

现代操作系统应用开发实验报告

学号：15331242

班级：周四上午班

姓名：明友芬

实验名称：HomeWork7

一. 参考资料

<http://blog.csdn.net/coolzy/article/details/8606803>

<http://blog.csdn.net/coolzy/article/details/8606803>

<https://msdn.microsoft.com/zh-cn/library/windows/apps/windows.data.xml.dom.xmldocument.aspx>

二. 实验步骤

- (1) 上网查找并申请手机归属地查询和天气预报的 API ;
- (2) 使用 HttpClient 访问网络 ;
- (3) 在 Mainpage.xaml 里面完成页面的设计 ;
- (4) 在 Mainpage.xaml.cs 文件里面完成查找手机号码归属地和输入城市查询天气预报的代码的编写 ;
- (5) 调试程序。

三. 实验结果截图

- (1) 根据自己申请的 API 查看手机号码归属地的内容，当输入的电话号码正确时得到

如下图 1 所示的 json 格式的返回信息：



图 1

当输入的电话号码不正确时，得到如下图 2 所示的错误的 json 格式的返回信息：



图 2

(2) 根据自己申请的天气预报 API 查询天气预报的信息，当输入的地址正确时得到如图 3 所示的 xml 格式的信息：

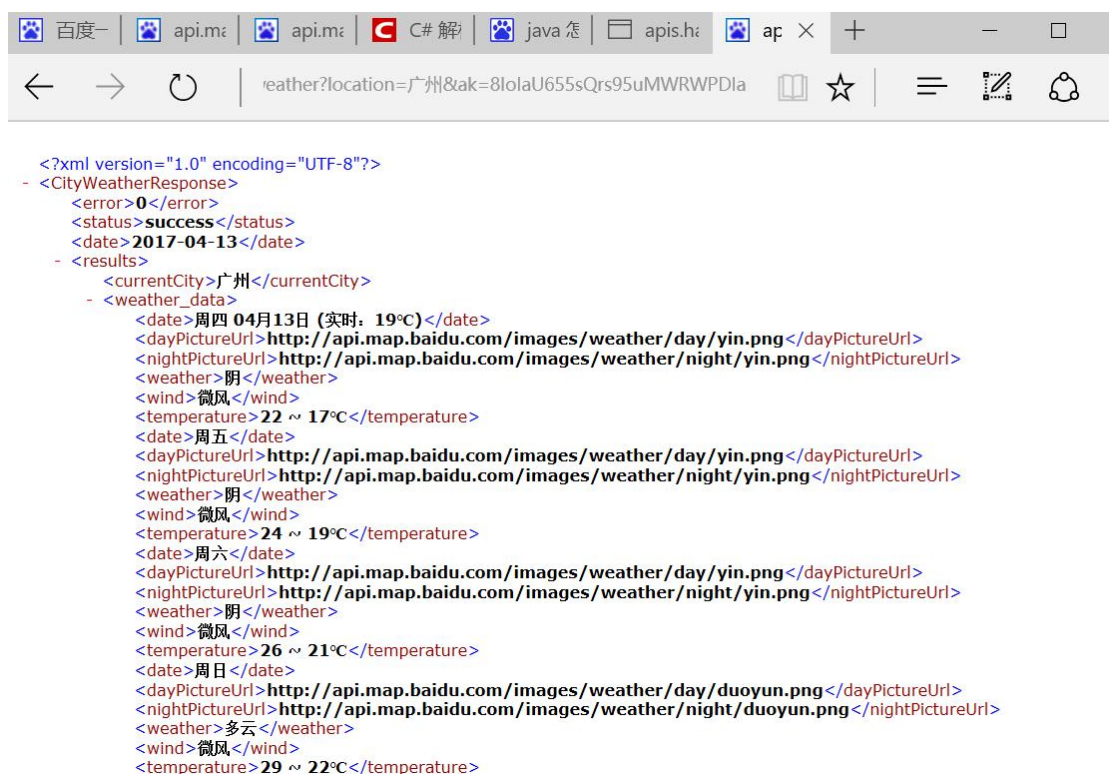


图 3

当输入的地址信息错误时，得到如下图 4 所示的返回值为 xml 格式的信息：

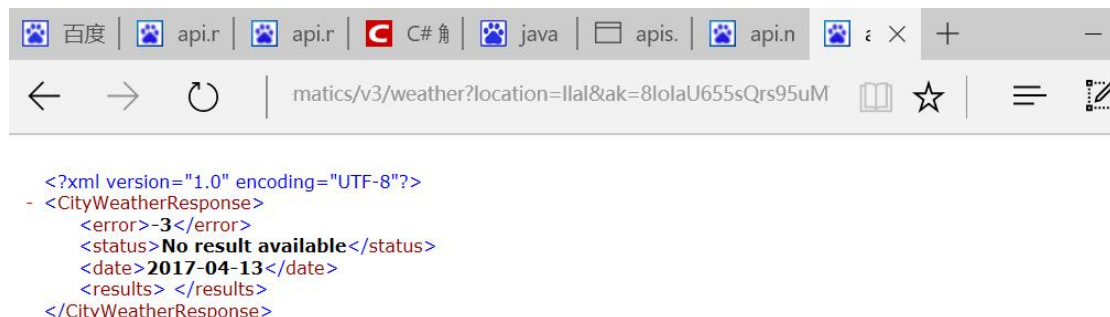


图 4

(3) 查询初始界面如图 5 所示：



图 5

(4) 输入正确格式的电话号码，查询归属地信息之后的返回结果如图 6 所示：



图 6

(5) 若输入的是错误的电话号码，那么会弹出一个弹框，显示查询失败，如图 7 所示：



图 7

(6) 在解析 json 格式的电话号码的信息的过程如图 8 所示:

```
async void PhoneNumberQuery(string phonenumber)
{
    string uri = "http://apis.haoservice.com/mobile?phone=" + phonenumber + "&key=05f9664210394675a523b7739891e504";
    HttpClient client = new HttpClient();
    string result = await client.GetStringAsync(uri);
    JsonTextReader json_ = new JsonTextReader(new StringReader(result));
    string jsonVal = "", error_code = "";
    string province = "";
    string city = "";
    string runner_ = "";
    while (json_.Read())
    {
        jsonVal += json_.Value;
        if (jsonVal.Equals("error_code")) // 读到"cityCode"时, 取出下一个json token (即城市对应代码)
        {
            json_.Read();
            error_code += json_.Value; // 该对象重载了"+=", 故可与字符串进行连接
        } if (jsonVal.Equals("province")) {
            json_.Read();
            province += json_.Value;
        } if (jsonVal.Equals("city")) {
            json_.Read();
            city += json_.Value;
        } if (jsonVal.Equals("company")) {
            json_.Read();
            runner_ += json_.Value;
        }
        jsonVal = "";
    }
    if (province == "") {
        var ii = new MessageBox("No such telephone number!").ShowAsync();
    } else {
        location.Text = province + city;
        runner.Text = runner_;
    }
}
```

图 8

使用 httpClient () 获取信息, 然后使用 JsonTextReader 读取字符串信息, 在使用一个循环获取自己想要的信息就好了, 在这里异常处理机制就是根据图 1 和图 2 中显示的 province 的差别来进行处理的, 在正确返回信息时 province 是有值的但是, 输入的电话号码信息错误的时候没有 province 信息。

(7) 输入正确的地址查询天气的截图如图 9 所示:

天气查询	
城市	<input type="text" value="广州"/>
日期	2017-04-13
天气	阴
温度	22 ~ 17°C
风	微风
<input type="button" value="提交"/>	

图 9

(8) 输入错误的地址的时候, 会出现一个弹框提示输入的信息是错误的, 如图 10 所示:



图 10

(9) 解析 xml 格式的天气信息的代码截图如下图 11 所示：

```
async void WeatherQuery(string str)
{
    //http://api.map.baidu.com/telematics/v3/weather?location=武汉&ak=8IoIaU655sQrs95uMWRWPDia
    string url = "http://api.map.baidu.com/telematics/v3/weather?location=" + str + "&ak=8IoIaU655sQrs95uMWRWPDia";
    HttpClient client = new HttpClient();
    string response = await client.GetStringAsync(url);
    XmlDocument document = new XmlDocument();
    document.LoadXml(response);
    XmlNodeList list = document.GetElementsByTagName("status");
    XmlNode node = list.Item(0);
    string i = node.InnerText;
    if (i == "success")
    {
        list = document.GetElementsByTagName("date");
        node = list.Item(0);
        date.Text = node.InnerText;
        list = document.GetElementsByTagName("weather");
        node = list.Item(0);
        weather.Text = node.InnerText;
        list = document.GetElementsByTagName("wind");
        node = list.Item(0);
        wind.Text = node.InnerText;
        list = document.GetElementsByTagName("temperature");
        node = list.Item(0);
        temperature.Text = node.InnerText;
    } else
    {
        var ii = new MessageBox("No such place!").ShowAsync();
    }
}
```

图 11

使用 HttpClient 获取 xml 格式的信息，然后使用系统自带的解析器来解析 xml 格式的文档，在这里的异常处理是根据图 2 和图 4 中显示出的正确和错误地址时候的信息的差异来获取的，在正确的时候 status 是 success，错误的时候不是，所以就用这个来标明地址是否合理，能否查询到正确的信息。

四 . 实验过程遇到的问题

本次实验不算难，没有遇到什么问题。基本上看完 ta 推荐的链接就可以完成了。

五 . 思考与总结

本次实验学会了自己申请 API 然后利用自己申请的 API 获取相应的信息 然后使用 xml 自带的解析器，或者使用 JsonTextReader 来分别解析 xml 和 json 格式的信息，最后

将解析得到的信息显示出来，收获还是挺大的。