

LẬP TRÌNH WEB BẰNG JAVASCRIPT



V. Các Thành phần cơ bản của JavaScript

1. Cách sử dụng biến và hàm trong JavaScript

a. Cách khai báo biến để lưu trữ thông tin:

Để khai báo biến nào đó bạn sử dụng từ khoá **var** ở đầu và không cần khai báo kiểu dữ liệu.

Cú pháp: `<var varname;>` hoặc `<var varname=value;>`

Ví dụ: `var str;` hoặc `var num=60;`

Tên biến được tạo thành từ các chữ cái, chữ số và bắt đầu là một chữ cái, JavaScript phân biệt chữ hoa và chữ thường.

Ta có thể dùng các toán tử +, -, *, / và phép gán để thao tác trên các biến của nó:

Ví dụ: `var a=55;`
`var b=a*2+10;`

b. Để hiển thị kết quả ra trang html, ta có thể dùng hàm write()

Cách viết: `window.document.write(<nội dung cần viết>);`

<nội dung cần viết> có thể là một biến, một biểu thức hay một chuỗi.

c. Chuỗi

Phép gán chuỗi: `var str="welcom to javascript";`

Phép nối chuỗi, ta sử dụng toán tử cộng +

d. Hàm alert();

Hàm này dùng để đưa ra một thông báo với nội dung được chỉ định trong thân của hàm:

Ví dụ: `alert("Đặt nội dung của bạn tại đây");`

e. Hàm prompt();

Hàm này yêu cầu người nhập nhập vào một số thông tin nào đó sau đó sẽ gán những gì người dùng nhập vào cho một biến.

Cú pháp: `prompt(<mess>,<tempinput>);`

<mess> xuất ra dòng thông báo

<tempinput> dữ liệu nhập mẫu

Ví dụ: `var name = prompt("what is your name? ", "input your name");`

Sau khi bạn nhập vào từ bàn phím, giá trị sẽ được lưu vào biến name.

f. Các hàm và phương thức về ngày tháng;

Trong JavaScript trang bị cho ta một đối tượng để lấy lại ngày giờ hệ thống của máy tính đó là đối tượng Date();

Cách khai báo như sau:

`var date = new Date();`

Sau đó ta có thể sử dụng những phương thức của đối tượng này để lấy lại ngày, tháng, năm hay giờ, phút, giây của thời gian trong hệ thống máy tính.

Một số phương thức của đối tượng này:

Phương thức	Mô tả
-------------	-------

getDate()	Trả về ngày của tháng 1..31
getDay()	Trả về ngày trong tuần 0:chủ nhật; 1:thứ hai...
getHours()	Trả về giờ trong ngày 0..23
getMinutes()	Trả về phút 0..59
getSeconds()	Trả về giây 0..59
getTime()	Trả về thời gian tính tính theo mili giây, tính từ 0:0:1-1-1970
getYear()	Trả về năm
getMonth()	Trả về tháng 0..11

Ví dụ:

```

var now = new Date();
var d = now.getDate();
var m = now.getMonth()+1;
var y = now.getYear();
var day=d+"/"+m+ "/" +y;
window.document.write(day);

```

Bài tập củng cố

1. Trong javascript các biến phải được khai báo kiểu dữ liệu của nó			
<input type="radio"/>	Đúng	<input type="radio"/>	Sai
2. Một hằng xâu kí tự phải được đặt giữa cặp dấu ngoặc sau:			
<input type="radio"/>	ngoặc kép ""	<input type="radio"/>	ngoặc đơn "
<input type="radio"/>			cả hai
3. Hàm prompt có bao nhiêu đối số			
<input type="radio"/>	2 đối số	<input type="radio"/>	3 đối số
<input type="radio"/>			không giới hạn đối số
4. Phương thức getDay() của đối tượng kiểu ngày Date trả về giá trị thuộc phạm vi			
<input type="radio"/>	Từ 0 đến 7	<input type="radio"/>	Từ 0 đến 30
<input type="radio"/>			Cả hai đều sai
5. Đối tượng kiểu Date trả về ngày giờ hệ thống của:			
<input type="radio"/>	Máy khách	<input type="radio"/>	Máy chủ

Xem kết quả

2. Các cấu trúc điều khiển trong javascript

a. Chuyển hướng người dùng đến trang web khác

Để chuyển hướng người sử dụng đến một trang web khác bạn có thể sử dụng như sau:

Cú pháp `window.location =URL;`

URL: là một tài nguyên, một trang web trên mạng hay máy của bạn.

Ví dụ: `window.location = "http://www.vnn.vn";`

b. Cấu trúc câu lệnh if-then

Đây là một câu lệnh điều kiện trong ngôn ngữ javascript, giống như trong các ngôn ngữ khác, cú pháp của câu lệnh này như sau:

```
if(<điều kiện>
    { <nhóm lệnh 1;> }
    [else{ <nhóm lệnh 2;> }]
```

Nếu điều kiện nhận giá trị **true** thì sẽ thực hiện **<nhóm lệnh 1>** còn không sẽ thực hiện **<nhóm lệnh 2 >**.

Chú ý: + Nếu sử dụng phép so sánh bằng trong biểu thức điều kiện bạn phải dùng hai dấu bằng(==) như trong ngôn ngữ java.

+ Có thể có hay không có **else** đối với câu lệnh **if**.

+ Có thể có các **if – else** lồng nhau.

c. Giá trị boolean và các phép so sánh

các phép toán so sánh trả về giá trị kiểu boolean:

Phép toán	Ý nghĩa
>	Lớn hơn
<	Nhỏ hơn
==	So sánh bằng
>=	Lớn hơn hay bằng
<=	Nhỏ hơn hay bằng
!=	Khác nhau

d. Toán tử or, and và mod

Toán tử **or** được ký hiệu bởi 2 dấu || thể hiện phép **hoặc** thường sử dụng trong biểu thức điều kiện.

Toán tử **and** được ký hiệu bởi 2 dấu && thể hiện phép **và** thường sử dụng trong biểu thức điều kiện.

Toán tử **mod** được ký hiệu bởi dấu % để lấy kết quả phần dư của phép chia.

Chú ý: Giống như java, để gộp nhiều câu lệnh đơn lại thành một câu lệnh ghép ta có thể sử dụng cặp dấu ngoặc móc {}.

3. Các sự kiện xử lý trên trang html của bạn

Phần này nói một số thủ thuật xử lý kích hoạt các sự kiện về mouse như: di chuyển mouse trên đối tượng, click mouse vào đối tượng...

a. **Sự kiện onClick():** Sự kiện này được xảy ra khi bạn kích chuột phải vào đối

tượng.

Ví dụ: ``

hay

`<input type="submit" onClick="javascript:closeWindow()">`

với hàm **closeWindow()** được viết như sau:

```
function closeWindow()
{
    top.window.close();
}
```

b. Sự kiện onMouseOver(): sự kiện này xảy ra khi ta di chuyển chuột lên một đối tượng liên kết nào đó.

c. Sự kiện onMouseOut(): sự kiện này xảy ra khi ta di chuyển chuột ra ngoài một đối tượng liên kết nào đó.

d. Sự kiện ondblclick() Sự kiện này được xảy ra khi bạn kích đôi chuột phải vào đối tượng.

e. Sự kiện onmousedown() Sự kiện này được xảy ra khi chuột phải được nhấn xuống trên đối tượng.

f. Sự kiện onmouseup() Sự kiện này được xảy ra khi chuột phải được nhả ra trên đối tượng.

Chú ý: Các Sự kiện *onkeypress()*, *onkeydown()*, *onkeyup()* được xảy ra khi ta nhấn một phím, nhấn xuống hay nhả phím khi ta chọn đối tượng đó.

Ví dụ:

` try `

h. Thay đổi màu nền

Để thay đổi màu nền của trang html bạn có thể sử dụng:

```
window.document.backgroundColor='red';
```

Ví dụ:

` change background `

i. Nạp ảnh nền và cách thay đổi ảnh nền

Để nạp một ảnh nền cho trang html, bạn cần khai báo trong thẻ `` như sau:

```

```

Để có thể thay đổi ảnh nền khi bạn click vào đối tượng, bạn phải đặt tên cho ảnh nền trong thẻ `` hiện tại giống như: **name="mypic"** sau đó bạn có thể làm cho ảnh nền thay đổi khác khi click chuột vào liên kết:

Ví dụ:

` change picture `

Bài tập củng cố

1. Trong javascript để mở một cửa sổ mới ta sử dụng			
<input type="radio"/> window.location	<input type="radio"/> window.open	<input type="radio"/> cả hai	
2. Biểu thức điều kiện so sánh bằng trong javascript:			
<input type="radio"/> cặp dấu bằng ==	<input type="radio"/> dấu bằng =	<input type="radio"/> cả hai	
3. Hàm close() để đóng cửa sổ đang mở có bao nhiêu đối số			
<input type="radio"/> 2 đối số	<input type="radio"/> 3 đối số	<input type="radio"/> không giới hạn đối số	<input type="radio"/> Không có
4. Sự kiện nào sẽ được xảy ra khi di chuyển chuột trên đối tượng			
<input type="radio"/> onClick	<input type="radio"/> onMouseOver	<input type="radio"/> onMouseOut	<input type="radio"/> Không có
5. Sự kiện nào xảy ra khi chuột phải được nhấn trên đối tượng			
<input type="radio"/> onClick	<input type="radio"/> onMouseup	<input type="radio"/> onMouseDown	

Xem kết quả

2. Các cấu trúc điều khiển trong javascript

a. Chuyển hướng người dùng đến trang web khác

Để chuyển hướng người sử dụng đến một trang web khác bạn có thể sử dụng như sau:

Cú pháp `window.location =URL;`

URL: là một tài nguyên, một trang web trên mạng hay máy của bạn.

Ví dụ: `window.location = "http://www.vnn.vn";`

b. Cấu trúc câu lệnh if-then

Đây là một câu lệnh điều kiện trong ngôn ngữ javascript, giống như trong các ngôn ngữ khác, cú pháp của câu lệnh này như sau:

```
if(<điều kiện>
    { <nhóm lệnh 1;> }
    [else{ <nhóm lệnh 2;> }]
```

Nếu điều kiện nhận giá trị **true** thì sẽ thực hiện **<nhóm lệnh 1>** còn không sẽ thực hiện **<nhóm lệnh 2 >**.

Chú ý: + Nếu sử dụng phép so sánh bằng trong biểu thức điều kiện bạn phải dùng hai dấu bằng(==) như trong ngôn ngữ java.

+ Có thể có hay không có **else** đối với câu lệnh **if**.

+ Có thể có các **if – else** lồng nhau.

c. Giá trị boolean và các phép so sánh

các phép toán so sánh trả về giá trị kiểu boolean:

Phép toán	Ý nghĩa
>	Lớn hơn
<	Nhỏ hơn
==	So sánh bằng
>=	Lớn hơn hay bằng
<=	Nhỏ hơn hay bằng
!=	Khác nhau

d. Toán tử or, and và mod

Toán tử **or** được ký hiệu bởi 2 dấu || thể hiện phép **hoặc** thường sử dụng trong biểu thức điều kiện.

Toán tử **and** được ký hiệu bởi 2 dấu && thể hiện phép **và** thường sử dụng trong biểu thức điều kiện.

Toán tử **mod** được ký hiệu bởi dấu % để lấy kết quả phần dư của phép chia.

Chú ý: Giống như java, để gộp nhiều câu lệnh đơn lại thành một câu lệnh ghép ta có thể sử dụng cặp dấu ngoặc móc {}.

3. Các sự kiện xử lý trên trang html của bạn

Phần này nói một số thủ thuật xử lý kích hoạt các sự kiện về mouse như: di chuyển mouse trên đối tượng, click mouse vào đối tượng...

a. **Sự kiện onClick():** Sự kiện này được xảy ra khi bạn kích chuột phải vào đối

tượng.

Ví dụ: ``

hay

`<input type="submit" onClick="javascript:closeWindow()">`

với hàm **closeWindow()** được viết như sau:

```
function closeWindow()
{
    top.window.close();
}
```

b. Sự kiện onMouseOver(): sự kiện này xảy ra khi ta di chuyển chuột lên một đối tượng liên kết nào đó.

c. Sự kiện onMouseOut(): sự kiện này xảy ra khi ta di chuyển chuột ra ngoài một đối tượng liên kết nào đó.

d. Sự kiện ondblclick() Sự kiện này được xảy ra khi bạn kích đôi chuột phải vào đối tượng.

e. Sự kiện onmousedown() Sự kiện này được xảy ra khi chuột phải được nhấn xuống trên đối tượng.

f. Sự kiện onmouseup() Sự kiện này được xảy ra khi chuột phải được nhả ra trên đối tượng.

Chú ý: Các Sự kiện *onkeypress()*, *onkeydown()*, *onkeyup()* được xảy ra khi ta nhấn một phím, nhấn xuống hay nhả phím khi ta chọn đối tượng đó.

Ví dụ:

` try `

h. Thay đổi màu nền

Để thay đổi màu nền của trang html bạn có thể sử dụng:

```
window.document.backgroundColor='red';
```

Ví dụ:

` change background `

i. Nạp ảnh nền và cách thay đổi ảnh nền

Để nạp một ảnh nền cho trang html, bạn cần khai báo trong thẻ `` như sau:

```

```

Để có thể thay đổi ảnh nền khi bạn click vào đối tượng, bạn phải đặt tên cho ảnh nền trong thẻ `` hiện tại giống như: **name="mypic"** sau đó bạn có thể làm cho ảnh nền thay đổi khác khi click chuột vào liên kết:

Ví dụ:

` change picture `

Bài tập củng cố

1. Trong javascript để mở một cửa sổ mới ta sử dụng			
<input type="radio"/> window.location	<input type="radio"/> window.open	<input type="radio"/> cả hai	
2. Biểu thức điều kiện so sánh bằng trong javascript:			
<input type="radio"/> cặp dấu bằng ==	<input type="radio"/> dấu bằng =	<input type="radio"/> cả hai	
3. Hàm close() để đóng cửa sổ đang mở có bao nhiêu đối số			
<input type="radio"/> 2 đối số	<input type="radio"/> 3 đối số	<input type="radio"/> không giới hạn đối số	<input type="radio"/> Không có
4. Sự kiện nào sẽ được xảy ra khi di chuyển chuột trên đối tượng			
<input type="radio"/> onClick	<input type="radio"/> onMouseOver	<input type="radio"/> onMouseOut	<input type="radio"/> Không có
5. Sự kiện nào xảy ra khi chuột phải được nhấn trên đối tượng			
<input type="radio"/> onClick	<input type="radio"/> onMouseup	<input type="radio"/> onMouseDown	

Xem kết quả

6. Nhập và nhận lại thông tin từ các biểu mẫu

Biểu mẫu là sự thiết kết, nơi kết hợp nhiều những đối tượng lên trên một trang html của bạn dùng để nhập/xuất giá trị vào/ra. Các đối tượng đó bao gồm: Textfield, checkboxes, radio button, select, textarea...

a. Cách đặt tên cho các đối tượng trong biểu mẫu

Để truy cập đến các đối tượng trong biểu mẫu, bạn cần phải đặt tên cho các đối tượng trong biểu mẫu đó,

Quy tắc đặt tên cho các đối tượng giống như đặt tên cho biến, đặt tên hàm, tên không có khoảng trống. Tên của các đối tượng được xác định trong từ khoá name của các thẻ cụ thể.

Ví dụ: `<input type="submit" name="OK">`

`<input type="text" name="T1">`

OK: là tên của nút submit

T1: là tên của TextField

b. Đọc và thiết lập giá trị cho các phần tử

+ *Đọc và thiết lập giá trị cho các trường văn bản:*

Để thiết lập giá trị hay nhận lại giá trị từ các trường văn bản bạn phải truy cập đối các đối tượng đó theo cú pháp sau:

window.document.formname.objectname.value

Trong đó formname là tên của form được xác định trong thẻ <form> thường đặt ở dòng đầu tiên của phần <body>

Ví dụ: `<form name="myform">`

objectname: là tên của đối tượng văn bản cần truy cập.

Để gán giá trị cho trường văn bản đó bạn viết như sau:

window.document.formname.objectname.value="giá trị";

Để nhận lại giá trị từ các trường văn bản bạn viết:

var st=window.document.formname.objectname.value;

Ví dụ:

```
<script language="JavaScript">
function docong() {
var a=document.form.T1.value;
var b=document.form.T2.value;
document.form.T3.value=a+' '+'+b+'=';
document.form.T4.value=a*1+b*1;
}
function clearS() {
document.form.T3.value="";
document.form.T4.value="";
window.status="Hay nhap lai cho a va b ";
return true;
}
</script>
```



```

    <option value="male" >
    <option value="female" >
  </select>

```

```

</form>

```

Lúc đó ta có thể biết menu thả xuống nào được chọn thông qua chỉ số của nó như sau:

```

if(window.document.myform.selectlist.options[0].checked==true)
    {alert("this is boy");
    }

```

Ngoài ra ta có thể xác định được phần tử nào được chọn thông qua phương thức `selectedIndex`, phương thức này sẽ trả về chỉ số phần tử được chọn trong menu thả xuống, cách viết như sau:

```

var num = window.document.myform.selectlist.selectedIndex;

```

Ta cũng có thể lấy lại giá trị từ phần tử nào đó được chọn trong menu thả xuống thông qua phương thức `value` như sau:

```

var num = window.document.myform.selectlist.selectedIndex;
var val = window.document.myform.selectlist.options[num].value;

```

+Xử lý các sự kiện bằng cách dùng các phần tử của biểu mẫu

Phần tử	Sự kiện	Hành động kích hoạt
Nút nhấn	onClick	
Hộp kiểm tra	onClick	
Nút chọn lựa	onClick	
Trường văn bản	onChange	Thay đổi nội dung trường văn bản và sau đó nhấp chuột bên ngoài trường văn bản đó.
Vùng văn bản	onChange	Thay đổi nội dung vùng văn bản và sau đó nhấp chuột bên ngoài vùng văn bản đó.
Chọn lựa	onChange	Thay đổi mục chọn lựa trong menu thả xuống
Biểu mẫu	OnSubmit	Nhấn Enter trong một trường văn bản hay nhấp chuột vào nút submit.

Ví dụ:

```

<form name= "my" onSubmit= "window.location= window.document.my.T1.value;
return false;" >

```

```

<input type ="text" name="T1" value="http://">

```

```

<input type="submit" value="go there">

```

```

</form>

```

Sau đây là một ví dụ sử dụng menu thả xuống như một công cụ mở một trang mới theo địa chỉ cho sẵn:

```

<script language="JavaScript">

```

```

function doit(site)
{ window.location=site;
}
</script>

```

Sau đó viết trong phần <body> như sau:

```

<form name="lam">
  <select name="sel" onChange= "doit(this.options[this.selectedIndex].value);">
  <option value="http://www.vnn.vn"> Trang thông tin việt nam
  <option value="http://www.w3c.org"> Trang Công nghệ IBM
  <option value="http://www.tegosoft.com"> Microsoft learning
  <option value="http://www.sun.ibm.com"> Java
  </select>

```

Chú ý: vì ta gọi trực tiếp trong đối tượng select, từ khoá this có thể thay thế cho đối tượng hiện hành này:

Ta có thể thay thế từ this này với cụm từ thường dùng:

window.document.lam.sel

Bài tập củng cố

1. Trong javascript để gán giá trị cho các đối tượng như Textbox, radio, checkbox, option... ta gán thông qua phương thức value của đối tượng đó			
<input type="radio"/>	Đúng	<input type="radio"/>	Sai
2. Để viết một hàm trả về một kết quả nào đó ta phải khai báo return trong thân hàm			
<input type="radio"/>	Đúng	<input type="radio"/>	Không đúng
3. Để gán thuộc tính được chọn (checked) cho đối tượng radio hay checkbox ta có thể gán true cho phương thức checked của đối tượng đó			
<input type="radio"/>	Không đúng	<input type="radio"/>	đúng
4. Từ khoá this có thể thay thế cho window.document.<formname>.<objectname> khi nào			
<input type="radio"/>	Gọi và xử lý trực tiếp trên đối tượng đó	<input type="radio"/>	Tất cả mọi khi
<input checked="" type="radio"/>	Không thay thế được		
5. Khi nhấn Enter trong trường văn bản hay click vào nút Submit thì sự kiện nào được gọi:			
<input type="radio"/>	OnClick()	<input type="radio"/>	OnSubmit()
<input checked="" type="radio"/>	OnChange()		

Xem kết quả

9. Xử lý chuỗi

a. Các phương thức kiểm tra xử lý chuỗi

- + **indexOf(s)**: Trả về vị trí của chuỗi s trong chuỗi mẹ tính từ đầu chuỗi
- + **lastIndexOf(s)**: Trả về vị trí của chuỗi s trong chuỗi mẹ tính từ cuối chuỗi trở về đầu chuỗi.
- + **charAt(i)**: Trả về kí tự tại vị trí thứ i của chuỗi.
- + **substring(m,n)**: Trả về một chuỗi con lấy từ chuỗi mẹ tại vị trí m, lấy n kí tự.
- + **split(s)**: Cắt chuỗi mẹ thành nhiều chuỗi con với ký tự ngăn cách các chuỗi là s, trị trả về của hàm này là một mảng các chuỗi con.

Ví dụ:

```
<html><head>
<script language="JavaScript">
<!--
function validdate(s){
if (s.split("/").length==3)
{   var d= s.split("/")[0];
    var m= s.split("/")[1];
    var y= s.split("/")[2];
    alert("Hom nay la ngay "+d+" thang "+m+" nam "+y);
}
else alert("invalid date");
}
-->
</script></head><body>
<form name="form">
<input type="text" name="t1" value ="0" size="20">
<input type="button" value="Kiem tra" size="20"
onClick="validdate(window.document.form.t1.value);">
</form></body></html>
```

Bạn cần chú ý thuộc tính maxlength, disabled, checked của đối các tượng text, button, radio, checkbox trong form.

Ví dụ:

```
<input type="text" name="t1" maxlength ="10" size="20">
<input type="button" value="See Hidden" disabled>
<input type="radio" name="Hidden" checked>
<input type="text" name="t2" disabled size="20">
```

b. Chương trình ví dụ:

Sau đây là một số hàm mẫu để xử lý dữ liệu khi nhập vào từ một số đối tượng:

+ Hàm kiểm tra dữ liệu nhập vào từ một TextField có đúng theo qui định hay không.

```

/*****
// hàm: validString
// form    -- Tên form
// object  -- Tên đối tượng
// lenMax  -- Độ dài lớn nhất
// lenMin  -- Độ dài nhỏ nhất
// space   -- Cho phép chuỗi chứa khoảng trống hay không (0- cho phép; 1- không )
// require -- Cho phép hay không cho phép bỏ qua (0- cho phép; 1- không)
/*****
function validString(form, object, lenMax, lenMin, space, require ){
    var tmp, tmp2, msg;
    tmp2 = " ";
    msg = "";
    tmp = document.forms(form).all(object).value;
    len1 = tmp.length;
    if (require == 1){
        if (len1 < 1){    msg = 'Invalid ';
            alert(msg);
            document.forms(form).all(object).focus();
            return false;
        }
    }
    if (lenMax != lenMin){
        if (len1 > lenMax){
            msg = 'Invalid ' + lenMax;
            alert(msg);
            document.forms(form).all(object).focus();
            return false;
        }
        if (len1 < lenMin){
            msg = 'Invalid ' + lenMin;
            alert(msg);
            document.forms(form).all(object).focus();
            return false;
        }
    }
    if (space == 1){
        for(i=0; i<len1; i++){
            if (tmp.charAt(i) == tmp2){
                msg = 'Invalid ';
                alert(msg);
            }
        }
    }
}

```

```

document.forms(form).all(object).focus();
return false;
}
}
}
return true;
}

```

10. Cookie và HTML động

a. Cookie

Là một mẫu thông tin mà khi bạn truy cập vào một website nó sẽ tạo ra và được lưu lại dưới đĩa cứng của người sử dụng.

+ Thiết lập một cookie

Ta thiết lập cookie cho document bằng cách:

```
document.cookie= <chuỗi cookie>
```

Sau đó ta sử dụng hàm escape() để mã hoá cookie cho bạn:

Ví dụ:

```

function setcookie()
{
    var name= prompt("Input your name “,”");
    var cookiename="username="+escape(name);
    document.cookie=cookiename;
}

```

+ Đọc một cookie

Ta sử dụng hàm unescape() giải mã cookie của bạn:

Ví dụ:

```

function readcookie()
{
    var thecookie= document.cookie;
    var cookiename= thecookie.split("=");
    var name= cookiename[1];
    name=unescape(name);
    alert("your name: "+name);
}

```

b. HTML động và khái niệm cơ bản về CSS.

+ Thẻ DIV

Thẻ này cho phép người dùng có thể định vị một thẻ HTML bất kỳ vị trí nào trên trang của bạn. Đây là cách làm cho trang web của bạn động, có thể thấy hình ảnh di chuyển được trên màn hình.

Ví dụ:

```

function movediv(which) {
    var thediv;
    if (window.document.all)
        { thediv=window.document.all.mydiv.style;
        }
}

```



```

else if(document.layers)
    {
        thediv=window.document.mydiv;
    }else
    {
        alert("xin chao");return;
    }
thediv.left=parseInt(thediv.left)+10*(Math.random()*10-5);
thediv.top=parseInt(thediv.top)+10*(Math.random()*10-5);
}

```

Sau đó trong phần <body> bạn viết như sau:

```
<body onClick="movediv()">
```

11. Một số phương thức thường dùng:

a. applet

Tất cả mọi applet trên trang được lưu vào mảng thông qua cách truy cập như sau:

```
window.document.applets[];
```

Để kích hoạt nó ta sử dụng phương thức start() và để tắt nó ta sử dụng phương thức stop().

Ví dụ: window.document.applets[0].start();

b. Area

Các thẻ area được lưu trong mảng sau:

```
window.document.links[];
```

Mỗi Area được lưu như một đối tượng và ta có thể truy cập nó như những đối tượng khác.

Một số phương thức thông dụng:

+ **confirm()**: Sẽ đưa ra một câu thông báo và Trả về true hay false nếu người dùng chọn YES hay NO.

+ **toUpperCase()**: Chuyển một chuỗi thành chuỗi chữ hoa.

+ **toLowerCase()**: Chuyển một chuỗi thành chuỗi chữ thường.

+ **focus()**: Sẽ di chuyển con trỏ đến đối tượng mà ta thiết đặt focus(), thường sử dụng trong việc kiểm tra lỗi và yêu cầu người dùng nhập lại.

+ **history.back()**: Trở lại trang trước đó.

+ **eval(s)**: Định giá trị của một biểu thức.

+ **bgColor()**: Thiết đặt màu nền.

+ **concat()**: ghép nối chuỗi.

Ví dụ:

```
function setcolorbg ()
{
    window.document.bgColor="#008800";
    return false;
}
```

c. Các hàm toán học:

+ **abs()**

- + *acos()*
- + *asin()*
- + *atan()*
- + *cos()*
- + *exp()*
- + *log()*
- + *max(a,b)*
- + *min(a,b)*
- + *pow(e,x)*
- + *random()*
- + *round()*
- + *sin()*
- + *sqrt()*
- + *tan()*

Bài tập củng cố

1. Để cắt một chuỗi tự thành các chuỗi con thông qua ký tự ngăn cách nào đó ta sử dụng phương thức			
<input type="radio"/> SubString()	<input type="radio"/> Split()	<input type="radio"/> Cả hai	
2. Để tra về vị trí của chuỗi con trong chuỗi mẹ tính từ đầu chuỗi ta sử dụng			
<input type="radio"/> IndexOf()	<input type="radio"/> LastIndexOf()	<input type="radio"/> SubString()	
3. Để giới hạn độ dài tối đa của một đối tượng Textbox khi nhập dữ liệu từ bàn phím ta sử dụng			
<input type="radio"/> maxlength	<input type="radio"/> Disabled	<input type="radio"/> Hidden	
4. Để thiết lập chế độ bảo mật của một trang web nào đó ta sử dụng			
<input type="radio"/> Session	<input type="radio"/> Cookie	<input type="radio"/> Cả hai có thể	
5. Để mã hoá một đối tượng cookie ta dùng			
<input type="radio"/> Unescape()	<input type="radio"/> escape()	<input type="radio"/> Cả hai	

[Xem kết quả](#)

Ngôn ngữ Javascript

Người soạn: Lê Thanh Lâm

I. Giới Thiệu

JavaScript là một ngôn ngữ lập trình được xây dựng sẵn trên các trình duyệt web, là một trong những cách tốt nhất cho phép bạn có thể bổ sung tính năng tương tác cho webSite của mình, nó là ngôn ngữ cho phép làm việc trực tiếp với các trình duyệt.

JavaScript rất đơn giản, dễ học, không cần phải có bất kỳ một phần mềm hay phần cứng nào yêu cầu khi thực thi chương trình JavaScript mà bạn chỉ cần có trình duyệt trên máy là đủ. Bạn không cần phải có một kiến thức về máy tính, về ngôn ngữ lập trình mới có thể viết được ngôn ngữ này.

JavaScript là một ngôn ngữ hoàn chỉnh, nó cho phép bạn khá nhiều công cụ để lập trình.

Ngày nay, JavaScript hầu như được sử dụng rất phổ dụng trên hầu hết các webSite trên thế giới.

II. Mục Đích

Giúp bạn có khả năng tự viết riêng những đoạn JavaScript riêng của mình, có khả năng đọc hiểu những chương trình JavaScript do người khác viết.

Giúp bạn có thể tìm kiếm thông tin, học về JavaScript thông qua một số webSite:

<http://www.w3schools.com>

<http://www.javascript.com>

<http://www.developer.earthweb.com/directories/pages/dir.javascript.html>

Thông qua một số ví dụ mẫu bạn có thể học hỏi cách xây dựng một chương trình JavaScript. Tạo các hiệu ứng trên các trang web của bạn rất xinh động.

III. Ưu Điểm

Tương tác trực tiếp với các trang web của bạn. Để sử dụng các chương trình viết bằng JavaScript bạn chỉ cần thêm các mã nguồn của bạn vào giữa thẻ <script> trong phần <head> của trang html của bạn:

```
<script language = "JavaScript">
```

```
.....code script.....
```

```
</script>
```

Sau đó bạn có thể sử dụng các hàm của mình trong trang html.

Hoặc bạn có thể chèn 1 file nguồn javascript như sau:

```
<script language="JavaScript" src="....."></script>
```

IV. Nhược điểm

JavaScript không thể tương tác, giao tiếp với máy chủ.

JavaScript không thể tạo ra các hình ảnh đồ họa.

JavaScript làm việc có thể khác nhau trên các trình duyệt khác nhau.

V. Các Thành phần cơ bản của javascript

1. Cách sử dụng biến và hàm trong JavaScript

2. Các cấu trúc điều khiển trong javascript

3. Các sự kiện xử lý trên trang html của bạn

4. Mở và thao tác các cửa sổ

5. Xây dựng hàm javascript cho bạn

6. Nhập và nhận lại thông tin từ các biểu mẫu

7. Mạng và vòng lặp

8. Định thời gian cho các sự kiện

9. Xử lý chuỗi

Bài tập củng cố

1. Theo bạn biết Javascript có thể nhúng vào được những loại trang web nào:			
<input type="radio"/> Trang HTML	<input type="radio"/> Trang Servlet	<input type="radio"/> Trang JSP	<input type="radio"/> Tất cả
2. Trong Javascript bạn có thể xử lý các hình ảnh đồ họa được không?			
<input type="radio"/> Được	<input type="radio"/> Không		
3. Việc viết những đoạn chương trình JavaScript trong trang web với mục đích			
<input type="checkbox"/> Tăng hiệu quả xử lý trên các trang	<input type="checkbox"/> Kiểm tra dữ liệu	<input type="checkbox"/> Làm cho tốc độ nhanh hơn	
4. Javascript có phải là một ngôn ngữ lập trình không?			
<input type="radio"/> Có	<input type="radio"/> Không		
5. Bạn có thể viết những đoạn mã javascript vào trang web của mình tại nơi nào			
<input type="radio"/> Title	<input type="radio"/> Head	<input type="radio"/> Body	

Xem kết quả

BÀI 1: GIỚI THIỆU VỀ ASP

1. GIỚI THIỆU VỀ ASP.

ASP (Active Server Pages) là một môi trường lập trình cung cấp cho việc kết hợp HTML, ngôn ngữ kịch bản (Scripting) như VBScript, Javascript, và các thành phần được viết trong các ngôn ngữ nhằm tạo ra một ứng dụng Internet mạnh mẽ và hoàn chỉnh.

2. ASP file là gì?

File được tạo với phần mở rộng .ASP. Trong file này chứa các thẻ HTML, Các kịch bản Scripting như VBScript, JavaScript hay các lời gọi đến các components(Như DLL và ActiveX control)

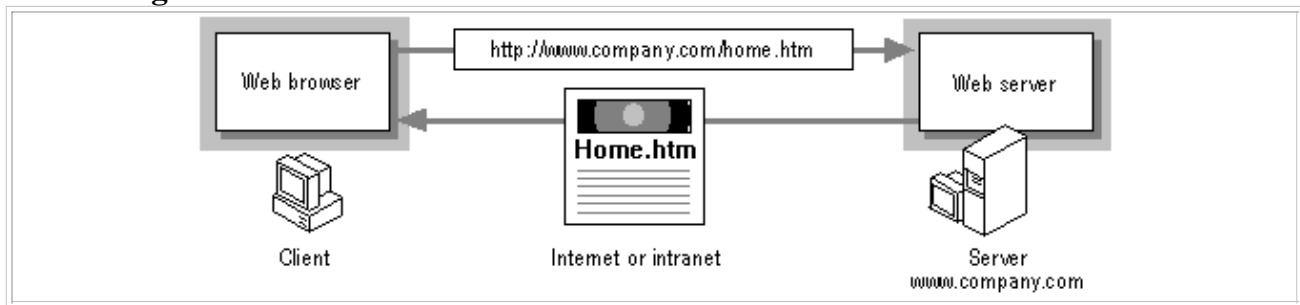
Các script của ASP được nằm trong cặp thẻ `<% %>`

Khi cần sửa đổi các file ASP ta chỉ cần ghi lại trên server thôi. Vào nhưng lần sau khi trang A được gọi, các Script trong file ASP tự động biên dịch lại.

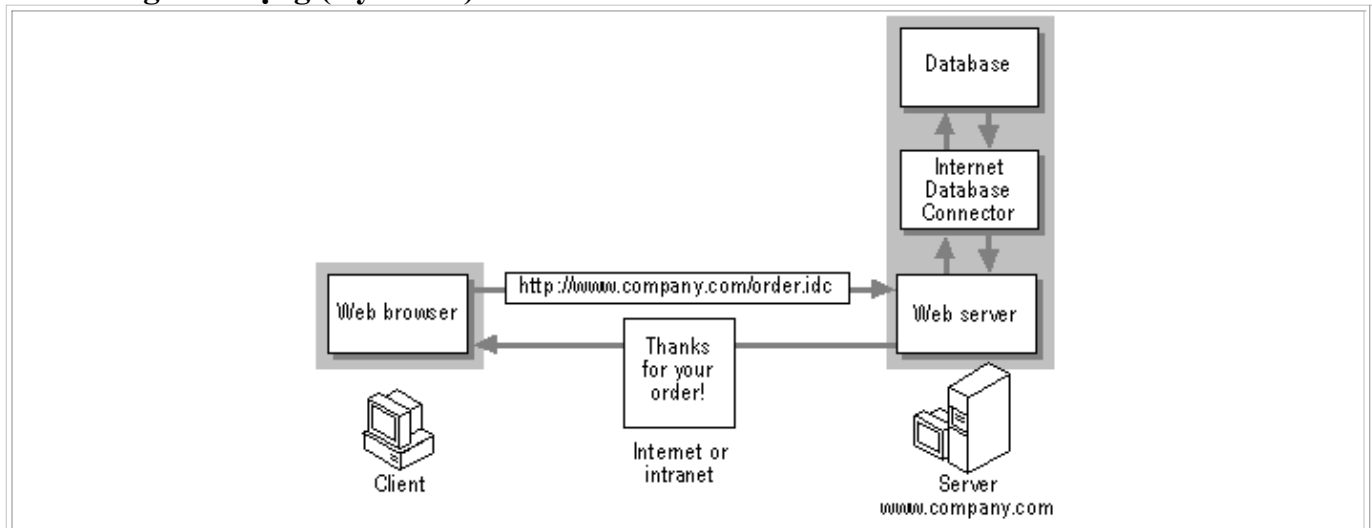
Công nghệ ASP được xây dựng trực tiếp bên trong ; IIS(WinNT, 2000, XP), Personal Webserver.

3. ASP làm việc như thế nào ?

Trang HTML tĩnh:



Trang web động (Dynamic):



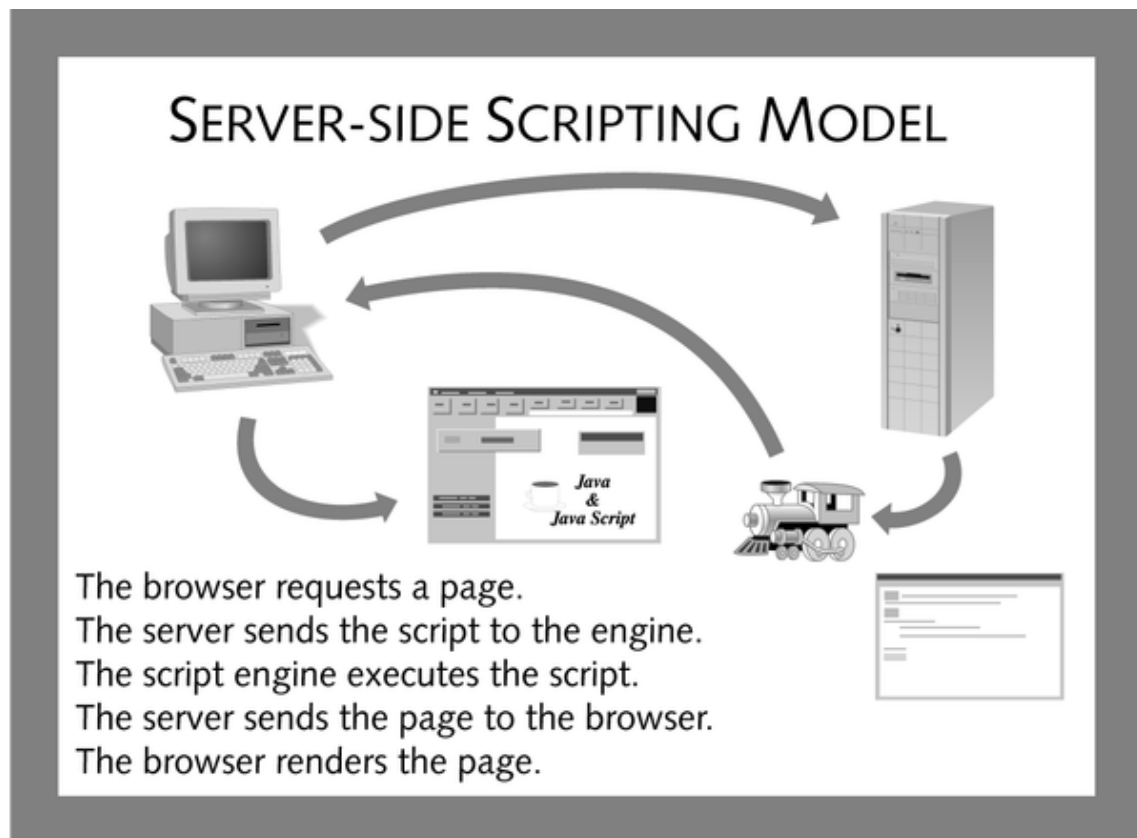


Figure 2-4 The server-side script model

4. Giới thiệu về IIS - Internet Information Server.

a. IIS là gì?

Microsoft Internet Information Server là một ứng dụng server chuyên giao thông tin bằng việc sử dụng giao thức chuyển đổi siêu văn bản HTTP

b. IIS có thể làm được gì?

Ø Xuất bản một Home page lên Internet.

Ø

Tạo các giao dịch thương mại điện tử trên Internet(Quá trình giao dịch, đặt hàng...)

Ø

Cho phép người dùng từ xa có thể truy xuất Cơ sở dữ liệu (Data Base Remote Access)

5. IIS hoạt động như thế nào ?

Web, về cơ bản thực sự là một hệ thống các yêu cầu (Request) và các đáp ứng (Response). IIS phản hồi lại các yêu cầu đòi thông tin của Web Browser. IIS lắng nghe các yêu cầu đó từ phía Users trên một mạng sử dụng WWW.

6. Cài đặt IIS.

Thực hành trên máy

7. Ví dụ về file ASP.

```
<html>
<body>
<%
response.write("Hello World!")
%>
</body>
</html>
```

```
<html>
<body>
<%= "Hello World!" %>
</body>
</html>
```

BÀI 2: LẬP TRÌNH VỚI ASP

1. Khai báo biến.

- Không bắt buộc nhưng nên khai báo để kiểm soát và bắt lỗi.
- Nên sử dụng <% Option Explicit %> ở ngay đầu mỗi tệp ASP.
- Cú pháp: **Dim biến 1, biến 2...**
- Để khai báo mảng:
 1. Dim a(10) : chỉ số chạy từ 0->10 do đó có 11 phần tử.
 2. Khi khai báo nên sử dụng các tiền tố:

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

Ví dụ:

```
<html>  
<body>  
<%  
dim name  
name="Donald Duck"  
response.write("My name is: " & name)  
%>  
</body>  
</html>
```

2. Khai báo hằng số.

CONST tên hằng= giá trị

3. Các đối tượng xây dựng sẵn trong ASP.

Trong ASP có 5 đối tượng phục vụ cho việc sử dụng toàn bộ.

- + **Request:** Lấy thông tin từ User
- + **Response:** Gửi thông tin từ Server tới User.
- + **Server:** Dùng để điều khiển IIS
- + **Session:** Dùng để lưu trữ các thông tin như các cài đặt, thay đổi cho một phiên làm việc hiện thời của User.
- + **Application:** Dùng để chia sẻ các thông tin cấp ứng dụng và điều khiển các thiết lập cho toàn bộ quá trình chạy ứng dụng

BÀI 3: CẤU TRÚC ĐIỀU KIỆN-LẶP.

1. Các cấu trúc điều khiển.

A. CẤU TRÚC ĐIỀU KIỆN IF..THEN..ELSE..IF

Cú pháp:

```
<%  
    If <Điều kiện 1> then  
        <Các câu lệnh>  
    Else  
        If <Điều kiện 2> then  
            <Các câu lệnh>  
        End if  
    End if  
%>
```

Ngoài ra có thể sử dụng cấu trúc IF.. THEN. . . ELSEIF. . . END IF

```
<%  
    If <Điều kiện 1> then  
        <Các câu lệnh 1>  
    ElseIf <Điều kiện 2> then  
        <Các câu lệnh 2>  
    Else  
        <Các câu lệnh 3>  
    End if  
%>
```

B. CẤU TRÚC LỰA CHỌN SELECT CASE...END SELECT

Cú pháp:

```
<%  
    Select Case <tên biến>  
        Case <giá trị 1>  
            <Nhóm lệnh 1>  
        Case < giá trị 2>  
            <Nhóm lệnh 2>  
        Case < giá trị n>  
            <Nhóm lệnh n>  
    Case Else  
        <Nhóm lệnh n+1>  
%>
```

Ví dụ: Hiện ngày giờ trên máy chủ và cho biết hôm nay là thứ mấy?

C. CẤU TRÚC LẶP TUẦN TỰ FOR...NEXT.

Dùng để lặp với số lần đã biết, tuy nhiên ta có thể ngắt bằng lệnh EXIT

FOR.

Cú pháp:

```
<%
```

```
FOR <tên biến>=<giá trị đầu> TO <giá trị biến> STEP <bước nhảy>
NEXT
%>
```

VÍ DỤ:

```
<html>
<body>
<%
Dim famname(5),i
famname(0) = "Jan Egil"
famname(1) = "Tove"
famname(2) = "Hege"
famname(3) = "Stale"
famname(4) = "Kai Jim"
famname(5) = "Borge"
For i = 0 to 5
    response.write(famname(i) & "<br />")
Next
%>
</body>
</html>
```

D. CÁC CẤU TRÚC LẶP .

D.1 DO WHILE....LOOP.

Cú pháp:

```
<%
    Do while <điều kiện>
        <Các câu lệnh>
    Loop
%>
```

D.2 WHILE....WEND.

Cú pháp:

```
<%
    While < điều kiện >
        <Các câu lệnh>
    Wend
%>
```

D.3 DO....LOOP UNTIL

Cú pháp:

```
<%
    DO
        <Các câu lệnh>
    Loop until <Điều kiện>
%>
```

E. CÁC VÍ DỤ .

2. Một số hàm hữu dụng trong ASP.

A. Các Hàm xử lý văn bản

- i. **TRIM(xâu as string)**: Bỏ khoảng trắng hai đầu kí tự
- ii. **LEFT(Xâu as string, n as interger)**: Lấy bên trái xâu n kí tự.
- iii. **RIGHT(Xâu as string, n as interger)**: Lấy bên phải xâu n kí tự.
- iv. **LCASE(Xâu as string)** : Chuyển xâu về chữ thường
- v. **UCASE(Xâu as string)** : Chuyển xâu về chữ hoa
- vi. **MID(xâu as string, n1, n2)**: Lấy n2 kí tự trong xâu bắt đầu từ vị trí n1.
- vii. **CSTR(Biến)**: Hàm chuyển đổi biến thành kiểu string
- viii. **Hàm JOIN/SPLIT(Xâu as string, kí tự ngăn cách)**:

Sẽ **Nối/Cắt** xâu thành một/nhiều đoạn bằng cách xác định kí tự ngăn cách ở trên và cho các đoạn đó lần lượt vào một mảng.

Ví dụ:

```
<%  
    x="Hà nội;Hải phòng; TPHCM"  
    y=split(x,";")  
    Response.write y(0)  
%>
```

B. Các hàm xử lý số.

- i. **SQR(n)**: Căn bậc hai của n
- ii. **INT(n)** : Lấy phần nguyên n
- iii. **MOD** : Phép đồng dư
- iv. **Toán tử ** : Chia nguyên
- v. **Round(số, n)** : Làm tròn số với n chữ số thập phân
- vi. **RND()**: Trả về số ngẫu nhiên bất kì trong khoảng [0,1]

C. Các hàm về ngày tháng thời gian.

BÀI 4: XÂY DỰNG ỨNG DỤNG ASP.

1. Các thẻ HTML.

Cần xem lại các tag HTML đặc biệt là: **FORM(name, method, action...)**, **INPUT(TEXT, BUTTON, SUBMIT...)**, **CHECKBOX, RADIO, OPTION....**

Mỗi thẻ TAG HTML phải được xác định tên của nó(name), giá trị của nó(value), và các thuộc tính khác.

Ví dụ: `<INPUT TYPE=TEXT NAME=T1 VALUE=TH READONLY>`

2. Thẻ FORM.

Trong một trang có thể có nhiều FORM, mỗi form xác định một tập các hoạt động của nó, form phải được đặt 1 tên(name), trong mỗi form có thể có nhiều đối tượng như TEXTBOX, BUTTON, SUBMIT, OPTION.....

Mỗi form có phương thức(Method) chuyển dữ liệu(nhận hay gửi dữ liệu nó bao gồm 2 phương thức **GET** hay **POST**), và phải xác định chuyển dữ liệu đến đâu thông qua **ACTION**, vì vậy để khai báo Form thông thường chúng ta phải khai báo như sau:

`<FORM NAME=formname METHOD=get/post ACTION=url>`

Ví dụ:

Cho form sau gồm Textbox có tên là T1

```
<form name=lam method="get" action="simpleform.asp">
First Name: <input type="text" name="T1" />
<br />
<input type="submit" value="Submit" />
</form>
```

Để nhận lại dữ liệu ta có thể sử dụng 1 trong 2 cách sau: **Request.QueryString** hoặc **Request.Form**.

* Sử dụng Request.QueryString

Lệnh **Request.QueryString** thường đi kèm với form sử dụng phương thức GET (method="get"). Thông tin gửi từ Form có phương thức GET sẽ được hiển thị trên thanh address bar của trình duyệt và nó bị giới hạn bởi thông tin được gửi(như số đối số, giá trị dữ liệu...).

Nếu bạn nhập giá trị cho T1 là DONGA thì trên thanh địa chỉ sẽ hiển thị như sau:

```
http://maychu1/05TH1a/06th0010/simpleform.asp?T1=DONGA
```

Giả sử ta có file ASP có tên "simpleform.asp" chứa đoạn mã sau:

```
<body>
Welcome :
<%
response.write(request.querystring("T1"))
%>
</body>
```

Kết quả hiển thị ra màn hình:

```
Welcome DONGA
```

* Sử dụng Request.Form

Câu lệnh Request.Form dùng để nhận giá trị từ form với phương thức GET(method="post"). Thông tin gửi từ form với phương thức POST sẽ không bị giới hạn bởi đối số và dữ liệu.

If a user typed "Bill" and "Gates" in the form example above, the URL sent to the server would look like this:

```
http://maychu1/05TH1a/06th0010/simpleform.asp
```

Giả sử file asp "simpleform.asp" chứa đoạn mã sau:

```
<body>
Welcome
<%
response.write(request.form("T1"))
%>
</body>
```

Kết quả:

```
Welcome DONGA
```

3. Thẻ INPUT.

4. Tạo vùng nhập liệu.

5. Tạo COMBO BOX.

Ví dụ 1: Sử dụng Request.QueryString

```
<html>
<body>
<form action="demo_reqquery.asp" method="get">
```

```

Your name: <input type="text" name="fname" size="20" />
<input type="submit" value="Submit" />
</form>
<%
dim fname
fname=Request.QueryString("fname")
If fname<>"" Then
    Response.Write("Hello " & fname & "!<br />")
    Response.Write("How are you today?")
End If
%>
</body>
</html>

```

Ví dụ 2: Sử dụng RADIO

```

<html>
<%
dim cars
cars=Request.Form("cars")
%>
<body>
<form action="demo_radiob.asp" method="post">
<p>Please select your favorite car:</p>

<input type="radio" name="cars"
<%if cars="Volvo" then Response.Write("checked")%>
value="Volvo">Volvo</input>
<br />
<input type="radio" name="cars"
<%if cars="Saab" then Response.Write("checked")%>
value="Saab">Saab</input>
<br />
<input type="radio" name="cars"
<%if cars="BMW" then Response.Write("checked")%>
value="BMW">BMW</input>
<br /><br />
<input type="submit" value="Submit" />
</form>
<%
if cars<>"" then
    Response.Write("<p>Your favorite car is: " & cars & "</p>")
end if
%>
</body>
</html>

```


BÀI 5: HỆ QUẢN TRỊ CSDL MS Access 2003

CẦN XEM LẠI CÁC KHÁI NIỆM

I. CÁC KHÁI NIỆM CƠ BẢN

- 1/ Hệ Quản Trị CSDL?
- 2/ Cơ Sở Dữ Liệu là gì?
- 3/ Table là gì?
- 4/ Khóa chính/Primary key?
- 5/ Khóa ngoại/Foreign key?
- 6/ Quan hệ/ relationship?
- 7/ Ràng buộc, tham chiếu dữ liệu/references?
- 8/ Duy nhất dữ liệu / Unique?
- 9/ Truy vấn/ query?

II. SQL - Structure Query Language

- 1/ Ngôn ngữ định nghĩa dữ liệu
- 2/ Ngôn ngữ thao tác dữ liệu

Bài 7: KẾT NỐI DATABASE.

1. ADO là gì?

ADO Có thể dùng để truy cập DL từ trang web.

- ADO là công nghệ của Microsoft
- ADO được viết tắt là **ActiveX Data Objects**
- ADO là một thành phần của Active-X
- ADO tự động cài đặt khi cài IIS
- ADO là môi trường lập trình giao tiếp với CSDL- database

1. Cách truy xuất database bằng ADO trong ASP.

Cách thông thường để truy cập CSDL từ trang web ASP như sau:

1. Create an ADO connection to a database
2. Open the database connection
3. Create an ADO recordset
4. Open the recordset
5. Extract the data you need from the recordset
6. Close the recordset
7. Close the connection

2. Tạo kết nối ADO (ADO Connection).

Trước khi truy cập dữ liệu, bạn phải thiết lập kết nối đến CSDL đó. Có 02 cách để thiết lập kết nối.

a/ Kết nối trực tiếp(DSN-less Database Connection)

Đây là cách đơn giản nhất thường kết nối với Microsoft Access database.

Giả sử bạn có một Database có tên "northwind.mdb" đặt tại thư mục "c:/webdata/", bạn có thể kết nối đến CSDL như sau:

```
<%  
set conn=Server.CreateObject("ADODB.Connection")  
conn.Provider="Microsoft.Jet.OLEDB.4.0"  
conn.Open "c:/webdata/northwind.mdb"  
%>
```

b/ Kết nối thông qua ODBC(ODBC Database Connection)

Nếu bạn có ODBC database có tên là "northwind" bạn có thể kết nối vào CSDL này như sau:

```
<%  
set conn=Server.CreateObject("ADODB.Connection")  
conn.Open "northwind"  
%>
```

Với kết nối ODBC connection, Bạn có thể kết nối đến bất kỳ CSDL nào và ở bất kỳ máy nào trong hệ thống mạng của bạn, miễn là ODBC connection tồn tại.

Cách tạo ODBC Connection với MS Access Database

1. Open the **ODBC** icon in your Control Panel.
2. Choose the **System DSN** tab.
3. Click on **Add** in the System DSN tab.
4. **Select** the Microsoft Access Driver. Click **Finish**.
5. In the next screen, click **Select** to locate the database.
6. Give the database a **Data Source Name** (DSN).
7. Click **OK**.

3. Sử dụng ADO Recordset

Để có thể đọc dữ liệu từ database, đầu tiên dữ liệu phải được nạp vào recordset.

Cách tạo ADO Recordset

Sau khi ADO Database Connection đã được tạo, tiếp theo ta phải tạo ADO Recordset.

Giả sử ta có database có tên "Northwind", Chúng ta có thể truy cập dữ liệu từ bản "Customers" như sau:

```
<%  
set conn=Server.CreateObject("ADODB.Connection")  
conn.Provider="Microsoft.Jet.OLEDB.4.0"  
conn.Open "c:/webdata/northwind.mdb"  
set rs=Server.CreateObject("ADODB.recordset")  
rs.Open "Customers", conn  
%>
```

Cách tạo ADO SQL Recordset

Chúng ta có thể truy cập dữ liệu từ bản "Customers" sử dụng câu lệnh SQL:

```
<%  
set conn=Server.CreateObject("ADODB.Connection")  
conn.Provider="Microsoft.Jet.OLEDB.4.0"  
conn.Open "c:/webdata/northwind.mdb"  
set rs=Server.CreateObject("ADODB.recordset")  
rs.Open "Select * from Customers", conn  
%>
```

Lấy dữ liệu từ Recordset

Sau khi recordset được mở, chúng ta có thể lấy dữ liệu recordset.

Chúng ta có thể truy xuất dữ liệu từ bản **Customers** trong CSDL **Northwind** như sau :

```
<%  
set conn=Server.CreateObject("ADODB.Connection")  
conn.Provider="Microsoft.Jet.OLEDB.4.0"  
conn.Open "c:/webdata/northwind.mdb"  
set rs=Server.CreateObject("ADODB.recordset")  
rs.Open "Select * from Customers", conn  
for each x in rs.fields  
    response.write(x.name)  
    response.write(" = ")  
    response.write(x.value)  
next  
%>
```

4. Một số ví dụ.

Hiển thị tên Field và giá trị của Field

Chúng ta có thể truy xuất dữ liệu từ bản **Customers** trong CSDL **Northwind** như sau :

```
<html>  
<body>  
<%  
set conn=Server.CreateObject("ADODB.Connection")  
conn.Provider="Microsoft.Jet.OLEDB.4.0"  
conn.Open "c:/webdata/northwind.mdb"  
set rs = Server.CreateObject("ADODB.recordset")  
rs.Open "SELECT * FROM Customers", conn  
do until rs.EOF  
    for each x in rs.Fields  
        Response.Write(x.name)
```

```
    Response.Write(" = ")
    Response.Write(x.value & "<br />")
next
Response.Write("<br />")
rs.MoveNext
loop
rs.close
conn.close
%>
</body>
</html>
```

Kết quả

```
CustomerID = ALFKI
CompanyName = Alfreds Futterkiste
ContactName = Maria Anders
ContactTitle = Sales Representative
Address = Obere Str. 57
City = Berlin
PostalCode = 12209
Country = Germany

CustomerID = BERGS
CompanyName = Berglunds snabbköp
ContactName = Christina Berglund
ContactTitle = Order Administrator
Address = Berguvsvägen 8
City = Luleå
PostalCode = S-958 22
Country = Sweden

CustomerID = CENTC
CompanyName = Centro comercial Moctezuma
ContactName = Francisco Chang
ContactTitle = Marketing Manager
Address = Sierras de Granada 9993
City = México D.F.
PostalCode = 05022
Country = Mexico

.....
.....
.....
```

Hiển thị tên Field và giá trị của Field trong HTML table

```
<html>
<body>
<%
set conn=Server.CreateObject("ADODB.Connection")
conn.Provider="Microsoft.Jet.OLEDB.4.0"
conn.Open "c:/webdata/northwind.mdb"
set rs = Server.CreateObject("ADODB.recordset")
rs.Open "SELECT Companyname, Contactname FROM Customers", conn
%>
<table border="1" width="100%">
<%do until rs.EOF%>
  <tr>
  <%for each x in rs.Fields%>
    <td><%Response.Write(x.value)%></td>
  <%next
  rs.MoveNext%>
  </tr>
<%loop
rs.close
conn.close
%>
</table>
</body>
</html>
```

Kết quả:

Alfreds Futterkiste	Maria Anders
Berglunds snabbköp	Christina Berglund
Centro comercial Moctezuma	Francisco Chang
Ernst Handel	Roland Mendel
FISSA Fabrica Inter. Salchichas S.A.	Diego Roel
Galería del gastrónomo	Eduardo Saavedra
Island Trading	Helen Bennett
Königlich Essen	Philip Cramer
Laughing Bacchus Wine Cellars	Yoshi Tannamuri
Magazzini Alimentari Riuniti	Giovanni Rovelli
North/South	Simon Crowther
Paris spécialités	Marie Bertrand
Rattlesnake Canyon Grocery	Paula Wilson
Simons bistro	Jytte Petersen

The Big Cheese	Liz Nixon
Vaffeljernet	Palle Ibsen
Wolski Zajazd	Zbyszek Piestrzeniewicz

Thêm Headers vào Table

```

<html>
<body>
<%
set conn=Server.CreateObject("ADODB.Connection")
conn.Provider="Microsoft.Jet.OLEDB.4.0"
conn.Open "c:/webdata/northwind.mdb"
set rs = Server.CreateObject("ADODB.recordset")
sql="SELECT Companyname, Contactname FROM Customers"
rs.Open sql, conn
%>
<table border="1" width="100%">
  <tr>
    <%for each x in rs.Fields
      response.write("<th>" & x.name & "</th>")
    next%>
  </tr>
  <%do until rs.EOF%>
    <tr>
      <%for each x in rs.Fields%>
        <td><%Response.Write(x.value)%></td>
      <%next
        rs.MoveNext%>
    </tr>
  <%loop
  rs.close
  conn.close
  %>
</table>
</body>
</html>

```

Kết quả:

Companyname	Contactname
Alfreds Futterkiste	Maria Anders
Berglunds snabbköp	Christina Berglund
Centro comercial Moctezuma	Francisco Chang
Ernst Handel	Roland Mendel

FISSA Fabrica Inter. Salchichas S.A.	Diego Roel
Galería del gastrónomo	Eduardo Saavedra
Island Trading	Helen Bennett
Königlich Essen	Philip Cramer
Laughing Bacchus Wine Cellars	Yoshi Tannamuri
Magazzini Alimentari Riuniti	Giovanni Rovelli
North/South	Simon Crowther
Paris spécialités	Marie Bertrand
Rattlesnake Canyon Grocery	Paula Wilson
Simons bistro	Jytte Petersen
The Big Cheese	Liz Nixon
Vaffeljernet	Palle Ibsen
Wolski Zajazd	Zbyszek Piestrzeniewicz

5. QUERY

Display Selected Data

Chúng ta chỉ hiển thị những bản ghi từ bản "Customers" mà trường "Companyname" bắt đầu bằng chữ A

```
<html>
<body>
<%
set conn=Server.CreateObject("ADODB.Connection")
conn.Provider="Microsoft.Jet.OLEDB.4.0"
conn.Open "c:/webdata/northwind.mdb"
set rs=Server.CreateObject("ADODB.recordset")
sql="SELECT Companyname, Contactname FROM Customers
WHERE CompanyName LIKE 'A%"
rs.Open sql, conn
%>
<table border="1" width="100%">
  <tr>
    <%for each x in rs.Fields
      response.write("<th>" & x.name & "</th>")
    next%>
  </tr>
  <%do until rs.EOF%>
    <tr>
      <%for each x in rs.Fields%>
        <td><%Response.Write(x.value)%></td>
      <%next
```



```

    rs.MoveNext%>
  </tr>
<%loop
rs.close
conn.close%>
</table>
</body>
</html>

```

Sort the Data

Hiển thị "Companyname" và "Contactname" từ bản "Customers", sắp xếp bởi "Companyname":

```

<html>
<body>
<%
set conn=Server.CreateObject("ADODB.Connection")
conn.Provider="Microsoft.Jet.OLEDB.4.0"
conn.Open "c:/webdata/northwind.mdb"
set rs = Server.CreateObject("ADODB.recordset")
sql="SELECT Companyname, Contactname FROM
Customers ORDER BY CompanyName"
rs.Open sql, conn
%>

<table border="1" width="100%">
  <tr>
    <%for each x in rs.Fields
      response.write("<th>" & x.name & "</th>")
    next%>
  </tr>
  <%do until rs.EOF%>
    <tr>
      <%for each x in rs.Fields%>
        <td><%Response.Write(x.value)%></td>
      <%next
      rs.MoveNext%>
    </tr>
  <%loop
  rs.close
  conn.close%>
</table>
</body>

</html>

```

Add a Record to a Table in a Database

Chúng ta có thể thêm 1 bản ghi mới vào CSDL Northwind. Trước tiên bạn phải tạo ra 01 form nhập liệu:

```
<html>
<body>
<form method="post" action="demo_add.asp">
<table>
<tr>
<td>CustomerID:</td>
<td><input name="custid"></td>
</tr><tr>
<td>Company Name:</td>
<td><input name="compname"></td>
</tr><tr>
<td>Contact Name:</td>
<td><input name="contname"></td>
</tr><tr>
<td>Address:</td>
<td><input name="address"></td>
</tr><tr>
<td>City:</td>
<td><input name="city"></td>
</tr><tr>
<td>Postal Code:</td>
<td><input name="postcode"></td>
</tr><tr>
<td>Country:</td>
<td><input name="country"></td>
</tr>
</table>
<br /><br />
<input type="submit" value="Add New">
<input type="reset" value="Cancel">
</form>
</body>
</html>
```

Khi bạn bấm vào nút **submit**(add New) sẽ chuyển sang file "demo_add.asp".
"demo_add.asp" chứa đoạn mã thêm 01 record mới vào bản Customers như sau:

```
<html>
<body>
<%
```

```

set conn=Server.CreateObject("ADODB.Connection")
conn.Provider="Microsoft.Jet.OLEDB.4.0"
conn.Open "c:/webdata/northwind.mdb"
sql="INSERT INTO customers (customerID,companyname,"
sql=sql & "contactname,address,city,postalcode,country)"
sql=sql & " VALUES "
sql=sql & "(" & Request.Form("custid") & ","
sql=sql & Request.Form("compname") & ","
sql=sql & Request.Form("contname") & ","
sql=sql & Request.Form("address") & ","
sql=sql & Request.Form("city") & ","
sql=sql & Request.Form("postcode") & ","
sql=sql & Request.Form("country") & ")")
on error resume next
conn.Execute sql,recaffected
if err<>0 then
    Response.Write("No update permissions!")
else
    Response.Write("<h3>" & recaffected & " record added</h3>")
end if
conn.close
%>

</body>
</html>

```

Update a Record in a Table

Bạn muốn cập nhật lại thông tin 1 record trong bản Customers trong CSDL: Northwind. Đầu tiên ta hiển thị tất cả các records trong bảng Customers, mỗi record lấy customerID làm tham số :

```

<html>
<body>
<%
set conn=Server.CreateObject("ADODB.Connection")
conn.Provider="Microsoft.Jet.OLEDB.4.0"
conn.Open "c:/webdata/northwind.mdb"
set rs=Server.CreateObject("ADODB.Recordset")
rs.open "SELECT * FROM customers",conn
%>
<h2>List Database</h2>
<table border="1" width="100%">
<tr>
<%
for each x in rs.Fields

```

```

    response.write("<th>" & ucase(x.name) & "</th>")
next
%>
</tr>
<% do until rs.EOF %>
<tr>
<form method="post" action="demo_update.asp">
<%
for each x in rs.Fields
    if lcase(x.name)="customerid" then%>
        <td>
            <input type="submit" name="customerID" value="<%=x.value%>">
        </td>
    <%else%>
        <td><%Response.Write(x.value)%></td>
    <%end if
next
%>
</form>
<%rs.MoveNext%>
</tr>
<%
loop
conn.close
%>
</table>
</body>
</html>

```

Khi bạn click vào "customerID" sẽ gọi đến file "demo_update.asp". tập tin "demo_update.asp" file này chứa đoạn mã tạo ra form chứa các field từ customerID đã. Form này có 01 nút lệnh "Update record" sẽ thực hiện việc lưu dữ liệu:

```

<html>
<body>
<h2>Update Record</h2>
<%
set conn=Server.CreateObject("ADODB.Connection")
conn.Provider="Microsoft.Jet.OLEDB.4.0"
conn.Open "c:/webdata/northwind.mdb"
cid=Request.Form("customerID")
if Request.form("companyname")="" then
    set rs=Server.CreateObject("ADODB.Recordset")
    rs.open "SELECT * FROM customers WHERE customerID='" & cid & "
%>

```

```

<form method="post" action="demo_update.asp">
<table>
<%for each x in rs.Fields%>
<tr>
<td><%=x.name%></td>
<td><input name="<%=x.name%>" value="<%=x.value%>"></td>
<%next%>
</tr>
</table>
<br /><br />
<input type="submit" value="Update record">
</form>
<%
else
sql="UPDATE customers SET "
sql=sql & "companyname='" & Request.Form("companyname") & "',"
sql=sql & "contactname='" & Request.Form("contactname") & "',"
sql=sql & "address='" & Request.Form("address") & "',"
sql=sql & "city='" & Request.Form("city") & "',"
sql=sql & "postalcode='" & Request.Form("postalcode") & "',"
sql=sql & "country='" & Request.Form("country") & "'"
sql=sql & " WHERE customerID='" & cid & "'"
on error resume next
conn.Execute sql
if err<>0 then
response.write("No update permissions!")
else
response.write("Record " & cid & " was updated!")
end if
end if
conn.close
%>
</body>
</html>

```

Delete a Record in a Table

Bạn muốn xóa 1 record trong bảng Customers trong CSDL Northwind. Trước tiên bạn phải hiển thị ra tất cả các record trong bảng đó để chọn xóa:

```

<html>
<body>
<%
set conn=Server.CreateObject("ADODB.Connection")
conn.Provider="Microsoft.Jet.OLEDB.4.0"
conn.Open "c:/webdata/northwind.mdb"

```

```

set rs=Server.CreateObject("ADODB.Recordset")
rs.open "SELECT * FROM customers",conn
%>
<h2>List Database</h2>
<table border="1" width="100%">
<tr>
<%
for each x in rs.Fields
    response.write("<th>" & ucase(x.name) & "</th>")
next
%>
</tr>
<% do until rs.EOF %>
<tr>
<form method="post" action="demo_delete.asp">
<%
for each x in rs.Fields
    if x.name="customerID" then%>
        <td>
            <input type="submit" name="customerID" value="<%=x.value%>">
        </td>
    <%else%>
        <td><%Response.Write(x.value)%></td>
    <%end if
next
%>
</form>
<%rs.MoveNext%>
</tr>
<%
loop
conn.close
%>
</table>
</body>
</html>

```

Nếu bạn click vào cột "customerID" một file mới được gọi đến "demo_delete.asp". File "demo_delete.asp" chứa đoạn mã hiển thị dữ liệu bạn muốn xóa. Tiếp đó có nút lệnh "Delete record" để xóa record mà bạn chọn:

```

<html>
<body>
<h2>Delete Record</h2>
<%

```

```

set conn=Server.CreateObject("ADODB.Connection")
conn.Provider="Microsoft.Jet.OLEDB.4.0"
conn.Open "c:/webdata/northwind.mdb"
cid=Request.Form("customerID")
if Request.form("companyname")="" then
    set rs=Server.CreateObject("ADODB.Recordset")
    rs.open "SELECT * FROM customers WHERE customerID='" & cid & "'
    %>
    <form method="post" action="demo_delete.asp">
    <table>
    <%for each x in rs.Fields%>
    <tr>
    <td><%=x.name%></td>
    <td><input name="<%=x.name%>" value="<%=x.value%>"></td>
    <%next%>
    </tr>
    </table>
    <br /><br />
    <input type="submit" value="Delete record">
    </form>
<%
else
    sql="DELETE FROM customers"
    sql=sql & " WHERE customerID='" & cid & "'"
    on error resume next
    conn.Execute sql
    if err<>0 then
        response.write("No update permissions!")
    else
        response.write("Record " & cid & " was deleted!")
    end if
end if
conn.close
%>
</body>
</html>

```

TỔNG QUAN VỀ NGÔN NGỮ LT WEB

CÔNG CỤ LẬP TRÌNH WEB HIỆN NAY

- MS(ASP, ASP.NET)
- JAVA SUN (JSP, SERVLET)
- PHP

CÁC THUẬT NGỮ CƠ BẢN

- **HTTP:** Hypertext Transfer Protocol
- **FTP:** File Transfer Protocol
- **HTML:** HyperText Markup Language(**DHTML**, **XHTML**)
- **XML:** eXtensible Markup Language
- **CSS:** Cascading Style Sheets
- **JAVASCRIPT:** (Multi Web Server Service)
- **VBSCRIPT:** (only MS Web Server Service)
- **ASP:** Active Server Pages
- **JSP:** JAVA Server Pages
- **CGI:** Common Gateway Interface
- **PWS:** Personal Web Server
- **IIS:** Internet Information Services
- **DOMAIN NAME**
- **HOSTING**
- **POST**

Một số đối tượng WEB

- **FLASH**
- **APPLET**
- **CSS**

Trong điều kiện và giới hạn của môn học này, chúng ta sẽ được nghiên cứu ngôn ngữ lập trình **JAVASCRIPT** và **ASP**.

**Phần JAVASCRIPT đã được giới thiệu trên
Nội dung cụ thể của NNLT ASP:**

Bài 1: Tổng Quan về NNLT ASP

Bài 2: CÁC THÀNH PHẦN CƠ BẢN

Bài 3: CÁC CẤU TRÚC ĐIỀU KHIỂN

Bài 4: XÂY DỰNG ỨNG DỤNG TRONG ASP

Bài 5: HỆ QUẢN TRỊ CSDL - SQL

Bài 6: KẾT NỐI CSDL - ADO

Bài 7: PHÂN TÍCH VÀ THIẾT KẾ HTTT QUẢN LÝ.