2.6 实践项目—爬取天气预报数据

深圳信息职业技术学院
Shenzhen Institute Of Information Technology
教师：黄锐军

目 录

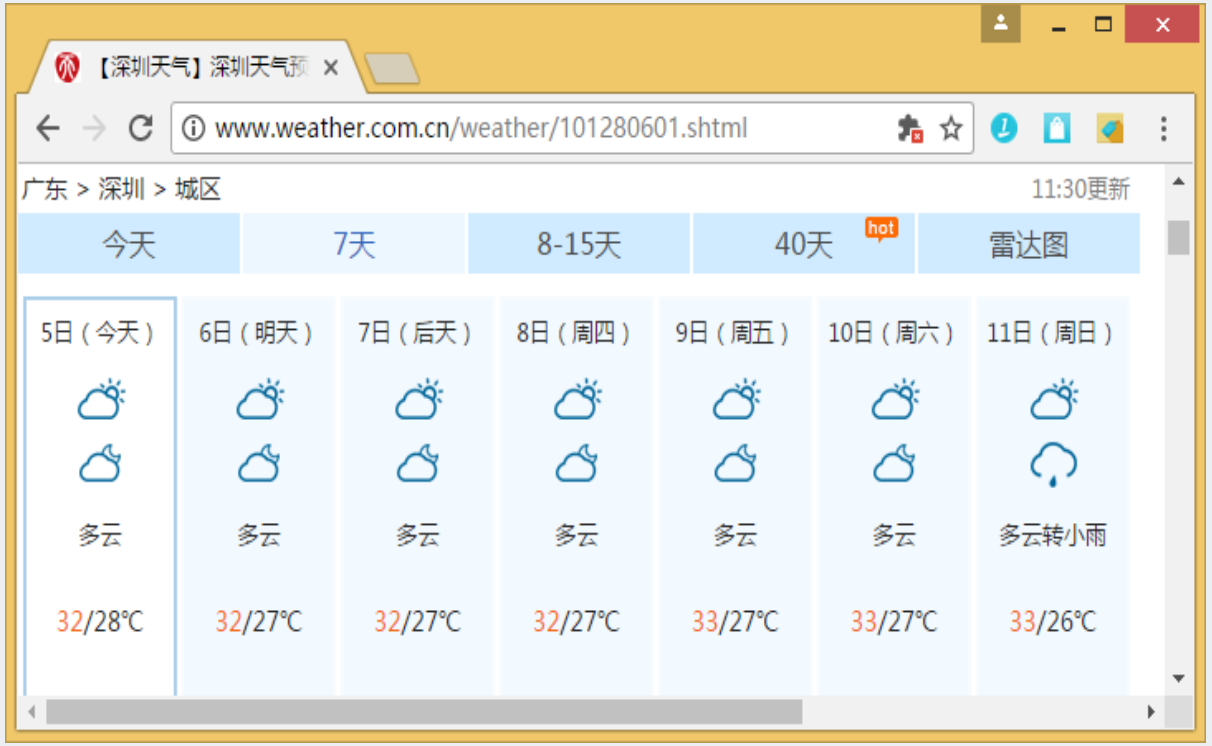
COMPANY

PART ONE

项目简介

项目简介

在中国天气网(http://www.weather.com.cn)中输入一个城市的名称，例如输入深圳，那么会转到地址http://www.weather.com.cn/weather1d/101280601.shtml的网页显示深圳的天气预报，其中101280601是深圳的代码，每个城市或者地区都有一个代码。如图2-6-1、2-6-2所示

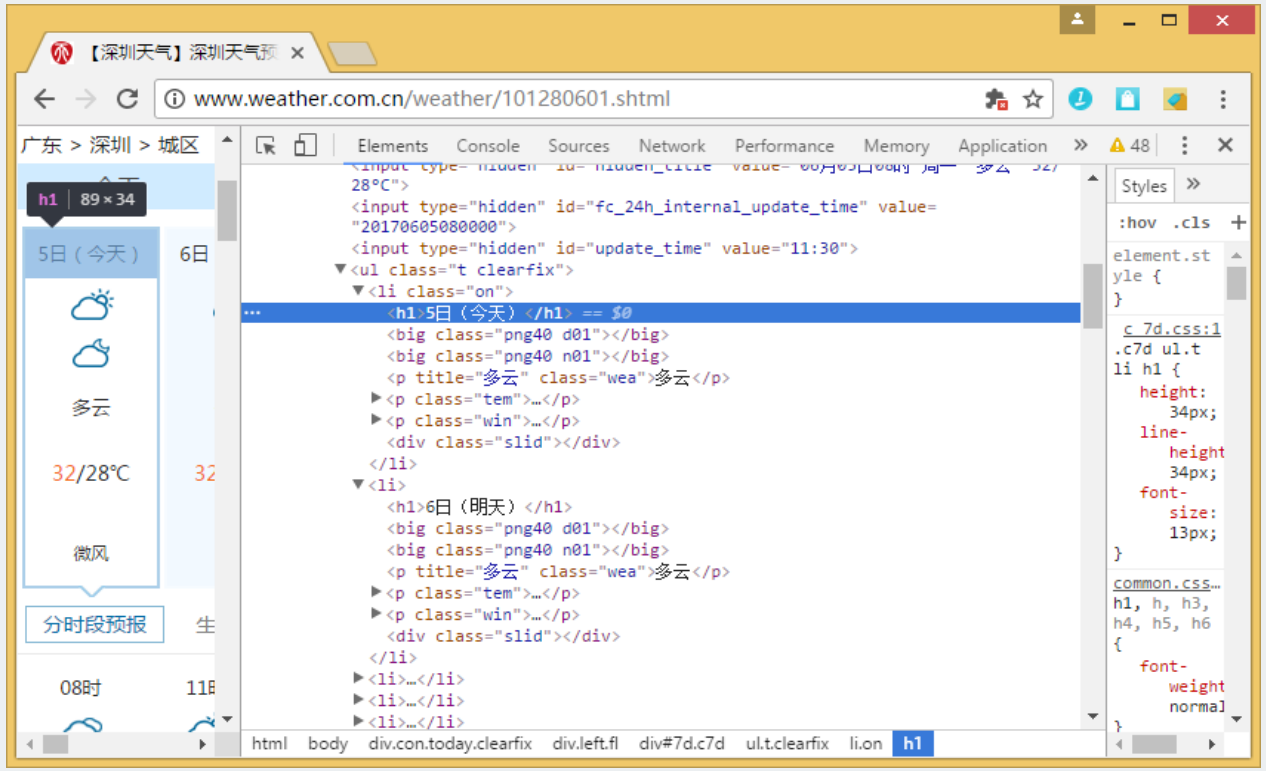

【深圳天气】深圳天气预
www.weather.com.cn/weather/101280601.shtml
广东 > 深圳 > 城区
11:30更新
今天
7天
8-15天
40天
hot
雷达图
5日（今天）
6日（明天）
7日（后天）
8日（周四）
9日（周五）
10日（周六）
11日（周日）
多云
多云
多云
多云
多云
多云
多云转小雨
32/28℃
32/27℃
32/27℃
32/27℃
33/27℃
33/27℃
33/26℃

PART TWO

HTML代码分析

HTML代码分析

用Chrome浏览器浏览网站，鼠标指向7天天气预报的今天位置，点击右键弹出菜单，选择“检查”就可以打开这个位置对应的HTML代码，如图2-6-3所示。



选择<ul class="t clearfix">元素，点击右键弹出菜单选择"Edit as HTML"，就可以进入编辑状态，复制整个HTML，结果如下：

```
<ul class="t clearfix">
<li class="on">
<h1>5日（今天）</h1>
<big class="png40 d01"></big>
<big class="png40 n01"></big>
<p title="多云" class="wea">多云</p>
<p class="tem">
<span>32</span>/<i>28℃</i>
</p>
<p class="win">
<em>
<span title="无持续风向" class=""></span>
<span title="无持续风向" class=""></span>
</em>
```

```
<i>微风</i>
</p>
<div class="slid"></div>
</li>
<li>
<h1>6日（明天）</h1>
<big class="png40 d01"></big>
<big class="png40 n01"></big>
<p title="多云" class="wea">多云</p>
<p class="tem">
<span>32</span>/<i>27℃</i>
</p>
<p class="win">
<em>
<span title="无持续风向" class=""></span>
<span title="无持续风向" class=""></span>
</em>
<span title="无持续风向"
class=""></span>
<span title="无持续风向"
class=""></span>
</em>
<i>微风</i>
</p>
<div class="slid"></div>
</li>
<li>
<h1>7日（后天）</h1>
<big class="png40 d01"></big>
<big class="png40 n01"></big>
<p title="多云" class="wea">多云</p>
<p class="tem">
```

```
<span>32</span>/<i>27℃</i>
</p>
<p class="win">
<em>
<span title="无持续风向" class=""></span>
<span title="无持续风向" class=""></span>
</em>
<i>微风</i>
</p>
<div class="slid"></div>
</li>
<li>
<h1>8日（周四）</h1>
<big class="png40 d01"></big>
<big class="png40 n01"></big>
<p title="多云" class="wea">多云</p>
```

```
<p class="tem">
<span>32</span>/<i>27℃</i>
</p>
<p class="win">
<em>
<span title="无持续风向"
class=""></span>
<span title="无持续风向"
class=""></span>
</em>
<i>微风</i>
</p>
<div class="slid"></div>
</li>
<li>
<h1>9日（周五）</h1>
```

```
<big class="png40 d01"></big>
<big class="png40 n01"></big>
<p title="多云" class="wea">多云</p>
<p class="tem">
<span>33</span>/<i>27℃</i>
</p>
<p class="win">
<em>
<span title="无持续风向" class=""></span>
<span title="无持续风向" class=""></span>
</em>
<i>微风</i>
</p>
<div class="slid"></div>
</li>
<li>
```

```
<h1>10日（周六）</h1>
<big class="png40 d01"></big>
<big class="png40 n01"></big>
<p title="多云" class="wea">多云</p>
<p class="tem">
<span>33</span>/<i>27℃</i>
</p>
<p class="win">
<em>
<span title="无持续风向"
class=""></span>
<span title="无持续风向"
class=""></span>
</em>
<i>微风</i>
</p>
```

```
<div class="slid"></div>
</li>
<li>
<h1>11日（周日）</h1>
<big class="png40 d01"></big>
<big class="png40 n07"></big>
<p title="多云转小雨" class="wea">多云转小雨</p>
<p class="tem">
<span>33</span>/<i>26℃</i>
</p>
<p class="win">
<em>
<span title="无持续风向" class=""></span>
<span title="无持续风向" class=""></span>
</em>
<i>微风</i>
</p>
<div class="slid"></div>
</li>
</ul>
```

PART Three

爬取天气预报数据

select查找子孙节点

在select(css)中的css有多个节点时，节点元素之间用空格分开，就是查找子孙节点，例如soup.select("div p")是查找所有 节点下面的所有子孙<p>节点。

例2-5-3：查找子孙节点

```
from bs4 import BeautifulSoup
doc="<div><p>A</p><span><p>B</p></span></div><div><p>C</p>
</div>"
soup=BeautifulSoup(doc,"lxml")
tags=soup.select("div p")
for tag in tags:
    print(tag)
```

程序结果：

http://example.com/elsie

http://example.com/lacie

http://example.com/tillie

另外我们通过

tags=soup.select("p a")

tags=soup.select("a")

tags=soup.select("p[class] a")

等也可以得到一样的结果。

因此：

- soup.select("a[href='http://example.com/elsie']") 查找href="http://example.com/elsie"的<a>节点；
- soup.select("a[href$='sie']") 查找href以"sie"结尾的<a>节点；
- soup.select("a[href^='http://example.com']") 查找href以"http://example.com"开始的<a>节点；
- soupselect("a[href*='example']") 查找href的值中包含"example"字符串的<a>节点；

程序结果：

A

B

C

其中tags=soup.select("div p")是查找 下面的所有子孙节点 ，因此包含下面的 B 。

PART Four

爬取与储存天气预报数据

爬取天气预报数据

通过分析HTML代码，我们可以编写爬取的程序爬取深圳7天的天气预报数据：

```
from bs4 import BeautifulSoup
from bs4 import UnicodeDammit
import urllib.request

url="http://www.weather.com.cn/weather/101280601.shtml"
try:
    headers={"User-Agent":"Mozilla/5.0 (Windows; U; Windows NT 6.0 x64; en-US; rv:1.9pre) Gecko/2008072421 Minefield/3.0.2pre"}
    req=urllib.request.Request(url,headers=headers)
    data=urllib.request.urlopen(req)
    data=data.read()
```

```
    dammit=UnicodeDammit(data,["utf-8","gbk"])
     data=dammit.unicode_markup
     soup=BeautifulSoup(data,"lxml")
     lis=soup.select("ul[class='t clearfix'] li")
     for li in lis:
         try:
             date=li.select('h1')[0].text
             weather=li.select('p[class="wea"]')[0].text
             temp=li.select('p[class="tem"]
span')[0].text+"/"+li.select('p[class="tem"] i')[0].text
             print(date,weather,temp)
         except Exception as err:
             print(err)
except Exception as err:
    print(err)
```

程序爬取结果：

5日（今天）多云 32/28℃

6日（明天）多云 32/27℃

7日（后天）多云 32/27℃

8日（周四）多云 32/27℃

9日（周五）多云 33/27℃

10日（周六）多云 33/27℃

11日（周日）多云转小雨 33/26℃

由此可见爬取的数据与我们直接从网站看到的是一样的。

PART Five

爬取与储存天气预报数据

爬取与储存天气预报数据

我们可以获取北京、上海、广州、深圳等城市的代码，爬取这些城市的天气预报数据，并存储到sqllite数据库weathers.db中，存储的数据表weathers是：

create table weathers (wCity varchar(16),wDate varchar(16),wWeather varchar(64),wTemp varchar(32),constraint pk_weather primary key (wCity,wDate))"

编写程序依次爬取各个城市的天气预报数据存储在数据库中，程序如下：

```
from bs4 import BeautifulSoup
from bs4 import UnicodeDammit
import urllib.request
import sqlite3

class WeatherDB:
    def openDB(self):
        self.con=sqlite3.connect("weathers.db")
        self.cursor=self.con.cursor()
        try:
            self.cursor.execute("create table weathers (wCity varchar(16),wDate
varchar(16),wWeather varchar(64),wTemp varchar(32),constraint pk_weather
primary key (wCity,wDate))")
        except:
            self.cursor.execute("delete from weathers")
```

```
    def closeDB(self):
        self.con.commit()
        self.con.close()
def insert(self,city,date,weather,temp):
        try:
            self.cursor.execute("insert into weathers (wCity,wDate,wWeather,wTemp)
values (?,?,?,?)" ,(city,date,weather,temp))
        except Exception as err:
            print(err)

    def show(self):
        self.cursor.execute("select * from weathers")
        rows=self.cursor.fetchall()
        print("%-16s%-16s%-32s%-16s" % ("city","date","weather","temp"))
        for row in rows:
            print("%-16s%-16s%-32s%-16s" % (row[0],row[1],row[2],row[3]))
```

```
class WeatherForecast:
    def __init__(self):
        self.headers = {
            "User-Agent": "Mozilla/5.0 (Windows; U; Windows NT 6.0 x64; en-US;
rv:1.9pre) Gecko/2008072421 Minefield/3.0.2pre"}
        self.cityCode={"北京":"101010100","上海":"101020100","广州":"101280101","
深圳":"101280601"}

def forecastCity(self,city):
        if city not in self.cityCode.keys():
            print(city+" code cannot be found")
            return

        url="http://www.weather.com.cn/weather/"+self.cityCode[city]+".shtml"
        try:
            req=urllib.request.Request(url,headers=self.headers)
```

```
            data=urllib.request.urlopen(req)
            data=data.read()
            dammit=UnicodeDammit(data,["utf-8","gbk"])
            data=dammit.unicode_markup
            soup=BeautifulSoup(data,"lxml")
            lis=soup.select("ul[class='t clearfix'] li")

for li in lis:
                try:
                    date=li.select('h1')[0].text
                    weather=li.select('p[class="wea"]')[0].text
                    temp=li.select('p[class="tem"]
span')[0].text+"/"+li.select('p[class="tem"] i')[0].text
                    print(city,date,weather,temp)
                    self.db.insert(city,date,weather,temp)
                except Exception as err:
```

```
                print(err)
        except Exception as err:
            print(err)

def process(self,cities):
        self.db=WeatherDB()
        self.db.openDB()

        for city in cities:
            self.forecastCity(city)

        #self.db.show()
        self.db.closeDB()

ws=WeatherForecast()
ws.process(["北京","上海","广州","深圳"])
print("completed")
```

北京 7日（今天）晴间多云，北部山区有阵雨或雷阵雨转晴转多云 31℃/17℃
北京 8日（明天）多云转晴，北部地区有分散阵雨或雷阵雨转晴 34℃/20℃
北京 9日（后天）晴转多云 36℃/22℃
北京 10日（周六）阴转阵雨 30℃/19℃
北京 11日（周日）阵雨 27℃/18℃
北京 12日（周一）阴转晴 28℃/20℃
北京 13日（周二）晴 32℃/21℃
上海 7日（今天）多云 30/21℃
上海 8日（明天）多云转阴 32/23℃
上海 9日（后天）阵雨 32/24℃
上海 10日（周六）中雨 27/22℃
上海 11日（周日）小雨转多云 29/22℃
上海 12日（周一）多云 30/22℃
上海 13日（周二）多云转阴 30/21℃

广州 7日（今天）多云 35/27℃
广州 8日（明天）多云 35/28℃
广州 9日（后天）多云 35/28℃
广州 10日（周六）多云 35/28℃
广州 11日（周日）多云 35/28℃
广州 12日（周一）雷阵雨 35/27℃
广州 13日（周二）雷阵雨转大雨 33/24℃
深圳 7日（今天）阵雨转多云 34/28℃
深圳 8日（明天）晴 34/28℃
深圳 9日（后天）晴 34/28℃
深圳 10日（周六）晴转阵雨 34/28℃
深圳 11日（周日）阵雨 33/27℃
深圳 12日（周一）阵雨 32/27℃
深圳 13日（周二）阵雨转中雨 32/25℃

THANK YOU