Listbox Web Server Control

Listbox web server control is equivalent to the <Select> tag in HTML. This control will be very useful to get inputs from user, were all values are pre-defined. For eg: inorder to accept the "State in which the user is living", we can have a list of all states available. In this session, we will learn how to populate a listbox control.

- How to populate a Listbox control
- Populating the listbox during runtime, using Databinding
- Selecting Multiple items in a Listbox
- How to Remove items
- Insert items to a particular position

ListBox - How to populate a Listbox control?

Populating a listbox control is very simple. It is the same way as we do in Classic ASP. In classic ASP, we use the <option> tag to add an item to the listbox. In ASP .NET, we use the tag <asp:ListItem> to populate the Listbox.

```
<HTML>
<HEAD>
<title>Listbox Web Server Control - by Das</title>

<script language="VB" runat="server">
Sub mySub(sender As Object, e As EventArgs)
    If lstStates.SelectedIndex > -1
        lblMsg.Text = "You selected: " + lstStates.SelectedItem.Text
    Else
        lblMsg.Text = "Didn't like any state?"
    End If
End Sub
</script>

</HEAD>
<body style="FONT: 10pt verdana">
<h3align=center>ASP.NET Listbox Server Control</h3>

<form ID="Form1" runat=server>
    <asp:ListBox id=lstStates Rows=1 runat="server">
        <asp:ListItem>Florida</asp:ListItem>
        <asp:ListItem>Georgia</asp:ListItem>
        <asp:ListItem>Ohio</asp:ListItem>
    </asp:ListBox>
    <asp:button Text="Submit" OnClick="mySub" runat="server" id=Button1 />
    <asp:Label id=lblMsg ForeColor="#ff0000" Runat=server />
</form>

</body>
</HTML>
```

How to dynamically populate a Listbox control?

We can add items to a Listbox control during run time. This is achieved by populating an arrayList. Once we have an arrayList, we can assign this arrayList to the Listbox control. Actually, we have to bind the arrayList to the listbox control. Binding is done using the method DataBind. Before any binding to take place, we should assign a datasource, in our case, it will be the arrayList. The following example does this.

```
<HTML>
<HEAD>
<title>Listbox Web Server Control - by Das</title>

<script language="VB" runat="server">

    Sub Page_Load(sender As Object, e As EventArgs)
    If Not IsPostBack
        Dim arrState As New ArrayList()
        arrState.Add("Ohio")
        arrState.Add("Michigan")
        arrState.Add("Wisconsin")
        arrState.Add("Indiana")
        lstStates.DataSource = arrState
        lstStates.DataBind()
    End If
End Sub

Sub mySub(sender As Object, e As EventArgs)
    If lstStates.SelectedIndex > -1
        lblMsg.Text = "You selected: " + lstStates.SelectedItem.Text
    Else
        lblMsg.Text = "Didn't like any state?"
    End If
End Sub
</script>

</HEAD>
<body style="FONT: 10pt verdana">
<h3align=center>ASP.NET Listbox Server Control</h3>

<form ID="Form1" runat=server>
    <asp:ListBox id=lstStates Rows=1 runat="server">
    </asp:ListBox>
    <asp:button Text="Submit" OnClick="mySub" runat="server" id=Button1 />
    <asp:Label id=lblMsg ForeColor="#ff0000" Runat=server />
</form>

</body>
</HTML>
```

Selecting Multiple items in a Listbox

To allow users to select multiple values from a listbox, you need to set two properties, such as **rows** and **SelectionMode**. The value for rows can be any integer and for Selectionmode, it will be **Multiple**. To retrieve the all the values selected by the user, we need to make use of the properties such as **Items.Count**. And the property, **selected** will tell us whether the item have been selected or not. See the following example.

```
<HTML>
<HEAD>
<title>Listbox Web Server Control - by Das</title>

<html>
<head>

<script language="VB" runat="server">
Sub mySub(sender As Object, e As EventArgs)
    If lstStates.SelectedIndex > - 1
        lblMsg.Text=""
        Dim ctr as Integer
        For ctr=0 to lstStates.Items.Count()-1
            If lstStates.items(ctr).selected
                lblMsg.Text += "<br> You selected <b>" & lstStates.items(ctr).Text
& "</b>"
            end if
        Next
    End If
End Sub
</script>

</head>
<body style="font: 10pt verdana">
<h3 align=center>ASP.NET Listbox Server Control</h3>

<form runat=server>
<asp:ListBox id="lstStates" Rows="5" SelectionMode="Multiple" runat="server">
    <asp:ListItem>Ohio</asp:ListItem>
    <asp:ListItem>Michigan</asp:ListItem>
    <asp:ListItem>Wisconsin</asp:ListItem>
    <asp:ListItem>Texas</asp:ListItem>
    <asp:ListItem>Dallas</asp:ListItem>
    <asp:ListItem>Indiana</asp:ListItem>
</asp:ListBox><br>
<asp:button Text="Submit" Tooltip="To select multiple items hold the CTRL KEY
and then select the items" OnClick="mySub" runat="server" /><br>
<asp:Label id="lblMsg" runat="server"/>
</form>
```

```
</body>
</html>
```

How to Remove items and to Insert items to a particular position?

If we allow a user to add item to a listbox, then we also may need to allow them to delete items from the listbox. The method **remove** is used to remove a selected item from the list box. Also, to insert a value to a particular position, say 2nd position, we need to use the method called **insert**. In the coming example, we will see how to remove an item, insert an item into a particular position and also another method to popopulate a Listbox.

```
<HTML>
<HEAD>
<title>Listbox Web Server Control - by Das</title>

<html>
<head>

<script language="VB" runat="server">

   Sub mySub(sender As Object, e As EventArgs)
       Dim myItem as New ListItem
       myItem.value="ohio"
       myItem.text="Ohio"
       lstStates.items.Add(myItem)
       myItem=Nothing

       myItem = New ListItem
       myItem.value="wisconsin"
       myItem.text="Wisconsin"
       lstStates.items.Add(myItem)
       myItem=Nothing

       myItem = New ListItem
       myItem.value="indiana"
       myItem.text="Indiana"
       lstStates.items.Add(myItem)
   End Sub

   Sub myRemove(sender As Object, e As EventArgs)
       lstStates.items.Remove(lstStates.selectedItem)
   End Sub

   Sub myInsert(sender As Object, e As EventArgs)
       if Len(Trim(txtAt.text)) > 0
           If IsNumeric(txtAt.text) and Cint(txtAt.text) <= lstStates.items.count()
               lstStates.items.Insert(txtAt.text, "Inserted item")
           end if
```

```
      end if
   End Sub
</script>

</head>
<body style="font: 10pt verdana">
<h3 align=center>ASP.NET Listbox Server Control</h3>

<form runat=server>
   <asp:ListBox id="lstStates" Rows="5" SelectionMode="Multiple"
runat="server">
   </asp:ListBox>

   <asp:button Text="Populate" OnClick="mySub" runat="server" />
   <asp:button Text="Remove" Tooltip="To remove an item, select and then click
this button" OnClick="myRemove" runat="server" />
   <asp:button Text="Insert" Tooltip="To insert an item, type the position in
which you would like to insert." OnClick="myInsert" runat="server" /> at
   <asp:textbox id="txtAt" runat="server" />
</form>

</body>
</html>
```

Conclusion

Listbox control will be useful to get inputs, which has a pre-determined number of values. Also, you should note that, in the method, PageLoad and mySub we have IF statements. Have you noticed that, we don't have the keyword "Then" for IF statements. "Then" is not mandatory, as it was in classic ASP