

Xây dựng ứng dụng Web HTML

NGUYỄN THỊ THÙY LIÊN

KHOA CNTT – ĐH PHENIKAA

LIEN.NGUYENTHITHUY@PHENIKAA-UNI.EDU.VN

Giới thiệu

HTML=HyperText Markup Language –
Ngôn ngữ đánh dấu siêu văn bản –
Ngôn ngữ để viết các trang web.

Do Tim Berner Lee phát minh và được
W3C (World Wide Web Consortium)
đưa thành chuẩn năm 1994.



HTML versions

Version	Year
HTML	1991
HTML+	1993
HTML 2.0	1995
HTML 3.2	1997
HTML 4.01	1999
XHTML	2000
HTML5	2012
HTML5 Recommendation	2014
HTML5.2	2017

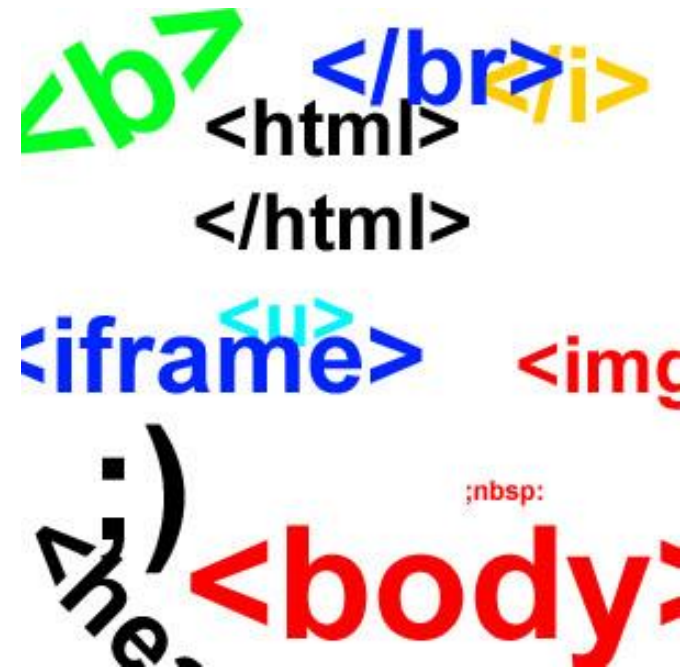


Đặc điểm

HTML sử dụng các thẻ (tags) để **định dạng** dữ liệu

HTML không phân biệt chữ hoa, chữ thường

Các trình duyệt thường không báo lỗi cú pháp HTML. Nếu viết sai cú pháp chỉ dẫn đến kết quả hiển thị không đúng với dự định.



Cấu trúc file HTML

```
<html>
```

```
<head>
```

```
<title>Page title</title>
```

```
</head>
```

```
<body>
```

```
<h1>This is a heading</h1>
```

```
<p>This is a paragraph.</p>
```

```
<p>This is another paragraph.</p>
```

```
</body>
```

```
</html>
```

Khai báo doctype

Khai báo doctype giúp trình duyệt hiển thị trang web chính xác.

HTML5

```
<!DOCTYPE html>
```

HTML 4.01

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"  
"http://www.w3.org/TR/html4/loose.dtd">
```

XHTML 1.0

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

Thẻ (tag)

Có nhiều thẻ, mỗi thẻ có 1 tên và mang ý nghĩa khác nhau.

<tagname> content **</tagname>**

Có 2 loại thẻ: thẻ đóng và thẻ mở

Cách viết thẻ:

- Thẻ mở: **<tên_thẻ>**

Ví dụ: <u>, <p>, ...

- Thẻ đóng tương ứng: **</tên_thẻ>**

Ví dụ: </u>, </p>

Chú ý: luôn có thẻ mở nhưng có thể không có thẻ đóng tương ứng. Ví dụ: ,
, <hr>, <meta>... không có thẻ đóng



Thẻ (tag) – thẻ HTML

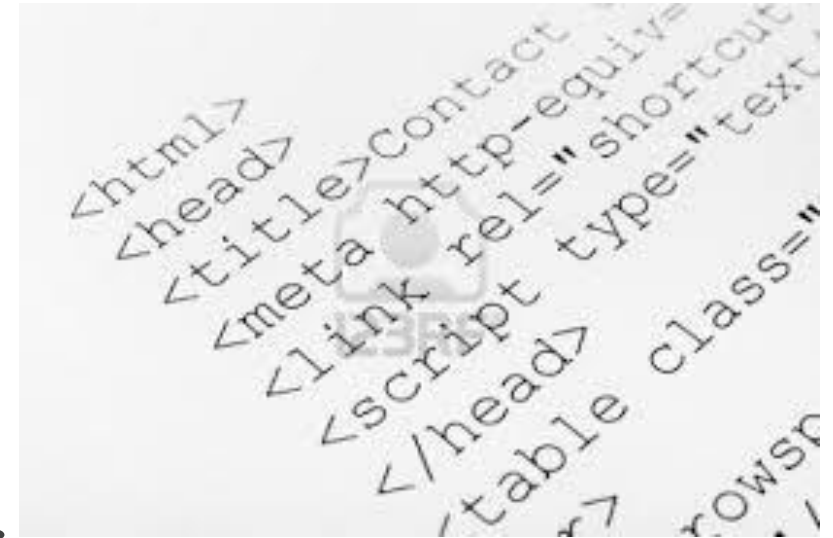
Thẻ **<html>...</html>** cho biết đây là tài liệu có định dạng HTML

```
1 <html>
2   <head>
3     <title>Lập trình mạng</title>
4   </head>
5   <body>
6     <h1>HTML cơ bản</h1>
7   </body>
8 </html>
```


Thẻ (tag) – thẻ HEAD

Thẻ **<head>...</head>** : chứa một số thông tin của trang:

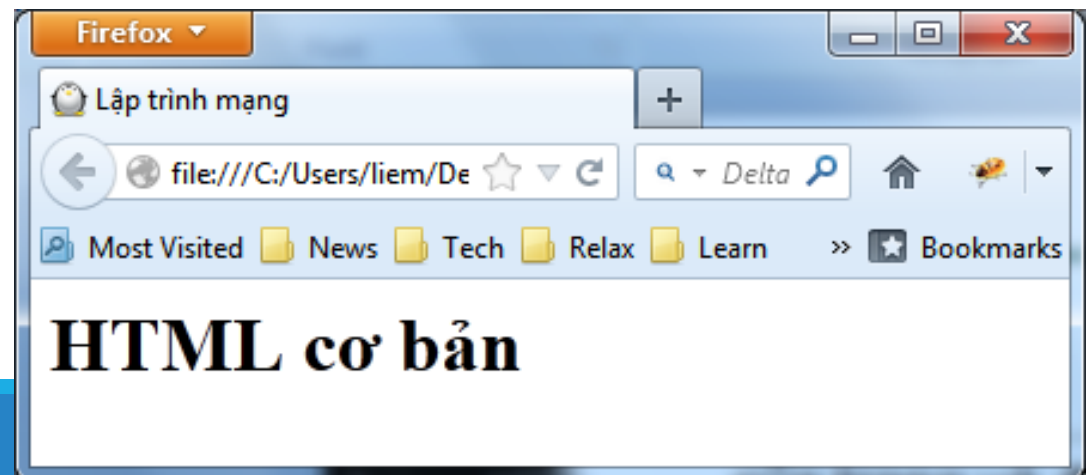
- tiêu đề: **<title>...</title>**
- Các thẻ mở rộng
- các đường link tới một số file khác



Nội dung trong thẻ head không được hiển thị trong cửa sổ trình duyệt.

Thẻ (tag) – thẻ HEAD

```
1 <html>
2   <head>
3     <title>Lập trình mạng</title>
4     <meta http-equiv="content-type" content = "text/html; charset=utf-8">
5     <meta name="title" content = "Lập trình mạng">
6     <link href="images/icon.ico" rel = "shortcut icon" type="image/x-icon">
7     <link rel="stylesheet" href="css/system.css" type="text/css">
8     <script type="text/javascript" src="js/common.js"></script>
9   </head>
10  <body>
11    <h1>HTML cơ bản</h1>
12  </body>
13 </html>
```



Một số thẻ meta thông dụng

Thẻ **<meta>**:

- Đặt ở giữa <head>...</head>
- Thường dùng quy định thuộc tính cho trang web
- Có tác dụng lớn với Search Engine
- 2 cách viết thẻ <meta>:

```
<META NAME="name" CONTENT="content">
```

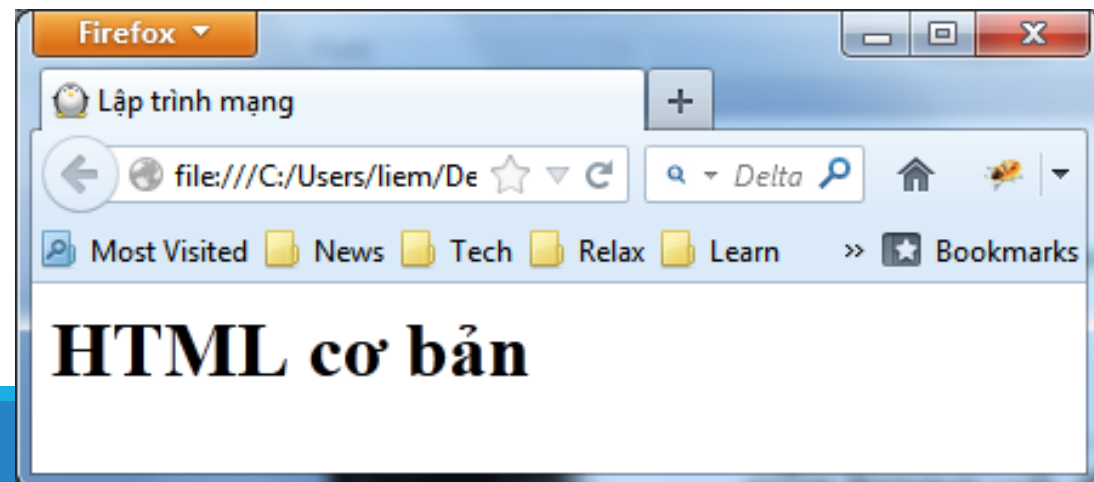
```
<META HTTP-EQUIV="name" CONTENT="content">
```

- `<META NAME="description" content="">`
- `<META NAME="keywords" content="">`
- `<META NAME="author" CONTENT="author's name">`
- `<META charset=utf-8">`

Thẻ (tag) – thẻ BODY

Thẻ **<body>...</body>** : chứa toàn bộ nội dung của trang và được hiển thị trên cửa sổ trình duyệt.

```
9      </head>
10     <body>
11         <h1>HTML cơ bản</h1>
12     </body>
13 </html>
```



Thuộc tính (property) của thẻ

Một thẻ có thể có các thuộc tính nhằm bổ sung tác dụng cho thẻ

Mỗi thuộc tính có tên thuộc tính (tên_TT)

Các thuộc tính đặt trong thẻ mở

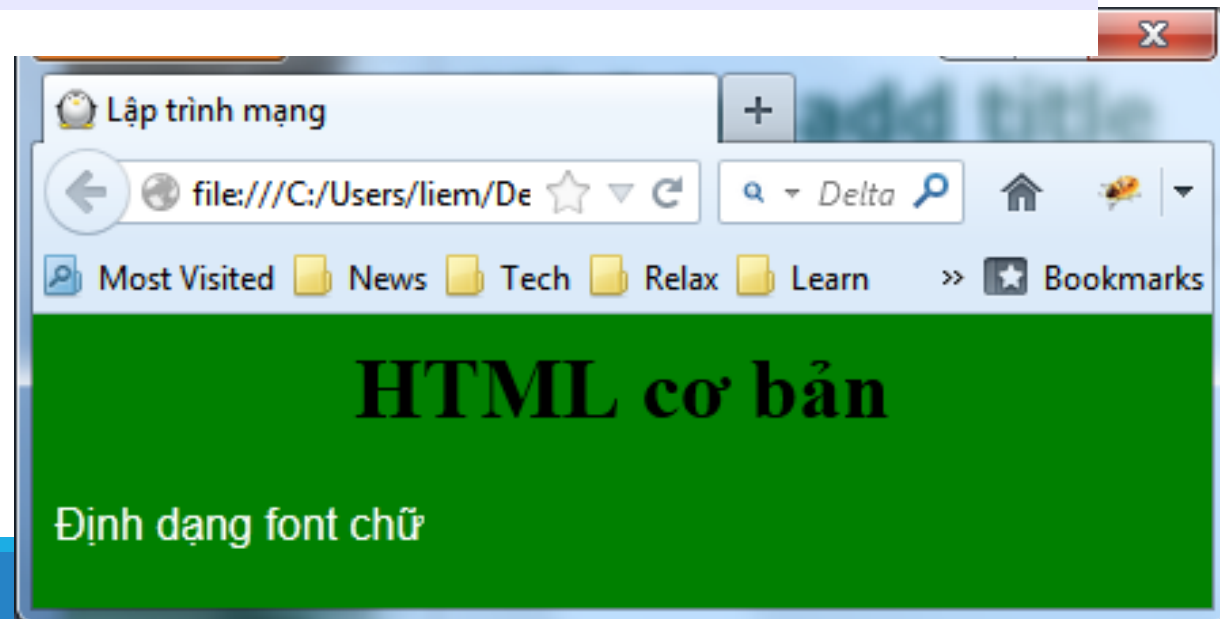
Viết thẻ có thuộc tính:

<tên_thẻ tên_TT1="giá_trị1" tên_TT2="giá_trị2"...>

```
10 <body bgcolor="green" >
11     <h1 align="center" >HTML cơ bản</h1>
12     <font color="white" face="arial"> Định dạng font chữ</font>
13 </body>
14 </html>
```

Thuộc tính (property) của thẻ

```
10 <body bgcolor="green" >
11     <h1 align="center" >HTML cơ bản</h1>
12     <font color="white" face="arial"> Định dạng font chữ</font>
13 </body>
14 </html>
```



Thuộc tính (property) của thẻ

Chú ý:

- Có thể thay đổi thứ tự, số lượng các thuộc tính mà không gây ra lỗi cú pháp

```
10 <body bgcolor="green" >
11     <h1 align="center" >HTML cơ bản</h1>
12     <font color="white"> Định dạng font chữ 1</font>
13     <font color="white" face="arial"> Định dạng font chữ 2</font>
14     <font face="arial" color="white"> Định dạng font chữ 3</font>
15 </body>
```

- Sự hỗ trợ các thẻ, thuộc tính ở mỗi trình duyệt là khác nhau. Chỉ giống nhau ở các thẻ, thuộc tính cơ bản.
- Thẻ đóng của thẻ có thuộc tính vẫn **viết bình thường** (</tên_thẻ>)

Trang web đầu tiên

Trang HTML có phần mở rộng (đuôi) là **.HTM** hoặc **.HTML**

Có thể tạo trang HTML bằng bất cứ trình soạn thảo “văn bản thuần” nào (Notepad, EditPlus, Turbo Pascal,...)

Có nhiều trình soạn thảo HTML cho phép NSD soạn thảo trực quan, kết quả sinh ra HTML tương ứng như:

- Microsoft FrontPage
- Macromedia Dreamweaver
- ...



Trang web đầu tiên (tt)

Soạn thảo:

- Mở trình soạn thảo văn bản thuần (VD Notepad) gõ đoạn mã dưới
- Ghi lại với tên “**CHAO.HTM**”

```
<html>

    <head>

        <title>Chao hoi</title>

    </head>

    <body>

        Chao mung ban den voi <U>HTML</U>!

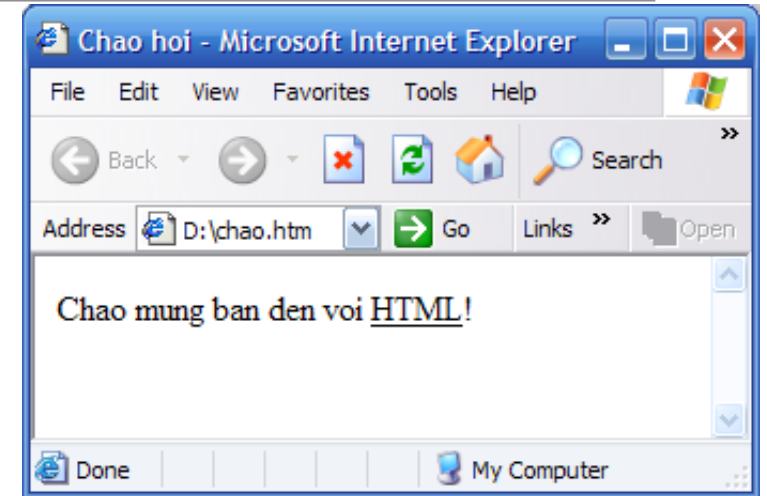
    </body>

</html>
```

Trang web đầu tiên (tt)

Thử nghiệm:

- Mở trình duyệt web (IE, firefox,chrome...)
- Vào **File/Open**, chọn file CHAO.HTM vừa ghi
- Nhấn **OK** → Có kết quả như hình bên



Thay đổi:

- Quay lại Notepad, sửa lại nội dung trang web rồi ghi lại
- Chuyển sang IE, nhấn nút Refresh (F5) → thấy kết quả mới



Ghi chú: Các thẻ được nêu tiếp theo mặc định đặt ở trong phần `<body>...</body>`

Soạn thảo văn bản

Văn bản được soạn thảo như bình thường trong các file HTML

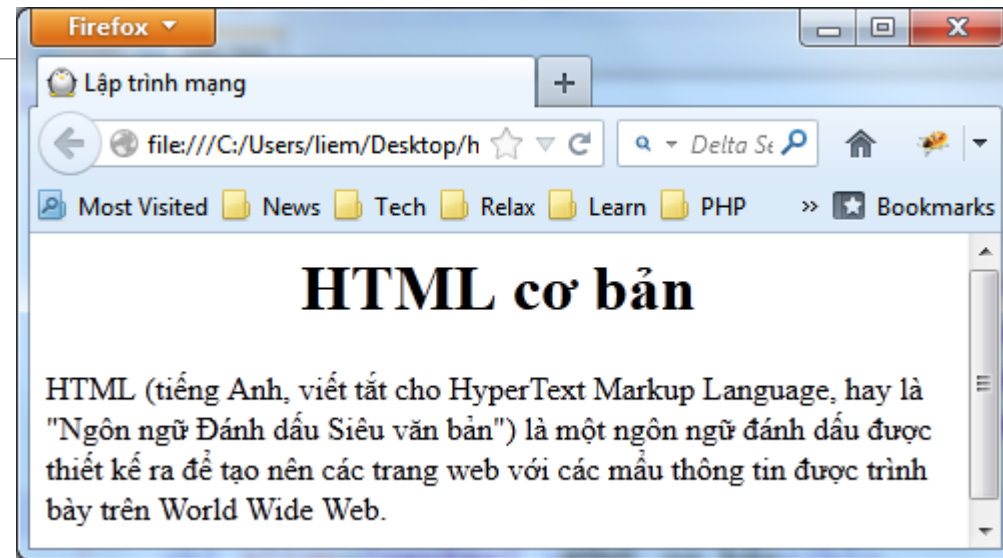
Lưu ý:

- Mọi khoảng trống, dấu xuống dòng trong HTML được thể hiện trên trang web là **1 khoảng trống duy nhất**
- Để gõ một số ký tự đặc biệt ta phải sử dụng mã:
 - Khoảng trống (trong trường hợp muốn có nhiều hơn 1 ký tự trống): ** **

Ký tự đặc biệt

Symbol Name	HTML Entity	Symbol
Copyright Sign	©	©
Registered Trademark Sign	®	®
Trademark Sign	™	™
Less Than	<	<
Greater Than	>	>
Ampersand	&	&
Non-breaking Space	 	
Em Dash	—	—
Quotation Mark	"	"
Euro	€	€
British Pound	£	£
Japanese Yen	¥	¥

Soạn thảo văn bản



```
12  <p>
13      HTML
14      (tiếng Anh, viết tắt cho HyperText Markup Language,
15      hay là "Ngôn ngữ Đánh dấu Siêu văn bản")
16      là một ngôn ngữ      đánh dấu được thiết kế
17      ra để tạo nên các trang web với các mẫu thông tin
18  </p>
```

Chú thích trong HTML

Ghi chú trong HTML:

<!-- Ghi nội dung chú thích ở đây-->

```
10 <!-- Nội dung trang web -->
11 <body>
12     <!-- begin giới thiệu HTML -->
13     <h1 align="center" >HTML cơ bản</h1>
14     <p>
15         HTML
16         (tiếng Anh, viết tắt cho HyperText Markup Language
17         hay là "Ngôn ngữ Đánh dấu Siêu văn bản")
18         là một ngôn ngữ đánh dấu được thiết kế
19         ra để tạo nên các trang web với các mẫu thông
20     </p>
21     <!-- end giới thiệu HTML -->
22 </body>
```

Các thẻ cơ bản HTML

NGUYỄN THỊ THÙY LIÊN

Thẻ body và thuộc tính

Thẻ **<body> </body>** : chứa nội dung của file HTML

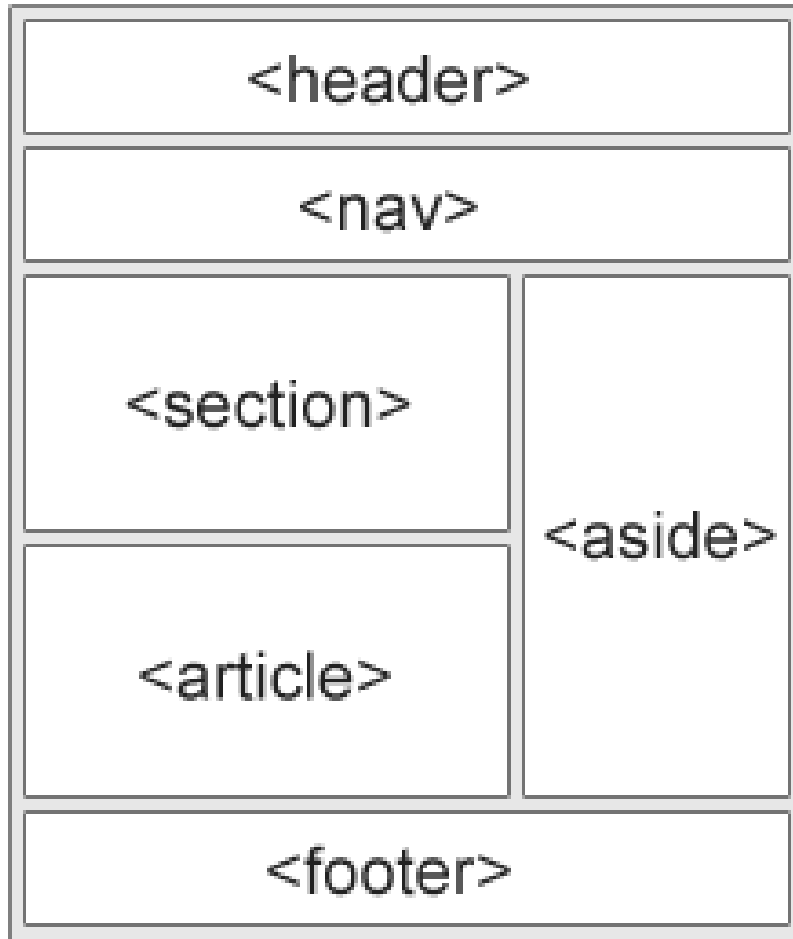
Thuộc Tính :

- **background**: Dùng cho định dạng file hình ảnh làm nền (.gif, .jpg, .bmp).
- **bgcolor**: Xác lập màu cho nền.
- **text**: màu chữ.
- **link**: màu cho liên kết chưa xem.
- **vlink**: màu cho liên kết đã xem.
- **alink**: màu cho liên kết đang xem.
- **leftmargin**: Canh lề trái.
- **topmargin**: Canh lề trên.

Thẻ body và thuộc tính

[illegible]

Các phần tử bố cục



```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>HTML5 Skeleton</title>
    <meta charset="utf-8">
  </head>
  <body>
    <header>
      <h1>HTML5 SKeleton</h1>
    </header>
    <nav>
      <ul>
        <li><a href="">HTML5 Semantic</a></li>
        <li><a href="">HTML5 Geolocation</a></li>
      </ul>
    </nav>
    <section>
      <h1>Famous Cities</h1>
      <article><p>some thing there</p></article>
      <article><p>some thing there</p></article>
    </section>
    <footer>
      <p>HTML5</p>
    </footer>
  </body>
</html>
```

Phần tử <header>

<header>

- Phần đầu của trang hoặc của section
- Có thể có nhiều <header> trong trang

```
<article>
  <header>
    <h1>heading of article</h1>
    <p>datetime:</p>
  </header>
  <p>content of article</p>
</article>
```

Phần tử <footer>

<footer>

- Phần chân của trang hoặc của section
- Có thể có nhiều <header> trong trang

```
<article>
  <header>
    <h1>heading of article</h1>
    <p>datetime:</p>
  </header>
  <p>content of article</p>
  <footer>
    <p>Posted by: Hege Refsnes</p>
  </footer>
</article>
```

Phần tử <nav>

<nav>

- Phần điều hướng trang web

```
<nav>  
  <a href="/index.html"> Trang chủ</a> |  
  <a href="/news.html"> Tin tức</a> |  
  <a href="/map.html"> Bản đồ</a> |  
  <a href="/contact.html"> Liên hệ</a>  
</nav>
```

Phần tử <main>

<main>

- Phần nội dung chính của trang
- Mỗi trang chỉ nên có 1 main

```
<main>
  <h1>Web Browsers</h1>
  <p>Google Chrome, Firefox, and Internet Explorer .</p>

  <article>
    <h1>Internet Explorer</h1>
    <p>Internet Explorer is a free web browser</p>
  </article>

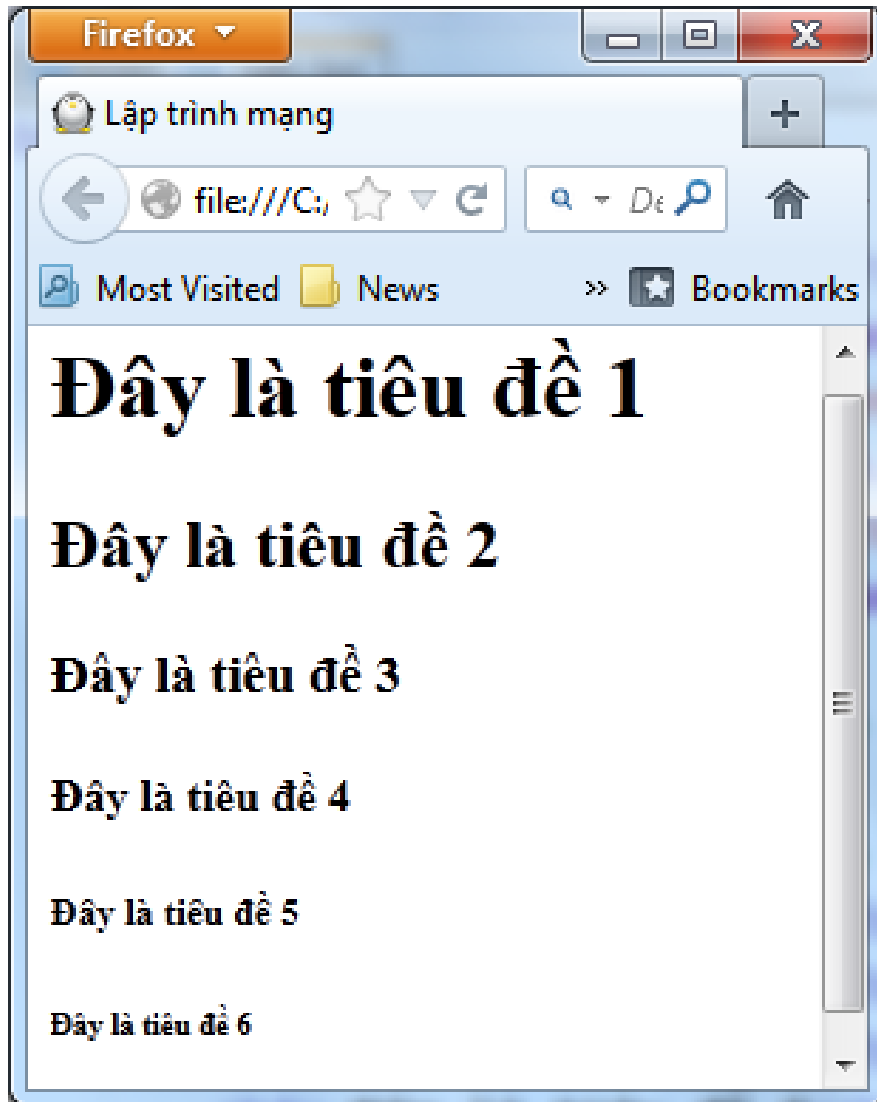
  <section>
    <h1>Mozilla Firefox</h1>
    <p>Firefox is a free, open-source web browser</p>
  </section>
</main>
```

Phần tử <aside>

<aside>

- Phần nội dung bên lề, mở rộng, liên quan đến nội dung chính

```
<p>Main content</p>
<aside>
  <h4>heading of aside content</h4>
  <p>aside content</p>
</aside>
```



Các thẻ tiêu đề - Headings

Headings được định dạng với các thẻ:

`<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>`, `<h6>`

- Trước và sau mỗi tiêu đề văn bản tự động xuống dòng
- Kích thước nhỏ dần từ h1 đến h6
- Sử dụng làm tiêu đề cho bài viết

***Thuộc tính:**

+ **align** = “căn chỉnh lề”: giá trị: "left", "right", "center", "justify"

Các thẻ định dạng văn bản, kí tự

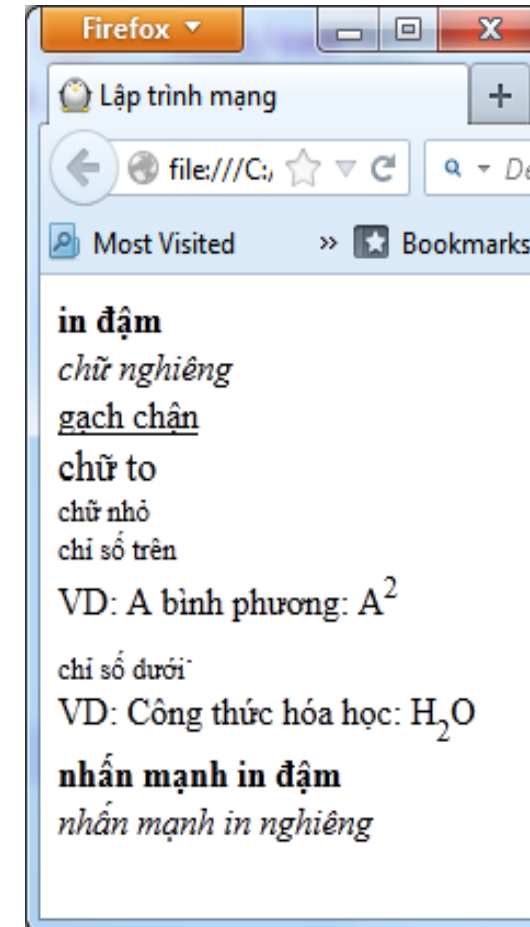
Tên thẻ	Định dạng
...	chữ đậm
<i>...</i>:	Chữ nghiêng
<u>....</u>	Chữ gạch chân
<big>..</big>	Chữ to
<small>...</small>	Chữ nhỏ
^{...}	Chỉ số trên
_{...}	Chỉ số dưới
...	Nhấn mạnh in đậm
...	Nhấn mạnh in nghiêng

Các thẻ định dạng văn bản, kí tự

Tên thẻ	Định dạng
<code><mark>..</mark></code>	Đánh dấu, highlight
<code>...</code>	Chữ bị gạch ngang
<code><address>...</address></code>	Thông tin tác giả, địa chỉ
<code><code>....</code></code>	Computer code
<code><kbd>....</kbd></code>	Keyboard input
<code><pre>...</pre></code>	Văn bản định dạng trước
<code><var>...</var></code>	Biến trong toán học
<code><q>...</q></code>	Quote

Các thẻ định dạng văn bản, kí tự

```
11 <body>
12     <b>in đậm</b> <br>
13     <i>chữ nghiêng</i><br>
14     <u>gạch chân</u> <br>
15     <big>chữ to</big><br>
16     <small>chữ nhỏ</small><br>
17     <sup>chỉ số trên</sup> <br>
18     VD: A bình phương: A<sup>2</sup> <br>
19     <sub>chỉ số dưới</sub>.<br>
20     VD: Công thức hóa học: H<sub>2</sub>O<br>
21     <strong>nhấn mạnh in đậm</strong><br>
22     <em>nhấn mạnh in nghiêng</em><br>
23
24 </body>
```



Các thẻ định dạng văn bản, kí tự

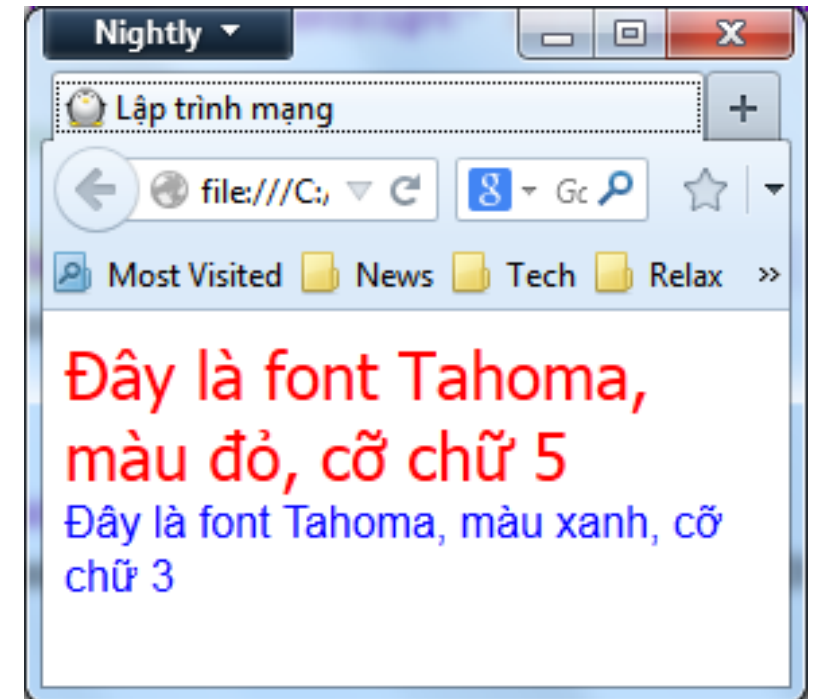
...: định dạng font chữ cho văn bản

* Thuộc tính:

- **face** = “tên font chữ” : .VnTime, Times New Roman, Arial
- **Size** = ”kích thước” : giá trị 1->7 mặc định là 3
- **Color** = “màu chữ”
 - Viết bằng tên tiếng Anh (red, blue,...)
 - Viết dạng #RRGGBB, RR, GG, BB ở dạng hexa. Ví dụ: #FFFFFF: Trắng, #FF0000: đỏ,...

Các thẻ định dạng văn bản, kí tự

```
12 <font face="Tahoma" size="5" color="red">  
13     Đây là font Tahoma, màu đỏ, cỡ chữ 5  
14 </font>  
15 <br>  
16 <font face="Arial" color="#0000FF">  
17     Đây là font Tahoma, màu xanh, cỡ chữ 3  
18 </font>
```



Phân đoạn và ngắt quãng văn bản

Thẻ **<p>..</p>**: định dạng đoạn văn bản

- Thuộc tính:

+ **align** = “căn chỉnh lề ”: giá trị: "left", "right", "center", "justify”

Thẻ **
**: sang dòng mới

Thẻ **<center>..</center>**: Định dạng hiển thị giữa trang

Thẻ **<div>...</div>**, **...**: Tạo khối cho văn bản

- Thuộc tính:

+ **align** = “căn chỉnh lề ”: giá trị: "left", "right", "center", "justify”

Phân đoạn và ngắt quãng văn bản

Thẻ **<hr>**: Tạo đường kẻ ngang

Thuộc tính:

- Align: canh lề với giá trị center, right, left
- Width: chỉ độ dài của đường thẳng: giá trị bằng pixel hoặc %. Mặc định 100%
- Size: chỉ độ dày của đường thẳng
- Noshade: chỉ đường thẳng được hiển thị bằng màu đặc thay vì có bóng.
- Color: màu đường kẻ

Danh sách

Danh sách dùng để liệt kê các phần tử

Một danh sách có nhiều phần tử, mỗi phần tử có thể là một danh sách con

Có 3 loại danh sách:

- Danh sách có thứ tự (**ordered list - ol**)
- Danh sách không có thứ tự (**unordered list -ul**)
- Danh sách mô tả
(**description list – dl**)

Unordered HTML List

- The first item
- The second item
- The third item
- The fourth item

Ordered HTML List

1. The first item
2. The second item
3. The third item
4. The fourth item

HTML Description List

The first item
Description of item
The second item
Description of item

Danh sách không có thứ tự

Danh sách không có thứ tự (**U**nordered **L**ist) tạo các danh sách các mục có bullet

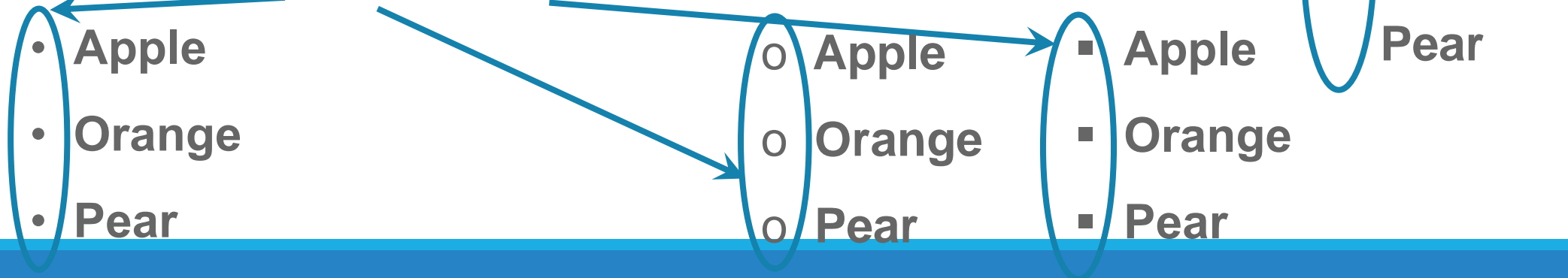
Thẻ **....**

Các phần tử: thẻ **....**

```
<ul type="disk">
  <li>Apple</li>
  <li>Orange</li>
  <li>Grapefruit</li>
</ul>
```

Thuộc tính:

- Type="kiểu bullet": disc, circle, square, none



Danh sách có thứ tự

```
<ol type="1">  
  <li>Apple</li>  
  <li>Orange</li>  
  <li>Grapefruit</li>  
</ol>
```

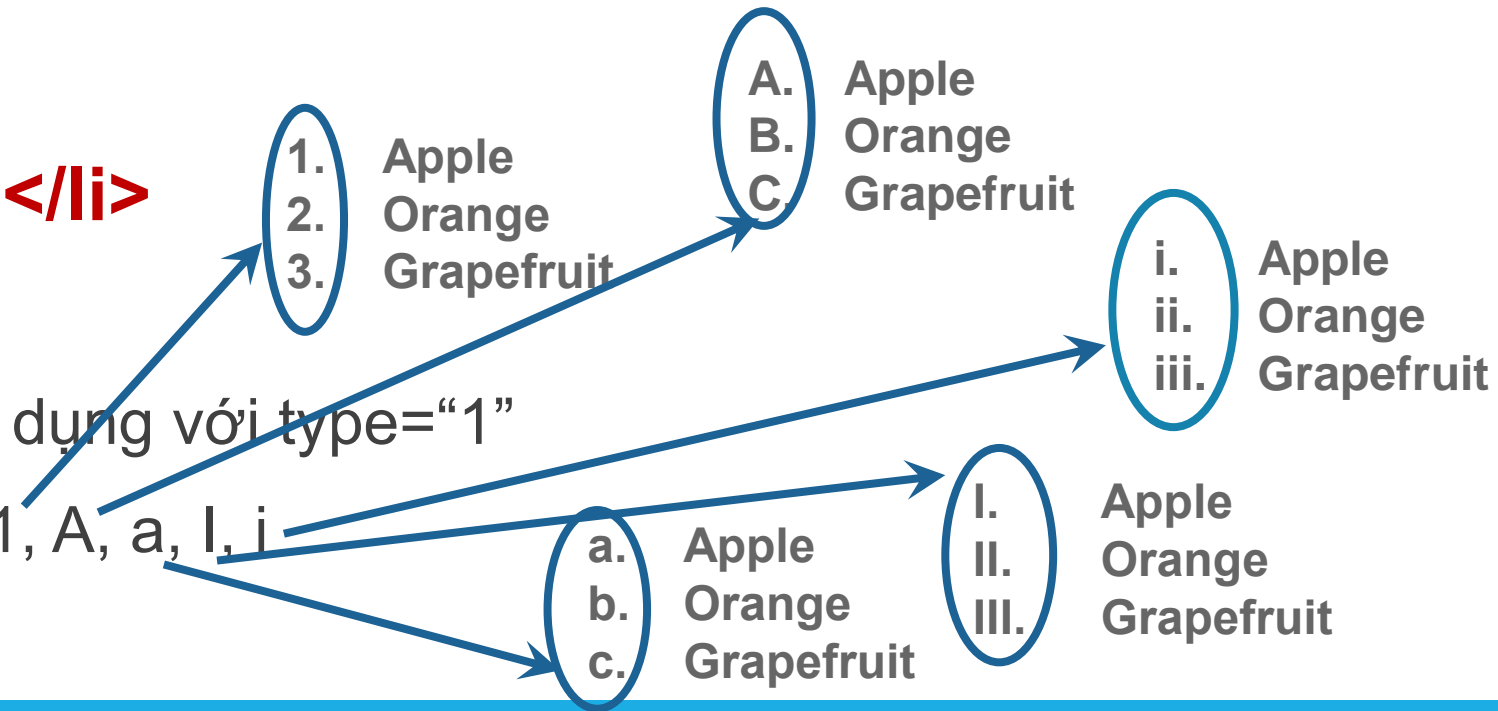
Danh sách có thứ tự (**O**rdered **L**ist) tạo các danh sách các mục có đánh số thứ tự

Thẻ **....**

Các phần tử: thẻ **....**

Thuộc tính:

- start = “giá trị bắt đầu”: áp dụng với type=“1”
- Type=“kiểu đánh thứ tự”: 1, A, a, I, i



Danh sách mô tả

Danh sách mô tả (**Description List**) tạo các danh sách với phần mô tả ở mỗi phần tử

Thẻ **<dl>....</dl>**

Các phần tử:

- thẻ **<dt>....</dt>** tiêu đề
- thẻ **<dd>...</dd>** mô tả

```
1 <dl>
2   <dt>Coffee</dt>
3   <dd>- black hot drink</dd>
4   <dt>Milk</dt>
5   <dd>- white cold drink</dd>
6 </dl>
```

A Description List

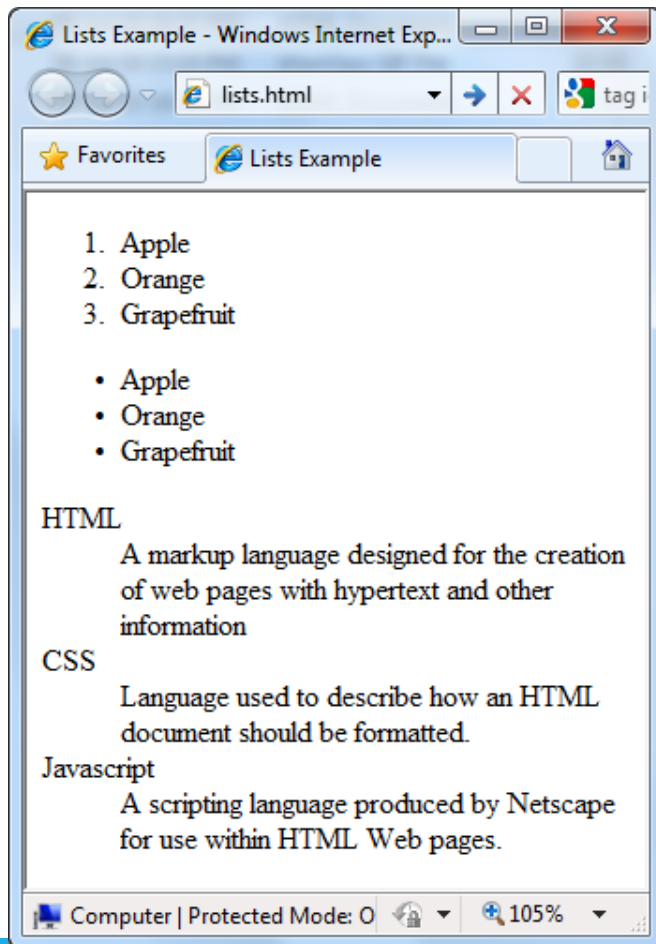
Coffee

- black hot drink

Milk

- white cold drink

Lists – Ví dụ



```
<ol type="1">
  <li>Apple</li>
  <li>Orange</li>
  <li>Grapefruit</li>
</ol>
```

```
<ul type="disc">
  <li>Apple</li>
  <li>Orange</li>
  <li>Grapefruit</li>
</ul>
```

```
<dl>
  <dt>HTML</dt>
  <dd>A markup language ...</dd>
  <dt>CSS</dt>
  <dd>Language used to ...</dd>
</dl>
```

Chèn ảnh

Thẻ **** không có thẻ đóng

Thuộc tính

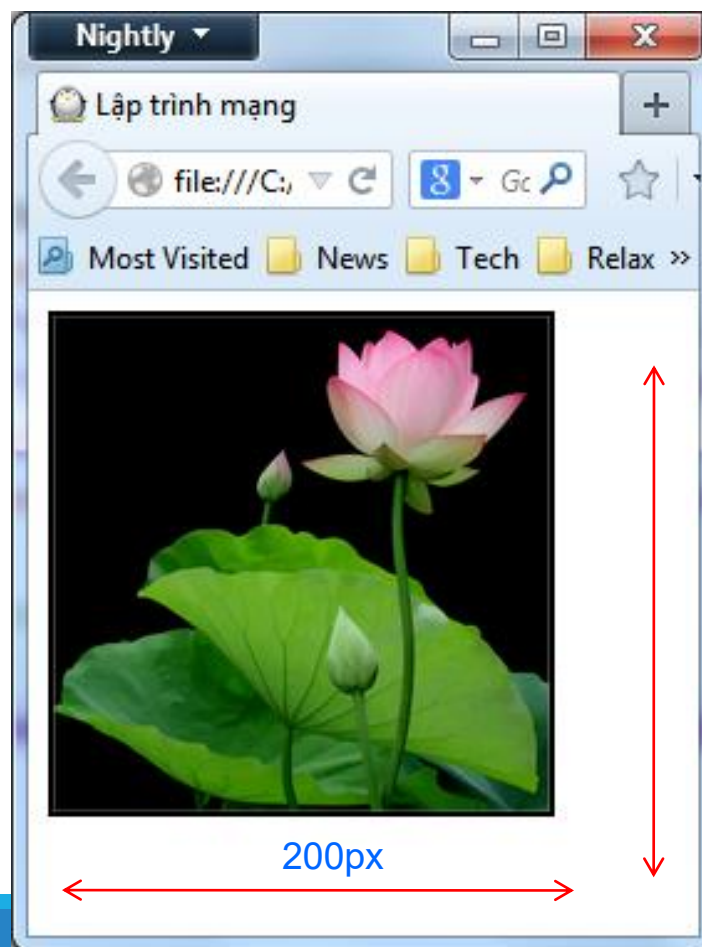
- **src**="địa chỉ ảnh": Nếu chèn ảnh trong cùng web site thì nên sử dụng đường dẫn tương đối.
- **alt**="chú thích cho ảnh": sẽ được hiển thị khi trình duyệt không hiện ảnh hoặc ảnh lỗi hoặc khi di chuyển chuột lên ảnh
- **width**="rộng", **height**="cao": độ rộng và độ cao của ảnh:
 - n: (n là số) Quy định độ rộng, cao là n pixels
 - n%: Quy định độ rộng, cao là n% độ rộng, cao của đối tượng chứa ảnh.
- **border**="n": n là số: kích thước đường viền ảnh. =0: ảnh không có đường viền
- **align**="căn chỉnh ảnh": left, right, top, texttop, middle, absmiddle, baseline, bottom, absbottom

Chèn ảnh

12

```

```



200px

200px

Vùng ảnh - <figure>

<figure> and <figcaption>

- <figure>: vùng ảnh và tiêu đề ảnh
- <figcaption>: Tiêu đề ảnh, tên ảnh

```
<figure>  
    
  <figcaption>Fig1. - name of figue.</figcaption>  
</figure>
```

Chèn image maps

Thẻ **<map>**: tạo ra một bức ảnh có thể click trên từng vùng khác nhau

- Thuộc tính:
 - name

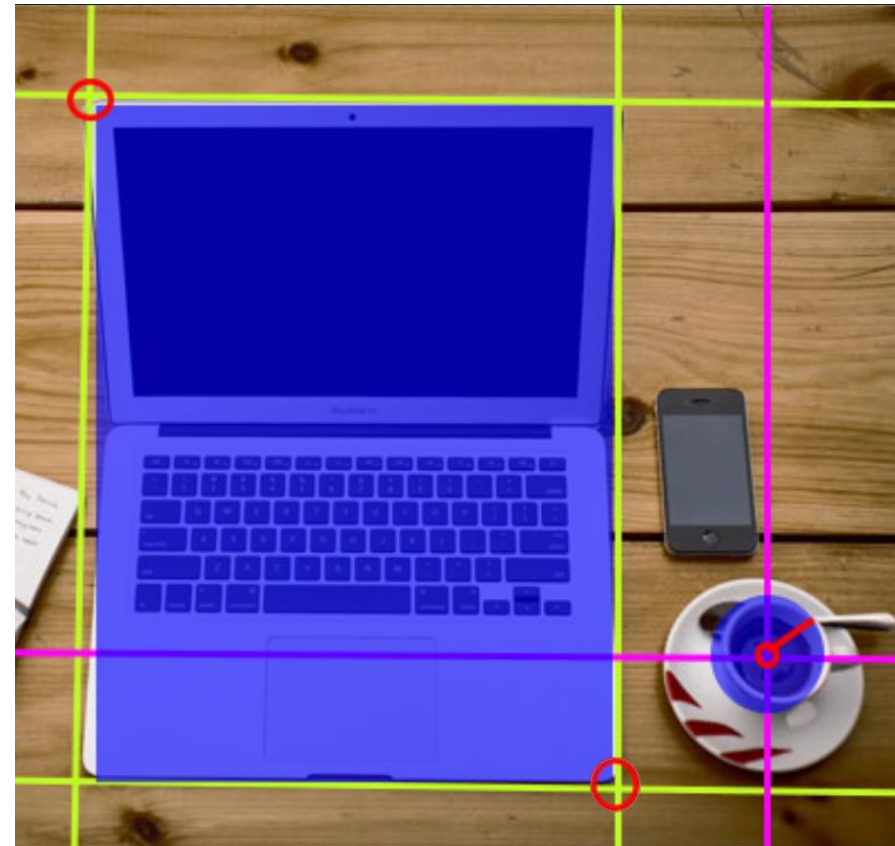
Thẻ **<area>**: một vùng có thể click trên bức ảnh

- Thuộc tính:
 - Shape: rect, circle, poly, default
 - coords: giới hạn vùng
 - href: đích liên kết

Chèn image maps

```


<map name="workmap">
  <area shape="rect"
        coords="34,44,270,350"
        alt="Computer"
        href="computer.html">
  <area shape="circle"
        coords="337,300,44"
        alt="Coffee"
        href="coffeel.htm">
</map>
```



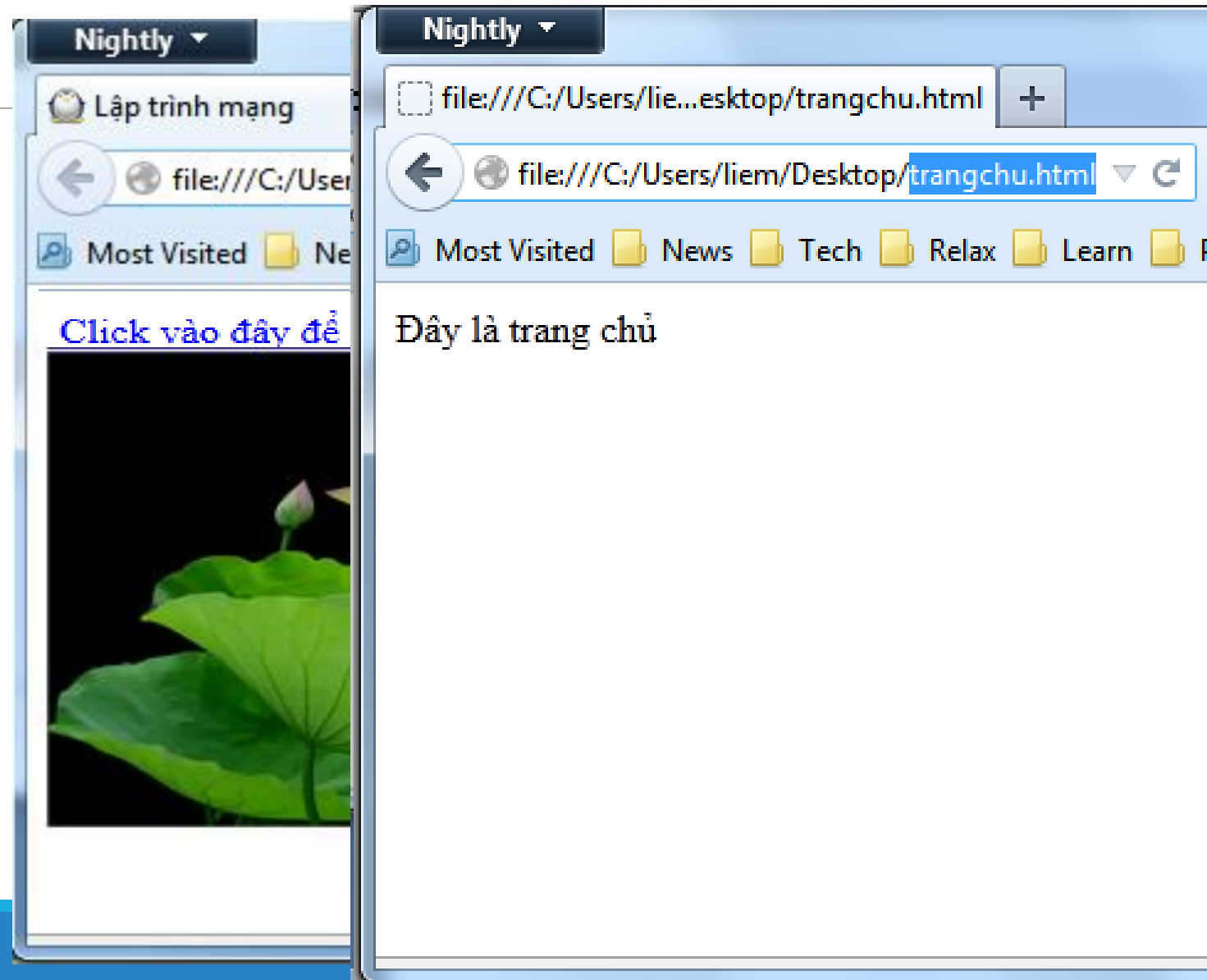
Liên kết trang web

Thẻ **<a>...** dùng để tạo một liên kết tới một trang web khác.

Thuộc tính:

- Href = “đích liên kết”: Địa chỉ url trang web cần liên kết tới
- **target**=“tên cửa sổ đích”. Tên CS phân biệt chữ hoa/thường. Có một số tên đặc biệt:
 - **_self**: cửa sổ hiện tại (mặc định) **_parent**: parent frame
 - **_blank**: cửa sổ mới **_top**: cửa sổ chính
- Chú ý:
 - Liên kết với địa chỉ e-mail thì đặt **href=“mailto:địa_chỉ_e-mail”**
 - Thực hiện lệnh JavaScript khi kích chuột vào thì đặt **href=“javascript:lệnh”**

Liên kết trang web



Liên kết trang web

```
12 <a href="trangchu.html"> Click vào đây để chuyển sang trang chủ</a>  
13 <br>  
14 <a href="tintuc.html">  
15       
16 </a>
```

Click vào đây để chuyển sang trang chủ



Liên kết đến các phần trang web

Còn gọi là kiểu liên kết “**named anchor**”

Được sử dụng để nhảy đến những phần khác nhau của trang web.

Sử dụng thuộc tính name trong thẻ neo a để đánh dấu điểm đích đến

```
<a name/id=“topicname”> Topic name </a>
```

Sử dụng dấu #{vị trí đến} làm giá trị của href của liên kết

```
<a href = “#topicname”>Topic name</a>
```

```
<a href = “test.html#topicname”>Topic name</a>
```

Tiến trình

Thẻ **<progress>**: biểu diễn tiến độ hoàn thành của một nhiệm vụ

- max: tổng số công việc mà nhiệm vụ yêu cầu
- value: mức độ hoàn thành



```
2 <label for="file">Lập trình web:</label>
3 <progress id="file" value="80" max="100"> Good </progress>
4 <label for="file">Lập trình Java:</label>
5 <progress id="file" value="70" max="100"> Good </progress>
6 <label for="file">Lập trình .NET:</label>
7 <progress id="file" value="90" max="100"> Expert </progress>
```

Title	Title	Title	Title	Title	Title
Data	Data	Data	Data	Data	Data
Data	Data	Data	Data	Data	Data
Data	Data	Data	Data	Data	Data
Data	Data	Data	Data	Data	Data
Data	Data	Data	Data	Data	Data

City	European date (DMS) & time	Y-M-D date & time	Offset	Change money	IP address	Name	Numbers
...
Fri Mar 22 21:48:49 UTC+0200 1997	01/03/1997	1997-03-22 06:07:56 PM	YMD -03:30	162,117,273.34
Sat 14 Feb 2002 04:24:20 UTC	06/07/99 06:40:01 AM	YMD -08:00	122,205,50.8
Sunday, Mar 16, 1996 4:47:13 PM	06/09/01 05:17:38 AM	YMD -11:36
06/28/2000	04/11/1997
Mon, 29 Oct 1979 00:44:01 UTC	...	07-08-13 00:01:33 AM
Sat, 9 Jan 1942 05:41:06 UTC	04/06/08	74-10-20
06/05/78	...	2006-05-20
Sunday, July 15, 2002 1:09:02 AM	01/02/1961 09:40:16 AM	007-9
Black River

```

htmltable1 - Notepad
File Edit Format View Help

<html>
<head>
<title>How To Create HTML Tables</title>
</head>
<body>
<table border=1 cellspacing=0 cellpadding=0>
<tr>
<td width= 110 valign=top>
<br>upper left corner
</td>
<td width=110 valign=top>
<br>upper right corner
</td>
</tr>
<tr>
<td width= 110 valign=top>
<br>left center cell
</td>
<td width=110 valign=top>
<br>right center cell
</td>
</tr>
<tr>
<td width= 110 valign=top>
<br>lower left corner
</td>
<td width=110 valign=top>
<br>lower right corner
</td>
</tr>
</table>
</body>
</html>

```

HTML Tables

Bảng biểu

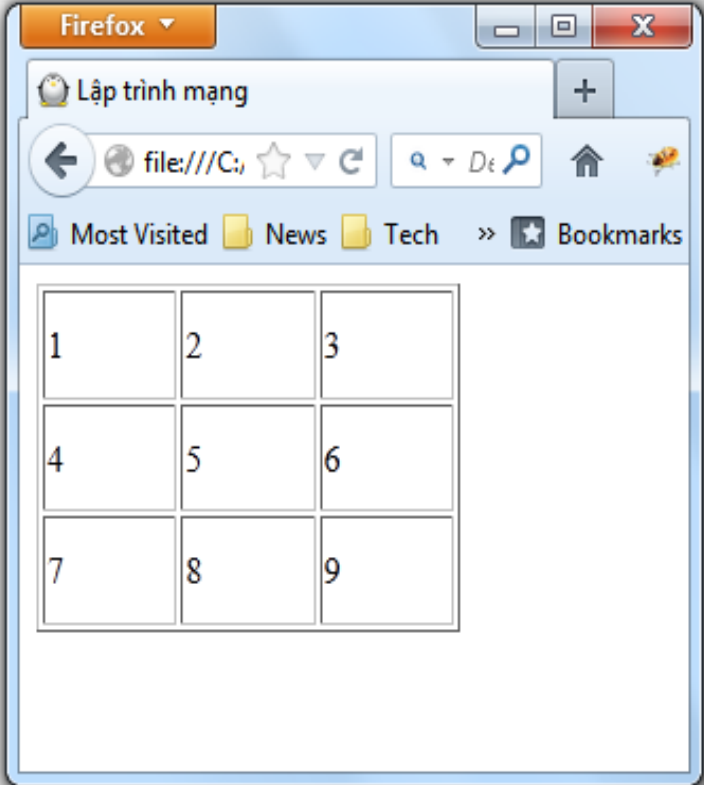
HTML coi một *bảng* gồm *nhiều dòng*, một *dòng* gồm *nhiều ô*, và *chỉ* có ô mới chứa dữ liệu của bảng.

Các thẻ:

- Tạo bảng: **<table>...</table>**: Mỗi bảng chỉ có 1 cặp thẻ này.
- Tạo dòng: **<tr>...</tr>**: Bảng có bao nhiêu dòng thì có bấy nhiêu cặp thẻ này
- Tạo ô:
 - Ô tiêu đề của bảng: **<th>...</th>**
 - Ô dữ liệu: **<td>...</td>**Tổng số thẻ **<td>** và **<th>** bằng số ô của bảng. Dòng có bao nhiêu ô thì có bấy nhiêu thẻ **<td>** và/hoặc **<th>** nằm trong cặp thẻ **<tr>...</tr>** tương ứng
- Chú ý: Để có được một ô trống trong bảng (ô không có dữ liệu) thì cần đặt nội dung ô là: ** **

Bảng biểu

```
12 <table border="1" width="200" height="150">
13   <tr>
14     <td>1</td>
15     <td>2</td>
16     <td>3</td>
17   </tr>
18   <tr>
19     <td>4</td>
20     <td>5</td>
21     <td>6</td>
22   </tr>
23   <tr>
24     <td>7</td>
25     <td>8</td>
26     <td>9</td>
27   </tr>
28 </table>
```



The screenshot shows a Firefox browser window with the title "Lập trình mạng". The address bar displays "file:///C:/". The bookmarks bar includes "Most Visited", "News", "Tech", and "Bookmarks". The main content area displays a 3x3 table with the following numbers:

1	2	3
4	5	6
7	8	9

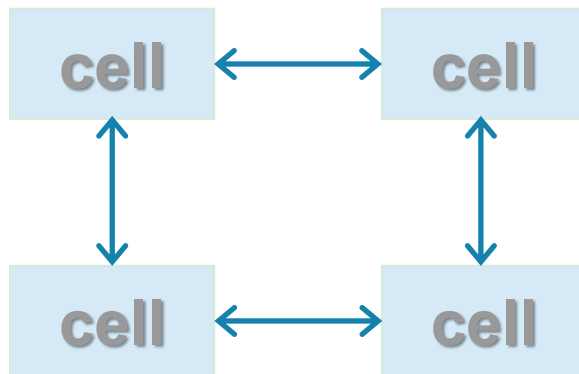
Bảng biểu -Thuộc tính của các thẻ

Thẻ **<table>**

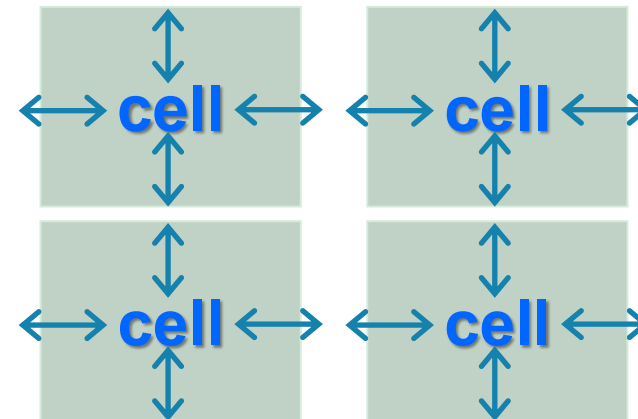
- **border**="số": kích thước đường viền. Đặt bằng 0 (mặc định): không có đường viền.
- **width**="rộng", **height**="cao": độ rộng và độ cao của bảng. Có thể đặt theo 2 cách:
 - n: (n là số) Quy định độ rộng, cao là n pixels
 - n%: Quy định độ rộng, cao là n% độ rộng, cao của đối tượng chứa bảng.
- **bgcolor**="màu": màu nền của bảng
- **background**="địa_chỉ_ảnh": Địa chỉ của file ảnh làm nền cho bảng. Nên sử dụng đường dẫn tương đối nếu có thể.

Bảng biểu -Thuộc tính của các thẻ

- **cellspacing**=“số”: Khoảng cách giữa 2 ô liên tiếp
- **cellpadding**=“số”: Khoảng cách từ border ô đến nội dung ô



◆ cellspacing



◆ Cellpadding

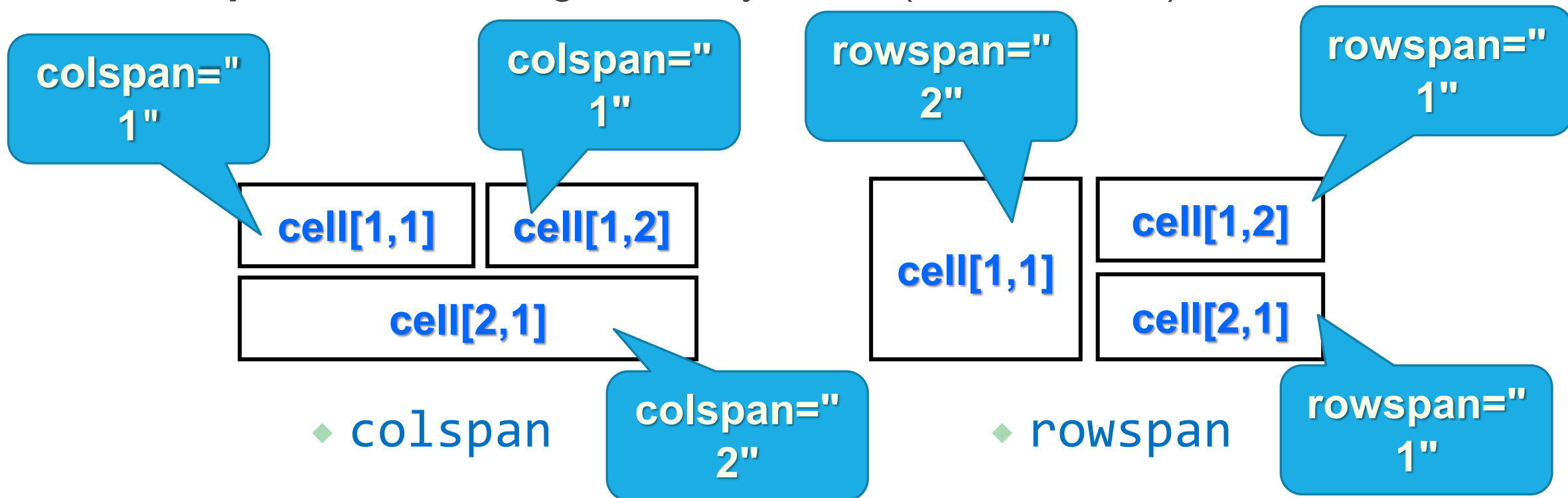
Bảng biểu -Thuộc tính của các thẻ (tt)

Thẻ **<td>**, **<th>**

- **bgcolor**="màu": màu nền của ô
- **background**="địa_chỉ_ảnh": Địa chỉ của file ảnh làm nền cho ô. Nên sử dụng đường dẫn tương đối nếu có thể.
- **width**="rộng", **height**="cao": độ rộng và độ cao của ô. Có thể đặt theo 2 cách:
 - n: (n là số) Quy định độ rộng, cao là n pixels
 - n%: Quy định độ rộng, cao là n% độ rộng, cao của bảng.
- **align**="căn_lề": cách căn chỉnh dữ liệu trong ô theo chiều ngang: **left**, **right**, **center**, **justify**.
- **valign**="căn_lề_đứng": cách căn chỉnh dữ liệu trong ô theo chiều đứng: **top**, **middle**, **bottom**.
- **nowrap**: nếu có sẽ làm cho dữ liệu trong ô không tự xuống dòng

Column and Row Span

- **colspan**="số": số cột mà ô này chiếm (mặc định là 1)
- **rowspan**="số": số dòng mà ô này chiếm (mặc định là 1)



Column and Row Span – Ví dụ

```
<table cellpadding="0">
  <tr class="1"><td>Cell[1,1]</td>
    <td colspan="2">Cell[2,1]</td></tr>
  <tr class="2"><td>Cell[1,2]</td>
    <td rowspan="2">Cell[2,2]</td>
    <td>Cell[3,2]</td></tr>
  <tr class="3"><td>Cell[1,3]</td>
    <td>Cell[2,3]</td></tr>
</table>
```

Cell[1,1]	Cell[2,1]	
Cell[1,2]	Cell[2,2]	Cell[3,2]
Cell[1,3]		Cell[2,3]

Bảng biểu – Phụ đề

Thẻ **<caption>** : chứa phụ đề của bảng

```
1 <table>
2   <caption>Monthly savings</caption>
3   <tr>
4     <th>Month</th>
5     <th>Savings</th>
6   </tr>
7   <tr>
8     <td>January</td>
9     <td>$100</td>
10  </tr>
11 </table>
```

Month	Savings
January	\$100
February	\$50

Bảng biểu – nhóm cột

Thẻ **<colgroup>** nhóm các cột thành từng nhóm giúp định dạng các cột cùng lúc.

- Thẻ colgroup đứng sau <caption>, trước <thead>, <tbody>, <tfoot>, <tr>

```
1 <table>
2   <colgroup>
3     <col span="2" style="background-color:red">
4     <col style="background-color:yellow">
5   </colgroup>
6   <tr>
7     <th>ISBN</th>
8     <th>Title</th>
9     <th>Price</th>
10  </tr>
11  <tr>
12    <td>3476896</td>
13    <td>My first HTML</td>
14    <td>$53</td>
15  </tr>
16 </table>
```

ISBN	Title	Price
3476896	My first HTML	\$53
5869207	My first CSS	\$49

Bảng biểu – thead, tbody, tfoot

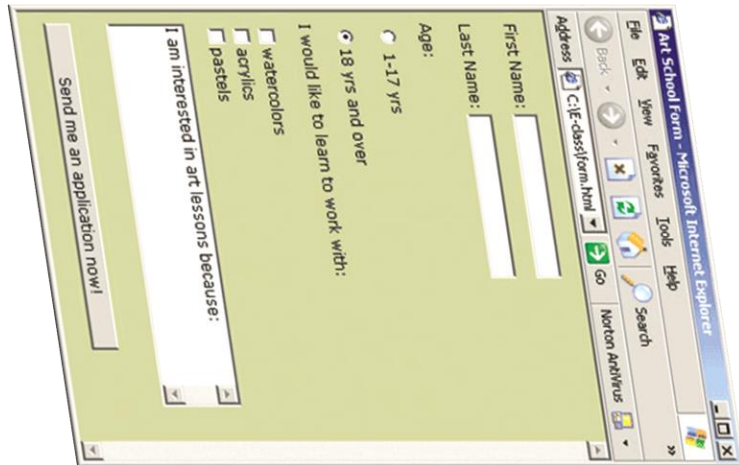
Thẻ <thead> : định nghĩa phần đầu của bảng

Thẻ <tbody>: định nghĩa phần thân của bảng

Thẻ <tfoot>: định nghĩa phần cuối của bảng

Month	Savings
January	\$100
February	\$80
Sum	\$180

```
1 <table>
2   <thead>
3     <tr>
4       <th>Month</th>
5       <th>Savings</th>
6     </tr>
7   </thead>
8   <tfoot>
9     <tr>
10      <td>Sum</td>
11      <td>$180</td>
12    </tr>
13  </tfoot>
14  <tbody>
15    <tr>
16      <td>January</td>
17      <td>$100</td>
18    </tr>
19    <tr>
20      <td>February</td>
21      <td>$80</td>
22    </tr>
23  </tbody>
24 </table>
```



Form nhập dữ liệu

Các đối tượng nhập dữ liệu

Cho phép người sử dụng nhập dữ liệu trên trang web. Dữ liệu này có thể được gửi về server để xử lý.

Người sử dụng nhập dữ liệu thông qua các điều khiển (controls). Có nhiều loại control:

1. Form
2. Oneline Textbox
3. Checkbox
4. Radio Button
5. Button
6. Combo box (drop-down menu)
7. Listbox
8. Hộp nhập văn bản nhiều dòng (TextArea)
9. ...



The image shows a screenshot of a web browser window titled "Registration Form - Mozilla Fi...". The browser's address bar shows "http://". The form itself is titled "Registration Form" and contains the following fields and controls:

- User name:
- Password:
- Gender: ☐ Male ☐ Female
- Click to accept our terms: ☐
- Buttons: Register, Reset

The status bar at the bottom of the browser window shows "Done" and a green checkmark icon.

Các đối tượng nhập dữ liệu (tt)

Tất cả các điều khiển đều có tên được quy định qua thuộc tính **name**. Tuy nhiên có một số điều khiển thì name không quan trọng (các điều khiển mà sau này không cần lấy dữ liệu)

Các điều khiển từ số 2. đến số 5 được định nghĩa nhờ thẻ **<input>** và thuộc tính **type** sẽ xác định là điều khiển nào sẽ được tạo ra.

Form

Sử dụng để **chứa mọi đối tượng khác**

Không nhìn thấy khi trang web được hiển thị

Quy định một số thuộc tính quan trọng như **method**, **action**.

Thẻ tạo form:

<form>...</form>

```
<form name="myForm" method="post"
action="path/to/some-script.php">
```

Các thuộc tính:

- **name**="tên_form": Không quan trọng lắm
- **action**="địa chỉ nhận dữ liệu": Nên sử dụng đường dẫn tương đối nếu nằm trong cùng 1 web
- **method**="phương thức gửi dữ liệu". Chỉ có 2 giá trị:
 - **GET** (mặc định)
 - **POST**

```
...
</form>
```

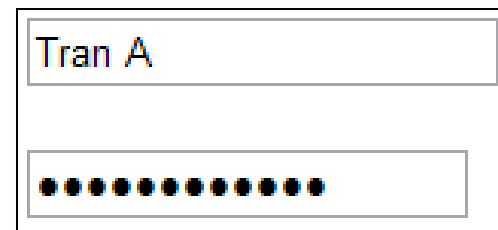
Hộp nhập văn bản 1 dòng (Online Textbox)

Sử dụng để nhập các văn bản ngắn (trên 1 dòng) hoặc mật khẩu

Thẻ: **<input>**

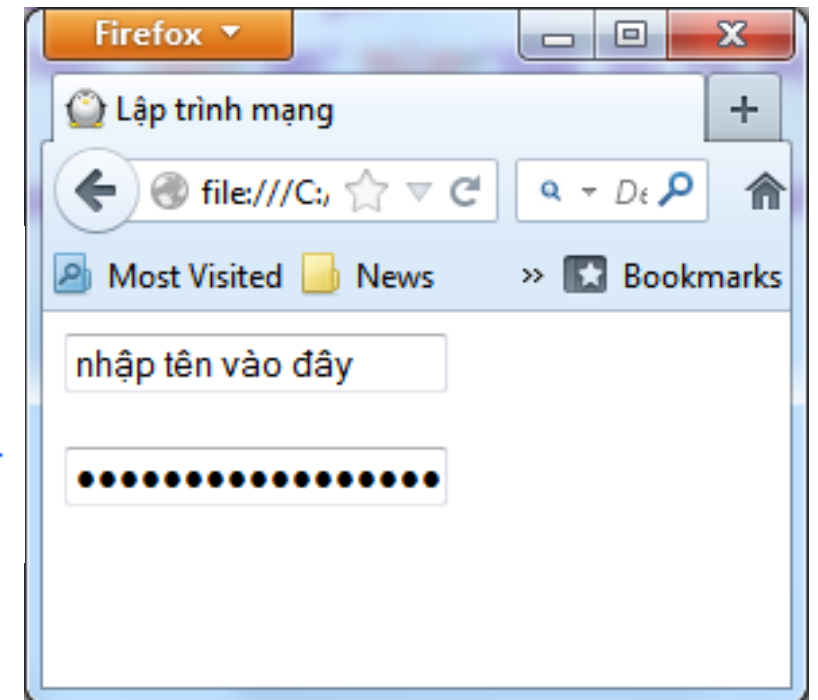
Thuộc tính:

- **name**="tên_đt": quan trọng
- **type**="text": Ô nhập văn bản thường
- **type**="password": ô nhập mật khẩu
- **value**="giá trị mặc định"



Hộp nhập văn bản 1 dòng (Online Textbox)

```
12 <form action="user.php" method="get">
13     <input type="text" name="ten" value="nhập tên vào đây">
14     <br><br>
15     <input type="password" name="matkhau" value="nhập mật khẩu vào đây">
16 </form>
```

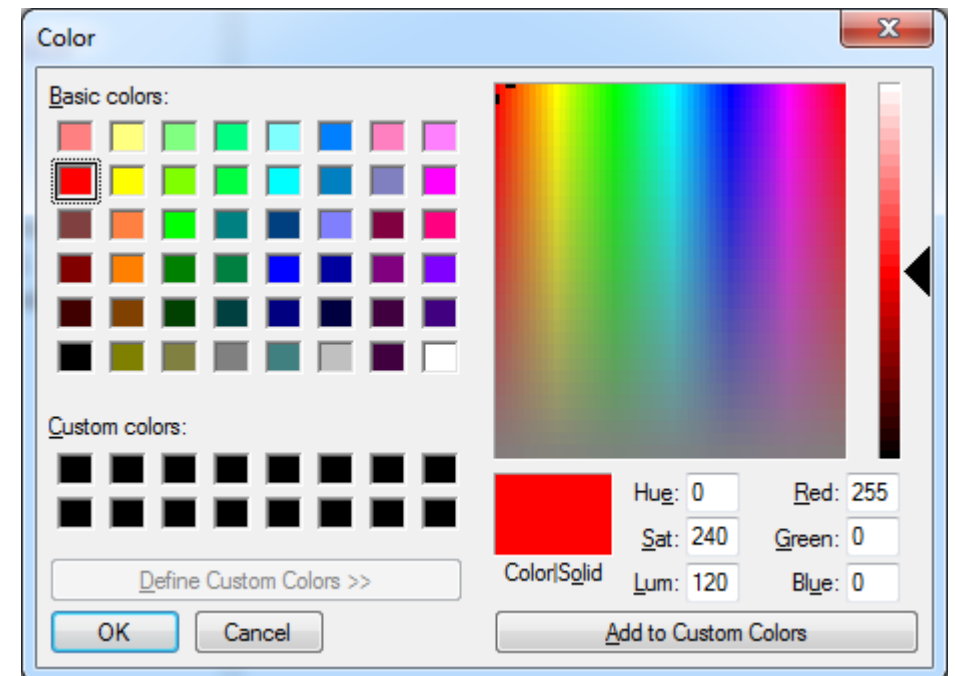


Input Types and attributes

`<input type="color">`

Select your favorite color:

```
<form>  
  Select your favorite color:  
  <input type="color" name="favcolor">  
</form>
```



Input Types and attributes

`<input type="range">`

Points:

```
<form>
  <input type="range" name="points" min="0" max="10">
</form>
```

`<input type="month">`

```
<form>
  Birthday (month and year) :
  <input type="month" name="bdaymonth">
</form>
```

Birthday (month and year):

December, 2016 ▾

Sun	Mon	Tue	Wed	Thu	Fri	Sat
27	28	29	30	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

Input Types and attributes

`<input type="week">`

```
<form>  
  Select a week:  
  <input type="week" name="week_year">  
</form>
```

Select a week:

December, 2016

Week	Sun	Mon	Tue	Wed	Thu	Fri	Sat
48	27	28	29	30	1	2	3
49	4	5	6	7	8	9	10
50	11	12	13	14	15	16	17
51	18	19	20	21	22	23	24
52	25	26	27	28	29	30	31

Input Types and attributes

`<input type="time">`

Select a time:

```
<form>
  Select a time:
  <input type="time" name="usr_time">
</form>
```

`<input type="datetime-local">`

```
<form>
  Birthday (date and time):
  <input type="datetime-local" name="bdaytime">
</form>
```

Birthday (date and time):


December, 2016

Sun	Mon	Tue	Wed	Thu	Fri	Sat
27	28	29	30	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

Input Types and attributes

`<input type="email">`

E-mail:

 Please include an '@' in the email address. 'abc' is missing an '@'.

```
<form>  
  E-mail:  
  <input type="email" name="email">  
</form>
```

Input Types and attributes

Attributes: **Autocomplete**:

- Autocomplete = on(form), off(input) hoặc ngược lại
- Tự động điền dữ liệu dựa trên lịch sử

```
<form autocomplete="on">  
  First name:<input type="text" name="fname"><br>  
  Last name: <input type="text" name="lname"><br>  
  E-mail: <input type="email" name="email" autocomplete="off"><br>  
  <input type="submit">  
</form>
```

Input Types and attributes

Attributes: **autofocus**:

- Trong thẻ <input> **First name:** `<input type="text" name="fname" autofocus>`
- Tự động focus khi tải trang

Attributes: **form**:

- ```
<form id="form1">
 First name: <input type="text" name="fname">

 <input type="submit" value="Submit">
</form>
```
- Trong thẻ <input> **Last name:** `<input type="text" name="lname" form="form1">`
  - Quy định thẻ thuộc form nào mặc dù nằm ngoài form

# Input Types and attributes

---

Attributes: **min** and **max**

- Dùng cho input types: number, range, date, datetime-local, month, week

Enter a date after 2000-01-01:

```
<input type="date" name="bday" min="2000-01-02">
```

Quantity (between 1 and 5):

```
<input type="number" name="quantity" min="1" max="5">
```

# Input Types and attributes

---

## Attributes: **placeholder**

- Dùng cho input types: text, search, url, tel, email, and password
- Sử dụng hiển thị giá trị mẫu

First name
Last name
Submit

```
<input type="text" name="fname" placeholder="First name">
<input type="text" name="lname" placeholder="Last name">
```

## Attributes: **Required**

- Dùng cho input types: text, search, url, tel, email, password, date pickers, number, checkbox, radio, and file
- Yêu cầu điền thông tin (không để trống)



# Checkbox

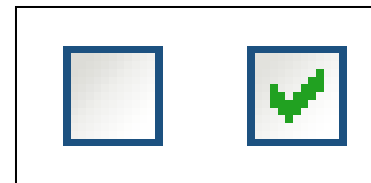
---

Cho phép **chọn nhiều** lựa chọn trong một nhóm lựa chọn được đưa ra bằng cách đánh dấu (“tích”).

Thẻ: **<input>**: mỗi ô nhập cần 1 thẻ

Thuộc tính:

- **name**=“tên\_đt”: quan trọng
- **type**=“checkbox”
- **value**=“giá trị”: đây là giá trị chương trình sẽ nhận được nếu NSD chọn ô này.
- **checked**: nếu có thì nút này mặc định được chọn



# Checkbox

---

```
<form action="dangky.html" method="get">
```

Sở thích: <br>

```
<input name="sothich[]" checked="checked" type="checkbox" value="du lich"> Du lịch
```

```


```

```
<input name="sothich[]" type="checkbox" value="doc sach"> Đọc sách
```

```


```

```
<input name="sothich[]" type="checkbox" value="mua sam"> Mua sắm
```

```
</form>
```

Sở thích:

☒ Du lịch

☐ Đọc sách

☐ Mua sắm

# Option Button (Radio Button)

---

Cho phép **chọn một** lựa chọn trong một nhóm lựa chọn được đưa ra.

Trên 1 form có thể có nhiều nhóm lựa chọn kiểu này.

Thẻ: **<input>**: Mỗi ô cần 1 thẻ

Thuộc tính:

- **name**="tên\_đt": quan trọng. Các đối tượng cùng tên thì thuộc cùng nhóm.
- **type**="radio"
- **value**="giá trị": đây là giá trị chương trình sẽ nhận được nếu NSD chọn ô này.
- **checked**: nếu có thì nút này mặc định được chọn



# Option Button (Radio Button)

---

```
<form action="dangky.html" method="get">
```

```
Giới tính:

```

```
 <input name="gioitinh" checked type="radio" value="nam"> Nam
```

```


```

```
 <input name="gioitinh" type="radio" value="nu"> Nu
```

```
</form>
```

Giới tính:

☒ Nam

☐ Nu

# Form Input Controls - button

---

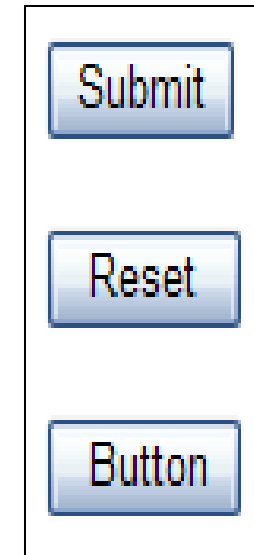
3 loại nút:

- Submit: gửi dữ liệu lên server
- Reset: Reset form – khôi phục form trở lại trạng thái ban đầu
- Normal: thường sử dụng trong javascript

Sử dụng thẻ **<input>** hoặc **<button>**

Thuộc tính

- **Type="submit"**: Submit button
- **Type="reset"**: Reset button
- **Type = "button"**: Ordinary button
- Value = "tiêu đề của nút"



# Combo Box (Drop-down menu)

Bao gồm một danh sách có nhiều phần tử. Tại một thời điểm chỉ có 1 phần tử được chọn

NSD có thể chọn 1 phần tử trong danh sách xổ xuống bằng cách kích vào mũi tên bên phải hộp danh sách.

Thẻ tạo hộp danh sách:

**<select>**Danh sách phần tử**</select>**

Thuộc tính:

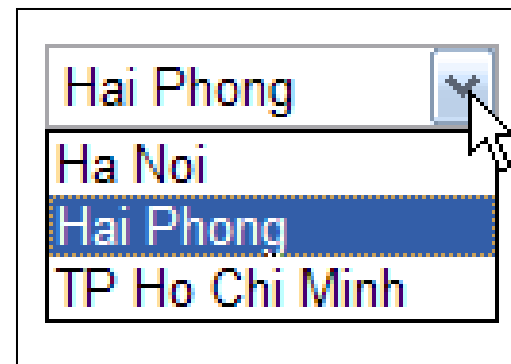
- **name**="tên\_ĐT": quan trọng

Thẻ tạo 1 phần tử trong danh sách:

**<option>**Tiêu đề phần tử**</option>**

Thuộc tính:

- **value**="giá trị": giá trị chương trình nhận được nếu phần tử được chọn
- **selected**: nếu có thì phần tử này mặc định được chọn



# Combo Box (Drop-down menu)

---

```
<form action="dangky.html" method="get">
```

Quê quán:

```
<select name="quequan">
```

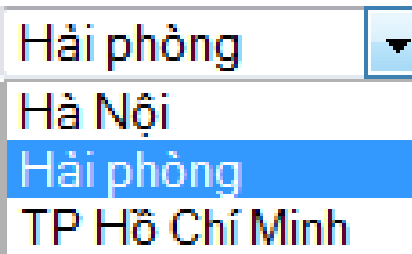
```
 <option value="hanoi">Hà Nội</option>
```

```
 <option selected value="haiphong">Hải phòng</option>
```

```
 <option value="hochiminh">TP Hồ Chí Minh</option>
```

```
</select>
```


```
</form>
```

Quê quán: 

# Option group

Thẻ **<optgroup>** được dùng để nhóm các lựa chọn thành nhóm.

```
1 <select>
2 <optgroup label="Swedish Cars">
3 <option value="volvo">Volvo</option>
4 <option value="saab">Saab</option>
5 </optgroup>
6 <optgroup label="German Cars">
7 <option value="mercedes">Mercedes</option>
8 <option value="audi">Audi</option>
9 </optgroup>
10 </select>
```



Swedish Cars
Volvo
Saab
German Cars
Mercedes
Audi



# Listbox

---

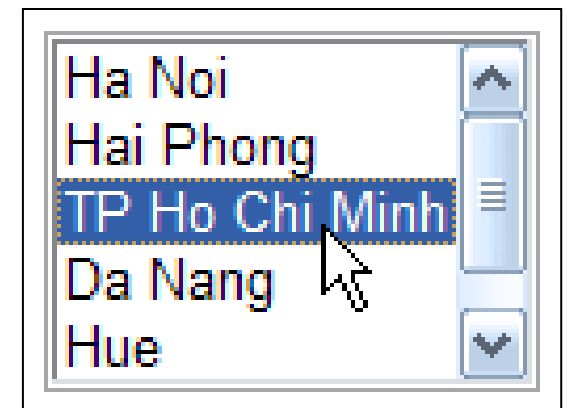
Tương tự như Combo box, tuy nhiên có thể nhìn thấy nhiều phần tử cùng lúc, có thể lựa chọn nhiều phần tử

Thẻ: **<select>...</select>**

Thuộc tính: tương tự của combo tuy nhiên có 2 thuộc tính khác:

- **size**="số dòng"
- **multiple**: cho phép lựa chọn nhiều phần tử cùng lúc

Thẻ **<option>...</option>** tương tự của combobox



# Hộp nhập văn bản nhiều dòng (TextArea)

Cho phép nhập văn bản dài trên nhiều dòng.

Thẻ:

**<textarea>**

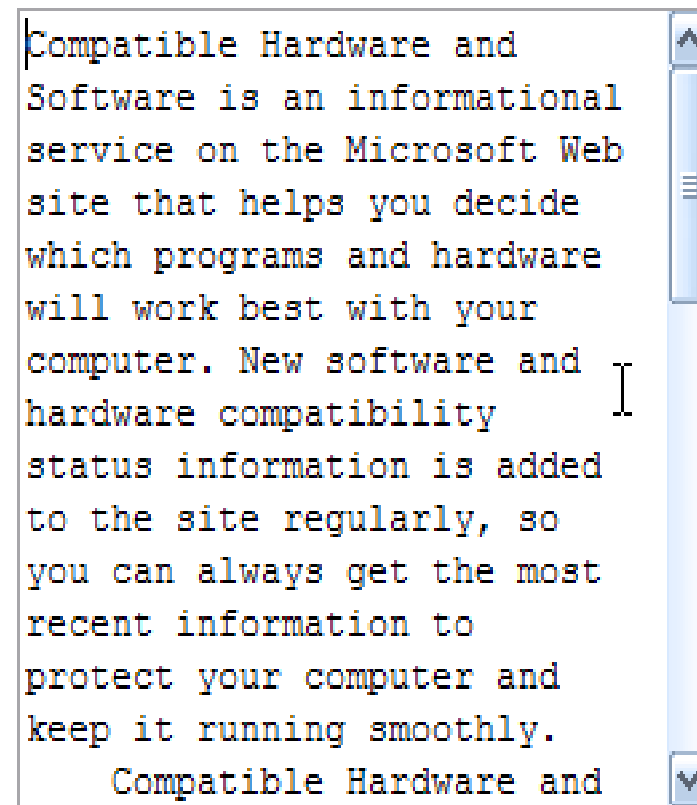
Nội dung mặc định

**</textarea>**

Thuộc tính:

- **name**="tên\_ĐT": quan trọng
- **rows**="số dòng"
- **cols**="số cột"

*rows* tính theo số dòng văn bản, *cols* tính theo số ký tự chuẩn trên dòng.



# HTML - <fieldset>, <legend>

Thẻ **<fieldset>** nhóm các đối tượng trong form thành nhóm

Thẻ **<legend>** phụ đề chung cho nhóm đối tượng

```
1 <form>
2 <fieldset>
3 <legend>Personalia:</legend>
4 Name: <input type="text">

5 Email: <input type="text">

6 Date of birth: <input type="text">
7 </fieldset>
8 </form>
```

Personalia: \_\_\_\_\_

Name:

Email:

Date of birth:

# HTML <label>

Thẻ **<label>** chứa tiêu đề cho các đối tượng input

- Khi nhấn vào tiêu đề sẽ tự động chọn input tương ứng

Male ☐  
Female ☐

Submit

```
1 <form action="demo_form.asp">
2 <label for="male">Male</label>
3 <input type="radio" name="sex" id="male" value="male">

4 <label for="female">Female</label>
5 <input type="radio" name="sex" id="female" value="female">

6 <input type="submit" value="Submit">
7 </form>
```

# TabIndex

---

Thuộc tính **tabindex** cho phép thiết lập thứ tự duyệt các đối tượng trong form khi nhấn phím TAB

- `tabindex="0"` (zero) – thứ tự mặc định
- Nếu  $X > Y$  phần tử có `tabindex="X"` được duyệt (focus) trước phần tử có `tabindex="Y"`
- Phần tử có `tabindex` là số âm thì bỏ qua

```
<input type="text" tabindex="10" />
```

Academic information

Degree

Student ID

Classes attended

Personal Details

First Name

Last Name

Gender: ☐ Male ☒ Female

Email

TERMS AND CONDITIONS...

# HTML Forms – Example

# Chế độ hiển thị block và inline

---

Mỗi phần tử HTML có chế độ hiển thị mặc định

## Block:

Address	Aside	Article	Div
Header	footer	figure	ul, li
Section	table	hr	....

## Inline:

a	b,i,u	span	small
Img	map	textarea	...

# Iframes

---

Sử dụng để hiển thị 1 trang web trong một trang web khác

**<iframe** src="*URL*" title = "description">**</iframe>**

- **url**: địa chỉ trang web con muốn hiển thị
- **Width**: độ rộng
- **Height**: độ cao
- **Frameborder**: đường viền

```
<iframe src="http://www.abc.com">
 <p>Your browser does not support iframes.</p>
</iframe>
```



# Đa phương tiện

---

Âm thanh nền: **<bgsound>**

- Thuộc tính:
  - **src**="địa chỉ file âm thanh"
  - **loop**="n": số lần lặp. -1: mặc định: mãi mãi.

# Flash

---

Thẻ **<object>** chèn file Flash

```
<object width="400" height="400" data="helloworld.swf"></object>
```

- **Width**: độ rộng
- **Height**: độ cao
- **Data**: địa chỉ file Flash

# HTML5 - Graphic

---

## <Canvas>

- Vẽ đồ họa trên trang web bằng cách sử dụng JavaScript
- Mặc định: vùng hình chữ nhật, không có đường viền và không có nội dung

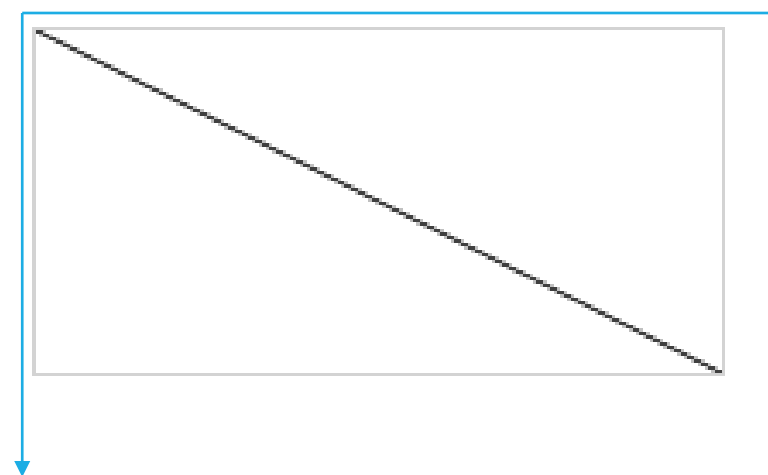
```
<canvas id="myCanvas" width="200" height="100"></canvas>
```

- Các phương thức vẽ:
  - Paths
  - Boxes
  - Circles
  - Text
  - Adding images.

# HTML5 - Canvas

## Vẽ đường – Line

- `lineWidth`: độ rộng
- `strokeStyle`: màu đường kẻ
- `lineCap`: butt | round | square



```
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");
ctx.moveTo(0,0);
ctx.lineTo(200,100);
ctx.stroke();
```

# HTML5 - Canvas

---

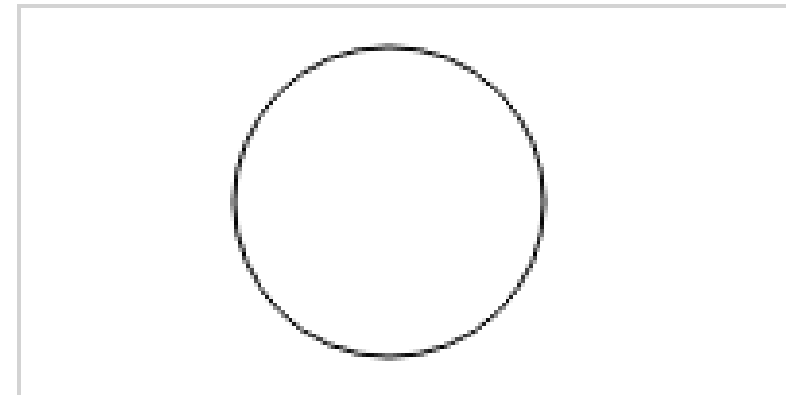
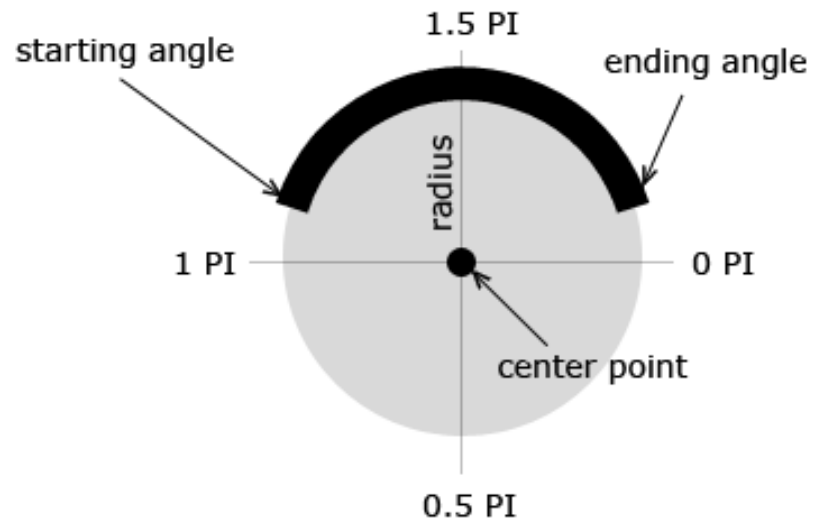
Vẽ hình chữ nhật: `rect(x,y, width, height)`



```
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");
ctx.beginPath();
ctx.rect(20, 20, 150, 100);
ctx.stroke();
```

# HTML5 - Canvas

Vẽ hình tròn: `arc(x,y, r, startangle, endAngle, counterClockwise)`



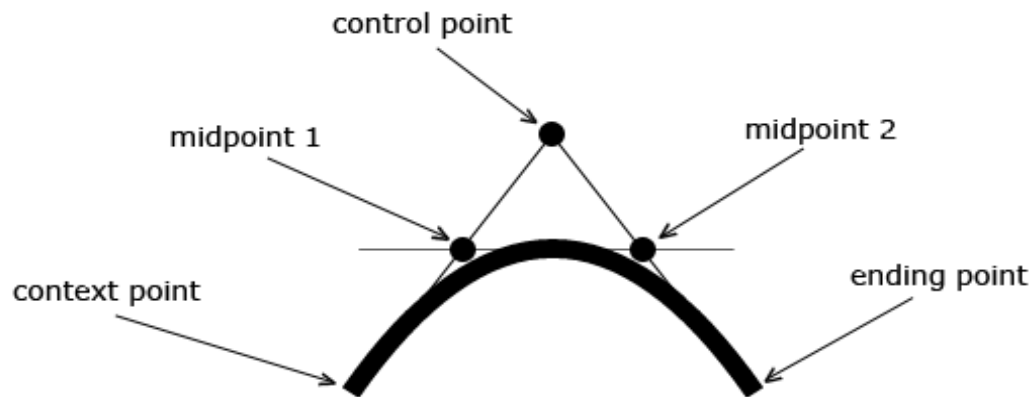
```
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");
ctx.beginPath();
ctx.arc(95,50,40,0,2*Math.PI);
ctx.stroke();
```

# HTML5 - Canvas

---

Vẽ đường cong quadratic:

`quadraticCurveTo(controlpoint, endpoint)`



```
context.beginPath();
context.moveTo(188, 150);
context.quadraticCurveTo(288, 0, 388, 150);
context.lineWidth = 10;

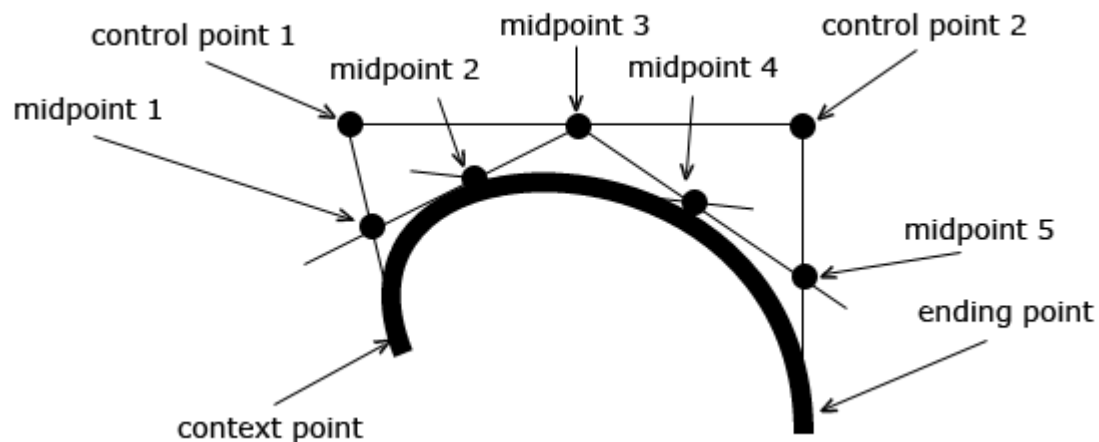
// line color
context.strokeStyle = 'black';
context.stroke();
```

# HTML5 - Canvas

---

## Vẽ đường cong Benzier

- `bezierCurveTo(controlpoint1, controlpoint2, endpoint)`



```
ctx.beginPath();
ctx.moveTo(20, 20);
ctx.bezierCurveTo(20, 100, 200, 100, 200, 20);
ctx.stroke();
```



# HTML5 - Canvas

---

Vẽ text



Hello World



Hello World

```
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");
ctx.font = "30px Arial";
ctx.fillText("Hello World",10,50);
ctx.strokeText("Hello World",10,50);
```

# HTML5 - Canvas

---

Vẽ màu:

- `fillStyle`: màu đối tượng
- `fillStroke`: màu viền đối tượng
- `fillRect(x1,y1, width, height)`: màu hình chữ nhật

# HTML5 - Canvas

---

## Linear Gradient

`createLinearGradient(x,y, x1,y1)`



```
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");

// Create gradient
var grd = ctx.createLinearGradient(0,0,200,0);
grd.addColorStop(0,"red");
grd.addColorStop(1,"white");

// Fill with gradient
ctx.fillStyle = grd;
ctx.fillRect(10,10,150,80);
```

# HTML5 - Canvas

---

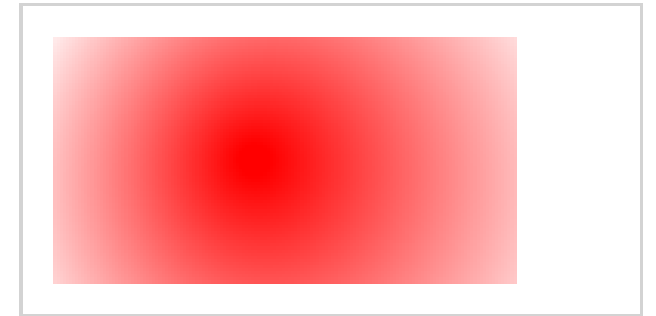
## Circular Gradient

`createRadialGradient(x,y,r,x1,y1,r1)`

```
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");

// Create gradient
var grd = ctx.createRadialGradient(75,50,5,90,60,100);
grd.addColorStop(0,"red");
grd.addColorStop(1,"white");

// Fill with gradient
ctx.fillStyle = grd;
ctx.fillRect(10,10,150,80);
```



# HTML5 - Canvas

---

Adding image:

drawImage(img, x,y, width, height)

```
<p>Image to use:</p>

```

```
<p>Canvas to fill:</p>
<canvas id="myCanvas" width="250" height="300"
style="border:1px solid #d3d3d3;">
```

```
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");
var img = document.getElementById("scream");
ctx.drawImage(img,10,10);
```

# HTML5 - Graphic

---

## <SVG> - scalable Vector Graphics

- Sử dụng để vẽ đồ họa trên web
- Được khuyến nghị sử dụng bởi W3C
- Phương thức
  - Vẽ paths
  - Boxes
  - Circles
  - Text
  - Graphic images

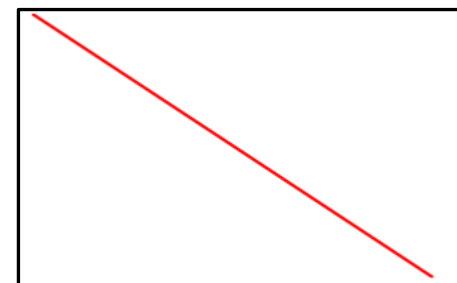
# HTML5 - SVG

---

## <line>:

- Vẽ đường thẳng

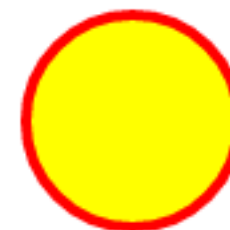
```
<svg height="210" width="500">
 <line x1="0" y1="0" x2="200" y2="200"
 style="stroke:rgb(255,0,0);stroke-width:2" />
</svg>
```



## <circle>

- Vẽ hình tròn

```
<svg width="100" height="100">
 <circle cx="50" cy="50" r="40"
 stroke="red" stroke-width="4" fill="yellow" />
 not support for svg.
</svg>
```

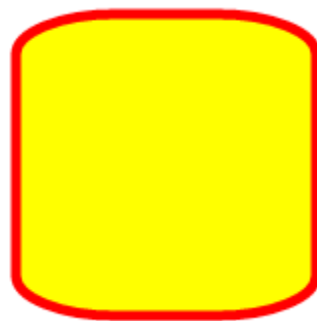


# HTML5 - SVG

---

## <rect>

- Vẽ hình chữ nhật



```
<svg width="200" height="180">
 <rect x="50" y="20" rx="50" ry="20" width="150" height="150"
 style="fill:yellow;stroke:red;stroke-width:5;opacity:0.5" />
 not support for svg.
</svg>
```



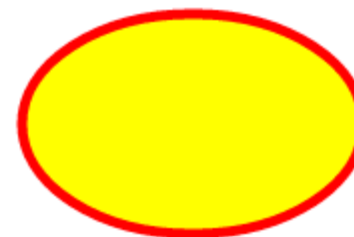
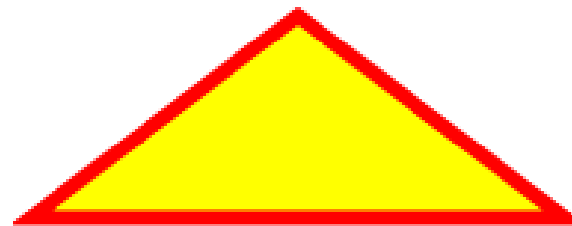
# HTML5 - SVG

---

## <polygon>

- Vẽ hình đa giác

```
<svg width="300" height="200">
 <polygon points="100,10 190,78 10,78 "
 style="fill:yellow;stroke:red;stroke-width:5;fill-rule:evenodd;" />
 not support for svg
</svg>
```



## <ellipse>

- Vẽ hình ellip

```
<svg height="130" width="500">
 <ellipse cx="100" cy="70" rx="85" ry="55"
 fill="yellow" stroke="red" stroke-width="5" />
</svg>
```

# HTML 5- canvas - svg

---

Canvas	SVG
<ul style="list-style-type: none"><li>•Resolution dependent</li><li>•No support for event handlers</li><li>•Poor text rendering capabilities</li><li>•You can save the resulting image as .png or .jpg</li><li>•Well suited for graphic-intensive games</li></ul>	<ul style="list-style-type: none"><li>•Resolution independent</li><li>•Support for event handlers</li><li>•Best suited for applications with large rendering areas (Google Maps)</li><li>•Slow rendering if complex (anything that uses the DOM a lot will be slow)</li><li>•Not suited for game applications</li></ul>

# HTML5 - Media

---

Tag	Description
<audio>	Defines sound or music content
<embed>	Defines containers for external applications (like plug-ins)
<source>	Defines sources for <video> and <audio>
<track>	Defines tracks for <video> and <audio>
<video>	Defines video or movie content

# HTML5 - <video>

---

## <video>

- Hiện thị video trên trang web
- Attributes:
  - **controls**: hiển thị các nút điều khiển: play, pause, volume
  - **autoplay**: tự động phát
- Width, height: kích thước
- **<source>**: đường dẫn file.

```
<video width="320" height="240" controls>
 <source src="movie.mp4" type="video/mp4">
 <source src="movie.ogv" type="video/ogg">
 Your browser does not support the video tag.
</video>
```

# HTML5 - < audio >

---

## <audio>

- Hiển thị một audio trên trang



- Attributes:

- **controls**: hiển thị các nút điều khiển: play, pause, volume
- **autoplay**: tự động phát

- **<source>**:

```
<audio controls>
 <source src="audio.mp3" type="audio/mp3">
 <source src="audio.ogg" type="audio/ogg">
 Your browser does not support the audio tag.
</audio>
```

# HTML5 - <object>/<embed>

---

## <object>/<embed>

- Hiển thị một đối tượng được nhúng

```
<object width="400" height="50" data="bookmark.swf"></object>
```

```
<object width="100%" height="500px" data="snippet.html"></object>
```

```
<object data="audi.jpeg"></object>
```