

Bài 1. Tổng quan CSS



Trước tiên để bắt đầu học CSS tôi xin giới thiệu chức năng của CSS và vì sao phải sử dụng CSS.

1. CSS là gì?

CSS viết tắt của Cascading Style Sheets được hiểu là ngôn ngữ định dạng website, thường dùng để định dạng các thành phần của 1 website như: font chữ, độ rộng, màu nền, vị trí,...

2. Vì sao phải sử dụng CSS?

Như chúng ta đã biết trong các thẻ HTML cũng có các thuộc tính định dạng, tuy nhiên các thuộc tính đó chưa đủ dùng cho 1 website thực tế, đồng thời với các bài toán lớn thì những thuộc tính đó xử lý rất khó. Ví dụ: Định dạng font chữ thành màu đỏ cho 100 đoạn văn, sửa nền cho 1000 thành phần,... trong trường hợp trên sử dụng CSS là 1 lựa chọn khôn ngoan nhất.

Đồng thời việc sử dụng chuẩn khi lập trình web sẽ có lợi cho việc SEO(Các bạn tự tìm hiểu thêm)

3. Dùng CSS ra sao?

Có lẽ đây là câu hỏi của tất cả chúng ta khi bắt đầu 1 ngôn ngữ lập trình nào đó.

Trong CSS có **3 cách khai báo** và **3 cách định dạng chính**

Khai báo:

- CSS trong: Được đặt trong cặp thẻ `<style></style>` và chứa ngay tại file .html cần xử lý(hay trong page cần định dạng).
- CSS nội tuyến: Là cách khai báo ngay tại thẻ html cần định dạng
`<p style="thuộc_tính:giá_trị;">Đây là đoạn văn sử dụng CSS nội tuyến(Phần in đậm)</p>`
- CSS ngoài: Là cách khai báo sử dụng 1 file .css ở bên ngoài file .html cần định dạng, trong thực tế chúng ta hay gặp dạng này vì 1 file .css có thể áp dụng cho nhiều page(nhiều file .html)

Bài 1. Tổng quan CSS



Cách làm: Tạo 1 file abc.css (với abc đặt tùy ý) bên trong đó bao gồm nội dung định dạng. Tại page(file .html) cần khai báo trong cặp thẻ <head></head> đoạn code sau:

```
<link rel="stylesheet" type="text/css" href="liên_kết_tới_file_css" />
```

Ví dụ: Chúng ta có 1 file index.html như sau

```
<html>
<head>
<title>Định dạng website với CSS</title>
<link rel="stylesheet" type="text/css" href="style.css" />
</head>
<body>
<p> Nội dung cần định dạng</p>
</body>
</html>
```

Bây giờ chúng ta tạo 1 file style.css ngang hàng với file index.html có nội dung

```
p{
color: red;
}
```

Sau khi làm như trên chúng ta sẽ có đoạn văn chuyển sang màu đỏ, đó là cách khai báo CSS.

Bây giờ chúng ta sẽ chuyển qua 1 phần nền tảng nhưng khá quan trọng.

Bài tập



Bài tập:

Định dạng nội dung bên dưới sang màu xanh(blue) biết thuộc tính định dạng màu là color sử dụng 3 cách: CSS trong, CSS ngoài, CSS nội tuyến.



Bài 2. Bộ chọn trong CSS

Để định dạng một website chúng ta cần xác định xem thành phần nào cần định dạng, khi sơn nhà nếu yêu cầu sơn nhà tắm không thể chui vào nhà ăn để sơn được. Vì vậy chúng ta có khái niệm bộ chọn(Tức là chọn thẳng nào để định dạng). Cấu trúc như sau(đối với CSS ngoài và CSS trong):

```
Bộ_chọn{  
Thuộc_tính: Giá_trị;  
}
```

Đối với CSS nội tuyến: <tên_thẻ style="thuộc_tính:giá_trị; thuộc_tính_2: giá_trị;"></tên_thẻ>

Chúng ta sẽ làm quen 3 bộ chọn chính trong CSS(Các bộ chọn khác tôi sẽ giới thiệu ở các bài sau):

1. Bộ chọn thẻ: Tức là chọn thẻ cần định dạng, ở đây Bộ_chọn chính là tên thẻ HTML chúng ta định dạng. Ví dụ cần định dạng đoạn sau trong file index.php <p>Đoạn chữ cần định dạng</p>

Chúng ta làm như sau:

- Cách 1: CSS ngoài, tạo 1 file style.css ngang hàng file index.html bên trên(đừng quên đoạn code liên kết hướng dẫn ở phần 3.1 nhé). Nội dung của file style.css như sau: **Bộ_chọn** chính là **thẻ html cần định dạng**, ở đây là thẻ p, thuộc tính giả sửa là định màu chữ nên thuộc tính là color, giá trị là red(màu đỏ). Như vậy áp dụng “công thức” bên trên ta có:

```
p{  
color:red;  
}
```

- Cách 2: CSS trong, ta chỉ việc thêm đoạn code định dạng trên vào cặp thẻ <style></style> rồi cho vào cặp thẻ <head></head>

Bài 2. Bộ chọn trong CSS



Minh họa: <http://jsbin.com/onerur/1/edit>

HTML ▼

```
<!DOCTYPE html>
<html>
<head>
<meta charset=utf-8 />

<title>Ví dụ về các bộ chọn trong CSS</title>

<!--CSS trong-->

<style>

  p{
    color:red;
  }

</style>

</head>
<body>

  <p> Đoạn chữ này cần định dạng</p>

</body>
</html>
```

2. Định dạng theo "công thức"

1. Xác định thành phần cần định dạng

Output

Đoạn chữ này cần định dạng



3. Xem kết quả

VIETSOURCE.NET



Bài 2. Bộ chọn trong CSS

2. Bộ chọn lớp:

Trong css, để định dạng các thành phần giống nhau chúng ta đặt các thẻ html thuộc tính class. Với các thẻ html có chung 1 giá trị class sẽ được định dạng như nhau. Giống như việc sơn 10 ngôi nhà ở khu chung cư A có tường màu xanh, cửa màu đỏ thì chúng ta sẽ đặt thuộc tính class="tên_lớp" cho các ngôi nhà đó.

Đối với bộ chọn lớp thì **Bộ_chọn** là **.tên_lớp** rồi tiếp tục thay vào "công thức"

Ví dụ: Định dạng màu chữ cho các lớp có class="mauchu", ở bên duoiwsi dòng thứ 2 không chuyển qua màu đỏ vì nó không có class="mauchu"

<http://jsbin.com/eveyus/1/edit>

HTML ▾	Output
<pre><!DOCTYPE html> <html> <head> <meta charset=utf-8 /> <title>JS Bin</title> <style> .mauchu{ color:red; } </style> </head> <body> <p class="mauchu">Đoạn 1 về bộ chọn lớp</p> <p>Đoạn 1 về bộ chọn lớp</p> <p class="mauchu">Đoạn 1 về bộ chọn lớp</p> Đoạn 1 về bộ chọn lớp </body> </html></pre>	<p>Đoạn 1 về bộ chọn lớp</p> <p>Đoạn 1 về bộ chọn lớp</p> <p>Đoạn 1 về bộ chọn lớp</p> <p>Đoạn 1 về bộ chọn lớp</p>

Bài 2. Bộ chọn trong CSS



3. Bộ chọn mã

Là bộ chọn trong đó các thẻ HTML cần định dạng được đặt 1 thuộc tính id="tên_id". Khác với class, ID chỉ có một(Tức là thẻ này không được dùng chung id của thẻ kia), vì vậy id được đặt cho các thành phần duy nhất trong HTML.

Ví dụ:

- Đúng: `<div id="header"> Nội dung</div>`
- Sai: Sai do 2 thành phần lại đặt cùng 1 id, trường hợp này chúng ta thay id thành class
`<div id="content"> Nội dung 1</div>`
`<p id="content"> Nội dung 2</p>`

Đối với bộ chọn mã khi xử dụng ta thêm ký tự thăng (#) vào trước tên_id

Ví dụ: <http://jsbin.com/ovuqok/1/edit>

```
HTML ▾
<!DOCTYPE html>
<html>
<head>
<meta charset=utf-8 />
<title>
VIETSOURCE.NET- CLB TIN HỌC TRƯỜNG ĐH CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG THÁI NGUYÊN
</title>
<style>
#noidung{
color: blue;
}
</style>
</head>
<body>
<p id="noidung">ID dùng cho các thành phần cần định dạng riêng, khác các thành phần khác
</p>
</body>
</html>
```

Output Run with JS Auto-r

ID dùng cho các thành phần cần định dạng riêng, khác các thành phần khác



Bài 1. Các bạn chuyển đoạn văn sau sang màu đỏ, biết thuộc tính là color màu là red(Sử dụng CSS với bộ chọn thẻ)

```
<div>VIETSOURCE.NET- DIỄN ĐÀN CHIA SẺ KIẾN THỨC LẬP TRÌNH</div>
```

Bài 2. Định dạng thành phần thứ 1 và thứ 3(Cặp thẻ div thứ 1 và cặp thẻ p thứ 3) bên dưới có độ rộng là 400px biết thuộc tính là width và màu nền là xanh(green), biết thuộc tính là background. (Sử dụng bộ chọn lớp)

```
<div> Ban Lập trình web</div>
```

```
<p> CLB Tin học</p>
```

```
<p> Trường ĐH Công nghệ thông tin& Truyền thông</p>
```

Bài 3. Định dạng thành phần thứ 2 bên dưới có màu nền đỏ, chữ màu trắng(biết giá trị màu trắng là white)

```
<p>BÀI 2. CSS SELECTOR <p>
```

```
<p> Chúc các bạn học tập tốt!</p>
```

HƯỚNG DẪN: 3 bài trên các bạn áp dụng “công thức” để định dạng. Để sẵn sàng cho các bài học sau các bạn cần nắm rõ kiến thức về các bộ chọn trong CSS và cách khai báo CSS(Cách dùng).

Chúc các bạn học tập tốt!

Thân,

Vũ Công Tịnh

Bài 3. Tìm hiểu về background trong CSS



Trước tiên, các bạn cần biết background là **1 tập các thuộc tính** trong css sử dụng để định dạng nền cho website. Bây giờ chúng ta sẽ tìm hiểu từng thuộc tính.

1. Background-color:

- Là thuộc tính sử dụng để định dạng màu nền cho website
- Giá trị:

+ Màu theo tên tiếng anh: red, green, blue, white, black,...

+ Là mã màu theo hệ hex. Tìm hiểu tại <http://vietsource.net/bang-ma-mau.html>

Ví dụ:

```
div{
    background-color: red;
}
```

Hoặc `background-color: #228654;`

2. Background-image

- Là thuộc tính định ảnh nền cho website
- ```
background-image: url('liên_kết_ảnh_nền');
```

Ví dụ:

```
div{
 background-image: url('images/vietsource.png');
}
```

# Bài 3. Tìm hiểu về background trong CSS



## 3. Background-repeat

- Xác định việc lặp lại của ảnh nền trong trường hợp ảnh nền nhỏ hơn kích thước phần định dạng
- Giá trị:
  - + Repeat: Lặp lại nếu ảnh nhỏ hơn kích thước phần cần định dạng
  - + No-repeat: Không lặp.
  - + Repeat-x: Lặp theo chiều ngang
  - + Repeat-y: Lặp theo chiều dọc website.

Ví dụ: <http://jsbin.com/ulabuv/2/edit>

```
<meta />
<head>
<meta charset=utf-8 />
<title>JS Bin</title>
<style>
 #doan1{
 background-image: url('http://vietsource.net/banner.png');
 background-repeat: repeat-x
 }
 #doan2{
 background-image: url('http://vietsource.net/banner.png');
 background-repeat: no-repeat
 }
</style>
</head>
<body>
<p id="doan1">
 Thông tin được đưa ra khi Bộ chỉ huy tối cao của quân đội Triều Tiên tuần trước cũng đơn phương công bố hiệp ước đình chiến năm 1953 không còn giá trị và trùng vào thời điểm Mỹ và Hàn Quốc tiến hành cuộc tập trận chung thường niên "Giải pháp Thèn chốt" quy mô lớn.
<p id="doan2">
 Theo báo chí Triều Tiên, sau khi lệnh được đưa ra, các cơ quan của đảng, các hiệp hội của người lao động ngay lập tức đã tổ chức các cuộc họp khẩn trên khắp cả nước và họ đã đảm nhận các vị trí trên chiến trường. Các đơn vị dân quân gồm công nhân và nông dân đã được triển khai. Ngoài ra, cũng có nhiều r
 đăng ký gia nhập quân đội và các binh sỹ đã nghỉ hưu cũng được huy động.
</p>
</body>
</html>
```

Output

Run with JS    Auto-run JS

Thông tin được đưa ra khi Bộ chỉ huy tối cao của quân đội Triều Tiên tuần trước cũng đơn phương công bố hiệp ước đình chiến năm 1953 không còn giá trị và trùng vào thời điểm Mỹ và Hàn Quốc tiến hành cuộc tập trận chung thường niên "Giải pháp Thèn chốt" quy mô lớn.

Theo báo chí Triều Tiên, sau khi lệnh được đưa ra, các cơ quan của đảng, các hiệp hội của người lao động ngay lập tức đã tổ chức các cuộc họp khẩn trên khắp cả nước và họ đã đảm nhận các vị trí trên chiến trường. Các đơn vị dân quân gồm công nhân và nông dân đã được triển khai. Ngoài ra, cũng có nhiều r  
đăng ký gia nhập quân đội và các binh sỹ đã nghỉ hưu cũng được huy động.

## Bài 3. Tìm hiểu về background trong CSS



### 4. Background-attachment

- Định kiểu hiển thị của nền.
- Giá trị:
  - + Fixed: Cố định. Khi nội dung quá dài mà nền nhỏ thì khi kéo thanh cuộn dọc, nền website sẽ chạy theo thanh cuộn.
  - + Scroll: Mặc định khi kéo thanh cuộn dọc nền vẫn giữ nguyên

### 5. Background-position

- Định vị trí nền của website
- Giá trị:
  - + Left: Bắt đầu từ bên trái
  - + Right: Bắt đầu từ bên phải
  - + Top: Bắt đầu từ trên.
  - + Bottom: Bắt đầu từ dưới
  - + Có thể dùng đơn vị % để định vị trí.

Ví dụ:

```
p{
 background-position: top left;
}
```

\* **Viết tắt:** Để tối ưu code và dễ cho quá trình thay đổi, sửa chữa chúng ta sử dụng cách viết tắt sau:

*Background: color image repeat attachment position;*

**Ví dụ:** *p{ background: red url("banner.png") no-repeat fixed top left;}*

# Bài tập



Bài tập: Thiết kế website như hình bên dưới sử dụng 3 cách khai báo: CSS trong, CSS ngoài, CSS nội tuyến. Sử dụng ảnh nền: <http://i.imgur.com/D7MBDfU.png>

Output Run with JS Auto-run JS

Nội dung này sẽ rất dài, sử dụng css định dạng khi kéo thanh cuộn dọc nền vẫn chạy theo( Nền như hình bên)					
Nội dung này sẽ rất dài, sử dụng css định dạng khi kéo thanh cuộn dọc nền vẫn chạy theo( Nền như hình bên)					
Nội dung này sẽ rất dài, sử dụng css định dạng khi kéo thanh cuộn dọc nền vẫn chạy theo( Nền như hình bên)					
Nội dung này sẽ rất dài, sử dụng css định dạng khi kéo thanh cuộn dọc nền vẫn chạy theo( Nền như hình bên)					
Nội dung này sẽ rất dài, sử dụng css định dạng khi kéo thanh cuộn dọc nền vẫn chạy theo( Nền như hình bên)					
Nội dung này sẽ rất dài, sử dụng css định dạng khi kéo thanh cuộn dọc nền vẫn chạy theo( Nền như hình bên)					
Nội dung này sẽ rất dài, sử dụng css định dạng khi kéo thanh cuộn dọc nền vẫn chạy theo( Nền như hình bên)					
Nội dung này sẽ rất dài, sử dụng css định dạng khi kéo thanh cuộn dọc nền vẫn chạy theo( Nền như hình bên)					
Nội dung này sẽ rất dài, sử dụng css định dạng khi kéo thanh cuộn dọc nền vẫn chạy theo( Nền như hình bên)					

Yêu cầu:  
- Sử dụng ảnh nền:  
<http://i.imgur.com/D7MBDfU.png>  
- Màu nền green  
- Ảnh nền căn như hình  
- Khi kéo thanh cuộn nền vẫn chạy theo

VIETSOURCE.NET

# Bài 4. Font trong CSS



## 1. Font-family

- Là thuộc tính định dạng font chữ cho website
- Giá trị là tên các font chữ.

Ví dụ: font-family: Arial, Verdana;

Có thể thêm nhiều font để tránh việc máy khách xem website không có font như chúng ta khai báo. Thứ tự ưu tiên các font là từ trái qua phải.

Ở ví dụ trên nếu máy người xem không có font Arial thì website sẽ được định dạng font Verdana.

Đối với các font có khoảng trắng thì phải đặt trong cặp dấu nháy kép ( “ )

Ví dụ: font-family: “Times New Roman”,Arial, Verdana;

## 2. Font-style

- Định kiểu chữ: Normal( bình thường), *Italic( in nghiêng)*, *Oblique( in xiên)*

## 3. Font-size

- Định kích thước cho chữ, giá trị là các giá trị độ dài, có thể sử dụng các đơn vị: px, em,...
- Tham khảo đơn vị trong CSS tại: <http://vietsource.net/don-vi-trong-css.html>

## 4. Font-variant

- Định dạng 1 số kiểu chữ như:  
small-caps, normal

HTML ▾

```
<!DOCTYPE html>
<html>
<head>
<meta charset=utf-8 />
<title>JS Bin</title>
<style>
 p{
 font-variant: small-caps;
 }
</style>
</head>
<body>
 <p> Định dạng chữ small-caps trong Font-variant</p>
</body>
</html>
```

Output

ĐỊNH DẠNG CHỮ SMALL-CAPS TRONG FONT-VARIANT

## Bài 4. Font trong CSS



### 5. Font-weight

- Định dạng kiểu in đậm (**bold**), in thường (normal) cho 1 thành phần nào đó

Ví dụ: 

```
p{
 font-weight: bold
}
```

- **Font thu gọn:**

Là cách viết tắt các thuộc tính về font, giúp cho nội dung code ngắn gọn, dễ tùy chỉnh:

*Font: style variant weight size/line-height family*

Ví dụ:

*font: italic smallcaps normal 13px Arial, "Time New Roman";*

**BÀI TẬP:** Sử dụng mẫu HTML như bên dưới, sử dụng css để định dạng sao cho kết quả như hình bên phải. Làm tối thiểu 3 cách

HTML ▾	Output
<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;head&gt; &lt;meta charset=utf-8 /&gt; &lt;title&gt;JS Bin&lt;/title&gt; &lt;/head&gt; &lt;body&gt;   &lt;p class="font" div="font_2"&gt;     Đây là 1 ví dụ về Font   &lt;/p&gt; &lt;/body&gt; &lt;/html&gt;</pre>	<p><b>ĐÂY LÀ 1 VÍ DỤ VỀ FONT</b></p>



## Bài 5. Text trong CSS

### 1. Color

- Định dạng màu chữ trong CSS

- Giá trị:

- + Tên màu tiếng Anh: gray, blue, green,...

- + Bảng mã màu hệ 16: Gồm ký tự # và chuỗi 6 ký tự liền kề. Ví dụ: #FF7692. Xem thêm:

- <http://vietsource.net/bang-ma-mau.html>

- + Mã màu RGB.

- + ...

### 2. Text-indent

- Định khoảng thụt vào của 1 dòng bất kỳ

### 3. Text-align

- Định vị trí của 1 đoạn văn bản.

- Giá trị:

- + Left: Căn trái

- + Center: Căn giữa

- + Right: Căn phải.

- + Justify: Căn đều.

### 4. Letter-spacing

- Định khoảng cách giữa các ký tự

Ví dụ:

VIETSOURCE.NET

letter-spacing: 10px;





## Bài 5. Text trong CSS

### 5. Word-spacing

- Định khoảng cách giữ các từ.

### 6. Text-decoration

- Underline: Chữ gạch chân.
- Line-through: Chữ gạch xiên
- Overline: Chữ có gạch ngang trên đầu
- Blink: Chữ nháy.
- None: Không có 4 giá trị trên

Ví dụ:

<http://jsbin.com/irumop/1/edit>

```
<!DOCTYPE html>
<html>
<head>
<meta charset=utf-8 />
<title>JS Bin</title>
<style type="text/css">
 /* Loại bỏ chấm tròn phía trước các danh sách(list)*/
 li{
 list-style: none;
 }
 /* Sử dụng text-decoration để bỏ gạch chân trong liên kết do thẻ a tạo ra*/
 a{
 text-decoration:none;
 }
</style>
</head>
<body>
<p>VIETSOURCE.NET</p>

Trang chủ
Diễn đàn
Tài liệu

<!-- Text-decoration thường thấy khi làm menu, dùng để loại bỏ gạch chân-->
</body>
</html>
```

VIETSOURCE.NET

[Trang chủ](#)  
[Diễn đàn](#)  
[Tài liệu](#)

## Bài 5. Text trong CSS



### 7. Text-transform

- Uppercase: Định chữ in hoa
- Lowercase: Định chữ thường
- Capitalize: Định kiểu in hoa chữ cái đầu.

**BÀI TẬP:** Thiết kế website như hình bên:

Yêu cầu:

- Sử dụng CSS để định dạng tất cả các thành phần của website
- Đối với text phải viết bằng chữ thường và sử dụng CSS để định dạng.

Vietsource.net - Diễn Đàn Chia Sẻ Kiến Thức Lập Trình

**HỌC PHP KHÔNG KHÓ**

ban lập trình web- CLB TIN HỌC

## Bài 6. Pseudo- Phần tử giả



Pseudo( hay dịch là Phần tử giả): Là 1 phần quan trọng mà hầu hết các website đều có trong các liên kết( Links) dùng để quy định cách hiển thị cho liên kết khi thực thi 1 hành động nào đó. Như:

### 1. Hover- di chuột qua

- Như đã nói bên trên do Pseudo áp dụng với liên kết nên trước tiên chúng ta cần phải xác định thành phần tác động đến là thẻ <a>
- Ví dụ:

```
a:hover{ color: red; background: white; text-decoration: blink;}
```

Tức là: Khi di chuột qua liên kết thì nội dung bên trong cặp thẻ {...} sẽ hoạt động, cụ thể: Khi di chuột qua liên kết sẽ có màu đỏ và nền trắng và nháy.

### 2. Link

- Định các thuộc tính cho liên kết, mặc định màu của liên kết là xanh chúng ta có thể chuyển qua màu khác.
- Ví dụ: *a:link { color: gold; }*

### 3. Active

- Định các thuộc tính cho liên kết khi liên kết đó hoạt động( khi nhấn giữ chuột vào liên kết đó), tuy nhiên trường hợp này thực tế ít khi sử dụng.
- Ví dụ: *a:active { background: gray; }*

### 4. Visited

- Định các thuộc tính khi liên kết đã được ghé thăm( Những liên kết chúng ta đã click ít nhất 1 lần)

# Bài 6. Pseudo- Phần tử giả



Ví dụ:

```
<title>JS Bin</title>
<style>
 li{
 list-style: none;
 }
 a{
 text-decoration: none;
 background: green;
 font-size: 25px;
 color: white
 }
 a:hover{
 background: white;
 color: black;
 }
</style>
</head>
<body>

<h2>Tin tức</h2>

 Giáo dục
 Kinh tế
 Văn hóa

```

Tin tức

- Giáo dục
- Kinh tế
- Văn hóa



## Bài 7- Box Model( Định dạng kiểu khối)

Box model: Định dạng kiểu khối( hay kiểu hộp) cho website. Bài này chúng ta sẽ tìm hiểu lần lượt 3 phần của Box model.

### 1. Border- Đường viền

- Trong border chúng ta có 3 thuộc tính

**a) Border-width:** Độ dày của đường viền.

- Cú pháp: border-width: apx bpx cpx dpx;

Với: a, b, c, d lần lượt là giá trị độ dày đường viền của các cạnh: bên trên, bên phải, bên dưới và bên trái( Theo thứ tự chiều kim đồng hồ)

Khi 1 trong 4 giá trị kia bị thiếu thì nó sẽ lấy giá trị độ dày của cạnh đối diện nó.

Ví dụ: border-width: 2px 3px 3px;

Như vậy: Cạnh trên độ dày là 2px, phải là 3px và dưới là 3px, như vậy bên trái bị thiếu nó sẽ lấy của cạnh đối diện là độ dày cạnh bên phải.

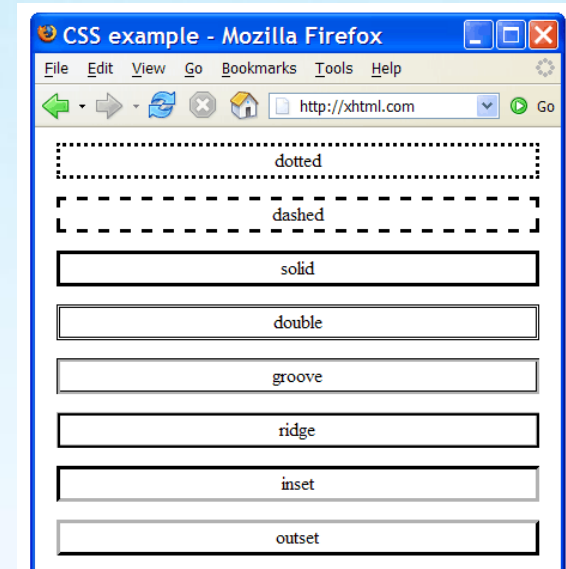
**b) Border-style:** Kiểu đường viền

- Chúng ta có 1 số giá trị về đường viền như hình bên

- Không nhất thiết phải định dạng cả 4 cạnh có viền mà có thể sử dụng 4 thuộc tính sau: border-top-style( viền trên), border-right-style( viền phải), border-bottom-style( viền dưới), border-left-style( viền trái).

**c) Border-color:** Màu đường viền

- Giá trị của border là các loại mã màu.





## Bài 7- Box Model( Định dạng kiểu khối)

Ví dụ:

```
p{
border-width: 2px;
border-style: solid;
border-color: blue;
}
```

Có thể viết tắt các thuộc tính về border như sau:

```
border: width style blue;
```

### 2. Padding- Biên

Là thuộc tính tạo khoảng cách đường viền và nội dung

```
HTML
:style>
p{
border: 3px solid green;
padding: 20px;
}
:/style>
:p> Padding sử dụng để định vị trí giữa nội dung và đường viền</p>
```

Output

Run with JS Auto-run JS

20px

Padding sử dụng để định vị trí giữa nội dung và đường viền

20px

Đối với padding chúng ta cũng có 4 thuộc tính: padding-top, padding-right, padding-bottom, padding-left

Sử dụng để căn chỉnh nội dung so với các viền tương ứng: trên, phải, dưới, trái.

Tương tự như border-width ở phần trên: Khi 1 thành phần bị thiếu nó sẽ lấy giá trị của thành phần đối diện.

Như ở ví dụ trên do chỉ có 1 giá trị: padding: 20px; nên khoảng cách từ nội dung tới đường viền đều bằng 20px. Các bạn sẽ để ý thấy hình trên nội dung cách đường viền bên phải “hình như” lớn hơn 20px, lí do nó như vậy là nội dung của chúng ta ít trong khi đó chiều rộng của border lấy theo kích thước chiều ngang trình duyệt( Các bạn viết nội dung dài ra để cho tràn xuống dòng dưới sẽ thấy kết quả).

Xem thêm : <http://bit.ly/vsnet-box-model>



## Bài 7- Box Model( Định dạng kiểu khối)

### 3. Margin- Lề

- Là thuộc tính sử dụng để căn chỉnh các thành phần với nhau hay 1 thành phần nào đó với lề( như hình).
- Tương tự Padding, Margin cũng có 4 thuộc tính để căn chỉnh 4 vị trí: top, right, bottom, left hay auto.

Ví dụ: Căn chỉnh website ra giữa

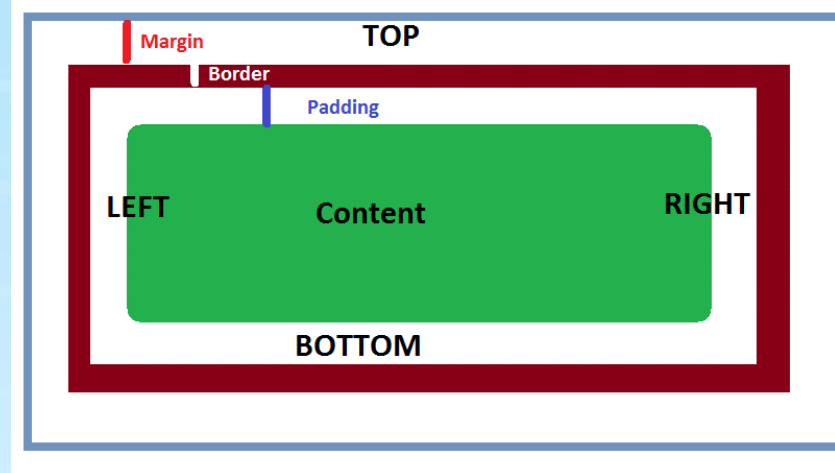
```
body{
margin: auto;
}
```

- Khi lập trình chúng ta thường sử dụng đoạn code sau để reset

```
*{
margin: 0px;
padding: 0px;
}
```

Như vậy: Toàn bộ các thành phần của website sẽ có margin và padding bằng 0, để tránh việc các trình duyệt hiển thị khác nhau.

Xem thêm: <http://bit.ly/vsnet-box-model>





## Bài 7- Box Model( Định dạng kiểu khối)

**BÀI TẬP:** Thiết kế website như hình bên dưới.



- Border: red
- Căn giữa website

- wrapper đặt width:960px, height:900px; background tùy ý
- header đặt width:100%, height:20%;
- content width:100%, height: 60%, background tùy ý
- footer width 100%, height:20%,

**VietSource.Net**



## Bài 8- Float & Clear



### 1. Float

- Là thuộc tính sử dụng để cố định thành phần của website về 1 phía: trái( left) hay phải( right).
- Thường sử dụng trong việc dàn trang, chia cột cho website.

Giá trị:

- Left: Dàn qua trái.
- Right: Dàn qua phải.
- None: Bình thường.

Bài toán: Sử dụng div kết hợp với CSS xây dựng website như hình dưới.

Hướng dẫn:

Xây dựng 2 cặp thẻ div tương ứng với 2 thành phần trong hình. Sử dụng float cho thành phần bên trái dàn sang bên trái.

sau khi làm xong các bạn có thể xem đáp án tại:

<http://jsbin.com/aputaj/1/edit>

Cột trái

Cột phải



## Bài 8- Float & Clear

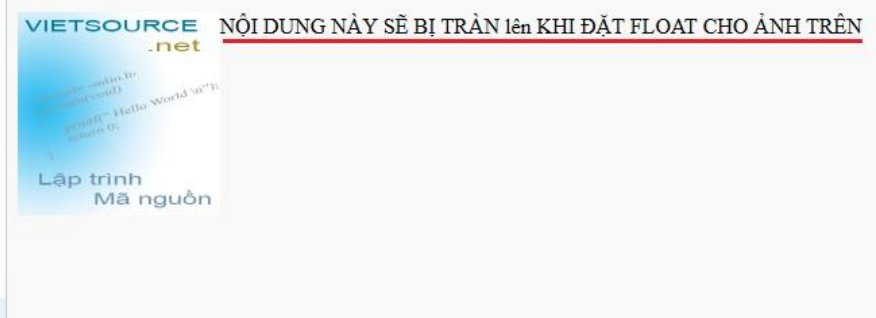
### 2. Clear

- Là thuộc tính sử dụng cho các thành phần đi cùng các thành phần sử dụng float. Sử dụng để định dạng cách tràn cho thành phần này khi thành phần trước nó đặt float và còn khoảng trống.
- Giá trị:
  - + Left: Tràn sang trái.
  - + Right: Tràn phải.
  - + Both: Không tràn, thường dùng để tránh tràn các thành phần trong website.

Ví dụ: <http://jsbin.com/ibaqev/1/edit>

```
<!DOCTYPE html>
<html>
<head>
<meta charset=utf-8 />
<title>JS Bin</title>
<style>
 img{ float: left;}
</style>
</head>
<body>

 <p>NỘI DUNG NÀY SẼ BỊ TRÀN lên KHI ĐẶT FLOAT CHO ẢNH TRÊN</p>
</body>
</html>
```



Khi đặt float cho ảnh thì phần bên phải ảnh còn khoảng trống nên các thành phần bên dưới sẽ tràn lên để lấp đầy chỗ trống đó. Để giải quyết vấn đề này chúng ta đặt 1 thành phần vào giữa 2 thành phần trên( giữa thẻ img và thẻ p) sau đó cho thành phần này clear: both.

Ví dụ:

`<br class="clear" />` hoặc `<div class="clear"></div>` hoặc ...

Sau đó sử dụng: `.clear{ clear: both;}` Xem kết quả: <http://jsbin.com/ibaqev/2/edit>

## Bài 8- Float & Clear



**Bài tập:** Thiết kế website như hình bên dưới sử dụng css( Không dung table), với tên các thành phần và kích thước cho trước. Xem thêm nội dung: <http://bit.ly/vsnet-float-clear>

The diagram shows a website layout with the following components and specifications:

- Header:** A solid blue bar at the top.
- Menu:** A light blue bar containing the text "Menu".
- Main Content Area:** A large yellow area divided into two columns. The left column is green and contains the text: "Trang chủ", "Giới thiệu", "Dịch vụ", and "Liên hệ".
- Footer:** A solid green bar at the bottom.

Specifications for the layout:

- **Border: red**
- **Căn giữa website**
- wrapper đặt width:960px, height:900px; background tùy ý
- header đặt width:100%, height:20%;
- content width:100%, height: 60%, background tùy ý
- footer width 100%, height:20%,

**VietSource.Net**



## Bài 9- Block, Inline

Trong bài này chúng ta sẽ tìm hiểu về Block và Inline trong CSS, tuy nhiên đây không phải thuộc tính mà là giá trị quy định cách hiển thị của 1 thành phần nào đó. Sử dụng trong thuộc tính display để quy định cách hiển thị.

- **Block** quy định các thành phần hiển thị theo từng khối riêng biệt, một số thẻ mặc định đã hiển thị theo khối như: p, ul, li, div, heading,... Tức là: Khi sử dụng các cặp thẻ này, mỗi nội dung ở 1 dòng riêng biệt không nằm cùng 1 dòng hay gọi là các khối. Hãy tưởng tượng như các cặp thẻ đó dù các bạn viết code trên cùng 1 dòng thì chúng cũng tự động xuống dòng, giống như việc xếp 10 cái hộp lên nhau, mỗi hộp là 1 khối.
- **Inline** quy định các thành phần nằm cùng 1 hàng ( hay 1 dòng), ví dụ như Menu trên website,...

Ngoài ra chúng ta còn 1 số giá trị sau: inline-block, inline-table, list-item, table-cell,...

Tham khảo thêm tại: [http://www.w3schools.com/cssref/pr\\_class\\_display.asp](http://www.w3schools.com/cssref/pr_class_display.asp)

Tham khảo video Block, Inline tại: <http://bit.ly/vsnet-display>

Ví dụ: <http://jsbin.com/evoxur/1/edit>

```
<!DOCTYPE html>
<html>
<head>
<meta charset=utf-8 />
<title>JS Bin</title>
<style>
 li
 list-style: none;
 display: inline
}
</style>
</head>
<body>

Menu 1
Menu 2
Menu 3

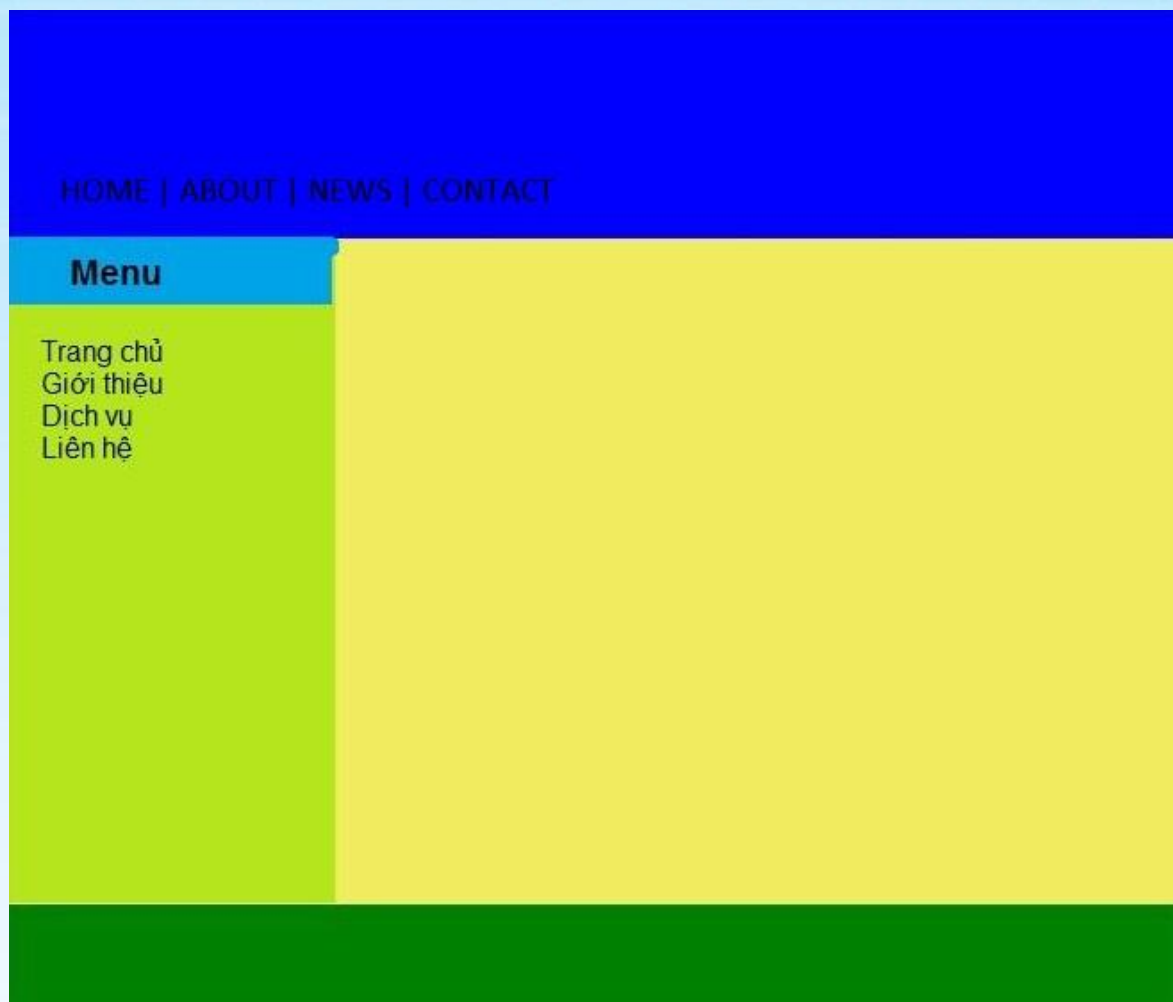
</body>
</html>
```

[Menu 1](#) [Menu 2](#) [Menu 3](#)

## Bài 9- Block, Inline



**Bài tập:** Thiết kế website như hình bên dưới( Nên làm lại, không nên sử dụng kết quả của Bài 8)



## Bài 10- Position



Như đã tìm hiểu bài trước về Float, chúng ta có thể sử dụng để dàn trang sang trái hoặc qua phải. Vậy muốn định 1 thành phần qua 1 vị trí khác chúng ta phải làm thế nào? Trong CSS chúng ta có thuộc tính Position sử dụng để định vị trí của thành phần này so với thành phần khác( Thành phần làm mốc) hoặc so với chính cửa sổ trình duyệt..

### 1. Fixed

- Sử dụng để cố định 1 thành phần so với cửa sổ trình duyệt, khi kéo thanh cuộn thì thành phần đó vẫn được cố định so với trình duyệt.

### 2. Relative

- Là cách cố định vị trí 1 cách tương đối của 1 thành phần nào đó so với vị trí bình thường( không định vị trí) của nó

### 3. Absolute

- Là 1 cách cố định 1 thành phần so với 1 thành phần khác 1 cách tuyệt đối. Khi sử dụng position thông thường thì thành phần mẹ phải đặt **Relative** để nó biết phải cố định tuyệt đối theo thành phần nào, nếu không tồn tại nó sẽ nhận <html> làm thành phần mẹ

### Bổ sung:

Đi kèm 3 thuộc tính trên chúng ta sử dụng thêm 4 thuộc tính: Top, Right, Bottom, Left với các giá trị tương ứng với cửa sổ trình duyệt( với Fixed), chính nó ( Relative), thành phần mẹ( Đối với Absolute).



## Bài 10- Position

Ví dụ: <http://jsbin.com/uvaros/1/edit>

```
HTML ▾
<!DOCTYPE html>
<html>
<head>
<meta charset=utf-8 />
<title>JS Bin</title>
<style>
 div{
 background: green;width: 100px;height: 100px;
 position: absolute;
 top: 50px;
 left: 50px;
 }
</style>
</head>
<body>
<div>Position</div>
</body>
</html>
```



Xem thêm : <http://bit.ly/vsnet-position>

Do 1 số thành viên VSNet yêu cầu làm TUT dạng văn bản để thuận tiện trong quá trình tham khảo, đồng thời sử dụng trong 1 số buổi sinh hoạt Ban lập trình web- CLB Tin học Trường ĐH CNTT& TT Thái Nguyên nên tôi viết lại những lý thuyết căn bản về HTML, CSS dưới dạng slide để các bạn tham khảo thêm . Những phần lý thuyết qua slide chưa phải là đầy đủ nhất nhưng hy vọng nó sẽ giúp ích được cho các bạn.

Toàn bộ video HTML, CSS: <http://bit.ly/vsnet-html-css>

Các bạn có thể tham khảo thêm 1 số tài liệu do tôi viết về 1 số kinh nghiệm lập trình web tại:

<http://vietsource.net>

<http://svictu.com>

Chúc các bạn học tập tốt!