

第 8 关、cookies

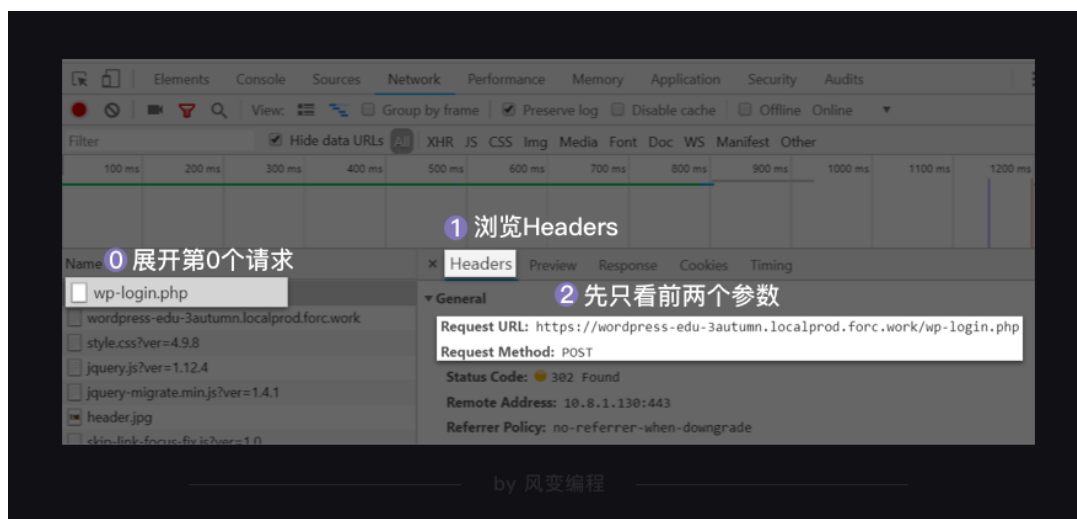
1、项目：发表博客评论

- (1) 网站：<https://wordpress-edu-3autumn.localprod.oc.forchange.cn/wp-login.php>；
- (2) 登录账户：spiderman；
- (3) 登录密码：crawler334566；
- (4) 确定目标：登录博客网站，并在文章下面评论留言。

1-1、post 请求

post 和 get 都可以带着参数请求，不过 get 请求的参数会在 url 上显示出来，而 post 请求的参数就不会直接显示，而是隐藏起来。像账号密码这种私密的信息，一般都是用 post 的请求。可以理解为，get 是明文显示，post 是非明文显示。

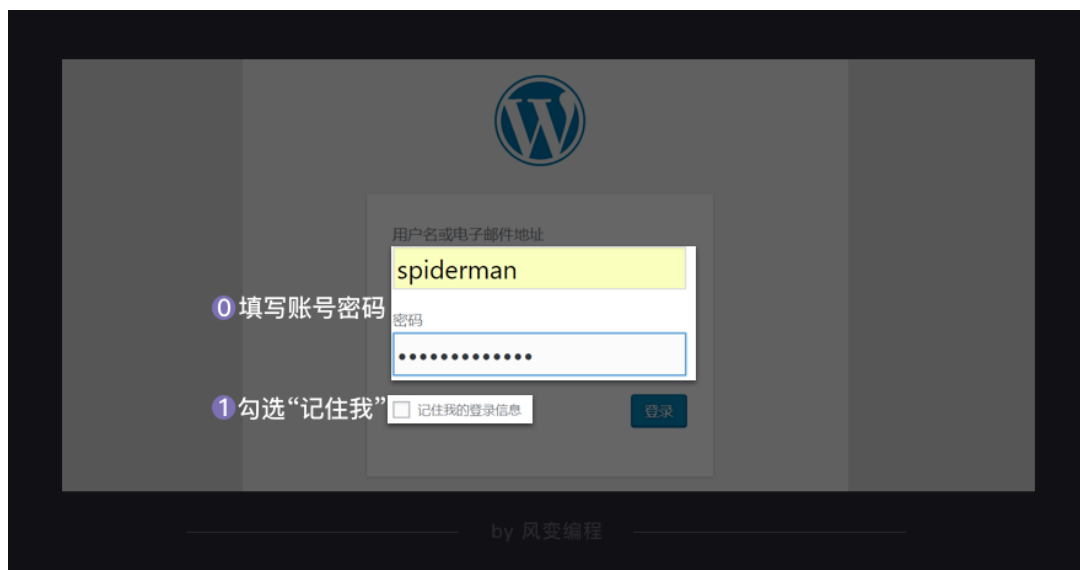
通常，get 请求会应用于获取网页数据，比如我们之前学的 `requests.get()`。post 请求则应用于向网页提交数据，比如提交表单类型数据（像账号密码就是网页表单的数据）。



1-2、cookies 及其用法

一般当你登录一个网站，你都会在登录页面看到一个可勾选的选项“记住我”，如果你勾选了，以后你再打开这个网站就会自动登录，这就是 cookie 在起作用。

cookies也是有时效性的，过期后就会失效。



```
1 import requests
2 #引入requests。
3 url = 'https://wordpress-edu-3autumn.localprod.oc.forchange.cn/wp-
  login.php'
4 #把请求登录的网址赋值给url。
5 headers = {
6 'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
  (KHTML, like Gecko) Chrome/70.0.3538.110 Safari/537.36'
7 }
8 #加请求头,前面有说过加请求头是为了模拟浏览器正常的访问,避免被反爬虫。
9 data = {
10 'log': 'spiderman', #写入账户
11 'pwd': 'crawler334566', #写入密码
12 'wp-submit': '登录',
13 'redirect_to': 'https://wordpress-edu-
  3autumn.localprod.oc.forchange.cn/wp-admin/',
14 'testcookie': '1'
15 }
16 #把有关登录的参数封装成字典,赋值给data。
17 login_in = requests.post(url,headers=headers,data=data)
18 #用requests.post发起请求,放入参数:请求登录的网址、请求头和登录参数,然后赋值给
  login_in。
19 cookies = login_in.cookies
20 #提取cookies的方法:调用requests对象(login_in)的cookies属性获得登录的
  cookies,并赋值给变量cookies。
21
22 url_1 = 'https://wordpress-edu-3autumn.localprod.oc.forchange.cn/wp-
  comments-post.php'
23 #我们想要评论的文章网址。
24 data_1 = {
25 'comment': input('请输入你想要发表的评论: '),
26 'submit': '发表评论',
27 'comment_post_ID': '13',
28 'comment_parent': '0'
29 }
30 #把有关评论的参数封装成字典。
31 comment = requests.post(url_1,headers=headers,data=data_1,cookies=cookies)
```

```
32 #用requests.post发起发表评论的请求，放入参数：文章网址、headers、评论参数、
    cookies参数，赋值给comment。
33 #调用cookies的方法就是在post请求中传入cookies=cookies的参数。
34 print(comment.status_code)
35 #打印出comment的状态码，若状态码等于200，则证明我们评论成功。
```

1-3、session 及其用法

所谓的会话，你可以理解成我们用浏览器上网，到关闭浏览器的这一过程。session 是会话过程中，服务器用来记录特定用户会话的信息。

session 和 cookies 的关系 —— cookies 中存储着 session 的编码信息，session 中又存储了 cookies 的信息。

【session 处理 cookies】

高级用法

本文档涵盖了 Requests 的一些高级特性。

会话对象

会话对象让你能够跨请求保持某些参数。它也会在同一个 Session 实例发出的所有请求之间保持 cookie，期间使用 urllib3 的 [connection pooling](#) 功能。所以如果你向同一主机发送多个请求，底层的 TCP 连接将会被重用，从而带来显著的性能提升。（参见 [HTTP persistent connection](#)）。

会话对象具有主要的 Requests API 的所有方法。

我们来跨请求保持一些 cookie:

```
s = requests.Session()

s.get('http://httpbin.org/cookies/set/sessioncookie/123456789')
r = s.get("http://httpbin.org/cookies")

print(r.text)
# '{"cookies": {"sessioncookie": "123456789"}}'
```

by 风变编程

```
1 import requests
2 #引用requests。
3 session = requests.session()
4 #用requests.session()创建session对象，相当于创建了一个特定的会话，帮我们自动保持
  了cookies。
5 url = 'https://wordpress-edu-3autumn.localprod.oc.forchange.cn/wp-
  login.php'
6 headers = {
7     'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36
  (KHTML, like Gecko) Chrome/71.0.3578.98 Safari/537.36'
8 }
9 data = {
10     'log': input('请输入账号: '), #用input函数填写账号和密码，这样代码更优雅，而不
    是直接把账号密码填上去。
11     'pwd': input('请输入密码: '),
12     'wp-submit': '登录',
13     'redirect_to': 'https://wordpress-edu-
    3autumn.localprod.oc.forchange.cn/wp-admin/',
```

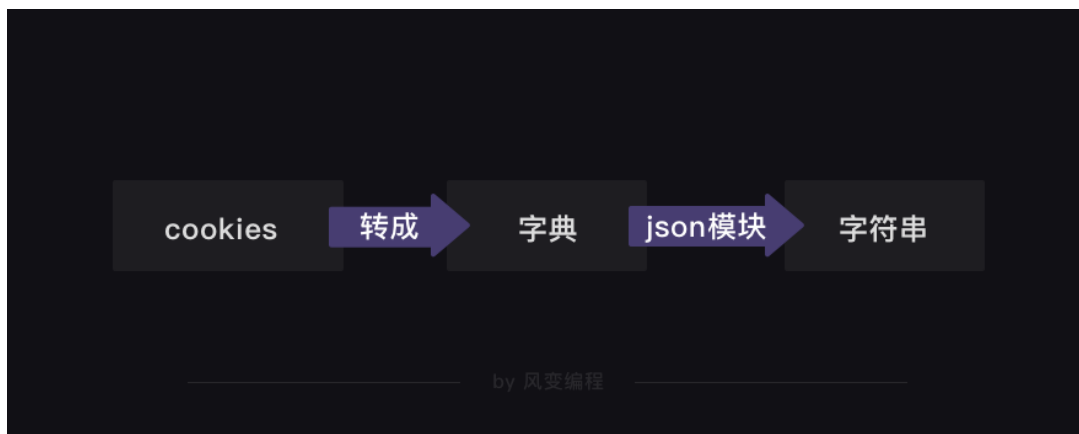
```

14     'testcookie': '1'
15 }
16 session.post(url, headers=headers, data=data)
17 #在创建的session下用post发起登录请求，放入参数：请求登录的网址、请求头和登录参数。
18
19 url_1 = 'https://wordpress-edu-3autumn.localprod.oc.forchange.cn/wp-
    comments-post.php'
20 #把我们想要评论的文章网址赋值给url_1。
21 data_1 = {
22     'comment': input('请输入你想要发表的评论：'),
23     'submit': '发表评论',
24     'comment_post_ID': '13',
25     'comment_parent': '0'
26 }
27 #把有关评论的参数封装成字典。
28 comment = session.post(url_1, headers=headers, data=data_1)
29 #在创建的session下用post发起评论请求，放入参数：文章网址，请求头和评论参数，并赋值
    给comment。
30 print(comment)
31 #打印comment

```

1-4、存储 cookies

json 模块能把字典转成字符串。我们可以先把 cookies 转成字典，然后再通过 json 模块转成字符串。这样，就能用 open 函数把 cookies 存储成 txt 文件。



••• cookies转化成字典的方法 •••

`requests.utils.dict_from_cookiejar (cj)`

从CookieJar返回键/值字典。

参数: **cj** - 从中提取cookie的CookieJar对象。

返回类型: 字典

••• json模块的使用方法 •••

JSON 函数

使用 JSON 函数需要导入 json 库: **import json**。

函数	描述
<code>json.dumps</code>	将 Python 对象编码成 JSON 字符串
<code>json.loads</code>	将已编码的 JSON 字符串解码为 Python 对象

by 风变编程

```
1 import requests,json
2 #引入requests和json模块。
3 session = requests.session()
4 url = ' https://wordpress-edu-3autumn.localprod.oc.forchange.cn/wp-
  login.php'
5 headers = {
6 'User-Agent':'Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36
  (KHTML, like Gecko) Chrome/71.0.3578.98 Safari/537.36'
7 }
8 data = {
9 'log': input('请输入你的账号:'),
10 'pwd': input('请输入你的密码:'),
11 'wp-submit': '登录',
12 'redirect_to': 'https://wordpress-edu-
  3autumn.localprod.oc.forchange.cn/wp-admin/',
13 'testcookie': '1'
14 }
15 session.post(url, headers=headers, data=data)
16
17 cookies_dict = requests.utils.dict_from_cookiejar(session.cookies)
18 #把cookies转化成字典。
19 print(cookies_dict)
20 #打印cookies_dict
21 cookies_str = json.dumps(cookies_dict)
22 #调用json模块的dumps函数,把cookies从字典再转成字符串。
23 print(cookies_str)
24 #打印cookies_str
25 f = open('cookies.txt', 'w')
```

```
26 #创建名为cookies.txt的文件，以写入模式写入内容。
27 f.write(cookies_str)
28 #把已经转成字符串的cookies写入文件。
29 f.close()
30 #关闭文件。
```

1-5、读取 cookies

读取 cookies 跟存储 cookies 时的顺序相反，要先把字符串转成字典，再把字典转成 cookies 本来的格式。



```
1 import requests,json
2 session = requests.session()
3 #创建会话。
4 headers = {
5     'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64)
6     AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.110 Safari/537.36'
7 }
8 #添加请求头，避免被反爬虫。
9 try:
10     #如果能读取到cookies文件，执行以下代码，跳过except的代码，不用登录就能发表评论。
11     cookies_txt = open('cookies.txt', 'r')
12     #以reader读取模式，打开名为cookies.txt的文件。
13     cookies_dict = json.loads(cookies_txt.read())
14     #调用json模块的loads函数，把字符串转成字典。
15     cookies = requests.utils.cookiejar_from_dict(cookies_dict)
16     #把转成字典的cookies再转成cookies本来的格式。
17     session.cookies = cookies
18     #获取cookies: 就是调用requests对象（session）的cookies属性。
19 except FileNotFoundError:
20     #如果读取不到cookies文件，程序报“FileNotFoundError”（找不到文件）的错，则执行以下
21     #代码，重新登录获取cookies，再评论。
22     url = 'https://wordpress-edu-3autumn.localprod.oc.forchange.cn/wp-
23     login.php'
24     #登录的网址。
25     data = {'log': input('请输入你的账号:'),
26            'pwd': input('请输入你的密码:'),
```

```

26         'wp-submit': '登录',
27         'redirect_to': 'https://wordpress-edu-
3autumn.localprod.oc.forchange.cn/wp-admin/',
28         'testcookie': '1'}
29     #登录的参数。
30     session.post(url, headers=headers, data=data)
31     #在会话下，用post发起登录请求。
32
33     cookies_dict = requests.utils.dict_from_cookiejar(session.cookies)
34     #把cookies转化成字典。
35     cookies_str = json.dumps(cookies_dict)
36     #调用json模块的dump函数，把cookies从字典再转成字符串。
37     f = open('cookies.txt', 'w')
38     #创建名为cookies.txt的文件，以写入模式写入内容
39     f.write(cookies_str)
40     #把已经转成字符串的cookies写入文件
41     f.close()
42     #关闭文件
43
44 url_1 = 'https://wordpress-edu-3autumn.localprod.oc.forchange.cn/wp-
comments-post.php'
45 #文章的网址。
46 data_1 = {
47     'comment': input('请输入你想评论的内容: '),
48     'submit': '发表评论',
49     'comment_post_ID': '13',
50     'comment_parent': '0'
51 }
52 #评论的参数。
53 comment = session.post(url_1, headers=headers, data=data_1)
54 #在创建的session下用post发起评论请求，放入参数：文章网址，请求头和评论参数，并赋值
    给comment。
55 print(comment.status_code)
56 #打印comment的状态码

```

如果cookies过期，就重新获取新的cookies：

```

1  import requests, json
2  session = requests.session()
3  headers = {
4      'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64)
    AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.110
    Safari/537.36'}
5
6  def cookies_read():
7      cookies_txt = open('cookies.txt', 'r')
8      cookies_dict = json.loads(cookies_txt.read())
9      cookies = requests.utils.cookiejar_from_dict(cookies_dict)
10     return (cookies)
11     # 以上4行代码，是cookies读取。
12

```

```
13 def sign_in():
14     url = ' https://wordpress-edu-3autumn.localprod.oc.forchange.cn/wp-
login.php'
15     data = {'log': input('请输入你的账号'),
16             'pwd': input('请输入你的密码'),
17             'wp-submit': '登录',
18             'redirect_to': 'https://wordpress-edu-
3autumn.localprod.oc.forchange.cn/wp-admin/',
19             'testcookie': '1'}
20     session.post(url, headers=headers, data=data)
21     cookies_dict = requests.utils.dict_from_cookiejar(session.cookies)
22     cookies_str = json.dumps(cookies_dict)
23     f = open('cookies.txt', 'w')
24     f.write(cookies_str)
25     f.close()
26     # 以上5行代码，是cookies存储。
27
28
29 def write_message():
30     url_2 = 'https://wordpress-edu-3autumn.localprod.oc.forchange.cn/wp-
comments-post.php'
31     data_2 = {
32         'comment': input('请输入你要发表的评论: '),
33         'submit': '发表评论',
34         'comment_post_ID': '13',
35         'comment_parent': '0'
36     }
37     return (session.post(url_2, headers=headers, data=data_2))
38     #以上9行代码，是发表评论。
39
40 try:
41     session.cookies = cookies_read()
42 except FileNotFoundError:
43     sign_in()
44     session.cookies = cookies_read()
45
46 num = write_message()
47 if num.status_code == 200:
48     print('成功啦! ')
49 else:
50     sign_in()
51     session.cookies = cookies_read()
52     num = write_message()
```