**实验报告**

**学号：117060400129** **姓名**： 李娟 **班级：** 应用统计一班  **指导老师：** 林卫中

**实验名称**： 课后习题练习

**实验要求：熟练应用程序做题**

**实验题目：爬虫**

**算法实现：大学排名：**

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("{:^2}{:^10}{:^6}{:^4}{:^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]:

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

def main(p):

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

html = getHTMLText(url)

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

fillUnivList(soup)

printUnivList(p)

main('江西')

**反爬虫机制：**

import requests

from bs4 import BeautifulSoup

allUniv = []

def getHTMLText(url):

try:

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

r.raise\_for\_status()

print(r.status\_code)

r.encoding = 'utf-8'

return r.text

except:

print(r.status\_code)

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("{:^2}{:^10}{:^6}{:^4}{:^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]:

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

def main(p):

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

html = getHTMLText(url)

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

fillUnivList(soup)

printUnivList(p)

getHTMLText("https://www.usnews.com/best-colleges/rankings/national-universities")

**USNEWS美国大学排名爬虫：**

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])

div2 = div.find('h3')

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)

singleUniv.append(lstrong[1].string)

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()

**编写爬虫抓取图片：**

import requests

import re

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 = 'https://image.baidu.com/search/index?tn=baiduimage&word=范冰冰'

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

download(getImg(html),'e:\\129\\f')

**美国大学排名前30名的学校**

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,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])

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(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(3)

**学费低于$50000（爬虫）**

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])

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(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)

**爬取图片**

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')

**实验结果：**