

MỤC LỤC

MỤC LỤC.....	1
PHẦN 1: CƠ BẢN VỀ HTML VÀ VBSCRIPT.....	2
I. CÁC THEỂ ĐỂ TẠO PHẦN TỬ HTML CƠ BẢN.....	2
1. Cú pháp chung:	2
2. Tạo một số phần tử cơ bản	2
II. NGÔN NGỮ VBSCRIPT	5
1. Nhúng câu lệnh VBSCRIPT vào trang web.....	5
2. Gắn kết mã lệnh VBSCRIPT lên các sự kiện	5
3. Ghi và đọc dữ liệu trong TextBox	6
4. Đọc dữ liệu trong ComboBox: Giống nh- Listbox	6
5. Đọc / Ghi giá trị trong CheckBox	6
PHẦN II : ASP	7
ĐỐI TƯỢNG RESPONSE	7
1. Chức năng	7
2. Một số ph- ơng thức dùng:	7
2.1 Ph- ơng thức write	7
2.2 Ph- ơng thức Redirect.....	7
ĐỐI TƯỢNG REQUEST	8
1. Chức năng	8
2. Một số thuộc tính và ph- ơng thức th- ờng dùng.....	8
2.1 Thuộc tính Form.....	8
2.2 Thuộc tính QueryString.....	11
ĐỐI TƯỢNG SERVER.....	15
1. Ph- ơng thức Mappath	15
2. Ph- ơng thức CreateObject.....	17
ĐỐI TƯỢNG SESSION – BIẾN SESSTION	17
1. Biến Sessstion	17
2. Đối t- ợng Session	17
ĐỐI TƯỢNG APPLICATION	20
1. Khái niệm biến toàn ứng dụng	21
2. Đối t- ợng Application.....	21
3. Ví dụ tổng hợp	21
CHÈN FILE VÀO TRANG ASP.....	22
1. ý nghĩa:.....	22
2. Cú pháp chèn file:	22
3. Ví dụ tổng hợp	22
KẾT NỐI VÀ THAO TÁC VỚI CSDL TRONG ASP.....	24
1. Qui tắc chung	24
2. Tạo đối t- ợng Connection và kết nối đến CSDL.....	24
3. Tạo đối t- ợng Recordset	25
4. Một số ví dụ	25
4.1 Hiện thị toàn bộ tên nhà cung cấp trong bảng Suppliers	25
4.2.1 Hiện thị danh sách các sản phẩm trong một bảng.....	27
PHỤ LỤC 1.....	28
PHỤ LỤC 2.....	32
DANH SÁCH CÁC HÀM TRONG VBSCRIPT	32
PHẦN III: THỰC HÀNH.....	35
BÀI THỰC HÀNH SỐ 1.....	35
BÀI THỰC HÀNH SỐ 2.....	37
BÀI THỰC HÀNH SỐ 3.....	42
BÀI THỰC HÀNH SỐ 4.....	46

PHẦN 1: CƠ BẢN VỀ HTML VÀ VBSCRIPT

I. CÁC THẺ ĐỂ TẠO PHẦN TỬ HTML CƠ BẢN

1. Cú pháp chung:

- ❖ `<Tên_Loại_Phần_Tử <Thuộc tính 1> = "Giá trị" <Thuộc Tính> = "Giá trị"... >`
- ❖ `<Tên_Loại_Phần_Tử Style = "Thuộc_tính: giá_trị; thuộc_tính : giá_trị ;....; >`

Trong đó :

Tên loại phần tử	Thuộc tính
Button	Name
Text	VALUE
File	MAXLENGTH
Hidden	ReadOnly
Select	Disable
TextArea	Cols, Rows
CheckBox	Multiple
	TYPE

Phần “giá trị” có thể đặt trong cặp ngoặc kép hoặc cặp ngoặc đơn !

Nếu đặt thuộc tính theo cách 2, thì có thể tham chiếu bảng sau (gọi là theo cú pháp CSS)

2. Tạo một số phần tử cơ bản

- ❖ Tạo nút nhấn

`<Input name="KiemTra" TYPE="button" VALUE="Kiểm tra dữ liệu">`

`<Input TYPE="Submit" VALUE="Đăng nhập">`

- ❖ Tạo ô nhập

`<Input name="HoVaTen" TYPE="text" VALUE="Nguyễn Văn A" size="20" MAXLENGTH="30">`

`<Input name="Khoa" TYPE="text" VALUE="Khoa Công nghệ thông tin" size="40" MAXLENGTH="50" readonly="true">`

`<Input name="Truong" TYPE="text" VALUE="Trường Đại học sư phạm kỹ thuật Hưng Yên" size="40" MAXLENGTH="50" Disabled="true">`

- ❖ Tạo vùng nhập

`<textarea name="GhiChu" cols="50" rows="5"> Nội dung ghi chú: </textarea>`

- ❖ Tạo ô nhập Password

`<Input name="MatKhau" type="password" value="123456" size="10" maxlength="20">`

- ❖ Tạo listbox

```
<select name="MonHoc" size="5" >
  <option>Visual Basic</option>
  <option>Lập trình .NET</option>
  <option>Lập trình ASP</option>
```

</select>

❖ Tạo ComboBox (chỉ cần bỏ thuộc tính size)

```
<select name="Mon" size = 1 onChange="Call DocGiaTriListBox">
    <option value = "Visual Basic">Visual Basic</option>
    <option value = "DOT_NET">Lập trình .NET</option>
    <option value = "ASP">Lập trình ASP</option>
</select>
```

❖ Tạo hộp kiểm

```
<Input name="chkVB" type="checkbox" value="Visual Basic" checked> Visual Basic
<Input name="chkASP" type="checkbox" value="ASP "> Active Server Pages
```

❖ Tạo nút Radio

```
<Input name="GiớiTinh" type="radio" value="Nam" checked>
<Input name="GiớiTinh" type="radio" value="Nữ" >
```

```
<Input name="TìnhTrang" type="radio" value="Đã lập gia đình" >
<Input name="TìnhTrang" type="radio" value="Độc thân" checked >
```

Tên giống nhau thì sẽ thuộc về cùng một nhóm (Groups)

❖ Phần tử chọn File

```
<Input name="ChonFile" type="file" size="30">
```

❖ Tạo textbox ẩn (Hidden).

```
<Input name="PhanTuAn" type="hidden" value="">
```

Tạo các phần tử và đặt thuộc tính:

+ Tạo một textbox và đặt thuộc tính font:

```
<FONT FACE = "Times New Roman">
    <Input type = text value = "Font chữ Unicode đây !">
</FONT>
```

+ Tạo một textbox và đặt thuộc tính thông qua phong cách CSS:

```
<Input type = text value = "Font Unicode" Style = "Font-Family:Times new roman">
```

+ Tạo một nhãn có font chữ xanh, có hiệu ứng:

```
<P Style ="Color:Blue; font-size:20; Text-Align:center">Xin chào </p>
```

+ Tạo một nút nhấn có màu nền đỏ:

```
<Input type = button style="font-family:arial; background-color:red" value = "Đỏ">
```



Kết quả

II. NGÔN NGỮ VBSCRIPT

1. Những câu lệnh VBSCRIPT vào trang web

Sử dụng thẻ <script Language = VBScript>

Ví dụ 1:

```
<script language="VBScript">
    MsgBox "Xin chào các bạn lớp TK34 !"
</SCRIPT>
```

Ví dụ 2:

```
<script language="VBScript">
    Sub Chao
        MsgBox "Xin chào các bạn lớp TK34!"
    End Sub

    '/// Gọi thủ tục Chao. Viết là Call Chao hoặc Chao đều được
    Call Chao
</SCRIPT>
```

Ví dụ 3:

```
<script language="VBScript">
    Function Tong(a, b)
        Tong = a + b
    End Function

    MsgBox "Tổng của 2 + 3 = " & Tong(2,3)
</SCRIPT>
```

2. Gắn kết mã lệnh VBSCRIPT lên các sự kiện

Mỗi phần tử trên form có thể có các sự kiện như OnClick, OnChange, OnKeyUp ...

Khi đó ta có thể yêu cầu trình duyệt thực thi các câu lệnh hoặc thủ tục VBScript khi các sự kiện này xảy ra.

Ví dụ 1: Đưa ra lời chào khi người dùng click vào nút nhấn:

```
<Input type = Button Value = "Click here..." OnClick = 'MsgBox "Xin chào lớp TK34 !" '>
```

Ví dụ 2: Thực hiện đưa ra lời chào khi người dùng click bằng việc gọi thủ tục

```
<script language="VBScript">
    Sub Chao
        MsgBox "Xin chào các bạn lớp TK34!"
    End Sub
</SCRIPT>

<Input type = Button Value = "Click here..." OnClick = "Chao"> HOẶC
<Input type = Button Value = "Click here..." OnClick = "Call Chao">
```

I. Đọc và ghi dữ liệu trong các phần tử HTML

Nhìn chung, để đọc hay thay đổi nội dung của một phần tử HTML, thông thường ta đọc hay thay đổi thuộc tính Value của nó. Cú pháp là : <Tên>.Value [= Giá trị mới]

3. Ghi và đọc dữ liệu trong TextBox

```
<Input type = Text name = "HoVaTen" Value = "Nguyễn Văn A">
<script language="VBScript">
    HoVaTen.Value = "Trần Thị B"
    MsgBox "Giá trị trong hộp HoVaTen bây giờ là : " & HoVaTen.Value
</SCRIPT>
```

1. Đọc dữ liệu trong Listbox

```
<script language="VBScript">
    '/// thủ tục này được gọi khi người dùng Click chọn ListBox
    Sub DocGiaTriListBox
        Dim MucChon
        MucChon = Mon.Value
        MsgBox "Bạn đã chọn mục : " & MucChon
    End Sub
</SCRIPT>

<select name="Mon" size = 1 onChange="Call DocGiaTriListBox">
    <option value = "Visual Basic">Visual Basic</option>
    <option value = "DOT_NET">Lập trình .NET</option>
    <option value = "ASP">Lập trình ASP</option>
</select>

<Input type="button" value="Đọc giá trị trong Listbox" onClick="DocGiaTriListBox">
```

4. Đọc dữ liệu trong ComboBox: Giống như Listbox

5. Đọc / Ghi giá trị trong CheckBox

```
<HTML>
<BODY>

<script language=vbscript>
    Sub HT
        IF vb.checked=true Then MsgBox "VB đã được chọn"
        IF asp.checked=true Then msgbox "ASP đã được chọn"
    End Sub
</SCRIPT>

<Input type=button value ="Click đây !" onclick="HT" />
VB : <Input name="VB" type=checkbox checked /> <br />
ASP <Input name ="ASP" type=checkbox /> <br />

</BODY>
</HTML>
```

PHẦN II : ASP

ĐỐI TƯỢNG RESPONSE

1. Chức năng

Dùng để trả kết quả về cho máy khách (Trình duyệt)

2. Một số phương thức dùng:

2.1 Phương thức write

Viết kết quả trả về cho trình duyệt (máy khách).

+ Cú pháp : **Response.write(<Xâu>)**

Trong đó <Xâu> là một xâu ký tự, có thể chứa các thẻ HTML.

+ Ví dụ:

Gửi xâu "Hello world" về cho trình duyệt.

```
<%@ Language = vbscript %>
<HTML>
<BODY>
<%
    Response.write("Hello world")
%>
</BODY>
</HTML>
```

2.2 Phương thức Redirect

Hướng người dùng sang một trang web khác. Ví dụ khi người dùng gõ www.abc.com/Trang1.asp thì chúng ta có thể hướng sang trang Trang2.asp để hiển thị.

+ Cú pháp:

Response.Redirect URL

+ Ví dụ:

Khi người dùng nhập vào trang Trang1.asp, ta sẽ hiển thị trang Trang2.Asp. Khi đó nội dung của trang 1 sẽ như sau:

```
<%@ Language = vbscript %>
<HTML>
<head>
<meta http-equiv="Content-Type" content="text/HTML; charset=utf-8">
<title>Redirect</title>
</head>
<BODY>
    .... Bạn phải thăm trang2.asp trước đã ! <BR>
    .....
    Response.redirect "Trang2.asp"
    .....
%>
</BODY>
</HTML>
```

Một ví dụ cần sử dụng đến phương thức này đó là, khi người dùng chưa thực hiện đăng nhập và vào ngay trang thao tác với CSDL thì ta cần redirect đến trang đăng nhập.

ĐỐI TƯỢNG REQUEST

Vấn đề giao tiếp (nhập xuất) giữa ng-ời dùng và webserver.

Mô hình nhập liệu từ ng-ời dùng và tiếp nhận dữ liệu của webserver: 2 con đ-ờng.

1. POST: Gửi cả một form nhập dữ liệu tới webserver
2. GET : Gửi thông qua liên kết URL

1. Chức năng

Nhận về các giá trị khi ng-ời dùng **Submit** đi. (Ví dụ khi ng-ời dùng điền các thông tin đăng ký tài khoản và nhấn vào nút **Submit** thì thông tin đó sẽ đ-ợc nhận về thông qua đối tượng **Request**).

Có hai cách để ng-ời dùng gửi thông tin đi:

Cách 1: Ng-ời dùng nhập thông tin trong các textbox, sau đó gửi các thông tin trong các textbox này tới một trang asp khác. Khi đó người ta gọi cách gửi đó là gửi theo kiểu “POST”.

Cách 2: Thông tin có thể gửi tới một trang khác bằng cách gắn vào sau mỗi URL của trang đó. Ví dụ để gửi hai thông tin là HoTen và Tuoi với giá trị t-ơng ứng là TK34 và 2 cho trang XuLy.asp thì có thể thực hiện nh- sau:

XuLy.asp?HoTen=TK34&Tuoi=2

Cú pháp để gắn vào sau trang nhận nh- sau:

TrangNhan.asp ? Thuộc_Tính 1 = Giá trị1 & Thuộc_Tính 2 = Giá trị 2 &

Cách gửi này gọi là phương thức gửi kiểu “GET”

2. Một số thuộc tính và phương thức th-ờng dùng

2.1 Thuộc tính Form

Dùng để lấy giá trị của từng thuộc tính mà trang khác gửi tới bằng ph-ương thức POST

Cú pháp:

Request.Form(“Tên_Thuộc_Tính_Cần_Lấy”)

Ví dụ: Trang1.asp có 2 ô textbox tên là HoTen và Tuoi. cho phép ng-ời dùng nhập vào và khi ng-ời dùng gửi đi (**Submit**) tới trang Trang2.asp. thì trang2.asp sẽ hiển thị lời chào tới ng-ời đó. Nội dung 2 file này sẽ nh- sau:

Trang1.asp

```
<HTML>
<head>
<meta http-equiv="Content-Type" content="text/HTML; charset=utf-8">
<title>Gửi dữ liệu sang trang khác</title>
</head>

<BODY>
Trang nhập thông tin : <br>
<form name="NhapLieu" action="Trang2.asp" method="post">
    Họ và tên :    <Input type="text" name="HoTen">          <br>
    Tuổi :        <Input type="text" name="Tuoi">            <br>

    <Input type="Submit" value="Gửi đi" name="cmdGui">
</form>

</BODY>
</HTML>
```


Trang2.asp

```

<HTML>
<head>
<meta http-equiv="Content-Type" content="text/HTML; charset=utf-8">
<title>Nhận dữ liệu gửi từ trang khác </title>
</head>

<BODY>
<%
    Dim HT , Tuoi
    HT = Request.Form(□HoTen□)
    Tuoi = Request.Form(□Tuoi□)
    Response.write(□Chào bạn □ & HT &□. Tuổi của bạn : □ & Tuoi)
%>
</BODY>
</HTML>

```

Chú ý:

- + Muốn gửi thông tin trong các điều khiển đi thì các điều khiển đó phải đặt trong một Form nào đó.
- + Trang web tiếp nhận thông tin đó sẽ đ- ọc đặt trong thuộc tính Action của Form.
- + Ph- ơng thức gửi đi đ- ọc chỉ định thông qua thuộc tính method. Đặt cho thuộc tính này giá trị hoặc là POST hoặc là GET.
- + Một trang có thể gửi (Post) đến chính trang đó. Điều này th- ờng đ- ọc áp dụng khi nội dung trang web là đơn giản (ví dụ trang Login), còn đối với các trang phức tạp (nhiều xử lý) thì nên gửi đến một trang riêng.

Ví dụ 2: Cho ng- ời dùng nhập vào năm sinh, sau đó tính toán và hiển thị tuổi của ng- ời đó. Cách thức ở đây là : Trang cho ng- ời dùng nhập năm sinh và trang để hiển thị kết quả là giống nhau.

Trang TinhTuoi.asp - Version 1

```

<%@language = vbscript %>
<HTML>
<head>
<meta http-equiv="Content-Type" content="text/HTML; charset=utf-8">
<title>Tinh tuoi</title>
</head>
<BODY>

<%
    IF Request.Form("NamSinh")="" THEN
        Response.write ("<form name=Form1 action='TinhTuoi.asp' method=POST>")
        Response.write (□Nhập nam sinh của bạn : <Input type=text name='NamSinh'>")
        Response.write (□<Input type=Submit value='Gui đi'>")
        Response.write ("</form>")
    ELSE
        Response.write ("Tuoi của bạn là : " & (year(now) - Request.Form("NamSinh")))
    END IF
%>

</BODY>
</HTML>

```

Trang TinhTuoi.asp - Version 2

```
<%@language = vbscript %>
<HTML>
<head>
<meta http-equiv="Content-Type" content="text/HTML; charset=utf-8">
<title>Tinh tuoi</title>
</head>
<BODY>

<%
    IF Request.Form("NamSinh")="" THEN
%>
    <FORM name=Form1 action="TinhTuoi.asp" method=POST>
        Nhap nam sinh cua ban : <Input type=text name="NamSinh" >
        <Input type=Submit value="Gui di">
    </FORM>
<%
    ELSE
        Response.write("Tuoi cua ban la : " & (year(now) - Request.Form("NamSinh")))
    END IF
%>
</BODY>
</HTML>
```

Version 3 — Sử dụng các trang xử lý riêng biệt

- ý tưởng: Ta tách làm 2 trang, một trang "Chuyên" hiển thị form nhập năm sinh, sau đó gửi tới một trang "chuyên" xử lý và hiển thị kết quả.
- Trang hiển thị form nhập do không có xử lý gì, do vậy ta chỉ cần tạo trang đó là trang HTML thông thường — **NhapNamSinh.htm**.
- Trang xử lý do phải dùng các câu lệnh ASP nên ta sẽ tạo trang ASP- TinhTuoi.asp

Trang NhapNamSinh.htm

```
<HTML>
<head>
<meta http-equiv="Content-Type" content="text/HTML; charset=utf-8">
<title>Nhap nam sinh </title>
</head>

<BODY>

    <FORM Action="TinhTuoi.asp" method=POST>
        Nhap nam sinh cua ban : <Input type=text name=NamSinh>
        <Input type=Submit value="Gui di">
    </FORM>

</BODY>
</HTML>
```

Trang TinhTuoi.asp

```
<%@language = vbscript %>
<HTML>
<head>
```

```

<meta http-equiv="Content-Type" content="text/HTML; charset=utf-8">
<title>Tinh tuoi</title>
</head>
<BODY>

<%
  IF Request.Form("NamSinh")= "" THEN
    Response.Redirect("NhapNamSinh.htm")
  ELSE
    Response.write ("Tuoi cua ban la: " & (year(now)-Request.Form("NamSinh")))
  END IF
%>

</BODY>
</HTML>

```

Cách (version) nào dễ dàng hơn đối với bạn ? 3 version có - u và nh- ọc điểm gì ? rút KL !

2.2 Thuộc tính QueryString

Dùng để đọc dữ liệu do trang khác gửi tới thông qua ph- ơng thức GET (Tức là gửi dữ liệu bằng cách gắn vào ngay sau liên kết - URL).

+ Cú pháp đọc dữ liệu gửi đến thông qua ph- ơng thức này nh- sau:

Request.QueryString("Tên")

+ Ví dụ 1 : Gửi tới trang TinhTuoi.ASP năm sinh có giá trị là 1980 ở dạng GET.

Khi đó, Hyperlink gửi đi có dạng :

<http://www.tk34.com/TinhTuoi.asp?NamSinh=1980>

Dòng địa chỉ trên có thể tạm dịch là "Gửi một thông tin có tên là NamSinh với giá trị 1980 đến trang TinhTuoi.ASP"

Để tiếp nhận và xử lý dữ liệu này, nội dung trang **TinhTuoi.asp** sẽ nh- sau:

TinhTuoi.asp - Version 1

```

<%@language = vbscript %>
<HTML>
<head>
<meta http-equiv="Content-Type" content="text/HTML; charset=utf-8">
<title>Tinh tuoi</title>
</head>
<BODY>

<%
  IF Request.QueryString("NamSinh")= "" THEN
    Response.write ("Bạn gõ địa chỉ có dạng: Domain/TinhTuoi.asp?NamSinh=So")
  ELSE
    Response.write ("Tuổi của bạn: " & (year(now)-Request.QueryString("NamSinh")))
  END IF
%>

</BODY>
</HTML>

```

Với phiên bản này, rõ ràng khi muốn gửi dữ liệu cho một trang chúng ta phải gõ một cách thủ công !

➔ Có thể gửi theo cách này nh- ng dễ dàng đ- ọc hơn không ? Ví dụ nh- cho ng- ời dùng nhập vào các ô textbox chẳng hạn ?

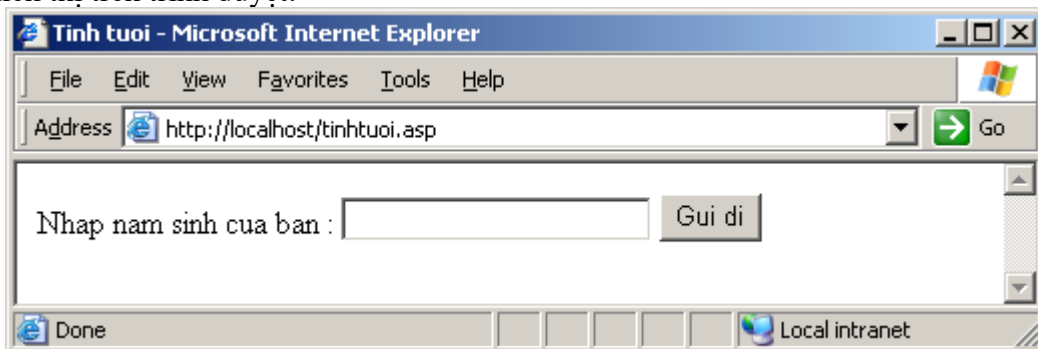
TinhTuoi.asp - Version 2

```
<%@language = vbscript %>
<HTML>
<head>
<meta http-equiv="Content-Type" content="text/HTML; charset=utf-8">
<title>Tinh tuoi</title>
</head>
<BODY>

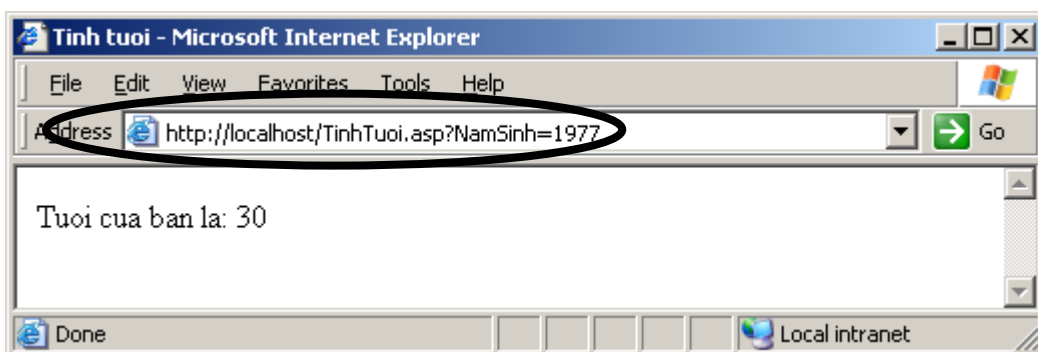
<%
IF Request.QueryString("NamSinh") = "" THEN
%>
    <FORM name=Form1 action="TinhTuoi.asp" method=GET>
        Nhập nam sinh của bạn : <Input type=text name="NamSinh" >
        <Input type=SUBMIT value="Gui đi">
    </FORM>
<%
ELSE
    Response.write ("Tuổi của bạn : " & (year(now)-Request.QueryString("NamSinh")))
END IF
%>

</BODY>
</HTML>
```

Kết quả hiển thị trên trình duyệt:



Khi lần đầu tiên mở trang TinhTuoi.asp



Sau khi nhập 1977 vào ô textbox (hãy để ý trên thanh địa chỉ)

Ví dụ 2: Giả lập một ứng dụng để hiển thị các sản phẩm và cho phép người dùng xem chi tiết mặt hàng khi người dùng click chọn hyperlink tương ứng.

Trong ứng dụng giả lập này, ta sẽ hiển thị 3 đầu mục sản phẩm có dạng :

HDD Seagate 40 GB	Xem chi tiết
HDD Seagate 80 GB	Xem chi tiết
HDD Seagate 160 GB	Xem chi tiết

Khi người dùng click vào Hyperlink "[Xem chi tiết](#)" ở bên cạnh mỗi sản phẩm thì ta hiển thị thông tin chi tiết về sản phẩm đó.

→ Khi người dùng chọn một mục thì cần phải có thông tin gửi trả lại trang xử lý để biết họ đã chọn sản phẩm nào để hiển thị thông tin phù hợp → ẩn chứa trong mỗi Hyperlink phải có những thông tin cần thiết, ví dụ một mã sản phẩm (MaSanPham) chẳng hạn → Dùng hình thức gửi GET là phù hợp !

Trang **ShowProduct.asp** - Hiển thị sản phẩm và chi tiết sản phẩm: 2 in 1

```
<HTML>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Hien thi san pham</title>
</head>
<BODY>

<%
    IF Request.QueryString("MaSanPham") = "" THEN
%>
        <table border=0 cols=2 width="100%">
            <TR>
                <TD>HDD Seagate 20 GB </td>
                <TD> <a href="ShowProduct.asp?MaSanPham=hdd20">Xem chi tiết</a>
            </td>
            </TR>

            <TR>
                <TD>HDD Seagate 40 GB </td>
                <TD> <a href="ShowProduct.asp?MaSanPham=hdd40">Xem chi tiết</a>
            </td>
            </TR>

            <TR>
                <TD>HDD Seagate 80 GB </td>
                <TD> <a href="ShowProduct.asp?MaSanPham=hdd80">Xem chi tiết</a>
            </td>
            </TR>
        </table>
<%
    ELSE
        SELECT CASE Request.QueryString("MaSanPham")
            CASE "hdd20"
                Response.write("Ổ cứng Seagate 20 GB, giá 40 $")
            CASE "hdd40"
                Response.write("Ổ cứng Seagate 40 GB, giá 80 $")
            CASE "hdd80"
                Response.write("Ổ cứng Seagate 80 GB, giá 100 $")
        END SELECT
    END IF
%>

</BODY>
```

```
</HTML>
```

??? **Request.QueryString("MaSanPham") = ""** có nghĩa gì ?

Phiên bản ở trên : Sử dụng một trang vừa hiển thị vừa xử lý dữ liệu gửi đến (kiểu 2 in 1).
Ta có thể sử dụng cơ chế 1 in 1, tức là: Trang hiển thị riêng và trang hiển thị chi tiết riêng:

Trang HienThiSP.htm

```
<HTML>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Hiển thị sản phẩm</title>
</head>
<BODY>

<table border=0 cols=2 width="100%">
  <TR>
    <TD>HDD Seagate 20 GB </td>
    <TD> <a href=" ShowDetail.asp?MaSanPham=hdd20">Xem chi tiết</a> </td>
  </TR>

  <TR>
    <TD>HDD Seagate 40 GB </td>
    <TD> <a href=" ShowDetail.asp?MaSanPham=hdd40">Xem chi tiết</a> </td>
  </TR>

  <TR>
    <TD>HDD Seagate 80 GB </td>
    <TD> <a href=" ShowDetail.asp?MaSanPham=hdd80">Xem chi tiết</a> </td>
  </TR>
</table>

</BODY>
</HTML>
```

Trang ShowDetail.asp

```
<HTML>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Hiển thị chi tiết sản phẩm </title>
</head>

<BODY>
<%
  IF Request.QueryString("MaSanPham") <> "" THEN
    SELECT CASE Request.QueryString("MaSanPham")
      CASE "hdd20"
        Response.write("Ổ cứng Seagate 20 GB, giá 40 $")
      CASE "hdd40"
        Response.write("Ổ cứng Seagate 40 GB, giá 80 $")
      CASE "hdd80"
        Response.write("Ổ cứng Seagate 80 GB, giá 100 $")
    END SELECT
  ELSE
```

```

        Response.write("Bạn vào trang ShowProduct.asp tr- ớc đã ! <BR>")
        Response.write ("<a href = 'HienThiSP.htm'> &lt; &lt; Quay trở lại </a>")
    END IF
%>

</BODY>
</HTML>

```

➔ Nhận xét : Gửi dữ liệu theo kiểu POST khi nào ? và GET khi nào ?

Một số bài tập: (Sử dụng cả ph- ơng pháp gửi POST và GET)

1. Viết trang web tra cứu từ điển. Cho ng- ời dùng nhập vào một từ tiếng Anh, trang web sẽ hiển thị nghĩa tiếng việt t- ơng ứng.
2. Viết trang web thực hiện tính toán với các phép toán, hàm đơn giản.

ĐỐI TƯỢNG SERVER

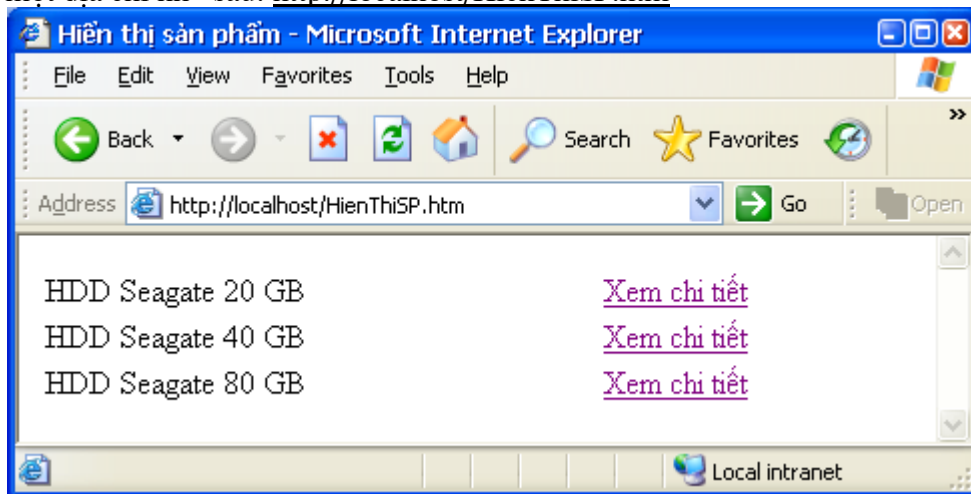
Là một đối t- ượng cung cấp một số hàm th- ờng dùng trong ứng dụng. Có 2 ph- ơng thức đ- ọc cung cấp bởi đối t- ượng server là Mappath và CreateObject.

1. Ph- ơng thức Mappath

Trả về đ- ường dẫn vật lý từ đ- ường dẫn (th- ực) ảo.

+ Cú pháp: Server.Mappath("Đường dẫn ảo/ tương đối")

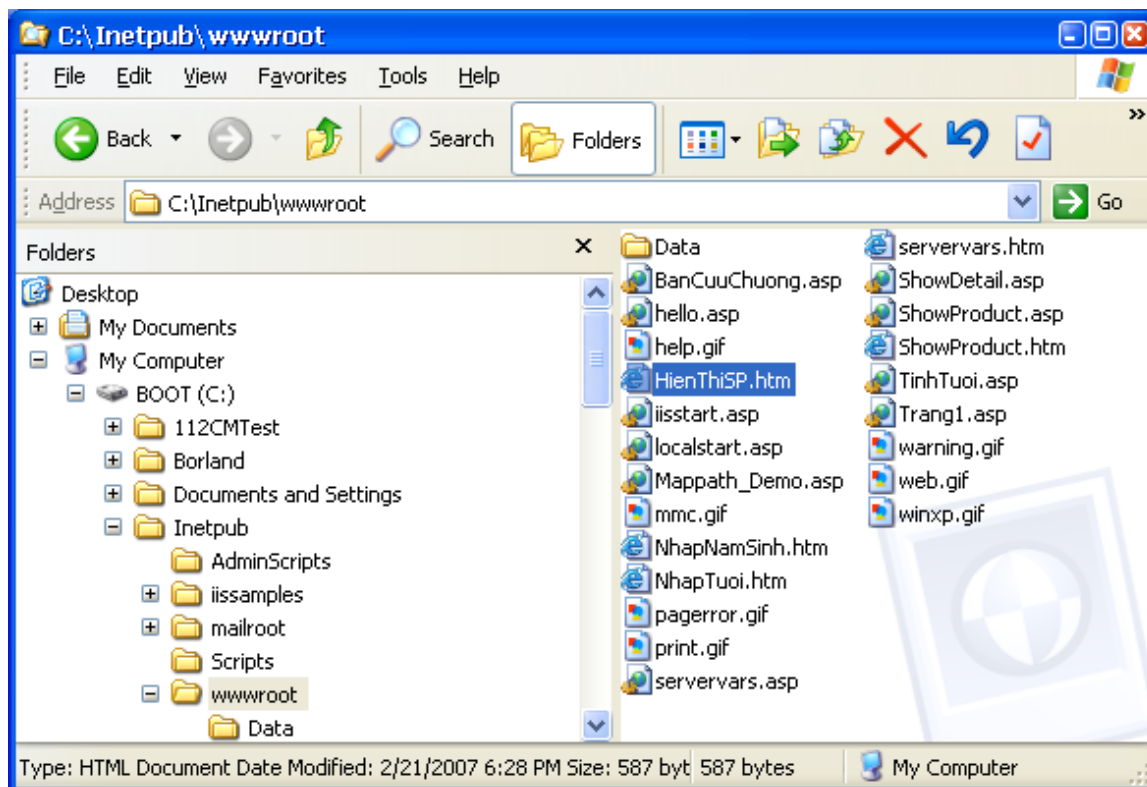
+ Ví dụ: Có một địa chỉ nh- sau: <http://localhost/HienThiSP.htm>



Vị trí ảo của tệp Hienthisip.htm trên webserver

Trang web HienThiSP.htm có đ- ường dẫn ảo là Localhost/HienThiSP.htm.

Đ- ường dẫn vật lý (nằm trên ổ cứng của trang này) là: c:\inetpub\wwwroot\HienThisp.htm



Vị trí vật lý của tệp Hienthisip.htm trên ổ cứng.

Trong trường hợp này, Server.MapPath("HienThiSP.htm") sẽ cho ta kết quả là c:\inetpub\wwwroot\HienThiSP.htm.

→ Như vậy hàm Mappath thực hiện ánh xạ từ đường dẫn ảo → đường dẫn vật lý.

Hàm này đặc biệt hay dùng khi chúng ta tham chiếu đến đường dẫn vật lý của các tệp CSDL khi kết nối.

Ví dụ:

Trang Mappath_Demo.asp

```
<%@ language = vbscript %>
<HTML>

<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
<title>Mappath</title>
</head>

<BODY>
Duong dan vat ly cua mot so trang web : <br>

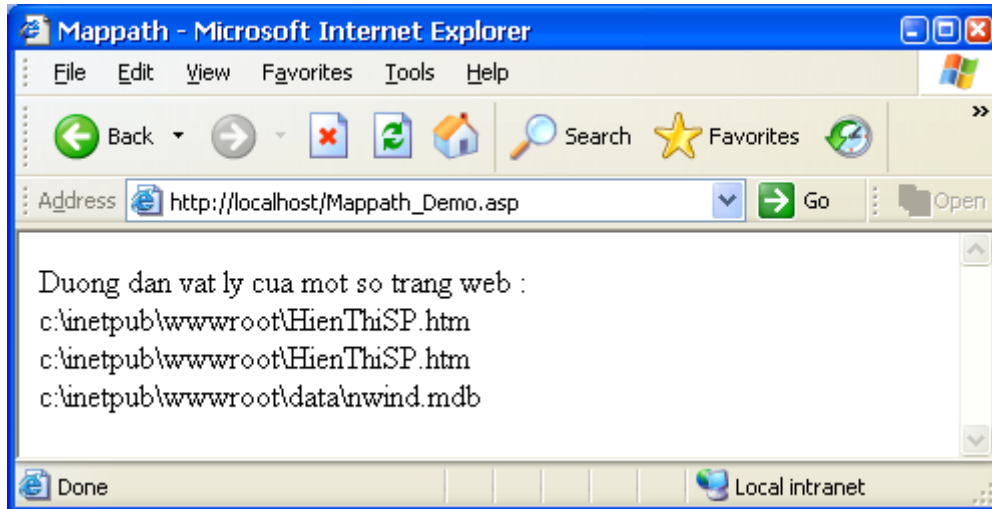
<%
    Response.write (server.mappath("HienThiSP.htm"))
    Response.write ("<BR>")

    Response.write (server.mappath("/HienThiSP.htm"))
    Response.write ("<BR>")

    Response.write (server.mappath("/data/nwind.mdb"))
    Response.write ("<BR>")
%>

</BODY>
```


Kết quả:



Kết quả

L- u ý: Nếu tệp không tồn tại hoặc đ- ờng dẫn ảo đ- a vào sai thì hàm trả về sẽ là một chuỗi gồm đ- ờng dẫn tới th- mục wwwroot cộng với chuỗi đặt trong mappath.

Hỏi: Không cần dùng đến Mappath có đ- ọc không ?

2. Ph- ơng thức CreateObject

Tạo ra một số đối t- ượng. Ví dụ tạo ra các đối t- ượng truy xuất đến CSDL.

+ Cú pháp:

Set Biến = Server.CreateObject("Xâu mô tả đối tượng cần tạo")

+ Ví dụ :

Tạo một đối t- ượng Connection và đối t- ượng Recordset

```
Dim Conn, Rs
Set Conn = Server.CreateObject("ADODB.Connection")
Set Rs = Server.CreateObject("ADODB.Recordset")
```

ĐỐI T- ƯỢNG SESSION – BIẾN SESSION

Đối t- ượng Session : Dùng để quản lý các thông tin liên quan đến một phiên làm việc.

Một phiên (Session) làm việc là gì ? : Mỗi khi mở trình duyệt (lần đầu tiên) vào website ứng dụng, ta đã tạo ra một phiên làm việc. Phiên làm việc kết thúc khi ta ra khỏi website ứng dụng hoặc khi tất cả các cửa sổ trình duyệt bị đóng.

1. Biến Session

Khi vào một website, ng- ười dùng có thể duyệt rất nhiều trang web của website đó. Nếu muốn l- u trữ thông tin về khách thăm này trong cả phiên làm việc thì có thể l- u vào các biến, gọi là biến Session.

2. Đối t- ượng Session

Là đối t- ượng dùng để quản lý (tạo, đọc, ghi) các biến session và một số thông số khác.

+ Cú pháp để tạo biến Session nh- sau:

Session("Tên_Biến") = <Giá trị>

Ví dụ : Tạo một biến tên là MaNguoiDung và gán giá trị là TK34

Session("MaNguoiDung") = "TK34"

+ Cú pháp để đọc giá trị của một biến session nh- sau:

Session("Tên_Biến")

Ví dụ:

Response.write("Mã người dùng là : " & Session("MaNguoiDung"))

Riêng với đối tượng Session, nó còn có các sự kiện. Các sự kiện này tự động được gọi mỗi khi một phiên làm việc được tạo ra. Các sự kiện này có tên là On_Start và On_END. Các sự kiện này được đặt trong file Global.asa.

File Global.asa

```
<script LANGUAGE = "VBScript" RUNAT = Server>

'/// Thủ tục này được gọi khi bắt đầu một phiên
SUB Session_OnStart
    .....
    Session("MaNguoiDung") = "Khách_Lạ"
    .....
END SUB

'/// Thủ tục này được gọi khi kết thúc một phiên
SUB Session_OnEND
    .....
END SUB
</SCRIPT>
```

Ví dụ đọc giá trị biến Session:

File Session_Demo.asp

```
<HTML>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
<title>Đọc giá trị của biến Session có tên là MaNguoiDung</title>
</head>
<BODY>

<%
    Response.write("<BR>Bạn là : " & Session("MaNguoiDung"))
%>

</BODY>
</HTML>
```

Ví dụ: Chỉ hiển thị các trang chủ (Home.asp) nếu người dùng có tên (UserName) là TK34 và mật khẩu là 123456. Nếu không thì hiển thị trang Login.asp để người dùng đăng nhập.

Trang Global.asa

```
<script LANGUAGE = "VBScript" RUNAT = Server>

'/// Thủ tục này được gọi khi bắt đầu một phiên
SUB Session_OnStart
    .....
    Session("MaNguoiDung") = ""
    Session("MatKhau") = ""
    Session("TrangThaiDangNhap") = "ch- a"
    .....
END SUB

'/// Thủ tục này được gọi khi kết thúc phiên làm việc
SUB Session_OnEND
```

'.....
END SUB
</SCRIPT>

```

Trang Login.asp
<HTML>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Dang nhap</title>
</head>

<BODY>

<%
    IF Request.Form("MaNguoiDung")= "TK34" and Request.Form("MatKhau")="123456" THEN
        Session("MaNguoiDung") = Request.Form("MaNguoiDung")
        Session("MatKhau") = Request.Form("MatKhau")
        Session("TrangThaiDangNhap") = "rôi"
        Response.write("Bạn đã đăng nhập thành công ! Bạn có thể vào ")
        Response.write("<a href = 'Home.asp'>Trang chủ </a>")

    ELSE '/// Tạo form đăng nhập
        Session("TrangThaiDangNhap") = "ch- a"
        Response.write("<form action='Login.asp' method = 'Post'>")
        Response.write("Tên đăng nhập: <Input type = text name = MaNguoiDung> <BR>")
        Response.write("Mật khẩu : <Input type = password name = MatKhau> <BR>")
        Response.write("<Input type = Submit value = 'Đăng nhập'>")
        Response.write("</form>")
    END IF
%>

</BODY>
</HTML>

```

```

Trang Home.asp
<HTML>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
<title>Trang chủ</title>
</head>

<BODY>
<%
    IF Session("TrangThaiDangNhap") = "rôi" THEN
        Response.write("<h2>Đây là trang chủ của lớp TK34 !</h2>")
        Response.write("<h1>Chào mừng bạn đã ghé thăm website của chúng tôi !")
    ELSE
        Response.write("Bạn ch- a đăng nhập ! <a href = 'Login.asp'> Đăng nhập.
</a>")
    END IF
%>

</BODY>
</HTML>

```

*L- u ý: Có thể tạo biến Session ở bất kỳ đâu chứ không nhất thiết ở trong file Global.asa
 ??? Dùng biến Session khi nào ?

ĐỐI TƯỢNG APPLICATION

Dùng để quản lý các biến có phạm vi toàn ứng dụng.

1. Khái niệm biến toàn ứng dụng

Biến toàn ứng dụng là biến có tác dụng đối với mọi ng-ời dùng truy cập vào website.

2. Đối tượng Application

Dùng để quản lý (Tạo, đọc, ghi) các biến có phạm vi toàn ứng dụng.

+ Cú pháp tạo biến Application:

Application("Tên_Biến") = <Giá trị>

+ Ví dụ: Tạo biến So_Nguoi_Truy_Cap

Application("So_Nguoi_Truy_Cap") = 0

+ Ví dụ : Đọc và ghi biến Application

Application("So_Nguoi_Truy_Cap") = Application("So_Nguoi_Truy_Cap") + 1

Response.write("Bạn là vị khách thứ: " & Application("So_Nguoi_Truy_Cap"))

L- u ý: Khi biến đã tồn tại thì lệnh trên đ- ọc hiểu là gán giá trị mới, còn nếu ch- a tồn tại thì đ- ọc hiểu là tạo biến.

Đối tượng Application cũng có 2 sự kiện đó là Application_OnStart và Application_OnEND. Sự kiện OnStart chỉ đ- ọc kích hoạt duy nhất một lần khi yêu cầu đầu tiên phát sinh. Sự kiện OnEND đ- ọc kích hoạt khi dịch vụ web dừng (unload).

Đối tượng Application có 2 ph- ơng thức là Lock và Unlock. Khi gọi ph- ơng thức Lock (khóa) thì tất cả các ứng dụng không đ- ọc phép thay đổi các giá trị Application. Để các ứng dụng khác đ- ọc phép thay đổi các biến Application thì gọi ph- ơng thức Unlock.

Mã lệnh viết cho 2 sự kiện này cũng đ- ọc đặt trong file Global.asa.

3. Ví dụ tổng hợp

Mỗi khi ng- ời dùng truy cập vào website thì hiển thị cho ng- ời đó biết là vị khách thứ bao nhiêu — chẳng hạn hiển thị trong trang Home.asp.

Trang Global.asa

```
<script LANGUAGE = "VBScript" RUNAT = Server>
```

```
    '/// Thu tục nay duoc gọi duy nhất một lần
```

```
    SUB Application_OnStart
```

```
        Application("SoKhachTruyCap") = 0
```

```
    END SUB
```

```
    '/// Thu tục nay duoc gọi khi IIS stop
```

```
    SUB Application_OnEND
```

```
        Application("SoKhachTruyCap") = 0
```

```
    END SUB
```

```
    '/// Thu tục nay duoc gọi khi bắt đầu một phiên
```

```
    SUB Session_OnStart
```

```
        '.....
```

```
        Application("SoKhachTruyCap") = Application("SoKhachTruyCap") + 1
```

```
        '.....
```

```
    END SUB
```

```
</SCRIPT>
```

Trang Home.asp

```
<HTML>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
<title>Trang chu</title>
</head>
<BODY>

<%
    Response.write("<h2>Day la trang chu cua lop TK34 !</h2>")
    Response.write("<h1>Chao mung ban da ghe tham website cua chung toi !")
    Response.write("<hr>")
    Response.write("Ban la vi khách thu : " &application("SoKhachTruyCap"))
%>

</BODY>
</HTML>
```

CHÈN FILE VÀO TRANG ASP

1. ý nghĩa: Rất nhiều trang có những phần thông tin giống hệt nhau (Ví dụ phần Đầu trang, chân trang hay phân kết nối đến CSDL) → Để nhanh chóng, dễ dàng sửa đổi, bảo trì, ASP cho phép ng- ời dùng chèn một trang vào trang web hiện tại.

2. Cú pháp chèn file:

```
<!-- #include Loại_Đ- ờng_Dẫn = Tên_File -->
+ Loại_Đ- ờng_Dẫn có thể là File hoặc Virtual.
```

Nếu Loại_Đ- ờng_Dẫn = File thì Tên_File là đ- ờng dẫn t- ơng đối tính từ th- mục hiện hành.

Nếu Loại_Đ- ờng_Dẫn = Virtual thì Tên_File sẽ là đ- ờng dẫn ảo đầy đủ.

Ví dụ 1:

```
<!-- #include File = "Home.asp" -->
```

Ví dụ 2 :

```
<!-- #include Virtual = "/Data/KetNoi.asp" -->
```

3. Ví dụ tổng hợp

Tạo một trang web chứa footer cung cấp thông tin liên hệ. Footer này xuất hiện trong tất cả các trang, trong đó có trang Home.asp.

Các b- ớc tiến hành: Tạo một trang chứa nội dung footer, đặt tên là Footer.asp
Sau đó Include vào các trang — ví dụ trang Home.asp

Trang Footer.asp

```
<hr>
<h4 style="text-align:center">
&copy by TK34 - UTEHY 2006. Telephone: 0321-123456/ Fax: 0321-713015; E-Mail:
TK34@googlegroups.com
</h4>
```

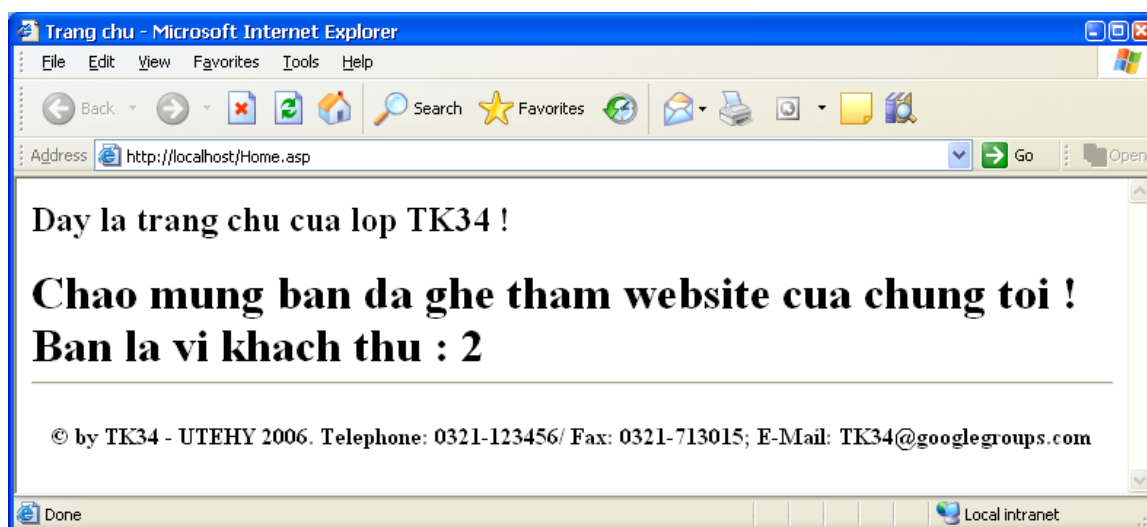
L- u ý:

- Thông thì các trang được Include như thế này có đuôi mở rộng là *.inc
- Nội dung các trang Include có thể chứa cả thẻ HTML và cả mã lệnh asp

Trang Home.asp (Có Include trang Footer.asp)

```
<HTML>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
<title>Trang chu</title>
</head>
<BODY>
<%
    Response.write("<h2>Day la trang chu cua lop TK34 !</h2>")
    Response.write("<h1>Chao mung ban da ghe tham website cua chung toi !")
    Response.write("<BR>")
    Response.write("Ban la vi khách thu : " &application("SoKhachTruyCap"))
%>

<!-- #include file = "Footer.asp" -->
</BODY>
</HTML>
```



Kết quả khi hiển thị trang home.asp

KẾT NỐI VÀ THAO TÁC VỚI CSDL TRONG ASP

1. Quy tắc chung

- Tạo đối tượng Connection và Recordset sử dụng ph- ơng thức Server.CreateObject
- Kết nối đến CSDL
- Thực thi các câu lệnh SQL thông qua ph- ơng thức Execute của đ/t- ơng Connection
- Đọc dữ liệu từ Recordset sử dụng vòng lặp
- Kết hợp với đối t- ơng **Response** để gửi trả kết quả về cho trình duyệt.

2. Tạo đối t- ơng Connection và kết nối đến CSDL

Qui - ớc: Tập cơ sở dữ liệu dùng trong suốt các ví dụ là tập nwind.mdb và đ- ọc đặt tại th- mục gốc của ứng dụng.

Trang KetNoi.asp

```
<HTML>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
```



```

<title>Ket noi </title>
</head>

<BODY>
<%
    Dim Conn
    Set Conn = Server.CreateObject("ADODB.Connection")
    Conn.ConnectionString = "Provider=Microsoft.Jet.oledb.4.0; Data Source = " &
                          Server.MapPath("nwind.mdb")

    Conn.CursorLocation = 3

    on error resume next
    Conn.Open

    IF Conn.State = 1 THEN
        Response.write("Da ket noi thanh cong !")
    ELSE
        Response.write("Khong the ket noi den CSDL !")
    END IF

    cn.Close
%>
</BODY>
</HTML>

```

Đối tượng Connection này giống nh- đối tượng Connection trong Visual Basic.
 Để thực hiện các câu lệnh SQL, sử dụng cú pháp:
 Cn.Execute (<Câu lệnh SQL>); với Cn là một đối tượng Connection

3. Tạo đối tượng Recordset

```

Dim Rs
Set Rs = Server.createObject("ADODB.Recordset")

```

Đối tượng Recordset này giống nh- đối tượng Recordset trong Visual Basic.
 Việc phối hợp giữa đối tượng Connection và Recordset cũng giống nh- trong VB.

4. Một số ví dụ

4.1 Hiện thị toàn bộ tên nhà cung cấp trong bảng Suppliers

```

<HTML>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
<title>Danh sach nha cung cap </title>
</head>

<BODY>
<%
    Dim Conn, Rs
    Set Conn = Server.CreateObject("ADODB.Connection")
    Conn.ConnectionString = "Provider=Microsoft.Jet.oledb.4.0; Data Source = " &
                          Server.MapPath("nwind.mdb")

    Conn.CursorLocation = 3

    on error resume next

```

```

Conn.Open

Set Rs = Server.CreateObject("ADODB.Recordset")

Set Rs = Conn.Execute("SELECT CompanyName from Suppliers")
do while not Rs.eof
    Response.write(Rs("CompanyName"))
    Response.write ("<BR>")
    Rs.MoveNext
loop

Conn.Close

```

```
%>
```

```
</BODY>
```

```
</HTML>
```

Version 2:

Kết nối và đọc dữ liệu sử dụng ph- ơng thức Open của đối t- ợng Recordset.

Đọc dữ liệu thông qua ph- ơng thức Open

```

<HTML>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
<title>Danh sách nhà cung cấp </title>
</head>

<BODY>
<%
Dim Conn, Rs
Set Conn = Server.CreateObject("ADODB.Connection")
Conn.ConnectionString = "Provider=Microsoft.Jet.oledb.4.0; Data Source = " &
Server.MapPath("nwind.mdb")

Conn.CursorLocation = 3

on error resume next
Conn.Open

Set Rs = Server.CreateObject("ADODB.Recordset")

Rs.Open "SELECT CompanyName from Suppliers", Conn, 1,3

Do while not Rs.eof
    Response.write(Rs("CompanyName"))
    Response.write ("<BR>")
    Rs.MoveNext
Loop

Conn.Close
%>
</BODY>
</HTML>

```

Dùng cách này, về sau ta có thể thêm và cập nhật CSDL thông qua Rs.

4.2.1 Hiển thị danh sách các sản phẩm trong một bảng

Thông tin hiển thị gồm: Tên sản phẩm, giá, ...

4.2.2 Thêm bản ghi vào trong CSDL

Xây dựng trang asp cho phép ng-ời dùng thêm các nhà cung cấp vào

4.3 Hiển thị danh sách các nhà cung cấp trong một ComboBox

4.4 Hiển thị Tên các nhà cung cấp (Companyname), và bên cạnh tên nhà cung cấp có thêm cột “Xem các sản phẩm” do nhà cung cấp đó sản xuất ra. Khi người dùng click vào Hyperlink này thì hiển thị tên các sản phẩm do nhà cung cấp đó đã sản xuất ra.

4.5 Tạo một form nhập để cho phép ng-ời dùng nhập thêm nhà cung cấp.

4.6 Tạo một form để tìm kiếm thông tin chi tiết nhà cung cấp. Thông tin nhập vào là mã nhà cung cấp (SupplierID)

4.7 Tạo một trang cập nhật thông tin về nhà cung cấp (bảng Suppliers)

4.8 Tạo một trang cho phép xóa bản ghi trong bảng Suppliers

4.9 Tạo một trang hiển thị thêm 3 cột “Add”, “Update”, “Delete” để cho người dùng có thể thêm, sửa đổi và xóa bản ghi trong bảng Suppliers.

4.10 Tạo một bảng Users, chứa tài khoản ng-ời dùng. Sau đó xây dựng trang Login.asp thực hiện việc đăng nhập.

4.10 Tạo một trang hiển thị danh mục các loại sản phẩm, mỗi khi ng-ời dùng click vào một chủng loại thì hiển thị các sản phẩm thuộc chủng loại đó.

4.11 Hiển thị danh sách sản phẩm theo từng trang — 10 bản ghi / 1 trang.

PHỤ LỤC 1

DANH SÁCH CÁC THẺ TRONG JAVASCRIPT

- Các thuộc tính áp dụng cho Font chữ

Thuộc tính	Các giá trị hợp lệ	Ví dụ
font-family	[Tên font hoặc kiểu]	font-family: Verdana, Arial;
font-style	normal hoặc italic	font-style:italic;
font-variant	normal hoặc small-caps	font-variant:small-caps;
font-weight	normal hoặc bold	font-weight:bold;
font-size	[xx-large x-large large medium small x-small xx-small] [larger smaller] <i>phần trăm</i> hoặc <i>length</i>	font-size:12pt;
Font	[font-style font-variant font-weight] ? font-size [/ line-height] ? font-family	font: bold 12pt Arial;

- Các thuộc tính màu và nền (Color and background properties)

Thuộc tính	Các giá trị hợp lệ	Ví dụ
Color	<i>Màu</i>	color: red
background-color	<i>Màu</i> hoặc transparent	background-color: yellow
background-image	Địa chỉ (url) hoặc Không đặt gì	background-image: url(house.jpg)
background-repeat	repeat repeat-x repeat-y no-repeat	background-repeat: no-repeat
background-attachment	scroll hoặc fixed	background-attachment: fixed
background-position	[position length] { 1,2 } [top center bottom] [left center right]	background-position: top center
background	transparent color url repeat scroll position	background: silver url(house.jpg) repeat-y

* Lưu ý: Các giá trị *in nghiêng* khi sử dụng bạn phải thay bằng giá trị cụ thể, ví dụ viết *phần trăm* hay *length* thì bạn phải thay các giá trị dạng phần trăm, ví dụ 50% đối với thuộc tính phần trăm (percentage) và 10, 20 cho thuộc tính *length*.

- Các thuộc tính áp dụng cho Text

Thuộc tính	Giá trị hợp lệ	Ví dụ
Letter-spacing	normal length	letter-spacing:5pt
text-decoration	none underline overline line-through	text-decoration:underline
vertical-align	Sub super	vertical-align: Sub
text-transform	capitalize uppercase lowercase none	text-transform:lowercase
text-align	left right center justify	text-align:center
text-indent	length percentage	text-indent:25px
line-height	normal number length percentage	line-height:15pt

- Các thuộc tính áp dụng cho các ô trong một bảng

Thuộc tính	Giá trị hợp lệ	Ví dụ
margin-top	length percentage auto	margin-top:5px
margin-right	length percentage auto	margin-right:5px
margin-bottom	length percentage auto	margin-bottom:1em
margin-left	length percentage auto	margin-left:5pt
Margin	length percentage auto {1,4}	margin: 10px 5px 10px 5px
padding-top	length percentage	padding-top:10%
padding-right	length percentage	padding-right:15px
padding-bottom	length percentage	padding-bottom:1.2em
padding-left	length percentage	padding-left:10pt; }
Padding	length percentage {1,4}	padding: 10px 10px 10px 15px
border-top-width	thin medium thick length	border-top-width:thin
border-right-width	thin medium thick length	border-right-width:medium
border-bottom-width	thin medium thick length	border-bottom-width:thick
border-left-width	thin medium thick length	border-left-width:15px
border-width	thin medium thick length {1,4}	border-width: 3px 5px 3px 5px
border-top-color	color	border-top-color:navajowhite
border-right-color	Color	border-right-color:whitesmoke
border-bottom-color	Color	border-bottom-color:black
border-left-color	Color	border-left-color:#C0C0C0
border-color	color {1,4}	border-color: green red white blue; }
border-top-style	none solid double groove ridge inset outset	border-top-style:solid
border-right-style	none solid double groove ridge	border-right-style:double

	inset outset	
border-bottom-style	none solid double groove ridge inset outset	border-bottom-style:groove
border-left-style	none solid double groove ridge inset outset	border-left-style:none
border-style	none solid double groove ridge inset outset	border-style:ridge; }
border-top	border-width border-style border-color	border-top: medium outset red
border-right	border-width border-style border-color	border-right: thick inset maroon
border-bottom	border-width border-style border-color	border-bottom: 10px ridge gray
border-left	border-width border-style border-color	border-left: 1px groove red
Border	border-width border-style border-color	border: thin solid blue
Float	none left right	float:none
clear	none left right both	clear:left

- Các thuộc tính phân loại - classification Properties

Thuộc tính	Giá trị hợp lệ	Ví dụ
Display	none block inline list-item	display:none
list-style-type	disk circle square decimal lower-roman upper-roman lower-alpha upper-alpha none	list-style-type:upper-alpha
list-style-image	url none	list-style-image:url(icFile.gif)
list-style-position	inside outside	list-style-position:inside
list-style	keyword position url	list-style: square outside url(icFolder.gif)

- Các thuộc tính định vị trí cho các phần tử

Thuộc tính	Giá trị hợp lệ	Ví dụ	Có thể áp dụng cho
Clip	Toạ độ của một hình chữ nhật auto	clip:rect(0px 200px 200px 0px)	tất cả các phần tử (all elements)
Height	length auto	height:200px	DIV, SPAN và các ptử bị thay thế
Left	length percentage auto	left:0px	Các phần tử được định vị tuyệt đối và tương đối
Overflow	visible hidden scroll auto	overflow:scroll	tất cả các phần tử

position	absolute relative static	position:absolute	tất cả các phần tử
Top	length percentage auto	top:0px	Các phần tử được định vị tuyệt đối và tương đối
Visibility	visible hidden inherit	visibility:visible	tất cả các phần tử
Width	length percentage auto	width:80%	DIV, SPAN and replaced elements
z-index	auto integer	z-index:-1	Các phần tử được định vị tuyệt đối và tương đối

- Thuộc tính liên quan đến in ấn - Printing Properties

Thuộc tính	Giá trị hợp lệ	Ví dụ
page-break-before	auto always left right	page-break-before:always
page-break-after	auto always left right	page-break-before:auto

-Pseudo Classes

Thuộc tính	Giá trị hợp lệ	Ví dụ
Cursor	auto crosshair default hand move e-resize ne-resize nw-resize n-resize se-resize sw-resize s-resize w-resize text wait help	{ cursor:hand; }
Active, hover, link, visited	n/a	a:hover { color:red; }
first-letter, first-line	any font manipulating declaration	p:first-letter{ float:left;color:blue }

Một số ví dụ áp dụng:

Ví dụ 1: Tạo một dòng văn bản “Welcome to CSS” có font chữ là Arial, in nghiêng và kích thước font chữ là 16 point.

```
<HTML>
<HEAD> </HEAD>
<BODY>
<P style=“font-family:arial; font-style:italic; font-size:16pt”>Welcome to </P>
</BODY>
</HTML>
<Xem kết quả>
```

Ví dụ 2: Tạo một textbox với màu nền là màu tím (magenta).

```
<HTML>
<HEAD> </HEAD>
<BODY>
<Input type = text style = “background-color:magenta”>
</BODY>
</HTML>
<Xem kết quả>
```

Ví dụ 3: Đặt 3 file ảnh nền tương ứng vào tài liệu, vào textbox và vào textarea
<HTML>

```

<HEAD> </HEAD>
<BODY style="background-image:url('anh1.jpg');background-repeat:no-repeat">
<Input style="background-image:url('anh2.jpg');background-repeat:repeat-x; width:100%">
<P>
<textarea style="background-image:url('anh3.jpg');background-repeat:repeat-y" cols=80 rows=25>
</textarea>
</BODY>
</HTML>
<Xem kết quả>

```

***Ghi chú:**

-Khi ta đặt là repeat-x thì ta có một dãy ảnh được xếp liên tiếp theo chiều ngang

-Khi ta đặt là repeat-y thì ta có một dãy ảnh được xếp liên tiếp theo chiều dọc

Ví dụ 4: Tạo một dòng văn bản, chứa xâu: "Trang chu của HY-Aptech", trong đó từ "Trang chu" có màu xanh (blue) và khi người dùng di chuyển chuột đến chữ "Trang chu" thì chuột chuyển thành hình bàn tay, và khi chuột click lên chữ này thì trang <http://www.aptech.ute> sẽ được mở.

```

<HTML>
<HEAD> </HEAD>
<BODY>
<TITLE>Tạo liên kết và xử lý sự kiện</TITLE>
<font style="cursor:hand;color:blue" onclick="window.open('http://www.aptech.ute');">Trang chu
</font> của HY-Aptech
</BODY>
</HTML>

```

====> [Xem kết quả](#)

Ghi chú: Khi muốn áp dụng các kiểu cho một số phần tử các bạn chỉ cần viết : style="Tên_Thuộc_tính : Giá_Trị;" trong định nghĩa thẻ. Trong đó cặp "Tên_Thuộc_tính : Giá_Trị;" có thể viết như cột ví dụ đã chỉ ra ở các bảng trên.

PHỤ LỤC 2

DANH SÁCH CÁC HÀM TRONG VBSCRIPT

Language Element	Description
<u>Abs Function</u>	Returns the absolute value of a number.
<u>Array Function</u>	Returns a Variant containing an array.
<u>Asc Function</u>	Returns the ANSI character code corresponding to the first letter in a string.
<u>Atn Function</u>	Returns the arctangent of a number.
<u>CBool Function</u>	Returns an expression that has been converted to a Variant of subtype Boolean .
<u>CByte Function</u>	Returns an expression that has been converted to a Variant of subtype Byte .
<u>CCur Function</u>	Returns an expression that has been converted to a Variant of subtype Currency .
<u>CDate Function</u>	Returns an expression that has been converted to a Variant of subtype Date .
<u>CDbl Function</u>	Returns an expression that has been converted to a Variant of subtype Double .
<u>Chr Function</u>	Returns the character associated with the specIFied ANSI character code.
<u>CInt Function</u>	Returns an expression that has been converted to a Variant of subtype Integer .
<u>CLng Function</u>	Returns an expression that has been converted to a Variant of subtype Long .
<u>Cos Function</u>	Returns the cosine of an angle.
<u>CreateObject Function</u>	Creates and returns a reference to an Automation object.
<u>CSng Function</u>	Returns an expression that has been converted to a Variant of subtype Single .
<u>CStr Function</u>	Returns an expression that has been converted to a Variant of subtype String .
<u>Date Function</u>	Returns the current system date.
<u>DateAdd Function</u>	Returns a date to which a specIFied time interval has been added.
<u>DateDiff Function</u>	Returns the number of intervals between two dates.
<u>DatePart Function</u>	Returns the specIFied part of a given date.
<u>DateSerial Function</u>	Returns a Variant of subtype Date for a specIFied year, month, and day.

<u>DateValue Function</u>	Returns a Variant of subtype Date .
<u>Day Function</u>	Returns a whole number between 1 and 31, inclusive, representing the day of the month.
<u>Eval Function</u>	Evaluates an expression and returns the result.
<u>Exp Function</u>	Returns e (the base of natural logarithms) raised to a power.
<u>Filter Function</u>	Returns a zero-based array containing subset of a string array based on a spec IF ied filter criteria.
<u>Fix Function</u>	Returns the integer portion of a number.
<u>FormatCurrency Function</u>	Returns an expression formatted as a currency value using the currency symbol defined in the system control panel.
<u>FormatDateTime Function</u>	Returns an expression formatted as a date or time.
<u>FormatNumber Function</u>	Returns an expression formatted as a number.
<u>FormatPercent Function</u>	Returns an expression formatted as a percentage (multiplied by 100) with a trailing % character.
<u>GetLocale Function</u>	Returns the current locale ID value.
<u>GetObject Function</u>	Returns a reference to an Automation object from a file.
<u>GetRef Function</u>	Returns a reference to a procedure that can be bound to an event.
<u>Hex Function</u>	Returns a string representing the hexadecimal value of a number.
<u>Hour Function</u>	Returns a whole number between 0 and 23, inclusive, representing the hour of the day.
<u>InputBox Function</u>	Displays a prompt in a dialog box, waits for the user to input text or click a button, and returns the contents of the text box.
<u>InStr Function</u>	Returns the position of the first occurrence of one string within another.
<u>InStrRev Function</u>	Returns the position of an occurrence of one string within another, from the end of string.
<u>Int Function</u>	Returns the integer portion of a number.
<u>IsArray Function</u>	Returns a Boolean value indicating whether a variable is an array.
<u>IsDate Function</u>	Returns a Boolean value indicating whether an expression can be converted to a date.
<u>IsEmpty Function</u>	Returns a Boolean value indicating whether a variable has been initialized.
<u>IsNull Function</u>	Returns a Boolean value that indicates whether an expression contains no valid data (Null).
<u>IsNumeric Function</u>	Returns a Boolean value indicating whether an expression can be evaluated as a number.
<u>IsObject Function</u>	Returns a Boolean value indicating whether an expression references a valid Automation object.
<u>Join Function</u>	Returns a string created by joining a number of substrings contained in an array.
<u>LBound Function</u>	Returns the smallest available subscript for the indicated dimension of an array.
<u>LCase Function</u>	Returns a string that has been converted to lowercase.
<u>Left Function</u>	Returns a spec IF ied number of characters from the left side of a string.
<u>Len Function</u>	Returns the number of characters in a string or the number of bytes required to store a variable.
<u>LoadPicture Function</u>	Returns a picture object. Available only on 32-bit platforms.
<u>Log Function</u>	Returns the natural logarithm of a number.
<u>LTrim Function</u>	Returns a copy of a string without leading spaces.
<u>Mid Function</u>	Returns a spec IF ied number of characters from a string.
<u>Minute Function</u>	Returns a whole number between 0 and 59, inclusive, representing the minute of the hour.
<u>Month Function</u>	Returns a whole number between 1 and 12, inclusive, representing the month of the year.
<u>MonthName Function</u>	Returns a string indicating the spec IF ied month.
<u>MsgBox Function</u>	Displays a message in a dialog box, waits for the user to click a button, and returns a value indicating which button the user clicked.
<u>Now Function</u>	Returns the current date and time according to the setting of your computer's system date and time.
<u>Oct Function</u>	Returns a string representing the octal value of a number.
<u>Replace Function</u>	Returns a string in which a spec IF ied substring has been replaced with another substring a spec IF ied number of times.

<u>RGB Function</u>	Returns a whole number representing an RGB color value.
<u>Right Function</u>	Returns a spec IF ied number of characters from the right side of a string.
<u>Rnd Function</u>	Returns a random number.
<u>Round Function</u>	Returns a number rounded to a spec IF ied number of decimal places.
<u>RTrim Function</u>	Returns a copy of a string without trailing spaces.
<u>ScriptEngine Function</u>	Returns a string representing the scripting language in use.
<u>ScriptEngineBuildVersion Function</u>	Returns the build version number of the scripting engine in use.
<u>ScriptEngineMajorVersion Function</u>	Returns the major version number of the scripting engine in use.
<u>ScriptEngineMinorVersion Function</u>	Returns the minor version number of the scripting engine in use.
<u>Second Function</u>	Returns a whole number between 0 and 59, inclusive, representing the second of the minute.
<u>SetLocale Function</u>	Sets the global locale and returns the previous locale.
<u>Sgn Function</u>	Returns an integer indicating the sign of a number.
<u>Sin Function</u>	Returns the sine of an angle.
<u>Space Function</u>	Returns a string consisting of the spec IF ied number of spaces.
<u>Split Function</u>	Returns a zero-based, one-dimensional array containing a spec IF ied number of substrings.
<u>Sqr Function</u>	Returns the square root of a number.
<u>StrComp Function</u>	Returns a value indicating the result of a string comparison.
<u>String Function</u>	Returns a repeating character string of the length spec IF ied.
<u>StrReverse Function</u>	Returns a string in which the character order of a spec IF ied string is reversed.
<u>Tan Function</u>	Returns the tangent of an angle.
<u>Time Function</u>	Returns a Variant of subtype Date indicating the current system time.
<u>Timer Function</u>	Returns the number of seconds that have elapsed since 12:00 AM (midnight).
<u>TimeSerial Function</u>	Returns a Variant of subtype Date containing the time for a spec IF ic hour, minute, and second.
<u>TimeValue Function</u>	Returns a Variant of subtype Date containing the time.
<u>Trim Function</u>	Returns a copy of a string without leading or trailing spaces.
<u>TypeName Function</u>	Returns a string that provides Variant subtype information about a variable.
<u>UBound Function</u>	Returns the largest available subscript for the indicated dimension of an array.
<u>UCase Function</u>	Returns a string that has been converted to uppercase.
<u>VarType Function</u>	Returns a value indicating the subtype of a variable.
<u>Weekday Function</u>	Returns a whole number representing the day of the week.
<u>WeekdayName Function</u>	Returns a string indicating the spec IF ied day of the week.
<u>Year Function</u>	Returns a whole number representing the year.

PHẦN III: THỰC HÀNH

BÀI THỰC HÀNH SỐ 1

MỤC TIÊU: Kết thúc bài thực hành sinh viên có thể

- Xây dựng trang Web theo cấu trúc Frame
- Hiển thị trang web trong một frame chỉ định
- Tạo các hiệu ứng cho liên kết

NỘI DUNG:

Yêu cầu: Tạo một trang Web có 3 frame với tên lần lượt là : **frameTren**, **frameTrai**, **framePhai**. frameTren sẽ nạp trang Banner.htm, frameTrai sẽ nạp trang Menu.htm, framePhai dùng để hiển thị trang Web mà liên kết được đặt trong frameTrai, ban đầu frame này sẽ nạp trang GioiThieu.htm.

Tạo trang Banner.htm

1. Mở chương trình soạn thảo, ví dụ FrontPage
2. Gõ nội dung như sau

```
<HTML>
<HEAD>
<TITLE>Trang Banner.htm</TITLE>
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=utf-8">
</HEAD>

<BODY>
    <H1 STYLE="width:100%; filter:glow(); text-align:center;">Siêu thị máy tính </H1>
</BODY>
</HTML>
```

Tạo trang GioiThieu.htm

```
<HTML>
<HEAD>
<TITLE>Trang Giới thiệu</TITLE>
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=utf-8">
</HEAD>

<BODY>
    <H1 Style="text-align:center;">Giới thiệu về công ty tại đây</H1>
    <H1 Style="text-align:center;">Tham khảo các trang Web trên mạng </H1>
</BODY>
</HTML>
```

Trang Menu.htm (Minh hoạ tạo 2 liên kết đến trang GioiThieu.htm và SanPham.asp)

```
<HTML>
<HEAD>
<TITLE>Trang Liên kết</TITLE>
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=utf-8">
</HEAD>
<BODY>
    <A Href = "GioiThieu.htm" Target = "Main">Giới thiệu </A>
    <A Href = "SanPham.asp" Target = "Main">Sản phẩm </A>
</BODY>
</HTML>
```

Trang Menu.htm (Version 2 - tạo hiệu ứng)

```
<HTML>
<HEAD>
<TITLE>Trang liên kết - tạo hiệu ứng</TITLE>
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=utf-8">
</HEAD>
<Style Type = "Text/CSS">
    A { Text-decoration: None} // Bỏ gạch chân dưới mỗi liên kết
    .MenuThuong {color : Blue; Background-color: White} // Định nghĩa lớp
    .MenuKichHoat {color : White; Background-color : Red} // Định nghĩa lớp
</Style>

<BODY>
    <A    onMouseOver="this.className='MenuKichHoat'"
        onMouseOut ="this.className='MenuThuong' "
        href = "GioiThieu.htm" Target = "Main">Giới thiệu
    </A>
    <BR>
    <A    onMouseOver="this.className='MenuKichHoat' "
        onMouseOut ="this.className='MenuThuong' "
        href = "SanPham.asp" Target = "Main">Sản phẩm
    </A>
</BODY>
</HTML>
```

Trang Index.htm (Chú ý : Thêm các thẻ <BODY>)

```
<Frameset rows = "20%,*">
    <Frame name = Banner Src = "Banner.htm">
    <FrameSet Cols= "20%,*">
        <Frame name = Menu Src = "Menu.htm">
        <Frame name = Main Src = "GioiThieu.htm">
    </FrameSet>
</FrameSet>
```

BÀI THỰC HÀNH SỐ 2

MỤC TIÊU: Sau bài thực hành này Sinh viên có thể

- Viết lệnh đọc và hiển thị dữ liệu trong một bảng trong trang Web
- Bổ sung dữ liệu vào trong bảng CSDL
- Cập nhật dữ liệu vào trong bảng CSDL
- Truyền dữ liệu giữa các trang bằng URL (Dạng **Request.QueryString(...)**)

NỘI DUNG THỰC HÀNH

Tạo CSDL tên là **Book.mdb**, CSDL này có 1 bảng là **tblSach** gồm các trường như sau:

tblSach : Table			
	Field Name	Data Type	Description
	MaSach	Text	Mã sách
	TenSach	Text	Tên sách
	TenTacgia	Text	Tên tác giả
	TenNXB	Text	Tên Nhà xuất bản
	SoTrang	Number	Số trang
	NamXB	Number	Năm xuất bản
	Gia	Number	Giá
	LoaiSach	Text	Loại sách
	DuongDan	Text	Đường dẫn tới file ảnh mô tả cuốn sách

1. Đọc dữ liệu trong bảng CSDL và hiển thị trên trang Web

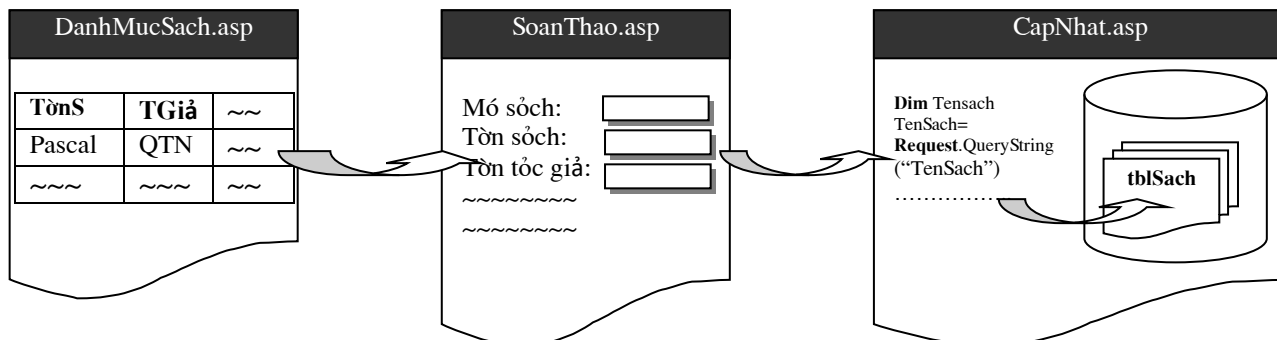
Tạo một trang **HienThiSach.asp** có nội dung như sau:

```
<%@LANGUAGE="VBSCRIPT" %>
<HTML>
<title>Đọc dữ liệu từ file CSDL</title>
<body style="font-family:arial">
<%
    Dim Cn, Rs
    Dim strCn
    '//// Tạo kết nối đến CSDL //////////////////////////////////////
    Set Cn = Server.CreateObject("ADODB.Connection")          '/// Tạo đối tượng Connection
    Cn.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0; Data Source=" & Server.MapPath("Book.mdb")
    Cn.CursorLocation = 3
    Cn.Open                                                     '/// Thực sự mở kết nối đến CSDL Book.mdb
    Set Rs = Cn.Execute("Select * from tblSach")                '/// Mở bảng và lưu trong biến RS

    '//// Đọc dữ liệu và hiển thị ra Bảng////////////////////////////////////
    Response.Write("<Table border = 1 style='width:100%'>")
    Response.Write("<TR><TH>Mô tả <TH>Tên sách <TH> Tên Tác giả <TD>")
    Response.Write("Tên NXB <TD> Năm XB <TD>Số trang <TD> Giá ")
    Do while not Rs.EOF
        Response.Write("<TR>")
        Response.Write("<TD><img width=100px src='" & Rs("DuongDan") &"'>")
        Response.Write("<TD>" & Rs("TenSach"))
        Response.Write("<TD>" & Rs("TenTacgia"))
        Response.Write("<TD>" & Rs("TenNXB"))
        Response.Write("<TD>" & Rs("NamXB"))
        Response.Write("<TD>" & Rs("SoTrang"))
        Response.Write("<TD>" & Rs("Gia"))
        Rs.MoveNext
    Loop
    Response.Write("</Table>")          '/// Đóng thẻ Table
</BODY>
</HTML>
```


2. Cập nhật dữ liệu (Cập nhật thông tin về một cuốn sách nào đó)

Ý tưởng: Để cập nhật dữ liệu cho thuận tiện, chúng ta sẽ xây dựng 3 trang: Một trang gọi là “**DanhMucSach.asp**” để hiển thị các sách hiện có. Mỗi một cuốn sách trong trang này sẽ có một liên kết tên là “**Cập nhật**”, liên kết này chỉ đến trang “**SoanThao.asp**”, Khi người dùng click vào liên kết thì trong liên kết có chứa “**MaSach**” sẽ được gửi đến trang “**SoanThao.asp**”. Trang “**SoanThao.asp**” sẽ hiển thị các thông tin trong các textbox về cuốn sách có mã bằng mã do trang “**DanhMucSach.asp**” gửi tới và cho người dùng thay đổi tại đây. Thông tin soạn xong sẽ được gửi đến trang “**CapNhat.asp**” để thực hiện việc cập nhật. Sơ đồ mô tả như hình dưới đây:



File: **DanhMucSach.asp**

```
<%@LANGUAGE="VBSCRIPT" %>
<title>Danh mục sách trong file CSDL</title>

<body style="font-family:arial">

<!-- #Include file= "KetNoi.inc" -->                                     '/// Nội dung file Ketnoi.inc xin xem ở phần dưới

<h1 align="center">Danh mục sách trong kho </h1>

<%
    Set Rs = Cn.Execute("Select * from tblSach")                        '/// Mở bảng và lưu trong biến RS

    '/// Đọc dữ liệu và hiển thị ra Bảng////////////////////////////////////
    Response.Write("<Table border = 1 style='width:100%'>")
    Response.Write("<TR> <TH>Mô tả <TH>Tên sách <TH> Tên Tác giả <TD>Tên NXB <TD> Sửa đổi")
    do while not Rs.EOF
        Response.Write("<TR>")
        Response.Write("<TD><img width=100px src='\" & Rs(\"DuongDan\") &'>")
        Response.Write("<TD>" & Rs("TenSach"))
        Response.Write("<TD>" & Rs("TenTacgia"))
        Response.Write("<TD>" & Rs("TenNXB"))
        Response.Write("<TD><A href = 'SoanThao.asp?MaSach=' & Rs("MaSach") &'>Sửa")
        Rs.MoveNext
    Loop
    Response.Write("</Table>")                                           '/// Đóng thẻ Table

%>
</BODY>
</HTML>
```

File: “**KetNoi.inc**”

```
<%
    Dim Cn, Rs
    Dim strCn

    Set Cn = Server.CreateObject("ADODB.Connection")                  '/// Tạo đối tượng Connection
    Cn.ConnectionString= "Provider = Microsoft.Jet.OLEDB.4.0; Data Source =" & Server.MapPath("Book.mdb")
    Cn.CursorLocation = 3
    Cn.Open                                                            '/// Thực sự mở kết nối đến CSDL Book.mdb

%>
```

File: SoanThao.asp

```
<%@LANGUAGE="VBSCRIPT" %>
<HTML>
<head>
<title>Hiển thị dữ liệu cho người dùng soạn thảo</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
</head>

<body style="font-family:arial">
<!-- #Include file= "KetNoi.inc" -->                                '// File KetNoi.inc xin xem ở trang trên

<h1 align="center">Thay đổi các trường và nhấn nút "Cập nhật"</h1>

<%
    Dim MaSach
    MaSach = Request.QueryString("MaSach")                        '// Lấy mã sách do trang DanhMucSach gửi tới

    'Chọn sách có Mã sách = Mã sách mà người dùng muốn sửa đổi
    Set Rs = Cn.Execute("Select * from tblSach where Masach = " & MaSach & " ")

    Response.Write("<FORM Action = 'CapNhat.asp' method = Post>")
    Response.Write("Mã sách : <Input type=text value = " & Rs("MaSach") & " " name = MaSach> <BR>")
    Response.Write("Tên sách : <Input type=text value = " & Rs("TenSach") & " " name = TenSach> <BR>")
    Response.Write("Tên tác giả : <Input type=text value = " & Rs("TenTacGia") & " " name = TenTacGia>
    <BR>")
    Response.Write("Tên nhà xuất bản: <Input type=text value = " & Rs("TenNXB") & " " name = TenNXB>
    <BR>")
    Response.Write("<Input type = Submit value = 'Cập nhật'">")
    Response.Write("</FORM>")
%>
</BODY>
</HTML>
```

File: CapNhat.asp

```
<%@LANGUAGE="VBSCRIPT" %>
<HTML>
<head>
<title>Cập nhật dữ liệu từ file CSDL</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
</head>

<BODY style="font-family:arial">
<!-- #Include file= "KetNoi.inc" -->                                '// File KetNoi.inc xin xem ở trang trên

<%
    Dim MaSach, TenSach, TenTacGia, TenNXB

    MaSach = Request.Form("MaSach")                                '// Lấy mã sách mà người dùng muốn
    TenSach = Request.Form("TenSach")
    TenTacGia = Request.Form("TenTacGia")
    TenNXB = Request.Form("TenNXB")

    Cn.execute "Update tblSach Set TenSach = " & TenSach & ", TenTacGia = " & TenTacGia & ", TenNXB = "
    & TenNXB & " Where MaSach = " & MaSach & " "
    Response.Write("Thông tin đã được cập nhật ! <a href=DanhMucSach.asp>Danh mục sách</A>")
%>

</BODY>
</HTML>
```


BÀI THỰC HÀNH SỐ 3

MỤC TIÊU: Sau bài thực hành này Sinh viên có thể

- Khai báo và sử dụng biến Session và biến Application
- Sử dụng biến toàn cục Application để đếm số lượng người truy cập
- Sử dụng biến Session để lưu tình trạng đăng nhập của người dùng
- Tạo trang đăng nhập cho người dùng

NỘI DUNG THỰC HÀNH

I. Tạo biến toàn cục

1. Biến Application

Biến Application là biến có phạm vi ảnh hưởng đến tất cả mọi người dùng. Nó tồn tại từ khi IIS Server chạy. Biến này thường được dùng để lưu các thông tin mang ý nghĩa toàn cục, ví dụ như số lượng người truy cập v.v...

Bài 01: Tạo biến toàn cục có tên là SoNguoiTruyCap, và hiển thị giá trị của nó trong trang có tên là : Test.asp

Bước 1: Tạo trang **Global.asa** có nội dung như sau (Lưu ý: Tên phải là **Global.asa**):

Trang Global.asa

```
<Script language = VBScript Runat = Server>  
    Sub Application_OnStart  
        Application("SoNguoiTruyCap") = 0           '// Tạo biến toàn cục tên là SoNguoiTruyCap  
    End Sub  
</SCRIPT>
```

Bước 2: Tạo trang **Test.asp** để hiển thị và thay đổi giá trị của biến toàn cục, có nội dung như sau :

Trang Test.asp

```
<%@ Language = VBScript %>  
<HTML>  
<BODY>  
<%  
    Response. Write("Giá trị của biến toàn cục là : " & Application("SoNguoiTruyCap"))  
    Application("SoNguoiTruyCap") = Application("SoNguoiTruyCap") + 1  
    Response. Write("<BR> Giá trị của biến bây giờ là : " & Application("SoNguoiTruyCap"))  
%>  
</BODY>  
</HTML>
```

*** Nhận xét ***:

- Vì **SoNguoiTruyCap** là một biến toàn cục, do vậy nó vẫn tồn tại ngay cả khi bạn đã đóng tất cả các cửa sổ trình duyệt. Sự thay đổi này có tác dụng đối với mọi người dùng truy nhập vào Website.
- Để tạo một biến toàn cục, chúng ta viết: `Application("<Tên biến>") = <Giá trị>` như ở trên.
- Thủ tục sự kiện `Application_OnStart` và `Session_OnStart` được gọi mỗi khi một trang ASP được mở, trong các thủ tục này chúng ta có thể viết các câu lệnh ASP, kể cả việc truy cập vào CSDL.

Bài 02: Đếm số lượt người truy cập vào Website

Yêu cầu: Xây dựng một trang Web có tên là `Index.asp`. Trong trang này sẽ hiển thị số lượng người đã truy cập.

Trang `Index.asp`

```
<%@ Language = VBScript %>
<HTML>
<BODY>
<H1 Align = Center> Chào mừng bạn đã đến với Website của lớp chỳnh tụi ! </H1>
<BR><BR><HR>

<%
    Response.Write("<H2> Bạn là vị khách thứ " & Application("SoNguoiTruyCap"))
%>

</BODY>
</HTML>
```

☞ **Lưu ý:** Vì mỗi một khách vào thăm trang Web thì thủ tục sự kiện `Session_OnStart` được gọi một lần, do vậy để đếm số lượng người truy cập thì trong thủ tục này ta chỉ cần tăng biến toàn cục `SoNguoiTruyCap` lên 1 đơn vị là đủ. Tức là trong tệp `Global.asa` bạn cần thêm các câu lệnh sau:

Trang `Global.asa`

```
<Script language = VBScript Runat = Server>
    Sub Application_OnStart
        Application("SoNguoiTruyCap") = 0        '/// Tạo biến toàn cục tên là SoNguoiTruyCap
    End Sub

    Sub Session_OnStart        '/// Thủ tục này được gọi khi có 1 người mới viếng thăm
        Application("SoNguoiTruyCap") = Application("SoNguoiTruyCap") + 1
    End Sub
</SCRIPT>
```

2. Biến Session

Biến Session là biến toàn cục dùng để lưu thông tin hoặc trạng thái của một người dùng cụ thể nào đó trong một phiên làm việc, Ví dụ: Dùng để lưu trạng thái đăng nhập (Thành công hay không thành công) của một người dùng.

Biến Session chỉ tồn tại trong thời gian Session được tạo ra, nó không ảnh hưởng đến các Session khác. Để tạo biến Session, cần đặt câu lệnh trong file Global.asa và trong thủ tục **Sub Session_OnStart** lệnh sau: **Session("<Tên biến>") = <Giá trị>**

Sau đây chúng ta sẽ sử dụng biến Session để lưu trạng thái đăng nhập của người dùng. Và tạo 1 trang đăng nhập.

Bài 02: Xây dựng 2 trang ASP có tên lần lượt là : **Xoa.asp** và trang **Login.asp**.

Trang xoa có nội dung như sau (Thực tế giống như trang Xoa.asp trong Project01): **“Loại bỏ sản phẩm “.** Và trang này chỉ hiển thị nếu người dùng đã đăng nhập với Tên là **“tk33”** và mật khẩu là **“123”**. Nếu người dùng chưa đăng nhập thì mở trang Login.asp để đăng nhập.

❶ Khai báo biến Session(“DaDangNhap”) trong file **Global.asa** như sau:

File GLOBAL.ASA

```
<Script language = VBScript Runat = Server>
    Sub Session_OnStart      '/// Thủ tục này được gọi khi có 1 người mới viếng thăm
        Session("DaDangNhap") = "chua"    '/// Biến Đã đăng nhập khởi tạo = "chua"
    End Sub
</SCRIPT>
```

❷ Xây dựng trang **Login.asp**

```
<HTML>
<TITLE>Đăng nhập hệ thống </TITLE>
<BODY>
    <FORM ACTION="Login.asp" METHOD="POST">
        <TABLE BGCOLOR="#FF9966" ALIGN="CENTER">
            <TR STYLE="color:white" BGCOLOR="#660000">
                <TD COLSPAN="2" ALIGN="CENTER"> Thông tin đăng nhập </TD>
            </TR>
            <TR>
                <TD>Tên đăng nhập :</TD>
                <TD><Input TYPE="TEXT" NAME="UserName">
            </TR>
            <TR>
                <TD>Mật khẩu:</TD>
                <TD><Input TYPE="PASSWORD" name = Password>
            </TR>
            <TR>
                <TD>&nbsp;</TD>
                <TD><Input TYPE="SUBMIT" VALUE="Đăng nhập">
            </TR>
        </TABLE>
    </FORM>
    <%
        IF Request.Form("UserName") = "tk33" And Request.Form("Password") = "123" Then
            Session("DaDangNhap") = "roi"          '/// Ghi nhớ lại là đã đăng nhập thành công rồi.
            Response.write("Bạn đã đăng nhập thành công ! <a href = 'index.htm'>Về trang chủ </a>")
        Else
            Response.write("Tên và mật khẩu sai ! Xin hãy nhập lại.")
        End IF
    %>
</BODY>
</HTML>
```

✎ Thực tế, việc kiểm tra người dùng ở trên, chúng ta sẽ so sánh với chi tiết người dùng trong CSDL.

XOA.SP

```
<HTML> <BODY>
<%
    IF Session("DaDangNhap") = "roi" Then '// Thực hiện các câu lệnh ở đây
        Response.write("Bản ghi đã bị xoá ! ")
        Response.write("Bạn muốn xoá tiếp hay không ?")
    Else
        Response.write("Bạn chưa đăng nhập. <a href='Login.asp'>Login</a>")
    End IF
%>
</BODY> </HTML>
```

Nâng cấp trang Login.asp

```
<HTML>
<TITLE>Đăng nhập hệ thống </TITLE>
<BODY>
    <FORM ACTION="Login.asp" METHOD="POST">
        <TABLE BGCOLOR="#FF9966" ALIGN="CENTER">
            <TR STYLE="color:white" BGCOLOR="#660000">
                <TD COLSPAN="2" ALIGN="CENTER"> Thông tin đăng nhập</TD>
            <TR>
                <TD>Tên đăng nhập :</TD>
                <TD><Input TYPE="TEXT" NAME="UserName">
            <TR>
                <TD>Mật khẩu:</TD>
                <TD><Input TYPE="PASSWORD" name = Password>
            <TR>
                <TD>&nbsp;</TD>
                <TD><Input TYPE="SUBMIT" VALUE="Đăng nhập">
            </TABLE>
        </FORM>

<!-- #Include File = "KetNoi.Inc" -->

<%
    Dim Ten, MatKhau, strSQL
    Ten = Request.Form("UserName")
    MatKhau = Request.Form("Password")
    IF Ten <> "" Then '// Chỉ kiểm tra nếu người dùng đã nhập Username & Password
        strSQL = "Select* from tblUserDetail Where Ten=" & Ten & " AND MatKhau =" & MatKhau & ""
        Set Rs = Cn.Execute(strSQL)
        IF Rs.RecordCount > 0 Then
            Session("DaDangNhap") = "roi"
            Response.write("Bạn đã đăng nhập thành công ! <a href = 'index.htm'>Về trang chủ </a>")
        Else
            Response.write("Tên và mật khẩu sai ! Xin hãy nhập lại.")
        End IF
    End IF
%>

</BODY>
</HTML>
```

trang Login.asp

BÀI THỰC HÀNH SỐ 4

Mục tiêu: Kết thúc bài thực hành, Sinh viên có thể:

- Tích hợp cơ chế bảo mật (Login) vào ứng dụng Web
- Xử lý được các mục dữ liệu truyền đi và nhận về với số lượng lớn
- Mô phỏng được hệ thống bán hàng trực tuyến.

Nội dung:

Xây dựng 3 trang có tên lần lượt là: **DatHang.asp**, **Thongke.asp** và **ThanhToan.asp**. Được mô tả theo chức năng như sau:

Trang **Dathang.asp** có chức năng hiển thị các mặt hàng và cho phép người dùng đặt số lượng tương ứng. Sau khi người dùng click vào nút "Đặt hàng" thì toàn bộ Tên sách, số lượng cũng như giá của mỗi loại sách sẽ được gửi sang trang **ThongKe.asp**. Trang **Thongke.asp** sẽ tiếp nhận các thông tin này đồng thời hiển thị thêm cột thành tiền và tổng số tiền mà khách hàng đã đặt mua để khách hàng xem lại một lần nữa trước khi thực hiện thanh toán. Trang này cũng một nút "Chấp nhận" và Sau khi người click vào nút này thì tổng số tiền mua sẽ được gửi đến trang **ThanhToan.asp** để thực hiện khấu trừ tiền của khách hàng và cộng thêm vào tài khoản của người bán (Giả định là mã số của người bán là 'seller').

Cơ sở dữ liệu sách có 2 bảng như sau:

tblSach	
Tên trường	Mô tả
MaSach	Mã sách
TenSach	Tên sách
TenTacGia	Tên tác giả
TenNXB	Tên NXB
SoTrang	Số trang
Gia	Giá

tblUserDetail	
Tên trường	Mô tả
MaNguoiDung	Mã người dùng
MatKhau	Mật khẩu
SoDuTaiKhoan	Số dư tài khoản

Trang **KetNoi.Inc** có nội dung như sau:

```
<%  
Dim Cn, Rs  
Dim strCn  
  
'///// Tạo kết nối sau đó thực hiện kết nối đến CSDL ///////////////////////////////////  
Set Cn = Server.CreateObject("ADODB.Connection") ' /// Tạo đối tượng Connection  
Cn.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0; Data Source =" &  
Server.MapPath("Book.mdb")  
  
Cn.CursorLocation = 3  
Cn.Open ' /// Thực sự mở kết nối đến CSDL Book.mdb  
%>
```

Trang DatHang.asp

```
<HTML>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Đặt hàng trực tuyến</title>
</head>
<BODY>
<!-- #Include file = "KetNoi.inc" -->
<h1 align =center> ĐẶT MUA SÁCH TRỰC TUYẾN </h1>
<HR>

<Form action = Thongke.asp method = post>
<TABLE Border = 1> <TR>    <TH>Tên sách    <TH>Số lượng    <TH> Đơn giá
<%
    Dim i
    i=1
    Set Rs = Cn.Execute("Select * from tblSach")
    Do while not Rs.EOF
        Response.write("<TR>")
        Response.write("<TD><Input NAME=TENSACH" & i & " Value=" & Rs("TenSach") & ">")
        Response.write("<TD> <Input NAME=SL" & i & " Value = 0>")
        Response.write("<TD> <Input NAME=Gia" & i & " Value = " & RS("Gia") & ">")
        i = i + 1
        Rs.MoveNext
    Loop
%>
</TABLE>

<HR>
<Input type = hidden Name = SoLuongSach value = <%=Rs.RecordCount%> >
<Input type = Submit Value= " Đặt hàng ">

</Form>
</BODY>
</HTML>
```

GLOBAL.ASA

```
<Script language = VBScript Runat = Server>
    Sub Sesssion_OnStart    '/// Thủ tục này được gọi khi có 1 người mới thăm
        Session("DaDangNhap") = "chua"    '/// Biến DaDangNhap khởi tạo= "chua"
        Session("UserName")    = ""        '/// Biến tên của người dùng
    End Sub
</SCRIPT>
```

```

<HTML>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Đặt hàng trực tuyến</title>
</head>
<BODY>

<h1 align =center> CÁC CUỐN SÁCH BẠN ĐÃ ĐẶT MUA </h1>
<H2 ALIGN=CENTER>Nếu bạn đồng ý mua, hãy click vào nút "Đồng ý mua" </H2>

<HR>

<Form action = Thanhtoan.asp method = post>
<TABLE Border = 1 Width=100%>
<TR><TD>Tên sách <TD>Số lượng <TD> Đơn giá <TD> Thành tiền

<%
    Dim i, TenSach, TongSoCuonSach, TongTien, Gia, SoLuong, ThanhTien

    TongTien = 0
    TongSoCuonSach = Request.form("SoLuongSach")

    For i = 1 to TongSoCuonSach

        TenSach = Request.Form("TenSach" &i)
        SoLuong = Request.form("SL" &i)
        Gia      = Request.form("Gia" &i)

        IF SoLuong > 0 Then          ' /// Chỉ hiển thị và tính toán khi số lượng> 0
            ThanhTien = SoLuong * Gia
            TongTien = TongTien + ThanhTien

            Response.Write("<TR>")
            Response.Write("<TD>" & TenSach)
            Response.Write("<TD>" & SoLuong)
            Response.Write("<TD>" & Gia)
            Response.Write("<TD>" & ThanhTien)

        End IF
    Next
    Response.Write("<TR><TD Colspan = 4>Tổng số tiền là : " & TongTien)

%>
<HR>
<Input type = hidden Name = TongTienMua value = <%= (TongTien)%> >
<Input type = Submit Value= " Đồng ý mua ">

</TABLE>
</Form>
</BODY>
</HTML>

```

Trang này có nhiệm vụ thông kê lại toàn bộ những cuốn sách và số tiền tương ứng. Sau đó chuyển tổng số tiền sang trang **ThanhToan.asp**.


```

<HTML>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Thanh toán trực tuyến</title>
</head>
<BODY>

<!-- #Include file = "KetNoi.inc" -->

<h1 align =center> KẾT QUẢ THANH TOÁN </h1>
<HR>

<%
    IF Session("DaDangNhap") = "roi" Then
        Dim TongTienMua, SoDuTaiKhoan, MaKhachHang, strSQL

        '/// Lấy mã khách hàng từ biến session khi Login
        MaKhachHang = Session("UserName")
        Set Rs = Cn.Execute("Select * from tblUserDetail Where MaNguoiDung = '" &
                               MaKhachHang & "'")

        '/// Lấy số tiền còn dư trong tài khoản của khách hàng
        SoDuTaiKhoan = Rs("SoDuTaiKhoan")

        '/// Lấy tổng số tiền mua sách do trang thongke.asp gửi sang
        TongTienMua = Request.Form("TongTienMua")

        IF int(SoDuTaiKhoan) >= int(TongTienMua) Then '/// Thực hiện Thanh toán

            '/// Cộng tiền vào số dư tài khoản cho người bán
            strSQL = "UPDATE tblUserDetail Set SoDuTaiKhoan = SoDuTaiKhoan + "
                               &TongTienMua

            strSQL = strSQL & " WHERE MaNguoiDung = 'Seller'"
            Cn.Execute strSQL

            '/// Trừ tiền vào số dư tài khoản của người mua hàng
            strSQL = "UPDATE tblUserDetail Set SoDuTaiKhoan = SoDuTaiKhoan - "
                               &TongTienMua

            strSQL = strSQL & " WHERE MaNguoiDung = '" & MaKhachHang & "'"
            Cn.Execute strSQL

            '// Thông báo thanh toán thành công
            Response.Write("Cảm ơn bạn đã mua hàng." )

        Else '/// Không đủ tiền thanh toán
            Response.Write("Rất tiếc, Bạn không đủ thanh toán <BR>")
            Response.write("Bạn hiện có: " &SoDuTaiKhoan & "<BR>")
            Response.write("Số tiền bạn phải thanh toán là : " &TongTienMua)
        End IF
    Else
        Response.Write("Bạn chưa đăng nhập ! <a href=DangNhap.asp>Đăng nhập </a>")
    End IF
%>

</BODY>
</HTML>

```

*** Hàm int (s) để chuyển một xâu thành số. Ví dụ: **int**("100") → 100

++ Seller là tài khoản của người bán hàng.