

Bài 3

CSS – Cascading Style Sheets

Viện CNTT & TT

Nội dung

- Giới thiệu CSS
- Định nghĩa Style
- Sử dụng và Phân loại CSS
- Selector trong CSS và phạm vi ảnh hưởng

Nội dung

- Giới thiệu CSS
- Định nghĩa Style
- Sử dụng và Phân loại CSS
- Selector trong CSS và phạm vi ảnh hưởng

Giới thiệu về CSS

- CSS = Cascading Style Sheets
- Dùng để mô tả cách hiển thị các thành phần trên trang WEB
- Ban đầu HMTL
 - Tập các quy tắc cho phép NSD xem các trang Web
 - Không hỗ trợ nhà thiết kế có nhiều quyền chi phối thiết kế
- => Các trình duyệt bổ sung thêm các thẻ HTML riêng, không theo chuẩn
- W3C (World Wide Web Consortium) ra đời CSS cho phép hỗ trợ nhà thiết kế

Giới thiệu về CSS

- Style tiết kiệm thời gian
- CSS dễ thay đổi
- Sự nhất quán
- Khả năng chi phối đa dạng
- Tạo ra định dạng chung cho toàn bộ Web

Thiết kế và lập trình Web

```
<html>
<body>
<p>
    <span style="font-size: 14pt; color: blue"><strong>
        Web Design : </strong></span>
        <em><span style="color: red">Photoshop, Macromedia Flash </span></em>
    </span>
</p>
<p>
    <span style="font-size: 14pt; color: blue"><strong>
        Web Programming : </strong></span>
        <em><span style="color: red">ASP, PHP, JSP </span></em>
    </span>
</p>
<p>
    <strong><span style="font-size: 14pt; color: blue">
        Debugger : </span></strong><em><span
            style="color: red">Zend, VS.PHP, VS.NET 2005 </span></em>
    </span>
</p>
</body>
</html>
```

Without CSS

```
<html>
<head>
    <style type="text/css">
        .classSubjet { color:#0000FF;
            font-weight: bold; font-size: 14px;}
        .classTool { font-style: italic;
            color: #FF0000; font-size: 12px;}
    </style>
</head>
<body>
<p>
    <span class="classSubjet">Web Design : </span>
    <span class="classTool">Photoshop, Macromedia Flash </span>
</p>
<p>
    <span class="classSubjet">Web Programming : </span>
    <span class="classTool">ASP, PHP, JSP</span>
</p>
<p>
    <span class="classSubjet">Debugger : </span>
    <span class="classTool">Zend, VS.PHP, VS.NET 2005</span>
</p>
</body>
</html>
```



With CSS

Nội dung

- Giới thiệu CSS
- Định nghĩa Style
- Sử dụng và Phân loại CSS
- Selector trong CSS và phạm vi ảnh hưởng

Định nghĩa Style

Kiểu 1

```
<tag style =  
    "property1:value1;  
    property2:value2;  
    .....  
    propertyN:valueN;">...</tag>
```

Kiểu 2

```
SelectorName {  
    property1:value1;  
    property2:value2;  
    .....  
    propertyN:valueN; }  
  
<tag class = "SelectorName">  
    .....  
</tag>
```

Ví dụ:

```
<h1 style="  
    color : blue;  
    font-family : Arial;"> DHKHTN </h1>
```

Ví dụ:

```
.TieuDe1 {  
    color: red;  
    font-family: Verdana, sans-serif; }
```

```
<h1 class="TieuDe1"> DHBKHN </h1>
```

Định nghĩa Style – Ghi chú

- Giống Ghi chú trong C++
- Sử dụng /*Ghi chú*/
- Ví dụ :
- **SelectorName {**
 property1:value1; /*Ghi chú 1*/
 property2:value2; /*Ghi chú 2*/

 propertyN:valueN; }

Nội dung

- Giới thiệu CSS
- Định nghĩa Style
- Sử dụng và Phân loại CSS
- Selector trong CSS và phạm vi ảnh hưởng

Sử dụng và Phân loại CSS – Phân loại

- Gồm 3 loại CSS
 - **Inline Style Sheet** (Nhúng CSS vào tag HTML)
 - **Embedding Style Sheet** (Nhúng CSS vào trang web)
 - **External Style Sheet** (Liên kết CSS với trang web)

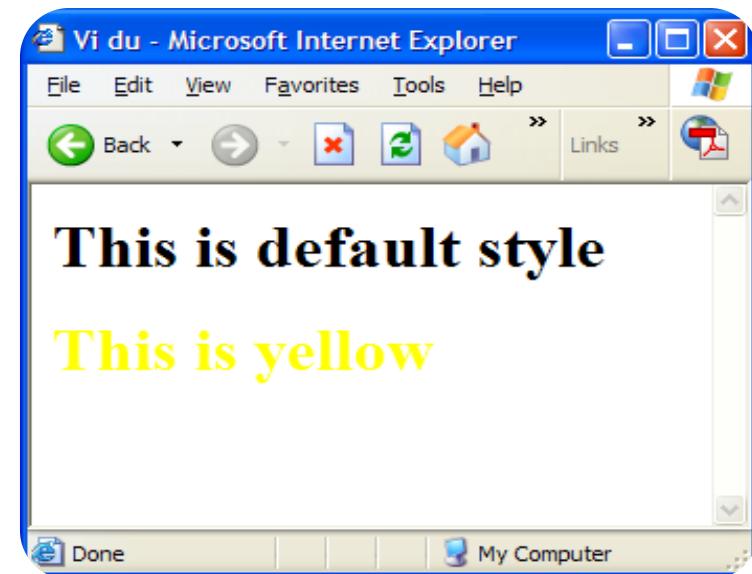
Sử dụng và Phân loại CSS - Inline Style Sheet

- Định nghĩa style trong thuộc tính style của từng tag HTML.
- Theo cú pháp **kiểu 1.**

```
<tag style = "property1:value1;...propertyN:valueN;"> ... </tag>
```

- Không sử dụng lại được.
- Ví dụ:

```
<H1 STYLE="color: yellow">This is yellow</H1>
```



Sử dụng và Phân loại CSS - Embedding Style Sheet

- Còn gọi là **Internal Style Sheet** hoặc **Document-Wide Style Sheet**
- Mọi định nghĩa style được đặt trong tag **<style>** của trang HTML.
- Định nghĩa style theo cú pháp **kiểu 2**.
- Trang HTML có nội dung như sau:

```
<head>
```

```
  <style type="text/css" >  
    <!--  
      SelectorName {  
        property1:value1;  
        property2:value2;  
        .....  
        propertyN:valueN; }  
    -->
```

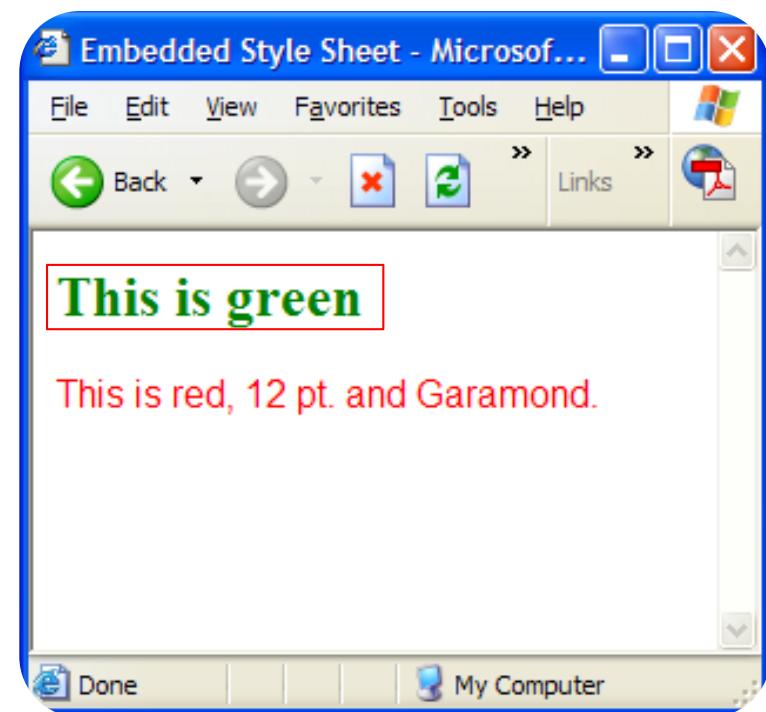
```
  </style>
```

```
</head>
```

Sử dụng và Phân loại CSS - Embedding Style Sheet

```
<HTML>
<HEAD>
    <TITLE>
        Embedded Style Sheet
    </TITLE>

    <STYLE TYPE="text/css">
        <!--
            P {color: red;
                font-size: 12pt;
                font-family: Garamond;}
            H2 {color: green;}
        -->
    </STYLE>
</HEAD>
<BODY BGCOLOR="#FFFFFF">
    <H2>This is green</H2>
    <P>This is red, 12 pt. and Garamond.</P>
</BODY>
</HTML>
```



Sử dụng và Phân loại CSS - External Style Sheet

- Mọi style đều lưu trong file có phần mở rộng là ***.CSS**.
- File CSS: lưu trữ nhiều style theo cú pháp **kiểu 2**.
- Trong file HTML: liên kết bằng tag **link**. Cú pháp:

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

- Trang HTML : Liên kết bằng tag **style** với **@import url**. Cú pháp

```
<head>
    <style type="text/css" media="all | print | screen" >
        @import url(URL);
    </style>
</head>
```

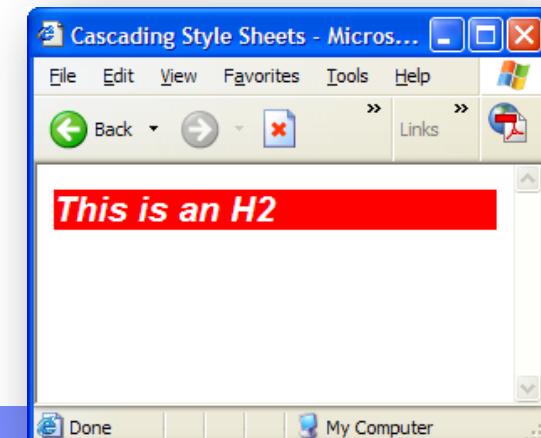
Sử dụng và Phân loại CSS - External Style Sheet

Trong tập tin **MyStyle.CSS**

```
H2
{
    FONT-WEIGHT: bold;
    FONT-SIZE: 16pt;
    COLOR: white;
    FONT-STYLE: italic;
    FONT-FAMILY: Arial;
    BACKGROUND-COLOR: red;
    font-color: white
}
```

Trong trang Web : **demo.htm**

```
<html>
<head>
    <title>Cass.....</title>
    <link HREF="MyStyle.css"
        REL="stylesheet" >
</head>
<body>
    <h2>This is an H2 </h2>
</body>
</html>
```

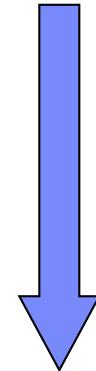


Sử dụng và Phân loại CSS – So sánh, Đánh giá

	Inline Style Sheet	Embedding Style Sheet	External Style Sheet
Khai báo	Kiểu 1	Kiểu 2	Kiểu 2
Cú pháp	<pre><p style="color:red;"> Test </p></pre>	<pre><style type="text/css"> .TieuDe1{color: red;} </style> <p class="TieuDe1"> Test </p></pre>	<pre><link rel="stylesheet" href="main.css" /> <p class="TieuDe1"> Test </p></pre>
Ưu điểm			
Khuyết điểm			

Sử dụng và Phân loại CSS – Độ ưu tiên

- Thứ tự ưu tiên áp dụng định dạng khi sử dụng các loại CSS (độ ưu tiên giảm dần) :
 1. Inline Style Sheet
 2. Embedding Style Sheet
 3. External Style Sheet
 4. Browser Default



PriorityCSS.css

```

1 .MyExColor {
2     color : Red;
3 }
4
5 div{
6     font-size:36px;
7 }

```

PriorityCSS.htm

```

1 <html>
2     <head>
3         <link rel="stylesheet" href="PriorityCSS.css" />
4
5         <style type="text/css">
6             .MyEmColor{
7                 color:blue;
8             }
9             div{
10                 font-size:20px;
11             }
12         </style>
13     </head>
14
15     <body>
16         <div style="color:pink"> Inline Style Sheet </div>
17         <div class="MyEmColor"> Embedding Style Sheet </div>
18         <div class="MyExColor"> External Style Sheet </div>
19
20         <hr>
21         <p>
22             <font style="font-size:14px">Font size = 14 </font> |
23             <font style="font-size:20px">Font size = 20 </font> |
24             <font style="font-size:36px">Font size = 36</font>
25
26             <div style="color:blue">
27                 <div class="MyExColor"> External Color or Document color ? </div>
28             </div>
29
30             <div> Font size = Embedding or External ? </div>
31             <div style="font-size:14px"> Font Size = Inline or Embedding ? </div>
32         </body>
33     </html>

```

I:\Lap trinh va Thiet ke web 1\Lý thuyết\anh Vu\B... (1)

Inline Style Sheet
Embedding Style Sheet
External Style Sheet

Font size = 14 | Font size = 20 | Font size = 36

External Color or Document color ?
Font size = Embedding or External ?
Font Size = Inline or Embedding ?

Nội dung

- Giới thiệu CSS
- Định nghĩa Style
- Sử dụng và Phân loại CSS
- Selector trong CSS và phạm vi ảnh hưởng

Selector

- Là tên 1 style tương ứng với một thành phần được áp dụng định dạng
- Các dạng selectors
 - HTML element selectors
 - Class selectors
 - ID selectors
 -

Ví dụ:

```
.TieuDe1 {  
color: red;  
font-family: Verdana, sans-serif; }
```

```
<h1 class="TieuDe1"> DHKHTN </h1>
```

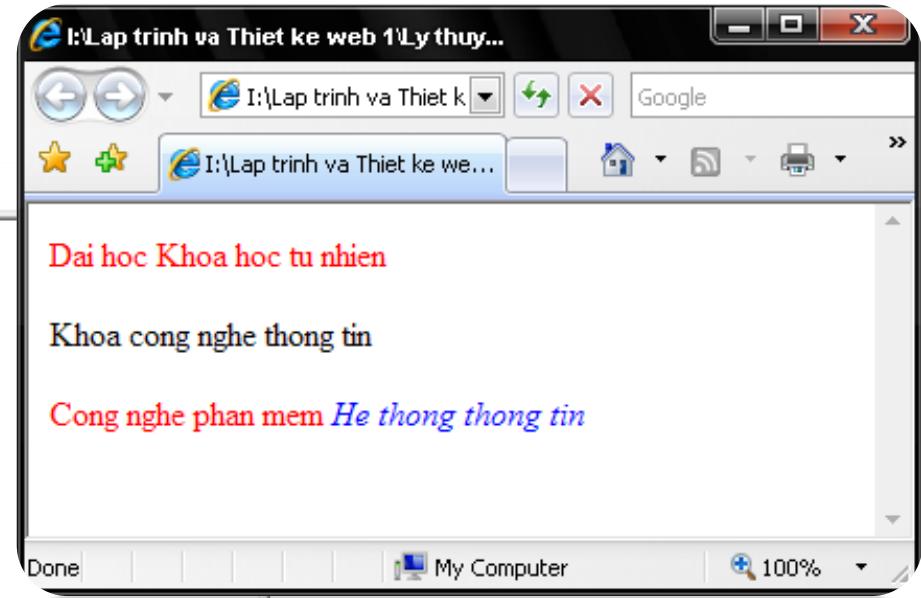
Selector trong CSS

Loại	Mô tả phạm vi ảnh hưởng	Ví dụ
element		
#id		
.class		
element . class		
Grouping		
Contextual		
Pseudo Class Pseudo element		

Selector trong CSS - Element

- Có hiệu ứng trên tất cả element cùng loại tag

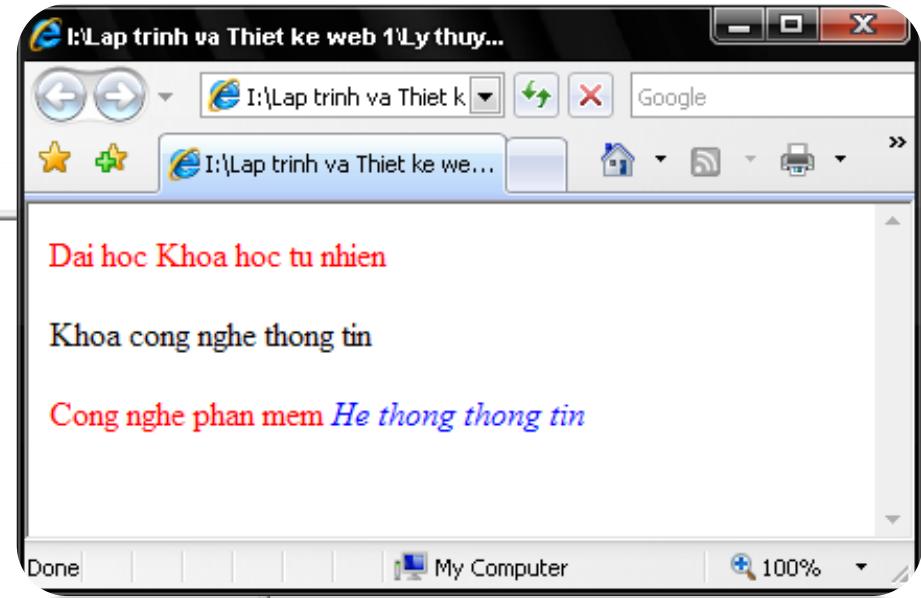
- Ví dụ :



The screenshot shows a web browser window with the title "I:\Lap trinh va Thiet ke web 1\Ly thuy...". The page content is as follows:

```
<html>
  <head>
    <style type="text/css">
      p {color:red}
      Em {color:blue}
    </style>
  </head>
  <body>
    <p> Truong Dai hoc Khoa hoc tu nhien </p>
    <div>Khoa cong nghe thong tin</div>
    <p> Cong nghe phan mem <em> He thong thong tin </em> </p>
  </body>
</html>
```

The browser displays the text "Dai hoc Khoa hoc tu nhien" in red, "Khoa cong nghe thong tin" in blue, and "Cong nghe phan mem He thong thong tin" where "Cong nghe phan mem" is red and "He thong thong tin" is blue.

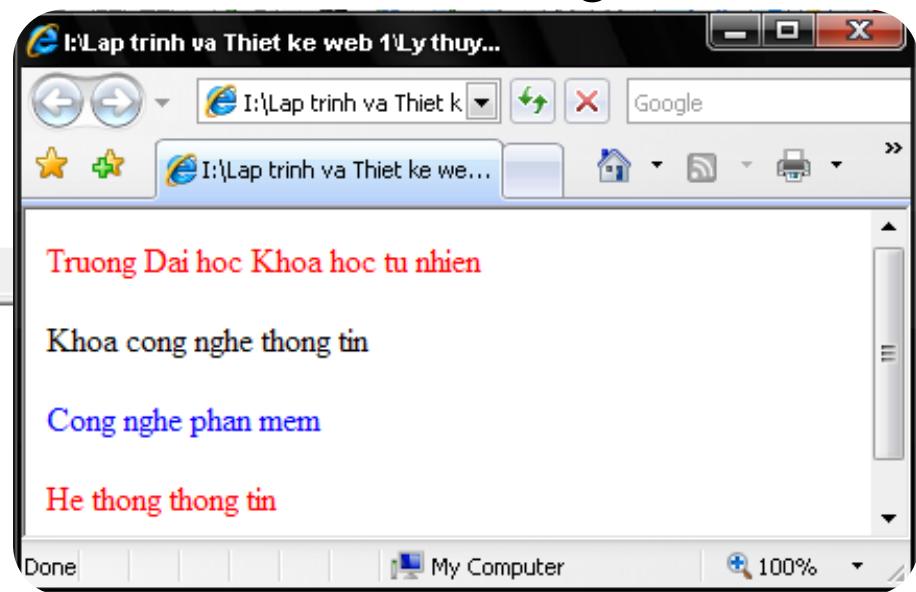


Selector trong CSS

Loại	Mô tả phạm vi ảnh hưởng	Ví dụ
element	Định dạng áp dụng cho ND tất cả các tag Element trong tài liệu Web	<code>h1 {color: red;}</code> /* ND của thẻ <h1> bị định dạng màu chữ=đỏ */
#id		
.class		
element . class		
Grouping		
Contextual		
Pseudo Class Pseudo element		

Selector trong CSS – ID rules

- Có hiệu ứng duy nhất trên một element có đúng id.
- Ví dụ :



The screenshot shows a Microsoft Internet Explorer window displaying the content of a file named "Test.htm". The HTML code includes a style block with two ID rules: #id001{color:red} and #id002{color:blue}. The first rule is applied to the text "Truong Dai hoc Khoa hoc tu nhien" and "Khoa cong nghe thong tin", which are displayed in red. The second rule is applied to the text "Cong nghe phan mem" and "He thong thong tin", which are displayed in blue. The browser interface shows the title bar, toolbar, and status bar.

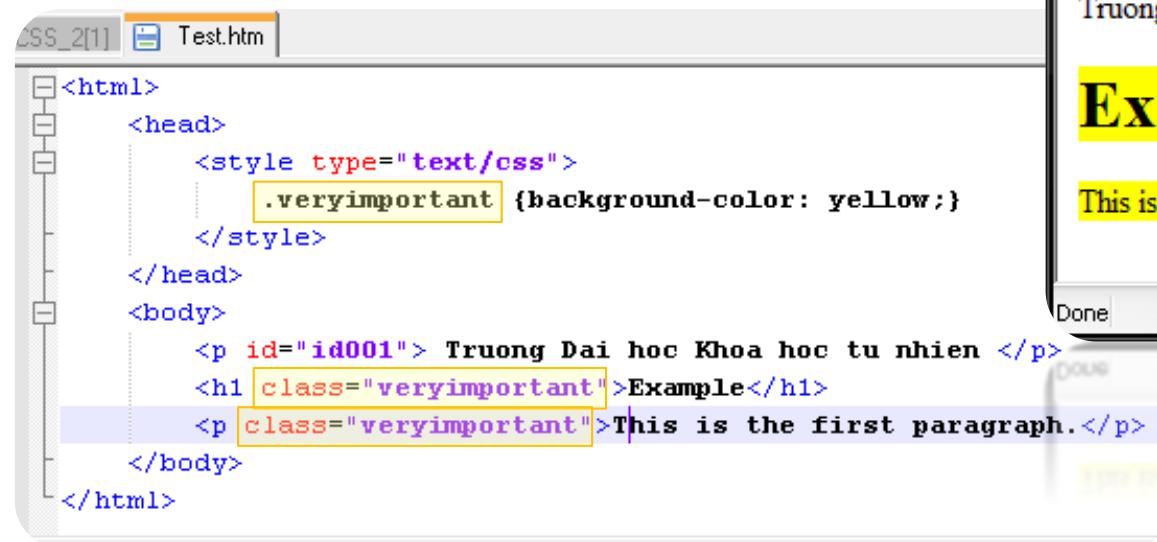
```
<html>
  <head>
    <style type="text/css">
      #id001{color:red}
      #id002{color:blue}
    </style>
  </head>
  <body>
    <p id="id001"> Truong Dai hoc Khoa hoc tu nhien </p>
    <p>Khoa cong nghe thong tin</p>
    <div id="id002">Cong nghe phan mem<div id="id001">He thong thong tin</div></div>
  </body>
</html>
```

Selector trong CSS

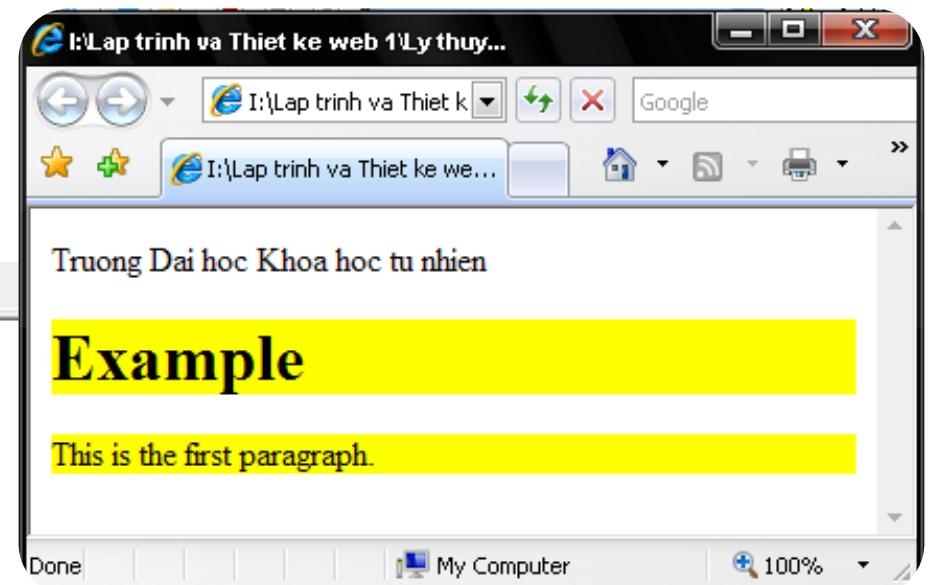
Loại	Mô tả phạm vi ảnh hưởng	Ví dụ
element	Định dạng áp dụng cho ND tất cả các tag Element trong tài liệu Web	<code>h1 {color: red;}</code> /* ND của thẻ <h1> bị định dạng màu chữ=đỏ */
#id	Định dạng áp dụng cho ND tất cả các tab có thuộc tính id trong tài liệu Web	<code>#test {color: green;}</code> /* ND của bất kỳ tag có thuộc tính id=test đều bị định dạng màu chữ=xanh lá */
.class		
element . class		
Grouping		
Contextual		
Pseudo Class Pseudo element		

Selector trong CSS – Class rules

- Có hiệu ứng trên tất cả các loại tag có cùng giá trị thuộc tính **class**.
- Ví dụ :



```
<html>
  <head>
    <style type="text/css">
      .veryimportant {background-color: yellow;}
    </style>
  </head>
  <body>
    <p id="id001"> Truong Dai hoc Khoa hoc tu nhien </p>
    <h1 class="veryimportant">Example</h1>
    <p class="veryimportant">This is the first paragraph.</p>
  </body>
</html>
```

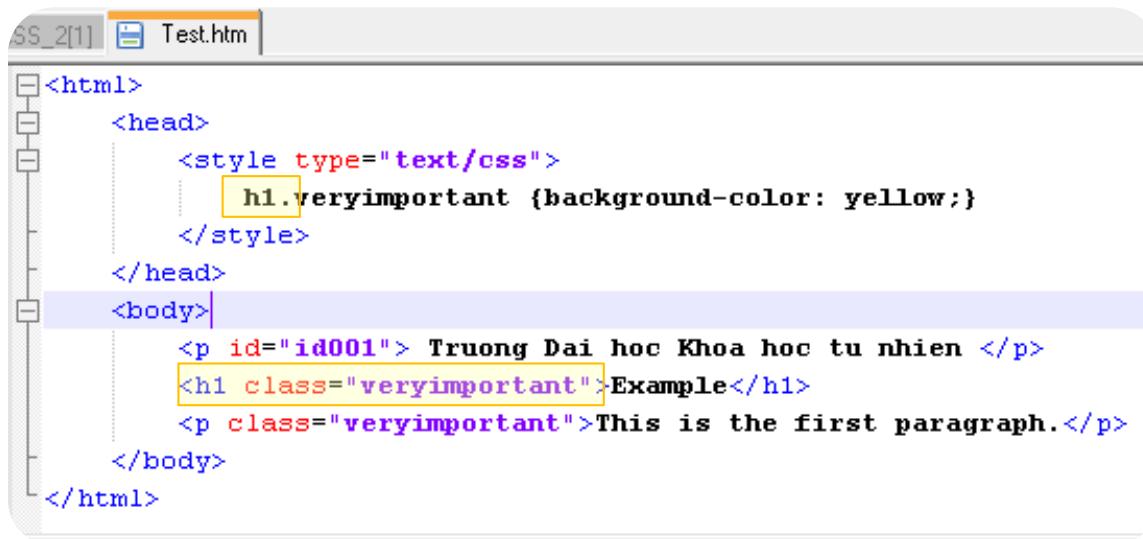


Selector trong CSS

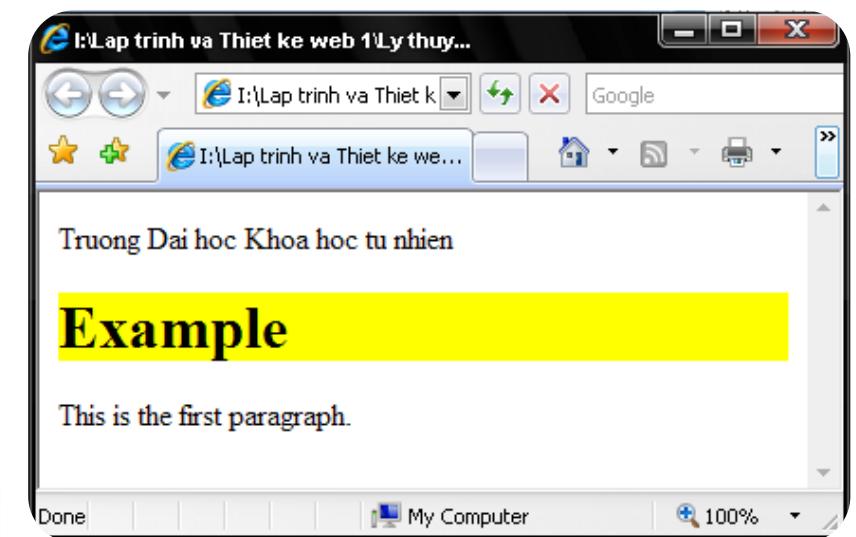
Loại	Mô tả phạm vi ảnh hưởng	Ví dụ
element	Định dạng áp dụng cho ND tất cả các tag Element trong tài liệu Web	<code>h1 {color: red;}</code> /* ND của thẻ <h1> bị định dạng màu chữ=đỏ */
#id	Định dạng áp dụng cho ND tất cả các tab có thuộc tính id trong tài liệu Web	<code>#test {color: green;}</code> /* ND của bất kỳ tag có thuộc tính id=test đều bị định dạng màu chữ=xanh lá */
.class	Định dạng áp dụng cho ND tất cả các tab có thuộc tính class trong tài liệu Web	<code>.note {color: yellow;}</code> /* ND của bất kỳ tag có thuộc tính class=note đều bị định dạng màu chữ=vàng */
element . class		
Grouping		
Contextual		
Pseudo Class Pseudo element		

Selector trong CSS – Kết hợp Element và Class

■ Ví dụ :



```
<html>
  <head>
    <style type="text/css">
      h1.veryimportant {background-color: yellow;}
    </style>
  </head>
  <body>
    <p id="id001"> Truong Dai hoc Khoa hoc tu nhien </p>
    <h1 class="veryimportant">Example</h1>
    <p class="veryimportant">This is the first paragraph.</p>
  </body>
</html>
```

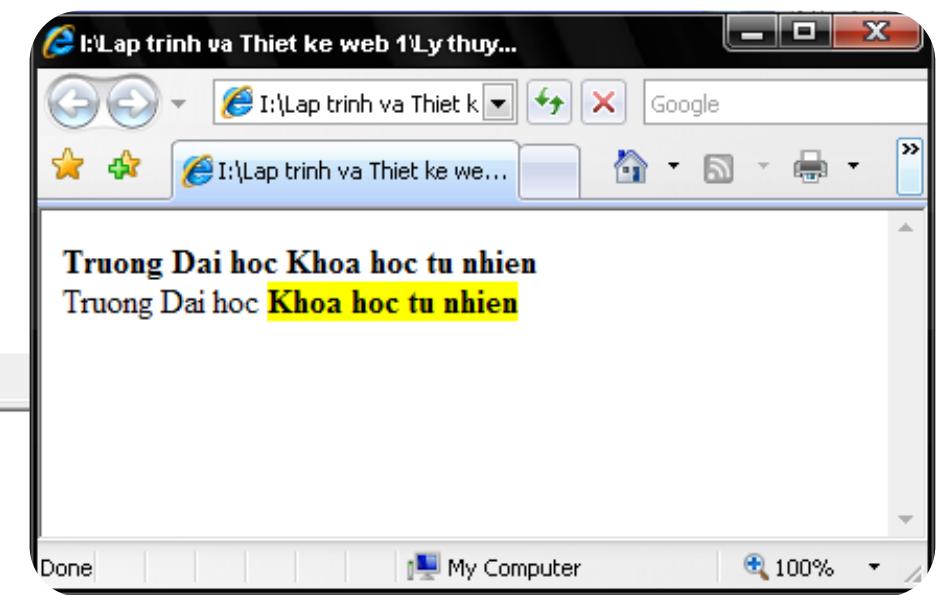


Selector trong CSS

Loại	Mô tả phạm vi ảnh hưởng	Ví dụ
element	Định dạng áp dụng cho ND tất cả các tag Element trong tài liệu Web	<code>h1 {color: red;}</code> /* ND của thẻ <h1> bị định dạng màu chữ=đỏ */
#id	Định dạng áp dụng cho ND tất cả các tab có thuộc tính id trong tài liệu Web	<code>#test {color: green;}</code> /* ND của bất kỳ tag có thuộc tính id=test đều bị định dạng màu chữ=xanh lá */
.class	Định dạng áp dụng cho ND tất cả các tab có thuộc tính class trong tài liệu Web	<code>.note {color: yellow;}</code> /* ND của bất kỳ tag có thuộc tính class=note đều bị định dạng màu chữ=vàng */
element . class	Định dạng áp dụng cho ND các tag Element có thuộc tính class tương ứng	<code>h1.note {text-decoration: underline;}</code> /* ND của các thẻ <h1> có thuộc tính class=note đều bị định dạng gạch chân */
Grouping		
Contextual		
Pseudo Class Pseudo element		

Selector trong CSS - Contextual Selection

- Định dạng được áp dụng cho nội dung trong chuỗi tag theo đúng thứ tự
- Ví dụ :

A screenshot of a code editor window titled "PriorityCSS_2[1]". The tab bar shows "Test.htm". The code editor displays the following HTML and CSS:

```
1 <html>
2   <head>
3     <style type="text/css">
4       div b {background-color: yellow;}
5     </style>
6   </head>
7   <body>
8     <b><div>Truong Dai hoc Khoa hoc tu nhien </div></b>
9     <div>Truong Dai hoc <b>Khoa hoc tu nhien</b></div>
10    </body>
11 </html>
```

The line "div b {background-color: yellow;}" is highlighted with a yellow box. The entire code editor interface is visible, showing the file structure and the code content.

11 </body>
10 </html>

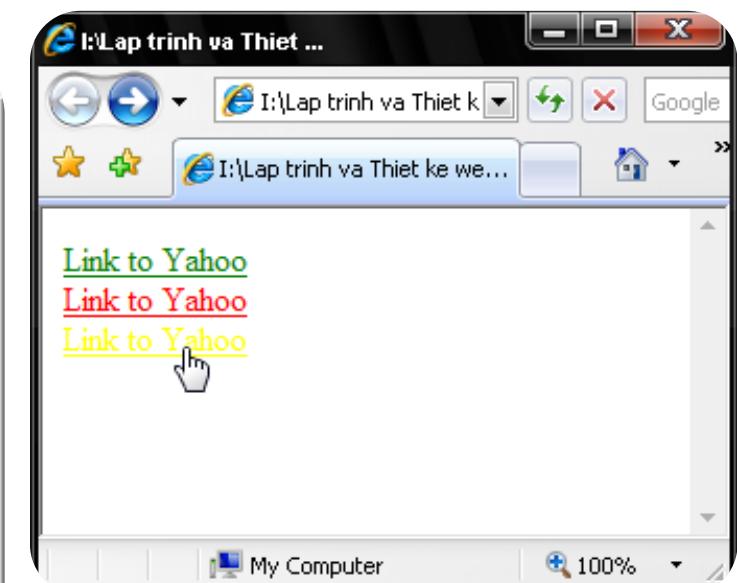
Selector trong CSS

Loại	Mô tả phạm vi ảnh hưởng	Ví dụ
element	Định dạng áp dụng cho ND tất cả các tag Element trong tài liệu Web	<code>h1 {color: red;}</code> /* ND của thẻ <h1> bị định dạng màu chữ=đỏ */
#id	Định dạng áp dụng cho ND tất cả các tab có thuộc tính id trong tài liệu Web	<code>#test {color: green;}</code> /* ND của bất kỳ tag có thuộc tính id=test đều bị định dạng màu chữ=xanh lá */
.class	Định dạng áp dụng cho ND tất cả các tab có thuộc tính class trong tài liệu Web	<code>.note {color: yellow;}</code> /* ND của bất kỳ tag có thuộc tính class=note đều bị định dạng màu chữ=vàng */
element . class	Định dạng áp dụng cho ND các tag Element có thuộc tính class tương ứng	<code>h1.note {text-decoration: underline;}</code> /* ND của các thẻ <h1> có thuộc tính class=note đều bị định dạng gạch chân */
Grouping	Định dạng áp dụng cho ND một nhóm các tag trong tài liệu.	<code>h1,h2,h3 {background-color: orange;}</code> /* ND của các thẻ <h1> <h2> <h3> đều bị định dạng màu nền = màu cam */
Contextual	Định dạng áp dụng cho ND các thẻ được lồng trong một thẻ cha nào đó	<code>p strong {color: purple;}</code> /* ND của các thẻ nằm trong thẻ <p> đều bị định dạng màu chữ=màu tía */
Pseudo Class Pseudo element		

Selector trong CSS – Pseudo Class

- Định dạng dựa vào trạng thái của liên kết, sự kiện chuột.
- Có thể kết hợp với Selector khác.

```
PriorityCSS_2[1] Test.htm
1 <html>
2   <head>
3     <style type="text/css">
4       a:link {color:red}
5       a:visited {color:green}
6       a:active {color:yellow }
7     </style>
8   </head>
9   <body>
10    <a href="http://www.yahoo.com">Link to Yahoo</a>
11    <br>
12    <a href="http://www.google.com">Link to Yahoo</a>
13    <br>
14    <a href="http://www.msn.com">Link to Yahoo</a>
15    <br>
16  </body>
17 </html>
```



```
T3 </p></td>
T4 </tr>
T5 </table>
T6 </div>
T7 </div>
T8 </div>
T9 </div>
T10 </div>
```

Một số quy ước về đơn vị trong CSS

■ Đơn vị chiều dài

Đơn vị	Mô tả
%	Phần trăm
In	Inch (1 inch = 2.54 cm)
cm	Centimeter
mm	Millimeter
pc	Pica (1 pc = 12 pt)
px	Pixels (điểm ảnh trên màn hình máy tính)
pt	Point (1 pt = 1/72 inch)
em	1 em tương đương kích thước font hiện hành, nếu font hiện hành có kích cỡ 14px thì 1 em = 14 px. Đây là một đơn vị rất hữu ích

Một số quy ước về đơn vị trong CSS

■ Đơn vị màu sắc

Đơn vị	Mô tả
Color-name	Tên màu. Ví dụ: black, white, red, green, blue, cyan, magenta,...
RGB (r,g,b)	Màu RGB với 3 giá trị R, G, B có trị từ 0 – 255 kết hợp với nhau tạo ra vô số màu.
RGB(%r,%g,%b)	Màu RGB với 3 giá trị R, G, B có trị từ 0 – 100% kết hợp.
Hexadecimal RGB	Mã màu RGB dạng hệ thập lục. Ví dụ: #FFFF: trắng, #000: đen,

Một số CSS cơ bản

1. CSS Background
2. CSS Text
3. CSS Font
4. Mô hình hộp CSS: Margin, Border, Padding
5. Lược đồ vị trí trong CSS

4.1. CSS Background

- Thuộc tính Background cho phép thiết lập màu nền, ảnh nền, lặp ảnh...

background-color

- Thường áp dụng cho thẻ: body và các thẻ hiển thị dữ liệu: DIV, P, H1... để đặt màu nền

background-image

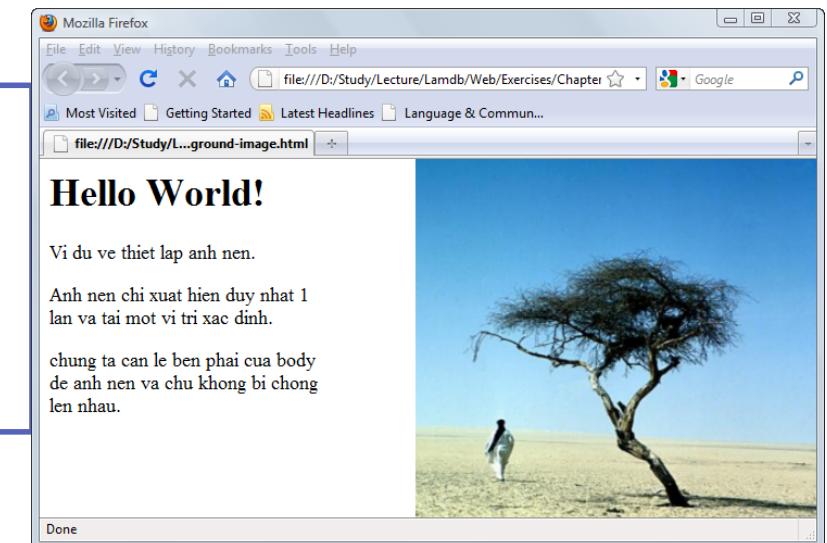
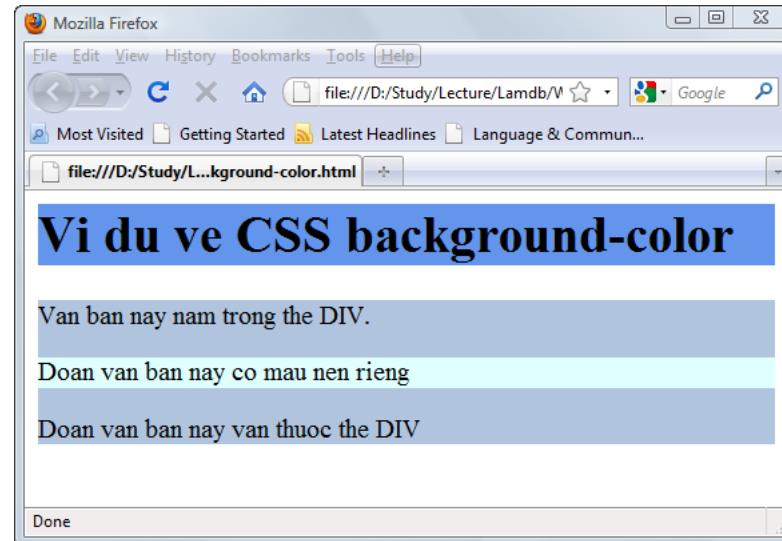
- Thường sử dụng để chèn một ảnh nền BODY
- Căn ảnh, xác định vị trí ảnh và cho phép lặp ảnh hay không

4.1. CSS Background

■ Ví dụ:

```
body{  
    background-color: "red";  
}  
  
h1{  
    background-color:#645eff;  
}
```

```
body{  
    background-image:url ('tree.png');  
    background-repeat:no-repeat;  
    background-position:top right;  
    background-attachment:fixed;  
    margin-right:300px;  
}
```



4.2. CSS Text

direction

- ltr: left to right. Chữ hiển thị từ bên trái sang phải
- rtl: right to left. Chữ hiển thị từ bên phải sang trái

letter-spacing

- Tăng/giảm khoảng cách giữa các kí tự so với trạng thái thông thường (normal)
 - Giá trị có thể là số dương hoặc âm

text-align

- Căn nội dung văn bản theo chiều ngang: left, center, right, justify

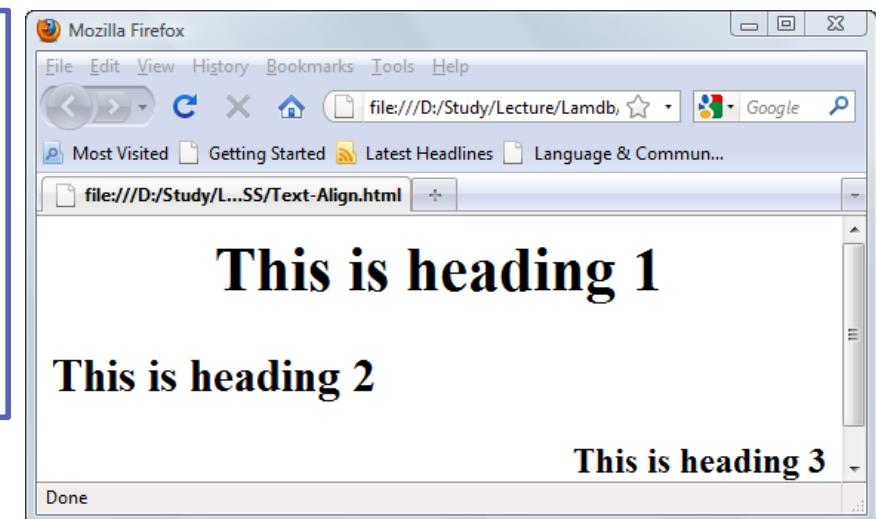
4.2. CSS Text

■ Ví dụ:

```
<style type="text/css">
    h1 {letter-spacing:2px}
    h2 {letter-spacing:-3px}
</style>
```



```
<style type="text/css">
    h1 {text-align:center}
    h2 {text-align:left}
    h3 {text-align:right}
</style>
```



4.3. CSS Font

font-family

- Các kiểu font: Liệt kê các font sử dụng, ngăn cách nhau bởi dấu ;

font-size

- Đặt kích thước cho văn bản. Mặc định kích thước thông thường là 16px

font-style

font-weight

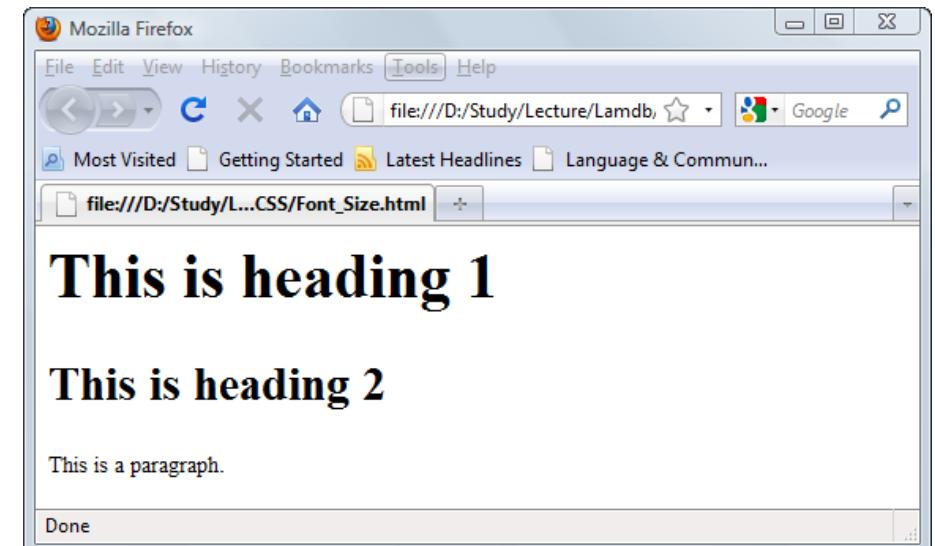


4.3. CSS Font

- Ví dụ:

```
<style type="text/css">
    p{font-family:"Times New
Roman",Georgia,Serif}
</style>
```

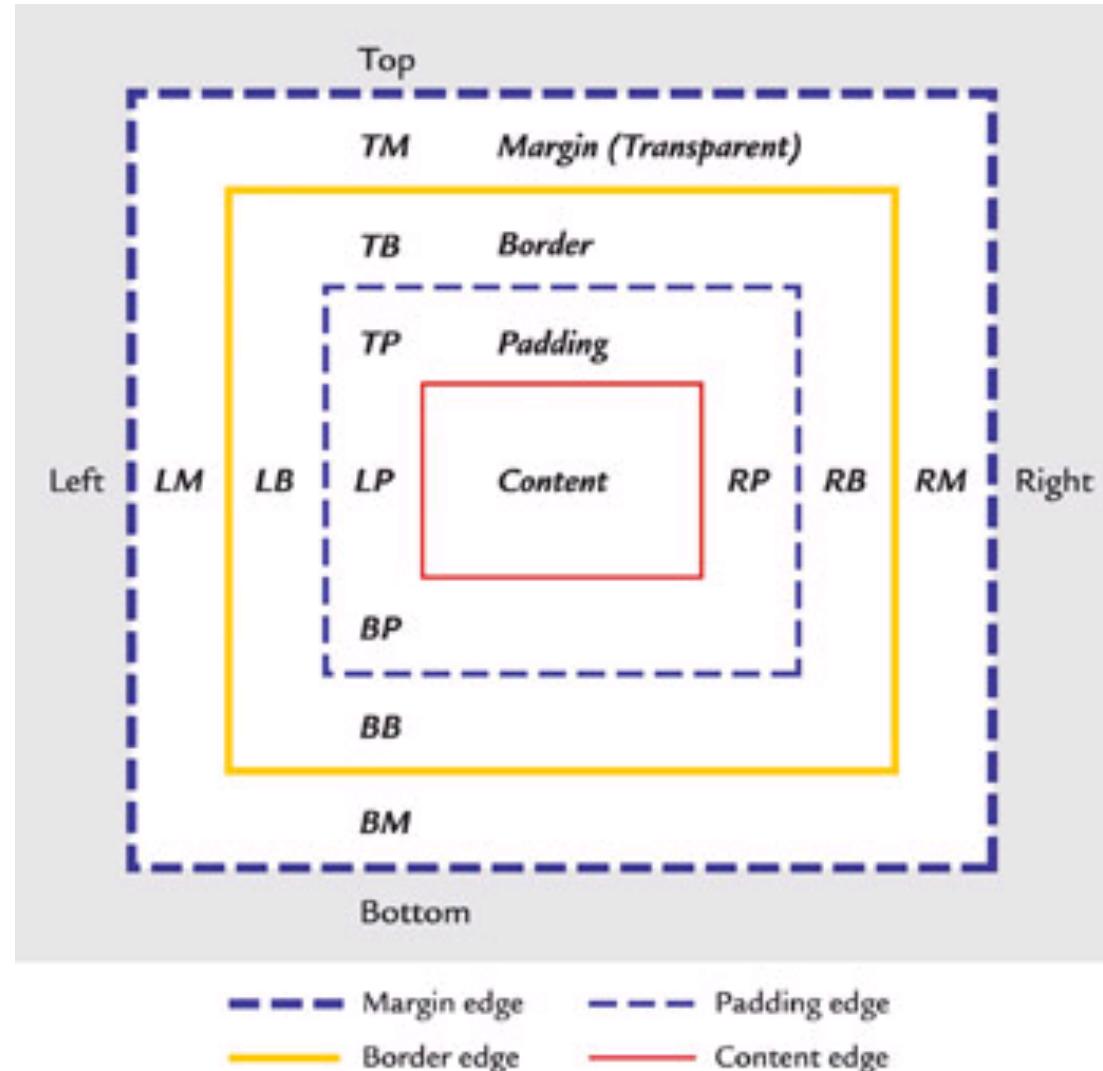
```
h1 {font-size:40px}
h2 {font-size:30px}
p {font-size:14px}
```



Mô hình hộp

- Trong CSS, box model (mô hình hộp) mô tả cách mà CSS định dạng khối không gian bao quanh một thành phần.
- Mỗi một phần tử trong trang được xem như là một hình chữ nhật được tạo ra từ thành phần content, padding, border và margin.
 - Padding bao quanh phần nội dung. Thường được sử dụng để tạo nền cho một phần tử.
 - Border tạo ra đường viền bao quanh phần padding.
 - Margin trong suốt và không thể nhìn thấy, chúng được sử dụng để ngăn cách giữa các thành phần.
 - Padding, border, và margin là lựa chọn không bắt buộc và có thể nhận giá trị bằng 0.

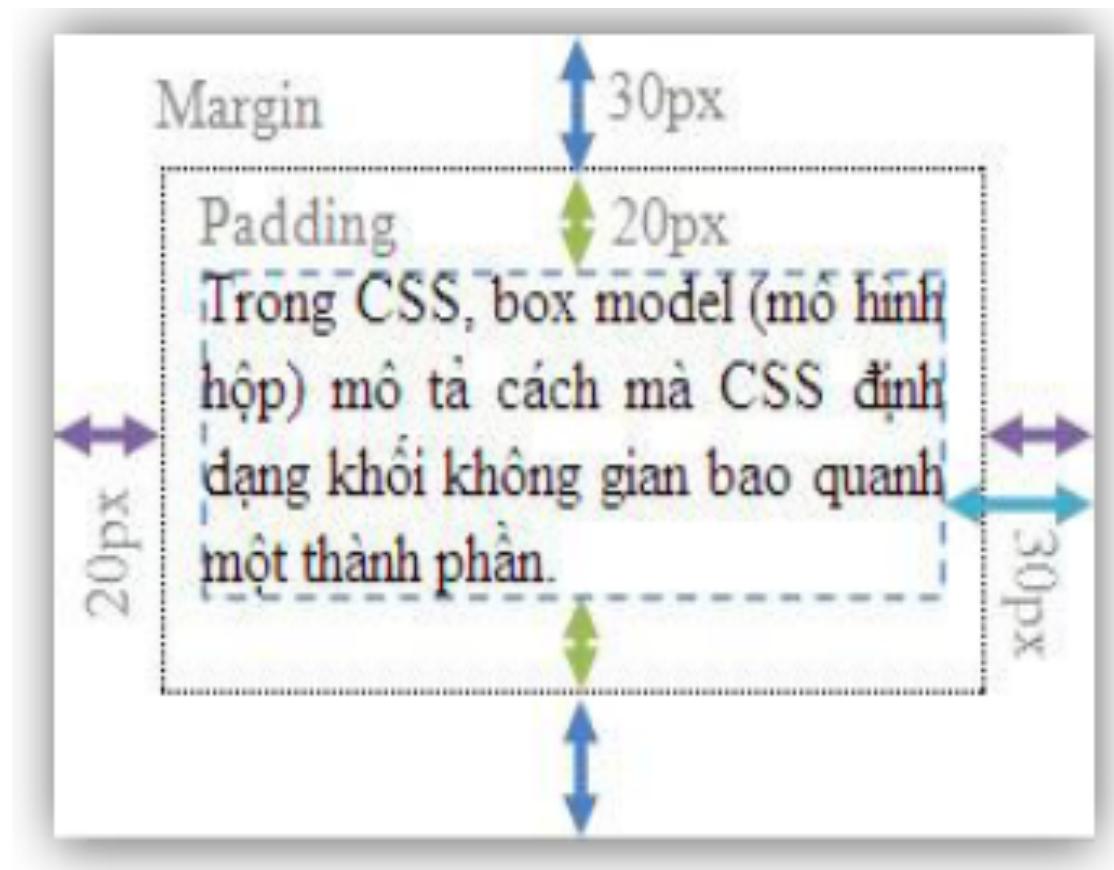
Mô hình hộp (con't)



Mô hình hộp

- Ví dụ:

```
p {  
    width:200px;  
    margin:30px 20px;  
    padding:20px 10px;  
    border:1px solid #000;  
    text-align:justify  
}
```



4.4. CSS Margin

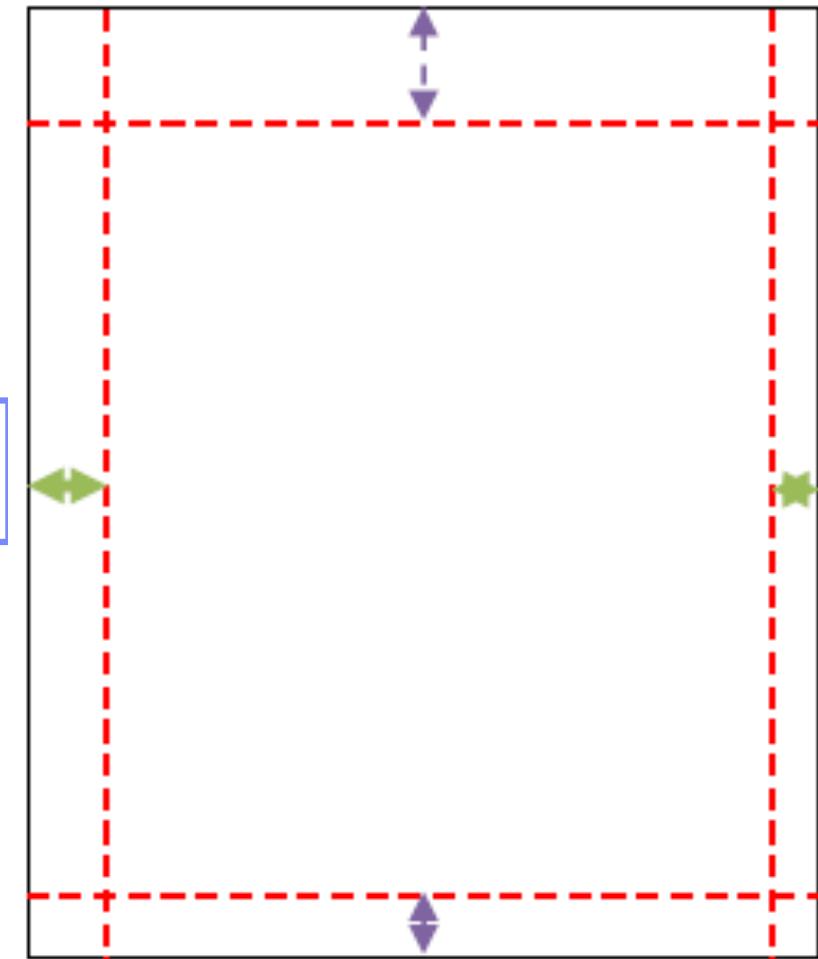
- Tương ứng với 4 phía ta có 4 thuộc tính:
 - margin-top:
 - margin-right:
 - margin-bottom:
 - margin-left:
- Để cho gọn chúng ta cũng có thể viết thuộc tính **margin** dưới dạng **shorthand**

```
div.margin { margin: 10px 4px 5px 9px } ;  
/* top | right | bottom | left */
```

4.4. CSS Margin

```
margin:25px 50px 75px 100px;  
margin:25px 50px 75px;  
margin:25px 50px;  
margin:25px;
```

```
body { margin:80px 30px 40px 50px;  
border:1px dotted #FF0000 }
```



4.4. CSS Margin

- Nếu muốn đặt màu cho đường bao chúng ta có thể sử dụng thuộc tính **outline-color**
- Để đặt độ rộng cho đường bao chúng ta đặt giá trị độ lớn cho thuộc tính **outline-width**
- Để chọn kiểu cho đường bao chúng ta sẽ đặt lần lượt các giá trị cho thuộc tính **outline-style**

4.4. CSS Border

- Chỉ ra độ rộng, màu và kiểu của vùng border của box. Thuộc tính này áp dụng cho tất cả các phần tử:

- border-color:
- border-width:
- border-style:

- Độ rộng của border:

```
div.borderwidth {
    border-width: 2px;
}
```

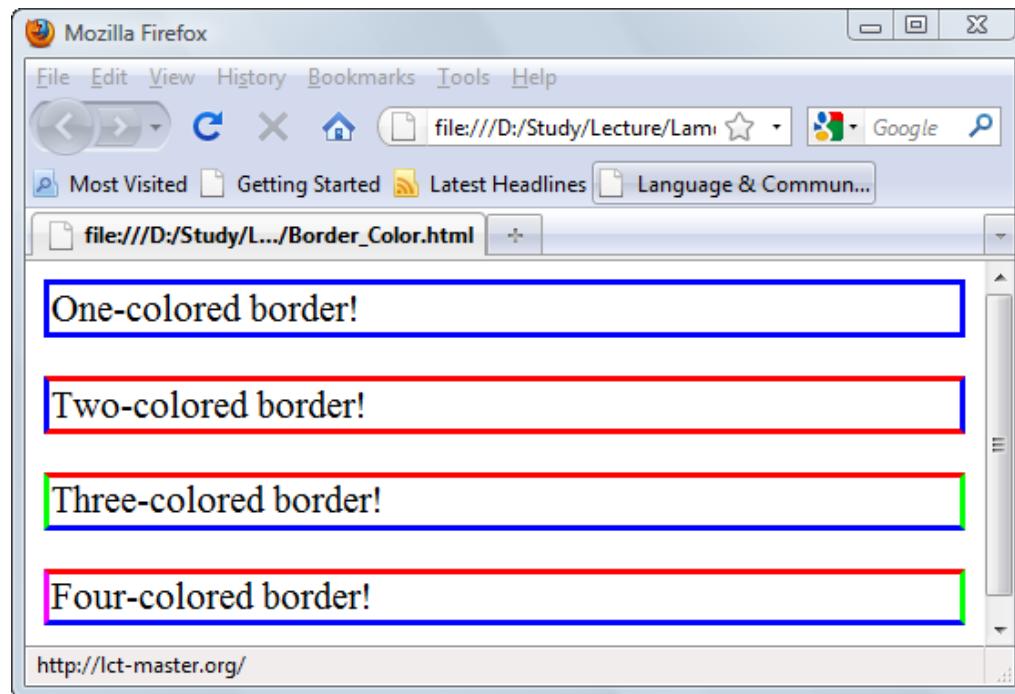
STT	Giá trị
1	thin
2	medium
3	thick
4	length

- 'border-top-width', 'border-right-width', 'border-bottom-width', 'border-left-width', và 'border-width'

4.4. CSS Border

border-color

- Có thể xác định từ 1 đến 4 màu cho các phía tương ứng
- Phải đi kèm với border-style



```
border-color:red green blue pink;  
border-color:red green blue;  
border-color:red green;  
border-color:red;
```

4.4. CSS Border

Thuộc tính **border-style**

- Xác định kiểu đường của border
- Tham chiếu kiểu giá trị <border-style>

với giá trị sau:

- **none**: No border.-> border-width:0
- **dotted**: đường chấm
- **dashed**: gạch nhỏ
- **solid**: đường đơn.
- **double**: đôi. Khoảng cách giữa 2 đường và độ dày của chúng bằng giá trị border-width
- **groove, ridge**
- **Inset, outset**

4.4. CSS Border

Thuộc tính border style

```

h1 {
    border-width:thin;
    border-color:#FF0000;
    border-style:solid
}
h2 {
    border-width:thick;
    border-color:#CCC;
    border-style:dotted
}

```

STT	border-style
1	none
2	hidden
3	dotted
4	dashed
5	solid
6	double
7	groove
8	ridge
9	inset
10	outset

Viền Kiểu Solid

Viền Kiểu Dotted

Viền Kiểu Dashed

Viền Kiểu Double

Viền Kiểu Groove

Viền Kiểu Ridge

Viền Kiểu Inset

Viền Kiểu Outset

4.5. CSS khác

- CSS Padding: thiết lập khoảng cách giữa nội dung và đường viền
- CSS List: thiết lập vị trí, kiểu, ảnh của mỗi thành phần trong danh sách
- CSS Table: thiết lập cách hiển thị bảng như khoảng cách giữa các ô, chú thích

5. Lược đồ vị trí trong CSS

- **Normal Flow** : theo mặc định của trình duyệt đối với những phần tử không thiết lập position:absolute hoặc fixed và không float
- **FLOATS**: được thiết lập theo chế độ của normal flow sau đó được dịch trái hoặc phải
- **Absolute Positioning** : liên quan tới vị trí tuyệt đối của phần tử liên quan tới khối nội dung
- **Relative Positioning** : Lược đồ vị trí được sử dụng bởi hai thuộc tính position và float

5. Lược đồ vị trí trong CSS

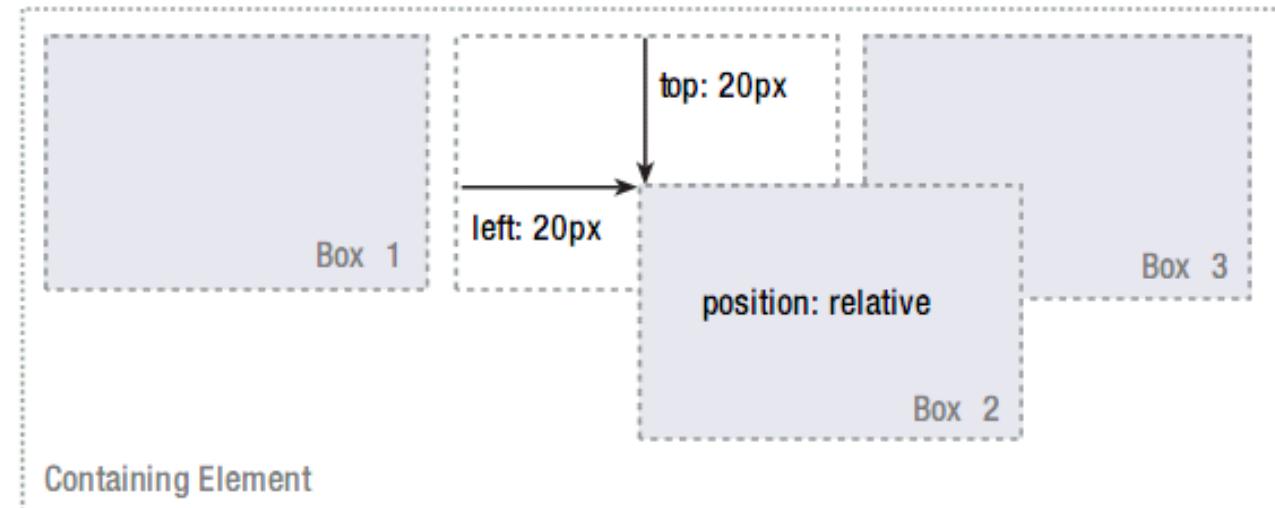
Thuộc tính **position**

- **Static:** Box là một box cơ bản, nằm theo normal flow. Thuộc tính left và top không có tác dụng
- **Relative:** Vị trí box được tính toán theo normal flow. Box sẽ được dịch chuyển so với normal flow.
- **Absolute:** Vị trí của box được xác định bởi các thuộc tính top, bottom, left, right không phụ thuộc vào normal flow
- **Fix:** Được tính giống như absolute nhưng sẽ cố định.

5.1. Relative positioning

- Một box ở vị trí relative được sinh ra khi thuộc tính **position** có giá trị **relative**.
- Sự dịch chuyển so với normal flow được xác định bởi các thuộc tính 'top', 'bottom', 'left', và 'right'.

```
#myBox {  
position: relative;  
left: 20px;  
top: 20px;  
}
```



5.1. Relative positioning

FLOATS

- Một float là một box được dịch chuyển sang trái hoặc phải so với dòng hiện thời.
- Đặc tính đáng quan tâm nhất của float đó là nội dung có thể chảy theo các phía của nó
- Nội dung chảy xuống bên phải của hộp định trái và bên trái của hộp định phải



5.1. Relative positioning

Thuộc tính **float**

- *Left*: Nội dung chạy bên phải của box, bắt đầu từ đỉnh. Thuộc tính display sẽ bị loại bỏ trừ khi giá trị display được gán bằng none.
- *Right*: Tương tự như left, nhưng nội dung theo bên trái của box, bắt đầu từ đỉnh.
- *None*

5.1. Relative positioning

Thuộc tính **float**

Ví dụ: thực hiện float ảnh logo sang trái để phần nội dung bên dưới tràn lên nằm cạnh logo.

```
#logo { float:left; }
```

Ví dụ: dùng float để chia 2 cột văn bản.

```
.column1
{
    width:45%;
    float:left;
    text-align:justify;
    padding:0 20px;
}

.column2
{
    border-right:1px
    solid #000
}
```

5.1. Relative positioning

Thuộc tính clear:

- Thuộc tính clear là một thuộc tính thường được gán vào các phần tử liên quan tới phần tử đã được **float** để quyết định hướng xử sự của phần tử này.
- Ở ví dụ trên, khi chúng ta float tấm ảnh qua trái thì mặc nhiên văn bản sẽ được tràn lên để lấp vào chỗ trống. Nhưng khi chúng ta đặt vào văn bản thuộc tính **clear** thì chúng ta có quyền quyết định xem phần văn bản đó có được tràn lên hay không.

5.1. Relative positioning

Thuộc tính clear:

- Thuộc tính clear có tất cả 4 thuộc tính:
 - left (tràn bên trái),
 - right (tràn bên phải),
 - both (không tràn) và
 - none.

5.2. Absolute positioning

- Một phần tử ở vị trí tuyệt đối được bố trí liên quan đến cha có vị trí gần nhất.
- Nội dung của phần tử định vị tuyệt đối không bị ảnh hưởng bởi các bất kì box nào.
- Sử dụng giá trị absolute và fixed của thuộc tính position
- Giả sử chúng ta muốn định vị một ảnh ở vị trí 70px cách đỉnh và 90px từ bên trái tài liệu, chúng ta sẽ viết CSS như sau:

```
img { position:absolute; top:70px; left:90px }
```

5.2. Absolute positioning

- Ví dụ: đặt bốn ảnh ở bốn góc tài liệu bằng định vị tuyệt đối.

```
#logo1 { position: absolute; top:50px; left:70px }
```

```
#logo2 { position: absolute; top:0; right:0 }
```

```
#logo3 { position: absolute; bottom:0; left:0 }
```

```
#logo4 { position: absolute; bottom:70px; right:50px }
```

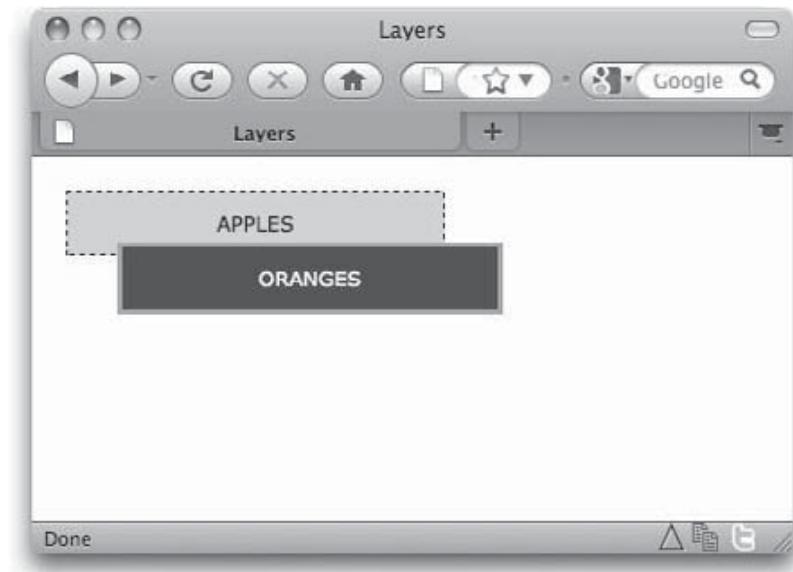
5.3. Layer

- Cách đặt các thành phần web ở các lớp khác nhau với thuộc tính z-index.
- Nói đơn giản hơn thì đó là cách đặt một thành phần này lên trên một thành phần khác.
- Với mục đích này, gán cho mỗi phần tử một con số. Theo đó, phần tử có số cao hơn sẽ nằm trên, phần tử có số thấp hơn sẽ nằm dưới.

5.3. Layer

- Ví dụ:

```
#apples {           #oranges {  
    position: absolute;  position: absolute;  
    top: 20px;          top: 50px;  
    left: 20px;         left: 50px;  
    width: 200px;       width: 200px;  
    z-index: 1;          z-index: 2;  
}  
}
```

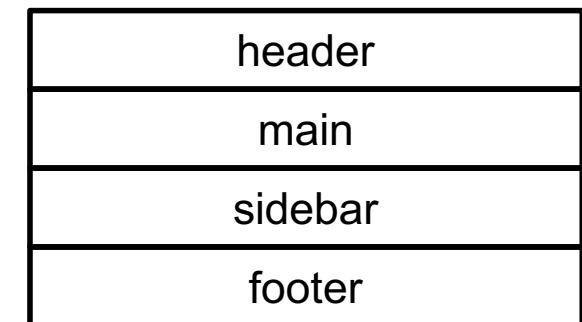


Ví dụ xây dựng layout web đơn giản với thẻ div

- Bộ cục của trang gồm 4 phần
 - header, sidebar, main, footer
- Mỗi phần được xác định bằng thẻ div và thuộc tính id

```
<div id="header">
  <p>Header content</p>
</div>
<div id="sidebar">
  <p>Sidebar content</p>
</div>
<div id="main">
  <h1>Main content</h1>
</div>
<div id="footer">
  <p>Footer content</p>
</div>
```

Trang Web ban đầu



Layout (1): header

- margin và padding thiết lập bằng 0 (**reset CSS**)
- Xác định CSS chung cả trang và cho phần header

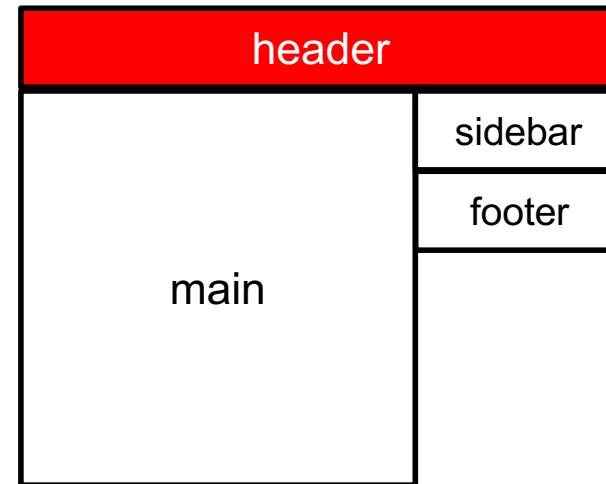
```
* {  
margin: 0;  
padding: 0;  
}  
  
body {  
background-color: white;  
color: black;  
}  
  
div#header {  
background-color: red;  
color: white;  
}
```



Layout (2): main

- Xác định CSS cho phần main

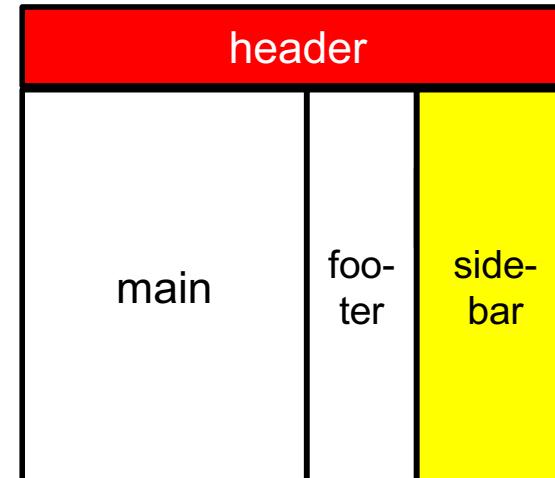
```
div#main {  
    float: left;  
    height: 400px;  
}
```



Layout (3): sidebar

- Xác định CSS cho phần sidebar
 - Thiết lập chiều cao và đẩy sang bên phải
 - Giới hạn chiều rộng sidebar là 25% phần chứa nó

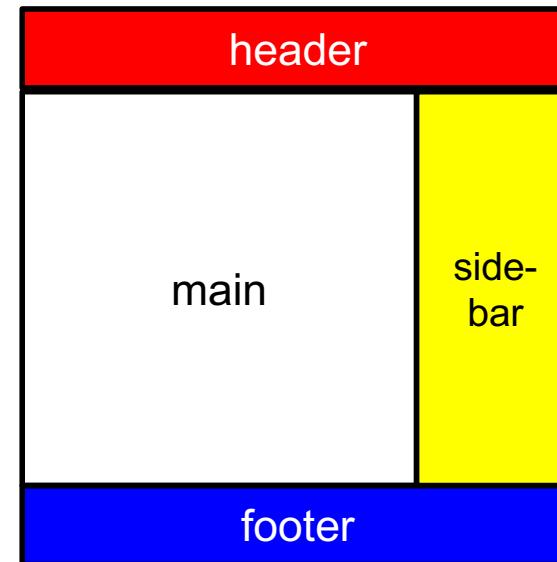
```
div#sidebar {  
    float: right;  
    height: 400px;  
    width: 25%;  
    background-color: yellow;  
    color: black;  
}
```



Layout (4): footer

- Xác định CSS cho phần footer
- Sử dụng “clear: both;”

```
div#footer {  
    clear: both;  
    background-color: blue;  
    color: white;  
}
```



Bài tập 1

- Bạn hãy viết đoạn mã CSS để trình bày style như mô tả dưới đây:
 - Màu nền cho cả trang: #99CC66
 - Font chữ: Arial
 - Tiêu đề của bảng: chữ in nghiêng, màu chữ là màu xanh, màu nền là #CCCCFF

Bài tập 2

- Sửa lại CSS
cho phù hợp



```

<head>
<title>CT 310 Final, Question
3</title>

<link href="../final09q3.css"
      rel="stylesheet"
      type="text/css" />

</head>
<body>
<div id="contents">
    <div id="logo1"> Three <br/>
                           Blind <br/>
                           Mice </div>
    <div id="logo2"> Farmer's <br/>
                           Wife's <br/>
                           Knives </div>
    <div id="toptext">
        Stories for kis? </div>
    <div id="bottomtext"> The End
    </div>
</div>
</body>
</html>
```

```

#contents {
    border-style:solid;
    border-color:#666;
    padding:8px;
    margin-left:auto;
    margin-right:auto;
    margin-top:16px;
    margin-bottom:16px;
    width:80%; }

#logo1 {
    float:left;
    font-weight:bold;
    border-color:black;
    background-color:#CCC;
    padding:6px; }

#logo2 {
    float:right;
    font-weight:bold;
    border-color:black;
    background-color:#CCC;
    padding:6px; }

#toptext {
    text-align:center;
    font-size:medium; }

#bottomtext {
    text-align:center;
    font-size:small; }
```

Bài tập 3

■ Hoàn thiện CSS

```
<html xmlns="http://www.w3.org/1999/xhtml" >
<head>
<title>CT 310 Final, Question 2 - Style 1</title>
<link href=".final09q2style1.css" rel="stylesheet"
      type="text/css" />
</head>
<body>
  <div id="contents">
    <div id="John"> John </div>
    <div id="Paul"> Paul </div>
    <div id="George"> George </div>
    <div id="Ringo"> Ringo </div>
  </div>
</body>
</html>
```

```
#John {
  ...
}
#Paul {
  ...
}
#George {
  ...
}
#Ringo {
  ...
}
```

Bài tập 3

The image shows seven windows arranged in a grid, each titled "CT 310 Final, Question 2 - Style [number]" (1 through 7). Each window contains a different arrangement of the names John, Paul, George, and Ringo.

- Style 1:** All names are stacked vertically on top of each other.
- Style 2:** Names are arranged in two columns: John and Ringo in the first column, and Paul and George in the second column.
- Style 3:** Names are arranged in three columns: John in the first column, George and Ringo in the second column, and Paul in the third column.
- Style 4:** Names are arranged in three columns: John in the first column, George in the second column, and Paul and Ringo in the third column.
- Style 5:** Names are arranged in three columns: John in the first column, Paul in the second column, and George and Ringo in the third column.
- Style 6:** Names are arranged in three columns: John in the first column, Paul in the second column, George in the third column, and Ringo below George.
- Style 7:** Names are arranged in three columns: John in the first column, George in the second column, and Paul and Ringo in the third column.