本节知识点概述

使用Requests爬取豆瓣短评

Requests介绍

Requests安装

如何安装Python第三方库

Requests的简单用法

实战环节

分析豆瓣短评网页

使用Requests下载数据

爬取网页通用框架

一定要知道的爬虫协议

什么是爬虫协议

如何查看爬虫协议

豆瓣的爬虫协议

Requests的使用

Requests: 让 HTTP 服务人类

1、 hon HTTP 库，

2、

3、导入

```
C:\Users\zaoshu>pip install requests
Collecting requests
  Using cached requests-2.18.4-py2.py3-none-any.whl
Requirement already satisfied: idna<2.7,>=2.5 in c:\users\zaoshu\appdata\local\programs\python\python36\lib\site-packages (from
requests)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\zaoshu\appdata\local\programs\python\python36\lib\site-packages (f
rom requests)
Requirement already satisfied: urllib3<1.23,>=1.21.1 in c:\users\zaoshu\appdata\local\programs\python\python36\lib\site-packages
 (from requests)
Requirement already satisfied: chardet<3.1.0,>=3.0.2 in c:\users\zaoshu\appdata\local\programs\python\python36\lib\site-packages
 (from requests)
Installing collected packages: requests
Successfully installed requests-2.18.4

C:\Users\zaoshu>python
Python 3.6.2rc1 (heads/3.6:268e1fb, Jun 17 2017, 19:01:44) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import requests
>>>
```

```
>>> r.encoding
'utf-8'
>>> r.text
u'{"type":"User"...'
>>> r.json()
{u'private_gists': 419, u'total_private_repos': 77, ...}
```

如何安装Python第三方库

1、使

2、下

```
C:\Users\zaoshu\Desktop
λ pip install lxml-3.8.0-cp36-cp36m-win_amd64.whl
Processing c:\users\zaoshu\desktop\lxml-3.8.0-cp36-cp36m-win_amd64.whl
Installing collected packages: lxml
Successfully installed lxml-3.8.0

C:\Users\zaoshu\Desktop
λ python
Python 3.6.2rc1 (heads/3.6:268e1fb, Jun 17 2017, 19:01:44) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import lxml
>>>
```

序包

$ pip install lxml-3.8.0-cp36-cp36m-win_amd64.whl

Requests的简单用法

Requests库的7个主要方法

方法	说明
requests.request()	构造一个请求，支撑以下各方法的基础方法
requests.get()	获取HTML网页的主要方法，对应于HTTP的GET
requests.head()	获取HTML网页头信息的方法，对应于HTTP的HEAD
requests.post()	向HTML网页提交POST请求的方法，对应于HTTP的POST
requests.put()	向HTML网页提交PUT请求的方法，对应于HTTP的PUT
requests.patch()	向HTML网页提交局部修改请求，对应于HTTP的PATCH
requests.delete()	向HTML页面提交删除请求，对应于HTTP的DELETE

我们只需要掌握
requests.get()

Requests.get的用法

导入Requests库

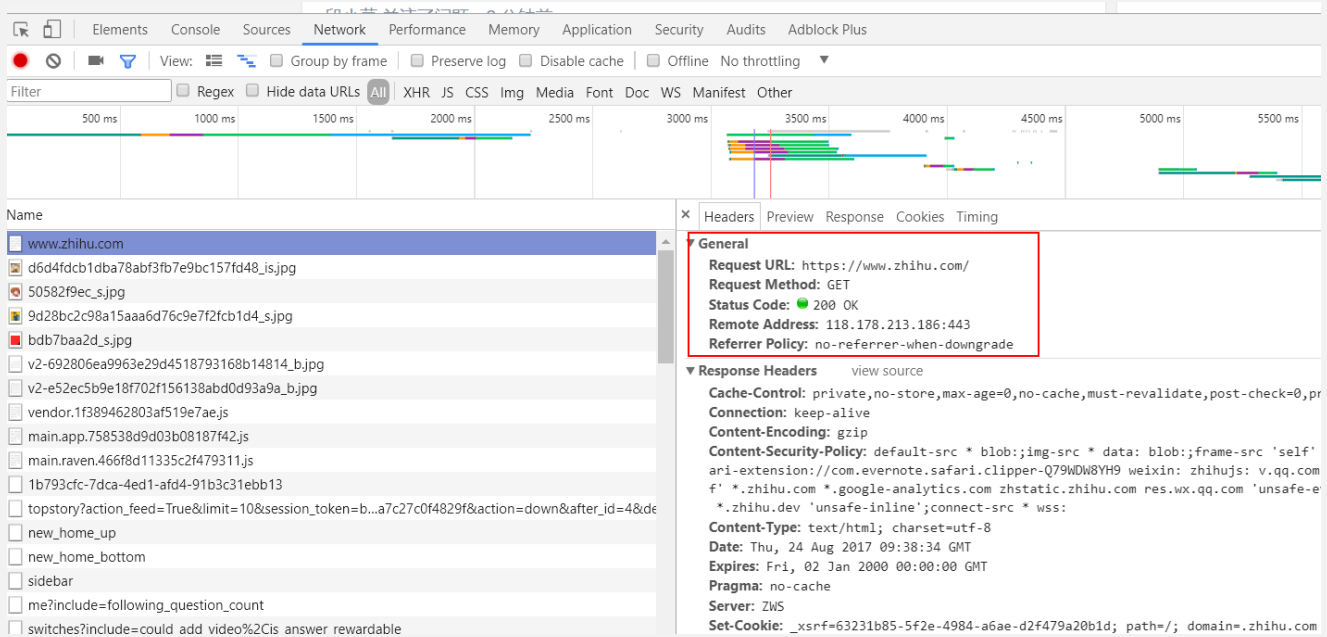
```
>>> import requests

>>> r = requests.get(url)
```

返回包含网页数据的Response

使用get方法发送请求

查看HTTP请求

右键——检查——Network——刷新



Requests的简单用法

Response对象的属性：

- r.status_code　　#http请求的返回状态，200表示连接成功
- r.text　　#返回对象的文本内容
- r.content　　#猜测返回对象的二进制形式
- r.encoding　　#分析返回对象的编码方式
- r.apparent_encoding　　#响应内容编码方式（备选编码方式）

Requests的简单用法

选择一个url，分别打印如下内容

```
>>> import requests
>>> r = requests.get('https://www.zhihu.com/')
>>> r.status_code
500
>>> r.text   #省略
>>> r.content   #省略
>>> r.encoding
'ISO-8859-1'
>>> r.apparent_encoding
'ascii'
```

分析豆瓣短评网页

试着打开这两个网页，阻止JavaScript加载

1. https://book.douban.com/subject/1084336/comments/
2. http://music.163.com

试试你常打开的网页

使用Requests下载数据

```
import requests

url = ' '
r = requests.get(url,timeout=20)
print(r.text)
print(r.raise_for_status())
```

导入Requests库

输入url
使用get方法
打印返回文本
抛出异常

尝试输入一个错误的url，看看什么结果？

豆瓣爬虫

暂停视频

你能否独立写出爬取豆瓣短评的爬虫

https://book.douban.com/subject/1084336/comments/

爬取网页通用框架

```python
import requests

def getHTMLText(url):
    try:
        r = requests.get(url,timeout=20)
        r.raise_for_status()
        r.encoding = r.apparent_encoding
        return r.text
    except:
        return "产生异常"

if __name__ == '__main__':
    url = " "
    print(getHTMLText(url))
```

- 定义函数
- 设置超时
- 异常处理
- 调用函数

爬取网页通用框架

```
import requests

def getHTMLText(url):
        try:
                r = requests.get(url,timeout=20)
                r.raise_for_status()
                r.encoding = r.apparent_encoding
                return r.text
        except:
                return "产生异常"

if __name__ == '__main__':
        url = " "
        print(getHTMLText(url))
```

为你的爬虫加上url
并运行

思考
为什么要这样写

什么是爬虫协议

爬虫协议

也被叫做robots协议，告诉网络蜘蛛哪些页面可以抓取，哪些页面不能抓取

文件规范

- ✓ 必须将robots.txt代码保存为文本文件
- ✓ 必须将该文件保存到网站的顶级目录下
- ✓ robots.txt 文件必须命名为 robots.txt

```
User-agent: yisouspider
Disallow: /baidu
Disallow: /s?
Disallow: /shifen/
Disallow: /homepage/
Disallow: /cpro
Disallow: /ulink?
Disallow: /link?
Disallow: /home/news/data/

User-agent: EasouSpider
Disallow: /baidu
Disallow: /s?
Disallow: /shifen/
Disallow: /homepage/
Disallow: /cpro
Disallow: /ulink?
Disallow: /link?
Disallow: /home/news/data/

User-agent: *
Disallow: /
```

如何查看爬虫协议

https://www.baidu.com/robots.txt

https://www.taobao.com/robots.txt

https://www.zhihu.com/robots.txt

。 。 。

http://eg.com/robots.txt

拦截所有的机器人：
User-agent: *
Disallow: /

允许所有的机器人：
User-agent: *
Disallow:

豆瓣的爬虫协议

我们可以爬豆瓣短评吗？

https://www.douban.com/robots.txt

https://book.douban.com/subject/1084336/comments/

```
User-agent: *
Disallow: /subject_search
Disallow: /amazon_search
Disallow: /search
Disallow: /group/search
Disallow: /event/search
Disallow: /celebrities/search
Disallow: /location/drama/search
Disallow: /forum/
Disallow: /new_subject
Disallow: /service/iframe
Disallow: /j/
Disallow: /link2/
Disallow: /recommend/
Disallow: /trailer/
Disallow: /doubanapp/card
Sitemap: https://www.douban.com/sitemap_index.xml
Sitemap: https://www.douban.com/sitemap_updated_index.xml
# Crawl-delay: 5

User-agent: Wandoujia Spider
Disallow: /
```

爬虫建议

- ✓ 爬取互联网公开数据
- ✓ 尽量放慢你的速度
- ✓ 尽量遵循robots协议
- ✓ 不要用于商业用途
- ✓ 不要公布爬虫程序与数据

百度诉360违反爬虫协议案宣判：360赔偿70万元

2014-08-07 15:30:32 | 来源：中国广播网 | 作者：孙莹 李佳思

关于百度诉360搜索引擎违反Robots协议、不正当竞争纠纷一案，北京第一中级人民法院今天（8月7日）上午十点做出一审判决。判决认为，被告360北京奇虎科技有限公司的行为违反了《反不正当竞争法》相关规定，应赔偿原告北京百度网讯科技有限公司、百度在线网络技术有限公司经济损失以及合理支出共计70万元，同时驳回百度公司的其他诉讼请求。

完成作业

- 掌握Requests基本用法
- 爬取豆瓣及其他网站
- 查看更多网站的robots协议

爬虫工程师

更多数据科学课程，上DC学院： class.pkbigdata.com

关注 DataCastle

关注造数

造数