**实验四 ASP.NET3.5 标准控件**

**一．实验目的**

1. 理解 ASP.NET3.5 页面事件处理流程。

2. 理解 ASP.NET3.5 标准控件的应用。

**二．实验预习**

1. 了解了 ASP.NET3.5 页面事件的处理流程。

2. 了解了怎么使用 ASP.NET3.5 标准控件。

**三．实验内容**

1.设计并实现一个简易的计算器

2.设计并实现一个用于查询教师课表的联动下拉列表框页面

3.设计并实现一个用于单项选择题的测试页面

**四．实验过程**

**1. 设计并实现一个简易的计算器**

（1）设计 Web 窗体 新建一个 Web 窗体，切换到设计视图。向页面添加 1 个用于布局的 1 行 1 列黑色边框表格。向表格中添加一个 TextBox 控件、6 个 Botton 控件。

（2）设置属性 Web 窗体中各控件的属性。

（3）编写事件、方法代码 在所有事件、方法外声明静态字段。静态字段保证了其中的数据可以在所有的事件、 方法中被访问，并能在事件代码或方法代码执行结束后保留数据。其中，静态字段 num1 用于存储算式中的第一个数字字符串，初始值为“0”；num2 用于存储算式中的第二个 数字字符串，初始值为“0”；total 用于存储将所有输入的数连接后的数字字符串，初 始值为“”；sign 用于存储运算符号，初始值为“”。

代码如下：

static string num1 = "0", num2 = "0", total = "", sign = "";

按钮btnOne 被单击时执行的事件

代码如下：

protected void Button1\_Click(object sender, EventArgs e)

{

total += "1";

txtDisplay.Text = total;

}

按钮 btnTwo 被单击时执行的事件代码如下：

protected void Button2\_Click1(object sender, EventArgs e)

{

total += "2";

txtDisplay.Text = total;

}

按钮 btnThree 被单击时执行的事件代码如下：

protected void Button3\_Click1(object sender, EventArgs e)

{

total += "3";

txtDisplay.Text = total;

}

按钮 btnAdd 被单击时执行的事件代码如下：

protected void Button4\_Click(object sender, EventArgs e)

{

if (sign.Length == 1)

{

Count();

num1 = txtDisplay.Text;

sign = "+";

}

else

{

num1 = txtDisplay.Text;

txtDisplay.Text = "";

total = "";

sign = "+";

}

}

按钮 btnSubtract 被单击时执行的事件代码如下：

protected void Button5\_Click(object sender, EventArgs e)

{

if (sign.Length == 1)

{

Count();

num1 = txtDisplay.Text;

sign = "-";

}

else

{

num1 = txtDisplay.Text;

txtDisplay.Text = "";

total = "";

sign = "-";

}

}

按钮 btnEqual 被单击时执行的事件代码如下：

protected void Button6\_Click(object sender, EventArgs e)

{

Count();

}

public void Count()

{

num2 = txtDisplay.Text;

if (num2 == "")

{

num2 = "0";

}

switch (sign)

{

case "+":

txtDisplay.Text = (int.Parse(num1) + int.Parse(num2)).ToString();

num1 = "0";

num2 = "0";

total = "";

sign = "";

break;

case "-":

txtDisplay.Text = (int.Parse(num1) - int.Parse(num2)).ToString();

num1 = "0";

num2 = "0";

total = "";

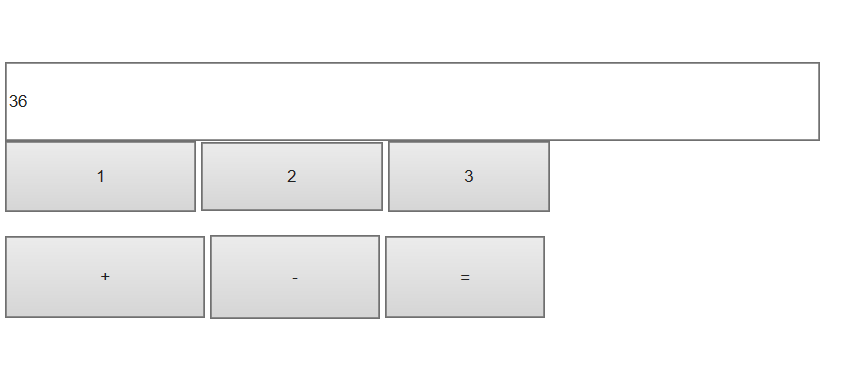
sign = "";

break;

}

}

最后，浏览建立的 Web 窗体查看效果。



**2. 设计并实现一个用于查询教师课表的联动下拉列表框页面**

（1）设计 Web 窗体 新建一个 Web 窗体，切换到设计视图。向页面输入“学年：” 、 “学期：”、 “分院：”和“教师：”等信息，适当设置字体大小。再向页面添加 4 个 DropDownList 控件，适当调整个控件的大小和位置。

（2）设置属性 Web 窗体中各控件的属性设置

（3）编写事件、方法代码 Web 窗体首次载入时执行的代码如下：

protected void Page\_Load(object sender, EventArgs e)

{

if (!IsPostBack){

BindYear();

BindTerm();

BindCollege();

BindTeacher();

}

}

当“分院”下拉列表框中选择项改变时，触发的事件代码如下：

protected void DropDownList3\_SelectedIndexChanged(object sender, EventArgs e)

{

BindTeacher();

}

自定义方法 BingYear()的代码如下：

protected void BindYear()

{

ddlYear.Items.Clear();

int startYear = DateTime.Now.Year - 10;

int currentYear = DateTime.Now.Year;

for (int i = startYear; i < currentYear; i++)

{

ddlYear.Items.Add(new ListItem((i-1).ToString()+"-"+i.ToString()));

}

ddlYear.SelectedValue=(currentYear-1).ToString()+"-"+currentYear.ToString();

}

自定义方法 BingTerm()的代码如下：

protected void BindTerm()

{

ddlTerm.Items.Clear();

for (int i = 0; i <= 2; i++)

{

ddlTerm.Items.Add(i.ToString());

}

}

自定义方法 BingCollege()的代码如下：

protected void BindCollege()

{

ddlCollege.Items.Clear();

ddlCollege.Items.Add(new ListItem("计?算?机¨²科?学¡ì学¡ì院o"));

ddlCollege.Items.Add(new ListItem("外ªa国¨²语®?学¡ì院o"));

ddlCollege.Items.Add(new ListItem("机¨²电Ì?学¡ì院o"));

}

自定义方法 BingTeacher()的代码如下：

protected void BindTeacher()

{

ddlTeacher.Items.Clear();

switch (ddlCollege.SelectedValue)

{

case "计?算?机¨²科?学¡ì学¡ì院o":

ddlTeacher.Items.Add(new ListItem("沈¦¨°世º¨¤庚y"));

ddlTeacher.Items.Add(new ListItem("沈¦¨°庚y"));

ddlTeacher.Items.Add(new ListItem("世º¨¤庚y"));

ddlTeacher.Items.Add(new ListItem("沈¦¨°世º¨¤"));

break;

case "外ªa国¨²语®?学¡ì院o":

ddlTeacher.Items.Add(new ListItem("张?三¨y"));

ddlTeacher.Items.Add(new ListItem("李¤?四?"));

ddlTeacher.Items.Add(new ListItem("顽ª?固¨¬"));

ddlTeacher.Items.Add(new ListItem("王ª?五?"));

break;

case "机¨²电Ì?学¡ì院o":

ddlTeacher.Items.Add(new ListItem("许¨ª晓t东?"));

ddlTeacher.Items.Add(new ListItem("许¨ª东?"));

ddlTeacher.Items.Add(new ListItem("晓t东?"));

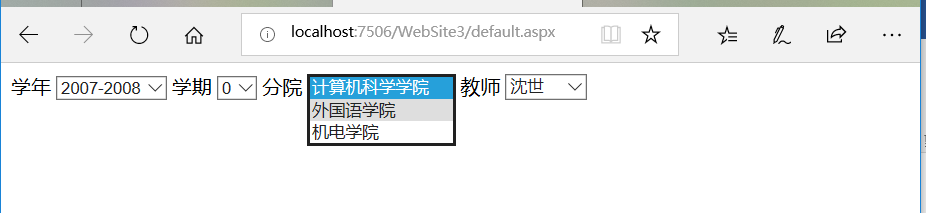
break;

}

}

}

最后，浏览建立的 Web 窗体查看效果。



**3. 设计并实现一个用于单项选择题的测试页面**

（1）设计 Web 窗体新建一个Web窗体，切换到设计视图。向页面添加 1个PlaceHolder 控 件、1个Botton控件和1个Label控件，适当调整个控件的大小和位置。

（2）设置属性 Web 窗体中各控件的初始属性设置。

（3）编写事件代码 Web 窗体首次载入时执行的代码如下：

protected void Page\_Load(object sender, EventArgs e)

{

Label lblQuesttion1 = new Label();

lblQuesttion1.ID = "lblQuestion1";

lblQuesttion1.Text = "1.web服¤t务?器¡Â控?件t";

PlaceHolder1.Controls.Add(lblQuesttion1);

RadioButtonList rdoltChoicel = new RadioButtonList();

rdoltChoicel.ID = "rdoltChoicel";

rdoltChoicel.Items.Add(new ListItem("A.Wizard ", "A"));

rdoltChoicel.Items.Add(new ListItem("B.input ", "B"));

rdoltChoicel.Items.Add(new ListItem("C.Adrotator ", "C"));

rdoltChoicel.Items.Add(new ListItem("D.Calender ", "D"));

}

按钮 btnSubmit 被单击时执行的事件代码如下：

protected void Button1\_Click1(object sender, EventArgs e)

{

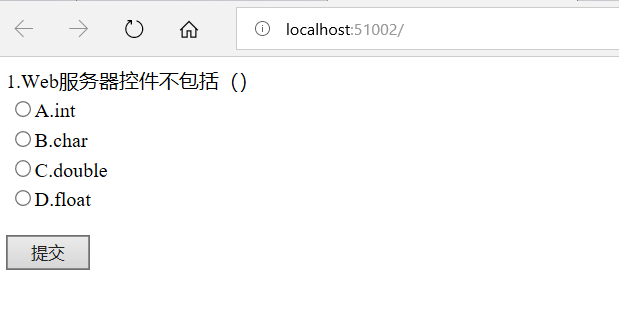
RadioButtonList rdoltChoicel = (RadioButtonList)PlaceHolder1.FindControl("rdoltChpicel");

lblDisplay.Text = "你?选?择?了¢?" + rdoltChoicel.SelectedValue;

PlaceHolder1.Controls.Add(rdoltChoicel);

}

（4）浏览建立的 Web 窗体进行测试。



**五．实验结果**

关于 ASP.NET 3.5 页面事件的处理流程主要是对控件的使用非常不熟悉。这个需要自己课下多多练习。

**六．心得体会**

通过本次实验，了解 ASP.NET 页面事大概件处理流程。