Python实验报告12

学号：117060400212 姓名：张佳奇 班级：应用统计学2班

指导老师:林卫中

实验名称：unit10 exercise

实验题目1：

中国大学排名爬虫

算法实现：

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(num):

print("{:^4}{:^10}{:^5}{:^8}{:^10}".format("排名","学校名称","省市","总分","培养规模"))

for i in range(num):

u=allUniv[i]

print("{:^4}{:^10}{:^5}{:^8}{:^10}".format(u[0],u[1],u[2],u[3],u[6]))

def main():

url = 'http://www.zuihaodaxue.cn/zuihaodaxuepaiming2016.html'

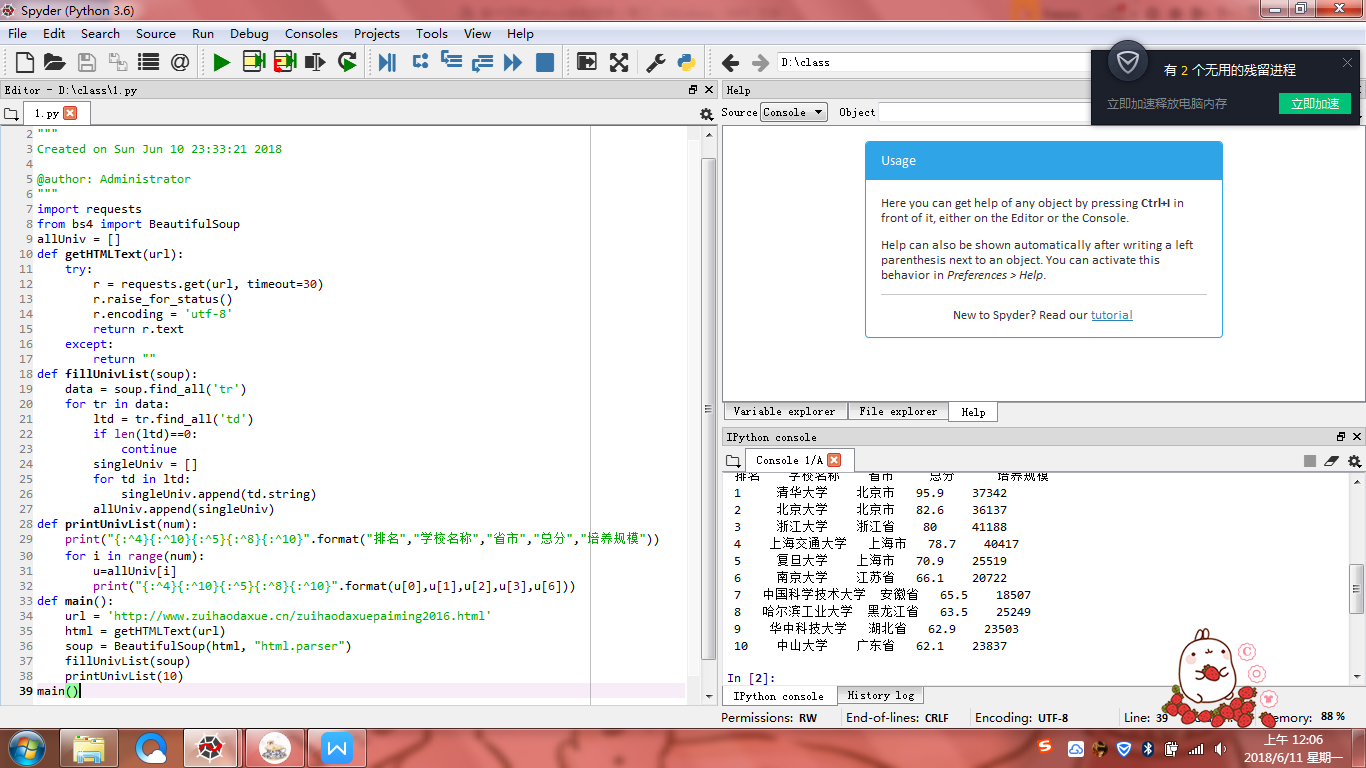
html = getHTMLText(url)

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

fillUnivList(soup)

printUnivList(10)

main()

实验结果：

实验题目2：

按照省份输出中国大学排名

算法实现：

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("{:^4}{:^10}{:^5}{:^8}{:^10}".format("排名","学校名称","省市","总分","培养规模"))

for u in allUniv:

if province in u[2]:

#if float(u[3])>50:

print("{:^4}{:^10}{:^5}{:^8}{:^10}".format(u[0],u[1],u[2],u[3],u[6]))

def main(province):

url = 'http://www.zuihaodaxue.cn/zuihaodaxuepaiming2018.html'

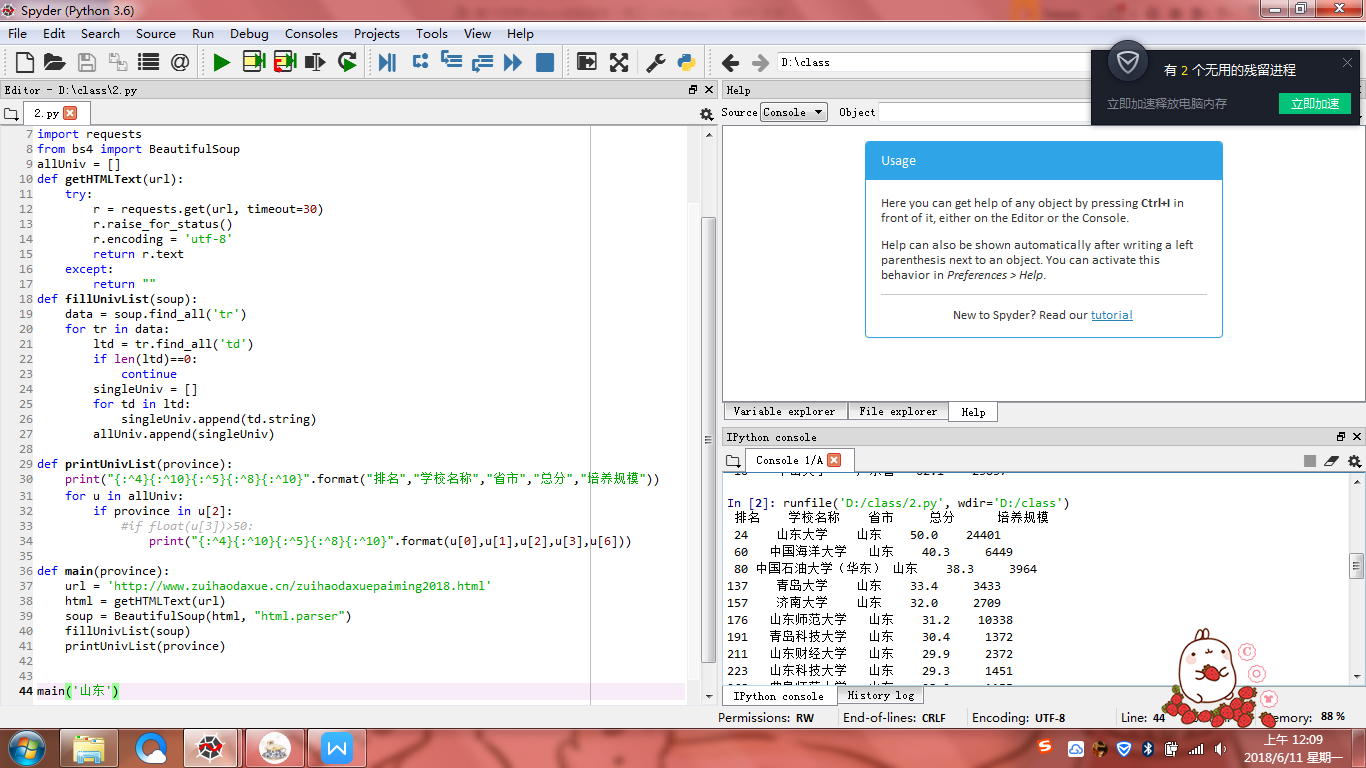
html = getHTMLText(url)

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

fillUnivList(soup)

printUnivList(province)

main('山东')

实验结果：

实验题目3：

美国大学排名爬虫

算法实现：

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 = []

#排名

div1 = div.find('div',{'style':'margin-left: 2.5rem;'})

rank = div1.get\_text().strip()

#学校名称

singleUniv.append(rank.split(' ')[0])

div2 = div.find('h3')

#City

singleUniv.append(div2.get\_text().strip())

address = div.find('div',{'class':re.compile('block-normal')})

singleUniv.append(address.string)

lstrong = div.find\_all('strong')

singleUniv.append(lstrong[0].string)#学费

singleUniv.append(lstrong[1].string)#培养规模

allUniv.append(singleUniv)

def printUnivList():

print("{:<6}{:<20}{:<10}{:<10}{:<10}".format("排名","学校名称","City","学费","培养规模"))

for u in allUniv:

print("{:<6}{:<20}{:<10}{:<10}{:<10}".format(u[0],u[1],u[2],u[3],u[4]))

def main():

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

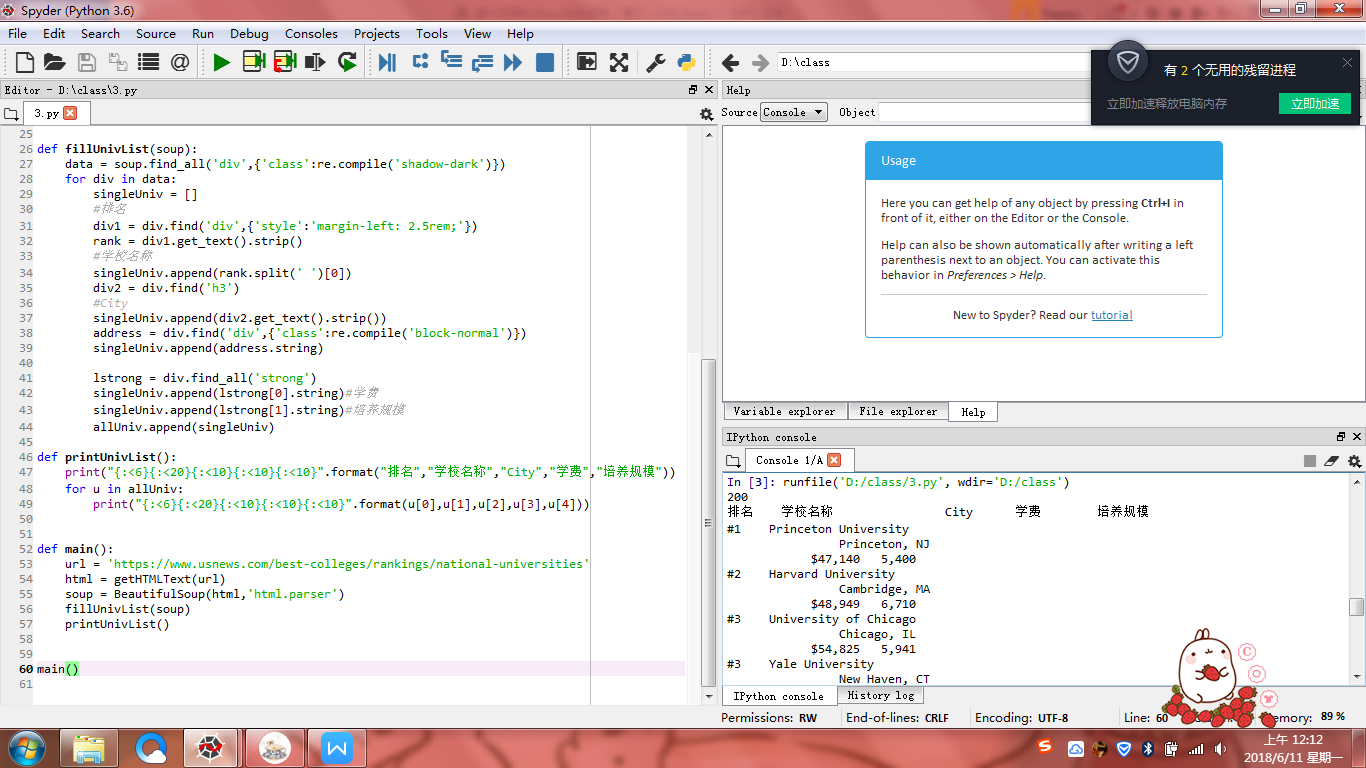
html = getHTMLText(url)

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

fillUnivList(soup)

printUnivList()

main()

实验结果：

实验题目4：

分析百度图片搜索返回结果的HTML代码，编写爬虫抓取图片并下载形成专题图片库

算法实现：

import requests

import re#引用正则表达式的库

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

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:

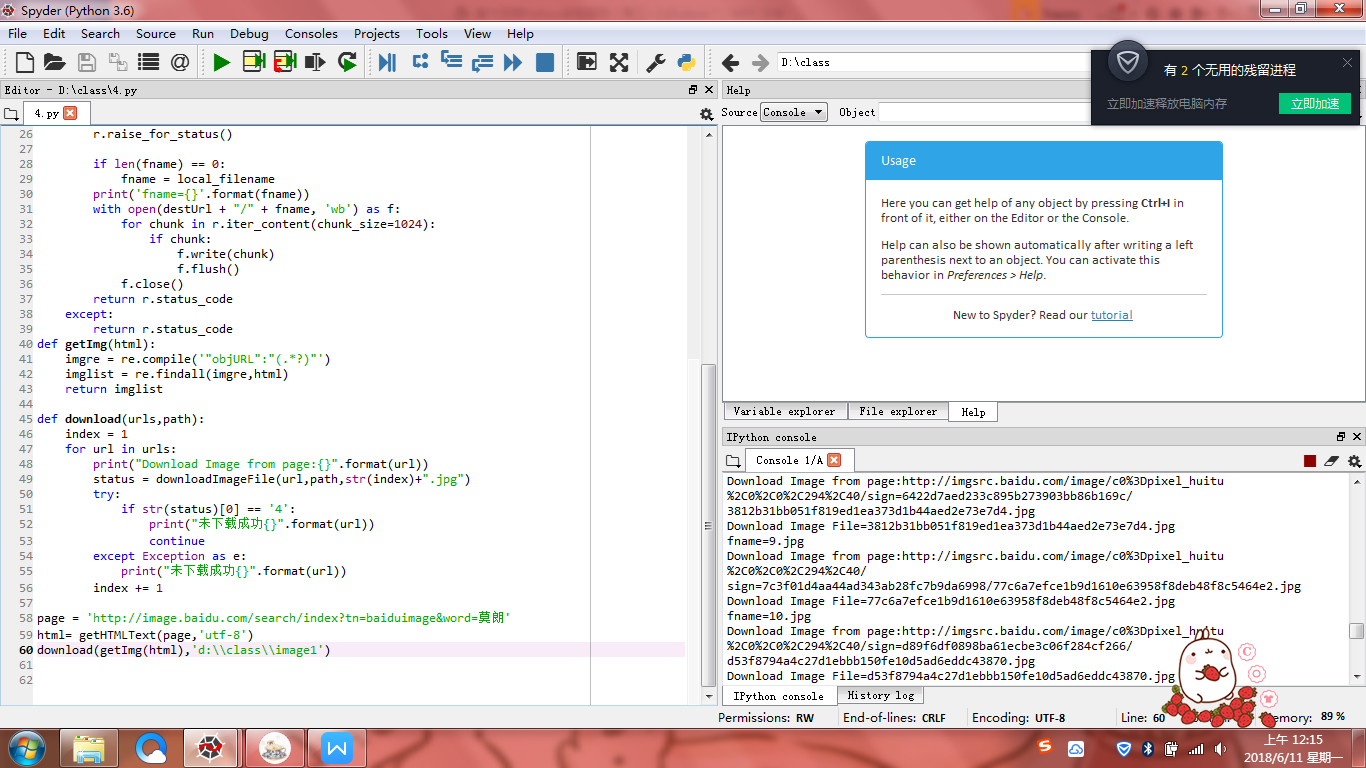
print("未下载成功{}".format(url))

index += 1

page = 'http://image.baidu.com/search/index?tn=baiduimage&word=莫朗'

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

download(getImg(html),'d:\\class\\image1')

实验结果：

小结：这周的代码真的真的很长，就是那种望而生畏的感觉。有了思路之后按照老师教的做起来就会好一些，千万千万要预习，要不然肯定跟不上。还有，老师在讲的时候就不要做了，要认真听。