

Data Binding --II

Outline of the Lecture

- ObjectList Control.
- Data Binding (Using Custom Class and ArrayList class) with object list mobile control.
- List Control.
- SelectionList Control.

ObjectList Control

- Strictly *databound* (using *DataSource* property and *DataBind* method).
- Use to show list or table of data objects.
- Support pagination.
- Similar to *DataGrid* in ASP.NET.

Data Binding (Using Custom Class and ArrayList class) with object list mobile control

- *IsPostBack* property allows you to determine whether the current page being posted back to itself. This property is very useful in deciding what to do when loading a Web Forms page, as you'll see later in the next example.

Example 18-1a

```
<%@ Register TagPrefix="mobile" Namespace="System.Web.UI.MobileControls"
    Assembly="System.Web.Mobile" %>
<%@ Page language="c#"
    Inherits="System.Web.UI.MobileControls.MobilePage" %>
<HEAD>
    <meta name="GENERATOR" content="Microsoft Visual Studio .NET 7.1">
    <meta name="CODE_LANGUAGE" content="C#">
    <meta name="vs_targetSchema"
content="http://schemas.microsoft.com/Mobile/Page">
```

Overview of ASP.NET

```
</HEAD>
<body Xmlns:mobile="http://schemas.microsoft.com/Mobile/WebForm">
<script language="C#" runat="server">
private class Accounts
{
    long LUserID;
    string SEmail,SFirst,SLast;
    bool Bstatus;
    public Accounts ( long id, string em, string f, string l, bool s)
        {
            LUserID= id;
            SEmail = em;
            SFirst = f;
            SLast = l;
            Bstatus = s;
        }
    public long UserID
        { get { return LUserID; } }
    public string Email
        { get { return SEmail; } }
    public string First
        { get { return SFirst; } }
    public string Last
        { get { return SLast; } }
    public bool Status
        { get { return Bstatus; } }
}
private void Page_Load ( object sender, EventArgs e)
{
    ArrayList AccountList = new ArrayList ();
    if (!IsPostBack)
    {
        AccountList.Add(new Accounts (1,"User1@yahoo.com","Sami","Issa",true));
        AccountList.Add(new Accounts (2,"User2@yahoo.com","Naji","Ali",false));
    }
}
```

Overview of ASP.NET

```

AccountList.Add(new Accounts (3,"User3@Hotmail.com","Najeeb","Issa",true));
AccountList.Add(new Accounts (4,"User4@yahoo.com","Fadi","Fawzi",true));
ObjectList1.DataSource =AccountList;
ObjectList1.DataBind ();
}
}
</script>
<mobile:Form id="Form1" runat="server">
<mobile:ObjectList id="ObjectList1" runat="server"
CommandStyle-StyleReference="subcommand"
LabelStyle-StyleReference="title"></mobile:ObjectList>
</mobile:Form>
</body>

```

Example 18-1b

```

using System;
using System.Collections;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Web;
using System.Web.Mobile;
using System.Web.SessionState;
using System.Web.UI;
using System.Web.UI.MobileControls;
using System.Web.UI.WebControls;
using System.Web.UI.HtmlControls;
namespace MobileWebApplication8
{
    /// <summary>
    /// Summary description for MobileWebForm1.
    /// </summary>
    public class MobileWebForm1 : System.Web.UI.MobileControls.MobilePage
    {
        protected System.Web.UI.MobileControls.ObjectList ObjectList1;

```

Overview of ASP.NET

```
protected System.Web.UI.MobileControls.Form Form1;
private void Page_Load(object sender, System.EventArgs e)
{
    // Put user code to initialize the page here
}
#region Web Form Designer generated code
override protected void OnInit(EventArgs e)
{
    //
    // CODEGEN: This call is required by the ASP.NET Web Form Designer.
    //
    InitializeComponent();
    base.OnInit(e);
}
/// <summary>
/// Required method for Designer support - do not modify
/// the contents of this method with the code editor.
/// </summary>
private void InitializeComponent()
{
    this.Load += new System.EventHandler(this.Page_Load);
}
#endregion
}
}
```

List Control

- *Declared* or *databound* list of items
- *Decorations: None | Bulleted | Numbered*
- *Items* can be set to act as *links*
- *Static* or *interactive* mode
- Support *pagination*
- Similar to *DataList* control in ASP.NET

Overview of ASP.NET

```
<mobile:List id="lstProduce" runat="server"
    Decoration="Bulleted">
    <Item Value="1" Text="Apple"></Item>
    <Item Value="3" Text="Banana"></Item>
</mobile:List>
```

SelectionList Control

- *Static* or *databound* list.
- Provides *UI* selection capability.
- Can display in different *modes*:
 - *Dropdown* | *ListBox* | *Radio Button* | *CheckBox* | *MultiSelect List*
- Selection doesn't trigger server event – requires *Command* control to submit form

```
<mobile:SelectionList id="slstProduce"
    runat="server" SelectType="Radio">
<Item Value="1" Text="Onion"></Item>
<Item Value="2" Text="Celery"></Item>
</mobile:SelectionList>
```