

第一步:右键浏览器空白处,选择检查,点击网络,ctrl+r刷新,点击名称列表中的任意一个,然后再右侧的标头中下拉,找到User-Agent复制后面的内容,
比如是Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/127.0.0.0 Safari/537.36
再找到referrer复制后面的内容,比如是
`https://www.3gbizhi.com/wallMV/index.html`

第二步:构造一个http(s)请求头

```
headers={  
    'User-Agent':'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36  
    (KHTML, like Gecko) Chrome/127.0.0.0 Safari/537.36',  
    'Referer':'https://www.3gbizhi.com/wallMV/index.html'  
}
```

注意事先需导入两个库

```
import requests  
import parsel  
4.url='http://www.3gbizhi.com/wallMV/index.html'  
r=requests.get(url=url,headers=headers)  
r表示服务器的应答,requests.get表示采用get请求(http请求的一种),headers是刚才的字典,url在单引号里,参数在左,变量在右.  
r.status_code输出应该是200,不然后续无法进行.接着print(r.text)  
输出html文本内容一共有1359行  
右键view查看.  
对于html,现在应该用xpath表达式筛选出里面的img标签.
```

第六步:在浏览器中找到xpath右键检查,按下ctrl+shift+c

随便点一个图片检查找到该元素右键复制,复制xpath,
比如是/html/body/main/div[4]/ul/li[11]/a/img,
去掉/html/body以及[]里的数字,就像这样//main/div/ul/li/a/img,
注意main前面是//

第七步:我们要使r.text有一个向量xpath,

首先要把它转化为一个parsel对象,例如parsel.Selector(r.text)

然后

```
print(pasel.Selector(r.text).xpath('//main/div/ul/li/a/img').getall())  
可以看到有一大堆输出,其中有一堆属性是lay-src,=号后的内容直接指向了图片所在  
的url地址(相当于点击"在新标签页中打开图片")
```

例如https://pic.3gbizhi.com/uploads/xxx.jpg

我们要获取这个属性,只需要

```
print(pasel.Selector(r.text).xpath('//main/div/ul/li/a/img/@lay-src').getall())  
返回一个图片列表.
```

第八步:for循环遍历这个列表

```
datalist=parel.Selector(r.text).xpath('//main/div/ul/li/a/img/@lay-src').getall(
)
```

```
for data in datalist:
```

```
    print(data)
```

这将把他们一行一行的输出,如果想把他们保存,只需要

```
number=1
```

```
for data in datalist:
```

```
    with open (f'www3gbizhi/{number}.jpg','wb') as p:
```

```
    #wb模式针对二进制数据的写入 f表示格式化,不然要加双斜杠
```

```
    #www3gbizh是路径,as表示把它重命名为p
```

```
        p.write(requests.get(url=data,headers=headers).content)
```

```
    #.content表示二进制格式的数据write表示写入
```

```
        number+=1
```

```
    #让number从一开始,不断+1
```

```
    #前提,同目录下一定要有一个空文件夹www3gbizhi
```

源码:

```
import requests
```

```
import parsel
```

```
number=1
```

```
for page in range(2,35): print('====={}====='.format(page))
```

```
    #{}是一个占位符,范围从1到34
```

```
    base_url="https://www.3gbizhi.com/wallMV/index_{}.html".format(page)#同理
```

```
    headers={'User-agent' : 'Mozilla/5.0 (Windows NT 10.0; WOW64)
```

```
    AppleWebKit/537.1 (KHTML ,like Gecko) Safari/537.1'}
```

```
    response=requests.get(url=base_url) html_data=response.text
```

```
    parse=parsel.Selector(html_data)
```

```
    data_list=parse.xpath('/html/body/main/div/ul/li/a/img/@lay-src').getall()
```

```
    for data in data_list:
```

```
        imgdata=requests.get(url=data,headers=headers).content
```

```
            with open (f'www3g/{number}.jpg',mode='wb') as f:
```

```
                f.write(imgdata)
```

```
            number+=1
```

总结,parsel只是用了一种正则表达式的方法,beauifulsoup甚至linux的perl命令也可以筛选(三个月内发表)

本文在三个月前被我写下,现在发现它还可以运行,高兴极了.