

第5章 数据处理控件 GridView

在ASP.NET中，GridView控件是数据信息列表显示处理控件功能最为强大的控件，当然也是学习的重要内容。本章将详细讲解GridView控件的使用方法。

本章示例代码参考：[/Code/ch05](#)。

5.1 GridView控件功能概述

学习目标

了解GridView功能。

显示表格数据是软件开发中的一个周期性任务。ASP.NET 提供了许多工具来在网格中显示表格数据，而 GridView 控件是应用最为广泛且功能最强大的控件。

通过使用 GridView 控件，可以显示、编辑和删除多种不同的数据源（例如数据库、XML 文件和公开数据的业务对象）中的数据，可以通过数据源控件自动绑定和显示数据，可以通过数据源控件对数据进行选择、排序、分页、编辑和删除，另外，还可以通过样式自定义 GridView 控件的外观，通过处理GridView的事件自定义 GridView 控件的行为。而所有这些操作基本上都是通过鼠标单击完成的，也就是说，初学者也能够很快学会使用GridView控件。

5.2 创建数据库

学习目标

了解数据库操作。

由于GridView是一个功能强大的数据处理控件，所以在使用前，首先要准备一个数据库，这里给数据库命名为mydb，在其中创建一张数据表test，表结构如图5-1所示。表中的数据可以参照ch05/test.sql脚本。

	列名	数据类型	允许 Null 值
id		int	<input type="checkbox"/>
name		varchar(50)	<input checked="" type="checkbox"/>
sex		char(2)	<input checked="" type="checkbox"/>
city		varchar(50)	<input checked="" type="checkbox"/>
birthday		datetime	<input checked="" type="checkbox"/>
note		varchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

图5-1 数据表test的表结构

5.3 使用GridView列表显示数据

学习目标

掌握GridView使用方法。

(1) 启动Visual Studio 2012，依次选择“文件”|“新建”|“网站”命令，创建一个ASP.NET空网站项目（C#），命名为Study_GridView_Control。

(2) 然后选择“文件”|“新建”|“文件”命令，弹出如图5-2所示的对话框，在“名称”文本框中输入gv01.aspx，并启用“将代码放在单独的文件中”复选框，单击“添加”按钮，完成以后显示如图5-3所示。



图5-2 添加新项-gv01.aspx对话框

(3) 在图5-3中, 选择下方的“设计”按钮, 切换到gv01.aspx的设计窗口, 然后从工具箱的“数据”选项卡中拖动GridView控件到gv01.aspx的设计窗口中, 如图5-4所示。



图5-3 页面gv01.aspx脚本显示窗口



图5-4 页面gv01.aspx设计窗口

(4) 在图5-4中, 选择“GridView任务”面板中“选择数据源”下拉列表的“新建数据源”项目, 弹出“数据源配置向导”对话框, 如图5-5所示。



图5-5 数据源配置向导-选择数据源类型对话框

(5) 在图5-5中选择“数据库”选项，系统自动生成一个ID并命名为SqlDataSource1，也可以修改成其他名称，然后单击“确定”按钮，显示如图5-6所示。

(6) 在图5-6中，单击“新建连接”按钮，弹出“选择数据源”对话框，如图5-7所示。

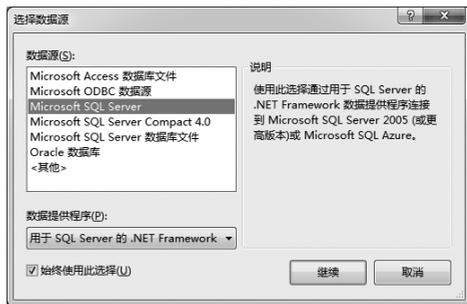
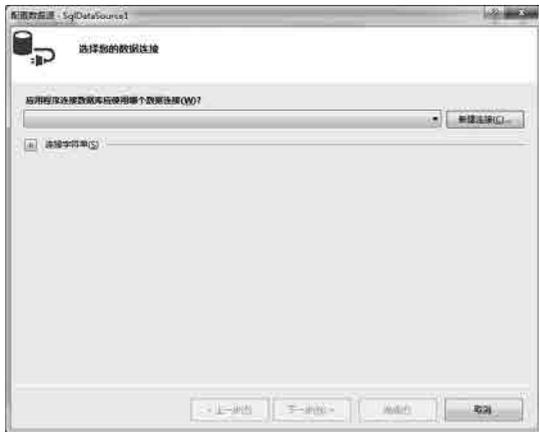


图5-6 配置数据源-选择您的数据连接对话框 (1) 图5-7 “选择数据源”对话框

(7) 在图5-7中，选择“Microsoft SQL Server”项目，然后单击“继续”按钮，弹出如图5-8所示的对话框。

(8) 在图5-8中，在“服务器名”文本框中输入SQL Server数据库服务器名称或IP，一般本地数据库就输入本机器名称，如HST04_PC，选择“使用SQL Server身份验证”单选按钮，账号输入sa，口令输入自己设置的口令，在“选择或输入数据库名称”下面的文本框中输入操作的数据库名称，如mydb，如图5-9所示。



图5-8 “添加连接”对话框 (1)

图5-9 “添加连接”对话框 (2)

(9) 在图5-9中，单击“测试连接”，弹出如图5-10所示提示框。



图5-10 测试连接提示对话框

(10) 然后在图5-9中，单击“确定”按钮，显示如图5-11所示。

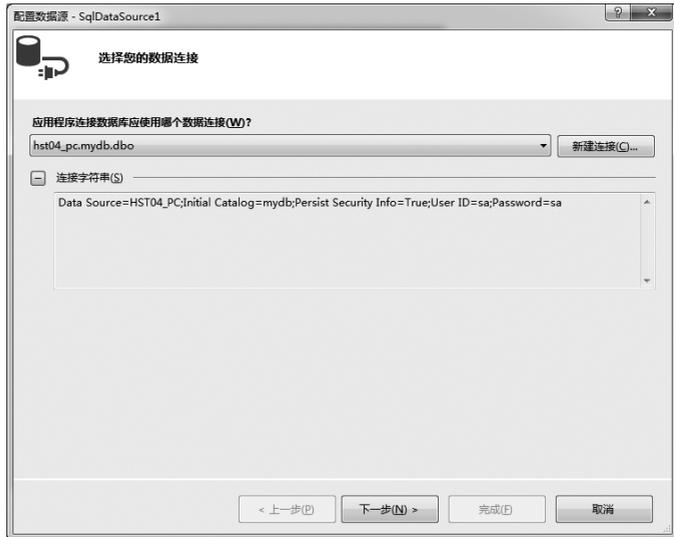


图5-11 配置数据源-选择您的数据连接对话框 (2)

(11) 在图5-11中，单击“下一步”按钮，显示如图5-12所示。

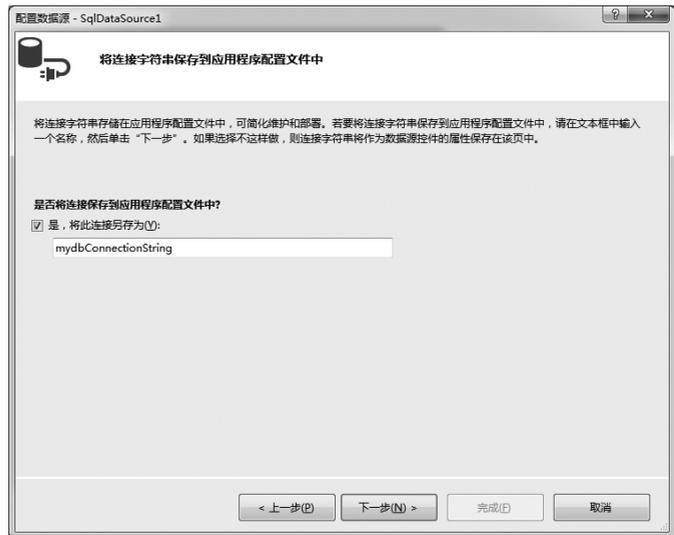


图5-12 配置数据源-保存连接到配置文件对话框

(12) 在图5-12中, 单击“下一步”按钮, 显示如图5-13所示, 在“名称”下拉列表中可以选操作表名称, 这里选择test表, 列选项默认为“*”, 选择具体所需的字段, 当然也可以选择右侧的“WHERE”条件设置、“ORDER BY”排序规则, 其中“高级”选项是GridView控件具备编辑、删除功能才有用的, 后续会讲到。



图5-13 配置数据源-配置Select语句对话框

(13) 在图5-13中, 单击“下一步”按钮, 在显示的对话框中单击“测试查询”按钮, 显示如图5-14所示。



图5-14 配置数据源-测试查询对话框

(14) 在图5-14中, 单击“完成”按钮, gv01.aspx设计窗口显示如图5-15所示。

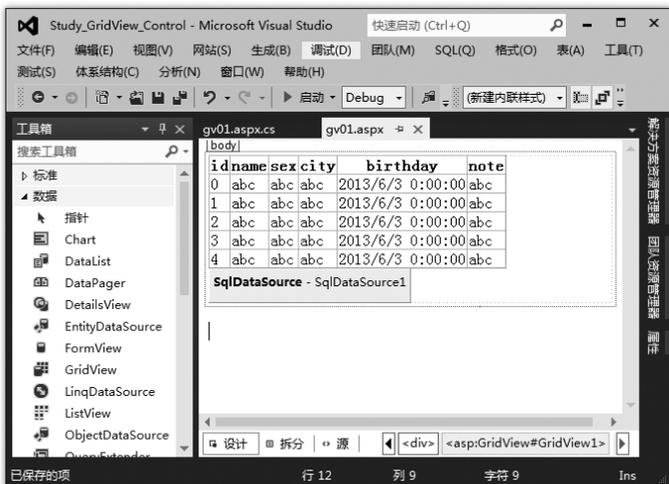


图5-15 GridView加载数据源后的设计窗口

(15) 在图5-15中右击，在弹出的快捷菜单中选择“在浏览器查看”命令，即可看到运行结果如图5-16所示。

到此为止，一个简单的使用GridView控件列表显示数据的例子就完成了。

id	name	sex	city	birthday	note
1	张三	男	北京	1995/5/14 0:00:00	无
2	李四	男	上海	1992/1/4 0:00:00	无
3	王曉慧	女	北京	1994/12/5 0:00:00	无
4	张亮	男	北京	1993/2/5 0:00:00	无
5	石贵	男	天津	1991/2/24 0:00:00	无
6	贾小西	女	上海	1993/7/25 0:00:00	无
7	李波	男	广州	1993/5/23 0:00:00	无
8	刘一平	男	广州	1994/4/9 0:00:00	无
9	刘越	男	北京	1990/10/8 0:00:00	无
10	李军	男	北京	1991/7/21 0:00:00	无
11	肖红莉	女	北京	1995/4/8 0:00:00	无
12	刘明	女	广州	1993/12/29 0:00:00	无
13	贾小贝	男	天津	1990/5/9 0:00:00	无
14	江南	女	北京	1990/7/20 0:00:00	无
15	魏小丽	女	天津	1994/4/4 0:00:00	无
16	李维明	男	北京	1990/6/27 0:00:00	无
17	张丽	女	上海	1994/6/1 0:00:00	无
18	许明双	男	北京	1993/7/28 0:00:00	无
19	张强	男	北京	1995/3/3 0:00:00	无
20	孟婷婷	女	深圳	1994/11/1 0:00:00	无

图5-16 页面gv01.aspx运行结果展示(1)

5.4 修改列标题

学习目标

了解GridView列操作。

(1) 在图5-15中，单击页面的GridView控件，然后单击GridView控件右上角的图标，显示如图5-17所示。



图5-17 “GridView任务”面板

(2) 在图5-17中，选择“编辑列”项目，在弹出的对话框中，在“选定的字段”中选择“id”字段，在右侧“外观”的“HeaderText”中输入“编号”，如图5-18所示，其他字段依此类推，全部修改完成后，显示如图5-19所示。

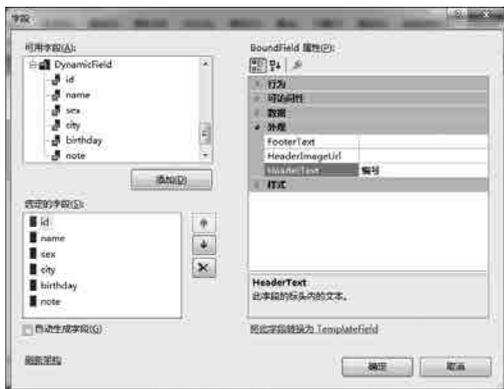


图5-18 GridView编辑列对话框(1)

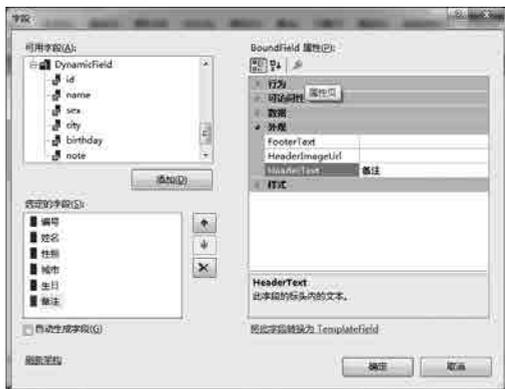


图5-19 GridView编辑列对话框(2)

(3) 在图5-19中,单击“确定”按钮,gv01.aspx设计窗口显示如图5-20所示。

(4) 在图5-20中右击,在弹出的快捷菜单中选择“在浏览器查看”命令,即可看到运行结果如图5-21所示。

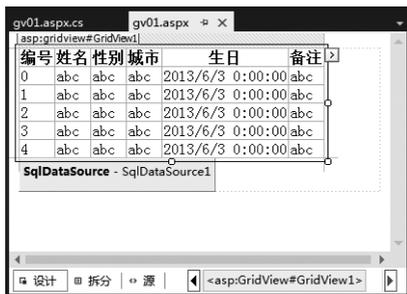


图5-20 页面gv01.aspx设计窗口
(修改字段标题后)



图5-21 页面gv01.aspx运行
结果展示(2)

5.5 启用分页和排序功能

学习目标

了解GridView分页和排序功能。

(1) 再次打开图5-17所示“GridView任务”面板,在图中选择“启用分页”、“启用排序”、“启用选定内容”复选框,如图5-22所示。

(2) 在图5-22中右击,在弹出的快捷菜单中选择“在浏览器查看”命令,即可看到运行结果如图5-23所示,其中的分页功能、排序功能已经实现了,而选择功能其实也已经实现了,看不到效果是由于没有设置选择项模板的样式定义,后续会讲到此内容。



图5-22 GridView启用分页等功能设计窗口



图5-23 页面gv01.aspx运行结果展示(3)

5.6 使用自动套用格式

学习目标

了解GridView自动套用格式功能。

(1) 在图5-17中，选择“自动套用格式”项目，弹出如图5-24所示对话框。

(2) 在图5-24中，选择“沙滩和天空”架构，然后单击“确定”按钮、在gv01.aspx的设计窗口中选择页面上的GridView控件，然后在右侧的“属性”窗口中设置GridView控件的字体大小为12px，如图5-25所示。

(3) 在图5-25中右击，在弹出的快捷菜单中选择“在浏览器查看”命令，即可看到运行结果如图5-26所示。



图5-24 “自动套用格式”对话框

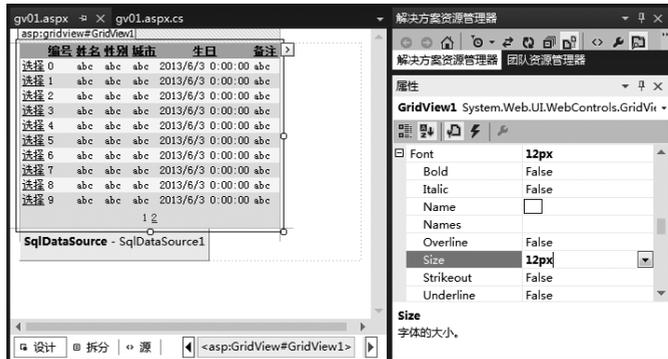


图5-25 页面gv01.aspx设计窗口（设置自动套用格式）



图5-26 页面gv01.aspx运行结果展示（4）

5.7 使用GridView编辑、删除数据操作

学习目标

了解使用GridView维护数据操作。

要使用GridView控件实现编辑、删除功能，首先必须保证操作的表已经设置了主键Primary key。接下来给出操作步骤。

(1) 在网站项目Study_GridView_Control中添加新页面gv02.aspx，切换到页面的设计窗口，从工具箱中拖动一个GridView控件到页面上，如图5-27所示。



图5-27 页面gv02.aspx设计窗口

(2) 在图5-27中，在“GridView任务”面板中的“选择数据源”下拉列表中选择“新建数据源”，弹出如图5-28所示对话框。



图5-28 数据源配置向导-选择数据源类型对话框

(3) 在图5-28中，选择“数据库”选项，然后单击“确定”按钮，弹出如图5-29所示的对话框。

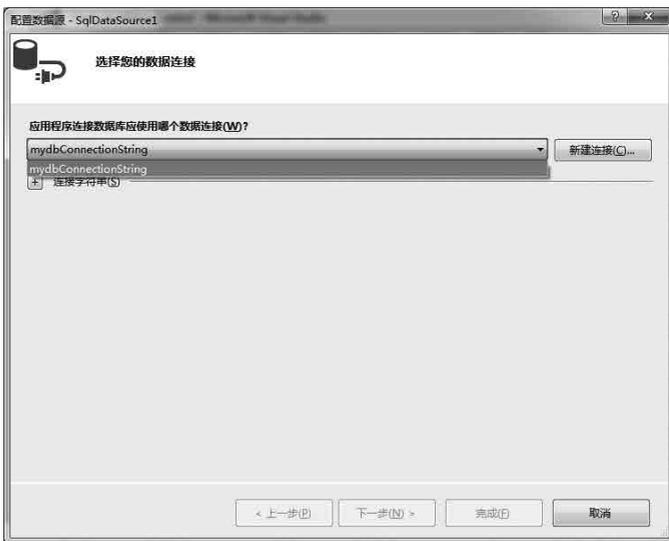


图5-29 配置数据源-选择您的数据连接对话框

(4) 由于之前已经保存了连接字符串，所以本次就直接选择 mydbConnectionString 即可，不用再新建连接字符串了。单击“下一步”按钮，显示如图5-30所示。



图5-30 配置数据源-配置Select语句对话框

(5) 在图5-30中，单击“ORDER BY”按钮，弹出如图5-31所示对话框。

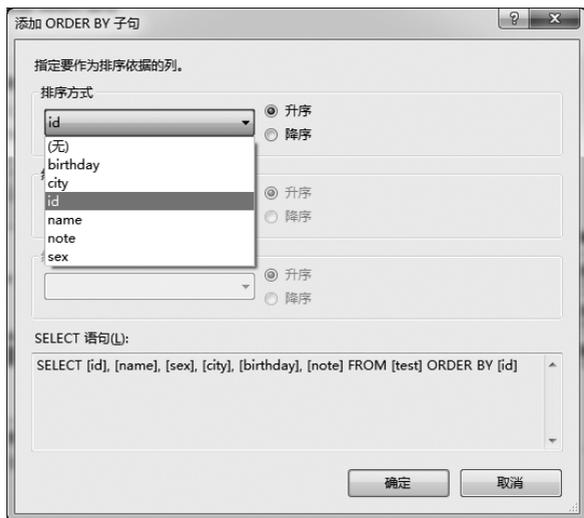


图5-31 “添加ORDER BY子句”对话框

(6) 在图5-31中，选择排序方式为“id”字段且为“升序”模式，然后单击“确定”按钮，就回到了图5-30，然后再单击图5-30中的“高级”按钮，弹出如图5-32所示对话框。



图5-32 “高级SQL生成选项”对话框

(7) 在图5-32中, 启用所有的复选框, 然后单击“确定”按钮, 回到图5-30, 然后单击图5-30中的“下一步”按钮, 弹出“测试查询”对话框, 直接单击“完成”按钮, 就回到了gv02.aspx的设计窗口, 如图5-33所示。

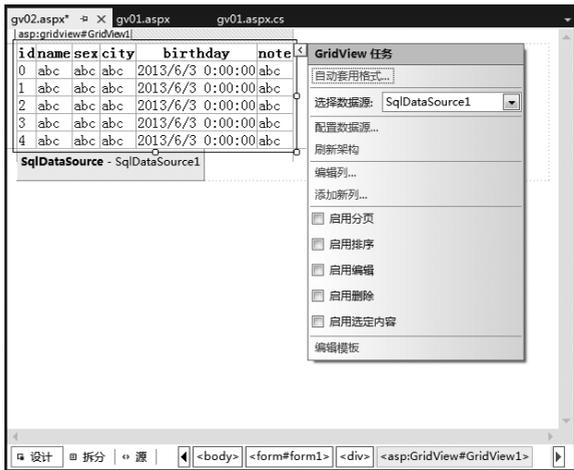


图5-33 页面gv02.aspx设计窗口(完成了高级SQL配置)

(8) 在图5-33中, 选择“启用分页”、“启用排序”、“启用编辑”、“启用删除”, 又回到gv02.aspx设计窗口, 如图5-34所示。

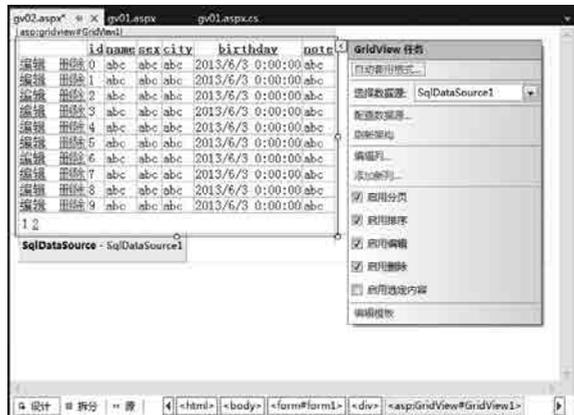


图5-34 页面gv02.aspx设计窗口(启用编辑等功能)

(9) 在图5-34中右击, 在弹出的快捷菜单中选择“在浏览器查看”命令, 即可看到运行结果如图5-35所示。

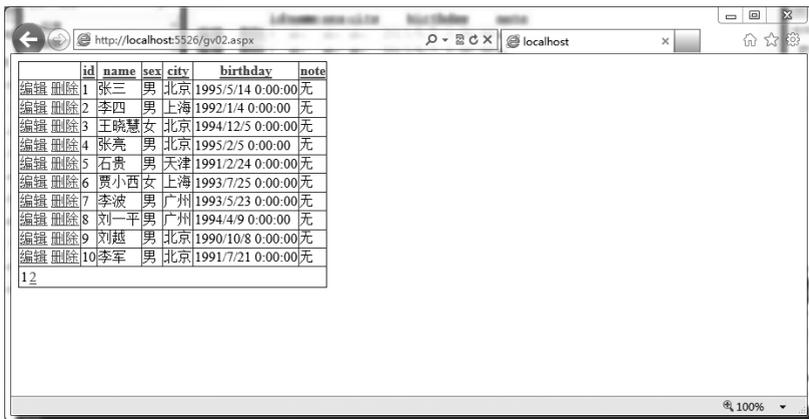


图5-35 页面gv02.aspx运行结果展示(1)

(10) 在图5-35中, 编辑功能、删除功能已经实现了, 但是不够完美, 删除操作直接进行删除, 没有提示功能, 编辑功能显示的表单样子实在不好, 比如性别应该使用单选按钮方式来选择修改。后面章节就来讲述这些问题的处理方法。

5.8 完善GridView删除功能

学习目标

了解GridView编辑列操作。

这里主要说明的是如何给“删除”链接增加一个提示对话框，以便用户在删除信息的时候可以取消删除操作，而不是一单击“删除”链接就直接删除了。

由于GridView控件是微软提供的集成服务器控件，所以不能直接给GridView控件中的“删除”链接增加代码控制，于是GridView提供了模板编辑功能来解决这个问题，而前提是首先要把对应的字段（列）变成模板列。

(1) 在把“编辑”、“删除”字段（列）变成模板列之前先用前面的方法修改一下所有列的标题，修改后的gv02.aspx页面设计窗口如图5-36所示。

(2) 在图5-36中，单击选中页面的GridView控件，然后单击右上角的小按钮，显示出“GridView任务”面板，在其中，继续选择“编辑列”项目，弹出如图5-37所示的对话框。

(3) 选中左侧“选定的字段”中的“维护”字段，然后单击右下方的“将此字段转换为TemplateField”，即可完成模板列的转换，然后单击“确定”按钮，再次回到gv02.aspx页面的设计窗口中。单击选中gv02.aspx页面的GridView控件，然后单击右上角的小按钮，显示出“GridView任务”面板，在其中，继续选择“编辑模板”项目，显示如图5-38所示。

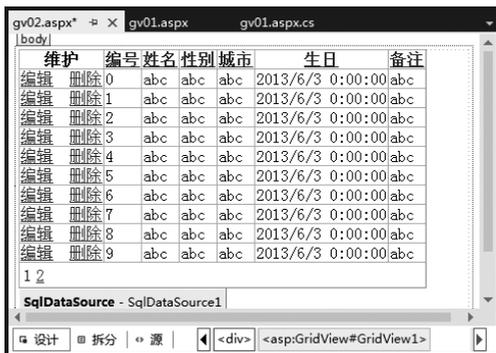


图5-36 页面gv02.aspx设计窗口（修改标题后）

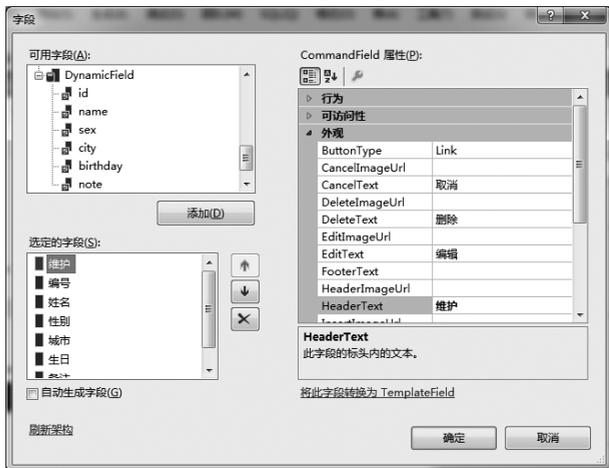


图5-37 GridView编辑列对话框

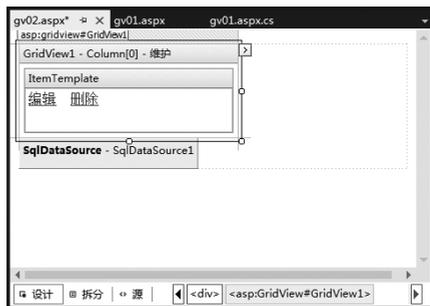


图5-38 编辑模板列窗口

(4) 在图5-38中,右击“删除”按钮,在弹出的快捷菜单中选择“属性”命令,进入右侧的控件属性编辑窗口,如图5-39所示,给“OnClick”属性增加图中所示代码即可,再运行gv02.aspx页面,单击“删除”按钮就会出现提示对话框,如图5-40所示,选择确定就是确认删除了,选择取消就是不删除了。

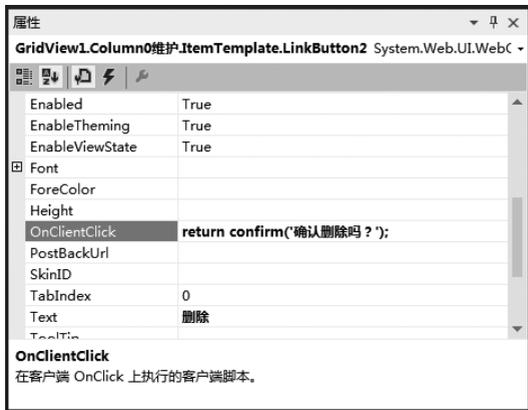


图5-39 GridView中删除按钮属性编辑窗口

图5-40 页面gv02.aspx删除提示对话框

5.9 完善GridView编辑功能

学习目标

进一步熟悉GridView使用。

5.9.1 转换模板列

编辑功能主要是完善编辑模板显示的表单效果,而要自定义编辑模板表单的显示效果,也必须把这些字段列先转换成模板列,转换方法和“维护”列转换为模板列的方法一样,这里就直接给出结果了,如图5-41所示。

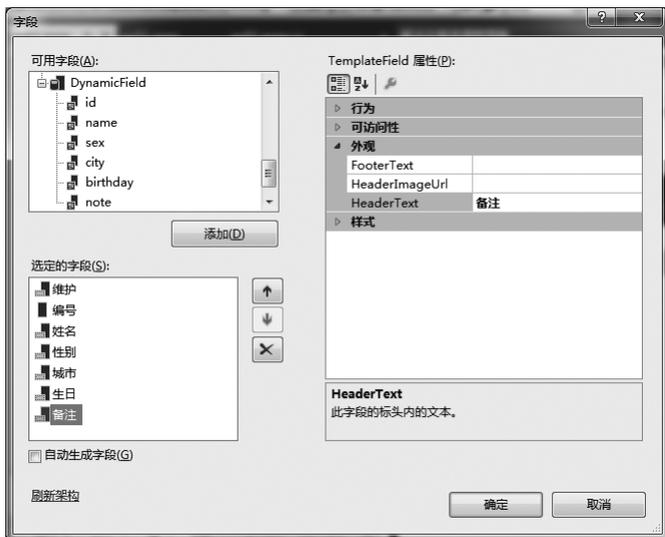


图5-41 修改gv02.aspx所有列都为模板列

其中的编号字段由于是数据表test的主键（Primary Key）字段，故不能修改，也就不需要转换模板列，可以看出，转换模板列后，图5-41中的列标题图标下面多了个小图标。由此可以直观看出哪些列是模板列，哪些不是模板列。

5.9.2 完善系统生成的SqlDataSource代码

(1) 全部字段（列）都转换成模板列之后，再回到gv02.aspx的HTML脚本页面，接下来给出gv02.aspx页面的完整HTML脚本代码。

```
<body>
  <form id="form1" runat="server">
    <div>
      <asp:GridView ID="GridView1" runat="server" AllowPaging=
        "True"AllowSorting="True"AutoGenerateColumns="False"
        DataKeyNames="id"DataSourceID="SqlDataSource1">
        <Columns>
          <asp:TemplateField HeaderText="维护" ShowHeader=
            "False">
            <EditItemTemplate>
              <asp:LinkButton ID="LinkButton1"
                runat="server" CausesValidation="True"
                CommandName="Update" Text="更新"></
                asp:LinkButton>
              &nbsp;<asp:LinkButton ID="LinkButton2"runat
                ="server" CausesValidation="False"Command
                Name="Cancel" Text="取消"></asp:LinkButton>
            </EditItemTemplate>
            <ItemTemplate>
              <asp:LinkButton ID="LinkButton1" runat=
                "server"CausesValidation="False" Command
                Name="Edit" Text="编辑 "></asp:LinkButton>
              &nbsp;<asp:LinkButton ID="LinkButton2"
                runat="server" CausesValidation="False"
                CommandName="Delete" OnClientClick=
                "return confirm('确认删除吗? ');" Text="
                删除"></asp:LinkButton>
            </ItemTemplate>
          </asp:TemplateField>
          <asp:BoundField DataField="id"
            HeaderText="编号" InsertVisible="False" ReadOnly=
            "True"SortExpression="id" />
          <asp:TemplateField HeaderText="姓名"SortExpression
            ="name">
            <EditItemTemplate>
              <asp:TextBox ID="TextBox1" runat="server"
                Text='<%= Bind("name") %>'></asp:TextBox>
            </EditItemTemplate>
            <ItemTemplate>
              <asp:Label ID="Label1" runat="server"
                Text='<%= Bind("name") %>'></asp:Label>
            </ItemTemplate>
          </asp:TemplateField>
        </Columns>
      </asp:GridView>
    </div>
  </form>
</body>
```

```

        </ItemTemplate>
    </asp:TemplateField>
    <asp:TemplateField HeaderText="性别" SortExpression
    ="sex">
        <EditItemTemplate>
            <asp:TextBox ID="TextBox2" runat="server"
            Text='<# Bind("sex") %>'></asp:TextBox>
        </EditItemTemplate>
        <ItemTemplate>
            <asp:Label ID="Label2" runat="server"Text=
            '<# Bind("sex") %>'></asp:Label>
        </ItemTemplate>
    </asp:TemplateField>
    <asp:TemplateField HeaderText="城市"SortExpression
    ="city">
        <EditItemTemplate>
            <asp:TextBox ID="TextBox3" runat="server"
            Text='<# Bind("city") %>'></asp:TextBox>
        </EditItemTemplate>
        <ItemTemplate>
            <asp:Label ID="Label3" runat="server" Text=
            '<# Bind("city") %>'></asp:Label>
        </ItemTemplate>
    </asp:TemplateField>
    <asp:TemplateField HeaderText="生日" SortExpression
    ="birthday">
        <EditItemTemplate>
            <asp:TextBox ID="TextBox4" runat="server"
            Text='<# Bind("birthday") %>'></asp:TextBox>
        </EditItemTemplate>
        <ItemTemplate>
            <asp:Label ID="Label4" runat="server" Text
            ='<# Bind("birthday") %>'></asp:Label>
        </ItemTemplate>
    </asp:TemplateField>
    <asp:TemplateField HeaderText="备注"SortExpression
    ="note">
        <EditItemTemplate>
            <asp:TextBox ID="TextBox5" runat="server"
            Text='<# Bind("note") %>'></asp:TextBox>
        </EditItemTemplate>
        <ItemTemplate>
            <asp:Label ID="Label5" runat="server"Text
            ='<# Bind("note") %>'></asp:Label>
        </ItemTemplate>
    </asp:TemplateField>
</Columns>
</asp:GridView>
    <asp:SqlDataSource ID="SqlDataSource1" runat="server"
    ConflictDetection="CompareAllValues" ConnectionString="<#$ Connectio
    nStrings:mydbConnectionString %>" DeleteCommand="DELETE FROM [test]

```

```

WHERE [id] = @original_id AND (([name] = @original_name) OR ([name]
IS NULL AND @original_name IS NULL)) AND (([sex] = @original_sex)
OR ([sex] IS NULL AND @original_sex IS NULL)) AND (([city] = @
original_city) OR ([city] IS NULL AND @original_city IS NULL)) AND
(([birthday] = @original_birthday) OR ([birthday] IS NULL AND @
original_birthday IS NULL)) AND (([note] = @original_note) OR
([note] IS NULL AND @original_note IS NULL))" InsertCommand="INSERT
INTO [test] ([name], [sex], [city], [birthday], [note]) VALUES (@
name, @sex, @city, @birthday, @note)" OldValuesParameterFormatStrin
g="original_{0}" SelectCommand="SELECT [id], [name], [sex], [city],
[birthday], [note] FROM [test] ORDER BY [id]" UpdateCommand="UPDATE
[test] SET [name] = @name, [sex] = @sex, [city] = @city, [birthday]
= @birthday, [note] = @note WHERE [id] = @original_id AND (([name]
= @original_name) OR ([name] IS NULL AND @original_name IS NULL))
AND (([sex] = @original_sex) OR ([sex] IS NULL AND @original_sex
IS NULL)) AND (([city] = @original_city) OR ([city] IS NULL AND @
original_city IS NULL)) AND (([birthday] = @original_birthday) OR
([birthday] IS NULL AND @original_birthday IS NULL)) AND (([note] =
@original_note) OR ([note] IS NULL AND @original_note IS NULL))">
    <DeleteParameters>
        <asp:Parameter Name="original_id" Type="Int32" />
        <asp:Parameter Name="original_name" Type="String" />
        <asp:Parameter Name="original_sex" Type="String" />
        <asp:Parameter Name="original_city" Type="String" />
        <asp:Parameter Name="original_birthday" Type="Date
Time" />
        <asp:Parameter Name="original_note" Type="String" />
    </DeleteParameters>
    <InsertParameters>
        <asp:Parameter Name="name" Type="String" />
        <asp:Parameter Name="sex" Type="String" />
        <asp:Parameter Name="city" Type="String" />
        <asp:Parameter Name="birthday" Type="DateTime" />
        <asp:Parameter Name="note" Type="String" />
    </InsertParameters>
    <UpdateParameters>
        <asp:Parameter Name="name" Type="String" />
        <asp:Parameter Name="sex" Type="String" />
        <asp:Parameter Name="city" Type="String" />
        <asp:Parameter Name="birthday" Type="DateTime" />
        <asp:Parameter Name="note" Type="String" />
        <asp:Parameter Name="original_id" Type="Int32" />
        <asp:Parameter Name="original_name" Type="String" />
        <asp:Parameter Name="original_sex" Type="String" />
        <asp:Parameter Name="original_city" Type="String" />
        <asp:Parameter Name="original_birthday" Type="Date
Time" />
        <asp:Parameter Name="original_note" Type="String" />
    </UpdateParameters>
</asp:SqlDataSource>
</div>

```

```
</form>
</body>
```

(2) 首先关注加粗部分代码,也就是SqlDataSource部分代码,这里是系统生成的代码,最后能修改一下,例如删除指令:

```
DeleteCommand="DELETE FROM [test] WHERE [id] = @original_id
AND (([name] = @original_name) OR ([name] IS NULL AND @original_
name IS NULL)) AND (([sex] = @original_sex) OR ([sex] IS NULL AND
@original_sex IS NULL)) AND (([city] = @original_city) OR ([city]
IS NULL AND @original_city IS NULL)) AND (([birthday] = @original_
birthday) OR ([birthday] IS NULL AND @original_birthday IS NULL))
AND (([note] = @original_note) OR ([note] IS NULL AND @original_
note IS NULL))"
```

只需要和主键id关联即可,不用和其他字段关联,故上述删除指令代码可以修改为:

```
DeleteCommand="DELETE FROM [test] WHERE [id] = @original_id"
```

再比如修改指令:

```
UpdateCommand="UPDATE [test] SET [name] = @name, [sex] = @sex,
[city] = @city, [birthday] = @birthday, [note] = @note WHERE [id] =
@original_id AND (([name] = @original_name) OR ([name] IS NULL AND
@original_name IS NULL)) AND (([sex] = @original_sex) OR ([sex] IS
NULL AND @original_sex IS NULL)) AND (([city] = @original_city) OR
([city] IS NULL AND @original_city IS NULL)) AND (([birthday] = @
original_birthday) OR ([birthday] IS NULL AND @original_birthday
IS NULL)) AND (([note] = @original_note) OR ([note] IS NULL AND @
original_note IS NULL))"
```

修改指令要修改的项目可能是多个字段,但是关联条件通常也是和主键id关联即可,不用和其他字段关联,故上述修改指令代码可以修改为:

```
UpdateCommand="UPDATE [test] SET [name] = @name, [sex] = @sex, [city] = @
city, [birthday] = @birthday, [note] = @note WHERE [id] = @original_id "
```

由于很少使用GridView完成数据添加功能,故可以把其中的添加指令INSERT直接删除。

后面的代码给出的是删除指令、添加指令、修改指令的参数设置部分代码,由于指令的变化,参数设置部分当然要跟着变了。已经没有添加指令了,故添加指令参数部分就可以完全删除了,而删除部分只和主键id有关,故其他参数可以删除了。而更新操作只和original_id这个原始值有关,故其他original参数就可以删除了。

接下来给出修改后的SqlDataSource部分代码,这样看来代码页不是很复杂了。

```
<asp:SqlDataSource ID="SqlDataSource1" runat="server"
ConflictDetection="CompareAllValues"
ConnectionString="<%$ ConnectionStrings:mydbConnectionString %>"
DeleteCommand="DELETE FROM [test] WHERE [id] = @original_id"
OldValuesParameterFormatString="original_{0}"
SelectCommand="SELECT [id], [name], [sex], [city], [birthday],
[note] FROM [test] ORDER BY [id]"
```

```

UpdateCommand="UPDATE [test] SET [name] = @name, [sex] = @sex,
[city] = @city, [birthday] = @birthday, [note] = @note WHERE [id] =
@original_id">
    <DeleteParameters>
        <asp:Parameter Name="original_id" Type="Int32" />
    </DeleteParameters>
    <UpdateParameters>
        <asp:Parameter Name="name" Type="String" />
        <asp:Parameter Name="sex" Type="String" />
        <asp:Parameter Name="city" Type="String" />
        <asp:Parameter Name="birthday" Type="DateTime" />
        <asp:Parameter Name="note" Type="String" />
        <asp:Parameter Name="original_id" Type="Int32" />
    </UpdateParameters>
</asp:SqlDataSource>

```

(3) 到此为止，系统生成的SqlDataSource代码完善完毕。

5.9.3 修改GridView行高、列宽

为了完善GridView编辑功能的显示效果和运行效果，需要调整GridView的行高和列宽设置，通常最好不要直接拖曳GridView控件改变控件的宽度、高度，这样一来就不好控制了。

1. 设置字体、字体大小

在页面gv02.aspx设计窗口，右击页面上的GridView控件，在弹出的快捷菜单中选择“属性”命令，然后打开GridView控件属性编辑窗口，设置整个GridView控件中的字体、字体大小，如图5-42所示。



图5-42 GridView属性编辑窗口

2. 调整行高

这里的行高，通常分为页眉行高、数据项行高、交错项行高、页脚行高、分页项行高。

对应GridView属性窗口中的如下属性。

- (1) 页眉行高：HeaderStyle中的Height设置为30px。
- (2) 数据项行高：RowStyle中的Height的Height设置为30px。
- (3) 交错项行高：AlternatingRowStyle中的Height的Height设置为30px。
- (4) 页脚行高：FooterStyle中的Height的Height设置为30px。
- (5) 分页项行高：PageStyle中的Height的Height设置为30px。

3. 调整列宽

调整列宽，首先要进入编辑列窗口，然后选择某个列，如编号，如图5-43所示。

在图5-43中，在“样式”的“ItemStyle”中，设置Width为100px，其他列的设置方法依此类推，依次设置如下。

“维护”：Width设置为100px。

“编号”：Width设置为60px。

“姓名”：Width设置为120px。

“性别”：Width设置为80px。

“城市”：Width设置为120px。

“生日”：Width设置为150px。

“备注”：Width设置为150px。

全部设置完成后，页面gv02.aspx运行显示效果如图5-44所示。

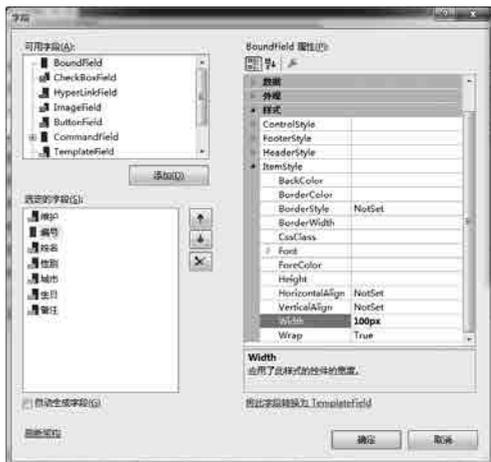


图5-43 编辑列调整列宽对话框

维护	编号	姓名	性别	城市	生日	备注
编辑 删除	1	张三	男	北京	1995/5/14 0:00:00	无
编辑 删除	2	李四	男	上海	1992/1/4 0:00:00	无
编辑 删除	3	王淑慧	女	北京	1994/12/5 0:00:00	无
编辑 删除	4	张亮	男	北京	1995/2/5 0:00:00	无
编辑 删除	5	石勇	男	天津	1991/2/24 0:00:00	无
编辑 删除	6	曹小西	女	上海	1993/7/25 0:00:00	无
编辑 删除	7	李波	男	广州	1993/5/23 0:00:00	无
编辑 删除	8	刘一平	男	广州	1994/4/9 0:00:00	无
编辑 删除	9	刘鹏	男	北京	1990/10/8 0:00:00	无
编辑 删除	10	李军	男	北京	1991/1/21 0:00:00	无

图5-44 页面gv02.aspx运行显示效果（设置行高列宽后）

5.9.4 处理姓名文本框编辑

页面gv02.aspx运行后，单击某行的“编辑”按钮，显示出来的编辑项姓名文本框显然太长了，影响了页面的显示效果，这就需要编辑姓名的编辑项模板，因此，本书上文才详细说明了设置模板列、行高、列宽的方法。

(1) 在gv02.aspx页面的设计窗口，单击页面GridView控件，然后单击右上角的小图标，进入“GridView任务”面板，如图5-45所示。



图5-45 “GridView任务”面板

(2) 在图5-45中，选择“GridView任务”面板最下面的“编辑模板”项目，显示窗口如图5-46所示。

(3) 在图5-46中，选择“Column[2] - 姓名”，显示如图5-47所示，其中ItemTemplate就是数据项（奇数项）模板的定义，AlternatingItemTemplate就是交错项（偶数项）模板的定义，EditItemTemplate就是编辑项模板的定义，HeaderTemplate就是页眉模板的定义，FooterTemplate就是页脚模板的定义，根据以上描述可以看出，这里需要处理的就是EditItemTemplate。

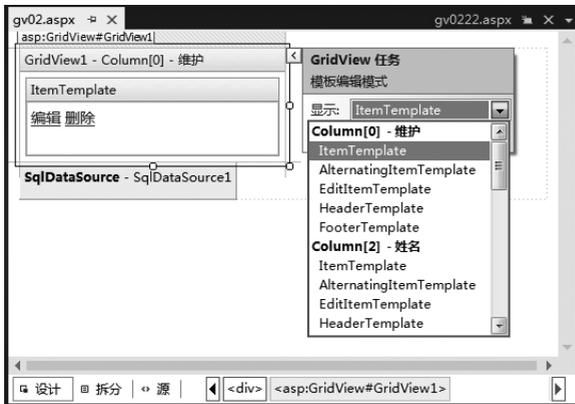


图5-46 模板编辑模式窗口

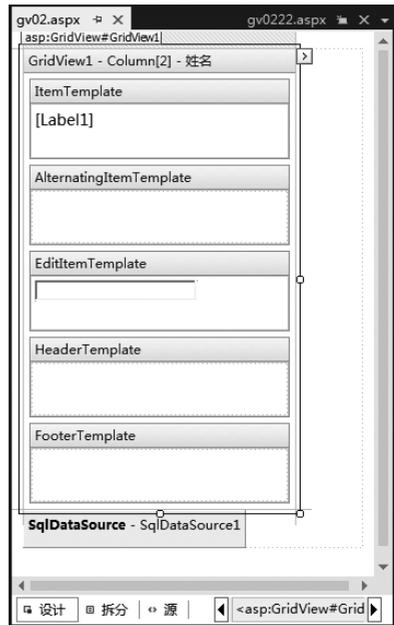


图5-47 “姓名”字段编辑模板窗口

可以看出，EditItemTemplate这里的文本框就是用户在gv02.aspx页面上单击“编辑”按钮后显示的编辑项模板样子，选择EditItemTemplate中的文本框，然后在属性窗口中设置该文本框的显示样子。

(4) 设置图5-47中的EditItemTemplate中的文本框的样式属性：

Width为80px，Height为20px，BorderColor为#CC0000，BorderStyle为Solid，BorderWidth为2px。设置完成后，显示如图5-48所示。

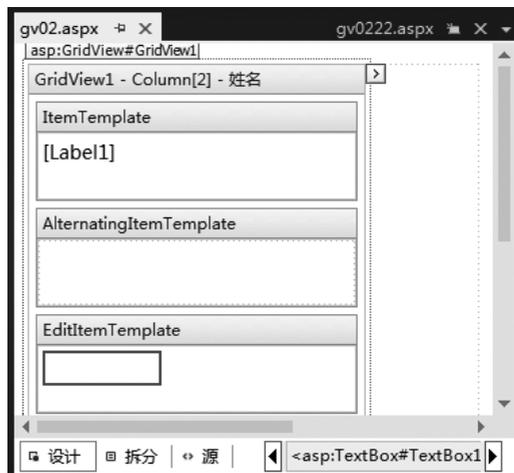


图5-48 修改“姓名”字段EditItemTemplate后的显示效果

(5) 接下来就是给姓名的编辑项模板文本框增加数据验证控件，验证EditItem-Template中的姓名文本框不能为空，用户必须填写。在图5-48中，从工具箱的验证控件中拖动一个必填验证控件（RequiredFieldValidator）到页面上，然后设置该验证控件的属性：

ControlToValidate为Text-Box1、ErrorMessage为*。

(6) 在gv02.aspx页面设计窗口中右击，在弹出的快捷菜单中选择“在浏览器查看”命令，即可看到运行结果，然后单击某行的“编辑”按钮，出现的错误窗口如图5-49所示。

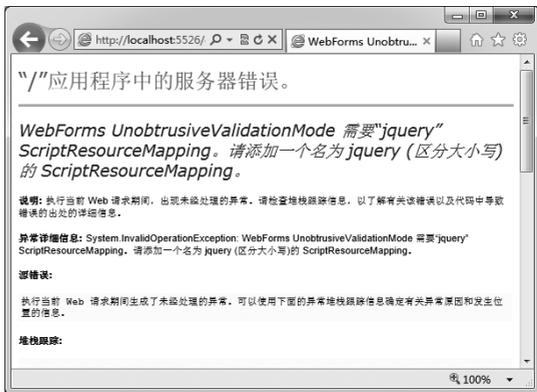


图5-49 页面gv02.aspx单击“编辑”按钮出现的错误窗口

(7) 出现上述错误，是因为Web Form使用UnobtrusiveValidationMode来验证的，所以可以设定不要使用UnobtrusiveValidationMode，在gv02.aspx的程序页面中的Page_Load事件中增加如下代码即可。

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class gv02 : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        UnobtrusiveValidationMode = UnobtrusiveValidationMode.None;
    }
}
```

(8) 再次运行gv02.aspx页面，单击某行的“编辑”按钮，显示如图5-50所示。

(9) 在图5-50中，如果清空姓名“王晓慧”，单击“更新”按钮，就不能提交了，后面就有个“*”提示不能更新，正确输入修改后的姓名，就可以更新了。

维护	编号	姓名
编辑 删除	1	张三
编辑 删除	2	李四
更新 取消	3	王晓慧
编辑 删除	4	张亮22
编辑 删除	5	石贵
编辑 删除	6	贾小西
编辑 删除	7	李波
编辑 删除	8	刘一平
编辑 删除	9	刘越
编辑 删除	10	李军

图5-50 页面gv02.aspx编辑模板窗口

5.9.5 处理性别单选按钮编辑

(1) 处理性别编辑模板，需要替换RadioButtonList控件，当然同样进入“性别”字段的编辑模板EditItemTemplate，如图5-51所示。

(2) 在图5-51中，删除EditItemTemplate中的文本框，然后从工具箱中拖动一个RadioButtonList控件到EditItemTemplate中，如图5-52所示。

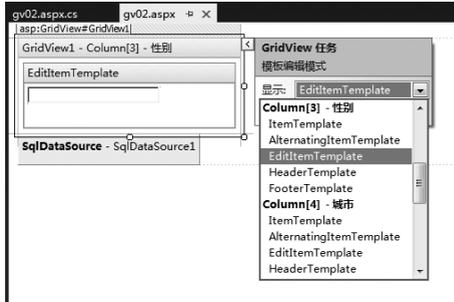


图5-51 “性别”字段编辑模板窗口 图5-52 修改“性别”字段的EditItemTemplate窗口(1)

(3) 在图5-52中，选择“RadioButtonList任务”面板中的“编辑项”项目，弹出如图5-53所示对话框。

(4) 在图5-53中，添加两个成员，其Text/Value依次为：男/男、女/女，然后单击“确定”按钮。gv02.aspx显示窗口如图5-54所示。

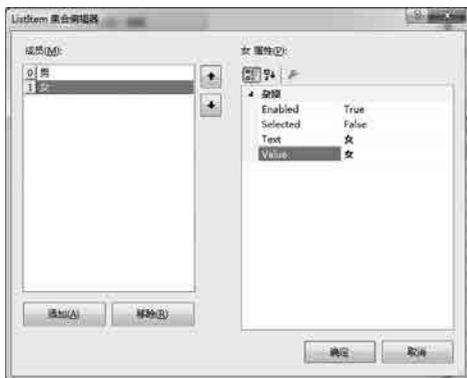


图5-53 “ListItem集合编辑器”对话框 图5-54 修改“性别”字段的EditItemTemplate窗口(2)

(5) 在图5-54中，选择“RadioButtonList任务”面板中的“编辑DataBindings...”项目，弹出如图5-55所示对话框。

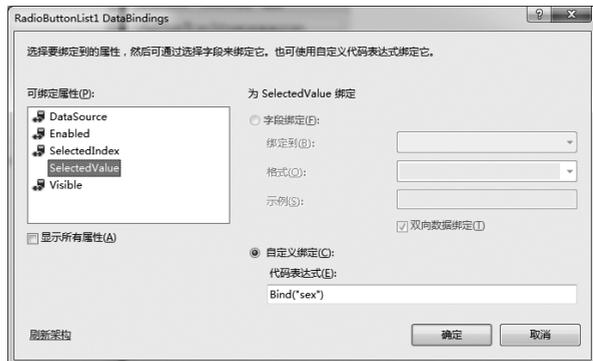


图 5-55 “RadioButtonList1 DataBindings”对话框

(6) 在图5-55中, 左侧的“可绑定属性”选择“SelectedValue”, 右侧选择“自定义绑定”并在“代码表达式”中输入绑定函数表达式“Bind("sex")”, 然后单击“确定”按钮。运行gv02.aspx页面, 然后单击某行的“编辑”按钮, 显示如图5-56所示。

维护	编号	姓名	性别
编辑 删除	1	张三	男
编辑 删除	2	李四	男
更新 取消	3	王晓慧	<input type="radio"/> 男 <input checked="" type="radio"/> 女
编辑 删除	4	张亮22	男
编辑 删除	5	石贵	男
编辑 删除	6	贾小西	女
编辑 删除	7	李波	男
编辑 删除	8	刘一平	男
编辑 删除	9	刘越	男
编辑 删除	10	李军	男

图5-56 页面gv02.aspx编辑模板运行显示效果(处理性别编辑模板后)(1)

(7) 在图5-56中, 很明显就可以通过选择的方式来修改性别了, 不过, 性别只有两个选择, 横向显示好像更好些, 修改RadioButtonList控件中的“RepeatDirection”属性为“Horizontal”(水平方式)即可(默认该属性取值为Vertical 垂直方式), 修改后的页面运行显示效果如图5-57所示。

维护	编号	姓名	性别
编辑 删除	1	张三	男
编辑 删除	2	李四	男
更新 取消	3	王晓慧	<input type="radio"/> 男 <input checked="" type="radio"/> 女
编辑 删除	4	张亮22	男
编辑 删除	5	石贵	男
编辑 删除	6	贾小西	女
编辑 删除	7	李波	男
编辑 删除	8	刘一平	男
编辑 删除	9	刘越	男
编辑 删除	10	李军	男

图5-57 页面gv02.aspx编辑模板运行显示效果(处理性别编辑模板后)(2)

5.9.6 处理城市下拉列表编辑

(1) 城市字段的EditItemTemplate是要把文本框替换成下拉列表DropDownList控件, 同样首先进入“城市”字段的编辑模板窗口, 如图5-58所示。

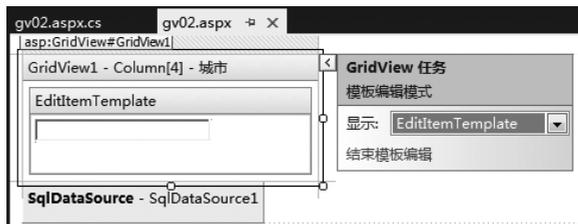


图5-58 页面gv02.aspx“城市”字段编辑模板窗口

(2) 在图5-58中, 删除EditItemTemplate中的文本框, 从工具箱中拖一个DropDownList控件到页面上, 如图5-59所示。



图5-59 页面gv02.aspx“城市”字段编辑模板修改控件窗口

(3) 在mydb数据库中创建一个city_table表，完整脚本代码如下（也可以参见ch05/city_table.sql）。

```
use mydb
go
create table city_table
(
    cityid int identity primary key,
    cityname varchar(50)
)
go
insert into city_table(cityname) select distinct city from test;
go
select * from city_table;
go
```

(4) 在图5-59中，选择“DropDownList任务”面板中的“选择数据源”项目，弹出如图5-60所示对话框。



图5-60 数据源配置向导-选择数据源对话框

(5) 在图5-60中，选择“选择数据源”下拉列表中的“新建数据源”，显示窗口又到了“选择数据源类型”，依然选择“数据库”项目，然后单击“下一步”按钮，显示如图5-61所示。



图5-61 配置数据源-选择您的数据连接对话框

(6) 在图5-61中，依然选择之前保存的mydb ConnectionString，然后单击“下一步”按钮，显示如图5-62所示。



图5-62 配置数据源-配置Select语句对话框

(7) 在图5-62中，单击“下一步”按钮进入“测试查询”对话框，在“测试查询”对话框中直接单击“完成”按钮，显示如图5-63所示。



图5-63 数据源配置向导-选择数据源对话框

(8) 在图5-63中，在DropDownList的显示的数据字段和值字段都选择cityname（这里的练习比较特殊，都选择cityname，通常的操作是：显示的数据字段取cityname，而值字段选择cityid），然后单击“确定”按钮，显示如图5-64所示。



图5-64 页面gv02.aspx编辑城市模板窗口

(9) 在图5-64中，选择“DropDownList任务”面板中的“编辑DataBindings...”项目，弹出如图5-65所示对话框。

(10) 在图5-65中，按照图中的选择，在代码表达式中输入“Bind("city)”，然后单击“确定”按钮，再次运行gv02.aspx页面，然后单击某行的“编辑”按钮，显示效果如图5-66所示，可以看出已经可以通过下拉列表来修改“城市”字段了。



图5-65 “DropDownList1 DataBindings”对话框

维护	编号	姓名	性别	城市
编辑 删除	1	张三	男	北京
编辑 删除	2	李四	男	上海
编辑 删除	3	王晓慧	女	北京
编辑 删除	4	张亮22	男	北京
编辑 删除	5	石贵	男	天津
更新 取消	6	贾小西	<input type="radio"/> 男 <input checked="" type="radio"/> 女	上海
编辑 删除	7	李波	男	北京 广州
编辑 删除	8	刘一平	男	北京 天津
编辑 删除	9	刘越	男	北京
编辑 删除	10	李军	男	北京

图5-66 页面gv02.aspx编辑模板运行效果

5.10 本章小结

本章以鼠标操作的方式详细讲解了ASP.NET核心服务器绑定控件GridView的各种使用方法，不是很难，要熟练、精通的话需要多加练习。

本章要点

- (1) GridView控件功能概述。
- (2) 使用GridView控件列表显示数据。
- (3) 使用GridView控件实现排序、分页功能。
- (4) 修改GridView控件的样式效果。
- (5) 转换模板列操作。
- (6) 使用GridView编辑、删除数据操作。

5.11 思考练习题

熟练操作、练习本章讲解的所有例子，并且能认真思考把这些操作方法用到实际项目开发中。