山东大学网络空间安全学院

Python高级程序设计课程实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| 学号：202100460065 | 姓名：李昕 | | 班级：21级密码2班 |
| 实验题目：实验八. urllib模块、requests模块+BeautifulSoup模块使用、Feapder框架 | | | |
| 实验学时：2 | | 实验日期：2022/11/15 | |
| 实验目的：熟悉模块的用法，练习编写爬虫。 | | | |
| 硬件环境：  AMD Ryzen 7 5800H with Radeon Graphics     3.20 GHz  机带 RAM 16.0 GB (13.9 GB 可用 | | | |
| 软件环境：  系统：Windows11  编译器：IDLE | | | |
| 实验步骤与内容：  本次实验包括三个内容：  （1）使用urllib模块或request模块读取网页内容，并利用BeautifulSoup模块进行内容解析，编写爬虫从http://www.cae.cn/cae/html/main/col48/column\_48\_1.html爬取中国工程院院士信息  （2）使用urllib模块或request模块读取网页内容，并利用BeautifulSoup模块进行内容解析，编写爬虫从<https://www.biqukan.com/2_2671>爬取小说《余罪》的内容。要求最终爬取的内容从第一章开始，且不存在大量空格等非必要字符。  （3）学习使用Feapder框架，编写爬虫，爬取中国工程院院士信息。  【BeautifulSoup爬取中国工程院院士信息】  1、beautifulsoup是一个解析器，可以特定的解析出内容，省去了我们编写正则表达式的麻烦。，beautiful Soup自动将输入文档转换为Unicode编码，输出文档转换为utf-8编码。不需要考虑编码方式，除非文档没有指定一个编码方式，这时，Beautiful Soup就不能自动识别编码方式了。  2.使用方法为：   1. BeautifulSoup(markup, “html.parser”)   代码实现：   1. from bs4 import BeautifulSoup 2. import requests 3. import re 4. import sys 5. nums = 0 6. names = [] 7. urls = [] 8. headers = { 9. 'User-Agent':'Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/93.0.4577.82 Safari/537.36' 10. }   *#头部信息* 11. def get\_download\_url(server,target,names,urls): 12. ''' 13. 获取要爬取的链接 14. ''' 15. req = requests.get(url = target,headers = headers, verify =  False ) 16. html = req.text 17. li\_bf = BeautifulSoup(html,features="lxml") 18. li = li\_bf.find\_all('li', class\_ = 'name\_list') 19. *#print(li)* 20. global nums 21. nums = len(li) 22. for each in li: 23. a\_bf = BeautifulSoup(str(each),features="lxml") 24. a = a\_bf.find\_all('a') *#每个院士姓名下就一个<a href...* 25. names.append(a[0].string) *#获取链接的名字* 26. tstr = server + a[0].get('href') *#获取"href"属性的值，即链接* 27. urls.append(tstr) 29. def get\_contents(target): 30. '''函数说明:获取院士信息''' 31. req = requests.get(url=target, headers = headers, ) 32. req.encoding = 'utf-8' 33. *#print(req.encoding)* 34. html = req.text 35. bf = BeautifulSoup(html,features="lxml") 36. div = bf.find\_all('div', class\_ = 'intro') 37. *#print(div)* 38. txt = '' 39. p\_bf = BeautifulSoup(str(div[0]),features="lxml") 40. p = p\_bf.find\_all('p') *#若干<p>…</p>* 41. for each in p: 42. txt += each.text 43. *#print(txt)* 44. return txt 45. def writer(name, filename, text): 46. write\_flag = True 47. with open(filename, 'a', encoding='utf-8') as f: 48. f.write(name + '\n') 49. f.writelines(text) 50. f.write('\n\n') 52. if \_\_name\_\_ == '\_\_main\_\_': 53. pat = r"\xa0+" 54. server = 'http://www.cae.cn' 55. target = 'http://www.cae.cn/cae/html/main/col48/column\_48\_1.html' 56. nmus = 0 57. get\_download\_url(server,target,names,urls) 58. print('工程院院士信息开始下载：') 59. print('nums = ',nums)   执行结果：    【爬取余罪小说内容】  1，通过开发者模式下阅读笔趣看小说网的代码，我们发现文章的所有内容都放在了一个名为div的“东西下面”，这个”东西”就是html标签。HTML标签是HTML语言中最基本的单位，HTML标签是HTML最重要的组成部分。   1. 关键字部分：   <div id="content", class="showtxt">  3，根据关键字部分，我们就可以使用Beautiful Soup提取我们想要的内容，根据关键字中的class，应用BeautifulSoup的select()方法可以得到小说正文的内容。  代码实现：   1. import requests 2. from bs4 import BeautifulSoup 3. import os 4. *# 本地写入* 5. headers={ 6. 'User-Agent': 'Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36' 7. } 8. def getPage(url): 9. re=requests.get(url,headers=headers) 10. re.raise\_for\_status() 11. re.encoding=re.apparent\_encoding 12. soup=BeautifulSoup(re.text,'lxml') 13. return soup 14. if \_\_name\_\_ == '\_\_main\_\_': 15. url='https://www.biqukan8.cc/2\_2671' 16. soup=getPage(url) 17. book\_name=soup.select('.info h2')[0].text 18. book\_author=soup.select('.small span')[0].text 19. last\_time=soup.select('.small .last')[0].text 20. last\_chapter=soup.select('.small .last a')[0].text 21. print("书名:",book\_name,book\_author,last\_time,"最新章节:",last\_chapter) 22. *# 以书名创建文件夹* 23. k=os.path.exists(book\_name) *# k为bool类型* 24. if k==False: 25. *# 没有才创建* 26. os.makedirs(book\_name) 27. *# 获取所有单章链接* 28. links=soup.select('.listmain dd a') 29. *# 前12个为最新章节，其余为正式章节，包含最新章节* 30. for item in links[12:]: 31. href='https://www.biqukan8.cc/'+item['href'] 32. print(href,item.string) 33. soup=getPage(href) 34. content=soup.select('.showtxt')[0].text 35. content=content.replace('app2();read2();  ','') 36. content=content.replace('  ','\n\n') 37. print(">>>正在写入文件") 38. with open("%s/%s.txt"%(book\_name,item.string),'w',encoding='utf-8') as file: 39. file.write(content) 40. print(">>>全部章节爬取完毕！")   运行结果：    【feapder爬取中国工程院院士信息】   1. 创建爬虫的命令行语句为：feapder create -s <spider\_name> ，首先在cmd命令行里创建爬虫项目：     2，在终端中进入到项目（myspider）下的spiders文件夹下，通过下面的命令创建一个目标爬虫文件  3, 下面代码书写在刚刚创建的爬虫文件中.  代码实现：   1. *# -\*- coding: utf-8 -\*-* 2. """ 3. Created on 2022-11-19 22:38:13 4. --------- 5. @summary: 6. --------- 7. @author: waldeinsamkeit 8. """ 9. import feapder 10. import re 12. *#写入文件函数，规定写入的格式* 13. def writer(filename, text): 14. with open(filename, 'a', encoding='utf-8') as f: 15. f.writelines(text) 16. f.write('\n\n') 18. *#轻量级爬虫 AirSpider* 19. class yuanshi(feapder.AirSpider): 20. def start\_requests(self): 21. url='https://www.cae.cn/cae/html/main/col48/column\_48\_1.html' 22. yield feapder.Request(url) 24. *#爬取院士名单和对应的链接地址* 25. def parse\_name(self, response): 26. *#利用xpath过滤得到目标标签* 27. name\_list=response.xpath("//\*[@class='name\_list']") 28. for name in name\_list: 29. *#extract\_first()：这个方法返回的是一个string字符串，是list数组里面的第一个字符串* 30. *#得到对应院士的链接* 31. href=name.xpath('.//@href').extract\_first() 32. yield feapder.Request(href,callback=self.parse\_next) 34. *#爬取院士信息* 35. def parse\_content(self,request,response): 36. *#利用xpath过滤得到目标标签* 37. intro=response.xpath("//\*[@class='intro']") 38. *#extract():这个方法返回的是一个数组list* 39. *#注意这里的解析是Unicode编码* 40. t=intro.xpath(".//p[contains(text(),'\u2002')]").extract() 41. *#转成字符串进行后续处理* 42. tt=''.join(t) 43. regp = r'<p>([\s\S]+?)</p>' 44. intropre = re.compile(regp) 45. introlist=re.findall(intropre,tt) 46. ttt='' 47. for cont in introlist: 48. ttt+=str(cont) 49. *#写入* 50. writer('工程院士信息.txt', ttt) 51. *#print(ttt)* 53. if \_\_name\_\_ == "\_\_main\_\_": 54. yuanshi().start()   运行结果： | | | |
| 结论分析与体会：  通过本次练习，我熟悉了解了Python的爬虫编程，学会了如何根据不同的网站编写不同的爬虫程序，同时我也初步了解了feapder爬虫的使用方法。 | | | |