

单元3-

ASP.NET常用服务器控件

1. ASP.NET控件简介
2. 基本类控件介绍及实作
实作1-基本控件实作会员注册
3. 列表选择控件介绍及实作
实作2-列表选择控件实作图书选择
4. 按钮选择控件介绍及实作
实作3-按钮选择控件实作问卷调查
5. 其它类控件介绍

LearnASP.NET-rev1.1 <http://www.coding24.org>



1. ASP.NET控件简介

LearnASP.NET-rev1.1 <http://www.coding24.org>

- 本节先介绍HTML标签（控件）、HTML服务器控件及Web服务器控件 3 类。我们常说的Web服务器控件也就是ASP.NET服务器控件，它是ASP.NET所特有的，是在服务器端运行的，也是我们快速建构Web Form项目时所最常用的编程的基本元素。基本概念认识后，再一一介绍各项Web服务器控件的功能及其使用方法。

ASP.NET有3种控件

- ASP.NET有3类控件支持其快速开发，**我们常用的是Web服务器控件**

```

<form id="form1" runat="server">
  <div>
    <input id="b1" type="button" onclick="Fun()" value="HTML 标签" />
    <input id="b2" runat="server" type="button" onclick="Fun()" value="HTML 服务器控件" />
    <asp:Button ID="b3" runat="server" Text="Web 服务器控件" OnClick="b3_Click" />
  </div>
</form>

```

b1及b2都可调用JS，但都无服务端代码

b3透过服务端代码控制b2的显示. 较耗资源但功能强大适用快速开发

localhost:52628 显示
你好，我是一个弹窗!
确定

HTML 标签 ****可由后置代码控制**** Web 服务器控件

Source: Examples/Unit03/ControlTypes.aspx

LearnASP.NET-rev1.1 http://www.coding24.org 3

- **HTML标签**：就是常做静态网页用的HTML语言标记，这些语言标记主要用于呈现UI，不能在服务器端控制的，只能在客户端通过JavaScript控制或调用JavaScript程序。语法如下（显示一个按钮）：`<input type="button" id="b1" value="提交"/>`
- **HTML服务器控件**：在HTML标签基础上再加`runat="server"`所构成的控件。HTML标签在客户端运行，而HTML服务器控件则是在服务器端运行的，这样服务器端的代码就能对其进行控制。语法如下（显示一个按钮）：`<input id="b2" type="button" value="提交" runat="server" />`。
或如下例：
`管理`
在上述例子中，可按用户的角色，若是管理者，在登录后adminLink这个链接可显示。
- **Web服务器控件（常用的）**：也称ASP.NET服务器控件，是ASP.NET所特有的，是在服务器端运行的，也是我们快速建构Web Form项目时所最常用的编程的基本元素。语法如下（显示一个按钮）：`<asp:Button ID="b3" runat="server" Text="提交"/>`
- **ControlTypes.aspx**

```

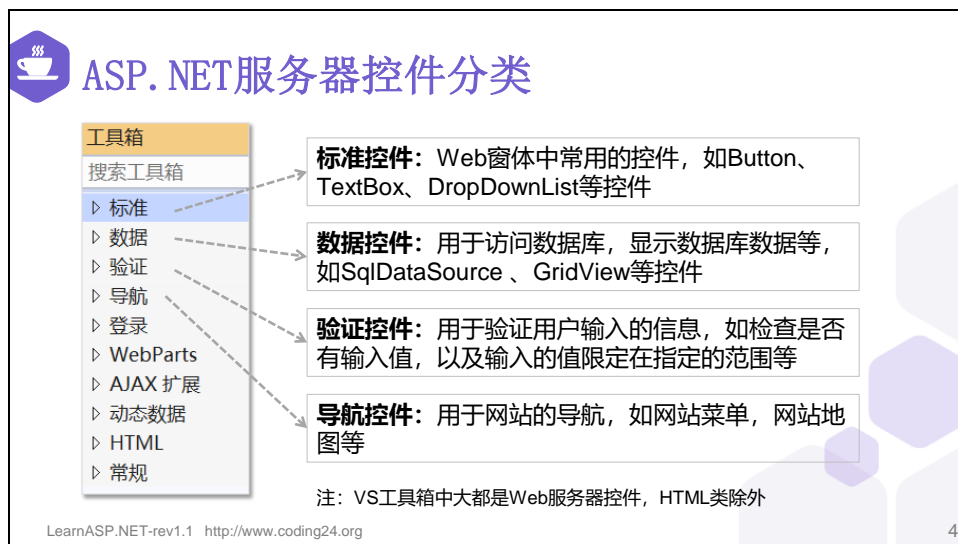
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="ControlTypes.aspx.cs"
Inherits="LearnASPNET.Examples.Unit03.ControlTypes" %>

<!DOCTYPE html>

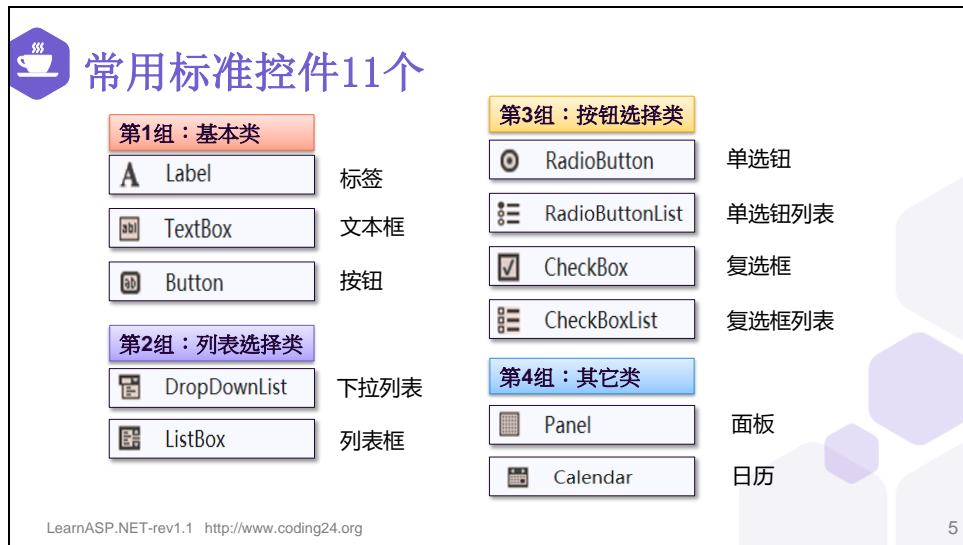
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
  <title>控件类型</title>
</script>

```

```
function Fun() {  
    alert("你好，我是一个弹窗！");  
}  
</script>  
</head>  
<body>  
    <form id="form1" runat="server">  
        <div>  
            <input id="b1" type="button" onclick="Fun()" value="HTML标签" />  
            <input id="b2" runat="server" type="button" onclick="Fun()" value="HTML  
服务器控件" />  
            <asp:Button ID="b3" runat="server" Text="Web服务器控件"  
OnClick="b3_Click" />  
        </div>  
    </form>  
</body>  
</html>
```



- **标准控件:** Web窗体中常用的控件, 如按钮、文本框、下拉列表等控件外, 还包含一些特殊用途的控件, 如日历等。
- **数据控件:** 用于连接访问数据库, 显示数据库数据等。
- **验证控件:** 用于验证用户输入的信息, 如检查是否有输入值, 以及输入的值限定在指定的范围等。
- **导航控件:** 用于网站的导航, 如菜单, 网站地图等。
- **其余控件类:** 在特定情境时使用, 较少用到。



- **Label**控件：标签，显示信息用。
- **TextBox**控件：文本框，让用户输入信息。
- **Button**控件：按钮，通常用于绑定事件过程，运行指定代码。
- **DropDownList**控件：下拉列表，单选，可让用户在列表中选择一项目（item），一次仅可呈现一项目。
- **ListBox**控件：列表，可单选或多选。可让用户在列表中选择一或多个项目（item），且一次可呈现多项目。
- **RadioButton**控件：单选按钮
- **RadioButtonList**控件：单选按钮列表
- **CheckBox**控件：复选框
- **CheckBoxList**控件：复选框列表
- **Image**控件：图像，显示图像
- **Panel**控件：面板，为其他控件提供一个容器



标准控件常见的属性

| 属性名 | 说明 | 属性名 | 说明 |
|-----------|-----------|------------|-----------------|
| ID | 识别名称，具唯一性 | Enabled | 是否启用控件 |
| Text | 显示的文本信息 | AccessKey | 控件的键盘快捷键 |
| Visible | 控件是否显示 | Attributes | 控件的所有属性集合 |
| Width | 控件的宽度 | Font | 控件的字体属性 |
| Height | 控件的高度 | ToolTip | 当鼠标悬停在控件上时显示的文本 |
| ForeColor | 前景(文字)景色 | BoderWidth | 控件的边框宽度 |
| BackColor | 控件的背景色 | | |

LearnASP.NET-rev1.1 <http://www.coding24.org>

6

- 标准控件常见的属性：各种常见的标准控件都有下面这些属性（按常用顺序），可透过属性视窗或透过程序设定以及改变控件的属性。
 - ID属性：识别名称，具唯一性
 - Text属性：显示的文本信息
 - Visible属性：控件是否显示
 - Width / Height属性：控件的宽度及高度
 - ForeColor / BackColor属性：控件的前景（文字） / 背景颜色
 - Enabled属性：设定控件是否启用，值为true/false
 - AccessKey属性：控件的键盘快捷键
 - Attributes属性：控件的所有属性集合
 - Font属性：控件的字体属性
 - ToolTip属性：当鼠标悬停在控件上时显示的文本
 - BoderWidth属性：控件的边框宽度
 - 此外尚有BoderStyle / CssClass等。



常用控件名简写建议（不强制）

| 控件名 | 简写 | 举例 | 控件名 | 简写 | 举例 |
|-----------------|-----|------------|--------------|-----|-------------|
| Label | lbl | lblMessage | CheckBox | cb | cbCategory |
| TextBox | txt | txtName | CheckBoxList | cbl | cblCity |
| Button | btn | btnSubmit | Image | img | imgCampus |
| ListBox | lb | lbSubject | Panel | pnl | pnlMain |
| DropDownList | ddl | ddlDept | Table | tbl | tblStuent |
| RadioButton | rb | rbGender | GridView | gv | gvProduct |
| RadioButtonList | rbl | rblGender | Calendar | cld | cldBirthday |

LearnASP.NET-rev1.1 <http://www.coding24.org>

7

- ASP.NET (C#)的命名规范（非强制）分为Pascal和Camel两种形式：
 - Pascal形式将标识符的首字母大写和后面连接的每个单词的首字母都大写。如，某个控件或类的属性名BackColor。
 - Camel形式将标识符的首字母小写，而每个后面连接的单词的首字母都大写。如，变量名typeName。
 - 控件其实是一个对象，是以Camel（小写开始）命名形式，其它如参数名，变量名，亦用此形。
- 微软的命名方式与Java社区的命名方式有较大的不同，生活在那也只好入境随俗了。
- 在快速开发时，控制只用于显示非用于代码控制，可考虑不花时间重命名。如Label在很多情况只用于呈现提示信息而已。



Page_Load事件和IsPostBack属性

- Page_Load事件：即页面载入要执行的事件程序
- Page_Load的执行分为两种情况：
 - 第一次加载页面时发生（响应客户的请求）
 - 页面回发（Post Back）到服务器时发生
- Page.IsPostBack属性

```
//若语句仅要运行一次,可判断IsPostBack的值  
if ( !IsPostBack ) {  
    //语句仅被运行一次,通常用于初值设定  
}
```

LearnASP.NET-rev1.1 <http://www.coding24.org>

8

- Page_Load事件：即页面载入要执行的事件过程
- Page_Load的执行分为两种情况;
 - Page_Load事件的执行是在第一次加载页面时发生(即为了响应客户的请求);
 - Page_Load事件的执行是在把该页面回发到服务器时发生;
- 透过if(!IsPostBack){//....} 语句，可在Page_Load事件中的代码仅被运行一次，通常是用作初值设定用途。



2. 基本类控件介绍及实作

LearnASP.NET-rev1.1 <http://www.coding24.org>



Label控件

- 用途：用于显示文本信息
- 常用属性：ID, Text / 常用方法及事件：无
- 语法：

```
<asp:Label ID="lblMessage" runat="server" Text="信息显示!"></asp:Label>
```

- 属性设置方式：

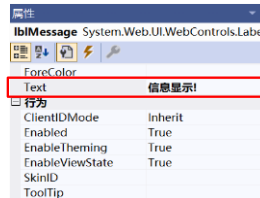
1. 透过属性视窗设定



2. 透过代码设定

//使用范例

```
lblCheck.Text = "用户名已占用!";
```



LearnASP.NET-rev1.1 <http://www.coding24.org>

10

- Label 控件用于在页面上显示文本信息且该文本信息是可透过代码控制的，是最常见的控件之一。



TextBox控件

- 用途：可以用于输入或者显示文字信息
- 语法：

```
<asp:TextBox ID="txtName" runat="server" AutoPostBack="True"
    OnTextChanged="txtName_TextChanged"></asp:TextBox>
```

| 属性、方法和事件 | 说明 |
|----------------|---|
| TextMode属性 | 设置文本框类型。如Password表示密码框，MultiLine表示多行文本框 |
| AutoPostBack属性 | 值true表示当文本框内容改变且把焦点移出时触发TextChanged事件 |
| Focus()方法 | 设置TextBox焦点 |
| TextChanged事件 | 当改变文本框中内容且焦点离开文本框后被触发 |

LearnASP.NET-rev1.1 <http://www.coding24.org>

11

- TextBox 控件可以用于输入或者显示文字信息，是控件中最常被使用之一。若TextMode属性设置为Password则为常见的密码输入框。



Button控件

- 用途：用于响应用户点击行为
- 语法：

```
<asp:Button ID="btnSubmit" runat="server" OnClick="btnSubmit_Click"
Text="提交" />
```

| 属性、方法和事件 | 说明 |
|---------------|--------------------------|
| OnClick属性 | 当按钮被点击时被执行的函数的名称 |
| PostBackUrl属性 | 控件被点击时从当前页面传送数据的目标页面 URL |
| Click事件 | 当按钮被点击时被触发 |

LearnASP.NET-rev1.1 <http://www.coding24.org>

12

- **Button** 控件用于显示一个按钮，是一个提交按钮。当这按钮它被点击时会触发 **PostBack**事件，同时页面传回服务器。可以编写一些事件程序如**Click**事件，在按钮被点击时来执行一些代码。



基本控件实例

- 在页面中布置了Label、TextBox及Button各1个。输入名字后点击按钮将TextBox内容呈现到Label及Button上，同时将Button作用

暂停。
页面设计:

名字:

显示信息

运行后:

名字: 张三

张三 您好, 欢迎您!

后置代码:

```
protected void btnSubmit_Click(object sender, EventArgs e)
{
    lblShow.Text = txtName.Text + " 您好, 欢迎您! ";
    btnSubmit.Text = txtName.Text + " 您好, 欢迎您! ";
    btnSubmit.Enabled = false;
}
```

Source: Examples/Unit03/BasicControls.aspx

LearnASP.NET-rev1.1 <http://www.coding24.org>

13

BasicControls.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="BasicControls.aspx.cs"
Inherits="LearnASPNET.Examples.Unit03.BasicControls" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
<title>基本控件范例</title>
</head>
<body>
<form id="form1" runat="server">
<div>
名字: <asp:TextBox ID="txtName" runat="server"></asp:TextBox>
<br />
<br />
<asp:Button ID="btnSubmit" runat="server" Text="确定"
OnClick="btnSubmit_Click" />
<br />
<br />
<asp:Label ID="lblShow" runat="server" Text="显示信息"></asp:Label>
</div>
</form>
</body>
</html>
```

BasicControls.aspx.cs

```
using System;

namespace LearnASPNET.Examples.Unit03
{
    public partial class BasicControls : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {

        }

        protected void btnSubmit_Click(object sender, EventArgs e)
        {
            lblShow.Text = txtName.Text + " 您好, 欢迎您! ";
        }
    }
}
```

```
        btnSubmit.Text = txtName.Text + " 您好，欢迎您！ ";  
        btnSubmit.Enabled = false;  
    }  
}
```



实作1-基本控件实作会员注册

Source: Labs/Unit03/Register.aspx

LearnASP.NET-rev1.1 <http://www.coding24.org>



实作1-基本控件实作会员注册

- 实作说明：本实作以Label/TextBox/Button控件实作会员注册功能

- 实作步骤：

1. 请使用实作方案 / 项目：LabASPNET，在Labs文件夹上添加子文件夹Unit03。（若无请按单元1实作2创建）
2. 在文件夹Unit03中添加「Web窗体」->命名为Register.aspx，页面设计如下：

LearnASP.NET-rev1.1 <http://www.coding24.org>

15



实作1-基本控件实作会员注册

3. 控件名称: txtName, txtPassword, txtEmail, lblCheck, lblMessage, btnSubmit, btnCancel
4. 控件属性: txtName->AutoPostBack: True, txtPassword->TextMode: Password
5. 重要代码如下:

```
protected void Page_Load(object sender, EventArgs e)
{
    if (!IsPostBack)
    {
        txtName.Focus();
    }
}

protected void btnReset_Click(object sender, EventArgs e)
{
    txtName.Text = txtPassword.Text = txtEmail.Text = "";
    lblCheck.Text = "*";
}
```

LearnASP.NET-rev1.1 <http://www.coding24.org>

16



实作1-基本控件实作会员注册

```
protected void txtName_TextChanged(object sender, EventArgs e)
{
    if (txtName.Text == "admin")
    {
        lblCheck.Text = "用户名已占用! ";
    }
    else
    {
        lblCheck.Text = "OK!";
    }
}

protected void btnSubmit_Click(object sender, EventArgs e)
{
    string message = $"你的信息如下: 帐号:{txtName.Text}, " +
        $"密码:{txtPassword.Text}, 邮箱:{txtEmail.Text}";
    lblMessage.Text = message;
}
```

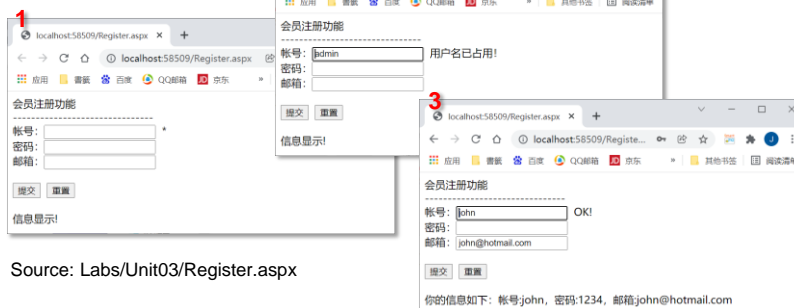
LearnASP.NET-rev1.1 <http://www.coding24.org>

17



实作1-基本控件实作会员注册

Register.aspx运行结果:



Source: Labs/Unit03/Register.aspx

LearnASP.NET-rev1.1 http://www.coding24.org

18

● Register.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Register.aspx.cs"
Inherits="LearnASPNET.Labs.Unit03.Register" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <title>实作:会员注册</title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            会员注册功能(若输入admin则有警告)<br />
            -----<br />
            帐号: <asp:TextBox ID="txtName" runat="server" AutoPostBack="True"
OnTextChanged="txtName_TextChanged"></asp:TextBox>
            &nbsp;<asp:Label ID="lblCheck" runat="server" Text="*"></asp:Label>
            <br />
            密码: <asp:TextBox ID="txtPassword" runat="server"
TextMode="Password"></asp:TextBox>
            <br />
            邮箱: <asp:TextBox ID="txtEmail" runat="server"></asp:TextBox>
            <br />
            <asp:Button ID="btnSubmit" runat="server" OnClick="btnSubmit_Click"
Text="提交" />
            &nbsp;<asp:Button ID="btnReset" runat="server" OnClick="btnReset_Click"
Text="重置" />
            <br />
            <asp:Label ID="lblMessage" runat="server" Text="信息显示!"></asp:Label>
        </div>
    </form>
</body>
</html>
```

● Register.aspx.cs

```
using System;

namespace LearnASPNET.Labs.Unit03
```

```
{
    public partial class Register : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            if (!IsPostBack)
            {
                txtName.Focus();
            }
        }

        protected void btnSubmit_Click(object sender, EventArgs e)
        {
            //string message = "你的信息如下: 帐号:" + txtName.Text + ", 密码:" +
            txtPassword.Text + ", 邮箱:" + txtEmail.Text;
            string message = $"你的信息如下: 帐号:{txtName.Text}, 密
            码:{txtPassword.Text}, 邮箱:{txtEmail.Text}";
            lblMessage.Text = message;
        }

        protected void btnReset_Click(object sender, EventArgs e)
        {
            txtName.Text = txtPassword.Text = txtEmail.Text = "";
            lblCheck.Text = "*";
        }

        protected void txtName_TextChanged(object sender, EventArgs e)
        {
            if (txtName.Text == "admin")
            {
                lblCheck.Text = "用户名已占用! ";
            }
            else
            {
                lblCheck.Text = "OK!";
            }
        }
    }
}
```



3. 列表选择控件介绍及实作

LearnASP.NET-rev1.1 <http://www.coding24.org>



DropDownList控件

- 用途：为一下拉式列表，让用户在列表选取一项，语法如下：

```
<asp:DropDownList ID="ddlCategory" runat="server" AutoPostBack="True" >  
  <asp:ListItem Value="1">桌面应用开发</asp:ListItem>  
  <asp:ListItem Value="2">Web应用开发</asp:ListItem>  
</asp:DropDownList>
```

| 属性/方法/事件 | 说 明 | 属性/方法/事件 | 说 明 |
|------------------|--------------------|------------------------|----------------|
| DataSource属性 | 设置数据源 | SelectedItem属性 | 当前选定项的项目 |
| DataTextField属性 | 对应数据源中的一个字段，显示其内容 | SelectedValue属性 | 当前选定项的Value属性值 |
| DataValueField属性 | 对应数据源中的一个字段，指定选项的值 | SelectedIndexChanged事件 | 当选择下拉列表中一项后被触发 |
| Items属性 | 列表中所有选项的集合 | DataBind()方法 | 绑定数据源 |

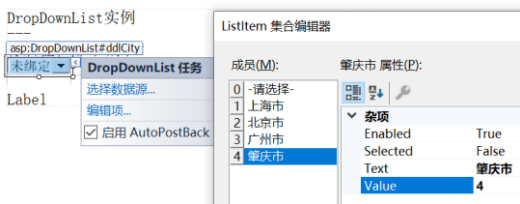
LearnASP.NET-rev1.1 <http://www.coding24.org>

20

- DataSource属性：设置数据源，一般是数据源控件或是一个数据集对象。
- DataTextField属性：对应数据源中的一个字段（如商品名），该字段所有内容将被显示于下拉列表中。
- DataValueField属性：对应数据源中的一个字段，指定下拉列表中每个可选项的值。
- Items属性：列表中所有选项的集合，常用Add()方法添加项，Clear()方法删除所有项。
- SelectedItem属性：当前选定项。
- SelectedValue属性：当前选定项的Value属性值。
- SelectedIndexChanged事件：当选择下拉列表中一项后被触发。
- DataBind()方法：绑定数据源。

DropDownList实例


1. 页面设计及控件输入资料:



2. 主要后置代码:

```
protected void ddlCity_SelectedIndexChanged(object sender, EventArgs e) {
    lblShow.Text = "你居住在: " + ddlCity.SelectedValue + "-" + ddlCity.SelectedItem.Text;
}
```

3. 运行结果:



Source: Examples/Unit03/DropDownList.aspx

LearnASP.NET-rev1.1 <http://www.coding24.org> 21

● DropDownList.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="DropDownList.aspx.cs"
Inherits="LearnASPNET.Examples.Unit03.DropDownList" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
<title>DropDownList实例</title>
</head>
<body>
<form id="form1" runat="server">
<div>
DropDownList实例<br />
---<br />
你居住在那个城市? <br />
<asp:DropDownList ID="ddlCity" runat="server" AutoPostBack="True"
OnSelectedIndexChanged="ddlCity_SelectedIndexChanged">
<asp:ListItem Value="0">-请选择-</asp:ListItem>
<asp:ListItem Value="1">上海市</asp:ListItem>
<asp:ListItem Value="2">北京市</asp:ListItem>
<asp:ListItem Value="3">广州市</asp:ListItem>
<asp:ListItem Value="4">肇庆市</asp:ListItem>
</asp:DropDownList>
<br />
<br />
<asp:Label ID="lblShow" runat="server" Text="Label"></asp:Label>
</div>
</form>
</body>
</html>
```

● DropDownList.aspx.cs

```
using System;

namespace LearnASPNET.Examples.Unit03
{
    public partial class DropDownList : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
        }
    }
}
```



```
    }  
  
    protected void ddlCity_SelectedIndexChanged(object sender, EventArgs e)  
    {  
        lblShow.Text = "你居住在: " + ddlCity.SelectedValue + "-" +  
        ddlCity.SelectedItem.Text;  
    }  
}
```



Listbox控件

- 用途：一次可呈现多个项目，让用户在列表中选择一或多项，语法如下：

```
<asp:ListBox ID="lstCity" runat="server">
  <asp:ListItem Value="1">广州市</asp:ListItem>
  <asp:ListItem Value="2">台北市</asp:ListItem>
</asp:ListBox>
```

| 属性、方法和事件 | 说明 |
|-----------------|--------------------------------------|
| SelectionMode属性 | 其值为Single表示仅允许选择一项；Multiple表示允许选择多项。 |
| 其它同DropDownList | |



实作2-列表选择控件实作 图书选择

Source: Labs/Unit03/BookSelect.aspx

LearnASP.NET-rev1.1 <http://www.coding24.org>



实作2-列表选择控件实作图书选择

- 实作说明：本实作以DropDownList/ListBox控件实作图书选择功能
- 实作步骤：

1. 请使用实作方案 / 项目：LabASP.NET，在Labs/Unit03添加页面。（若无请按单元1实作2创建）
2. 页面添加：添加「Web窗体」->命名为BookSelect.aspx，页面设计如下：

请选择你喜欢的IT书籍：

=请选择分类= ▾ DropDownList

未绑定
ListBox

你选择的书籍 Label

LearnASP.NET-rev1.1 <http://www.coding24.org>

24



实作2-列表选择控件实作图书选择

3. 控件名称: ddlCategory, lstBooks

4. 控件属性: ddlCategory ->
AutoPostBack: True, lstBooks ->
AutoPostBack: True

5. 控件内容如下:

DropDownList内容:

0, =请选择分类=
1, 桌面应用开发
2, Web应用开发
3, 行动应用开发

ListBox内容1:

"=请选择书籍=", "0"
"C#桌面应用项目开发教程", "1"
"Java桌面应用开发24H", "2"
"基于Electron跨平台桌面应用开发", "3"

ListBox内容2:

"=请选择书籍=", "0"
"Web全栈开发: 从入门到实战", "1"
"JavaScript从入门到精通", "2"
"ASP.NET Core 从入门到实战", "3"

ListBox内容3:

"=请选择书籍=", "0"
"移动开发Android Studio开发实战", "1"
"iOS App程序开发实务攻略", "2"
"Swift开发iOS App快速入门与实战", "3"



实作2-列表选择控件实作图书选择

```
ddlCategory_SelectedIndexChanged()                protected void lstBooks_SelectedIndexChanged()
{
    //清除lstBooks的内容
    lstBooks.Items.Clear();
    switch (ddlCategory.SelectedValue)
    {
        case "0": //只当提示
            lstBooks.Items.Add(new ListItem("请选择书籍=", "0"));
            lstBooks.Enabled = false;
            break;
        case "1": //桌面应用开发
            lstBooks.Items.Add(new ListItem("请选择书籍=", "0"));
            lstBooks.Items.Add(new ListItem("C#桌面应用项目开发教程", "1"));
            lstBooks.Items.Add(new ListItem("Java桌面应用开发24H", "2"));
            lstBooks.Enabled = true;
            // .....
    }
}
```

LearnASP.NET-rev1.1 <http://www.coding24.org>

26



实作2-列表选择控件实作图书选择

BookList.aspx运行结果:



Source: Labs/Unit03/BookSelect.aspx

LearnASP.NET-rev1.1 <http://www.coding24.org>

27

● BookSelect.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="BookSelect.aspx.cs"
Inherits="LearnASPNET.Labs.Unit03.BookSelect" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <title>实作：图书选择</title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            请选择你喜欢的IT书籍：<br />
            <br />
            <asp:DropDownList ID="ddlCategory" runat="server" AutoPostBack="True"
OnSelectedIndexChanged="ddlCategory_SelectedIndexChanged">
                <asp:ListItem Selected="True" Value="0">请选择分类</asp:ListItem>
                <asp:ListItem Value="1">桌面应用开发</asp:ListItem>
                <asp:ListItem Value="2">Web应用开发</asp:ListItem>
                <asp:ListItem Value="3">行动应用开发</asp:ListItem>
            </asp:DropDownList>
            <br />
            <br />
            <asp:ListBox ID="lstBooks" runat="server" AutoPostBack="True"
OnSelectedIndexChanged="lstBooks_SelectedIndexChanged"></asp:ListBox>
            <br />
            <br />
            <asp:Label ID="lblResult" runat="server" Text="你选择的书籍"
"></asp:Label>
        </div>
    </form>
</body>
</html>
```

● BookSelect.aspx.cs

```
using System;
using System.Web.UI.WebControls;

namespace LearnASPNET.Labs.Unit03
{
```

```

public partial class BookSelect : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {

    }

    protected void ddlCategory_SelectedIndexChanged(object sender, EventArgs e)
    {
        //清除lstBooks的内容
        lstBooks.Items.Clear();
        switch (ddlCategory.SelectedValue)
        {
            case "0": //只当提示
                lstBooks.Items.Add(new ListItem("=请选择书籍=", "0"));
                lstBooks.Enabled = false;
                break;
            case "1": //桌面应用开发
                lstBooks.Items.Add(new ListItem("=请选择书籍=", "0"));
                lstBooks.Items.Add(new ListItem("C#桌面应用项目开发教程",
"1"));
                lstBooks.Items.Add(new ListItem("Java桌面应用开发24H", "2"));
                lstBooks.Items.Add(new ListItem("基于Electron跨平台桌面应用开发",
"3"));
                lstBooks.Enabled = true;
                break;
            case "2": //Web应用开发
                lstBooks.Items.Add(new ListItem("=请选择书籍=", "0"));
                lstBooks.Items.Add(new ListItem("Web全栈开发：从入门到实战",
"1"));
                lstBooks.Items.Add(new ListItem("JavaScript从入门到精通",
"2"));
                lstBooks.Items.Add(new ListItem("ASP.NET Core 从入门到实战",
"3"));
                lstBooks.Enabled = true;
                break;
            case "3": //行动应用开发
                lstBooks.Items.Add(new ListItem("=请选择书籍=", "0"));
                lstBooks.Items.Add(new ListItem("移动开发Android Studio开发实战",
"1"));
                lstBooks.Items.Add(new ListItem("iOS App程式开发实务攻略",
"2"));
                lstBooks.Items.Add(new ListItem("Swift开发iOS App快速入门与实战",
"3"));
                lstBooks.Enabled = true;
                break;
        }

        protected void lstBooks_SelectedIndexChanged(object sender, EventArgs e)
        {
            lblResult.Text = $"你选择的书籍：{ddlCategory.SelectedItem.Text} ->
{lstBooks.SelectedItem.Text}";

            //也可如下:传统写法
            //lblResult.Text = "你选择的书籍：" + ddlCategory.SelectedItem.Text + "-"
            + lstBooks.SelectedItem.Text;

            //也可如下:使用string.Format()
            //lblResult.Text = string.Format("你选择的书籍：{0}-{1}",
ddlCategory.SelectedItem.Text, lstBooks.SelectedItem.Text);
        }
    }
}

```




4. 按钮选择控件介绍及实作

LearnASP.NET-rev1.1 <http://www.coding24.org>



RadioButtonList/RadioButton控件

- **RadioButtonList**: 用于多种选择中只能选择一项的情况

| 属性、方法和事件 | 说明 |
|------------------------|-------------------------------|
| Items属性 | 设置/取得RadioButtonList中的项目 |
| SelectedItem属性 | 取得所选取的项目 |
| SelectedIndex属性 | 取得所选取的项目索引 |
| RepeatDirection属性 | Horizontal水平排列 / Vertical垂直排列 |
| SelectedIndexChanged事件 | 当改变文本框中内容且焦点离开文本框后被触发 |

- RadioButton提供单选项，将GroupName属性为同一名称可形成一组。
- RadioButton控件是否选中使用Checked属性，而获取RadioButtonList控件的选中项使用SelectedItem属性



CheckBoxList/CheckBox控件

- CheckBoxList: 复选框组, 用于可多选的情况。

| 属性、方法和事件 | 说明 |
|------------------------|-------------------------------|
| Items属性 | 设置/取得CheckBoxList中的项目 |
| SelectedItem属性 | 取得所选取的项目 |
| SelectedIndex属性 | 取得所选取的项目索引 |
| RepeatDirection属性 | Horizontal水平排列 / Vertical垂直排列 |
| SelectedIndexChanged事件 | 当改变文本框中内容且焦点离开文本框后被触发 |

- CheckBox提供单选项, 将GroupName属性为同一名称可形成一组。
- 判断CheckBox控件是否选中的属性是Checked, 而CheckBoxList控件是集合控件, 判断项目是否选中的属性是成员的Selected属性。

按钮选择控件实例

2. 控件输入资料:

RadioButtonList 任务

选择数据源: 成员(M): 男 属性(P): 性别

编辑项: 0 男 1 女

启用 AutoPostBack: ☐

CheckBoxList 任务

选择数据源: 成员(M): C# 属性(P): 专长的编程语言

编辑项: 0 C# 1 Java 2 Python 3 Swift 4 GoLang

启用 AutoPostBack: ☐

1. 页面设计

请选择你的基本信息:

1. 性别: ☒ 男 ☐ 女

2. 专长的编程语言 (可复选): ☐ C# ☒ Java ☒ Python ☒ Swift ☐ GoLang

确定

3. 输入主要代码

```
btnSubmit_Click() {
    StringBuilder sb = new StringBuilder();
    sb.Append("选择如下-->性别: ");
    sb.Append(rblGender.SelectedItem.Text + " / 专长的编程语言: ");
    foreach(ListItem item in cblLanguage.Items) {
        if (item.Selected) {
            sb.Append(item.Text + " ");
        }
    }
    lblShow.Text = sb.ToString();
}
```

Source: Examples/Unit03/ButtonSelectControls.aspx

4. 运行结果

请选择你的基本信息:

1. 性别: ☒ 男 ☐ 女

2. 专长的编程语言 (可复选): ☐ C# ☒ Java ☒ Python ☒ Swift ☐ GoLang

选择如下-->性别: 男 / 专长的编程语言: Java Python Swift

LearnASP.NET-rev1.1 <http://www.coding24.org> 31

● ButtonSelectControls.aspx

```
<%@ Page Language="C#" AutoEventWireup="true"
CodeBehind="ButtonSelectControls.aspx.cs"
Inherits="LearnASPNET.Examples.Unit03.ButtonSelectControls" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
<title>按钮控件范例</title>
</head>
<body>
<form id="form1" runat="server">
<div>
    请选择你的基本信息: <br />
    -----<br />
    1. 性别: <asp:RadioButtonList ID="rblGender" runat="server"
RepeatDirection="Horizontal">
        <asp:ListItem Selected="True" Value="1">男</asp:ListItem>
        <asp:ListItem Value="2">女</asp:ListItem>
    </asp:RadioButtonList>
    <br />
    2. 专长的编程语言 (可复选): <br />
    <asp:CheckBoxList ID="cblLanguage" runat="server"
RepeatDirection="Horizontal">
        <asp:ListItem Value="0">C#</asp:ListItem>
        <asp:ListItem Value="1">Java</asp:ListItem>
        <asp:ListItem Value="2">Python</asp:ListItem>
        <asp:ListItem Value="3">Swift</asp:ListItem>
        <asp:ListItem Value="4">GoLang</asp:ListItem>
    </asp:CheckBoxList>
    <br />
    <asp:Button ID="btnSubmit" runat="server" Text="确定"
OnClick="btnSubmit_Click" />
    <br />
    <asp:Label ID="lblShow" runat="server" Text="Label"></asp:Label>
</div>
</form>
</body>
</html>
```

● ButtonSelectControls.aspx.cs

```
using System;
using System.Text;
using System.Web.UI.WebControls;

namespace LearnASPNET.Examples.Unit03
{
    public partial class ButtonSelectControls : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {

        }

        protected void btnSubmit_Click(object sender, EventArgs e)
        {
            //放结果字符串
            StringBuilder sb = new StringBuilder();
            sb.Append("选择如下-->性别: ");
            sb.Append(rblGender.SelectedItem.Text + " / 专长的编程语言: ");

            //先取得checkboxList各项
            foreach(ListItem item in cbllLanguage.Items)
            {
                if (item.Selected)
                {
                    sb.Append(item.Text + " ");
                }
            }
            //显示结果
            lblShow.Text = sb.ToString();
        }
    }
}
```



实作3-按钮选择控件实作 问卷调查

Source: Labs/Unit03/SportSurvey.aspx

LearnASP.NET-rev1.1 <http://www.coding24.org>



实作3-按钮选择控件实作问卷调查

- 实作说明：以RadioButtonList/CheckBoxList控件实作问卷调查功能

- 实作步骤：

1. 请使用实作方案 / 项目：LabASP.NET，在Labs/Unit03添加页面。（若无请按单元1实作2创建）
2. 页面添加：添加Web窗体，命名为SportSurvey.aspx，页面设计如下：

用户姓名:

性别: ☒ 男 ☐ 女 RadioButtonList

你喜欢的运动项目为何（可多选）？

☐ 未绑定 CheckBoxList

显示基本信息及兴趣

LearnASP.NET-rev1.1 <http://www.coding24.org>

33



实作3-按钮选择控件实作问卷调查

3. 控件名称: txtName, rblGender, cblSport, btnSubmit, lblShow

4. 控件属性: rblGender -> RepeatDirection: Horizontal,
RepeatLayout: Flow

5. 控件内容代码如下:

CheckBoxList内容:

"足球", "1"
"游泳", "2"
"登山", "3"
"跑步", "4"

透过初值设置

```
protected void Page_Load(object sender, EventArgs e)
{
    // 运动项目初值设定
    if (!IsPostBack)
    {
        cblSport.Items.Add(new ListItem("足球", "1"));
        cblSport.Items.Add(new ListItem("游泳", "2"));
        cblSport.Items.Add(new ListItem("登山", "3"));
        cblSport.Items.Add(new ListItem("跑步", "4"));
    }
}
```




实作3-按钮选择控件实作问卷调查

```
protected void btnSubmit_Click(object sender, EventArgs e)
{
    //遍历复选框列表中所有被运动项目
    string sports = "";
    foreach (ListItem item in cblSport.Items)
    {
        if (item.Selected)
        {
            sports = sports + item.Text + "&nbsp;";
        }
    }
    //显示最终信息
    lblShow.Text = txtName.Text + ", " + rblGender.SelectedItem.Text + ", 兴
趣: " + sports;
}
```

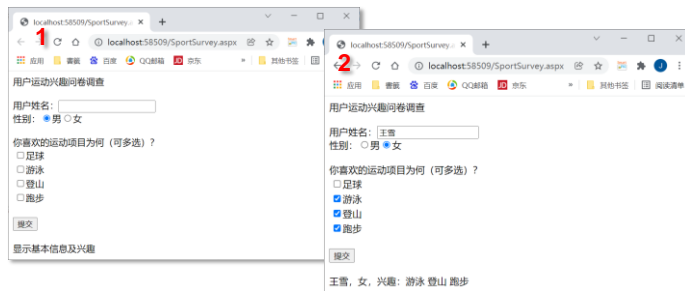
LearnASP.NET-rev1.1 <http://www.coding24.org>

35



实作3-按钮选择控件实作问卷调查

SportSurvey.aspx运行结果:



Source: Labs/Unit03/SportSurvey.aspx

LearnASP.NET-rev1.1 <http://www.coding24.org>

36

● SportSurvey.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="SportSurvey.aspx.cs"
Inherits="LearnASP.NET.Labs.Unit03.SportSurvey" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <title>实作: 问卷调查</title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            用户运动兴趣问卷调查<br />
            <br />
            用户姓名: <asp:TextBox ID="txtName" runat="server"></asp:TextBox>
            <br />
            性别: <asp:RadioButtonList ID="rblGender" runat="server"
RepeatDirection="Horizontal" RepeatLayout="Flow">
                <asp:ListItem Selected="True" Value="M">男</asp:ListItem>
                <asp:ListItem Value="F">女</asp:ListItem>
            </asp:RadioButtonList>
            <br />
            <br />
            你喜欢的运动项目为何 (可多选)? <br />
            <asp:CheckBoxList ID="cblSport" runat="server">
            </asp:CheckBoxList>
            <br />
            <asp:Button ID="btnSubmit" runat="server" OnClick="btnSubmit_Click"
Text="提交" />
            <br />
            <br />
            <asp:Label ID="lblShow" runat="server" Text="显示基本信息及兴趣"
"/></asp:Label>
        </div>
    </form>
</body>
</html>
```

● SportSurvey.aspx.cs

```
using System;
```

```
using System.Web.UI.WebControls;

namespace LearnASPNET.Labs.Unit03
{
    public partial class SportSurvey : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            // 运动项目初值设定
            if (!IsPostBack)
            {
                cblSport.Items.Add(new ListItem("足球", "1"));
                cblSport.Items.Add(new ListItem("游泳", "2"));
                cblSport.Items.Add(new ListItem("登山", "3"));
                cblSport.Items.Add(new ListItem("跑步", "4"));
            }
        }

        protected void btnSubmit_Click(object sender, EventArgs e)
        {
            //遍历复选框列表中所有被运动项目
            string sports = "";
            foreach (ListItem item in cblSport.Items)
            {
                if (item.Selected)
                {
                    sports = sports + item.Text + "&nbsp;";
                }
            }
            //显示最终信息
            lblShow.Text = txtName.Text + ", " + rblGender.SelectedItem.Text + ",  
兴趣: " + sports;
        }
    }
}
```



5. 其它类控件介绍

LearnASP.NET-rev1.1 <http://www.coding24.org>



Calendar控件

- 用途：让用户检视及选取日期。在日历方格中还可以显示行程讯息。

| 属性、方法和事件 | 说明 |
|--------------------|------------------------|
| SelectionMode属性 | 设置选择日期的方式，那些日/周/月份可以选取 |
| SelectedDate属性 | 选定的日期 |
| TodaysDate属性 | 获取或设置今天的日期的值 |
| SelectionChanged事件 | 当用户选取控件上的日/周/月份时被触发 |

//常用范例

//把Calendar控件被选取的日期指定到TextBox中,减少输入错误
txtDate.Text = Calendar1.SelectedDate.ToLongDateString();



Panel控件

- 用途：Panel 控件用作其它控件的容器，常用于显示和隐藏控件组。


| 属性、方法和事件 | 说明 |
|----------------|---------------------|
| GroupingText属性 | 设置Panel 的标题 |
| ScrollBars属性 | 设置Panel 中滚动栏的位置和可见性 |

//范例:将页面分为3区, 依次呈现

```
Page_Load() {  
    if (!IsPostBack) {  
        pnlStep1.Visible = true; //设置pnlStep1为起启页面  
        pnlStep2.Visible = false; //初设pnlStep2不可见  
        pnlStep3.Visible = false; //初设pnlStep3不可见  
    }  
}
```

LearnASP.NET-rev1.1 <http://www.coding24.org>

39



Panel使用范例

第一步：输入基本信息

姓名：

电话：

Panel

下一步

第二步：请选择你的早餐

主食：
☐ 巨无霸汉堡 ☐ 深海鳕鱼堡 ☐ 厚牛培根堡

饮料（热咖啡）：
☐ 不需要 ☒ 非常需要

Panel

下一步

第三步：信息及订单确认

订购人：张三
 电话：18814148311
 主食：巨无霸汉堡
 饮料：非常需要

Panel

订单送出

```
protected void Page_Load(object sender, EventArgs e) {
    if (!IsPostBack) {
        pnlStep1.Visible = true; //设置pnlStep1可见
        pnlStep2.Visible = false; //设置pnlStep2不可见
        pnlStep3.Visible = false; //设置pnlStep3不可见
    }
}
```

Source: Examples/Unit03/Panel.aspx

LearnASP.NET-rev1.1 <http://www.coding24.org>
40

● Panel.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Panel.aspx.cs"
Inherits="LearnASPNET.Examples.Unit03.Panel" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <title>Panel范例</title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <asp:Panel ID="pnlStep1" runat="server">
                第一步：输入基本信息<br />
                -----<br />
                姓名： <asp:TextBox ID="txtName" runat="server"></asp:TextBox><br />
                电话： <asp:TextBox ID="txtPhone" runat="server"></asp:TextBox><br />
                <br />
                <asp:Button ID="btnStep1" runat="server" Text="下一步"
                OnClick="btnStep1_Click"/>
            </asp:Panel>
            <asp:Panel ID="pnlStep2" runat="server">
                第二步：请选择你的早餐<br />
                -----<br />
                主食：<asp:RadioButtonList ID="rblBreakfirst" runat="server"
                RepeatDirection="Horizontal">
                    <asp:ListItem Value="0">巨无霸汉堡</asp:ListItem>
                    <asp:ListItem Value="1">深海鳕鱼堡</asp:ListItem>
                    <asp:ListItem Value="2">厚牛培根堡</asp:ListItem>
                </asp:RadioButtonList>
                <br />
                饮料（热咖啡）： <asp:RadioButtonList ID="rblCoffee" runat="server"
                RepeatDirection="Horizontal">
                    <asp:ListItem Value="0">不需要</asp:ListItem>
                    <asp:ListItem Value="1">非常需要</asp:ListItem>
                </asp:RadioButtonList>
                <br />
                <asp:Button ID="btnStep2" runat="server" Text="下一步"
                OnClick="btnStep2_Click"/>
            </asp:Panel>
        </div>
    </form>
</body>
</html>
```

```

        <asp:Panel ID="pnlStep3" runat="server">
            第三步: 信息及订单确认<br />
            -----<br />
            <asp:Label ID="lblMsg" runat="server"></asp:Label><br />
            <br />
            <asp:Button ID="btnSubmit" runat="server" Text="订单送出"
OnClick="btnSubmit_Click"/>
        </asp:Panel>
    </div>
</form>
</body>
</html>

```

● Panel.aspx.cs

```

using System;

namespace LearnASPNET.Examples.Unit03
{
    public partial class Panel : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            if (!IsPostBack)
            {
                pnlStep1.Visible = true; //设置pnlStep1可见
                pnlStep2.Visible = false; //设置pnlStep2不可见
                pnlStep3.Visible = false; //设置pnlStep3不可见
            }
        }

        protected void btnStep1_Click(object sender, EventArgs e)
        {
            pnlStep1.Visible = false;
            pnlStep2.Visible = true;
            pnlStep3.Visible = false;
        }

        protected void btnStep2_Click(object sender, EventArgs e)
        {
            pnlStep1.Visible = false;
            pnlStep2.Visible = false;
            pnlStep3.Visible = true;
            lblMsg.Text = "订购人: " + txtName.Text + "<br />电话: "
                + txtPhone.Text
                + "<br />主食: " + rblBreakfirst.SelectedItem.Text
                + "<br />饮料: " + rblCoffee.SelectedItem.Text;
        }

        protected void btnSubmit_Click(object sender, EventArgs e)
        {
        }
    }
}

```




单元小结

- 本单元介绍了ASP.NET Web服务器控件中的常用标准控件，包含了基本的Label / TextBox / Button，列表选择类的DropDownList / ListBox，按钮选择RadioButtonList / CheckBoxList及其它等。
- 完成了3个实作练习，透过这些练习应更能掌握前述各项服务器控件的使用方法。