Python第六组汇报

天气网爬虫

CONTENT

1

成员介绍

吴坤：软件工程，组长，负责编写代码

董江枫：网络与新媒体，负责讲解

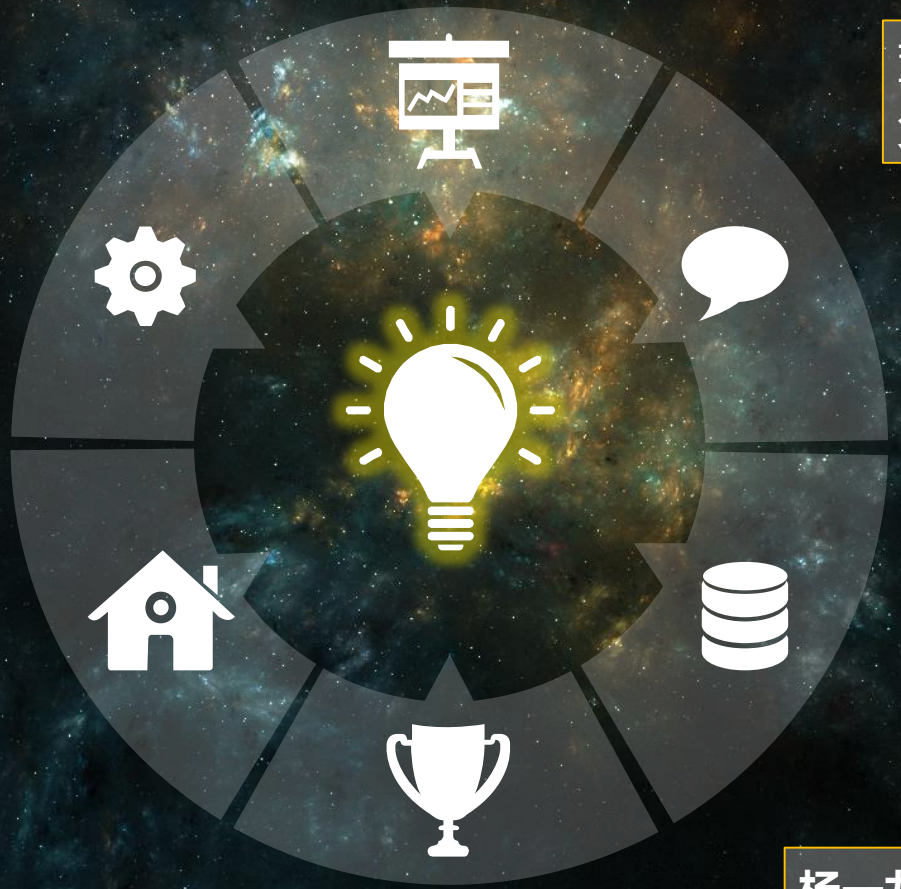
李金阳：计算机科学与技术，负责编写代码

杨登富：计算机科学与技术，负责实验报告

赵张毅：材料物理，负责测试填表

张基伟：软件工程，负责代码调试

杨一龙：统计学，负责PPT制作



2 功能演示

22	2日（今天	多云		6	
23	3日（明天	阵雨转阴	14	7	
24	4日（后天	多云	12	3	
25	5日（周二	多云转晴	9	2	
26	6日（周三	多云	11	4	
27	7日（周四	多云	12	3	
28	8日（周五	多云	9	2	

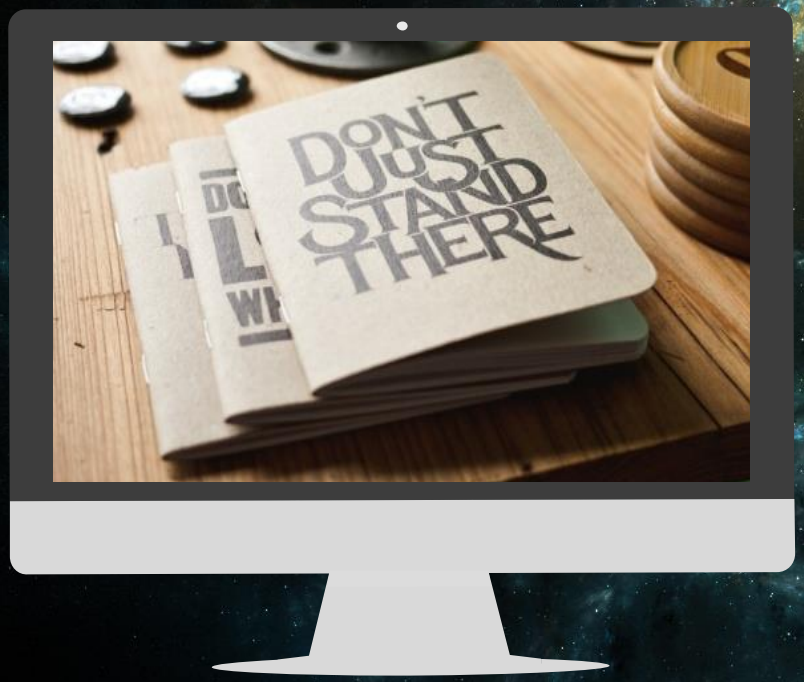
weather

	A	B	C	D	E	F
1	27日（今天	多云		12		
2	28日（明天	小雨	17	12		
3	29日（后天	小雨	13	8		
4	30日（周四	阴转多云	11	7		
5	1日（周五	多云转阴	12	6		
6	2日（周六	阴转多云	13	8		
7	3日（周日	多云转小雨	16	9		

weather

	A	B	C	D	E	F
1	12日（今天	晴		3		
2	13日（明天	多云转阴	10	4		
3	14日（后天	阴	9	6		
4	15日（周五	小雨	10	5		
5	16日（周六	中到大雨转	7	0		
6	17日（周日	多云转晴	6	-2		
7	18日（周一	多云	8	-1		

3 总体设计



网络爬虫是一个自动提取网页的程序，它为搜索引擎从万维网上下载网页，是搜索引擎的重要组成部分，传统网络爬虫是一个从一个或很多初始网页上的URL，在抓取网页的过程中，不断从当前页面上抽取新的URL放入队列，直到满足系统的一定条件。

我们可以利用网络爬虫爬取网页中的信息，并且保存下来，以便查看或使用。使用python编写网络爬虫更加有效率并且简便。

4

特色和创新点

1、对网页进行抓取时没使用Scrapy框架

2、对网页进行抓取使用了Beautiful Soup，能够抓取天气网上的近几天的天气报告，并将数据保存为csv格式，能够查阅

天气网爬虫.py - C:\Users\home-pc\Desktop\天气网爬虫.py

File Edit Format Run Options Windows Help

```
# coding : UTF-8
import requests
import csv
import random
import time
import socket
import http.client
# import urllib.request
from bs4 import BeautifulSoup

def get_content(url , data = None):
    header={
        'Accept': 'text/html,application/xhtml+xml,application/xml;q=0.9,image/w
        'Accept-Encoding': 'gzip, deflate, sdch',
        'Accept-Language': 'zh-CN,zh;q=0.8',
        'Connection': 'keep-alive',
        'User-Agent': 'Mozilla/5.0 (Windows NT 6.3; WOW64) AppleWebKit/537.36 (K
    }
    timeout = random.choice(range(80, 180))
    while True:
        try:
            rep = requests.get(url,headers = header,timeout = timeout)
            rep.encoding = 'utf-8'
            # req = urllib.request.Request(url, data, header)
            # response = urllib.request.urlopen(req, timeout=timeout)
            # html1 = response.read().decode('UTF-8', errors='ignore')
            # response.close()
            break
        # except urllib.request.HTTPError as e:
        #         print( '1:', e)
        #         time.sleep(random.choice(range(5, 10)))
        #
        # except urllib.request.URLError as e:
        #     print( '2:', e)
        #     time.sleep(random.choice(range(5, 10)))
        except socket.timeout as e:
            print( '3:', e)
            time.sleep(random.choice(range(8,15)))

        except socket.error as e:
```

Ln: 1 Col: 0

特色和创新点

天气网爬虫.py - C:\Users\home-pc\Desktop\天气网爬虫.py

File Edit Format Run Options Windows Help

```
            print( '4:', e)
            time.sleep(random.choice(range(20, 60)))

        except http.client.BadStatusLine as e:
            print( '5:', e)
            time.sleep(random.choice(range(30, 80)))

        except http.client.IncompleteRead as e:
            print( '6:', e)
            time.sleep(random.choice(range(5, 15)))

    return rep.text
    # return html_text
def get_data(html_text):
    final = []
    bs = BeautifulSoup(html_text, "html.parser")  # 创建BeautifulSoup对象
    body = bs.body # 获取body部分
    data = body.find('div', {'id': '7d'})  # 找到id为7d的div
    ul = data.find('ul')  # 获取ul部分
    li = ul.find_all('li')  # 获取所有的li

    for day in li: # 对每个li标签中的内容进行遍历
        temp = []
        date = day.find('h1').string  # 找到日期
        temp.append(date)  # 添加到temp中
        inf = day.find_all('p')  # 找到li中的所有p标签
        temp.append(inf[0].string,)  # 第一个p标签中的内容（天气状况）加到temp中
        if inf[1].find('span') is None:
            temperature_highest = None # 天气预报可能没有当天的最高气温（到了傍晚，就是这
        else:
            temperature_highest = inf[1].find('span').string  # 找到最高温
            temperature_highest = temperature_highest.replace('℃', '')  # 到了晚
        temperature_lowest = inf[1].find('i').string  # 找到最低温
        temperature_lowest = temperature_lowest.replace('℃', '')  # 最低温度后面有
        temp.append(temperature_highest)   # 将最高温添加到temp中
        temp.append(temperature_lowest)   #将最低温添加到temp中
        final.append(temp)   #将temp加到final中

    return final
def write_data(data, name):
```

Ln: 6 Col: 0

```
    file_name = name
    with open(file_name, 'a', errors='ignore', newline='') as f:
            f_csv = csv.writer(f)
            f_csv.writerows(data)

if __name__ == '__main__':
    url ='http://www.weather.com.cn/weather/101190401.shtml'
    html = get_content(url)
    result = get_data(html)
    write_data(result, 'weather.csv')
```

Ln: 63 Col: 0

THANK YOU!

PRESENTED BY OFFICEPLUS