

## Exemplos para construção de páginas em ASP

### 1.) Abertura de uma Base de Dados

O código necessário para fazer a abertura de uma Base de Dados Access e execução de uma string SQL é muito simples. No exemplo vamos utilizar uma Base e Dados chamada «basedados.mdb» e que contém pelo menos uma tabela chamada «tabela» com os campos chamados «campo1», «campo2» e «campo3». Este campos vão ser lidos para dentro de 3 variáveis c1, c2 e c3 respectivamente.

Depois, dentro de um ciclo *WHILE..LOOP* o conteúdo dos campos vai ser lido e mostrado até que não existam mais registos.

```
<%
accessdb=Server.MapPath("basedados.mdb")
cn="driver={Microsoft Access Driver (*.mdb)};"
cn=cn & "dbq=" & accessdb
set rs = server.createobject("ADODB.Recordset")
sql = "SELECT * FROM tabela"
rs.Open sql, cn
do while (Not rs.EOF)
    c1=rs("campo1")
    c2=rs("campo3")
    c3=rs("campo3")
    Response.write c1 & " - " & c2 & " - " & c3 & "<br>"
    rs.movenext
Loop
rs.close
set rs=nothing
%>
```

## 2.) Listagem Página a Página

Este exemplo é bastante útil porque, por vezes, torna-se necessário fazer uma listagem de registos página a página. Isto quer dizer que, embora a instrução SQL de selecção de registos retorne múltiplos registos, eles serão apresentados com um número máximo e fixo por página. Facilmente poderemos navegar de página em página. Este exemplo pressupõe que é passado um parâmetro indicador da página que se pretende ver.

Neste exemplo são apresentados 10 registos por página. Para este exemplo foi utilizada uma base de dados ACCESS chamada «BaseDados.mdb» com pelo menos uma tabela chamada «Tabela». Por sua vez esta tabela tem três campos, o «campo1», «campo2» e o «campo3».

Ter em atenção que é necessário fazer o include do *adovbs.inc*, que se encontra em `c:\Programas\Ficheiros comuns\System\Ado`.

Em alternativa podemos somente declarar as três constantes que necessitamos. Podemos fazer da seguinte maneira:

```
Const adOpenStatic = 3
```

```
Const adLockReadOnly = 1
```

```
Const adCmdText = &H0001
```

Vejamos então o código.

```
<!--#include file="adovbs.inc"-->
```

```
<%
```

```
'Esta constante contém o número de linhas por página
```

```
Const NumeroLinhasPorPagina = 10
```

```
'+++++ Recebe o parâmetro da página a ver +++++
```

```
PaginaActual=Request.QueryString("pagina")
```

```
if PaginaActual="" then
```

```
    PaginaActual=1 'A variável PaginaActual vai conter a página
```

```
end if 'que se pretende ver
```

```
'+++ Estabelece a conexão à base de dados e cria RecordSet++++
```

```
accessdb=Server.MapPath("BaseDados.mdb")
```

```
cn="driver={Microsoft Access Driver (*.mdb)};"
```

```
cn=cn & "dbq=" & accessdb
```

```
set rs = server.createobject("ADODB.Recordset")
```

```
sql="SELECT * FROM Tabela"
```

```
rs.Open sql, cn, adOpenStatic, adLockReadOnly, adCmdText
```

```
'As constantes utilizadas estão definidas no adovbs.inc
```

```
NumeroRegistos=rs.RecordCount
```

```
rs.PageSize=NumeroLinhasPorPagina
```

```
NumeroPaginas=rs.PageCount
```

```
ContadorLinhas=0
```

```
if Not(NumeroRegistos=0) then
```

```
    Response.write "Página " & PaginaActual & " de " & NumeroPaginas
```

```
    rs.AbsolutePage=PaginaActual 'Posiciona-se na página pretendida
```

```
    While Not rs.EOF and ContadorLinhas<NumeroLinhasPorPagina
```

```
        Response.write rs("campo1") & " - " & rs("campo2") & " - " & rs("campo3") &
"<br>"
        rs.movenext
        ContadorLinhas=ContadorLinhas+1
    Wend
else
    Response.write "<p class='Verdana12azul'><b>Não Existem Registos</b>"
end if
%>
```

### 3.) Leitura de Arquivos de Texto em ASP

Por vezes torna-se útil utilizar arquivos de texto (.txt) em páginas ASP. Neste pequeno exemplo vemos como instanciar um objecto de Ficheiro (FileSystemObject), abrir um ficheiro e ler o seu conteúdo. Eis o código necessário:

```
<%  
ficheiro=Server.MapPath("FicheiroTexto.txt")  
Set FS=Server.CreateObject("Scripting.FileSystemObject")  
Set TStream=FS.OpenTextFile(ficheiro)  
While Not TStream.AtEndOfStream  
    Response.write Trim(TStream.Readline) & "<br>"  
Wend  
Set TStream=nothing  
Set FS=nothing  
%>
```

#### 4.) Utilização de Cookies

*Cookies* são como variáveis de ambiente gravadas no computador cliente utilizando o objecto *Response*. Depois de criado, pode ser lido utilizando o objecto *Request*

```
<%
```

```
'Para criar um Cookie a instrução é muito simples
```

```
Response.Cookies("IdUtilizador") = 176987
```

```
'Com a instrução apresentada seria criado um Cookie com o nome «idUtilizador» e com o valor de 176987
```

```
'Para ler um Cookie e mostrá-lo no écran a instrução é muito simples
```

```
Response.write Request.Cookies("idUtilizadpr")
```

```
'Podemos ainda colocar data de validade a um Cookie
```

```
Response.Cookies("idUtilizador").Expires = #31/12/2001#
```

```
'Este Cookie tinha validade até ao final do ano
```

```
'Para a data de validade podíamos usar qualquer
```

```
'uma das funções do VBScript relacionadas com datas
```

```
como por exemplo:
```

```
Response.Cookies("idUtilizador").Expires = DateAdd("m",1,Now)
```

```
'Este Cookie perderia a validade ao fim
```

```
de um mês, a contar de agora
```

```
%>
```

É necessário ter atenção, porque os *Cookies* são enviados ao computador cliente como parte do cabeçalho HTTP. Por isso devem ser transmitidos ao cliente antes de outra coisa qualquer.

O exemplo seguinte provocava erro.

```
<html>
```

```
<h1>Os Cookies são Excelentes !!...</h1>
```

```
<%
```

```
Response.Cookies("IdUtilizador") = 176987
```

```
%>
```

```
</html>
```

## 5.) Criar Tabela com dados vindos de um Banco de Dados

Vamos usar uma base de dados *Access* chamada «BaseDados.mdb», que contém pelo menos uma tabela com o nome de «Tabela». Esta tabela tem quatro campos: Campo1, Campo2, Campo3 e Campo4.

Neste exemplo vamos fazer a abertura da base de dados ligeiramente diferente do utilizado nos outros exemplos. Dedicamos alguma atenção a essa secção.

```
<%
' Declare as variáveis... é sempre uma boa prática!
Dim cn ' Conexão ADO
Dim rs ' Recordset ADO
Dim AccessDB ' Base da Dados (*.mdb) com o caminho

' Conversão de caminho relativo para caminho absoluto.
' Se pretender pode digitar o caminho absoluto,
' mas assim podemos mudar este código de servidor
' que continua a funcionar.
AccessDB = Server.MapPath("BaseDados.mdb")

' Desta vez vamos fazer uma ligação
' à base de dados algo diferente dos outros exemplo.
' Em primeiro criamos uma conexão à base de dados
' chamada «BaseDados.mdb»
Set cn = Server.CreateObject("ADODB.Connection")

' Depois abrimos base de dados:
cn.Open "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & AccessDB & ";"

' Executamos a query usando
' o objecto de conexão. Ele automaticamente cria e
' retorna o recordset e guarda-o na nossa variável
Set rs = cn.Execute("SELECT * FROM Tabela")

' Agora vamos mostrar os dados constantes
' do recordset. Utilizamos um ciclo de repetição em todo o recordset
' mostrando os campos da tabela e usamos o Movenext para mudar para o
' registo seguinte. O ciclo para quando se atingir o fim de ficheiro (EOF)
%>

<TABLE BORDER="1">
<%
Do While Not rs.EOF
    %>
```

```
<TR>
  <TD><%= rs.Fields("Campo1").Value %></TD>
  <TD><%= rs.Fields("Campo2").Value %></TD>
  <TD><%= rs.Fields("Campo3").Value %></TD>
  <TD><%= rs.Fields("Campo4").Value %></TD>
</TR>
<%
rs.MoveNext
Loop
%>
</TABLE>
<%
' Fecha o recordset e a conexão e desinstancia os objectos
rs.Close
Set rs = Nothing
cn.Close
Set cn = Nothing
%>
```

## 6.) Contador de acessos (utilizando um arquivo de texto)

```
<%  
' Declaração de Variáveis  
Dim FS, TStream ' Objecto para o FileSystemObject  
Dim Ficheiro ' Nome do ficheiro que vai ter a contagem  
Dim Contador ' Variavel que contém a contagem  
Dim UsaImagens ' Variável booleana que indica se usa imagens ou texto  
Dim I  
  
' Determina se usa imagens ou texto normal  
' Se preferir pode só colocar esta variavel True ou False  
' em vez de passar parâmetro  
UsaImagens = CBool(Request.QueryString("imagens"))  
  
' Determina o nome do ficheiro  
' que vai conter a contagem.  
' O nome tira partido do nome do ficheiro de  
' onde é chamado este script e depois adiciona-lhe .cnt.  
' Por exemplo se este script se chamar «contador.asp»  
' então o ficheiro que vai conter a contagem  
' vai-se chamar contador.asp.cnt.  
Ficheiro = Server.MapPath(Request.ServerVariables("SCRIPT_NAME") & ".cnt")  
  
' Cria o FileSystem Object  
Set FS = Server.CreateObject("Scripting.FileSystemObject")  
  
' Abre o ficheiro (1 = ParaLeitura, True = Cria)  
Set TStream = FS.OpenTextFile(Ficheiro, 1, True)  
  
' Lê do ficheiro a contagem actual  
If Not TStream.AtEndOfStream Then  
    ' A variável contador fica com a contagem actual  
    Contador = CLng(TStream.ReadAll)  
Else  
    ' Se não existe ficheiro ou está vazio iguala a zero  
    Contador = 0  
End If  
  
' Fecha e desinstancia o objecto  
TStream.Close  
Set TStream = Nothing  
  
' Incrementa o contador  
Contador = Contador + 1
```



```

' Cria um novo ficheiro sobrepondo-se ao anterior
Set TStream = FS.CreateTextFile(Ficheiro, True)

' Escreve a nova contagem
TStream.Write Contador

' Fecha e desinstancia o objecto
TStream.Close
Set TStream = Nothing

' Desinstancia o objecto FileSystemObject
Set FS = Nothing

' Até agora foi feita a parte complicada do contador
' Vamos agora mostrar os resultados
If UsaImagens Then
    ' Vai percorrer o contador para indentificar
    ' dígito por dígito e depois vai buscar as imagens correspondentes
    ' Pode fazer o download as imagens em
    ' http://csgnet.brinkster.net/jgouveia/exemplosasp/counter_imgs.zip
    ' Todas as imagens tem o nome de digit_?.zip, onde
    ' ? é o dígito correspondente
    ' As imagens devem ser colocadas num directorio chamado «imagens»
    For I = 1 to Len(Contador)
        ' Mostra a imagem
        Response.Write "<IMG SRC=../imagens/digit_"
        Response.Write Mid(Contador, I, 1)
        Response.Write ".gif" ALT=""
        Response.Write Mid(Contador, I, 1)
        Response.Write "" WIDTH=""20" HEIGHT=""27">"
    Next
Else
    ' No caso de não se querer imagens
    ' Mostra só o conteúdo da variável «Contador»
    Response.Write Contador
End If
%>

```

## 7.) Contador de Acessos (utilizando um Banco de Dados Access)

Para este exemplo é utilizado uma base de dados chamada «contador.mdb» com pelo menos uma tabela chamada «acessos». Esta tabela, por sua vez, tem pelo menos dois campos, são eles: nome\_pagina e numero\_acessos.

Este contador de acessos dá para utilizar em mais de uma página.

```
<B><% = RetornaIncrementaContagem() %></B>
<%
' A contagem é feita numa função
' para que não haja problemas de coincidência de nomes de variáveis.
' Por exemplo, os nomes.
Function RetornaIncrementaContagem()
    ' Excerto do adovbs.inc para
    ' que assim não seja necessário
    ' fazer o include do mesmo.
    Const adOpenKeyset = 1
    Const adLockPessimistic = 2
    Const adCmdText = &H0001

    ' Declaração de variáveis locais
    Dim AccessDB
    Dim cn
    Dim NomeFicheiroPagina
    Dim SQL
    Dim rs
    Dim Contador

    ' Vai buscar o nome do ficheiro da página para assim
    ' poder retornar a contagem da página correcta.
    ' Controlo a string SQL para fazer isso
    NomeFicheiroPagina = Request.ServerVariables("SCRIPT_NAME")
    SQL = "SELECT nome_pagina, numero_acessos FROM acessos"
    SQL = SQL + " WHERE nome_pagina='" & NomeFicheiroPagina & "'"

    ' Abre o Recordset
    AccessDB = Server.MapPath("contador.mdb")
    cn = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source="
    cn = cn & AccessDB & ";"

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

    rs.Open SQL,cn,adOpenKeyset, adLockPessimistic, adCmdText

    ' Se existirem registos lê o valor actual
```

```
' senão criamos um.  
If rs.EOF Then  
    rs.AddNew  
  
    Contador = 0  
  
    rs.Fields("nome_pagina").Value = NomeFicheiroPagina  
Else  
    rs.MoveFirst  
  
    Contador = rs.Fields("numero_acessos").Value  
End If  
  
' Incrementamos um e actualizamos a BD  
rs.Fields("numero_acessos").Value = Contador + 1  
rs.Update  
  
' Fechamos a conexão  
rs.Close  
Set rs = Nothing  
  
' Retorna a contagem (antes do incremento)  
RetornaIncrementaContagem = Contador  
End Function  
>
```