

项目 3 设计制作网络调查系统

本项目学习目标

- 掌握常见控件的使用方法。
- 掌握 ASP.NET 内置对象的使用方法。
- 掌握 Calendar 控件的使用方法。

本项目实现了一个简单的网络调查系统的设计。本网络调查系统的设计思路为：首先用户设置调查项并生成调查问卷,对网上的浏览者进行调查,网上浏览者对调查问卷进行相应的选择和回答后,提交一个调查结果,调查结果在服务器端汇总,得出调查结论。本系统将为使用者提供一个迅速、便捷的得到用户反馈信息的方式,及时地了解用户对某些问题的真实态度,为了解用户的反应以及为接下来的决策提供参考和依据。

在系统的设计中,我们将用到 ASP.NET 的一些常用控件,比如选择控件,包括单选控件和复选控件以及选择控件组;另外我们还会用到 ASP.NET 常见的内置对象,如 Response、Request、Server、Application、Session 等。同时我们还将接触到其他一些控件。

任务 3.1 使用常见控件

3.1.1 使用 RadioButton 和 RadioButtonList 控件

1. 要求与目的

要求：制作如图 3-1 所示的学生注册界面,要求用户填写姓名,选择自己的性别,以及选择年级,年级包括“2007 级”、“2008 级”和“2009 级”。填写完成后单击“提交”按钮,显示出用户所填内容。

目的：

- 掌握单选按钮控件 RadioButton 常用的属性。
- 掌握单选列表控件 RadioButtonList 常用的属性。
- 掌握单选按钮控件 RadioButton 的编程方法。
- 掌握单选列表控件 RadioButtonList 的编程方法。

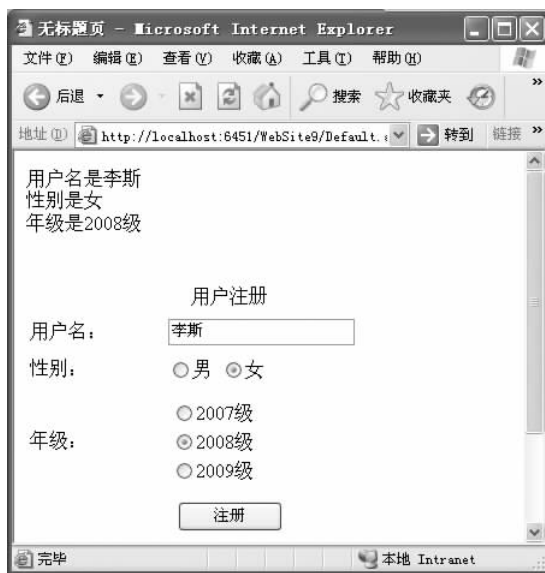


图 3-1 界面预览效果

2. 操作步骤

(1) 界面设计

打开 Visual Studio 2012, 新建一个 ASP.NET 空网站, 语言选择 Visual C#, 然后添加一个 ASP.NET 页面。首先做一个布局表格, 在表格中输入如下文本: “用户注册”、“用户名”、“性别”、“年级”。在对应的位置添加如下控件: 一个 TextBox 文本框控件, 用于存放用户名; 两个 RadioButton 按钮控件, 用于提供给用户来选择性别; 一个 RadioButtonList 控件, 用于提供给用户来选择年级, 如图 3-2 所示。



图 3-2 设计界面

布局完成后, 修改控件的属性。分别单击两个 RadioButton 控件的右侧, 修改其 Text 属性为“男”和“女”。

再单击 RadioButtonList 控件的智能标签, 选择“编辑项”选项, 出现如图 3-3 所示的

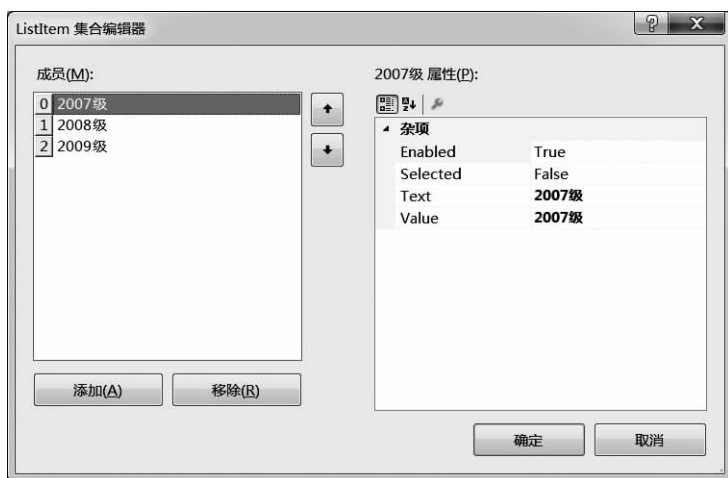


图 3-3 编辑单选项

界面。

在 ListItem 集合编辑器中添加三项,分别是“2007 级”、“2008 级”、“2009 级”,添加完成后单击“确定”按钮。

最后修改 Button 按钮的 Text 属性为“注册”。

界面设计好之后,Default.aspx 页面的 HTML 源文件如代码 3-1 所示。

代码 3-1 主要控件的 HTML 源文件

```
<table class="style1">
  <tr>
    <td colspan="2" style="text-align: center">
      用户注册</td>
  </tr>
  <tr>
    <td class="style2">
      用户名: </td>
    <td>
      <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
    </td>
  </tr>
  <tr>
    <td class="style2">
      性别: </td>
    <td>
      <asp:RadioButton ID="RadioButton1" runat="server" Text="男" />
      <asp:RadioButton ID="RadioButton2" runat="server" Text="女" />
    </td>
  </tr>
  <tr>
    <td class="style2">
      年级: </td>
```

```

        <td>
            <asp:RadioButtonList ID="RadioButtonList1" runat="server">
                <asp:ListItem>2007 级</asp:ListItem>
                <asp:ListItem>2008 级</asp:ListItem>
                <asp:ListItem>2009 级</asp:ListItem>
            </asp:RadioButtonList>
        </td>
    </tr>
    <tr>
        <td colspan="2" style="text-align: center">
            <asp:Button ID="Button1" runat="server" Height="25px" Text="注册" Width=
            "87px" />
        </td>
    </tr>
</table>

```

(2) 编写事件代码

双击 Button 按钮控件,进入该按钮的单击事件,编写程序如代码 3-2 所示。

代码 3-2 Button 按钮的单击事件

```

protected void Button1_Click(object sender, EventArgs e)
{
    string a1=TextBox1.Text;
    string a2="";
    if (RadioButton1.Checked)
    {
        a2="男";
    }
    else if (RadioButton2.Checked)
    {
        a2="女";
    }
    string a3=RadioButtonList1.SelectedItem.ToString();
    Response.Write("用户名是"+a1+"<br>");
    Response.Write("性别是"+a2+"<br>");
    Response.Write("年级是"+a3+"<br>");
}

```

(3) 编译、运行并测试程序

代码编写完之后,按 F5 键编译并运行该网站,再输入数据进行测试,效果如图 3-4 所示。

3. 相关知识点

(1) RadioButton 控件常用的属性

- AutoPostBack 属性:用于设置当单击 RadioButton 控件时,是否自动将信息回送到服务器。为 True 时,表示回送;为 False 时(默认),表示不回送。
- Checked 属性:用于获取或设置单选按钮的选中状态。属性值为布尔型值。为

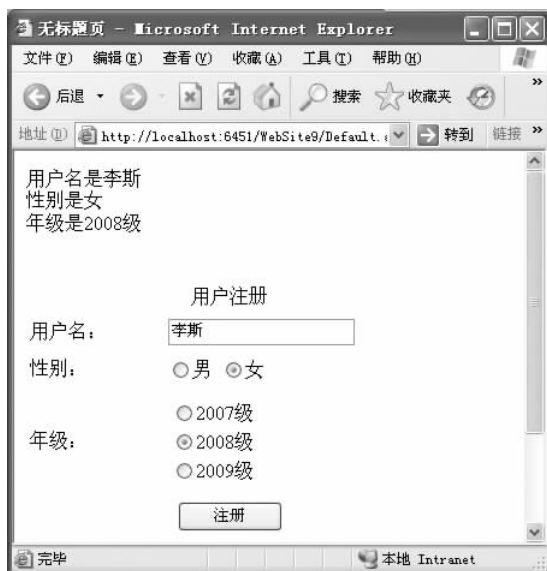


图 3-4 界面运行效果

True 时,表示选中;为 False 时(默认),表示不选中。

- Text 属性: 用于获取或设置单选按钮的文本。
- TextAlign 属性: 用于指定单选按钮显示文本的位置。为 Right(默认)时,文本显示在单选按钮的右边;为 Left 时,文本显示在单选按钮的左边。

(2) RadioButton 控件常用的事件

Checkedchange 事件: 当向服务器回送信息时,如果 checked 属性值发生了变化,将引发该事件。

(3) RadioButtonList 控件常用的属性

- AutoPostBack 属性: 用于设置当单击 RadioButtonList 控件时,是否自动将相关信息回送到服务器。True 表示回送;False(默认)表示不回送。
- DataSource 属性: 用于指定填充列表控件的数据源。
- DataTextField 属性: 用于指定 DataSource 中的一个字段,该字段的值对应于列表项的 Text 属性。
- DataValueField 属性: 用于指定 DataSource 中的一个字段,该字段的值对应于列表项的 Value 属性。
- Items 属性: 表示列表中各个选项的集合,如 RadioButtonList1.Items(i)表示第 i 个选项,i 从 0 开始。每个选项都有以下 3 个基本属性。
 - ◆ Text 属性: 表示每个选项的文本。
 - ◆ Value 属性: 表示每个选项的选项值。
 - ◆ Selected 属性: 表示该选项是否被选中。
- Count 属性: 通过 Items.Count 属性可获得 CheckBoxList 控件的选项数。
- Add 方法: 通过 Items.Add 方法可以向 CheckBoxList 控件添加选项。

- Remove 方法：通过 Items.Remove 方法，可从 CheckBoxList 控件中删除指定的选项。
- Insert 方法：通过 Items.Insert 方法，可将一个新的选项插入到 CheckBoxList 控件中。
- Clear 方法：通过 Items.Clear 方法可以清空 CheckBoxList 控件中的选项。
- RepeatColumns 属性：用于指定在 CheckBoxList 控件中显示选项时占用几列。默认值为 0，表示任意多列。
- RepeatDirection 属性：用于指定 CheckBoxList 控件的显示方向。为 Vertical 时，列表项以列优先排列的形式显示；为 Horizontal 时，列表项以行优先排列的形式显示。
- RepeatLayout 属性：用于设置选项的排列方式。为 Table(默认)时，以表结构显示；为 Flow 时，不以表结构显示。
- SelectedIndex 属性：用于获取或设置列表中选定项的最低序号索引值。如果列表控件中只有一个选项被选中，则该属性表示当前选定项的索引值。
- SelectedItem 属性：用于获取列表控件中索引值最小的选定项。如果列表中只有一个选项被选中，则该属性表示当前选定项。通过该属性可获得选定项的 Text 和 Value 属性值。
- TextAlign 属性：用于指定列表中各项文本的显示位置。当该属性值为 Right(默认)时，文本显示在单选按钮的右边；当属性值为 Left 时，文本显示在单选按钮的左边。

(4) RadioButtonList 控件常用的事件

SelectIndexChanged 事件：当用户选择了列表中的任意选项时，都将触发 SelectedIndexChanged 事件。

3.1.2 使用 CheckBox 控件和 CheckBoxList 控件

1. 要求与目的

要求：在前面操作的基础上增加用户注册的功能，添加用户提交的信息。增添一个用户爱好选择项，以及本学期选修课程选择项，效果如图 3-5 所示。

目的：

- 了解 CheckBox 控件的常用属性。
- 了解 CheckBoxList 控件的常用属性。
- 掌握 CheckBox 控件常用的编程方法。
- 掌握 CheckBoxList 控件常用的编程方法。

2. 操作步骤

(1) 界面设计

在项目 3 前面操作的基础上添加两项，分别是个人爱好和本学期选修课程栏目。其

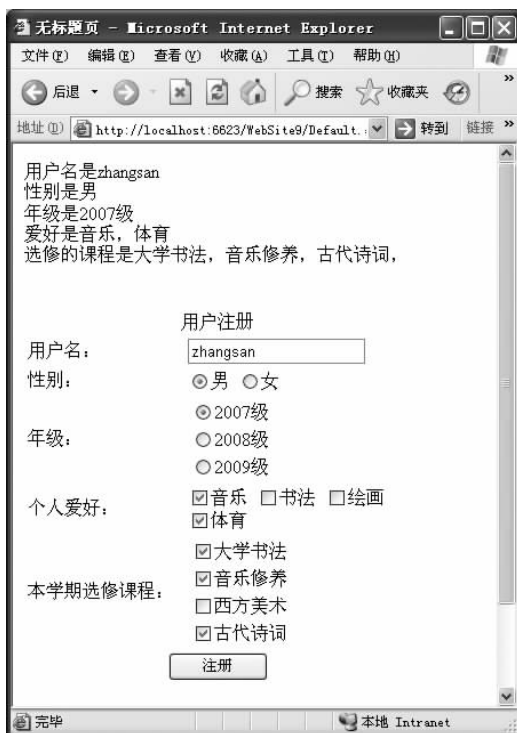


图 3-5 运行效果预览

中个人爱好使用四个 CheckBox 复选按钮控件来实现, 本学期所选课程使用一个 CheckBoxList 复选按钮列表控件来实现。效果如图 3-6 所示。



图 3-6 设计界面

在做好布局之后, 修改控件的属性。首先分别右击四个 CheckBox 控件, 修改其 Text 属性分别为“音乐”、“书法”、“绘画”、“体育”。单击 CheckBoxList 复选按钮列表控件的智能标签, 选择“编辑项”选项, 打开如图 3-7 所示的 ListItem 集合编辑器界面。

在该编辑器对话框中添加四个选项, 分别是“大学书法”、“音乐修养”、“西方美术”、“古代诗词”。



图 3-7 “ListItem 集合编辑器”对话框

界面设计好之后,Default.aspx 页面的主要 HTML 源文件如代码 3-3 所示。

代码 3-3 主要控件的 HTML 源文件

```
<table class="style1">
  <tr>
    <td colspan="2" style="text-align: center">
      用户注册</td>
  </tr>
  <tr>
    <td class="style2">
      用户名: </td>
    <td>
      <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
    </td>
  </tr>
  <tr>
    <td class="style2">
      性别: </td>
    <td>
      <asp:RadioButton ID="RadioButton1" runat="server" Text="男" />
      <asp:RadioButton ID="RadioButton2" runat="server" Text="女" />
    </td>
  </tr>
  <tr>
    <td class="style2">
      年级: </td>
    <td>
      <asp:RadioButtonList ID="RadioButtonList1" runat="server">
        <asp:ListItem>2007 级</asp:ListItem>

```

```

        <asp:ListItem>2008 级</asp:ListItem>
        <asp:ListItem>2009 级</asp:ListItem>
    </asp:RadioButtonList>
</td>
</tr>
<tr>
    <td class="style2">
        个人爱好: </td>
    <td>
        <asp:CheckBox ID="CheckBox1" runat="server" Text="音乐" />
        <asp:CheckBox ID="CheckBox2" runat="server" Text="书法" />
        <asp:CheckBox ID="CheckBox3" runat="server" Text="绘画" />
        <asp:CheckBox ID="CheckBox4" runat="server" Text="体育" />
    </td>
</tr>
<tr>
    <td class="style2">
        本学期选修课程: </td>
    <td>
        <asp:CheckBoxList ID="CheckBoxList1" runat="server">
            <asp:ListItem>大学书法</asp:ListItem>
            <asp:ListItem>音乐修养</asp:ListItem>
            <asp:ListItem>西方美术</asp:ListItem>
            <asp:ListItem>古代诗词</asp:ListItem>
        </asp:CheckBoxList>
    </td>
</tr>
<tr>
    <td colspan="2" style="text-align: center">
        <asp:Button ID="Button1" runat="server" Height="25px"
Text="注册" Width="87px" />
    </td>
</tr>
</table>

```

(2) 编写事件代码

双击“注册”按钮,进入该按钮的单击事件,编写程序如代码 3-4 所示。

代码 3-4 “注册”按钮的单击事件

```

protected void Button1_Click(object sender, EventArgs e)
{
    string a1=TextBox1.Text;
    string a2="";
    if (RadioButton1.Checked)
    {
        a2="男";
    }
    else if (RadioButton2.Checked)

```

```
{
    a2="女";
}
string a3=RadioButtonList1.SelectedItem.ToString();
string a4="";
if (CheckBox1.Checked)
{
    a4+=CheckBox1.Text+",";
}
if (CheckBox2.Checked)
{
    a4+=CheckBox2.Text+",";
}
if (CheckBox3.Checked)
{
    a4+=CheckBox3.Text+",";
}
if (CheckBox4.Checked)
{
    a4+=CheckBox4.Text;
}

string a5="";
Response.Write("用户名是"+a1+"<br>");
Response.Write("性别是"+a2+"<br>");
Response.Write("年级是"+a3+"<br>");
Response.Write("爱好是"+a4+"<br>");
for(int i=0;i<CheckBoxList1.Items.Count;i++)
{
    if(CheckBoxList1.Items[i].Selected==true)
    {
        a5+=CheckBoxList1.Items[i].Value.ToString()+",";
    }
}
Response.Write("选修的课程是"+a5);
}
```

(3) 编译、运行并测试程序

代码编写完之后,保存文件,再编译、运行并测试该程序,效果如图 3-8 所示。

3. 相关知识点

(1) CheckBox 控件常用的属性

- AutoPostBack 属性:用于设置当单击 CheckBox 控件时,是否自动将信息回送到服务器。True 表示回送;False(默认)表示不回送。
- Checked 属性:指示是否选中该控件。默认值是 False。
- TextAlign 属性:确定如何对齐文本。默认值 Right。
- Text 属性:用于获取或设置复选框的文本。