

VBScript

Lê Anh Nhật

Đt: 0912.844.866

Email: leanhnhat@tuyenquang.edu.vn

Chương 1 - Giới thiệu

1.1. Cơ bản về VBScript.

VBScript là gì?

Một Script trong tài liệu HTML nói chung có hai phần:

- + Phần <HEAD> của tài liệu sẽ có một Script Container đặc biệt chứa định nghĩa của các hàm được script dùng.*
- + Lời gọi hàm làm nhiệm vụ truyền và nhận các giá trị.*

1.2. Tại sao dùng VBScript

- *VBScript là tập con của Visual Basic*
- *Làm việc với ActiveX: Mặc dù Microsoft bắt đầu hỗ trợ javascript và các ngôn ngữ Script khác truy cập các ActiveX Component, VBScript vẫn là sự lựa chọn của Microsoft và hầu hết những người ủng hộ ActiveX.*
- *Dùng kỹ thuật Dynamic HTML.*

Chương 2 - Script cơ sở

2.1. Thẻ Script

- Các câu lệnh VBScript được viết trong thẻ:

<Script Language="VBScript">

....

</Script>

2.2. Đối tượng Document

- Để viết ra một chuỗi ta gọi Document với hàm viết ra Write:

`Document.Write(String)`

- Ví dụ:

```
<SCRIPT LANGUAGE="VBScript">
```

```
    Document.Write("Le Anh Nhat")
```

```
</SCRIPT>
```

2.3. VBScript và thẻ HTML

□ Ví dụ:

```
<Script Language="VBScript">  
    Document.Write("Hello<br>")  
</Script>
```

- trong đó
 là thẻ HTML được VBScript viết ra, có nghĩa là xuống dòng.
- Ta có thể dùng lệnh write để viết ra các thẻ của HTML.

2.4. Chú thích trong VBScript

- Chú thích trên 1 dòng ta dùng từ khoá **Rem** ở đầu.
- Ngoài ra có thể dùng dấu nháy đơn (') ở đầu dòng để viết câu chú thích.
- Chú ý khi viết mã chương trình nên thụt đầu dòng thích hợp.

Chương 3 - form và điều khiển web

3.1. Form

Form cho phép bạn nhận thông tin hay phản hồi từ người dùng.

Tạo Form:

<form Method = <phương thức>

Action = “URL” >

<input> yêu cầu thông tin bằng một trong
nhiều cách khác nhau

</form>

<Phương thức>: nhận giá trị *Post* hoặc *Get*

□ Ví dụ:

```
<h2>Employment Application</h2>
<form name="frmEmployees">
<table border="0" width="288"> <tr> <td width="80">First
Name:</td> <td width="194"></td> </tr> <tr> <td
width="80">Last Name:</td> <td width="194"></td> </tr>
<tr> <td width="80">Full Name:</td> <td
width="194"></td> </tr>
</table> </form>
```

Sự kiện trên Form

| Tên sự kiện | Chức năng |
|--------------------|--|
| OnClick | Phản ứng khi người dùng Click chuột lên một thành phần của Form |
| OnFocus | Phản ứng khi người dùng đặt Focus vào thành phần nào đó |
| OnBlur | Phản ứng khi người dùng chuyển Focus ra khỏi thành phần nào đó. |
| OnChange | Phản ứng khi người dùng thay đổi dữ liệu hay thay đổi vùng chọn trong thành phần nào đó. |
| OnSelect | Phản ứng khi người dùng chọn nội dung của thành phần nào đó. |

3.2. Text box

- Cú pháp

`<input type="text" ...>`

- Các thuộc tính hay dùng:

- size - kích thước.
- MaxLength – Kích thước tối đa
- value – giá trị ghi trong textbox
- name - đặt tên cho text box.

- Các sự kiện

- onFocus, onKeyUp, onKeyDown, onKeyPress, onChange, onBlur.

□ Ví dụ

```
<input
  type="text"
  name="txtFirstName"
  size="10"
  onChange="form.txtFullName.value =
form.txtFirstName.value + ' ' +
form.txtLastName.value"
>
```

3.3. Nút nhấn

- Tạo nút nhấn
`<input type="button">`
- Các thuộc tính của nút nhấn
 - value - tên hiển thị trên nút.
 - name - tên đặt của nút nhấn.
- Các sự kiện
 - onClick - khi click chuột vào nút nhấn.

□ Ví dụ:

```
<input
  type="button"
  value="Evaluate"
  name="btnEvaluate"
  onClick="form.txtFullName.value =
form.txtLastName.value + ', ' +
form.txtFirstName.value"
>
```

3.4. Nút submit

- Dạng:

```
<input type="submit">
```

- Thuộc tính

- Value = string: dòng chữ ghi trên nút

- Sự kiện:

- onClick.

3.5. Nút Reset

- Dạng

`<input type="reset">`

- Các thuộc tính khác

- value – giá trị ghi trên nút bấm
- name – tên của nút bấm

□ Ví dụ:

```
<input      type="reset"  
           value="Clear Application"  
           name="btnReset"  
>
```

3.6. Vùng văn bản

- Cú pháp

`<textarea>`

Text...

`</textarea>`

- Thuộc tính

- cols - số cột

- rows - số hàng

- Text: hướng dẫn người sử dụng nhập thông tin

3.7. Hộp mật khẩu

- Cú pháp

`<input type="password">`

- Các thuộc tính

- size - kích thước
- MaxLength – Kích thước tối đa
- name - tên
- value – giá trị

3.8. Tạo hộp kiểm tra

- Cú pháp

`<input type="checkbox">` Sting

- Các thuộc tính

- checked - xác định hộp được đánh dấu
- UnChecked - hộp chưa được đánh dấu

- name – Tên

- String - chuỗi thông báo.

- Sự kiện

- onClick.

3.9. Nút chọn (Radio)

- Cú pháp

`<input type="radio"...>String`

- Các thuộc tính

- checked - đã chọn hay không, có giá trị true hoặc false.
- name – tên
- Value: Giá trị
- String – xâu thông báo

*Thuộc tính **Name** phải giống nhau và thuộc tính **Value** phải khác nhau*

- Các sự kiện

- onClick.

3.10. Hộp combo

- Cú pháp

`<select>`

`<Option Selected Value = “gia trị 1”> String1`

`<Option Value = “gia trị 2”> String2`

`<Option Value = “gia trị 3”> String3`

...

`</select>`

- Thuộc tính

- `selected` trong option, giá trị true, false.

3.11. Hộp danh sách

- Cú pháp

```
<select size="4">
```

```
  <option>...
```

```
  <option>...
```

```
  ...
```

```
</select>
```

- Thuộc tính

- selected trong option, giá trị true, false.

-
- multiple trong select - chọn 1 hay nhiều dòng.
 - Sự kiện
 - onChange -

3.12. Trường Hidden

**<Input Type = Hidden Name = “name”
Value = “giá trị”>**

Được thiết kế để truyền (ngầm) giá trị đến Web Server và Script. Giá trị truyền thường là một từ khoá, giá trị kiểm tra hay một chuỗi bất kỳ

Chương 4 - Các biến

4.1. Khai báo biến

- Có thể khai báo không cần từ khoá, ví dụ:
`<SCRIPT LANGUAGE="VBScript">`
NumberOfStudents
`</SCRIPT>`
- Hay sử dụng từ khoá **Dim**, ví dụ
`<SCRIPT LANGUAGE="VBScript">`
Dim NumberOfStudents
`</SCRIPT>`

* Khai báo biến mảng:

+ Mảng 1 chiều:

Dim <Tên biến mảng>(số phần tử)

Ví dụ: **Dim A(50)**

+ Mảng 2 chiều:

Dim <Tên biến mảng>(số dòng, số cột)

Ví dụ: **Dim B(10, 20)**

+ Khai báo mảng có phần tử cố định:

Dim <Tên biến mảng>(phần tử 1, phần tử 2,....)

+ Khai báo mảng động: **Dim <Tên biến mảng>()**

Ví dụ: **Dim mang()**

+ Có thể khai báo lại mảng động:

ReDim <Tên biến mảng>(số phần tử)

* Khai báo hằng:

Const <Tên_hằng> = <giá trị>

- Với <giá trị>:
 - Kiểu xâu ký tự: Để trong “...”
 - Kiểu ngày: Để trong #....#
 - Kiểu Logic: True hoặc False
 - Kiểu số:

Chương 5 - Sự hoạt động

5.1. Sự hoạt động đầy đủ

- Phép gán =
tên_bến = biến
 - Ví dụ:

```
<Script Language="VBScript">  
    Dim Salary  
    Salary = 12.55  
    Document.Write(Salary);  
</Script>
```

□ Phép cộng +

- Sử dụng cộng các biến: value1+value2
- Ví dụ

```
<SCRIPT LANGUAGE="VBScript">  
    Document.Write( 125 + 48)  
</SCRIPT>
```

□ Nối chuỗi: &

- Có thể nối các chuỗi với nhau nhờ phép &, ví dụ: value1 & value2
- Ví dụ:

```
<Script Language="VBScript">  
    Option Explicit  
    Dim FirstName, LastName  
    FirstName = "Roger"  
    LastName = "Lemerre"  
    document.write(FirstName & LastName)  
</Script>
```

□ Phép trừ (-)

- Ta có thể trừ hai biến cho nhau:
value1 - value2
- Ví dụ:

```
<SCRIPT LANGUAGE="VBScript">  
    Document.Write( 208.55 - 42.16)  
</SCRIPT>
```

□ Phép nhân (*)

- Có thể nhân hai biến với nhau
value1 * value2
- Ví dụ:

```
<Script Language="VBScript">  
    Document.Write(15.55 * 240.25)  
</Script>
```

-
- Phép chia số nguyên (\)
 - Hai số nguyên có thể chia cho nhau cho kết quả nguyên:
 $kq = \text{value1} \ \backslash \ \text{value2}$

□ Phép chia (/)

- Có thể chia hai biến cho nhau:
 $kq = \text{value1} / \text{value2}$
- Ví dụ:

```
<Script Language="VBScript">  
    Document.Write("Daily Earnings: ")  
    Document.Write(425.65 / 7)  
</Script>
```

-
- Phép chia lấy dư
 - Sử dụng Mod
$$kq = \text{value1 Mod value2}$$
 - Ví dụ:

□ Sự âm

- Dùng dấu - để thể hiện số âm.
- Ví dụ:

```
<Script Language="VBScript">  
    Dim Temperature  
    Temperature = -32  
</Script>
```

-
- Luỹ thừa: ^
 - Dùng ^ để chỉ x^y :
 x^y
 - Ghép dòng ta sử dụng (_) để chỉ dòng chưa kết thúc.

□ Ví dụ

```
<Script Language="VBScript">
  Dim dblHours
  Dim dblMonday, dblTuesday, dblWednesday, _
dblThursday, dblFriday, dblSaturday, dblSunday
  Dim dblSalary
  Dim dblResult
  dblHours = dblMonday + dblTuesday + _
dblWednesday + dblThursday + _
dblFriday + dblSaturday + dblSunday
  dblSalary = txtSalary
  dblResult = dblHours * dblSalary
</Script>
```

Chương 6 - Thủ tục và hàm

6.1. Thủ tục cơ bản

□ Thủ tục sub

■ Cú pháp

SUB <tên_thủ_tục>([ds tham số])

 Thân thủ tục

END SUB

■ Ví dụ

```
Sub DisplayFullName()  
    Dim FirstName, LastName  
    Dim FullName  
    FullName = FirstName & " " & LastName  
End Sub
```


-
- Gọi một thủ tục
 - Gọi tên thủ tục đã tạo ra
`CALL <tên thủ tục>`
 - Ví dụ:

```
Sub Detailer()  
    DisplayFullName  
End Sub
```

6.2. Các đối số

- Sự kết thúc một đối số
 - Thủ tục có thể không có đối số, hoặc có thể nhiều hơn hoặc bằng 1 đối số, ví dụ

```
Sub CalculateArea(Radius)
    Dim dblPI
    Dim dblArea
    dblPI = 3.14159
    dblArea = Radius * Radius * dblPI
End Sub
```

-
- Gọi một thủ tục có đối số

```
Sub Result()  
    Dim dblHours, dblSalary  
    CalcAndShowSalary dblHours, dblSalary  
End Sub  
Sub CalcAndShowSalary(Hours, Salary)  
    Dim dblResult  
    dblResult = Hours * Salary  
    txtResult = dblResult  
End Sub
```

-
- Hoặc sử dụng từ khoá **Call**, ví dụ

```
Sub Result()  
    Dim dblHours As Double  
    Dim dblSalary As Double  
    dblHours = txtHours  
    dblSalary = txtSalary  
    Call CalcAndShowSalary(dblHours, dblSalary)  
End Sub
```

6.3. Hàm

□ Tạo một hàm

■ Cú pháp

FUNCTION <tên_hàm>(ds tham số)

Thân hàm

END FUNCTION

Ví dụ

```
Function CalculateArea(Radius)
    CalculateArea=Radius*Radius*3.14159
End Function
```

-
- Gọi hàm như gọi thủ tục
 - $\langle \text{Biến} \rangle = \langle \text{Tên hàm} \rangle (\text{các tham số})$

6.4. Các hàm đã có

- $\text{Abs}(x) = |x|$
- Array - cho một biến bao gồm một mảng.
- Asc - trả về mã ký tự ANSI, tương ứng với ký tự đầu tiên trong chuỗi.
- Atn - tính tang của 1 số.
- CBool - dạng boolean phụ.
- CByte - dạng Byte phụ.

- Các hàm chuyển đổi kiểu dữ liệu

| Hàm | Ý nghĩa |
|--------------|-----------------------------|
| Cdate | Chuyển sang kiểu ngày tháng |
| Cint | Chuyển sang kiểu Integer |
| Cstr | Chuyển sang kiểu string |
| Cbyte | Chuyển sang kiểu byte |
| Cdbl | Chuyển sang kiểu Double |
| Cbool | Chuyển sang kiểu Boolean |

- Các hàm thao tác với chuỗi

| Tên hàm | Ý nghĩa |
|--------------------------|---|
| Len (string) | Lấy chiều dài chuỗi |
| Ucase(string) | Chuyển chuỗi thành chữ in hoa |
| Lcase(string) | Chuyển chuỗi thành chữ in thường |
| Ltrim(string) | Cắt bỏ các khoảng trống bên trái chuỗi |
| Rtrim(string) | Cắt bỏ các khoảng trống bên phải chuỗi |
| Trim | Cắt bỏ các khoảng trống cả hai bên chuỗi |
| Left(string, n) | Trích ra chuỗi con có n ký tự từ bên trái của chuỗi |
| Mid(string, n, m) | Trích ra chuỗi con có m ký tự từ vị trí n của chuỗi |
| Right(string, n) | Trích ra chuỗi con có n ký tự từ bên phải của chuỗi |

- Các hàm ngày tháng

| Tên hàm | Ý nghĩa |
|---------|----------------------|
| Date | Ngày hiện tại |
| Time | Giờ hiện tại |
| Now | Ngày và giờ hiện tại |

- Các hàm kiểm tra

- * **Isdate**: Kiểm tra có phải đúng kiểu ngày tháng không?
- * **IsNumeric**: Kiểm tra có phải đúng kiểu số không?

-
- `IsEmpty` - cho giá trị boolean nếu biến đã được tạo.
 - `IsNull` - cho giá trị boolean nếu biến chưa có dữ liệu.
 - `IsNumeric(x)` - cho giá trị true nếu x là biến số, false còn lại.
 - `IsObject` - cho giá trị boolean nếu biến là đối tượng tự động.

-
- Join -
 - LBound -
 - LCase -
 - Left -
 - Len - trả về số ký tự trong 1 chuỗi.
 - LoadPicture - trả về đối tượng ảnh.
 - Log - logarit tự nhiên của 1 số.

-
- LTrim -
 - Mid -
 - Minute - trả về phút, 1..59.
 - Month - trả về tháng, 1..12.
 - MonthName -
 - MsgBox - Hiển thị thông báo ở một hộp dialog
 - Now - cho ngày và thời gian hiện tại.

-
- Oct - đổi số thành chữ.
 - Replace -
 - RGB - về màu sắc.
 - Right -
 - Rnd - sinh số ngẫu nhiên.
 - Round - làm tròn số.
 - RTrim -

-
- ScriptEngineBuildVersion -
 - ScriptEngineMajorVersion -
 - ScriptEngineMinorVersion -
 - Second - số giây của phút, 1..59.
 - SetLocale -
 - Sgn - lấy phần nguyên.
 - Sin - Sine của một góc.

-
- Space -
 - Split -
 - Sqr(x) = căn bậc hai của x.
 - StrComp -
 - Tan - tang của một góc.
 - Time -
 - Timer -

-
- Weekday -
 - WeekdayName - ngày của tuần.
 - Year - số năm.

6.5. Các phép toán

- Phép gán =
 - Ví dụ:
value1 = value2
- Phép so sánh khác <>
 - Ví dụ:
value1 <> value2
- Phép so sánh nhỏ hơn <
 - Ví dụ:
value1 < value2

6.6. Điều kiện If...Then

IF <điều kiện> THEN

<dãy lệnh>

IF <điều kiện> THEN

<dãy lệnh1>

ELSE

<dãy lệnh 2>

END IF

□ Ví dụ

```
Sub CheckPassword()  
    Dim Password, ConfPassword, Result  
    Password = Document.frmRegistration.txtPassword.Value  
    ConfPassword = _  
        Document.frmRegistration.txtConfirmPass.Value  
    If Password <> ConfPassword Then  
        Document.frmRegistration.txtResult.Value="Your_ Passwords Do  
Not Match"  
    End If  
End Sub
```

6.7. Lệnh lựa chọn **Select case**

SELECT CASE <biểu thức>

 CASE <giá trị 1>

 <dãy lệnh 1>

 CASE <giá trị 2>

 <dãy lệnh 2>

.....

 [CASE ELSE

 <dãy lệnh n>]

END SELECT

6.8. Lặp Do ... Loop

Dạng 1: Đúng thì lặp, sai thì thoát ra ngoài

Do While điều kiện

các câu lệnh

Loop

Do

các câu lệnh

Loop While điều kiện

Dạng 2: Sai thì lặp, đúng thì thoát ra ngoài

Do Until điều kiện

các câu lệnh

Loop

Do

các câu lệnh

Loop Until điều kiện

6.9. Lặp For ... Next

a, **FOR...NEXT**

FOR <biến đếm> = <giá trị đầu> **TO** <giá trị cuối>
[**STEP** <bước nhảy>]

<dãy lệnh 1>

[**EXIT FOR**]

<dãy lệnh 2>

NEXT <biến đếm>

6.10. For Each ... Next

- Cú pháp

For Each phần_tử **In** Tập_hợp

các câu lệnh

Next phần_tử