



ASP-FR est hébergé par



Design et développement :



Paginer l'affichage des résultats dans une page ASP

Si votre base de données contient un nombre importants d'enregistrements, une page ASP qui en affiche le contenu risque de s'avérer longue à charger, voire parfois de générer une erreur en raison du délai d'exécution. Pour remédier à cela, nous allons voir comment, de façon simple, limiter l'affichage à un certain nombre d'enregistrements par page. Il s'agit en fait de pages virtuelles car, vous l'aurez deviné, une seule page ASP suffit à gérer l'ensemble de cette pagination.

Notre base de travail habituelle est toujours structurée comme suit:

Nom du champ	Paramètres
Id	Type NuméroAuto
Nom	Type Texte, Longueur 50 car.
Prenom	Type Texte, Longueur 30 car.
Age	Type Numérique (entier long)

Créons une page ASP vierge, que nous allons nommer **lire.asp**. Pour afficher nos enregistrements de façon paginée, nous allons devoir donner quelques petites instructions à notre page ASP:

```
<%
Select Case Request.QueryString("Direction")
Case ""
Session("CurrentPage") = 1
Case "Next"
Session("CurrentPage") = Session("CurrentPage") + 1
Case "Prev"
Session("CurrentPage") = Session("CurrentPage") - 1
End Select

Const adOpenKeyset = 1

Set objConnection = Server.CreateObject("ADODB.Connection")
objConnection.Open "mabase"
Dim strSQL
strSQL = strSQL & "SELECT * FROM table1 order by ID"
Set objRecordset = Server.CreateObject("ADODB.Recordset")
objRecordset.PageSize = 5
objRecordset.Open strSQL, objConnection, adOpenKeyset
objRecordset.AbsolutePage = CLng(Session("CurrentPage"))
%>
```

Très schématiquement, les instructions ci-dessus servent d'une part à dire à la page ASP ce qu'elle doit faire lorsqu'elle rencontre les paramètres **Next** (page suivante) ou **Prev** (page précédente), et d'autre part à stocker les données sélectionnées dans un jeu d'enregistrements (recordset).

Ensuite nous dimensionnons une variable nommée **i**, que nous allons faire s'auto-incréments de 1 à 5 (5 étant

dans notre exemple le nombre d'enregistrements à afficher par page). Notez que le simple fait de changer cette dernière valeur dans la ligne **objRecordset.PageSize** ci-dessus suffit à redimensionner la pagination sans autre manipulation.

```
<%  
Dim i  
For i = 1 To objRecordset.PageSize  
'une boucle de type For... Next est créée, à l'intérieur de laquelle nous allons afficher un à un les enregistrements  
>%>
```

```
<% =objRecordset("Nom") %> &nbsp; <% =objRecordset("Prenom") %><br>
```

On passe ensuite aux enregistrements suivants, en testant à chaque fois s'il existe encore des enregistrements. Dans la négative, on quitte la boucle **For... Next**.

```
<% objRecordset.MoveNext  
If objRecordset.EOF then  
exit for  
End If  
Next  
>%>
```

La boucle est terminée, nous allons donc interroger le jeu d'enregistrements pour savoir s'il existe des enregistrements suivants. Mais comme c'est la même page qui sert également à afficher la suite, nous allons aussi tester s'il existe des enregistrements précédents.

```
<%If CLng(Session("CurrentPage")) > 1 Then %>  
<a href="lire.asp?Direction=Prev">Enregistrements précédents</a>  
<% End If %>
```

```
<%If CLng(Session("CurrentPage")) < objRecordset.Pagecount Then %>  
<a href="lire.asp?Direction=Next">Enregistrements suivants</a>  
<% End If %>
```

Et pour terminer, on ferme la connection **ET** le jeu d'enregistrements.

```
<%  
objRecordset.Close  
objConnection.Close  
Set objRecordset = Nothing  
Set objConnection = Nothing  
>%>
```

[Jean Lançon](#), [MJ Presse](#)

[Retour à la liste des articles](#)



[Hébergement](#) [NT Server](#), [ASP](#), [ODBC](#), [FrontPage 2000](#), [SQL Server](#), [SSL](#), [commerce électronique](#), etc.
[En savoir plus](#)

[ARTICLES](#) | [Applications](#) | [Best of](#) | [FAQ](#) | [Forums](#) | [VBScript](#) | [JScript](#) | [Livres](#) | [Liens](#) | [Contact](#) | [ACCUEIL](#)