

第七关（案例实训）发表评论

2022年12月13日 8:43

1. 今天学什么

今天我们学习的目标有三点：

- 1) 学习从开发者工具中找到网页请求，利用 `post()` 函数实现在闪光读书网站中发表评论；
- 2) 从 URL 中寻找规律，结合循环语句实现爬取《乌合之众》的多页评论；
- 3) 对整个爬虫课的知识点进行回顾和梳理，搭建起知识框架。

接下来就开始我们今天的学习吧！

2. 发表评论

还记得网络爬虫的定义吗？网络爬虫是一个能够自动化访问网页的程序。

自动化访问网页的过程中，它并不仅可以爬取网页上的数据，还能模拟网页上的各种功能。

就像我们使用 `requests.post()` 函数实现了登录功能。本节课，我们会尝试使用 `requests.post()` 函数，来实现发表评论功能。

2.1 项目分析与拆解

先试试我写好的代码来发表评论吧。

在运行代码之前，跟我来看看，在网页中，是如何判断发表评论成功的。

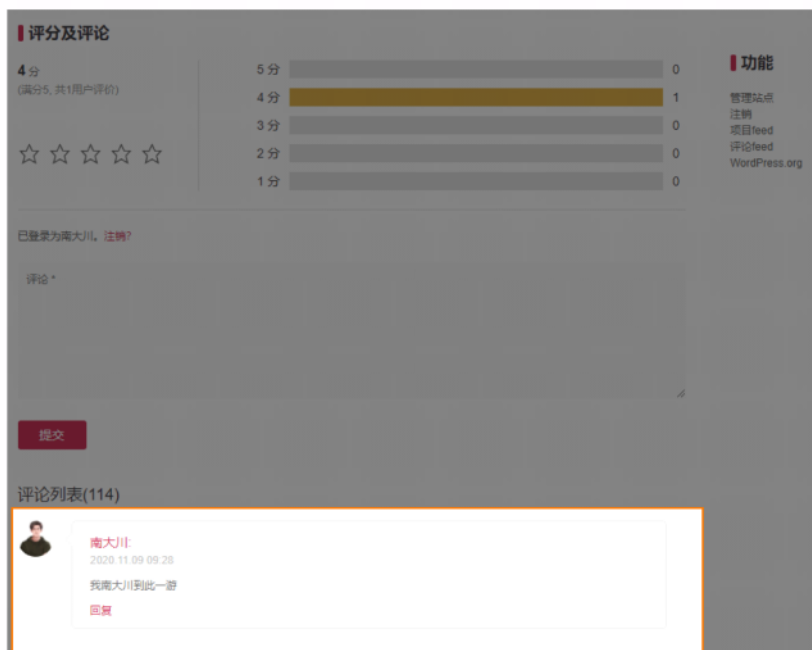
先发布一条评论：

- 1) 打开网页：<https://wp.forchange.cn/psychology/11069/>
- 2) 在网页下面找到评论框（评论相关信息需要登录状态才能看到）
- 3) 在其中输入评论内容，点击提交按钮。



by 风骚编程

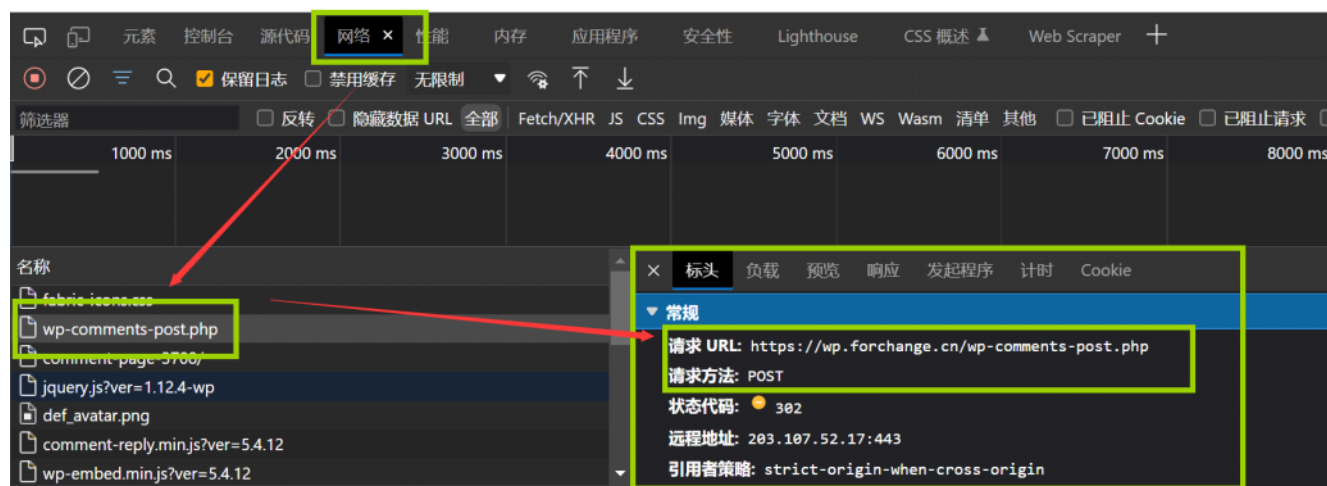
之后如果能在网页中找到刚刚发表的评论信息，就说明发表评论成功了。



by 风变编程

评论递交之前 先打开 开发者模式 找到网络菜单，点击，递交

评论 *

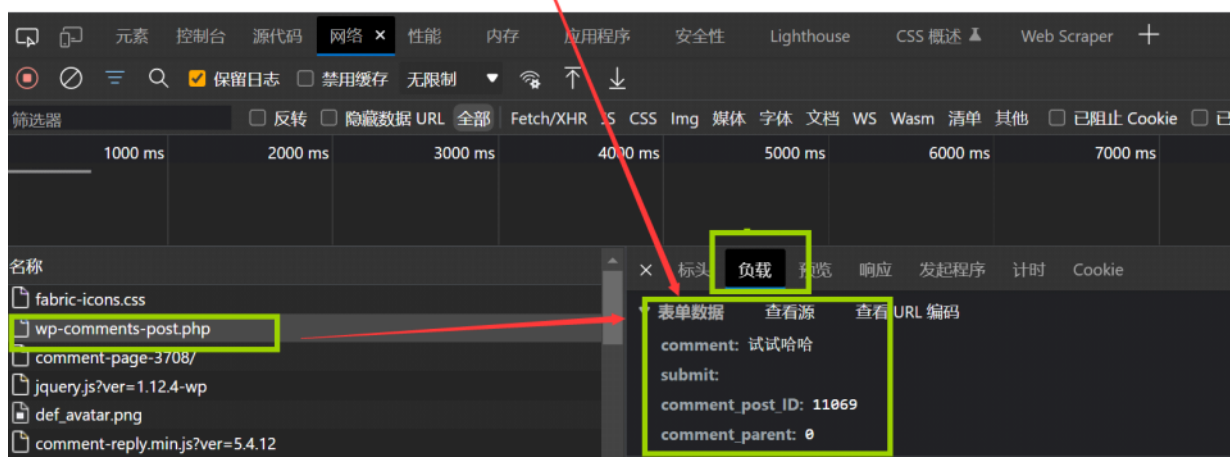


可以看到请求的URL和请求方式。这个一会爬虫需要用到

评论列表(39167)

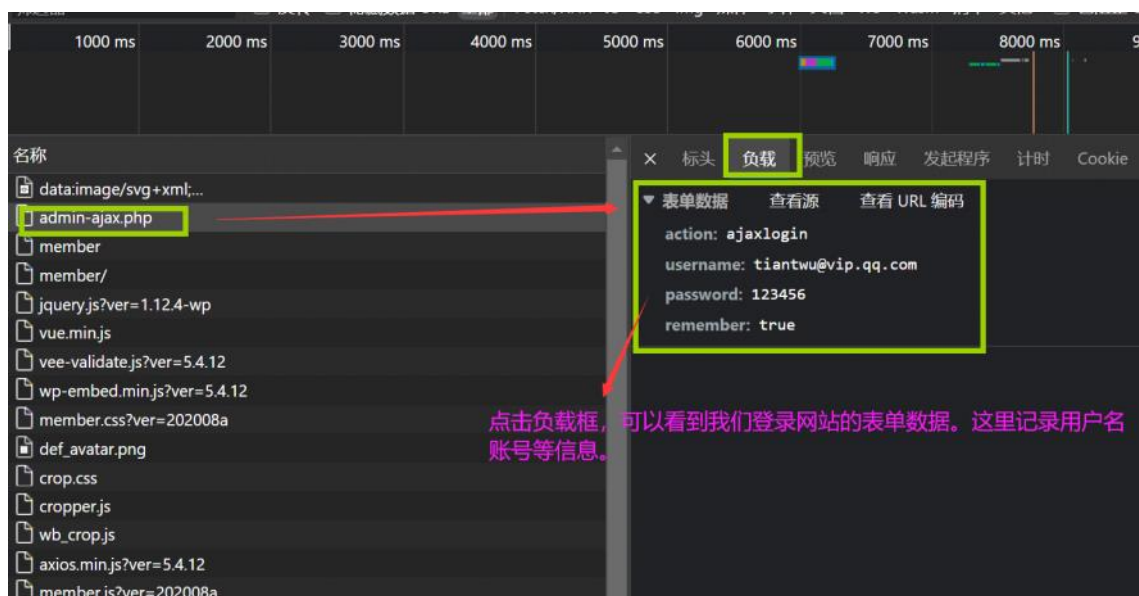


再次点击负载选项卡，可以看到表单数据中comment就是我们刚才发布的评论。这个数据做成字典格式，用于代码发布评论使用。



下面请体验代码，发布一条评论：

在发布评论前，还是需要先登录网站，登录方法和上节课一样。在登录网址中找到表单数据。



点击负载框，可以看到我们登录网站的表单数据。这里记录用户名账号等信息。

所以步骤就是先登录再发布，登录的时候 使用登录的表单数据，发布的时候 使用发布的表单数据

体验代码：

```
# 发表一条评论
import requests
```

```

login_url = 'https://wp.forchange.cn/wp-admin/admin-ajax.php'
username = input('输入用户名: ')
password = input('输入密码: ')
login_data = {
    'action': 'ajaxlogin',
    'username': username,
    'password': password,
    'remember': 'true'
}

# 登录网站
login_res = requests.post(login_url, data = login_data)
if login_res.status_code == 200:
    print('登录成功')
else:
    print('登录失败')

# 设置要请求的书籍评论页链接
comment_url = 'https://wp.forchange.cn/wp-comments-post.php'
# 输入要评论的内容
comment = input('请输入评论内容: ')
# 设置书籍评论页链接中, 请求体的表单数据
comment_data = {
    'comment': comment,
    'submit': '',
    'comment_post_ID': 11069,
    'comment_parent': 0
}

# 请求评论页
comment_res = requests.post(url = comment_url, data=comment_data,
cookies=login_res.cookies)
if comment_res.status_code == 200:
    print('评论成功')
else:
    print('评论失败')

```

```

1  # 发表一条评论
2
3  import requests
4
5  login_url = 'https://wp.forchange.cn/wp-admin/admin-ajax.php'
6
7  username = input('输入用户名: ')
8  password = input('输入密码: ')
9
10 login_data = {
11     'action': 'ajaxlogin',
12     'username': username,
13     'password': password,
14     'remember': 'true'
15 }
16
17 # 登录网站
18 login_res = requests.post(login_url, data = login_data)
19 if login_res.status_code == 200:
20     print('登录成功')
21 else:
22     print('登录失败')

```

```

23
24 # 设置要请求的书籍评论页链接
25 comment_url = 'https://wp.forchange.cn/wp-comments-post.php'
26 # 输入要评论的内容
27 comment = input('请输入评论内容: ')
28
29 # 设置书籍评论页链接中, 请求体的表单数据
30 comment_data = {
31     'comment': comment,
32     'submit': '',
33     'comment_post_ID': 11069,
34     'comment_parent': 0
35 }
36
37 # 请求评论页
38 comment_res = requests.post(url = comment_url, data=comment_data, cookies=login_res.cookies)
39 if comment_res.status_code == 200:
40     print('评论成功')
41 else:
42     print('评论失败')
43

```

```

44
✓ 23.2s
登录成功
评论成功

```

3. 多页评论

现在开始我们第二个项目, 爬取闪光读书网站中书籍的多页评论。

3.1 项目分析与拆解

在开始这个项目之前, 我有几点想跟你说明一下。

- 1) 这次任务是爬取闪光图书网站中的多页评论信息, 并保存到本地, 方便查看。
- 2) 为了方便教学, 所以我只爬取《乌合之众》中前 3 页的评论信息。其中每一页最多有 10 条评论, 前 3 页也就是 30 条评论信息。
- 3) 想看到书籍的评论信息, 需要注册、登录网站。

了解这几点后, 就可以运行一下我写的代码, 看看我是如何将三页评论全部爬取下来的吧。

```

# 爬取多页书籍评论
import requests
from bs4 import BeautifulSoup
import csv

login_url = 'https://wp.forchange.cn/wp-admin/admin-ajax.php'
username = input('输入用户名: ')
password = input('输入密码: ')

# 设置登录表单
login_data = {
    'action': 'ajaxlogin',
    'username': username,
    'password': password,
    'remember': 'true'
}

# 设置空列表用来存储书籍字典
book_list = []

```

```

# 登录网站
login_res = requests.post(login_url, data = login_data)
if login_res.status_code == 200:
    print('登录成功')
else:
    print('登录失败')

# 设置要爬起书籍评论页前3页链接
for i in range(1, 4):
    book_url = f'https://wp.forchange.cn/psychology/11069/comment-page-{i}/'
    #comments'

    # 请求评论页
    book_res = requests.get(url = book_url, cookies=login_res.cookies)
    if book_res.status_code == 200:
        print(f'请求评论第{i}页成功')
    else:
        print('请求评论页失败')

    # 解析评论页
    book_soup = BeautifulSoup(book_res.text, 'html.parser')
    # 获取评论页面的所有评论总和
    book_item = book_soup.find_all('div', class_='comment-txt')
    # 循环遍历 取出里面数据
    for book in book_item:
        name = book.find('div', class_='hd').find('cite', class_='fn').text
        time = book.find('div', class_='hd').find('p', class_='date').text
        content = book.find('div', class_='bd').find('p').text
        # 使用字典来存储数据, 用列表转换数据
        book_dict = {'name':name, 'time':time, 'content':content}
        book_list.append(book_dict)

# 将数据存储到本地
with open(r'D:\PythonTest\风变python学习资料\Python爬虫\多页评论.csv', 'w', encoding='utf-8-sig', newline='') as f:
    book_csv = csv.DictWriter(f, fieldnames=['name', 'time', 'content'])
    book_csv.writeheader()
    book_csv.writerows(book_list)
    print('数据存储成功')

```

```

1 # 爬取多页书籍评论
2
3 import requests
4 from bs4 import BeautifulSoup
5 import csv
6
7 login_url = 'https://wp.forchange.cn/wp-admin/admin-ajax.php'
8
9 username = input('输入用户名: ')
10 password = input('输入密码: ')
11
12 # 设置登录表单
13 login_data = {
14     'action': 'ajaxlogin',
15     'username': username,
16     'password': password,
17     'remember': 'true'
18 }
19 # 设置空列表用来存储书籍字典
20 book_list = []

```



```

22 # 登录网站
23 login_res = requests.post(login_url,data = login_data)
24 if login_res.status_code == 200:
25     print('登录成功')
26 else:
27     print('登录失败')
28

```

```

30 # 设置要爬起书籍评论页前3页链接
31 for i in range(1,4):
32     book_url = f'https://wp.forchange.cn/psychology/11069/comment-page-{i}/#comments'
33
34     # 请求评论页
35     book_res = requests.get(url = book_url, cookies=login_res.cookies)
36     if book_res.status_code == 200:
37         print(f'请求评论第{i}页成功')
38     else:
39         print('请求评论页失败')
40

```

```

41 # 解析评论页
42 book_soup = BeautifulSoup(book_res.text,'html.parser')
43 # 获取评论页面的所有评论总和
44 book_item = book_soup.find_all('div',class_='comment-txt')
45 # 循环遍历 取出里面数据
46 for book in book_item:
47     name = book.find('div',class_='hd').find('cite',class_='fn').text
48     time = book.find('div',class_='hd').find('p',class_='date').text
49     content = book.find('div',class_='bd').find('p').text
50
51 # 使用字典来存储数据,用列表转换数据
52 book_dict = {'name':name,'time':time,'content':content}
53 book_list.append(book_dict)
54

```

```

54
55 # 将数据存储到本地
56 with open(r'D:\PythonTest\风变python学习资料\Python爬虫\多页评论.csv','w',
57         encoding='utf-8-sig',newline='') as f:
58     book_csv = csv.DictWriter(f,fieldnames=['name','time','content'])
59     book_csv.writeheader()
60     book_csv.writerows(book_list)
61     print('数据存储成功')
62

```

✓ 13.1s

登录成功
 请求评论第1页成功
 请求评论第2页成功
 请求评论第3页成功
 数据存储成功

	A	B	C	D
1	name	time	content	
2	此次热可可:	2020.9.17 11:09	乌合之众》中指出个人一旦进入群体中，他的个性便湮灭，群体	
3	happyant521:	2021.2.5 06:02	真的能回复耶! 哈哈	
4	hh163:	2022.6.7 10:06	当之无愧的改变世界的二十本书	
5	15534078485:	2021.3.12 16:03	评论区+1楼	
6	hh163:	2022.6.7 10:06	当之无愧的改变世界的二十本书	
7	xj123456:	2021.5.14 09:05	到此 一游，哈哈哈哈哈	
8	hh163:	2022.6.7 10:06	当之无愧的改变世界的二十本书	
9	tomhanxisi:	2021.6.21 22:06	我tomhanxisi到此一游!!! 学习爬虫	
10	hh163:	2022.6.7 10:06	当之无愧的改变世界的二十本书	
11	tomhanxisi:	2021.6.21 22:06	good!	
12	hh163:	2022.6.7 10:06	当之无愧的改变世界的二十本书	
13	hongwei80:	2021.6.25 14:06	到这来的都目的不纯啊! 哈哈!!!	
14	hh163:	2022.6.7 10:06	当之无愧的改变世界的二十本书	
15	hongwei80:	2021.6.25 14:06	到这来的都目的不纯啊! 哈哈!!!	
16	hh163:	2022.6.7 10:06	当之无愧的改变世界的二十本书	
17	vx15801767605:	2021.6.26 00:06	各位都是来爬虫的吧，来+我vx15801767605，我们建个爬虫群	
	+	...	Sheet1	

4. 总结

今天的项目学习到这就结束了，让我们梳理一下今天学了什么吧!

项目一：发表评论

- 1) 在 **Network** 中找到发表评论对应的请求网址;
- 2) 使用 **requests.post()** 函数带着网址 **url**、请求体 **data** 和储存登录信息的 **cookies** 模拟发表评论。

项目二：爬取多页评论

- 1) 打开网页上多页评论链接，查找规律;
- 2) 通过 **for** 循环+ **字符串.format()** 方法拼接出多页评论的网址;
- 3) 使用 **requests.get()** 函数带着模拟登录闪光读书网站得到的 **cookies** 信息对多页网址发起请求;

最后，是我们本节课使用到的知识总结图：

