饿了么爬虫

使用 chrome 模拟饿了么移动端，用 python 模拟访问链接获得 json 数据

1 模拟访问商家列表

2 模拟访问商家数据

3 模拟访问评论数据和图片

1

```
In [1]: from requests import Session#
        import os                      #
        import requests                #cookies
        from time import sleep         #
        import json                    #jsonstr
```

2

```
In [2]: path = '/media/pi/6D726347A049AF0B/elem/'

In [3]: session = Session()
        #
        def crawele(i):
            global session             #sessioncookies
            headers = {
                "Origin": "https://h5.ele.me",
                "Referer": "https://h5.ele.me/shop/",
                "User-Agent":
                "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:61.0) Gecko/20100101 Firefox/61.0"
            }
            session.headers.update(headers)
            url1 = "https://crayfish.elemecdn.com/perf.js@json/sdk-config/WEB://h5.ele.me/shop/"
            request1 = session.get(url1)
            print(request1.status_code,end='|')
            #new_user_check
            headers = {
                'Accept':
                'application/json, text/plain, */*',
                'Accept-Encoding':
```

```
    'gzip, deflate, br',
    'Accept-Language':
    'zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2',
    'x-shard': 'loc=113.538542,22.348906'
}
cookiestr = "ubt_ssid=n3ckjy0vz6lsssq55jahsz4gs2p1zf6b_2019-01-21; _utrace=3cdf
isg=BGFhX7rTmXb1PDUBcVyIt259qC1S7tRinacsa0MM2WWjsK-oX8C1yS0fFriFCs4m04"
cookies = {i.split('=')[0]: i.split('=')[1] for i in cookiestr.split(';')}
cookies = requests.utils.cookiejar_from_dict(cookies)
session.cookies.update(cookies)
url2 = "https://h5.ele.me/restapi/eus/v2/new_user_check"
request2 = session.get(url2)
print(request2.status_code,end='|')
user_id = request2.json()['user_id']
#batch_shop
url3 = 'https://h5.ele.me/pizza/shopping/restaurants/{0}/batch_shop?user_id={1}&c
request3 = session.get(url3)
print(request3.status_code,end='|')
#reverse_geo_coding
url4 = 'https://h5.ele.me/restapi/bgs/poi/reverse_geo_coding?latitude=22.348906&lon
request4 = session.get(url4)
print(request4.status_code,end='|')
#service-worker
url5 = 'https://shadow.elemecdn.com/crayfish/h5.ele.me/service-worker'
request5 = session.get(url5)
print(request5.status_code,end='|')
#cart_client
cookiestr = "ubt_ssid=n3ckjy0vz6lsssq55jahsz4gs2p1zf6b_2019-01-21; _utrace=3cdf
EEG2erjb45NazTjIWAfwL5lEO23ej5q7WD795I"
cookies = {i.split('=')[0]: i.split('=')[1] for i in cookiestr.split(';')}
cookies = requests.utils.cookiejar_from_dict(cookies)
session.cookies.update(cookies)
payloadstr = '{"sub_channel":"","geohash":"webxpskpzf11","user_id":0,"add_on_type
payload = json.loads(payloadstr)
payload['user_id']=user_id
payload['restaurant_id']= i
url6 = 'https://h5.ele.me/restapi/booking/v1/cart_client'
request6 = session.post(url6, json=payload)
print(request6.status_code,end='|')
#batch comment
cookiestr = "ubt_ssid=n3ckjy0vz6lsssq55jahsz4gs2p1zf6b_2019-01-21; _utrace=3cdf
y6MSzZswYt1bit?Rw/VDSAZvmU="
cookies = {i.split('=')[0]: i.split('=')[1] for i in cookiestr.split(';')}
cookies = requests.utils.cookiejar_from_dict(cookies)
session.cookies.update(cookies)
headers = {
    'x-shard': 'shopid={0};loc=113.538542,22.348906'.format(i)
}
session.headers.update(headers)
url7 = 'https://h5.ele.me/pizza/ugc/restaurants/{0}/batch_comments?has_content=tr
request7 = session.get(url7)
```

```
    print(request7.status_code,end='|')
    result7 = request7.json()
    print()
    print(request7.json()['tags'][0])
    alllimit = result7['tags'][0]['count']
    #40
    totalresult=[]
    offlist = [40 for i in range(alllimit//40)]              #
    offlist.append(alllimit%40)                              #

    print('            ')
    for offn,offlimit in enumerate(offlist):
        sleep(5)
        url8 = 'https://h5.ele.me/restapi/ugc/v3/restaurants/{0}/ratings?has_content=tru
        request8 = session.get(url8)     #
        print(request8.status_code,end='|')
        for eachresult in request8.json():                   #
            totalresult.append(eachresult)
    return [request1.json(),request2.json(),request3.json(),request4.json(),request5.json(
```

In [4]:

```
#jsonimage  Id
def image_hash(js):
    lists=[]
    for i in js:
        try:
            for j in i['order_images']:
                if j['image_hash']!='':
                    lists.append(j['image_hash'])
        except:
            pass
    return lists
#json
def createjfile(listtry,path):
    print()
    print('')
    for i,j in enumerate(listtry):
        path1=path+'/{0}.json'.format(i)
        h = open(path1,'w')
        h.write(json.dumps(j))
        h.close()
        print('>>>{0}'.format(i),end='|')
#
def mkdir(path):
    folder = os.path.exists(path)
    if not folder:                                  #
        os.makedirs(path)                           #makedirs
        print (">>>{0}---OK".format(path))
    else:
        print (">>>{0}---exist".format(path))
#
def createshopfile(pathshop):
    for i in pathshop:
        mkdir(i)
```

3

```
In [5]: shopnum = 44
        shop_id_name={}
        offlist = [30 for i in range(shopnum//30)]
        headers = {
            'Accept':
            'application/json, text/plain, */*',
            'Accept-Encoding':
            'gzip, deflate, br',
            'Accept-Language':
            'zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2',
            'x-shard': 'loc=113.538542,22.348906'
        }
        cookiestr = "ubt_ssid=n3ckjy0vz6lsssq55jahsz4gs2p1zf6b_2019-01-21; _utrace=3cdf7f34
        RhiR415HLL1wlY5iqP6UPMWlrzGzz18gG+S6kBTi5V; isg=BHNzl4YWK189V-cLV_rQ7eDzAn5
        cookies = {i.split("=")[0]: i.split("=")[1] for i in cookiestr.split("; ")}
        cookies = requests.utils.cookiejar_from_dict(cookies)
        session.cookies.update(cookies)
        for offn,offlimit in enumerate(offlist):
            url = 'https://h5.ele.me/restapi/shopping/v3/restaurants?latitude=22.348906&longitud
            resquests = session.get(url)
            shop_id_name.update({i['restaurant']['id']:i['restaurant']['name'] for i in resquests.jso

In [6]: pathshop = [path+i.strip() for i in list(shop_id_name.values())]
        #
        createshopfile(pathshop)

>>>/media/pi/6D726347A049AF0B/elem/Hey Juice()---exist
>>>/media/pi/6D726347A049AF0B/elem/()---exist
>>>/media/pi/6D726347A049AF0B/elem/1---exist
>>>/media/pi/6D726347A049AF0B/elem/COMEBUY()---exist
>>>/media/pi/6D726347A049AF0B/elem/ů---exist
>>>/media/pi/6D726347A049AF0B/elem/()---exist
>>>/media/pi/6D726347A049AF0B/elem/---exist
>>>/media/pi/6D726347A049AF0B/elem/---exist
>>>/media/pi/6D726347A049AF0B/elem/Homie----exist
>>>/media/pi/6D726347A049AF0B/elem/()---exist
>>>/media/pi/6D726347A049AF0B/elem/---exist
>>>/media/pi/6D726347A049AF0B/elem/COMEBUY()---exist
>>>/media/pi/6D726347A049AF0B/elem/---exist
>>>/media/pi/6D726347A049AF0B/elem/---exist
>>>/media/pi/6D726347A049AF0B/elem/(14)---exist
```

```
>>>/media/pi/6D726347A049AF0B/elem/(15)---exist
>>>/media/pi/6D726347A049AF0B/elem/---exist
>>>/media/pi/6D726347A049AF0B/elem/---exist
>>>/media/pi/6D726347A049AF0B/elem/()---exist
>>>/media/pi/6D726347A049AF0B/elem/ůůů---exist
>>>/media/pi/6D726347A049AF0B/elem/---exist
>>>/media/pi/6D726347A049AF0B/elem/---exist
>>>/media/pi/6D726347A049AF0B/elem/---exist
>>>/media/pi/6D726347A049AF0B/elem/()---exist
>>>/media/pi/6D726347A049AF0B/elem/---exist
>>>/media/pi/6D726347A049AF0B/elem/---exist
>>>/media/pi/6D726347A049AF0B/elem/---exist
>>>/media/pi/6D726347A049AF0B/elem/()---exist
>>>/media/pi/6D726347A049AF0B/elem/---exist
>>>/media/pi/6D726347A049AF0B/elem/---exist
```

```
In [11]: shop_id_name = {'E17485065127105825388': '','E8571605648348391946': ''}
```

```
In [12]: from hometts import tran
         imagelist = []
         session = Session()
         for id_name in shop_id_name.items():
             try:
                 print(id_name)
                 tran(id_name[1]+'')
                 resultlist = crawele(id_name[0])
                 createjfile(resultlist,path+id_name[1])
                 imagess = image_hash(resultlist[-1])
                 imagelist+=imagess
                 print('2')
                 tran('2')
                 sleep(120)
             except:
                 tran(id_name[1]+'')
         print('='*30+'OK')
```

```
('E17485065127105825388', '')
200|200|200|200|200|200|200|
{'count': 170, 'name': '', 'unsatisfied': False}

200|200|200|200|200|

>>>0|>>>1|>>>2|>>>3|>>>4|>>>5|>>>6|>>>7|2
('E8571605648348391946', '')
200|200|200|200|200|200|200|
{'count': 205, 'name': '', 'unsatisfied': False}
```

```
200|200|200|200|200|200|

>>>0|>>>1|>>>2|>>>3|>>>4|>>>5|>>>6|>>>7|2
==================================OK
```

```
In [5]:  #
         def getshopdata(path, name):
             data = []
             datalist = os.listdir(path + '/' + name)
             h = open(path + '/' + name + '/' + '7.json', 'rb')
             datai = h.read()
             h.close()
             data.append(json.loads(datai))
             return json.loads(datai)


         #
         def getattr(path, shopname):
             trydata = getshopdata(path, shopname)
             attrsdata = []
             for i in trydata:
                 try:
                     attrsdata.append(i['order_images'][0]['image_hash'])
                 except:
                     pass
             return attrsdata


         #
         def getallattr(path, shoplist):
             allattr = []
             for shopname in shoplist:
                 try:
                     attrdata = getattr(path, shopname)
                     for item in attrdata:
                         allattr.append(item)
                 except:
                     print('this file is fail:{0}'.format(shopname))
             return allattr
```

```
In [6]: imagelist = getallattr(path,os.listdir(path))
```

```
this file is fail:Image
```

```
In [7]: len(imagelist)
```

```
Out[7]: 6229
```

```
In [8]: from hometts import tran
        pathims = path + 'Image'
        mkdir(pathims)
        for id, images in enumerate(imagelist):
            result = session.get(
                'https://fuss10.elemecdn.com/{0}/{1}/{2}.jpeg?imageMogr2/thumbnail/60x60/f
                .format(images[0], images[1:3], images[3:]))
            print(result.status_code, end='|')
            pathim = pathims + '/{0}.png'.format(images)
            h = open(pathim, 'wb')
            h.write(result.content)
            if id%100 == 0 :
                tran('{0}{1}'.format(id,(len(imagelist)-id)/len(imagelist)*100))
                tran('3')
                sleep(180)
                tran('')
```

```
>>>/media/pi/6D726347A049AF0B/elem/Image---exist
200|

/root/anaconda3/lib/python3.7/site-packages/urllib3/connectionpool.py:857: InsecureRequestWarni
  InsecureRequestWarning)


200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|200|
```

```
In [ ]:
```