

需求：假如一个人从2010年1月1日开始，每月第一个交易日买入1手股票，每年最后一个交易日卖出所有股票，到今天为止，那么他的收益如何？

股票数据为stock.csv

In [1]:

```
from pandas import DataFrame
import pandas as pd
```

In [15]:

```
# 在读取数据的时候把字符串时间转为date类型，并把列索引/date转为行索引
df = pd.read_csv("../data/stock/stock.csv", index_col="date", parse_dates=["date"])
```

In [16]:

```
df.head()
```

Out[16]:

	Unnamed: 0	open	close	high	low	volume	code
date							
2001-08-27	0	5.392	5.554	5.902	5.132	406318.00	600519
2001-08-28	1	5.467	5.759	5.781	5.407	129647.79	600519
2001-08-29	2	5.777	5.684	5.781	5.640	53252.75	600519
2001-08-30	3	5.668	5.796	5.860	5.624	48013.06	600519
2001-08-31	4	5.804	5.782	5.877	5.749	23231.48	600519

In [17]:

```
df.drop("Unnamed: 0", axis=1, inplace=True)
```

In [18]:

```
df.head()
```

Out[18]:

	open	close	high	low	volume	code
date						
2001-08-27	5.392	5.554	5.902	5.132	406318.00	600519
2001-08-28	5.467	5.759	5.781	5.407	129647.79	600519
2001-08-29	5.777	5.684	5.781	5.640	53252.75	600519
2001-08-30	5.668	5.796	5.860	5.624	48013.06	600519
2001-08-31	5.804	5.782	5.877	5.749	23231.48	600519

In [20]:

```
# 最后一个开盘价格
price_last = df['open'][-1]
# 选2010到2019的数据
df = df['2010':'2019']
#Pandas提供了resample函数用便捷的方式对时间序列进行重采样，根据时间粒度的变大或者变小分为降采样和
# 因为是每月第一个交易日买入1手股票，所以根据1M进行采样，取第一个first数据，完整的一年会采样得到12个数据
df_monthly = df.resample("1M").first()
# 每年最后一个交易日卖出所有股票，所以根据1Y进行采样，取最后一个last数据。这里去除最后一年，因为2019年数据不完整
df_yearly = df.resample("1Y").last()[:-1] #去除最后一年
# cost_money 初始置0
cost_money = 0
hold = 0 #每年持有的股票
for year in range(2010, 2020):
    # 这里的sum相当于把一年12个月的第一天的数据加起来，至于*100是因为1手股票等于100股
    cost_money -= df_monthly.loc[str(year)]['open'].sum()*100
    # 持有股票数量
    hold += len(df_monthly[str(year)]['open']) * 100
    if year != 2019:
        # 非2019年，会在每年最后一个交易日卖出股票来挣钱
        cost_money += df_yearly[str(year)]['open'][0] * hold
        # 卖完股票持有股票数置0
        hold = 0 #每年持有的股票
# 2019年手上还有股票因为没有到年底，没有出售，此时也应加上这些股票的价值
cost_money += hold * price_last

print(cost_money)
```

310250.69999999984

In []: