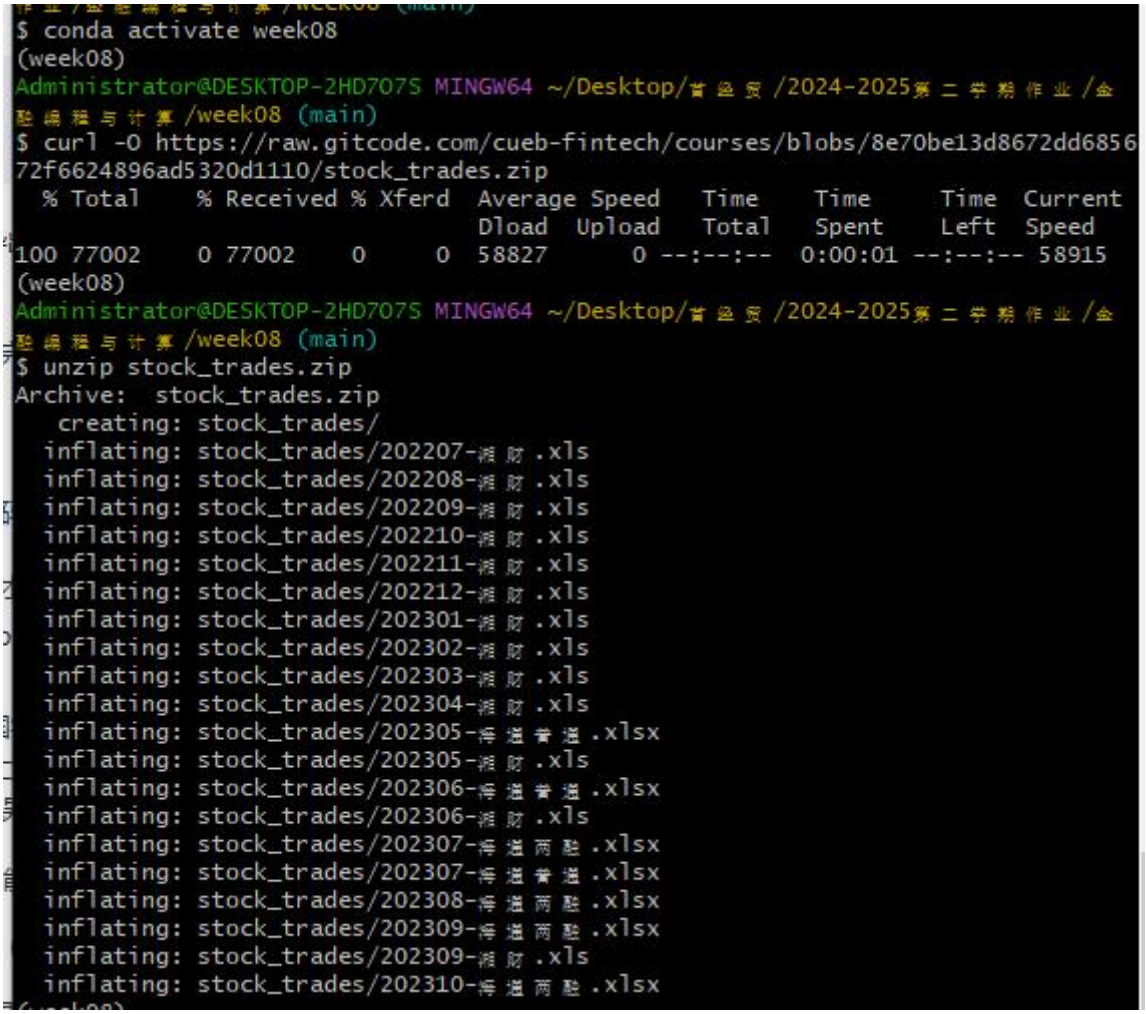
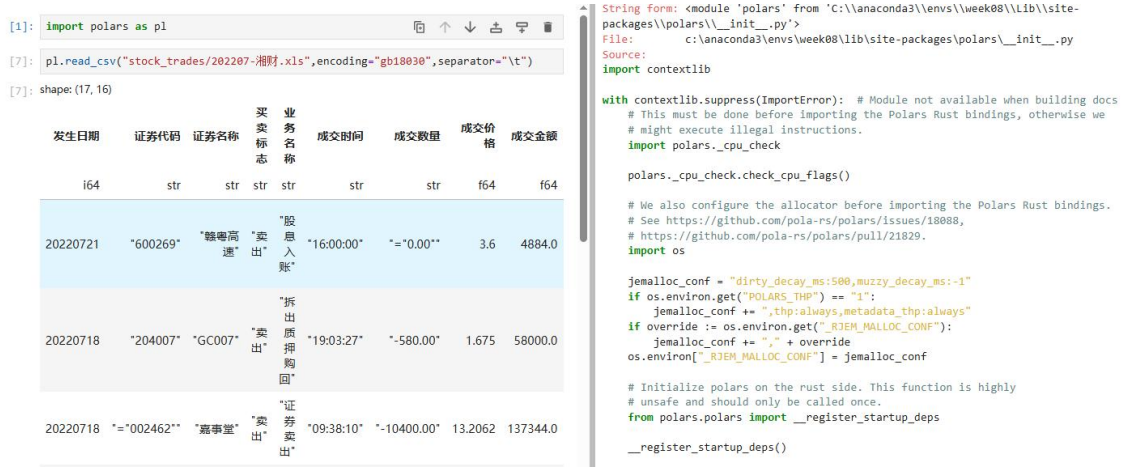
Part1-3

(1) 配置环境和下载数据



(2) 读取数据，反复调试参数，和解码格式



```
[1]: import polars as pl
[7]: pl.read_csv("stock_trades/202207-湘财.xls",encoding="gb18030",separator="\t")
[7]: shape: (17, 16)
```

发生日期	证券代码	证券名称	买卖标志	业务名称	成交时间	成交数量	成交价格	成交金额
i64	str	str	str	str	str	str	f64	f64
20220721	"600269"	"赣粤高速"	"卖出"	"股息入账"	"16:00:00"	"="0.00""	3.6	4884.0
20220718	"204007"	"GC007"	"卖出"	"拆出质押购回"	"19:03:27"	"-580.00"	1.675	58000.0
20220718	"="002462""	"嘉事堂"	"卖出"	"证券卖出"	"09:38:10"	"-10400.00"	13.2062	137344.0

```
String form: <module 'polars' from 'C:\\anaconda3\\envs\\week08\\Lib\\site-packages\\polars\\__init__.py'>
File:        c:\anaconda3\envs\week08\lib\site-packages\polars\__init__.py
Source:
import contextlib

with contextlib.suppress(ImportError):  # Module not available when building docs
    # This must be done before importing the Polars Rust bindings, otherwise we
    # might execute illegal instructions.
    import polars._cpu_check

    polars._cpu_check.check_cpu_flags()

    # We also configure the allocator before importing the Polars Rust bindings.
    # See https://github.com/pola-rs/polars/issues/18088,
    # https://github.com/pola-rs/polars/pull/21829.
    import os

    jemalloc_conf = "dirty_decay_ms:500,muzzy_decay_ms:-1"
    if os.environ.get("POLARS_THP") == "1":
        jemalloc_conf += ",thp:always,metadata_thp:always"
    if override := os.environ.get("_RJEM_MALLOC_CONF"):
        jemalloc_conf += "," + override
    os.environ["_RJEM_MALLOC_CONF"] = jemalloc_conf

    # Initialize polars on the rust side. This function is highly
    # unsafe and should only be called once.
    from polars.polars import __register_startup_deps

    __register_startup_deps()
```

(3) 初步数据格式转换

[96]:
```
df= pl.read_csv("stock_trades/202207-湘财.xls",encoding="gb18030",separator="\t",infer_schema=False)
df=df.with_columns(
    pl.col("发生日期").str.to_date("%Y%m%d"),
    pl.col("证券代码").str.strip_prefix("=").str.strip_chars('"'),
)
df[: ,"证券代码"].unique().to_list()
```

[96]:
```
['000900',
 '600269',
 '600648',
 '600408',
 '600894',
 '204007',
 '601077',
 '002462',
 '600015',
 '601992']
```

:
```
df = pl.read_csv(
    "stock_trades/202207-湘财.xls",
    encoding="gb18030",
    separator="\t",
    infer_schema=False,
)
df = df.with_columns(
    pl.selectors.all().str.strip_prefix("=").str.strip_chars('"'),
)
df = df.with_columns(
    pl.col("发生日期").str.to_date("%Y%m%d"),
    pl.col("证券代码").str.strip_prefix("=").str.strip_chars('"'),
    pl.col("成交时间").str.to_time(),
    pl.col(
        "成交数量",
        "成交价格",
        "成交金额",
        "发生金额",
        "手续费",
        "印花税",
        "过户费",
        "其他费",
    ).cast(pl.Float64),
)
df = df.filter(
    pl.col("业务名称").is_in(["证券卖出", "证券买入"]),
)
df
```

: 16)

证券代码	证券名称	买卖标志	业务名称	成交时间	成交数量	成交价格	成交金额	发生金额	手续费	印花税	过户费	其他费	备注	币种
str	str	str	str	time	f64	f64	f64	f64	f64	f64	f64	f64	str	str

(4) 合并湘财数据

```
d1 = pl.concat(df)
```

```
d1.with_columns(
    券商=pl.lit("湘财"),
)
```

shape: (257, 17)

发生日期	证券代码	证券名称	买卖标志	业务名称	成交时间	成交数量	成交价格	成交金额	发生金额	手续费	印花税	过户费	其他费	备注	币种	券商
date	str	str	str	str	time	f64	f64	f64	f64	f64	f64	f64	f64	str	str	str
2022-07-18	"002462"	"嘉事堂"	"卖出"	"证券卖出"	09:38:10	-10400.0	13.2062	137344.0	137184.67	21.98	137.35	1.38	0.0	"证券卖出"	"人民币"	"湘财"
2022-07-18	"600408"	"安泰集团"	"买入"	"证券买入"	09:44:52	47000.0	3.19	149930.0	-149955.5	23.99	0.0	1.51	0.0	"证券买入"	"人民币"	"湘财"
2022-07-18	"600648"	"外高桥"	"买入"	"证券买入"	09:44:31	11900.0	12.6066	150019.0	-150044.49	24.0	0.0	1.49	0.0	"证券买入"	"人民币"	"湘财"
2022-07-18	"600269"	"赣粤高速"	"买入"	"证券买入"	09:43:38	40700.0	3.69	150183.0	-150208.53	24.03	0.0	1.5	0.0	"证券买入"	"人民币"	"湘财"
2022-07-18	"600015"	"华夏银行"	"买入"	"证券买入"	09:42:51	30000.0	5.07	152100.0	-152125.86	24.34	0.0	1.52	0.0	"证券买入"	"人民币"	"湘财"
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
2023-06-19	"603967"	"中创物流"	"卖出"	"证券卖出"	10:18:46	-5000.0	9.13	45650.0	45596.59	7.3	45.65	0.46	0.0	"证券卖出"	"人民币"	"湘财"
2023-06-12	"300641"	"正丹股份"	"卖出"	"证券卖出"	13:22:32	-9600.0	5.05	48480.0	48423.76	7.76	48.48	0.48	0.0	"证券卖出"	"人民币"	"湘财"

(5) 统一名称后合并数据量为 363 个

[220]:
```
d2 = d2.select(
    券商=pl.col("券商"),
    交易日期=pl.col("成交日期"),
    交易时间=pl.col("成交时间"),
    证券代码=pl.col("证券代码"),
    买卖标志=pl.col("操作").replace({"卖": "卖出", "买": "买入"}),
    成交价格=pl.col("成交价格"),
    成交数量=pl.col("成交数量").abs(),
    成交金额=pl.col("成交金额"),
    手续费=pl.col("手续费"),
    印花税=pl.col("印花税"),
    过户费=pl.col("过户费"),
    其他费=pl.col("其他费"),
    发送金额=pl.col("发生金额"),
)
```

[221]:
```
d3 = d3.select(
    券商=pl.col("券商"),
    交易日期=pl.col("成交日期"),
    交易时间=pl.col("成交时间"),
    证券代码=pl.col("证券代码"),
    买卖标志=pl.col("操作").replace({"卖": "卖出", "买": "买入"}),
    成交价格=pl.col("成交价格"),
    成交数量=pl.col("成交数量").abs(),
    成交金额=pl.col("成交金额"),
    手续费=pl.col("手续费"),
    印花税=pl.col("印花税"),
    过户费=pl.col("过户费"),
    其他费=pl.col("其他费"),
    发送金额=pl.col("发生金额"),
)
```

[225]:
```
pl.concat([d1, d2, d3])
```

[225]: shape: (363, 13)

(6) 保存数据为三种格式

```
df=pl.concat([d1, d2, d3])
```

```
df.write_parquet("stock_trades.qarquet")
```

```
df.write_csv("stock_trades.csv")
```

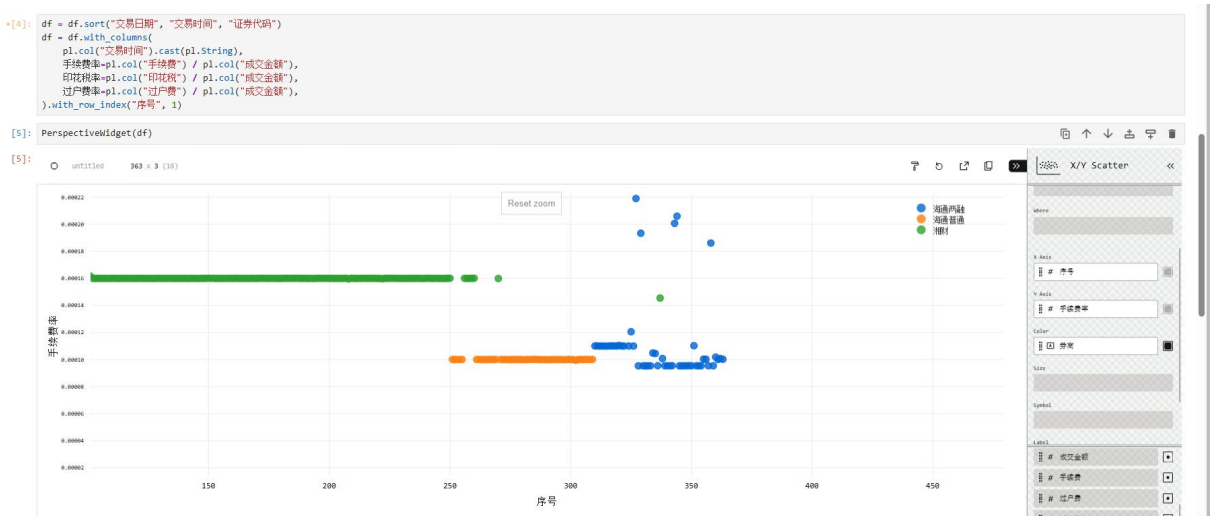
```
df.write_excel("stock_trades.xlsx")
```

Part4

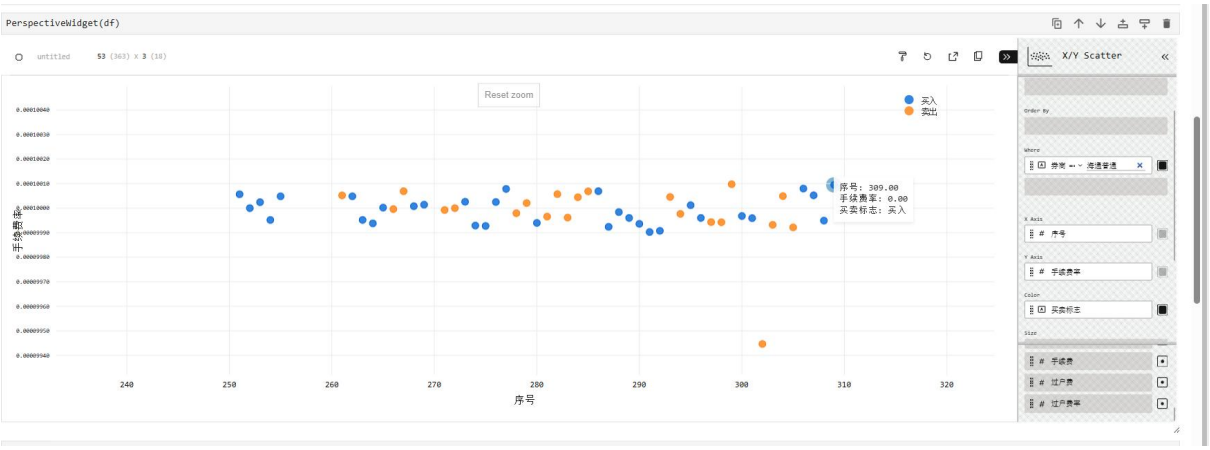
计算相关费率并进行可视化

```
df = df.sort("交易日期", "交易时间", "证券代码")
df = df.with_columns(
    pl.col("交易时间").cast(pl.String),
    手续费率=pl.col("手续费") / pl.col("成交金额"),
    印花税率=pl.col("印花税") / pl.col("成交金额"),
    过户费率=pl.col("过户费") / pl.col("成交金额"),
).with_row_index("序号", 1)
```

```
PerspectiveWidget(df)
```



观察不同券商的费率，比较而来：海通普通的手续费率在万分之 1 上下浮动，湘财稳定在万分之 1.6，较高。



Part5

分组汇总，对结余数量为负的证券进行剔除

```
d1 = df.join(
    df.group_by("证券代码", "证券名称")
    .agg(
        结余数量=(
            pl.when(pl.col("买卖标志") == "卖出")
            .then(-pl.col("成交数量"))
            .when(pl.col("买卖标志") == "买入")
            .then(pl.col("成交数量"))
            .sum()
        ) # 确保条件链的括号闭合
    )
    .filter(pl.col("结余数量") < 0),
    on="证券代码",  # 将 join() 的参数对齐到 join() 方法内
    how="anti",
)
```

```
d1
```

shape: (358, 18)

序号	券商	交易日期	交易时间	证券代码	证券名称	买卖标志	成交价格	成交数量	成交金额	手续费	印花税	过户费	其他费	发生金额	手续费率	印花税率	过户费率
u32	str	date	str	str	str	str	f64	f64	f64	f64	f64	f64	f64	f64	f64	f64	f64
1	"湘财"	2022-07-11	"09:33:37"	"000900"	"现代投资"	"买入"	4.05	34400.0	139320.0	22.29	0.0	1.39	0.0	-139342.29	0.00016	0.0	0.00001
2	"湘财"	2022-07-11	"09:34:24"	"601077"	"渝农商行"	"买入"	3.65	38300.0	139795.0	22.37	0.0	1.38	0.0	-139818.75	0.00016	0.0	0.00001
3	"湘财"	2022-07-11	"09:36:30"	"600894"	"广日股份"	"买入"	6.54	21400.0	139956.0	22.39	0.0	1.41	0.0	-139979.8	0.00016	0.0	0.00001
4	"湘财"	2022-07-11	"09:37:25"	"601992"	"金隅集团"	"买入"	2.59	54000.0	139860.0	22.38	0.0	1.42	0.0	-139883.8	0.00016	0.0	0.00001
5	"湘财"	2022-07-11	"09:38:16"	"002462"	"嘉事堂"	"买入"	13.51	10400.0	140504.0	22.48	0.0	1.41	0.0	-140526.48	0.00016	0.0	0.00001
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…
359	"海通两融"	2023-10-31	"09:31:53"	"002956"	"西麦食品"	"卖出"	14.13	5000.0	70650.0	6.74	35.35	0.0	0.0	70607.91	0.000095	0.0005	0.0
360	"海通两融"	2023-10-31	"09:39:57"	"603214"	"爱婴室"	"买入"	15.84	3100.0	49104.0	5.0	0.0	0.51	0.0	-49109.51	0.000102	0.0	0.00001
361	"海通两融"	2023-10-31	"09:40:55"	"300132"	"青松股份"	"买入"	5.21	9600.0	50016.0	5.0	0.0	0.0	0.0	-50021.0	0.0001	0.0	0.0

观察股票的持仓过程

用 tushare 导入股票日行情数据

```
[39]: import tushare as ts
```

```
[40]: pro = ts.pro_api()
```

```
[41]: hq = pro.daily(
          ts_code="002426.SZ",
          start_date=format(start_date, "%Y%m%d"),
          end_date=format(end_date, "%Y%m%d"),
      )
      hq=pl.from_pandas(hq)
      hq
```

[41]: shape: (318, 11)

ts_code	trade_date	open	high	low	close	pre_close	change	pct_chg	vol
str	str	f64	f64	f64	f64	f64	f64	f64	f64
"002426.SZ"	"20231031"	2.48	2.68	2.47	2.56	2.51	0.05	1.992	3.1488e6
"002426.SZ"	"20231030"	2.34	2.6	2.34	2.51	2.36	0.15	6.3559	3.4470e6
"002426.SZ"	"20231027"	2.38	2.38	2.29	2.36	2.42	-0.06	-2.4793	1.1816e6
"002426.SZ"	"20231026"	2.44	2.45	2.37	2.42	2.45	-0.03	-1.2245	852886.0
"002426.SZ"	"20231025"	2.4	2.48	2.39	2.45	2.4	0.05	2.0833	960007.15
…	…	…	…	…	…	…	…	…	…
"002426.SZ"	"20220715"	2.33	2.34	2.26	2.27	2.38	-0.11	-4.6218	677739.03
"002426.SZ"	"20220714"	2.38	2.43	2.35	2.38	2.4	-0.02	-0.8333	443516.0
"002426.SZ"	"20220713"	2.3	2.43	2.29	2.4	2.3	0.1	4.3478	754958.0
"002426.SZ"	"20220712"	2.36	2.37	2.27	2.3	2.36	-0.06	-2.5424	637041.0
"002426.SZ"	"20220711"	2.39	2.41	2.35	2.36	2.39	-0.03	-1.2552	452000.05

导入代码范围内的其他股票行情数据

```
import pandas as pd
from tqdm.notebook import tqdm
```

```
hq = [
    pl.from_pandas(
        pro.daily(
            ts_code=ts_code,
            start_date=format(start_date, "%Y%m%d"),
            end_date=format(end_date, "%Y%m%d"),
        )
    )
    for ts_code in tqdm(ts_codes)
]
```

```
100% ████████████████████████████████ 149/149 [00:06<00:00, 30.14it/s]
```

与 d1 用交易日期和证券代码进行匹配

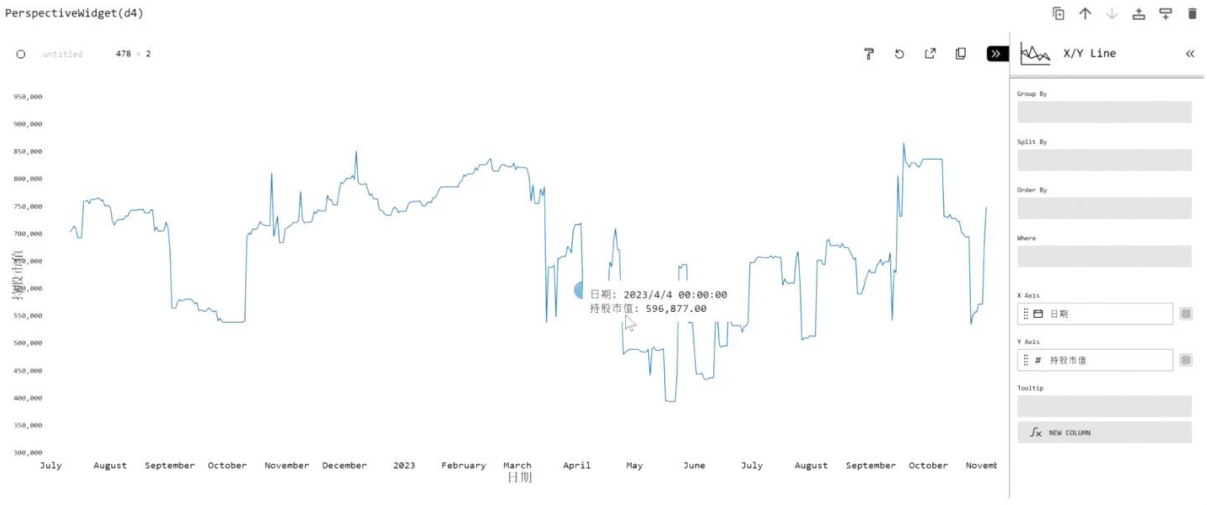
```
d1.join(
    hq, left_on=["交易日期", "证券代码"], right_on=["trade_date", "ts_code"], how="left"
)
```

shape: (358, 27)

序号	券商	交易日期	交易时间	证券代码	证券名称	买卖标志	成交价格	成交数量	成交金额	手续费	印花税	过户费	其他费	发生金额	手续费率	印花税率	过户费率	open	high	low	close	pre_close	change	pct_ch
u32	str	date	str	str	str	str	f64	f64	f64	f64	f64	f64	f64	f64	f64	f64	f64	f64	f64	f64	f64	f64	f64	f6
1	"湘财"	2022-07-11	"09:33:37"	"000900"	"现代投资"	"买入"	4.05	34400.0	139320.0	22.29	0.0	1.39	0.0	-139342.29	0.00016	0.0	0.00001	4.08	4.13	4.04	4.12	4.06	0.06	1.477
2	"湘财"	2022-07-11	"09:34:24"	"601077"	"渝农商行"	"买入"	3.65	38300.0	139795.0	22.37	0.0	1.38	0.0	-139818.75	0.00016	0.0	0.00001	3.65	3.68	3.64	3.66	3.65	0.01	0.27
3	"湘财"	2022-07-11	"09:36:30"	"600894"	"广日股份"	"买入"	6.54	21400.0	139956.0	22.39	0.0	1.41	0.0	-139979.8	0.00016	0.0	0.00001	6.57	6.57	6.49	6.51	6.57	-0.06	-0.913
4	"湘财"	2022-07-11	"09:37:25"	"601992"	"金隅集	"买入"	2.59	54000.0	139860.0	22.38	0.0	1.42	0.0	-139883.8	0.00016	0.0	0.00001	2.61	2.62	2.58	2.59	2.61	-0.02	-0.766

```
d4=d3.join(
    hq,
    left_on=["日期", "证券代码"],
    right_on=["交易日期", "证券代码"],
    how="left",
).sort("日期", "证券代码").with_columns(
    close=pl.col("close").full_null(strategy="forward").over("证券代码")
).with_columns(
    持股市值=pl.col("结余数量")*pl.col("colse")).group_by("日期").agg(
    pl.col("持股市值").sum())
)
```

可视化股票持仓市值



导入沪深 300 数据

```python
ihq = pro.index_daily(
    ts_code="000300.SH",
    start_date=format(start_date, "%Y%m%d"),
    end_date=format(end_date, "%Y%m%d"),
    fileds="ts_code,trade_date,pct_chg",
)
```

```python
pl.from_pandas(ihq).write_parquet("index_daily.parque")
```

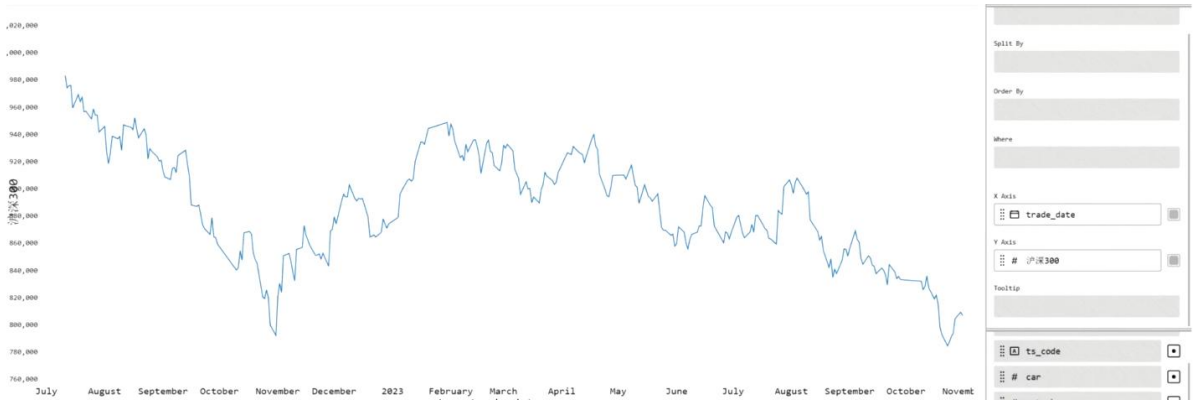
```python
ihq
```

```python
ihq = pl.read_parquet("index_daily.parquet")
ihq = (
    ihq.with_columns(
        pl.col("pct_chg") / 100 + 1,
    )
    .sort("trade_date")
    .with_columns(
        car=pl.col("pct_chg").cum_prod(),
    )
    .with_columns(
        沪深300=pl.col("car") * 100_0000,
    )
)
ihq
```

shape: (318, 5)

ts_code	trade_date	pct_chg	car	沪深300
str	str	f64	f64	f64
"000300.SH"	"20220711"	0.983254	0.983254	983254.0
"000300.SH"	"20220712"	0.990585	0.973997	973996.66359
"000300.SH"	"20220713"	1.001818	0.975767	975767.389524
"000300.SH"	"20220714"	1.000142	0.975906	975905.948494
"000300.SH"	"20220715"	0.982983	0.959299	959298.956968
…	…	…	…	…
"000300.SH"	"20231025"	1.004969	0.791288	791288.453778
"000300.SH"	"20231026"	1.002764	0.793476	793475.575065
"000300.SH"	"20231027"	1.013727	0.804368	804367.614284
"000300.SH"	"20231030"	1.006003	0.809196	809196.233072
"000300.SH"	"20231031"	0.996856	0.806652	806652.120115

可视化查看本金变化



调整列宽

```
d5 = d4.join(ihq, left_on="日期", right_on="trade_date")
d5.unpivot(on=["总资产", "沪深300"], index="日期")
```

shape: (636, 3)

日期	variable	value
date	str	f64
2022-07-11	"总资产"	1.0035e6
2022-07-12	"总资产"	1.0082e6
2022-07-13	"总资产"	1.0143e6
2022-07-14	"总资产"	1.0105e6

进行市值和总资产的可视化对比