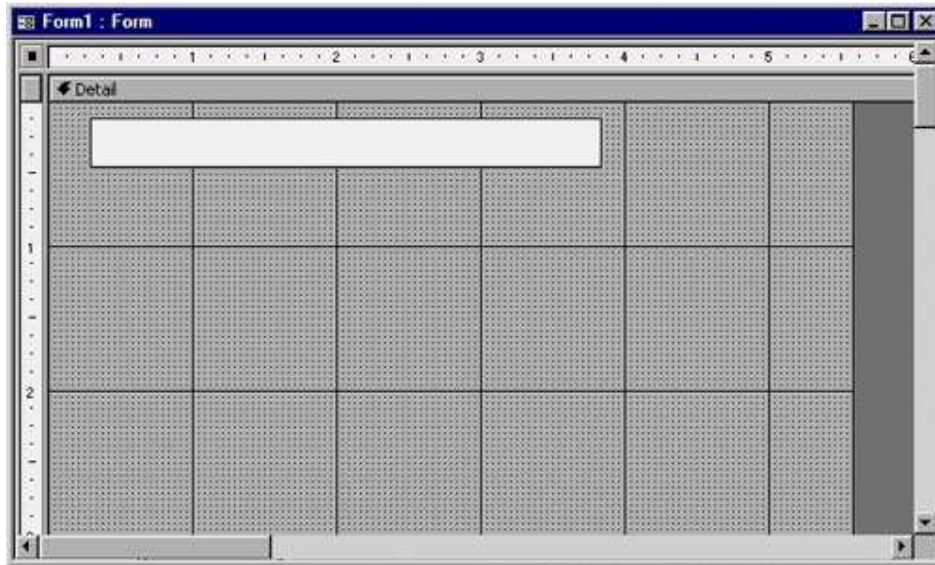
## Đặt các đối tượng lên Form

Giả sử bạn muốn tạo một Form do đó bạn có thể xem lại các bản ghi trong bảng My Friends lại một lần. Tiếp đó, bạn muốn đặt các thuộc tính cho Form. Bắt đầu với thao tác đầu tiên bằng cách sử dụng cửa sổ thiết kế Form. Đưa một đối tượng lên form, sau đó đặt các đối tượng khác. Và lặp lại quá trình cho đến khi hoàn thành.

Thao tác đầu tiên trong việc tạo Form là điều chỉnh kích thước của lưới thiết kế Form cho phù hợp. Sau đó, có thể thay đổi màu nền cho Form. Để đổi màu nền Form, chọn Form đó (chỉ cần kích chuột vào lưới form), sau đó điều chỉnh màu bằng cách sử dụng công cụ **Fill/Back Color** trên thanh công cụ (công cụ này trông giống như một chổi vẽ). Kích chuột vào mũi tên xuống phía bên phải công cụ, Access sẽ hiển thị một bảng màu. Bạn có thể chọn bất kỳ màu nào. Kích chuột vào màu đó. Nếu bạn đặt một màu khác với màu hiện thời, chú ý rằng Access sẽ thay toàn bộ lưới thiết kế với màu đó.

## Đặt đối tượng Label

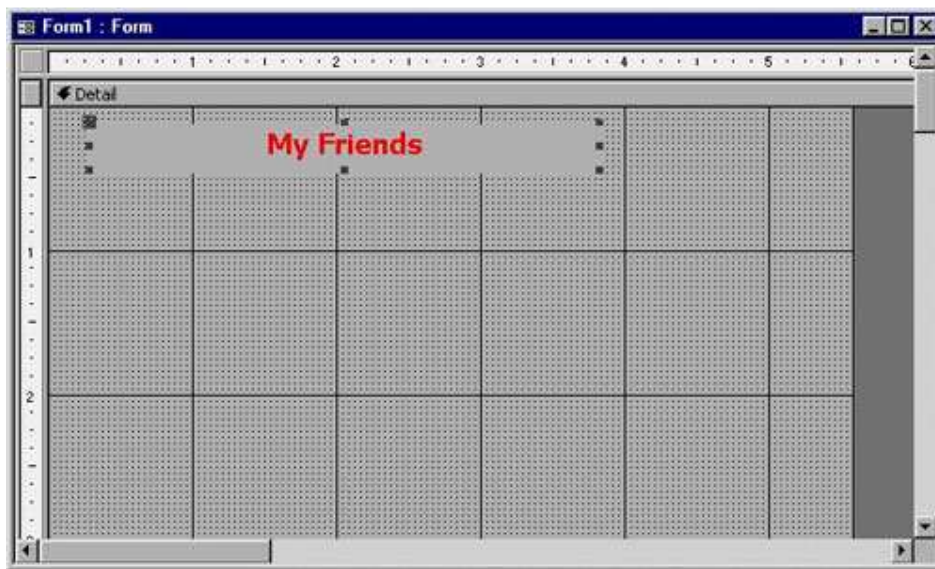
Tiếp theo, có thể bạn muốn có một vài đoạn chữ nhỏ như là tiêu đề trên Form. Để gán một tiêu đề, bạn đặt đối tượng Label lên form bằng cách chọn từ hộp công cụ. Đặt Label lên form và điều chỉnh theo kích thước phù hợp. Trường hợp này được minh họa như hình 4.



Hình 4: Sau khi đặt đối tượng Label

Lúc này Access đợi bạn nhập nội dung mà muốn xuất hiện trên label. Hãy nhập từ My Friends và sau đó ấn Enter. Nội dung đó sẽ xuất hiện ở phía bên trái hộp Label và các điều khiển xuất hiện xung quanh. Bạn có thể thiết lập màu cho đối tượng. Thiết lập màu nền (sử dụng công cụ Fill/BackColor) và màu đường viền (công cụ Line/Border Color) hoặc chọn màu cùng với màu nền form hay màu trong suốt (Transparent). Sau đó bạn có thể thiết lập màu chữ bằng cách sử dụng công cụ Font/Fore Color.

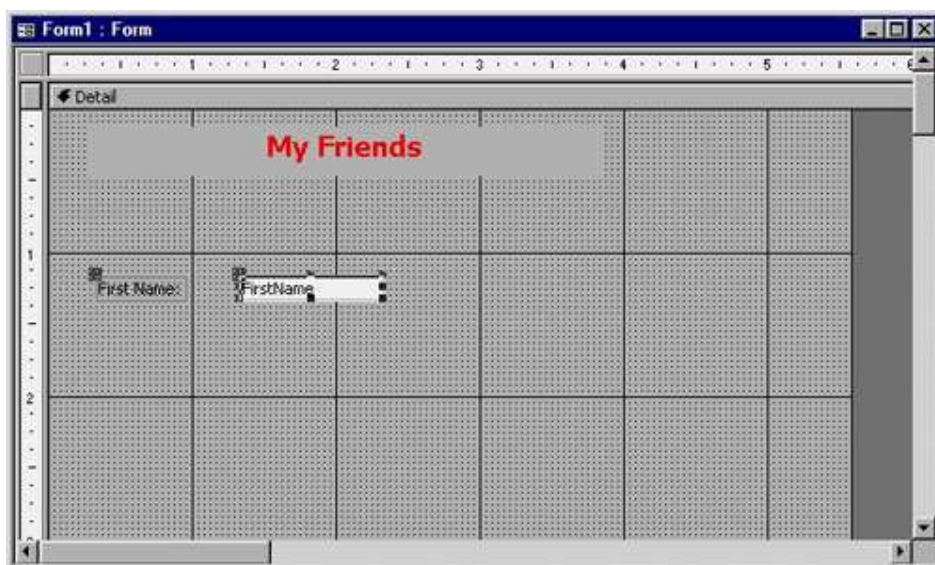
Phải chú ý tới công cụ phía trên màn hình. Khi bạn đã chọn một đối tượng chứa văn bản, các điều khiển nội dung đó được kích hoạt. Bạn có thể thiết lập các thuộc tính như font, kích thước, ... Bạn nên thay đổi các thuộc tính cho Label này để nội dung trông như một tiêu đề Form. Hình 5 hiển thị minh họa cho thiết lập các thuộc tính Label (cỡ chữ 14, Tahoma, đậm, đặt giữa hộp Label).



Hình 5: Kết thúc tiêu đề Form

### Đặt các trường lên Form

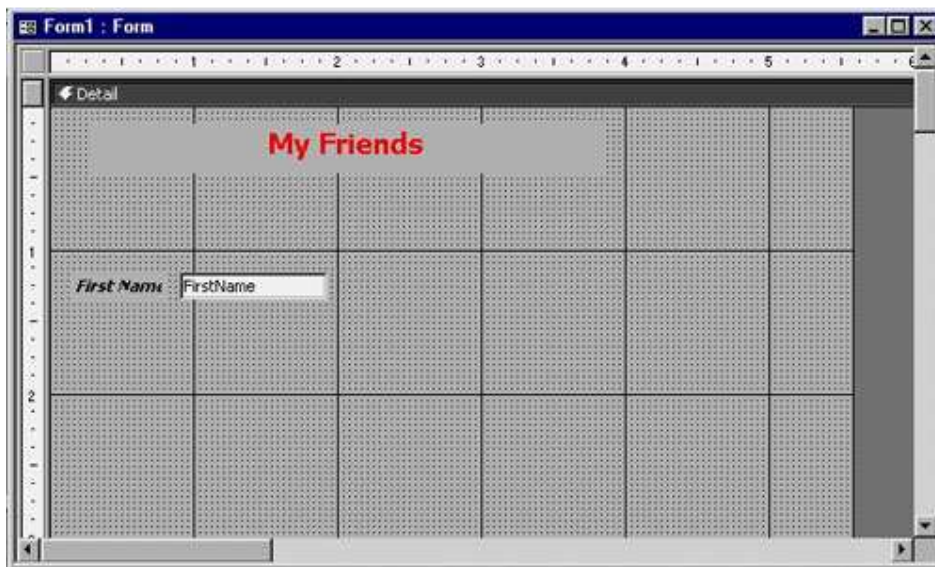
Để bắt đầu, chọn **Field List** từ thực đơn **View**. Sau khi chọn, danh sách các trường sẽ hiển thị như nguồn dữ liệu (bảng My Friends). Tiếp đó, sử dụng chuột để chọn một trường và kéo nó lên Form. Ví dụ, kích chuột vào trường **FirstName** và kéo lên form. Khi nhả chuột, Access đặt hai đối tượng lên Form: một Label và một Text box. Đối tượng label chứa thuộc tính **Caption** cho trường, đối tượng Text box chứa tên của trường đó. Hình minh họa sau:



Hình 6: Đặt một trường lên form

Bạn có thể sử dụng chuột để điều chỉnh vị trí của đối tượng Label và Text box trên màn hình. (Nếu bạn di chuyển Label và Text box để chúng tách hẳn nhau) Sử dụng màu và định dạng trên thanh công cụ để điều chỉnh các đối tượng sao cho chúng dễ nhìn hơn trên Form. Khi hoàn thành, các đối tượng Label và Textbox sẽ xuất hiện như minh họa hình 7.



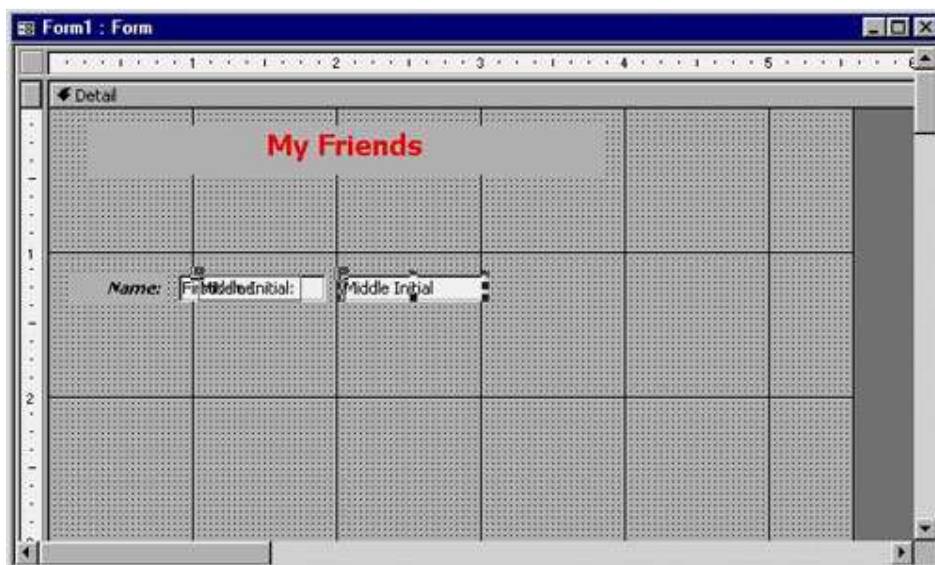


Hình 7: Định dạng các điều khiển.

Nó sẽ thuận hơn nếu thay đổi nội dung trong đối tượng Label để chỉ còn Name thay vì FirstName. Theo cách này, người dùng có thể nhập toàn bộ tên vào phía bên phải đối tượng Label. Để làm điều này, chọn đối tượng Label, khi bạn di chuyển con trỏ chuột lên trên nội dung trong Label, chú ý rằng con trỏ thay đổi thành điểm chèn. Kích chuột và bạn có thể thay đổi nội dung trong đối tượng Label một cách dễ dàng. Khi hoàn thành, ấn Enter, Access sẽ thực hiện sự thay đổi đó.

**Chú ý:** Bất cứ lúc nào bạn thay đổi nội dung trong một Label, bạn cũng có thể muốn thay đổi kích thước của đối tượng Label. Thực hiện bất kỳ sự điều chỉnh bạn cho là cần thiết để đối tượng hợp lý hơn.

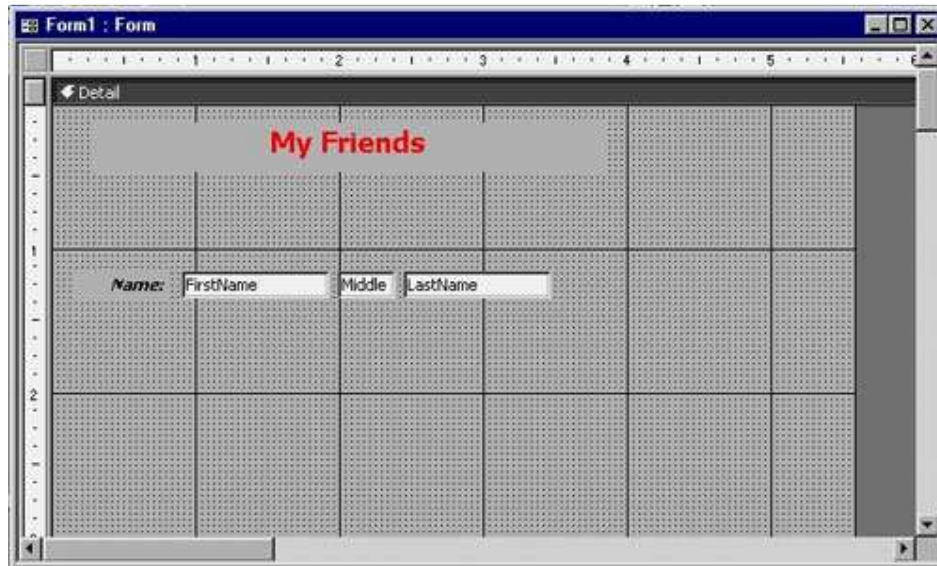
Lúc này bạn muốn đặt thêm trường lên Form. Bạn thực hiện cũng với cách làm như đặt trường đầu tiên. Chỉ cần kích chuột vào trường Middle Initial trong danh sách trường và kéo lên lưới thiết kế. Vị trí con trỏ chuột sao cho nó chỉ ở một chút bên phải Text Box đã sử dụng cho trường FirstName. Khi bạn thả chuột, Access đặt hai đối tượng lên form như hình sau:



Hình 8: Đặt trường thứ 2 lên Form

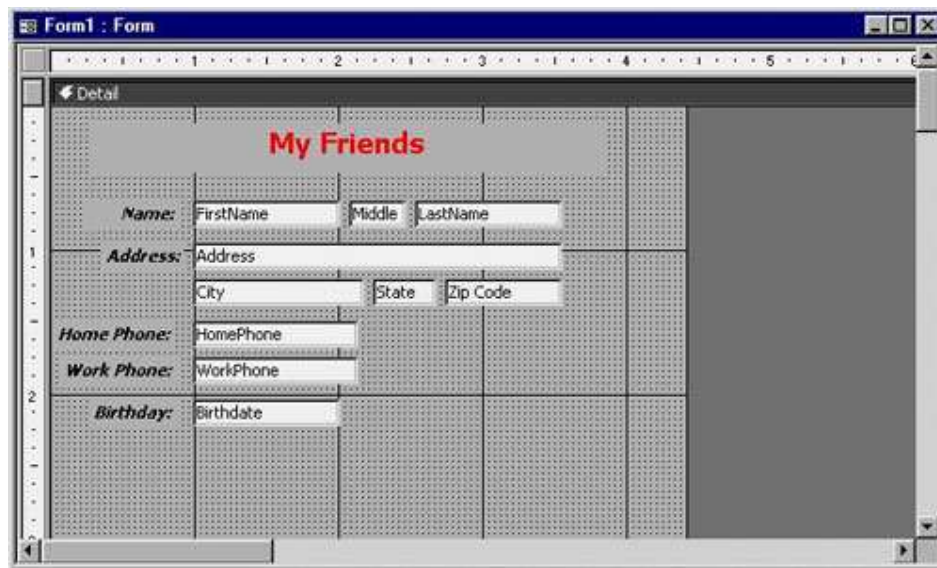
Chú ý rằng, đối tượng Label nằm chồng lên trường đã đặt trước đó. Do vậy mà bạn không nhất thiết phải đặt tên vào nhãn (vì đối tượng nhãn trước đó là Name đã phù hợp cho toàn bộ rỗng), sử dụng chuột để chọn nó và nhấn phím DEL. Hộp Label sẽ biến mất nhưng Text box thì vẫn còn. Lúc này bạn có thể đặt vị trí Textbox và thay đổi các thuộc tính, kích thước của nó sao cho phù hợp với các đối tượng khác.

Bạn có thể đặt trường LastName với cách như vậy. Kết quả được hiển thị như hình 9



Hình 9: Hoàn thành dòng đầu tiên của Form

Những điều cơ bản nhất là đặt các trường lên Form. Bạn nên chỉnh trường đầu tiên và đặt những trường còn lại và thay đổi các thuộc tính của chúng sao cho đẹp mắt nhất. Sử dụng các thao tác như đã làm với 3 trường đầu tiên. (Không cần đặt trường Christmas Card, nó sẽ được nói đến trong phần tiếp theo). Hình 10 hiển thị cửa sổ Design sau khi thiết lập các trường cho Form.



Hình 10: Form gần như hoàn thành

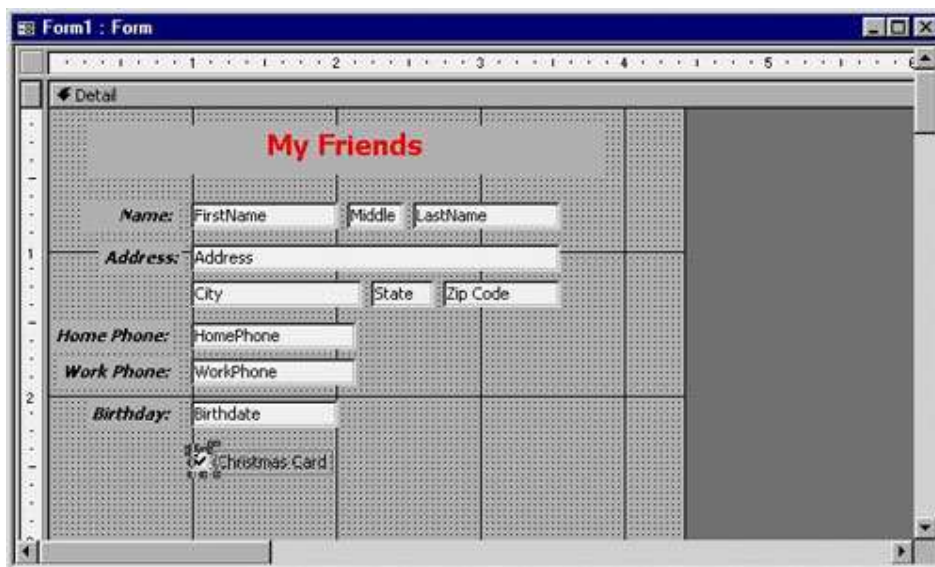
## Đặt đối tượng Check Box

Trường cuối cùng bạn cần đặt là trường Christmas Card. Bạn có thể nhớ lại Chương 2, "Học Access trong 1 giờ", trường Yes/No cho biết bạn có gửi một thẻ Christmas tới một cá nhân hay không. Có thể trình bày trường Yes/No với bất kỳ các đối tượng sau:

- **Toggle Button**
- **Option Button**
- **Check Box**

Bất kỳ những điều khiển này có thể cho biết trạng thái Yes/No hay tình trạng On/Off, sự khác nhau giữa chúng chỉ là hình thức. Tuy nhiên, tác dụng trên một người dùng có thể khá khác nhau. Ví dụ, một Toggle button cho biết nếu một số tình trạng on hay off, trong khi hai trạng thái khác chính xác hơn tương ứng với Yes hoặc No. Thực tế, đối tượng bạn sử dụng phụ thuộc vào việc bạn chọn, nhưng trường hợp này, nên chọn đối tượng Check Box.

Để đặt một hộp chọn cho trường Christmas Card, kích chuột vào công cụ Check Box trong hộp công cụ. Sau đó, kéo trường Christmas từ danh sách trường lên Form. Khi nhả chuột, Access đặt hai đối tượng lên Form: đối tượng Check Box và một đối tượng Label. Minh họa như hình sau:



Hình 11: Đặt hộp chọn

Chú ý rằng, không giống như các trường khác bạn đã đặt trước đó, Access đặt đối tượng Label ở phía bên phải đối tượng Check Box (thay vì phía bên trái). Access đặt phía bên phải để tuân theo các hướng dẫn giao diện cho Windows. Hãy định vị các đối tượng trên form và định dạng chúng sao cho thích hợp nhất. Bạn cũng có thể muốn thay đổi nội dung trong đối tượng Label để phù hợp với mục đích của hộp Check Box, có thể thay đổi với nội dung như Send Christmas card.

### Tìm hiểu các thuộc tính

Ở phần này, bạn cũng đã khá quen thuộc với khái niệm về thuộc tính. Bạn biết rằng Access sử dụng các thuộc tính để mô tả cách một đối tượng xuất hiện hoặc cách Access xử lý nó như thế nào. Bạn cũng biết rằng các trường, các bảng và truy vấn đều có những thuộc tính. Các đối tượng bạn đặt trên một Form cũng có thể có các thuộc tính. Các thuộc tính chính xác phụ thuộc vào đối tượng bạn đã chọn.

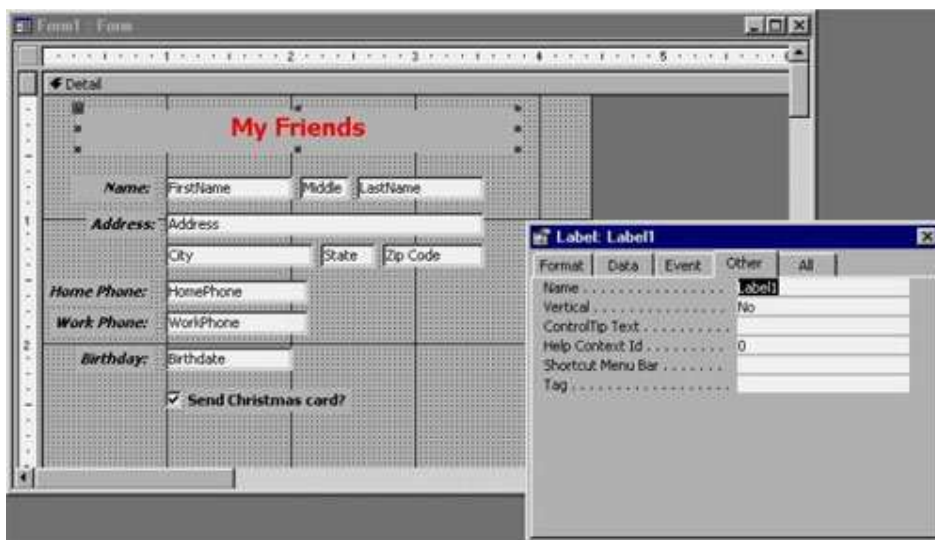
Thông thường, các thuộc tính miêu tả thuộc tính của một đối tượng. Ví dụ, độ cao, rộng và vị trí là thuộc tính của một đối tượng. Kể cả màu sắc và phông chữ. Khi làm việc qua các phần trước, bạn đã thiết lập các thuộc tính cho một vài đối tượng mặc dù bạn có thể chưa có kiến thức về nó.

## Thay đổi thuộc tính đối tượng

Bạn có thể xem và thay đổi các thuộc tính của một đối tượng bằng một trong hai phương thức sau:

- Kích đúp chuột vào đối tượng
- Chọn đối tượng và kích chuột vào công cụ Properties trên thanh công cụ.

Khi đã thực hiện một trong hai cách trên, Access hiển thị hộp thoại Properties cho đối tượng đó. Ví dụ, Hình 12 hiển thị hộp thoại Properties cho đối tượng Label đầu tiên mà bạn đã đặt lên Form (đối tượng được sử dụng làm tiêu đề cho Form).



Hình 12: Hộp thoại Properties cho đối tượng Label

Tất cả có 38 thuộc tính cho đối tượng này. Đối với các đối tượng khác, có thể có nhiều thuộc tính hơn hoặc có thể có ít hơn. Số thuộc tính là bao nhiêu không quan trọng, nhưng quan trọng là làm cách nào để biết thay đổi một thuộc tính.

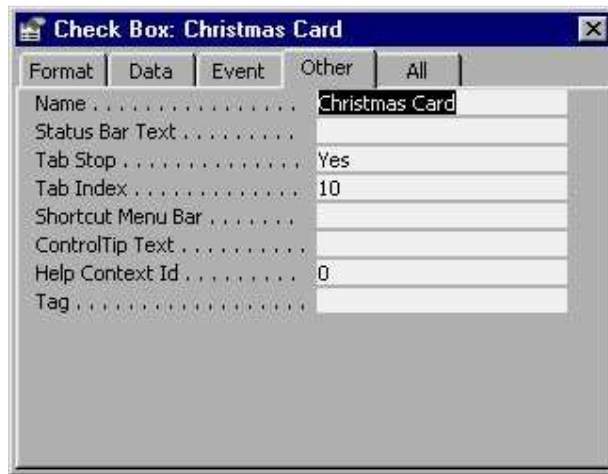
Để thay đổi một thuộc tính, đầu tiên phải hiển thị hộp thoại Properties. Tiếp đó, sử dụng chuột để chọn mục thuộc tính nào trong các tab ở hộp thoại đó. Sau đó, chọn thuộc tính bạn muốn thay đổi. Bạn có thể nhập một giá trị mới hoặc sử dụng mũi tên xổ xuống để chọn một giá trị thích hợp. Khi thay đổi một thuộc tính, kết quả sẽ thay đổi ngay.

**Chú ý:** Khi thay đổi một số thuộc tính, sẽ dễ dàng khi không sử dụng hộp thoại Properties. Ví dụ, để thay đổi màu sắc thì sử dụng thanh công cụ sẽ nhanh hơn. Nếu có thể, sử dụng một cách khác để thay đổi thuộc tính khi một trong các thuộc tính sẵn có.

Vì có rất nhiều các thuộc tính khác nhau sẵn có, bạn phải mở đầu là tự hỏi những thuộc tính phổ biến được sử dụng cho cái gì. Cách tốt nhất để tìm đáp án là chọn một thuộc tính và xem trạng thái của Access. Bạn sẽ tìm thấy thông tin mà cung cấp nhanh nhất để trợ giúp những gì

mà một thuộc tính thực hiện. Nếu bạn cần thêm thông tin trợ giúp, bạn có thể sử dụng hệ thống trợ giúp trực tuyến. Hệ thống trợ giúp miêu tả đầy đủ mục đích và cách sử dụng của một số các thuộc tính.

Ví dụ về thay đổi thuộc tính. Chọn đối tượng Check Box mà bạn đã đặt lên Form ở phần trước. Chú ý rằng các thuộc tính trong hộp thoại Properties thay đổi tương ứng với đối tượng mới mà bạn chọn. Khi chọn tab Other trong hộp thoại, nó sẽ xuất hiện như hình sau:



Hình 13: Hộp thoại Properties cho đối tượng Check Box

Một thuộc tính giúp ích mà bạn có thể thay đổi là thuộc tính Status Bar Text. Đây là thuộc tính chỉ ra những gì Access nên hiển thị trên thanh trạng thái khi điều khiển Christmas Card được chọn. Tất cả bạn cần làm là chọn thuộc tính Status Bar Text và bắt đầu nhập nội dung bạn muốn hiển thị. Ví dụ, nhập nội dung Should this person receive a Christmas card? Khi hoàn thành với hộp thoại Properties, đóng nó lại.

**Chú ý:** Rất nhiều thuộc tính cho các đối tượng trên Form cũng giống các thuộc tính cho các trường trong Access. Bạn có thể xem lại trong Chương 3 để biết thêm thông tin về các thuộc tính.

## Thay đổi thuộc tính Form

Form cũng có các thuộc tính. Bạn có thể thấy những thuộc tính này bằng cách chọn Form. Để chọn Form, kích chuột vào các khối nhỏ màu xám phía bên trái thanh ngang ở trên cùng cửa sổ Detail. Nếu hộp thoại Properties được mở, bạn sẽ thấy các thuộc tính cho toàn bộ Form. Có 81 thuộc tính có thể được áp dụng cho các Form, và Access thiết lập nhiều thuộc tính một cách tự động khi bạn chỉnh sửa Form. Bạn đã thay đổi một thuộc tính cho Form ở ngay đầu chương khi bạn thay đổi thuộc tính Record Source cho Form.

Nhiều thuộc tính rất hữu ích cho ngôn ngữ lập trình cấp cao. Tuy nhiên, bạn có thể sử dụng trực tiếp một trong số đó. Ví dụ, bạn muốn thay đổi từ mà xuất hiện trong thanh tiêu đề Form. Theo mặc định, thanh tiêu đề sẽ hiển thị từ "Form", trước nó là dấu hai chấm và tên Form. Tuy nhiên, vì bạn làm việc với bảng chứa thông tin về những người bạn của bạn, nên bạn muốn thanh tiêu đề hiển thị tên.

Để thực hiện thay đổi này, chỉ cần chỉnh sửa thuộc tính Caption cho Form. Bất cứ từ nào bạn đặt trong thuộc tính Caption sẽ xuất hiện trên thanh tiêu đề của Form. Khi thay đổi thuộc tính Caption tôi thay thế bằng tên của mình, Access sẽ thay đổi như hình 14 sau:

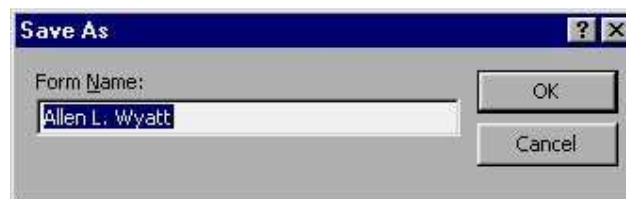
The image shows a screenshot of an Access form window. The title bar at the top reads 'Allen L. Wyatt'. The form itself has a title 'My Friends' in red text. Below the title, there are several input fields: 'Name' with 'John' and 'Davis' entered, 'Address' with '123 Maple Drive', 'Rapid City', 'SD', and '57701-', 'Home Phone' with '(605) 331-2028', 'Work Phone' with '(605) 303-2226', and 'Birthday' with '2/ 9/64'. At the bottom, there is a checkbox labeled 'Send Christmas card?'. The status bar at the bottom left shows 'Record: 1 of 19'.

Hình 14: Form sau khi thay đổi thuộc tính Caption

**Chú ý:** Access không thay đổi thanh tiêu đề cho đến khi bạn thực sự mở Form. Nếu bạn sử dụng cửa sổ thiết kế, Form sẽ sử dụng thanh tiêu đề mặc định.

## Lưu Form

Khi đã hoàn thành việc tạo và thay đổi Form, bạn cần lưu nó lại. Access tự động cho bạn một tên mặc định khi bạn đóng cửa sổ thiết kế. Nếu bạn đã thực hiện thay đổi form (hoặc nếu bạn đang thiết kế Form lần đầu). Access sẽ hỏi bạn có muốn lưu sự thay đổi không. Tác dụng của việc kích chuột vào nút Yes phụ thuộc vào việc bạn có chỉnh sửa hay tạo Form không. Nếu bạn chỉnh sửa Form, Access sẽ lưu sự thay đổi đó dưới tên Form đã sử dụng trước đây. Nếu bạn đang tạo Form, Access sẽ hỏi bạn để cung cấp một tên Form, như hình sau:

The image shows a 'Save As' dialog box. The title bar says 'Save As'. Inside the dialog, there is a label 'Form Name:' followed by a text box containing 'Allen L. Wyatt'. To the right of the text box are two buttons: 'OK' and 'Cancel'.

Hình 15: Bạn có thể xác định tên cho Form

Trong trường hợp này, Access đưa ra gợi ý một tên dựa vào phụ đề tôi đã đặt cho Form (bạn hãy nhớ, tên này được đặt khi thiết lập sự thay đổi thuộc tính Form). Tên này có thể được thay đổi theo nhiều cách khác nhau. Bạn nên sử dụng tên để thể hiện cho ý nghĩa của Form. Trong trường hợp tôi đã thay đổi tên Form thành Reviewing Friends. Khi ấn phím ENTER, Access lưu Form và trở lại cửa sổ Database.

# 37: Tổng kết chương 9

**Form cung cấp một cách thuận lợi và tiện dụng để biểu diễn thông tin và nhập dữ liệu từ phía người dùng.** Nếu bạn đang phát triển một ứng dụng sử dụng Access, nó không hiếm để có một vài biểu mẫu khác nhau, mỗi biểu mẫu có một ý nghĩa khác nhau. Xuyên suốt hiệu quả sử dụng Form, bạn sẽ làm tăng dần tính hữu ích cho cơ sở dữ liệu.

Chương này đã giúp bạn hiểu được cách xây dựng các Form riêng. Trước khi chuyển sang chương 10, hãy chắc chắn là bạn nắm được những nội dung sau:

- Forms là một công cụ bạn có thể sử dụng để thay đổi hoặc thêm thông tin trong một bảng hay truy vấn. Từ cách sắp xếp của người dùng, các Form sẽ tiện lợi hơn ở chế độ Datasheet.
- Bạn có thể tạo các Form bằng cách sử dụng Form Wizards, hoặc tạo chúng từ đầu.
- Bạn có thể tạo một số Form giới hạn với Form Wizards. Nhưng bạn sẽ có nhiều tùy chọn hơn nếu tạo Form ngay từ đầu.
- Sử dụng hộp công cụ thiết kế Form để chọn những đối tượng có thể được đặt lên Form.
- Như phần lớn các đối tượng khác trong Access, bạn có thể thiết lập các thuộc tính cho các đối tượng trong Form. Những thuộc tính này xác định các thuộc tính để điều khiển đối tượng.

Trong Chương 10, bạn sẽ học cách sử dụng một số công cụ bổ sung được cung cấp bởi Access để thực hiện dữ liệu một cách hợp lý hơn.

## Bài 38: Sử dụng tính năng AutoCorrect

**Nếu bạn là một người sử dụng Word (bộ xử lý từ phổ biến của Microsoft) bạn có thể rất quen thuộc với tính năng AutoCorrect.** Công cụ này cũng thích nghi với Access để trợ giúp bạn thực hiện hiệu quả hơn trong việc nhập dữ liệu.

Mục đích của **AutoCorrect** là tự động thay thế những gì bạn nhập vào. Điều này nghe có vẻ kỳ lạ, nhưng nó thực sự là một lợi ích. Ví dụ, nếu bạn thường xuyên bị lỗi chính tả khi nhập dữ liệu, bạn có thể tác động Access để nhận ra từ đó và thay thế nó bằng một từ thích hợp. Tương tự, bạn có thể xác định đoạn mã ngắn mà có thể tự thay thế cho các từ hoặc các cụm từ dài.

Sử dụng AutoCorrect có lẽ là tính năng trợ giúp mạnh mẽ khi nhập dữ liệu. Ví dụ: khi bạn nhập từ **Amalgamated Widgets Worldwide**, bạn có thể định nghĩa cụm từ đó trong AutoCorrect là **aww** để khi nhập từ đó, nó sẽ tự động thay thế thành Amalgamate Widgets Worldwide.

### Thêm các mục từ

Để thêm thông tin vào AutoCorrect, chọn **AutoCorrect** từ thực đơn **Tools**. Access hiển thị hộp thoại AutoCorrect như hình 1.



Hình 1: Hộp thoại AutoCorrect

Ở phía trên hộp thoại AutoCorrect bạn có thể điều khiển một số thay đổi tự động. Gần phía dưới hộp thoại là danh sách mục từ của AutoCorrect. Mỗi mục từ được tạo bởi 2 phần: phần phía bên trái (Replace) là phần nhập mã từ tắt cho từ hay cụm từ dài ở phần phía bên phải, phần bên phải (With) thì bạn nhập từ hay cụm từ muốn được tự động hiện ra khi gõ mã từ tắt.

Ví dụ: khi bạn phải nhập cụm từ Amalgamated Widgets Worldwide nhiều. Để không mất thời gian, bạn làm như sau: Ở phần Replace, bạn nhập mã từ tắt là aww, còn ở phần With, bạn nhập Amalgamated Widgets Worldwide, sau đó kích vào nút Add để thêm vào danh sách các mục từ AutoCorrect. Bây giờ, bạn chỉ cần nhập mã từ aww, ngay lập tức nó sẽ tự động đổi thành cụm từ Amalgamated Widgets Worldwide.

Như vậy, để thêm các mục từ tùy thích, bạn thực hiện bằng cách sử dụng hộp Replace và With. Tất cả bạn cần làm là nhập vào hộp Replace và nhập vào hộp With từ mà bạn muốn nó được thay (làm tương tự như ví dụ với cụm từ Amalgamated Widgets Worldwide).

### Loại bỏ mục từ

Chỉ khi bạn thêm các mục từ vào danh sách AutoCorrect, bạn cũng có thể xóa chúng. Để làm điều này, chỉ cần chọn mục từ trong hộp thoại AutoCorrect (hình 1) và kích chuột vào nút Delete. Access sẽ bỏ mục từ từ danh sách và không còn tự động thay đổi từ đã được sử dụng.

Bạn nên chú ý rằng Access không đòi hỏi xác minh sự xóa bỏ; nó chỉ thực hiện xóa. Tuy nhiên, các thông tin đã xóa vẫn còn trong hộp Replace và With sau khi xóa. Điều này có nghĩa nếu bạn nhận ra mục từ được xóa mà không muốn xóa, ngay lập tức bạn có thể kích chuột vào nút Add để nhập lại mục từ trong danh sách AutoCorrect.

## Bài 39: Phân tích bảng

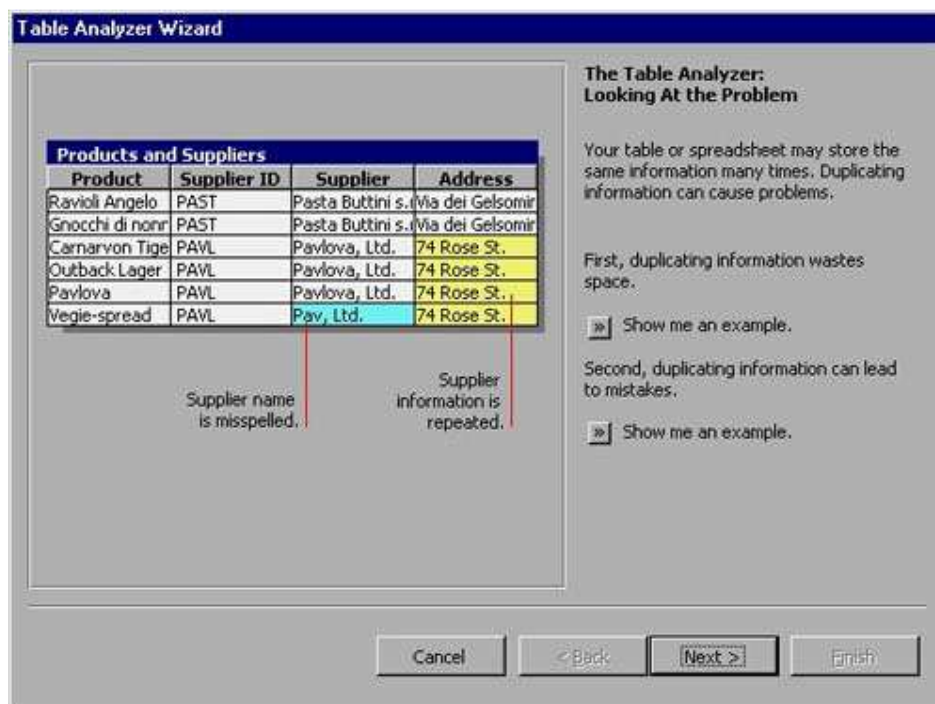


**Access gồm một công cụ rất mạnh mà bạn có thể sử dụng để phân tích dữ liệu trong một hoặc nhiều bảng.** Mục đích của việc phân tích là xác định nếu có ít dư thừa trong việc lưu trữ dữ liệu trong bảng. "Ít dư thừa" ở đây có nghĩa là Access cố gắng loại bỏ bất kỳ dữ liệu nào mà được lưu trữ trong bảng nhiều hơn một lần. Ví dụ, trường "company name" nên được đưa ra ngoài bảng gốc và được thay thế bằng một bảng mới. Mặc dù kết quả sẽ là 2 bảng (bảng gốc và bảng mới), nhưng cơ sở dữ liệu sẽ hiệu quả hơn vì thông tin không bị nhân đôi trong bảng gốc.

Một ví dụ có lẽ là cách tốt để chứng minh quá trình phân tích. Ví dụ trong phần này dựa trên cơ sở dữ liệu Accounts.mdb. Cơ sở dữ liệu này chứa một bảng tên là Invoices, có trên 1000 hóa đơn trong đó. Một số trường tối thiểu được duy trì trong bảng nhưng có khá nhiều sự dư thừa trong dữ liệu.

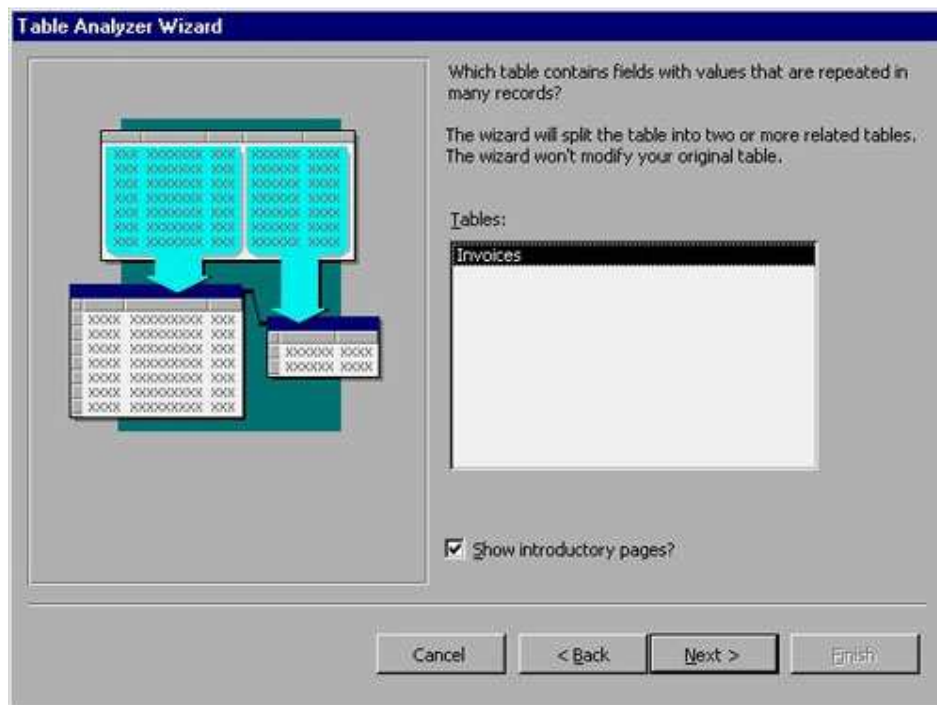
## Bắt đầu phân tích

Với cơ sở dữ liệu Accounts, kích chuột vào nút **Tables** trong cửa sổ **Database**. Kích chọn bảng **Invoices**. Sau đó, chọn **Analyze** từ thực đơn **Tools**. Access hiển thị một trình đơn phụ từ việc chọn bảng như hình sau:



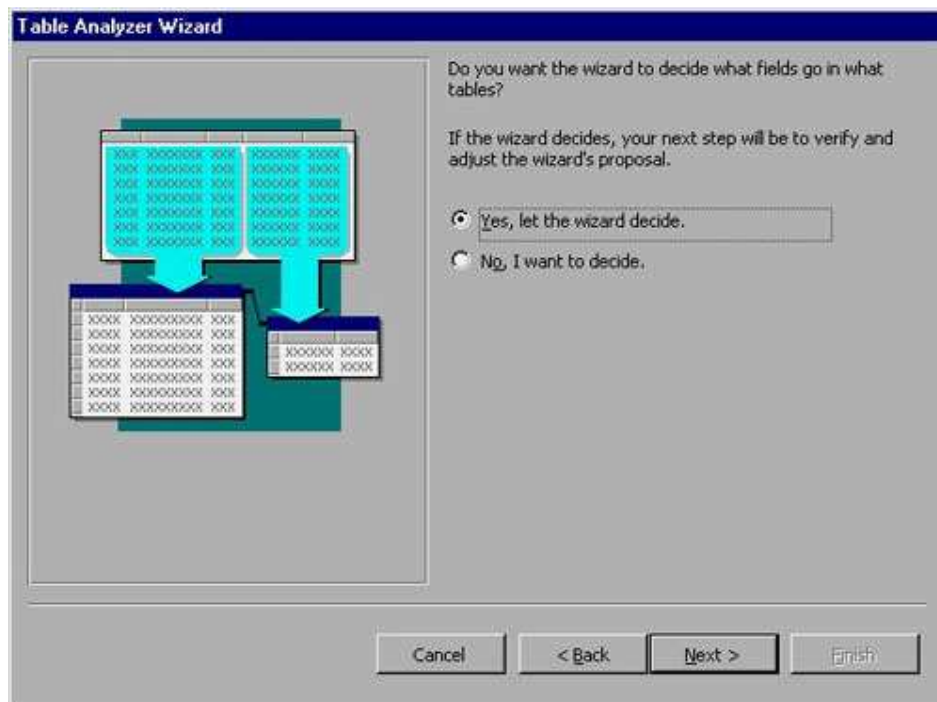
Hình 1: Hộp thoại Table Analyzer Wizard.

Việc đầu tiên trên màn hình Wizard là việc giải thích mục đích của Table Analyzer Wizard. Bạn có thể đọc hết các thông tin nếu muốn, sau đó kích chuột vào nút Next khi đã sẵn sàng. Hộp thoại hiển thị như hình sau:



Hình 2: Chọn bảng để phân tích

Hầu hết các cơ sở dữ liệu có thể có nhiều hơn một bảng. Thật may là cơ sở dữ liệu này chỉ chứa một bảng có tên Invoices. Vì đây là bảng đã mặc định được chọn, kích chuột vào nút Next. Access hiển thị hộp thoại tiếp theo:

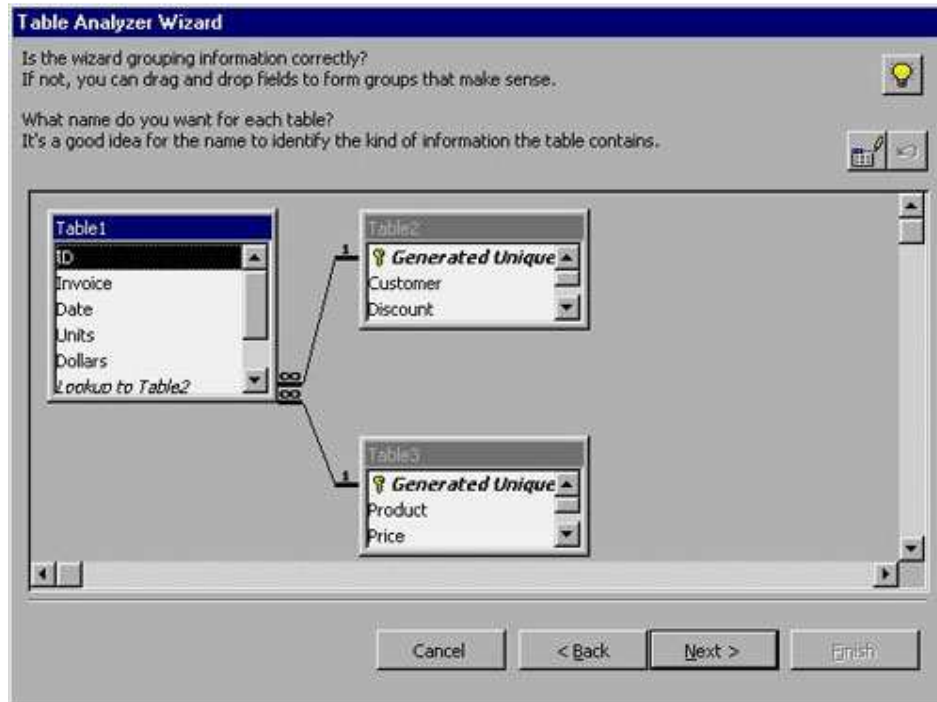


Hình 3: Bắt đầu quá trình phân tích

Ở bước này bạn cần quyết định là mình muốn cho phép Wizard lựa chọn các trường trong bảng chọn hay bạn muốn thực hiện theo cách thủ công. Vì công việc của một Wizard thực hiện khá dễ dàng, ta chọn **Yes, let the wizard decide**. Thậm chí nếu Wizard thực hiện lựa chọn không tốt, bạn có thể hủy bỏ quyết định ở bước tiếp theo.

## Xác nhận việc tách bảng

Khi bạn đã sẵn sàng, kích chuột vào nút Next. Access hiển thị như hình sau:



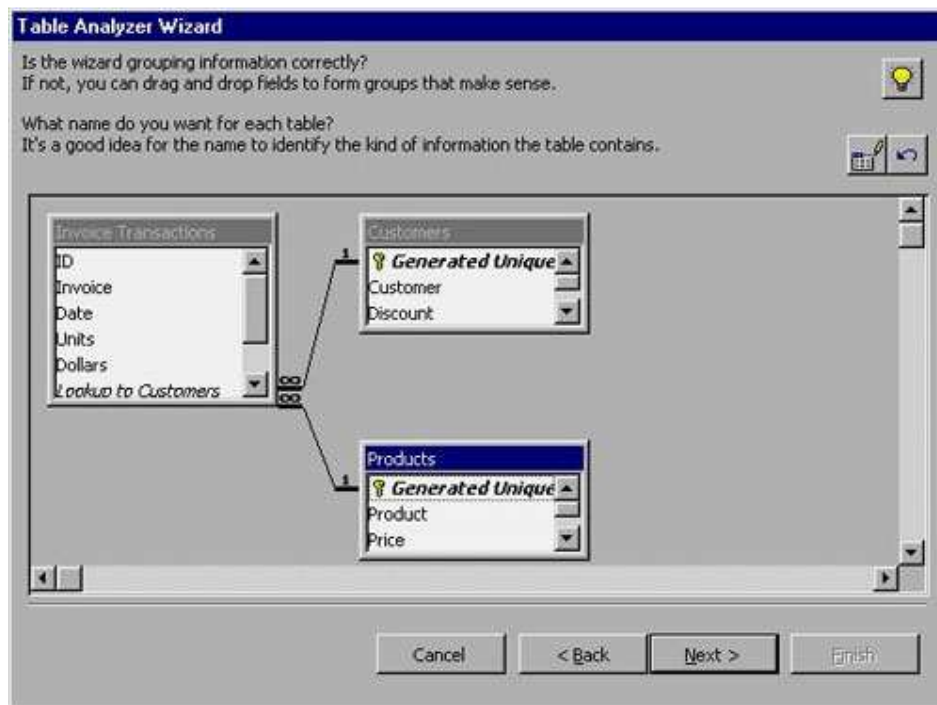
Hình 4: Xác minh việc tách bảng

**Chú ý:** Table Analyzer Wizard thiết lập các mối quan hệ giữa các bảng. Bạn sẽ tìm hiểu rõ hơn mối liên hệ được thực hiện như thế nào ở chương 13 “Tìm hiểu mối quan hệ dữ liệu”

Ở phần này, Wizard cho bạn thấy những gì nó đề ra để làm. Đối với bảng **Invoices** gốc, Wizard dự định chia thành 3 bảng. Hãy xem các trường mà Wizard gợi ý chuyển thành mỗi bảng, rất dễ để thấy mỗi bảng chứa thông tin các hóa đơn giao dịch, các khách hàng, và các sản phẩm. Sử dụng hộp thoại này, bạn có thể đặt tên cho các bảng.

Để thấy cách thực hiện như thế nào, phải chắc chắn Table1 được chọn (hình 4). Sau đó kích chuột vào nút **Rename** (ở gần phía góc trên bên phải hộp thoại). Access hiển thị một hộp thoại nhỏ cho phép bạn xác định tên bảng mới. Đối với Table1, tên nên được đặt là Invoice Transactions. Sau đó, kích chuột vào nút OK. Access cập nhật tên như trong thanh tiêu đề đối với bảng có tên mới.

Bạn nên lặp lại công việc này để đặt tên cho Table2 và Table3 là Customers và Products. Kích chuột lên thanh tiêu đề mỗi bảng và sau đó đổi tên. Khi hoàn thành, các bảng sẽ xuất hiện như hình sau:



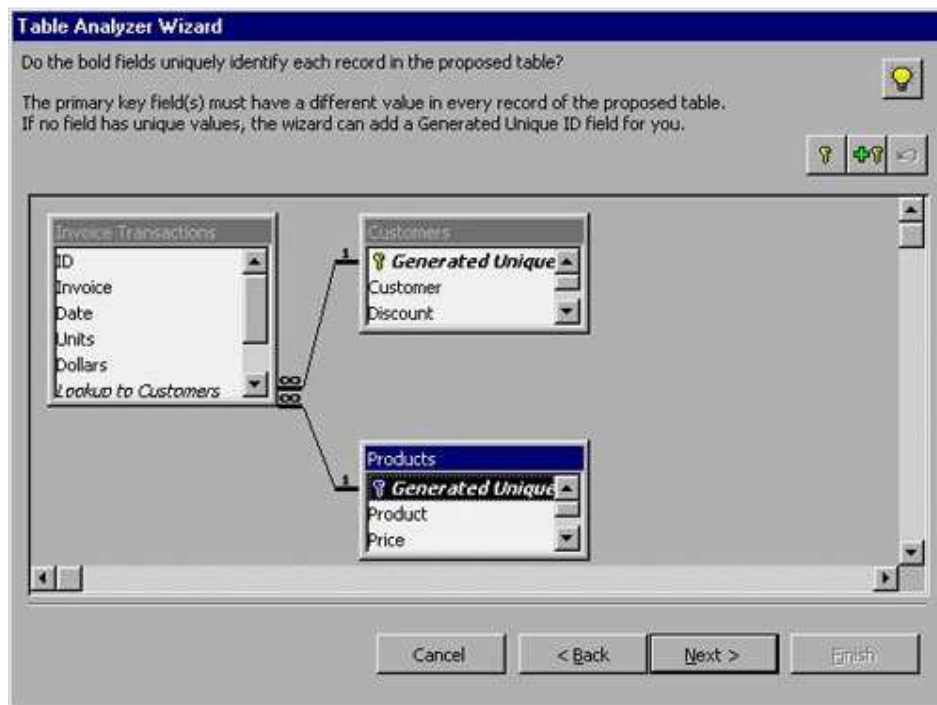
Hình 5: Các bảng đã được đổi tên.

Những việc khác bạn có thể làm trong hộp thoại này (ngoài việc đặt tên cho các bảng) là ghi đề cách Wizard chia bảng. Khi thực hiện phân tích, bạn sẽ muốn chắc chắn rằng mỗi bảng được đưa ra phải chứa các trường liên quan từ bảng gốc. Trong ví dụ này, Wizard đã thực hiện rất tốt, nó chỉ ra rằng chúng ta đã có tên bảng Customers cần có một CustomerID duy nhất cũng như CustomerName và Discount.

Nếu bạn cần ghi đề việc phân chia của Wizard, tất cả cần làm là kích chuột vào một tên trường và kéo nó từ một bảng tới bảng khác. Tuy nhiên, trong ví dụ này không thật sự cần thiết phải làm như vậy.

### Xác nhận khóa chính

Khi kích chuột vào nút Next, Access hiển thị bước tiếp theo như hình sau:



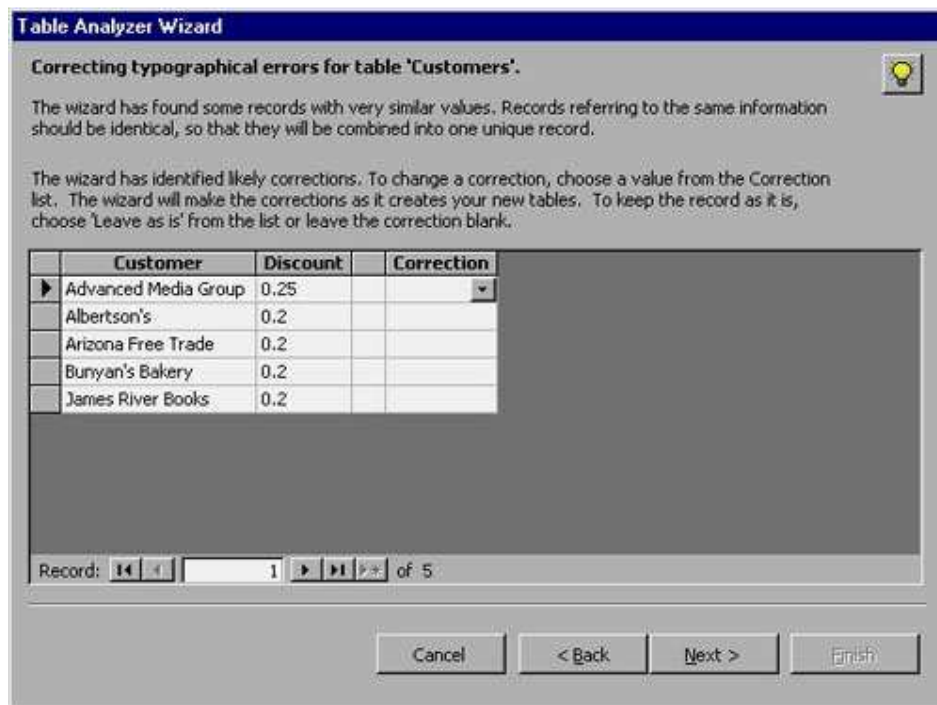
Hình 6: Thiết lập khóa chính

Hộp thoại này tương tự như hộp thoại đã sử dụng trong phần trước. Tuy nhiên, khác nhau ở các nút điều khiển phía trên cùng hộp thoại. Wizard sẽ hỏi để xác nhận mỗi bảng phải có một khóa chính. Những khóa chính cho các bảng Customers và Products được tạo bởi Wizard. Tuy nhiên, Wizard quên rằng trường ID sẽ là một khóa chính trong bảng Invoice Transactions.

Để xác định trường ID là một khóa, đầu tiên kích chọn tên trường. Sau đó kích chọn công cụ mà biểu thị chiếc khóa. Access đáp lại bằng cách đặt một khóa nhỏ ở phía bên trái trường ID trong bảng Invoice Transaction.

### Lỗi Correcting Typographical

Khi kích chuột vào nút Next, Wizard sẽ phân tích dữ liệu trong bảng được gợi ý và hiển thị những gì có thể được cho là lỗi. Access thực hiện điều này với mỗi bảng mới được tạo bởi Wizard. Vì hai bảng mới (Customerds và Products) được tạo bởi Wizard trong ví dụ này, điều này có nghĩa Access hiển thị hai hộp thoại typographical correction. Hộp thoại đầu tiên cho bảng Customer như hình sau:

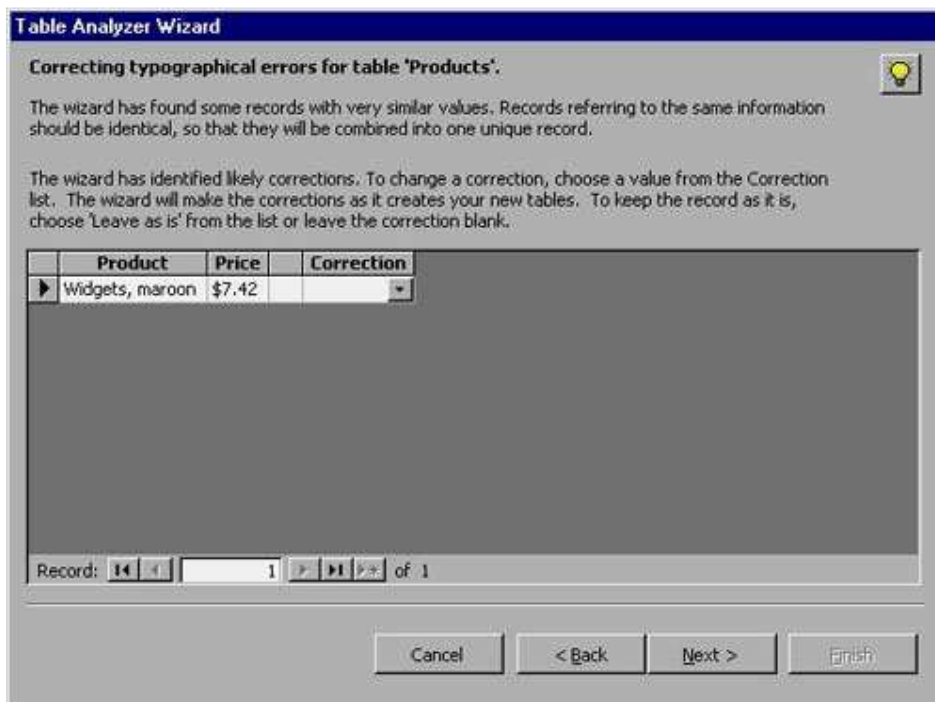


Hình 7: Lỗi Correcting typographical đối với bảng Customers

Đây là hộp thoại đã hiển thị 5 tên khách hàng mà cho rằng có thể bị lỗi. Nó đưa ra kết quả này bằng cách sử dụng một thuật giải bên trong. Nếu chính tả của dữ liệu đúng với dữ liệu trong bảng thì nó được cho là lỗi tiềm ẩn. Tất cả bạn cần làm là sử dụng danh sách kéo xuống trong cột Correction để chọn chính tả đúng hoặc thiết lập (Leave As Is).

**Chú ý:** Nếu bạn đang phân tích một bảng lớn mà đã làm việc với nhiều người trong một thời gian dài, rất có thể bạn sẽ gặp nhiều lỗi trong tài liệu. Phải chắc chắn bạn dành thời gian để sửa những lỗi cần thiết trong bước này; nó sẽ dễ hơn là kiểm tra lỗi sau đó.

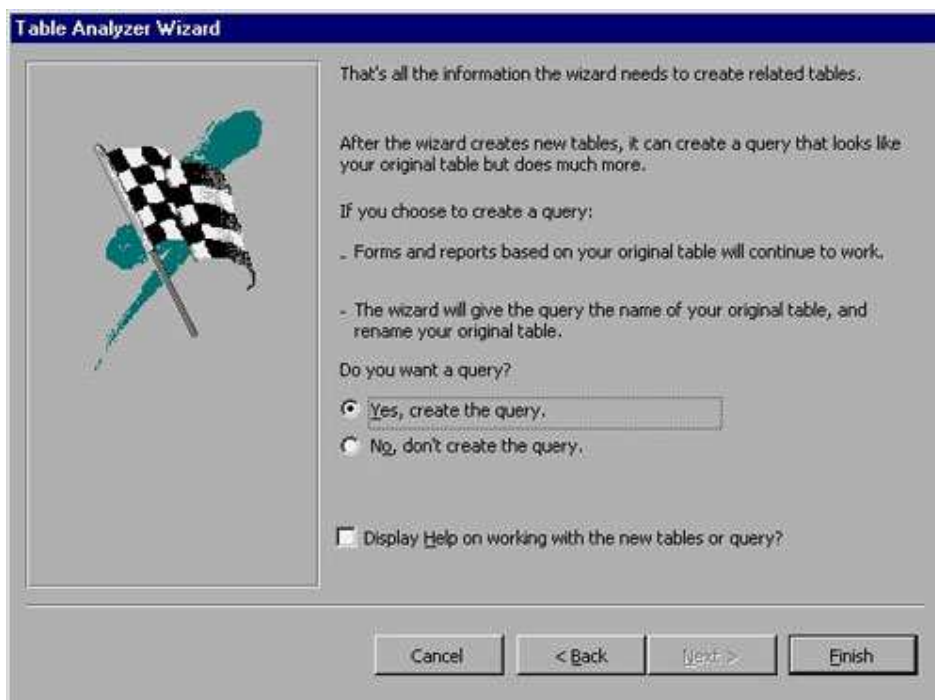
Trong ví dụ bảng Customers này, không có lỗi chính tả. Bạn nên thay đổi cột Corrections bằng tất cả các bản ghi (Leave As Is) mà thiết lập được chọn. Khi hoàn thành, kích chuột vào nút Next. Wizard sau đó hiển thị bảng Products như hình 8.



Hình 8: Lỗi Correcting typographical đối với bảng Customers.

Nó chỉ xảy ra như vậy khi một bản ghi được xác định là có thể bị lỗi. Tất nhiên, trong bảng riêng của bạn thì có thể có lỗi. Bạn nên xuyên suốt quá trình sửa lỗi thật chính xác như làm với bảng Customers. Trong trường hợp này, bạn chọn tùy chọn Leave As Is cho cột Correction. Kết thúc việc phân tích.

Khi kích chuột vào nút Next, Wizard hiển thị bước cuối cùng như hình sau



Hình 9: Bước phân tích cuối cùng.

Bước cuối cùng trong Table Analysis Wizard là nơi bạn xác định nếu muốn tạo một truy vấn trong như bảng gốc. Nếu không muốn tạo truy vấn bạn có thể tích chọn No, don't create the query và kích nút Finish để kết thúc quá trình phân tích bảng. Trong trường hợp bạn muốn tạo truy vấn, nó có nghĩa rằng Wizard tạo một truy vấn là sự pha trộn của 3 bảng, được kết hợp như bảng gốc.

Kích chọn Yes, create the query và kích nút Finish để hoàn thành quá trình phân tích bảng. Access hiển thị truy vấn mới lên màn hình như hình 10.

ID	Company	Customer	Discount	Invoice#	Date	Shipping Products	Product	Price	Units
1	Kilburn Manufacturing, Inc.	Kilburn Manufacturing	15%	1241	1/25/99	Widgets, plastic, 8.50	Widgets, green	16.47	144
2	Western Consolidated, Inc.	Western Consolidated	0%	2006	2/26/99	Widgets, plastic, 7.50	Widgets, purple	47.31	12
3	Bethany Copies, Inc.	Bethany Copies	0%	2752	3/29/99	Widgets, plastic, 7.50	Widgets, purple	47.31	52
4	Bethany Copies, Inc.	Bethany Copies	0%	22613	4/26/99	Widgets, plastic, 7.50	Widgets, purple	47.31	44
5	Country Touch Decals, Inc.	Country Touch Decals	0%	2518	5/26/99	Widgets, black, 7.50	Widgets, black	17.4	66
6	Western Consolidated, Inc.	Western Consolidated	0%	22017	6/26/99	Widgets, black, 7.50	Widgets, black	47.31	26
7	Kilburn Manufacturing, Inc.	Kilburn Manufacturing	15%	27475	7/26/99	Widgets, black, 7.50	Widgets, black	47.31	16
8	Country Touch Decals, Inc.	Country Touch Decals	0%	27475	7/26/99	Widgets, black, 7.50	Widgets, black	47.31	16
9	Country Touch Decals, Inc.	Country Touch Decals	0%	23007	7/26/99	Widgets, green, 6.97	Widgets, green	16.37	20
10	Vale Central Systems, Inc.	Vale Central Systems	0%	26667	7/13/99	Widgets, black, 3.64	Widgets, black	16.34	18
11	Kilburn Manufacturing, Inc.	Kilburn Manufacturing	15%	2434	7/26/99	Widgets, plastic, 8.50	Widgets, green	16.47	21
12	Ving via Western Shipping, Inc.	Ving via Western Shipping	0%	3122	8/26/99	Widgets, black, 7.50	Widgets, black	47.31	30
13	Hudson Computers, Inc.	Hudson Computers	0%	5122	8/26/99	Widgets, black, 7.50	Widgets, black	47.31	26
14	Bethany Copies, Inc.	Bethany Copies	0%	5951	9/26/99	Widgets, plastic, 7.50	Widgets, purple	47.31	20
15	United Electric, Inc.	United Electric	0%	5112	9/26/99	Widgets, black, 3.64	Widgets, black	16.34	17
16	Vale Central Systems, Inc.	Vale Central Systems	0%	5067	9/26/99	Widgets, green, 6.97	Widgets, green	16.37	21
17	Kilburn Manufacturing, Inc.	Kilburn Manufacturing	15%	5951	9/26/99	Widgets, plastic, 7.50	Widgets, purple	47.31	20
18	Kilburn Manufacturing, Inc.	Kilburn Manufacturing	15%	5951	9/26/99	Widgets, plastic, 8.50	Widgets, green	16.47	144
19	Kilburn Manufacturing, Inc.	Kilburn Manufacturing	0%	5951	9/26/99	Widgets, black, 7.50	Widgets, black	47.31	42
20	Kilburn Manufacturing, Inc.	Kilburn Manufacturing	0%	5951	9/26/99	Widgets, black, 7.50	Widgets, black	47.31	30
21	Kilburn Manufacturing, Inc.	Kilburn Manufacturing	15%	5413	9/26/99	Widgets, plastic, 8.50	Widgets, green	16.47	144

Hình 10: Truy vấn mới được tạo bởi Table Analysis Wizard.

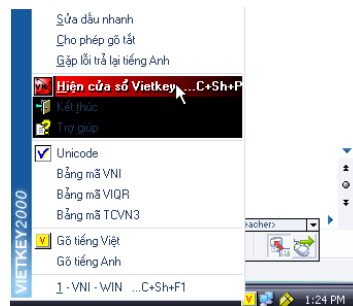


Bài giảng thực hành

MS. Access

**I. Cài đặt Vietkey với bảng mã Unicode:**

- B1:** Trỏ **Mouse Pointer** vào biểu tượng **Vietkey** trên **Taskbar** (**Right Click**)
- B2:** Chọn mục “**Hiện cửa sổ Vietkey**”
- B3:** Chọn mục “**Kiểu gõ**”
- B4:** Chọn kiểu gõ **TELEX** (hoặc **VNI**)
- B5:** Chọn “**Bảng mã**”
- B6:** Chọn “**Unicode dạng sẵn**”
- B7:** Click vào mục **Taskbar** (để trả **Vietkey** về **Taskbar**)



**II. Cài đặt cấu hình cho Windows**

- B1:** Start \ **Settings** \ **Control Panel**
- B2:** Chọn **Regional and Language Options**
- B3:** Chọn thẻ **Regional Options** - Chọn **Customize**
- B4:** Chọn thẻ **Numbers**:

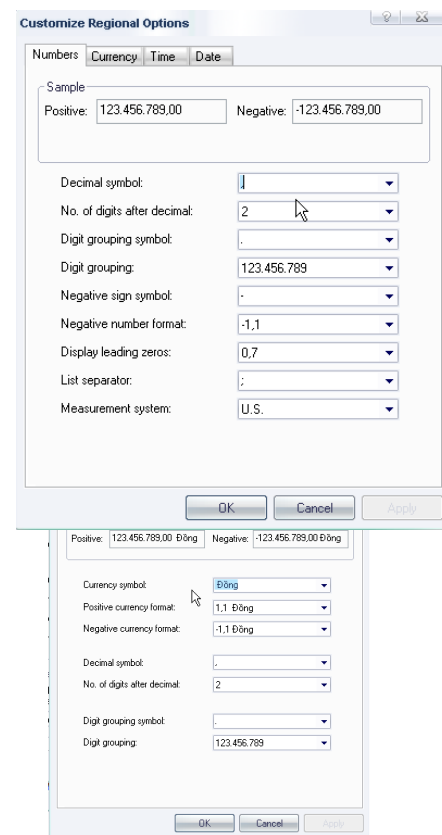
- Chọn **Decimal symbol** (gõ vào dấu ,)
- Chọn **Digit grouping symbol** (gõ vào dấu .)
- List separator** (gõ vào dấu ;)
- Apply + OK**

- B5:** Chọn thẻ **Currency**:

- Currency symbol** (gõ vào chữ “đồng”)
- Decimal symbol** (gõ vào dấu ,)
- Digit grouping symbol** (gõ vào dấu .)
- Apply + OK**

- B6:** Chọn thẻ **Date**:

- Gõ vào dd/mm/yyyy (ở mục **Short date**)
- Apply + OK**





**B7:** Click vào nút **Close Button** để đóng tất cả cửa sổ, trở về **Windows**

### III. Tạo thư mục lưu trữ File Access:

**B1:** Trở **Mouse Pointer** vào nút **Start (Right Click)**

**B2:** Chọn **Explore**

**B3:** Click chọn tên ổ đĩa (ở đây ta chọn ổ đĩa D)

**B4:** Trở **Mouse Pointer** vào vùng trống trên **Right Panel (Right Click)**

Chọn **New**

Chọn **Folder**

Gõ vào tên thư mục (trong HĐH **Windows**, tên thư mục có thể có khoảng trắng)

**Enter**

☞ Nếu gõ sai tên thư mục:

Trở **Mouse Pointer** vào tên thư mục (**Right Click**)

Gõ lại tên thư mục

**Enter**

### IV. Tạo File Access:

**B1:** **Start \ Programs \ Microsoft Office \ Microsoft Access**

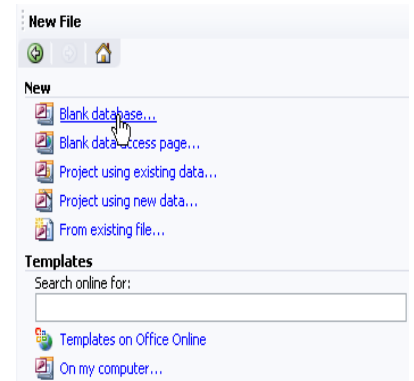
**B2:** Chọn **Create a new file** (tạo **File Access** mới)  
(Nếu chọn **Open** là mở **File Access** đã có trên đĩa)

**B3:** Chọn **Blank Database**

Chọn ổ đĩa và thư mục để lưu trữ **File Access**

Đặt tên cho **File Access**

Chọn **Create**



### V. Cách Import đối tượng của Access từ File Access khác

**Import** trong **Access** có nghĩa là **Copy**

**B1:** Chọn **Tables** (ở mục **Objects**)

**B2:** Trở **Mouse Pointer** vào chỗ trống trong **Access (Right Click)**

Chọn **Import**

Chọn thư mục lưu trữ **File Access** dữ liệu để **Import**

Click vào **File Access** muốn chọn - Chọn **Import**

**B3:** Chọn thẻ **Tables** (sẽ thấy các **Tables** có sẵn trong **File Access** mẫu)

Chọn **Select All - OK**

### VI. Đặt chú thích cho các Tables

**B1:** Chọn **Details**

**B2:** Trở **Mouse Pointer** vào **Table (Right Click)**

Chọn **Properties**

Gõ nội dung chú thích trong khung **Description**

Apply

(Tuần tự như trên, đặt chú thích cho tất cả các Tables)

### VII. Cách tạo bảng (Table) trong Access:

B1: Chọn Tables – New – Design View – Ok

B2: Mô tả cấu trúc.

**Field Name** : Tên vùng tin (dùng chữ hoa đầu, không có khoảng trắng)

**Data Type** : Kiểu dữ liệu của vùng tin (Text, Number, Yes/No,Date/Times...)

**Description** : Chú thích nội dung vùng tin

**Field Properties** (Thuộc tính của vùng tin):

- **Field Size** : Kích thước vùng tin
- **Format** : Định dạng vùng tin
- **Input Mask** : Khuôn dạng nhập dữ liệu
- **Caption** : Phần hiển thị trên tiêu đề của Datasheet
- **Validation Rule**: Biểu thức kiểm tra tính hợp lệ
- **Validation Text** : Câu thông báo khi nhập dữ liệu không hợp lệ
- **Indexed**: None – Yes(Duplicvates OK) – Yes (No Duplicates)

(Hướng dẫn tới 11/03/2006)

### VIII. Định khuôn dạng nhập liệu kiểu ngày

B1: Click vào Table - Chọn Design (mở ra ở chế độ thiết kế)

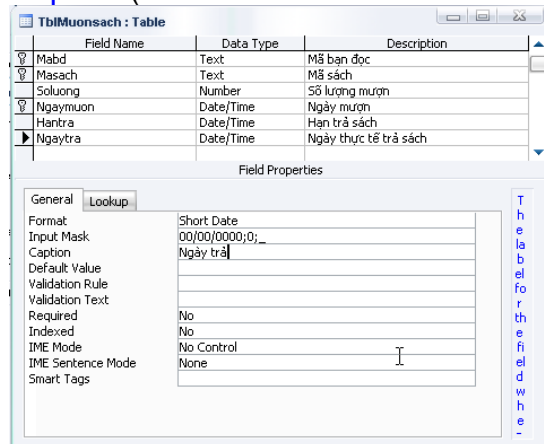
B2: Click chọn Field kiểu Date/Time

B3: Mục Field Properties:

**Format**: chọn Short Date

**Input Mask** (khuôn dạng nhập liệu): gõ vào 00/00/0000;0;\_

**Caption** (chú thích tên Field trên Datasheet)



### IX. Cách Lookup dữ liệu từ Table khác:

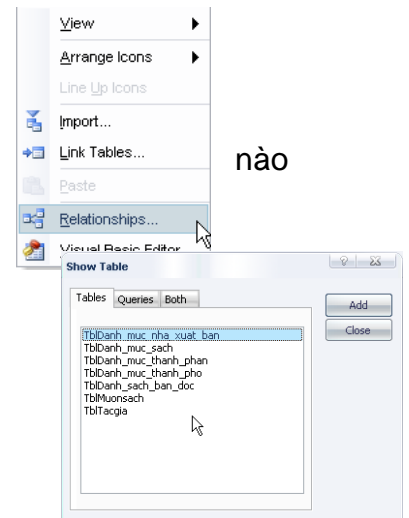
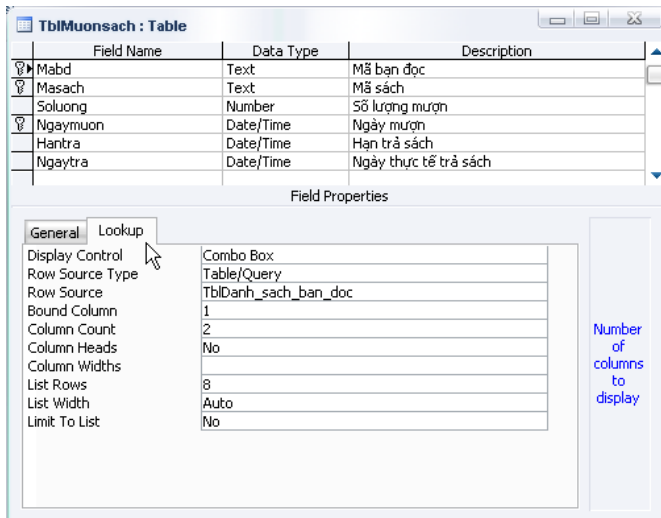
B1: Click chọn Field muốn tạo Lookup

B2: Chọn Lookup (ở mục Field Properties)

B3: Chọn Combo box (ở mục Display Control)

B4: Chọn tên Table chứa dữ liệu muốn đưa vào Combobox (mục Row Source)

B5: Column Count: chọn số Field muốn hiển thị trong Combobox



nào

**X. Tạo quan hệ (Relationship) giữa các Tables:**

**B1:** Đóng tất cả các Tables trước khi tạo quan hệ (nếu có Table đang mở => sẽ không tạo quan hệ được).

**B2:** Click chọn Table

Trở Mouse Pointer vào vị trí trống (Right Click)  
Chọn Relationship

**B3:** Access sẽ hiển thị cửa sổ Show Table

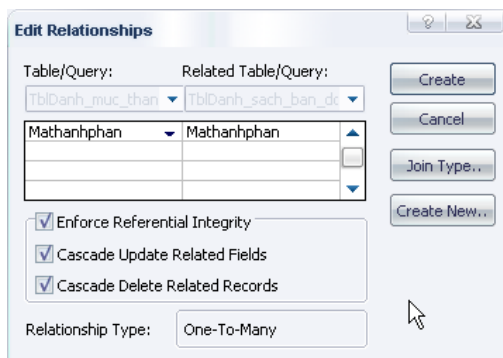
Giữ phím SHIFT và quét chọn (phím ↓) tất cả các Tables  
Chọn ADD (chỉ được nhấn nút ADD ĐÚNG MỘT LẦN)  
Chọn Close (đóng cửa sổ Show Table)

**B4:** Tạo quan hệ giữa các Tables

Chú ý: Kéo (Drag) từ khoá chính và thả (Drop) vào khoá ngoại.

Đánh dấu chọn (để đảm bảo toàn vẹn quy chiếu)

- ✓ Enforce Referential Integrity  
(Toàn vẹn quy chiếu giữa các Tables, nghĩa là khi bảng này thay đổi sẽ dẫn đến thay đổi tại các Tables có quan hệ)
- ✓ Cascade Update Related Fields  
(Khi cập nhật Parent Table => sẽ cập nhật Child Table)
- ✓ Cascade Delete Related Records  
(Khi xóa mẫu tin của Parent Table => sẽ xóa mẫu tin của Child Table)



Kế tiếp [Click](#) chọn nút [Create](#)

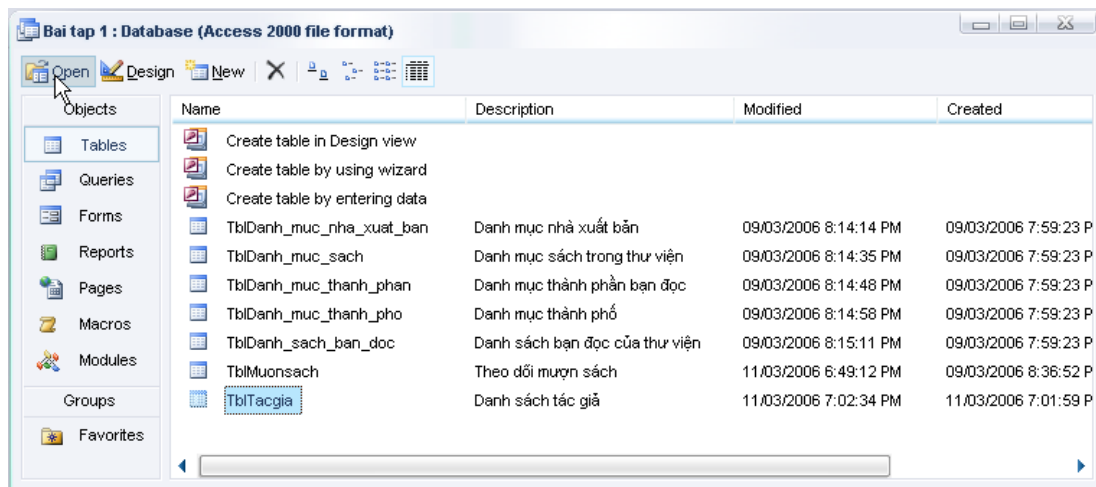
### XI.Cách nhập số liệu vào Table:

**B1:** [Click](#) chọn [Table](#) muốn nhập số liệu

**B2:** Chọn [Open](#)

**B3:** Nhập số liệu

**B4:** Kết thúc nhập số liệu – [Click](#) vào nút [Close](#) để đóng lại



# Form Builder

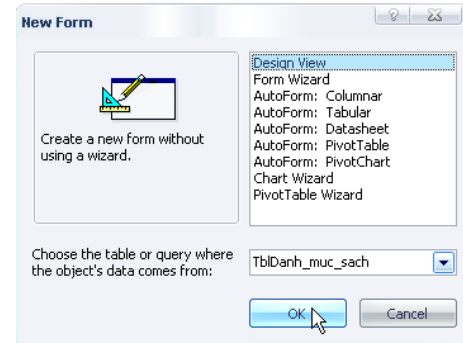
## Các dạng thiết kế Form

### 1.1. Dạng bình thường:

Bao gồm các thao tác sau:

**B1:** Chọn **Forms** (ở phần **Objects**)

- Chọn **New**
- Chọn **Table** sẽ đưa vào **Form**
- Chọn **Design View**
- **OK**



**B2:** Quét chọn hết các **Fields** trong **Table**

(Click chọn **Field** đầu tiên, kế tiếp dùng phím **Shift + ↓** để chọn

**B3:** **Drag and Drop** để kéo và thả các **Fields** vào **Form**

Sau khi thả các **Fields** vào **Form** => đóng **Table** lại

- Nếu muốn hiển thị **Table** thì Click vào Icon **Fields List**
- Muốn hiển thị **Toolbar**: chọn **View \ Toolbars \ Form Design**
- Muốn hiển thị lưới (**Grid**) thiết kế:

- Trở **Mouse Pointer** vào **Form** (**Right Click**)
- Chọn **Grid**

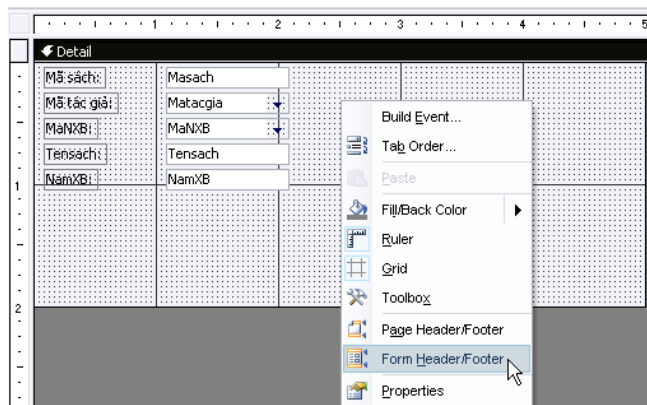
**B4:** Tạo **Form Header/Footer**

■ **Header** là phần tiêu đề của **Form**, chứa chuỗi giới thiệu chương trình.

■ **Footer** là phần tiêu đề cuối của **Form**, chứa các nút điều khiển

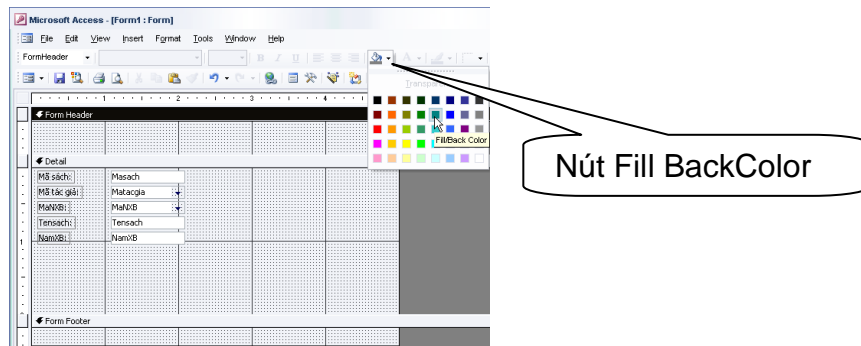
Trở **Mouse Pointer** vào **Form** (**Right Click**)

Chọn **Form Header/Footer**



Click chọn vào **Form Header** - Chọn màu (**Color**) cho **Form Header**

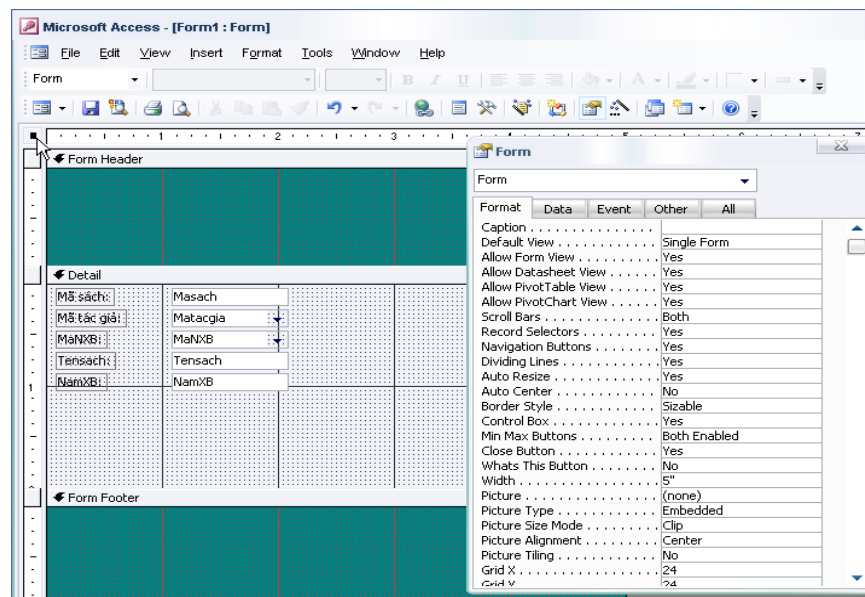
Click chọn vào **Form Footer** - Chọn màu cho **Form Footer**



Nếu không thấy **Toolbar** thì chọn: **View \ Toolbars \ Formatting (Form/Report)**

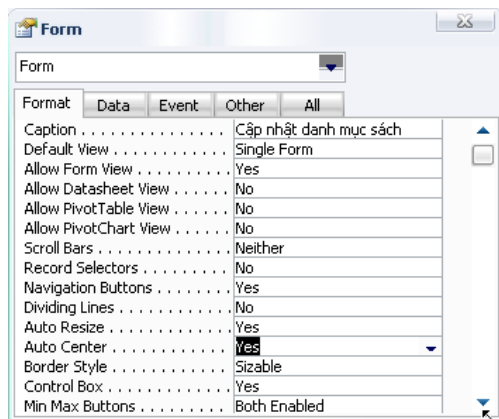
**B5:** Xác lập thuộc tính cho **Form**:

- Trở **Mouse Pointer** vào **Form (Right Click)**
- Chọn **Properties**
- Click** vào ô đánh dấu **Form**
- Chọn thẻ **Format**
- Thay đổi giá trị các thuộc tính sau:



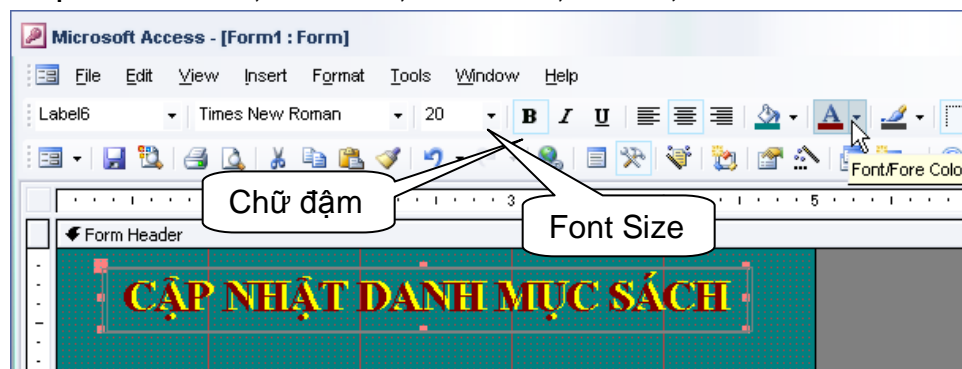
Các thuộc tính như sau:





**B6:** Trình bày tiêu đề trên **Form Header**

- Chọn **View \ Toolbox** (bật **Toolbox**, là thanh công cụ chứa các đối tượng thiết kế)
- Click chọn đối tượng **Label** (trên **Toolbox**)
- Vẽ **Label** trên **Form**
- Nhập nội dung cho **Label** (Nhập xong, Click ra ngoài)
- Click vào đường viền (**Border**) của **Label** để định dạng **Label**
- Chọn: **FontName**, **FontSize**, **FontColor**, **Center**, ...



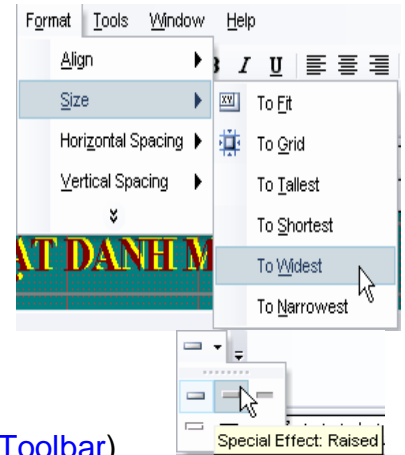
Cách tạo chữ nổi:

- Click vào đường viền của **Label**
- Nhấn CTRL + C (**Copy**)
- Click ra ngoài (ta sẽ không thấy các "ô neo" chung quanh **Label**)
- Nhấn CTRL + V (**Paste**)
- Click vào **Label** mới được tạo - chọn **Font Color**
- Dùng phím CTRL+→ để di chuyển cho 2 **Labels** so le với nhau

**B7:** Điều chỉnh phần **Details**

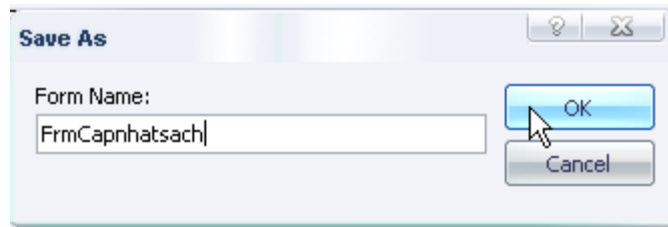
- Đánh dấu các đối tượng **Label**, **Textbox** trong phần **Detail**
- Giãn khoảng cách theo chiều dọc (**Vertical**):  
**Format \ Vertical Spacing \ Increase**
- Dùng phím Shift + (↓↑) để tăng kích thước **Label**, **Textbox**
- Chọn **Font Size**: 12
- Di chuyển các **Textbox**, **Combobox**,.....:

- ✓ Bật **Properties Table**
  - ✓ Đánh dấu các **Textbox, Combobox, ...** (không đánh dấu các **Labels**)
  - ✓ Gõ vào trị số mới cho thuộc tính **Left** (khoảng cách từ lề trái đến các **Textbox, Combobox, ...**)
  - ✓ Điều chỉnh kích thước chiều ngang các **Textbox, Combobox, ...**
  - ✓ Điều chỉnh kích thước các **Labels**:
    - ❖ Điều chỉnh một **Label** trước để làm chuẩn
    - ❖ Đánh dấu toàn bộ các **Label**
    - ❖ **Format \ Size \**
      - To Fit** (vừa đúng kích thước)
      - To Grid** (theo kích thước khung lưới)
      - To Tallest** (theo Label cao nhất)
      - To Shortest** (theo Label thấp nhất)
      - To Widest** (theo Label rộng nhất)
      - To Narrowest** (theo Label hẹp nhất)
- Có thể tạo nền cho các **Label** bằng cách:
- ❖ Đánh dấu tất cả **Labels**
  - ❖ Chọn **Raised** (ở nút **Special Effect** trên **Toolbar**)
  - ❖ Chọn màu nền **Fill BackColor** và màu chữ **Font ForeColor**



**B8:** Click vào nút **Save** để lưu lại

Gõ vào tên **Form** (dùng **Prefix Frm...**)



Đặt tên cho phần Description của Form

## 1.2. Tạo nút điều khiển trên Form

### 1.2.1. Các nút di chuyển mẫu tin (Navigation Buttons):

**B1:** Mở **Form** ở chế độ **Design**

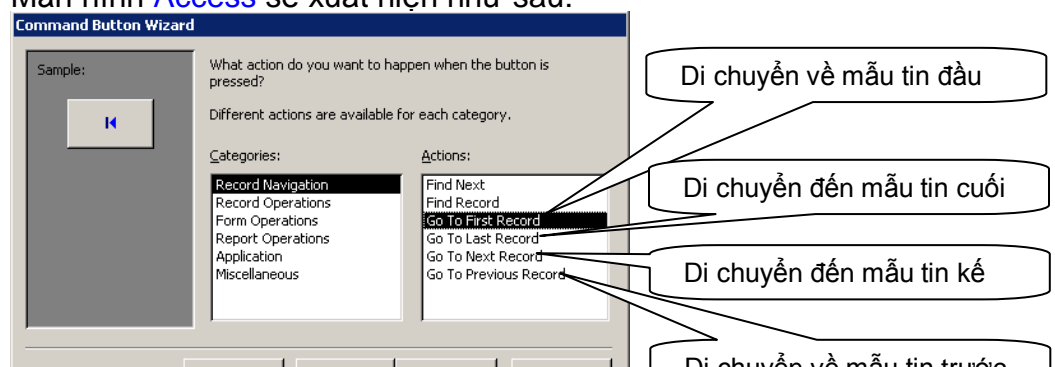
- Click chọn **Form** muốn tạo nút
- Chọn **Design – Maximize Form** để dễ thiết kế

**B2:** Bật **Toolbox** (**View \ Toolbox**)

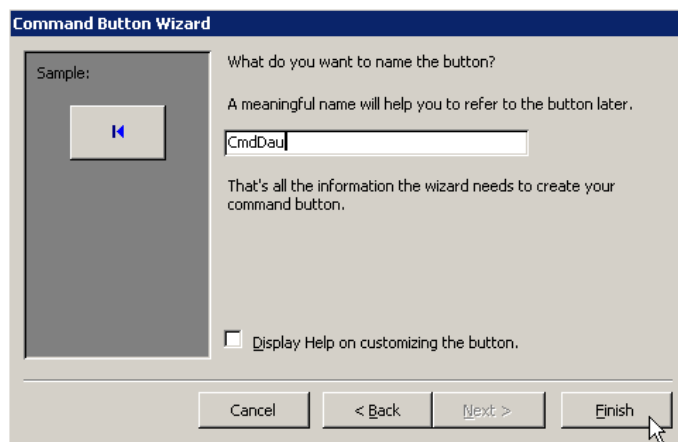
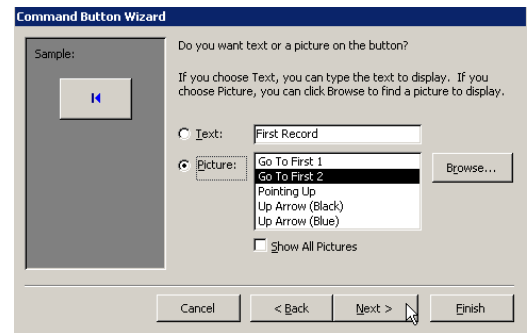
**B3:** Các bước đưa nút vào **Form**; nguyên tắc khi đưa vào **Form**, các nút phải ở phần **Form Footer**

- Click chọn nút **Control Wizards**
- Click chọn đối tượng **Command Button** (trên **Toolbox**)
- Vẽ nút vào **Form Footer**

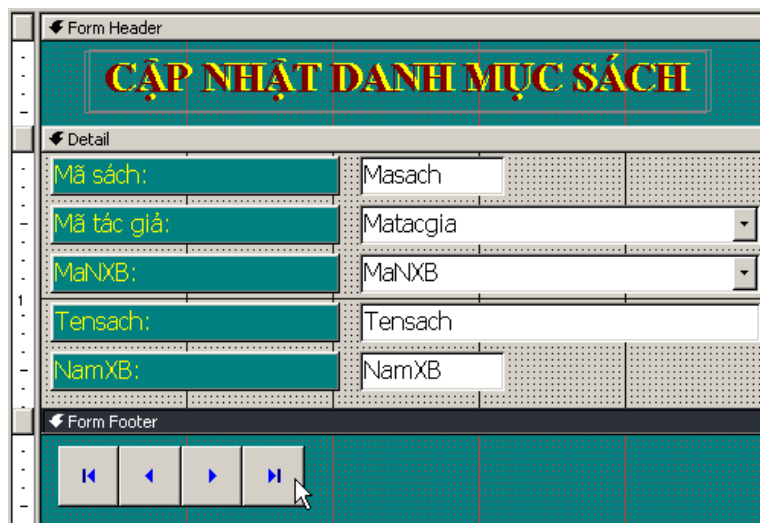
Màn hình **Access** sẽ xuất hiện như sau:



- Click chọn mục **Record Navigation** (trong khung **Categories**)
  - Click chọn **Go To First Record** (trong khung **Actions**)
  - Kế tiếp chọn **Next**
  - Nếu: chọn **Text** (tên nút sẽ có chữ)  
chọn **Picture** (nút sẽ có hình)
  - Kế tiếp chọn **Next**
- Màn hình sẽ xuất hiện như sau:



- Đặt tên cho nút (**CmdDau**, **CmdTruoc**, **CmdKe**, **CmdCuoi**)
- Click **Finish** để kết thúc.



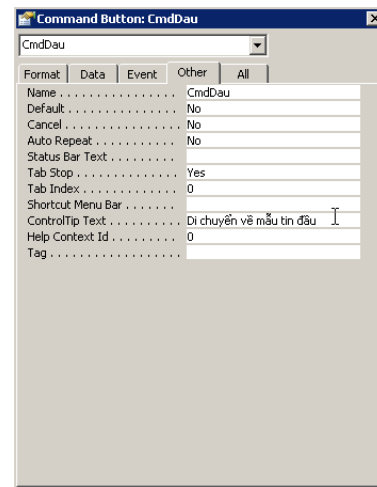
Chú ý: Lệnh **Format \ Align** dùng để sắp xếp vị trí  
Lệnh **Format \ Size** dùng để điều chỉnh kích thước

Sửa thuộc tính của nút:

- ✓ Bật **Properties Table**
- ✓ Click vào nút muốn sửa
- ✓ Click chọn thẻ **Other**
- ✓ Sửa lại thuộc tính **ControlTip Text**

Kết thúc việc thiết kế **Form**:

- ✓ Đóng **Form**
- ✓ Click vào **Form** - Chọn **Open** để thi hành



### 1.2.2. Tạo các nút xử lý mẫu tin

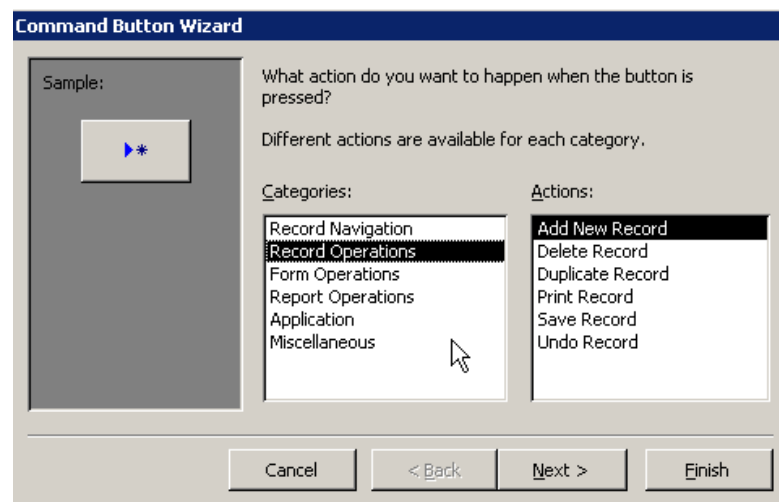
**B1..B3:** Giống như mục a)

Khi hiện bảng **Command Button Wizard**:

Chọn **Record Operations** (trong khung **Categories**)

Tuần tự chọn các nút sau (cho từng lần tạo nút):

- ✓ **Add New Record** (thêm mẫu tin mới - **CmdThem**)
- ✓ **Delete Record** (xoá mẫu tin hiện hành - **CmdXoa**)
- ✓ **Save Record** (lưu mẫu tin hiện hành - **CmdLuu**)
- ✓ **Undo Record** (huỷ bỏ thông tin đã nhập - **CmdUndo**)
- ✓ **Print Record** (in mẫu tin hiện hành - **CmdIn**)



### 1.3. Tạo nút thoát trên **Form**:

Công dụng: tạo nút thoát với hàm **MsgBox()**

Thao tác:

**B1:** Nếu có nút thoát rồi thì mở ra để thêm hàm **MsgBox()**

Nếu chưa có nút thoát thì tạo nút mới và viết hàm **MsgBox()**

**B2:** Bật **Properties Table**

Click chọn nút thoát

Chọn thẻ **Events**

Chọn biến cố (**Event**) **On Click**

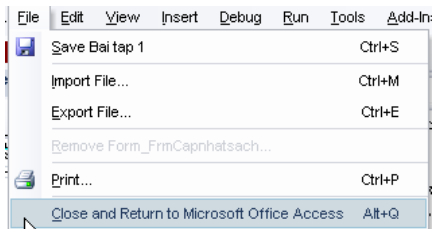
Chọn Code Builder

**B3:** Tìm đến **Code** của nút thoát (**CmdDong**) và bổ sung **Code**

```
Private Sub CmdDong_Click()  
    On Error GoTo Err_CmdDong_Click  
  
    If MsgBox("Muon thoat khoi chuong trinh?", vbOKCancel, "Quan ly danh muc sach") = vbOK Then  
        DoCmd.Close  
    End If  
Exit_CmdDong_Click:  
    Exit Sub  
  
Err_CmdDong_Click:  
    MsgBox Err.Description  
    Resume Exit_CmdDong_Click  
  
End Sub
```

**Chú ý:** dùng phím **Spacebar** để chọn khi viết hàm **MsgBox()**

**B4:** Chọn **File \ Close and Return to Microsoft Office Access** để thoát về **Access**



### 1.4. Tạo Listbox bằng chương trình Wizard

**B1:** Mở **Form** ở chế độ **Design**

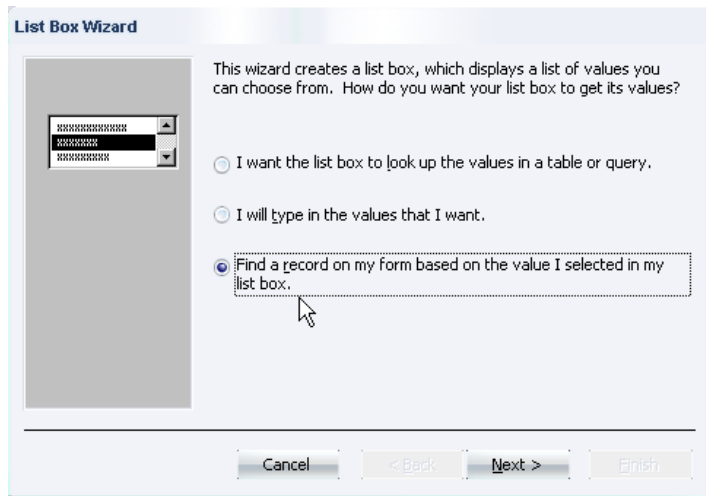
Bật **Toolbox**

Click bật nút **Control Wizards**

Click chọn đối tượng **Listbox** (trong **Toolbox**)

Vẽ **Listbox** trong **Form**

**B2:** Màn hình sẽ xuất hiện như sau:



Chọn **Find a record on my form based on the value I selected in my listbox**

Kế tiếp chọn **Next**

**B3:** Chọn tên **Field** đưa vào **Listbox** (**Field** được chọn sẽ nằm trong khung **Selected Fields**)

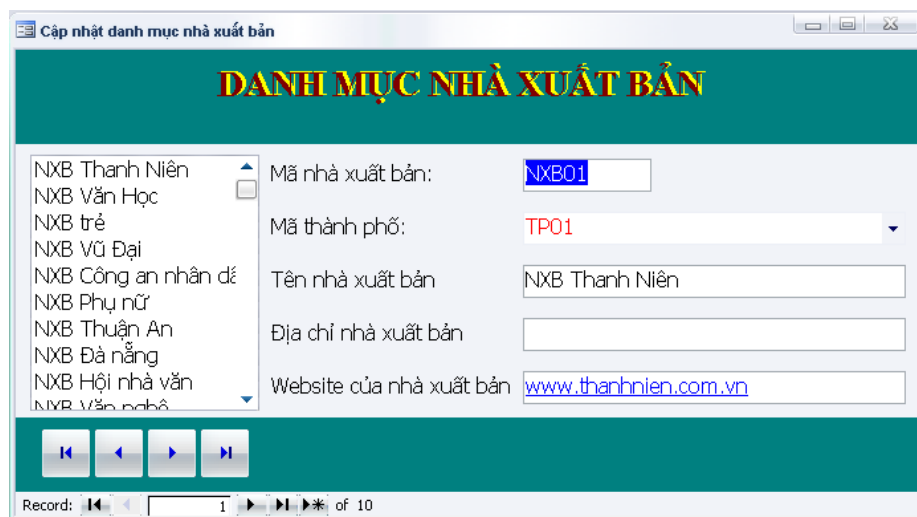
Kế tiếp chọn **Next – Next**



**B4:** Đặt tên cho **Listbox**  
 Chọn **Finish** để kết thúc

**B5:** Sửa lại **Listbox** trên **Form**:  
 Xoá **Label** của **Listbox**  
 Chọn **FontSize**

Kết quả:

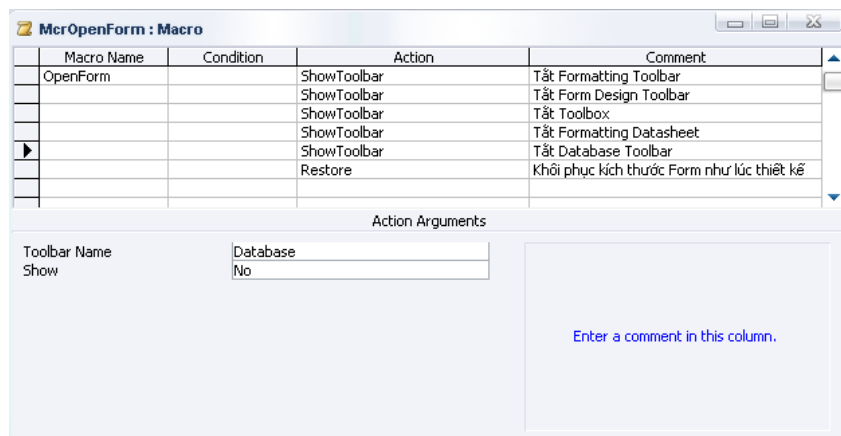


# Macro Builder

## Các dạng Macro

### 1.5. Tạo Macro để mở Form

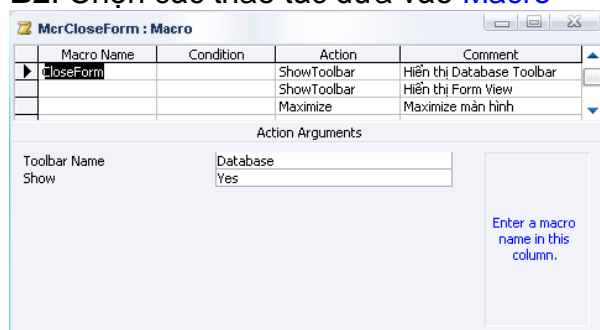
- Khi mở Form, chương trình phải đóng tất cả các thanh công cụ (Toolbars)
  - Giữ lại kích thước của Form như lúc thiết kế
- Tất cả những lệnh này sẽ được lưu vào Macro và gắn vào biến cố On Open của một Form
- Thao tác tạo Macro:  
B1: Chọn Macro – New  
B2: Chọn các lệnh đưa vào Macro



B3: Lưu Macro (McrOpenForm)

### 1.6. Tạo Macro dùng để đóng Form:

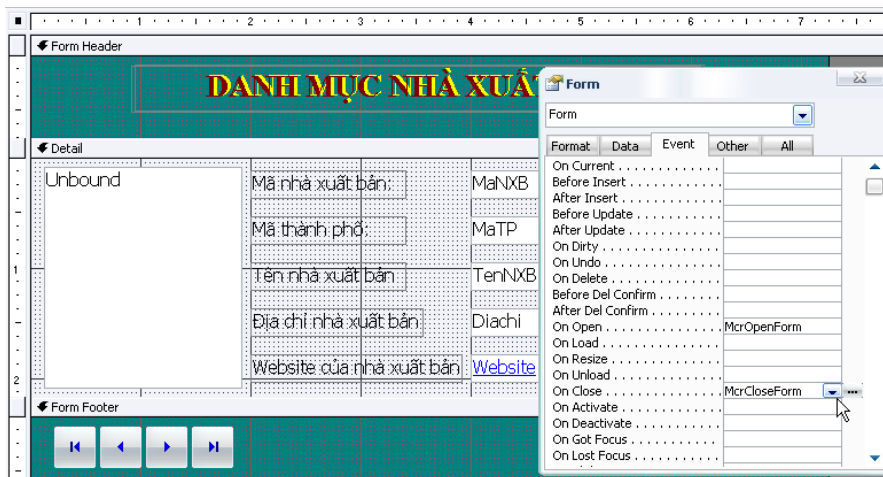
- Khi đóng Form, ta phải mở lại các thanh công cụ
  - Maximize màn hình.
- Tất cả các thao tác này được lưu vào Macro McrCloseForm
- Thao tác:  
B1: Chọn Macros – New  
B2: Chọn các thao tác đưa vào Macro



**B3:** Lưu trữ Macro

**GẮN MACROS VÀO FORM**

- B1:** Mở Form ở chế độ Design  
 Bật Properties Table - Chọn Events  
 Click chọn FormBar
- B2:** Chọn biến cố (Event) On Open  
 Gắn Macro McrOpenForm vào Event này
- B3:** Chọn Event On Close  
 Gắn Macro McrCloseForm vào Event này
- B4:** Đóng Form – Lưu trữ lại





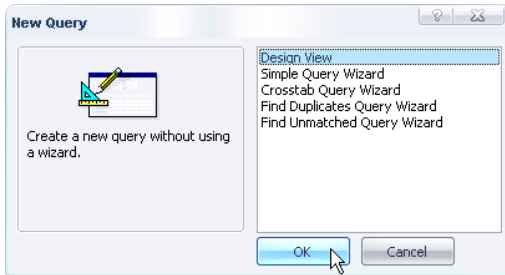
# Truy vấn dữ liệu (SQL)

## I. CÁCH TẠO QUERY TRONG ACCESS

### 1. Dạng 1:

B1: Chọn **Queries** – **New**

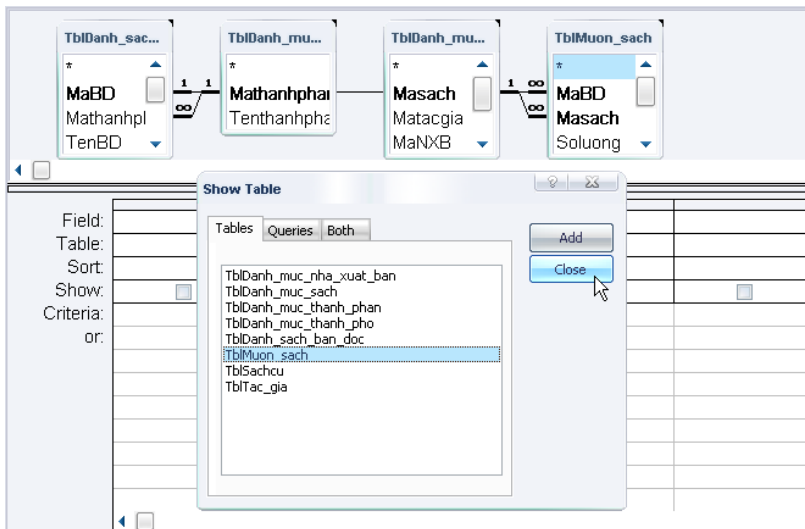
Chọn **Design View** - **OK**



B2: Click chọn từng **Table(s)** muốn đưa vào **Query** – **Add**

Kết thúc việc chọn: nhấn **Close**

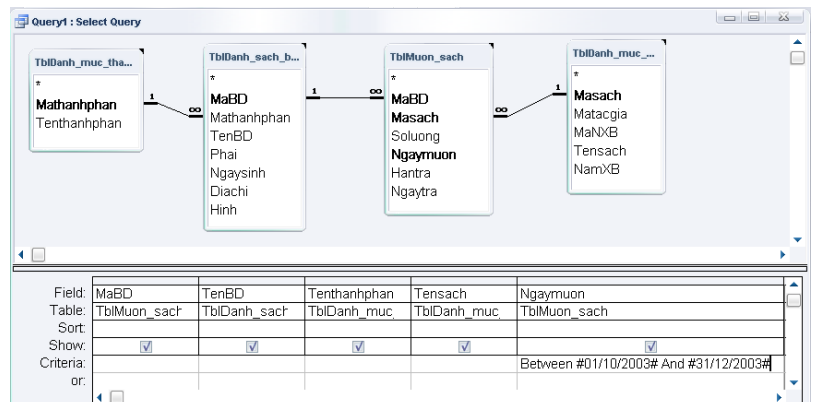
Chú ý: khi chọn **Table(s)** đưa vào **Query**, ta **PHẢI ĐỌC CÂU HỎI** để biết thông tin cần lấy được cung cấp bởi **Table(s)** nào?



B3: Trình bày lại các **Tables** trên màn hình thiết kế (để dễ quan sát thông tin cần truy xuất)

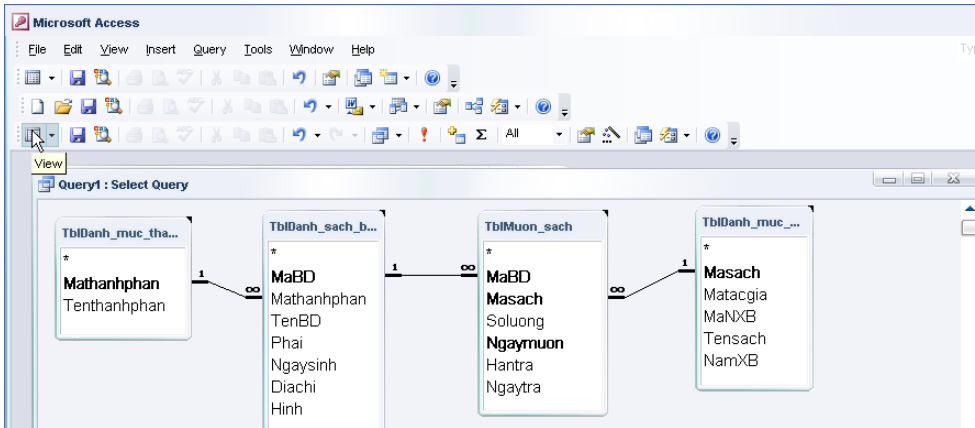
### Giải thích:

- ✓ **Field:** vùng thông tin muốn truy xuất
- ✓ **Table:** tên bảng dữ liệu cung cấp thông tin tương ứng



- ✓ **Sort:** sắp xếp dữ liệu (trị mặc nhiên là không – None)
- ✓ **Show:** hiển thị hoặc che dấu thông tin
- ✓ **Criteria:** đưa vào điều kiện trích lọc thông tin.

**B4:** Click vào nút **View** (trên **Query Design Toolbar**) để xem kết quả, nếu không thấy **Query Design Toolbar** thì chọn **View \ Toolbars \ Query Design Toolbar** để hiển thị **Toolbar** này.



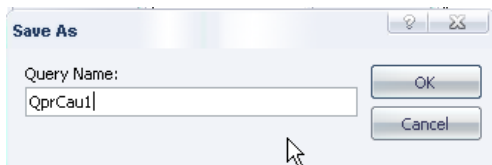
Kết quả truy vấn dữ liệu như sau:

Mã ban đọc	Tên ban đọc	Tên thành phần	Tensach	Ngày mượn sách
BD0001	Nguyễn Thị met	Nội trợ	36 cách cai rượu	01/12/2003
BD0001	Nguyễn Thị met	Nội trợ	36 món khai vị	07/12/2003
BD0001	Nguyễn Thị met	Nội trợ	Access Basic	20/12/2003
BD0002	Trần Anh Thư	Nội trợ	Một đời lưu lạc	01/10/2003
BD0002	Trần Anh Thư	Nội trợ	Trắng sáng vườn chuối	15/11/2003
BD0003	Lê Văn Thắng	Học sinh	Tin học căn bản	05/11/2003
BD0003	Lê Văn Thắng	Học sinh	Phân tích thiết kế hệ thống	09/12/2003
BD0005	Huỳnh Văn Thông	Học sinh	Trắng sáng vườn chuối	05/10/2003
BD0006	Phạm Ngọc Trâm Anh	Học sinh	Toán cao cấp	10/12/2003
BD0012	Nguyễn Thị Thu Thanh	Giáo viên	36 món khai vị	10/10/2003
BD0012	Nguyễn Thị Thu Thanh	Giáo viên	Lý thuyết đồ thị	01/12/2003
BD0012	Nguyễn Thị Thu Thanh	Giáo viên	Access Basic	25/11/2003
BD0012	Nguyễn Thị Thu Thanh	Giáo viên	Trắng sáng vườn chuối	12/10/2003
BD0013	Trần Thị Minh Thư	Giáo viên	Vật lý đại cương	15/10/2003
BD0013	Trần Thị Minh Thư	Giáo viên	Access Basic	20/10/2003
BD0013	Trần Thị Minh Thư	Giáo viên	Phân tích thiết kế hệ thống	05/11/2003
BD0014	Nguyễn Văn Tuy	Giáo viên	36 cách cai rượu	05/12/2003
BD0014	Nguyễn Văn Tuy	Giáo viên	Hệ cơ sở dữ liệu	15/12/2003
BD0016	Hà Thanh Phương	Công nhân viên	Nghệ thuật nấu cháo hành	30/10/2003
BD0016	Hà Thanh Phương	Công nhân viên	36 món khai vị	20/10/2003
BD0016	Hà Thanh Phương	Công nhân viên	Trắng sáng vườn chuối	05/11/2003

Click vào nút **View** để trở lại màn hình thiết kế **Query**

**B5:** Click chọn nút **Save** để lưu **Query**

Đặt tên cho **Query** (Qpr...)

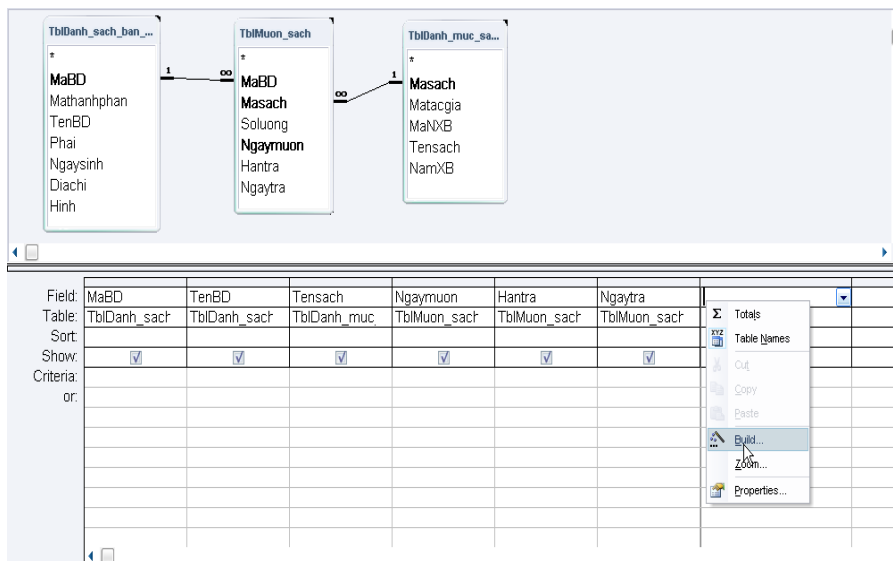


**2. Dạng 2:** Thêm **Field** mới chứa biểu thức tính toán trong **Query**

B1 ...B4: thực hiện như **Query** dạng 1

B5: Trở **Mouse Pointer** vào **Field** mới muốn tạo (**Right Click**)

Chọn **Build**



**B6:** Xây dựng biểu thức tính toán

- ✓ Click vào dấu “+” để mở rộng danh sách Tables
  - ✓ Click vào Table muốn chọn (các Fields tương ứng sẽ hiển thị ở khung bên cạnh)
  - ✓ Double Click vào Field muốn chọn (Field được chọn sẽ được đưa lên khung xây dựng biểu thức)
  - ✓ Click chọn dấu phép toán +, -, x, /, .....v.v...
- (Tiếp tục như trên cho đến khi xây dựng xong biểu thức)

- ✓ OK đóng cửa sổ xây dựng biểu thức

**B7:** Xoá chữ “Expr...” và gõ vào tên Field mới.

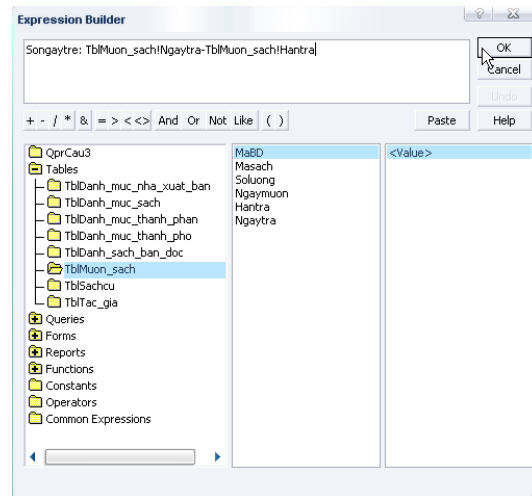
Tên Field:

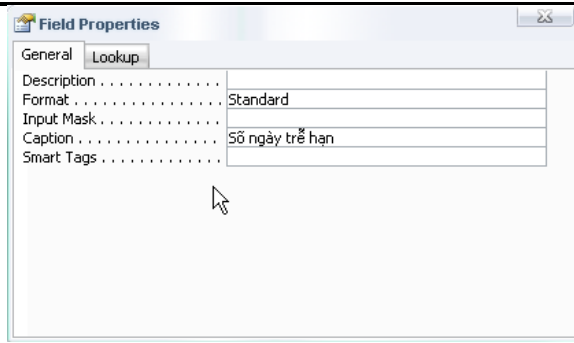
- ✓ Không có khoảng trắng (Blank)
- ✓ Không có dấu tiếng Việt

**B8:** Trỏ Mouse Pointer vào Field mới tạo (Right Click)

Chọn Properties

Sửa thuộc tính: Format , Caption, ...







**Giáo trình**

# Microsoft Access

## MỤC LỤC

<b>CHƯƠNG I. LÀM VIỆC VỚI MICROSOFT ACCESS</b> .....	<b>7</b>
I. VÀO MÔI TRƯỜNG ACCESS .....	7
II. CHƯƠNG TRÌNH ACCESS .....	8
<b>II.1. Tập chứa Chương trình Access</b> .....	<b>8</b>
<b>II.2. Một hệ Chương trình do Access tạo ra (hệ Chương trình Access)</b> .....	<b>8</b>
III. TẠO MỘT CSDL MỚI .....	9
IV. LÀM VIỆC VỚI CSDL ĐÃ TỒN TẠI .....	10
<b>CHƯƠNG II. BẢNG</b> .....	<b>13</b>
I. KHÁI NIỆM BẢNG .....	13
II. TẠO BẢNG .....	13
<b>II.1. Tạo bảng bằng Design View.1</b> .....	<b>14</b>
<b>II.2. Tạo bảng bằng Datasheet View :</b> .....	<b>15</b>
<b>II.3. Tạo bảng bằng Table Wizard</b> .....	<b>16</b>
<b>II.4. Tạo bảng bằng Import Table</b> .....	<b>16</b>
<b>II.5. Tạo bảng bằng Link Table</b> .....	<b>16</b>
III. THUỘC TÍNH CỦA TRƯỜNG .....	16
<b>III.1. Công dụng:</b> .....	<b>16</b>
<b>III.2. Cách đặt giá trị cho các thuộc tính.</b> .....	<b>17</b>
<b>III.3. Tổng quan về các thuộc tính của trường</b> .....	<b>17</b>
<b>III.4. Chi tiết về các thuộc tính của trường</b> .....	<b>17</b>
<i>III.4.1. Thuộc tính Field Size</i> .....	<i>17</i>
<i>III.4.2. Thuộc tính Format</i> .....	<i>18</i>
<i>III.4.3. Thuộc tính Input Mask (mặt nạ nhập liệu)</i> .....	<i>19</i>
<i>III.4.4. Thuộc tính Default Value</i> .....	<i>21</i>
<i>III.4.5. Thuộc tính Required (trường bắt buộc phải có số liệu)</i> .....	<i>21</i>
<i>III.4.6. Thuộc tính AllowZerolength</i> .....	<i>21</i>
<i>III.4.7. Thuộc tính ValidationRule</i> .....	<i>21</i>
<i>III.4.8. Thuộc tính Indexed</i> .....	<i>22</i>
IV. THAY ĐỔI THIẾT KẾ, CHỈNH SỬA CẤU TRÚC BẢNG .....	22
V. ĐẶT KHOÁ CHÍNH .....	22
<b>V.1. Định nghĩa</b> .....	<b>22</b>
<b>V.2. Lợi ích</b> .....	<b>23</b>
<b>V.3. Xử lý khoá chính</b> .....	<b>23</b>
VI. LƯU CẤU TRÚC BẢNG VÀ ĐẶT TÊN BẢNG .....	23
VII. THAY ĐỔI CẤU TRÚC BẢNG VÀ NHẬP DỮ LIỆU .....	24
<b>VII.1. Thay đổi cấu trúc bảng</b> .....	<b>24</b>
<b>VII.2. Nhập dữ liệu vào bảng</b> .....	<b>24</b>
<b>VII.3. Chuyển đổi giữa hai chế độ nhập dữ liệu và thiết kế bảng</b> .....	<b>25</b>
VIII. LẬP QUAN HỆ GIỮA CÁC BẢNG .....	25
<b>VIII.1. Các loại quan hệ trong Access</b> .....	<b>25</b>
<b>VIII.2. Cách tạo quan hệ</b> .....	<b>26</b>
<b>VIII.3. Chỉnh sửa quan hệ</b> .....	<b>28</b>
<b>VIII.4. Tính toàn vẹn tham chiếu</b> .....	<b>28</b>
<b>VIII.5. Xem và điều chỉnh các quan hệ đã có trong CSDL</b> .....	<b>30</b>
<i>VIII.5.1. Xem các quan hệ đã có</i> .....	<i>30</i>
<i>VIII.5.2. Chỉnh sửa</i> .....	<i>30</i>
<i>VIII.5.3. Xóa</i> .....	<i>30</i>
IX. THUỘC TÍNH CỦA BẢNG .....	30

IX.1. Các thuộc tính của bảng.....	30
IX.2. Đặt thuộc tính. ....	30
<b>CHƯƠNG III. TRUY VẤN (QUERY) .....</b>	<b>32</b>
I. CÁC LOẠI TRUY VẤN.....	32
II. CÁC BƯỚC CHÍNH ĐỂ TẠO (THIẾT KẾ) MỘT TRUY VẤN MỚI.....	32
III- TẠO QUERY BẰNG DESIGN VIEW.....	33
<b>III. 1-Tạo Select Query .....</b>	<b>33</b>
III.1.1. Tạo Select Query dùng để chọn các trường, các bản ghi và tính toán .....	34
III.1.2. Tạo Select Query dùng để tính tổng theo nhóm dữ liệu. ....	42
III.1.3 Tạo Select Query dùng hỏi -đáp dữ liệu.....	44
<b>III.2. Tạo Crosstab Query.....</b>	<b>45</b>
<b>III.3. Tạo Action Query .....</b>	<b>46</b>
III.3.1. Make Table Query (Truy vấn tạo bảng).....	46
III.3.2. Append Query (Truy vấn nối) .....	46
III.3.3. Update Query (Truy vấn cập nhật) .....	46
III.3.4. Delete Query (Truy vấn xóa) .....	46
IV. TẠO QUERY BẰNG WIZARD.....	47
<b>IV.1. Tạo Select Query bằng wizard .....</b>	<b>47</b>
<b>IV.2. Tạo Crosstab Query bằng wizard .....</b>	<b>47</b>
V- TẠO, SỬA QUERY BẰNG MÃ LỆNH SQL .....	47
<b>V.1-Cấu trúc các câu lệnh tạo Query .....</b>	<b>47</b>
<b>V.2.Cách tạo và sửa Query bằng SQL .....</b>	<b>48</b>
<b>CHƯƠNG V. MẪU BIỂU .....</b>	<b>49</b>
I. KHÁI NIỆM CHUNG.....	49
<b>I.1. Cấu trúc của mẫu biểu .....</b>	<b>49</b>
<b>I.2. Công dụng của các ô điều khiển: .....</b>	<b>49</b>
I.2.1. Thể hiện dữ liệu. ....	49
I.2.2. Nhập dữ liệu từ bàn phím.....	49
I.2.3. Thực hiện hành động. ....	49
I.2.4. Tổ chức giao diện Chương trình. ....	49
I.2.5.Tổ chức hệ Menu Bar cho chương trình. ....	49
II. TẠO MẪU BIỂU TỰ ĐỘNG VÀ BẰNG WIZARD .....	50
III. TẠO FORM BẰNG DESIGN VIEW .....	53
<b>III.1. Giới thiệu các loại điều khiển.....</b>	<b>53</b>
III.1.1. Các loại điều khiển.....	53
III.1.2. Cách dùng.....	53
<b>III.2. Điều khiển TextBox .....</b>	<b>54</b>
III.2.1. Tạo text box bị buộc (Buond).....	54
III.2.1. Tạo text box tính toán (Buond) - Unbound.....	55
<b>III.3. Điều khiển Label (nhãn).....</b>	<b>55</b>
<b>III.4. Thuộc tính của điều khiển.....</b>	<b>56</b>
III.4.1. Mở bảng thuộc tính của một ô điều khiển .....	56
III.4.2. Một số thuộc tính hay dùng. ....	56
<b>III.5. Các loại điều khiển dùng để cập nhập dữ liệu từ bảng chọn .....</b>	<b>56</b>
III.5.1. Cách dùng.....	57
III.5.2. Cách tạo.....	57
III.5.3. Phân loại ListBox và ComBo Box theo nguồn dữ liệu. ....	57
III.5.4. Tạo LIST/COMBO BOX không dùng WIZARD .....	57
III.5.5. Dùng Wizard tạo List Box/Combo Box .....	58
III.5.6. Dùng điều khiển Check Box, Toggle Button, Option Button.....	59

III.5.7. Sử dụng nhóm lựa chọn (Option Group).....	59
<b>III.6. Tạo Command Buttons (nút lệnh)</b> .....	<b>60</b>
IV. MẪU BIỂU TRÊN NHIỀU BẢNG .....	61
IV.1. Công dụng.....	61
IV.1. Quan hệ giữa các mẫu biểu chính và phụ.....	61
V.3. Cách tổ chức mẫu biểu chính và mẫu biểu phụ. ....	61
<b>CHƯƠNG VI. BÁO BIỂU</b> .....	<b>63</b>
I. KHÁI NIỆM VỀ BÁO BIỂU, SO SÁNH BÁO BIỂU VỚI MẪU BIỂU .....	<b>Error!</b>
<b>Bookmark not defined.</b>	
I.1. Công dụng của báo biểu. ....	Error! Bookmark not defined.
I.2. Sự giống nhau giữa báo biểu và mẫu biểu.....	Error! Bookmark not defined.
I.3. Một số đặc điểm khác của báo biểu .....	Error! Bookmark not defined.
II. XÂY DỰNG BÁO BIỂU BẰNG HỘP CÔNG CỤ .....	63
II.1. Các vấn đề thường gặp khi xây dựng báo biểu.....	63
II.2. Các công cụ thường sử dụng .....	63
II.3. Các bước xây dựng báo biểu .....	64
II.4. Thực hiện in báo biểu .....	65
III. SẮP XẾP VÀ TẬP HỢP DỮ LIỆU THEO NHÓM .....	65
III.1. Sắp xếp dữ liệu .....	65
III.2. Phân nhóm dữ liệu .....	66
III.3. Đầu nhóm và cuối nhóm.....	66
III.3.1. Tạo đầu nhóm và cuối nhóm trên báo biểu. ....	66
III.3.2. Đặt các ô điều khiển trên phân đầu/cuối nhóm. ....	66
III.3.3. Tổng hợp dữ liệu trên từng nhóm.....	66
III.3.4. In theo nhóm .....	67
IV. NHÓM DỮ LIỆU THEO TRƯỜNG PHÂN NHÓM .....	67
IV.1. Cách phân nhóm đối với trường/biểu thức kiểu số .....	67
IV.1.1. Phân nhóm theo giá trị. ....	67
IV.1.2. Phân nhóm theo miền giá trị.....	67
IV.2. Cách phân nhóm đối với trường/biểu thức kiểu Date/time .....	67
IV.2.1. Phân nhóm theo giá trị. ....	67
IV.2.2. Phân nhóm theo miền giá trị.....	67
IV.3. Cách phân nhóm đối với trường/biểu thức kiểu Text .....	68
IV.3.1. Phân nhóm theo giá trị.....	68
IV.3.2. Phân nhóm theo các ký tự đầu.....	68
V. TỔNG HỢP DỮ LIỆU TRÊN BÁO BIỂU .....	68
V.1. Các hàm dùng để tổng hợp dữ liệu.....	68
V.2. Tổng hợp dữ liệu trên mỗi nhóm.....	68
V.3. Tổng hợp dữ liệu trên toàn báo biểu .....	68
V.4. So sánh dữ liệu tổng hợp trên các phần của báo biểu.....	68
V.5. Không in một số ô điều khiển của báo biểu.....	69
VI. BÁO BIỂU CHÍNH VÀ BÁO BIỂU PHỤ .....	69
VII. XÂY DỰNG BÁO BIỂU TỰ ĐỘNG BẰNG WIZARD .....	69
<b>CÁC BÀI TẬP THỰC HÀNH</b> .....	<b>71</b>



## GIỚI THIỆU MICROSOFT ACCESS

Microsoft Access là hệ quản trị cơ sở dữ liệu trên môi trường Windows, trong đó có sẵn các công cụ hữu hiệu và tiện lợi để tự động sản sinh chương trình cho hầu hết các bài toán thường gặp trong quản lý, thống kê, kế toán. Với Access, người dùng không phải viết từng câu lệnh cụ thể như trong Pascal, C hay Foxpro mà chỉ cần tổ chức dữ liệu và thiết kế các yêu cầu, công việc cần giải quyết.

Hiện nay thường sử dụng 4 phiên bản Access là: Access 2.0 trong bộ Microsoft Office 4.3 , Access 7.0 for Windows 95, Access 97 trong bộ Microsoft Office 97, Access 2000 trong bộ Microsoft Office 2000.

Sáu đối tượng công cụ mà Access cung cấp là: Bảng (Table), Truy vấn (Query), mẫu biểu (Form), báo biểu (Report), Macro và đơn thể (Module).

Bảng có cấu trúc tương tự như một tệp DBF của Foxpro được dùng để lưu trữ dữ liệu của cơ sở dữ liệu (CSDL). Một CSDL thường gồm nhiều bảng có quan hệ với nhau.

Truy vấn là công cụ mạnh của Access dùng để tổng hợp, sắp xếp, tìm kiếm dữ liệu trên các bảng. Khi thực hiện truy vấn sẽ nhận được một tập hợp kết quả thể hiện trên màn hình dưới dạng bảng, gọi là DynaSet. DynaSet chỉ là bảng kết quả trung gian, không được ghi lên đĩa và nó sẽ bị xoá khi kết thúc truy vấn. Tuy nhiên có thể sử dụng một DynaSet như một bảng để xây dựng các truy vấn khác. Chỉ với truy vấn đã có thể giải quyết khá nhiều dạng toán trong quản trị cơ sở dữ liệu.

Mẫu biểu thường dùng để tổ chức cập nhật dữ liệu cho các bảng và thiết kế giao diện Chương trình. Tuy có thể nhập liệu trực tiếp vào các bảng, nhưng mẫu biểu sẽ cung cấp nhiều khả năng nhập liệu tiện lợi nhất Nhận dữ liệu từ một danh sách, nhận các hình ảnh, nhập dữ liệu đồng thời trên nhiều bảng. Mẫu biểu còn cho phép nhập các giá trị riêng lẻ (không liên quan đến bảng) từ bàn phím. Mẫu biểu còn có một khả năng quan trọng khác là tổ chức giao diện Chương trình dưới dạng một bảng nút lệnh hoặc một hệ thống menu.

Báo biểu là công cụ tuyệt vời phục vụ công việc in ấn, nó cho các khả năng :

- In dữ liệu dưới dạng bảng.
- In dữ liệu dưới dạng biểu bảng.
- Sắp xếp dữ liệu trước khi in.
- Sắp xếp và phân nhóm dữ liệu tới 10 cấp. Cho phép thực hiện các phép toán để nhận dữ liệu tổng hợp trên mỗi nhóm. Ngoài ra, dữ liệu tổng hợp nhận được

trên các nhóm lại có thể đưa vào các công thức để nhận được sự so sánh, đối chiếu trên các nhóm và trên toàn báo cáo.

- In dữ liệu của nhiều bảng có quan hệ trên một báo cáo.

Cũng cần nói thêm, việc chọn Font chữ, cỡ chữ, kiểu in và việc trình bày trên báo cáo được tiến hành rất đơn giản.

Macro bao gồm một dãy các hành động (Action) dùng để tự động hoá một loạt các thao tác. Macro thường dùng với mẫu biểu để tổ chức giao diện Chương trình.

Đơn thể là nơi chứa các hàm, thủ tục viết bằng ngôn ngữ Access Basic. Mặc dù các công cụ mà Access cung cấp khá đầy đủ, nhưng lẽ dĩ nhiên không thể bao quát được mọi vấn đề đa dạng của thực tế. Các hàm, thủ tục của Access Basic sẽ trợ giúp giải quyết những phần việc khó mà công cụ không làm nổi.

## CHƯƠNG I. LÀM VIỆC VỚI MICROSOFT ACCESS

Chương này trình bày các vấn đề sau:

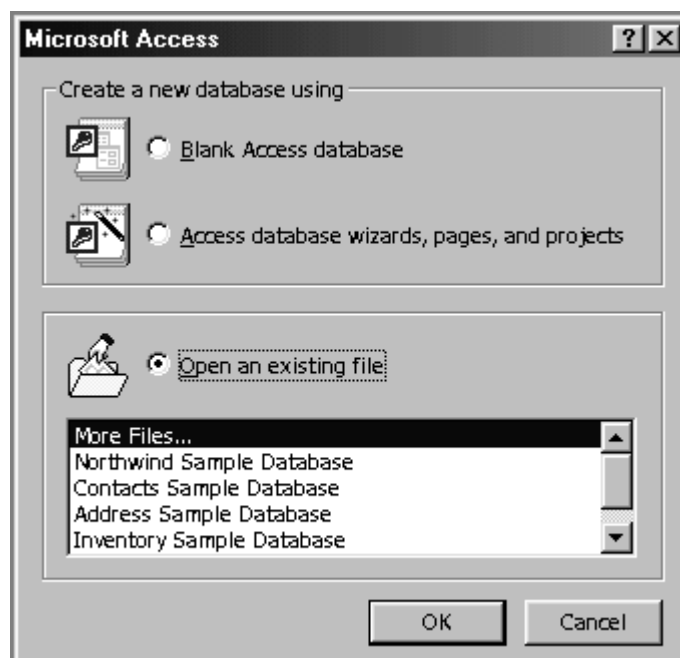
- Cách vào môi trường của Access.
- Khái niệm cơ sở dữ liệu (CSDL). Đó là hệ Chương trình do Access tạo ra và được lưu trên một tệp có đuôi MDB. Một CSDL gồm 6 nhóm đối tượng là: Bảng, truy vấn, mẫu biểu, báo biểu, macro và module.
- Tạo CSDL mới và làm việc với CSDL đã có.
- Các cửa sổ chính và các thao tác cơ bản trong Access.

### I. VÀO MÔI TRƯỜNG ACCESS.

Chương trình Access được xây dựng và thực hiện trong môi trường Access vì vậy chúng ta cần biết cách vào môi trường Access. Để làm điều này trình tự thao tác như sau:

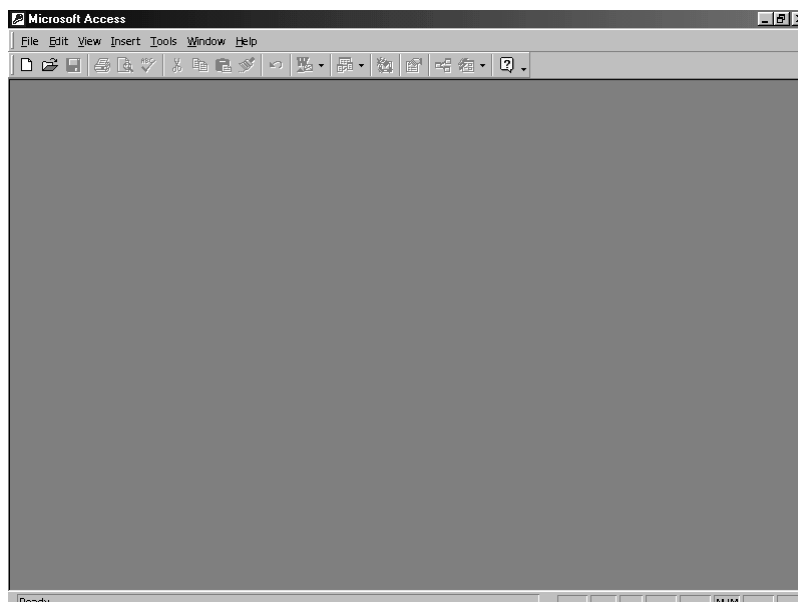
1. Khởi động Windows nên đang ở môi trường DOS.
2. Chọn Start, Programs, Microsoft Access .

Kết quả ta nhận được cửa sổ sau trên đó có các tùy chọn:



- Nếu muốn mở một CSDL đã có thì chọn trong danh sách Open an Existing Database rồi bấm OK.
- Nếu muốn tạo một CSDL mới thì chọn Blank Database rồi bấm OK.

- Nếu muốn tạo một CSDL mới theo những chủ đề có sẵn thì chọn Database Wizard rồi bấm OK.
- Nếu chưa quen với cách mở CSDL đã có và cách tạo CSDL mới thì ta bấm chuột tại nút Cancel. Khi đó sẽ mở cửa sổ Microsoft Access có dạng:



## II. CHƯƠNG TRÌNH ACCESS .

### II.1. Tập chứa Chương trình Access

Mỗi một tập Chương trình thường có một đuôi qui định, ví dụ các tập Chương trình C có đuôi là .C, tập Chương trình Pascal có đuôi là PAS, tập Chương trình Foxpro có đuôi là .PRG. Một tập Chương trình do Access tạo ra có đuôi là .MDB

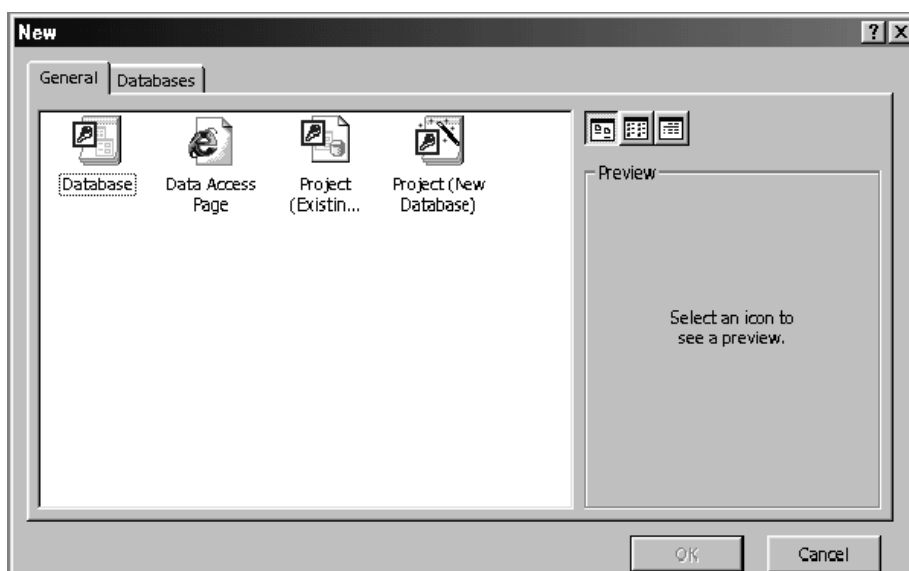
### II.2. Một hệ Chương trình do Access tạo ra (hệ Chương trình Access)

Chương trình Access gọi là một Database (CSDL). Trong các ngôn ngữ truyền thống như C, Pascal, Foxpro, một hệ Chương trình gồm các tập Chương trình và các tập dữ liệu được tổ chức một cách riêng biệt. Nhưng trong Access toàn bộ Chương trình và dữ liệu được chứa trong một tập duy nhất có đuôi .MDB. như vậy thuật ngữ hệ Chương trình hay CSDL được hiểu là tổ hợp bao gồm cả Chương trình và dữ liệu. Để ngắn gọn nhiều khi ta gọi là Chương trình thay cho thuật ngữ hệ Chương trình. như vậy dưới đây khi nói đến Chương trình hay hệ chương trình hay CSDL thì cùng có nghĩa đó là một hệ phần mềm gồm cả Chương trình và dữ liệu do Access tạo ra.

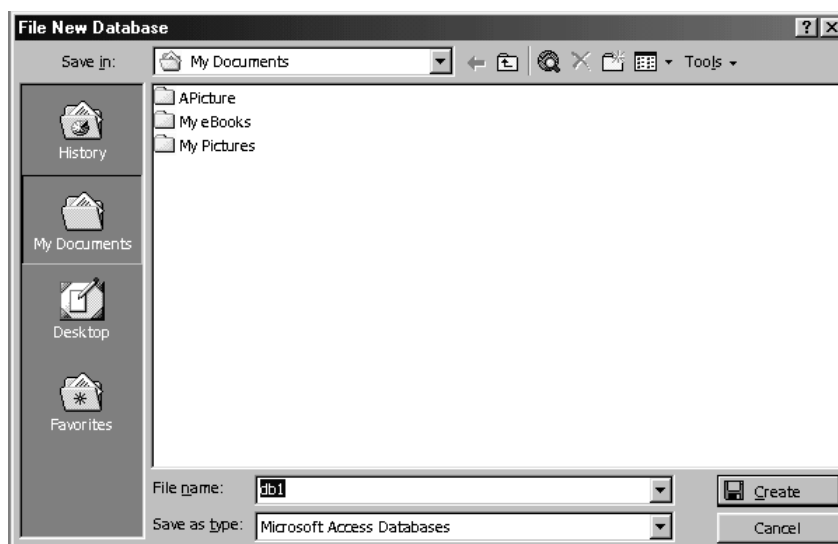
### III. TẠO MỘT CSDL MỚI .

Nên xây dựng trước một thư mục mới chứa CSDL cần tạo, ví dụ thư mục: TG ACCESS Vì nếu ta chứa các CSDL trong thư mục ACCESS thì chúng có thể sẽ bị mất mỗi khi cài đặt lại Access, và khó quản lý.

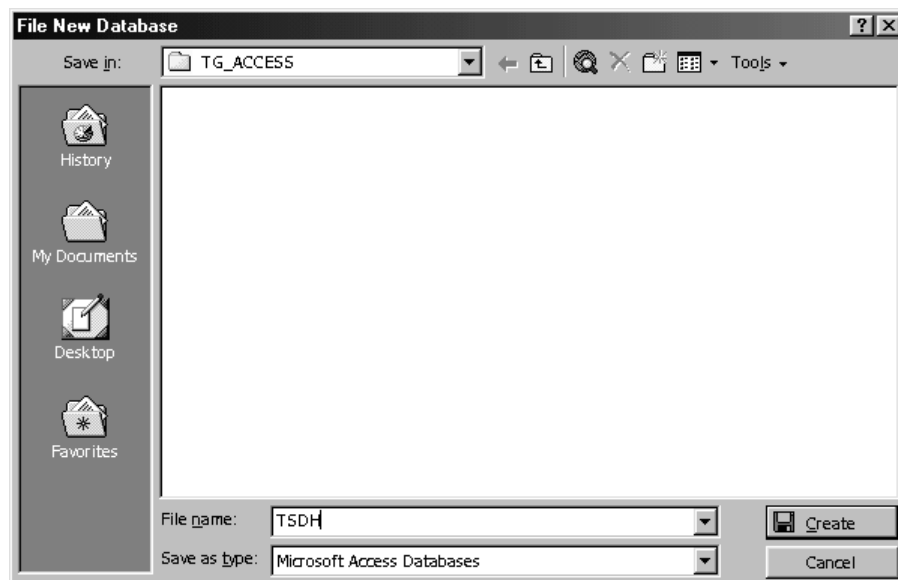
Bước 1: Từ cửa sổ Microsoft Access: Chọn menu File, chức năng New Database (hoặc kích chuột tại biểu tượng New Database ), được cửa sổ sau:



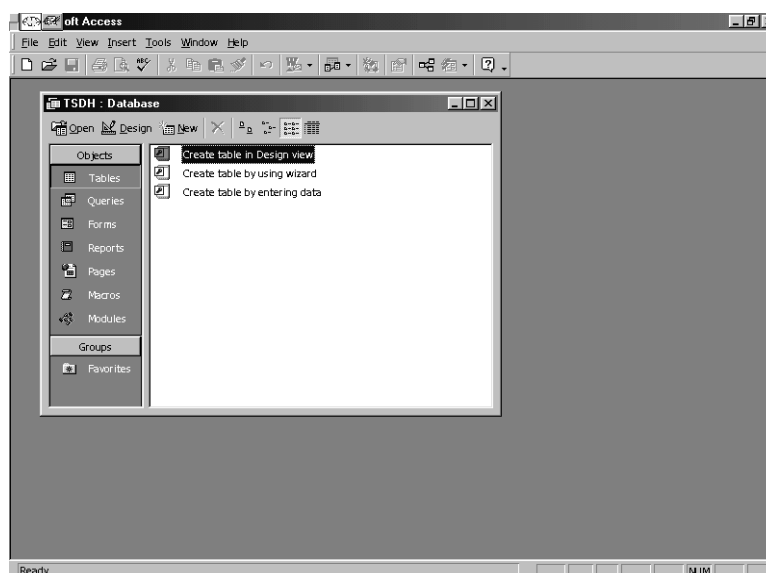
Bước 2: Nhấn đúp chuột tại biểu tượng Blank DataBase, hoặc nhấn nút OK để hiện ra cửa sổ:



Bước 3: Chọn thư mục sẽ chứa tệp CSDL cần tạo (ví dụ thư mục TG ACCES trong hộp Save in), sau đó đặt tên CSDL trong hộp File name (ví dụ TSDH), cửa sổ tương ứng có dạng:



Kích chuột tại nút Create. → Xuất hiện cửa sổ Database:



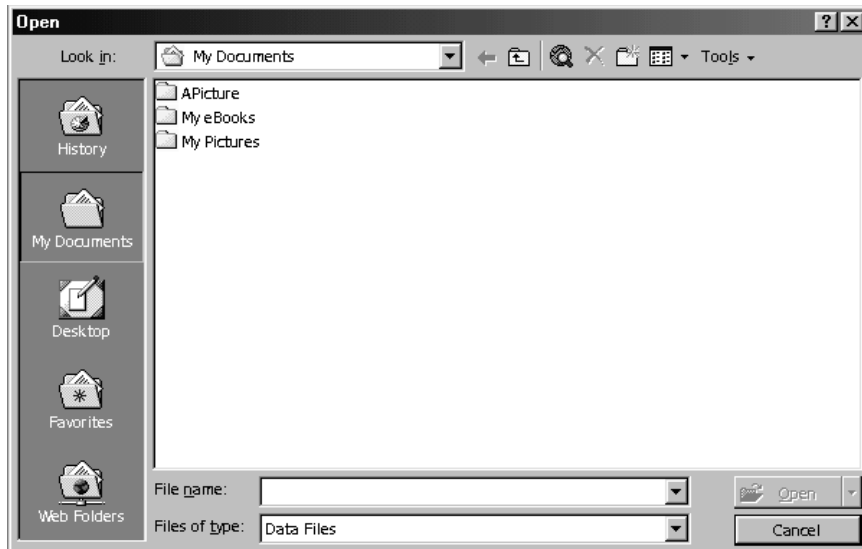
Đây là một trong những cửa sổ rất quan trọng của Access. Cửa sổ bao gồm:

- Hệ menu với các menu: File, Edit, View, ...
- Tiêu đề Database: TSDH cho biết tên của CSDL.
- Các đối tượng công cụ (các mục): Table, Query, Form, Report, Macro, Modul dùng để xây dựng các đối tượng trong Chương trình Access

#### IV. LÀM VIỆC VỚI CSDL ĐÃ TỒN TẠI

Giả sử đã có CSDL TSDH trong thư mục C:\TG\_ACCESS. Để làm việc với CSDL trên (xem, sửa, bổ sung, thực hiện chương trình) ta lần lượt thao tác như sau:

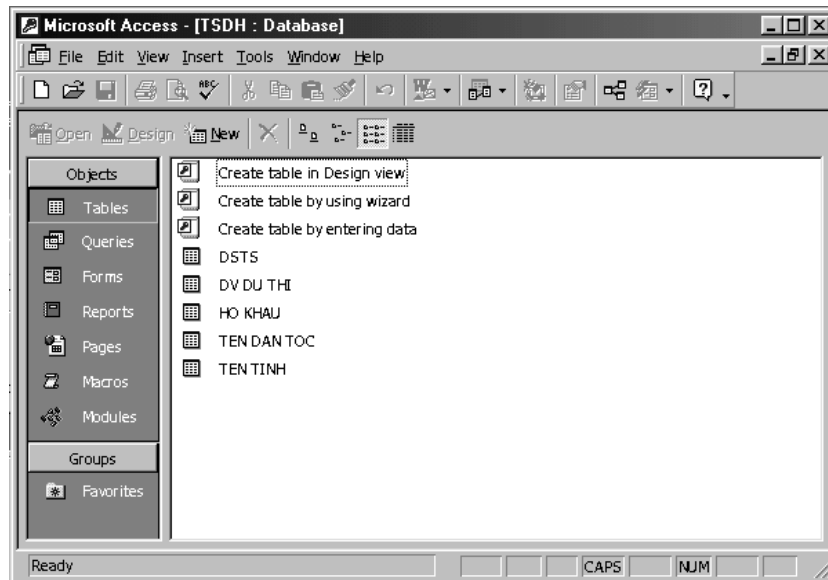
B1: Từ cửa sổ Microsoft Access chọn menu File, Open Database hoặc kích chuột tại biểu tượng Open Database, kết quả nhận được cửa sổ:



B2: Chọn thư mục chứa Database cần mở trong hộp Look in. Kết quả sẽ hiện danh sách các CSDL (các tệp có đuôi .MDB) trong thư mục này:



B3. Chọn CSDL cần mở trong hộp File Name - giả sử chọn CSDL TSDH, sau đó kích chuột tại nút OK. Kết quả nhận được cửa sổ Database của CSDL TSDH như sau:



Khi mở một CSDL, nếu muốn chọn mục nào thì ta bấm chuột vào TAB của mục đó.



## CHƯƠNG II. BẢNG

Trong chương này sẽ trình bày cách sử dụng công cụ TABLE để làm việc với các bảng gồm:

- Tạo bảng mới
- Đặt khoá chính và tạo các chỉ mục cho bảng
- Nhập dữ liệu vào bảng
- Chỉnh sửa cấu trúc của bảng
- Thiết lập quan hệ giữa các bảng
- Sử dụng thuộc tính của trường để trình bày dạng dữ liệu của trường và kiểm tra tính hợp lệ của dữ liệu khi cập nhật.

### I. KHÁI NIỆM BẢNG

Bảng là nơi chứa dữ liệu của một đối tượng nào đó.

Một cơ sở dữ liệu (CSDL) thường gồm nhiều bảng.

Một bảng gồm nhiều trường có các kiểu khác nhau như: Text, Number, Date/Time...

Các bảng trong một CSDL thường có quan hệ với nhau.

### II. TẠO BẢNG

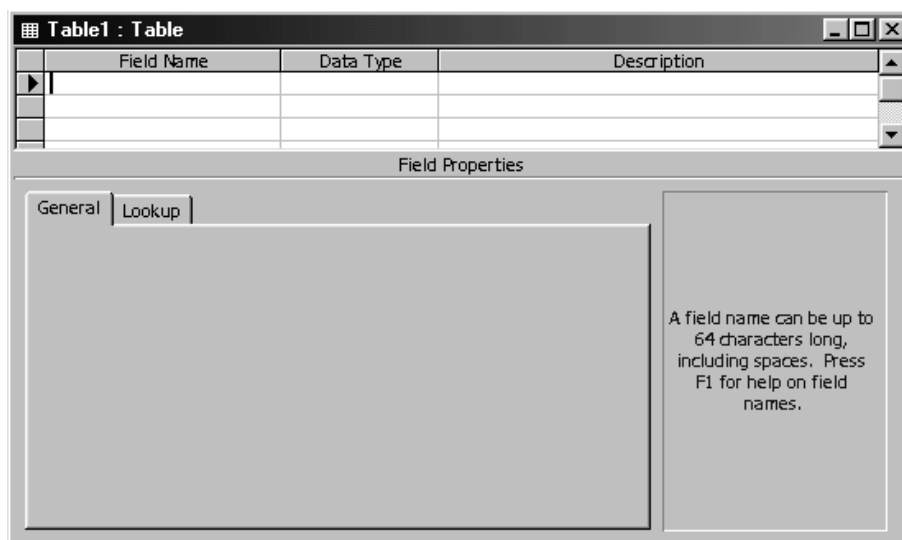
Từ cửa sổ Database chọn mục Table

- Chọn nút New dùng để tạo bảng mới
  - Nút Open dùng để mở nhập liệu cho bảng được chọn.
  - Nút Design dùng để xem, sửa cấu trúc của bảng được chọn
- Để tạo bảng mới ta chọn nút New, kết quả có được:



## II.1. Tạo bảng bằng Design View.

Chọn Design View trong cửa sổ New Table, kết quả nhận được cửa sổ thiết kế bảng như sau:



Cửa sổ Table được chia làm 2 phần:

- Phần trên gồm 3 cột: Field Name, Data Type và Description, dùng để khai báo các trường của bảng, mỗi trường khai báo trên 1 dòng.
- Phần dưới dùng để qui định các thuộc tính cho các trường.

*B1: Gõ tên trường ở ô trong cột Fieldname*

Tên trường (Field Name): Là một dãy không quá 64 ký tự, bao gồm chữ cái, chữ số, khoảng trống.

## B2. Chọn kiểu trường trong cột Data Type

Access gồm các kiểu sau:

Tên trường	Mô tả	Độ lớn
Text	Ký tự	dài tối đa 255 Byte
Memo	Ký tự	dài tối đa 64000 Byte
Number	Số nguyên, thực	dài : 1 , 2, 4 hoặc 8 Byte
Date/time	Ngày tháng/giờ	dài 8 Byte
Currency	Tiền tệ	dài 8 Byte
AutoNumber	Số	dài 8 Byte
Yes/No	Boolean	1 Bit
OLE OObject	Đối tượng	1 Giga Byte
Hyperlink	Ký tự hoặc kết hợp ký tự và số	
Lookup Wizard	Cho phép chọn giá trị từ bảng khác	

B3. *Mô tả (Discreption)*: Để giải thích cho rõ hơn một trường nào đó Văn bản mô tả sẽ được hiển thị khi nhập số liệu cho các trường. Phần này có thể có hoặc không.

## B4. Xác định các thuộc tính của trường trong bảng Properties

(các thuộc tính này sẽ được nêu chi tiết hơn trong phần sau, tại đây ta có thể dùng các giá trị thuộc tính mặc định).

## II.2. Tạo bảng bằng Datasheet View :

B1: Từ cửa sổ Database, chọn Table Tab, chọn New

B2: Chọn Datasheet View từ hộp thoại New Table, OK

B3: Đặt tên trường : Kích đúp chuột vào tên trường → Nhập tên trường mới → Enter

B4: Nhập các bản ghi dữ liệu

B5: Ghi bảng : Mở Menu File, chọn Save, đặt tên, chọn OK

### **II.3. Tạo bảng bằng Table Wizard**

B1: Từ cửa sổ Database, chọn Table Tab, chọn New

B2: Chọn Table Wizard từ hộp thoại New Table, OK

B3: Chọn các trường cần thiết, Kích nút next

\* Đổi tên trường :chọn Rename Field, gõ tên trường mới,OK

B4: chọn Next, đặt tên bảng → Finish

B5: Nhập dữ liệu vào bảng

### **II.4. Tạo bảng bằng Import Table**

B1: Từ cửa sổ Database, chọn Table Tab, chọn New

B2: Chọn Import Table từ hộp thoại New Table, OK

B3: Chọn thư mục, kiểu file và tên file cần lấy

B4: Chọn Import

### **II.5. Tạo bảng bằng Link Table**

B1: Từ cửa sổ Database, chọn Table Tab, chọn New

B2: Chọn Link Table từ hộp thoại New Table, OK

B3: Chọn thư mục, kiểu file và tên file cần lấy

B5: Chọn Link

## **III. THUỘC TÍNH CỦA TRƯỜNG**

### **III.1. Công dụng:**

- Điều khiển hình thức thể hiện dữ liệu.
- Nhập liệu theo mẫu.
- Kiểm tra dữ liệu, ngăn cản nhập sai.
- Tăng tốc độ tìm kiếm.

### III.2. Cách đặt giá trị cho các thuộc tính.

Trong cửa sổ thiết kế bảng, mỗi khi chọn một trường ở nửa trên thì nửa dưới thể hiện các thuộc tính của trường vừa chọn. Mỗi thuộc tính nằm trên một dòng.

Lúc đầu mỗi thuộc tính hoặc chưa dùng (bỏ trống) hoặc có giá trị mặc định, ví dụ giá trị mặc định của thuộc tính FieldSize của trường kiểu Text là 50, của trường kiểu Number là Double.

Giá trị của thuộc tính có thể gõ trực tiếp từ bàn phím (như thuộc tính FieldSize của trường Text) hoặc có thể chọn từ một danh sách của Combo Box (như thuộc tính FieldSize của trường Number).

### III.3. Tổng quan về các thuộc tính của trường

Field Size.	Số ký tự của trường Text, hoặc kiểu của trường number.
Format	Dạng hiển thị dữ liệu kiểu ngày và số.
DecimalPlaces	Số chữ số thập phân trong kiểu number và currency.
InputMask (Mặt nạ nhập)	Quy định khuôn dạng nhập liệu
Caption	Đặt nhãn cho trường. Nhãn sẽ được hiển thị khi nhập liệu thay vì tên trường (nhãn mặc định).
Default Value	Xác định giá trị mặc định của trường.
Validation Rule	Quy tắc dữ liệu hợp lệ. Dữ liệu phải thoả mãn quy tắc này mới được nhập.
Required	Không chấp nhận giá trị rỗng. Cần phải nhập một dữ liệu cho trường.
AllowZeroLength	Chấp nhận chuỗi rỗng trong trường Text, Memo.
Indexed	Tạo chỉ mục để tăng tốc độ tìm kiếm trên trường này

### III.4. Chi tiết về các thuộc tính của trường

#### III.4.1. Thuộc tính Field Size

■ Với trường Text Độ dài mặc định là 50 Độ dài hợp lệ có thể đặt là từ 1 → 255

■ Với trường Number: Mặc định là Double (8 byte), có thể chọn các kiểu sau từ Combo Box:

FieldSize	Miền giá trị	Số byte lưu trữ
Byte	0→255	1
Integer	-32768→32767	2
LongInteger	-2147483648→214743647	4
Single	$-3.4 \cdot 10^{38} \rightarrow 3.4 \cdot 10^{38}$	4
Double	$-1.797 \cdot 10^{308} \rightarrow 1.797 \cdot 10^{308}$	8

### III.4.2. Thuộc tính Format.

Nếu bỏ qua Format, Access sẽ trình bày dữ liệu theo dạng General.

Các giá trị của thuộc tính Format đối với trường Number (giả định Decimal Places=2 - Hai chữ số phần thập phân)

Giá trị Format	Số	Được trình bày
General Number	1234.5	1234.5
Fixed	1234.5	1234.50
	1234.568	1235.57
Standard	1234.5	1,234.50 (dấu phẩy ngăn cách hàng ngàn)
Percent	0.824	82.40 %
Scientific	1234.5	1.23E+03
Currency	1234.5	1,234.50(UK) (dấu phẩy hàng ngàn)

Ghi chú:

1. General Number: Không phụ thuộc vào Decimal Places, hiện số chữ số thập phân cần thiết nhất

2. Decimal Places = n : Mọi dạng (trừ General Number) hiện đúng n chữ số thập phân.

3. Decimal Places = Auto : Khi đó:

Dạng Fixed: 0 số lẻ

Dạng khác (trừ General): 2 số lẻ

Các giá trị của thuộc tính Format đối với trường DATE/TIME.

Giá trị Format	Ngày/giờ được trình bày
General Date	1/31/92 4:30:00 PM (U.S) 31/01/92 16:30:00 (U.K)
Long Date	Friday, January 31 , 1992 (U.S) 31 January 1992 (U.K)
Medium Date	31-Jan- 1992
Short Date	1/31/92 (U.S) 31/01/92(U.K)
Long Time	4:30:00 PM
Medium Time	04:30 phần mềm
Short Time	16:30

Các giá trị của thuộc tính Format đối với trường YES/NO

Giá trị Format	ý nghĩa
Yes/No	Giá trị logic là Yes và No
True/False	Giá trị logic là True và False
On/Off	Giá trị logic là On và Off

### ***III.4.3. Thuộc tính Input Mask (mặt nạ nhập liệu).***

■ *Công dụng:*

1. Tạo khuôn dạng nhập liệu cho dễ nhìn. Trên khuôn dạng có thể thấy các vị trí để nhập liệu và các ký tự phân cách (ví dụ dấu chấm phân cách phần nguyên và phần phân, dấu gạch ngang để phân cách các cụm ký tự của số tài khoản,...)
2. Kiểm tra tính hợp lệ của mỗi ký tự gõ vào. Tại mỗi vị trí trên khuôn dạng có thể quy định lớp ký tự được phép gõ (ví dụ nếu quy định các chữ số thì Access sẽ không nhận các ký tự không phải là chữ số).
3. Tự động biến đổi ký tự được nhập (ví dụ chuyển sang chữ hoa).
4. Che dấu thông tin gõ vào: Dùng mặt nạ kiểu Password, khi đó các ký tự gõ vào được thể hiện thành dấu \*.

■ *Cách tạo mặt nạ nhập liệu:* Mặt nạ nhập liệu là một dãy ký tự gồm các loại sau:

1. Ký tự khuôn dạng: Mỗi ký tự khuôn dạng thể hiện một vị trí giành cho ký tự gõ vào và quy định lớp ký tự được gõ.
  - 0 vị trí dành cho chữ số 0..9, bắt buộc
  - 9 vị trí dành cho chữ số 0..9, không bắt buộc
  - # vị trí dành cho chữ số, dấu + - dấu cách
  - L vị trí dành cho một chữ cái, bắt buộc
  - ? vị trí dành cho chữ cái hoặc dấu cách, không bắt buộc
  - A vị trí dành cho ký tự chữ hoặc số, bắt buộc
  - a vị trí dành cho ký tự chữ hoặc số, không bắt buộc
  - & vị trí dành cho một ký tự bất kỳ, bắt buộc
  - C vị trí dành cho một ký tự bất kỳ, không bắt buộc
2. Ký tự chuyển đổi gồm:
  - Ký tự < dùng để đổi các ký tự đứng sau < sang chữ thường.
  - Ký tự > dùng để đổi các ký tự đứng sau < sang chữ hoa.
3. Ký tự canh phải là ký tự chấm than. Các ký tự nhập vào sẽ được dồn sang phải. Ví dụ nếu dùng mặt nạ: Input Mask: !aaaa (a là ký tự khuôn dạng giành cho các ký tự chữ và số, không bắt buộc phải nhập). Nếu nhập 2 ký tự HA thì 2 ký tự này sẽ dồn bên phải và 3 dấu cách đặt bên trái.



4. Các ký tự phân cách. Các ký tự này được hiển thị trên khuôn để tách các phần trong dãy dữ liệu nhập vào với mục đích dễ quan sát, dễ kiểm tra. Có thể dùng bất kỳ ký tự nào ngoài các ký tự trong 3 điểm trên làm ký tự phân cách. Sau đây là một số ký tự phân cách hay dùng:

- Dấu . để phân cách phần nguyên và phần phân.
- Dấu , để phân cách hàng nghìn, triệu, tỷ,...
- Dấu - hoặc 1 dùng để phân cách trong dữ liệu Date/Time (tùy thuộc cách thiết lập trong mục Control Panel, International)

#### ***III.4.4. Thuộc tính Default Value***

Dùng thuộc tính này để đặt giá trị mặc định cho trường. Giá trị mặc định có thể là một hằng hay một hàm của Access.

#### ***III.4.5. Thuộc tính Required (trường bắt buộc phải có số liệu)***

Muốn bắt buộc trường phải có số liệu ta đặt thuộc tính Required thành Yes.

#### ***III.4.6. Thuộc tính AllowZerolength***

Nếu đặt là Yes sẽ cho phép các trường Text và memo nhận các chuỗi rỗng.

#### ***III.4.7. Thuộc tính ValidationRule***

Dùng thuộc tính này để kiểm tra sự hợp lệ của dữ liệu nhập vào. Muốn vậy trong thuộc tính ValidationRule ta đặt một biểu thức (điều kiện) hợp lệ.

Ví dụ:

<>	giá trị nhập vào phải khác 0
0 or >100	giá trị nhập vào phải bằng 0 hoặc lớn hơn 100
Like "K???"	phải nhập 4 ký tự, ký tự đầu phải là K
Like "CTY*"	ba ký tự đầu phải là CTY
<#/1/92#	ngày nhập trước năm 1992
>#/1/91# and <#/1/92#	ngày nhập phải trong năm 1991

### III.4.8. Thuộc tính Indexed

Thuộc tính này có thể nhận các giá trị:

No	Không tạo chỉ mục hoặc xoá chỉ mục đã lập
Yes (Duplicatates Ok)	Tạo chỉ mục
Yes (No Duplicatates)	Tạo chỉ mục kiểu Unique (các giá trị cần khác nhau như thể khoá chính).

## IV. THAY ĐỔI THIẾT KẾ, CHỈNH SỬA CẤU TRÚC BẢNG

Khi thiết kế bảng, thường có các yêu cầu sau:

- Xoá một hoặc nhiều trường
- Thay đổi tên, kiểu, mô tả và thuộc tính của trường
- Thay đổi vị trí của trường
- Thêm trường mới

Dưới đây sẽ trình bày cách để thực hiện những yêu cầu trên:

- Xoá trường: Chọn trường, bấm phím Delete hoặc chọn Delete Row từ Menu Edit. *Chú ý:* Để xoá đồng thời nhiều trường, ta có thể dùng các phím Shift và Ctrl để chọn các trường cần xoá, rồi bấm phím Delete hoặc chọn Edit, Delete Row.
- Để thay đổi nội dung (tên, kiểu, ...) của một trường, ta chỉ việc dùng bàn phím để thực hiện các thay đổi cần thiết.
- Để chèn thêm một trường mới ta làm như sau: Chọn trường mà trường mới sẽ được chèn vào trước, rồi chọn Edit, Insert Row.
- Di chuyển trường: Chọn trường cần di chuyển, rồi kéo trường tới vị trí mới.

## V. ĐẶT KHOÁ CHÍNH

### V.1. Định nghĩa

Khoá chính là một hoặc nhiều trường xác định duy nhất một bản ghi.

## V.2. Lợi ích.

Access tự động tạo chỉ mục (Index) trên khoá nhằm tăng tốc độ truy vấn và các thao tác khác.

Khi xem mẫu tin (dạng bảng hay mẫu biểu), các mẫu tin sẽ được trình bày theo thứ tự khoá chính.

Khi nhập số liệu, Access kiểm tra sự trùng nhau trên khoá chính.

Access dùng khoá chính để tạo sự liên kết giữa các bảng.

## V.3. Xử lý khoá chính

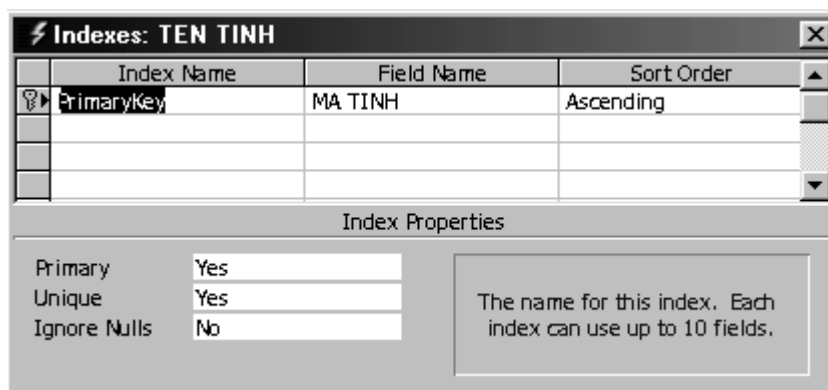
### ■ Đặt khoá chính

Chọn các trường làm khoá chính.

Chọn Edit, Primary key hoặc nhấp biểu tượng khoá trên thanh công cụ → Các trường được chọn làm khoá sẽ có hình chìa khoá ở đầu.

### ■ Đặt lại khoá chính - Chọn lại các trường làm khoá, rồi cũng thực hiện như trên.

### ■ Xoá khoá chính: Khoá chính là không bắt buộc. Khi đã thiết lập khoá chính, mà ta lại muốn xoá đi thì cách làm như sau: Chọn View → Indexes. Kết quả nhận được cửa sổ Indexes trong đó chứa các trường của khoá chính như sau:

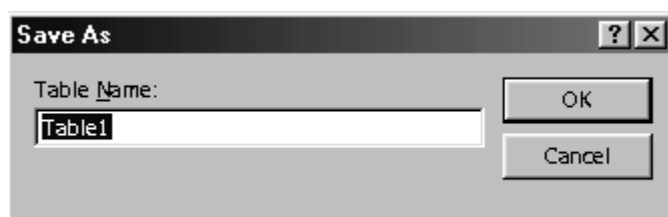


Chọn các trường (của khoá chính, mỗi trường trên một dòng) cần xoá rồi bấm phím Delete.

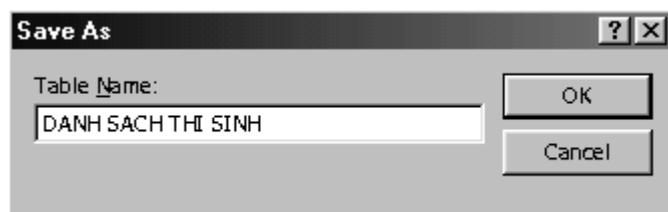
## VI. LƯU CẤU TRÚC BẢNG VÀ ĐẶT TÊN BẢNG

Sau khi hoàn chỉnh việc thiết kế, ta cần ghi cấu trúc bảng lên (ra và đặt tên cho bảng. Cách làm như sau:

Chọn File, Save hoặc bấm chuột tại biểu tượng save trên thanh công cụ . Khi ghi lần đầu thì Access sẽ hiện cửa sổ Save as:



Trong hộp Table Name ta đưa vào tên bảng, rồi bấm OK. Ví dụ chọn tên bảng là [DANH SACH THI SINH]:



Nếu trong thiết kế chưa đặt khoá chính thì Access sẽ hỏi người sử dụng có muốn Access tự động đặt khoá chính (bằng trường ID có kiểu AutoNumber) nếu chọn YES, ngược lại nếu chọn NO thì bảng không có khoá chính.

## **VII. THAY ĐỔI CẤU TRÚC BẢNG VÀ NHẬP DỮ LIỆU.**

### **VII.1. Thay đổi cấu trúc bảng**

Trong cửa sổ Tables chọn bảng cần thay đổi cấu trúc, rồi bấm nút Design. Kết quả nhận được cửa sổ thiết kế của bảng được chọn. Trong cửa sổ thiết kế ta có thể xem và thay đổi cấu trúc bảng như đã khi sử dụng cách tạo bảng bằng Design. Cuối cùng ghi lại các thay đổi.

### **VII.2. Nhập dữ liệu vào bảng.**

Để nhập dữ liệu vào bảng đã tồn tại, ta làm như sau:

Từ cửa sổ Database chọn mục Table để mở cửa sổ Tables.

Trong cửa sổ Tables chọn bảng cần nhập dữ liệu, rồi bấm nút Open.

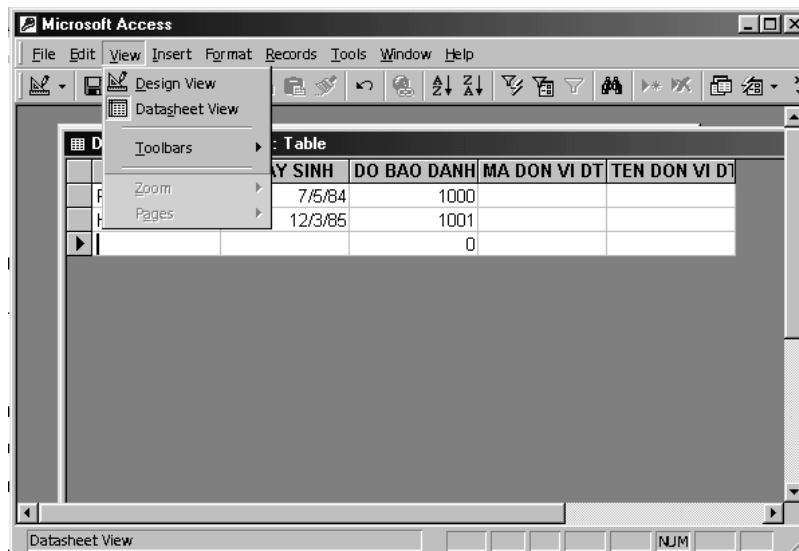
Kết quả nhận được cửa sổ nhập liệu chứa các bản ghi đã nhập.

Ta có thể xem, sửa và bổ sung các bản ghi mới trong cửa sổ nhập liệu.

*Chú ý:* Để chọn Font chữ thích hợp, ta dùng chức năng Font của menu Format.

### VII.3. Chuyển đổi giữa hai chế độ nhập dữ liệu và thiết kế bảng.

Khi đang ở cửa sổ thiết kế, ta có thể chuyển sang cửa sổ nhập liệu và ngược lại, bằng cách dùng menu View. Chọn menu View sẽ nhận được các chức năng sau: Datasheet View – Hiện cửa sổ nhập dữ liệu, Design View – Hiện cửa sổ thiết kế



### VIII. LẬP QUAN HỆ GIỮA CÁC BẢNG.

Access dùng quan hệ để đảm bảo những ràng buộc toàn vẹn giữa các bảng liên quan trong các phép thêm, sửa xoá bản ghi.

Nguyên tắc đặt quan hệ là chỉ định một hoặc một nhóm trường chứa cùng giá trị trong các mẫu tin có liên quan.

Thông thường đặt quan hệ giữa khoá chính của một bảng với trường nào đó của bảng khác (bảng này gọi là bảng quan hệ), các trường này thường cùng tên, cùng kiểu.

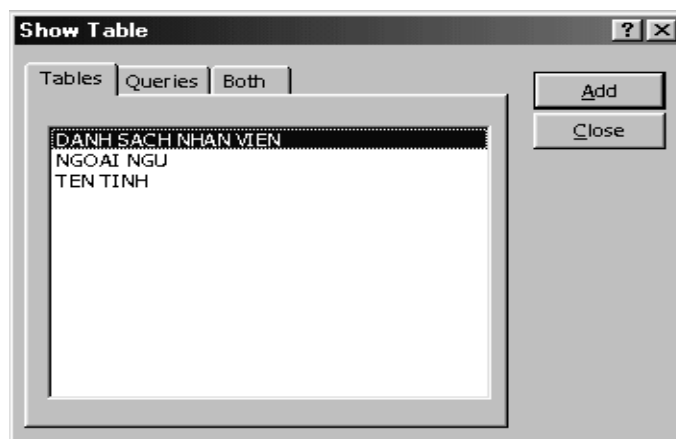
#### VIII.1. Các loại quan hệ trong Access.

\* Quan hệ 1-1: Các trường sử dụng để tạo quan hệ trong hai bảng đều là khoá chính. Khi đó, mỗi bản ghi trong bảng quan hệ phải có một bản ghi tương ứng trong bảng khoá chính, ngược lại, với mỗi giá trị khóa trong bảng chính thì chỉ có duy nhất một bản ghi tương ứng trong bảng quan hệ.

\* Quan hệ 1-n: trường liên kết dùng trong bảng chính phải là khoá chính, còn trường trong bảng quan hệ không phải là khoá chính của bảng đó (khoá ngoại).

## VIII.2. Cách tạo quan hệ

B1 : Trong cửa sổ Database, từ menu Tools chọn Relationships để mở cửa sổ Add Table:



Tác dụng của 3 nút bên trên cửa sổ:

- Nút Tables để hiện các bảng.
- Nút Queries để hiện các truy vấn.
- Nút Both để hiện các bảng và các truy vấn.

Chọn các bảng và truy vấn để đưa vào quan hệ. Sử dụng các phím Ctrl hoặc Shift để chọn nhiều bảng hoặc truy vấn, sau đó bấm nút Add. Sau khi chọn xong, bấm nút Close để đóng cửa sổ Show Table.

B2: Chọn một trường từ bảng chính (Primary table) và kéo sang trường tương ứng của bảng quan hệ, rồi bấm chuột tại Create để tạo quan hệ. Khi đó sẽ có đường thẳng nối giữa hai trường biểu diễn quan hệ vừa tạo, lúc này kiểu quan hệ mặc định theo trường chúng ta chọn để tạo quan hệ. Trong trường hợp muốn định nghĩa (hoặc thay đổi kiểu quan hệ) ta bấm nút Join Type để hiện các kiểu quan hệ:

- Chọn kiểu quan hệ 1 để tạo mối quan hệ 1-1 (Chú ý: quan hệ này chỉ tạo được khi hai trường đều là khóa chính) → bấm OK rồi bấm nút
- Chọn kiểu quan hệ 2 để tạo quan hệ 1-n từ bảng chính sang bảng quan hệ.
- Chọn kiểu quan hệ 3 để tạo quan hệ 1-n từ bảng quan hệ sang bảng chính

### **Ví dụ:**

Giả sử CSDL NHAN SU gồm 3 bảng: [DANH SACH NHAN VIEN], [TEN TINH] và [NGOAI NGU] có cấu trúc như sau:

Field Name	Data Type	Description
Ma NV	Number	Ma nhan vien
Ten NV	Text	Ten nhan vien
Ngay sinh	Date/Time	
Ma tinh	Number	Que nhan vien
HS luong	Number	He so luong

Field Name	Data Type	Description
Ma NV	Number	
Ngoai ngu	Text	
Trinh do	Text	

Field Name	Data Type	Description
Ma tinh	Number	
Ten tinh	Text	

Ta tạo các quan hệ sau:

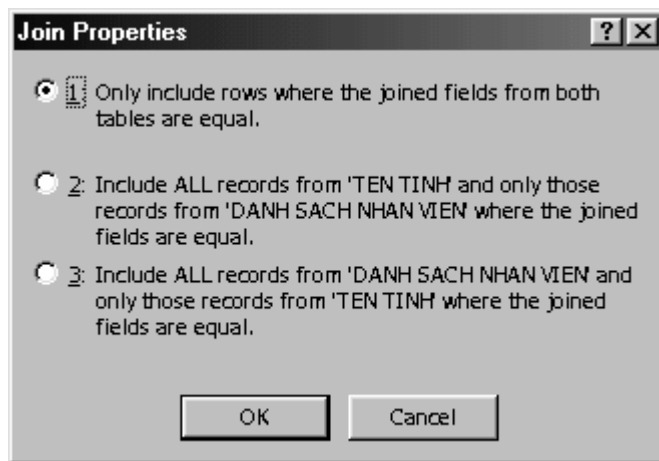
- Quan hệ một – một giữa bảng [DANH SACH NHAN VIEN] và bảng [TEN TINH] theo cặp trường quan hệ [Ma tinh] – [Ma tinh] với ý nghĩa: một mã tỉnh chỉ tương ứng với một tên tỉnh.
- Quan hệ một – nhiều giữa bảng [DANH SACH NHAN VIEN] và bảng [NGOAI NGU] theo cặp trường quan hệ [Ma NV] – [Ma NV] với ý nghĩa: một nhân viên có thể biết nhiều ngoại ngữ.

Cách tạo các quan hệ:

- Trong cửa sổ Show Table, chọn 3 bảng trên rồi nhấn Add
- Nhấn nút Close để đóng cửa sổ Show Table, kết quả nhận được 3 bảng vừa chọn trong cửa sổ Relationships như sau:

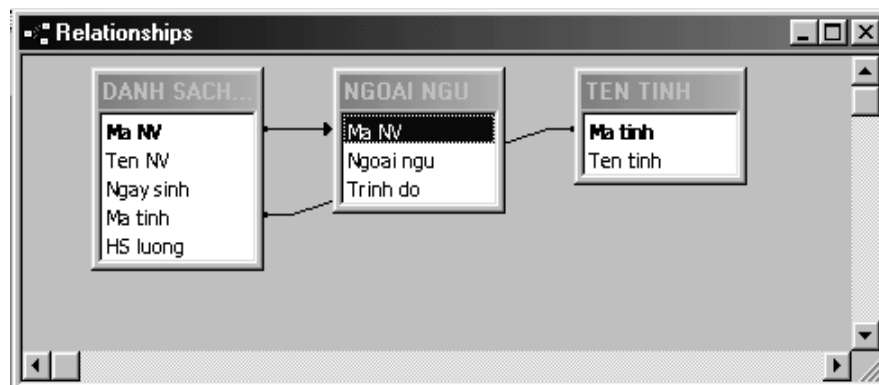
<b>DANH SACH...</b> Ma NV Ten NV Ngay sinh Ma tinh HS luong	<b>NGOAI NGU</b> Ma NV Ngoai ngu Trinh do	<b>TEN TINH</b> Ma tinh Ten tinh
--	--	--

- Nhấn chuột tại trường [Ma tinh] của bảng [DANH SACH NHAN VIEN], kéo sang trường [Ma tinh] của bảng [TEN TINH], thả chuột rồi nhấn nút Join Type để chọn kiểu quan hệ:



- Chọn kiểu quan hệ 1, nhấn OK để tạo quan hệ
- Tương tự, nhấn chuột tại trường [Ma NV] của bảng [DANH SACH NHAN VIEN], kéo sang trường [Ma NV] của bảng [NGOAI NGU], thả chuột rồi nhấn nút Join Type để chọn kiểu quan hệ:
- Chọn kiểu quan hệ 2, nhấn OK để tạo quan hệ.

Kết quả có các đường quan hệ giữa các bảng như sau:



### VIII.3. Chỉnh sửa quan hệ.

Có thể kéo bảng tới các vị trí khác để nhận được các đường quan hệ dễ xem và đẹp hơn.

Để xoá một quan hệ vừa lập, ta kích chuột tại đường quan hệ (sẽ thấy đường đậm hơn), rồi bấm phím Delete.

### VIII.4. Tính toàn vẹn tham chiếu

Sau khi tạo mỗi quan hệ xong chúng ta cần phải tạo tính toàn vẹn tham chiếu cho các quan hệ. Với tính chất này các quan hệ sẽ đảm bảo khi có sự thay đổi về mặt dữ liệu trên các bản ghi của bảng chính (bảng bên 1 trong quan hệ) thì các bản ghi liên quan bên bảng quan hệ cũng thay đổi theo, ngoài ra nó đảm bảo mỗi bản ghi thêm mới trong bảng quan hệ thì phải có bản ghi tương ứng (về mặt khoá) trên bảng chính.



Để tạo toàn vẹn tham chiếu ta chọn mục:

- Enforce Referential Integrity: Đảm bảo việc nhập đúng (chỉ nhập được các bản ghi trên bảng quan hệ khi giá trị trường dùng làm khoá liên kết đã có trên bảng chính)

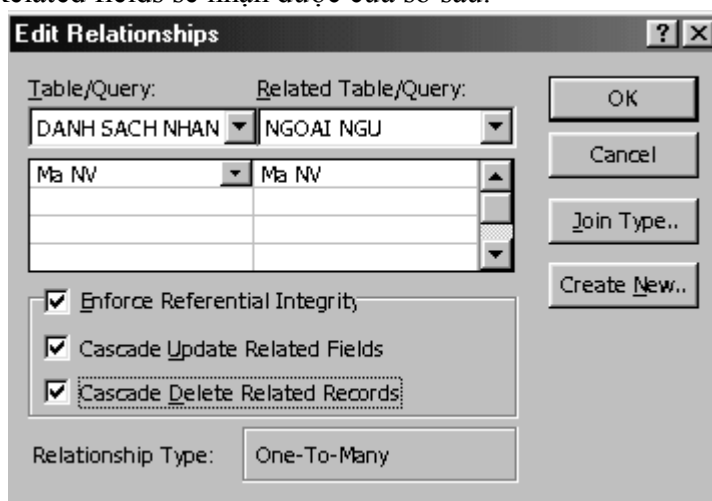
Khi đã đánh dấu Enforce Referential Integrity thì có thể sử dụng thêm các tùy chọn sau:

+ Cascade Update Related fields : Khi sửa giá trị trường khoá trong bảng chính, giá trị tương ứng của các bản ghi trong trường quan hệ sẽ bị sửa theo

+ Cascade Delete Related fields : Khi xoá một bản ghi trong bảng chính, các bản ghi tương ứng trong bảng quan hệ sẽ bị xoá.

**Ví dụ:**

Khi thiết lập quan hệ giữa bảng [DANH SACH NHAN VIEN] và bảng [NGOAI NGU], nếu chọn các nút Enforce Referential Integrity, Cascade Update Related fields, và Cascade Delete Related fields sẽ nhận được cửa sổ sau:



Ý nghĩa: Khi xoá, sửa các bản ghi trong bảng chính, Access sẽ điều chỉnh bảng quan hệ để tính vẹn toàn tham chiếu được thoả mãn, cụ thể:

- Cascade Update: Khi sửa giá trị trường khoá trong bảng chính, giá trị tương ứng của các bản ghi trong trường quan hệ sẽ bị sửa theo. Ví dụ, khi thay đổi mã nhân viên trong bảng [DANH SACH NHAN SU] thì mã nhân viên tương ứng trong bảng [NGOAI NGU] cũng bị sửa theo.

- Cascade Delete: Khi xoá một bản ghi trong bảng chính, các bản ghi tương ứng trong trường quan hệ sẽ bị xoá theo. Ví dụ, khi xoá một bản ghi trong bảng [DANH SACH NHAN SU] thì các bản ghi tương ứng trong bảng [NGOAI NGU] cũng bị xoá theo.

*Chú ý:* Mọi quan hệ giữa các bảng có thể thiết lập sau này khi thiết kế một truy vấn từ nhiều bảng.

## **VIII.5. Xem và điều chỉnh các quan hệ đã có trong CSDL.**

### ***VIII.5.1. Xem các quan hệ đã có***

Từ menu Tools chọn RelationShips

Từ RelationShips chọn Show All để xem tất cả quan hệ

Muốn xem các quan hệ của một bảng, chọn bảng đó, chọn Show Direct từ RelationShips.

### ***VIII.5.2. Chỉnh sửa***

Chọn quan hệ cần sửa → Bấm nút phải chuột → Chọn mục Edit Relationship để mở cửa sổ Relationships

Tiến hành chỉnh sửa trong cửa sổ này (như chọn kiểu quan hệ, chọn tính toàn vẹn tham chiếu,...)

### ***VIII.5.3. Xoá***

Cho hiện các quan hệ

Chọn quan hệ cần xoá (di chuột đến và bấm)

Bấm phím Delete hoặc chọn Delete từ menu Edit

## **IX. THUỘC TÍNH CỦA BẢNG**

\* Tác dụng: áp dụng trên toàn bảng, toàn bộ mẫu tin.

### **IX.1. Các thuộc tính của bảng.**

Thuộc tính	Công dụng
Description	Mô tả những nét chung của bảng
Validation Rule	Access kiểm tra quy tắc (điều kiện) này trước khi cho nhập một mẫu tin vào bảng
Validation Text	Thông báo lỗi khi một bản ghi vi phạm qui tắc

### **IX.2. Đặt thuộc tính.**

1. Mở bảng trong chế độ Design View.

2. Từ menu View chọn Table Properties để hiển thị các dòng thuộc tính của bảng.
3. Đặt điều kiện sau vào dòng thuộc tính Validation Rule:

Ví dụ: Validation Rule: [NG\_VL] > [NG\_SINH] and [TUOI] <= 60 (phóng to thu nhỏ bằng Shift F12)

Với điều kiện trên, khi nhập dữ liệu Access sẽ kiểm tra các lỗi: ngày sinh sau ngày vào làm việc và tuổi không quá 60.

4. Chọn Save từ menu File để ghi các thay đổi.

## CHƯƠNG III. TRUY VẤN (QUERY)

Truy vấn là một công cụ mạnh của Access dùng để: Tổng hợp, sắp xếp và tìm kiếm dữ liệu.

### I. CÁC LOẠI TRUY VẤN

Loại thông dụng nhất là Select Query (truy vấn chọn) với các khả năng như:

- Chọn bảng, query khác làm nguồn dữ liệu.
- Chọn các trường hiển thị.
- Thêm các trường mới là kết quả thực hiện các phép tính trên các trường của bảng nguồn.
- Đưa vào các điều kiện tìm kiếm, lựa chọn.
- Đưa vào các trường dùng để sắp xếp.

Sau khi truy vấn thực hiện, dữ liệu rút ra được tập hợp vào một bảng kết quả gọi là Dynaset, nó hoạt động như một bảng. Mỗi lần mở truy vấn, Access lại tạo một Dynaset gồm kết quả mới nhất của các bảng nguồn.

Có thể chỉnh sửa, xoá, bổ sung thông tin vào các bảng nguồn thông qua Dynaset.

Ngoài truy vấn Select Query ra, còn có các loại truy vấn khác là:

- Crosstab Query: Thể hiện dữ liệu dạng hàng cột.
- Action Query: Tạo bảng mới, thêm, xoá, sửa các mẫu tin trong bảng.
- Union Query: Kết hợp các trường tương ứng từ hai hay nhiều bảng.
- Data-definition Query: Truy vấn được xây dựng từ một câu lệnh SQL dùng để tạo mới hoặc thay đổi cấu trúc của bảng.

### II. CÁC BƯỚC CHÍNH ĐỂ TẠO (THIẾT KẾ) MỘT TRUY VẤN MỚI

Truy vấn thực chất là một câu lệnh SQL được xây dựng nhờ công cụ của Access dùng để tổng hợp dữ liệu từ các bảng nguồn. Khi thực hiện truy vấn, ta nhận được bảng tổng hợp gọi là bảng Dynaset. Dynaset không phải là bảng, nó không được ghi vào đĩa và nó sẽ bị xoá ngay sau khi kết thúc (đóng) truy vấn.

Truy vấn là một công cụ mạnh của Access dùng để: Tổng hợp, sắp xếp và tìm kiếm dữ liệu.

Một truy vấn đã tạo có thể sử dụng để tạo các truy vấn khác (cũng như thể một thủ tục có thể được sử dụng để xây dựng thủ tục khác).

Các bước chính để xây dựng một truy vấn gồm:

1. Chọn nguồn dữ liệu cho truy vấn mới, gồm: các bảng và các truy vấn đã được tạo từ trước .

2. Tạo lập quan hệ giữa các bảng, truy vấn nguồn.

3. Chọn các trường từ các bảng, truy vấn nguồn để đưa vào truy vấn mới.

4. Đưa vào các điều kiện để chọn lọc các mẫu tin thoả mãn các điều kiện đưa vào. Nếu không đưa vào các điều kiện để chọn lọc, thì kết quả (Dynaset) của truy vấn bao gồm tất cả các bản ghi từ các bảng, truy vấn nguồn.

5. Chọn các trường dùng để sắp xếp các mẫu tin trong Dynaset. Nếu không chọn trường sắp xếp, thì các mẫu tin trong Dynaset được hiển thị theo thứ tự như trong các bảng nguồn.

6. Xây dựng các cột (trường) mới từ các trường đã có trong bảng, truy vấn nguồn. Ví dụ: Trường mới là tổng, hiệu, tích, thương của hai trường có sẵn nào đó. Khi xây dựng trường mới, có thể sử dụng các phép toán, các hàm chuẩn của Access và các hàm tự lập bằng ngôn ngữ Access Basic .

### **III- TẠO QUERY BẰNG DESIGN VIEW**

Trong phần này, ta xét một ví dụ điển hình và thực hiện từng bước tạo một truy vấn thông qua ví dụ đó:

Giả sử trong CSDL TSINH có 3 bảng:

- Bảng DSTS (danh sách thí sinh) gồm các trường:
  - HO TEN: kiểu Text
  - SOBD (Số báo danh): kiểu Integer
  - NAM SINH: Kiểu Date/time
  - TEN TINH (tỉnh quê quán của thí sinh): kiểu Text
- Bảng BD\_PHACH (báo danh và phách) gồm các trường:
  - SOBD (Số báo danh): kiểu Integer
  - PHACH (số phách): kiểu Integer

■ Bảng PH\_DIEM (phách và điểm) gồm các trường:

- PHACH (số phách): kiểu Integer
- DTOAN (điểm toán): kiểu Double
- DLY (điểm lý): kiểu Double
- DHOA (điểm hoá): kiểu Double

Yêu cầu xây dựng một bảng tổng hợp từ 3 bảng trên gồm các trường:

- HO TEN
- SOBD
- TEN TINH
- DTOAN
- DLY
- DHOA
- TONG DIEM

Trường TONG DIEM là trường mới, là tổng các trường DTOAN, DLY, và DHOA.

Bảng tổng hợp chỉ gồm các thí sinh có tổng điểm lớn hơn hoặc bằng điểm chuẩn (giả sử điểm chuẩn là 16), sắp xếp theo chiều tăng của trường TEN TINH và thứ tự giảm của trường TONG DIEM.

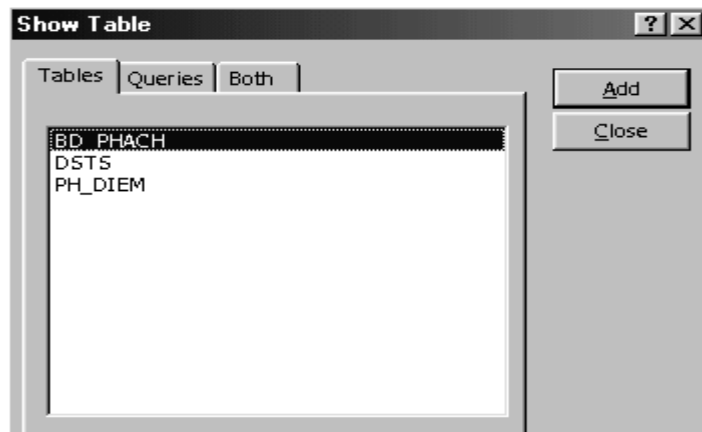
Để giải quyết bài toán trên, ta xây dựng truy vấn KET QUA TS (kết quả tuyển sinh) theo từng bước.

### III. 1-Tạo Select Query

#### III.1.1. Tạo Select Query dùng để chọn các trường, các bản ghi và tính toán

**Bước 1:** Mở cửa sổ Query Designer

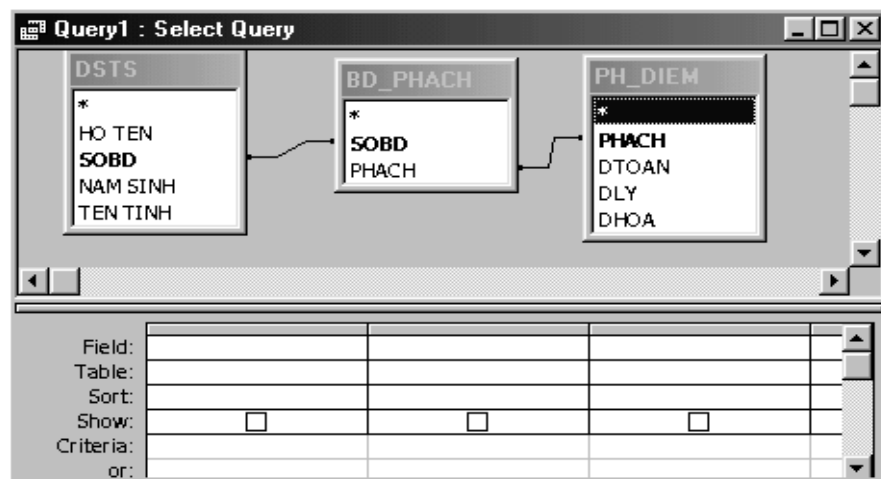
- Từ Database Windows, chọn Queries Tab, chọn New
- Chọn Design View, OK
- Đưa các bảng hay truy vấn hoặc cả hai vào truy vấn từ hộp thoại Show Table:



. Chọn tên các bảng: DSTS, BD\_PHACH, PH\_DIEM, sau đó nhấn nút Add

. Close : Đóng hộp thoại Show Table

Kết quả nhận được cửa sổ Select query:

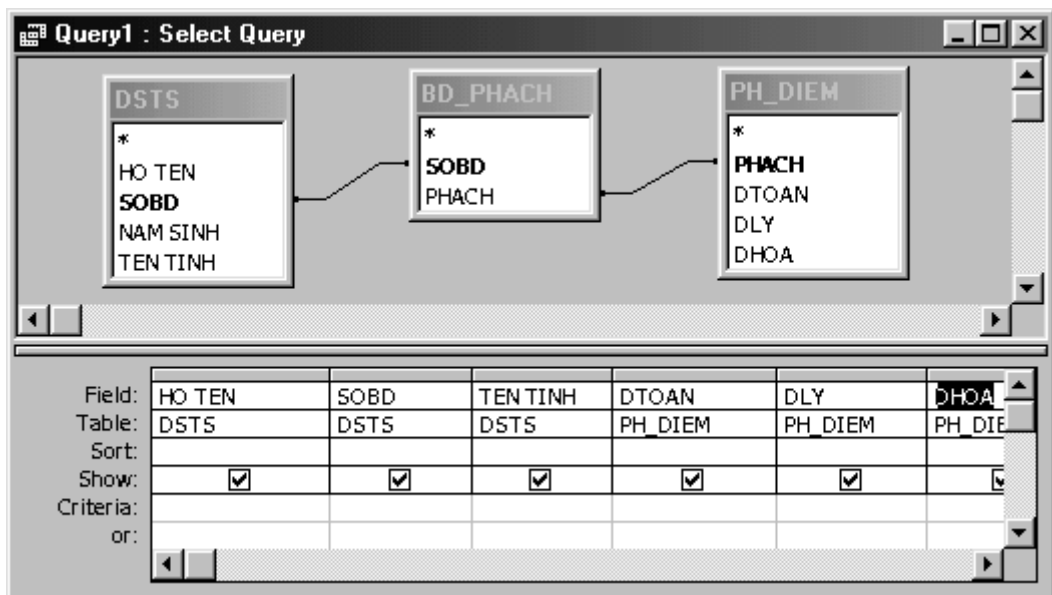


**Bước 2:** Tạo mối quan hệ giữa các bảng:

- Quan hệ 1-1 giữa bảng DSTS và bảng BD\_PHACH theo trường quan hệ SOBD
- Quan hệ 1-1 giữa bảng BD\_PHACH và bảng PH\_DIEM theo trường quan hệ PHACH

**Bước 3:** Chọn các trường đưa vào truy vấn, bằng cách kéo tên trường trong các bảng, truy vấn ở bên trên vào dòng Field ở phần dưới. (Có thể chọn nhiều trường rồi kéo một lần).

Trong ví dụ, ta chọn các trường: HO TEN, SOBD, TEN TINH của bảng DSTS; trường PHACH của bảng BD\_PHACH và trường DTOAN, DLY, DHOA của bảng PH\_DIEM. Kết quả nhận được:



**Bước 4** . Sắp xếp lại, chèn và xoá các trường trong QBE (Phần thiết kế query gọi là QBE), điều chỉnh độ rộng các cột.

■ Di chuyển trường:

- Đưa chuột lên ngay đầu tên trường sao cho hiện ra dấu mũi tên đen quay xuống thì bấm chuột để chọn trường (cả khối trường được chọn sẽ đổi màu đen).
- Đưa chuột vào khối chọn rồi kéo tới vị trí mới.

■ Chèn thêm trường:

- Chọn trường muốn chèn từ danh sách (phần trên)
- Kéo trường đó vào một cột trong vùng QBE (phần dưới)

■ Xóa trường trong vùng QBE

- Chọn trường cần xoá (như khi chọn để di chuyển)
- Bấm phím Delete hoặc chọn Delete từ menu Edit

■ Điều chỉnh độ rộng cột trong vùng QBE: Nếu cột trong vùng QBE quá lớn hay quá nhỏ đối với dữ liệu của trường, ta có thể chỉnh lại kích thước cột cho phù hợp theo các cách sau:

- Đưa chuột đến đầu cạnh bên phải của cột muốn chỉnh sao cho hiện ra mũi tên hai đầu thì kéo đến kích thước mới.



- Chỉnh nhanh độ rộng cột vừa vặn nhất đối với dữ liệu trong cột bằng cách:
  - Bấm kép vào đường chia cột tại đầu bên phải cột cần chỉnh (đưa chuột vào sao cho hiện ra dấu mũi tên hai đầu).
  - Nếu chọn nhiều cột thì chỉ cần bấm đúp chuột theo cách trên ở tại một cột trong vùng chọn. (để chọn nhiều cột có thể dùng phím Shift, hoặc kéo chuột).
  - Khi đã chỉnh xong, nếu ta lại nhập thêm một giá trị dài hơn độ rộng đã được chỉnh, thì ta lại phải chỉnh lại lần nữa vì Access không tự điều chỉnh lại được.

**Bước 5:** Thể hiện tên bảng trong QBE. (Bước này có thể không cần)

Nhằm tạo cho ta cảm giác thuận tiện khi quan sát và thao tác. Để có thêm dòng table (trong QBE) chứa tên bảng/truy vấn nguồn, ta làm như sau: Chọn Table Names từ menu View

Chú ý: Nếu muốn Access tự động thể hiện tên bảng mỗi lần mở truy vấn ở chế độ Design view hoặc tạo truy vấn mới, thì ta chọn Options từ menu Tools, sau đó chọn Tables/Queries trong cửa sổ Options, rồi chọn mục Show Table Names.

**Bước 6:** . Đổi tên trường trong vùng QBE, thêm trường biểu thức.( (Bước này có thể không cần)

Tên mới sẽ được hiện thị trong bảng kết quả (bảng Dynaset) của truy vấn. Như vậy có thể dùng các tên mới có ý nghĩa hơn so với tên trường.

\* Cách đưa vào tên mới: Gõ tên mới ngay trước tên trường. Đặt dấu: giữa tên mới và tên trường ( <Tên trường>:<Tên thay thế>)

Ví dụ: Ta có thể đặt tên mới cho trường [DTOAN], {DLY} và [DHOA] như sau: DIEM TOAN : DTOAN, DIEM LY : DLY, DIEM HOA : DHOA

Chú ý: Trong các hàm, thủ tục (viết bằng Access Basic) sẽ dùng lên mới (không dùng tên trường).

- Cách đưa vào các trường mới (trường biểu thức) : Chọn một ô còn trống trên hàng Field, gõ vào đó tên trường biểu thức, sau đó đặt dấu “:” rồi đưa vào một biểu thức( <Tên trường> : <Biểu thức> ). Trong biểu thức có thể dùng tên

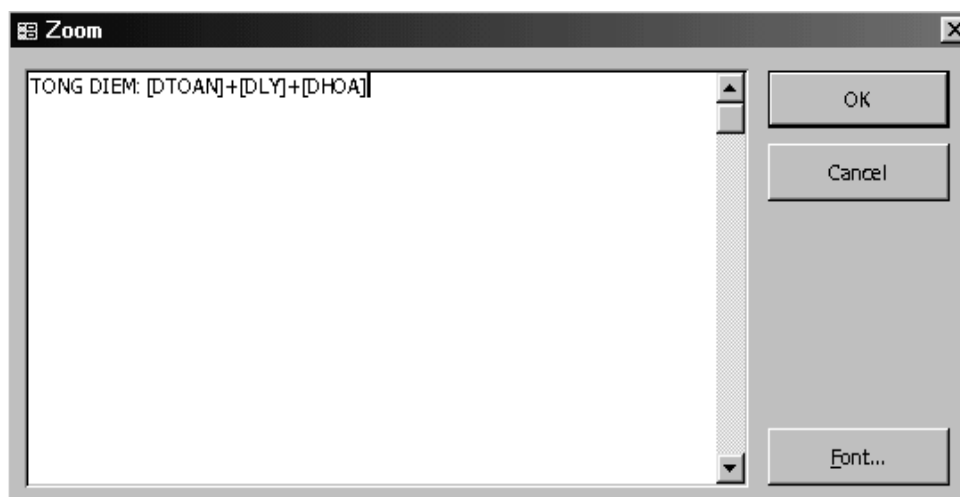
các trường, các hằng, các phép tính, các hàm của chuẩn Access và các hàm tự lập.

Trong ví dụ này ta đưa vào trường mới [TONG DIEM] như sau: TONG DIEM:[DTONAN]+[DLY]+[DHOA]

Chú ý: Khi soạn thảo trên một ô của dòng Field, ta có thể phóng to ô đó bằng cách:

- Chọn ô (trên dòng Field) để soạn thảo.
- Bấm nút chuột phải. Kết quả hiện một hộp có chứa các dòng: Build, Zoom, Properties... → Chọn mục Zoom. Kết quả hiện cửa sổ Zoom.
- Soạn thảo các nội dung cần thiết trong cửa sổ Zoom, rồi bấm nút OK để trở về cửa sổ Select Query.

Dưới đây là cửa sổ Zoom của trường TONG DIEM:



Chú ý:

- Có thể phóng to ô lập biểu thức bằng cách ấn Shift +F2
- Các trường tham gia trong biểu thức phải đặt trong dấu [ ]
- Nếu trường tham gia biểu thức nằm ở bảng khác ta phải chỉ rõ tên bảng chứa trường đó như sau:

*[<Tên bảng>].[<Tên trường>]*

**\*Các hàm thường dùng khi lập biểu thức tính**

- Nhóm hàm văn bản :

+Hàm Left : Lấy n ký tự bên trái chuỗi

=*Left*(<Chuỗi> , n)

+Hàm *Right*: Lấy n kí tự bên phải chuỗi

=*Right*(<Chuỗi> , n)

+Hàm *Mid* : Lấy n kí tự từ vị trí m của chuỗi

=*Mid*(<Chuỗi>, m , n)

- Nhóm hàm ngày tháng

+ Hàm *DatePart*

. C/năm : lấy ra các thành phần của ngày tháng

. Cú pháp : *DatePart*(<Thành phần>, <Ngày tháng>)

. Thành phần ngày tháng đặt trong dấu “ ... “ gồm :

d : Ngày

w : Ngày trong tuần

ww: Tuần

m: Tháng

q : quý

yyyy : Năm

+ Hàm *Date*() : Ngày -tháng- năm hiện tại

*Day*(<Ngày tháng>) : lấy ra ngày

*Month*(<Ngày tháng>) : lấy ra tháng

*Year*(<Ngày tháng>) : lấy ra năm

- Hàm điều kiện (*IIF*)

+ C/năm : lấy ra giá trị thoả mãn tiêu chuẩn

+ Cú pháp trường hợp biểu thức có 2 giá trị để lựa chọn :

*IIF*(<ĐK>, <Giá trị 1>, <Giá trị 2>)

ĐK là một biểu thức logic cho kết quả đúng (True) or sai (False):

Nếu ĐK đúng (True): lấy giá trị 1, Nếu ĐK sai (False): lấy giá trị 2

Giá trị lấy ra có thể là : Số, biểu thức, chuỗi kí tự đặt trong dấu nháy kép (“ “)

Trong trường hợp biểu thức có n lựa chọn ta phải dùng các hàm *IIF* lồng nhau.

**Bước 7:** Định thứ tự sắp xếp.

Mục đích là để cho bảng kết quả hiện ra theo thứ tự nào đó.

*Thao tác:*

- Nhấn chuột vào ô Sort của trường cần sắp xếp
- Chọn Chiều Ascending (giảm dần) hoặc Descending (tăng dần).
- Nếu sắp theo nhiều trường thì thứ tự ưu tiên từ trái sang phải.

Trong ví dụ này, ta chọn:

- Trường TÊN TINH chiều Ascending
- Trường TONG DIEM chiều Descending

**Bước 8:** Lập tiêu chuẩn lựa chọn (điều kiện tìm kiếm).

Mục đích để chỉ hiện những thông tin cần quan tâm trong bảng kết quả, chẳng hạn như chỉ hiện ra những thí sinh đỗ theo điểm chuẩn nào đó.

*Cách làm:*

- Gõ trực tiếp một biểu thức điều kiện vào ô Criteria của các trường cần đặt điều kiện.
- Hoặc có thể dùng cửa sổ xây dựng biểu thức hoặc cửa sổ Zoom (Cách mở cửa sổ Zoom đã trình bày bên trên). Để mở cửa sổ xây dựng biểu thức cho ô Criteria của trường nào, ta bấm nút phải chuột tại ô Criteria của trường, rồi chọn mục Build.

Trong ví dụ này, ta đưa vào ô Criteria của trường TONG DIEM biểu thức điều kiện:  $\geq 16$

**\* Các phép toán và biểu thức dùng để lập tiêu chuẩn chọn các bản ghi**

- Toán tử so sánh : =; >; <; >=; <=; <>
- Toán tử Between <Giá trị đầu> And <Giá trị cuối>
- Phép toán Logic
  - + AND : hội các ĐK
  - + OR : Tuyến
- Biểu thức kết hợp các chuỗi ký tự: &
- Ký tự thay thế (toán tử Like)
  - + ? : Thay thế cho 1 ký tự
  - + \* : Thay thế cho 1 nhóm ký tự

Ví dụ : Đưa ra DS khách hàng có tên đầu là Công ty

Like “Công ty\* ”

- Biểu thức Null và Not is Null.(đưa ra danh sách trống hoặc không trống).

*Chú ý:* Các điều kiện viết trên cùng dòng Criteria là các điều kiện đồng thời thoả mãn (Đk AND). Các điều kiện viết trên các dòng khác nhau là điều kiện OR.

**Bước 9:** Không thể hiện (ẩn) một số trường trong truy vấn.

Mục đích để chỉ hiện những trường cần quan tâm trong bảng kết quả

Cách làm: Bấm chuột tại ô tích (hình vuông) trong hàng Show của trường cần ẩn để đưa về dạng rỗng . Nếu muốn bỏ ẩn ta bấm chuột lần nữa tại ô này.

**Bước 10.** Thiết lập thuộc tính các trường (trong truy vấn)

Nhằm quy định cách thức hoạt động của truy vấn nói chung.

Có thể đặt thuộc tính cho từng trường hoặc cả truy vấn.

Để mở bảng thuộc tính của trường nào: trước tiên cần chọn trường đó (Bấm chuột tại ô Field của trường cần chọn), sau đó hoặc chọn Properties từ menu View hoặc bất nút phải và mục Properties.

Để mở bảng thuộc tính của truy vấn ta làm như sau: Chọn cả truy vấn (bằng cách bấm chuột bên ngoài QBE và ngoài các bảng/truy vấn nguồn ở phần trên) sau đó tiến hành như khi mở bảng thuộc tính của trường.

Sau khi đã mở bảng thuộc tính (của trường hay truy vấn), ta tiến hành chọn các giá trị thích hợp cho các thuộc tính (như khi đặt thuộc tính cho các trường của bảng).

■ Thuộc tính của truy vấn và bảng/truy vấn nguồn Theo mặc nhiên trường trong truy vấn kế thừa các thuộc tính trong bảng/truy vấn nguồn. Vì vậy nếu không đặt lại các thuộc tính cho một trường trong truy vấn, thì trường sẽ có các thuộc tính như trong bảng/truy vấn nguồn.

**Bước 11.** Các dạng hiển thị truy vấn. Truy vấn có thể hiển thị theo các dạng sau :

- Design View (dạng thiết kế)
- SQL View (câu lệnh SQL tương ứng)
- Datasheet View (Bảng kết quả của truy vấn)

Khi đang thiết kế truy vấn (truy vấn đang ở dạng Query Design) ta có thể chuyển sang các dạng khác bằng cách:

+ Chọn menu View. Kết quả nhận được các mục của menu View mà 3 mục đầu là tiêu đề 3 dạng nói trên. → Muốn xem truy vấn ở dạng nào ta chọn mục tương ứng với dạng đó

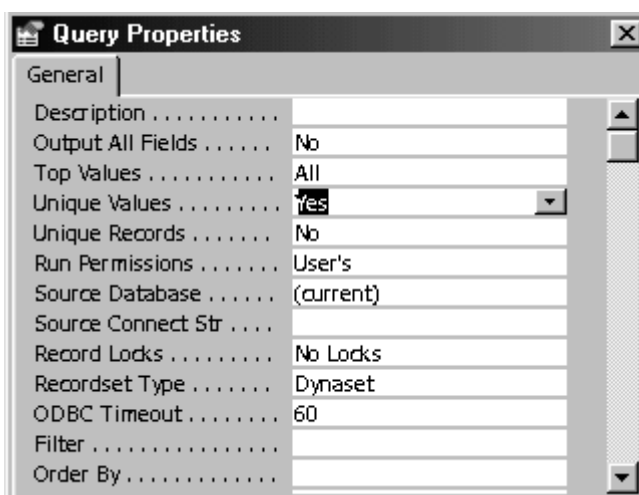
**Bước 12.** Ghi truy vấn Sau khi đã hoàn chỉnh việc thiết kế cần ghi cấu trúc của truy vấn lên (ra. Cách làm: Dùng Save từ menu File, hoặc bấm chuột tại biểu tượng ghi trên thanh công cụ . Access sẽ yêu cầu đặt tên với lần ghi đầu tiên.

**Bước 13.** Ra khỏi cửa sổ thiết kế truy vấn để trở về cửa sổ Database Cách làm: Chọn Close từ menu File hoặc đóng cửa sổ thiết kế truy vấn

**Bước 14.** Thực hiện truy vấn đã có. Ta làm như sau: Chọn truy vấn cần thực hiện (trong cửa sổ Queries) rồi bấm nút Open. Khi đó sẽ nhận được bảng kết quả (bảng Dataset) của truy vấn. Ta có thể xem, sửa và bổ sung dữ liệu trên bảng Dataset.

**Bước 15.** Hiển thị các mẫu tin không trùng lặp

Khi một truy vấn cần đưa ra các bản ghi mà không có các bản ghi trùng lặp về giá trị thì ta chọn thuộc tính Unique Values của truy vấn và đặt giá trị Yes cho thuộc tính này:



### III.1.2. Tạo Select Query dùng để tính tổng theo nhóm dữ liệu.

\* Chức năng: cho phép thiết lập câu hỏi với nhóm dữ liệu

Ví dụ:

- Công ty nhận được bao nhiêu đơn đặt hàng trong tháng
- Giá trung bình các mặt hàng trong nhóm điện tử là ?
- Số lượng bán ra trong tuần của mỗi mặt hàng ?

\* Các hàm thường dùng :

. SUM : Tính tổng các giá trị trong trường

- . AVERAGE : Tính TB cộng các giá trị trong trường.
- . MAX : Tính giá trị lớn nhất trong trường
- . MIN : Tính giá trị nhỏ nhất trong trường
- . COUNT : Đếm số giá trị khác rỗng trong trường
- . STDEV : Tính độ lệch chuẩn các giá trị trong trường
- . First : Tính giá trị đầu tiên trong trường
- . Last : Tính giá trị cuối trong trường

....

**\* Tính tổng tất cả các giá trị trong trường.**

- >Tạo truy vấn mới
- > Đưa các bảng cần thiết vào truy vấn
- > Mở View / Totals hoặc kích nút Totals trên thanh toolbar
- > Đặt các tùy chọn :
  - . Field : Các trường cần thiết
  - . Total : Chọn hàm cần tính tương ứng với mỗi trường cần tính
- > Thực hiện truy vấn : Mở Query / Run

Ví dụ : Tính giá Tb và tính tổng số mặt hàng

**\* Tính tổng theo từng nhóm DL**

- >Tạo truy vấn mới
- > Đưa các bảng cần thiết vào truy vấn
- > Mở View / Totals hoặc kích nút Totals / toolbar
- > Đặt các tùy chọn :
  - .Trên dòng Field : đặt các trường cần thiết
  - . Trên dòng Total :
    - Chọn Group By : Phân nhóm và sắp xếp các mẫu tin trong từng nhóm theo các trường phân nhóm. Nhóm là dãy các bản ghi có giá trị như nhau trên các trường phân nhóm ( Chú ý: Luôn luôn sắp xếp theo chiều tăng, nếu thay đổi thì chọn phần Sort và sắp xếp lại).
    - Chọn hàm cần tính tương ứng với mỗi trường cần tính. Thực hiện phép tính theo từng nhóm trên các trường tính toán (có hàm Sum, Avg, ...) trên ô Total.
    - Chọn Expression ứng với trường biểu thức (nếu có)
    - Chọn Where ứng với trường lấy tiêu chuẩn

. Criteria : Lập tiêu chuẩn chọn nhóm DL cần tính

**Chú ý:** Điều kiện, tiêu chuẩn tham gia phân nhóm và tính tổng:

- Có chữ Where trên ô Total
- Có biểu thức điều kiện trên ô Criteria

### III.1.3 Tạo Select Query dùng hỏi -đáp dữ liệu

- > Tạo truy vấn mới
- > Đưa các bảng cần thiết vào truy vấn
- > Mở View / Totals hoặc kích nút Totals trên toolbar
- > Chọn các trường cần thiết trên dòng Field
- > Nhập các câu hỏi với dữ liệu ứng với trường cần hỏi trên dòng Criteria (khi chạy Query này sẽ hiện lên câu hỏi cho người sử dụng nhập dữ liệu vào, dữ liệu được nhập vào thay cho điều kiện lọc dữ liệu trên query)

Chú ý :

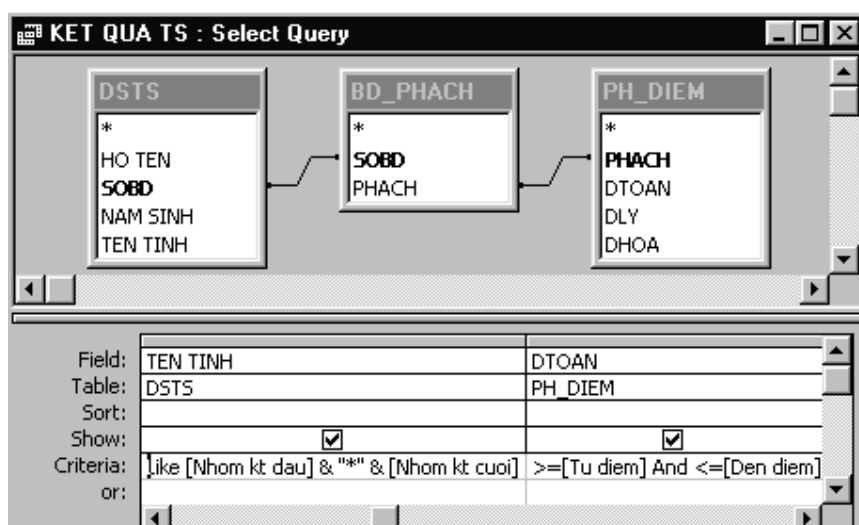
- Câu hỏi tùy đặt phải đặt trong dấu [ ... ] tối đa 255 kí tự
- Có thể đưa biểu thức hỏi kèm cùng các phép toán, biểu thức đã nói ở trên để tạo ra một điều kiện cụ thể:

**Ví dụ:** Cần tạo truy vấn từ các bảng CSDL TSINH để tìm các thí sinh theo các điều kiện:

- ✓ Nhóm ký tự đầu của tên tỉnh
- ✓ Nhóm ký tự cuối của tên tỉnh
- ✓ Giá trị nhỏ nhất của điểm toán
- ✓ Giá trị lớn nhất của điểm toán

Các bước thiết kế truy vấn:

- Bước 1: Chọn 3 bảng và đưa các trường HO TEN, SOBD, TEN TINH, DTOAN, DLY, DHOA vào truy vấn.
- Bước 2: Đưa thêm trường TONG DIEM.
- Bước 3: Đưa vào ô Criteria của trường TEN TINH nội dung:  
Like [Nhóm ký tự đầu] & "\*" & [Nhóm ký tự cuối]
- Bước 4: Đưa vào ô Criteria của trường DTOAN nội dung:  
>=[Từ điểm] And <=[Đến điểm], cửa sổ truy vấn khi đó có dạng:





Khi thực hiện truy vấn trên, Access sẽ lần lượt yêu cầu dữ vào 4 giá trị:

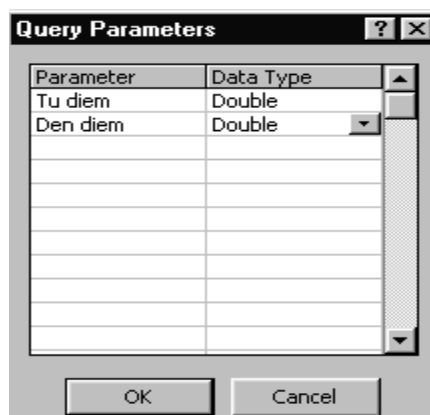
- ✓ Nhóm ký tự đầu của tên tỉnh
- ✓ Nhóm ký tự cuối của tên tỉnh
- ✓ Giá trị nhỏ nhất của điểm toán
- ✓ Giá trị lớn nhất của điểm toán

Sau đó sẽ đưa ra danh sách các thí sinh thoả mãn các yêu cầu cần tìm.

### **\* Quy định kiểu dữ liệu cho các thông số**

Để qui định kiểu dữ liệu cho các thông số, ta làm như sau:

1. Mở truy vấn thông số ở chế độ Design View
2. Chọn mục Parameters từ menu Query để mở cửa sổ Query Parameters
3. Lần lượt đưa vào tên thông số và chọn kiểu dữ liệu thích hợp. Ví dụ, chọn kiểu Double cho các thông số “Từ điểm” và “Đến điểm”:



4. Chọn OK

## **II.2. Tạo Crosstab Query**

\*Tạo truy vấn mới

- > Đưa các bảng cần thiết vào truy vấn
- > Mở Query / Crosstab Query
- > Chọn các trường cần thiết trên dòng Field

-> Trên dòng Total

- . Chọn Group ứng với trường cần nhóm
- . Chọn hàm tính toán ứng với trường cần tính
- . Chọn Expression ứng với trường biểu thức
- . Chọn Where ứng với trường lấy điều kiện

-> Trên dòng Crosstab:

- . Chọn Row Heading ứng với trường lấy làm tiêu đề dòng
- . Chọn Column Heading ứng với trường lấy làm tiêu đề cột

. Chọn Value ứng với trường tính giá trị

-> Thực hiện Query : Mở Query / Run

-> Ghi truy vấn

### **III.3. Tạo Action Query**

#### **III.3.1. Make Table Query (Truy vấn tạo bảng)**

*B1:* Tạo truy vấn mới

*B2:* - Mở Query / *Make Table Query*

-Đặt tên bảng dữ liệu mới vào dòng Table name, chọn OK

*B3:* Thiết kế truy vấn

-> Chọn các trường cần lấy ở các ô trên dòng Field

-> Chọn cách sắp xếp Dữ liệu tương ứng trên dòng Sort

-> Lập tiêu chuẩn chọn các bản ghi ở ô trên dòng Criteria

->Ghi và thực hiện Query

#### **III.3.2. Append Query (Truy vấn nối)**

*B1:* Tạo truy vấn mới

*B2:* Mở Query / *Append Query*

-> Chọn tên bảng nhận dữ liệu (đích) ở dòng Table name, chọn OK

*B3:* *Thiết kế truy vấn*

-> Chọn các trường cần lấy ở các ô trên dòng Field

-> Chọn cách sắp xếp Dữ liệu tương ứng trên dòng Sort

-> Chọn các trường nhận dữ liệu ở bảng đích trên dòng Append to

-> Lập tiêu chuẩn chọn các bản ghi ở ô trên dòng Criteria

->Ghi và thực hiện Query

#### **III.3.3. Update Query (Truy vấn cập nhật) QBE**

*B1:* Tạo truy vấn mới

*B2:* Mở Query / *Update Query*

*B3:* *Thiết kế truy vấn*

-> Chọn các trường cần thiết ở các ô trên dòng Field

-> Chọn cách sắp xếp Dữ liệu tương ứng trên dòng Sort

-> Lập biểu thức ở ô trên dòng Update to ứng với trường cần cập nhật  
biểu thức : có thể gồm Các giá trị, các trường, các phép toán, các hàm,...

-> Lập tiêu chuẩn chọn các bản ghi ở ô trên dòng Criteria

->Ghi và thực hiện Query

#### **III.3.4. Delete Query (Truy vấn xóa)**

*B1:* Tạo truy vấn mới

*B2:* Mở Query / *Delete Query*

### *B3: Thiết kế truy vấn*

-> Chọn các trường cần thiết ở các ô trên dòng Field

-> Lập tiêu chuẩn chọn các bản ghi cần xoá ở ô trên dòng Criteria ứng với trường lấy tiêu chuẩn.

-> Ghi và thực hiện Query

## **IV. TẠO QUERY BẰNG WIZARD**

### **IV.1. Tạo Select Query bằng wizard**

\* Chọn Queries tab, chọn New

\* Chọn Simple Query Wizard

\* Thực hiện các bước theo chỉ dẫn của hộp thoại Simple Query Wizard

B1: Chọn tên bảng và các trường cần thiết, chọn Next

B2: Chọn chế độ

-Detail : Lấy thông tin chi tiết

-Summary : Tóm tắt dữ liệu theo nhóm

B3 : đặt tên query, Chọn Finish

### **IV.2. Tạo Crosstab Query bằng wizard**

\* Chọn Queries tab, chọn New

\* Chọn Crosstab Query Wizard

\* Thực hiện các bước theo chỉ dẫn của hộp thoại Crosstab Query Wizard

B1: Chọn bảng dữ liệu nguồn, chọn Next

B2: Chọn trường lấy làm tiêu đề dòng(Row heading ), chọn Next

B3: Chọn trường lấy làm tiêu đề cột(Column Heading), chọn Next

B4 : Chọn trường cần tính giá trị (Value) và hàm cần tính, chọn Next

B5 : Đặt tên Query, chọn Finish

## **V- TẠO, SỬA QUERY BẰNG MÃ LỆNH SQL**

### **V.1-Cấu trúc các câu lệnh tạo Query**

```
SELECT Tên bảng.Trường1, Tên bảng.trường 2,...
```

```
FROM Tên bảng dữ liệu
```

```
WHERE Điều kiện;
```

Ví dụ : Cho bảng LILICH(MANV, HOTEN, GIOITINH)

Tạo truy vấn đưa ra danh sách các nhân viên Nam ta viết như sau:

```
SELECT LILICH.MANV, LILICH.HOTEN, LILICH.GIOITINH  
FROM LILICH
```

```
WHERE (((LILICH.GIOITINH)="yes"));
```

## V.2.Cách tạo và sửa Query bằng SQL

\* Chọn New từ Query tab

\* Đưa bảng hay truy vấn vào truy vấn

\* Mở View, chọn SQL View

Ta có cửa sổ, trên cửa sổ đó:

-> Viết các câu lệnh chọn các trường cần thiết và điều kiện chọn bản ghi theo cấu trúc trên

-> Ghi và thực hiện truy vấn

Ví dụ1 : Cho bảng DIEM(MASV,TIN,ANH,KT)

Đưa ra danh sách những sinh viên có điểm tin  $\geq 5$  và điểm Anh  $\geq 0$  và điểm Kt  $\geq 5$  ta viết như sau:

```
SELECT DIEM.masv, DIEM.Tin, DIEM.Anh, DIEM.KT
```

```
FROM DIEM
```

```
WHERE (((DIEM.Tin) $\geq 5$ ) AND ((DIEM.Anh) $\geq 5$ ) AND ((DIEM.KT) $\geq 5$ ));
```

Ví dụ2 : Cho bảng DIEM(MASV,TIN,ANH,KT)

Đưa ra danh sách những sinh viên có điểm tin  $\geq 5$  hoặc Tổng điểm  $\geq 18$  ta viết như sau:

```
SELECT DIEM.MaSV, DIEM.Tin, DIEM.Anh, DIEM.KT, [tin]+[anh]+[kt] AS  
Tongdiem
```

```
FROM DIEM
```

```
WHERE (((DIEM.Tin) $\geq 5$ ) OR ((([tin]+[anh]+[kt]) $\geq 18$ )));
```

## CHƯƠNG 5

### MẪU BIỂU

Mẫu biểu là phương tiện giao tiếp cơ bản giữa người dùng và Access. Thông tin trong biểu mẫu được lấy từ bảng và có thể thiết kế biểu mẫu cho nhiều mục đích khác nhau như:

- Hiện thị và chỉnh sửa dữ liệu: sử dụng biểu mẫu để thay đổi, bổ sung, xoá dữ liệu.
- Nhập dữ liệu: Có thể thiết kế biểu mẫu dùng để nhập dữ liệu mới vào bảng
- Và còn nhiều mục đích khác.

#### I. KHÁI NIỆM CHUNG

##### I.1. Cấu trúc của mẫu biểu

Mẫu biểu (Form) gồm các ô điều khiển. Có nhiều loại ô điều khiển như:

- Hộp văn bản (Text Box)
- Nhãn (Label)
- Nút lệnh (Command Button).
- Hộp lựa chọn (Combo Box)
- Hộp danh sách (List Box)

##### I.2. Công dụng của các ô điều khiển:

###### I.2.1. Thể hiện dữ liệu.

Mẫu biểu có thể gắn với một bảng/truy vấn nguồn hoặc không gắn với một bảng/truy vấn nào cả (mẫu biểu Unbound). Các ô điều khiển có thể buộc vào (liên kết với) các trường của bảng/truy vấn nguồn để thể hiện dữ liệu và cập nhật dữ liệu vào các trường này. Có thể dùng hộp văn bản để hiển thị kết quả tính toán.

###### I.2.2. Nhập dữ liệu từ bàn phím

Có thể nhập dữ liệu từ bàn phím thông qua các hộp văn bản.

###### I.2.3. Thực hiện hành động.

Các nút lệnh có thể gắn với một Macro hoặc một thủ tục xử lý sự kiện. Sau đó mỗi khi bấm chuột vào nút lệnh thì Macro hoặc thủ tục gắn với nó sẽ được thực hiện. Có thể sử dụng các nút lệnh để có các menu đơn giản cho chương trình.

###### I.2.4. Tổ chức giao diện chương trình.

Giao diện chương trình có thể được tổ chức dưới dạng một mẫu biểu. Tiêu đề chương trình và các hướng dẫn sử dụng có thể đưa vào các nhãn. Trong mẫu biểu các thể kéo vào các hình ảnh để trang trí.

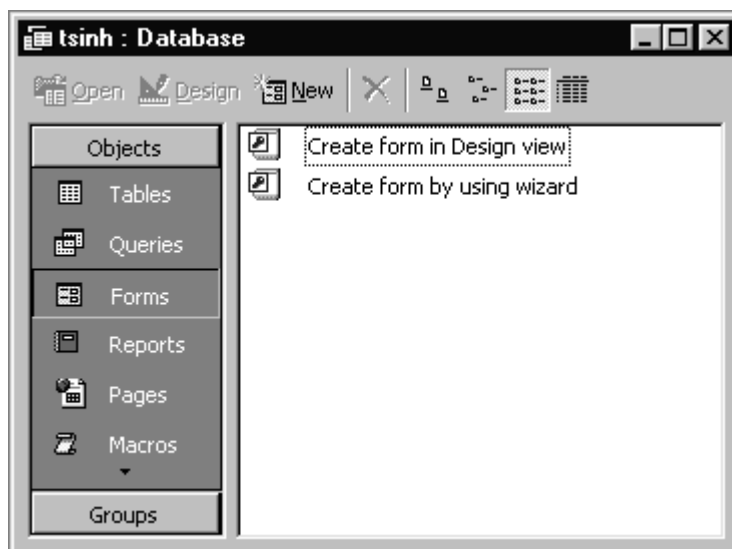
###### I.2.5. Tổ chức hệ Menu Bar cho chương trình.

Hệ Menu Bar gồm các menu ngang (nằm trên hàng ngang ở đầu màn hình). Mỗi menu ngang gồm các menu thành phần. Menu thành phần có thể là chức năng thực hiện ngay hoặc lại có thể là một menu khác.

## II. TẠO MẪU BIỂU TỰ ĐỘNG VÀ BẢNG WIZARD

Nếu chỉ dùng biểu mẫu để thể hiện và cập nhật dữ liệu thì cách nhanh nhất là dùng cách tạo tự động hoặc công cụ Wizard. Để tạo mẫu biểu bằng cách tự động hoặc bằng Wizard, ta thực hiện theo các bước sau:

**B1.** Từ cửa sổ Database của một cơ sở dữ liệu nào đó, chọn mục Forms. Kết quả sẽ mở cửa sổ Forms với ba nút lựa chọn như sau:



- Nút New dùng để tạo mẫu biểu mới.
- Nút Open dùng để thực hiện mẫu biểu.
- Nút Design dùng để mở cửa sổ thiết kế mẫu biểu.

**B2.** Để tạo mẫu biểu mới ta chọn nút New, kết quả nhận được cửa sổ:



**B3.** Chọn một bảng/truy vấn nguồn trong ô: Choose the table... hoặc không chọn bảng/truy vấn nguồn.

**Ví dụ:** Chọn bảng [KHACH HANG] làm nguồn cho mẫu biểu cần xây dựng, bảng này có 4 trường kiểu Text:

- MAKH (Mã khách hàng) - DIA CHI (địa chỉ)
- TEN KH (tên khách hàng) - DIEN THOAI (điện thoại)

**B4.** Chọn một trong các dạng mẫu biểu tự động và Wizard

**B.4.1.** Nếu chọn “Autoform: Columnar” hoặc “AutoForm: Tabular” hoặc “AutoForm: Datasheet” thì được ngay mẫu biểu dạng như sau:

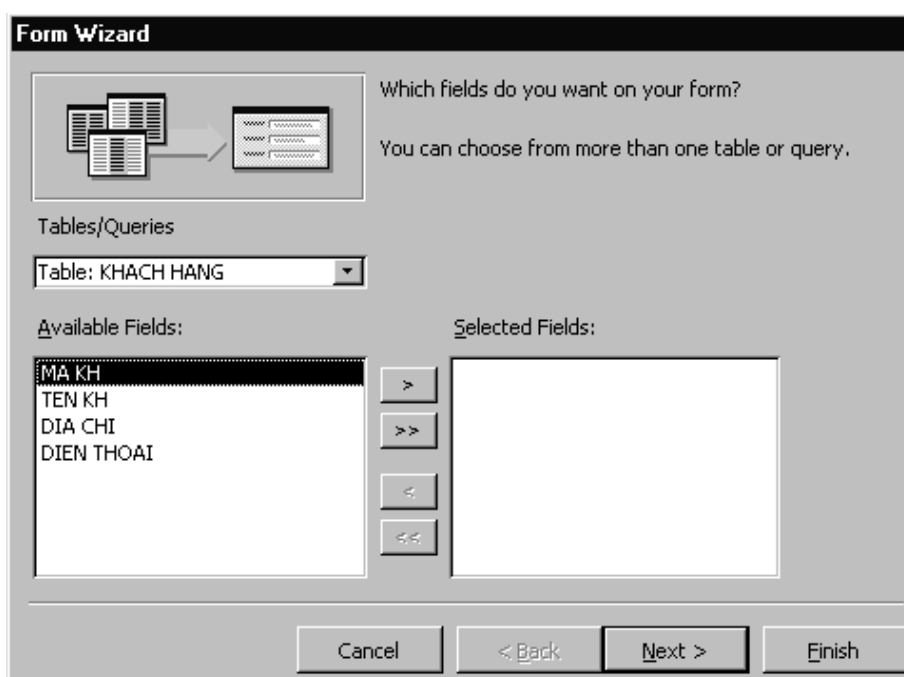


MA KH	01
TEN KH	Lê Tuấn Anh
DIA CHI	Hà Nội
DIEN THOAI	04 8567267

Record: 1 of 17

Trong mẫu biểu sẽ bao gồm tất cả các trường của bảng/truy vấn nguồn và chúng có các cách thức thể hiện (giao diện) khác nhau tùy theo ta chọn mục nào.

**B.4.2.** Nếu chọn Form Wizard sẽ hiện một cửa sổ:



Which fields do you want on your form?  
You can choose from more than one table or query.

Tables/Queries  
Table: KHACH HANG

Available Fields:  
MA KH  
TEN KH  
DIA CHI  
DIEN THOAI

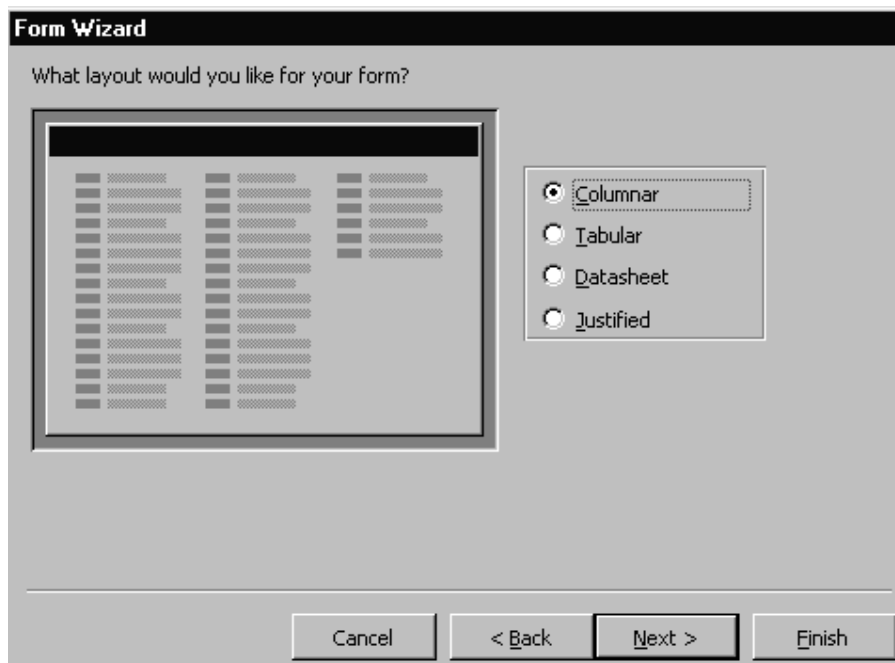
Selected Fields:

Cancel < Back Next > Finish

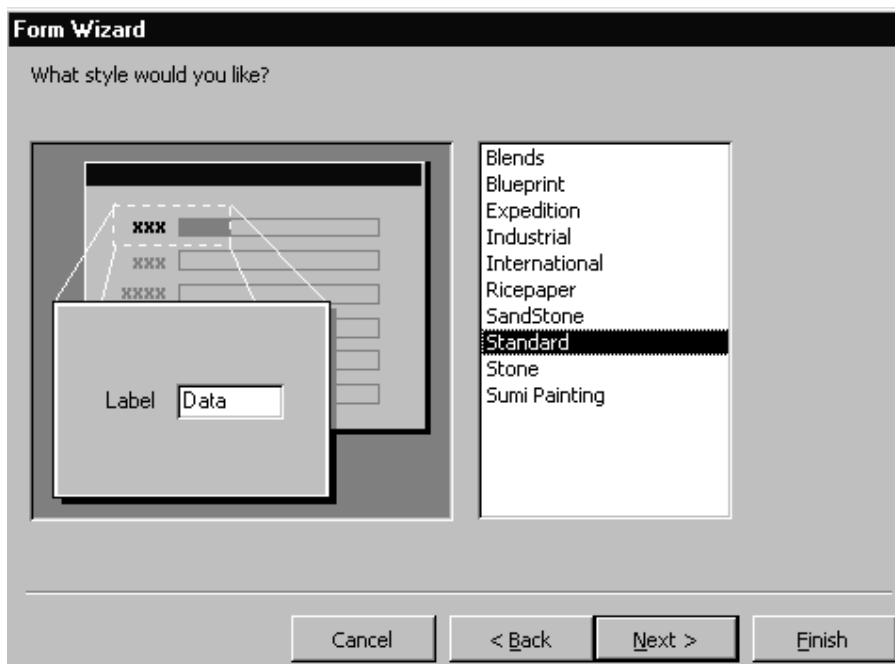
Tại đây ta thao tác tiếp như sau:

Chọn các trường muốn đưa vào mẫu biểu tại cột “Available fields”, rồi bấm vào nút Add để đưa sang cột “Field order on form”. Nếu muốn đưa sang tất cả các trường ta bấm tại nút All. Nếu muốn bỏ trường nào đã chọn trong cột “Field order on form”, thì ta bấm chuột tại trường đó rồi bấm nút remove, còn nếu muốn bỏ tất cả ta bấm vào nút remove all.

Giả sử ta chọn hai trường [TEN KH] và [DIA CHI] rồi chọn nút Next, kết quả hiện cửa sổ:

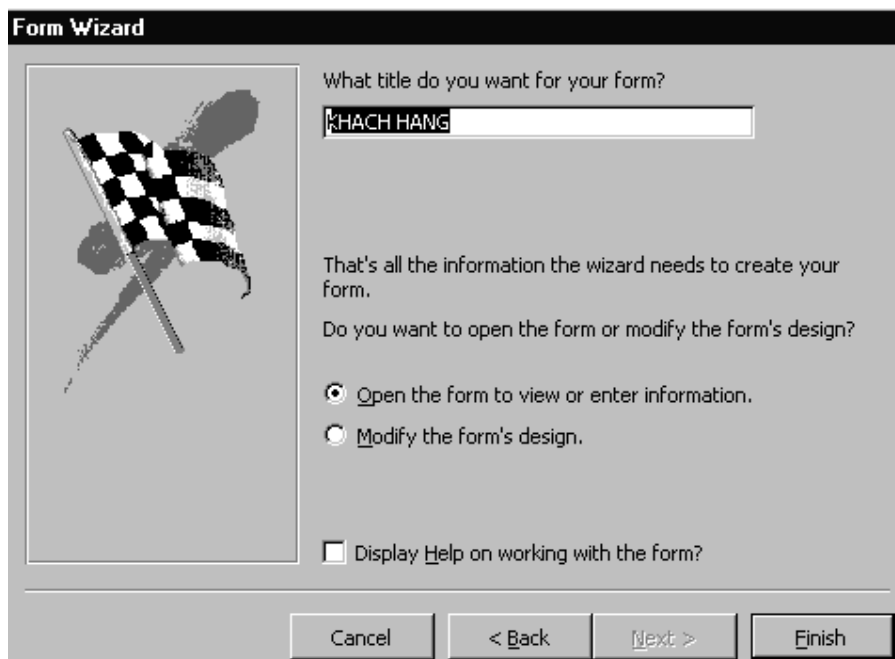


+ Chọn một trong các dạng thể hiện nêu trong cửa sổ trên. Giả sử chọn “Columnar” rồi bấm nút Next. Kết quả sẽ hiện cửa sổ:



+ Chọn một trong các kiểu trình bày mẫu biểu trong cửa sổ trên. Giả sử chọn “Standard” rồi bấm nút Next. Kết quả sẽ hiện cửa sổ:





+ Gõ tên mẫu biểu tại hộp màu trắng. Access dùng tên bảng/truy vấn nguồn làm tên mẫu biểu mặc định, ta có thể giữ nguyên hoặc thay đổi theo ý muốn → Bấm Finish để kết thúc, kết quả nhận được mẫu biểu:



### III. TẠO FORM BẰNG DESIGN VIEW

#### III.1. Giới thiệu các loại điều khiển.

##### III.1.1. Các loại điều khiển.

Khi thiết kế mẫu biểu, thường dùng các loại điều khiển sau:

- Hộp văn bản (text box)
- Nhãn (Label)
- Hộp lựa chọn (Combo box)
- Hộp danh sách (List box)
- Nút lệnh (Command Button)
- Nhóm lựa chọn (Option Group)

##### III.1.2. Cách dùng.

Có thể dùng các ô điều khiển theo ba cách: Bound, Unbound, hoặc Calculated với ý nghĩa:

1. Điều khiển loại Bound (buộc vào một trường nào đó): Dùng để truy xuất tới trường nào đó. Các giá trị cập nhật có thể: Picture, text

2. Điều khiển loại Unbound (không buộc): Không liên quan đến trường nào cả, chúng thường dùng để nhập dữ liệu từ bàn phím, trình bày tiêu đề, trang trí hình vẽ, hộp, đường,...

3. Điều khiển tính toán (Calculated): Dùng để thể hiện giá trị của một biểu thức. Ví dụ:  $GiaMoi = 0.75 * [DON\ GIA]$  Trong đó ô GIA MOI là nhân đi kèm với điều khiển tính toán. Ở đây điều khiển tính toán biểu thị 75% của trường [DON GIA].

*Chú ý:* Điều khiển tính toán này chỉ dùng để hiển thị dữ liệu.

\* **Hộp công cụ:** Hộp công cụ là một bảng chữ nhật chứa nhiều nút có dạng sau:



Mỗi nút trên hộp công cụ ứng với một loại điều khiển được dùng để tạo các ô điều khiển thuộc loại tương ứng. Hộp công cụ được sử dụng như một phương tiện chính để xây dựng các mẫu biểu.

\* **Cách tạo một điều khiển bằng hộp công cụ:**

1. Chọn loại điều khiển: Bấm vào nút tương ứng trên hộp công cụ.
2. Di chuột đến vị trí cần đặt điều khiển vào biểu mẫu và bấm
3. Thực hiện các sửa đổi cần thiết.

### III.2. Điều khiển TextBox

Text box có thể dùng cả 3 cách Bound (bị buộc), Unbound (không bị buộc) hoặc Calculated (tính toán)

#### III.2.1. Tạo Text box bị buộc (Bound)

\* **Cách 1:** Cách đơn giản nhất để tạo các hộp văn bản bị buộc là dùng Filed List.

Thao tác:

**B1:** Từ cửa sổ Database chọn mục Form, kết quả mở cửa sổ Forms trong đó chứa các mẫu biểu đã có. Để tạo một mẫu biểu mới, ta bấm nút New, kết quả xuất hiện hộp Select a Table/Query.

**B2.** Trong hộp Choose the table... chọn một bảng/truy vấn nguồn.

**B3.** Chọn Design View để tự thiết kế mẫu biểu, rồi bấm OK. Kết quả hiện cửa sổ Form và một mẫu biểu rỗng.

**B4.** Chọn View, Field List để mở hộp Field List chứa danh sách các trường của bảng/ truy vấn nguồn. Sau đó kéo các trường xuất hiện trên danh sách vào mẫu biểu, Access sẽ tạo các TextBox bị buộc vào các trường được chọn.

\* **Cách 2:** Dùng hộp công cụ Toolbox, ba bước đầu thực hiện như cách 1, sau đó:

**B1:** Chọn nút Text box trên hộp công cụ (nếu chưa có hộp công cụ thì chọn mục Toolbox trong menu View).

**B2:** Di chuột đến một vị trí trên mẫu biểu, bấm chuột tại đó. Kết quả sẽ hiện ra tại vị trí bấm một hộp văn bản kiểu Unbound và một nhãn đi kèm bên phải có dạng sau: Trong mẫu biểu: Nhãn có tên mặc định bắt đầu bằng Text, trong hộp văn bản có chữ Unbound.

**B3:** Sửa Unbound bằng một tên trường của bảng/truy vấn nguồn. Khi đó hộp văn bản sẽ bị buộc vào trường này. Hoặc:

Mở cửa sổ Properties : View/Properties

Đặt các tính chất sau :

+Name : tên của Textbox

+Control Source : Chọn tên trường dữ liệu buộc vào Textbox

**B4.** Sửa nhãn cho có nghĩa

### **III.2.1. Tạo Text box tính toán - Unbound**

**B1:** Mở hộp Toolbox : View/Toolbox

Chọn công cụ Text box trong Toolbox và tạo một Text box trên form.

**B2:** Mở cửa sổ Properties : View/Properties

Đặt các tính chất sau :

Name : tên của Textbox;

Control Source : Gõ biểu thức cần tính hoặc gõ biểu thức trực tiếp trên TextBox

Chú ý: Nếu tạo TextBox Unbound thì không đặt biểu thức trong Control Source (hoặc để trống TextBox)

### **III.3. Điều khiển Label (nhãn)**

Nhãn luôn luôn là điều khiển Unbound

Điều khiển nhãn có 2 thuộc tính cần nhớ:

- Tiêu đề, sẽ được hiện ra màn hình (Caption)

- Tên (name)

Nhãn thường dùng để chứa các dòng chữ có tính giải thích, ghi chú.

*Chú ý:* Khi muốn soạn thảo nhiều dòng trên nhãn, ta dùng tổ hợp hai phím Ctrl và Enter để chuyển xuống dòng tiếp theo.

Khi thay đổi Font chữ kích cỡ chữ thì nội dung có thể không khớp với kích thước của nhãn. Để điều chỉnh ta làm như sau:

1. Chọn nhãn
2. Chọn Size từ menu Format
3. Chọn mục To fit

\* *Gắn nhãn cho hộp văn bản (Text Box)*

Để gắn cho Text Box một nhãn (giả sử đã xoá nhãn cũ) ta làm như sau:

1. Bấm chuột tại nút nhấn
2. Bấm chuột tại một vùng trống nào đó của Form Hoặc đưa trực tiếp vào vị trí cần thiết – trong trường hợp này không thực hiện bước 3,4,5). Kết quả sẽ tạo ra một nhãn rỗng. → Soạn tiêu đề cho nhãn.
3. Chọn Cut từ menu Edit (để xoá nhãn vừa tạo trên mẫu biểu, đồng thời đưa nó vào Clipboard).
4. Chọn Text box cần gắn nhãn
5. Chọn Paste từ menu Edit.

#### **III.4. Thuộc tính của điều khiển.**

Mỗi đối tượng trong Access có rất nhiều thuộc tính (Property). Access dùng thuộc tính để làm việc với các đối tượng. Trong mục này sẽ trình bày cách sử dụng một số thuộc tính của các ô điều khiển.

##### **III.4.1. Mở bảng thuộc tính của một ô điều khiển**

Để mở bảng thuộc tính của một điều khiển, ta làm như sau:

1. Chọn điều khiển cần mở bảng thuộc tính
2. Hoặc chọn Properties từ menu View - Hoặc bấm đúp tại điều khiển Hoặc bấm nút phải chuột, rồi chọn mục Properties. Kết quả hiện ra bảng các thuộc tính.

##### **III.4.2. Một số thuộc tính hay dùng.**

1. Name: tên điều khiển, dùng trong các hàm, thủ tục.
2. Caption: tiêu đề, thường dùng đối với Label.
3. Control Source: nguồn dữ liệu, thường dùng đối với text box.
4. Format: qui định hình thức hiển thị dữ liệu.
5. Default Value: giá trị mặc định.
6. Validation Rule: qui tắc dữ liệu hợp lệ.
7. Validation Text: văn bản hợp lệ.
8. Input Mask: mặt nạ nhập liệu.
9. Status Bar Text: Hướng dẫn sử dụng điều khiển (hướng dẫn này sẽ hiện tại dòng trạng thái)

#### **III.5. Các loại điều khiển dùng để cập nhập dữ liệu từ bảng chọn**

Các loại điều khiển dùng trong mục đích này gồm:

- Combo Box (Hộp Combo)
- List Box (Hộp danh sách)
- Check Box (Hộp kiểm tra)
- Option Button (Nút lựa chọn)
- Toggle Button (Nút bật tắt)
- Option Group (Nhóm lựa chọn)

Combo Box và List Box có thể dùng để tạo danh sách lựa chọn cho một trường bất kỳ. Danh sách trong List Box luôn luôn được thể hiện. Danh sách trong Combo Box được

mở khi kích chuột tại nút mũi tên bên phải và sau khi chọn một mục trong danh sách thì danh sách tự động đóng lại. Các điều khiển Check Box, Option Button và Toggle Button được dùng để chọn lựa giá trị Yes hoặc No cho trường logic. Điều khiển Option Group được dùng để chọn một số nguyên từ 1 đến N cho trường nguyên.

### **III.5.1. Cách dùng**

Các điều khiển này đều có thể dùng theo hai cách: Unbound và Bound. Khi được buộc vào một trường nào đó của bảng truy vấn nguồn, chúng vừa có tác dụng thể hiện dữ liệu và cập nhật dữ liệu cho trường tương ứng. Khi cập nhật dữ liệu, người dùng không phải nhập từ bàn phím mà dùng chuột để chọn một giá trị nào đó trong bảng danh sách giá trị có sẵn.

### **III.5.2. Cách tạo.**

Các điều khiển này có thể tạo bằng Wizard hoặc không dùng Wizard theo cùng một cách như sau: Trong cửa sổ Form (đang thiết kế)

1. Chọn hoặc không chọn Control Wizards bằng cách bấm vào biểu tượng Control Wizards trên thanh công cụ Toolbox.
2. Chọn loại điều kiện trên Toolbox (List Box, Combo Box, Option Button , . . . )
3. Kéo một trường từ hộp Field list vào mẫu biểu. Bằng cách đó đã tạo được một điều khiển buộc vào một trường. Nếu muốn tạo điều khiển Unbound, thì ta chỉ việc bấm chuột tại một vị trí trên Form.
4. Nếu dùng Wizard thì lần lượt trả lời các câu hỏi, bấm Next, cuối cùng bấm Finish.
5. Nếu không dùng Wizard thì sử dụng các thuộc tính của điều khiển vừa tạo.

### **III.5.3. Phân loại ListBox và Combo Box theo nguồn dữ liệu.**

#### *\* Cấu trúc của List Box và Combo Box*

List box và Combo box có cấu trúc và cách xây dựng hoàn toàn giống nhau. Chúng chỉ khác nhau ở cách thể hiện như đã nói. Mỗi một List box và Combo box có một nguồn dữ liệu để xây dựng bảng chọn. Điểm mấu chốt trong xây dựng List/Combo box là xác định rõ nguồn dữ liệu của chúng.

#### *\* Nguồn dữ liệu của List/Combo box*

Nguồn dữ liệu tạo thành bảng chọn của List/Combo box bao gồm:

1. Các bản ghi của một bảng/truy vấn.
2. Dãy giá trị đưa vào khi tạo List/Combo box (Value list).
3. Tên các trường của một bảng/truy vấn (Field list).
4. Các bản ghi của một truy vấn tạo bởi câu lệnh SELECT.
5. Danh sách tạo từ kết quả trả về của một hàm trong Access Basic.

### **III.5.4. Tạo LIST/COMBO BOX không dùng WIZARD**

Trước hết phải tắt chức năng Control Wizard bằng cách bấm vào biểu tượng Control Wizard .

#### *\* Các thuộc tính sau để xác định nguồn dữ liệu của List/Combo box*

Loại	Thuộc tính	Thuộc tính
------	------------	------------

(Nguồn dữ liệu )	Row Rource type	Row Source
Các bản ghi của một bảng/truy vấn	Table/query	Tên bảng/truy vấn
Các bản ghi từ câu lệnh Select	Table/Query	Câu lệnh SELECT
Danh sách giá trị tự đặt	Value list	Danh sách' giá trị Phân cách nhau bởi chấm phẩy
Tên các trường của một bảng/truy vấn	Field List	Tên bảng truy vấn
Các giá trị trả về của một hàm Access Basic	Tên hàm đó	Để trống

*\* Các thuộc tính khác:*

Ngoài hai thuộc tính quan trọng là Row Rource type và Row Source dùng để xác định nguồn dữ liệu tạo thành bảng chọn của List/Combo box, chúng ta cần sử dụng thêm các thuộc tính sau:

- Column Count: Số cột trong danh sách, tính từ trái sang phải. Ví dụ: Nếu đặt giá trị của thuộc tính này bằng 2, thì bảng chọn sẽ gồm 2 cột chứa giá trị hai trường đầu của nguồn dữ liệu của List/Combo box.
- Bound Column: Cột được chọn làm nguồn dữ liệu của danh sách. Ví dụ nếu đặt giá trị thuộc tính này bằng 2, thì giá trị cột 2 sẽ được chọn, các cột khác chỉ đóng vai trò hiển thị.
- Column widths: 1 ; 1 .5 (Độ rộng các cột, nếu viết như trên thì độ rộng cột một là 1 cm, độ rộng cột hai là 1.5 cm).
- Limit To List: No/Yes (Yes: Chỉ chọn trong danh sách, No: Có thể nhập giá trị ngoài danh sách)
- Width: bề rộng của danh sách, bằng tổng bề rộng các cột.
- Height: 5 cm (bề dài của danh sách. Danh có thể có nhiều hàng, nhưng ta chỉ cho hiện ra trong các hàng trong phạm vi 5 cm)

### ***III.5.5. Dùng Wizard tạo List Box/Combo Box***

Dùng công cụ Wizard có thể tạo được 2 loại Combo/List Box sau:

1. Combo/List Box có nguồn dữ liệu là một bảng/truy vấn.
2. Combo/List Box có nguồn dữ liệu là một dãy giá trị đặt vào thuộc tính Row Source

#### **Cách tạo:**

1. Trước hết phải chọn chức năng Control Wizard bằng cách bấm vào biểu tượng Control Wizard (nếu nó chưa được bật).
2. Chọn nút Combo Box hoặc List Box trên hộp công cụ.
3. Kéo một trường của bảng/truy vấn nguồn vào mẫu biểu. Access sẽ tạo một List/Combo Box buộc vào trường này.

4. Lần lượt trả lời các câu hỏi do Wizard đặt ra để xác định nguồn dữ liệu và các đặc trưng khác của List/Combo Box. Sau khi trả lời một câu hỏi bấm Next để sang câu hỏi tiếp, cuối cùng bấm Finish.

**Chú ý:**

1. Nếu tạo List/Combo Box lấy nguồn dữ liệu từ bảng/query thì khi cửa sổ đầu tiên của Wizard xuất hiện chọn “I want the combo box (list box) to look up the value in table or query”.
2. Nếu tạo List/Combo Box lấy nguồn dữ liệu từ danh sách tự tạo thì khi cửa sổ đầu tiên của Wizard xuất hiện chọn “I will type in the value that I want” . Sau đó nhập giá trị vào cửa sổ tiếp theo.
3. Nếu dùng nguồn dữ liệu từ bảng/query thì chọn bao nhiêu trường thì có bấy nhiêu cột, còn nếu tự tạo dữ liệu thì cần xác định rõ số cột cần dùng ở bước 2 (Number of column)
4. Nếu có từ hai trường trở lên thì tại bước 3 cần chọn trường (cột) để lấy giá trị đưa vào trường gắn với Combo box / list box.
5. Tại bước cuối cùng cần xác định rõ trường gắn với Combo box / list box (thường đã có giá trị mặc định do việc kéo trường ở bước 3 trong cách tạo ).

**III.5.6. Dùng điều khiển Check Box, Toggle Button, Option Button**

**\* Giới thiệu chung**

Hộp kiểm tra (Check box), nút lựa chọn (Option button) và nút bật tắt (Toggle button) thường được sử dụng để nhận các giá trị Yes hoặc No. Các điều khiển này chỉ khác nhau ở hình thức, còn cách sử dụng hoàn toàn giống nhau. Khi được chọn, các điều khiển nhận giá trị Yes, khi không được chọn chúng có giá trị No. Cách nhận biết việc chọn các điều khiển này như sau:

Check box được chọn khi có dấu “✓” trong hộp.

Option button được chọn khi có dấu “■” trong nút

Toggle button được chọn khi nó trông như bị nhấn xuống

**\* Cách tạo các điều khiển buộc vào các trường Yes/No.**

Các điều khiển nói trên thường được dùng để nhập dữ liệu cho các trường Yes/No. Cách tạo như sau:

1. Chọn một trong các nút: Check box, Option button, Toggle button
2. Kéo một trường kiểu Yes/No vào mẫu biểu. Kết quả tạo được một điều khiển buộc vào trường vừa kéo và một nhãn đi kèm. Tiêu đề mặc định của nhãn chính là tên trường được kéo.

**Chú ý:** Riêng đối với Toggle button thì không có nhãn đi kèm. Muốn có nhãn, thì ta phải tự tạo bằng cách sử dụng nút Label trên hộp công cụ.

**III.5.7. Sử dụng nhóm lựa chọn (Option Group)**

**Công dụng:**

Buộc vào các trường có kiểu Byte hoặc Integer. Giúp cho thao tác chọn lựa của người sử dụng được dễ dàng hơn, vì không phải gõ dữ liệu vào hộp văn bản, mà chỉ cần bấm vào một phần tử của nhóm để chọn.

**Cách tạo:** Trong cửa sổ Form, lần lượt thực hiện:

1. Tắt chức năng Control Wizards.
2. Chọn nút Option Group trên Toolbox
3. Kéo một trường (có giá trị nguyên) vào Form. Kết quả: Tạo một điều khiển Option Group (và một nhãn kèm theo) buộc vào trường vừa kéo. Điều khiển là một hình chữ nhật rộng, ta có thể chỉnh lại kích thước cho phù hợp.
4. Tạo các nút lựa chọn (hoặc các hộp kiểm tra) bên trong điều khiển nhóm lựa chọn vừa xây dựng ở bước trên.
5. Sửa lại các nhãn của các nút lựa chọn cho hợp với ý nghĩa của bài toán

### III.6. Tạo Command Buttons (nút lệnh)

■ Chức năng : là đối tượng dùng để thực hiện một thao tác (hành động) nào đó khi kích hoạt (nhấn nút lệnh).

■ Thao tác:

1. Mở mẫu biểu ở chế độ Design
2. Bật chức năng Control Wizard
3. Chọn biểu tượng Command Button trên thanh công cụ
4. Bấm chuột tại vị trí trên mẫu biểu nơi muốn đặt nút lệnh.
5. Trả lời từng câu hỏi của Wizard, cuối cùng bấm Finish

■ Các hành động thường dùng(Action): (Dùng trong các bước Wizard

- Record Navigation : tìm và di chuyển bản ghi
  - + Find record : tìm kiếm bản ghi
  - + Go to First record: Chuyển đến bản ghi đầu
  - + Go to Last record: Chuyển đến bản ghi cuối
  - + Go to Next record: Chuyển đến bản ghi tiếp
  - + Go to Previous record: Chuyển đến bản ghi trước đó
- Record Operations : gồm các hành động
  - + Add new record : Thêm bản ghi mới
  - + Delete Record: Xoá bản ghi hiện thời
  - + Duplicate record: nhân đôi bản ghi
  - + Save Record: Ghi bản ghi hiện thời
  - + Print Record: In bản ghi hiện thời.
- Form Operations : gồm các hành động với Form:
  - + Open Form : Mở một Form
  - + Close Form : Đóng Form

.....
- Report Operations : gồm các hành động với Report:
  - + Preview Report : Xem trước khi in
  - + Print Report: In Report.



.....

- Applications : làm việc với các ứng dụng
  - + Quit Applications : thoát khỏi ứng dụng
  - + Run Applications : chạy một ứng dụng
  - + Run excel: chạy excel
  - + Run Word: chạy word
- Miscellaneous : những hành động khác
  - + Print Table
  - + Run Query
  - + Run Macro.

#### **IV. MẪU BIỂU TRÊN NHIỀU BẢNG**

Phần này trình bày cách tổ chức nhập dữ liệu đồng thời trên nhiều bảng có quan hệ với nhau, trong đó có một bảng chính và các bảng phụ. Quan hệ giữa bảng chính và mỗi bảng phụ là quan hệ một - nhiều. Cách tổ chức như sau:

1. Xây dựng cho mỗi bảng một mẫu biểu. Mẫu biểu ứng với bảng chính gọi là mẫu biểu chính, các mẫu biểu còn lại gọi là mẫu biểu phụ.
2. Trên mẫu biểu chính, tạo các điều khiển kiểu SubForm buộc vào các mẫu biểu phụ. Sử dụng các thuộc tính của điều khiển SubForm để thiết lập mối quan hệ giữa mẫu biểu chính và mẫu biểu phụ, sao cho trên điều khiển SubForm chỉ thể hiện các bản ghi của mẫu biểu phụ có liên quan đến bản ghi đang xét trên mẫu biểu chính.

##### **IV.1. Công dụng.**

Dùng để cập nhật dữ liệu đồng thời cho nhiều bảng, truy vấn:

- Mẫu biểu chính thể hiện thông tin của một bảng/truy vấn.
- Mẫu biểu phụ thể hiện thông tin của một bảng/truy vấn khác có liên quan đến bản ghi đang xét trong mẫu biểu chính.

##### **IV.1. Quan hệ giữa các mẫu biểu chính và phụ**

Các bảng dữ liệu cần có quan hệ một - nhiều.

- Mẫu biểu chính thể hiện bảng/truy vấn bên một.
- Mẫu biểu phụ thể hiện bảng/truy vấn bên nhiều.

\* *Giới hạn các mẫu biểu chính và phụ:*

- Trong một mẫu biểu chính có thể có nhiều mẫu biểu phụ (gọi là mẫu biểu phụ cấp một).
- Số mẫu biểu phụ cấp một là không hạn chế.
- Trong mỗi mẫu biểu phụ cấp một cho phép đặt các mẫu biểu phụ khác (gọi là mẫu biểu phụ cấp hai).
- Access chỉ cho phép tổ chức đến các mẫu biểu phụ cấp hai.

##### **V.3. Cách tổ chức mẫu biểu chính và mẫu biểu phụ.**

■ *Trình tự thực hiện:*

1. Tạo một mẫu biểu chính và các mẫu biểu phụ một cách độc lập. Lúc đó giữa chúng còn chưa có quan hệ gì với nhau.
2. Sử dụng các thuộc tính (của mẫu biểu): View Allowed và Default View để qui định dạng trình bày của các mẫu biểu phụ, ý nghĩa của các thuộc tính này được cho trong bảng sau:

<b>View Allowed</b>	<b>Default View</b>	<b>Dạng trình bày của mẫu biểu</b>
Datasheet	Datasheet	Dạng bảng
Form	Single/Continuos Form	Dạng biểu
Both	Datasheet hoặc Single/Continuos Form	Cả hai dạng trên

Chú ý: Đối với các mẫu biểu phụ ta thường đặt giá trị sau cho các thuộc tính: View Allowed: Datasheet và Default View: Datasheet

3. Mở mẫu biểu chính trong chế độ Design. Bấm phím F11 để xuất hiện đồng thời cửa sổ Database.
4. Kéo một mẫu biểu phụ từ cửa sổ Database vào trong cửa sổ thiết kế của mẫu biểu chính. Kết quả: Access tạo một điều khiển kiểu SubForm buộc vào mẫu biểu phụ và một nhãn đi kèm

Tên(thuộc tính Name) của điều khiển SubForm được đặt mặc định là tên của mẫu biểu phụ được kéo.

Tiêu đề (thuộc tính Caption) của nhãn đi kèm cũng được đặt mặc định là tên của mẫu biểu phụ được kéo. Ta có thể sửa lại các thuộc tính trên nếu muốn.

*Chú ý.* Để sửa thiết kế của mẫu biểu phụ, ta bấm đúp nút trái chuột.tại điều khiển SubForm tương ứng. Khi đó sẽ nhận được cửa sổ thiết kế của mẫu biểu phụ .

5. Tạo sự liên kết giữa mẫu biểu chính và mẫu biểu phụ. Dùng các thuộc tính (của điều khiển kiểu Subform được tạo trong bước 4) là LinkChildFields và LinkMasterFields để điền các trường liên kết giữa mẫu biểu chính và mẫu biểu phụ. Nói một cách cụ thể hơn:

+ Trong dòng thuộc tính LinkmasterFields ta ghi các trường liên kết của mẫu biểu chính.

+ Trong dòng thuộc tính LinkchildFields ta ghi các trường liên kết của mẫu biểu phụ.

## CHƯƠNG 6

### BÁO BIỂU

Báo biểu là một kiểu biểu mẫu đặc biệt được thiết kế cho khâu in ấn, trong báo biểu, Access tổ hợp dữ liệu trong bảng và truy vấn để có thể in theo những yêu cầu cụ thể.

#### I. XÂY DỰNG BÁO BIỂU BẰNG HỘP CÔNG CỤ

##### I.1. Các vấn đề thường gặp khi xây dựng báo biểu

Khi tạo báo biểu ta thường phải giải quyết các vấn đề sau:

1. Xây dựng đầu biểu (thường đặt ở đầu trang) .
2. Xây dựng thân biểu (ở thân báo biểu) và chỉnh lý để thân biểu khớp với đầu biểu.
3. Tạo nền mờ cho một đoạn văn bản cần nhấn mạnh.
4. Chọn Font chữ, cỡ chữ, kiểu in (đậm, nghiêng), căn lề.
5. Đánh số thứ tự cho báo biểu.
6. Ngắt trang theo yêu cầu (chuyển sang trang mới).
7. Đánh số trang

##### I.2. Các công cụ thường sử dụng

1. Điều khiển Line, Rectangle để vẽ đường thẳng và hình chữ nhật.
2. Chức năng Duplicate của menu Edit dùng để nhân bản một nhóm điều khiển được chọn.
3. Các chức năng Align và Size của menu Format để chỉnh lý các ô điều khiển tạo nên đầu biểu và thân biểu.
4. Sử dụng tổ hợp phím Ctrl + các phím mũi tên để di chuyển cả khối các điều khiển được chọn theo các bước ngắn.
5. Sử dụng thanh công cụ Formatting (Form/report) để chọn các trình bày cho ô điều khiển.
6. Để tạo cột số thứ tự, ta làm như sau:
  - Tạo một điều khiển Text box kiểu Unbound. Gán số 1 cho điều khiển bằng cách đặt = 1 vào thuộc tính Control Source hoặc gõ trực tiếp =1 vào ô điều khiển.
  - Đặt thuộc tính Running Sum là:
    - i. Over Group nếu muốn đánh số thứ tự cho từng nhóm.
    - ii. Over All nếu muốn đánh số thứ tự cho toàn báo biểu.
7. Sử dụng điều khiển Page Break trên hộp công cụ để thực hiện ngắt trang
8. Để in số trang ta dùng một hộp văn bản kiểu tính toán (trong đó ghi công thức =Page). Hộp này có thể đặt ở đầu trang, hoặc cuối trang.

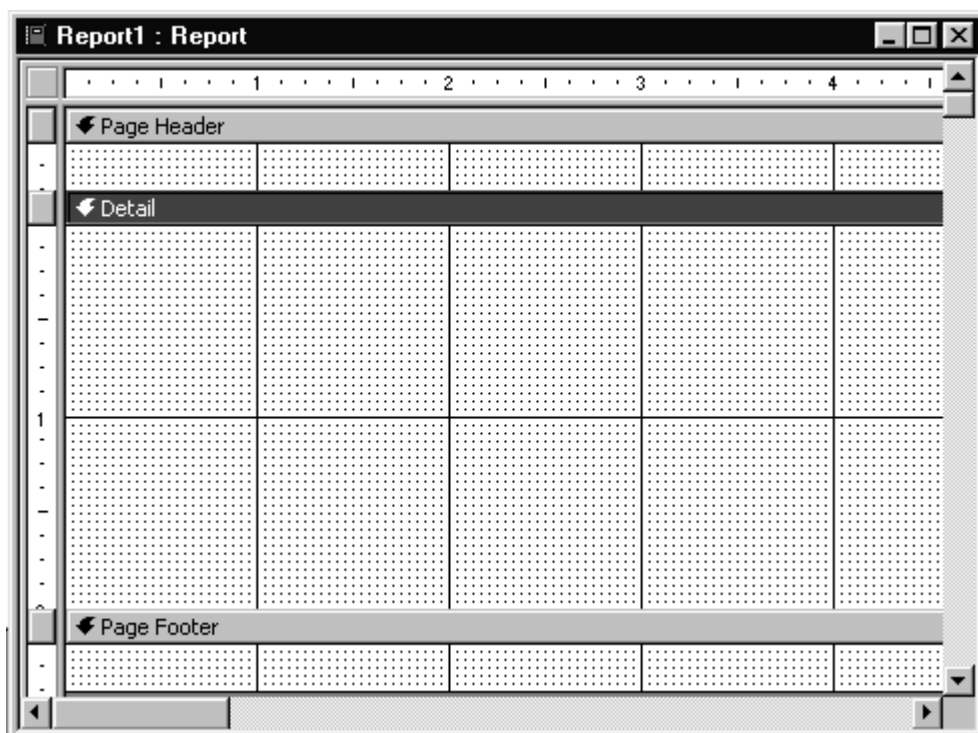
### I.3. Các bước xây dựng báo biểu

1. Từ cửa sổ Database, chọn mục Report, rồi chọn New. Kết quả xuất hiện cửa sổ New Report



2. Chọn một bảng hoặc truy vấn nguồn trong hộp Choose the table... hoặc không chọn bảng/truy vấn nguồn. Có thể bỏ qua việc chọn bảng/truy vấn nếu không cần sử dụng nguồn dữ liệu hoặc chọn nguồn dữ liệu sau này nhờ dùng thuộc tính Record Source.
  - Ở đây, cũng giống như với mẫu biểu, ta thấy có cách tạo báo biểu ứng với 3 lựa chọn: Design View, những lựa chọn tự động (có từ AutoForm) và những lựa chọn dùng công cụ Wizard (những lựa chọn có từ Wizard).
  - Nếu chọn dùng công cụ Wizard thì báo biểu được xây dựng bằng công cụ Wizards của Access và ta chỉ việc trả lời các câu hỏi mà Access yêu cầu.
  - Nếu chọn Design View thì báo biểu được thiết kế theo ý của người sử dụng bằng cách sử dụng hộp công cụ (Tool Box).
  - Nếu lựa chọn tự động thì khuôn dạng của báo biểu sẽ được tạo tự động. Trong trường hợp này ta luôn phải chọn bảng/truy vấn nguồn cho báo biểu.

Trước hết ta chọn cách thiết báo biểu theo ý của người sử dụng, bằng cách chọn Design View, rồi bấm OK. Kết quả nhận được thiết kế báo biểu



3. Sử dụng hộp công cụ để tạo các điều khiển trên báo biểu.
4. Sử dụng các công cụ để thực hiện các yêu cầu khi xây dựng báo biểu như kẻ đầu biểu, thân biểu, chọn font chữ, kiểu in, đánh số thứ tự, ngắt trang, . . . .
5. Sau khi hoàn tất thiết kế, ta ghi và đặt tên cho báo biểu, rồi trở về cửa sổ Database.

#### **I.4. Thực hiện in báo biểu**

Để in kết quả của báo biểu ra máy in, ta làm như sau.

- Chọn báo biểu cần thực hiện in từ cửa sổ Database
- Chọn chức năng Print từ menu File.
- *Chú ý:* Nếu báo biểu đang ở chế độ Print Preview hoặc Design, thì ta cũng có thể in kết quả của nó bằng cách chọn File → Print.

## **II. SẮP XẾP VÀ TẬP HỢP DỮ LIỆU THEO NHÓM**

### **II.1. Sắp xếp dữ liệu**

Khi xây dựng báo biểu ta có thể sắp xếp để kết quả in ra theo một trình tự nào đó. Do vậy ta có thể không cần sử dụng truy vấn để sắp xếp trước dữ liệu cho báo biểu.

#### ■ *Khả năng sắp xếp:*

- Có thể sắp xếp theo trường
- Có thể sắp xếp theo một biểu thức chứa các trường
- Có thể sắp xếp tối đa trên 10 trường hoặc biểu thức

#### ■ *Muốn sắp xếp dữ liệu trên báo biểu ta làm như sau:*

- Mở báo biểu trong chế độ Design View
- Chọn mục Sorting And Grouping từ men View để hiện cửa sổ

- Đưa các trường và biểu thức dùng để sắp xếp vào cột Field/expression (mỗi trường hoặc biểu thức đặt trên một dòng).
- Chọn thứ tự sắp xếp (tăng - Ascending hoặc giảm - Descending) trong cột Sort Order. Nếu sắp xếp theo nhiều trường hoặc biểu thức, thì thứ tự ưu tiên lấy từ trên xuống.

## II.2. Phân nhóm dữ liệu

■ Có thể phân nhóm trên bất kỳ trường hay biểu thức nào đã được sắp thứ tự (đã đưa vào cột Field/expression).

■ Thứ tự phân nhóm: Giả sử ta sắp xếp và phân nhóm theo ba trường theo thứ tự từ trên xuống dưới là Field 1 , Field2, Field3. Khi đó dữ liệu được phân nhóm như sau:

- Trước hết các bản ghi được phân nhóm theo Field1 . Ta nhận được các nhóm cấp 1.
- Trong mỗi nhóm cấp 1, các bản ghi lại được phân nhóm theo Field2. Ta nhận được các nhóm cấp 2.
- Trong mỗi nhóm cấp 2, các bản ghi lại được phân nhóm theo Field3. Ta nhận được các nhóm cấp

■ Cách nhóm dữ liệu theo trường phân nhóm

- Cách nhóm dữ liệu được quy định bằng các thuộc tính Group On và Group Interval của trường phân nhóm trong cửa sổ Sorting And Grouping. Ví dụ nếu đặt thuộc tính Group On là Each Value và thuộc tính Group Interval là 1 , thì dữ liệu được phân nhóm theo giá trị của trường/biểu thức phân nhóm. Khi đó các bản ghi có cùng giá trị trên trường/biểu thức phân nhóm sẽ được nhóm lại.

## II.3. Đầu nhóm và cuối nhóm

### II.3.1. Tạo đầu nhóm và cuối nhóm trên báo biểu.

Mỗi nhóm (bất kỳ cấp nào) cũng có thể thêm phần đầu nhóm và phần cuối nhóm trên mẫu biểu. Cách thêm hoặc hủy chúng như sau:

- Mở cửa sổ Sorting And Grouping
- Chọn trường hay biểu thức phân nhóm
- Đặt thuộc tính Group Header là Yes để thêm phần đầu nhóm hoặc No để hủy phần đầu nhóm.
- Đặt thuộc tính Group Footer là Yes để thêm phần cuối nhóm hoặc No để hủy phần cuối nhóm.

### II.3.2. Đặt các ô điều khiển trên phần đầu/cuối nhóm.

Trên các phần đầu nhóm và cuối nhóm ta có thể đặt bất kỳ điều khiển nào của báo biểu.

### II.3.3. Tổng hợp dữ liệu trên từng nhóm

Tại đầu/cuối nhóm, ta có thể sử dụng một số hàm chuẩn của Access để thực hiện việc tổng hợp dữ liệu của mỗi nhóm. Ví dụ dùng hàm Sum để tính tổng các giá trị của một trường số trên các bản ghi của mỗi nhóm, dùng hàm Count để đếm số bản ghi của nhóm.

### **II.3.4. In theo nhóm**

Thuộc tính Keep Together trong cửa sổ Sorting and Grouping cho 2 khả năng in đầ  
hầu của nhóm như sau:

- Nếu Keep Together: Whole Group thì nhóm sẽ bắt đầu in từ trang mới nếu phần còn lại của trang không chứa nổi tất cả các bản ghi của nhóm.
- Nếu , Keep Together: With First Detail thì . trên mỗi trang, tiêu đề nhóm sẽ được in kèm với ít nhất một bản ghi.

## **III. NHÓM DỮ LIỆU THEO TRƯỜNG PHÂN NHÓM**

### **III.1. Cách phân nhóm đối với trường/biểu thức kiểu số**

Có hai cách phân nhóm: Theo giá trị và theo miền giá trị.

#### **III.1.1. Phân nhóm theo giá trị**

Để phân nhóm theo giá trị ta đặt: Group On: Each Value. Khi đó các bản ghi được sắp xếp theo thứ tự tăng hoặc giảm của trường/biểu thức phân nhóm, sau đó các bản ghi có cùng giá trị trên trường/biểu thức phân nhóm sẽ được đưa vào một nhóm.

#### **III.1.2. Phân nhóm theo miền giá trị**

Để phân nhóm theo miền giá trị ta đặt:

- Thuộc tính Group On là Interval
- Thuộc tính Group Interval là một giá trị

Khi đó miền phân nhóm là các đoạn có độ dài bằng giá trị đưa vào thuộc tính Group Interval và mốc là giá trị 0. Chẳng hạn, nếu đặt các thuộc tính: Group On: Interval và Group Interval: 5 thì miền phân nhóm là các đoạn có độ dài 5 và mốc là 0. Đó là các đoạn: [-10 -6], [-5,-1], [0,4], [5,9], [10,14], . .

Các bản ghi có giá trị của trường/biểu thức phân nhóm rơi vào cùng một đoạn sẽ được đưa vào một nhóm.

*Chú ý:* Các bản ghi được sắp xếp theo nhóm. Nhưng trong cùng một nhóm thì các bản ghi chưa hẳn đã được sắp xếp theo giá trị của trường/biểu thức phân nhóm

### **III.2. Cách phân nhóm đối với trường/biểu thức kiểu Date/time**

Có hai cách phân nhóm: Theo giá trị và theo miền giá trị.

#### **III.2.1. Phân nhóm theo giá trị**

Để phân nhóm theo giá trị ta đặt: Group On: Each Value Khi đó các bản ghi được sắp xếp theo thứ tự tăng hoặc giảm của trường/biểu thức phân nhóm, sau đó các bản ghi có cùng giá trị trên trường/biểu thức phân nhóm sẽ được đưa vào một nhóm.

#### **III.2.2. Phân nhóm theo miền giá trị**

Miền giá trị có thể như theo một trong các đơn Year, Qtr (quí), Month, Week, Day...

Để phân nhóm theo miền giá trị ta đặt:

- Thuộc tính Group On là một trong các đơn vị trên.
- Thuộc tính Group Interval là một biểu thức số Khi đó miền phân nhóm được xác định bằng hai thuộc tính trên.

Chẳng hạn, nếu đặt các thuộc tính: ' Group On: Month Group Interval: 6 thì miền phân nhóm là các khoảng thời gian 6 tháng một bắt đầu tính từ tháng thứ nhất trong năm.

### III.3. Cách phân nhóm đối với trường/biểu thức kiểu Text

#### III.3.1. Phân nhóm theo giá trị

Để phân nhóm theo giá trị ta đặt: Group On: Each Value. Khi đó các bản ghi được sắp xếp theo thứ tự tăng hoặc giảm của trường/biểu thức phân nhóm và đưa vào một nhóm.

#### III.3.2. Phân nhóm theo các ký tự đầu

Để phân nhóm theo các ký tự đầu ta đặt:

- Thuộc tính Group On là Prefix Character
- Thuộc tính Group Interval là một giá trị nguyên n

Khi đó các bản ghi trùng nhau trên n ký tự đầu được phân nhóm và đưa vào một nhóm

## IV. TỔNG HỢP DỮ LIỆU TRÊN BÁO BIỂU

### IV.1. Các hàm dùng để tổng hợp dữ liệu.

Cho phép tổng hợp dữ liệu trên các nhóm và trên toàn bộ mẫu biểu bằng cách sử dụng các hàm sau:

- Sum	Tính tổng
- Avg	Tính giá trị trung bình
Min	Tìm giá trị nhỏ nhất
Max	Tìm giá trị lớn nhất
Count	Đếm số bản ghi
First	Cho bản ghi đầu tiên
Last	Cho bản ghi cuối cùng

### IV.2. Tổng hợp dữ liệu trên mỗi nhóm.

Để tổng hợp dữ liệu của các bản ghi trong nhóm, ta tạo các điều khiển Unbound tại đầu hoặc cuối nhóm và đặt vào ô điều khiển các hàm nói trên.

Ví dụ: Khi xây dựng bảng tổng hợp lương cho một công ty, ta cần tính tổng lương và số người của mỗi đơn vị thuộc công ty. Muốn vậy ta xây dựng một báo cáo và nhóm dữ liệu theo trường DV (đơn vị). Tại đầu hoặc cuối nhóm DV ta đưa vào các điều khiển sau:

Số người:	=Count([HT])
Tổng lương:	=Sum([NC]*[ML])

Trong đó HT (họ tên), NC (ngày công) và ML (mức lương) là các trường của bảng/truy vấn nguồn của báo biểu.

### IV.3. Tổng hợp dữ liệu trên toàn báo biểu

Để tổng hợp dữ liệu trên các bản ghi của toàn bộ báo biểu, ta cũng làm như cách trên, nhưng các điều khiển được đặt tại đầu hoặc cuối báo biểu.

### IV.4. So sánh dữ liệu tổng hợp trên các phần của báo biểu

Khi xây dựng các báo biểu thường phải so sánh, đối chiếu số liệu tổng hợp giữa các phần với nhau. Ví dụ khi tổng hợp lương của công ty cần biết tổng lương của mỗi đơn vị chiếm bao nhiêu phần trăm tổng lương của toàn công ty.

Để so sánh dữ liệu tổng hợp trên các phần của báo biểu ta làm như sau: .



1. Tạo các ô điều khiển tại đầu hoặc cuối mỗi phần để nhận dữ liệu tổng hợp trên các phần. Dùng thuộc tính Nam đặt tên cho các ô điều khiển.

2. Sử dụng các ô điều khiển nói trên để thực hiện các phép so sánh dữ liệu tổng hợp trên các phần khác nhau của báo biểu. Cách làm như sau: Tạo các ô điều khiển mới (tại bất kỳ vị trí nào trên báo biểu), rồi đặt vào đó các biểu thức chứa tên các ô điều khiển đã xây dựng ở bước 1 .

#### **IV.5. Không in một số ô điều khiển của báo biểu.**

Nếu đặt thuộc tính Visible của ô là No thì nội dung của ô đó không in trên giấy, nhưng vẫn có khoảng trống đúng bằng kích thước của ô điều khiển.

Nếu đặt thuộc tính Height của điều khiển là 0 thì nội dung hoàn toàn bị bỏ qua trên kết quả in của báo biểu.

#### **V. BÁO BIỂU CHÍNH VÀ BÁO BIỂU PHỤ**

Cách tổ chức báo biểu chính và báo biểu phụ hoàn toàn giống như việc tổ chức mẫu biểu chính, phụ đã trình bày trong phần mẫu biểu. Trình tự thực hiện như sau:

1. Tạo báo biểu chính và phụ hoàn toàn độc lập. Lúc này chúng chưa có quan hệ gì với nhau

2. Mở báo biểu chính trong chế độ Design View, bấm phím F11 để đồng thời mở cửa sổ Databbse.

3. Kéo một báo biểu phụ từ cửa sổ Database vào trong cửa sổ thiết kế của báo biểu chính. Kết quả Access tạo một điều khiển kiểu SubReport buộc vào báo biểu phụ và một nhãn đi kèm.

Tên (thuộc tính name) của báo biểu phụ được đặt mặc định cho báo biểu phụ được kéo. Ta có thể sửa nếu muốn.

4. Tạo sự liên kết giữa báo biểu chính và báo biểu phụ.

Dùng các thuộc tính của SubReport như sau:

LinkChildFields: Trường liên kết của báo biểu phụ

LinkMasterFields: Trường liên kết của báo biểu chính

\* *Tác dụng của liên kết:* trên báo biểu phụ chỉ hiện các bản ghi cùng giá trị liên kết với các bản ghi hiện hành (đang xét) của báo biểu chính. Hay nói cách khác chỉ các bản ghi của báo biểu phụ có liên quan đến báo biểu chính mới được in trên báo biểu chính.

\* *Chú ý:* Nếu không tạo liên kết thì toàn bộ bản ghi của báo biểu phụ sẽ hiện trên báo biểu chính.

#### **VI. XÂY DỰNG BÁO BIỂU TỰ ĐỘNG BẰNG WIZARD**

1. Từ cửa sổ Database chọn mục Report rồi bấm nút New. Kết quả hiện cửa sổ New Report.

2. Chọn một bảng hoặc truy vấn làm nguồn dữ liệu cho báo biểu tại mục Choose the Table or Query... .

3 . Chọn một trong hai kiểu báo biểu tự động (có từ AutoReport) hoặc chọn Report Wizard trong danh sách trên dưới đây sẽ trình bày một số cách chọn trên.

#### ***Tạo báo biểu tự động kiểu Columnar***

Chọn mục AutoReport: Columnar trong cửa sổ New Report, rồi bấm OK. Kết quả nhận được cửa sổ Report. Nếu muốn thay đổi lại thì chọn View --> Design View để thay đổi.

### ***Tạo báo tự động kiểu Tabular***

Chọn mục AutoReport: Tabular trong cửa sổ New Report, rồi bấm OK. Kết quả nhận được cửa sổ Report. Nếu muốn thay đổi lại thì chọn View --> Design View để thay đổi.

### ***Tạo báo tự động bằng công cụ Wizard***

1. Chọn mục AutoReport: Tabular trong cửa sổ New Report rồi bấm OK.
2. Chọn các trường đưa vào báo biểu --> bấm Next
3. Chọn các trường dùng để phân nhóm --> bấm Next
4. Chọn các trường để sắp xếp nếu muốn --> bấm Next.
5. Chọn các dạng hiển thị tại mục Layout, chọn hướng in tại Orientation (VD chọn Portrait), --> bấm Next.
6. Chọn một trong các mẫu trang trí --> bấm Next (ở đây ta chọn Bold), rồi bấm Next.
7. Sửa lại tiêu đề cho báo biểu (tiêu đề mặc định là tên bảng/truy nguồn) --> bấm Finish.
8. Có thể chọn View, Design View để chuyển về cửa sổ thiết kế, rồi tiến hành đổi lại font chữ và tiến hành các chỉnh sửa khác nếu muốn

## CÁC BÀI TẬP THỰC HÀNH

### BÀI THỰC HÀNH SỐ 1

1-Tạo 2 bảng sau:

BanHang

Tên cột	Kiểu dữ liệu
STT	Auto Number
NgayBan	Date/Time
MaHang	Text
SoLuong	Number
TongTien	Currency

LoaiHang

Tên cột	Kiểu dữ liệu
MaHang	Text
TenHang	Text
DonGia	Currency

2-Nhập dữ liệu cho 2 bảng trên. Mỗi bảng khoảng 5 bản ghi.

3-Thiết lập mối quan hệ giữa 2 bảng thông qua MaHang.

4-Trong bảng **BanHang** hãy:

- Chèn trước cột TongTien một cột mới có tên là NguoiBan(Text)
- Chèn sau cột NguoiBan một cột mới có tên là DaThanhToan(yes/No)
- Di chuyển cột NgayBan xuống phía sau cột SoLuong.
- Đặt 3 chữ số thập phân cho cột TongTien.

5-Mở bảng BanHang ra:

- Tìm những bản ghi có MaHang là M1.
- Đặt chiều cao của hàng là 15.
- Đặt độ rộng của cột là Best Fit.
- Di chuyển cột NguoiBan ra phía sau cột DaThanhToan.
- giấu 2 cột NguoiBan và TongTien.
- Hiện lại 2 cột.
- Chỉ hiện các đường lưới dọc, ẩn đi các đường lưới ngang.
- Đặt màu nền của bảng màu xanh, còn các đường lưới màu vàng.
- Đặt font chữ cho dữ liệu trong bảng là .Vntime cỡ 14.
- Sắp xếp tăng dần cột MaHang, sắp xếp giảm dần cột NgayBan.
- Lọc ra những bản ghi có MaHang là M1.
- Chèn thêm 1 bản ghi.
- Xoá bản ghi đầu tiên.

### BÀI THỰC HÀNH SỐ 2

#### 1, khởi động Access và tạo bảng sau (danh sách)

Stt	Họ tên	Số báo danh	Ngày sinh
1	Trần Thịnh	99A001	12/12/78
2	Mạnh Vũ	99A002	09/09/78
3	Trọng Minh	99A003	01/09/77

#### 2, Sửa

- Ngày sinh người số 2 là 20/08/80
- Họ tên người thứ 3 là “Nguyễn Văn Phú”
- Thêm một người mới sau

4	Trọng Sỹ	99A004	01/09/76
---	----------	--------	----------

**3, Tạo bảng mới sau tên là(Ket Qua)**

Số báo danh	Toán	Lý	Hoá
99A001	6	8	9
99A002	7	7	8
99A003	8	8	7
99A004	8	8	8

**4, Hãy tạo mối quan hệ giữa hai bảng (Danh Sach và Ket qua) Qua trường Số báo danh****5, Tạo bảng sau tên là (Địa Chi)**

Số báo danh	địa chỉ
99A001	166- Nguyễn Văn cừ- Gia Lâm- Hà nội
99A002	Số 10- Hoàng Quốc Việt -Hà nội
99A003	193-Mai dịch -Hà nội
99A004	Số 234- Hoàng Quốc Việt -Hà nội

**6, Tạo mối Quan hệ giữa 3 bảng Danh sach - ket qua -địa chi****7,Sắp xếp cột toán tăng dần****8, Sắp xếp cột ngày sinh giảm dần****9,Tìm người có Số báo danh 99A003****10,Tìm người có Địa chỉ (193-Mai dịch -Hà nội)****11,Lọc dữ liệu.**a. Người có Toán $\geq$ 6b. Người có Lý $\geq$ 5 và Hoá  $\geq$ 7

c. Người có Ngày sinh=01/09/77

d. Người có Toán,Lý,Hoá $\geq$ 4**BÀI THỰC HÀNH SỐ 3****1. Tạo Bảng 1 tên là (Đơn đặt hàng)**

Mã đơn đặt hàng	Ngày đặt hàng	Mã vật tư
Vt001	12/12/99	T01
Vt002	11/11/99	T02
Vt003	10/10/99	T03
Vt004	02/02/99	T04
Vt005	03/03/98	T05
Vt006	02/06/99	T06

**2. Bảng 2 (Danh mục vật tư)**

Mã vật tư	Tên vật tư
T01	Ti vi
T02	Tủ lạnh
T03	Điều hoà
T04	Máy tính
T05	Túi sách
T06	Tủ tường

**3. Bảng 3 (Số lượng)**

Mã đơn đặt hàng	Số lượng
Vt001	500
Vt002	200
Vt003	400
Vt004	100

Vt005	300
Vt006	600

#### 4. Bảng 4 (Đơn giá)

Mã vật tư	Đơn giá
T01	5000
T02	6000
T03	4000
T04	6000
T05	8000
T06	5000

#### 5. Tạo mối Quan hệ Sau

- Bảng1 và Bảng 2
- Bảng1 và Bảng 3
- Bảng2 và Bảng 4
- Bảng1 và Bảng 2,3,4

6. lọc ra những mã vật tư là **T01** và **T02**

7. lọc ra với điều kiện Số lượng >300

8. Xoá mối Quan hệ trên giữa 3 Bảng

### BÀI THỰC HÀNH SỐ 4

#### 1, Tạo Bảng có tên là (Danh sách)

MNV	Họ đệm	Tên	Quê Quán	Ngày sinh	Giới tính
M01	Trần Văn	Đức	Hà nam	02/02/78	1
M02	Hoàng Văn	Tú	Hà nội	03/03/77	1
M03	Nguyễn Như	Tùng	Hà nội	05/05/81	1
M04	Trần Thu	Nga	Hà nội	05/06/77	0
M05	Nguyễn Như	Trọng	Hải phòng	02/02/78	1

#### 2, Bảng 2 có tên là (Đơn vị công tác)

MNV	Tên phòng ban	Chức vụ	Năm công tác
M01	Hành chính	Tp	12
M02	Tài vụ	PP	13
M03	Kế hoạch	Tp	12
M04	Kỹ thuật	Tp	11
M05	Đời sống	Nv	10

#### 3, Tạo Bảng 3 có tên là (Tiền lương)

MNV	Lương cơ bản	Tạm ứng
M01	500	100
M02	600	200
M03	200	100
M04	300	200
M05	500	250

4. Liên kết các bảng trên và tạo các query sau để truy vấn dữ liệu từ CSDL:

4.1. **QryDanh sách** gồm các thông tin sau :họ đệm, tên, quê quán, Ngày sinh, giới tính, chức vụ, năm công tác, lương cơ bản, phụ cấp, bảo hiểm xã hội, bảo hiểm y tế, được lĩnh, phụ cấp, bảo hiểm xã hội, bảo hiểm y tế, được lĩnh.

Trong đó:

- + Phụ cấp =15% Lương cơ bản
- + Bảo hiểm Xã hội=5% Lương cơ Bản

- + Bảo hiểm Y Tế=2% Lương cơ Bản  
 + Được Lĩnh=(Lương cơ bản+Phụ cấp)-(Bảo hiểm Xã hội+Bảo hiểm Y tế +Tạm ứng).
- 4.2. Tạo **query2** gồm những người là nam giới và quê ở Hà nội, với các thông tin như **QryDanhsach**.
- 4.3. Tạo **query3** gồm các thông tin: họ đệm, tên, tên phòng ban, chức vụ, được lĩnh.  
 Căn cứ vào năm công tác để thêm tiền vào cột được lĩnh như sau:  
 >=10 năm : thêm 50000<sup>d</sup>  
 >=5 và <10 năm: thêm 30000<sup>d</sup>  
 <5 năm: thêm 20000<sup>d</sup>
- 4.4. Tạo **query4** Đưa ra danh sách những người có năm công tác >=10 và ở phòng hành chính hoặc phòng tài vụ (với các thông tin như Query3).

## BÀI THỰC HÀNH SỐ 5

Tạo CSDL sau

**Bảng 1: có tên (Khách hàng)**

Số máy	Họ đệm	Tên	Địa chỉ
5635465	Trần Văn	Đức	193-Mai dịch -Hà nội
9271726	Hoàng Văn	Tú	166- Nguyễn Văn cử- Gia Lâm- HN
8237901	Nguyễn Như	Tùng	Số 10- Hoàng Quốc Việt –HN
7655194	Trần Ngọc	Hùng	168- Nguyễn Văn cử- Gia Lâm- HN
8390233	Nguyễn Như	Trọng	Số 234- Hoàng Quốc Việt -HN

**Bảng 2 có tên là** Danh sách gọi

Số máy	Số máy bị gọi	Thời gian gọi(phút)
5635465	04-7655194	12
9271726	08-7655412	3
9271726	034-8810073	11
5635465	04-8837219	5
8237901	08-7584343	14
8237901	030-768596	6
8390233	04-7655194	14
7655194	04-7985654	18
7655194	030-8832599	6

Căn cứ vào cách tính tiền sau để tính toán:

Mã vùng là:

04=1000đ/phút

08=3000đ/phút

034=2500đ/phút

030=1500đ/phút

Tạo các Query sau để truy vấn dữ liệu:

1.Query1: gồm các thông tin sau: số máy, số máy bị gọi, thời gian gọi, họ đệm, tên, địa chỉ, tiền cước, tiền phụ trội, tiền phải trả.

Trong đó:

- **Tiền Cước =Thời gian \*Tiền/phút**

- **Tiền phụ trội= 2% Tiền Cước**, nếu thời gian gọi >=5 và <10 phút  
 4% **Tiền cước**, nếu thời gian gọi >10 phút

-**Tiền phải trả = Tiền cước+Tiền phụ trội**

2.Query2: gồm danh sách những khách hàng có mã vùng 04.

3.Query3:gồm những khách hàng có thời gian gọi >=10 phút.

4.Query4:gồm danh sách những số máy mà máy **5635465** gọi đến.

5.Query5:gồm danh sách Họ tên và Địa chỉ những máy gọi đến máy **04-7655194**

## BÀI THỨC HÀNH SỐ 6

Sử dụng truy vấn Query để làm các Bài tập sau:

1, Giả sử công việc của ta là quản lý một cửa hàng băng đĩa nhạc ta có 4 chủng loại băng hình được đánh mã số là A001,A002,A003,A004 Tiền thuê băng 1 ngày là 2000đ cho mọi loại băng.

Hãy tạo CSDL gồm bảng sau:

SO LUONG

Stt	Mã BH	Người thuê	Ngày thuê	Ngày trả
1	A01	Hùng	02/02/02	11/11/02
2	A02	Huy	03/03/02	10/10/02
3	A01	Hiệp	09/09/02	09/09/02
4	A03	Hiếu	08/08/02	08/08/02
5	A04	Thắng	07/07/02	09/08/02
6	A03	Trương	08/08/02	09/09/02

DIACHI

Người thuê	Địa chỉ
Hùng	193-Mai dịch -Hà nội
Huy	166- Nguyễn Văn cừ- Gia Lâm- HN
Hiệp	Số 10- Hoàng Quốc Việt -HN
Hiếu	168- Nguyễn Văn cừ- Gia Lâm- HN
Thắng	Số 234- Hoàng Quốc Việt -HN
Trương	Số 23- Hoàng Quốc Việt -HN

TENBANG

Mã BH	Tên Băng
A01	Cải LươNg
A02	Ca nhạc
A03	Trưởng
A04	Phim Tập

- Tạo **Query1**: để tính cột **Tiền phải trả** trong Query này bao gồm toàn bộ bảng SOLUONG
- Tạo **Query2**: để hiện những ai thuê mã băng hình A01, trong Query này bao gồm các cột Tên băng, Người thuê, Địa chỉ .
- Tạo **Query3**: để hiện những ai thuê mã băng hình A01,A03 hoặc A04, trong Query này bao gồm các cột Tên băng, Người thuê, Địa chỉ
- Tạo **Query4**: để hiện những ai thuê từ tháng 3 đến tháng 10 năm 2002 trong Query này bao gồm các cột Tên băng, Người thuê, Địa chỉ
- Tạo **Query5**: để tính tiền cho mọi người từ ngày thuê đến ngày hiện tại là bao nhiêu tiền, không tính cho những người đã trả (nghĩa là ngày trả <DATE() ) trong Query5 này bao gồm các cột Tên Băng, Người thuê, Địa chỉ và Tiền.

Truy vấn theo nhóm (các cột trong query lấy cho phù hợp)

- Tạo Query6: Tính tổng nhóm theo mã băng hình xem mỗi băng hình cho thuê được bao nhiêu tiền.
- Tạo Query7: Tính tổng trung bình nhóm theo mã băng hình xem mỗi băng hình cho thuê **TB** được bao nhiêu tiền.
- Tạo Query8: Xem tiền thuê lớn nhất của mỗi loại băng hình là bao nhiêu.

- Tạo Query9: Xem tiền thuê nhỏ nhất của mỗi loại băng hình là bao nhiêu.
- Tạo Query10: Xem mỗi loại băng hình có bao nhiêu người thuê.
- Tạo Query11: Xem mỗi loại băng hình có bao nhiêu người thuê trước ngày 05/05/02.

## BÀI THỰC HÀNH SỐ 7

Tạo CSDL sau

Stt	Ngày tháng	Mã mặt hàng	Loại	Số Lượng
1	10/10/02	M1	A	2
2	10/10/02	M2	B	5
3	09/09/02	M3	C	4
4	08/08/02	M2	C	3
5	08/08/02	M1	B	6
6	09/09/02	M3	A	4

- Truy vấn tổng ( Group By) Theo ngày tháng và tính tổng số lượng.
- Truy vấn tổng ( Group By) Theo Mã mặt hàng và tính tổng số lượng.
- Truy vấn tổng ( Group By) Theo loại và tính tổng số lượng.
- Truy vấn Crosstab ứng với Ngày tháng (RowHeading) Mã mặt hàng (Colum Heading) Và số lượng (Value).
- Truy vấn Crosstab lấy Mã mặt hàng làm Row Heading, Loại làm Colum Heading và Số lượng làm Value.

## BÀI THỰC HÀNH SỐ 8

1, Xây dựng CSDL như sau:

### SOLUONG

Số TT	Ngày tháng	Mã mặt hàng	Số Lượng
1	10/10/02	M1	2
2	10/10/02	M2	5
3	09/10/02	M3	4
4	08/10/02	M2	3
5	08/10/02	M1	6
6	09/10/02	M3	4

### DOANH-SO

Ngày tháng	Tổng tiền trong ngày	Đã thanh toán
08/10/02	150000	100000
09/10/02	200000	150000
10/10/02	180000	150000

### LOAIHANG

Mã mặt hàng	Tên hàng
M1	Máy tính
M2	Máy in
M3	Máy photocopy

### DONGIA

Tên hàng	Đơn giá
Máy tính	120000
Máy in	50000



Máy photocopy	150000
---------------	--------

- Tạo một truy vấn bao gồm các trường Ngày tháng, Tên hàng, Số lượng, Đơn giá và Tổng= Đơn Giá \*Số lượng.
  - Tạo một truy vấn bao gồm các cột Ngày tháng, Mã mặt hàng, Tên hàng, Số lượng, Đơn giá nhưng chỉ gồm 2 mặt hàng M1 và M3.
- 2, Sử dụng phương tiện truy vấn **Query wizard**  
 Vẫn sử dụng CSDL trên
- Sử dụng simple Query Wizard để tạo một select Query bao gồm các cột ngày tháng, Mã mặt hàng, Tên hàng, Số lượng, Đơn giá.
  - Sử dụng Crosstab Query Wizard để tạo với ngày tháng làm Row Heading, Mã mặt hàng làm Colum Heading và tính tổng số lượng.
- 3, Sửa đổi dữ liệu bằng các truy vấn
- Trong bảng đơn giá ta cần thấy giảm đơn giá của các mặt hàng xuống 12% hãy dùng update Query để sửa.
  - Trong DOANH SO Tăng toàn bộ cột đã thanh toán lên 1000000đ
  - Trong bảng số lượng hãy xoá đi những bản ghi có Mã hàng là M1 và ngày tháng là 09/10/02.
  - Trong bảng SOLUONG hãy xoá những bản ghi có Số lượng =2.
  - Trong bảng SOLUONG hãy xoá đi những bản ghi có Số lượng=1 và Mã mặt hàng là M3.

## BÀI THỰC HÀNH SỐ 9

Xây dựng CSDL sau:

### DiemThi

SBD	TongDiem	DoiTuong
1	15	1
2	16.5	2
3	18	3
4	15	3
5	16	2

- 1-Hãy tăng 3 điểm cho những SBD có DoiTuong=1
- 2-Hãy tăng 2 điểm cho những SBD có DoiTuong=2.
- 3-Hãy tăng 14% điểm cho toàn bộ.
- 4-Xoá những bản ghi (sau khi đã thực hiện 1,2,3) mà TongDiem<=18.
- 5-Cho bảng sau:  
 DiemThi1

SBD	TongDiem	DoiTuong
101	15	1
102	16.5	1
103	18	3
104	15	2
105	16	2

-Hãy ghép bảng DiemThi vào bảng DiemThi1.

6-Từ bảng DiemThi1 hãy tạo 1 bảng mới có tên là DiemDo với điều kiện TongDiem>=16.

### BÀI THỰC HÀNH SỐ 10

Tạo các bảng sau:

Bảng 1 có tên là MUONSACH (mượn sách)

Tên cột	Kiểu dữ liệu
STT	Autonumber
MASV	Text
MASACH	Text
NGAYMUON	Date/Time
NGAYTRA	Date/Time
GHICHU	Text

Bảng 2 có tên là SINHVIEN (sinh viên) dùng để lưu trữ thông tin về sinh viên. Sẽ được đặt mối quan hệ với bảng MUONSACH qua cột MASV.

Tên cột	Kiểu dữ liệu
MASV	Text
HOTEN	Text
TENLOP	Text
KHOA	Text

Bảng 3 có tên là TENSACH (tên sách) dùng để lưu trữ thông tin về sách cho mượn, sẽ được đặt mối quan hệ với bảng MUONSACH qua cột MASACH.

Tên cột	Kiểu dữ liệu
MASACH	Text
TENSACH	Text
TACGIA	Text
NGONNGU	Text

Hãy nhập 5 bản ghi cho mỗi bảng trên.

Thiết lập quan hệ giữa 3 bảng và tạo các Query thực hiện các yêu cầu sau:

- 1-Hiện các cột HOTEN, TENSACH, TACGIA, NGONNGU, NGAYMUON, NGAYTRA.
- 2-Hiện các cột HOTEN, TENSACH, NGAYMUON, NGAYTRA cho những sinh viên mượn sách quá 7 ngày.
- 3-Hiện các cột HOTEN, TENSACH, NGAYMUON, NGAYTRA cho những sinh viên mượn sách trước ngày 25/9/2001.
- 4-Điền vào cột GHICHU của bảng MUONSACH là “Mượn quá hạn” cho những sinh viên mượn sách quá 15 ngày so với ngày hiện tại.
- 5-Tạo ra một bảng mới có tên là SACH\_TIENG\_ANH gồm các cột HOTEN, TENLOP, KHOA, TENSACH, TACGIA với điều kiện là sách tiếng Anh.
- 6-Tạo ra một bảng mới là KHOA\_TIN\_HOC gồm các cột HOTEN, TENLOP, KHOA, TENSACH, TACGIA với điều kiện là của những sinh viên khoa tin học.
- 7-Xoá hết tất cả những bản ghi của những sinh viên thuộc khoa CNTT trong bảng SINHVIEN.
- 8-Xoá tất cả những bản ghi của sinh viên thuộc lớp 5A12.
- 9-Điền vào cột GHICHU trong bảng MUONSACH là “ Không mượn thêm” với những sinh viên mượn sách quá 18 ngày mà chưa trả.

## BÀI THỰC HÀNH SỐ 11

Sử dụng CSDL gồm 3 bảng trong bài thực số 9 để tạo biểu mẫu.

1-Sử dụng chế độ **form wizard** để tạo các form:

a-Tạo form nhập dữ liệu cho bảng MUONSACH: Lấy hết các cột để nhập, dạng Columnar với kiểu Clouds.

b-Tạo form nhập dữ liệu cho bảng SINHVIEN: Lấy hết các cột để nhập, dạng Tabular với kiểu Flax.

c-Tạo form nhập dữ liệu cho bảng TENSACH: Lấy hết các cột để nhập, dạng Justified với kiểu International.

2-Sử dụng chế độ **design view** để tạo form cho việc nhập dữ liệu của bảng MUONSACH. Form có hình dạng như sau:

The screenshot shows a form window with the title 'Form1 : Form'. The form content is as follows:

TRƯỜNG CAO ĐẲNG CÔNG NGHIỆP HÀ NỘI	
QUẢN LÝ THƯ VIỆN	
Mã sinh viên	S001
Mã sách:	T001
Ngày mượn	16/08/01
Ngày trả:	17/09/01
Ghi chú	
KẾT THÚC	

## BÀI THỰC HÀNH SỐ 12

Trong bài tập này ta sẽ tạo ra các form giúp cho việc nhập dữ liệu được thuận tiện hơn. Tuy nhiên trước hết ta hãy tạo cơ sở dữ liệu sau (Bài toán về quản lý một cửa hàng):

Bảng HOADON:

Tên cột	Kiểu dữ liệu
STT	Autonumber
MAHANG	Text
SOLUONG	Number
LOAITIEN	Text

Bảng MAHANG:

Tên cột	Kiểu dữ liệu
MAHANG	Text
TENHANG	Text
DONGIA	Currency

Bảng TIENTE (Tiền tệ)

Tên cột	Kiểu dữ liệu
LOAITIEN	Text

TYGIA	Number
-------	--------

Tạo form sau:

**\*Trong đó:**

- Tên hàng** là một **combo box** với nguồn dữ liệu lấy từ bảng MAHANG.
- Loại tiền thanh toán** là một **combo box** với nguồn dữ liệu lấy từ bảng TIENTE.
- Tổng tiền** được tự động tính toán. (phải qui ra tiền Việt(VND) thông qua tỷ giá qui đổi trong bảng TIENTE).

*\* Một số dạng đề thi*

## ĐỀ BÀI 1

**Câu 1:** tạo các bảng:

Bảng DSNV(danh sách nhân viên) gồm:

- MaNV(Text)
- HoVaTen(Text)
- NgaySinh(Date/Time)

Bảng TĐNN(trình độ ngoại ngữ):

- MaNV(Text)
- TenNN(Text)
- TrinhDo(Text)

\*Tạo mối quan hệ và toàn vẹn dữ liệu.

Chú ý: Một người có thể biết 1 ngoại ngữ hoặc không biết ngoại nào.

**Câu 2:** tạo các form để tổ chức nhập dữ liệu theo mẫu:

Trong đó: Tên ngoại ngữ và trình độ là subform.

\*Chú ý: Sử dụng form vừa tạo để nhập ít nhất 10 nhân viên.

**Câu 3:** Tạo các Query thực hiện các yêu cầu sau:(mỗi yêu cầu 1 query)  
1-lập bảng tổng hợp có dạng

Họ và tên	Ngày sinh	Số ngoại ngữ đã biết

2-Tìm những người không biết ngoại ngữ nào.

3-Tìm những người sinh trước năm 1970 và có trình độ tiếng Anh C

4-Liệt kê nhân viên biết cùng một ngoại ngữ và cùng một trình độ (Ngoại ngữ và trình độ do người sử dụng nhập vào).

## ĐỀ BÀI 2

**Câu 1:** Tạo các bảng:

Bảng DSTRUONG (danh sách trường) gồm:

-MaTruong(Text)

-TenTruong(Text)

Bảng KQTHI (kết quả thi) gồm:

-MaTruong(Text)

-HoTenSV(Text)

-Diem(Double)

\*Tạo mối quan hệ và toàn vẹn dữ liệu

**Chú ý:** Mỗi trường có từ 2 đến 4 sinh viên tham gia.

**Câu 2:** Tạo các form để tổ chức nhập dữ liệu theo mẫu:

Trong đó:

1. Họ tên sinh viên và điểm là Subform.
2. Combo box tên trường gồm: Đại học Quốc gia Hà nội, Đại học Bách Khoa, Đại học thuỷ lợi, Đại học Giao thông vận tải.

**Chú ý:** Sử dụng form vừa tạo để nhập danh sách dự thi và điểm của sinh viên cả 4 trường (mỗi trường ít nhất 2 sinh viên).

**Câu 3:** Tạo query thực hiện các yêu cầu sau:

1. Lập bảng tổng hợp:

Tên trường	Số sinh viên	Tổng điểm	Điểm trung bình

- 2-Tìm các trường mà sinh viên có điểm thi cao nhất.
- 3-Tìm các trường mà có số sinh viên dự thi nhiều nhất.
- 4-Tìm những sinh viên có điểm thi lớn hơn một điểm nào đó nhập từ bàn phím.

### ĐỀ BÀI 3

**Câu 1:** Tạo các bảng:

Bảng DSKH (danh sách khách hàng) gồm:

- MaKH (Text)
- HoVaTen (Text)
- NamSinh (Integer)

Bảng CUOCGOI (cuộc gọi) gồm:

- MaKH (Text)
- NoiGoi Den(Text)
- SoPhut (Integer)
- DonGia (Double)

\*Tạo các mối quan hệ và toàn vẹn dữ liệu.

**Chú ý:** Một khách hàng có thể gọi nhiều lần.

**Câu 2:** Tạo các form để tổ chức nhập dữ liệu theo mẫu:

Trong đó:

1-Nơi gọi đến, số phút, đơn giá là subform.

2-combo box Nơi gọi đến gồm: Hà nội, Nam định, Huế, Đà nẵng.

**Chú ý:** sử dụng form vừa tạo để nhập Danh sách khách hàng (có khách hàng gọi nhiều lần). Số lượng khách hàng  $\geq 10$ .

**Câu 3:** Tạo các Query thực hiện các yêu cầu sau:(Mỗi yêu cầu 1 query)

1.lập 1 bảng tổng hợp:

Họ tên khách hàng	Số cuộc gọi	Tổng số tiền

2.Tìm các khách hàng có gọi đến “Huế” và có tổng tiền cho tất cả các cuộc gọi cao nhất.

3.Tìm các khách có tuổi trong khoảng từ 40 đến 50.

4.Tìm các khách có tổng tiền lớn hơn một mức nào đó nhập từ bàn phím.

#### ĐỀ BÀI 4:

**Câu 1:** Tạo các bảng:

Bảng PHONGBAN gồm:

-Phongban (Text)

Bảng DSCB (danh sách cán bộ) gồm:

-Phongban (Text)

-HoTen (Text)

-NgàySinh (Date/Time)

-LuongChinh (Double)

-PhuCap (Double)

Tạo các mối quan hệ và toàn vẹn dữ liệu.

Chú ý: Một phòng ban có nhiều nhân viên.



**Câu 2:** Tạo các form để tổ chức nhập dữ liệu theo mẫu:

Trong đó:

- Họ tên, ngày sinh, lương chính, phụ cấp là Subform.
- Combo box phòng ban gồm: Tài chính, tổ chức, hành chính, bảo vệ.

**Chú ý:** Sử dụng form vừa tạo để nhập danh sách nhân viên cho các phòng ban (mỗi phòng ban ít nhất 2 nhân viên).

**Câu 3:** Tạo các query thực hiện các yêu cầu sau (mỗi yêu cầu 1 query).

1-Lập bảng tổng hợp:

Phòng ban	Số người	Tổng lương	Lương trung bình

2-Tìm các nhân viên có lương cao nhất ở phòng “tổ chức”.

3-Tìm các nhân viên có tuổi >20.

4-Tìm những nhân viên có mức lương lớn hơn một giá trị nào đó được nhập từ bàn phím.(Mức lương=lương chính+phụ cấp).

## ĐỀ BÀI 5

**Câu 1:** Tạo các bảng.

Bảng DSSV(danh sách sinh viên) gồm:

- MaSV (Text)
- HoVaTen (Text)
- NgaySinh (Text)

Bảng MONHOC (môn học) gồm:

- MaSV (Text)
- TenMon (Text)
- DiemThi (Double)

■ Tạo các mối quan hệ và toàn vẹn dữ liệu.

**Chú ý:** Một sinh viên có điểm nhiều môn.

**Câu 2:** Tạo các form để tổ chức nhập dữ liệu theo mẫu:

Trong đó:

- Tên môn, điểm thi là Subform.
- Combo box tên môn gồm: Toán, Lý , Hoá, Sinh.

Chú ý: Sử dụng Form vừa tạo để nhập danh sách sinh viên và điểm. Số sinh viên  $\geq 10$ .

**Câu 3:** Tạo các query thực hiện các yêu cầu sau:( mỗi yêu cầu 1 query)

1-Lập bảng tổng hợp:

Họ tên	Ngày sinh	Tổng số điểm thi	Điểm trung bình

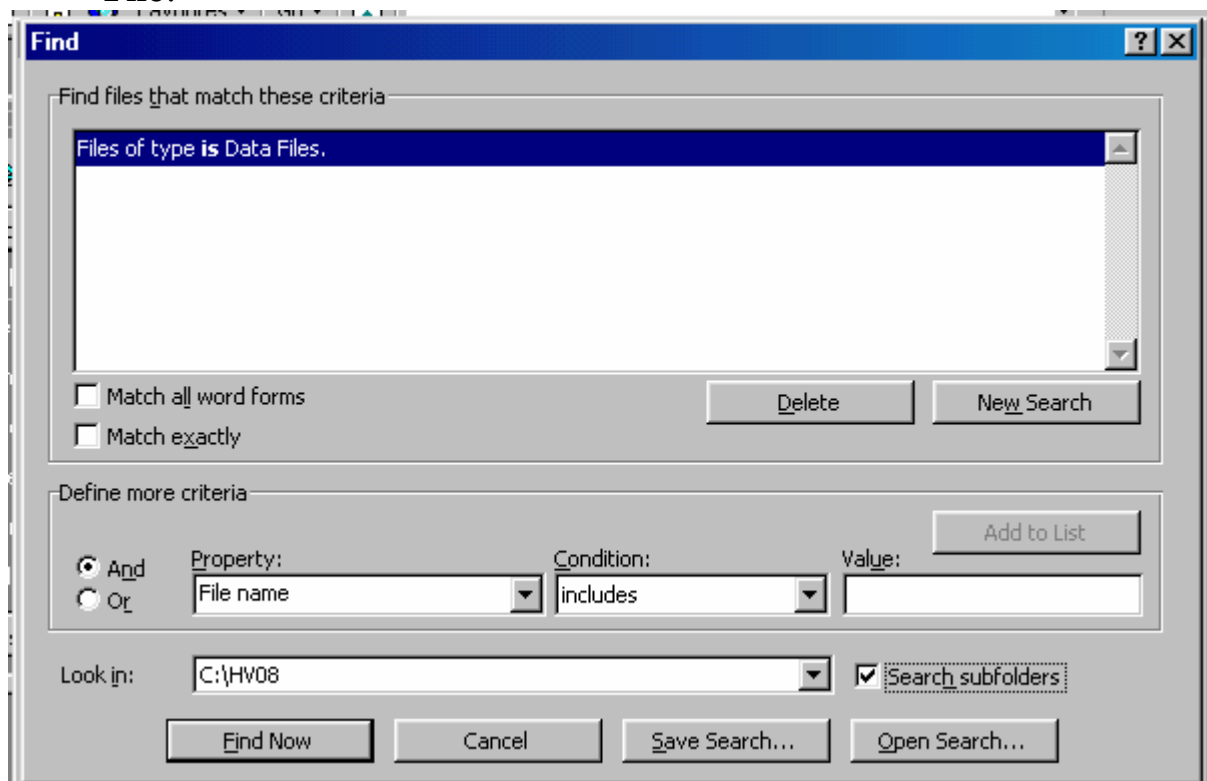
2-Tìm các sinh viên có số lượng điểm cao nhất.

3-tìm các sinh viên sinh năm 1982 có tổng điểm cao nhất.

4-Tìm danh sách sinh viên có điểm trung bình lớn hơn điểm trung bình được nhập từ bàn phím.

**Chương 1:****Xử lý các File trong Microsoft Access 2000****I. Khảo sát việc tìm kiếm File****1. Cách tìm kiếm các File:**

Các hộp thoại **Open**, **Import**, và **Link (Menu File)**, và các hộp thoại **Database To Convert From**, **Database To Compact From**, **Database To Save As MDE As**, và các hộp thoại **Encl/Decrypt Database (Menu Tools)**, tất cả đều giúp bạn tìm kiếm các **File** trên ổ đĩa cứng và trên bất cứ ổ mạng nào. Bạn có thể tìm kiếm một **File** theo tên **File**, theo kiểu, theo ngày được chỉnh sửa mới nhất, theo **Text** bên trong **File** ngoại trừ **cơ sở dữ liệu Microsoft Access** và để án **Microsoft Access**. Bạn cũng có thể tìm kiếm một **File** bằng cách chỉ định các tính chất **File** chẳng hạn như tác giả, chủ đề, chủ điểm, tiêu đề, hoặc ngày tạo **File**.

**2. Tìm kiếm các File bằng cách sử dụng các tính chất của File:**

- Các tính chất **File** là các chi tiết về **File** để giúp bạn nhận ra **File** đó
- **Ví dụ:** một tiêu đề mô tả, tên tác giả, chủ điểm, các từ khóa mà bạn gán vào cho **File**, hoặc thông tin quan trọng khác.

- Một số tính chất **File** được cài đặt trước: Các tính chất này bao gồm tên tác giả, tiêu đề, và chủ điểm đã có sẵn và bạn nhập vào một giá trị cho nó.
- **Ví dụ**, bạn có thể bổ sung từ khóa "employees" vào tất cả các **File cơ sở dữ liệu** nhân sự rồi tìm kiếm tất cả các **File** theo từ khóa đó.
- Tạo ra các tính chất **File** cho riêng bạn: Bạn có thể nhập các tính chất **File** bổ sung chẳng hạn một tiêu đề mô tả hoặc các từ khóa chính nhằm nhận biết **File** đó, rồi nhanh chóng tìm **File** bằng cách tìm thông tin này.
- Một số tính chất tự động được cập nhật: Những tính chất này bao gồm số liệu thống kê chẳng hạn như kích cỡ **File** và ngày tháng tạo **File** cũng như ngày chỉnh sửa sau cùng.
- **Ví dụ**, bạn có thể sử dụng các tính chất này để tìm kiếm tất cả các **File** được tạo sau ngày 3 tháng 3 năm 1998, hoặc những **File** đã được chỉnh sửa vào ngày hôm qua.

### 3. Tìm kiếm các File bằng cách sử dụng các tiêu chuẩn đặc biệt:

Trong hộp thoại **Open**, hãy nhấp **Tools**, rồi nhấp **File** để tìm kiếm một **File** bằng cách sử dụng các tính chất ngoài văn bản (**NonText Properties**) hoặc bằng cách sử dụng nhiều dạng kết hợp tiêu chuẩn cao cấp.

**Ví dụ:** bạn có thể tìm kiếm các **File** có chứa **Text** "budget" hoặc có từ khóa "Budget99".

### 4. Lưu tiêu Chuẩn tìm kiếm

Nếu bạn luôn luôn tìm kiếm thông tin giống nhau, hoặc nếu bạn có một mục cài đặt tìm kiếm phức tạp bằng cách sử dụng nhiều tiêu chuẩn tìm kiếm, bạn có thể lưu mục tìm kiếm để sử dụng sau đó.

### 5. Thay đổi tiêu chuẩn tìm kiếm đã được lưu

Nếu bạn đã lưu tiêu chuẩn tìm kiếm và muốn thay đổi tiêu chuẩn đó, thì bạn cũng có thể thực hiện được.

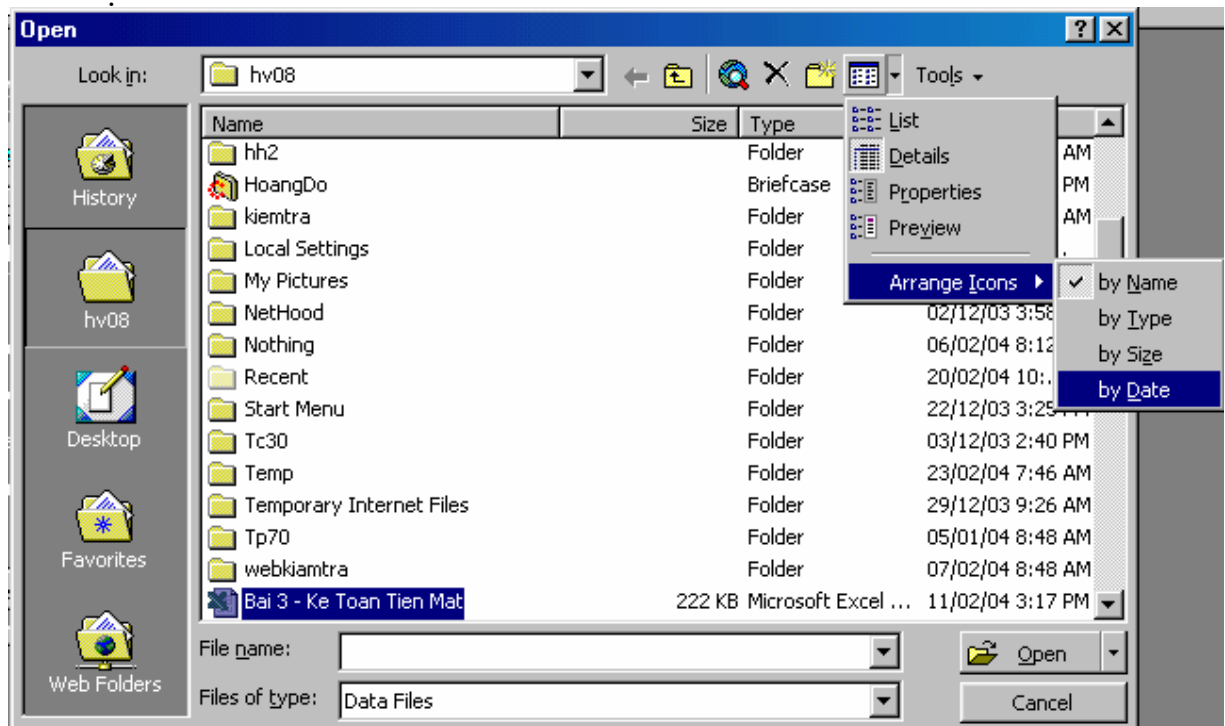
## II. Tìm các File

Nếu bạn không thể tìm **File** mà mình muốn mở bằng cách sử dụng các phím tắt trên **Places Bar** hoặc bằng cách tìm trong các **Folder** mà bạn thường xuyên lưu các tài liệu vào trong đó, thì bạn có thể nhập vào các

tiêu chuẩn tìm kiếm từ bên trong **Access** (trong hộp thoại **Open**, nhấp **Tools**, rồi nhấp **Find** – như hình trên).

### III. Phân loại một danh sách các File

1. Trên **Menu File**, nhấp **Open**.
2. Nhấp mũi tên kế **View**, trở đến **Arranges Icons**, rồi nhấp lệnh mà bạn muốn.

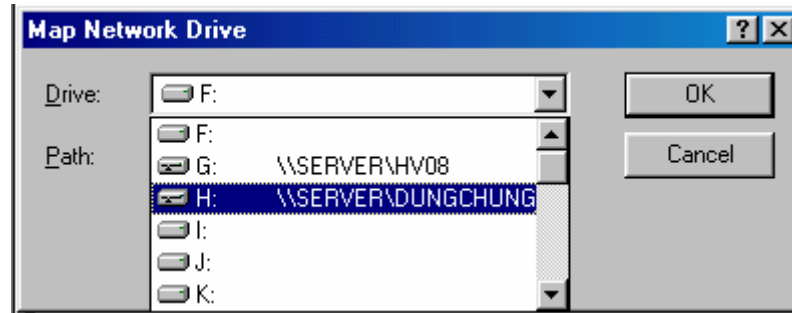


### Lưu ý:

Bạn có thể phân loại một danh sách các **File** trong các hộp thoại **Import** và **Link** trên **Menu File**, hoặc các hộp thoại **Database to Convert From**, **Database To Compact From**, **Database to Save As LVLDE**, và **Encryp/Decrypt Database** trên **Menu Tools**.

### IV. Nối kết vào một ổ mạng:

1. Trên **Menu File** nhấp **Open**
2. Nhấp **Tools**, rồi nhấp **Map Network Drive**.
3. Trong hộp thoại **Drive**, nhấp mẫu tự ổ đĩa mà bạn muốn sử dụng để kết nối vào mạng
4. Trong hộp **Path**, hãy nhập vào đường dẫn của **Folder** được chia sẻ mà bạn muốn nối vào.



## V. Tìm kiếm các File

1. Trên **Menu File**, nhấp **Open**
2. Nhấp **Tools**, rồi nhấp **Find**
3. Trong hộp **Look In**, hãy nhập ổ đĩa hoặc **Folder** mà bạn muốn tìm kiếm.
4. Để tìm tất cả các **Folder** con của **Folder** được chọn và hiển thị chứa các **File** đang được tìm, hãy chọn hộp kiểm **Search SubFolders**.
5. Trong hộp **Properties**, hãy nhấp **File Name**.
6. Trong hộp **Condition** hãy nhấp tùy chọn bạn muốn.
7. Trong hộp **Value** gõ nhập tất cả hoặc một phần của tên **File**.
8. Nhấp **Add to List**
9. nhấp **Find Now**.

### Lưu ý:

- Để tìm kiếm nhiều ổ đĩa cùng một lúc hãy gõ nhập tên mỗi một ổ đĩa được tách nhau bằng một dấu chấm phẩy trong hộp **Look In**.
- **Ví dụ**, gõ nhập C:\; D:\.
- Bạn sẽ không thể tìm một **File** bằng cách tìm kiếm **Text** bên trong một **cơ sở dữ liệu Microsoft Access** hoặc một **File** đề án **Microsoft Access**.
- Bạn cũng có thể tìm kiếm các **File** bằng cách sử dụng các hộp thoại **Import** và **Link** trên **Menu File**, hoặc các hộp thoại **Database to Convert From**, **Database To Compact From**, **Database to Save As MDE**, và **Encryp/Decrypt Database** trên **Menu Tools**.

## VI. Hủy bỏ một mục tìm kiếm đã được lưu

1. Trên **Menu File**, nhấp **Open**.
2. Nhấp **Tools**, rồi nhấp **Find**.
3. Nhấp **Open Search**.
4. Nhấp tên của mục tìm kiếm, rồi nhấp **Delete**.

**Lưu ý:**

Bạn cũng có thể xóa bỏ một mục tìm kiếm đã được lưu trong các hộp thoại Import và **Link** trên **Menu File**, hoặc các hộp thoại **Database to Convert From**, **Database To Compact From**, **Database to Save as MDE**, và **Encrypt/Decrypt Database** trên **Menu Tools**.

**VII. Tìm kiếm các File bằng cách sử dụng các tính chất File**

1. Trên **Menu File**, nhấp **Open**.
2. Nhấp **Tools**, rồi nhấp **Find**.
3. Trong hộp **Look In**, nhấp ổ đĩa hoặc **Folder** mà bạn muốn tìm kiếm.
- Để tìm kiếm nhiều ổ đĩa cùng một lúc, hãy gõ nhập tên ổ đĩa được tách rời nhau bằng một dấu chấm phẩy trong hộp **Look In**.
- **Ví dụ:** gõ nhập C:\;D:\
4. Để tìm kiếm tất cả các **Folder** con của **Folder** đã chọn đồng thời để hiển thị các **Folder** có chứa các **File** đang tìm kiếm, hãy chọn hộp kiểm **Search SubFolders**.
5. Trong phần Define more Criteria, hãy xác định tiêu chuẩn tìm kiếm bằng cách chọn các tùy chọn mà bạn muốn trong các hộp **Property**, **Condition**, và **Value**, rồi nhấp **Add to List**.

**Lưu ý:**

Nếu bạn đang tìm kiếm một **File Office** hơn là **cơ sở dữ liệu Microsoft Access** hoặc đề án **Microsoft Access Project**, thì bạn cũng có thể tìm kiếm một cụm từ bên trong **File**, chẳng hạn như "apple pie". Trong hộp **Property**, hãy chọn **Text or Property**, rồi trong hộp **Value**, hãy nhập vào cụm từ trong dấu ngoặc kép ("").

6. Để tạo ra tiêu chuẩn tìm bổ sung hãy nhấp **And** hoặc **Or**, rồi lặp lại bước 5.
  7. Chọn bất cứ tùy chọn nào khác mà bạn muốn.
- ⇒ Để có trợ giúp **Help** về một tùy chọn, hãy nhấp dấu ? rồi nhấp lên tùy chọn đó.
8. Để lưu tiêu chuẩn tìm kiếm sao cho bạn có thể sử dụng chúng sau này, hãy nhấp **Save Search**, gõ nhập một tên trong hộp **Name for this Search**, rồi nhấp **OK**.
- ⇒ Bạn có thể sử dụng các tên dài để mô tả mục tìm kiếm lên đến 80 ký tự và bạn cũng có thể đưa vào các khoảng trống.

9. Để bắt đầu cuộc tìm kiếm, hãy nhấp **Find Now**.

### **VIII. Tìm kiếm các File bằng cách sử dụng các tiêu chuẩn tìm kiếm đã được lưu:**

Bạn phải có một mục kiếm đã được lưu để sử dụng tiêu chuẩn tìm kiếm đã được lưu.

1. Trên **Menu File**, nhấp **Open**.
2. Nhấp **Tools**, rồi nhấp **Find**.
3. Nhấp **Open Search**.
4. Nhấp tên mục tìm kiếm bạn muốn dùng, rồi nhấp **Open**.
5. Nhấp **Find Now**.

#### **Lưu ý:**

Bạn cũng có thể tìm kiếm các **File** bằng cách sử dụng các hộp thoại **Import** và **Link** trên **Menu File**, hoặc các hộp thoại **Database to Convert From**, **Database To Compact From**, **Database to Save As MDE**, và **Encrypt/Decrypt Database** trên **Menu Tools**.

### **IX. Sử dụng các tính chất File để định vị trí các File**

#### **1. Bàn về việc sử dụng các tính chất file để định vị trí các File**

- ⇒ Các tính chất **File** là các chi tiết về một **File** nhằm giúp bạn nhận ra nó
- ⇒ **Ví dụ**, một tiêu đề mô tả, tên tác giả, chủ điểm, các từ khóa nhằm nhận rõ các chủ điểm hoặc các thông tin quan trọng khác về **File**. Bạn có thể cài đặt các tính chất **File** dành cho **cơ sở dữ liệu** mà bạn đang làm việc và về bất cứ **cơ sở dữ liệu Microsoft Access** nào, các đề án **Microsoft Access**, hoặc các trang truy cập dữ liệu.
- ⇒ Có ba kiểu tính chất khác nhau mà bạn có thể dùng để định vị trí các **File**.

#### **a. Các tính chất file được tạo sẵn (Built-in File Properties):**

- ⇒ Các tính chất **File** được tạo sẵn là các tính chất (chẳng hạn như tác giả, tiêu đề và chủ điểm) đã có để bạn có thể nhập một giá trị.
- ⇒ **Ví dụ**, bạn có thể nhập tên tác giả của **cơ sở dữ liệu** để có thể tìm kiếm tất cả các **cơ sở dữ liệu** do tác giả đó tạo ra.

#### **b. Các tính chất File tùy ý:**

Bạn có thể tạo các tính chất **File** tùy ý nhằm có chứa bất cứ giá trị nào mà bạn gán.

**Ví dụ**, bạn có thể tạo một tính chất **cơ sở dữ liệu** để lưu giữ số phiên bản của **cơ sở dữ liệu**.



### ***c. Các tính chất File được cập nhật tự động :***

- ⇒ ***Các số liệu thống kê chẳng*** hạn kích cỡ của ***cơ sở dữ liệu***, ngày tạo ra, ngày mà ***cơ sở dữ liệu được*** chỉnh sửa lần lượt là các tính chất ***cơ sở dữ liệu*** tự động được bảo quản cho bạn.
- ⇒ ***Ví dụ***, bạn có thể tìm kiếm tất cả các ***cơ sở dữ liệu*** được tạo sau ngày 3 tháng 3 năm 2004, hoặc được chỉnh sửa ngày hôm qua.

### ***2. Thông tin tổng hợp tài liệu của một cơ sở dữ liệu Microsoft Access hoặc đề án Microsoft Access***

- ⇒ Khi ***cơ sở dữ liệu*** đang mở hãy nhấp ***Database Properties*** trên ***Menu File***.
- ⇒ Nhấp nhãn ***Summary***, đổi chỉ định thông tin yêu cầu mà bạn muốn.
- ⇒ Để nhận được trợ giúp ***Help*** về một tùy chọn, hãy nhấp lên dấu ***?***, rồi nhấp lên tùy chọn đó.
- ⇒ Nhấp ***OK***.

### ***3. tạo các tính chất cơ sở dữ liệu tùy ý để sử dụng trong việc định vị một cơ sở dữ liệu Microsoft Access***

- ⇒ Với ***cơ sở dữ liệu*** đang mở hãy nhấp ***Database Properties*** trên ***Menu File***.
- ⇒ Nhấp nhãn ***Custom*** rồi gõ nhập kiểu giá trị dành cho tính chất mới trong hộp ***Name***, ***Type*** và ***Value***.
- ⇒ Lúc đã hoàn tất việc bổ sung các tính chất, hãy nhấp ***OK***.

### ***4. Chỉnh sửa hoặc xóa bỏ các tính chất cơ sở dữ liệu tùy ý***

- ⇒ với ***cơ sở dữ liệu*** đang mở, hãy nhấp ***Database Properties*** trên ***Menu File***. Nhấp chọn ***Custom***
- ⇒ nhấp tên của tính chất mà bạn muốn xóa bỏ hoặc chỉnh sửa trong cột ***Name*** trong hộp ***Properties***.
- ⇒ để chỉnh sửa tính chất, hãy gõ nhập các thay đổi trong các hộp ***Type*** hoặc ***Value***, rồi nhấp ***Modify***.
- ⇒ Để xóa bỏ tính chất này hãy nhấp ***Delete***.

#### ***Lưu ý:***

Nếu bạn thử chỉnh sửa tên của tính chất, thì nó trở thành một tính chất mới, và bạn chỉ có thể bổ sung nó. Để chỉnh sửa tên của tính chất này hãy xóa bỏ tính chất đang có và bổ sung nó với một tên mới.

### ***5. Xem một danh sách các đối tượng cơ sở dữ liệu trong cơ sở dữ liệu Microsoft Access đang có hoặc đề án Access.***

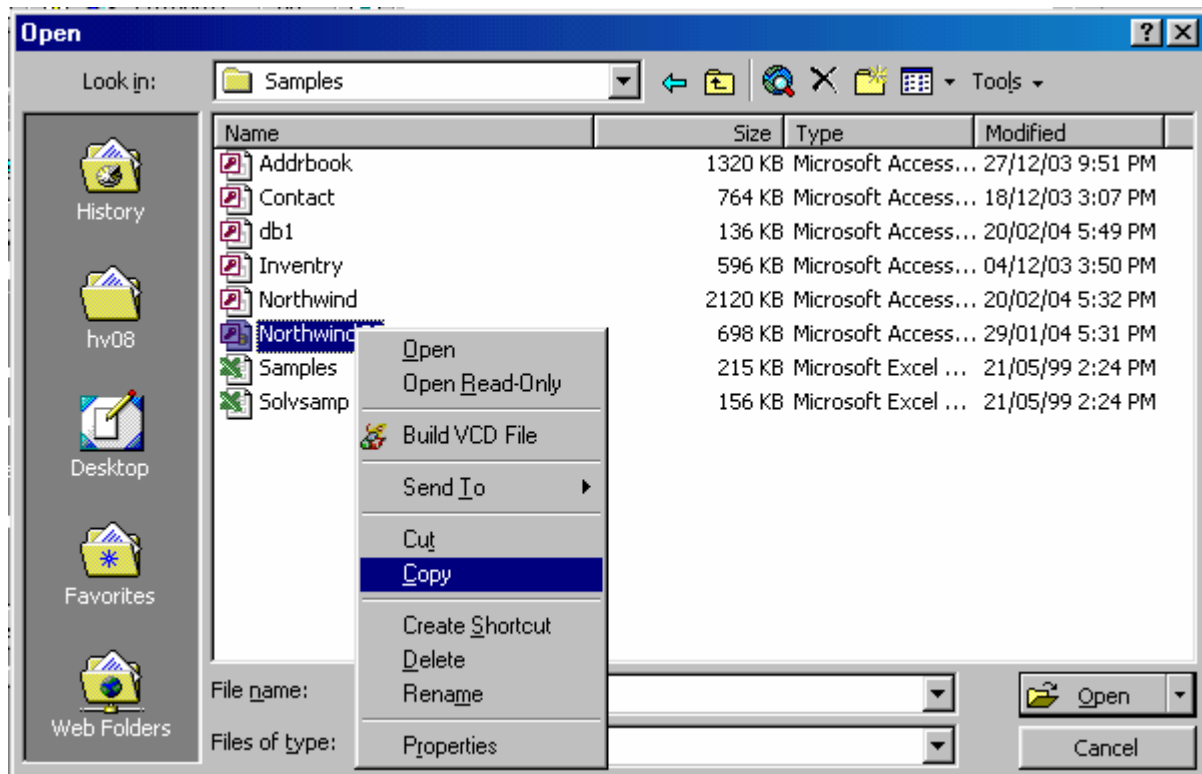
- ⇒ Với cơ sở dữ liệu Microsoft Access hoặc đề án Microsoft Access đang mở, hãy nhấp **Database Properties** trên Menu File
- ⇒ Nhấp chọn Contents

**Lưu ý:**

- ⇒ Ngoại trừ các tính chất trên nhãn **General** và **Statistic** trong hộp thoại **Database Properties** và các tính chất của một trang truy cập dữ liệu, các tính chất **File** đều có thể được xử lý bằng cách sử dụng Visual Basic for **Applications**.
- ⇒ Lúc bạn thử định vị một **cơ sở dữ liệu Microsoft Access**, đề án **Microsoft Access**, hoặc trang truy cập dữ liệu, bằng cách sử dụng hộp thoại **Open**, bạn có thể sử dụng tiêu chuẩn tìm kiếm để xác định các tính chất của **File** mà bạn đang tìm kiếm.
- ⇒ **Ví dụ**, bạn có thể tìm kiếm một **cơ sở dữ liệu** mà tính chất Contents của nó chứa một **Table** có tên là Suppliers.

**X. Sao chép một File**

1. Nhấp **Open**.
2. Trong hộp **Look In**, nhấp ổ đĩa hoặc **Folder** có chứa **File** mà bạn muốn sao chép.
3. Trong danh sách **Folder**, nhấp đúp lên các **Folder** cho đến khi bạn mở được **Folder** có chứa **File** mà mình muốn.  
Nếu bạn không tìm thấy **File** trong danh sách **Folder**, bạn có thể tìm kiếm nó.
4. Nhấp phải lên **File** mà bạn muốn sao chép, rồi nhấp **Copy** trên **Menu** tắt.

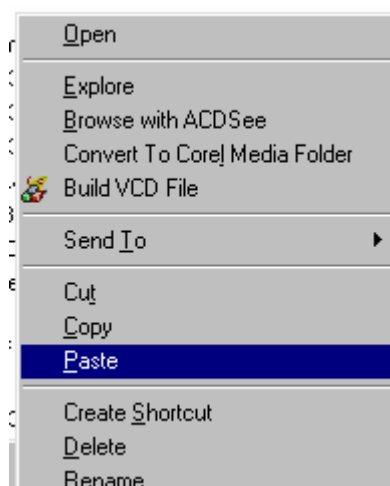


5. Trong hộp **Look In**, nhấp ổ đĩa hoặc **Folder** mà bạn muốn sao chép **File** vào đó.

6. Trong danh sách **Folder**, hãy nhấp lên các **Folder** cho đến lúc bạn mở được **Folder** mà mình muốn sao chép **File** vào đó

7. Nhấp phải lên danh sách **Folder** (phải chắc chắn có một **File** không được chọn), rồi nhấp **Paste** trên **Menu** tắt.

⇒ Lệnh **Paste** không có sẵn nếu một **File** được chọn trong danh sách **Folder**.



## XI. Xóa bỏ một File

### 1. Nhấp Open

2. Trong hộp thoại **Look In**, nhập ổ đĩa, **Folder** chứa **File** mà bạn muốn xóa bỏ.
3. Trong danh sách **Folder** hãy nhấp đúp các **Folder** cho đến khi bạn mở được **Folder** có chứa **File** mà mình muốn.
4. Nếu không tìm thấy **File** trong danh sách **Folder** bạn có thể tìm kiếm nó.
5. Nhấp phải lên **File** mà bạn muốn xóa bỏ đi, rồi chọn **Delete** trên **Menu** tắt (Menu tắt tương tự như hình trên – Phần Copy File).

## XII. Đặt lại tên một File

1. Nhấp **Open**
2. Trong hộp thoại **Look In**, nhập ổ đĩa, **Folder** chứa **File** mà bạn muốn đặt lại tên.
3. Trong danh sách **Folder** hãy nhấp đúp các **Folder** cho đến khi bạn mở được **Folder** có chứa **File** mà mình muốn.
4. Nếu không tìm thấy **File** trong danh sách **Folder** bạn có thể tìm kiếm nó.
5. Nhấp phải lên **File** mà bạn muốn, rồi chọn **ReName** trên **Menu** tắt (Menu tắt tương tự như hình trên – Phần Copy File).
6. Gõ nhập tên mới và nhấn **Enter**.

## XIII. Di chuyển một File:

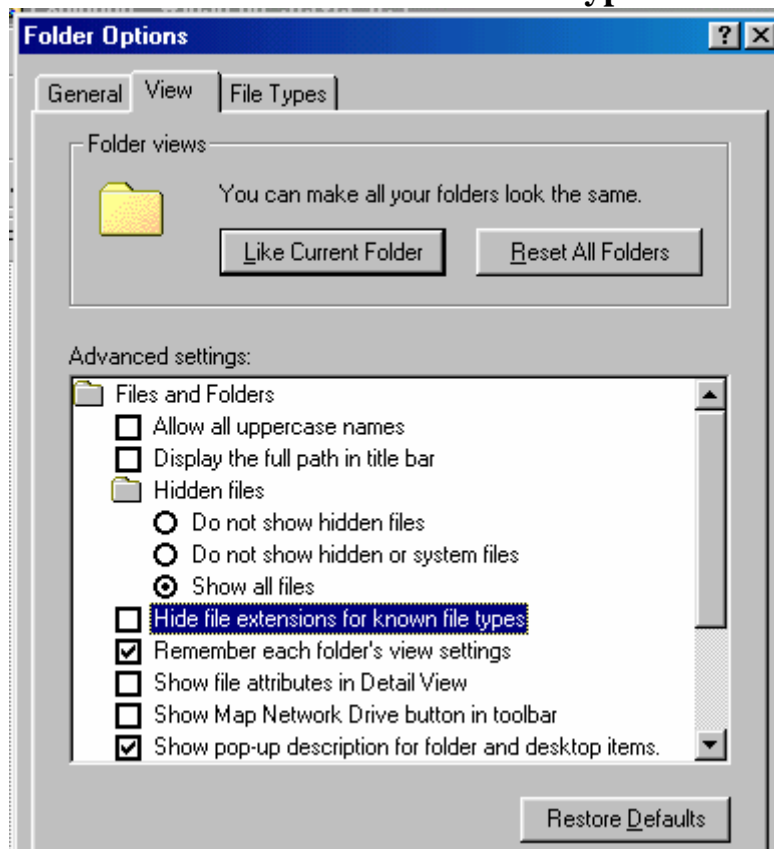
1. Nhấp **Open**
2. Trong hộp thoại **Look In**, nhập ổ đĩa, **Folder** chứa **File** mà bạn muốn di chuyển.
3. Trong danh sách **Folder** hãy nhấp đúp các **Folder** cho đến khi bạn mở được **Folder** có chứa **File** mà mình muốn.
4. Nếu không tìm thấy **File** trong danh sách **Folder** bạn có thể tìm kiếm nó.
5. Nhấp phải lên **File** mà bạn muốn, rồi chọn **Cut** trên **Menu** tắt.
6. Trong hộp thoại **Look In**, nhấp lên ổ đĩa, **Folder** chứa **File** mà bạn muốn di chuyển vào đó.
7. Trong danh sách **Folder** hãy nhấp đúp các **Folder** cho đến khi bạn mở được **Folder** mà mình muốn di chuyển vào đó.
8. Nhấp phải lên danh sách **Folder** (phải chắc chắn 1 **File** không được chọn) và nhấp **Paste** trên **Menu** tắt (Menu tắt tương tự như hình trên – Phần Copy File).

## Lưu ý:

Lệnh **Paste** không có sẵn nếu một **File** được chọn trong danh sách **Folder**.

#### XIV. Xem tất cả các kiểu File và phần mở rộng tên File.

1. Trong **My Computer**, hoặc trong Windows Explorer hãy mở **Folder** mà bạn muốn xem các **File** trong đó.
2. Trên **Menu View**, nhấp chọn **Option**.
3. Để xem tất cả các kiểu **File** hãy nhấp chọn Show All **File** trên nhãn **View**.
4. Để xem tất cả các phần mở rộng tên **File** trong Windows, hãy xóa hộp kiểm Hide MS-DOS **File** Extensions For **File** Types Thai Are Registered trên nhãn **View**.
5. Để xem tất cả các phần mở rộng tên **File** trong Windows, hãy xóa hộp kiểm Hide **File** Extensions For **Known** **File** Types trên nhãn **View**.



Lưu ý: Hai cài đặt sau cùng điều khiển tình trạng các phần mở rộng tên **File** có được hiển thị trong danh sách **Files of Type** của các hộp thoại **Open**, **Export**, **Import**, **Link** hay không.

#### XV. Nơi tối ưu để lưu giữ các File

Lần đầu tiên lúc bạn mở các hộp thoại **Open** và **File New Database** trong **Microsoft Access**, **Folder My Documents** là **Folder** hoạt động mặc định.

### 1. **Folder My Documents**

**Folder My Documents** là **Folder** tối ưu để lưu giữ cơ sở dữ liệu **Microsoft Access**, đề án **Microsoft Access**, các trang truy cập dữ liệu, và các **File** khác mà bạn đang làm việc, chẳng hạn như các tài liệu, các trang bảng tính, và các mục trình diễn. Để nhanh chóng truy cập các **File** trong **Folder My Documents**, hãy nhấp **My Documents** trên **Places Bar**.

### 2. **Folder Favorites**

**Folder Favorites** là một nơi lý tưởng để lưu trữ các biểu tượng tắt vào các **File** và **Folder** mà bạn dùng thường xuyên, kể cả các mục điều khiển vị trí mạng. **File** hoặc **Folder** gốc thì không di chuyển, thay vì vậy một biểu tượng tắt được tạo ra để đi vào **File** hoặc **Folder** đó. Việc lưu giữ các biểu tượng tắt trong **Folder Favorites** cho bạn một cách nhanh chóng để truy cập vào bất cứ **File** nào mà không cần phải nhớ nơi mà **File** đó đang ở. Để nhanh chóng truy cập vào các **File** trong **Folder Favorites**, hãy nhấp **Favorites** trên **Places Bar**.

### 3. **Web Folders**

**Web Folders** là một nơi lý tưởng để lưu các **File** của trang truy cập dữ liệu nhằm sử dụng trên Internet hoặc intranet. Các **File** này nằm trên **Web Server** nhằm cho phép những người sử dụng khác dễ dàng truy cập vào chúng. Để nhanh chóng truy cập các **File** được lưu giữ trên một **Web Server**, hãy nhấp **Web Folders** trên **Places Bar**.

## Chương 2

# Thiết kế cơ sở dữ liệu

### I. Khảo sát việc thiết kế một cơ sở dữ liệu

Trước khi bạn sử dụng **Microsoft Access** để thực sự xây dựng các **Table**, các **Form**, và các đối tượng khác sẽ làm nên **cơ sở dữ liệu** của bạn, điều quan trọng là phải dành thời gian để thiết kế **cơ sở dữ liệu**. Cho dù bạn đang sử dụng một **cơ sở dữ liệu Microsoft Access** hoặc một đề án **Microsoft Access**, thì việc thiết kế tốt **cơ sở dữ liệu** là nền tảng chủ yếu để tạo một **cơ sở dữ liệu** nhằm thực hiện những gì mà bạn muốn một cách hiệu quả, chính xác.

### II. Các bước để thiết kế một cơ sở dữ liệu

Chủ điểm này cung cấp thông tin về các bước căn bản để thiết kế một **cơ sở dữ liệu**.

#### 1. Xác định mục đích của cơ sở dữ liệu

Bước đầu tiên trong việc thiết kế một **cơ sở dữ liệu** đó là xác định mục đích và cách mà nó được dùng. Bạn cần biết thông tin nào bạn muốn từ cơ sở dữ liệu. Từ đó, bạn có thể xác định chủ điểm nào bạn cần để lưu trữ các sự kiện về nó (**Table**) và loại dữ liệu nào bạn cần lưu trữ về mỗi một chủ điểm (các trường trong **Table**).

#### 2. Xác định các Table mà bạn cần

Việc xác định các **Table** có thể là bước nghiêm ngặt nhất trong qui trình thiết kế **cơ sở dữ liệu**. Điều này do bởi các kết quả mà bạn muốn từ **Table** của mình - các **Report** mà bạn muốn in, các **Form** mà bạn muốn sử dụng, các câu hỏi mà bạn muốn trả lời - sẽ không cung cấp đầy đủ về các đầu mối về cấu trúc của các **Table** tạo ra chúng.

Bạn không cần thiết kế **Table** của bạn bằng cách sử dụng **Microsoft Access**. Thật sự ra, điều tốt hơn đó là cần phải phác thảo và hoạch định lại công việc thiết kế trước tiên trên giấy. Lúc bạn thiết kế **Table**, hãy chia nhãn các mảng thông tin bằng cách giữ các nguyên tắc thiết kế căn bản dưới đây:

⇒ Một **Table** không nên chứa thông tin trùng lặp và thông tin sẽ không được trùng lặp giữa các **Table**. ở phương diện này, một **Table** trong một

**cơ sở dữ liệu** quan hệ thì khác với một **Table** trong một trình ứng dụng FLAT-File chẳng hạn như một trang bảng tính.

- ⇒ Lúc mỗi mảng thông tin được lưu giữ chỉ trong một **Table**, thì bạn cập nhật nó ở một chỗ. Điều này hữu hiệu hơn và nó cũng hạn chế khả năng các mục trùng lặp có chứa thông tin khác nhau.
- ⇒ **Ví dụ**, bạn sẽ muốn lưu giữ mỗi một địa chỉ khách hàng và số điện thoại của khách hàng đó chỉ một lần trên một **Table** mà thôi.
- ⇒ Mỗi **Table** cũng nên chứa thông tin về mỗi một chủ điểm.
- ⇒ Lúc mỗi một **Table** có chứa một sự kiện duy nhất về một chủ điểm, thì bạn có thể lưu giữ thông tin về mỗi một chủ điểm một cách độc lập với các chủ điểm khác.
- ⇒ Lấy **Ví dụ**, bạn nên lưu giữ địa chỉ của khách hàng ở một **Table** khác với đơn đặt hàng của khách hàng, để bạn có thể xóa bỏ một đơn đặt hàng mà vẫn giữ lại thông tin về khách hàng đó.

### 3. Xác định các trường (Field) mà bạn cần

- ⇒ Mỗi một **Table** có chứa thông tin về chủ điểm giống nhau, và mỗi một trường trong một **Table** thì có chứa các sự kiện riêng biệt về chủ điểm của **Table**. Lấy **Ví dụ**, một **Table** khách hàng có thể chứa các trường tên công ty, trường địa chỉ, trường thành phố, và trường số điện thoại. Lúc phác thảo các trường cho mỗi một **Table**, hãy nhớ các lời khuyên sau đây:
- ⇒ Liên hệ mỗi một trường trực tiếp với chủ điểm của **Table**.
- ⇒ Đừng đưa vào dữ liệu được dẫn xuất hoặc được tính toán (tức là dữ liệu kết quả của một biểu thức hay biến đổi).
- ⇒ Đưa vào tất cả thông tin cần thiết.
- ⇒ Lưu giữ thông tin theo các phần hợp lý nhỏ nhất (**Ví dụ**, First Name và Last Name hơn là Name).

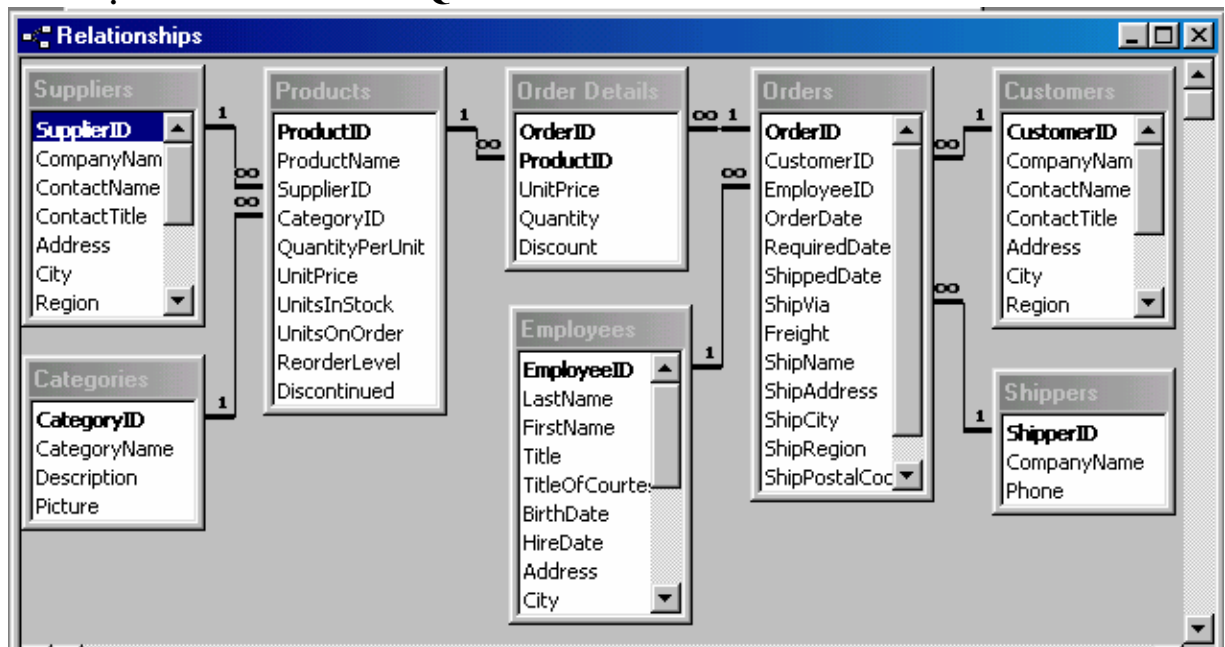
### 4. Chỉ định trường hoặc các trường với các giá trị đồng nhất trong với một Record (bản ghi)

- ⇒ Để **Microsoft Access** nối thông tin được lưu giữ trong các **Table** riêng biệt
- ⇒ **Ví dụ**, để nối một khách hàng với tất cả các đơn đặt hàng của khách hàng đó mỗi **Table** trong **cơ sở dữ liệu** của bạn phải chứa một trường hoặc tập hợp các trường có các đặc tính đồng nhất với mỗi một **Record** riêng biệt trong **Table**. Một trường hoặc tập hợp các trường như thế được gọi là một từ khóa (**Primary key**).



## 5. Xác định các quan hệ giữa các **Table**

- ⇒ Bây giờ chúng ta đã chia thông tin thành các **Table** và các trường khóa chính định danh, bạn cần một phương pháp để báo cho **Microsoft Access** cách đưa thông tin có liên quan trở về với nhau theo một cách thức đầy đủ ý nghĩa. Để thực hiện điều này, bạn cần xác định các quan hệ giữa các **Table** trong một **cơ sở dữ liệu Microsoft Access**.
- ⇒ Bạn sẽ nhận thấy rằng việc xem các quan hệ trong một **cơ sở dữ liệu** hiện đang được thiết kế tốt là một điều ích lợi.
- ⇒ **Ví dụ**, mở **cơ sở dữ liệu** mẫu Northwind, rồi nhấp Relationships trên menu **Tools** để xem các quan hệ giữa các **Table** của nó. Hoặc có thể bạn muốn xem các quan hệ trong đề án **Access** mẫu Northwind và **cơ sở dữ liệu** mẫu **Microsoft SQL Server**.



## 6. Tinh chỉnh mục thiết kế của bạn

- ⇒ Sau khi đã thiết kế các **Table**, **Field**, và các quan hệ mà bạn cần, đã đến lúc phải nghiên cứu việc thiết kế và dò tìm bất cứ sai lầm nào hiện vẫn còn.
- ⇒ Việc thay đổi thiết kế **cơ sở dữ liệu** bây giờ thì dễ dàng hơn là thực hiện điều đó sau khi bạn đã điền dữ liệu vào các **Table**.
- ⇒ Sử dụng **Microsoft Access** để tạo ra các **Table**, chỉ định các quan hệ giữa các **Table** và nhập dữ liệu mẫu đủ trong các **Table** để bạn có thể thử nghiệm bản thiết kế. Để thử nghiệm các quan hệ trong **cơ sở dữ liệu**, hãy xem bạn có thể tạo các **Query** để nhận các câu trả lời mà mình muốn không. Việc tạo bảng phác thảo thô của các **Form** và **Report**

cũng như xem chúng có minh họa dữ liệu theo ý bạn muốn hay không. Hãy tìm kiếm các mục trùng lặp không cần thiết về dữ liệu và hạn chế chúng. Nếu bạn tìm thấy các sự cố, thì hãy tinh chỉnh việc thiết kế.

### 7. Nhập dữ liệu và tạo các đối tượng cơ sở dữ liệu khác

⇒ Lúc bạn hài lòng với cấu trúc **Table** đáp ứng được các nguyên tắc thiết kế được mô tả ở đây, thì đã đến lúc bạn phải tiếp tục bổ sung tất cả dữ liệu đang có vào **Table**. Sau đó, bạn có thể tạo ra bất cứ **Query**, **Form**, **Report**, các trang truy cập dữ liệu, **Macro** và các Module mà bạn muốn.

### 8. Sử dụng công cụ phân tích Microsoft Access

⇒ **Microsoft Access** có chứa hai công cụ có thể giúp bạn tinh chỉnh việc thiết kế **cơ sở dữ liệu** của **Microsoft Access**. **Table Analyzer Wizard** có thể phân tích việc thiết kế một **Table** một lần, có thể đưa ra các cấu trúc **Table** mới và các quan hệ nếu có và có thể chia một **Table** thành các **Table** quan hệ mới nếu điều này có ý nghĩa.

⇒ **Performance Analyzer** có thể phân tích toàn bộ **cơ sở dữ liệu** và đưa ra một số đề nghị để cải tiến nó. **Wizard** cũng có thể thực thi các phần đề nghị này.

⇒ Để có những ý tưởng bổ sung về việc thiết kế một **cơ sở dữ liệu Microsoft Access**, bạn có thể muốn xem phần thiết kế của **cơ sở dữ liệu** mẫu Northwind hoặc một trong những **cơ sở dữ liệu** mà bạn tạo ra với **Database Wizard**.

⇒ Hãy xem **Microsoft SQL Server Documentation** để có những ý tưởng bổ sung về việc phân tích **cơ sở dữ liệu Microsoft SQL Server** nhằm cải tiến hoạt động của nó.

## III. Tạo một cơ sở dữ liệu mới

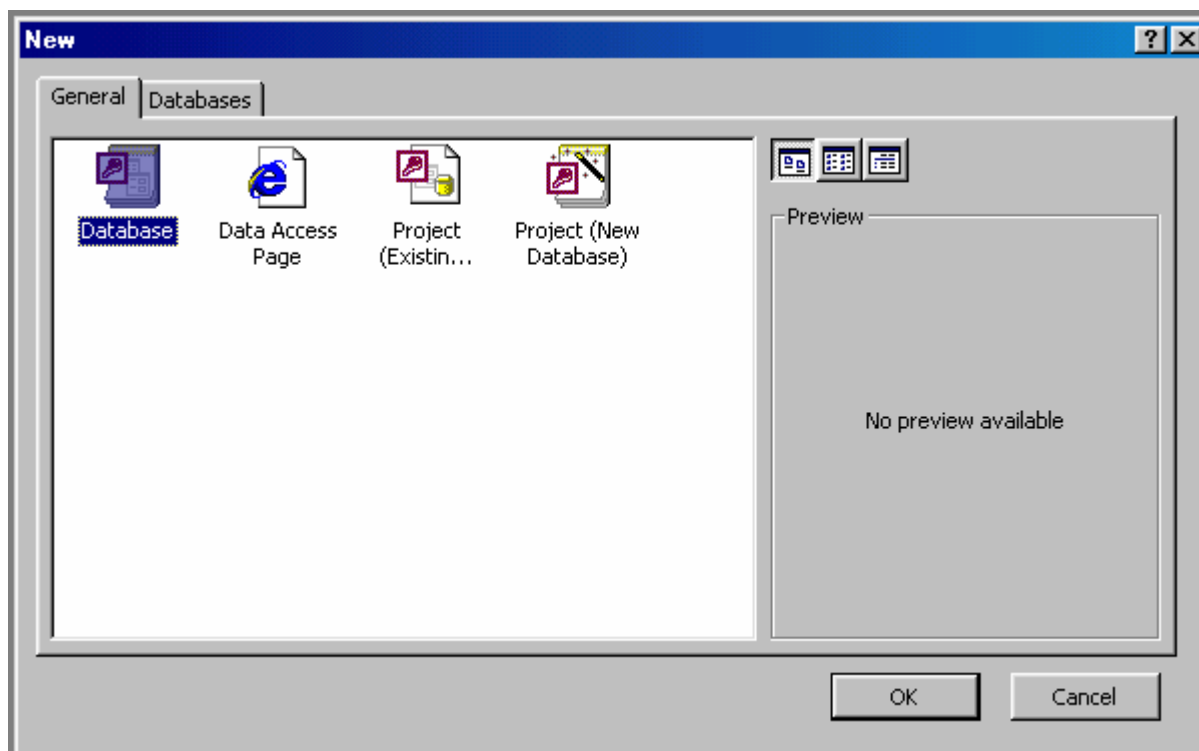
Trên Menu File gọi New

Hoặc ấn tổ hợp phím Ctrl\_N, sẽ cho ta hộp thoại New sau (hình):

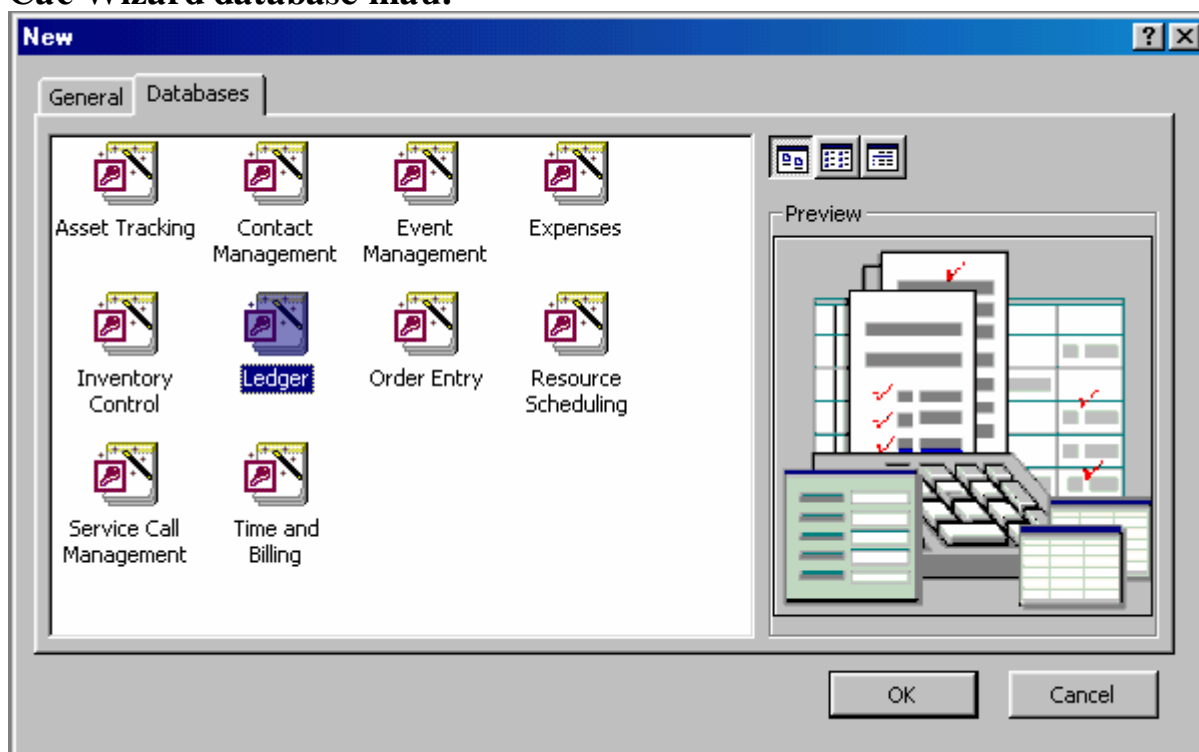
Trong phần gờ Tab General có các mục sau:

Database:	tạo cơ sở dữ liệu mới
Data access pages:	tạo cơ sở dữ liệu với trang Web HTML
Project (existing database):	tạo dự án với cơ sở dữ liệu có sẵn
Project (New database):	tạo dự án với cơ sở dữ liệu mới

Trong gờ Tab database: là các Wizard Database có sẵn giúp ta dễ thiết kế tạo dự án với cơ sở dữ liệu theo mẫu có sẵn theo hướng dẫn của Wizard Database.

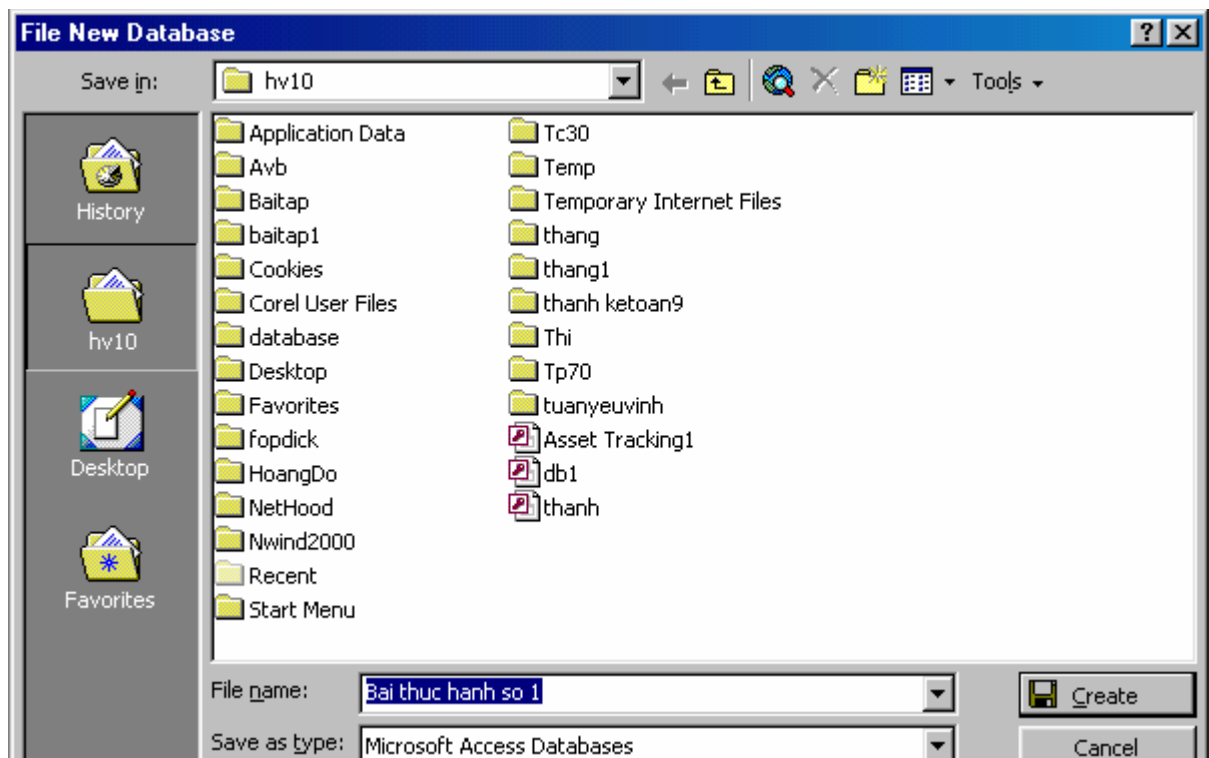


### Các Wizard database mẫu:



**Đánh tên vào hộp File name (ví dụ: Bai thuc hanh so 1)**

Chú ý: Trong danh sách **Folder**, hãy nhấp lên các **Folder** cho đến lúc bạn mở được **Folder** mà mình muốn lưu **File** vào đó.



#### IV. Khảo sát việc tạo một trình ứng dụng

- ⇒ Sau khi bạn quen thuộc với **Microsoft Access** và đã học cách tạo các **Table, Query, Form, Report**, các thanh lệnh tùy ý, v.v..., bạn có thể bổ sung các **Macro** và mã **Visual Basic for Applications** để buộc các đối tượng này lại với nhau trong một trình ứng dụng. Một trình ứng dụng sẽ tổ chức các tác vụ có quan hệ sao cho người dùng có thể tập trung vào một công việc trong tầm tay, chứ không theo cách mà trình ứng dụng hoạt động hoặc theo cách mà chương trình được dùng để phát triển trình ứng dụng. Để biết thêm thông tin về việc tạo một trình ứng dụng, hãy xem phần tạo các trình ứng dụng với **Forms** và **Reports**.
- ⇒ Bạn cũng có thể tham khảo Office Assistant về các chủ điểm đặc biệt này.
- ⇒ Bạn cũng có thể điều khiển hình thức và bản chất của các trình ứng dụng lúc nó khởi đầu bằng cách cài đặt các tùy chọn trong hộp thoại Startup (trên **Menu Tools**).
- ⇒ **Ví dụ**, bạn có thể chỉ **Form** nào trình ứng dụng của bạn sẽ hiển thị lúc khởi động, hoặc bạn có thể chỉ định **Text** mới xuất hiện trong thanh tiêu đề trong khi trình ứng dụng đang chạy.
- ⇒ Nếu bạn dự kiến phân bố trình ứng dụng của mình cho những người sử dụng khác, bạn cần phải xác lập tính an toàn để bảo vệ các đối tượng mà bạn đã tạo ra, cũng như dữ liệu trong **Table** của mình.

- ⇒ Sau khi trình ứng dụng của bạn đã được phân bố, hãy vẫn sử dụng các công cụ do **Microsoft Access** cung cấp để nén định kỳ và sửa chữa **cơ sở dữ liệu**. Phải bảo đảm cung cấp cho người sử dụng một cách để điều khiển các vị trí nhằm thực hiện tốt điều này.
- ⇒ Hiện thị hoặc thay đổi các tính chất tổng quát của một đối tượng **cơ sở dữ liệu**.
  1. Trong mục **Objects** trong cửa sổ **Database**, hãy nhấp kiểu đối tượng **cơ sở dữ liệu** của các tính chất bạn muốn thay đổi
  2. Nhấp đối tượng **cơ sở dữ liệu** có chứa các tính chất bạn muốn thay đổi.
  3. Nhấp **Properties** trên thanh công cụ.
 Để có trợ giúp **Help** về một mục, hãy nhấp ? rồi nhấp lên mục đó.

#### Lưu ý

- ⇒ Nội dung của trường **Description** trong hộp thoại **Properties** là những gì được hiển thị bên dưới **Description** trong cửa sổ **Database** lúc bạn nhấp **Details** trên thanh công cụ.
- ⇒ Trong đề án **Microsoft Access**, bạn không thể thay đổi các tính chất của một **Table**, **View**, sơ đồ **Database**, hoặc thủ tục được lưu trữ bởi vì các đối tượng này nằm trong **cơ sở dữ liệu Microsoft SQL Server**. Tuy nhiên, bạn có thể thay đổi các tính chất từ một **Form**, **Report**, **Macro**, hoặc module, bởi vì các đối tượng này nằm bên trong đề án **Access**, chứ không nằm trong **cơ sở dữ liệu Microsoft SQL Server**; bạn cũng có thể thay đổi các tính chất của một trang truy cập dữ liệu.

### V. Hiện thị hoặc thay đổi tính chất tổng quát của một đối tượng cơ sở dữ liệu

1. Trong mục **Objects** trong cửa sổ **Database**, hãy nhấp kiểu đối tượng cơ sở dữ liệu của các tính chất bạn muốn thay đổi
  2. Nhấp đối tượng cơ sở dữ liệu có chứa các tính chất bạn muốn thay đổi.
  3. Nhấp **Properties** trên thanh công cụ.
- Để có trợ giúp **Help** về một mục, hãy nhấp ? rồi nhấp lên mục đó.

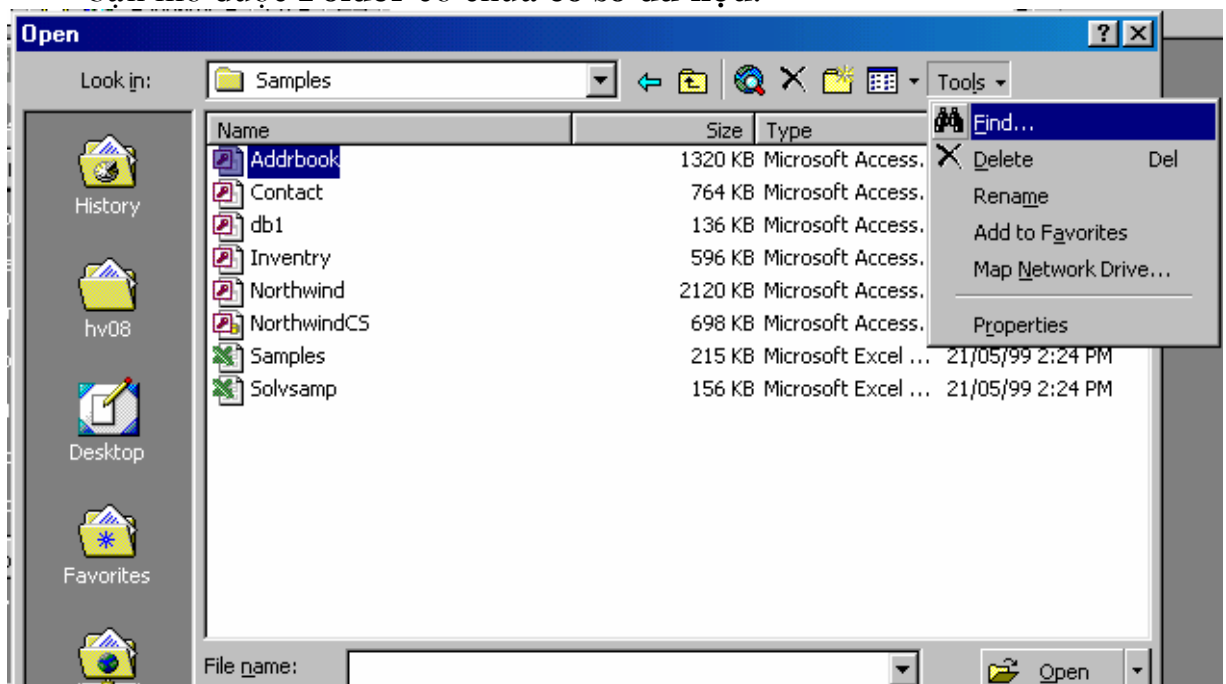
#### Lưu ý

- ❖ Nội dung của trường **Description** trong hộp thoại **Properties** là những gì được hiển thị bên dưới **Description** trong cửa sổ **Database** lúc bạn nhấp **Details** trên thanh công cụ.
- ❖ Trong đề án **Microsoft Access**, bạn không thể thay đổi các tính chất của một **Table**, **View**, sơ đồ **Database**, hoặc thủ tục được lưu trữ bởi vì các

đối tượng này nằm trong cơ sở dữ liệu Microsoft SQL Server. Tuy nhiên, bạn có thể thay đổi các tính chất từ một Form, Report, macro, hoặc module, bởi vì các đối tượng này nằm bên trong đề án Access, chứ không nằm trong cơ sở dữ liệu Microsoft SQL server; bạn cũng có thể thay đổi các tính chất của một trang truy cập dữ liệu.

## VI. Mở một cơ sở dữ liệu Microsoft Access

1. Trên men **File**, nhấp **Open**.
2. Nhấp một biểu tượng trong **Places Bar**, hoặc trong hộp **Look In**, nhấp ổ đĩa hoặc **Folder** có chứa **cơ sở dữ liệu Microsoft Access** mà bạn muốn.
3. Trong danh sách **Folder**, hãy nhấp đúp lên các **Folder** cho đến khi bạn mở được **Folder** có chứa **cơ sở dữ liệu**.



⇒ Nếu bạn không thể tìm thấy **cơ sở dữ liệu** mà mình muốn mở, hãy nhấp **Tools**, rồi nhấp **Find** (như đã nói trong chương 1). Trong hộp thoại **Find**, nhập vào tiêu chuẩn tìm kiếm bổ sung. Để có trợ giúp về một tùy chọn, hãy nhấp dấu ? và nhấp lên tùy chọn đó.

**Thực hiện một trong các thao tác sau đây:**

⇒ Để mở **cơ sở dữ liệu** nhằm truy cập chia sẻ trong một môi trường nhiều người dùng, sao cho bạn và những người khác đều có thể đọc và viết lên **cơ sở dữ liệu**, hãy nhấp **Open**.

- ⇒ Để mở cơ sở dữ liệu với truy cập thực thi (**Exclusive Access**), hãy nhập mũi tên nằm kê' bên nút **Open**, rồi nhập **Open Exclusive**.
- ⇒ Để mở cơ sở dữ liệu ở dạng truy cập chỉ đọc (**Read-Only Access**) sao cho bạn có thể xem nó nhưng không thể chỉnh sửa nó, hãy nhập mũi tên kê' nút **Open**, rồi nhập **Open Read-Only**.
- ⇒ Để mở cơ sở dữ liệu ở dạng truy cập chỉ đọc và cũng để ngăn những người sử dụng khác khởi mở nó, hãy nhập mũi tên kê' nút **Open**, rồi nhập **Open Exclusive Read-Only**.
- ⇒ Để có nội dung về một tùy chọn trong hộp thoại hãy nhấp dấu ?, rồi nhấp lên mục đó.

#### Lưu ý:

- ⇒ Để xem trong hộp thoại **Open** một danh sách các bước tắt để vào cơ sở dữ liệu mà bạn đã mở trước đây, hãy nhấp **History** trên **Places Bar**.
- ⇒ Để mở một trong các cơ sở dữ liệu mà bạn trước đây mở sau cùng, hãy nhấp tên của **File** nằm ở đáy của **Menu File**. **Microsoft Access** mở cơ sở dữ liệu với các cài đặt tùy chọn giống nhau mà nó đã có ở lần cuối cùng khi bạn mở. Nếu danh sách của các **File** vừa mới được sử dụng không hiển thị, hãy nhấp **Options** trên **Menu Tools**, nhấp nhãn **General**, rồi chọn hộp kiểm **Recently used File List**.
- ⇒ **Microsoft Access** có thể tự động tạo một đó bạn có thể sử dụng để mở trực tiếp cơ sở dữ liệu ở lần kế tiếp.
- ⇒ Bạn có thể trực tiếp mở một **File** dữ liệu trong một dạng **File** ngoại chẳng hạn **DBASE**, **Paradox**, **Microsoft Exchange**, **Microsoft Excel**; bạn cũng có thể trực tiếp mở nguồn dữ liệu **ODBC** nào, chẳng hạn **Microsoft SQL Server** hoặc **Microsoft Foxpro**. **Microsoft Access** tự động tạo một cơ sở dữ liệu **Access** mới trong cùng **Folder** với **File** dữ liệu và bổ sung các liên kết vào mỗi một **Table** trong cơ sở dữ liệu ngoại.

## VII. Tự động tạo một bước tắt vào một cơ sở dữ liệu Access hoặc đề án Access trong Folder Favorites

- ⇒ **Folder Favorites** lưu giữ các bước tắt để vào các cơ sở dữ liệu **Microsoft Access**, các đề án **Microsoft Access**, các **Folder** và các **File** khác mà bạn sử dụng thường xuyên, kể cả các vị trí điều khiển, để cho bạn không cần phải tìm kiếm các **File** từng lúc. **File** gốc hoặc **Folder** gốc thì không di chuyển.

### 1. Nhấp **Open** trên thanh công cụ

2. Để tạo một bước tắt vào một **Folder**, **Folder** con, **cơ sở dữ liệu Access** hoặc đề án **Access** mới xuất hiện trong danh sách bên dưới hộp **Look In**, hãy chọn **File** hoặc **Folder** đó, nhấp **Tools**, rồi nhấp **Add To Favorites**

### VIII. Bổ sung cơ sở dữ liệu Access hiện hành vào Folder Favorites

⇒ Bạn có thể bổ sung **cơ sở dữ liệu Microsoft Access** hiện hành vào **Folder Favorites** để có thể mở nó một lần nữa mà không cần phải nhớ đường dẫn mà bạn đã nhập để có được **File** lần đầu tiên.

1. Nếu thanh công cụ **Web** không hiển thị, thì trên **Menu View** hãy trở đến **Toolbars**, rồi nhấp **Web**.
2. Trên thanh công cụ **Web**, nhấp **Favorites** rồi nhấp **Add to Favorites**.
3. nhấp **Add**

### IX. Mở một cơ sở dữ liệu Microsoft Access trên mạng

- ⇒ Trên thanh **Menu** file nhấn **Open**
- ⇒ Trong hộp **File Name** gõ nhập một địa chỉ UNC dành cho việc chia sẻ mạng mà bạn muốn, rồi nhấn **Enter**.
- ⇒ **Ví dụ:** gõ nhập **\\Computer1\Files** để mở một **cơ sở dữ liệu Microsoft Access** hoặc đề án **Microsoft Access** trên thư mục **File** chia sẻ của **Computer1 Server**.
- ⇒ Trong danh sách **Folder**, định vị trí và mở **cơ sở dữ liệu Microsoft Access** hoặc đề án **Microsoft Access** mà bạn muốn.

### X. Thay đổi diện mạo của các đối tượng trong cửa sổ Database

- ⇒ Trong cửa sổ **Database**, nhấp chuột một trong các nút sau đây trên thanh công cụ của cửa sổ **Database**: **Large Icons**, **Small Icons**, **List** hoặc **Details**.
- ⇒ Để xếp loại các đối tượng, trên **Menu View**, hãy trở đến **Arrange Icons**, rồi nhấp một trong các mục sau đây: **By Name**, **by Type**, **by Created**, hoặc **by Modified**.

### XI. Hiển thị hoặc che giấu các biểu tượng tắt để tạo đối tượng cơ sở dữ liệu mới:

- ⇒ Lúc bạn nhấp **Objects** trong cửa sổ **Database** và nhấp một kiểu đối tượng, danh sách **Objects** có chứa các biểu tượng tắt mà bạn có thể dùng để tạo đối tượng **cơ sở dữ liệu** mới của kiểu đó.



- ⇒ Ví dụ, lúc bạn nhấp **Report**, các biểu tượng tắt **Created Report** in **Design View** và **Create Report by using Wizard** xuất hiện trong danh sách **Objects**. Bạn có thể hiển thị hoặc che giấu các biểu tượng tắt của đối tượng mới này.
- ⇒ Trên **Menu Tools**, nhấp **Options**. Nhấp nhãn **View**.

**Thực hiện một trong các thao tác sau đây:**

- ⇒ Để hiển thị các biểu tượng tắt của đối tượng mới trong cửa sổ **Database**, hãy chọn hộp kiểm **New Object ShortCut**.
- ⇒ Để che giấu các biểu tượng tắt của đối tượng mới, hãy xóa hộp kiểm **New Object ShortCut**.

## **XII. Xóa bỏ một đối tượng cơ sở dữ liệu**

1. Đóng đối tượng **cơ sở dữ liệu** mà bạn muốn xóa bỏ. Trong một môi trường nhiều người dùng, hãy khẳng định rằng tất cả những người sử dụng đều đã đóng đối tượng **cơ sở dữ liệu cần xóa**.
2. Bên dưới mục **Objects** trong cửa sổ **Database**, hãy nhấp kiểu đối tượng trong **cơ sở dữ liệu** mà bạn muốn xóa bỏ.
3. Nhấp đối tượng trong danh sách **Object**, rồi nhấn **DELETE**.

**Lưu ý:**

Lúc bạn xóa bỏ một trang truy cập dữ liệu từ **cơ sở dữ liệu** hiện hành, **Microsoft Access** sẽ hỏi bạn có muốn xóa bỏ liên kết và trang, hoặc chỉ xóa bỏ liên kết. Việc chỉ xóa bỏ liên kết sẽ xóa đi trang đó khỏi **cơ sở dữ liệu** hiện hành, nhưng nó sẽ không xóa **File HTML** khỏi máy tính của bạn.

## **XIII. Lưu một đối tượng cơ sở dữ liệu**

1. Nhấp **Save** trên thanh công cụ.
2. Nếu bạn đang lưu đối tượng **cơ sở dữ liệu** lần đầu tiên, hãy gõ nhập tên tuân theo quy tắc đặt tên đối tượng **cơ sở dữ liệu** trong hộp thoại **Save As**, rồi nhấp **OK**. Nếu bạn nhấp **cancel** trong hộp **Save as**, thì đối tượng không được lưu.

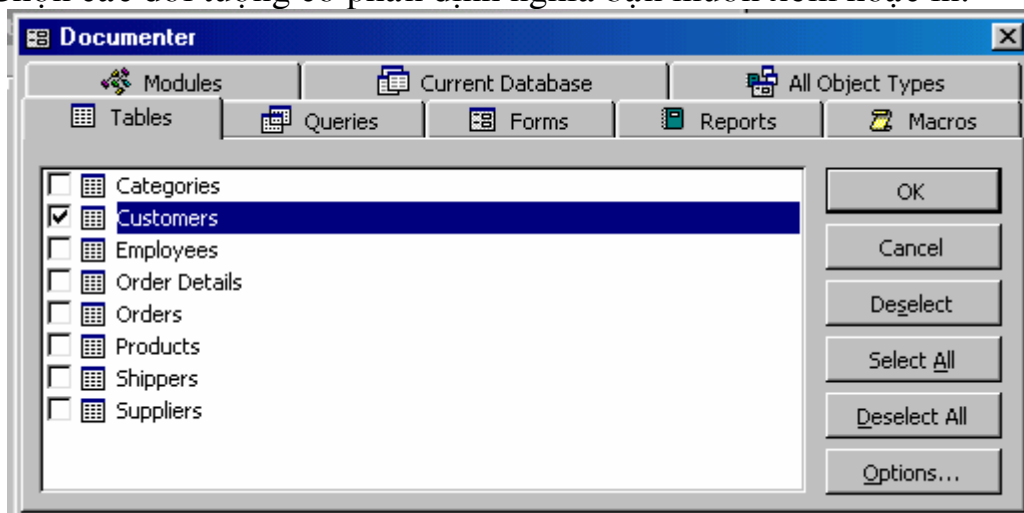
**Lưu ý:**

- ❖ Bạn không cần lưu một dữ liệu mới trong các **Record**. **Access** tự động lưu một **Record** lúc bạn thực hiện bất cứ điều nào sau đây: di chuyển chủ điểm đến một **Record** khác; đóng **Form** đang hoạt động, trang dữ liệu, trang truy cập dữ liệu, hoặc **cơ sở dữ liệu**; hoặc thoát khỏi **Access**.

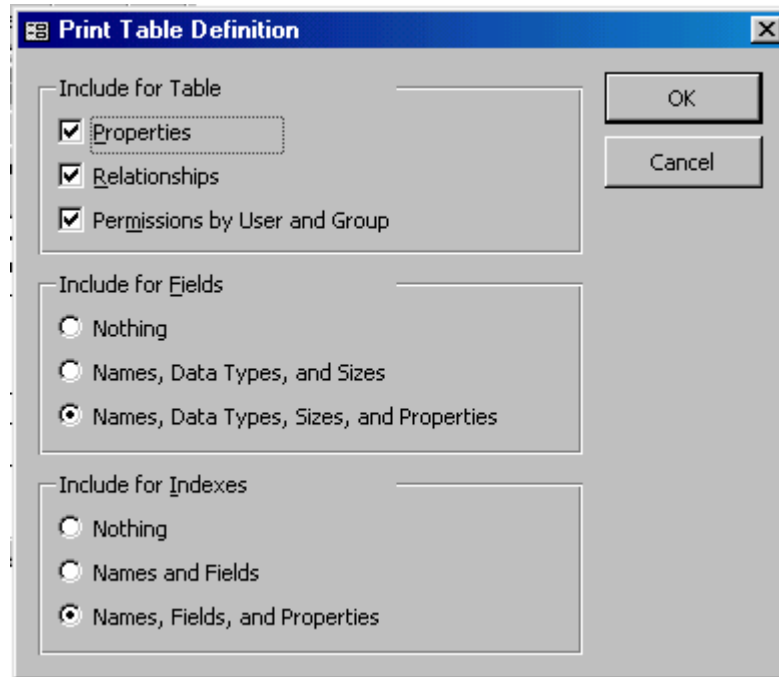
- ❖ Nếu bạn thử lưu một **Query** mới, và bạn nhận được một thông báo lỗi rằng tên mà bạn đã nhập hiện đang có ở một đối tượng khác, hãy thử sử dụng một tên khác. **Cơ sở dữ liệu** có thể chứa các **Query** đã được tạo ra trong mã Visual Basic for **Applications** (VBA) với **ActiveX Data Object** (ADO). Các **Query** được tạo ra với ADO thì không xuất hiện trong danh sách **Object** trong cửa sổ **Database**.

#### XIV. Xem, in, xuất, hoặc lưu các đặc trưng thiết kế các đối tượng cơ sở dữ liệu

- ❖ Trên **Menu Tools**, trở đến **Analyze**, rồi nhấn **DocumEnter**.
- ❖ Nhấp nhãn tương ứng với kiểu đối tượng **cơ sở dữ liệu** mà bạn muốn xem hoặc in.
- ❖ Bạn có thể nhấp nhãn **All Object Types** để hiển thị một danh sách hoàn chỉnh các đối tượng trong **cơ sở dữ liệu**.
- ❖ Chọn các đối tượng có phần định nghĩa bạn muốn xem hoặc in.



- ❖ Nhấp **Options** để chỉ định tính năng nào của đối tượng được chọn bạn muốn in, rồi nhấn **OK**.



- ❖ Nhấp **OK**

*Bây giờ bạn có thể thực hiện một trong các thao tác sau đây:*

- ❖ Để in các định nghĩa, hãy nhấp **Print** trên thanh công cụ.
- ❖ Bạn có thể muốn kiểm tra lại chiều dài của các định nghĩa trước khi in, bởi vì một số định nghĩa, đặc biệt là các định nghĩa dành cho các **Form** và **Report**, có thể dài nhiều trang.
- ❖ Để xuất định nghĩa sang một **File HTML**, một trang bảng tính Excel, một **File** định dạng Rich **Text Format**, một **File** text MS\_DOS, hoặc một **File** trong dạng **Report Snapshot**, hãy nhấp **Export** trên **Menu File**, rồi hoàn tất hộp thoại **Export Report** "**ReportName**" to.

**Chương 3:**

# Làm việc với Table

**I. Table : Nội dung và cách làm việc**

- ❖ **Microsoft Access** cung cấp hai cách để tạo một **Table**. Bạn có thể tạo một **Table** trống để nhập dữ liệu của riêng mình, hoặc bạn có thể tạo một **Table** bằng cách sử dụng dữ liệu đang có từ nguồn khác.
- ❖ Một **Table** là một tập hợp dữ liệu về một chủ điểm đặc biệt, chẳng hạn các sản phẩm hoặc các nhà cung cấp. Việc sử dụng một **Table** riêng biệt cho mỗi một chủ điểm có nghĩa rằng bạn lưu trữ dữ liệu đó chỉ một lần, để làm cho **cơ sở dữ liệu** của bạn hiệu quả hơn, và giảm thiểu các lỗi nhập dữ liệu. Các **Table** tổ chức dữ liệu thành các cột (được gọi là trường – **Field**) và các hàng (được gọi là các bản ghi – **Record**).
- ❖ Một trường nói chung thì có liên quan đến hai **Table** để **Microsoft Access** có thể đưa dữ liệu lại với nhau từ hai **Table** này nhằm xem chỉnh sửa hoặc in.
- ❖ Trong khung xem **Table Design View**, bạn có thể tạo toàn bộ một **Table** từ ban đầu, hoặc bổ sung, xóa bỏ, hoặc tùy nghi sử dụng các trường trong một **Table** đang có.
- ❖ Trong một khung xem **Table DataSheet View**, bạn có thể bổ sung, chỉnh sửa, xem, hoặc thực hiện những công việc khác với dữ liệu trong một **Table**. Bạn cũng có thể hiển thị các **Record** từ các **Table** liên quan với **Table** hiện hành bằng cách hiển thị các trang dữ liệu con bên trong trang dữ liệu chính. Với một vài hạn chế, bạn có thể làm việc với dữ liệu trong các trang dữ liệu con theo nhiều cách giống như bạn làm việc với dữ liệu trong trang dữ liệu chính.

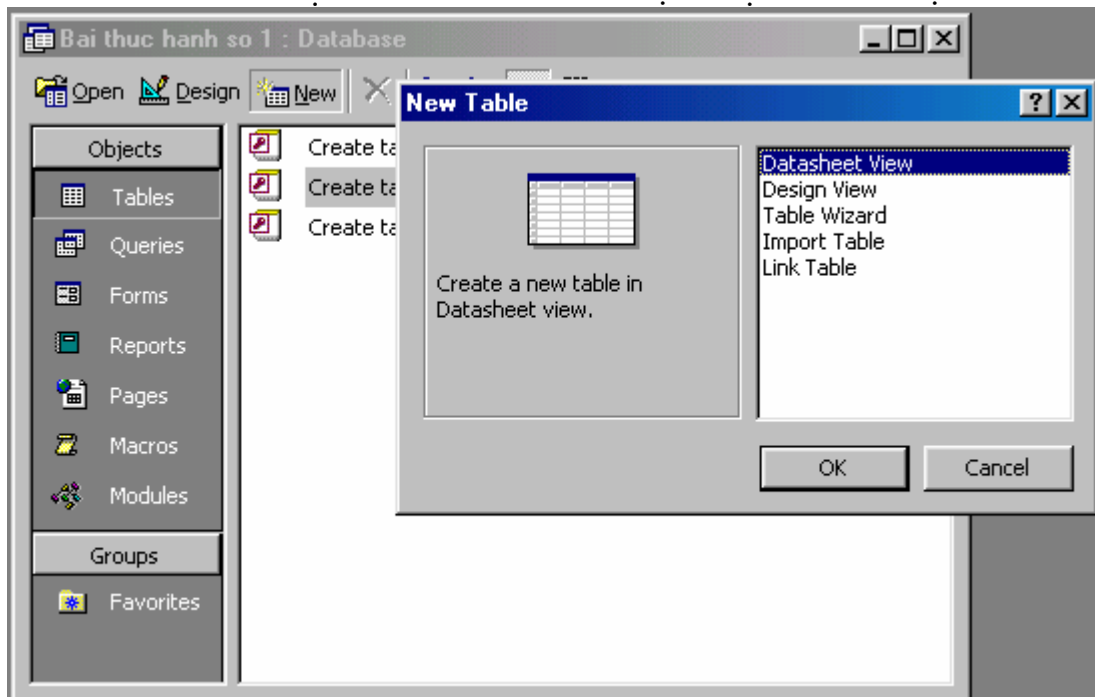
**II. Tạo Tables**

Trong Project (dự án) hoặc Database (cơ sở dữ liệu), hãy chọn **gõ table**. Chọn **New** để tạo mới một **Table**, sẽ cho ta hộp thoại **New Table** sau:

Trong đó:

- **Datasheet view**: tạo table mới với tên trường là **Field1, field2...** kiểu dữ liệu (**Data Type**) mặc định là dạng **text**.
- **Design view**: tạo mới **Table** với **Data Type** do ta tự thiết kế.
- **Table wizard**: tạo mới **Table** với **Data Type** theo mẫu hướng dẫn.

- Import table: tạo mới Table với dữ liệu lấy từ ngoài vào
- Link table: tạo mới Table với dữ liệu được link từ một file khác.

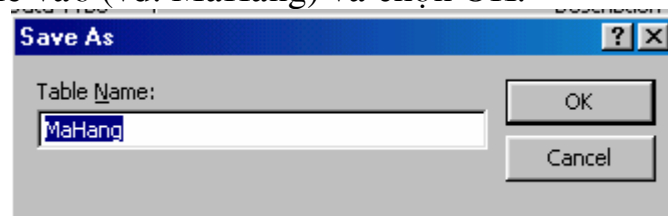


### Chú ý:

Khi tạo Table xong phải chọn khóa, nếu không sẽ hỏi “There is on Primary Key (khóa Index dùng để thiết lập quan hệ say này)...”

Khi ta chọn Save (hoặc Ctrl\_S) để lưu Table, Access sẽ cho ra hộp thoại đặt tên cho Table

Ta đánh tên Table vào (vd: MaHang) và chọn OK.



*Chú ý: Các kiểu định dạng cho trường Field được ấn định như sau*

- 0 : cho các số nguyên không âm
- 9 : cho các số nguyên có âm
- # : cho các số nguyên âm
- ? : cho các kí tự A->Z
- a : cho các kí tự A->Z, 0->9
- & : cho các ký tự bất kỳ bắt buộc
- c : cho các ký tự bất kỳ không bắt buộc
- > : Upper case
- < : Lower case
- ! : điền từ trái qua phải

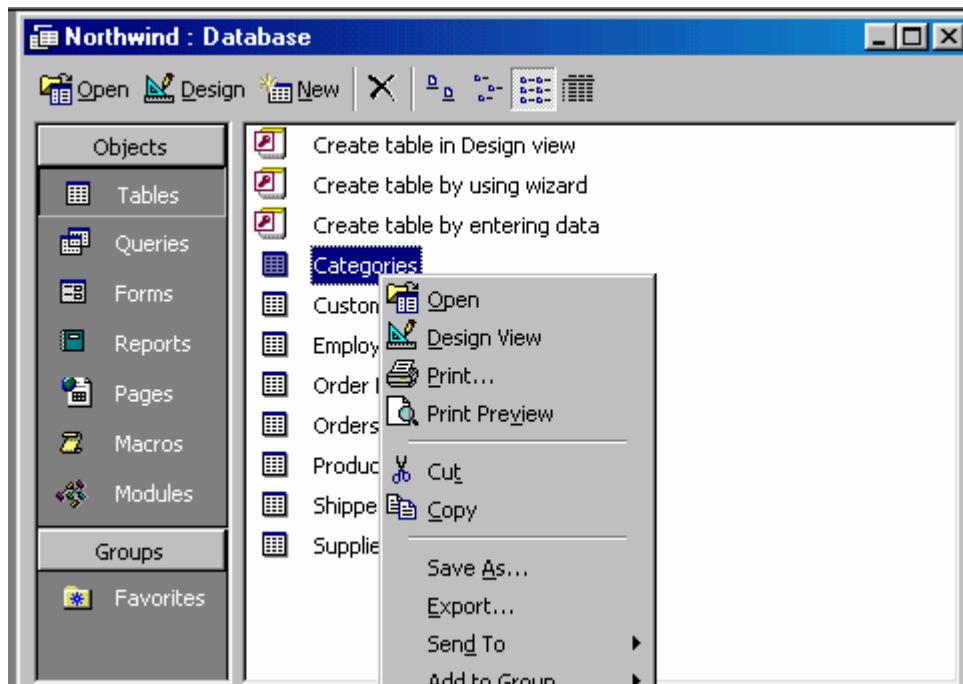
- \ : kí tự theo sau nó là một hằng (vd: \a có nghĩa a là một hằng số)
  - Nếu muốn nhập dữ liệu vào table, chọn table ấn chuột phải và chọn Open.
  - Nếu muốn sửa đổi cấu trúc table, ấn chuột phải trên table và chọn Design table

### III. Tùy chọn Tables

- ❖ Trong **Table Design View**, bạn có thể tùy nghi sử dụng một **Table** theo nhiều cách.
- ❖ Trong **Table Design View**, bạn có thể bổ sung hoặc xóa bỏ các trường, hoặc tùy nghi sử dụng các trường bằng cách cài đặt các tính chất.
- ❖ Nếu bạn muốn theo dõi dữ liệu được bổ sung trong một **Table**, hãy thêm vào nhiều trường. Nếu một tên trường đang có không mô tả đủ, bạn có thể đổi lại tên trường đó.
- ❖ Việc cài đặt kiểu dữ liệu của trường xác định loại giá trị nào bạn có thể đưa vào trong một trường.
- ❖ **Ví dụ:** bạn không thể nhập **Text** vào một trường **Currency** (trường tiền tệ). Bằng cách chọn một kiểu dữ liệu phù hợp, bạn có thể đảm bảo rằng dữ liệu được nhập vào ở dạng đúng để phân loại, thực hiện các phép tính, và thao tác khác.
- ❖ Các tính chất trường là một tập hợp các đặc trưng để cung cấp phần điều khiển bổ sung trên một trường đang hoạt động.
- ❖ Ví dụ, phụ thuộc vào **Regional Settings** trong Windows **Control Panel**, hãy cài đặt tính chất **Format** sang **Currency** dành cho một trường với kiểu dữ liệu **Number** or **Currency** tự động bổ sung dấu phẩy, dấu Đô la và hai chữ số thập phân (\$1,234.50).
- ❖ Bạn sử dụng một nhãn độc lập (Unique Tag), gọi là khóa chính (**Primary key**) để xác định mỗi một **Record** trong **Table**. Y như một bảng số xe, **Primary Key** chỉ định cho một **Record**.

### IV. Mở một Table

- ❖ Trong cửa sổ **Database**, nhấp **Table** trong mục **Objects**.
- ❖ Nhấp tên của **Table** mà bạn muốn mở.
- ❖ Để mở **Table** trong **Design View**, hãy nhấp **Design** trên thanh công cụ cửa sổ **Database**.
- ❖ Để mở **Table** trong **Data sheet View**, hãy nhấp **Open** trên thanh công cụ cửa sổ **Database**.



### Lưu ý:

- ❖ Sau khi bạn đã mở một **Table**, bạn có thể dễ dàng hoán chuyển giữa hai khung xem bằng cách nhấp lên nút **View** trên thanh công cụ.

### V. Hoán chuyển giữa các khung xem của một Table

- ❖ Các **Table** có hai khung xem (**View**): **Design View** và **DataSheet View**.
- ❖ Bạn có thể sử dụng **Design View** để tạo và chỉnh sửa cấu trúc của một **Table**. Bạn sử dụng **DataSheet View** để xem, bổ sung, xóa bỏ, và chỉnh sửa dữ liệu trong một **Table**.
- ❖ Nhấp nút **View** trên thanh công cụ.
- ❖ Nút **View** là một nút ngắt chuyển. Lúc **Table** của bạn được hiển thị trong **DataSheet View**, chỉ có nút chuyển sang **Design View** là được hiển thị, và ngược lại.

Customer ID	Company Name	Contact Name	Contact Title	
ALFKI	Alfreds Futterkiste	Maria Anders	Sales Representative	Obere Str. 57
ANATR	Ana Trujillo Emparedados	Ana Trujillo	Owner	Avda. de la C
ANTON	Antonio Moreno Taquería	Antonio Moreno	Owner	Mataderos 2
AROUT	Around the Horn	Thomas Hardy	Sales Representative	120 Hanover
BERGS	Berglunds snabbköp	Christina Berglund	Order Administrator	Berguvsvägen
BLAUS	Blauer See Delikatessen	Hanna Moos	Sales Representative	Forsterstr. 57
BLONP	Blondel père et fils	Frédérique Citeaux	Marketing Manager	24, place Klé
BOLID	Bólido Comidas preparadas	Martín Sommer	Owner	C/ Araquil, 67
BONAP	Bon app'	Laurence Lebihan	Owner	12, rue des E
BOTTM	Bottom-Dollar Markets	Elizabeth Lincoln	Accounting Manager	23 Tsawasse
BSBEV	B's Beverages	Victoria Ashworth	Sales Representative	Fauntleroy C
CACTU	Cactus Comidas para llevar	Patricio Simpson	Sales Agent	Cerrito 333
CENTC	Centro comercial Moctezuma	Francisco Chang	Marketing Manager	Sierras de G

## VI. Tối ưu hóa cách thực hiện chức năng của Table nói chung

- ❖ Có nhiều nội dung mà bạn phải thực hiện để tối ưu hóa **Table** của mình. Bên cạnh việc tuân theo các tip bạn có thể sử dụng **PerFormance Analyzer** để phân tích các **Table** đặc biệt trong **cơ sở dữ liệu** đặc biệt của mình.

1. Thiết kế các **Table** mà không có dữ liệu thừa: Một **cơ sở dữ liệu** được thiết kế tốt là một **cơ sở dữ liệu** có khả năng truy xuất và cập nhật nhanh dữ liệu. Nếu **Table** đang có dữ liệu thừa, bạn có thể sử dụng **Table Analyzer Wizard** để tách các **Table** thành các **Table** quan hệ nhằm lưu giữ dữ liệu hiệu quả hơn.

2. Chọn các kiểu dữ liệu phù hợp cho các trường: Bạn có thể lưu khoảng trống trong **cơ sở dữ liệu** và cải tiến các thao tác liên kết bằng cách chọn kiểu dữ liệu phù hợp cho các trường. Lúc định nghĩa một trường, hãy chọn kiểu dữ liệu nhỏ nhất hoặc kích cỡ trường phù hợp với dữ liệu trong trường đó.

3. Tạo các Index cho các trường mà bạn phân loại, nối, hoặc cài đặt tiêu chuẩn dữ liệu: Bạn có thể tạo ra các bước cải tiến về tốc độ vấn an bằng cách chỉnh sửa các trường trên cả hai phía cửa Join, hoặc bằng cách tạo ra một quan hệ giữa các trường này và tạo chỉ mục cho bất cứ trường nào đã được dùng để cài đặt tiêu chuẩn dành cho **Query**. Tìm các **Record**



thông qua hộp thoại **Find** thì nhanh hơn nhiều so với một trường được tạo chỉ mục.

Các chỉ mục (Index) không phải là luôn thích hợp cho tất cả các trường. Các chỉ mục bổ sung vào kích thước của **File \*.mdb**, làm giảm thiểu khả năng đồng xử lý (tức là khả năng có nhiều người dùng cùng chỉnh sửa một trang tại một thời điểm) trong các trình ứng dụng đang dùng, và làm giảm

khả năng hoạt động lúc bạn cập nhật các dữ liệu trong các trường đã được tạo Index, hoặc lúc bạn bổ sung hay xóa bỏ các **Record**. Tốt nhất là bạn nên thí nghiệm để xác định lại trường nào nên tạo Index. Việc bổ sung một Index có thể làm tăng tốc độ một **Query** lên một giây, nhưng làm chậm quá trình bổ sung mặt hàng dữ liệu khoảng hai giây và tạo ra các sự cố khóa nghiêm trọng. Hoặc nó có thể thu được lợi ích chung đáng kể, phụ thuộc vào các trường được tạo Index.

**Ví dụ**, việc bổ sung một Index vào một trang Postalcode có thể rất ít lợi ích về mặt thực hiện một trường CompanyName và trường LastName trong **Table** đã tạo Index rồi. Không kể đến kiểu **Query** mà bạn tạo ra, bạn chỉ nên tạo index cho các trường có các giá trị đồng nhất.

4. Trong một Index đa trường, chỉ sử dụng nhiều trường trong Index nếu thấy cần.

## VII. Xác định các quy tắc có hiệu lực để điều khiển lúc nào thì một Record có thể được lưu

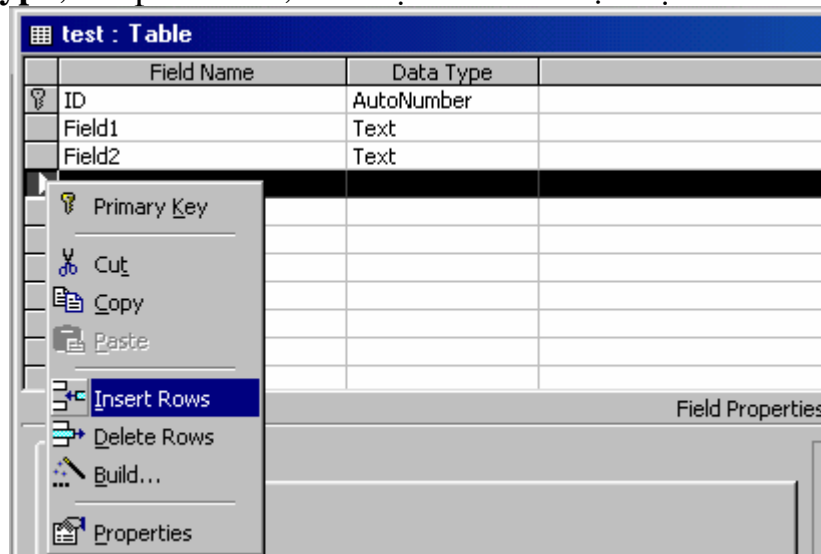
- ❖ Mở một **Table** trong **Design View**
- ❖ Nhấp **Properties** trên thanh công cụ để hiển thị trang tính chất của **Table**
- ❖ Trong hộp tính chất **ValidationRule**, hãy gõ nhập một quy tắc có hiệu lực bằng cách sử dụng **Expression Builder**.
- ❖ **Ví dụ**: bạn có thể xác định biểu thức đúng "[RequiredDate]<=[OrderDate]+30" để bảo đảm rằng ngày được nhập vào trong trường **RequiredDate** nằm trong vòng 30 ngày của trường **OrderDate**.
- ❖ Trong hộp tính chất **ValidationText**, hãy gõ nhập một thông tin mà bạn muốn Microsoft Access phải hiển thị lúc qui tắc bị vi phạm.

## VIII. Khảo sát Query của Table Analyzer Wizard

- ❖ **Table Analyzer Wizard** chia một **Table** có chứa thông tin trùng lặp thành các **Table** mà ở đó mỗi một kiểu thông tin được lưu giữ chỉ một lần. Điều này làm cho **cơ sở dữ liệu** hiệu quả hơn và dễ dàng được cập nhật hơn, đồng thời giảm thiểu kích thước của nó. Sau khi **Wizard** chia dữ liệu, bạn vẫn có thể xem và làm việc với dữ liệu ở một nơi bằng cách yêu cầu **Wizard** tạo ra một **Query**.
- ❖ Bạn có thể sử dụng **Query** để truy nhập dữ liệu từ nhiều **Table** cùng một lúc. **Query** cũng đưa ra các tính năng nhằm tiết kiệm thời gian để tăng độ chính xác của dữ liệu.
- ❖ Từ các **Table** mới, **Query** thừa kế các trường LookUp, nhằm giúp bạn thay đổi các giá trị trường bằng cách chọn các giá trị từ một danh sách thay vì gõ nhập đúng một giá trị.
- ❖ Các Form và **Report** trước đây dựa trên **Table** gốc giờ đây tự động đặt sơ sở trên **Query**, bởi vì **Wizard** gán tên **Table** gốc cơ sở vào **Query** và đặt lại tên **Table**. Đối với một trang truy cập dữ liệu dựa trên **Table** gốc, bạn phải thay đổi bằng tay cài đặt tính chất **RecordSource** của nó từ tên **Table** sang tên **Query**.

## IX. Bổ sung một trường vào một Table trong Design View.

- ❖ Mở **Table** trong **Design View**
- ❖ Để chèn trường bên trong **Table**, hãy nhấp hàng năm bên dưới nơi mà bạn muốn bổ sung trường, rồi nhấp **Insert Row** trên thanh công cụ.
- ❖ Để bổ sung trường vào phần cuối của **Table** hãy nhập hàng trống đầu tiên
- ❖ Trong cột **Data Type**, hãy giữ mặt định (**Text**), hoặc nhấp trong cột **Data Type**, nhấp mũi tên, rồi chọn kiểu dữ liệu bạn muốn.



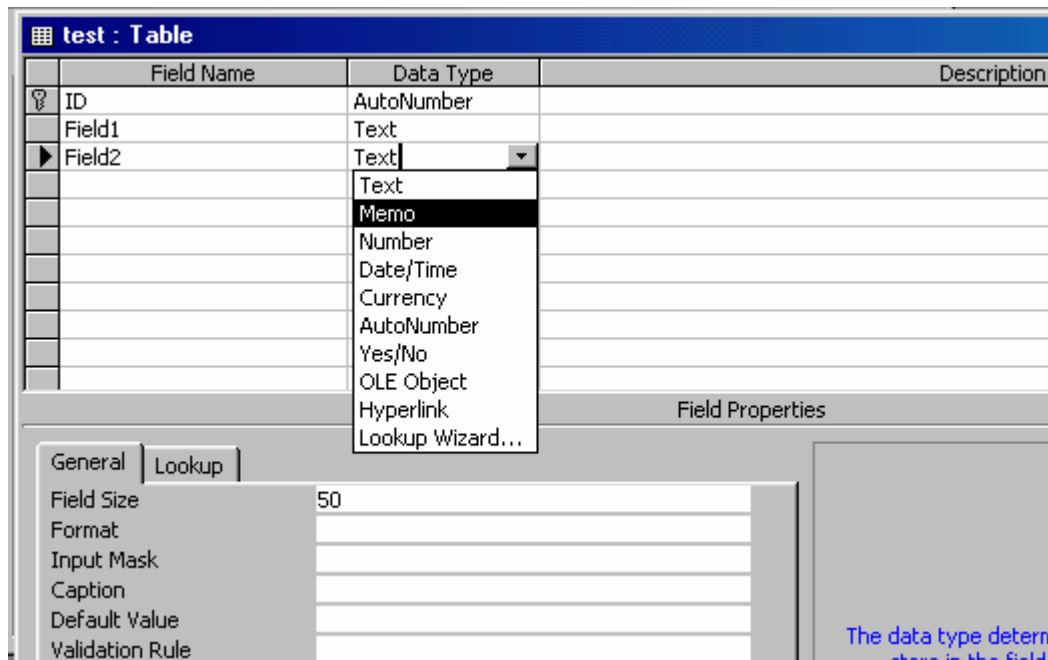
- ❖ Trong cột **Description**, hãy gõ nhập phần mô tả thông tin mà trường này có chứa. Phần mô tả này sẽ được hiển thị trên thanh trạng thái lúc bổ sung dữ liệu vào trường và được chứa trong **Object** definition của **Table**. Phần mô tả tùy ý.
- ❖ Nếu muốn bạn hãy cài đặt tính chất của trường dành cho trường nằm ở phần dưới cửa sổ.

## X. Sao chép một định nghĩa của trường trong Table Design View

- ❖ Mở **Table** trong **Design View**
- ❖ Chọn trường mà bạn muốn sao chép
- ❖ Để chọn một trường, hãy nhấp công cụ chọn hàng của trường đó (**Row Selector**).
- ❖ Để chọn một nhóm các trường, hãy rê qua qua các **Row Selector** của các trường đó.
- ❖ Nhấp **Copy** trên thanh công cụ rồi nhấp vào hàng trống thứ nhất.
- ❖ Nhấp **Paste** trên thanh công cụ.
- ❖ Gõ nhập tên mới dành cho trường được sao chép.

## XI. Tạo một trường dành cho Text hoặc Memo

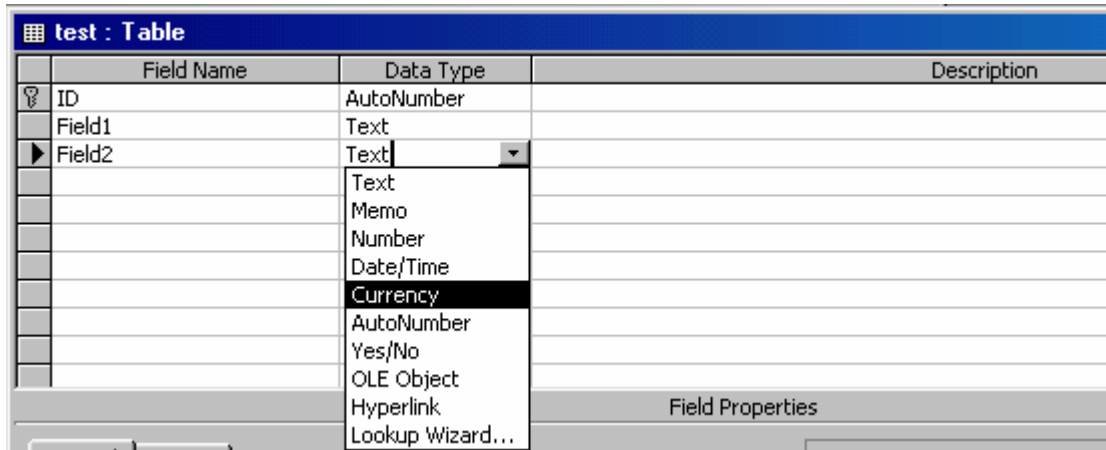
- ❖ Mở một **Table** trong **Design View**
- ❖ Để chèn trường bên trong **Table**, hãy nhấp hàng nằm bên dưới nơi mà bạn muốn bổ sung trường, rồi nhấp **Insert Row** trên thanh công cụ.
- ❖ Để bổ sung trường vào cuối **Table**, hãy nhấp dòng trống đầu tiên.
- ❖ Trong cột **Field Name**, hãy gõ nhập tên trường, tuân theo quy tắc đặt tên đối tượng của **Microsoft Access**.
- ❖ Trong cột **Data Type**, hãy nhấp lên mũi tên rồi chọn **Text** hoặc **Memo**.
- ❖ Cài đặt các tính chất trường khác nếu muốn.



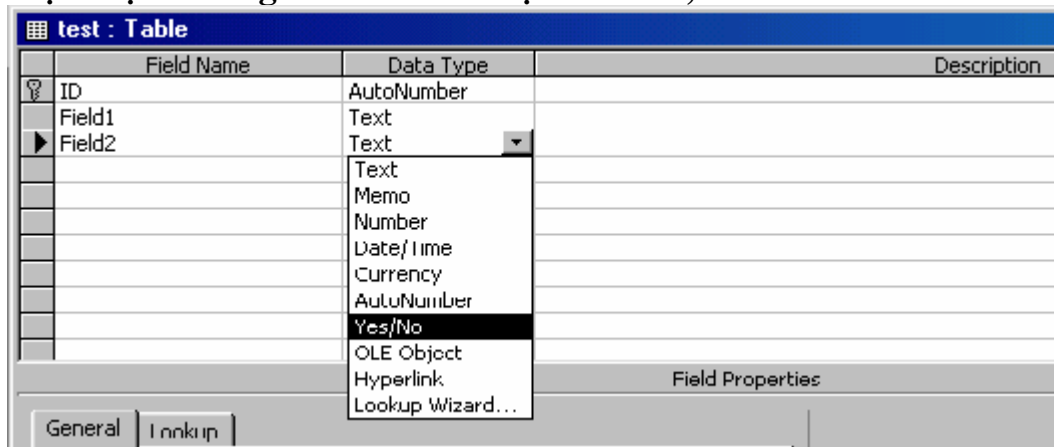
- ❖ **Lưu ý:** kích thước trường mặc định dành cho các trường **Text** là 50 kí tự. Bạn có thể thay đổi kích thước trường này cho một trường đặc biệt bằng cách cài đặt tính chất **Field Size**. Bạn có thể thay đổi chính kích thước trường mặc định bằng cách nhấp lên lệnh **Option** trên **Menu Tools**, rồi nhấp nhấn **Table/queries**.

## XII. Tạo một trường dành cho các số hoặc tiền tệ

- ❖ Mở một **Table** trong **Design View**.
- ❖ Để chèn trường bên trong một **Table**, hãy nhấp vào hàng nằm bên dưới nơi mà bạn muốn bổ sung trường, rồi nhấp **Insert Row** trên thanh công cụ.
- ❖ Để bổ sung trường vào phần cuối câu **Table** hãy nhấp vào hàng trống đầu tiên.
- ❖ Trong cột **Field Name**, hãy gõ nhập tên dành cho trường, tuân theo quy tắc đặt tên đối tượng của **Microsoft Access**.
- ❖ Trong cột **Data Type**, hãy nhấp mũi tên và chọn hoặc **Number** hoặc **Currency**.
- ❖ Nếu bạn đang tạo ra một trường **Number**, trong phần cuối của cửa sổ hãy cài đặt tính chất **Field Size** sang kích cỡ mà bạn muốn.
- ❖ Để cài đặt một dạng hiển thị cho trường của bạn ở phần đáy cửa sổ, hãy nhấp **Format**, nhấp mũi tên rồi chọn dạng mà bạn muốn. Bạn cũng có thể tạo một dạng hiển thị tùy ý, nếu dạng được định nghĩa trước không đáp ứng nhu cầu của mình.
- ❖ Cài đặt các tính chất trường khác nếu muốn.



### XIII. Tạo một trường dành cho dữ liệu Yes/No, True/false và On/Off



- ❖ Mở một **Table** trong **Design View**.
- ❖ Để chèn trường bên trong một **Table**, hãy nhấp **Table** bên dưới nơi mà bạn muốn bổ sung trường, rồi nhấp **Insert Rows** trên thanh công cụ.
- ❖ Để bổ sung trường vào phần cuối của **Table**, hãy nhấp vào hàng trống đầu tiên.
- ❖ Trong cột **Field Name**, hãy gõ nhập tên dành cho trường, tuân theo qui tắc đặt tên đối tượng của Microsoft **Access**.
- ❖ Trong cột **Data Type**, hãy nhấp mũi tên và chọn hoặc **Yes/No**.
- ❖ để hiển thị True/False hoặc On/Off thay vì Yes/No (theo mặc định) trong trường, ở phần dưới cửa sổ trên nhãn **General**, hãy nhấp vào hộp **Format** và chọn dạng mà bạn muốn.
- ❖ Cài đặt các tính chất trường khác, nếu bạn muốn.

### XIV. Tạo một trường dành cho ngày tháng hoặc thời gian

- ❖ *Mở một Table trong Design View.*
- ❖ Để chèn trường bên trong **Table**, hãy nhấp hàng bên dưới nơi mà bạn muốn bổ sung trường, rồi nhấp **InsertRows** trên thanh công cụ.

Field Name	Data Type	Description
ID	AutoNumber	
Field1	Text	
Field2	Text	
	Memo	
	Number	
	Date/Time	
	Currency	
	AutoNumber	
	Yes/No	
	OLE Object	
	Hyperlink	
	Lookup Wizard...	

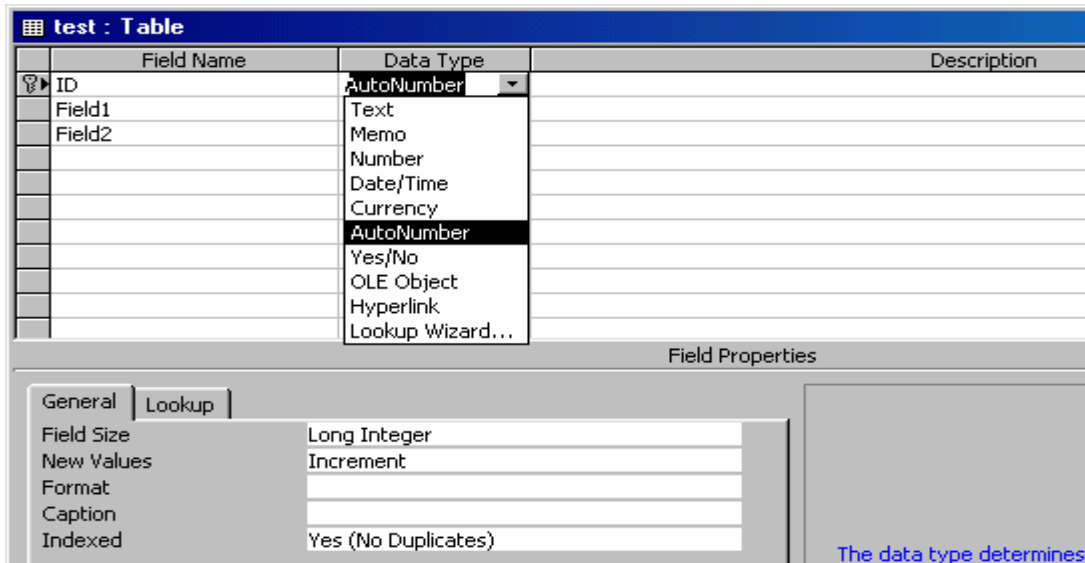
- ❖ Để bổ sung trường vào phần cuối của **Table**, hãy nhấp vào hàng trống đầu tiên.
- ❖ Trong cột **Field** Nam, hãy gõ nhập tên của trường, tuân theo qui tắc đặt tên đối tượng của Microsoft Access.
- ❖ Trong cột **Data Type**, hãy nhấp mũi tên và chọn **Date/Time**
- ❖ Để cài đặt một dạng hiển thị dành cho trường của bạn, hãy nhấp vào hộp **Format** rồi chọn dạng mà bạn muốn. Bạn cũng có thể tạo một dạng hiển thị tùy ý, nếu các dạng được xác định trước không đáp ứng nhu cầu của mình.
- ❖ Cài đặt các tính chất trường khác, nếu bạn muốn.

#### Lưu ý:

- ❖ Việc lưu giữ các ngày tháng và thời gian trong trường Date/Time bảo đảm được rằng ngày tháng và thời gian sẽ được phân loại hoàn chỉnh.
- ❖ Cũng vậy, các thay đổi được tạo ra cho các dạng ngày tháng và thời gian sẽ được xác định bằng cách nhấp đúp lên **Regional Settings** trong Windows **Control** Panel cũng sẽ được phản ảnh tự động trong các trường Date/time.

#### XV. Tạo một trường tự động để tạo ra các số

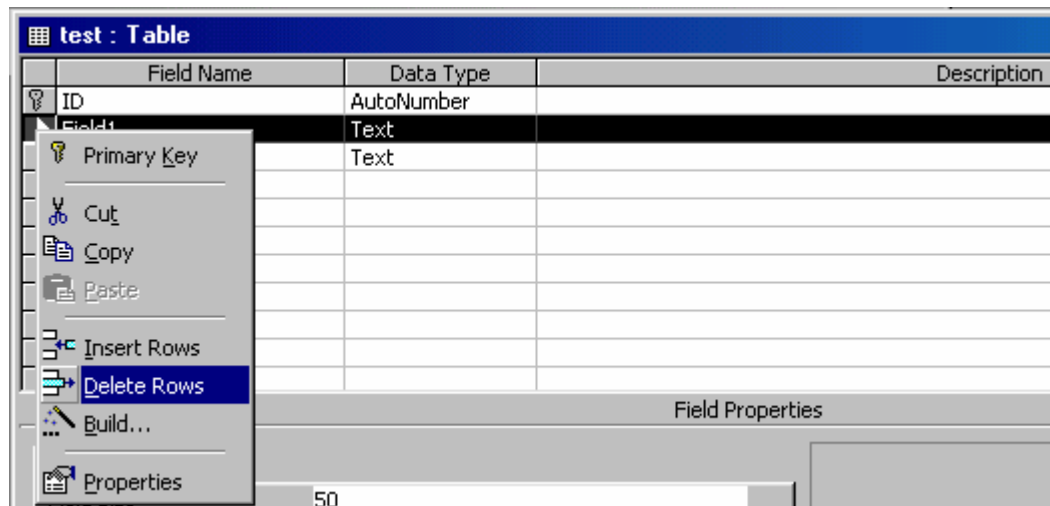
- ❖ Mở một **Table** trong **Design View**.
- ❖ Để chèn trường bên trong **Table**, hãy nhấp hàng bên dưới nơi mà bạn muốn bổ sung trường, rồi nhấp **Insert** Rows trên thanh công cụ.
- ❖ Để bổ sung trường vào phần cuối của **Table**, hãy nhấp vào hàng trống đầu tiên.
- ❖ Trong cột **Field** Nam, hãy gõ nhập tên của trường, tuân theo qui tắc đặt tên đối tượng của Microsoft Access.
- ❖ Trong cột **Data Type**, hãy nhấp mũi tên và chọn **Auto Number**



- ❖ Để tạo một giá số cho **AutoNumber**, hãy để lại các cài đặt tính chất ở phần cuối của cửa sổ y như chúng đang có (tính chất **Field Size** được cài đặt sang long Integer và tính chất new **Value** được cài đặt sang Increment (theo mặc định).
- ❖ Để tạo một **Auto Number** ngẫu nhiên, trong phần cuối của cửa sổ trong nhãn **General**, hãy cài đặt tính chất new **Values** sang random.
- ❖ Cài đặt các tính chất trường khác, nếu bạn muốn.

## XVI. Hủy bỏ một trường khỏi một Table

- ❖ Xóa bỏ một trường khỏi một **Table** trong **Design View**.
- ❖ Mở **Table** trong **Design View**
- ❖ Chọn trường mà bạn muốn xóa bỏ.
- ❖ Để chọn trường, hãy nhấp lên bộ chọn hàng của trường đó (**Row Selector**).
- ❖ Để chọn một nhóm các trường, hãy rê qua các bộ chọn hàng dành cho các trường này.
- ❖ Nhấp **Delete Row** trên thanh công cụ.
- ❖ Xóa bỏ một trường khỏi một **Table** trong **DataSheet View**.
- ❖ Mở một **Table** trong **Data View**
- ❖ Nhấp bộ chọn trường của cột mà bạn muốn xóa bỏ.
- ❖ Trên **Menu Edit**, nhấp **Delete Column**.



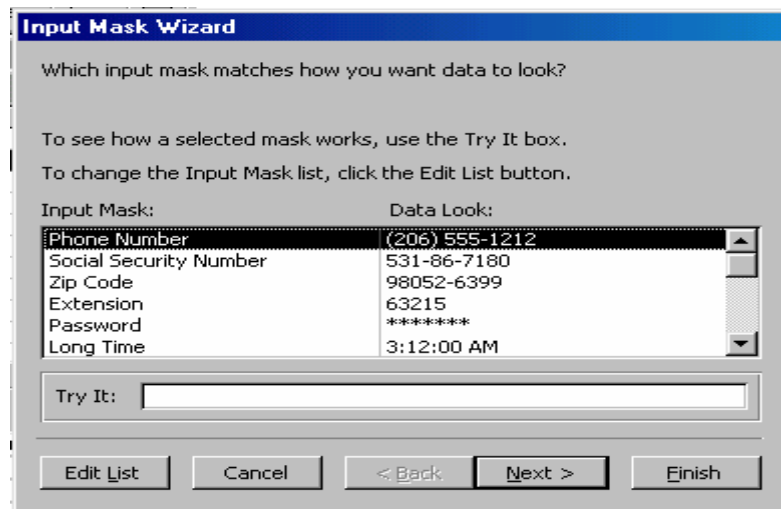
## XVII. Thay đổi kích thước của trường dành cho các trường Text và Number

- ❖ Sử dụng thủ tục này để điều khiển số tối đa các ký tự có thể được nhập vào một trường **Text** hoặc dãy các kiểu giá trị số có thể được nhập vào một trường **Number**.
- ❖ Trong **Table Design View**, ở phần trên cửa sổ, nhấp trường có tính chất **Field Size** bạn muốn cài đặt.
- ❖ Trong phần dưới của cửa sổ trên nhãn **General**, hãy nhấp chọn hộp tính chất **FieldSize**.
- ❖ Đối với một trường **Text**, hãy gõ nhập tối đa các ký tự cho phép trong trường lên đến 255.
- ❖ Đối với trường **Number**, hãy nhập mũi tên và chọn kích thước trường mà bạn muốn.

## XVIII. Tạo một Input Mask (mặt nạ nhập)

- ❖ Bạn có thể sử dụng tính chất **Inputmask** để điều khiển cách nhập dữ liệu trong một trường trong một **Table** hoặc **Query**, hoặc trong một mục điều khiển trên **Form** hoặc **Report**.
- ❖ Xác định một mặt nạ nhập (**Input Mask**) cho một trường trong một **Table**
- ❖ Mở một **Table** trong **Design View**.
- ❖ Ở phần trên của cửa sổ, hãy nhấp trường mà bạn muốn xác định một mặt nạ nhập.
- ❖ Trong phần dưới của cửa sổ trên nhãn **General**, hãy nhấp nút **Build** nằm kế hộp tính chất **Inputmask** để khởi động **Input Mask Wizard**, rồi tuân theo các chỉ dẫn trong các hộp thoại **Wizard**.





- ❖ **Lưu ý:** bạn cũng có thể nhập vào các định nghĩa mặt nạ nhập trong trang tính chất mà không cần sử dụng **Wizard**. Trong tất cả các trường hợp, cách
- ❖ dễ dàng nhất đó là sử dụng **Wizard**, nhưng đối với trường **Number** và **Currency**, thì bạn phải nhập vào định nghĩa mặt nạ nhập bằng tay.
- ❖ Định nghĩa mặt nạ nhập dành cho một trường trong một **Query**
- ❖ Mở một **Query** trong **Design View**.
- ❖ Trong **Query/Design Grid**, hãy đặt con trỏ chèn ở cột dành cho trường mà bạn muốn thay đổi. Bạn có thể đặt con trỏ chèn ở bất cứ hàng nào của trường đó.
- ❖ Nhấp **Properties** trên thanh công cụ để mở trang tính chất dành cho trường đó.
- ❖ Trên nhãn **General**, hãy nhấp nút build kế tính chất **Input Mask** để khởi động inputmask **Wizard**, rồi tuân theo chỉ dẫn trong các hộp thoại **Wizard**.

#### ***Tạo một mặt nạ nhập cho một mục điều khiển***

- ❖ Mở một **Form** trong **Design View**.
- ❖ Chọn một **Text** box hoặc combo box, rồi nhấp **Properties** trên thanh công cụ để mở trang tính chất của mục điều khiển.
- ❖ Thực hiện một trong các thao tác sau đây:
- ❖ Gõ nhập định nghĩa mặt nạ nhập.
- ❖ Trong một **cơ sở dữ liệu Microsoft Access (.bdb)**, hãy nhấp nút **Build** trong hộp tính chất **Inputmask** để khởi động **Input Mask Wizard**. Rồi tuân theo các chỉ dẫn trong các hộp thoại **Wizard**.

**Chương 4:**

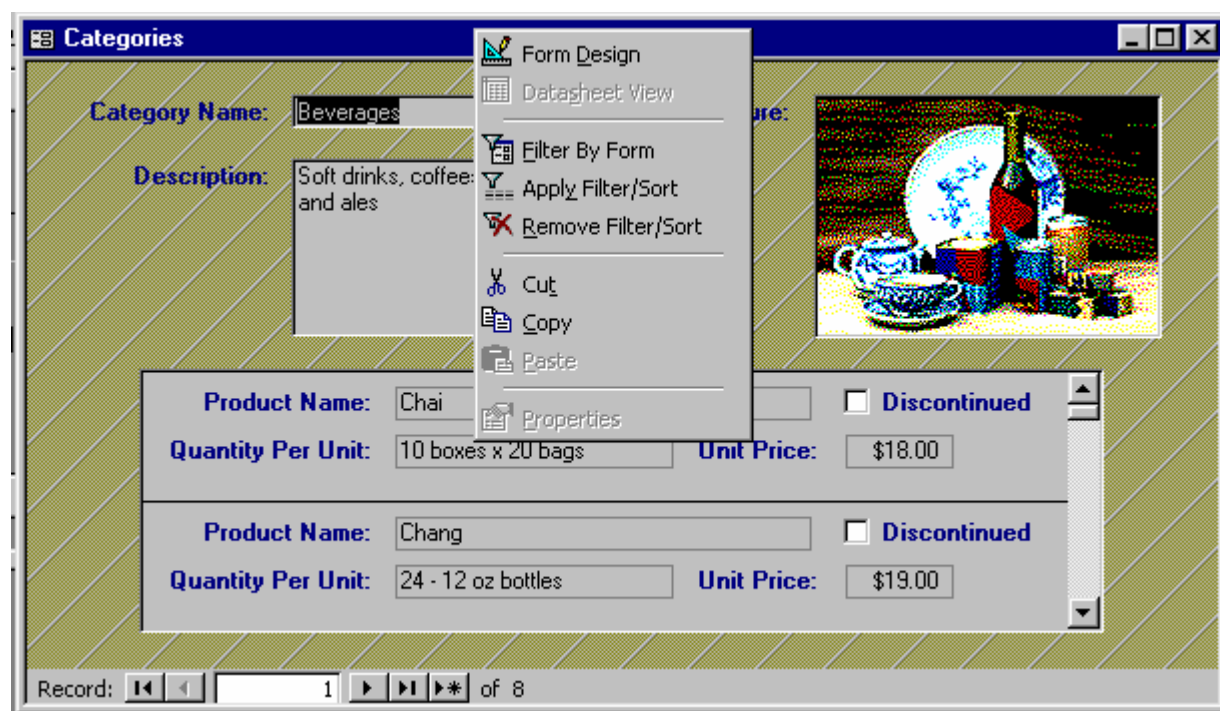
# Làm việc với Form (Query).

**I. Các cách để làm việc với dữ liệu trong một Form**

Khi làm việc với dữ liệu trong các form ta có thể hiểu là tạo Query hay vấn tin trên cơ sở dữ liệu.

**1. Các cách để làm việc với dữ liệu trong Form**

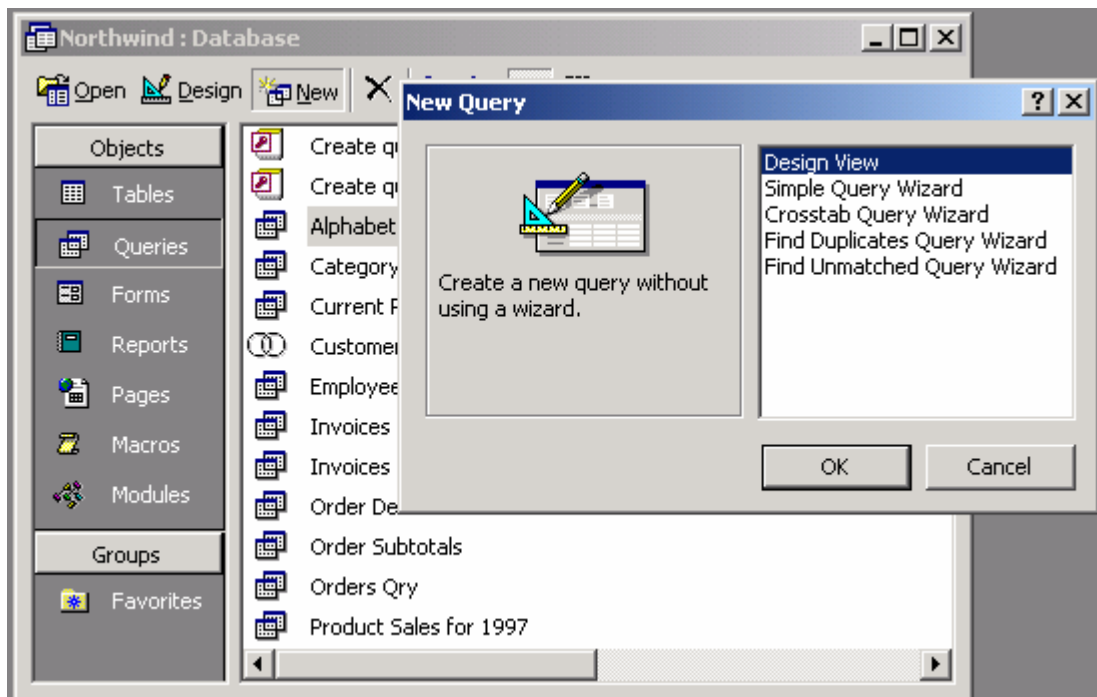
- Trong khung xem **FORM**, bạn có tất cả các công cụ cần thiết để làm việc với dữ liệu của mình.

**2. Các loại query:**

1. Select Query : Query chọn số liệu
2. Cross Tab Query : Query tham chiếu chéo
3. Make table Query : Query tạo ra Table
4. Update Query : Query cập nhật dữ liệu
5. Append Query : Query nhập nối dữ liệu
6. Delete Query : Query xóa dữ liệu
7. Pass-through Query : Query chuyển giao (dữ dụng ngôn ngữ vấn tin cơ sở dữ liệu)
8. Data defination Query: Query định nghĩa dữ liệu (sử dụng các lệnh của ngôn ngữ SQL để tạo, sử đổi cấu trúc Table)

**3. Tạo mới Query:**

Trong Database Project chọn gờ Query, tiếp tục chọn New.



a. Chọn Query Wizard: Access sẽ hướng dẫn, có 4 loại Query Wizard sau:

- Simple Query Wizards: tạo bảng tham chiếu đơn, giống như chép các record trong Table ra một Table khác để truy vấn dữ liệu.
- Crosstab Query Wizards: tạo bảng tham chiếu chéo số liệu.
- Find duplicates Query Wizards: tạo bảng Record trùng lặp trong Table
- Find unmatched Query Wizards: tìm các record trong table các record nào không có quan hệ với table khác.

b. Nếu chọn Design View: tạo New Query (người dùng tự thiết kế).

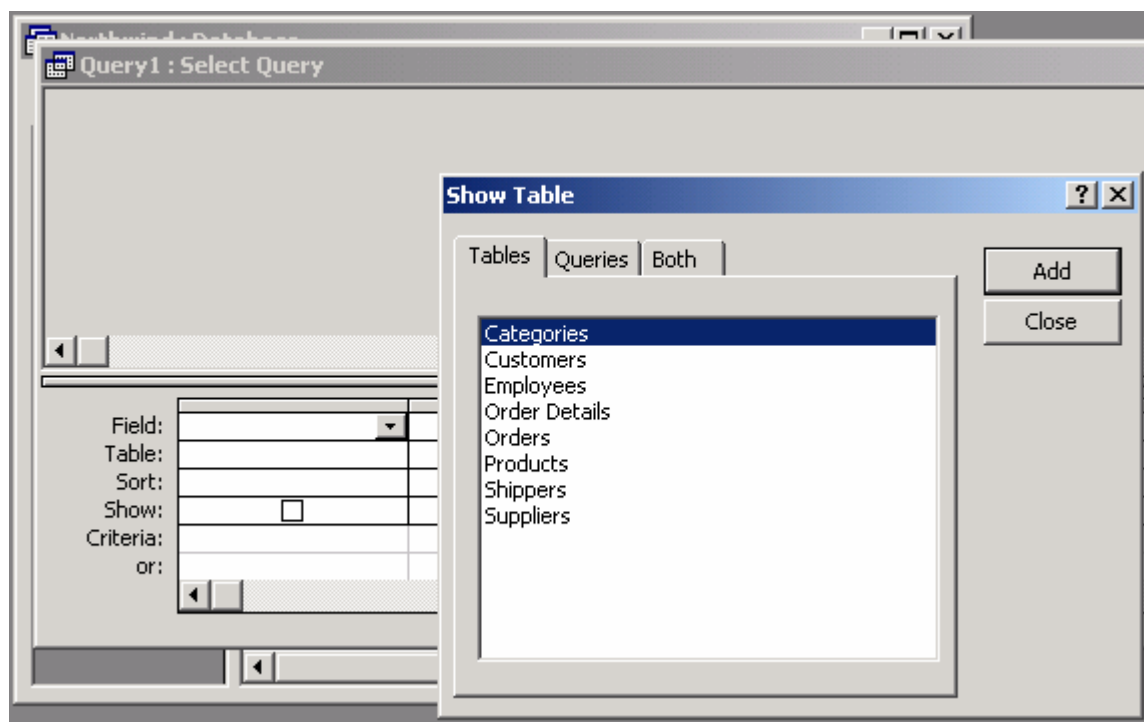
Trong khung Show Table:

Trong gờ Table: nhấp chọn Table cần tạo Query, nhấp Add để đưa Table vào vùng Select Query.

Nếu Query cần truy vấn đến Query khác nhấp chọn gờ Query. Chọn Query nhấp Add để đưa Table vào vùng Select Query.

Thiết lập quan hệ giữa các Table (hay Query) trong Query bằng cách chọn lựa các trường quan hệ từ một Table gắn vào trường của Table khác.

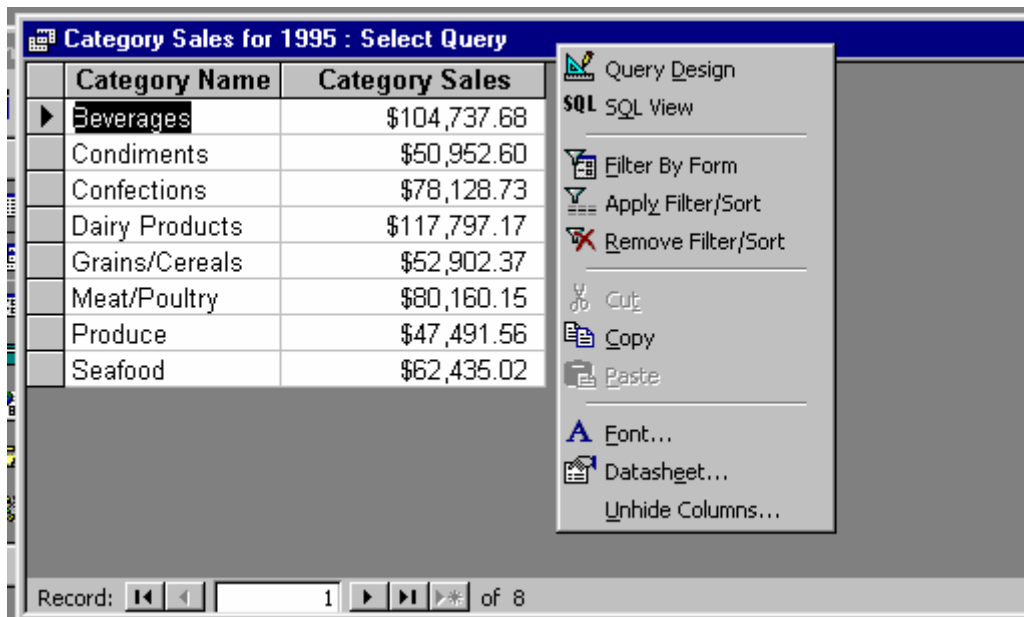
- Sort: sắp xếp tăng/giảm (Ascending/Descending)
- Criteria: Đặt điều kiện cho một trường (vd: 0>Date>32)
- Show: show/hide.



trong hộp thoại Query Wizards sẽ hỏi “  
nếu chọn Design View

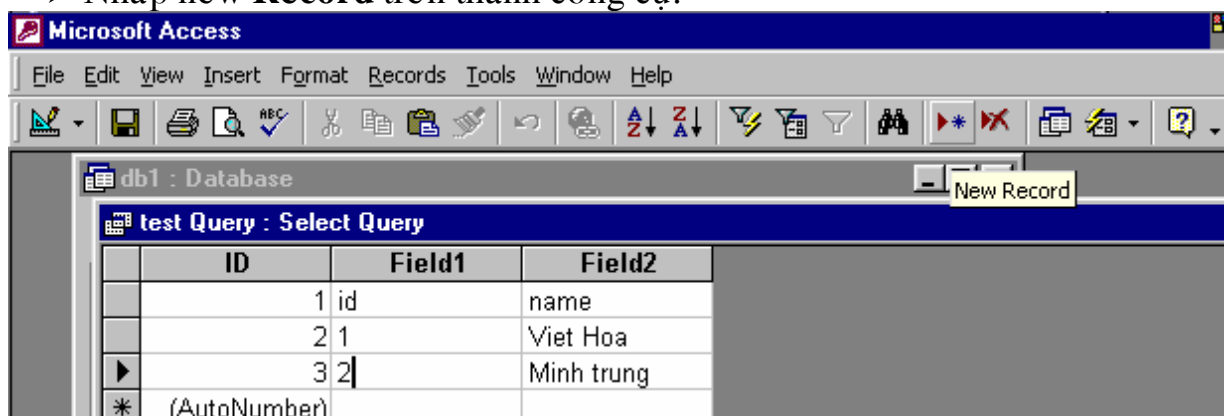
## II. Các cách để làm việc với dữ liệu trong DataSheet của Query

- Có nhiều cách để làm việc với dữ liệu trong **DataSheet** của **Query**.
- Trong lúc bạn đang xem xét các kết quả của **Query**, bạn có thể hiển thị dữ liệu liên quan đến các **Record** trong trang dữ liệu bằng cách chèn một trang dữ liệu con cho dữ liệu quan hệ. Mặc dầu có nhiều hạn chế, bạn vẫn có thể làm việc với dữ liệu trong một trang dữ liệu con theo nhiều cách giống như bạn làm việc với dữ liệu chính trong trang dữ liệu chính.
- **Ví dụ**, bạn có thể lọc và phân loại dữ liệu.



### III. Bổ sung dữ liệu mới vào DataSheet hoặc Form View

- Mở một **DataSheet**, hoặc **Form** trong **Form View**
- Nhấp **new Record** trên thanh công cụ.



- Gõ nhập dữ liệu mà bạn muốn, rồi nhấn **Tab** để chuyển sang trường kế tiếp.
- Ở cuối **Record**, hãy nhấn **Tab** để chuyển sang **Record** kế tiếp.

### IV. Lưu một Record trong DataSheet hoặc Form View

- **Microsoft Access** tự động lưu **Record** mà bạn đang bổ sung hoặc chỉnh sửa ngay khi bạn di chuyển điểm chèn đến **Record** khác, hoặc đóng **Form** hoặc trang dữ liệu mà bạn đang làm việc.
- Để lưu dữ liệu trong một record khi bạn đang chỉnh sửa nó, hãy nhấp **Save Record** trên **Menu Records**.

### V. Xóa bỏ một Record trong DataSheet hoặc Form View

- Mở một **Data sheet**, hoặc **Form** trong **Form View**.
- Nhấp **Record** mà bạn muốn xóa bỏ.
- Nhấp **Delete Record** trên thanh công cụ.



## VI. Sao chép hoặc di chuyển một trường, Control, Text, Macro Action.

- Bạn có thể sao chép hoặc xóa bỏ các hạng mục, chẳng hạn như trường, **Control**, **Text**, hoặc **Macro Action**. Bạn cũng có thể sao chép các đối tượng **cơ sở dữ liệu**.
- Chọn hạng mục mà bạn muốn sao chép hoặc di chuyển.
- Để sao chép hạng mục, hãy nhấp **Copy** trên thanh công cụ.
- Để di chuyển hạng mục, hãy nhấp **Cut** trên thanh công cụ.
- Nhấp vị trí nơi mà bạn muốn đặt hạng mục.
- Nhấp **Paste** trên thanh công cụ.

## VII. Hiển thị tất cả dữ liệu đang có trong Page View.

- Nếu người sử dụng khác đang nhập dữ liệu cùng lúc với bạn, thì trang truy cập dữ liệu có thể không hiển thị các thay đổi vừa mới tạo ra cho **cơ sở dữ liệu Microsoft Access** hoặc **Microsoft Access Project**. Bạn có thể tải dữ liệu mới nhất từ **cơ sở dữ liệu**.
- Nhấp phải lên trang đó, rồi nhấp refresh, hoặc nhấn F5.
- Cảnh báo: nếu bạn nhấp refresh trước khi bạn lưu các thay đổi, **Microsoft Access** cảnh báo bạn rằng các thay đổi có thể bị xáo trộn, và bạn phải nhấp cancel để nhấn việc thất thoát các thay đổi cho **Record** hiện có.

## VIII. Nhập một HyperLink (siêu phù hợp) bằng cách dán một HyperLink hoặc địa chỉ HyperLink

1. Định hướng đến một hyperlink hoặc địa chỉ **HyperLink** theo một trong các cách sau đây:

- Mở tài liệu **Microsoft Office** có chứa một **HyperLink** được xác định.
- Mở một tài liệu được tạo ra với bất cứ trình ứng dụng nào và chứa một địa chỉ **HyperLink** có hiệu lực không được xác định dưới dạng một **HyperLink**.
- Mở một bộ trình duyệt **Web** chẳng hạn như **Microsoft Internet Explorer**.
- Thực hiện mỗi trong các thao tác sau đây:
- Trong tài liệu **Microsoft Office**, hãy nhấp phải lên **HyperLink** được xác định.
- Trở đến **HyperLink** trên **Menu** tắt, rồi nhấp **Copy HyperLink**.
- Trong bất cứ tài liệu nào, hãy chọn địa chỉ **HyperLink**, rồi nhấp **Copy** trên **Menu Edit**.
- Trong *Internet Explorer*, hãy chọn địa chỉ trong hộp **Address** hoặc chọn một địa chỉ **HyperLink** có hiệu lực ở bất cứ nơi nào trên một trang, hãy nhấp phải, rồi nhấp **Copy** hoặc **Copy ShortCut**.

**Hoán chuyển đến Microsoft Access và thực hiện một trong các thao tác sau đây:**

- Trong cơ sở dữ liệu *Microsoft Access (.mdb)*, hãy mở **Table** hoặc **Query** trong **DataSheet View**.
- Trong một cơ sở dữ liệu *Microsoft Access* hoặc *Microsoft Access Project (.adp)*, hãy mở **Form** trong **DataSheet View** hoặc **Form View**.
- Đặt điểm chèn trong trường **HyperLink Field** trong một trang dữ liệu hoặc trong hộp **Text** hoặc mục **Control** khác chấp nhận **HyperLink** trên **Form** đó.
- Thực hiện một trong các thao tác sau đây:
- Để dán một **HyperLink** đã được xác định, hãy nhập **Paste as HyperLink** trên **Menu Edit**.
- Để dán một địa chỉ **HyperLink** không được xác định dưới dạng một **HyperLink**, hãy nhấp **Paste** trên **Menu Edit**. Lúc bạn dán một địa chỉ theo cách đó, thì **Microsoft Access** nhận biết protocol trong địa chỉ đó và chuyển **Text** sang một **HyperLink**.
- **Microsoft Access** bổ sung **HyperLink** đến trường hoặc **Control**. Bạn có thể nhấp **HyperLink** để mở tài liệu đích hoặc trang **Web**.

**Chương 5:**

# Làm việc với Record.

**I. Bổ sung một Record mới trong Page View**

1. Trên thanh công cụ định hướng Record, hãy nhấp new Record.
2. Gõ nhập dữ liệu mà bạn muốn trong trường thứ nhất, rồi nhấn **Tab** để đi đến trường kế tiếp.
3. Lúc bạn đã hoàn tất việc nhập dữ liệu cho mỗi trường, hãy nhấp **Save Record** để lưu dữ liệu **Record**.

**II. Lưu một Record trong Page View**

- Trên thanh công cụ định hướng **Record**, hãy nhấp **Save Record** để lưu dữ liệu trong một **Record** trong khi bạn đang chỉnh sửa nó.

**Lưu ý:**

- **Record** được tự động lưu ngay khi bạn di chuyển đến một **Record** khác, hãy áp dụng hoặc xóa bỏ một bộ lọc, hoặc là hoán chuyển sang một
- trang **Web** khác.

**III. Xóa bỏ một Record trong Page View**

- Lưu ý quan trọng: một khi bạn xóa bỏ một **Record**, bạn không thể phục hồi mục xóa bỏ từ **cơ sở dữ liệu** được.
- Di chuyển đến **Record** hoặc sử dụng một bộ lọc để tìm **Record** mà bạn muốn xóa bỏ.
- Trên thanh công cụ định hướng **Record**, hãy nhấp **Delete Record**.

**IV. Sao chép hoặc di chuyển dữ liệu từ một trường sang một trường khác**

- Mở một **DataSheet** hoặc **Form** trong **Form View**
- Chọn dữ liệu mà bạn muốn sao chép hoặc cắt.
- Để sao chép dữ liệu hãy nhấp **Copy** trên thanh công cụ.
- Để di chuyển dữ liệu hãy nhấp **Cut** trên thanh công cụ.
- Để thay đổi giá trị đang có của trường đích, hãy chọn toàn bộ trường.
- Để chèn dữ liệu vào dữ liệu đang có, hãy đặt điểm chèn tại nơi mà bạn muốn dán dữ liệu.
- Nhấp **Paste** trên thanh công cụ.

**V. Sao chép hoặc di chuyển dữ liệu từ nhiều trường vào DataSheet View**

- Mở một **DataSheet** hoặc **Form** trong **Form View**



- Chọn các trường hoặc cột mà bạn muốn sao chép hoặc cắt.
- Để sao chép dữ liệu, hãy nhấp **Copy** trên thanh công cụ.
- Để di chuyển dữ liệu hãy nhấp **Cut** trên thanh công cụ.
- Hãy chọn các trường đích, hoặc hoán chuyển sang trình ứng dụng mà bạn muốn dán dữ liệu vào đó, rồi chọn trường đích.
- Nhấp **Paste** trên thanh công cụ.

## VI. Sao chép hoặc di chuyển các Record vào một trang dữ liệu

- Mở một **DataSheet** hoặc **Form** trong **Form View**
- Chọn **Record** hoặc các **Record** mà bạn muốn sao chép hoặc di chuyển.

### Lưu ý:

Nếu bạn muốn sao chép hoặc di chuyển các **Record** từ một trình ứng dụng khác, trước khi bạn chọn nó, bạn phải chắc chắn dữ liệu được sắp xếp trong một trang bảng tính hoặc **Table**, hoặc được tách nhau bởi các kí tự **Tab**.

- Để sao chép **Record**, hãy nhấp **Copy** trên thanh công cụ.
- Để di chuyển dữ liệu hãy nhấp **Cut** trên thanh công cụ.
- Mở **DataSheet** mà bạn muốn dán **Record** vào đó. Nếu cần hãy sắp xếp lại các cột của **DataSheet** để nó phù hợp với thứ tự của dữ liệu mà bạn đang sao chép hoặc di chuyển.
- Để thay thế các **Record** bằng các **Record** mà bạn đã chọn, hãy chọn các **Record** mà bạn muốn thay thế, rồi nhấp **Paste** trên thanh công cụ.
- Để đính các **Record** vào cuối trong bảng tính, hãy nhấp **Paste Append** trên **Menu Edit**.
- Nếu trang bảng tính nguồn có nhiều trường hơn là trang bảng tính đích, thì **Microsoft Access** sẽ không dán các trường dư.
- Để di chuyển dữ liệu hãy nhấp **Cut** trên thanh công cụ.

## VIII. Sao chép hoặc di chuyển các Record sang một Form

- Mở một **DataSheet** hoặc **Form** trong **Form View**
- Chọn **Record** hoặc các **Record** mà bạn muốn sao chép hoặc di chuyển.
- Để sao chép **Record** hãy nhấp **Copy** trên thanh công cụ.
- Để di chuyển dữ liệu hãy nhấp **Cut** trên thanh công cụ.
- Mở **Form** mà bạn muốn dán **Record** vào đó.
- Để thay đổi các **Record** bằng các **Record** mà bạn đã chọn, hãy chọn các **Record** mà bạn thay thế, rồi nhấp **Paste** trên thanh công cụ.
- Để đính các **Record**, hãy nhấp **Paste Append** trên **Menu Edit**.

- **Microsoft Access** dán các **Record** vào trường có tên giống như các trường nguồn, không kể đến thứ tự của chúng trên **Form**. Nếu **Form** mà từ đó bạn đang sao chép hoặc di chuyển các **Record** lại có các trường không có trong **Form** mà bạn đang di Chuyển hoặc sao chép đến, thì **Microsoft Access** sẽ hỏi bạn có muốn chỉ dán các trường với các tên phù hợp hay không. Nếu bạn không có các tên trường phù hợp, thì các trường được dán theo thứ tự **Tab** trong **Form** đích, có thể nó không theo thứ tự như ý bạn muốn. Vì lý do đó, tốt nhất là bạn phải dán các trường với các tên nguồn và đích khác nhau vào một trang dữ liệu thay vì một **Form**.

### **IX. Sao chép hoặc di chuyển các Record từ Microsoft Access đến trình ứng dụng khác**

- Mở một **DataSheet** hoặc **Form** trong **Form View**.
- Chọn dữ liệu mà bạn muốn sao chép hoặc di chuyển.
- Để sao chép dữ liệu, hãy nhấp **Copy** trên thanh công cụ
- Để di chuyển dữ liệu, hãy nhấp **Cut** trên thanh công cụ.
- Mở trình ứng dụng mà bạn muốn dán dữ liệu.
- Để thay thế dữ liệu bằng dữ liệu từ **Microsoft Access**, hãy chọn dữ liệu mà bạn muốn thay thế.
- Để chèn dữ liệu vào dữ liệu đang có, hãy định vị trí điểm chèn ở nơi mà bạn muốn dán dữ liệu.

#### **Lưu ý:**

- Nếu bạn đang dán vào một trang bảng tính Microsoft Excel, hãy chọn Ô mà bạn muốn làm tiêu đề cột đầu tiên.
- Nếu bạn đang dán vào một tài liệu Microsoft Word for Windows, hãy di chuyển điểm chèn đến nơi mà bạn đặt **Record**. **Record** được dán dưới dạng một **Table** trong Word.
- Nhấp **Paste** trên **Menu Edit**. Nếu trình ứng dụng của bạn không có lệnh này hãy sử dụng phương pháp so sánh cho trình ứng dụng của bạn.

#### **Lưu ý 1:**

- Lúc bạn dán các **Record** vào trình ứng dụng khác, Microsoft **Microsoft** dán các tên trường vào hàng đầu tiên theo sau đó là dữ liệu.
- Nếu bạn đang sao chép từ một trang bảng tính có chứa các trang bảng tính
- con, **Microsoft Access** chỉ có thể sao chép một cấp trang bảng tính một lần.

## X. Sao chép hoặc di chuyển các Record từ một trình ứng dụng khác vào Microsoft Access

- Trong trình ứng dụng khác, hãy chọn một hoặc nhiều hàng mà bạn muốn sao chép hoặc di chuyển. Nếu bạn đang sao chép từ một trình xử lý văn bản, thì phải chắc chắn các **Record** được sắp xếp trong một **Table** hoặc được tách nhau bởi các kí tự **Tab** trước khi chọn chúng.
- Nếu bạn đang dán vào một trang bảng tính, phải chắc chắn rằng các cột phù hợp với thứ tự của dữ liệu mà bạn muốn sao chép hoặc di chuyển.
- Nếu bạn đang dán vào một **Form**, và các **Control** tương ứng trên **Form** có cùng tên giống như các tên cột của dữ liệu mà bạn đang sao chép, **Microsoft Access** sẽ dán dữ liệu sang các **Control** phù hợp. Nếu các tên cột không phù hợp, hoặc không có tên cột, thì **Microsoft Access** dán dữ liệu theo thứ tự **Tab (Tab Order)** của **Form**.
- Để sao chép **Record**, hãy nhấp **Copy** trên **Menu Edit**.
- Để di chuyển các **Record**, hãy nhấp **Cut** trên **Menu Edit**

### Lưu ý

- Nếu trình ứng dụng của bạn không có các lệnh này, hãy sử dụng phương pháp so sánh cho trình ứng dụng.
- Trong **Microsoft Access**, hãy mở trang bảng tính hoặc **Form** mà bạn muốn dán các **Record**.

### *Thực hiện một trong các bước sau đây:*

- Để thay thế các **Record**, hãy chọn các **Record** này, rồi nhấp **Paste** trên thanh công cụ. Trong một **Form**, nếu **Record Selector** không hiển thị, hãy chọn **Record** hiện có, nhập **Select Record** trên **Menu Edit**.
- Để đính dữ liệu làm các **Record** mới, hãy nhấp **Paste Append** trên **Menu Edit**.

## XI. Thu thập và dán nhiều hạng mục

- Chọn hạng mục đầu tiên mà bạn muốn sao chép
- Trên thanh công cụ **Clipboard** hãy nhấp **Copy**.
- Để hiển thị thanh công cụ **Clipboard**, hãy trở đến Toolbar trên **Menu View** rồi nhấp **Clipboard**.
- Nếu hạng mục mà bạn muốn sao chép nằm trong một chương trình khác, hãy chuyển sang chương trình đó.
- Chọn hạng mục kế tiếp mà bạn muốn sao chép.
- Trên thanh công cụ **Clipboard**, hãy nhấp **Copy**.

- Nếu thanh công cụ **Clipboard** không có sẵn, hãy nhấp **Copy** trên men **Edit**.
- Lặp lại các bước từ 3 đến 5 cho đến khi bạn sao chép tất cả các hạng mục: (lên đến 12) mà bạn muốn sao chép.
- Nhấp nơi mà bạn muốn các hạng mục phải được dán.
- Để dán tất cả các hạng mục mà bạn đã sao chép, hãy nhấp **Paste All** trên thanh công cụ **Clipboard**.
- Nếu bạn không muốn dán tất cả các hạng mục đã sao chép, hoặc nếu **Paste All** không có sẵn, bạn có thể dán các hạng mục đặc biệt.

## **XII. Khảo sát về việc thu thập và dán nhiều hạng mục Sao chép nhiều hạng mục**

- Bạn có thể sử dụng Office **Clipboard** để thu thập và dán nhiều hạng mục. **Ví dụ**, bạn có thể sao chép **Text** trong một tài liệu **Microsoft Word**, hoán Chuyển sang Microsoft Excel và sao chép một đối tượng vẽ, hoán chuyển sang **Microsoft Powerpoint** và sao chép một danh sách được đánh dấu đầu
- dòng, hoán chuyển sang **Microsoft Access** và sao chép một trang dữ liệu, rồi hoán chuyển sang **Microsoft Word** và dán tập hợp các hạng mục được sao chép. Office **Clipboard** tự động sao chép nhiều hạng mục lúc bạn thực hiện bất cứ các thao tác sau đây:
- Sao chép hoặc cắt hai hạng mục khác nhau trong chương trình giống nhau.
- Sao chép một hạng mục, dán một hạng mục, rồi sao chép hạng mục khác trong cùng chương trình.
- Sao chép một hạng mục hai lần.
- Sử dụng thanh công cụ **Clipboard**
- Bạn cũng có thể sao chép nhiều hạng mục bằng cách sử dụng thanh công cụ **Clipboard** (**Menu View**, **Menu con Toolbars**) và nhấp **Copy**. Nếu bạn mở Office **Clipboard** trong một chương trình **Microsoft Office**, Office **Clipboard** xuất hiện lúc bạn hoán chuyển sang chương trình Office khác. Thanh công cụ **Clipboard** không có sẵn trong các khung xem mà ở đó các lệnh **Copy**, **Cut**, và **Paste** không có sẵn.
- Lúc thanh công cụ **Clipboard** được hiển thị, bạn có thể nhấp **Paste All** để dán tất cả các hạng mục mà bạn đã sao chép, hoặc bạn có thể nhấp một hạng mục riêng biệt để chỉ dán hạng mục đó. Lệnh **Paste** (**Menu Edit**) chỉ dán hạng mục sau cùng mà bạn đã sao chép. Tuy nhiên, sau

khi bạn đã dán bất cứ nội dung nào từ thanh công cụ **Clipboard**, hoặc một hạng mục riêng biệt hoặc tất cả các hạng mục, thì lệnh **Paste (Menu Edit)** dán nội dung sau cùng đã được dán từ thanh công cụ **Clipboard**.

- Office **Clipboard** và Windows **Clipboard**
- Office **Clipboard** phân biệt với Windows **Clipboard**. Lúc bạn sao chép nhiều hạng mục vào Office **Clipboard**, hạng mục sau cùng mà bạn sao chép thì luôn luôn được sao chép vào Windows **Clipboard**. Lúc bạn xóa Office **Clipboard**, Windows **Clipboard** cũng bị xóa. Lúc bạn sử dụng lệnh **Paste (men Edit)**, nút **Paste**, hoặc phím tắt **Paste (CTRL+V)** thì bạn dán nội dung của Windows **Clipboard**.
- Dung lượng của office **Clipboard** .
- Office **Clipboard** có thể chứa đến 12 hạng mục.
- Nếu bạn sao chép một hạng mục riêng biệt lúc Office **Clipboard** đã có chứa 12 hạng mục rồi, bạn nhận được một thông báo hỏi bạn rằng hạng mục đầu tiên có bị xóa đi hay Không. Nếu không, thì hạng mục đang có, tức là hạng mục thứ 13 không được sao chép. Các hạng mục được thu thập vẫn còn lại trong Office **Clipboard** cho đến khi bạn thoát tất cả các chương trình của Office đang chạy trên máy tính.
- Bạn không thể dán nhiều hạng mục hoặc hiển thị Office **Clipboard** trong một chương trình khác với **Microsoft Excel, Word, Access, OutloOK**, hoặc **Powerpoint**. Tuy nhiên, nếu Office **Clipboard** được hiển thị trong bất cứ chương trình Office, thì bất cứ hạng mục nào mà bạn cắt hoặc sao chép trong bất cứ chương trình nào đó cũng đều được thu thập một cách tự động.

### **XIII. Xóa tất cả các hạng mục khỏi office Clipboard**

- Trên thanh công cụ **Clipboard**, nhấp **clear Clipboard**.
- Để hiển thị thanh công cụ **Clipboard**, trở đến Toolbar trên **Menu View** rồi nhấp **Clipboard**.

### **XIV. Xem nội dung của office Clipboard**

- Trên **Menu View**, trở đến **Toolbars**, rồi nhấp **Clipboard**. Nếu thanh công cụ **Clipboard** bị neo cố định hãy nhấp **Items**.

### **XV. Hiển thị trường dữ liệu từ một trường trong một Table hoặc Query.**

- Mở một **Form**, một **Report** hoặc một trang truy cập trong **Design View**.

- Nếu bạn muốn sử dụng **Wizard** để trợ giúp trong việc tạo ra **Control**, phải chắc chắn rằng công cụ **Control Wizards** trong **Toolbox** được nhấn.
- Hiện thị danh sách trường bằng cách nhấp **Field List** trên thanh công cụ.
- Thực hiện một trong các thao tác sau đây:
- Chọn một trong nhiều trường trong danh sách trường trên một **Form** hoặc **Report**.

**Thực hiện một trong các thao tác sau:**

- Rê trường từ danh sách trường và đặt chúng trên **Form** hoặc **Report**.
- Rê trường từ danh sách trường và đặt nó trên trang truy cập dữ liệu. Trong hộp thoại layout **Wizard**, hãy nhấp individual **Controls**.

**Lưu ý:**

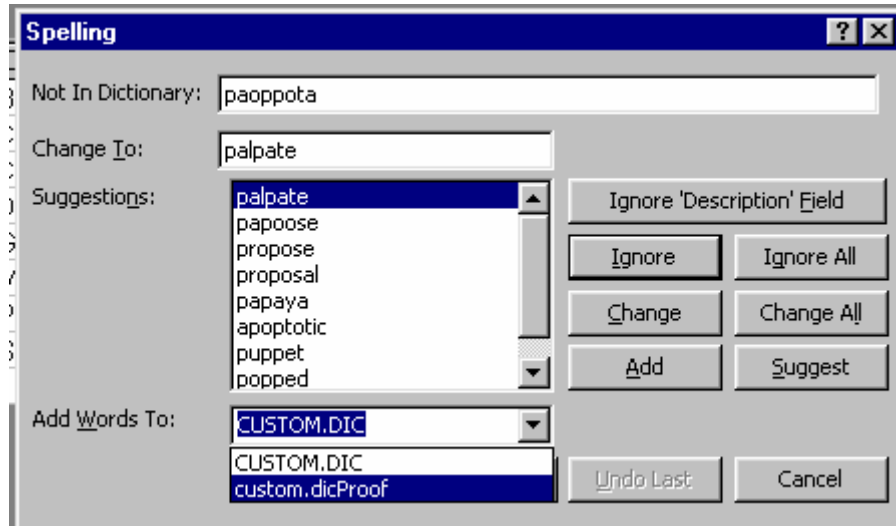
- Bạn phải rê trường, **Table** hoặc **Query** từ danh sách trường. Nếu bạn nhấp trường, **Table** hoặc **Query** trong danh sách trường rồi nhập **Form**, **Report**, hoặc trang truy cập dữ liệu, thì **Microsoft Access** sẽ không tạo **Control**. **Access** đặt một hộp **Text** trên **Form**, **Report** hoặc trang truy cập dữ liệu cho mỗi một trường mà bạn chọn trong danh sách trường. Nếu bạn chọn một **Table** hoặc **Query** trong danh sách trường của trang truy cập dữ liệu, thì **Access** đặt một hộp **Text** cho mỗi một trường trong **Table** hoặc **Query** trên trang truy cập dữ liệu đó. Mỗi một hộp **Text** thì bao quanh một trường trong nguồn dữ liệu được gạch dưới. Mỗi một hộp **Text** cũng có một nhãn được đính kèm theo mục đích.
- Đặt góc bên trái phía trên cửa biểu tượng nơi mà bạn muốn góc bên trái phía trên cửa **Control** chính (không có label cửa nó) có mặt, và nhả nút chuột.

## **XVI. Kiểm tra chính tả dữ liệu trong DataSheet View**

1. Mở một **DataSheet**.
2. Chọn các **Record**, các cột các trường, hoặc **Text** bên trong một trang mà bạn muốn kiểm tra chính tả.
3. Nhấp **Spelling** trên thanh công cụ.

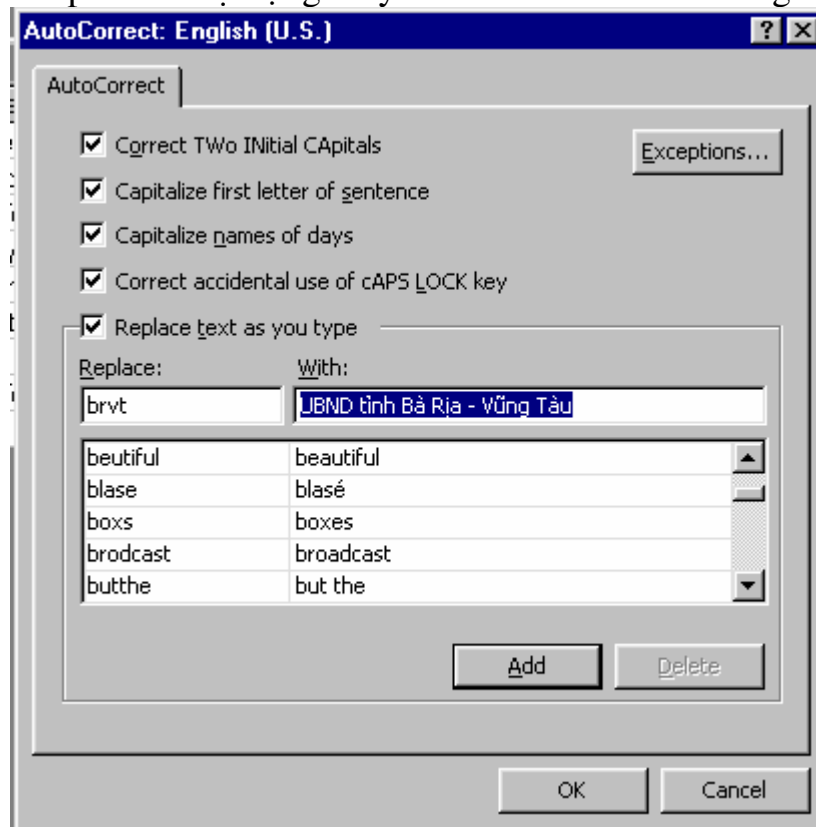
## **XVII. Kiểm tra chính tả dữ liệu trong Form View**

1. Mở một **Form View**.
2. Chọn trường, hoặc **Text** bên trong một trang mà bạn muốn kiểm tra chính tả.
3. Nhấp **Spelling** trên thanh công cụ.



### XVIII. Tự động chỉnh sửa lỗi lúc nhập dữ liệu

- Sử dụng **AutoCorrect** để tự động chỉnh sửa lỗi **Text** mà bạn thường xuyên mắc phải, và tự động thay thế các chữ viết tắt bằng các tên dài.

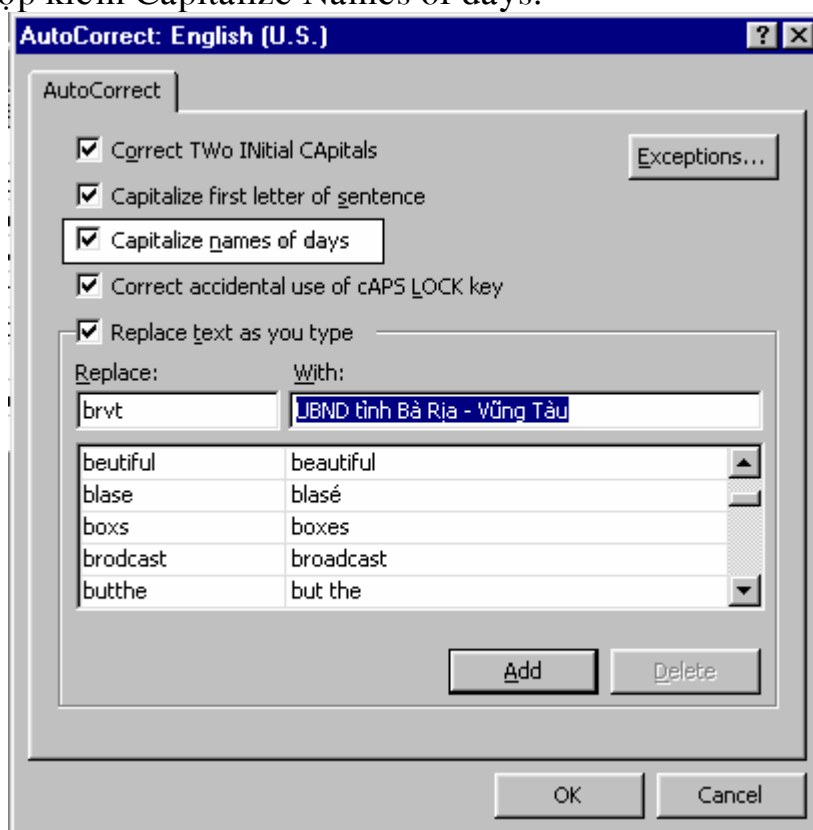


- **Ví dụ**, bạn có thể nhập vào một từ "brvt" dành cho chữ "UBND tỉnh Bà Rịa – Vũng Tàu". Bất cứ lúc nào bạn gõ nhập brvt theo sau bởi một khoảng trống hoặc dấu chấm câu **Microsoft Access** tự động thay thế "brvt" bằng UBND tỉnh Bà Rịa – Vũng Tàu.
- Trên **Menu Tools**, nhấp **AutoCorrect**.
- Chọn hộp kiểm replace **Text** as you **Type**.

- Trong hộp replace, gõ nhập **Text** mà bạn muốn chỉnh sửa một cách tự động.
- Trong hộp with, hãy gõ nhập **Text** được chỉnh sửa.
- Nhấp **Add**.

### XIX. Tự động viết hoa các tên, ngày lúc nhập dữ liệu

- Trên **Menu Tools** nhấp **AutoCorrect**.
- Chọn hộp kiểm Capitalize Names of days.



### XX. Chỉnh sửa hai mẫu tự được viết hoa trong một hàng lúc nhập dữ liệu

- Trên **Menu Tools**, nhấp **AutoCorrect**
- Chọn hộp kiểm correct Two initial capitals.

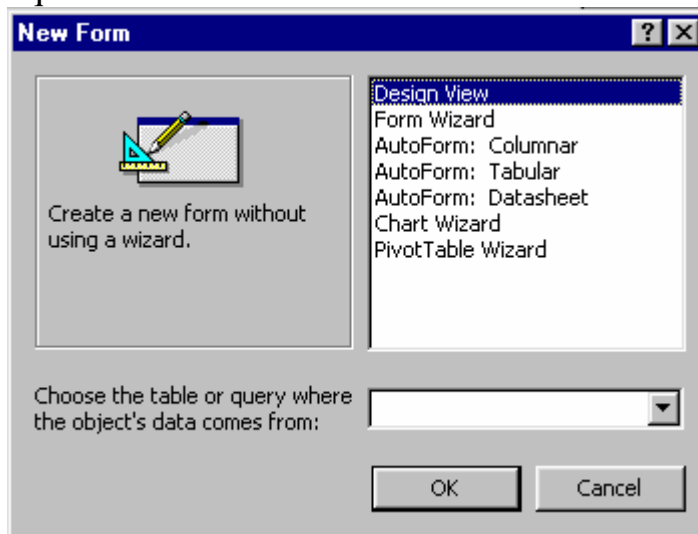


## Chương 6:

# Các tác vụ căn bản về Form

## I. Tạo ra một Form

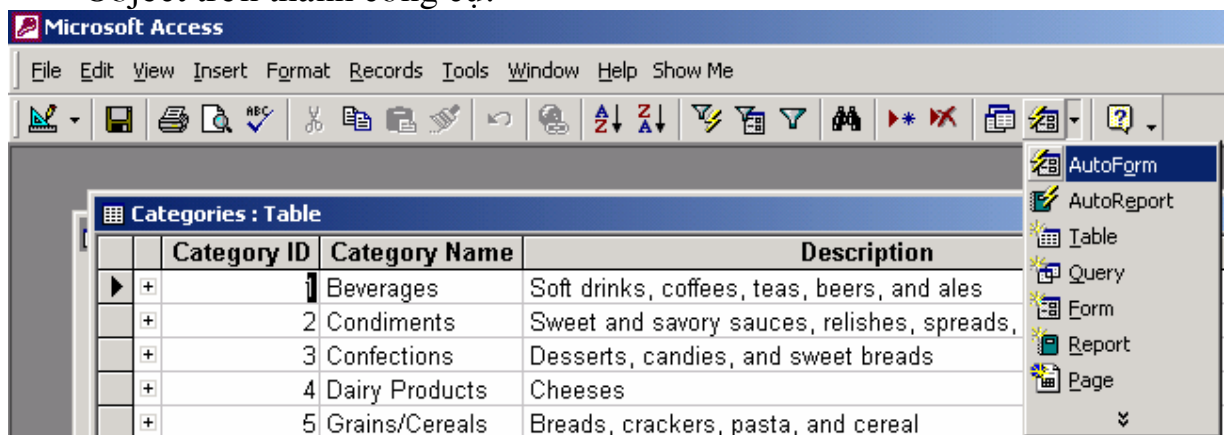
- Tạo ra một **Form** bằng cách sử dụng **AutoForm**: Với **AutoForm**, bạn chọn một nguồn **Record** hoặc là Columnar, **Tabular**, hoặc là thiết kế **DataSheet**; **AutoForm** tạo một **Form** nhằm sử dụng tất cả các trường từ nguồn **Record** mà bạn đã chọn và tất cả các trường từ các nguồn **Record** liên quan của nó.



- Tạo một **Form** với một **Wizard**: **Wizard** yêu cầu bạn các câu hỏi chi tiết về các nguồn **Record**, các trường, layout, và dạng và bạn muốn và tạo ra một **Form** dựa trên câu trả lời của bạn.

## II. Tạo một Form hoặc Report từ các Record được lọc

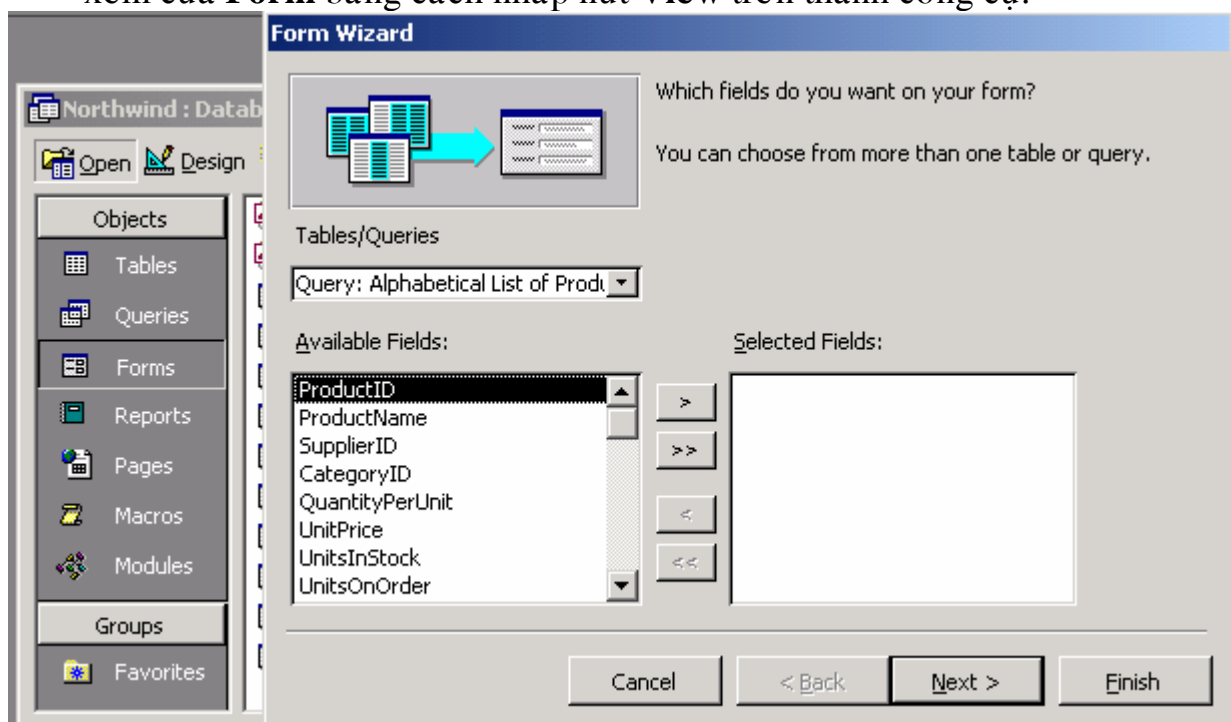
- Với các **Record** được lọc đã hiển thị, hãy nhấp mũi tên nằm kế nút New Object trên thanh công cụ.



- Để sử dụng auto**Form** hoặc auto**Report**, hãy nhấp auto**Form** hoặc auto**Report**.
- Để chỉ định kiểu **Form** hoặc **Report** mà bạn muốn tạo hãy nhấp **Form** hoặc **Report**, rồi thực hiện mục chọn lựa của mình.
- Đối tượng mới sử dụng làm nguồn **Record** của nó, **Table** hoặc **Query** mà ở đó bộ lọc được tạo ra và nó kế thừa bộ lọc.
- Hoàn tất thiết kế hoặc tạo ra bất cứ thay đổi nào mà bạn muốn cho **Form** hoặc **Report**, rồi lưu nó.

### III. Mở một Form

- Trong cửa sổ **Database**, hãy nhấp **Forms** trong phần **Objects**.
- Nhấp **Form** mà bạn muốn mở
- Nhấp nút **Open** để mở **Form** trong **Form View**, hoặc nút **Design** để mở **Form** trong **Design View**.
- Khi bạn đã mở một **Form**, bạn có thể dễ dàng hoán chuyển giữa 3 khung xem của **Form** bằng cách nhấp nút **View** trên thanh công cụ.



### IV. Các khung xem của một Form

- Các **Form** có 3 khung xem: **Design View**, **Form View** và **DataSheet View**.

- Để tạo một **Form**, bạn làm việc trong **Design View**, việc xem một **Form** trong **Design View** thì giống như ngồi cạnh một bàn làm việc mà chung quanh là các công cụ hữu ích.
- Sau khi bạn đã tạo một **Form** trong **Design View**, bạn có thể xem nó trong **Form View** hoặc **DataSheet View**.

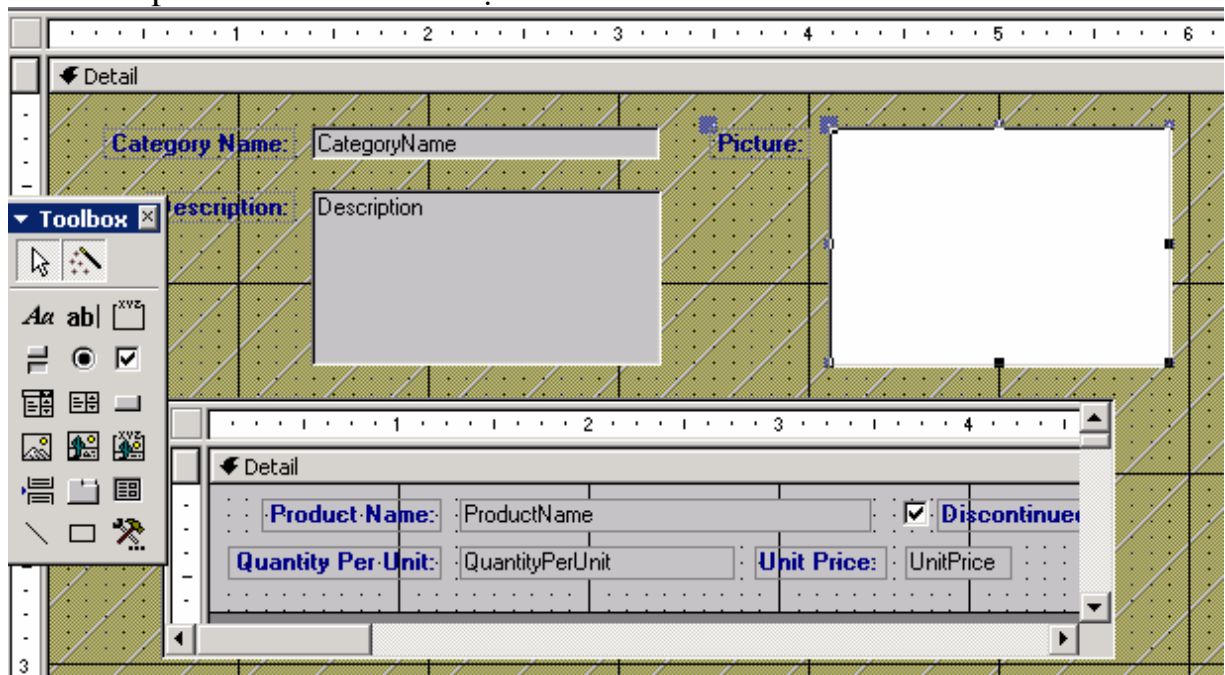
## V. Các Form mẫu trong Form Wizard:

Khi chọn form Wizard, Access sẽ hướng dẫn ta chọn một trong các loại Form mẫu sau:

- Single column : mẫu một cột
- Tubular : mẫu dạng bảng
- Graph : mẫu đồ thị
- Main/SubForm : mẫu Form bên trong có Form con khác
- Auto Form : mẫu tự động

## VI. Chọn một Form hoặc một Report

- Mở 1 **Form** trong **Design View** hoặc 1 **Report** trong **Design View**.
- Nhấp **Form Selector** hoặc **Report Selector**.
- Nếu các cây thước vẫn không hiển thị, bạn có thể nhấp select **Form** hoặc select **Report** trên **Menu Edit**, hoặc bạn có thể nhấp thang xám về bên phải của **Form** để chọn nó.



## VI. Chọn một mục chọn trên Form hoặc Report

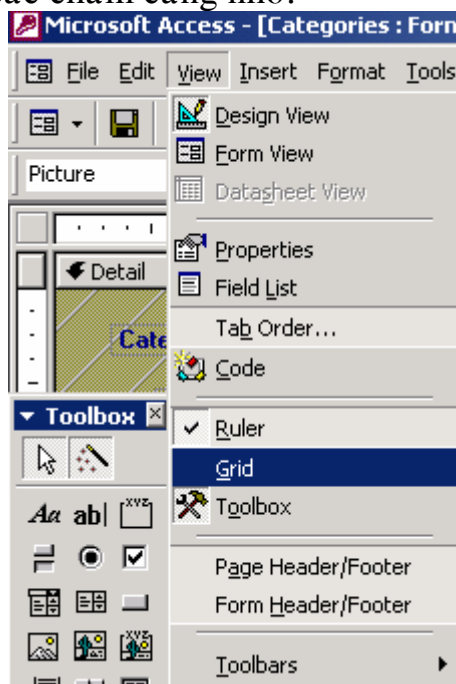
- Mở một **Form** hoặc **Report** trong **Design View**.
- Nhấp **Selector** hoặc bất cứ nơi nào trên thanh chọn.
- Bạn cũng có thể nhấp chọn bất cứ nơi nào trên nền mục chọn để chọn nó.

## VII. Hiện thị hoặc che giấu lưới trên một Form, Report hoặc trang truy cập dữ liệu.

- Mở một **Form**, **Report** hoặc một trang truy cập dữ liệu trong **Design View**
- Trên **Menu View** nhấp grid.

## VIII. Thay đổi khoảng cách giữa các chấm trên lưới Form hoặc Report

- Mở một **Form** hoặc một **Report** trong **Design View**.
- Nhấp đúp **Form Selector** hoặc **Report Selector** để mở trang đặc tính.
- Để thay đổi các dấu chấm nằm ngang, hãy gõ nhập một giá trị mới dành cho tính chất GridX. Để thay đổi các dấu chấm thẳng đứng, hãy gõ nhập một giá trị mới dành cho tính chất GridY. Con số chỉ định càng lớn thì khoảng cách giữa các chấm càng nhỏ.



## IX. Cài đặt các tính chất dành cho một Form, Report, section hoặc Control

- Mở một **Form** hoặc một **Report** trong **Design View**. Nếu tính chất AllowDesignChanges của một **Form** được cài đặt sang **All Views**, bạn cũng có thể cài đặt các tính chất trong **Form View** và **DataSheet View**.
- Thực hiện một trong các thao tác sau đây:
- Để đặt các tính chất dành cho một **Form** hoặc **Report**, hãy nhập đúp lên **Form Selector** hoặc **Report Selector** để mở trang tính chất dành cho **Form** hoặc **Report** đó.
- Để cài đặt các tính chất dành cho một mục trên một **Form** hoặc **Report**, hãy nhập đúp lên **section Selector** để mở trang tính chất dành cho mục đó.
- Để cài đặt các tính chất dành cho một **Control**, phải chắc chắn **Control** đó được chọn, rồi nhập **Properties** trên thanh công cụ. Bạn cũng có thể nhập đúp lên một **Control** để mở trang tính chất của nó.
- Trong trang tính chất, hãy nhấp tính chất mà bạn muốn cài đặt, rồi thực hiện một trong các thao tác sau đây:
- Nếu một mũi tên xuất hiện trong hộp **Property**, hãy nhập nó rồi chọn một giá trị từ danh sách.
- Gõ nhập một cài đặt hoặc biểu thức vào hộp **Property**.
- Nếu nút **Build** xuất hiện kê' bên hộp **Property**, hãy nhập nó để hiển thị một **builder** hoặc để hiển thị một hộp thoại cho bạn chọn lựa các **builder**.
- Ví dụ, bạn có thể sử dụng **Query Builder** để thay đổi thiết kê' của một **Query** trực thuộc dành cho một **Form** hoặc **Report**.
- Nếu cần nhiều chỗ để nhập hoặc cài đặt một tính chất hãy nhấp **Shift\_F12** (hoặc nhấp phải, rồi nhấp **Zoom**) để mở hộp **Zoom**.

## X. Cài đặt kích thước của một Form lúc lần đầu tiên mở nó

- Mở một **Form** trong **Design View**.
- Định kích thước của **Form** bằng cách rê các đường biên của cửa sổ **Form**.
- Nhấp **Save** trên thanh công cụ.

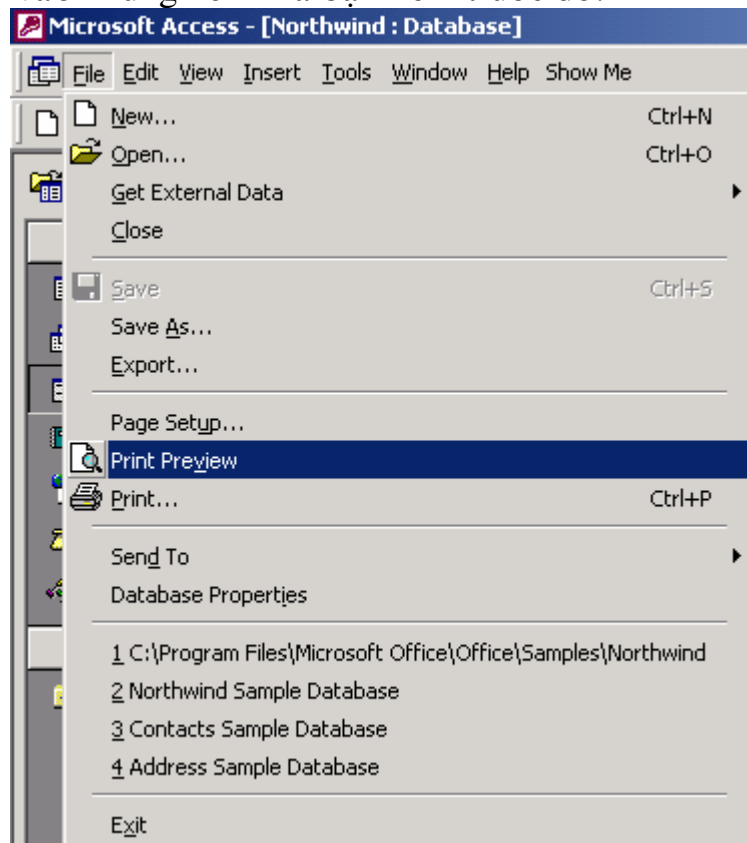
## XI. Chuyển đổi sang phiên bản được lưu của một Form hoặc Report

- Bạn có thể bỏ qua bất cứ thay đổi nào - kể cả các thay đổi đối với mã **Microsoft Visual Basic** - mà bạn đã tạo ra trong **Design View** của một **Form** hoặc **Report** và chuyển đổi nó sang phiên bản được lưu lần cuối của **Form** hoặc **Report** đó.
- Để chuyển đổi phiên bản được lưu sau cùng, hãy thực hiện :
- Trên men **File**, nhấp **Revert**.

**Lưu ý:** Lệnh này không có sẵn trừ khi bạn đã mở một **Form** hoặc **Report** trong **Design View** và đã thực hiện các thay đổi cho nó. Lệnh **Revert** không có sẵn trong **Design View** của một trang truy cập dữ liệu. Cách mà **Microsoft Access** hiển thị một **Form**

## XII. Xem trước một Form

- Để xem trước một **Form** từ **Design View**, **Form View**, hoặc **DataSheet View**
- Nhấp **Print PreView** trên thanh công cụ.
- Để trở về **Design View**, **Form View** hoặc **DataSheet View** hãy nhấp **View** trên thanh công cụ.
- Để xem trước một **Form** từ cửa sổ **Database**
- Nhấp **Form** trong phần **Objects**
- Chọn **Form** mà bạn muốn xem trước.
- Nhấp **Print PreView** trên thanh công cụ.
- Cách mà **Microsoft Access** hiển thị một **Form** trong **Print PreView** phụ thuộc vào khung xem mà bạn xem trước đó.



## XIII. In các Record được chọn trong một Form hoặc DataSheet

- 1. Mở **Form** trong **Form View** hoặc **DataSheet View**.

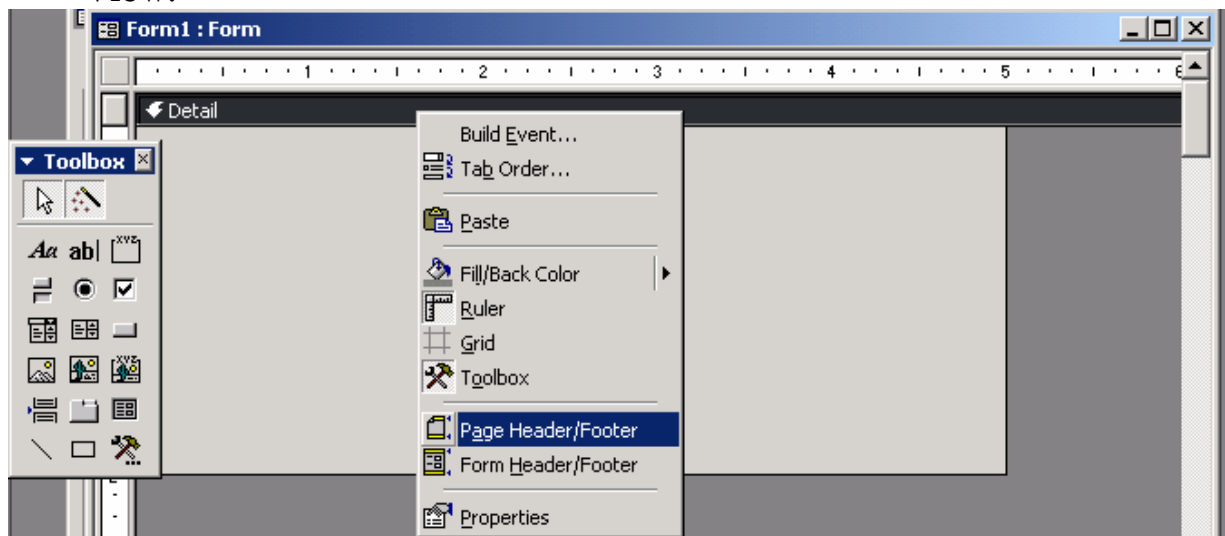
- 2. Chọn Record mà bạn muốn in bằng cách nhấp **Record Selector** hoặc bằng cách nhấn SHLFT+SPACEBAR (nếu bạn đang ở trong chế độ **Edit**, nhấn F2 trước khi nhấn SHLFT+SPACEBAR). Nếu bạn chọn nhiều **Record**, hãy nhấp **Record Selector** đầu tiên rồi rê ngang qua **Record** cuối cùng mà bạn muốn in.
- Trên **Menu File** nhấp chọn **Print**
- Trong hộp thoại **Print** trong phần **Print range**, hãy nhấp **selected Records**.

#### XIV. Cho phép thay đổi với các tính chất của Form trong Form và DataSheet View

- Mở **Form** trong **Design View**
- Nhấp đúp lên **Form Selector** để mở trang tính chất dành cho **Form** đó.
- Cài đặt tình chất AllowDesignChanges sang **All Views**.
- Lúc tính chất AllowDesignChanges được cài đặt sang **All Views**, bạn có thể chỉnh sửa các tính chất của **Form** và các **Control** của nó bằng cách hiển thị trang tính chất trong **Form View** và **DataSheet View**.

#### XV. Bổ sung hoặc xóa bỏ một Header

- Mở **Form** trong **Design View**.
- Nhấp **Form Header/Footer** hoặc **Pages Header/Footer** trên **Menu View**.

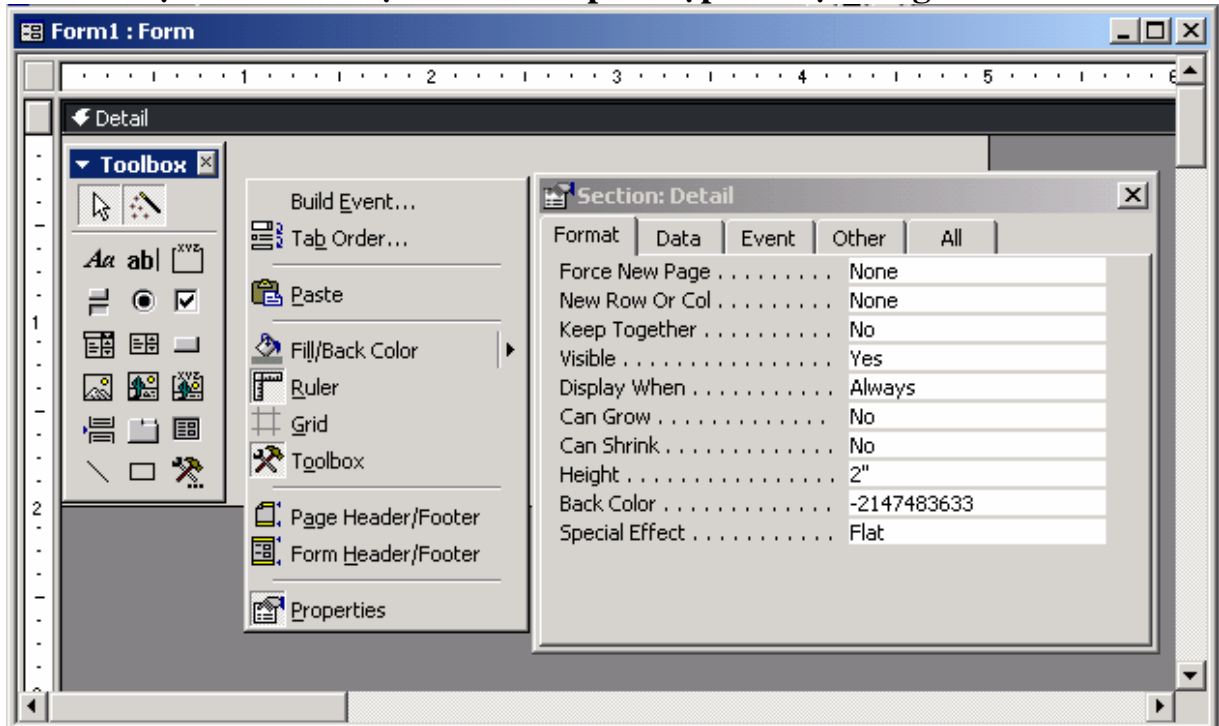


#### XVI. Che giấu các dòng chia mục trong một Form (selection divider lines)

- Mở **Form** trong **Design View**.
- Nhấp đúp lên **Form Selector** để mở trang tính chất .

- Trong hộp DividingLines, nhấp NO.

## XVII. Định kích cỡ một Form cho phù hợp với nội dung của nó



- Mở Form trong Design View.
- Nhấp Size to Fit Form trên Menu Windows.
- Nhấp Save trên thanh công cụ để lưu kích cỡ của Form.
- Lúc bạn nhấp Size To Fit Form, Microsoft Access chỉnh sửa kích cỡ của Form phụ thuộc vào cài đặt tính chất DefaultView của Form đó là single Form hoặc Continuous Forms.
- Single Form. Nếu Record được hiển thị nhỏ hơn cửa sổ Form, Size To Fit Form đặt sang kích cỡ của Record.
- Còn nếu Record lớn hơn cửa sổ Form, thì Size To Fit Form mở rộng cửa sổ Form để hiển thị càng nhiều Record càng tốt.
- Continuous Forms: Nếu chỉ có phần Record ở đáy được hiển thị, thì Size To Fit Form nhỏ Record đó. Còn nếu chỉ có một Record được hiển thị theo từng thành phần, thì Size To Fit Form mở rộng cửa sổ để minh họa càng nhiều Record càng tốt.

## XVIII. Định tâm một Form trong cửa sổ trình ứng dụng lúc lần đầu tiên nó được mở

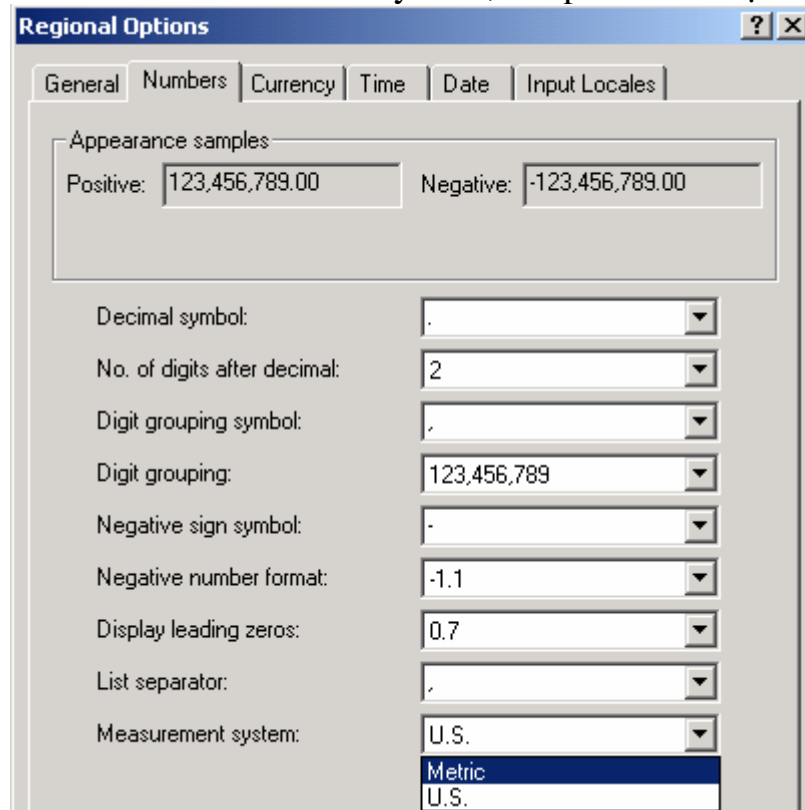
- 1. Mở một Form trong Design View.
- 2. Nhấp đúp lên Form Selector để mở trang tính chất.
- Cài đặt tính chất Autocenter sang Yes.



- Lần kế tiếp lúc bạn mở **Form**, nó sẽ được đặt ở giữa trong cửa sổ trình ứng dụng.

### XIX. Thay đổi đơn vị đo của thước trong một Form hoặc Report

- Nhấp nút start của Windows, trở đến settings, rồi nhấp **Control Panel**.
- Nhấp đúp biểu tượng **Regional Settings**.
- Nhấp nhãn **Number**.
- Trong danh sách Measurement System, nhấp Metric hoặc US.



### XX. Thay đổi thứ tự Tab trong một Form.

- Mở **Form** trong **Design View**.
- Trên **Menu View**, nhấp **Tab Order**.
- Trong phần section, hãy nhấp mục mà bạn muốn thay đổi.

#### **Thực hiện một trong các thao tác sau đây:**

- Nếu bạn muốn **Microsoft Access** tạo ra thứ tự nhãn từ trái sang phải và từ trên xuống dưới, hãy nhấp **Auto Order**.
- Nếu bạn muốn tạo ra thứ tự nhãn theo ý thích của mình, hãy nhấp **Selector** dành cho **Control** mà bạn muốn di chuyển. (Nhấp và rê để chọn nhiều **Control** cùng một lúc). Nhấp **Selector** một lần nữa và rê **Control** đến vị trí mong muốn trong danh sách.
- Nhấp **OK**.

- Hoán chuyển sang **Form View** để thử nghiệm **Tab Order**. (Trong **Design View**, **Tab Order** luôn luôn là thứ tự mà ở đó bạn đã tạo ra các **Control**).

### XXI. Xóa bỏ Menu tắt khỏi một Form

- Mở một **Form** trong **Design View**.
- Nhấp đúp **Form Selector** để mở trang tính chất của **Form**.
- Trong hộp tính chất **ShortCutMenu**, hãy nhấp No.

### XXII. Xóa bỏ Record Selector khỏi một Form

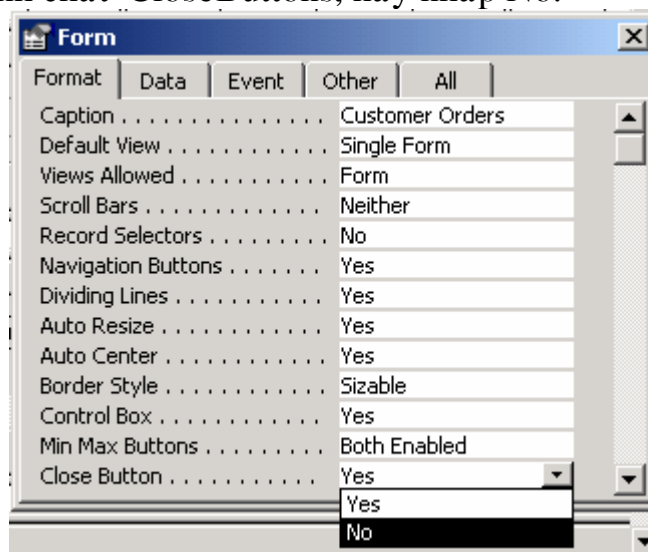
- Mở 1 **Form** trong **Design View**.
- Nhấp đúp **Form Selector** để mở tính chất của **Form**.
- Trong hộp tính chất **Recorder Selectors**, hãy nhấp No.

### XXIII. Xóa bỏ một nút điều khiển

- Mở 1 **Form** trong **Design View**.
- Nhấp đúp **Form Selector** để mở tính chất của **Form**.
- Trong hộp tính chất **NavigationButtons**, hãy nhấp No.

### XXIV. Hủy chọn nút Close trên Form

- Mở 1 **Form** trong **Design View**.
- Nhấp đúp **Form Selector** để mở tính chất của **Form**.
- Trong hộp tính chất **CloseButtons**, hãy nhấp No.



## Chương 7:

**SubForm – Nội dung và cách hoạt động.****I. Một subform là một Form nằm bên trong một Form.**

- Một **subForm** là một **Form** nằm bên trong một **Form**. **Primary Form** (**Form** sơ cấp) được gọi là **Main Form** (**Form** chính), và **Form** nằm bên trong **Form** đó được gọi là **subForm** (**Form** con). Một tổ hợp **Form/subForm** thường được tham chiếu dưới dạng một **Form** phủ hệ (hierarchical **Form**), một master/detail **Form** (**Form** chủ/chi tiết), hoặc một parent/child **Form** (**Form** bố/con).
- Các **subForm** đặc biệt hữu ích lúc bạn muốn minh họa dữ liệu từ các **Table** hoặc **Query** với quan hệ một-đối-nhiều.
- **Ví dụ**, bạn có thể tạo một **Form** với một **subForm** để minh họa dữ liệu từ một **Table** Categories và một **Table** Products. Dữ liệu trong **Table** Categories là phía “**one**” của quan' hệ. Dữ liệu trong **Table** Products là phía “**many**” của quan hệ mỗi một hạng mục thì có nhiều sản phẩm.
- **Main Form** và **subForm** theo kiểu **Form** này được liên kết với nhau để chỉ có **subForm** là hiển thị các **Record** có liên quan đến **Record** hiện hành trong **main Form**.
- Lấy **Ví dụ**, lúc **main Form** hiển thị hạng mục Beverages, thì **subForm** chỉ hiển thị các sản phẩm trong hạng mục Beverages.
- Lúc bạn sử dụng một **Form** với một **subForm** để nhập các **Record** mới, **Microsoft Access** lưu **Record** đang có trong **main Form** lúc bạn nhập **subForm**. Điều này bảo đảm rằng các **Record** trong **Table** “**many**” sẽ có một **Record** trong **Table** “**one**” liên quan đến nó. Nó cũng tự động lưu mỗi một **Record** khi bạn bổ sung nó vào **subForm**.
- Một **subForm** có thể được hiển thị dưới dạng một **DataSheet**, dưới dạng hình minh họa trước đó, hoặc nó có thể được hiển thị dưới dạng một **Form** đơn hoặc một **Form** liên tục. Một **main Form** chỉ có thể được hiển thị dưới dạng một **Form** đơn (single **Form**).
- Một **main Form** thì có bất cứ số **subForm** nào nếu bạn đặt mỗi một **subForm** vào **main Form**. Bạn cũng có thể kết tổ lên đến 10 cấp **subForm**. Điều này có nghĩa rằng bạn có một **subForm** bên trong một **main Form**, và bạn có thể có **subForm** khác bên trong **subForm** đó, và v.v...

- **Ví dụ**, bạn có thể có một main **Form** hiển thị các khách hàng, một **subForm** nhằm hiển thị đơn đặt hàng, và **subForm** khác nhằm hiển thị chi tiết về đơn đặt hàng.

## II. Các kiểu subForm

- Lúc bạn tạo một **subForm**, bạn có thể thiết kế để nó được hiển thị chỉ trong **DataSheet View**; chỉ trong **Form View** dưới dạng **Form** đơn hoặc **Form** liên tục (single or continuous **Form**); hoặc trong cả hai khung xem. Một **subForm** hiển thị dưới dạng một **Form** đơn hoặc **Form** liên tục đều có thể là đơn giản hoặc nó có thể được tùy nghi sử dụng để chứa một **Header**, **Footer**, v.v...

## III. Microsoft Access liên kết các main Form và subForm

- Bạn thường sử dụng một **subForm** để hiển thị dữ liệu từ các **Table** hoặc **Query** đã có quan hệ một-đối-nhiều.
- **Ví dụ**, một **Table** Categories và một **Table** Products có một quan hệ một-đối-nhiều - mỗi một hạng mục đều có nhiều sản phẩm.
- Lúc bạn tạo một **Form** và **subForm** dựa trên các **Table** đã có quan hệ một-đối-nhiều, một main **Form** biểu thị phía "one" (phía "một") của quan hệ và **subForm** biểu thị phía "**many**" (phía "nhiều") của quan hệ. Main **Form** được đồng bộ với **subForm** để **subForm** chỉ hiển thị các **Record** có liên quan đến **Record** trong main **Form**. Nếu bạn sử dụng một **Wizard** để tạo một **subForm**, hoặc nếu bạn rê một **Form**, **Table**, hoặc **Query** từ cửa sổ **Database** sang **Form** khác để tạo ra một **subForm**, thì **Microsoft Access** tự động đồng bộ hóa main **Form** với **subForm** nếu cả hai điều nghiệm đúng phần sau đây:
- Bạn đã xác định các quan hệ dành cho các **Table** mà bạn chọn, hoặc bạn đã xác định các quan hệ dành cho các **Table** thuộc **Query** mà bạn chọn (**Tables underlying queries**).
- Main **Form** dựa trên một **Table** với từ khóa chính và **subForm** thì dựa trên một **Table** có chứa một trường có cùng tên với từ khóa chính và có cùng kiểu dữ liệu hoặc kiểu dữ liệu tương thích và cùng kích cỡ trường.
- **Ví dụ**, nếu từ khóa chính của **Table** thuộc main **Form** là một trường **AutoNumber** và tính chất **FieldSize** được cài đặt sang Long Integer, thì trường tương ứng trong **Table** thuộc **subForm** sẽ là một trường **Number** với tính chất **FieldSize** được cài đặt sang Long Integer. Nếu bạn chọn

một **Query** hoặc nhiều **Query**, thì các **Table** trực thuộc dành cho **Query** hoặc các **Query** đó phải đáp ứng các điều kiện này.

### Lưu ý:

**Microsoft Access** sử dụng các tính chất **LinkMasterFields** và **LinkChildFields** của **subForm Control** để liên kết với main **Form** và **subForm**. Nếu vì một lý do nào đó, **Microsoft Access** không liên kết **Form** với **subForm**, thì bạn có thể tự mình liên kết các **Form**.

## IV. Liên kết một main Form và một subForm

- Nếu bạn tạo một **subForm** nhờ **Form Wizard** hoặc **SubForm Wizard**, **Microsoft Access** tự động liên kết main **Form** với **subForm** trong một số điều kiện.

Order ID	Order Date	Required Date	Shipped Date
10389	20-Dec-1996	17-Jan-1997	24-Dec-1996
10410	10-Jan-1997	07-Feb-1997	15-Jan-1997
10411	10-Jan-1997	07-Feb-1997	21-Jan-1997
10431	30-Jan-1997	13-Feb-1997	07-Feb-1997

Product Name	Unit Price	Quantity	Discount	Extended Price
Outback Lager	\$12.00	30	0%	\$360.00
Tarte au sucre	\$39.40	20	0%	\$788.00
Pâté chinois	\$19.20	15	0%	\$288.00
Ikura	\$24.80	16	0%	\$396.80

- Mở main **Form** trong **Design View**.
- Phải chắc chắn **subForm Control** được chọn, rồi nhấp **Properties** trên thanh công cụ để hiển thị trang tính chất dành cho **subForm Control**.
- Gõ nhập các trường liên kết hoặc các **Control** trong các tính chất **LinkchildFields** và **LinkmasterFields**. Để nhập nhiều trường hoặc **Control** liên kết hãy tách các tên trường và **Control** bằng một dấu chấm phẩy. Nếu bạn nhập vào nhiều trường hoặc tên **Control**, bạn phải nhập vào các tên theo thứ tự giống nhau trong cả hai tính chất.
- Trong hộp tính chất **LinkchildFields**, hãy nhập tên của mỗi một trường liên kết trong **subForm**. (Bạn không thể sử dụng tên của một **Control** trong tính chất này).

- Trong hộp tính chất **LinkmasterFields**, hãy nhập tên của mỗi một trường tính chất hoặc **Control** trong main **Form**.
- Nếu bạn không chắc chắn những gì mà trường liên kết hoặc **Control** cần cho mỗi một tính chất, hãy nhập nút **Build** kê' bên hộp tính chất để mở **SubForm/supReport Field Linker**.
- Hoán chuyển sang **Form** việen để thử nghiệm liên kết.

## V. Chỉ định khung xem (View) dành cho một subForm

- Nếu bạn sử dụng **Form Wizard** hoặc **SubForm Wizard** để tạo một **subForm**, **Microsoft Access** cài đặt tính chất **DefaultView** của **subForm** phù hợp với các chọn lựa mà bạn tạo ra trong hộp thoại **Wizard**. Sử dụng thủ tục này nếu bạn đang tạo ra một **subForm** cho chính mình hoặc nếu bạn muốn thay đổi khung xem mà bạn đã chỉ định trong **Wizard**.
  - Mở min **Form** trong **Design View**. **SubForm** được hiển thị trong **Design View** trong **subForm Control**.
  - Phải bảo đảm **subForm Control** được chọn. Sau khi nó được chọn, bạn có thể nhấp một lần nữa bên trong **Control** để cài đặt các tính chất của **Form**, y như bạn sẽ thực hiện nếu bạn mở một **subForm** trong **Design View** từ cửa sổ **Database**.
  - Nhấp đúp **Form Selector** để mở trang tính chất của **subForm**.
  - Trong hộp tính chất **DefaultView**, hãy nhấp **Single Form**, **Continuous Forms**, hoặc **DataSheet**.
  - Trong hộp tính chất **Viewsallowed**, hãy nhấp cài đặt **Form** nếu bạn không muốn phải chuyển sang **DataSheet View**. Hãy nhấp cài đặt **DataSheet** nếu bạn không muốn chuyển sang **Form View**. Nếu bạn muốn có thể chuyển tới và lùi giữa các khung xem **DataSheet View** và **Form View**, hãy nhấp cài đặt **Both**.

## VI. Di chuyển giữa một Main Form và một subForm trong Form View bằng cách sử dụng phím TAB

- Để di chuyển từ trường sau cùng trong main **Form** tới trường đầu tiên trong **subForm**, hãy nhấn **TAB**.
- Để di chuyển từ trường đầu tiên trong **subForm** trở về với trường sau cùng trong main **Form**, hãy nhấn **CTRL+SHLFT+TAB**.
- Để di chuyển từ trường sau cùng trong **subForm** sang trường kế tiếp trong main **Form** hoặc sang trường đầu tiên trong **Record** kế tiếp hoặc trang kế tiếp của main **Form**, hãy nhấn **CTRL+TAB**. Nếu tính chất

Cycle của **Form** được cài đặt sang **Current Record** hoặc **Current Page**, bạn phải sử dụng các nút navigation hoặc các phương pháp khác được cung cấp cho **Form** đó để di chuyển sang **Record** kế tiếp.

- Sử dụng chuột: Để hoán chuyển từ main **Form** sang **subForm**, hãy nhấp bất cứ nơi nào trong **subForm**.
- Để hoán chuyển từ **subForm** trở về lại với main **Form**, hãy nhấp một **Control** trên main **Form** (chứ không phải được nhấp trên nền của main **Form**).

## VII. Hoán chuyển giữa **DataSheet View** và **Form View** trong một **subForm**

- Mở main **Form** trong **Form View**.
- Nhấp bất cứ nút nào trong **subForm**.
- Trên **Menu View**, hãy nhấp **SubForm DataSheet**.
- **Lưu ý:** Tính chất **Views Allowed** dành cho **subForm** phải được cài đặt sang **Both** để hoán chuyển các khung xem.

## VIII. Tạo ra một **subForm** trông giống như một **DataSheet** để hiển thị một **Header** và **Footer**

- Bạn có thể hiển thị một **Form Header** và **Footer** trên một **subForm** và sắp xếp các **Text** box và các **Control** khác để **subForm** trông giống như một **DataSheet** lúc bạn mở **Form** trong **Form View**.
- Mở **Form** mà bạn muốn sử dụng làm **subForm** trong **Design View**.
- Bổ sung một **Form Header** và **Footer** nếu bạn chưa thực hiện điều đó (Trên **men View**, nhấp **Form Header/footer**).
- Trong mục chi tiết, hãy sắp xếp các hộp **Text** và các **Control** khác nhằm hiển thị dữ liệu trong một hàng đơn.
- Trong **Form Header**, sắp xếp các nhãn dành cho các **Text** box và các mục **Control** khác. Phải chắc chắn rằng bạn giống thẳng hàng các nhãn trong **Form Header** với các **Control** liên kết của chúng trong mục chi tiết.
- Trong **Form Footer**, nếu bạn muốn tính tổng các **Record** trong **subForm**, hãy tạo các **Control** được tính toán.
- Định kích thước phù hợp cho mỗi một mục.
- **Ví dụ.** hãy tạo ra kích thước của mục chi tiết của một hàng trong một **DataSheet**. Nếu bạn không có bất cứ **Control** nào trong **Form Footer**,

hãy cài đặt tính chất Height của nó sang 0 hoặc cài đặt tính chất Visible của nó sang No.

- Nhấp đúp lên **Form Selector** để mở trang tính chất dành cho **Form** đó.
- trong hộp tính chất **DefaultView**, hãy nhấp **Continuous Forms**. Cài đặt này hiển thị các **Record** trong một **Form** lần lượt từ **Record** này đến **Record** khác giống như một **DataSheet**.

## IX. Tạo một Control tính toán trên một Form hoặc Report

- Mở một **Form** trong **Design View** hoặc một **Report** trong **Design View**.
- Nhấp công cụ trong **Toolbox** dành cho kiểu **Control** mà bạn muốn sử dụng làm **Control** tính toán.
- **Lưu ý:** Một **Text** box là một kiểu **Control** phổ biến nhất mà bạn dùng để hiển thị một giá trị tính toán, nhưng bạn có thể sử dụng bất cứ **Control** nào đã có tính chất **Controlsource**.
- Trên **Form** hoặc **Report**, hãy nhấp lên bất cứ nơi nào mà bạn muốn đặt **Control**.

*Thực hiện các thao tác sau đây:*

- Nếu **Control** là một **Text** box, bạn có thể gõ nhập biểu thức trực tiếp trong **Text** box.
- Đối với bất cứ **Control** nào, bạn phải chắc chắn rằng **Control** đó đã được chọn, hãy nhấp **Property** trên thanh công cụ để hiển thị trang tính chất của **Control**, rồi gõ nhập một biểu thức vào hộp tính chất **Controlsource**. Để sử dụng **Expression Builder** để tạo ra biểu thức trong **Microsoft Access Database (.mdb)**, hãy nhấp nút **Build** nằm kề hộp tính chất **Controlsource**.

## X. Tính một tổng dành cho một Record trong một Form

- Mở **Form** trong **Design View**
- Nhấp công cụ **Textbox** trong **Toolbox**.
- Trong mục chi tiết của **Form**, hãy nhấp nơi mà bạn muốn đặt **Textbox**.
- Thực hiện một trong các nội dung sau đây:
- Đặt điểm chèn trong **Text** box và gõ nhập một biểu thức nhằm tính một tổng.
- Chắc chắn **Text** box được chọn, nhấp **Properties** trên thanh công cụ để mở trang tính chất của nó, và gõ nhập biểu thức vào hộp tính chất **Controlsource**. Trong **Microsoft Access Database (.mdb)**, để sử dụng **Expression Builder** nhằm tạo ra biểu thức, hãy nhấp nút **Build** nằm kề' hộp tính chất **Controlsource**.



## XI. Tính một tổng trong một subForm và hiển thị nó trên một Form

- Bạn có thể sử dụng một **Form** có một **subForm** để tính một tổng dành cho một nhóm các **Record**.
- **Main Form** và **subForm** đều dựa trên các **Table** với quan hệ một-đối-nhiều. **Main Form** minh họa các **Record** từ phía “**one**” của quan hệ; **subForm** thì minh họa các **Record** từ phía “**many**” của quan hệ.
- **Ví dụ**, hình sau đây minh họa một **Form** Orders có chứa một **subForm**.
- Để tính tổng con order, bạn bổ sung hộp **Text** vào **Form Footer** của **subForm** và chỉ định một biểu thức trong hộp tính chất **Controlsource** dành cho **Text** box. Biểu thức này sử dụng hàm Sum.
- **Ví dụ**, biểu thức sau đây tính tổng lượng dành cho tất cả các sản phẩm trong Orders **Form**:  

$$= \text{Sum}([\text{Quantity}]*[\text{UnitPrice}])$$
- Để hiển thị tổng con order trên main **Form**, hãy tạo một **Textbox** trong main **Form** nhằm sử dụng một biểu thức tham chiếu đến **Control** trên **subForm** có chứa giá trị tính toán.
- **Ví dụ**, bạn có thể chỉ định biểu thức sau đây lòng hộp tính chất **Controlsource** của hộp **Text** trên main **Form**

## XII. Ngưng tác dụng cha Form View hoặc DataSheet View dành cho một Form .

- Mở một **Form** trong **Design View**.
- Nhấp đúp **Form Selector** để mở trang tính chất của **Form**.
- Trong hộp tính chất **Viewsallowed**, hãy nhấp **Form** để hủy bỏ hoạt động của **DataSheet View**, hoặc nhấp **DataSheet** để hủy bỏ hoạt động của **Form View**.
- **Lưu ý**: Bạn có thể sử dụng tính chất này cùng với tính chất **DefaultView** để xác định cách mà những người sử dụng sẽ xem một **Form** và họ có xem nó trong các khung xem khác hay không.

## XIV. Tạo các Macro cho Form trong một Macro group

- Trong cửa sổ **Database**, hãy nhấp **Macros**, rồi nhấp nút **New** trên thanh công cụ cửa sổ **Database**.
- Đối với thủ tục này, bạn sẽ tạo ra bốn **Macro** trong một nhóm **Macro**.
- Tạo một **Macro** nhằm mở một unbound **Form**. Hãy bắt đầu bằng cách nhấp **Macro Names** để hiển thị cột **Macro Name**. Gõ nhập tên **Macro**,



**Chương 8:**

# Làm việc với Report

**I. Mở một Report**

- Trong **Database Windows**, nhấp **Report** trong **Objects**.
- Nhấp **Report** mà bạn muốn mở
- Trong thanh công cụ cửa sổ **Database**, nhấp nút Design để mở **Report** trong **Design View**, hoặc nhấp nút **PreView** để mở **Report** trong **Print PreView**.

**II. Tạo một Report****1. Các loại Report**

- Single column (báo biểu một cột): biểu diễn các trường trong một record trên một cột từ trên xuống.
- Groups/totals (báo biểu kết nhóm/tổng cộng): dùng các biểu báo chi tiết cần lấy kết số cho một chỉ tiêu hay số liệu cho từng nhóm đối tượng dạng Table.
- Summary (biểu báo tóm lược): tương tự Group/Total
- Mailing (biểu báo thư tín): in ấn nhãn, bì thư.
- Tabular (báo biểu dạng bảng): tương tự bảng tính Excel
- MS Word mailing Merge (ghép trộn thư tín với MS Word)
- Auto Report (báo biểu tự động): tương tự single column cho một Table.

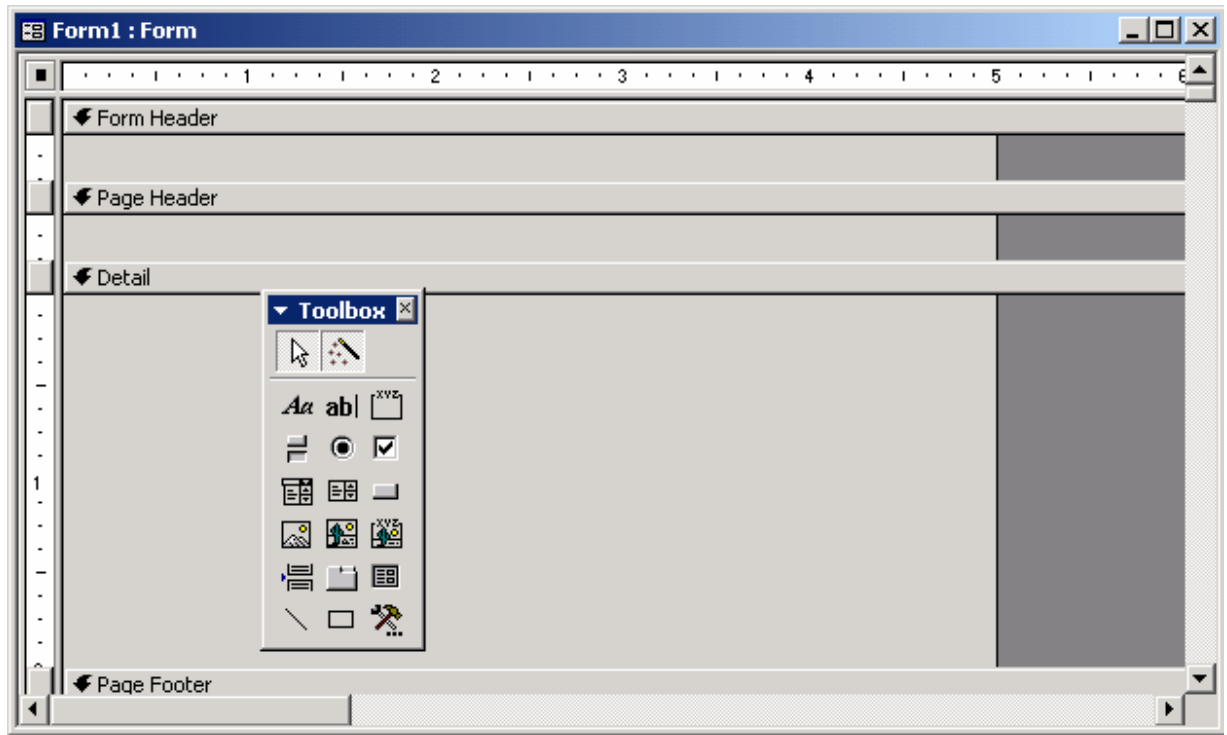
**2. Tạo mới Report:**

Chọn gờ Report trong Database, chọn New, tiếp tục chọn Design View và OK sẽ cho màn hình thiết kế Report: trong phần Design View là một màn hình trống của một Report (chỉ có phần details), người dùng phải tự thiết kế lấy với thanh công cụ Toolbox.

Chú ý: Nếu cần nhiều công cụ ta nhấp nút more (hình ...) trên thanh công cụ Toolbox.

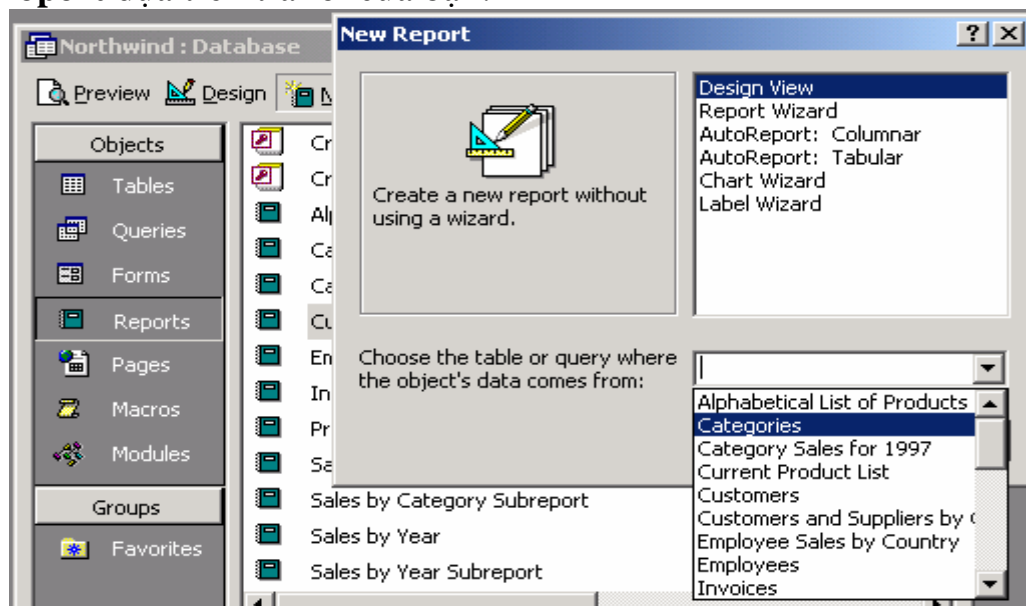
*Màn hình thiết kế Report có các phần chính sau*

- Report Header: đầu để báo biểu, thường ghi tiêu đề và ngày hiện tại.
- Page Header: tiêu đề đầu trang thường chứa tên của các cột (trường)
- Details: chi tiết của nội dung của trường khi chạy Report.
- Page Footer: chân trang, thường ghi số trang trong một Report
- Report Footer: Chân báo biểu, thường có số lũy kế cho trường đến.



### 3. Tạo một Report bằng cách sử dụng AutoReport.

- Với **AutoReport**, bạn chọn nguồn **Record** hoặc columnar hoặc **Tabular** layout; **AutoReport** sử dụng tất cả các trường từ nguồn **Report** và áp dụng **autoFormat** mà bạn đã dùng cho **Report** này.
- Tạo một **Report** với **Wizard**. **Wizard** hỏi bạn các câu hỏi chi tiết về nguồn **Report**, trường, thiết kế và dạng mà bạn muốn rồi tạo một **Report** dựa trên trả lời của bạn.



### III. Tạo một Form hoặc Report từ các Record được lọc

- Với các **Record** được lọc đang hiển thị, hãy nhấp mũi tên nằm kế nút **New Object** trên thanh công cụ.
- Để sử dụng **autoReport** hoặc **autoReport Wizard**, nhấp **autoForm** hoặc **autoReport**
- Để chỉ định kiểu **Form** hoặc **Report** mà bạn muốn tạo, hãy nhấp **Form** hoặc **Report**, rồi thực hiện mục chọn lựa của mình.
- *Đối tượng mới sử dụng **Table** hoặc **Query** qua đó bộ lọc được tạo ra làm nguồn **Record** của nó và nó thừa kế bộ lọc.*
- Hoàn tất thiết kế hoặc thực hiện bất cứ thay đổi nào khác mà bạn muốn cho **Form** hoặc **Report**, rồi lưu nó.

### IV. Hoán chuyển giữa các khung xem của một Report

- **Report** có ba khung xem: **Design View**, **Print PreView**, và **Layout PreView**.
- Bạn sử dụng **Design View** để tạo một **Report** hoặc thay đổi cấu trúc của một **Report**. Bạn sử dụng **Print PreView** để xem dữ liệu của một **Report** khi nó xuất hiện trên mỗi một trang. Bạn sử dụng **Layout Preview** để xem thiết kế của **Report** mà chỉ có chứa mẫu dữ liệu trong **Report** đó.
- Mở **Report** trong bất cứ khung xem nào.
- Nhấp nút **View** trên thanh công cụ để thay đổi khung xem do hình đồ họa chỉ định.

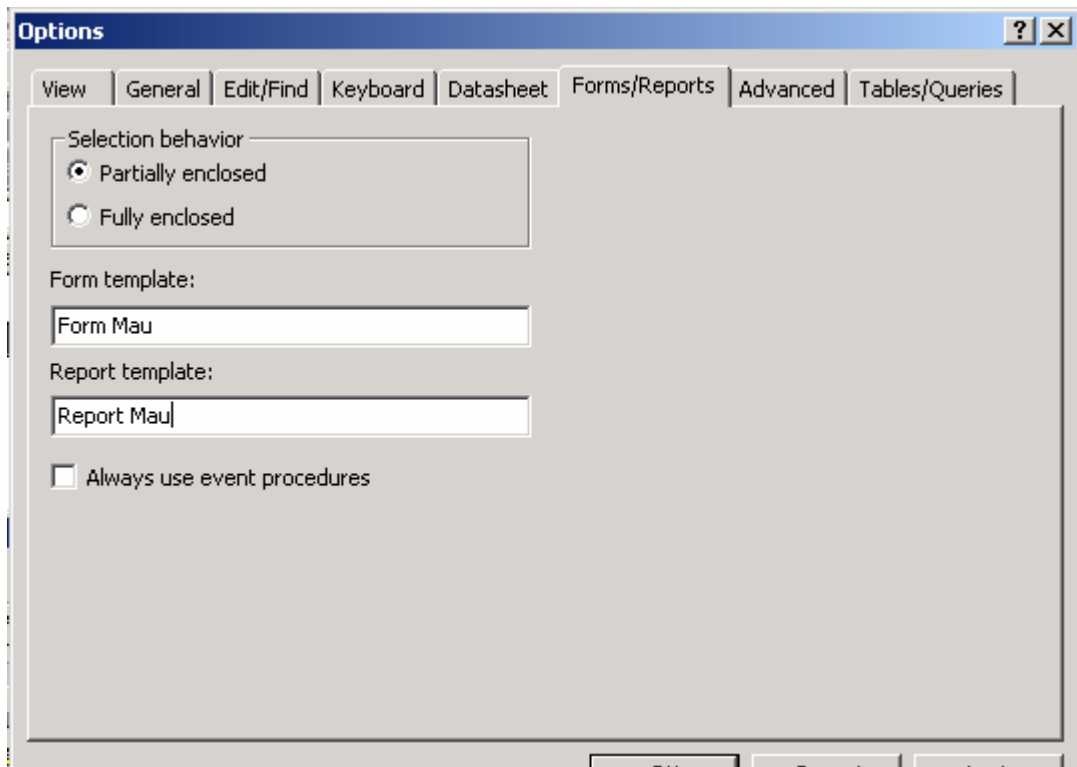
### V. Các bảng mẫu Form và Report

- Lúc bạn tạo **Form** hoặc **Report** mà không sử dụng **Wizard**, Microsoft **Access** sử dụng bảng mẫu để xác định các đặc trưng của **Form** hoặc **Report** này.
- Bảng mẫu xác định loại mục nào mà một **Form** hoặc **Report** sẽ có và xác định kích thước (kiểu) của mỗi mục đó. Bảng mẫu cũng có chứa tất cả các cài đặt tính chất mặc định dành cho **Form** hoặc **Report** và các section và **Control** của nó.
- Bảng mẫu mặc định dành cho các **Form** và **Report** gọi là **Normal** (chuẩn). Tuy nhiên, bạn cũng có thể sử dụng bất cứ **Form** hoặc **Report** nào đang có để làm mẫu. Bạn cũng có thể tạo một **Form** hoặc **Report** bằng cách chỉ sử dụng lại một mẫu.

**Lưu ý:**

- Việc thay đổi bảng mẫu không làm thay đổi bất cứ **Form** hoặc **Report** đang có.
- Một bảng mẫu không tạo ra các **Control** trên bảng **Form** hoặc **Report** mới.
- **Microsoft Access** lưu các cài đặt dành cho các tùy chọn **Form** Template và **Report** Template trong **File** thông tin nhóm công việc của **Microsoft Access**, chứ không phải trong dữ liệu **Microsoft Access** (**File** .mdb) hoặc đề án **Microsoft Access** (**File** .adp). Lúc bạn thay đổi một cài đặt tùy chọn, thay đổi sẽ được dụng cho bất cứ **cơ sở dữ liệu** hoặc đề án **Access** nào mà bạn mở hoặc tạo. Để xem tên của bảng mẫu hiện đang được sử dụng dành cho các **Form** hoặc **Report** mới, nhấp **Options** trên men **Tools**, rồi nhấp lên nhãn **Forms/Reports**.
- Để sử dụng các bảng mẫu trong **cơ sở dữ liệu** hoặc đề án **Access**, hãy sao chép hoặc xuất các bảng mẫu sang chúng.
- Nếu bảng mẫu của bạn không nằm trong **cơ sở dữ liệu** hoặc đề án **Access**, **Microsoft Access** sử dụng bảng mẫu Normal cho bất cứ **Form** hoặc **Report** mới nào. Tuy nhiên, tên của các bảng mẫu xuất hiện trong các tùy chọn **Form** Template và **Report** Template trong mỗi một **cơ sở dữ liệu** hoặc đề án **Access** trong hệ thống **cơ sở dữ liệu** của bạn, thậm chí cả khi các bảng mẫu không nằm trong một **cơ sở dữ liệu** hoặc đề án **Access**

**VI. Chỉ định bảng mẫu mới dành cho các Form và Report**



- Trên **Menu Tools** nhấn **Option**
- Nhấp nhấn **Forms/Reports**
- Gõ nhập tên bảng mẫu mới trong hộp **Form Template** hoặc **Report Template**.

#### Lưu ý:

Bạn có thể sử dụng một **Form** hoặc **Report** đang có làm bảng mẫu, hoặc bạn có thể tạo một **Form** hoặc **Report** đặc biệt để sử dụng làm một bảng mẫu.

- Nhấp **OK**.

### VII. Chuyển phiên bản đã được lưu của một Form hoặc Report

- Bạn có thể hủy bỏ bất cứ thay đổi nào kể cả các thay đổi cho các mã **Microsoft Visual Basic** mà bạn tạo ra trong **Design View** cho một **Form** hoặc **Report** và chuyển chúng sang phiên bản lưu sau cùng của một **Form** hoặc **Report** đó.
- Để chuyển sang phiên bản được lưu sau cùng hãy thực hiện
- Trên **Menu File** nhấp **Revert**.

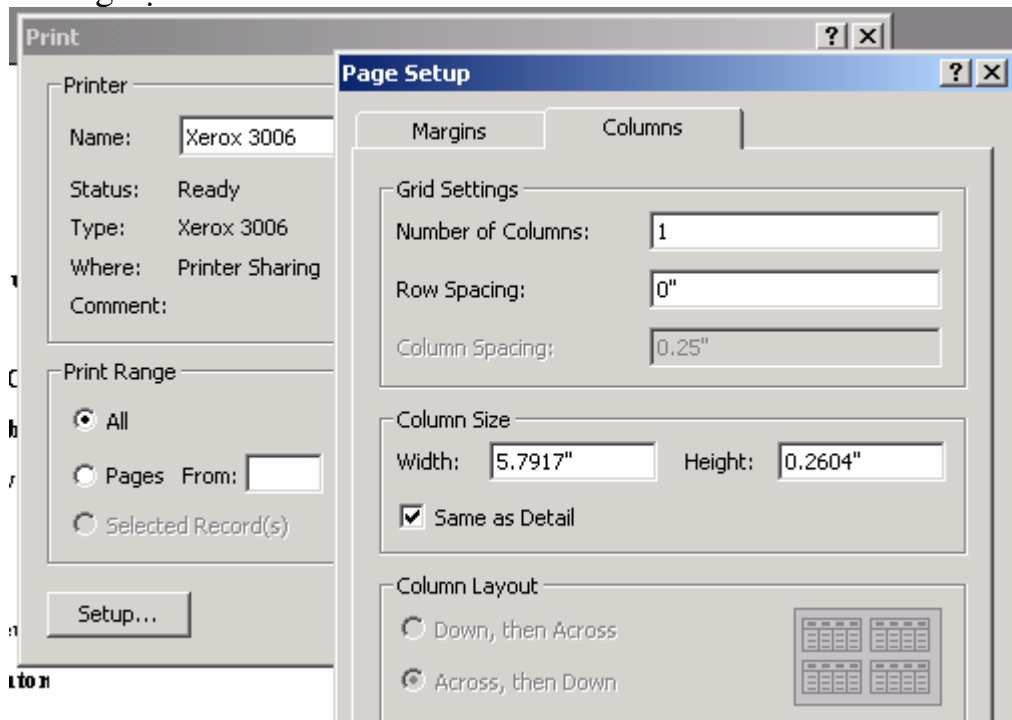
#### Lưu ý:

Lệnh này không có sẵn trừ khi bạn mở một **Form** hoặc **Report** trong **Design View** và thực hiện các thay đổi cho nó. Lệnh **Revert** cũng không có sẵn trong **Design View** hoặc trang truy cập dữ liệu.

## VIII. In một Report

- Trước khi in một **Report** lần đầu tiên, bạn phải kiểm tra lại lề, hướng trang, và các tùy chọn xác lập trang khác.
- Chọn **Report** trong cửa sổ **Database**, hoặc mở **Report** trong **Design View**, **Print PreView**, hoặc **Layout PreView**.
- Trên men **File**, nhấp **Print**.
- Nhập các cài đặt mà bạn muốn vào **Print**
- Trong phần **Printer** hãy chỉ định một máy in.
- Trong phần **Print range**, hãy chỉ định một trang hoặc một loạt các trang.
- Trong phần **Copies**, hãy chỉ định số bản in.
- Nhấp **OK**.

**Tip:** Để in một **Report** mà không cần đến hộp thoại, hãy nhấp **Print** trên thanh công cụ.



## IX. Xem trước một Table, Form, hoặc Report theo các mức phóng đại khác nhau

- Trên thanh công cụ **Print PreView**, nhấp mũi tên nằm kế **Zoom** box, rồi nhấp tỉ lệ phần trăm độ phóng đại, hoặc gõ nhập tỉ lệ phần trăm độ phóng đại mà bạn muốn.

**Lưu ý:**



- Lúc chọn **Fit**, **Microsoft Access** chỉnh sửa độ phóng đại của trang cho phù hợp với kích cỡ cửa sổ.
- Bạn có thể hoán chuyển giữa các độ phóng đại mà bạn chọn và **Fit** bằng cách đặt con trỏ trên **Table**, **Form**, hoặc **Report**, và nhấp lúc con trỏ chuyển thành một kính lúp (Magnifying Glass).
- Bạn có thể sử dụng nút định hướng (Navigation) ở đáy của cửa sổ **Print PreView** để di chuyển giữa các trang. Bạn có thể di chuyển bên trong trang hiện có bằng cách sử dụng các thanh cuộn, hoặc, nếu bạn có một chuột thông minh (**Microsoft Intelli-mouse**), hãy sử dụng nút bánh xe.

## X. Xem trước hai hoặc nhiều trang cùng một lúc

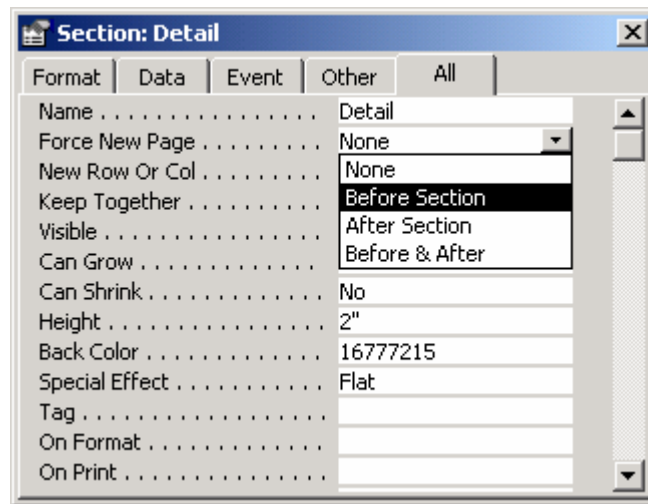
- Xem trước hai trang
- Mở **Table**, **Query**, **Form**, **Report**, hoặc module trong **Print PreView**.
- Nhấp **Two Pages** trên thanh công cụ
- để xem nhiều trang
- Mở **Table**, **Query**, **Form**, **Report**, hoặc module trong **Print PreView**.
- Nhấp **Multiple Pages** trên thanh công cụ
- Rê để chọn số trang cần hiển thị.
- để xem một trang
- Mở **Table**, **Query**, **Form**, **Report**, hoặc module trong **Print PreView**.
- Nhấp **One Pages** trên thanh công cụ

Lưu ý;

Bạn có thể sử dụng các nút điều khiển ở đáy của cửa sổ **Print PreView** để di chuyển giữa các trang. Bạn có thể di chuyển bên trong trang đang có bằng thanh cuộn, hoặc, nếu bạn có một chuột thông minh, hãy sử dụng nút bánh xe.

## XI. In mỗi Record, Group hoặc Section

- Mở **Form** trong **Design View** hoặc **Report** trong **Design View**.
- Nhấp đúp section **Selector** để mở trang tính chất.
- Cài đặt tính chất **Force New Page** của mục chọn sang một trong các cài đặt sau đây:
  - *None (mặc định):* In mục chọn trên trang hiện hành.
  - *Before section:* in mục chọn ở phần trên của mỗi trang mới.
  - *After section:* in mục chọn ở phần trên của một trang mới.
  - *Before & after:* in mục chọn ở phần trên của một trang mới, rồi in các phần sau đó ở đỉnh của trang mới.



### Lưu ý:

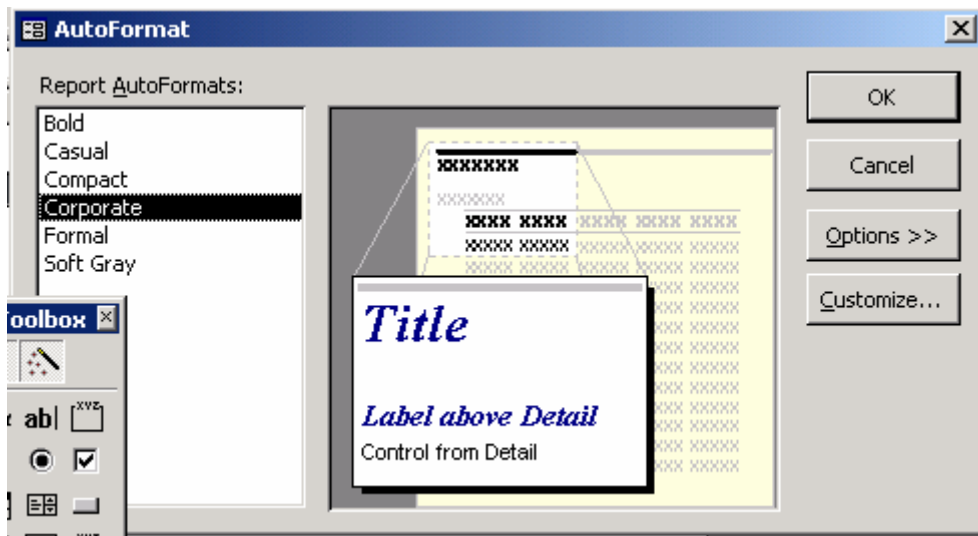
- Tính chất **ForcenewPage** có sẵn trong tất cả các mục chọn ngoại trừ **Header** và **Footer** của trang đó.
- Để in mỗi **Record** trên trang riêng biệt, hãy cài đặt tính chất **ForcenewPage** của mục chi tiết sang **After Section**. Để in mỗi nhóm trên một trang riêng biệt, hãy cài đặt tính chất **ForcenewPage** của tiêu đề nhóm sang **Before Section** hoặc tính chất **ForcenewPage** của tiêu đề nhóm sang **After Section**.

## XII. Định dạng một Form, Report, hoặc Control

*Mở Form trong Design View hoặc Report trong Design View. Thực hiện một trong các thao tác sau:*

- Nếu bạn muốn định dạng toàn bộ **Form** hoặc **Report**, hãy nhấp **Form Selector** hoặc **Report Selector**.
- Nếu bạn muốn định dạng một mục riêng biệt, hãy nhấp lên **section Selector**.
- Nếu bạn muốn định dạng một hoặc nhiều **Control**, hãy chọn các **Control**.
- Nhấp **AutoFormat** trên thanh công cụ
- Nhấp một trong các dạng trong danh sách.
- Nếu bạn chỉ định các thuộc tính nào được áp dụng (font, màu, hoặc đường biên), nhấp **Options**.

## XIII. Tùy nghi sử dụng chế độ AutoFormat dành cho Form hoặc Report



- Mở một **Form** hoặc **Report** đang có trong **Design View**.
- Trên **Menu Format**, nhấn auto**F**ormat.
- Trong hộp thoại auto**F**ormat, nhấn một chế độ định dạng trong danh sách.
- Nhấn nút **O**ption, phải chắc chắn các thuộc tính mà bạn muốn chỉnh sửa đã được chọn.
- Nhấn nút **C**ustomize, rồi nhấn lên một tùy chọn tùy ý. Bạn có thể thực hiện một trong các thao tác sau:
  - *Tạo một chế độ định dạng tự động dựa trên **Form** hoặc **Report**.*
  - *Cập nhật chế độ định dạng tự động dựa trên **Form** hoặc **Report**.*
  - *Xóa bỏ chế độ định dạng tự động khỏi danh sách*
- Nhấn **O**K hai lần để đóng các hộp thoại.

#### XIV. Bổ sung một hình ảnh nền vào- một Report

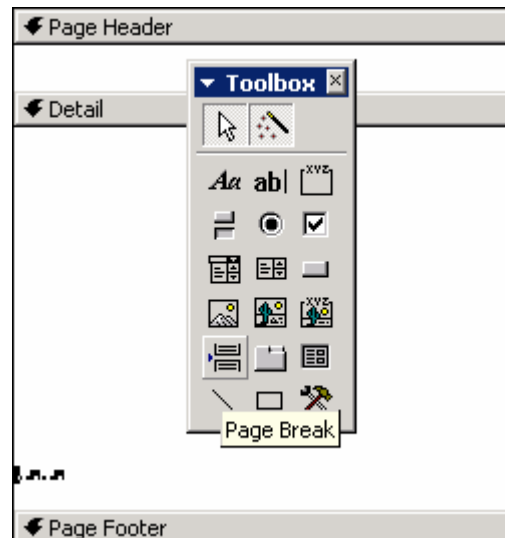
- Bằng cách bổ sung một hình ảnh nền vào một **Report**, bạn có thể mô phỏng một vân tô. Một hình ảnh nền trong một **Report** áp dụng cho toàn bộ trang.
- Mở **Report** trong **Design View**.
- Nhấn đúp **R**eport selector để mở trang tính chất.
- Cài đặt ảnh chất Picture sang một **F**ile .bmp, .ico, .wmf, dib, hoặc .emf. Nếu bạn đã cài đặt bộ lọc hình đó họa cho các trình ứng dụng, bạn hãy sử dụng bất cứ **F**ile nào được bộ lọc này hỗ trợ. Nếu bạn chẳng chắc chắn đường dẫn hoặc tên **F**ile, nhấn nút **B**uild.
- Trong hộp tính chất **P**icture**T**ype, hãy chỉ định xem bạn muốn hình ảnh được nhúng hoặc được liên kết Lúc bạn nhúng hình ảnh, nó được lưu trong **F**ile cơ sở dữ liệu. Nếu bạn nhúng hình ảnh như vậy vào **F**orm

hoặc **Report**, nó được bổ sung vào file **cơ sở dữ liệu** một lần nữa. Lúc bạn liên kết một hình ảnh, bạn không cần lưu nó trong **File cơ sở dữ liệu** nhưng bạn phải giữ một bản sao của nó trên đĩa cứng. Nếu bạn muốn sử dụng hữu hiệu cài đặt của mình, hãy chỉ định cài đặt **Linked**.

- Bạn có thể xác định tỷ lệ của hình ảnh bằng cách cài đặt tính chất **PictureSizeMode**. Tính chất này có 3 cài đặt như sau:
- **Clip**: hiển thị hình ảnh theo kích thước thật. Nếu hình ảnh lớn hơn diện tích bên trong các lề, hình sẽ bị cắt phần dư.
- **Stretch**: định kích cỡ cho phù hợp với diện tích bên trong lề. Cài đặt này có thể làm biến dạng hình ảnh.
- **Zoom**: định kích thước hình ảnh cho phù hợp hoặc chiều cao hoặc chiều rộng của diện tích bên trong lề. Cài đặt này sẽ không cắt tỉa hoặc làm biến dạng tỉ lệ của nó.

## XV. Bổ Sung một dấu ngắt trang vào Report

- Trong các **Report**, bạn sử dụng một dấu ngắt trang (**Page break Control**) để đánh dấu nơi bạn muốn bắt đầu một trang mới bên trong một mục.
- **Ví dụ**, nếu bạn muốn trang tiêu đề của **Report** và nội dung giới thiệu được in trên các trang riêng, hãy đặt một dấu ngắt trang trong **Report Header** sau các **Control** mà bạn muốn xuất hiện trên trang tiêu đề và trước các **Control** dành cho trang thứ hai.
- Mở **Report** trong **Design View**.
- Nhấp công cụ **Page Break** trong **Toolbox**.
- nhấp **Report** nơi mà bạn muốn đặt dấu ngắt trang. Cài đặt dấu ngắt trang bên trên hoặc dưới 1 **Control** để tránh phải tách dữ liệu trong **Control**.
- **Microsoft Access** đánh dấu ngắt trang bằng một vạch chấm ngắn ở cạnh bên trái của **Report**.
- Nếu bạn muốn mỗi một nhóm hoặc **Record** trong **Report** bắt đầu ở một trang mới. cài đặt tính chất **ForcenewPage** của group **Header**, group **Footer**,
- hoặc mục chi tiết.

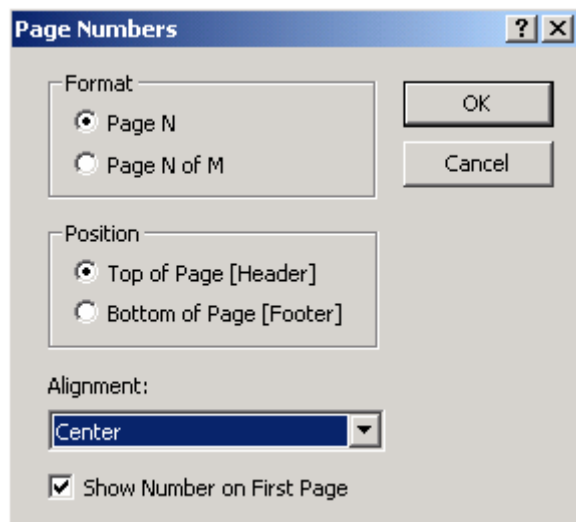


### Lưu ý:

- **Report Catalog** trong **cơ sở dữ liệu** mẫu Northwind là một **Ví dụ** về sử dụng dấu ngắt trang trong **Report Header**. Để xem **Report** này, bạn hãy
- mở **cơ sở dữ liệu** Northwind trong **Folder Samples** của **Folder Office**, rồi mở **Report Catalog** trong **Design View**.

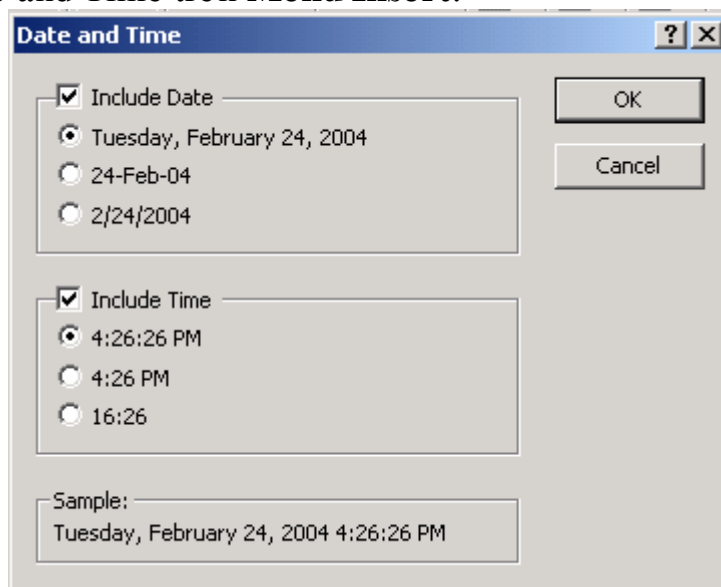
### XVI. Bổ sung số trang vào một Form hoặc Report

- Mở **Form** hoặc **Report** trong **Design View**.
- Nhấp **Page Numbers** trên men Insert.
- trong hộp thoại **Pages Numbers**: chọn dạng, vị trí và chế độ giống hàng dành cho số trang đối với tùy chọn giống hàng thì các tùy chọn sau đây là có sẵn:
  - Left: bổ sung hộp **Text** bên lề trái
  - Center: bổ sung hộp **Text** chính giữa lề phải và lề trái
  - Ritgh: bổ sung hộp **Text** bên lề phải.
  - Inside: bổ sung hộp **Text** bên các lề trái và phải. Số trang lẻ thì được in bên trái, và các trang chẵn thì được in bên phải.
  - OutSide: bổ sung hộp **Text** bên các lề trái và phải. Số trang chẵn thì được in bên trái, và các trang lẻ thì được in bên phải.
- Để minh họa một con số trên trang đầu tiên, chọn hộp kiểm **Show Number on first Page**.
- **Microsoft Access** sử dụng một biểu thức để tạo các số trang.



## XVII. Bổ sung ngày tháng và thời gian hiện hành vào Form hoặc Report

- Mở **Form** hoặc **Report** trong **Design View**
- Nhấp **Date and Time** trên **Menu Insert**.



- Đưa vào một ngày tháng, chọn hộp kiểm include date, rồi nhấp lên mục dạng ngày tháng.
- Để đưa vào thời gian, chọn hộp kiểm include time, rồi nhấp lên mục dạng thời gian.
- **Microsoft Access** bổ sung một **Textbox** vào **Form** và cài đặt tính chất **Control source** của nó sang một biểu thức. Nếu có một **Form** hoặc một **Report Header**, **Microsoft Access** bổ sung **Textbox** cho mục đó. Còn nếu không nó sẽ bổ sung **Textbox** cho mục chi tiết.

## XVIII. Bỏ qua Header hoặc Footer của trang trên các trang với Report Header hoặc Footer

- Mở **Report** trong **Design View**
- Nhấp đúp **Report Selector** để mở trang tính chất.
- Cài đặt tính chất **PageHeader** hoặc **PageFooter** sang một trong các cài đặt sau:
  - **All Pages** (mặc định): in **Header** hoặc **Footer** của trang trên tất cả các trang.
  - **Not with Rpt Hdr**: không in **Header** hoặc **Footer** của trang trên trang đã có **Report Header**.
  - **Not with Rpt Ftr**: không in **Header** hoặc **Footer** của trang trên trang đã có **Report Footer**.
  - **Not with Rpt Hdr/Ftr**: không in **Header** hoặc **Footer** của trang trên trang đã có **Report Header** hoặc **Footer**.

### **XIX. Thay đổi kích cỡ của Header, Footer hoặc mục chọn khác trên Form hoặc Report**

- Bạn có thể tăng hoặc giảm chiều cao của các mục chọn **Form** và **Report** riêng biệt. Tuy nhiên một **Form** hoặc **Report** chỉ có một chiều rộng. Lúc bạn thay đổi chiều rộng của mục chọn, bạn thay đổi luôn chiều rộng của **Form** hoặc **Report** đó.
- Thay đổi chiều cao, chiều rộng
- Mở **Form** hoặc **Report** trong **Design View**
- Thay đổi chiều cao hoặc rộng bằng cách đặt trỏ ở đáy cạnh hoặc bên phải của mục chọn.
- Rê con trỏ lên hoặc xuống nhằm thay đổi chiều cao của mục chọn. Rê con trỏ về phía trái hoặc phải để thay đổi chiều rộng.
- Để thay đổi cả chiều cao lẫn chiều rộng, hãy đặt trỏ ở góc phải phía dưới của mục chọn và rê nó theo dạng đường chéo.

# Mục lục

## Chương 1 . Xử lý các File trong Access 2000

- I. Khảo sát việc tìm kiếm các File
- II. Tìm các File
- III. Phân loại một danh sách các File
- IV. Nối kết vào một ổ mạng
- V. Tìm kiếm các File
- VI. Hủy bỏ một mục tìm kiếm đã được lưu
- VII. Tìm kiếm các File bằng cách sử dụng các tính chất File
- VIII. Tìm kiếm các File bằng cách sử dụng các tiêu chuẩn tìm kiếm đã được lưu
- IX. Sử dụng các tính chất File để định vị trí các File
- X. Sao chép một File
- XI. Xóa bỏ một File
- XII. Đặt lại tên một File
- XIII. Di chuyển một File
- XIV. Xem tất cả các kiểu File và phần mở rộng tên File
- XV. Nơi tối ưu để lưu giữ các File

## Chương 2. Thiết kế cơ sở dữ liệu

- I. Khảo sát việc thiết kế một cơ sở dữ liệu
- II. Các bước để thiết kế một cơ sở dữ liệu
- III. Tạo một cơ sở dữ liệu mới
- IV. Khảo sát việc tạo một trình ứng dụng
- V. Hiện thị hoặc thay đổi các tính chất tổng quát của một đối tượng cơ sở dữ liệu
- VI. Mở một cơ sở dữ liệu Microsoft" Access
- VII. Tự động tạo một bước tắt vào một cơ sở dữ liệu Access hoặc đề án Access trong Folder Favorites
- VIII. Bổ sung cơ sở dữ liệu Access hiện hành vào Folder Favorites
- IX. Mở một cơ sở dữ liệu Access trên mạng
- X. Thay đổi diện mạo của các đối tượng trong cửa sổ Database
- XI. Hiện thị hoặc che giấu các biểu tượng tắt để tạo các đối tượng cơ sở dữ liệu mới
- XII. Xóa bỏ một đối tượng cơ sở dữ liệu
- XIII. Lưu một đối tượng cơ sở dữ liệu



XIV. Xem, in, xuất, hoặc lưu các đặc trưng thiết kế các đối tượng cơ sở dữ liệu

### Chương 3. Làm việc với Table

I. Table : nội dung và cách làm việc

II. Tạo Tables

III. Tùy chọn Tablez

IV. Mở một Table

V. Hoán chuyển giữa các khung xem của một Table

VI. Tối ưu hóa cách thực hiện chức năng của Table nói chung

VII. Xác định các quy tắc có hiệu lực để điều khiển lúc nào thì một Record có thể được lưu

VIII. Khảo sát Query của Table Analyzer Wizard

IX. Bổ sung một trường vào một Table trong Design View

X. Sao chép một định nghĩa của trường trong Table Design View

XI. Tạo một trường dành cho Text hoặc meo

XII. Tạo một trường dành cho các số hoặc tiền tệ

XIII. Tạo một trường dành cho dữ liệu Yes/no, True/False, và On/off

XIV. Tạo một trường dành cho ngày tháng hoặc thời gian

XV. Tạo một trường để tự động tạo ra các số AutoNum

XVI. Hủy bỏ một trường khỏi một Table

XVII Thay đổi kích thước của trường dành cho các trường Text và Number

XVIII. Tạo một Input Mask (mặt nạ nhập)

### Chương 4. Làm việc với Form

I. Các cách để làm việc với dữ liệu trong một Form

II. Các cách để làm việc với dữ liệu trong DataSheet của Query

III. Bổ sung dữ liệu mới vào DataSheet hoặc Form View

IV. Lưu một Record trong DataSheet hoặc Form View

V. Xóa bỏ một Record trong DataSheet hoặc Form View

VI. Sao chép hoặc di chuyển một trường Control, Text, Macro Action

VII Hiện thị tất cả dữ liệu đang có trong Page View

VIII. Nhập một HyperLink (siêu phù hợp ) bằng cách dán một HyperLink hoặc địa chỉ HyperLink

### Chương 5. Làm việc với các Record

I. Bổ sung một Record mới trong Page View

II. Lưu một Record trong Page View

- III. Xóa bỏ một Record trong Page View
- IV. Sao chép hoặc di chuyển dữ liệu từ một trường sang trường khác
- V. Sao chép hoặc di chuyển dữ liệu từ nhiều trường vào DataSheet View
- VI. Sao chép hoặc di chuyển các Record vào một trang dữ liệu
- VIII. Sao chép hoặc di chuyển các Record sang một Form
- IX. Sao chép hoặc di chuyển các Record từ Microsoft Access đến trình ứng dụng khác
- X. Sao chép hoặc di chuyển các Record từ một trình ứng dụng khác vào Microsoff Acces
- XI. Thu thập và dán nhiều hạng mục
- XII. Khảo sát về việc thu thập và dự án nhiều hạng mục
- XIII. Xóa tất cả các hạng m~c khỏi Office Clipboard
- XIV. Xem nội dung của Office Clipboard
- XV. Hiển thị dữ liệu từ một trường trong một Table hoặc một Query
- XVI. Kiểm tra chính tả dữ liệu trong DataSheet View
- XVII. Kiểm tra chính tả cửa dữ liệu trong Form View
- XVIII. Tự động chỉnh sửa lỗi lúc nhập dữ liệu
- XIX. Tự động viết hoa các tên ngày lúc nhập dữ liệu
- XX. Chỉnh sửa hai mẫu tự được viết hoa trong một hàng lúc nhập dữ liệu

## Chương 6. Các tác vụ căn bản về Form

- I. Tạo ra một Form
- II. Tạo một Form hoặc Report từ các Record được lọc
- III. Mở một Form
- IV. Các khung xem của một Form
- V. Các Form mẫu trong Form Wizard
- VI. Chọn một Form hoặc Report
- VII. Chọn một mục chọn trên một Form hoặc Report
- VIII. Hiển thị hoặc che giấu lưới trên một Form, Report hoặc tranh truy cập dữ liệu
- IX. Thay đổi khoảng cách giữa các chấm trên lưới Form hoặc Report
- X. Cài đặt các tính chất dành cho một Form, Report Action hoặc Control
- XI. Cài đặt kích thước của một Form lúc lần đầu tiên mở nó
- XII. Chuyển đổi sang phiên bản được lưu của một
- XIII. xem trước một Form
- XIV. In các Record được chọn trong một Form hoặc DataSheet

- XV. Cho phép thay đổi đối với các tính chất của Form trong Form và DataSheet View
- XVI. Bổ sung hoặc xóa bỏ một Header
- XVII. Che giấu các dòng chia mục trong một Form (section divider lines)
- XVIII. kích cỡ một Form cho phù hợp với nội dung của nó
- XIX. Định tâm một Form trong cửa sổ trình ứng dụng lúc lần đầu tiên nó được mở
- XX. Thay đổi đơn vị đo của thước trong một Form hoặc Report
- XXI. Thay đổi thứ tự tay trong một Form
- XXII. Xóa bỏ Menu tắt khỏi một Form
- XXIII. Xóa bỏ Record Selector khỏi một Form
- XXIV. Xóa bỏ các nút điều khiển
- XXV. Hủy chọn nút Close trên một Form

## Chương 7. SubForm - Nội dung và cách hoạt động

- I. Một subForm là một Form nằm bên trong một Form.
- II. Các kiểu subForm
- III. Microsoft Access liên kết các main Form và subForm
- IV. Liên kết một main Form và một subForm
- V. Chỉ định khung xem (View) dành cho một subForm
- VI. Di chuyển giữa một main Form và một subForm trong Form View
- VII. Hoán Chuyển giữa DataSheet View và Form View trong một subForm
- VIII. Tạo ra một subForm trông giống như một DataSheet để hiển thị một Header và Footer
- IX. Tạo một Control tính toán trên một Form hoặc Report
- X. Tính một tổng dành cho một Record trong một Form
- XI. Tính một tổng trong một subForm và hiển thị nó trên một Form
- XII. Ngưng tác dụng của Form View hoặc DataSheet View dành cho. một Form
- XIII. Tạo một Form ở dạng chỉ đọc
- XIV. Tạo các Macro c'lo Form trong một Macro group

## Chương 8. Làm việc với Report

- I. Mở một Report
- II. Tạo một Report
- III. Tạo một Form hoặc Report từ các Record được lọc
- IV. Hoán chuyển giữa các khung xem của một Report
- V. Các bảng mẫu Form và Report

- VI. Chỉ định bảng mẫu mới dành cho các Form và Report
- VII. Chuyển phiên bản đã được lưu của một Form hoặc Report
- VIII. In một Report
- IX. Xem trước một Table, Form, hoặc Report theo các mức phóng đại khác nhau
- X. Xem trước hai hoặc nhiều trang cùng một lúc
- XI. In mỗi Record, group hoặc section
- XII. Định dạng một Form, Report, hoặc Control
- XIII. Tùy nghi sử dụng chế độ AutoFormat dành cho Form hoặc Report
- XIV. Bổ sung một hình ảnh nền vào một Report
- XV. Bổ sung một dấu ngắt trang vào Report
- XVI. Bổ sung số trang vào một Form hoặc Report
- XVII. Bổ sung ngày tháng và thời gian hiện hành vào Form hoặc Report
- XVIII. Bỏ qua Header hoặc Footer của trang trên các trang với Report Header hoặc Footer
- XIX. Thay đổi kích cỡ của Header, Footer, hoặc mục chọn khác trên một Form hoặc Report

"Don't study, don't know - Studying you will know!"

NGUYEN TRUNG HOA

GIÁO TRÌNH MÔN HỌC

# LẬP TRÌNH TRỰC QUAN

*Tài liệu tham khảo*



**PHẦN I**

**MICROSOFT ACCESS**





## **BÀI 1. MỞ ĐẦU**

Lịch sử phát triển của Tin học luôn gắn liền với việc tìm kiếm các phương pháp lập trình để giúp cho người sử dụng triển khai các ứng dụng một cách dễ dàng, nhanh chóng và hiệu quả.

Như chúng ta đã biết, mỗi loại máy tính chỉ có thể hiểu và thực hiện được các lệnh cũng như chương trình theo một loại ngôn ngữ dành riêng được gọi là ngôn ngữ máy. Tuy nhiên, nếu triển khai các ứng dụng trong thực tế mà phải viết chương trình trực tiếp bằng ngôn ngữ máy thì sẽ rất phức tạp, đòi hỏi thời gian và công sức rất lớn, nhiều khi không thể thực hiện được. Vì vậy, người ta tìm cách xây dựng một ngôn ngữ lập trình riêng gắn với các ngôn ngữ tự nhiên, thuận lợi cho việc triển khai các ứng dụng. Khi thực hiện các chương trình bằng ngôn ngữ này phải qua một bước dịch chương trình đó sang ngôn ngữ máy để nó có thể thực hiện. Từ trước đến nay có rất nhiều ngôn ngữ lập trình được ra đời và phục vụ đắc lực cho việc khai các ứng dụng trên máy tính.

Trong giai đoạn đầu, các ngôn ngữ lập trình tuy dễ sử dụng hơn ngôn ngữ máy nhưng rất khó với các lập trình viên vì đặc điểm chưa đủ mạnh để dễ dàng triển khai các thuật toán. Chương trình chưa có tính cấu trúc chặt chẽ về mặt dữ liệu cũng như tổ chức chương trình. Vì vậy, việc triển khai các ứng dụng trong thực tế bằng các ngôn ngữ lập trình này là rất khó khăn.

Giai đoạn 2 là thời kỳ của các ngôn ngữ lập trình có cấu trúc. Các ngôn ngữ lập trình này có đặc điểm là có tính cấu trúc chặt chẽ về mặt dữ liệu và tổ chức chương trình. Một loạt các ngôn ngữ lập trình có cấu trúc ra đời và được sử dụng rộng rãi như : PASCAL, C, PROLOG...

Giai đoạn 3 là thời kỳ của lập trình hướng đối tượng và phương pháp lập trình có bước biến đổi mạnh. Trong các ngôn ngữ lập trình có cấu trúc thì một ứng dụng bao gồm hai thành phần riêng là dữ liệu và chương trình. Tuy chúng có quan hệ chặt chẽ nhưng là hai đối tượng riêng biệt. Trong phương pháp lập trình hướng đối tượng thì mỗi một đối tượng lập trình sẽ bao hàm cả dữ liệu và phương thức hành động trên dữ liệu đó. Vì vậy, việc lập trình sẽ đơn giản và mang tính kế thừa cao, tiết kiệm được thời gian lập trình.

Tuy nhiên, với các phương pháp lập trình trên đều đòi hỏi lập trình viên phải nhớ rất nhiều câu lệnh với mỗi lệnh có một cú pháp và tác dụng riêng, khi viết chương trình phải tự lắp nối các lệnh để có một chương trình giải quyết từng bài toán riêng biệt.

Trong xu hướng phát triển mạnh mẽ hiện nay của Tin học, số người sử dụng máy tính tăng lên rất nhanh và máy tính được sử dụng trong hầu hết các lĩnh vực của đời sống nên đòi hỏi các ngôn ngữ lập trình cũng phải đơn giản, dễ sử dụng và mang tính đại chúng cao. Chính vì vậy phương pháp lập trình trực quan ra đời. Đặc điểm của các ngôn ngữ lập trình trực quan là dễ sử dụng, triển khai các ứng dụng một cách nhanh chóng.

Hiện nay các ngôn ngữ lập trình, hệ quản trị cơ sở dữ liệu theo hướng trực quan thường dùng như : Visual Basic, Visual Foxpro, Visual C, Delphi...

Trong chương trình này giới thiệu một số chương trình lập trình thường dùng như Access, Basic và VB .Net để làm quen với phương pháp lập trình trực quan trong việc triển khai một số các ứng dụng.

Đặc điểm nổi bật của phương pháp lập trình trực quan là :

- Cho phép xây dựng chương trình theo một hướng khác gọi là event - driven programming (lập trình theo tính hướng), nghĩa là một chương trình ứng dụng được viết theo kiểu này đáp ứng dựa theo tình huống xảy ra lúc thực hiện chương trình. Tình huống này bao gồm người sử dụng ấn một phím tương ứng, chọn lựa một nút lệnh hoặc gọi một lệnh từ một ứng dụng khác chạy song song cùng lúc.
- Người lập trình trực tiếp tạo ra các khung giao diện (interface), ứng dụng thông qua các thao tác trên màn hình dựa vào các đối tượng (object) như hộp hội thoại hoặc nút điều khiển (control button), những đối tượng này mang các thuộc tính (properties) riêng biệt như : màu sắc, Font chữ.. mà ta chỉ cần chọn lựa trên một danh sách cho sẵn.
- Khi dùng các ngôn ngữ lập trình trực quan ta rất ít khi phải tự viết các lệnh, tổ chức chương trình... một cách rắc rối mà chỉ cần khai báo việc gì cần làm khi một tình huống xuất hiện.
- Máy tính sẽ dựa vào phần thiết kế và khai báo của lập trình viên để tự động tạo lập chương trình.

Như vậy với lập trình trực quan người lập trình viên giống như một nhà thiết kế, tổ chức để tạo ra các biểu mẫu, đề nghị các công việc cần thực hiện và máy tính sẽ dựa vào đó để xây dựng chương trình. Trong chương trình này ta sẽ xét cách sử dụng hệ quản trị cơ sở dữ liệu Microsoft Access và ngôn ngữ lập trình Visual Basic.

## **BÀI 2. ACCESS**

### **2.1. Giới thiệu**

Microsoft Access là một phần mềm quản lý cơ sở dữ liệu rất mạnh và được sử dụng rộng rãi hiện nay. Nó cho phép người sử dụng quản lý, bảo trì và khai thác số liệu được lưu trữ một cách có tổ chức trên máy tính.

Access nằm trong bộ Microsoft Office của công ty Microsoft. Trong chương trình này chúng tôi giới thiệu trên phiên bản Access 98, đây là phiên bản mới có nhiều cải tiến so với các phiên bản trước đây.

Để sử dụng được Access 98, máy tính phải thỏa mãn các yêu cầu cơ bản sau :

- CPU Pentium trở lên.
- Bộ nhớ RAM 32 MB trở lên.
- Hệ điều hành Windows 95 trở đi.

Trong phiên bản này chúng ta được hưởng một số công cụ bổ sung so với các phiên bản cũ trước đây như : truy cập dữ liệu Access từ các trang Web, quản lý các tập tin có chứa các liên kết đến những tập tin khác, hỗ trợ đa ngữ, quản lý dễ dàng các đối tượng đồ họa, sử dụng các Macro hỗ trợ cho tự động hóa việc quản lý dữ liệu...

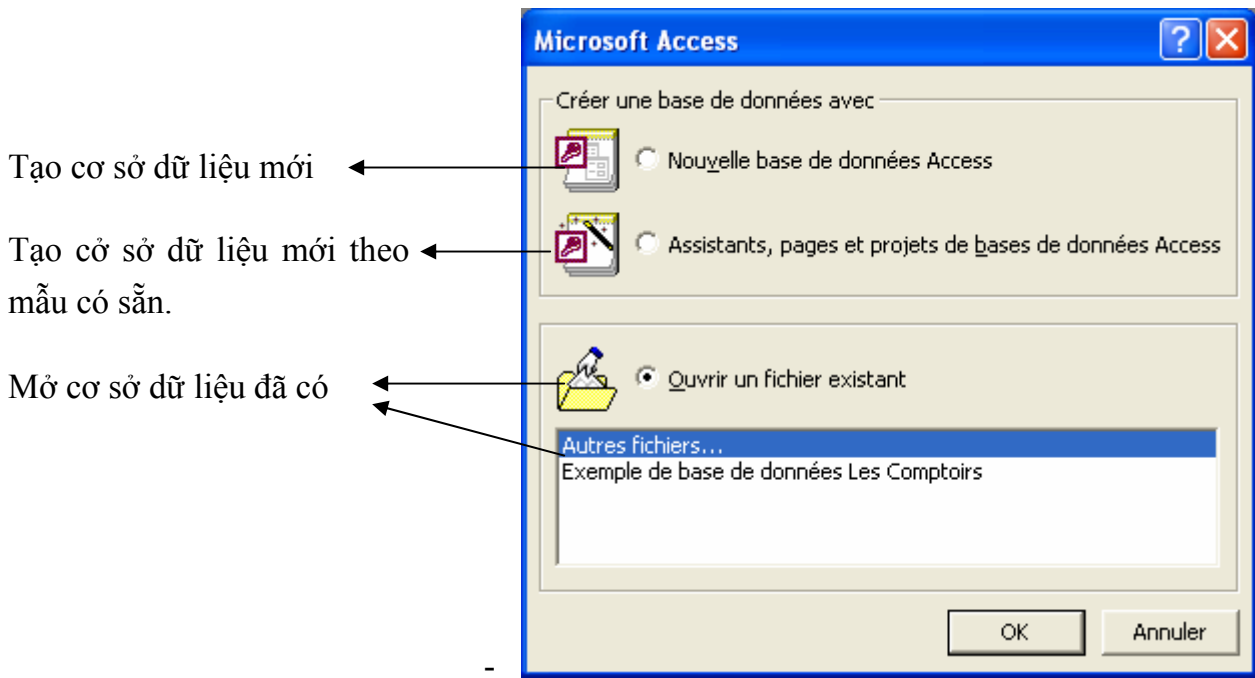
### **2.2. Khởi động ACCESS**

Sau khi đã cài đặt Microsoft Office (chọn component Access), mỗi lần làm việc với Access chúng ta khởi động :

- Bật máy tính
- Chọn Start --> Program --> Microsoft Access

*Hoặc nhấn đúp chuột tại biểu tượng của Access trên Desktop.*

Lúc đó sẽ xuất hiện làm việc của ACCESS như sau :



### 2.3. Khái niệm về cơ sở dữ liệu trong Access

Cơ sở dữ liệu là một tập hợp các dữ liệu liên quan đến một chủ đề hay một mục đích quản lý nào đó. Các thành phần của cơ sở dữ liệu Access bao gồm :

- TABLE (bảng) : là thành phần cơ bản của cơ sở dữ liệu, nó cho phép lưu trữ dữ liệu phục vụ công tác quản lý. Trong một Table, số liệu được tổ chức thành các trường (Field) và các bản ghi (Record).
- QUERY (vấn tin) : là công cụ để truy vấn thông tin và thực hiện các thao tác trên dữ liệu. Query cho phép liên kết các dữ liệu từ nhiều Table khác nhau, chọn lựa các thông tin cần quan tâm, nó là nền tảng để xây dựng các báo cáo theo yêu cầu thực tế.
- FORM (mẫu) : cho phép xây dựng các mẫu nhập số liệu giống như trong thực tế. Ta có thể cùng lúc nhập số liệu vào nhiều Table khác nhau thông qua SubForm.
- REPORT (báo cáo) : là các báo cáo số liệu để thông báo kết quả cho người sử dụng. Trong Report ta có thể kết hợp với Query để tạo các báo cáo theo những yêu cầu khác nhau trong thực tế. Trên Report bao gồm số liệu, hình ảnh, đồ thị... để mô tả cho số liệu.

- MACRO (lệnh ngầm) : là một tập hợp các lệnh nhằm tự động thực hiện các thao tác thường gặp. Khi gọi Macro, Access sẽ tự động thực hiện một dãy các lệnh tương ứng, nó được xem là một cụ lập trình đơn giản, cho phép người sử dụng chọn lựa công việc tùy theo tình huống hiện tại.
- MODULE (đơn thể) : một dạng tự động hóa chuyên sâu hơn Macro, đó là những hàm riêng của người sử dụng được viết bằng ngôn ngữ VBA. Ta chỉ nên sử dụng Module trong trường hợp các Macro không đáp ứng được yêu cầu đó.

## **2.4. Các phép toán**

### 2.4.1 Các phép toán Logic

- Not : cho kết quả ngược lại
- And : cho kết quả đúng chỉ khi cả hai đều đúng.
- Or : cho kết quả sai chỉ khi cả hai đều sai.
- Xor : cho kết quả đúng khi hai điều kiện có giá trị trái nhau.
- Epv : cho kết quả đúng chỉ khi hai điều kiện có cùng giá trị.

### 2.4.2 Các phép toán số học

- ^ : lũy thừa.
- \* : nhân.
- / : chia
- \ : chia lấy phần nguyên.
- Mod : chia lấy phần dư
- + : cộng
- - : trừ

2.4.3 Các phép toán so sánh : >, >=, <, <=, = và <>

2.4.4 Dấu rào :

- " ... " : rào giá trị chuỗi. Ví dụ : "Nguyễn Văn A"
- [ ... ] : rào tên biến. Ví dụ : [diem1] + [diem2]
- #mm/dd/yy# : rào giá trị ngày. Ví dụ : #01/01/68#

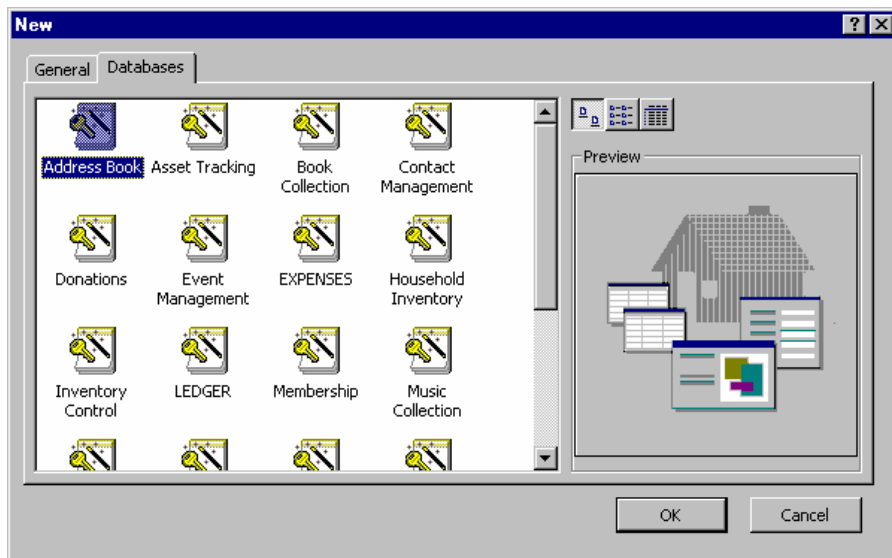
## BÀI 3. LÀM VIỆC VỚI CƠ SỞ DỮ LIỆU

### 3.1. TẠO CƠ SỞ DỮ LIỆU

#### 3.1.1 Tạo cơ sở dữ liệu bằng WIZARD

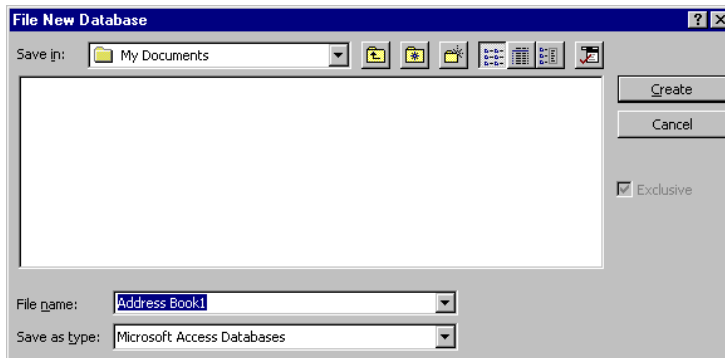
Cho phép tạo cơ sở dữ liệu theo sự hướng dẫn của ACCESS thông qua các mẫu có sẵn. Thông thường các cơ sở dữ liệu này không phù hợp với cách tổ chức cơ sở dữ liệu thường dùng nên nếu tạo cơ sở dữ liệu theo kiểu này đòi hỏi phải sửa đổi nhiều. Không nên tạo cơ sở dữ liệu theo kiểu này.

- Bước 1 : ngay sau khi khởi động ACEESS ta chọn vào nút Database Wizard và OK.
- Bước 2 : lúc đó trên màn hình xuất hiện cửa sổ sau :



Lúc này ta chọn một mẫu cơ sở dữ liệu ở trên bằng cách Double Click chuột tại biểu tượng tương ứng rồi chọn OK.

- Bước 3 : lúc đó trên màn hình xuất hiện cửa sổ sau :



Lúc này phải vào tên của cơ sở dữ liệu trong mục : **File name** :, sau đó chọn **Create**

Tiếp tục trên màn hình sẽ xuất hiện các cửa sổ yêu cầu khai báo danh sách các Table, các Field, kiểu màn hình, các mẫu báo cáo, tiêu đề và biểu tượng của cơ sở dữ liệu...

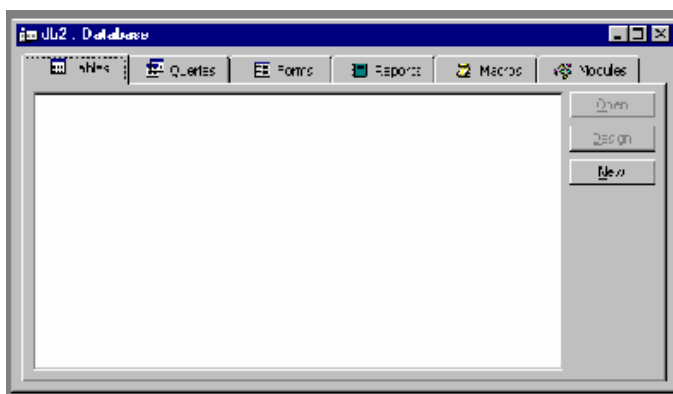
Trong các bước đó ta chỉ việc lựa chọn theo yêu cầu và Double Click vào **Next** để chuyển sang cửa sổ kế tiếp cho đến màn hình cuối thì chọn **Finish**.

### 3.1.2 Tạo cơ sở dữ liệu trống

Thông thường ta phải sử dụng mục này để tạo một cơ sở dữ liệu cho mình. ACCESS sẽ tạo ra một cơ sở dữ liệu trống và ta tự định nghĩa cho mình các Table, Query, Report, Form, Macro và Module riêng.

- Bước 1 : khi khởi động chọn Blank Database hoặc chọn File - New Database
- Bước 2 : khai báo tên của ổ đĩa, thư mục, tập tin cần tạo. Chọn Create

Lúc đó ta nhận được cơ sở dữ liệu mới, xuất hiện màn hình :

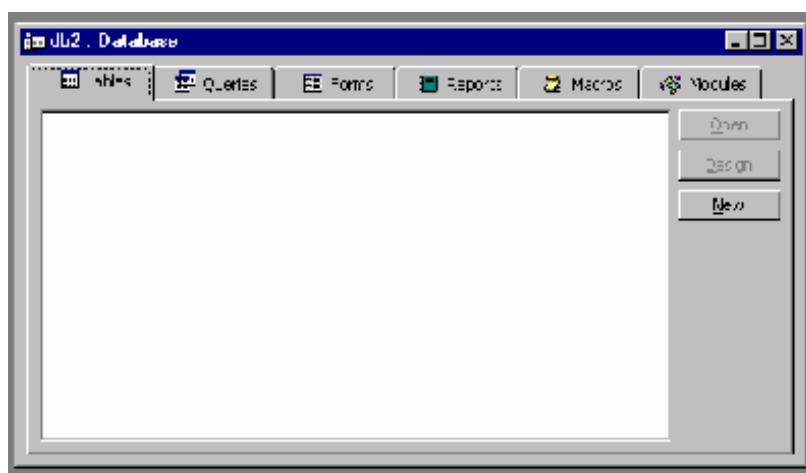




Thông thường ta phải sử dụng mục này để tạo một cơ sở dữ liệu cho mình. ACCESS sẽ tạo ra một cơ sở dữ liệu trống và ta tự định nghĩa cho mình các Table, Query, Report, Form, Macro và Module riêng.

- Bước 1 : khi khởi động chọn Blank Database hoặc chọn File - New Database
- Bước 2 : khai báo tên của ổ đĩa, thư mục, tập tin cần tạo. Chọn Create

Lúc đó ta nhận được cơ sở dữ liệu mới, xuất hiện màn hình :



### 3.2. Hiệu chỉnh cơ sở dữ liệu

Sau khi đã tạo cơ sở dữ liệu ta có thể làm việc với cơ sở dữ liệu trên thông qua Table, Report, Form, Record, Query và Module qua cửa sổ trên.

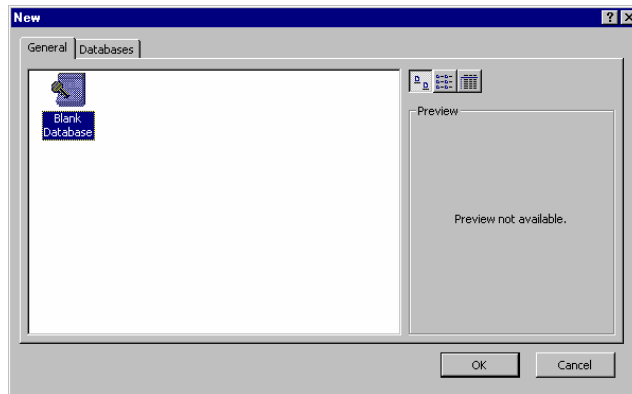
Ta sẽ xét cách thao tác lên từng thành phần một của cơ sở dữ liệu trong các bài kế tiếp.

## BÀI THỰC HÀNH

Trong tập tài liệu này cuối mỗi bài sẽ có bài thực hành và bài tập, các bài này xây dựng theo một hệ thống chung và khi đến cuối chương trình sẽ có một hệ thống chương trình hoàn chỉnh để quản lý điểm cho sinh viên..

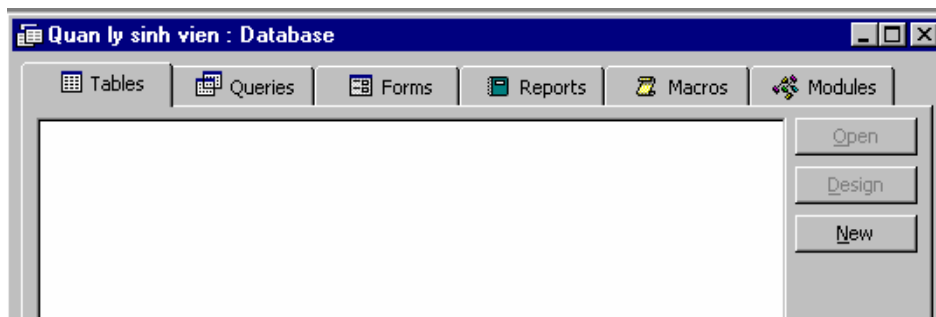
Tạo một cơ sở dữ liệu trống có tên là Quản lý sinh viên.

- Bước 1 : Chọn File - New Database
- Bước 2 : chọn nút General, bấm chuột vào biểu tượng Blank Database, sau đó chọn OK. Nếu muốn tạo CSDL theo mẫu thì chọn nút Database, sau đó chọn biểu tượng tương ứng.



- Bước 3 : gõ vào tên cơ sở dữ liệu cần tạo là **Quản lý sinh viên** trong mục **File name**. Qui định thư mục cần lưu trữ Database trong mục **Save in**. Sau đó chọn nút **Create**

Lúc này ta có cửa sổ làm việc với cơ sở dữ liệu **Quản lý sinh viên** như sau :



Lúc này ta có thể làm việc với các thành phần của Database như Table, Query, Form, Report, Macro và Modules.

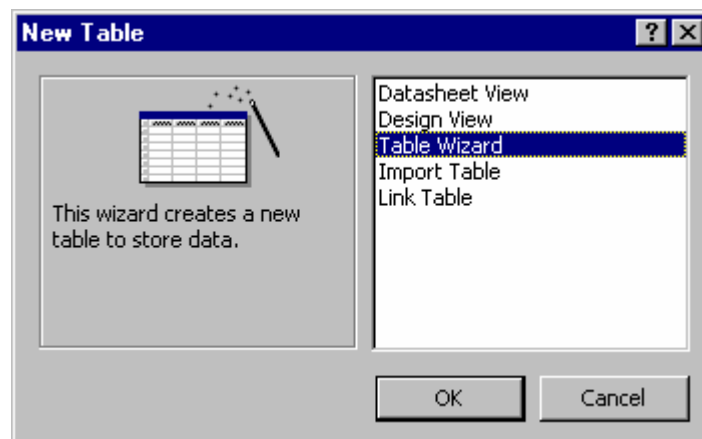
## BÀI 4. LÀM VIỆC VỚI TABLE

Table là thành phần cơ bản của cơ sở dữ liệu trong Access, nó có nhiệm vụ lưu trữ các số liệu phục vụ quá trình quản lý.

### 4.1. Tạo cấu trúc của Table

Để lưu trữ số liệu trên Table trước hết ta phải tạo cấu trúc của Table bằng cách qui định tên của Table, tên và thuộc tính của các trường.

Ta có thể tạo Table bằng cách chọn New trong hộp thoại cơ sở dữ liệu hoặc chọn trên thanh thực đơn Insert - Table, lúc đó xuất hiện cửa sổ cho phép chọn cách tạo Table như sau :



#### 4.1.1 Tạo Table bằng Wizard

Phương pháp này cho phép tạo Table theo các mẫu có sẵn của Access.

- Bước 1: chọn Table Wizard trong hộp trên rồi OK
- Bước 2: chọn tên Table, tên trường theo mẫu có sẵn của ACCESS và sửa đổi lại theo yêu cầu thực tế. Chọn NEXT để thực hiện các công việc kế tiếp như sửa tên trường, tên Table và sau cùng chọn FINISH để kết thúc.

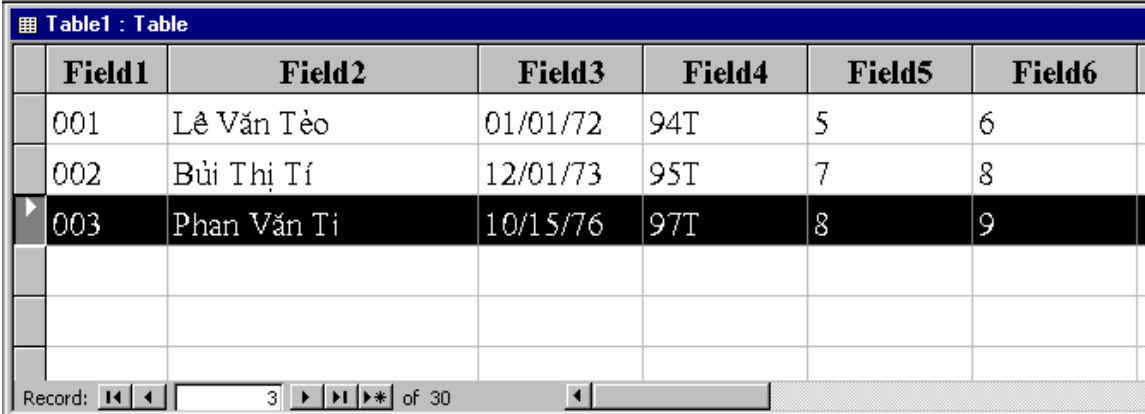
#### 4.1.2 Tạo Table bằng DATASHEET VIEW

Phương pháp này cho phép tạo Table theo cách sử dụng một mẫu biểu cho trước và ACCESS dựa vào đó để tạo ra Table.

## Lập trình trực quan

- Bước 1: chọn Datasheet View trong hộp rồi OK
- Bước 2: Nhập vào nội dung của bảng mẫu khi máy đưa ra một mẫu Table với các Column có tên là Field1, Field2...

Ví dụ : để tạo Table lưu trữ điểm sinh viên ta nhập :

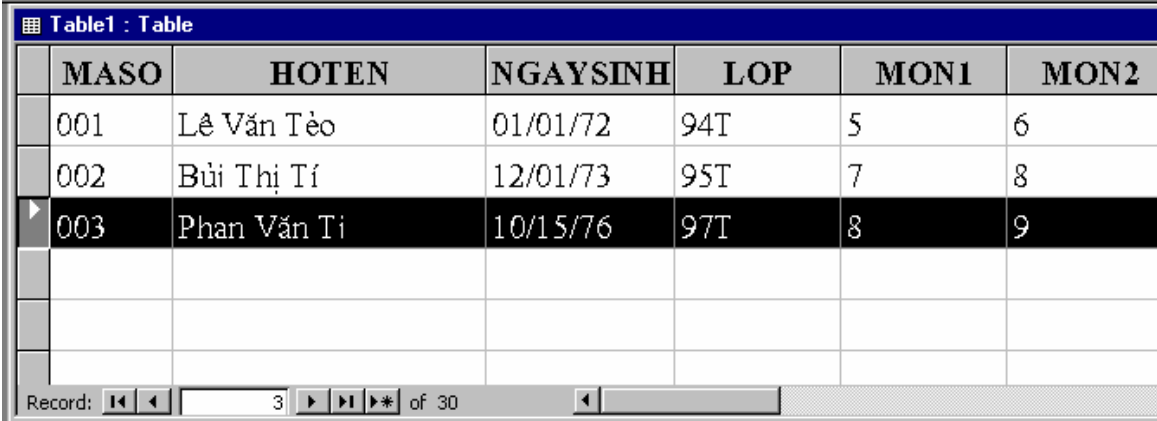


Field1	Field2	Field3	Field4	Field5	Field6
001	Lê Văn Tèo	01/01/72	94T	5	6
002	Bùi Thị Tí	12/01/73	95T	7	8
003	Phan Văn Ti	10/15/76	97T	8	9

Record: 3 of 30

- Bước 3: hiệu chỉnh lại tên trường bằng cách đưa dấu chuột vào đỉnh cột cần sửa và nhấn nút chốt bên phải rồi chọn Rename Column (Hoặc để con trỏ ở ô có cột cần sửa chọn trên thực đơn Format - Reneme Column). Sau đó gõ lại tên trường.

Ví dụ ta nhập lại tên các trường trên Table cũ như sau :



MASO	HOTEN	NGAYSINH	LOP	MON1	MON2
001	Lê Văn Tèo	01/01/72	94T	5	6
002	Bùi Thị Tí	12/01/73	95T	7	8
003	Phan Văn Ti	10/15/76	97T	8	9

Record: 3 of 30

- Bước 4: đóng Talbe (chọn File - Close)
  - Máy hỏi có ghi hay không, chọn Yes. để ghi, No nếu không.
  - Đặt tên cho Table trong bảng Save As
  - Máy hỏi có đặt khóa cơ sở Primary Key hay không, nếu có thì Yes, không thì No.

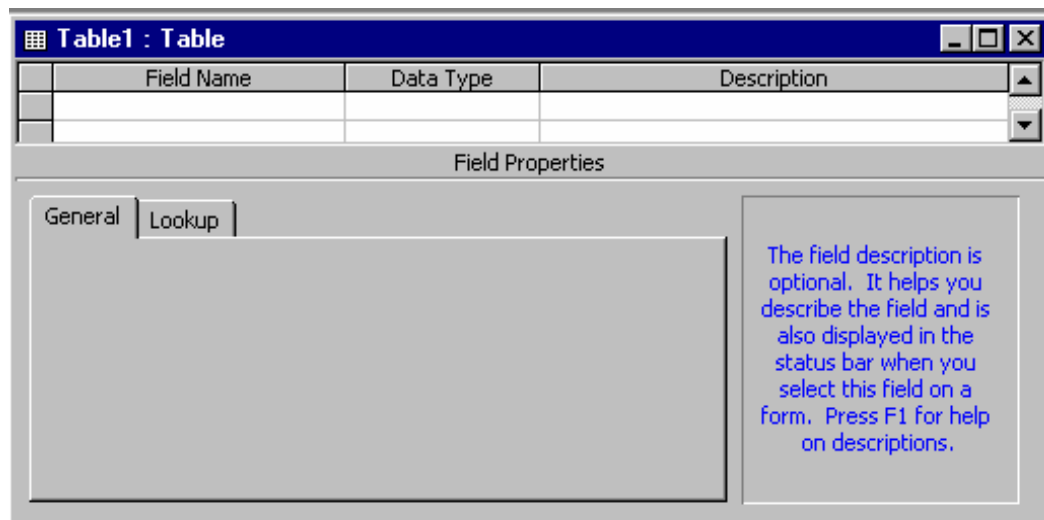
Lúc này máy sẽ tự định nghĩa một Table theo mẫu vừa tạo. Nếu muốn hiệu chỉnh thêm thì chọn Design.

Chú ý : tên trường và tên Table dài tối đa là 64 ký tự, bắt đầu bằng 0..9 hoặc A..Z, có thể là ký tự trống nhưng không có dấu chấm câu. Số trường tối đa trong một Table là 255. Độ lớn tối đa một Table là 1 GB.

#### 4.1.3 Tạo Table bằng DESIGN VIEW

Phương pháp này cho phép tạo Table hoàn toàn do người sử dụng qui định.

- Bước 1: chọn Design View trong hộp rồi OK
- Bước 2: xuất hiện màn hình thiết kế Table như sau :



- Field name : khai báo tên của trường.
- Data Type : khai báo kiểu dữ liệu tương ứng của trường.
- Description : nội dung mô tả cho trường. Nội dung được dùng làm tiêu đề cho trường khi thiết lập các Form hay Report khi dùng Wizard.

Trong mục Data Type, chúng ta có thể chọn một trong các kiểu sau :

Tên	Ý nghĩa
Text	Chứa tập hợp các ký tự tùy ý, dài tối đa 255 ký tự
Memo	Dài tối đa 65535 ký tự
Number	Chứa giá trị số

Date/Time	Giá trị ngày hoặc giờ
Currency	Tiền tệ, có đơn vị tính
Auto Number	Giá trị số nhưng không thay đổi được dạng thể hiện
Yes/No	Giá trị True hoặc False
Hyperlink	Nội dung là văn bản hay kết hợp giữa văn bản và số được sử dụng như một địa chỉ hyperlink (siêu liên kết)
Lookup Wizard	Chọn một giá trị trong danh sách các giá trị cho trước

Chú ý : tương ứng với mỗi kiểu dữ liệu sẽ khai báo thêm các thuộc tính của nó trong Field Properties gồm các thuộc tính chung (General) và thuộc tính nhập số liệu (Lookup).


Ví dụ : với kiểu dữ liệu Text ta khai thêm :

Tên	Ý nghĩa
Field Size	Độ rộng tối đa chứa sẵn
Format	Cách hiển thị giá trị
Input Mask	Qui định mẫu nhập liệu
Caption	Một chú thích khác cho Field, dùng với Form, Report
Default Value	Giá trị cho trước
Validation Rule	Qui định cách kiểm tra số liệu nhập
Validation Text	Thông báo khi nhập số liệu sai
Required	Chọn Yes nếu bắt buộc phải nhập nội dung
Allow Zero Length	Chọn Yes nếu chấp nhận giá trị rỗng
Indexed	Có chỉ mục hay không, nếu có thì được trùng hay không (No, Yes Duplicate OK, Yes No Duplicate)

## 4.2. Nhập số liệu vào Table

Sau khi đã tạo xong Table ta có thể nhập số liệu vào đó bất kỳ lúc nào bằng cách :

- Double Click vào tên Table cần nhập.
- Để vệt sáng ở tên Table cần nhập rồi chọn Open

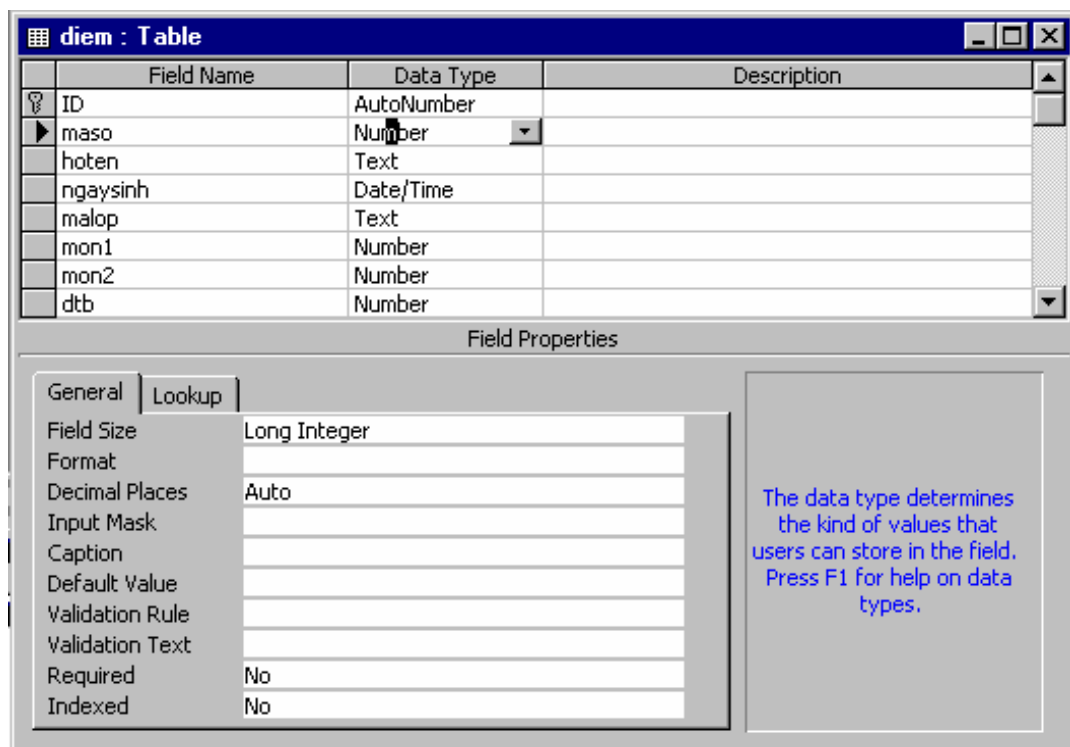
Chú ý : trong quá trình nhập ta có thể điều chỉnh độ rộng các cột cho thích hợp bằng cách đưa dấu chuột về cạnh bên phải của tiêu đề cột cho xuất hiện dấu  rồi Drag chuột để điều chỉnh. Khi đóng ta lưu Layout bằng cách trả lời Yes

### 4.3. Hiệu chỉnh Table

Ta có thể hiệu chỉnh Table để : thay đổi cấu trúc bản ghi, sửa đổi nội dung bản ghi hoặc cách trình bày.

#### 4.3.1 Thay đổi cấu trúc bản ghi

- Chọn tên của Table cần hiệu chỉnh.
- Chọn Design
- Hiệu chỉnh lại qua bảng :



Ta có thể thay đổi các thông tin liên quan đến các trường trong Table từ tên trường, kiểu, các thuộc tính, thêm bớt các trường...

#### 4.3.2 Thay đổi nội dung bản ghi

- Chọn tên của Table cần hiệu chỉnh.
- Chọn Open hoặc Double Click tại đó.
- Hiệu chỉnh số liệu giống như trong Excel.

### 4.3.3 Thay đổi cách trình bày

- Chọn tên của Table cần hiệu chỉnh.
- Chọn Open hoặc Double Click tại đó.
- Chọn Format để định dạng, sau đó :
  - Font : thay đổi kiểu chữ.
  - Cells : thay đổi cách thể hiện như : Gridlines Shown (che hay hiện đường lưới), Cell Effect (trình bày ô số liệu phẳng, nhô lên hoặc lõm xuống), Gridline Color (màu sắc của nét gạch), Background Color (màu nền của ô).
  - Column Width : qui định độ rộng cột.
  - Hide Column : che bớt cột. Nếu muốn hiện lại chọn Unhide Column.

### 4.4. Khai thác số liệu trên Table

Cho phép khai thác số liệu một cách tức thời khi đang làm việc trực tiếp trên Table. Nếu muốn tự động hóa công tác khai thác thông tin và có các báo cáo đẹp mắt thì ta phải dùng Report, Query, Macro hoặc lập trình bằng Visual Basic.

#### 4.4.1 Tìm và thay thế

Cho phép tìm và thay thế nội dung trên một trường nào đó trong Table.

- Đưa con trỏ về trường cần tìm và thay thế.
- Chọn Edit - Replace

#### 4.4.2 Thay đổi vị trí trường

- Chọn cột cần thay đổi vị trí (đưa dấu chuột lên tiêu đề trường).
- Drag chuột để đưa trường về vị trí mới.

#### 4.4.3 Sắp xếp

- Chọn trường làm khóa để sắp xếp.



- Chọn trên thanh thực đơn Record - Sort (hoặc chọn biểu tượng)
- Chọn sắp tăng dần (Sort Ascending) hoặc giảm dần (Descending).

#### 4.4.4 Lọc bản ghi

- Chọn trên thanh thực đơn Record - Filter (hoặc chọn biểu tượng)
- Chọn Filter by Form

Quy định cách lọc :

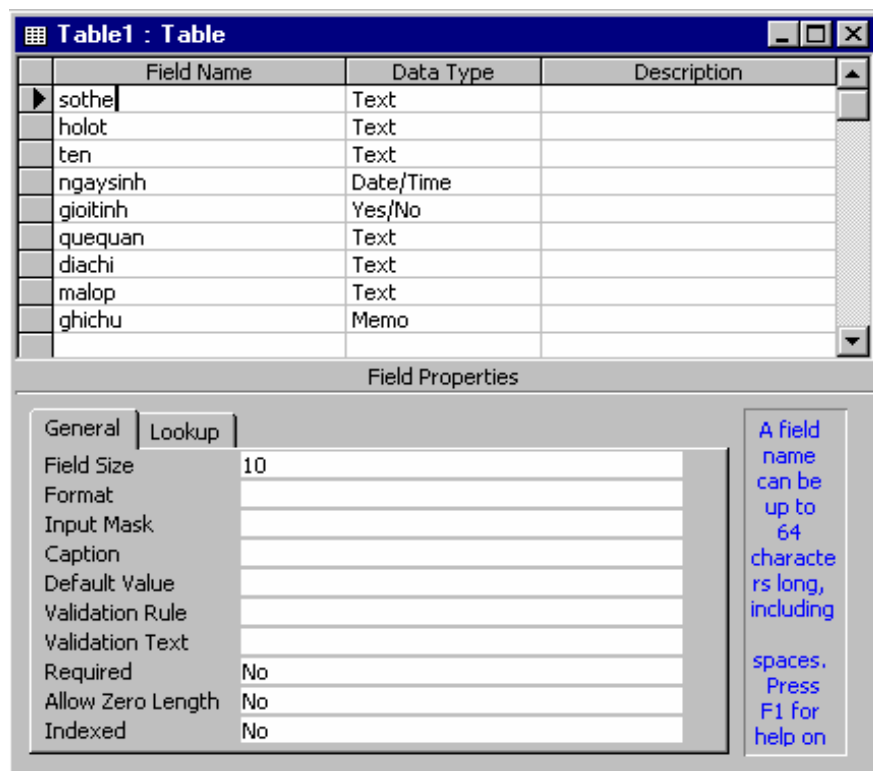
- Muốn lọc theo trường nào ta chỉ việc bấm chốt vào trường đó và chọn giá trị làm điều kiện để lọc.
- Bấm phím phải của chuột chọn Apply Filter. Lúc này chỉ còn các bản ghi thỏa mãn điều kiện.
- Nếu muốn hủy lọc thì bấm phím phải của chuột chọn Remove Filter. Lúc này hiện tất cả các bản ghi như ban đầu.

Chú ý : trong quá trình lọc ta có thể dùng các điều kiện với các phép toán so sánh và quan hệ.

## BÀI THỰC HÀNH

**Câu 1 :** Tạo Table quản lý hồ sơ sinh viên gồm các thông tin : số thẻ sinh viên, họ lót, tên, ngày sinh, giới tính, quê quán, mã lớp, địa chỉ và ghi chú.

- Bước 1: chọn nút Table, sau đó New
- Bước 2: chọn Design View, rồi và xuất hiện màn hình thiết kế Table và khai báo các trường như sau :



Lúc đó ta có bảng sau :

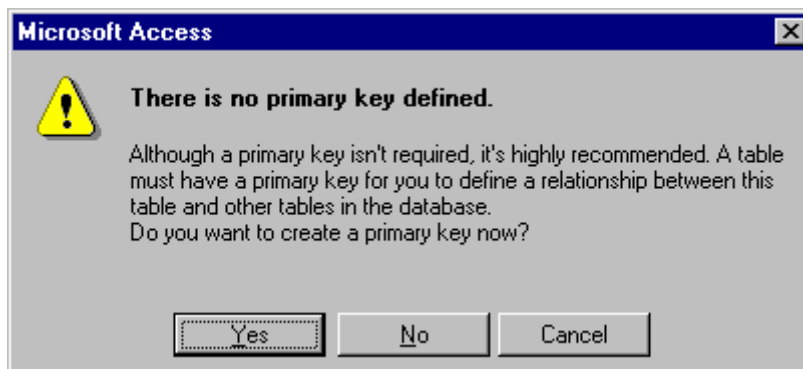
Field Name	Data Type	Field Size	Format
Sothe	Text	6	
Holot	Text	27	
Ten	Text	6	
Ngaysinh	Date/Time		Sort Date
Gioitinh	Yes/No		True/False
Ghichu	Memo		

## Lập trình trực quan

- Bước 3: đóng cửa sổ khai báo cấu trúc Table (bấm chuột vào dấu X ở góc trên bên phải) và chọn Yes để ghi lại cấu trúc.
- Bước 4 : gõ vào tên của Table là Ho So và chọn OK trên cửa sổ sau :



- Bước 5 : qui định có định nghĩa khóa chính hay không qua màn hình sau :



Chọn No (không định nghĩa khóa).

Lúc này ta có một Table vừa được định nghĩa xong.

**Câu 2 :** Nhập số liệu vào Table :

- Đưa con trỏ về tên Table.
- Chọn Open.
- Nhập số liệu qua màn hình như sau.

Quan ly sinh vien : Table									
	sothe	holot	ten	ngaysinh	gioitinh	quequan	diachi	malop	gichu
▶	001	Lê Văn	Huy	1/1/72	<input checked="" type="checkbox"/>	Đà Nẵng	14 Hải Phòng	94T	
	002	Phan Huy	Chú	3/20/74	<input checked="" type="checkbox"/>	Hà Nội	12 Lê Lai	95T	
	003	Võ Văn	Lễ	1/3/74	<input checked="" type="checkbox"/>	Huế	14 Lê Lợi	94T	
	004	Trần Thị	Cuội	1/24/72	<input type="checkbox"/>	Nha Trang	12 Hòa Cường	93T	
*					<input type="checkbox"/>				

Chú ý : trong quá trình nhập số liệu ta thường dùng các thao tác sau :

- Điều chỉnh độ rộng cột : đưa chuột về cạnh phải tiêu đề cột rồi Drag chuột.

- Đổi kiểu chữ : chọn Format - Font , chọn kiểu chữ thích hợp.

**Câu 3 :** sắp xếp theo thứ tự ABC của tên

- Chọn trường làm khóa để sắp xếp là Tên (để con trỏ ở trường đó).
- Chọn trên thanh thực đơn Record - Sort (hoặc chọn biểu tượng)
- Chọn sắp tăng dần : Sort Ascending

**Câu 4 :** chỉ xem các sinh viên lớp 94T

- Chọn trên thanh thực đơn Record - Filter (hoặc chọn biểu tượng)
- Chọn Filter by Form
- Qui định cách lọc theo mẫu sau :



Ho So: Filter by Form									
	sothe	holot	ten	ngaysinh	gioitinh	diachi	quequan	malop	ghic
▶								94T	
								93T	
								94T	
								95T	

- Đưa con chuột về trường Malop Click chuột và chọn tên lớp là 94T
- Bấm phím phải của chuột chọn Apply Filter. Lúc này chỉ còn các bản ghi thỏa mãn điều kiện.
- Nếu muốn hủy lọc thì bấm phím phải của chuột chọn Remove Filter. Lúc này hiện tất cả các bản ghi như ban đầu.

**Câu 5 :** xóa một bản ghi.

- Bấm chuột ở phần tiêu đề dòng của bản ghi cần xóa (cả bản ghi đổi màu).
- Bấm phím Delete
- Máy hỏi có muốn xóa hay không (You are about to delete n records ?), chọn Yes nếu muốn xóa, No nếu không.

**Câu 6 :** thay đổi cách thể hiện ô.

- Chọn Format - Cells.
- Qui định cách thể hiện.

**Bài tập :** tương tự tạo các Table để lưu trữ danh mục lớp, lưu trữ học phí, học bổng và thực hiện các thao tác lên bản ghi.

## **BÀI 5. LÀM VIỆC VỚI QUERY**

### **5.1. Khái niệm**

Query là một công cụ cho phép người sử dụng thống kê số liệu, xây dựng các báo cáo tổng hợp dưới nhiều hình thức khác nhau trên dữ liệu gốc trong Table.

Muốn làm việc được với Query trước hết ta phải có Database và Table với dữ liệu nhập vào sẵn.

Query còn được dùng để tạo ra dữ liệu phục vụ cho các công cụ khác như Report, Form và cho cả một Query khác.

Tùy theo mục đích khai thác ta có thể sử dụng một trong các loại Query sau :

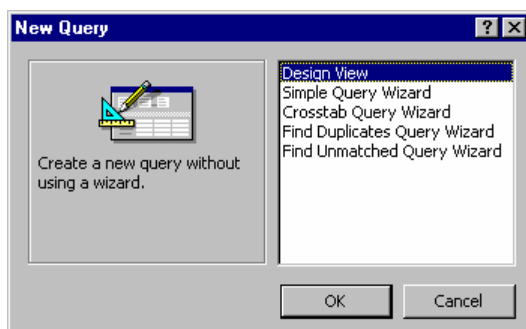
- Select Query : cho phép chọn lựa các bản ghi, tạo thêm các vùng tính toán và trả về kết quả là các bản ghi thỏa mãn điều kiện. Ta có thể dùng Query để thao tác trên nhiều Table cùng lúc.
- Append Query : nối thêm dữ liệu từ các bản ghi của một hay nhiều Table vào cuối một Table khác.
- Make-Table Query : tạo ra một Table mới từ một Dynaset (Dynamic Dataset). Cho phép tạo các Table dự phòng, trích bản ghi để lưu trữ trước khi xóa các bản ghi này khỏi Table hiện hành.
- Delete Query : xóa một nhóm các bản ghi từ một hay nhiều Table.
- Cross Tab Query : Query tham chiếu chéo, được dùng để tạo nhóm dữ liệu và trả về kết quả dưới dạng một bản tính kèm theo số cộng ngang, cộng dọc. Ta thường dùng loại này để tạo dữ liệu phục vụ cho các Report và Chart.
- Find Duplicate Query : tìm trong Table những bản ghi có giá trị giống nhau ở trên tất cả các trường.

- Find Unmatched Query : tìm những bản ghi mà giá trị của nó không trùng với giá trị của bất cứ một bản ghi nào trên một Table khác.
- Union Query : nối các bản ghi của hai hay nhiều Table thành một danh sách chung.
- Pass-Through Query : Query chuyển giao, dùng để gửi lệnh trực tiếp đến hệ ngôn ngữ SQL (Structured Query Language) của ACCESS.
- Data Definition Query : sử dụng các lệnh của ngôn ngữ SQL để tạo hoặc sửa đổi cấu trúc của một Table trong Database.

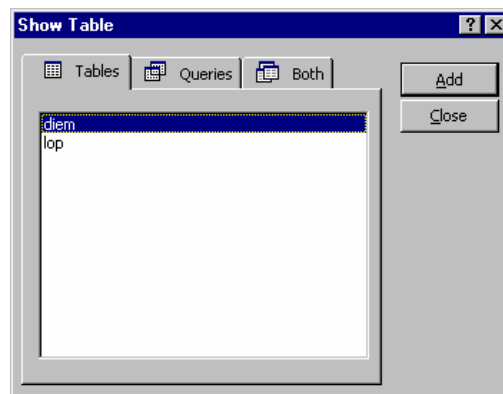
## **5.2. Cách tạo QUERY**

Muốn tạo Query ta thực hiện qua các bước sau :

- Bước 1: trong hộp Database ta chọn nút Query , chọn New
- Bước 2: chọn kiểu tạo Query qua cửa sổ sau

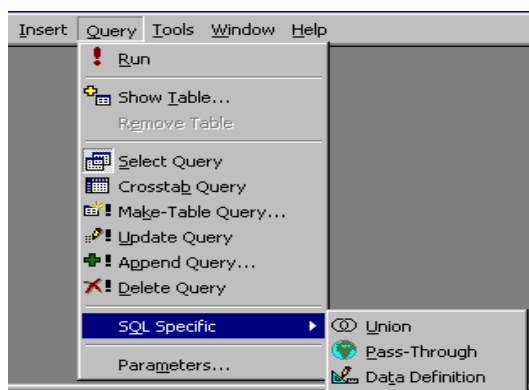


- Bước 3: chọn Table phục vụ cho việc xây dựng Query qua cửa sổ sau :



- Table : để xem danh sách tên các Table đã tạo trước đó.
- Query : để xem danh sách các Query đã có.

- Both : xem danh sách cả Table và Query.
  - Add : bổ sung Table hoặc Query được chọn cho việc tạo Query mới.
  - Close : đóng cửa sổ chọn.
- Bước 5: thiết kế Query theo yêu cầu. Nếu muốn thay đổi loại Query thì ta chọn trên thanh thực đơn chức năng Query sau đó chọn trong danh sách loại Query.

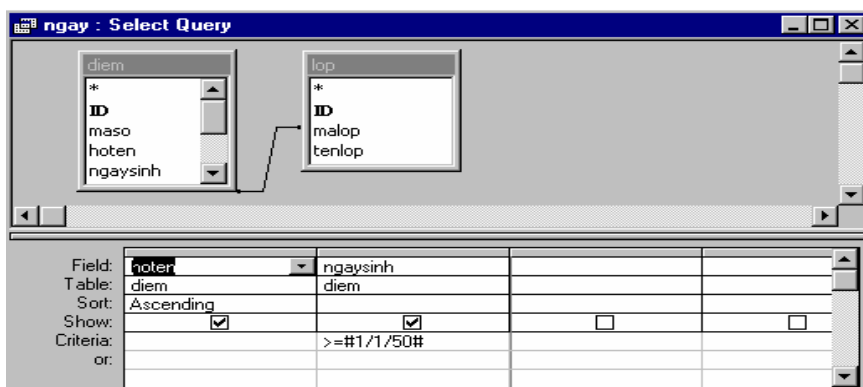


- Bước 6: tùy theo từng loại Query ta có cách thiết kế riêng. Sau khi tạo Query xong ta đóng lại và đặt tên cho Query khi máy yêu cầu.

Sau đây ta xét cách tạo một số Query thường được sử dụng (chỉ làm rõ cho bước 6).

### 5.2.1 Select Query

Sau khi chọn 5 bước trên và loại Query là Select Query màn hình sẽ xuất hiện cửa sổ khai báo Query như sau :



- Field : chọn tên các trường có liên quan đến điều kiện, sắp xếp hoặc cần xem.
- Table : hiển thị tên của Table mà trường được chọn trực thuộc vào nó.



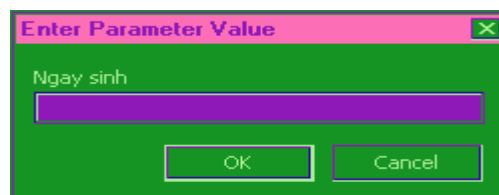
- Sort : qui định việc sắp xếp tăng (Ascending) hoặc giảm (Descending) theo nó.
- Show : cho phép hiển thị nội dung của trường hay không ( : không,  $\surd$  : có).
- Criteria : qui định điều kiện để lọc các bản ghi cần ghi. Nếu các điều kiện viết trên cùng dòng này thì ngầm định là quan hệ AND nếu viết trên dòng phía dưới thì quan hệ OR

Đóng cửa sổ Select Query và ghi tên của Query cần lưu trữ lên đĩa.

Chú ý :

- Để linh hoạt khi thay đổi điều kiện của Query trong mục Criteria thay vì gõ giá trị làm điều kiện ta nhập vào tên biến nhớ. Lúc đó mỗi khi gọi Query máy sẽ yêu cầu nhập vào giá trị tương ứng.

Ví dụ : muốn xem danh sách sinh viên sinh vào một ngày nào đó ta đưa con trỏ về ô Criteria của này sinh gõ [ngày sinh]. Khi gọi Query thực hiện sẽ xuất hiện cửa sổ hỏi ngày sinh :



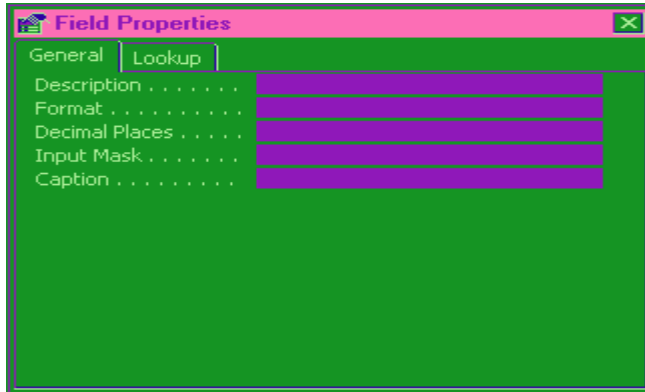
- Nếu muốn tính tổng theo từng bộ phận thì ta bấm phím phải của chuột và chọn Total, lúc đó trong Query xuất hiện một dòng Total để ta qui định phương thức tính toán.

Khi bấm vào Total ta thấy xuất hiện các hàm tính toán : Sum (tính tổng), Avg (tính giá trị trung bình), Min (tìm giá trị nhỏ nhất), Max (tìm giá trị lớn nhất), Count (đếm số lượng các giá trị), StDev (độ lệch chuẩn của các giá trị), Var (sự biến thiên của các giá trị) và các tùy chọn khác là : Group By (định nghĩa nhóm muốn tính toán), Expression (tạo ra một biểu thức tính toán, Where (chỉ định điều kiện khi tính toán).

- Nếu muốn tạo ra một trường mà nội dung của nó được tính toán từ một biểu thức bất kỳ thì ta đưa con trỏ đến cột đó viết biểu thức cần tính.

Cách viết : Tên trường : biểu thức. Ví dụ : DTB:([mon1+[mon2])/2

- Nếu muốn thay đổi thêm cho cột cần thể hiện thì ta đưa đầu chuột đến cột đó nhấn phím phải của chuột làm xuất hiện thực đơn rồi chọn Properties và qui định qua :

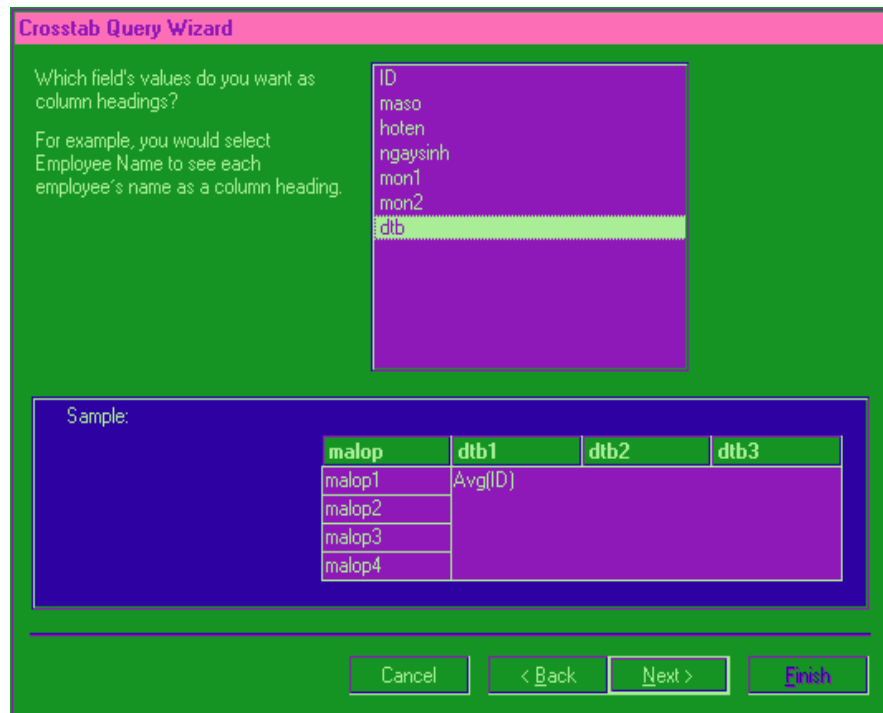


- Description : dòng chú thích.
- Format : qui định khuôn dạng cách thể hiện nội dung dữ liệu.
- Decimal Places : số chữ số ở phần thập phân.
- Input Mask : cách thức nhập số liệu.
- Caption : tiêu đề sử dụng thay cho tên trường.

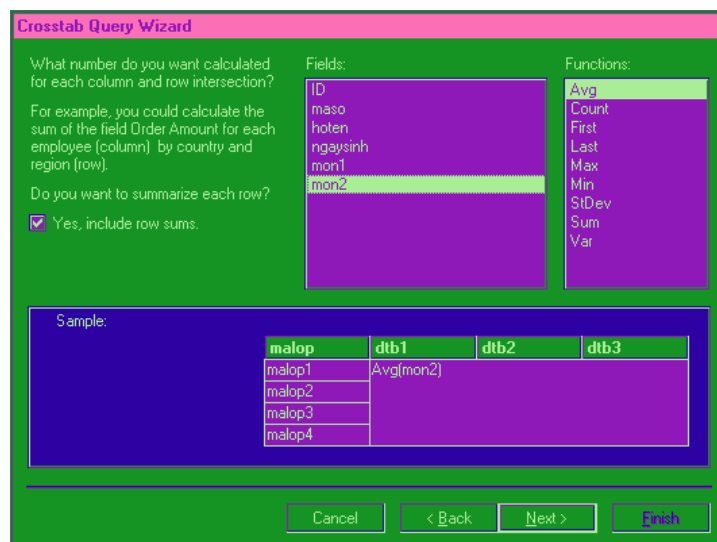
### 5.2.2 Cross Tab Query

Đây là loại Query cho phép lập bảng tham chiếu chéo : tổng hợp từ một đến nhiều chỉ tiêu theo hàng, trên mỗi hàng lại tổng hợp một chỉ tiêu khác theo cột, vùng giao nhau giữa hàng và cột thể hiện trị số tổng hợp của một chỉ tiêu thứ ba.

Chọn Cross Tab Query, sau đó chọn tên các Table chứa số liệu để lập Query hoặc tên Query cơ sở để tạo Query kế tiếp. Sau đó chọn Next để xuất hiện cửa sổ :

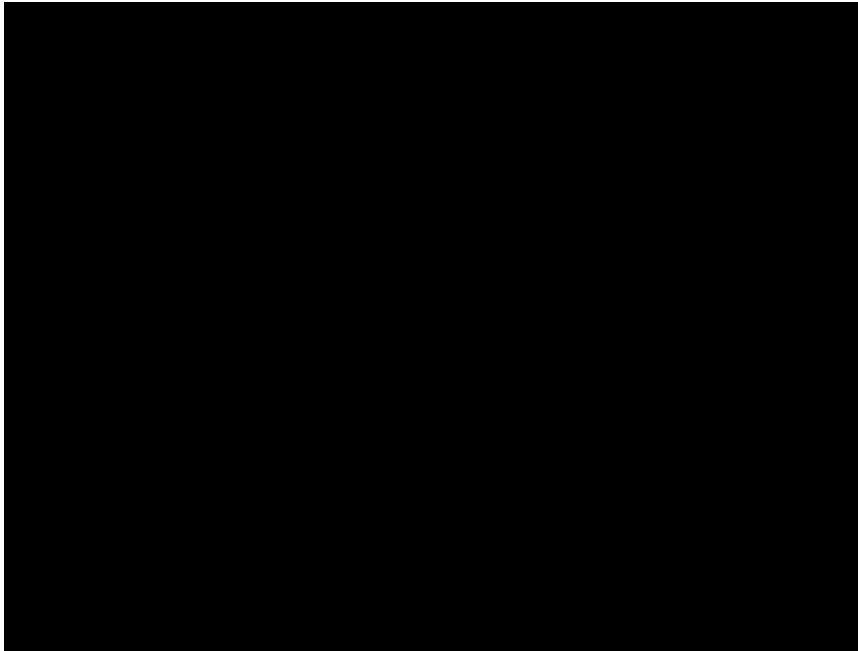


- Trong Available Field ta chọn tên trường làm tiêu đề dòng. Số trường tối đa được chọn là 3.
  - Select Filed : chứa tên các trường được chọn.
- Chọn Next để xuất hiện màn hình kế tiếp :



- Chọn tên trường sử dụng làm tiêu đề cột.

- Chọn Next để sang bước kế tiếp.
- Chọn giá trị số cần tính tại mỗi giao điểm dòng và cột.
- Chọn Next để chuyển sang bước kế tiếp.



- Gõ vào tên của Query cần tạo.
- Chọn Finish để hoàn tất tạo Query.
- Chú ý : tương tự tạo các Query còn lại.

### **5.3. Hiệu chỉnh QUERY**

Nếu muốn hiệu chỉnh lại Query thì ta đưa con trỏ về tên của Query và chọn Design để thực hiện thiết kế lại.

Nếu muốn xem lệnh tạo Query thì trong quá trình Design ta bấm nút phải của chuột rồi chọn SQL View.

### **5.4. Thực hiện QUERY**

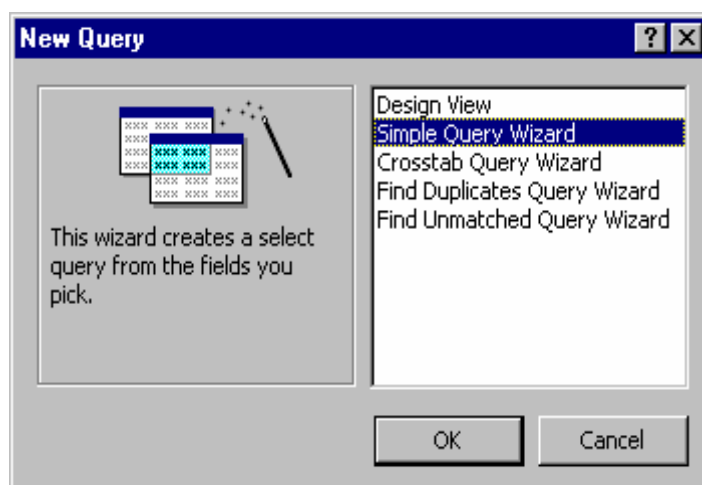
Đưa con trỏ về tên của Query và chọn Open ( hoặc Double Click chuột tại đó).

## BÀI THỰC HÀNH

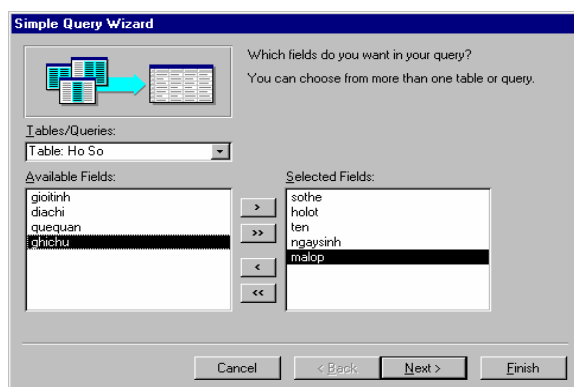
**Câu 1** : tạo Query để xem danh sách của một lớp bất kỳ nào đó nhập từ bàn phím.

Bước 1: bấm chuột vào nút Query, chọn New.

Bước 2 : chọn Simple Query Wizard, sau đó chọn OK từ bảng sau :



Bước 4 : chọn tên các trường cần đưa vào danh sách là: sothe, holot, ten, ngaysinh, malop. Sau đó chọn Next



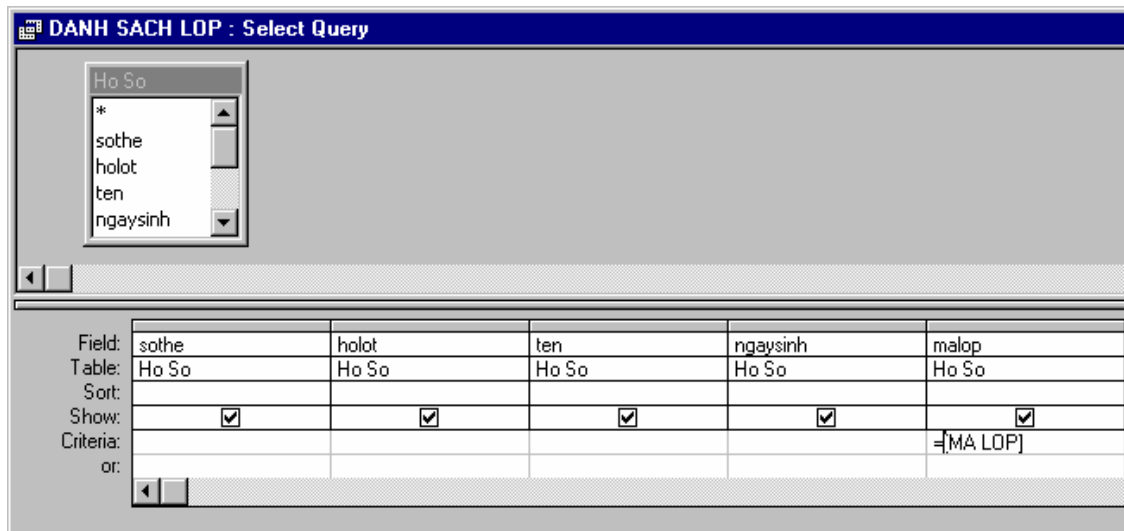
Bước 5 : qui định tên của Query là DANH SACH LOP, chọn Finish để hoàn tất.



Bước 6 : lúc này sẽ xuất hiện danh sách gồm các bản ghi với nội dung các trường đã chọn. Tuy nhiên lúc này trong danh sách xuất hiện sinh viên của tất cả các lớp có trong Table.

Bước 7 : hiệu chỉnh lại Query để chỉ xem danh sách từng lớp tùy ý.

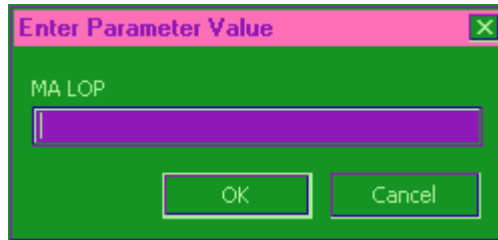
- Đưa con trỏ về tên Query cần hiệu chỉnh (DANH SACH LOP).
- Chọn Design.
- Đưa con trỏ về mục Criteria, cột Malop gõ vào điều kiện là =[MA LOP]



- Đóng cửa sổ Design Query. Chọn Yes để ghi lại Query mới.

Lúc này mỗi khi sử dụng Query (đưa con trỏ về tên Query và chọn Open) thì máy sẽ hỏi mã của lớp cần xem qua cửa sổ:

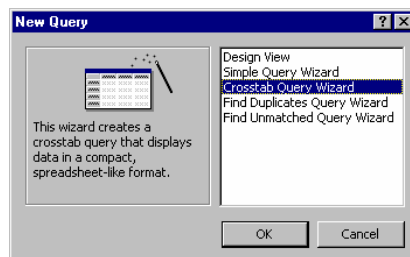
Ta nhập vào mã lớp cần xem, chọn OK. Lúc này danh sách của lớp được chỉ định sẽ hiện



lên màn hình.

**Câu 2** : tạo Query để thống kê số lượng sinh viên theo từng lớp, mỗi lớp thì cho biết số lượng nam, nữ.

Bước 1 : bấm chuột vào nút Query, sau đó chọn New



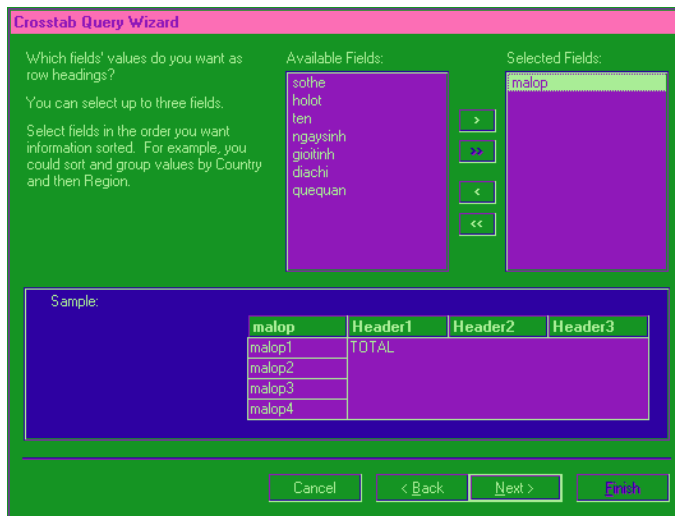
Bước 2 :Chọn Crosstab Query Wizard

- Double Click chuột tại nút OK

Bước 3: chọn tên của Table là Ho So, sau đó chọn Next để chuyển sang bước kế tiếp.

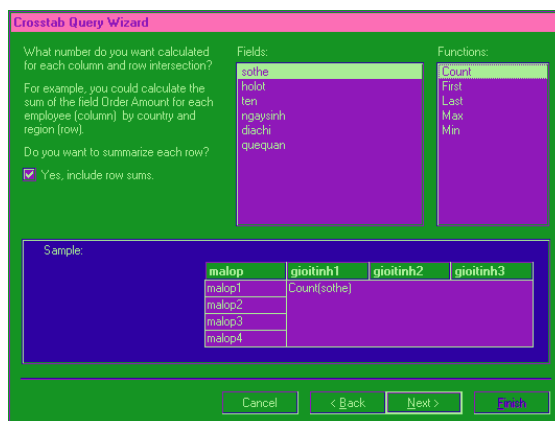


Bước 4: chọn tên trường làm tiêu đề dòng là MALOP, chọn Next

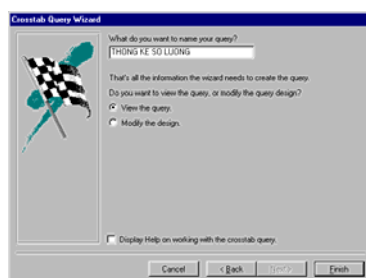


Bước 5: chọn tên trường làm tiêu đề cột là GIOITINH, sau đó chọn Next. Ta lưu ý tên trường làm tiêu đề dòng và cột là tên của trường sẽ được thống kê theo một kiểu tính nào đó. Trong trường hợp này ta đếm số lượng sinh viên theo từng lớp và trong mỗi lớp đếm số lượng nữ và nam là bao nhiêu.

Bước 6 :qui định công thức tính. Trong trường hợp này ta chọn hàm COUNT theo số thẻ. Sau đó chọn Next



Bước 7 :qui định tên của Query cần lưu trữ trên đĩa là THONG KE SO LUONG. Sau đó chọn Finish.





Bước 8: lúc này trên màn hình xuất hiện cửa sổ cho xem ố liệu sau khi đã thống kê như sau:

THONG KE SO LUONG : Crosstab Query				
	malog	Total Of sothe	-1	0
	93T	1		1
	94T	2	2	
▶	95T	1	1	

- Đóng cửa sổ trên.

Bước 9: nếu muốn thay đổi tiêu đề cột thì ta vào Design, thay chữ Total Of sothe bằng chữ Số lượng

**Bài tập:** từ Table điểm ta thực hiện các Query để : xem bảng điểm từng lớp, xem danh sách thi lại, thống kê số lượng theo xếp loại từng lớp.

## BÀI 6. LÀM VIỆC VỚI REPORT

### 6.1. Khái niệm

Report cho phép người sử dụng thiết kế các mẫu báo cáo theo yêu cầu để xem và in ấn các báo cáo đó ra giấy.

Report là một công cụ rất mạnh để chúng ta có thể tạo ra một báo cáo đẹp mắt. Ta dễ dàng điều chỉnh kích cỡ, kiểu dáng của mọi thành phần trong Report. Đa số các thông tin trong Report được lấy từ các Table, Query, các lệnh của SQL..

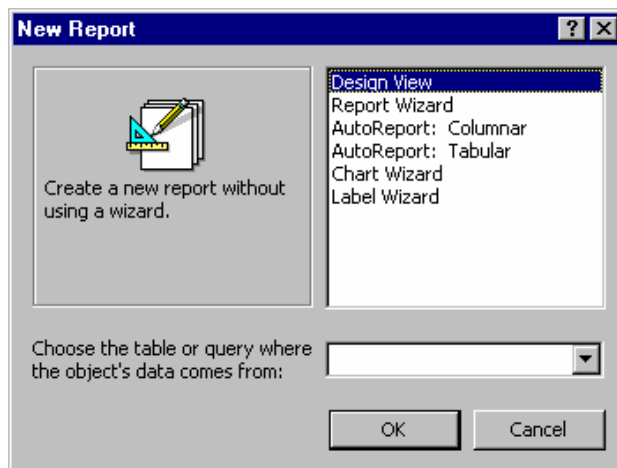
Có hai dạng Report chính là :

- Columnar Report : báo cáo dạng cột. Thường sử dụng để in các phiếu theo mẫu cho trước. Ví dụ : in lại phiếu thu, phiếu chi, phiếu báo điểm...
- Tabular Report : báo cáo dạng bảng. Đây là loại thường sử dụng để in các bảng kê. Trong trường hợp này các trường bố trí trên cột đứng, bản ghi bố trí trên dòng ngang.

### 6.2. Cách tạo Report

Bước 1 : chọn vào nút Report, tiếp đến chọn New

Bước 2 : chọn phương pháp và loại Report cần tạo qua cửa sổ :



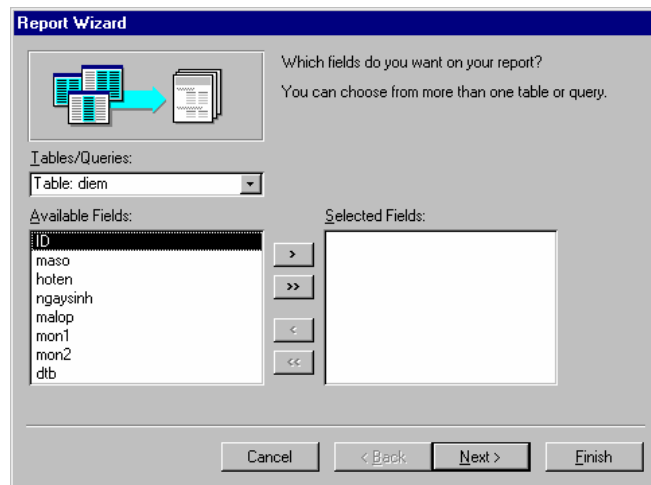
- Choose the table or query where the object's data comes from : chọn tên của Table hoặc Query chứa số liệu cơ sở của Report.
- Design View : tự thiết kế Report từ màn hình trắng.
- Report Wizard : thiết kế Report với sự trợ giúp của ACCESS.
- AutoReport - Columnar : tự động tạo báo cáo dạng cột.
- AutoReport - Tabular : tự động tạo báo cáo dạng bảng.
- Chart Wizard : tạo đồ thị mô tả với sự trợ giúp của ACCESS
- Label Wizard : tạo nhãn với sự trợ giúp của ACCESS.

Chọn OK để chuyển sang bước kế tiếp khai báo Report.

Sau đây giới thiệu phân thiết kế Report theo các kiểu chọn trên :

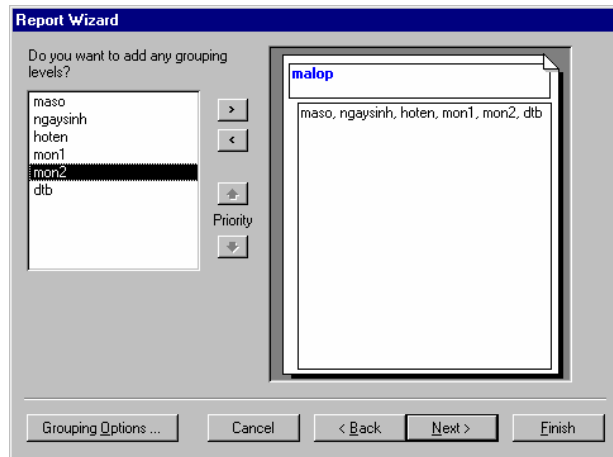
### **Report Wizard :**

Sau khi chọn Report Wizard xuất hiện cửa số :

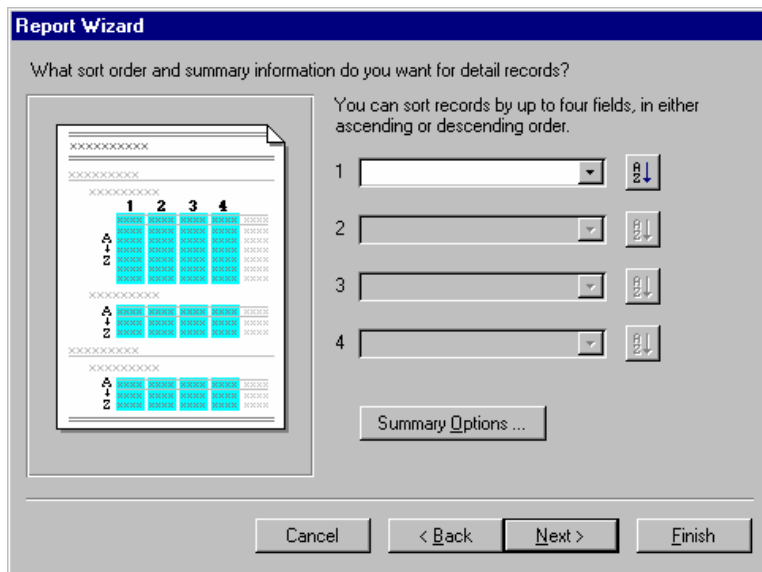


- Chọn tên các trường có nội dung cần xem trong Report từ Available Fields để chuyển sang Selected Fields.
- Chọn Next chuyển sang bước tiếp theo.

- Chọn tên trường cần dồn nhóm theo nó. Nếu không muốn kết nhóm thì bỏ qua bước này bằng cách chọn Next. Ví dụ : ta muốn in danh sách sinh viên trong trường theo từng nhóm là lớp thì chọn trường nhóm là lớp.

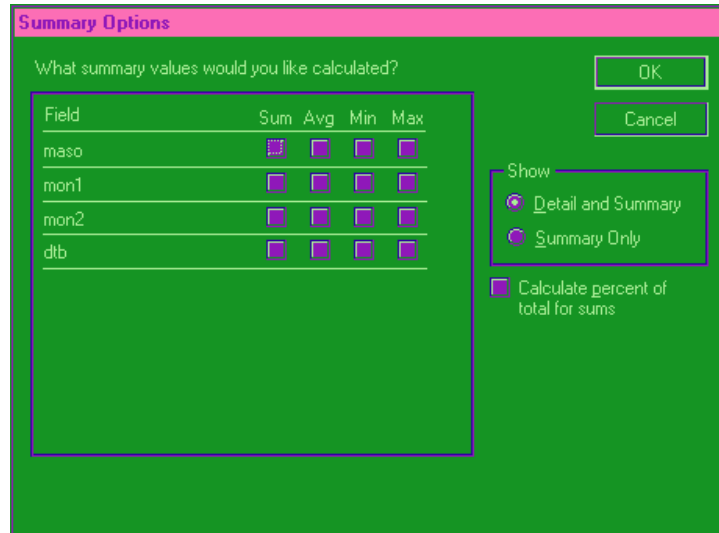


- Grouping Option : qui định thêm thông tin về phương pháp tạo nhóm.
- Chọn Next để chuyển sang bước tiếp theo :



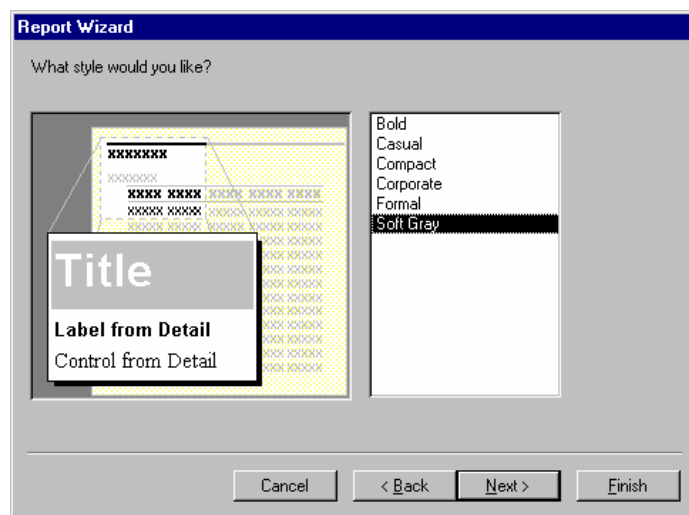
- You can sort records by up to four fields, in either ascending or descending order : qui định việc sắp xếp bản ghi theo thứ tự tăng hoặc giảm tối đa theo bốn khóa. Nếu muốn sắp xếp thì ta chọn tên trường khóa và qui định tăng hoặc giảm.

- Summary Option : Cho phép thực hiện tính toán theo các trường. Lúc đó xuất hiện mẫu khai báo :



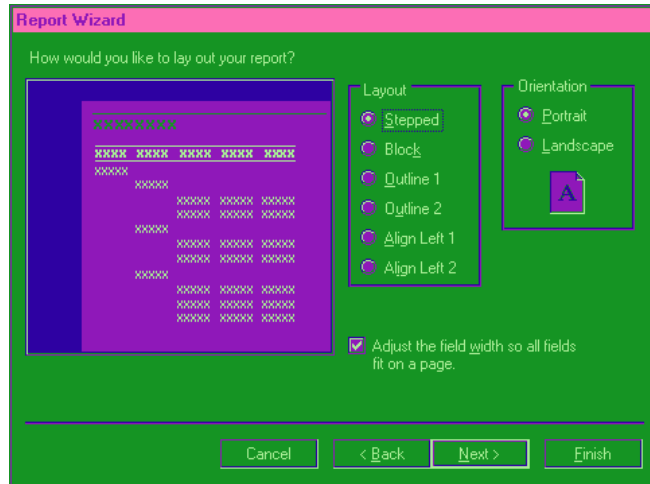
Lúc này ta phải qui định tên trường và công thức tính toán cho từng trường đó. Trong Table có bao nhiêu trường kiểu số thì có bấy nhiêu trường có thể được tính toán. Trong mục Show nếu chọn Detail and summary sẽ có các dòng chi tiết lẫn các dòng tóm tắt, nếu chọn Summary Only thì chỉ có các dòng tính tổng. Nếu chọn Calculate Percent thì sẽ có tính tính tỉ lệ phần trăm.

- Sau đó chọn Next để tiếp tục.



Qui định cách trình bày Report theo một trong 6 kiểu trên mục Layout. Mục Orientation cho phép qui định cách bố trí theo chiều ngang hay dọc của trang giấy.

- Chọn Next qua bước tiếp :



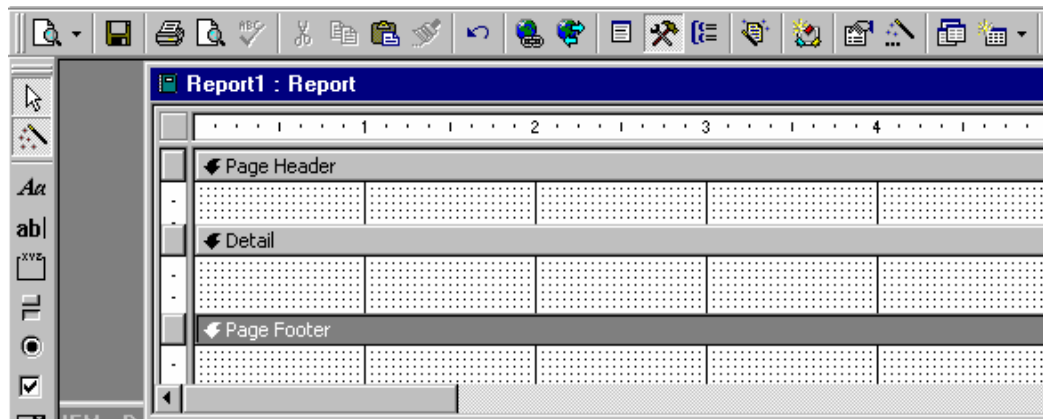
Qui định cách trình bày tiêu đề. Sau đó chọn Next để hoàn tất việc khai báo. Lúc này khai báo tên của Report để ghi vào đĩa và chọn Finish để hoàn tất.

**AutoReport Columnar** : tự động tạo ra một báo cáo dạng cột từ một Table cho trước.

**AutoReport Tabular** : tự động tạo ra một báo cáo dạng bảng từ một Table cho trước

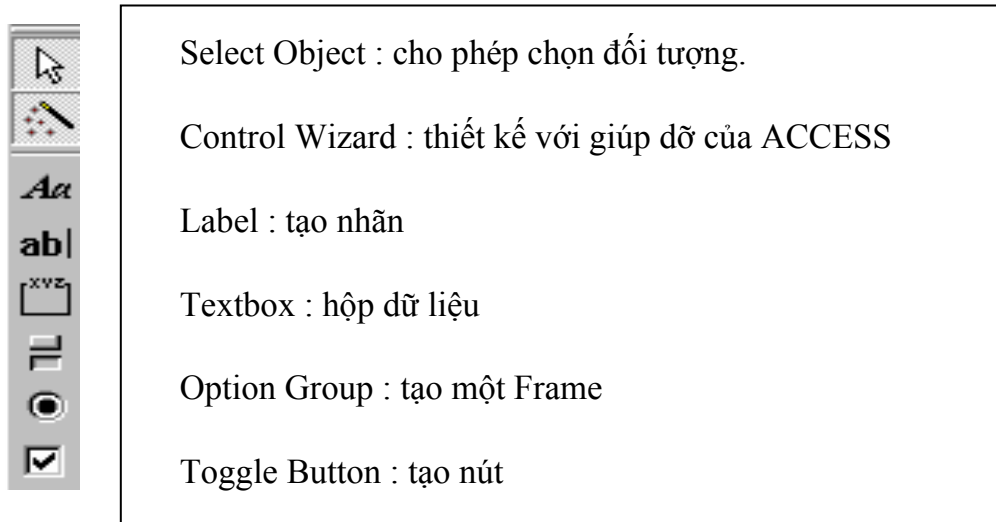
**Design View** : cho phép người sử dụng tự thiết kế một Report từ đầu đến cuối.

- Chọn tên của Table hoặc Query chứa dữ liệu cơ sở rồi chọn kiểu tự thiết kế Design View. Lúc đó trên màn hình xuất hiện cửa sổ cho phép thiết kế Report như sau :



- Page Header : cho phép ghi nội dung tiêu đề đầu mỗi trang in.

- Detail : ghi nội dung của các dòng trong Report.
- Page Footer : nội dung ở cuối mỗi trang in.
- Nếu muốn tạo tiêu đề được in một lần ở đầu báo cáo ta chọn trên thanh thực đơn Cột bên trái xuất hiện thanh công cụ để phục vụ người dùng thiết kế Report, ý nghĩa của chúng như sau :



Ngoài ra còn có nhiều công cụ khác.

### **6.3. Hiệu chỉnh Report**

Sau khi đã thiết kế và lưu trữ Report, nếu muốn hiệu chỉnh lại Report thì:

- Trong hộp Database đưa con trỏ đến tên Report cần hiệu chỉnh.
- Chọn Design.
- Hiệu chỉnh giống như trong Design View.

### **6.4. Thực hiện Report**

Nếu muốn xem hoặc in Report ta chọn trên thanh thực đơn :

- File → Print (để in) hoặc Print Preview (để xem)

Hoặc bấm vào các biểu tượng tương ứng trên thanh công cụ

## BÀI THỰC HÀNH.

Tạo Report để in danh sách một lớp tùy ý nhập từ bàn phím theo mẫu :

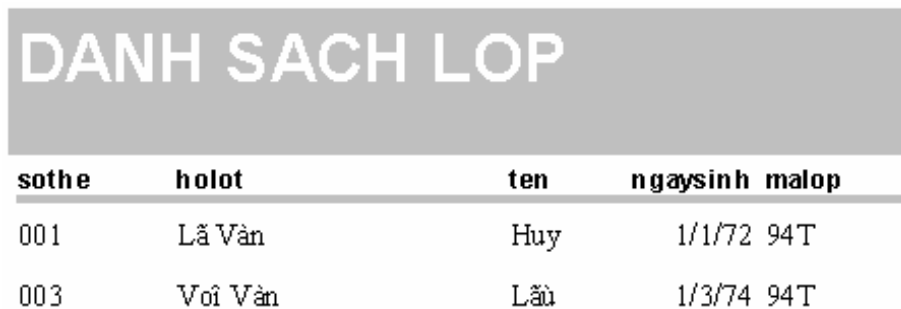
### DANH SÁCH LỚP

Số thẻ	Họ và tên	Ngày sinh	Ghi chú

Bước 1: bấm chuột vào nút Report, chọn New

Bước 2: chọn kiểu Report là AutoReport Tabular, tên dữ liệu gốc là Query DANH SACH LOP mà ta tạo trước đó trong bài thực hành về Query. Sau đó chọn OK

Bước 3: chờ một lát để máy tự động tạo ra Report theo kiểu ngầm định của ACCESS. Sau đó nó sẽ tự động thực hiện Report để cho ta xem Report vừa tạo



sothe	holot	ten	ngaysinh	malop
001	Lã Văn	Huy	1/1/72	94T
003	Voi Văn	Lâu	1/3/74	94T

xong. Lúc này ta nhập vào tên lớp để xem. Ví dụ ta nhập tên lớp là 94T lúc này trên màn hình sẽ xuất hiện báo cáo :

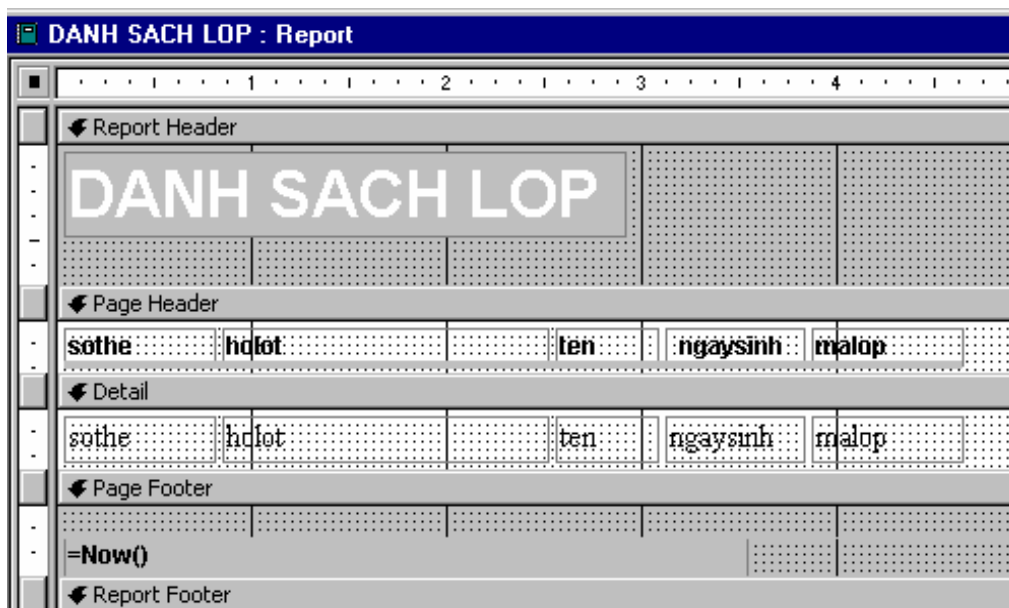
Bước 4: đóng cửa sổ và lưu Report vào đĩa với tên là DANH SACH LOP

Bước 5: chọn vào nút Design để hiệu chỉnh lại báo cáo theo qui định ở trên.

Lúc này ta phải Design lại mẫu cho giống như yêu cầu.



Tương tự ta thực hiện các Report khác như bảng điểm, danh sách thi lại, thống kê...



## **BÀI 7. LÀM VIỆC VỚI FORM**

### **7.1. Khái niệm :**

Form là môth công cụ cho phép người sử dụng thiết kế các mẫu nhập và xem số liệu giống như mẫu biểu ngoài thực tế.

Ta thấy khi nhập liệu bằng Table thì tuy việc nhập liệu rất dễ dàng nhưng các mẫu nhập của nó khác nhiều với trong thực tế tạo cảm giác khó chịu khi chưa quen. Vì vậy, khi xây dựng chương trình cho nhiều người sử dụng đặc biệt là những người không chuyên dùng máy tính thì ta phải tạo ra các mẫu biểu giống hệt trong thực tế để dễ dàng khi sử dụng.

Dữ liệu sử dụng trong Form được lấy từ Query hoặc Table

### **7.2. Thiết kế Form :**

Ta có thể tạo Form bằng nhiều cách khác nhau như :

- Dùng AutoForm : tự động tạo Form từ Table hoặc Query cho trước.
- Dùng Wizard : tạo Form qua sự giúp đỡ của ACCESS.
- Tự thiết kế Form

Trên thực tế để tạo Form một cách nhanh chóng, dễ dàng và đẹp mắt ta thường sử dụng Form Wizard để tạo ra mẫu Form và sau đó tự hiệu chỉnh lại những chỗ cần thiết.

#### **Sử dụng AutoForm :**

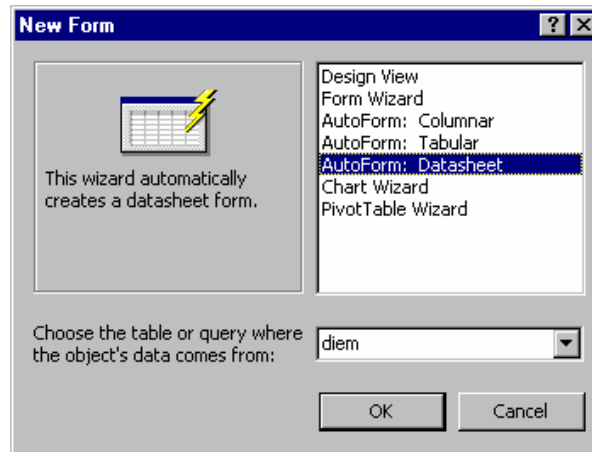
Auto Form cho phép tự động tạo Form với tất cả các trường trong Table hoặc Query.

Bước 1: Từ cửa sổ Database bấm vào nút Form sau đó chọn New

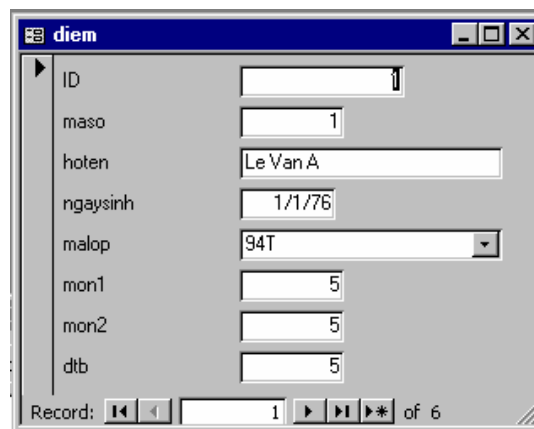
Bước 2: Xuất hiện cửa sổ :

- Chọn tên của Table hoặc Query chứa số liệu cơ sở của Form.

- Chọn loại Form và phương pháp tạo Form. Trong trường hợp này ta chọn AutoForm và kèm theo loại Form (Columnar : dạng cột, Tabular : dạng bảng và Datasheet : dạng bảng tính).



Bước 3: Form sẽ được tự động tạo ra trên cơ sở các trường của Table hoặc Query. Ví dụ : chọn Table là DIEM và loại Form là Columnar ta sẽ nhận được kết quả là Form sau :



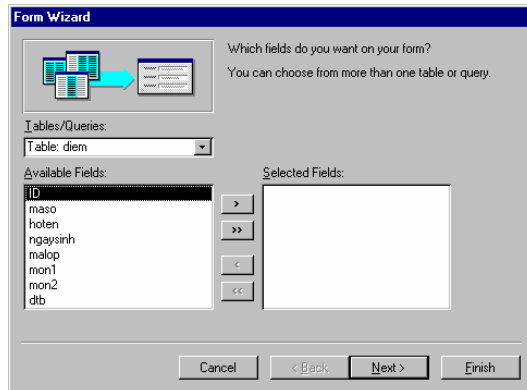
Bước 4: đóng Form bằng cách bấm vào nút có dấu X và ghi tên của Form để lưu trữ lên đĩa qua cửa sổ sau :



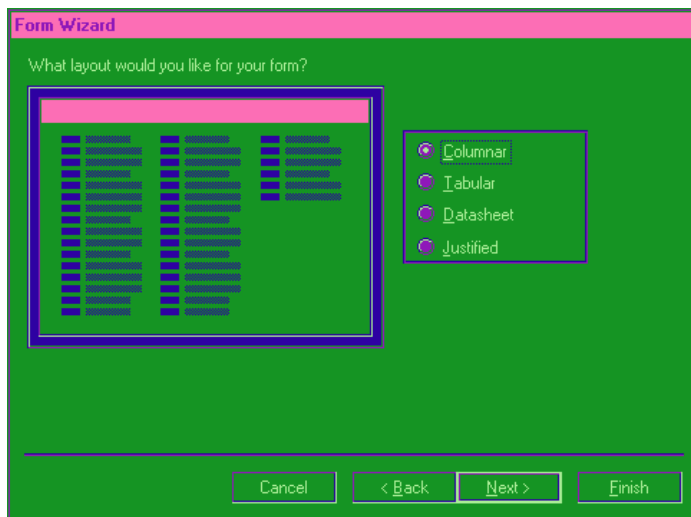
### **Sử dụng Form Wizard :**

Bước 1: chọn Form Wizard trên hộp thoại và tên Table hoặc Query chứa dữ liệu.

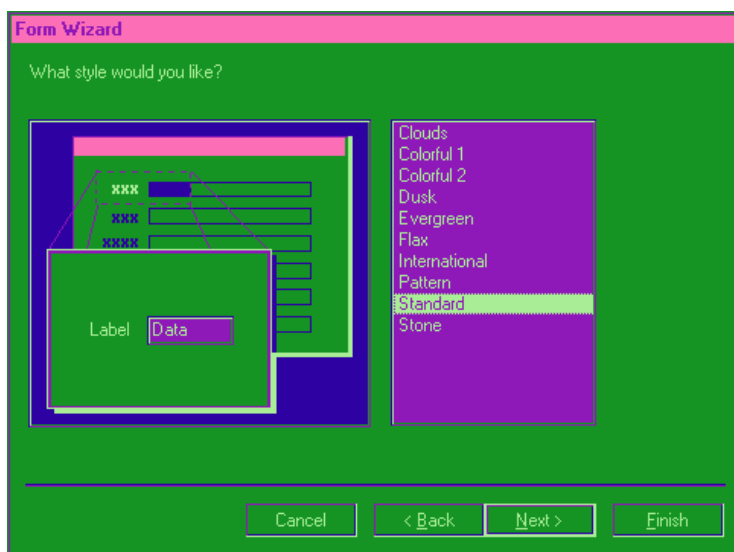
Bước 2: qui định tên các trường cần đưa vào Form rồi chọn Next qua cửa sổ sau :



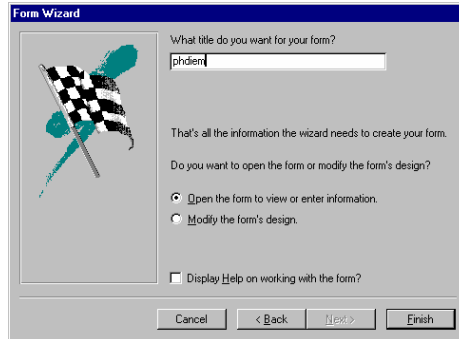
Bước 3: qui định loại Form theo các mẫu cho sẵn rồi chọn Next qua cửa sổ :



Bước 4: qui định cách thức trình bày Form rồi chọn Next qua cửa sổ:

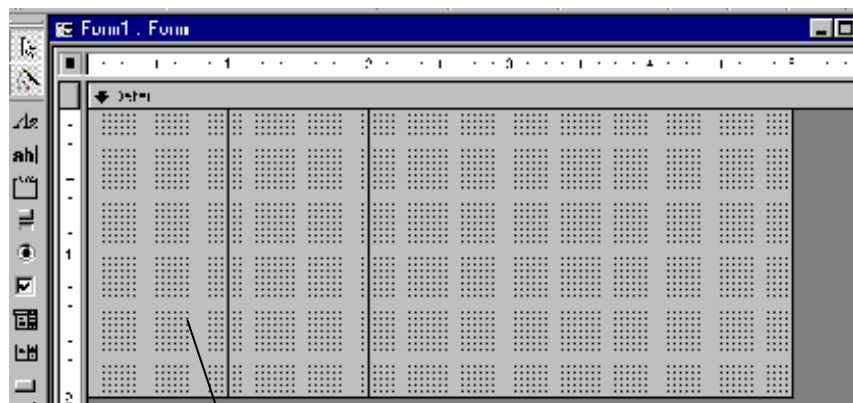


Bước 5: nhập vào tên của Form để lưu trữ lên đĩa rồi chọn Finish để hoàn tất việc thiết kế Form qua màn hình :



### **Tự thiết kế Form**

- Chọn tên Table hoặc Query, chọn kiểu thiết kế Form là Design View rồi OK.
- Xuất hiện màn hình để thiết kế Form như sau :



Màn hình chứa nội dung của Form

Thanh công cụ : chứa các công cụ để phục vụ cho việc thiết kế.

Tên gọi và ý nghĩa của các nút chọn trên thanh công cụ theo thứ tự từ trái sang phải như sau:



- Select : chọn đối tượng cần hiệu chỉnh.
- Control Wizard: trợ giúp của Wizard khi tạo Control.
- Label : tạo nhãn.

- TextBox: nội dung biểu thức hoặc trường.
- Option Group: nhóm chọn việc.
- Toggle Button: tạo nút bật tắt.
- Option Button: tạo nút chọn một trong nhiều giá trị.
- CheckBox: hộp đánh dấu để chọn nhiều giá trị cùng lúc.
- ComboBox: hộp chọn cặp.
- ListBox: hộp xem, chọn trong một danh sách.
- Command Button: tạo nút lệnh.
- Image: tranh ảnh.
- Unbound Object Frame: tạo một khung hình cố định.
- Bound Object Frame: tạo khung hình không cố định.
- Page Break: tạo dấu phân trang.
- Tab Control: tạo Tab điều khiển để chọn trang.
- SubForm: tạo Form con.
- Line: vẽ đường thẳng.
- Rectangle: vẽ hình chữ nhật.
- More Control: chọn sử dụng các nút điều khiển từ nhiều chương trình khác.

Muốn đưa một công cụ vào trong Form ta có thể tự thiết kế nút đó hoặc sử dụng Control Wizard của ACCESS :

a. Nếu dùng Control Wizard :

- Bấm nút Control Wizard (cho nó có màu sáng).
- Bấm vào công cụ cần chọn để đưa vào Form.

- Drag chuột vào trong Form tại khu vực cần đặt công cụ đó.
- Khai báo các thông tin cần thiết theo chỉ dẫn của ACCESS.

b. Tự thiết kế :

- Bấm chuột vào công cụ cần chọn để đưa vào Form.
- Drag chuột vào trong Form tại khu vực cần đặt công cụ đó.
- Khai báo các thông tin cần thiết. Nếu muốn sửa đổi các thuộc tính thì bấm đúp chuột tại công cụ vừa tạo để khai báo lại.

### **7.3. Hiệu chỉnh Form**

- Chọn tên của Form cần hiệu chỉnh, chọn Design
- Sửa đổi giống như trong phần tự thiết kế.

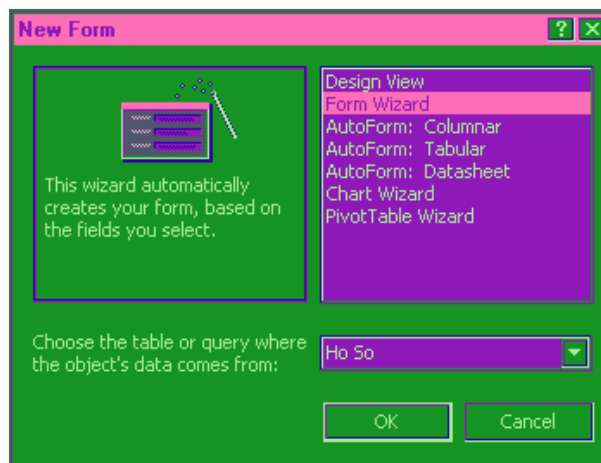
### **7.4. Thực hiện Form**

- Chọn tên của Form.
- Chọn Open

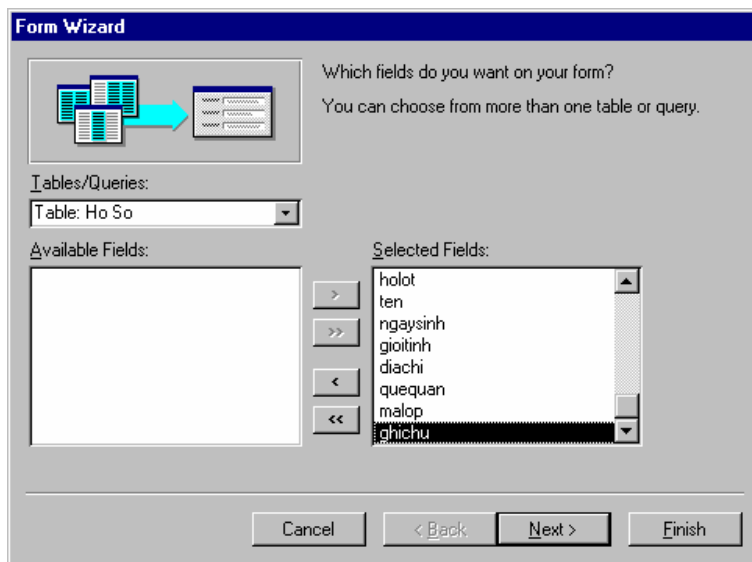
## BÀI THỰC HÀNH

Tạo Form để nhập hồ sơ.

Bước 1: bấm con chuột vào Form, chọn New



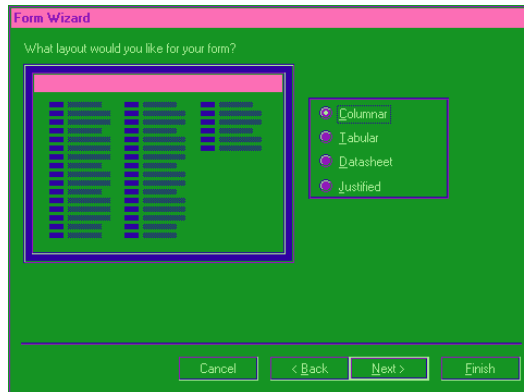
Bước 2: chọn vào mục Form Wizard, tên Table mà ta cần nhập hoặc xem số liệu trên đó là: Ho So. Sau đó chọn OK



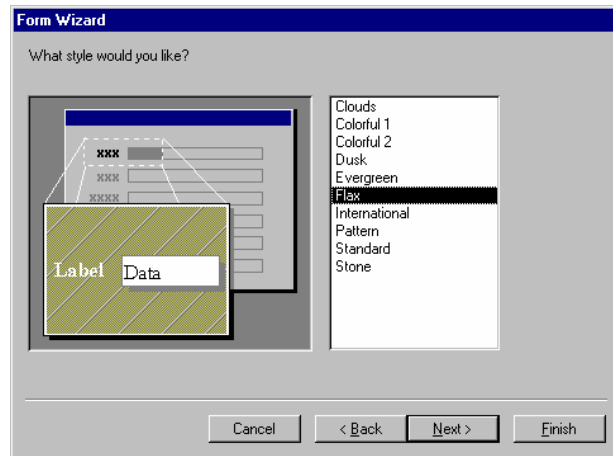
Bước 3: chọn tên các trường mà ta cần nhập hoặc xem. Trong trường hợp này ta bấm chuột vào >> để chọn tất cả các trường. Tiếp đến chọn Next

Bước 4: chọn cách nhập hoặc xem trên Form. Ta chọn Column, tiếp đến chọn Next

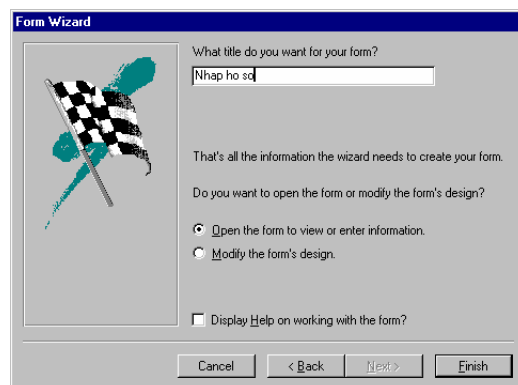




Bước 5: chọn màu sắc và cách thể hiện dữ liệu trên Form. Ta chọn Flax sau đó bấm vào Next để chuyển sang bước kế tiếp.



Bước 6: qui định tên của Form để lưu trữ lên đĩa. Chọn Finish để hoàn tất.



Bước 7: lúc này ACCESS tự động tạo ra Form để ta nhập và xem số liệu. Màn hình xem và nhập như sau :

Bước 8: đóng cửa sổ nhập và chọn Design để thiết kế lại Form cho nó đẹp và dễ sử dụng hơn.



- Đổi các tiêu đề thành tiếng Việt có dấu. Chọn Font chữ VN Times new roman.
- Thêm vào các nút bấm (Command Button) là : Mới, Thoát và D.sách.
- Mới : để mỗi khi bấm vào nút này ta sẽ có một màn hình trống để nhập vào bản ghi mới.

Cách làm :

- Bấm chuột chọn Control Wizard
  - Bấm chuột chọn Command Button
  - Drag chuột để chỉ định vị trí cần đặt nút bấm
  - Chọn Record Operations
  - Chọn AddNew Record, chọn Next
  - Qui định dòng ghi chú trên nút là Mới
  - Chọn Finish để hoàn tất công việc .
- Thoát : để mỗi khi bấm vào nút này ta sẽ kết thúc làm việc với Form.

Cách làm :

- Bấm chuột chọn Control Wizard
- Bấm chuột chọn Command Button
- Drag chuột để chỉ định vị trí cần đặt nút bấm

- Chọn Form Operations
  - Chọn Close Form, chọn Next
  - Qui định dòng ghi chú trên nút là Thoát
  - Chọn Finish để hoàn tất công việc .
- D.sách : để mỗi khi bấm vào nút này ta sẽ xem danh sách của một lớp tùy ý từ Report đã tạo trước đó.

Cách làm :

- Bấm chuột chọn Control Wizard
- Bấm chuột chọn Command Button
- Drag chuột để chỉ định vị trí cần đặt nút bấm
- Chọn Report Operations
- Chọn Preview Report, chọn Next
- Qui định tên Report cần mở là : DANH SACH LOP
- Qui định dòng ghi chú trên nút là D.Sách
- Chọn Finish để hoàn tất công việc .

Bước 9: đóng cửa sổ Design và chọn Yes để ghi lại phần vừa thiết kế.

Khi nào cần nhập hồ sơ ta chọn tên của Form và Open, lúc đó ta có màn hình nhập :

**NHẬP HỒ SƠ**

Số thẻ: 002      Mã lớp: 95T

Họ lót: Phan Huy      Ghi chú:

Tên: Chú

Ngày sinh: 3/20/74

Giới tính:

Địa chỉ: 12 Lê Lai

Quê quán: Đà Nẵng

Mới      D. Sách      Thoát

Record: 1 of 6

**Bài tập:** tương tự hãy thêm một số chức năng khác vào Form trên. Tạo các Form để nhập điểm, nhập danh mục lớp...

## BÀI 8. MACRO VÀ HỆ THỐNG THỰC ĐƠN

### 8.1. MACRO

#### 8.1.1 1. Khái niệm :

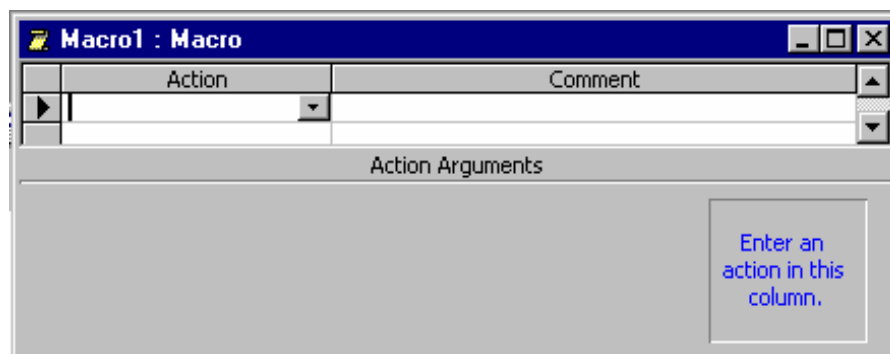
Macro là một hay một tập hợp các hành động (Action) liên tiếp được định nghĩa và lưu trữ với một tên xác định. Macro cho phép tự động hóa các công việc cần thực hiện.

Có ba loại Macro chính là :

- Macro kết hợp nhiều hành động : là Macro được kết hợp bởi nhiều hành động liên tiếp nhau. Khi tên Macro được gọi các hành động này sẽ lần lượt được tự động thực hiện.
- Macro Group : là một tập hợp các Macro có các tính năng giống nhau. Nó cho phép quản lý cơ sở dữ liệu dễ dàng hơn. Để thi hành một Macro trong Macro Group ta chỉ tên của nó như sau : Tên Macro Group.Tên Macro thực hiện.
- Macro theo điều kiện : là Macro mà các hành động chỉ được thi hành khi thỏa mãn điều kiện nào đó. Điều kiện là một biểu thức được chỉ định trong Condition.

#### 8.1.2 Cách tạo Macro

- Bước 1: trong cửa sổ Database chọn nút Macro, tiếp đến chọn New
- Bước 2 :xuất hiện cửa sổ để khai báo Macro như sau :



- Trong Action ta chọn một hành động cần thực hiện. Ta có thể chọn nhiều hành động tương ứng với nhiều dòng.
- Trong cột Comment ta có thể ghi rõ chú thích về hành động. Cột này không bắt buộc nhưng nó giúp người sử dụng dễ dàng khi bảo trì hệ thống vì biết được ý đồ thực hiện khi thiết kế.
- Trong mục Action Arguments ta có thể chỉ định các đối số cho Action nếu cần thiết.

### **8.1.3 Thực hiện Macro**

Để thực hiện Macro ta có thể chọn tên của Macro trong Database rồi chọn tiếp Open

Hoặc gọi tên Macro trong khi sử dụng Form, Report...

## **8.2. Hệ thống thực đơn**

Ta có thể sử dụng Macro để xây dựng hệ thống thực đơn cho phép lựa chọn công việc một cách dễ dàng và tiện lợi. Thông qua hệ thống thực đơn ta có thể liên kết tất cả các đối tượng trên Database thành một hệ thống chương trình thống nhất tiện lợi cho người sử dụng chương trình.

### **8.2.1 Cách tạo thực đơn:**

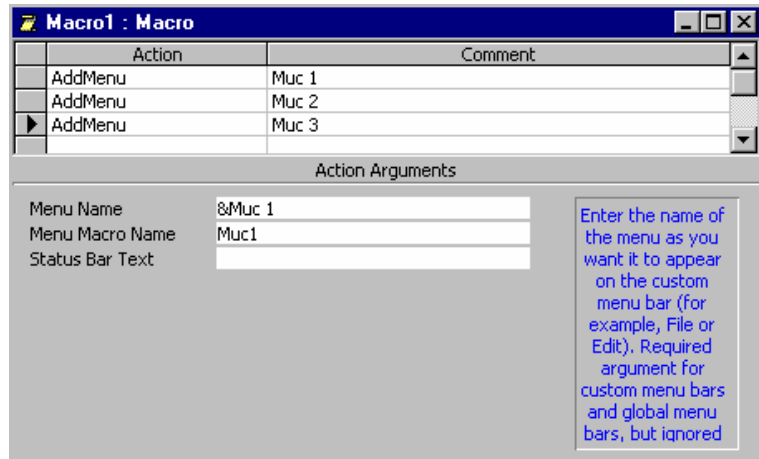
Giả sử ta muốn tạo một hệ thống thực đơn gồm các mục như sau :

Mục 1	Mục 2	Mục 3
Mục 1-1	Mục 2-1	Mục 3-1
Mục 1-2	Mục 2-2	Mục 3-2
...	...	...
Mục 1-n	Mục 2-n	Mục 3-n

Trong hệ thống thực đơn này các mục nằm ngang gọi là Menu cấp 1, mỗi cột đứng là một Menu cấp 2 (ta có 3 Menu cấp 2) và tương tự có thể tạo Menu các cấp thấp hơn (Ví dụ : chọn vào Mục 1-1 thì xuất hiện các mục Mục 1-1-1, Mục 1-1-2...).

**Bước 1:** tạo menu cấp 1.

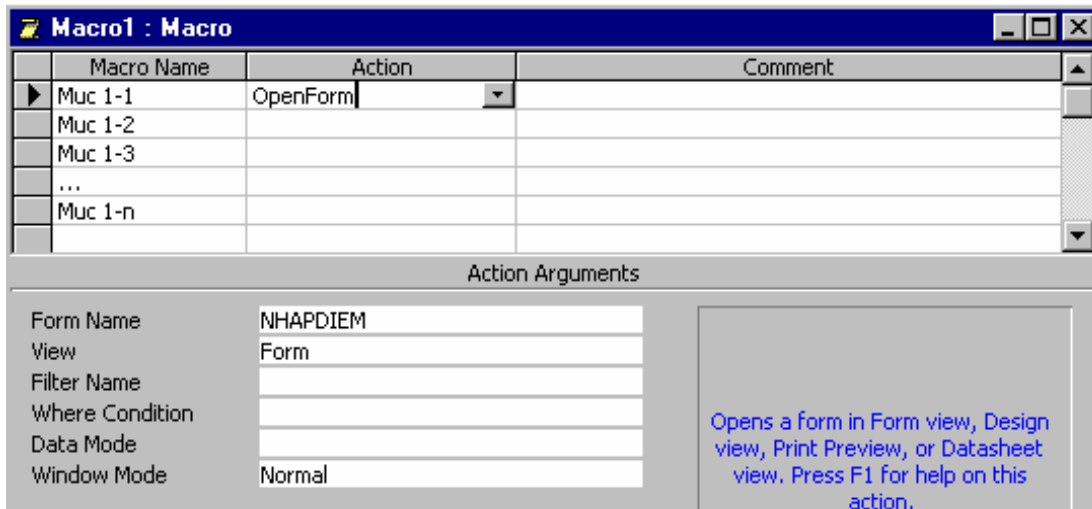
- Bấm dấu chuột vào nút Macro, chọn New.
- Khai báo vào bảng sau :



- Action : lựa chọn hành động là AddMenu cho cả ba
- Comment : ghi dòng chú thích. Mục này không cần.
- Menu Name : ghi nội dung dòng chữ sẽ hiện trên thanh thực đơn. Trong trường hợp này ta đặt tên là : Mục 1, Mục 2, Mục 3. Nếu muốn xuất hiện dấu gạch chân dưới chữ cái dùng làm phím nóng thì thêm vào trước chữ &
- Menu Macro Name : tên của Macro. Ta phải nhớ tên này để sau này gọi lại trong khi tạo menu cấp 2. Trong trường hợp ta đặt tên các Macro là : Muc1, Muc2, Muc3
- Status Bar Text : nội dung dòng chữ sẽ xuất hiện trên thanh Menu Bar khi ta chọn vào mục này.
- Ta đóng cửa sổ này bằng cách bấm chuột vào góc trên bên phải nơi có dấu X và đặt tên cho Macro là MainMenu (Tên này ta tự qui định).

**Bước 2:** tạo các menu cấp 2.

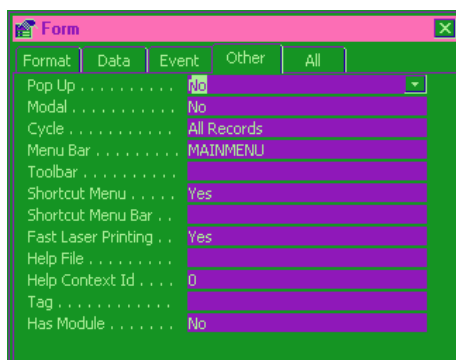
- Vào hộp Database chọn nút Macro rồi chọn New.
- Xuất hiện cửa sổ giống bước 1, ta chọn thêm View - Macro Name, sẽ xuất hiện cửa sổ mới như sau :



- Macro Name : gõ vào tên các mục trên Menu cấp hai thứ nhất. Những chữ này sẽ được in ra trên thanh thực đơn.
- Action : hành động cần thực hiện khi ta chọn vào chức năng này. Ta chọn các Action này trong danh sách mà ACCESS cho trước.
- Action Argument : khai báo các tham số liên quan đến Action.
- Đóng cửa sổ này và gõ vào tên của Macro để lưu trữ lên đĩa. Tên này phải trùng với tên của Macro (mà ta đã khai báo trong Menu Macro Name ở bước 1). Trong trường này ta gõ tên là Muc1.
- Tương tự, ta tạo hai Macro cấp 2 khác và đặt tên là Muc2, Muc3

**Bước 3:** gắn Menu lên một Form hoặc Report.

- Trong cửa sổ Database chọn Form (hoặc Report). Chọn New. Bấm chuột vào hộp Properties trên thanh Menu Bar để xuất hiện hộp thoại :





### **8.2.2 Sử dụng thực đơn**

Khi nào muốn dùng thực đơn chọn viện ta chỉ việc mở Form có gắn với thanh thực đơn được tạo.

## BÀI THỰC HÀNH

Tạo hệ thống thực đơn để nối các Form, Report ở các bài thực hành trước vào một ứng dụng chung.

Thực đơn như sau :

Nhập số liệu      Xem báo cáo Kết thúc

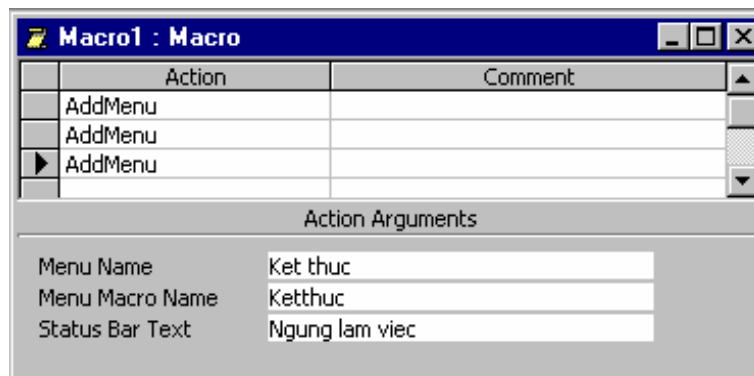
Hồ sơ              Danh sách lớp

Tổ chức lớp      Thống kê số lượng

... ..

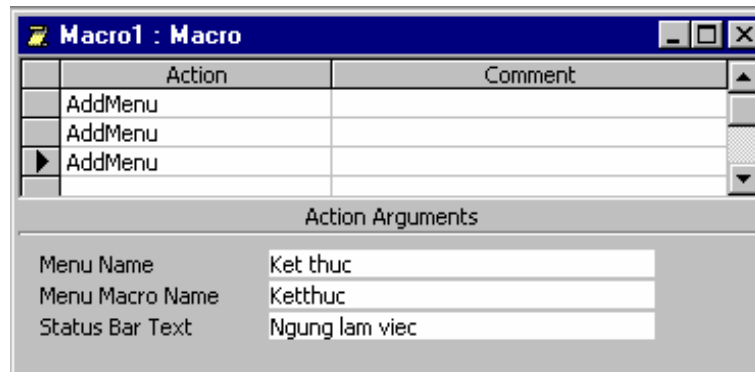
**Bước 1:** tạo Macro cho thực đơn nằm ngang..

- Trong cửa sổ Database chọn nút Macro, tiếp đến chọn New
- Xuất hiện cửa sổ để khai báo Macro dành cho thực đơn nằm ngang. Ta gõ vào tên các Macro con, dòng thông báo trên thanh thực đơn.



Ta khai vào các mục trên cửa sổ như sau :

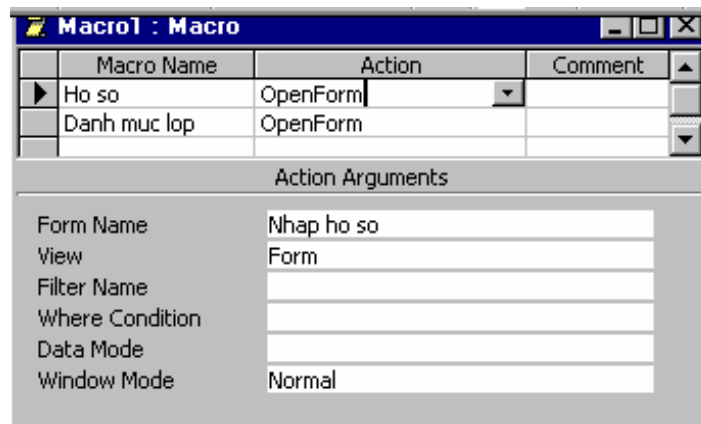
Action	Menu Name	Menu Macro Name
AddMenu	Nhap so lieu	Nhap
AddMenu	Xem bao cao	Baocao
AddMenu		
Ket thuc	Ketthuc	



- Đóng cửa sổ và ghi tên Macro là QUAN LY SINH VIEN

**Bước 2:** tạo các Macro đúng.

- Tạo Macro nhập :
  - Chọn Macro, New
  - chọn View, Macro Name để có màn hình khai báo sau :



Ta khai các mục như sau :

Macro Name	Action	Form Name	View
Ho sơ	OpenForm	Nhap ho so	Form
Danh muc lop	OpenForm	Nhap lop	Form

Đóng cửa sổ trên và ghi lại tên của Macro là Nhap (tên này giống như tên trong Macro qui định trong bước trước).

- Tạo Macro baocao :

- Chọn Macro, New
- Chọn View, Macro Name để có màn hình khai báo sau :

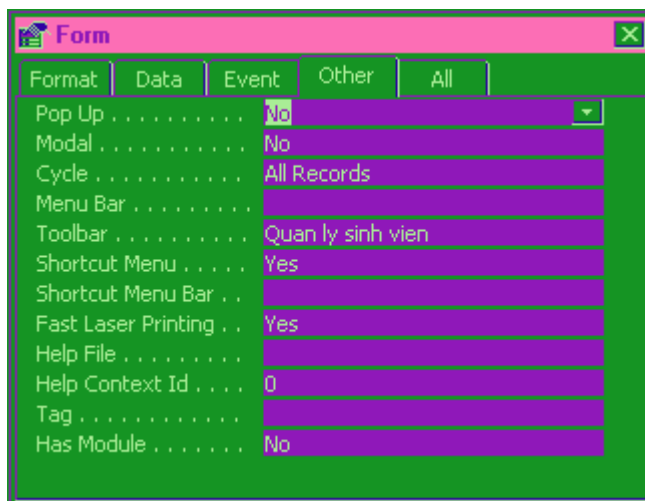
Macro Name	Action	Form Name	View
Danh sach lop	OpenReport	Danh sach lop	Print Preview
Thong ke so luong	OpenQuery	Thong ke so luong	Print Preview
...	...	..	..

Đóng cửa sổ và ghi với tên baocao

- Tạo Macro Ketthuc. Tương tự như trên nwgng ở mục Action ta chọn Close.

**Bước 3:** gắn Macro với một Form.

- Chọn Form, chọn New.
- Double Click chuột vào nền của Form để chọn Properties.
- Khai vào mục ToolBar tên của Macro ngang là : QUAN LY SINH VIEN



- Đóng cửa sổ tạo Form và đặt tên là MENU.
- Như vậy, từ đây về sau nếu muốn chọn công việc trên thanh thực đơn ta chỉ việc mở Form có tên là MENU

**Bài tập :** thêm vào thực đơn các chức năng khác.



**PHẦN II**

**VISUAL BASIC**







## **BÀI 9. MỞ ĐẦU**

### **9.1. Giới thiệu**

Ngôn ngữ BASIC (Beginner's All Purpose Symbolic Instruction Code) đã có từ năm 1964. BASIC rất dễ học và dễ dùng. Trong vòng 15 năm đầu, có rất nhiều chuyên gia Tin Học và công ty tạo các chương trình thông dịch (Interpreters) và biên dịch (Compilers) cho ngôn ngữ này làm BASIC trở nên rất phổ dụng.

Năm 1975, Microsoft tung ra thị trường sản phẩm đầu tay Microsoft BASIC và tiếp đó Quick BASIC (còn gọi là QBASIC) thành công rực rỡ. Quick BASIC phát triển trong nền Windows nhưng vẫn khó khăn khi tạo giao diện kiểu Windows. Sau đó nhiều năm, Microsoft bắt đầu tung ra một sản phẩm mới cho phép chúng ta kết hợp ngôn ngữ dễ học BASIC và môi trường phát triển lập trình với giao diện bằng hình ảnh (Graphic User Interface - GUI) trong Windows. Đó là Visual Basic Version 1.0 vào năm 1991.

Trước đó, chúng ta không có một giao diện bằng hình ảnh (GUI) với một IDE (Integrated Development Environment) giúp các chuyên gia lập trình tập trung công sức và thì giờ vào các khó khăn liên quan đến doanh nghiệp của mình. Mỗi người phải tự thiết kế giao diện qua thư viện có sẵn Windows API (Application Programming Interface) trong nền Windows. Điều này tạo ra những trở ngại không cần thiết làm phức tạp việc lập trình.

Visual Basic giúp chúng ta bỏ qua những khó khăn đó, các chuyên gia lập trình có thể tự vẽ cho mình giao diện cần thiết trong ứng dụng (application) một cách dễ dàng và như vậy, tập trung nỗ lực giải đáp các vấn đề cần giải quyết trong doanh nghiệp hay kỹ thuật.

Ngoài ra, còn nhiều công ty phụ phát triển thêm các thủ tục, hàm (modules), công cụ (tools, controls) hay ứng dụng (application) phụ giúp dưới hình thức VBX cộng thêm vào giao diện chính nên VB càng lúc càng thêm phong phú.

Khi Visual Basic phiên bản 3.0 được giới thiệu, thế giới lập trình lại thay đổi lần nữa. Với phiên bản này, chúng ta có thể thiết kế các ứng dụng (application) liên quan đến cơ sở dữ liệu (Database) trực tiếp tác động (interact) đến người dùng qua DAO (Data Access Object). Ứng dụng này thường gọi là ứng dụng trực diện (front-end application).

Phiên bản 4.0 và 5.0 mở rộng khả năng VB nhắm đến hệ điều hành Windows 95.

Phiên bản 6.0 cung ứng một phương pháp mới nối với cơ sở dữ liệu (Database) qua sự kết hợp của ADO (Active Data Object). ADO còn giúp các chuyên gia phát triển mạng nối với cơ sở dữ liệu (Database) khi dùng Active Server Pages (ASP).

Lưu ý ở đây, tất cả các khái niệm và công dụng của Modules, Tools, Controls, DAO, ADO hay ASP sẽ được trình bày trong các bài học sau. Tuy nhiên, VB phiên bản 6.0 (VB6) không cung ứng tất cả các đặc trưng của kiểu mẫu ngôn ngữ lập trình hướng đối tượng (Object Oriented Language - OOL) như các ngôn ngữ C++, Java.

Visual Basic là một ngôn ngữ lập trình trực quan và thường được sử dụng hiện nay. Giống như các ngôn ngữ khác, khi lập trình ta buộc phải tuân theo các qui tắc, trình tự Logic nhất định nhưng nếu so với các ngôn ngữ lập trình có cấu trúc như Turbo Pascal, C... thì Visual Basic đi theo một phương pháp lập trình mới. Visual Basic xây dựng một môi trường làm việc dưới dạng các biểu mẫu (Form), các hộp điều khiển (Control Box), thiên về các đối tượng (Object oriented), những thủ tục được xử lý theo tình huống và các phương thức (Method).

Khi làm việc với Visual Basic người lập trình có nhiệm vụ chính là thiết kế biểu mẫu, các khung giao diện, các nút lệnh và công việc sẽ thực hiện tương ứng trên đó; các lệnh, các chỉ thị phải được viết ra sẽ hạn chế tối đa.

Một trong những điểm khác biệt rõ ràng nhất giữa Visual Basic và các ngôn ngữ lập trình có cấu trúc là một ngôn ngữ xử lý theo tình huống (event - driven language) và một ngôn ngữ xử lý theo thủ tục (procedural - language).

Đối với các ngôn ngữ xử lý theo thủ tục thì một chương trình ứng dụng sẽ cho thi hành một cách Logic theo từng lệnh một tùy theo cách sắp xếp, tổ chức của người viết chương trình. Còn ngôn ngữ xử lý theo tình huống thì các chỉ thị chương trình chỉ được thực hiện khi gặp một tình huống đặc biệt xảy ra. Mỗi một tình huống tương ứng một thủ tục được thực hiện và các thủ tục này trong chương trình là hoàn toàn độc lập.

Visual Basic được xem là một ngôn ngữ lập trình xây dựng trên cơ sở của một phương pháp lập trình hiện đại được MicroSoft đưa vào thị trường vào năm 1991 với phiên bản 1.0. Trong hai năm sau đó thì lần lượt các phiên bản 2.0 và 3.0 ra đời. Hiện nay phiên bản 4.0 đang được sử dụng rộng rãi và có những thay đổi, bổ sung quan trọng so với các phiên bản trước.

### 9.2. Các khái niệm thường dùng

Đối tượng (Object) : là một tập hợp bao gồm chương trình và dữ liệu liên quan với nhau tạo thành một đơn vị xử lý độc lập. Khi khai báo đối tượng xong ta chỉ cần truyền cho nó các tham số cần thiết khi muốn đối tượng hoạt động. Khi đã hoàn tất khai báo, thực hiện thử để kiểm tra ta có thể lưu trữ đối tượng để sử dụng trong các chương trình khác.

Trong Visual Basic các đối tượng chính là biểu mẫu (FORM) và hộp điều khiển (CONTROL).

Biểu mẫu (Form) : là một khung cửa sổ hiện trên màn hình và nó có thể chứa một dãy các hộp điều khiển trên đó. Tất cả các dữ liệu muốn nhập, xem đều được trình bày trên biểu mẫu này.

Hộp điều khiển (Control Box) : là một đối tượng đặt trên Form, mỗi một hộp điều khiển sẽ tương ứng với một chức năng nào đó sẽ được thực hiện.

Mỗi một đối tượng ta có thể khai báo cho nó một số các thuộc tính riêng như màu sắc, kích thước, giá trị... và các thuộc tính này có thể thay đổi trong quá trình thực hiện chương trình.

Thủ tục tình huống (Event procedure) : là một dãy các chỉ thị lệnh và sẽ được tự động thực hiện khi xảy ra tình huống tương ứng. Một đối tượng có thể bao gồm nhiều thủ tục tình huống như vậy.

Phương thức (Method) : là các lệnh thao tác lên một đối tượng để thực hiện các xử lý theo yêu cầu nào đó (giống như một chương trình con). Mỗi phương thức sẽ mang một tên xác định và nếu muốn thực hiện phương thức ta viết như sau : <tên đối tượng>.<tên phương thức>[tham số]

Ví dụ : Form1.Print "In lên màn hình", trong đó đối tượng là Form1, phương thức là Print và tham số là nội dung "In lên màn hình".

### 9.3. Làm việc với Visual Basic

#### 9.3.1 Cài đặt :

- a. Yêu cầu về thiết bị:

- CPU AT386 trở lên.
- Ổ đĩa cứng còn trống 10MB trở lên.
- RAM 2MB trở lên.
- Màn hình màu EGA hoặc VGA.
- Con chuột.

b. Cài đặt:

- Khởi động WINDOWS.
- Chèn đĩa CD ROM vào ổ.
- Chọn File → Run, sau đó gõ D:\SETUP hoặc E:\SETUP (thông thường máy sẽ tự khởi động phần Setup)

### 9.3.2 Khởi động

Nếu trên máy tính đang sử dụng có cài đặt chương trình Visual Basic thì ta tiến hành khởi động như sau :

- Khởi động WINDOWS
- Double Click tại biểu tượng của Visual Basic.

### 9.3.3 Màn hình làm việc

Sau khi khởi động sẽ xuất hiện màn hình làm việc của Visual Basic với các cửa sổ chính như sau :

**Menu Bar   Tools Bar   Tools Box   Form Project   Properties**

a. Menu Bar (thanh thực đơn): ghi các chức năng lệnh của Visual Basic để người sử dụng có thể chọn thực hiện. Muốn chọn một chức năng trên đó ta có thể thực hiện :

- Cách 1 : Click chuột tại tên chức năng cần thực hiện.
- Cách 2 : Alt + <Chữ cái đầu>

- Cách 3 : F10, chọn và gõ Enter tại chức năng đó.

b. Tools Bar (thanh các công cụ): chứa các biểu tượng và mỗi biểu tượng sẽ tương ứng với một lệnh được thực hiện. Đây là những lệnh thường được sử dụng trong Visual Basic. Muốn chọn một lệnh nào đó ta chỉ cần Double Click tại biểu tượng tương ứng.

c. Tools Box (hộp công cụ): chứa các biểu tượng và mỗi biểu tượng tương ứng với một hộp điều khiển (Control Box). Nó cho phép chọn hộp điều khiển để đưa vào biểu mẫu trong quá trình thiết kế. Muốn đưa một hộp điều khiển vào biểu mẫu ta thực hiện qua các bước sau :

- Click chuột tại biểu tượng tương ứng.
- Drag chuột tại khu vực cần đặt hộp điều khiển trên Form.
- Khai báo các thuộc tính và thủ tục tương ứng.

d. Form Window (cửa sổ biểu mẫu): là cửa sổ dành để thiết kế chương trình. Trên này ta sẽ đặt các hộp điều khiển để xem, nhập số liệu hoặc các nút để chọn thủ tục cần thực hiện.

e. Project Window (cửa sổ dự án) : là cửa sổ liệt kê tên các tập tin, biểu mẫu và các đơn thể chương trình thuộc ứng dụng hiện hành. Ta có thể xem một biểu mẫu hoặc bộ mã lệnh chương trình nếu Double Click tại tên tương ứng.

f. Properties (cửa sổ thuộc tính): là cửa sổ cho xem các thuộc tính gắn liền với một đối tượng.

### 9.3.4 Kết thúc

Cho phép ngừng làm việc với Visual Basic và quay về lại môi trường WINDOWS.

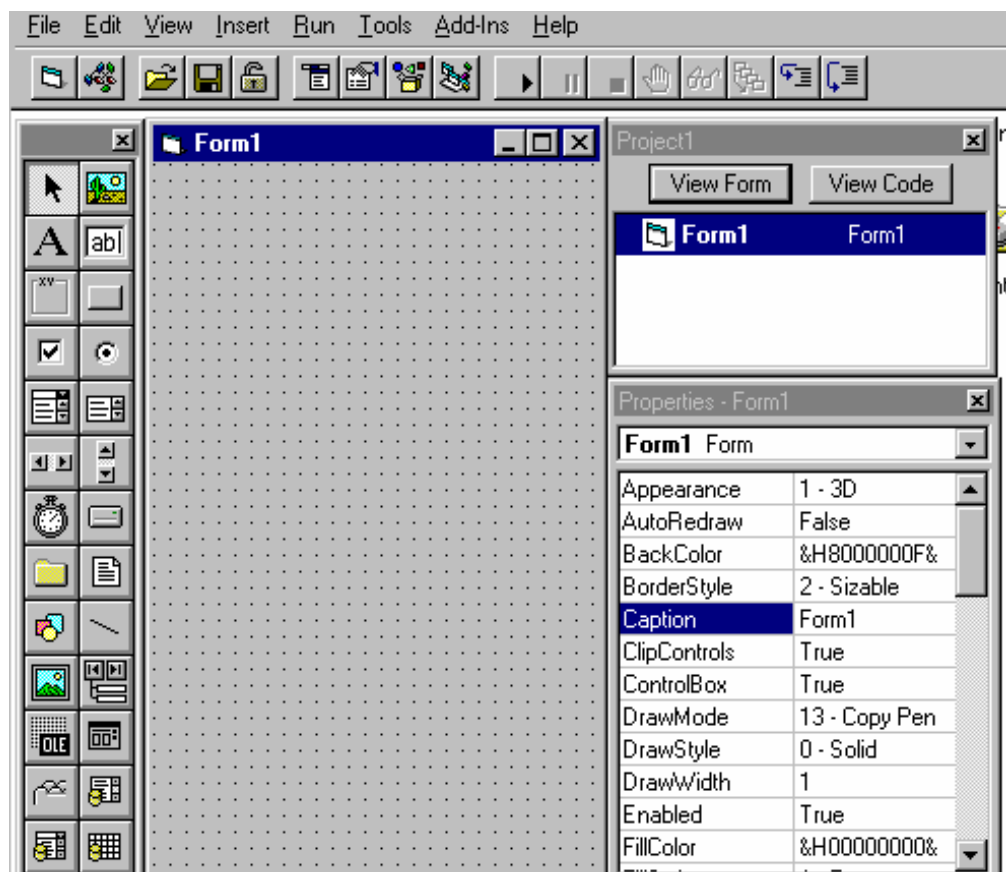
- Chọn File.
- Chọn Exit.

## BÀI 10. LẬP TRÌNH TRONG VISUAL BASIC

Khi lập trình trong Visual Basic, mọi ứng dụng đều thường bắt đầu bằng biểu mẫu (Form), đây là nơi hiển thị tất cả các thông tin liên quan đến ứng dụng.

Khi thiết kế chương trình ta thường qua các bước :

- Gắn các hộp điều khiển vào biểu mẫu.
- Thay đổi thuộc tính của hộp điều khiển.
- Viết thủ tục tình huống cho hộp điều khiển đó.

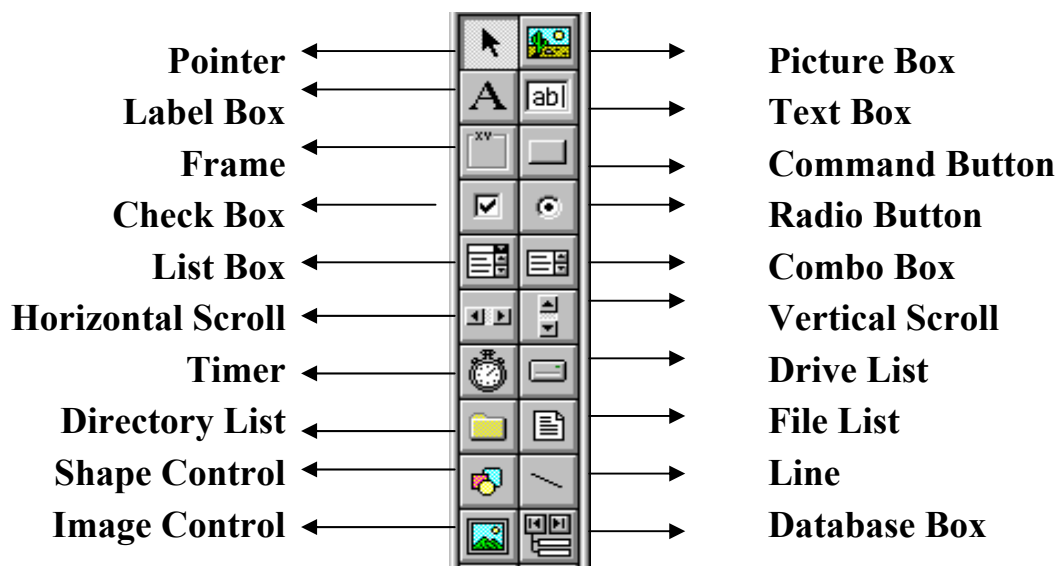


- Trong phần này ta sẽ giới thiệu tổng quát về các bước trên.

## 10.1. Làm việc với hộp điều khiển

Hộp điều khiển là thành phần cơ bản nhất của biểu mẫu, nó quyết định hình dạng và hoạt động của ứng dụng. Một ứng dụng có linh hoạt, cung cấp đủ tiện nghi cho người dùng hay không sẽ phụ thuộc phần lớn vào việc sử dụng và khai thác các hộp điều khiển.

10.1.1 Các loại hộp điều khiển : trên thanh Tools Bar có các nút điều khiển thường sử dụng như :



Ý nghĩa các nút điều khiển như sau :

Tên gọi	Ý nghĩa
Pointer	Di chuyển điểm đặt
Label Box	Ghi nội dung dòng văn bản. Nội dung cố định.
Frame	Tạo khung hình chữ nhật chứa nhiều hộp điều khiển
Check Box	Dùng chọn giá trị True hoặc False, nhiều hơn một mục
List Box	Liệt kê các giá trị để chọn lựa
Horizontal Scroll	Thanh cuộn ngang để thay đổi phần màn hình cần xem
Timer	Dùng bẫy tình huống theo thời gian. Kiểm tra quá hạn
Directory List	Liệt kê tên các thư mục
Shape Control	Cung cấp công cụ để vẽ hình.
Image Control	Hiển thị hình ảnh trong tập tin Bitmap





### 10.1.3 Hiệu chỉnh hộp điều khiển :

- Thay đổi vị trí : Drag chuột tại hộp điều khiển trên biểu mẫu.
- Thay đổi kích thước : bấm chuột vào hộp, đưa dấu chuột về góc phải bên dưới rồi Drag chuột.
- Xóa hộp điều khiển : bấm chuột lên hộp rồi gõ phím DELETE.
- Sao chép : Ctrl + C để chép nút vào bộ nhớ.
- Dán : Ctrl + V để dán nút từ bộ nhớ ra màn hình.
- Nhóm các hộp thành một khối : giữ phím Ctrl đồng thời Click chuột lần lượt tại các hộp cần nhóm lại với nhau.

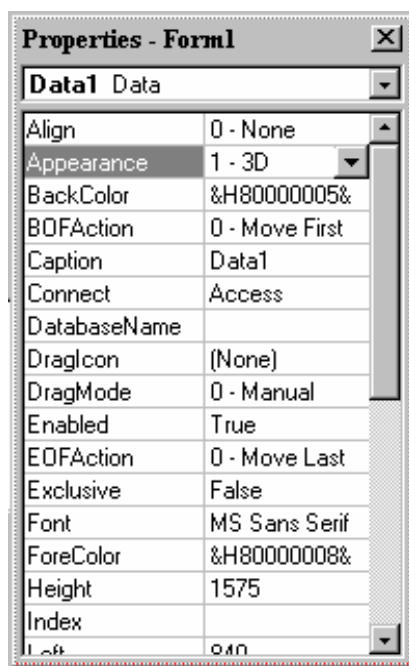
## 10.2. THUỘC TÍNH

Thuộc tính là tập hợp các mô tả về đối tượng mà người sử dụng có thể thay đổi được. Ta có thể thay đổi các thuộc tính khi thiết kế biểu mẫu hoặc trong quá trình thực hiện chương trình.

Thông thường, khi thiết kế biểu mẫu ta phải khai báo các thuộc tính có sẵn và ít bị thay đổi. Sau đó khi thực hiện chương trình nếu muốn thay đổi các tham số mô tả thuộc tính thì ta tiếp tục thực hiện các thay đổi đó.

### 10.2.1 Khi thiết kế :

- Click chuột vào đối tượng (hộp điều khiển hoặc biểu mẫu) mà ta cần thay đổi thuộc tính.
- Gọi cửa sổ Properties. (nhấn phím F4 hoặc chọn Window - Properties).
- Xuất hiện cửa sổ chứa các thuộc tính như bên. Ta thực hiện thay đổi các thuộc tính theo yêu cầu.
- Cửa sổ thuộc tính :



### 10.2.2 Khi thực hiện chương trình

Muốn thay đổi thuộc tính trong quá trình thực hiện chương trình ta viết trong chương trình dòng lệnh:

[OBJECT.]<PROPERTY> = <NEW VALUE>

### 10.2.3 Các loại thuộc tính :

Khi làm việc với các đối tượng ta thường sử dụng các thuộc tính sau :

- **BorderStyle** : qui định dạng xuất hiện của đường viền biểu mẫu. Ta chọn :
  - 0 (none) : không có đường viền. Biểu mẫu không có các nút Minimize, Maximize và hộp trình đơn điều khiển. Không được xô dịch và thay đổi kích thước biểu mẫu.
  - 1 (Fixed Single) : đường viền là gạch đơn. Có các nút thu phóng kích thước. Không thay đổi ích thước.
  - 2 (Sizeble) : cho phép thay đổi kích thước.
  - 3 (Fixed Double): đường viền nét đôi, cố định vị trí.
- **Caption** : thay đổi nội dung dòng chú thích sẽ hiện lên tại đối tượng.

- Control Box : qui định sự có mặt hay không của thanh trình đơn điều khiển trên biểu mẫu. Thuộc tính này không thay đổi được khi chạy chương trình.
- MinButton và MaxButton : qui định có các nút phóng to (Maximum) hoặc thu nhỏ (Minimum) trên biểu mẫu hay không. Chọn giá trị True hoặc False.
- Enable : qui định việc bật (nếu giá trị True) hoặc tắt (giá trị False) sự hoạt động của nút điều khiển hoặc biểu mẫu.
- Name : khai báo tên của đối tượng. Tên này giống như tên biến, nó sẽ được sử dụng khi cần làm việc với các đối tượng trong các đoạn mã chương trình.
- Font : thay đổi kiểu chữ. Ta có thể sử dụng các thuộc tính sau liên quan đến Font chữ: FontName (tên kiểu chữ), Font Size (kích thước chữ), FontBold (chữ đậm), FontItalic (chữ nghiêng), FontStrikethru (gạch xóa), FontTransparent (nếu True thì có nền), FontUnderline (gạch chân).
- Height : thay đổi chiều cao của đối tượng.
- Width : thay đổi độ rộng của đối tượng
- Left : chuyển dịch đối tượng theo phương ngang.
- Top : chuyển dịch đối tượng theo phương đứng.
- ScaleMode : qui định đơn vị đo lường.
  - 0 : cho biết đơn vị do người dùng ấn định.
  - 1 : trị ngầm định là Twip (1440 Twips = 1 inch).
  - 2 : Point (72 Points = 1 inch).
  - 3 : Pixed (đơn vị đo nhỏ nhất theo độ phân giải màn hình).
  - 4 : Ký tự.
  - 5 : Inch.
  - 6 : mm

- 7 : cm

- Icon : xác định biểu tượng sẽ được hiển thị vào lúc chạy chương trình khi biểu mẫu này bị thu nhỏ.
- MousePointer : qui định hình dạng của dấu chuột.
- Visible : qui định biểu mẫu được bật hay tắt.
- WindowState : qui định trạng thái của cửa sổ. (0 : bình thường), 1 (thu nhỏ thành biểu tượng) và 2 (phóng to tối đa).
- BackColor : qui định màu nền.
- ForeColor : qui định màu chữ.

Ngoài ra, còn có nhiều thuộc tính khác nhưng ít được dùng. Trong các phần sau nếu gặp sẽ giải thích thêm.

### 10.3. Thủ tục tình huống:

Mỗi một đối tượng (biểu mẫu hoặc hộp điều khiển) đều có thể kèm theo một thủ tục để khi ta kích vào đối tượng này thì thủ tục sẽ được thực hiện. Các thủ tục theo tình huống này tương tự như các Valid trong FOXPRO.

- Cách gọi thủ tục tình huống :

**Cách 1 :** Double Click chuột vào đối tượng tương ứng.

**Cách 2 :** Click chuột vào đối tượng sau đó nhấn phím F7 (hoặc chọn View - Code).

- Cách viết : các chỉ thị được viết vào giữa Sub .... End Sub

#### Chú ý :

- Sau khi đã thiết kế xong biểu mẫu ta lưu trữ ứng dụng vào đĩa bằng cách chọn File - Save rồi đặt tên cho biểu mẫu.
- Muốn thực hiện biểu mẫu ta chọn Run – Start



## **10.4. Thay đổi thuộc tính :**

### 10.4.1 Hộp Text :

- Click chuột vào hộp Text1, gõ phím F4.
- Đổi các thuộc tính :
  - MultiLine : True.
  - ScrollBar : Both.
  - Name : txt

### 10.4.2 Các hộp Command Button :

- Lần lượt Click chuột vào các hộp Command1, Command2, Command3 và Command4 và nhấn phím F4.
- Sau đó, ở mỗi lần ta thay đổi :
  - Caption : Cắt, Sao chép, Dán và Xóa (đổi tên tương ứng).
  - Name : cmdcut, cmdcopy, cmdpaste và cmddele

### 10.4.3 Các hộp Check Box :

- Lần lượt Click chuột vào các hộp Check 1, Check 2, Check 3 và nhấn phím F4.
- Sau đó, ở mỗi lần ta thay đổi :
  - Caption : In đậm, In nghiêng và Gạch chân (đổi tên tương ứng).
  - Name : chkbold, chkitalic, chkunder.

### 10.4.4 Đổi Font :

- Giữ phím Control đồng thời click chuột tại tất cả các đối tượng để tạo nhóm đối tượng.
- Gõ phím F4 gọi cửa sổ Properties.
- Đổi Font sang tên là Vntimes New Roman, Size 10.

## 10.5. Viết các thủ tục tình huống :

10.5.1 Thủ tục của Form : đây là thủ tục chứa các chỉ thị khởi tạo giá trị ban đầu.

- Bấm Double Click chuột vào nền của Form.
- Gõ vào nội dung chương trình như sau :

```
Private Sub Form_Load()  
    txt = ""           'Khởi tạo biến trắng  
    txt.FontBold = False       'Không đậm  
    txt.FontItalic = False  
    txt.FontUnderline = False  
    chkbold.Value = 0         'Chưa chọn in đậm  
    chkitalic.Value = 0  
    chkunder.Value = 0  
End Sub
```

- Sau đó vào hộp Object chọn tiếp General để gõ vào chỉ thị khai báo biến nhớ ClipText :

```
Private Sub Text1_Change()  
    Dim ClipText As String     'Khởi tạo biến chuỗi tên là ClipText  
End Sub
```

10.5.2 Thủ tục của các hộp Command :

- Lần lượt Double Click chuột tại các hộp Cắt, Sao chép, Dán và Xóa và gõ vào các thủ tục tương ứng sau :

```
Private Sub cmdcut_Click()  
    ClipText = txt.SelText     'Chép khối vào biến tạm ClipText  
    txt.SelText = ""         'Xóa khối khỏi văn bản  
    txt.SetFocus             'Tham chiếu đến biến TXT  
End Sub  
Private Sub cmdcopy_Click()  
    ClipText = txt.SelText  
    txt.SetFocus  
End Sub  
Private Sub cmdpaste_Click()  
    txt.SelText = ClipText     'Gán khối bởi giá trị biến ClipText  
    txt.SetFocus
```

```
End Sub
Private Sub cmdDel_Click()
    txt.Text = ""
    txt.SetFocus
End Sub
```

### 10.5.3 Thủ tục của các hộp Check Box :

- Lần lượt Double Click chuột tại các hộp In đậm, In nghiêng, Gạch chân và gõ vào các thủ tục sau :

```
Private Sub chkBold_Click()
    txt.FontBold = Not (txt.FontBold)      'Ngược lại với giá trị cũ
    txt.SetFocus
End Sub
Private Sub chkItalic_Click()
    txt.FontItalic = Not (txt.FontItalic)
    txt.SetFocus
End Sub
Private Sub chkUnder_Click()
    txt.FontUnderline = Not (txt.FontUnderline)
    txt.SetFocus
End Sub
```

## 10.6. Ghi và thực hiện chương trình :

- Chọn File - Save để lưu trữ biểu mẫu vào đĩa với tên là VANBAN
- Chọn Run - Start để bắt đầu thực hiện chương trình.

**Chú ý :** nếu muốn lưu trữ và xem nội dung chương trình dưới dạng văn bản thì ta thực hiện như sau :

### 10.6.1 Lưu trữ :

- Chọn File.
- Chọn File Save As
- Chọn kiểu tập tin là TEXT, đặt tên là Vanban.TXT.



### 10.6.2 Xem mã lệnh :

- Vào một chương trình soạn thảo văn bản bất kỳ.
- Gọi văn bản dạng TEXT có tên Vanban.TXT.
- Lúc này ta có nội dung chương trình như sau :

```
VERSION 4.00
Begin VB.Form Form1
    Caption           =   "Form1"
    ClientHeight      =   3390
    ClientLeft        =   1140
    ClientTop         =   1665
    ClientWidth       =   6495
    Height            =   3870
    Left              =   1080
    LinkTopic         =   "Form1"
    ScaleHeight       =   3390
    ScaleWidth        =   6495
    Top               =   1245
    Width             =   6615
Begin VB.Frame Frame1
    Caption           =   "Frame1"
    BeginProperty Font
        name            =   "VNtimes new roman"
        charset         =   1
        weight          =   400
        size            =   9.75
        underline       =   0    'False
        italic          =   0    'False
        strikethrough   =   0    'False
    EndProperty
    Height            =   2055
    Left              =   4680
    TabIndex          =   5
    Top               =   120
    Width             =   1695
Begin VB.CheckBox chkunder
```

```
    Caption          =   "Gạch chân"
BeginProperty Font
    name              =   "VNTimes new roman"
    charset           =   1
    weight            =   400
    size              =   9.75
    underline         =   0   'False
    italic            =   0   'False
    strikethrough    =   0   'False
EndProperty
Height              =   375
Left                =   240
TabIndex           =   8
Top                 =   1560
Width               =   1215
End
Begin VB.CheckBox chkitalic
    Caption          =   "In nghiêng"
BeginProperty Font
    name              =   "VNTimes new roman"
    charset           =   1
    weight            =   400
    size              =   9.75
    underline         =   0   'False
    italic            =   0   'False
    strikethrough    =   0   'False
EndProperty
Height              =   375
Left                =   240
TabIndex           =   7
Top                 =   960
Width               =   1215
End
Begin VB.CheckBox chkbold
    Caption          =   "In đậm"
BeginProperty Font
    name              =   "VNTimes new roman"
    charset           =   1
```

```
        weight          = 400
        size            = 9.75
        underline       = 0   'False
        italic          = 0   'False
        strikethrough   = 0   'False
    EndProperty
    Height              = 375
    Left                = 240
    TabIndex            = 6
    Top                 = 360
    Width               = 1215
End
End
Begin VB.CommandButton cmddel
    Caption              = "Xóa"
    BeginProperty Font
        name              = "VNtimes new roman"
        charset           = 1
        weight            = 400
        size              = 9.75
        underline         = 0   'False
        italic            = 0   'False
        strikethrough     = 0   'False
    EndProperty
    Height              = 495
    Left                = 4920
    TabIndex            = 4
    Top                 = 2520
    Width               = 1215
End
Begin VB.CommandButton cmdpaste
    Caption              = "Dán"
    BeginProperty Font
        name              = "VNtimes new roman"
        charset           = 1
        weight            = 400
        size              = 9.75
        underline         = 0   'False
```

```
        italic           = 0   'False
        strikethrough    = 0   'False
    EndProperty
    Height               = 495
    Left                 = 3480
    TabIndex             = 3
    Top                  = 2520
    Width                = 1215
End
Begin VB.CommandButton cmdcopy
    Caption              = "Sao chép"
    BeginProperty Font
        name              = "VNTimes new roman"
        charset           = 1
        weight            = 400
        size              = 9.75
        underline         = 0   'False
        italic            = 0   'False
        strikethrough     = 0   'False
    EndProperty
    Height               = 495
    Left                 = 1920
    TabIndex             = 2
    Top                  = 2520
    Width                = 1215
End
Begin VB.CommandButton cmdcut
    Caption              = "Cắt"
    BeginProperty Font
        name              = "VNTimes new roman"
        charset           = 1
        weight            = 400
        size              = 9.75
        underline         = 0   'False
        italic            = 0   'False
        strikethrough     = 0   'False
    EndProperty
    Height               = 495
```

```
        Left           = 360
        TabIndex      = 1
        Top           = 2520
        Width         = 1095
    End
    Begin VB.TextBox txt
        BeginProperty Font
            name          = "VNtimes new roman"
            charset       = 1
            weight        = 400
            size          = 9.75
            underline     = 0    'False
            italic        = 0    'False
            strikethrough = 0    'False
        EndProperty
        Height          = 2055
        Left            = 360
        MultiLine       = -1    'True
        ScrollBars      = 3    'Both
        TabIndex        = 0
        Text            = "THU1.frx":0000
        Top             = 120
        Width           = 3735
    End
End
Attribute VB_Name = "Form1"
Attribute VB_Creatable = False
Attribute VB_Exposed = False
Private Sub chkbold_Click()
    txt.FontBold = Not (txt.FontBold)
    txt.SetFocus
End Sub

Private Sub chkitalic_Click()
    txt.FontItalic = Not (txt.FontItalic)
    txt.SetFocus
End Sub
```

```
Private Sub chkunder_Click()
txt.FontUnderline = Not (txt.FontUnderline)
txt.SetFocus
End Sub

Private Sub cmdcopy_Click()
ClipText = txt.SelectedText
txt.SetFocus
End Sub

Private Sub cmdcut_Click()
ClipText = txt.SelectedText 'Chép khối lên bộ nhớ
txt.SelectedText = ""
txt.SetFocus
End Sub

Private Sub cmddel_Click()
txt.Text = ""
txt.SetFocus
End Sub

Private Sub cmdpaste_Click()
txt.SelectedText = ClipText
txt.SetFocus
End Sub

Private Sub Form_Load()
Dim ClipText As String
txt = ""
txt.FontBold = False
txt.FontItalic = False
txt.FontUnderline = False
chkbold.Value = 0
chkitalic.Value = 0
chkunder.Value = 0
End Sub

Private Sub Text1_Change()
Dim ClipText As String
End Sub
```

## **BÀI 11. BIẾN NHỚ**

### **11.1. Khái niệm :**

Trong Visual Basic ta có thể lưu trữ các giá trị để phục vụ cho quá trình xử lý dữ liệu dưới bốn hình thức là : biến, hằng, mảng và bản ghi.

Biến và hằng tại mỗi thời điểm chỉ lưu trữ được một giá trị còn mảng và bản ghi thì lưu trữ được nhiều giá trị cùng lúc.

Tên các đại lượng do người sử dụng tự qui định nhưng phải thỏa mãn :

- Không dài quá 40 ký tự.
- Ký tự đầu tiên là chữ cái, các ký tự đi sau có thể là số, dấu \_ (gạch dưới).
- Ký tự cuối cùng có thể là một trong các hậu tố cho biết kiểu dữ liệu như : %, &, !, #, @ và #.
- Tên biến không được trùng với các từ dành riêng (Reserved word).
- Visual Basic không phân biệt ký tự chữ in với chữ thường.

### **11.2. Khai báo biến :**

Ta có thể định nghĩa biến bằng một trong các cách sau :

11.2.1 Khai báo bằng : **DIM <Tên biến> AS <Tên kiểu dữ liệu>**

Trong đó tên kiểu dữ liệu được chọn từ một trong các kiểu sau :

- Integer (2 byte): là số nguyên bình thường thuộc [-32768, 32767].
- Long (4 byte): số nguyên dài thuộc [-2147483648, 2147483647].
- Single (4 byte): số thực độ chính xác đơn thuộc [-3.4E+38, 3.4E+38]. Giữ lại 7 chữ số sau dấu thập phân.

- Double (8 byte): số thực độ chính xác kép thuộc  $[-1.8E+308, 1.8E+308]$ . Giữ lại 16 chữ số sau dấu thập phân
- Currency (8 byte) : lưu trữ các giá trị là tiền tệ. Chứa được tối đa 15 chữ số bên trái và 4 chữ số bên phải dấu thập phân.
- String :chứa chuỗi ký tự có chiều dài thay đổi từ 0 đến 65535 ký tự.
- String\*num : chứa chuỗi ký tự có chiều dài định trước. Khi khai báo phải ấn định trước độ dài từ 0 đến 32767 ký tự.
- Varian : lưu trữ đồng thời hai thông tin là : giá trị và kiểu dữ liệu.
  - Phạm vi : các chỉ thị DIM có thể xuất hiện ở cấp đơn thể, cấp biểu mẫu hoặc cấp thủ tục. Nếu ta khai báo DIM ở cấp nào thì các biến chỉ có hiệu lực trong cấp đó.
  - Ví dụ : DIM sotien AS long

DIM luong AS Currency

DIM hoten AS String

### 11.2.2 Cách viết : **DIM <Tên biến><ký tự hậu tố>**

Ta có thể không cần dùng AS <kiểu dữ liệu> mà ghi trực tiếp ký hiệu hậu tố khai báo kiểu dữ liệu vào sau tên biến để định kiểu.

Ta có các ký tự hậu tố (Suffix Character) như sau :

- % : Integer.
- & : Long.
- ! : Single.
- # : Double.
- @ : Currency.
- \$ : String.

**Ví dụ :**



DIM hoten\$

DIM sotien@

DIM luong%

### 11.2.3 Khai báo biến toàn cục : **GLOBAL <Tên biến> AS <Tên kiểu dữ liệu>**

Trong trường hợp biến sẽ có hiệu lực trên tất cả các đơn thể của chương trình. Lệnh này chỉ được sử dụng ở phần declarations của đơn thể.

Ví dụ : GLOBAL hoten AS String

Lúc này biến hoten sẽ có hiệu lực trên toàn chương trình.

### 11.2.4 Khai báo nhiều biến : **DefType <miền ký tự>**

Trong Visual Basic nếu một biến được sử dụng mà không khai báo thì Visual Basic ngầm hiểu là biến Varian. Với chỉ thị DefType ta có thể chuyển biến Varian thành một kiểu dữ liệu khác.

- Visual Basic chấp nhận các chỉ thị DefType sau :
  - DefInt : Integer.
  - DefLng : Long.
  - DefSng : Single.
  - DefDbl : Double.
  - DefCur : Currency.
  - DefStr : String.
- DefVar : Varian.
- Miền ký tự : cho biết khoảng ký tự. Từ đây về sau những biến nào có ký tự bắt đầu thuộc miền ký tự trên sẽ có kiểu dữ liệu như trong DefType qui định.

**Ví dụ :** DefInt S

Từ đây về sau các biến như : Sotien, Soluong, S... đều có kiểu dữ liệu là Integer vì có ký tự mở đầu là S

Tuy nhiên chỉ thị này không ảnh hưởng đến các biến được khai báo bởi chỉ thị DIM hoặc mang các hậu tố qui định kiểu dữ liệu.

### **11.3. Khai báo hằng :**

Hằng (Constant) là một đại lượng có giá trị không thay đổi trong suốt thời gian thực hiện chương trình.

Cách khai báo :

CONST <Tên hằng> = <Biểu thức>, <Tên hằng 2> = <Biểu thức 2>, ...

Tên hằng cũng theo qui ước giống như tên biến. Ta có thể gắn ký tự hậu tố để định kiểu (% , !, &, #, \$, @) cho hằng. Tuy nhiên, sau này khi dùng tên hằng trong chương trình thì không được viết hậu tố này.

**Ví dụ :**

```
CONST diemgioi% = 7
```

Sau này trong chương trình ta chỉ viết :

```
IF dtb% >= diemgioi THEN MsgBox "Sinh viên này xếp loại Giỏi"
```

Phạm vi hoạt động của hằng giống như của biến. Nếu muốn khai báo hằng có tác dụng toàn cục thì viết :

```
GLOBAL CONST <Tên hằng> = <Biểu thức>
```

### **11.4. Khai báo mảng :**

Mảng (Array) là đại lượng có thể lưu trữ được nhiều giá trị khác nhau tại cùng một thời điểm thông qua các phần tử của nó.

#### **11.4.1 Khai báo mảng :**

- Khai báo bằng DIM :

- Cách viết : DIM <Tên mảng>(phần tử) AS <Kiểu dữ liệu>

Hoặc DIM <Tên mảng><Hậu tố kiểu>(phần tử)

- Tên mảng do người sử dụng tự qui định. Giống như cách khai báo tên biến.

- Phần tử : khai báo số lượng các phần tử trong mảng. Có nhiều cách :

Số lượng tối đa. Trong trường hợp này phần tử bắt đầu là không.

**Ví dụ :** DIM a(10) AS Integer hoặc DIM a%(10)

Lúc này sẽ có các phần tử là a(0), a(1), .... a(10) và mỗi phần tử chứa một số nguyên.

Phần tử bắt đầu đến phần tử cuối. Qui định rõ các các phần tử đầu đến cuối.

**Ví dụ :** DIM A(5 TO 10) AS Single hoặc DIM A!(5 TO 10)

Lúc này có các phần tử là a(5), a(6), ..., a(10) và mỗi phần tử là số thực độ chính xác đơn..

Mảng nhiều chiều. Giữa các chiều ngăn cách bởi dấu , (phẩy).

**Ví dụ :** DIM a(10,20) AS Integer hoặc DIM a%(10,20)

Lúc này sẽ có các phần tử là a(0,0), a(0,1), .... a(10,20) và mỗi phần tử chứa một số nguyên.

DIM A(1 TO 5, 1 TO 7) AS Single

Lúc này sẽ có các phần tử là a(1,1), a(1,2), .... a(5,7) và mỗi phần tử chứa một số thực.

Phạm vi hoạt động của biến mảng cũng giống như các biến bình thường khác.

- Khai báo bằng GLOBAL: lúc này biến sẽ có tác dụng trên toàn chương trình. Khai báo này phải đặt trong phần khai báo của đơn thể chứ không đặt trong thủ tục hoặc biểu mẫu.

#### 11.4.2 Sử dụng mảng :

Biến mảng được sử dụng trong chương trình giống như các biến thông thường khác, tuy nhiên phải chỉ rõ số hiệu phần tử của nó.

Ví dụ : khai báo biến mảng tháng và lưu trữ số thứ tự của tháng trong năm. Sau đó in các tên tháng ra màn hình.

```
DIM thang(1 TO 12) AS Integer
FOR I = 1 TO 12
    Thang(I) = I
NEXT I
FOR I = 1 TO 12
    Print "Thang :" + Str(thang(I))
NEXT I
```

### **11.5. Khai báo bảng ghi :**

Bản ghi là kiểu dữ liệu đặc biệt bao gồm nhiều giá trị thuộc nhiều kiểu dữ liệu khác nhau. Biến bản ghi được sử dụng nhiều để giải quyết các bài toán trong quản lý số liệu.

#### 11.5.1 Khai báo :

Khai báo bản ghi bắt buộc phải được đặt ở phân đoạn Declaration của đơn thể chương trình mà không thể đặt ở cấp biểu mẫu hoặc thủ tục. Biến bản ghi đương nhiên phải là biến toàn cục.

Cách khai báo :

```
TYPE      <Tên bản ghi>
    <Tên trường 1>   AS  <Tên kiểu dữ liệu 1>
    <Tên trường 2>   AS  <Tên kiểu dữ liệu 2>
    .....
    <Tên trường n>   AS  <Tên kiểu dữ liệu n>
END TYPE
DIM <Tên biến> AS <Tên bản ghi>
```

Tên bản ghi do người sử dụng tự qui định theo qui tắc của tên biến.

#### 11.5.2 Sử dụng biến bản ghi :

Các biến bản ghi được sử dụng như các biến bình thường khác nhưng phải chỉ rõ tên trường ở phía sau và ngăn cách với tên biến bởi dấu . (chấm).

Cách viết : <Tên biến bản ghi> . <Tên trường>

## 11.6. Biến đổi (convert) từ loại dữ liệu này qua loại dữ liệu khác

Nhiều lúc ta cần phải convert data type của một variable từ loại này qua loại khác, VB6 cho ta một số các Functions dưới đây. Xin lưu rằng khi gọi các Functions này, nếu chúng ta đưa một data value bất hợp lệ thì có thể bị lỗi.

Function	Chú thích
CBool ()	Đổi parameter ra True hay False. Nếu Integer khác 0 thì được đổi thành True
CByte ()	Đổi parameter ra một con số từ 0 đến 255 nếu có thể được, nếu không thì là 0.
CDate ()	Đổi parameter ra Date
CDbl ()	Đổi parameter ra Double precision floating point number
CInt ()	Đổi parameter ra Integer
CSng ()	Đổi parameter ra Single precision floating point number
CStr ()	Đổi parameter ra String

Ngoài các Function nói trên chúng ta cũng có thể dùng **Function Val** để convert một String ra Number. Lưu ý là khi Function Val process một String nếu nó gặp một character nào không phải là digit hay decimal point thì nó không process tiếp nữa. Do đó nếu Input String là "\$25.50" thì Val returns con số 0 vì \$ không phải là một digit. Nếu Input String là "62.4B" thì Val returns 62.4.

CDbl là Function dùng để convert một String ra số an toàn nhất. Input String có thể chứa các dấu , và . (thí dụ: 1,234,567.89) tùy theo nơi chúng ta ở trên thế giới (thí dụ như Âu Châu hay Mỹ). CSng cũng làm việc giống như CDbl nhưng nếu con số lớn hơn 1 triệu nó có thể bị bug.

Cái bug hay gặp của CSng là nếu Input String không có gì cả (tức là InputString="") thì Function CSng cho chúng ta Type Mismatch Error. Do đó để khắc phục khuyết điểm này chúng ta có thể viết cho mình một Function tạm đặt tên là CSingle để dùng thế cho CSng :

```
Function CSingle(strNumber) As Single
    If Trim(strNumber) = "" Then
        CSingle = 0#
    Else
        CSingle = CSng(strNumber)
    End If
End Function
```

## BÀI 12. CÁC CẤU TRÚC ĐIỀU KHIỂN

Tương tự như các ngôn ngữ lập trình khác trong Visual Basic ta có thể sử dụng các cấu trúc điều khiển trong chương trình để có thể chọn lựa công việc thực hiện hoặc tự động lặp lại nhóm chỉ thị nhiều lần.

### 12.1. Cấu trúc chọn :

#### 12.1.1 Cấu trúc : **IF <Điều kiện> THEN <Chỉ thị>**

Khi gặp cấu trúc này nếu điều kiện có giá trị True thì thực hiện chỉ thị nếu điều kiện có giá trị False thì bỏ qua chỉ thị đó.

**Ví dụ :**

```
IF dtb > 5 THEN Print "Bạn đủ điểm"
```

Trong trường hợp này chỉ có duy nhất một chỉ thị.

#### 12.1.2 Cấu trúc : **IF <Điều kiện> THEN <Chỉ thị 1> ELSE <Chỉ thị 2>**

Khi gặp cấu trúc này nếu điều kiện có giá trị True thì thực hiện chỉ thị 1 nếu điều kiện có giá trị False thì thực hiện chỉ thị 2.

**Ví dụ :**

```
IF dtb > 5 THEN Print "Bạn đủ điểm" ELSE Print "Bạn thiếu điểm"
```

**Chú ý :**

- Nếu muốn sau THEN hoặc ELSE có nhiều chỉ thị cần thực hiện thì phải viết xuống dòng và cuối cấu trúc này phải có END IF.

Cách viết :

```
IF <Điều kiện> THEN
    <Chỉ thị 1>
    .....
    <Chỉ thị n>
```

```
ELSE
    <Chỉ thị 1'>
    .....
    <Chỉ thị n'>
END IF
```

### 12.1.3 Cấu trúc : **Select Case <Biểu thức>**

```
Case <Liệt kê biểu thức 1>
    <Khối chỉ thị 1>
Case <Liệt kê biểu thức 2>
    <Khối chỉ thị 2>
.....
[Case Else
    <Khối chỉ thị n>]
End Select
```

Trong đó :

- **Biểu thức** : là một thức chuỗi hoặc số. Nếu giá trị của biểu thức ở đây trùng với giá trị của các biểu thức được liệt kê nào bên dưới thì khối chỉ thị tương ứng được thực hiện.
- **Liệt kê biểu thức I** : là biểu thức sẽ được đem so sánh với biểu thức đầu. Trong phần này biểu thức liệt kê có thể được viết dưới các dạng sau :
  - **Biểu thức** : số hoặc chữ.
  - **Biểu thức 1 TO Biểu thức 2** : chỉ ra đoạn giá trị nằm giữa biểu thức 1 và 2.
  - **IS <phép so sánh> <biểu thức>** : chỉ ra phép so sánh và giá trị so sánh.
- **Khối chỉ thị I** : là các chỉ thị cần thực hiện trong trường hợp giá trị của biểu thức thứ I trùng với giá trị của biểu thức đầu. Ở đây có thể gồm nhiều chỉ thị được viết trên nhiều dòng.

Ví dụ : viết chương trình nhập vào tuổi một người và cho biết người đó thuộc lứa tuổi nào.

```
Sub Form_Click()
    Dim Cauhoi, Tuoi      'Khai báo biến Cauhoi và Tuoi
    Cauhoi = "Bạn bao nhiêu tuổi :"
```

```
Tuoi = InputBox(Cauhoi) ' Nhập tuổi vào biến tuoi
Select Case Tuoi
    Case 1 TO 12
        MsgBox "Bạn ở tuổi Nhi đồng" 'In ra dòng thông báo
    Case 13 TO 18
        MsgBox "Bạn ở tuổi Thiếu niên"
    Case 18 TO 25
        MsgBox "Bạn ở tuổi Thanh niên"
    Case 25 TO 60
        MsgBox "Bạn đã là Người lớn"
    Case IS > 60
        MsgBox "Bạn ở tuổi Về hưu"
    Case Else
        MsgBox "Bạn không phải con người"
End Select
End Sub
```

## 12.2. Cấu trúc lặp : cho phép tự động lặp đi lặp lại nhóm lệnh nhiều lần.

### 12.2.1 Cấu trúc :

```
FOR <Biến đếm> = <Giá trị đầu> TO <Giá trị cuối> [STEP n]
    [Khởi chỉ thị 1]
    [Exit For]
    [Khởi chỉ thị 2]
NEXT <biến đếm>
```

- **Biến đếm** : là tên biến dùng để kiểm tra số lần lặp.
- **Giá trị đầu** : là giá trị khởi gán lần đầu tiên cho biến đếm khi thực hiện vòng lặp.
- **Giá trị cuối** : là giá trị cuối cùng của biến đếm. Khi biến đếm đạt đến giá trị này thì vòng lặp thực hiện lần cuối và dừng quá trình lặp.
- **STEP n** : chỉ định bước nhảy n để thực hiện thay đổi giá trị của biến đếm sau mỗi lần lặp.
- **Khởi chỉ thị** : liệt kê các chỉ thị cần thực hiện trong mỗi lần lặp.
- **Exit For** : nếu trong vòng lặp mà gặp chỉ thị này thì sẽ ngưng vòng lặp.



## Lập trình trực quan

---

Vòng lặp trên cho phép tự động thực hiện các chỉ thị với số lần lặp xác định trước.

Ví dụ : viết đoạn lệnh in ra các số nguyên từ 1 đến 10.

```
FOR so! = 1 TO 10                                'biến số là Single
  Print so!
NEXT so!
```

Ví dụ : viết đoạn lệnh in ra các số với bước nhảy 0.25 và từ 0 đến 10.

```
FOR so! = 0 TO 10 STEP 0.25                      'biến số là Single
  Print so!
NEXT so!
```

### 12.2.2 Cấu trúc :

```
WHILE <Điều kiện>
  [Khởi chỉ thị]
Wend
```

- Điều kiện : qui định điều kiện để thực hiện vòng lặp. Nếu điều kiện có giá trị True thì thực hiện khối chỉ thị, gặp Wend sẽ quay trở lại kiểm tra điều kiện. Quá trình trên kết thúc khi điều kiện có giá trị False.

- Khởi chỉ thị : các chỉ thị cần thực hiện trong vòng lặp.

Ví dụ : viết đoạn lệnh in ra các số nguyên từ 1 đến 10.

```
So! = 1
While so! <= 10
  Print so!
  So! = so! + 1
Wend
```

### 12.2.3 Cấu trúc :

```
DO [WHILE | UNTIL <Điều kiện>]
  [Khởi chỉ thị]
  [Exit Do]
  [Khởi chỉ thị]
LOOP [WHILE | UNTIL <Điều kiện>]
```

- Điều kiện : qui định điều kiện để thực hiện vòng lặp.

## Lập trình trực quan

---

- Nếu dùng WHILE thì điều kiện có giá trị True thì thực hiện khối chỉ thị, nếu False kết thúc vòng lặp.
- Nếu dùng UNTIL thì điều kiện có giá trị False thì thực hiện khối chỉ thị, nếu True kết thúc vòng lặp.

Ta có thể đặt điều kiện ở đầu hoặc cuối vòng lặp đều được.

- Khối chỉ thị : các chỉ thị cần thực hiện trong vòng lặp.
- Exit Do : cho phép dừng vòng lặp mà không cần qua kiểm tra điều kiện.

Ví dụ : viết đoạn lệnh in ra các số nguyên từ 1 đến 10.

```
So! = 1
Do While so! <= 10
    Print so!
    So! = so! + 1
Loop
```

Hoặc :

```
So! = 1
Do
    Print so!
    So! = so! + 1
Loop While so! <= 10
```

Hoặc :

```
So! = 1
Do
    Print so!
    So! = so! + 1
Loop Until so! > 10
```

### 12.3. Nhãn :

Trong Visual Basic ta có thể chuyển đến thực hiện ở một đoạn chương trình hoặc một dòng lệnh mới bằng cách dùng nhãn hoặc số thứ tự dòng lệnh.

### 12.3.1 Nhãn :

Là một đoạn chỉ thị lệnh bất kỳ trong chương trình được gán một tên xác định. Khi cần thực hiện đoạn chỉ thị này ta chỉ việc nhảy về nhãn đó.

Mỗi nhãn được dùng trong biểu mẫu hoặc đơn thể phải là duy nhất. Không thể sử dụng hai nhãn trùng tên trong một biểu mẫu, thủ tục, hộp điều khiển...

- Cách viết tên\_nhãn: <Nhóm chỉ thị>
- Cách gọi :
  - Cách 1 : sử dụng lệnh GOTO <tên\_nhãn>
  - Cách 2 : sử dụng lệnh ON <biểu thức số> GOTO <liệt kê các tên nhãn>

Trong trường hợp này biểu thức số có giá trị từ 1 đến 255 và tên nhãn có số thứ tự tương ứng với biểu thức số sẽ được thực hiện.

**Ví dụ :** ON stt GOTO nhan1, nhan2, nhan3

Nếu stt có giá trị 1 thì nhan1 được thực hiện, stt là 2 thì nhan2 thực hiện và stt là 3 thì nhan3 được thực hiện.

**Ví dụ :**

```
Sub Form_Click()  
    Print "Giáo trình"  
    GOTO Nhan1  
    Print "Không in"  
    Nhan1:  
    Print "Lập trình trực quan"  
End Sub
```

Lúc này kết quả trên màn hình ta nhận được :

Giáo trình

Lập trình trực quan

Còn dòng lệnh Print "Không in" sẽ không thực hiện.

### 12.3.2 Số thứ tự dòng lệnh :

Là phương pháp đánh số ở trước mỗi dòng lệnh và khi cần ta có nhảy đến vị trí này bất cứ lúc nào.

Mỗi số được dùng trong biểu mẫu hoặc đơn thể phải là duy nhất. Không thể sử dụng hai số trùng giá trị để đánh số dòng lệnh trong một biểu mẫu, thủ tục, hộp điều khiển...

Các số dùng đánh số dòng lệnh là tùy ý không bắt buộc phải đánh số theo thứ tự tăng hay giảm dần mà ngẫu nhiên, không bắt buộc phải đánh số tất cả các chỉ thị lệnh mà thích đánh số vào dòng lệnh nào cũng được.

Cách gọi : GOTO <giá trị số>

Khi thực hiện lệnh này Visual Basic sẽ chuyển đến dòng lệnh được đánh số tương ứng.

#### **Ví dụ :**

100	MsgBox	"Dòng lệnh mang số 100"
101	MsgBox	"Dòng lệnh mang số 101"
57	MsgBox	"Dòng lệnh mang số 57"
GOTO	101	

## BÀI 13. METHOD

Method là các chương trình được xây dựng sẵn để phục vụ cho việc thực hiện các thao tác thường gặp. Method có tác dụng gần giống như lệnh, thủ tục hoặc hàm được xây dựng sẵn trong các ngôn ngữ lập trình có cấu trúc. Thông thường Method chỉ tác dụng lên một lớp các đối tượng.

Sau đây ta sẽ xét một số Method thường được sử dụng.

### 13.1. Circle Method

Cú pháp : [Object].Circle [Step] (X,Y), Radius [, [Color], [Start], [End], [Aspect]]

- Object : tên của biểu mẫu hoặc khung hình mà ta cần vẽ hình tròn trên đó.
- Step : cho biết đây là tọa độ tương đối so với vị trí hiện hành do hai thuộc tính CurrentX và CurrentY cung cấp.
- X, Y : chỉ định tọa độ tâm của hình tròn, ellipse hoặc cung tròn.
- Radius : chỉ định bán kính.
- Color : chỉ định màu sắc. Màu tương ứng với giá trị là một số nguyên.
- Start, End : trị số tính theo Radian, cho biết điểm xuất phát và điểm kết thúc khi vẽ một cung tròn hoặc Ellipse.
- Aspect : trị số cho biết góc xoay mặt phẳng chứa hình tròn để tạo ra hình Ellipse.

Tác dụng : cho phép tạo ra một hình tròn, cung tròn hoặc hình Ellipse theo yêu cầu người sử dụng.

Ví dụ 1: Form.Circle (1000, 2000), 500

Vẽ hình tròn có tâm là điểm (1000,2000) và bán kính là 500. (các đơn vị tính theo Fixed).

Ví dụ 2 : vẽ một dãy các hình tròn đồng tâm với màu sắc tùy ý.

```
Private Sub Hinhtron_Click()  
Dim CX, CY, Radius          ' Declare variable.  
ScaleMode = 3              ' Set scale to pixels.  
CX = ScaleWidth / 2        ' Set X position.  
CY = ScaleHeight / 2       ' Set Y position.  
If CX > CY Then Limit = CY Else Limit = CX  
For Radius = 0 To Limit    ' Set radius.  
    Circle (CX, CY), Radius, RGB(Rnd * 255, Rnd * 255, Rnd *  
255)  
Next Radius  
End Sub
```

### **13.2. Line Method**

Cú pháp : [Object].Line [Step] (X1, Y1) - [Step] (X2, Y2)[,Color][,BF]

- Object : tên của biểu mẫu hoặc khung hình mà ta cần vẽ đường thẳng trên đó.
- X1, Y1 : chỉ định tọa độ điểm xuất phát.
- X2, Y2 : chỉ định tọa độ điểm kết thúc.
- Color : chỉ định màu sắc. Màu tương ứng với giá trị là một số nguyên.
- Step : cho biết đây là tọa độ tương đối so với vị trí hiện hành do hai thuộc tính CurrentX và CurrentY cung cấp.
- B (Box) : vẽ một khung hình chữ nhật. Lúc này điểm xuất phát và điểm kết thúc là hai góc hình chữ nhật.
- F (Fill) : khung hình chữ nhật sẽ được tô màu.

Tác dụng : cho phép tạo ra một đoạn thẳng hoặc khung hình chữ nhật theo yêu cầu người sử dụng.

Ví dụ : vẽ các hình và các đường thẳng với nhiều màu sắc và hình dạng khác nhau.

```
Private Sub duongthang_Click()  
Dim CX, CY, F, F1, F2, I    ' Declare variables  
ScaleMode = 3              ' Set ScaleMode to pixels.
```

```
CX = ScaleWidth / 2      ' Get horizontal center.
CY = ScaleHeight / 2     ' Get vertical center.
DrawWidth = 8           ' Set DrawWidth.
For I = 50 To 0 Step -2
    F = I / 50           ' Perform interim
    F1 = 1 - F: F2 = 1 + F ' calculations.
    ForeColor = QBColor(I Mod 15) ' Set foreground color.
    Line (CX * F1, CY * F1)-(CX * F2, CY * F2), , BF
Next I
DoEvents                ' Yield for other processing.
If CY > CX Then         ' Set DrawWidth.
    DrawWidth = ScaleWidth / 25
Else
    DrawWidth = ScaleHeight / 25
End If
For I = 0 To 50 Step 2  ' Set up loop.
    F = I / 50          ' Perform interim
    F1 = 1 - F: F2 = 1 + F ' calculations.
    Line (CX * F1, CY)-(CX, CY * F1) ' Draw upper-left.
    Line -(CX * F2, CY) ' Draw upper-right.
    Line -(CX, CY * F2) ' Draw lower-right.
    Line -(CX * F1, CY) ' Draw lower-left.
    ForeColor = QBColor(I Mod 15) ' Change color each time.
Next I
DoEvents                ' Yield for other processing.
End Sub
```

### 13.3. Cls Method

Cú pháp : [object].Cls

Tác dụng : xóa màn hình của Form.

Ví dụ :

```
Private Sub Xoa_Click()
    Dim Msg ' Declare variable.
    AutoRedraw = -1 ' Turn on AutoRedraw.
    ForeColor = QBColor(15) ' Set foreground to white.
```

```
BackColor = QBColor(1) ' Set background to blue.
FillStyle = 7 ' Set diagonal crosshatch.
Line (0, 0)-(ScaleWidth, ScaleHeight), , B ' Put box on form.
Msg = "This is information printed on the form background."
CurrentX = ScaleWidth / 2 - TextWidth(Msg) / 2 ' Set X position.
CurrentY = 2 * TextHeight(Msg) ' Set Y position.
Print Msg ' Print message to form.
Msg = "Choose OK to clear the information and background "
Msg = Msg & "pattern just displayed on the form."
MsgBox Msg ' Display message.
Cls ' Clear form background.
End Sub
```

### **13.4. Hide Method**

Cú pháp : [Object.]Hide

Tác dụng : che cửa sổ Form.

Ví dụ : che và làm xuất hiện lại cửa sổ Form đang làm việc..

```
Private Sub Chehien_Click()
Dim Msg ' Declare variable.
Hide ' Hide form.
Msg = "Choose OK to make the form reappear."
MsgBox Msg ' Display message.
Show ' Show form again.
End Sub
```

### **13.5. Show Method**

Cú pháp : [Object.]Show

Tác dụng : làm xuất hiện cửa sổ Form.

```
Private Sub Chehien_Click()
Dim Msg ' Declare variable.
Hide ' Hide form.
Msg = "Choose OK to make the form reappear."
MsgBox Msg ' Display message.
```



```
Show      ' Show form again.  
End Sub
```

### 13.6. Item Method

Cú pháp : [Object.]Item(Index)

Tác dụng : sắp xếp lại các thành viên trong Collection theo thứ tự của khóa chỉ định trong Index.

Ví dụ :

```
Dim SmithBillBD As Object  
Dim SmithAdamBD As Object  
Set SmithBillBD = Birthdays.Item("SmithBill")  
Set SmithAdamBD = Birthdays("SmithAdam")
```

### 13.7. Move Method

Cú pháp : [Object.]Move Left [, Top][, Width][, Height]

- Object: tên Object cần chuyển dịch.
- Left : qui định giá trị cần dịch chuyển sang bên trái.
- Top : qui định dịch chuyển lên phía trên.
- Width : qui định độ rộng mới của đối tượng.
- Height : qui định độ cao mới của đối tượng.

Tác dụng : cho phép di chuyển và điều chỉnh kích thước của đối tượng.

Ví dụ :

```
Private Sub dichuyen_Click()  
    Dim Inch, Msg      ' Declare variables.  
    Msg = "Choose OK to resize and move this form by "  
    Msg = Msg & "changing the value of properties."  
    MsgBox Msg      ' Display message.  
    Inch = 1440      ' Set inch in twips.
```

```
Width = 4 * Inch      ' Set width.
Height = 2 * Inch     ' Set height.
Left = 0              ' Set left to origin.
Top = 0               ' Set top to origin.
Msg = "Now choose OK to resize and move this form "
Msg = Msg & "using the Move method."
MsgBox Msg           ' Display message.
Move Screen.Width-2*Inch, Screen.Height-Inch, 2*Inch, Inch
End Sub
```

### **13.8. Point Method**

Cú pháp :            [Object.]Point (X, Y)

- X : Hoành độ của điểm cần vẽ.
- Y : Tung độ của điểm cần vẽ.

Tác dụng : trả về một điểm có toạ độ xác định.

Ví dụ : tô màu bằng các dấu chấm.

```
Private Sub vediem_Click()
Dim LeftColor, MidColor, Msg, RightColor 'Declare variables.
AutoRedraw = -1 ' Turn on AutoRedraw.
Height = 3 * 1440 ' Set height to 3 inches.
Width = 5 * 1440 ' Set width to 5 inches.
BackColor = QBColor(1) ' Set background to blue.
ForeColor = QBColor(4) ' Set foreground to red.
Line (0, 0)-(Width / 3, Height), , BF ' Red box.
ForeColor = QBColor(15) ' Set foreground to white.
Line (Width / 3, 0)-((Width / 3) * 2, Height), , BF
LeftColor = Point(0, 0) ' Find color of left box,
MidColor = Point(Width / 2, Height / 2) ' middle box, and
RightColor = Point(Width, Height) ' right box.
Msg = "The color number for the red box on the left side of "
Msg = Msg & "the form is " & LeftColor & ". The "
Msg = Msg & "color of the white box in the center is "
Msg = Msg & MidColor & ". The color of the blue "
Msg = Msg & "box on the right is " & RightColor & "."
```

```
MsgBox Msg ' Display message.  
End Sub
```

### **13.9. Print Method**

Cú pháp : [Object.]Print OutputList

Tác dụng : cho phép in giá trị các biểu thức trong OutputList ra đối tượng. OutputList là một danh sách các biểu thức cần in. Object là tên đối tượng mà ta cần in lên đó.

Nếu muốn in máy in thì tên Object là Printer. Ví dụ :

```
Private Sub Command1_Click()  
Dim MyVar  
MyVar = "Chúc các bạn lập trình thật tốt với Visual Basic."  
Print MyVar  
End Sub
```

### **13.10. PrintForm Method**

Cú pháp : [Object.]PrintForm

Tác dụng : cho phép in tất cả các hình ảnh của biểu mẫu ra giấy. Nếu không chỉ rõ tên Form thì Form đang làm việc sẽ được in. Object ở đây là tên Form cần in.

Ví dụ

```
Private Sub Command1_Click()  
Dim Msg ' Declare variable.  
On Error GoTo ErrorHandler ' Set up error handler.  
PrintForm ' Print form.  
Exit Sub  
ErrorHandler:  
Msg = "The form can't be printed."  
MsgBox Msg ' Display message.  
Resume Next  
End Sub
```

### 13.11. PSet Method

Cú pháp : [Object.]Pset [Step] (X, Y)[, Color]

- Object : An object expression that evaluates to an object in the Applies To list. If object is omitted, the Form with the focus is assumed to be object.
- Step : A keyword specifying that the coordinates are relative to the current graphics position given by the CurrentX and CurrentY properties.
- (X, Y) : Single-precision values indicating the horizontal (x-axis) and vertical (y-axis) coordinates of the point to set.
- Color: Long integer value indicating the RGB color specified for point.

Tác dụng : tương tự như Point Method.

Ví dụ : vẽ các chấm điểm với màu sắc và vị trí ngẫu nhiên trên cửa sổ Form.

```
Private Sub Form_Click ()
    Dim CX, CY, Msg, XPos, YPos      ' Declare variables.
    ScaleMode = 3      ' Set ScaleMode to pixels.
    DrawWidth = 5      ' Set DrawWidth.
    ForeColor = QBColor(4)      ' Set background to red.
    FontSize = 24      ' Set point size.
    CX = ScaleWidth / 2      ' Get horizontal center.
    CY = ScaleHeight / 2      ' Get vertical center.
    Cls      ' Clear form.
    Msg = "Chúc mừng năm mới!"
    CurrentX = CX - TextWidth(Msg) / 2      ' Horizontal position.
    CurrentY = CY - TextHeight(Msg)      ' Vertical position.
    Print Msg      ' Print message.
    Do
        XPos = Rnd * ScaleWidth      ' Get horizontal position.
        YPos = Rnd * ScaleHeight      ' Get vertical position.
        PSet (XPos, YPos), QBColor(Rnd * 15)      ' Draw confetti.
    DoEvents      ' Yield to other
    Loop      ' processing.
End Sub
```

### 13.12. Refresh Method

Cú pháp : [Object.]Refresh

Tác dụng : cho phép "làm tươi" lại đối tượng, nghĩa là nó cho phép vẽ lại hình ảnh của Object.

Ví dụ :

```
Private Sub Form_Click ()
    Dim FNMA, I, Msg ' Declare variables.
    File1.Pattern = "TestFile.*" ' Set file pattern.
    For I = 1 To 8 ' Do eight times.
        FNMA = "TESTFILE." & I
        Open FNMA For Output As FreeFile ' Create empty file.
        File1.Refresh ' Refresh file list box.
        Close ' Close file.
    Next I
    Msg = "Choose OK to remove the created test files."
    MsgBox Msg ' Display message.
    Kill "TESTFILE.*" ' Remove test files.
    File1.Refresh ' Update file list box.
End Sub
```

### 13.13. Scale Method

Cú pháp : [Object.]Scale [(X1, Y1) - (X2, Y2)]

- Object : tên của đối tượng cần định lại hệ thống tọa độ.
- (X1, Y1) : tọa độ góc trên bên trái.
- (X2, Y2) : tọa độ góc phải bên dưới.

Tác dụng : qui định lại hệ thống tọa độ theo yêu cầu người sử dụng. Nếu không có (X1, Y1) và (X2, Y2) thì trả hệ thống tọa độ về giá trị ngầm định.

Ví dụ :

```
Private Sub Tile_Click()
    Dim I, OldFontSize ' Declare variables.
    Width = 8640: Height = 5760 ' Set form size in twips.
    Move 100, 100 ' Move form origin.
    AutoRedraw = -1 ' Turn on AutoRedraw.
    OldFontSize = FontSize ' Save old font size.
    BackColor = QBColor(7) ' Set background to gray.
    Scale (0, 110)-(130, 0) ' Set custom coordinate system.
    For I = 100 To 10 Step -10
        Line (0, I)-(2, I) ' Draw scale marks every 10 units.
        CurrentY = CurrentY + 1.5 ' Move cursor position.
        Print I ' Print scale mark value on left.
        Line (ScaleWidth - 2, I)-(ScaleWidth, I)
        CurrentY = CurrentY + 1.5 ' Move cursor position.
        CurrentX = ScaleWidth - 9
        Print I ' Print scale mark value on right.
    Next I
    ' Draw bar chart.
    Line (10, 0)-(20, 45), RGB(0, 0, 255), BF ' First blue bar.
    Line (20, 0)-(30, 55), RGB(255, 0, 0), BF ' First red bar.
    Line (40, 0)-(50, 40), RGB(0, 0, 255), BF
    Line (50, 0)-(60, 25), RGB(255, 0, 0), BF
    Line (70, 0)-(80, 35), RGB(0, 0, 255), BF
    Line (80, 0)-(90, 60), RGB(255, 0, 0), BF
    Line (100, 0)-(110, 75), RGB(0, 0, 255), BF
    Line (110, 0)-(120, 90), RGB(255, 0, 0), BF
    CurrentX = 18: CurrentY = 100 ' Move cursor position.
    FontSize = 14 ' Enlarge font for title.
    Print "Widget Quarterly Sales" ' Print title.
    FontSize = OldFontSize ' Restore font size.
    CurrentX = 27: CurrentY = 93 ' Move cursor position.
    Print "Planned Vs. Actual" ' Print subtitle.
    Line (29, 86)-(34, 88), RGB(0, 0, 255), BF ' Print legend.
    Line (43, 86)-(49, 88), RGB(255, 0, 0), BF
    Scale
End Sub
```

Chú ý : ta có thể thay đổi đơn vị đo trong Visual Basic bằng cách thay đổi trị số của ScaleMode trong bộ thuộc tính Properties.

### **13.14. SetFocus Method**

Cú pháp : [Object.]SetFocus

Tác dụng : cho phép tham chiếu đến Object có tên được chỉ định để thực hiện các thay đổi trên đó nếu có.

Ví dụ : Vehinh.SetFocus

### **13.15. Show Method**

Cú pháp : [Object.]Show [Style]

Style để qui định trạng thái và nó có giá trị 0 hoặc 1.

Tác dụng : cho phép làm xuất hiện đối tượng có tên được chỉ định.

Ví dụ :

```
Private Sub Hienhinh_Click()  
    Dim Msg ' Declare variable.  
    Hide    ' Hide form.  
    Msg = "Choose OK to make the form reappear."  
    MsgBox Msg ' Display message.  
    Show    ' Show form again.  
End Sub
```

### **13.16. TextHeight và TextWidth Methods**

Cú pháp : [Object.]TextHeight (String)

[Object.]TextWidth (String)

- Object : tên của đối tượng đã được ấn định kích cỡ Font chữ mà ta dựa vào đó để tính chiều cao và chiều rộng của đoạn văn bản cần thể hiện.
- String : nội dung chuỗi ký tự mà Method sẽ tính toán chiều cao và chiều rộng.

Tác dụng : tính toán và trả về kết quả là chiều cao và chiều rộng của String.

Ví dụ :

```
Private Sub Inchu_Click()  
    Dim HalfWidth, HalfHeight, Msg ' Declare variable.  
    AutoRedraw = -1 ' Turn on AutoRedraw.  
    BackColor = QBColor(4) ' Set background color.  
    ForeColor = QBColor(15) ' Set foreground color.  
    Msg = "Visual Basic" ' Create message.  
    FontSize = 48 ' Set font size.  
    HalfWidth = TextWidth(Msg) / 2 ' Calculate one-half width.  
    HalfHeight = TextHeight(Msg) / 2 ' Calculate one-half height.  
    CurrentX = ScaleWidth / 2 - HalfWidth ' Set X.  
    CurrentY = ScaleHeight / 2 - HalfHeight ' Set Y.  
    Print Msg ' Print message.  
End Sub
```



## **BÀI 14. HÀM**

Trong Visual Basic đã xây dựng sẵn các hàm để phục vụ cho việc xử lý dữ liệu một cách dễ dàng và nhanh chóng. Trong phần này ta xét một số hàm thường dùng.

### **14.1. Các hàm xử lý chuỗi :**

#### **14.1.1 Tìm chiều dài chuỗi : `LEN(String)`**

Trả về kết quả là số ký tự có trong String.

Ví dụ : `LEN("ABCD")` trả về kết quả là 4.

#### **14.1.2 Chuyển sang chữ thường :**

`LCase(String)` hoặc `Lcase$(String)`

Trả về kết quả là chuỗi ký tự mới sau khi đổi chuỗi cũ sang chữ thường. Nếu có dấu \$ thì trả về kết quả thuộc kiểu dữ liệu Variant nếu có \$ kết quả trả về kiểu String.

Ví dụ : `LCase("ABCD")` trả về kết quả là abcd.

#### **14.1.3 Chuyển sang chữ in :**

`UCase(String)` hoặc `Ucase$(String)`

Trả về kết quả là chuỗi ký tự mới sau khi đổi chuỗi cũ sang chữ in. Nếu có dấu \$ thì trả về kết quả thuộc kiểu dữ liệu Variant nếu có \$ kết quả trả về kiểu String.

Ví dụ : `UCase("abcd")` trả về kết quả là ABCD.

#### **14.1.4 Lấy các ký tự bên trái :**

`Left(String,n)` hoặc `Left$(String,n)`

Trả về kết quả là chuỗi ký tự mới gồm n ký tự bên trái của chuỗi cũ.

Ví dụ : `Left("ABCD",2)` trả về kết quả là AB

14.1.5 Lấy các ký tự bên phải:

`Right(String,n)` hoặc `Right$(String,n)`

Trả về kết quả là chuỗi ký tự mới gồm n ký tự bên phải của chuỗi cũ.

Ví dụ : `Right("ABCD",2)` trả về kết quả là CD

14.1.6 Lấy nhóm ký tự bất kỳ:

`Mid(String,m,n)` hoặc `Mid$(String,m,n)`

Trả về kết quả là chuỗi ký tự mới gồm n ký tự bắt đầu từ ký tự thứ m của chuỗi cũ.

Ví dụ : `Mid("ABCD",2,2)` trả về kết quả là BC

14.1.7 Bỏ các ký tự trống:

`Trim(String)` hoặc `Trim$(String)`

Trả về kết quả là chuỗi ký tự mới sau khi đã vớt bỏ các ký tự trống ở hai đầu chuỗi cũ.

Ví dụ : `Trim(" AB ")` trả về kết quả là "AB"

14.1.8 Bỏ các ký tự trống bên trái:

`LTrim(String)` hoặc `LTrim$(String)`

Trả về kết quả là chuỗi ký tự mới sau khi đã vớt bỏ các ký tự trống bên trái của chuỗi cũ.

Ví dụ : `LTrim(" AB ")` trả về kết quả là "AB "

14.1.9 Bỏ các ký tự trống bên phải:

`RTrim(String)` hoặc `RTrim$(String)`

Trả về kết quả là chuỗi ký tự mới sau khi đã vớt bỏ các ký tự trống bên phải của chuỗi cũ.

Ví dụ : `RTrim(" AB ")` trả về kết quả là " AB"

14.1.10 Đổi mã số sang ký tự:

Chr(mã số) hoặc Chr\$(mã số)

Trả về kết quả là một ký tự tương ứng với mã số trong bảng mã ANSI. Mã số là một số nguyên từ 0 đến 255.

Ví dụ : Chr(65) trả về kết quả là "A"

14.1.11 Đổi ký tự sang mã số:

Asc(Ký tự)

Trả về kết quả là một số kiểu Integer tương ứng với ký tự trong bảng mã ANSI.

Ví dụ : Asc("A") trả về kết quả là 65.

14.1.12 Đổi chuỗi sang số:

Val(biểu thức chuỗi)

Trả về kết quả là một số sau khi đổi chuỗi dạng số (kiểu String) sang giá trị số.

Ví dụ : Val("123") + Val("213") trả về kết quả là 336

14.1.13 Đổi số sang chuỗi:

Str\$(biểu thức số)

- Trả về kết quả là một chuỗi ký tự sau khi đổi số sang.

Ví dụ : Str(123) + Str(213) trả về kết quả là "123213"

14.1.14 Định dạng chuỗi:

Format\$(biểu thức [, dạng])

Trả về kết quả là một chuỗi ký tự được định dạng theo một khuôn mẫu cho trước. Biểu thức ở đây có thể là số hoặc chuỗi.

- Các ký tự định dạng số :

- # : hiển thị số nếu có còn không thì không hiện gì cả.
- 0 : hiển thị số nếu có còn không thì xuất hiện ký tự 0.
- . : hiển thị dấu chấm ở vị trí khai báo.
- , : hiển thị dấu phẩy ở vị trí khai báo.
- % : nhân biểu thức với 100 rồi xuất hiện dấu %.

Ví dụ :

So! = 1234.5			
Format(so, "#.###")		kết quả	1234.5
Format(so, "###,###")		kết quả	1,234.5
Format(so, "0.000")		kết quả	1234.5000
Format(so, "0%")		kết quả	1234500%
Format(so, "\$0.00")		kết quả	\$1234.50

- Các ký tự định dạng chuỗi ký tự :

- & : hiển thị ký tự nếu có còn không thì không hiện gì cả.
- & : hiển thị ký tự nếu có còn không thì hiện lên một ký tự trắng.
- < : đổi chuỗi sang chữ thường.
- > : đổi chuỗi sang chữ in.

Ví dụ :

Format("visual basic", ">")	trả về	"VISUAL BASIC"
Format("VISUAL BASIC", ">")	trả về	"visual basic"

#### 14.1.15 Tìm chuỗi con:

InStr[\$]([số,] chuỗi 1, chuỗi 2[, so sánh])

Trong đó :

- Số : nếu có thì nó qui định vị trí bắt đầu tìm kiếm. Không có thì tìm từ đầu.

- So sánh : là qui định phương thức tìm. Nếu so sánh là giá trị 1 thì không phân biệt chữ in với chữ thường, nếu giá trị 0 thì có phân biệt chữ in với chữ thường.
- Chuỗi 1 : chuỗi mẹ. Đây là chuỗi có thể chứa chuỗi cần tìm.
- Chuỗi 2 : chuỗi con. Đây là chuỗi cần tìm xem có được chứa trong chuỗi 1 hay không.

Hàm này trả về kết quả là giá trị số. Nếu bằng 0 nghĩa là không tìm thấy, nếu một số lớn thì không thì đó là vị trí xuất hiện chuỗi 2 trong chuỗi 1.

Ví dụ : `InStr("I Love You", "Love")` trả về kết quả là 3

`InStr("I Love You", "love", 0)` trả về kết quả 0.

### 14.2. Các hàm xử lý số :

- |              |  |
|--------------|--|
| 1. SIN(góc)  | Tính sin của góc.                        |
| 2. COS(góc)  | Tính Cosin của một góc                   |
| 3. TAN(góc)  | Tính Tang của một góc                    |
| 4. ATAN(số)  | Tính Arctang của một góc                 |
| 5. EXP(số)   | Expenential                              |
| 6. LOG(số)   | Logarithm                                |
| 7. CCUR(số)  | Chuyển đổi một số về kiểu Currency       |
| 8. CINT(số)  | Chuyển đổi một số về kiểu Integer        |
| 9. CLNG(số)  | Chuyển đổi một số về kiểu Long           |
| 10. CSNG(số) | Chuyển đổi một số về kiểu Single         |
| 11. CDBL(số) | Chuyển đổi một số về kiểu Double         |
| 12. FIX(số)  | Bỏ phần thập phân để đổi thành số nguyên |
| 13. INT(số)  | Qui tròn về số nguyên.                   |

- |               |                             |
|---------------|-----------------------------|
| 14. RND[(số)] | Tạo một số ngẫu nhiên.      |
| 15. ABS(số)   | Trị tuyệt đối               |
| 16. SGN(số)   | Dấu âm/dương của một con số |
| 17. SQR(số)   | Lấy căn bậc hai.            |

## BÀI 15. DÙNG LIST CONTROLS

Có hai loại List controls dùng trong VB6. Đó là Listbox và Combobox. Cả hai đều hiển thị một số dòng để ta có thể lựa chọn. Listbox chiếm một khung chữ nhật, nếu chiều ngang nhỏ thì có khi không hiển thị đầy đủ một dòng, nếu chiều dài không đủ để hiển thị tất cả mọi dòng thì Listbox tự động cho ta một vertical scroll bar để cho biết còn có nhiều dòng bị che và ta có thể xem các dòng ấy bằng cách dùng vertical scroll bar.

Combobox thường chỉ hiển thị một dòng, nhưng ta có thể chọn hiển thị bất cứ dòng nào khác. Combobox giống như một tập hợp của một Textbox nằm phía trên và một Listbox nằm phía dưới.



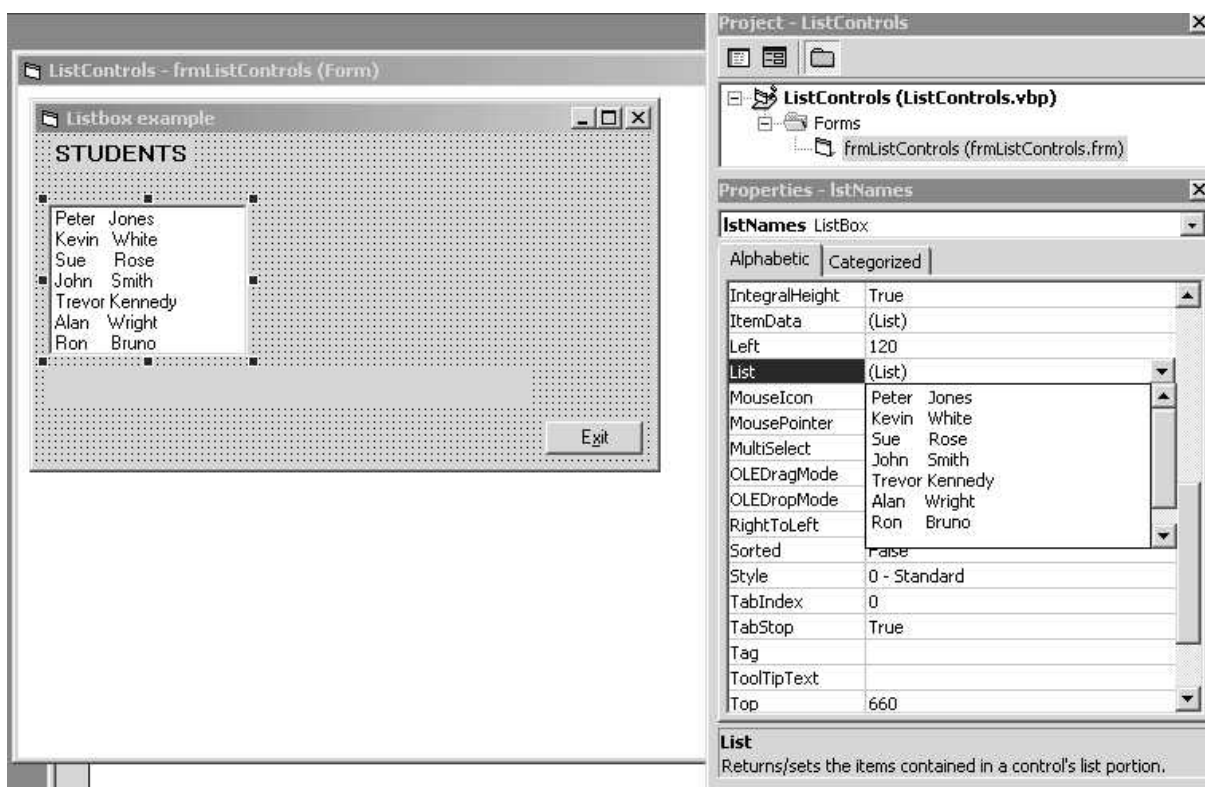
Listbox có rất nhiều công dụng vì nó rất uyển chuyển khi sử dụng. Trong bài này chúng ta sẽ xem xét các ứng dụng sau của Listbox :

- Hiển thị nhiều sự lựa chọn để người sử dụng có thể chọn bằng cách click hay drag-drop
- Những cách dùng Property Sorted
- Cách dùng Multiselect
- Dùng để hiển thị Events
- Dùng để Search hay xử lý text
- Cách dùng Itemdata song song với các Items của danh sách
- Dùng làm Queue

## 15.1. Listbox

### 15.1.1 Hiện thị nhiều sự lựa chọn

Ta hãy bắt đầu viết một chương trình gồm có một Listbox tên **lstNames** nằm trong một Form. Trong **lstNames** ta đánh vào tên của bảy người, mỗi lần xuống dòng nhớ đánh **Ctrl-Enter**, thay vì chỉ **Enter**, nếu không VB6 ngầm hiểu đã đánh xong nên tự đóng cửa sổ Property List. Các tên này là những dòng sẽ hiện ra trong Listbox khi ta bắt đầu chạy chương trình.

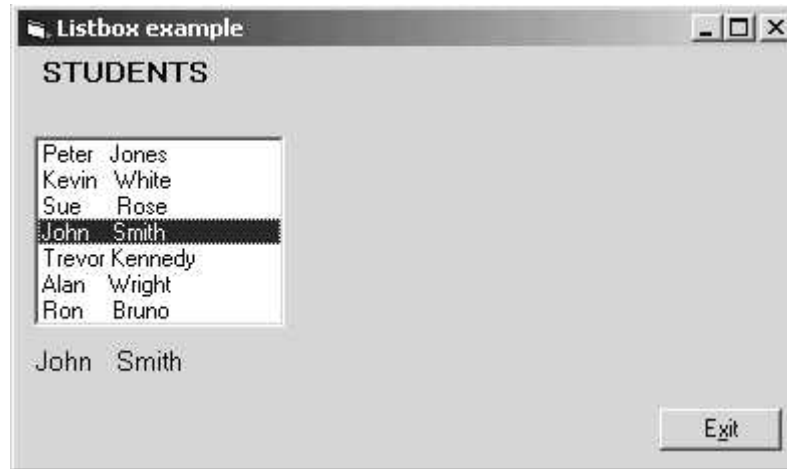


Ngoài **lstNames** ta cho thêm một Label với Caption **STUDENTS** để trang trí, và một Label khác tên **lblName**. Mỗi khi người sử dụng click lên dòng tên nào ta muốn hiển thị dòng tên ấy trong **lblName**. Sau cùng ta cho vào một CommandButton tên **CmdExit** để cho dừng dừng chương trình. Ta sẽ có chương trình như sau:

```
Private Sub lstNames_Click()  
    lblName.Caption = lstNames.List (lstNames.ListIndex)  
End Sub  
Private Sub CmdExit_Click()  
    End  
End Sub
```



Giả sử ta click vào tên John Smith trên Listbox, ta sẽ thấy tên ấy cũng được hiển thị trong Label lblName.



Trong ví dụ này, Listbox lstNames có 7 dòng (**Items**). Con số Items này là Property **ListCount** của Listbox. Các Items của Listbox được đếm từ **0 đến ListCount-1**. Trong trường hợp này là từ 0 đến 6.

Khi người sử dụng click lên một dòng, Listbox sẽ generate **Event lstNames\_Click**. Lúc bấy giờ ta có thể biết được người sử dụng vừa mới Click dòng nào bằng cách hỏi Property **ListIndex** của lstNames, nó sẽ có value từ 0 đến ListCount-1. Lúc chương trình mới chạy, chưa ai Click lên Item nào của Listbox thì ListIndex = -1.

Những Items trong Listbox được xem như một mảng chuỗi ký tự. Array này được gọi là **List**. Do đó, ta nói đến Item thứ nhất của Listbox lstNames bằng cách viết **lstNames.List(0)**, và tương tự như vậy, Item cuối cùng là **lstNames.List(lstNames.ListCount-1)**.

Ta có thể nói đến item vừa được Clicked bằng hai cách:

- lstNames.List(lstNames.ListIndex)
- lstNames.text.

### 15.1.2 Save content của Listbox

Bây giờ để lưu trữ nội dung của lstNames, ta thêm một CommandButton tên CmdSave. Ta sẽ viết code để khi người sử dụng click nút CmdSave chương trình sẽ mở một Output text file và viết mọi items của lstNames vào đó:

```
Private Sub CmdSave_Click()  
    Dim i, FileName, FileNumber  
    FileName = App.Path  
    ' Make sure FileName ends with a backslash  
    If Right(FileName, 1) <> "\" Then FileName = FileName & "\"  
    FileName = FileName & "MyList.txt" 'output text file MyList.txt  
    ' Obtain an available filename from the operating system  
    FileNumber = FreeFile  
    ' Open the FileName as an output file  
    Open FileName For Output As FileNumber  
    ' Now iterate through each item of lstNames  
    For i = 0 To lstNames.ListCount - 1  
        ' Write the List item to file  
        Print #FileNumber, lstNames.List(i)  
    Next  
    Close FileNumber ' Close the output file  
End Sub
```

**App** là một Object đặc biệt đại diện cho chính chương trình đang chạy. Ở đây ta dùng Property **Path** để biết lúc chương trình đang chạy thì thực thi module EXE của nó nằm ở đâu. Lý do là ta thường để các files liên hệ cần thiết cho chương trình lẫn quản hoặc ngay trong folder của chương trình hay trong một subfolder, chẳng hạn như **data**, **logs**, ..v.v.. App còn có một số Properties khác cũng rất hữu dụng như **PrevInstance**, **Title**, **Revision** ..v.v.

Nếu mới khởi động một chương trình mà thấy `App.PrevInstance = True` thì lúc bấy giờ cũng có một copy khác của chương trình đang chạy. Nếu cần ta End program này để tránh chạy 2 bản sao của chương trình cùng một lúc.

`App.Title` và `App.Revision` cho ta tin tức về Title và Revision của chương trình đang chạy. Để viết ra một Text file ta cần phải Open nó trong **mode Output** và khai báo từ đây trở đi sẽ dùng một con số (`FileNumber`) để đại diện tập tin thay vì dùng chính `FileName`. Để tránh dùng một `FileNumber` đã hiện hữu, tốt nhất ta hỏi xin hệ điều hành cung cấp cho mình một con số chưa ai dùng bằng cách gọi **Function FreeFile**. Con số `FileNumber` này còn được gọi là **FileHandle** (Handle là tay cầm). Sau khi ta `Close FileNumber` con số này trở nên FREE và hệ điều hành sẽ có thể dùng nó lại.

Do đó chúng ta phải tránh gọi FreeFile liên tiếp hai lần, vì OS sẽ cho chúng ta cùng một con số. Tức là, sau khi gọi FreeFile phải dùng nó ngay bằng cách Open một File rồi mới gọi FreeFile lần kế để có một con số khác.

Để ý cách dùng chữ **Input, Output** cho files là relative (tương đối) với vị trí của chương trình (nó nằm trong memory của computer). Do đó từ trong memory viết ra đĩa cứng thì gọi là Output. Ngược lại đọc từ một Text file nằm trên hard disk vào memory cho chương trình ta thì gọi là Input.

### 15.1.3 Load một Text file vào Listbox

Trong bài này, thay vì đánh các Items của Listbox vào Property List của lstNames ta có thể populate (làm đầy) lstNames bằng cách đọc các Items từ một Text file. Ta thử thêm một CommandButton tên CmdLoad. Ta sẽ viết code để khi người sử dụng click nút CmdLoad chương trình sẽ mở một Input text file và đọc từng dòng để bỏ vào lstNames:

```
Private Sub CmdLoad_Click()  
    Dim i, FileName, FileNumber, anItem  
    ' Obtain Folder where this program's EXE file resides  
    FileName = App.Path  
    ' Make sure FileName ends with a backslash  
    If Right(FileName, 1) <> "\" Then FileName = FileName & "\"  
    FileName = FileName & "MyList.txt"  
    ' Obtain an available filename from the operating system  
    FileNumber = FreeFile  
    ' Open the FileName as an input file  
    Open FileName For Input As FileNumber  
    lstNames.Clear ' Clear the Listbox first  
    ' Now read each line until reaching End-Of-File  
    Do While NOT EOF(FileNumber)  
        Line Input #FileNumber, anItem ' Read a line from the file  
        lstNames.AddItem anItem ' Add this item to the lstNames  
    Loop  
    Close FileNumber ' Close the input file  
End Sub
```

Để **đọc từ một Text file** ta cần phải Open nó trong **mode Input**.

Trước khi populate lstNames ta cần phải xóa tất cả mọi items có sẵn bên trong. Để thực hiện việc đó ta dùng **method Clear** của Listbox.

Sau đó ta dùng **method AddItem** để cho thêm từng dòng vào trong Listbox. By default, nếu ta không nói nhét vào ở chỗ dòng nào thì AddItem nhét Item mới vào dưới chót của Listbox.

Nếu muốn nhét dòng mới vào ngay trước item thứ 5 (ListIndex = 4), ta viết:

```
lstNames.AddItem newItemString, 4 ' newItemString contains  
' To insert a new Item at the beginning of the Listbox, write:  
lstNames.AddItem newItemString, 0
```

Nhớ là mỗi lần chúng ta Add một Item vào Listbox thì ListCount của Listbox tăng 1. Muốn xóa một item từ Listbox ta dùng **method RemoveItem**, ví dụ như muốn xóa item thứ ba (ListIndex=2) của lstNames, ta viết:

```
lstNames.RemoveItem 2
```

Mỗi lần chúng ta RemoveItem từ Listbox the ListCount của Listbox giảm đi một đơn vị 1. Do đó nếu chúng ta dùng Test dựa vào ListCount của một ListBox để nhảy ra khỏi một Loop thì phải coi chừng tránh làm cho value ListCount thay đổi trong Loop vì AddItem hay RemoveItem.

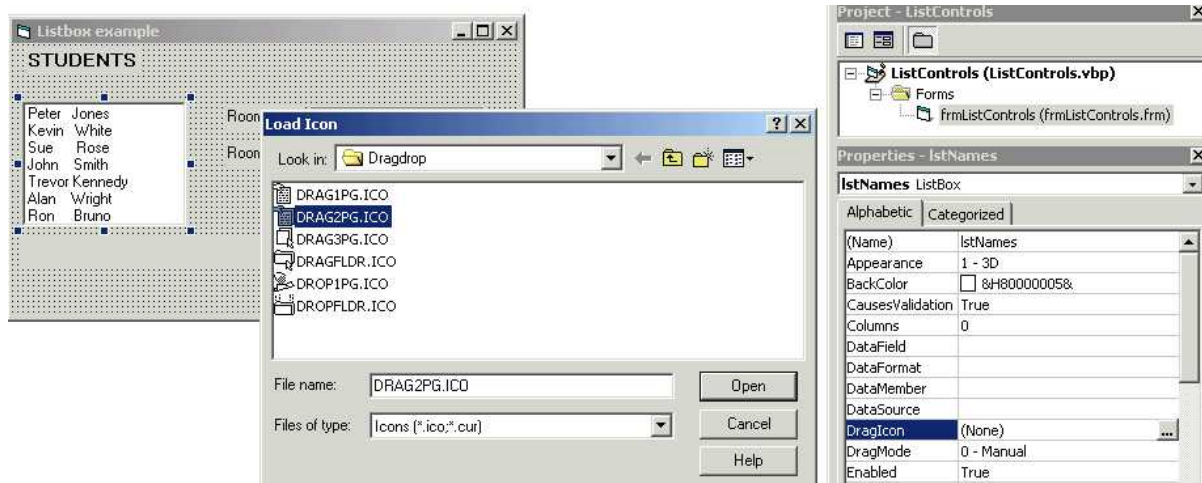
Ta đọc từng dòng của một Text file bằng cách dùng **Line Input #FileNumber**. Khi đọc đến cuối File, system sẽ cho ta value EOF(FileNumber) = True. Ta dùng value ấy để cho chương trình nhảy ra khỏi While.. Loop.

**Câu Do While NOT EOF(FileNumber) có nghĩa Trong khi chưa đến End-Of-File của Text File đại diện bởi FileNumber** thì đọc từng dòng và bỏ vào Listbox.

## 15.2. Drag-Drop

Ta đã xem qua Click Event của Listbox. Bây giờ để dùng Drag-Drop cho Listbox chúng ta hãy đặt 2 Labels mới lên Form. Cái thứ nhất tên gì cũng được nhưng có Caption là Room A. Hãy gọi Label thứ hai là lblRoom và cho Property BorderStyle của nó bằng Fixed Single. Kế đến select cả hai Labels (Click a Label then hold down key Ctrl while clicking the second Label) rồi click copy và paste lên Form. VB6 sẽ cho chúng ta Array của hai lblRoom labels.

Để cho lstNames một DragIcon, chúng ta click lstNames, click Property DragIcon để pop-up một dialog cho chúng ta chọn một dragdrop icon từ folder C:\Program Files\Microsoft Visual Studio\Common\Graphics\Icons\Dragdrop, chẳng hạn như DRAG2PG.ICO:

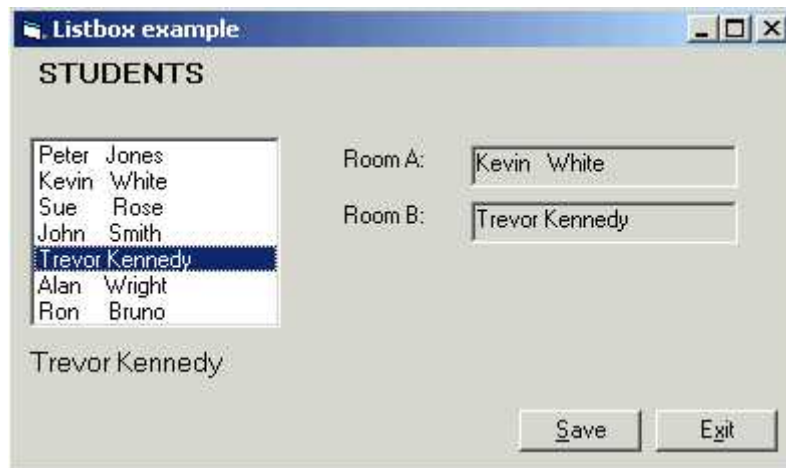


Ta sẽ dùng Event MouseDown của lstNames để pop-up DragIcon hình 2 trang giấy cho UserDrag nó qua bên phải rồi bỏ xuống lên một trong hai lblRoom. Khi DragIcon rơi lên lblRoom, lblRoom sẽ generate Event DragDrop. Ta sẽ dùng Event DragDrop này để assignproperty Text của Source (tức là lstNames, mục control từ nó phát xuất Drag action) vào Property Caption của lblRoom. Lưu ý vì ở đây ta dùng cùng một tên cho cả hai lblRoom nên chỉ cần viết code ở một chỗ để handle Event DragDrop.

```
Private Sub lstNames_MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)
    ' Start Pop-up DragIcon and start Drag action
    lstNames.Drag
End Sub

Private Sub lblRoom_DragDrop(Index As Integer, Source As Control, X As Single, Y As Single)
    ' Assign Property Text of Source (i.e. lstNames) to Caption
    lblRoom(Index).Caption = Source.Text
End Sub
```

Kết quả sau khi Drag hai tên từ Listbox qua Labels là như sau:



### 15.3. Dùng Property Sorted

Trong ví dụ trên ta có thể quyết định vị trí của một Item mới khi ta nhét nó vào Listbox. Đôi khi ta muốn các Items của Listbox được tự động sắp theo thứ tự Alphabet. Chúng ta có thể set **Property Sorted = True** để thực hiện chuyện này. Có một giới hạn là chúng ta phải cho Property Sorted một value (True hay False) trong lúc design, chớ trong khi chạy chương trình chúng ta không thể làm cho Property Sorted của Listbox thay đổi.

Giả dụ ta muốn sort các Items của một Listbox khi cần. Vậy thì ta làm sao? Giải pháp rất đơn giản. Chúng ta tạo một Listbox tên lstTemp chẳng hạn. Cho nó Property Visible= False (để không ai thấy nó) và Property Sorted=True. Khi cần sort lstNames chẳng hạn, ta copy content của lstNames bỏ vào lstTemp, đoạn Clear lstNames rồi copy content (đã được sorted) của lstTemp trở lại lstNames.

Lưu ý là ta có thể AddItem vào một Listbox với Property Sorted=True, nhưng không thể xác định nhét Item vào trước dòng nào, vì vị trí của các Items do Listbox quyết định khi nó sort các Items.

Ta hãy cho thêm vào Form một CommandButton mới tên **CmdSort** và viết code cho Event Click của nó như sau:

```
Private Sub CmdSort_Click()  
    Dim i  
    lstTemp.Clear ' Clear temporary Listbox  
    ' Iterate though every item of lstNames  
    For i = 0 To lstNames.ListCount - 1  
        ' Add the lstNames item to lstTemp
```

```
lstTemp.AddItem lstNames.List(i)
Next
lstNames.Clear ' Clear lstNames
' Iterate though every item of lstTemp
For i = 0 To lstTemp.ListCount - 1
    ' Add the lstTemp item to lstNames
    lstNames.AddItem lstTemp.List(i)
Next
lstTemp.Clear ' Tidy up - clear temporary Listbox
End Sub
```

Nhân tiện, ta muốn có option để sort các tên theo FirstName hay Surname. Việc này hơi rắc rối hơn một chút, nhưng nguyên tắc vẫn là dùng cái sorted Listbox vô hình tên lstTemp.

Chúng ta hãy đặt lên phía trên lstName hai Labels mới tên lblFirstName và lblSurName và cho chúng Caption "FirstName" và "SurName".

Từ đây ta Load file "MyList.txt" vào lstNames bằng cách Click button CmdLoad chứ không Edit Property List của lstNames để enter Items lúc design nữa. Ngoài ra ta dùng dấu phẩy (,) để tách FirstName khỏi SurName trong mỗi tên chứa trong file MyList.txt. Content của file MyList.txt bây giờ trở thành như sau:

```
Peter, Jones
Kevin, White
Sue, Rose
John, Smith
Trevor, Kennedy
Alan, Wright
Ron, Bruno
```

Ta sẽ sửa code trong Sub CmdLoad\_Click lại để khi nhét tên vào lstNames, FirstName và SurName mỗi thứ chiếm 10 characters.

Để các chữ trong Items của lstNames sắp dòng ngay ngắn ta đổi Font của lstNames ra Courier New. Courier New là một loại Font mà chiều ngang của tất cả các chữ **là như nhau**, trong khi hầu hết các Fonts khác như Arial, Times Roman ..v.v. là **Proportional Spacing**, có nghĩa là độ rộng các ký tự là khác nhau.

Listing mới của Sub CmdLoad\_Click trở thành như sau:

```
Private Sub CmdLoad_Click()  
    Dim i, Pos  
    Dim FileName, FileNumber, anItem  
    Dim sFirstName As String*10 ' fixed length string of 10 chars  
    Dim sSurName As String * 10 ' fixed length string of 10 chars  
    ' Obtain Folder where this program's EXE file resides  
    FileName = App.Path  
    ' Make sure FileName ends with a backslash  
    If Right(FileName, 1) <> "\" Then FileName = FileName & "\"  
    FileName = FileName & "MyList.txt"  
    ' Obtain an available filenumber from the operating system  
    FileNumber = FreeFile  
    ' Open the FileName as an input file , using FileNumber  
    Open FileName For Input As FileNumber  
    lstNames.Clear ' Clear the Listbox first  
    ' Now read each line until reaching End-Of-File  
    Do While Not EOF(FileNumber)  
        Line Input #FileNumber, anItem ' Read a line from the file  
        ' Now separate FirstName from SurName  
        Pos = InStr(anItem, ",") ' Locate the comma ", "  
        ' The part before "," is FirstName  
        sFirstName = Left(anItem, Pos - 1)  
        sFirstName = Trim(sFirstName) ' Trim off any blank spaces  
        ' The part after "," is SurName  
        sSurName = Mid(anItem, Pos + 1)  
        sSurName = Trim(sSurName) ' Trim off any blank spaces  
        lstNames.AddItem sFirstName & sSurName  
        ' Add this item to the bottom of lstNames  
    Loop  
    Close FileNumber ' Close the input file  
End Sub
```

Vì FirstName nằm ở bên trái của mỗi Item nên sort theo FirstName cũng giống như sort cả Item. Việc ấy ta đã làm bằng Sub CmdSort\_Click rồi, do đó khi người sử dụng click Label lblFirstName ta chỉ cần gọi CmdSort\_Click như sau:

```
Private Sub lblFirstName_Click()  
    CmdSort_Click  
End Sub
```



Để sort theo SurName ta cần phải tạm thời để SurName qua bên trái của Item trước khi bỏ vào lstTemp. Ta thực hiện chuyện này bằng cách hoán chuyển vị trí của FirstName và SurName trong Item trước khi bỏ vào lstTemp. Sau đó, khi copy các Items từ lstTemp để đặt vào lại lstNames ta lại nhớ hoán chuyển FirstName và SurName để chúng nằm đúng lại vị trí. Code để sort tên theo SurName cũng giống như CmdSort\_Add nhưng sửa đổi chút ít như sau:

```
Private Sub lblSurName_Click()  
    Dim i, anItem  
    Dim sFirstName As String*10 'fixed length string of 10 chars  
    Dim sSurName As String * 10 ' fixed length string of 10 chars  
    lstTemp.Clear ' Clear temporary Listbox  
    ' Iterate though every item of lstNames  
    For i = 0 To lstNames.ListCount - 1  
        anItem = lstNames.List(i)  
        ' Identify FistName and SurName  
        sFirstName = Left(anItem, 10)  
        sSurName = Mid(anItem, 11)  
        ' Swap FirstName/SurName positions before adding to lstTemp  
        lstTemp.AddItem sSurName & sFirstName  
    Next  
    lstNames.Clear ' Clear lstNames  
    ' Iterate though every item of lstTemp  
    For i = 0 To lstTemp.ListCount - 1  
        anItem = lstTemp.List(i)  
        sSurName = Left(anItem, 10) ' SurName now is on the left  
        sFirstName = Mid(anItem, 11)  
        ' Add FirstName/SurName in correct positions to lstNames  
        lstNames.AddItem sFirstName & sSurName  
    Next  
    lstTemp.Clear ' Tidy up - clear temporary Listbox  
End Sub
```

Các Items trong lstNames sorted theo SurName hiện ra như sau:



## BÀI 16. TỰ TẠO OBJECT

Từ trước đến giờ, ta lập trình VB6 bằng cách thiết kế các Forms rồi viết codes để xử lý các Events của những controls trên Form khi người sử dụng click một Button hay Listbox, ..v.v.. Nói chung, cách ấy cũng hữu hiệu để triển khai chương trình, nhưng nếu ta có thể hưởng được các lợi ích sau đây thì càng tốt hơn :

- Dùng lại được code đã viết trước đây trong một dự án khác
- Dễ nhận diện được một lỗi (error) phát xuất từ đâu
- Dễ triển khai một dự án lớn bằng cách phân phối ra thành nhiều dự án nhỏ
- Dễ bảo trì

Lập trình theo hướng đối tượng là thiết kế các bộ phận phần mềm của chương trình, gọi là **Objects** sao cho mỗi bộ phận có thể tự lo liệu công tác của nó giống như một module **làm việc độc lập**. Câu hỏi đặt ra là các **Sub** hay **Function** mà chúng ta đã từng viết để xử lý từng giai đoạn trong chương trình có thể đảm trách vai trò của một module độc lập không?

Có một cách định nghĩa khác cho Object là một Object gồm có data structure và các Subs/Functions làm việc trên các data ấy. Thông thường, khi ta dùng Objects không cần giám sát chúng thực hiện như thế nào, ngược lại nếu khi có sự cố gì thì ta muốn chúng báo cáo cho ta biết.

Trong VB6, các Forms, Controls hay ActiveX là những Objects mà ta vẫn sử dụng. Lấy ví dụ như Listbox. Một Listbox tự quản lý các items hiển thị bên trong nó. Ta biết listbox List1 đang có bao nhiêu items bằng cách hỏi List1.ListCount. Ta biết item nào vừa mới được selected bằng cách hỏi List1.ListIndex. Ta thêm một item vào listbox bằng cách gọi method AddItem của List1, ..v.v.. Nói cho đúng ra, Object là một thực thể của một Class. Nếu Listbox là một Class thì List1, List2 là các thực thể của Listbox.

Ngay cả một form tên frmMyForm mà ta viết trong VB6 chẳng hạn, nó cũng là một Class. Thông thường ta dùng thẳng frmMyForm như sau:

```
frmMyForm.Show
```

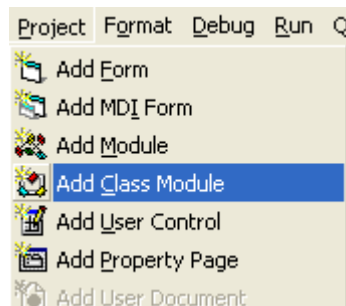
Trong trường hợp này thật ra frmMyForm tuy là một Class nhưng được dùng y như một Object. Nếu cần thiết, ta có thể tạo ra hai, ba Objects của Class frmMyForm cùng một lúc như trong ví dụ sau:

```
Dim firstForm As frmMyForm
Dim secondForm As frmMyForm
Set firstForm = New frmMyForm
Set secondForm = New frmMyForm
firstForm.Show
secondForm.Show
```

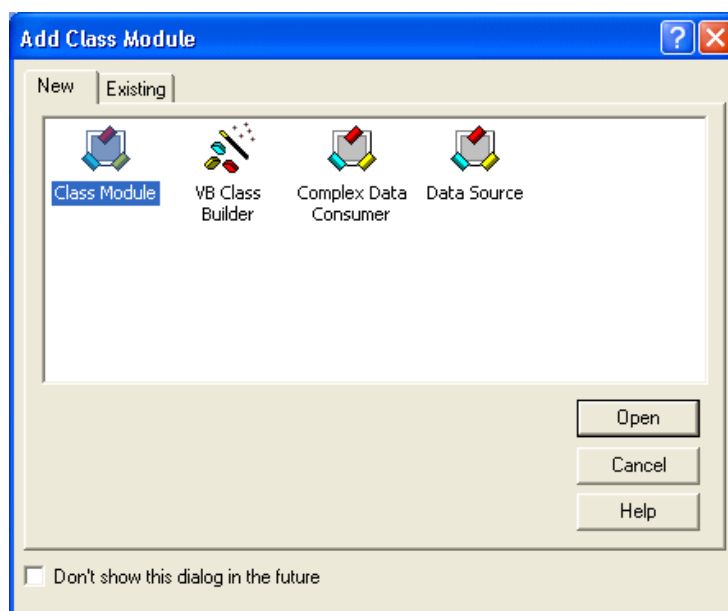
Trong ví dụ trên ta khai báo firstForm và secondForm là những Objects của Class frmMyForm. Sau đó ta làm nên (**instantiate**) các Objects firstForm và secondForm bằng statements **Set... = New...** firstForm và secondForm còn được gọi là các **instances** của Class frmMyForm. Class giống như cái khuôn, còn Objects giống như những cái bánh làm từ khuôn ấy. Chắc chúng ta đã để ý thấy trong VB6 từ dùng hai từ Class và Object lẫn lộn nhau. Điều này cũng không quan trọng, miễn là chúng ta nắm vững ý nghĩa của chúng.

VB6 có yểm trợ **Class** mà ta có thể triển khai và instantiate các Objects của nó khi dùng. Một Class trong VB6 có chứa **data** riêng của nó, có những **Subs** và **Functions** mà ta có thể gọi. Ngoài ra Class còn có thể **Raise Events**, tức là báo cho ta biết khi chuyện gì xảy ra bên trong nó. Cũng giống như Event Click của CommandButton, khi người sử dụng clicks lên button thì nó Raise Event Click để cho ta xử lý trong Sub myCommandButton\_Click(), chẳng hạn. Class trong VB6 không có hỗ trợ Visual components, tức là không có chứa những controls như TextBox, Label .v.v.. Tuy nhiên, ta có thể lấy những control có sẵn từ bên ngoài rồi đưa cho Object của Class dùng.

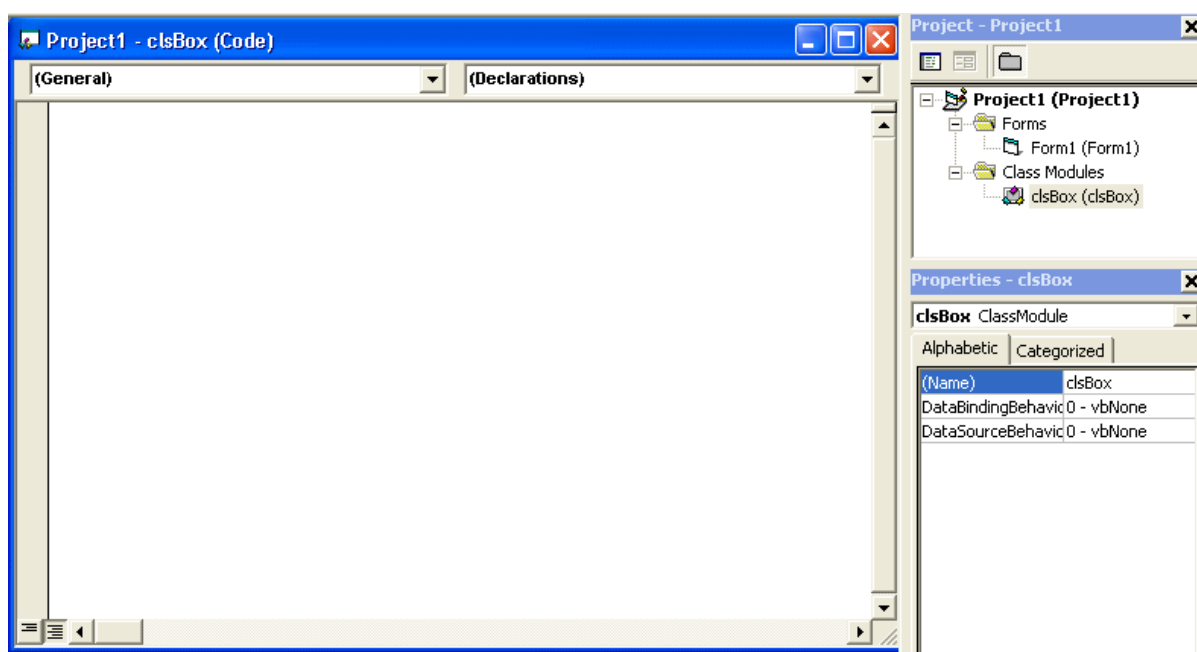
Bây giờ chúng ta hãy bắt đầu viết một Class. Chúng ta hãy mở một Project mới loại Standard EXE Visual Basic. Sau đó dùng Menu Command chọn **Add Class Module**:



Khi Add Class Module dialog hiện ra chọn **Class Module** và click Open.



Chúng ta sẽ thấy mở ra một khung trống và Project Explorer với Properties Window. Trong Properties Window, hãy sửa Name property của Class thành clsBox như dưới đây:



Kế đó đánh vào những dòng code dưới đây, trong đó có biểu diễn cách dùng Class clsBox.

```
Option Explicit
Private mX As Integer
Private mY As Integer
Private mWidth As Integer
Private mHeight As Integer
```

```
Public Property Let X(ByVal vValue As Integer)
    mX = vValue
End Property

Public Property Get X() As Integer
    X = mX
End Property

Public Property Let Y(ByVal vValue As Integer)
    mY = vValue
End Property

Public Property Get Y() As Integer
    Y = mY
End Property

Public Property Let Width(ByVal vValue As Integer)
    mWidth = vValue
End Property

Public Property Get Width() As Integer
    Width = mWidth
End Property

Public Property Let Height(ByVal vValue As Integer)
    mHeight = vValue
End Property

Public Property Get Height() As Integer
    Height = mHeight
End Property

Public Sub DrawBox(Canvas As Object)
    Canvas.Line (mX, mY)-(mX + mWidth, mY + mHeight), , B
End Sub

Public Sub ClearBox(Canvas As Object)
```

```
Canvas.Line (mX, mY) - (mX + mWidth, mY + mHeight),  
Canvas.BackgroundColor, B  
End Sub
```

Class `clsBox` có 4 Properties: `X`, `Y`, `Width` và `Height`. Ta sẽ dùng một ví dụ cụ thể là một `Box` từ `clsBox`. Mỗi `Box` có tọa độ (`X`,`Y`) và kích thước chiều rộng và chiều cao (`width`, `height`) của nó. Thật ra ta có thể dùng `Public statement` để khai báo các biến `X`, `Y`, `Width` và `Height`. Nhưng ở đây ta cố ý declare chúng là `Private`, dưới dạng `mX`, `mY`, `mWidth` và `mHeight`. Khi ta muốn thay đổi các trị số của chúng, ta sẽ dùng cùng một cách viết code như bình thường (ví dụ: `myBox.X = 80`). Nhưng khi chương trình xử lý `assignment statement` ấy, nó sẽ thực thi một loại `method` (giống như `Sub`) gọi là **Property Let X (vValue)**. Ta thấy ở đây `vValue` được assigned cho `mX` (i.e. `mX = vValue`), cái `Private variable` của `X`. Như thế công việc này cũng chẳng khác gì sửa đổi một `Public variable X`. Tuy nhiên, ở đây ta có thể viết thêm code trong `Property Let X` để nó làm gì cũng được.

Mỗi lần chúng ta dùng `Property Window` để edit `Font size`, `forcolor` hay `backcolor` thì chẳng những các `properties` ấy của `Label` thay đổi, mà kết quả của sự thay đổi được có hiệu lực ngay lập tức, nghĩa là `Label` được hiển thị trở lại với trị số mới của `property`. Đó là vì trong `method Property` có cả code bảo `Label` thực hiện `redisplay`.

Ngược lại, khi ta dùng `property X` của `Object myBox`, không phải ta chỉ đọc trị số thôi mà còn thực thi cả cái `method Property Get X`. Nói tóm lại, `Property` cho ta cơ hội để thực thi một `method` mỗi khi người sử dụng đọc hay viết trị số `variable` ấy.

Ví dụ như nếu ta muốn kiểm soát để chỉ chấp nhận trị số tọa độ `X` mới khi nó không phải là số âm. Ta sẽ sửa `Property Let X` lại như sau:

```
Public Property Let X(ByVal vValue As Integer)  
    If (vValue >= 0) Then  
        mX = vValue  
    End If  
End Property
```

`Property` có thể là **Read Only** hay **Write Only**. Nếu muốn một `Property` là `Read Only` thì ta không cung cấp `Property Let`. Nếu muốn một `Property` là `Write Only` thì ta không cung cấp `Property Get`. Ngoài ra nếu làm việc với **Object**, thay vì `Data type` thông thường, thì ta phải dùng **Property Set**, thay vì `Property Let`.

Ví dụ ta cho clsBox một Property mới, gọi là Font dùng object của class stdFont của VB6. Trong clsBox ta declare một Private variable mFont và viết một **Property Set Font** như sau:

```
Private mFont As StdFont
Public Property Set Font (ByVal newFont As StdFont)
    Set mFont = newFont
End Property
```

Ta sẽ dùng property Font của myBox (thuộc Class clsBox) như sau:

```
' Declare an object of Class StdFont of VB6
Dim myFont As StdFont
Set myFont = New StdFont
myFont.Name = "Arial"
myFont.Bold = True
Dim myBox As clsBox
Set myBox = New clsBox
Set myBox.Font = myFont ' Call the Property Set method
```

Class clsBox có hai Public Subs, **DrawBox** và **ClearBox**. ClearBox cũng vẽ một box như DrawBox, nhưng nó dùng BackColor của màn ảnh (canvas), nên coi như xóa cái box có sẵn. Do đó, nếu muốn, chúng ta có thể sửa Sub DrawBox lại một chút để nhận một Optional draw color như sau:

```
Public Sub DrawBox(Canvas As Object, Optional fColor As Long)
    If IsMissing(fColor) Then
        Canvas.Line (mX, mY)-(mX + mWidth, mY + mHeight), , B
    Else
        Canvas.Line (mX, mY)-(mX + mWidth, mY + mHeight), fColor, B
    End If
End Sub
```

Trong ví dụ trên, Optional parameter fColor được tested bằng function **IsMissing**. Nếu fColor là BackColor của canvas thì ta sẽ có hiệu quả của ClearBox.

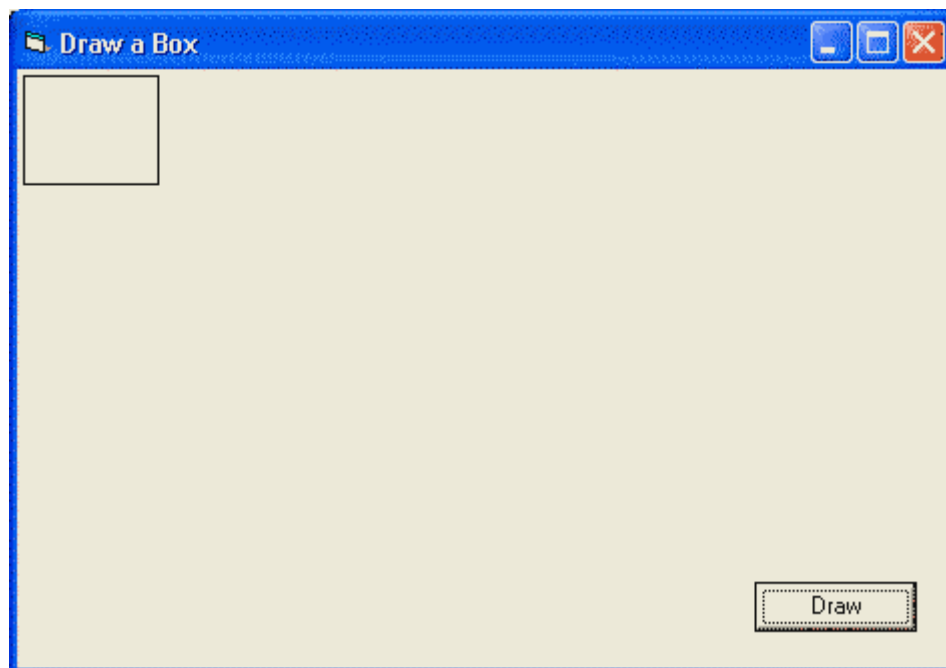
Trong form chính của chương trình dùng để test clsBox, mỗi khi ta refer đến một object thuộc class clsBox, IDE Intellisense sẽ hiển thị các Properties và Subs/Functions của clsBox như trong hình dưới đây:



```
CmdDraw

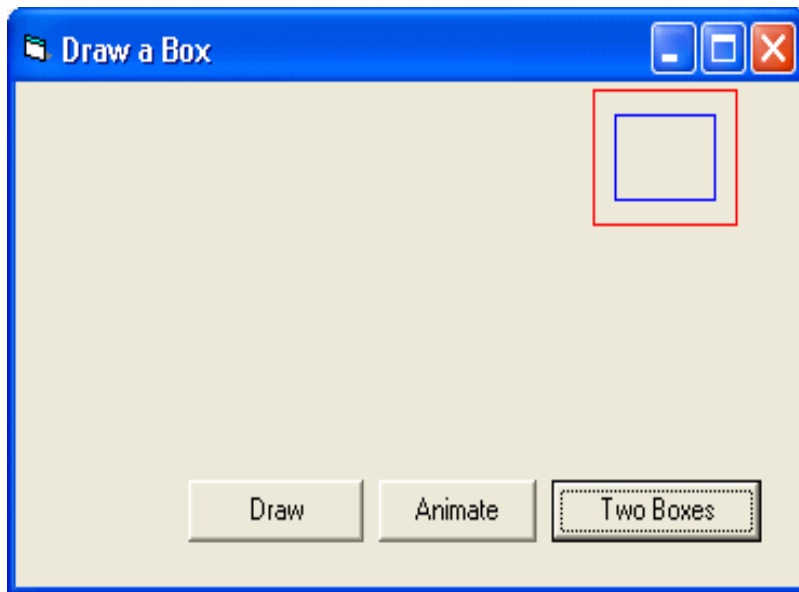
Private Sub CmdDraw_Click()
    Dim myBox As clsBox
    Set myBox = New clsBox
    With myBox
        .X = 50
        .Y = 50
        .Width = 1000
    End With
    myBox.DrawBox
End Sub
```

Trong chương trình này, mỗi khi ta click nút **Draw** thì một Box được instantiate, cho tọa độ X,Y và kích thước Width, Height, rồi được vẽ ra ngay trên form. Chữ **Me** trong code nói đến chính cái form **frmClass**.



Để cho chương trình thú vị hơn, khi người sử dụng clicks nút **Animate**, ta sẽ cho một box màu đỏ chạy từ trái qua phải.

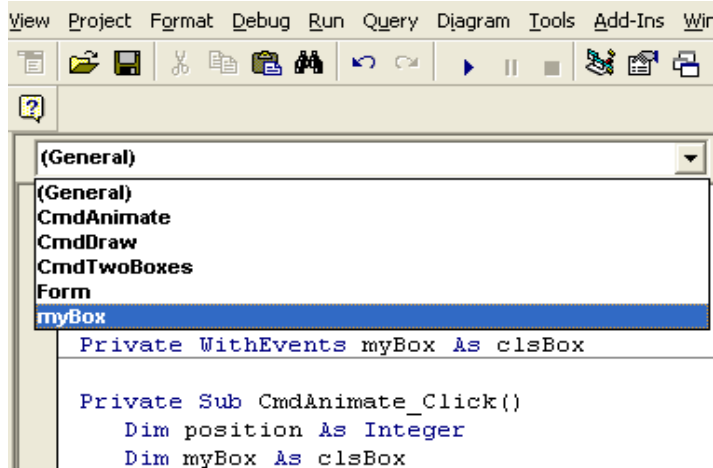
Khi người sử dụng clicks nút **Two Boxes** ta sẽ vẽ hai boxes, hộp trong màu xanh, hộp ngoài màu đỏ, và cho chúng chạy từ trái sang phải. Ở đây ta biểu diễn cho thấy mình muốn instantiate bao nhiêu boxes từ clsBox cũng được, và dĩ nhiên mỗi box có một bộ properties với giá trị riêng của nó.



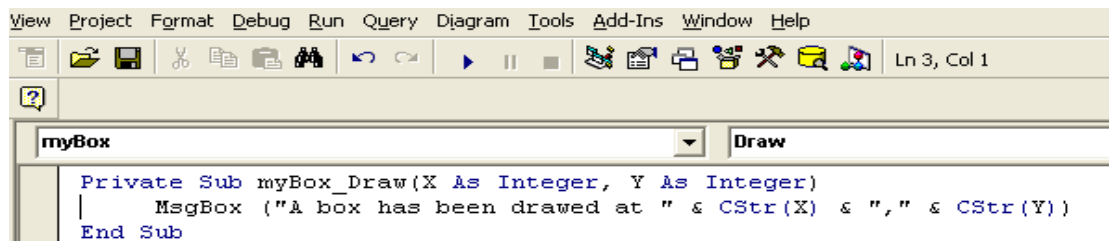
Ta có thể lập trình để cho Object báo cáo chương trình chủ của nó khi có một biến cố (Event) xảy ra bên trong Class.

Ta thử khai báo một Event tên Draw trong clsBox, và viết code để mỗi khi Sub DrawBox executes thì Class sẽ Raise một event Draw.

```
Public Event Draw(X As Integer, Y As Integer)
Public Sub DrawBox(Canvas As Object, Optional fColor As Long)
    If IsMissing(fColor) Then
        Canvas.Line (mX, mY)-(mX + mWidth, mY + mHeight), , B
    Else
        Canvas.Line (mX, mY)-(mX + mWidth, mY + mHeight), fColor, B
    End If
    RaiseEvent Draw(mX, mY)
End Sub
```

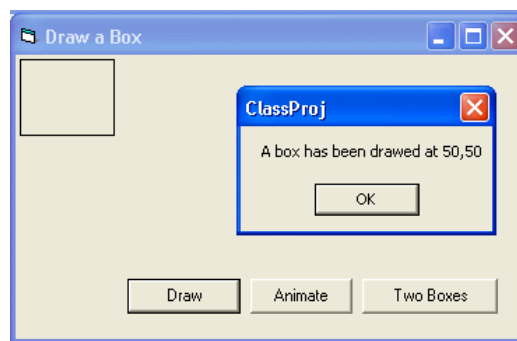


Bây giờ, trong frmClass thay vì chỉ declare Dim myBox as clsBox, ta sẽ declare Private WithEvents myBox as clsBox. Ngay sau đó, chữ myBox sẽ hiện ra trong danh sách các Object có hỗ trợ Event của frmClass. Kế đó ta sẽ viết code để handle Event Draw của myBox, tức là ta cung cấp code cho Private Sub myBox\_Draw (X as Integer, Y as Integer). Ở đây ta chỉ hiển thị một thông điệp báo cáo một hộp vừa được vẽ ở đâu.



```
View Project Format Debug Run Query Diagram Tools Add-Ins Window Help
Ln 3, Col 1
myBox Draw
Private Sub myBox_Draw(X As Integer, Y As Integer)
    MsgBox ("A box has been drawn at " & CStr(X) & ", " & CStr(Y))
End Sub
```

Khi chạy chương trình, mỗi lần một clsBox object thực hiện Sub DrawBox ta sẽ thấy frmClass hiển thị một message giống như dưới đây.



Nhớ rằng, ta declare một Object với WithEvents khi ta muốn handle các Events của nó. Trong ví dụ trên frmClass là chủ của myBox và nó handles Event Draw của myBox. Tương tự như vậy, ngay cả ở bên trong một Class, nếu Class ấy được giao cho một Object có thể Raise Events (ví dụ như TextBox, ListBox, Timer .v.v..), chúng ta cũng có thể khai báo Object ấy với các sự kiện kèm theo để nó có thể quản lý các Events của Object.

Trong ví dụ dưới đây ta viết codes này trong một Class đã được giao cho một Textbox khi form chính gọi Sub InitObject để đưa cho Object một TextBox:

```
Private WithEvents mTextBox As TextBox
Public Sub InitObject(givenTextBox As TextBox)
    Set mTextBox = givenTextBox
End Sub
Private Sub mTextBox_KeyPress(KeyAscii As Integer)
    ' Place your code here to handle this event
End Sub
```

## BÀI 17. DEBUG

Bugs là những lỗi của chương trình mà ta phát hiện khi chạy nó. Debug là công việc loại tất cả những lỗi trong chương trình để nó chạy êm xuôi trong mọi tình huống.

Thông thường muốn sửa một cái bug nào trước hết ta phải tìm hiểu lý do khiến nó xuất hiện. Một khi đã biết được duyên cớ rồi ta sẽ nghĩ ra cách giải quyết. Nói chung, có hai loại bugs : hoặc là chương trình không làm đúng chuyện cần phải làm vì lập trình viên hiểu lầm **Specifications** hay được cho tin tức sai lạc, hoặc là chương trình bỏ sót chi tiết cần phải có. Trường hợp này ta giải quyết bằng cách giảm thiểu sự hiểu lầm qua sự nâng cấp khả năng truyền thông.

Chương trình không thực hiện đúng như ý lập trình viên muốn, tức là lập trình viên muốn một đằng mà bảo chương trình làm một ngã vì vô tình không viết chương trình đúng cách. Trường hợp này ta giải quyết bằng cách dùng những Software Tools (kể cả ngôn ngữ lập trình) thích hợp, và có những quá trình làm việc có hệ thống.

Có nhiều yếu tố ảnh hưởng đến chất lượng của một chương trình như chức năng của chương trình, cấu trúc của các bộ phận, kỹ thuật lập trình và phương pháp debug. Debug không hẳn nằm ở giai đoạn cuối của dự án mà tùy thuộc rất nhiều vào các yếu tố kể trên trong mọi giai đoạn triển khai.

### 17.1. Đặc tả chương trình (Program Specifications)

Dầu chương trình lớn hay nhỏ, trước hết ta phải xác định rõ ràng và tỉ mỉ nó cần phải làm gì, bao nhiêu người dùng, mạng như thế nào, database lớn bao nhiêu, phải chạy nhanh đến mức nào .v.v..

Có nhiều chương trình phải bị thay đổi nữa chừng vì lập trình viên hiểu lầm điều khách hàng muốn. Do đó trong sự liên hệ với khách hàng ta cần phải hỏi đi, hỏi lại, phản hồi với khách hàng nhiều lần điều ta hiểu bằng thư từ, tài liệu, để khách xác nhận là ta biết đúng ý họ trước khi xúc tiến việc thiết kế chương trình. Nếu sau này khách đổi ý, đó là quyền của họ, nhưng họ phải trả tiền thay đổi (**variation**).

### 17.1.1 Cấu trúc các bộ phận

Chương trình nào cũng có một kiến trúc tương tự như một cỗ máy. Mỗi bộ phận càng đơn giản càng tốt và cách ráp các bộ phận phải như thế nào để ta dễ thử. Trong khi thiết kế ta phải biết trước những yếu điểm của mỗi bộ phận nằm ở đâu để ta chuẩn bị cách thử chúng. Ta sẽ không hề tin bộ phận nào hoàn hảo cho đến khi đã thử nó, dù nó đơn giản đến đâu.

Nếu ta muốn dùng một kỹ thuật gì trong một hoàn cảnh nào mà ta không biết chắc nó chạy không thì nên thử riêng rẽ nó trước. Phương pháp ấy được gọi là **Prototype**.

Ngoài ra, ta cũng nên xây dựng những kịch bản test cho những trường hợp đặc biệt, điển hình là bad data - khi người sử dụng bấm lung tung hay database chứa nhiều rác.

Nếu chương trình chạy trong **real-time** (tức là data thu nhập qua Serial Com Port, Data Acquisition Card hay mạng), chúng ta cần phải lưu ý những trường hợp khác nhau tùy theo việc gì xảy ra trước, việc gì xảy ra sau. Lúc bấy giờ Logic của chương trình sẽ tùy thuộc vào trạng thái (**State**) của data. Tốt nhất là nghĩ đến những **Scenarios** để có thể thử từng giai đoạn và tình huống.

Ngày nay với kỹ thuật hướng đối tượng, ở giai đoạn thiết kế này là lúc quyết định các Data Structures (tables, records ..v.v.) và con số Forms với Classes. Nhớ rằng mỗi Class gồm có một Data Structure và những Subs/Functions/Properties làm việc (operate) trên data ấy. Data structure phải chứa đầy đủ những chi tiết (data fields, variables) ta cần. Kế đó là những cách chương trình process data. Subs/Functions nào có thể cho bên ngoài gọi thì ta cho nó **Public**, còn những Subs/Functions khác hiện hữu để phục vụ bên trong class thì ta cho nó **Private**.

### 17.1.2 Kỹ thuật lập trình

Kiến thức cơ bản của lập trình viên và các thói quen của họ rất quan trọng. Nói chung, những người hấp tấp, nhảy vào viết chương trình trước khi suy nghĩ hay cân nhắc chín chắn thì sau này bugs xuất hiện nhiều là điều tự nhiên.

### 17.1.3 Dùng Subs và Functions

Nếu ở giai đoạn thiết kế kiến trúc của chương trình ta chia ra từng Class, thì khi lập trình ta lại thiết kế chi tiết về Subs, Functions ..v.v., mỗi thứ sẽ cần phải thử như thế nào. Nếu ta có thể chia công việc ra từng giai đoạn thì mỗi giai đoạn có thể mà một call đến một **Sub**. Thứ gì cần phải tính ra hay lấy từ nơi khác thì có thể được thực hiện bằng một **Function**.

Nhớ rằng điểm khác biệt chính giữa một Sub và một Function là Function cho ta một kết quả mà không làm thay đổi những **parameters** ta đưa cho nó. Trong khi đó, đầu rằng Sub không cho ta gì một cách rõ ràng nhưng nó có thể thay đổi trị số (value) của bất cứ parameters nào ta chuyển cho nó **ByRef**. Nhắc lại là khi ta chuyển một parameter **ByVal** cho một Sub thì giống như ta đưa một **copy** (bản sao) của variable đó cho Sub, Sub có thể sửa đổi nó nhưng nó sẽ bị bỏ qua, không ảnh hưởng gì đến **original** (bản chính) variable.

Ngược lại khi ta chuyển một parameter **ByRef** cho một Sub thì giống như ta đưa bản chính của variable cho Sub để nó có thể sửa đổi vậy.

Do đó để tránh trường hợp vô tình làm cho trị số một variable bị thay đổi vì ta dùng nó trong một Sub/Function chúng ta nên dùng ByVal khi chuyển nó như một parameter vào một Sub/Function.

Thật ra, chúng ta có thể dùng ByRef cho một parameter chuyển vào một Function. Trong trường hợp đó dĩ nhiên variable ấy có thể bị sửa đổi. Điều này gọi là phản ứng phụ (**side effect**), vì bình thường ít ai làm vậy. Do đó, nếu chúng ta thật sự muốn vượt ngoài qui ước thông thường thì nên Comment rõ ràng để cảnh báo người sẽ đọc chương trình chúng ta sau này.

Ngoài ra, mỗi lập trình viên thường có một **Source Code Library** của những Subs/Functions ưng ý. Chúng ta nên dùng các Subs/Functions trong Library của chúng ta càng nhiều càng tốt, vì chúng đã được thử nghiệm rồi.

## 17.2. Một số lưu ý

### 17.2.1 Đùng sợ Error

Mỗi khi chương trình có một Error, hoặc là **Compilation Error** (vì ta viết code không đúng văn phạm, ngữ vựng), hoặc là Error trong khi chạy chương trình, thì chúng ta không nên sợ nó. Hãy bình tĩnh đọc cái **Error Message** để xem nó muốn nói gì. Nếu không hiểu ngay thì đọc đi đọc lại vài lần và suy nghiệm xem có tìm được sự hướng dẫn nào không. Khi lập trình chúng ta sẽ gặp Errors rất nhiều, nên chúng ta phải tập bình tĩnh đối diện với chúng.

### 17.2.2 Dùng Comment (Chú thích)

Lúc viết code nhớ thêm Comment đầy đủ để bất cứ khi nào trở lại đọc đoạn code ấy trong tương lai chúng ta không cần phải dựa vào tài liệu nào khác mà có thể hiểu ngay lập tức mục đích của một Sub/Function hay đoạn code.

Như thế không nhất thiết chúng ta phải viết rất nhiều Comment nhưng hãy có điểm nào khác thường, bí hiểm thì chúng ta cần thông báo và giải thích tại sao chúng ta làm cách ấy. Có thể sau này ta khám phá ra đoạn code có bugs; lúc đọc lại có thể ta sẽ thấy đầu rằng ý định và thiết kế đúng nhưng cách lập trình có phần thiếu kiểm soát chẳng hạn.

Tính ra trung bình một lập trình viên chỉ làm việc 18 tháng ở mỗi chỗ. Tức là, gần như chắc chắn code chúng ta viết sẽ được người khác đọc và bảo trì ( debug và thêm bớt). Do đó, code phải càng đơn giản, dễ hiểu càng tốt. Đừng lo ngại là chương trình sẽ chạy chậm hay chiếm nhiều bộ nhớ, vì ngày nay computer chạy rất nhanh và bộ nhớ rất rẻ. Khi nào ta thật sự cần phải quan tâm về vận tốc và bộ nhớ thì điều đó cần được thiết kế cẩn thận chứ không phải dựa vào những tiêu xảo về lập trình.

### 17.2.3 Đặt tên các variables có ý nghĩa

Trong thực tế chúng ta gặp rất nhiều khó khăn khi làm việc với các variables có tên vắn tắt như K, L, AA, XY. Ta không có một chút ý niệm gì về chúng, mục đích sử dụng chúng làm gì. Thay vào đó, nếu ta đặt các tên variables như NumberOfItems, PricePerUnit, Discount .v.v.. thì sẽ dễ hiểu hơn.

Một trong những bugs khó thấy nhất là ta dùng cùng một tên cho **local variable** (variable declared trong Sub/Function) và **global variable** (variable declared trong Form hay Basic Module). Local variable sẽ che đậy global variable cùng tên, nên nếu chúng ta muốn nói đến global variable trong hoàn cảnh ấy chúng ta sẽ dùng làm local variable.

### 17.2.4 Dùng Option Explicit

Chúng ta nên trung thành với cách dùng **Option Explicit** ở đầu mỗi Form, Class hay Module. Nếu có variable nào đánh vắn sai VB6 IDE sẽ cho chúng ta biết ngay. Nếu chúng ta không dùng Option Explicit, một variable đánh vắn sai được xem như một variable mới với giá trị 0 hay "" (empty string).

Nói chung chúng ta nên thận trọng khi assign một data type cho một variable với data type khác. Chúng ta phải biết rõ chúng ta đang làm gì để khỏi bị phản ứng phụ (side effect).

### 17.2.5 Desk Check

Kiểm lại code trước khi compile. Khi ta compile code, nếu không có error chỉ có nghĩa là Syntax của code đúng, không có nghĩa là logic đúng. Do đó ta cần phải biết chắc là code ta viết sẽ làm đúng điều ta muốn bằng cách đọc lại code trước khi compile nó lần đầu tiên. Công việc này gọi là **Desk Check** (Kiểm trên bàn). Một chương trình được Desk Checked kỹ sẽ cần ít debug và chứa ít bugs không ngờ trước. Lý do là mọi scenarios đã được tiên liệu chu đáo.

### 17.2.6 Soạn một Test Plan

**Test Plan** liệt kê tất cả những gì ta muốn thử và cách thử chúng. Khi thử theo Test Plan ta sẽ khám phá ra những bug và tìm cách loại chúng ra. Hồ sơ ghi lại lịch sử của Test Plan (trục trặc gì xảy ra, chúng ta đã dùng biện pháp nào để giải quyết) sẽ bổ ích trên nhiều phương diện. Ta sẽ học được từ kinh nghiệm Debug và biết rõ những thứ gì trong dự án đã được thử theo cách nào.

## 17.3. Các kỹ thuật xử lý lỗi

### 17.3.1 Xử lý Error lúc Run time

Khi EXE của một chương trình viết bằng VB6 đang chạy, nếu gặp Error, nó sẽ hiển thị một Error Dialog cho biết lý do gây lỗi. Sau khi chúng ta click OK, chương trình sẽ ngưng. Nếu chúng ta chạy chương trình trong VB6 IDE, chúng ta có dịp bảo chương trình ngừng ở trong source code chỗ có Error bằng cách bấm button Debug trong Error Dialog. Tiếp theo đó chúng ta có thể tìm hiểu trị số các variables để đoán nguyên do của Error. Do đó, nếu chúng ta bắt đầu cho dùng một chương trình chúng ta viết cho nội bộ đơn vị, nếu tiện thì trong vài tuần đầu, thay gì chạy EXE của chương trình, chúng ta chạy source code trong VB6 IDE. Nếu có bug nào xảy ra, chúng ta có thể cho chương trình ngừng trong source code để debug.

Khi chúng ta dùng statement: *ON Error Resume Next*

Thì từ chỗ đó trở đi, nếu chương trình gặp Error, nó sẽ bỏ qua (ignore) hoàn toàn. Điểm này tiện ở chỗ giúp chương trình EXE của ta tránh bị treo ngay lập tức tại điểm xuất hiện bug. Nhưng nó cũng bất lợi là khi khách hàng cho hay họ gặp những trường hợp lạ, không giải thích được (vì Error bị ignored mà không ai để ý), thì ta cũng bí luôn, có thể không biết bắt

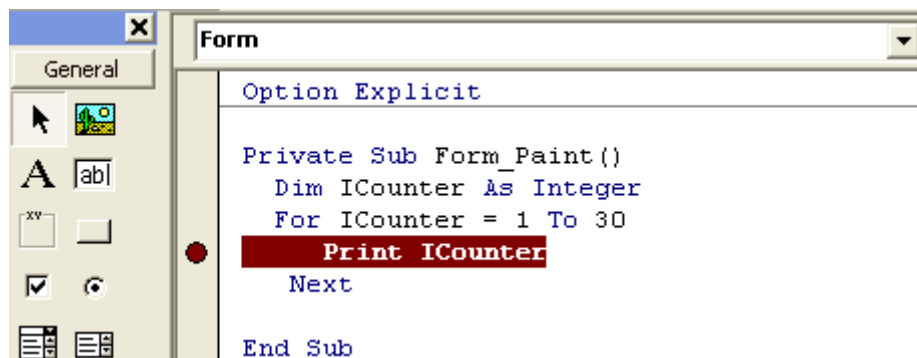


đầu từ đâu để debug. Do đó, dĩ nhiên trong lúc debug ta không nên dùng nó, nhưng trước khi giao cho khách hàng chúng ta nên cân nhắc kỹ trước khi dùng.

### 17.3.2 Dừng Breakpoints

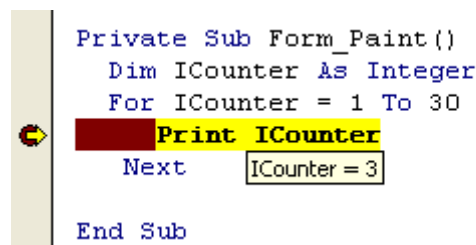
Cách hay nhất để theo dõi execution của chương trình là dùng Breakpoint để làm cho chương trình ngừng lại ở một chỗ ta muốn ở trong code, rồi sau đó ta cho chương trình bước từng bước. Trong dịp này ta sẽ xem xét trị số của những variables để coi chúng có đúng như dự định không.

Chúng ta đoán trước execution sẽ đi qua chỗ nào trong code, chọn một chỗ thích hợp rồi click bên trái của dòng code, chỗ dấu chấm tròn đỏ như trong hình dưới đây:



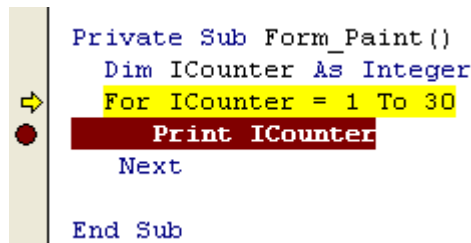
Nếu chúng ta click lên dấu chấm tròn đỏ một lần nữa thì là hủy bỏ nó. Một cách khác để đặt một breakpoint là để editor cursor lên dòng code rồi bấm **F9**. Nếu chúng ta bấm F9 lần nữa khi cursor nằm trên dòng đó thì là hủy bỏ break point.

Lúc chương trình đang dừng lại, chúng ta có thể xem trị số của một variable bằng cách để cursor lên trên variable ấy, tooltip sẽ hiện ra như trong hình dưới đây:



Có một số chuyện khác chúng ta có thể làm trong lúc này. Chúng ta có thể nắm dấu chấm tròn đỏ kéo (drag) nó ngược lên một hay nhiều dòng code để nó sẽ thực thi trở lại vài dòng code. Chúng ta cho chương trình thực thi từng dòng code bằng cách bấm **F8**. Menu command tương đương với nó là **Debug | Step Into**. Sẽ có lúc chúng ta không muốn chương trình bước

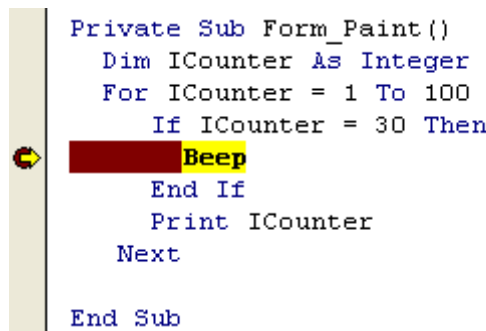
vào bên trong một Sub/Function mà muốn việc thực thi một Sub/Function như một bước đơn giản. Trong trường hợp đó, chúng ta dùng Menu command **Debug | Step Over** hay **Shift-F8**.



```
Private Sub Form_Paint()  
    Dim ICounter As Integer  
    For ICounter = 1 To 30  
        Print ICounter  
    Next  
End Sub
```

Nhớ là để cho chương trình chạy lại chúng ta bấm **F5**, tương đương với Menu command **Run | Continue**.

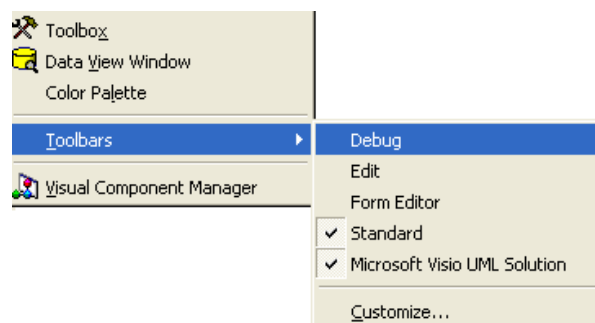
Có khi chúng ta muốn chương trình ngừng ở giữa một For Loop khi Iterator value có một trị số khá lớn. Nếu ta để sẵn một breakpoint ở đó rồi cứ bấm F5 nhiều lần thì hơi bất tiện. Có một phương pháp hữu hiệu là dùng một IF statement để thử khi Iterator value có trị số ấy thì ta ngừng ở breakpoint tại statement **Beep** (thay gì statement **Print ICounter**) như trong hình dưới đây:



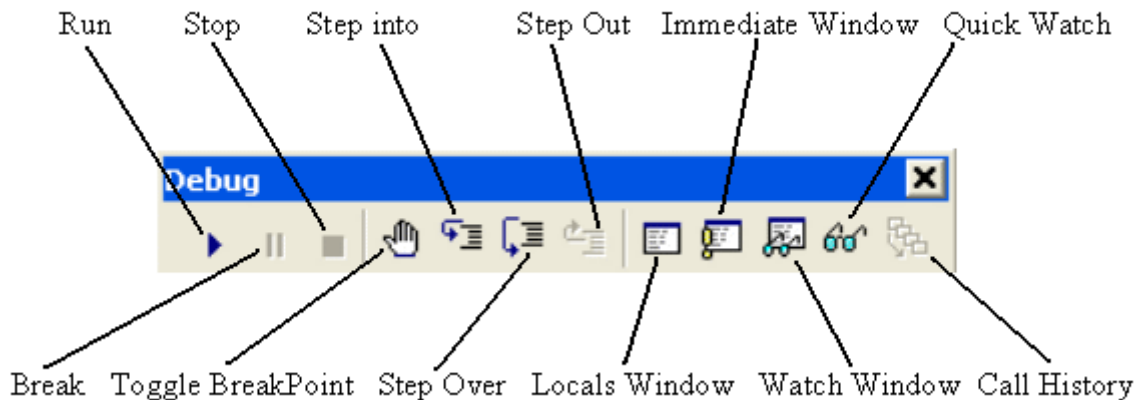
```
Private Sub Form_Paint()  
    Dim ICounter As Integer  
    For ICounter = 1 To 100  
        If ICounter = 30 Then  
            Beep  
        End If  
        Print ICounter  
    Next  
End Sub
```

Muốn hủy bỏ mọi breakpoints chúng ta dùng Menu command **Debug | Clear All Breakpoints**.

Để tiện việc debug, chúng ta có thể dùng **Debug Toolbar** bằng cách hiển thị nó với Menu command **View | Toolbars | Debug**



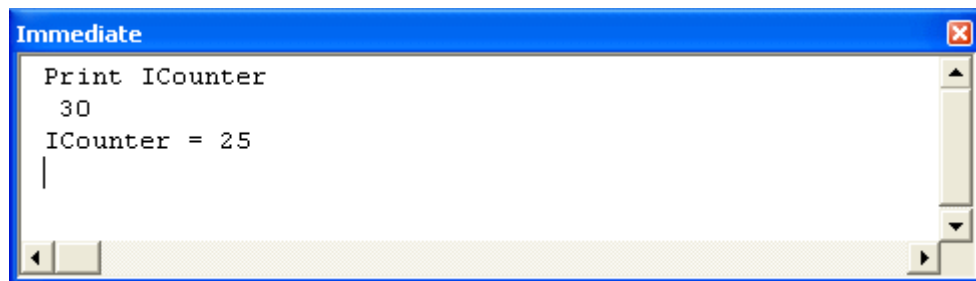
VB6 IDE sẽ hiển thị Debug Toolbar như sau:



### 17.3.3 Dừng Immediate Window

**Immediate Window** cho phép ta thực thi những VB statement trong khi chương trình đang dừng lại. Ta có thể dùng một Print statement để hiển thị trị số của một variable hay kết quả của một Function, gọi một Sub hay thay đổi trị số một variable trước khi tiếp tục cho chương trình chạy lại.

Để hiển thị Immediate Window, dùng Menu command **View | Immediate Window**.



Thay vì đánh "**Print ICounter**" chúng ta cũng có thể đánh "**? ICounter**". Nhớ là mỗi VB Statement chúng ta đánh trong Immediate Window sẽ được executed ngay khi chúng ta bấm **Enter**. Chúng ta có thể dừng lại bất cứ VB statement nào trong Immediate Window, chỉ cần bấm Enter ở cuối dòng ấy.

### 17.3.4 Theo dấu chân chương trình (Tracing)

Đôi khi không tiện để ngừng chương trình nhưng chúng ta vẫn muốn biết chương trình đang làm gì trong một Sub. Chúng ta có thể để giữa code của một Sub/Function một statement giống như dưới đây.

Debug.Print Format ( Now,"hh:mm:ss ") & "(Sub ProcessInput) Current Status:" & Status để chương trình hiển thị trong Immediate Window value của Status khi nó thực thi bên trong Sub ProcessInput lúc mấy giờ.

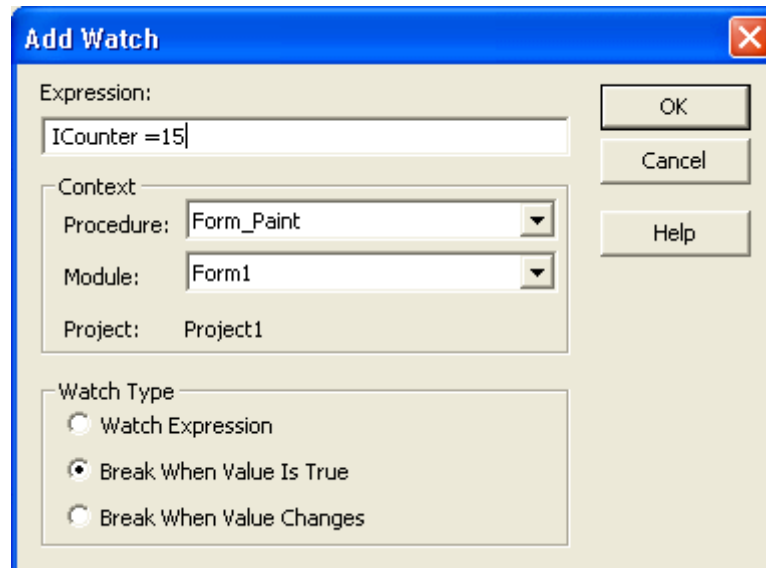
Có một cách khác là thay vì cho hiển thị trong Immediate Window chúng ta cho viết xuống (**Log**) vào trong một text file. Dưới đây là một Sub điển hình chúng ta có thể dùng để Log một Event message:

```
Sub LogEvent (ByVal GivenFileName, ByVal Msg As String, HasFolder
As Boolean, IncludeTimeDate As Integer)
    ' Append event message Msg to a text Logfile GivenFileName
    ' If GivenFileName is fullPathName then HasFolder is true
    ' IncludeTimeDate = 0 : No Time or Date
    ' = 1 : Prefix with Time
    ' = 2 : Prefix with Time and Date
    Dim FileNo, LogFileName, theFolder
    If HasFolder Then
        LogFileName = GivenFileName
    Else
        If Right (App.Path, 1) <> "\" Then
            theFolder = App.Path & "\"
        Else
            theFolder = App.Path
        End If
        LogFileName = theFolder & GivenFileName
    End If
    FileNo = FreeFile
    If Dir(LogFileName) <> "" Then
        Open LogFileName For Append As FileNo
    Else
        Open LogFileName For Output As FileNo
    End If
    Select Case IncludeTimeDate
    Case 0 ' No Time or Date
        Print #FileNo, Msg
    Case 1 ' Time only
        Print #FileNo, Format (Now, "hh:nn:ss ") & Msg
    Case 2 ' Date & Time
        Print #FileNo, Format (Now, "dd/mm/yyyy hh:nn:ss ") & Msg
```

```
End Select
Close FileNo
End Sub
```

### 17.3.5 Dừng Watch Window

Đôi khi chúng ta muốn chương trình ngừng không phải ở một chỗ nào nhất định, nhưng khi trị số của một variable hay của một expression là bao nhiêu, có thể là chúng ta không biết tại sao một variable tự nhiên có một trị số như vậy. Ví dụ chúng ta muốn chương trình ngừng lại khi **ICounter = 15**. Chúng ta có thể dùng Menu command **Debug | Add Watch**. VB6 IDE sẽ hiển thị dialog dưới đây. Chúng ta đánh **ICounter = 15** vào textbox **Expression** và click option box **Break When Value Is True** trong hộp **Watch Type**. Làm như vậy có nghĩa là ta muốn chương trình ngừng khi ICounter bằng 15.



### 17.3.6 Dừng phương pháp loại suy (Elimination Method)

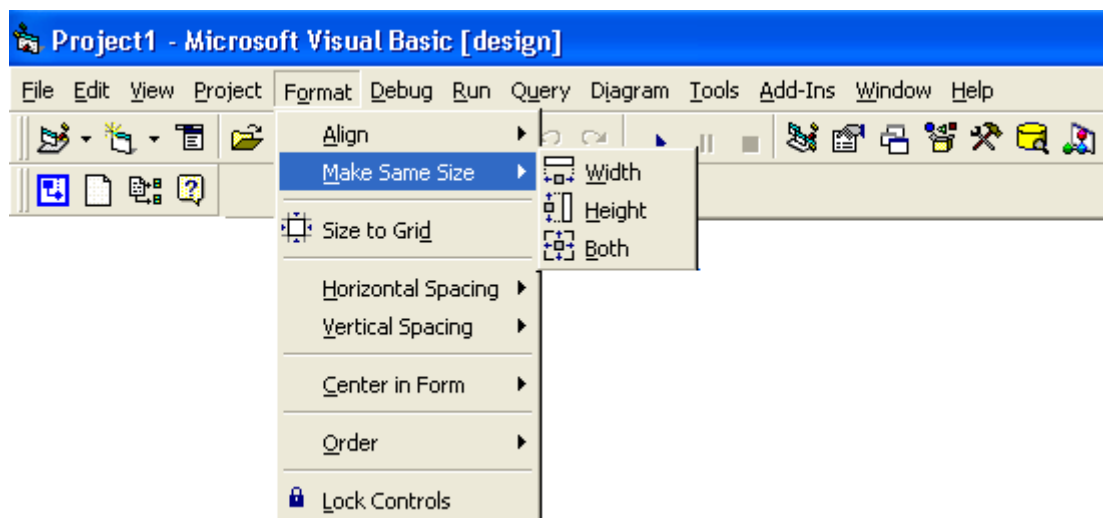
Có một phương pháp rất thông dụng khi debug là loại bỏ những dòng code nghi ngờ để xem bug có biến mất không. Nó được gọi là **Elimination Method**. Nếu bug biến mất thì những dòng code đã được loại bỏ là thủ phạm. Chúng ta có thể Comment Out một số dòng cùng một lúc bằng cách highlight các dòng ấy rồi click **Comment Block** trên Edit ToolBar.



## BÀI 18. DÙNG MENU

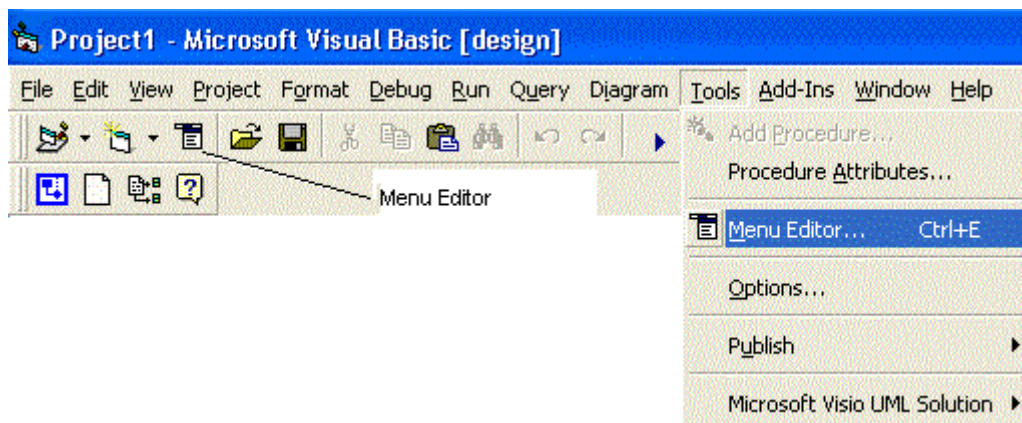
Menu trong Windows là nơi tất cả các commands của một chương trình được sắp xếp thứ tự theo từng loại để giúp ta dùng dễ dàng.

Có hai loại menu ta thường gặp : **drop-down (thả xuống)** menu và **pop-up (hiện lên)** menu. Ta dùng drop-down menu làm Menu chính cho chương trình. Thông thường nó nằm ở phía trên chóp màn hình. Nằm dọc theo chiều ngang là Menu Bar, nếu ta click lên một command trong Menu Bar thì chương trình sẽ thả xuống một menu với những MenuItem nằm dọc theo chiều thẳng đứng. Nếu ta click lên MenuItem nào có dấu hình tam giác nhỏ bên phải thì chương trình sẽ popup một Menu như trong hình dưới đây (khi ta click **Format | Make Same Size**):

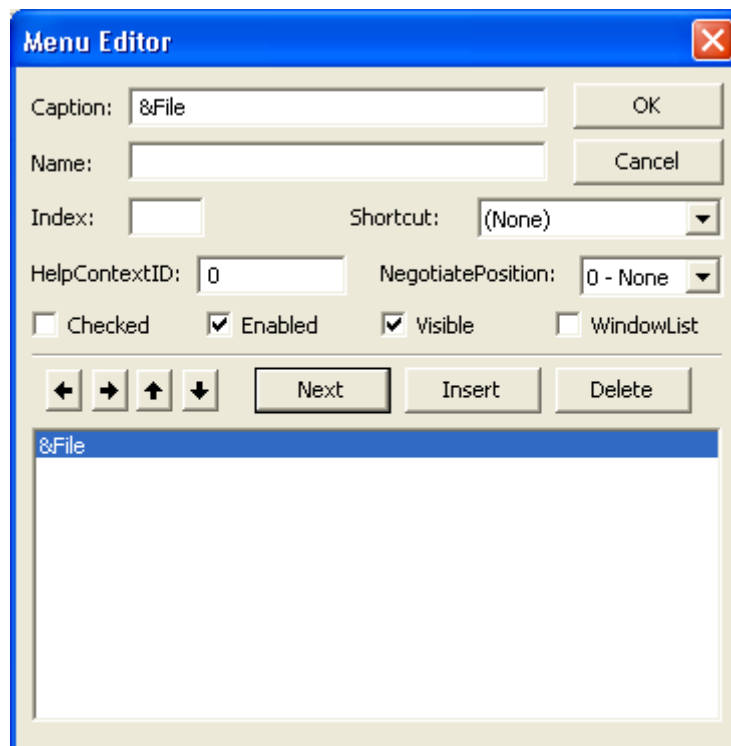


### 18.1. Main Menu

Ta dùng **Menu Editor** để tạo hoặc sửa một Menu cho chương trình. Menu thuộc về một Form. Do đó, trước hết ta select một Form để làm việc với Designer của nó (chớ không phải code của Form). Kế đó ta dùng Menu Command **Tools | Menu Editor** hay click lên icon của Menu Editor trên Toolbar để làm cho Menu Editor hiện ra.

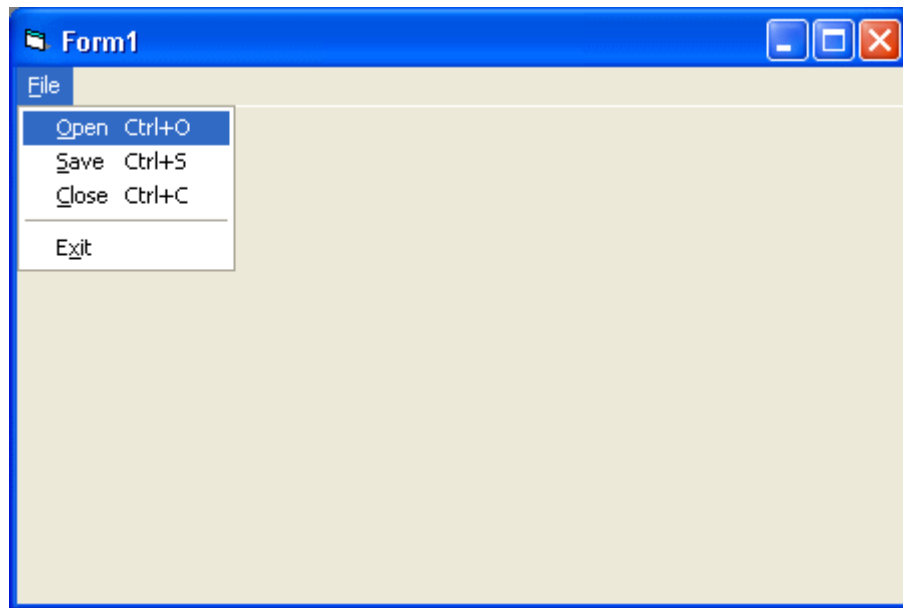


Đầu tiên có một vệt màu xanh nằm trong khung trắng của Menu Editor, nơi sẽ hiển thị Caption của Menu Command đầu tiên của Form. Khi ta đánh chữ **&File** vào Textbox **Caption**, nó cũng hiện ra trên vệt xanh nói trên. Kế đó, chúng ta có thể đánh tên của Menu Command vào Textbox **Name**. Dù ta cho Menu Command một tên nhưng ta ít khi dùng nó, trừ trường hợp muốn nó visible/invisible (hiện ra/biến mất). Bình thường ta dùng tên của MenuItems nhiều hơn.

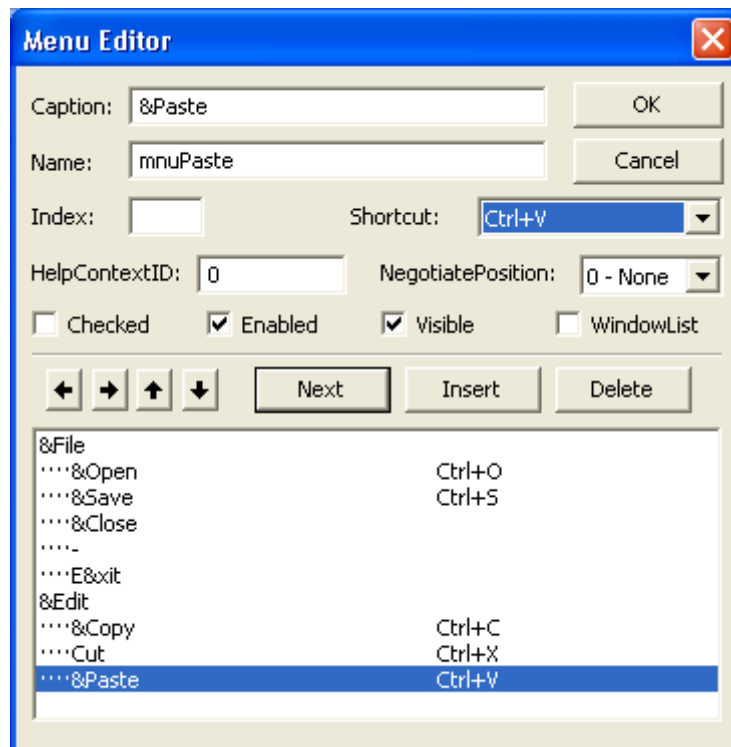


Để có một Menu như trong hình dưới đây ta còn phải edit thêm vào các MenuItems Open, Save, Close và Exit.





Hình dưới đây cho thấy tất cả các MenuItem của Menu Command File đều nằm thụt qua bên phải với bốn dấu chấm (...) ở phía trước. Khi ta click dấu tên chỉ qua phải thì MenuItem ta đang Edit sẽ có thêm bốn dấu chấm, tức là thụt một bậc trong Menu (Nested).



Tương tự như vậy, khi ta click dấu tên chỉ qua trái thì MenuItem ta đang Edit sẽ mất bốn dấu chấm, tức là trôi một bậc trong Menu.

Nếu muốn cho người sử dụng dùng Alt key để xử dụng Menu, chúng ta đánh thêm dấu **&** trước character chúng ta muốn trong menu Caption. Ví dụ **Alt-F** sẽ thả xuống Menu của Menu Command File.

Nếu chúng ta đặt cho MenuItem **&Open** tên **mnuOpen**, thì khi chúng ta Click lên Caption nó trên Form trong lúc thiết kế, VB6 IDE sẽ hiển thị cái vỏ của **Sub mnuOpen\_Click()**, giống như Sub cmdButton\_Click() của một CommandButton:

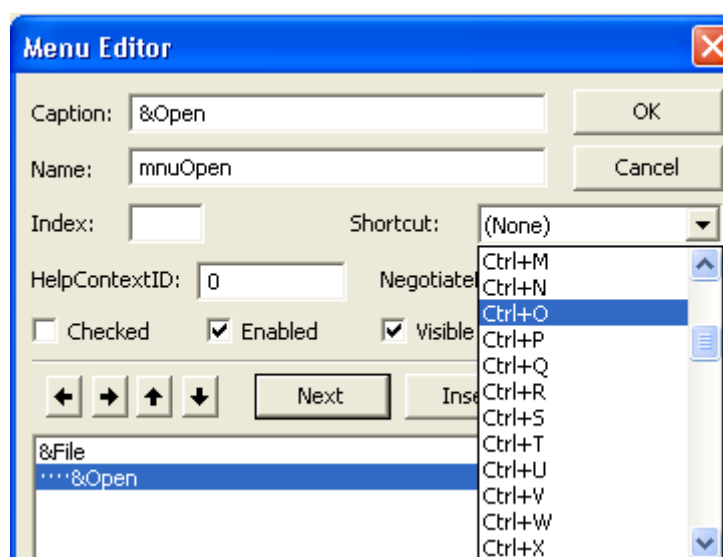
```
Private Sub mnuOpen_Click()  
    MsgBox "You clicked mnuOpen"  
End Sub
```

Trong ví dụ trên ta đánh thêm một Statement để hiển thị một message đơn giản **"You clicked mnuOpen"**. Chúng ta có thể đặt cho một MenuItem tên gì cũng được, nhưng người ta thường dùng prefix **mnu** để dễ phân biệt một menuItem Event với một CommandButton Event. Do đó, ta có những tên mnuFile, mnuOpen, mnuSave, mnuClose, mnuExit.

Cái gạch ngang giữa MenuItems Close và Exit được gọi là **Menu Separator**. Chúng ta có thể nhét một Menu Separator bằng cách cho Caption nó bằng **dấu trừ (-)**.

Ngoài Alt key ta còn có thể cho người sử dụng dùng Shortcut của menuItem. Để cho MenuItem một Shortcut, chúng ta chọn cho nó một Shortcut từ ComboBox **Shortcut** trong Menu Editor.

Trong hình dưới đây ta chọn **Ctrl+O** cho mnuOpen.



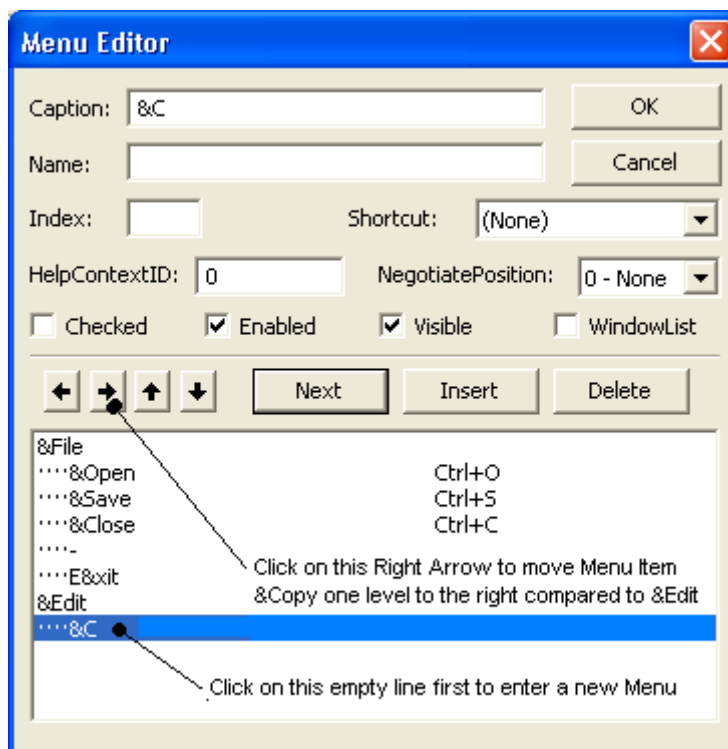
By default, menuItem được Enabled và Visible. Lúc thiết kế chúng ta có thể cho MenuItem giá trị khởi đầu của Enabled và Visible bằng cách dùng Checkboxes Enabled và Visible.

Trong khi chạy chương trình (at runtime), chúng ta cũng có thể thay đổi các values Enabled và Visible như sau:

```
mnuSave.Enabled = False  
mnuOpen.Visible = False
```

Khi một MenuItem có Enabled=False thì nó bị mờ và người sử dụng không dùng được.

Chúng ta dùng các dấu mũi tên chỉ lên và xuống để di chuyển MenuItem đã được selected lên và xuống trong danh sách các MenuItems. Chúng ta dùng button **Delete** để hủy bỏ MenuItem đã được selected, **Insert** để nhét một MenuItem mới ngay trên MenuItem đã được selected và **Next** để chọn MenuItem ngay dưới MenuItem đã được selected.

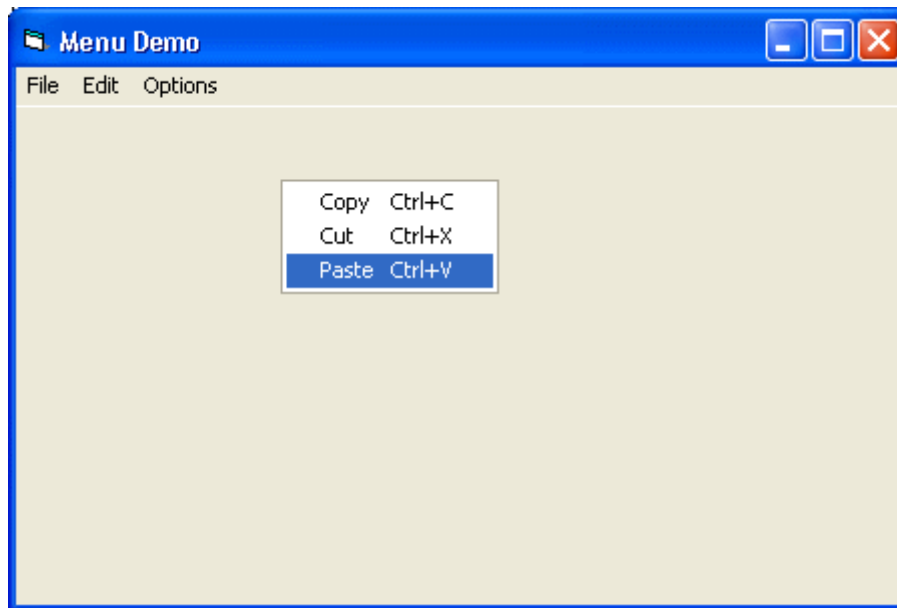


## 18.2. Pop-up Menu

Đối với người sử dụng, đang khi làm việc với một Object trong Windows tiện nhất là ta có thể làm hiển thị Context Menu (Menu áp dụng cho đúng tình huống) bằng một Mouse click. Thông thường đó là Right Click và cái Context Menu còn được gọi là Pop-up Menu. Chính cái

Pop-Up menu thật ra là Drop-down menu của một Menu Bar Command. Bình thường Menu Bar Command ấy có thể visible hay invisible (tàn hình).

Trong hình dưới đây, khi người sử dụng Right click trên Form, mnuEdit sẽ hiện lên. Nếu bình thường chúng ta không muốn cho người sử dụng dùng nó trong Main Menu thì chúng ta cho nó invisible:



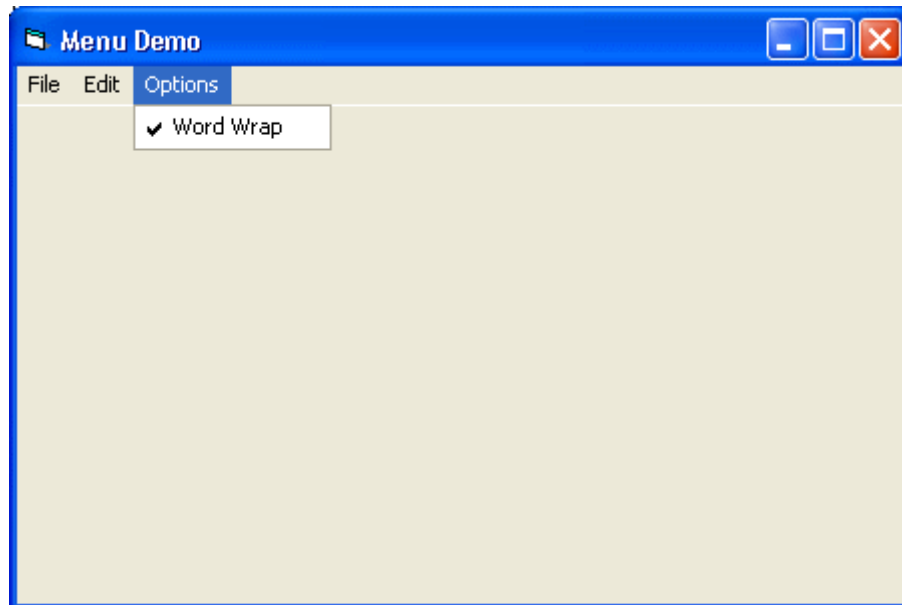
Code làm cho Popup menu hiện lên được viết trong Event Mousedown của một Object mà tính cờ ở đây là của chính cái Form:

```
Private Sub Form_MouseDown(Button As Integer, Shift As Integer, X
As Single, Y As Single)
    ' Popup the Edit Menu if User clicked the Right Button of the
Mouse
    If Button = vbRightButton Then
        PopupMenu mnuEdit
    End If
End Sub
```

Ngay cả khi chúng ta muốn cho mnuEdit bình thường là invisible, chúng ta cũng nên để cho nó visible trong lúc đầu để tiện bỏ code vào dùng để xử lý Click Events của những MenuItem thuộc về mnuEdit như mnuCopy, mnuCut và mnuPaste.

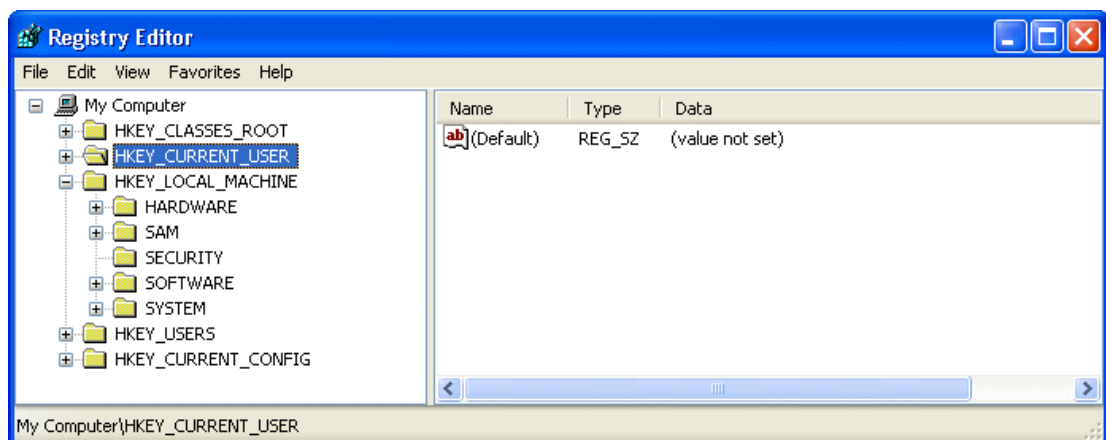
### 18.3. Chứa menu Settings trong Registry

Giả sử chương trình chúng ta cho người sử dụng một Option WordWrap như dưới đây:



Chúng ta muốn chương trình nhớ Option mà người sử dụng đã chọn, để lần tới khi người sử dụng khởi động chương trình thì Option WordWrap còn giữ nguyên giá trị như cũ.

Cách tiện nhất là chứa value của Option WordWrap như một **Key** trong **Registry**. Registry là một loại cơ sở dữ liệu đặc biệt của Windows Operating System dùng để chứa những dữ kiện liên hệ đến Users, Hardware, Configurations, ActiveX Components ..v.v. dùng trong computer. Trong Registry, data được sắp đặt theo từng loại theo đẳng cấp. Chúng ta có thể Edit trực tiếp trị số các Keys trong Registry bằng cách dùng **Registry Editor**.





Khi chứa value của một thứ gì (ta gọi là **Key**) vào Registry chúng ta có thể sắp đặt cho nó nằm trong **Section** nào tùy ý. Ở đây ta đặt ra hai Sections tên **Location** để chứa Top, Left của Form và tên **Settings** để chứa Key mnuWordWrap.Checked.

Muốn cho chương trình có các giá trị của Keys chứa trong Registry khi nó khởi động ta chỉ cần dùng **Function GetSetting** trong **Sub Form\_Load** để đọc vào từ Registry như dưới đây:

```
Private Sub Form_Load()  
    ' Initialise Location of the form by reading the Settings from  
the Registry  
    Me.Left = Val(GetSetting(App.Title, "Location", "Left", "0"))  
    Me.Top = Val(GetSetting(App.Title, "Location", "Top", "0"))  
    ' Initialise setting of WordWrap in the menu  
    mnuWordWrap.Checked = ( GetSetting(App.Title, "Settings",  
"WordWrap", "False") = "True" )  
End Sub
```

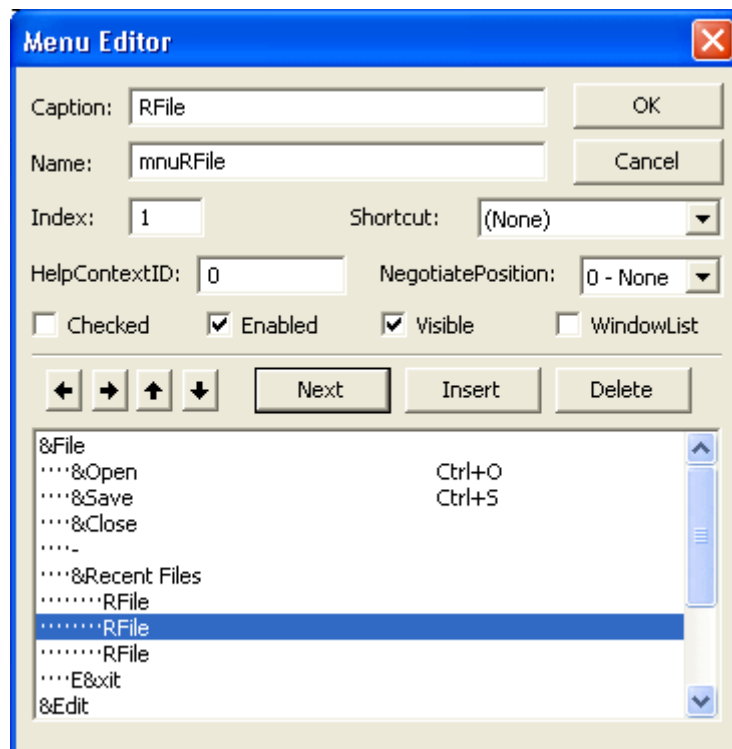
Lúc đầu khi chưa có gì trong Registry thì **"0"** (string "0" được converted bởi Val ra 0) là default value cho Left và Top, còn **"False"** là default value của mnuWordWrap.Checked.

Ngoài ra ta cũng muốn chương trình nhớ tên của ba Files User dùng gần đây nhất. Tức là trong Drop-down của Menu Command File sẽ có MenuItem **Recent Files** để hiển thị từ một đến ba tên Files, cái mới nhất nằm trên hết. Trước hết, ta cần tạo ra 3 SubmenuItem có cùng tên mnuRFile nhưng mang **Index** bằng 0,1 và 2 (chúng ta đánh vào Textbox **Index**). Ta sẽ dùng Captions của chúng để hiển thị tên các Files. Lúc chưa có Filename nào cả thì MenuItem **Recent Files** sẽ bị làm mờ đi (tức là mnuRecentFiles.Enabled = False ).

Ta sẽ chứa tên các Files như một String trong Section Settings của Registry. Ta phân cách tên các Files bằng delimiter character |. Ví dụ:

```
"LattestFileName.txt|OldFileName.txt|OldestFilename.txt"
```

Mỗi lần người sử dụng Open một File ta sẽ thêm tên File ấy vào trong Registry và bất cứ lúc nào chỉ giữ lại tên của 3 Files mới dùng nhất.



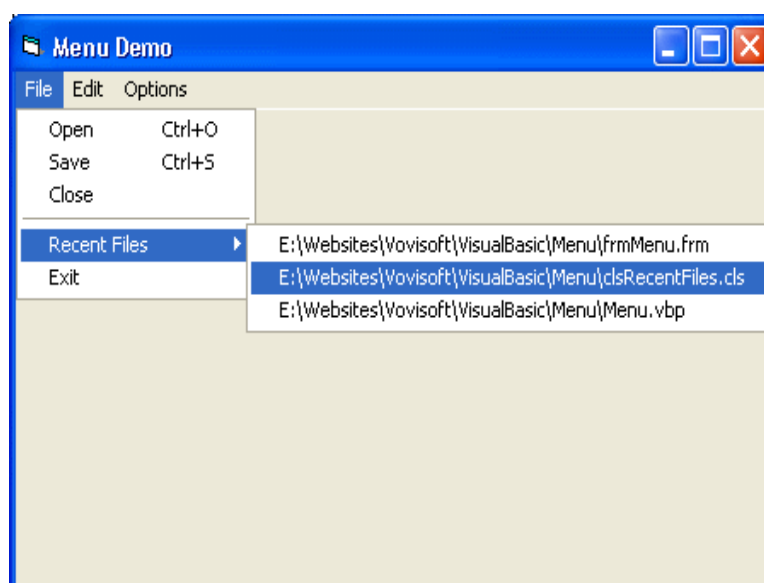
Dưới đây là code dùng để thêm tên File mới dùng nhất vào Registry:

```
Private Sub mnuOpen_Click()  
    ' Initialise Folder in Common Dialog  
    CommonDialog1.InitDir = App.Path  
    ' Launch the dialog  
    CommonDialog1.ShowOpen  
    ' Save the Filename in the Registry, using Object  
myRecentFiles  
    myRecentFiles.AddFile CommonDialog1.FileName  
End Sub
```

Code dùng trong Sub Form\_Load để đọc tên RecentFiles và hiển thị trong Menu:

```
'  
Set myRecentFiles = New clsRecentFiles  
    ' Pass the form handle to it  
    ' This effectively loads the most recently used FileNames to  
menu  
myRecentFiles.Init Me
```





Ta sẽ dùng một Class tên **clsRecentFiles** để đặc biệt lo việc chứa tên Files vào Registry và hiển thị tên các Files ấy trong Menu. Bên trong clsRecentFiles ta cũng dùng [clsString](#), là một Class giúp ta ngắt khúc String trong Registry ra tên của các Files dựa vào chỗ các delimiter character |.

```
' Class Name: clsRecentFiles
' This Class saves the most Recent FileNames used in the Registry
' in form of a String delimited by |.
' Up to MaxFiles Filenames maybe stored.
' You need to pass the Form that contains the menu to it.
' The assumption is that you have created an array of MenuItem
' named mnuRFile to display the FileNames
'
Const MaxFiles = 3 ' Maximum number of FileNames to remember
Private myForm As Form
Private RecentFiles As clsString
Public Sub Init(TForm As frmMenu)
    Set myForm = TForm
    Set RecentFiles = New clsString
    ' Read the Most Recent Filename String from the Registry
    RecentFiles.Text = GetSetting(App.Title, "Settings",
"RecentFiles", "")
    ' Assign the Delimiter character and tokenise the String
    (i.e. split it) into FileNames
    RecentFiles.Delimiter = "|"
```

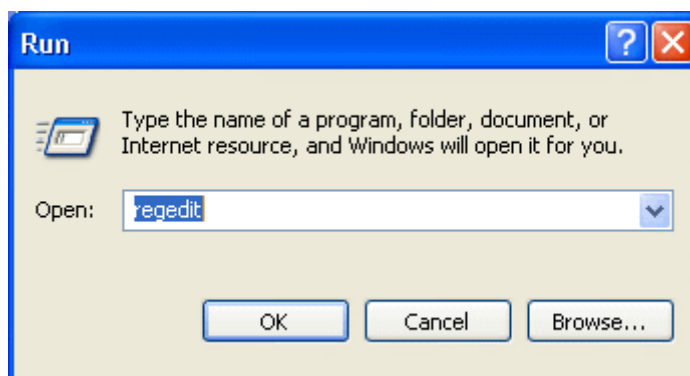
```
UpdateMenu
End Sub

Public Sub AddFile(FileName As String)
    ' Add the latest FileName to the list and update the Registry
    ' Prefix the FileName to the existing MostRecentFileName
String
    RecentFiles.Text = FileName & "|" & RecentFiles.Text
    ' Discard the oldest FileNames if the total number is greater
than MaxFiles
    If RecentFiles.TokenCount > MaxFiles Then
        Dim TStr As String
        Dim i As Integer
        ' Reconstitute the String that contains only the most
recent MaxFiles FileNames
        For i = 1 To MaxFiles
            TStr = TStr & RecentFiles.TokenAt(i) & "|"
        Next
        ' Remove the last delimiter character on the right
        RecentFiles.Text = Left(TStr, Len(TStr) - 1)
    End If
    ' Update the String in the Registry
    SaveSetting App.Title, "Settings", "RecentFiles",
RecentFiles.Text
    UpdateMenu
End Sub

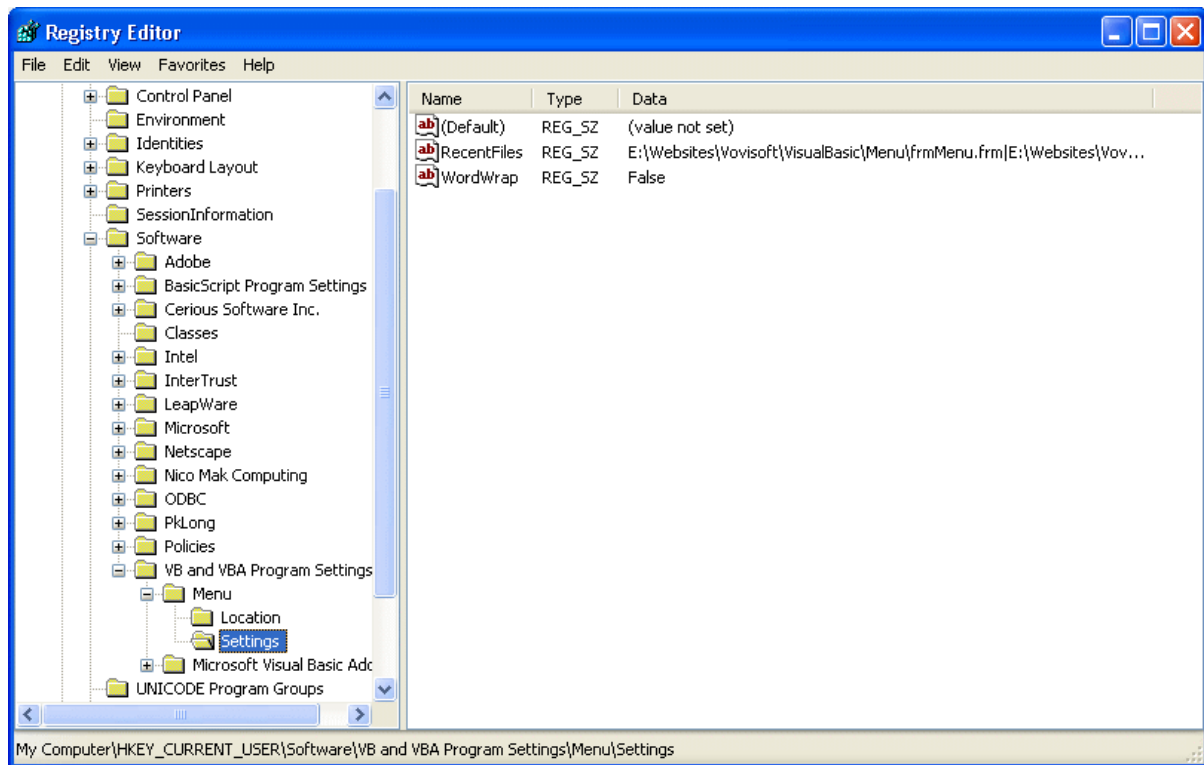
Private Sub UpdateMenu()
    ' Hiện thị the most recent Filenames in the menu
    Dim i As Integer
    ' If there is no FileNames to hiển thị then disable the
MenuItem entry
    If RecentFiles.TokenCount = 0 Then
        myForm.mnuRecentFiles.Enabled = False
        Exit Sub
    Else
        ' Otherwise enable the MenuItem entry
        myForm.mnuRecentFiles.Enabled = True
    End If
End Sub
```

```
End If
    ' Assign FileName to Caption of mnuRFile array and make the
MenuItem elements visible
    For i = 1 To RecentFiles.TokenCount
        myForm.mnuRFile(i - 1).Caption = RecentFiles.TokenAt(i) '
Assign to Caption
        myForm.mnuRFile(i - 1).Visible = True ' Make the MenuItem
visible
        If i = MaxFiles Then Exit For ' This line maybe
unnecessary
    Next
    ' Make the rest of the MenuItem array mnuRFile invisible if
there are less than MaxFiles
    If RecentFiles.TokenCount < MaxFiles Then
        For i = RecentFiles.TokenCount To MaxFiles - 1
            myForm.mnuRFile(i).Visible = False
        Next
    End If
End Sub
```

Chúng ta có thể chạy Line Command **RegEdit** sau khi click **Start | Run**



để xem chi tiết của các Keys mà chương trình đã chứa trong Sections **Location** và **Settings** của Folder **HKEY\_CURRENT\_USER\Software\VB and VBA Program Settings\Menu**



## BÀI 19. DÙNG DIALOGS

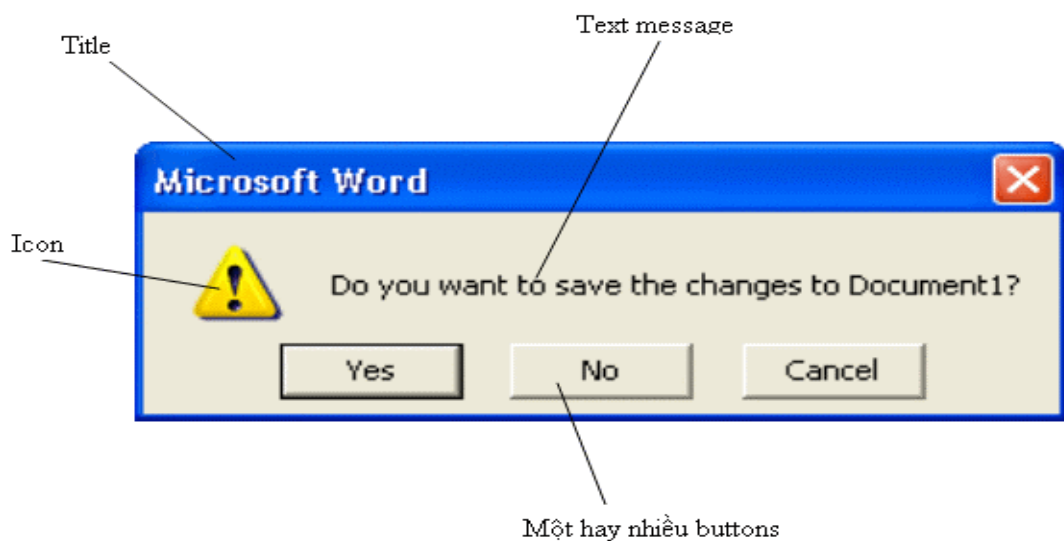
Dialogs (hội thoại) được dùng để hiển thị tin tức và nhận thông tin từ chuột hay bàn phím từ người sử dụng tùy theo tình huống. Chúng được dùng để tập trung sự chú ý của người sử dụng vào công việc hiện tại của chương trình nên rất hữu dụng trong các chương trình của Windows

Có nhiều dạng Dialogs, mỗi thứ áp dụng cho một hoàn cảnh riêng biệt. Trong chương này ta sẽ bàn qua 4 loại Dialogs chính và nghiên cứu về khi nào và cách nào ta dùng chúng:

- Message Boxes
- Input Boxes
- Common Dialogs
- Custom Dialogs

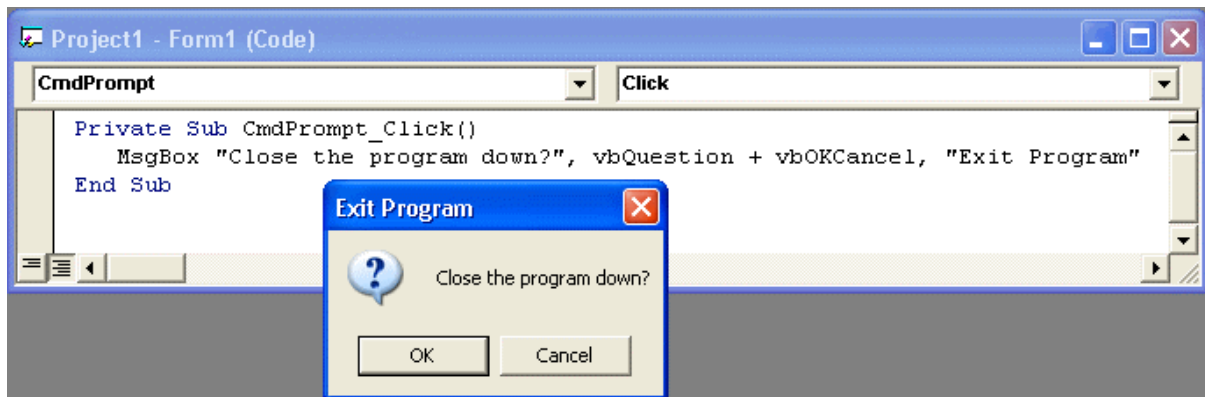
### 19.1. Message Boxes

**Message Boxes** được dùng để nhắc nhở người sử dụng một chuyện gì, và đòi hỏi một phản ứng nào đó từ người sử dụng. Ví dụ như khi ta chấm dứt chương trình MSWord mà chưa lưu trữ hồ sơ thì MSWord sẽ nhắc ta lưu trữ nó bằng Dialog dưới đây:



Trong trường hợp này người sử dụng có thể click một trong 3 buttons. Nếu click **Yes** thì sẽ xúc tiến việc lưu trữ hồ sơ trước khi kết thúc chương trình MSWord. Nếu click **No** thì MSWord sẽ lặng lẽ kết thúc. Nếu click **Cancel** thì có nghĩa người sử dụng đổi ý việc chấm dứt chương trình và trở lại tiếp tục dùng MSWord.

Ta dùng routine **MsgBox** để hiển thị Message Box như coding trong hình dưới đây:



Parameter (thông số) thứ nhất của MsgBox là text message **Close the program down?**, parameter thứ nhì là tập hợp của icon (vbQuestion) và số buttons (vbOKCancel) bằng cách cộng hai constants: **vbQuestion + vbOKCancel** (hai buttons OK và Cancel), parameter thứ ba là title (tiêu đề) của Dialog.

Trong ví dụ MSWord bên trên Constant của icon và buttons là **vbExclamation + vbYesNoCancel** (ba buttons Yes, No và Cancel).

Ta chọn số và loại buttons theo bảng dưới đây:

Constant	Các buttons
vbOKOnly	OK
vbOKCancel	OK Cancel
vbYesNo	Yes No
vbRetryCancel	Retry Cancel
vbYesNoCancel	Yes No Cancel
vbAbortRetryIgnore	Abort Retry Ignore

Constant của các icons ta có thể dùng là **vbCritical**, **vbQuestion**, **vbExclamation** và **vbInformation**.

Khi một Message Box được mở ra, cả chương trình ngừng lại và đợi người sử dụng phản ứng. Ta nói Message Box được hiển thị trong **Modal Mode**, nó dành mọi sự chú ý và tạm ngưng các execution khác trong cùng chương trình. Sau khi người sử dụng click một button, Message Box sẽ biến mất và chương trình sẽ tiếp tục chạy từ dòng code ngay dưới dòng MsgBox.

Trong ví dụ trên ta dùng MsgBox như một Sub, nhưng ta cũng có thể dùng MsgBox như một Function để biết người sử dụng vừa mới click button nào. Function MsgBox returns một value (trả về một giá trị) mà ta có thể thử để theo đó thì hành. Ví dụ như:

```
Private Sub CmdPrompt_Click()  
    Dim ReturnValue As Integer  
    ReturnValue = MsgBox("Close the program down", vbQuestion +  
vbOKCancel, "Exit Program")  
    Select Case ReturnValue  
    Case vbOK  
        MsgBox "You clicked OK"  
    Case vbCancel  
        MsgBox "You clicked Cancel"  
    End Select  
End Sub
```

Các trị số Visual Basic intrinsic constants mà Function MsgBox returns là:

Trị số	Tên	Const
1	OK	vbOK
2	Cancel	vbCancel
3	Abort	vbAbort
4	Retry	vbRetry
5	Ignore	vbIgnore
6	Yes	vbYes
7	No	vbNo

Chúng ta có thể hiển thị Text message trong Message Box thành nhiều dòng bằng cách dùng Constant **vbCrLf** (CarriageReturn và LineFeed) để đánh dấu những chỗ ngắt khúc như sau:

```
MsgBox "This is the first line" & vbCrLf & " followed by the  
second line"
```

Nếu chúng ta thấy mình thường dùng MsgBox với cùng một icon và những buttons, nhưng có Text message khác nhau, chúng ta có thể viết một Global Subroutine trong .BAS module để dùng lại nhiều lần. Ví dụ chúng ta có một Global Sub như sau:

```
Public Sub DisplayError(ByVal ErrMess As String )
    MsgBox ErrMess, vbCritical + vbOKOnly, "Error"
End Sub
```

Mỗi lần muốn hiển thị một Error message chúng ta chỉ cần gọi Sub DisplayError với Text message mà không sợ dùng lầm lẫn icon. Sau này muốn đổi cách hiển thị Error message chỉ cần edit ở một chỗ. Nếu người sử dụng muốn chúng ta lưu trữ tất cả mọi errors xảy ra lúc run-time, chúng ta chỉ cần thêm vài dòng code trong Sub Hiển thịError để viết Error message vào một text file.

## 19.2. Input Boxes

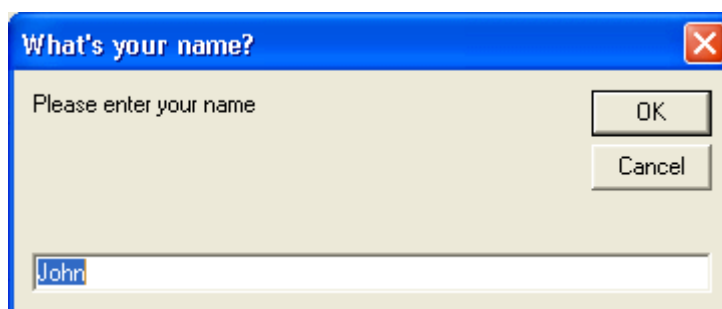
Với Message Boxes, người sử dụng chỉ có thể click lên một button. Đôi khi ta muốn người sử dụng đánh vào thêm một ít dữ kiện, trong trường hợp ấy ta có thể dùng **Input Boxes**.

Input Boxes giống giống Message Box, nhưng nó chuyên nhận input data từ người sử dụng và không hiển thị một icon. Ví dụ:

```
Private Sub CmdGreeting_Click()
    Dim strReply As String
    strReply = InputBox$("Please enter your name", "What 's your
name?", "John", 2000, 1000)
    MsgBox "Hi " & strReply & ", it 's great to meet you!",
vbOKOnly, "Hello"
End Sub
```

Để ý các parameters của **Function InputBox\$**. Parameter thứ nhất là Text message, parameter thứ hai là Title của Dialog, parameter thứ ba là Default Input Value. Đây là value được hiển thị sẵn trong Input Box khi nó xuất hiện, nếu đó là input user thường đánh vào thì người sử dụng chỉ cần click nút **OK** là đủ. Hai parameters cuối cùng là Optional (tùy chọn, có cũng được, không có cũng không sao). Nó là X,Y coordinates của Input Box trong đơn vị **twips**. Hệ thống tọa độ lấy góc trên bên trái làm chuẩn với X=0, Y=0.





Input Box có hai dạng Functions:

- **InputBox\$** - returns một String đang hoàng
- **InputBox** - returns một String nằm trong Variant variable

Nếu chúng ta click nút Cancel thì returned Value là empty string, chúng ta có thể test empty string để nhận diện trường hợp này.

Dưới đây là một ví dụ dùng Function InputBox:

```
Private Sub CmdFortuneTeller_Click()  
    Dim varValue As Variant  
    Dim intAge As Integer  
    varValue = InputBox("Please enter your age", "How old are  
you?", "18")  
    If IsNumeric(varValue) Then  
        intAge = Val(varValue)  
        If intAge < 20 Then  
            MsgBox "You are a young and ambitious person", vbOKOnly,  
"Observation"  
        Else  
            MsgBox "You are a matured and wise person", vbOKOnly,  
"Observation"  
        End If  
    Else  
        MsgBox "Oh oh! - please type your age!", vbCritical +  
vbOKOnly, "Input Error"  
    End If  
End Sub
```

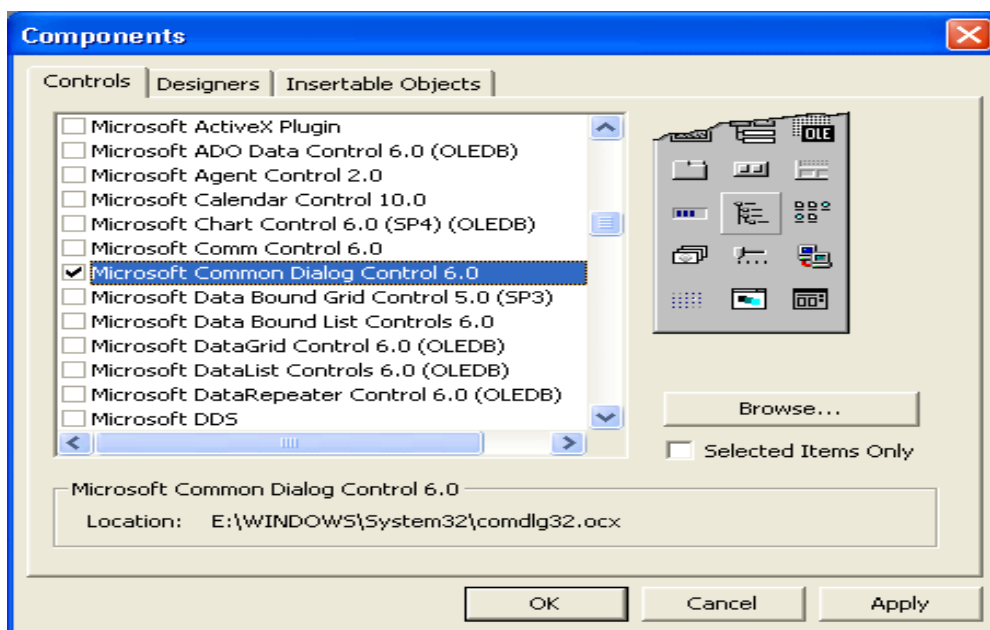
Mặc dầu Input Boxes rất dễ dùng, trên thực tế rất ít khi ta dùng nó vì những lý do sau đây:

- Ta không thể làm gì được trong lúc người sử dụng input data, phải đợi sau khi người sử dụng click OK thì mới bắt đầu xử lý input textstring. Ngược lại nếu ta dùng một Textbox trong một Form thông thường, ta có thể code trong các Event handlers của Events **KeyPress** hay **Change** để kiểm soát các keystrokes của người sử dụng.
- Input Boxes chỉ cho ta đánh vào một text string duy nhất. Nhiều khi ta muốn người sử dụng đánh vào nhiều thứ nên cần phải có một form riêng.
- Sau cùng, Input Boxes xem không đẹp mắt. Chương trình dùng Input Boxes có vẻ như không chuyên nghiệp, do đó ta cần phải dùng **Custom Dialogs**.

### 19.3. Common Dialogs

Chúng ta có thể ý thấy hầu như mọi chương trình trong Windows đều có cùng những dialogs để Open và Save files ? Và hầu như tất cả chương trình đều có cùng dialogs để chọn màu, font chữ hay để in ? Đó là vì các Dialogs thông dụng ấy thuộc về Common Dialog Library của MSWindows và cho phép các chương trình gọi.

Muốn dùng các Dialogs ấy trong VB6 ta phải reference **Comdlg32.ocx** bằng IDE Menu command **Project | Components...** rồi chọn và Apply **Microsoft Common Dialog Control 6.0**.



Microsoft Common Dialog Control 6.0 cho ta sáu dạng Dialogs tùy theo gọi Method nào:

Tên	Method
Open File	ShowOpen
Save File	ShowSave
Color	ShowColor
Font	ShowFont
Print	ShowPrinter
Help	ShowHelp

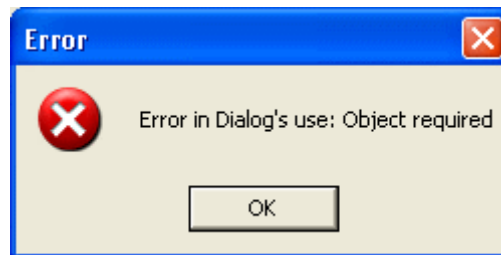
## 19.4. Open và Save File Dialogs

Chúng ta hãy mở một Project mới với một button tên CmdOpen trong Form1 và đánh vào code sau đây cho Sub CmdOpen\_Click:

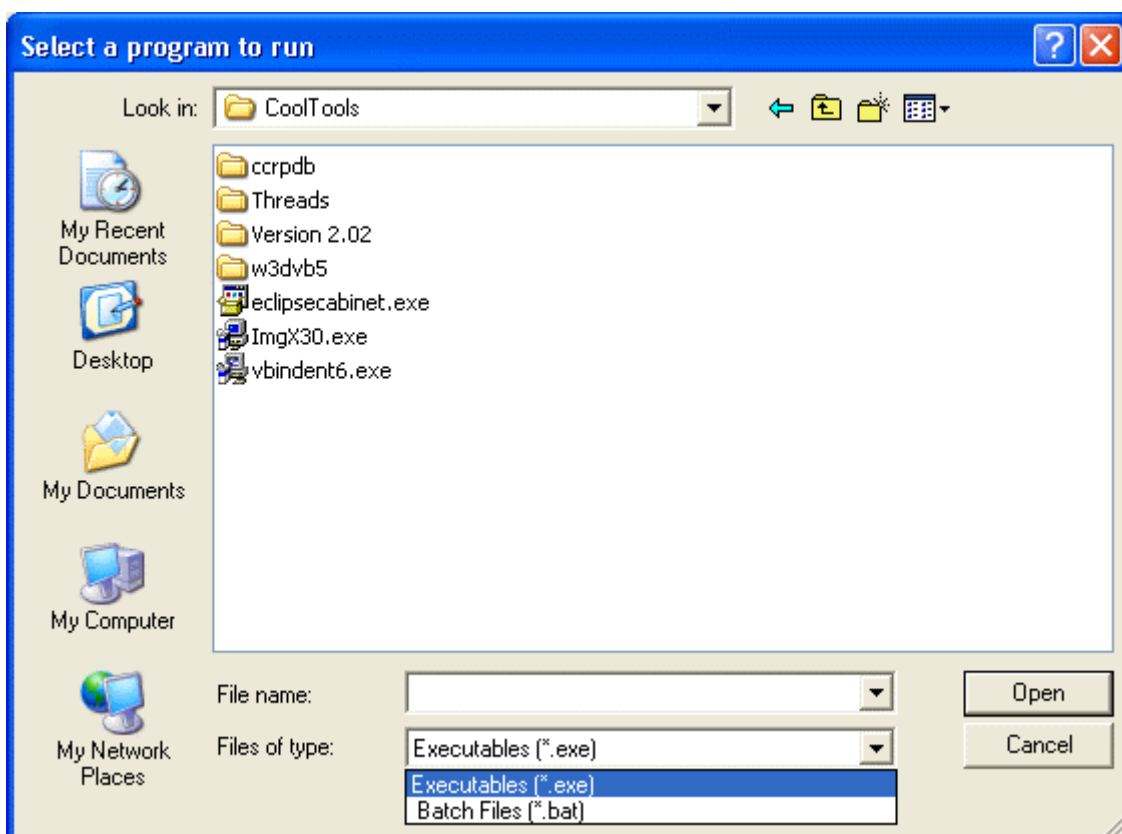
```
Private Sub CmdOpen_Click()  
    On Error GoTo DialogError  
    With CommonDialog1  
        .CancelError = True ' Generate Error number cdlCancel if  
user click Cancel  
        .InitDir = "E:\VB6" ' Initial (i.e. default ) Folder  
        .Filter = "Executables (*.exe) | *.exe| Batch Files  
(* .bat) | *.bat"  
        .FilterIndex = 1 ' Select "Executables (*.exe) | *.exe"  
as default  
        .DialogTitle = "Select a program to run"  
        .ShowOpen ' Launch the Open Dialog  
        MsgBox "You selected " & .FileName, vbOKOnly +  
vbInformation, "Open Dialog"  
    End With  
    Exit Sub  
DialogError:  
    If Err.Number = cdlCancel Then  
        MsgBox "You clicked Cancel!", vbOKOnly + vbInformation,  
"Open Dialog"  
        Exit Sub  
    Else  
        MsgBox "Error in Dialog's use: " & Err.Description,  
vbOKOnly + vbCritical, "Error"
```

```
Exit Sub
End If
End Sub
```

Hãy chạy chương trình ấy và click button **Open**, chương trình sẽ hiển thị error message dưới đây:



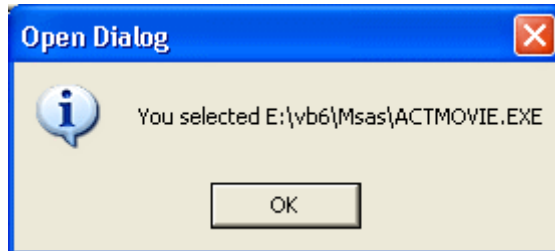
Lý do là ta quên bỏ một Microsoft Common Dialog Control 6.0 vào Form1. Vậy chúng ta hãy doubleclick icon của nó trong ToolBox. Bây giờ hãy chạy chương trình lại và click button Open để hiển thị **Open Dialog**.



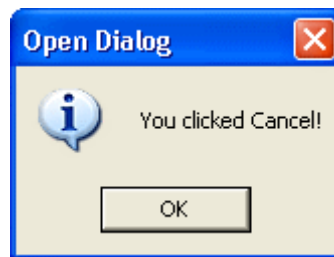
Chúng ta có thể chọn folder nào tùy ý bằng cách di chuyển từ folder này qua folder khác hay thay đổi disk drive. Nếu chúng ta click vào bên phải của combobox **File of type**, nó sẽ dropdown để cho thấy chúng ta có thể chọn một trong hai loại Files như liệt kê trong statement:

```
.Filter = "Executables (*.exe) | *.exe| Batch Files (*.bat) | *.bat"
```

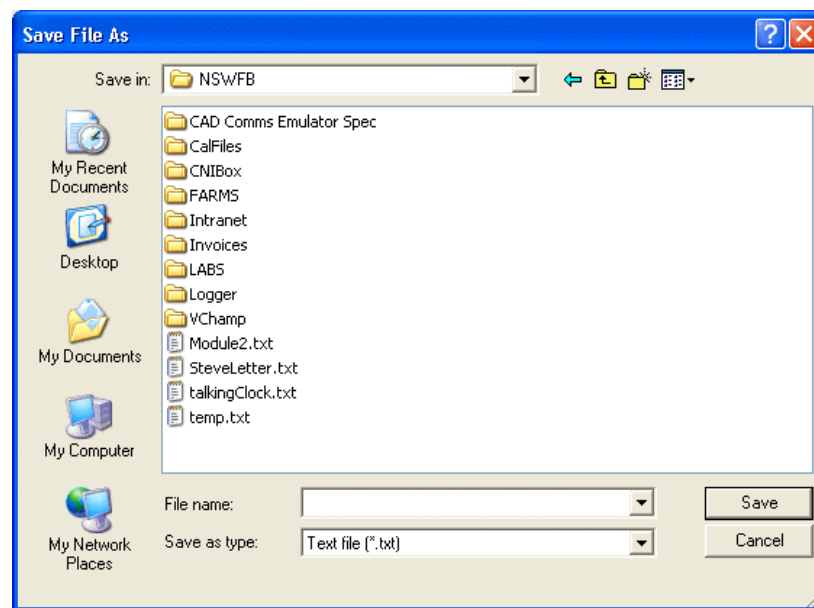
Sau khi chọn một Filename có sẵn hay đánh một tên vào **File name** textbox, chúng ta click **Open**. Sau đó, `CommonDialog1.FileName` sẽ chứa tên file chúng ta đã chọn hay đánh vào.



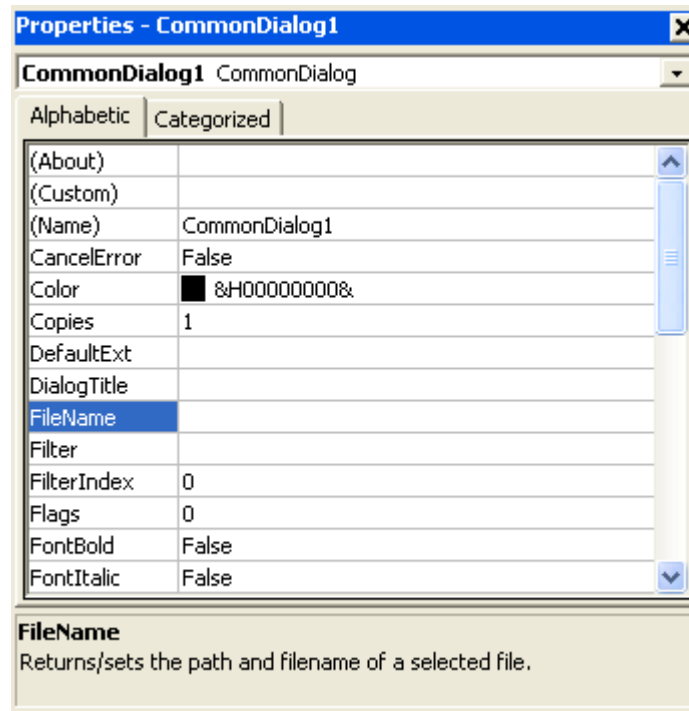
Vì ta cho `.CancelError = True` nên nếu người sử dụng click Cancel chương trình sẽ generate một Error số **32755 (cdlCancel)**. Ở đây ta bắt Error ấy bằng cách dùng **On Error GoTo DialogError** và thử `Err.Number= cdlCancel` để hiển thị Error message dưới đây:



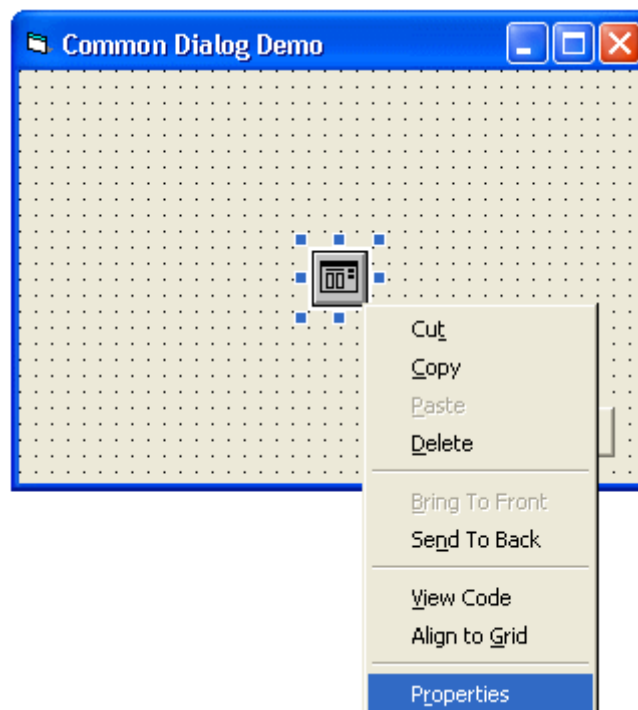
**Save Dialog** cũng tương tự như **Open Dialog**, ta dùng method **ShowSave** để hiển thị nó.



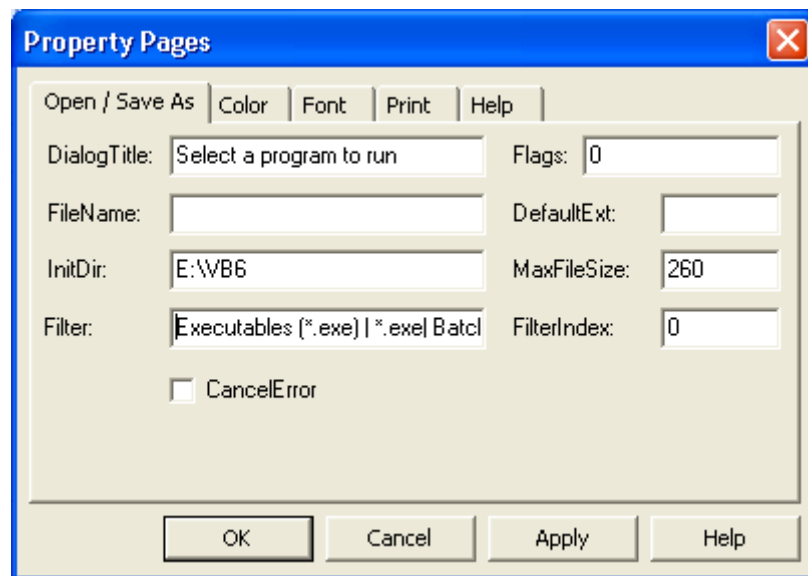
Trong ví dụ trên ta định nghĩa các properties của `CommonDialog1` bằng code. Chúng ta cũng có thể dùng Properties Windows để định nghĩa chúng như dưới đây:



Ngoài ra, chúng ta cũng có thể dùng các trang Properties của CommonDialog1 để định nghĩa Properties lúc thiết kế bằng cách right click CommonDialog1 trên Form1 rồi chọn **Properties**:



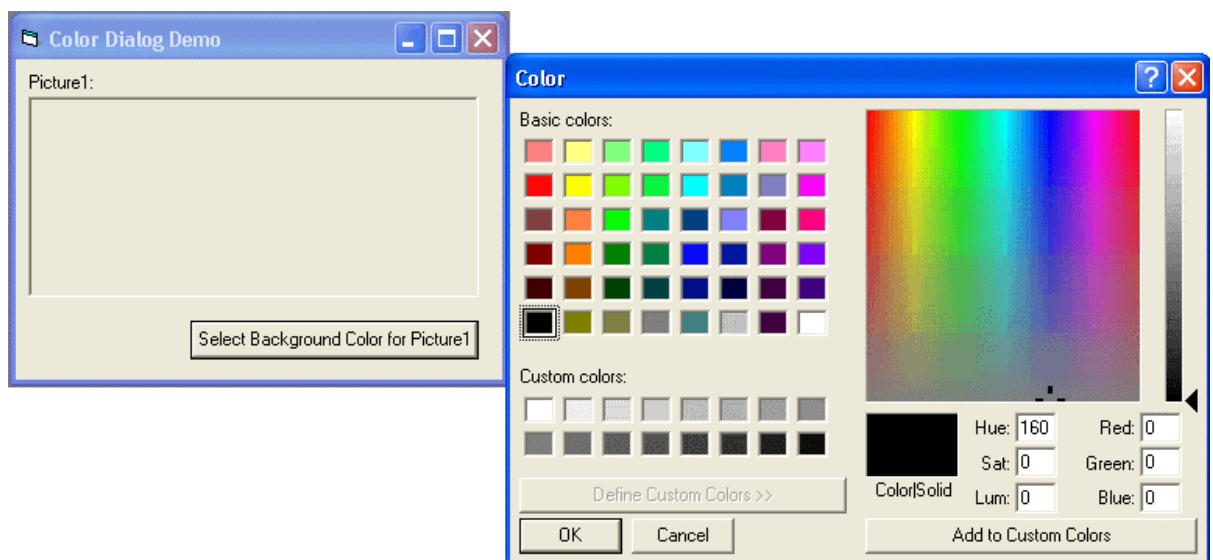
Properties Pages Dialog sẽ hiển thị với Tab **Open/Save As** có sẵn lúc đầu, chúng ta có thể đánh các tin tức như sau:



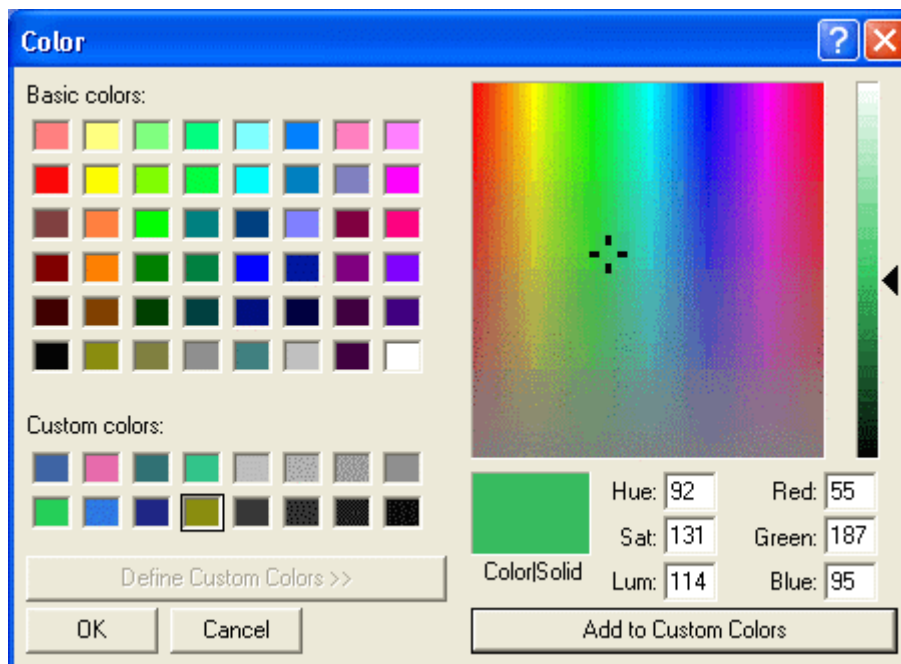
## 19.5. Các loại Dialog có sẵn để dùng

### 19.5.1 Color Dialog

**Color Dialog** cho người sử dụng một cách chọn màu rất dễ dùng. Ngoài những màu có sẵn, người sử dụng có thể tự tạo ra một màu rồi cho nó thêm vào trong bảng màu được cung cấp, gọi là **Windows Palette** bằng cách click button **Add to Custom Colors**.



Chúng ta tạo ra một màu bằng cách click chỗ có màu theo ý trong bảng màu lớn hình vuông rồi nắm hình tam giác bên phải kéo lên, kéo xuống để thay đổi độ đậm của màu như hiển thị trong hộp vuông **Color|Solid**. Khi vừa ý với màu hiển thị, chúng ta click button **Add to Custom Colors**, màu ấy sẽ được cho thêm vào nhóm **Custom Colors** nằm phía dưới, bên trái.



Ta dùng method **ShowColor** để hiển thị Color Dialog. Sau khi người sử dụng đã chọn một màu rồi, ta có thể trực tiếp assign nó cho property ForeColor hay BackColor của một control. Trong ví dụ dưới đây cái màu mà người sử dụng vừa chọn được assigned cho background của picturebox Picture1:

```
Private Sub CmdSelectColor_Click()  
    On Error GoTo NoColorChosen  
    With CommonDialog1  
        .CancelError = True  
        ' Entire dialog box is hiển thị, including the Define  
Custom Colors section  
        .Flags = cdlCCFullOpen  
        .ShowColor ' Launch the Color Dialog  
        Picture1.BackColor = .Color ' Assign selected color to  
background of Picture1  
    End With  
    Exit Sub  
NoColorChosen:  
    ' Get here if user clicks the Cancel button  
    MsgBox "You did not select a color!", vbInformation,  
"Cancelled"  
    Exit Sub  
End Sub
```

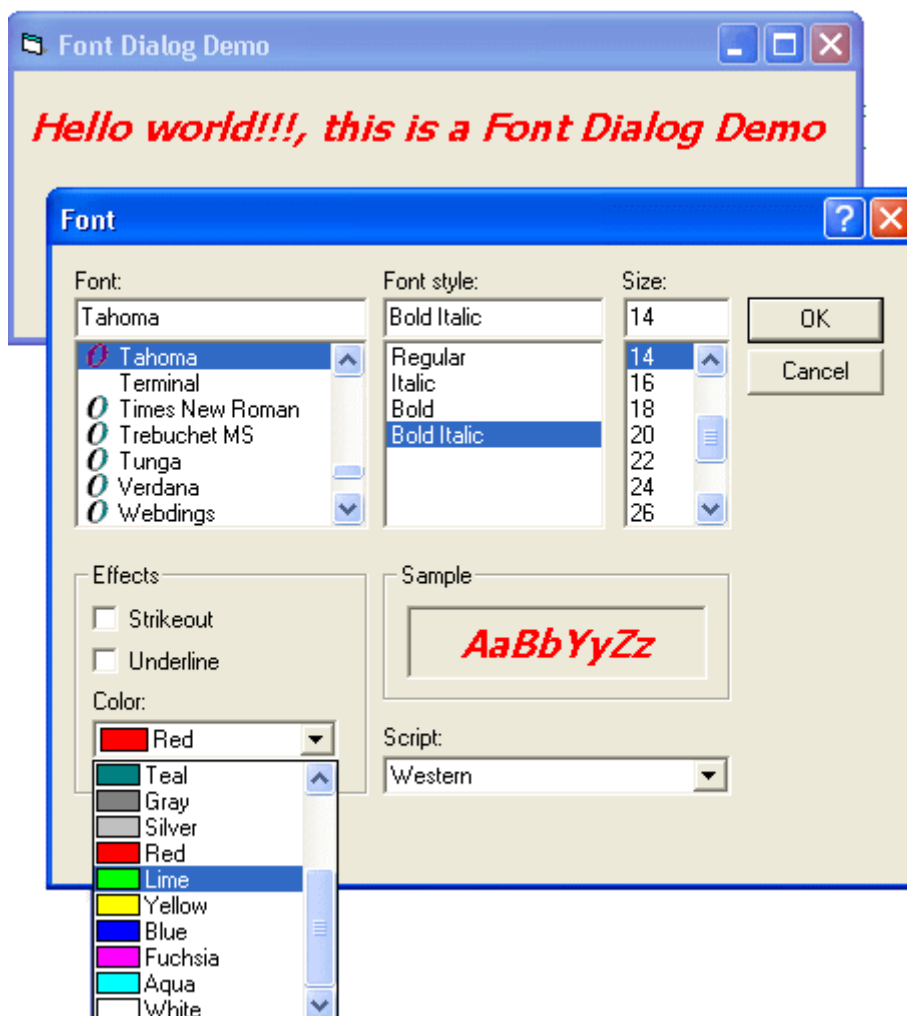


19.5.2 Font Dialog

**Font Dialog** cho ta chọn Font cho màn ảnh hay printer và chọn màu để dùng cho chữ của Font. Ta dùng method **ShowFont** để hiển thị FontDialog. Các chi tiết trình bày trong Font Dialog tùy thuộc vào trị số của Flags như sau:

Constant	Trị số	Hiệu quả
cdlCFScreenFonts	1	Chỉ hiển thị các Fonts printer hỗ trợ
cdlCFPrinterFonts	2	Chỉ hiển thị các Fonts của màn ảnh, chưa chắc tất cả đều được printer hỗ trợ
cdlCFBoth	3	Hiển thị các Fonts màn ảnh và printer
cdlCFScalableOnly	&H20000	Chỉ hiển thị các scalable Fonts như TrueType fonts mà chúng ta đã cài vào máy

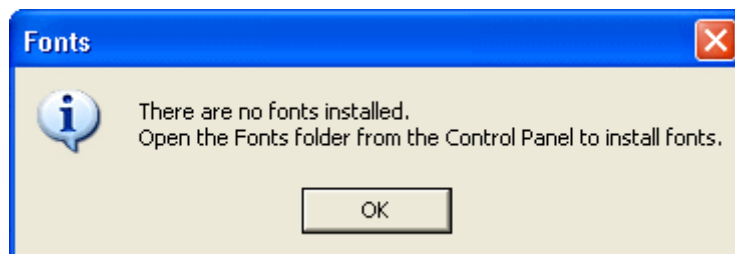
Nếu chúng ta muốn cho người sử dụng tùy chọn để chọn màu thì thêm 256 vào trị số của Flags.



Dưới đây là code để cho người sử dụng chọn Font và màu của Label1.

```
Private Sub CmdSelectFont_Click()  
    On Error GoTo NoFontChosen  
    CommonDialog1.CancelError = True  
    ' Causes the dialog box to list only the screen fonts  
supported by the system.  
    CommonDialog1.Flags = cdlCFScreenFonts + 256 ' Add 256 to  
include Color option  
    CommonDialog1.ShowFont ' Launch the Font Dialog  
With Label1.Font  
    .Bold = CommonDialog1.FontBold  
    .Italic = CommonDialog1.FontItalic  
    .Name = CommonDialog1.FontName  
    .Size = CommonDialog1.FontSize  
    .Strikethrough = CommonDialog1.FontStrikethru  
    .Underline = CommonDialog1.FontUnderline  
End With  
    Label1.ForeColor = CommonDialog1.Color  
    Label1.Caption = "Hello world!!!, this is a Font Dialog Demo"  
Exit Sub  
NoFontChosen:  
    MsgBox "No font was chosen!", vbInformation, "Cancelled"  
Exit Sub  
End Sub
```

Chú ý: Nếu chúng ta quên cho Flags một trong những hằng số nói trên chương trình sẽ cho một Error message như sau:

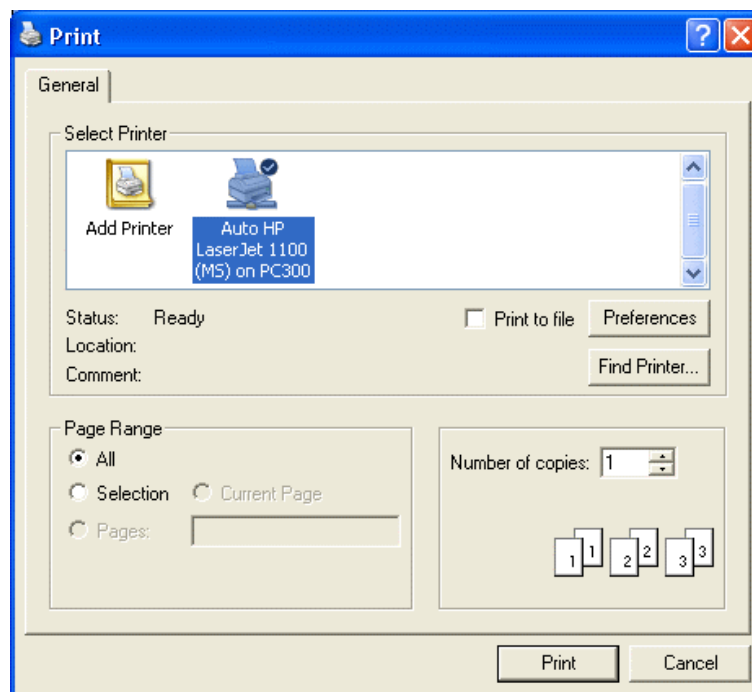


### 19.5.3 Print Dialog

**Print Font** cho ta một giao diện cũng giống như trong Microsoft Office để chọn những tùy chọn về việc in. Với Print Dialog ta có thể chọn printer nào với những đặc tính nào bằng cách click button **Properties** hay button **Preferences**. Ta cũng có thể quyết định in từ trang nào đến

trang nào của document và in bao nhiêu copies. Chỉ có điều phải lưu ý là nếu người sử dụng dùng Print Dialog để chọn một Printer khác mà trong Print Dialog ta đã chọn Property **PrinterDefault = True** thì Printer ấy sẽ trở thành Default Printer và nó cũng sẽ có hiệu lực vĩnh viễn trong cả Windows cho đến khi người sử dụng thay đổi lại.

Khác với Color và Font Dialogs, Print Dialog không đòi hỏi ta phải cho một trị số của Property Flags. Ta chỉ cần dùng Method **ShowPrinter** để hiển thị Print Dialog. Ba properties thường được dùng nhất sau khi người sử dụng chọn các tùy chọn của Print Dialog là **Copies**, **FromPage** và **ToPage**. Để cho người sử dụng các default values của những properties này, chúng ta có thể để sẵn các trị số trước khi hiển thị Print Dialog.



Dưới đây là code mẫu dùng print Dialog:

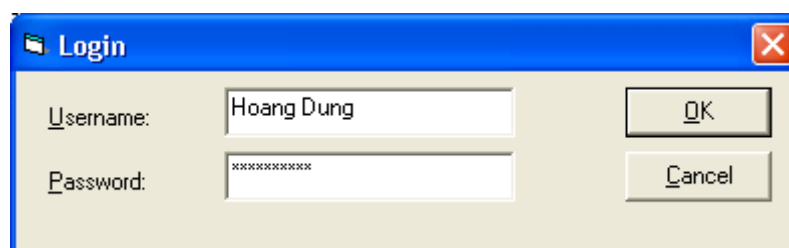
```
Private Sub CmdSelectPrinter_Click()  
    With CommonDialog1  
        .FromPage = 1  
        .ToPage = 1  
        .Copies = 1  
        .ShowPrinter  
    End With  
End Sub
```



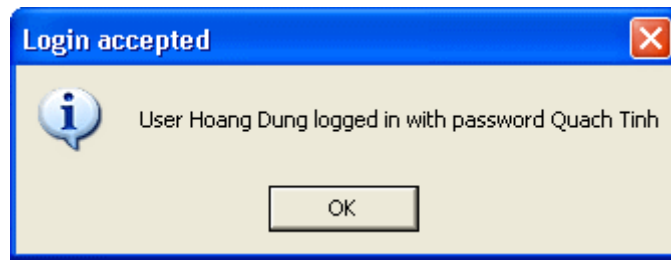
```
GetUserInfo strUserName, strPassword
  If strUserName = "" Then
    MsgBox "Login failed or aborted", vbInformation, "login
Aborted"
  Else
    MsgBox "User " & strUserName & " logged in with password "
& strPassword, vbInformation, "Login accepted"
    ' Check UserName and Password here
    ' If valid password then show the Main form of the program
which is implemented separately...
    ' frmMain.Show
  End If
End Sub

Private Sub GetUserInfo(ByRef sUserName As String, ByRef
sPassword As String)
  ' Invoke frmLogin form in Modal mode
  frmLogin.Show vbModal
  ' As soon as frmLogin is hidden, the execution gets here
  sUserName = frmLogin.txtUserName ' assign the form's
txtUserName to sUserName
  sPassword = frmLogin.txtPassword ' assign the form's
txtPassword to sPassword
  Unload frmLogin ' Unload form frmLogin
End Sub
```

Login form được hiển thị như dưới đây:



Sau khi user điền chi tiết và click **OK**, tạm thời ta chỉ hiển thị một thông điệp để xác nhận các chi tiết ấy.



Trong tương lai, chúng ta có thể viết thêm code để kiểm tra xem tên và mật khẩu có hiệu lực không. Có một vài chi tiết về form frmLogin để nó làm việc giống một Common Dialog:

Ta cho property `BorderStyle` của frmLogin là `Fixed Dialog`.

Ta cho **Property PasswordChar** của textbox txtPassword bằng "\*" để khi người sử dụng điền mật khẩu, ta chỉ thấy một dòng dấu hoa thị.

Ta cho Property `StartPosition` của form là `CenterScreen`.

**Property Default** của button `cmdOK` là `True` để khi người sử dụng ấn phím **Enter** trong form là coi như tương đương với click button cmdOK.

Tương tự như thế, **Property Cancel** của button `cmdCancel` là `True` để khi người sử dụng ấn phím **Esc** trong form là coi như tương đương với click button cmdCancel.

Tạm thời coding của event click của cmdOK và cmdCancel chỉ đơn giản như liệt kê dưới đây:

```
Sub Main()  
    Dim strUserName As String  
    Dim strPassword As String  
    ' Call local Sub getUserInfo to obtain UserName and Password  
    GetUserInfo strUserName, strPassword  
    If strUserName = "" Then  
        MsgBox "Login failed or aborted", vbInformation, "login  
Aborted"  
    Else  
        MsgBox "User " & strUserName & " logged in with password "  
& strPassword, vbInformation, "Login accepted"  
        ' Check UserName and Password here  
        ' If valid password then show the Main form of the program  
        which is implemented separately...  
        frmMain.Show  
    End If  
End Sub
```

```
End If
End Sub

Private Sub GetUserInfo(ByRef sUserName As String, ByRef
sPassword As String)
    ' Invoke frmLogin form in Modal mode
    frmLogin.Show vbModal
    ' As soon as frmLogin is hidden, the execution gets here
    sUserName = frmLogin.txtUserName ' assign the form's
txtUserName to sUserName
    sPassword = frmLogin.txtPassword ' assign the form's
txtPassword to sPassword
    Unload frmLogin ' Unload form frmLogin
End Sub
```

## BÀI 20. DÙNG ĐỒ HỌA

Visual Basic 6 có cho ta một số phương tiện về đồ họa (graphics) để trang điểm cho các cửa sổ thêm phong phú, thân thiện, dễ làm việc và đẹp mắt hơn. Dù rằng các phương tiện về đồ thị này không mạnh đủ cho ta viết những chương trình trò chơi (games) nhưng có thể đáp ứng các nhu cầu cần thiết thông thường.

Khi nói đến đồ họa, ta muốn phân biệt nó với Text thông thường. Ví dụ ta dùng Notepad để edit một bài thơ trong một cửa sổ (Ví dụ : Hôm qua em đi tỉnh về ...). Trong lúc bài thơ đang được hiển thị ta có thể sửa đổi dễ dàng bằng cách dùng bàn phím để đánh thêm các chữ mới vào, dùng các nút Delete, Backspace để xóa các chữ. Đó là ta làm việc với Text.

Bây giờ, trong khi bài thơ còn đang hiển thị, ta dùng một chương trình Graphic như PhotoImpact Capture của ULead để chụp cái hình cửa sổ của bài thơ (active window) thành giống như một photo, thì ta có một Graphic. Sau đó, muốn sửa đổi bài thơ từ graphic này ta phải dùng một graphic editor như MSPaint, PaintShopPro, v.v.. Các chữ trong hình cũng có cùng dạng graphic như ta thấy một photo, nên muốn edit phải dùng một cọ với màu sơn.

### 20.1. Màu (color) và độ mịn (resolution)

Ta nói một tấm hình tốt vì nó có màu sắc sảo và rõ ràng. Một graphic trong Windows gồm có nhiều đốm nhỏ, mỗi đốm, được gọi là một **pixel**, có khả năng hiển thị 16, 256, ... màu khác nhau.

#### 20.1.1 Độ mịn (resolution)

Thông thường độ mịn (resolution) của màn ảnh ta dùng là 800x600, tức là chiều ngang có 800 pixels và chiều cao có 600 pixels. Sau này, để xem các hình rõ hơn ta còn dùng độ mịn 1028x768 với card SuperVGA và Monitor tốt. Ta nói card SuperVGA có đến 2MB RAM, tại sao phải cần đến 2MB để hiển thị graphic đẹp?

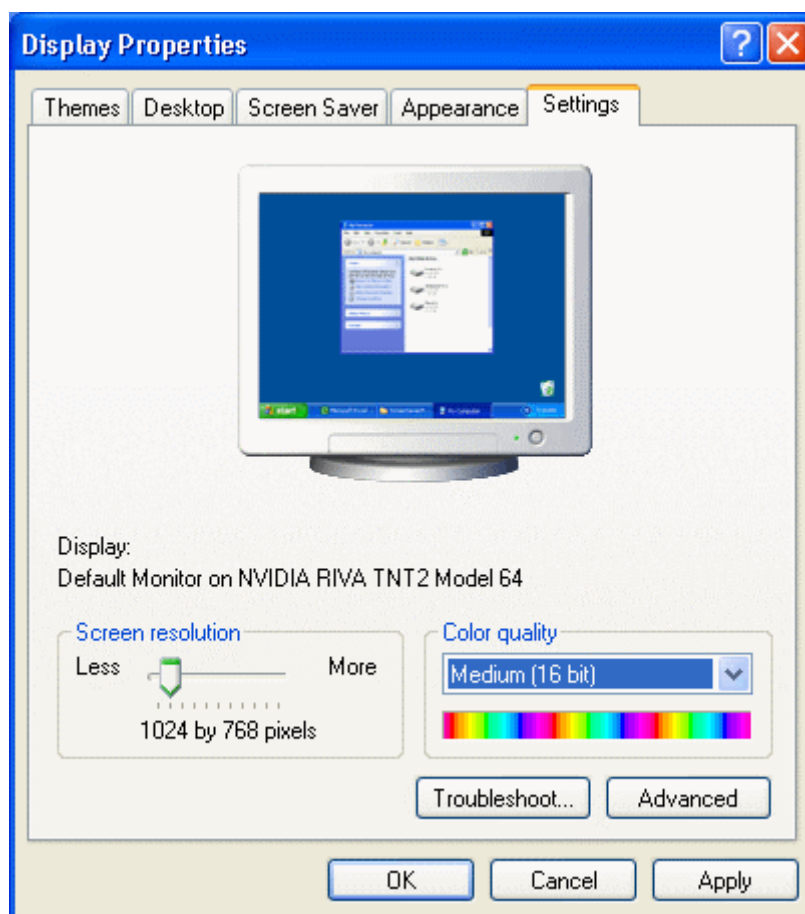
Nếu màu của mỗi pixel được biểu diễn bởi một byte dữ kiện thì với một byte ta có thể chứa một con số từ 0 đến 255. người ta đồng ý với nhau theo một quy ước rằng số 0 tượng trưng cho màu đen, số 255 tượng trưng cho màu trắng chẳng hạn. Nếu độ mịn của màn ảnh là 1024x768 thì ta sẽ cần  $1024 \times 768 = 786432$  bytes, tức là gần 0,8 MB.



Một byte có 8 bits. Đôi khi ta nghe nói 16 bit color, ý nói thay vì một byte, người ta dùng đến 2 bytes cho mỗi pixel. Như vậy mỗi pixel này có khả năng hiển thị  $2^{16} = 65536$  màu khác nhau. Muốn dùng 16 bit color cho SuperVGA, ta cần phải có  $1024 \times 768 \times 2 = 1572864$  bytes, tức là gần 1,6 MB. Đó là lý do tại sao ta cần 2MB RAM. Lưu ý là RAM của VGA (Vector Graphic Adapter) card không liên hệ gì với RAM của bộ nhớ computer.

Nên nhớ rằng cùng một graphic hiển thị trên hai màn ảnh có cùng độ mịn, ví dụ như 800x600, nhưng kích thước khác nhau, ví dụ như 14 inches và 17 inches, thì dĩ nhiên hình trên màn ảnh 17 inches sẽ lớn hơn, nhưng nó vẫn có cùng một số pixels, có điều pixel của nó lớn hơn pixel của màn ảnh 14 inches.

Nói một cách khác, nếu ta dùng màn ảnh lớn hơn thì graphic sẽ lớn hơn nhưng không có nghĩa là nó rõ hơn. Muốn thấy rõ chi tiết, ta phải làm cho graphic có độ mịn cao hơn. Ta thay đổi **Hiển thị Properties** của một màn ảnh bằng cách right click lên desktop rồi select **Properties**, kế đó click Tab **Settings** rồi chọn **Screen resolution** và **Color quality** giống như hình dưới đây:



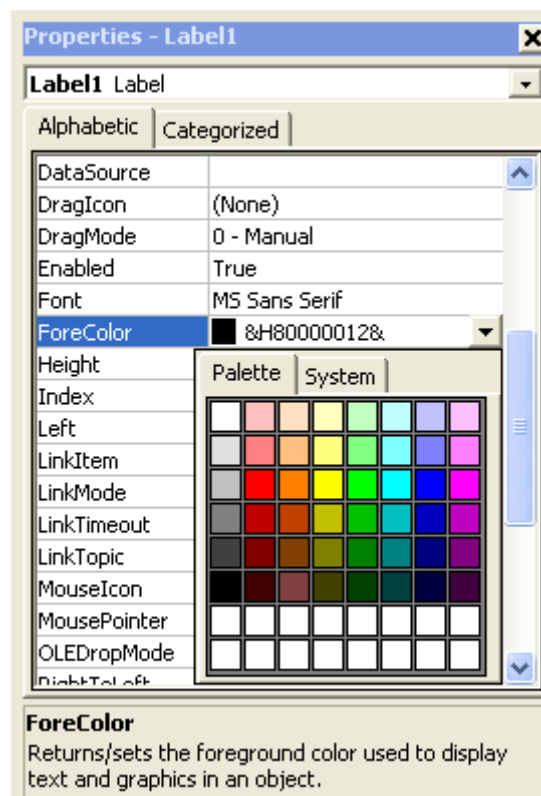
Khi ta tăng độ mịn của màn ảnh, các hình ảnh sẽ nhỏ lại vì kích thước của pixel được thu nhỏ lại. Do đó, ta có thể cho hiển thị nhiều thứ hơn trên desktop. Phẩm chất của các graphic vẫn không thay đổi, mặc dầu hình nhỏ hơn. Nhớ là muốn hình rõ hơn thì khi cấu tạo và chứa graphic, ta phải dùng một độ mịn cao. Giống như khi chụp hình, muốn hình đẹp ta cần cái máy chụp hình dùng phim lớn của thợ chuyên nghiệp và focus kỹ lưỡng, thay vì dùng máy rẻ tiền tự động, chỉ đưa lên là bấm chụp được.

### 20.1.2 Màu (color)

Khi ta dùng chỉ có một bit (chỉ có trị số 0 hay 1) cho mỗi pixel thì ta chỉ có trắng hay đen. Lúc ấy ta có thể dùng một byte (8 bits) cho 8 pixels. Dầu vậy, nếu độ mịn của graphic cao đủ, thì hình cũng đẹp. Thử xem các tuyệt tác photos trắng đen của Cao Đàm, Cao Lĩnh thì biết. Các máy Fax dùng nguyên tắc scan hình giấy cỡ A4 ra thành những pixels trắng đen rồi gửi qua đường dây điện thoại qua đầu kia để tái tạo lại hình từ những dữ kiện pixels.

Visual Basic 6 cho ta chỉ định một con số vào mỗi màu VB có thể hiển thị, hay chọn trực tiếp một màu từ Dialog. Có bốn cách:

Chúng ta chỉ định trực tiếp một con số hay chọn một màu từ cái Palette.



Chúng ta chọn một trong các hằng số định nghĩa sẵn trong VB, gọi là **intrinsic color constants** (intrinsic có nghĩa nôm na là cây nhà lá vườn hay in-built), chẳng hạn như **vbRed** , **vbBlue**. Danh sách của intrinsic color constants lấy từ VB6 online help được liệt kê dưới đây:

<b>Constant</b>	<b>Value</b>	<b>Description</b>
<b>vbBlack</b>	0x0	Black
<b>vbRed</b>	0xFF	Red
<b>vbGreen</b>	0xFF00	Green
<b>vbYellow</b>	0xFFFF	Yellow
<b>vbBlue</b>	0xFF0000	Blue
<b>vbMagenta</b>	0xFF00FF	Magenta
<b>vbCyan</b>	0xFFFF00	Cyan
<b>vbWhite</b>	0xFFFFFFFF	White

Dùng **Function QBColor** để chọn một trong 16 màu. Function QBColor xuất phát từ thời Quick Basic (QBasic) của Microsoft. QBasic là tiền thân của Visual Basic. Trong QBasic chúng ta có thể dùng các con số 1,2,3 .. để chỉ định các màu Blue, Green, Cyan , .v.v..Function QBColor giản tiện hóa cách dùng màu, người sử dụng không cần phải bận tâm về cách trộn ba thứ màu căn bản Red, Green, Blue. Chúng ta viết code một cách đơn giản như:

```
Label1.ForeColor = QBColor(2)  
QBColor(Color As Integer) As Long
```

Dưới đây là trị số các màu ta có thể dùng với Function QBColor.

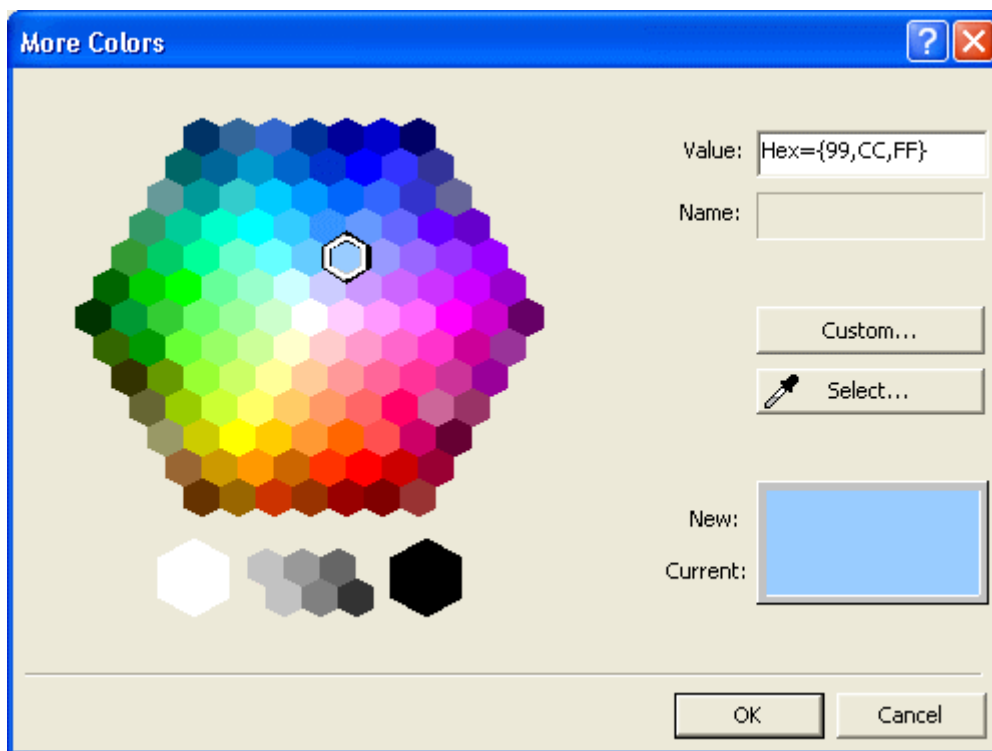
<b>Trị số</b>	<b>Màu</b>	<b>Trị số</b>	<b>Màu</b>
0	Black	8	Gray
1	Blue	9	Light Blue
2	Green	10	Light Green
3	Cyan	11	Light Cyan
4	Red	12	Light Red
5	Magenta	13	Light Magenta
6	Yellow	14	Light Yellow
7	White	15	Bright White

Dùng **Function RGB** để trộn ba màu **Red**, **Green** và **Blue**. Trong cái bảng liệt kê các intrinsic color constants phía trên, nếu để ý chúng ta sẽ thấy vbWhite(0xFFFFFFFF) là tổng số

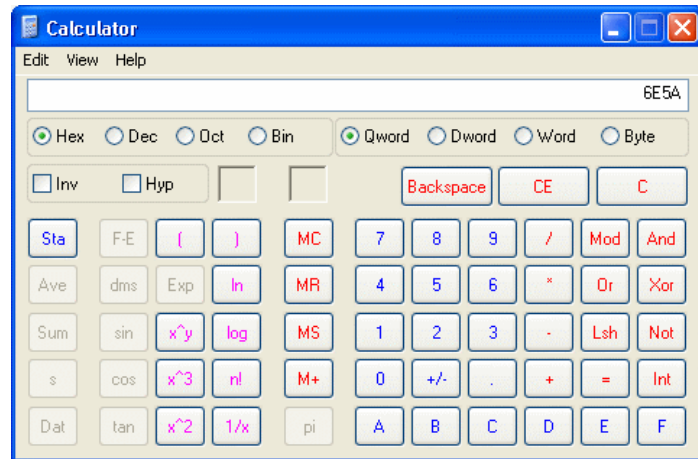
của vbRed(0x0000FF), vbGreen(0x00FF00) và vbBlue(0xFF0000). Một màu được biểu diễn bằng sự pha trộn của ba thành phần màu căn bản, mỗi màu bằng một byte có trị số từ 0 đến 255. 0 là không dùng màu ấy, 255 là dùng tối đa màu ấy.

Hệ thống số ta dùng hằng ngày là Thập Phân. Trị số 0xFF của vbRed là con số 255 viết dưới dạng Thập lục phân (Hexadecimal hay **Hex** cho gọn và ở đây được đánh dấu bằng **0x** trước con số để phân biệt với số Thập phân). Trong hệ thống số Hex ta đếm từ 0 đến 9 rồi A,B,C,D,E,F rồi qua số dòng thập lục 10, 11,.., 19, 1A, 1B, ..1E,1F,20,21..v.v. Tức là thay vì chỉ dùng 10 symbols từ 0 đến 9 trong Thập phân, ta dùng 16 symbols từ 0 đến F. Muốn biết thêm về hệ thống số Hex hãy đọc bài [Cơ số Nhị Phân](#).

Trong hình dưới đây là một ví dụ cho thấy màu xanh nhạt đã được chọn gồm ba thành phần Blue(0x990000= 153\*256\*256), Green(0xCC00= 204\*256) và Red(0xFF= 255):



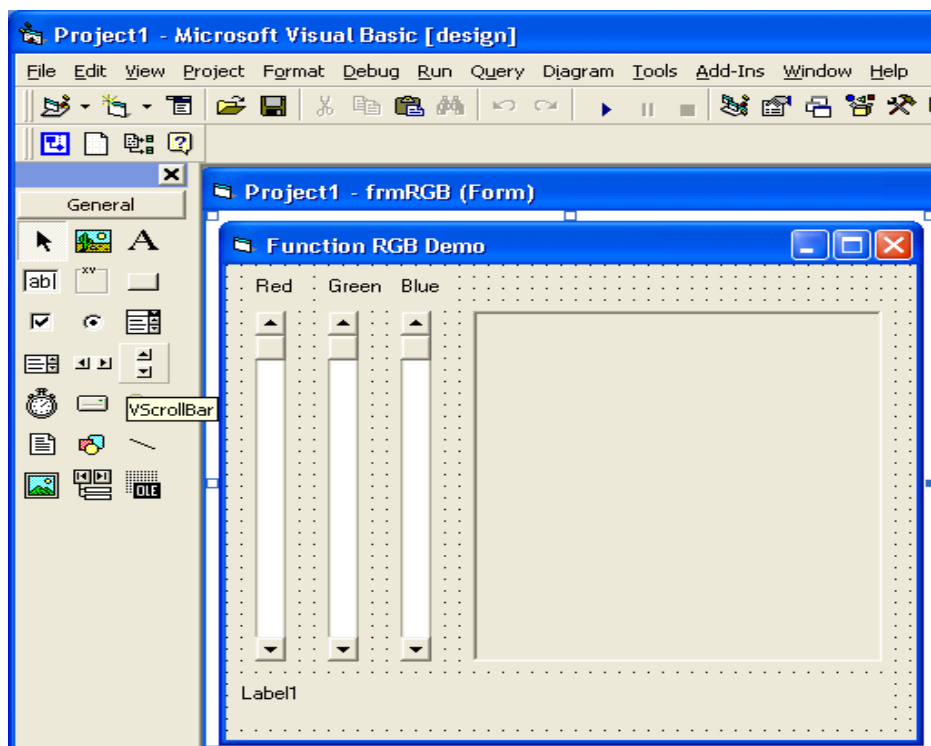
**Ghi chú:** Chúng ta có thể dùng Windows Calculator để hoán chuyển số giữa các dạng Decimal, Binary và Hexadecimal. Chọn **View|Scientific** thay vì **View|Standard**.



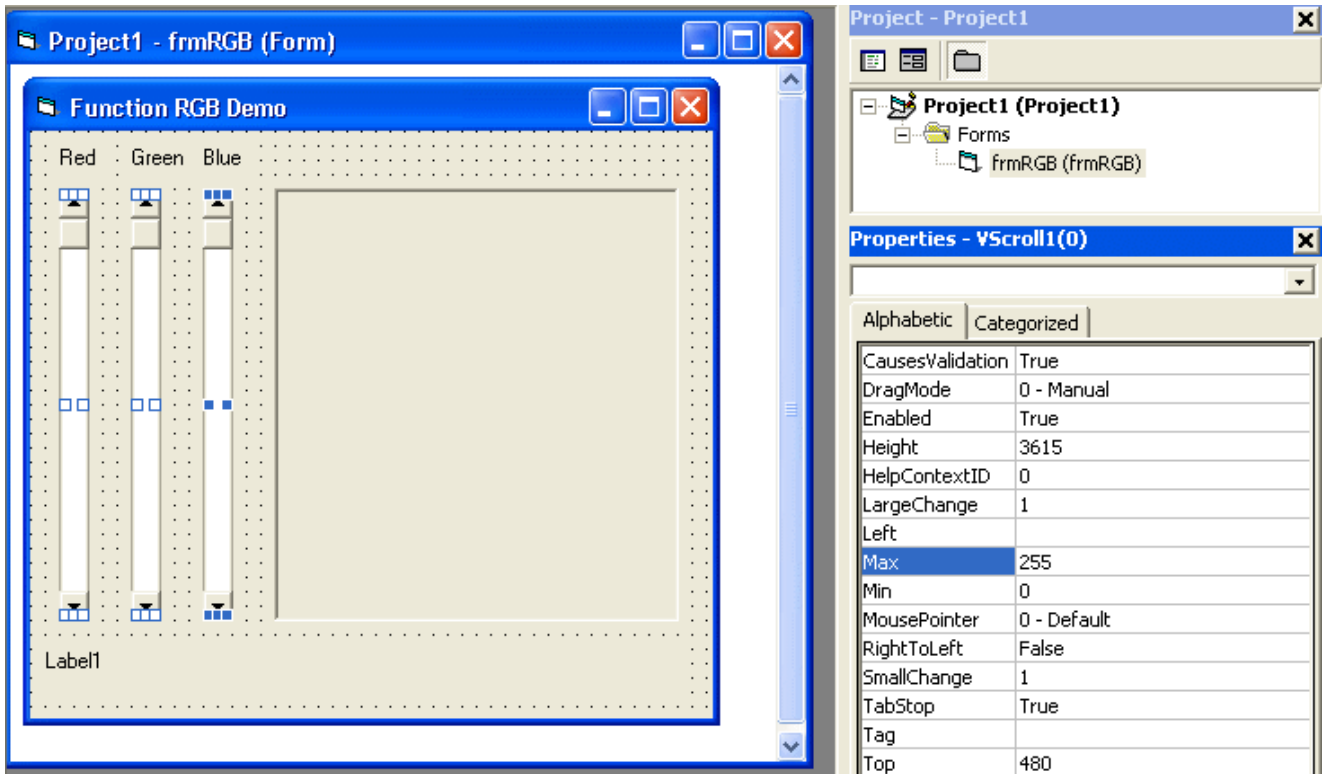
## 20.2. Function RGB

Để áp dụng Function RGB, ta sẽ viết một chương trình VB6. Chúng ta hãy khởi động một chương trình VB6 mới, bỏ vào một Label tên Label1 với Caption **Red** và một Vertical Scroll tên **VScroll1**. Kế đó select cả hai Label1 và VScroll1 rồi Copy và Paste hai lần để là thêm hai cặp. Đổi Caption của hai Label mới này ra **Green** và **Blue**. Bây giờ ta có một Array ba Vertical Scrolls cùng tên VScroll1, với index là 0,1 và 2.

Đặt một PictureBox tên **picColor** vào bên phải ba cái VScrolls. Thêm một Label phía dưới, đặt tên nó là **lblRGBValue**, nhớ clear caption của nó, đừng có để chữ Label1 như dưới đây:



Bây giờ select cả ba VScrolls và edit value của **property Max** trong cửa sổ Properties thành **255**, ý nói khi kéo cái bar của một VScroll1 lên xuống ta giới hạn trị số của nó từ Min là 0 đến Max là 255.

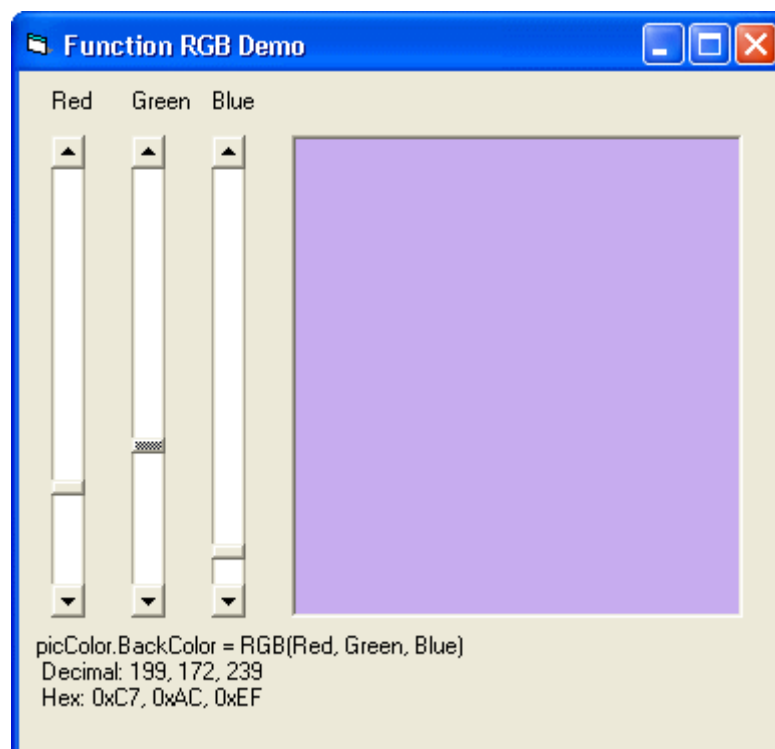


Chuyện chính ta phải làm là viết code để xử lý **Event Change** của các VScrolls. Vì chúng là một Array nên ta có thể dùng một Sub duy nhất để handle events đến từ cả ba VScrolls. Mỗi lúc một trong 3 VScrolls thay đổi trị số ta sẽ trộn ba màu Red, Green, Blue biểu diễn bởi trị số của 3 VScrolls thành màu **BackColor** của PictureBox **picColor**. Đồng thời ta cho hiển thị trị số của ba thành phần màu Red, Green và Blue trong Label **lblRGBValue**. Chúng ta hãy double click lên một trong 3 VScrolls rồi viết code như sau:

```
Private Sub VScroll1_Change(Index As Integer)
    ' Use Function RGB to mix 3 colors VScroll1(0) for Red,
    ' VScroll1(1) for Green and VScroll1(2) for Blue
    ' and assign the result to BackColor of PictureBox picColor
    picColor.BackColor = RGB(VScroll1(0).Value, VScroll1(1).Value,
VScroll1(2).Value)
    ' Variable used to prepare hiển thị string
    Dim strRGB As String
    ' Description of what is hiển thị
```

```
strRGB = "picColor.BackColor = RGB(Red, Green, Blue) " &  
vbCrLf  
' Values of Red, Green, Blue in Decimal  
strRGB = strRGB & " Decimal: " & VScroll1(0).Value & ", " &  
VScroll1(1).Value & ", " & VScroll1(2).Value & vbCrLf  
' Values of Red, Green, Blue in Hexadecimal  
strRGB = strRGB & " Hex: 0x" & Hex(VScroll1(0).Value) & ", 0x"  
& Hex(VScroll1(1).Value) & ", 0x" & Hex(VScroll1(2).Value)  
' Assign the resultant string to caption of Label lblRGBValue  
lblRGBValue.Caption = strRGB  
End Sub
```

Chúng ta hãy khởi động chương trình rồi nắm các bar của 3 VScrolls kéo lên, kéo xuống để xem kết quả. Cửa sổ của chương trình sẽ có dạng giống như dưới đây:



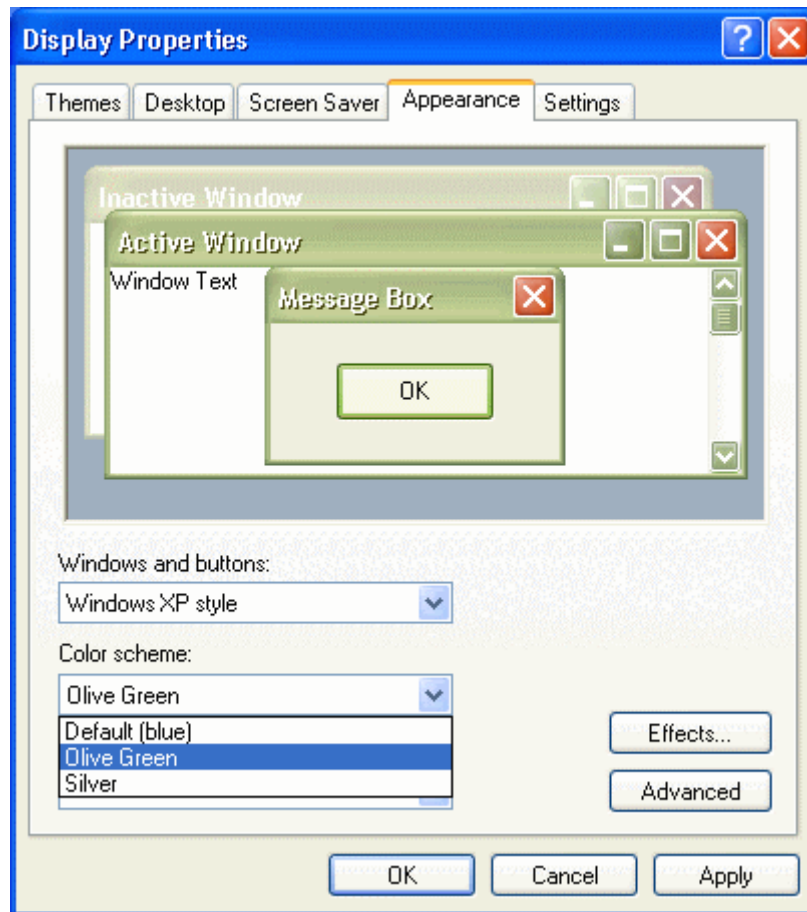
### 20.3. Color Mapping

Nếu dùng Hex Calculator đổi con số 0xFFFFFFFF ra decimal ta sẽ được 16777215, nếu kể cả số 0 ta sẽ có tổng cộng 16777216 màu. Lúc này ta bàn về 8bit (1 byte) và 16bit (2 bytes) color, nhưng ở đây ta nói chuyện 3 byte color. Như thế có thể màn ảnh không đủ khả năng để cung cấp mọi màu mà Function RGB tính ra. Vậy VGA card sẽ làm sao?

Ví dụ một card VGA chỉ hỗ trợ đến 8 bits. Nó sẽ cung cấp 256 màu khác nhau. Nếu Function RGB đòi hỏi một màu mà VGA card có thể cung cấp chính xác thì tốt, nếu không nó sẽ tìm cách dùng hai hay ba đóm gần nhau để trộn màu và cho ta ảo tưởng màu ta muốn. Công tác này được gọi là **Color Mapping** và cái màu được làm ra được gọi là **custom color**.

## 20.4. Dùng Intrinsic Color Constants

Một trong những features của MSWindows là cho ta chọn Color Scheme của Windows theo sở thích. Bình thường, Color Scheme của Windows là Blue, nhưng ta có thể chọn Olive Green hay Silver, nếu ta muốn.

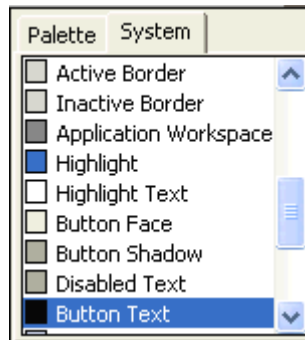


Chỉ tuy nhiên nếu ta đã dùng một màu đỏ đậm để hiển thị tuyệt đẹp thứ gì trong chương trình VB6 mà bây giờ người sử dụng tự nhiên thay đổi Color Scheme thành Olive Green chẳng hạn khiến cho màu đỏ đậm ấy coi chẳng giống ai trong cái Color Scheme mới.

Để tránh trường hợp ấy, thay vì nói thẳng ra là màu gì (xanh hay đỏ) ta nói dùng màu **vbActiveTitlebar** hay **vbDesktop**, .v.v. Dùng Intrinsic Color Constant sẽ bảo đảm màu ta



dùng sẽ được biến đổi theo Color Scheme mà người sử dụng chọn để khỏi bị trường hợp cái màu trở nên chẳng giống ai. Lúc thiết kế, ta cũng có thể chọn Intrinsic Color Constant từ Tab **System** khi chọn màu.



### 20.5. Graphic files

Khi một hình Graphic được lưu trữ theo dạng số pixels với màu của chúng như đã nói trên thì ta gọi là một **Bit Map** và tên file của nó trong disk có extension **BMP** ví dụ như **House.bmp**. Lưu trữ kiểu này cần rất nhiều memory và rất bất tiện để gởi đi hay hiển thị trên một trang Web. Do đó người ta dùng những kỹ thuật để giảm thiểu lượng memory cần để chứa graphic nhưng vẫn giữ được chất lượng của hình ảnh. Có hai dạng Graphic files rất thông dụng trên Web, mang tên với extensions là **JPG** và **GIF**. Đặc biệt với GIF files ta có thể chứa cả hoạt họa (animation), tức là một GIF file có thể chứa nhiều hình (gọi là **Frames**) để chúng lần lượt thay nhau hiển thị, cho người xem có cảm tưởng một vật đang di động.

## BÀI 21. CƠ SỞ DỮ LIỆU (DATABASE)

### 21.1. Table, Record và Field

Nói đến cơ sở dữ liệu, ta lập tức nghĩ đến SQLServer, Access hay Oracle .v.v., những nơi chứa rất nhiều dữ liệu để ta có thể lưu trữ hay lấy chúng ra một cách tiện lợi và nhanh chóng. Hầu hết các chương trình ta viết đều có truy cập cơ sở dữ liệu, và ta dùng nó như một công cụ để làm việc với rất nhiều dữ liệu trong khi tập trung vào việc lập trình phần giao diện với người dùng (người sử dụng).

Do đó ta cần có một kiến thức căn bản về kiến trúc của cơ sở dữ liệu để hiểu lý do tạo sao ta thiết kế hay truy cập nó theo những cách nhất định.

Ta sẽ dùng Access Database **biblio.mdb**, nằm ở **C:\Program Files\Microsoft Visual Studio\VB98\biblio.mdb** để minh họa các ý niệm cần biết về cơ sở dữ liệu.

Trong database này có 4 **tables**: **Authors** (tác giả), **Publishers** (nhà xuất bản), **Titles** (đề mục) và **Title Author**.

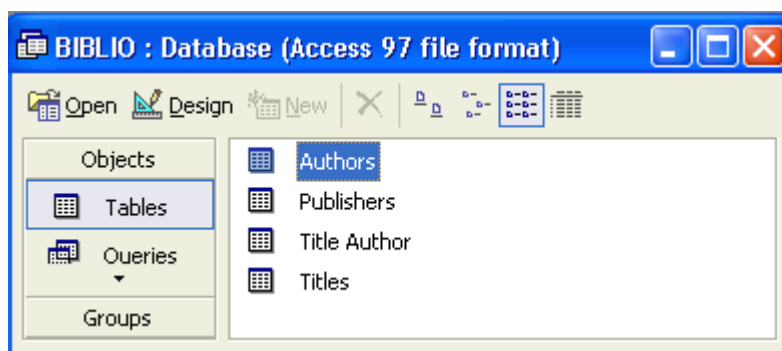
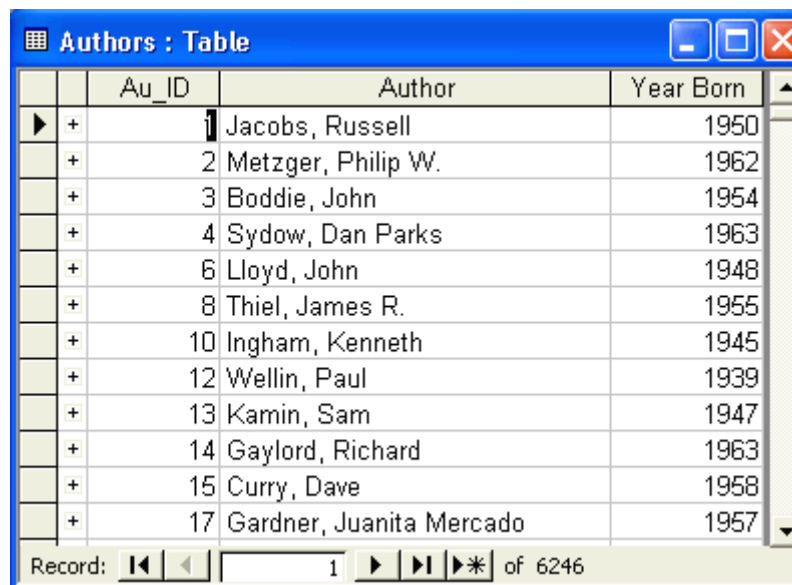


Table Authors chứa nhiều **records**. Mỗi bản ghi trong table Authors chứa 3 **fields**: **Au\_ID**, **Author** và **Year Born** (năm sinh). Ta có thể trình bày Table Authors dưới dạng một spreadsheet như sau:



	Au_ID	Author	Year Born
▶ +	1	Jacobs, Russell	1950
+ +	2	Metzger, Philip W.	1962
+ +	3	Boddie, John	1954
+ +	4	Sydow, Dan Parks	1963
+ +	6	Lloyd, John	1948
+ +	8	Thiel, James R.	1955
+ +	10	Ingham, Kenneth	1945
+ +	12	Wellin, Paul	1939
+ +	13	Kamin, Sam	1947
+ +	14	Gaylord, Richard	1963
+ +	15	Curry, Dave	1958
+ +	17	Gardner, Juanita Mercado	1957

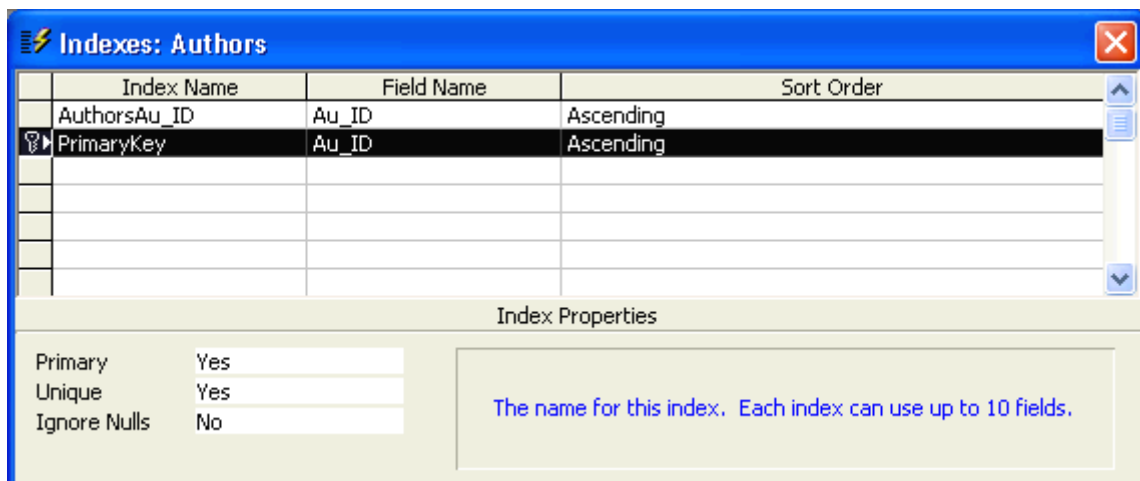
Vì cùng một field của các records hiển thị trong cùng một cột của spreadsheet, nên ta cũng nói đến một field như một **column** (cột). Và vì mỗi data record chiếm một row (dòng) của spreadsheet, nên có khi ta cũng nói đến một bản ghi như một **row**.

Thật tình mà nói, ta không cần phải có một computer để lưu trữ hay làm việc với một table như Authors này. Ta đã có thể dùng một hộp card, trên mỗi card ta ghi các chi tiết Au\_ID, Author và Year Born của một Author. Như thế mỗi tấm card tương đương với một bản ghi và nguyên cái hộp là tương đương với Table Authors.

Ta sẽ sắp các card trong hộp theo thứ tự của số Au\_ID để có thể truy cập bản ghi nhanh chóng khi biết Au\_ID. Chỉ khổ một nỗi, nếu muốn biết có bao nhiêu tác giả, trong số 300 card trong hộp, già hơn 50 tuổi thì phải mất vài phút mới có thể trả lời được. Database trong computer nhanh hơn một hệ thống bằng tay (Manual) là ở chỗ đó.

## 21.2. Primary Key và Index

Để tránh sự trùng hợp, thường thường có một field của bản ghi, ví dụ như Au\_ID trong Table Authors, được dành ra để chứa một trị số độc đáo (unique). Tức là trong Table Authors chỉ có một bản ghi với field Au\_ID có trị số ấy mà thôi. Ta gọi nó là **Primary Key**.

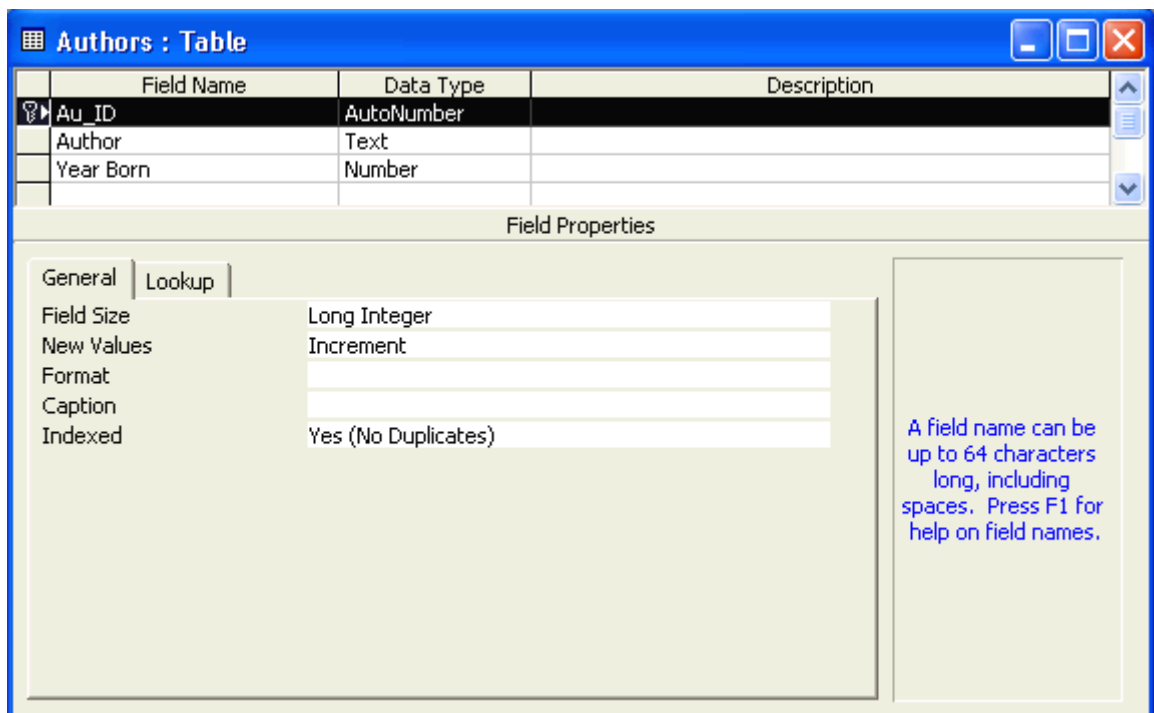


Không phải lúc nào ta cũng muốn truy cập một bản ghi Author dựa vào Au\_ID. Nhiều khi ta muốn dùng chính tên của Author để truy cập, do đó ta cũng cần phải sort sẵn các records theo thứ tự alphabet. Ta cũng có thể hợp nhiều fields lại để sort các records. Thật ra, chính các records không cần phải được dời đi để nằm đúng vị trí thứ tự. Ta chỉ cần nhớ vị trí của nó ở đâu trong table là đủ rồi.

Cái field hay tập hợp của nhiều fields (ví dụ surname và firstname ) để dùng vào việc sorting này được gọi là **Index** (ngón tay chỉ). Một Table có thể có một hay nhiều Index. Mỗi Index sẽ là một table nhỏ của những **pointers**, chứa vị trí của các records trong Table Authors. Nó giống như mục lục index ở cuối một cuốn sách chứa trang số để chỉ ta đến đúng phần ta muốn tìm trong quyển sách.

Khi thiết kế một Table ta chỉ định **Datatype** của mỗi field để có thể kiểm tra data cho vào có hợp lệ hay không. Các Datatypes thông dụng là Number, String (để chứa Text), Boolean (Yes/No), Currency (để chứa trị số tiền) và Date (để chứa date/time). Datatype Number lại gồm có nhiều loại datatypes về con số như Integer, Long (integer chiếm 32 bits), Single, Double, .v.v.

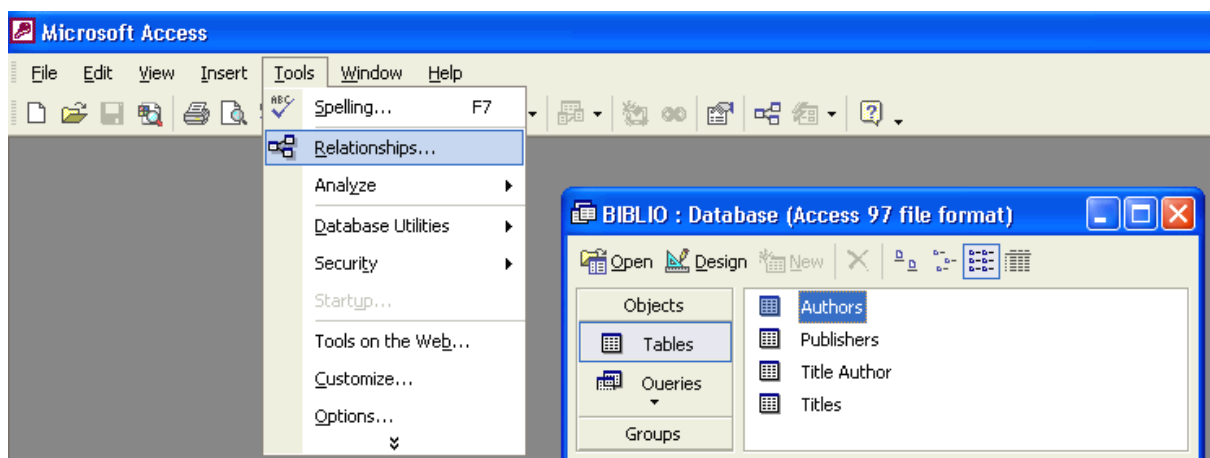
Dưới đây là Datatypes của các fields trong bản ghi Author:



Có loại Datatype đặc biệt tên là **AutoNumber**. Thật ra nó là Long nhưng trị số được phát sinh tự động mỗi khi ta thêm một bản ghi mới vào Table. Ta không làm gì hơn là phải chấp nhận con số ấy.

### 21.3. Relationship và Foreign Key

Bây giờ, nếu chúng ta đang chạy Microsoft Access để quan sát database biblio.mdb, chúng ta có thể dùng Menu Command **Tools | Relationships** như sau để xem sự liên hệ (relationships) giữa các tables.





Nếu giả sử ta bắt đầu thiết kế database với Table Titles, rồi quyết định tách các chi tiết về nhà xuất bản để vào một table mới, tên Publishers, thì kỹ thuật ấy được gọi là normalization. Nói một cách khác, normalization là thiết kế các tables trong database làm sao để mỗi loại mảnh dữ kiện (không phải là Key) chỉ xuất hiện ở một chỗ.

Trong mỗi liên hệ one-to-many giữa tables Publishers và Titles, field PubID là Primary Key trong table Publishers. Trong table Titles, field PubID được gọi là **Foreign Key**, có nghĩa rằng đây là Primary Key của một table lạ (foreign). Hay nói một cách khác, trong khi làm việc với table Titles, lúc nào cần chi tiết một nhà xuất bản, ta sẽ lấy chìa khóa lạ (Foreign Key) dùng làm Primary Key của Table Publishers để truy cập bản ghi ta muốn. Để ý là chính Table Titles có Primary Key ISBN của nó.

### 21.4. Relational Database

Một database có nhiều tables và hỗ trợ các liên hệ, nhất là one-to-many, được gọi là **Relational Database**. Khi thiết kế một database, ta sẽ tìm cách sắp đặt các dữ liệu từ thế giới thật bên ngoài vào trong các tables. Ta sẽ quyết định chọn các cột (columns/fields) nào, chọn Primary Key, Index và thiết lập các mối liên hệ, tức là đặt các Foreign Key ở đâu.

### 21.5. Các lợi ích

Trong số các lợi ích của một thiết kế Relational Database có:

- Sửa đổi dữ kiện, cho vào records mới hay delete (gạch bỏ) records có sẵn rất hiệu quả (nhanh).
- Truy cập dữ kiện, làm báo cáo (Reports) cũng rất hiệu quả. Vì dữ kiện được sắp đặt thứ tự và có quy củ nên ta có thể tin cậy tính tình của database. Vì hầu hết dữ kiện nằm trong database, thay vì trong chương trình ứng dụng, nên database tự có documentation (tài liệu cắt nghĩa).
- Dễ sửa đổi chính cấu trúc của các tables.

## 21.6. Integrity Rules (các quy luật liên chính)

**Integrity Rules** được dùng để nói về những qui luật cần phải tuân theo trong khi làm việc với database để đảm bảo là database còn tốt. Có hai loại quy luật: luật tổng quát (General Integrity Rules) và luật riêng cho database (Database-Specific Integrity Rules). Các luật riêng này thường tùy thuộc vào các quy luật về mâu dịch (Business Rules).

### 21.6.1 General Integrity Rules

Có hai quy luật liên chính liên hệ hoàn toàn vào database: Entity (bản thể) Integrity Rule và Referential (chỉ đến) Integrity Rule.

**Entity Integrity Rule** nói rằng **Primary Key** không thể thiếu được, tức là không thể có trị số **NULL**. Quy luật này xác nhận là vì mỗi Primary Key đưa đến một row độc đáo trong table, nên dĩ nhiên nó phải có một trị số đàng hoàng.

Lưu ý là Primary Key có thể là một **Composite Key**, tức là tập hợp của một số keys (columns/fields), nên nhất định không có key nào trong số các columns là **NULL** được.

**Referential Integrity Rule** nói rằng database không thể chứa một Foreign Key mà không có Primary Key tương ứng của nó trong một table khác. Điều ấy hàm ý rằng:

- Ta không thể thêm một Row vào trong một Table với trị số Foreign Key trong Row ấy không tìm thấy trong danh sách Primary Key của table bên phía **one** (1) mà nó liên hệ.
- Nếu có thay đổi trị số của Primary Key của một Row hay xóa một Row trong table bên phía **one** (1) thì ta không thể để các records trong table bên phía **many** ( $\infty$ ) chứa những rows trở thành mồ côi (orphans).

Nói chung, có ba tùy chọn (options) ta có thể chọn khi thay đổi trị số của Primary Key của một Row hay xóa một Row trong table bên phía **one** (1):

- **Disallow** (không cho làm): Hoàn toàn không cho phép chuyện này xảy ra.
- **Cascade** (ảnh hưởng dây chuyền): Nếu trị số Primary Key bị thay đổi thì trị số Foreign Key tương ứng trong các records của table bên phía **many** ( $\infty$ ) được thay đổi theo. Nếu Row chứa Primary Key bị deleted thì các records tương ứng trong table bên phía **many** ( $\infty$ ) bị deleted theo.



- **Nullify** (cho thành NULL): Nếu Row chứa Primary Key bị deleted thì trị số Foreign Key tương ứng trong các records của table bên phía **many** ( $\infty$ ) được đổi thành NULL, để hàm ý đừng có đi tìm thêm chi tiết ở đâu cả.

### 21.6.2 Database-Specific Integrity Rules

Những quy luật liên chính nào khác không phải là Entity Integrity Rule hay Referential Integrity Rule thì được gọi là Database-Specific Integrity Rules. Những quy luật này dựa vào chính loại database và nhất là tùy thuộc vào các quy luật về mậu dịch (Business Rules) ta dùng cho database, ví dụ như mỗi bản ghi về tiền lương của công nhân phải có một field Số Thuế (Tax Number) do sở Thuế Vụ phát hành cho công dân. Lưu ý là các quy luật này cũng quan trọng không kém các quy luật tổng quát về liên chính. Nếu ta không áp dụng các Database-Specific Integrity Rules nghiêm chỉnh thì database có thể bị hư và không còn dùng được.

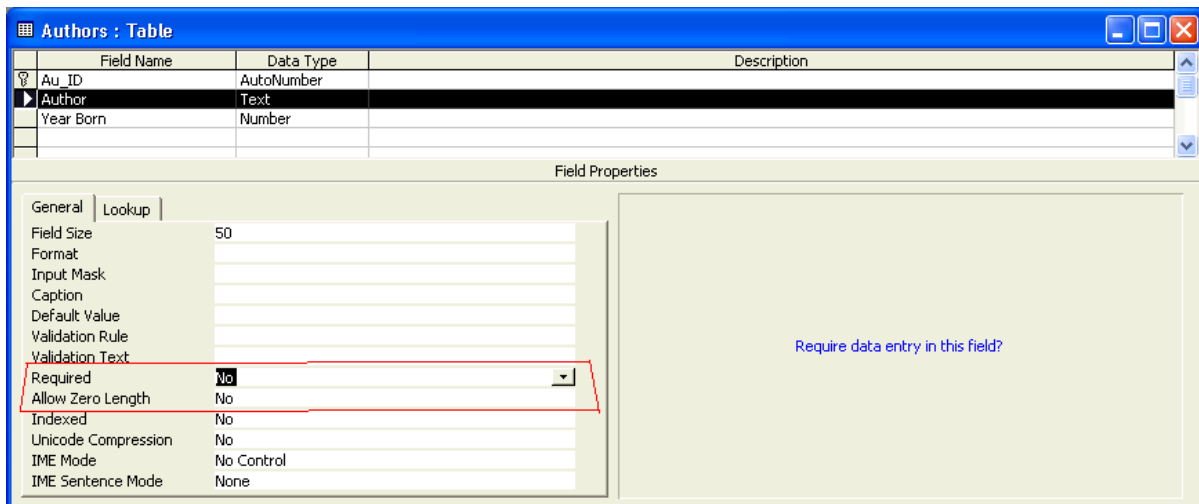
## 21.7. Microsoft Access Database Management System (MSAccess DBMS)

Microsoft Access Database Management System gồm có Database Engine và những công cụ đi chung để cung cấp cho người sử dụng một môi trường làm việc thân thiện với database, như Database Design (thiết kế các tables và mối liên hệ), Data entry và báo cáo (reports). Kèm theo với Visual Basic 6.0 khi ta mua là một copy của Database Engine của MSAccess. Tên nó là **Jet Database Engine**, cái lõi của MSAccess DBMS. Các chương trình VB6 có thể truy cập database qua Jet Database Engine.

Nếu trên computer của chúng ta có cài sẵn MSAccess, thì chúng ta có thể dùng đó để thiết kế các tables của database hay cho data vào các tables.

## 21.8. Properties Required và Allow Zero Length

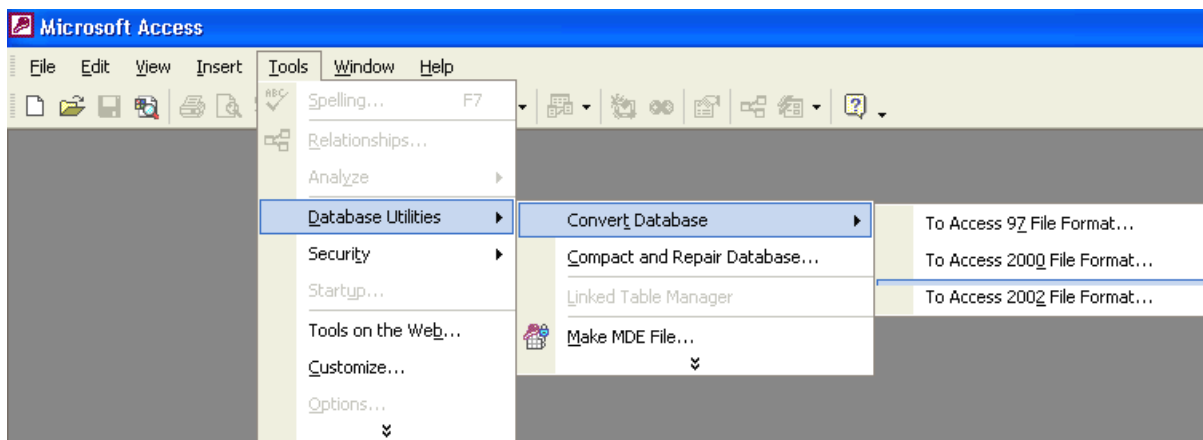
Khi thiết kế một table field, lưu ý property Required và nhất là property Allow Zero Length của Text. Nếu property Required của một field là Yes thì ta không thể update (viết) một bản ghi với field ấy có trị số NULL. Nếu một Text field có property Allow Zero Length là No thì ta không thể update một bản ghi khi field ấy chứa một empty string.



Khi ta tạo một bản ghi lần đầu, nếu không cho trị số của một field, thì field ấy có trị số là NULL. Thường thường, Visual Basic 6.0 không thích NULL value nên ta phải thử một field với Function IsNULL() để đảm bảo nó không có trị số NULL trước khi làm việc với nó. Nếu IsNULL trả về trị số False thì ta có thể làm việc với field ấy. Nhớ là khi trị số NULL được dùng trong một expression, ngay cả khi chương trình không cho Error, kết quả cũng là NULL.

## 21.9. Làm việc với các versions khác nhau

Nếu máy chúng ta đang chạy MSAccess2002 thì chúng ta có thể làm việc với Access database file version 97, 2000 và 2002. Nếu cần phải convert từ version này qua version khác, chúng ta có thể dùng Access DBMS Menu Command **Tools | Database Utilities | Convert Database | To Access 2002 File Format...** Nếu muốn giữ nguyên version, chúng ta có thể convert database qua File Format 2002 để sửa đổi, rồi sau đó convert trở lại File Format cũ.

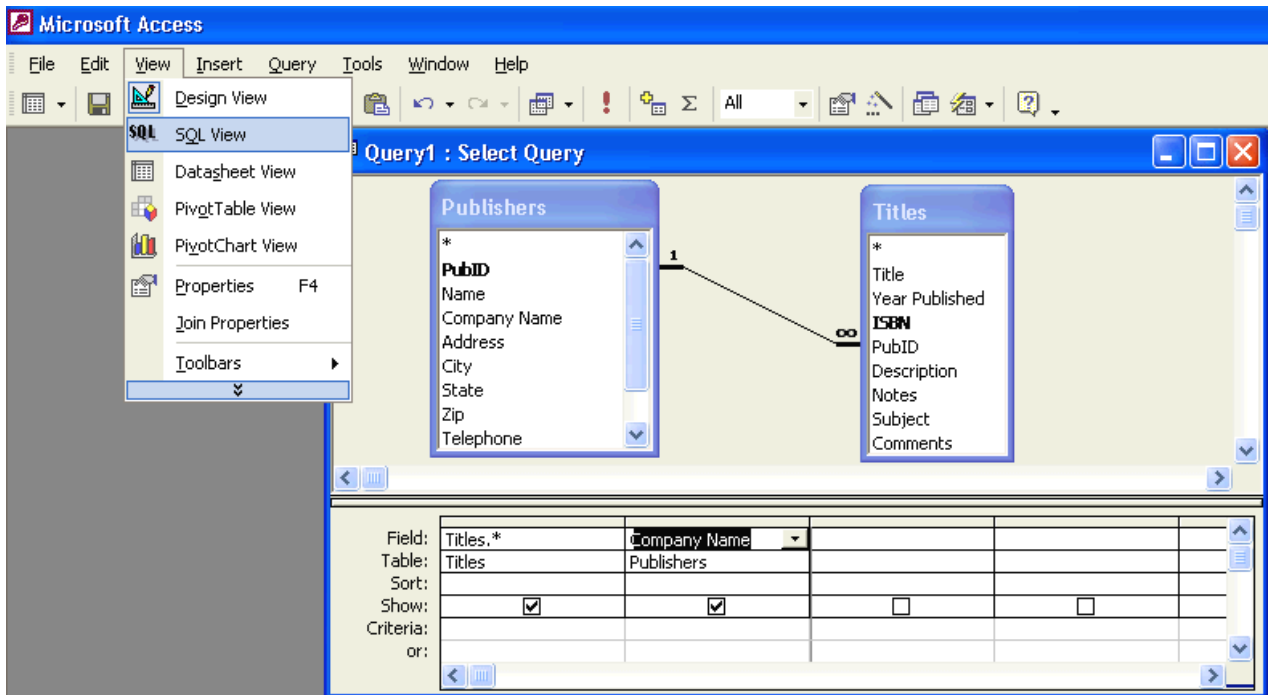


Access database file lớn lên rất nhanh, vì các records đã bị deleted vẫn còn nằm nguyên, nên mỗi tuần chúng ta nên nhớ nén nó lại để bỏ hết các records đã bị deleted bằng cách dùng

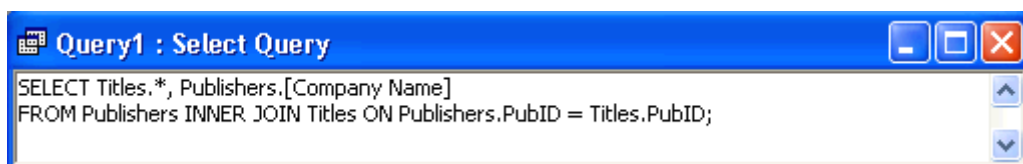
Access DBMS Menu Command **Tools** | **Database Utilities** | **Compact and Repair Database...** hay dùng **function DBEngine.CompactDatabase** trong VB6.

## 21.10. Dùng Query để viết SQL

Một cách để truy cập database là dùng ngôn ngữ **Structured Query Language (SQL)** theo chuẩn do ISO/IEC phát hành năm 1992, gọi tắt là **SQL92**. Tất cả mọi database thông dụng đều hỗ trợ SQL, mặc dầu nhiều khi chúng còn cho thêm nhiều chức năng rất hay nhưng không nằm trong chuẩn. Các lệnh SQL thông dụng là **SELECT**, **UPDATE**, **INSERT** và **DELETE**. Ta có thể dùng phương tiện thiết kế Query của MSAccess để viết SQL. Sau khi thiết kế Query bằng cách drag drop các fields, chúng ta có thể dùng Menu Command **View** | **View SQL** như sau:

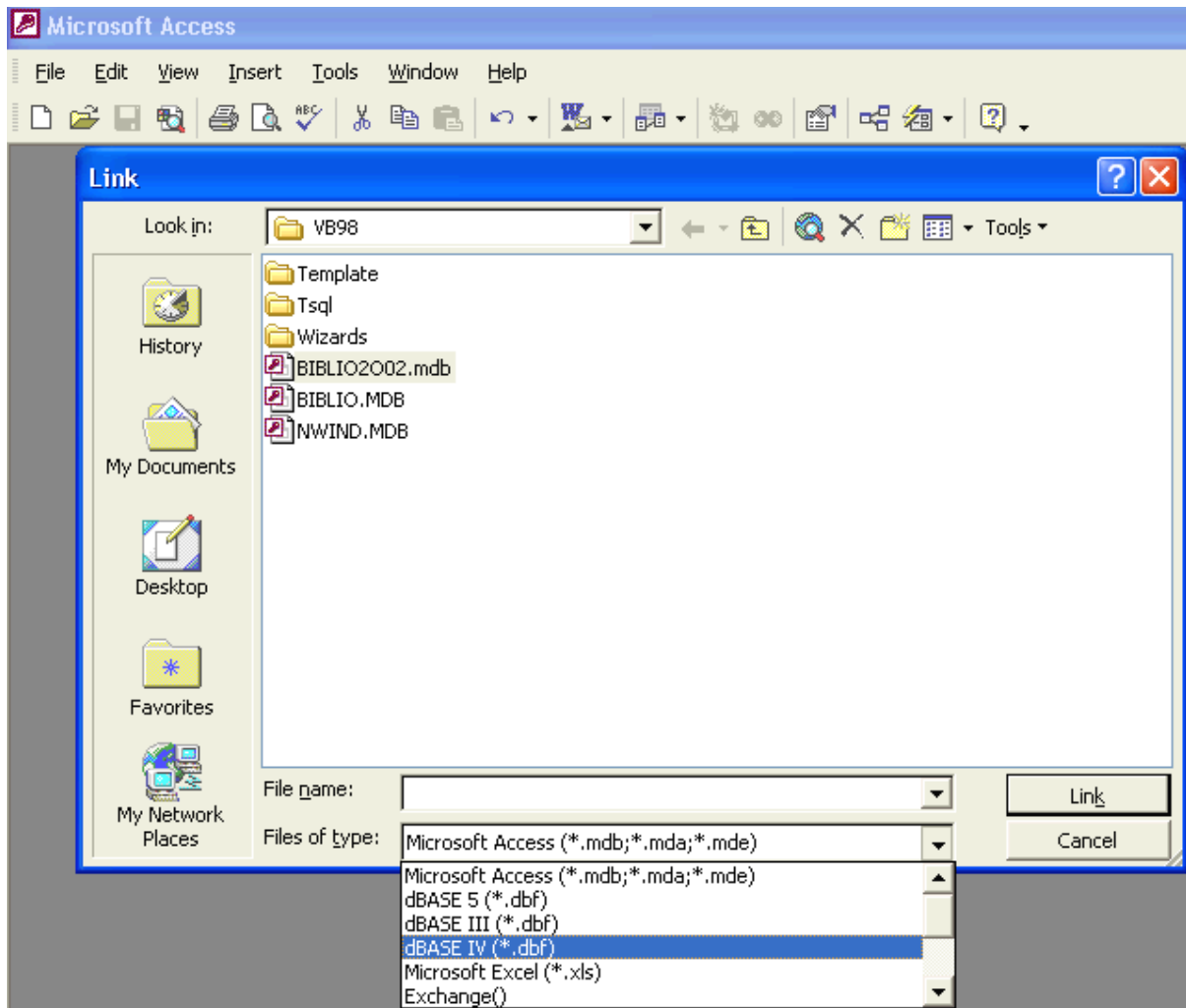


Tiếp theo đây là SQL statement của Query bên trên mà chúng ta có thể copy để paste vào trong code VB6:



### 21.11. Dùng Link Table để làm việc trực tiếp với database loại khác

Ta có thể dùng một database loại khác, như DBase, trực tiếp trong VB6 như dùng một Access database bình thường. Muốn thiết lập mối nối ấy, chúng ta dùng Menu Command **File | Get External Data | Link Tables...** rồi chọn loại DBase và chính file của table mà chúng ta muốn dùng để nhét nó vào Access database đang mở:



### 21.12. Database Server và một số khái niệm

Dù Jet Database Engine là một relational database rất tốt và hiệu năng, nó thuộc loại **File Based database**, tức là nó thụ động, không chạy một mình nhưng phải tùy thuộc vào chương trình dùng nó. File Based database không thích hợp với những ứng dụng có nhiều người dùng cùng một lúc.

Trong khi đó, một Database Server như **SQLServer** chạy riêng để phục vụ bất cứ chương trình khách (client) nào cần. Database Server thích hợp cho các ứng dụng có nhiều người sử dụng vì chỉ có một mình nó chịu trách nhiệm truy cập dữ liệu cho mọi clients. Nó có thể chứa nhiều routines địa phương, gọi là **Stored Procedures**, để thực hiện các công tác client yêu cầu rất hiệu năng. Database Server thường có cách đối phó hữu hiệu khi có sự cố về phần cứng như đĩa hư hay cúp điện. Ngoài ra, Database Server có sẵn các phương tiện về an ninh và backup. Nó cũng có thêm các chức năng để dùng cho mạng.

Ngày nay ta thu thập dữ liệu dưới nhiều hình thức như Email, Word documents, Spreadsheets. Không nhất thiết dữ liệu luôn luôn được chứa dưới dạng table của những records và không nhất thiết dữ liệu luôn luôn được lưu trữ trong một database đang hoạt động. Dù vậy, chúng vẫn được xem như database dưới mắt một chương trình ứng dụng. Do đó, ta dùng từ **Data Store** (kho dữ liệu) thay thế cho database để nói đến nơi chứa dữ liệu. Và đối với chương trình sử dụng dữ liệu, ta nói đến **Data Source** (nguồn dữ liệu) thay vì database.

Khi lập trình bằng VB6 để truy cập database, ta nhìn database một cách trừu tượng, tức là đầu nó là Access, DBase, SQLServer hay Oracle ta cũng xem như nhau. Nếu có thay đổi loại database bên dưới, cách lập trình của ta cũng không thay đổi bao nhiêu.

Trong tương lai, một **XML file** cũng có thể được xem như một database nho nhỏ. Nó có thể đứng một mình hay là một table trích ra từ một database chính huy. XML là một chuẩn mà ta có thể dùng để import/export dữ liệu với tất cả mọi loại database hỗ trợ XML. Ta có thể trao đổi dữ liệu trên mạng Internet dưới dạng XML. Ngoài ra, thay vì làm việc trực tiếp với một database lớn, ta có thể trích ra vài tables từ database ấy thành một XML file. Kế đó ta chỉ lập trình với XML file cho đến khi kết thúc sẽ trộn (merge/reconcile) tập tin XML với database lớn. Nếu phần lớn các chương trình áp dụng được thiết kế để làm việc cách này, thì trong tương lai ta không cần một Database Server thật mạnh.

## BÀI 22. SỬ DỤNG CONTROL DATA

### 22.1. Control Data

Từ VB5, Visual Basic cho lập trình viên một control để truy cập cơ sở dữ liệu, tên nó chỉ đơn sơ là **Data**. Như ta biết, có một cơ sở dữ liệu Microsoft gói kèm theo VB6 - đó là **Jet Database Engine**. Jet Database Engine là công cụ xử lý dữ liệu của MS Access Database Management System.

Cho đến VB5, Microsoft cho ta ba kỹ thuật chính:

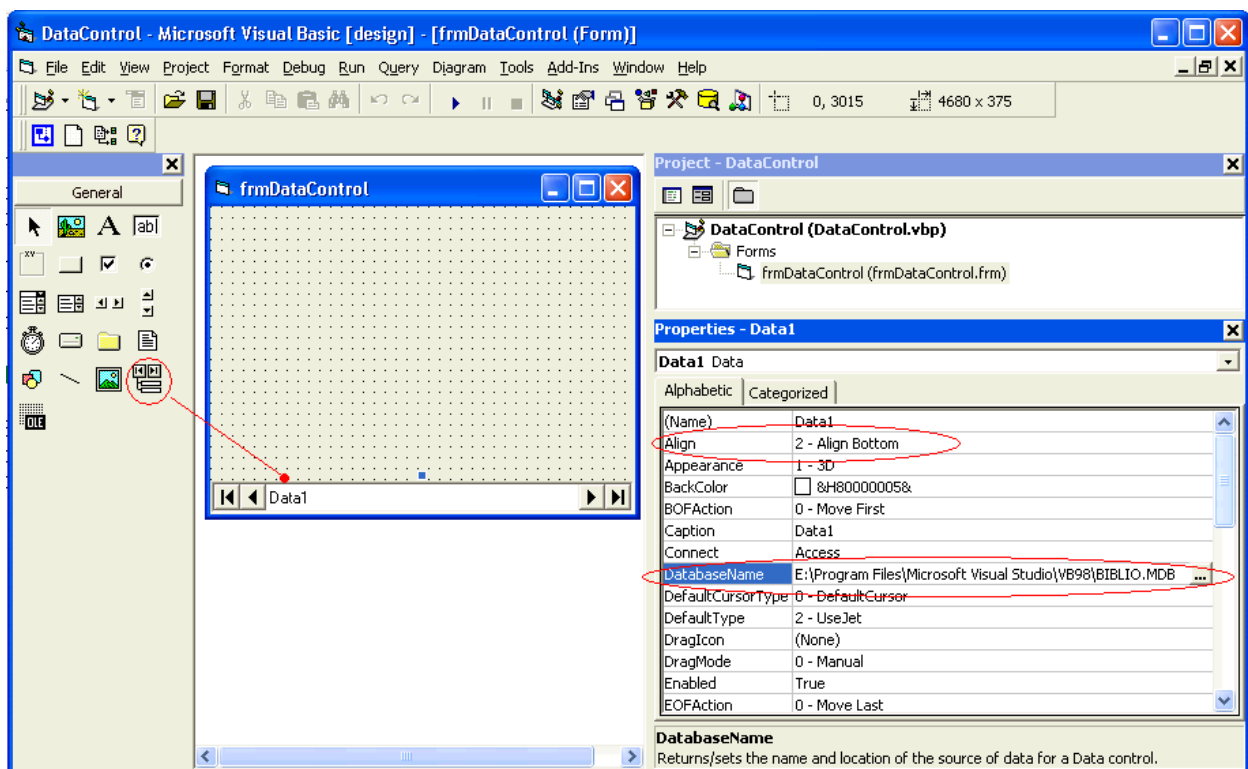
- **DAO (Data Access Objects)**: DAO là kỹ thuật đặc biệt của Microsoft, chỉ để dùng với Jet Database Engine. Nó rất dễ dùng, hiệu năng và tiện, nhưng bị giới hạn trong phạm vi MS Access. Dầu vậy, nó rất thịnh hành vì rất dễ sử dụng và mang lại hiệu quả cao.
- **ODBC (Open Database Connectivity)**: ODBC được thiết kế để cho phép người sử dụng nối với đủ loại databases mà chỉ dùng một method duy nhất. Điều này cất bớt gánh nặng cho lập trình viên, để chỉ cần học một kỹ thuật lập trình duy nhất mà có thể làm việc với bất cứ loại database nào. Nhất là khi sau này nếu cần phải thay đổi loại database, như nâng cấp từ Access lên SQLServer chẳng hạn, thì sự sửa đổi về coding rất ít. Khi dùng ODBC chung với DAO, ta có thể cho Access Database nối với các databases khác. Có một bất lợi của ODBC là hơi phức tạp khi sử dụng.
- **RDO (Remote Data Object)**: Một trong những lý do chính để RDO được thiết kế là giải quyết khó khăn về sự rắc rối của ODBC. Cách lập trình với RDO đơn giản như DAO, nhưng thật ra nó dùng ODBC nên cho phép người sử dụng nối với nhiều databases. Tuy nhiên, RDO không được thịnh hành lắm.

VB6 tiếp tục hỗ trợ các kỹ thuật nói trên, và cho thêm một kỹ thuật truy cập database mới, rất quan trọng, đó là **ADO (ActiveX Data Objects)**. Trong một bài tới ta sẽ khảo sát về ADO với những ưu điểm của nó. Tuy nhiên, vì DAO rất đơn giản và hiệu năng nên ta vẫn có thể tiếp tục dùng nó rất hữu hiệu trong hầu hết các áp dụng. Do đó bài này và bài kế sẽ tập trung vào những kỹ thuật lập trình phổ biến với DAO.

Cách dùng giản tiện của control Data là đặt nó lên một Form rồi làm việc với những Properties của nó. Chúng ta hãy bắt đầu một dự án VB6 mới, cho nó tên **DataControl** bằng cách click tên project trong Project Explorer bên phải rồi edit property Name trong Properties Window.

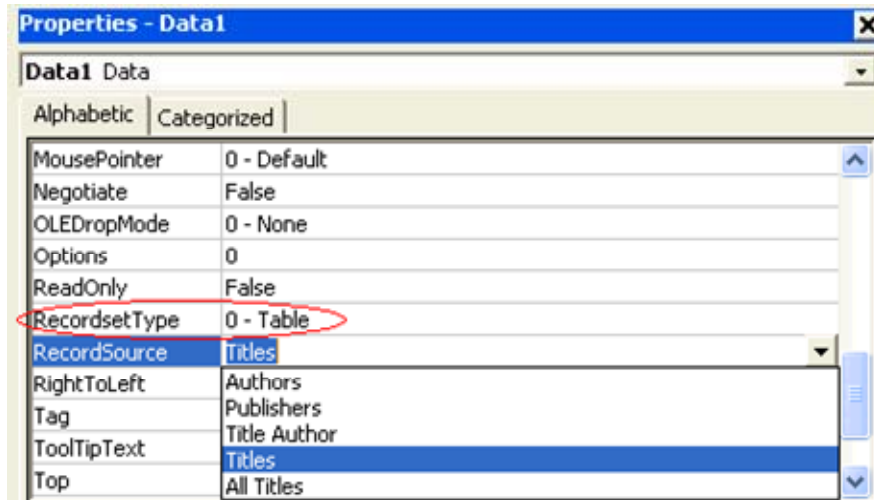
DoubleClick lên Icon của Control Data trong Toolbox. Một Control Data tên **Data1** sẽ hiện ra trên Form. Muốn cho nó nằm bên dưới Form, giống như một StatusBar, hãy set **property Align** của nó trong Properties Window thành **2 - Align Bottom**.

Click bên phải dòng **property DatabaseName**, kế đó click lên nút browse có ba chấm để chọn một file Access database từ giao thoại cho Data1. Ở đây ta chọn **E:\Program Files\Microsoft Visual Studio\VB98\BIBLIO.MDB**, trong computer của chúng ta có thể nó nằm trên disk C hay D.



Trong chương trình này ta muốn làm việc với table Titles của database BIBLIO.MDB, để xem và edit các records. Để ý property DefaultType của Data1 có trị số 2- UseJet, tức là dùng kỹ thuật DAO, thay vì dùng kỹ thuật ODBC.

Khi chúng ta click lên **property Recordsource** của Data1, rồi click lên cái tam giác nhỏ bên phải, một ComboBox sẽ mở ra cho ta thấy danh sách các tables trong database. Chúng ta hãy chọn **Titles**. Để ý **property RecordsetType** của Data1 có trị số là **0 - Table**:



Thuật ngữ mới mà ta sẽ dùng thường xuyên khi truy cập dữ liệu trong VB6 là Recordset (bộ các bản ghi). Recordset là một tập hợp các bản ghi, nó có thể chứa một số các bản ghi hay không có bản ghi nào cả. Một bản ghi trong Recordset có thể là một bản ghi lấy từ một Table. Trong trường hợp ấy có thể ta lấy về tất cả records trong table hay chỉ những records thỏa mãn một điều kiện, ví dụ như ta chỉ muốn lấy các bản ghi của những sách xuất bản trước năm 1990 (Year Published < 1990).

Một bản ghi trong Recordset cũng có thể là tập hợp các cột (columns) từ hai hay nhiều tables qua các mối liên hệ one-to-one và one-to-many. Ví dụ như khi lấy các records từ table Titles, ta muốn có thêm chi tiết tên công ty (Company Name) và điện thoại (Telephone) của nhà xuất bản (table Publishers) bằng cách dùng Foreign Key PubID trong table Titles làm Primary Key trong table Publishers để lấy các chi tiết ấy. Nếu chúng ta chưa nắm vững ý niệm Foreign Key thì hãy đọc lại bài Database.

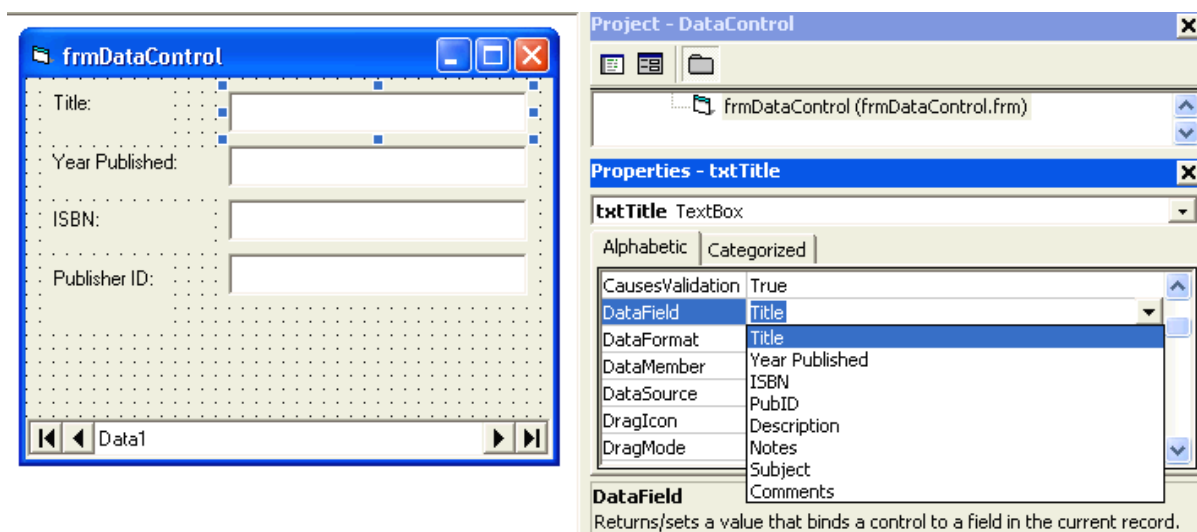
Trong trường hợp ấy ta có thể xem như có một virtual table là tập hợp của hai tables Titles và Publishers.

Bây giờ chúng ta hãy đặt lên Form 4 labels với captions: Title, Year Published, ISBN và Publisher ID. Kế đó cho thêm 4 textboxes tương ứng và đặt tên chúng là txtTitle, txtYearPublished, txtISBN và txtPublisherID.

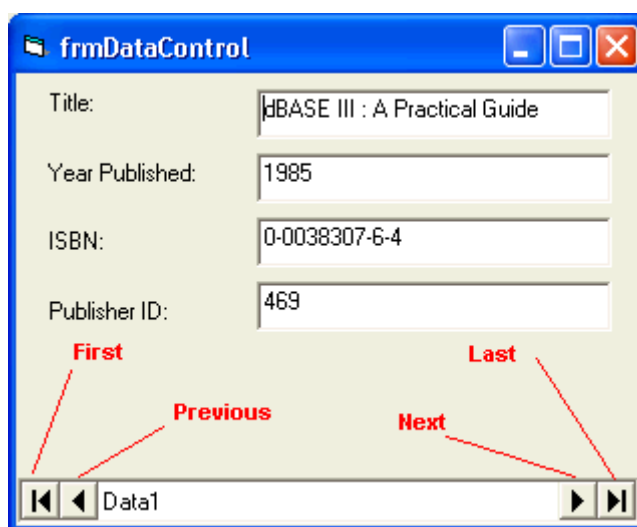


Chọn textbox txtTitle, rồi set property Datasource của nó trong Properties Window thành Data1. Khi click lên property Datafield của txtTitle và mở ComboBox ra chúng ta sẽ thấy liệt kê tên các Fields trong table Titles. Đó là vì Data1 được coi như trung gian lấy table Titles từ database. Ở đây ta sẽ chọn cột Title.

Lập lại công tác này cho 3 textboxes kia, và chọn các cột Year Published (năm xuất bản), ISBN (số lý lịch trong thư viện quốc tế), và PubID (số lý lịch nhà xuất bản) làm Datafield cho chúng.



Tới đây, mặc dầu chưa viết một dòng code, ta có thể chạy chương trình được rồi. Nó sẽ hiển thị chi tiết của bản ghi đầu tiên trong table Titles như dưới đây:



Chúng ta có thể bấm các nút di chuyển Navigator Buttons để đi đến các bản ghi đầu (first), trước (previous), kế (next) và cuối (last). Mỗi lần chúng ta di chuyển đến một bản ghi mới là chi tiết của bản ghi ấy sẽ hiển thị. Nếu không dùng các Navigator Buttons, ta cũng có thể code

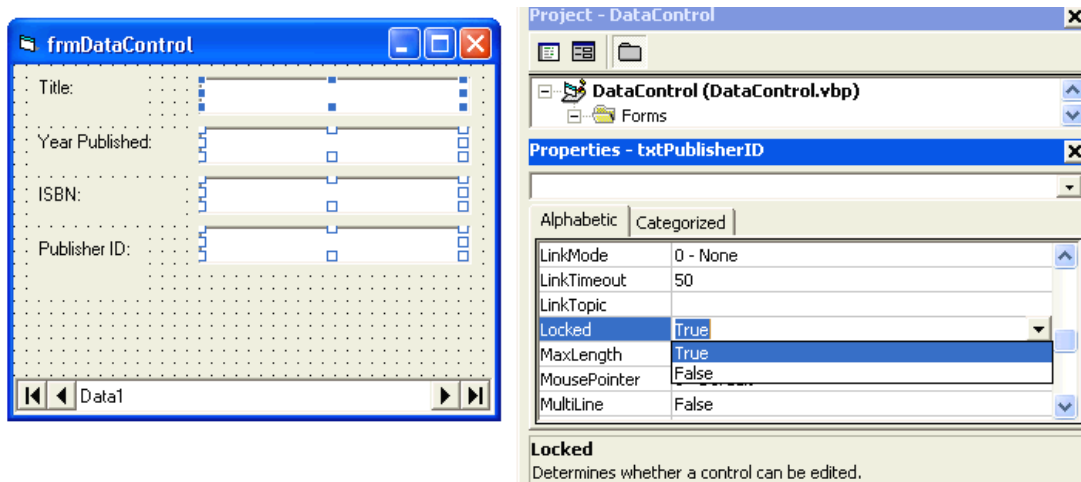
để làm công tác tương đương bằng cách gọi các Recordset methods MoveFirst, MovePrevious, MoveNext và MoveLast.

Khi bản ghi cuối của Recordset đang hiển thị, nếu ta gọi method MoveLast thì property EOF (End-Of-File) của Recordset trở thành True. Tương tự như vậy, khi bản ghi thứ nhất của Recordset đang hiển thị, nếu ta gọi method MovePrevious thì property BOF (Begin-Of-File) của Recordset trở thành True. Nếu một Recordset không có chứa một bản ghi nào cả thì cả hai properties EOF và BOF đều là True.

Đặc tính hiển thị dữ liệu trong các textbox theo đúng bản ghi hiện thời (current record) được gọi là data binding hay data bound (ràng buộc dữ liệu) và control TextBox hỗ trợ chức năng này được nói là Data Aware (nhận biết dữ liệu).

Khi bản ghi đầu tiên đang hiển thị, nếu chúng ta edit Year Published để đổi từ 1985 thành 1983 rồi click Navigator button Next để hiển thị bản ghi thứ nhì, kế đó click Navigator button Previous để hiển thị lại bản ghi đầu tiên thì chúng ta sẽ thấy là field Year Published của bản ghi đầu tiên đã thật sự được thay đổi (updated) thành 1983.

Điều này có nghĩa rằng khi Data1 navigates từ bản ghi này đến bản ghi khác thì nếu bản ghi này đã có sự thay đổi vì người sử dụng edited, nó lưu trữ sự thay đổi đó trước khi di chuyển. Chưa chắc là chúng ta muốn điều này, do đó, nếu chúng ta không muốn người sử dụng tình cờ edit một bản ghi thì chúng ta có thể set property Locked của các textboxes ấy thành True để người sử dụng không thể edit các textboxes như trong hình dưới đây:



## 22.2. Chỉ định vị trí Database lúc chạy chương trình

Cách chỉ định tên DatabaseName trong giai đoạn thiết kế (at design time) ta đã dùng trước đây tuy tiện lợi nhưng hơi nguy hiểm, vì khi ta cài chương trình này lên computer của khách, chưa chắc file database ấy nằm trong một folder có cùng tên. Ví dụ trên computer mình thì database nằm trong folder E:\Program Files\Microsoft Visual Studio\VB98, nhưng trên computer của khách thì database nằm trong folder C:\VB6\DataControl chẳng hạn. Do đó, khi chương trình khởi động ta nên xác định lại vị trí của database. Giả dụ ta muốn để database trong cùng một folder với chương trình đang chạy, ta có thể dùng **property Path** của Application Object **App** như sau:

```
Dim AppFolder As String
Private Sub Form_Load()
    ' Fetch Folder where this program EXE resides
    AppFolder = App.Path
    ' make sure it ends with a back slash
    If Right(AppFolder, 1) <> "\" Then AppFolder = AppFolder & "\"
    ' Assign Full path database filename to Data1
    Data1.DatabaseName = AppFolder & "BIBLIO.MDB"
End Sub
```

Với cách code nói trên ta sẽ đảm bảo chương trình tìm thấy file database đúng chỗ, không cần biết người ta cài chương trình chúng ta ở đâu trong hard disk của computer khách.

## 22.3. Thêm bớt các Records

Chương trình trên dùng cũng tạm được, nhưng nó không cho ta phương tiện để thêm (add), bớt (delete) các records. Bây giờ chúng ta hãy để vào Form 5 buttons tên: **cmdEdit**, **cmdNew**, **cmdDelete**, **cmdUpdate** và **cmdCancel**.

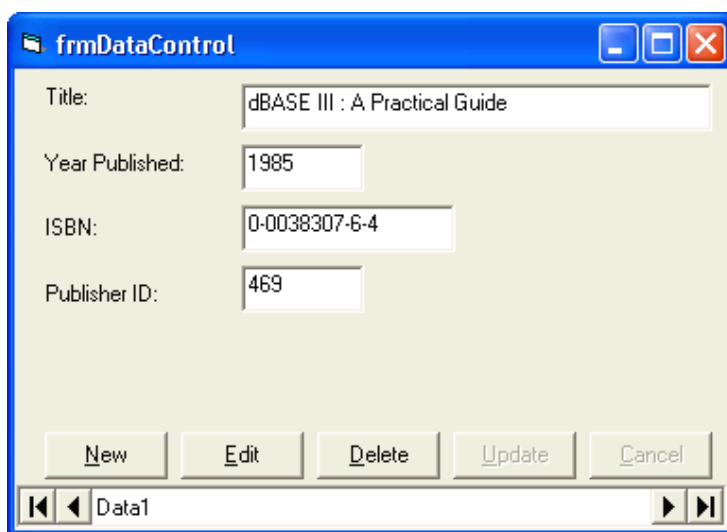
Mặc dầu chúng ta không thấy, nhưng thật ra Control Data **Data1** có một **property Recordset** và khi ta dùng Navigator buttons là di chuyển từ bản ghi này đến bản ghi khác trong Recordset ấy. Ta có thể nói đến nó bằng Notation (cách viết) **Data1.Recordset**, và mỗi lần muốn lấy Recordset mới nhất từ database ta dùng **method Refresh** như **Data1.Recordset.Refresh**.

Lúc chương trình mới khởi động, người sử dụng đang xem (browsing) các records thì hai buttons **Update** và **Cancel** không cần phải làm việc. Do đó ta sẽ nhân tiện Lock (khóa) các textboxes và disable (làm cho bất lực) hai buttons này vì không cần dùng chúng.

Trong **Sub SetControls** dưới đây, ta dùng một parameter gọi là **Editing** với trị số False hay True tùy theo người sử dụng đang Browse hay Edit, ta gọi là **Browse mode** và **Edit mode**. Trong **Edit mode**, các Textboxes được unlocked (mở khóa) và các nút **cmdNew**, **cmdDelete** và **cmdEdit** trở nên bất lực:

```
Sub SetControls(ByVal Editing As Boolean)
    ' Lock/Unlock textboxes
    txtTitle.Locked = Not Editing
    txtYearPublished.Locked = Not Editing
    txtISBN.Locked = Not Editing
    txtPublisherID.Locked = Not Editing
    ' Enable/Disable buttons
    CmdUpdate.Enabled = Editing
    CmdCancel.Enabled = Editing
    CmdDelete.Enabled = Not Editing
    cmdNew.Enabled = Not Editing
    CmdEdit.Enabled = Not Editing
End Sub
```

Trong **Browse mode**, Form có dạng như sau:



Sub SetControls được gọi trong **Sub Form\_Load** khi chương trình khởi động và trong **Sub CmdEdit** khi người sử dụng click nút **Edit** như sau:

```
Private Sub Form_Load()  
    ' Fetch Folder where this program EXE resides  
    AppFolder = App.Path  
    ' make sure it ends with a back slash  
    If Right(AppFolder, 1) <> "\" Then AppFolder = AppFolder & "\"  
    ' Assign Full path database filename to Data1  
    Data1.DatabaseName = AppFolder & "BIBLIO.MDB"  
    ' Place controls in Browse Mode  
    SetControls (False)  
End Sub  
Private Sub CmdEdit_Click()  
    ' Place controls in Edit Mode  
    SetControls (True)  
End Sub
```

Khi ta xóa một bản ghi trong recordset, vị trí của bản ghi hiện tại (current record) vẫn không thay đổi. Do đó, sau khi xóa một bản ghi ta phải **MoveNext**. Tuy nhiên, nếu ta vừa xóa bản ghi cuối của Recordset thì sau khi MoveNext, **property EOF** của Recordset sẽ thành True. Thành ra ta phải kiểm tra điều đó, nếu đúng vậy thì lại phải **MoveLast** để hiển thị bản ghi cuối của Recordset như trong code của **Sub cmdDelete\_Click** dưới đây:

```
Private Sub CmdDelete_Click()  
    On Error GoTo DeleteErr  
    With Data1.Recordset  
        ' Delete new record  
        .Delete  
        ' Move to next record  
        .MoveNext  
        If .EOF Then .MoveLast  
    End With  
    Exit Sub  
DeleteErr:  
    MsgBox Err.Description  
    Exit Sub  
End Sub
```

Trong lúc code, ta Update (cập nhật) một bản ghi trong Recordset bằng method **Update**. Nhưng ta chỉ có thể gọi method Update của một Recordset khi Recordset đang ở trong **Edit** hay **AddNew mode**. Ta đặt một Recordset vào Edit mode bằng cách gọi **method Edit** của

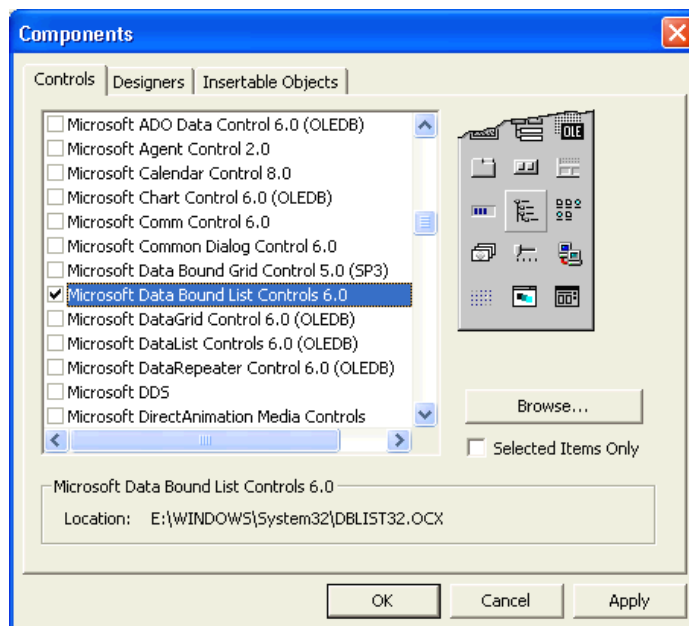
Recordset, ví dụ như **Data1.Recordset.Edit**. Tương tự như vậy, ta đặt một Recordset vào AddNew mode bằng cách gọi **method AddNew** của Recordset, ví dụ như **Data1.Recordset.AddNew**.

```
Private Sub cmdNew_Click()  
    ' Place Recordset into Recordset AddNew mode  
    Data1.Recordset.AddNew  
    ' Place controls in Edit Mode  
    SetControls (True)  
End Sub
```

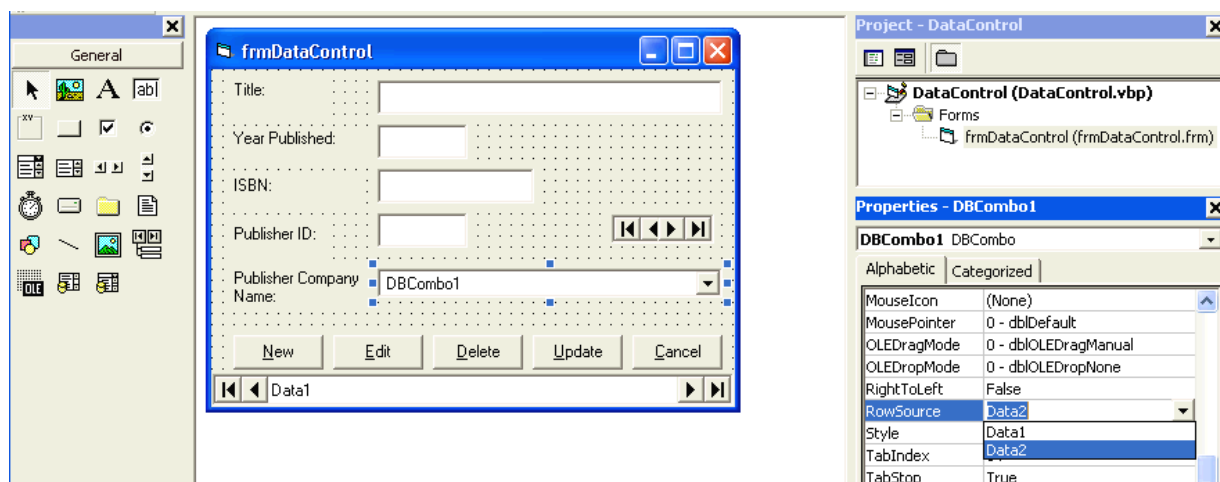
Sau khi Recordset gọi method Update thì Recordset ấy ra khỏi AddNew hay Edit modes. Ta cũng có thể tự thoát ra khỏi AddNew hay Edit modes, hay nói cho đúng hơn là hủy bỏ mọi pending (đang chờ đợi) Update bằng cách gọi **method CancelUpdate**, ví dụ như **Data1.Recordset.CancelUpdate**.

## 22.4. Dùng DataBound Combo

Trong chương trình hiện tại ta chỉ hiển thị lý lịch nhà xuất bản (PubID) của Title, chứ không có thêm chi tiết. Nếu chương trình lưu trữ PubID, nhưng hiển thị được Company Name của nhà xuất bản cho ta làm việc để khỏi phải nhớ các con số thì sẽ tốt hơn. Ta có thể thực hiện điều đó bằng cách dùng Control DBCombo (Data Bound Combo). Chúng ta hãy dùng IDE Menu Command Project | Components... để chọn Microsoft Data Bound List Controls 6.0 rồi click Apply.



Kế đó, thêm một DBCombo tên DBCombo1 vào Form. Vì ta cần một Recordset khác để cung cấp Table Publisher cho DBCombo1, nên chúng ta hãy thêm một control Data thứ nhì tên Data2 vào Form. Cho Data2, hãy set property DatabaseName thành E:\Program Files\Microsoft Visual Studio\VB98\BIBLIO.MDB và property RecordSource thành Publishers. Để không cho người ta thấy hình Data2 lúc run-time, chúng ta hãy set property Visible nó thành False.



Mục đích của chúng ta khi dùng DBCombo1 là hiển thị Company Name của nhà xuất bản, nhưng đằng sau lưng thì không có gì thay đổi, tức là ta vẫn làm việc với PubID cho các record Title của Data1. Khi người sử dụng click lên DBCombo1 để chọn một nhà xuất bản, thì ta theo Company Name đó mà chứa PubID tương ứng trong record Title của Data1. Do đó có nhiều thứ ta phải sắp đặt cho DBCombo1 như sau:

Property	Value	Chú thích
RowSource	Data2	Đây là datasource của chính DBCombo1. Nó cung cấp table Publishers.
Listfield	Company Name	Khi RowSource phía trên đã được chọn rồi, Combo của property Listfield này sẽ hiển thị các fields của table Publishers. Company Name là field của RowSource mà ta muốn hiển thị trên DBCombo1.
DataSource	Data1	Đây là datasource của bản ghi mà ta muốn. edit, tức là bản ghi của table Titles
Datafield	PubID	Field (của record Title) sẽ được thay đổi.
BoundColumn	PubID	Field trong RowSource (table Publishers) tương ứng với item user chọn trong DBCombo1 (Company Name).

Khi trong Edit mode user chọn một Company Name khác trong DBCombo1 rồi click nút Update chúng ta sẽ thấy Textbox txtPublisherID cũng đổi theo và hiển thị con số lý lịch PubID mới. Nếu trước khi Update chúng ta muốn thấy PubID mới hiển thị trong Textbox txtPublisherID thì chúng ta có thể dùng Event Click của DBCombo1 như sau:

```
Private Sub DBCombo1_Click(Area As Integer)
    ' Hiển thị new PuBID
    txtPublisherID.Text = DBCombo1.BoundText
End Sub
```

**Property BoundText** của DBCombo1 là trị số của BoundColumn mà ta có thể truy cập (viết hay đọc) được. Ví dụ như chúng ta muốn mỗi khi thêm một bản ghi Title mới thì default PubID là 324, tức là Company Name= "GLOBAL ENGINEERING". Chúng ta có thể assign trị số 324 vào property BoundText của DBCombo1 trong Sub cmdNew\_Click như sau:

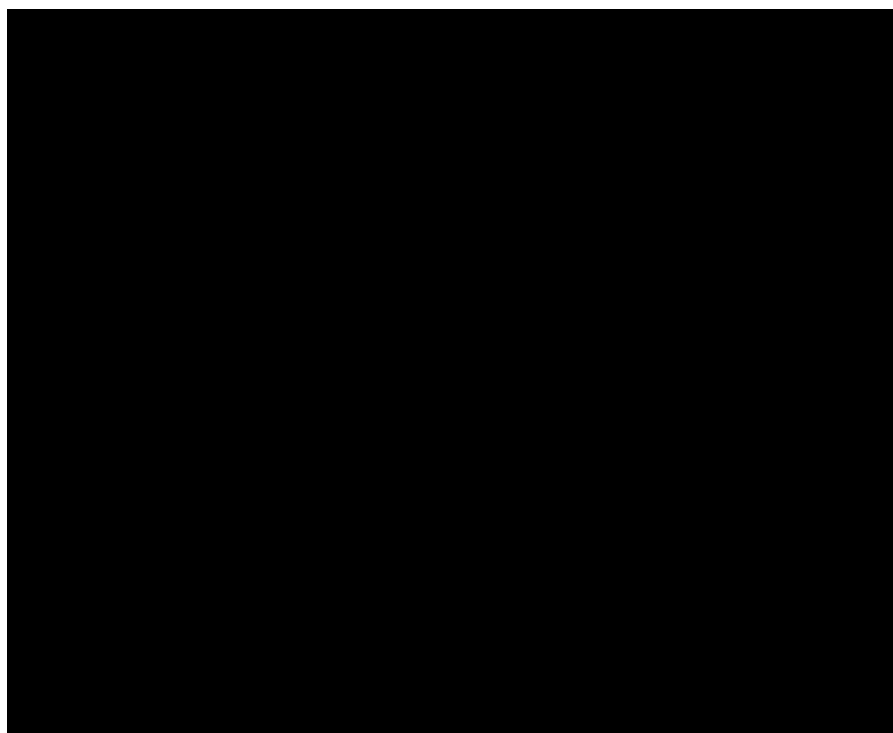
```
Private Sub cmdNew_Click()
    ' Place Recordset into Recordset AddNew mode
    Data1.Recordset.AddNew
    ' Default Publisher is "GLOBAL ENGINEERING", i.e. PubID=324
    DBCombo1.BoundText = 324
    ' Place controls in Edit Mode
    SetControls (True)
End Sub
```



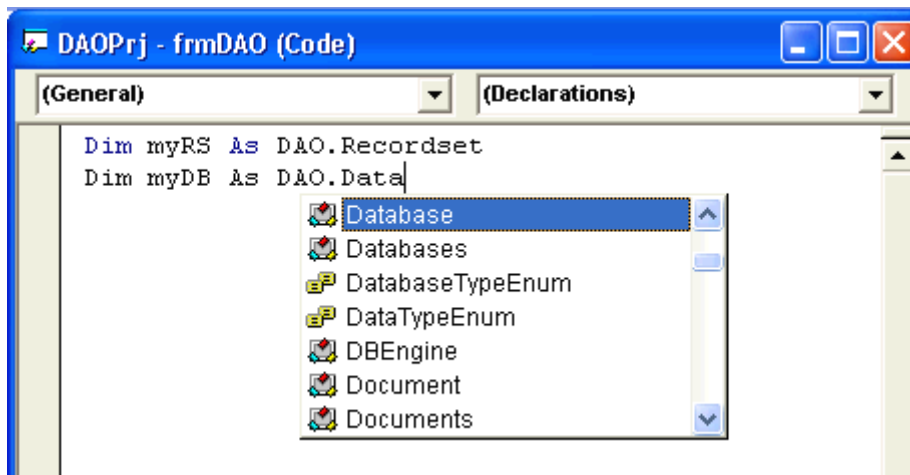
## BÀI 23. LẬP TRÌNH VỚI KỸ THUẬT DAO

### 23.1. Reference DAO

Trong bài này ta sẽ học những cách lập trình căn bản với MS Access database qua kỹ thuật DAO mà không cần dùng đến **Control Data** như trong bài trước. Ta sẽ cần đến vài Objects trong thư viện DAO, do đó nếu chúng ta mở một dự án VB6 mới thì hãy dùng Menu Command **Project | References...** để chọn **Microsoft DAO 3.51 Object Library** bằng cách click cái checkbox bên trái như trong hình dưới đây.



Sau đó trong code của Form chính ta sẽ declare variable **myDatabase** cho một instance của **DAO database** và variable **myRS** cho một **DAO recordset**. Ở đây ta nói rõ Database và Recordset là thuộc loại **DAO** để phân biệt với Database và Recordset thuộc loại **ADO (ActiveX Data Object)** sau này.



Bây giờ chúng ta hãy đặt lên Form chính, tên **frmDAO**, 4 labels với captions: **Title**, **Year Published**, **ISBN** và **Publisher ID**. Kế đó cho thêm 4 textboxes tương ứng và đặt tên chúng là **txtTitle**, **txtYearPublished**, **txtISBN** và **txtPublisherID**.

Điều ta muốn làm là khi Form mới được loaded, nó sẽ lấy về từ database một Recordset chứa tất cả records trong **table Titles** theo thứ tự về mẫu tự (alphabetical order) của **field Title** và hiển thị bản ghi đầu tiên.

### 23.2. Dùng keyword SET

Chuyện trước hết là mở một Database Object dựa vào tên đầy đủ (full path name) của Access database:

```
' Open main database
Set myDB = OpenDatabase(AppFolder & "BIBLIO.MDB")
```

Đề ý chữ **Set** trong câu code trên. Đó là vì **myDB** là một **Pointer** đến một Object. Mặc dầu từ đây về sau ta sẽ dùng **myDB** như một Database theo cách giống như bất cứ variable thuộc data type nào khác, nhưng khi chỉ định lần đầu là nó từ đâu đến thì ta dùng chữ **Set**, để nói rằng thật ra **myDB** không phải là Object Database, nhưng là Pointer đến Object Database.

Mục đích là VB6 runtime dynamically allocates (dành ra cho khi cần) một phần trong bộ nhớ (memory) để chứa Object Database khi ta nhận được nó từ execution của **Method OpenDatabase**. Dầu vị trí chỗ chứa Object Database trong bộ nhớ không nhất định, nhưng vì ta nắm con trỏ chỉ đến vị trí ấy nên ta vẫn có thể làm việc với nó một cách bình thường. Con trỏ đó là value (trị số) của variable **myDB**. Vì value này không phải là Object, nhưng nó chứa **memory address** chỉ đến (**point to** hay **refer to**) Object Database, nên ta gọi nó là Pointer.

Lập trình dùng Pointer nói chung rất linh động mang lại hiệu quả cao trong các ngôn ngữ như C, Pascal, C++ ,v.v.. Tuy nhiên, lập trình viên phải nhớ trả lại Operating System phần memory mình dùng khi không còn cần nó nữa để Operating System lại allocate cho Object khác. Nếu công việc quản lý dùng lại memory không ổn thỏa thì có những mảnh memory nằm rải rác mà Operating Sytem không biết. Dần dần Operating System sẽ không còn memory dư nữa. Ta gọi hiện tượng ấy là **memory leakage (rỉ)**. Các ngôn ngữ sau này như Java, C# đều không dùng Pointer nữa. Visual Basic không muốn lập trình viên dùng Pointer. Chỉ trong vài trường hợp đặc biệt VB6 mới lộ ra cho ta thấy thật ra ở trong hậu trường VB6 Runtime dùng Pointer, như trong trường hợp này.

Tương tự như vậy, vì Recordset là một Pointer đến một Object, ta cũng dùng **Set** khi chỉ định một DAO Recordset lấy về từ **Method OpenRecordset** của database myDB.

```
'Open recordset
Set myRS=myDB.OpenRecordset("Select*from Titles ORDER BY Title")
```

Cái parameter loại String ta dùng cho method OpenRecordset là một **Lệnh (Statement) SQL**. Nó chỉ định cho database lấy tất cả mọi fields (columns) (**Select \***) của mỗi bản ghi từ Table Titles (**from Titles**) làm một Recordset và sort các records trong Recordset ấy theo alphabetical order của field Title (**ORDER BY Title**).

Nhớ là Recordset này cũng giống như **property Recordset** của một Control Data mà ta dùng trong bài trước. Bây giờ có Recordset rồi, ta có thể hiển thị chi tiết của bản ghi đầu tiên nếu Recordset ấy có ít nhất một bản ghi. Ta kiểm tra điều ấy dựa vào **property RecordCount** của Recordset như trong code dưới đây:

```
Private Sub Form_Load()
    ' Fetch Folder where this program EXE resides
    AppFolder = App.Path
    ' make sure it ends with a back slash
    If Right(AppFolder, 1) <> "\" Then AppFolder = AppFolder & "\"
    ' Open main database
    Set myDB = OpenDatabase(AppFolder & "BIBLIO.MDB")
    'Open recordset
    Set myRS=myDB.OpenRecordset("Select * from Titles ORDER BY
Title")
    ' if Recordset is not empty then hiển thị the first record
    If myRS.RecordCount > 0 Then
        myRS.MoveFirst ' move to first record
    End If
End Sub
```

```
        Hiển thịrecord ' hiển thị details of current record
    End If
End Sub
```

Sau khi dùng **method MoveFirst** của Recordset để định vị con trỏ hiện tại ở bản ghi đầu tiên, ta hiển thị trị số các fields của bản ghi bằng cách assign chúng vào các textboxes của Form như sau:

```
Private Sub Hiển thịrecord()
    ' Assign record fields to the appropriate textboxes
    With myRS
        ' Assign field Title to textbox txtTitle
        txtTitle.Text = .Fields("Title")
        txtYearPublished.Text = .Fields("[Year Published] ")
        txtISBN.Text = .Fields("ISBN")
        txtPublisherID.Text = .Fields("PubID")
    End With
End Sub
```

Để ý vì field **Year Published** gồm có hai chữ nên ta phải đặt tên của field ấy giữa hai dấu ngoặc vuông (||). Để tránh bị phiền phức như trong trường hợp này, khi chúng ta đặt tên database field trong lúc thiết kế một table hãy dán dính các chữ lại với nhau, đừng để rời ra. Ví dụ như dùng **YearPublished** thay vì **Year Published**.

### 23.3. Các nút di chuyển

Muốn có các nút Navigators tương đương với của một Control Data, chúng ta hãy đặt lên Form 4 buttons mang tên **CmdFirst**, **CmdPrevious**, **CmdNext** và **CmdLast** với captions: <<, <, >, >>.

Code cho các nút này cũng đơn giản, nhưng ta phải coi chừng khi người sử dụng muốn di chuyển quá bản ghi cuối cùng hay bản ghi đầu tiên. Ta phải kiểm tra xem **EOF** có trở thành True khi người sử dụng click CmdNext, hay **BOF** có trở thành True khi người sử dụng click CmdPrevious:

```
Private Sub CmdNext_Click()
    myRS.MoveNext ' Move to next record
    ' Display record details if has not gone past the last record
    If Not myRS.EOF Then
```

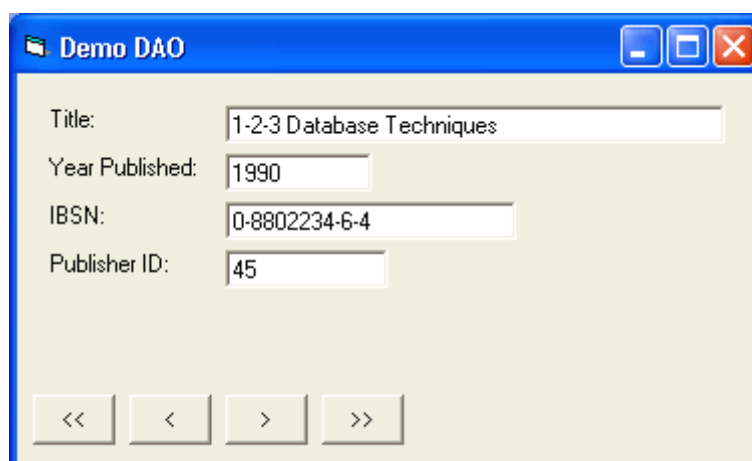
```
        Displayrecord ' hiển thị details of current record
    Else
        myRS.MoveLast ' Move back to last record
    End If
End Sub

Private Sub CmdPrevious_Click()
    myRS.MovePrevious ' Move to previous record
    ' Display record details if has not gone past the first record
    If Not myRS.BOF Then
        Displayrecord ' hiển thị details of current record
    Else
        myRS.MoveFirst ' Move back to first record
    End If
End Sub

Private Sub CmdFirst_Click()
    myRS.MoveFirst ' Move back to first record
    Displayrecord ' hiển thị details of current record
End Sub

Private Sub CmdLast_Click()
    myRS.MoveLast ' Move back to last record
    Displayrecord ' hiển thị details of current record
End Sub
```

Khi chạy chương trình chúng ta sẽ thấy nó hiển thị chi tiết của Bản ghi đầu tiên khác với trong bài trước đây vì các records đã được sorted:



The screenshot shows a window titled "Demo DAO" with a light beige background. It contains four text input fields arranged vertically, each with a label to its left: "Title:" with the value "1-2-3 Database Techniques", "Year Published:" with "1990", "ISBN:" with "0-8802234-6-4", and "Publisher ID:" with "45". At the bottom of the window, there are four buttons with navigation symbols: "<<", "<", ">", and ">>".

## 23.4. Thêm bớt các Records

Giống như chương trình trong bài rồi, ta sẽ thêm phương tiện để thêm (add), bớt (delete) các bản ghi. Bây giờ chúng ta hãy để vào Form 5 buttons tên: **cmdEdit**, **cmdNew**, **cmdDelete**, **cmdUpdate** và **cmdCancel**.

Chỗ nào trong chương trình trước ta dùng **Data1.Recordset** thì bây giờ ta dùng **myRS**.

Ta sẽ dùng lại **Sub SetControls** với parameter **Editing** có trị số False hay True tùy theo người sử dụng đang Browse hay Edit. Trong **Browse mode**, các Textboxes bị Locked (khóa) và các nút **cmdUpdate** và **cmdCancel** trở nên bất lực. Trong **Edit mode**, các Textboxes được unlocked (mở khóa) và các nút **cmdNew**, **cmdDelete** và **cmdEdit** trở nên bất lực.

Vì ở đây không có Data Binding nên đợi cho đến khi **Update** ta mới đặt Recordset vào **AddNew** hay **Edit mode**. Do đó ta chỉ cần nhớ là khi người sử dụng edits là đang sửa đổi một bản ghi hiện hữu hay thêm một bản ghi mới. Ta chứa trị số Boolean ấy trong variable **AddNewRecord**. Nếu người sử dụng sắp thêm một bản ghi mới thì **AddNewRecord = True**, nếu người sử dụng sắp Edit một bản ghi hiện hữu thì **AddNewRecord = False**.

Ngoài ra, khi người sử dụng sắp thêm một bản ghi mới bằng cách click nút New thì ta phải tự clear (làm trắng) hết các textboxes bằng cách assign Empty string vào text property của chúng như sau:

```
' If Editing existing record then AddNewRecord = False
' Else AddNewRecord = true
Dim AddNewRecord As Boolean

Private Sub ClearAllFields()
    ' Clear all the textboxes
    txtTitle.Text = ""
    txtYearPublished.Text = ""
    txtISBN.Text = ""
    txtPublisherID.Text = ""
End Sub

Private Sub cmdNew_Click()
    ' Remember that this is Adding a new record
    AddNewRecord = True
```

```
' Clear all textboxes
ClearAllFields
' Place controls in Edit Mode
SetControls (True)
End Sub
Private Sub CmdEdit_Click()
' Place controls in Edit Mode
SetControls (True)
' Remember that this is Editing an existing record
AddNewRecord = False
End Sub
```

Nếu người sử dụng clicks Cancel trong khi đang edit các textboxes, ta không cần gọi **method CancelUpdate** vì Recordset chưa bị đặt vào AddNew hay Edit mode. Ở đây ta chỉ cần hiển thị lại chi tiết của current record, tức là hủy bỏ những gì người sử dụng đang đánh vào:

```
Private Sub CmdCancel_Click()
' Cancel update
SetControls (False)
' Redisplay details or current record
Displayrecord
End Sub
```

Lúc người sử dụng clicks Update, chúng ta có dịp để kiểm tra data xem có field nào bị bỏ trống (nhất là **Primary Key ISBN** bắt buộc phải có trị số) hay có gì không valid bằng cách gọi **Function GoodData**. Nếu GoodData trả lại một trị số False thì ta không xúc tiến với việc Update. Nếu GoodData trả về trị số True thì ta đặt Recordset vào AddNew hay Edit mode tùy theo trị số của Boolean variable AddNewRecord.

Giống như khi hiển thị chi tiết của một bản ghi ta phải assign từng Field vào textbox, thì bây giờ khi Update ta phải làm ngược lại, tức là assign property Text của từng textbox vào Record Field tương ứng. Sau cùng ta gọi **method Update** của recordset và cho các controls trở lại Browse mode:

```
Private Function GoodData() As Boolean
' Check Data here. If Invalid Data then GoodData = False
GoodData = True
End Function
Private Sub CmdUpdate_Click()
```

```
' Verify all data, if Bad then do not Update
If Not GoodData Then Exit Sub
' Assign record fields to the appropriate textboxes
With myRS
    If AddNewRecord Then
        .AddNew ' Place Recordset in AddNew Mode
    Else
        .Edit ' Place Recordset in Edit Mode
    End If
    ' Assign text of txtTitle to field Title
    .Fields("Title") = txtTitle.Text
    .Fields("[Year Published]") = txtYearPublished.Text
    .Fields("ISBN") = txtISBN.Text
    .Fields("PubID") = txtPublisherID.Text
    ' Update data
    .Update
End With
' Return controls to Browse Mode
SetControls (False)
End Sub
```

Cũng vì không có Data Binding, nên khi người sử dụng xóa một bản ghi, sau khi di chuyển qua bản ghi kế tiếp ta phải tự hiển thị chi tiết của bản ghi đó như sau:

```
Private Sub CmdDelete_Click()
    On Error GoTo DeleteErr
    With myRS
        .Delete ' Delete new record
        .MoveNext ' Move to next record
        If .EOF Then .MoveLast
        Displayrecord ' Display details of current record
    End With
    Exit Sub
DeleteErr:
    MsgBox Err.Description
    Exit Sub
End Sub
```

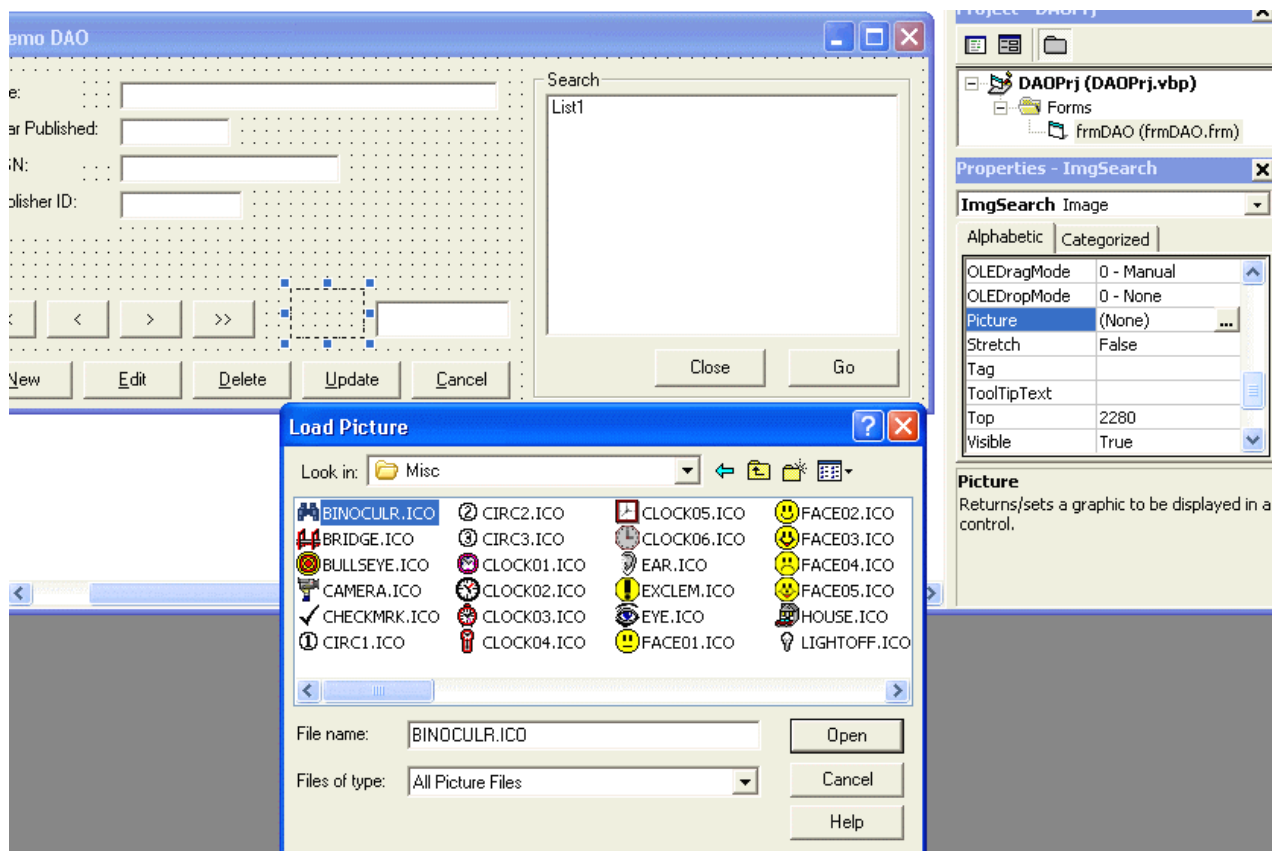


## 23.5. Tìm một bản ghi

Tiếp theo đây, ta muốn liệt kê các sách có tiêu đề chứa một chữ hay câu nào đó, ví dụ như chữ "Guide". Kế đó người sử dụng có thể chọn một sách bằng cách chọn tiêu đề sách ấy và click nút Go. Chương trình sẽ locate (tìm ra) bản ghi của sách ấy và hiển thị chi tiết của nó.

Bây giờ chúng ta hãy cho vào Form một textbox tên txtSearch và một Image tên ImgSearch. Kế đó đặt một frame tên fraSearch vào Form. Để lên frame này một listbox tên List1 để hiển thị tiêu đề các sách, và hai buttons tên CmdClose và CmdGo, với caption Close và Go. Sau khi select một sách trong List1, người sử dụng sẽ click nút Go để hiển thị chi tiết sách ấy. Nếu đổi ý, người sử dụng sẽ click nút Close để làm biến mất frame fraSearch.

Bình thường frame fraSearch chỉ hiện ra khi cần, nên lúc đầu hãy set property Visible của nó thành False. Ta sẽ cho ImgSearch hiển thị hình một ống dòm nên chúng ta hãy click vào bên phải property Picture trong Properties Window để chọn Icon BINOCULR.ICO từ folder E:\Program Files\Microsoft Visual Studio\Common\Graphics\Icons\Misc:

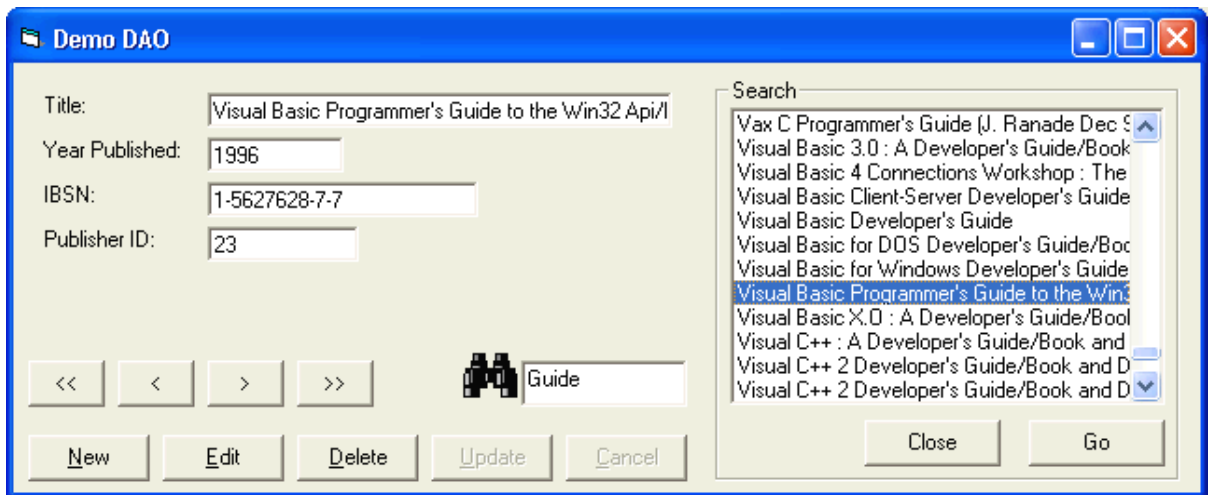


Cái Primary Key của table Titles là ISBN. Khi người sử dụng select một sách ta muốn biết ISBN của sách ấy để locate (định chỗ) nó trong Recordset myRS. Do đó trong khi thêm tiêu đề của một sách vào List1, ta đồng thời thêm ISBN của sách ấy vào một Listbox thứ hai tên

List2. Ta chỉ sẽ dùng List2 sau hậu trường, nên hãy set property Visible của nó thành False. Dưới đây là code để load tiêu đề sách và ISBN vào các Listboxes:

```
Private Sub ImgSearch_Click()  
    ' Show Search Frame  
    fraSearch.Visible = True  
    Dim SrchRS As DAO.Recordset  
    Dim SQLCommand As String  
    ' Define SQL statement  
    SQLCommand = "Select * from Titles where Title LIKE '" & "*" &  
txtSearch & "*" & "' ORDER BY Title"  
    ' Fetch all records having Title containing the text pattern  
given by txtSearch  
    Set SrchRS = myDB.OpenRecordset(SQLCommand)  
    ' If Recordset is not Empty then list the books' titles in  
List1  
    If SrchRS.RecordCount > 0 Then  
        List1.Clear ' Clear List1  
        ' We use List2 to contain the Primary Key ISBN  
corresponding to the books in List1  
        List2.Clear ' Clear List2  
        With SrchRS  
            ' Iterate through the Recordset until EOF  
            Do While Not SrchRS.EOF  
                ' Hiện thị Title in List1  
                List1.AddItem .Fields("Title")  
                ' Store corresponding ISBN in List2  
                List2.AddItem .Fields("ISBN")  
                .MoveNext ' Move to next record in the Recordset  
            Loop  
        End With  
    End If  
End Sub
```

Khi người sử dụng Click ImgSearch với text pattern là chữ Guide, ta sẽ thấy hình dưới đây:



Trong SELECT statement bên trên ta dùng operator **LIKE** trên text pattern, chữ **Guide**, có **wildcard character (\*)** ở hai bên. Wildcard character là chỗ có (hay không có) chữ gì cũng được. Trong trường hợp này có nghĩa là hễ có chữ Guide trong tiêu đề sách là được, không cần biết nó nằm ở đâu. Ngoài ra sự chọn lựa này **Không có Case Sensitive**, tức là chữ **guide**, **Guide** hay **GUIDE** đều được cả.

Khi người sử dụng clicks nút Go, ta sẽ dùng **method FindFirst** của Recordset myRS để định chỗ của bản ghi có trị số Primary Key là dòng text trong List2 tương ứng với tiêu đề được chọn trong List1 như sau:

```
Private Sub CmdGo_Click()  
    Dim SelectedISBN As String  
    Dim SelectedIndex As Integer  
    Dim Criteria As String  
    ' Index of line selected by user in List1  
    SelectedIndex = List1.ListIndex  
    ' Obtain corresponding ISBN in List2  
    SelectedISBN = List2.List(SelectedIndex)  
    ' Define Search criteria - use single quotes for selected text  
    Criteria = "ISBN = '" & SelectedISBN & "'"   
    ' Locate the record, it will become the current record  
    myRS.FindFirst Criteria  
    ' Hiện thị details of current record  
    Hiện thịrecord  
    ' Make fraSearch disappeared  
    fraSearch.Visible = False  
End Sub
```

Lưu ý là trong string Criteria, vì ISBN thuộc loại text, chứ không phải là một con số, nên ta phải kẹp nó giữa hai dấu ngoặc đơn.

### 23.6. Bookmark

Khi di chuyển từ bản ghi này đến bản ghi khác trong Recordset, đôi khi ta muốn đánh dấu vị trí của một bản ghi để có dịp sẽ trở lại. Ta có thể thực hiện điều ấy bằng cách ghi nhớ **Bookmark** của Recordset.

Ví dụ khi người sử dụng clicks nút Go, ta muốn nhớ vị trí của bản ghi lúc ấy để sau này quay trở lại khi người sử dụng clicks nút **Go Back**. Chúng ta hãy thêm vào Form một button tên **CmdGoBack** với Caption **Go Back**. Ta sẽ thêm một variable tên **LastBookmark** loại data type **Variant**:

```
Dim LastBookMark As Variant
```

Lúc đầu button CmdGoBack invisible, và chỉ trở nên visible sau khi người sử dụng clicks nút Go. Ta thêm các dòng codes sau vào Sub CmdGo\_Click() như sau:

```
' Remember location of current record
LastBookMark = myRS.BookMark
CmdGoback.Visible = True
```

Dưới đây là code để quay trở lại vị trí current record trước đây trong Recordset:

```
Private Sub CmdGoback_Click()
    ' Reposition record to last position
    myRS.BookMark = LastBookMark
    ' Rehiển thị details or current record
    Displayrecord
End Sub
```

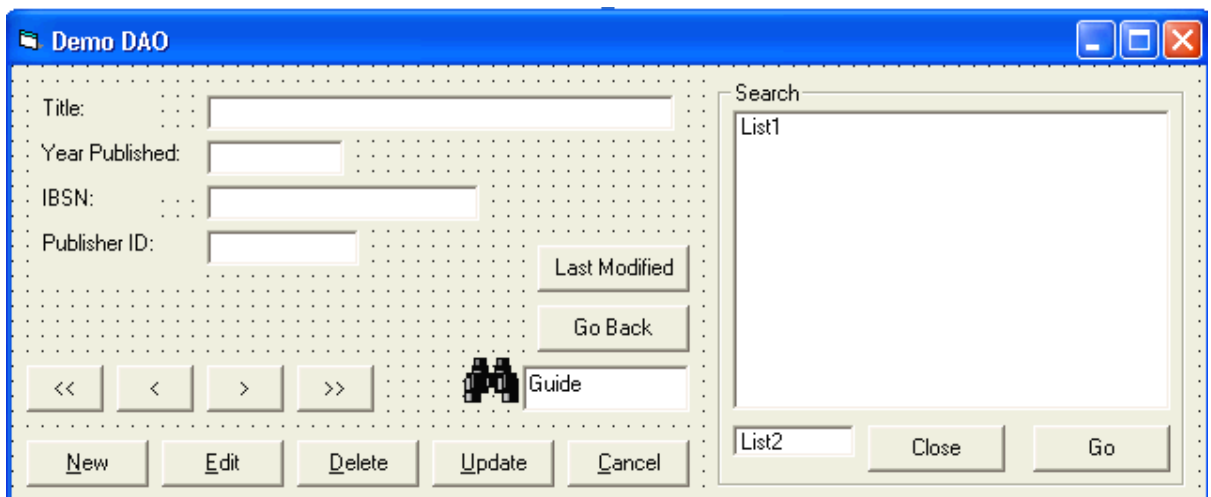
### 23.7. LastModified

**LastModified** là vị trí của bản ghi vừa mới được sửa đổi hay thêm vào trong Recordset. Để thử điều này chúng ta hãy thêm một button invisible tên **CmdLastModified** với caption là **Last Modified**. Button này chỉ hiện ra sau khi người sử dụng clicks Update.

Bất cứ lúc nào chúng ta Click nút CmdLastModified, bản ghi mới vừa được sửa đổi hay thêm vào sẽ hiển thị:

```
Private Sub CmdLastModified_Click()  
    ' Reposition record to last position  
    myRS.BookMark = myRS.LastModified  
    ' Redisplay details or current record  
    Displayrecord  
End Sub
```

Dưới đây là hình của Form lúc đang được thiết kế:

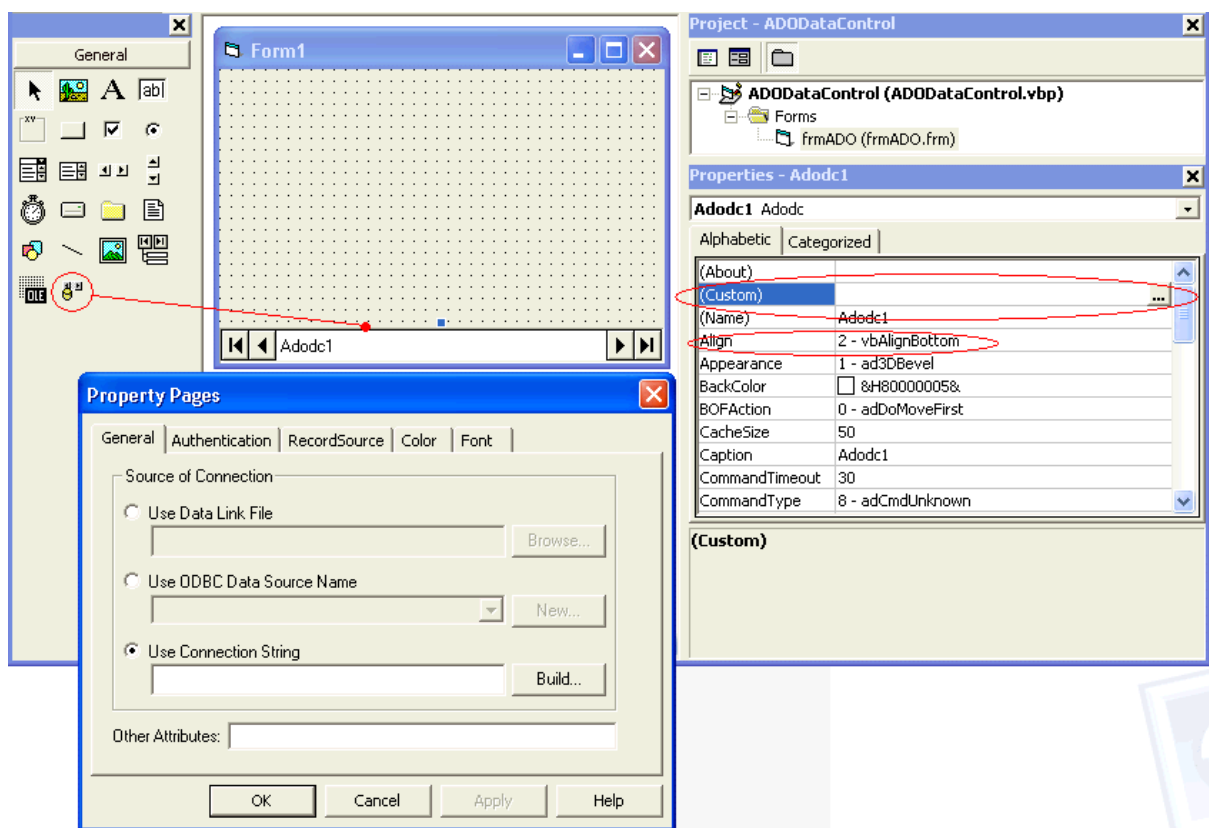




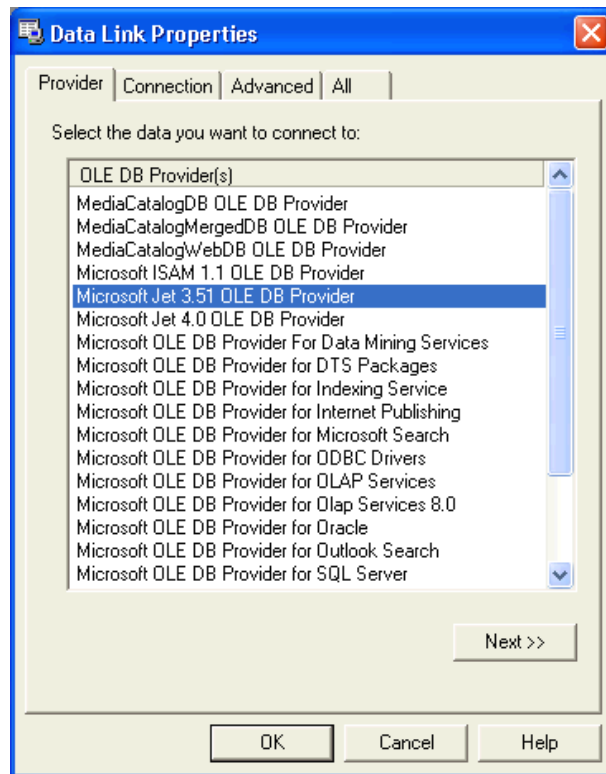
Chúng ta hãy bắt đầu một dự án VB6 mới, cho nó tên ADODataControl bằng cách click tên project trong Project Explorer bên phải rồi edit property Name trong Properties Window. Sửa tên của form chính thành frmADO, và đánh câu ADO DataControl Demo vào Caption của nó.

DoubleClick lên Icon của Control Data ADO trong Toolbox. Một Control Data ADO tên Adodc1 sẽ hiện ra trên Form. Muốn cho nó nằm bên dưới Form, giống như một StatusBar, hãy set property Align của nó trong Properties Window thành 2 - vbAlignBottom.

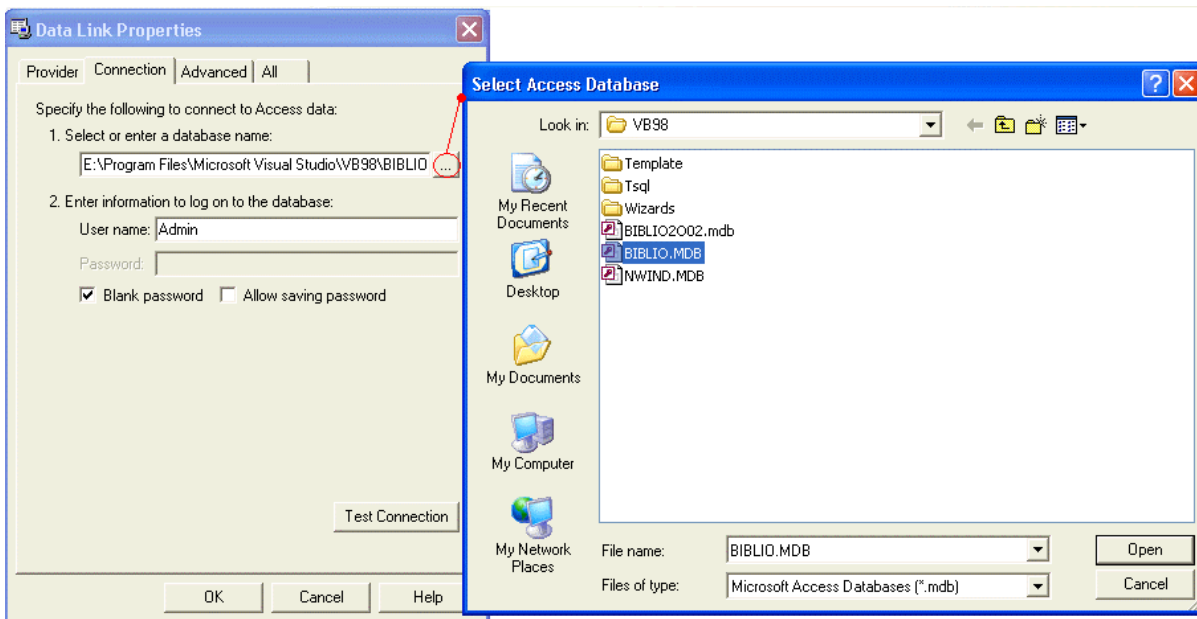
Click bên phải dòng property (Custom), kế đó click lên nút browse có ba chấm để giao thoại Property Pages hiện ra. Trong giao thoại này, trên Tab General chọn Radio (Option) Button Use Connection String rồi click nút Build....



Trong giao thoại Data Link Properties, Tab Provider, chọn Microsoft Jet 3.51 OLE DB Provider, rồi click nút Next >> hay Tab Connection.



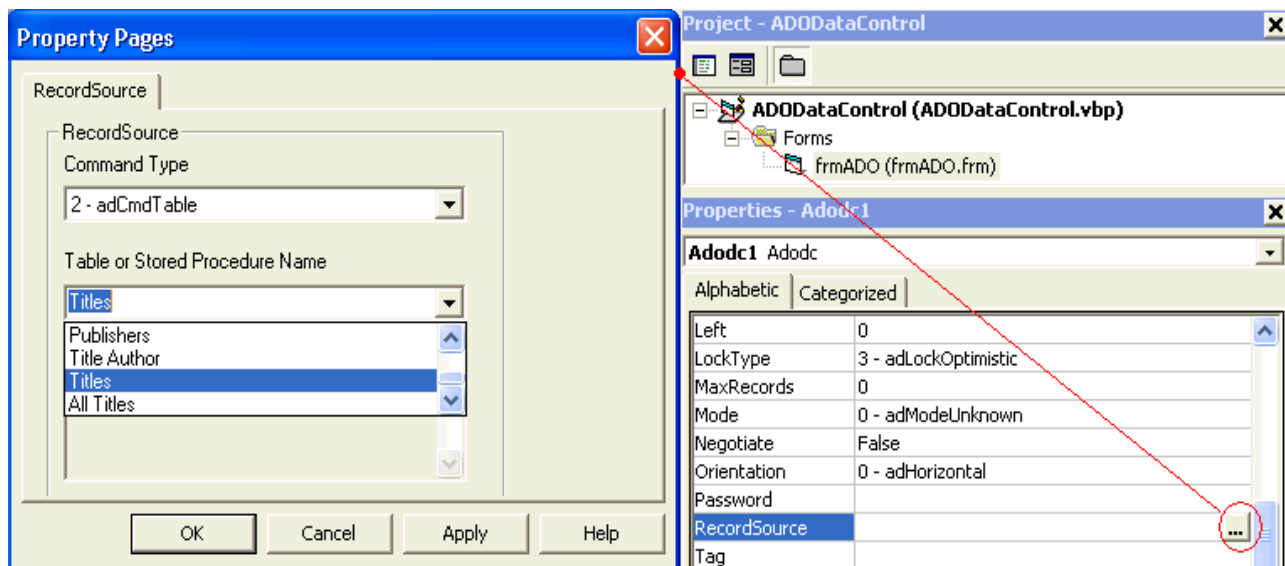
Ở chỗ Select or enter a database name ta chọn E:\Program Files\Microsoft Visual Studio\VB98\BIBLIO.MDB, trong computer của chúng ta có thể file ấy nằm trên disk C hay D. Sau đó, chúng ta có thể click nút Test Connection phía dưới để thử xem connection có được thiết lập tốt không.



Lập connection xong rồi, ta chỉ định muốn lấy gì về làm Recordset bằng cách click property **Recordsource** của Adodc1. Trong giao diện Property Pages của nó chọn 2-



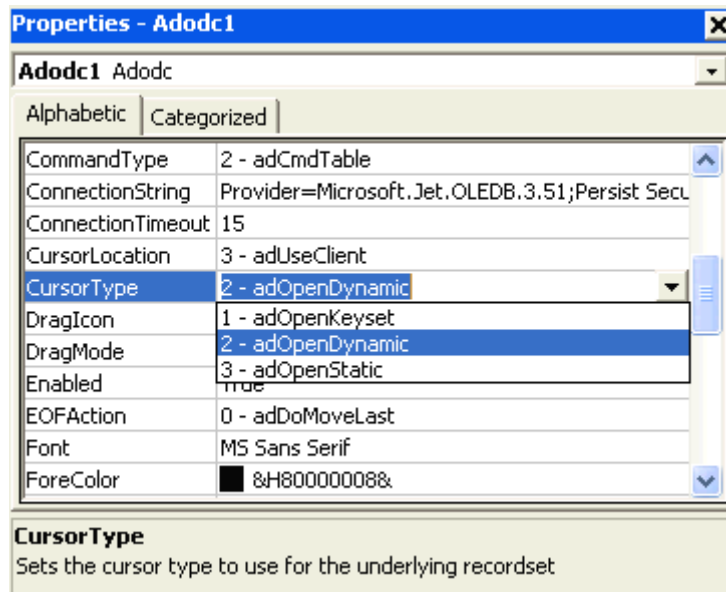
**adCmdTable** làm **Command Type**, kể đó mở Combo box cho **Table or Stored Procedure Name** để chọn **table Titles**.



Tùy theo cách ta dùng Recordset trong ADO, nó có ba loại và được gọi là **Cursor Type**. Cursor chẳng qua là một tên khác của Recordset:

- **Static Cursor:** Static Cursor cho chúng ta một static copy (bản sao cứng ngắt) của các records. Trong lúc chúng ta dùng Static Cursor, nếu có ai khác sửa đổi hay thêm, bớt gì vào recordset chúng ta sẽ không thấy.
- **Keyset Cursor:** Keyset Cursor hơn Static Cursor ở chỗ trong lúc chúng ta dùng nó, nếu có ai sửa đổi bản ghi nào chúng ta sẽ biết. Nếu ai xóa bản ghi nào, chúng ta sẽ không thấy nó nữa. Tuy nhiên chúng ta sẽ không biết nếu có ai thêm một bản ghi nào vào recordset.
- **Dynamic Cursor:** Như chữ **sống động (dynamic)** hàm ý, trong lúc chúng ta đang dùng một Dynamic Cursor, nếu có ai khác sửa đổi hay thêm, bớt gì vào recordset chúng ta sẽ thấy hết.

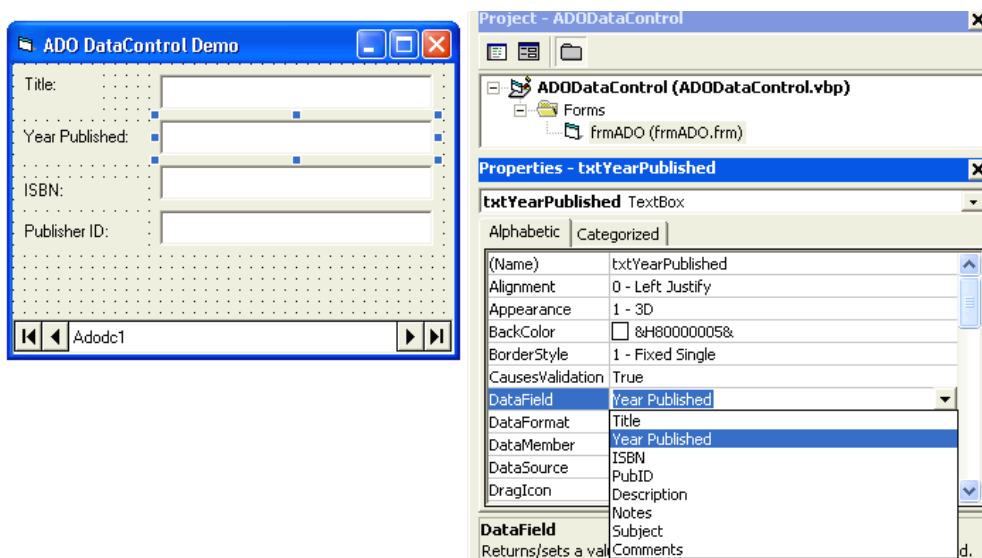
Chúng ta hãy chọn trị số **2-adOpenDynamic** cho property **Cursor Type** của **Adodc1**:



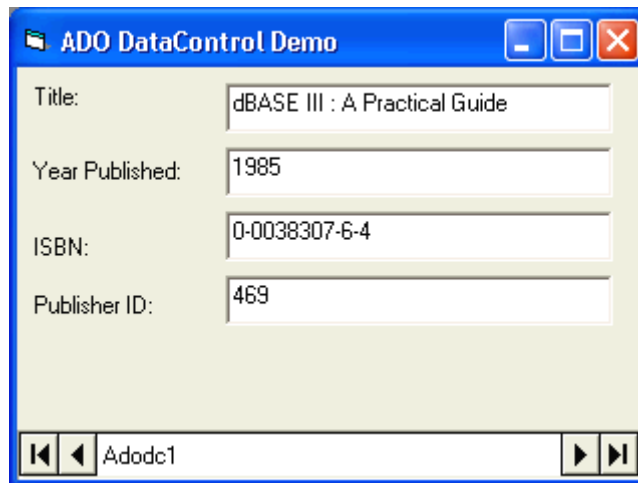
Bây giờ chúng ta hãy đặt lên Form 4 labels với captions: **Title**, **Year Published**, **ISBN** và **Publisher ID**. Kế đó cho thêm 4 textboxes tương ứng và đặt tên chúng là **txtTitle**, **txtYearPublished**, **txtISBN** và **txtPublisherID**.

Để thực hiện Data Binding, chúng ta hãy chọn textbox **txtYearPublished** (năm xuất bản), rồi set **property Datasource** của nó trong Properties Window thành **Adodc1**. Khi click lên **property DataField** của txtYearPublished và mở ComboBox ra chúng ta sẽ thấy liệt kê tên các Fields trong table Titles. Đó là vì Adodc1 được coi như trung gian lấy table Titles từ database. Ở đây ta sẽ chọn cột Year Published.

Lập lại công tác này cho 3 textboxes kia, và chọn các cột Title (Tiêu đề), ISBN (số lý lịch trong thư viện quốc tế), và PubID (số lý lịch nhà xuất bản) làm DataField cho chúng.



Đến đây, mặc dầu chưa viết một dòng code nào, chúng ta có thể chạy chương trình và nó sẽ hiển thị như dưới đây:

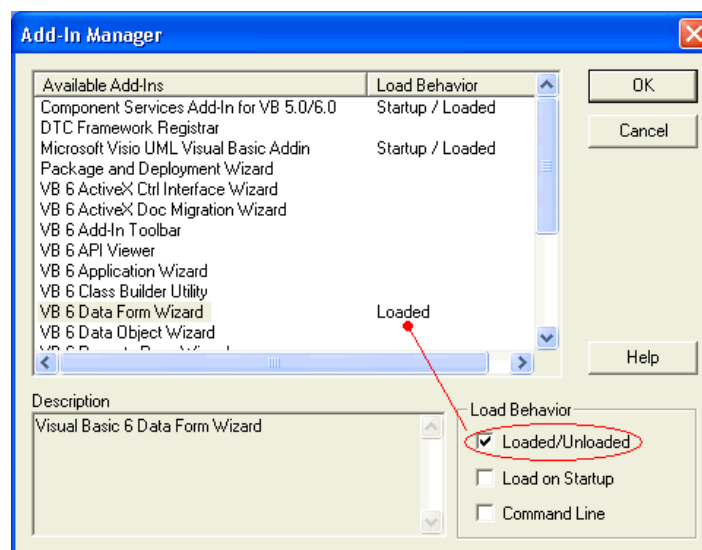


### 24.2. Data Form Wizard

Để giúp lập trình viên thiết kế các data forms nhanh hơn, VB6 cho ta **Data Form Wizard** để generate (phát sinh) ra một form có hỗ trợ Edit, Add và Delete records.

Bây giờ chúng ta hãy khởi động một standard project VB6 mới, tên **ADOCClass** và copy MS Access file **BIBLIO.MDB**, tức là database, vào trong cùng folder của dự án mới này.

Muốn dùng Data Form Wizard, trước hết ta phải thêm nó vào môi trường phát triển (IDE) của VB6. Chúng ta hãy dùng IDE Menu Command **Add-Ins | Add-In Manager...**. Chọn **VB6 Data Form Wizard** trong giao thoại, rồi click Checkbox **Loaded/Unloaded** để chữ **Loaded** hiện bên phải dòng "VB6 Data Form Wizard" như trong hình dưới đây:

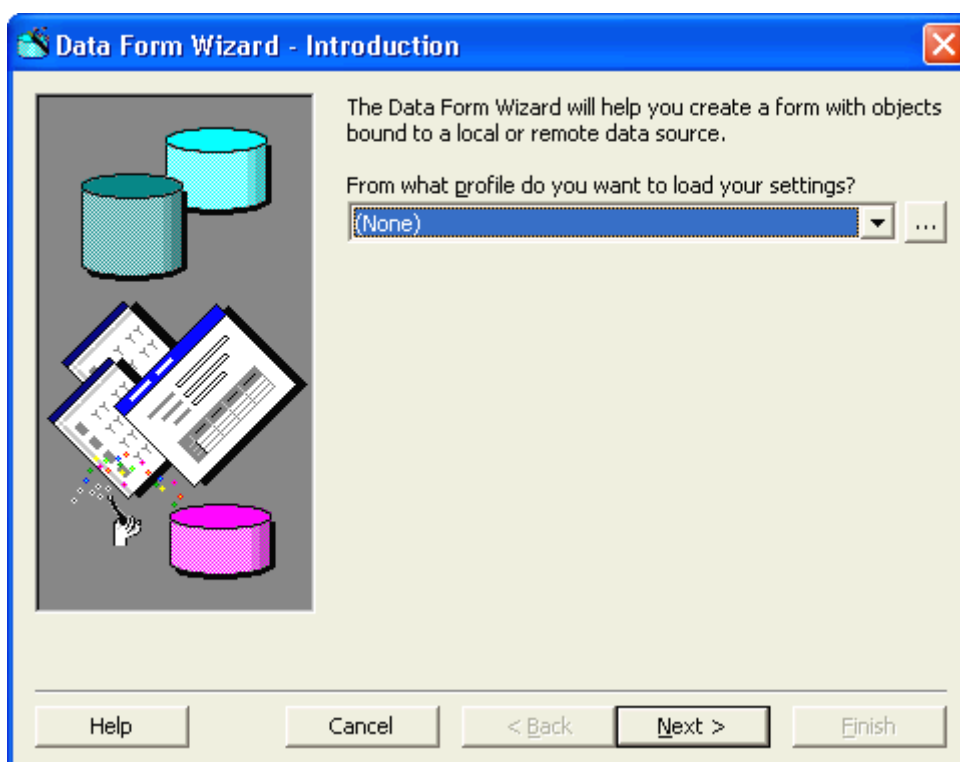


Nếu chúng ta muốn mỗi lần khởi động VB6 IDE là có sẵn Data Form Wizard trong menu Add-Ins thì ngoài option Loaded, chúng ta click thêm check box **Load on Startup**.

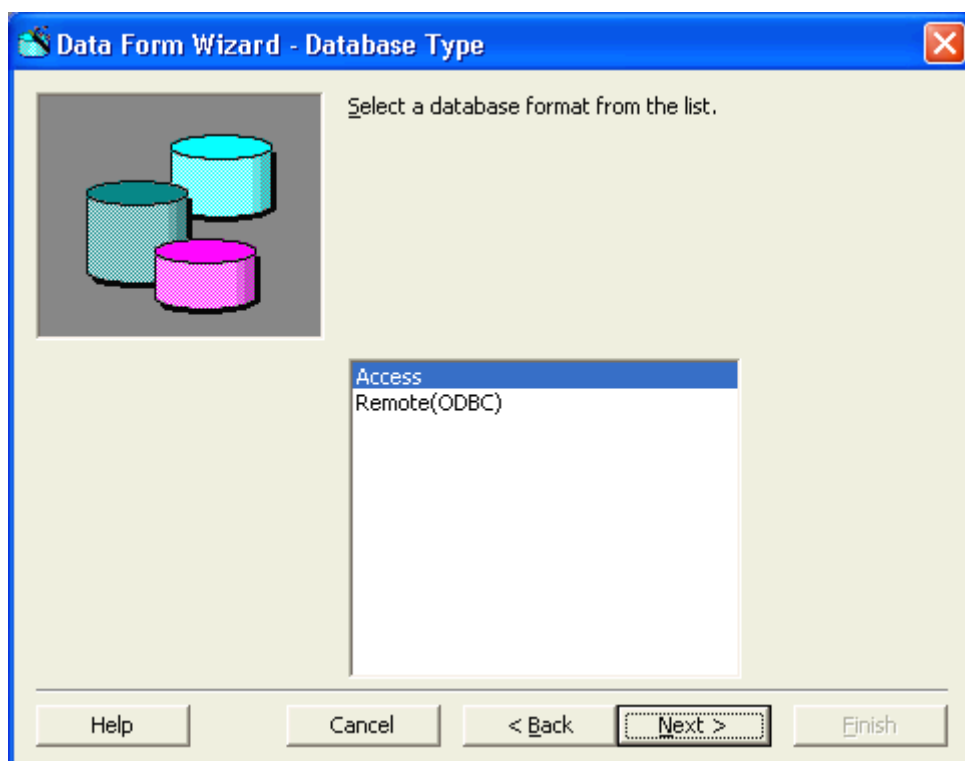
Một **Add-In** là một menu Item mới mà ta có thể thêm vào một chương trình ứng dụng có sẵn. Thường thường, người ta dùng Add-Ins để thêm chức năng cho một chương trình, làm như là chương trình đã có sẵn chức năng ấy từ đầu. Chúng ta hãy khởi động Data Form Wizard từ IDE Menu Command mới **Add-Ins | Data Form Wizard...**



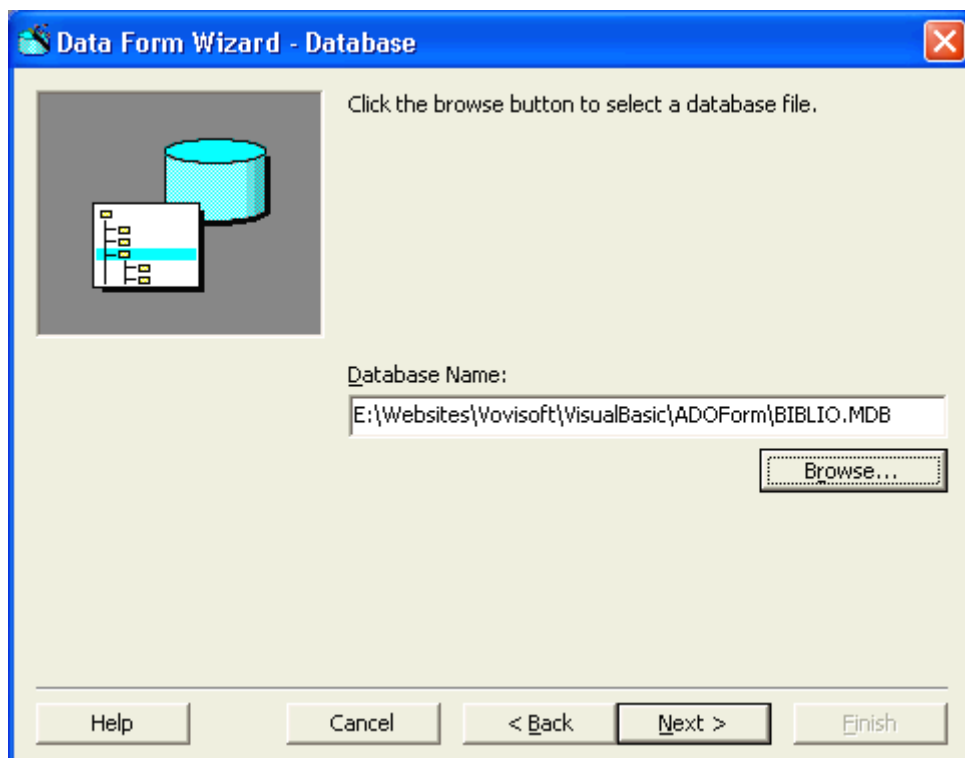
Khi trang Data Form Wizard - Introduction hiện ra, click Next



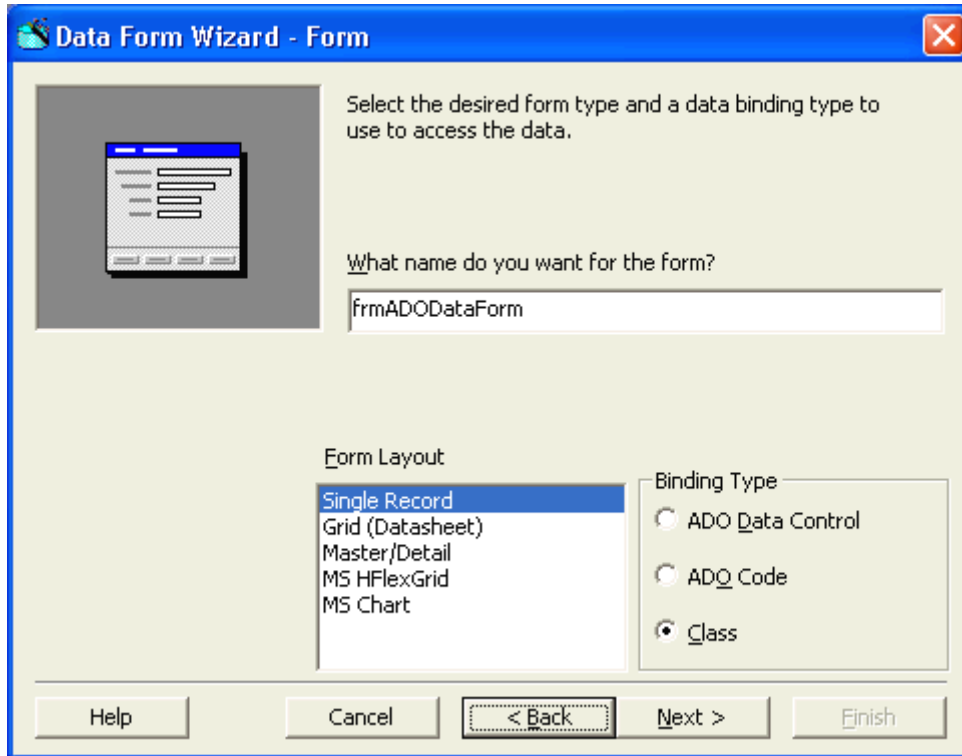
Trong trang kế đó chọn **Access** làm **Database Type**.



Trong trang Database, click **Browse** để chọn một MS Access database file. Ở đây ta chọn file **BIBLIO.MDB** từ chính folder của chương trình này. Đoạn click **Next**.



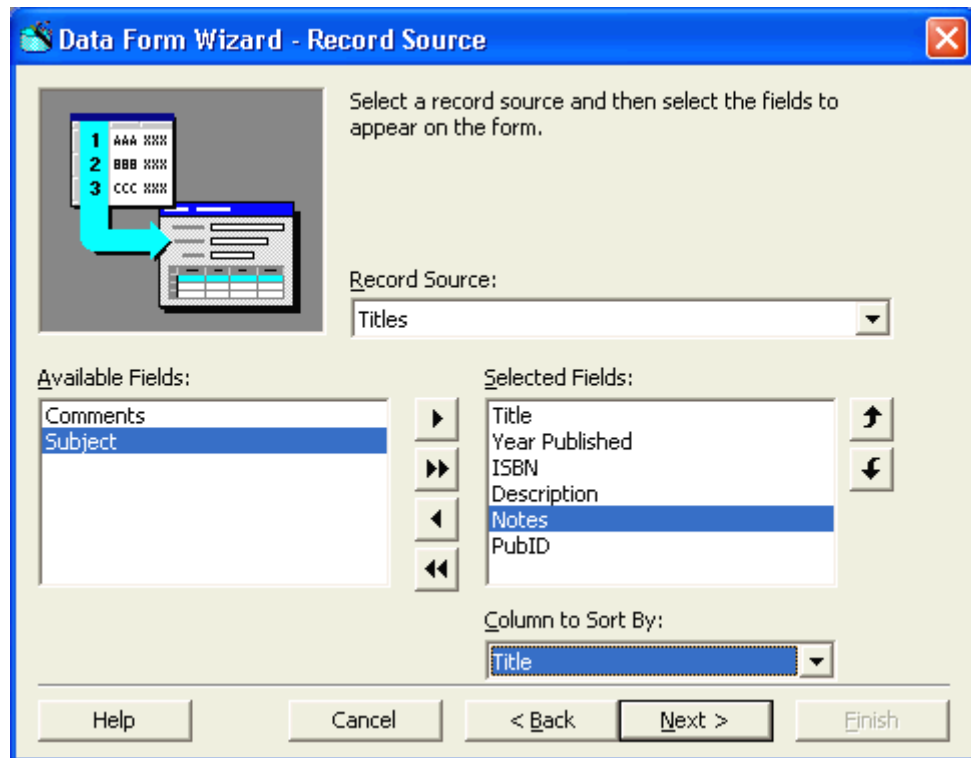
Trong trang Form, ta chọn **Single Record** cho **Form Layout** và **Class** cho **Binding Type**.  
Đoạn click **Next**. Nếu ta chọn **ADO Data Control** thì kết quả sẽ giống giống như khi ta dùng  
Control Data DAO như trong một bài trước.



Trong trang bản ghi Source ta chọn **table Titles**. Listbox của **Available Fields** sẽ hiển thị các trường của table Titles. Sau khi chọn một field bằng cách click lên tên field ấy trong Listbox, nếu chúng ta click hình tam giác chỉ qua phải thì tên field ấy sẽ được dời qua nằm dưới cùng trong Listbox **Selected Fields** bên phải.

Nếu chúng ta click hình hai tam giác chỉ qua bên phải thì tất cả mọi fields còn lại bên trái sẽ được dời qua bên phải. Chúng ta cũng có thể sắp đặt vị trí của các selected fields bằng cách click lên tên field ấy rồi click hình mũi tên chỉ lên hay xuống để di chuyển field ấy lên hay xuống trong danh sách các fields.

Ngoài ra, chúng ta hãy chọn Title làm **Column to Sort By** trong cái Combobox của nó để các records trong Recordset được sắp xếp theo thứ tự ABC (alphabetical order) của field Tiêu đề (Title).

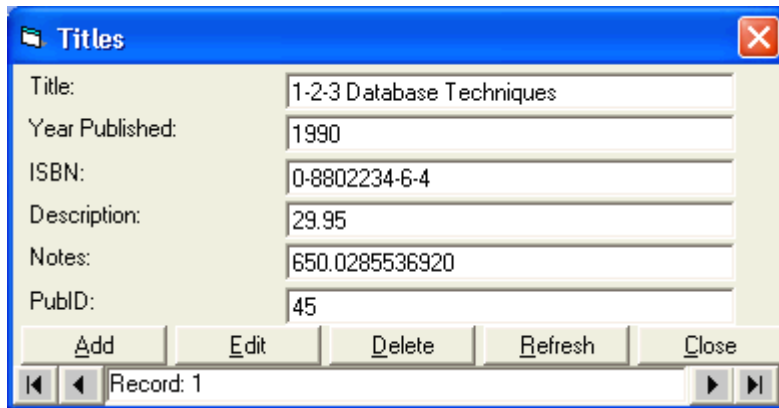


Trong trang Control Selection, ta sẽ để y nguyên để có đủ mọi buttons. Chúng ta hãy click **Next**.



Khi Data Form Wizard chấm dứt, nó sẽ generate form **frmADODDataForm**. Chúng ta hãy remove Form1 và dùng Menu Command **Project | ADODDataControl Properties...** để đổi

**Startup Object** thành frmADODataForm. Thế là tạm xong chương trình để Edit các records của table Titles.



Chúng ta hãy quan sát cái Form và phần code được Data Form Wizard generated. Trong frmADODataForm, các textboxes làm thành một array tên txtFields. Mọi textbox đều có property **DataField** định sẵn tên field của table Titles. Ví dụ như **txtFields(2)** có DataField là **ISBN**. Form chính không dùng Control Data ADO nhưng dùng một Object của **class clsTitles**.

Phần Initialisation của class clsTitles là Open một Connection và lấy về một Dataset có tên **DataMember** là **Primary** như sau:

```
Private Sub Class_Initialize()  
    Dim db As Connection  
    Set db = New Connection  
    db.CursorLocation = adUseClient  
    ' Open connection  
    db.Open "PROVIDER=Microsoft.Jet.OLEDB.3.51;Data  
Source=E:\Websites\Vovisoft\VisualBasic\ADOFrm\BIBLIO.MDB;"  
    ' Instantiate ADO recordset  
    Set adoPrimaryRS = New Recordset  
    ' Retrieve data for Recordset  
    adoPrimaryRS.Open "select Title, [Year  
Published], ISBN, Description, Notes, PubID from Titles Order by Title",  
_ db, adOpenStatic, adLockOptimistic  
    ' Define the only data member, named Primary  
    DataMembers.Add "Primary"  
End Sub
```



Về vị trí của database, nếu chúng ta không muốn ở một folder nào thì dùng `App.Path` để xác định mối liên hệ giữa vị trí của database và folder của chính chương trình đang chạy, ví dụ như:

```
db.Open "PROVIDER=Microsoft.Jet.OLEDB.3.51;Data Source=" &  
App.Path & "\BIBLIO.MDB;"
```

Trong **Sub Form\_Load**, ta có thể dùng **For Each** để đi qua hết các textboxes trong array `txtFields`. Vì property `Datasource` của textbox là một Object nên ta dùng keyword **Set** để point nó đến Object **PrimaryCLS**. Đồng thời ta cũng phải chỉ định tên của `DataMember` của mỗi textbox là **Primary**:

```
Private Sub Form_Load()  
    ' Instantiate an Object of class clsTitles  
    Set PrimaryCLS = New clsTitles  
    Dim oText As TextBox  
    ' Iterate through each textbox in the array txtFields  
    ' Bind the text boxes to the data source, i.e. PrimaryCLS  
    For Each oText In Me.txtFields  
        oText.DataMember = "Primary"  
        ' Use Set because property Datasource is an Object  
        Set oText.DataSource = PrimaryCLS  
    Next  
End Sub
```

Khi sự di chuyển từ bản ghi này đến bản ghi khác chấm dứt, chính `Recordset` có raise **Event MoveComplete**. Event ấy được handled (giải quyết) trong class `clsTitles` bằng cách lại raise **Event MoveComplete** để nó được handled trong `Form`.

```
Dim WithEvents adoPrimaryRS As Recordset  
Private Sub adoPrimaryRS_MoveComplete(ByVal adReason As  
ADODB.EventReasonEnum, _  
    ByVal pError As ADODB.Error, adStatus As  
ADODB.EventStatusEnum, ByVal pRecordset As ADODB.Recordset)  
    ' Raise event to be handled by main form  
    RaiseEvent MoveComplete  
End Sub
```

Muốn handle Event trong `clsTitles` ta phải declare recordset `adoPrimaryRS` với `WithEvents`:

Và trong `Form` ta cũng phải declare (object `clsTitles`) `PrimaryCLS` với `WithEvents`:

```
Private WithEvents PrimaryCLS As clsTitles
```

Trong Form, Event MoveComplete sẽ làm hiển thị vị trí tuyệt đối (Absolute Position) của bản ghi bằng code dưới đây:

```
Private Sub PrimaryCLS_MoveComplete()  
    'This will hiển thị the current record position for this  
recordset  
    lblStatus.Caption="Record: " & CStr(PrimaryCLS.AbsolutePosition)  
End Sub
```

Khi người sử dụng clicks **Refresh**, các textboxes sẽ được hiển thị lại với chi tiết mới nhất của bản ghi từ trong recordset, nhờ khi có ai khác đã sửa đổi bản ghi. **Method Requery** của clsTitles lại gọi method Requery của Recordset như sau:

```
Private Sub cmdRefresh_Click()  
    'This is only needed for multi user applications  
    On Error GoTo RefreshErr  
    ' fetch the latest copy of Recordset  
    PrimaryCLS.Requery  
    Exit Sub  
RefreshErr:  
    MsgBox Err.Description  
End Sub  
  
'In Class clsTitles  
Public Sub Requery()  
    ' Fetch latest copy of record  
    adoPrimaryRS.Requery  
    DataMemberChanged "Primary"  
End Sub
```

## MỤC LỤC

<b>BÀI 1. MỞ ĐẦU</b> .....	<b>1</b>
<b>BÀI 2. ACCESS</b> .....	<b>3</b>
2.1. Giới thiệu .....	3
2.2. Khởi động ACCESS.....	3
2.3. Khái niệm về cơ sở dữ liệu trong Access .....	4
2.4. Các phép toán.....	5
2.4.1 Các phép toán Logic .....	5
2.4.2 Các phép toán số học .....	5
2.4.3 Các phép toán so sánh : >, >=, <, <=, = và <> .....	6
2.4.4 Dấu rào : .....	6
<b>BÀI 3. LÀM VIỆC VỚI CƠ SỞ DỮ LIỆU</b> .....	<b>7</b>
3.1. TẠO CƠ SỞ DỮ LIỆU.....	7
3.1.1 Tạo cơ sở dữ liệu bằng WIZARD .....	7
3.1.2 Tạo cơ sở dữ liệu trống .....	8
3.2. Hiệu chỉnh cơ sở dữ liệu .....	9
<b>BÀI 4. LÀM VIỆC VỚI TABLE</b> .....	<b>11</b>
4.1. Tạo cấu trúc của Table.....	11
4.1.1 Tạo Table bằng Wizard.....	11
4.1.2 Tạo Table bằng DATASHEET VIEW.....	11
4.1.3 Tạo Table bằng DESIGN VIEW .....	13
4.2. Nhập số liệu vào Table.....	14
4.3. Hiệu chỉnh Table.....	15
4.3.1 Thay đổi cấu trúc bản ghi.....	15
4.3.2 Thay đổi nội dung bản ghi .....	15
4.3.3 Thay đổi cách trình bày.....	16
4.4. Khai thác số liệu trên Table.....	16
4.4.1 Tìm và thay thế .....	16
4.4.2 Thay đổi vị trí trường.....	16
4.4.3 Sắp xếp.....	16
4.4.4 Lọc bản ghi.....	17
<b>BÀI 5. LÀM VIỆC VỚI QUERY</b> .....	<b>22</b>
5.1. Khái niệm.....	22
5.2. Cách tạo QUERY .....	23
5.2.1 Select Query .....	24
5.2.2 Cross Tab Query .....	26
5.3. Hiệu chỉnh QUERY .....	28
5.4. Thực hiện QUERY.....	28
<b>BÀI THỰC HÀNH</b> .....	<b>29</b>
<b>BÀI 6. LÀM VIỆC VỚI REPORT</b> .....	<b>34</b>
6.1. Khái niệm.....	34
6.2. Cách tạo Report .....	34
6.3. Hiệu chỉnh Report.....	39

6.4. Thực hiện Report.....	39
<b>BÀI THỰC HÀNH.....</b>	<b>40</b>
<b>BÀI 7. LÀM VIỆC VỚI FORM.....</b>	<b>42</b>
7.1. Khái niệm :.....	42
7.2. Thiết kế Form : .....	42
7.3. Hiệu chỉnh Form.....	47
7.4. Thực hiện Form .....	47
<b>BÀI THỰC HÀNH .....</b>	<b>48</b>
<b>BÀI 8. MACRO VÀ HỆ THỐNG THỰC ĐƠN.....</b>	<b>53</b>
8.1. MACRO.....	53
8.1.1 1. Khái niệm : .....	53
8.1.2 Cách tạo Macro .....	53
8.1.3 Thực hiện Macro.....	54
8.2. Hệ thống thực đơn .....	54
8.2.1 Cách tạo thực đơn: .....	54
8.2.2 Sử dụng thực đơn.....	57
<b>BÀI THỰC HÀNH .....</b>	<b>58</b>
<b>BÀI 9. MỞ ĐẦU .....</b>	<b>61</b>
9.1. Giới thiệu.....	61
9.2. Các khái niệm thường dùng .....	63
9.3. Làm việc với Visual Basic.....	63
9.3.1 Cài đặt : .....	63
9.3.2 Khởi động .....	64
9.3.3 Màn hình làm việc .....	64
9.3.4 Kết thúc.....	65
<b>BÀI 10. LẬP TRÌNH TRONG VISUAL BASIC .....</b>	<b>66</b>
10.1. Làm việc với hộp điều khiển.....	67
10.1.1 Các loại hộp điều khiển : trên thanh Tools Bar có các nút điều khiển thường sử dụng như : .....	67
10.1.2 Thêm hộp điều khiển lên biểu mẫu .....	68
10.1.3 Hiệu chỉnh hộp điều khiển : .....	69
10.2. THUỘC TÍNH .....	69
10.2.1 Khi thiết kế : .....	69
10.2.2 Khi thực hiện chương trình.....	70
10.2.3 Các loại thuộc tính : .....	70
10.3. Thủ tục tình huống:.....	72
<b>BÀI THỰC HÀNH .....</b>	<b>73</b>
10.4. Thay đổi thuộc tính : .....	74
10.4.1 Hộp Text : .....	74
10.4.2 Các hộp Command Button : .....	74
10.4.3 Các hộp Check Box : .....	74
10.4.4 Đổi Font : .....	74
10.5. Viết các thủ tục tình huống : .....	75
10.5.1 Thủ tục của Form : đây là thủ tục chứa các chỉ thị khởi tạo giá trị ban đầu.	75

10.5.2	Thủ tục của các hộp Command :	75
10.5.3	Thủ tục của các hộp Check Box :	76
<b>10.6.</b>	<b>Ghi và thực hiện tương trình :</b>	<b>76</b>
10.6.1	Lưu trữ :	76
10.6.2	Xem mã lệnh :	77
<b>BÀI 11.</b>	<b>BIẾN NHỚ</b>	<b>83</b>
11.1.	Khái niệm :	83
11.2.	Khai báo biến :	83
11.2.1	Khai báo bằng	83
11.2.2	Cách viết	84
11.2.3	Khai báo biến toàn cục	85
11.2.4	Khai báo nhiều biến	85
11.3.	Khai báo hằng :	86
11.4.	Khai báo mảng :	86
11.4.1	Khai báo mảng :	86
11.4.2	Sử dụng mảng :	87
11.5.	Khai báo bảng ghi :	88
11.5.1	Khai báo :	88
11.5.2	Sử dụng biến bản ghi :	88
11.6.	Biến đổi (convert) từ loại dữ liệu này qua loại dữ liệu khác	89
<b>BÀI 12.</b>	<b>CÁC CẤU TRÚC ĐIỀU KHIỂN</b>	<b>90</b>
12.1.	Cấu trúc chọn :	90
12.1.1	Cấu trúc : <b>IF</b>	90
12.1.2	Cấu trúc : <b>IF ... ELSE</b>	90
12.1.3	Cấu trúc : <b>Select Case &lt;Biểu thức&gt;</b>	91
12.2.	Cấu trúc lặp	92
12.2.1	Cấu trúc :	92
12.2.2	Cấu trúc :	93
12.2.3	Cấu trúc :	93
12.3.	Nhãn :	94
12.3.1	Nhãn :	95
12.3.2	Số thứ tự dòng lệnh :	96
<b>BÀI 13.</b>	<b>METHOD</b>	<b>97</b>
13.1.	Circle Method	97
13.2.	Line Method	98
13.3.	Cls Method	99
13.4.	Hide Method	100
13.5.	Show Method	100
13.6.	Item Method	101
13.7.	Move Method	101
13.8.	Point Method	102
13.9.	Print Method	103
13.10.	PrintForm Method	103
13.11.	PSet Method	104
13.12.	Refresh Method	105

13.13.	Scale Method .....	105
13.14.	SetFocus Method.....	107
13.15.	Show Method.....	107
13.16.	TextHeight và TextWidth Methods.....	107
<b>BÀI 14.</b>	<b>HÀM.....</b>	<b>109</b>
14.1.	Các hàm xử lý chuỗi :.....	109
14.1.1	Tìm chiều dài chuỗi : LEN(String) .....	109
14.1.2	Chuyển sang chữ thường : .....	109
14.1.3	Chuyển sang chữ in : .....	109
14.1.4	Lấy các ký tự bên trái : .....	109
14.1.5	Lấy các ký tự bên phải:.....	110
14.1.6	Lấy nhóm ký tự bất kỳ:.....	110
14.1.7	Bỏ các ký tự trống:.....	110
14.1.8	Bỏ các ký tự trống bên trái: .....	110
14.1.9	Bỏ các ký tự trống bên phải:.....	110
14.1.10	Đổi mã số sang ký tự: .....	111
14.1.11	Đổi ký tự sang mã số: .....	111
14.1.12	Đổi chuỗi sang số: .....	111
14.1.13	Đổi số sang chuỗi: .....	111
14.1.14	Định dạng chuỗi:.....	111
14.1.15	Tìm chuỗi con: .....	112
14.2.	Các hàm xử lý số : .....	113
<b>BÀI 15.</b>	<b>DÙNG LIST CONTROLS .....</b>	<b>115</b>
15.1.	Listbox .....	116
15.1.1	Hiện thị nhiều sự lựa chọn.....	116
15.1.2	Save content của Listbox .....	117
15.1.3	Load một Text file vào Listbox .....	119
15.2.	Drag-Drop .....	120
15.3.	Dùng Property Sorted .....	122
<b>BÀI 16.</b>	<b>TỰ TẠO OBJECT.....</b>	<b>127</b>
<b>BÀI 17.</b>	<b>DEBUG .....</b>	<b>136</b>
17.1.	Đặc tả chương trình (Program Specifications).....	136
17.1.1	Cấu trúc các bộ phận.....	137
17.1.2	Kỹ thuật lập trình.....	137
17.1.3	Dùng Subs và Functions .....	137
17.2.	Một số lưu ý.....	138
17.2.1	Dùng sự Error .....	138
17.2.2	Dùng Comment (Chú thích) .....	139
17.2.3	Đặt tên các variables có ý nghĩa .....	139
17.2.4	Dùng Option Explicit.....	139
17.2.5	Desk Check.....	140
17.2.6	Soạn một Test Plan .....	140
17.3.	Các kỹ thuật xử lý lỗi .....	140
17.3.1	Xử lý Error lúc Run time .....	140
17.3.2	Dùng Breakpoints .....	141

17.3.3	Dùng Immediate Window .....	143
17.3.4	Theo dấu chân chương trình (Tracing) .....	143
17.3.5	Dùng Watch Window.....	145
17.3.6	Dùng phương pháp loại suy (Elimination Method).....	145
<b>BÀI 18.</b>	<b>DÙNG MENU .....</b>	<b>147</b>
18.1.	Main Menu.....	147
18.2.	Pop-up Menu .....	151
18.3.	Chứa menu Settings trong Registry .....	153
<b>BÀI 19.</b>	<b>DÙNG DIALOGS .....</b>	<b>161</b>
19.1.	Message Boxes .....	161
19.2.	Input Boxes.....	164
19.3.	Common Dialogs .....	166
19.4.	Open và Save File Dialogs .....	167
19.5.	Các loại Dialog có sẵn để dùng .....	171
19.5.1	Color Dialog.....	171
19.5.2	Font Dialog .....	173
19.5.3	Print Dialog.....	174
19.5.4	Help Dialog.....	176
19.6.	Custom Dialogs.....	176
<b>BÀI 20.</b>	<b>DÙNG ĐỒ HỌA .....</b>	<b>180</b>
20.1.	Màu (color) và độ mịn (resolution) .....	180
20.1.1	Độ mịn (resolution).....	180
20.1.2	Màu (color) .....	182
20.2.	Function RGB.....	185
20.3.	Color Mapping.....	187
20.4.	Dùng Intrinsic Color Constants.....	188
20.5.	Graphic files.....	189
<b>BÀI 21.</b>	<b>CƠ SỞ DỮ LIỆU (DATABASE).....</b>	<b>190</b>
21.1.	Table, Record và Field .....	190
21.2.	Primary Key và Index.....	191
21.3.	Relationship và Foreign Key .....	193
21.4.	Relational Database.....	195
21.5.	Các lợi ích.....	195
21.6.	Integrity Rules (các quy luật liên chính).....	196
21.6.1	General Integrity Rules .....	196
21.6.2	Database-Specific Integrity Rules .....	197
21.7.	Microsoft Access Database Management System .....	197
21.8.	Properties Required và Allow Zero Length.....	197
21.9.	Làm việc với các versions khác nhau .....	198
21.10.	Dùng Query để viết SQL.....	199
21.11.	Dùng Link Table để làm việc trực tiếp với database loại khác.....	200
21.12.	Database Server và một số khái niệm.....	200
<b>BÀI 22.</b>	<b>SỬ DỤNG CONTROL DATA .....</b>	<b>202</b>

<b>22.1.</b>	Control Data .....	202
<b>22.2.</b>	Chỉ định vị trí Database lúc chạy chương trình.....	207
<b>22.3.</b>	Thêm bớt các Records .....	207
<b>22.4.</b>	Dùng DataBound Combo .....	210
<b>BÀI 23.</b>	<b>LẬP TRÌNH VỚI KỸ THUẬT DAO .....</b>	<b>213</b>
<b>23.1.</b>	Reference DAO.....	213
<b>23.2.</b>	Dùng keyword SET.....	214
<b>23.3.</b>	Các nút di chuyển.....	216
<b>23.4.</b>	Thêm bớt các Records .....	218
<b>23.5.</b>	Tìm một bản ghi.....	221
<b>23.6.</b>	Bookmark .....	224
<b>23.7.</b>	LastModified .....	224
<b>BÀI 24.</b>	<b>LẬP TRÌNH VỚI ADO .....</b>	<b>226</b>
<b>24.1.</b>	Control Data ADO.....	226
<b>24.2.</b>	Data Form Wizard .....	231





# Microsoft Office 2003

**Học phần:**

**Microsoft Access 2003**

**Nguyễn Duy Thành**

Trung tâm CNTT – ĐH Huế

Y!M: [duythanh\\_bluestar](#)

E-mail: [ndthanh@hueuni.edu.vn](mailto:ndthanh@hueuni.edu.vn)



## Nội dung

- **Phần I**
  - Tổng quan về Microsoft Access
- **Phần II**
  - Bảng dữ liệu
- **Phần III**
  - Truy vấn
- **Phần IV**
  - Biểu mẫu
- **Phần V**
  - Báo cáo



# Phần I

## Tổng quan về Microsoft Access



## Microsoft Access (1)

- Là hệ quản trị cơ sở dữ liệu (QTCSDL) tương tác người sử dụng chạy trong môi trường hệ điều hành Microsoft Windows.
- Cung cấp công cụ hiệu lực trong công tác tổ chức, tìm kiếm và biểu diễn thông tin.
- Cung cấp khả năng thao tác dữ liệu, khả năng liên kết và công cụ truy vấn mạnh mẽ giúp quá trình tìm kiếm thông tin nhanh chóng.



## Microsoft Access (2)

- Cung cấp khả năng kết xuất dữ liệu cho phép người sử dụng thiết kế những biểu mẫu và báo cáo phức tạp; có thể truy xuất dữ liệu và kết hợp các biểu mẫu và báo cáo trong một tài liệu và trình bày kết quả theo hướng chuyên nghiệp.
- Cung cấp các Wizard (thuật sĩ – thao tác được cài đặt sẵn) và các lệnh sẵn có (macro) giúp dễ dàng tự động hóa công việc mà không cần lập trình.

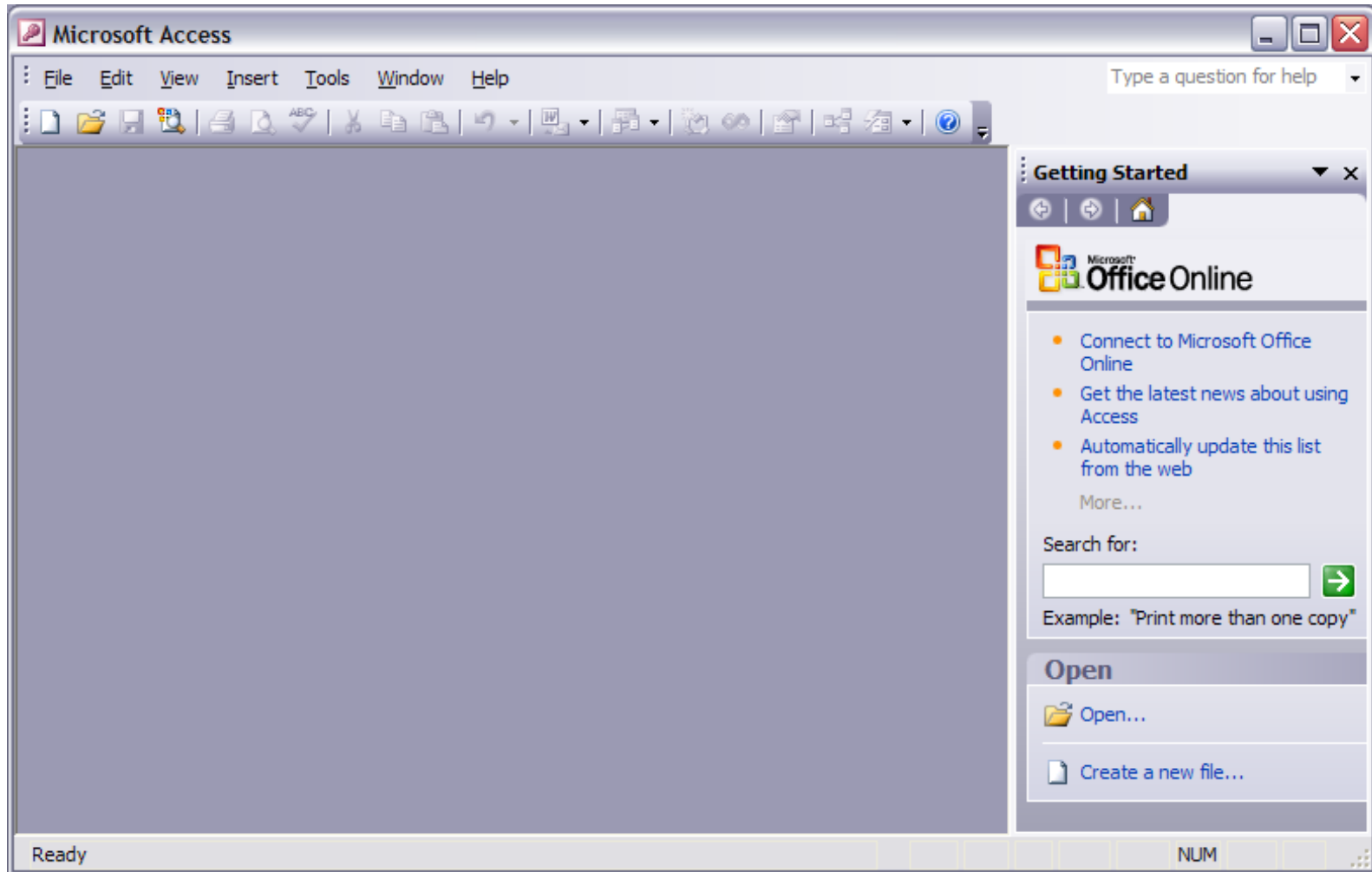


## Khởi động MS Access

- Đối với MS Access 2000/XP:
  - Start -> Program -> Microsoft Access
- Đối với MS Access 2003:
  - Start -> Program -> Microsoft Office -> Microsoft Office Access 2003
  - Hoặc trên màn hình Desktop, double click vào biểu tượng Microsoft Access.
- Sau khi khởi động thành công, màn hình làm việc của MS Access có dạng như sau:

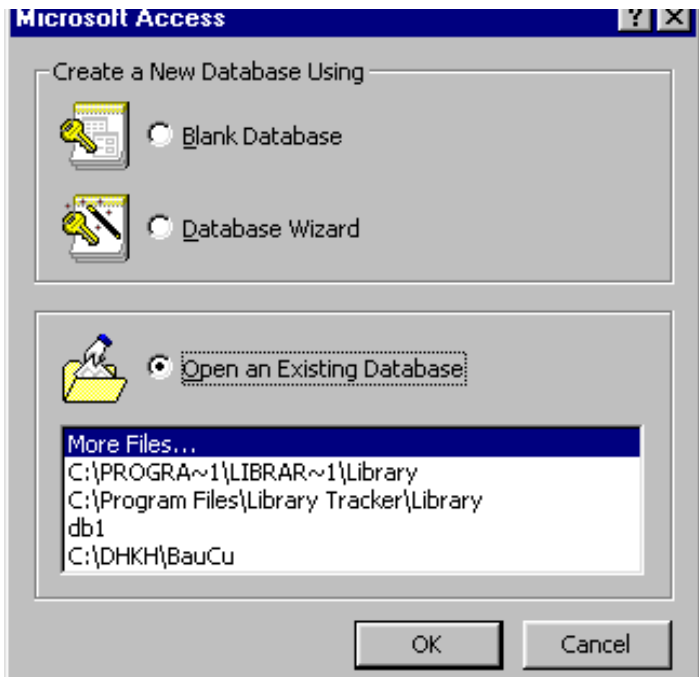


# Màn hình khởi động (1)

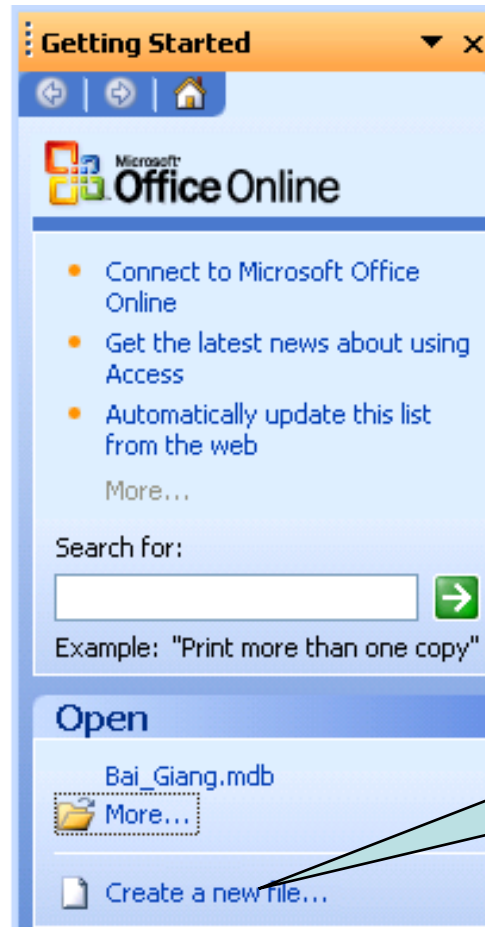


## Màn hình khởi động (2)

- MS Access 2000



- MS Access XP hoặc 2003




Tạo CSDL mới





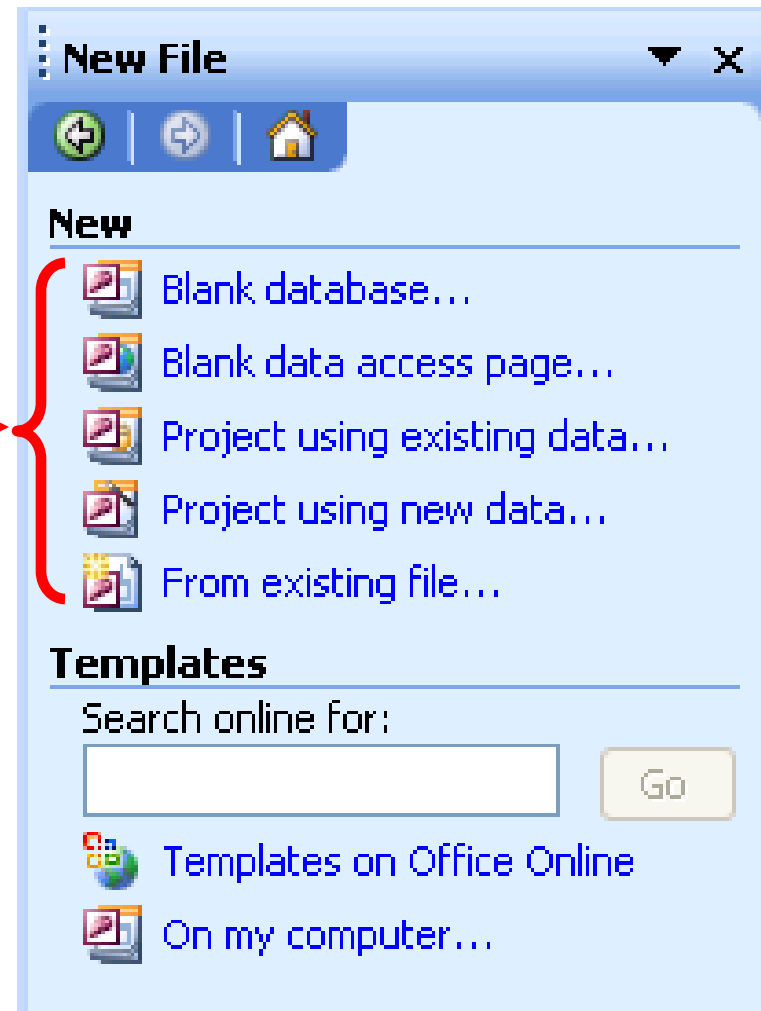


## Thoát khỏi MS Access

- Sử dụng một trong 3 cách sau:
  - File -> Exit
  - Alt + F4
  - Click nút  trên thanh title.

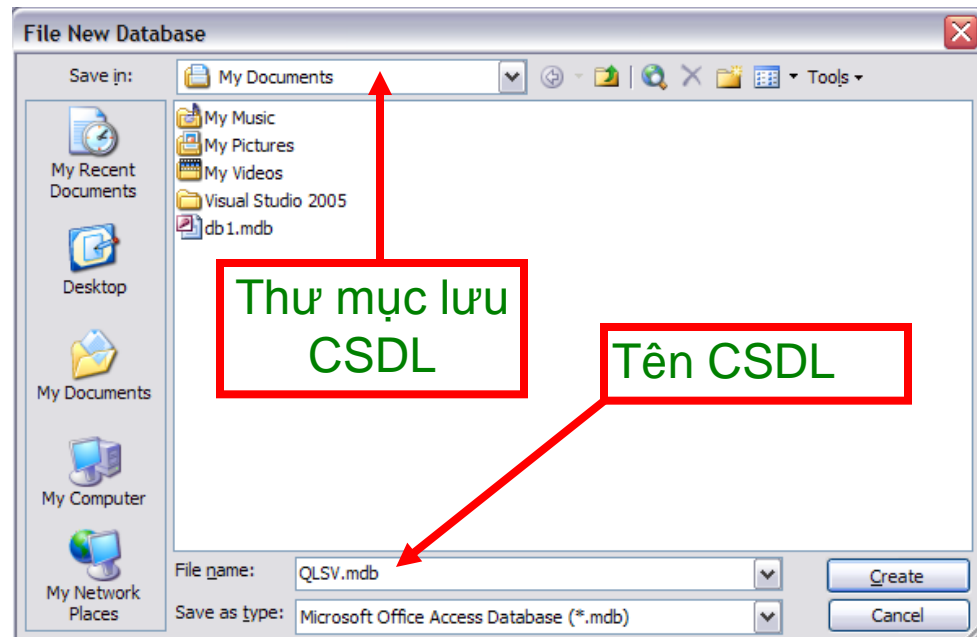
## Tạo cơ sở dữ liệu (1)

- Có 3 cách:
  - Trên menu File, chọn New
  - Click vào biểu tượng New  trên thanh công cụ
  - Chọn chức năng *Create a new file* trong ngăn tác vụ **Getting Started**.
- Ngăn tác vụ **New file** xuất hiện bao gồm: 
- Chọn loại cơ sở dữ liệu muốn tạo. Chẳng hạn như chọn *Blank database*.



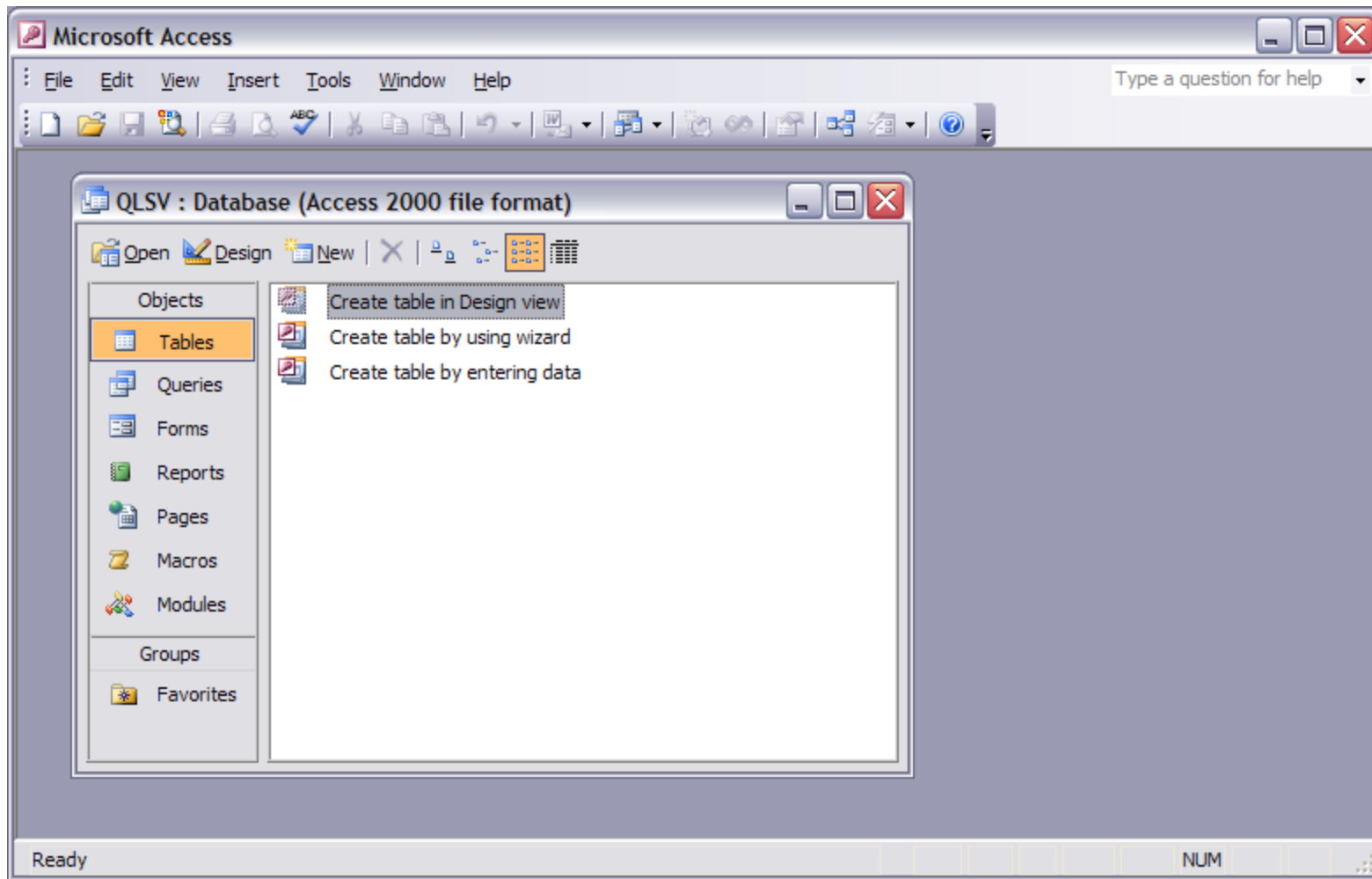
## Tạo cơ sở dữ liệu (2)

- Xuất hiện hộp thoại File New Database.
- Chọn thư mục lưu CSDL trong ô Save in.
- Nhập tên file CSDL trong ô File name và nhấn Create.
- Xuất hiện cửa sổ làm việc của CSDL.



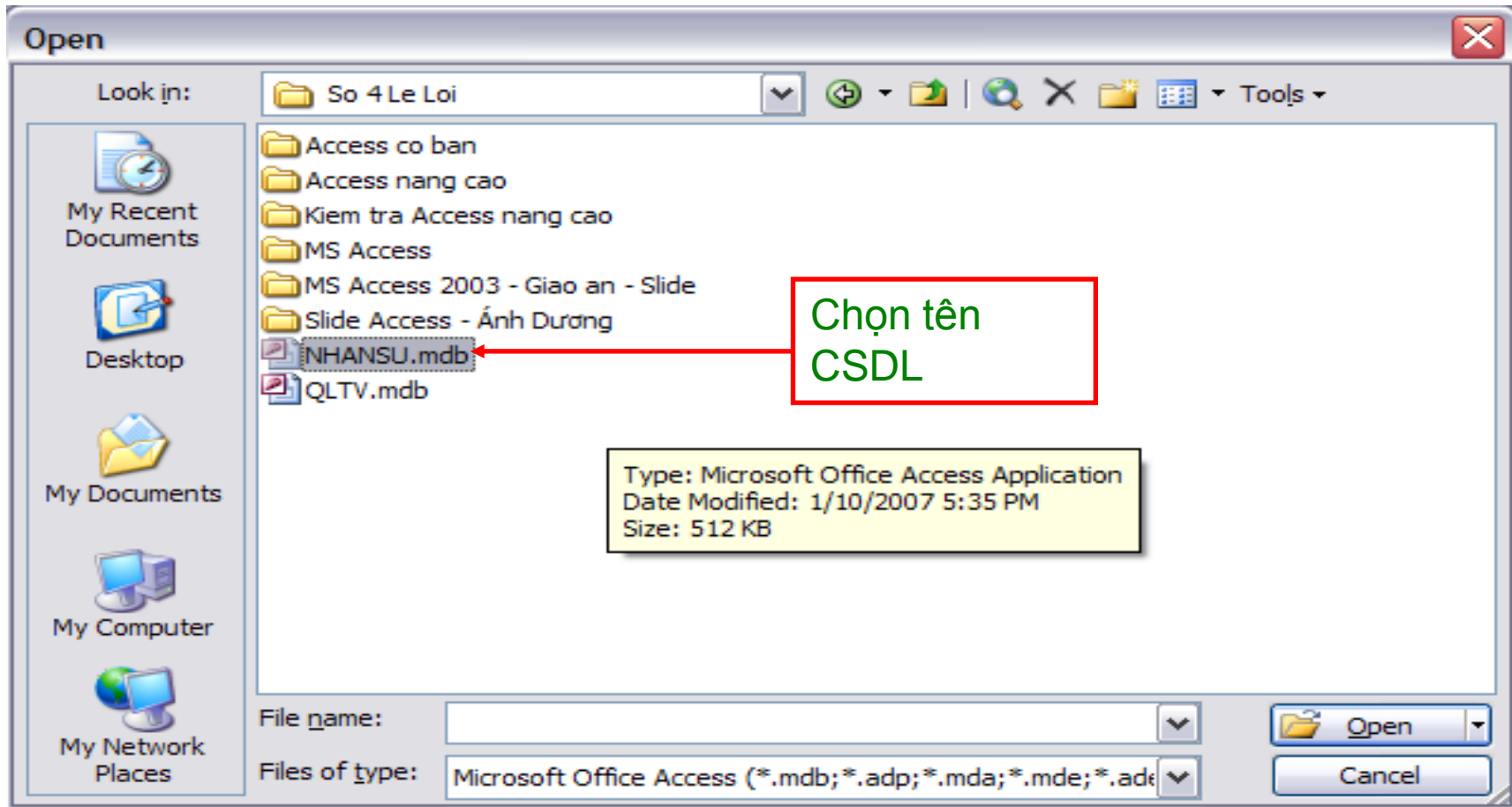


# Cửa sổ làm việc với CSDL QLSV






## Mở một file CSDL có sẵn





Microsoft Office Access

 **Security Warning: Unsafe expressions are not blocked.**  
Do you want to block unsafe expressions?

Security Warning

Opening "E:\My Documents\1Giao trinh Trung tam Tin hoc\Van phong\MSAccess\So 4 Le Loi\NHANSU.mdb"

This file may not be safe if it contains code that was intended to harm your computer.  
Do you want to open this file or cancel the operation?

NHANSU : Database (Access 2000 file format)

Open Design New | X | | | | | | | |

Objects

- Tables
- Queries
- Forms
- Reports
- Pages
- Macros
- Modules

Groups

- Favorites

Create table in Design view  
Create table by using wizard  
Create table by entering data  
DSPhong  
DSPXuong  
NhanVien  
ThuNhap\_NV



## Đóng cơ sở dữ liệu

- Thực hiện một trong ba cách sau đây:
  - File -> Close
  - Ctrl + F4
  - Click nút close × trên thanh menu.



## Các thành phần cơ bản

- Để hỗ trợ nhiều cách quản lý CSDL, Access hỗ trợ một số đối tượng quản lý sau:
  - Bảng (Tables)
  - Truy vấn (Queries)
  - Biểu mẫu (Forms)
  - Báo cáo (Reports)
  - Trang web (Pages)
  - Tập lệnh (Macros)
  - Đơn thể chương trình (Modules)



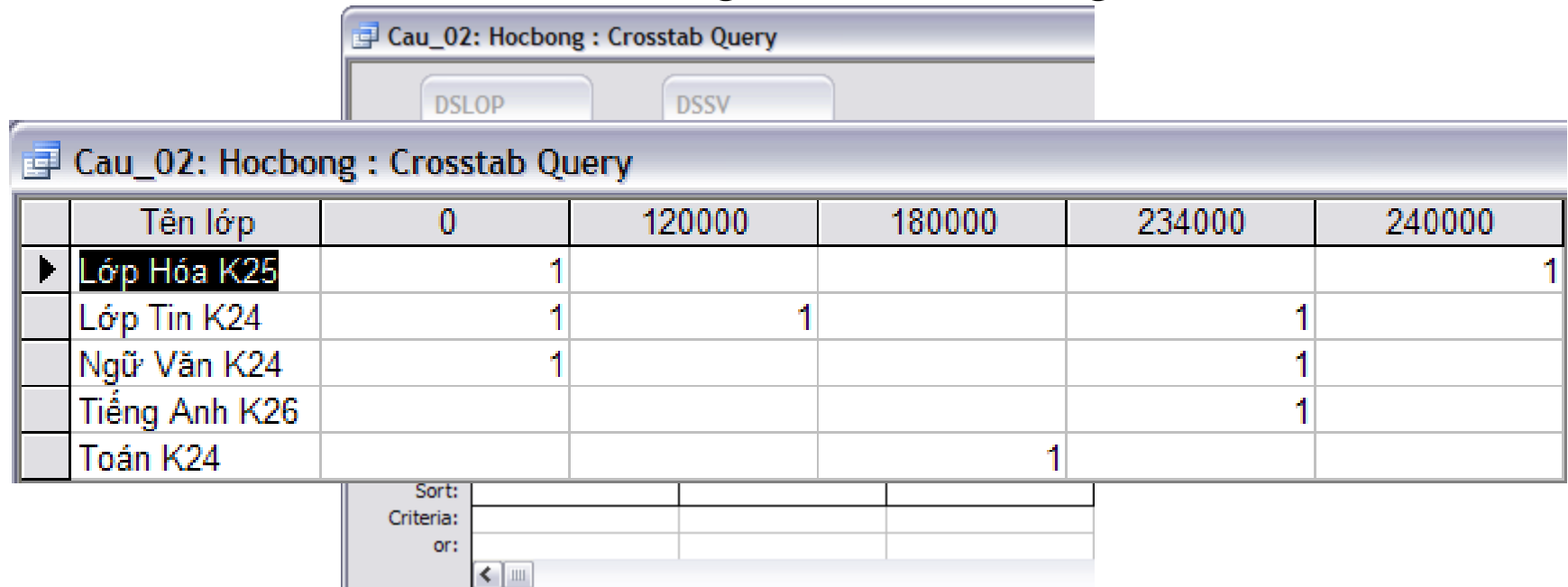
## Bảng (Tables)

- Là thành phần cơ bản của CSDL Access.
- Trong mỗi bảng gồm nhiều **trường (Field)**.
- Một CSDL có thể có nhiều bảng.
- Lưu trữ các thông tin cần quản lý.

	Mã SV	Mã lớp	Họ tên SV	Ngày sinh	Quê quán	Giới tính	Học bổng
▶ +	SV10	TI24	Trần Thu Dung	02/02/1982	Nghệ An	Nam	234000
+	SV09	TA26	Nguyễn Thanh Nhân	05/06/1984	Thái Bình	Nam	234000
+	SV07	VA25	Trần Lê Anh Thư	05/03/1984	Thanh Hóa	Nam	234000
+	SV02	TI24	Hoàng Lan Hương	10/05/1981	TT Huế	Nam	0
+	SV08	VA25	Đỗ Trung Hiếu	20/10/1983	Quảng Ngãi	Nữ	0
+	SV06	HO25	Đặng Minh Quang	04/03/1983	Hà Tĩnh	Nữ	0
+	SV05	HO25	Hữu Nghiêm	04/06/1982	Khánh Hòa	Nữ	240000
+	SV03	TO24	Lê Mạnh Quân	15/05/1982	Nghệ An	Nữ	180000
+	SV01	TI24	Nguyễn Trường Giang	12/03/1982	Quảng Bình	Nữ	120000
*						Nữ	0

# Truy vấn (Queries)

- Là công cụ để truy vấn thông tin và thực hiện các thao tác trên dữ liệu từ một hay nhiều bảng.
- Ngoài ra còn có thể dùng để tạo ra các báo cáo, biểu mẫu có đặc tính định dạng cao hơn bảng.



The screenshot shows a Microsoft Access Crosstab Query window titled "Cau\_02: Hocbong : Crosstab Query". The window has two buttons, "DSLOP" and "DSSV". Below the window is a table with the following data:

Tên lớp	0	120000	180000	234000	240000
Lớp Hóa K25	1				1
Lớp Tin K24	1	1		1	
Ngữ Văn K24	1			1	
Tiếng Anh K26				1	
Toán K24			1		

Below the table, there are fields for "Sort:", "Criteria:", and "or:" with a scroll bar at the bottom.



# Biểu mẫu (Forms)

DM NhanVien

Mã Nhân viên	NV01	Phụ cấp	0
Họ lót	Nguyễn Văn	Tạm ứng	336,000
Tên	An		
Ngày sinh	12/02/1975		
Phái nam	<input checked="" type="checkbox"/>		
Địa chỉ	TP Huế		
Mã phòng ban	04		
Mã chức vụ	DT		
Mã trình độ	TS		
Lương cơ bản	1,120,000		
Hệ số	3.20		

Record: 1 of 5



### LoạiSach

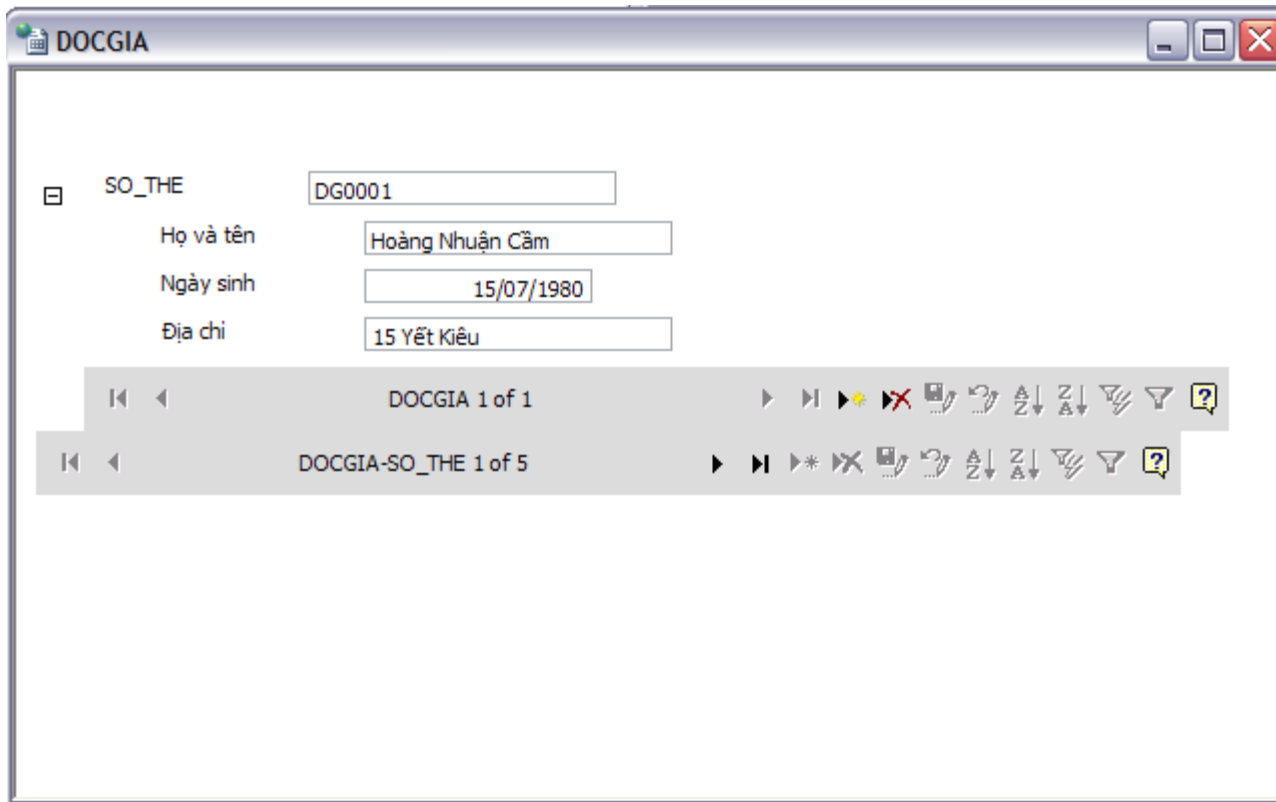
TenSach	TenTacGia	Mã sách	Năm xuất bản	Số trang sách
<b>Đề mèn phiếu lưu ký</b>				
	<i>Tô Hoài</i>	DMPL	1/5/2000	200
<b>Kinh tế vĩ mô</b>				
	<i>Trần Trung Kiên</i>	KTVM	1/5/1998	250
<b>Kỹ thuật LTHT</b>				
	<i>Nguyễn Mạnh Hải</i>	LTHT	7/20/2000	300
<b>Thiên văn Vật lý</b>				
	<i>Đỗ Ngọc Bảo</i>	TVVL	7/12/1999	200
<b>Tin học đại cương</b>				
	<i>Lê Thanh Sơn</i>	THDC	2/5/2005	250
<b>Văn học nước ngoài</b>				
	<i>Hoàng Diệu Linh</i>	VHNN	5/9/2004	250
<b>Vật lý trị liệu</b>				
	<i>Đặng Minh Châu</i>	VLTL	8/14/2000	300

- Dùng thức
- Là có ta có Exce

ều hình  
y vấn.  
ng phú,  
oft

## Trang web (Pages)

- Dùng để cập nhật, hiển thị dữ liệu lên trang web.



The screenshot shows a web browser window with the title "DOCGIA". The main content area displays a form with the following fields:

SO_THE	<input type="text" value="DG0001"/>
Họ và tên	<input type="text" value="Hoàng Nhuận Cầm"/>
Ngày sinh	<input type="text" value="15/07/1980"/>
Địa chỉ	<input type="text" value="15 Yết Kiêu"/>

Below the form, there are two navigation bars. The top bar shows "DOCGIA 1 of 1" and the bottom bar shows "DOCGIA-SO\_THE 1 of 5". Both bars include standard navigation icons (back, forward, search, etc.).



## Tập lệnh (Macros)

- Là một tập hợp các lệnh nhằm tự động hóa những thao tác thường nhật thay vì phải lặp đi lặp lại một cách nhàm chán.
- Khi cho thực hiện một Macro, MS Access sẽ cho thực hiện một loạt các thao tác đã quy định trước.

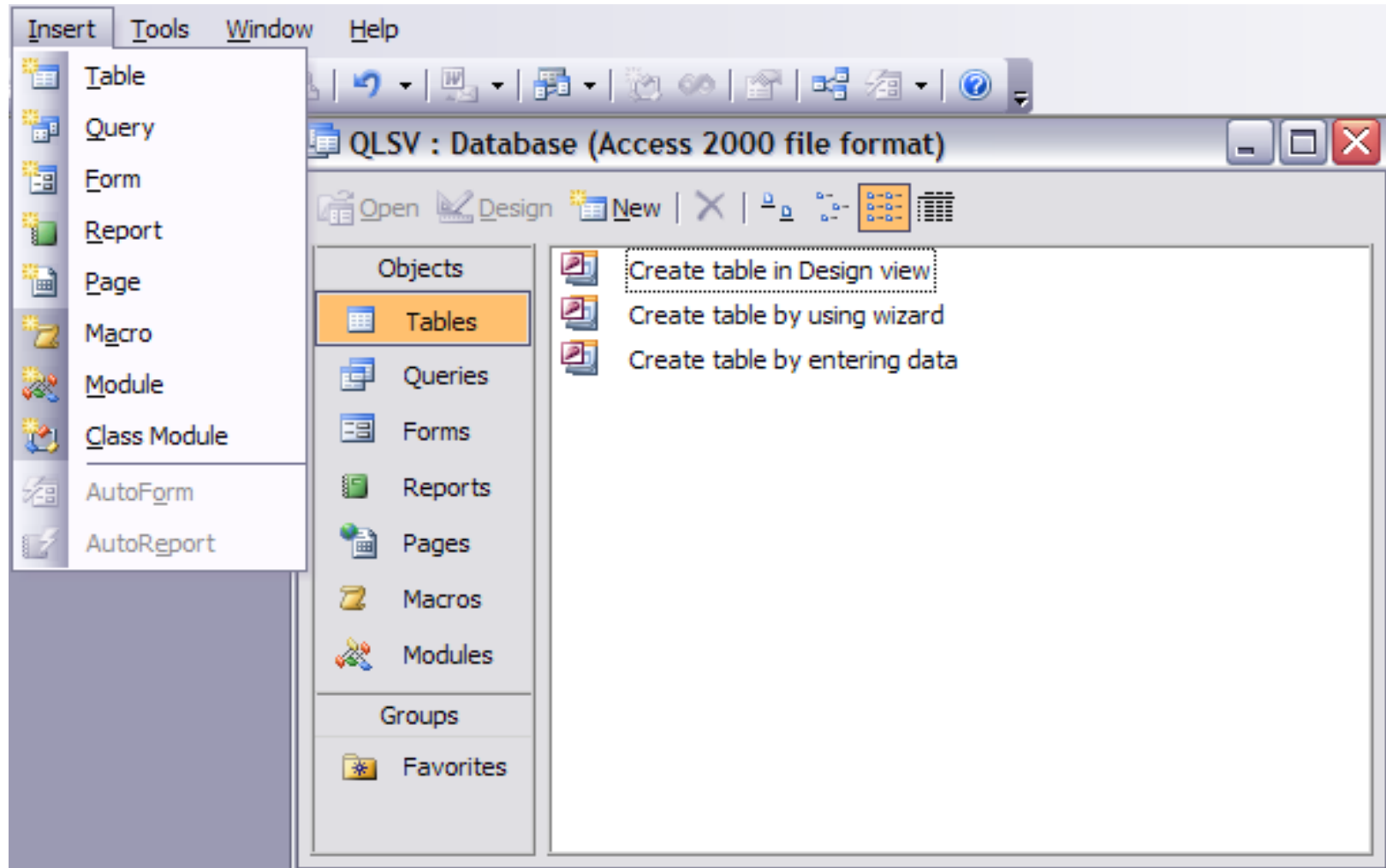


## Đơn thể chương trình (Modules)

- Là một dạng tự động hóa cao cấp hơn Macro.
- Là những đoạn chương trình được viết bằng ngôn ngữ Visual Basic.
- Do người sử dụng tự làm để tự động hóa một số phức tạp mà Macro không đảm trách nổi.
- Phần này dùng trong các yêu cầu quản lý dữ liệu phức tạp.

# Tạo một đối tượng mới

- Table
- Query
- Form
- Report
- Page
- Macro
- Module
- Class Module
- AutoForm
- AutoReport







# Sửa đổi một đối tượng có sẵn

- QLNHANSU : Database (Access 2000 file format)
- The screenshot shows the Microsoft Access 2000 interface for a database named 'QLNHANSU'. The 'Design' view is active. The left-hand 'Objects' pane shows a tree view with 'Tables' selected. The main workspace displays a list of tables: 'CongTac', 'DanhSachNhanVienDangVien', 'DanhSachNhanVienKeToan' (highlighted with a dashed border), 'DanhSachNhanVienLuuTru', 'DM ChucVu', 'DM NhanVien', 'DM PhongBan', and 'HoSo'. The top menu bar includes 'Open', 'Design', and 'New'.



# Phần II

## Bảng dữ liệu

# Bảng dữ liệu

DSSV : Table

Field Name	Data Type	Description
Masv	Text	
Malop	Text	
Hotensv	Text	
Ngaysinh	Date/Time	
Quequan	Text	
Gioitinh	Yes/No	
Hocbong	Number	

Field Properties

General    Lookup

Field Size	50
Format	
Input Mask	
Caption	Họ tên SV
Default Value	
Validation Rule	
Validation Text	
Required	No
Allow Zero Length	Yes
Indexed	No
Unicode Compression	Yes
IME Mode	No Control
IME Sentence Mode	None
Smart Tags	

A field name can be up to 64 characters long, including spaces.  
Press F1 for help on field names.

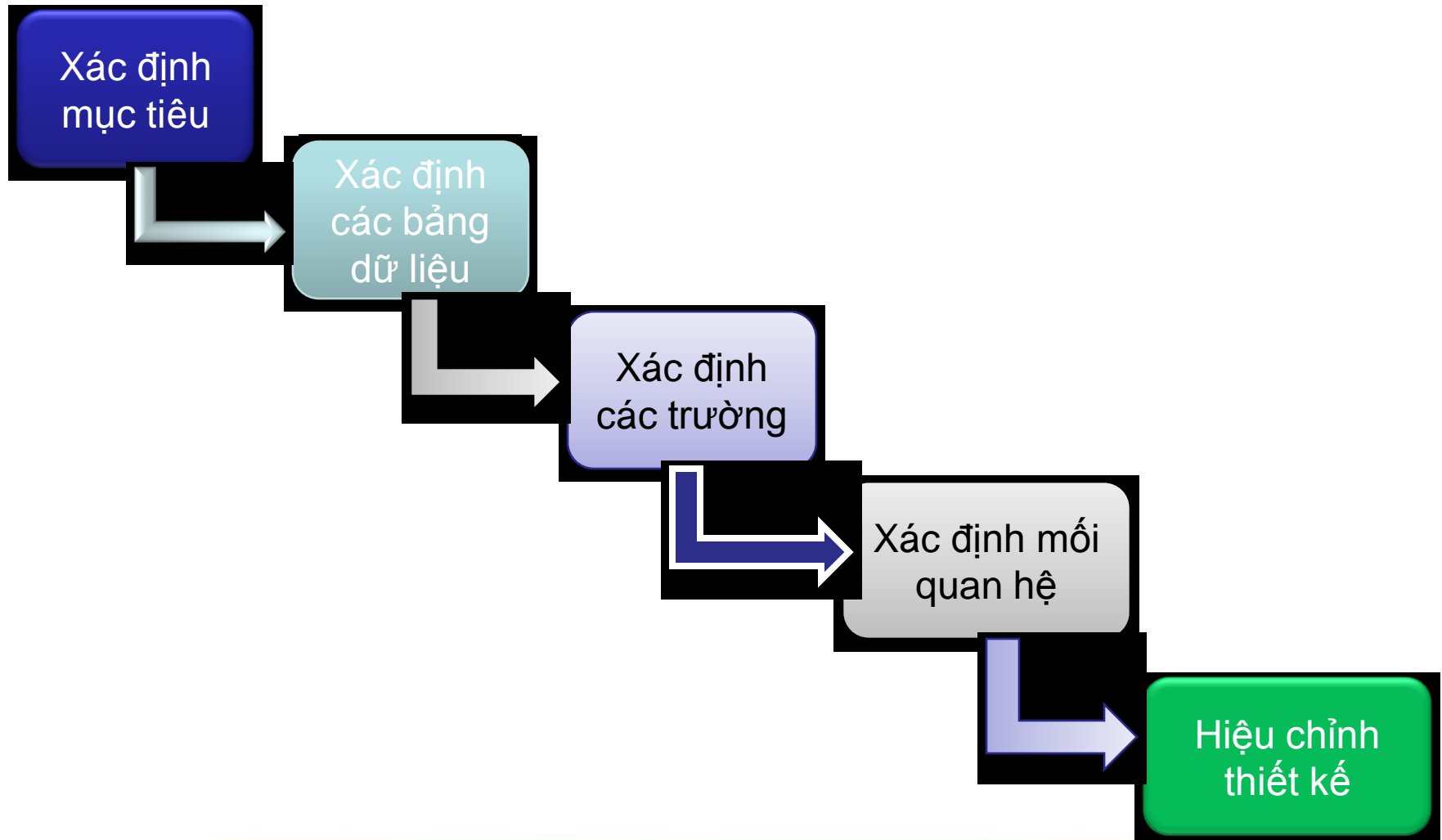


## Thiết kế CSDL

- Dựa vào các yêu cầu của bài toán quản lý, đầu tiên, ta phải thiết kế CSDL.
- Việc thiết kế này phụ thuộc vào các khả năng mà một Hệ QTCSDL có.
- Đối với Access, chúng ta nên tiến hành thiết kế CSDL theo các bước sau:



## Các bước thiết kế CSDL





## Khái niệm bảng

- **Bảng** là nơi chứa dữ liệu về một đối tượng thông tin nào đó, như SINH VIÊN, NHÂN SỰ. Bảng bao gồm nhiều **trường (field)**.
- Mỗi hàng trong bảng được gọi là một **bản ghi (record)**, chứa các nội dung riêng của đối tượng đó.
- Mỗi bản ghi của bảng đều có cùng chung cấu trúc, tức là cùng chung các trường.
- Trong một CSDL có thể chứa nhiều bảng. Mỗi bảng lưu trữ nhiều thông tin về một đối tượng nào đó, mỗi một thông tin đều có kiểu đặc trưng riêng (kiểu dữ liệu các trường).



Field Name	Data Type	Description
Masv	Text	
Malop	Text	
Hotensv	Text	
Ngaysinh	Date/Time	
Ququan	Text	

Mã SV	Mã lớp	Họ tên SV	Ngày sinh	Quê quán	Giới tính	Học bổng
+ SV10	TI24	Trần Thu Dung	02/02/1982	Nghệ An	Nam	234000
+ SV09	TA26	Nguyễn Thanh Nhân	05/06/1984	Thái Bình	Nam	234000
+ SV07	VA25	Trần Lê Anh Thư	05/03/1984	Thanh Hóa	Nam	234000
+ SV02	TI24	Hoàng Lan Hương	10/05/1981	TT Huế	Nam	0
+ SV08	VA25	Đỗ Trung Hiếu	20/10/1983	Quảng Ngãi	Nữ	0
+ SV06	HO25	Đặng Minh Quang	04/03/1983	Hà Tĩnh	Nữ	0
+ SV05	HO25	Hữu Nghiêm	04/06/1982	Khánh Hòa	Nữ	240000
+ SV03	TO24	Lê Mạnh Quân	15/05/1982	Nghệ An	Nữ	180000
+ SV01	TI24	Nguyễn Trường Giang	12/03/1982	Quảng Bình	Nữ	120000
*					Nữ	0

Allow Zero Length	Yes
Indexed	No
Unicode Compression	Yes
IME Mode	No Control
IME Sentence Mode	None
Smart Tags	





## Một số đặc trưng của bảng

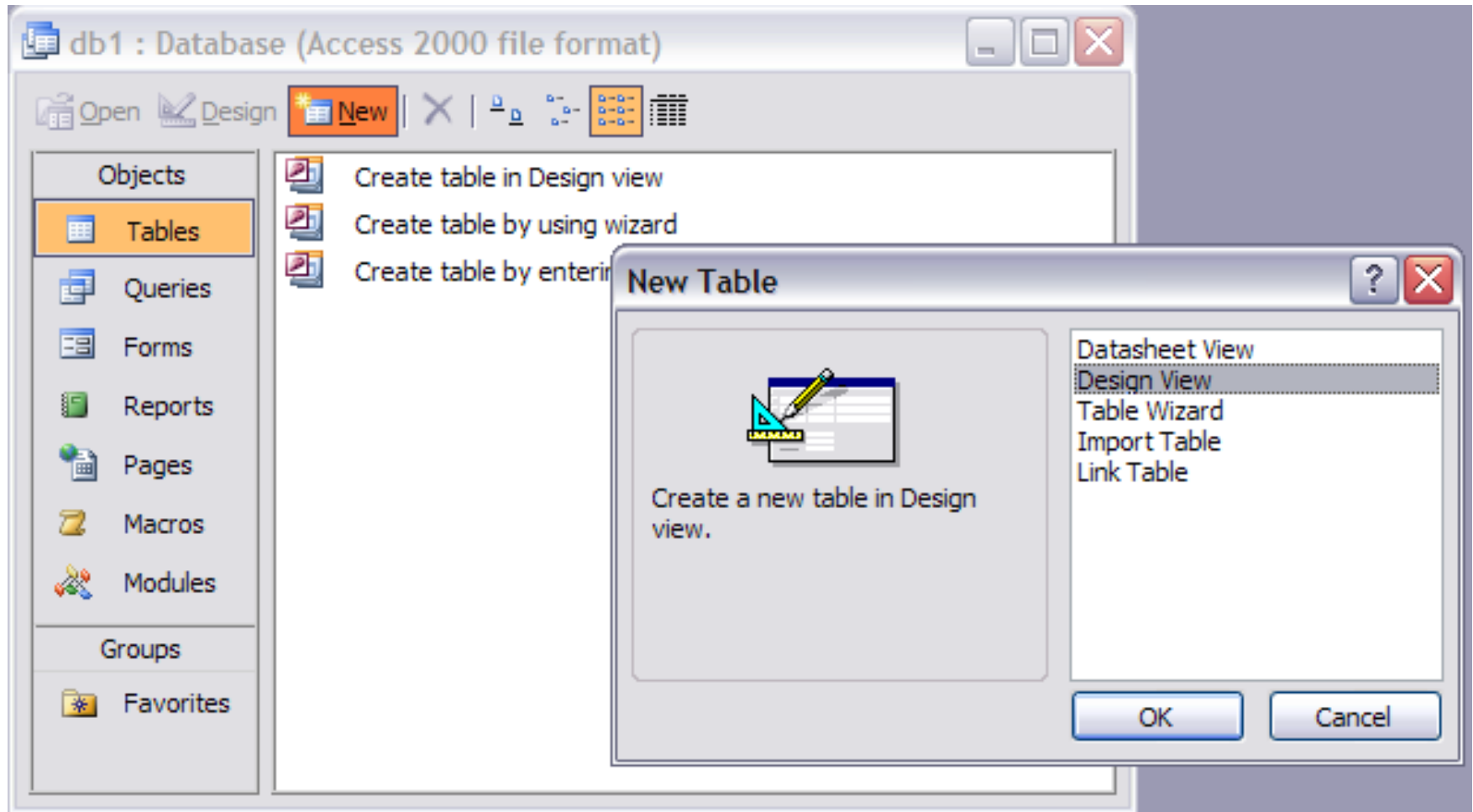
Thuộc tính	Trị tối đa
Số ký tự của tên bảng	64
Số ký tự của tên trường	64
Số trường trong một bảng	255
Kích thước bảng	1 gigabyte
Kích thước của trường OLE Object	1 gigabyte
Số các chỉ mục trong bảng	32
Số trường trong một chỉ mục	10
Số ký tự trong chuỗi dữ liệu	255



## Tạo bảng dữ liệu

- Có 5 cách để tạo bảng:
  - **Datasheet View:** chúng ta không cần thiết kế cấu trúc bảng trước. Access cung cấp sẵn 10 trường, 21 hàng để nhập các mẫu tin, chúng ta chỉ việc đặt lại tên trường.
  - **Design View:** cách này tạo ra một bảng dữ liệu đáp ứng đúng hơn với yêu cầu của bài toán.
  - **Table Wizard:** một trình thuật sĩ cung cấp một phương pháp đơn giản để thực hiện việc tạo bảng. Access cung cấp sẵn một số trường thường dùng để người sử dụng có thể chọn.
  - **Import Table:** Nhập khẩu từ một bảng có sẵn từ một CSDL khác.
  - **Link Table:** Liên kết tới các bảng ở các CSDL khác.

# Tạo bảng bằng Design View





# Màn hình Design View

DSSV : Table

	Field Name	Data Type	Description
	Masv	Text	
	Malop	Text	
	Hotensv	Text	
	Ngaysinh	Date/Time	
	Quequan	Text	
	Gioitinh	Yes/No	
	Hocbong	Number	

Field Properties

General **Lookup**

Field Size	50
Format	
Input Mask	
Caption	Họ tên SV
Default Value	
Validation Rule	
Validation Text	
Required	No
Allow Zero Length	Yes
Indexed	No
Unicode Compression	Yes
IME Mode	No Control
IME Sentence Mode	None
Smart Tags	

A field name can be up to 64 characters long, including spaces. Press F1 for help on field names.



## Đặt tên trường

- Có thể đặt tên trường tùy ý.
- Nhưng không quá 64 ký tự.
- Đặt tên trường ngắn gọn, dễ nhớ.



## Các kiểu dữ liệu của trường

Kiểu DL	Miền Giá Trị	Kích Thước
<i>Text</i>	Ký tự, chữ số	Tối đa 255 ký tự
<i>Number</i>	Số (nguyên, thực)	1, 2, 4 hay 8 byte
<i>Date/Time</i>	Ngày/Giờ.	8 byte
<i>Currency</i>	Giá trị tiền tệ.	8 byte
<i>Yes/No</i>	Giá trị logic: Đúng/Sai (Yes/No)	1 bit
<i>Memo</i>	Đoạn văn bản	Tối đa 64.000 ký tự
<i>OLE Object</i>	Các đối tượng liên kết và nhúng	Có thể lên đến 1Gb
<i>AutoNumber</i>	Số được Access tự động gán vào	
<i>Hyperlink</i>	Liên kết	
<i>Looup Wizard</i>	Chọn một giá trị trong danh sách các trị	



## Các thuộc tính, định dạng của trường

- Sau khi chọn kiểu dữ liệu, Access sẽ hiển thị các thuộc tính tương ứng với kiểu dữ liệu ở bên dưới.
- Ta có thể xác định: dữ liệu thực sự lưu trong bảng, kiểm tra độ chính xác của dữ liệu khi nhập vào, định dạng dữ liệu khi hiển thị, qui định cách nhập dữ liệu...
- Mỗi kiểu dữ liệu sẽ có các thuộc tính khác nhau.



# Thuộc tính, định dạng theo kiểu dữ liệu

Kiểu DL	TT/ĐD	Giải thích
<i>Text</i>	<b>Field Size</b> <b>Format</b> <b>Input Mask</b> <b>Caption</b> <b>Default Value</b> <b>Validation Rule</b> <b>Validation Text</b> <b>Required</b> <b>Allow Zero Length</b> <b>Indexed</b>	<b>Độ rộng trường</b> <b>Định dạng dữ liệu</b> <b>Mặt nạ nhập dữ liệu</b> <b>Tên trường</b> <b>Giá trị mặc định</b> <b>Điều kiện nhập dữ liệu</b> <b>Chuỗi thông báo nhập liệu</b> <b>Bắt buộc</b> <b>Cho phép chuỗi có độ dài 0</b> <b>Chỉ mục</b>





# Thuộc tính, định dạng theo kiểu dữ liệu

Kiểu DL	TT/ĐD	Giải thích
<i>Number</i>	<b>Field Size</b> <b>Format</b> <b>Decimal Places</b> <b>Input Mask</b> <b>Caption</b> <b>Default Value</b> <b>Validation Rule</b> <b>Validation Text</b> <b>Required</b> <b>Indexed</b>	<b>Độ rộng trường</b> <b>Định dạng dữ liệu</b> <b>Số chữ số thập phân</b> <b>Mặt nạ nhập dữ liệu</b> <b>Tên trường</b> <b>Giá trị mặc định</b> <b>Điều kiện nhập dữ liệu</b> <b>Chuỗi thông báo nhập liệu</b> <b>Bắt buộc</b> <b>Chỉ mục</b>



## Thuộc tính, định dạng theo kiểu dữ liệu

Kiểu DL	TT/ĐD	Giải thích
<i>Memo</i>	<b>Format</b> <b>Caption</b> <b>Default Value</b> <b>Validation Rule</b> <b>Validation Text</b> <b>Required</b> <b>Allow Zero Length</b>	<b>Định dạng dữ liệu</b> <b>Tên trường</b> <b>Giá trị mặc định</b> <b>Điều kiện nhập dữ liệu</b> <b>Chuỗi thông báo nhập liệu</b> <b>Bắt buộc</b> <b>Cho phép chuỗi có độ dài 0</b>



# Thuộc tính, định dạng theo kiểu dữ liệu

Kiểu DL	TT/ĐD	Giải thích
<i>Date/Time</i>	<b>Format</b> <b>Input Mask</b> <b>Caption</b> <b>Default Value</b> <b>Validation Rule</b> <b>Validation Text</b> <b>Required</b> <b>Indexed</b>	<b>Định dạng dữ liệu</b> <b>Mặt nạ nhập dữ liệu</b> <b>Tên trường</b> <b>Giá trị mặc định</b> <b>Điều kiện nhập dữ liệu</b> <b>Chuỗi thông báo nhập liệu</b> <b>Bắt buộc</b> <b>Chỉ mục</b>



## Thuộc tính, định dạng theo kiểu dữ liệu

Kiểu DL	TT/ĐD	Giải thích
<i>Currency</i>	<b>Format</b> <b>Decimal Places</b> <b>Input Mask</b> <b>Caption</b> <b>Default Value</b> <b>Validation Rule</b> <b>Validation Text</b> <b>Required</b> <b>Indexed</b>	<b>Định dạng dữ liệu</b> <b>Số chữ số thập phân</b> <b>Mặt nạ nhập dữ liệu</b> <b>Tên trường</b> <b>Giá trị mặc định</b> <b>Điều kiện nhập dữ liệu</b> <b>Chuỗi thông báo nhập liệu</b> <b>Bắt buộc</b> <b>Chỉ mục</b>



# Thuộc tính, định dạng theo kiểu dữ liệu

Kiểu DL	TT/ĐD	Giải thích
<i>Yes/No</i>	<b>Format</b> <b>Caption</b> <b>Default Value</b> <b>Validation Rule</b> <b>Validation Text</b> <b>Required</b> <b>Indexed</b>	<b>Định dạng dữ liệu</b> <b>Tên trường</b> <b>Giá trị mặc định</b> <b>Điều kiện nhập dữ liệu</b> <b>Chuỗi thông báo nhập liệu</b> <b>Bắt buộc</b> <b>Chỉ mục</b>
<i>Object</i>	<b>Caption</b> <b>Required</b>	<b>Tên trường</b> <b>Bắt buộc</b>



## Lưu ý

- Giữa các kiểu dữ liệu, có một số thuộc tính giống nhau, nhưng dữ liệu ở dạng nào thì thuộc tính sẽ phù hợp cho dạng đó.
- Ví dụ: trường Field Size trong kiểu Text và Number.
- ➔ Trong thực hành: phải xác định miền giá trị mà kiểu dữ liệu đó có. Từ đó, có thể xác định các giá trị cho thuộc tính một cách đúng đắn.



## Mô tả trường

- Đây là phần để ghi chú thích, chỉ rõ ý nghĩa dữ liệu trường đó lưu giữ.
- Chỉ góp phần làm cho các bảng rõ ràng, dễ hiểu, dễ đọc...
- Có thể có hoặc không.



# Ví dụ thiết kế bảng (Design)

**DSSV : Table**

Field Name	Data Type	Description
Malop	Text	
Hotensv	Text	
Ngaysinh	Date/Time	
Quequan	Text	
Gioitinh	Yes/No	
Hocbong	Number	

Field Properties

General    Lookup

Field Size	4
Format	
Input Mask	
Caption	Mã lớp
Default Value	
Validation Rule	
Validation Text	
Required	No
Allow Zero Length	Yes
Indexed	No
Unicode Compression	Yes
IME Mode	No Control
IME Sentence Mode	None
Smart Tags	

A field name can be up to 64 characters long, including spaces. Press F1 for help on field names.





## Ví dụ thiết kế bảng (Open)

	Mã SV	Mã lớp	Họ tên SV	Ngày sinh	Quê quán	Giới tính	Học bổng
▶ +	SV10	TI24	Trần Thu Dung	02/02/1982	Nghệ An	Nữ	234000
+	SV09	TA26	Nguyễn Thanh Nhân	05/06/1984	Thái Bình	Nữ	234000
+	SV07	VA25	Trần Lê Anh Thư	05/03/1984	Thanh Hóa	Nữ	234000
+	SV02	TI24	Hoàng Lan Hương	10/05/1981	TT Huế	Nữ	0
+	SV08	VA25	Đỗ Trung Hiếu	20/10/1983	Quảng Ngãi	Nam	0
+	SV06	HO25	Đặng Minh Quang	04/03/1983	Hà Tĩnh	Nam	0
+	SV05	HO25	Hữu Nghiêm	04/06/1982	Khánh Hòa	Nam	240000
+	SV03	TO24	Lê Mạnh Quân	15/05/1982	Nghệ An	Nam	180000
+	SV01	TI24	Nguyễn Trường Giang	12/03/1982	Quảng Bình	Nam	120000
*						Nam	0



# Quy định thuộc tính, định dạng cho trường



## Field Size

- **Kiểu Text:** Quy định độ dài tối đa của chuỗi.
- **Kiểu Number:**
  - *Byte:* 0..255
  - *Integer:* -32768..32767
  - *Long Integer:* -3147483648.. 3147483647
  - *Single:*  $-3,4 \times 10^{38} .. 3,4 \times 10^{38}$  (Tối đa 7 số lẻ)
  - *Double:*  $-1.797 \times 10^{308} .. 1.797 \times 10^{308}$  (Tối đa 15 số lẻ)
  - *Decimal Places:* Quy định số chữ số thập phân (Chỉ sử dụng trong kiểu Single và Double)
  - Đối với kiểu Currency mặc định Decimal places là 2.



## Format

- Kiểu Text: **<Phần 1>;<Phần 2>;<Phần 3>**
  - **<Phần 1>**: Chuỗi định dạng tương ứng trong trường hợp có chứa văn bản.
  - **<Phần 2>**: Chuỗi định dạng tương ứng trong trường hợp không chứa văn bản (Chuỗi rỗng "").
  - **<Phần 3>**: Chuỗi định dạng tương ứng trong trường hợp null (Khi chưa nhập dữ liệu).



## Các ký tự định dạng

Ký tự	Công dụng	Ví dụ
@	Bắt buộc phải nhập ký tự văn bản	
&	Không bắt buộc	
<	Chuyển thành in thường	ABC → abc
>	Chuyển thành in hoa	Abc → ABC
“Chuỗi ký tự”	Chuỗi ký tự giữa 2 dấu nháy	
\<ký tự>	Ký tự nằm sau dấu \	
[black] [white] hoặc [<số>] từ 0-56	Màu	[Green]

## Format

- Kiểu Number và Currency

Dạng	Dữ liệu hiển thị
General number	3456.789
Currency	\$3,456.79
Fixed	3456.79
Standard	3,456.79
Percent	123,00%
Scientific	3.46E+03



## Format

- Do người sử dụng định dạng:
  - **<Phần 1>;<Phần 2>;<Phần 3>;<Phần 4>**
    - *Phần 1*: Chuỗi định dạng tương ứng trong trường hợp số dương.
    - *Phần 2*: Chuỗi định dạng tương ứng trong trường hợp số âm.
    - *Phần 3*: Chuỗi định dạng tương ứng trong trường hợp số bằng zero.
    - *Phần 4*: Chuỗi định dạng tương ứng trong trường hợp Null.



## Các ký tự định dạng

Ký hiệu	Mô tả
. (period)	Dấu phân cách thập phân
, (comma)	Dấu phân cách hàng ngàn
0	Hiển thị chữ số hoặc số 0
#	Hiển thị chữ số (không bắt buộc)
%	Tính phần trăm (x100)
\$	Hiển thị ký tự \$
E- hoặc e- E+ hoặc e+	Ký pháp khoa học. Phải dùng với các đại diện chứa số





## Kiểu Date/Time

Dạng	Ngày giờ được hiển thị
General Date	1/31/92 4:30:00 PM
Long Date	Friday, January 31, 1992
Medium Date	31 - Jan - 1992
Short Date	1/31/92
Long Time	4:30:00 PM
Medium Time	04:30 PM
Short Time	16:30



## Kiểu Date/Time (tt)

Ký hiệu	Mô tả
: (colon)	Dấu phân cách giờ
/	Dấu phân cách ngày
D	Ngày trong tháng (1 – 31)
DD	Ngày trong tháng (01 – 31)
DDD	Ngày trong tuần (Sun – Sat)
DDDD	Ngày trong tuần (Sunday – Saturday)
W	Ngày trong tuần (1 – 7)
WW	Tuần trong năm (1 – 54)
M	Tháng trong năm (1 – 12)
MM	Tháng trong năm (01 – 12)
MMM	Tháng trong năm (Jun – Dec)



## Kiểu Date/Time (tt)

Ký hiệu	Mô tả
MMMM	Tháng trong năm (January – December)
Q	Quý trong năm (1 – 4)
Y	Số ngày trong năm (1 – 365)
YY	Năm (01 – 99)
YYYY	Năm (1900 - 9999)
H	Giờ (0-23)
HH	Giờ (00-23)
N	Phút (0-59)
NN	Phút (00-59)
S	Giây (0-59)
SS	Giây (00-59)



## Ví dụ

Ký hiệu	Mô tả
<b>Ddd/mmm/d/yyyy</b>	<b>Mon/Jun/2/1998</b>
<b>Mm/dd/yyyy</b>	<b>01/02/1998</b>



## Kiểu Yes/No

- Access có định dạng sau:
  - Yes/No                    ⇔            Đúng/Sai                    ⇔            -1/0
  - True/False                ⇔            Đúng/Sai                    ⇔            -1/0
  - On/Off                      ⇔            Đúng/Sai                    ⇔            -1/0
- Do người sử dụng định dạng:
  - **<Phần 1>;<Phần 2>;<Phần 3>**
    - *Phần 1*: bỏ trống.
    - *Phần 2*: cho giá trị trường đúng.
    - *Phần 3*: cho giá trị trường sai.



## Ví dụ

Định dạng	Hiển thị	
	Trường hợp True	Trường hợp False
;"Nam";"Nữ"	Nam	Nữ
;"Có";"Không"	Có	Không



## Một số ví dụ định dạng do người sử dụng

Định dạng	Giá trị	Hiển thị
<i>"Tài khoản "0000</i>	123	<i>Tài khoản 0123</i>
<i>mmm yyyy</i>	31-Jan-92	<i>Jan 1992</i>
<i>hh:mm "giờ"</i>	4:00 PM	<i>16:00 giờ</i>



## Ví dụ (tt)

Định dạng NSD	Hiển thị
0;(0);"Không có DL"	Số dương thể hiện bình thường, số âm được đặt trong dấu (), trường rỗng thì hiển thị "không có DL".
+0.0;-0;0.0	Thêm dấu cộng ở đầu số dương và dấu trừ ở đầu số âm. Nếu trường không có DL thì biểu diễn là 0.
@;"Chuỗi rỗng"; "Không có DL"	Văn bản trong trường thể hiện bình thường, nhưng nếu chuỗi rỗng thì hiển thị "Chuỗi rỗng", nếu trường không có DL.
;"Nam"[Red];"Nữ"[Blue ]	Nếu trường có giá trị "Yes" thì sẽ hiển thị chữ "Nam" màu đỏ, ngược lại sẽ hiển thị chữ "Nữ" màu xanh.





## Các ký tự dùng cho các kiểu dữ liệu

Ký hiệu	Mô tả
Space	Hiển thị ký tự trắng như bình thường.
“ABC”	Hiển thị chuỗi ABC.
!	Canh trái thay vì canh phải.
[color]	Đặt màu hiển thị. Các màu có sẵn: Black, Blue, Green, Red, Cyan, Magenta, Yellow, White.



## Tạo mặt nạ nhập dữ liệu (Input Mask)

- Nếu tất cả các dữ liệu trong trường có cùng dạng thức, bạn có thể đơn giản hóa công việc bằng cách tạo một mẫu để nhập dữ liệu, gọi là **mặt nạ dữ liệu**.
- Dùng mặt nạ nhập liệu bạn không phải gõ các ký tự định dạng khi nhập dữ liệu, hệ thống sẽ tự động trình bày các ký tự này trong trường khi bạn nhập dữ liệu.



## Tạo mặt nạ nhập dữ liệu (tt)

- Mặt nạ còn có công dụng khác là dữ liệu nhập vào phải đúng thì Access mới chấp nhận đưa vào trường.
- Ngoài ra, mặt nạ còn có chức năng bảo vệ, che dấu thông tin những thông tin quan trọng
  - Giả sử tạo mặt nạ kiểu password để khi nhập vào thay vì hiển thị ký tự thật thì lại hiển thị ký tự (\*)



## Các ký tự sử dụng trong mặt nạ

Mặt nạ	Ý nghĩa
0	Nhập (0..9), bắt buộc nhập (không cho phép + và -)
9	Nhập (0..9), Không bắt buộc nhập (không cho phép + và -), chấp nhận khoảng trắng
#	Chữ số hoặc khoảng trắng, không bắt buộc, chấp nhận + hoặc -
L	chữ cái (A-Z), bắt buộc nhập
?	Chỉ nhập mẫu tự (A-Z), không bắt buộc nhập
A	Chữ cái hoặc chữ số, bắt buộc nhập
a	Chữ cái hoặc chữ số, không bắt buộc nhập



## Các ký tự sử dụng trong mặt nạ (tt)

Mặt nạ	Ý nghĩa
<b>&amp;</b>	Bất kỳ ký tự nào và khoảng trắng, bắt buộc nhập.
<b>C</b>	Bất kỳ ký tự nào và khoảng trắng, không bắt buộc
<b>. , : ; - /</b>	Các ký hiệu dùng trong dạng Number và Date/Time
<b>&lt;</b>	Làm cho các ký tự thành in thường
<b>&gt;</b>	Làm cho các ký tự thành in hoa
<b>!</b>	Nhập vào từ phải sang trái
<b>\</b>	Ký tự đi sau sẽ được đưa thẳng vào dữ liệu



# Ví dụ

Mặt nạ nhập	Mẫu thể hiện
(000)000-0000	(206)555-0248
(999) 000-0000	(205)555-0248      ()555-0248
(000)AAA-AAAA	(206)555-TELE
000-0000a999	555-7000x201
0000-00-0000	0539-45-1999
>L?????L?000L0	GREETING237M3    MAY R 345B7
>L0L 0L0	T1F6M2
00000-9999	48115-            48115-6580
>L<????????????????	Paint            Beautiful
ISBN 0-&&&&&&&&-0	ISBN I-55615-507-7      BN 0-13-964262-5
>LL00000-0000	DB51392-0493



## Giá trị mặc định (Default Value)

- Giá trị mặc định là giá trị sẽ được hiển thị khi bắt đầu tiến hành nhập dữ liệu, thường các giá trị này hay xuất hiện.
- Giá trị mặc định có thể là *hằng* hoặc *biểu thức*
  - **Ví dụ:** Dùng hằng số 1 cho trường kiểu Number.  
Dùng hằng là “SP” cho trường kiểu Text.
- **Lưu ý:** Access tự động đặt giá trị mặc định cho các trường kiểu number là 0. Bạn có thể xóa số 0 này và thay bằng số khác.



## Thuộc tính Required (bắt buộc)

- Thuộc tính Required quy định một trường bắt buộc phải có.
- Nếu muốn trường phải có dữ liệu thì đặt thuộc tính required là **yes**.
- Sau khi thiết lập như vậy thì Access sẽ không chấp nhận trường đó có giá trị **Null** (Null – giá trị rỗng, tức không có số liệu).





## Giá trị rỗng và chuỗi rỗng

- Đối với trường kiểu Text và Memo còn có một thuộc tính liên quan là ***Allow zero length***.
- Thuộc tính này quy định trường đó có chứa ***chuỗi rỗng*** hay không (là chuỗi không chứa bất kỳ ký tự nào).
- Muốn cho trường nào đó chấp nhận chuỗi rỗng thì thiết lập thuộc tính Allow zero length là **Yes**.



## Giá trị hợp lệ (Validation)

- Kiểm tra tính hợp lệ của dữ liệu nhập vào của một trường ở thuộc tính Validation Rule.
- Khi dữ liệu nhập vào vi phạm quy tắc về tính hợp lệ thì sẽ thông báo ở thuộc tính Validation text.



## Lập biểu thức hợp lệ dữ liệu

- Bạn cần phát biểu quy tắc hợp lệ dữ liệu thành các biểu thức.
- Biểu thức xác định một giá trị như thế nào là hợp lệ.
  - Ví dụ: Biểu thức  $\geq 0$  hoặc “USA”, “UK”, “Canada”



## Một số toán tử

- Các toán tử quan hệ:  $>$ ,  $<$ ,  $>=$ ,  $<=$ ,  $=$ ,  $<>$ , **OR**, **AND**, **NOT**.
- Toán tử kiểm tra vùng: **Between...and...** ( $>= \dots$  And  $<= \dots$ )
- Toán tử thuộc vào: **IN** (<Danh sách các giá trị>)
- Kiểm tra phù hợp mẫu: **Like** “...” (toán tử này có thể dùng cho kiểu Date/Time, nhưng cũng đưa vào dưới dạng chuỗi)
- Toán tử kiểm tra giá trị rỗng: **is Null**, **is Not Null**
  - **Chú ý:** Nếu biểu thức kiểu ngày thì đặt giữa 2 dấu #.
  - Ví dụ:  $<\#30/04/75\#$   $\Leftrightarrow$  trước ngày 30/04/1975
  - $\text{IN}(\text{“SP”}, \text{“NL”}, \text{“KT”})$
  - Like “Truong Dai hoc\*”
  - Like “03/\*/2005”



## Một số ví dụ

Biểu thức hợp lệ	Dữ liệu phù hợp
$<>0$	Nhập một số khác 0
0 or $>100$	Giá trị là 0 hoặc lớn hơn 100
Like "K???"	Nhập 4 tối đa là 4 ký tự, và ký đầu là K
Like "*/12/95"	Nhập dữ liệu trong tháng 12/1995
$<\#1/1/92\#$	Phải nhập trước năm 1992
$>=\#1/1/91\#$ AND $<\#1/1/92\#$	Phải nhập trong năm 1991
IN ( "KH", "SP", "NL", YK")	Giá trị phải có trong danh sách liệt kê
Between $\#1/1/98\#$ And $\#15/3/98\#$	Nhập dữ liệu trong khoảng thời gian



## Thêm giá trị mới (New Value)

- Đối với kiểu Auto Number, thuộc tính này cho phép quy định cách thức mà trường tự động điền số khi thêm bản ghi mới.
- Có 2 giá trị cho thuộc tính này
  - Increase ⇔ Tăng dần
  - Random ⇔ Lấy số ngẫu nhiên



## Tạo lập chỉ mục (Indexed)

- Trong quá trình tìm kiếm hay sắp xếp trên một trường nào đó, bạn có thể tăng tốc độ các tác vụ này bằng cách tạo chỉ mục trên trường đó.
- Cách mà Access dùng chỉ mục trên bảng cũng giống như bạn tìm mục lục trong một cuốn sách.
- Muốn tạo chỉ mục cho một trường nào bạn chỉ cần thiết lập thuộc tính **Indexed** của trường đó.




## Các giá trị của thuộc tính Indexed

Giá trị	Ý nghĩa
No	Không hoặc xóa chỉ mục
Yes (OK duplicates)	Cho phép tạo chỉ mục trùng nhau
Yes (No duplicates)	Không cho phép tạo chỉ mục trùng nhau

- **Chú ý: Chỉ tạo được thuộc tính chỉ mục cho các trường không phải là kiểu Memo.**

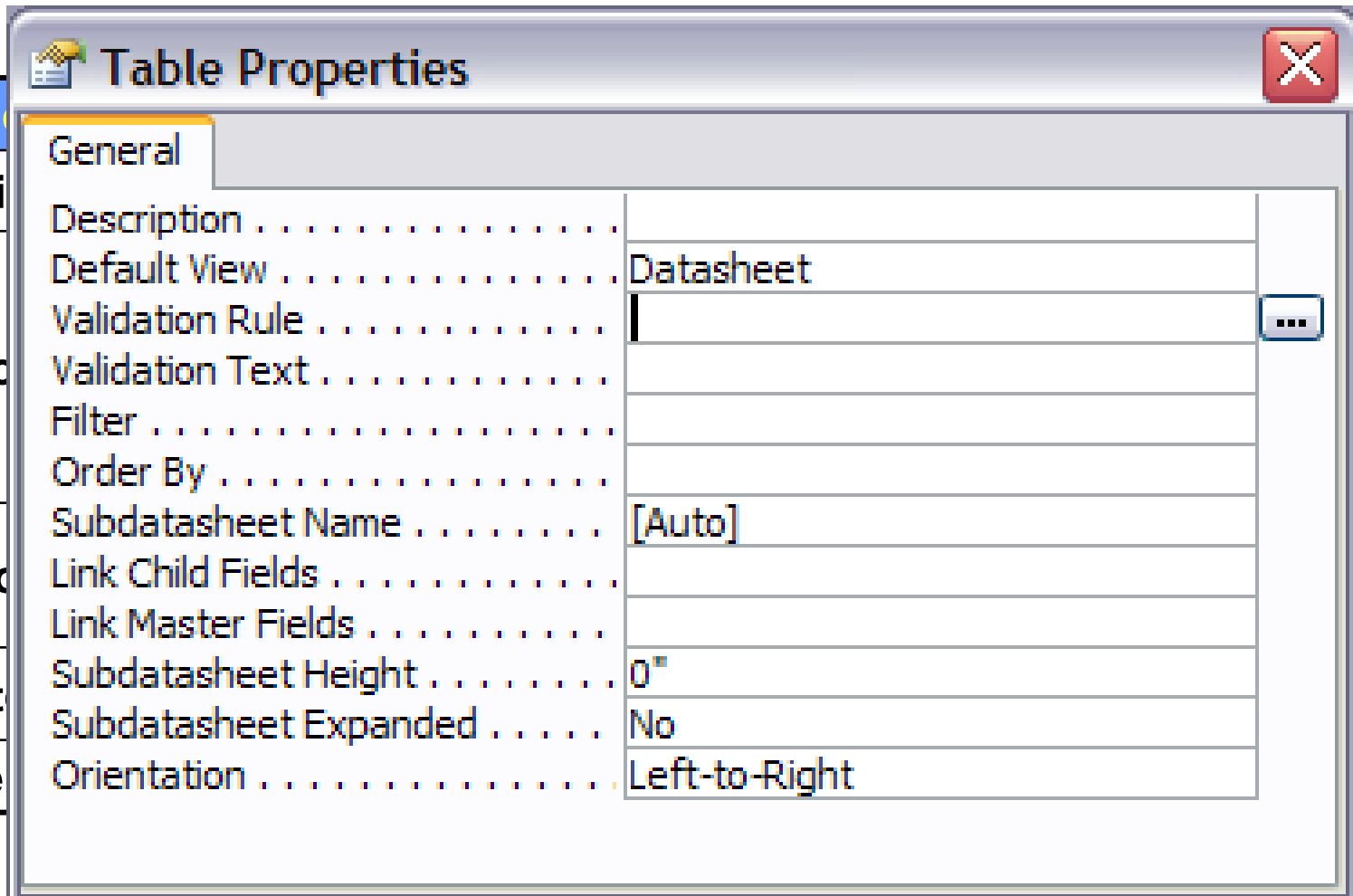


## Thuộc tính của bảng

- Mở bảng trong chế độ Design View.
- Thực hiện lệnh View/Properties hoặc Click biểu tượng  trên thanh công cụ. Lúc đó sẽ xuất hiện bảng Table Properties.
- Click chọn các mục và đặt thuộc tính cần thiết.
- Nếu đang thiết lập biểu thức hợp lệ Validation Rule mà muốn sử dụng công cụ xây dựng biểu thức của Access thì click chọn nút  bên cạnh và chọn Expression Builder.
- Thực hiện lệnh File/Save hoặc click nút  trên thanh công cụ để lưu lại những thay đổi này.

# Giải thích một số thuộc tính của bảng

Các thuộc tính
Descri
Validatio
Validatio
Filt
Orde



**Table Properties**

General

Description . . . . .	
Default View . . . . .	Datasheet
Validation Rule . . . . .	
Validation Text . . . . .	
Filter . . . . .	
Order By . . . . .	
Subdatasheet Name . . . . .	[Auto]
Link Child Fields . . . . .	
Link Master Fields . . . . .	
Subdatasheet Height . . . . .	0"
Subdatasheet Expanded . . . . .	No
Orientation . . . . .	Left-to-Right



## Khóa chính


- Tên tiếng Anh: **Primary Key**.
- **Khóa chính** là trường nhận các giá trị không trùng lặp (duy nhất).
- Chức năng:
  - Phân biệt các bản ghi với nhau.
  - Giúp cho việc tìm kiếm nhanh chóng hơn.
  - Thiết lập quan hệ giữa các bảng.
- Ta có thể thiết lập một hoặc nhiều trường làm khóa chính.
- Access tự động ngăn không cho ta nhập các giá trị trùng nhau vào các trường khóa chính.




## Chọn khóa chính

- Chọn trường không có giá trị trùng lặp (duy nhất).
- Các trường phải có kiểu dữ liệu không phải là Memo, OLE Object, Hyperlink, Lookup Wizard.


## Thiết lập khóa chính

- Thực hiện theo hai cách:
  - Chọn Edit/Primary Key, chọn trường khóa chính.
  - Click  trên thanh công cụ.
- Sau khi thiết kế bảng xong và thoát khỏi Design View, mà chưa thiết lập khóa chính thì Access sẽ hiển thị thông báo để hỏi chúng ta có muốn thiết lập khóa chính không.
  - Nếu muốn thì chọn Yes.
  - Còn không thì chọn No.
- Hủy bỏ khóa chính: thực hiện lệnh View / Indexes, trong hộp thoại này chọn và xóa đi những trường khóa đã thiết lập.

## Lưu bảng


- Có thể thực hiện 1 trong 2 cách sau:
  - Thực hiện lệnh *File/Save*.
  - Click nút  trên thanh công cụ của mục này (Table Design).
- Nếu bảng được lưu lần đầu tiên thì sẽ xuất hiện hộp thoại để nhập tên bảng vào, sau đó Click OK.
- Nếu sau khi thiết kế hay hiệu chỉnh bảng mà chưa lưu lại những thay đổi đó thì Access sẽ cho xuất hiện bảng thông báo để khẳng định lại những thay đổi trong bảng đó.
  - Đồng ý: Click Yes.
  - Không đồng ý: Click No.

## Hiệu chỉnh thiết kế bảng

- Từ cửa sổ Database, Chọn Table, chọn Bảng, click Design, Access sẽ cho hiển thị cửa sổ thiết kế bảng.
- Di chuyển trường:
  - Đưa con trỏ ra đầu trường cần di chuyển, nhấn và kéo đến vị trí mới.
- Chèn trường:
  - Chọn trường hiện thời sẽ là trường nằm sau trường được chèn vào.
  - Thực hiện *Edit/Insert Rows* hoặc Click phải chuột chọn Insert Rows hoặc chọn nút  trên thanh công cụ.






## Hiệu chỉnh thiết kế bảng (tt)

- Xóa trường:
  - Chọn trường cần xóa.
  - Thực hiện lệnh Edit/Delete Rows.
  - Hoặc chọn nút  trên thanh công cụ.
  - Hoặc click phải chuột, chọn Delete Rows.



## Bổ sung mẫu tin cho bảng

- Sau khi hoàn thành việc thiết kế cấu trúc của bảng, ta tiến hành nhập dữ liệu, tức là bổ sung các mẫu tin cho bảng.
- Chuyển sang chế độ Datasheet, mỗi hàng tương ứng với một mẫu tin.
- Ở đây có một số ký hiệu mà ta cần biết ý nghĩa của chúng:
  -  Mẫu tin hiện thời
  -  Mẫu tin đang nhập dữ liệu
  -  Mẫu tin mới



## Nhập dữ liệu cho các trường

- **Trường kiểu Date/Time:** Nhập theo quy định của Control Panel.
- **Trường kiểu Yes/No:** Nhập **Yes/No**, hoặc **True/False**, hoặc **On/Off**.
- **Trường kiểu Memo:** Nhập như trường kiểu Text; nhưng để xuống dòng dùng tổ hợp phím CTRL+ENTER.



## Nhập dữ liệu cho các trường (tt)

- **Trường kiểu OLE Object:** thực hiện lệnh *Insert\Object...*, xuất hiện hộp thoại *Insert Object*.
  - Trong đó, chọn *Create New* để tạo mới một đối tượng OLE từ hộp Object Type, hoặc chọn *Create from File* để chọn một file đã có sẵn.
- **Trường kiểu AutoNumber:** không nhập DL cho trường này, đây là trường được Access tự động đánh số.



## Thiết lập quan hệ giữa các bảng

- Thiết lập quan hệ giữa các bảng làm cho các bảng riêng biệt kết hợp lại với nhau và đưa ra những dữ liệu mà người sử dụng yêu cầu.
- Ví dụ: NHANVIEN và PHONGBAN.



## Các loại quan hệ



- Quan hệ Một - Nhiều (1 -  $\infty$ ):
  - Đây là quan hệ phổ biến nhất.
  - Mỗi mẫu tin trong bảng A (bên 1) có thể có nhiều mẫu tin tương ứng trong bảng B (bên  $\infty$ ), nhưng mỗi mẫu tin trong B có tối đa một mẫu tin trong A.
  - Ví dụ: LOP và HOCVIEN, NHANVIEN và PHONGBAN.
- Quan hệ Một - Một (1 - 1):
  - Mỗi mẫu tin bên bảng này tương ứng với một mẫu tin duy nhất bên kia và ngược lại.
  - Ví dụ: HOCVIEN và THEHOCVIEN



## Các loại quan hệ (tt)

- Quan hệ Nhiều – Nhiều ( $\infty - \infty$ )
  - Mỗi bản ghi trong bảng A có thể có nhiều hoặc không có bản ghi nào trong bảng B và ngược lại.
  - Khi gặp mối quan hệ nhiều – nhiều, để không gây sự trùng lặp và dư thừa dữ liệu thì chúng ta tách quan hệ nhiều – nhiều thành hai quan hệ một – nhiều bằng cách tạo ra một bảng phụ chứa khóa chính của 2 bảng đó.

## Thiết lập quan hệ

- Mở cửa sổ Relationship: *Tool\Relationship*, hoặc nhấn nút  trên thanh công cụ.
- Khi đó xuất hiện hộp thoại Show Table:
- Chọn các bảng hoặc truy vấn cần lập quan hệ trong Show Table như sau:
  - Chọn hiển thị là bảng (Table) hoặc truy vấn (Query), hoặc cả hai (Both).
  - Chọn bảng hoặc truy vấn, nhấn nút Add.
  - Nhấn nút Close để đóng hộp thoại.
- Nếu không có hộp thoại Show Table thì ta nhấp nút  hoặc chọn *Relationships/Show Table* để hiển thị hộp thoại này.



## Thiết lập quan hệ (tt)

- Muốn xóa một bảng khỏi quan hệ ta chọn bảng cần xóa rồi thực hiện lệnh *Relationship/Hide Table*.
- Thiết lập quan hệ:
  - Dùng chuột kéo trường cần thiết lập quan hệ của bảng này sang trường cần thiết lập quan hệ của bảng kia. Khi đó xuất hiện hộp thoại:
  - Để chọn loại quan hệ bắt tùy chọn *Enforce Referential Integrity* (Access tự động thiết lập quan hệ *One To One* (1-1) hoặc *One To Many* (1-∞)).
  - Chọn nút **Create**.
  - Có thể chọn **Join Type** để chỉ định tính cách quan hệ giữa hai bảng.








## Tham chiếu toàn vẹn


- Trong hộp thoại Relationships, nếu bật tùy chọn ***Enforce Referential Integrity*** thì ta sẽ thấy xuất hiện 2 mục:
  - *Cascade Update Related Fields*: Cho phép khi bạn sửa thông tin ở bảng bên một (đầu One) thì Access tự động sửa lại thông tin tương ứng ở bảng bên nhiều (đầu Many).
  - *Cascade Delete Related Records*: Khi bạn xoá một Record ở đầu One thì Access tự động xoá những Record tương ứng ở đầu Many.

## Xem và điều chỉnh các quan hệ

- Xem các quan hệ đã có sẵn:
  - Thực hiện lệnh Tools\Relationships hoặc nhấn nút  trên thanh công cụ.
  - Xem tất cả các quan hệ đã được tạo sẵn ta thực hiện Relationships \ Show all hoặc chọn nút  trên thanh công cụ.
  - Xem những quan hệ liên quan đến một bảng nào đó, thực hiện lệnh Relationships \ Show Direct hoặc chọn nút  trên thanh công cụ.




## Chỉnh sửa quan hệ đã có

- Thực hiện lệnh Tools \ Relationships hoặc chọn nút  trên thanh công cụ.
- Nhấp vào đường biểu diễn quan hệ nào muốn chỉnh sửa. Access sẽ hiển thị hộp thoại Relationships.
- Thiết lập các tùy chọn quan hệ, click OK.



## Xóa các quan hệ đã có

- Thực hiện lệnh Tools \ Relationships hoặc chọn nút  trên thanh công cụ.
- Chọn đường biểu diễn quan hệ mà bạn muốn xóa.
- Thực hiện lệnh Edit \ Delete hoặc nhấn phím Delete.