

ASP .NET 3.5 Web Application Development

Content:

1. ASP .NET 3.5 Web Application Development

Contents

1	ASP .NET 3.5 WEB APPLICATION DEVELOPMENT	3
1.1	ASP .NET AJAX.....	4
1.2	ASP .NET SILVERLIGHT	6
1.3	ADO.NET DATA SERVICES	9
1.4	ASP .NET DYNAMIC DATA	13

1 ASP .NET 3.5 WEB APPLICATION DEVELOPMENT

Overview

- ASP .NET AJAX
- ASP .NET Silverlight
- ASP .NET Data Services
- ASP .NET Dynamic Data

1.1 ASP .NET AJAX

■ Create a New Web Site

1. Start **Visual Studio .Net 2008** and select the **File | New | Web Site** menu command. In New Web Site Dialog Box. For **Templates** select **ASP .NET 3.5 Extension Web Site**. For **Languages** select **C#** and then click **OK** button.

2. Add the **UpdatePanel**.

3. Add **Three Button** inside the **UpdatePanel**. Set the **Text** as "Key 1", "Key 2", "Key 3"

4. Add **Two Label** inside the **UpdatePanel**.

5. In **Page Load** Event add:

```
Label1.Text = DateTime.Now.ToString();
```

6. Create **ButtonClick** Event Handler and associate to the **Click** Event for the three **Button**.

```
public void ButtonClick(object sender, EventArgs e)
{
    Label2.Text = ((Button)sender).Text;
}
```

7. Press **F5** to run the Web Site and view the function, the browser cannot "Back".

8. **Move** three **Button** outside the **UpdatePanel**.

9. Press **F5** to run the Web Site and view the function, the browser can "Back", but the whole page content is back including the date time.

10. Create a static private class variable **key**, initialize the value as "s".

```
private static String key = "s";
```

11. Create **OnNavigateHistory** Event Handler and associate to the **Navigate** Event for the **ScriptManager**.

```
public void OnNavigateHistory(object sender, HistoryEventArgs e) {
    Label2.Text = Server.HtmlEncode(e.State[key]);
}
```

12. Set the **EnableHistory** of the **ScriptManager** to "true".

13. In **UpdatePanel**, select **Triggers** properties, add the three **Button Click** event as **AsyncPostBack** Event.

14. In **ButtonClick** Event Handler, add the below code.

```
ScriptManager.GetCurrent(this).AddHistoryPoint(key, Label2.Text);
```

15. Press **F5** to run the Web Site and view the function, the browser can "Back", but only the key value content is "Back", the date time value is real time generated.

16. Set the **EnableStateHash** of the **ScriptManager** to "false".

17. Press **F5** to run the Web Site and view the function, the history point value append in the QueryString.

18. Remarks:

UpdatePanel cannot “Back”, as when “Back” the AJAX PostBack will call again, so the data will be updated, rather than load the previous page data.

In order to enable the Browser “Back” and show some data in previous history point, we need to add **HistoryPoint**, it is used to make the Browser can “back” and store previous page state value.

The **Trigger** of **AsyncPostBack** is used if we want to set the AsyncPostBack outside the UpdatePanel.

1.2 ASP .NET Silverlight

■ Add a New Page for Media Control

19. Add a New Page to the current project.
20. Add the **ScriptManager**.
21. Add the **MediaPlayer**.
22. Add a **New Folder** "media"
23. Add the **expressionstudio.wmv** file inside the media folder.
24. In the **MediaPlayer** control, in the smart tag, set the **Skin** as "Expression" and **Media Source** as "expressionstudio.wmv".
25. In the **MediaPlayer** control, set the **AutoPlay** as "true", **width** as "640", **height** as "480".
26. Press **F5** to run the Web Site and view the function.
27. In the **MediaPlayer** control, set the **AutoPlay** as "false"
28. Add a **HTML Button** below the MediaPlayer, set the **Value** property as "Play", in the HTML view, set the **onclick** property to "onPlay()".
29. Inside the <head>, add the below code.

```
<script type="text/javascript">
function onPlay()
{
    $find('MediaPlayer1').play();
    $get('Button1').disabled = "disabled";
}
</script>
```

30. Press **F5** to run the Web Site and view the function.
31. Add the **00.jpg, 01.jpg, 40.jpg** file inside the media folder.
32. In the **MediaPlayer** control, set the **Chapters** property, three Chapters, **Title** as "Opening", "Designing", "Order Checking", **Position** as "00", "10", "40" and associate to corresponding Thumbnail Image property.
33. Press **F5** to run the Web Site and view the function.
34. Add the **plane.wmv, sports.wmv** file inside the media folder.
35. Add the **playlist.asx** file inside the project.
36. In the **MediaPlayer** control, in the smart tag, set the **Media Source** as "playlist.asx".
37. Press **F5** to run the Web Site and view the function.
38. In the **MediaPlayer** control, set the **onClientCurrentStateChanged** property as "onStateChanged" and **onClientMediaOpened** as "onMediaOpened"

39. Add **four Label** below the play button for “Movie URL”, “Movie Time”, “Movie Duration”, “State”.
40. Inside the <script>, add the below code.

```
var _clip = 0;
    function onMediaOpened(sender, args) {
        var movieUrl = sender.get_mediaElement().Source;
        $get('Label1').innerHTML = "Loading Media: " + sender.get_mediaSource() + ":
Clip#: " + ++_clip;
        $get('Label2').innerHTML =
String(parseInt(sender.get_position()*10000.0)/10000.0).substring(0, 10);
        $get('Label3').innerHTML = sender._duration;
    }

    function onStateChanged(sender, args)
    {
        $get('Label4').innerHTML = sender._oldState;
    }
```

41. Press **F5** to run the Web Site and view the function.

■ Add a New Page for Silverlight Control

42. Add a New Page to the current project.
43. Add the **calculator.js** and **calculator.xaml** to the project.
44. Add the **ScriptManager**.
45. Add the **Silverlight** control, set the width "320", height "340", source "calculator.xaml"
46. In the **Silverlight** control, set the ClientType property "Custom.Calculator".
47. In the **ScriptManager**, in the Script property, add the **name** "SilverlightControl.js", the **path** "calculator.js"
48. Press **F5** to run the Web Site and view the function.

1.3 ADO.NET Data Services

■ Add a New Page

49. Add a new item **ADO.NET Entity Data Model**, name as **Northwind.edmx**.
50. In the Entity Data Model Wizard, select **Generate from Database**, and click Next.
51. Create a new connection to the database by using the **New Connection** button, or use one of the available databases if you have one pre-configured. Click Next and set Entities as **NorthwindEntities**.
52. In the last window, the wizard presents the list of tables, views, and stored-procedures available from the database. Leave the default selection for **tables** and, for simplicity, uncheck views and stored-procedures. Click Finish and set **NorthwindModel** to close the wizard.
53. Add a new item **ADO.NET Data Service**, name as **Northwind.svc**.
54. Replace the below code in the Northwind.svc

```
using System;
using System.Web;
using System.Collections.Generic;
using System.ServiceModel.Web;
using System.Linq;
using Microsoft.Data.Web;
using NorthwindModel;

public class Northwind : WebDataService<NorthwindEntities>
{
    public static void
        InitializeService(IWebDataServiceConfiguration config)
    {
        config.SetResourceContainerAccessRule("*",
            ResourceContainerRights.All);
    }
}
```

55. Press **F5** to run the Web Site and view the function.

[http://localhost:\[PortNo\]/\[ProjectName\]/Northwind.svc](http://localhost:[PortNo]/[ProjectName]/Northwind.svc)

[http://localhost:\[PortNo\]/\[ProjectName\]/Northwind.svc/Products](http://localhost:[PortNo]/[ProjectName]/Northwind.svc/Products)

[http://localhost:\[PortNo\]/\[ProjectName\]/Northwind.svc/Products\(1\)](http://localhost:[PortNo]/[ProjectName]/Northwind.svc/Products(1))

56. Add a new page, add below code inside <head>

```
<script type="text/javascript">
function doQuery() {
    var northwindService = new Sys.Data.DataService("Northwind.svc");
    northwindService.query("Customers", cbSuccess, cbFailure);
}
</script>
```

57. Add a callback function for failed operations

```
function cbFailure(error, context, operation) {
    $get("spanResults").innerHTML = "Error occurred while performing operation " + operation
    + ".";
}
```

58. Add a callback function for success operations

```
function cbSuccess(result, context, operation) {
    // Get Customers list and put into a table.
    var sb = new Sys.StringBuilder();
    sb.append("<table>");

    var firstRowOutput = false;
    for (idx in result) {
        var customer = result[idx];
        if (!firstRowOutput) {
            // Display the header row.
            sb.append("<tr>");
            for (key in customer) {
                if (key != "__metadata") {
                    sb.append("<th>");
                    sb.append(key);
                    sb.append("</th>");
                }
            }
            sb.append("</tr>");
            firstRowOutput = true;
        }

        // Display the data.
        sb.append("<tr>");
        for (key in customer) {
            if (key != "__metadata") {
                sb.append("<td>");
                sb.append(customer[key]);
                sb.append("</td>");
            }
        }
        sb.append("</tr>");
    }

    sb.append("</table>");
    $get("spanResults").innerHTML = sb.toString();
}
```

59. Add **ScriptManager**, add script name as "**MicrosoftAjaxDataService.js**"

60. Add the below HTML Markup

```
<p>Click the button below to get a list of all the Customers  
  from the Northwind database.</p>  
  
<input id="ButtonQuery" type="button"  
  value="Perform Query"  
  onclick="doQuery()" /><br />  
<br />  
<span id="spanResults"></span>  
</div>
```

61. Press **F5** to run the Web Site and view the function.

1.4 ASP .NET Dynamic Data

■ Add a New Page

62. Start **Visual Studio .Net 2008** and select the **File | New | Web Site** menu command. In New Web Site Dialog Box. For **Templates** select **Dynamic Data Web Site**. For **Languages** select C# and then click **OK** button.
63. In the web.config, set the **enableTemplates** as "true"
64. In the project, add the ASP .NET folder "App_Data".
65. In App_Data, add **northwind.mdf**.
66. In the project, add the ASP .NET folder "App_Code".
67. In App_Data, add **LINQ to SQL Class** name it "Northwind.dbml".
68. In Server Explorer, drag and drop the **Categories, Customers, Employees** and **Products** tables to the dbml view.
69. Press **F5** to run the Web Site and view the function.