

第 5 章 服务器控件



本章
导读

Web 应用需要和用户进行交互，而用户的信息必须通过各种表单传递给浏览器和服务器。本章学习常见的 ASP.NET Web 服务器控件的使用方法，通过实例给出了 DW CS3 中 Web 服务器控件的设计要点。



本章要点

- 表单基础
- Web 服务器控件

5.1 任务概述：用 ASP.NET 控件设计用户注册页面

用户登录或用户注册页面是一个典型的动态页面。在第 3 章“HTML 语言基础”中，我们利用静态网页的表单标记，设计了用户注册页面。本章我们将学习如何用 ASP.NET 控件，设计“动态”的用户注册页面。

5.2 表单

首先，我们学习基本的 ASP.NET 控件知识。

网页由 HTML 标记组成，包含在<Html>和</Html>标记之间，而表单是网页的一部分，包含在<Form>和</Form>标记之间。

5.2.1 Web 表单

表单是 Web 开发中的重要概念。ASP.NET 中有两类表单，Web 表单和 HTML 表单。

HTML 表单是指静态网页标记<Form>表示的表单。HTML 表单有两个重要的属性：Action 和 Method，分别用于指定处理表单内部数据的程序名称和数据传送的方法。在 HTML 表单中，Method 的值可以是 Get 或 Post，分别对应 HTTP 协议中的 Get 和 Post 方法。Get 方法表示将表单控件的信息经过编码之后，通过 URL 发送，在浏览器的地址栏中可以看到 Get 方法提交的数据。而 Post 方法的浏览器地址栏看不到表单的提交信息。因此，从安全性角度看，Post 方法安全性更好，并且可以发送较大的数据量。

HTML 表单只在浏览器端运行，用于向服务器提交用户的请求。

Web 表单也是用<Form>标记定义的。Web 表单标记中有一个重要的属性：

Runat="Server"。正是这个属性使 Web 表单与 HTML 表单有了本质的不同。

Web 表单标记格式：

```
<Form Runat="Server">
```

.....

```
</Form>
```

Web 表单在提交时默认采用 Post 方法，当表单标记未指定“action”属性时，表明由当前页面来处理。



注意

一个 ASP.NET 页面只能包含一个<Form Runat="Server">的 Web 表单。另外，虽然<Form>标记不显示任何信息，但<Form>是一个容器，表单项只有定义在<Form>中，才能将接收到的数据向 Web 服务器提交。

5.2.2 HTML 服务器控件和 Web 服务器控件

Web 表单由两类不同的控件组成：HTML 服务器控件和 Web 服务器控件。

HTML 服务器控件是在 HTML 表单基础上，通过增加 Runat="server"和 Id 属性形成的。例如，在静态网页中，一个输入用户名的 HTML 文本框标签为：

```
<input type="text" name="user" value="用户名">
```

在 ASP.NET 网页中，对应的 HTML 服务器控件为：

```
<input type="text" id="user" value="用户名" runat="server">
```

ASP.NET 中保留 HTML 服务器控件的目的是方便那些对于 HTML 表单非常熟悉的设计人员，能沿用类似的语法格式顺利过渡到使用 ASP.NET 技术。

Web 服务器控件是 ASP.NET 中增加的新型控件，具有强大的页面显示和事件处理能力。许多 Web 服务器控件类似于常见的 HTML 表单项，如按钮和文本框。但是，其他一些控件包含复杂的行为，如日历控件或管理数据连接的控件。

5.3 Web 服务器控件

Web 服务器控件以标记<asp: 控件名>开始，控件中包含 RunAt="Server"属性，一般也包含 Id 属性，Id 属性用于标识控件。Web 服务器控件可以有两种结束方式：以</asp: 控件名>结尾，或在开始标记的最后加斜杠<asp: 控件名 />。

Web 服务器控件既包括传统的窗体控件，例如按钮、文本框等控件，也包括其他窗体控件，例如在网格中显示数据、选择日期、验证表单等复杂控件。

5.3.1 Web 服务器控件的分类

ASP.NET 中的 Web 服务器控件可以分成四类：

- 基本 Web 控件：可以映射到 HTML 控件的 Web 控件，功能更强，实现起来更简单。
- 列表控件：用于大量数据的显示。
- 多功能控件：提供一些特定功能的控件，其对应的功能在 HTML 表单项中是没有的，如日历控件、广告控件。

● 验证控件：提供数据验证的控件，对应的功能在 HTML 表单项中也是没有的。其中，根据功能的不同，基本 Web 控件又可进一步分为以下三类：

①用于文本输入和显示的控件，如 TextBox、Label。

②用于控制传送的控件，如 Button、LinkButton、ImageButton、HyperLink。

③用于选择的控件，如 CheckBox、CheckBoxList、RadioButton、RadioButtonList、ListBox、DropDownList。

5.3.2 Dreamweaver CS3 中使用 Web 服务器控件概述

DW CS3 提供了可视化的界面，用于设计 Web 服务器控件。利用 DW CS3 的“插入记录”菜单，可以方便地在 ASP.NET 页面中增加 Web 服务器控件。

1. 增加 Web 服务器控件的方法

在 DW CS3 中，选择“文件”→“新建”→“ASP.NET VB”，点击“创建”按钮，在出现的工作区中有几种增加 Web 服务器控件的方法。

点击“插入记录”菜单，如图 5.1 所示。在下拉菜单中的“标签”和“ASP.NET 对象”两个菜单项中，包含了 Web 服务器控件。下面分别介绍从这两个菜单项中插入 ASP.NET 的 Web 服务器控件的方法。

(1) 从“插入记录”→“标签”菜单中选取 Web 服务器控件。

选取“标签”菜单项后，出现如图 5.2 所示的“标签选择器”，分成三个窗口，左上部的窗口是标签的分类。选择其中的“ASP.NET 标签”，可以进一步展开成五类控件，本章学习的控件包含在其中“Web 服务器控件”分类中。右上部的窗口是按字母排序的控件标签。下部是“标签信息窗口”。



图 5.1 包含 WEB 服务器控件的菜单项



图 5.2 标签选择器

(2) 从“插入记录”→“ASP.NET 对象”菜单中选取 Web 服务器控件。

如果选取“ASP.NET 对象”菜单，出现如图 5.3 所示的下拉菜单，下拉菜单中列出了十个常用的 ASP.NET 控件。



图 5.3 ASP.NET 对象

(3) 从“插入”面板中选取 Web 服务器控件。

除此之外，在“插入”面板组中也包含“ASP.NET”面板。点击“插入”面板组中的“ASP.NET”，可以看到如图 5.4 所示的 Web 服务器控件图标。



图 5.4 “插入”面板组中的 ASP.NET 控件图标

2. 自动加入默认表单

控件必须放在表单中才能起作用，也就是说 ASP.NET 中的 Web 服务器控件代码必须放在<Form Runat= "server">标记中。DW CS3 提供了自动加入默认表单的功能。如果是在网页中第一次加入一个服务器控件，DW CS3 会自动加入一个默认表单<Form runat="server">，服务器控件被包含在该表单中。

5.4 用于文本输入和显示的控件

ASP.NET 的 Web 服务器控件中有两个用于文本输入和显示的控件，它们是文本框控件<ASP:TextBox>和标签控件<ASP:Label>。

5.4.1 文本框控件<asp:textbox >

文本框控件<asp:textbox>用于文本输入和显示，可以实现 HTML 标签中的文本框<input

type=text>、密码框<input type=password>和多行文本框<textarea>的功能。格式如下:

```
<asp:textbox id="控件名" runat="server" text="文本内容"> </asp:textbox>
```

或

```
<asp:textbox id="控件名" runat="server" text="文本内容" />
```

显然,后者相当于将结束标记以一个“/”的形式,缩写在开始标记中。

在 DW CS3 中,从“插入对象”菜单中选取“ASP.NET 对象”中的“ASP:文本框”,将其插入网页中,出现如图 5.5 所示的对话框。点击“确定”按钮后,文本框就插入到网页中。



图 5.5 文本框设计界面

如果是第一次加入 Web 服务器控件, DW CS3 还会自动在网页代码中加入<Form runat="server">标记,如图 5.6 所示,其中上半部分的圆角矩形框中为包含<Form runat="server">标记的文本框代码,下半部分为 DW CS3 的设计窗口,已生成一个文本框,红色的虚线框代表表单。

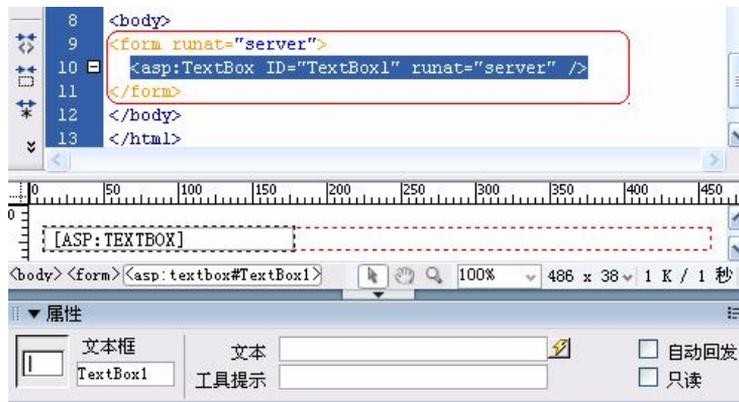


图 5.6 插入文本框后的代码及“属性”面板

在图 5.6 的最下面是“属性”面板,其中列出了文本框的 ID 值和文本值等部分属性。将 DW CS3 窗口最大化后,可以看到 DW CS3 的右下角有如图 5.7 所示的几个小图标,最下面是个倒立小三角形,点击该标记可以进一步展开“属性”面板,显示较多的属性设置内容。“属性”面板展开后,小三角变成朝上,点击该图标,可以关闭扩展“属性”面板。

点击扩展“属性”面板中“文本模式”旁的下拉箭头,出现“单行”、“多行”和“密码”三个选项。如果要插入多行文本框,则应该选择“多行”选项,同时,还需要在“行

数”属性中定义多行文本框能显示的行数；如果插入密码框，则应该选择“密码”选项，这样，文本框中的内容会显示成“*”号。插入的文本框默认情况下作为普通单行文本框，因此“单行”选项可以不用设置。

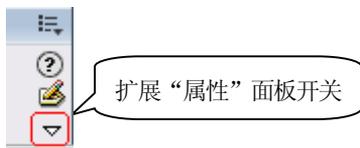


图 5.7 扩展“属性”面板的倒立小三角形图标

点击扩展“属性”面板右下角的图标，可以弹出如图 5.8 所示的文本框标签编辑器。在标签编辑器中包含了控件的所有属性，可以直接在标签编辑器中对 ASP.NET 控件进行编辑。



图 5.8 文本框的标签编辑器

文本框中最重要的属性是文本值“Text”。要获取文本框的内容或者设置文本框的显示文本，都是通过 Text 属性实现。

5.4.2 标签控件<asp:label>

标签控件<asp:label>用于在页面的某个位置显示文本信息。和文本框类似，标签控件的重要属性是文本值“Text”，要让标签控件显示文本信息，可以通过赋值给 Text 属性实现。格式：

```
<asp:label id="控件名" runat="server" text="标签内容">文本</asp:label>
```

或

```
<asp:label id="控件名" runat="server" text="标签内容" />
```

示例：

```
<asp:label text="信息提示" runat="server" id="lbcontent" />
```

将 DW CS3 工作区的文档窗口切换到“拆分”视图或“设计”视图中。从“插入记录”菜单中选取“ASP.NET 对象”中的“ASP:标签”，将其插入网页中，出现如图 5.9 所示的对话框。点击“确定”按钮，在网页中插入标签控件。

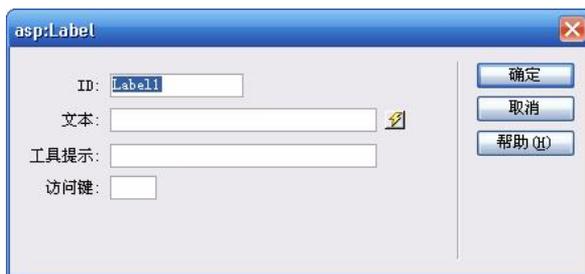


图 5.9 标签控件的设计界面

5.5 用于控制传送的控件

用于控制传送的控件包括按钮控件<asp:button>、链接式按钮控件<asp:linkButton>和图片按钮控件<asp:imageButton>, 主要是实现按钮的触发功能。由于超级链接的事件传送特性与上述三个按钮类似, 所以将超级链接控件<asp:hyperLink>也放在控制传送的控件中一并介绍。

5.5.1 按钮控件<asp:button>

按钮控件是网页中的常见元素, 用于执行一个命令或向服务器提交表单。ASP.NET 中的按钮格式为:

```
<asp:button id="控件名" runat="server" text="按钮面板上的文字" onClick="事件名" />
```

按钮控件典型的事件是 OnClick 事件, 该事件在点击按钮时触发。属性 text 是按钮的面板文字, 需要设置。按钮的 ID 属性一般可以忽略不定义。

在 DW CS3 工作区文档窗口的“拆分”视图或“设计”视图中, 从“插入记录”菜单中选取“ASP.NET 对象”中的“ASP:按钮”, 将其插入网页中, 出现如图 5.10 所示的对话框。设置按钮的面板文本为“确定”, 点击“确定”按钮, 在网页中插入按钮控件。



图 5.10 按钮控件的设计界面

按钮控件的设计中, 定义事件是非常重要的步骤。在 DW CS3 中, 按钮的标签编辑器提供了事件名称的设定。点击按钮“属性”面板的标签编辑器图标, 弹出如图 5.11 所示的标签编辑器窗口。另一种打开标签编辑器的方法是, 选中设计窗口中的按钮, 点击鼠标右键, 在

弹出菜单中选择“编辑标签”，也会出现标签编辑器窗口。选择窗口左半部分的“OnClick”事件，在窗口右半部分的空白处输入事件名称“btnClick”。点击“确定”按钮后，在网页中生成如下代码：

```
<asp:Button ID="Button1" runat="server" Text="确定" OnClick="btnClick" />
```

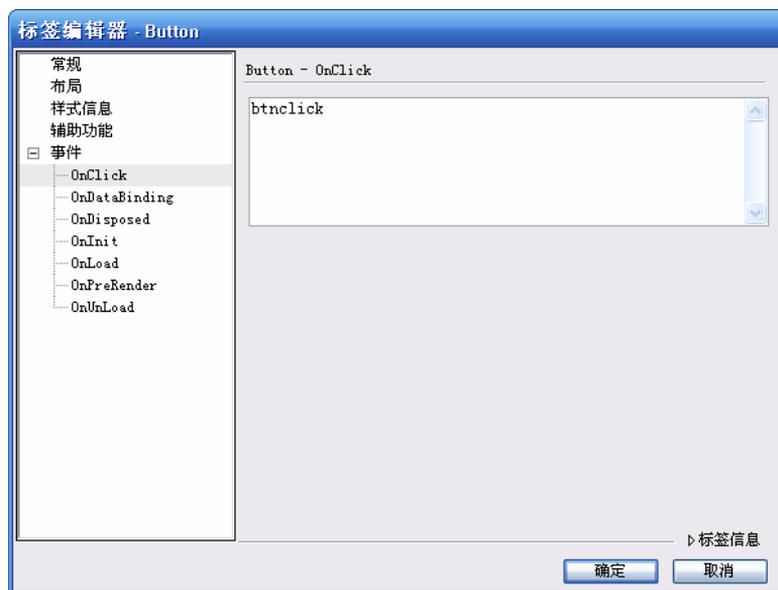


图 5.11 按钮控件的标签编辑器

上面的代码只是在按钮中声明了事件名称，事件的处理代码要另外在过程中定义。下面的例子说明了如何定义按钮和事件。

例 5-1（5-1.aspx）设计一个计算成绩的页面，输入三项成绩，当用户点击“汇总”按钮后，显示总成绩。

（1）首先在“D:\Book”目录中新建一个子目录 Controls（即“D:\Book\Controls”），本章的所有实例均保存在该目录下。启动 DW CS3，在站点的“Controls”子目录下，新建一个 ASP.NET 网页，将其命名为 5-1.aspx，并将文档窗口切换到“拆分”视图。

（2）将光标移至“拆分”视图的设计窗口中，点击“插入记录”→“ASP.NET 对象”→“ASP:标签”，在弹出的标签设计窗口中，将标签控件的文本设置为“计算机：”，如图 5.12 所示。点击“确定”按钮。可以看到，在设计窗口增加了“计算机：”输入提示，在代码窗口增加了<asp:label>标记。

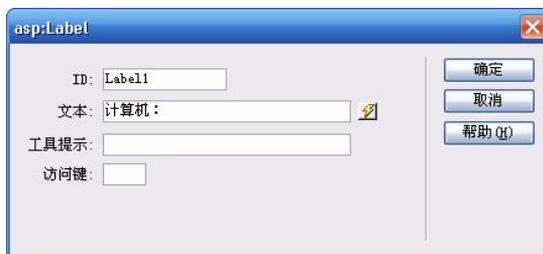


图 5.12 例 5-1 中的“计算机：”标签设计界面

(3) 将光标移至设计窗口的“计算机:”标签旁, 点击“插入记录”→“ASP.NET 对象”→“ASP:文本框”, 在弹出的标签设计窗口中, 可以看到文本框的控件 ID 为“TextBox1”, 点击“确定”按钮。可以看到设计窗口的“计算机:”后, 出现了一个文本框, 如图 5.13 所示。

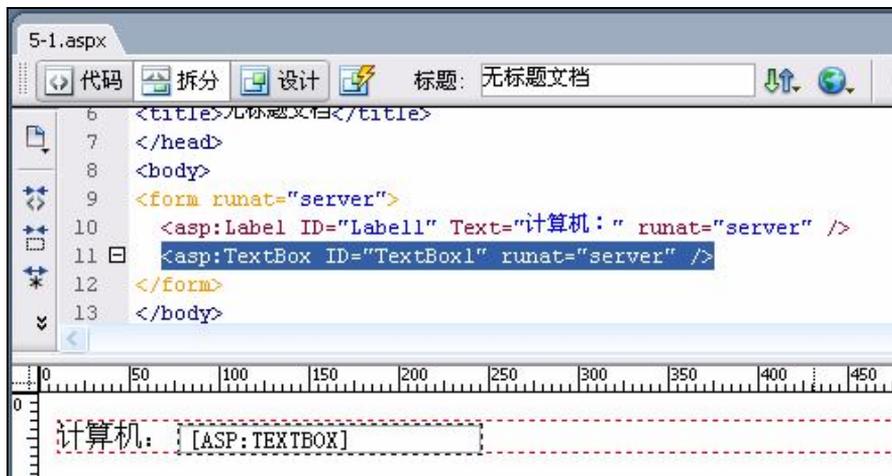


图 5.13 例 5-1 中输入“计算机:”和文本框后的界面

(4) 将光标移至设计窗口的文本框旁, 按回车键换行。

(5) 直接输入“英语:”, 然后从“插入记录”→“ASP.NET 对象”菜单增加“ASP:文本框” TextBox2。

(6) 仿照步骤(4)、(5), 再增加一行“数学:”及相应的文本框 TextBox3, 在“密码:”标签旁增加一个文本框 TextBox3。

(7) 下面在网页中插入按钮控件。将光标移至设计窗口的 TextBox3 文本框旁, 按回车键换行。点击“插入记录”→“ASP.NET 对象”→“ASP:按钮”, 将按钮控件的文本设置为“汇总”, 点击“确定”按钮, 在网页中插入“汇总”按钮。

(8) 选中“汇总”按钮, 点击鼠标右键, 在弹出菜单中选择“编辑标签”, 打开“汇总”按钮的标签编辑器, 选择其中的 OnClick 事件, 在标签编辑器的空白编辑处录入按钮的事件名称“click1”, 点击“确定”按钮, 关闭标签编辑器。

(9) 将光标移至“汇总”按钮旁, 仿照步骤(7)、(8), 增加“取消”按钮, 事件名称设置为“click2”, 点击“确定”按钮。

(10) 按回车键换行, 再添加一个标签按钮, 用于显示总成绩。点击“插入记录”→“ASP.NET 对象”→“ASP:标签”, 点击“确定”按钮, 在网页中增加一个标签。至此, 界面设计完成。上述步骤生成的 ASP.NET 代码及界面如图 5.14 所示。

下面编写事件处理代码。

(11) 将光标移到“拆分”视图中代码窗口的</head>和<body>标记中, 在代码窗口录入图 5.15 中第 8 行到第 22 行所示的代码, 并存盘。

(12) 点击文档窗口的地球图标, 在浏览器中观察运行结果。分别输入三门课程的成绩, 点击“汇总”按钮后, 查看计算结果, 如图 5.16 所示。

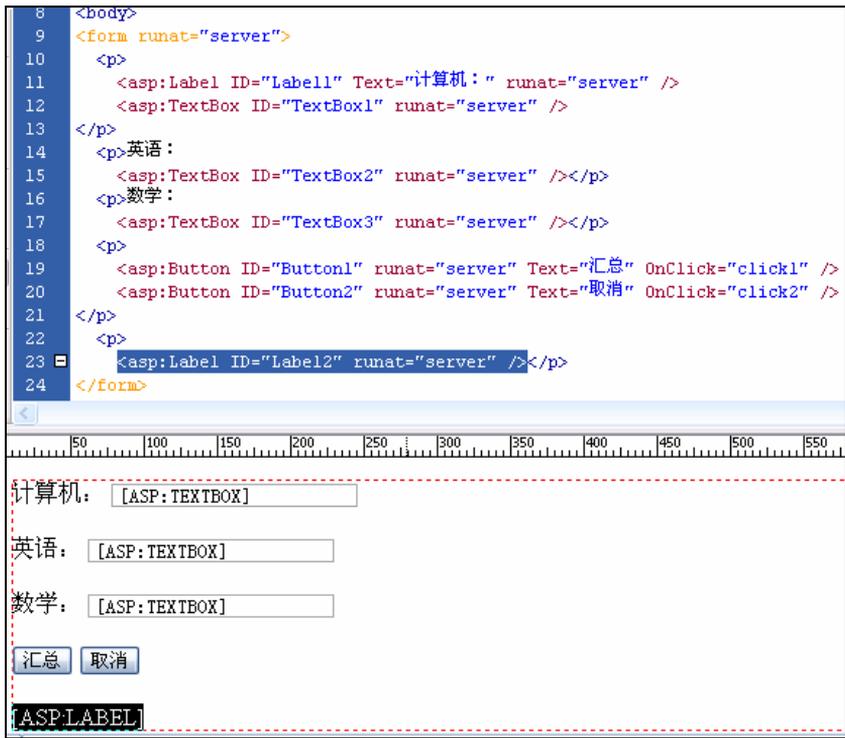


图 5.14 例 5-1 的设计窗口

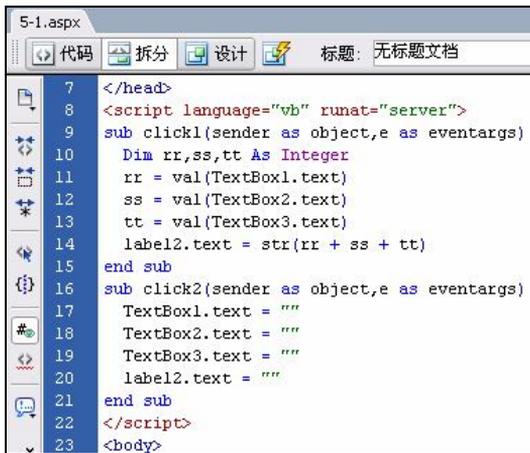


图 5.15 例 5-1 的事件处理代码



图 5.16 例 5-1 的运行结果

代码说明：

① 在<script>的开始和结束标记内，是两个过程 click1 和 click2。每个过程都带有两个参数：Object 类型的参数 Sender 和 EventArgs 类型的参数 E。Sender 一般表示事件的发出控件，E 表示此事件的参数。不同的事件，E 可能有所不同。这两个参数非常重要，在用 ASP.NET 设计事件处理方法的时候，一般都要提供这两个参数。

② Click1 过程实现成绩计算，用 val(textbox1.text)将文本框中的文本内容转换为数值，以便进行加法计算。用 str(rr+ss+tt)将数值转换为文本，赋给标签显示。

③ 用于显示总成绩的标签 ID 是 Label2，所以在代码中对 Label2 的 text 赋值。标签 Label1 显示“计算机：”。

④ Click2 过程是实现输入成绩的文本框和汇总标签的清空。

5.5.2 链接按钮控件<asp:linkbutton>

链接按钮控件的显示效果与链接类似，但是会触发服务器端事件。格式为：

```
<asp:linkbutton id="控件名" runat="server" text="按钮面板上的文字" onClick="事件名"/>
```

链接按钮的属性与事件与普通按钮控件类似，设计时主要是定义按钮的面板文字及事件代码。由于是 linkButton，故链接按钮显示式样不是普通按钮，而是类似一般的超级链接。

例 5-2 (5-2.aspx) 用链接按钮控件 linkButton 代替例 5-1 中的按钮控件，重新设计计算成绩网页。

操作步骤如下：

(1) 在“D:\Book\Controls”目录下，新建一个 ASP.NET 网页，将其命名为 5-2.aspx，将文档窗口切换到“拆分”视图。

(2) 步骤 (2) ~ (10) 同例 5-1，只是插入链接按钮时，不是从“插入记录”菜单的“ASP.NET 对象”菜单中插入“ASP:按钮”，而是从“插入记录”菜单下的“标签”菜单项中，插入“ASP.NET 标签”中的“ASP:LinkButton”，如图 5-17 所示。点击“插入”按钮后，在链接按钮控件的标签编辑器中，继续输入按钮控件的面板文本和事件名称，如图 5.18 所示。

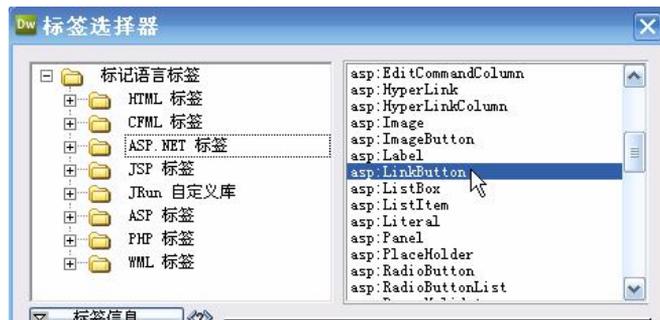


图 5.17 例 5-2 插入 LinkButton 控件



图 5.18 LinkButton 的文本属性和事件

(3) 图 5.19 是使用 LinkButton 控件的计算成绩网页运行效果。可以看到，“汇总”和“取消”按钮显示为链接式样。



图 5.19 例 5-2 的运行效果

5.5.3 图片按钮控件<asp:imagebutton>

图片按钮控件可以实现一个图片形式的按钮，可以触发服务器端事件。格式为：

```
<asp: imagebutton id="控件名" runat="server"  
imageUrl="图片的路径" onClick="事件名"/>
```

在图片按钮控件的属性中，图片的路径 `imageUrl` 是非常重要的。按钮上需要显示的图片是以文件的形式保存在机器中的，只有设定了图片的路径，才能在网页中显示出该图片。另外需要注意的一点是，在 HTML 标记中，图片的路径是通过“`src`”属性给出的，而在 ASP.NET 的服务器控件中，图片路径是通过“`imageUrl`”属性给出的。

在 DW CS3 中，从“插入记录”的“标签”菜单中选取“ASP.NET 标签”中的“ASP:ImageButton”，打开图片控件的标签编辑器，如图 5.20 所示。点击“浏览”按钮，可以选择图片文件的来源。

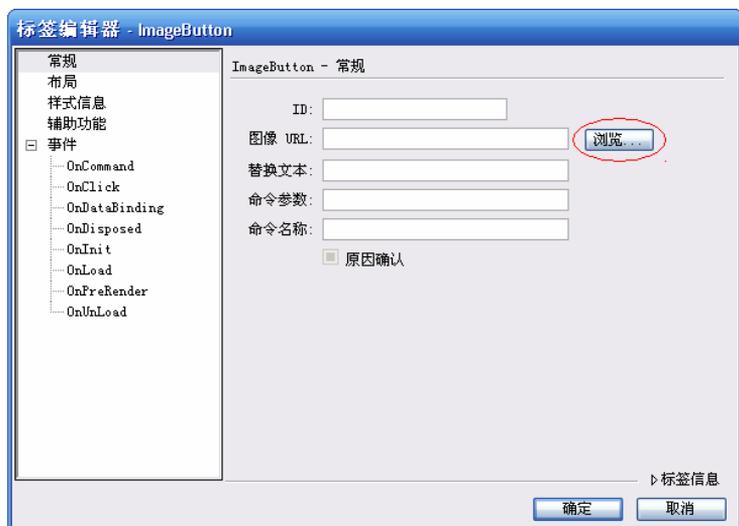


图 5.20 图片控件的标签编辑器

5.5.4 超级链接控件<asp:hyperlink>

Hyperlink 控件用于创建超级链接，可以是文本超级链接，也可以是图片超级链接。用户点击超级链接控件，将会打开另外的网页。格式为：

```

<asp:hyperlink id="控件名" runat="server"
Text="控件上显示的文字"
NavigateUrl="链接的网址"
imageUrl="当用图片代替文本来描述链接时, 图片的路径"
target="打开链接的窗口"/>

```

链接控件的链接网址是通过“NavigateUrl”属性定义的,而在 HTML 标记中,链接网址是通过“Href”属性给出的。如果是文字链接,则要定义“Text”的值;如果是图片链接,则要定义图片的路径“ImageUrl”。Target 属性用于定义打开链接网页的窗口,若不写,则取默认值,即在原窗口打开链接。

在 DW CS3 中,从“插入记录”的“标签”菜单项中选取“ASP.NET 标签”中的“ASP:HyperLink”,可以在网页中插入链接控件。链接控件的标签编辑器如图 5.21 所示。“导航 URL”即“NavigateUrl”,点击旁边的“浏览”按钮,选择要打开的网页文件;“图像 URL”即“ImageUrl”,点击旁边的“浏览”按钮,选择要作为链接源的图片文件。“目标”即“Target”,点击输入框旁的下拉箭头,可以打开下拉列表,用于设置链接网址在什么窗口打开。“_top”表示在没有框架的全窗口中显示链接网页;“_parent”表示在父框架窗口显示链接网页;“_self”表示在超级链接所在的窗口显示链接网页;“_blank”表示在一个新的没有框架的窗口中显示链接网页。

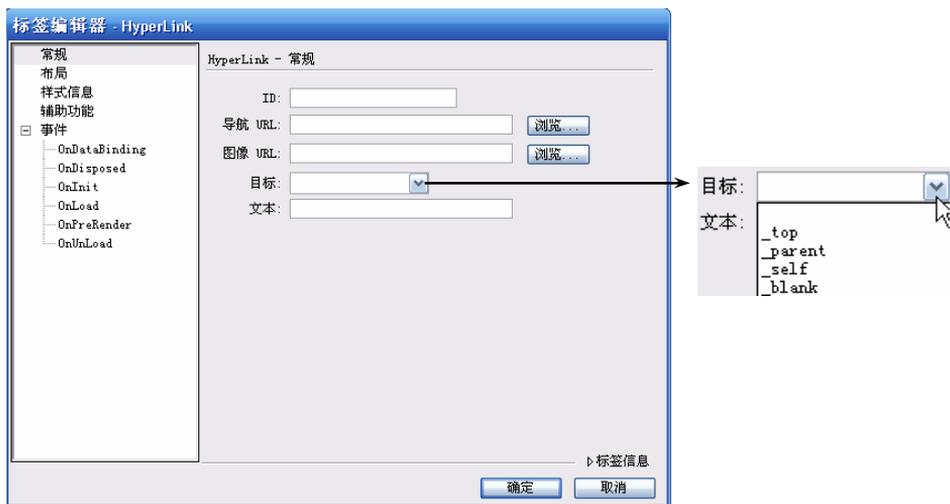


图 5.21 链接控件的标签编辑器

例 5-3 (5-3.aspx) 建立一个超级链接页面,可以分别链接例 5-1、例 5-2。

操作步骤如下:

(1) 在“D:\Book\Examples\Controls”目录下,新建一个 ASP.NET 网页,将其命名为 5-3.aspx。

(2) 在文档窗口的“拆分”视图中,插入两个链接控件,分别设置链接文本为:“例 5-1”和“例 5-2”,并设置“导航 URL”为 5-1.aspx 和 5-2.aspx。设计完成后的 DW CS3 文档窗口如图 5.22 所示。

(3) 例 5-3 的运行结果如图 5.23 所示。点击各链接,会执行相应的 ASP.NET 页面。



图 5.22 例 5-3 的设计界面



图 5.23 例 5-3 的运行结果

5.6 用于选择的控件

用于选择的控件主要包括复选框、单选按钮和下拉列表。其中，下拉列表是 Web 服务器控件中新增加的。Web 服务器中的复选框有两种：单个复选框 `CheckBox` 以及复选框列表 `CheckBoxList`。单选按钮也有两种：单个单选按钮 `RadioButton` 以及单选按钮列表 `RadioButtonList`。复选框列表、单选按钮列表和下拉列表是 ASP.NET 中新增加的控件，功能更为强大。

5.6.1 复选框 `<asp:checkbox>`

复选框用于提供一组选项，用户可以在其中选择多项，功能类似 HTML 中的 `checkbox` 表单项，只是增加了服务器端事件。格式为：

```
<asp:checkbox id="控件名"runat="server"
autoPostBack="是否回发事件 (True|False) "
checked="是否被选中 (True|False) "
text="控件的标题文字"
textAlign="控件的对齐方式 (Right|Left) "
onCheckedChanged="事件响应函数的名字"/>
```

复选框是否被选中，是由 `checked` 属性决定的。如果被选中，则 `checked` 值为 `True`，否则为 `False`。

复选框的事件是 `onCheckedChanged`，这是在选项改变时触发的服务器端事件。默认情况下，`onCheckedChanged` 事件并不会立即被提交到服务器处理。如果将 `AutoPostBack` 属性

设置为 True, 则 onCheckedChanged 事件会立即被送到服务器处理。

例 5-4 (5-4.aspx) 建立一个页面, 用于选择喜欢的课程。

操作步骤如下:

(1) 在“D:\Book\Controls”目录下, 新建一个 ASP.NET 网页, 将其命名为 5-4.aspx, 将文档窗口切换到“拆分”视图。在设计窗口中, 插入三个复选框。复选框的文本分别设置为: “网络技术”、“数据库技术”和“ASP.NET 网页设计”, 并在“自动回发”前的小方框中打勾, 如图 5.24 所示。注意, 插入第二个和第三个复选框时, 要在红色的虚线框(表单标记)内插入。



图 5.24 复选框的设计界面

(2) 在网页的三个复选框之后, 插入一个标签, 用于显示提示信息。

(3) 下面设置复选框的事件。选中第一个复选框, 点击鼠标右键打开复选框的标签编辑器, 在事件 OnCheckedChanged 右边的窗口中输入“chkClick”, 点击“确定”按钮。同样的方法, 把其他两个复选框的 OnCheckedChanged 事件右边的窗口中也输入“chkClick”, 完成后的界面设计代码如图 5.25 所示。

```
<form runat="server">
  <p>
    <asp:CheckBox AutoPostBack="true" ID="CheckBox1" runat="server" Text="网络技术"
    OnCheckedChanged="chkclick" />
  </p>
  <p>
    <asp:CheckBox AutoPostBack="true" ID="CheckBox2" runat="server" Text="数据库技术"
    OnCheckedChanged="chkclick" /></p>
  <p>
    <asp:CheckBox AutoPostBack="true" ID="CheckBox3" runat="server" Text="ASP.NET网页设计"
    OnCheckedChanged="chkclick" /></p>
  <p>
    <asp:Label ID="Label1" runat="server" /></p>
</form>
```

图 5.25 三个复选框和一个标签的设计

(4) 在网页中</head>和<body>标记之间录入如图 5.26 所示的“chkClick”事件处理代码。

```
<script language="vb" runat="server">
  sub chkclick(sender as object, e as eventargs)
    label1.text="您选择了:<br>"
    if checkBox1.checked then label1.text=label1.text & checkBox1.text & "<br>"
    if checkBox2.checked then label1.text=label1.text & checkBox2.text & "<br>"
    if checkBox3.checked then label1.text=label1.text & checkBox3.text & "<br>"
  end sub
</script>
```

图 5.26 复选框的事件处理代码

(5) 在浏览器中查看，运行结果如图 5.27 所示。



图 5.27 例 5-4 的运行结果

代码说明:

① 复选框的 `autoPostBack` 属性设为 `True`，因此，选项一旦改变，立即触发服务器端事件。

② 三个复选框的服务器端事件均指定为“`chkClick`”。在这个服务器的事件处理代码中，首先将标签的文本初始化为“您选择了: `
`”，然后，根据复选框的 `checked` 属性是否为“`True`”，判断选项是否被选中。若被选中，则更新标签控件的显示内容。

③ 在对标签控件的文本赋值中，除初始化标签值时是直接赋值外，其余的三个标签赋值表达式中都用到了字符串连接运算符“`&`”，用于将标签原有的文本和被选中的复选框文本内容相连接，然后赋值给标签的文本。这样，多个项目被选中时，选取结果都可以在标签中显示出来。

5.6.2 复选框列表`<asp:checkboxlist>`

复选框列表是一个包含若干复选框的控件组。格式为:

```
<asp:checkboxList id="控件名" runat="server"
autoPostBack="是否回发事件 (True|False) "
textAlign="控件的对齐方式 (Right|Left) "
cellPadding="控件显示时各个选项之间的距离值"
dataSource="<% 数据绑定表达式 %>"
dataTextField="绑定字段"
dataValueField="绑定字段"
repeatColumns="列表项的列数"
repeatDirection="列表项的方向 (Vertical|Horizontal) "
repeatLayout="列表项的呈现方式 (Flow|Table) "
onSelectedIndexChanged="事件响应函数的名字">
<asp:listitem value="列表项的关联值"
Text="列表项的文本"
selected="是否被选中 (True|False) "/>
</asp:checkboxList>
```

复选框列表的代码块中, 内嵌的<asp:listitem>表示每一个复选框成员。当 autoPostBack 属性为 True 时, 选项变化时立即触发服务器端事件 onSelectedIndexChanged。从标签代码可以看到, 复选框列表还可以绑定数据库。

复选框列表中, 所有选项可以用数组 “items” 表示; 被选中的选项可以用 “selected Item” 表示。每一个复选框是否被选中, 是由<asp:listitem>的 “selected” 属性决定的。

例 5-5 (5-5.aspx) 用复选框列表实现课程选择。

操作步骤如下:

(1) 在 “D:\Book\Controls” 目录下, 新建一个 ASP.NET 网页, 将其命名为 5-5.aspx, 将文档窗口切换到 “拆分” 视图。从 “插入记录” 的 “ASP.NET 对象” 菜单中插入 “asp:复选框列表” 控件。选中新插入的复选框列表控件, 在 DW CS3 的扩展属性窗口, 找到复选框列表的 “列表项...” 属性按钮, 如图 5.28 所示。点击 “列表项...” 按钮, 弹出 “列表项” 对话框, 如图 5.29 所示。

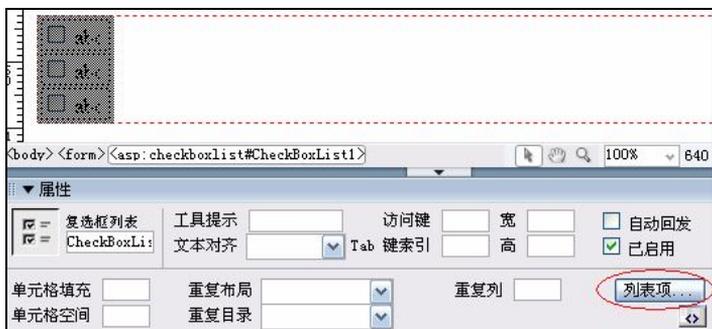


图 5.28 复选框列表控件的属性



图 5.29 “列表项”对话框

(2) 在 “列表项” 对话框的 “标签” 处, 输入要添加的列表项的标签和值的内容, 点击 “+” 号, 文本即可添加到列表项, 分别将 “网络技术”、“数据库技术” 和 “ASP.NET 网页设计” 文本添加到列表项中, 点击 “确定” 按钮, 则复选框列表控件及其列表项设计完成。

(3) 下面定义页面的事件。在复选框列表下增加一个按钮控件, 按钮的文本设为 “提交”, 按钮的 “onClick” 事件定义为 “click”。再在按钮下增加一个标签。设计好的界面代码

如图 5.30 所示。

```
<form runat="server">
  <p>
    <asp:CheckBoxList ID="CheckBoxList1" runat="server">
      <asp:ListItem value="1">网络技术</asp:ListItem>
      <asp:ListItem value="2">数据库技术</asp:ListItem>
      <asp:ListItem value="3">ASP.NET网页设计</asp:ListItem>
    </asp:CheckBoxList>
  </p>
  <p>
    <asp:Button ID="Button1" runat="server" Text="提交" OnClick="click" />
  </p>
  <p>
    <asp:Label ID="Label1" runat="server" />
  </p>
</form>
```

图 5.30 例 5-5 的界面代码

(4) 在网页中增加按钮事件的处理代码，如图 5.31 所示。

```
<script language="vb" runat="server">
  sub click(sender as object, e as EventArgs)
    dim i as integer
    label1.text="您选择了:<br>"
    for i = 0 to checkBoxList1.items.count-1
      if checkBoxList1.items(i).selected then
        label1.text=label1.text & checkBoxList1.items(i).text & "<br>"
      end if
    next
  end sub
end sub
</script>
```

图 5.31 提交按钮的事件代码

(5) 在浏览器中查看，运行结果如图 5.32 所示。

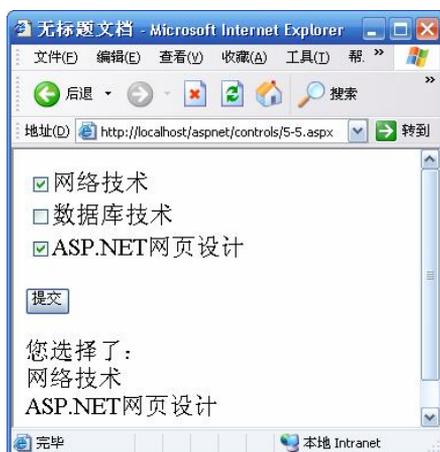


图 5.32 例 5-5 的运行结果

代码说明：

① 例 5-5 是利用按钮事件实现服务器端事件的。当选择了“提交”按钮后，在提交按钮的“click”事件中，检查哪个复选框被选中，并通过标签控件给出选中的复选框的文本内容。

② 在按钮事件中, 通过 For/Next 循环判断哪个复选框被选中。checkboxList 控件的 Items 属性是一个集合, 代表复选框列表中的各个成员。Items 集合的 count 属性表示集合中的元素个数, 循环次数就是集合中的元素的数目。循环变量 i 从 0 开始, 直到循环次数减 1。由于 VB.NET 中的数组下标是从 0 开始的, 因此 Items(i)正好表示了 checkboxList 控件中的各个成员。这些成员的 selected 属性为 True, 表示被选中。这样就利用循环的方式访问了所有的选项。

5.6.3 单选按钮<asp:radioButton>

单选按钮用于规定在一组选项中只能选取一个, 功能类似 HTML 中的 Radio 表单项, 只是增加了服务器端事件。格式为:

```
<asp:radioButton id="控件名" runat="server"
autoPostBack="是否回发事件 (True|False) "
checked="是否被选中 (True|False) "
groupName="控件组的名字"
text="控件的标题文字"
textAlign="控件的对齐方式 (Right|Left) "
onCheckedChanged="事件响应函数的名字"/>
```

类似于复选框控件, 单选按钮是否被选中是通过 checked 属性设定的, 单选按钮中的服务器端事件是 onCheckedChanged。如果 AutoPostBack 属性设置为 True, 则选项改变引起的 onCheckedChanged 事件会立即被送到服务器处理。

与复选框控件不同的是, 单选按钮中增加了 groupName 属性, 用于将单选按钮组成一组。groupName 属性值相同的单选按钮为同一组, 一组单选按钮中只能有一项被选中。不同组的单选按钮, 每一组都能选取一个。

例 5-6 (5-6.aspx) 建立一个页面, 用于选择所在的系别。

操作步骤如下:

(1) 在“D:\Book\Controls”目录下, 新建一个 ASP.NET 网页, 将其命名为 5-6.aspx, 将文档窗口切换到“拆分”视图。在设计窗口中, 插入三个单选按钮控件。单选按钮控件的文本分别设置为: “计算机系”、“电子系”和“经济系”, 并将“自动回发”前的小方框中选中, 如图 5.33 所示。



图 5.33 单选按钮界面设计

(2) 分别将三个单选控件“属性”面板的“组名称”设置为“dept”, 如图 5.34 所示。

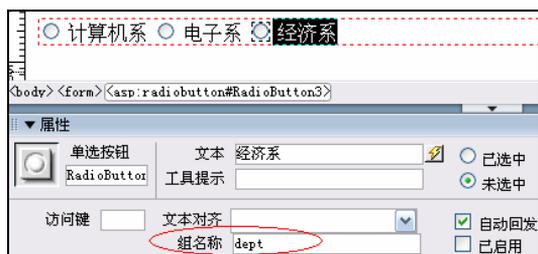


图 5.34 单选按钮的组名称属性

(3) 分别打开三个单选按钮的标签编辑器，将 OnCheckedChanged 事件的名称设置为“radClick”。

(4) 在网页的三个复选框之后，插入一个标签，用于显示提示信息。完成后的界面代码如图 5.35 所示。

```
<form runat="server">
  <p>
    <asp:RadioButton AutoPostBack="true" GroupName="dept" ID="RadioButton1" runat="server" Text="计算机系" OnCheckedChanged="radclick" />
    <asp:RadioButton AutoPostBack="true" GroupName="dept" ID="RadioButton2" runat="server" Text="电子系" OnCheckedChanged="radclick" />
    <asp:RadioButton AutoPostBack="true" GroupName="dept" ID="RadioButton3" runat="server" Text="经济系" OnCheckedChanged="radclick" /> </p>
    <p>
      <asp:Label ID="Label1" runat="server" /></p>
  </form>
```

图 5.35 单选按钮的设计代码

(5) 在网页中加入如图 5.36 所示的事件处理代码。

```
<script language="vb" runat="server">
  sub radclick(sender as object, e as eventargs)
    if radioButton1.checked then labell.text="您在计算机系"
    if radioButton2.checked then labell.text="您在电子系"
    if radioButton3.checked then labell.text="您在经济系"
  end sub
</script>
```

图 5.36 例 5-6 的事件处理代码

(6) 在浏览器中查看网页，其中出现三个系别单选按钮，任意点击一个选项，网页上立即出现所选择的项目，如图 5.37 所示。



图 5.37 例 5-6 的运行结果

代码说明:

① 三个单选按钮控件的组名都是“dept”，因此组成一组单选按钮，一次只能选择一个选项。

② 单选按钮的 autoPostBack 属性设为 True，因此，选项一旦改变，立即触发服务器端事件。

③ 三个选项的服务器端事件均指定为“radClick”。在该过程中，首先根据单选按钮的 checked 属性是否为 True，判断哪个单选按钮被选中。找到后，将该单选按钮的文字内容赋值给标签控件的文本，利用标签控件显示选项的内容。

5.6.4 单选按钮列表<asp:radiobuttonlist>

单选按钮列表是一个包含若干单选按钮的控件组，常用于需要显示多个单选按钮，并且对每个单选按钮都有类似处理方式的情形。格式为：

```
<asp:radioButtonList id="控件名" runat="server"
    autoPostBack="是否回发事件 (True|False) "
    textAlign="控件的对齐方式 (Right|Left) "
    cellPadding="控件显示时各个选项之间的距离值"
    dataSource="<% 数据绑定表达式 %>"
    dataTextField="绑定字段"
    dataValueField="绑定字段"
    repeatColumns="列表项的列数"
    repeatDirection="列表项的方向 (Vertical|Horizontal) "
    repeatLayout="列表项的呈现方式 (Flow|Table) "
    onSelectedIndexChanged="事件响应函数的名字">
    <asp:listitem value="列表项的关联值"
        Text="列表项的文本"
        selected="是否被选中 (True|False) "/>
</asp:radioButtonList>
```

与复选框列表控件类似，在单选按钮列表的代码块中，内嵌的<asp:listitem>表示每一个单选按钮成员。当 autoPostBack 属性为 True 时，选项变化时立即触发服务器端事件 onSelectedIndexChanged。复选按钮列表可以绑定数据库。

与复选框列表控件不同的是，单选按钮列表中仅能有一个选项被选中。

5.6.5 下拉列表控件<asp:dropdownlist>

DropDownList 控件是 ASP.NET 中新增加的控件，以下拉方式显示各选项，称为下拉列表控件。除了显示方式不同外，在事件处理方面，DropDownList 控件与前面介绍的单选按钮列表和复选框列表类似。格式为：

```
<asp:dropDownList id="控件名" runat="server"
    autoPostBack="是否回发事件 (True|False) "
    dataSource="<% 数据绑定表达式 %>"
    dataTextField="绑定字段"
    dataValueField="绑定字段"
    rows="列表项的行数"
```

```

selectionMode="选择模式 (Single|Multiple) "
onSelectedIndexChanged="事件响应函数的名字">
<asp:listitem value="列表项的关联值"
Text="列表项的文本"
selected="是否被选中 (True|False) "/>
</asp:dropDownList>

```

由于是以下拉方式显示各控件，所以 DropDownList 控件就不存在类似单选按钮列表和复选框列表中的排列方式属性了。同样，Items 可用于表示各个选项的集合。

例 5-7 (5-7.aspx) 用下拉列表选择城市。

(1) 在“D:\Book\Controls”目录下，新建一个 ASP.NET 网页，将其命名为 5-7.aspx，将文档窗口切换到“拆分”视图。从“插入记录”的“ASP.NET 对象”菜单中插入一个“asp:下拉列表”控件，在弹出的下拉列表控件设计窗口中将“自动回发”属性选中，点击“确定”按钮关闭窗口。可以看到在设计窗口已生成一个红色虚线框住的下拉列表控件。选中这个下拉列表控件后，展开“属性”面板的扩展属性，点击其中的“列表项...”，在打开的“列表项”对话框中添加下拉列表控件的列表项标签为“北京”、“上海”和“广州”，相应值分别为“1”、“2”和“3”后，关闭“列表项”对话框。打开下拉列表控件的标签编辑窗口，将事件 onSelectedIndexChanged 定义为 change。

(2) 在下拉列表控件下添加一个标签。设计好的界面及代码如图 5.38 所示。

```

<form runat="server">
  <p>
    <asp:DropDownList AutoPostBack="true" ID="DropDownList1" runat="server"
    OnSelectedIndexChanged="change">
      <asp:ListItem value="1">北京</asp:ListItem>
      <asp:ListItem value="2">上海</asp:ListItem>
      <asp:ListItem value="3">广州</asp:ListItem>
    </asp:DropDownList>
  </p>
  <p>
    <asp:Label ID="Label1" runat="server" /></p>
</form>

```

图 5.38 例 5-7 的界面设计代码

(3) 在<head>和<body>标签之间输入如图 5.39 所示的事件处理代码。

```

<script language="vb" runat="server">
  sub change(sender as object,e as eventargs)
    dim i as integer
    label1.text="您选择了：" & DropDownList1.SelectedItem.text
  end sub
</script>

```

图 5.39 例 5-7 的事件处理代码

(4) 在浏览器中查看运行结果。点击下拉列表中的城市名后，页面显示下拉列表中显示的城市，如图 5.40 所示。

代码说明：

① 该例中下拉列表控件设置了 autoPostBack=True，因此，当下拉选项改变时，会自动将 onSelectedIndexChanged 事件传送回服务器，服务器调用 Change 事件对选项改变进行处理。

② 下拉列表控件的 SelectedItem 属性表示被选中的选项, 通过 SelectedItem.text 可以得到被选中选项的标签文本。复选框列表控件和单选按钮列表控件也同样可以通过这一属性获得被选项。



图 5.40 例 5-7 的运行结果

5.7 任务实现: 用 ASP.NET 控件设计用户注册页面

前面几节我们已学习了 ASP.NET 中的各种控件, 通过这些控件, 可以实现网页和用户的一些简单交互。下面我们就利用所学控件, 来设计一个用户注册页面, 并能将用户已输入的信息反馈到浏览器显示。

例 5-8 (5-8.aspx) 设计一个校园网用户注册页面, 要求输入以下内容: 用户名、密码、性别、所在系、爱好、邮箱。

操作步骤如下:

(1) 在“D:\Book\Controls”目录下, 新建一个 ASP.NET 网页, 将其命名为 5-8.aspx, 将文档窗口切换到“设计”视图。在设计窗口添加用户注册页面的控件, 如图 5.41 所示。其中, “性别”选项包含两个单选按钮, “爱好”选项由复选框按钮列表构成, 标签控件用于在点击“提交”按钮后显示用户已输入信息。



图 5.41 用户注册的界面设计

界面的控件所对应的代码如下，只显示表单部分：

```
<form runat="server">
  <p>用户名:
  <asp:TextBox ID="TextBox1" runat="server" />
</p>
  <p>密码:
  <asp:TextBox ID="TextBox2" TextMode="Password" runat="server" /></p>
  <p>请再输入一次:
  <asp:TextBox ID="TextBox3" TextMode="Password" runat="server" /></p>
  <p>性别:
  <asp:RadioButton ID="RadioButton1" Text=" 男 " runat="server" GroupName="sex"
Checked="true" />
<asp:RadioButton ID="RadioButton2" Text="女" runat="server" GroupName="sex" /></p>
  <p>系:
  <asp:DropDownList ID="DropDownList1" runat="server">
    <asp:ListItem value="1">计算机系</asp:ListItem>
    <asp:ListItem value="2">电子系</asp:ListItem>
    <asp:ListItem value="3">经济系</asp:ListItem>
  </asp:DropDownList>
  <p>
  <p>爱好:
  <asp:CheckBoxList ID="CheckBoxList1" runat="server" RepeatDirection="Horizontal">
    <asp:ListItem value="1">阅读</asp:ListItem>
    <asp:ListItem value="2">运动</asp:ListItem>
    <asp:ListItem value="3">音乐</asp:ListItem>
    <asp:ListItem value="4">旅游</asp:ListItem>
  </asp:CheckBoxList>
  <p>
  <p>邮箱:
  <asp:TextBox ID="TextBox4" runat="server" />
</p>
  <p>
  <asp:Button ID="Button1" runat="server" Text="提交" OnClick="click1" />
  <asp:Button ID="Button2" runat="server" Text="取消" OnClick="click2" /> </p>
</p>
  <p>
  <asp:Label ID="Label1" runat="server" /></p>
</form>
```

(2) 下面编写事件处理代码。将光标移到“代码”视图窗口的</head>和<body>标记中，录入如下代码后存盘。

```
<script language="vb" runat="server">
Sub click1(sender As Object, e As EventArgs)
    Dim s, temp As String
    Dim i,j As Integer

    s = "欢迎你,"
    s = s & TextBox1.Text

    If RadioButton1.checked Then
        s = s & "先生"
```

```

ElseIf RadioButton2.checked Then
    s = s & "女士"
End If

If TextBox2.Text<>TextBox3.Text Or TextBox2.Text = Nothing Then
    Label1.Text = "<font color=red>密码为空或两次密码不一样，请重新输入! </font"
    TextBox2.Text = ""
    TextBox3.Text = ""
    return
End If

s = s & "。你的系是: " & DropDownList1.SelectedItem.Text

j 用来统计输入爱好的数量
For i = 0 to CheckBoxList1.Items.count-1
    If CheckBoxList1.Items(i).Selected Then
        temp = temp & CheckBoxList1.Items(i).Text & " "
        j += 1
    End If
Next

j=0 表明没有输入爱好，则不输出
If j > 0 Then
    s = s & "。你的爱好是: " & temp & "。"
End If

s = s & "你的邮箱是: " & TextBox4.Text
Label1.Text = s
End Sub

Sub click2(sender As Object, e As EventArgs)
    Dim i As Integer
    TextBox1.Text = ""

    RadioButton1.checked = True
    RadioButton2.checked = False

    TextBox2.Text = ""
    TextBox3.Text = ""

    DropDownList1.SelectedItem.Selected = False

    For i = 0 to CheckBoxList1.Items.count-1
        CheckBoxList1.Items(i).Selected = False
    Next

    TextBox4.Text = ""

```

```
Label1.Text = ""  
End Sub  
</script>
```

(3) 运行结果如图 5.42 所示。



图 5.42 例 5-8 的运行结果

实训

1. 建立一个登录网页，当用户名和密码都为 asp.net 时，输出登录成功提示，否则提示失败。要求在界面设计时用表格排版。
2. 用单选按钮列表<asp:RadioButtonList>替换例 5-5 中的复选框列表，并通过单选按钮列表的自动回送事件，实现课程的选择。
3. 试一试，如果控件没有放在<form runat="server">标记中，会有什么结果？



习题五

1. 简要说明 HTML 表单和 Web 表单之间的区别？
2. 使用 Label 控件有什么好处？
3. 简述所学 ASP.NET 控件的常用属性及用法。