*本文是提取雪球数据token的过程记录*

**1.提取过程**

首先我们依然打开chrome的开发者工具,点击network的标签,选择XHR。如图:

从图中可以轻易的获取到雪球数据的api地址和请求参数等信息。下面将API复制到Postman中尝试获取数据。

[*https://stock.xueqiu.com/v5/stock/realtime/quotec.json?symbol=SH000001,SZ399001,SZ399006&\_=1541640828575*](https://stock.xueqiu.com/v5/stock/realtime/quotec.json?symbol=SH000001,SZ399001,SZ399006&_=1541640828575)

可以看到，成功获得了报价数据。再试试其他API。

[*https://stock.xueqiu.com/v5/stock/history/trade.json?symbol=SH510050&count=10*](https://stock.xueqiu.com/v5/stock/history/trade.json?symbol=SH510050&count=10)

这次报了未登录的错误，说明雪球对其中一些API进行的权限验证，一些则没有。我们可以尝试从合法的请求信息中寻找验证口令。

经观察，token一定是保存在了请求的Cookie中了。将所有Cookie列出来，去除明显不需要的。只剩下如下几个。从多次请求数据中发现有些值并不影响成功请求数据。

aliyungf\_tc=AQAAAJglyDOjpQwAVYYKcPUhPe30XE/a;

xq\_a\_token=a8d434ddd975f5752965fa782596bd0b5b008376;

xq\_a\_token.sig=ke78qTMMk1J4blZPe-jY53Uy9Ec;

xq\_r\_token=437547d929e3cc54630bfd58136879694e1ae4a9;

xq\_r\_token.sig=iYuNwCitZuVpyfkOu6\_LLtaQn6E;

u=781511537516883;

device\_id=a4d5e9838df630d4ed110e85576d0482;

Hm\_lvt\_1db88642e346389874251b5a1eded6e3=1511537517;

Hm\_lpvt\_1db88642e346389874251b5a1eded6e3=1511537543

继续用排除法，在postman中加入信息，不断请求，看结果，找出必要的token。(基本锁定在带token字符的那几个里）

最后得出结果,只有xq\_a\_token是必须的。

xq\_a\_token=a8d434ddd975f5752965fa782596bd0b5b008376;

**2.提取雪球数据代码**

我将自己查到的api进行了整理，写成了一个python包，方便使用。

安装pysnowball函数包

pip install pysnowball

python简单示例，更多api请查文档：<https://github.com/uname-yang/pysnowball>

import pysnowball as ball

ball.set\_token('xq\_a\_token=a8d434ddd975f5752965fa782596bd0b5b008376;')

ball.earningforecast('SZ002027')

结果：

{

"list": [

{

"eps": 4.78,

"roe": null,

"pb": null,

"pe": 8.4,

"forecast\_year": "2018"

},

{

"eps": 5.49,

"roe": null,

"pb": null,

"pe": 7.32,

"forecast\_year": "2019"

},

{

"eps": 6.12,

"roe": null,

"pb": null,

"pe": 6.56,

"forecast\_year": "2020"

}

]

}

**3.结论**

雪球数据的验证信息在请求头的Cookie中，将xq\_a\_token=xxx;部分加入请求即可通过登录验证。