任务：facebook从2018年一月起，什么时候在什么地点发了什么内容

# Python

**Prophet** facebook趋势预测

**Octoparse** 爬虫软件

**Python爬虫抓站的技巧总结**

<http://www.pythonclub.org/python-network-application/observer-spider>

**HTTP代理获取+自动更换器**

<https://pan.baidu.com/s/1bnjfvpt>

**爬取知乎**

https://blog.csdn.net/chinwuforwork/article/details/77725959

**Python安装及其教程**

<https://www.liaoxuefeng.com/wiki/001374738125095c955c1e6d8bb493182103fac9270762a000/001375502200090e998439175cc4268b0ea703b3b4ed55e000>

**Python非官方包**

<https://www.lfd.uci.edu/~gohlke/pythonlibs/>

## Requests API

pip install requests

<http://docs.python-requests.org/zh_CN/latest/index.html>

**要安装的包**

**python -m pip install --upgrade pip 更新pip**

pip install requests

pip install zhihu-oauth

pip install xlwt 写入excel

pip install --upgrade setuptools 更新setuptools

1. **基本get请求**

>>> response = requests.get('http://www.baidu.com')

>>> response.status\_code

>>> response.encoding

1. **添加headers和查询参数**

>>> import requests

>>> kw = {'wd':'北京'}

>>> headers = {'User\_Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/54.0.2840.99 Safari/537.36' }

>>> response = requests.get('http://www.baidu.com/?',params = kw, headers = headers)

##返回unicode格式数据

>>> response.text

##返回字节流数据

>>> response.content

##查看完整url地址

>>> response.url

##查看响应头部字符编码

>>> response.encdoing

##查看响应码

>>> response.status\_code

1. **基本POST请求**

>>> import requests

>>> formdata = {"type" : "AUTO", "i":"i love python", "doctype":"json", "xmlVersion":"1.8","keyfrom":"fanyi.web","ue":"UTF-8","action":"FY\_BY\_ENTER","typoResult":"true"}

>>> url = "http://fanyi.youdao.com/translate?smartresult=dict&smartresult=rule&smartresult=ugc&sessionFrom=null"

>>> headers = {"User-Agent":"Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/51.0.2704.103 Safari/537.36"}

>>> response = requests.post(url, data = formdata, headers = headers)

>>> reponse.text

1. **代理**

根据协议类型，选择不同的代理IP，防止ip被封

>>>import requests

# 根据协议类型，选择不同的代理

>>>proxies = {"http": "http://xx.xx.xx.xx:1234"}

>>>response = requests.get("http://www.baidu.com", proxies = proxies)

>>>response.text

1. **拿到请求中包含的cookie**

>>> import requests

>>> response = requests.get("http://www.baidu.com/")

>>> cookiejar = response.cookies

>>> cookiedict = requests.utils.dict\_from\_cookiejar(cookiejar)

>>> print(cookiejar)

<RequestsCookieJar[<Cookie BDORZ=27315 for .baidu.com/>]>

>>> cookiejar

<RequestsCookieJar[Cookie(version=0, name='BDORZ', value='27315', port=None, port\_specified=False, domain='.baidu.com', domain\_specified=True, domain\_initial\_dot=True, path='/', path\_specified=True, secure=False, expires=1535613076, discard=False, comment=None, comment\_url=None, rest={}, rfc2109=False)]>

>>> print(cookiedict)

{'BDORZ': '27315'}

>>>

# Java

钱洋的达人课 gitbook.cn

1. 设置代理 <https://www.jianshu.com/p/d66129e00065>
2. 模拟登录及获得群组信息 <https://juejin.im/post/5aa50b0df265da2385327c15>

<https://github.com/xuguobiao/FacebookGrabber/>

1. Jsoup解析文档

<https://www.ibm.com/developerworks/cn/java/j-lo-jsouphtml/index.html>

<https://jsoup.org/apidocs/org/jsoup/select/Selector.html>

1. Java出现Exception in thread "main" javax.net.ssl.SSLProtocolException: handshake alert: unrecognized\_name 异常，是因为网站证书不被server信任，要手动添加网站证书

获取网站证书：chrome右键 -> 检查 -> Security -> 详细信息 -> 获取网站证书

Cmd用管理员打开：keytool -import -keystore cacerts -trustcacerts -alias "Indiaculture Server CA - G3" -file indiaculture.cer

Cacerts文件在：E:\sys\java\jre1.8.0\_91\lib\security>

-alias后的名字可以改，要唯一

-keytool的默认密码是changeit

5． <https://blog.csdn.net/EndisenChen/article/details/70766947>

将数据导出至excel

数据和excel交互操作

http://rensanning.iteye.com/blog/1538591

# Tips

无法从静态上下文中引用非静态变量

->比如说main里面不能调用自定义的function，要生成一个实例（new一个对象）

打印Map<String,String>

Map<String, String> map = getMyMap();

List<String> keys = new ArrayList<String>(map.keySet());

Collections.sort(keys, someComparator);

for (String key: keys) {

System.out.println(key + ": " + map.get(key));

}

<https://mbasic.facebook.com/profile.php?id=100004208416622&refid=46&__xts__%5B0%5D=12.%7B%22unit_id_click_type%22%3A%22graph_search_results_item_tapped%22%2C%22click_type%22%3A%22result%22%2C%22module_id%22%3A1%2C%22result_id%22%3A100004208416622%2C%22session_id%22%3A%22c5380d07080eadf2be3870da33fe19b8%22%2C%22module_role%22%3A%22ENTITY_USER%22%2C%22unit_id%22%3A%22browse_rl%3Ab92f38d4-7eac-dfea-b77f-99185521c296%22%2C%22browse_result_type%22%3A%22browse_type_user%22%2C%22unit_id_result_id%22%3A100004208416622%2C%22module_result_position%22%3A0%7D>

<https://mbasic.facebook.com/profile/timeline/stream/?cursor=tmln_strm%3A1509013766%3A-2248225520920774828%3A1&profile_id=100004208416622&replace_id=u_0_4&refid=17>

<https://mbasic.facebook.com/profile/timeline/stream/?cursor=tmln_strm%3A1475434819%3A935692024283068038%3A1&profile_id=100004208416622&replace_id=u_0_0>

# Selenium

<https://www.youtube.com/watch?v=5FUdrBq-WFo>