**实验报告14**

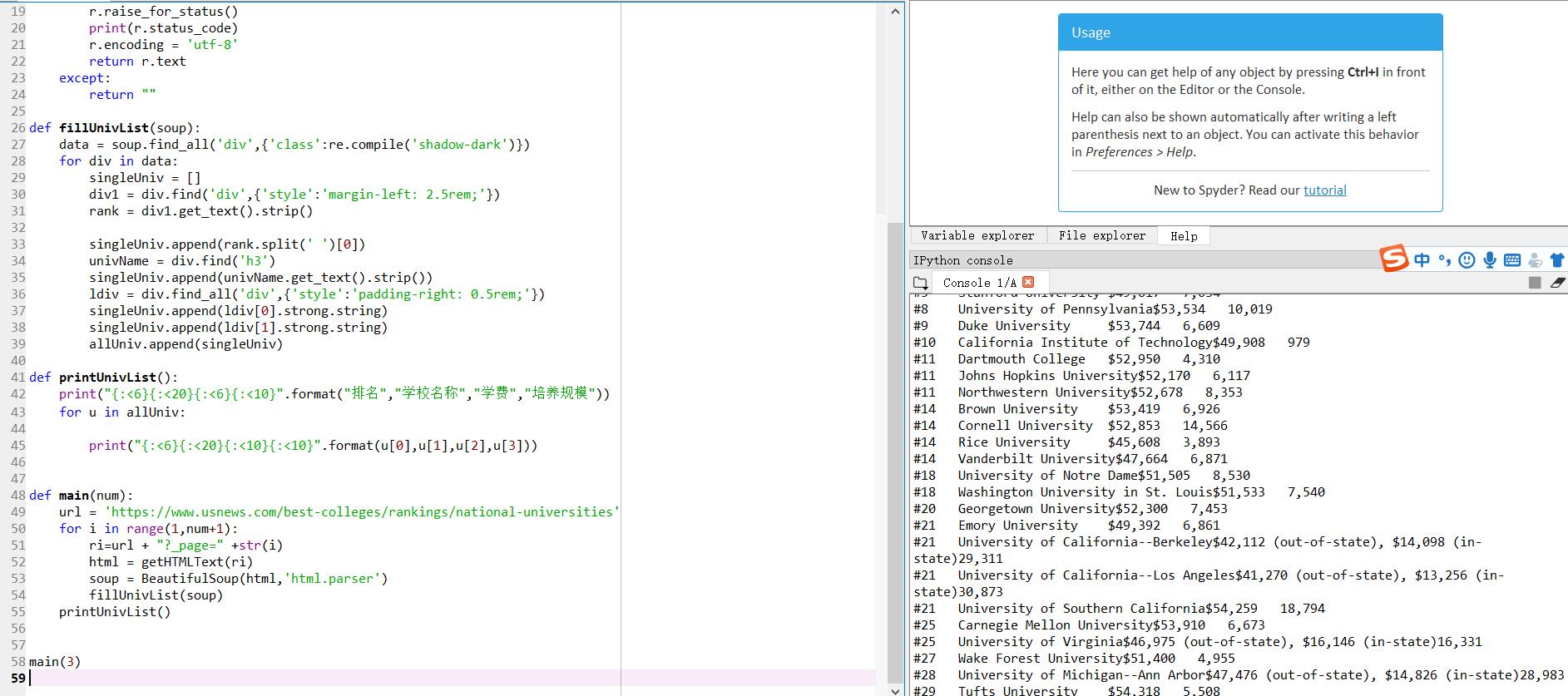
**学号：**117060400123 **姓名**：黄茜洋 **班级：**应统一班**指导老师：**林卫中**实验名称**：第十章网络爬虫和自动化

**实验要求：**（1）掌握网络爬虫的基本方法

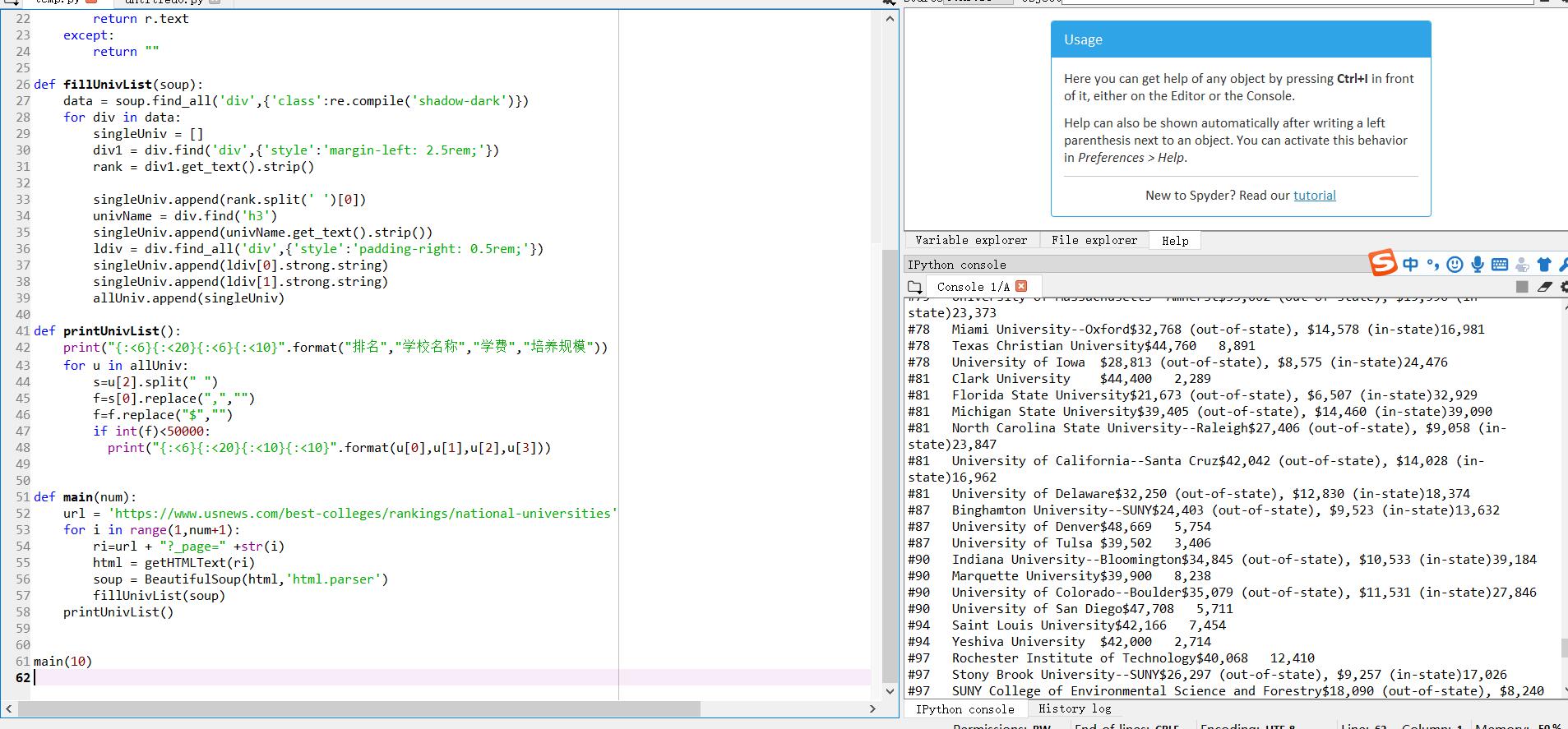
（2）运用beautifulsoup4库解析和处理HTML

**算法实现：**

**1.**

import requestsimport refrom bs4 import BeautifulSoupallUniv=[]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(num): 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) printUnivList()main(3)

2.

import requestsimport refrom bs4 import BeautifulSoupallUniv=[]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: 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): 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) printUnivList()main(10)

3

from selenium import webdriverfrom selenium.webdriver.common.by import Byimport requestsimport timedef 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

from bs4 import BeautifulSoupimport redef getBaiduPicBtHeadless(url, desurl): browser = webdriver.Firefox() browser.get(url) lis = browser.find\_elements(By.CLASS\_NAME, "imgitem") for li in lis: print(li.get\_attribute('data-objurl')) downloadImageFile(li.get\_attribute('data-objurl'), desurl) browser.quit()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 datalistdef 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 + 1main('范冰冰',3, 'D:/baidupic')

实验总结：经过第二次的实验，对爬虫又有了个新的了解，知道了如何用f12查看源代码，也知道了如何下载三四十张图片。