



# Lập trình Web

# Mục tiêu môn học

---

- ▶ Cung cấp các kiến thức cơ bản về thiết kế web
  - ▶ **HTML, CSS, JavaScript**
- ▶ Tìm hiểu các thành phần cơ bản của trang web
- ▶ Các bước xây dựng trang web tinh
- ▶ Xây dựng & triển khai trang web tinh.
- ▶ Sử dụng công cụ thiết kế web phổ biến như Dreamweaver, và các công cụ hỗ trợ khác như photoshop, flash...

# Nội dung môn học

---

- ▶ Phần 1: kiến thức cơ bản
  - ▶ HTML (HyperText Markup Language)
  - ▶ CSS (Cascading Style Sheets)
  - ▶ Ngôn ngữ JavaScript.
- ▶ Phần 2: công cụ thiết kế web
  - ▶ Dreamweaver
  - ▶ Photoshop
  - ▶ Flash
  - ▶ Một số công cụ hỗ trợ thiết kế web khác.

# Giới thiệu môn học

---

- ▶ Giảng viên:
  - ▶ ThS. Nguyễn Hà Giang
- ▶ Các môn dạy:
  - ▶ Kỹ thuật lập trình, CTDL & GT, OOP, Lập trình Visual C++ & MFC, Lập trình C# Desktop Application, LT Web Application ASP.NET, Mã nguồn mở (PHP & MySQL), CC&MT PTPM.
- ▶ Hướng nghiên cứu:
  - ▶ Data Mining, Fuzzy Data Mining, Fuzzy Association Rule Mining
  - ▶ Semantic Web Mining,
- ▶ Email: [nguyenha.giang@yahoo.com](mailto:nguyenha.giang@yahoo.com)



## Chương 0

# Các kiến thức cơ bản

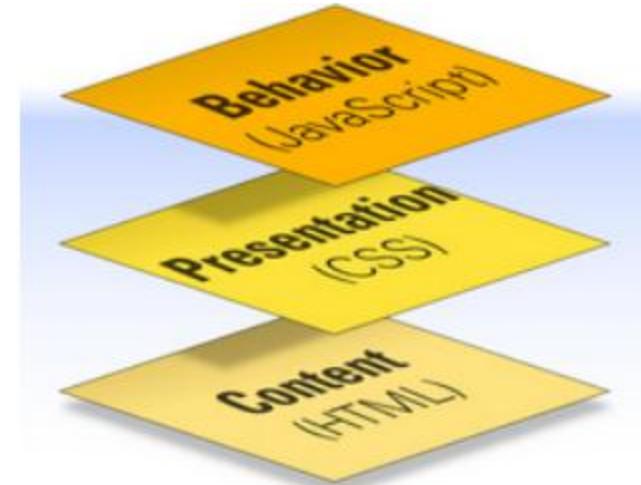
# Mục tiêu

---

- ▶ Giới thiệu **mô hình 3 lớp** trong thiết kế web
- ▶ Các khái niệm và thuật ngữ cơ bản về mạng và môi trường web
- ▶ Cấu trúc và cú pháp của ngôn ngữ **HTML**
- ▶ Cú pháp và thuộc tính của **CSS** (chuẩn định dạng cách trình bày của trang web)
- ▶ Tổng quan về ngôn ngữ script, cú pháp, cách xử lý sự kiện và thao tác trên đối tượng HTML của ngôn ngữ **JavaScript**.

# Mô hình 3 lớp trong TK Web

- ▶ Lớp nội dung
  - ▶ Cung cấp **nội dung thông tin** cho người truy cập.
  - ▶ Bao gồm các văn bản, hình ảnh, các liên kết...
  - ▶ Nội dung này sẽ được tổ chức theo cấu trúc của **ngôn ngữ đánh dấu siêu văn bản** như HTML, XHTML...



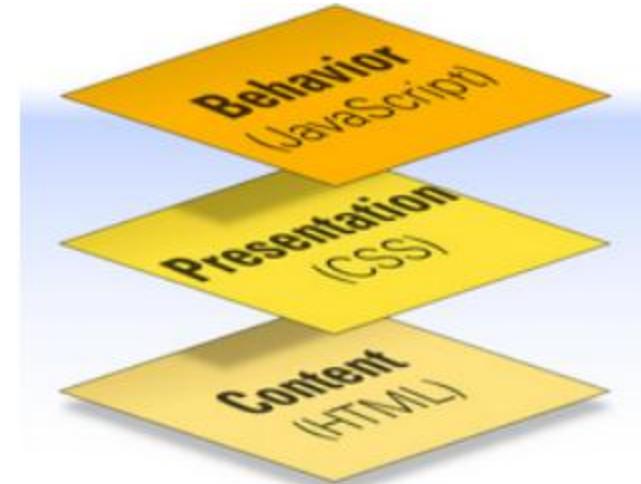
# Mô hình 3 lớp trong TK Web

## ▶ Lớp trình bày

- ▶ Quy định **cách trình bày trang web**. Lớp này định nghĩa các **định dạng** hay **kiểu mẫu** cho các thành phần trong trang web.
- ▶ Các định nghĩa này được lưu trong một file riêng theo cú pháp của **chuẩn định dạng CSS**

## ▶ Lớp hành vi

- ▶ Cho phép thực hiện một số **hành vi/thao tác** trên các thành phần của trang web thông qua **ngôn ngữ script** (JavaScript, VBScript)



# Mô hình 3 lớp trong TK Web

---

- ▶ Một số ưu điểm của mô hình
  - ▶ Share resources
    - ▶ Dùng chung toàn bộ file CSS hay JS cho toàn bộ các trang web trong website.
    - ▶ Khi thay đổi trên tập tin này thì toàn bộ site sẽ được đổi
  - ▶ Faster downloads
    - ▶ Khi user truy cập trang web, các CSS và JS chỉ **tải ở lần truy cập đầu tiên**, trình duyệt sẽ cache lại cho lần sau.
  - ▶ Multi-person teams
    - ▶ Thuận tiện cho việc **phân chia công việc**.
    - ▶ Các thành viên trong nhóm sẽ được phân công theo từng lớp mô hình. Do đó công việc ở các lớp có thể làm đồng thời.

# Các khái niệm & thuật ngữ cơ bản

---

- ▶ **WWW** (World Wide Web): mạng toàn cầu các máy tính sử dụng Internet để trao đổi tài liệu web.
- ▶ **Protocol**: là tập hợp các quy tắc được thống nhất giữa hai máy tính nhằm thực hiện trao đổi dữ liệu được chính xác.
  - ▶ Các giao thức thông dụng: TCP, HTTP, FTP, SMTP...
- ▶ **Web page**: một tài liệu (thường là HTML) được thiết kế để phân phối trên môi trường web.
- ▶ **Web site**: là tập hợp các web page có liên quan đến 1 công ty hay cá nhân

# Các khái niệm & thuật ngữ cơ bản

---

- ▶ **Home page**: là web page có mức cao nhất, gọi là trang chủ của website.
- ▶ **IP Address**: một con số xác định duy nhất cho mỗi máy tính trên Internet
  - ▶ VD: 192.168.10.1
- ▶ **Domain name**: tên xác định website
  - ▶ VD: www.huflit.edu.vn
- ▶ **DNS (Domain name service)**: một chương trình chạy trên server, chuyển tên miền sang IP và ngược lại.

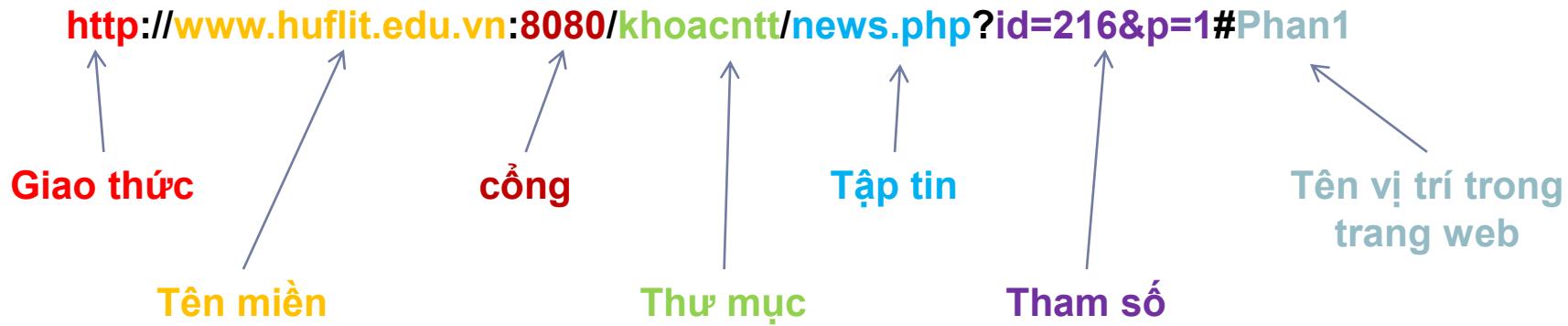
# Các khái niệm & thuật ngữ cơ bản

---

- ▶ **ISP (Internet Service Provider):**
  - ▶ Nhà cung cấp dịch vụ Internet (cung cấp các dịch vụ truy cập Internet và nơi lưu trữ web).
- ▶ **Web host:**
  - ▶ Một web server cung cấp dịch vụ lưu trữ cho web site của các công ty, tổ chức hay cá nhân.
- ▶

# Các khái niệm & thuật ngữ cơ bản

- ▶ **URL** (Uniform Resource Locator): Một địa chỉ web, là một chuẩn để xác định các tài liệu (trang) web trên Internet.



# Các khái niệm & thuật ngữ cơ bản

---

- ▶ **Port:** là con số xác định kênh nhập/xuất được sử dụng bởi một ứng dụng Internet.
- ▶ Một máy server có thể cung cấp nhiều dịch vụ, do đó cần có cơ chế để phân biệt, giúp client khai thác đúng dịch vụ cần thiết.
- ▶ Hai dịch vụ khác nhau phải chạy trên hai cổng khác nhau.
- ▶ VD: web server thường dùng cổng 80, ftp server dùng cổng 21, smtp server dùng cổng 25...

# Các khái niệm & thuật ngữ cơ bản

---

- ▶ **Web client** (Web Browser): là phần mềm dùng để truy cập và hiển thị nội dung trang web.
- ▶ Một số web browser thông dụng như: IE, Firefox, Opera, Safari, Chrome...
- ▶ **Web server**: một máy tính phân phối dịch vụ và thông tin cho máy tính khác.
- ▶ Một số web server thông dụng: IIS, Apache, Tomcat...

# Các khái niệm & thuật ngữ cơ bản

---

- ▶ **Server**: chứa dữ liệu, tài nguyên và dịch vụ cho phép máy khác có thể khai thác và truy cập.
  - ▶ Một máy chủ có thể dùng cho một hay nhiều mục đích.
  - ▶ Tên máy chủ thường được gắn với mục đích sử dụng
    - ▶ VD: Web server, File server, Mail server...
- ▶ **Client**: là máy tính dùng để kết nối và khai thác các tài nguyên trên máy chủ
  - ▶ Việc kết nối client với server và việc khai thác dịch vụ của server tạo nên mô hình Client/Server
  - ▶ Một máy tính vừa có thể là server vừa là client.

# Các khái niệm & thuật ngữ cơ bản

---

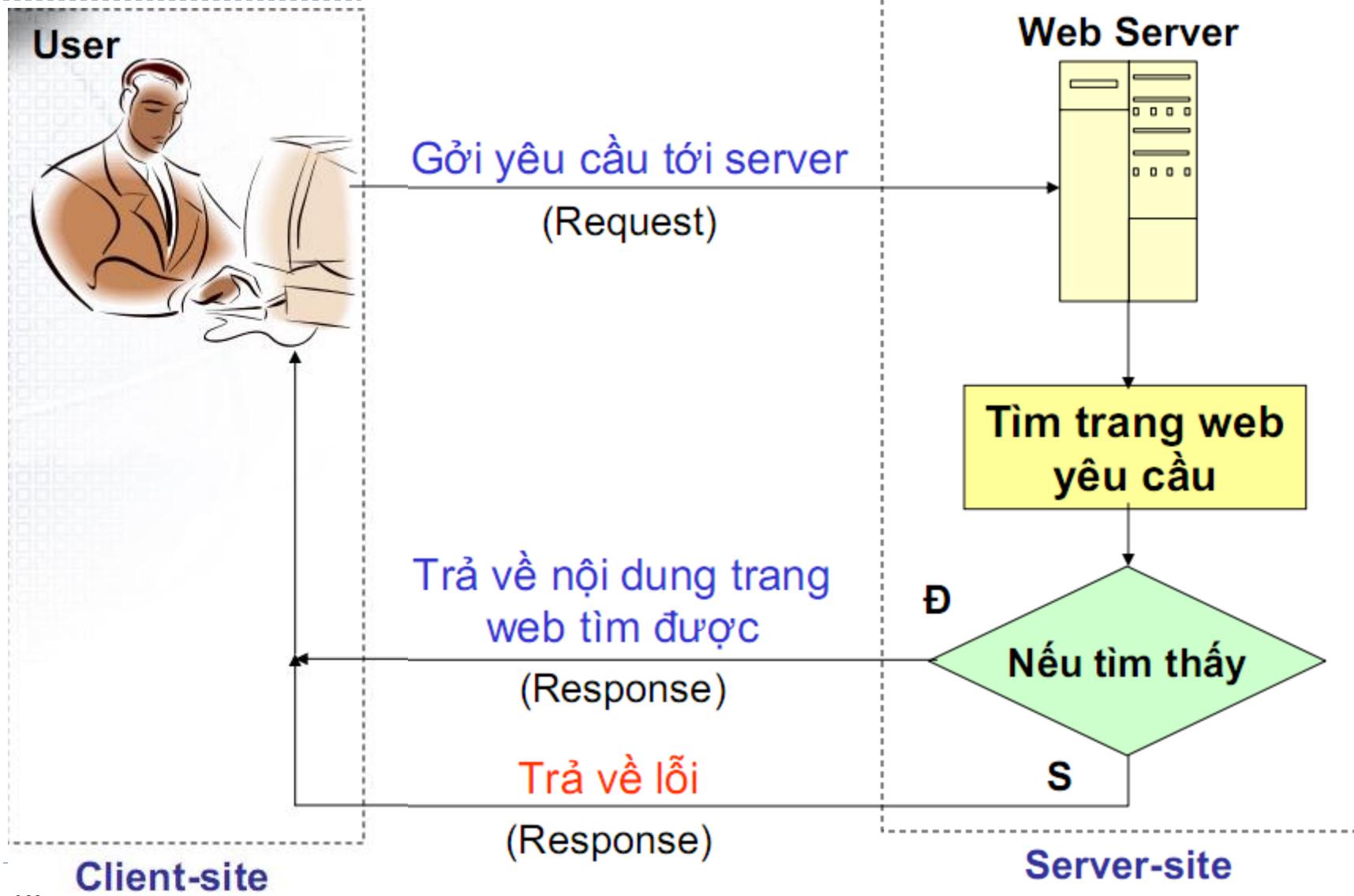
## ► **Trang web tĩnh:**

- ▶ Chứa nội dung cố định (thường là HTML, để cập nhật nội dung phải cập nhật trực tiếp trên HTML).
- ▶ Không cho phép sử dụng tương tác, cập nhật dữ liệu trên trang web.
- ▶ Một trang web chứa các hình ảnh chuyển động cũng có thể là trang web tĩnh!

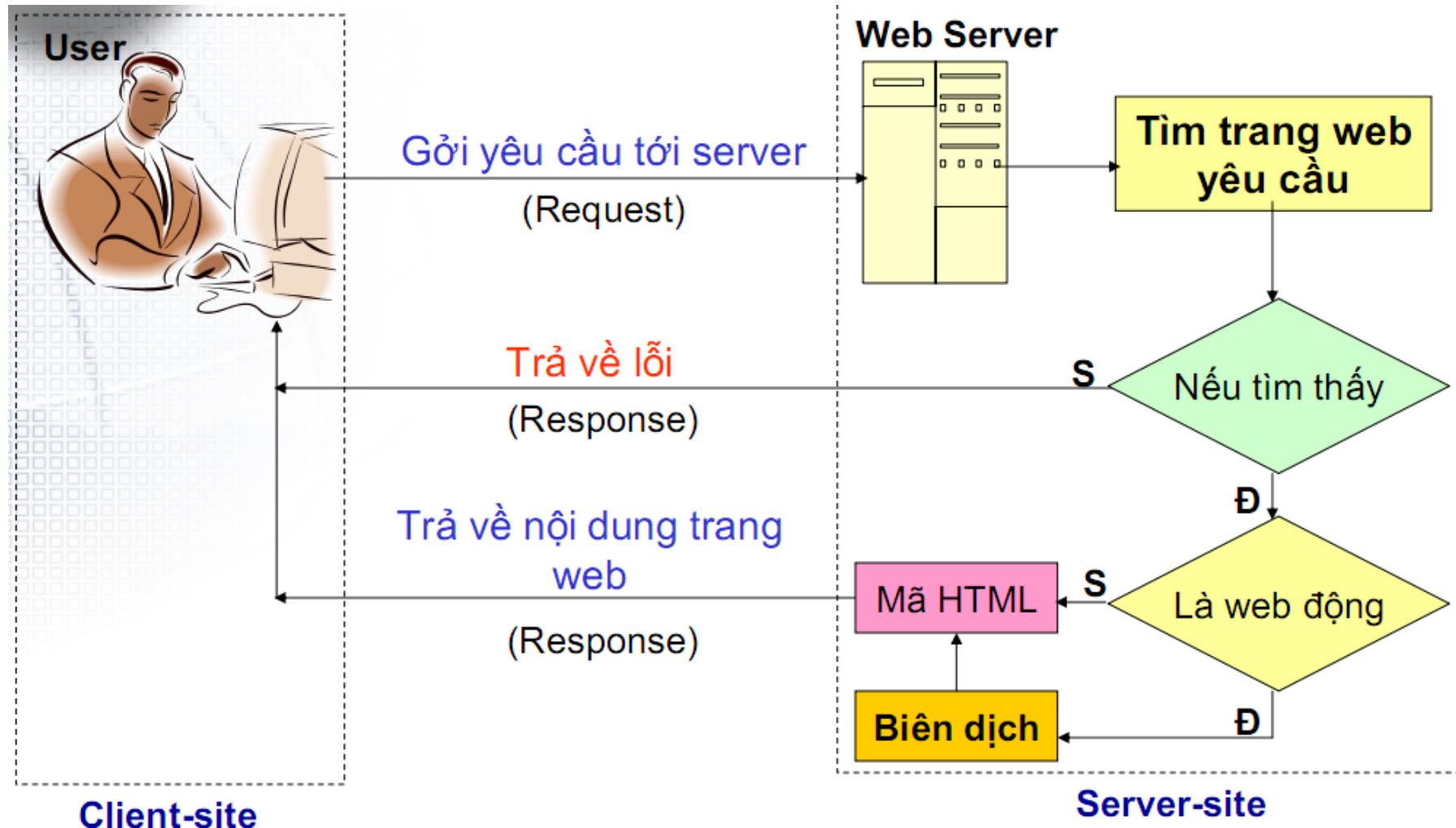
## ► **Trang web động:**

- ▶ Kết hợp HTML và mã lệnh.
- ▶ Mã được thực thi trực tiếp trên server, gửi kết quả là HTML về người sử dụng.
- ▶ Có khả năng tương tác với người sử dụng!

# Xử lý yêu cầu với web tĩnh



# Xử lý yêu cầu với web động





## Chương 1

# Ngôn ngữ HTML

# Nội dung

---

- ▶ Giới thiệu ngôn ngữ HTML
- ▶ Cấu trúc tổng quát trang HTML
- ▶ Các thẻ HTML cơ bản
- ▶ Các ký tự đặc biệt
- ▶ Thiết kế bảng
- ▶ Chia khung
- ▶ Tạo form

# HTML - nền tảng của web

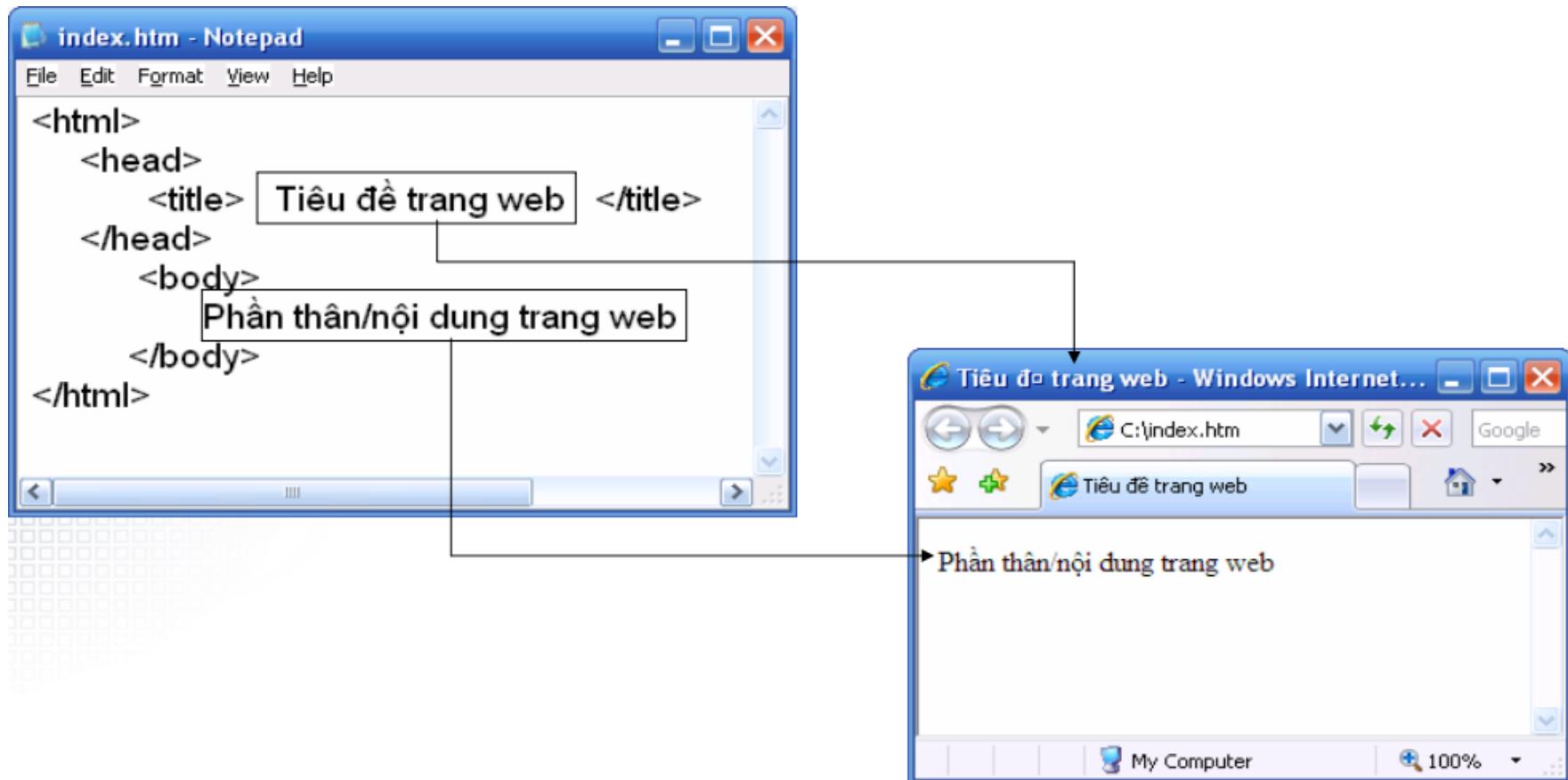
---

- ▶ **HTML**: HyperText Markup Language – ngôn ngữ đánh dấu siêu văn bản.
- ▶ Do **Tim Berner-Lee** phát minh và trở thành ngôn ngữ chuẩn để tạo trang web.
- ▶ HTML dùng các thẻ (tags) để định dạng dữ liệu
- ▶ Tạo khung/bảng cho trang web



Tim Berner-Lee

# Cấu trúc tổng quát trang HTML



Kết quả thể hiện trên web browser

# Các thẻ HTML cơ bản

---

- ▶ Các thành phần cơ bản (**tag, element, property**)
- ▶ Các thẻ HTML cơ bản.
  - ▶ Thẻ định dạng trang: `<body>`
  - ▶ Thẻ định dạng văn bản: `<font>`, `<p>`, `<b>`, `<i>`, `<u>`...
  - ▶ Thẻ tạo siêu liên kết (hyperlink): `<a>`
  - ▶ Thẻ định dạng danh sách: `<ul>`, `<ol>`, `<li>`
  - ▶ Thẻ chèn hình ảnh: `<img>`

# Các thành phần cơ bản

---

- ▶ **Thẻ** (tag): là một tập hợp ký hiệu được định nghĩa trong HTML có ý nghĩa đặc biệt.
  - ▶ thẻ bắt đầu bởi "<" và kết thúc bởi ">"
  - ▶ thẻ mở: <**tên thẻ**>
  - ▶ thẻ đóng: </**tên thẻ**>
- ▶ **Phần tử** (element): có thể bao gồm thẻ mở, thẻ đóng và nội dung bên trong cặp thẻ mở, đóng.
  - ▶ Có hai loại phần tử trong HTML
    - ▶ **Phần tử chứa nội dung**: bao gồm thẻ mở và thẻ đóng
      - VD: <p> Nội dung </p>
    - ▶ **Phần tử rỗng**: bao gồm một thẻ
      - VD: <br> có thể viết là </br> hoặc <br />

## Các thành phần cơ bản

---

- ▶ **Thuộc tính:** mỗi thẻ có thể có một hay nhiều thuộc tính đi kèm. Thuộc tính được nhập ngay trước ngoặc đóng ‘>’ của thẻ mở.
- ▶ Có thể có nhiều thuộc tính trong một thẻ, các thuộc tính phân cách nhau khoảng trắng.
- ▶ Các giá trị thuộc tính có thể đặt trong “ ”, ‘ ’ hoặc không có.
- ▶ Nếu giá trị là chuỗi ký tự có khoảng trắng bên trong, bắt buộc phải dùng “ ” hay ‘ ’ để bao chuỗi lại
- ▶ VD: <table border=“1” cellpadding=“5”>

# Thẻ định dạng trang

## ► Một số thuộc tính của thẻ body

Thuộc tính	Ý nghĩa	Ví dụ
bgcolor="color"	Thiết lập màu nền cho trang web	<body bgcolor="#00FF00">
background="url"	Thiết lập ảnh nền cho trang web	<body background="image.jpg">
leftmargin="num" topmargin="num"	Thiết lập lề trái và lề trên cho trang web	<body leftmargin="0" topmargin="0">
text="color"	Thiết lập màu chữ của văn bản, kề cả các đề mục	<body text="#FF0000">
alink="color" vlink="color" link="color"	Xác định màu sắc cho các siêu liên kết trong văn bản (active link, visited link, link)	<body alink ="#FF0000" vlink = "#00FF00" link = "#0000FF">

# Thẻ định dạng văn bản

Thẻ + Thuộc tính	Ý nghĩa	Ví dụ
<code>&lt;font face="fontname" size="fontsize" color="color"&gt;</code>	Thiết lập font chữ cho văn bản	<code>&lt;font face="Arial" size="5" color="#FF00FF"&gt;</code>
<code>&lt;p align=?&gt;</code> “?” = “center”   “justify”, “left”   “right”	Định dạng một đoạn văn bản	<code>&lt;p align="justify"&gt; Đoạn văn bản này được canh đều 2 biên trái phải &lt;/p&gt;</code>
<code>&lt;b&gt;</code>	Định dạng chữ in đậm	<code>&lt;b&gt; Đoạn văn bản in đậm &lt;/b&gt;</code>
<code>&lt;i&gt;</code> hay <code>&lt;em&gt;</code>	Định dạng chữ in nghiêng	<code>&lt;i&gt; Đoạn văn bản in nghiêng &lt;/i&gt;</code>
<code>&lt;u&gt;</code>	Định dạng chữ in gạch dưới	<code>&lt;u&gt; Đoan văn bản in gach dưới &lt;/u&gt;</code>
<code>&lt;strike&gt;</code>	Định dạng chữ in gạch ngang chữ	<code>&lt;strike&gt; Đoan văn bản bị gach ngang chữ &lt;/strike&gt;</code>

# Thẻ định dạng văn bản

Thẻ	Ý nghĩa	Ví dụ
<code>&lt;sup&gt;</code>	Định dạng chỉ số trên	$ax<sup>2</sup> \rightarrow ax^2$
<code>&lt;sub&gt;</code>	Định dạng chỉ số dưới	$H<sub>2</sub>O \rightarrow H_2O$
<code>&lt;h1&gt;, ..., &lt;h6&gt;</code>	Định dạng tiêu đề từ kích thước 1 đến 6 (tiêu đề 1 có kích thước lớn nhất)	<code>&lt;h1&gt;Tiêu đề 1&lt;/h1&gt;</code> <b>→ Tiêu đề 1</b>
<code>&lt;pre&gt;</code>	Bỏ qua định dạng của các thẻ HTML bên trong (không bỏ qua ký tự khoảng trắng, tab và xuống dòng)	<code>&lt;pre&gt;&lt;b&gt;Xin chào&lt;/b&gt;các bạn&lt;/pre&gt;</code> <b>→ &lt;b&gt;Xin chào&lt;/b&gt;các bạn</b>
<code>&lt;br&gt;</code>	Kết thúc dòng hiện tại và chuyển sang dòng mới	<code>&lt;p&gt;Dòng 1&lt;br&gt;Dòng 2&lt;/p&gt;</code> <b>→ Dòng 1 Dòng 2</b>

# Thẻ tạo hyperlink

---

- ▶ **Hyperlink**: cho phép người dùng có thể duyệt từ trang web này đến trang web khác.
- ▶ Gồm 3 phần:
  - ▶ Nguồn: chứa nội dung hiển thị khi user truy cập đến, có thể trang web khác, một đoạn film, một hình ảnh hoặc hộp thoại gửi email.
  - ▶ Nhãn: có thể là dòng văn bản, hình ảnh để người dùng kích vào. Nếu là văn bản thì thường được gạch dưới.
  - ▶ Đích đến (target): xác định vị trí nguồn hiển thị.

# Thẻ tạo hyperlink

---

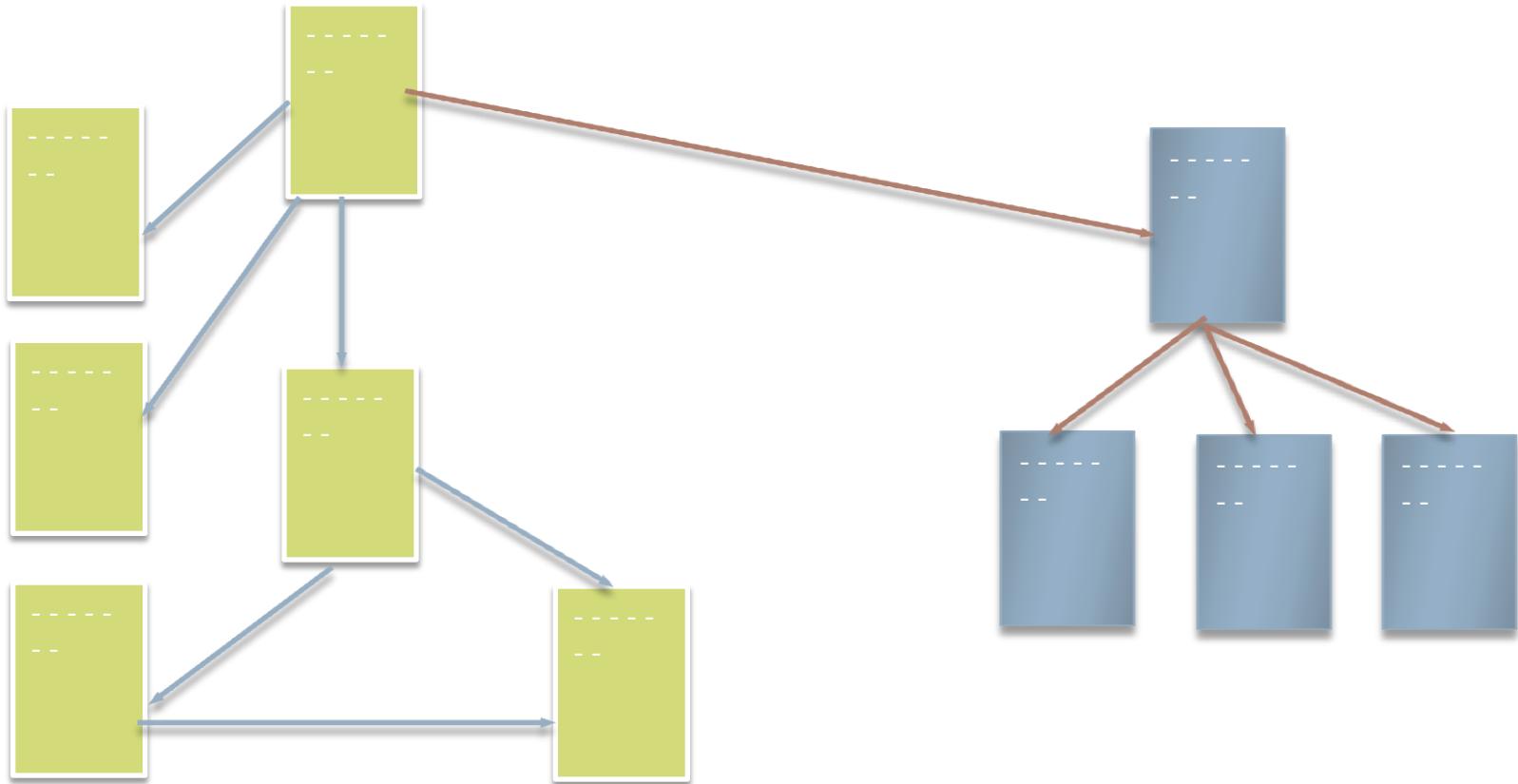
- ▶ Có hai dạng liên kết:
  - ▶ Internal hyperlink: là liên kết với các phần trong cùng một tài liệu hoặc liên kết các trang trong cùng website.
  - ▶ External hyperlink: là liên kết với các trang trên website khác
- ▶ Cú pháp:

**<A HREF="URL"> Nhãn </A>**

URL: Địa chỉ của trang liên kết

Nhãn: Có thể là dòng text hoặc hình ảnh hoặc một button

# Thẻ tạo hyperlink



# Thẻ tạo hyperlink

---

- ▶ Dùng *URL tương đối* để liên kết đến các trang trong cùng một website

Ví dụ:

<HTML>

```
<HEAD> <TITLE> Using links</TITLE></HEAD>
<BODY>
<A HREF = “Page1.htm”>
    Open Page1
</A>
</BODY>
</HTML>
```

# Thẻ tạo hyperlink

---

- ▶ Dùng *URL tuyệt đối* để liên kết đến các trang trong website khác

Ví dụ:

```
<html>
    <head><title>Link</title></head>
    <body>
        <A href="http://www.google.com">Trang google</A>
        <br>
        <A href="http://www.yahoo.com">Trang Yahoo</A>
    </body>
</html>
```

# Thẻ tạo hyperlink

---

- ▶ Liên kết đến các phần trong cùng trang:
  - ▶ Nếu nội dung trang web quá dài thì nên tạo bookmark, để giúp người nhanh chóng chuyên đến phần nội dung cụ thể nào đó.
  - ▶ Gồm hai bước
    - ▶ Tạo bookmark:  
**<A name="tên Bookmark"> Tiêu đề </A> Nội dung phần văn bản**
    - ▶ Tạo liên kết đến bookmark  
**<A Href ="#tên Bookmark"> Nhãn của liên kết </A>**

# Thẻ tạo hyperlink

---

- ▶ Liên kết đến các bookmark trong trang khác:

**<A href="http://www.site.com/path/Page.htm#tên Bookmark">**

- ▶ Ví dụ: trang main

<HTML>

<HEAD><TITLE> Main document</TITLE></HEAD>

<BODY>

    <A HREF = “C:\Doc1.htm#Internet”>Internet</A><br>

    <A HREF = “C:\Doc1.htm#HTML”>Introduction to  
    HTML</A><br>

</BODY>

</HTML>

# Thẻ tạo hyperlink

Các thuộc tính của thẻ <a>

Thuộc tính	Ý nghĩa
<code>href = "URL"</code>	URL là địa chỉ một tài liệu, hay có thể có dạng "#<LocationName>" khi liên kết đến một vị trí xác định trong trang web hiện tại
<code>tagert = "_blank"   "_self"</code>	Quy định cách mở tài liệu do URL trả tới. " <u>_blank</u> " mở ở cửa sổ (của trình duyệt) mới, " <u>_self</u> " mở ở cửa sổ hiện tại.
<code>name = "LocationName"</code>	Đặt tên cho vị trí đặt thẻ. Thuộc tính này được sử dụng khi cần liên kết đến một vị trí trong cùng trang hiện tại.
<code>tabindex = "number"</code>	Quy định thứ tự tab cho liên kết.
<code>title = "message"</code>	Chuỗi " <u>message</u> " sẽ được hiển thị khi di chuyển chuột trên liên kết.

## Thẻ tạo hyperlink

---

- ▶ Nên tạo chú thích cho liên kết giúp người đọc biết trang cần tới
- ▶ Nên dùng một màu thống nhất cho tất cả các liên kết
- ▶ Khi link không chỉ tới trang HTML mà tới một tài liệu dạng khác: doc, pdf, ppt... thì nên tạo biểu tượng bên cạnh.
- ▶ Không nên tạo link đến trang đang xây dựng.

# Thẻ định dạng danh sách

Mã HTML	Ý nghĩa
<pre>&lt;ul&gt;   &lt;li&gt;Item 1&lt;/li&gt;   &lt;li&gt;Item 2&lt;/li&gt; &lt;/ul&gt;</pre>	Tạo danh sách không đánh số thứ tự, dạng: <ul style="list-style-type: none"><li>• Item 1</li><li>• Item 2</li></ul>
<pre>&lt;ol&gt;   &lt;li&gt;Item 1&lt;/li&gt;   &lt;li&gt;Item 2&lt;/li&gt; &lt;/ol&gt;</pre>	Tạo danh sách có đánh số thứ tự, dạng: <ol style="list-style-type: none"><li>1. Item 1</li><li>2. Item 2</li></ol>

<ul> = Unordered Lists, <ol> = Ordered Lists, <li> = List Items

# Thẻ định dạng danh sách

---

## ▶ UnOrder List – UL

```
<UL Type= Shape1>
    <LI Type= Shape 2> Nội dung 1
    <LI Type= Shape 2> Nội dung 2
    ...
</UL>
```

- ▶ Shape 1, Shape 2 là bullet tự động đặt ở đầu danh sách.
- ▶ Shape 1: ảnh hưởng đến toàn danh sách
- ▶ Shape 2: ảnh hưởng đến một mục danh sách
- ▶ Các loại shape: circle (tròn rỗng), square (vuông), Disc (tròn đặc)

# Thẻ định dạng danh sách

---

## ▶ Order List – OL

```
<OL Type=x Start =n >
  <LI Type =x | Value=m> Nội dung 1
  <LI Type =x | Value=m> Nội dung 2
  ...
</OL>
```

- ▶ x là loại ký tự dùng trong ds
  - ▶ A: chữ hoa
  - ▶ a: chữ thường
  - ▶ I: số La Mã hoa
  - ▶ i: số La Mã thường
  - ▶ n: là giá trị bắt đầu
  - ▶ m: là giá trị thiết lập riêng cho item.
  - ▶ x1: là ký tự sử dụng của dòng này

# Danh sách định nghĩa

---

- ▶ Trong HTML có tag đặc biệt dùng để tạo danh sách định nghĩa phục vụ cho việc tra cứu, diễn giải.
- ▶ Cú pháp:

**<DL>**

**<DT>**Nhập từ muốn định nghĩa

**<DD>**Nhập nội dung định nghĩa

...

**</DL>**

# Danh sách định nghĩa

## ► Minh họa

<DL>

<DT>Pixel

<DD> Short for picture element. A pixel is a small dots that make up an image on the screen. Pixel depth refers to the number of colours which may be displayed.

<DT>Resolution

<DD> The quality of the display on a monitor. The higher the resolution, the sharper the image. The number of pixels that can be displayed on a screen defines resolution.

<DT>Scanner

<DD> A hardware device that allows the user to make electronic copies of graphics or text.

</DL>

### Pixel

Short for picture element. A pixel refers to the small dots that make up an image on the screen. Pixel depth refers to the number of colours which may be displayed.

### Resolution

The quality of the display on a monitor. The higher the resolution, the sharper the image. The number of pixels that can be displayed on a screen defines resolution.

### Scanner

A hardware device that allows the user to make electronic copies of graphics or text.

# Hình ảnh trên web

---

- ▶ Các loại ảnh:
  - ▶ **GIF** (Graphics Interchange Format): được sử dụng phổ biến nhất trong các tài liệu HTML, dễ chuyển tải, ngay cả các kết nối sử dụng MODEM tốc độ chậm, hỗ trợ 256 màu GIF. Các file GIF được định dạng không phụ thuộc phần nền.
  - ▶ **JPEG** (Joint PhotoGraphic Expert Group): có phần mở rộng .JPG, là **loại ảnh nén mất thông tin**, nghĩa là ảnh sau khi bị nén không giống như ảnh gốc. Tuy nhiên, trong quá trình phát lại thì ảnh cũng rõ gần như ảnh gốc. JPEG hỗ trợ hơn 16 triệu màu và thường được sử dụng cho các ảnh có màu thực.
  - ▶ **PNG** (Portable Network Graphics): **nén không mất dữ liệu**.

# thẻ chèn hình ảnh

---

## ► Cú pháp:

**<IMG Src=URL >**

- ▶ URL: thường sử dụng địa chỉ tương đối
- ▶ Nếu hình chứa trong thư mục image và tập tin .htm chứa trong thư mục html thì địa chỉ của hình chèn có dạng **Src=“..image/hinh1.gif”**
- ▶ Nếu hình và tập tin .html chứa trong cùng một thư mục thì địa chỉ hình chèn có dạng: **Src=“hinh1.gif”**

# Thẻ chèn hình ảnh - <img>

Thuộc tính	Ý nghĩa
src="URL"	URL đến tập tin ảnh
alt="description"	Định nghĩa mô tả ngắn gọn cho hình, mô tả này sẽ được hiển thị khi di chuyển chuột trên ảnh hay khi trình duyệt bị tắt chức năng duyệt ảnh
width="n" height="n"	Đặt kích thước hình theo chiều rộng (width) và chiều cao (height)
align = "top"   "middle"   "bottom"   "left"   "right"	Canh lề ảnh theo văn bản xung quanh
border="n"	Đặt kích thước đường viền bao quanh ảnh
hspace="n" vspace="n"	Định khoảng trống xung quanh ảnh theo phía trái phải (hspace), trên dưới (vspace) (tính theo pixel)

# Thẻ chèn hình ảnh

---

- ▶ Có thể chỉ xác định cho thuộc tính width hoặc height, khi đó thuộc tính còn lại sẽ được tính theo tỷ lệ
  - ▶ Vd: kích thước ảnh là 200,100 nếu xác định width là 100 thì height sẽ được tính là 50!
- ▶ Giá trị URL của thuộc tính src
  - ▶ Có thể dùng địa chỉ tương đối (không có tên miền, chỉ xác định tên thư mục và tên tập tin, nếu ảnh nằm trong site chứa trang web)
  - ▶ Hay địa chỉ tuyệt đối (URL đầy đủ đến tập tin ảnh).

# Thẻ chèn hình ảnh

---

## ▶ VD

- **Dùng địa chỉ tương đối**

- Thư mục images nằm cùng vị trí với thư mục của trang web:  
``  
(hay ``)
- Thư mục images nằm bên ngoài thư mục hiện tại 1 cấp:  
``
- Thư mục images nằm trong thư mục gốc của site:  
``

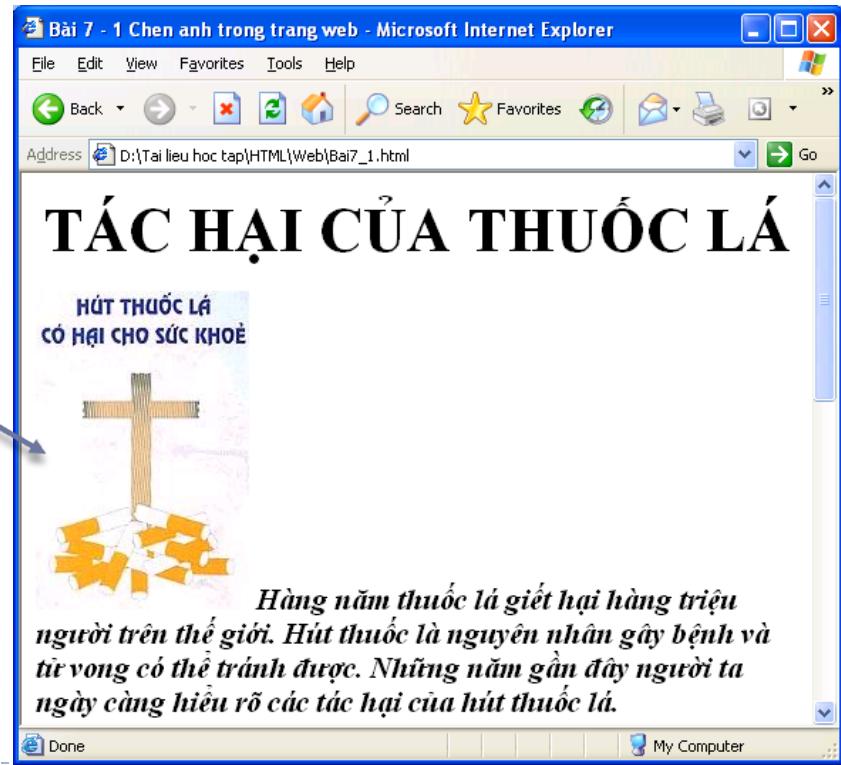
- **Dùng địa chỉ tuyệt đối**

- ``

# Thẻ chèn hình ảnh

- ▶ Các hình ảnh nên lưu trong thư mục riêng (như thư mục **image**)

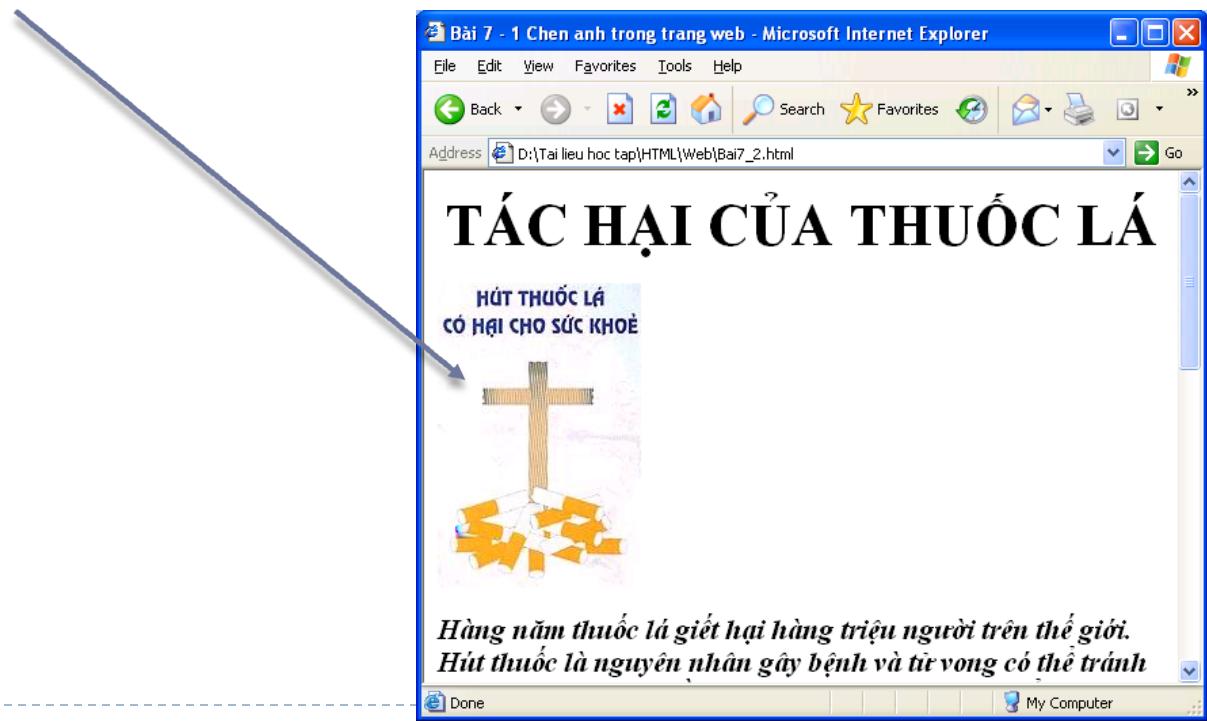




# Thẻ chèn hình ảnh

- Để hình ảnh xuất hiện riêng một dòng, ta chỉ cần thêm tag `<p>`.
- Ví dụ:

```
<p></p>
```



# Thẻ chèn hình ảnh

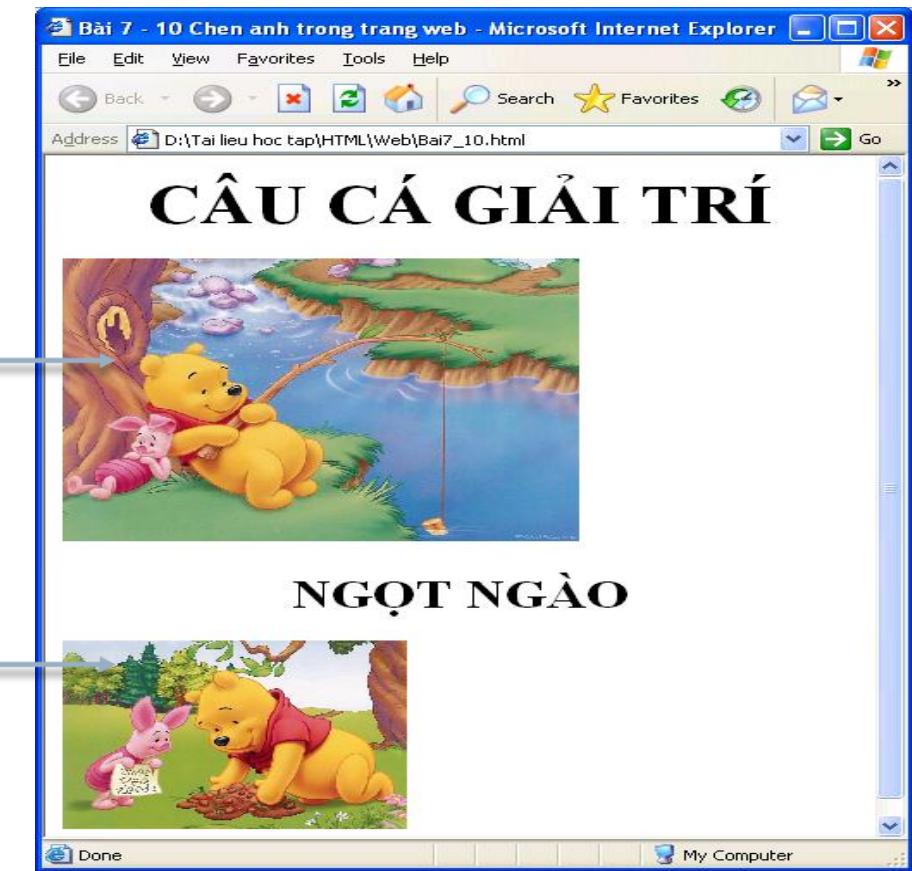
- ▶ Khi thêm hai thuộc tính Width và height giúp cho web hiển thị nhanh hơn.

```

```

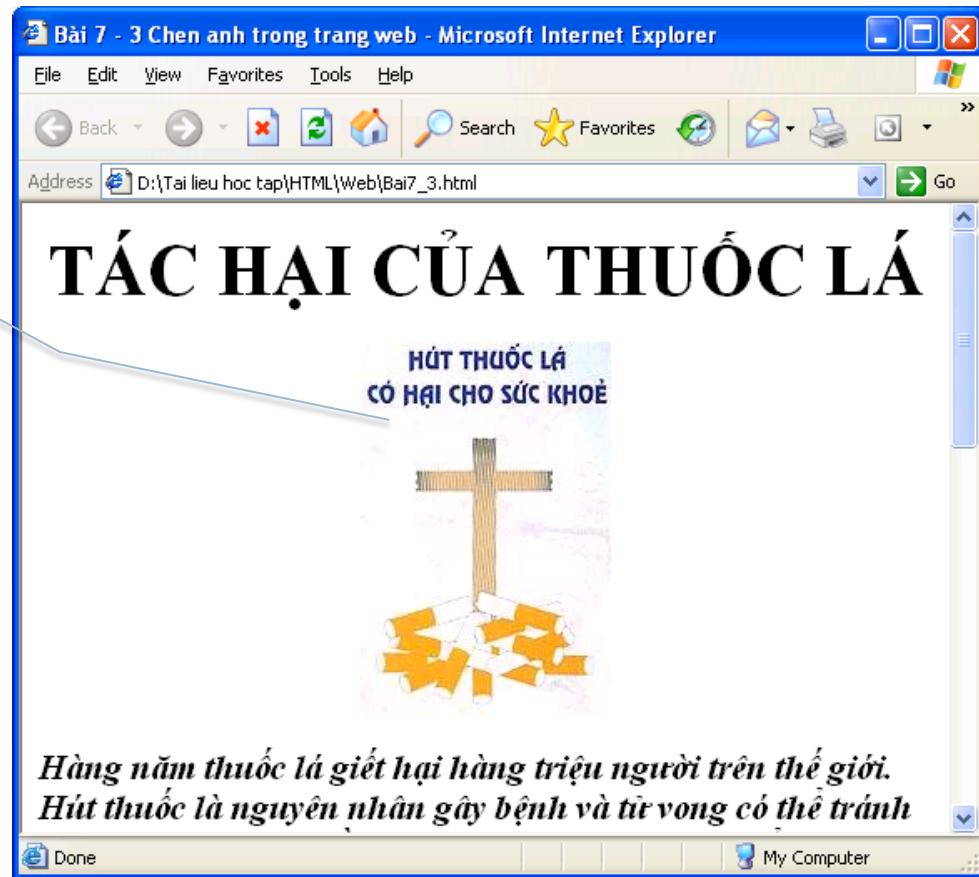
```

```



# Thẻ chèn hình ảnh

- ▶ Sắp xếp hình ảnh với văn bản xung quanh



Chèn ảnh vào  
giữa thì làm  
thế nào?

# Thẻ chèn hình ảnh

---

- ▶ Đặt tag <p> có canh lề trước tag <img>

<p align= align style> trước tag <img>  
align style:

**Left**: hình ảnh chèn vào bên trái văn bản

**Center**: hình ảnh chèn vào giữa văn bản

**Right**: hình ảnh chèn vào bên phải văn bản

# Thẻ chèn hình ảnh

---

- ▶ Sắp xếp trong hàng của văn bản và hình ảnh

`<img align=value src=FileName.gif>`

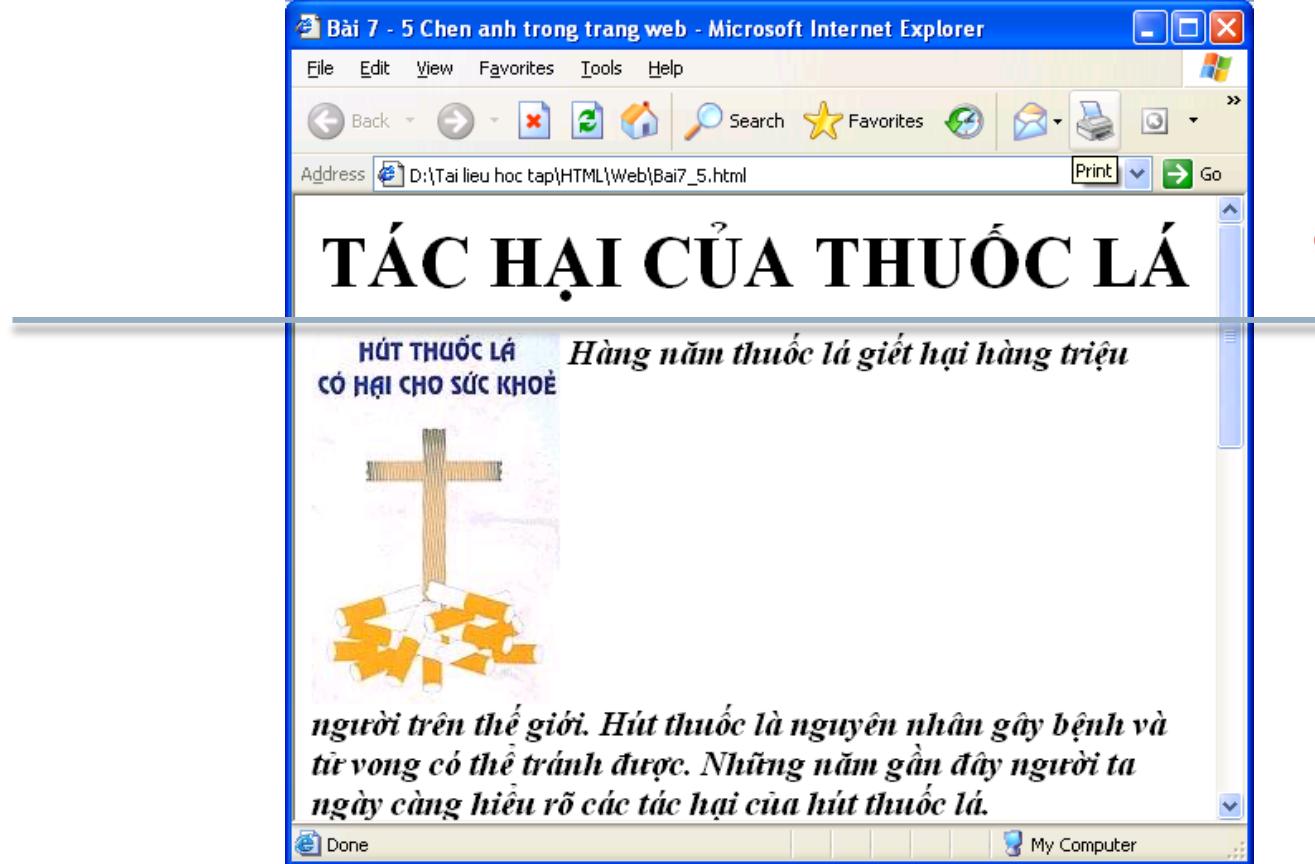
Trong đó:

- **Value** có các giá trị sau:

- **Top**: Dòng văn bản ngang với đỉnh của ảnh.
- **Middle**: Dòng văn bản nằm khoảng giữa ảnh.
- **Bottom**: Dòng văn bản bằng với đáy của ảnh. (luôn mặc định)

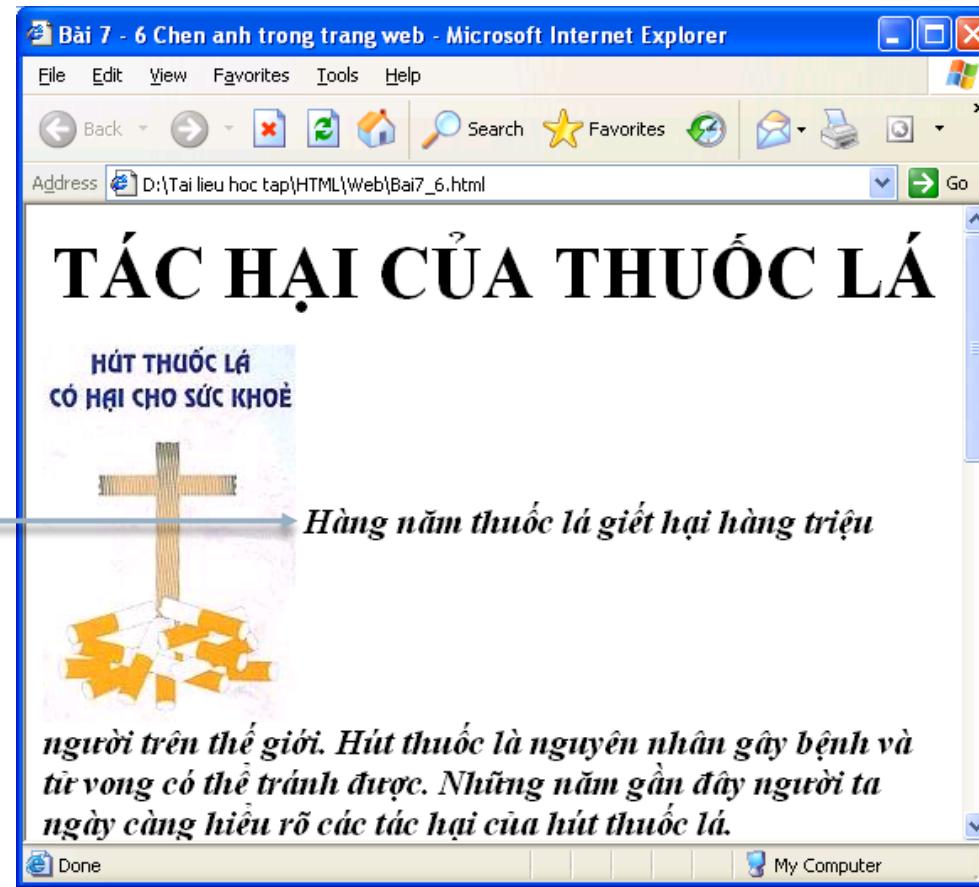
# Thẻ chèn hình ảnh





# Thẻ chèn hình ảnh

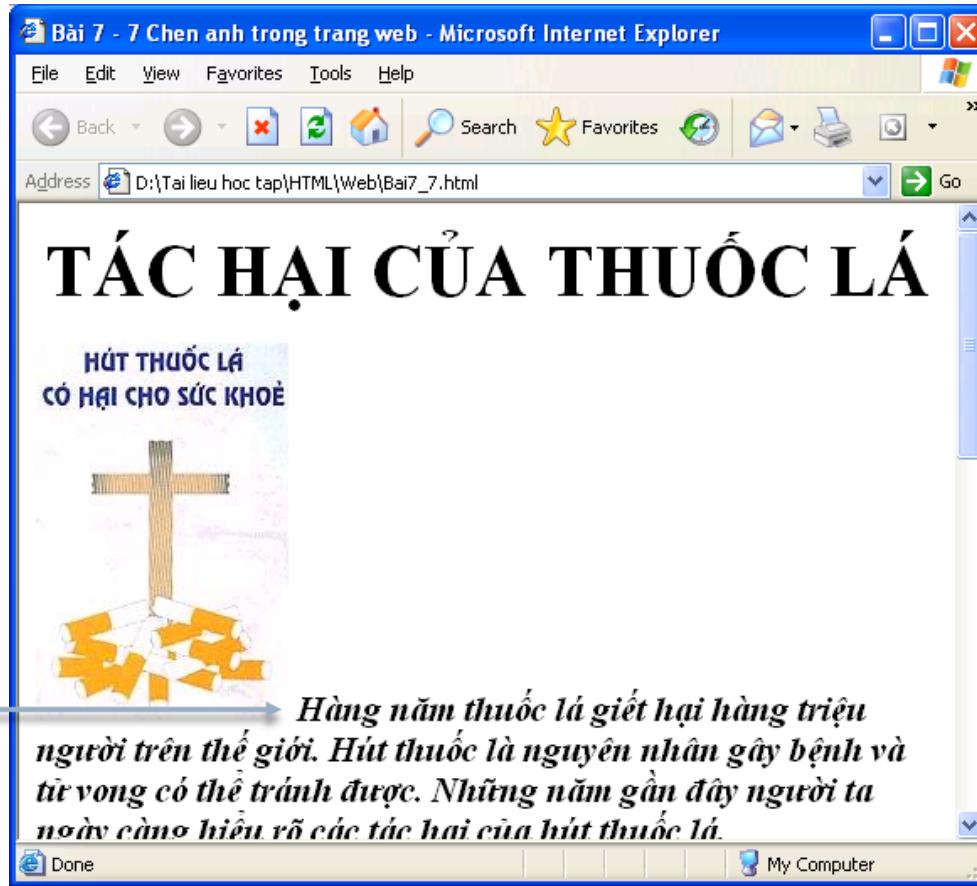
▶ 



align=middle

# Thẻ chèn hình ảnh

► 



align=bottom

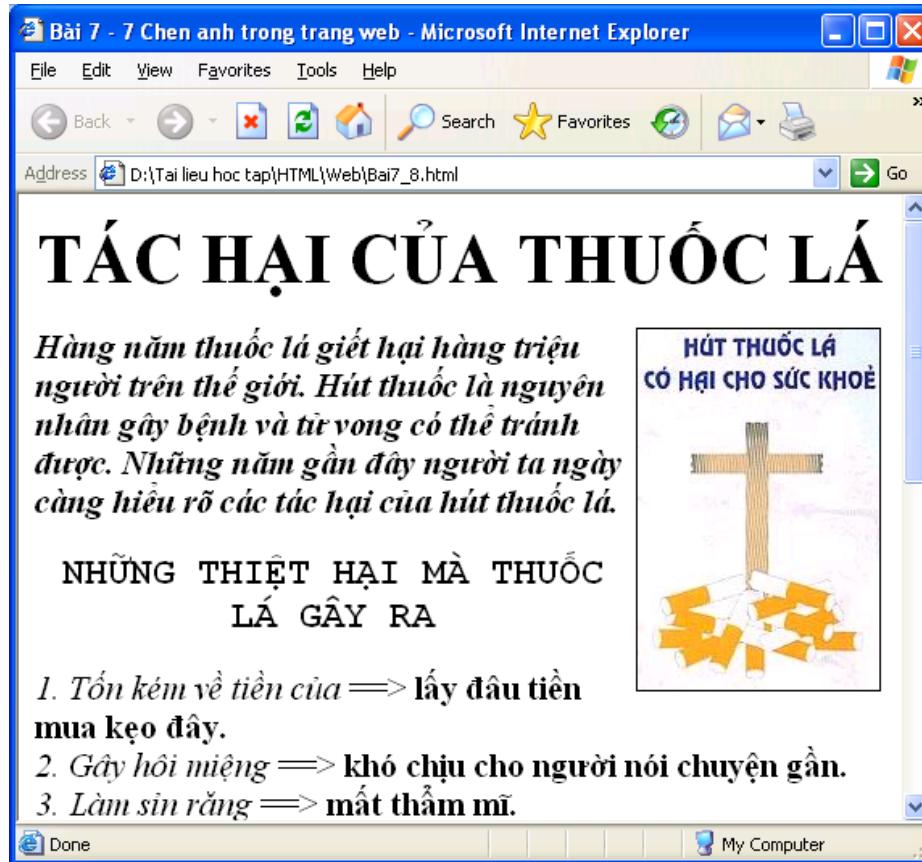
# Thẻ chèn hình ảnh

---

- Ngoài ra ta có thẻ thêm từ khóa **align=right/left** vào trong tag **<img>**.
- Để có viền khung hoặc không, ta sử dụng khoá **Border=N** trong tag **<img>**.
  - Trong đó **N** là số nguyên **lớn hơn hoặc bằng 0**, chỉ độ dày của viền khung.

# Thẻ chèn hình ảnh

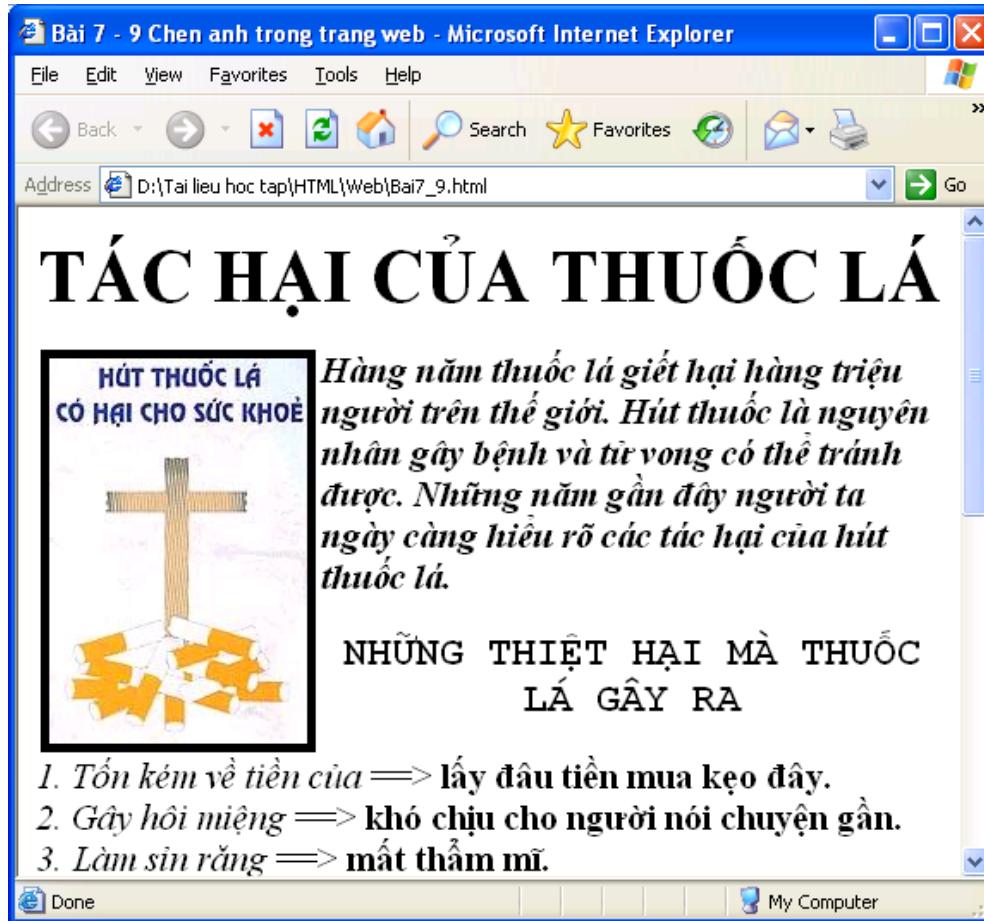
- 



# Thẻ chèn hình ảnh



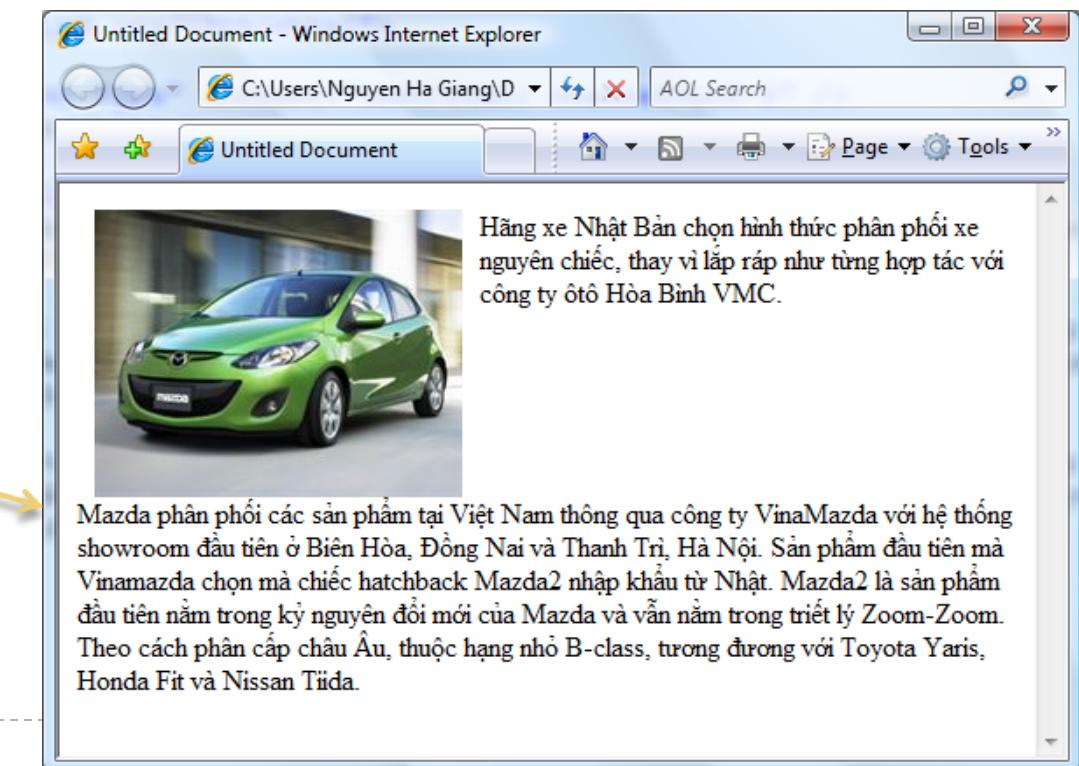
align="left"



# Thẻ chèn hình ảnh

- ▶ Khi sử dụng align = left/right, thì text sẽ wrap xung quanh ảnh cho đến phần cuối cùng của ảnh
- ▶ Muốn bỏ tác dụng này dùng <br clear="all">

<br clear="all">



# Các ký tự đặc biệt

<b>Result</b>	<b>Description</b>	<b>Entity Name</b>	<b>Entity Number</b>
	non-breaking space	&nbsp;	&#160;
<	less than	&lt;	&#60;
>	greater than	&gt;	&#62;
&	ampersand	&amp;	&#38;
"	quotation mark	&quot;	&#34;
'	apostrophe	&apos;	&#39;

<b>Result</b>	<b>Description</b>	<b>Entity Name</b>	<b>Entity Number</b>
¢	cent	&cent;	&#162;
£	pound	&pound;	&#163;
¥	yen	&yen;	&#165;
€	euro	&euro;	&#8364;
§	section	&sect;	&#167;
©	copyright	&copy;	&#169;
®	registered trademark	&reg;	&#174;
×	multiplication	&times;	&#215;
÷	division	&divide;	&#247;

# Thiết kế bảng

- Cấu trúc bảng tổng quát

```
<table>
  <tr>
    <th>Column I</th>
    <th>Column II</th>
  </tr>
  <tr>
    <td>Row 1, Col 1</td>
    <td>Row 1, Col 2</td>
  </tr>
  <tr>
    <td>Row 2, Col 1</td>
    <td>Row 2, Col 2</td>
  </tr>
</table>
```

Thẻ	Ý nghĩa
<table>	Tạo bảng
<tr>	Tạo dòng
<th>	Tạo cột tiêu đề
<td>	Tạo cột



Column I	Column II
Row 1, Col 1	Row 1, Col 2
Row 2, Col 1	Row 2, Col 2

# Thiết kế bảng - thuộc tính thẻ Table

Thuộc tính	Ý nghĩa
bgcolor="color"	Đặt màu nền
background="URL"	Đặt hình nền
border="n"	Độ dày của đường viền
bordercolor="color"	Màu của đường viền
cellpadding="n"	Khoảng cách giữa nội dung và đường viền
cellspacing="n"	Khoảng cách giữa các ô
width="n" height="n"	Kích thước chiều ngang và chiều cao

# Thiết kế bảng - thuộc tính thẻ <tr>

Thuộc tính	Ý nghĩa
bgcolor="color"	Đặt màu nền
bordercolor="color"	Màu của đường viền
align= "center"   "justify"   "left"   "right"	Canh lè theo phương ngang
valign= "bottom"   "middle"   "top"	Canh lè theo phương đứng

# Thiết kế bảng - thẻ <th>, <td>

Thuộc tính	Ý nghĩa
bgcolor="color"	Đặt màu nền
background="URL"	Đặt hình nền
bordercolor="color"	Màu của đường viền
align= "center"   "justify"   "left"   "right"	Canh lè theo phương ngang
valign= "bottom"   "middle"   "top"	Canh lè theo phương đứng
rowspan="n" colspan="n"	Gộp nhiều dòng/cột lại thành 1
width="n" height="n"	Kích thước chiều ngang và chiều cao

# Thiết kế bảng

```
<table border="5" cellpadding="20" cellspacing="10" width="600" height="400">
  <tr bgcolor="#6699FF">
    <th>Header 1</td>
    <th>Header 2</td>
    <th>Header 3</td>
  </tr>
  <tr align="center" valign="top">
    <td>Row 1, Col 1</td>
    <td valign="middle">Row 1, Col 2</td>
    <td valign="bottom">Row 1, Col 3</td>
  </tr>
  <tr align="center" valign="middle">
    <td background="texture.bmp">Row 2, Col 1</td>
    <td colspan="2" bgcolor="#FFCC00">Row 2, Col 2</td>
  </tr>
</table>
```

Header 1	Header 2	Header 3
Row 1, Col 1	Row 1, Col 2	Row 1, Col 3
Row 2, Col 1	Row 2, Col 2	

# Dùng table để trình bày trang

---

- ▶ Bảng thường được dùng để trình bày bố cục (layout) trang web
- ▶ Dùng table để
  - ▶ thiết kế một trang thể hiện văn bản dạng cột báo chí,
  - ▶ phân trang thành các vùng có chủ đề khác nhau
- ▶ Mỗi **cell** trong table có thể sử dụng bất cứ tag HTML.
  - ▶ Chèn một danh sách có thứ tự trong cell
  - ▶ Chèn một tag <h1>
  - ▶ Hoặc chèn một table vào trong cell

# Dùng table để trình bày trang

- Tạo một trang có một dòng và 2 cột

```
<table>
  <tr>
    <td>
      <ol>
        <li>Home Page
        <li>View the Catalog
        <li>Place an Order
      </ol>
    </td>
    <td>
      <table>
        <!--Nội dung của table 2-->
      </table>
    </td>
  </tr>
</table>
```

Phần danh mục  
liên kết

Bảng 2


# Dùng table để trình bày trang

```
<table>
  <tr>
    <td colspan =2>
      <!--Chèn hình logo-->
      chèn logo
    </td>
  </tr>
  <tr>
    <td rowspan =2>
      <!--Nội dung 1 -->
      Nội dung 1
    </td>
    <td>
      <!--Nội dung 2 -->
      Nội dung 2
    </td>
  </tr>
  <tr>
    <td>
      <!--Nội dung 3 -->
      Nội dung 3
    </td>
  </tr>
</table>
```

Table 2

chèn logo

Nội dung 2

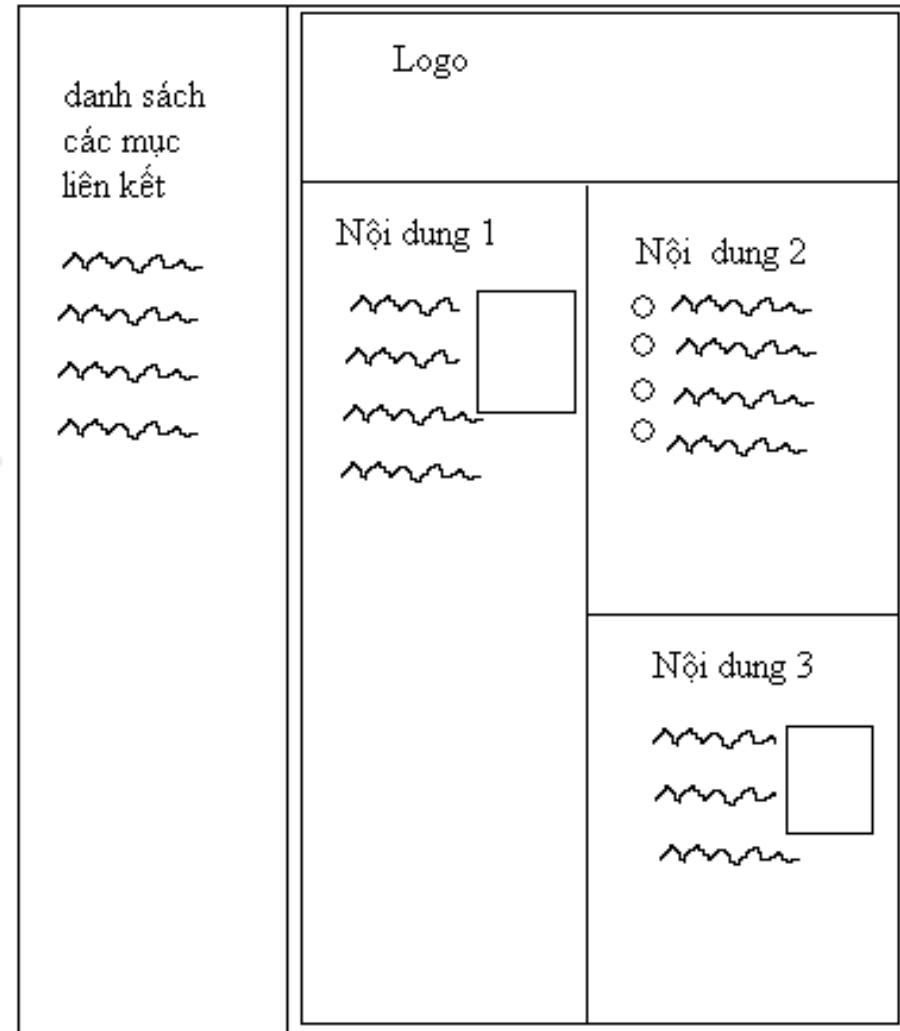
Nội dung 1

Nội dung 3

# Dùng table trình bày trang

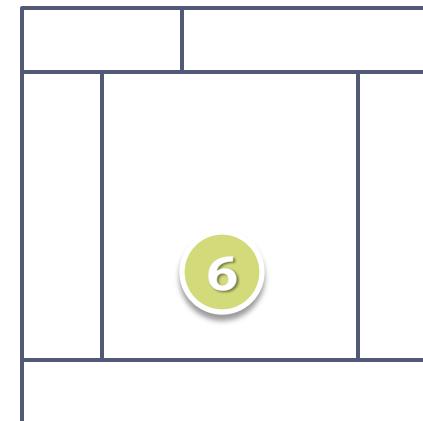
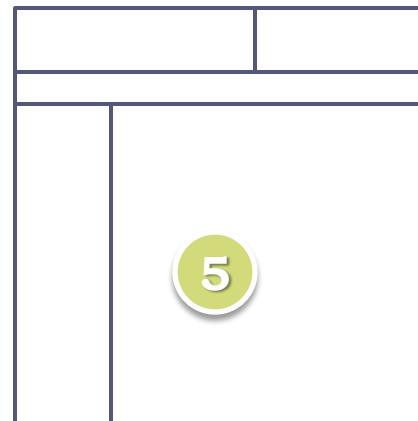
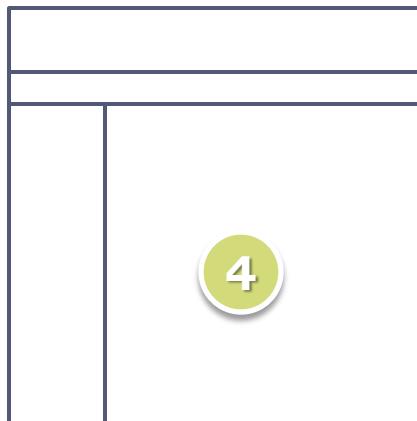
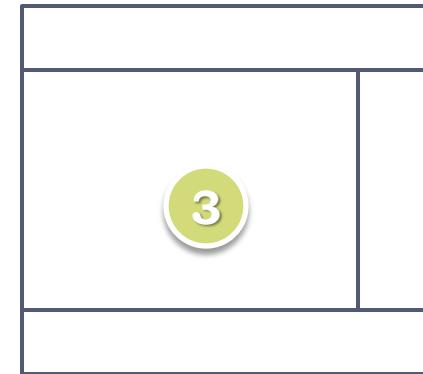
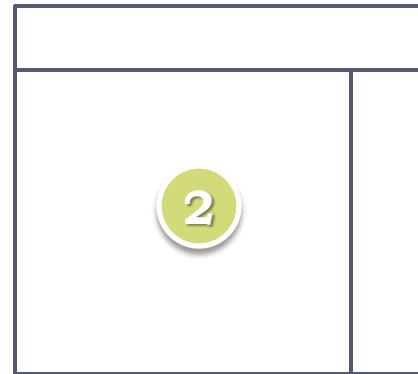
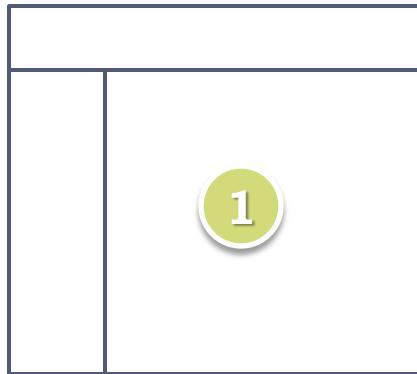
- ▶ Kết quả trình bày khi ghép lại

**Bộ cục của một trang web**



# Dùng table trình bày trang

- Thiết kế mẫu sau



# Frame

---

- ▶ Mở rộng khả năng hiển thị trang web bằng cách cho phép chia miền hiển thị thành **nhiều vùng**
- ▶ Vùng được chia gọi là frame, có đặc điểm sau:
  - ▶ Có thể truy cập đến **URL độc lập** với frame khác
  - ▶ Có thể thay đổi kích thước khung nhìn, hoặc không cho phép thay đổi đối với user
- ▶ Trong trang đã dùng frame thì không có tag body

# Frame

---

- ▶ Cú pháp của Frame

<HTML>

<HEAD>

</HEAD>

<FRAMESET>

<frame ...>

</FRAMESET>

</HTML>

# Frame

---

## Thẻ Frameset

```
<Frameset rows = "n1, n2, ... "
           cols = "m1, m2, ..."
           border = "N"
           framespacing = "N"
           bordercolor = #xxyyzz
           frameborder = "yes/no">
```

- ▶ **rows**: chỉ chia hàng, tùy theo tham số.
- ▶ **cols**: chỉ chia cột, tùy theo tham số.
- ▶ **border**: độ dày đường viền.
- ▶ **framespacing**: khoảng cách các frame.

# Frame

---

- ▶  $n_1, n_2, m_1, m_2, \dots$ : là giá trị thuộc tính được tính bằng pixel hoặc phần trăm tương đối. Và có thể là dấu '\*', frame tương ứng sẽ tự động điều chỉnh.
- ▶ N: là độ dày tương ứng với các thuộc tính.
- ▶ frameborder: thiết lập đường viền cho toàn bộ tập frame, có giá trị yes hoặc no.

# Frame

---

## ▶ Ví dụ

<frameset rows = "30%, \*">

*chia thành 2 hàng, hàng thứ nhất chiếm 30% màn hình, còn lại là hàng thứ 2.*

<frameset cols = "\*", 100, \*">

*chia 3 cột, cột thứ 2 có độ rộng 100 điểm, các cột còn lại sẽ tự động phân chia.*

<frameset rows = "30%, \*" border=0 framespacing = 5 frameborder = 0>:

*đặt đường viền, khoảng cách, ...*

# Frame

---

- ▶ Thủ thuật frame

- ▶ Cú pháp

```
<frame  src="URL"
       name="tên cửa sổ"
       width = "N"
       height = "M"
       scrolling = "yes/no/auto"
       frameborder = "yes/no"
       framespacing = "N"
       bordercolor = #xxyyzz
       noresize >
```

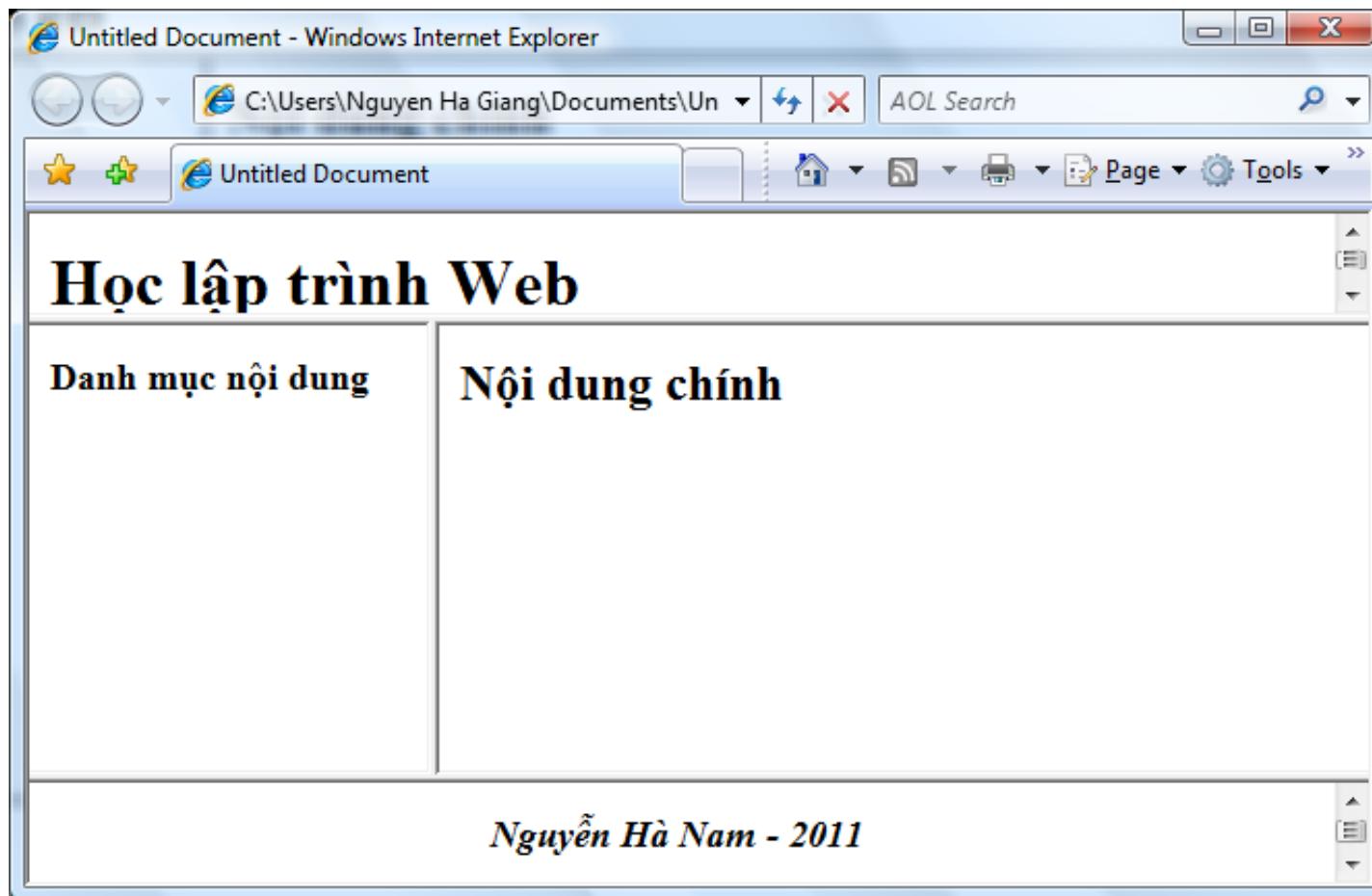
# Frame

---

- **noresize**: nếu có thuộc tính này thì người sử dụng không thể thay đổi kích thước hiển thị frame.
- **name**: gán một tên cho một frame, nó có thể làm đích cho các liên kết từ các tài liệu khác.
- **width**: chiều rộng frame.
- **height**: chiều cao frame.
- **scrolling**: đặt thuộc tính thanh cuộn.

# Frame

- ▶ VD tạo trang web có 4 frame như sau



# Frame

---

- ▶ Phần HTML của trang chính

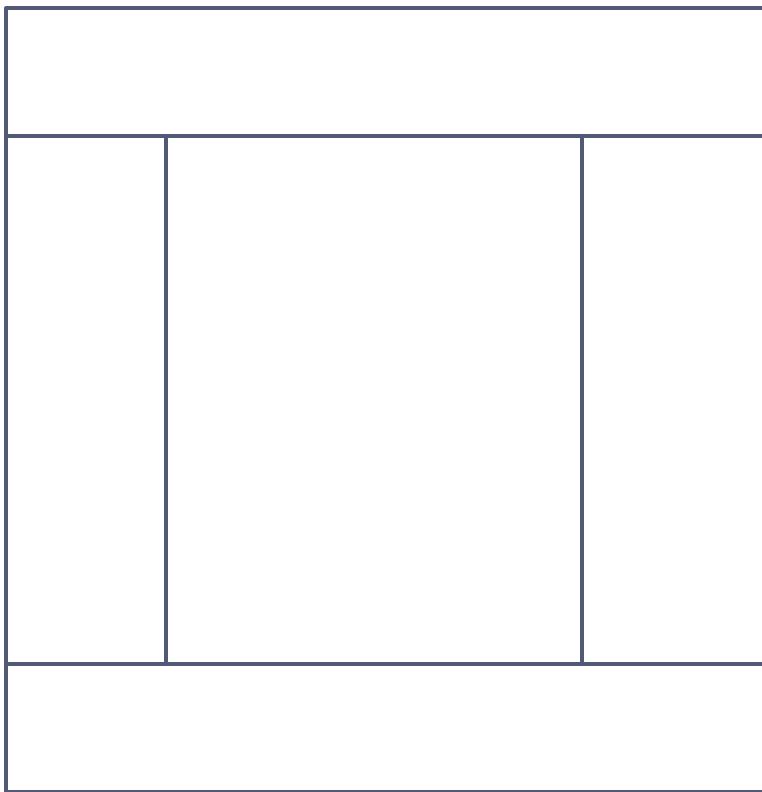
```
<frameset rows="50,* ,50">
    <frame src="header.html" />
    <frameset cols="30%,* " >
        <frame src="left.html" />
        <frame src="main.html" />
    </frameset>
    <frame src="footer.html" />
    <noframes> Trình duyệt không hỗ trợ Frame
</noframes>
</frameset>
```

- ▶ Tạo các trang con Header.html, left.html, main.html, footer.html

# Frame

---

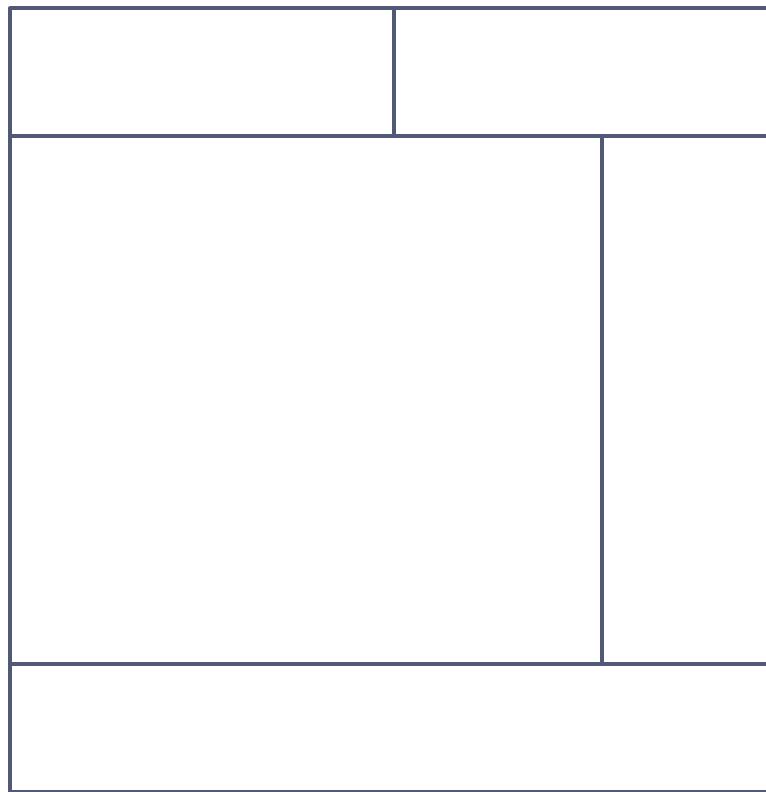
- Thiết kế frame có dạng sau



# Frame

---

- Thiết kế frame có dạng sau



# Form

---

- ▶ Form dùng để nhận thông tin từ người sử dụng hay phản hồi thông tin về người sử dụng.
- ▶ Người dùng có thể có các yêu cầu
  - ▶ Nhập vào câu trả lời, ý kiến
  - ▶ Chọn câu trả lời từ danh sách
  - ▶ Chọn câu trả lời từ một hoặc một số tùy chọn
- ▶ Dữ liệu có thể xử lý tại **client** hoặc có thể xử lý ở **server**. Sau đó kết quả trả về cho người dùng.
- ▶ Sử dụng tag <form> để tạo form trong trang web

# Form

## ▶ Thuộc tính của tag form

Thuộc tính	Ý nghĩa
action="url"	Địa chỉ của trang web/chương trình xử lý dữ liệu được nhập trên form
method="get"   "post"	Phương thức HTTP dùng để gửi dữ liệu đến địa chỉ URL được xác định trong thuộc tính action. Nếu không chỉ rõ phương thức gửi dữ liệu, mặc định sử dụng phương thức "get"
name = "form_name"	Tên của form
target= "frame_name"   " <u>blank</u> "   "_self"	Chỉ định cửa sổ hiển thị kết quả (trong một frame, cửa sổ mới hay trong cửa sổ hiện tại)
enctype= "application/x-www-form-urlencoded"   "text/plain" "multipart/form-data"	Quy định định cách định dạng dữ liệu trước khi gửi. Nếu không xác định rõ, enctype mặc định sẽ là "application/x-www-form-urlencoded"

# Form

---

- ▶ Phương thức HTTP (HTTP methods)
  - ▶ **Post**: dữ liệu chứa trong phần thân của request
  - ▶ **Get**: dữ liệu được gởi kèm theo URL được mô tả trong action, có dạng sau
    - ▶ URL?name1= value1& name2=value2
- ▶ Lưu ý:
  - ▶ Khi dùng Get, tất cả thông tin thu được sẽ hiển thị lên address bar
  - ▶ Thường dùng get khi cần bookmark trang hiện hành do các thông tin cần lấy không mang tính quan trọng, bảo mật.
  - ▶ Post thường được dùng để che dấu thông tin

# Form

Thành phần	Ý nghĩa
TextBox	Hộp nhập dữ liệu trên một dòng
TextArea	Hộp nhập dữ liệu trên nhiều dòng
PasswordBox	Hộp nhập mật khẩu
FileField	Hộp chọn tập tin
Button	Nút (submit dữ liệu hay xử lý thao tác nào đó khi người sử dụng click chuột vào nút này)
RadioButton	Chọn lựa một trong các lựa chọn được liệt kê
CheckBox	Chọn lựa một hay nhiều lựa chọn được liệt kê
ComboBox	Chọn một mục trong danh sách được liệt kê
ListBox	Chọn một hay nhiều mục trong danh sách

# Form

## ► Textbox

Mã HTML	Kết quả trên trình duyệt
<pre>&lt;form action="form_action.asp" method="get"&gt; First name: &lt;input type="text" name="fname" value="Mickey" /&gt; &lt;br /&gt; Last name: &lt;input type="text" name="lname" value="Mouse" /&gt; &lt;br /&gt; &lt;input type="submit" value="Submit" /&gt; &lt;/form&gt;</pre>	<p>First name: Mickey</p> <p>Last name: Mouse</p> <p>Submit</p>

## ► TextArea

Mã HTML	Kết quả trên trình duyệt
<pre>&lt;textarea rows="2" cols="20"&gt; The cat was playing in the garden. Suddenly a dog showed up..... &lt;/textarea&gt;</pre>	<p>The cat was playing in the garden. Suddenly a dog</p>

# Form

- ▶ Textbox, passwordBox, Reset, Submit Button

Mã HTML	Kết quả trên trình duyệt
<pre>Username: &lt;input type="text" name="user"&gt; &lt;br /&gt; Password: &lt;input type="password" name="password"/&gt; &lt;br /&gt; &lt;input type="reset" value="Reset"/&gt; &lt;input type="submit" value="Submit"/&gt;</pre>	

- ▶ FileField

Mã HTML	Kết quả trên trình duyệt
<pre>&lt;form action="#" method="post" name="MyForm"&gt; &lt;input name="MyFile" type="file" size="20"&gt; &lt;/form&gt;</pre>	

# Form

## ▶ RadioButton

Mã HTML	Kết quả trên trình duyệt
Male: <input type="radio" checked="checked" name="Sex" value="male" />   Female: <input type="radio" name="Sex" value="female" />	Male: <input checked="" type="radio"/> Female: <input type="radio"/>

## ▶ CheckBox

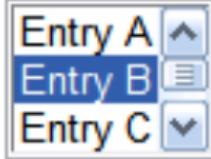
Mã HTML	Kết quả trên trình duyệt
I have a bike: <input type="checkbox" name="vehicle" value="Bike">   I have a car: <input type="checkbox" name="vehicle" value="Car">   I have an airplane: <input type="checkbox" name="vehicle" value="Airplane">	I have a bike: <input type="checkbox"/> I have a car: <input type="checkbox"/> I have an airplane: <input type="checkbox"/>

# Form

## ▶ ComboBox

Mã HTML	Kết quả trên trình duyệt
<pre>&lt;select name="cars"&gt; &lt;option value="volvo"&gt;Volvo&lt;/option&gt; &lt;option value="saab"&gt;Saab&lt;/option&gt; &lt;option value="fiat" selected="selected"&gt;Fiat&lt;/option&gt; &lt;option value="audi"&gt;Audi&lt;/option&gt; &lt;/select&gt;</pre>	

## ▶ ListBox

Mã HTML	Kết quả trên trình duyệt
<pre>&lt;select size="3" multiple="multiple"&gt; &lt;option&gt;Entry A&lt;/option&gt; &lt;option selected="selected"&gt;Entry B&lt;/option&gt; &lt;option&gt;Entry C&lt;/option&gt; &lt;/select&gt;</pre>	

# Form

---

- ▶ **FieldSet**: nhóm các đối tượng giống nhau vào một phần logic

<Fieldset>

    <Legend Align="left/right">Chú thích

    </Legend>

    Các thành phần trong nhóm

</Fieldset>

- ▶ Tag legend: tạo chú thích của nhóm
- ▶ Cho phép canh lề chú thích

# Form

---

```
<FORM>
<FIELDSET>
    <LEGEND>Position</LEGEND>
    Application for the post of: <INPUT name= 'name'
    type= 'text' tabindex= '1'>
</FIELDSET>
<FIELDSET>
    <LEGEND>Educational Qualifications</LEGEND>
    <INPUT name= 'qualif' type='radio' value= 'grad'
    tabindex= '5'> Graduate
    <INPUT name= 'qualif' type='radio' value='postgrad'
    tabindex='5'> Postgraduate
</FIELDSET>
</FORM>
```

# Form

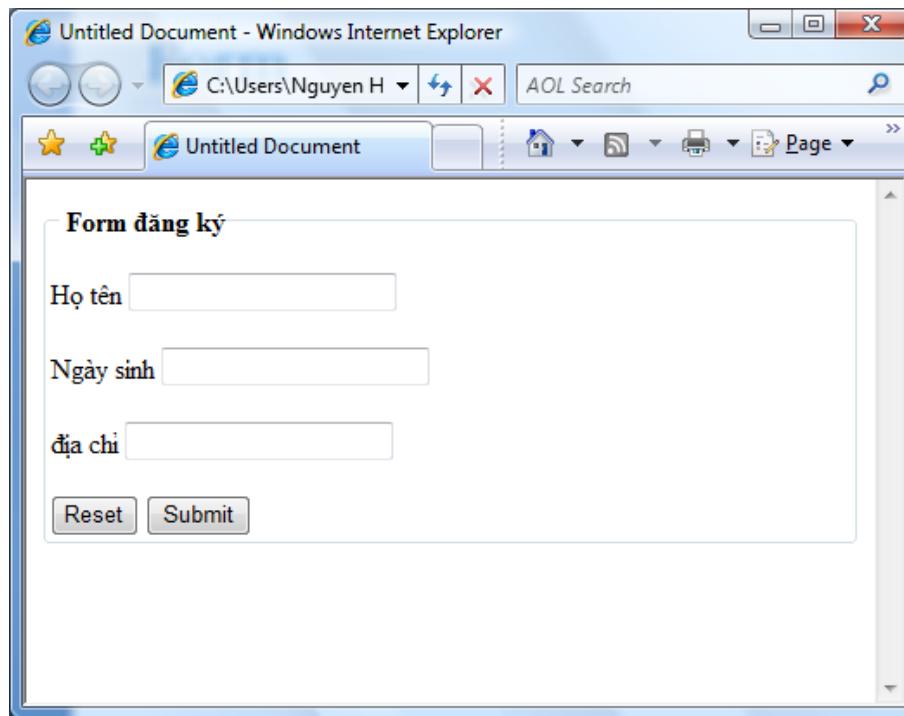
## ▶ Kết quả

The screenshot shows a Windows Internet Explorer window titled "Untitled Document - Windows Internet Explorer". The address bar displays the path "C:\Users\Nguyen Ha Giang\Documents\Untitled". The toolbar includes standard buttons for Back, Forward, Stop, AOL Search, Home, Print, and Tools. The main content area contains a form with the following fields:

- Position**: A text input field labeled "Application for the post of:".
- Educational Qualifications**: A group of radio buttons labeled "Graduate" and "Postgraduate".

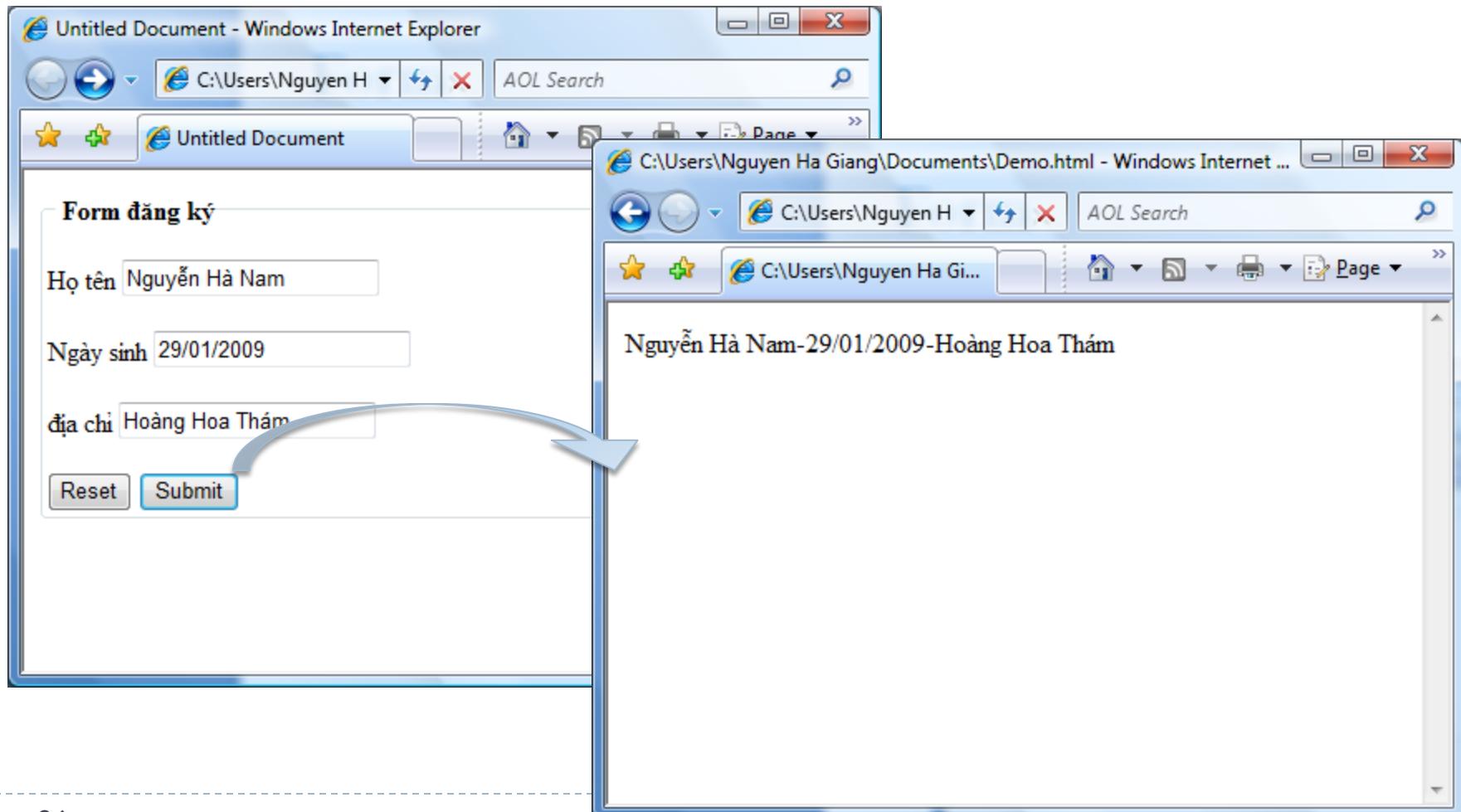
# Form

- ▶ Tạo form có dạng sau
  - ▶ Form cho phép user nhập vào các thông tin và hiển thị lại các thông tin user đã nhập.



# Form

## ▶ Nhập liệu



# XHTML

---

- ▶ XHTML = EXtensible HyperText Markup Language
- ▶ XHTML là sự kết hợp giữa HTML và XML
- ▶ XHTML gồm tất cả các phần tử (element) trong HTML 4.1 và được kết hợp theo cú pháp của XML

# Tại sao sử dụng XHTML

---

- ▶ Vì XHTML là sự kết hợp giữa HTML và XML, do đó tài liệu XHTML chính là một tài liệu HTML nhưng được bổ sung thêm **sự chặt chẽ** trong cú pháp của XML
- ▶ (đảm bảo cấu trúc của trang web luôn thoả điều kiện “well-formed”)
- ▶ Trình duyệt có thể sẽ hiển thị trang XHTML nhanh hơn trang HTML (vì không mất thời gian để kiểm tra và sửa lỗi “well-formed”)

# Well-formed element

---

- ▶ Đối với các phần tử có chứa nội dung, phải có đầy đủ thẻ mở và thẻ đóng.
- ▶ Đối với các phần tử rỗng, phải được thêm vào ký tự kết thúc “/”.
- ▶ Các thẻ lồng nhau phải đúng trật tự, không chồng lấp lên nhau.
- ▶ Thỏa well-formed element  
`<br /> <hr /> `
- ▶ Không thỏa well-formed element  
`<i> <b> wrong </i>`

# Well-formed document

---

- ▶ Các phần tử phải well-formed
- ▶ Tồn tại phần tử đơn, xem như phần tử gốc, chứa tất cả các phần tử khác

*<!-- Wrong! Not well-formed HTML! -->*

```
<p>Normal <em>emphasized <strong>strong emphasized</em>
strong</strong></p>
```

*<!-- Correct: Well-formed HTML. -->*

```
<p>Normal <em>emphasized <strong>strong emphasized</strong></em>
<strong>strong</strong></p> <p>Alternatively <em>emphasized</em>
<strong><em>strong emphasized</em> strong</strong></p>
```

# So sánh HTML vs XHTML

---

## ▶ XHTML

- ▶ Các thẻ lồng nhau phải **đúng trật tự**
- ▶ Các thẻ và các thuộc tính của các phần tử cần phải được viết bằng **chữ thường**
- ▶ Các thuộc tính của các phần tử phải được đóng **nháy kép**. Ví dụ: <table border="1" width="100%">
- ▶ Không cho phép giản lược các thuộc tính.
  - ▶ Ví dụ trong HTML có thể viết <option selected> nhưng trong XHTML cần phải được viết lại <option selected="selected"> và phải được đóng bằng thẻ </option>

# So sánh HTML vs XHTML

---

- ▶ Các thẻ được gọi là noempty cần phải được đóng bằng một thẻ đóng.
- ▶ Những thẻ không gọi là noempty như `<br>` có thể được đóng bằng một thẻ `</br>` hoặc nếu không phải được viết `<br />`.
- ▶ Thuộc tính **name** được thay thế bằng thuộc tính **id**
- ▶ Phải có khai báo **DOCTYPE** trước thẻ `<html>`

# Cấu trúc tối thiểu tập tin XHTML

---

- ▶ <!DOCTYPE Doctype goes here>
- <html>
- <head>
- <title>Title goes here</title>
- </head>
- <body>
- Body text goes here
- </body>
- </html>