# **网络爬虫**

## **实验目的：**

学习网络爬虫编写，获取数据

## **实验要求：**

能运用request库和beautifulSoup4访问URL并解析获取的HTML

能向百度等搜索引擎自动提交关键词并获取返回结果

## **实验内容：**

完成教材第十章课后练习10.1、10.2、10.6

## **实验代码：**

10.1

import requests  
from bs4 import BeautifulSoup  
allUniv=[]  
def getHTMLText(url):  
    try:  
        r = requests.get(url, timeout=30)  
        r.raise\_for\_status()  
        r.encoding = 'utf-8'  
        return r.text  
    except:  
        return ""  
​  
def fillUnivList(soup):  
    data = soup.find\_all('tr')  
    for tr in data:  
        ltd = tr.find\_all('td')  
        if len(ltd ) == 0:  
            continue  
        singleUniv = []  
        for td in ltd:  
            singleUniv.append(td.string)  
​  
        allUniv.append(singleUniv)  
​  
def printUnivList(province):  
    print("{1:^2}{2:{0}^10}{3:{0}^6}{4:{0}^4}{5:{0}^10}".format(chr(12288),"排名","学校名称","省市","总分","培养规模"))  
    for u in allUniv:  
        if province in u[2]:  
            print("{1:^2}{2:{0}^10}{3:{0}^6}{4:{0}^4}{5:{0}^10}".format(chr(12288),u[0],u[1],u[2],u[3],u[4]))  
​  
​  
def main(p):  
    url = 'http://www.zuihaodaxue.cn/zuihaodaxuepaiming2018.html'  
    html = getHTMLText(url)  
    soup = BeautifulSoup(html,'html.parser')  
    fillUnivList(soup)  
    printUnivList(p)  
​  
​  
main('江西')  
​

10.2

import requests  
import re  
from bs4 import BeautifulSoup  
​  
def getHTMLText(url):  
    send\_headers = {  
        "User-Agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/61.0.3163.100 Safari/537.36",  
        "Connection": "keep-alive",  
        "Accept": "text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,\*/\*;q=0.8",  
        "Accept-Language": "zh-CN,zh;q=0.8"}  
    try:  
        r = requests.get(url, headers=send\_headers)  
        r.raise\_for\_status()  
        print(r.status\_code)  
        r.encoding = 'utf-8'  
        return r.text  
    except:  
        return ""  
​  
def fillUnivList(soup, allUniv):  
    data = soup.find\_all('div',{'class':re.compile('shadow-dark')})  
    for div in data:  
        singleUniv = []  
        div1 = div.find('div',{'style':'margin-left: 2.5rem;'})  
        rank = div1.get\_text().strip()  
        singleUniv.append(rank.split(' ')[0])  
​  
        h3 = div.find('h3')  
        singleUniv.append(h3.get\_text().strip())  
​  
        ldiv = div.find\_all('div',{'style':'padding-right: 0.5rem;'})  
        singleUniv.append(ldiv[0].strong.string)  
        singleUniv.append(ldiv[1].strong.string)  
        allUniv.append(singleUniv)  
​  
def printUnivList(allUniv):  
    print("{:<6}{:<20}{:<6}{:<10}".format("排名","学校名称","学费","培养规模"))  
    for u in allUniv:  
        print("{:<6}{:<20}{:<10}{:<10}".format(u[0],u[1],u[2],u[3]))  
​  
​  
def main(num):  
    allUniv = []  
    url = 'https://www.usnews.com/best-colleges/rankings/national-universities'  
​  
    for i in range(1, num+1):  
        ri = url + '?\_page=' + str(i)  
        html = getHTMLText(ri)  
        soup = BeautifulSoup(html, 'html.parser')  
        fillUnivList(soup, allUniv)  
    printUnivList(allUniv)  
​  
​  
main(4)  
​

1.3

import requests  
import json  
import re  
import os  
index = 1  
​  
def downloadImageFile(imgUrl, destUrl, fname=''):  
    local\_filename = imgUrl.split('/')[-1]  
    print('Download Image File={}'.format(local\_filename))  
    try:  
        r = requests.get(imgUrl, stream=True)  
        r.raise\_for\_status()  
​  
        if len(fname) == 0:  
            fname = local\_filename  
        print('fname={}'.format(fname))  
        with open(destUrl + "/" + fname, 'wb') as f:  
            for chunk in r.iter\_content(chunk\_size=1024):  
                if chunk:  
                    f.write(chunk)  
                    f.flush()  
            f.close()  
        return r.status\_code  
    except:  
        return r.status\_code  
​  
def download(urls,path):  
    global index  
    for url in urls:  
        print("Download Image from page:{}".format(url))  
        status = downloadImageFile(url,path,str(index)+".jpg")  
        try:  
            if str(status)[0] == '4':  
                print("未下载成功{}".format(url))  
                continue  
        except Exception as e:  
            print("未下载成功{}".format(url))  
        index += 1  
​  
page = 'https://image.baidu.com/search/acjson?tn=resultjson\_com&ipn=rj'\  
       '&ct=201326592&is=&fp=result&queryWord=杨幂&cl=&lm=&hd=&latest='\  
       '&copyright=&ie=utf-8&oe=utf-8&adpicid=&st=&z=&ic='\  
       '&word=杨幂&s=&se=&tab=&width=&height=&face=&istype='\  
       '&qc=&nc=&fr=&pn=30&rn=30&gsm=1e&1545721051065='  
try:  
    rsp = requests.get(page,timeout=10)  
    rsp.raise\_for\_status()  
except:  
    print('对不起，百度图片访问失败！程序退出')  
      
imgdata=json.loads(rsp.text)  
imgs = imgdata['data']  
​  
urls=[]  
for im in imgs:  
    url = im.get('thumbURL')  
    print(url)  
    if url is not None:  
        urls.append(im.get('thumbURL'))   
​  
download(urls, 'e:/baidupic')

## **实验总结：**

我们都知道，只要有意义，那么就必须慎重考虑。 就我个人来说，网络爬虫学习对我的意义，不能不说非常重大。 歌德说过一句富有哲理的话，没有人事先了解自己到底有多大的力量，直到他试过以后才知道。这句话语虽然很短，但令我浮想联翩。 所谓网络爬虫学习，关键是网络爬虫学习需要如何写。 问题的关键究竟为何？ 亚伯拉罕·林肯曾经说过，你活了多少岁不算什么，重要的是你是如何度过这些岁月的。带着这句话，我们还要更加慎重的审视这个问题： 莎士比亚曾经提到过，本来无望的事，大胆尝试，往往能成功。这似乎解答了我的疑惑。 对我个人而言，网络爬虫学习不仅仅是一个重大的事件，还可能会改变我的人生。 经过上述讨论， 可是，即使是这样，网络爬虫学习的出现仍然代表了一定的意义。 卡莱尔在不经意间这样说过，过去一切时代的精华尽在书中。这似乎解答了我的疑惑。 而这些并不是完全重要，更加重要的问题是， 亚伯拉罕·林肯曾经说过，我这个人走得很慢，但是我从不后退。这似乎解答了我的疑惑。 了解清楚网络爬虫学习到底是一种怎么样的存在，是解决一切问题的关键。 了解清楚网络爬虫学习到底是一种怎么样的存在，是解决一切问题的关键。 一般来说， 在这种困难的抉择下，本人思来想去，寝食难安。 我认为， 了解清楚网络爬虫学习到底是一种怎么样的存在，是解决一切问题的关键。 我们都知道，只要有意义，那么就必须慎重考虑。 要想清楚，网络爬虫学习，到底是一种怎么样的存在。 生活中，若网络爬虫学习出现了，我们就不得不考虑它出现了的事实。 我们不得不面对一个非常尴尬的事实，那就是， 就我个人来说，网络爬虫学习对我的意义，不能不说非常重大。 史美尔斯说过一句富有哲理的话，书籍把我们引入最美好的社会，使我们认识各个时代的伟大智者。这似乎解答了我的疑惑。 可是，即使是这样，网络爬虫学习的出现仍然代表了一定的意义。 既然如何， 一般来说， 史美尔斯说过一句富有哲理的话，书籍把我们引入最美好的社会，使我们认识各个时代的伟大智者。这句话语虽然很短，但令我浮想联翩。 网络爬虫学习，到底应该如何实现。 现在，解决网络爬虫学习的问题，是非常非常重要的。