

Re: Populate a combo with an array

Source:

<http://www.tech-archive.net/Archive/Internet-Server/microsoft.public.inetsrvr.asp.general/2004-05/1238.html>

From: Dave Anderson (GTSPXOESSGOQ_at_spammotel.com)

Date: 05/19/04

Date: Wed, 19 May 2004 13:01:40 -0500

Lukelrc wrote:

```
>
> <select name="txtAvailable" rows="4" id="txtAvailable">
> <Script Language = "vbscript" Runat = "Server">
> For i = 0 to Count -1
> Response.Write"<OPTION>" & ListOfFiles(i) & "</OPTION>"
> next
> </script></select>
>
> I placed the vbscript in the appropriate place in the body. But when i
> open the page instead of writing the HTML where i placed the code, it
> has appended it right at the bottom of the page. Does anyone know
> where i'm going wrong or if there's a better way of doing this?
```

This is an order-of-execution issue (<http://aspfaq.com/show.asp?id=2045>). If you want to interlace server-side scripting with HTML, you will need <% %> blocks.

I usually abstract as much processing as possible before writing any HTML. In your case, I would build a single string to represent the entire array...

```
FilenameOptionList = CreateOptions(ListOfFiles,"")
```

...then insert it inline:

```
<SELECT><%=FilenameOptionList%></SELECT>
```

You can define the function anywhere in the script, either in <% %> blocks or in <SCRIPT RUNAT="Server"> blocks. Here is a sample function:

```
=====
Function CreateOptions(byVal A(), byVal DefaultValue)
    Dim i, val
    For i = 0 To UBound(A)
        val = Server.HtmlEncode(A(i))
        If A(i) = DefaultValue Then
            A(i) = "<OPTION VALUE="" & val & "" SELECTED>" & _
```

microsoft.public.inetserver.asp.general: Re: Populate a combo with an array

```
        val & "</OPTION>"
    Else
        A(i) = "<OPTION VALUE="" & val & "">" & _
            val & "</OPTION>"
    End If
Next
CreateOptions = Join(A,vbCrLf)
End Function
```

Note that this example is capable of preselecting the control:

```
... = CreateOptions(ListOfFiles,Request.Form("txtAvailable").Item)
```

Lastly, it bears mentioning that in many cases, this abstraction allows you to simply use `<SELECT><%=CreateOptions(...)%)></SELECT>`

More complicated functions can be written to create OPTION lists from 2D arrays, wherein the .value and .text properties differ (`<OPTION VALUE="TN">Tennessee</OPTION>`). Either way, this type of function is entirely self-contained, so it can be placed into a common include for reuse wherever needed.

If you use JScript on the server side, you can even do something like this...

```
Array.prototype.toOptions = function(val) {
    for (var i=0,a=[]; i<this.length; i++)
        a[i] = "\r\n<OPTION VALUE=\"" + this[i] +
            (this[i]==val ? "\" SELECTED>" : "\">") +
            this[i] + "</OPTION>"
    return a.join("")
}
```

...in which case:

```
<%=ListOfFiles.toOptions()%)>
```

would suffice, but:

```
<%=ListOfFiles.toOptions(Request.Form("txtAvailable").Item)%)>
```

works even better, and doesn't require a different method signature.

--

Dave Anderson

Unsolicited commercial email will be read at a cost of \$500 per message. Use of this email address implies consent to these terms. Please do not contact me directly or ask me to contact you directly for assistance. If your question is worth asking, it's worth posting.