



# CÔNG NGHỆ WEB

## CÔNG NGHỆ WEB VÀ HTML





- Mục đích của môn học
- Các khái niệm cơ bản trong môi trường web
- Cấu trúc của một trang HTML
- Các thẻ HTML





# MỤC ĐÍCH MÔN HỌC

- Cung cấp cho sinh viên các kiến thức về vai trò của Client và Server trong world wide web,
  - Các nguyên tắc làm việc chung nhất của ngôn ngữ Client và Server,
  - Các protocol cần thiết để làm việc với ứng dụng web.
- Hình thành kỹ năng thiết kế, xây dựng và triển khai các hệ thống website





# MỤC ĐÍCH MÔN HỌC

- Tài liệu tham khảo:

[1] Bài giảng Công nghệ web, Chu Thị Hường

[2] Bài giảng lập trình Web, Chu Thị Hường

[3] PROFESSIONAL ASP.NET MVC 5, Jon Galloway, Brad Wilson, K. Scott Allen, David Matson

[4] Professional LINQ, Scott Klein

[5] Mobile ASP.NET MVC 5, Eric Sowell

[6] Bootstrap for ASP.NET MVC, Pieter van der Westhuizen

[7] Internet





# MỤC ĐÍCH MÔN HỌC

- Công cụ lập trình:
  - + Bộ Visual Studio;
  - + Hệ quản trị SQL Server





# CÁC KHÁI NIỆM CƠ BẢN

- **World Wide Web:** Dịch vụ World wide web được xây dựng theo mô hình Client/Server:
  - + Một máy phục vụ cho việc lưu trữ các tài liệu Hypertext gọi là Web Server.
  - + Một máy tính cùng với phần mềm có khả năng hiểu được các tài liệu Hypertext và giao tiếp được với Web Server gọi là Web Browser hay web client.
  - + Khi một người sử dụng có một yêu cầu một tài liệu nào đó bằng cách gửi đến Web Server một URL thì Web Server sẽ phục vụ tài liệu đó và Web Browser sẽ hiển thị nó lên màn hình với khuôn dạng thích hợp.





# CÁC KHÁI NIỆM CƠ BẢN

- URLs (Uniform Resource Locators): Là một con trỏ được dùng với mục đích đơn giản là xác định vị trí tài nguyên của môi trường Internet. URL có cấu trúc chuỗi có định dạng như sau:

`http: // <host> [: <port>] [ <path> [? <query>]]`

+ Ví dụ:

`http://www.mta.edu/default.aspx?tabid=234`

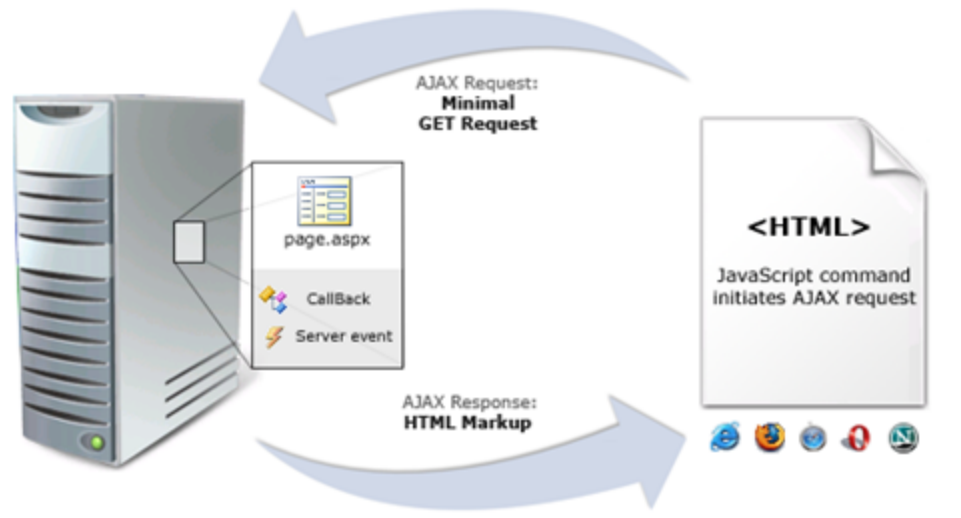




# CÁC KHÁI NIỆM CƠ BẢN

- HTTP (Hypertext Transfer Protocol): Web Browser và Web Server giao tiếp với nhau thông qua một giao thức được gọi là HTTP. Sự kết nối Http qua 4 giai đoạn:

- + Tạo kết nối
- + Thực hiện yêu cầu
- + Phản hồi
- + Kết thúc kết nối

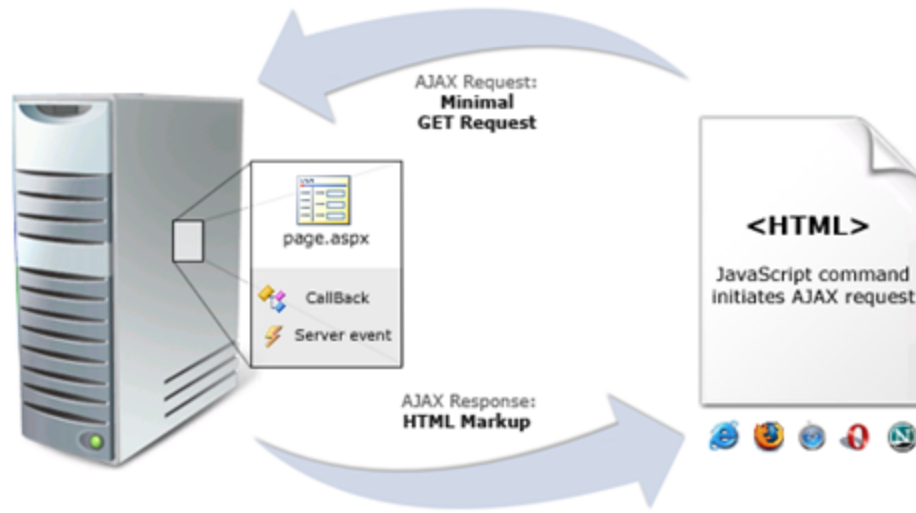






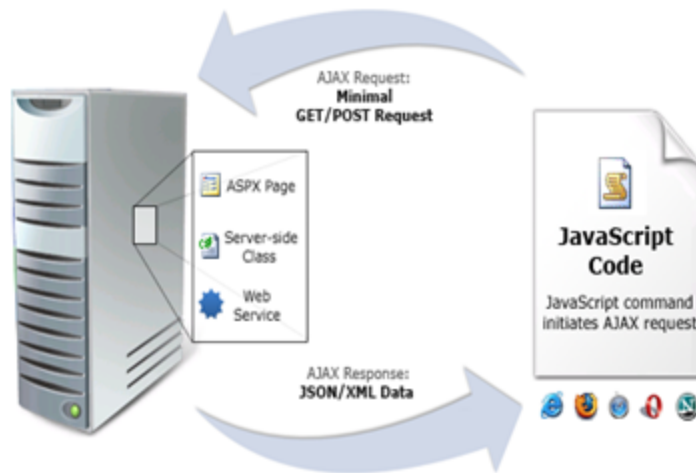
# CÁC KHÁI NIỆM CƠ BẢN

- HTTP (Hypertext Transfer Protocol):
  - + Server side: Khi máy trạm gửi yêu cầu lên máy chủ. Trên máy chủ sẽ xử lý các công việc và thực hiện các thao tác với cơ sở dữ liệu sau đó sẽ gửi thông tin về dưới dạng ngôn ngữ HTML để trình duyệt hiển thị.



# CÁC KHÁI NIỆM CƠ BẢN

- HTTP (Hypertext Transfer Protocol):
  - + Client Side: Khi client gửi Request. Server xử lý các nghiệp vụ và thao tác với hệ quản trị CSDL. Sau đó sẽ trả thông tin về Client. Trên Client, với các thư viện JQuery, javascript ... sẽ thực hiện việc hiển thị dữ liệu trên trình duyệt.





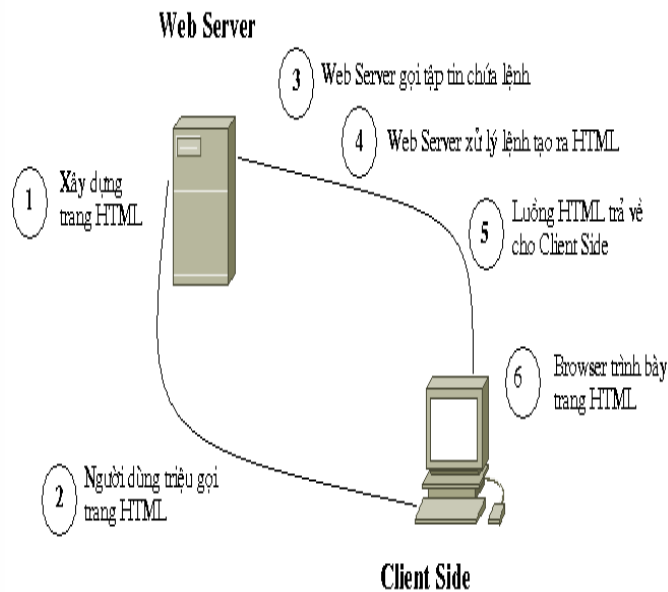
# CÁC KHÁI NIỆM CƠ BẢN

- Web Server: Là một phần mềm được sử dụng ở máy phục vụ và luôn "lắng nghe" yêu cầu của người sử dụng từ một cổng truyền thông nào đó.
  - + Ví dụ: ASP.NET, PHP, JSP,
- Web Browser (trình duyệt Web): là một phần mềm được dùng ở phía người sử dụng,
  - + Khởi tạo các con trỏ địa chỉ URL để gửi tới Web Server.
  - + Diễn dịch các tài liệu có nội dung dưới dạng HTML và hiển thị.

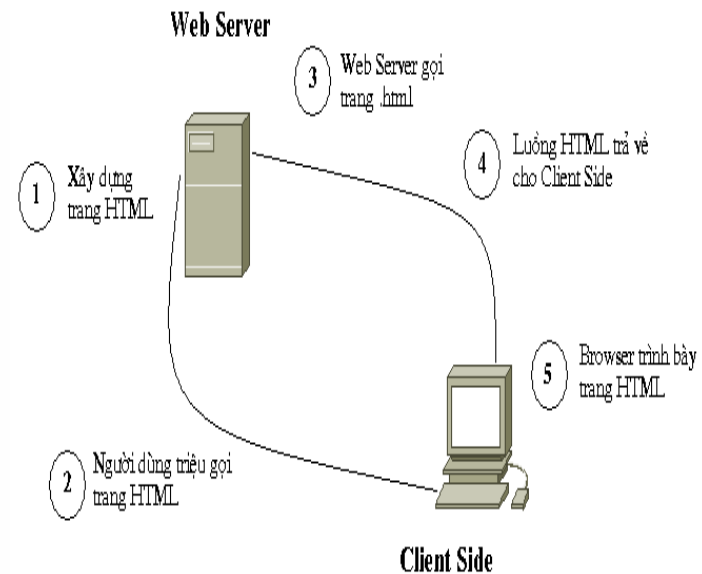


# CÁC KHÁI NIỆM CƠ BẢN

## ■ Web động và tĩnh:



Quy trình triệu gọi trang Web động



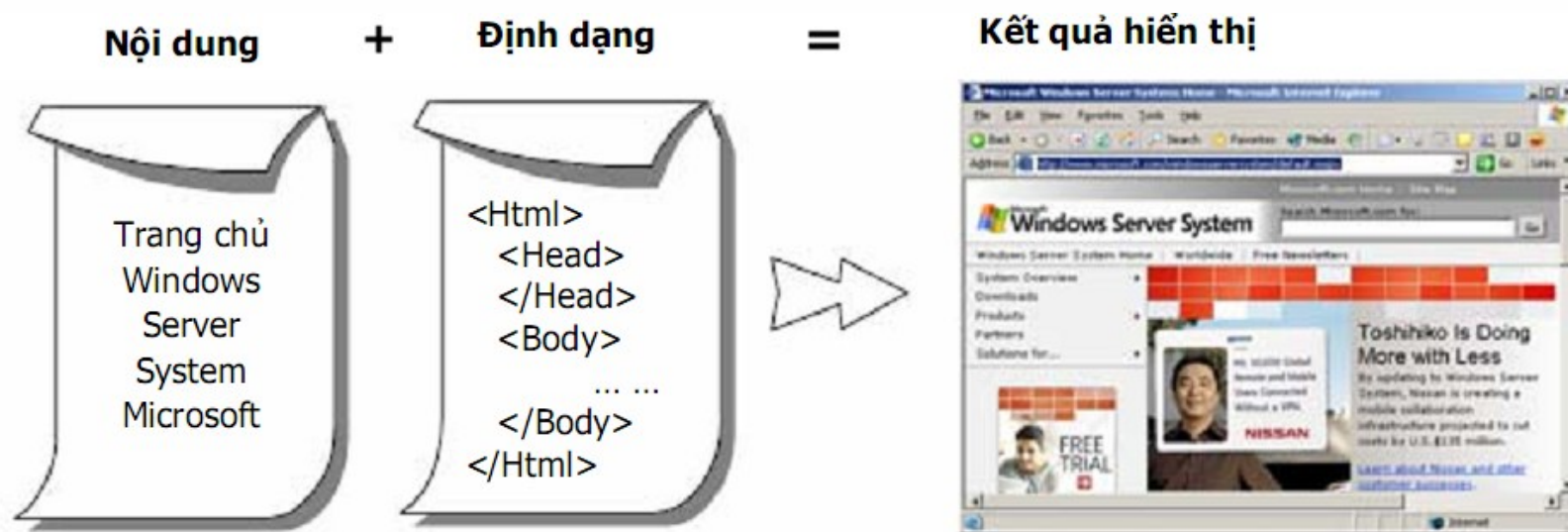
Quy trình triệu gọi trang Web tĩnh





# CẤU TRÚC TRANG HTML

- HTML (HyperText Markup Language) là một ngôn ngữ mô tả tài liệu.





# CẤU TRÚC TRANG HTML

## ■ Cấu trúc trang HTML gồm 3 phần:

- + Phần HTML: Mọi tài liệu HTML đều bắt đầu từ thẻ mở `<html>` và kết thúc bằng thẻ đóng `</html>`

`<html>...</html>`

- + Phần HEAD: Bắt đầu từ thẻ `<head>` và kết thúc bằng thẻ `</head>`. Tất cả các thông tin chứa trong thẻ này đều không xuất hiện trên trang Web, thường chứa các thẻ `<title>` `</title>`, `<meta>`, `<link>`, `<script>`

- Thẻ tiêu đề: `<title>... </title>`

- + Phần BODY: Bắt đầu bằng thẻ `<body>` và kết thúc bằng thẻ `</body>`. Các thông tin trong thẻ `<body>` được xuất hiện trong trang.







# CẤU TRÚC TRANG HTML

## ■ Cấu trúc trang HTML gồm 3 phần:

```
<html>
  <head>
    <title>
      Welcome to My Web
    </title>
    <meta ...>
    <link ...>
    <script ...> </script>
  </head>
  <body>
    <p> Thân trang </p>
  </body>
</html>
```





# CẤU TRÚC TRANG HTML

## ■ Cấu trúc trang HTML:

### + Chú ý:

- HTML quy định không phân biệt chữ hoa và chữ thường.
- Thẻ HTML có hai loại:
  - ✓ Loại có thẻ mở kèm thẻ đóng,
  - ✓ Loại thứ hai là loại chỉ có thẻ mở không có thẻ đóng.







# CÁC THẺ HTML

## ■ Thẻ đoạn và ngắt:

- + Thẻ đoạn `<p>...</p>`: dùng để đánh dấu sự bắt đầu của một đoạn mới (Thẻ `<p>` có thể không cần thẻ đóng)
- + Thẻ ngắt `<br>`: Bổ sung một ký tự xuống dòng
- ✓ Thuộc tính canh lề: Thuộc tính `align` được sử dụng để canh lề cho các phần tử HTML: văn bản, đối tượng, hình ảnh, đoạn,... Thuộc tính `align` nhận các giá trị:
  - ✓ Left: Canh trái
  - ✓ Right: Canh phải
  - ✓ Center: Canh giữa
  - ✓ Justify: Canh đều hai bên.





# CÁC THẺ HTML

- Thẻ khối: Thẻ `<div>` và `<span>` được sử dụng để nhóm nội dung với nhau:

- + `<div> ... </div>`: Dùng để định nghĩa nội dung mức khối, chia tài liệu thành các thành phần có liên quan với nhau.

`<div> ... </div>`

- + Phần `<span> ... </span>`: Định nghĩa nội dung trong dòng (một khoảng các ký tự)

`<span> ... </span>`





# CÁC THẺ HTML

- Các thẻ định dạng ký tự thường dùng:
  - + `<b> ... </b>`: In đậm,
  - + `<i>...</i>`: In nghiêng,
  - + `<u></u>`: Gạch chân,
  - + `<sup>... </sup>`: Chỉ số trên
  - + `<sub> ... </sub>`: Chỉ số dưới
  - + `<em>... </em>`: Nhấn mạnh văn bản
  - + `<code>...</code>`: S/dụng trong trích dẫn mã chương trình.
  - + `<var>...</var>`: S/dụng trong trích dẫn biến chương trình
  - + `<cite>...</cite>`: S/dụng cho trích dẫn và tham chiếu.





# CÁC THẺ HTML

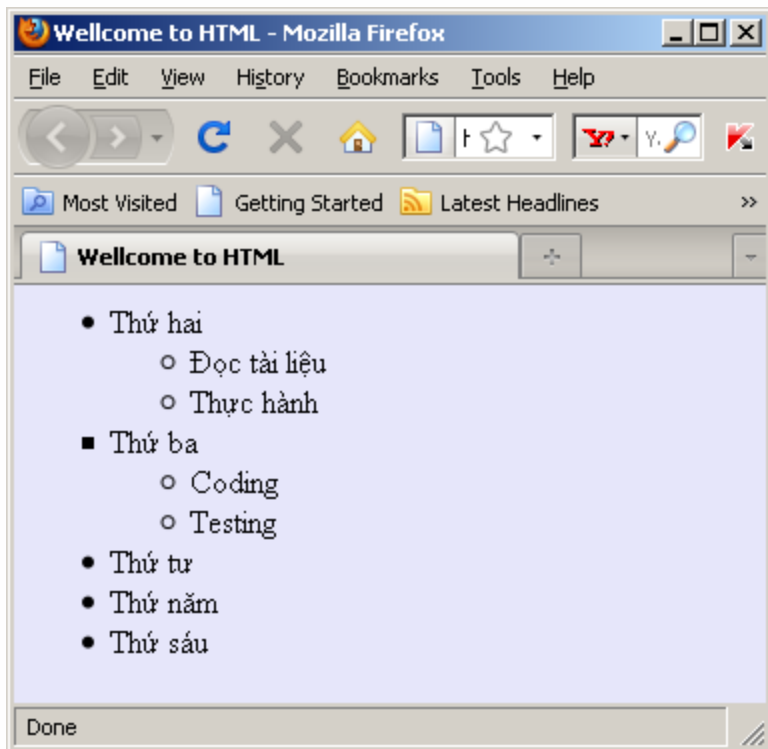
- Danh sách: Danh sách dùng để nhóm dữ liệu có logic với nhau, thường có
  - + Danh sách không có thứ tự
  - + Danh sách có thứ tự
  - + Danh sách định nghĩa





# CÁC THẺ HTML

- Danh sách không có thứ tự (bulleted list): Thuộc tính Type dùng để định dạng các bullets. `<ul>...</ul>`, mỗi phần tử trong danh sách được đánh dấu bằng thẻ `<li>...</li>` (thẻ `</li>` không bắt buộc)



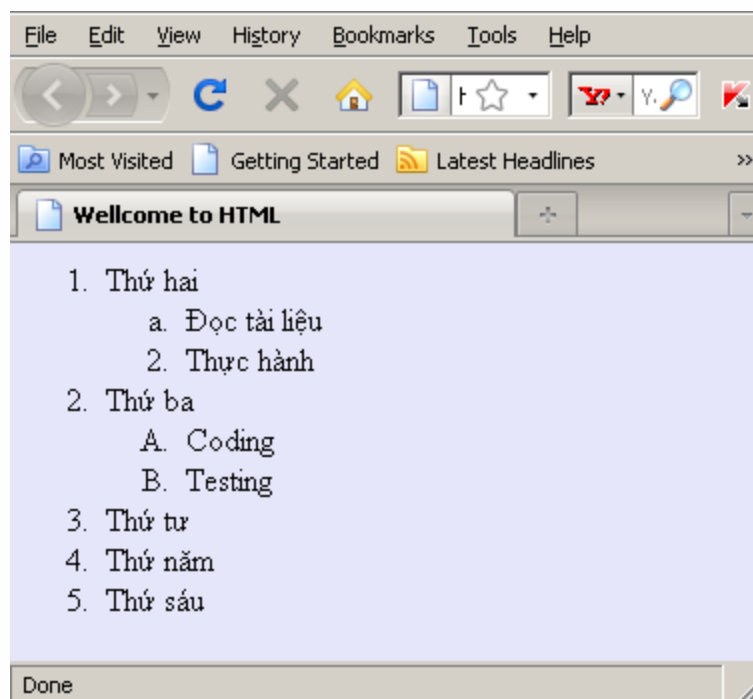
```
<ul>
  <li Type="DISC"> Thứ hai
    <ul >
      <li > Đọc tài liệu
      <li> Thực hành
    </ul>
  <li type="Square"> Thứ ba
    <ul>
      <li type="circle">
        Coding </li>
      <li> Testing</li>
    </ul>
  <li> Thứ tư
  <li> Thứ năm
  <li> Thứ sáu
</ul>
```





# CÁC THẺ HTML

- + Danh sách có thứ tự: Nằm trong cặp `<ol>... </ol>`. Dùng thuộc tính `Type` để định dạng kiểu thứ tự, `Start` để xác định số khởi đầu danh sách.



```
<ol>
  <li> Thứ hai
    <ol >
      <li Type="a"> Đọc tài liệu
      <li> Thực hành
    </ol>
  <li> Thứ ba
    <ol type="A" >
      <li > Coding </li>
      <li> Testing</li>
    </ol>
  <li> Thứ tư
  <li> Thứ năm
  <li> Thứ sáu
</ol>
```





# CÁC THẺ HTML

- Thẻ `<font>...<font>`: Dùng để định dạng font chữ, sử dụng các thuộc tính sau:
  - + Color: Màu chữ (tên màu hoặc mã thập lục phân)
  - + Size: Kích thước font (từ 1 đến 7)
  - + Face: Kiểu font chữ
- Thẻ `<hr>`: Vẽ đường thẳng nằm ngang.
  - + align: Canh lề đường thẳng
  - + width: Độ dài đường thẳng
  - + size: Độ dày đường thẳng
  - + Noshade: Không đổ bóng.





# CÁC THẺ HTML

- ``: Chèn ảnh vào trang web
  - + Src: Chỉ đường dẫn ảnh
  - + Alt: Chuỗi text thay thế khi không có ảnh
  - + Align: Canh lề của ảnh với văn bản xung quanh
  - + Height: Chiều cao của ảnh
  - + Width: Chiều rộng của ảnh







# CÁC THẺ HTML

- Thẻ `marquee`: cho phép khai báo một chuỗi chuyển động theo nhiều hướng khác nhau trên trang Web. Có 4 chiều di chuyển chuỗi tùy thuộc vào tham số hướng là UP, DOWN, BACK, RIGHT.

```
<MARQUEE DIRECTION=RIGHT SCROLLDELAY=2 SCROLLAMOUNT=1>  
  <font face="arial" size="1" color=#ff33ff>  
    Ashley- Ana -Chanthaly - Kathleen- Lena</font>  
</marquee>
```

```
<MARQUEE DIRECTION=UP SCROLLDELAY=2 SCROLLAMOUNT=1  
STYLE="filter:wave(add=5, phase=1, freq=5,  
strength=20);"><font face="arial" size="1" color=#ff33ff>  
  <BR>Ashley<BR>Ana<BR>Chanthaly<BR>Kathleen<BR>Lena  
  </font>  
</marquee>
```





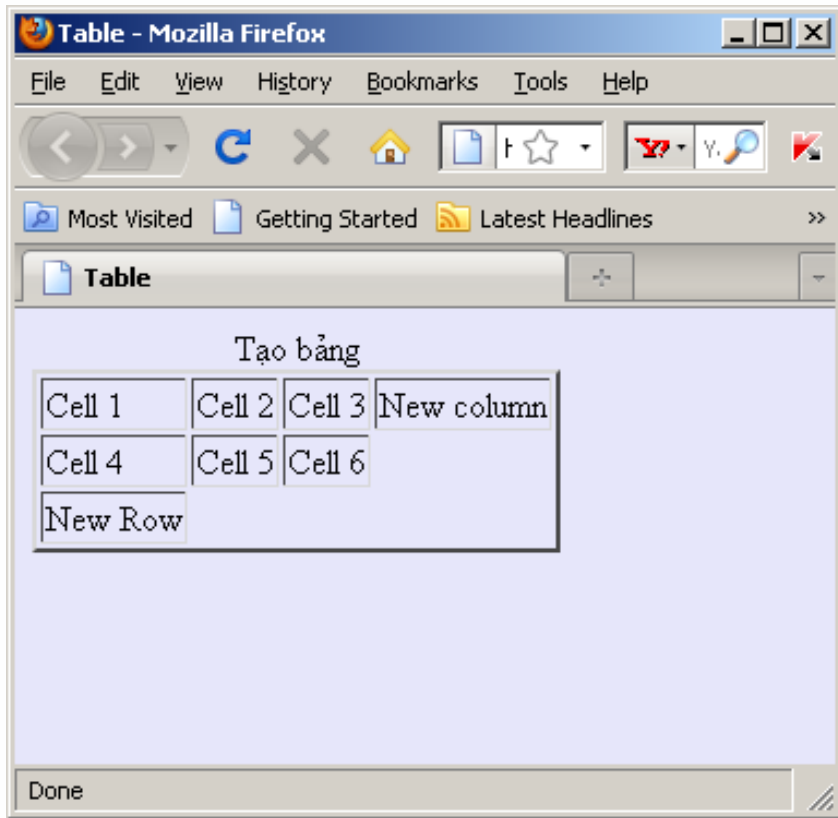
# CÁC THẺ HTML

- Tạo bảng: Dùng thẻ `<table>... </table>`, trong thẻ `<table>` ta định nghĩa các thẻ:
  - + `<td> ...</td>`: Định nghĩa một ô, đây là đơn vị cơ bản của bảng.
  - + `<tr>...</tr>`: Định nghĩa hàng của bảng. Các ô tạo thành hàng, các hàng tạo thành bảng.
  - + `<th>...</th>`: Định nghĩa tiêu đề cho cột
  - + `<Caption>...</Caption>`: Định nghĩa tiêu đề bảng



# CÁC THẺ HTML

## ■ Tạo bảng:



```
<table border="2">
  <caption>Tạo bảng</caption>
  <tr>
    <td>Cell 1</td>
    <td>Cell 2</td>
    <td>Cell 3</td>
    <td>New column</td>
  </tr>
  <tr>
    <td>Cell 4</td>
    <td>Cell 5</td>
    <td>Cell 6</td>
    <td>New Row
  </td>
  </tr>
</table>
```





# CÁC THẺ HTML

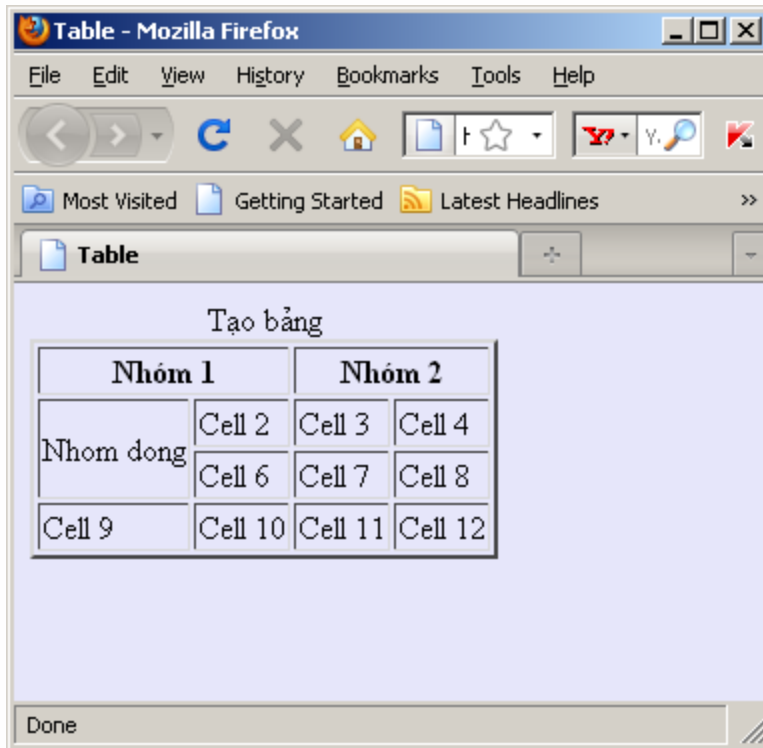
- Tạo bảng: Các thuộc tính định dạng:
  - + border: Định dạng đường viền.
  - + Width: Độ rộng bảng
  - + Trộn ô:
    - Colspan: Kết hợp các cột thành một, thường dùng với thẻ <th>
    - Rowspan: Kết hợp các dòng thành một, thường dùng với thẻ <td>
  - + Định dạng ô:
    - Align: Canh lề ngang
    - Valign: Canh lề dọc
    - Cellspacing: Định nghĩa không gian giữa các ô
    - Cellpadding: Định nghĩa không gian giữa đường viền ô với nội dung văn bản.





# CÁC THẺ HTML

## ■ Tạo bảng:



```
<table border="2">
  <caption>Tạo bảng</caption>
  <th colspan="2">Nhóm 1</th>
  <th colspan="2">Nhóm 2</th>
  <tr>
    <td rowspan="2">Nhóm dòng</td>
    <td>Cell 2</td>
    <td>Cell 3</td>
    <td>Cell 4</td>
  </tr>
  <tr>
    <td>Cell 6</td>
    <td>Cell 7</td>
    <td>Cell 8</td>
  </tr>
  <tr>
    <td>Cell 9</td>
    <td>Cell 10</td>
    <td>Cell 11</td>
    <td>Cell 12</td>
  </tr>
</table>
```





# CÁC THẺ HTML

- Thẻ liên kết: Dùng để liên kết đến một tài liệu hay file khác hoặc đến một phần khác trong tài liệu đó:
  - + Một phần khác trong tài liệu.
  - + Một tài liệu khác
  - + Một phần trong tài liệu khác.

```
<a href=protocol://host.domain:port/path  
/filename> HyperText </a>
```





# CÁC THẺ HTML

## ■ Thẻ liên kết:

```
<a href=protocol://host.domain:port/path  
/filename> HyperText </a>
```

- + Host.domain: Là địa chỉ Internet của máy chủ
- + Port: Cổng phục vụ của máy chủ đích
- + Hypertext: Văn bản hoặc hình ảnh mà người dùng cần nhấp vào để kích hoạt liên kết.





# CÁC THẺ HTML

## ■ Thẻ liên kết:

- + Liên kết đến tài liệu khác

- + Sử dụng hình ảnh:

```
<a href="doc2.html">  
      
</a>
```

- + Sử dụng nhãn:

```
<a href="doc2.html">  
    Next  
</a>
```

- Chú ý: Khi người dùng “nhảy” đến một tài liệu khác nên có sự cung cấp điều hướng đến trang chủ hoặc đến trang khác.







# CÁC THẺ HTML

## ■ Thẻ liên kết:

- + Liên kết đến tài liệu khác

- + Sử dụng hình ảnh:

- Đường dẫn tuyệt đối chỉ ra đầy đủ từ thư mục gốc đến file.

`C:\mydirectory\HTML\Doc2.html`

- Đường dẫn tương đối chỉ ra vị trí liên quan đến file hiện tại

`<a href="..\Doc3.html">Next</a>`

- Chú ý: Liên kết đến các tài liệu không có liên quan thì nên dùng đường dẫn tuyệt đối. Nếu có các nhóm tài liệu liên quan với nhau thì nên dùng đường dẫn tương đối với các tài liệu trong nhóm và đường dẫn tuyệt đối cho các tài liệu không liên quan trực tiếp đến chủ đề.





# CÁC THẺ HTML

## ■ Thẻ liên kết:

+ Liên kết đến một phần khác trong tài liệu: Dùng thẻ neo `<a>` (anchor) gồm hai phần

■ Đánh dấu vị trí nhày đến:

`<a name="marker">Topic name </a>`

■ Liên kết đến điểm đánh dấu:

`<a href="#marker"> Topic name </a>`

Dấu “#” đặt trước tên siêu liên kết để báo cho trình duyệt biết đây là liên kết đến một điểm trong tài liệu.





# CÁC THẺ HTML

## ■ Thẻ liên kết:

- + Liên kết đến một phần khác trong tài liệu khác:  
Tương tự gồm hai phần :

- Đánh dấu vị trí nhảy đến trên tài liệu khác:

```
<a name="marker">Topic name </a>
```

- Liên kết đến tài liệu tại vị trí điểm đánh dấu:

```
<a href="Doc3.html#marker"> Topic name </a>
```

## + Sử dụng Email:

```
<a href="mailto:sales@sun.com">
```

```
Welcome
```

```
</a>
```





# CÁC THẺ HTML

- Thẻ `<Form>`: Được sử dụng để tạo một vùng trên trang như một mẫu biểu, bao gồm các thuộc tính:
  - + `Action="URL"` : Xác định vị trí của script sẽ xử lý mẫu biểu hoàn chỉnh và đã được gửi đi.
  - + `Method="Post|Get"` : Xác định phương thức dữ liệu được gửi đi.
    - Nếu GET: Trình duyệt tạo ra QueryString trong địa chỉ URL
    - Nếu POST: Dữ liệu trên biểu mẫu được gửi đến kịch bản xử lý như một khối dữ liệu.





# CÁC THẺ HTML

- Các phần tử nhập dữ liệu: Người dùng có thể nhập dữ liệu thông qua các điều khiển Input.
  - + Các điều khiển này được sử dụng với phần tử `<Form>`.
  - + Cũng có thể sử dụng chúng bên ngoài biểu mẫu để tạo giao diện người dùng.





# CÁC THẺ HTML

- Các phần tử nhập dữ liệu:
  - + Phần tử `<Input>` xác định loại điều khiển trên biểu mẫu, gồm các thuộc tính:
    - `Type`: Xác định loại phần tử. Có thể lựa chọn một trong các thành phần sau: Text, Password, CheckBox, Radio, Submit, Reset, File, Hidden và Button. Mặc định là Text.
    - `Name`: Tên của điều khiển. Phạm vi hoạt động của thuộc tính Name nằm trong phần tử Form.
    - `Value`: Là thuộc tính tùy chọn (trừ Radio). Xác định giá trị khởi tạo của điều khiển.
    - `Size`: Xác định độ rộng ban đầu của điều khiển. Đối với kiểu Text hay Password kích thước tính bằng ký tự, còn lại tính bằng pixels.





# CÁC THẺ HTML

- Các phần tử nhập dữ liệu:
  - + Phần tử `<Input>` xác định loại điều khiển trên biểu mẫu, gồm các thuộc tính:
    - `Maxlength`: Xác định số ký tự lớn nhất có thể nhập vào điều khiển Text và Password.
    - `Check`: Xác định nút có được chọn hay không chọn được sử dụng trong điều khiển Radio và CheckBox.
    - `SRC="URL"`: Chỉ định đường dẫn file ảnh trong điều khiển input dạng Image.







# CÁC THẺ HTML

## ■ Các phần tử nhập dữ liệu:

### + Nút bấm: Dạng nút lệnh

```
<input type="button" value="Back" name="back" onclick="window.history.go(-1);">
```

### + Văn bản: Tạo điều khiển nhập văn bản một dòng, thuộc tính size xác định số ký tự có thể hiển thị trong điều khiển, maxlength xác định số phần tử tối đa có thể nhập vào, value xác định giá trị khởi đầu:

```
<input type="text" name="txtUsername" size=20 maxlength=50>
```







# CÁC THẺ HTML

## ■ Các phần tử nhập dữ liệu:

- + CheckBox: Người dùng có thể lựa chọn một hoặc nhiều CheckBox. Khi một phần tử được chọn thì cặp tên/giá trị được nhận cùng với biểu mẫu.

```
<input type="checkbox" value="P" name="chkHow"
        checked>Newspaper
```

- + Radio: Sử dụng tương tự như checkbox, nhưng radio thường được sử dụng trong tập các giá trị loại trừ nhau. Các radio trong cùng nhóm phải cùng tên:

```
<input type="radio" value="M" name="rdGender"
checked>Male
```

```
<input type="radio" value="F" name="rdGender"
> Female
```





# CÁC THẺ HTML

## ■ Các phần tử nhập dữ liệu:

- + Submit: Khi người dùng nhấn vào nút Submit, biểu mẫu được chuyển đến vị trí xác định trong thuộc tính ACTION. Cặp tên/giá trị của nút Submit được nhận cùng biểu mẫu.

```
<input type="submit" value="Save"
      name="submit">
```

- + Ảnh: Điều khiển này dùng để tạo ra một nút Submit dạng ảnh:

```
<input type="image" src="gogo.gif"
      name="search">
```





# CÁC THẺ HTML

## ■ Các phần tử nhập dữ liệu:

- + Reset: Khi người dùng nhấp vào nút này, các giá trị của tất cả các điều khiển được tái thiết lập về các giá trị ban đầu, được xác định trong các giá trị của thuộc tính:

```
<input type="reset" name="back">
```

- + Hidden: *hidden* là một dạng thẻ *input* dạng *text* nhưng không hiển thị trên trang Web, không cho phép người dùng nhập liệu, thay vào đó ta cần định nghĩa trước một giá trị cụ thể cho thẻ này.

```
<input type="hidden" name="txtFrom"
      value="N">
```





# CÁC THẺ HTML

- Thẻ TextArea: Tạo ra vùng văn bản, gồm các thuộc tính
  - + Cols: Độ rộng của cột, số ký tự
  - + Rows: Số hàng
  - + Value: Giá trị của điều khiển

Trong trường hợp muốn gán giá trị mặc định ta không cần khai báo thêm thuộc tính *value*, thay vào đó giá trị cụ thể sẽ khai báo giữa thẻ mở và thẻ đóng của *textarea*.

```
<textarea name="txtRequire" cols="20"
rows="3">Please enter here ...</textarea>
```





# CÁC THẺ HTML

- Thẻ *Select*: Thẻ *Select* cho phép người dùng chọn phần tử trong tập các phần tử được định nghĩa trước. Thẻ *Select* có hai dạng tương tự như *combo box* và *listbox*.
  - + Option: Thẻ *Select* chứa ít nhất một thẻ *Option*
  - + OptGoup: Nhóm các lựa chọn vào một nhóm
  - + Selected: Là thuộc tính logic cho phép chọn trước một phần tử.
  - + Lable: Xác định văn bản được hiển thị trên lựa chọn.





# CÁC THẺ HTML

## ■ Thẻ Select:

- Khai báo thẻ *select* dạng combo box như sau:

```
<select name="city" >
  <option value="" selected>..Select City ..</option>
  <option value="HCM">Ho Chi Minh</option>
  <option value="HAN">Ha Noi</option>
  <option value="HUE">Hue</option>
</select>
```

- Khai báo thẻ *select* dạng *listbox*, ta khai báo bắt đầu từ *select* với tên là *multiple* như sau:

```
<select name="industry" multiple >
  <option value="" selected>..Select Industry..</option>
  <option value="AUT">Auto</option>
  <option value="MED">Medical</option>
  <option value="ENG">Engineering</option>
</select>
```





# CÁC THẺ HTML

- Thẻ Lable: Được dùng để gán nhãn cho các điều khiển. Để gán nhãn cần xác định thuộc tính ID của điều khiển được gán nhãn.

```
<label for="txtUsername"> Tên đăng nhập</label>
<input type="text" name="txtUsername" size=20
      maxlenght=50>
<LABEL for="txtFullname">Họ tên</LABEL>
<input type="text" name="txtFullname" value=""
      size=20 maxlenght=50>
```







# CÁC THẺ HTML

- Thẻ Frame: Mục đích chính của Frame là nhằm chia cửa sổ của trình duyệt ra là hai hay nhiều vùng nhỏ hơn, để khi đó có thể tải các tài liệu HTML khác nhau vào trong các vùng.

Để tạo các Frame sử dụng thẻ `<frameset>` `</frameset>`. Thẻ này có hai thuộc tính cơ bản là ROWS và COLS dùng để mô tả số hàng và số cột chia thành frame.

```
<frameset border="False" frameboder="0" rows="80,*">  
  <frame name="frame1" src="frame1.htm" scrolling="no"  
    noresize>  
  <frame name="frame2" src="frame2.htm" scrolling="auto"  
    noresize>  
</noframes>
```





# CÁC THẺ HTML

## ■ Thẻ Frame:

```
<html>
<head>
<title> Tao Frame </title>
</head>
<frameset border="False" frameboder="0" rows="80,*">
  <frame name="frame1" src="frame1.htm" scrolling="no" noresize>
    <frameset border="False" frameboder="0" cols="80,*">
      <frame name="frame2" src="frame2.htm" scrolling="auto" noresize>
      <frame name="frame3" src="vidul_11.html" scrolling="auto"
noresize>
    </frameset>
  </frameset>
</frameset>
<noframes>
  <body>
  </body>
</noframes>
</html>
```





# THẢO LUẬN – CÂU HỎI



Biên soạn: Chu Thị Hường – Bộ môn HTTT – Khoa CNTT

- Thực hành các ví dụ
- Tìm hiểu mở rộng HTML5





# BÀI TẬP LỚN MÔN HỌC

- Giao bài tập lớn môn học



Biên soạn: Chu Thị Hường – Bộ môn HTTT – Khoa CNTT