**实验7 调用Web Service实现天气预报**

实验相关材料位于：<ftp://192.168.40.100/>下的“**--- 梁其洋**”目录下（可以通过搜索名字找到）

作业提交到：[ftp://192.168.40.14/梁其洋/网络编程\*\*班/实验\*](ftp://192.168.40.14/梁其洋/网络编程**班/实验*)

要求：1. 提交时**将本文档重新命名**，文档命名规则：学号+姓名+实验\*，例如，**517300614400 张三 实验\*.docx （一定要学号在前姓名在后，方便自动排序！）**

2. 你们没有FTP文件的删除权限，**如果需要提交新版本，在姓名后加序号后提交即可，例如，517300614400张三（1）实验\*.docx**

实验7.1 WPF调用WebService实现天气预报

实验步骤：

（1）新建一个WPF应用程序，在MainWindow.xaml的<Grid>标签中添加

<Label Content="输入城市名称：" Height="29" HorizontalAlignment="Left" Margin="12,12,0,0" Name="label1" VerticalAlignment="Top" Width="108" />

<Label Content="显示天气" Height="29" HorizontalAlignment="Left" Margin="94,91,0,0" Name="label3" VerticalAlignment="Top" Width="209" />

<Label Content="天气概况：" Height="29" HorizontalAlignment="Left" Margin="12,91,0,0" Name="label4" VerticalAlignment="Top" Width="76" />

<Label Content="天气实况：" Height="29" HorizontalAlignment="Left" Margin="12,164,0,0" Name="label5" VerticalAlignment="Top" Width="76" />

<TextBox Height="29" HorizontalAlignment="Left" Margin="139,12,0,0" Name="textBox1" VerticalAlignment="Top" Width="81" />

<Button Content="查询" Height="28" HorizontalAlignment="Left" Margin="228,13,0,0" Name="button1" VerticalAlignment="Top" Width="48" Click="button1\_Click" />

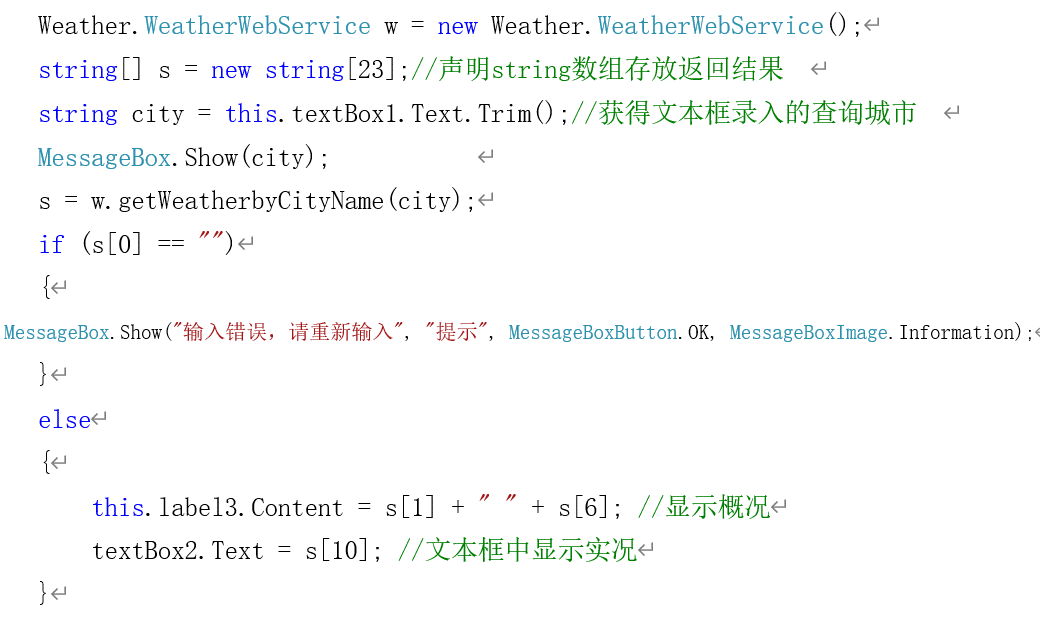
<Image Height="29" HorizontalAlignment="Right" Margin="0,91,364,0" Name="image1" Stretch="Fill" VerticalAlignment="Top" Width="33" />

<TextBox Height="71" HorizontalAlignment="Left" Margin="94,0,0,130" Name="textBox2" VerticalAlignment="Bottom" Width="363" TextWrapping="Wrap" />

（2）在MainWindow.xaml.cs中实现button1的Click事件：

private void button1\_Click(object sender, RoutedEventArgs e)

{

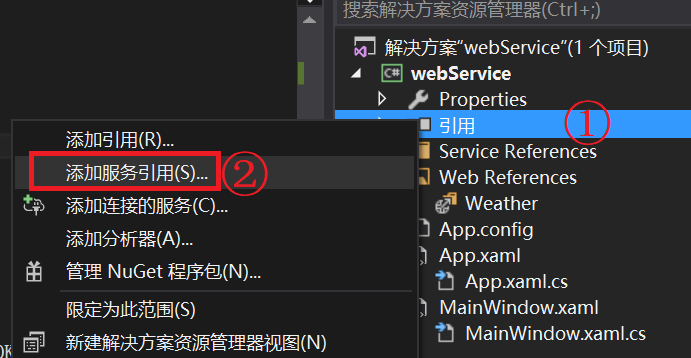


}

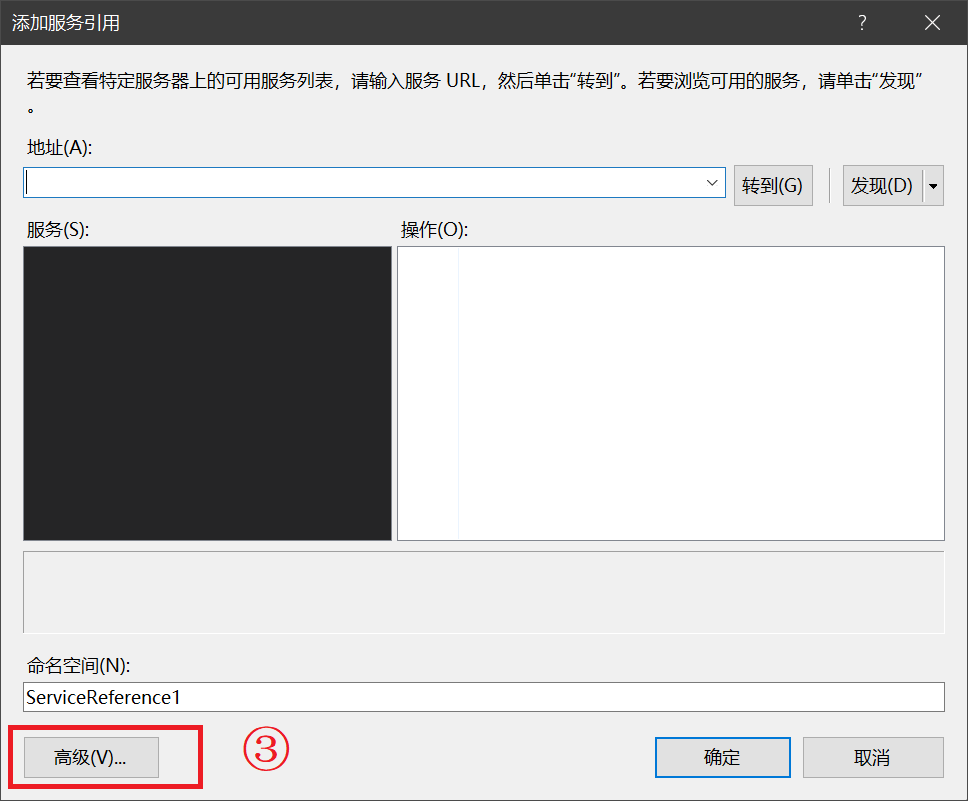
说明：String(0)—省份，**String(1)—城市**，String(2)—城市代码，String(3)—城市图片名称，String(4)—最后更新时间。String(5)—当天的气温，**String(6)—概况**，String(7)—风向和风力，String(8)—天气趋势开始图片名称(以下称：图标一)，String(9)—天气趋势结束图片名称(以下称：图标二)，**String(10)—现在的天气实况**，String(11)—天气和生活指数。String(12)—第二天的气温，String(13)—概况，String(14)—风向和风力，String(15)—图标一，String(16)—图标二。String(17) —第三天的 气温，String(18)—概况，String(19)—风向和风力，String(20)—图标一，String(21)—图标二。String(22)—查询的城市或地区的介绍

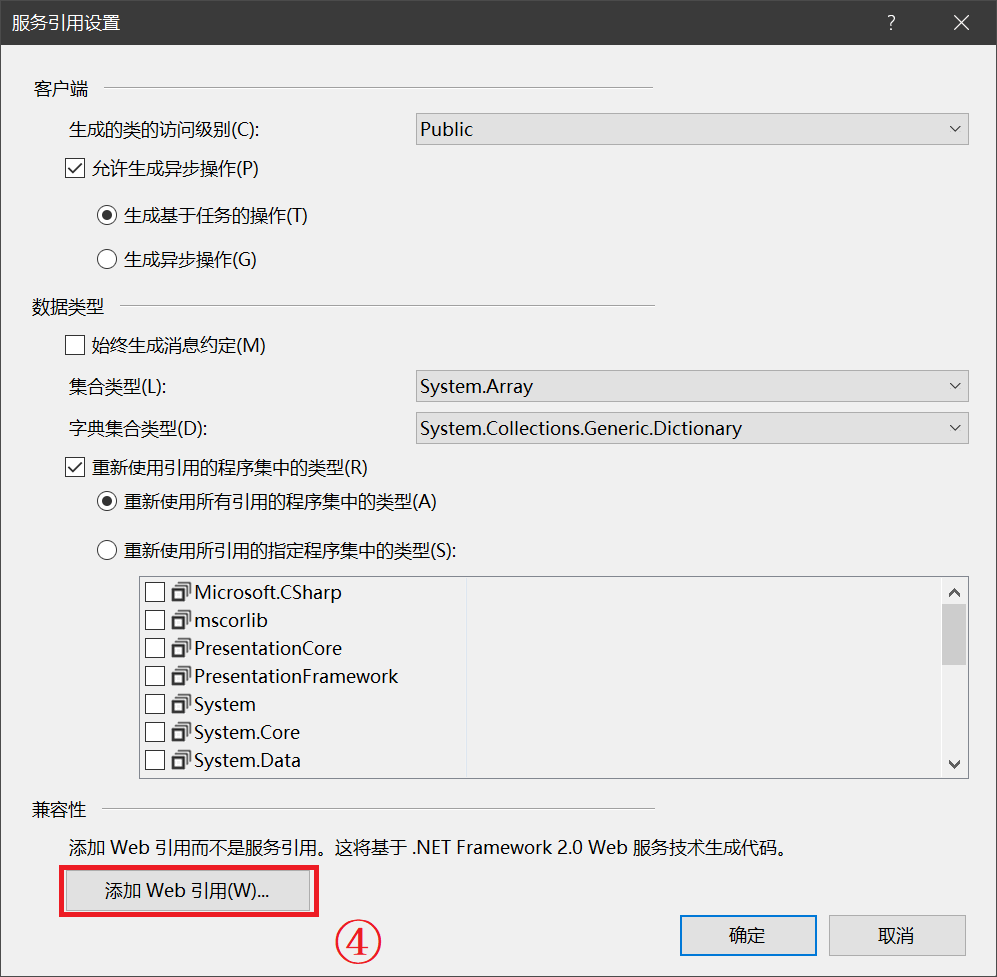
（3）添加Web服务：

在工程名下，引用—添加服务引用，



程序运行截图：





在标记⑤处添加以下web服务：

[**http://www.webxml.com.cn/WebServices/WeatherWebService.asmx**](http://www.webxml.com.cn/WebServices/WeatherWebService.asmx)



程序运行截图：

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

using Weather;

namespace sy7\_2

{

/// <summary>

/// Interaction logic for MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

}

private void button1\_Click(object sender, RoutedEventArgs e)

{

Weather.WeatherWebServiceSoapClient w = new Weather.WeatherWebServiceSoapClient("WeatherWebServiceSoap");

string[] s = new string[23];//声明string数组存放返回结果

string city = this.textBox1.Text.Trim();//获得文本框录入的查询城市

s = w.getWeatherbyCityName(city);

if (s[8] == "")

{

MessageBox.Show("输入错误，请重新输入", "提示", MessageBoxButton.OK, MessageBoxImage.Information);

}

else

{

this.label3.Content = s[1] + " " + s[6];

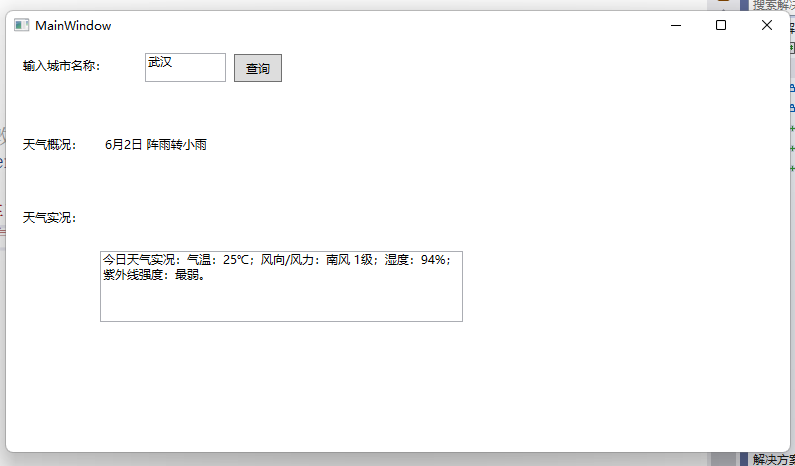
textBox2.Text = s[10];

}

}

}

}



思考：从Web Servevice获取的信息没有很好的分类展示，例如天气图片就没有显示到窗体上。