

**ĐẠI HỌC ĐÀ NẴNG**  
**TRƯỜNG ĐẠI HỌC BÁCH KHOA**  
***KHOA CÔNG NGHỆ THÔNG TIN***



**BÀI GIẢNG MÔN HỌC**

**CÔNG NGHỆ WEB**

# TÀI LIỆU THAM KHẢO

1. Internet nối kết mạng (Lê Quang Liêm)
2. Internetworking với TCP/IP (Nguyễn Quốc Cường)
3. Architecis of the Web (Ried Robert.H)
4. Ebook “Thương mại Việt Nam hội nhập thế giới ”
- 5.Các Website TMĐT : Yahoo, Amazon, Ebay, AltaVista,...

<http://www.ecommerce-guide.com>

<http://www.online-commerce.com>

<http://www.ecommercetimes.com>

<http://creating-a-site.com>

# NỘI DUNG CHƯƠNG TRÌNH

## Chương 1 : Internet và Web

Giới thiệu cung cấp bối cảnh về Internet và Web, giới thiệu các khái niệm cơ bản như mạng Internet, các trang web đóng vai trò trung gian trong Internet, nhằm bao gồm các website gồm những thành phần nào.

## Chương 2 : Ngôn ngữ HTML

Mô tả bối cảnh nội dung của ngôn ngữ HTML là một ngôn ngữ để tạo ra các trang web.

## Chương 3: JavaScript và VBScript

Mô tả bối cảnh cách sử dụng các ngôn ngữ JavaScript và VBScript để tạo ra các trang web động.

## Chương 4: Công nghệ Web động

Mô tả các công nghệ động như ASP, JSP, SERVLET, PHP (AJAX), ...

## Chương 5: Cài đặt và cài đặt Web Server

Mô tả bối cảnh cách cài đặt và cài đặt một server web để quản lý và triển khai website.

# CHƯƠNG 1 : INTERNET VÀ WEB

## I. KHÁI NIỆM VỀ MẠNG MÁY TÍNH

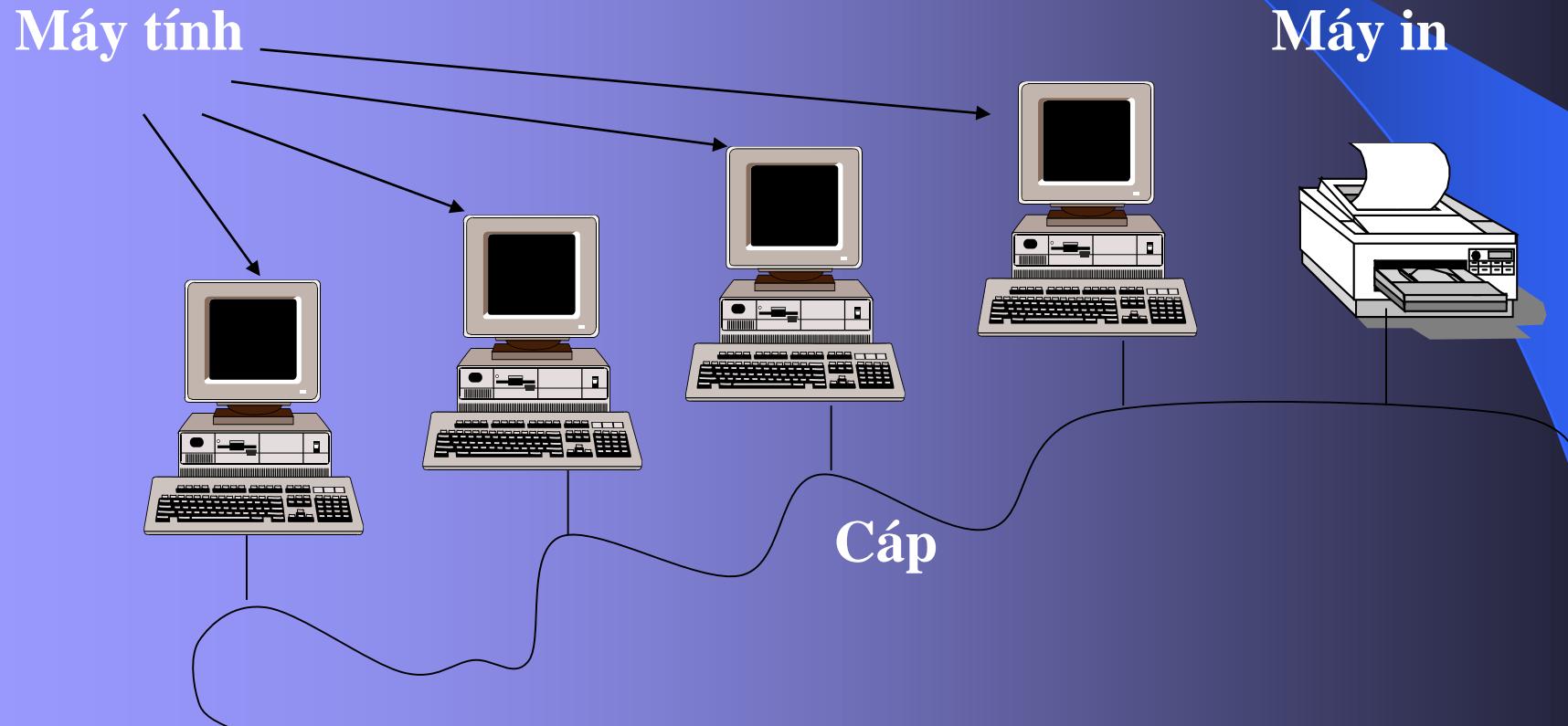
☞ *Mạng máy tính (network) là bao gồm các máy tính được nối với nhau sao cho chúng có thể dùng chung dữ liệu và các thiết bị.*

☞ *Các máy tính trong mạng có thể dùng chung các loại tài nguyên sau:*

- *Dữ liệu*
- *Thông điệp*
- *Hình ảnh*
- *Máy fax*
- *Modem*
- *Máy in*
- *Các tài nguyên phần cứng khác*

# CHƯƠNG 1 : INTERNET VÀ WEB

## Hệ thống mạng máy tính đơn giản



# **CHƯƠNG 1 : INTERNET VÀ WEB**

## **II. CÁC LOẠI MẠNG MÁY TÍNH**

### **1. MẠNG NỘI BỘ (LOCAL AREA NETWORK - LAN)**

Mạng nội bộ (LAN) là một mạng của các máy tính được nối với nhau trong một phạm vi hẹp như trong một tòa nhà, một công sở nhờ một số loại cáp dẫn, không dùng thuê bao điện thoại.

### **2. MẠNG DIỆN RỘNG (WIDE AREA NETWORK - WAN)**

Mạng diện rộng (WAN) là một mạng của nhiều mạng nội bộ nối với nhau thông qua đường dây điện thoại hoặc nhờ các công nghệ khác như hệ thống điện tử viễn thông hoặc vệ tinh.

# CHƯƠNG 1 : INTERNET VÀ WEB

## II. CÁC LOẠI MẠNG MÁY TÍNH

### 3. MẠNG INTERNET LÀ GÌ?



# CHƯƠNG 1 : INTERNET VÀ WEB

## II. CÁC LOẠI MẠNG MÁY TÍNH

### 3. MẠNG INTERNET LÀ GÌ?

Internet là liên mạng toàn cầu lớn nhất bao trùm lên tất cả các mạng khác phân bố trên phạm vi toàn thế giới.

+ Giúp các máy tính kết nối và trao đổi thông tin với nhau mà không cần biết vị trí địa lý của chúng

+ Cho phép truy cập lên các tài nguyên lưu trữ ở xa.

+ Cho phép trao đổi thư điện tử (Email), trò chuyện (Chat), gọi điện thoại, gửi tin nhắn,..

+ Các tư liệu thông tin được tổ chức theo từng trang (trang Web). Các trang này được liên kết với nhau nhờ các liên kết hay các chỉ mục.

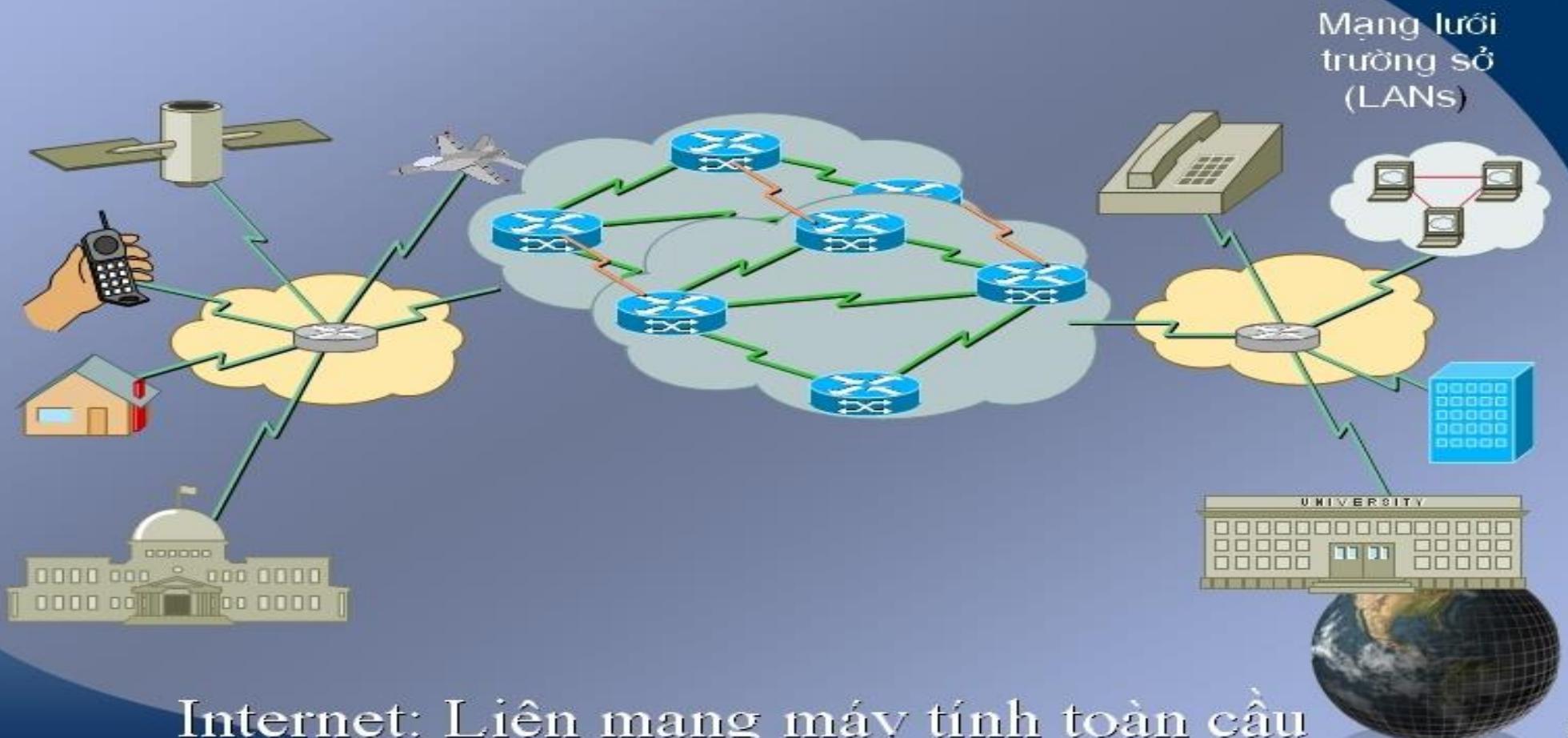
# CHƯƠNG 1 : INTERNET VÀ WEB

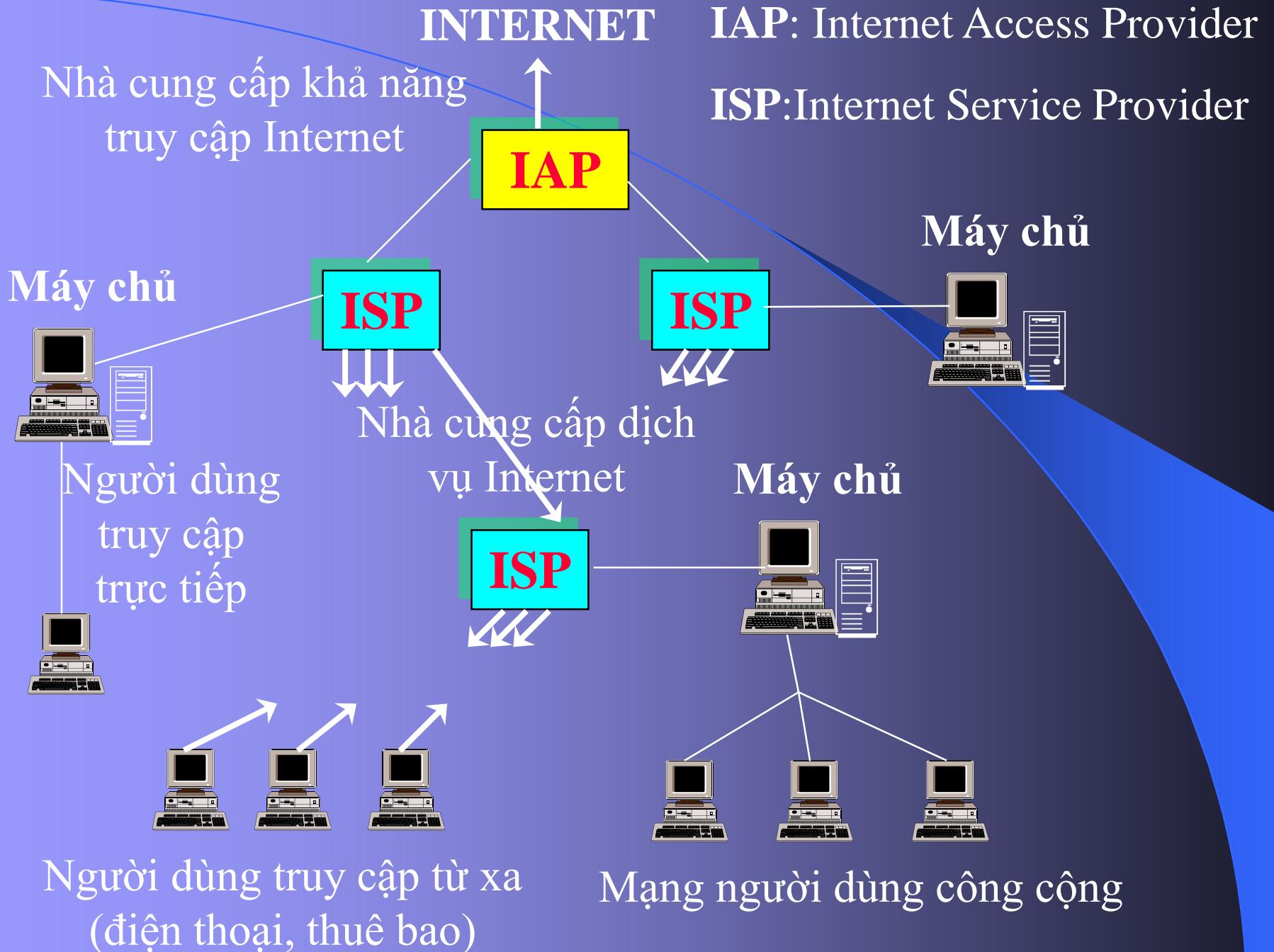
## Lịch sử phát triển Internet

- Tiền thân của Internet là ARPANET, 1960
- Năm 1983, giao thức TCP/IP chính thức được coi như một chuẩn cho việc kết nối máy tính
- Năm 1984, ARPANET đã được chia ra thành hai phần: phần thứ nhất vẫn được gọi là ARPANET, dành cho việc nghiên cứu và phát triển; phần thứ hai được gọi là MILNET, là mạng dùng cho các mục đích quân sự.
- WWW ra đời, 1989
- Ngày nay chúng ta thấy Internet xuất hiện trong mọi lĩnh vực : thương mại, chính trị, quân sự, nghiên cứu, giáo dục, văn hoá, xã hội ...

# CHƯƠNG 1 : INTERNET VÀ WEB

## Mô hình mạng Internet





# CHƯƠNG 1 : INTERNET VÀ WEB

## Kết nối Internet

- Để kết nối được Internet cần :

- Máy tính (...)
- Modem (...)
- Đường dây điện thoại
- Nhà cung cấp dịch vụ
  - + America Online
  - + VDC (VNPT)
  - + FPT, Viettel, Netnam...
- Các phần mềm Internet

# CHƯƠNG 1 : INTERNET VÀ WEB

## III. CÁC KHÁI NIỆM

### 1. GIAO THỨC TCP/IP

TCP/IP là một giao thức để kết nối và truyền dữ liệu giữa các máy tính với nhau.

+ Mục đích chính của IP là cung cấp khả năng kết nối các mạng con thành liên mạng để truyền dữ liệu. IP là một giao thức kiểu không liên kết.

+ TCP là một giao thức kiểu "có liên kết", nghĩa là phải thiết lập liên kết giữa một cặp thực thể TCP trước khi chúng trao đổi dữ liệu với nhau.

# CHƯƠNG 1 : INTERNET VÀ WEB

## III. CÁC KHÁI NIỆM

### 2. ĐỊA CHỈ IP :

Địa chỉ IP dùng để các máy tính liên lạc được với nhau và phân biệt được nhau.

+ IP tĩnh và IP động

Cấu trúc của địa chỉ IP này gồm một chuỗi số 32 bit và được chia thành 4 nhóm, cách nhau bởi dấu chấm ( . ). Mỗi nhóm gồm 3 chữ số có giá trị từ 000 đến 255.

Ví dụ : địa chỉ IP của một máy tính : 192.168.2.3

# CHƯƠNG 1 : INTERNET VÀ WEB

## III. CÁC KHÁI NIỆM

### 3. Tên miền (Domain Name)

**Địa chỉ IP** là những dãy số, do đó người dùng Internet rất khó nhớ. Vì vậy người ta dùng khái niệm tên miền (Domain Name) để ánh xạ tới địa chỉ IP. Tuy nhiên tên miền chỉ là tên dành cho người dùng dễ nhớ, cốt lõi của địa chỉ Internet vẫn là IP.

Tên miền của một máy chủ gồm các chuỗi ký tự phân cách nhau bằng dấu chấm (.).

Khuôn dạng tên miền: **xxx.xxx.xxx.xxx**

- Nhóm ký tự cuối cùng đại diện cho tên nước.
- Nhóm kế tiếp đại diện cho loại cơ quan
- Nhóm kế tiếp đại diện cho tên cơ quan
- Nhóm đầu tiên đại diện cho tên máy tính

# CHƯƠNG 1 : INTERNET VÀ WEB

## III. CÁC KHÁI NIỆM

### 3. Tên miền (Domain Name)

Ví dụ:

<i>Địa chỉ IP</i>	<i>Địa chỉ theo tên miền</i>
113.115.12.37	www.pacific.net.sg
203.162.16.235	home.vnn.vn
203.162.7.48	fpt.com.vn
110.32.26.74	www.yahoo.com

# CHƯƠNG 1 : INTERNET VÀ WEB

## III. CÁC KHÁI NIỆM

### 4. Máy chủ (Server)

- ☞ Là một máy tính được nối vào mạng và có khả năng cung cấp thông tin cho máy tính khác trong mạng.
- ☞ Là máy trong mạng mà các máy tính khác trong mạng có thể truy cập được.

### 5. Máy khách (Client)

- ☞ Là một máy tính được nối vào mạng và có khả năng truy cập vào các máy khác để yêu cầu đáp ứng thông tin. (Máy của những người sử dụng bình thường).

# CHƯƠNG 1 : INTERNET VÀ WEB

## IV. CÁC DỊCH VỤ CỦA INTERNET

### 1. Dịch vụ thư điện tử (E-mail) :

- Electronic mail (E-mail) : Là một dịch vụ của Internet giúp cho việc trao đổi thông điệp giữa những người dùng hay nhóm người dùng trên mạng
- Dựa trên giao thức chuẩn Internet: Simple Mail Transfer Protocol (SMTP)
- Khả năng gửi tới nhiều người cùng một thời điểm
- Nhanh chóng chuyển giao được tài liệu
- Chi phí thấp

# **CHƯƠNG 1 : INTERNET VÀ WEB**

## **IV. CÁC DỊCH VỤ CỦA INTERNET**

### **1. Dịch vụ thư điện tử (E-mail) :**

#### **Các mô hình hoạt động E-Mail :**

- Mô hình thông điệp trực tiếp
- Mô Hình hộp thư lưu
- Mô hình Internet

# CHƯƠNG 1 : INTERNET VÀ WEB

## IV. CÁC DỊCH VỤ CỦA INTERNET

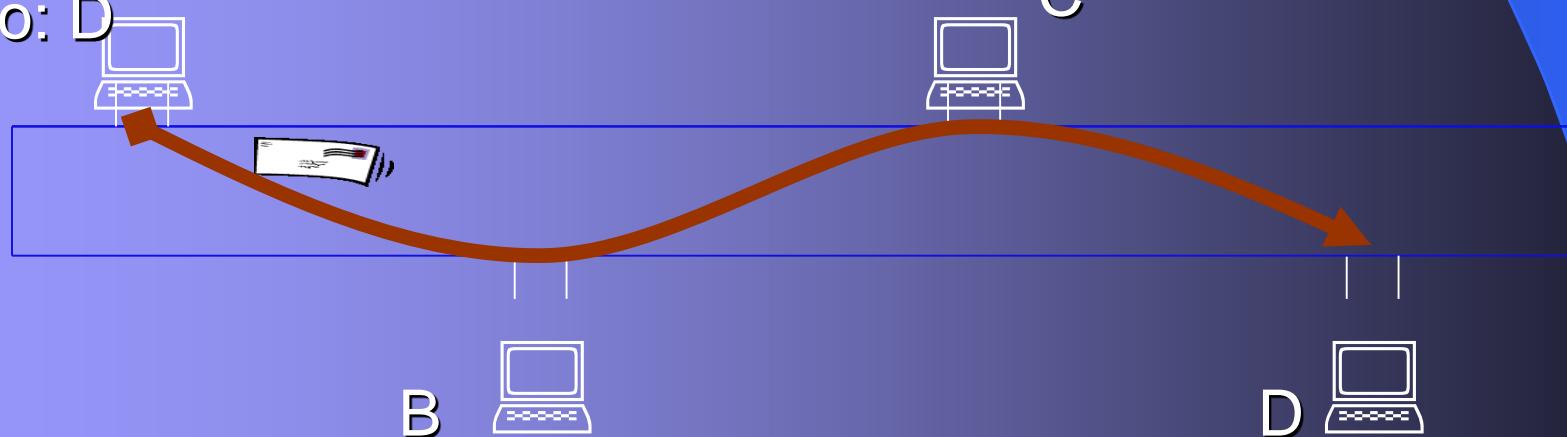
### 1. Dịch vụ thư điện tử (E-mail) :

#### Mô hình thông điệp trực tiếp

Các thông điệp được gửi trực tiếp ngay lập tức tới máy đang hoạt động trong mạng nội bộ

from: A

to: D



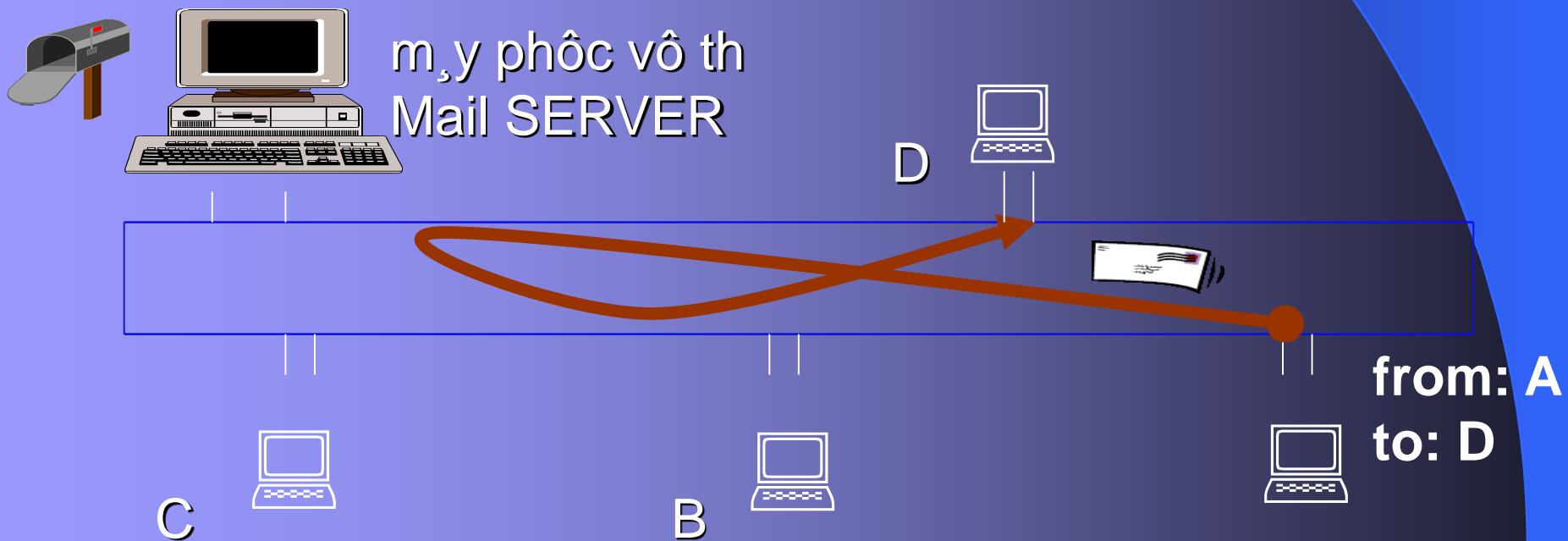
# CHƯƠNG 1 : INTERNET VÀ WEB

## IV. CÁC DỊCH VỤ CỦA INTERNET

### 1. Dịch vụ thư điện tử (E-mail) :

#### Mô hình hộp thư lưu

Thông điệp được gửi gián tiếp tới một máy phục vụ đang hoạt động trong mạng nội bộ



# CHƯƠNG 1 : INTERNET VÀ WEB

## Các thành phần cơ bản

- Mail server: Chương trình phục vụ thư
- Mail Client: Chương trình cho người sử dụng
- Cách thức giao nhận thư

# CHƯƠNG 1 : INTERNET VÀ WEB

## Tài khoản thư điện tử

- Tài khoản đăng nhập thư điện tử gồm:  
    Tên đăng ký (account name)  
    Mật khẩu (Password)
- Địa chỉ thư của người sử dụng sau khi đăng ký:  
    tên\_đăng\_ký@tên\_miền

*Trong đó: tên miền là tên định danh của máy phục vụ thư - tên được cấp bởi tổ chức quản lý tên miền internet*

Ví dụ: [mvha@ud.edu.vn](mailto:mvha@ud.edu.vn)

# **CHƯƠNG 1 : INTERNET VÀ WEB**

## **Thư điện tử - Webmail**

Là các địa chỉ E-mail miễn phí như:

<http://www.yahoo.com>

<http://www.hotmail.com>

<http://vol.vnn.vn>

<http://mail.emailpinoy.com>

<http://Saigon.vnn.vn/webmail>

[www.mailexcite.com](http://www.mailexcite.com)

[www.netaddress.com](http://www.netaddress.com)

# **CHƯƠNG 1 : INTERNET VÀ WEB**

## **IV. CÁC DỊCH VỤ CỦA INTERNET**

### **2. World Wide Web (www)**

- World Wide Web (WWW) hoặc Web là một dịch vụ của Internet
- Là dịch vụ cung cấp thông tin một cách toàn diện nhất cho người dùng.
- Web chứa thông tin bao gồm văn bản, hình ảnh, âm thanh và thậm chí cả video được kết hợp với nhau.
- Web là kho thông tin khổng lồ: phong phú về nội dung, đa dạng về hình thức, thường xuyên được cập nhật, đổi mới và phát triển không ngừng

# CHƯƠNG 1 : INTERNET VÀ WEB

## IV. CÁC DỊCH VỤ CỦA INTERNET

### 3. Chat

- Là hình thức hội thoại trực tiếp trên Internet, là một dịch vụ của Internet
- Dịch vụ này cho phép hai hay nhiều người có thể cùng trao đổi thông tin trực tuyến với nhau thông qua bàn phím và màn hình.

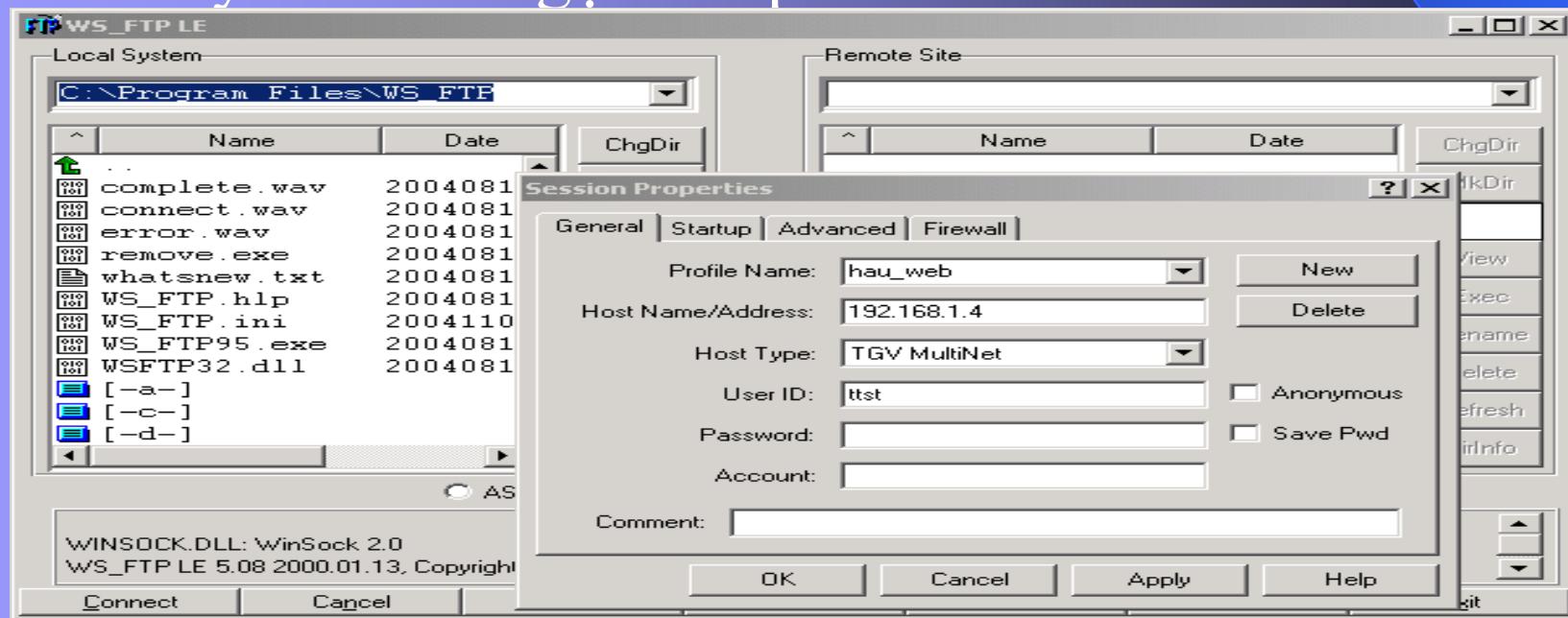
# CHƯƠNG 1 : INTERNET VÀ WEB

## IV. CÁC DỊCH VỤ CỦA INTERNET

### 4. FTP (File Transfer Protocol)

Dịch vụ này cho phép truyền các tập tin từ máy tính này đến máy tính khác trong mạng theo giao thức FTP.

Lấy file từ máy tính ở xa về người ta gọi là Download, gửi file đến máy tính ở xa gọi là Upload



# CHƯƠNG 1 : INTERNET VÀ WEB

## IV. CÁC DỊCH VỤ CỦA INTERNET

### 5. Dịch vụ Telnet

- \* Cho phép người sử dụng đăng nhập vào máy tính chủ từ xa thông qua một máy khác trên mạng
- \* Hệ thống ở xa sẽ yêu cầu xác nhận tên đăng ký và mật khẩu để đăng nhập
- \* Sau khi kết nối thì những gì ta nhập từ bàn phím của người sử dụng đều được gửi đến máy tính từ xa
- \* *Lệnh: Telnet <tên host> hoặc <địa chỉ IP>*
- \* *Trong đó: <tên host> là tên máy chủ muốn truy cập. <địa chỉ IP> là địa chỉ của máy chủ muốn truy cập.*

# CHƯƠNG 1 : INTERNET VÀ WEB

## IV. CÁC DỊCH VỤ CỦA INTERNET

### 6. Các dịch vụ khác :

- \* **Gopher** – Tra cứu thông tin theo thực đơn
- \* **Dịch vụ WAIS** (Wide Area Information Server)

Tìm kiếm thông tin theo diện rộng

- \* **Mailing list:** Danh sách trao đổi trên Email

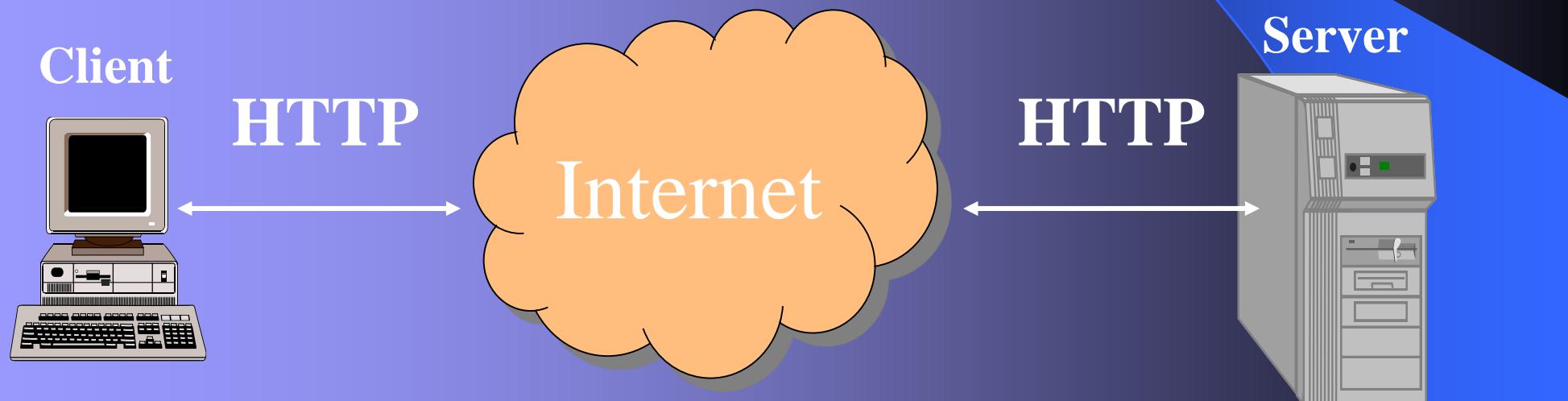
Các dạng mailing list :

- *Mailing list không có nhà quản trị*
- *Mailing list có nhà quản trị (thông dụng)*
- *Mailing list một chiều*

# CHƯƠNG 1 : INTERNET VÀ WEB

## V. KIẾN TRÚC CƠ BẢN CỦA WORLD WIDE WEB (WWW)

### 1. Mô hình chung :



**Client** gửi yêu cầu  
thông tin lên Internet

**Server** nhận yêu cầu, xử lý  
và gửi kết quả về cho Client

# CHƯƠNG 1 : INTERNET VÀ WEB

## V. KIẾN TRÚC CƠ BẢN CỦA WEB (WWW)

### 2. Các khái niệm :

#### a. Địa chỉ URL :

URL (Uniform Resource Locator ). Là địa chỉ đầy đủ chỉ đến một tài nguyên thông tin trên Internet.

Ví dụ:

<http://www.hssv.vnn.vn/dethidapan/index.html>

Trong đó:

http:// Gọi là tên dịch vụ (có thể gọi là phần giao thức).

www.hssv.vnn.vn/ Là phần tên máy chủ.

dethidapan/index.html Đường dẫn đến tài nguyên.

# CHƯƠNG 1 : INTERNET VÀ WEB

## V. KIẾN TRÚC CƠ BẢN CỦA WEB (WWW)

### 2. Các khái niệm :

#### a. Địa chỉ URL :

Lỗi Ých cña URL lµ cung cÊp mét ph¬ng ph,p ®¬n gi¶n, ®ång b  cho viÖc ®Þnh danh c,c tui nguy n cã s½n tr n Internet, th ng qua r t nhiÒu giao th c kh, c nhau nh lµ HTTP, FTP, Telnet,...

B¶ng d i ®cy m« t¶ mét s e khu n d ng URL chu n ®ang ®îc d ng r t ph e bi n:

# CHƯƠNG 1 : INTERNET VÀ WEB

## V. KIẾN TRÚC CƠ BẢN CỦA WEB (WWW)

### 2. Các khái niệm :

#### a. Địa chỉ URL :

Dụng vó	Khu«n dìng URL	Ngczęm ®Đnh
HTTP	<u>Http://Host:Port/Path</u>	Port: 80
FTP	<u>Ftp://User.Password@Host:Port/Path</u>	User: Anonymous Password: ®Đpa chØ e-mail Port: 21
GOPHER	<u>Gopher://Host:Port/Path</u>	Port: 70
MAILTO	<u>Mailto:Address@Host</u>	Kh«ng
TELNET	<u>Telnet://User:Password@Host:Port</u>	Port: 23

# CHƯƠNG 1 : INTERNET VÀ WEB

## V. KIẾN TRÚC CƠ BẢN CỦA WEB (WWW)

### 2. Các khái niệm :

#### b. Giao thức HTTP :

Web Browser và Web Server giao tiếp với nhau thông qua một giao thức được gọi là HTTP. Sự kết nối bằng Http qua 4 giai đoạn:

- + **Tạo kết nối** : Web Browser giao tiếp với Web Server nhờ địa chỉ Internet và số cổng (ngầm định là 80) được đặc tả trong URL
- + **Thực hiện yêu cầu** : Web Browser gửi thông tin tới Web Server để yêu cầu phục vụ. Việc gửi và nhận thông tin ở đây theo phương thức POST, GET
- + **Phản hồi**: Web Server gửi một phản hồi về Web Browser nhằm đáp ứng yêu cầu của Web Browser.
- + **Kết thúc kết nối**: Khi kết thúc quá trình trao đổi giữa Web Browser và Web Server thì sự kết nối chấm dứt.

# CHƯƠNG 1 : INTERNET VÀ WEB

## V. KIẾN TRÚC CƠ BẢN CỦA WEB (WWW)

### 2. Các khái niệm :

#### c. Web Server

- ☞ Là một phần mềm được sử dụng trên máy chủ (máy phục vụ) để lắng nghe các yêu cầu từ các máy khách Client (Web Client) và đáp ứng yêu cầu cho máy Client.
- ☞ Có rất nhiều Web Server chạy trên nhiều platform : Web Server của NSCA trên UNIX, Web Server dùng trên môi trường Windows, Web Server của Oracle được thiết kế để tận dụng khả năng truy cập vào cơ sở dữ liệu hùng mạnh của Oracle.

# CHƯƠNG 1 : INTERNET VÀ WEB

## V. KIẾN TRÚC CƠ BẢN CỦA WEB (WWW)

### 2. Các khái niệm :

#### d. Web Browser

- ☞ Là một phần mềm được sử dụng trên máy khách (Client), dùng để gửi các yêu cầu lên **Web Server** và nhận các kết quả đáp ứng từ Server đồng thời hiển thị lên với khuôn dạng thích hợp.
- ☞ Các trình duyệt : Nestcape Navigator, Mosaic NCSA, Internet Explore

# CHƯƠNG 1 : INTERNET VÀ WEB

## V. KIẾN TRÚC CƠ BẢN CỦA WEB (WWW)

### 3. Mô hình xử lý Client/Server :

Như chúng ta đã biết, mọi vấn đề được thực hiện trên máy tính đều phải thông qua việc viết chương trình.

Một chương trình có thể chia làm 3 lớp như sau:



# CHƯƠNG 1 : INTERNET VÀ WEB

## V. KIẾN TRÚC CƠ BẢN CỦA WEB (WWW)

### 3. Mô hình xử lý Client/Server :

+ Lớp Presentation (lớp trình bày): Lớp này quản lý giao diện người sử dụng, cho phép người sử dụng xem các thông tin, báo biểu của chương trình, hoặc cho phép người sử dụng nhập các thông tin đầu vào.

+ Lớp Application Logic (lớp logic ứng dụng): Lớp này được xem như là trái tim của ứng dụng. Nó bao gồm các module cần thiết để hoàn thiện các chức năng của ứng dụng.

+ Lớp Data Management (lớp quản lý dữ liệu): lớp này bao gồm tất cả các dữ liệu được sử dụng bởi ứng dụng, cũng như các quy tắc hợp lệ khác nhau được yêu cầu để đảm bảo tính toàn vẹn dữ liệu.

# CHƯƠNG 1 : INTERNET VÀ WEB

## V. KIẾN TRÚC CƠ BẢN CỦA WEB (WWW)

### 3. Mô hình xử lý Client/Server :

Một mô hình xử lý Client/Server phải có 3 thành phần cơ bản :

Front-end Client

Back-end Server

Mạng máy tính (Network)

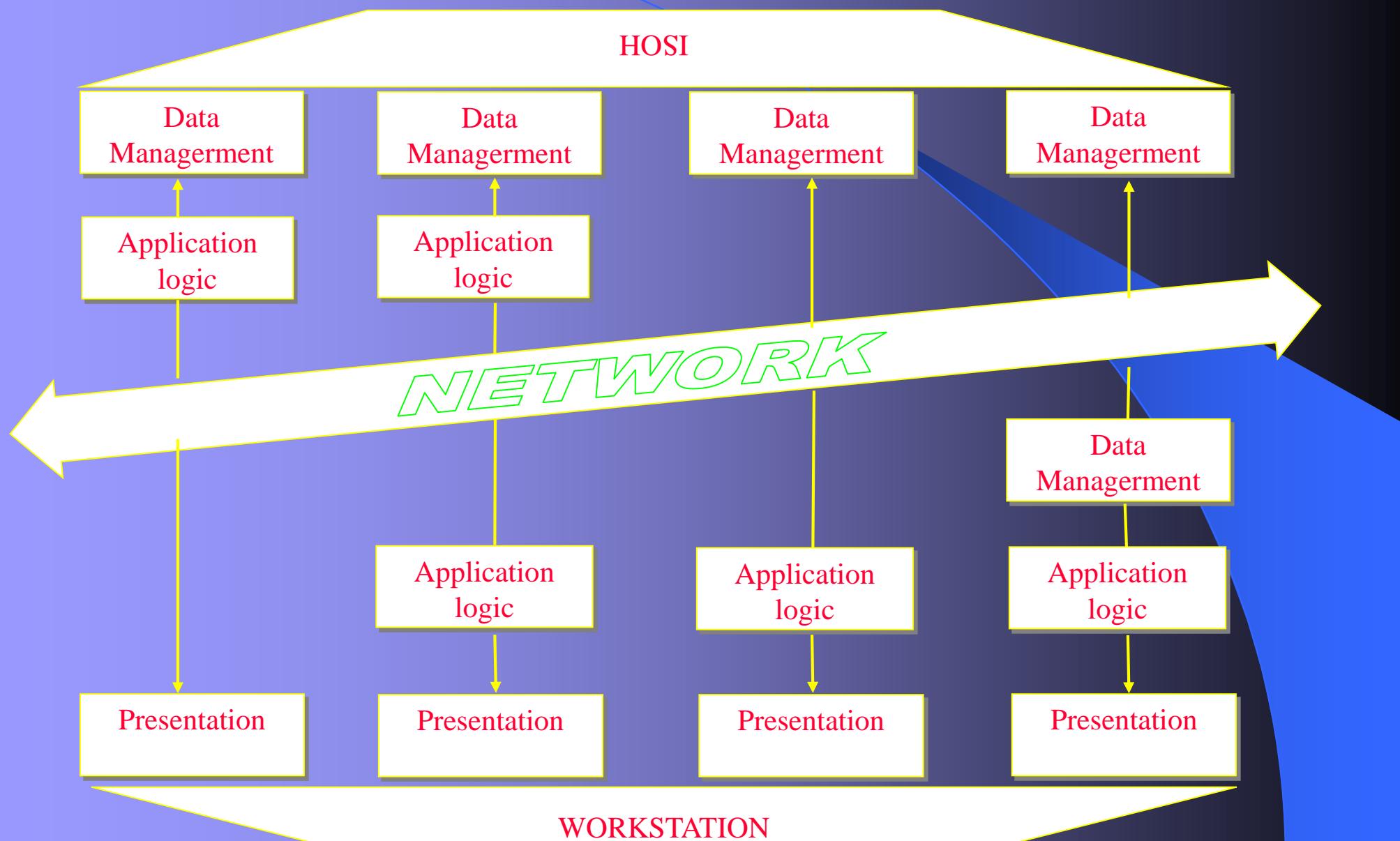
Một chương trình **Front-end Client** chạy trên trạm làm việc (Workstation), mà ở đó người sử dụng giao tiếp với ứng dụng để yêu cầu cung cấp dịch vụ, như truy vấn dữ liệu.

Chương trình **Back-end Server** chạy trên máy chủ (Host) tiếp nhận thông tin và cung cấp dịch vụ được yêu cầu, như phản hồi truy vấn.

**Mạng máy tính** có chức năng truyền tải thông tin.

# V. KIẾN TRÚC CƠ BẢN CỦA WEB (WWW)

## 3. Mô hình xử lý Client/Server : 4 mô hình xử lý phổ biến



# **CHƯƠNG 1 : INTERNET VÀ WEB**

## **V. KIẾN TRÚC CƠ BẢN CỦA WEB (WWW)**

### **3. Mô hình xử lý Client/Server :**

+ **Mô hình đầu tiên là mô hình trình bày từ xa :**

mô hình này máy chủ thực hiện truy cập dữ liệu, xử lý tính toán. Kết quả tính toán được trả về và trình bày trên máy trạm.

+ **Mô hình thứ hai là mô hình xử lý phân tán :**

mô hình này chia sẻ năng lượng tính toán trên máy chủ cho máy trạm. Máy trạm ngoài chức năng quản lý giao diện còn được phân bổ một số chức năng xử lý thích hợp.

# **CHƯƠNG 1 : INTERNET VÀ WEB**

## **V. KIẾN TRÚC CƠ BẢN CỦA WEB (WWW)**

### **3. Mô hình xử lý Client/Server :**

+ **Mô hình thứ 3 là mô hình quản lý dữ liệu từ xa** : mô hình này tận dụng khả năng tính toán của máy trạm. Máy chủ chỉ có chức năng quản lý dữ liệu, còn máy trạm có nhiệm vụ xử lý tính toán và quản lý giao diện người sử dụng.

+ **Mô hình cuối cùng là mô hình quản lý dữ liệu phân tán**: đặc điểm của mô hình này là toàn bộ năng lượng tính toán tập trung trên máy trạm. Tuy nhiên một phần dữ liệu được quản lý bởi máy chủ.

# CHƯƠNG 2 : NGÔN NGỮ HTML

## (HyperText Markup Language)

### I. GIỚI THIỆU

#### 1. Giới thiệu về HTML

☞ Khi chúng ta thiết kế một trang Web dễ dàng dưới dạng các hình ảnh trực quan, nhưng cốt lõi bên trong, cấu trúc của một trang Web được qui định bởi ngôn ngữ siêu liên kết HTML.

☞ HTML là chữ viết tắt của Hypertext Markup Language ( ngôn ngữ đánh dấu siêu văn bản ) là hàng loạt các thẻ (Tag) với *các qui ước* để tạo ra các trang Web và các trang Web được hiển thị bởi những trình duyệt Web ( Web browser ).

# CHƯƠNG 2 : NGÔN NGỮ HTML (HyperText Markup Language)

## I. GIỚI THIỆU

### 1. Giới thiệu về HTML

- ☞ HTML lµ mét ng«n ng÷ m« t¶i tµi liÖu, ®Úñc h×nh thµnh tõ mét d¹ng ng«n ng÷ m« t¶i tµi liÖu tæng qu,t cã tªn lµ SGML (Standard Generalized Markup Language), ®©y lµ ng«n ng÷ do h·ng IBM ®Ò xÜíng tõ nh÷ng n·m 1960.
- ☞ Thùc chÊt SGML lµ mét ng«n ng÷ m,y tÝnh nh»m m« t¶i tµi liÖu, cho phĐp ta thiÕt lÆp mèi quan hÖ gi÷a c,c thµnh phÇn kh,c nhau trong cÙng mét tµi liÖu, hoÆc gi÷a nhiÒu tµi liÖu thuéc c,c d¹ng d÷ liÖu kh,c nhau

# CHƯƠNG 2 : NGÔN NGỮ HTML

## (HyperText Markup Language)

### I. GIỚI THIỆU

#### 2. Các khái niệm

##### a. Khái niệm TAG

Một Tag của HTML là đơn vị nhỏ nhất có thể trình diễn trên Web bởi nó chỉ định một cách rõ ràng về cách hiển thị nội dung. Các Tag trong HTML thường được gán là các thuộc tính.

Ví dụ: Đó là một tag đánh dấu văn bản màu xanh lá cây trong HTML ta có:

<B> dãy ký tự </B>

# CHƯƠNG 2 : NGÔN NGỮ HTML (HyperText Markup Language)

## I. GIỚI THIỆU

### 2. C,c kh,j niÖm

#### b. Thuéc tÝnh cña Tag

Mgi Tag trong HTML ®Òu cã mét vµi thuéc tÝnh (hay cßn gäi lµ tham sè) dñng ®Ó bæ sung hoÆc thay ®æi c,ch thÓ hiÖn th«ng thUêng cña Tag ®ã.

VÝ dô: ĐÓ hiÓn thÞ h×nh ¶nh trªn trang web ta dñng tag nhU sau:

```
<IMG SRC="background.gif" WIDTH="100" HEIGHT="150">
```

Trong ®ã:

- + ChØ thÞ IMG b,o cho tr×nh duyÖt biÕt b¾t ®Çu hiÓn thÞ h×nh ¶nh
- + Thuéc tÝnh SRC cho biÕt h×nh ¶nh cÇn hiÓn thÞ
- + Thuéc tÝnh WIDTH cã gi, trÞ 100, HEIGHT cã gi, trÞ 150 b,o cho tr×nh duyÖt biÕt ®é réng vµ ®é cao cña h×nh ¶nh cÇn hiÓn thÞ lµ 100 vµ 150

# CHƯƠNG 2 : NGÔN NGỮ HTML (HyperText Markup Language)

## I. GIỚI THIỆU

### 2. CÁC KHÍA NIỀM

#### c. Tag chèo và Tag rỗng

- + Tag chèo: Một tag HTML có thể chỉ định một hành động hoặc cung cấp thông tin cho phần tử HTML. Tag chèo thường bao gồm một khung trống để xác định cách hiển thị hoặc xử lý phần tử.
- + Tag rỗng: tag rỗng là tag HTML không có khung trống. Các tag rỗng thường được sử dụng để tạo khoảng trắng hoặc phân tách các phần tử HTML.

# CHƯƠNG 2 : NGÔN NGỮ HTML

## (HyperText Markup Language)

### I. GIỚI THIỆU

#### 3. Cấu trúc một trang Web ở dạng HTML

Một tệp HTML thường gồm 2 phần chính: phần đầu tiên (Head) và phần chính (Body).

Dưới đây là cấu trúc của một tệp HTML:

```
<HTML>    <-- tag mở đầu tệp HTML
    <HEAD>    <-- tag mở đầu phần Head
                ----- <-- các tag khác thuộc về phần Head      </HEAD>    <-- tag đóng phần Head
    b, o nội dung của phần Head
    <BODY>    <-- tag mở đầu phần Body
                ----- <-- nội dung của phần Body      </BODY>    <-- tag đóng phần Body
    b, o nội dung của phần Body
</HTML>    <-- tag đóng kết thúc tệp HTML
```

# CHƯƠNG 2 : NGÔN NGỮ HTML (HyperText Markup Language)

## II. CÁC TAG CƠ BẢN CỦA HTML

### 1. Các tag trong phần đầu tiên của HTML

#### a. Tag <!DOCTYPE>

Tag này dùng ghi thông tin về version HTML, p đóng trong phần đầu tiên của Web. Thường ngày ta dùng phần đầu tiên trong file HTML.

- Ví dụ:
- <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Draft//EN">
- hay <!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML Level 1//EN">

# CHƯƠNG 2 : NGÔN NGỮ HTML

## (HyperText Markup Language)

### II. CÁC TAG CƠ BẢN CỦA HTML

#### a. Tag `<TITLE>`

§ayı lü lo<sup>1</sup>i tag chøa, dīng ®Ó thiÕt lËp ti<sup>a</sup>u ®Ò cñña tµi liÖu

VÝ dô:

`<Title>VÝ dô cho tiau ®Ò t1o web</Title>`

#### c. Tag `<BASE>`

Thµnh phÇn này dīng ®Ó chØ ®Þnh ®Þa chØ c¬ b¶n cho c,c mèi nèi li<sup>a</sup>n kÕt. Coi nh-  
úr ®ayı lü ®Þa chØ gèc ®Ó t×m c,c ®Þa chØ kh,c.

VÝ dô: `<BASE HREF="http://www.pc3.com/Scripts/">`

# CHƯƠNG 2 : NGÔN NGỮ HTML

## (HyperText Markup Language)

### II. CÁC TAG CƠ BẢN CỦA HTML

#### 1. Các tag trong phần Cơ bản của HTML

##### d. Tag <META>

Tag này cho phép ta tự do đặt nội dung tin vào phần Cơ bản của tài liệu web như: @pa chØ tr¶ lêu thÙ, các bài tin tức bao, thời gian lùm tÙ-i trang web, thời gian quay h¹n thời gian...

Các thuộc tính của tag:

- + Content: gán giá trị cho phần thuộc tính
- + HTTP-EQUIV: kết nối các siêu liên kết với phần header của một Server (HTTP)
- + Name: tên của thuộc tính phong phú siêu liên kết, @Uic xem giềng nhÙ lµ HTTP-EQUIV nếu không @Uic xác định.

Ví dụ: <META HTTP-EQUIV="Refresh" Content="10">

Tag này bao cho trách nhiệm duy trì bài viết trong vòng 10 giây trước khi trang web mới load.

# CHƯƠNG 2 : NGÔN NGỮ HTML (HyperText Markup Language)

## II. CÁC TAG CƠ BẢN CỦA HTML

### 1. Các tag trong phần Cơ bản của HTML

#### e. Tag <SCRIPT>

§Cy lµ tag chøa, dñng ®Ó chøa c,c hµm (hoÆc c,c thñ tôc) ®Ùic viÕt b»ng JavaScript (hoÆc VBScript). Tag nµy cã thuéc tÝnh Language ®Ó chØ ra ng«n ng÷ ®Ùic sô dông lµ ng«n ng÷ g×.

VÝ dô:

```
<SCRIPT LANGUAGE="JavaScript">  
    function Vidu(){  
        //c,c lÖnh cñh hµm n»m ë ®Óy.  
    }  
</SCRIPT>
```

# CHƯƠNG 2 : NGÔN NGỮ HTML (HyperText Markup Language)

## II. CÁC TAG CƠ BẢN CỦA HTML

2. Các Tag trong phần thô sơ lõi

### a. Tag BODY

§ Có thể lõi tag chứa, định rõ bao lây các tag khác trong phần thô sơ chính lõi HTML.

Cách số đông:

```
<BODY>  
<!--các tag khác nhau-->  
</BODY>
```

# CHƯƠNG 2 : NGÔN NGỮ HTML

## (HyperText Markup Language)

### II. CÁC TAG CƠ BẢN CỦA HTML

#### 2. Các Tag trong phông nền túi lõi

Thuộc tính	Chức năng
Bgcolor="#RRBBGG"	Xác định màu nền (Background) cho trang web, tên màu nền phải bao gồm ký tự #, tiếp sau là 6 chữ cái, mỗi chữ cái là 1 số Hexa (từ 0 đến F) đứng trước các ký tự #. Các ký tự #, tiếp sau là 6 chữ cái, mỗi chữ cái là 1 số Hexa (từ 0 đến F) đứng trước các ký tự #.
BACKGROUND="Image"	Xác định hình nền cho trang web
COLOR="#RRBBGG"	Xác định màu của văn bản trong phông nền túi lõi, tên màu phải bao gồm ký tự #, tiếp sau là 6 chữ cái, mỗi chữ cái là 1 số Hexa (từ 0 đến F) đứng trước các ký tự #.
Topmargin="n"	Canh lề trên cho trang web (giá trị mặc định bằng Pixel)

# CHƯƠNG 2 : NGÔN NGỮ HTML

## (HyperText Markup Language)

### II. CÁC TAG CƠ BẢN CỦA HTML

#### 2. Các Tag trong phần thô sơ nhất

##### b. Tag HEADING

§Yêu l<sup>u</sup> lo<sup>1</sup>i tag ch<sup>ø</sup>a d<sup>ë</sup>ng Ó l<sup>um</sup> n<sup>æ</sup>i b<sup>Æ</sup>t m<sup>ét</sup> ti<sup>a</sup>u Ò hay m<sup>ét</sup> m<sup>ô</sup>c n<sup>u</sup>o ã trong trang web.

C<sup>ã</sup> 6 Heading t<sup>-</sup>ng øng v<sup>ii</sup> 6 m<sup>ô</sup>c ré kh<sup>,c</sup> nhau, v<sup>ii</sup> ré Ưu ti<sup>a</sup>n g<sup>i</sup>¶m d<sup>ç</sup>n t<sup>õ</sup> 1 Õn 6.

Trong HTML ta s<sup>ö</sup> d<sup>ông</sup> tag <Hn> Ó th<sup>ó</sup> hi<sup>Õn</sup> Heading, trong ã n l<sup>u</sup> m<sup>ét</sup> s<sup>è</sup> i t<sup>õ</sup> 1-> 6.

# CHƯƠNG 2 : NGÔN NGỮ HTML

## (HyperText Markup Language)

### II. CÁC TAG CƠ BẢN CỦA HTML

#### 2. Các Tag trong phần thô của HTML

##### b. Tag HEADING

Cách số dòng	Thứ hiđn trn Browser
<H1>Tiêu đề 1</H1>	Tiêu đề 1
<H2>Tiêu đề 2</H2>	Tiêu đề 2
<H3>Tiêu đề 3</H3>	Tiêu đề 3
<H4>Tiêu đề 4</H4>	Tiêu đề 4
<H5>Tiêu đề 5</H5>	Tiêu đề 5
<H6>Tiêu đề 6</H6>	Tiêu đề 6

# CHƯƠNG 2 : NGÔN NGỮ HTML (HyperText Markup Language)

## II. CÁC TAG CƠ BẢN CỦA HTML

### 2. Các Tag trong phần thôon tui liêu

#### c. Tag Table

+ §Ó m« t¶ b¶ng ta dïng Tag <TABLE> vµ </TABLE>. Mçi hñung trong b¶ng ®íc x,c ®þnh b»ng cÆp tag <TR> vµ </TR>, mçi « cña b¶ng ®íc x,c ®þnh b»ng cÆp tag <TD> vµ </TD>.

+ Muèn x,c ®þnh tiªu ®Ò cña b¶ng ta sö dông tag <CAPTION> vµ </CAPTION>

+ Ngoµi ra cã thÓ sö dông c,c thuéc tÝnh HEIGHT, WIDTH, BORDER ®Ó thay ®æi ®é cao, ®é réng, vµ ®Uêng viÒn cña b¶ng

# CHƯƠNG 2 : NGÔN NGỮ HTML (HyperText Markup Language)

## II. CÁC TAG CƠ BẢN CỦA HTML

### 2. Các Tag trong phần th@n t@i liÖu

#### c. Tag Table

VÝ dô:

<TABLE BORDER=1>

```
<CAPTION>VÝ dô vÒ c, ch t@o b@ng </CAPTION>
<TR>
    <TH>STT</TH>
    <TH>Néi dung</TH>
</TR>
<TR>
    <TD>1</TD>
    <TD>Néi dung « thø 2 hñung 2</TD>
</TR>
<TR>
    <TD>2</TD>
    <TD>Néi dung « thø 2 hñung 3</TD>
</TR>
<TR>
    <TD>3</TD>
    <TD>Néi dung « thø 2 hñung 4</TD>
</TR>
</TABLE>
```

Ví dụ về cách tạo bảng

STT	Nội dung
1	Nội dung ở thứ 2 hàng 2
2	Nội dung ở thứ 2 hàng 3
3	Nội dung ở thứ 2 hàng 4

# CHƯƠNG 2 : NGÔN NGỮ HTML (HyperText Markup Language)

## II. CÁC TAG CƠ BẢN CỦA HTML

### 2. Các Tag trong phần th@n t@i liÖu

#### d. Tag Form

+ BiÖu mÉu cho phĐp ta tËp hîp c,c kiÓu d÷ liÖu tõ trang web vµ g i l n Server x o l y th ng qua c,c trÙêng nh p liÖu nhÙ Text box, c,c n t ®iÒu khiÓn nhÙ Check box, Radio button, vµ Command button,...

T t c¶ c,c trÙêng n y ® u ph¶i n»m gi a c p tag <FORM> vµ </FORM>

M t Form d÷ liÖu c c c,c thu c t nh sau:

# CHƯƠNG 2 : NGÔN NGỮ HTML (HyperText Markup Language)

## II. CÁC TAG CƠ BẢN CỦA HTML

### 2. Các Tag trong phần thô n tui liêú

#### d. Tag Form

Thuộc tính	Chức năng
NAME="form_name"	Tên của Form
ACTION="URL_execute"	Chỉ định tên của một file trên Server, để cho phép xử lý dữ liệu từ Form này gửi lên.
METHOD="POST/GET"	Xác định phương thức mà Browser sẽ gửi dữ liệu lên Server. Gi, truy cập POST để gửi dữ liệu, GET để lấy dữ liệu từ Server.

# CHƯƠNG 2 : NGÔN NGỮ HTML (HyperText Markup Language)

## II. CÁC TAG CƠ BẢN CỦA HTML

### 2. Các Tag trong phần thônh tin liên

#### d. Tag Form

Trong 1 Form có thể bao gồm nhiều trường để lấy, mỗi trường để lấy  
đều có đặc điểm riêng biệt như INPUT và có 3 thuộc tính: TYPE, NAME, và VALUE.

- + Type : dùng để xác định kiểu truy cập vào trường
- + Name : dùng để xác định tên của trường
- + Value : dùng để xác định giá trị của trường

Ví dụ: <INPUT TYPE="Text" Name="T1" Value="V1">

# CHƯƠNG 2 : NGÔN NGỮ HTML

## (HyperText Markup Language)

## II. CÁC TAG CƠ BẢN CỦA HTML

### 2. Các Tag trong phần thôp tại liӦu

#### d. Tag Form

Text Box	<INPUT TYPE="Text" >	Cho phĐp nhĒp dŨ liӦu chØ træn m�t h�ng
Password box	<INPUT TYPE="Password" >	D�ng ®Ó nhĒp m�t kh�u, khi s�o d�ng tr�ng n�y dŨ liӦu nhĒp v�o s� ®�c hi�n th� b�ng c,c d�u *
Hidden box	<INPUT TYPE="Hidden" >	D�ng ®Ó x,c ®�nh m�t s� gi, tr� b�e sung cho vi�c x�o l�y c�a ch�ng trinh m�u kh�ng c�n hi�n th� ch�ng.
Check box	<INPUT TYPE="Checkbox" >	L�u ch�n nhi�u gi, tr� m�t l�c

# CHƯƠNG 2 : NGÔN NGỮ HTML (HyperText Markup Language)

## II. CÁC TAG CƠ BẢN CỦA HTML

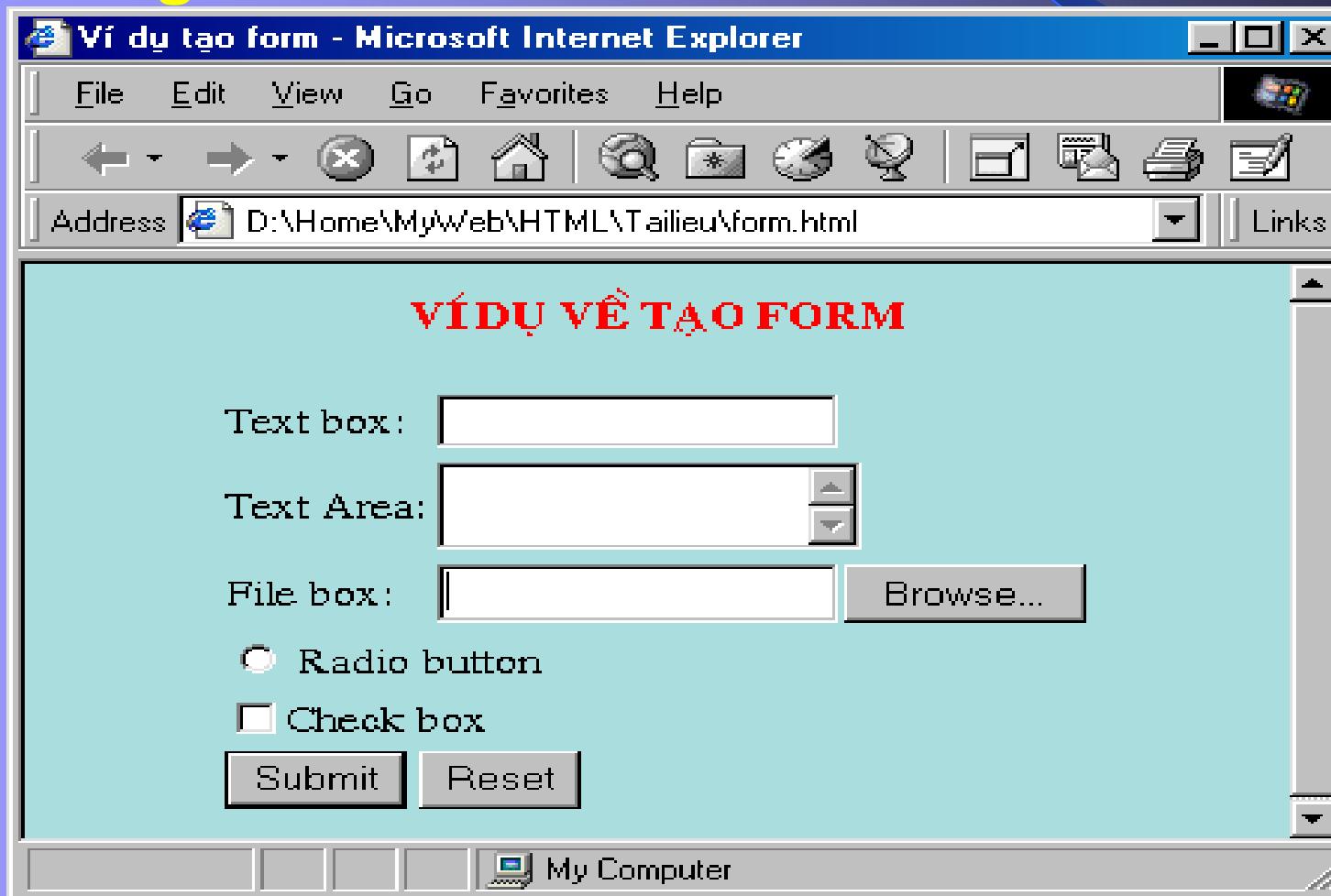
### d. Tag Form

Radio button	<INPUT TYPE="Radio" >	Lùa chän mét trong nhìÙu gi, trØ
Text window	<TEXTAREA>...</TEXTAREA>	Cho phØp nhËp dÙ liÖu træn nhìÙu hµng
File	<INPUT TYPE="File" >	Më cõa sæ liÖt kª c,c file vµ th mõc træn m,y tÝnh ®Ó chän file
Select box	<SELECT Name="select_name"> <OPTION VALUE="nvalue"> Đò mõc chän lùa </OPTION> ..... </SELECT>	Cho phØp chän 1 ®Ò mõc thÝch hîp trong danh s,ch ®æ xuèng cñã c,c ®Ò mõc
Button	<INPUT TYPE="Submit/Reset/Button">	Göi sè liÖu lªn Server (nÕu Type="Submit"), hoÆc trå vØ gi, trØ ngÇm ®Þnh cñã c,c trÙêng træn form (nÕu Type="Reset")

# CHƯƠNG 2 : NGÔN NGỮ HTML (HyperText Markup Language)

## II. CÁC TAG CƠ BẢN CỦA HTML

### d. Tag Form



# CHƯƠNG 2 : NGÔN NGỮ HTML

## (HyperText Markup Language)

### II. CÁC TAG CƠ BẢN CỦA HTML

#### d. Tag Form

```
<html>
<head>
<title>VÝ dô t1o form</title>
</head>
<body bgcolor="#AADDAA" topmargin=10>
<p align="center">
<font color="Red" size="3" face=".VnTimeH">Vi du ve tao Form</font>
<FORM name="form1" ACTION="Xem.php" METHOD="post" >
<table>
<tr>
<td>Text box:</td>
<td><input type="text" name="m_textbox" size=20></td>
</tr>
<tr>
<td>Text Area: </td>
<td>
<TEXTAREA name="m_area" style="font-family:.Vntime" cols=20 rows=2>
</TEXTAREA></td>
</tr>
```

# CHƯƠNG 2 : NGÔN NGỮ HTML

## (HyperText Markup Language)

### II. CÁC TAG CƠ BẢN CỦA HTML

#### d. Tag Form

```
<tr>
<td>File box: </td>
<td><INPUT TYPE="file" NAME="m_file"></td>
</tr>
<tr>
<td colspan=2><INPUT type="radio" name="m_radio"> Radio button</td>
</tr>
<tr>
<td colspan=2><INPUT type="checkbox" name="m_check">Check box</td>
</tr>
<tr>
<td colspan=2>
<INPUT TYPE="submit" name="m_sub" VALUE="Submit">
<INPUT TYPE="submit" name="m_res" VALUE="Reset">
<INPUT TYPE="hidden" NAME="m_hidden"
</td>
</tr>
</table>
</Form>
</p>
</body>
</html>
```

# CHƯƠNG 2 : NGÔN NGỮ HTML (HyperText Markup Language)

## II. CÁC TAG CƠ BẢN CỦA HTML

### 2. Các Tag trong phần th@n t@i liÖu

#### e. Tag Frame

Môc ®Ých cñã Frame lµ nh»m chia cõa sæ tr×nh duyÖt ra lµm hai hay nhiÒu vñng nhá h¬n, ®Ó khi ®ã cã thÓ t¶i c,c t@i liÖu HTML kh,c nhau vµo trong mçi vñng gióp ngUêi sö dông cã thÓ theo dãi nhiÒu trang web ®ång thêi.

Mçi vñng nhá gäi lµ mÔt Frame, tËp hîp c,c vñng nµy gäi lµ Frameset.

§Ó t¹o c,c Frame sö dông cÆp tag <FRAMESET> vµ </FRAMESET>, cÆp tag nµy cã hai thuéc tÝnh c¬ b¶n lµ ROWS vµ COLS dñng ®Ó m« t¶i sè hµng vµ sè cét cÇn chia thµnh Frame.

# CHƯƠNG 2 : NGÔN NGỮ HTML (HyperText Markup Language)

## II. CÁC TAG CƠ BẢN CỦA HTML

### 2. Các Tag trong phần th@n t@i l@i

#### e. Tag Frame

Ví dô: <FRAMESET COLS="150,\*">  
                <FRAME NAME="menu" SRC="Menu.htm">  
                <FRAME NAME="main" SRC="Main.htm">  
       </FRAMESET>

+ Số n m· tr@n chia c@a s@e ra l@um hai cét, cét thø nh@t r@eng 150 pixel, cét thø hai l@u k@o@ng tr@eng c@n l@i c@n c@a s@e.

+ Trong c@p tag <FRAMESET> c@n chøa c,c tag <FRAME> d@ng @Ó x,c @Pnh c,c trang web s@i @-ic hi@n th@ trong m@i Frame khi Frameset @U@c n@p l@n l@Cn @Çu ti@n.

+ M@i Frame c@ thu@c t@Ynh NAME d@ng @Æt t@n cho Frame @ã, @ång th@i l@um @Ých @Ó c,c trang web kh,c nh@y @Ön frame @ã

# CHƯƠNG 2 : NGÔN NGỮ HTML

## (HyperText Markup Language)

### II. CÁC TAG CƠ BẢN CỦA HTML

#### e. Tag Frame

File Frame.htm cỡ nội dung:

- <html>
- <head>
- <title>Vý dô t'ô frames</title>
- </head>
- <frameset border="false" frameborder="0" rows="80,\*">
- <frame name="frame1" src="frame1.htm" scrolling="no" noresize>
- <frame name="frame2" src="frame2.htm" scrolling="auto" noresize>
- </frameset>
- <noframes>
- <body>
- </body>
- </noframes>
- </html>

# CHƯƠNG 2 : NGÔN NGỮ HTML

## (HyperText Markup Language)

### II. CÁC TAG CƠ BẢN CỦA HTML

#### e. Tag Frame

- File frame1.htm cỡ nội dung:

```
<html>
<head>
<title>Frame 1</title>
</head>
<body bgcolor="#AADDFF" topmargin=10>
<center>
<font color="Red" size="3" face=".VnTimeH">Frame 1</font></b>
<hr>
Vý dô t'о frame 1
</center>
</body>
</html>
```

# CHƯƠNG 2 : NGÔN NGỮ HTML

## (HyperText Markup Language)

### II. CÁC TAG CƠ BẢN CỦA HTML

#### e. Tag Frame

- File frame2.htm cã néi dung:

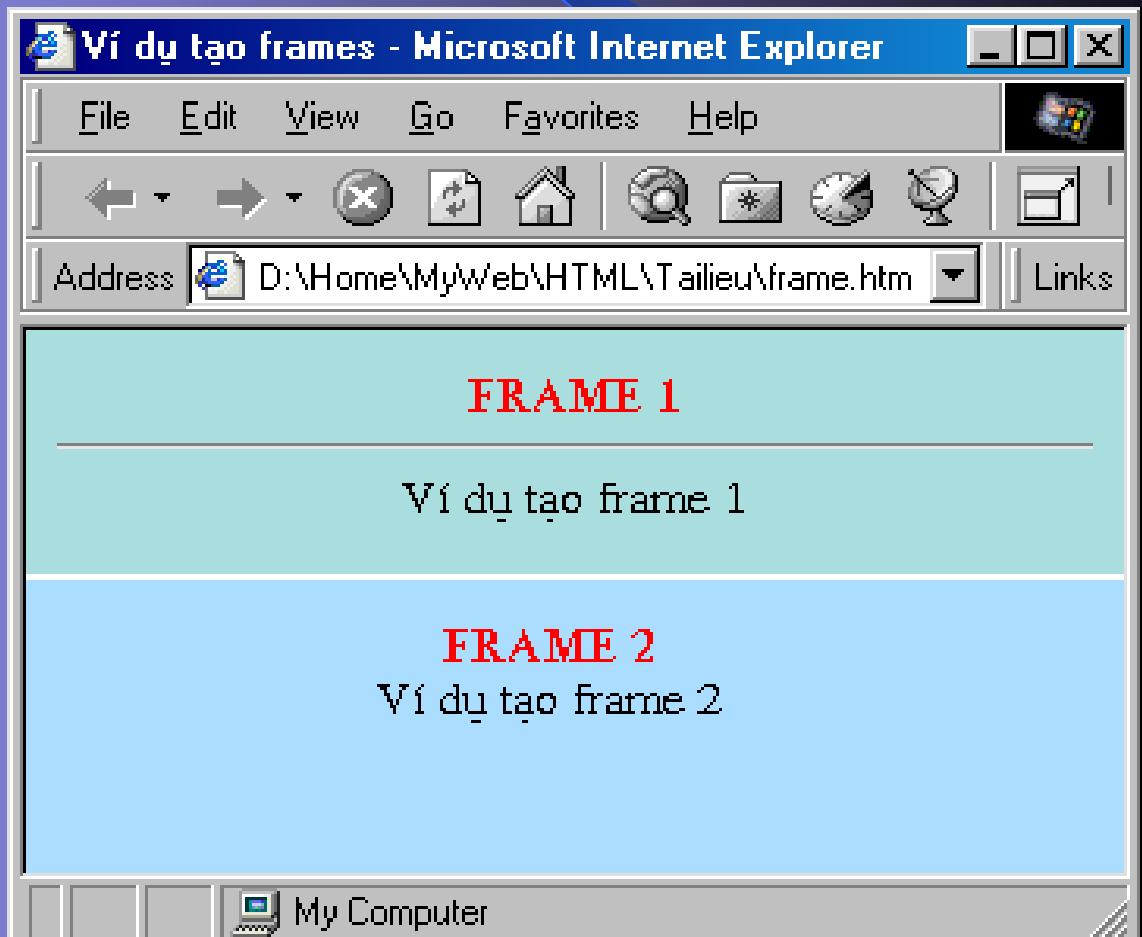
```
<html>
<head>
<title>Frame 2</title>
</head>
<body bgcolor="#AADDFF" topmargin=10>
<center>
<font color="Red" size="3" face=".VnTimeH">Frame 2</font></b>
<hr>
<p>Vý dô t1o frame 2</p>
</center>
</body>
</html>
```

# CHƯƠNG 2 : NGÔN NGỮ HTML (HyperText Markup Language)

## II. CÁC TAG CƠ BẢN CỦA HTML

### e. Tag Frame

Kết quả hiển thị sau:



# Ch<sup>n</sup>g 3: Ng<sup>o</sup>n ng<sup>o</sup> VBScript v<sup>u</sup> JavaScript

## I. GIÓI THIỆU

- + Script c<sup>a</sup> th<sup>O</sup> xem nh<sup>U</sup> m<sup>et</sup> c<sup>o</sup>ng c<sup>o</sup> nh<sup>U</sup> m<sup>et</sup> h<sup>c</sup> tr<sup>i</sup> ng<sup>U</sup> i<sup>Ep</sup> tr<sup>x</sup>nh t<sup>o</sup> ra c<sup>,c</sup> <sup>o</sup>ng d<sup>o</sup>ng nhanh ch<sup>a</sup>ng, <sup>R</sup>on gi<sup>ll</sup>n ch<sup>o</sup> kh<sup>c</sup>ng mang t<sup>Y</sup>nh chuy<sup>a</sup>n nghi<sup>Op</sup> nh<sup>U</sup> m<sup>et</sup> ng<sup>o</sup>n ng<sup>o</sup> VBScript l<sup>u</sup> m<sup>et</sup> trong nh<sup>o</sup>ng c<sup>o</sup>ng c<sup>o</sup> nh<sup>U</sup> v<sup>E</sup>y. V<sup>i</sup>i c<sup>,c</sup> t<sup>Ep</sup> l<sup>On</sup>h c<sup>i</sup>ng v<sup>i</sup>i c<sup>,c</sup> <sup>R</sup>ei t<sup>U</sup><sup>l</sup>ing h<sup>c</sup> tr<sup>i</sup> s<sup>1/2</sup>n trong Internet Explorer v<sup>u</sup> Internet Information Server, VBScript cho ph<sup>D</sup>p ng<sup>e</sup>i s<sup>o</sup> d<sup>o</sup>ng vi<sup>Ot</sup> c<sup>,c</sup> ch<sup>U</sup><sup>-</sup>ng tr<sup>x</sup>nh ch<sup>Y</sup>y tr<sup>a</sup>n web v<sup>i</sup>i c<sup>o</sup> ph<sup>,p</sup> ho<sup>m</sup>n to<sup>m</sup>n gi<sup>eng</sup> Visual Basic.
- + JavaScript c<sup>o</sup>ng l<sup>u</sup> m<sup>et</sup> lo<sup>1</sup>i Script cho ph<sup>D</sup>p ph<sup>,t</sup> tri<sup>On</sup> c<sup>,c</sup> <sup>o</sup>ng d<sup>o</sup>ng tr<sup>a</sup>n Internet <sup>o</sup> c<sup>T</sup> ph<sup>Y</sup>a Client v<sup>u</sup> Server b<sup>»</sup>ng c<sup>,ch</sup> vi<sup>Ot</sup> c<sup>,c</sup> ch<sup>U</sup><sup>-</sup>ng tr<sup>x</sup>nh v<sup>i</sup>i c<sup>o</sup> ph<sup>,p</sup> ho<sup>m</sup>n to<sup>m</sup>n gi<sup>eng</sup> Java.

# Ch<sup>n</sup>g 3: Ng<sup>c</sup>n ng<sup>d</sup> VBScript v<sup>p</sup> JavaScript

## II. Ng<sup>c</sup>n ng<sup>d</sup> JavaScript

- + T<sup>U</sup>ng t<sup>u</sup> nh<sup>U</sup> VBScript, JavaScript l<sup>u</sup> m<sup>t</sup> ng<sup>c</sup>n ng<sup>d</sup> script, d<sup>i</sup>ng ®Ó ph<sup>t</sup> triÓn c<sup>,c</sup> <sup>c</sup>ng d<sup>o</sup>ng Internet <sup>ë</sup> Client v<sup>p</sup> Server.
- + JavaScript c<sup>a</sup> thÓ viÓt k<sup>l</sup>m trong file HTML ®Ó x<sup>C</sup>y d<sup>u</sup>ng m<sup>t</sup> trang Web chÆt ch<sup>l</sup>, phong phó v<sup>O</sup> n<sup>e</sup>i dung v<sup>p</sup> h<sup>x</sup>nh th<sup>o</sup>c nh<sup>U</sup> : kiÓm tra tÝnh t<sup>U</sup>ng thÝch v<sup>p</sup> th<sup>c</sup>ng b<sup>,o</sup> l<sup>c</sup>i trong qu<sup>s</sup>, tr<sup>x</sup>nh nh<sup>Ep</sup> liÓu...
- + C<sup>,ch</sup> viÓt JavaScript trong m<sup>t</sup> trang HTML nh<sup>U</sup> sau:

```
<Script Language="JavaScript">  
    //C,c hum cna JavaScript ®Æt ë ®Cy  
</Script>
```

# Ch<sup>n</sup>g 3: Ng<sup>c</sup>n ng<sup>d</sup> VBScript v<sup>u</sup> JavaScript

## II. Ng<sup>c</sup>n ng<sup>d</sup> JavaScript

### 1. M<sup>e</sup>t s<sup>e</sup> ®<sup>p</sup>nh ngh<sup>U</sup>a

#### 1.1. C<sup>,c</sup> kiÓu d<sup>d</sup> liÖu :

Trong JavaScript c<sup>a</sup> c<sup>,c</sup> kiÓu d<sup>d</sup> liÖu :

- **Integer:** G<sup>a</sup>m c<sup>,c</sup> s<sup>e</sup>  $\in \{1, 2, 3, \dots, 10, 16, 8, 2\}$
- VÝ dô: 42, 0xFFFF, -345...
- **S<sup>e</sup> th<sup>u</sup>c :** VÝ dô: 3.1415, -3.1E12, 2e-12 ... l<sup>u</sup> ý E, e <sup>®</sup>Øu c<sup>a</sup> ý ngh<sup>U</sup>a nh<sup>U</sup> nhau.
- **Boolean:** C<sup>a</sup> gi, true hoÆc false
- **Chuçi (String):** M<sup>e</sup>t chuçi l<sup>u</sup> m<sup>e</sup>t d<sup>y</sup> c<sup>,c</sup> k<sup>y</sup> t<sup>u</sup> n»m gi<sup>d</sup>a hai cÆp dÊu ("") hoÆc (""). VÝ dô:  
"blah", 'blah', "1234", “dong mot\n dong hai”.

# Ch<sup>u</sup>ngh<sup>o</sup> 3: Ng<sup>u</sup>n ng<sup>u</sup> VBScript v<sup>u</sup> JavaScript

## II. Ng<sup>u</sup>n ng<sup>u</sup> JavaScript

### 1. M<sup>u</sup>t s<sup>e</sup> ®<sup>u</sup>nh ngh<sup>u</sup>a

#### 1.2. Bi<sup>u</sup>nh

+ Bi<sup>u</sup>nh l<sup>u</sup>p m<sup>u</sup>t ®<sup>u</sup>i l<sup>u</sup>ng d<sup>u</sup>ng ®<sup>u</sup>O l<sup>u</sup>u tr<sup>u</sup>d<sup>u</sup> li<sup>u</sup> m<sup>u</sup> gi, tr<sup>u</sup>b c<sup>u</sup>na n<sup>a</sup>c<sup>a</sup> th<sup>o</sup> thay ®<sup>u</sup>ei trong qu<sup>u</sup>, tr<sup>u</sup>xnh th<sup>u</sup>c hi<sup>u</sup>nh ch<sup>u</sup>ng tr<sup>u</sup>xnh.

+ §Ó bi<sup>u</sup>u di<sup>u</sup>nh bi<sup>u</sup>nh ng<sup>u</sup>rei ta s<sup>o</sup>d<sup>u</sup>ng t<sup>a</sup>n bi<sup>u</sup>nh, khi khai b<sup>o</sup>m<sup>u</sup>t bi<sup>u</sup>nh th<sup>x</sup> t<sup>a</sup>n bi<sup>u</sup>nh ®<sup>u</sup>c c<sup>u</sup>p ph<sup>u</sup>t m<sup>u</sup>t ®<sup>u</sup>pa ch<sup>u</sup>Ø c<sup>u</sup>th<sup>o</sup> trong b<sup>o</sup>nhí m<sup>u</sup>y t<sup>u</sup>nh.

+ T<sup>a</sup>n bi<sup>u</sup>nh l<sup>u</sup>p chu<sup>u</sup>i g<sup>u</sup>am c<sup>c</sup>ký t<sup>u</sup>, ch<sup>u</sup>s<sup>e</sup>, v<sup>u</sup> d<sup>u</sup>eu g<sup>u</sup>ch d<sup>u</sup>rii ( )

+ T<sup>a</sup>n bi<sup>u</sup>nh trong ng<sup>u</sup>n ng<sup>u</sup> JavaScript c<sup>a</sup> ph<sup>u</sup>on bi<sup>u</sup>Øt ch<sup>u</sup>s<sup>e</sup> hoa v<sup>u</sup> ch<sup>u</sup>s<sup>e</sup> th<sup>u</sup>ng.

+ §Ó ®<sup>u</sup>nh ngh<sup>u</sup>a m<sup>u</sup>t bi<sup>u</sup>nh b<sup>o</sup>ng c<sup>c</sup>ch d<sup>u</sup>ng t<sup>o</sup>kho, var

V<sup>u</sup>y dô: var x=12

Trong JavaScript c<sup>a</sup>th<sup>o</sup> d<sup>u</sup>ng bi<sup>u</sup>nh m<sup>u</sup> kh<sup>u</sup>ng c<sup>c</sup>nh khai b<sup>o</sup>.

# Ch<sup>n</sup>g 3: Ng<sup>c</sup>n ng<sup>d</sup> VBScript v<sup>u</sup> JavaScript

## II. Ng<sup>c</sup>n ng<sup>d</sup> JavaScript

### 1. M<sup>e</sup>t s<sup>e</sup> ®<sup>p</sup>n<sup>h</sup> ngh<sup>U</sup>a

#### 1.3. C<sup>c</sup>c ph<sup>D</sup>p to<sup>,n</sup>

+ Các phép toán số học

+ C<sup>c</sup>c ph<sup>D</sup>p to<sup>,n</sup> so s<sup>,nh</sup>

+ C<sup>c</sup>c ph<sup>D</sup>p to<sup>,n</sup> thao t<sup>c</sup>c tr<sup>a</sup>n bit

+ C<sup>c</sup>c ph<sup>D</sup>p to<sup>,n</sup> logical

### 2 C<sup>h</sup>u tr<sup>c</sup>c c<sup>c</sup> l<sup>Ö</sup>nh ®<sup>i</sup>Øu khiØn trong JavaScript

+ L<sup>^</sup>nh if... Else

+ L<sup>^</sup>nh for

+ L<sup>^</sup>nh while

.....

# Ch<sup>u</sup>ngh<sup>o</sup>ng 3: Ng<sup>u</sup>ng<sup>u</sup> VBScript v<sup>u</sup> JavaScript

## III. Xö lý c<sup>h</sup>c sù kiÖn tr<sup>a</sup>n trang Web

### 1. Sù kiÖn v<sup>u</sup> ®èi tU<sup>u</sup>ng

+ §bnh nghÜa sù kiÖn: sù kiÖn l<sup>u</sup>m mét h<sup>u</sup>nh<sup>u</sup> ®éng x<sup>u</sup>y ra khi ngU<sup>u</sup>êi sö d<sup>o</sup>ng l<sup>u</sup>m mét viÖc g<sup>x</sup> ®ã tr<sup>a</sup>n trang web.

VÝ dô : KÝch (Click) mét nót, thay ®æi (Change) d<sup>o</sup> liÖu trong trU<sup>u</sup>ng Textbox, di chuyÖn con trá chuét (Mouse) tr<sup>a</sup>n mét si<sup>a</sup>u li<sup>a</sup>n kÖt (Hyperlink),...

+ V<sup>u</sup> nhU<sup>u</sup> v<sup>u</sup> Ý b<sup>u</sup>n ph<sup>u</sup>i tiÖn h<sup>u</sup>nh<sup>u</sup> viÖt c<sup>h</sup>c l<sup>u</sup>nh<sup>u</sup> ®Ó khi c<sup>h</sup>c sù kiÖn n<sup>u</sup>y x<sup>u</sup>y ra th<sup>x</sup> ®bnh hU<sup>u</sup>ng ngU<sup>u</sup>êi sö d<sup>o</sup>ng ®, p<sup>u</sup> øng ®óng nhu cÇu cñ<sup>u</sup> øng d<sup>o</sup>ng nhU<sup>u</sup> : th<sup>u</sup>ng b<sup>,o</sup> nhËp sai d<sup>o</sup> liÖu, d<sup>o</sup> liÖu chU<sup>u</sup>a ®Çy ®ñ,...

# Ch<sup>u</sup>ngh<sup>o</sup>ng 3: Ng<sup>u</sup>ng<sup>u</sup> VBScript v<sup>u</sup> JavaScript

## III. X<sup>o</sup> lý c<sup>h</sup>c sù kiÖn tr<sup>a</sup>n trang Web

### 1. Sù kiÖn v<sup>u</sup> ®èi t<sup>u</sup>ring

- + ViÖc viÖt c<sup>h</sup>c lÖnh ®Ó ®,p øng c<sup>h</sup>c sù kiÖn nøy gäi l<sup>u</sup> ®iÒu khiÖn sù kiÖn.  
Trong JavaScript th<sup>x</sup> c<sup>h</sup>ch sö döng c<sup>h</sup>c ®iÒu khiÖn nhU sau: **trUíc ti<sup>a</sup>n l<sup>u</sup> t<sup>o</sup> "on"** v<sup>u</sup> theo sau l<sup>u</sup> t<sup>a</sup>n sù kiÖn
- VÝ dô: Khi kÝch mét nót th<sup>x</sup> sù kiÖn Click x<sup>t</sup>y ra, ®Ó ®,p øng sù kiÖn nøy b<sup>1</sup>n dïng ®iÒu khiÖn onClick.
- + B<sup>t</sup>ng dU<sup>u</sup>ii ®C<sup>y</sup> liÖt k<sup>a</sup> c<sup>h</sup>c sù kiÖn v<sup>u</sup> c<sup>h</sup>c ®iÒu khiÖn sù kiÖn t<sup>u</sup>ng øng cho tÊt c<sup>h</sup>c sù kiÖn tr<sup>a</sup>n web nhU sau :

Sự kiện	Sự kiện	Mô tả	Đối tượng
blur	onBlur	Người sử dụng di chuyển focus ra khỏi phần tử trang web.	Textbox, Textarea, Select list
click	onClick	Người sử dụng click vào một phần tử trang web.	Button, Radio button, Check button, Submit button, Reset button, Link
change	onChange	Người sử dụng thay đổi giá trị của phần tử trang web.	Textbox, Textarea, Select list
focus	onFocus	Người sử dụng click vào một phần tử trang web.	Textbox, Textarea, Select list
select	onSelect	Người sử dụng chọn dữ liệu trong các phần tử trang web.	Textbox, Textarea
load	onLoad	Nâng cấp trang web vào trình duyệt.	Document
unload	onUnload	Thoát khỏi một trang web.	Document
mouseover	onMouseOver	Người sử dụng di chuyển con trỏ chuột trên một liên kết.	Link
submit	onSubmit	Người sử dụng click vào nút submit.	Submit button

# Ch<sup>n</sup>g 3: Ng<sup>c</sup>n ng<sup>d</sup> VBScript v<sup>m</sup> JavaScript

## III. X<sup>o</sup> l<sup>y</sup> c,c s<sup>u</sup> ki<sup>Ö</sup>n tr<sup>a</sup>n trang Web

V<sup>Y</sup> d<sup>ô</sup>: Đ<sup>c</sup>y l<sup>u</sup> m<sup>et</sup> trang HTML ho<sup>m</sup>n ch<sup>Ø</sup>nh khi k<sup>Ý</sup>ch v<sup>m</sup>o n<sup>o</sup>t "Click here" s<sup>i</sup> xu<sup>Ê</sup>t hi<sup>Ö</sup>n c<sup>c</sup>u th<sup>«</sup>ng b<sup>,o</sup> "Hello..."

- <Script language="JavaScript">
- **function thongbao()**
- {
- **alert("Hello...");**
- }
- </Script>

# Ch<sup>u</sup>ngh<sup>o</sup> 3: Ng<sup>o</sup>n ng<sup>o</sup> VBScript v<sup>u</sup> JavaScript

## III. X<sup>o</sup> l<sup>y</sup> c<sup>,c</sup> s<sup>u</sup> ki<sup>On</sup> tr<sup>a</sup>n trang Web

V<sup>Y</sup> d<sup>o</sup>: Đ<sup>C</sup>y l<sup>u</sup> m<sup>et</sup> trang HTML ho<sup>m</sup>n ch<sup>O</sup>nh khi k<sup>Y</sup>ch v<sup>u</sup>o n<sup>o</sup>t "Click here" s<sup>i</sup> xu<sup>E7</sup> hi<sup>On</sup> c<sup>@</sup>u th<sup>u</sup>ng b<sup>,o</sup> "Hello..."

- <BODY>
- <Form name="form1">
- <input type="button" value="Click here" onClick="thongbao()">
- </Form>
- </BODY>

# Ch<sup>u</sup>ngh<sup>o</sup>ng 3: Ng<sup>u</sup>ng<sup>u</sup> VBScript v<sup>u</sup> JavaScript

## III. X<sup>o</sup> lý c<sup>o</sup>c s<sup>u</sup> kiÖn tr<sup>a</sup>n trang Web

### 1. S<sup>u</sup> kiÖn v<sup>u</sup> ®<sup>u</sup> tU<sup>u</sup>ing

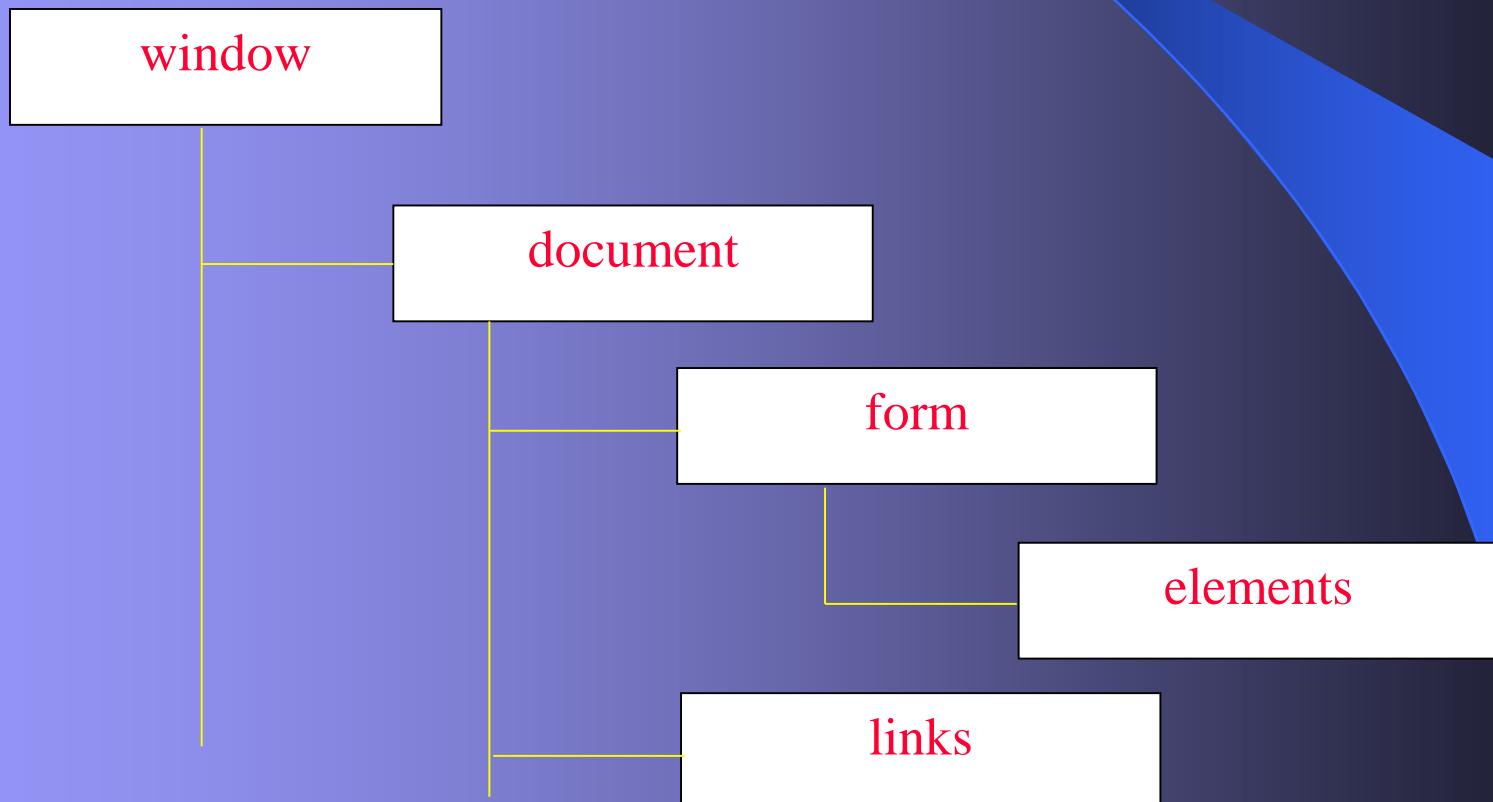
§Ó ®, p øng ®óng s<sup>u</sup> kiÖn cho c<sup>o</sup>c ®<sup>u</sup>i tU<sup>u</sup>ing cÇn ®iÖu khiÖn, b<sup>1</sup>n c<sup>a</sup> thÓ h<sup>x</sup>nh dung ra c<sup>o</sup>c ®<sup>u</sup>i tU<sup>u</sup>ing tr<sup>a</sup>n m<sup>e</sup>t trang web nh<sup>u</sup>s sau:

- + §<sup>u</sup>i tU<sup>u</sup>ing lín nh<sup>Ê</sup>t c<sup>a</sup>n tr<sup>x</sup>nh duyÖt l<sup>u</sup> window
- + Trong window ch<sup>o</sup>a c<sup>o</sup>c t<sup>u</sup>i liÖu HTML (m<sup>u</sup>i t<sup>u</sup>i liÖu l<sup>u</sup> m<sup>e</sup>t ®<sup>u</sup>i tU<sup>u</sup>ing document)
- + Trong document c<sup>a</sup> thÓ ch<sup>o</sup>a nhiÖu Form (c<sup>a</sup> ®<sup>u</sup>i tU<sup>u</sup>ing l<sup>u</sup> form), v<sup>u</sup> c<sup>o</sup>c li<sup>a</sup>n k<sup>Ô</sup>t (c<sup>a</sup> ®<sup>u</sup>i tU<sup>u</sup>ing l<sup>u</sup> links)
- + Trong Form c<sup>a</sup> thÓ ch<sup>o</sup>a nhiÖu phÇn tö (c<sup>a</sup> ®<sup>u</sup>i tU<sup>u</sup>ing l<sup>u</sup> elements)

# Ch<sup>u</sup>ngh<sup>o</sup>ng 3: Ng<sup>u</sup>ng<sup>o</sup> VBScript v<sup>u</sup> JavaScript

## III. X<sup>o</sup> l<sup>y</sup> c<sup>,c</sup> s<sup>u</sup> kiÖn tr<sup>a</sup>n trang Web

### 1. S<sup>u</sup> kiÖn v<sup>u</sup> ®<sup>e</sup>i tU<sup>u</sup>ng



# Ch-ng 3: Ng«n ng÷ VBScript vµ JavaScript

## III. Xö lý c,c sù kiÖn trªn trang Web

### 2. C,c phU-ng thøc vµ thuéc tÝnh cñaa c,c ®èi tU'ing

+ §Ó tham chiÖu ®Ön c,c thuéc tÝnh (hay c,c ph-ng thøc) cñaa mét ®èi tU'ing nµo ®ã b¹n cÇn ®Æc t¶ tªn cñaa ®èi tU'ing ®ã cïng víi c,c ®èi tîng cÊp cao h-n nã.

VÝ dô: ĐÆ g,n gi, trØ xØu chuçi "Hello" cho 1 trªng textbox (cã tªn lµ txtbox) trong mét form (cã tªn lµ form1) b¹n sö dÔnh sau:

```
document.form1.txtbox.value="Hello"
```

+ Trong ®ã : **value** lµ thuéc tÝnh cñaa ®èi tîng **textbox**, c,c thµnh phÇn **document**, **form1** lµ c,c ®èi tîng cha cñaa nã.

### III. Xö lý c,c sù kiÖn trªn trang Web

#### 2. C,c phU-ng thøc vµ thuéc tÝnh cña c,c ®èi tU-ing

##### 2.1 §èi tU-ing window

- \* C,c phU-ng thøc:

PhU-ng thøc	Môc ®Ých
<code>alert("message")</code>	ĐƯa ra hép thoïi th«ng b,o víi néi dung lµ <i>message</i>
<code>confirm("message")</code>	ĐƯa ra hép thoïi yªu cÇu ngUêi sö dông x,c nhËn l¹i c«ng viÖc cÇn hµnh ®éng (chän <b>OK</b> hay <b>Cancel</b> ). PhU-ng thøc nµy tr¶ vÒ gi, tr¶ <b>true</b> (khi chän OK) hoÆc <b>false</b> (khi chän Cancel)
<code>prompt("message",variable)</code>	ĐƯa ra hép thoïi cho phĐp ngUêi sö dông nhËp dÙ liÖu ®Çu vµo, dÙ liÖu sau khi nhËp ®Uîc g,n cho biÖn <i>variable</i> . <i>message</i> lµ dßng th«ng b,o.
<code>close()</code>	Đãng cõa sæ trình duyÖt
<code>open("URL","name")</code>	Më trang web cä ®pa chØ lµ <i>URL</i> trong cõa sæ cä tªn lµ <i>name</i>

### III. Xo lý c, c sù kiÖn trªn trang Web

#### 2. C, c phÙng thøc vµ thuéc tÝnh cña c, c ®èi tÙng

##### 2.1 §èi tÙng window

- \* C, c thuéc tÝnh:

Thuéc tÝnh	Môc ®Ých
status	ChØ ®Pnh th«ng tin n»m trªn thanh tr¹ng thi ë cuèi cöa sæ

- §Ó khi n¹p trang web lªn mµ thanh trang thi cã dßng th«ng b,o "Chµo c,c b¹n..." thx b¹n dïng tag <BODY> nhÚ sau:

```
<BODY onLoad="window.status='Chµo c,c b¹n...'">
```

- \* C, c ®iÒu khiÓn sù kiÖn:

§èi tÙng window cã hai ®iÒu khiÓn sù kiÖn lµ **onLoad** vµ **onUnload**

# Ch-ng 3: Ng«n ng÷ VBScript vµ JavaScript

## III. Xö lý c,c sù kiÖn trªn trang Web

### 2. C,c phÙ-ng thøc vµ thuéc tÝnh cña c,c ®èi tÙring

2.2. §èi tÙring document

2.3. §èi tÙring form

2.4. §èi tÙring links

2.5. §èi tÙring text, textarea, vµ password

2.6. §èi tÙring select (gièng nh listbox)

2.7. §èi tÙring button, submit, reset

2.8. §èi tÙring checkbox, radio

### III. Xö lý c,c sù kiÖn trªn trang Web

#### 2. C,c phU-ng thøc vµ thuéc tÝnh cña c,c ®èi tU-ing

##### 2.1 §èi tU-ing window

- \* C,c phU-ng thøc:

PhU-ng thøc	Môc ®Ých
alert("message")	ĐƯa ra hép thoïi th«ng b,o víi néi dung lµ message
confirm("message")	ĐƯa ra hép thoïi yªu cÇu ngUêi sö dông x,c nhËn l¹i c«ng viÖc cÇn hµnh ®éng (chän OK hay Cancel). PhU-ng thøc nµy tr¶ vÒ gi, tr¶ true (khi chän OK) hoÆc false (khi chän Cancel)
prompt("message",variable)	ĐƯa ra hép thoïi cho phĐp ngUêi sö dông nhËp dÙ liÖu ®Çu vµo, dÙ liÖu sau khi nhËp ®Uîc g,n cho biÖn variable. message lµ dßng th«ng b,o.
close()	Đãng cõa sæ trình duyÖt
open("URL","name")	Më trang web cä ®pa chØ lµ URL trong cõa sæ cä tªn lµ name

### III. Xo lý c, c sù kiÖn trªn trang Web

#### 2. C, c phÙng thøc vµ thuéc tÝnh cña c, c ®èi tÙng

##### 2.1 §èi tÙng window

\* C, c thuéc tÝnh:

Thuéc tÝnh	Môc ®Ých
status	ChØ ®Pnh th«ng tin n»m trªn thanh tr¹ng thi, i ë cuèi cöa sæ

§Ó khi n¹p trang web lªn mµ thanh trang thi, i cã dßng th«ng b, o "Chµo c,c b¹n..." thx b¹n dïng tag <BODY> nhU sau:

```
<BODY onLoad="window.status='Chµo c,c b¹n...'">
```

\* C, c ®iÒu khiÓn sù kiÖn:

§èi tÙng window cã hai ®iÒu khiÓn sù kiÖn lµ **onLoad** vµ **onUnload**

### III. Xo lý c, c sù ki Ôn trªn trang Web

#### 2. C, c phºt ng thøc vµ thuéc tÝnh cña c, c ®èi tÙng

##### 2.2 §èi tÙng document

- \* C, c phºt Ong thÙc:

Phºt ng thøc	Môc ®Ých
write( <i>expression</i> ) hoÆc writeln kh,c write ë chç lµ sau khi viÕt xong thx xuèng hµng tiÕp theo.	ViÕt biÓu thøc <i>expression</i> ra trang web trªn cña tr×nh duyÖt. NÕu cã nhiÒu biÓu thøc cÇn viÕt thx c, c biÓu thøc nµy c, ch nhau dÊu phÈy (). writeln kh,c write ë chç lµ sau khi viÕt xong thx xuèng hµng tiÕp theo.

VÝ dô: §Ó viÕt mét biÓu thøc  $5+6=11$  ra tr×nh duyÖt b¹n sö dÔng có ph,p nhÚt sau:  
`document.write("5 + 6 =",5+6)`

## II. Xử lý các thuộc tính trang Web

### 2. Các phương thức thuộc tính của các thẻ trang

#### 2.2 §èi tÙng document

\* Các thuộc tính:

Thuộc tính	Môc ®Ých
bgcolor="#rrggbb"	Xác ®nh màu nền cho trang
fgcolor="#rrggbb"	Xác ®nh màu chìa cho trang
linkColor="#rrggbb"	Xác ®nh màu cho các liên kết khi chưa click vào them lần đầu
vlinkColor="#rrggbb"	Xác ®nh màu cho các liên kết khi click vào them lần sau
alinkColor="#rrggbb"	Xác ®nh màu cho liên kết khi nã ®truc kÝch href

## II. Xử lý các sự kiện trên trang Web

### 2. Các phương thức thuộc tính của các事件

#### 2.2 Sự kiện document

- Ví dụ: Thay đổi màu nền của trang web khi người dùng nhấp vào tae hiphieu (hay tên) trong « textbox » kích nút "bgColor"

```
<html>
<head>
<script language="JavaScript">
function bgrColor()
{
    document.bgColor=document.form1.txtcolor.value
}
</script>
</head>
```

## II. Xử lý các sự kiện trên trang Web

### 2. Các phương thức vuông góc với nhau của các事件

#### 2.2 Sự kiện document

- Vý ðô: Thay đổi màu nền của trang web khi người dùng nhấp vào bất kỳ mảng nào (hay tan mảng) trong « textbox » kích nút "bgColor"

```
<body>
<form name="form1">
    <input type="text" name="txtcolor">
    <input type="button" value="bgColor" onClick="bgrColor()">
</form>
</body>
</html>
```

## II. Xo lý c,c sù kiÖn trªn trang Web

### 2. C,c phƯ¬ng thøc vµ thuéc tÝnh cña c,c ®èi tÙing

#### 2.3 §èi tÙng form

##### \* C,c phƯ¬ng thÙc

PhƯ¬ng thøc	Môc ®Ých
submit()	Cho thùc thi ®ng dông (tøc lµ b¾t ®Çu gõi sè liÖu lªn Server xo lý)

##### \* C,c thuéc tÝnh

Thuéc tÝnh	Môc ®Ých
Action	§Æc t¶ URL cña Server mµ d÷ liÖu tø form sї ®UÍc gõi ®Ön
Method	§Æc t¶ phƯ¬ng thøc trao ®æi thÙng tin gi÷a Client vµ Server (POST hay GET)
Name	X,c ®Þnh tªn cña form
Target	X,c ®Þnh tªn cña cöa sæ (hay cña frame) mµ khi d÷ liÖu ph¶n hãi vÒ tø Server sї ®UÍc hiÓn thØ trªn cña sæ ®ã

\* C,c ®iÙu khiÓn sù kiÖn : ChØ cã ®iÙu khiÓn sù kiÖn **onSubmit**

### III. Xö lý c,c sù kiÖn trªn trang Web

#### 2. C,c phÙ-ng thøc vµ thuéc tÝnh cña c,c ®èi tÙng

##### 2.4 §èi tÙng links

\* C,c thuéc tÝnh

Thuéc tÝnh	Môc ®Ých
Target	X,c ®Þnh tªn cña cõa sæ (hay cña frame) mµ khi kÝch vµo liªn kÕt thx trang ®íc liªn kÕt sї n¹p vµo cõa sæ ®ã.

\* C,c ®iÒu khiÓn sù kiÖn : Bao gm onClick vµ onMouseOver

VÝ dô sau cho xuÊt hiÖn c u th ng b,o "Click here" ® thanh tr ng th,i khi b n di chuy n con tr  chu t trªn d ng v n b n liªn kÕt "onMouseOver Sample"

```
<a href="" onMouseOver="window.status='Click here'">onMouseOver Sample </a>
```

### III. Xử lý các sự kiện trên trang Web

#### 2. Các phương thức thuộc tính của các thẻ input

##### 2.5 Seite text, textarea, password

\* Các phương thức

Phương thức	Môc đích
blur()	Di chuyển focus ra khỏi thẻ input
Focus()	Di chuyển focus vào thẻ input
Select()	Selekt khung nhập liệu

\* Các thuộc tính

Thuộc tính	Môc đích
defaultValue	Xác định giá trị mặc định cho thẻ input
Name	Xác định tên của thẻ input
Value	Xác định giá trị hiện hành của thẻ input

\* Các sự kiện : Bao gồm onBlur, onChange, onFocus, và onSelect

### III. Xö lý c,c sù kiÖn tr'an trang Web

#### 2. C,c phUng thöc vµ thuéc tÝnh cña c,c ®èi tUng

##### 2.5 Sèi tñg text, textarea, password

VÝ dô: Tô ra mét form trong ®ã cã mét textbox đing ®Ó nhËp liÖu. Sau khi nhËp sè liÖu vµ dding phÝm Tab ®Ó di chuyÓn focus khái « nhËp liÖu thx tEt c¶ vñn b¶n trong « ®ã sї chuyÓn sang ch÷ hoa,vµ khi n'p trang web lñ thx focus (hay con trá) ®Uíc ®Æt t'í « nhËp liÖu

```
<html>
<head>
<script language="JavaScript">
    function ToUpcase()
    {
        st=document.form1.txtbox.value
        document.form1.txtbox.value=st.toUpperCase()
    }
</script>
</head>
```

### III. Xö lý c,c sù kiÖn trªn trang Web

#### 2. C,c phÙng thøc vµ thuéc tÝnh cñia c,c ®èi tÙng

##### 2.5 §èi tÙng text, textarea, password

VÝ dô: T¹o ra mét form trong ®ã cã mét textbox dïng ®Ó nhËp liÖu. Sau khi nhËp sè liÖu vµ dïng phÝm Tab ®Ó di chuyÓn focus khái « nhËp liÖu thx tÊt c¶ v n b¶n trong « ®ã sї chuyÓn sang ch÷ hoa,vµ khi n¹p trang web l n thx focus (hay con trá) ®Uc ®Æt t¹i « nhËp liÖu

```
<body onLoad="document.form1.txtbox.focus()">  
    <form name="form1">  
        <input type="text" name="txtbox" onBlur="ToUppercase()">  
    </form>  
</body>  
</html>
```

### III. Xử lý các sự kiện trên trang Web

#### 2. Các phương thức vật lý của các thẻ input

##### 2.6 Seite input select (gièng nhau listbox)

\* Các thuộc tính

Thuộc tính	Môc đích
Length	Xác định số phần tử trong danh sách
Name	Xác định tên của input
SelectedIndex	Trả về (hoặc thiết lập) chỉ mục của phần tử trong danh sách xác định (nghĩa là phần tử thứ mấy xác định, bắt đầu từ 0)

\* Các sự kiện : Bao gồm onBlur, onChange, và onFocus

### III. Xö lý c,c sù kiÖn trªn trang Web

#### 2. C,c phÙng thøc vµ thuéc tÝnh cña c,c ®èi tÙng

##### 2.6 Sèi tÙng select (giÙng nhÙ listbox)

VÝ dô: T¹o ra mÔt form chøa 1 select box (gåm 4 phÇn tö tõ 0 ®Õn 3), 1 textbox ®Ó nhËp liÖu, vµ 1 button. Khi ngÙêi sö dÔng nhËp mÔt sè vµo « textbox vµ kÝch nót button thx phÇn tö trong select box cä chØ mÔc b»ng sè ®Ùc nhËp sї ®Ùc chän

```
<html>
<head>
<script language="JavaScript">
    function SelectID()
    {
        s=document.form1.txtbox.value
        document.form1.lstbox.selectedIndex=s
        document.form1.txtbox.focus()
    }
</script>
</head>
```

### III. Xö lý c,c sù kiÖn trªn trang Web

#### 2. C,c phÙ-ng thøc vµ thuéc tÝnh cña c,c ®èi tÙng

##### 2.6 Sèi tÙng select (giÙng nhÙ listbox)

VÝ dô: T¹o ra mÔt form chøa 1 select box (gåm 4 phÇn tö tõ 0 ®Õn 3), 1 textbox ®Ó nhËp liÖu, vµ 1 button. Khi ngÙêi sö dÔng nhËp mÔt sè vµo « textbox vµ kÝch nót button thx phÇn tö trong select box cä chØ mÔc b»ng sè ®Ùc nhËp sї ®Ùc chän

```
<body onLoad="document.form1.txtbox.focus()">  
<form name="form1">  
    <select name="lstbox">  
        <option>Index 0</option>  
        <option>Index 1</option>  
        <option>Index 2</option>  
        <option>Index 3</option>  
    </select>  
    <input type="text" name="txtbox" size=5>  
    <input type="button" value="Click" onClick="SelectID()">  
</form>  
</body>  
</html>
```

### III. Xử lý các sự kiện trên trang Web

#### 2. Các phương thức xử lý thuộc tính của các thẻ input

##### 2.7 Xử lý button, submit, reset

\* Các thuộc tính của button

Thuộc tính	Mô tả
Name	Xác định tên của thẻ input
Value	Xác định giá trị của thẻ input

\* Các thuộc tính của button

### III. Xö lý c,c sù kiÖn trªn trang Web

#### 2. C,c phWng thøc vµ thuéc tÝnh cña c,c ®èi tUng

##### 2.8 §èi tÙng checkbox, radio

\* C,c thuéc tÝnh

Thuéc tÝnh	Môc ®Ých
Checked	X,c ®Þnh tr¹ng thi® Uic chän hay kh«ng (true hay false) cña ®èi tUng hay mét mÙng ®èi tUng
defaultChecked	ChØ ®Þnh gi, trÞ ngÇm ®Þnh cho ®èi tUng ®Uic chän
name	X,c ®Þnh tªn cña ®èi tUng
value	X,c ®Þnh gi, trÞ cña ®èi tUng

\* C,c ®iÒu khiÓn sù kiÖn : ChØ cã onClick

# CHƯƠNG 4 : TRIỂN KHAI WEBSITE

## I. CÁCH THỨC TẠO WEBSITE

### 1. Các lưu ý đối với các nhà thiết kế Web:

- + Những địa chỉ website nào tốt nhất bạn đã từng thiết kế?
- + Những site với chi phí đắt nhất và rẻ nhất mà bạn đã tạo ra là gì?
- + Bạn mất thời gian bao lâu để tạo ra một website?
- + Bạn thực hiện việc đồ họa trong các website như thế nào?
- + Bạn sẽ giúp chúng tôi quảng bá website của chúng tôi như thế nào?
- + Đăng ký với các nhà cung cấp dịch vụ tìm kiếm như thế nào?

# CHƯƠNG 4 : TRIỂN KHAI WEBSITE

## I. CÁCH THỨC TẠO WEBSITE

### 2. Cách thức tạo ra một Website:

+ Để tạo ra một Website có chất lượng đòi hỏi bạn phải có một kỹ năng thành thực và một sự nghiên cứu kỹ về Web. Bạn phải biết mình sẽ làm gì và không nên làm gì để đưa tất cả các ý tưởng đó vào việc xây dựng một Website.

+ Để tạo ra một Website bạn cần phải làm theo những bước sau đây:

- **Bước 1:** Đây là giai đoạn định hướng. Bạn cần phải đề cập đến những vấn đề sau đây:

# CHƯƠNG 4 : TRIỂN KHAI WEBSITE

## I. CÁCH THỨC TẠO WEBSITE

### 2. Cách thức tạo ra một Website:

- Những ý tưởng tổng quan của bạn.
- Mục đích của bạn cần đạt tới đối với website .
- Đối tượng mà bạn cần nhắm tới là ai.
- Bạn đã có những thông tin gì trong tay và bạn sử dụng chúng như thế nào.
- Bạn sẽ tổ chức thông tin như thế nào để bảo đảm sự truyền đạt thông tin là được sáng tỏ.

# CHƯƠNG 4 : TRIỂN KHAI WEBSITE

## I. CÁCH THỨC TẠO WEBSITE

### 2. Cách thức tạo ra một Website:

- Bước 2 :

- Tiến hành tổ chức các phần mục và các thông tin mà bạn có trên site của bạn.
- Tạo ra các nhánh, các tiêu đề và các tiêu đề phụ để bạn có thể tìm kiếm thông tin hữu ích một cách dễ dàng để không lãng phí thời gian đối với các thông tin mà bạn không quan tâm.
- Lựa chọn các từ khoá thích hợp để thuận tiện cho việc tìm kiếm và sử dụng các dịch vụ tìm kiếm.

# CHƯƠNG 4 : TRIỂN KHAI WEBSITE

## I. CÁCH THỨC TẠO WEBSITE

### 2. Cách thức tạo ra một Website:

- Bước 3 :

- Lựa chọn các hình ảnh đưa lên site của bạn từ thư viện điện tử hoặc từ trên đĩa CDROM. Đó có thể là những hình ảnh về sản phẩm, văn phòng làm việc của bạn, các chuyên gia chính của công ty.
- Chuyển đổi các hình ảnh đến vị trí thích hợp và có sự chỉnh sửa về màu sắc và kích cỡ cho phù hợp.

# **CHƯƠNG 4 : TRIỂN KHAI WEBSITE**

## **I. CÁCH THỨC TẠO WEBSITE**

### **2. Cách thức tạo ra một Website:**

**- Bước 4 :**

+ Khi bạn đã có bộ khung của mình thì bạn bắt đầu chuẩn bị tạo ra website bằng việc sử dụng ngôn ngữ siêu văn bản (HTML). Tiến hành chuyển đổi các văn bản text của mình tới HTML mà bạn có thể làm bằng Word, Netscape, Homesite và một vài các gói thông tin được lựa chọn khác.

+ Bạn có thể lựa chọn khi bạn thiết kế website của bạn hoặc là bạn tham gia vào các khoá đào tạo về thiết kế

+ Hoặc bạn có thể thuê các chuyên gia bên ngoài về thiết kế web để giúp cho bạn.

# CHƯƠNG 4 : TRIỂN KHAI WEBSITE

## I. CÁCH THỨC TẠO WEBSITE

### 2. Cách thức tạo ra một Website:

#### Bước 5:

- Lựa chọn nhà cung cấp dịch vụ Internet
- Thiết lập tên miền của bạn.
- Đăng ký tên website của bạn với các nhà tìm kiếm.
- Quảng cáo và khuếch trương website của bạn đối với các khách hàng mục tiêu bằng cách gửi thư, truyền thanh, truyền hình, biển hiệu quảng cáo....
- Một điều rất quan trọng là các thông tin của bạn phải được cập nhật hàng ngày.

# CHƯƠNG 4 : TRIỂN KHAI WEBSITE

## I. CÁCH THỨC TẠO WEBSITE

### 2. Cách thức tạo ra một Website:

Bước 6:

- Thông qua các công cụ tìm kiếm tiện ích đảm bảo rằng Website của bạn phải thật nổi bật :
  - + Lycos, Alta Vista, Google...
- Đây là việc tốn rất nhiều thời gian.

# CHƯƠNG 4 : TRIỂN KHAI WEBSITE

## II. Môi trường cạnh tranh với website:

Khi một Website Thương mại điện tử ra đời, vấn đề cần thiết là phải cạnh tranh được với một số Website lớn đang chiếm giữ thị trường. Để trên khía cạnh quản lý một trang web, phải tìm cách để Website hướng sự chú ý của khách hàng đến Công ty.

### Yêu cầu cần đặt ra khi làm website và ứng dụng thực tế :

- 1 . Giao diện thân thiện, dễ sử dụng
- 2 . Hỗ trợ đa ngôn ngữ(khuyến cáo)
- 3 . Trợ giúp ngắn gọn, dễ hiểu
- 4 . Thông tin chính xác
- 5 . Công cụ tìm kiếm nhanh, hiệu quả và dễ sử dụng.
- 6 . Chức năng đầy đủ
- 7 . Hỗ trợ các mục tin tức cần thiết
- 8 . Có những phần download miễn phí để gây ấn tượng cho khách hàng
- 9 . Domain ngắn gọn, dễ nhớ
10. Website truy cập nhanh

...

# Active Server Pages

## I. Tæng quan VÒ ASP

### 1. Giïi thiÖu

ASP lµ c«ng cô hç trî cho viÖc t¹o ra c,c trang web hµng m¹nh (c,c trang nµy thUêng ®íc gäi lµ c,c trang ASP, vµ cã phÇn më réng lµ \*.asp).

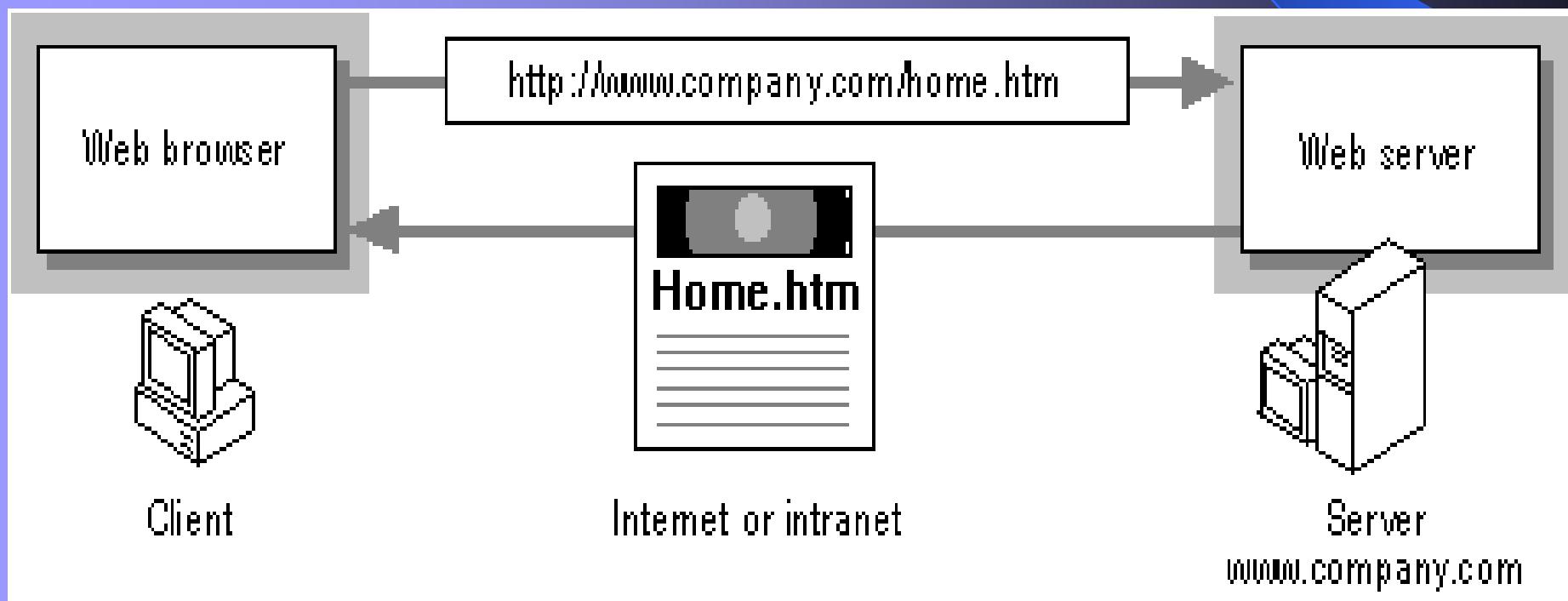
Trang ASP cã thÓ xem lµ mét trang HTML më réng, v× ngoµi tÝnh n»ng ®Çy ®ñ cña mét tµi liÖu HTML, trong nã cßn cã thÓ tÝch hîp c,c c©u lÖnh cña VBScript (hoÆc JavaScript) ®Ó truy cËp tµi nguyªn hÖ thèng, xö lý tÝnh to,n sè liÖu, sö dông c,c c©u lÖnh cña ng«n ng÷ SQL kÕt hîp víi c,c ®èi tUring ®íc cung cÊp s½n trong ASP ®Ó truy cËp c¬ sè d÷ liÖu nhU Access, SQL Server, Oracle,...

# Active Server Pages

I. Tæng quan vÒ ASP

2. ASP lµm viÖc nhU thÓ nµo ?

Trang Web tÙnh :

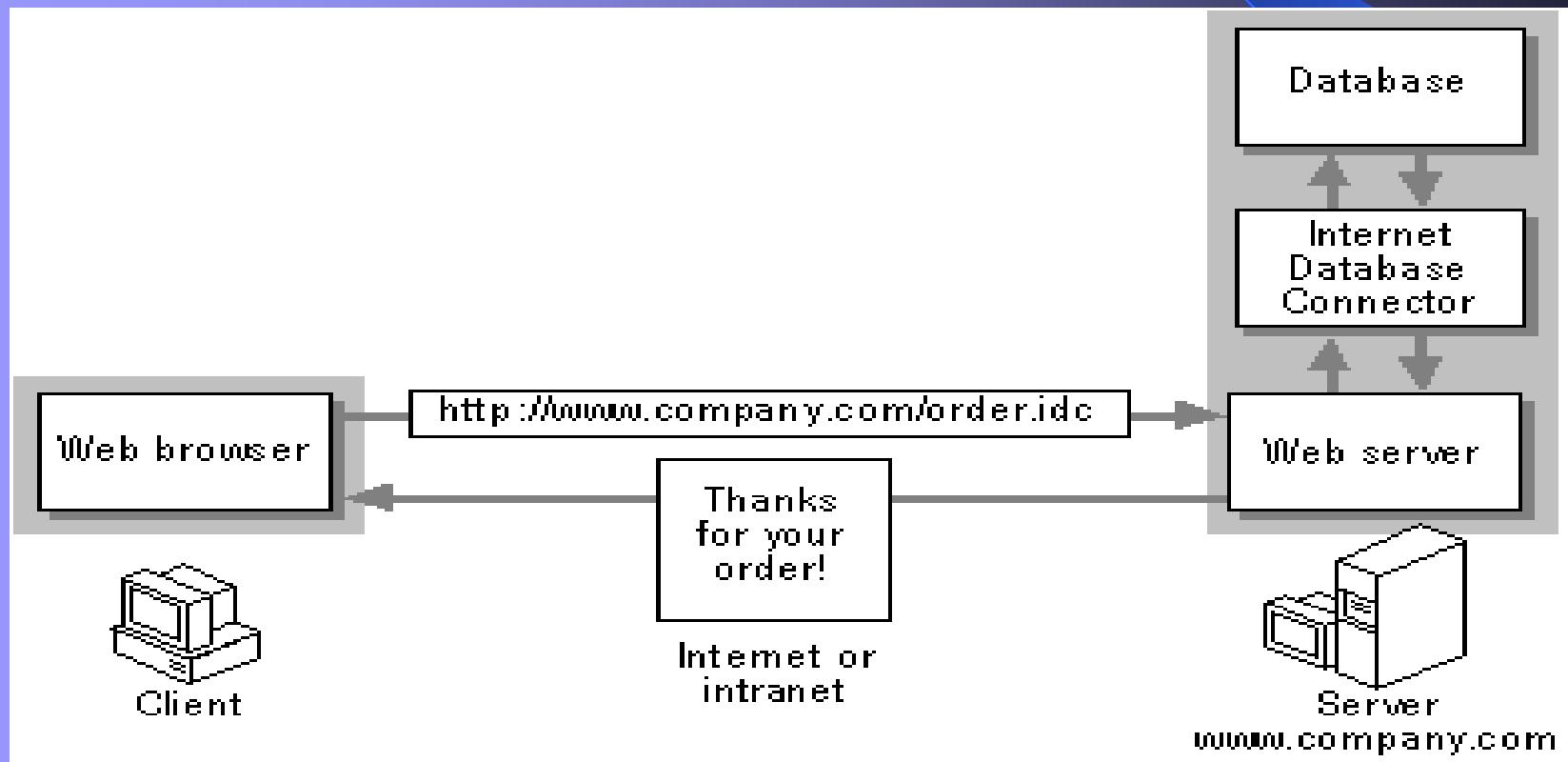


# Active Server Pages

## I. Tæng quan VÒ ASP

### 2. ASP lµm viÖc nhU thÔ nµo ?

Trang web đÔng (Dynamic):

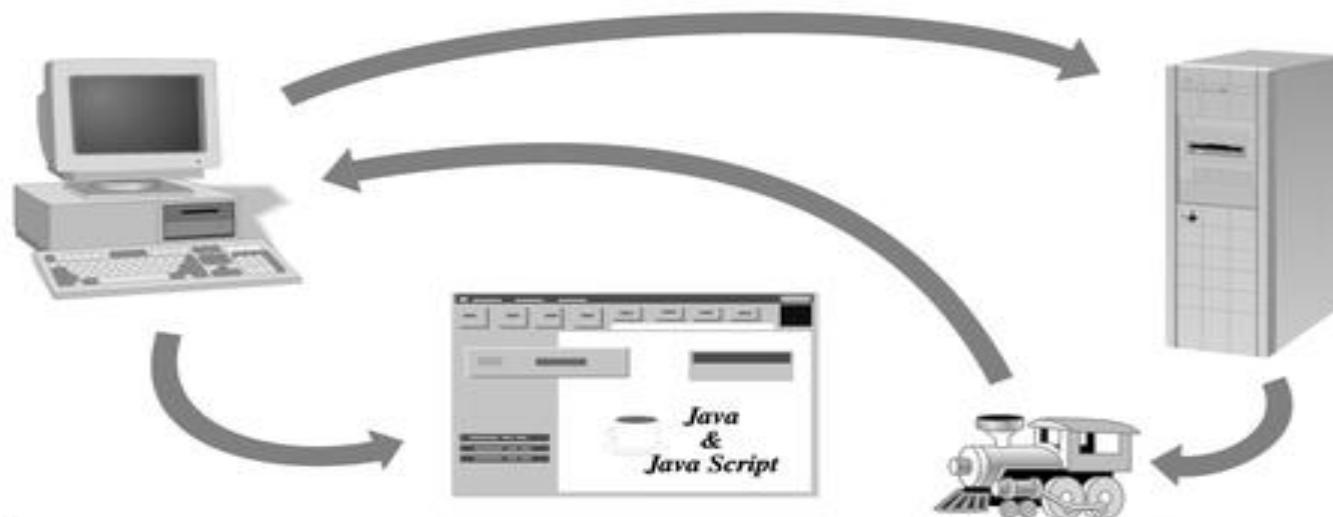


# Active Server Pages

I. Tæng quan VÒ ASP

2. ASP lµm viÖc nh thÓ nµo ?

## SERVER-SIDE SCRIPTING MODEL



The browser requests a page.

The server sends the script to the engine.

The script engine executes the script.

The server sends the page to the browser.

The browser renders the page.

# Active Server Pages

## I. Tæng quan VÒ ASP

### 3. C, ch x@y dÙng mét trang ASP

C,c file ASP cã phÇn më réng lµ .ASP

Khi chÌn th m c,c lÖnh nµy b n cÇn ch  ý:

- + C,c lÖnh chÌn th m vµo ph¶i n»m gi÷a kho, <% ... %>
- + C,c lÖnh th m vµo cã thÓ ®Æt bÊt kú t i v  trÝ nµo trong file HTML
- + C,c c@u ghi ch  trong ASP (kh ng kÓ c,c ghi ch  c a HTML) ®Øu ®-  
U c b t ®Çu b»ng d u nh,y ('')
- + C,c lÖnh kh ng ph n bi t ch  hoa, ch  th ng.

# Active Server Pages

## I. Tæng quan vÒ ASP

### 3. C,ch x©y dÙng mét trang ASP

VÝ dÔ : Mét file asp ®¬n gi¶n lµ céng hai sè vµ tr¶ kÔt qu¶ vÒ cho tr×nh duyÖt.

<%

'b¾t ®Çu cho phĐp viÖt c,c lÖnh ASP

Dim a,b,c

'khai b,o 3 biÖn a,b,c

a=10

'g,n 10 cho biÖn a

b=20

'g,n 20 cho biÖn b

c=a+b

'tÝnh tæng c=a+b

'kÔt thóc viÖt lÖnh cho ASP

%>

# Active Server Pages

## I. Tæng quan vÒ ASP

### 3. C, ch x©y dÙng mét trang ASP

<!--b¾t ®Çu phÇn tµi liÖu HTML-->

- <HTML>
- <HEAD>
- <Title>VÝ dô vÒ ASP</Title>
- <HEAD>
- <BODY>
- <!-- in kÕt qu¶ ra trang web-->
- **Tæng cña a vµ b lµ <%=c%>**
- </BODY>
- </HTML>

# Active Server Pages

## I. Tæng quan vØ ASP

### 3. C, ch x©y dÙng mét trang ASP

- Víi vÝ dô ta chó ý ®Ön lÖnh `<%=c%>`, lÖnh nµy cho phĐp viÖt kÖt qu¶ sau khi tÝnh to,n ra trang web
- NhÚP vËy ®Ó viÖt mét kÖt qu¶ ra trang web b¹n thÙc hiÖn theo cÙ ph,p:

`<%=bieu_thuc%>`

- Trong ®ã *bieu\_thuc* cÙ thÓ lµ mét biÖn, mét biÓu thøc, hay mét hµm cÇn in kÖt qu¶.

# Active Server Pages

## II. LÊP TRÌNH VỚI ASP

### 1. Khai báo biến

- + Khi khai báo biến bằng cách khai báo `OPTION EXPLICIT` kèm so với biến lối.
- + `NĂN SỐ ĐÔNG <% Option Explicit %>` để ngay lập tức mực lập ASP.
- + Có phím: `DIM biến 1, biến 2...`
- + Số khai báo mang:

`DIM a(10) : chung se chay to 1->10`

- + Khi khai báo năn số đồng cùi thì:

# Active Server Pages

## II. LÊP TRÌNH VỚI ASP

### 1. Khai báo biến.

Dạng dữ liệu	Tiền tố
Boolean	Bln
Byte	Byt
Double	Dbl
Integer	Int
Long	Lng
Object	Obj
String	Str
ADO command	Cmd
ADO connection	Cnn
.....	

# Active Server Pages

## II. LÊP TRÌNH VỚI ASP

### 2. Khai báo hằng số

CONST tên\_hằng\_số = giá\_trị

### 3. Các câu lệnh điều kiện

#### a. Câu lệnh IF..THEN..ELSE..IF

Có phím:

- <% If <điều\_kiện\_1> then
- <những\_câu\_lệnh>
- Else
- If <điều\_kiện\_2> then
- <những\_câu\_lệnh>
- End if
- End if
- %>

# Active Server Pages

## II. LỆP TRÌNH VỚI ASP

### 3. CÁC CÔNG THỨC TRÓC ®IÒU KHIÖN

Ngoài ra cần thõ sö dông : IF.. THEN....ELSEIF...END IF

- <%
- If <§iÒu kiÖn 1> then
  - <CáC cÔu lÖnh 1>
- ElseIf <§iÒu kiÖn 2> then
  - <CáC cÔu lÖnh 2>
- Else
  - <CáC cÔu lÖnh 3>
- End if
- %>

# Active Server Pages

## II. LÃp trÃnh vÃi ASP

### 3. C,c cÊu tróc ®iÙu khiÓn

#### b. CÊu tróc `Select case...end select`

Có ph,p:

`<% Select Case <tenbien>`

`Case <gia tri 1>`

`<NhÃm lÖnh 1>`

`Case <gia tri 2>`

`<NhÃm lÖnh 2>`

`Case <gia tri n>`

`<NhÃm lÖnh n>`

`Case Else`

`<NhÃm lÖnh n+1>`

`End Select`

`%>`

# Active Server Pages

## II. LÃp trænh vÃi ASP

### 3. C, c cÊu tróc ®iÒu khiÓn

#### c. CÊu tróc lÆp tuÇn tÙ For...Next.

Dïng ®Ó lÆp vÃi sè lÇn ®· biÖt, tuy nhia³n ta cã thÓ ng³t b»ng lÖnh EXIT FOR.

Có ph,p:

<%

FOR <tenbien>=<giatri dau> TO <Giatri bien> STEP <Buoc nhay>

NEXT

%>

# Active Server Pages

## II. LÃp trÃnh vÃi ASP

### 3. C,c cÊu tróc ®iÒu khiÓn

#### d. CÊu tróc lÆp kh,c

\* DO WHILE....LOOP :

Có ph,p:

- <%

- Do while <§iÒukiÖn>

- <C,c c©u lÖnh>

- Loop

- %>

# Active Server Pages

## II. LÃp trÃnh vÃi ASP

### 3. C,c cÊu tróc ®iÒu khiÓn

#### d. CÊu tróc lÆp kh,c

\* WHILE....wend

- Có ph,p:

- <%

- While <§iÒukiÖn>

- <C,c c©u lÖnh>

- Wend

- %>

# Active Server Pages

## II. LÃp trÃnh vÃi ASP

### 3. C,c cÊu tróc ®iÒu khiÓn

#### d. CÊu tróc lÆp kh,c

\* DO....Loop until

Có ph,p:

<%

DO

<C,c c@u lÖnh>

Loop until <§iÒu kiÖn>

%>

# Active Server Pages

## II. LÊP TRÌNH VỚI ASP

### 4. Một số hàm hữu dụng trong ASP

#### a. Các hàm xử lý văn bản

- **TRIM(x@u as string)**: Bả khoảng trống hai @&u kÝ tù
- **LEFT(X@u as string, n as interger)**: Lêy b@n tr,i x@u n kÝ tù.
- **RIGHT(X@u as string, n as interger)**: Lêy b@n ph@i x@u n kÝ tù.
- **LCASE(X@u as string)** : Chuyển x@u vÒ ch÷ thêng
- **UCASE(X@u as string)** : Chuyển x@u vÒ ch÷ hoa
- **MID(x@u as string, n1, n2)**: Lêy n2 kÝ tù trong x@u b@t @&u tõ v@ trÝ n1.
- **CSTR(BiÕn)**: H@m chuyển @&ei biÕn thành kiÕu string

# Active Server Pages

## II. LÃp trænh viïi ASP

### 4. M t s  h m h u d ng trong ASP

#### a. C c h m x  l y v n b n

H m JOIN/SPLIT(X u as string, k y t  ng n c, ch): S i N i/C t x u th nh m t/nhi u  o n b ng c, ch x, c  pnh k y t  ng n c, ch   tr n v u cho c, c  o n  a l t v o m t m ng.

V y du:

<%

x="H u n i;H i ph ng;TPHCM"

y=split(x,";")

Response.write y(0)

'y(0)="H u n i"

%>

# Active Server Pages

## II. LỆP TRÌNH VỚI ASP

### 4. Một số hàm hữu dụng trong ASP

#### b. Các hàm xử lý số

- **SQR(n)**: Căn bậc hai của n
- **INT(n)**: Lấy phần nguyên n
- **MOD**: Phép chia dư
- **To,n tö \**: Chia nguyên
- **Round(số, n)**: Lấy tròn số với n chữ số sau dấu phẩy
- **RND()**: Trả về số ngẫu nhiên từ 0 đến 1

#### c. Các hàm về ngày tháng, thời gian.

# Active Server Pages

## III. Các thành phần trong ASP

Trong ASP ® cung cấp s½n 5 thành phần (Component) cho phép tạo các trang web ®éng mét cách nhanh chóng.

- + **Database Access:** ®áp dụng thành phần cho phép truy cập các loại cơ sở dữ liệu.
- + **Ad Rotator:** cho phép hiển thị các quảng cáo mới từ ®éng theo thời gian quy ®®nh.
- + **Browser Capabilities:** xác ®®nh các khả năng của trình duyệt, ®éc ®iÓm về phiên bản (version) của các trình duyệt duy ®t khi truy cập Website.
- + **Content Linking:** tạo ra các liên kết chung các nội dung (gièng nhau hoặc lồng) của các trang web.
- + **File Access:** cung cấp các chức năng truy cập tui hò thèng file và thay đổi các trang.

# Active Server Pages

## III. C,c thµnh phÇn trong ASP

### 1. Qu, trxnh truy cËp C¬ së d÷ liÖu

DataBase Access sö dông mét ®èi tîng gäi lµ ADO (ActiveX Data Object) vµ kÕt hîp kh¶ n¤ng cña ODBC (®UÍc gi¶i thÝch è phÇn sau trong chÙ¬ng nµy) ®Ó lÊy th¤ng tin trong mét c¬ së d÷ liÖu. ViÖc truy cËp c¬ së d÷ liÖu cã thÓ tiÖn hµnh qua c,c bUÍc sau:

- + §¤ng ký mét tªn ®¹i diÖn cho nguân d÷ liÖu cÇn truy cËp vñi ODBC (sÏ gi¶i thÝch è phÇn sau)
- + T¹o mét nèi kÕt tñi nguân d÷ liÖu cÇn truy cËp
- + Më C¬ së d÷ liÖu ®Ó truy cËp
- + Sö dông ng¤n ng÷ SQL ®Ó truy vÊn d÷ liÖu
- + Xö lý d÷ liÖu
- + §ãng nèi kÕt.

# Active Server Pages

## III. Các thành phần trong ASP

### 1. Quá trình truy cập CSDL để lấy dữ liệu

- Một loạt các bước tiến trình truy cập cần thiết mìn tàng qua trình sau:

Ví dụ như số dòng file Vidu.asp là truy cập CSDL để lấy dữ liệu Access (với file cần tên VIDU.MDB)

Giả sử rỗng trong file này có 1 bảng (Table) với tên là KHACHHANG gồm các trường *STT, Hoten, Diachi* (các trường này trong Access đặc biệt là số từ 0 đến 2).

Sau khi đăng ký file VIDU.MDB trong ODBC với tên VIDU (tên này là bất kỳ, không cần trùng với tên file .MDB, và việc đăng ký cho tên hiển thị một lần, các lần sau đó việc truy cập dữ liệu sẽ không cần đăng ký nữa)

Trong file Vidu.asp ta viết các lệnh sau:

# Active Server Pages

## III. Các thành phần trong ASP

### 1. Quá trình truy cập cơ sở dữ liệu

- <%

'Tạo ra một biến tên conn để kết nối với nguồn dữ liệu

- set conn=Server.CreateObject("ADODB.Connection")

'Sử dụng phương thức OPEN của biến conn để kết nối đến CSDL cần truy cập

- conn.open "VIDU"

'Biến sql chứa câu lệnh SQL để lấy dữ liệu từ bảng KHACHHANG

- sql="SELECT \* FROM KHACHHANG"

'Sử dụng phương thức Execute của biến conn để thực hiện câu lệnh SQL

'Kết quả sau khi truy vấn được hiển thị trong biến RS

- Set RS = Conn.Execute(sql)
- %>

# Active Server Pages

- <HTML>
- <Body>
- <%

'Sö dōng vßng lÆp ®Ó kiÓm tra RS cã cßn chøa d÷ liÖu hay kh«ng

- Do While NOT RS.EOF
- %>

STT: <%=RS(0)%>, Hä tªn: <%=RS(1)%>, §þa chØ: <%=RS(2)%>

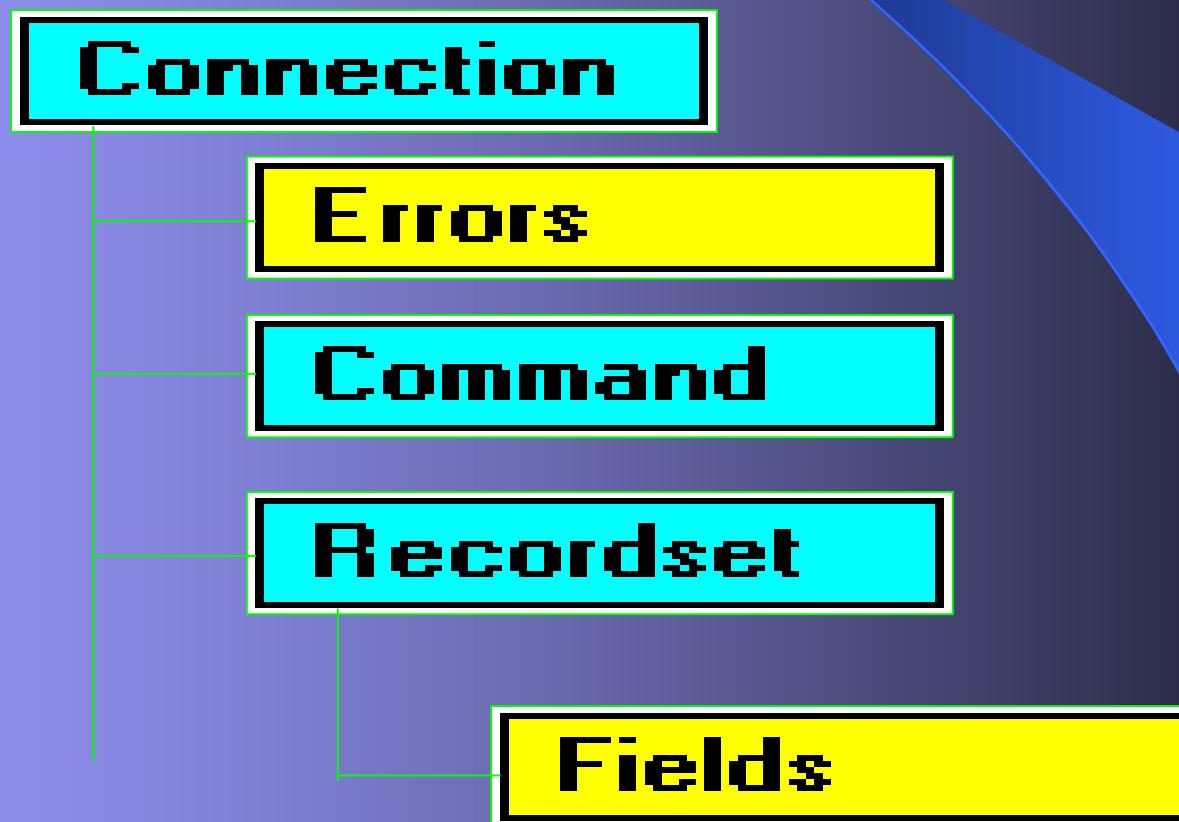
- <% RS.MoveNext  
Loop  
Conn CLOSE  
%>  
</BODY>  
</HTML>
- 'di chuyÓn ®Õn Record tiÕp theo  
'quay l¹i vßng lÆp  
'®âng nèi kÕt

# Active Server Pages

## III. Các thành phần trong ASP

### 2. Các khái niệm xoay quanh ADO

§èi tûng ADO cho phép tạo ra các khái niệm con khác:



# Active Server Pages

## III. Các thành phần trong ASP

### 2. Các Rõi tông xô lý d÷ liÖu

- + **Đèi tông Connection** : m« t¶ sù nèi kÕt túi nguân d÷ liÖu. Đèi tÙng nµy cho phĐp t¹o ra 3 Rõi tÙng con lµ Errors, Command, vµ RecordSet
- + **Đèi tông Errors** : chøa chi tiÕt c,c lçi vÒ truy cËp d÷ liÖu
- + **Đèi tông Command** : ®Þnh nghÜa c,c lÖnh ®Ó truy cËp d÷ liÖu
- + **Đèi tông RecordSet** : chøa tÊt c¶ c,c record d÷ liÖu ®Uic lÊy vÒ tõ c,c b¶ng (table) trong c¬ së d÷ liÖu, Rõi tÙng nµy cho phĐp t¹o ra mét Collection lµ Fields
- + **Collection Fields** : dñng ®Ó chøa tÊt c¶ c,c trÙêng n»m trong mét Recordset

# Active Server Pages

## III. Các tham số trong ASP

### 2. Các lỗi tông xô lý do lỗi

#### 2.1. Các phương thức, thuộc tính của lỗi tông Connection

##### Các Thuộc tính:

- ConnectionTimeout: Xác định khoảng thời gian chờ kết nối từ nguồn dữ liệu
- CommandTimeout: Xác định khoảng thời gian chờ câu lệnh truy vấn từ dữ liệu

# Active Server Pages

## III. C,c thµnh phÇn trong ASP

### 2.1. C,c phÙ-ng thøc, thuéc tÝnh cña ®èi tÙng Connection

#### C,c phÙ-ng thøc:

**Connection:** më mét nèi kÕt tñi nguân d÷ liÖu

Có ph,p: *connection.OpenConnectionString*

Trong ®ã *ConnectionString* lµ mét chuçi cung cÊp th«ng tin cho viÖc nèi kÕt

**Execute:** thùc hiÖn mét cÇu lÖnh ®îc ®Æc t¶

Có ph,p: *Set recordset= connection.Execute(Command)*

- Trong ®ã: *Command* lµ cÇu lÖnh cÇn thùc hiÖn

- Recordset:* lµ ®èi tÙng Recordset

- Connection:* lµ ®èi tÙng Connection

**Close:** ®ãng mét ®èi tÙng ®ang më vµ c,c ®èi tÙng kh,c liªn quan

Có ph,p: *object.Close*

# Active Server Pages

## III. Các thành phần trong ASP

### 2. Các khái niệm xoay quanh Recordset

#### 2.2. Các phím tắt thường dùng, thuộc tính của Recordset

- Các Thuộc tính:

+ **BOF** : thuộc tính này trả về giá trị TRUE nếu vị trí trỏ của Recordset hiện đang trỏ đến record đầu tiên trong Recordset, ngược lại trả về giá trị FALSE

+ **EOF** : trả về giá trị True nếu vị trí trỏ của Recordset hiện đang ở cuối cùng, ngược lại trả về giá trị False

# Active Server Pages

## III. Các thành phần trong ASP

### 2.2. Các phương thức, thuộc tính của Recordset

#### Các phương thức:

- **Close:** Rỗng Rồi đóng Rỗng mõi vñ c,c Rồi đóng kh,c lñan quan  
Có ph,p: *object.Close*
- **MoveFirst:** Di chuyển tñi record RÇu tiän trong 1 Recordset  
Có ph,p: *recordset.MoveFirst*
- **MoveLast:** Di chuyển tñi record cuèi cÙng trong 1 Recordset  
Có ph,p: *recordset.MoveLast*
- **MoveNext:** Di chuyển tñi record tiếp theo trong 1 Recordset  
Có ph,p: *recordset.MoveNext*
- **MovePrevious:** Di chuyển tñi record trước Record hiện hành trong 1 Recordset  
Có ph,p: *recordset.MovePrevious*

# Active Server Pages

## III. Các thành phần trong ASP

### 3. Các khái niệm trong ASP

ASP cung cấp sáu khái niệm cho phép bạn xử lý thông tin trao đổi giữa Client và Server.

Đối tượng	Chức năng
Request	Lấy thông tin từ Client gửi lên Server
Response	Gửi thông tin từ Server về Client
Server	Cung cấp các chức năng truy cập và xử lý thông tin thuộc về các ứng dụng trên Server
Session	Lưu trữ thông tin về từng phiên làm việc của một Client
Application	Chia sẻ thông tin chung между các ứng dụng cho tất cả các Client

# Active Server Pages

## III. Các thành phần trong ASP

### 3. Các khái niệm trong ASP

#### 3.1. Sesi và Request

Có phím chung: `Request[.Collection](variable)`

Trong đó `Collection` là một tập hợp các biến sau:

- Form: dùng để lấy giá trị của các trường trong Form HTML, khi Form có số đông `method="post"`

Có phím dùng cho Form:

`Request.Form(parameter)[(index)]`

+ `parameter`: tên của một trường cần nhận giá trị, truy

+ `Index` là giá trị của trường cần truy cập lần thứ nhất, truy khéo nhau của `parameter`.

# Active Server Pages

## III. Các thành phần trong ASP

### 3. Các công nghệ khai thác trong ASP

#### 3.1. Xét URL Request

- **QueryString:** dùng để lấy tất cả các biến mà sau dấu hỏi (?) trong mét URL (gọi là Query String), hoặc khi mét form dùng Method="GET"

`Request.QueryString(variable)[(index)]`

- + *variable* là tên của biến trong chuỗi query string có sẵn
- + *Index* là tham số thứ nhất cần cho phép nhận một hoặc nhiều biến

# Active Server Pages

## III. Các thành phần trong ASP

### 3. Các khái niệm trong ASP

#### 3.1. Khái niệm Request

Ví dụ: Giả sử có một URL như sau:

- /Scripts/Sample.asp?hoten=NguyenvanA&tuoi=20

Tết các nhau gác nhau sau dấu hỏi (?) là Query String. Trong Query String chia ra làm hai phần phân nhau bởi dấu &. Trong mỗi phần có thể bao gồm hai thành phần phân nhau bởi dấu bằng (=), phần trước dấu bằng (=) là tên biến, phần sau dấu bằng là giá trị của biến.

Như vậy trong file Sample.asp mùn sẽ hiển thị hoten và tuoi ở dòng lệnh sau:

```
<%  
a=Request.QueryString("hoten")'biến a=NguyenvanA  
b=Request.QueryString("tuoi")'biến b=20  
%>
```

# Active Server Pages

## III. Các thành phần trong ASP

### 3. Các thành phần khai thác trong ASP

#### 3.1. Xét trường Request

- ServerVariables: Là gì, trả về những gì khi mua truy cập (được ASP cung cấp sẵn)

Có phím: Request.ServerVariables (*variable*)

Trong đó *variable* là tên của biến mua truy cập cung cấp.

Variable có các giá trị sau:

+ REQUEST\_METHOD: Là gì, trả về những gì mua Client yêu cầu Server.

Ví dụ:

```
<FORM ACTION = "/scripts/submit.asp" METHOD = "post">
```

```
Request.ServerVariables("REQUEST_METHOD")="post".
```

# Active Server Pages

## III. Các thành phần trong ASP

### 3. Các thành phần khai thác trong ASP

#### 3.1. Xét Request

- + **PATH\_INFO:** Xác định đường dẫn (virtual path) của file đang thi hành
- + **SCRIPT\_NAME:** Xác định tên của script đang thi hành.
- + **LOGON\_USER:** Lấy Username của người logon vào máy
- + **REMOTE\_ADDR:** Lấy địa chỉ IP của máy truy cập có thông tin
- + **SERVER\_NAME:** Lấy tên hay địa chỉ IP của Server
- + **REMOTE\_HOST:** Lấy địa chỉ IP của Server.

# Active Server Pages

## III. Các thành phần trong ASP

### 3. Các Rèi tÙng khc trong ASP

#### 3.2. Sèi tÙng RESPONSE

Sèi tÙng nay dñg ®Ó gõi d÷ liÖu vÒ Client

Có ph.p:

`Response.property|method`

Trong ®ã:

*properties* gm c,c gi, trP sau:

- + **Buffer:** X,c ®Þnh cn nn s dng vng ®Öm cho c,c trang thng tin hay khng. Khi nn cn gi, trP True thx Server s khng tr¶ li bt k mt yu cu no ®Ôn client cho ®Ôn khi trang hiÖn ti ®· x ly xong, hoÆc cho ®Ôn khi gi phÙ¬ng thc Response.Flush hay Response.End. Nn ®Æt Response.Buffer = dng ®Çu tin trong file (.ASP).
- Có ph.p: Response.Buffer=True/False
- + **Expires:** X,c ®Þnh thi gian mu trang thng tin nm trong vng ®Öm ht t,c dng
- Có ph.p: Response.Expires [= *number*]
- Trong ®ã *number* lu s pht mu trang ht thi hn

# Active Server Pages

## III. Các thành phần trong ASP

### 3. Các lệnh xử lý khai thác trong ASP

#### 3.2. Lệnh xử lý RESPONSE

Method gồm các gì, truy sau:

+ Write: dùng để in mét biến thõa ra màn hình.

Có phím: Response.Write Biểu\_thức

+ Redirect : dùng để nhảy sang một URL khác.

Có phím: Response.Redirect URL

Ví URL lùi một bước về mực ban đầu của chương trình.

+ Clear: Xoá vĩnh viễn các biến HTML.

Phương thức này sẽ ghi lại nội dung trả về bằng cách Response.Buffer = True.

Có phím: Response.Clear

+ End: đóng và xóa file .asp và trả về kết quả hiển thị

Có phím: Response.End

+ Flush: gọi ngay lập tức để lưu trữ trong bộ nhớ ra màn hình.

Phương thức này sẽ ghi lại nội dung trả về bằng cách Response.Buffer = TRUE.

Có phím: Response.Flush

# Active Server Pages

## III. Các thành phần trong ASP

### 3. Các công nghệ khai thác trong ASP

#### 3.3. Session SESSION

- Session là trung tâm của ASP, nó lưu trữ thông tin cho mỗi phiên lướt web (còn gọi là User), khi User này kết thúc sẽ bị WebSite xóa bỏ.
- Những thông tin đặc biệt lưu trữ trong Session sẽ không bị mất khi User này nhảy từ trang này qua trang khác trên Web Site còn lại.
- Server sẽ tự động tạo ra Session khi một trang web tải lên đồng thời xác định yêu cầu của người dùng số lượng, Server sẽ phán đoán Session khi kết thúc phiên lướt web, hoặc cả yêu cầu phán đoán.

Có phím: `Session.property|method`

# Active Server Pages

## III. Các thành phần trong ASP

### 3. Các khái niệm khái niệm trong ASP

#### 3.3. Session

\* *Properties* của các gi, trp sau:

- + **SessionID**: Trl vO ®pnh danh cña Sesion cho tõng User, ®pnh danh nuy lµ duy nhEt cho mci User (gièng nh m· sè ®Ó qu¶n lý User)

Có ph,p: **Session.SessionID**

- + **TimeOut**: ThiÓt lËp thêi gian hÖt h¹n cho ®èi tîng Session trong tõng øng dÔng, ®Uïc tÝnh b»ng phót
- NÕu trong kho¶n thêi gian nuy mµ User kh«ng Refresh hoÆc yªu cÇu th«ng tin thx Session kÖt thóc.

Có ph,p: **Session.Timeout [= nMinutes]**

- Víi *nMinutes* lµ sè phót cÇn thiÓt lËp (ngÇm ®pnh lµ 20 phót)

# Active Server Pages

## III. Các thành phần trong ASP

### 3. Các thành phần khai thác trong ASP

#### 3.3. Session

\*Method cần có gì, trả sau:

- + **Abandon**: dừng hoạt động, huỷ bỏ Session vĩnh viễn phẳng khái bé nhí.

Có phép: **Session.Abandon**

Trong khai thác SESSION có bốn cách hai sự kiện lặp **Session\_OnEnd** và **Session\_OnStart**.

Sự kiện **Session\_onStart** thường xảy ra khi Server tạo ra Session mới, sự kiện **Session\_onEnd** thường xảy ra Session bị dừng, huỷ (bị ngắt phanh thông qua **Abandon** hay thuộc tính **Timeout**).

# Active Server Pages

## III. C, c thµnh phÇn trong ASP

### 3. C, c ®èi tÙng kh,c trong ASP

#### 3.3. §èi tÙng SESSION

- Có ph,p cña hai sù kiÖn nµy ®îc viÖt nhU' sau:
- <SCRIPT LANGUAGE=ScriptLanguage RUNAT=Server>
- Sub Session\_OnStart
- ... c,c lÖnh cho khëi t'o phi'an lµm viÖc ®Æt ë ®©y
- End Sub
- </SCRIPT>
- <SCRIPT LANGUAGE=ScriptLanguage RUNAT=Server>
- Sub Session\_OnEnd
- ... c,c lÖnh cho kÖt thóc phi'an lµm viÖc ë ®©y
- End Sub
- </SCRIPT>
- Hai sù kiÖn nµy ®îc so¹n trong mét file cã tªn lµ Global.asa, vµ file nµy ph¶i ®Ùc ®Æt ë thU' mÙc gÙc cña øng dÙng ®ang thi hµnh.

# Active Server Pages

## III. C,c thµnh phÇn trong ASP

### 3. C,c ®èi tÙng kh,c trong ASP

#### 3.4. §èi tÙng Application

DÙng ®Ó chia sї th«ng tin gi÷a c,c user ®ang dÙng chung mét øng dÙng nµo ®ã.

Do mét biÕn Application cã thÓ dÙng chung cho tÊt c¶ c,c user, n n t i mét th i ®iÓm n u b n mu n thay ®æi gi, tr c a biÕn Application th  s  dÙng ph¬ng th c

- + Lock ®Ó kho, sù thay ®æi ¢c,c User kh,c
- + Sau ®ã s  dÙng ph¬ng th c Unlock ®Ó tr¶ l i tr ng th,i ban ®Çu c a n .

# Active Server Pages

## III. Các thành phần trong ASP

### 3. Các công nghệ khai thác trong ASP

#### 3.4. Seite Using Application

Có phím:

- + **Application(var)**: dùng để chia sẻ giá trị trên toàn Application. Ví dụ var lẻ tản mạn
- + **Application.Lock**: ngăn chặn khung cho client sử dụng cùng lúc
- + **Application.Unlock**: cho phép sử dụng lại

Ví dụ: Đếm số lần truy cập 25 cho biến num sau

Application("num") = 25

Biến num sẽ trả về truy cập bởi tất cả các Client.

Hỗn hợp với việc hiển thị số lần truy cập num ra màn hình có phím :

<% Response.write Application("num") %>

# Active Server Pages

## III. C,c thμnh phÇn trong ASP

### 3. C,c ®èi tÙng kh,c trong ASP

#### 3.4. §èi tÙng Application

- + Còng gièng nhƯ ®èi tÙng Session, ®èi tÙng Application cã hai sù kiÖn lµ Application\_OnStart, Application\_OnEnd, sù kiÖn thø nhÊt ®Uíc ®Ua ra khi b¾t ®Çu thÙc thi øng dÔng vµ sù kiÖn thø hai cho kÕt thÓc mét øng dÔng.
- + C¶ hai sù kiÖn nµy ®Uíc so¹n th¶o trong file Global.asa, vµ file nµy ph¶i ®Uíc ®Æt ë thƯ mÔc gÙc mµ øng dÔng hiÖn t¹i ®ang ho¹t ®éng. Trong file Global.asa sù kiÖn Application\_OnStart ®Uíc Uu tiªn t×m thÊy vµ sö dÔng tr-  
Uíc sù kiÖn Session\_OnStart. Sù kiÖn Application\_OnEnd cã Uu tiªn sau Session\_OnEnd.

# Active Server Pages

## III. Các thành phần trong ASP

### 3. Các thành phần trong ASP

#### 3.4. Sự kiện Application

Có phép có hai sự kiện nhảy sau:

```
<SCRIPT LANGUAGE=ScriptLanguage RUNAT=Server>
```

```
    Sub Application_OnStart
```

```
        ... các lệnh khi bắt đầu thực hiện ứng dụng
```

```
    End Sub
```

```
</SCRIPT>
```

```
<SCRIPT LANGUAGE=ScriptLanguage RUNAT=Server>
```

```
    Sub Application_OnEnd
```

```
        ... các lệnh khi kết thúc ứng dụng
```

```
    End Sub
```

```
</SCRIPT>
```

# Active Server Pages

## IV. §”ng ký nguân d÷ liÖu víi ODBC

### 1. Giíi thiÖu

NhƯ c,c b1n ®· biÖt lµ mçi lo¹i c¬ së d÷ liÖu ®Òu cung cÊp mÔt phU¬ng ph,p riªng ®Ó truy cÊp th«ng tin trong c¬ së d÷ liÖu ®ã, ®iÒu nµy ®· g®y kh«ng Ýt khä kh^n cho viÖc truy cÊp th«ng tin c¬ së d÷ liÖu trªn m¹ng Internet.

Do ®ã trong nhiÒu n”m, c,c tËp ®oµn trªn thÕ giíi ®· ®Çu tU' rÊt nhiÒu tiÒn cña vµo viÖc x®y dung mét phU¬ng ph,p lËp tr×nh nh»m cung cÊp mét giao diÖn lËp tr×nh øng dông duy nhÊt mµ cã thÓ truy cÊp ®íc nhiÒu lo¹i c¬ së d÷ liÖu cña nhiÒu hÖ qu¶n trØ C¬ së d÷ liÖu kh,c nhau.

§Ó gi¶i quyÖt vÊn ®Ò nµy, tËp ®oµn Microsoft ®· ®a ra mét chuÈn gäi lµ **Open DataBase Connectivity** (ODBC: sù nèi kÖt C¬ së d÷ liÖu më).

# Active Server Pages

## IV. Sóng ký nguân d÷ liÖu víi ODBC

### 1. Gií thiÖu



- **Drivers Manager (Tr×nh qu¶n lý c,c tr×nh ®iÒu khiÓn):** ®©y lµ mét bé phËn thi hµnh nhÚ' mét kî trung gian gi÷a øng dÔng vµ tr×nh ®iÒu khiÓn (Driver). Drivers Manager lÊy th«ng tin tõ øng dÔng vµ kÝch ho¹t tõng tr×nh ®iÒu khiÓn thÝch hîp cho viÖc truy cËp tõng lo¹i c¬ së d÷ liÖu.
- **Driver (Tr×nh ®iÒu khiÓn):** cung cÊp c,c tr×nh ®iÒu khiÓn ®Ó truy cËp c¬ së d÷ liÖu. Tr×nh ®iÒu khiÓn nµy chÞu sù ®iÒu khiÓn cña c,c c©u truy vÊn SQL vµ tr¶ kÕt qu¶ vÒ cho øng dÔng.
- **DataSource (Nguân d÷ liÖu):** ®©y lµ nguồn d÷ liÖu cÇn truy cËp ®Ó lÊy th«ng tin.

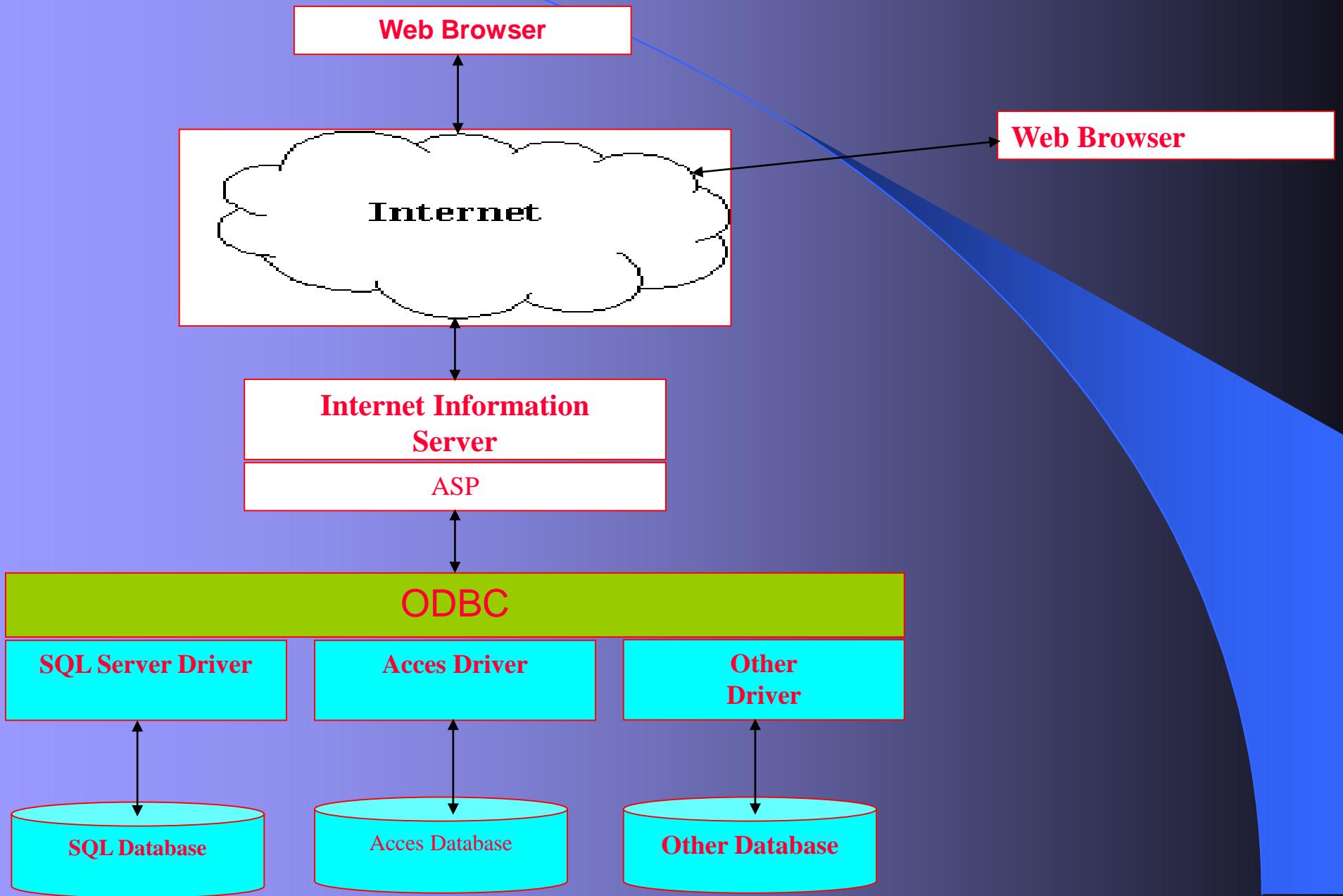
# Active Server Pages

## IV. §”ng ký nguân d÷ liÖu vїi ODBC

### 1. Giїi thiÖu

NhƯ vїy viÖc sö dông ASP kÕt h?p vїi ODBC ®Ó truy cËp th«ng tin trong c,c c¬ së d÷ liÖu qua m¹ng diÖn r eng cã thÓ ®Uc m« t¶ ng¾n g n qua s¬ ®å sau:

# Active Server Pages



# Active Server Pages

## IV. §”ng ký nguân d÷ liÖu vїi ODBC

### 1. Giїi thiÖu

- + Tõ Web Browser, ngїi sõ dông y u c u truy v n d÷ liÖu b»ng c, ch g i c,c c u l nh SQL ®Ôn Web Server (l u IIS), IIS nh n th y vi c truy c p c¬ s  d÷ liÖu c n ®Ôn ASP th  sõ dông ASP.
- + ASP s  g i c,c th ng tin c n thi t ®Ôn ODBC v  ODBC s  l u  ch n tr nh ®i u khi n th ch h p ®  truy c p c¬ s  d÷ liÖu.
- + Sau khi x  l y xong, k t qu  tr  ng c l i cho ng u i sõ dông c ng   tr n Web Browser.

# Active Server Pages

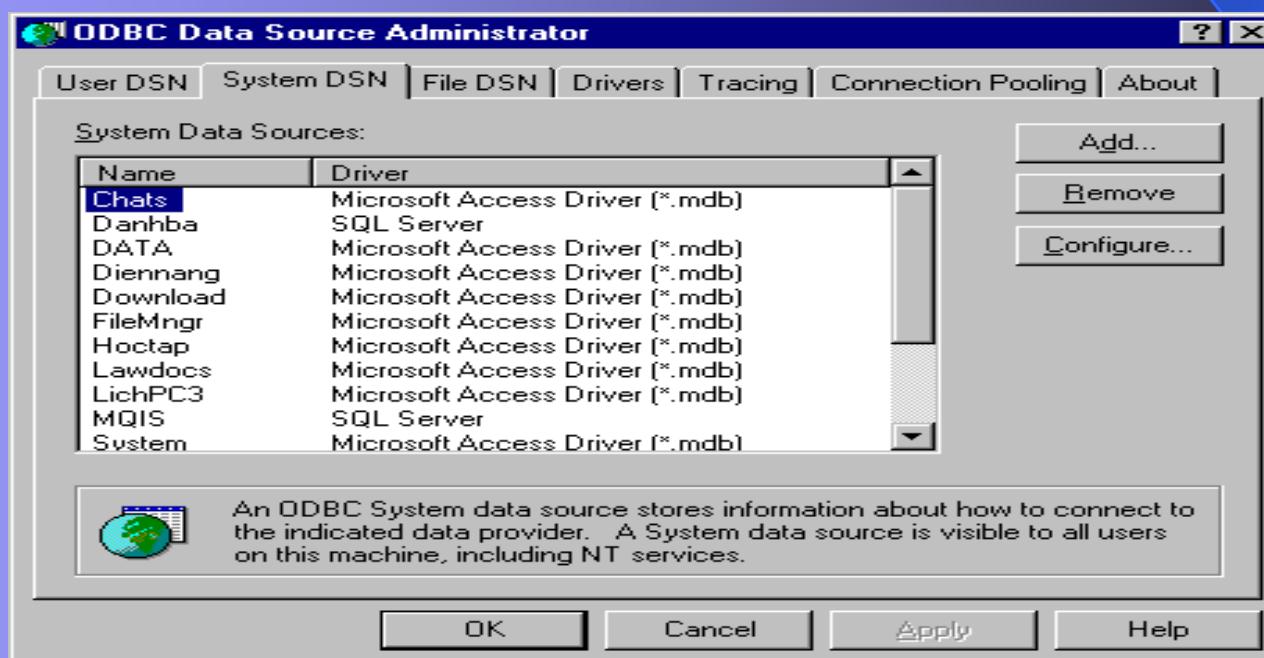
## IV. §”ng ký nguân d÷ liÖu víi ODBC

### 2. §”ng ký nguân d÷ liÖu víi ODBC

§Ó ®”ng ký d÷ liÖu víi ODBC tiÕn hµnh theo c,c bíc sau:

a. Tõ menu Start-> chän Settings -> chän Control Panel

b. KÝch ®«i biÓu tîng ODBC (32 bit), xuÊt hiÖn mµn h×nh nhÙ sau:



# Active Server Pages

## IV. §̄ng ký nguân d÷ liÖu víi ODBC

### 2. §̄ng ký nguân d÷ liÖu víi ODBC

Trong phÇn System Data Source liÖt kª tªn c,c c¬ së d÷ liÖu ®· ®U'ic ®·ng ký. Trong phÇn nµy g m c  hai phÇn:

+ Name: tªn ®· ®·ng ký (ch  y lµ tªn nµy kh ng ph¶i lµ tªn c na c¬ së d÷ liÖu, mµ ®©y chØ lµ tªn ®¹i diÖn cho c¬ së d÷ liÖu)

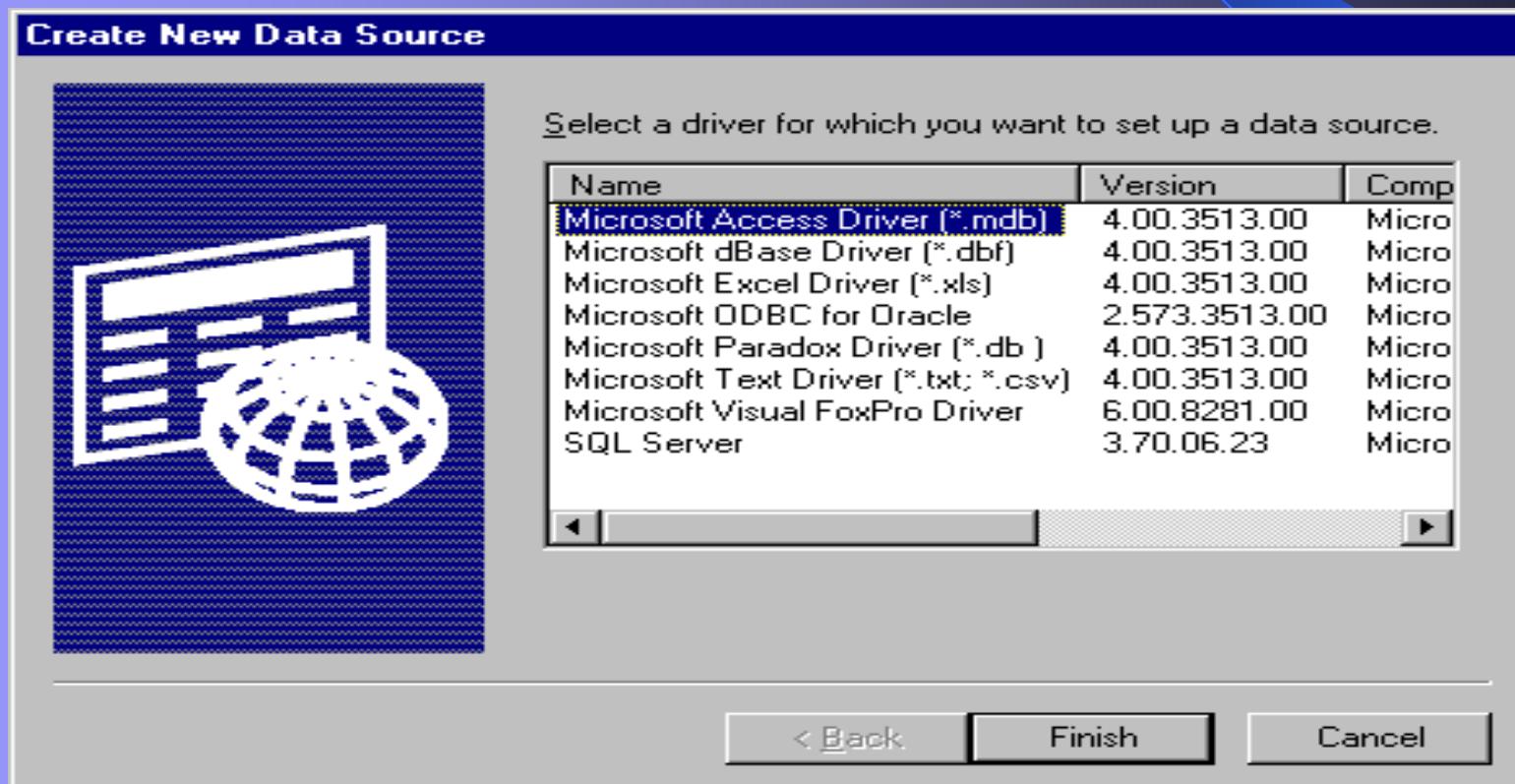
+ Driver: Tr×nh ®iÖu khiÓn ®Ó truy cËp t ng lo¹i c¬ së d÷ liÖu c  thÓ (nhU' Access, SQL Server, ...)

# Active Server Pages

## IV. Sđng ký nguân d÷ liÖu víi ODBC

### 2. Sđng ký nguân d÷ liÖu víi ODBC

c. SÓ ®“ng ký nguân d÷ liÖu mii, kÝch nót Add, xuÊt hiÖn cöa sæ nh sau:



# Active Server Pages

## IV. §̄ng ký nguân d÷ liÖu víi ODBC

### 2. §̄ng ký nguân d÷ liÖu víi ODBC

Trong cña sæ nøy cho phĐp b1n chän lùa tr×nh ®iÙu khiÓn cña tõng lo¹i c¬  
së d÷ liÖu mµ b1n sö dông.

- + NÕu C¬ së d÷ liÖu lµ SQL th× chän SQL Server
- + NÕu C¬ së d÷ liÖu lµ Access th× chän Microsoft Access Driver,...

Gi¶ sö chän Microsoft Access Driver, vµ kÝch nót Finish

# Active Server Pages

## IV. §“ng ký nguân d÷ liÖu víi ODBC

### 2. §“ng ký nguân d÷ liÖu víi ODBC

d. Sau khi chän Driver vµ kÝch nót Finish, xuÊt hiÖn cöa sæ nhÙ' bªn dÙ'íi:

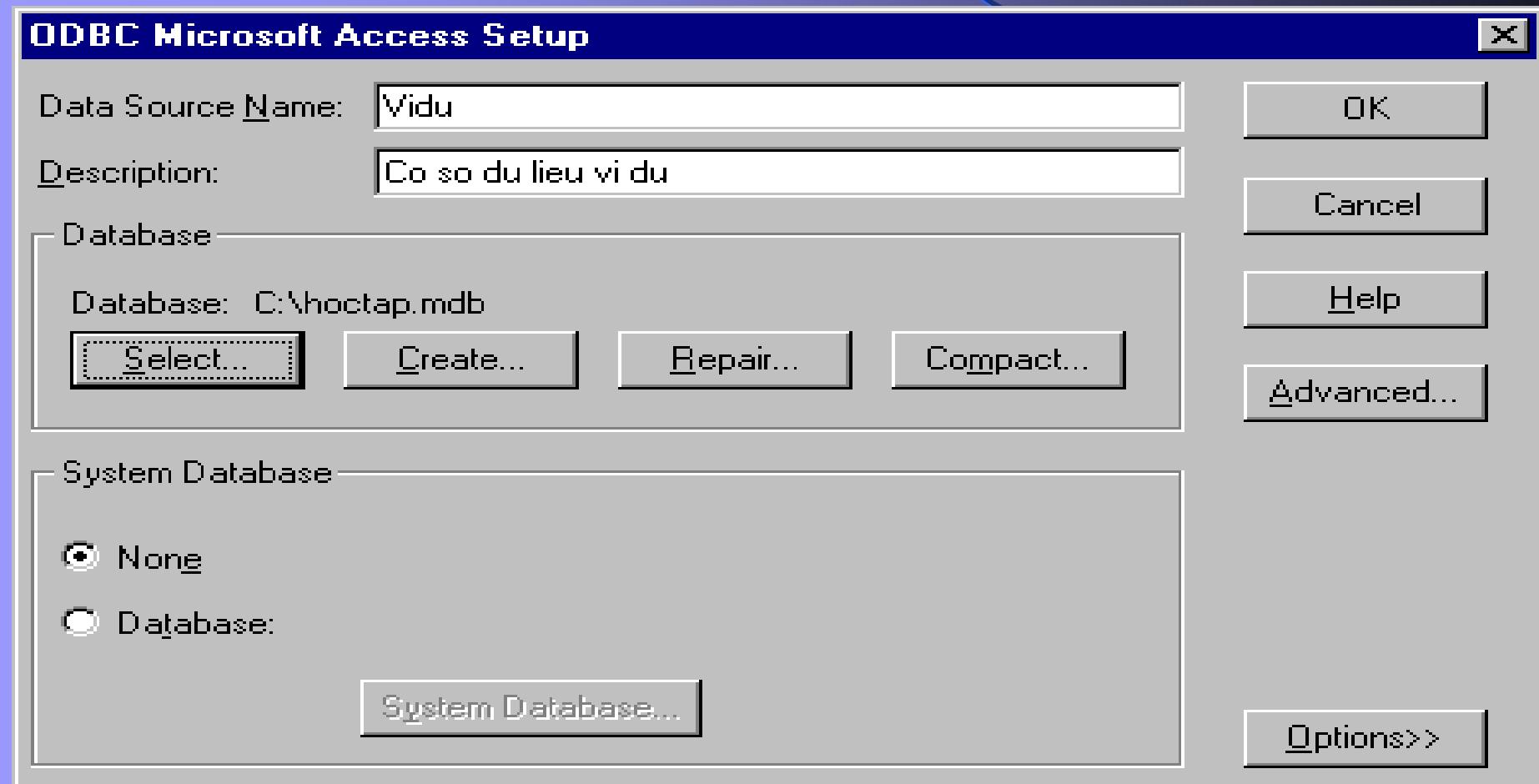
- Data Source Name: nhËp vµo tªn ®¹i diÖn cho c¬ së d÷ liÖu (tªn nµy sї ®U'ic dïng ®Ó nèi kÕt tui c¬ së d÷ liÖu trong ASP, vÝ dô c©u lÖnh trong ASP: conn.open "Vidu")
- Description: nhËp vµo phÇn m« t¶ cho c¬ së d÷ liÖu (phÇn nµy cã thÓ bá trèng)
- Database: tªn cña c¬ së d÷ liÖu cÇn truy cËp, ®Ó chän file C¬ së d÷ liÖu b¹n kÝch nót Select...

Sau khi chän C¬ së d÷ liÖu b¹n kÝch nót OK, vµ tiÖp têc kÝch OK ®Ó kÕt thóc viÖc ®“ng ký c¬ së d÷ liÖu víi ODBC.

# Active Server Pages

## IV. Sáng ký nguân d÷ liÖu víi ODBC

### 2. Sáng ký nguân d÷ liÖu víi ODBC



# Active Server Pages

## IV. §”ng ký nguân d÷ liÖu vїi ODBC

### 3. Giїi thiÖu vÒ IIS - Internet Information Server.

#### a. IIS lµ g×?

Microsoft Internet Information Server lµ mét øng dông server chuyÓn giao th«ng tin b»ng viÖc sö dông giao thøc chuyÓn ®æi siªu v n b¶n HTTP

#### b. IIS cã thÓ lµm ®Uic g×?

- XuÊt b¶n mét Home page l n Internet.
- T o c,c giao d ch th¬ng m i ®iÖn t o tr n Internet( Qu, tr nh giao d ch, ®Æt h ng...)
- Cho ph p ng i d ng t o xa cã thÓ truy xuÊt C¬ s  d÷ liÖu ( DataBase Remote Access)

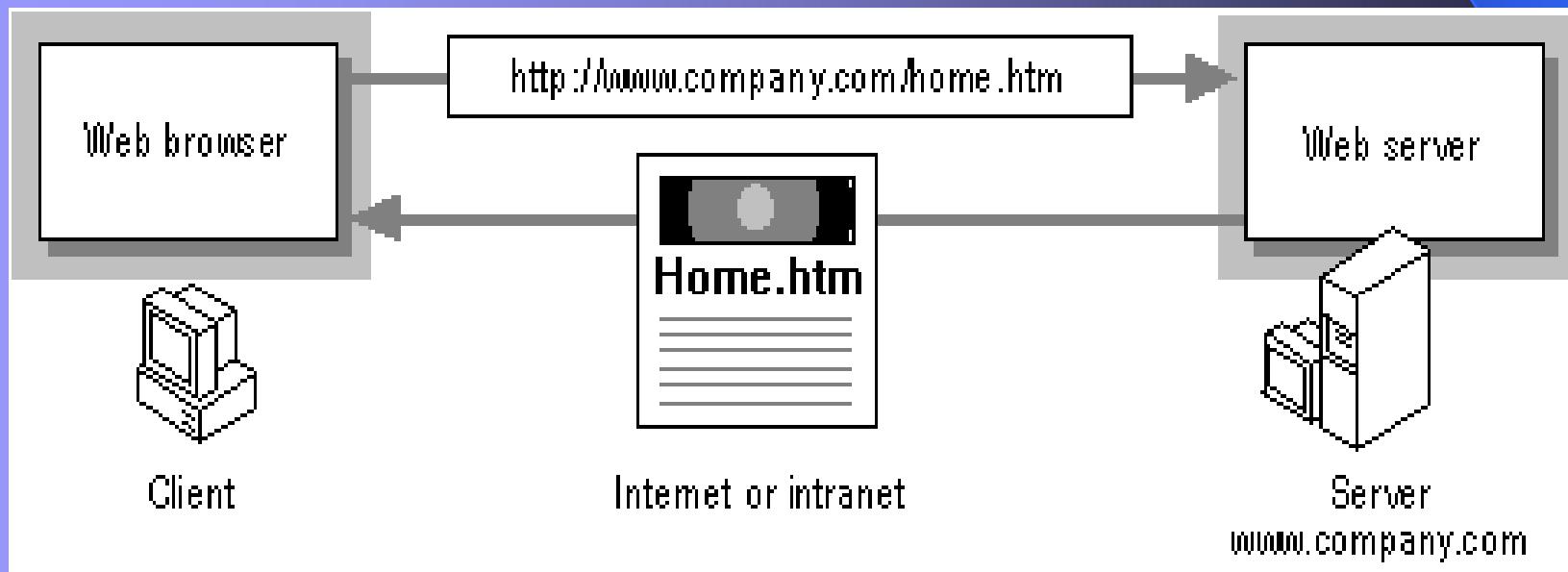
# Active Server Pages

## IV. §”ng ký nguân d÷ liÖu vii ODBC

### 3. Giii thiÖu vÒ IIS - Internet Information Server.

#### c. IIS ho’t ®éng nhU thÓ nµo ?

Web vÒ c¬ b¶n thuc sù lµ mét hÖ thèng c,c y u cÇu (Request) vµ c,c ®,p øng (Response). IIS ph¶n hai l¹i c,c y u cÇu ®Bi th«ng tin cña Web Browser. IIS l¾ng nghe c,c y u cÇu ®ã tõ phÝa Users tr n m t m ng s  d ng WWW.



# jspEC Project

# Java Server Pages

## Néi Dung

- \* Giới thiệu
- \* Cách hoạt động của JSP
- \* Cách xây dựng Web bằng JSP
- \* Các có phép của trong JSP
- \* Néoing vụ chuyển hình giữa các trang
- \* Các kèi tông mÆc Rõnh của JSP
- \* Phép thõc truy xuất sẽ để liêu

# Java Server Pages

## I. Giới Thiệu

JSP lµ c¹nh tranh cña Sun ®èi víi c«ng nghÖ ASP cña Microsoft.

ASP còng lµ c,c trang Web thùc hiÖn c¬ chÖ diÔn dÞch thay c,c lÖnh Java lµ c,c lÖnh Visual Basic. ASP kh«ng cã c¬ chÖ diÔn dÞch ra m· nhÞ ph@n nh JSP.

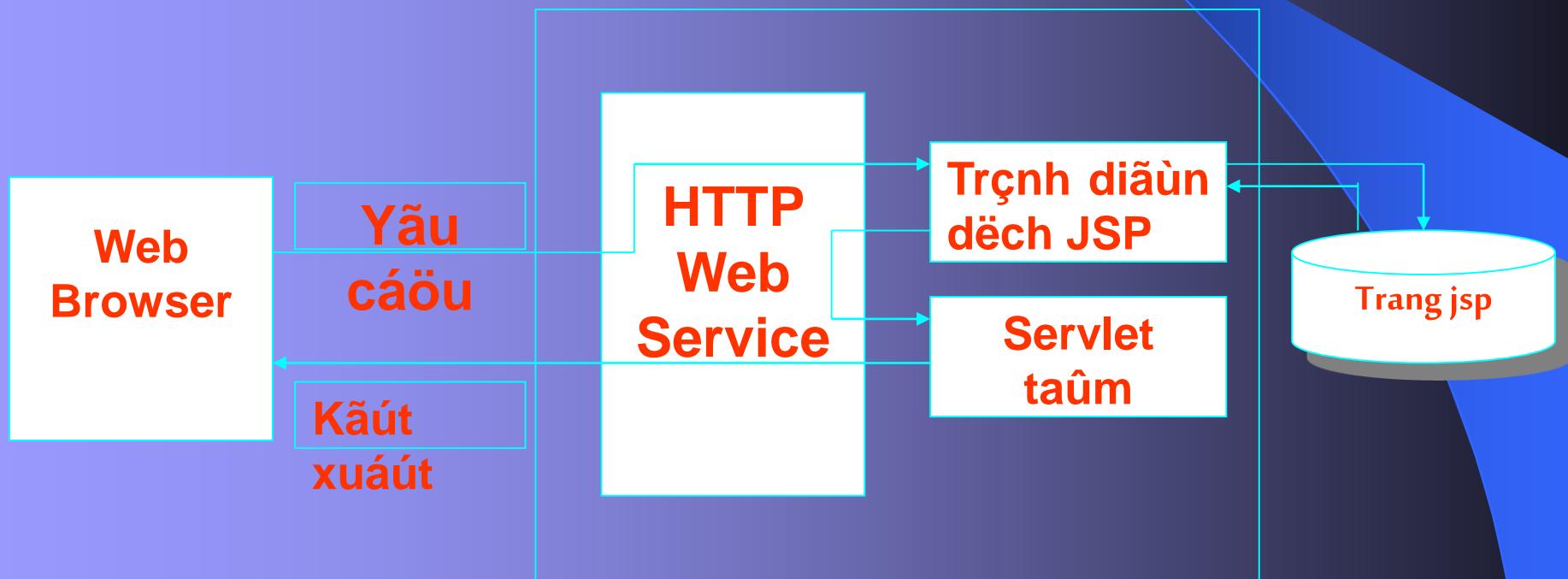
JSP (Java Server Page) lµ c,ch ®¬n gi¶n ®Ó ng«i d«ng tiÖp cËn ®âc híng lËp tr×nh Web phÝa m,y chñ hiÖu qñá vµ nhanh h¬n.

NÖu nh ®èi víi Servlet b¹n ph¶i viÖt m· Java vµ biªn dÞch b»ng tay tríc khi ®a vµo sö dông víi tr×nh chñ Web Server th× JSP kh«ng cÇn ®iÒu nµy

# Java Server Pages

## II. Cử Chỗ Hoạt Séng Cña JSP

Cử chỗ hoát ®éng cña trang JSP ®îc minh häa nhữ h×nh sau :



Java Web Server

# Java Server Pages

## II. C̄-Ch̄t Hōt S̄éng Cña JSP :

- + Khi b̄n ȳu c̄u m̄t trang JSP, tr̄nh chñ s̄i ®äc trang JSP t̄o ®Üa c̄ong,
- + Bé diÔn dÞch JSP s̄i diÔn dÞch m̄ lÖnh Java chøa trong trang JSP thµnh m̄t servlet.
- + Sau ®ã tr̄nh chñ Java Web Server s̄i triÖu gäi servlet tr̄i kÕt xuÊt thuÇn HTML vÒ cho tr̄nh kh, ch

# Java Server Pages

## III. Cach Xay Dung Web Bang JSP

Mot trang JSP vui phan me reng luu .jsp nhu luu mot trang HTML, ca thong tin them cua cua lanh Java bang cai

<%...%>

hoc: <jsp:scriptlet>... </jsp:scriptlet>.

Giua cai deu nhay luu cua cua lanh cua ngon ngua Java nham trong phan thon (<Body>... </Body>) cua tui liou, hoc phan Richter tien truuc cai tag <HTML> cua tui liou.

# Java Server Pages

## III. Cach Xay Dung Web Bang JSP

Muc tinh cua truc tinh cua met trang JSP

<%

---- Cac cua lanh Java e Roy ----

%>

<HTML>

---- Cac thanh phan Robanh nghua khac ----

<BODY>

<%

---- Cac cua lanh Java e Roy ----

%>

---- Cac Robanh dung khac cua HTML ----

</BODY>

</HTML>

# Java Server Pages

## IV. C,c Có Ph,p C¬ B¶n Cña JSP

TU¬ng tÙ thÎ HTML, thÎ lÖnh JSP cÙng bao gÙm thÎ mÙ vµ thÎ ®æng. Mçi thÎ cÙ  
c,c thuéc tÝnh qui ®Þnh c, ch sö dÙng thÎ.

### 1. ThÎ <jsp:scriptlet> hay <% %>

Víi thÎ nµy b¹n cÙ thÓ ®Æt c,c ®o¹n m· lÖnh Java ë gi÷a cÆp thÎ tu¬ng tÙ mét  
ch¬ng tr×nh Java th«ng thUêng. Bªn trong m· Java nÕu muèn kÕt xuÊt d÷ liÖu  
HTML b¹n gäi phU¬ng thøc println() cña ®èi tÙng out.

```
<jsp:scriptlet>
```

```
-----  
m· lÖnh Java
```

```
-----  
</jsp:scriptlet>
```

# Java Server Pages

## IV. C,c Có Ph,p C¬ B¶n Cña JSP

### 1. Thí <jsp:scriptlet> hay <% %>

Khi gÆp thi nµy, bé diÔn dÞch JSP sї biÕt ®Uric lÖnh cÇn thùc hiÖn tiÕp theo lµ m· lÖnh Java. B1n hoµn toµn sö dÔng mäi có ph,p cña Java è ®Cy, kÓ c¶ lÖnh chó thÝch th«ng thêng. NÕu muèn viÕt t¾t thx cã thÓ sö dÔng cÆp thi <% %> thay thÓ.

<%

-----

m· lÖnh Java

-----

%>

# Java Server Pages

## IV. C,c Có Ph,p C¬ B¶n Cña JSP

### 2. HiÓn thÞ kÕt xuÊt b»ng cõ ph,p <%= %>

Thay v× sõ dông có ph,p <%= %> ®Ó diÔn ®¹t mét khèi g m nhiÒu lÖnh, ta c  thÓ sõ dông có ph,p <%= %> chØ ®Ó hiÓn thÞ kÕt xuÊt cña m t gi, trÞ biÔn hay h m n o ®ã.

Ri ng biÓu th c n m trong thi  <%= %> kh ng ® c t n c ng b»ng d u ch m ph y () nh u c,c lÖnh hay khai b,o biÔn cña ng n ng  Java.

VÝ dô:

Welcome <%=username%>

username : l  t n biÔn do ng i d ng ® ng nh p.

# Java Server Pages

## IV. C,c Có Ph,p C¬ B¶n Cña JSP

### 3. Chln chó thÝch vuo trang JSP :

CÙng nhÚ Java, JSP cho phĐp b¹n dñng cÓ ph,p // ®Ó chÓ thÝch mÙt dßng m· lÖnh trong khi cÓ ph,p /\* \*/ ,p dÔng cho nhiÖu dßng. C,c dßng chÓ thÝch sÙ bá qua khi tr×nh chñ diÔn dÞch trang JSP.

```
<%
// LÊy biÖn d÷ liÖu mang tªn username tõ ®èi tîng session
String username=session.getAttribute("username");
%>
```

JSP cßn cung cÊp cho b¹n cÓ ph,p chÓ thÝch <%-- --%>. TÊt c¶ c,c khÙi lÖnh Java vµ HTML n»m gi÷a hai dÊu chÓ thÝch nµy sÙ ®íc tr×nh biªn dÞch trang bá qua khÙng quan tØm ®Ön.

VÝ dô : <%--
out.println("You will never see this line");
--%>

# Java Server Pages

## IV. C,c Có Ph,p C¬ B¶n Cña JSP

### 4. Thí khai b,o ph¬ng thøc vµ biÕn b»ng <%! %>

Ta cã thÓ dñg thí <%! %> ®Ó khai b,o mét biÕn hoÆc mét phU¬ng thøc dñg cho toµn trang jsp. BiÕn khai b,o ph¶i theo có ph,p cña ng«n ng÷ Java. Ph¬ng thøc vµ biÕn sau ®ã cã thÓ ®Uc gäi bÊt kú n¬i ®Cºu trong trang JSP.

### 5. Thí chØ dÉn biªn dÞch trang <%@ page%>

Thí <%@ page%> chØ dÉn mét sè tÝnh chÊt biªn dÞch ,p dông cho toµn trang jsp. Ta cã thÓ sö dông thí nµy ®Ó khai b,o c,c thuµ viÖn import cña Java, chØ ®Þnh tïy chän trang jsp cã cÇn gi÷ trªn cache bé nhí cña tr×nh chñ ®Ó t”ng tÙc hay kh«ng.

VÝ dô:

```
<%@ page import="java.sql.*" %>
```

# Java Server Pages

## IV. C,c Có Ph,p C¬ B¶n Cña JSP

### 6. Nhóng Vµ ChuyÓn Híng Gi÷a C,c Trang

Trong qu, tr×nh xö lý vµ chuyÓn d÷ liÖu cho form b¹n cã thÓ nhóng vµ triÖu gäi nh÷ng trang JSP kh,c víi trang hiÖn hµnh.

Cã thÓ ®a mét trang HTML hoÆc JSP vµo trang JSP hiÖn t¹i ®Ó hc trî thªm th viÖn hay tÝnh n÷ng bæ sung nµo ®ã. Hay ghi nhËn ®Uc d÷ liÖu submit tõ tr×nh kh,ch, trang JSP nhËn ®Uc d÷ liÖu cã thÓ chuyÓn híng hoÆc triÖu gäi ®Õn trang JSP kh,c.

# Java Server Pages

## IV. C,c Có Ph,p C¬ B¶n Cña JSP

### 6.1. Nhóng file vµo trang víi chØ thØ include file :

Ta cã thÓ nhóng mét trang jsp hay trang html vµo trang jsp hiÖn hµnh b»ng thÎ <%@ include file %>.

ThÎ nµy t¬ng tù nhU chØ dÉn #include trong ng«n ng÷ C. Có ph,p ®Çy ®ñ cña thÎ nµy lµ :

```
<%@ include file="URL or Filepath"%>
```

### 6.2. Sö dÔng thÎ <jsp:include>

ChØ thØ <@ include %> chØ dÖng ®Ó nhóng c,c m· nguân tÜnh. NÕu muèn nhóng kÖt qu¶ kÖt xuÊt tõ c,c trang jsp, servlet hay html kh,c vµo trang hiÖn hµnh b¹n sö dÔng thÎ <jsp: include> víi có ph,p nh sau :

```
<jsp:include page="filename" flush="true" />
```

Lu ý: <jsp: include> ®a kÖt qu¶ kÖt xuÊt cña trang ®íc nhóng vµo trang hiÖn hµnh, trong khi <%@ include %> dÖng ®Ó ®Ua m· nguân cña trang ®íc nhóng vµo trang hiÖn hµnh.

# Java Server Pages

## IV. C,c Có Ph,p C¬ B¶n Cña JSP

### 6.3 ChuyÓn tiÖp ®Õn trang kh,c víi thî <jsp:forward>

Ta sö dông thî <jsp:forward> ®Ó triÖu gäi vµ chuyÓn d÷ liÖu cho trang jsp kh,c xö lý.

VÝdô: Trang Page1.jsp chØ hiÓn thþ th«ng tin ®Ó ngêi dïng nhËp vµo username vµ password

```
<html>
  <form method=post action=page1.jsp>
    username : <input type=text name="username" >
    password : <input type=password name="password" >
    <input type=submit value="Login" >
  </form>
</html>
```

# Java Server Pages

## IV. C,c Có Ph,p C¬ B¶n Cña JSP

### 6.3 ChuyÓn tiÕp ®Õn trang kh,c vїi thÎ <jsp:forward>

Trang Page2.jsp kiÓm tra nÕu username kh,c r ng sї chuyÓn tiÕp quyÒn x o l y file Page3.jsp, ng c l i nÕu username b ng r ng sї ®a ra th ng b,o y u c u ng i d ng nh p v o.

```
<% if (!request.getParameter("username").equals(" ")) {  
    %>  
<jsp:forward page="Page3.jsp" />  
<%} else { out.println(" Please input username and password ");}  
%>  
<html>  
    Your username :<%=request.getParameter("username")%>  
<br>  
    Your password :<%=request.getParameter("password")%>;  
<br>  
</html>
```

# Java Server Pages

## IV. Các Cách Chuyển Trang Của JSP

### 6.4. Chuyển hướng sang trang mới với sendRedirect()

Ngoài việc chuyển hướng bằng cách sử dụng phương thức sendRedirect() để chuyển hướng về trang JSP khác xem lý.

Sử dụng nhau giữa chuyển hướng (forward) và chuyển hướng (redirect) để lệnh sendRedirect() thực hiện truy cập giao diện trang mà không chuyển tham số của trang hiện tại cho trang chuyển hướng.

Ví dụ: Tạo Page2.jsp bằng cách chuyển hướng đến trang Page3.jsp như sau :

```
<%
```

```
-----
```

```
response.sendRedirect("Page3.jsp");
```

```
%>
```

# Java Server Pages

## V. C, c §èi TÙ̄ng MÆc §Þnh Cñá Trang JSP

Trxnh diÔn dÞch JSP cho phÐp ta sö dông mét sè ®èi tÙ̄ng ®· khai b,o tríc. §iÒu nµy sї gióp ta viÓt m· lÖnh trong trang jsp nhanh h¬n servlet.

1. §èi tîng out : XuÊt ph,t tõ líp java.ioPrintWriter. B1n cã thÓ sö dông ®èi tÙ̄ng nµy ®Ó ®Þnh d¹ng kÕt xuÊt gõi vÒ m,y kh,ch.

VÝ dô: <% out.println("Result "+7\*3); %>

# Java Server Pages

## V. C,c §èi TƯ̄ng MÆc §Þnh Cña Trang JSP

### 2. §èi tîng Request :

XuÊt ph,t tõ líp HttpServletRequest, ®èi tƯ̄ng nµy gióp b¹n lÊy vÒ c,c tham sè hay d÷ liÖu do tr×nh kh,ch chuyÓn lªn.

Mét trong nh÷ng øng dông th«ng thuÊng nhÊt cña ®èi tƯ̄ng request lµ n¾m gi÷ c,c tham sè. B¹n cã thÓ thÊy ®iÒu nµy b»ng c,ch gäi ph¬ng thøc getParametter() cña request.

PhÙ¬ng thøc nµy kÕ thoa tõ líp cha lµ javax.servlet.ServletRequest. PhÙ¬ng thøc nµy nhËn tªn tham sè vµ tr¶ vÒ gi, trÞ chuçi tÙ¬ng øng víi tªn cña tham sè ®ã.

VÝdô: LÊy vÒ c,c th«ng tin ®“ng nhËp tõ file Bc\_td1.jsp

```
String USERNAME=request.getParameter("username");
```

```
String PASSWORD=request.getParameter("password");
```

# Java Server Pages

## V. C,c §èi TƯ̄ng MÆc §Þnh Cña Trang JSP

### 3. §èi tûng Session :

XuÊt ph,t tõ lïp HttpSession. B¹n sô dông ®èi tûng session ®Ó theo dãi kÕt nèi vµ lñu vÕt mét phiªn lµm viÖc gi÷a tr×nh kh,ch vµ tr×nh chñ.

§èi tûng session tham chiÕu ®Õn ®èi tûng javax.servlet.http.HttpSession. §èi tûng session ®Úc khëi t¹o b»ng c,ch gäi phÚ¬ng thøc pageContext.getSession() ®Ó t¹o ra trang servlet.

```
session = pageContext.getSession();
```

Khi cÇn mét biÕn nµo ®ã cã gi, trØ toµn cÔc tõ khi më cho ®Õn khi kÕt thøc tr×nh duyÖt ®ã, b¹n nªn nghÜ ®Õn session. §èi tûng session sї thÙc hiÕn viÖc ®ã gióp b¹n. §èi tûng session t¹o biÕn cÔc bé cho phØp lñu mét gi, trØ nµo ®ã tõ trang JSP nµy sang trang JSP kh,c.

# Java Server Pages

## V. Các Điều Tự động Mặc Định Của Trang JSP

### 3. Điều Tự động Session :

Výdô: Là lưu trữ thông tin vào phiến lưu trữ session ta thục hiện như sau:

```
session.putValue("NGAY", NGAY);
session.putValue("THANG", THANG);
session.putValue("NAM", NAM);
```

Sau khi ghi thông tin vào session, khi cẩn lây vào gi, trê trong session ta thục hiện như sau :

**String NGAY=(String) session.getValue("NGAY");**

**String THANG=(String) session.getValue("THANG");**

**String NAM=(String) session.getValue("NAM");**

# Java Server Pages

## V. C,c §èi TÙng MÆc §Þnh Cña Trang JSP

### 4. §èi tÙng Response :

§èi tÙng response cho phĐp ph¶n h¢i th¢ng tin x¢ lý t¢ tr×nh chñ tr¢ vÒ tr×nh duyÖt. Th¢c ra ®èi tÙng response thÙêng g¢i phÙ¬ng th¢c response.getWriter() ®Ó th¢c hiÖn kÖt xuÊt. C,c phÙ¬ng th¢c mµ ®èi tÙng response thÙêng ®îc liÖt kª nhÙ' sau :

Lia'n kÖt tr¢c tiÖp ®Ön trang Bc\_td2.jsp

```
response.sendRedirect("Bc_td2.jsp");
```

§Þnh kÖt xuÊt ra HTML hay text

```
response.setContentType("text/html");
```

LÊy d¢ liÖu trong vинг ®Öm nh¢ OutputStream();

```
response.getOutputStream();
```

# Java Server Pages

## V. C, c §èi TÙ'ing MÆc §Þnh Cña Trang JSP

### 5. §èi tÙ'ing Exception :

§èi tÙ'ing exception chØ tân t¹i trong trang xö lý lçi (errorPage). Nã dñng ®Ó tham chiÕu ®Õn nguyªn nh©n g©y ra lçi mµ trang error cã liªn quan.

# Java Server Pages

## VI. Truy XuÊt CSDL B»ng JSP :

### 1. JDBC vµ tr×nh ®iÒu khiÓn truy xuÊt c¬ së d÷ liÖu

C«ng viÖc lËp tr×nh mµ JSP thUêng thùc hiÖn nhÊt ®ã lµ lÙu tr÷ th«ng tin vµ truy xuÊt c¬ së d÷ liÖu (database) cung cËp th«ng tin cho tr×nh kh, ch.

ViÖc truy xuÊt c¬ së d÷ liÖu trong JSP dùa vµo tr×nh truy xuÊt JDBC mµ Java hç trî. JDBC lµ c,c tr×nh ®¹o diÔn cho phĐp b¹n truy cËp vµo c¬ së d÷ liÖu cña c,c hÖ nhÙ Access, MS SQL Server, Oracle,... b»ng ng«n ng÷ truy vÊn SQL. Cã 4 kiÓu JDBC nh sau:

- + *JDBC sô dông cÇu nèi ODBC*
- + *JDBC kÕt nèi trùc tiÖp vµi c,c tr×nh ®iÒu khiÓn c¬ së d÷ liÖu.*
- + *JDBC kÕt nèi th«ng qua c,c øng dông m¹ng trung gian.*
- + *JDBC kÕt nèi th«ng qua c,c tr×nh ®iÒu khiÓn ®Æc thi ë xa*

# Java Server Pages

## VI. Truy XuÊt CSDL B»ng JSP :

### 1. JDBC vµ tr×nh ®iÒu khiÓn truy xuÊt c¬ së d÷ liÖu

+ KiÓu 1: JDBC sö dÔng cÇu nÈi ODBC

NÕu b¹n ®· tÔng lµm viÖc vÍi c,c hÖ c¬ së d÷ liÖu trªn windows b¹n sї biÖt ®Õn c,ch kÕt nÈi ODBC (Open Database Connectivity).

ODBC lµ c,ch kÕt nÈi tæng qu,t nhÊt ®Õn mÄi hÖ c¬ së d÷ liÖu th«ng qua tr×nh qu¶n lý ODBC cña Windows. Mçi hÖ c¬ së d÷ liÖu cung cÊp mÙt tr×nh ®iÒu khiÓn (driver) cã kh¶n n»ng ®äc hiÓu ®Uric c¬ së d÷ liÖu cña chóng.

Tr×nh ®iÒu khiÓn nµy sї ®Uric ®»ng kÝ vÍi bÉ qu¶n lý ODBC. Java cung cÊp cho b¹n kh¶n n»ng sö dÔng JDBC ®Ó gäi ODBC vµ b»ng c,ch nµy b¹n cã thÓ kiÓm so,t rÊt nhiÓu hÖ qu¶n trØ c¬ së d÷ liÖu kh,c nhau.

# Java Server Pages

## VI. Truy XuÊt CSDL B»ng JSP :

1. JDBC vµ tr»nh ®iÓu khiÓn truy xuÊt c¬ së d÷ liÖu

+ KiÓu 2: JDBC kÕt nèi trùc tiÕp víi c,c tr»nh ®iÓu khiÓn c¬ së d÷

liÖu.

Khi Java ph,t triÓn, c,c nhµ cung cÊp hç trî c,c tr»nh ®iÓu khiÓn JDBC kÕt nèi trùc tiÕp víi c,c tr»nh ®iÓu khiÓn ®Æc thi cña mçi hÖ c¬ së d÷ liÖu (native database driver) mµ kh«ng cÇn ph¶i qua trung gian ODBC cña windows n÷a. C,c kÕt nèi nµy nhanh vµ hiÖu qu¶ h¬n kiÓu 1 JDBC-ODBC

# Java Server Pages

## VI. Truy XuÊt CSDL B»ng JSP :

1. JDBC vµ tr»nh ®iÒu khiÓn truy xuÊt c¬ së d÷ liÖu

+ KiÓu 3: JDBC kÕt nèi th«ng qua c,c øng dông m¹ng trung gian.

Ngøy nay, c,c øng dông ®Òu g¾n liÒn víi Internet. Theo m« h»nh lËp tr»nh ph@n t,n, øng dông trung gian (middle-ware) trªn my chñ sї chÞu tr,ch nhiÖm kÕt nèi víi c¬ së d÷ liÖu.

NhiÖm vô cña tr»nh kh,ch lµ sö dông nh÷ng tr»nh ®iÒu khiÓn JDBC gän nhÑ cã kh¶ n»ng giao tiÓp víi øng dông trung gian qua Internet ®Ó truy vÊn d÷ liÖu.

# Java Server Pages

## VI. Truy XuÊt CSDL B»ng JSP :

1. JDBC vµ tr»nh ®iÒu khiÓn truy xuÊt c¬ së d÷ liÒu

+ KiÓu 4: JDBC kÕt nèi th«ng qua c,c tr»nh ®iÒu khiÓn ®Æc thi ë xa.

M« h»nh kÕt nèi theo kiÓu 3 do ph¶i th«ng qua øng d»ng trung gian (middle-ware) n¤n cã thÓ ¶nh h¢ng ®Õn tèc ®é truy xuÊt c¬ së d÷ liÒu phÝa m,y kh,ch. Java ®a ra kÕt nèi JDBC kiÓu 4 cho phĐp tr»nh kh,ch sö d»ng c,c tr»nh ®iÒu khiÓn gän nhÑ kÕt nèi trùc tiÕp vµo tr»nh ®iÒu khiÓn c¬ së d÷ liÒu ®Æc thi (thuÇn java) tr¤n m,y chñ ë xa th«ng qua m¹ng Internet.

# Java Server Pages

## VI. Phư¬ng Tho¬c Truy Xu¬t CSDL B»ng JSP :

### 2. Ph¬ng Tho¬c Truy Xu¬t CSDL B»ng Jsp

mét k¬t n¬i v¬i CSDL :

§Ó k¬t n¬i v¬i CSDL, JDBC ®ßi hái hai y¬u tè lµ **tr¬nh ®i¬u khiÓn** (tU¬ng øng v¬i c,c kiÓu k¬t n¬i) vµ **th¬ng tin ®Ó k¬t n¬i** (®pa chØ m,y chñ, t¬n tui kho¶n, mËt khÈu,...).

+ **N¬p tr¬nh ®i¬u khiÓn**: Ta sö d¬ng tr¬nh ®i¬u khiÓn mi¬n phÝ do Sun cung c¬p mang t¬n JdbcOdbcDriver. Ta khai b¬o t¬n l¬p nµy nh sau :

```
String drivername="sun.jdbc.odbc.JdbcOdbcDriver";
```

TiÓp ®Ön ta gäi phư¬ng tho¬c tÜnh forName cña l¬p Class ®Ó n¬p vµ kh¬i t¬o tr¬nh ®i¬u khiÓn :

```
Class.forName("drivername").newInstance();
```

2.1 Thùc hiÖn

# Java Server Pages

## VI. Phép Thoát Truy Xuất CSDL B»ng JSP :

### 2. Phép Thoát Truy Xuất CSDL B»ng Jsp

#### 2.1 Thủc hiÖn mét kÕt nèi víi CSDL :

+ *T’o th«ng tin kÕt nèi*: Lúc nøy ta ®· hojn tojn cã kh¶ n”ng sö dông tr»nh ®iÙu khiÓn JDBC-ODBC ®Ó truy xuÆt d÷ liÙu. Tuy nhiªn tr»nh ®iÙu khiÓn cÇn biÕt thªm nh÷ng th«ng tin cô thÓ nh tui kháan ®”ng nhËp, th«ng tin kÕt nèi,...

Ta ®Æt nh÷ng th«ng tin nøy trong chuçi kÕt nèi nhU sau :

```
String url="jdbc:odbc:t'an CSDL";
```

```
String username="";
```

```
String password="";
```

ë ®©y chuçi jdbc:odbc: lµ b¾t buéc víi tr»nh ®iÙu khiÓn JDBC-ODBC. TiÕp sau chuçi nøy lµ th«ng tin kÕt nèi cô thÓ ®Õn CSDL do ODBC qui ®Þnh.

# Java Server Pages

## VI. PhƯ-ng Thøc Truy XuÊt CSDL B»ng JSP :

### 2. PhƯ-ng Thøc Truy XuÊt CSDL B»ng Jsp

#### 2.1 Thùc hiÖn mét kÕt nèi víi CSDL :

+ *T'о th«ng tin kÕt nèi:*

TiÕp tõc ta t'о mét ®èi tîng Connection b»ng c, ch gäi ph-ng thøc getConnection() cña lïp DriverManager ®Ó y'a u cÇu trxnh ®iÒu khiÓn n'p bëi Class.forName() tríc ®ã tiÕp nhËn th«ng tin vµ thùc thi kÕt nèi nh sau:

```
Connection conn = null;
```

```
conn = DriverManager.getConnection(url, "username", "password");
```

url : chuçi n'a u l'a n ®Æc ®iÓm cña CSDL cä d'ng jdbc:subprotocol:subname

subprotocol : giao thøc con t-nqg øng víi lo'i CSDL

subname : t'a n cña CSDL

username : t'a n ngîi dñng khi ®-ng nhËp vµo CSDL

password : mËt khÈu ngîi dñng khi ®-ng nhËp CSDL

# Java Server Pages

## VI. Phép Tác Truy Xuất CSDL Bằng JSP :

### 2. Phép Tác Truy Xuất CSDL Bằng Jsp

#### 2.2 Thúc hiÖn c,c lÖnh :

C,c chÜng tr×nh JSP chØ thÙc hiÖn ®Üc c,c lÖnh trªn CSDL th«ng qua ®èi tÜring Statement. Sëi tÜring nµy ®Üc dïng ®Ó thÙc thi mäi c©u lÖnh sql.

Ta t¹o ®èi tÜring Statement tõ kÕt nèi *conn*

```
Statement stmt =null;  
stmt = conn.createStatement();
```

C,c c©u lÖnh sql cã thÓ ®îc thÙc hiÖn tøc thx th«ng qua ®èi tÜring Statement, cã thÓ lµ mét c©u lÖnh biªn dÞch trÜc (®èi tÜring PreparedStatement) hay cã thÓ lµ mét lÖnh gäi c,c thñ tøc cµi s½n (store procedure) trong CSDL (®èi tÜring CallableStatement).

# Java Server Pages

## VI. Phép Thoát Truy Xuất CSDL Bằng JSP :

### 2. Phép Thoát Truy Xuất CSDL Bằng Jsp

#### 2.2 Thực hiện câu lệnh:

```
rs=stmt.executeQuery(sql);
```

hoặc: rs=stmt.executeUpdate(sql);

Câu lệnh sql cần thoát thực hiện thông qua:

+ phím thoát `executeQuery()` -nếu không quay lại mét ®èi tÙng `ResultSet`

+ phím thoát `executeUpdate()`-nếu không quay lại mét sè guyán cho biêt tæng sè câu dñng d÷ liêu chÞu t,c dñng cña câu lệnh vña thực hiện.

# Java Server Pages

## VI. PhƯ̄ng Thøc Truy XuÊt CSDL B»ng JSP :

### 3. §o¹n lÖnh dñng JSP truy cËp d÷ liÖu trªn Server

// N¹p tr×nh ®iÒu khiÓn JDBC

- `String drivername="sun.jdbc.odbc.JdbcOdbcDriver";`
- `Class.forName(drivername).newInstance();`

// Th«ng tin kÕt nèi JDBC-ODBC

- `String url="jdbc:odbc:Database_Name";`
- `String username="";`
- `String password="";`

# Java Server Pages

## VI. Phép Thoát Truy Xuất CSDL Bằng JSP :

### 3. Sử dụng lệnh đăng JSP truy cập dữ liệu trên server

- `Connection con=null;`  
// Thực hiện kết nối
- `con=DriverManager.getConnection(url, username, password);`
- `Statement stmt=null;`
- `stmt=con.createStatement();`  
// Thực thi câu lệnh truy vấn với kết nối quét trả về
- `ResultSet rs=null;`  
// Tạo câu lệnh truy vấn
- `String sql="select * from TABLE_NAME";`
- `rs=stmt.executeQuery(sql);`

# NGÔN NGỮ LẬP TRÌNH PHP

## I. Giới thiệu về PHP

### 1. Giới thiệu

+ PHP (*Personal Home Page*) là một kịch bản trên Server (Server Script)

+ Là một mảng truyềng lập trình cung cấp cho việc kết hợp HTML, ngôn ngữ kịch bản (Scripting) như Javascript, và các thunh phcn đặc biệt trong các ngôn ngữ nhaum tạo ra một ứng dụng Internet minh mẽ và hoàn chỉnh.

+ PHP là một phcn mở mảng nguồn mẻ (Open Source)

# NGÔN NGỮ LẬP TRÌNH PHP

## I. Giới thiệu về PHP

### 2. PHP file là gì?

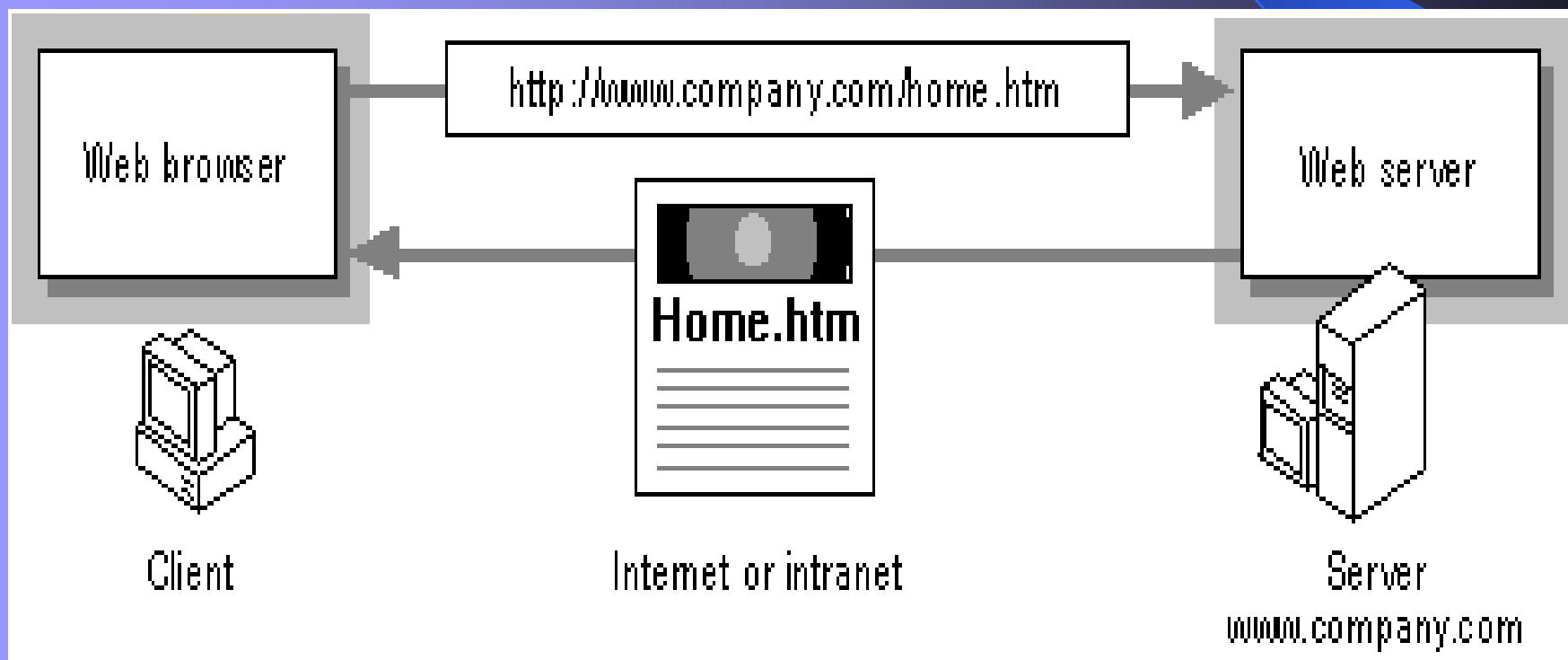
- + File có tên viết phẳng mẽ rẽng .PHP. Trong file này chứa các thẻ HTML, các kí tự bao gồm Scripting như JavaScript.
- + Khi cần sửa sai các file PHP ta cần ghi lại trên server riêng. Vào những lần sau khi trang PHP được tải, các Script trong file PHP từ đó sẽ被执行.
- + Các trắc mực gãy gièng ngón ngón lặp trình C

# NGÔN NGỮ LẬP TRÌNH PHP

## I. Giới thiệu về PHP

### 3. Cách thức lồng vi Óc cña trang PHP

+ Trang HTML tÙnh:

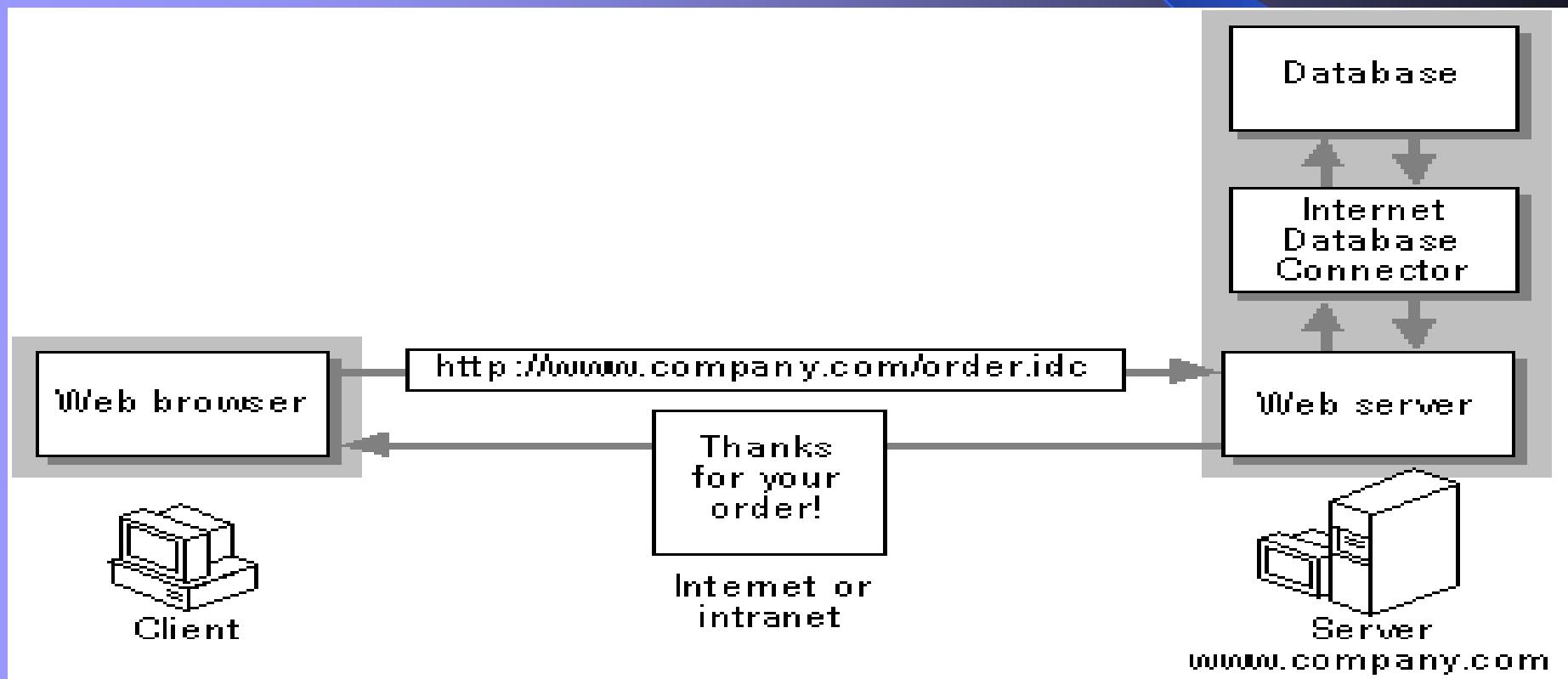


# NGÔN NGỮ LẬP TRÌNH PHP

## I. Giới thiệu về PHP

### 3. Cách thức lưu trữ của trang PHP

+ Trang PHP :

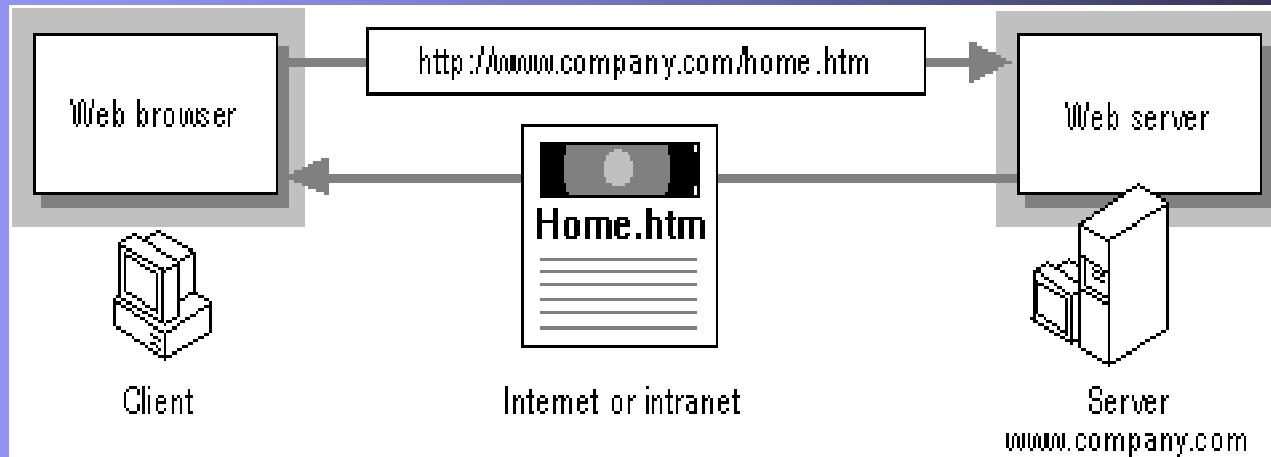


# NGÔN NGỮ LẬP TRÌNH PHP

## I. Giới thiệu về PHP

### 4. Web Server Apache thực thi trang PHP

- + Web, vÒ c¬ b¶n thÙc sÙ lÙ mét hÖ thÙng c,c yªu cÇu (Request) vµ c,c ®,p øng (Response).
- + Webserver ph¶n h¬i l¹i c,c yªu cÇu ®ßi th«ng tin cña Web Browser. N¬ l¾ng nghe c,c yªu cÇu ®ã t¬ phÝa Users trªn mét m¹ng sô dông WWW.



# NGÔN NGỮ LẬP TRÌNH PHP

## II. Cách xây dựng Web bằng PHP

### 1. Nhúng PHP vào HTML

Có 4 cách để nhúng mã PHP vào trong HTML:

+ Đóng cùp thi <? ..... ?>

+ Đóng <?php .....?>

+ Đóng <Script language = "php">

.....

</script>

+ Đóng <% ..... %>

# NGÔN NGỮ LẬP TRÌNH PHP

## II. Cách xây dựng Web bằng PHP

### 1. Nhúng PHP vào HTML

Ví dụ 1: Lưu file sau lên đĩa với tên vd.php

```
<html>
```

```
<head>
```

```
    <title>Testing page</title>
```

```
</head>
```

```
<body><?php echo "Hello you!"; ?>
```

```
</body>
```

```
</html>
```

# NGÔN NGỮ LẬP TRÌNH PHP

## II. Cách xây dựng Web bằng PHP

### 2. Một số chú ý khi bắt đầu lập trình PHP

#### a. Lệnh echo :

Lệnh này dùng để xuất chuỗi văn bản về cho trình duyệt.

Ví dụ: Câu lệnh *echo "Hello, world!"*; trình duyệt sẽ nhận được chuỗi văn bản Hello, world!.

Câu lệnh *echo 1+2;* sẽ trả về cho trình duyệt chuỗi văn bản 3.

Và câu lệnh *echo 5+2, "Hello world!"*; sẽ trả về trình duyệt chuỗi 7Hello world!

# NGÔN NGỮ LẬP TRÌNH PHP

## II. Cách xây dựng Web bằng PHP

### 2. Một số chú ý khi bắt đầu lập trình PHP

#### b. Phân cách các lệnh bằng dấu chấm phẩy (;

Tương tự như các ngôn ngữ lập trình khác, một câu lệnh của PHP được kết thúc bằng dấu chấm phẩy (;

Ví dụ:      `echo 1+2;`      `echo "Hello, world!"`;

#### c. Chú thích trong chương trình

- Chú thích 1 dòng được bắt đầu bằng // hoặc #, và những gì được ghi từ đó về sau là chú thích.

- Chú thích nhiều dòng được bắt đầu bằng /\* và kết thúc bằng \*/, những gì ở giữa là chú thích.

# NGÔN NGỮ LẬP TRÌNH PHP

## II. Cach xay dung Web bang PHP

### 2. Mot so chuc nang cua PHP

#### c. Chuc nang chay trinh

Ví dụ:

```
<?php  
//Day la chuc nang 1 dong  
#Day cung la chuc nang 1 dong  
/* Day la chuc nang nhieu dong Doan chuong trinh  
sau se in ra chuoi abc123 */  
echo "abc123";  
?>
```

# NGÔN NGỮ LẬP TRÌNH PHP

## III. Lập trình PHP

### 1. *Kiểu dữ liệu :*

*PHP hỗ trợ 8 kiểu dữ liệu chính:*

- + 4 kiểu dữ liệu vô hướng: boolean, integer, float (double), string.
- + 2 kiểu dữ liệu tổ hợp: array, object.
- + 2 kiểu dữ liệu đặc biệt : resource, NULL

# NGÔN NGỮ LẬP TRÌNH PHP

## III. Lập trình PHP

### 1. Kiểu dữ liệu :

+ *Kiểu Boolean* : mang 1 trong 2 giá trị *TRUE* (đúng) hoặc *FALSE* (sai).

Ví dụ: <?php \$a = TRUE; \$b = FALSE;

//phép toán == kiểm tra xem 2 biểu thức

\$c = (7==2);

//vì 7 khác 2 nên \$c mang giá trị *FALSE*

\$d = ("abc" == "abc"); // \$d mang giá trị *TRUE* ?>

?>

# NGÔN NGỮ LẬP TRÌNH PHP

## III. Lập trình PHP

### 1. *Kiểu dữ liệu :*

+ *Kiểu Integer :*

- Mang các giá trị số nguyên ..., -2, -1, 0, 1, 2,...
- Trên hầu hết các hệ thống, kiểu số nguyên có kích thước 32 bit, mang giá trị từ -2147483647 cho đến 2147483648.

Ví dụ:

```
<?php $a = 1234; $b = -123; $c = 0123;  
?>
```

# NGÔN NGỮ LẬP TRÌNH PHP

## III. Lập trình PHP

### 1. Kiểu dữ liệu :

+ *Kiểu Float (Double)* : là kiểu số thực, có thể mang bất cứ giá trị số thực nào. Trên hầu hết các hệ thống, kiểu số thực có kích thước 64 bit.

Ví dụ: <?php

```
$a = 1.234;
```

```
$b = 1.2e3; //= 1.2*10^3 = 1200
```

```
$c = 7E-10; //= 7*(10^-10) = 0.0000000007
```

```
$d = -1.23;
```

```
?>
```

# NGÔN NGỮ LẬP TRÌNH PHP

## III. Lập trình PHP

### 1. *Kiểu dữ liệu :*

+ *Kiểu String* : lưu giữ một chuỗi ký tự, mỗi ký tự có kích thước 1 byte. Nội dung string được đặt giữa hai dấu nháy, nháy đơn ('') hoặc nháy kép ("").

Ví dụ: <?php

\$a = 'Đây là chuỗi được đặt giữa dấu nháy đơn';

\$b = "Đây là chuỗi được đặt giữa dấu nháy kép";

\$c = 'Đây là chuỗi được đặt giữa dấu nháy đơn với "vài dấu nháy kép ở giữa"';

\$d = "chuỗi đặt giữa dấu nháy kép với 'vài dấu nháy đơn ''";

# NGÔN NGỮ LẬP TRÌNH PHP

## III. Lập trình PHP

### 1. Kiểu dữ liệu :

+ Nếu muốn sử dụng dấu nháy đơn trong một chuỗi được bọc bởi dấu nháy đơn, hoặc sử dụng dấu nháy kép đặt giữa chuỗi được bọc bởi dấu nháy kép thì để thêm ký tự \ (ký tự escape) ở phía trước.

Ví dụ: <?php \$a = 'Dấu \'nháy đơn\' ở giữa chuỗi';

//\$a mang giá trị: Dấu 'nháy đơn' ở giữa chuỗi

\$b = "Dấu \"nháy kép\" ở giữa chuỗi";

//\$b mang giá trị: Dấu "nháy kép" ở giữa chuỗi

\$c = "Dùng ký tự \\ ở giữa câu \\ thì sao?";

//\$c mang giá trị: Dùng ký tự \ ở giữa câu \ thì sao?

?>

# NGÔN NGỮ LẬP TRÌNH PHP

## III. Lập trình PHP

### 1. *Kiểu dữ liệu :*

- Ngoài ra, nếu để một biến vào giữa một chuỗi được bọc với dấu nháy kép, giá trị của biến sẽ được thay thế vào trong chuỗi.

Ví dụ: <?php \$a = 1;

\$b = 2;

\$c = 3;

\$d = "\$a \$b \$c";

//\$d sẽ mang giá trị là chuỗi "1 2 3"

# NGÔN NGỮ LẬP TRÌNH PHP

## III. Lập trình PHP

### 1. *Kiểu dữ liệu :*

+ *Kiểu Array* : *Array là một mảng gồm nhiều phần tử. Array được tạo qua lệnh Array*

Ví dụ:

```
<?php $a = Array(1,2,3);  
      echo $a[0]; //in ra giá trị 1  
      echo $a[2]; //in ra giá trị 3  
      $a[1] = 5; //giờ đây $a = Array(1,5,3)  
?>
```

# NGÔN NGỮ LẬP TRÌNH PHP

## III. Lập trình PHP

### 1. *Kiểu dữ liệu :*

- *Mảng còn có thể được tạo thành bởi các cặp (khoá, giá trị).*

Ví dụ:

```
<?php $a = Array(  
    "khoá 1" => "giá trị 1",  
    "khoá 2" => "giá trị 2",  
    "khoá 3" => "giá trị 3");  
echo $a["khoá 1"]; //in ra: giá trị 1  
?  
?
```

# NGÔN NGỮ LẬP TRÌNH PHP

## III. Lập trình PHP

### 1. *Kiểu dữ liệu :*

+ *Kiểu NULL* : Đây là một giá trị đặc biệt, báo cho PHP biết rằng một biến nào đó chưa/không mang giá trị nào

Ví dụ: <?php \$a = 1; // \$a mang giá trị 1

\$a = NULL; // bây giờ \$a không mang giá trị

\$a = 2; // giờ đây \$a mang giá trị 2

// hàm unset sẽ làm cho 1 biến có giá trị là NULL

unset(\$a); // lúc này \$a lại là NULL

?>

# NGÔN NGỮ LẬP TRÌNH PHP

## III. Lập trình PHP

### 2. Biến trong PHP :

Một biến trong PHP được bắt đầu bằng ký tự \$ và đi theo ngay sau đó là tên của biến.

- + Biến trong PHP phân biệt chữ hoa và chữ thường. \$Abc và \$abc là hai biến hoàn toàn khác nhau.
- + Tên biến chỉ được bao gồm các ký tự chữ cái (a..z hoặc A...Z), chữ số (0...9) và ký tự gạch dưới (\_); nhưng tên biến không được bắt đầu bằng ký tự gạch dưới hoặc chữ số. Ví dụ: \$a: biến có tên là a  
\$abc123: biến có tên là abc123

# NGÔN NGỮ LẬP TRÌNH PHP

## III. Lập trình PHP

### 2. Biến trong PHP :

Ví dụ : Các tên biến sau là không hợp lệ :

`$_abc` Không hợp lệ! bắt đầu bằng ký tự gạch dưới

`$1abc` Không hợp lệ! bắt đầu bằng chữ số

`$nguyễn` Không hợp lệ! tên biến có ký tự đặc biệt (ê)

\* *Tầm vực (scope) của biến* : là ngữ cảnh mà ở trong đó biến được định nghĩa.

Ví dụ: <?php \$a = 1; //tầm vực của biến \$a từ đây  
include 'b.php'; //trải dài tới bên trong file b.php  
//tới cuối file vẫn còn hợp lệ  
?>

# NGÔN NGỮ LẬP TRÌNH PHP

## III. Lập trình PHP

### 3. Biểu thức :

+ Biểu thức là nền tảng quan trọng của PHP. Hầu như mọi thứ bạn ghi trong file php đều là biểu thức. Ta xét câu lệnh đơn giản sau:

$\$a = 5;$

$\$b = \$a;$

Tuy nhiên biểu thức trong PHP có thể phức tạp hơn

Ví dụ:       $\$a = 1; \$b = 2; \$c = 3; \$d = \$a + \$b + \$c;$

# NGÔN NGỮ LẬP TRÌNH PHP

## III. Lập trình PHP

### 4. Toán tử :

\* *Toán tử kết hợp các giá trị hoặc biểu thức lại với nhau và tạo ra một giá trị mới.*

Ví dụ: Trong biểu thức  $1+2$  thì “+” là toán tử kết hợp hai giá trị 1 và 2 lại với nhau tạo ra giá trị mới là 3.

\* Các toán tử trong PHP được chia thành 3 nhóm:

- Các toán tử áp dụng trên một giá trị, ví dụ như toán tử ++ hoặc --

- Các toán tử kết hợp hai hoặc nhiều giá trị, ví dụ như toán tử +, -, \*, /

- Toán tử ?: dùng để chọn 1 trong 2 giá trị

# NGÔN NGỮ LẬP TRÌNH PHP

## III. Lập trình PHP

### 4. Toán tử :

Toán tử	Sử dụng	Tương tự
<code>+=</code>	<code>\$a+=\$b</code>	<code>\$a*=\$a+\$b</code>
<code>++</code>	<code>\$a++</code>	<code>\$a*=\$a+1</code>
<code>-=</code>	<code>\$a-=\$b</code>	<code>\$a*=\$a-\$b</code>
<code>--</code>	<code>\$a- -</code>	<code>\$a*=\$a-1</code>
<code>*=</code>	<code>\$a*=\$b</code>	<code>\$a*=\$a*\$b</code>
<code>/=</code>	<code>\$a/= \$b</code>	<code>\$a*=\$a/\$b</code>
<code>%=</code>	<code>\$a%=\$b</code>	<code>\$a*=\$a%\$b</code>
<code>.=</code>	<code>\$a.=\$b</code>	<code>\$a*=\$a.\$b</code>

# NGÔN NGỮ LẬP TRÌNH PHP

## III. Lập trình PHP

### 5. Các phép toán cơ bản :

a. Các phép toán số học :

+, -, \*, /, % (module)

b. Các phép toán về chuỗi : Ta sử dụng dấu . để nối 2 chuỗi.

Ví dụ:

```
<?php $a= “Xin chao”; $b=“cac ban”;  
$a=$a . ”tat ca”;  
echo $a . $b;  
?>
```

# NGÔN NGỮ LẬP TRÌNH PHP

## III. Lập trình PHP

### 5. Các phép toán cơ bản :

#### c. Các phép toán so sánh :

Phép toán so sánh dùng để so sánh hai giá trị, nếu sử dụng để so sánh hai biểu thức, kết quả trả về là True hay False tùy vào kết quả so sánh.

Toán tử	Sử dụng	Tương tự
<code>==</code>	<code>\$a==\$b</code>	\$a bằng \$b
<code>====</code>	<code>\$a=====b</code>	\$a bằng và cùng kiểu \$b
<code>!=</code>	<code>\$a!=\$b</code>	\$a khác \$b
<code>&lt;&gt;</code>	<code>\$a&lt;&gt;\$b</code>	\$a khác \$b
<code>&lt;</code>	<code>\$a&lt;\$b</code>	\$a nhỏ hơn \$b
<code>&lt;=</code>	<code>\$a&lt;=\$b</code>	\$a nhỏ hơn hoặc bằng \$b
<code>&gt;</code>	<code>\$a&gt;\$b</code>	\$a lớn hơn \$b
<code>&gt;=</code>	<code>\$a&gt;=\$b</code>	\$a lớn hơn hoặc bằng \$b

# NGÔN NGỮ LẬP TRÌNH PHP

## III. Lập trình PHP

### 5. Các phép toán cơ bản :

#### d. Phép toán logic

!      Phép toán phủ định (not)

&&    Phép toán và (and)

||      Phép toán hoặc (or)

and    Phép toán và (and)

or      Phép toán hoặc (or).

#### e. Phép toán Error

Trong trường hợp biểu thức hay phép toán báo lỗi, nhưng ta không muốn trình duyệt hiển thị lỗi ta dùng phép toán @ trước biểu thức

# NGÔN NGỮ LẬP TRÌNH PHP

## III. Lập trình PHP

### 5. Các phép toán cơ bản :

#### e. Phép toán Error

Ví dụ:

```
<?php  
$a=10;  
$b=0;  
$c=@($a/$b);  
echo “Gia tri b = “.$c;  
?>
```

# NGÔN NGỮ LẬP TRÌNH PHP

## III. Lập trình PHP

### 6. Biến hàm

Là các thư viện hàm trong PHP cho phép thao tác và kiểm tra biến với nhiều cách khác nhau.

Các hàm sau trả về true nếu đúng, ngược lại là sai

- `is_array()` : Kiểm tra biến có phải kiểu mảng không
- `is_double()` : Kiểm tra biến có phải kiểu double không
- `is_float()` : Kiểm tra biến có phải kiểu float không
- `is_integer()` : Kiểm tra biến có phải kiểu int không
- `is_string()` : Kiểm tra biến có phải kiểu chuỗi không
- `is_object()` : Kiểm tra biến có phải kiểu đối tượng không

# CẤU TRÚC ĐIỀU KHIỂN

## I. Cấu trúc lựa chọn :

### 1. Câu lệnh *if*

Cú pháp đơn giản nhất của câu lệnh if có dạng như sau:

**if ( biểu thức ) câu lệnh;**

Câu lệnh if trên được diễn giải như sau: nếu biểu thức trả về giá trị TRUE (hoặc tương đương với TRUE sau khi chuyển đổi) thì câu lệnh sẽ được thực thi; ngược lại (khi biểu thức trả về giá trị FALSE) thì bỏ qua không thực thi câu lệnh nữa.

Cú pháp nâng cao của câu lệnh if có dạng như sau:

```
if ( biểu thức ) câu lệnh 1;  
else    câu lệnh 2;
```

Câu lệnh if trên được diễn giải như sau: nếu **biểu thức** trả về giá trị TRUE thì **câu lệnh 1** sẽ được thi hành, ngược lại thì **câu lệnh 2** sẽ được thi hành.

Các lệnh if có thể được lồng vào nhau để tạo ra câu lệnh if phức tạp hơn:

```
if( biểu thức 1 )
    if( biểu thức 2 ) câu lệnh 1;
    else câu lệnh 2;
else câu lệnh 3;
```

Nếu biểu thức 1 trả về giá trị FALSE thì câu lệnh 3 sẽ được thực hiện, ngược lại xét tiếp biểu thức 2: nếu biểu thức 2 trả về giá trị TRUE thì thực hiện câu lệnh 1, ngược lại nếu biểu thức 2 trả về giá trị FALSE thì thực hiện câu lệnh 2.

```
if ( biểu thức 1 ) câu lệnh 1;  
else if ( biểu thức 2 ) câu lệnh 2;  
else if ( biểu thức 3 ) câu lệnh 3;  
else câu lệnh 4;
```

Nếu **biểu thức 1** trả về TRUE thì thực hiện **câu lệnh 1** (các câu lệnh 2,3,4 không thực hiện), nếu **biểu thức 1** trả về FALSE và **biểu thức 2** trả về TRUE thì **câu lệnh 2** được thực hiện, nếu **biểu thức 1** trả về FALSE **biểu thức 2** trả về FALSE và **biểu thức 3** trả về TRUE thì **câu lệnh 3** được thực hiện. Nếu cả 3 biểu thức 1,2,3 đều trả về FALSE thì thực hiện **câu lệnh 4**.

Ngoài ra PHP còn cung cấp từ khoá **elseif**, chính là ghép giữa từ khoá **else** và **if**.

## 2. Câu lệnh switch

Cú pháp :

```
switch ($bt) {  
    case h1: các lệnh1;  
    break;  
    case h2: các lệnh 2;  
    break;  
    .....  
    case hn: các lệnh n;  
    break;  
    default : các lệnh n+1;  
}
```

## 2. Câu lệnh switch

```
<?php
switch ($x) {
case 1: echo "Number 1";
          break;
case 2: echo "Number 2";
          break;
case 3: echo "Number 3";
          break;
default : echo "No number between 1 and 3";
}
?>
```

## 2. Câu lệnh switch

Câu lệnh switch hoạt động như là một loạt câu lệnh if ghép lại với nhau. Ta hãy xem câu lệnh if sau:

```
if ( $a == "abc" )
    {echo "Giá trị của a là abc"; }
else if ( $a == "def" )
    {echo "Giá trị của a là def"; }
else if ( $a == "123" )
    { echo "Giá trị của a là 123"; }
else
    { echo "Giá trị khác"; }
```

Có thể được viết lại bằng câu lệnh switch như sau:

```
switch ( $a )
{
    case "abc": echo "Giá trị của a là abc";
                  break;
    case "def": echo "Giá trị của a là def";
                  break;
    case "123": echo "Giá trị của a là 123";
                  break;
    default: echo "Giá trị khác";
}
//end switch
```

# Cấu trúc điều khiển

## II. Cấu trúc lệnh lặp:

### 1. Câu lệnh *while* :

*while ( biểu thức ) câu lệnh;*

Trong khi biểu thức còn trả về giá trị TRUE thì tiếp tục thực hiện câu lệnh, sau khi thực hiện câu lệnh thì kiểm tra lại biểu thức, nếu vẫn còn trả về giá trị TRUE thì lại tiếp tục thực hiện câu lệnh...cứ tiếp tục như vậy cho tới khi nào biểu thức trả về giá trị FALSE thì ngưng.

Ví dụ: In ra các số từ 1 tới 10 với câu lệnh while:

```
<?php $i = 1;  
while ( $i <= 10 )  
{ echo $i, "<br>";  
 $i++;  
}  
//end while  
?>
```

Ghi chú: Câu lệnh **\$i++** tương đương với **\$i = \$i+1;**, câu lệnh này sẽ tăng giá trị của \$i lên 1 qua mỗi lần lặp.

## 2. Câu lệnh *do-while*

Câu lệnh do-while cũng tương tự như câu lệnh while, chỉ khác một điểm là câu lệnh được thực hiện trước rồi biểu thức mới được kiểm tra sau, nếu biểu thức còn trả về giá trị TRUE thì tiếp tục thực hiện câu lệnh.

Cú pháp của câu lệnh do-while như sau:

*do*

{ *câu lệnh*;

}

*while ( biểu thức );*

*Ví dụ:*

In ra các số từ 1 tới 10 với câu lệnh **do-while**:

```
<?php $i =0;  
do  
{  
    $i++;  
    echo $i, “<br>”;  
} while ( $i <10 );  
?>
```

### 3. Câu lệnh for

Câu lệnh for cũng dùng để tạo vòng lặp. Đây là một trong những câu lệnh phức tạp nhất của PHP, cú pháp của nó như sau:

**for ( biểu thức 1; biểu thức 2; biểu thức 3 )  
    câu lệnh;**

Đầu tiên biểu thức 1 được thực hiện, Tiếp theo biểu thức 2 được kiểm tra Nếu trả về TRUE thì câu lệnh được thực hiện và sau đó thực hiện biểu thức 3. Nếu trả về FALSE thì kết thúc câu lệnh for.Kiểm tra lại biểu thức 2 và lặp lại quá trình như trên.

## *Ví dụ:*

In ra các số từ 1 tới 10 với câu lệnh for:

```
<?php for ( $i = 1; $i < 10; $i++ )  
    { echo $i, “<br>”; }  
//end for  
?>
```

## 4. Câu lệnh *foreach*

Câu lệnh foreach chỉ làm việc với array. Câu lệnh foreach có 2 dạng cú pháp như sau:

*foreach ( \$array as \$value ) câu lệnh;*

*foreach ( \$array as \$key => \$value ) câu lệnh;*

*Ví dụ 1:*

```
<?php  
$a = array('a' => '1', 'b' => '2', 'c' => '3');  
foreach ( $a as $value )  
{ echo $value, "\n"; }  
//end foreach  
?>
```

Chương trình trên sẽ in ra 3 số 1, 2 và 3.

## Ví dụ 2:

```
<?php  
$a = array('a' => '1', 'b' => '2', 'c' => '3');  
foreach ( $a as $key => $value )  
    { echo $key, "=", $value, "\n"; }  
//end foreach  
?>
```

Chương trình trên sẽ in ra 3 chuỗi **a=1**, **b=2** và **c=3**.

## 5. Câu lệnh ***break***

Câu lệnh break sẽ dừng việc thực thi của các vòng lặp for, foreach, while, do-while và switch. Ở phần trước ta đã thấy câu lệnh break được sử dụng trong câu lệnh switch.

Ví dụ: sử dụng câu lệnh break trong vòng lặp for:

```
for ( $i=1; $i<=10; $i++ )  
{ echo $i;  
if ( $i == 5 ) break;  
}
```

## 6. Câu lệnh continue

Câu lệnh continue áp dụng lên các vòng lặp, lệnh continue sẽ bỏ qua lần lặp hiện thời và tiếp tục thực hiện các lần lặp tiếp theo.

Ví dụ:

```
for ( $i=1; $i<=5; $i++ )  
{ if ( $i == 2 ) continue;  
echo $i; }
```

Khi \$i đạt giá trị 2, câu lệnh echo \$i; sẽ được bỏ qua không thi hành nữa do câu lệnh continue. Và như vậy, đoạn lệnh trên khi chạy sẽ in ra các giá trị 1,3,4,5 (không có giá trị 2).

# HÀM

## 1. Định nghĩa :

Trong lập trình, có một số đoạn mã được dùng nhiều lần ở nhiều nơi khác nhau trong chương trình. PHP cung cấp một giải pháp đó là hàm do người dùng định nghĩa.

Ta có thể đưa đoạn mã đó vào trong một hàm, và ở chỗ nào cần dùng đoạn mã đó ta chỉ cần gọi hàm, khi cần sửa đổi, ta chỉ cần sửa đổi một chỗ duy nhất là nội dung của hàm chứ không cần phải sửa ở nhiều nơi trong chương trình.

Cú pháp để tạo hàm như sau:

```
function tênHàm($tham_số1, $tham_số2,...)  
{  
    cac lệnh trong thân hàm ;  
    return $kết_quả_trả_về;  
}  
//end
```

Khi cần sử dụng hàm ở chỗ nào, ta chỉ cần dùng cú pháp :  
**tênHàm(các tham số cần thiết);**

*Ví dụ:*

```
<?php  
function testing()  
{ echo "Testing function";  
echo 1;  
echo 2;  
echo 3;  
} //end testing  
testing(); testing(); testing();  
?>
```

Tên hàm cũng như tên biến , nhưng được phép bắt đầu bằng ký tự gạch dưới .

## *Tham số của hàm*

Hàm có thể nhận vào các tham số, ví dụ:

```
<?php  
function testing($a)  
{ echo "Tham số là $a"; }  
//end testing testing(123);  
testing("abc");  
?>
```

Ở ví dụ trên, lời gọi hàm testing(123); sẽ in ra dòng Tham số là 123 và lời gọi hàm testing("abc"); sẽ in ra dòng Tham số là abc.

Ta có thể gán giá trị mặc định cho tham số của hàm:

```
<?php  
    function testing($a="mặc định")  
    { echo "Tham số là ". $a;  
    } //end  
testing();  
?>
```

Khi tham số tương ứng của hàm không được truyền, tham số đó sẽ nhận giá trị mặc định. Đoạn chương trình ví dụ ở trên khi chạy sẽ in ra dòng Tham số là mặc định.

## Giá trị trả về từ hàm

Hàm còn thể trả về 1 giá trị cho nơi gọi:

```
<?php  
function binh_phuong($a)  
{      $ketqua = $a * $a;  
      return $ketqua;  
} //end testing  
echo binh_phuong(2);  
?>
```

Đoạn chương trình trên khi chạy sẽ in ra số 4. Câu lệnh return biểu\_thúc; sẽ kết thúc hàm và trả về giá trị của biểu\_thúc cho nơi gọi.

Chú ý: bên trong hàm, biến cục bộ sẽ được dùng thay vì biến toàn cục.

Ví dụ:

```
<?php  
$a = 1; //biến toàn cục //hàm do tự tạo  
function test()  
    { echo $a; }  
//end test  
?>
```

Để truy cập tới các biến toàn cục ở bên trong một hàm do người dùng định nghĩa, ta có thể dùng một trong hai cách sau:

### Cách 1:

```
<?php $a = 1; //biến toàn cục  
//hàm do tự tạo  
function test()  
{ //tùy khoá global báo cho php biết là bên test  
//bây giờ ta sẽ dùng biến toàn cục  
global $a;  
echo $a; //in ra giá trị: 1  
} //end test
```

## Cách 2:

```
<?php  
$a = 1; //biến toàn cục  
//hàm do tự tạo  
function test()  
{ echo $GLOBALS['a']; //in ra giá trị: 1  
} //end test  
?>
```

## 2. Các hàm thông dụng của PHP

### a. Hàm xem thông tin `phpinfo()`

Các tham số:

**INFO\_GENERAL**

**INFO\_CREDITS**

**INFO\_CONFIGURATION**

**INFO\_MODULES**

**INFO\_ENVIRONMENT**

**INFO\_VARIABLES**

**INFO\_LICENSE**

**INFO\_ALL**

```
<html>
<body>
    <?php
        phpinfo();
    ?>
    <?php
        phpinfo(INFO_GENERAL);
    ?>
</body>
</html>
```

## b. Hàm hiển thị thông tin Server

```
<?php  
    echo "Hỗ trợ: " . $_SERVER["HTTP_REFERER"]  
    . "<br />";  
    echo "Hỗ trợ Browser: " .  
        $_SERVER["HTTP_USER_AGENT"] . "<br />";  
    echo "Địa chỉ IP: " .  
        $_SERVER["REMOTE_ADDR"];  
?>
```

## c. Hàm Header()

Chuyển đến một trang nào đó

```
<?php  
header("Location: http://www.ud.edu.vn/");  
?  
<html> <body>  
.....</body> </html>
```

Lưu ý: Khi dùng hàm này cần:

- \* Phải đảm bảo các lệnh trước đó không có tác dụng xuất bất cứ dữ liệu nào ra trình duyệt thì nó mới có tác dụng (*kể cả trước đó là các thẻ HTML*)

### **3. Truy c $\ddot{e}$ p bi $\tilde{O}$ n t $\ddot{o}$ form**

Trong PHP khai b $\ddot{o}$   $\textcircled{R}$ Ó l $\hat{E}$ y gi $\acute{c}$ , tr $\textcircled{B}$  bi $\tilde{O}$ n FORM t $\ddot{o}$  trang HTML t $\textcircled{n}$ g  $\textcircled{R}$ èi  $\textcircled{R}$ n g $\acute{i}$ nh, t $\textcircled{a}$ n c $\tilde{a}$ n m $\acute{e}$ t  $\textcircled{R}$ èi t $\tilde{a}$ ng Form trong HTML c $\acute{o}$ ng ch $\acute{Y}$ nh l $\mu$  t $\textcircled{a}$ n bi $\tilde{O}$ n d $\ddot{e}$ ng trong PHP.

VD:

```
<?php echo "ten cua ban la: ".$txtUser;  
      echo "Mat khau cua ban la: ".$txtPass;  
?>
```

Trong  $\textcircled{R}$ ã txtUser v $\mu$  txtPass l $\mu$  t $\textcircled{a}$ n  $\textcircled{R}$ èi t $\tilde{a}$ ng d $\ddot{e}$ ng text trong thi FORM.

Sö dông m¶ng h»ng

```
$HTTP_POST_VARS["t n bi n form"];
```

```
$HTTP_GET_VARS["t n bi n form"];
```

VD:<?php

```
echo "ten cua ban la: ". $HTTP_POST_VARS ["txtUser"];  
?>
```

Trong ® a txtUser l u t n ® i t ng d ng text trong thi FORM.

#### 4. Khai bõ h»ng trong PHP

- + TÊt c¶ c,c h»ng sè trong PHP lµ ch÷ hoa
- + Có ph,p khai b,o:

**define("TENHANG",gi,trø g,n h»ng)**

VÝ dô:

**define("PI",3.14);**

## 5. Sö dōng lại m· ch¬ng tr×nh trong PHP

PHP cung cÊp 2 c¸ch ®Ó khai b,o sö dōng bÊt kú ®o¹n m· ch¬ng tr×nh nµo ®· cã s½n, b»ng c¸ch sö dōng 2 khai b,o  
*require()* vµ *include()*.

### a. Sö dōng khai b,o *require()*:

Có ph,p: *require(string path\_file)*;

VÝdô : T¹o 1 file require1.php cã néi dung nhÚ sau

```
<?php  
$a=2;  
$b=3;  
$c=$a+$b;  
echo "a + b = ".$c."<br>";  
?>
```

- t<sup>1</sup>o m $\hat{0}$ t file php cã néi dung

<?php

*require("require1.php");*

?>

## Các lưu ý:

- File  $\hat{0}$ c gäi cã ph $\hat{C}$ n m $\acute{e}$  r $\acute{e}$ ng t $\acute{i}$ y y.

- C,c  $\hat{0}$ o<sup>1</sup>n m $\cdot$  PHP trong File  $\hat{0}$ c gäi ph $\ddot{U}$ i  $\hat{R}$ Æt trong c $\acute{a}$ p th $\hat{i}$   
<?php ...?>

## b. Sử dụng khai báo *include()*:

Có phím: *include(string path\_file);*

Gièng nhì trêng hîp trâng, nhìng include() vµ *require()* kh, c nhau lµ: TÊt c¶ néi dung ®îc n¹p ln b»ng *require()*

Webserver ph¶i dÞch tÊt c¶ néi dung bn trong. Do ®ã trong nh÷ng file n¹p b»ng *require()* kh«ng nn dng c,c cÊu tróc ®iÒu khiÓn.

# CÁC HÀM XỬ LÍ CHUỖI TRONG PHP

Xo lý chuỗi trong ứng dụng Web lµ mét phÇn chÝnh cña PHP, ®iÓm m¹nh cña PHP lµ kh¶ n°ng xo lý chuỗi.

## 1. Hµm xo lý kho¶ng tr¶ng trªn chuỗi:

+Hµm kiÓm tra chuỗi r¶ng:

Có ph,p: *isempty(string str);*

NÔu chuỗi str trçng tr× tr¶ vÒ true, ngîc l¹i tr¶ vÒ false.

+Hµm xää bá kho¶ng tr¶ng bªn tr,i vµ bªn ph¶i chuỗi  
vµ: *chop(string str);*

VÝ dô:

*\$name=ltrim(\$name);*

*\$pass=chop(\$pass);*

Có ph,p: *ltrim(string str);*

## 2. Hỗn hợp thay @æi kiÔu ch÷ cñã chuçi:

Hỗn	ĐiÔn gi¶i	Sö dông	KÕt qu¶
<b>strupper</b>	Ch÷ hoa	<b>strupper</b> ("toi")	TOI
<b>strlower</b>	Ch÷ thêng	<b>strlower</b> ("TOI")	toi
<b>ucfirst</b>	Ch÷ hoa ký tù ®Çu tiªn	<b>ucfirst</b> ("toi va ban")	Toi va ban
<b>ucword</b>	Ch÷ hoa ký tù ®Çu mci tõ	<b>ucword</b> ("toi va ban")	Toi Va Ban

### 3. §Þnh dñng chuçi ®Ó ®a vµo vµ lÊy ra tõ CSDL:

§Ó lu trÙ mÔt chuçi vµo CSDL mµ chuçi ®ã cho chøa c,c ký tù ' hay " hay \ thx ph¶i th m dÊu \ vµo tríc khi lu.

Ta sö dñng hµm *AddSlashes(string str)*.

Hµm nµy sї tù t m c,c ký tù tr n vµ th m \ vµo tríc.

Ng c l i ta d ng hµm :

*StripSlashes(string str)* ®Ó lo i b  \.

VÝ dô: \$noidung=AddSlashes(\$noidung);

\$noidung=StripSlashes(\$noidung);

#### 4. Hຸm kÕt hີp hay tັch chuçi:

+ Tັch chuçi thູnh 2 chuçi con b»ng mét ký tù ph©n c, ch vµ ®a vµo m¶ng.

Có ph,p:

*array=explode(string phan\_cach, string str);*

VÝ dô : tັch ®pa chØ Email thູnh 2 phçn:

*\$manga=explode("@",\$email);*

*echo manga[1];*

+ Hຸm substr();

Có ph,p:

*string substr(string str, int i [,int l]);*

Hຸm nµy tr¶ vÒ chuçi con cña chuçi **str**, b¾t ®Çu tõ vµ trÝ i víi l ký tù. Tham sè l lµ tiý chän, nÕu kh«ng cã thx tr¶ vÒ ®Õn hÕt chuçi **str**

## 5. H $\ddot{u}$ m so s $\acute{u}$ nh chu $\acute{u}$ i:

Ngo $\ddot{u}$ i to $\acute{n}$  t $\ddot{o}$  == d $\ddot{e}$ ng so s $\acute{u}$ nh 2 chu $\acute{u}$ i, ch $\ddot{o}$ ng ta c $\acute{B}$ n c $\acute{a}$  th $\acute{O}$  d $\ddot{e}$ ng c $\acute{c}$ c h $\ddot{u}$ m :

***int strcmp(string str1, string str2);***

N $\tilde{O}$ u str1= str2 : =0

str1>str2 : >0

str1<str2 : <0

So s $\acute{u}$ nh str1 v $\mu$  str2 c $\acute{a}$  ph $\acute{O}$ n bi $\acute{O}$ t ch $\acute{u}$  hoa, ch $\acute{u}$  thêng, ngo $\ddot{u}$ i ra c $\acute{B}$ n c $\acute{a}$  c $\acute{c}$  h $\ddot{u}$ m

***int strcasecmp(string str1, string str2);***

***int strnatcmp(string str1, string str2);***

## 6. Hμm kiÓm tra chiÒu dui chuçi:

có ph,p:

*int strlen(string str);*

## 7. Hμm t×m kiÓm vµ thay thÕ chuçi

+ Hμm t×m chuçi:

*string strstr(string str1, string str2);*

Hμm nµy t×m chuçi str2 trong str1, nÕu cã tr¶ vÒ chuçi mïi lµ chuçi b¾t ®Çu b»ng chuçi str2 ®Õn hÕt chuçi str1.

VÝ dô:

<?php

*\$a="xin chao cac ban than men";*

*echo strstr(\$a, "c"). "<br>";*

?>

+ Hμm t×m vB trÝ chuçi con:

*int strpos(string str1, string str2);*

Hμm nμy tr¶ vÒ vB trÝ t×m thÊy cña chuçi str2 trong str1, nÕu kh«ng t×m thÊy th× tr¶ vÒ -1, nÕu cã nhiÒu chuçi giÙng nhau, th× chØ tr¶ vÒ vB trÝ t×m thÊy ®Çu tiªn.

(Vb trÝ cña chuçi tÝnh tõ 0)

VÝ dô:

<?php

**\$a="xin chao cac ban than men";**

**\$vitri=strpos(\$a,"chao");**

**echo "vi tri la: ".\$vitri.<br>;**

?>

+ Hμm thay thÓ chuçi:

*string str\_replace(string str1, string str2, string str);*

- . Hμm nμy tr¶ vÒ 1 chuçi b»ng c, ch thay thÓ tÊt c¶ c,c chuçi str1 trong str b»ng chuçi str2.
- . §Ó khèng chÓ chØ cho thay ®æi vÞ trÝ ta dïng hμm:

*string substr\_replace(string str1, string str2, int start [,int n]);*

Thay the str2 vµo trong str1, bat ®Çu tõ ký tù cã vÞ trÝ lµ start vµ cã chiÒu dµl lµ n ký tù VÝ dô:

<?php

**\$a="xin chao cac ban va tat ca cac ban than men";**

**\$vitri=strpos(\$a, "chao");**

**\$chuoi=str\_replace("ban","em",\$a);**

**\$chuoi2=substr\_replace(\$chuoi,"ban",13,3);**

**echo "chuoi: ".\$chuoi."<br>";**

**echo "chuoi 2 ".\$chuoi2."<br>";**

?>

# HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU MySQL

# HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU MYSQL

## I. Truy xuất cơ sở dữ liệu bằng PHP

### 1. Cơ sở dữ liệu MySQL:

- + Để có thể liên kết và truy xuất dữ liệu trong PHP, trước tiên ta cần tìm hiểu cách thao tác làm việc trên cơ sở dữ liệu trong MySQL.
- + MySQL là một hệ quản trị cơ sở dữ liệu vừa và nhỏ.
- + Khi làm việc với cơ sở dữ liệu MySQL, bạn *đăng ký kết nối, tạo cơ sở dữ liệu, quản lý người dùng, phân quyền sử dụng, thiết kế đối tượng Table của cơ sở dữ liệu và xử lý dữ liệu.*

# HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU MYSQL

## I. Truy xuất cơ sở dữ liệu bằng PHP

### 1. Cơ sở dữ liệu MySQL:

- + Nếu lần đầu tiên tạo kết nối cơ sở dữ liệu, chúng ta cần phải tạo một Connection, cung cấp tên Server hay IP của máy chứa MySQL.
- + Trong trường hợp máy chứa cơ sở dữ liệu MySQL là máy đang sử dụng, bạn có thể sử dụng tên **localhost**. Username mặc định của cơ sở dữ liệu MySQL là **root** và Password là **root**.
- + Cuối cùng bạn bắt đầu tạo cơ sở dữ liệu và làm việc với cơ sở dữ liệu.

# HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU MYSQL

## I. Truy xuất cơ sở dữ liệu bằng PHP

### 2. *Truy xuất cơ sở dữ liệu trong PHP:*

a. **Thiết lập kết nối cơ sở dữ liệu:** PHP cung cấp cho chúng ta hàm kết nối cơ sở dữ liệu có cú pháp:

```
int mysql_pconnect("servername","username","password")
```

Ngoài ra có thể sử dụng phát biểu như sau:

```
$link=mysql_connect("servername","username","password") or die("Could not connect to MySQL Database");
```

# HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU MYSQL

## I. Truy xuất cơ sở dữ liệu bằng PHP

### 2. *Truy xuất cơ sở dữ liệu trong PHP:*

#### b. Mở một cơ sở dữ liệu :

- + Sau khi khai báo kết nối đến cơ sở dữ liệu MySQL thành công, bạn khai báo phát biểu để mở cơ sở dữ liệu.
- + Nếu chọn cách 1 thì ta khai báo:
  - `mysql_select_db("tên CSDL");`
- + Nếu chọn cách 2 thi ta khai báo:
  - `mysql_select_db("tên CSDL",$link);`

# HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU MYSQL

## I. Truy xuất cơ sở dữ liệu bằng PHP

### 2. *Truy xuất cơ sở dữ liệu trong PHP:*

Ví dụ: Trong CSDL MySQL có cơ sở dữ liệu có tên là **DULIEU**. Để thực hiện kết nối đến cơ sở dữ liệu này ta khai báo đoạn lệnh sau:

```
<?php  
    //Khai báo kết nối  
    $link = mysql_connect("localhost", "root", "") or die ("Khong the  
    ket noi den CSDL MySQL");  
  
    //Lựa chọn cơ sở dữ liệu  
    mysql_select_db("DULIEU", $link);  
?>
```

# HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU MYSQL

## I. Truy xuất cơ sở dữ liệu bằng PHP

2. *Truy xuất cơ sở dữ liệu trong PHP:*

c. *Đóng kết nối dữ liệu :*

+ Khi không sử dụng kết nối cơ sở dữ liệu, bạn có thể khai báo phát biểu đóng kết nối như sau:

***mysql\_close(database\_connection);***

Ví dụ: Đóng kết nối cơ sở dữ liệu trên:

***mysql\_close(\$link);***

# HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU MYSQL

## I. Truy xuất cơ sở dữ liệu bằng PHP

2. *Truy xuất cơ sở dữ liệu trong PHP:*

*d. Truy vấn cơ sở dữ liệu:*

+ Để truy vấn cơ sở dữ liệu ta có thể dùng hàm `mysql_query()` với tham số bắt buộc là chuỗi SQL, chuỗi kết nối là tham số tùy chọn như một trong ba phát biểu sau:

`int mysql_query( string sql);`

`int mysql_query( string sql, [int db_connection]);`

`int mysql_query( string database, string sql, [int db_connection]);`

# HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU MYSQL

## I. Truy xuất cơ sở dữ liệu bằng PHP

### 2. Truy xuất cơ sở dữ liệu trong PHP:

```
<?php  
//Khai báo kết nối  
$link = mysql_connect("localhost", "root", "") or die  
("Khong the ket noi den CSDL MySQL");  
//Lựa chọn cơ sở dữ liệu  
mysql_select_db("DULIEU", $link);  
$sql = "Select * from TblUser";  
$result = mysql_query($sql, $link);  
?>
```

# HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU MYSQL

## I. Truy xuất cơ sở dữ liệu bằng PHP

### 2. Truy xuất cơ sở dữ liệu trong PHP:

+ Để biết được số lượng mẫu tin trả về ta có thể dùng hàm :

*int mysql\_num\_rows();*

+ Bạn có thể đọc từng mẫu tin bằng các hàm sau:

*mysql\_fetch\_array();*

*mysql\_fetch\_assoc();*

```
<?php  
  
$link = mysql_connect("localhost", "user", password);  
mysql_select_db("database", $link);  
  
$result = mysql_query("SELECT * FROM table1", $link);  
$num_rows = mysql_num_rows($result);  
  
echo "$num_rows Rows\n";  
  
?>
```

```
$row = mysql_fetch_array($result);  
$row = mysql_fetch_assoc($result);
```

Kết quả trả về từ hai hàm này là một **array**. Hai hàm này trả về dòng kết quả hiện thời của câu lệnh select và chuyển con trỏ tới dòng tiếp theo hoặc giá trị FALSE nếu như không còn dòng nào để trả về nữa.

**mysql\_error(\$conn);**

Trả về thông báo lỗi của MySQL nếu như một lệnh trước đó có lỗi.

## *Sự khác nhau của mysql\_fetch\_array() và mysql\_fetch\_assoc()*

+ Hàm **mysql\_fetch\_array()** sẽ trả về một array mà phần tử thứ [0] sẽ tương ứng với cột đầu tiên của table, phần tử thứ [1] sẽ tương ứng với cột thứ hai của table,..

Ví dụ : while ( \$row = mysql\_fetch\_array(\$result) )

```
{ echo "Username = ".$row[0]."<br>\n";
echo "Password = ".$row[1]."<br>\n"; }
```

+ Hàm **mysql\_fetch\_assoc()** sẽ trả về một array mà các phần tử sẽ được truy cập qua tên, với tên được lấy từ tên các cột của table

Ví dụ : while ( \$row = mysql\_fetch\_assoc(\$result) )

```
{ echo "Username = ".$row['username']. "<br>\n";
echo "Password = ".$row['password']. "<br>\n"; }
```

# THỰC THI CÂU LỆNH SELECT

```
<?php
$result = mysql_query("SELECT id,email
FROM people WHERE id = '42' ");
if (!$result) {
    echo 'Could not run query: ' . mysql_error();
    exit;
}
$row = mysql_fetch_row($result);

echo $row[0]; // 42
echo $row[1]; // the email value
?>
```

```
.....
```

```
.....
```

```
while ($row = mysql_fetch_assoc($result)) {  
    echo $row["userid"];  
    echo $row["fullname"];  
    echo $row["userstatus"];  
}  
mysql_free_result($result);
```

```
?>
```

# THỰC THI CÂU LỆNH INSERT

```
<?php
$link = mysql_connect('localhost', 'user', 'password');
if (!$link)
{
    die('Could not connect: ' . mysql_error());
}
mysql_select_db('mydb',$link);
$sql = "INSERT INTO thanhvien (id, user, pass, email) VALUES (1,
    'admin','123')";
$result = mysql_query($sql, $link);
//chèn thêm 1 dòng vào table
if ( !$result ) die("Không thể thực hiện được câu lệnh
    SQL:".mysql_error($link));
echo "Số lượng row được chèn: ".mysql_affected_rows($link)."<br>\n";
//ta chỉ chèn 1 dòng nên hàm mysql_affected_rows sẽ trả về 1
?>
```

# THỰC THI CÂU LỆNH UPDATE

```
<?php
$link = mysql_connect('localhost', 'user','password');
if (!$link)
{
    die('Could not connect: ' . mysql_error());
}
mysql_select_db('mydb',$link);
$sql = "UPDATE thanhvien SET pass='abc' WHERE username='admin'";
$result = mysql_query($sql, $link);
//đổi password của admin
if ( !$result ) die("Không thể thực hiện được câu lệnh SQL:
".mysql_error($link));
echo "Số lượng row được thay đổi:
".mysql_affected_rows($link)."<br>\n";
?>
```

# THỰC THI CÂU LỆNH DELETE

```
<?php  
$link = mysql_connect('localhost', 'user', 'password');  
if (!$link)  
{  
    die('Could not connect: ' . mysql_error());  
}  
  
mysql_select_db('mydb',$link);  
$sql = "DELETE * FROM thanhvien";  
$result = mysql_query($sql, $link);  
//xoá hết tất cả các account  
if ( !$result ) die("Không thể thực hiện được câu lệnh SQL:  
    ".mysql_error($link));  
echo "Số lượng row được xoá: ".mysql_affected_rows($conn)."  
?>
```

# HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU MYSQL

## II. CÀI ĐẶT PHP, MYSQL, APACHE

### 1. Cài đặt Apache trên Window:

#### + Kiểm tra sau khi cài đặt:

- Sau khi khởi động Apache nó sẽ lắng nghe trên Port 80.

- Để kiểm tra Apache server có hoạt động không bạn kết nối đến Apache web server và truy cập nhập địa chỉ <http://localhost/> hoặc <http://127.0.0.1>

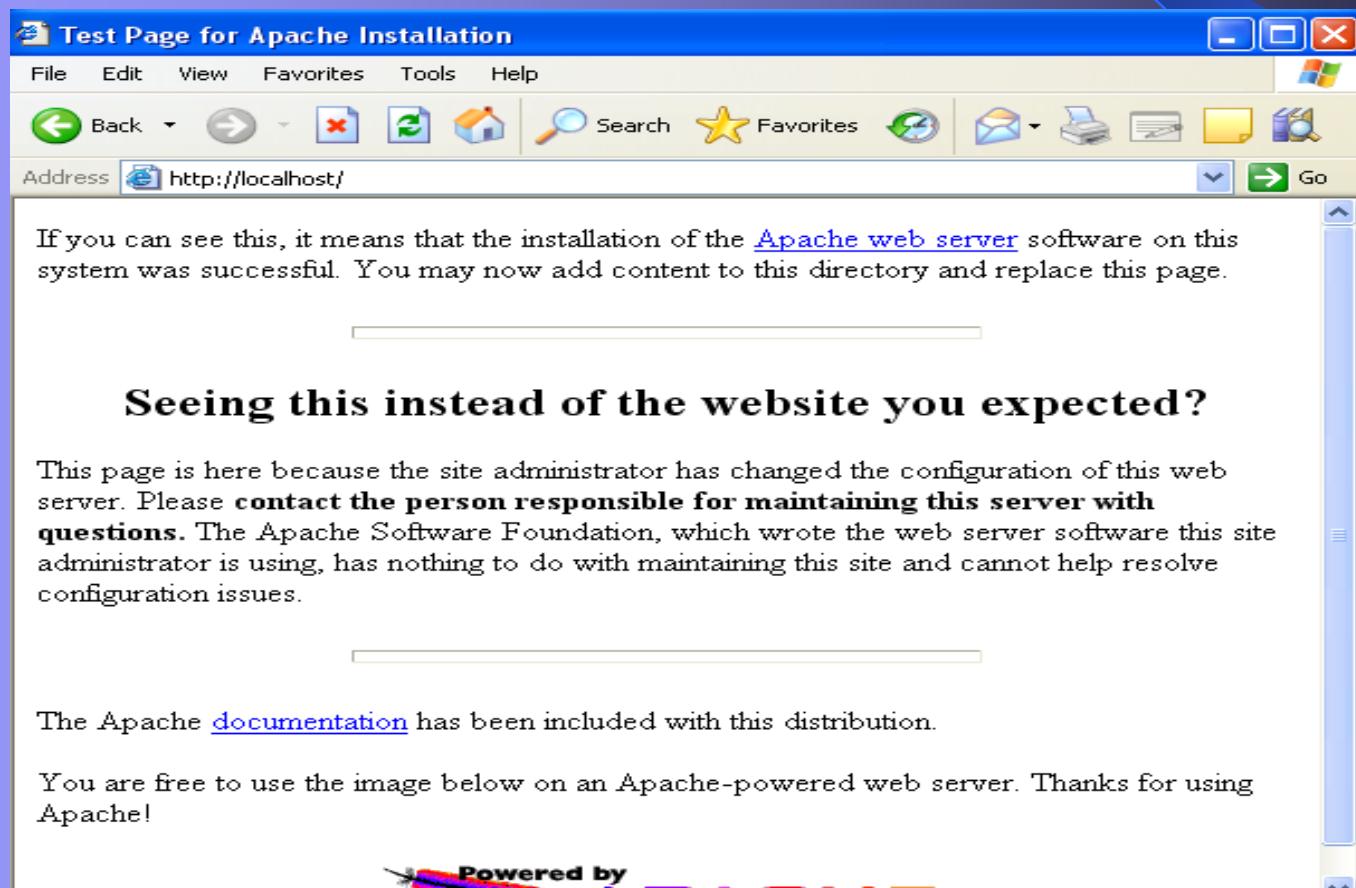
- Nếu Apache Web server hoạt động được thì nó sẽ hiển thị trang nào chứa các liên kết đến tài liệu apache như hình sau:

# HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU MYSQL

## II. CÀI ĐẶT PHP, MYSQL, APACHE

### 1. Cài đặt Apache trên Window:

+ Kiểm tra sau khi cài đặt :



# HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU MYSQL

## II. CÀI ĐẶT PHP, MYSQL, APACHE

### 2. Cài đặt MySQL :

+ Nhấp đúp chuột lên tập tin cài đặt Setup.exe màn hình chào MySQL Server and Client xuất hiện như hình sau. Bấm Next để tiếp tục :



# HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU MYSQL

## II. CÀI ĐẶT PHP, MYSQL, APACHE

### 2. Cài đặt MySQL :

+ Đợi cho đến khi quá trình cài đặt kết thúc.  
Bấm nút Finish để kết thúc quá trình cài đặt.

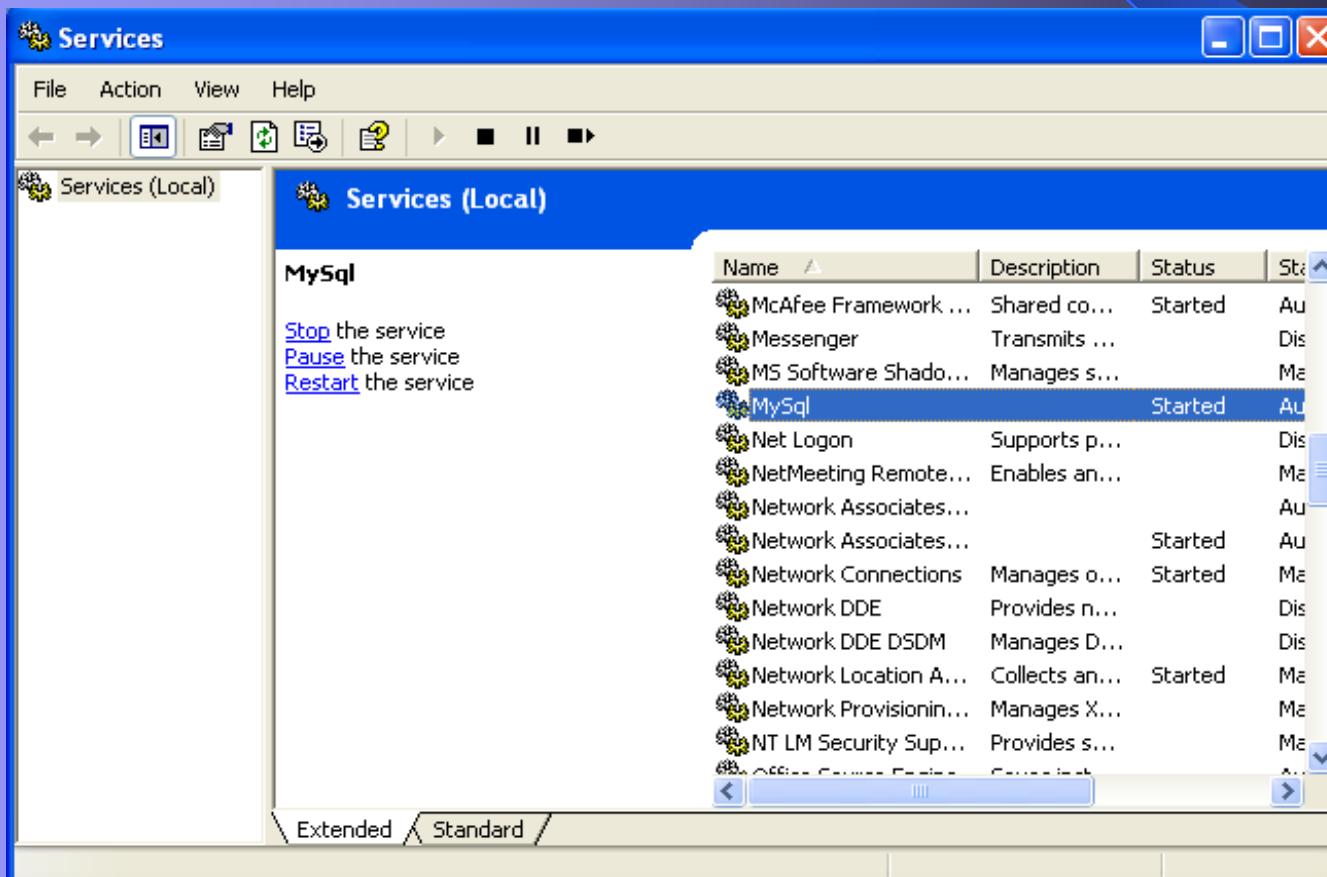


# HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU MYSQL

## II. CÀI ĐẶT PHP, MYSQL, APACHE

### 2. Cài đặt MySQL :

Khởi động MySQL: (Dùng Server)



# HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU MYSQL

## II. CÀI ĐẶT PHP, MYSQL, APACHE

### 3. Cài đặt PHP:

+ Nhấp đôi chuột lên chương trình cài đặt. Màn hình chào chương trình cài đặt PHP xuất hiện. Bấm Next để tiếp tục

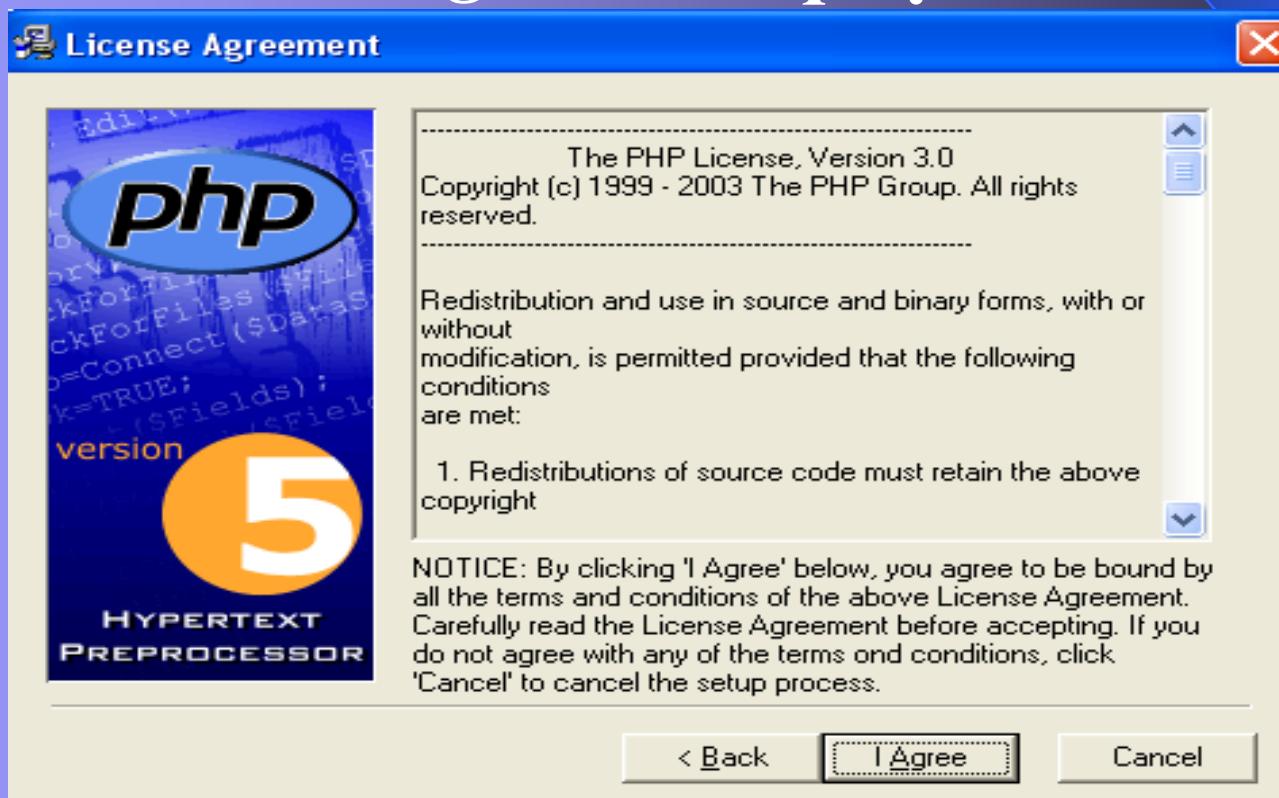


# HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU MYSQL

## II. CÀI ĐẶT PHP, MYSQL, APACHE

### 3. Cài đặt PHP:

+ Màn hình thỏa thuận bản quyền xuất hiện như hình sau. Bấm I Agree để tiếp tục.



# HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU MYSQL

## II. CÀI ĐẶT PHP, MYSQL, APACHE

### 3. Cài đặt PHP:

+ Tiếp theo chọn kiểu cài đặt, hãy chọn Standard.

Bấm Next để tiếp tục

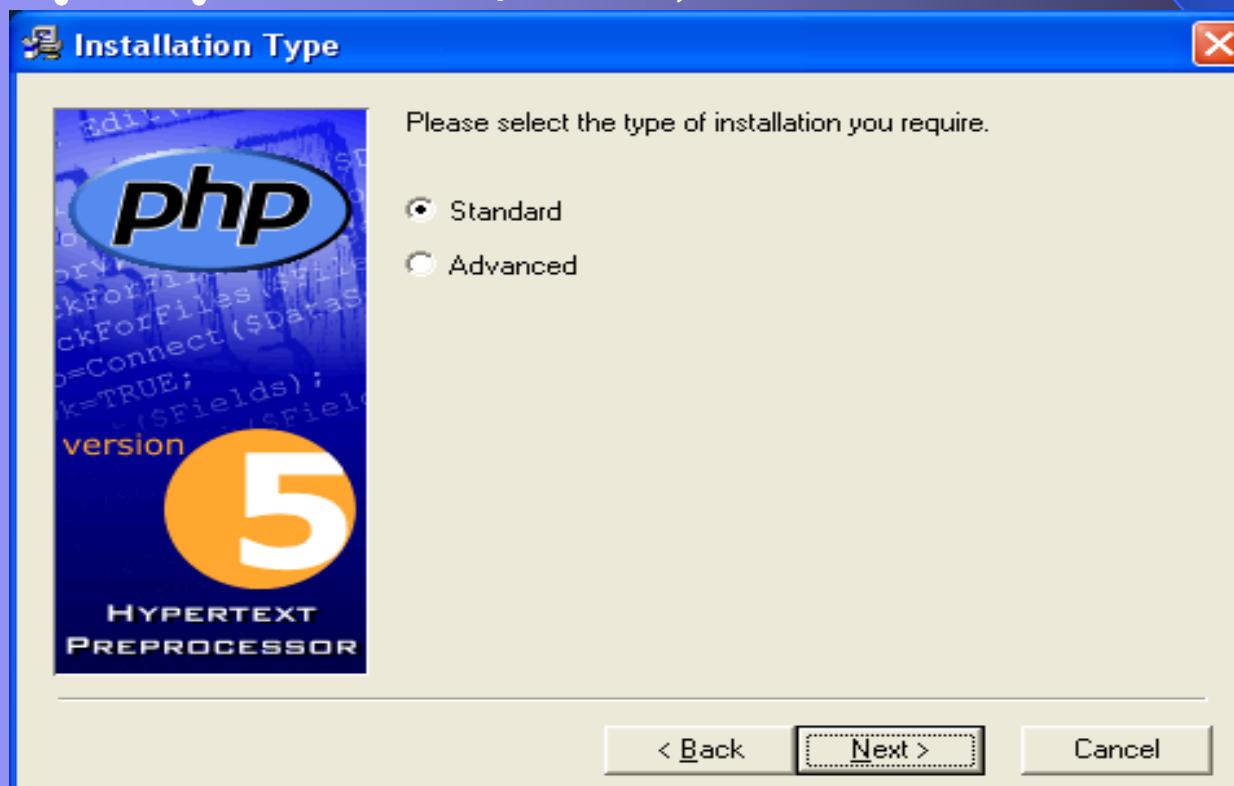


# HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU MYSQL

## II. CÀI ĐẶT PHP, MYSQL, APACHE

### 3. Cài đặt PHP:

+ Tiếp theo phép bạn chọn thư mục cài đặt PHP (mặc định là c:\PHP). Bấm Next để tiếp tục



# HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU MYSQL

## II. CÀI ĐẶT PHP, MYSQL, APACHE

### 3. Cài đặt PHP:

- + Màn hình thiết lập Mail xuất hiện như hình sau.  
Bấm Next để tiếp tục



# HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU MYSQL

## II. CÀI ĐẶT PHP, MYSQL, APACHE

### 3. Cài đặt PHP:

- + Tiếp theo bạn chọn kiểu Web server bạn muốn cấu hình để chạy PHP, trong trường hợp này chúng ta dùng Apache.
- + Bấm Next để tiếp tục. Tiếp tục quá trình cài đặt.
- + Cuối cùng bạn nhận được thông báo cho biết quá trình cài đặt PHP đã kết thúc

# CÔNG NGHỆ AJAX

## NỘI DUNG

1. Giới thiệu
2. Cơ chế và nguyên lý hoạt động
3. Mô hình tương tác Asynchronous
4. Đối tượng XMLHttpRequest
5. Các công nghệ hỗ trợ liên quan
6. Ajax và ứng dụng

# CÔNG NGHỆ AJAX

## I. Khái quát chung :

### 1. Giới thiệu :

+ Ajax - Asynchronous JavaScript and XML (JavaScript và XML không đồng bộ) là công cụ cho phép tăng tốc độ ứng dụng Web bằng cách cắt nhỏ dữ liệu và chỉ hiển thị những gì cần thiết thay vì phải tải đi tải lại toàn bộ trang Web.

+ Ajax không phải là một công nghệ đơn lẻ mà nó là tập hợp của nhiều công nghệ.

**AsynchronousJavascript + CSS + DOM + XMLHttpRequest**

# CÔNG NGHỆ AJAX

## I. Khái quát chung :

### 1. Giới thiệu :

+ Ajax bao gồm:

- Thể hiện Web theo tiêu chuẩn XHTML và CSS.
- Nâng cao tính năng động và phản hồi bằng DOM (Document Object Model).
- Trao đổi và truy cập/tác động lên thông tin, dữ liệu bằng XSL và XSLT.
- Nhận thông tin không đồng bộ (Synchronous) dùng XMLHttpRequest.
- Và tất cả các kỹ thuật trên được liên kết lại với nhau bằng Javascript.

# CÔNG NGHỆ AJAX

## I. Khái quát chung :

### 2. Cơ chế và nguyên lý hoạt động :

\* **Cơ chế :** Như vậy Ajax nó đóng vai trò như một lớp trung gian giữa giao diện trên trình duyệt và máy chủ xử lý thông tin. Có thể mô tả một cách cụ thể cách thức hoạt động của Ajax như sau :

- Ajax thực hiện tương tác với máy chủ bằng cách sử dụng đối tượng XMLHttpRequest, nhận kết quả trả về dưới dạng XML và phân tích kết quả bằng công nghệ DOM
- Tương tác giữa Ajax và giao diện người dùng được thực hiện thông qua các mã Javascript và XHTML+CSS.

# CÔNG NGHỆ AJAX

## I. Khái quát chung :

### 2. Cơ chế và nguyên lý hoạt động :

\* **Cơ chế :** - Công nghệ Ajax thì sẽ cho phép tạo ra một Ajax Engine. Khi đó các yêu cầu gửi Request và nhận Response sẽ do Ajax Engine thực hiện.

- Web server gửi trả dữ liệu dưới dạng XML và Ajax Engine sẽ tiếp nhận, sau đó thực hiện phân tích chuyển hóa thành XHTML+CSS cho trình duyệt hiển thị.

- Các việc này được thực hiện trên Client nên giảm tải rất nhiều cho Server, đồng thời với người sử dụng thì sẽ thấy kết quả hiển thị ngay tức thì mà không cần nạp lại nguyên cả trang.

# CÔNG NGHỆ AJAX

*I. Khái quát chung :*

*2. Cơ chế và nguyên lý hoạt động :*

\**Nguyên lý hoạt động : Để minh họa cho cơ chế Ajax xin đưa ra hai mô hình tương tác như sau :*

+ *Mô hình thứ nhất là mô hình tương tác giữa máy chủ và máy khách trong các ứng dụng Web truyền thống.*

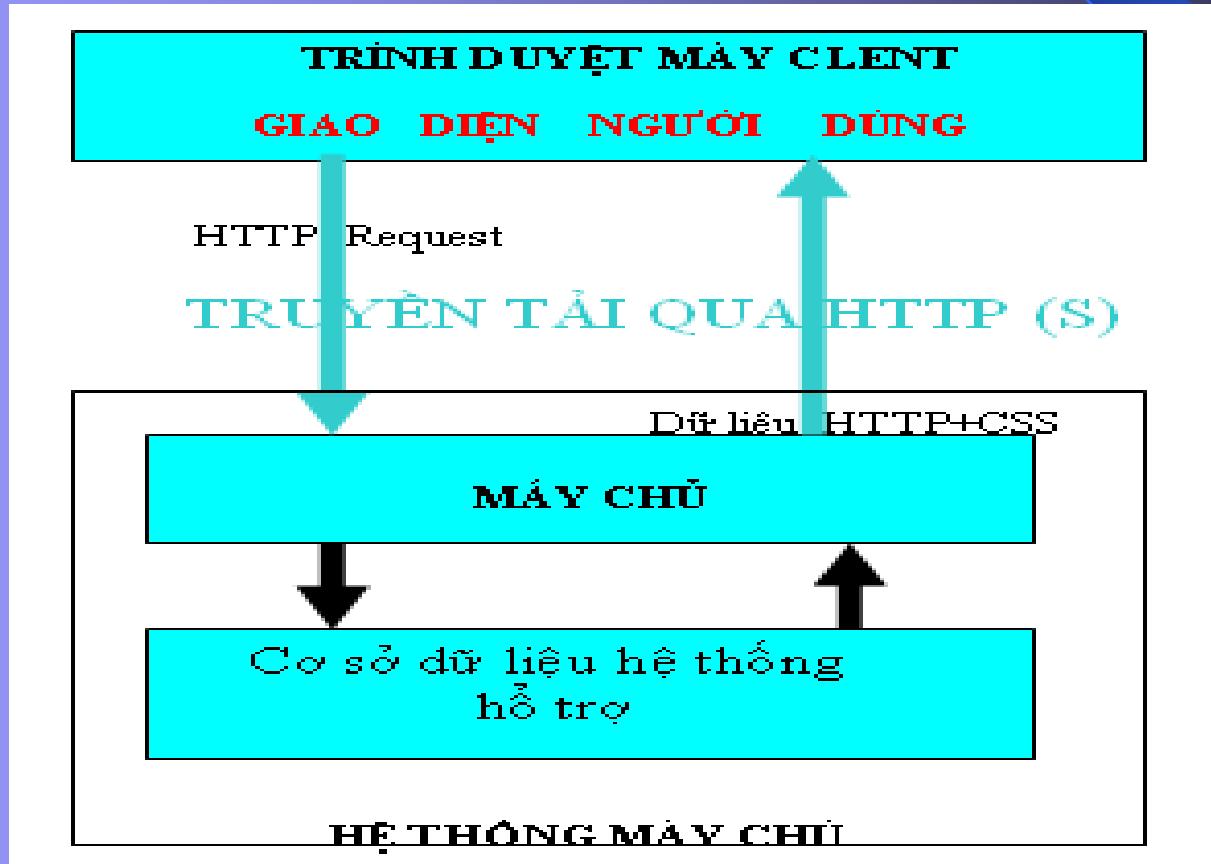
+ *Mô hình thứ hai thể biểu diễn quá trình tương tác giữa máy chủ và máy khách trong các ứng dụng Web có sử dụng công nghệ Ajax.*

# CÔNG NGHỆ AJAX

*1. Khái quát chung :*

*2. Cơ chế và nguyên lý hoạt động :*

*\*Nguyên lý hoạt động :*

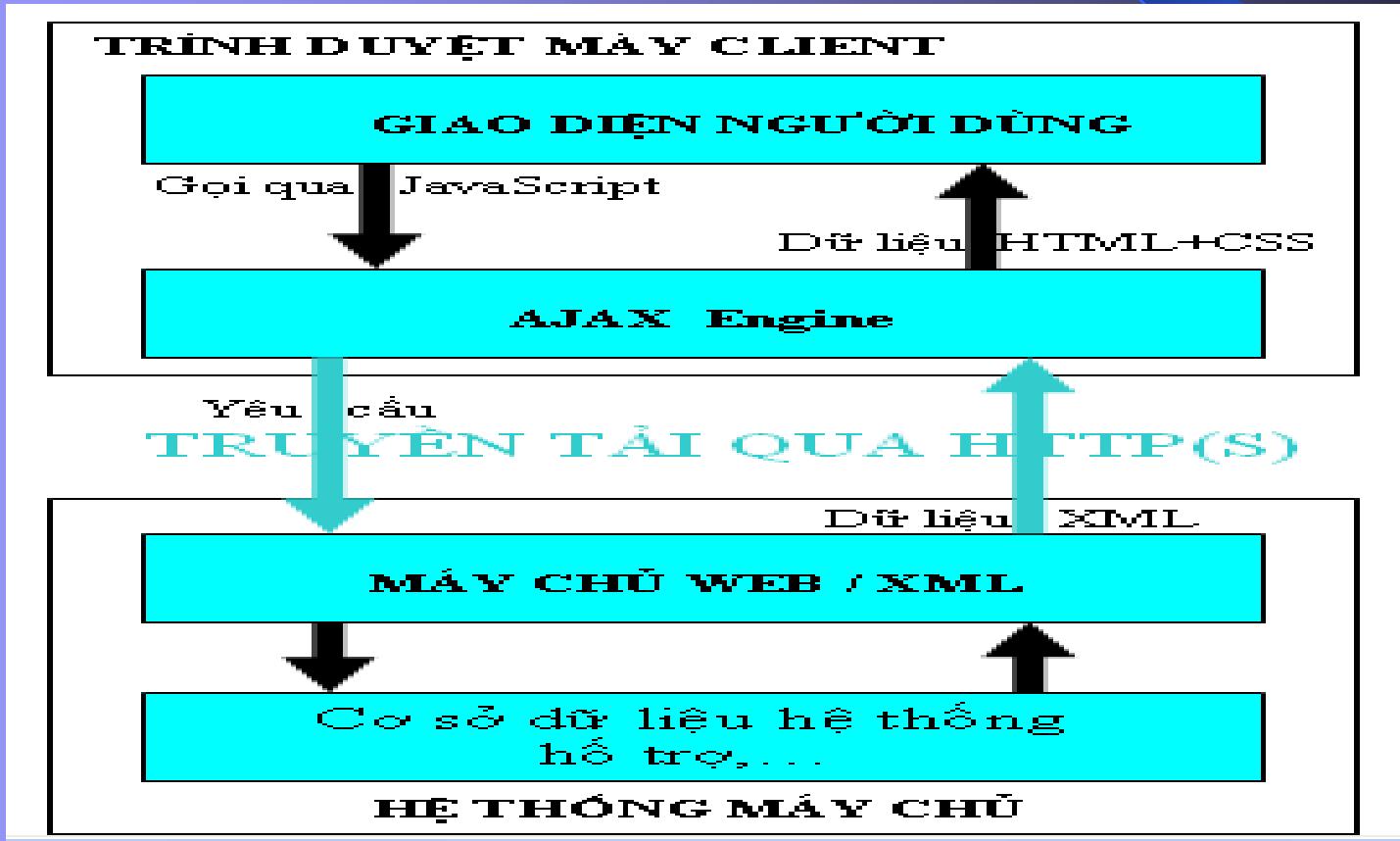


# CÔNG NGHỆ AJAX

*1. Khái quát chung :*

*2. Cơ chế và nguyên lý hoạt động :*

*\*Nguyên lý hoạt động :*



# CÔNG NGHỆ AJAX

*I. Khái quát chung :*

*2. Cơ chế và nguyên lý hoạt động :*

*\*Nguyên lý hoạt động :*

*+ Một trong những điểm mấu chốt của công nghệ Ajax là không tương tác trực tiếp với máy chủ như truyền thống mà là qua một lớp trung gian của Ajax.*

*+ Chính điều đó sẽ tạo ra sự ngạc nhiên về cơ chế xử lý này vì sẽ phải qua thêm một lớp trung gian và như vậy thì sẽ làm chậm đi quá trình tương tác.*

*Tuy nhiên thực tế thì không như vậy, nguyên nhân là :*

# CÔNG NGHỆ AJAX

*1. Khái quát chung :*

*2. Cơ chế và nguyên lý hoạt động :*

*\*Nguyên lý hoạt động :*

*- Không phải lúc nào cũng cần phải tương tác với máy chủ như trong trường hợp của Gmail, một khi giao diện của Gmail đã được tải về xong thì những tác vụ như chuyển từ thư mục này sang xem thư mục khác không nhất thiết phải đòi hỏi gọi lại máy chủ.*

*- Bằng cách chỉ nhận những thông tin cần thiết (ở dạng XML), dung lượng truyền tải giữa máy khách và máy chủ sẽ giảm đi rất nhiều. Điều này nghĩa là giao diện của ứng dụng chỉ cần tải về một lần (lần đầu)*

# CÔNG NGHỆ AJAX

*1. Khái quát chung :*

*2. Cơ chế và nguyên lý hoạt động :*

*\*Nguyên lý hoạt động :*

*- Ajax sẽ không phải thực hiện tải lại toàn bộ giao diện đó mỗi khi có tương tác với máy chủ. Thay vào đó Ajax sẽ gửi yêu cầu đến máy chủ và nhận kết quả từ máy chủ về những gì đã thay đổi sau khi máy chủ thực hiện yêu cầu đó.*

*- Ajax sau đó sẽ thông báo cho phần giao diện (qua các lời gọi Javascript) thực hiện các thay đổi tương ứng trên giao diện*

# CÔNG NGHỆ AJAX

*1. Khái quát chung :*

*2. Cơ chế và nguyên lý hoạt động :*

*\*Nguyên lý hoạt động :*

*- Các tương tác giữa phần giao diện với Ajax là tương tác nội bộ trên trình duyệt. Việc này giúp cho việc thực hiện các thay đổi được diễn ra rất nhanh (gần như ngay tức thì).*

*- Vì thế khi duyệt một trang hỗ trợ Ajax, người sử dụng không bao giờ thấy một cửa sổ trống (blank) và biểu tượng đồng hồ cát – là những dấu hiệu cho thấy máy chủ đang thực hiện công việc.*

# CÔNG NGHỆ AJAX

## I. Khái quát chung :

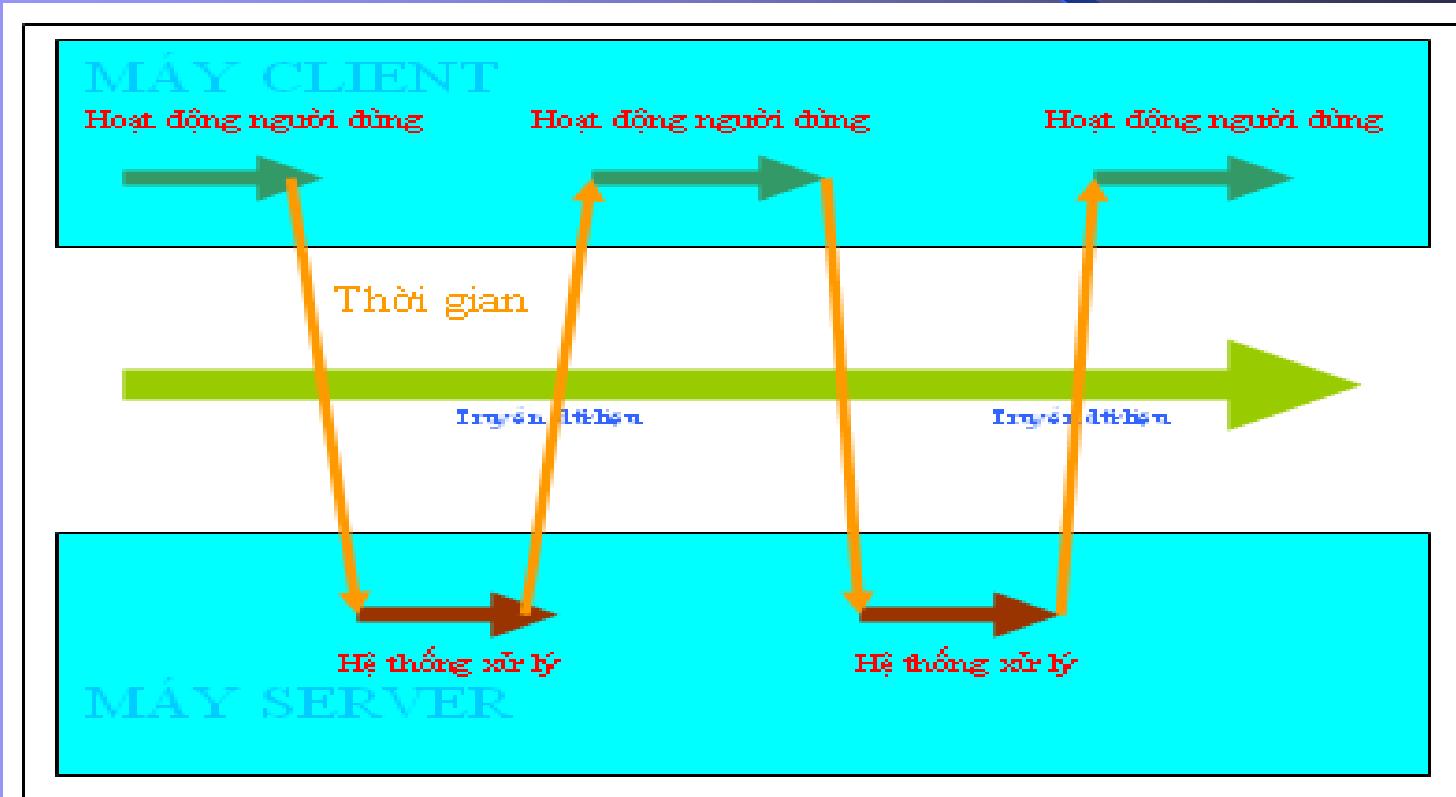
### 3. Mô hình tương tác Asynchronous

- + Sự cải tiến trong mô hình tương tác máy chủ - máy khách là một trong những điểm khác biệt của Ajax
- + Mô hình tương tác của các ứng dụng web truyền thống (Mô hình đồng bộ )

# CÔNG NGHỆ AJAX

## I. Khái quát chung :

### 3. Mô hình tương tác Asynchronous



Mô hình tương tác của các ứng dụng web truyền thống  
(Mô hình đồng bộ )

# CÔNG NGHỆ AJAX

## I. Khái quát chung :

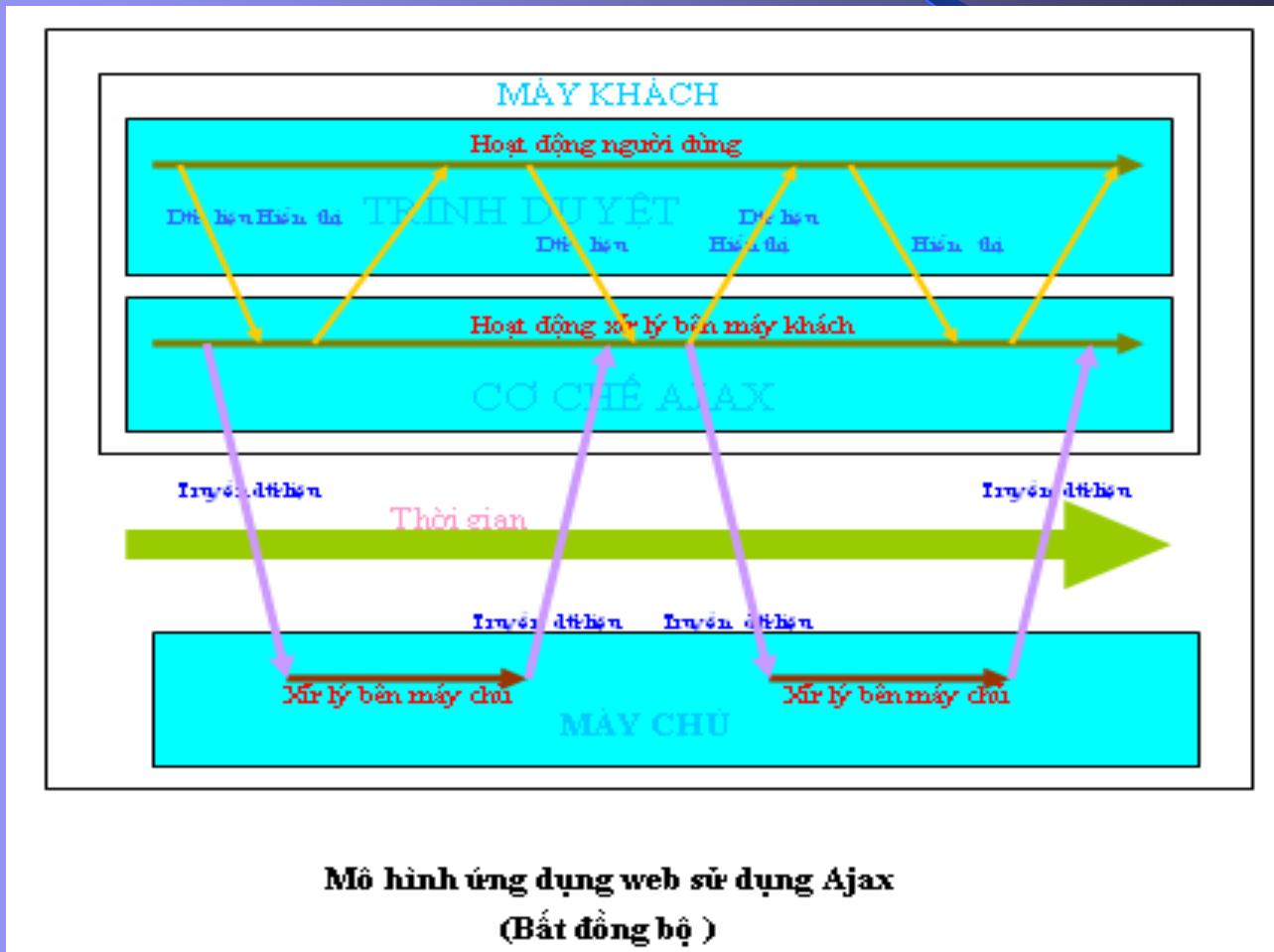
### 3. Mô hình tương tác Asynchronous

- + Theo mô hình trên có sự thiếu hợp lý ở chỗ: Khi người dùng có yêu cầu và gửi lên máy chủ thì chờ cho tới khi máy chủ xử lý xong và sau khi nhận được kết quả từ máy chủ trả về thì người dùng mới có thể tiếp tục thực hiện các thao tác tiếp theo.
- + Như vậy thì đặt trong trường hợp nếu ta cần xử lý dữ liệu lớn thì không còn cách nào khác là phải chờ khá lâu.
- + Đây chính là nhược điểm của các công nghệ Web truyền thống và Ajax đã giải quyết được vấn đề này .

# CÔNG NGHỆ AJAX

## I. Khái quát chung :

### 3. Mô hình tương tác Asynchronous



# CÔNG NGHỆ AJAX

## I. Khái quát chung :

### 3. Mô hình tương tác Asynchronous

+ Để ý thấy rằng tương tác giữa giao diện máy khách của chúng ta với Ajax diễn ra mang tính chất ngẫu nhiên và rất thường xuyên

+ Tương tác giữa Ajax và máy chủ xảy ra với chu kỳ dài hơn nhiều. Chính vì vậy việc xử lý những yêu cầu thực hiện trên máy chủ sẽ không ảnh hưởng gì đến quá trình tương tác giữa người dùng với máy client. Do đó làm giảm đi khoảng thời gian đợi như ở các mô hình truyền thống.

# CÔNG NGHỆ AJAX

## III. Các khái niệm liên quan Ajax :

### 1. Ngôn ngữ XML

+ XML (Extensible Markup Language) - là một bộ quy luật bao gồm các quy tắc để chia một tài liệu ra làm nhiều phần, rồi đánh dấu và ráp các phần khác nhau lại để dễ nhận diện chúng.

+ Tổ hợp Web toàn cầu W3C gọi XML là “một cú pháp thông dụng cho việc hiển thị cấu trúc trong dữ liệu”.

# CÔNG NGHỆ AJAX

## III. Các khái niệm liên quan Ajax :

### 1. Ngôn ngữ XML

+ Mục tiêu chính của XML là tách biệt nội dung và dạng biểu diễn của văn bản (Baldwin).

+ Cùng một nội dung ta có thể biểu diễn dưới nhiều hình thức khác nhau : chẳng hạn in ra giấy hoặc đưa lên Web, và có thể áp dụng nhiều thuộc tính trên văn bản như in đậm, in nghiêng, in gạch dưới, dùng màu,...

+ Ngoài ra nhờ việc sử dụng các ký tự thông thường mà con người có thể hiểu được nội dung của văn bản ngay khi còn đang ở dạng thô, chưa được áp dụng bất kỳ dạng biểu diễn nào

# CÔNG NGHỆ AJAX

## III. Các khái niệm liên quan Ajax :

### 1. Ngôn ngữ XML

Một trang XML cần phải theo đúng các quy luật sau đây :

- + Trang XML phải bắt đầu bằng câu tuyên bố XML
- + Mỗi bộ phận, gọi là “element” phải nằm giữa Tag Pair.
- + Nếu Tag nào không chứa gì ở giữa thì phải chấm dứt bằng ”>”

# CÔNG NGHỆ AJAX

## III. Các khái niệm liên quan Ajax :

### 1. Ngôn ngữ XML

Một trang XML cần phải theo đúng các quy luật sau đây : + Một trang XML phải có một element độc nhất chứa tất cả các element khác. Đó là root của cây biểu diễn trang XML .

+ Các Tag Pair không được xen kẽ nhau, các Tag Pair phải đánh vần đúng y như nhau kể cả chữ hoa, chữ thường (<STUDENT> và </student> là bất hợp lệ) và các Attribute đều phải nằm giữa hai ngoặc kép (standalone=yes là bất hợp lệ, phải viết standalone="yes").

# CÔNG NGHỆ AJAX

## III. Các khái niệm liên quan Ajax :

### 1. Ngôn ngữ XML

Ví dụ :

```
<?xml version="1.0" standalone="yes"?>
```

```
<gicungduoc>
```

*Chào XML ! Mạnh giỏi ?*

```
</gicungduoc>
```

# CÔNG NGHỆ AJAX

## III. Các khái niệm liên quan Ajax :

### 2. Đối tượng DOM (Document Object Model)

+ Mô hình đối tượng tài liệu là một giao diện lập trình ứng dụng (API).

+ Thông thường DOM có dạng một cây cấu trúc dữ liệu và được dùng để truy xuất các tài liệu HTML và XML.

+ Mô hình DOM hoạt động độc lập với hệ điều hành và dựa theo kỹ thuật lập trình hướng đối tượng để mô tả tài liệu.

+ Trong Ajax mô hình DOM đóng vai trò phân tách dữ liệu truyền dưới dạng XML.

# CÔNG NGHỆ AJAX

## III. Các khái niệm liên quan Ajax :

### 2. Đối tượng DOM

+ Để phân tích một văn bản XML bằng máy tính ta thường dùng một trong hai hoặc cả hai mô hình phân tích là SAX (Simple API for XML) và DOM.

+ Theo mô hình phân tích DOM, toàn bộ nội dung của một văn bản XML sẽ được lưu giữ vào bộ nhớ trong dưới dạng một cấu trúc cây

# CÔNG NGHỆ AJAX

## III. Các khái niệm liên quan Ajax :

### 3. Giới thiệu về XHTML

+ Trong Ajax XHTML cùng với CSS đóng vai trò hiển thị dữ liệu cho trang Web.

+ XHTML là một chuẩn kết hợp giữa HTML và XML và trở thành một định dạng riêng cho web. XHTML cho phép HTML được mở rộng bằng các thẻ sở hữu.

+ Ngoài ra XHTML cũng được mã hoá chặt chẽ hơn HTML và phải tuân thủ nhiều quy tắc cấu trúc hơn HTML

# CÔNG NGHỆ AJAX

## III. Các khái niệm liên quan Ajax :

### 3. Giới thiệu về XHTML

+ Trong XHTML tất cả các thẻ phải được đóng, các thuộc tính phải được cho vào dấu ngoặc kép (""), và đặc biệt tất cả các thẻ và thuộc tính đều phải viết bằng chữ cái thường.

+ Mặc dù điều này có thể XHTML bị xem là cứng nhắc nhưng chính điều này lại là ưu điểm của nó: quy chuẩn và nguyên tắc.

+ XHTML là phiên bản kết hợp của HTML nên có những điểm khác cơ bản với HTML

# CÔNG NGHỆ AJAX

## III. Các khái niệm liên quan Ajax :

### 3. Giới thiệu về XHTML

- Các File XHTML cần phải well-formed có nghĩa là các thẻ `<htm>` `<head>` `<body>`...phải tuân thủ theo chuẩn .

- Các thẻ lồng nhau phải đúng trật tự .  
- Các thẻ và các thuộc tính của các phần tử phải được viết bằng chữ thường.

- Các thuộc tính của các phần tử phải được đóng nháy kép: ví dụ `<table border="1" width="100%">` ,...

- Các thẻ được gọi là noempty cần phải được đóng bằng một thẻ đóng( `<br>` ...`</br>`)

# CÔNG NGHỆ AJAX

## III. Các khái niệm liên quan Ajax :

### 3. Giới thiệu về XHTML

- Không cho phép giản lược các thuộc tính.

Ví dụ : trong HTML có thể viết <option selected> nhưng trong XHTML phải viết đủ <option selected="selected"> và không được đóng bằng thẻ </option>.

- Thuộc tính name được thay bằng thuộc tính id.

Ví dụ :  được thay thế bằng 

- Phải khai báo DOCTYPE trước thẻ <html>

# CÔNG NGHỆ AJAX

## III. Các khái niệm liên quan Ajax :

### 3. Giới thiệu về XHTML

*<!DOCTYPE Doctype goes here>*

*<html>*

*<head>*

*<title>Vi du ve XHTML</title>*

*</head>*

*<body>*

*This is XHTML page*

*</body>*

*</html>*

# CÔNG NGHỆ AJAX

## III. Các khái niệm liên quan Ajax :

### 4. CSS (*Cascading Style Sheets*)

+ Tập định kiểu theo tầng, được dùng để miêu tả cách trình bày các tài liệu viết bằng ngôn ngữ HTML, XHTML, XML, SVG, XUL...

+ Các thông số kỹ thuật của CSS do World Wide Web Consortium (W3C) quản lý. Trong Ajax CSS cùng với XHTML đóng vai trò hiển thị dữ liệu cho trang web .

# CÔNG NGHỆ AJAX

## III. Các khái niệm liên quan Ajax :

### 5. DHTML

+ Hay HTML động tạo một trang Web bằng cách kết hợp các thành phần ngôn ngữ đánh dấu HTML tinh, ngôn ngữ lệnh máy khách (như JavaScript ) và ngôn ngữ định dạng CSS và DOM.

+ Do có khả năng phong phú, DHTML còn được sử dụng như một công cụ để xây dựng các trò chơi đơn giản trên trình duyệt .

# CÔNG NGHỆ AJAX

## III. Các khái niệm liên quan Ajax :

### 6. Đối tượng XMLHttpRequest

+ Để hiểu được cơ chế hoạt động của Ajax thì chúng ta phải biết được XMLHttpRequest là gì ? Như nhiều chuyên gia đã nhận định một trong những điểm quan trọng nhất của Ajax là nằm ở XMLHttpRequest

+ Điều đó cho thấy rằng, vì XMLHttpRequest có vai trò quan trọng đặc biệt trong công nghệ như vậy nên một ứng dụng của Ajax có thực thi và mang hiệu quả hay không phụ thuộc vào việc trình duyệt có hỗ trợ đối tượng này. Nếu một trình duyệt không hỗ trợ XMLHttpRequest sẽ không thể diễn ra quá trình trao đổi giữa máy client và máy server

# CÔNG NGHỆ AJAX

## III. Các khái niệm liên quan Ajax :

### 6. Đối tượng XMLHttpRequest

+ Thông thường trong một ứng dụng web, để có thể gửi một HTTP Request đến máy chủ bằng JavaScript bạn cần tạo một đối tượng của lớp cung cấp tính năng này. Trong Internet Explorer (IE) thì lớp này tồn tại dưới dạng một ActiveX có tên là XMLHTTP

+ Đối tượng XMLHttpRequest không phải là một chuẩn của W3C (World Wide Web Consortium). Đối tượng XMLHttpRequest được hỗ trợ ở IE 5.0+, Safari 1.2, Mozilla 1.0/Firefox, Opera 8.0 và Netcape 7+.

*var httpReqquest = new XMLHttpRequest();*

# CÔNG NGHỆ AJAX

## III. Các khái niệm liên quan Ajax :

### 6. Đối tượng XMLHttpRequest

+ Do sự khác biệt này nên để các ứng dụng có thể chạy được trên các trình duyệt chúng ta phải khai báo:

```
if(window.XMLHttpRequest)
{
    //Mozilla,Safari,..
    httpRequest=new XMLHttpRequest();
}
else if(window.ActiveXObject)
{
    //IE..
    httpRequest=new
    ActiveXObject("Microsoft.XMLHTTP");
}
```

# CÔNG NGHỆ AJAX

## III. Các khái niệm liên quan Ajax :

### 6. Đối tượng XMLHttpRequest

+ Do ActiveX trên IE rất nguy hiểm cho nhiều người dùng nên trong nhiều trường hợp tính năng này bị cấm. Vì vậy bạn cần phải kiểm tra trình duyệt trước khi gọi đối tượng XMLHttpRequest. Quá trình kiểm tra này được thực hiện thông qua giá trị của window.ActiveXObject

Ví dụ : `if(window.ActiveXObject) {  
 xmlhttp=new  
 ActiveXObject("Microsoft.XMLHTTP");  
}  
else{...}`

# Ng«n ng÷ Truy vÊn d÷ liÖu (SQL)

## I. Giíi thiÖu

Ng«n ng÷ SQL (Structured Query Language) ®U'c h·ng m,y tÝnh IBM ®a ra vµo n·m 1970.

Lµ mét ng«n ng÷ cã cÊu tróc, cã nghÜa lµ b»ng tËp h·p c,c dßng lÖnh sї chØ cho hÖ qu¶n trb CSDL biÖt lµ ngU'ei sö dÔng sї lµm g× chø kh«ng chØ ra lµm nhU' thÖ nµo.

Ng«n ng÷ SQL cung cÊp mét c«ng cô truy vÊn (query) nh»m cho phĐp ngU'ei sö dÔng ®Æt c©u hái vÏi d÷ liÖu ®ang chøa trong c,c b¶ng, d÷ liÖu ®îc rót ra tõ nh÷ng truy vÊn cã thÓ cã nguân tõ mét hay nhiÒu b¶ng kh,c nhau.

Sau ®ã tham kh¶o kÖt qu¶ nµy, xem xDt, ph©n tÝch nã kü lU'ing ®Ó cã nh÷ng xö lý th«ng tin thiÖt thuc h¬n.

# Ng«n ng÷ Truy vÊn d÷ liÖu (SQL)

## II. C,c c©u lÖnh SQL

Trong mÔc nµy tr×nh bµy mét sè lÖnh c¬ b¶n dïng ®Ó thao t,c trªn c¬ së d÷ liÖu nh: chän d÷ liÖu, thªm vµo d÷ liÖu mí, xo, d÷ liÖu cò, söa ®æi d÷ liÖu.

**1. LÖnh Select :** LÖnh nµy dïng ®Ó lÊy d÷ liÖu trong mét b¶ng tho¶ m·n ®iÒu kiÖn nµo ®ã.

Có ph,p:

Select [\*|*fields*] From *tablename*

[*Where expression*]

[*Group by fields1*]

[*Having expression1*]

[*Order by {colname }[ASC | DESC]*]

# Ng«n ng÷ Truy vÊn d÷ liÖu (SQL)

## II. C,c cØu lÖnh SQL

- **\***: tÊt c¶ c,c trêng ®Òu ®îc chän.
- **column**: tªn cét.
- **tablename**: lµ tªn cña mét Table hoÆc nhiÒu Table mµ c,c Record ®îc chän.
- **expression**: lµ biÓu thøc ®iÒu kiÖn ®îc ®Æt ra mµ c,c b¶n ghi ®îc chän tõ **tablename** ph¶i tho¶i m·n biÓu thøc nµy.
- **expression1**: mét biÓu thøc ®iÒu kiÖn ®i sau mÖnh ®Ò Having. MÖnh ®Ò Having th«ng thêng ®i kÌm víi mÖnh ®Ò Group by, **expression1** lµ biÓu thøc kh«ng t,c ®éng vµo topn b¶ng ®îc chØ ra bëi mÖnh ®Ò From mµ chØ t,c ®éng lÇn lít tõng nhãm c,c b¶n ghi ®· ®îc chØ ra bëi mÖnh ®Ò Group by.
- **Order by**: đing ®Ó s¾p c,c b¶n ghi t“ng hay gi¶m

# Ngo;n ngd; Truy vEn dd; liÖu (SQL)

## II. Cc c@u lÖnh SQL

### 1. LÖnh Select

- VÝ dô: Giả sÙ' cã mét Table cã ta;n NHANSU nh sau:**

Maso	Hoten	Ngaysinh	Nghenghiep
1	L <e>a; V<e>n TuÊn</e></e>	10/8/1990	H <e>a;c sinh</e>
2	NguyÔn V <e>n H<e>u</e></e>	7/8/1970	K <e>u;s</e>
3	NguyÔn Th <e>p Hoa</e>	30/07/1980	Sinh vi <e>a;n</e>

# Ng<sup>o</sup>n ng<sup>o</sup> Truy v<sup>E</sup>n d<sup>o</sup> l<sup>i</sup>Öu (SQL)

## II. C<sup>c</sup> c<sup>c</sup> l<sup>i</sup>Önh SQL

### 1. L<sup>i</sup>Önh Select

- N<sup>o</sup>u s<sup>o</sup> d<sup>o</sup>ng l<sup>i</sup>Önh: **Select Hoten, Ngaysinh From nhansu**
- S<sup>i</sup> ch<sup>a</sup>n t<sup>E</sup>t c<sup>¶</sup> c<sup>c</sup> b<sup>¶</sup>n ghi nhng chØ gi<sup>i</sup>i h<sup>1</sup>n b<sup>e</sup>i hai tr<sup>B</sup>ng Hoten v<sup>u</sup> Ngaysinh.  
Cô thÓ g<sup>o</sup>m c<sup>c</sup> b<sup>¶</sup>n ghi sau:

L <sup>a</sup> V <sup>a</sup> n Tu <sup>E</sup> n	10/8/1990
Nguy <sup>o</sup> n v <sup>a</sup> n H <sup>u</sup>	7/8/1970
Nguy <sup>o</sup> n Th <sup>b</sup> Hoa	30/07/1980

# Ng«n ng÷ Truy vÊn d÷ liÖu (SQL)

## II.C,c c©u lÖnh SQL

**2. LÖnh Insert Into :** LÖnh nµy dïng ®Ó chÌn thªm mét Record d÷ liÖu mÙi vµo cho 1 b¶ng.

- Có ph,p:

*Insert Into table[(field1[,field2[,...]])] Values(value1[, value2[,...]])*

- Trong ®ã:

*Table:* tªn cña Table cÇn bæ sung b¶n ghi

*filed1,field2:* tªn cña c,c trªng cÇn bæ sung d÷ liÖu

*value1,value2:* gi, tr¶c cÇn bæ sung øng vÙi c,c trªng *filed1,field2...*

# Ng«n ng÷ Truy vÊn d÷ liÖu (SQL)

## II. C,c cÓu lÖnh SQL

### 2. LÖnh Insert Into

- VÝ dô:
- Gi¶ sö ta cã ®îc Table Nhansu nh trªn, b©y giê bæ sung vµo b¶n ghi mii:  
`Insert Into nhansu (Hoten, Ngaysinh, Nghenghiep) Values("NguyÊn v n An","10/7/1995","H c sinh")`
- Sau khi th c hiÖn lÖnh n y b¶ng Nhansu b©y giê nh u sau:

# Ng<sup>o</sup>n ng<sup>o</sup> Truy v<sup>E</sup>n d<sup>o</sup> liÖu (SQL)

## II.C, c c<sup>o</sup>u lÖnh SQL

### 2. LÖnh Insert Into

Maso	Hoten	Ngaysinh	Nghenghiep
1	L <sup>a</sup> Văn TuÊn	10/08/1990	Häc sinh
2	NguyÔn văn Hµ	07/08/1970	Kü s
3	NguyÔn Th <sup>b</sup> Hoa	30/07/1980	Sinh vi <sup>a</sup> n
4	NguyÔn văn An	10/07/1995	Häc sinh

# Ng«n ng÷ Truy vÊn d÷ liÖu (SQL)

## II. C,c c©u lÖnh SQL

**3. LÖnh Update:** LÖnh nµy dñng ®Ó thay ®æi d÷ liÖu cho c,c trÙêng trong mét b¶ng tho· m·n ®iÖu kiÖn nµo ®ã

- Có ph,p:

*Update table Set Field1=value1[,...] Where expression*

- Trong ®ã:

*table:* lµ tªn cña Table chøa b¶n ghi cÇn söa ®æi.

*Field1:* trÙêng cÇn thay ®æi d÷ liÖu

*Value 1:* gi, trØ d÷ liÖu mii

*expression:* lµ biÓu thøc, mµ c,c b¶n ghi ®îc cËp nhËt ph¶i tho¶ m·n biÓu thøc nµy

# Ng«n ng÷ Truy vÊn d÷ liÖu (SQL)

## II. C,c cØu lÖnh SQL

### 3. LÖnh Update

- VÝ dô:
- Söa ®æi trêng Nghenghiep cã gi, trØ mïi thµnh “CNVC” cho tÊt c¶ c,c record tho¶ m·n ®iÙu kiÖn lµ trêng Ngaysinh="07/08/1970":

Update	Nhansu	Set	Nghenghiep="CNVC"	Where
Ngaysinh="07/08/1970"				

# Ng<sup>o</sup>n ng<sup>o</sup> Truy v<sup>E</sup>n d<sup>o</sup> liÖu (SQL)

## II. C<sup>c</sup> c<sup>o</sup>u lÖnh SQL

### 3. LÖnh Update

Table Nhansu b<sup>c</sup>y giê nh sau:

Maso	Hoten	Ngaysinh	Nghenghiep
1	L <sup>a</sup> Văn TuÊn	10/08/1990	Häc sinh
2	NguyÔn văn H <sup>u</sup>	07/08/1970	CNVC
3	NguyÔn Th <sup>p</sup> Hoa	30/07/1980	Sinh vi <sup>a</sup> n
4	NguyÔn văn An	10/07/1995	Hoc sinh

# Ng«n ng÷ Truy vÊn d÷ liÖu (SQL)

## II. C,c cÓu lÖnh SQL

### 4. LÖnh Delete

LÖnh nµy dïng ®Ó xo, mét hoÆc nhiÒu record trong mét b¶ng tho¶ m·n ®iÒu kiÖn nµo ®ã.

- Có ph,p:

*Delete \* From **table** Where **expression***

- Trong ®ã:

**table:** lµ tªn Table chøa c,c b¶n ghi cÇn xo,.

**\***: tÊt c¶ c,c trêng cña b¶n ghi ®Òu b¶ xo,.

**expression:** lµ tÊt c¶ c,c b¶n ghi ®îc xo, ph¶i tho¶ m·n biÓu thøc nµy.

# Ng<sup>«</sup>n ng<sup>÷</sup> Truy v<sup>È</sup>n d<sup>÷</sup> liÖu (SQL)

## II. C<sup>c</sup> c<sup>c</sup>u lÖnh SQL

### 4. LÖnh Delete

VÝ dô:

- Xo<sup>t</sup>t<sup>Ê</sup>t c<sup>T</sup> c<sup>c</sup> b<sup>T</sup>n ghi trong Table Nhansu c<sup>a</sup> trêng Nghenghiep=“Häc sinh”.
- Delete \* From nhansu Where Nghenghiep=“Häc sinh”

B<sup>C</sup>y giê table Nhansu trê th<sup>µ</sup>nh:

Maso	Hoten	Ngaysinh	Nghenghiep
2	NguyÔn vân Hµ	07/08/1970	CNVC
3	NguyÔn Th <sup>b</sup> p Hoa	30/07/1980	Sinh vi <sup>a</sup> n

# Ng«n ng÷ Truy vÊn d÷ liÖu (SQL)

Chú ý: Trong mệnh đề WHERE, các phép toán được sử dụng là:

Phép toán	Mô tả
=	So sánh bằng
<>	So sánh khác
>	Lớn hơn
<	Nhỏ hơn
>=	Lớn hơn hoặc bằng
<=	Nhỏ hơn hoặc bằng
BETWEEN	Nằm giữa một khoảng
LIKE	So sánh mẫu chuỗi

Lưu ý: Trong một số phiên bản của SQL, phép toán  $\neq$  có thể được viết dưới dạng !=

# Ng«n ng÷ Truy vÊn d÷ liÖu (SQL)

## \* Sử dụng dấu nháy

Lưu ý rằng ở ví dụ trên ta đã sử dụng hai dấu nháy đơn ('') bao quanh giá trị điều kiện 'Sandnes'.

SQL sử dụng dấu nháy đơn bao quanh các giá trị ở dạng chuỗi văn bản (text). Nhiều hệ CSDL còn cho phép sử dụng dấu nháy kép (""). Các giá trị ở dạng số không dùng dấu nháy để bao quanh.

### + Với dữ liệu dạng chuỗi văn bản:

Câu lệnh đúng:

```
SELECT * FROM Persons WHERE FirstName = 'Tove'
```

Câu lệnh sai:

```
SELECT * FROM Persons WHERE FirstName = Tove
```

+ VỚI DỮ LIỆU DẠNG SỐ:

Câu lệnh đúng:

```
SELECT * FROM Persons WHERE Year > 1965
```

Câu lệnh sai:

```
SELECT * FROM Persons WHERE Year > '1965'
```

## \* Phép toán điều kiện LIKE

Phép toán LIKE được dùng để tìm kiếm một chuỗi mẫu văn bản trên một cột.

### Cú pháp :

Cú pháp của phép toán LIKE như sau:

**SELECT tên\_cột FROM tên\_bảng  
WHERE tên\_cột LIKE mẫu**

Một ký hiệu % có thể được sử dụng để định nghĩa các ký tự đại diện. % có thể được đặt trước và/hoặc sau mẫu.

## + Sử dụng LIKE

Câu lệnh SQL sau sẽ trả về danh sách những người có tên bắt đầu bằng chữ O:

```
SELECT * FROM SINHVIEN  
WHERE ten LIKE 'O%'
```

Câu lệnh SQL sau sẽ trả về danh sách những người có tên kết thúc bằng chữ a:

```
SELECT * FROM sinhvien  
WHERE ten LIKE '%a'
```

Câu lệnh SQL sau sẽ trả về danh sách những người có tên kết chứa chuỗi la:

```
SELECT * FROM sinhvien  
WHERE ten LIKE '%la%'
```

# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## I. CÊU HXNH TRXNH CHÑ WEB SERVER

### 1. GIÏI THIÖU VÒ JRUN

- + §Ó nghiän cøu vÒ JSP tríc tiän ph¶i cã trxnh chñ Web Server “hiÓu Java”. Trxnh chñ ®ã ph¶i cã kh¶ n¤ng th¤ng dÞch vµ gäi thùc thi ®îc c,c ®¬n thÓ Java theo ®Æc t¶ cña Sun.
- + MÆc dï hiÖn nay cã rÊt nhiÓu trxnh chñ Web dµnh cho Java nh Java Web Server, TomCat, IPLanet, Jrun,...Nhng do Jrun viÖt b»ng Java n¤n b¹n cã thÓ sö dÔng vµ cµi ®Æt Jrun c¶ træn m¤i trêng Linux lÉn Unix.
- + Trxnh Jrun cung cÊp kh¶ n¤ng triÖu gäi m· Java th¤ng qua Servlet. Jrun cung cÊp cho ta hai dÞch vô Web ho¹t ®éng træn hai cæng:
  - Cæng 8100 lµ dÞch vô Web th¤ng thêng vµ cæng 8000 lµ dÞch vô Web phôc vô cho cÊu hxnh trxnh chñ.

Còng nh nh÷ng Web server kh,c, t×m hiÓu Jrun bao g m c,c b c sau:

- Cµi ®Æt trxnh chñ Jrun.
- CÊu hxnh kÖt n i v i c,c Web server kh,c.
- Qu¶n trþ Jrun.
- T×m hiÓu c,c file quan tr ng cña Jrun.

# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## I. CẤU HÌNH TRÌNH CHỦ WEB SERVER

### 2. Cài Đặt Jrun

#### 2.1. Cài Đặt JDK

+ Tríc khi cài Đặt trình chún Jrun Web Server ta phái cài Đặt JDK1.3. Ta sö dông gãi cài Đặt j2sdk1\_3\_0-win.exe giúi nĐn. MÆc Đopenhagen JDK cña Java sї ®íc cài vµo th môc C:\Jdk1.3. Khi Jrun yáu cÇu chØ Đopenhagen JDK ta h·y chØ ®êng dÉn ®Õn th môc ®ã.

#### 2.2. Cài Đặt Jrun Web Server

+ Yáu cÇu vÒ phÇn cøng vµ phÇn mÒm :

- 32 MB Ram
- 20 MB dung lîng Đuga cøng
- HÖ Điều hµnh : Windows9x hay NT/2000, Unix vµ Linux.
- Sö dông trình duyÖt Netscape hay Internet Explorer
- M«i trêng m,y Tô JRE (Java Runtime Environment)
- JDK

# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## I. CẤU HÌNH TRÌNH CHẠY Web Server

### 2. Cài đặt Jrun

#### 2.3. Cấu trúc thư mục của Jrun

<i>Thư mục</i>	<i>Nội dung</i>
/bin	Chứa các tập tin chung trinh cua Jrun
/connectors	Chứa các tập tin Web server kết nối
/docs	Chứa các trang tui lieu html hình ảnh số đông
/lib	Chứa các tập tin thư viện .jar
/lib/ext	Chứa các tập tin .jar gồm servlet.jar & ejb.jar
/logs	Chứa các tập tin log của Jrun
/samples	Chứa các trang jsp và servlet mẫu
/servers	Chứa các config đồng Web của Jrun
/servers/lib	Chứa các tập tin .jar và các file class
/servlets	Chứa các tập tin .class của servlet truy cập
/uninst	Thông tin về hướng dẫn cài đặt Jrun

# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## I. CẤU HÌNH TRÌNH CHỦ WEB SERVER

### 3. CÁC SỰ KIỆN KHI ÔN (Jrun Management Console)

Jrun Management Console l<sup>à</sup> m<sup>ột</sup> ứng dụng Web nhằm giúp cho thao tác truy cập giao diện Web. Ở đây cung cấp các công cụ quản lý Jrun.

Để đăng nhập JMC như một người quản trị Web Server, ta cần thực hiện theo các bước sau:

Nhập địa chỉ URL : <http://localhost:8000>

+ Chọn Start -> Program -> Jrun3.1 -> Jrun Admin Server

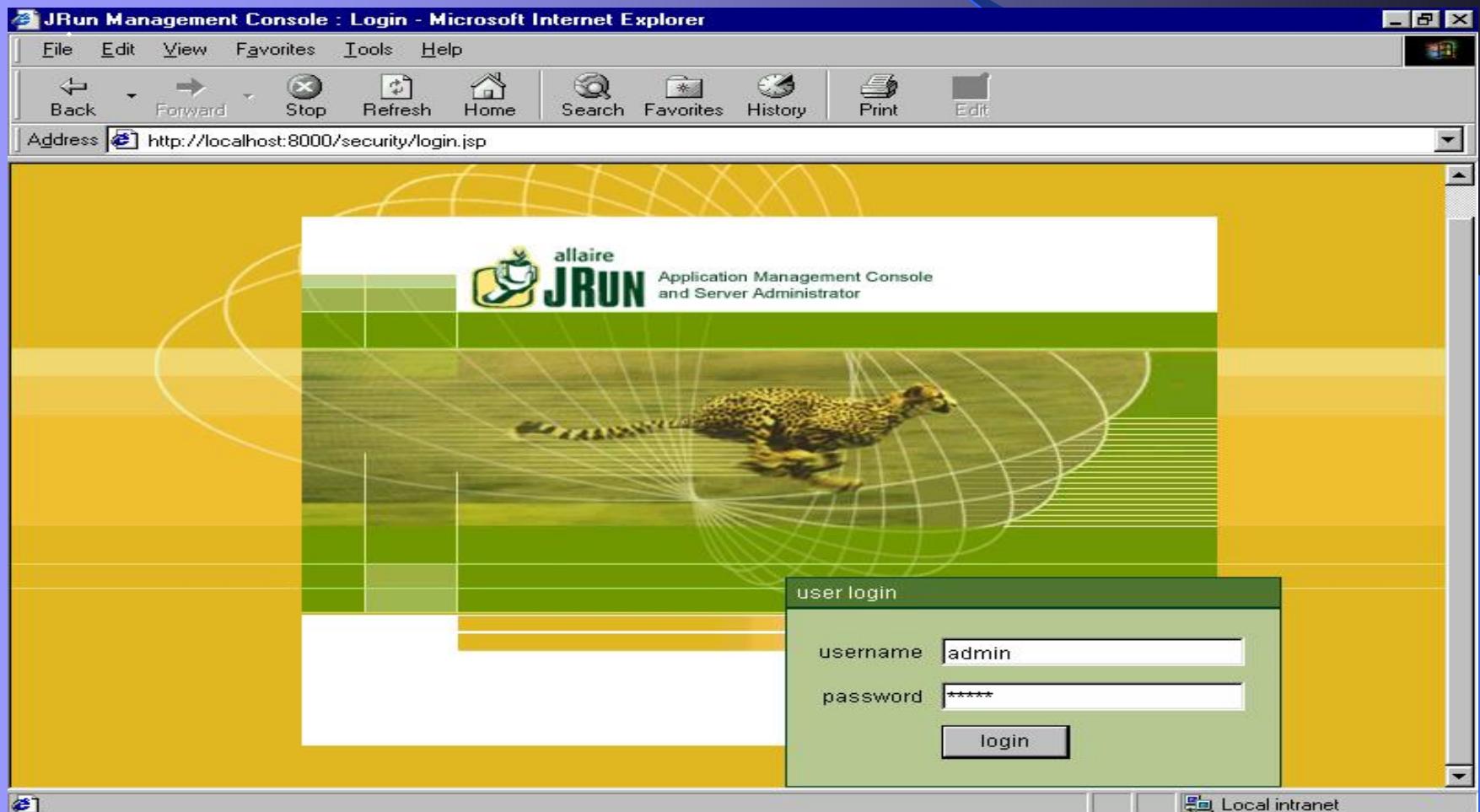
+ Chọn Start -> Program -> Jrun3.1 -> Jrun Management Console

Nếu muốn cài đặt Jrun sẽ hiển thị màn hình đăng nhập (login) yêu cầu b<sup>ất</sup> k<sup>ết</sup> nối cùng cung cấp tên người dùng (username) và mật khẩu (password) để đăng nhập vào phan quản trị (Administrator) của trình chủ.

# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## I. Cài Đặt và Cấu Hình Trình Chạy Web Server

### 3. Cài đặt và cấu hình (JRun Management Console)



# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## I. CẤU HÌNH TRÌNH CHỦ WEB SERVER

### 3. CÁC SỰ KIỆN KHI ÔN (JRun Management Console)

Sau khi đăng nhập, các thông tin quan trọng sẽ hiển thị

The screenshot shows the JRun Management Console running in Microsoft Internet Explorer. The title bar reads "JRun Management Console - Microsoft Internet Explorer". The address bar shows the URL "http://localhost:8000/index.jsp". The main content area displays the "Welcome to JRun" page for the "Developer Edition" (Version 3.1). The left pane is a tree view of the management console structure, including "JRun Admin Server", "JRun Default Server" (selected), "JDBC Data Sources", "Java Settings", "Log File Settings", "Application Hosts", "External Web Server", "JRun Web Server", "SSL", "Log Viewer", "Web Applications", and "Enterprise JavaBeans". The right pane contains two sections: "About JRun Management Console" which provides a brief overview of the tool's purpose and usage, and a larger area below it that lists documentation links like "Release Notes", "Product Documentation", "Quick Start Tour", and "Example Applications", as well as online resources such as "Product Registration", "Support Forums", "Knowledge Base", "Developer Center", and "Check for Updates".

# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## I. CẤU HÌNH TRÌNH CHỦ Web Server

### 3. Cấu hình JRun Management Console

Sau khi cài đặt, các thông tin quản trị sẽ hiển thị

The screenshot shows the JRun Management Console running in Microsoft Internet Explorer. The title bar reads "JRun Management Console - Microsoft Internet Explorer". The address bar shows the URL "http://localhost:8000/index.jsp". The main content area has a green header bar with the JRun logo and the text "Application Management Console and Server Administrator". Below this is a navigation menu with links: "(admin)", "welcome", "connector wizard", "password change", "manage JMC users", "serial number", "key search", and "logout". The left pane is a tree view of the JRun Admin Server structure, including "JRun Admin Server", "JRun Default Server" (selected), "JDBC Data Sources", "Java Settings", "Log File Settings", "Application Hosts", "External Web Server", "JRun Web Server", "SSL", "Log Viewer", "Web Applications", and "Enterprise JavaBeans". The right pane displays the "Welcome to JRun" page for the "Developer Edition" of JRun 3.1.15506. It includes links for "Documentation" (Release Notes, Product Documentation, Quick Start Tour, Example Applications), "Online Resources" (Product Registration, Support Forums, Knowledge Base, Developer Center, Check for Updates), and an "About JRun Management Console" section. The "About" section provides a brief overview of the console's purpose and usage.

JRun Management Console - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Print Edit

Address http://localhost:8000/index.jsp

**JRUN** Application Management Console and Server Administrator

(admin) welcome connector wizard password change manage JMC users serial number key search logout

ha

**JRun Admin Server**

**JRun Default Server**

- JDBC Data Sources
- Java Settings
- Log File Settings
- Application Hosts
- External Web Server
- JRun Web Server
- SSL
- Log Viewer
- Web Applications
- Enterprise JavaBeans

Welcome to JRun

**Developer Edition**

JRun 3.1  
Version 3.1.15506

**Documentation**

Release Notes  
Product Documentation  
Quick Start Tour  
Example Applications

**Online Resources**

Product Registration  
Support Forums  
Knowledge Base  
Developer Center  
Check for Updates

About JRun Management Console

This console serves as an HTML tool that enables you to configure various aspects of JRun.

This window contains two panes. The left pane provides a "tree" view of the JRun Management Console, starting from the root level. The right pane displays the contents of the folder or object currently selected in the tree. An object may be a function (e.g., logout) or a group of properties (e.g., MIME Filters).

To preview the contents of a folder in the left pane, click the + sign preceding the folder. To close an open folder in the left pane, click the - sign. To select an object, click the object. The right pane then displays the appropriate data. To edit data displayed, either click the Edit button or click a property; then make your changes in the editor window that appears. In most cases, you will need to restart JRun after any changes.

Local intranet

# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## I. CÊU HÌNH TRÌNH CHỈ WEB SERVER

### 3.1 CÊU HÌNH QUẢN TRỊ WEB SERVER

Thêng bình sô dông trang quản trị ®Ó quản lý, khéi ®éng, thay ®æi tham sè cho trình chỉ Default Web Server phôc vô øng dông Web.

#### 3.1.1 CÊU HÌNH JRUN Default Web Server

Ta chØ cÇn cÊu hÌnh theo c,c thÙng tin sau :

+ Web Server Address	*
+ Client IP filter	*
+ Web Server Port	8100
+ Idle thread timeout	300
+ Minimum thread count	1
+ Maximum active request	100
+ Maximum concurrent request	100
+ Jrun Web Server	on

# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## I. CÊU HÌNH TRÌNH CHỈN WEB SERVER

### 3.1 CÊU HÌNH QUẢN TRỊ WEB SERVER

#### 3.1.2 TÙ TỌ MÉT WEB SERVER

+ Nguội Jrun Default Server, b1n cã thÓ tù t1o mét Web Server kh,c cho riæng mænh

Lu ý : træn mét m,y cã thÓ cã nhiÖu Web Server ch1y cæng mét lóc miÔn sao chóng ®îc cÊu hÌnh vÍi sè hiÖu cæng kh,c nhau.

+ §Ó t1o Web Server kh,c, b1n chæn nh,nh gèc bæn khung trang tr,i. NhÊn nót Add a server bæn khung trang ph¶i vµ ®Æt tæn cho Web Server còng nh c,c thæng tin cÊu hÌnh kh,c mµ Jrun yæu cÇu

# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## I. CÊU HÌNH TRÌNH CHỦ Web Server

### 3.1 CÊU HÌNH QUẢN TRỊ Web Server

#### 3.1.2 Tù t'ò mét Web Server cña ri'ng b'ìn

The screenshot shows the JRun Management Console interface within a Microsoft Internet Explorer window. The title bar reads "JRun Management Console - Microsoft Internet Explorer". The address bar shows "http://localhost:8000/index.jsp". The main content area displays the "Managing JRun Servers" page. On the left, there's a sidebar with icons for "JRun Admin Server" and "JRun Default Server". The main panel has a heading "JRun > Managing JRun Servers" and a descriptive text about managing servers. A table lists two servers:

	Display Name	Root Directory	Status	Start/Restart
	JRun Admin Server	C:/Program Files/Allaire/JRun/servers/admin	running	<input type="button" value="restart"/>
	JRun Default Server	C:/Program Files/Allaire/JRun/servers/default	running	<input type="button" value="restart"/>

At the bottom of the main panel, there's a button labeled "add a server". The status bar at the bottom of the browser window says "Done, but with errors on page." and "Local intranet".

# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## I. CẤU HÌNH TRÌNH CHỦ Web Server

### 3.2 CẤU HÌNH VỤ TỰ ÔNG DÔNG WEB (Web Application)

#### 3.2.1 Tạo mới ông dông Web

Thường thêng khi tạo mới ông dông Web, ta cần phải cấu hình ông dông này trong IIS Web Server. Sau khi tạo mới ông dông trên Web Server của IIS, ta có thể cấu hình theo các bước sau:

+ Trước khi cấu hình IIS, ta cần chú ý rằng phải có IP của máy chủ i IIS đang chạy. Nếu không có tên miền, ta cần thử số port localhost hay 127.0.0.1 để rằng có thể kết nối.

# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## I. CẤU HÌNH TRÌNH CHỦ WEB SERVER

### 3.2 CẤU HÌNH VỊ TRÍ ỞNG DÔNG WEB (Web Application)

#### 3.2.1 Tạo mới ứng dụng Web

+ Chẩn Jrun Web Server (Admin Server hay Default Server)

+ Bấm vào link Create Application

+ Chẩn Jrun Admin Server hay Jrun Default Server.

+ Nhập tên của ứng dụng. Ví dụ : DATN

+ Nhập vào địa chỉ URL cho ứng dụng.

+ Chẩn thô mục mà ta đặt các trang JSP của ứng dụng.

Ví dụ: C:/DATN/Jsp | Ưu thô mục chứa các trang JSP của ứng dụng.

+ Sau khi nhấn thong tin ta nhận được là Create.

+ Tiếp theo khai báo ứng dụng Web Server với tên mà ta đã đặt ở bước trước (Jrun Default Server).

+ Cuối cùng ta thử trên trình duyệt web để kiểm tra xem có thể truy cập không:

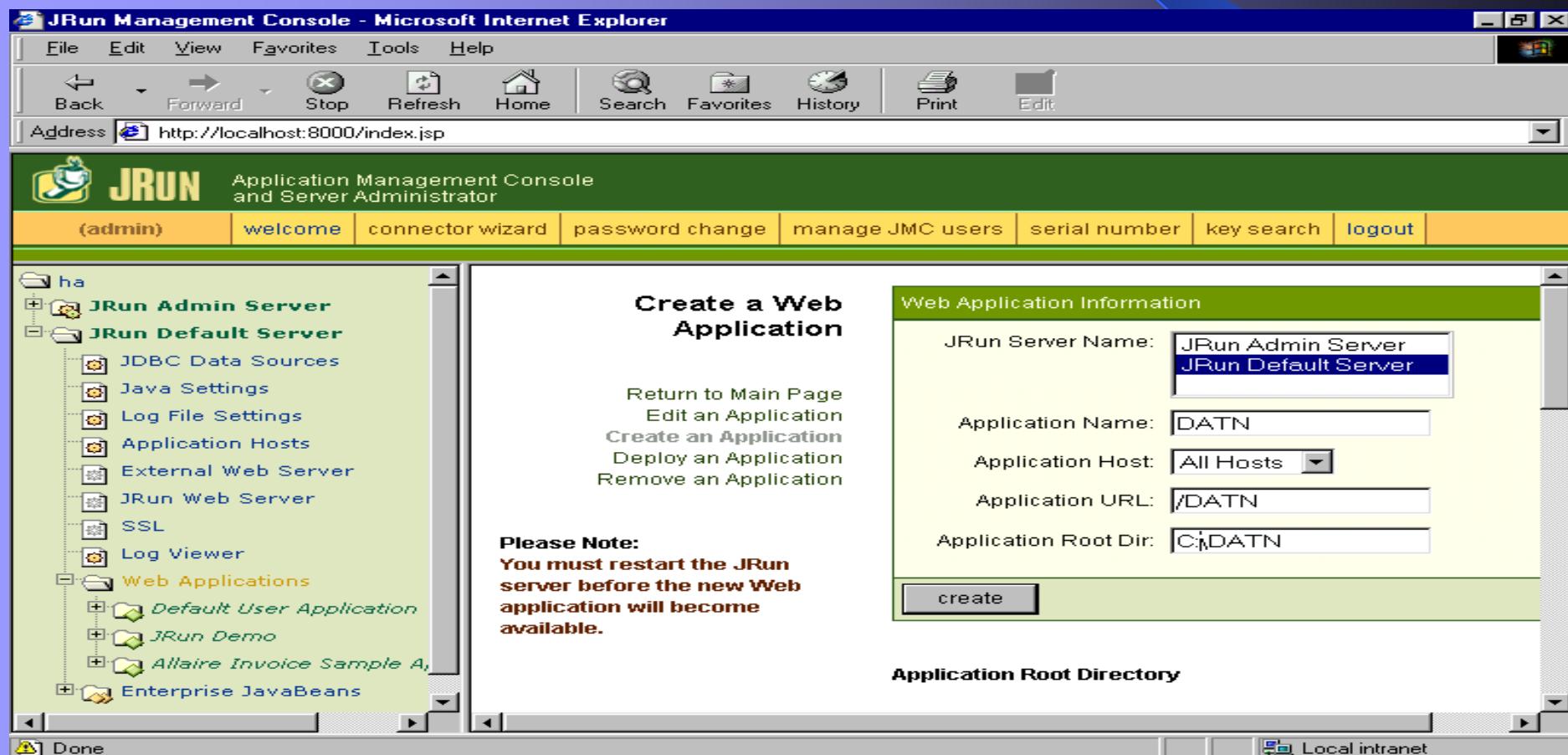
<http://localhost:8100/DATN/Jsp/index.jsp>

# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## I. Cài Đặt và Cấu Hình Trình Chạy Web Server

### 3.2 Cấu hình vựng trống Web (Web Application)

#### 3.2.1 Tạo mới vựng Web



# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## I. CÊU HXNH TRXNH CHÑ WEB SERVER

### 3.2 CÊU HXNH VËT TÔNG DÔNG WEB (Web Application)

#### 3.2.2 XÃA MÉT TÔNG DÔNG WEB

Do nhu cÇu sö dông ®«i khi b¹n t¹o ra mét øng dông web, nhng cã thÓ sau mét thêi gian sö dông b¹n kh«ng cÇn dïng ®Õn nã n÷a, ta cã thÓ xää nã ®i. §Ó xää mét øng dông web:

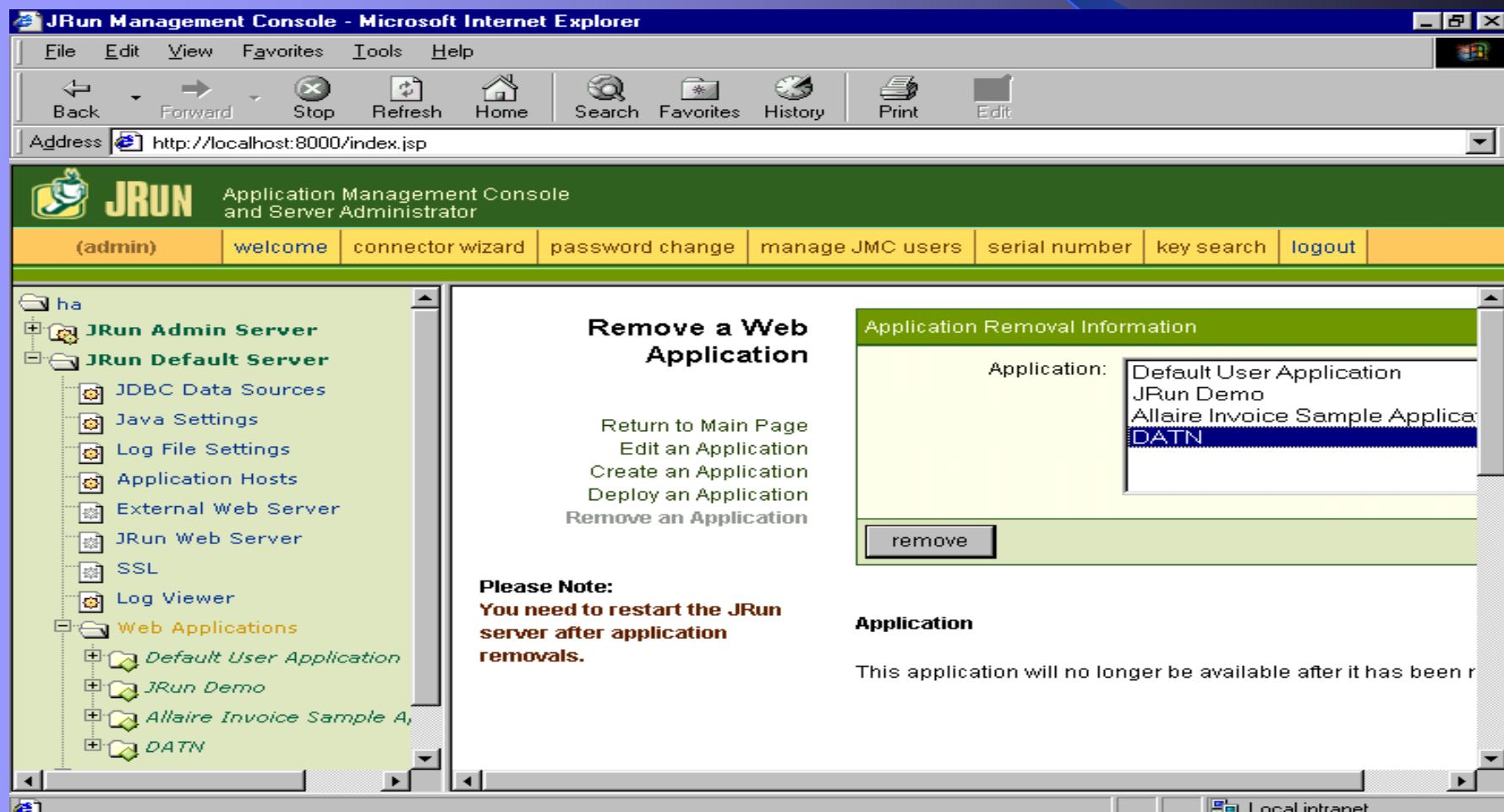
- + Chän Web Application tõ mµn h×nh qu¶n trÞ cña Jrun
- + TiÕp ®Õn chän Web Application. Jrun sї hiÓn thÞ danh s,ch c,c øng dông ta ®· ®“ng ký hoÆc t¹o ra tríc ®ã.
- + §Ó lo¹i bá øng dông Web, ta chän môc Remove Application.
- + Khi Jrun hiÓn thÞ danh s,ch c,c ®Ò môc cÇn xää, b¹n chän tªn cña øng dông rãi chän Remove.
- + §Ó Jrun nhËn ra øng dông ®· bÞ xää hoµn toµn, cÇn khëi ®éng l¹i trxnh chñ Jrun Web Server.

# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## I. Cài Đặt và Cấu Hình Trình Chạy Web Server

### 3.2 Cài Đặt và Cấu Hình Ứng Dụng Web (Web Application)

#### 3.2.2 Xóa Một Ứng Dụng Web



# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## I. CÊU HXNH TRXNH CHÑ WEB SERVER

### 3.2 CÊU HXNH VËT TÔNG DÔNG WEB (Web Application)

#### 3.2.3 CËP NHËT MËT TÔNG DÔNG WEB

Sau khi t¹o ra mét tông dông Web nÕu cÇn bæ sung hoÆc thay ®æi, ta cã thÓ cËp nhËt l¹i néi dung tông dông theo c,c bíc sau:

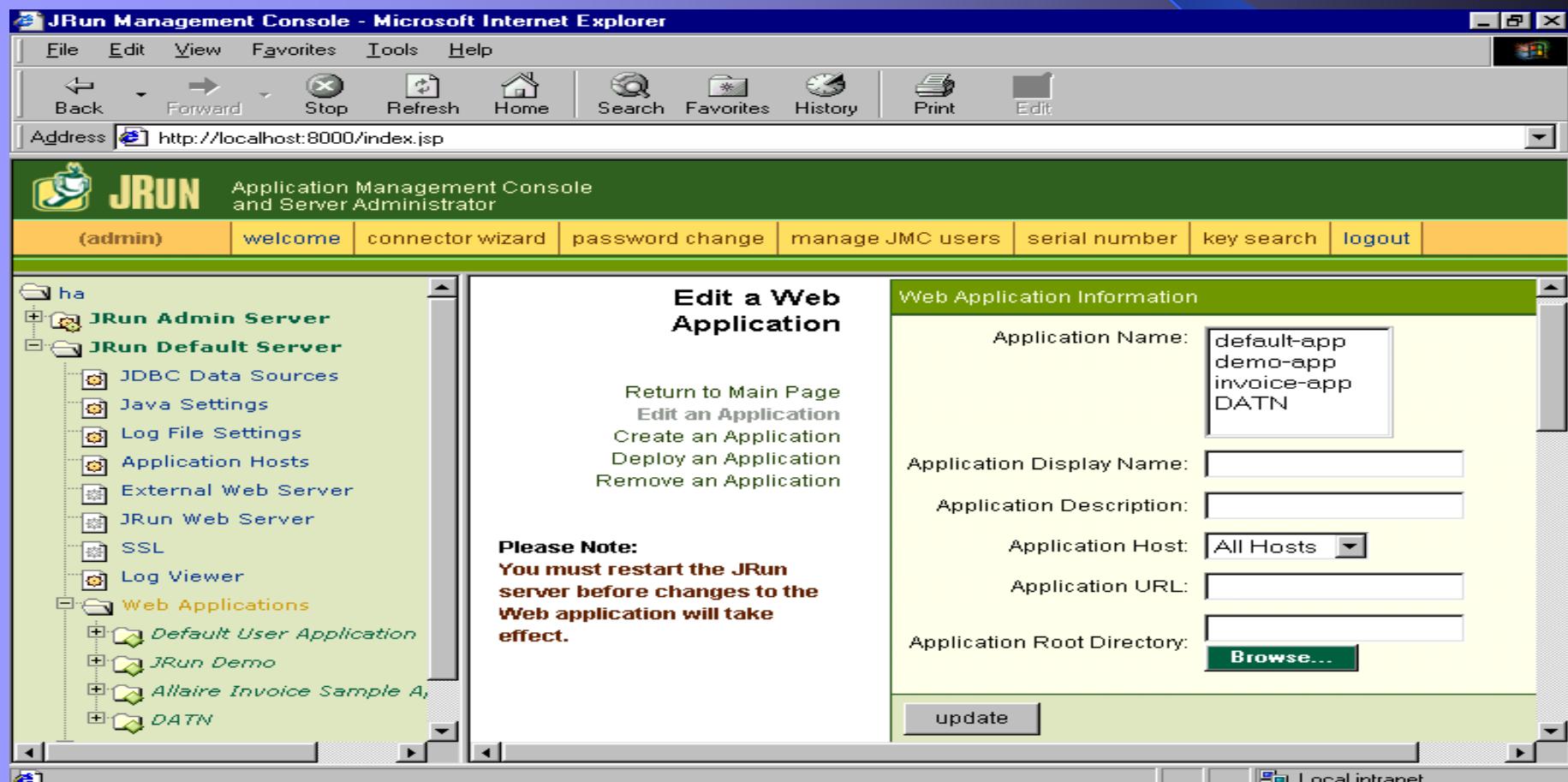
- + Chän Web Application tõ mµn h×nh qu¶n trÞ cña Jrun Web Server.
- + Jrun sї hiÓn thÞ danh s, ch c,c tông dông ®· ®· ng ký tríc ®ã.
- + Chän môc Edit Application.
- + Khi Jrun hiÓn thÞ danh s, ch tông dông, ta cÇn chän lÊy tông dông muèn cËp nhËt.
- + Th«ng tin tông dông sї hiÓn ra ®Ó ta bæ sung hoÆc söa ®æi.
- + CÇn ph¶i khëi ®éng l¹i trxnh chñ Jrun Web Server.

# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## I. Cài Đặt và Cấu Hình Trình Chạy Web Server

### 3.2 Cài Đặt và Cấu Hình Vụ Tạo Ông Dòng Web (Web Application)

#### 3.2.3 Cập Nhật Mật Ông Dòng Web



# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## I. CÊU HÌNH TRÌNH CHỈN WEB SERVER

### 4. TÍO VỤ QUẢN LÝ USER TRONG JRUN WEB SERVER

ŞÈI VÍI JRUN WEB SERVER TA CÃ THÔ RA MÉT SÈ THÙNH VIÂN GIỮI HÌN CHO PHĐP TRUY CẤP TRÌNH CHỈN WEB SERVER. TA CÃ THÔ THÙC HIÖN THEO CÁC BÍC SAU :

- + CHÄN MÔC MANAGE JMC USERS TÖ TRANG QUẢN TRỊ CÑA JRUN.
- + NHËP TÀN USER VỤ PASSWORD CÑA THÙNH VIÂN ®Ã. MÇI IÇN TA CHØ CÇN ®ÜNG KÝ CHO HAI THÙNH VIÂN.
- + TA CHÄN NÓT UPDATE JMC UESRS ®Ó CẤP NHËT USER VỤO C¬ SË D÷ LIÖU CÑA JRUN.

# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## I. Cài Đặt và Cấu Hình Trình Chạy Web Server

### 4. Tạo Vụ Quản Lý User Trong JRun Web Server

The screenshot shows the JRun Management Console interface within Microsoft Internet Explorer. The title bar reads "JRun Management Console - Microsoft Internet Explorer". The address bar shows "http://localhost:8000/index.jsp". The main content area displays the "Manage JMC Users" page under the "welcome" tab. On the left, a navigation tree includes "JRun Admin Server", "JRun Default Server", "Web Applications" (with "Default User Application", "JRun Demo", "Allaire Invoice Sample A", and "DATN" listed), and "Log Viewer". The central panel contains instructions for managing user passwords and permissions, and a note about leading/trailing spaces in user names. To the right is a table titled "Currently Available JMC Users" with two rows. Each row has a checkbox, a "User Name" field, a "Password" field, and a "Permissions" dropdown menu containing "All servers", "JRun Admin Server", and "JRun Default Server". A "Update JMC Users" button is at the bottom of the table.

	User Name	Password	Permissions
<input type="checkbox"/>			All servers JRun Admin Server JRun Default Server
<input type="checkbox"/>			All servers JRun Admin Server JRun Default Server

# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## I. CẤU HÌNH TRÌNH CHỦ WEB SERVER

### 5. Kiểm tra lỗi thông qua Log Files

Thông thường bắt đầu khi lỗi xảy ra và sau đó Jrun sẽ lưu lại file logs. Những file logs này nằm trong thư mục [Jrun\_Home]\logs.

Mỗi file log lưu lại những sự kiện phát sinh khác nhau. Thông thường có 3 file log sau :

+ default-err.log : file này lưu lại những lỗi của Jrun xảy ra trong quá trình đột

+ default-even.log : lưu lại những lỗi trong quá trình thực hiện truy xuất tài nguyên của Java.

+ default-out.log : lưu lại những lỗi liên quan đến môi trường bên ngoài. Ví dụ : Khi bạn gọi thông tin về một пара không cần lỗi sẽ được ghi vào default-out.log.

- Tóm lại: Số lượng lỗi theo dõi quá trình viễn thông của Jrun, ta có thể dùng file log để xác định lỗi của hệ thống.

# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## II. X<sup>©</sup>y Dùng C- Ch<sup>Ô</sup> Connection Poll

- + Nhu cÇu vÒ kÕt nèi c¬ së d÷ liÖu trung t@m ®Ó truy vÊn d÷ liÖu trong c,c øng dông cña Web nguy cµng t¬ng.
  - + C,c hÖ c¬ së d÷ liÖu ph¶i tiªu tèn rÊt nhiÖu tui nguyªn ®Ó duy tr× liªn tôc kÕt nèi trong qu, tr×nh truy vÊn.
  - + Ta chØ nªn më kÕt nèi truy vÊn ®Õn c¬ së d÷ liÖu chØ khi nµo thËt sù cÇn thiÕt. Mét khi ®· hoµn tÊt truy xuÊt, b¹n ph¶i tr¶ l¹i kÕt nèi cho ngêi dïng kh,c truy cËp. MÆc dï vËy, mçi mét kÕt nèi ®Õn c¬ së d÷ liÖu ®Òu ph¶i ®ßi hái cã thêi gian thiÕt lËp, ph@n bæ tui nguyªn,...
  - + C,c tr×nh chñ Web server ®a ra c¬ chØ POLL cho phÐp c,c yªu cÇu vÒ kÕt nèi ®Õn c¬ së d÷ liÖu chØ ®îc më mét lÇn vµ dïng chung bëi c,c trang JSP víi nhau.

# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## II. XỬ LÝ DÙNG CẢ CHỖ Connection Pool

### 1. Giải thích:

- + Vì vậy, cần yêu cầu của ứng dụng truy vấn để lấy một kết nối trống ra trong thời gian ngắn
- + Một trang JSP trích khi truy xuất để lấy số yêu cầu của ứng dụng để cùng một kết nối nội bộ thèng số lần tra xem kết nối nội bộ. Vì thế kết nối nội bộ có thể là tệp XML hoặc kết nối nội bộ.
- + Trang JSP sau khi số đồng xong không cần kết nối nội bộ mà trả lời kết nối nội bộ cho nội bộ thèng. Nội bộ thèng số lần kết nối nội bộ trả lời ngay lập tức không cần chờ đợi.
- + Trong một trang JSP nếu phiền lạm vi ứng dụng khác, nó có thể truy xuất cả số để trả lời kết nối nội bộ, nội bộ thèng số cùng một kết nối nội bộ. Vì thế kết nối nội bộ có thể là tệp XML hoặc kết nối nội bộ.

# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## II. XÓy DÙNG CÁC CHỖ Connection Pool

### 1. Giới thiệu :

Ta sẽ cung cấp một cách để chia sẻ kết nối cho các pool (cách chia sẻ kết nối). Trong đó có hai cách truy cập vào pool: yêu cầu một kết nối (getConnection()) và trả lại kết nối (releaseConnection()). Khai báo giao diện mang tên IconnectionPool sẽ cung cấp thông tin này.

```
// Interface định nghĩa cách truy cập vào pool cung cấp chia sẻ kết nối
public interface IconnectionPool
{
    public Connection getConnection() throws SQLException;
    public void releaseConnection(Connection conn) throws SQLException;
}
```

Ta có thể sử dụng Pool trong cách để cung cấp kết nối cho ứng dụng. Khi tạo ra một pool, ta chỉ cần khai báo cách truy cập vào pool để lấy kết nối. Sau đó, ta có thể truy cập vào pool để lấy kết nối và trả kết nối.

# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## II. XÓy DÙNG CÁC CHỖ CONNECTION POOL

### 2. CÁC CHỖ POOL HỖ TRỢ BỘ WEB SERVER

- + Hiện nay hầu hết các trình ch‰n Web server ®®n cung cấp sẵn các chỗ Pool gäi lµ Data Source.
- + Vì Öc qu¶n lý vµ ®®n hµnh Pool sї do trình ch‰n Web server ®®m nhiÖm vµ b¹n chØ cÇn cung cấp cho trình ch‰n nh÷ng th«ng tin t¹o Pool lµ ®ñ.
- + Chóng ta sї cÊu h×nh vµ sô dông các chỗ Pool hỗ tr¢i bëi trình ch‰n Jrun. Tríc h‰t ta mµ trang <http://localhost:8100/> ®®n vµo trang quản lý tr¢ cña Jrun. TiÕp ®®n ta chæn Default Jrun Server\JDBC Data Source. NhÊn nót Add trang cung cấp th«ng tin ®®n cÊu h×nh cho DataSource hiÓn tr¢ sau :

# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## II. Xây Dùng Cú Chỗ Connection Pool

### 2. Cú Chỗ Pool Hỗ Trợ Bép Web Server

The screenshot shows the JRun Management Console interface in Microsoft Internet Explorer. The title bar reads "JRun Management Console - Microsoft Internet Explorer". The address bar shows "http://localhost:8000/index.jsp?". The main content area displays the "JRun Default Server > JDBC Data Sources" page. On the left, there is a navigation tree with nodes like "JRun Admin Server", "JRun Default Server" (selected), "JDBC Data Sources" (selected), "Java Settings", "Log File Settings", "Application Hosts", "External Web Server", "JRun Web Server", "SSL", "Log Viewer", "Web Applications", and "Enterprise JavaBeans". The right panel shows a table of currently available JDBC data sources:

	Name	Driver	URL
	TaglibDataSource	com.pointbase.jdbc.jdbcUniversalDriver	jdbc:pointbase://embedded/sample
	Sample9bDataSource	com.pointbase.jdbc.jdbcUniversalDriver	jdbc:pointbase://embedded/sample
	KDPHUYENC	allaire.jrun.jdbc.JRunDriver	jdbc:jrun:sqlserver://127.0.0.1:1433;date
	forum	allaire.jrun.jdbc.JRunDriver	jdbc:jrun:sqlserver://127.0.0.1:1433;date
	Baomat	sun.jdbc.odbc.JdbcOdbcDriver	jdbc:odbc:Baomat

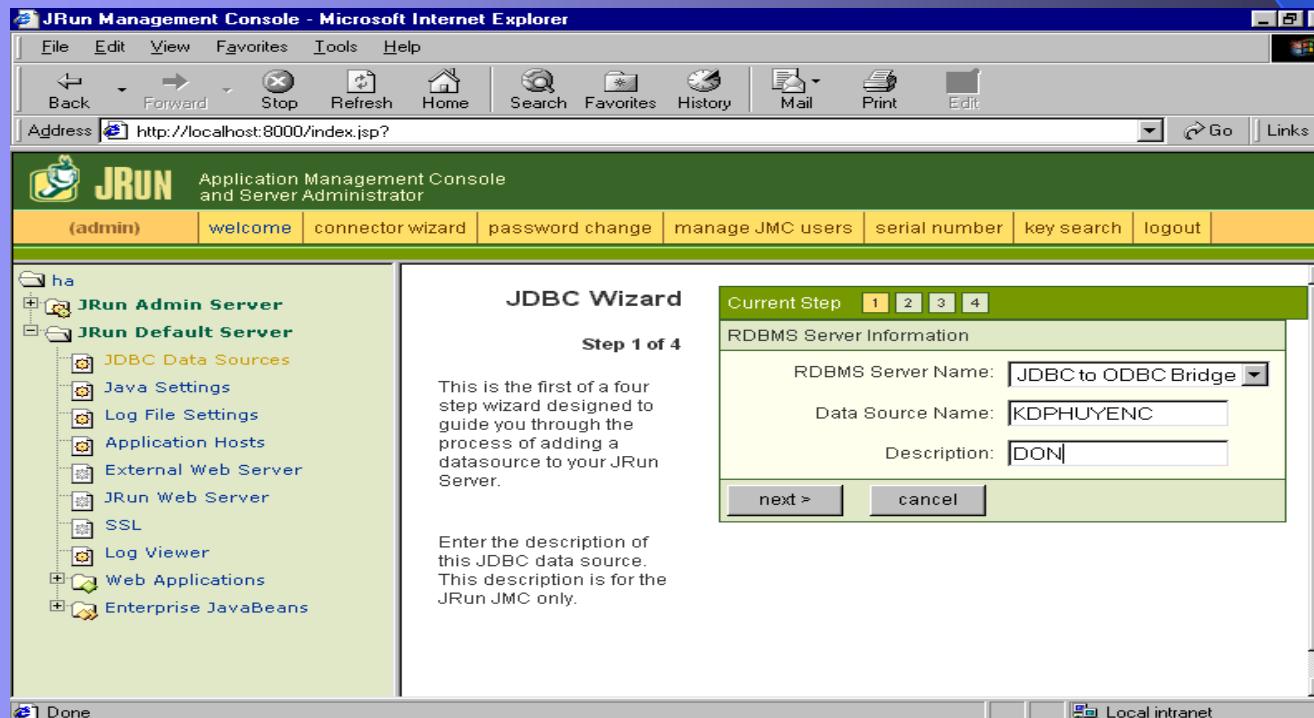
A "add" button is located at the bottom of the table. At the very bottom of the page, there are links for "Done" and "Local intranet".

# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## II. XÂY DÙNG CƠ CHẾ Connection Pool

### 2. CƠ CHẾ Pool Hệ Trí Bộ Web Server

+ JRun cho phép số lượng lớn kết nối JDBC. Trong trường hợp này ta cần mô tả cách JDBC to ODBC Bridge. Mô tả Data Source Name ta gán vào tên *CSDL nguồn*. Sau đó sẽ chỉ định cách tham chiếu Pool sau này.

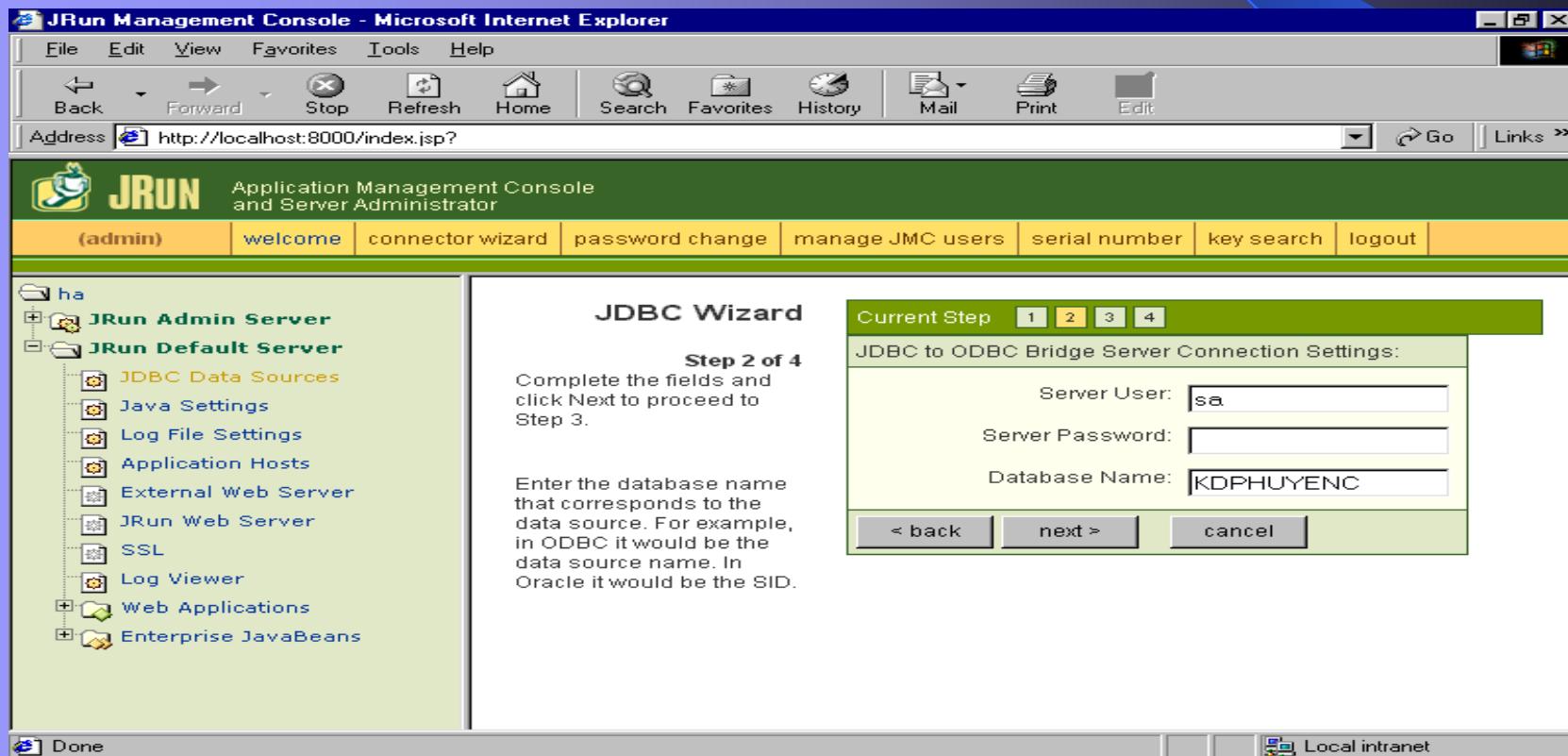


# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## II. X@y Dùng C¬ ChÕ Connection Pool

### 2. C¬ ChÕ Pool Hç Trî Bëi Web Server

TiÕp tõc ta nhËp vµo c,c th«ng tin liªn quan ®Õn chuçi kÕt nèi c¬ së d÷ liÖu.

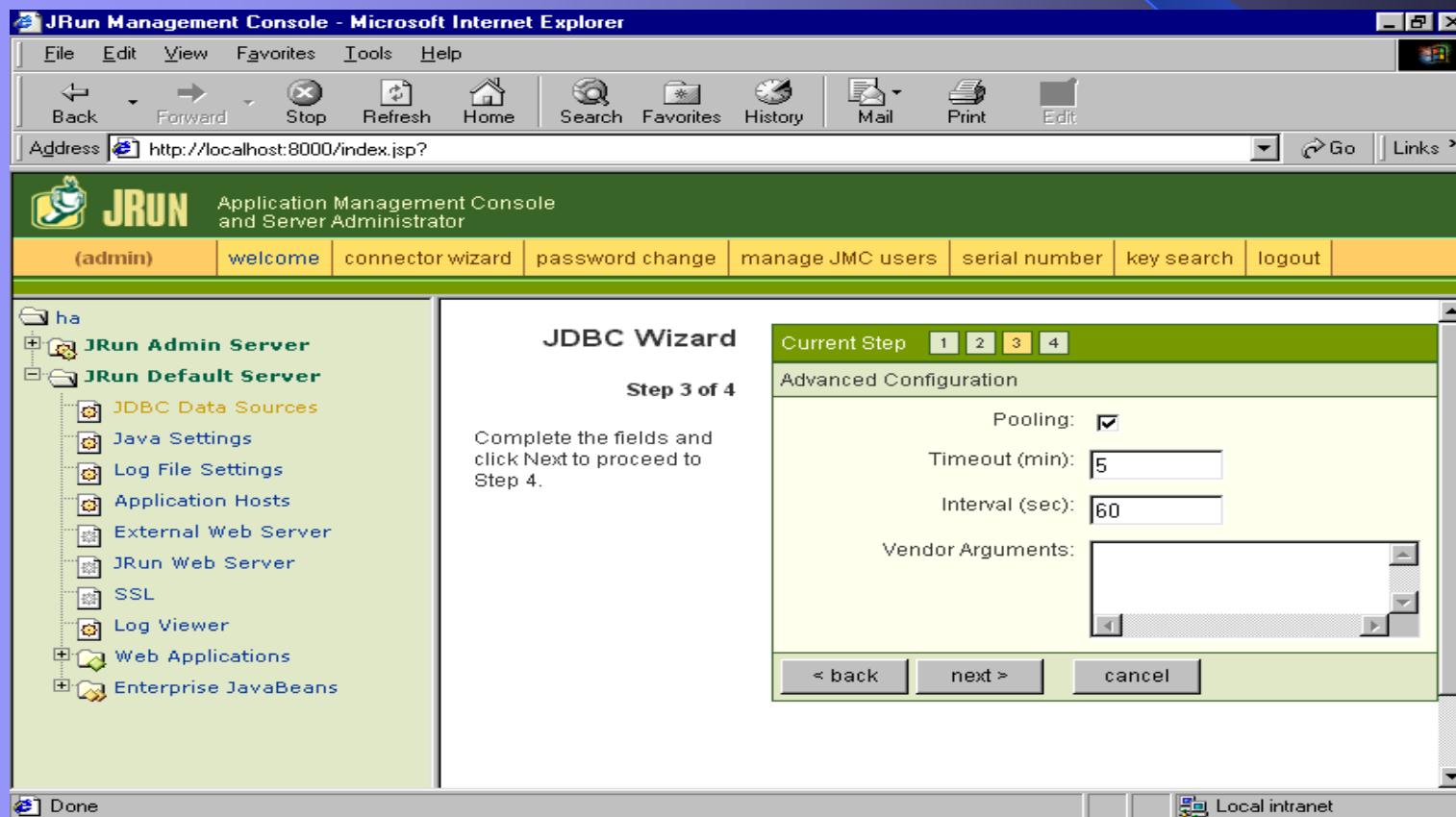


# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## II. X@y Dùng C¬ ChÕ Connection Pool

### 2. C¬ ChÕ Pool Hç Trî Bëi Web Server

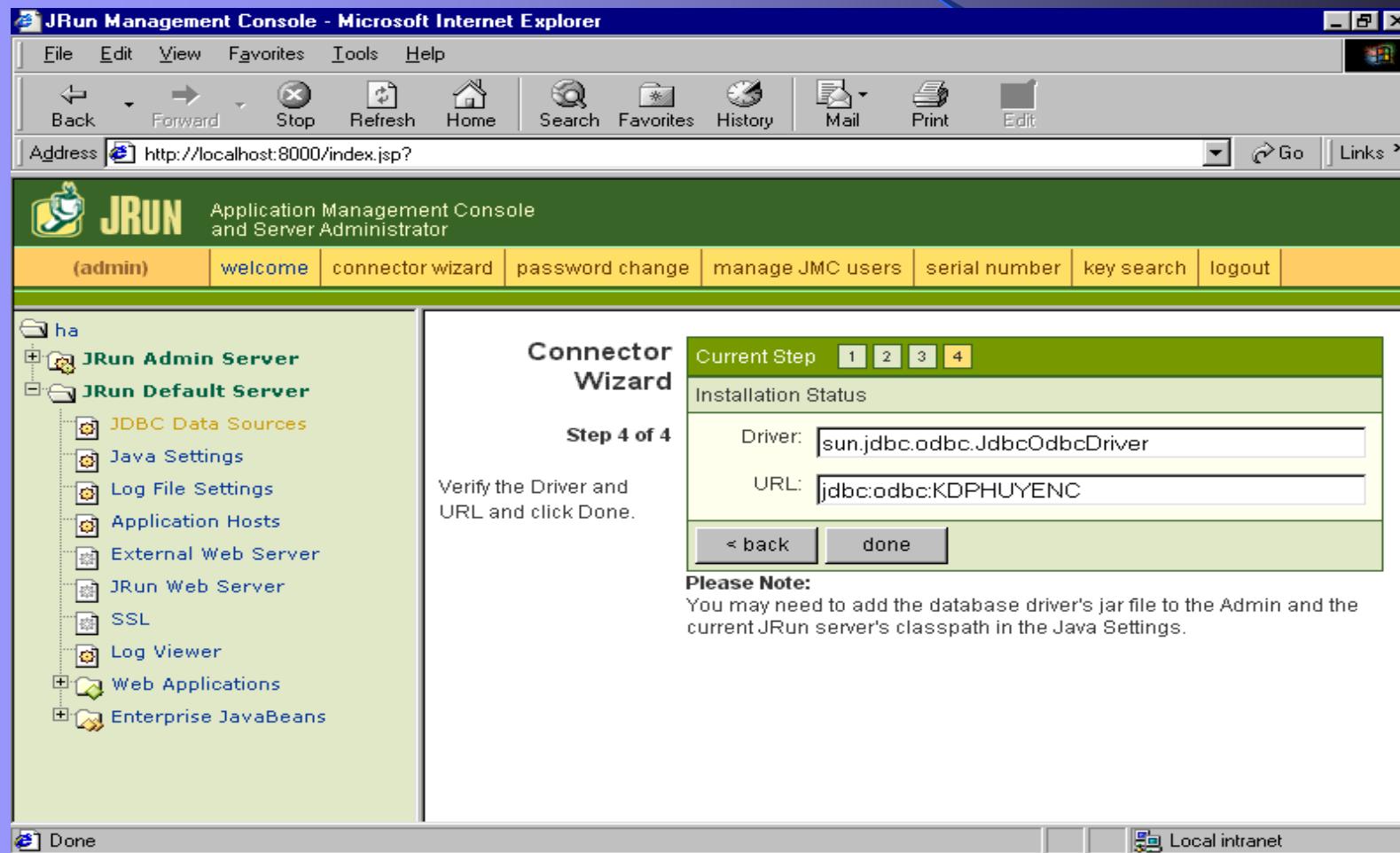
TiÕp tôc nhÊn Next Jrun sї hoµn tÊt nh÷ng thao t,c cßn l¹i.



# CÀI ĐẶT VÀ CẤU HÌNH WEB SERVER

## II. Xây Dùng Cửa Chắc Connection Pool

### 2. Cửa Chắc Pool Hệ Trí Bép Web Server



# Đề tài kết thúc học phần

## Tạo Website theo các chủ đề sau :

- 1. Soạn trang web cá nhân của nhóm bao gồm: Họ và tên, Công việc và chức vụ hiện nay, Địa chỉ liên lạc, Điện thoại, Fax, Email, Mobile Phone, ... Ngoài ra các nhóm có thể bổ sung các thông tin khác.
- 2. Website quản lý quảng cáo
- 3. Website báo điện tử
- 4. Website dịch vụ giải trí như ECards, Điện hoa, ...
- 5. Website giới thiệu âm nhạc
- 6. Website bán hàng
- 7. Website giới thiệu khu di tích Hội An .
- 8. Xây dựng diễn đàn trên mạng
- 9. E-banking
- 10. Website Fastfood Online

# **Đề tài thi kết thúc học phần**

## **Tạo Website theo các chủ đề sau :**

- 11. Website quản lý thuê trả phòng khách sạn
- 12. Website quản lý việc cấp phát thuốc
- 13. Website quản lý giường bệnh
- 14. Website quản lý bệnh án
- 15. Website quản lý lịch chạy tàu
- 16. Website quản lý nhà hàng
- 17. Website quản lý khu di tích
- 18. Website quản lý thuê bao Internet
- 19. Website quản lý dịch vụ 1080
- 20. Website quản lý danh bạ điện thoại

# Đề tài thi kết thúc học phần

Tạo Website theo các chủ đề sau :

- 21. Website giới thiệu trường học
- 22. Website cho sinh viên thi trắc nghiệm
- 23. Website giáo trình điện tử môn học bất kỳ
- 24. Website quản lý lịch thực hành phòng máy
- 25. Website hỗ trợ công tác đào tạo của trường đại học
- 26. Website phục vụ công tác Đoàn trường đại học
- 27. Website quản lý ký túc xá
- 28. Website quản lý tour du lịch
- 29. Website về dịch vụ việc làm
- 30. Website về dịch vụ nhà đất
- 31. Website quản lý cước phí
- 32. Website giới thiệu, hướng dẫn sử dụng ĐTDD

