

2

常用标准控件

【学习目标】

通过本章知识的学习,读者首先对服务器控件有些初步了解;掌握 TextBox、Label、Button、DropDownList 等常用控件的使用方法,以及利用表格进行网页页面布局的方法技巧,并利用本章知识设计和实现用户注册页面。通过本章内容的学习,读者可以达到以下学习目的:

了解服务器控件基础知识。

掌握文本控件 (Label、TextBox 控件) 的使用方法。

掌握选择控件 (RadioButtonList、CheckBoxList、DropDownList 控件等) 的使用方法。

掌握按钮控件 (Button、LinkButton、ImageButton 控件等) 的使用方法。

掌握利用表格进行页面布局的方法。

掌握网页设计中容器控件的使用方法。

2.1 情景分析

在动态 Web 应用程序中,经常会遇到用户与服务器进行交互的情况。例如,用户通过页面填写个人信息提交给服务器,服务器将用户信息收集、保存,然后根据实际情况做出相应的处理操作。

企业网站为了给网站会员提供有针对性的服务,要求建立会员注册、登录和会员管理等页面。首先,要求用户通过页面填写个人信息注册会员;其次,将会员信息提交给服务器保存;然后,用户再次访问网站时就可以输入会员账号和密码进行登录。本章的会员注册仅完成会员

注册信息填写（如图 2-1 所示）和信息显示功能（如图 2-2 所示）。用户信息并未保存到数据库，有关数据库操作内容将在后面章节进行介绍。

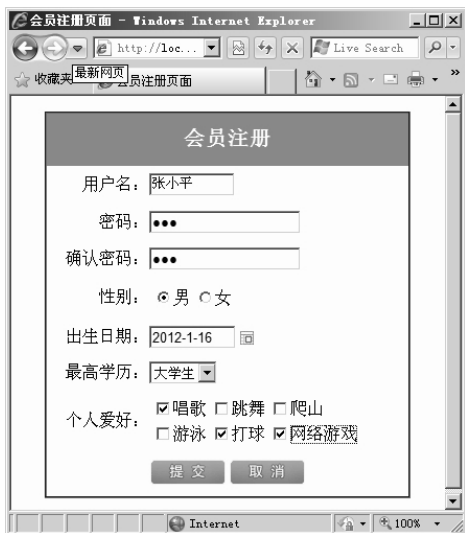


图 2-1 会员注册页面

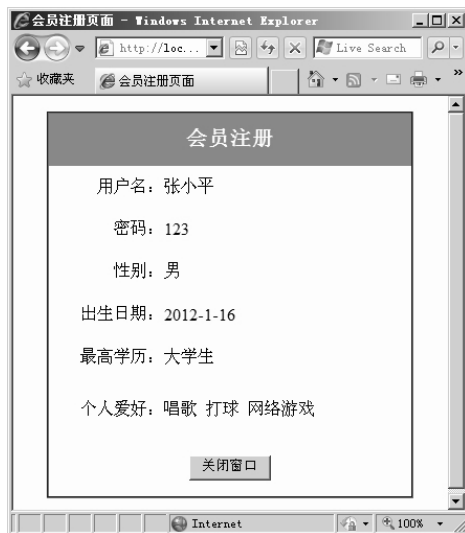


图 2-2 会员信息显示

2.2 服务器控件概述

ASP.NET 服务器控件是运行在服务器端，并且封装了用户界面和其他功能的组件。控件的含义表明它不仅是具有呈现外观作用的元素，而且是一种对象，一种定义 Web 应用程序用户界面的组件。VS 2008 提供了多种类型的服务器控件，如 Web 标准服务器控件、HTML 服务器控件、验证控件和数据控件等。

1. 服务器控件的属性和事件

服务器控件的属性是指控件中具有与用户界面特征相关的或与运行状态有关的字段。大部分服务器控件的属性可分为五类：布局、数据、外观、行为和杂项。布局类属性与页面控件元素的设置有关，如控件尺寸大小等；数据类属性包括与数据绑定相关的属性，如 `DataSource` 等；外观类属性包括背景色、字体格式等；行为类属性与控件运行时相关，如 `Enable` 和 `Visible` 等；杂项表示除此之外的其他属性。

事件是指程序得以运行的触发器（如 `Button` 控件的 `Click` 事件等），当用户与 Web 页面进行交互时被触发，并通过执行事件程序做出相应的响应。与传统客户端窗体中的事件或基于客户端的 Web 应用程序中的事件相比，由服务器控件引发事件的工作方式稍有不同。前者在客户端引发和处理事件；后者将与服务器控件关联的事件在客户端引发，由 ASP.NET 页面框架在 Web 服务器上处理。对于在客户端引发的事件，Web 窗体控件事件模型要求在客户端捕获

事件信息，并且通过 HTTP 发送将事件消息传输到服务器。页面框架解释该发送以确定所发生的事件，然后在要处理该事件的服务器上调用代码中的相应方法。

2. 服务器控件的特点

服务器控件具有如下特点：

公共对象模型。服务器控件是基于公共对象模型的，因此它们可以相互共享大量属性，这是软件复用思想的体现。例如，Label 控件和 Button 控件都有设置背景颜色的属性，它们都使用同一属性——BackColor。

保存视图状态。传统的 HTML 元素是无视图状态的。当页面在服务器端和客户端之间来回传送时，服务器控件会自动保存视图状态。

数据绑定模型。ASP.NET 服务器控件为使用多种数据源提供了方便，可以快速实现数据绑定和访问，大大简化了动态网页的创建过程。

用户定制。服务器控件为网页开发者提供了多种机制来定制自己的页面，例如，可以通过设置服务器控件的 CSS 属性来设置其外观。

配置文件。服务器控件在 Web 应用程序级别上，可通过 web.config 文件对程序进行配置，这使得开发人员可对程序的行为进行统一的控制或改变，而无须对应用程序本身进行重新编译或修改。

创建浏览器特定的 HTML。当浏览器申请某个页面时，服务器控件会确定浏览器的类型，然后生成适合该浏览器显示的 HTML 代码。

2.3 常用服务器控件

2.3.1 文本控件

1. Label 控件

Label 控件用于在页面上显示文本信息，它不但支持静态文本显示，而且还支持用户以编程方式动态显示文本。Label 控件常用的属性有 ID、Text 和 Font 属性等。其中，ID 表示控件标识，Text 表示控件显示的文本内容，Font 表示字体格式设置，如大小、颜色等。

【例 2-1】利用 Label 控件动态显示改变文本内容与显示格式（Ex2-1.aspx）。

（1）右击“解决方案资源管理器”中的 trying 网站，选择快捷菜单中的“添加新项”命令，打开“添加新项”对话框，“语言”项选择 Visual C#，并取消选择“将代码放在单独的文件中”复选项，具体设置如图 2-3 所示。

（2）在页面设计视图下输入“注意查看显示效果：”，并添加 1 个 Label 控件和 1 个 Button 控件。设置 Label 控件 ID 属性值为“lblmess”，Text 属性值为“Label 控件内容”；设置 Button 控件的 Text 属性值为“改变内容”，如图 2-4 所示。

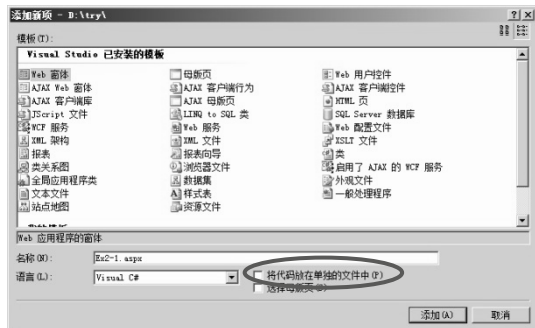


图 2-3 “添加新项”对话框

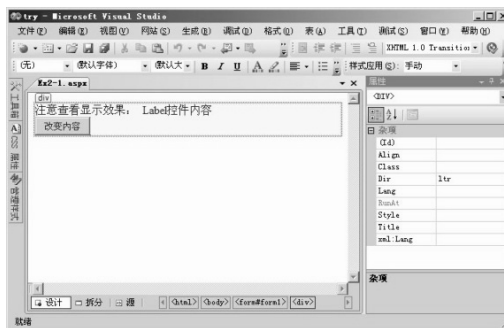


图 2-4 页面设计视图

(3) 双击“改变内容”按钮，界面切换到源视图，在 Button1_Click 事件内输入以下代码：

```
protected void Button1_Click(object sender, EventArgs e)
{
    lblmes.Text = DateTime.Now.ToString();
    lblmes.ForeColor = System.Drawing.Color.Red;
    lblmes.Font.Bold = true;
}
```

(4) 保存文件，页面源视图效果如图 2-5 所示。按 F5 键启动调试，运行结果如图 2-6 所示，单击“改变内容”按钮后，“Label 控件内容”变成了红色、加粗的当前时间。

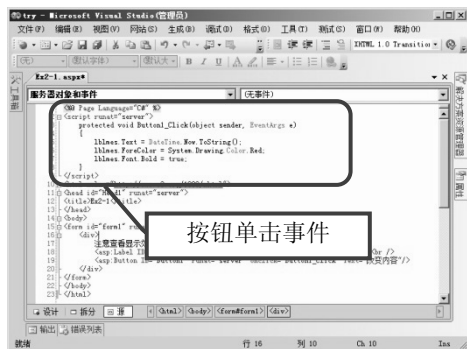


图 2-5 页面源视图



图 2-6 运行结果

由于在“添加新项”对话框中未选择“将代码放在单独的文件中”复选项，该案例采用了单文件模式创建网页，即网页的 HTML 标记和事件代码内容一起保存在以.aspx 为扩展名的文件里。

Ex2-1.aspx 源代码如下：

```
<%@ Page Language="C#" %>
<script runat="server">
    protected void Button1_Click(object sender, EventArgs e)
    {
```

```
        lblmes.Text = DateTime.Now.ToString();
        lblmes.ForeColor = System.Drawing.Color.Red;
        lblmes.Font.Bold = true;
    }
</script>
<html xmlns="http://www.w3.org/1999/xhtml">
<head id="Head1" runat="server">
<title>Ex2-1</title>
</head>
<body>
<form id="form1" runat="server">
    <div>
        注意查看显示效果:
        <asp:Label ID="lblmes" runat="server" Text="Label 控件内容"></asp:Label><br />
        <asp:Button ID="Button1" runat="server" onclick="Button1_Click" Text="改变内容"/>
    </div>
</form>
</body>
</html>
```

程序说明:

“`lblmes.Text = DateTime.Now.ToString();`”语句中, `DateTime.Now` 属性的作用是获取一个 `DateTime` 对象, 并将其设置为系统当前日期和时间。`ToString()` 方法的作用是返回当前对象的字符串。

“`lblmes.ForeColor = System.Drawing.Color.Red;`”语句中, `lblmes.ForeColor` 指的是 `lblmes` 控件的 `ForeColor` 属性, `System.Drawing.Color.Red` 表示 `System.Drawing` 命名空间的颜色值。

“`lblmes.Font.Bold = true;`”表示 `lblmes` 控件的 `Font` 字体 `Bold` 属性值为 `true`, 即控件文本字体以加粗格式显示。

提示: 通常情况下, 页面上显示的静态文本是使用 HTML 标签或静态文字显示, 而不使用 Label 控件, 因为 Label 控件作为服务器控件会占用一定的服务器资源。

尽管 VS 2008 提供了单文件模式创建页面的功能, 但从代码分离的设计思想出发, 建议读者选择“将代码放在单独的文件中”复选项。

2. TextBox 控件

TextBox 控件又称文本框控件, 是用于输入任何类型的文本、数字或其他字符的文本区域。同时, TextBox 控件也可以设置为只读控件, 用于文本显示。

TextBox 控件的常用属性及说明如表 2-1 所示。

表 2-1 TextBox 控件的常用属性及说明

属性	说明
ID	控件唯一标识
Text	控件要显示的文本

续表

属性	说明
TextMode	控件的输入模式，有 SingleLine（单行）、MultiLine（多行）和 Password（密码）三种，默认为 SingleLine
Width	控件的宽度
MaxLength	控件可接收的最大字符数
AutoPostBack	控件内容修改后，是否自动回发到服务器。常和控件的 TextChanged 事件配合使用
Visible	控件是否可见
Enabled	控件是否可用
Rows	控件中文本显示的行数，该属性在 TextMode 为 MultiLine 时有效

【例 2-2】利用 TextBox 控件制作用户登录页面，如图 2-7 所示。用户名最大支持 3 个字符，光标移出姓名文本框时，屏幕上出现动态提示文字；密码输入时以黑点显示（Ex2-2.aspx）。

（1）在页面上输入“姓名：”，并添加 1 个 TextBox 控件。分别设置 TextBox 控件的 ID 属性为 txtname，Width 属性为 80px，MaxLength 属性为 3，AutoPostBack 属性为 True。

（2）单击属性窗口上端的事件图标，将属性窗口切换到事件窗口，双击 TextChanged 事件或者双击 txtname 控件，打开 txtname_TextChanged 事件编辑窗口，输入以下代码：

```
protected void txtname_TextChanged(object sender, EventArgs e)
{
    Response.Write("你的姓名是：" + txtname.Text);
}
```

（3）在页面上输入“密码：”，并添加 1 个 TextBox 控件。设置 TextBox 控件 TextMode 属性为 Password。

（4）在页面上添加 1 个 Button 控件，设置 Text 为“登录”，保存网页。按 F5 键启动调试，运行结果如图 2-7 所示。当姓名内容发生更新时，页面上方的提示内容会随之更新。



图 2-7 TextBox 控件用法

Ex2-2.aspx 文件代码如下:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Ex2-2.aspx.cs" Inherits="Ex2_2" %>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Ex2-2</title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            姓名: <asp:TextBox ID="txtname" runat="server" AutoPostBack="True" Width="80px" MaxLength="3" ontextchanged
            ="txtname_TextChanged" />
            <br />
            密码: <asp:TextBox ID="TextBox2" runat="server" TextMode="Password" />
            <br />
            <asp:Button ID="Button1" runat="server" Text="登录" />
        </div>
    </form>
</body>
</html>
```

Ex2-2.aspx.cs 文件主要代码如下:

```
protected void txtname_TextChanged(object sender, EventArgs e)
{
    Response.Write("你的姓名是: " + txtname.Text);
}
```

程序说明:

修改 TextBox 控件内容后, 将引发控件的 OnTextChanged 事件。但需要注意的是, TextBox 控件的 AutoPostBack 属性必须设置为 True。

Response.Write 的作用是向客户端浏览器输出字符串。其中, “+” 是字符串连接运算符, 起到连接字符串的作用, 如 “中国” + “你好” 等价于 “中国你好”。

2.3.2 选择控件

1. RadioButton 控件

RadioButton 控件是单选按钮控件, 当用户选择某个单选按钮时, 同组中的其他选项不能被同时选中。RadioButton 控件的常用属性及说明如表 2-2 所示。

表 2-2 RadioButton 控件的常用属性及说明

属性	说明
ID	控件唯一标识
Text	控件关联的文本标签
GroupName	控件所属的控件组名

续表

属性	说明
Checked	控件是否被选中
AutoPostBack	单击控件时是否自动回发到服务器
Enabled	判断控件是否可用

【例 2-3】利用 RadioButton 控件实现考试系统中单选题的操作,如图 2-8 所示。当用户不选择答案单击“提交”按钮时,页面弹出“请选择答案!”提示;当用户选择正确答案 B 时,页面提示“恭喜你,回答正确!”;否则提示“对不起,正确答案是 B!”(Ex2-3.aspx)。

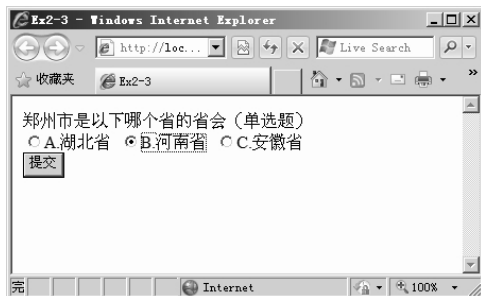


图 2-8 RadioButton 控件用法

Ex2-3.aspx 文件代码如下:

```
<%@ Page Language="C#" CodeFile="Ex2-3.aspx.cs" Inherits="Ex2_3" %>
<html>
<body>
    <form id="form1" runat="server">
        郑州市是以下哪个省的省会 (单选题) <br />
        <asp:RadioButton ID="R1" runat="server" GroupName="sel1" Text="A.湖北省" />
        <asp:RadioButton ID="R2" runat="server" GroupName="sel1" Text="B.河南省" />
        <asp:RadioButton ID="R3" runat="server" GroupName="sel1" Text="C.安徽省" /> <br />
        <asp:Button ID="Button1" runat="server" onclick="Button1_Click" Text="提交" />
    </form>
</body>
</html>
```

Ex2-3.aspx.cs 文件的主要代码如下:

```
protected void Button1_Click(object sender, EventArgs e)
{
    //如果没有选择答案
    if (R1.Checked == false && R2.Checked == false && R3.Checked == false)
        Response.Write("<script>alert('请选择答案!')</script>");
    else
    {
        //正确答案是 B
        Response.Write("<script>alert('恭喜你,回答正确!')</script>");
    }
}
```



```
//判断所选答案是否正确
if (R2.Checked == true)
    Response.Write("<script>alert('恭喜你，回答正确！')</script>");
else
    Response.Write("<script>alert('对不起，正确答案是 B！')</script>");
}
```

程序说明：

3 个 RadioButton 控件的 GroupName 属性值都是 sel1，只有 GroupName 属性值相同才能保证这些控件成为一组。

在后台代码文件中，以“//”标记开头表示该行为注释行，该标记后面的语句不被执行。

“if (R2.Checked == true)”语句中的 R2.Checked 的值为 true 时，表示 RadioButton 控件 R2 被选中；值为 false 时，表示未被选中。

“if (R1.Checked == false && R2.Checked == false && R3.Checked == false)”语句中，“&&”符号在程序中是“逻辑并”运算符。如有逻辑表示式 A、B，当且仅当 A、B 值都为 true 时，A&&B 值为 true。相对地，运算符“||”表示“逻辑或”运算。

2. RadioButtonList 控件

由于每个 RadioButton 控件在 RadioButton 组中是相互独立的，若判断同组中的多个 RadioButton 控件是否被选中，需要判断所有 RadioButton 控件的 Checked 属性，效率较低。RadioButtonList 控件有效地解决了这个问题，它为读者提供了一组 RadioButton，大大方便了用户操作。RadioButtonList 控件的常用属性及说明如表 2-3 所示。

表 2-3 RadioButtonList 控件的常用属性及说明

属性	说明
ID	控件唯一标识
AutoPostBack	单击控件时是否自动回发到服务器，响应 OnSelectedIndexChanged 事件
CellPadding	各项目之间的距离，单位是像素
Items	返回 RadioButtonList 控件中的 ListItem 的对象
RepeatDirection	选择项目的排列方向，默认为 Vertical
RepeatColumns	设置一行旋转选择项目的个数，默认为 0（表示忽略该项）
SelectedItem	返回被选择的 ListItem 对象
TextAlign	设置各项目所显示文字在按钮左边还是右边，默认为 Right

【例 2-4】 利用 RadioButtonList 控件的 AutoPostBack 属性和 OnSelectedIndexChanged 事件实现性别单选和提示文字即时更新，如图 2-9 所示。如用户选择“男”，页面下面的文字立即变为“你选择的是：男；对应的值为：1”（Ex2-4.aspx）。



图 2-9 RadioButtonList 控件用法

Ex2-4.aspx 文件代码如下:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Ex2-4.aspx.cs" Inherits="Ex2_4" %>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Ex2-4</title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <asp:RadioButtonList ID="rblsex" runat="server" RepeatDirection="Horizontal" onselectedindexchanged
            ="rblsex_SelectedIndexChanged" AutoPostBack="True">
                <asp:ListItem Value="1">男</asp:ListItem>
                <asp:ListItem Value="0">女</asp:ListItem>
            </asp:RadioButtonList>
            你选择的是: <asp:Label ID="lbl1" runat="server" Text="Label"></asp:Label>
            ; 对应的值为: <asp:Label ID="lbl2" runat="server" Text="Label"></asp:Label>
        </div>
    </form>
</body>
</html>
```

Ex2-4.aspx.cs 文件主要代码如下:

```
protected void rblsex_SelectedIndexChanged(object sender, EventArgs e)
{
    lbl1.Text = rblsex.SelectedItem.Text;
    lbl2.Text = rblsex.SelectedItem.Value;
}
```

程序说明:

RadioButtonList 控件的候选项由 ListItem 控件实现。如<asp:ListItem Text="男" Value="1" />或者<asp:ListItem Value="1">男</asp:ListItem>, Text 值有这两种写法, 其他控件类似。

“lbl1.Text = rblsex.SelectedItem.Text;”语句中的 rblsex.SelectedItem.Text 表示 rblsex 控件当前被选择项的 Text 值, 即 ListItem 控件的 Text 值。而 rblsex.SelectedItem.Value 表示 rblsex 控件当前被选择项的 Value 值, 即 ListItem 控件的 Value 值。

若使控件的 `OnSelectedIndexChanged` 事件（或者 `Selectedindexchanged` 事件）即时生效，必须设置控件的 `AutoPostBack` 属性为 `True`。

提示：RadioButtonList 控件所涉及的 ListItem 选择项，读者可以在选中控件时，单击控件右上方的  按钮，选择“RadioButtonList 任务”中的“编辑项...”命令，通过图形界面来实现选择项的添加和编辑。

3. CheckBox 控件

CheckBox 控件用来表示是否选取了某个选项，常用于具有是/否、真/假选项的选取。CheckBox 控件和 RadioButton 控件的区别在于前者允许多选。由于 CheckBox 控件和 RadioButton 控件的常用属性大体相同。

【例 2-5】利用 CheckBox 控件实现考试系统中不定项选择题的操作，如图 2-10 所示。当用户选择 ABC 三个答案时，提示“回答正确”；否则提示“错误，正确答案为 ABC”（Ex2-5.aspx）。

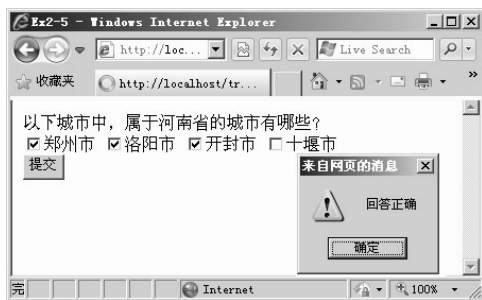


图 2-10 CheckBox 控件用法

Ex2-5.aspx 文件代码如下：

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Ex2-5.aspx.cs" Inherits="Ex2_5" %>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Ex2-5</title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            以下城市中，属于河南省的城市有哪些? <br />
            <asp:CheckBox ID="ckb1" runat="server" Text="郑州市" />
            <asp:CheckBox ID="ckb2" runat="server" Text="洛阳市" />
            <asp:CheckBox ID="ckb3" runat="server" Text="开封市" />
            <asp:CheckBox ID="ckb4" runat="server" Text="十堰市" /><br />
            <asp:Button ID="Button1" runat="server" Text="提交" onclick="Button1_Click" />
        </div>
    </form>
</body>
</html>
```

Ex2-5.aspx.cs 文件主要代码如下:

```
protected void Button1_Click(object sender, EventArgs e)
{
    if (!ckb1.Checked && !ckb2.Checked && !ckb3.Checked && !ckb4.Checked)
        Response.Write("<script>alert('请选择答案!')</script>");
    else
    {
        if (ckb1.Checked && ckb2.Checked && ckb3.Checked)
            Response.Write("<script>alert('回答正确')</script>");
        else
            Response.Write("<script>alert('错误, 正确答案为 ABC')</script>");
    }
}
```

程序说明:

if(ckb1.Checked)和 if(ckb1.Checked==true)的作用相同, 都表示 ckb1 控件被选中。而 if(!ckb1.Checked)和 if(ckb1.Checked==false)的作用相同, 都表示 ckb1 控件未被选中。

4. CheckBoxList 控件

用 CheckBox 控件可以实现多选功能, 但在判断被选中的选项时, 需要对每一个对象都进行判断。CheckBoxList 控件和 RadioButtonList 控件类似, 可以方便地判断用户选中的选项。

【例 2-6】利用 CheckBoxList 控件实现用户个人爱好选择功能页面, 如图 2-11 所示。根据用户所选择的爱好不同, 页面下方出现的选择结果动态变化 (Ex2-6.aspx)。

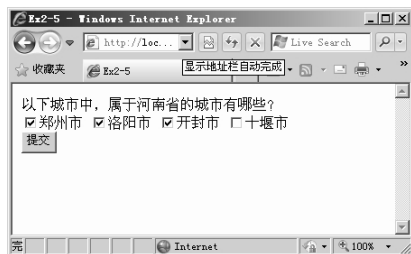


图 2-11 CheckBoxList 控件用法

Ex2-6.aspx 文件主要代码如下:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Ex2-6.aspx.cs" Inherits="Ex2_6" %>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Ex2-6</title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            请选择你的爱好:
            <asp:CheckBoxList ID="ckl1" runat="server" RepeatColumns="3" RepeatDirection="Horizontal">
                <asp:ListItem>唱歌</asp:ListItem>
                <asp:ListItem>跳舞</asp:ListItem>
            </asp:CheckBoxList>
        </div>
    </form>
</body>
</html>
```

```
<asp:ListItem>游泳</asp:ListItem>
<asp:ListItem>爬山</asp:ListItem>
<asp:ListItem>旅行</asp:ListItem>
<asp:ListItem>钓鱼</asp:ListItem>
</asp:CheckBoxList>
<asp:Button ID="Button1" runat="server" Text="提交" onclick="Button1_Click" />
<asp:Label ID="lblmes" runat="server"/>
</div>
</form>
</body>
</html>
```

Ex2-6.aspx.cs 文件主要代码如下:

```
protected void Button1_Click(object sender, EventArgs e)
{
    string result = "你选择的是: ";
    for (int i = 0; i < ck11.Items.Count; i++)
    {
        if (ck11.Items[i].Selected)
            result += ck11.Items[i].Text + "&nbsp;&nbsp;&nbsp;";
    }
    lblmes.Text = result;
}
```

程序说明:

“for (int i = 0; i < ck11.Items.Count; i++)”语句中, ck11.Items.Count 表示 CheckBoxList 控件选项的数目, 利用 for 循环可以方便地实现选项是否被选择的检查功能。

“if (ck11.Items[i].Selected)”语句中, ck11.Items[i].Selected 表示 CheckBoxList 控件的第 i 个选项被选中。需要注意的是, CheckBoxList 选项是用 Selected 值来判断的, 而非 Checked 属性。

5. DropDownList 控件

DropDownList 控件是一个下拉式列表控件, 功能和 RadioButtonList 控件类似, 用户可以从下拉列表框中选择单一选项。DropDownList 控件适合具有较多选项的情况, 能够使程序界面更加紧凑。

【例 2-7】利用 DropDownList 控件实现用户出生地选择功能, 如图 2-12 所示。根据用户所选择的出生地不同, 单击“提交”按钮后提示不同的信息 (Ex2-7.aspx)。



图 2-12 DropDownList 控件用法

Ex2-7.aspx 文件代码如下:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Ex2-7.aspx.cs" Inherits="Ex2_7" %>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Ex2-7</title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            出生地:
            <asp:DropDownList ID="ddlbir" runat="server" Width="86px">
                <asp:ListItem Selected="True" Value="1">北京市</asp:ListItem>
                <asp:ListItem Value="2">上海市</asp:ListItem>
                <asp:ListItem Value="3">重庆市</asp:ListItem>
                <asp:ListItem Value="4">天津市</asp:ListItem>
            </asp:DropDownList>
            <br />
            <asp:Button ID="Button1" runat="server" onclick="Button1_Click" Text="提交" />
            <asp:Label ID="lblmes" runat="server" Text="你的出生地是: " />
        </div>
    </form>
</body>
</html>
```

Ex2-7.aspx.cs 文件主要代码如下:

```
protected void Button1_Click(object sender, EventArgs e)
{
    lblmes.Text += ddlbir.SelectedItem.Text;
}
```

程序说明:

代码“<asp:ListItem Selected="True" Value="1">”中, Selected="True"的作用是将该选项作为默认选中项。

“lblmes.Text += ddlbir.SelectedItem.Text;”语句中, ddlbir.SelectedItem.Text 获取的是选项的 Text 值,即显示出来的文本(如北京市、上海市等),而 ddlbir.SelectedValue 表示选项的 Value 值(如北京市对应为 1、上海市对应为 2)。

提示: 代码“lblmes.Text += ddlbir.SelectedItem.Text”中出现了“+=”运算符,作用相当于 lblmes.Text = lblmes.Text + ddlbir.SelectedItem.Text。例如, string str = "你是"; str += "大学生"; 而最终 str 变量值为“你是大学生”。

6. Calendar 控件

Calendar 控件是日历控件,用于选择日期。可以结合 TextBox 控件一起使用,实现日期输入,从而用规范并简化的日期格式输入。

【例 2-8】 利用 Calendar 控件实现用户入团日期输入功能,如图 2-13 所示。用户通过选择 Calendar 控件的日期,使其自动出现在 TextBox 控件文本框中 (Ex2-8.aspx)。



图 2-13 Calendar 控件用法

Ex2-8.aspx 文件代码如下：

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Ex2-8.aspx.cs" Inherits="Ex2_8" %>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
  <title>Ex2-8</title>
</head>
<body>
  <form id="form1" runat="server">
    <div>
      你的入团时间是:
      <asp:TextBox ID="txtdate" runat="server" Enabled="False"/>
      <asp:Calendar ID="Calendar1" runat="server" onselectionchanged="Calendar1_SelectionChanged">
      </asp:Calendar>
    </div>
  </form>
</body>
</html>
```

Ex2-8.aspx.cs 文件主要代码如下：

```
protected void Calendar1_SelectionChanged(object sender, EventArgs e)
{
    txtdate.Text = Calendar1.SelectedDate.ToShortDateString();
}
```

程序说明：

代码“<asp:TextBox ID="txtdate" runat="server" Enabled="False"/>”中，Enabled="False"表示控件不可手工编辑。

使用 Calendar 控件时，用户可以通过控件右上方的  展开“Calendar 任务”，通过设置“自动套用格式...”来设置控件外观。

提示：由于 Calendar 控件占用页面较多，从页面美观出发，读者可以在 TextBox 控件后添加一个按钮，用于控制 Calendar 控件显示“Calendar1.Visible = true;”。

2.3.3 按钮控件

1. Button 控件

Button 控件是读者使用频率最高的控件之一，用户通过单击 Button 来执行该控件的 Click

事件。Button 控件的常用属性有 Id、Text、PostBackUrl 及 OnClick 事件。其中，PostBackUrl 属性用于设置单击控件所发送的 URL 地址。

下面接着例 2-8 来继续操作，添加 1 个 Button 控件控制 Calendar 控件的显示。

【例 2-9】利用 Button 控件控制 Calendar 控件的显示，效果如图 2-14 所示。单击“显示日历”按钮，出现日历；用户选择日期后，日期出现在文本框中，日历窗口关闭（Ex2-9.aspx）。



图 2-14 Button 控件用法

Ex2-9.aspx 文件主要代码如下：

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Ex2-9.aspx.cs" Inherits="Ex2_8" %>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Ex2-8</title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            你的入团时间是: <asp:TextBox ID="txtdate" runat="server" Enabled="False"/>
            <asp:Button ID="Button1" runat="server" onclick="Button1_Click" Text="显示日历" />
            <asp:Calendar ID="Calendar1" runat="server"
                onselectionchanged="Calendar1_SelectionChanged" Height="125px"
                Visible="False">
            </asp:Calendar>
        </div>
    </form>
</body>
</html>
```

Ex2-9.aspx.cs 文件主要代码如下：

```
protected void Calendar1_SelectionChanged(object sender, EventArgs e)
{
    txtdate.Text = Calendar1.SelectedDate.ToShortDateString();
    Calendar1.Visible = false;
}
protected void Button1_Click(object sender, EventArgs e)
{
    Calendar1.Visible = true;
}
```


程序说明:

设置 Calendar1 控件的 Visible 属性为 False, 实现控件在网页加载后不可见。Button1 的 Click 事件利用代码 “Calendar1.Visible= true;” 来显示控件。而 Calendar1_SelectionChanged 实现文本框文本输入, 并再次隐藏日历控件。

2. LinkButton 控件

LinkButton 控件又称超链接按钮控件, 该控件在功能上与 Button 控件相同, 但样式以超链接形式显示。LinkButton 控件有一个 PostBackUrl 属性, 该属性用于设置单击控件时链接到的网址。

【例 2-10】利用 LinkButton 控件 PostBackUrl 属性实现超链接功能, 如图 2-15 所示。用户单击 “打开 Ex2-9” 链接, 页面将转向 Ex2-9.aspx 页面 (Ex2-10.aspx)。



图 2-15 LinkButton 控件用法

Ex2-10.aspx 文件代码如下:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Ex2-10.aspx.cs" Inherits="Ex2_10" %>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Ex2-10</title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            超链接页面<asp:LinkButton ID="LinkButton1" runat="server" PostBackUrl="~/Ex2-9.aspx">打开 Ex2-9
        </asp:LinkButton>
        </div>
    </form>
</body>
</html>
```

程序说明:

PostBackUrl="~/Ex2-9.aspx"表示超链接指向的新页面地址, 地址的 “~/” 是相对路径的同级目录的含义, 即 Ex2-9.aspx 和 Ex2-10.aspx 保存在同一文件夹里。

3. ImageButton 控件

ImageButton 控件是图片按钮控件，用户单击控件上的图片引发控件的 Click 事件。ImageButton 控件有一个 ImageUrl 属性，该属性用于设置按钮上显示的图片位置。其他属性和用法与 Button 控件相同。

【例 2-11】利用 ImageButton 控件达到美化按钮的效果，如图 2-16 所示 (Ex2-11.aspx)。



图 2-16 ImageButton 控件用法

Ex2-11.aspx 文件代码如下：

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Ex2-11.aspx.cs" Inherits="Ex2_11" %>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Ex2-11</title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            你的入团时间是:
            <asp:TextBox ID="txtdate" runat="server" Enabled="False"/>
            <asp:ImageButton ID="ImageButton1" runat="server" ImageUrl="~/images/date.GIF"
                onclick="ImageButton1_Click" />
            <asp:Calendar ID="Calendar1" runat="server" Visible="False">
            </asp:Calendar>
        </div>
    </form>
</body>
</html>
```

Ex2-11.aspx.cs 文件主要代码如下：

```
protected void ImageButton1_Click(object sender, ImageClickEventArgs e)
{
    Calendar1.Visible = true;
}
```

程序说明：

ImageUrl 属性表示 ImageButton 控件的图片位置。地址 “~/images/butimage.gif” 表示与程序文件同级目录内的 Images 文件夹中的 butimage.gif 文件。

4. FileUpload 控件

FileUpload 控件是用于客户端文件上传到服务器的控件。该控件显示 1 个文本框和 1 个浏览按钮，用户可以直接在文件框中输入完整的文件名，也可以通过“浏览”按钮选择文件。FileUpload 控件的常用属性及说明如表 2-4 所示。

表 2-4 FileUpload 控件的常用属性及说明

属性	说明
ID	控件唯一标识
FileContent	获取指定上传文件的 Stream 对象（Stream 数据类型）
FileName	获取上传文件在客户端的文件名称
HasFile	获取一个布尔值，用于表示控件是否已经包含一个文件
PostedFile	获取一个与上传文件相关的 HttpPostedFile 对象，使用该对象可以获取上传文件的相关属性

除了上述常用属性外，FileUpload 控件还有一个 SaveAs 方法，用于将上传的文件保存到服务器。

【例 2-12】利用 FileUpload 控件实现文件上传操作，如图 2-17 所示。用户单击页面上的“浏览...”按钮，选择要上传的文件，单击“上传”按钮，文件将上传到服务器网站的根目录下（Ex2-12.aspx）。

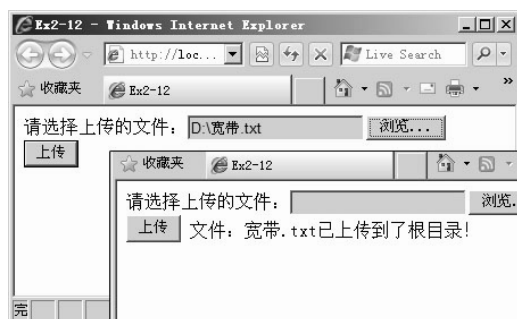


图 2-17 LinkButton 控件用法

Ex2-12.aspx 文件代码如下：

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Ex2-12.aspx.cs" Inherits="Ex2_12" %>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Ex2-12</title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
```

```

        请选择上传的文件: <asp:FileUpload ID="fulfile" runat="server" />
        <br />
        <asp:Button ID="Button1" runat="server" onclick="Button1_Click" Text="上传" />
        <asp:Label ID="lblmes" runat="server" Text=""></asp:Label>
    </div>
</form>
</body>
</html>

```

Ex2-12.aspx.cs 文件主要代码如下:

```

protected void Button1_Click(object sender, EventArgs e)
{
    if (fulfile.HasFile)
    {
        string strname = fulfile.FileName;
        fulfile.SaveAs(Server.MapPath(strname));
        lblmes.Text += "文件: " + strname + "已上传到了根目录! ";
    }
    else
    {
        Response.Write("请选择上传文件! ");
    }
}

```

程序说明:

“if (fulfile.HasFile)”语句中, fulfile.HasFile 表示 fulfile 控件的 HasFile 属性值。该语句与 if (fulfile.HasFile==true)作用相同, 用于判断用户是否选择了上传文件。

“string strname = fulfile.FileName;”语句中, 使用 FileUpload 控件的 FileName 属性获取上传文件的文件名 (包括文件扩展名)。

fulfile.SaveAs(Server.MapPath(strname));语句, 使用 FileUpload 控件的 SaveAs 方法将上传文件保存到服务器; Server.MapPath()方法获取服务器上的物理路径 (即绝对路径)。

“fulfile.SaveAs(Server.MapPath(strname));”整体的作用是将上传文件以原文件名保存到服务器根目录下。

提示: Server.MapPath()方法获取服务器的物理路径。其中, Server.MapPath("/")获取应用程序根目录所在的路径; Server.MapPath("./")获取当前页面所在目录的路径, 等价于 Server.MapPath(""); Server.MapPath("../")获取当前页面所在目录的上级目录路径。

2.3.4 表格控件

网站开发过程中, 表格是页面布局的一种重要手段。使用 Table 表格、tr 表格行和 td 表格单元格进行页面布局, 操作简单、快捷, 大大提高了开发效率。表格常用的属性及说明如表 2-5 所示。

表 2-5 表格控件的常用属性及说明

属性	说明
Boder	表格边框宽度
CellPadding	单元格边框与内容的间距
CellSpacing	单元格间距
Align	表格、单元格水平方向对齐方式，有 Left、Right 和 Center 三种
Valign	单元格竖直方向对齐方式，有 baseline、bottom、middle 和 top 四种
Style	表格、单元格样式

在 VS 2008 环境下，有标准服务器控件和 HTML 控件两种 Table。这里使用的是 HTML 控件类的 Table，或者通过“表格”→“插入表”命令（注：本书无特别声明时，Table 控件均指 HTML 服务器控件类的 Table 控件）。

【例 2-13】利用表格控件实现系统登录页面布局设计，如图 2-18 所示（Ex2-13.aspx）。

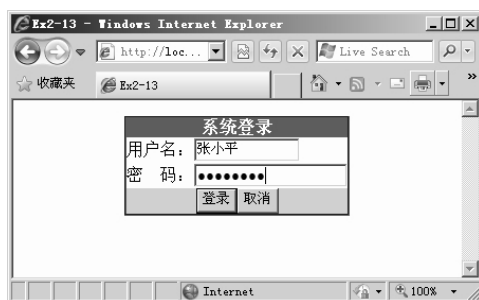


图 2-18 Table 控件用法

Ex2-13.aspx 文件代码如下：

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Ex2-13.aspx.cs" Inherits="Ex2_13" %>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Ex2-13</title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <table align="center" cellpadding="0" cellspacing="0" class="style4" style="border: 2px groove #336699;">
                <tr>
                    <td align="center" class="style4" colspan="2" valign="middle" style="background-color: #336699;
                        font-size: 16px; color: #FFFFFF; font-weight: bold;">系统登录
                </td>
            </tr>
            <tr>
                <td align="right" class="style4">用户名: </td>
```

```
<td align="left" class="style4">
<asp:TextBox ID="TextBox1" runat="server" Width="100px"></asp:TextBox>
</td>
</tr>
<tr>
<td align="right" class="style2">
密 码:
</td>
<td align="left" class="style2">
<asp:TextBox ID="TextBox2" runat="server" TextMode="Password"></asp:TextBox>
</td>
</tr>
<tr>
<td align="center" colspan="2" style="background-color: #CCCCFF" valign="middle">
<asp:Button ID="Button1" runat="server" Text="登录" /><asp:Button ID="Button2" runat="server"
Text="取消" />
</td>
</tr>
</table>
</div>
</form>
</body>
</html>
```

2.4 会员注册页面设计

会员注册页面是网站开发过程中经常用到的，本节将利用本章所学知识进行会员注册页面设计，最终效果如图 2-1 和图 2-2 所示。首先，读者事先准备好 3 张用于 ImageButton 控件的素材图片（分别是“提交”、“取消”和“日历显示”）。

具体操作步骤如下：

（1）新建 Addmember.aspx，在页面添加 Panel 控件，设置 ID 为 Panel1。在 Panel1 控件内添加一个 9 行 2 列的 Table 控件。

（2）将 Table 左边单元格 td 的 align 属性全部设置为 right（即右对齐），右边单元格设置为 left（即左对齐）。将第一行的 2 个单元格合并，输入“会员注册”，并设置单元格格式。

（3）左边单元格自上向下依次输入“用户名：”、“密码：”、“确认密码：”、“性别：”、“出生日期：”、“最高学历：”、“个人爱好：”。

（4）右边单元格自上向下依次添加相应的服务器控件，并设置其属性如表 2-6 所示。

表 2-6 表格控件的常用属性及说明

左侧内容	控件 ID	控件类型	说明
用户名	txtname	TextBox	设置 Width 属性为 82px
密码	txtpwd	TextBox	设置 TextMode 属性为 Password

续表

左侧内容	控件 ID	控件类型	说明
确认密码	txtpwd2	TextBox	设置 TextMode 属性为 Password
性别	rbtnlsex	RadioButtonList	设置 RepeatDirection 属性为 Horizontal, ListItem 为 “男”、“女”
出生日期	Txtdate	TextBox	设置 Enabled 属性为 "False", Width 属性为 83px
	Ibtndate	ImageButton	设置 ImageUrl="~/images/图标.JPG", onclick="ibtndate_Click"
	clddate	Calendar	设置 onselectionchanged="clddate_SelectionChanged", Visible="False"
最高学历	ddlschool	DropDownList	设置 ListItem 为研究生、大学生、中学生、小学生
个人爱好	chblove	CheckBoxList	设置 RepeatDirection="Horizontal", RepeatColumns="3"

(5) 将表格最下面一行两个单元格合并, 添加两个 ImageButton 控件来表示“提交”和“取消”。

(6) 按照上面的操作方法, 在 Table 后面再添加一个 Panel 控件 Panel2。在 Panel2 内添加 8 行 2 列的 Table 控件, 表格第一个单元格和左边 1 列内容与上面表格内容相同; 右边列单元格里, 依次添加 Label 服务器控件, ID 属性依次为 lblname、lblpwd、lblsex、lbldate、lblschool、lbllove 等。

(7) 最下面一行两个单元格合并, 添加一个 Button 控件, 设置 onclick="Button1_Click" 和 Text="关闭窗口"。

(8) 添加 page_load 和其他 Button 按钮单击事件, 详见后台代码。

Addmember.aspx 文件代码如下:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Addmember.aspx.cs" Inherits="Addmember" %>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
<title>会员注册页面</title>
</head>
<body>
<form id="form1" runat="server">
<div>
<asp:Panel ID="Panel1" runat="server">
<table align="center" cellpadding="0" cellspacing="0" style="border: 2px groove #FF3300;width: 350px; height: 370px;">
<tr>
<td align="center" colspan="2" style="font-size: 20px; color: #FFFFFF; font-weight:bold; background-color: #FF6600; height: 50px;">会员注册</td>
</tr>
<tr>
<td align="right">用户名: </td>
<td align="left">
<asp:TextBox ID="txtname" runat="server" Width="82px"/></td>
</tr>
```

```

<tr>
    <td align="right">密码: </td>
    <td align="left">
        <asp:TextBox ID="txtpwd" runat="server" TextMode="Password"/></td>
</tr>
<tr>
    <td align="right">确认密码: </td>
    <td align="left">
        <asp:TextBox ID="txtpwd2" runat="server" TextMode="Password"/></td>
</tr>
<tr>
    <td align="right">性别: </td>
    <td align="left">
        <asp:RadioButtonList ID="rbtlsex" runat="server" RepeatDirection="Horizontal">
            <asp:ListItem>男</asp:ListItem>
            <asp:ListItem>女</asp:ListItem>
        </asp:RadioButtonList>
    </td>
</tr>
<tr>
    <td align="right">出生日期: </td>
    <td align="left">
        <asp:TextBox ID="txtdate" runat="server" Width="83px"/>
        <asp:ImageButton ID="ibtndate" runat="server" ImageUrl="~/images/图标.JPG"
            onclick="ibtndate_Click" />
        <asp:Calendar ID="clddate" runat="server" Visible="False" BackColor="#FFFFCC"
            BorderColor="#FFCC66" BorderWidth="1px" DayNameFormat="Shortest"
            Font-Names="Verdana" Font-Size="8pt" ForeColor="#663399" Height="200px"
            onselectionchanged="clddate_SelectionChanged" ShowGridLines="True"
            Width="220px">
            <SelectedDayStyle BackColor="#CCCCFF" Font-Bold="True" />
            <SelectorStyle BackColor="#FFCC66" />
            <TodayDayStyle BackColor="#FFCC66" ForeColor="White" />
            <OtherMonthDayStyle ForeColor="#CC9966" />
            <NextPrevStyle Font-Size="9pt" ForeColor="#FFFFCC" />
            <DayHeaderStyle BackColor="#FFCC66" Font-Bold="True" Height="1px" />
            <TitleStyle BackColor="#990000" Font-Bold="True" Font-Size="9pt"
                ForeColor="#FFFFCC" />
        </asp:Calendar>
    </td>
</tr>
<tr>
    <td align="right">最高学历: </td>
    <td align="left">
        <asp:DropDownList ID="dlschool" runat="server">
            <asp:ListItem>研究生</asp:ListItem>
            <asp:ListItem Selected="True">大学生</asp:ListItem>
            <asp:ListItem>中学生</asp:ListItem>
        </asp:DropDownList>
    </td>
</tr>

```



```

        <asp:ListItem>小学生</asp:ListItem>
    </asp:DropDownList>
</td>
</tr>
<tr>
    <td align="right">个人爱好: </td>
    <td align="left" style="height: 60px">
        <asp:CheckBoxList ID="chblove" runat="server" RepeatColumns="3" RepeatDirection="Horizontal">
            <asp:ListItem>唱歌</asp:ListItem>
            <asp:ListItem>跳舞</asp:ListItem>
            <asp:ListItem>爬山</asp:ListItem>
            <asp:ListItem>游泳</asp:ListItem>
            <asp:ListItem>打球</asp:ListItem>
            <asp:ListItem>网络游戏</asp:ListItem>
        </asp:CheckBoxList>
    </td>
</tr>
<tr>
    <td align="center" colspan="2">
        <asp:ImageButton ID="btnOk" runat="server" ImageUrl="~/images/btuok.gif"
            onclick="btnOk_Click1" />
        <asp:ImageButton ID="Ibtncancel" runat="server" ImageUrl="~/images/btureset.gif"
            onclick="Ibtncancel_Click" />
    </td>
</tr>
</table>
</asp:Panel>
<asp:Panel ID="Panel2" runat="server">
    <table align="center" cellpadding="0" cellspacing="0" style="border: 2px groove #FF3300;width: 350px; height: 370px;">
        <tr>
            <td align="center" colspan="2" style="font-size: 20px; color: #FFFFFF; font-weight:bold; background-color: #FF6600; height: 50px;">会员注册</td>
        </tr>
        <tr>
            <td align="right" style="width: 110px">用户名: </td>
            <td align="left"><asp:Label ID="lblname" runat="server" Text="Label"/></td>
        </tr>
        <tr>
            <td align="right">密码: </td>
            <td align="left"><asp:Label ID="lblpwd" runat="server" Text="Label"/></td>
        </tr>
        <tr>
            <td align="right">性别: </td>
            <td align="left"><asp:Label ID="lblsex" runat="server" Text="Label"/></td>
        </tr>
        <tr>
            <td align="right">出生日期: </td>

```

```

        <td align="left"><asp:Label ID="lbldate" runat="server" Text="Label"/></td>
    </tr>
    <tr>
        <td align="right">最高学历: </td>
        <td align="left"><asp:Label ID="lblschool" runat="server" Text="Label"/></td>
    </tr>
    <tr>
        <td align="right">个人爱好: </td>
        <td align="left" style="height: 60px"><asp:Label ID="lbllove" runat="server"/></td>
    </tr>
    <tr>
        <td align="center" colspan="2"><asp:Button ID="btnclose" runat="server" onclick="btnclose_Click"
            Text="关闭窗口" /></td>
    </tr>
</table>
</asp:Panel>
</div>
</form>
</body>
</html>

```

Addmember.aspx.cs 文件主要代码如下:

```

protected void Page_Load(object sender, EventArgs e)
{
    Panel2.Visible = false;
    Panel1.Visible = true;
    txtpwd.Attributes["value"] = txtpwd.Text;
    txtpwd2.Attributes["value"] = txtpwd2.Text;
}
protected void ibtndate_Click(object sender, ImageClickEventArgs e)
{
    clddate.Visible = true;
}
protected void clddate_SelectionChanged(object sender, EventArgs e)
{
    txtdate.Text = clddate.SelectedDate.ToShortDateString();
    clddate.Visible = false;
}
protected void btnOk_Click(object sender, ImageClickEventArgs e)
{
    Panel1.Visible = false;
    Panel2.Visible = true;
    lbldate.Text = txtdate.Text;
    lbllove.Text = "";
    for (int i = 0; i < chblove.Items.Count; i++)
    {
        if (chblove.Items[i].Selected)
            lbllove.Text += " " + chblove.Items[i].Text;
    }
}

```

```
lblname.Text = txtname.Text;
lblpwd.Text = txtpwd.Text;
lblschool.Text = ddlschool.SelectedItem.Text;
lblsex.Text = rbtsex.SelectedValue;

}
protected void btnclose_Click(object sender, EventArgs e)
{
    Response.Write("<script>window.close();</script>");
}
protected void btnOk_Click1(object sender, ImageClickEventArgs e)
{
    Panel1.Visible = false;
    Panel2.Visible = true;
    lbldate.Text = txtdate.Text;
    lbllove.Text = "";
    for (int i = 0; i < chblove.Items.Count; i++)
    {
        if (chblove.Items[i].Selected)
            lbllove.Text += " " + chblove.Items[i].Text;
    }
    lblname.Text = txtname.Text;
    lblpwd.Text = txtpwd.Text;
    lblschool.Text = ddlschool.SelectedItem.Text;
    lblsex.Text = rbtsex.SelectedValue;
}
protected void Ibtncancel_Click(object sender, ImageClickEventArgs e)
{
    Response.Redirect("addmember.aspx");
}
```

程序说明：

程序用到了 `page_load` 事件，该事件在页面加载时被引发。其中，设置 `Panel` 控件的 `Visible` 属性为 `true` 和 `false`，来控制 `Panel1` 容器和容器内元素可见，`Panel2` 容器和容器内元素不可见。“`txtpwd.Attributes["value"] = txtpwd.Text;`”的作用是可以有效防止传回服务器时 `Password` 类型文本框中的内容丢失。

`btnclose_Click` 事件中，“`Response.Write("<script>window.close();</script>");`”的作用是利用 `JavaScript` 脚本语言关闭当前窗口。

`Ibtncancel_Click` 事件中，“`Response.Redirect("addmember.aspx");`”是利用 `Response` 的 `Redirect` 方法将页面转向 `addmember.aspx`，从而实现页面的再次加载，即重新打开 `addmember.aspx` 页面。

提示：由于程序中使用较多的服务器控件，用户在设置控件的 `ID` 属性时，建议用户遵循“见名知意”的原则。例如，“提交按钮”`ImageButton` 的 `ID` 属性可以用“`Ibtnsubmit`”标记。其中，“`Ibtn`”是 `ImageButton` 的缩写，而 `submit` 是“提交”的意思；“用户名”文本框标记为“`txtname`”；“关闭”按钮标记为“`btnclose`”等。

2.5 知识拓展

2.5.1 Panel 控件

Panel 控件在 Web 页面设计中为其他控件提供一个容器，用户可以将 Panel 内的控件作为一个整体进行管理。通常，Panel 控件在使用时，常常通过页面事件代码控制其 Visible 属性，来实现控件与控件内元素的显示和隐藏。

【例 2-14】利用 Panel 控件实现页面部分内容的显示和隐藏，如图 2-19 所示。用户单击“查看详细说明”链接时，出现详细说明文字内容和按钮；当用户单击“已阅读，隐藏”按钮时，说明文字和按钮再次被隐藏（Ex2-14.aspx）。

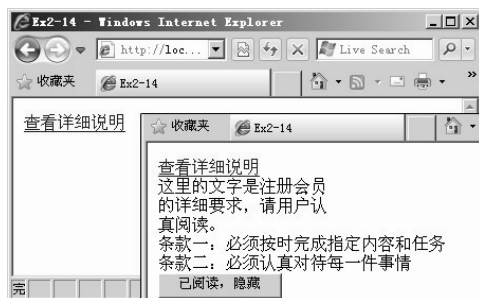


图 2-19 Panel 控件用法

Ex2-14.aspx 文件代码如下：

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Ex2-14.aspx.cs" Inherits="Ex2_14" %>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Ex2-14</title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <asp:LinkButton ID="lbtnlook" runat="server" onclick="lbtnlook_Click">查看详细说明</asp:LinkButton>
            <asp:Panel ID="Panel1" runat="server" Visible="False">
                <asp:Label ID="lbl1" runat="server" Text="这里的文字是注册会员的详细要求，请用户认真阅读。"
                    Width="160px"/><br />
                <asp:Label ID="lbl2" runat="server" Text="条款一：必须按时完成指定内容和任务"/><br />
                <asp:Label ID="lbl3" runat="server" Text="条款二：必须认真对待每一件事情"/><br />
                <asp:Button ID="btnok" runat="server" onclick="btnok_Click" Text="已阅读，隐藏" />
            </asp:Panel>
        </div>
    </form>
</body>
</html>
```

Ex2-14.aspx.cs 文件主要代码如下：

```
protected void btnok_Click(object sender, EventArgs e)
{
    Panel1.Visible = false;
}
protected void lbtnlook_Click(object sender, EventArgs e)
{
    Panel1.Visible = true;
}
```

2.5.2 Image 控件

Image 控件主要用于页面显示图像信息。VS 2008 工具箱中有标准类的 Image 控件和 HTML 类的 Image 控件两种，前者表示标准服务器控件，常用于显示动态变化的图像；后者是 HTML 服务器控件，用于页面一般静态图片的显示。这里仅针对标准服务器控件 Image 进行介绍。

【例 2-15】利用 Image 控件实现用户个性头像的选择，如图 2-20 所示。用户通过下拉菜单进行头像选择，显示图片随之变化。该实例操作前，用户要在网站的 images 文件夹里放置 4 张头像图片，文件名分别为 man1.jpg、wom1.jpg、man2.jpg 和 wom2.jpg（Ex2-15.aspx）。



图 2-20 Image 控件用法

Ex2-15.aspx 文件代码如下：

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Ex2-15.aspx.cs" Inherits="Ex2_15" %>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Ex2-15</title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            请选择头像:
            <asp:DropDownList ID="ddl1" AutoPostBack="True" runat="server" onselectedindexchanged=
                "ddl1_SelectedIndexChanged">
                <asp:ListItem Value="man1" Selected="True">男生头像 1</asp:ListItem>
```

```
<asp:ListItem Value="wom1">女生头像 1</asp:ListItem>
<asp:ListItem Value="man2">男生头像 2</asp:ListItem>
<asp:ListItem Value="wom2">女生头像 2</asp:ListItem>
</asp:DropDownList><br />
<asp:Image ID="Image1" runat="server" Height="140px" Width="140px" />
</div>
</form>
</body>
</html>
```

Ex2-15.aspx.cs 文件主要代码如下:

```
protected void Page_Load(object sender, EventArgs e)
{
    Image1.ImageUrl = "~/images/" + ddl1.SelectedValue + ".jpg";
}
protected void ddl1_SelectedIndexChanged(object sender, EventArgs e)
{
    Image1.ImageUrl = "~/images/" + ddl1.SelectedValue + ".jpg";
}
```

程序说明:

Page_Load 事件中, “Image1.ImageUrl = “~/images/” + ddl1.SelectedValue + “.jpg”;” 的作用是在页面加载时显示 DropDownList 控件默认选项对应的头像。

2.5.3 ListBox 控件

ListBox 控件是和 DropDownList 控件功能类似的列表框控件, 用户可以从列表框中选择选项。ListBox 控件和 DropDownList 控件的区别在于前者允许用户一次选择多个选项, 而后者只允许一次选择一个选项。ListBox 控件的常用属性及说明如表 2-7 所示。

表 2-7 ListBox 控件的常用属性及说明

属性	说明
Items	返回 ListBox 控件中 ListItem 对象
Rows	控件内容显示的行数
SelectedIndex	控件被选中的 Index 值
SelectedItem	控件被选中的 ListItem 对象
SelectedItems	返回控件被多选时的 ListItem 集合
SelectionMode	设置控件是否支持多选, 值为 Multiple 支持多选。默认值为 Single

【例 2-16】利用 ListBox 控件实现用户个人爱好多项选择, 如图 2-21 所示。用户通过在选项中选择相应的项, 单击“提交”按钮后, 右边的提示文字随之变化。该案例支持使用 Ctrl 键和 Shift 键进行多项选择 (Ex2-16.aspx)。



图 2-21 ListBox 控件用法

Ex2-16.aspx 文件主要代码如下:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Ex2-16.aspx.cs" Inherits="Ex2_16" %>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Ex2-16</title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            请选择你的爱好: <br />
            <asp:ListBox ID="ListBox1" runat="server" Height="114px" SelectionMode="Multiple" Width="84px">
                <asp:ListItem>唱歌</asp:ListItem>
                <asp:ListItem>跳舞</asp:ListItem>
                <asp:ListItem>游泳</asp:ListItem>
                <asp:ListItem>爬山</asp:ListItem>
                <asp:ListItem>旅行</asp:ListItem>
                <asp:ListItem>钓鱼</asp:ListItem>
            </asp:ListBox>
            <asp:Button ID="Button1" runat="server" onclick="Button1_Click" Text="提交" />
            <asp:Label ID="Label1" runat="server" Text="Label"></asp:Label>
        </div>
    </form>
</body>
</html>
```

Ex2-16.aspx.cs 文件主要代码如下:

```
protected void Button1_Click(object sender, EventArgs e)
{
    string sss = "你选择的是: ";
    for (int i = 0; i < ListBox1.Items.Count; i++)
    {
        ListItem item = ListBox1.Items[i];
        if (item.Selected)
        {
            sss += item.Text + " ";
        }
    }
}
```

```
    }  
    }  
    Label1.Text = sss;  
}
```

程序说明：

将 ListBox 控件的 SelectionMode 属性设置为 Multiple，这样使得 ListBox 支持 Ctrl 键和 Shift 键选择多项。

“string sss = "你选择的是：";”表示声明 1 个 string 字符串变量 sss 并赋初值。相应地，
“ListItem item = ListBox1.Items[i];”语句则是声明 1 个 ListItem 变量 item 并赋初值。