

Lập trình Web 1

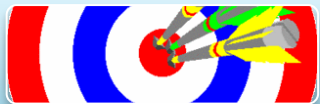


Khoa Công nghệ thông tin
Bộ môn Công nghệ phần mềm

Giới thiệu môn học

Nguyễn Huy Khánh
nhkhanh@fit.hcmus.edu.vn

Nội dung



Mục tiêu môn học



Nội dung môn học



Công cụ sử dụng



Kiến thức yêu cầu



Tài liệu tham khảo



Đánh giá kết quả



Liên hệ

Mục tiêu môn học



- Nắm vững các khái niệm căn bản của quá trình thiết kế, cài đặt và triển khai một ứng dụng web
- Thiết kế trang Web và cài đặt một ứng dụng Web hoàn chỉnh
- Triển khai và khai thác ứng dụng web trên Intranet/Internet

Nội dung môn học



- Giới thiệu về Internet
- Thiết kế trang web DHTML
 - Ngôn ngữ HTML
 - Ngôn ngữ CSS
 - Ngôn ngữ JavaScript
 - HTML DOM
- Tạo web động bằng
 - Ngôn ngữ PHP
 - Cơ sở dữ liệu MySQL

Công cụ sử dụng



- **Thiết kế trang web HTML**
 - Notepad++
 - Adobe Dreamweaver
- **Lập trình Javascript**
 - Visual Studio
 - Adobe Dreamweaver
- **Lập trình PHP**
 - Dreamweaver
 - Eclipse PDT
 - Netbeans
- **Gói web server**
 - XAMPP
 - WampServer

Kiến thức yêu cầu



- Kỹ năng lập trình C++
- Kiến thức về hệ quản trị cơ sở dữ liệu quan hệ
- Kỹ năng sử dụng ngôn ngữ truy vấn SQL

Tài liệu tham khảo



● Ebook

- Beginning HTML with CSS and XHTML - Modern Guide and Reference
- CSS Mastery Advanced Web Standers Solutions Second Edition
- Build Your Own Database Driven Website Using PHP & MySQL 4th Edition
- PHP and MySQL: Create - Modify - Reuse

● Web

- <http://www.w3.org/>
- <http://www.w3schools.com/>





- Diễn đàn môn học (Moodle)
 - <http://courses.fit.hcmus.edu.vn/course/view.php?id=1170>
- Email: nhkhanh@fit.hcmus.edu.vn
- Quy tắc gửi mail
 - Tiêu đề: **[Web1-Lop] – Nội dung tiêu đề**
 - Trong mail ghi rõ họ tên và MSSV
 - Ví dụ:
 - **[Web1-08NT] – Hiện tên cột trong DataGrid**

Lập trình Web 1



Khoa Công nghệ thông tin
Bộ môn Công nghệ phần mềm

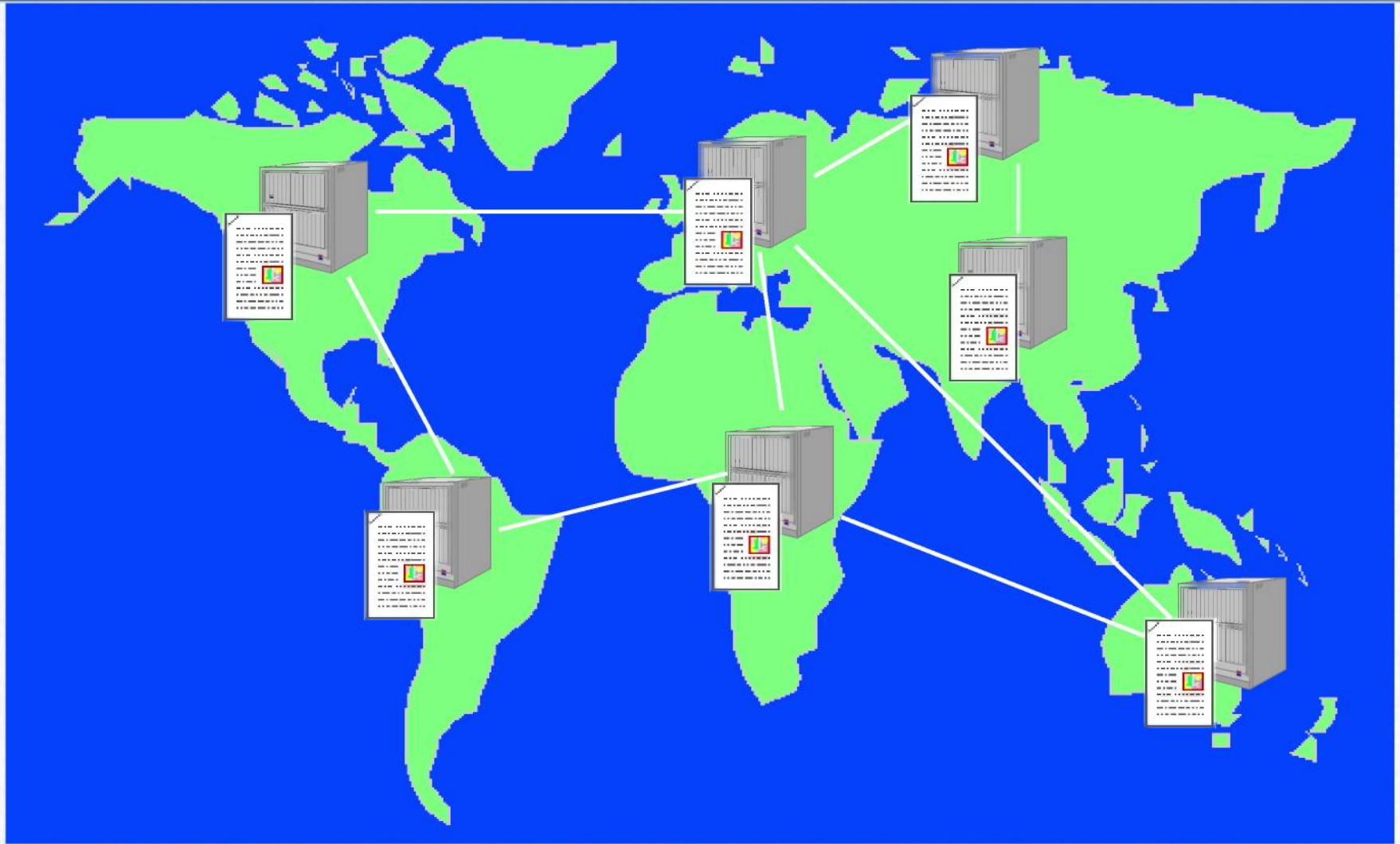
Giới thiệu Internet và Web

Nguyễn Huy Khánh
nhkhanh@fit.hcmus.edu.vn

Nội dung

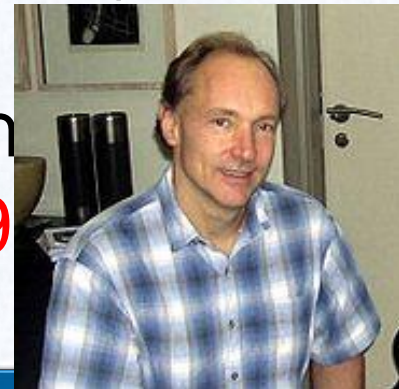
- Giới thiệu lịch sử
- Các khái niệm căn bản
- Các dịch vụ cơ bản của Internet
- Khai thác tài nguyên trên Internet
- Các bước thiết lập website

Internet - Mạng các máy tính

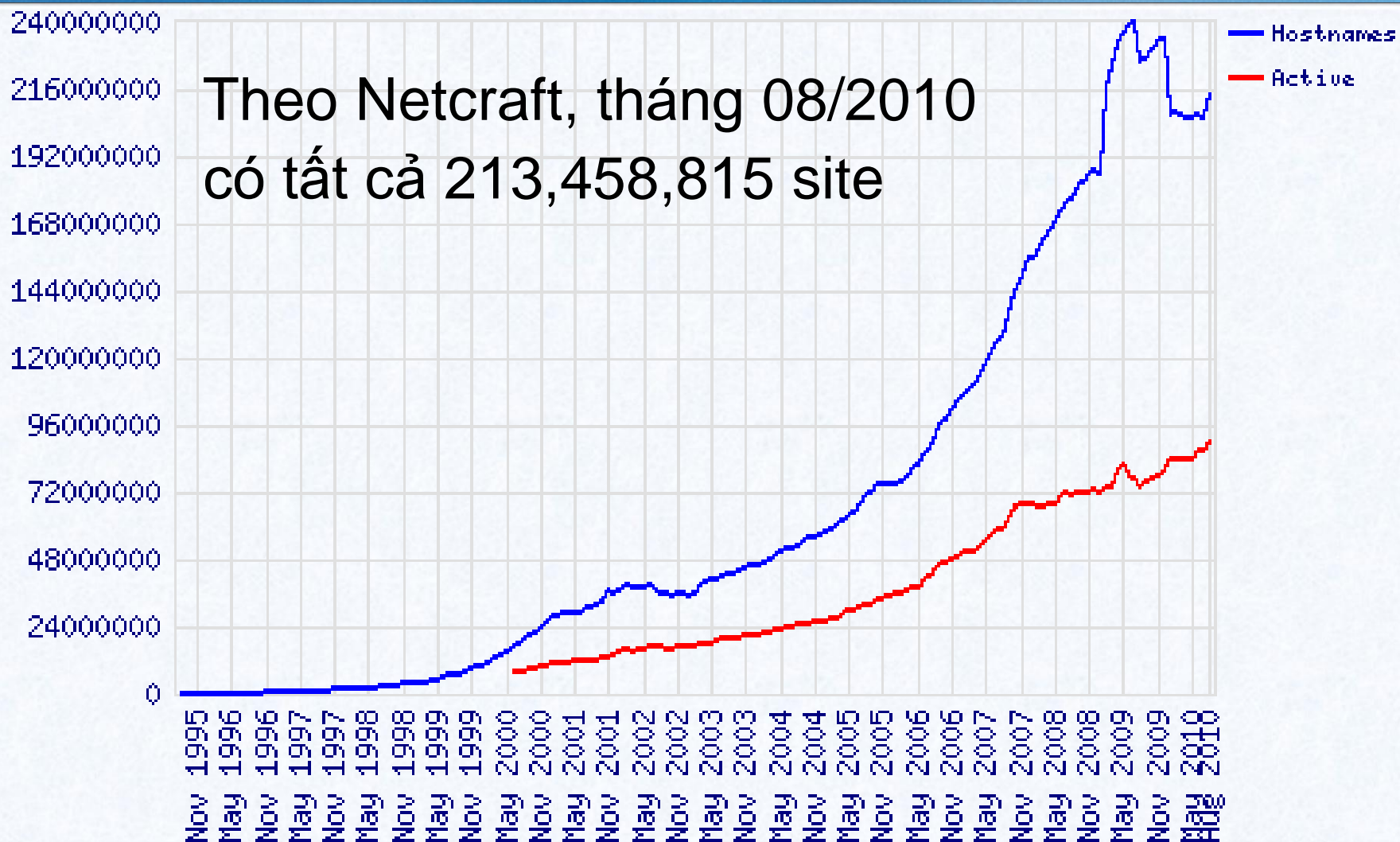


Lịch sử

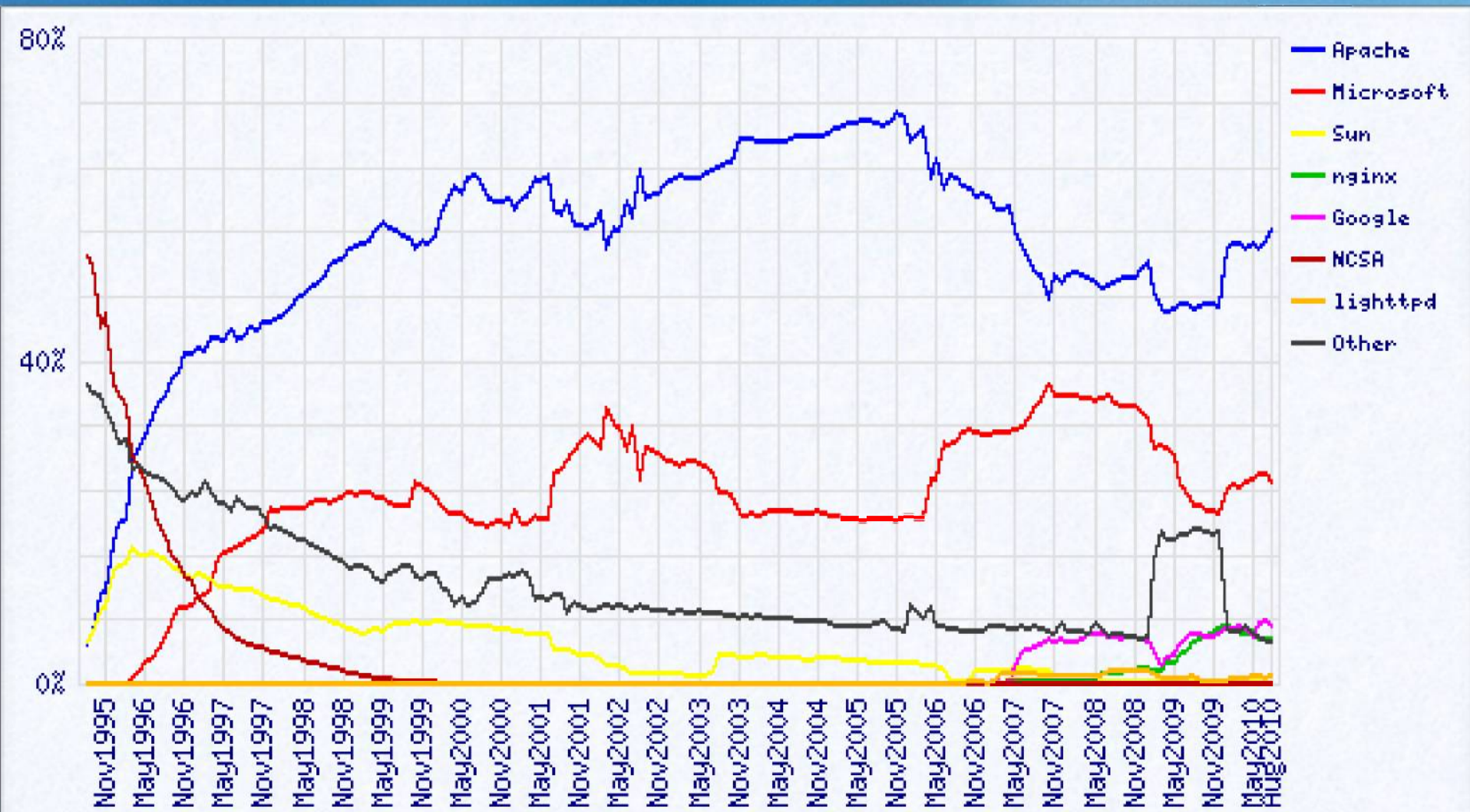
- Mạng Internet xuất phát từ mạng ARPANET của Mỹ (1969) – mạng giữa các trường Đại học
- 1989 - Tim Berners-Lee phát minh ra giao thức World Wide Web, sau đó tạo ra ngôn ngữ Hypertext Markup Language - HTML
- **25/12/1990** – Tim viết trình duyệt (Web Browser) và Web server đầu tiên (CERN HTTPd)
- Từ năm 1993 Internet phát triển rất nhanh
- Việt nam có Internet vào tháng **11/1995**



Lịch sử



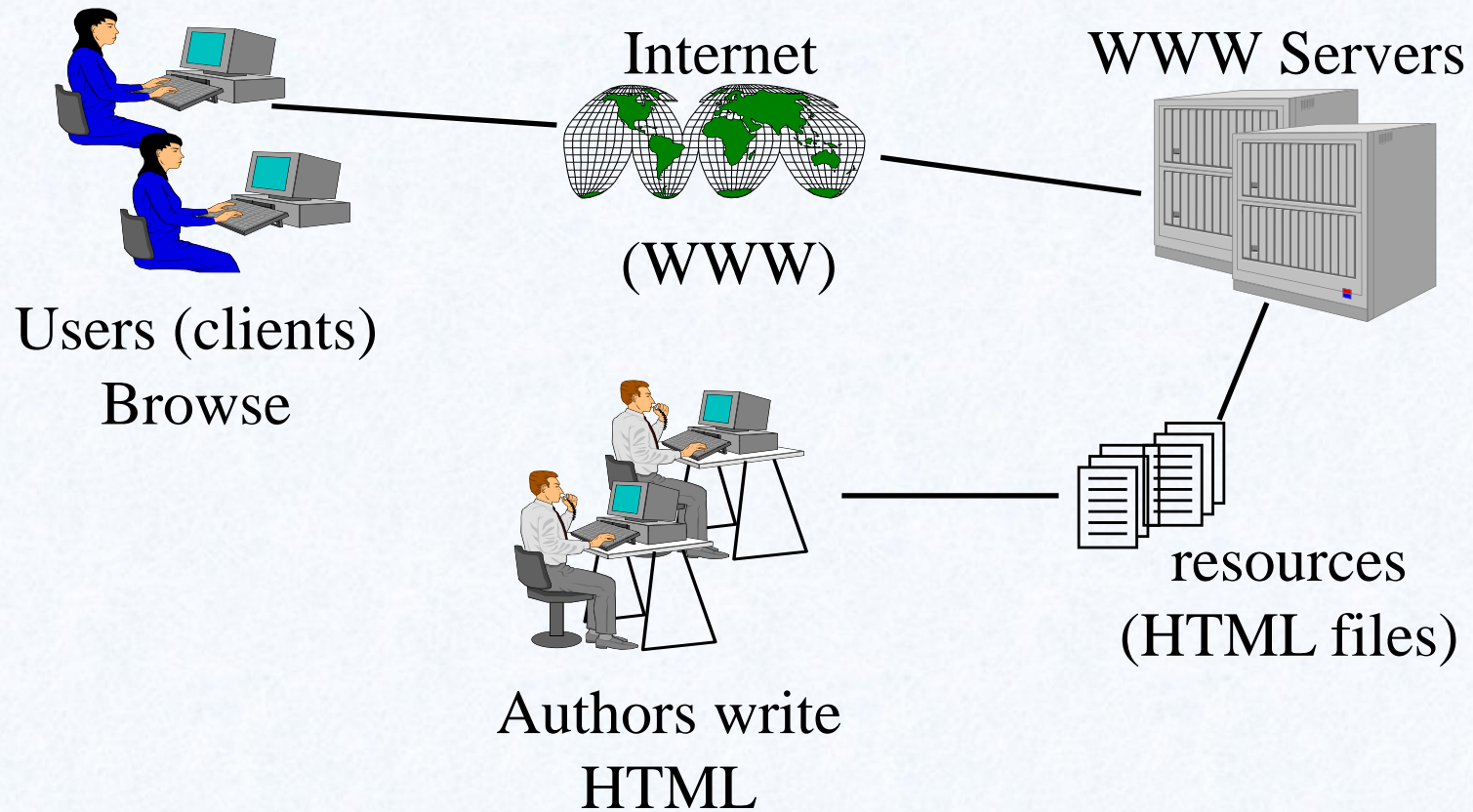
Thị phần Web server



Web là gì?

- Là các dịch vụ **phân tán** cung cấp thông tin **multimedia** dựa trên **hypertext**
 - **phân tán**: thông tin được đặt trên nhiều máy chủ khắp thế giới
 - **multimedia**: thông tin bao gồm text, graphics, sound, video
 - **hypertext**: là kỹ thuật được sử dụng để truy cập thông tin
- Cung cấp truy cập vào các tài nguyên mạng
 - FTP, News, ...

Web hoạt động như thế nào?



Nội dung

- Giới thiệu lịch sử
- Các khái niệm căn bản
- Các dịch vụ cơ bản của Internet
- Khai thác tài nguyên trên Internet
- Các bước thiết lập website

Web – Các khái niệm chính

- **Web** = protocol + language + naming infrastructure
- **HTTP - HyperText Transport Protocol**
 - Là giao thức giao tiếp giữa WWW client and server
- **HTML - HyperText Markup Language**
 - Ngôn ngữ biểu diễn các tài liệu WWW
- **URL - Uniform Resource Locator**
 - Địa chỉ web (xác định duy nhất)

Web - Các khái niệm chính

• Địa chỉ IP – IP Address

- Là 1 con số 32 bit, chia thành 4 số 8 bit, vd: 203.162.33.44 (gồm 2 phần: network address, host address)
- Xác định đối tượng nhận và gửi thông tin trên Internet
- Để biết IP: **ping** www.intel.com

```
C:\>ping www.intel.com

Pinging a961.g.akamai.net [125.56.199.26] with 32 bytes of data:

Reply from 125.56.199.26: bytes=32 time=110ms TTL=50
Reply from 125.56.199.26: bytes=32 time=112ms TTL=50
Reply from 125.56.199.26: bytes=32 time=111ms TTL=50
Reply from 125.56.199.26: bytes=32 time=110ms TTL=50

Ping statistics for 125.56.199.26:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milliseconds:
        Minimum = 110ms, Maximum = 113ms
```

```
C:\>tracert www.intel.com

Tracing route to a961.g.akamai.net [125.56.199.26]
over a maximum of 30 hops:
  0  <1 ms    <1 ms    <1 ms    192.168.1.1
  1  15 ms     14 ms    14 ms    adsl.viettel.vn [115.74.64.1]
  2  11 ms     14 ms    14 ms    203.113.190.1
  3  11 ms     14 ms    14 ms    203.113.158.28
  4  14 ms     14 ms    14 ms    203.113.158.138
  5  77 ms     79 ms    80 ms    pos13-3.cr02.hkg04.pccwbtn.net [63.203.113.158]
  6  75 ms     78 ms    75 ms    TenGE9-2.br01.hkg04.pccwbtn.net [63.203.113.150]
  7  113 ms    110 ms   107 ms   singtel.ge5-13.br01.hkg04.pccwbtn.net [63.203.113.150]
  8  112 ms    110 ms   214 ms   203.208.172.181
  9  111 ms    109 ms   109 ms   ge-4-1-0-0.sngc3-dr1.ix.singtel.com [63.203.113.150]
 10  110 ms    110 ms   113 ms   125.56.199.26
 11  110 ms    110 ms   113 ms   125.56.199.26

Trace complete.
```

Web - Các khái niệm chính

- **Tên miền – Domain name**

- Là tên giao dịch của công ty hay tổ chức trên Internet.
- Ví dụ: www.intel.com :
 - Là địa chỉ của 1 máy chủ thuộc tổ chức (công ty) INTEL
 - Có địa chỉ IP là **125.56.199.27**, có tên máy chủ là WWW
 - Là tên miền cấp 1 (.com, org, .edu, .biz, .net,)

- **Ánh xạ giữa tên miền và địa chỉ IP**

- Do DNS server – Domain name system (service) đảm trách

Các trình duyệt



Internet Explorer



Mozilla Firefox



Google Chrome



Apple Safari



Opera



Netscape



AOL Explorer



Text browser



Avant Browser



Maxthon



Flock



K-Meleon

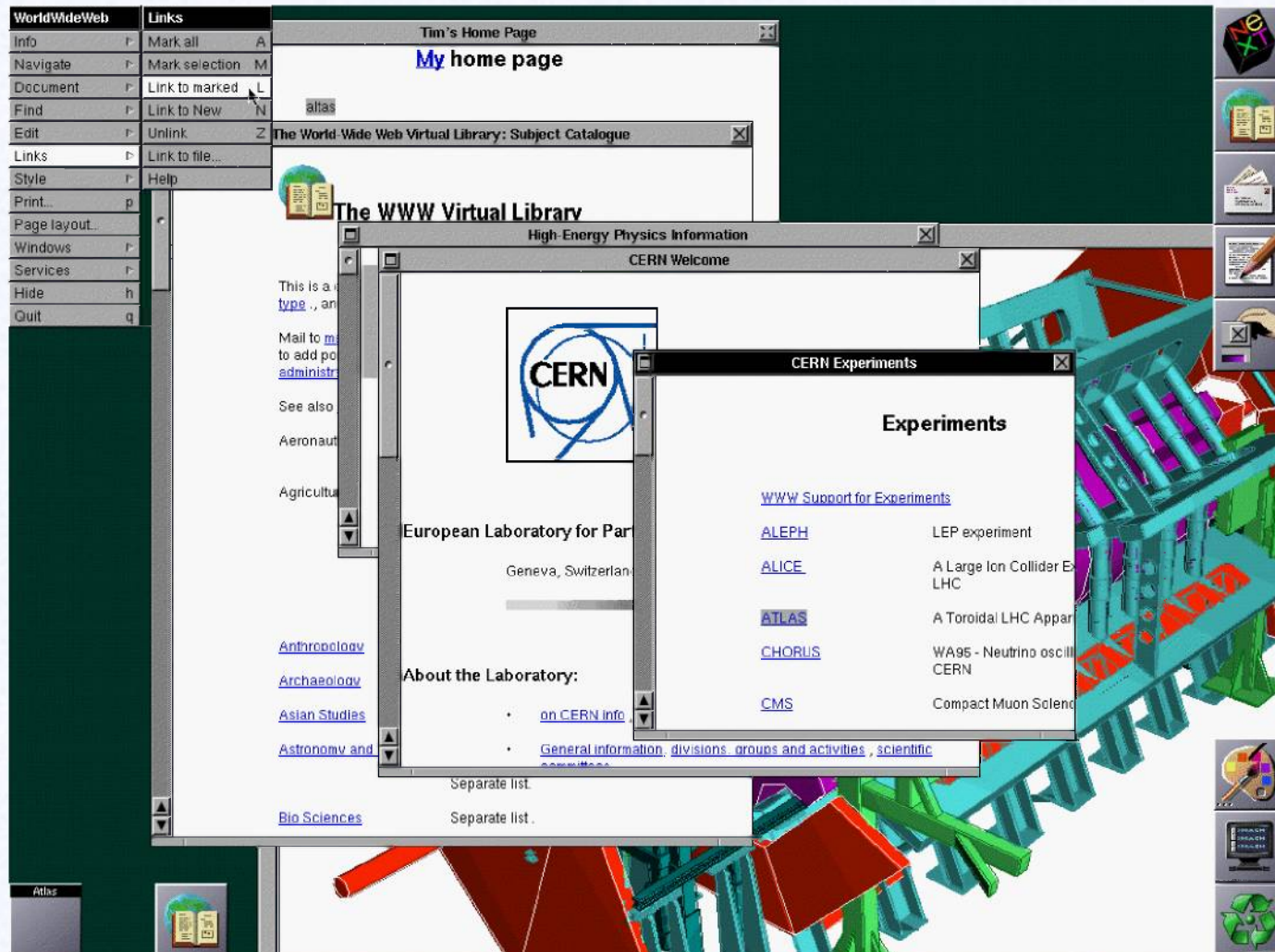


**Epiphany
(Gnome)**



**Konqueror
(KDE 4.2)**

Trình duyệt đầu tiên – WorldWideWeb (Nexus)



Web – Các khái niệm chính Browser (Lynx)

```
CA http://images.google.com.vn/imghp?hl=en&tab=wi
Preferences (p1 of 5)

Web Images News Groups Scholar Gmail more v
Blogs Calendar Photos Documents Reader Sites
even more ↵

Sign in

Google
Preferences Preferences Help : About Google
Save your preferences when finished and return to search.
Save Preferences

Global Preferences <changes apply to all Google services>

[warning]

Your cookies seem to be disabled.
Setting preferences will not work until you enable cookies in your
browser.
How do I enable cookies?

<NORMAL LINK> Use right-arrow or <return> to activate.
Arrow keys: Up and Down to move. Right to follow a link; Left to go back.
H)elp O)ptions P)rint G)o M)ain screen Q)uit /=search [delete]=history list
```

Thị phần web browser

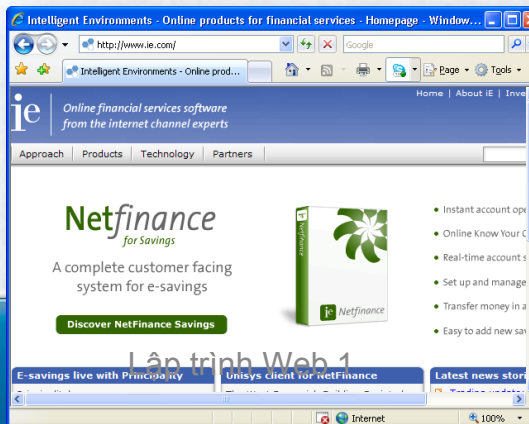


Theo <http://marketshare.hitslink.com/report.aspx?qprid=0>, vào lúc 21:48:01 ngày 16/09/2010



Web browser

- Hiển thị (nếu có thể) các tài nguyên khác nhau
- Khả năng hiển thị :
 - Text-only (Lynx, ...)
 - Graphic (MSIE, Netscape, ...)
- Hiển thị được nhiều loại ảnh
 - TEXT, GIF, JPEG, sound, video, postscript,...
- Hỗ trợ nhiều giao thức
 - HTTP, FTP, SMTP, POP, ...
- Có thể “*plug-in*” các công cụ vào browser để tăng tính năng (3D animation, SWF, ...)



Web – các khái niệm chính

URL – định vị các tài nguyên Internet

- **URL** là định danh duy nhất cho các tài nguyên Internet

- **Chỉ ra:**

- Cách truy cập
- Vị trí tài nguyên

- **Cú pháp chung:**

protocol://host_name[:port_num][/path][/file_name]

http://www.microsoft.com:8080/en/us/default.aspx

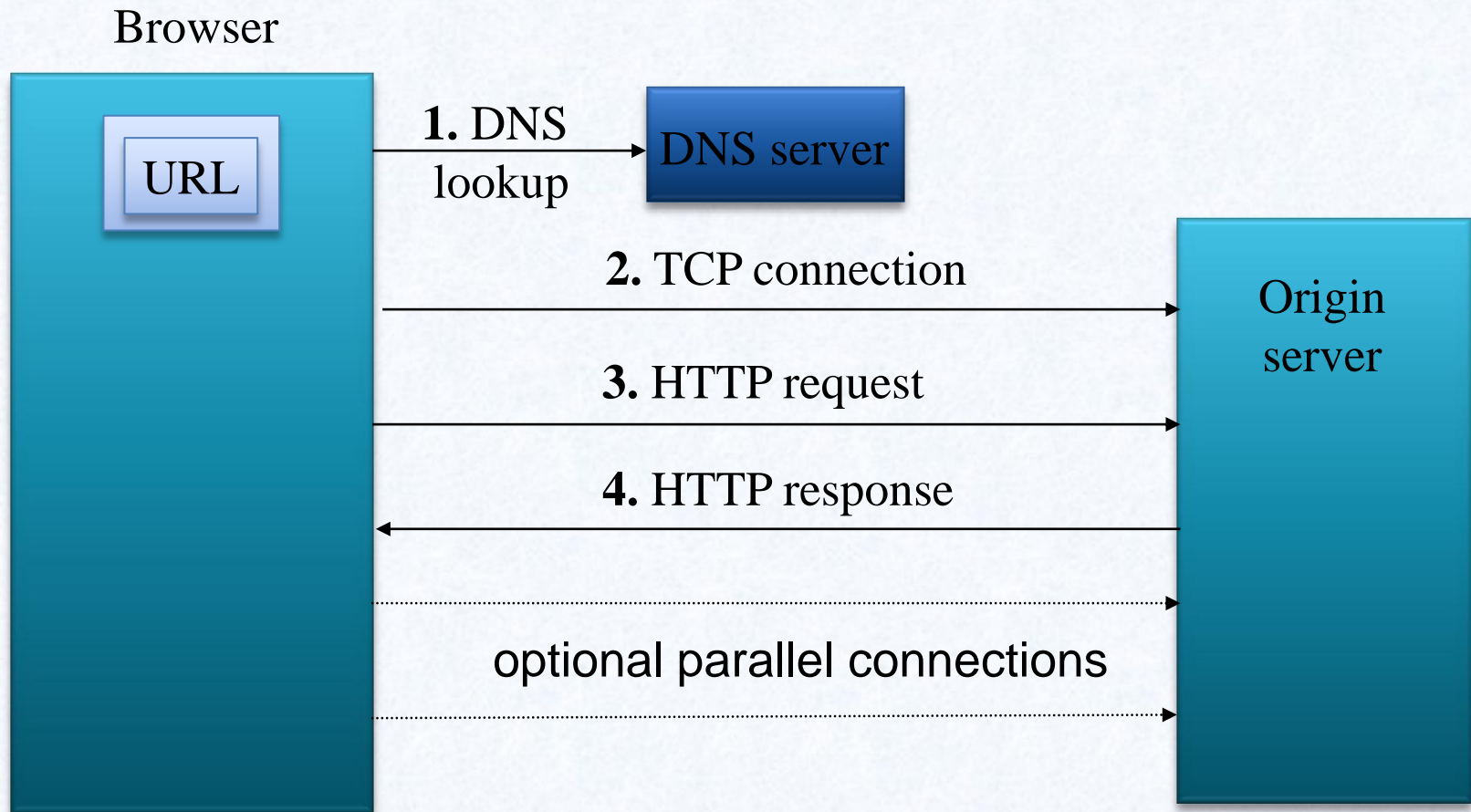
protocol

server name

port

directory/file name on the server

Các giao tác chuẩn trên Web



Nội dung

- Giới thiệu lịch sử
- Các khái niệm căn bản
- Các dịch vụ cơ bản của Internet
- Khai thác tài nguyên trên Internet
- Các bước thiết lập website

Các dịch vụ cơ bản của Internet

World Wide Web – WWW

The screenshot shows the Yahoo! homepage as it appeared in the early 2000s. The browser window is titled "Yahoo! - Windows Internet Explorer" and the address bar shows "http://www.yahoo.com/". The page features the classic "YAHOO!" logo, a search bar, and a navigation menu with categories like "Answers", "Autos", "Finance", "Games", "Groups", "HotJobs", "Maps", "Mobile Web", "Movies", "Music", "Personals", "Real Estate", "Shopping", "Sports", "Tech", "Travel", "TV", and "Yellow Pages". The main content area includes a "Featured" section with a headline "Too close for comfort" about Brad Pitt, a "Check your mail status" section, "Yahoo! Travel" with "Plan Your Getaway" offers, and a "Be a Better Best Friend" section promoting Yahoo! Messenger. The page also displays market information for Dow and Nasdaq, and is sponsored by Scottrade.

The screenshot shows a website for "HOSTING COMPANY" with the tagline "HIGH TECHNOLOGY WEBHOSTING". The page features a blue and white color scheme with a server rack graphic. It lists several hosting plans: "STANDARD PLAN", "10% DISCOUNT PLAN", "CRUISE SAVING PLAN", and "CORPORATE PLAN". Each plan includes details about storage space, bandwidth, and other features. There is also a "CHECK YOUR MAIL STATUS" section and a "BE A BETTER BEST FRIEND" section promoting Yahoo! Messenger. The website footer includes contact information and a copyright notice for 2007.

Các dịch vụ cơ bản của Internet

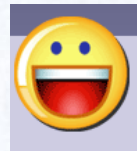
• Thư điện tử – Email (Electronic mail)

- Là dịch vụ trao đổi các thông điệp qua mạng viễn thông
- Sử dụng giao thức SMTP/POP3 để gửi/nhận email
- Địa chỉ email có dạng : name@domainname
VD: *nhkhanh@fit.hcmus.edu.vn* :
 - nhkhanh – tên tài khoản email,
 - fit.hcmus.edu.vn – tên miền

• Truyền, tải tập tin – FTP – File Transfer Protocol

- Là dịch vụ trao đổi các tập tin giữa các máy tính trên Internet

• Tán gẫu – Chat,...



MSN Live



• Mạng xã hội

facebook



Zing me
BETA

Thêm bạn thêm vui

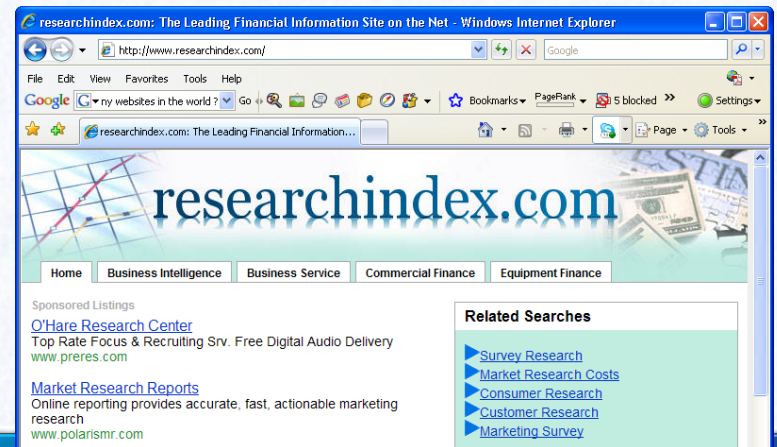
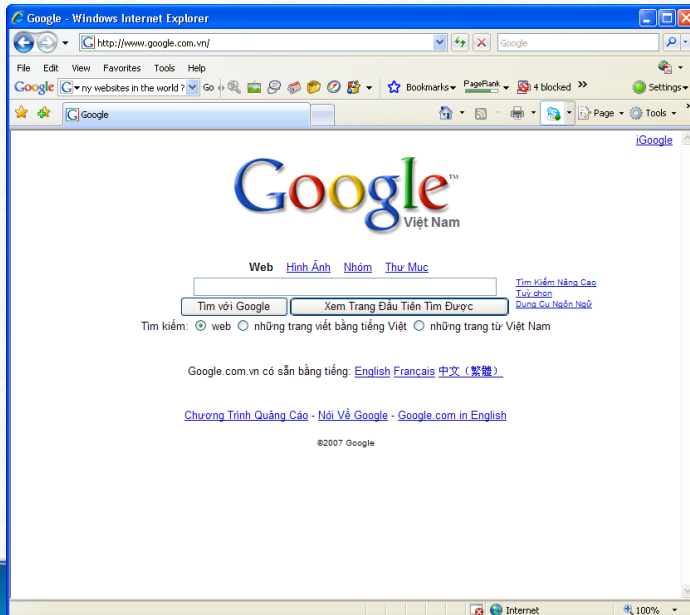
Nội dung

- 1. Giới thiệu lịch sử
- 2. Các khái niệm căn bản
- 3. Các dịch vụ cơ bản của Internet
- 4. Khai thác tài nguyên trên Internet
- 5. Các bước thiết lập website

Khai thác thông tin trên Internet

● Tra cứu thông tin

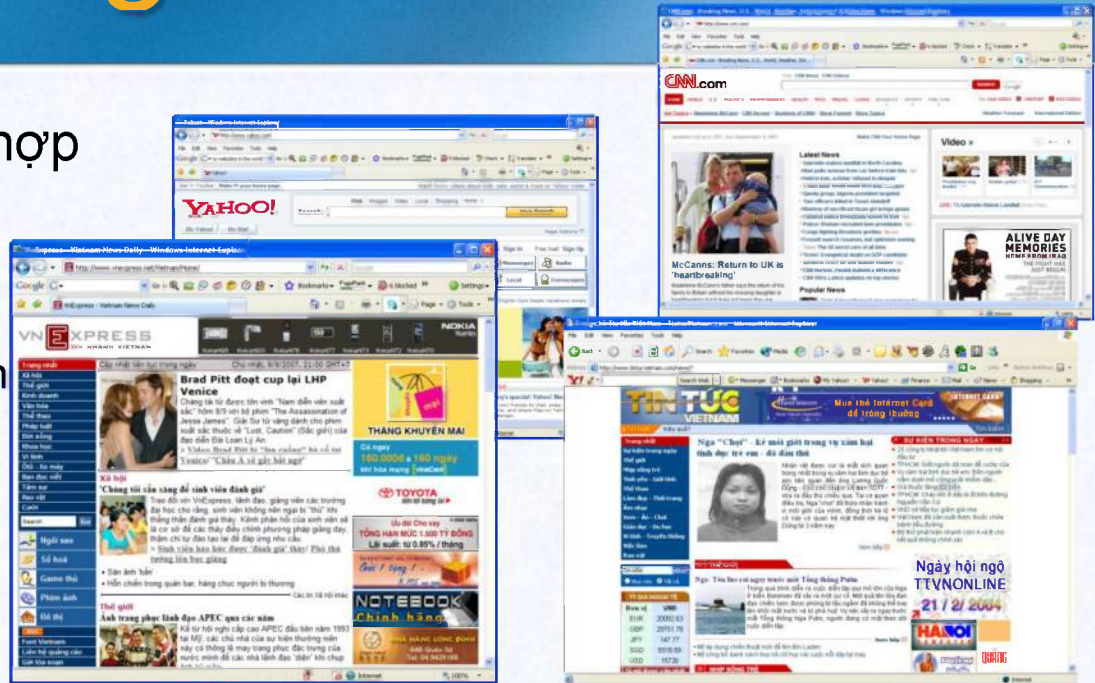
- www.google.com
- www.yahoo.com
- www.lycos.com
- www.altavista.com
- www.researchindex.com (tra cứu bài báo khoa học)



Khai thác thông tin trên Internet

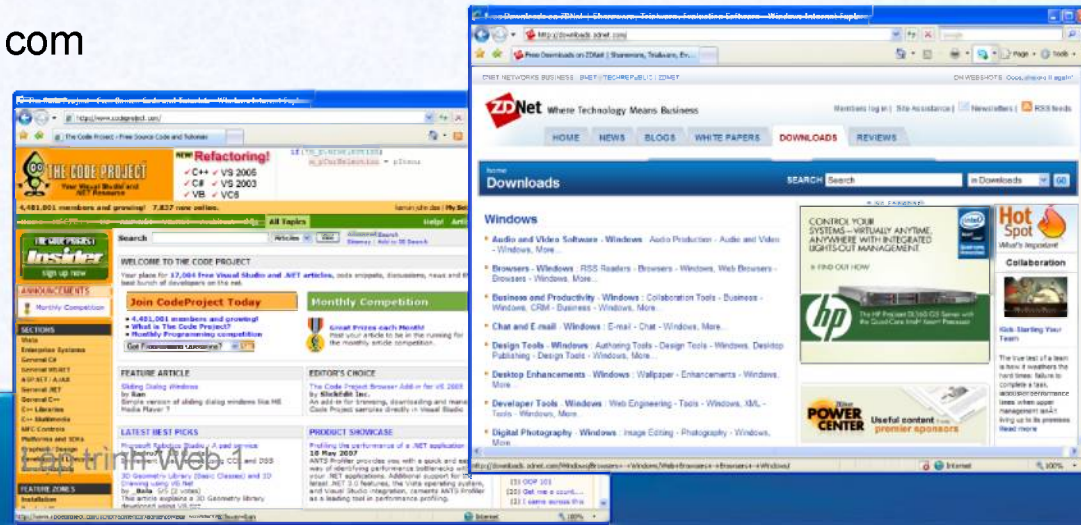
Tin tức, thông tin tổng hợp

- www.yahoo.com
- www.cnn.com
- www.vnexpress.net
- www.tintucvietnam.com
- www.dantri.com



Thư viện phần mềm

- ZDNet – www.zdnet.com/downloads/
- Cnet – www.download.cnet.com



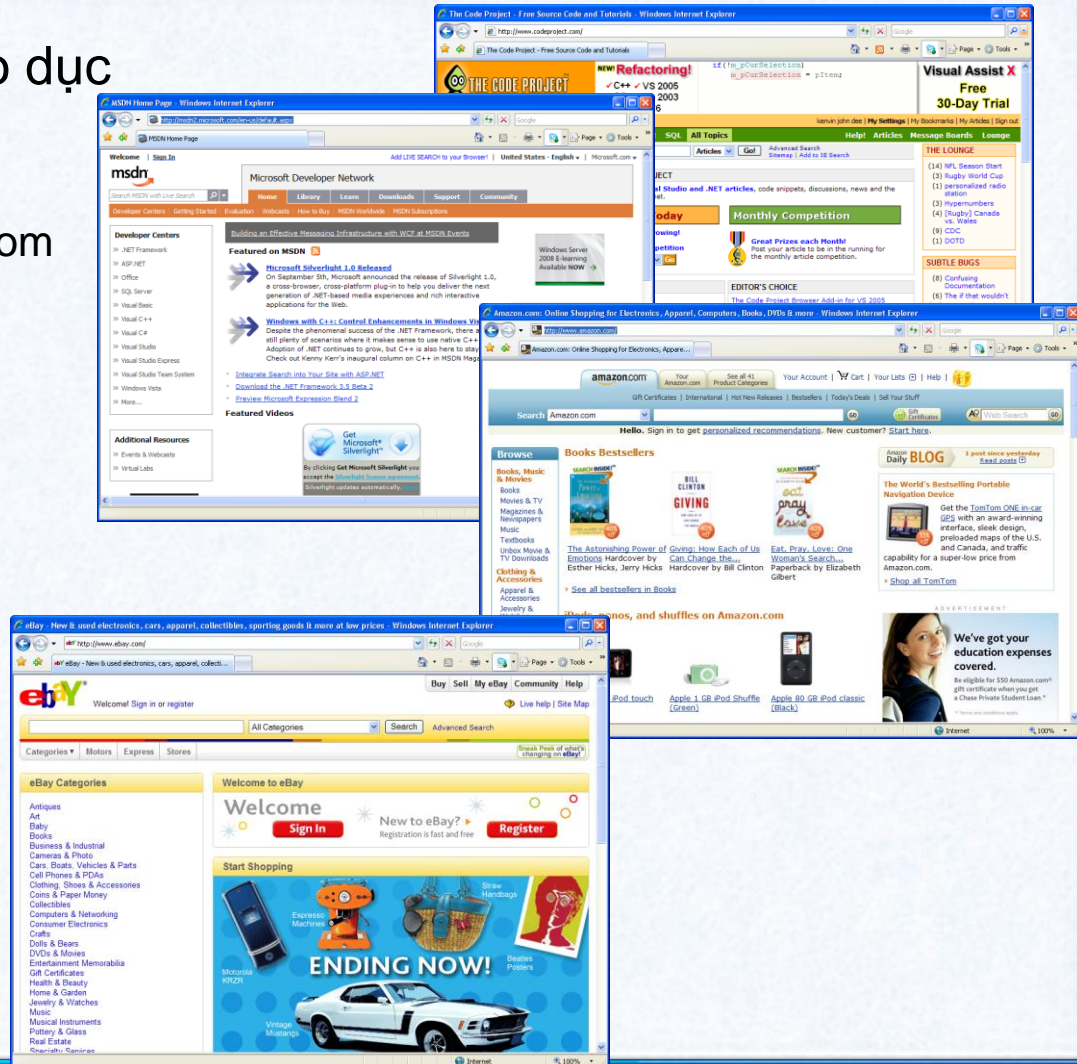
Khai thác thông tin trên Internet

● Nghiên cứu, khoa học, giáo dục

- www.codeproject.com
- msdn.microsoft.com
- www.programmersheaven.com

● Mua bán trực tuyến

- Amazon (mua bán sách):
www.amazon.com
- eBay (đấu giá)
www.ebay.com
- Yahoo shopping
<http://shopping.yahoo.com>



Một số công cụ khai thác tài nguyên Internet

● Download Website Offline

- Cho phép download website về và truy cập offline
- Teleport Pro – <http://www.tenmax.com>
- Offline Explorer Enterprise - <http://www.metaproducts.com>

● Trình download miễn phí

- Orbit Downloader
- Free Download Manager
- FlashGet

● Tự động tải tập tin trên one-click hosting

- JDownloader
- FreeRapid
- Mipony

Một số thuật ngữ thông dụng khác

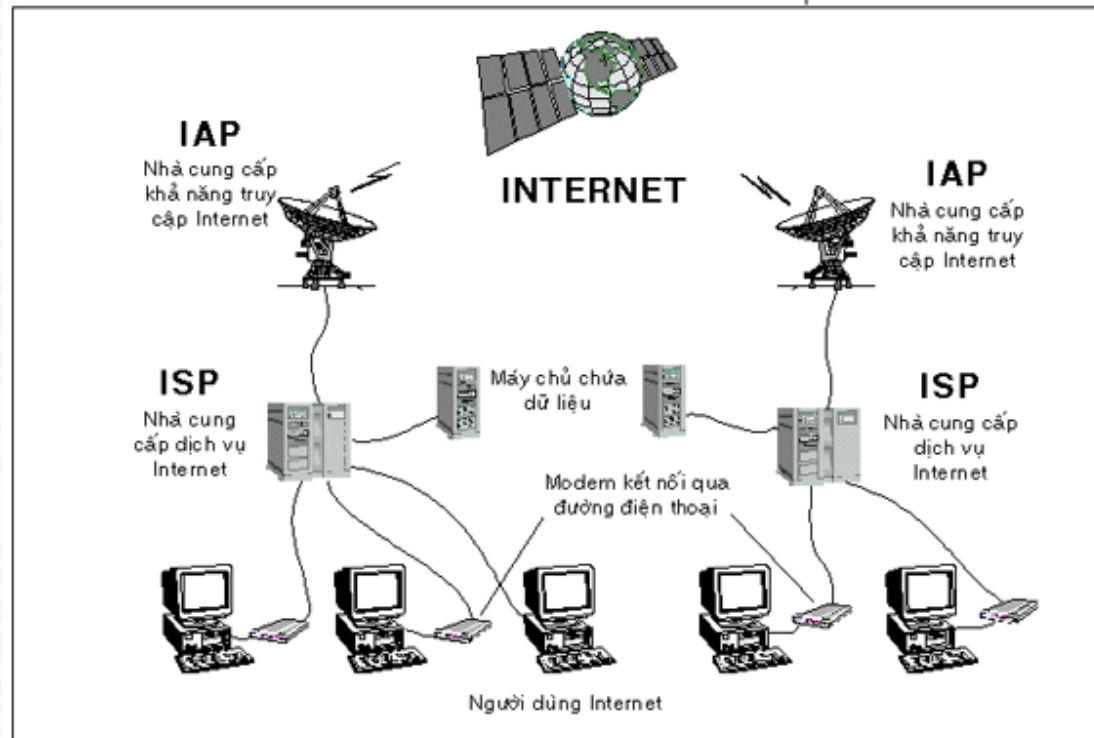
- **Internet:** Mạng máy tính toàn cầu kết nối các mạng máy tính khắp nơi trên thế giới. Tập các giao thức được dùng gọi chung là TCP/IP.
- **Intranet:** Mạng cục bộ có kiến trúc tương tự mạng Internet.
- **Website:** Tập hợp các trang web. Website của các tổ chức hay cá nhân trên mạng bao gồm tập hợp các trang web liên quan đến tổ chức này.
- **Webpage:** Là trang web. Có thể hiển thị các thông tin dưới dạng văn bản, hình ảnh, âm thanh, ...

Một số thuật ngữ thông dụng khác

- **Homepage:** Trang chủ hay còn gọi là trang nhà. Thường là trang đầu tiên (mặc định) khi truy cập một website.
- **Hyperlink:** siêu liên kết. Dùng để liên kết các trang web và dịch vụ của các website trên Internet.
- **Search engines:** Máy tìm kiếm. Các máy tìm kiếm thông dụng hiện nay là Yahoo, Google, Altavista, ...
- **HTTP, FTP, SMTP, POP3,:** Đây là các giao thức được dùng cho các dịch vụ web, ftp, email trên Internet.

Một số thuật ngữ thông dụng khác

- **IAP (Internet Access Provider):** Nhà cung cấp đường truyền Internet.
- **ISP (Internet Service Provider):** Nhà cung cấp dịch vụ Internet. Một số ISP hiện nay ở Việt Nam: VDC, FPT, SPT, NetNam, Viettel, ...



Một số thuật ngữ thông dụng khác

- **Web 1.0:** Thế hệ đầu tiên của WWW, chủ yếu là các trang web tĩnh
- **Web 2.0:** Thế hệ thứ hai của WWW, là sự chuyển dịch từ web tĩnh sang web động với các dữ liệu dễ dàng chia sẻ và mạng xã hội
- **Web 3.0:** Dự kiến thế hệ thứ ba của WWW, liên quan với việc gán ngữ nghĩa cho nội dung của trang web

Nội dung

- Giới thiệu lịch sử
- Các khái niệm căn bản
- Các dịch vụ cơ bản của Internet
- Khai thác tài nguyên trên Internet
- Các bước thiết lập website

Các bước thiết lập Website

- Xác định yêu cầu Website
- Mua tên miền
- Thuê chỗ hosting
- Thiết kế Website
- Đưa vào hoạt động
- Duy trì thông tin, bảo dưỡng website

Một số câu hỏi

- Web tĩnh và động ?

Tổng kết

- Lịch sử, khái niệm Internet, Web
- Các dịch vụ cơ bản trên Internet
- Khai thác các tài nguyên trên Internet

Bài thực hành

THAO TÁC VÀ SỬ DỤNG MẠNG INTERNET

Bài tập thực hành

- Thiết lập nhóm, tạo Google group cho nhóm
- Khảo sát các dịch vụ hosting miễn phí hỗ trợ PHP

Lập trình Web 1



Khoa Công nghệ thông tin
Bộ môn Công nghệ phần mềm

HTML căn bản

Nguyễn Huy Khánh
nhkhanh@fit.hcmus.edu.vn

Mục tiêu



- Hiểu về nguồn gốc và ý nghĩa của HTML
- Hiểu về cấu trúc của tài liệu HTML
- Biết cách tạo một trang HTML với các thẻ HTML cơ bản

Nội dung

Giới thiệu về HTML

Cấu trúc tài liệu HTML

Các thẻ cơ bản

Các thẻ danh sách

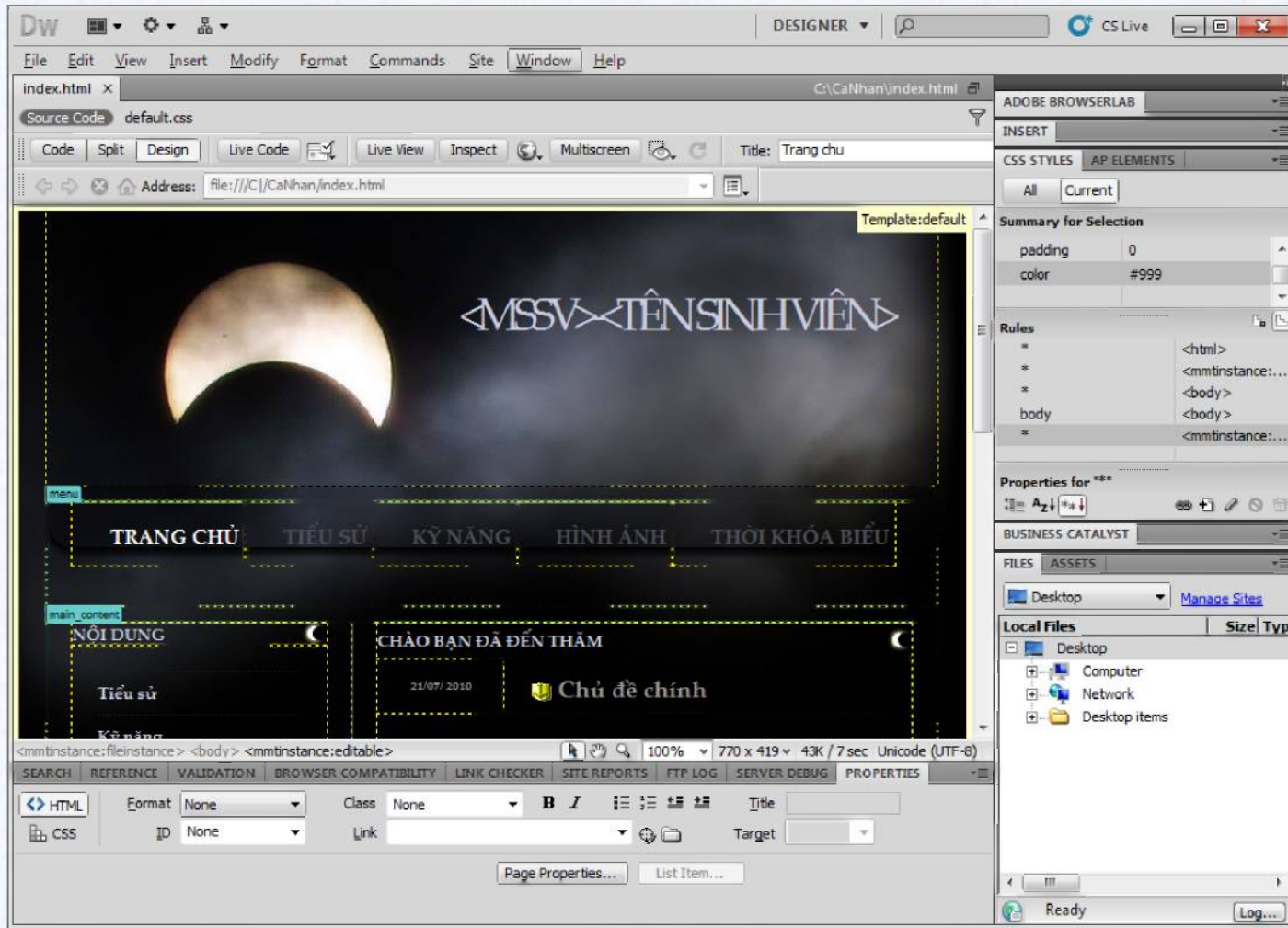
Thẻ liên kết trang

Giới thiệu về HTML

- **HTML** (HyperText Markup Language) - Ngôn ngữ đánh dấu siêu văn bản
- Là ngôn ngữ xây dựng trang Web.
- Chứa các chỉ dẫn cho trình duyệt Web hiển thị một trang Web

- Một trang web gồm có 2 phần chính:
 - Dữ liệu của trang web (văn bản, âm thanh, hình ảnh...)
 - Các thẻ (**thẻ**) HTML dùng để định dạng mô tả cách thức các dữ liệu hiển thị trên trình duyệt

Adobe Dreamweaver CS5



Ví dụ một tài liệu HTML

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title>Welcome to HTML</title>
  <style type="text/css">
    body
    {
      background-color: Orange;
    }
  </style>
</head>
<body>
  <h3>
    Hello World !</h3>
</body>
</html>
```



Hello World !

Phiên bản HTML

- Do tổ chức W3C qui định
- Các chuẩn cơ bản:
 - HTML 4.01
 - XHTML 1.0 Transitional
 - XHTML 1.0 Strict
 - XHTML 1.1
 - HTML 5

Kiểm tra tài liệu HTML có viết đúng chuẩn?

<http://validator.w3.org/>

Thẻ (thẻ) HTML

- <thẻ mở> **Dữ liệu** </thẻ đóng>
- Tên thẻ → luôn mang tính gợi nhớ
 - Ví dụ: **b** ~ Bold, **i** ~ Italic, **p** ~ Paragraph
- Thẻ bao gồm cả thẻ đóng
, <hr/>

** Dòng chữ này được in đậm **

- Cú pháp chung

<thẻ Tên_thuộc_tính='giá_trị'> Dữ liệu </thẻ>

- Ví dụ :

- <div >Lập trình web 1</div>
- <div id="txtDiv" style="color:#0000CC">Thương mại Điện tử 2</div>

[\[Kết quả chạy trên trình duyệt IE\]](#)

Cú pháp

Mã HTML	Hiển thị
<code></code> Đây là một dòng được in Đậm <code></code> <code><h3></code> Mức chữ ở tiêu đề 3 <code></h3></code>	Đây là một dòng được in Đậm Mức chữ ở tiêu đề 3

Lưu ý :

- Giá trị Thuộc tính của Thẻ nên đặt trong nháy kép
- Không phân biệt chữ **HOA** và **thường**
- Bỏ qua các **khoảng trắng thừa và các dấu ngắt dòng, xuống dòng**

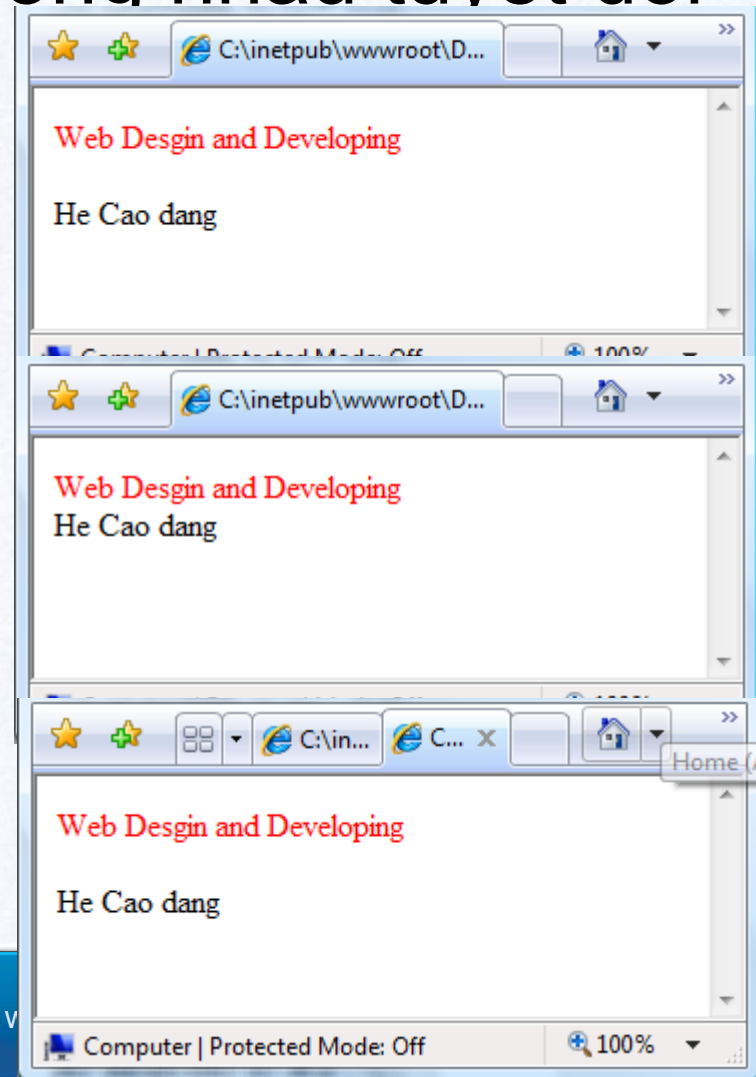
Cú pháp

- Lưu ý: Các thẻ phải lồng nhau tuyệt đối

```
<html>
  <body>
    <font color='red'>
      <p>Web Desgin and Developing</p>
    </font>
    <div> He Cao dang </div>
  </body>
</html>
```


```
<html>
  <body>
    <font color='red'>
      <p>Web Desgin and Developing
    </font>
    <div> He Cao dang </div>
  </p>
</body>
</html>
```

```
<html>
  <body>
    <font color='red'>
      <p>Web Desgin and Developing
    </font>
  </body>
</html>
  <div>
    <p>
      <font color='red'>He Cao dang
    </font>
  </p>
</div>
```



Câu hỏi

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title>Welcome to HTML</title>
  <style type="text/css">
    body
    {
      background-color: Orange;
    }
  </style>
</head>
<body>
  <h3>
    Hello World !</h3>
</body>
</html>
```



Có bao nhiêu
Thẻ HTML ?

Nội dung

Giới thiệu về HTML

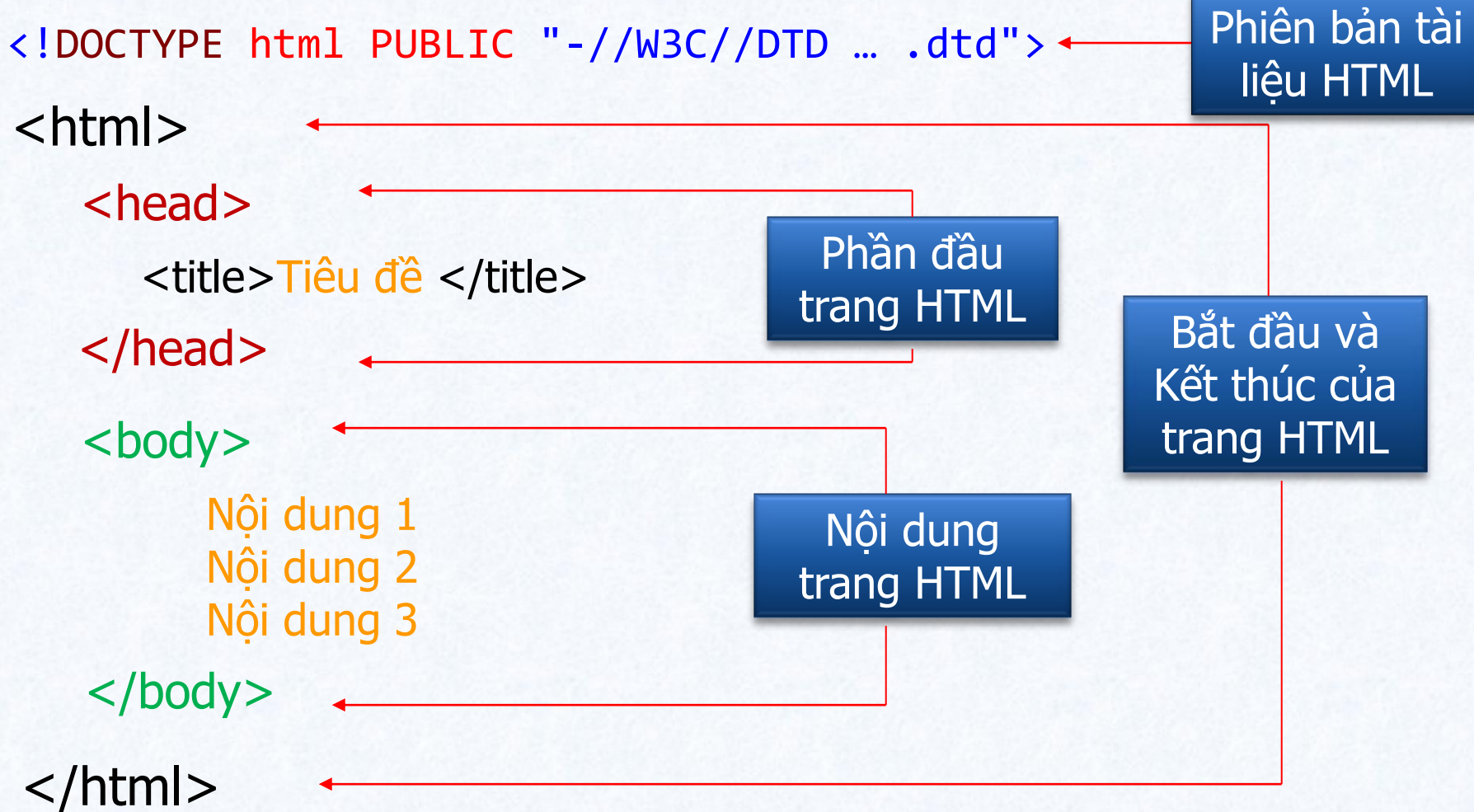
Cấu trúc tài liệu HTML

Các thẻ cơ bản

Các thẻ danh sách

Thẻ liên kết trang

Cấu trúc tài liệu HTML



Cấu trúc tài liệu HTML

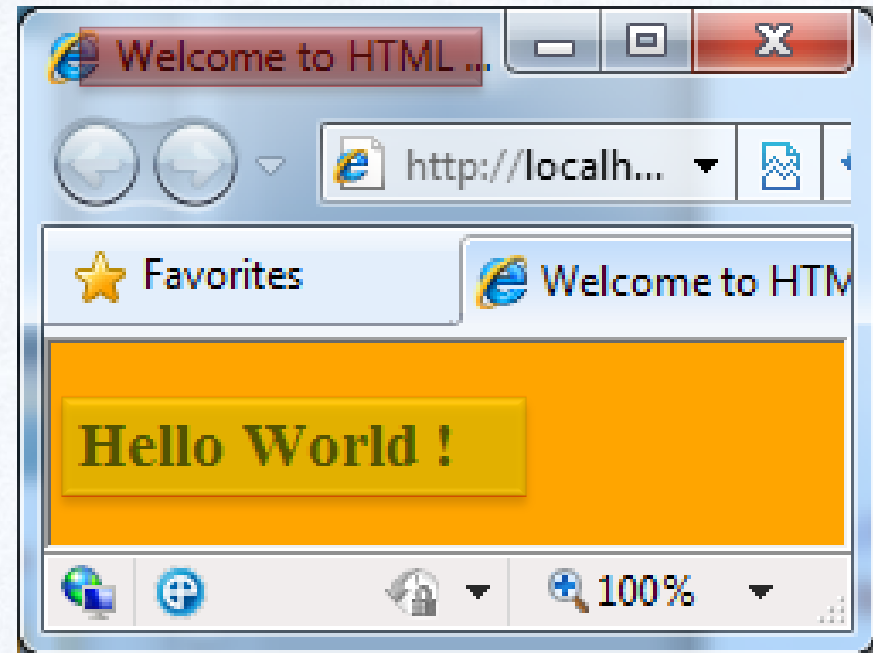
- **DOCTYPE** : Không phải là thẻ HTML, nó giúp trình duyệt biết được phiên bản HTML trang web đang sử dụng
- `<html></html>` : Định nghĩa phạm vi của văn bản HTML
- `<head></head>` : Định nghĩa các mô tả về trang HTML. Thông tin trong thẻ này không được hiển thị trên trang web
- `<title></title>` : Mô tả tiêu đề trang web
- `<body></body>` : Xác định vùng thân của trang web, nơi chứa các thông tin

Ví dụ một tài liệu HTML

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title>Welcome to HTML</title>
  <style type="text/css">
    body
    {
      background-color: Orange;
    }
  </style>
</head>
<body>
  <h3>
    Hello World !</h3>
</body>
</html>
```

Ví dụ một tài liệu HTML

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title>Welcome to HTML</title>
  <style type="text/css">
    body
    {
      background-color: Orange;
    }
  </style>
</head>
<body>
  <h3>
    Hello World !</h3>
</body>
</html>
```



Nội dung

Giới thiệu về HTML

Cấu trúc tài liệu HTML

Các thẻ cơ bản

Các thẻ danh sách

Thẻ liên kết trang

Các thẻ Cơ bản

- Các thẻ xử lý văn bản
- Thẻ hình ảnh
- Thẻ âm thanh

Khối, chuỗi văn bản

- Các thẻ định dạng khối văn bản
 - Tiêu đề (Heading) : `<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>`, `<h6>`
 - Đoạn văn bản (Paragraph): `<p>`
 - Danh sách (List Items): ``
 - Đường kẻ ngang (Horizontal Rules): `<hr />`
- Các thẻ định dạng chuỗi văn bản
 - Định dạng chữ : ``, `<i>`, `` và ``
 - Tạo siêu liên kết : `<a>`
 - Xuống dòng : `
`

Ví dụ về Heading

● HEADING

```
<html>
  <head>
    <title>Introduction to HTML</title>
  </head>

  <body>
    <h1>Introduction to HTML - h1</h1>
    <h2>Introduction to HTML - h2</h2>
    <h3>Introduction to HTML - h3</h3>
    <h4>Introduction to HTML - h4</h4>
    <h5>Introduction to HTML - h5</h5>
    <h6>Introduction to HTML - h6</h6>
  </body>
</html>
```

Ví dụ về Paragraph

● PARAGRAPH - <P>

```
<html>
<head>
  <title>Welcome to HTML</title>
</head>
<body bgcolor='lavender' >
  <h3>My first HTML document</h3>
  <p>
    This is going to be real fun
  <h2>Using another heading</h2>
  </p>
  <p align='center' >Another paragraph
  element</p>
</body>
</html>
```

Ví dụ về Paragraph

- **HORIZONTAL RULES**

- **<HR ...>**

- Thuộc tính :

- align : Canh hàng đường kẻ ngang so với trang web
- width : Chiều dài đường kẻ ngang
- size : Bề rộng của đường kẻ ngang
- noshade : Không có bóng

- **<HR noshade size='5' align='center' width='40%' ></HR>**

- **<HR size='15' align='right' width='80%' ></HR>**

Ví dụ thẻ Định dạng chữ

- Định dạng

``This text is bold``

``This text is strong ``

`<big>`This text is big `</big>`

``This text is emphasized ``

`<i>`This text is italic `</i>`

`<small>`This text is small `</small>`

This text contains a_{2}

This text contains $x^{2} = a \times a$

Ví dụ thẻ Định dạng chữ

- Định dạng

 Renders as emphasized text

 Renders as strong emphasized text

<DFN> Defines a definition term </DFN>

<CODE> Defines computer code text </CODE>

<KBD> Defines keyboard text </KBD>

<VAR> Defines a variable part of a text </VAR>

<CITE> Defines a citation </CITE>

<BLINK>Computer Sciences</BLINK>

Computer Sciences

<INS>Computer Sciences</INS>

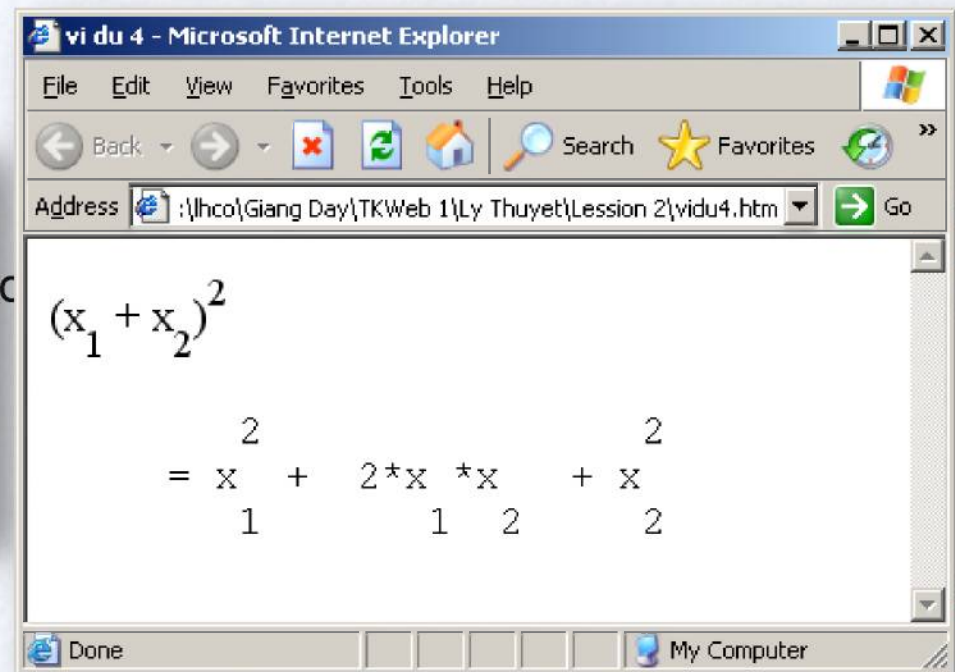
<SAMP> Defines sample computer code </SAMP>

WYSIWYG với thẻ <pre>

- Hiện thị đúng dạng văn bản đã soạn thảo (khoảng trắng, xuống dòng, thẻ,...)

Ví dụ:


```
<html>
<head><title> vi du 4 </title></head>
<body>
(x<sub>1</sub> + x<sub>2</sub>)<sup>2</sup>
<pre>
      2           2
    = x  +  2*x  *x  + x
      1           1 2      2
</pre>
</body>
</html>
```



Ký tự đặc biệt

- Làm sao hiển thị các ký hiệu đặc biệt ?
- Dấu <, >, &
- Dấu nháy kép “
- Các ký tự đặc biệt : @ © ®
-



Thẻ hình ảnh

- `` : Không có thẻ đóng
- Các thuộc tính của thẻ ``:
 - `src` : Đường dẫn đến file hình ảnh
 - `ALT` : Chú thích cho hình ảnh trong trường hợp không có hình / tooltip
 - `Position`: Top, Bottom, Middle
 - `Border` : Độ dày nét viền quanh ảnh (default=0)
- Đặt ảnh nền cho trang web
 - Sử dụng thẻ `<body Background='Image Path'>`

Thẻ hình ảnh (tt)

Thẻ HTML	Ý nghĩa
<code></code>	Định nghĩa một hình ảnh trong tài liệu HTML
<code><map></code>	Định nghĩa một <u>ảnh bản đồ</u> trong tài liệu HTML
<code><area></code>	Định nghĩa vùng click chuột cho ảnh bản đồ

Thẻ âm thanh

- **<bgsound>** : Không có thẻ đóng
- Thuộc tính của thẻ **<bgsound>**
 - src : Đường dẫn đến file âm thanh
 - loop : Số lần lặp (bằng -1 : Lặp vô hạn)
 - **<bgsound>** Thường đặt trong thẻ **<head>** của trang web.
- Ví dụ: `<bgsound src='batman.mid' loop='1'>`
 - [Ví dụ - Âm thanh](#)

Xử lý âm thanh nền (tt)

● Cách sử dụng thẻ OBJECT

```
<object data="Graduation.mp3" type="application/x-  
mplayer2" width="0" height="0">  
  <param name="filename" value="Graduation.mp3">  
  <param name="playcount" value="true">  
  <param name="autostart" value="true">  
</object>
```

- Chạy được trên nhiều trình duyệt IE, Fire Fox, Chrome, Safari...

Xử lý âm thanh nền (tt)

● Cách thông thường

```
<body>
```

```
  <bgsound src="Graduation.mp3 " loop="-1" />
```

```
</body>
```

● Khuyết điểm

- Chỉ chạy được trên IE
- Các trình duyệt khác không hiểu **<bgsound>**

Nội dung

Giới thiệu về HTML

Cấu trúc tài liệu HTML

Các thẻ cơ bản

Các thẻ danh sách

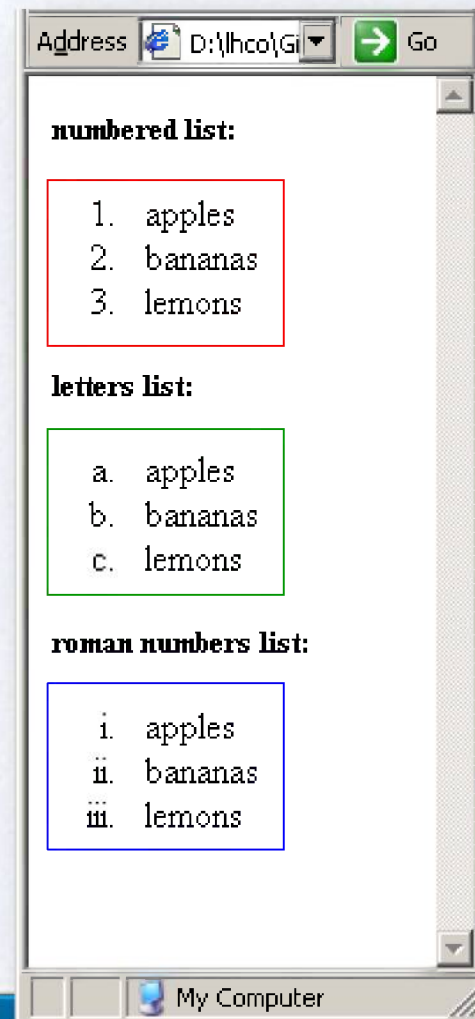
Thẻ liên kết trang

Các thẻ Danh sách

Kiểu danh sách	Thẻ	Phần tử trong DS
Danh sách có thứ tự	<code></code>	<code></code>
Danh sách không có thứ tự	<code></code>	<code></code>
Danh sách tự định nghĩa	<code><dl></code>	<code><dt></code> , <code><dd></code>
Danh sách lồng nhau		
Dạng khác	<code><menu></code> <code><dir></code>	<code></code>

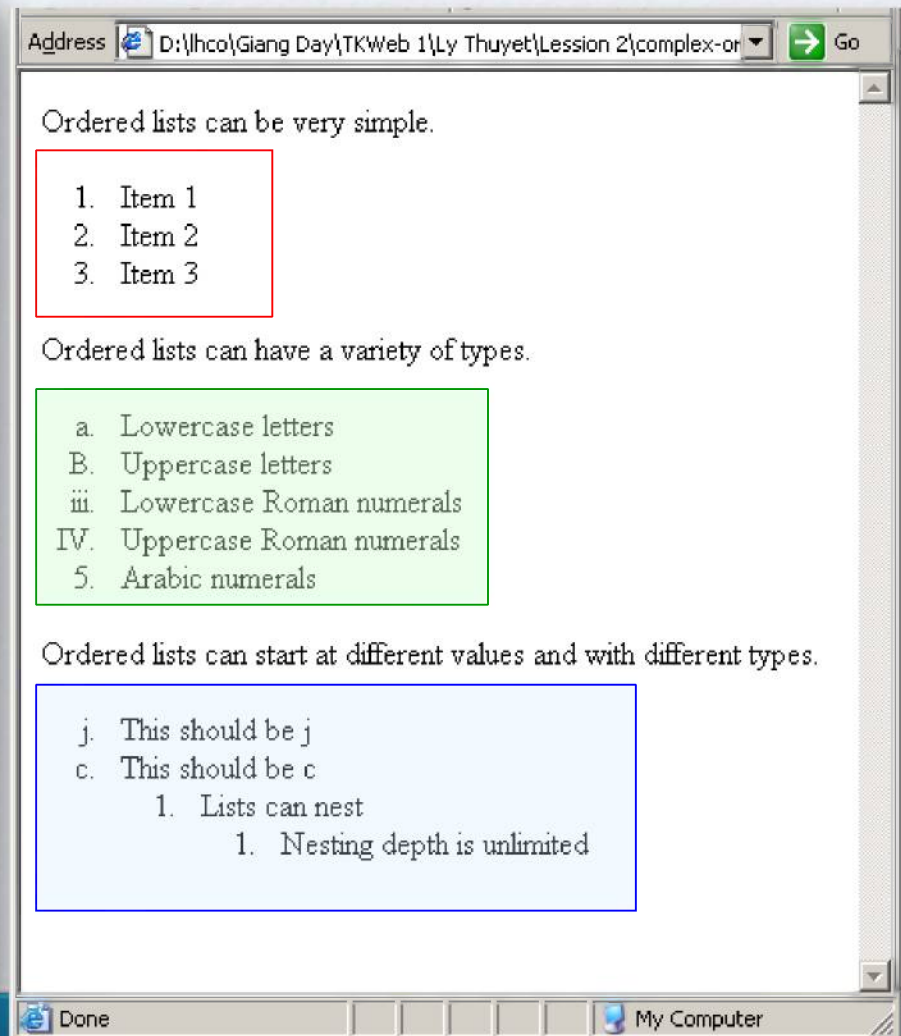
Danh sách có thứ tự

```
1 <html>
2 <head>
3 <title> danh sach co thu tu </title></head>
4 <body>
5 <h5>numbered list:</h5>
6 <ol>
7   <li>apples</li>
8   <li>bananas</li>
9   <li>lemons</li>
10 </ol>
11 <h5>letters list:</h5>
12 <ol type="a">
13   <li>apples</li>
14   <li>bananas</li>
15   <li>lemons</li>
16 </ol>
17 <h5>roman numbers list:</h5>
18 <ol type="i">
19   <li>apples</li>
20   <li>bananas</li>
21   <li>lemons</li>
22 </ol>
23 </body>
24 </html>
25
```



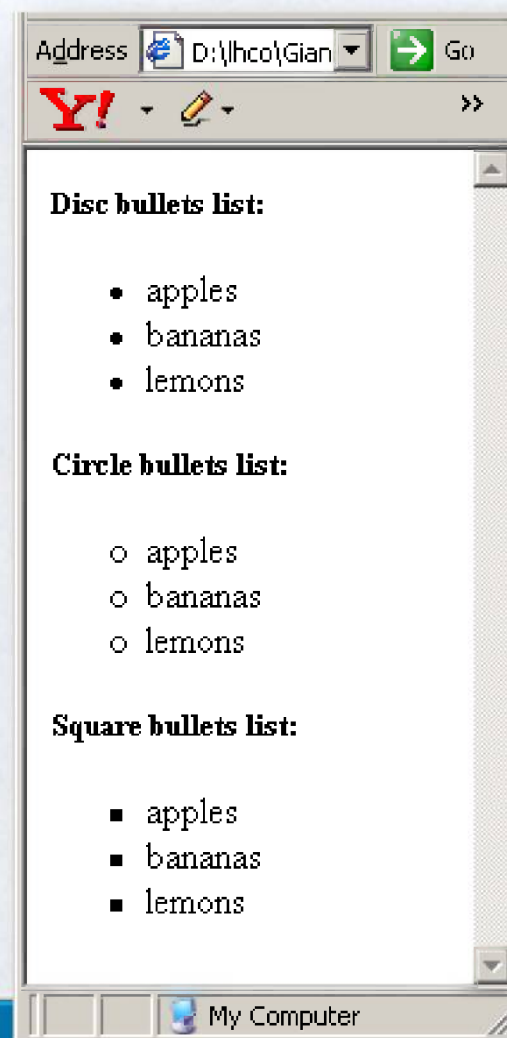
Danh sách có thứ tự

```
1 <html>
2 <head><title>Ordered List Example</title></head>
3 <body>
4 <p>Ordered lists can be very simple.</p>
5 <ol>
6   <li>Item 1</li>
7   <li>Item 2</li>
8   <li>Item 3</li>
9 </ol>
10 <p>Ordered lists can have a variety of types.</p>
11 <ol>
12   <li type="a">Lowercase letters</li>
13   <li type="A">Uppercase letters</li>
14   <li type="i">Lowercase Roman numerals</li>
15   <li type="I">Uppercase Roman numerals</li>
16   <li type="1">Arabic numerals</li>
17 </ol>
18 <p>Ordered lists can start at different values
19 and with different types.</p>
20 <ol start="10" type="a">
21 <li>This should be j</li>
22 <li value="3">This should be c
23   <ol>
24     <li>Lists can nest
25       <ol>
26         <li>Nesting depth is unlimited</li>
27       </ol>
28     </li>
29   </ol>
30 </li>
31 </ol>
32 </body>
33 </html>
```



Danh sách không có thứ tự

```
1 <html>
2 <head>
3 <title> danh sach khong thu tu </title></head>
4 <body>
5 <h5>Disc bullets list:</h5>
6 <ul type="disc">
7   <li>apples</li>
8   <li>bananas</li>
9   <li>lemons</li>
10 </ul>
11 <h5>Circle bullets list:</h5>
12 <ul type="circle">
13   <li>apples</li>
14   <li>bananas</li>
15   <li>lemons</li>
16 </ul>
17 <h5>Square bullets list:</h5>
18 <ul type="square">
19   <li>apples</li>
20   <li>bananas</li>
21   <li>lemons</li>
22 </ul>
23 </body>
24 </html>
25
```

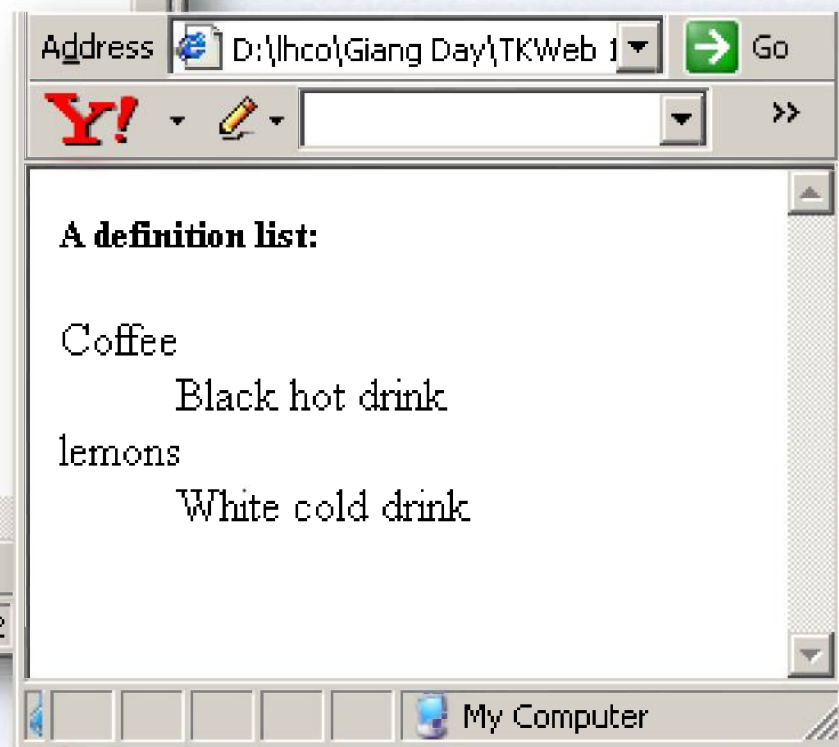


Danh sách tự định nghĩa

```
1 <html>
2 <head><title> danh sach dinh nghia
  </title></head>
3 <body>
4 <h5>A definition list:</h5>
5 <dl>
6     <dt>Coffee</dt>
7     <dd>Black hot drink</dd>
8     <dt>lemons</dt>
9     <dd>White cold drink</dd>
10 </dl>
11 </body>
12 </html>
13
14
```

complex-unorderlis defList.htm

For Help, press F1 In 2



Nội dung

Giới thiệu về HTML

Cấu trúc tài liệu HTML

Các thẻ cơ bản

Các thẻ danh sách

Thẻ liên kết trang

URL

- Các dạng địa chỉ HTTP: 7 trường hợp

TH1: <http://server/>

Ví dụ: <http://www.yahoo.com/>

TH2: <http://server/file>

Ví dụ: <http://games.yahoo.com/index.php>

TH3: <http://server/directory/>

Ví dụ: <http://games.yahoo.com/games/>

TH4: <http://server/directory/file>

Ví dụ: <http://games.yahoo.com/download/Dominoes.exe>

URL

TH5: <http://server/directory/file#marker>

Ví dụ : [http://games.yahoo.com/index.php# Puzzle](http://games.yahoo.com/index.php#Puzzle)

TH6: <http://server/directory/file?parameters>

Ví dụ: <http://www.google.com.vn/search?hl=vi&q=Teach&meta=1>

TH7: <http://server:port/directory/file>

Ví dụ : <http://www.microsoft.com:8080/products/greetings.html>

Thẻ <a> (anchor)

● Cú pháp :

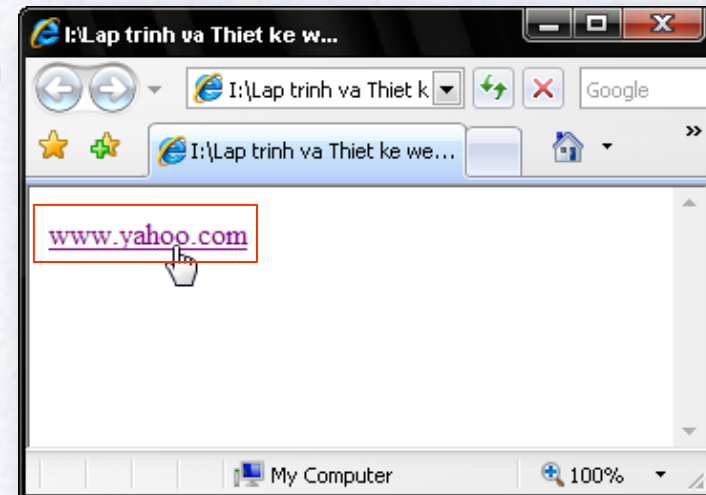
```
<a href="URL" target="..."> Linked content </a>
```

● Thuộc tính target của thẻ <a>

- name: tải trang web vào frame có tên NAME
- _blank: tải trang web vào cửa sổ mới
- _parent: tải trang web vào cửa sổ cha của nó
- _self: tải trang web vào chính cửa sổ hiện hành
- _top: tải trang web vào cửa sổ cao nhất

● Ví dụ :

```
<html>  
  <body>  
    <a href='http://www.yahoo.com'>www.yahoo.com </a>  
  </body>  
</html>
```

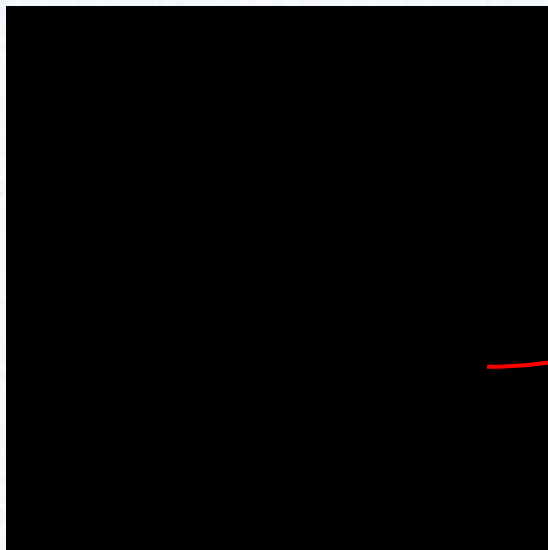


Phân loại liên kết

- Phân loại :
 - Liên kết ngoại (external link)
 - Liên kết nội (internal link)
 - Liên kết email (email link)

Liên kết ngoại (External Link)

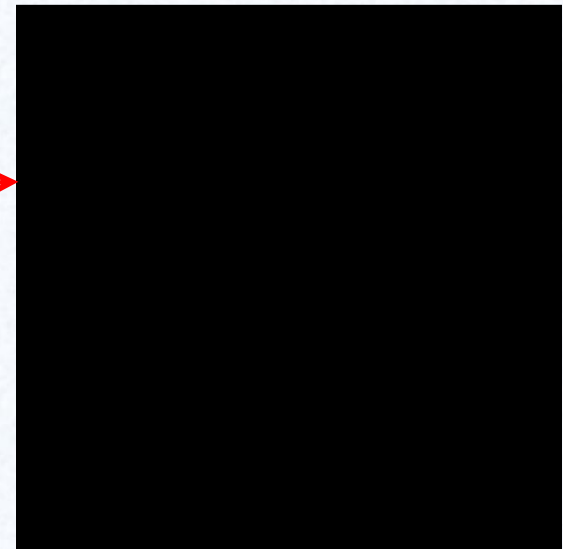
```
<a href="URL"> Text đại diện </a>
```



Trang hiện tại
baihoc1.htm



Click
chuột



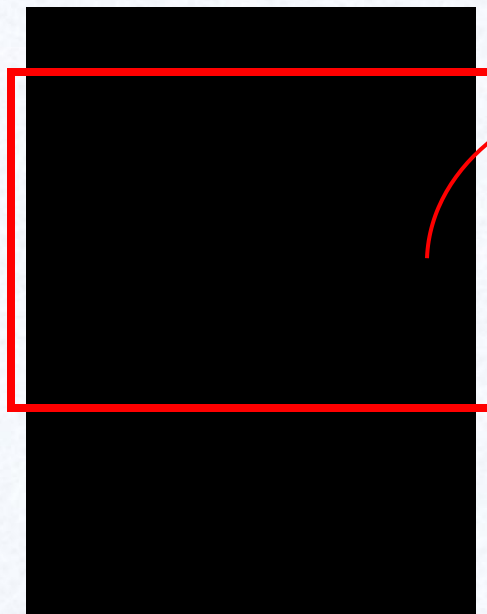
Trang có địa chỉ
xác định từ URL
baihoc2.htm

Liên kết nội (Internal Link)

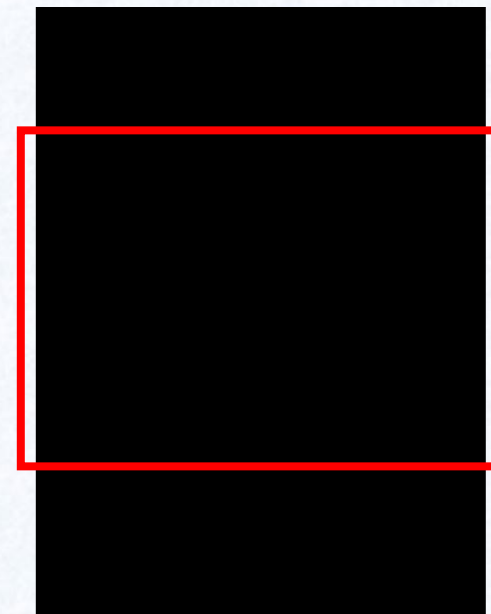
```
<a name="TenViTri"> Vi tri bat dau  
</a>
```

```
<a href="#TenViTri"> Text đại diện  
</a>
```

Nội
dung
trang
khi
chưa
liên
kết

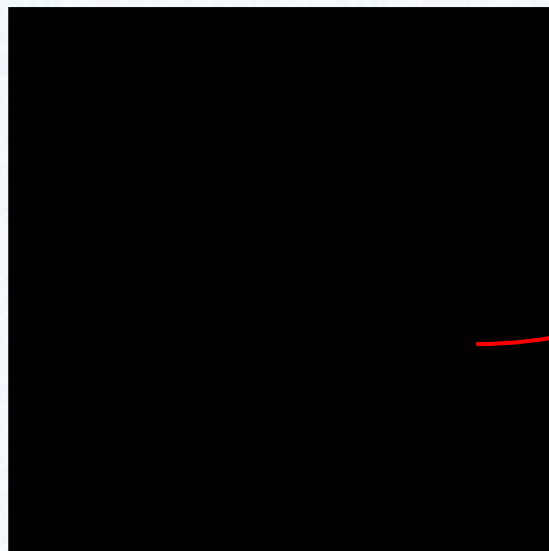


Click
chuột

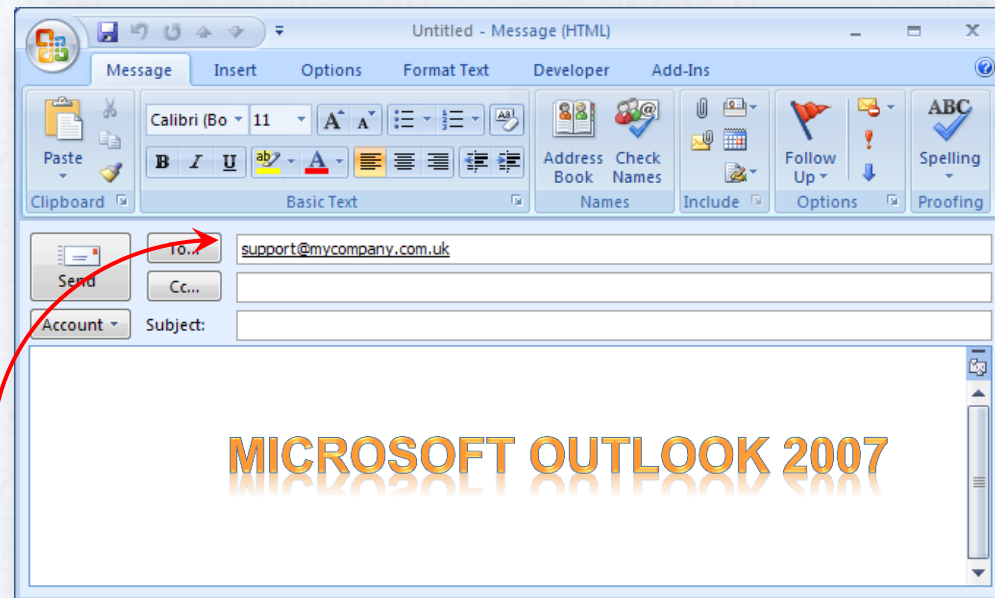


Nội
dung
trang
khi
bấm
liên
kết

Liên kết Email



Click
chuột



```
<a href="mailto:emailAddress">Liên hệ Admin</a>
```

Phân loại địa chỉ URL

- Địa chỉ URL phân làm 2 loại :
 - Địa chỉ tuyệt đối : Là vị trí tuyệt đối so với Mạng Internet
 - Địa chỉ tương đối : Là vị trí tương đối so với trang web hiện hành đang chú liên kết.

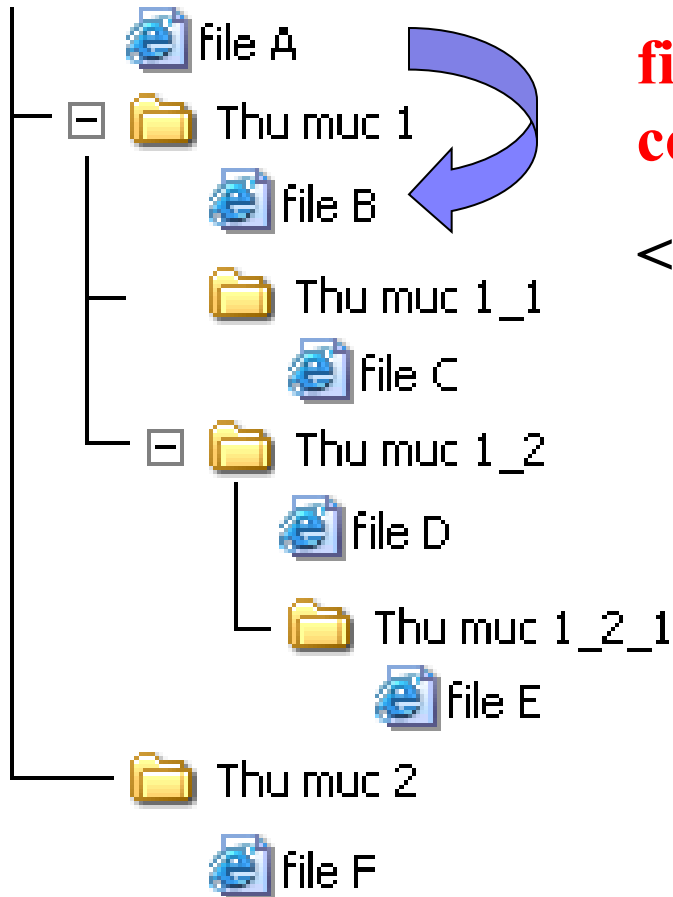
```
<a href="URL" target="....."> Linked content </a>
```

- Một số ký hiệu đường dẫn đặc biệt:

Ký hiệu	Ý nghĩa
/	Trở về thư mục gốc của website
./	Thư mục hiện tại của trang web sử dụng link (mặc định)
../	Quay ra thư mục cha / đi ngược lại 1 cấp thư mục

Ví dụ

[-] folder root # 127.0.0.1/demo



file A có link đến file B, vậy trong file A có HTML element:

`liên kết đến B`

URL =

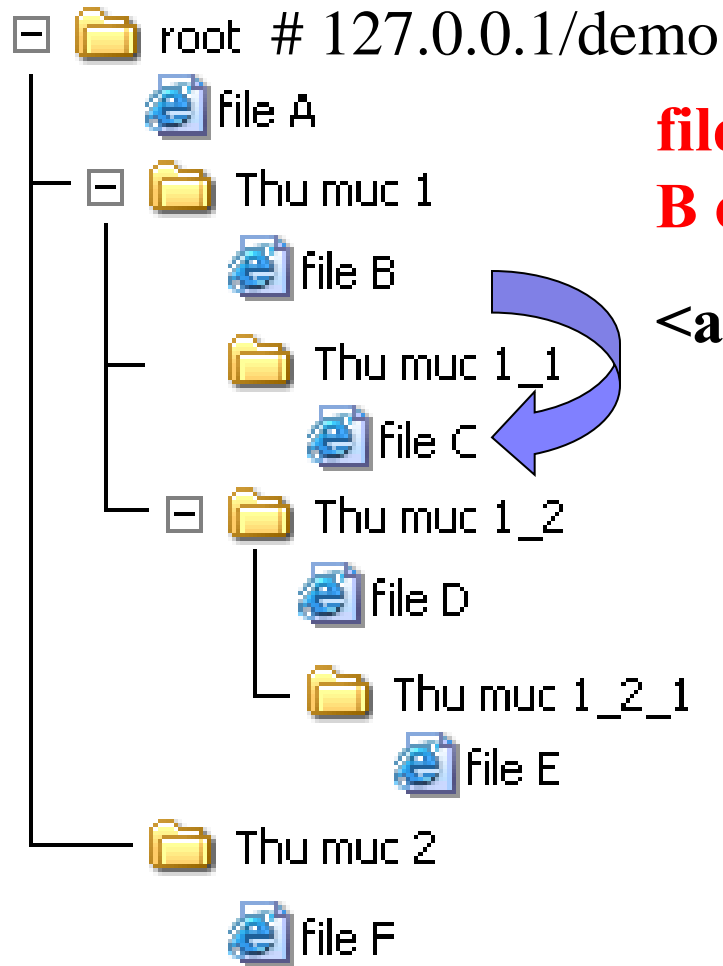
<http://127.0.0.1/demo/Thu muc 1/file B.htm>

</demo/Thu muc 1/file B.htm>

<./Thu muc 1/file B.htm>

<Thu muc 1/file B.htm>

Ví dụ



file B có link đến file C, vậy trong file B có HTML element:

`liên kết đến C`

URL =

<http://127.0.0.1/demo/Thu muc 1/>

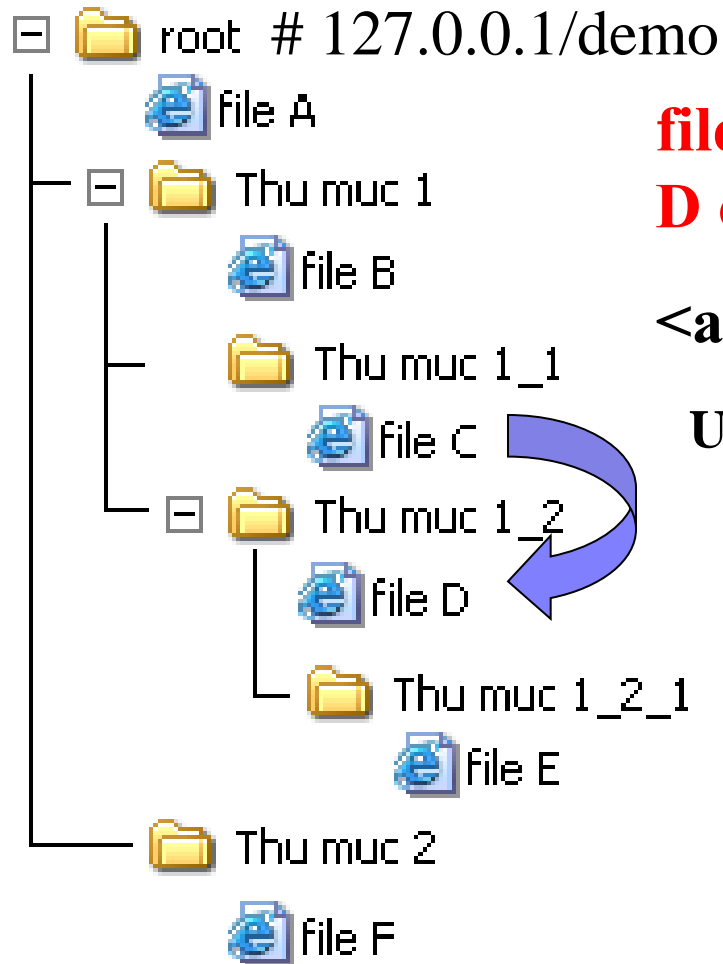
Thu muc 1_1/file C.htm

/demo/Thu muc 1/Thu muc 1_1/file C.htm

./Thu muc 1_1/file C.htm

Thu muc 1_1/file C.htm

Ví dụ



file C có link đến file D, vậy trong file D có HTML element:

`liên kết đến D`

URL =

<http://127.0.0.1/demo/Thu muc 1/>

Thu muc 1_2/file D.htm

/demo/Thu muc 1/Thu muc 1_2/file D.htm

../Thu muc 1_2/file D.htm

../Thu muc 1_2/file D.htm

Ví dụ

[-] folder root # 127.0.0.1/demo

file A

[-] folder Thu muc 1

file B

folder Thu muc 1_1

file C

[-] folder Thu muc 1_2

file D

folder Thu muc 1_2_1

file E

folder Thu muc 2

file F

file D có link đến file F, vậy trong file F có HTML element:

`liên kết đến F`

URL =

<http://127.0.0.1/demo/Thu muc 2/file F.htm>

</demo/Thu muc 2/file F.htm>

<../../Thu muc 2/file F.htm>

<../../Thu muc 2/file F.htm>

Ví dụ

[-] folder root # 127.0.0.1/demo

file A

[-] folder Thu muc 1

file B

folder Thu muc 1_1

file C

[-] folder Thu muc 1_2

file D

folder Thu muc 1_2_1

file E

folder Thu muc 2

file F

file F có link đến file E, vậy trong file F có HTML element:

`liên kết đến E`

URL =

http://127.0.0.1/demo/Thu muc 1/ Thu muc 1_2/Thu muc 1_2_1/file E.htm
[/demo/Thu muc 1/Thu muc 1_2/ Thu muc 1_2_1/file E.htm](http://127.0.0.1/demo/Thu muc 1/Thu muc 1_2/ Thu muc 1_2_1/file E.htm)
[../Thu muc 1 /Thu muc 1_2/Thu muc1_2_1/ file E.htm](http://127.0.0.1/Thu muc 1 /Thu muc 1_2/Thu muc1_2_1/ file E.htm)

Ví dụ

[-] folder root # 127.0.0.1/demo

file A

[-] folder Thu muc 1

file B

folder Thu muc 1_1

file C

[-] folder Thu muc 1_2

file D

folder Thu muc 1_2_1

file E

folder Thu muc 2

file F

file E có link đến file A ở vị trí xác định, vậy trong file A có HTML element:

``

`liên kết đến A tại vị trí B`

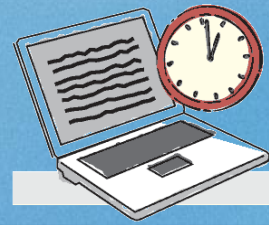
URL = [../ ../ ../file A.htm#positionB](http://127.0.0.1/Thu_muc_1_2_1/file_E.html#positionB)

Tổng kết



- Tìm hiểu về HTML và các chuẩn phiên bản HTML
- Tìm hiểu cấu trúc của một tài liệu HTML
- Biết được cách sử dụng các thẻ HTML cơ bản:
 - Các thẻ định dạng văn bản
 - Các thẻ hình ảnh
 - Các thẻ âm thanh
 - Các thẻ danh sách

Bài tập nhóm



- Sử dụng kiến thức đã học để tạo trang web hiển thị thông tin nhóm, bao gồm:
 - Hình chụp chung của các thành viên của nhóm
 - Thông tin chi tiết nhóm gồm tên nhóm, danh sách thành viên và MSSV
 - Sử dụng image map để khi nhấp vào mặt một thành viên sẽ chuyển đến trang thông tin thành viên



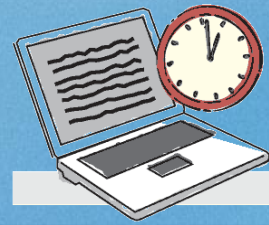
Bài tập nhóm



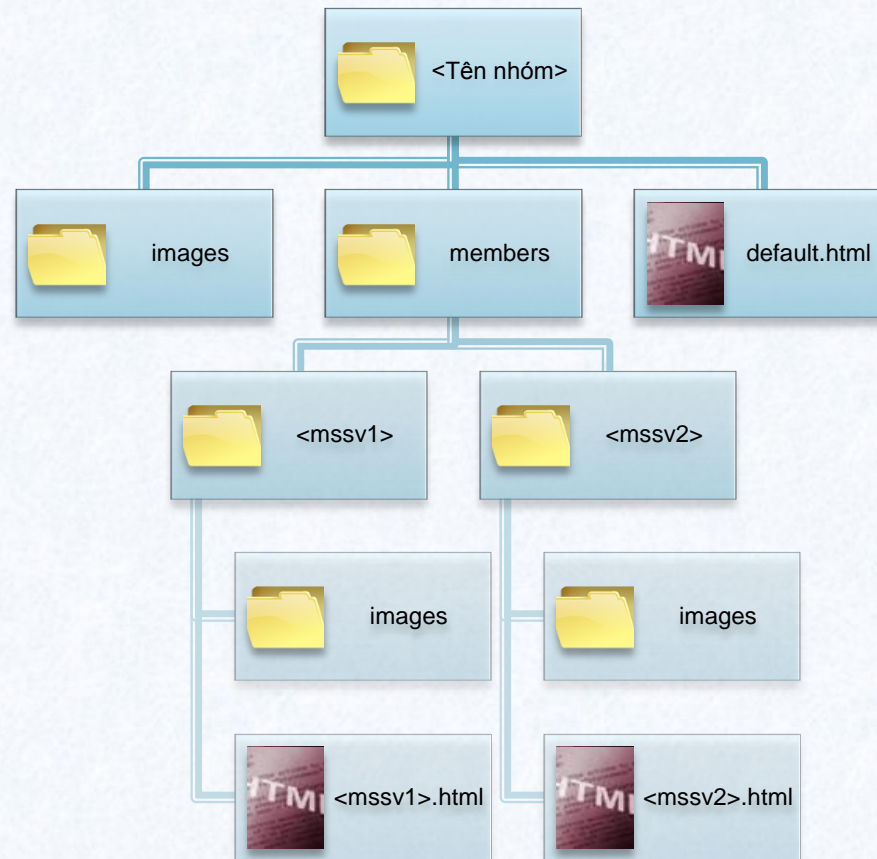
- Nội dung trang thông tin thành viên, bao gồm:
 - Hình đại diện
 - Họ tên
 - Mã số sinh viên
 - Ngày tháng năm sinh
 - Email
 - Điện thoại
 - Liên kết quay về trang thông tin nhóm



Bài tập nhóm



- Cấu trúc tập tin và thư mục của bài nộp:



Tham khảo thêm



- <http://w3schools.com>
- <http://w3.org>
- Wrox, Beginning HTML, XHTML, CSS and JavaScript, 2010

Lập trình Web 1



Khoa Công nghệ thông tin
Bộ môn Công nghệ phần mềm

Table

Nguyễn Huy Khánh
nhkhanh@fit.hcmus.edu.vn

Mục tiêu



- Hiểu được cấu trúc của bảng trong HTML và biết cách sử dụng các thẻ kẻ bảng
- Biết cách sử dụng Table trong trang web để trình bày nội dung
- Biết cách sử dụng Table để dàn trang

Bảng đơn giản

```
<table border="1" width="100%">  
  <caption>Bảng HTML đơn giản</caption>  
  <tr>  
    <th>0.1</th>  
    <th>0.2</th>  
  </tr>  
  <tr>  
    <td>1.1</td>  
    <td>1.2</td>  
  </tr>  
  <tr>  
    <td>2.1</td>  
    <td>2.2</td>  
  </tr>  
</table>
```

0.1	0.2
1.1	1.2
2.1	2.2

Các thẻ kẻ bảng

Thẻ	Ý nghĩa
<code><table></code>	Khởi tạo một bảng
<code><caption></code>	Tiêu đề của bảng
<code><tr></code>	Khởi tạo một dòng. Thẻ con của thẻ <code><table></code>
<code><th></code>	Khởi tạo một ô tiêu đề. Thẻ con của thẻ <code><tr></code>
<code><td></code>	Khởi tạo một ô. Thẻ con của thẻ <code><tr></code>

Giá vàng 9999

ĐVT: tr.đ/lượng

Loại	Mua	Bán
SBJ	31,520	31,540
SJC	31,490	31,560

Các thuộc tính cơ bản

Một số thuộc tính của các thẻ :

Thuộc tính	Ý nghĩa
<code>colspan</code>	Giãn cột cho ô
<code>rowspan</code>	Giãn dòng cho ô
<code>cellpadding</code>	Quy định khoảng cách từ biên của ô đến nội dung
<code>cellspacing</code>	Quy định khoảng cách giữa các ô với nhau

Thuộc tính colspan & rowspan

```
<table border="1">
  <tr>
    <th>Name</th>
    <th colspan="2">Telephone</th>
  </tr>
  <tr>
    <td>Bill Gate</td>
    <td>(505)256-3600</td>
    <td>0844 41 45 387</td>
  </tr>
</table>
```

Name	Telephone	
Bill Gate	(505)256-3600	0844 41 45 387

```
<table border="1">
  <tr>
    <th>Name</th>
    <th>Telephone</th>
  </tr>
  <tr>
    <td rowspan="2">Bill Gate</td>
    <td>(505)256-3600</td>
  </tr>
  <tr>
    <td>0844 41 45 387</td>
  </tr>
</table>
```

Name	Telephone
Bill Gate	(505)256-3600
	0844 41 45 387

Kết hợp colspan và rowspan

```
<table border="1">
  <tr>
    <th>Name</th>
    <th>Telephone</th>
    <th>Country</th>
    <th>Center</th>
  </tr>
  <tr>
    <td rowspan="2">Bill Gate</td>
    <td>(505)256-3600</td>
    <td colspan="2" rowspan="3">silicon valley - usa</td>
  </tr>
  <tr>
    <td>0844 41 45 387</td>
  </tr>
  <tr>
    <td>Steve Job</td>
    <td>1-800-676-2775</td>
  </tr>
</table>
```

Name	Telephone	Country	Center
Bill Gate	(505)256-3600	silicon valley - usa	
	0844 41 45 387		
Steve Job	1-800-676-2775		

Thuộc tính cellpadding & cellspacing

```
<table border="1" cellspacing="15">
  <tr>
    <th>Name</th>
    <th colspan="2">Telephone</th>
  </tr>
  <tr>
    <td>Bill Gate</td>
    <td>(505)256-3600</td>
    <td>0844 41 45 387</td>
  </tr>
</table>
```

Name	Telephone	
Bill Gate	(505)256-3600	0844 41 45 387

```
<table border="1" cellpadding="15">
  <tr>
    <th>Name</th>
    <th colspan="2">Telephone</th>
  </tr>
  <tr>
    <td>Bill Gate</td>
    <td>(505)256-3600</td>
    <td>0844 41 45 387</td>
  </tr>
</table>
```

Name	Telephone	
Bill Gate	(505)256-3600	0844 41 45 387

Ví dụ

Cell that spans two columns:

Name	Telephone	
Bill Gates	555 77 854	555 77 855

Cell that spans two rows:

First Name:	Bill Gates
Telephone:	555 77 854
	555 77 855

Without cellspacing:

First	Row
Second	Row

With cellspacing:

First	Row
Second	Row

Without cellpadding:

First	Row
Second	Row

With cellpadding:

First	Row
Second	Row

Đặt màu nền – Bảng

```
<table border="1" width="100%" style="background:red">  
  <caption>Bảng HTML đơn giản</caption>  
  <tr>  
    <th>0.1</th>  
    <th>0.2</th>  
  </tr>  
  <tr>  
    <td>1.1</td>  
    <td>1.2</td>  
  </tr>  
  <tr>  
    <td>2.1</td>  
    <td>2.2</td>  
  </tr>  
</table>
```

Bảng HTML đơn giản

0.1	0.2
1.1	1.2
2.1	2.2

Đặt màu nền – Dòng

```
<table border="1" width="100%">
  <caption>Bảng HTML đơn giản</caption>
  <tr>
    <th>0.1</th>
    <th>0.2</th>
  </tr>
  <tr style="background:red">
    <td>1.1</td>
    <td>1.2</td>
  </tr>
  <tr>
    <td>2.1</td>
    <td>2.2</td>
  </tr>
</table>
```

Bảng HTML đơn giản

0.1	0.2
1.1	1.2
2.1	2.2

Đặt màu nền - Ô

```
<table border="1" width="100%">
  <caption>Bảng HTML đơn giản</caption>
  <tr>
    <th>0.1</th>
    <th>0.2</th>
  </tr>
  <tr>
    <td style="background:red">1.1</td>
    <td>1.2</td>
  </tr>
  <tr>
    <td>2.1</td>
    <td>2.2</td>
  </tr>
</table>
```

Bảng HTML đơn giản

0.1	0.2
1.1	1.2
2.1	2.2

Đặt màu nền – Cột

```
<table border="1" width="100%">
  <caption>Bảng HTML đơn giản</caption>
  <tr>
    <th style="background:red">0.1</th>
    <th>0.2</th>
  </tr>
  <tr>
    <td style="background:red">1.1</td>
    <td>1.2</td>
  </tr>
  <tr>
    <td style="background:red">2.1</td>
    <td>2.2</td>
  </tr>
</table>
```

Bảng HTML đơn giản

0.1	0.2
1.1	1.2
2.1	2.2

Thẻ colgroup và col

Thẻ	Ý nghĩa
<code><col></code>	Định dạng chung cho một cột trong bảng Thẻ con của thẻ <code><table></code>
<code><colgroup></code>	Định dạng chung cho một nhóm cột trong bảng Thẻ con của thẻ <code><table></code>

The <col>

```
<table border="1">
  <caption>iPhone Price</caption>
  <col style="background-color:#09F" />
  <col style="background-color:#F30" />
  <tr>
    <th>Type</th>
    <th>Price</th>
  </tr>
  <tr>
    <td>iPhone 3GS - 8GB</td>
    <td>$99</td>
  </tr>
  <tr>
    <td>iPhone 4G - 16GB</td>
    <td>$199</td>
  </tr>
  <tr>
    <td>iPhone 3GS - 32GB</td>
    <td>$299</td>
  </tr>
</table>
```

Type	Price
iPhone 3GS - 8GB	\$99
iPhone 4G - 16GB	\$199
iPhone 3GS - 32GB	\$299

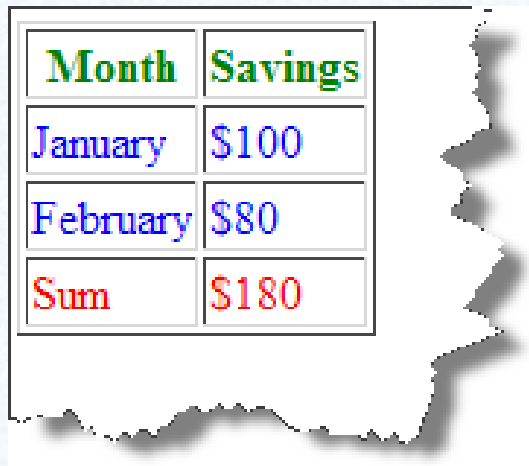
The <colgroup>

```
<table border="1">
  <colgroup span="2" style="background-color:#09F" />
  <colgroup style="background-color:#F30" />
  <tr>
    <th>Type</th>
    <th>Price</th>
    <th>Made in</th>
  </tr>
  <tr>
    <td>iPhone 3GS - 8GB</td>
    <td>$99</td>
    <td>USA</td>
  </tr>
  <tr>
    <td>iPhone 4G - 16GB</td>
    <td>$199</td>
    <td>USA</td>
  </tr>
  <tr>
    <td>iPhone 3GS - 32GB</td>
    <td>$299</td>
    <td>USA</td>
  </tr>
</table>
```

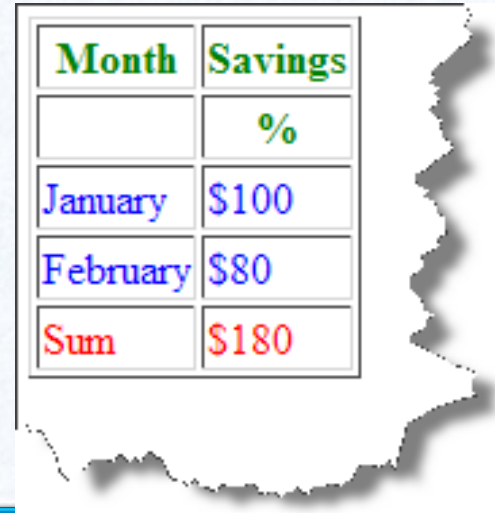
Type	Price	Made in
iPhone 3GS - 8GB	\$99	USA
iPhone 4G - 16GB	\$199	USA
iPhone 3GS - 32GB	\$299	USA

Quy định cấu trúc bảng

Thẻ	Ý nghĩa
<thead>	Quy định nhóm các dòng tiêu đề bảng
<tbody>	Quy định nhóm các dòng nội dung chính
<tfoot>	Quy định nhóm các dòng chân bảng



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



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

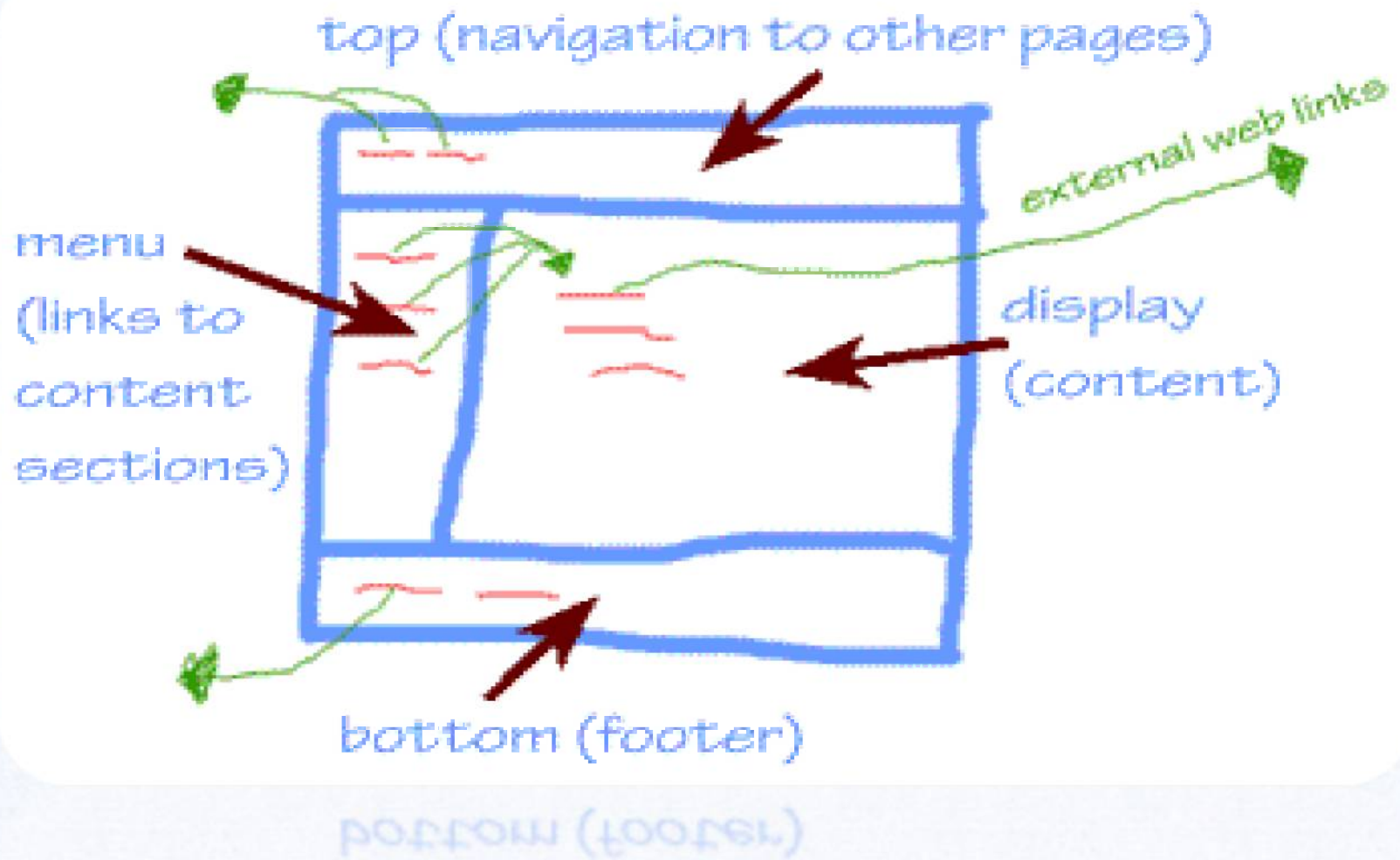
Qui định cấu trúc bảng

```
<table border="1" width="100%">
  <caption>Bảng HTML đơn giản</caption>
  <thead style="background-color:Red">
    <tr>
      <th>0.1</th>
      <th>0.2</th>
    </tr>
  </thead>
  <tfoot style="background-color:Blue">
    <tr>
      <td>3.1</td>
      <td>3.2</td>
    </tr>
  </tfoot>
  <tbody style="background-color:Green">
    <tr>
      <td>1.1</td>
      <td>1.2</td>
    </tr>
    <tr>
      <td>2.1</td>
      <td>2.2</td>
    </tr>
  </tbody>
</table>
```

Bảng HTML đơn giản

0.1	0.2
1.1	1.2
2.1	2.2
3.1	3.2

Dàn trang bằng Table



Ví dụ dàn trang bằng table

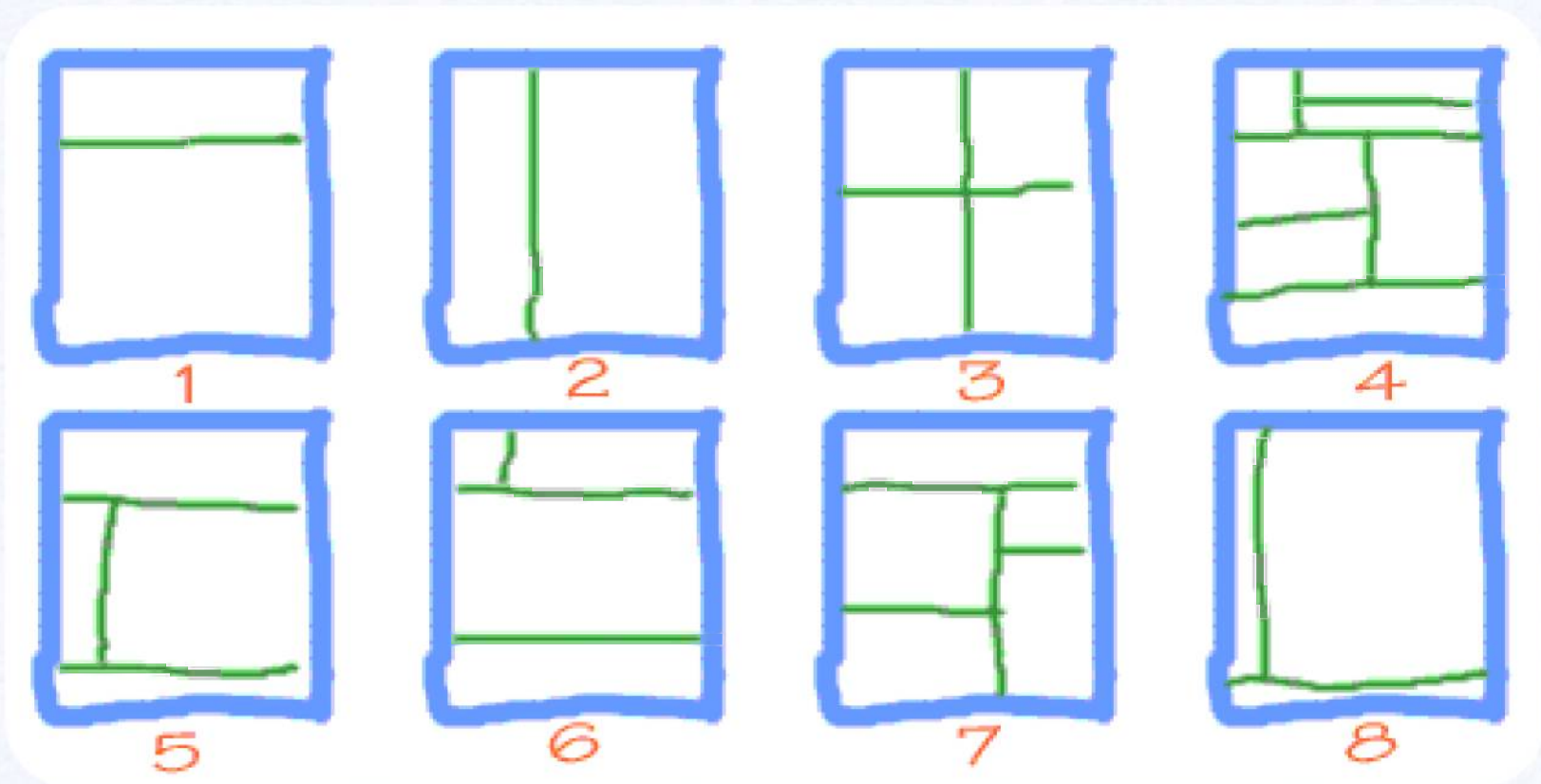
The image displays a website layout for the Faculty of Information Technology (Khoa Công Nghệ Thông Tin) at Ho Chi Minh City Natural Science University (ĐH. Khoa Học Tự Nhiên TP. HCM). The layout is structured as follows:

- Header:** Located at the top, it includes the university logo, navigation links (Webmail, Trang môn học, Liên hệ, Trang chủ, Góp ý, English), a search bar, and a login button (ĐĂNG NHẬP).
- Banner:** A large image showing students and faculty members in a library setting, with the text "Banner" overlaid.
- Menu:** A horizontal navigation bar containing links for "TRANG CHỦ", "TIN TỨC", "GIỚI THIỆU", "NGHIÊN CỨU", and "ĐÀO TẠO".
- Content:** The main area below the menu, featuring a "GIỚI THIỆU KHOA" section with a list of items and a "Content" label.
- Left Sidebar:** Contains sections for "GIÁO VIÊN" (Faculty), "SINH VIÊN" (Students), and "CỰU SINH VIÊN" (Alumni), each with a list of names and icons.

On the right side, an HTML editor window is visible, showing the underlying code for the page structure. The code includes a table with the following structure:

```
<table align="center" id="...">  
  <tbody>  
    <tr id="control">  
      <tr id="header">  
        <tr id="banner">  
      <tr>  
        <tr id="content">
```

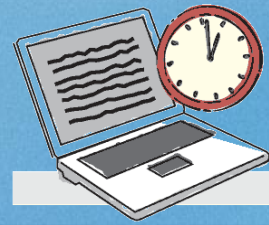
Một vài kiểu dàn trang



Trình bày nội dung

	Rẻ Nhất	Mới Nhất	Nhiều Nhất	
			 10.660.000 VNĐ	
	NB CPQ CQ42 - 260TU (XB800PA)			
	<p>Chi tiết: CPU: Intel® Dual-Core P6000 (2*1.86GHz, 3MB L3, 1066) RAM: 2GB DDR3 - 1066 HDD: 250GB SATA ODD: DVD-RW Graphics: Intel® HD Graphics (Share) Display: 14" HD BV LED Network: 10/100 Mbps, WLAN, Bluetooth Other: Webcam, Card Reader, HDMI Battery: 6 Cell Weight: 2.2Kg OS: DOS Bảo Hành :12 Tháng, Carry Bag TẶNG MOUSE OPTICAL</p>			

Bài tập nhóm



- Tìm ít nhất 4 trang web có cùng nội dung với trang web nhóm lựa chọn để làm đồ án cuối kì
 - Liệt kê và chụp hình các chức năng của các trang web đó
 - So sánh tính năng giữa các trang web này



Lập trình Web 1



Khoa Công nghệ thông tin
Bộ môn Công nghệ phần mềm

CSS căn bản

Nguyễn Huy Khánh
nhkhanh@fit.hcmus.edu.vn

Mục tiêu



- Hiểu được cấu trúc vai trò và ý nghĩa của CSS trong trang web
- Hiểu được ưu khuyết điểm của các loại CSS
- Biết cách xác định phạm vi ảnh hưởng của CSS đối với thẻ HTML

Nội dung

Giới thiệu CSS

Cách thức định nghĩa CSS

Sử dụng và Phân loại CSS

Selector trong CSS và phạm vi ảnh hưởng

Giới thiệu về CSS

- CSS = Cascading Style Sheets
- Dùng để mô tả cách hiển thị các thành phần trên trang WEB
- Sử dụng tương tự như dạng TEMPLATE
- Có thể sử dụng lại cho các trang web khác
- Có thể thay đổi thuộc tính từng trang hoặc cả site nhanh chóng (cascading)

Nội dung

Giới thiệu CSS

Cách thức định nghĩa CSS

Sử dụng và Phân loại CSS

Selector trong CSS và phạm vi ảnh hưởng

Định nghĩa Style

• Kiểu 1

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

Ví dụ:

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

• Kiểu 2

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

Ví dụ:

```
.TieuDe1 {  
  color: red;  
  font-family: Verdana, sans-serif; }  
<h1 class="TieuDe1"> DHKHTN </h1>
```

Định nghĩa Style – Ghi chú

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

Nội dung

Giới thiệu CSS

Cách thức định nghĩa CSS

Sử dụng và Phân loại CSS

Selector trong CSS và phạm vi ảnh hưởng

Phân loại CSS

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

Inline Style Sheet

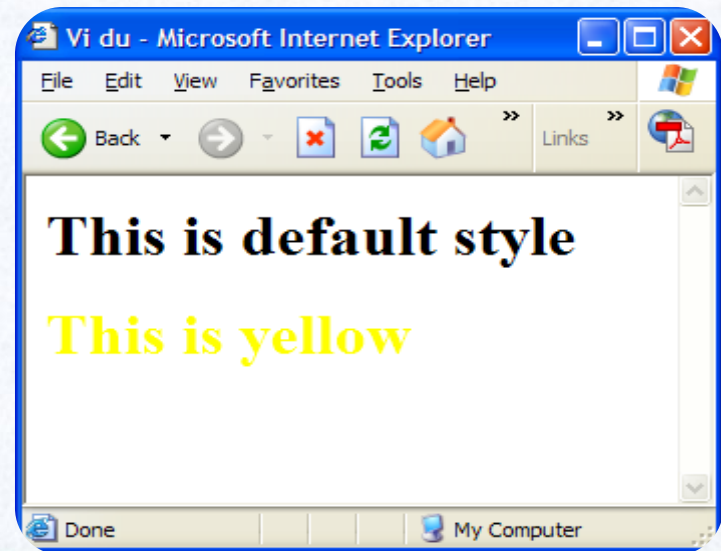
- Định nghĩa style trong thuộc tính style của từng tag HTML.
- Theo cú pháp **kiểu 1**.

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

- Không sử dụng lại được.

- Ví dụ:

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



Embedding Style Sheet

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

```
<head>
```

```
  <style type="text/css" >
```

```
    <!--
```

```
    SelectorName {  
        property1:value1;  
        property2:value2;  
        .....  
        propertyN:valueN;}
```

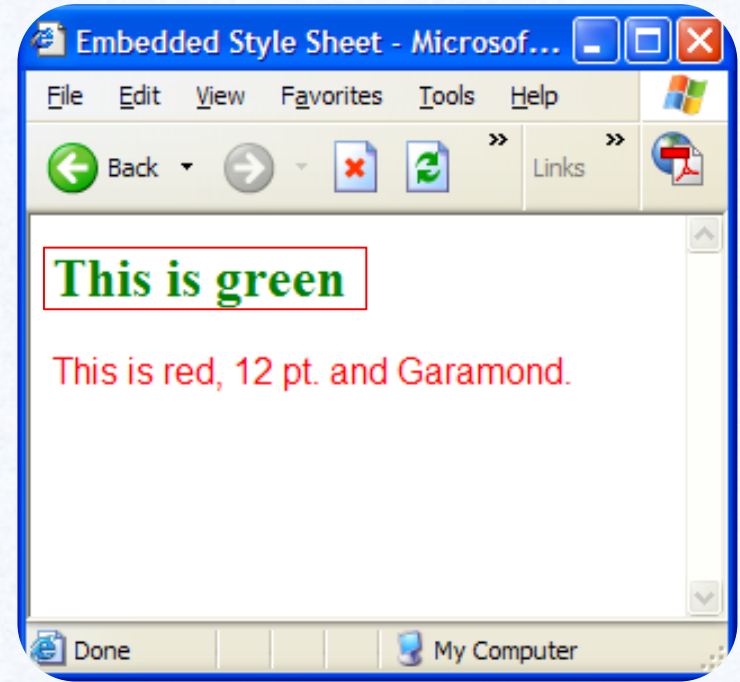
```
    -->
```

```
  </style>
```

```
</head>
```

Embedding Style Sheet

```
<html>
<head>
  <title>Embedded Style Sheet</title>
  <style type="text/css">
    <!--
    P {color: red;
        font-size: 12pt;
        font-family: Arial;}
    H2 {color: green;}
    -->
  </style>
  <style type="text/css">
    body
    {
      background-color: #FFFFFF;
    }
  </style>
</head>
<body>
  <h2>
    This is green</h2>
  <p>
    This is red, 12 pt. and Garamond.</p>
</body>
</html>
```



External Style Sheet

- Mọi style đều lưu trong file có phần mở rộng là ***.CSS**.
- File CSS: lưu trữ nhiều style theo cú pháp **kiểu 2**.

- Trong file HTML: liên kết bằng tag link. Cú pháp:

```
<head>
```

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

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

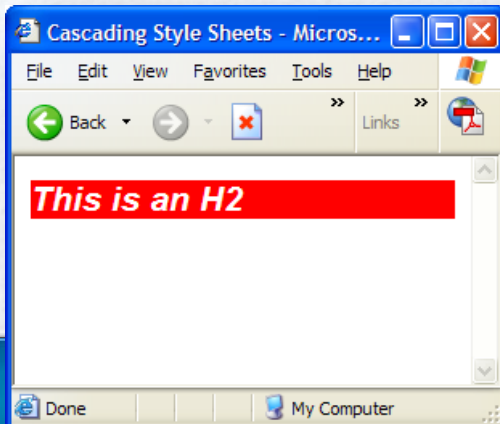
```
<head>
```

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

External Style Sheet

Trong tập tin **MyStyle.CSS**

```
h2
{
    font-weight: bold;
    font-size: 16pt;
    color: white;
    font-style: italic;
    font-family: Arial;
    background-color: red;
    font-color: white;
}
```



Trong trang Web : **demo.htm**

```
<html>
<head>
    <title>Cascading Style
    Sheets</title>
    <link href="MyStyle.css"
    rel="stylesheet"
    type="text/css">
</head>
<body>
    <h2>
        This is an H2</h2>
</body>
</html>
```

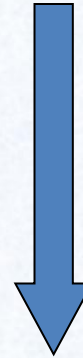
So sánh, Đánh giá

	Inline Style Sheet	Embedding Style Sheet	External Style Sheet
Khai báo	Kiểu 1	Kiểu 2	Kiểu 2
Cú pháp	<pre><p style="color:red;"> Test </p></pre>	<pre><style type="text/css"> .TieuDe1{color: red;} </style> <p class="TieuDe1"> Test </p></pre>	<pre><link rel="stylesheet " href="main.css" /> <p class="TieuDe1"> Test </p></pre>
Ưu điểm	<ul style="list-style-type: none">• Dễ dàng quản lý Style theo từng tag của tài liệu web.• Có độ ưu tiên cao nhất	<ul style="list-style-type: none">• Dễ dàng quản lý Style theo từng tài liệu web.• Không cần tải thêm các trang thông tin khác cho style	<ul style="list-style-type: none">• Có thể thiết lập Style cho nhiều tài liệu web.• Thông tin các Style được trình duyệt cache lại
Khuyết điểm	<ul style="list-style-type: none">• Cần phải Khai báo lại thông tin style trong từng tài liệu Web và các tài liệu khác một cách thủ công.• Khó cập nhật style	<ul style="list-style-type: none">• Cần phải khai báo lại thông tin style cho các tài liệu khác trong mỗi lần sử dụng	<ul style="list-style-type: none">• Tốn thời gian download file *.css và làm chậm quá trình biên dịch web ở trình duyệt trong lần đầu sử dụng

Độ ưu tiên

- Thứ tự ưu tiên áp dụng định dạng khi sử dụng các loại CSS (độ ưu tiên giảm dần) :

1. Inline Style Sheet
2. Embedding Style Sheet
3. External Style Sheet
4. Browser Default



```

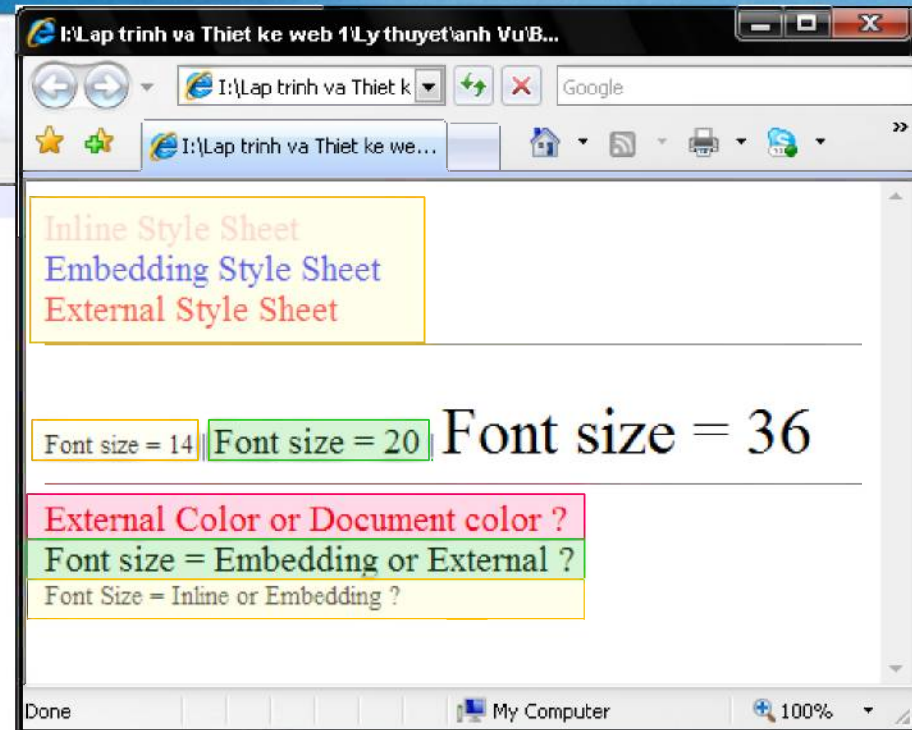
PriorityCSS.css
1  .MyExColor {
2     Color : Red;
3  }
4
5  div{
6     font-size:36px;
7  }

```

```

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

```



Nội dung

Giới thiệu CSS

Cách thức định nghĩa CSS

Sử dụng và Phân loại CSS

Selector trong CSS và phạm vi ảnh hưởng

Selector

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

Ví dụ:

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

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

Selector trong CSS

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

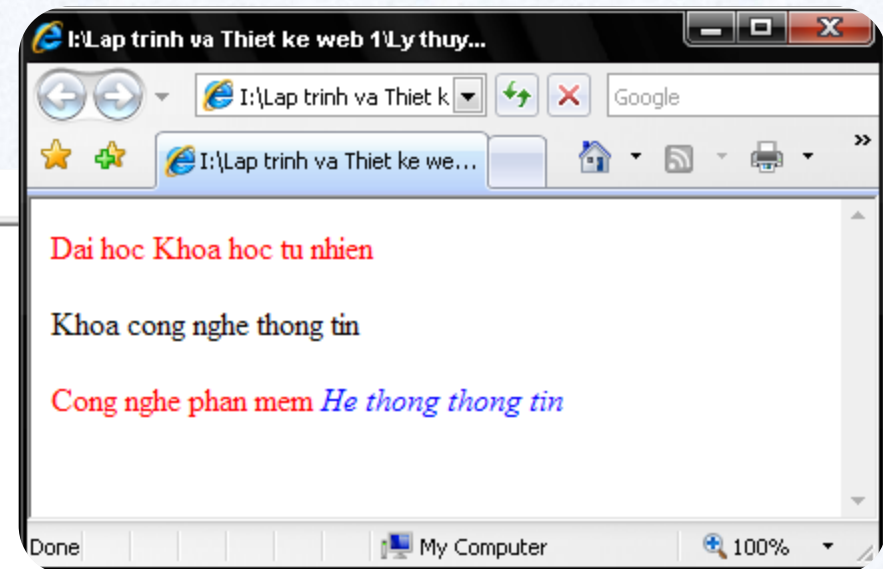
Element rules

- Có hiệu ứng trên tất cả element cùng loại tag
- Ví dụ :

```
Test.htm
<html>
  <head>
    <style type="text/css">
      p {color:red}
      Em {color:blue}
    </style>
  </head>
  <body>
    <p> Truong Dai hoc Khoa hoc tu nhien </p>

    <div>Khoa cong nghe thong tin</div>

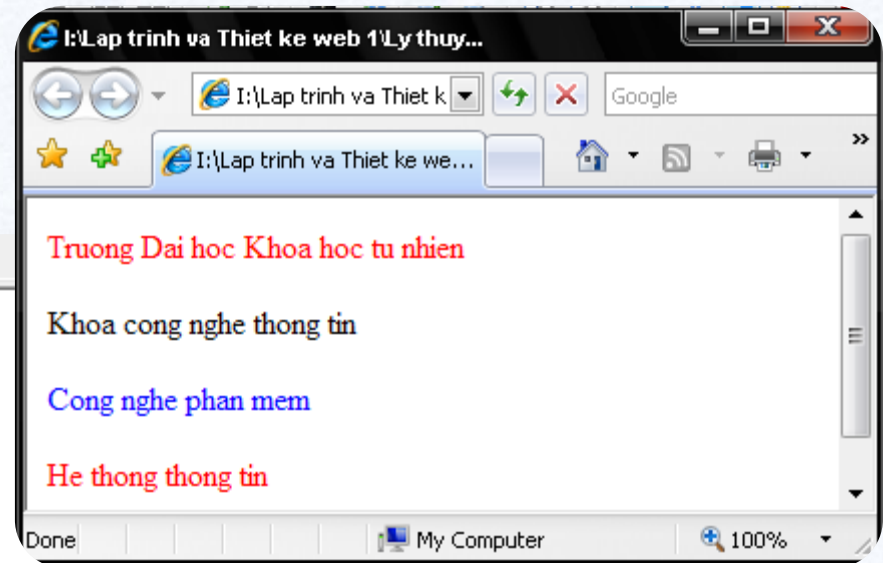
    <p> Cong nghe phan mem <em> He thong thong tin </em> </p>
  </body>
</html>
```



ID rules

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

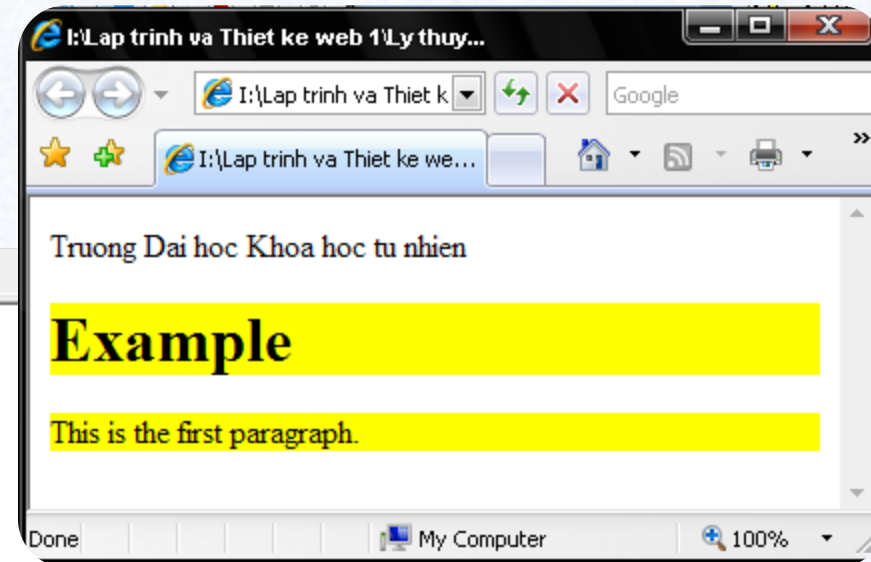
```
Test.htm
<html>
<head>
  <style type="text/css">
    #id001{color:red}
    #id002{color:blue}
  </style>
</head>
<body>
  <p id="id001"> Truong Dai hoc Khoa hoc tu nhien </p>
  <p>Khoa cong nghe thong tin</p>
  <div id="id002">Cong nghe phan mem<p id="id001">He thong thong tin</p></div>
</body>
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



Class rules

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

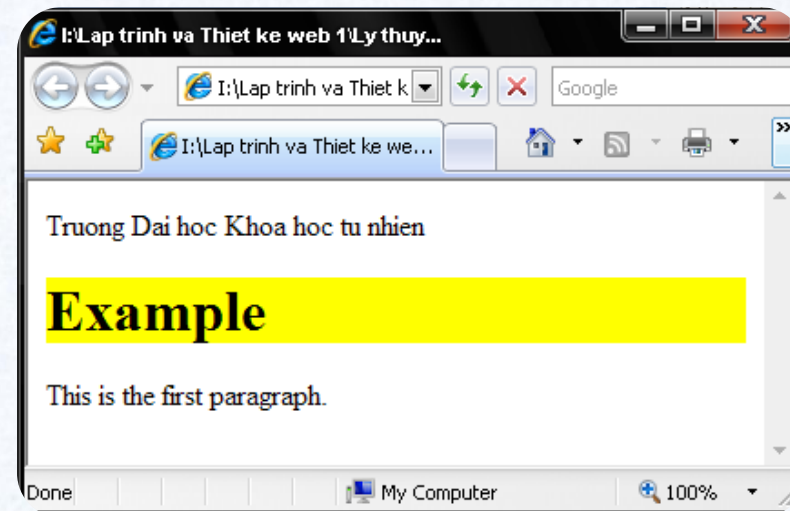
```
Test.htm
<html>
  <head>
    <style type="text/css">
      .veryimportant {background-color: yellow;}
    </style>
  </head>
  <body>
    <p id="id001"> Truong Dai hoc Khoa hoc tu nhien </p>
    <h1 class="veryimportant">Example</h1>
    <p class="veryimportant">This is the first paragraph.</p>
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



Kết hợp Element và Class

- Ví dụ :

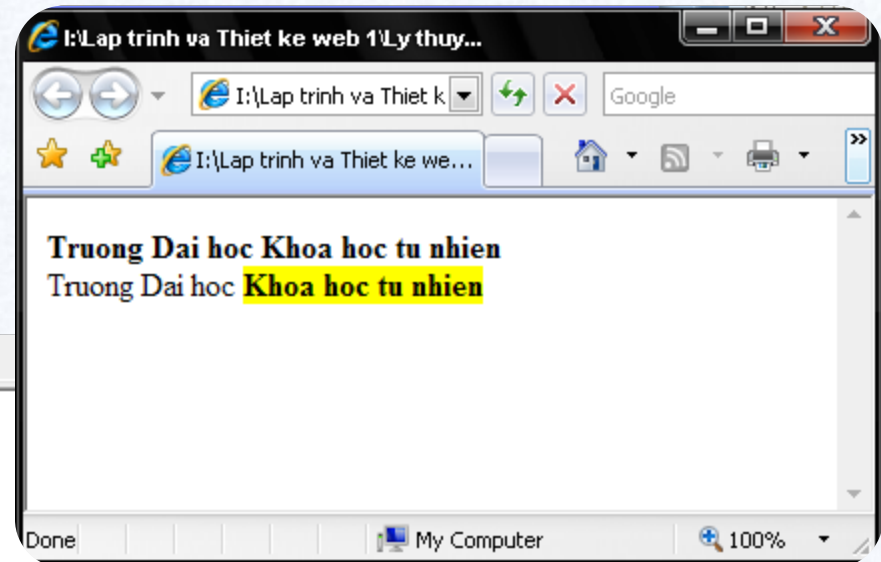
```
<html>
<head>
  <style type="text/css">
    h1.veryimportant {background-color: yellow;}
  </style>
</head>
<body>
  <p id="id001"> Truong Dai hoc Khoa hoc tu nhien </p>
  <h1 class="veryimportant">Example</h1>
  <p class="veryimportant">This is the first paragraph.</p>
</body>
</html>
```



Contextual Selection

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

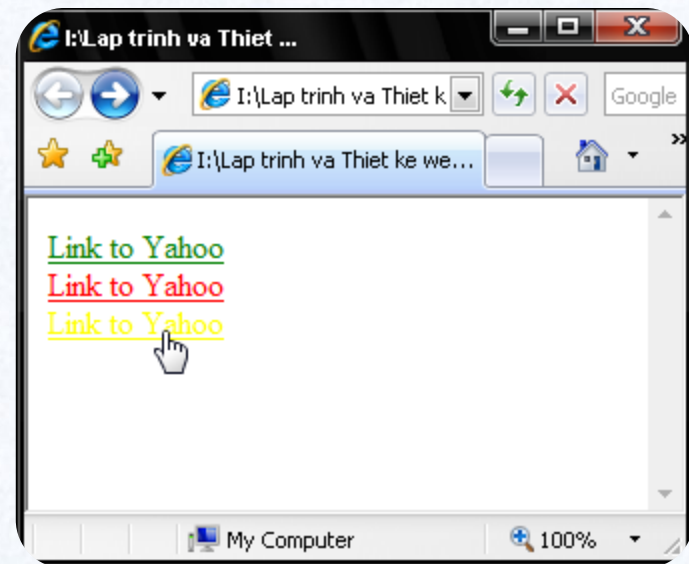
```
1 <html>
2   <head>
3     <style type="text/css">
4       div b {background-color: yellow;}
5     </style>
6   </head>
7   <body>
8     <b><div>Truong Dai hoc Khoa hoc tu nhien </div></b>
9     <div>Truong Dai hoc <b>Khoa hoc tu nhien</b></div>
10  </body>
11 </html>
```



Pseudo Class

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

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



Bài tập nhóm



- Làm lại bài tập nhóm ở phần HTML Căn bản và trình bày lại trang bằng table và.
- Lưu ý
 - Sử dụng tối đa CSS để trình bày trang web (không sử dụng thuộc tính của thẻ HTML khi CSS có thể thực hiện được chức năng tương tự)
 - Chỉ sử dụng External Style Sheet và lưu trong tập tin default.css



Tham khảo thêm



- Wrox, Beginning HTML, XHTML, CSS and JavaScript, 2010
- <http://www.w3schools.com/css/>

Lập trình Web 1



Khoa Công nghệ thông tin
Bộ môn Công nghệ phần mềm

CSS Nâng cao

Nguyễn Huy Khánh
nhkhanh@fit.hcmus.edu.vn

Mục tiêu



- Biết cách sử dụng các thuộc tính của CSS trong việc dàn trang web

Nội dung

CSS cho Table

Box model

Thuộc tính display và visibility

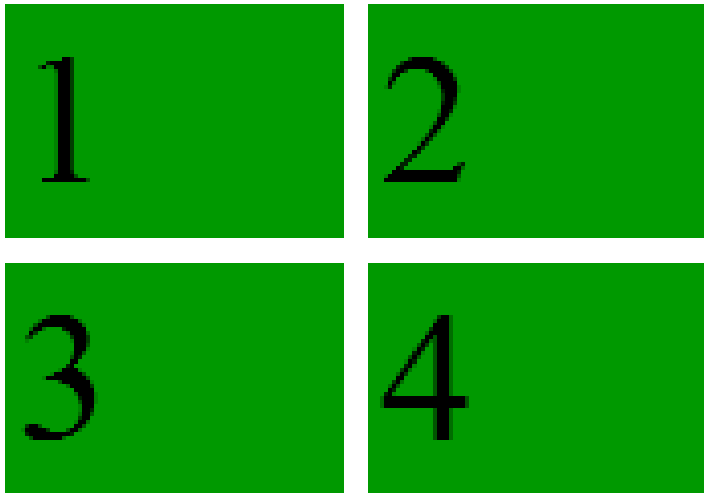
Thuộc tính position

Thuộc tính float

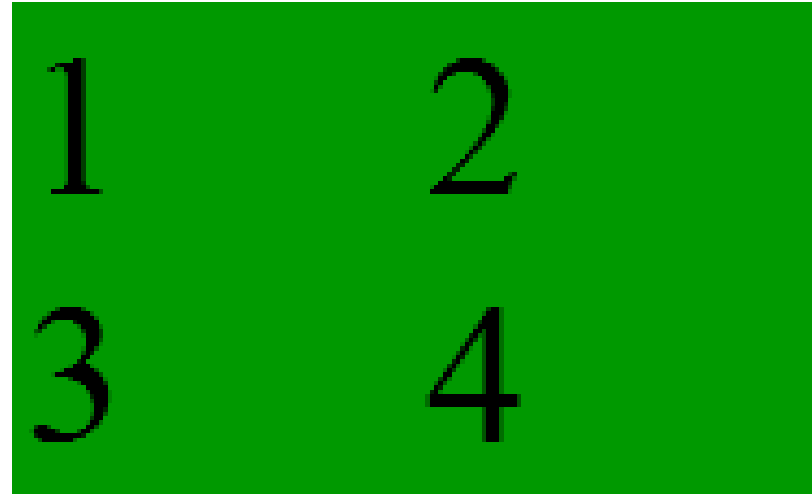
Canh chỉnh phần tử

CSS cho Table

- Dùng `border-spacing` để thay thế `cellspacing`



```
table
{
    width: 100%;
}
```

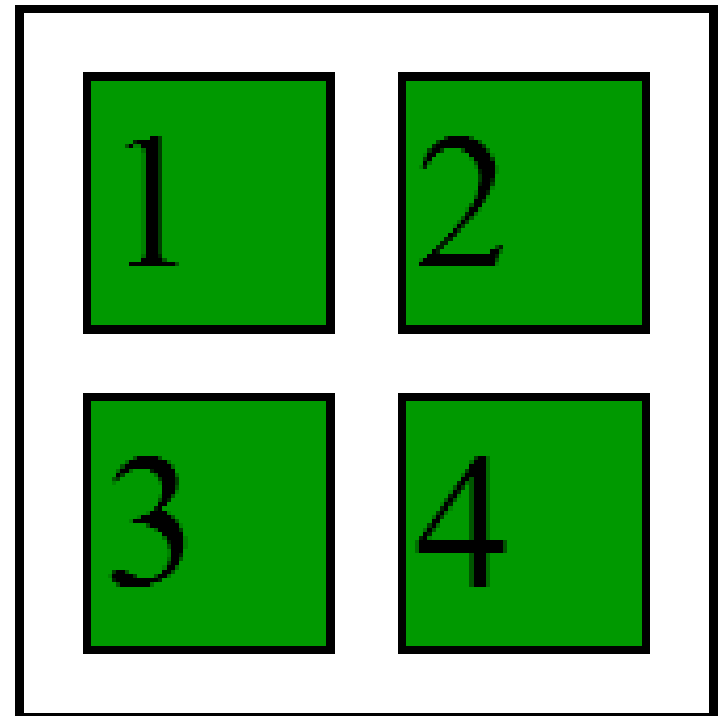


```
table
{
    width: 100%;
    border-spacing: 0px;
}
```

CSS Table Border

- Kiểu kẻ biên `separate`

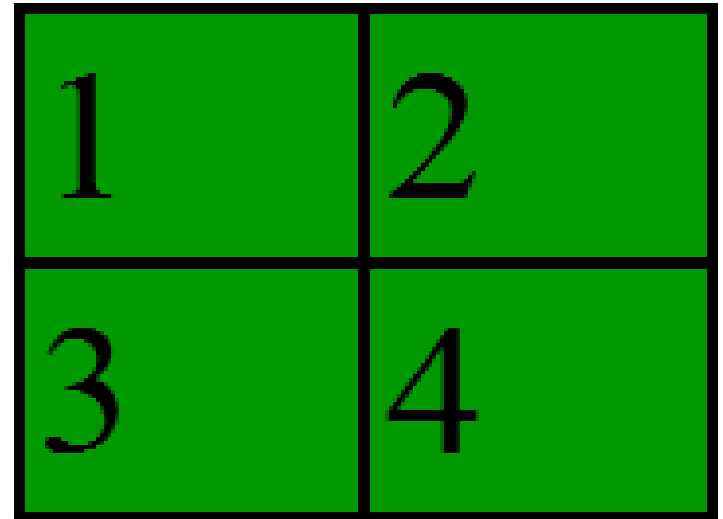
```
table
{
  width: 100%;
  border:1px solid Black;
  border-collapse:separate;
  border-spacing:5px;
}
td
{
  background-color: #009900;
  border:1px solid Black;
}
```



CSS Table Border

- Kiểu kẻ biên `collapse`

```
table
{
  width: 100%;
  border:1px solid Black;
  border-collapse:collapse;
  border-spacing:5px;
}
td
{
  background-color: #009900;
  border:1px solid Black;
}
```



1	2
3	4

CSS Table canh chữ

- Canh ngang: `text-align`
- Canh dọc: `vertical-align`

```
td
{
    background-color: #009900;
    border:1px solid Black;
    text-align:center;
    vertical-align:bottom;
    height:30px;
}
```

1	2
3	4

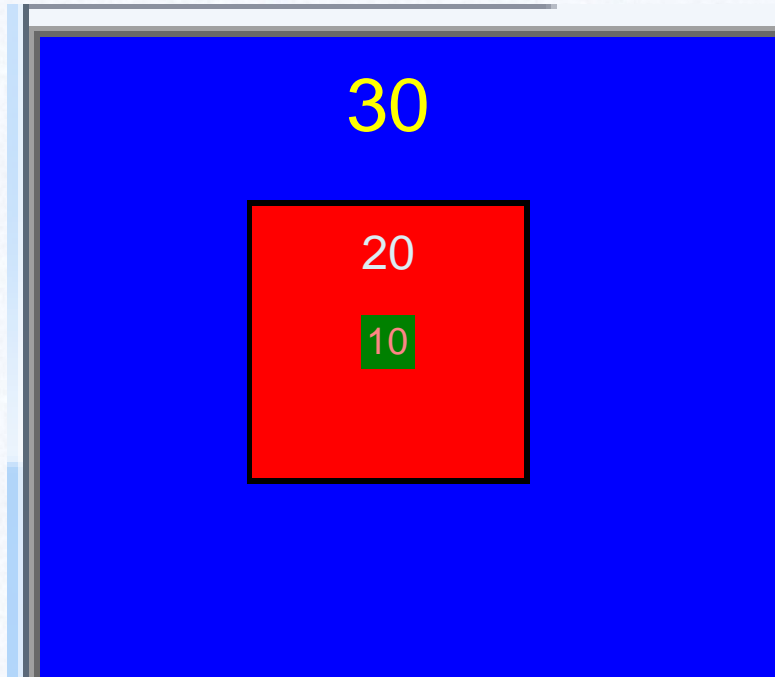
Box model



http://w3schools.com/css/css_boxmodel.asp

Ví dụ về box model

```
<body style="background-color: Blue">  
  <div style="width: 10px; height: 10px; padding: 20px; border: 1px solid black;  
    margin: 30px; background-color: Red">  
    <div style="width: 100%; height: 100%; background-color: Green">  
    </div>  
  </div>  
</body>
```



Thuộc tính Display và Visibility

● Ẩn một phần tử

● `display: none`

```
<p>  
Đại học <span style="display: none">Khoa học</span> <span>tự nhiên</span>  
</p>
```

Đại học tự nhiên

● `visibility: hidden`

```
<p>  
Đại học <span style="visibility: hidden">Khoa học</span> <span>tự nhiên</span>  
</p>
```

Đại học tự nhiên

Thuộc tính Display và Visibility

- Phân loại phần tử:
 - Block: chiếm trọn chiều rộng và tự động xuống hàng trước và sau phần tử
 - h1
 - p
 - div
 - Inline: chỉ chiếm đúng bề ngang phần nội dung
 - a
 - b
 - span
- Có thể dùng thuộc tính **display** để thay đổi loại phần tử

```
Li {display:inline;}
```

[HTML](#) [CSS](#) [JavaScript](#) [XML](#)

```
<ul>  
  <li><a href="/html/default.asp" target="_blank">HTML</a></li>  
  <li><a href="/css/default.asp" target="_blank">CSS</a></li>  
  <li><a href="/js/default.asp" target="_blank">JavaScript</a></li>  
  <li><a href="/xml/default.asp" target="_blank">XML</a></li>  
</ul>
```

Cách xác định vị trí

- Sử dụng thuộc tính **position**

position	Chiếm chỗ	Ý nghĩa
<code>static</code>	✓	Giá trị mặc định không ảnh hưởng bởi top, bottom, left, right
<code>fixed</code>	✗	Cố định vị trí theo trình duyệt, không thay đổi vị trí khi kéo thanh trượt của trình duyệt
<code>relative</code>	✓	Tọa độ của phần tử được tính theo vị trí đứng bình thường của phần tử
<code>absolute</code>	✗	Khi không đặt vị trí của phần tử, phần tử sẽ ở vị trí đứng bình thường và cho phép phần tử khác chiếm chỗ. Tọa độ của phần tử (nếu có) được tính theo phần tử cha đầu tiên có position khác <code>static</code> .

CSS Float

- Thuộc tính **float** cho biết phần tử sẽ trôi về trái hoặc phải.
- Những phần tử tiếp theo phần tử được **float** sẽ dồn vào chiếm chỗ trống của phần tử đó.

```
<h1>Ví dụ về float: right</h1>
```

```

```

```
<p>
```

```
Trường đại học Khoa học Tự nhiên - TP. Hồ Chí Minh. Trường đại học Khoa học Tự nhiên  
- TP. Hồ Chí Minh. Trường đại học Khoa học Tự nhiên - TP. Hồ Chí Minh. Trường đại  
học Khoa học Tự nhiên - TP. Hồ Chí Minh. Trường đại học Khoa học Tự nhiên - TP.  
Hồ Chí Minh. Trường đại học Khoa học Tự nhiên - TP. Hồ Chí Minh. Trường đại học  
Khoa học Tự nhiên - TP. Hồ Chí Minh. </p>
```

Ví dụ về float: right

```
Trường đại học Khoa học Tự nhiên - TP. Hồ Chí Minh.  
Trường đại học Khoa học Tự nhiên - TP. Hồ Chí Minh.  
Trường đại học Khoa học Tự nhiên - TP. Hồ Chí Minh.  
Trường đại học Khoa học Tự nhiên - TP. Hồ Chí Minh.  
Trường đại học Khoa học Tự nhiên - TP. Hồ Chí Minh.  
Trường đại học Khoa học Tự nhiên - TP. Hồ Chí Minh. Trường đại học  
Khoa học Tự nhiên - TP. Hồ Chí Minh.
```



CSS Float

- Sử dụng thuộc tính **clear** để cấm không phần tử sử dụng vùng trống còn lại của phần tử **float** trước đó

```
<h1>Ví dụ về float và clear</h1>
```

```

```

```

```

```
<p style="clear:both">Trường đại học Khoa học Tự nhiên - TP. Hồ Chí Minh.</p>
```

Ví dụ về float và clear



Trường đại học Khoa học Tự nhiên - TP. Hồ Chí Minh.

Canh phần tử block

- Canh giữa

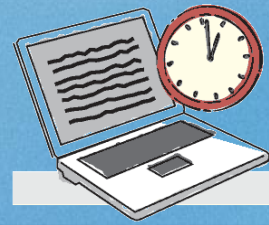
```
.center
{
    margin-left: auto;
    margin-right: auto;
    width: 70%;
    background-color: #b0e0e6;
}
```

- Canh phải

```
.right
{
    position: absolute;
    right: 0px;
    width: 300px;
}
```

```
.right
{
    float: right;
    width: 300px;
}
```

Bài tập nhóm



- Tìm kiếm hoặc tạo template cho trang chủ của đề án cuối kì
- Hình dung trước và điền tĩnh nội dung của trang chủ



JAVASCRIPT & DOM HTML

Nội dung

- Giới thiệu về Javascript
- Nhúng Javascript vào trang web
- Kiểu dữ liệu & cú pháp trong Javascript
- Xử lý sự kiện
- DOM HTML với Javascript
- Ví dụ

Nội dung

- *Giới thiệu về Javascript*
- Nhúng Javascript vào trang web
- Kiểu dữ liệu & Các cú pháp Javascript
- Xử lý sự kiện
- DOM HTML với Javascript
- Ví dụ

Giới thiệu về Javascript

- Là ngôn ngữ Client-side script hoạt động trên trình duyệt của người dùng (client)
- Chia sẻ xử lý trong ứng dụng web. Giảm các xử lý không cần thiết trên server.
- Giúp tạo các hiệu ứng, tương tác cho trang web.

Giới thiệu về Javascript

- Client-Side Script:
 - Script được thực thi tại Client-Side (trình duyệt): Thực hiện các tương tác với người dùng (tạo menu chuyển động, ...) , kiểm tra dữ liệu nhập, ...
- Server-Side Script:
 - Script được xử lý tại Server-Side, nhằm tạo các trang web có khả năng phát sinh nội dung động. Một số xử lý chính: kết nối CSDL, truy cập hệ thống file trên server, phát sinh nội dung html trả về người dùng...

Giới thiệu về Javascript

- o Khi trình duyệt (Client browser) truy cập trang web có chứa các đoạn mã xử lý tại server-side. Server (run-time engine) sẽ thực hiện các lệnh Server-side Scripts và trả về nội dung HTML cho trình duyệt

Nội dung html trả về chủ yếu bao gồm: mã html, client-script.

Nội dung

- Giới thiệu về Javascript
- **Nhúng Javascript vào trang web**
- Kiểu dữ liệu & Các cú pháp Javascript
- Xử lý sự kiện
- DOM HTML với Javascript
- Ví dụ

Nhúng Javascript vào trang web

Định nghĩa script trực tiếp trong trang html:

```
<script type="text/javascript">  
  <!--  
  // Lệnh Javascript  
  -->  
</script>
```

Nhúng sử dụng script cài đặt từ 1 file .js khác:

```
<script src="xxx.js"></script>
```

Nhúng Javascript vào trang web

```
<html>  
  <script type="text/javascript">  
    <head>  
      some javascript statements  
      </script>  
      some statements  
    </script>  
  </head>  
  <body>  
    <script type="text/javascript">  
      some statements  
    </script>  
    <script src="Tên_file_script.js">method()</script>  
    <script type="text/javascript">  
      // gọi thực hiện các phương thức được định nghĩa  
      // trong "Tên_file_script.js"  
    </script>  
  </body>  
</html>
```

Nhúng Javascript vào trang web

- Đặt giữa tag `<head>` và `</head>`: script sẽ thực thi ngay khi trang web được mở.
- Đặt giữa tag `<body>` và `</body>`: script trong phần body được thực thi khi trang web đang mở (sau khi thực thi các đoạn script có trong phần `<head>`).
- Số lượng đoạn client-script chèn vào trang không hạn chế.

VD: Nhúng Javascript vào trang web

```
<html>
```

```
<body>
```

```
document.write("Hello world!");
```

```
<script type="text/javascript">
```

```
document.write("Hello world!");
```

```
</script>
```

```
</body>
```

```
</html>
```



Nội dung

- Giới thiệu về Javascript
- Nhúng Javascript vào trang web
- Kiểu dữ liệu & Các cú pháp Javascript
- Xử lý sự kiện
- DOM HTML với Javascript
- Ví dụ

Biến số trong Javascript

- Cách đặt tên biến
 - Bắt đầu bằng một chữ cái hoặc dấu _
 - A..Z,a..z,0..9,_ : phân biệt HOA, Thường
- Khai báo biến
 - Sử dụng từ khóa **var**
Ví dụ: **var count=10,amount;**
 - Không cần khai báo biến trước khi sử dụng, biến thật sự tồn tại khi bắt đầu sử dụng lần đầu tiên

Kiểu dữ liệu trong Javascript

Kiểu dữ liệu	Ví dụ	Mô tả
Object	<code>var listBooks = new Array(10) ;</code>	Trước khi sử dụng, phải cấp phát bằng từ khóa <code>new</code>
String	<code>"The cow jumped over the moon."</code> <code>"40"</code>	Chứa được chuỗi unicode Chuỗi rỗng <code>"</code>
Number	<code>0.066218</code> <code>12</code>	Theo chuẩn IEEE 754
boolean	<code>true / false</code>	
undefined	<code>var myVariable ;</code>	<code>myVariable = undefined</code>
null	<code>connection.Close();</code>	<code>connection = null</code>

1 Biến trong javascript có thể lưu bất kỳ kiểu dữ liệu nào.

Đổi kiểu dữ liệu

- Biến tự đổi kiểu dữ liệu khi giá trị mà nó lưu trữ thay đổi

Ví dụ:

```
var x = 10; // x kiểu Number  
x = "hello world !"; // x kiểu String
```

- Có thể cộng 2 biến khác kiểu dữ liệu

Ví dụ:

```
var x;  
x = "12" + 34.5; // KQ: x = "1234.5"
```

- Hàm `parseInt(...)`, `parseFloat(...)`: Đổi KDL từ chuỗi sang số.

Hàm trong Javascript

- Dạng thức khai báo chung:

```
function Tên_hàm(thamso1, thamso2,..)  
{  
    .....  
}
```

- Hàm có giá trị trả về:

```
function Tên_hàm(thamso1, thamso2,..)  
{  
    .....  
    return (value);  
}
```

Hàm trong Javascript

- Ví dụ:

```
function Sum(x, y)
{
    tong = x + y;
    return tong;
}
```

- Gọi hàm:

```
var x = Sum(10, 20);
```

Các quy tắc chung

- Khối lệnh được bao trong dấu {}
- Mỗi lệnh nên kết thúc bằng dấu ;
- Cách ghi chú thích:
 - // Chú thích 1 dòng
 - /* Chú thích
nhiều dòng */

Câu lệnh if

```
if (condition)  
{  
    statement[s] if true  
}  
else  
{  
    statement[s] if false  
}
```

Ví dụ:

```
var x = 5, y = 6, z;
```

```
- if (x == 5) { if (y == 6) z = 17; } else z = 20;
```

Câu lệnh switch

switch (*expression*)

```
{  
    case label :  
        statementlist  
    case label :  
        statementlist  
        ...  
    default :  
        statement list  
}
```

Ví dụ :

```
var diem = "G";  
switch (diem) {  
    case "Y":  
        document.write("Yếu");  
        break;  
    case "TB":  
        document.write("Trung bình");  
        break;  
    case "K":  
        document.write("Khá");  
        break;  
    case "G":  
        document.write("Giỏi");  
        break;  
    default:  
        document.write("Xuất sắc")  
}
```

Vòng lặp for

```
for ([initial expression]; [condition];  
      [update expression]) {  
    statement[s] inside loop  
}
```

Ví dụ:

```
var myarray = new Array();  
for (i = 0; i < 10; i++)  
{  
    myarray[i] = i;  
}
```


Vòng lặp while

```
while (expression)
```

```
{
```

```
    statements
```

```
}
```

Ví dụ:

```
var i = 9, total = 0;
```

```
while (i < 10)
```

```
{
```

```
    total += i * 3 + 5;
```

```
    i = i + 5;
```

```
}
```

Vòng lặp do.. while

```
do  
  {  
    statement  
  } while (expression);
```

Ví dụ:

```
var i = 9, total = 0;  
do  
{  
    total += i * 3 + 5;  
    i = i + 5;  
} while (i > 10);
```

Nội dung

- Giới thiệu về Javascript
- Nhúng Javascript vào trang web
- Kiểu dữ liệu & Các cú pháp Javascript
- *Xử lý sự kiện*
- DOM HTML với Javascript
- Ví dụ

Các sự kiện thông dụng

- Các sự kiện được hỗ trợ bởi hầu hết các đối tượng
 - onClick
 - onFocus
 - onChange
 - onBlur
 - onMouseOver
 - onMouseOut
 - onMouseDown
 - onMouseUp
 - onLoad
 - onSubmit
 - onResize
 -

Xử lý sự kiện cho các thẻ HTML

- Cú pháp 1:

```
<TAG eventHandler = "JavaScript Code">
```

- Ví dụ:

```
<body>
```

```
<INPUT TYPE="button" NAME="Button1"
```

```
VALUE="OpenSesame!"
```

```
onClick="window.open('mydoc.html');">
```

```
</body>
```

- Lưu ý: Dấu “...” và ‘...’

Xử lý sự kiện bằng function

```
<head>  
  <script language="Javascript">  
    function GreetingMessage()  
    {  
      window.alert("Welcome to my  
world");  
    }  
  </script>  
</head>  
  
<body onload="GreetingMessage()">  
</body>
```

Xử lý sự kiện bằng thuộc tính

- Gán tên hàm xử lý cho 1 object event

`object.eventhandler = function_name;`

- Ví dụ:

```
<head>
  <script language="Javascript">
    function GreetingMessage ()
    {
      window.alert("Welcome to my world");
    }

    window.onload = GreetingMessage ()
  </script>
</head>

<body>
</body>
```

Ví dụ: onclick Event

```
<SCRIPT LANGUAGE="JavaScript">
```

```
function compute(frm)
```

```
{
```

```
var x = frm.expr.value;
```

```
result.innerHTML = x*x;
```

```
}
```

```
</SCRIPT>
```

```
<FORM name="frm">
```

```
X = <INPUT TYPE="text" NAME="expr" SIZE=15>
```

```
<BR><BR>
```

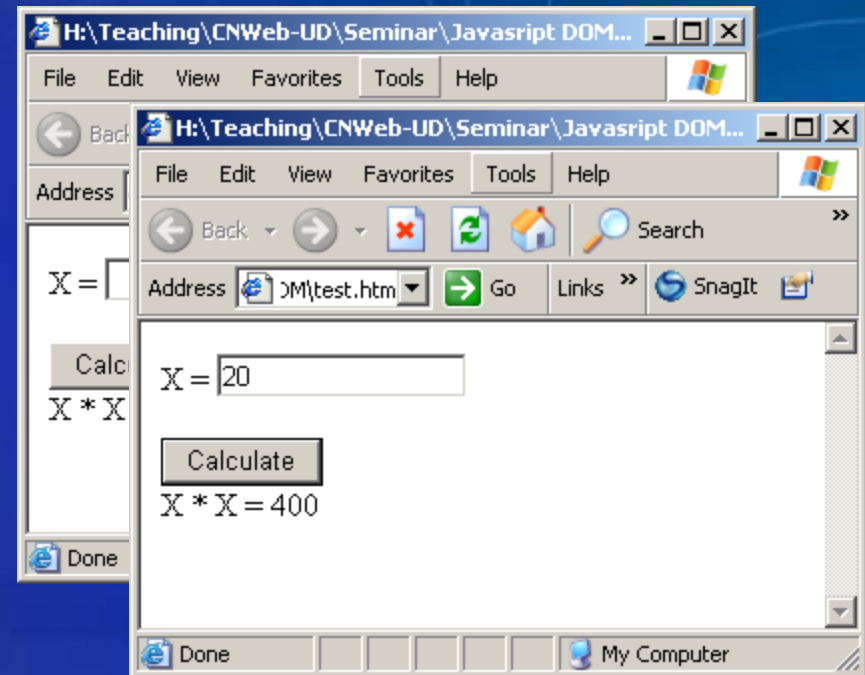
```
<INPUT TYPE="button" VALUE="Calculate"
```

```
ONCLICK="compute(this.form)">
```

```
<BR>
```

```
X * X = <SPAN ID="result"></SPAN>
```

```
</FORM>
```



Ví dụ: onFocus - onBlur



```
<BODY BGCOLOR="lavender">
```

```
<FORM>
```

```
<INPUT type="text" name="myTextbox"
```

```
onblur="(document.bgColor='aqua')"
```

```
onfocus="(document.bgColor='dimgray')">
```

```
</FORM>
```

```
</BODY>
```

Nội dung

- Giới thiệu về Javascript
- Nhúng Javascript vào trang web
- Kiểu dữ liệu & Các cú pháp Javascript
- Sử dụng các đối tượng trong Javascript
- Xử lý sự kiện
- **DOM HTML với Javascript**
- Ví dụ

Đối tượng HTML DOM

- DOM = Document Object Model
- Là tập hợp các đối tượng HTML chuẩn được dùng để **truy xuất** và **thay đổi thành phần HTML** trong trang web (thay đổi nội dung tài liệu của trang)
- Một số đối tượng của DOM: window, document, history, link, form, frame, location, event, ...

Đối tượng Window - DOM

- Là thể hiện của đối tượng **cửa sổ trình duyệt**
- Tồn tại khi mở 1 tài liệu HTML
- Sử dụng để truy cập thông tin của các đối tượng trên cửa sổ trình duyệt (tên trình duyệt, phiên bản trình duyệt, thanh tiêu đề, thanh trạng thái ...)

Đối tượng Window - DOM

- **Properties**

- document
- event
- history
- location
- name
- navigator
- screen
- status

- **Methods**

- alert
- confirm
- prompt
- blur
- close
- focus
- open

Đối tượng Window - DOM

- Ví dụ:

```
<html>
```

```
<body>
```

```
<script type="text/javascript">
```

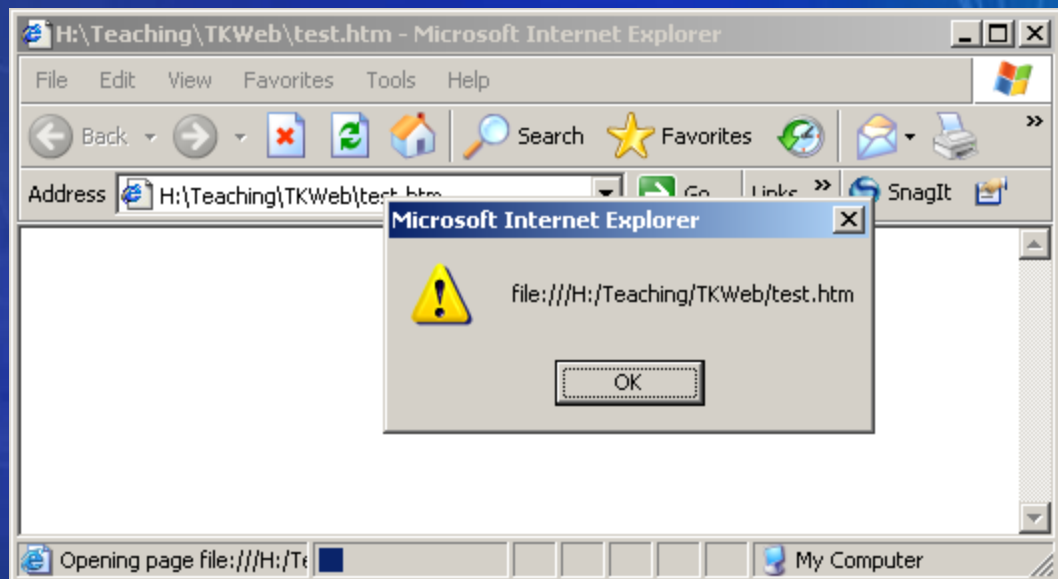
```
var curURL = window.location;
```

```
window.alert(curURL);
```

```
</script>
```

```
</body>
```

```
</html>
```



Đối tượng Document - DOM

- Biểu diễn cho **nội dung trang HTML** đang được hiển thị trên trình duyệt
- Dùng để lấy thông tin về tài liệu, các thành phần HTML và xử lý sự kiện



Đối tượng Document - DOM

● Properties

- aLinkColor
- bgColor
- **body**
- fgColor
- linkColor
- title
- **URL**
- vlinkColor
- forms[]
- images[]
- **childNodes[]**

● Methods

- close
- open
- **createTextNode(" text ")**
- **createElement("HTMLtag")**
- **getElementById("id")**
- ...

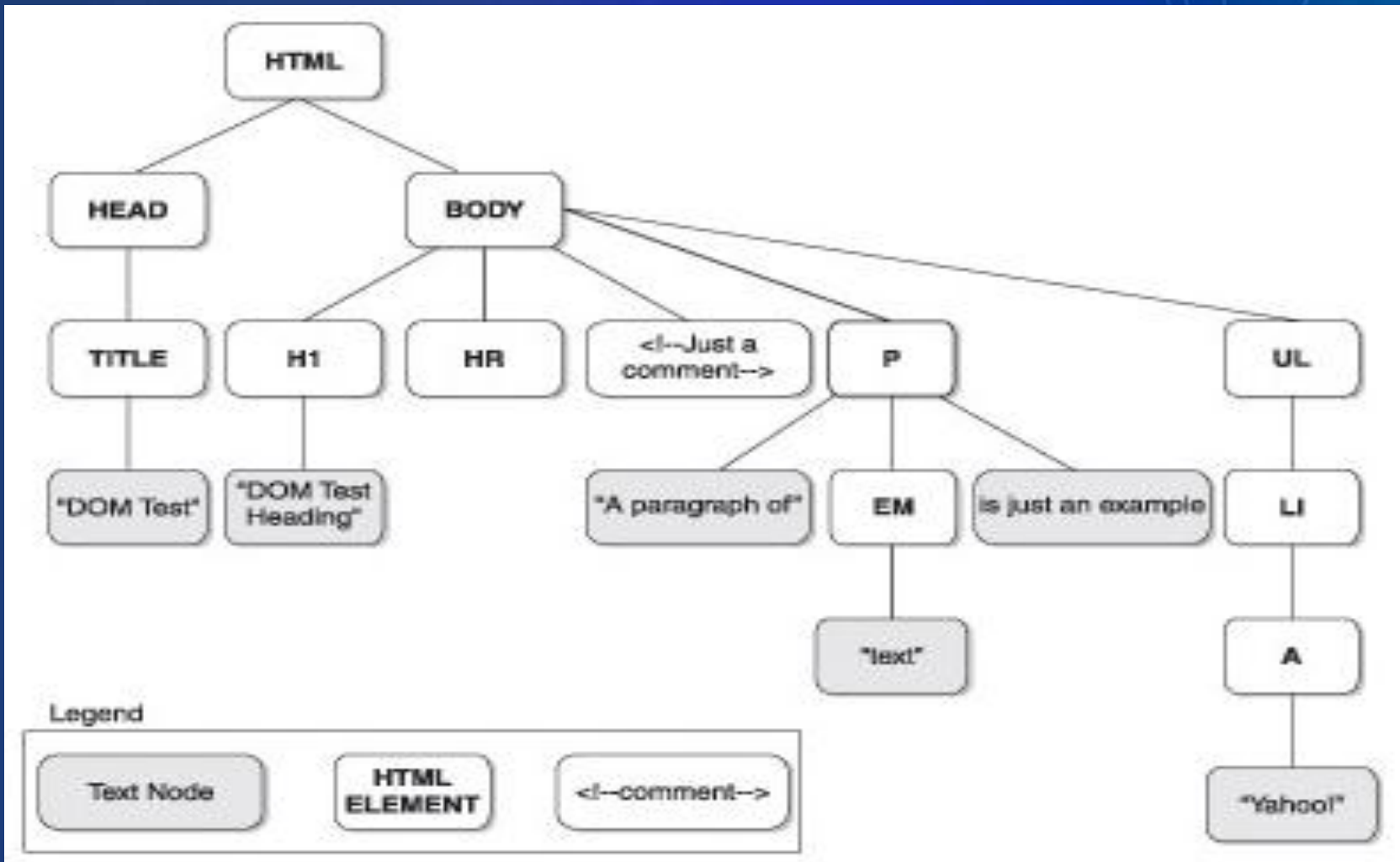
Đối tượng Document - DOM

- Biểu diễn nội dung của tài liệu theo cấu trúc cây

```
<html>
  <head>
    <title>DOM Test</title>
  </head>
  <body>
    <h1>DOM Test Heading</h1>
    <hr />
    <!-- Just a comment -->
    <p id="p1" >A paragraph of <em>text</em>
      is just an example</p>
    <ul>
      <li>
        <a href="http://www.yahoo.com" > Yahoo!
        </a>
      </li>
    </ul>
  </body>
</html>
```

Đối tượng Document - DOM

- Cấu trúc cây nội dung tài liệu



Đối tượng Document - DOM

- Các loại DOM Node chính

Node Type Number	Loại	Mô tả	Vi dụ
1	Element	(X)HTML or XML element	<code><p>...</p></code>
2	Attribute	Thuộc tính của HTML hay XML element	<code>align="center"</code>
3	Text	Nội dung chứa trong HTML or XML element	<code>This is a text fragment!</code>
8	Comment	HTML comment	<code><!-- This is a comment --></code>
9	Document	Đối tượng tài liệu gốc, thường là element nằm ở cấp cao nhất trong cây cấu trúc của tài liệu	<code><html></code>
10	DocumentType	Định nghĩa loại tài liệu	<code><!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd"></code>

Đối tượng Document - DOM

- getElementById (id1)

Trả về node có giá trị thuộc tính id = id1

Ví dụ:

```
//<p id="id1" >  
//   some text  
//</p>
```



Text Node

```
var node = document.getElementById("id1");  
var nodeName = node.nodeName; // p  
var nodeType = node.nodeType; // 1  
var nodeValue = node.nodeValue; // null  
var text      = node.innerText; // some text
```

Đối tượng Document - DOM

- createElement (nodeName)

Cho phép tạo ra 1 node HTML mới tùy theo đối số nodeName đầu vào

Ví dụ:

```
var imgNode = document.createElement("img");  
imgNode.src = "images/test.gif";
```

```
// 
```

Đối tượng Document - DOM

- `createTextNode (content)`

Ví dụ:

```
var textNode = document.createTextNode("New  
text");  
var pNode = document.createElement("p");  
pNode.appendChild(textNode);  
  
// <p>New text</p>
```

Đối tượng Document - DOM

- appendChild (newNode)

Chèn node mới **newNode** vào cuối danh sách các node con của một node.

Ví dụ:

```
//<p id="id1" >  
//   some text  
//</p>
```

```
var pNode = document.getElementById("id1");  
var imgNode = document.createElement("img");  
imgNode.src = "images/test.gif";  
pNode.appendChild(imgNode);
```

```
//<p id="id1" >  
//   some text  
//</p>
```

Đối tượng Document - DOM

- innerHTML

Chỉ định nội dung HTML bên trong một node.

Ví dụ:

```
//<p id="para1" >  
// some text  
//</p>
```

```
var theElement = document.getElementById("para1");  
theElement.innerHTML = "Some <b> new </b> text";
```

```
// Kết quả :  
// <p id="para1" >  
// Some <b> new </b> text  
// </p>
```


Đối tượng Document - DOM

- innerText

Tương tự innerHTML, tuy nhiên bất kỳ nội dung nào đưa vào cũng được xem như là text hơn là các thẻ HTML.

Ví dụ:

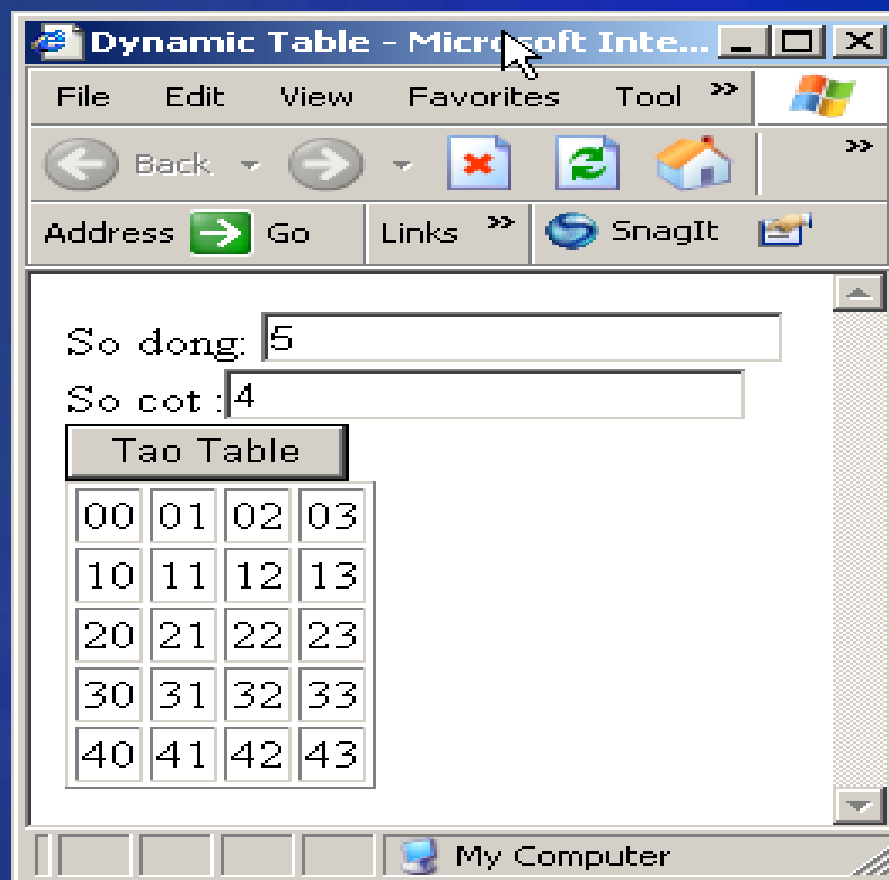
```
var theElement = document.getElementById("para1");
theElement.innerText = "Some <b> new </b> text";
// Kết quả hiển thị trên trình duyệt
// bên trong thẻ p: "Some <b> new </b> text"
```

Nội dung

- Giới thiệu về Javascript
- Kiểu dữ liệu & Các cú pháp Javascript
- Nhúng Javascript vào trang web
- Sử dụng các đối tượng trong Javascript
- Xử lý sự kiện
- DOM HTML với Javascript
- Ví dụ

Ví dụ: Dynamic Table

- Viết trang web cho phép tạo table có số dòng, số cột do người dùng nhập vào.



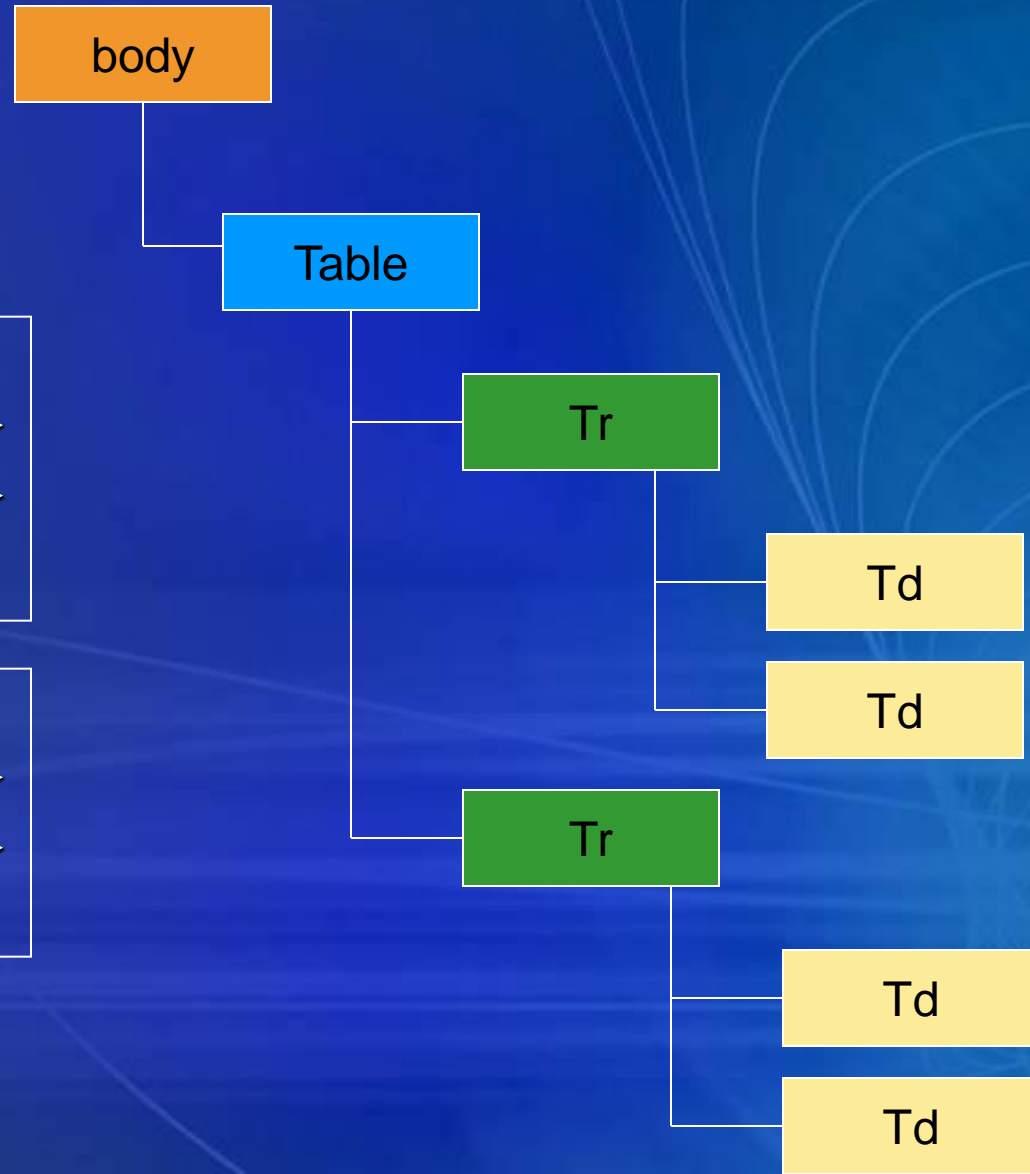
Ví dụ: Dynamic Table

<Table>

```
<Tr>  
  <td> 12 </td>  
  <td> 13 </td>  
</Tr>
```

```
<Tr>  
  <td> 21 </td>  
  <td> 22 </td>  
</Tr>
```

</Table>



Ví dụ:

body

Table

TBody

Tr

Td

Td

Tr

Td

Td

<Table>

<Tbody>

```
<Tr>  
  <td> 12 </td>  
  <td> 13 </td>  
</Tr>
```

```
<Tr>  
  <td> 21 </td>  
  <td> 22 </td>  
</Tr>
```

</Tbody>

</Table>

Ví dụ: Dynamic Table

- `Document.createElement(...)`: Tạo một đối tượng thẻ DOM HTML
- `Object.appendChild(...)`: Thêm một đối tượng thẻ DOM HTML như là nút con.

Ví dụ: Dynamic Table

```
function CreateTable(divTable)
{
    var tagTable = document.createElement("table");
    tagTable.border = 1;
    var tagTBody = document.createElement("tbody");
    tagTable.appendChild(tagTBody);

    var nDong = txtSoDong.value;
    var nCot = txtSoCot.value;

    for (i=0; i<nDong; i++)
    {
        var tagTR = document.createElement("tr");
        for (j=0; j<nCot; j++)
        {
            var tagTD = document.createElement("td");
            var textNode = document.createTextNode(i+" "+j);
            tagTD.appendChild(textNode);

            tagTR.appendChild(tagTD);
        }

        tagTBody.appendChild(tagTR);
    }

    divTable.appendChild(tagTable);
}
```

Lập trình Web 1



Khoa Công nghệ thông tin
Bộ môn Công nghệ phần mềm

PHP căn bản

Nguyễn Huy Khánh
nhkhanh@fit.hcmus.edu.vn

Nội dung

- Giới thiệu PHP
- Cơ chế hoạt động của WebServer
- Cú pháp & Quy ước trong PHP

Nội dung

- Giới thiệu PHP
- Cơ chế hoạt động của WebServer
- Cú pháp & Quy ước trong PHP

Giới thiệu về PHP – Lịch sử phát triển

- **PHP** : [Rasmus Lerdorf](#) in 1994 (được phát triển để phát sinh các form đăng nhập sử dụng giao thức HTTP của Unix)
- **PHP 2 (1995)** : Chuyển sang [ngôn ngữ script xử lý trên server](#). Hỗ trợ CSDL, Upload File, khai báo biến, mảng, hàm đệ quy, câu điều kiện, biểu thức, ...
- **PHP 3 (1998)** : Hỗ trợ ODBC, [đa hệ điều hành](#), giao thức email (SNMP, IMAP), bộ phân tích mã PHP (parser) của [Zeev Suraski](#) và [Andi Gutmans](#)
- **PHP 4 (2000)** : Trở thành một thành phần độc lập cho các webserver. Parse đổi tên thành [Zend Engine](#). Bổ sung các tính năng bảo mật cho PHP
- **PHP 5 (2005)** : Bổ sung Zend Engine II hỗ trợ [lập trình HĐT](#), [XML](#), [SOAP](#) cho Web Services, SQLite
- Phiên bản mới nhất của PHP là version [PHP 5.3.3](#) (www.php.net)



Giới thiệu về PHP – PHP là gì ?

- **PHP** viết tắt của **P**HP **H**ypertext **P**reprocessor
- Là ngôn ngữ server-side script, tương tự như ASP, JSP, ... thực hiện trên WebServer
- Tập tin PHP có đuôi là **.php**
- Cú pháp ngôn ngữ giống ngôn ngữ **C & Perl**

Ưu điểm
PHP ?



Giới thiệu về PHP – Ưu điểm 1

- PHP được sử dụng làm
 - Server Side Scripting
 - CommandLine Scripting (cron – Linux, Task Scheduler – Windows, Text Processing)
 - Xây dựng ứng dụng Desktop – PHP GTK

Giới thiệu về PHP – Ưu điểm 2

● Đa môi trường (Multi-Platform)

- **Web Servers:** Apache, [Microsoft IIS](#), Caudium, Netscape Enterprise Server
- **Hệ điều hành:** UNIX (HP-UX, OpenBSD, Solaris, Linux), Mac OSX, [Windows NT/98/2000/XP/2003/vista](#)
- **Hệ QTCSDL:** Adabas D, dBase, Empress, FilePro (read-only), Hyperwave, IBM DB2, Informix, Ingres, InterBase, FrontBase, mSQL, Direct MS-SQL, [MySQL](#), ODBC, Oracle (OCI7 and OCI8), Ovrimos, PostgreSQL, SQLite, Solid, Sybase, Velocis, Unix dbm

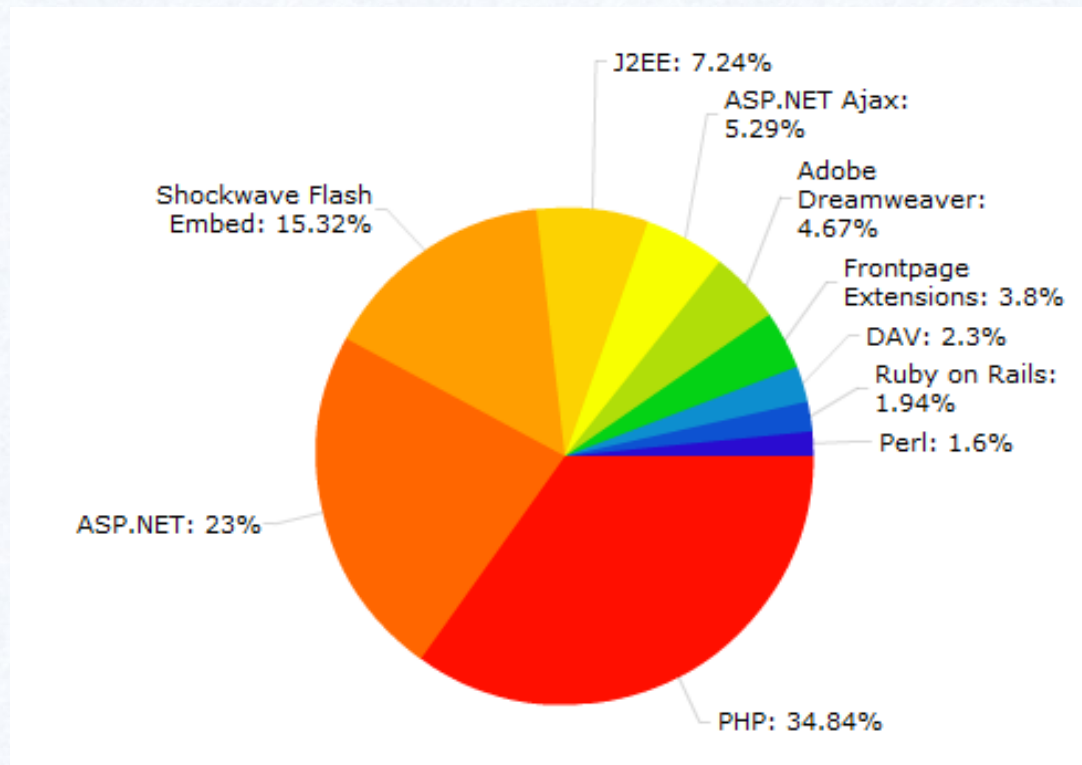
Giới thiệu về PHP – Ưu điểm 3

- **Miễn phí**

	PHP
Software	Free
Platform	Free (Linux)
Development Tools	Free (PHP Coder , jEdit , ...)

Giới thiệu về PHP – Ưu điểm 4

- PHP được sử dụng nhiều nhất trong số các server script



Theo <http://trends.builtwith.com/framework/PHP> vào ngày 01/10/2010

Giới thiệu về PHP – Một số website lớn



PHP at Yahoo!

<http://www.yahoo.com>

The Internet's most trafficked site



Portal



Portal



Course Management System



Wiki



Customer Relationship Management



e-Commerce



Portal



Bulletin Board



Content Management System



Help Desk

Giới thiệu về PHP – Cần gì để chạy PHP ?

● Download PHP

- Download PHP for free here:
<http://www.php.net/downloads.php>

● Download MySQL Database

- Download MySQL for free here:
<http://www.mysql.com/downloads/index.html>

● Download Apache Server

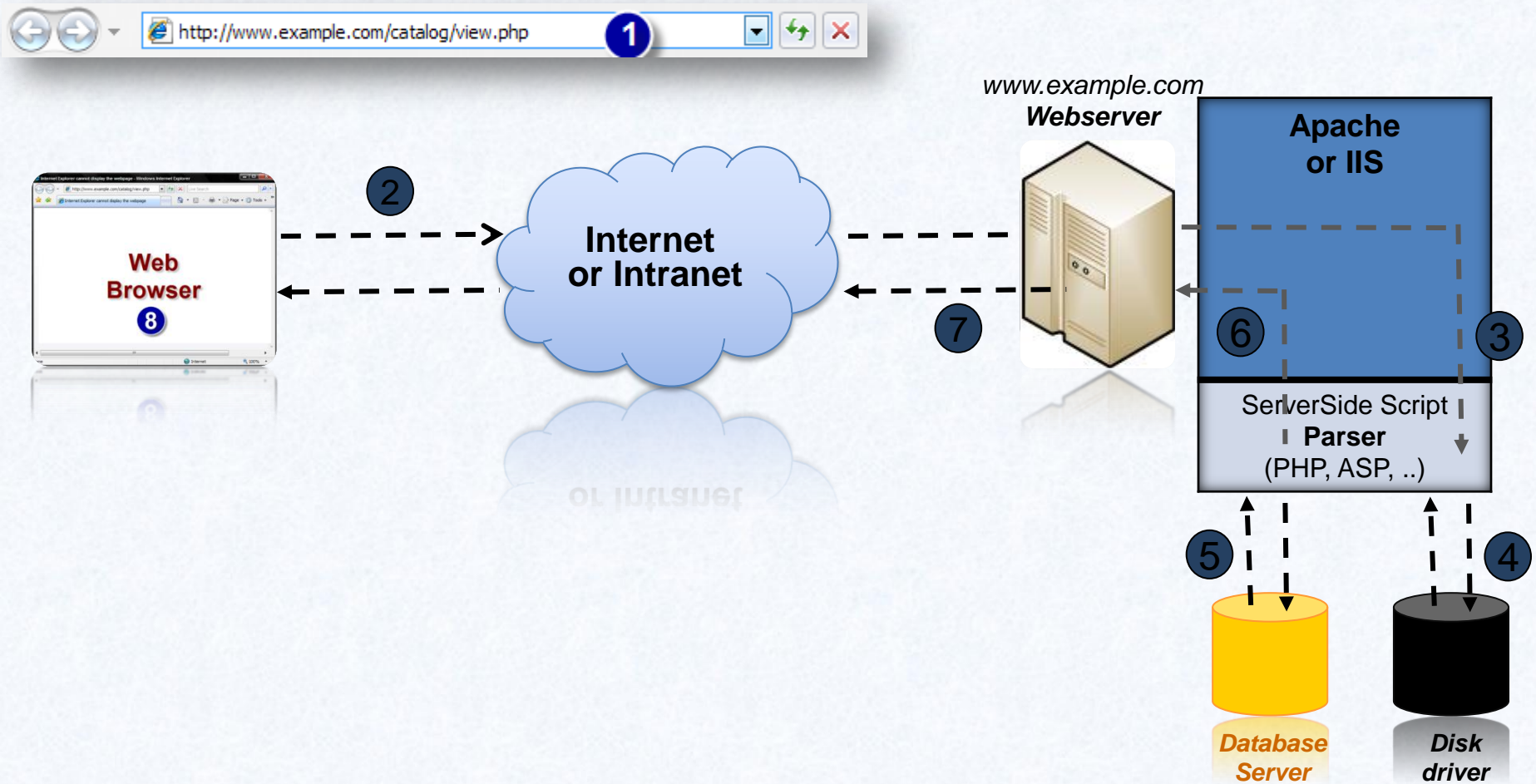
- Download Apache for free here:
<http://httpd.apache.org/download.cgi>

→ Download **WAMP, LAMP, XAMPP**

Nội dung

- Giới thiệu PHP
- Cơ chế hoạt động của WebServer
- Cú pháp & Quy ước trong PHP

Cơ chế hoạt động của WebServer



Cơ chế hoạt động của WebServer

```
1 <html>
2 <head>
3     <title>Test Server Script Parser</title>
4 </head>
5 <body>
6
7     <h1>Server Script Parser</h1>
8     Hello world HTML
9     <br />
10    <br />
11
12    <% response.write("Hello ASP Parser !!!") %>
13    <br />
14    <br />
15
16    <?php echo "Hello PHP Parser !!!" ?>
17    <br />
18    <br />
19
20 </body>
21 </html>
22
```

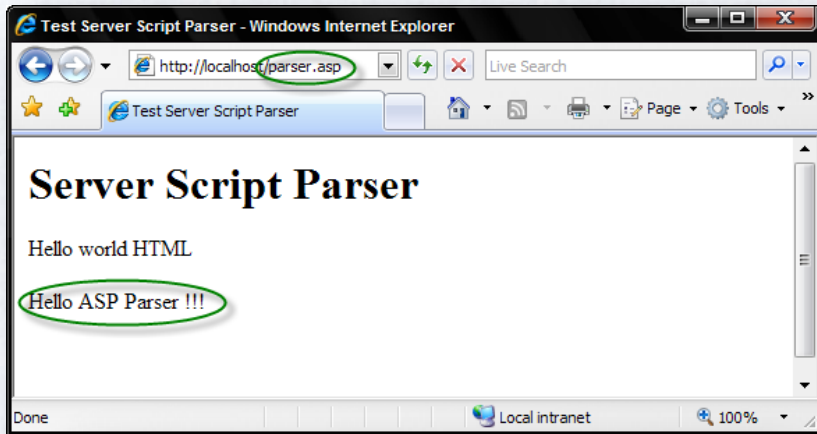
HTML Code

ASP Code

PHP Code

Cơ chế hoạt động của WebServer

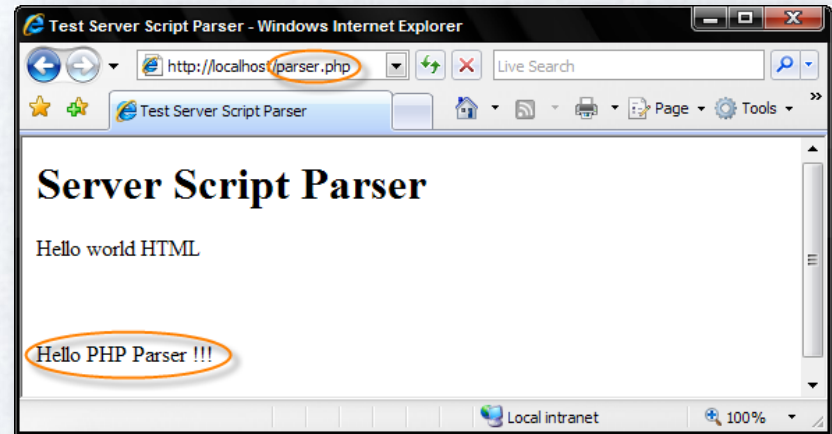
● Parser.asp



```
parser[1] - Notepad
File Edit Format View Help
<html>
<head>
  <title>Test Server Script Parser</title>
</head>
<body>

  <h1>Server Script Parser</h1>
  Hello world HTML
  <br />
  <br />
  Hello ASP Parser !!!
  <br />
  <br />
  <?php echo "Hello PHP Parser !!!" ?>
  <br />
  <br />
</body>
</html>
```

● Parser.php



```
parser[1] - Notepad
File Edit Format View Help
<html>
<head>
  <title>Test Server Script Parser</title>
</head>
<body>

  <h1>Server Script Parser</h1>
  Hello world HTML
  <br />
  <br />
  <% Response.Write("Hello ASP Parser !!!")%>
  <br />
  <br />
  Hello PHP Parser !!! <br />
  <br />
</body>
</html>
```

Nội dung

- Giới thiệu PHP
- Cơ chế hoạt động của WebServer
- Cú pháp & Quy ước trong PHP

Cú pháp & Quy ước trong PHP

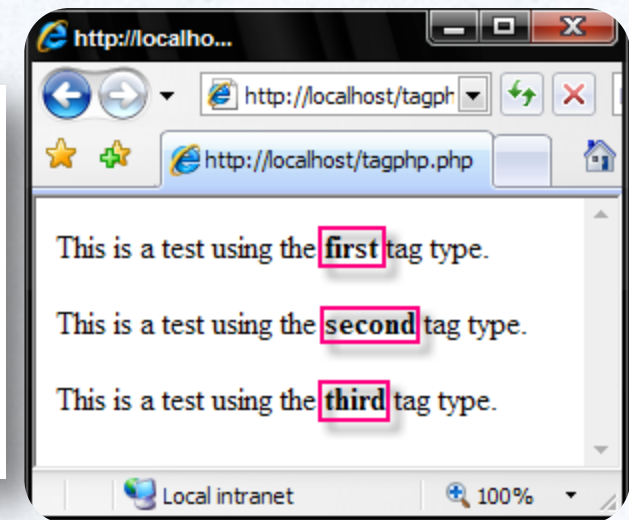
- Quy ước
- Khai báo biến
- Kiểu dữ liệu
- Toán tử
- Cấu trúc điều khiển
- Hàm
- Lớp đối tượng

Quy ước

- Mã lệnh **PHP** được đặt trong các cặp thẻ sau:

Thẻ mở	Thẻ đóng
<?>	?>
<?php	?>
<script language="php">	</script>

```
1 <?
2   echo "<P>This is a test using the <b>first</b> tag type.</P>";
3 <?>
4
5 <?php
6   echo "<P>This is a test using the <b>second</b> tag type.</P>";
7 <?>
8
9 <script language="php">
10  echo "<P>This is a test using the <b>third</b> tag type.</P>";
11 </script>
```



Quy ước

- Tất cả các câu lệnh php đều cách nhau bởi dấu “;”
- **Không phân biệt** khoảng trắng, Tab, xuống dòng trong câu lệnh

```
<?php print "Hello"; print " World!"; ?>
<?php
    Print "Hello"
        print " World!";
?>
```

- **Ghi chú** : Theo cú pháp ghi chú của C++ & Perl

// Đây là ghi chú

Đây là ghi chú

/* Đây là ghi
chú nhiều dòng*/

Cú pháp & Quy ước trong PHP

- Quy ước
- Khai báo biến
- Kiểu dữ liệu
- Toán tử
- Cấu trúc điều khiển
- Hàm
- Lớp đối tượng

Khai báo biến

`$ten_bien = value;`

● Không khai báo kiểu dữ liệu

● Biến tự động được khởi tạo ở lần đầu tiên gán giá trị cho biến

● Tên biến :

- Có thể bao gồm các Ký tự (A..Z, a..z), Ký số (0..9), _, \$
- **Không** được bắt đầu bằng ký số (0..9)
- **Phân biệt** chữ hoa – chữ thường

Ví dụ :

`$size` `$my_drink_size` `$_drinks`
`$drink4you`
`$$2hot4u` `$drink-size` **x**

Khai báo biến

● Variable variables

- Cho phép thay đổi tên biến

- Ví dụ:

```
$varname = "my_variable";
```

```
$$varname = "xyz"; // $my_variable = "xyz"
```

● Hằng số - Constants

- Ví dụ:

```
define("MY_CONST", 10);
```

```
echo MY_CONST;
```

Cú pháp & Quy ước trong PHP

- Quy ước
- Khai báo biến
- Kiểu dữ liệu
- Toán tử
- Cấu trúc điều khiển
- Hàm
- Lớp đối tượng

Kiểu dữ liệu

- boolean (bool)
- integer (int)
- double (float, real)
- string
- array
- object

1 Biến trong PHP có thể lưu bất kỳ kiểu dữ liệu nào.

Kiểu dữ liệu (tt)

● Chuyển kiểu dữ liệu

- Cách 1 (**automatic**)
\$var = "100" + 15;
\$var = "100" + 15.0;
\$var = 39 . " Steps";
- Cách 2: (**datatype**) \$var
- Cách 3: **settype**(\$var, "**datatype**")

\$var	(int)\$var	(bool)\$var	(string)\$var
null	0	false	""
true	1		"1"
false	0		""
"6 feet"	6	true	
"foo"	0	true	

Kiểm dữ liệu (tt)

- Kiểm tra kiểu dữ liệu

gettype

is_string

isset

is_integer

is_array

unset

is_double

is_object

empty

Ví dụ:

```
$var = "test";  
if (isset($var))  
    echo "Variable is Set";  
if (empty($var))  
    echo "Variable is Empty";
```

Kiểu số - int, float

● Một số hàm xử lý số

- abs
- ceil
- Floor
- round
- pow
- sqrt
- log
- log10
- decbin
- bindec
- dechex
- hexdec
- srand(seed)
- rand
- rand(min, max)
- ...

● Ví dụ

```
// Generate a seed
```

```
$seed = (float) microtime( ) * 100000000;
```

```
// Seed the pseudo-random number generator
```

```
srand($seed);
```

```
// Generate some random numbers
```

```
print rand(); // between 0 and getmaxrand( )
```

```
print rand(1, 6); // between 1 and 6 (inclusive)
```

Kiểu chuỗi - string

- Toán tử nối chuỗi : dấu chấm .

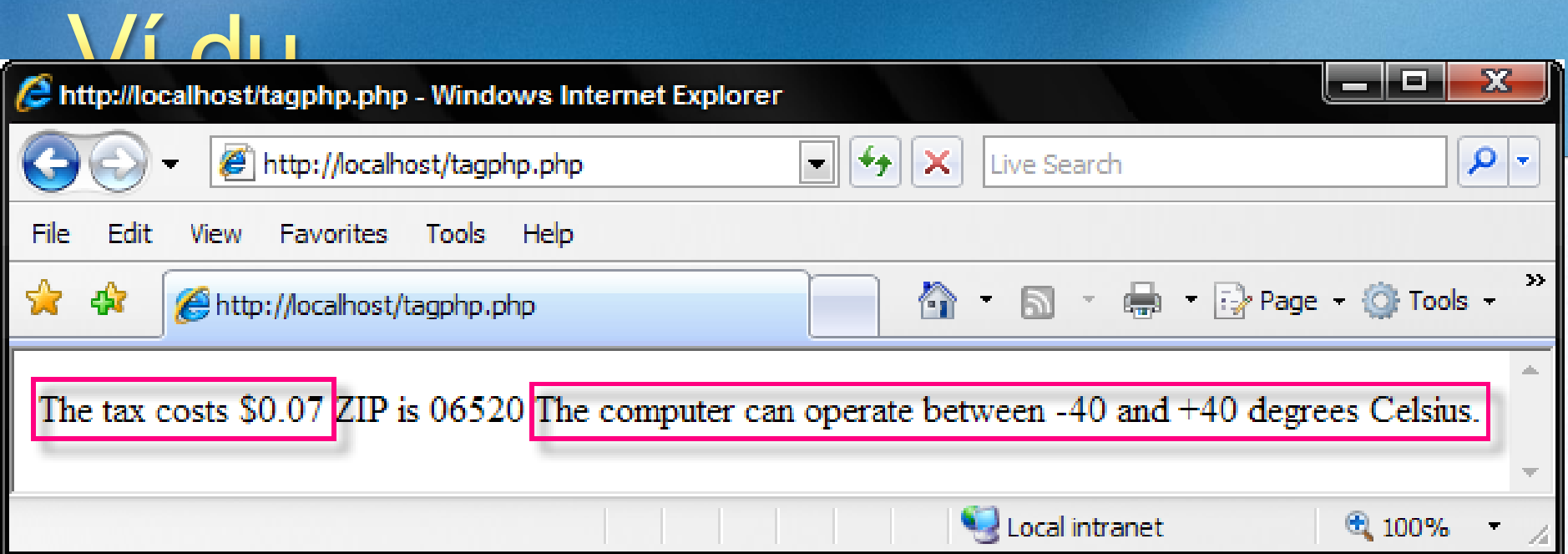
```
$s = "Hello" . " World"; // $s = "Hello World"
```

- Phân biệt dấu nháy đơn và nháy kép

```
$user = "Bill";  
print 'Hi $user'; // Hi $user  
print "Hi $user"; // Hi Bill  
print 'Hi' . $user; // ????  
print 'Hi' . '$user'; // ????
```

- Một số hàm xử lý chuỗi

● printf	trim	strtolower
● str_pad	str_replace	strtoupper
● strlen	substr	strcasecmp
● ...		



```
<?
$tax = 0.075;
printf('The tax costs $%.2f', $tax);

$zip = '6520';
printf("ZIP is %05d", $zip);

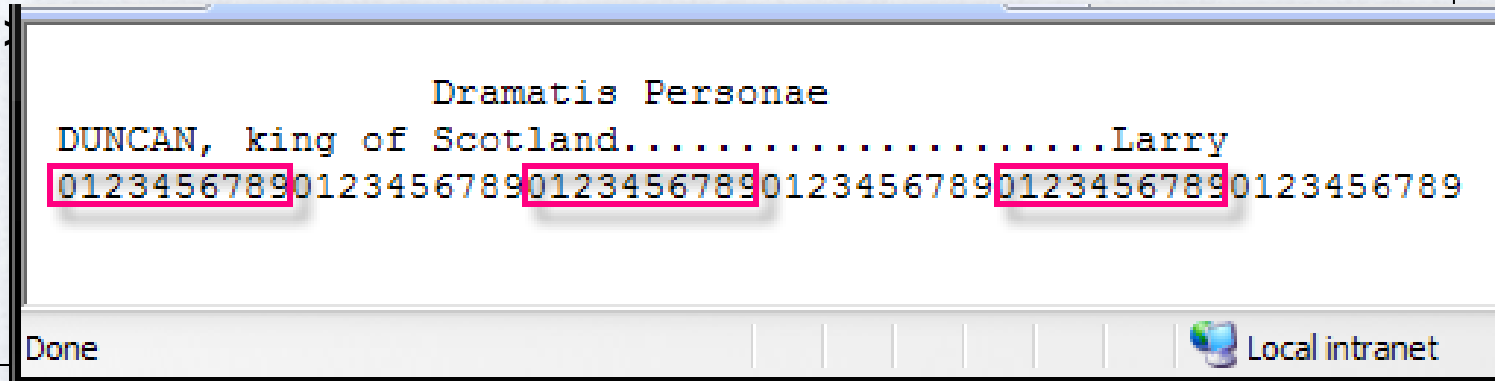
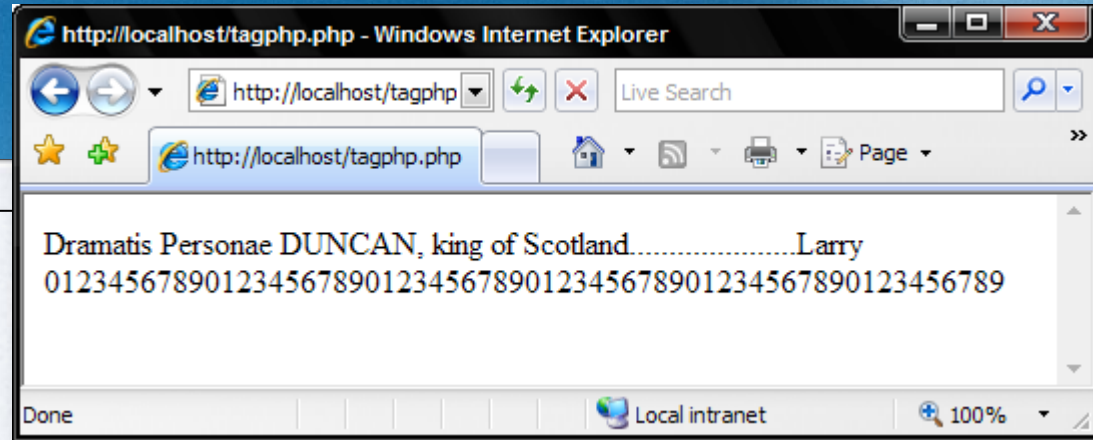
$min = -40; $max = 40;
printf("The computer can operate between %+d and %+d
degrees Celsius.", $min, $max);
?>
```

Ví dụ

```
<?
echo "<pre>"

// Print a heading
echo str_pad("Dramatis Personae", 50, " ", STR_PAD_BOTH) . "\n";

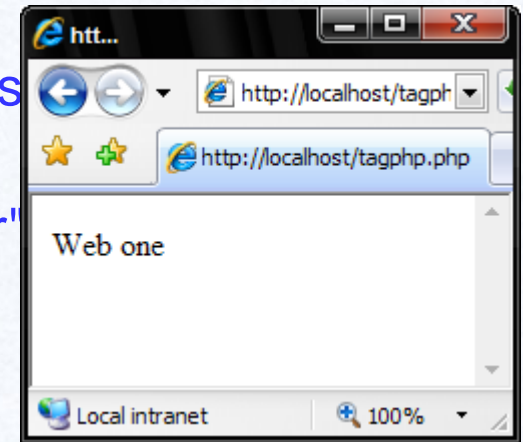
// Print an index line
echo  str_pad("DUNCAN, king of Scotland", 30, ".")
     . str_pad("Larry", 20, ".", STR_PAD_LEFT)
     . "\n";
echo "</pre>
?>
```



Mảng - array

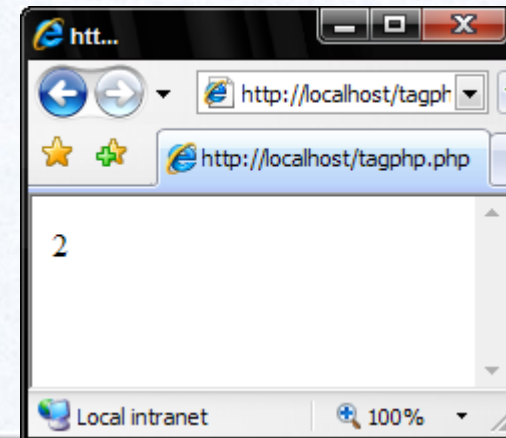
● Numbered array

```
$words = array("Web", "Database", "Applications");  
echo $words[0];  
$numbers = array(1=>"one", "two", "three", "four");  
echo $numbers[1];
```



● Associated array

```
$array = array("first"=>1, "second"=>2, "third"=>3);  
echo $array["second"];
```



Mảng - array

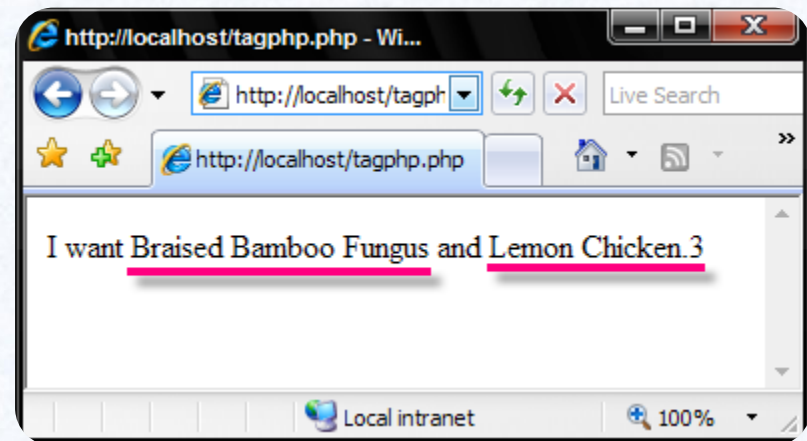
Một số hàm xử lý trên mảng

- `count` `is_array` `sort` `asort` `ksort` `usort`
- `min` `array_reverse` `rsort` `arsort` `krsort` `uasort`
- `max` `uksort`

Ví dụ:

```
$dinner = array( 'Sweet Corn and Asparagus',  
                'Lemon Chicken',  
                'Braised Bamboo Fungus');
```

```
sort($dinner);  
print "I want $dinner[0] and $dinner[1].";  
$dishes = count($dinner);  
print $dishes;
```



Mảng - array

- Một số hàm liên quan đến mảng
 - `reset(array)`
 - `array_push(array, elements)` : Thêm elements vào cuối mảng
 - `array_pop(array)` : Lấy phần tử cuối ra khỏi mảng
 - `array_unshift(array, elements)` : Thêm elements vào đầu mảng
 - `array_shift(array)` : Lấy phần tử đầu ra khỏi mảng
 - `array_merge(array, array)` : kết 2 mảng lại và trả ra mảng mới
 - `shuffle(array)` : Sort random mảng
 - `sort(array, flag)` : flag = {sort_regular, sort_numeric, sort_string, sort_locale_string}

Cú pháp & Quy ước trong PHP

- Quy ước
- Khai báo biến
- Kiểu dữ liệu
- Toán tử
- Cấu trúc điều khiển
- Hàm
- Lớp đối tượng

Toán tử

Loại	Toán tử	Ghi chú
	new .	
	. [] ()	
Toán học	+ - * / % ++ --	
So sánh	< > <= >= != == === !==	
Luận lý	&& ?: ,	
Xử lý bit	! ~ << >> >>> AND OR XOR	
Gán	= += -= *= /= %= >>= <<= &= = ^= .=	
Ép kiểu	(kiểu dữ liệu)	(int) (double) (string)...

Cú pháp & Quy ước trong PHP

- Quy ước
- Khai báo biến
- Kiểu dữ liệu
- Toán tử
- Cấu trúc điều khiển
- Hàm
- Lớp đối tượng

Cấu trúc điều khiển

- Điều kiện **if**
- Điều khiển **switch**
- Vòng lặp **for**
- Vòng lặp **while**
- Vòng lặp **do.. While**
- Vòng lặp **foreach**

- Từ khóa **break, continue**

Điều kiện if

```
if (condition)
{
    statement[s] if true
}
else (condition)
{
    statement[s] if false
}
```

Ví dụ:

```
$x = 5;
```

```
if ($x < 4)
```

```
    echo "$x is less than 4";
```

```
else
```

```
    print '$x isn't less than 4';
```

\$x isn't less than 4

Điều khiển switch

Ví dụ:

```
switch (expression)
{
    case label :
        statementlist
        break;
    case label :
        statementlist
        break;
    ...
    default :
        statementlist
}
```

```
$menu = 3;
switch ($menu) {
    case 1:
        echo "You picked one";
        break;
    case 2:
        echo "You picked two";
        break;
    case 3:
        echo "You picked three";
    case 4:
        echo "You picked four";
        break;
    default:
        echo "You picked another
option";
}
```

You picked three You picked four

Vòng lặp for

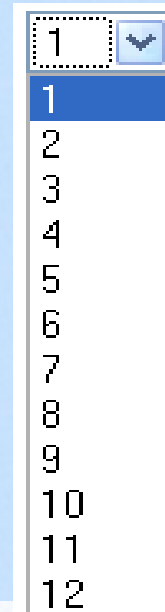
```
for ([initial expression]; [condition]; [update expression])  
{  
    statement[s] inside loop  
}
```

- **Ví dụ:**

```
print "<select>";
```

```
for ($i = 1; $i <= 12; $i++) {  
    print "<option>$i</option>";  
}
```

```
print "</select>";
```



1	▼
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	

Vòng lặp while, do...while

```
while (expression)
{
    statements
}
```

```
do
{
    statements
}while (expression);
```

Ví dụ:

```
$i = 1; $j = 9;
```

```
while ($i <= 10) {
```

```
    $temp = $i * $j;
```

```
    print "$j * $i = $temp<br>";
```

```
    $i++;
```

```
}
```

9	x	1	=	9
9	x	2	=	18
9	x	3	=	27
9	x	4	=	36
9	x	5	=	45
9	x	6	=	54
9	x	7	=	63
9	x	8	=	72
9	x	9	=	81
9	x	10	=	90

Vòng lặp foreach

```
foreach (array as variable)
{
    statements
}
```

Ví dụ:

breakfast	Walnut Bun
lunch	Cashew Nuts and White Mushrooms
dinner	Eggplant with Chili Sauce

```
$meal = array('breakfast' => 'Walnut Bun',
              'lunch' => 'Cashew Nuts and White Mushrooms',
              'dinner' => 'Eggplant with Chili Sauce');
```

```
print "<table border='1'>\n";
foreach ($meal as $key => $value) {
    print "<tr><td>$key</td><td>$value</td></tr>\n";
}
print '</table>';
```

Cú pháp & Quy ước trong PHP

- Quy ước
- Khai báo biến
- Kiểu dữ liệu
- Toán tử
- Cấu trúc điều khiển
- Hàm
- Lớp đối tượng

Hàm - function

```
function functionName ([parameter1]...[,parameterN])  
{  
    statement[s] ;  
}
```

```
function functionName ([parameter1]...[,parameterN])  
{  
    statement[s] ;  
    return .... ;  
}
```

Hàm – Phạm vi biến

```
<?php  
function doublevalue($var=5)  
{  
    global $temp;  
    $temp = $var * 2;  
}
```

```
$temp = 5;  
doublevalue();  
echo "\$temp is: $temp";  
>
```

\$temp is:

\$temp is: 10

Hàm – Tham trị vs Tham biến

```
<?php
function doublevalue( $var)
{
    $var = $var * 2;    &
}

```

```
$variable = 5;
doublevalue($variable);
echo "\$variable is: $variable";
?>
```

\$variable is: 5

\$variable is: 10

Hàm – include & require

```
// functions.inc
<?php
function bold($string)
{
    echo "<b>" . $string . "</b>\n";
}
?>
```

```
// index.php
<html>
<head>
    <title>Simple Function Call</title>
</head>
<body bgcolor="#ffffff">
<?
require "functions.inc";

bold("this is bold");

$myString = "this is bold";
bold($myString);
?>
</body></html>
```

Cú pháp & Quy ước trong PHP

- Quy ước
- Khai báo biến
- Kiểu dữ liệu
- Toán tử
- Cấu trúc điều khiển
- Hàm
- Lớp đối tượng

Lớp đối tượng - class

```
class class_name() [extends superclass_name]
{
    var $attribute;
    ...
    function method_name()
    {
        $this->attribute = ...;
    }
    ...
}
$a = new class_name (...);
```


Lớp đối tượng - class

- Ví dụ:

```
class Counter {  
    var $count = 0;  
    var $startPoint = 0;  
  
    function increment( ) {  
        $this->count++;  
    }  
}  
  
$aCounter = new Counter;  
$aCounter->increment( );  
  
echo $aCounter->count; // prints 1  
$aCounter->count = 101;
```

Lập trình Web 1



Khoa Công nghệ thông tin
Bộ môn Công nghệ phần mềm

Web form & PHP

Nguyễn Huy Khánh
nhkhanh@fit.hcmus.edu.vn

Mục tiêu



- Hiểu được cách thức gửi dữ liệu đến web server thông qua web form
- Biết cách sử dụng các thẻ của web form
- Hiểu và phân biệt hai cách truyền dữ liệu GET và POST
- Biết cách nhận dữ liệu GET/POST bằng PHP

Nội dung



Giới thiệu về Form

Các thành phần của Form

Một số thuộc tính tiện ích của Form và Input

Phân biệt phương thức GET/POST

Cơ chế Truyền và Nhận dữ liệu giữa các trang web

Truyền/Nhận qua phương thức GET

Truyền/Nhận qua phương thức POST

Ví dụ

Giới thiệu Form

- Được dùng để nhận dữ liệu từ phía người dùng
- Giúp gửi yêu cầu của người dùng đến trang xử lý trong ứng dụng web
- Thẻ `<form>` dùng để chứa các thành phần khác của form
- Những thành phần nhập liệu được gọi là Form Field
 - text field
 - password field
 - multiple-line text field
 -

Thẻ <Form>

- Là container chứa các thành phần nhập liệu khác.

```
<FORM NAME="..." ACTION="..." METHOD="...">  
    <!-- các thành phần của Form -->  
</FORM>
```

- Các thuộc tính của </FORM>

- **NAME** : tên FORM
- **ACTION** : chỉ định trang web nhận xử lý dữ liệu từ FORM này khi có sự kiện click của button SUBMIT.
- **METHOD** : Xác định phương thức chuyển dữ liệu (POST,GET)

Thẻ <Form> - Ví dụ

Dangnhap.htm

```
<html>
  <body>
    <form Name="Dangnhap"
          Action="/admin/xlDangnhap.php"
          Method="Post">
      .....
    </form>
  </body>
</html>
```


Nội dung



Giới thiệu về Form

Các thành phần của Form

Một số thuộc tính tiện ích của Form và Input

Phân biệt phương thức GET/POST

Cơ chế Truyền và Nhận dữ liệu giữa các trang web

Truyền/Nhận qua phương thức GET

Truyền/Nhận qua phương thức POST

Ví dụ

Các thành phần của Form

- Gồm các loại Form Field sau:
 - Text field
 - Password field
 - Hidden Text field
 - Check box
 - Radio button
 - File Form Control
 - Submit Button, Reset Button, Generalized Button
 - Multiple-line text field
 - Label
 - Pull-down menu
 - Scrolled list
 - Field Set

Các thành phần của Form – Text Field

- Dùng để nhập một dòng văn bản
- Cú pháp

```
<INPUT
  TYPE           = "TEXT"
  NAME           = string
  READONLY
  SIZE           = variant
  MAXLENGTH     = long
  TABINDEX      = integer
  VALUE         = string
  .....
>
```

Các thành phần của Form – Password Field

- Dùng để nhập mật khẩu
- Cú pháp

```
<INPUT  
  TYPE           = "PASSWORD"  
  NAME           = string  
  READONLY  
  SIZE           = variant  
  MAXLENGTH      = long  
  TABINDEX       = integer  
  VALUE          = string  
  .....
```

password field :

- Ví dụ

```
<input type="Password" name="txtPassword" value="123456abc1234"  
  size="20" maxlength="30">
```

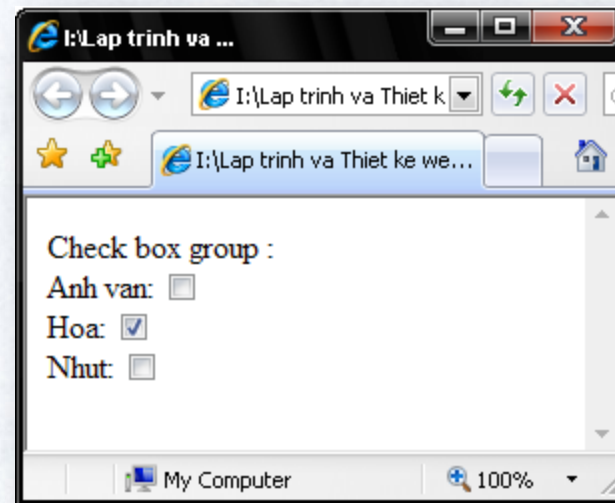
Các thành phần của Form – Check box

Cú pháp

```
<input  
  TYPE      = "checkbox"  
  NAME      = "text"  
  VALUE     = "text"  
  [checked]  
>
```

Ví dụ

```
<html>  
  <body>  
    Check box group : <br>  
    Anh van: <input type="checkbox" name="Languages1" value="En"><br>  
    Hoa: <input type="checkbox" name="Languages2" value="Chz" checked><br>  
    Nhut: <input type="checkbox" name="Languages3" value="Jp"><br>  
  </body>  
</html>
```



Các thành phần của Form – Radio button

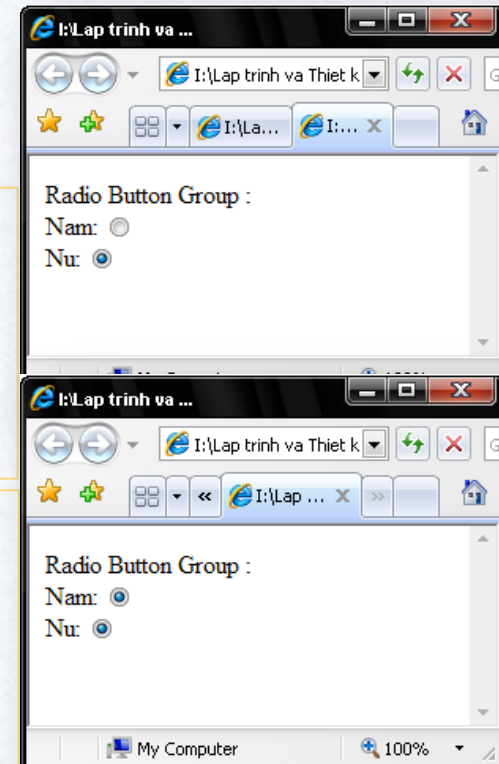
Cú pháp

```
<input  
  TYPE      = "radio"  
  NAME      = "text"  
  VALUE     = "text"  
  [checked]  
>
```

Ví dụ

```
<html>  
  <body>  
    Radio Button Group : <br>  
    Nam: <input type="radio" name="sex" value="nam" checked><br>  
    Nu: <input type="radio" name="sex" value="nu" checked ><br>  
  </body>  
</html>
```

```
<html>  
  <body>  
    Radio Button Group : <br>  
    Nam: <input type="radio" name="sex1" value="nam" checked><br>  
    Nu: <input type="radio" name="sex2" value="nu" checked ><br>  
  </body>  
</html>
```



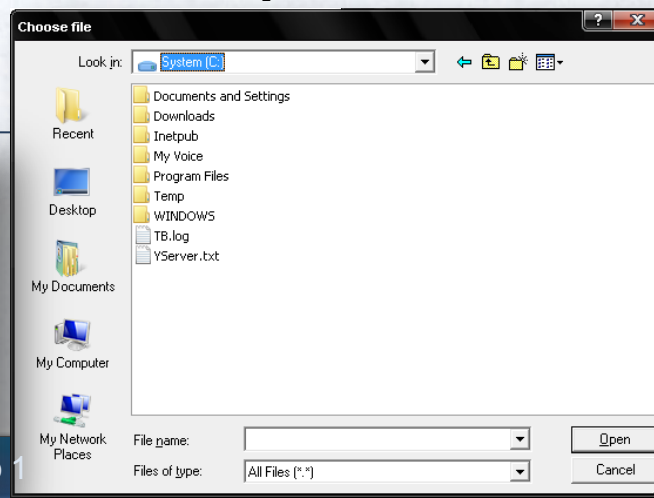
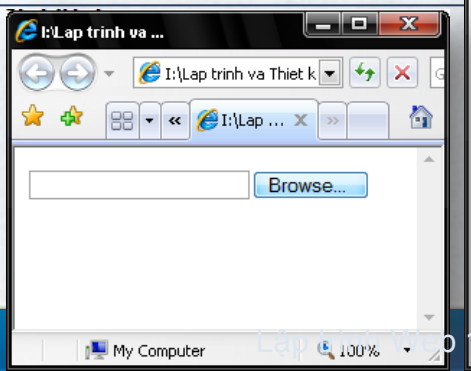
Các thành phần của Form – File Form Control

- Dùng để upload 1 file lên server
- Cú pháp

```
<form action="..." method="post" enctype="multipart/form-data"
name="...">
    <input TYPE="FILE" NAME="...">
</form>
```

- Ví dụ

```
<html>
<body>
    <form name="frmMain" action="POST" enctype="multipart/form-data">
        <input type="file" name="fileUpload">
    </form>
</body>
</html>
```



Các thành phần của Form – Submit button

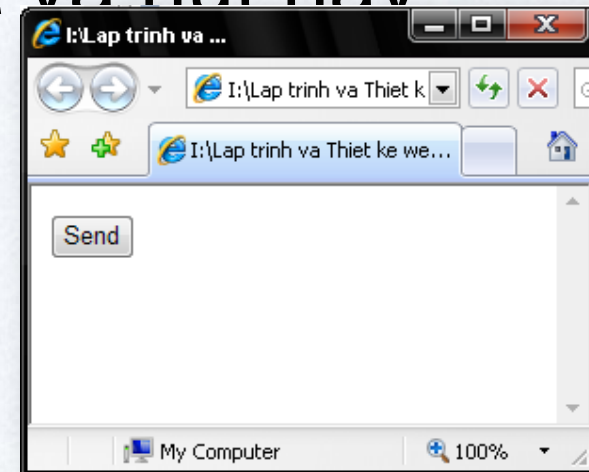
- Nút phát lệnh và gửi dữ liệu của form đến trang xử lý.
- Mỗi form chỉ có **một** nút submit và nút này **được viền đậm**

- Cú pháp:

```
<input TYPE="submit" name="..." value="...">
```

- Ví dụ:

```
<input type="submit" name="btnSend" value="Send">
```



Form – Reset Button

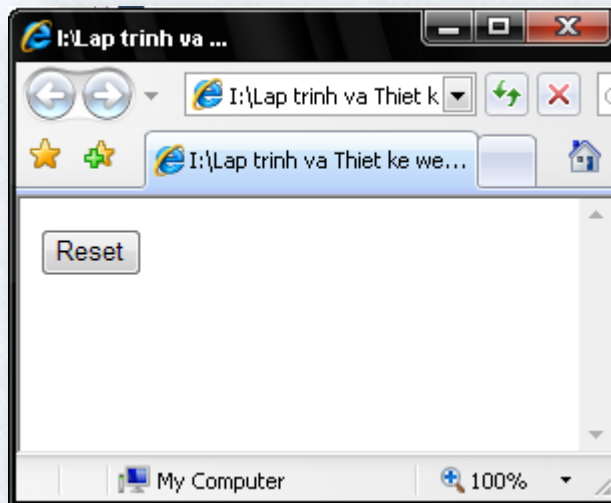
- Dùng để trả lại giá trị mặc định cho các control khác trong form

- Cú pháp

```
<input TYPE="reset" name="..." value="...">
```

- Ví dụ

```
<input type="reset" name="btnReset" value="Rest">
```



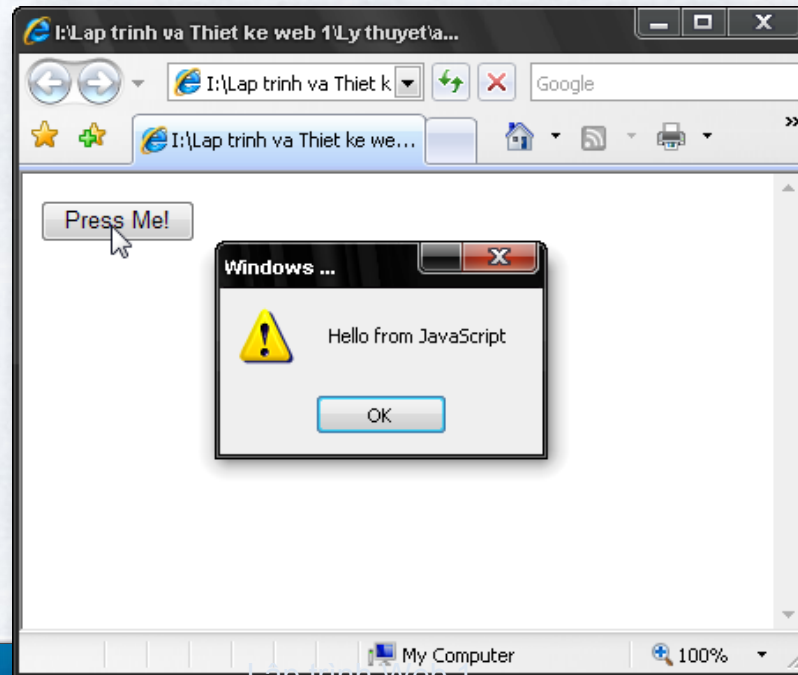
Form - Generalized Button

● Cú pháp

```
<input type="button" name="..." value="..." onclick="script">
```

● Ví dụ

```
<input type="button" name="btnNormal" value="Press Me!"  
onclick="alert('Hello from JavaScript');" >
```



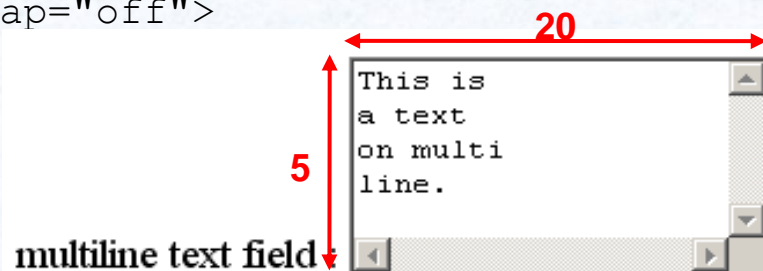
Form – Multiline Text Field

- Dùng để nhập văn bản nhiều dòng
- Cú pháp

```
<TEXTAREA  
  COLS           = long  
  ROWS           = long  
  DISABLED  
  NAME           = string  
  READONLY  
  TABINDEX       = integer  
  WRAP           = OFF | PHYSICAL | VIRTUAL> .....  
</TEXTAREA>
```

• Ví dụ

```
<textarea cols="7" rows="5" wrap="off">  
  This is a text on multiline.  
</textarea>
```



Form - Label

- Dùng để gán nhãn cho một Form Field, đồng thời nhận chuyển các sự kiện nhận được đến form field.
- Cú pháp

```
<LABEL  
  FOR = IDString  
  CLASS=string  
  STYLE=string  
>
```

- Ví dụ

```
<label for="Languages">Anh văn: </label>  
<input type="checkbox" name="Languages" id="Languages" value="Eng">
```

Anh văn:

Form – Pull-down Menu

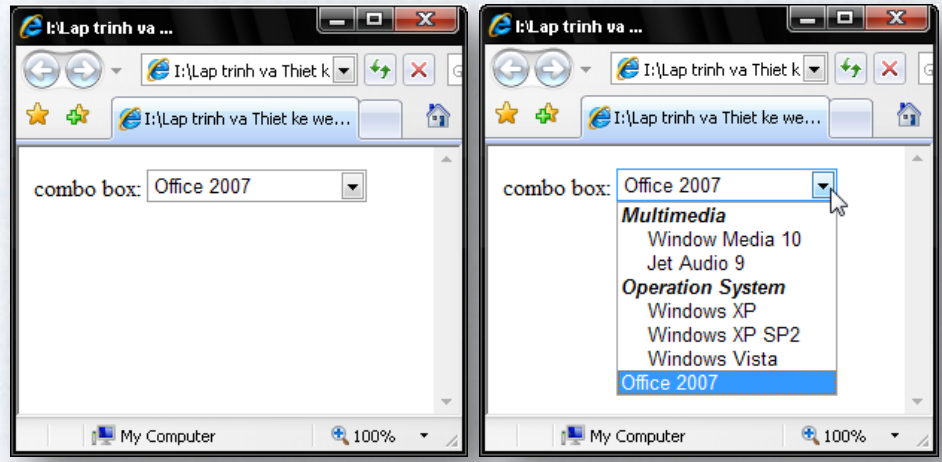
- Dùng để tạo ra một combo box
- Cú pháp

```
<Select name="...">
  <optgroup label="...">
    <option [selected] value="..." >.....</option>
    .....
  </optgroup>

  <option [selected] value="..." >.....</option>
  .....
</select>
```

Form – Pull-down Menu

```
<html>
  <body>
    combo box:
    <select name="DSSoftware">
      <optgroup label="Multimedia">
        <option value="WM10">Window Media 10</option>
        <option value="JA9">Jet Audio 9</option>
      </optgroup>
      <optgroup label="Operation System">
        <option value="WXP">Windows XP</option>
        <option value="WXPSP2">Windows XP SP2</option>
        <option value="WVT">Windows Vista</option>
      </optgroup>
      <option selected value="Office07">Office 2007</option>
    </select>
  </body>
</html>
```

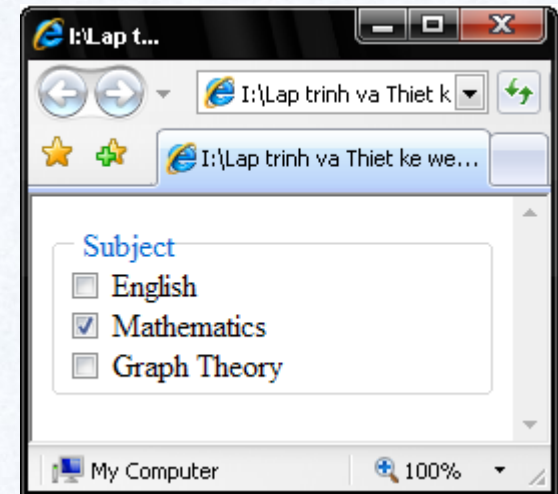


Form – Field Set

- Dùng để tạo ra Group box, nhóm các thành phần nhập liệu trong form

- Cú pháp

```
<fieldset>
  <legend>GroupBox's Name</legend>
  <input .....>
  ...
</fieldset>
```



- Ví dụ

```
<html>
<body>
  <fieldset>
    <legend>Subject</legend>
    <input type="checkbox" name="Subjects" value="Eng"> English<br>
    <input type="checkbox" name="Subjects" value="Math" checked> Mathematics<br>
    <input type="checkbox" name="Subjects" value="GraphTheory"> Graph Theory<br>
  </fieldset>
</body>
</html>
```

Nội dung



Giới thiệu về Form

Các thành phần của Form

Một số thuộc tính tiện ích của Form và Input

Phân biệt phương thức GET/POST

Cơ chế Truyền và Nhận dữ liệu giữa các trang web

Truyền/Nhận qua phương thức GET

Truyền/Nhận qua phương thức POST

Ví dụ

Thuộc tính của Form và Input

The screenshot shows a Yahoo! Mail inbox in Internet Explorer. A context menu is open over an email, listing actions like 'Open', 'Print', 'Reply to Sender', etc., with keyboard shortcuts. The email list includes columns for From, Subject, Date, and Size. The context menu items and their shortcuts are:

Action	Shortcut
Open	Enter
Open into New Window	Shift+Enter
Print	Ctrl+p
Reply to Sender	r
Reply to All	a
Forward	f
Select All Messages	Ctrl+a
Mark as Read	k
Mark as Unread	Shift+k
Flag for Follow-up	l
Clear Flag	Shift+l
Delete	
This is Spam	
Add Sender to Contacts	
View Full Headers	

The email list contains the following entries:

From	Subject	Date	Size
trung tran	[CrSe:48] ebook sensor network	Sat, 10/6/07 10:31 AM	4KB
Tran Minh Triet	[CrSe:47] Chuong trinh Mindjet MindMane	Fri, 10/5/07 11:35 PM	48KB
Tran Minh Triet	[CrSe:46] Mot website tong hop thc	Fri, 10/5/07 11:08 PM	6KB
Tran Duy Thao	Tong hop HDTH bo mon CNPM	Fri, 9/14/07 4:15 AM	259KB
chi ngoc	templates WORD!	Sat, 9/1/07 12:24 PM	1.4MB
chi ngoc	templates	Sat, 9/1/07 12:17 PM	58KB
Yahoo! 360 Alerts	Connect to andytriv	Fri, 8/24/07 3:15 PM	26KB
Yahoo! 360 Alerts	Open	Fri, 8/24/07 1:26 PM	15KB
Yahoo! 360 Alerts	Open into New Window	Wed, 8/22/07 4:19 PM	26KB
CodeGuru Upd	Print	Wed, 8/22/07 12:01 AM	14KB
information@a	Reply to Sender	Mon, 8/20/07 10:01 AM	49KB
Admin Thodia	Reply to All	Wed, 8/8/07 11:13 PM	7KB
Yahoo! 360 Alerts	Forward	Wed, 8/8/07 10:31 PM	30KB

adconion
MEDIA GROUP

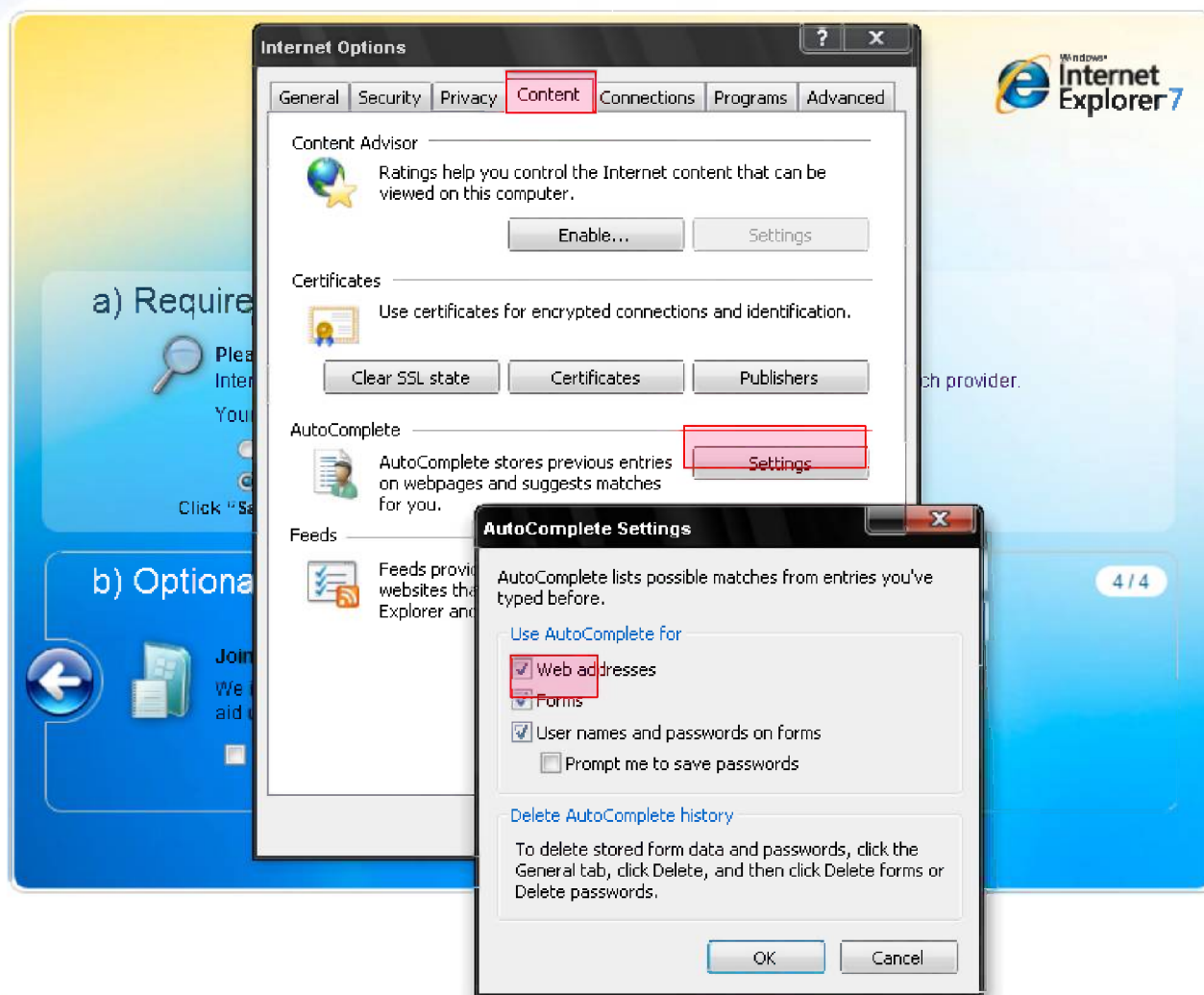
One Of The
World's Largest
Independent
Ad Networks.

High Quality.
High Reach.

Thuộc tính của Form và Input

- **Accesskey=char**
 - Tạo phím nóng cho form fields.
 - Áp dụng cho tất cả form fields.
 - Cách nhấn **Alt + char**
 - Tránh các phím tắt của browser.
- **Title = string**
 - Tạo tooltip cho form fields.
 - Áp dụng cho tất cả form fields.
- **Autocomplete = ON/OFF**
 - Gợi ý tự động khi nhập liệu.
 - Áp dụng cho tất cả tag form, input.

Bật chế độ Auto-complete cho IE



Nội dung



Giới thiệu về Form

Các thành phần của Form

Một số thuộc tính tiện ích của Form và Input

Phân biệt phương thức GET/POST

Cơ chế Truyền và Nhận dữ liệu giữa các trang web

Truyền/Nhận qua phương thức GET

Truyền/Nhận qua phương thức POST

Ví dụ

Phân biệt GET/POST - GET

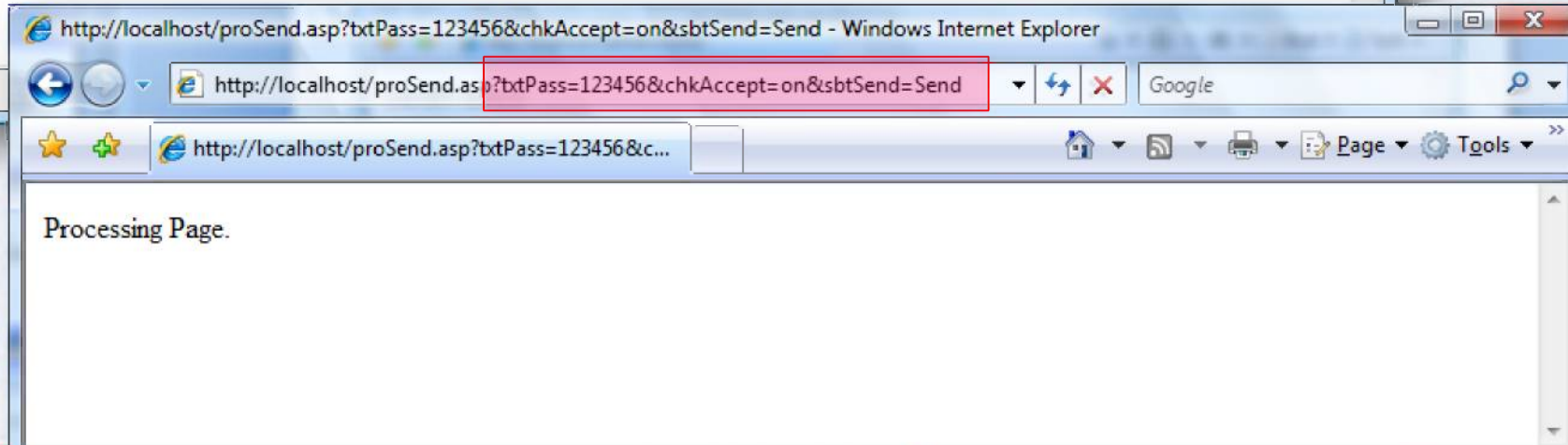
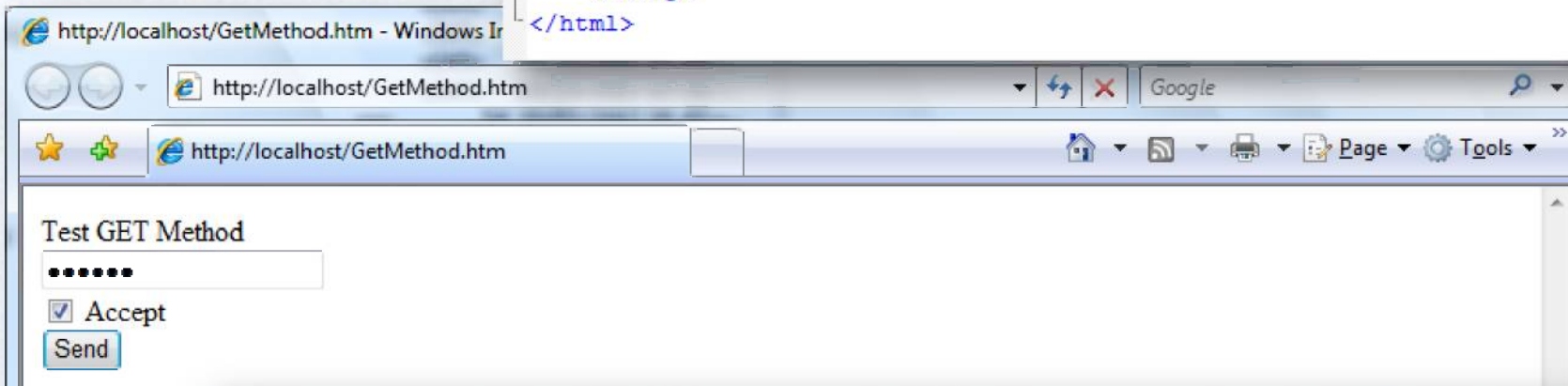
- Các đối số của Form được ghi chèn vào đường dẫn URL của thuộc tính action trong tag <Form>
- Khối lượng dữ liệu đối số được truyền đi của Form bị giới hạn bởi chiều dài tối đa của một URL trên Address bar. (tối đa của một URL là 2048 bytes)

Phân biệt GET/POST - POST

- Các đối số của Form được truyền “ngầm” bên dưới
- Khối lượng dữ liệu đối số được truyền đi của Form **không** phụ thuộc vào URL → Không bị giới hạn

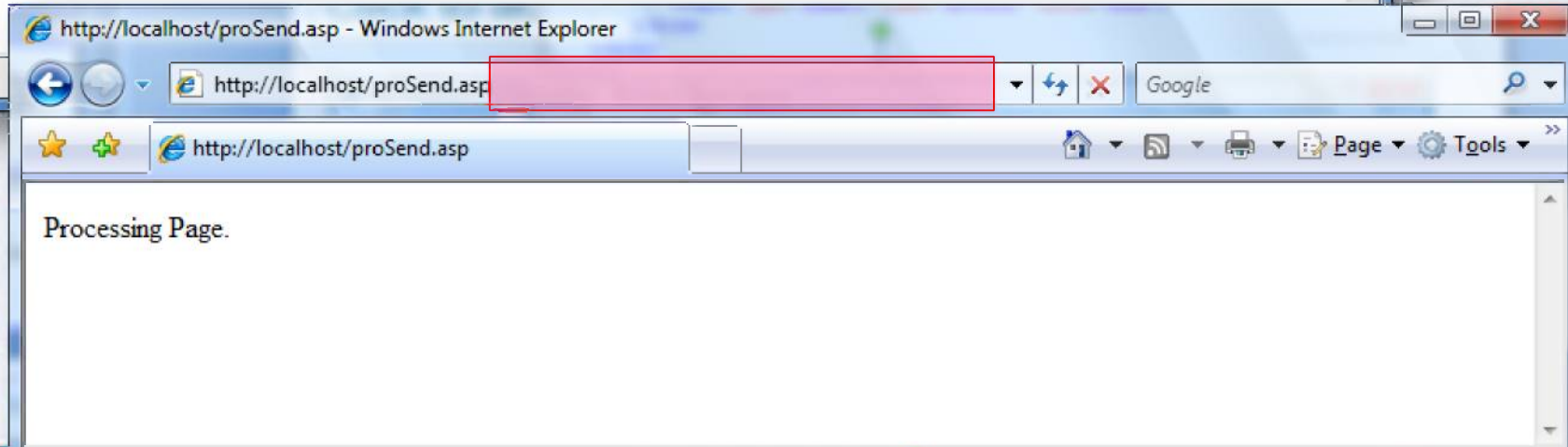
Ví dụ về GET

```
<html>
<body>
  <form action="proSend.asp" method="GET">
    Test GET Method <br>
    <input type="password" name="txtPass" value="123456"><br>
    <input type="checkbox" name="chkAccept"> Accept<br>
    <input type="Submit" name="sbtSend" value="Send">
  </form>
</body>
</html>
```



Ví dụ về POST

```
<html>
  <body>
    <form action="proSend.asp" method="POST">
      Test POST Method <br>
      <input type="password" name="txtPass" value="123456"><br>
      <input type="checkbox" name="chkAccept"> Accept<br>
      <input type="submit" name="sbtSend" value="Send">
    </form>
  </body>
</html>
```



Tự động chuyển hướng trang web

- Tự động chuyển hướng trang web sang trang web khác (URL) sau một khoảng thời gian t (tính theo giây)

- Cú pháp

```
<head>  
  <META HTTP-EQUIV="refresh" CONTENT="t;URL=URL">  
</head>
```

Nội dung



Giới thiệu về Form

Các thành phần của Form

Một số thuộc tính tiện ích của Form và Input

Phân biệt phương thức GET/POST

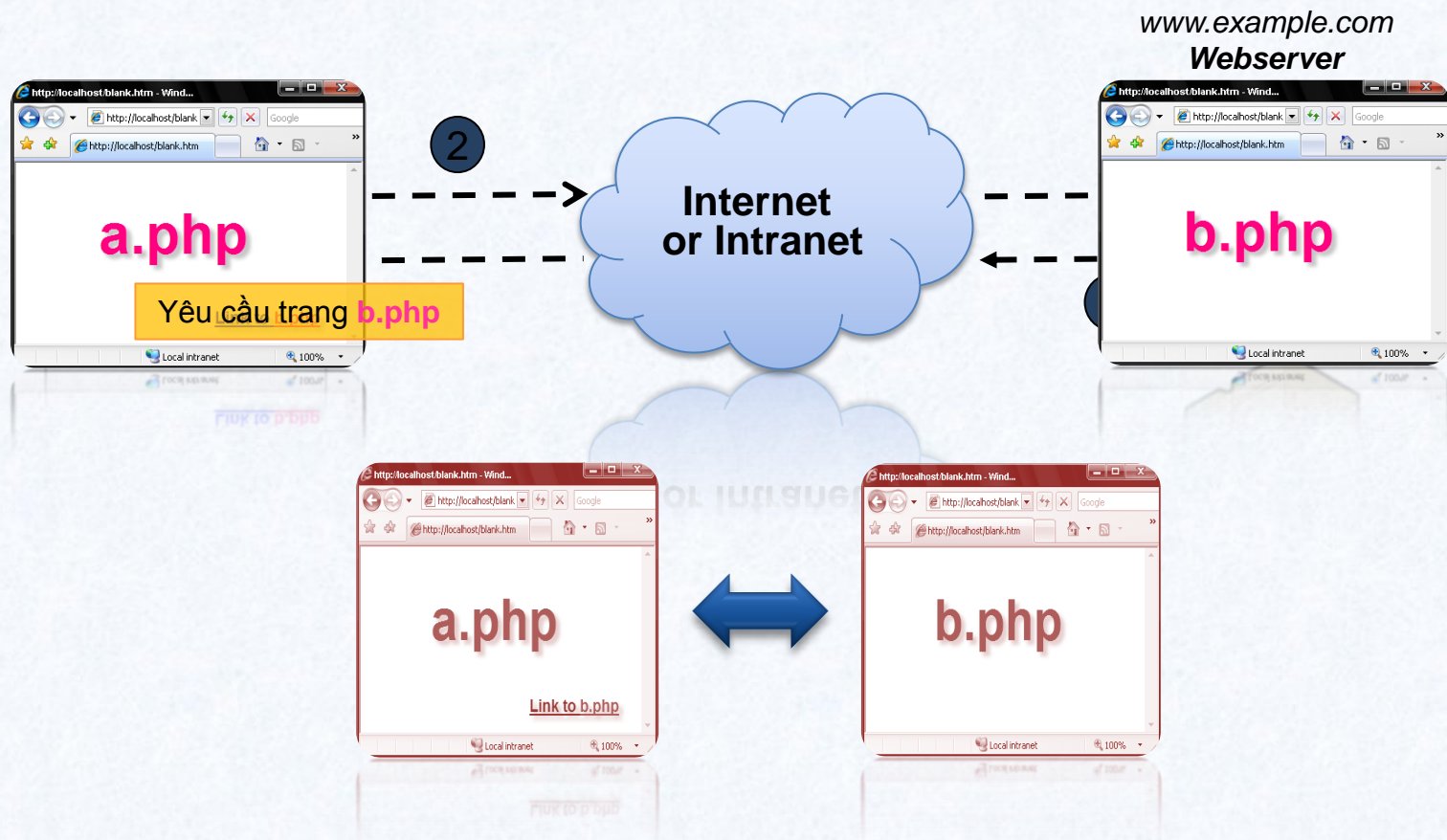
Cơ chế Truyền và Nhận dữ liệu giữa các trang web

Truyền/Nhận qua phương thức GET

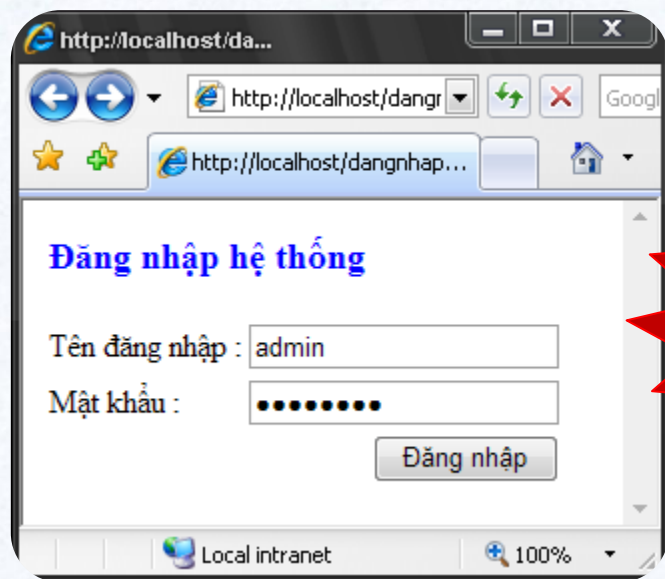
Truyền/Nhận qua phương thức POST

Ví dụ

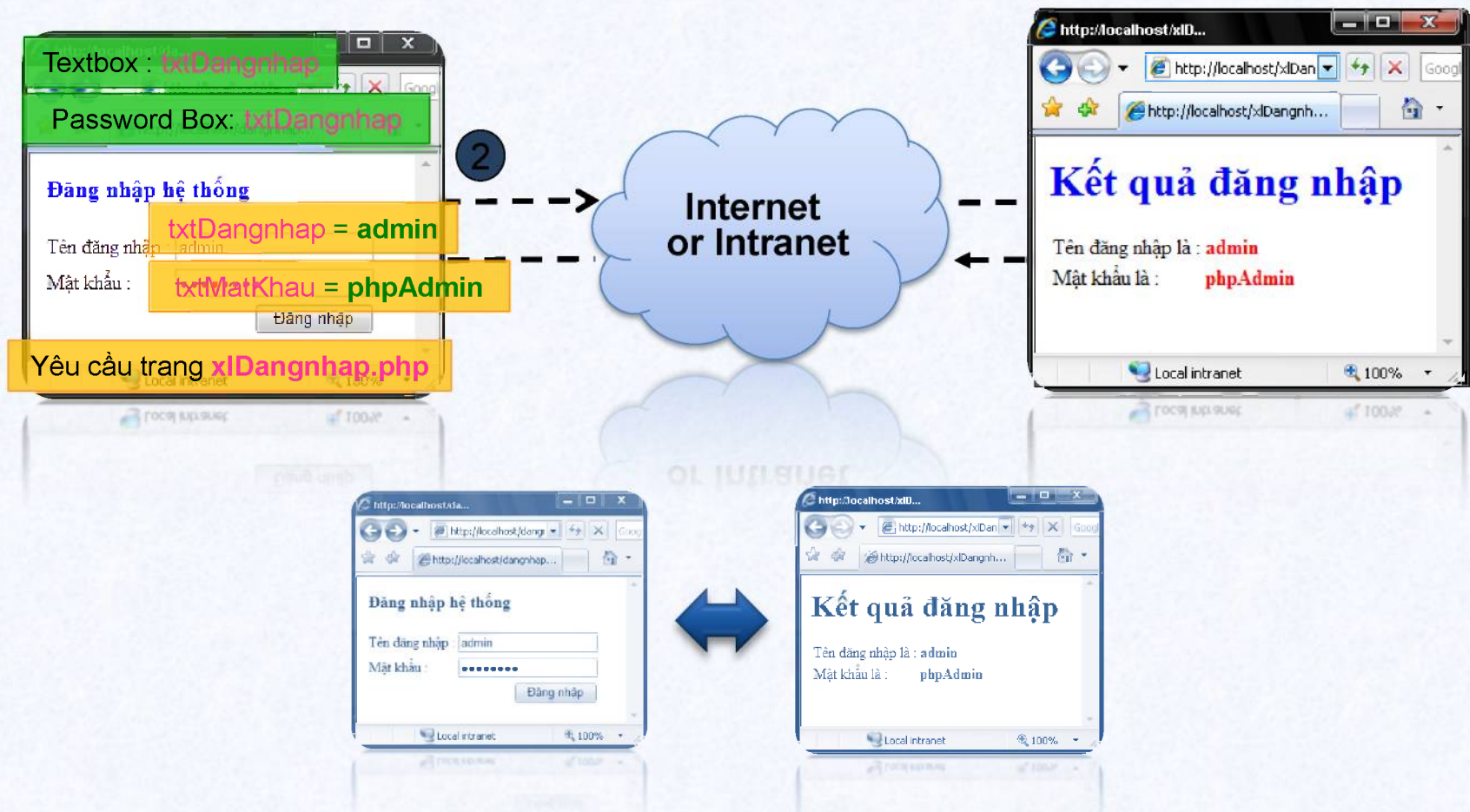
Cơ chế truyền nhận dữ liệu



Cơ chế truyền nhận dữ liệu



Cơ chế truyền nhận dữ liệu



Cơ chế truyền nhận dữ liệu

- Trang web **nhập** dữ liệu :
 - Sử dụng đối tượng `<form>`
 - Nhập liệu thông qua các `formfield`
 - Thực hiện việc truyền dữ liệu thông qua `Submit`

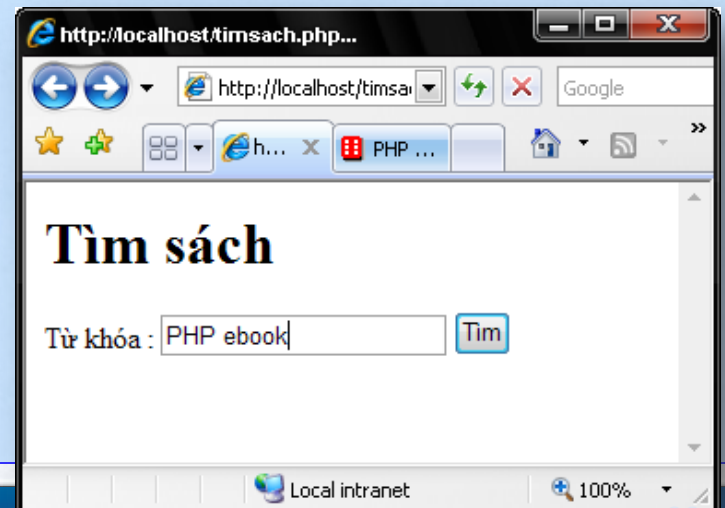
```
<FORM ACTION="URL" METHOD="GET/POST">  
    ...  
    <input type="submit" value="Xử  
lý">  
</FORM>
```

- Trang web **nhận** dữ liệu (**URL**): Sử dụng các biến toàn cục của PHP
 - `$_POST["FieldName"]`
 - `$_GET["FieldName"]`
 - `$_REQUEST["FieldName"]`

Trang web nhập liệu

timSach.php

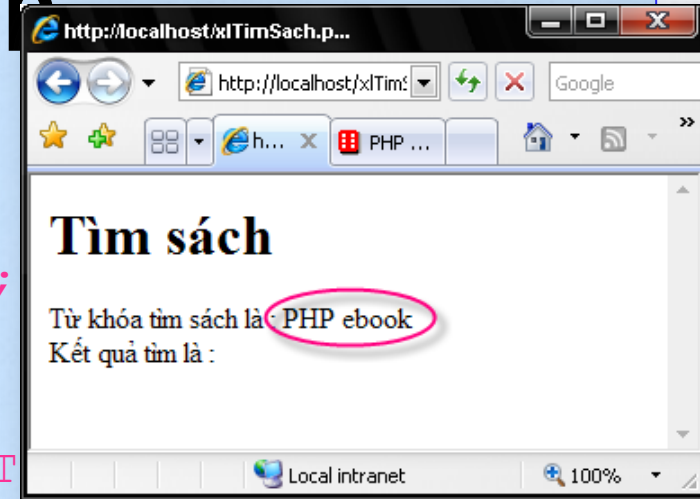
```
<html>
  <body>
    <h1>Tìm sách</h1>
    <form action="xlTimSach.php" Method="GET" >
      Từ khóa : <input type="text" name="txtTukhoa"/>
      <input type="submit" value="Tìm"/>
    </form>
  </body>
</html>
```



Trang web nhập liệu

xltimSach.php

```
<html>
  <body>
    <?php
      $sTukhoa = $_REQUEST["txtTukhoa"];
    ?>
    <h1>Tìm sách</h1>
    Từ khóa tìm sách là : <?php echo $sT
  <br />
  Kết quả tìm là :
</body>
</html>
```

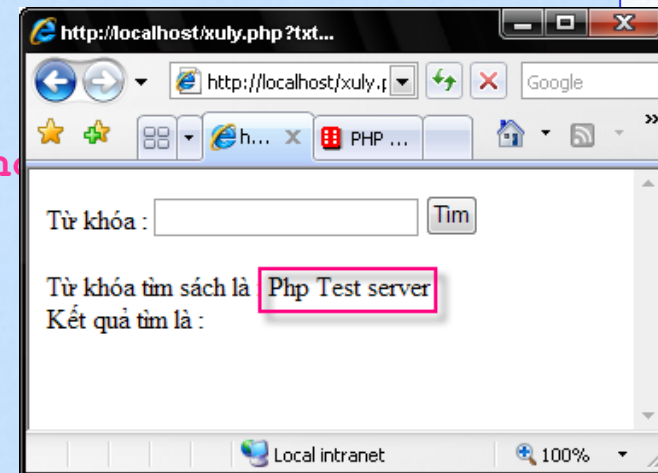
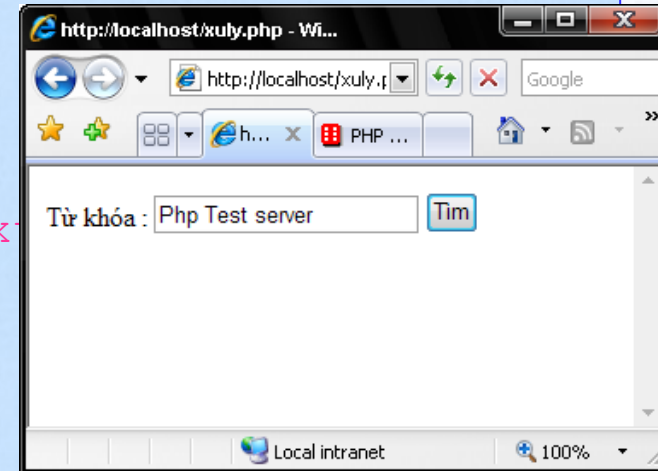


Trang web nhập liệu

xuly.php

```
<html>
  <body>
    <form action="xuly.php" Method="GET" >
      Từ khóa : <input type="text" name="txt" value="Tukhoa" />
      <input type="submit" value="Tìm" />
    </form>

    <?php
      $sTukhoa = $_REQUEST["txtTukhoa"];
      if (isset($sTukhoa))
      {
        print "Từ khóa tìm sách là : $sTukhoa";
        echo "<br>Kết quả tìm là : ";
      }
    ?>
  </body>
</html>
```



Nội dung



Giới thiệu về Form

Các thành phần của Form

Một số thuộc tính tiện ích của Form và Input

Phân biệt phương thức GET/POST

Cơ chế Truyền và Nhận dữ liệu giữa các trang web

Truyền/Nhận qua phương thức GET

Truyền/Nhận qua phương thức POST

Ví dụ

Truyền/Nhận qua Phương thức GET

- Tham số truyền đi qua địa chỉ URL
 - `http://domain/pathfile?fieldname1=value1&fieldname2=value2`
 - Ví dụ:
 - `http://localhost/xuly.php?txtTukhoa=PHPHost`
- Nhận dữ liệu thông qua biến toàn cục của PHP
 - `$_GET["FieldName"]`
 - `$_REQUEST["FieldName"]`

Phương thức GET – Ưu khuyết điểm

● Khuyết điểm

- Không thích hợp để truyền dữ liệu có tính **bảo mật** (password)
- **Dung lượng** Dữ liệu truyền đi có **giới hạn**
- URL submit bằng phương thức GET được lưu lại trên server

● Ưu điểm

- Người dùng có thể **bookmark** địa chỉ URL
- Người dùng có thể **Giả lập phương thức GET** để truyền dữ liệu mà không cần thông qua FORM

Ví dụ

File: GET.HTM

```
<HTML>
```

```
<HEAD>
```

```
<TITLE>Input data</TITLE>
```

```
</HEAD>
```

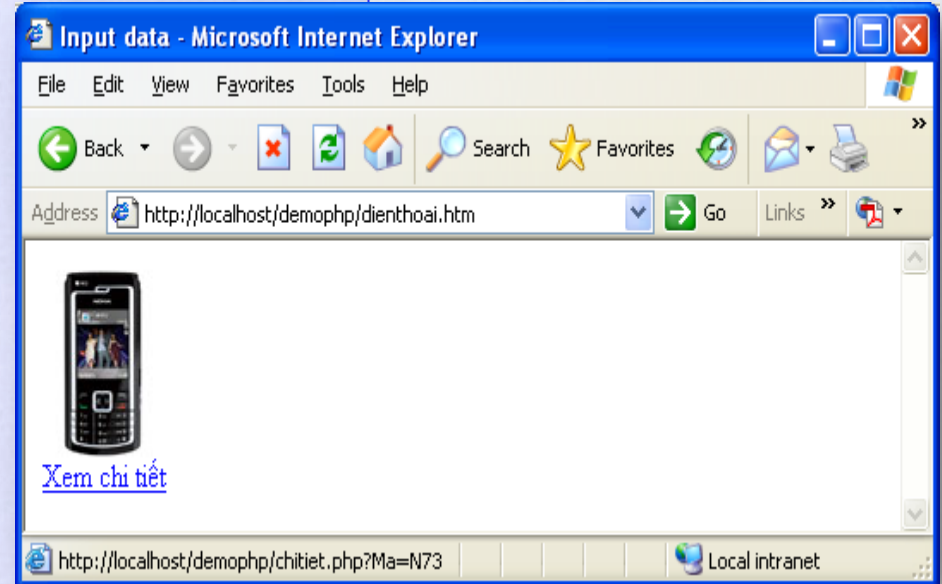
```
<BODY>
```

```
<IMG SRC="images/N72.jpg"><br>
```

```
<A HREF='chitiet.php?Ma=N72'>Xem chi  
tiết</A>
```

```
</BODY>
```

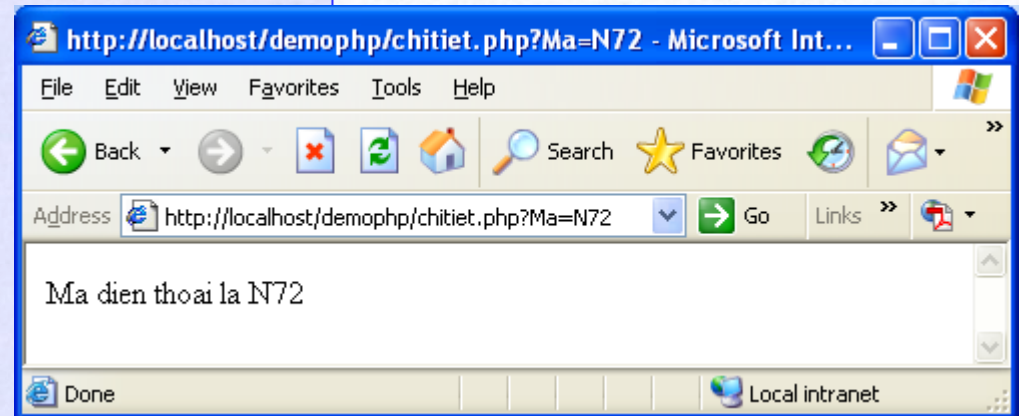
```
</HTML>
```



Ví dụ (tt)

File: CHITIET.PHP

```
<HTML>
<HEAD>
<TITLE>Input data</TITLE>
</HEAD>
<BODY>
<?php
    echo "Ma dien thoai la " .
        $_GET["Ma"];
?>
</BODY>
</HTML>
```



Nội dung



Giới thiệu về Form

Các thành phần của Form

Một số thuộc tính tiện ích của Form và Input

Phân biệt phương thức GET/POST

Cơ chế Truyền và Nhận dữ liệu giữa các trang web

Truyền/Nhận qua phương thức GET

Truyền/Nhận qua phương thức POST

Ví dụ

Truyền/Nhận qua Phương thức POST

- Tham số truyền đi được ẩn bên trong FORM
- Nhận dữ liệu thông qua biến toàn cục của PHP
 - `$_POST["FieldName"]`
 - `$_REQUEST["FieldName"]`

Phương thức POST – Ưu khuyết điểm

• Ưu điểm

- Bảo mật hơn phương thức GET
- Không giới hạn dung lượng dữ liệu truyền đi

• Khuyết điểm

- Kết quả trang web trả về không thể bookmark
- Có thể gây ra lỗi nếu người dùng muốn quay lại trang kết quả (nhấn nút Back hoặc Refresh) do bị expired
- Dữ liệu có thể không truyền đi được do vấn đề về security

Nội dung



Giới thiệu về Form

Các thành phần của Form

Một số thuộc tính tiện ích của Form và Input

Phân biệt phương thức GET/POST

Cơ chế Truyền và Nhận dữ liệu giữa các trang web

Truyền/Nhận qua phương thức GET

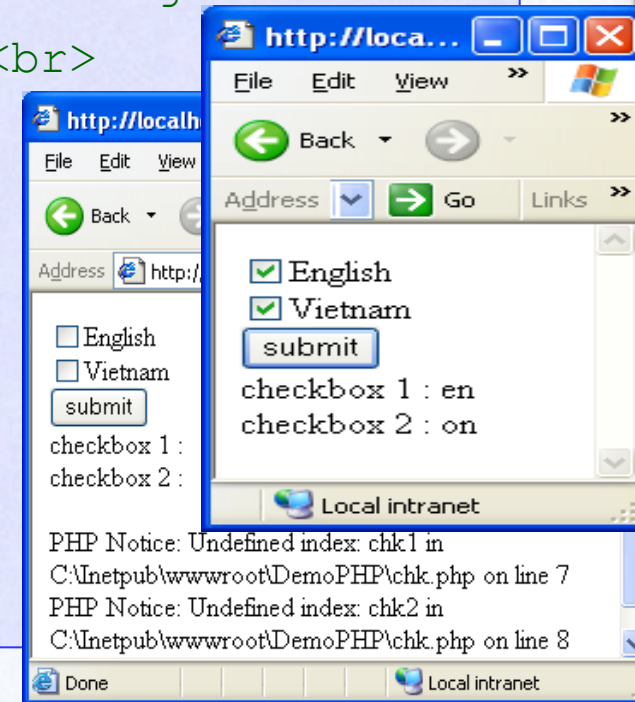
Truyền/Nhận qua phương thức POST

Ví dụ

Truyền/Nhận dữ liệu từ Checkbox

File: Checkbox.php

```
<html><body>
<form method="get" action="checkbox.php">
<input type="checkbox" name="chk1" value="en">English <br>
<input type="checkbox" name="chk2">Vietnam<br>
<input type="submit" value="submit"><br>
</form>
<?php
    echo "checkbox 1 : " . $_REQUEST['chk1'];
    echo "checkbox 2 : " . $_REQUEST['chk2'];
?>
</body></html>
```



Truyền/Nhận dữ liệu từ Radio Button

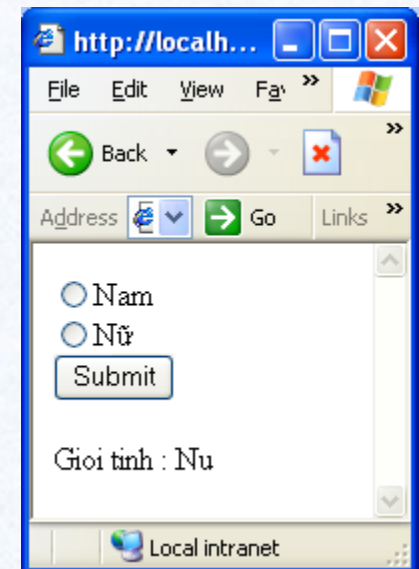
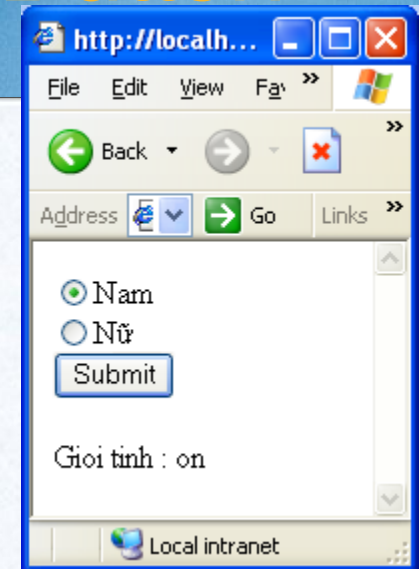
File: RADIO.PHP

```
<html><body>
<form action="radio.php" method="GET">
<input type="RADIO" NAME="radGT">Nam<br>
<input type="RADIO" NAME="radGT"
  VALUE="Nu">Nữ<br>
<input type="SUBMIT" VALUE="Submit">
</form>

<?php
  if (isset($_GET['radGT'])) {
    echo "Gioi tinh : " . $_GET['radGT'];
  }
?>
```

```
</body></html>
```

13/11/2010

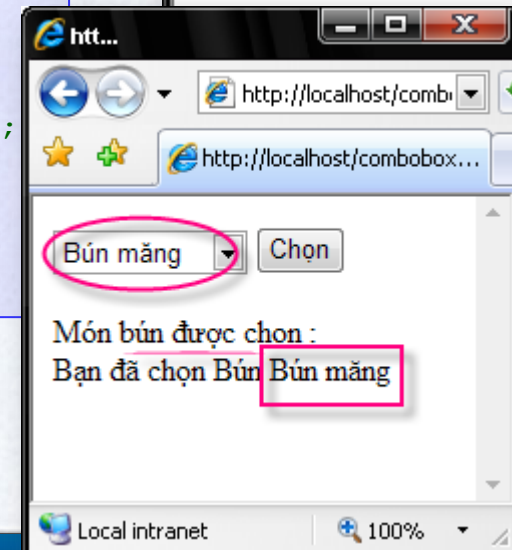
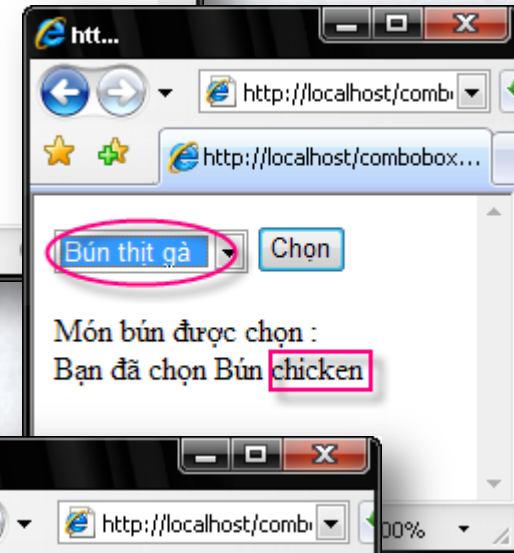
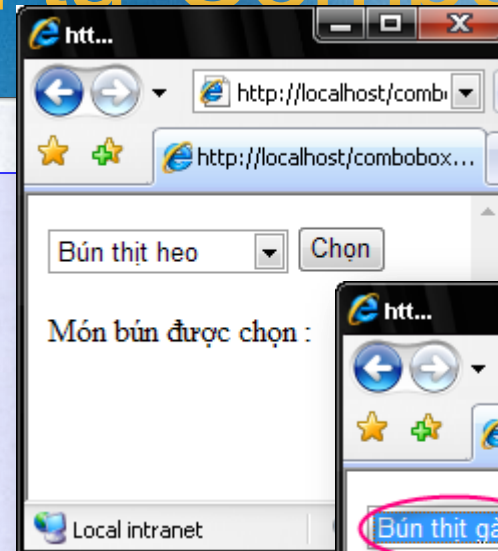


Truyền/Nhận dữ liệu từ ComboBox

File: COMBOBOX.PHP

```
<html>
<body>
  <form method="POST" action="combobox.php">
    <select name="lunch">
      <option value="pork">Bún thịt heo</option>
      <option value="chicken">Bún thịt gà</option>
      <option>Bún măng</option>
    </select>
    <input type="submit" name="submit" value="Chọn" />
  </form>

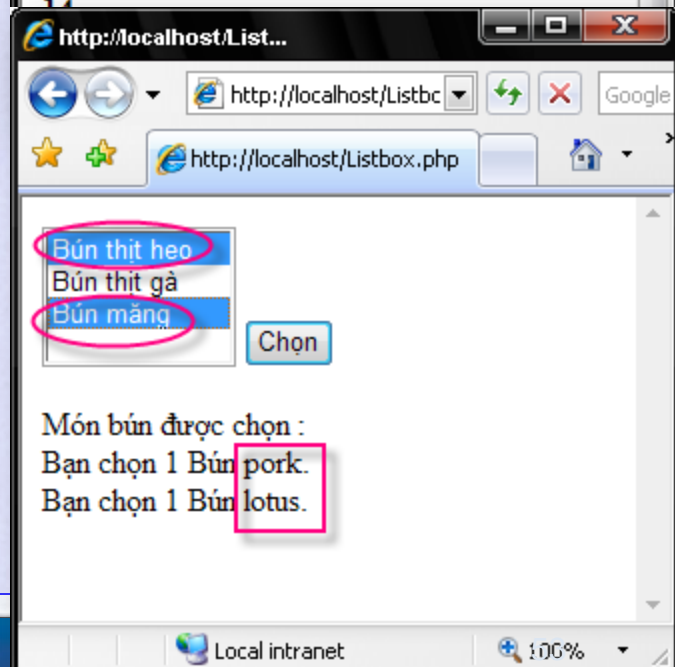
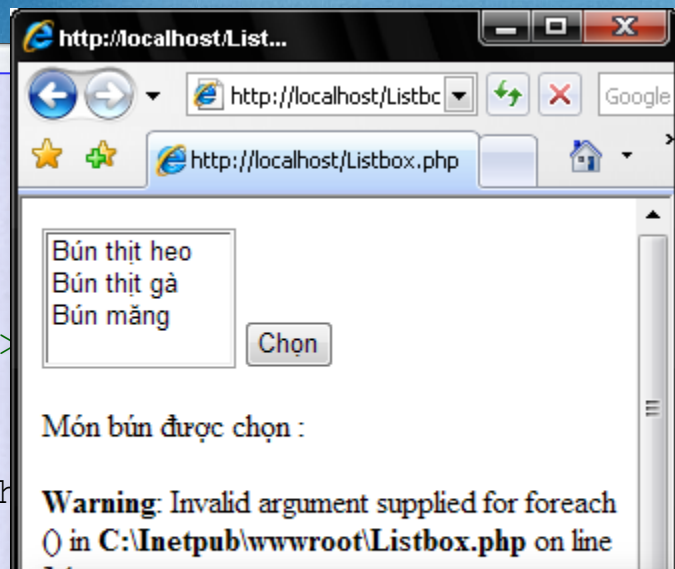
  Món bún được chọn :<br/>
  <?php
    if (isset($_POST['lunch']))
    {
      print "Bạn đã chọn Bún ". $_POST['lunch'] . "<br/>";
    }
  ?>
</body>
</html>
```



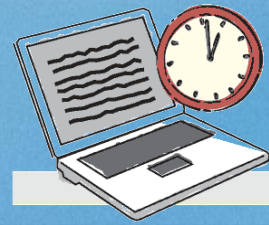
Truyền/Nhận dữ liệu từ Listbox - sử dụng Mảng

File: LISTBOX.PHP

```
<html> <body>
  <form method="POST" action="Listbox.php">
    <select name="lunch[]" multiple>
      <option value="pork">Bún thịt heo</option>
      <option value="chicken">Bún thịt gà</option>
      <option>Bún măng</option>
    </select>
    <input type="submit" name="submit" value="Chọn" />
  </form>
  Món bún được chọn : <br/>
  <?php
    foreach ($_POST['lunch'] as $choice) {
      print "Bạn đã chọn 1 Bún $choice. <br/>";
    }
  ?>
</body></html>
```



Bài tập nhóm



- Xây dựng trang đăng kí cho đề án của nhóm
- Khi bấm vào nút đăng kí, nội dung người dùng điền vào sẽ chuyển sang trang `xldangki.php`
- Trang `xldangki.php` sẽ hiển thị những thông tin người dùng đã truyền vào

