

## 实验 4 服务器控件的使用

### 目的与要求

- (1) 掌握常用的 Web 服务器控件的使用方法。
- (2) 熟悉各控件的常用属性和方法。
- (3) 掌握验证控件的使用方法。

### 内容和步骤

【实验题 4.1】使用 Web 服务器控件 Label、TextBox 和 Button，设计如图 p4.1 所示的页面。使用 C#语言，实现基本的加减乘除运算功能，并将计算结果输出。

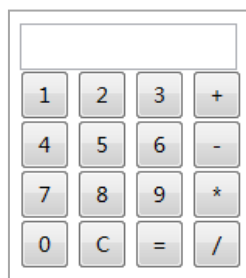


图 4.1 实验题 4.1 界面

步骤如下：

(1) 启动 Visual Studio 2010，单击“创建：项目...”链接，即弹出“新建项目”对话框，左侧“项目类型”列表框中选择“Visual C# → Web”；在“模板”列表框中选择“ASP.NET Web 应用程序”。定义项目名称，并设置项目存储位置。

(2) 在 Design 视图，从 Toolbox 窗口选择若干控件，设置各控件属性及布局。

(3) 编写逻辑代码：

```
protected void Page_Load(object sender, EventArgs e)
{
    if (!IsPostBack) { LabelNum1.Text = LabelNum2.Text = "0"; }
}
protected void getInputValue(object sender, EventArgs e)
{
    TBResult.Text += ((Button)sender).Text;
}
protected void Dispose(object sender, EventArgs e)
{
    switch (((Button)sender).Text)
    {
        case "+":
            LabelNum1.Text = TBResult.Text;
            LabelOperator.Text = "+";
            TBResult.Text += LabelOperator.Text;
            break;
        case "-":
            LabelNum1.Text = TBResult.Text;
            LabelOperator.Text = "-";
            TBResult.Text += LabelOperator.Text;
            break;
        case "*":
            LabelNum1.Text = TBResult.Text;
            LabelOperator.Text = "*";
            TBResult.Text += LabelOperator.Text;
            break;
        case "/":
            LabelNum1.Text = TBResult.Text;
            LabelOperator.Text = "/";
            TBResult.Text += LabelOperator.Text;
            break;
    }
}
```

```

        case "C":
            TBResult.Text = "";
            LabelNum1.Text = LabelNum2.Text = "0";
            LabelOperator.Text = "";
            break;
    }
}

protected void calculate(object sender, EventArgs e)
{
    switch (LabelOperator.Text)
    {
        case "+":
            LabelNum2.Text = TBResult.Text.Substring(LabelNum1.Text.Length + 1);
            TBResult.Text += "=" + (Int32.Parse(LabelNum1.Text) +
                Int32.Parse(LabelNum2.Text)).ToString();

            break;
        case "-":
            LabelNum2.Text = TBResult.Text.Substring(LabelNum1.Text.Length + 1);
            TBResult.Text += "=" + (Int32.Parse(LabelNum1.Text) -
                Int32.Parse(LabelNum2.Text)).ToString();

            break;
        case "*":
            LabelNum2.Text = TBResult.Text.Substring(LabelNum1.Text.Length + 1);
            TBResult.Text += "=" + (Int32.Parse(LabelNum1.Text) *
                Int32.Parse(LabelNum2.Text)).ToString();

            break;
        case "/":
            LabelNum2.Text = TBResult.Text.Substring(LabelNum1.Text.Length + 1);
            TBResult.Text += "=" + (double.Parse(LabelNum1.Text) /
                double.Parse(LabelNum2.Text)).ToString();

            break;
    }
}
}
}

```

(3) 编译运行，结果如 p4.2 所示。

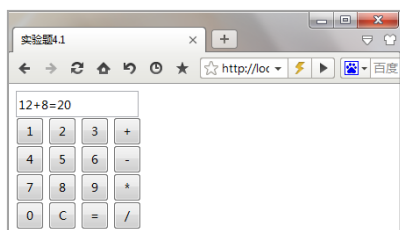


图 p4.2 实验题 4.1 的运行结果

**【实验题 4.2】**使用 Web 服务器控件 TextBox、Button、CheckBox、RadioButton、DropDownList、Image、Calendar、FileUpload 等以及各种验证控件，设计如图 p4.3 所示的页面，实现用户注册信息的填写，以及用户名、密码的必需性验证，两次密码的一致性验证，邮箱格式正确性的验证，以及照片上传与显示等功能。

其中学历右侧的下拉选择框包含中学、高中、大学、硕士、博士几项内容。

图 p4.3 实验题 4.2 界面示例

(1) 启动 Visual Studio 2010，单击“创建：项目...”链接，即弹出“新建项目”对话框，左侧“项目类型”列表框中选择“Visual C# → Web”；在“模板”列表框中选择“ASP.NET Web 应用程序”。定义项目名称并设置项目存储位置。

(2) 在 Design 视图，从 Toolbox 窗口选择若干控件，设置控件的属性及布局。

(3) 编写逻辑代码：

```
protected void btnSubmit_Click(object sender, EventArgs e)
{ Response.Write(@"<script language='javascript'>alert('提交完成!')</script>"); }
protected void btnUploadPic_Click(object sender, EventArgs e)
{ string strSavePath = @"d:\Upload\";
  if (!System.IO.Directory.Exists(strSavePath))
  { System.IO.Directory.CreateDirectory(strSavePath); }
  if (FileUpload1.HasFile)
  { string strFileName = DateTime.Now.Ticks.ToString() + FileUpload1.FileName; ;
    FileUpload1.SaveAs(Page.Server.MapPath("/") + strFileName);
    Image1.ImageUrl = strFileName;
  }
  else
  { Response.Write(@"<script language='javascript'>alert('请选择要上传的照片')</script>");
  }
}
```

(4) 编译运行，结果如 p4.4 所示。

图 p4.4 实验题 4.2 的运行结果

## 思考与练习

(1) 将【实验题 4.1】中的 Button 控件换成 ImageButton 控件，将三个 Visible 属性为 False 的 Label 控件换成 HiddenField 控件，重新编写实现计算器功能。

(2) 修改【实验题 4.2】中的日历控件属性和样式，改变所显示的日历的外观。

(3) 在【实验题 4.2】中增加身份证号、手机号码和固定电话三个文本输入框，并加入格式验证，其中，身份证号应为 15 位或 18 位数字，且 17 位数字的最后一位可为“x”。