Tushare:

https://github.com/waditu/Tushare

<http://tushare.waditu.com/index.html#id5>

重新配置，Anaconda（有了Python），然后安装IDE（Pycharm等），第二步，为PyCharm配置anaconda环境。进入PyCharm，依次点击File->Default Setting->Project Interpreter，点击齿轮选择Add Local，点击第三个选项卡system Interpreter，将路径选为anaconda安装路径下的python.exe，确定后成功配置anaconda环境。之后，在该环境下创建的工程就可以使用anaconda中已有的库了。

1.1.1

安装Anaconda包含python所有依赖包；

1.1.2

在Anaconda中安装pip install tushare(升级为pip install tushare --upgrade)

1.1.3

使用

>>> import tushare as ts

>>> ts.get\_hist\_data('601608') #一次性获取全部日数据

2.如何获得代码

Import tushare

Import pandas

Import datetime

# Step 1.Get tickers online

tickersRawData = tushare.get\_stock\_basics()

tickers = tickersRawData.index.tolist()

#Step 2. Save the ticker list to a local file

dateToday = datetime.datetime.today().strftime(‘%Y%m%d’)

file = ‘../02.data/00.Ticker’+dateToday+’.csv’

tickersRawData.to\_csv(file)

print (‘Tickers saved.’)

3.1.1

a[100] = {630, 2131, 2505, 600157, 600139, 601326, 300043, 600020, 2610, 300255, 603011, 300291, 600293, 633, 601598, 2477, 600552, 2519};

测试，把连续三天上涨的放在一个函数里。

把这些票比较，在写一个是否历史最低价的函数；

['300539', '300539', '300539', '600536', '600536', '300255', '300255', '300255', '002368', '002368', '002234', '600136', '600136', '600136', '000818', '000813', '000813', '000813', '002609', '002609', '603730', '603730', '603730', '002359', '002359', '300481', '300481', '300481', '300125', '002385', '002385', '002385'] 381

Traceback (most recent call last):

File "D:/00001000/Python/GetEveryday.py", line 62, in <module>

op = history['open']

TypeError: 'NoneType' object is not subscriptable

Process finished with exit code 1

注意当这一天没有数据时，>>> print(ts.get\_hist\_data('600145')) #结果为None，会报错

解决办法一：判断是否为空：

if history is None： i+=1 continue

python中利用第一种：if X is None;第二种：if not X；当X为None, False, 空字符串"", 0, 空列表[], 空字典{}, 空元组()这些时，not X为真，即无法分辨出他们之间的不同。

解决办法二：用try except语句

try： 条件except：i+=1 continue #可以达到同样的效果

1.1.1

当j<3时，表示三天内的，当问有三个涨幅大于1%的，为何只有1-382号是符合的，而后面没有符合，这column肯定是得到的数据就是排序过的，按涨幅，

因为i= 381 ，即600145是什么也没有None，应该有错误检测机制