**数据结构实验报告12**

**学号：** 117060400204 **姓名**： 李涛 **班级：**应用统计2班

**指导老师：** 林卫中

**实验名称**： 第十二次上机实验

**实验要求：**网络爬虫

**实验题目：程序练习题**

**算法实现：**

**一**

**# -\*- coding: utf-8 -\*-**

**"""**

**Created on Fri Jun 8 14:33:04 2018**

**@author: Administrator**

**"""**

**import requests**

**import re**

**from bs4 import BeautifulSoup**

**allUniv=[]**

**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):**

**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])**

**univName = div.find('h3')**

**singleUniv.append(univName.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():**

**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():**

**url = 'https://www.usnews.com/best-colleges/rankings/national-universities?\_page='**

**for i in range(1,4):**

**ri = url + str(i)**

**html = getHTMLText(ri)**

**soup = BeautifulSoup(html,'html.parser')**

**fillUnivList(soup)**

**printUnivList()**

**main()**

二

import requests

def getHTMLText(url,coding='gbk'):

try:

r = requests.get(url,timeout=30)

print(r)

r.raise\_for\_status()

r.encoding = coding

return r.text

except:

return ""

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

三

from bs4 import BeautifulSoup

import re

import requests

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 getMorePages(kw, pages):

params = []

for i in range(30, 30\*pages+30, 30):

params.append({

'ipn': 'rj',

'ct': 201326592,

'is': '',

'fp': 'result',

'queryWord': kw,

'cl': 2,

'lm': -1,

'ie': 'utf-8',

'oe': 'utf-8',

'adpicid': '',

'st': -1,

'z': '',

'ic': 0,

'word': kw,

's': '',

'se': '',

'tab': '',

'width': '',

'height': '',

'face': 0,

'istype': 2,

'qc': '',

'nc': 1,

'fr': '',

'pn': i,

'rn': 30,

'gsm': '1e',

'1528253616462': ''

})

url = 'https://image.baidu.com/search/acjson?tn=resultjson\_com'

datalist = []

for param in params:

dj = requests.get(url, params=param).json()

data = dj['data']

if data is not None and len(data) > 0:

datalist.append(data)

return datalist

def main(kw, pages, desurl):

datalist = getMorePages(kw, pages)

index = 1

for data in datalist:

for i in data:

if i.get('thumbURL') is not None:

ir = i.get('thumbURL')

downloadImageFile(ir, desurl, str(index)+'.jpg')

index = index + 1

main('范冰冰',3, ‘e:/baidupic')

**四**

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:

s = u[2].split(' ')

f = s[0].replace(',','')

f = f.replace('$','')

if int(f) < 50000:

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(3)

五

import requests

import re

from bs4 import BeautifulSoup

allUniv=[]

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):

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])

univName = div.find('h3')

singleUniv.append(univName.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():

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():

url = 'https://www.usnews.com/best-colleges/rankings/national-universities'

html = getHTMLText(url)

soup = BeautifulSoup(html,'html.parser')

fillUnivList(soup)

printUnivList()

main()

六

from NetSpider import \*

from bs4 import BeautifulSoup

import re

def getImg(html):

imgre = re.compile('"objURL":"(.\*?)"')

imglist = re.findall(imgre,html)

return imglist

def download(urls,path):

index = 1

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/index?tn=baiduimage&word=范冰冰'

html= getHTMLText(page,'utf-8')

download(getImg(html),’e:/360Downloads/baidupic')

七

import requests

from selenium import webdriver

from selenium.webdriver.common.by import By

from NetSpider import \*

from bs4 import BeautifulSoup

import re

allMusics=[]

url = 'http://music.163.com/#/discover/playlist/?order=hot&cat=%E5%85%A8%E9%83%A8&limit=35&offset=0'

def getHTMLTextByHeadless(url):

broswer = webdriver.Firefox()

while url != 'javascript:void(0)':

broswer.get(url)

broswer.switch\_to.frame("contentFrame")

data = broswer.find\_element\_by\_id("m-pl-container").find\_elements\_by\_tag\_name("li")

for d in data:

music = []

nb = d.find\_element(By.CLASS\_NAME,'nb').text

if '万' in nb:

n = nb[:-1]

if int(n) > 500:

music.append(n)

al = d.find\_elements\_by\_tag\_name('a')

title = al[0].get\_attribute('title')

music.append(title)

hr = al[0].get\_attribute('href')

music.append(hr)

author = al[3].get\_attribute('title')

music.append(author)

allMusics.append(music)

url = broswer.find\_element\_by\_css\_selector("a.zbtn.znxt").get\_attribute('href')

broswer.close()

def fillMusicList():

html = getHTMLText(url, coding='utf-8')

soup = BeautifulSoup(html,'html.parser')

ul = soup.find('ul', {'id':'m-pl-container'})

divs = ul.find\_all('div', {'class': re.compile('u-cover')})

for div in divs:

music = []

nb = div.find('span', {'class': 'nb'}).string

if '万' in nb:

n = nb[:-1]

if int(n) > 500:

music.append(n)

a = div.find('a')

title = a.title

music.append(title)

hr = musicurl + a.href

music.append(hr)

a2 = div.find('a', {'class':re.compile('nm nm-icn f-thide')})

author = a2.string

music.append(author)

allMusics.append(music)

def printMusicList():

print('{:<8}{:<20}{:<8}{:<20}'.format('下载次数', '歌曲名称', '歌手', '歌曲链接地址'))

for music in allMusics:

print('{:<8}{:<20}{:<8}{:<20}'.format(str(music[0]), music[1], music[3], music[2]))

#fillMusicList()

#printMusicList()

getHTMLTextByHeadless(url)

printMusicList()