

Chapter 2 Developing Web Forms

李莉

2017-9

实验 1：创建一个 Web Form
实验 2：实现页面导航

实验 1：创建一个 Web Form

实验目的

熟悉 Web 服务器控件的使用。

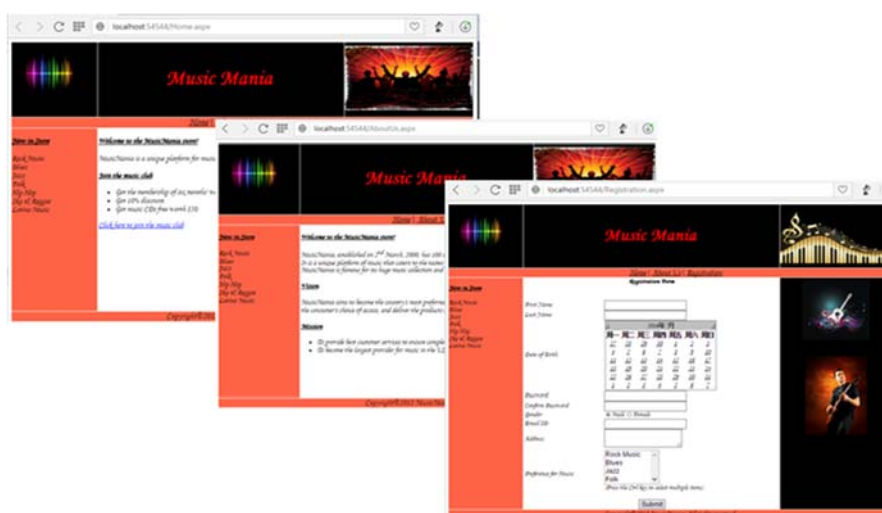
本实验参考《Activity Book》Activity2.1 和 Activity2.2。

本实验需要用到的图片等资源已随同实验指南一同发布。

问题描述

MusicMania 是一家在线音乐商店，它为音乐爱好者提供大量唱片。

该网站需要创建三个页面：Home、Registration、About Us。其效果图如下图所示：



对于 Registration 页面，用户输入出生日期的方式要求如下：首先，点击页面上的日历图片，然后页面弹出日历控件。用户选中日期后，日历控件消失，用户选择的日期出现在文本框。

此外，用户在 Registration 页面输入的信息需要进行验证，要求如下：所有输入域都不能为空。两次口令输入要一致，邮件格式正确。

问题分析

遵循 Smart UI 设计模式，先完成三个 ASP 页面的页面设计，然后添加相应的功能逻辑。

通过观察效果图，三个页面的布局类似，区别在于中间区域的内容。完成第一个窗体后，部分代码可用于其他窗体的设计。

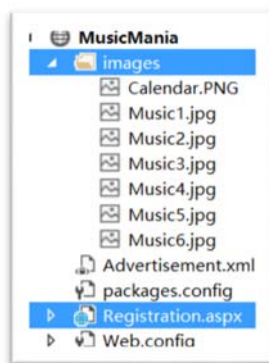
Registration 页面不同于其他两个页面的地方在于它的中心区域是一个 web form。为了实现用户选择日期的效果，需要用到图片按钮、日历控件、文本框，并配合服务器脚本。

对于用户输入的验证，ASP 提供的验证控件可满足其需求，无需编写额外代码。

解决方案

下面是创建 Registration 页面的步骤，其他两个页面可自行发挥。

步骤一：打开 Visual studio，新建一个网站项目。项目模板选择空网站，网站名为 MusicMania。在网站中添加一个新 web form，名为 Registration.aspx。在网站新建一个 images 文件夹，将创建页面所需的图片文件全部添加进去。



步骤二：设计页面布局。根据效果图，页面大体布局如下：

采用表格布局。在窗体中添加一个四行三列的表格，代码如下：

```
<table style="width:100%; font-family: 'Monotype Corsiva'">
  <tr>
    <td style="width:20%"></td>
    <td></td>
    <td style="width:20%"></td>
  </tr>
  <tr>
    <td colspan="3"></td>
  </tr>
  <tr>
    <td style="width:20%"></td>
    <td></td>
    <td style="width:20%"></td>
  </tr>
  <tr>
    <td colspan="3"></td>
  </tr>
</table>
```

步骤三：实现页面的 Logo 部分，对应表格的第一行。前面两个单元格的内容分别是一张图片和一行文本。最右边单元格需要在页面加载时随机显示图片，考虑采用 AdRotator 控件。对于 AdRotator 控件，需要创建一个 xml 文档来存储相关广告信息。因此，在项目中添加一个名为 Advertisement 的 XML 文档，文档内容如下：

```
<?xml version="1.0" encoding="utf-8" ?>
<Advertisements>
  <Ad>
    <ImageUrl>~/images/Music4.jpg</ImageUrl>
    <Height>150px</Height>
    <Width>300px</Width>
  </Ad>
  <Ad>
    <ImageUrl>~/images/Music5.jpg</ImageUrl>
```

```

        <Height>150px</Height>
        <Width>300px</Width>
    </Ad>
    <Ad>
        <ImageUrl>~/images/Music6.jpg</ImageUrl>
        <Height>150px</Height>
        <Width>300px</Width>
    </Ad>
</Advertisements>

```

回到 web form, 填写表格第一行三个单元格的内容。

```

<tr style="background-color:black">
    <td style="width:20% align="center">
        <asp:Image ID="Image1" runat="server" ImageUrl="~/images/Music1.jpg"/>
    </td>
    <td style="font-size: 50px; font-weight: bold; color: #FF0000; text-align:center">
        Music Mania
    </td>
    <td style="width:20%">
        <asp:AdRotator ID="AdRotator1" runat="server"
        AdvertisementFile="~/Advertisement.xml" AlternateText="MusicMania"/>
    </td>
</tr>

```

运行页面, 查看效果。通过刷新页面, 可看到第三单元格会随机显示三张图片中的一张。



步骤四：实现页面导航栏, 对应表格第二行。在单元格内添加 3 个超链接控件。设置每个超链接的 URL 和文本属性。

```

<tr>
    <td colspan="3" style="font-size: 22px; background-color:tomato; text-align:center">
        <asp:HyperLink ID="HyperLink1" runat="server" NavigateUrl="Home.aspx"
        ForeColor="Black">Home</asp:HyperLink>|
        <asp:HyperLink ID="HyperLink2" runat="server" NavigateUrl="AboutUs.aspx"
        ForeColor="Black">About Us</asp:HyperLink>|
        <asp:HyperLink ID="HyperLink3" runat="server" NavigateUrl="Registration.aspx"
        ForeColor="Black">Registration</asp:HyperLink>
    </td>
</tr>

```

运行页面, 查看效果：



步骤五：实现表格第三行的两个边条。第三行左边的单元格内容是一些文本, 右边的单元格是上下排列的两张图片。

```

<tr>
    <td style="width:20%; background-color:tomato; vertical-align:top">
        <u><b><br />New in Store</b></u>
        <br /><br />
        Rock Music<br />
        Blues<br />
        Jazz<br />
        Folk<br />
    </td>
    <td style="width:80%; background-color:tomato; vertical-align:top">
        <asp:Image ID="Image2" runat="server" ImageUrl="~/images/Music2.jpg"/>
        <br />
        <asp:Image ID="Image3" runat="server" ImageUrl="~/images/Music3.jpg"/>
    </td>
</tr>

```

```

        Hip Hop<br />
        Ska & Reggae<br />
        Latino Music
    </td>
</td></td>
<td style="width:20%; text-align: center; background-color: Black">
    <asp:Image ID="Image2" runat="server" ImageUrl="~/images/Music2.jpg" />
    <br /><br /><br />
    <asp:Image ID="Image3" runat="server" ImageUrl="~/images/Music3.jpg" />
</td>
</tr>

```

运行页面，查看效果：



步骤五：实现页面底部版权信息，对应表格第四行。该行只有一个单元格，内容是一些文本。

```

<tr style="background-color: tomato">
    <td colspan="3" style="text-align: center">
        <i>Copyright&copy;2013 MusicMania. All rights reserved.</i>
    </td>
</tr>

```

运行页面，查看效果。



步骤六：实现页面中心的注册区域，对应表格第三行的中间单元格。该区域也采用表格布局，大体设计如下：

Registration Form

First Name:

Last Name:

Date of Birth:

2016年7月

周一

周二

周三

周四

周五

周六

周日

27

28

29

30

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

1

2

3

4

5

6

7

Password:

Confirm Password:

Gender:

☒ Male ☐ Female

Email ID:

Address:

Preference for Music:

Rock Music

Blues

Jazz

Folk

(Press the Ctrl key to select multiple items)

Submit

根据上面的设计，在单元格内嵌入一个 11 行两列的表格。

```
<td>
  <table>
    <tr><td colspan="2"></td></tr>
    <tr><td></td><td></td></tr>
    <tr><td></td><td></td></tr>
    <tr><td></td><td></td></tr>
    <tr><td></td><td></td></tr>
    <tr><td></td><td></td></tr>
    <tr><td></td><td></td></tr>
    <tr><td></td><td></td></tr>
    <tr><td></td><td></td></tr>
    <tr><td></td><td></td></tr>
    <tr><td colspan="2"></td></tr>
  </table>
</td>
```

每个单元格填入的服务器控件如下表所示：

行数	单元格 1	单元格 2	备注
第 1 行	Label		Text 属性设为 Registration Form
第 2 行	Label	TextBox	输入 First Name
第 3 行	Label	TextBox	输入 Last Name
第 4 行	Label	Calender	
第 5 行	Label	TextBox	TextMode 属性设为 Password
第 6 行	Label	TextBox	TextMode 属性设为 Password
第 7 行	Label	RadioButton	两个 RadioButton，GroupName 属性设为 Gender
第 8 行	Label	TextBox	
第 9 行	Label	TextBox	TextMode 属性设为 MultiLine
第 10 行	Label	ListBox	通过 ListItem 添加列表项
第 11 行	Button		用于提交表单

上述表格对应的代码如下：

```
<table>
  <tr>
    <td colspan="2" style="text-align: center">
      <asp:Label ID="Label1" runat="server" Font-Bold="True" Text="Registration Form"></asp:Label>
    <br /><br /><br />
  </td>
</tr>
```

```

        </td>
    </tr>
    <tr>
        <td>
            <asp:Label ID="Label2" runat="server" Text="First Name:"></asp:Label>
        </td>
        <td>
            <asp:TextBox ID="fname" runat="server"></asp:TextBox>
        </td>
    </tr>
    <tr>
        <td>
            <asp:Label ID="Label3" runat="server" Text="Last Name:"></asp:Label>
        </td>
        <td>
            <asp:TextBox ID="lname" runat="server"></asp:TextBox>
        </td>
    </tr>
    <tr>
        <td>
            <asp:Label ID="Label4" runat="server" Text="Date of Birth:"></asp:Label>
        </td>
        <td>
            <asp:Calendar ID="Dob" runat="server" ></asp:Calendar>
        </td>
    </tr>
    <tr>
        <td>
            <asp:Label ID="Label5" runat="server" Text="Password:"></asp:Label>
        </td>
        <td>
            <asp:TextBox ID="Pwd" runat="server" TextMode="Password"></asp:TextBox>
        </td>
    </tr>
    <tr>
        <td>
            <asp:Label ID="Label6" runat="server" Text="Confirm Password:"></asp:Label>
        </td>
        <td>
            <asp:TextBox ID="cpwd" runat="server" TextMode="Password"></asp:TextBox>
        </td>
    </tr>
    <tr>
        <td>
            <asp:Label ID="Label7" runat="server" Text="Gender:"></asp:Label>
        </td>
        <td>
            <asp:RadioButton ID="RadioButton1" runat="server" Checked="True" GroupName="Gender" Text="Male"
            />
            <asp:RadioButton ID="RadioButton2" runat="server" GroupName="Gender" Text="Female" />
        </td>
    </tr>
    <tr>
        <td>
            <asp:Label ID="Label8" runat="server" Text="Email ID:"></asp:Label>
        </td>
        <td>
            <asp:TextBox ID="emailid" runat="server"></asp:TextBox>
        </td>
    </tr>
    <tr>
        <td>
            <asp:Label ID="Label9" runat="server" Text="Address:"></asp:Label>
        </td>
        <td>
            <asp:TextBox ID="address" runat="server" TextMode="MultiLine"></asp:TextBox>
        </td>
    </tr>
    </tr>

```

```

<td>
    <asp:Label ID="Label10" runat="server" Text="Preference for Music:"></asp:Label>
</td>
<td>
    <asp:ListBox ID="preference" runat="server">
        <asp:ListItem>Rock Music</asp:ListItem>
        <asp:ListItem>Blues</asp:ListItem>
        <asp:ListItem>Jazz</asp:ListItem>
        <asp:ListItem>Folk</asp:ListItem>
        <asp:ListItem>HipHop</asp:ListItem>
        <asp:ListItem>Ska & Reggae</asp:ListItem>
        <asp:ListItem>Latino Music</asp:ListItem>
    </asp:ListBox>
    <br />
</td>
</tr>
<tr>
<td colspan="2" style="text-align: center">
    <asp:Button ID="Button1" runat="server" Text="Submit" />
</td>
</tr>
</table>

```

运行页面，查看效果

步骤七：通过查看页面效果，发现日期选择的效果跟实验要求不完全相符。日期控件不是一直显示在页面上，而是用户点击一个图片后才显示出来，并且用户选完日期后会消失。其效果如下图所示：

为实现这一效果，可以利用服务器控件的 Visible 属性来实现控件在页面上的显示与否。为达到上述效果，需要在 Calendar 控件所在单元格另外添加一个 TextBox 控件和一个

ImageButton 控件。其作用如下：

行数	单元格 1	单元格 2	备注
第 4 行	Label	Calender	页面首次加载时不可见, 当用户点击 ImageButton 时可见, 用户选择日期后再次不可见。
第 4 行		TextBox	页面首次加载时可见, 用户选择日期后, 日期信息显示在文本框。
第 4 行		ImageButton	页面首次加载时可见。用户点击按钮时, 让 Calender 可见, 自己不可见。

该单元格内容如下：

```
<td>
    <asp:TextBox ID="dobval" runat="server" ReadOnly="True"></asp:TextBox>
    <asp:Calendar ID="Dob" runat="server" Visible="False"></asp:Calendar>
    <asp:ImageButton ID="ImageButton1" runat="server" CausesValidation="False"
    ImageUrl="~/images/Calendar.PNG"/> <br />
</td>
```

为了实现上述控件跟用户的交互, 需要为 ImageButton 和 Calender 控件各添加一个事件处理：

控件	事件	事件处理
ImageButton	Click	让 Calender 控件可见。
Calender	SelectionChanged	让 Calender 控件不可见, 读取 Calender 控件的日期, 设为 TextBox 文本。

具体代码如下：

```
<td>
    <asp:TextBox ID="dobval" runat="server" ReadOnly="True"></asp:TextBox>
    <asp:Calendar ID="Dob" runat="server" OnSelectionChanged="Dob_SelectionChanged"
    Visible="False"></asp:Calendar>
    <asp:ImageButton ID="ImageButton1" runat="server" CausesValidation="False"
    ImageUrl="~/images/Calendar.PNG" OnClick="ImageButton1_Click" /> <br />
</td>
```

```
protected void ImageButton1_Click(object sender, ImageClickEventArgs e)
{
    Dob.Visible = true;
}
protected void Dob_SelectionChanged(object sender, EventArgs e)
{
    dobval.Text = Dob.SelectedDate.ToShortDateString();
    Dob.Visible = false;
}
```

运行页面, 查看交互效果。

步骤八 :实验还要求对用户输入进行验证。因此, 需要为输入控件添加合适的验证控件, 并设置验证控件的提示信息等属性。

行数	单元格 1	单元格 2	备注
第 1 行	Label		Text 属性设为 Registration Form
第 2 行	Label	TextBox RequiredFieldValidator	输入 First Name
第 3 行	Label	TextBox RequiredFieldValidator	输入 Last Name
第 4 行	Label	TextBox Calender ImageButton	

		RequiredFieldValidator	
第 5 行	Label	TextBox RequiredFieldValidator	TextMode 属性设为 Password
第 6 行	Label	TextBox CompareValidator	TextMode 属性设为 Password
第 7 行	Label	RadioButton	两个 RadioButton, GroupName 属性设为 Gender
第 8 行	Label	TextBox RequiredFieldValidator RegularExpressionValidator	
第 9 行	Label	TextBox RequiredFieldValidator	TextMode 属性设为 MultiLine
第 10 行	Label	ListBox	通过 ListItem 添加列表项
第 11 行	Button		用于提交表单

具体代码如下：

```

<table>
  <tr>
    <td colspan="2" style="text-align: center">
      <asp:Label ID="Label1" runat="server" Font-Bold="True" Text="Registration
Form"></asp:Label>
      <br /><br /><br />
    </td>
  </tr>
  <tr>
    <td>
      <asp:Label ID="Label2" runat="server" Text="First Name:"></asp:Label>
    </td>
    <td>
      <asp:TextBox ID="fname" runat="server"></asp:TextBox>
      <asp:RequiredFieldValidator ID="RequiredFieldValidator1" runat="server"
ControlToValidate="fname" ErrorMessage="Please enter your first
name."></asp:RequiredFieldValidator>
    </td>
  </tr>
  <tr>
    <td>
      <asp:Label ID="Label3" runat="server" Text="Last Name:"></asp:Label>
    </td>
    <td>
      <asp:TextBox ID="lname" runat="server"></asp:TextBox>
      <asp:RequiredFieldValidator ID="RequiredFieldValidator2" runat="server"
ControlToValidate="lname" ErrorMessage="Please enter your last
name."></asp:RequiredFieldValidator>
    </td>
  </tr>
  <tr>
    <td>
      <asp:Label ID="Label4" runat="server" Text="Date of Birth:"></asp:Label>
    </td>
    <td>
      <asp:TextBox ID="dobval" runat="server" ReadOnly="True"></asp:TextBox>
      <asp:Calendar ID="Dob" runat="server" OnSelectionChanged="Dob_SelectionChanged"
Visible="False"></asp:Calendar>
      <asp:ImageButton ID="ImageButton1" runat="server" CausesValidation="False"
ImageUrl="~/images/Calendar.PNG" OnClick="ImageButton1_Click" /> <br />
      <asp:RequiredFieldValidator ID="RequiredFieldValidator3" runat="server"
ControlToValidate="dobval" ErrorMessage="Please select your date of
birth."></asp:RequiredFieldValidator>
    </td>
  </tr>

```

```

</tr>
<tr>
<td>
<asp:Label ID="Label5" runat="server" Text="Password:"></asp:Label>
</td>
<td>
<asp:TextBox ID="Pwd" runat="server" TextMode="Password"></asp:TextBox>
<asp:RequiredFieldValidator ID="RequiredFieldValidator4" runat="server"
ControlToValidate="Pwd" ErrorMessage="Please enter your
password."></asp:RequiredFieldValidator>
</td>
</tr>
<tr>
<td>
<asp:Label ID="Label6" runat="server" Text="Confirm Password:"></asp:Label>
</td>
<td>
<asp:TextBox ID="cpwd" runat="server" TextMode="Password"></asp:TextBox>
<asp:RequiredFieldValidator ID="RequiredFieldValidator5" runat="server"
ControlToValidate="cpwd" ErrorMessage="Please re-enter your
password."></asp:RequiredFieldValidator><br />
<asp:CompareValidator ID="CompareValidator1" runat="server"
ControlToCompare="Pwd" ControlToValidate="cpwd" ErrorMessage="Password mismatch. Please
re-enter password."></asp:CompareValidator>

</td>
</tr>
<tr>
<td>
<asp:Label ID="Label7" runat="server" Text="Gender:"></asp:Label>
</td>
<td>
<asp:RadioButton ID="RadioButton1" runat="server" Checked="True"
GroupName="Gender" Text="Male" />
<asp:RadioButton ID="RadioButton2" runat="server" GroupName="Gender"
Text="Female" />
</td>
</tr>
<tr>
<td>
<asp:Label ID="Label8" runat="server" Text="Email ID:"></asp:Label>
</td>
<td>
<asp:TextBox ID="emailid" runat="server"></asp:TextBox>
<asp:RequiredFieldValidator ID="RequiredFieldValidator6" runat="server"
ControlToValidate="emailid" ErrorMessage="Please enter your email
ID."></asp:RequiredFieldValidator><br />
<asp:RegularExpressionValidator ID="RegularExpressionValidator1"
runat="server" ControlToValidate="emailid" ErrorMessage="Please provide email ID in an
appropriate format." ValidationExpression="\w+([-+.' ]\w+)*@\w+([-+.' ]\w+)*\. \w+([-+.' ]\w+)*"></asp:RegularExpressionValidator>

</td>
</tr>
<tr>
<td>
<asp:Label ID="Label9" runat="server" Text="Address:"></asp:Label>
</td>
<td>
<asp:TextBox ID="address" runat="server" TextMode="MultiLine"></asp:TextBox>
<asp:RequiredFieldValidator ID="RequiredFieldValidator7" runat="server"
ControlToValidate="address" ErrorMessage="Please enter your
address."></asp:RequiredFieldValidator>

</td>
</tr>
<tr>
<td>
<asp:Label ID="Label10" runat="server" Text="Preference for

```

```
Music:"></asp:Label>
</td>
<td>
    <asp:ListBox ID="preference" runat="server">
        <asp:ListItem>Rock Music</asp:ListItem>
        <asp:ListItem>Blues</asp:ListItem>
        <asp:ListItem>Jazz</asp:ListItem>
        <asp:ListItem>Folk</asp:ListItem>
        <asp:ListItem>HipHop</asp:ListItem>
        <asp:ListItem>Ska & Reggae</asp:ListItem>
        <asp:ListItem>Latino Music</asp:ListItem>
    </asp:ListBox>
    <br />
</td>
</tr>
<tr>
<td colspan="2" style="text-align: center">
    <asp:Button ID="Button1" runat="server" Text="Submit" />
</td>
</tr>
</table>
```

运行页面查看效果。如果遇到以下错误提示，请打开网站配置文件 Web.config，加入以下代码。因为项目中并未添加客户端脚本库，所以关闭客户端验证模式。

"/"应用程序中的服务器错误。

WebForms UnobtrusiveValidationMode 需要 jquery"ScriptResourceMapping。请添加一个名为 jquery (区分大小写)的 ScriptResourceMapping。

```
<appSettings>
  <add key="ValidationSettings:UnobtrusiveValidationMode" value="None"/>
</appSettings>
```

运行页面，测试日期选择，还有输入域的验证是否有效，尝试提交表单。

实验总结

通过本实验熟悉 ASP 页面的开发流程、web 服务器控件的使用。

思考题

1. Registration 页面采用内联样式，存在不少重复的样式代码，请改善。
2. 本实验采用 html 的 <table> 元素进行页面布局，请尝试使用 ASP.NET 的 table 服务器控件进行布局，它们有什么区别？
3. registration.aspx 页面，用户姓名、口令、邮箱、地址的输入均使用了 textbox 控件，如果使用 html 元素，需要使用哪些控件来实现上述信息的输入？
4. registration.aspx 页面，如果希望所有错误提示信息出现在一个消息框里面，而不是显示在控件旁边，该如何实现？
5. registration 页面，用户在输入信息时，发生了几次回传 (postback)？何时发生？
6. 如果用户可以选择多个喜欢的音乐类别，该如何实现？

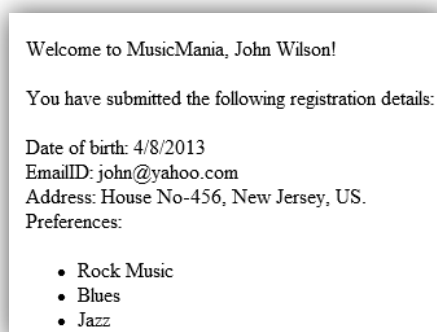
实验 2：实现页面导航

实验目的

理解 Postback 和页面生命周期。

问题描述

在实验 1 基础上，当用户提交注册页面后，网站在另一个页面 RegistrationDetail 显示用户注册信息。效果如下图所示：



问题分析

在 Registration 页面，当用户点击提交按钮时，页面会发生回传，提交给服务器端处理。默认情况，页面是回传给自身。例如 Registration 页面的回传会触发服务器端 Registration 页面类的执行。

实验要求 Registration 页面回传后返回的是 RegistrationDetail 页面，并且要在 RegistrationDetail 页面显示 Registration 页面的信息。从一个页面转向另一个页面，有两种结局方案：

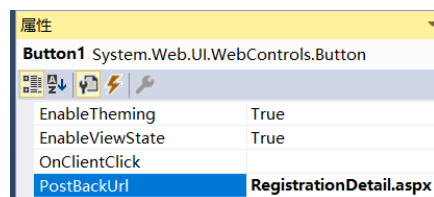
方案一：Registration 页面触发回传后，在提交按钮的 Click 事件处理中调用 Server.Transfer()方法转向 RegistrationDetail 页面。同时，将用户填写的注册信息以 Querystring 的形式附加在 URL 中。这种方法的缺点是 URL 长度有限，而且构建 Querystring 需要编写额外代码。

方案二：采用跨页回传。直接将 Registration 页面提交按钮的 PostbackUrl 设为 RegistrationDetail 页面。当回传到 RegistrationDetail 页面，利用 PreviousPage 属性，可以访问 Registration 页面的控件属性，很方便地获取用户注册信息。

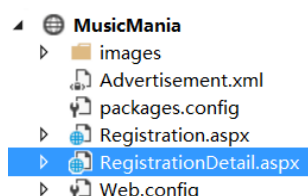
解决方案

下面的步骤实现跨页回传。

步骤一：打开 Registration 页面，设置提交按钮的 PostbackUrl 属性为 RegistrationDetail.aspx。



步骤二：在网站中添加一个新 Web 窗体 RegistrationDetail。



步骤三：打开 RegistrationDetail 页面的代码文件，编辑以下代码。

```
protected void Page_Load(object sender, EventArgs e)
{
    if(PreviousPage!=null)
    {
        TextBox fname = (TextBox)PreviousPage.FindControl("fname");
        TextBox lname = (TextBox)PreviousPage.FindControl("lname");
        Response.Write("<p>Welcome to MusicMania, "+fname.Text+" "+lname.Text+"!</p>");
        Response.Write("<br/><p>You have submitted the following registration
details:</p><br/>");
        TextBox date = (TextBox)PreviousPage.FindControl("dobval");
        Response.Write("<p>Date of birth: "+date.Text+"</p>");
        TextBox email = (TextBox)PreviousPage.FindControl("emailid");
        Response.Write("<p>EmailID: "+email.Text+"</p>");
        TextBox address = (TextBox)PreviousPage.FindControl("address");
        Response.Write("<p>Address: " + address.Text + "</p>");
        ListBox preference = (ListBox)PreviousPage.FindControl("preference");
        List<ListItem> items = new List<ListItem>();
        foreach(ListItem lt in preference.Items)
        {
            if (lt.Selected)
                items.Add(lt);
        }
        Response.Write("<p>Preferences:</p><ul>");
        foreach (ListItem i in items)
            Response.Write("<li>"+i.Value+"</li>");
        Response.Write("</ul>");
    }
}
```

运行 Registration 页面，提交表单，查看 RegistrationDetail 页面内容。

实验总结

通过该实验理解跨页回传，在目标页面如何访问源页面信息。

思考题

1. 如果想要读取 Registration 页面的性别信息并在 RegistrationDetail 页面显示，该如何实现？