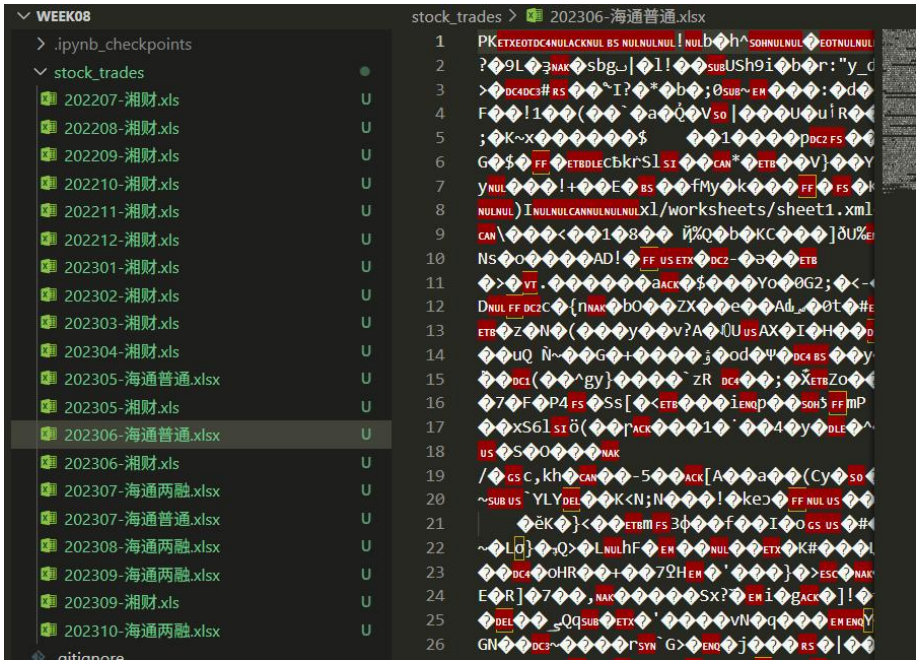
金融计算与编程第八周学习笔记

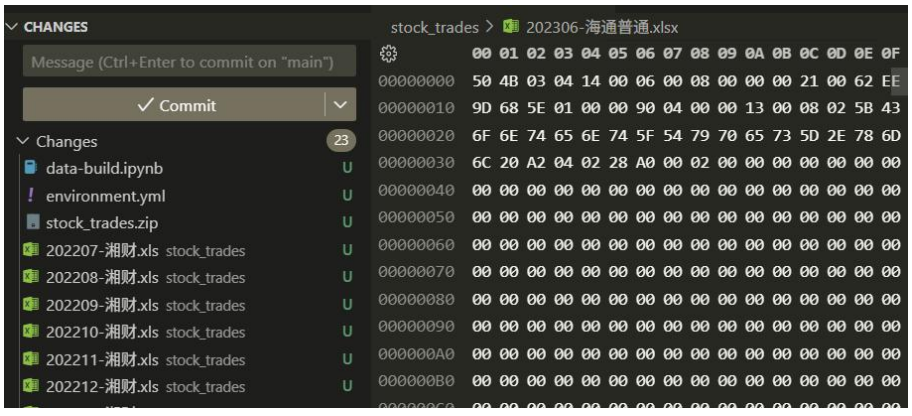
1. 新建并激活环境。

```
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 77002    0 77002    0     0  90462      0 --:--:-- --:--:-- --:--:-- 90697
(week08)
ASUS@%C▒▒▒▒u▒▒▒▒▒n MINGW64 ~/repo/week08 (main)
$ ls -l
total 101
-rw-r--r-- 1 ASUS 197121 18805 May  8 20:43 LICENSE
-rw-r--r-- 1 ASUS 197121  2239 May  8 20:43 README.md
-rw-r--r-- 1 ASUS 197121   293 May  8 20:59 environment.yml
-rw-r--r-- 1 ASUS 197121 77002 May  8 21:08 stock_trades.zip
(week08)
ASUS@%C▒▒▒▒u▒▒▒▒▒n MINGW64 ~/repo/week08 (main)
$ ls -lh
total 101K
-rw-r--r-- 1 ASUS 197121  19K May  8 20:43 LICENSE
-rw-r--r-- 1 ASUS 197121 2.2K May  8 20:43 README.md
-rw-r--r-- 1 ASUS 197121  293 May  8 20:59 environment.yml
-rw-r--r-- 1 ASUS 197121  76K May  8 21:08 stock_trades.zip
(week08)
ASUS@%C▒▒▒▒u▒▒▒▒▒n MINGW64 ~/repo/week08 (main)
$ unzip stock_trades.zip

Archive:  stock_trades.zip
```

2. 在 jypter 里面文件路径可以自动补全（Tab 键），Vscode 打不开二进制文件，安装 Hex Editor 可以查看二进制文件，





有乱码的原因是，VS Code 将二进制字节 解码 (decode) 为文本代码(Unicode，对应着各国字符) 时，默认使用了错误的编解码器 (encoding)类似于各种文字，例如什么藏文蒙文之类的，不用对翻译方式翻译不对。当初导出/保存这个文件所用的券商交易软件，应该是使用了 GB18030 编解码器 (简体中文 Windows 操作系统的默认选择) 将文本代码 (Unicode) 编码 (encode) 为二进制字节。而现代软件 (尤其是在 macOS、Ubuntu 等类 Unix 操作系统里，以及 Windows 下的 VS Code) 默认都使用的是 UTF-8 编解码器。解码所用的如果与编码所用的不匹配，就会翻译错误，显示出乱码。

选择 GB18030 编解码器，改变解码器就能正常查看：

3. 是 CSV 格式，而且分隔符 (seporator) 不是逗号 (,)，而是 TAB (\t)。语句不显示东西，表达式显示。

Int64 为 polars 的类型。

加列会改变架构（schema），加行不会。

```
'="0.00"',
'股息入账:赣粤高速600269; 权益股数:40700;',
'人民币')
```

```
[43]: type(df.row(0))
```

```
[43]: tuple
```

```
[47]: r = df.rows()
```

```
[48]: type(r)
```

```
[48]: list
```

4. 需要清洗数据

```
df[:,"证券代码"].unique().to_list()
```

```
[34]:
['600269',
 '="000900"',
 '204007',
 '600648',
 '600015',
 '601992',
 '="002462"',
```

5. 在函数内整体得缩进

```
[85]: def read_df_湘财(f: str | Path) -> pl.DataFrame:
          df = pl.read_csv(
              f,
              encoding="gb18030",
              separator="\t",
              infer_schema=False,
          )
          df = df.with_columns(
              pl.selectors.all().str.strip_prefix("=").str.strip_chars('"'),
              pl.col("发生日期").cast(pl.Utf8).str.to_date("%Y-%m-%d"),
              pl.col("成交时间").cast(pl.Utf8).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(["证券买入", "证券卖出"]),
          )
          return df
```

```
[ ]:
```

```
[112]: d1.with_columns(
           券商=pl.lit("湘财"),
       )
```

[112]: 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"

处理完数据记得保存

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	成交金额2	成交金额D	发生金额D
str	date	time	str	str	str	f64	f64	f64	f64	f64	f64	f64	f64	f64	f64	f64
"湘财"	2022-07-18	09:38:10	"002462"	"嘉事堂"	"卖出"	13.2062	10400.0	137344.0	21.98	137.35	1.38	0.0	137184.67	137344.48	-0.48	1.38
"湘财"	2022-07-18	09:44:52	"600408"	"安泰集团"	"买入"	3.19	47000.0	149930.0	23.99	0.0	1.51	0.0	-149955.5	149930.0	0.0	-299860.0
"湘财"	2022-07-18	09:44:31	"600648"	"外高桥"	"买入"	12.6066	11900.0	150019.0	24.0	0.0	1.49	0.0	-150044.49	150018.54	0.46	-300038.0

6. 语句不显示表达式会显示。
7. 出现了一个奇怪的现象：

```
[11]: df.with_columns(
          手续费率=pl.col("手续费")/pl.col("成交金额")
      )
```

[11]: shape: (363, 15)

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
str	date	time	str	str	str	f64	f64	f64	f64	f64	f64	f64	f64	f64
"湘财"	2022-07-18	09:38:10	"002462"	"嘉事堂"	"卖出"	13.2062	10400.0	137344.0	21.98	137.35	1.38	0.0	137184.67	0.00016
"湘财"	2022-07-18	09:44:52	"600408"	"安泰集团"	"买入"	3.19	47000.0	149930.0	23.99	0.0	1.51	0.0	-149955.5	0.00016
"湘财"	2022-07-18	09:44:31	"600648"	"外高桥"	"买入"	12.6066	11900.0	150019.0	24.0	0.0	1.49	0.0	-150044.49	0.00016

后面的逗号不能忘：

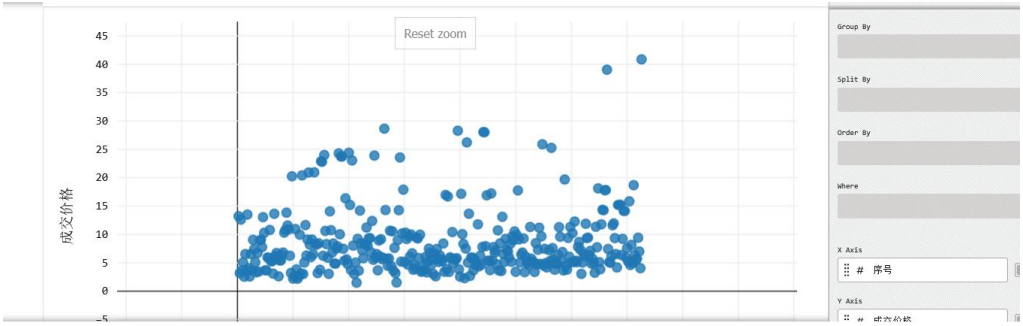
```
[12]: df.with_columns(
          手续费率=pl.col("手续费")/pl.col("成交金额"),
          印花税率=pl.col("印花税")/pl.col("成交金额"),
          过户费率=pl.col("过户费")/pl.col("成交金额")
      )
```

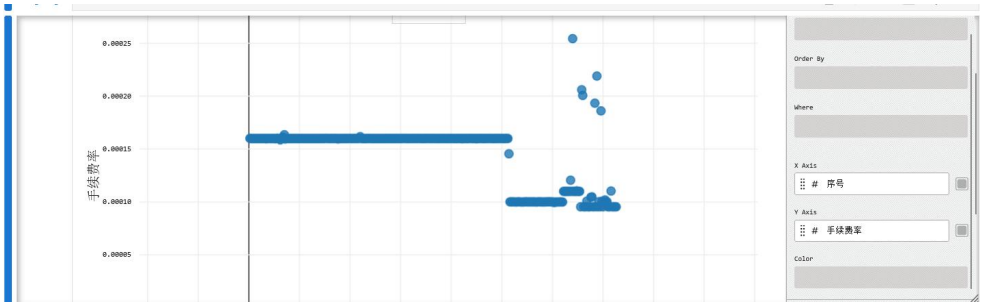
"湘财"	2022-07-18	09:38:10	"002462"	"嘉事堂"	"卖出"	13.2062	10400.0	137344.0	21.98	137.35	1.38	0.0	137184.67	0.00016	0.001	0.00001
"湘财"	2022-07-18	09:44:52	"600408"	"安泰集团"	"买入"	3.19	47000.0	149930.0	23.99	0.0	1.51	0.0	-149955.5	0.00016	0.0	0.00001

```
[27]: PerspectiveWidget(df)
```

[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
1	湘财	2022/7/17	09:38:10	002462	嘉事堂	卖出	13.21	10,400.00	137,344.00	21.98	137.35	1.38	0.00	137,184.67	0.00
2	湘财	2022/7/17	09:44:52	600408	安泰集团	买入	3.19	47,000.00	149,930.00	23.99	0.00	1.51	0.00	-149,955.50	0.00
3	湘财	2022/7/17	09:44:31	600648	外高桥	买入	12.61	11,900.00	150,019.00	24.00	0.00	1.49	0.00	-150,044.49	0.00
4	湘财	2022/7/17	09:43:38	600269	赣粤高速	买入	3.69	40,700.00	150,183.00	24.03	0.00	1.50	0.00	-150,208.53	0.00
5	湘财	2022/7/17	09:42:51	600015	华夏银行	买入	5.07	30,000.00	152,100.00	24.34	0.00	1.52	0.00	-152,125.86	0.00
6	湘财	2022/7/17	09:39:28	601992	金隅集团	卖出	2.57	54,000.00	138,686.00	22.19	138.69	1.38	0.00	138,523.74	0.00
7	湘财	2022/7/17	09:39:06	600894	广日股份	卖出	6.54	21,400.00	139,996.00	22.40	140.02	1.46	0.00	139,832.12	0.00
8	湘财	2022/7/17	09:38:30	601077	渝农商行	卖出	3.58	38,300.00	137,114.00	21.94	137.13	1.38	0.00	136,953.55	0.00
9	湘财	2022/7/10	09:38:16	002462	嘉事堂	买入	13.51	10,400.00	140,504.00	22.48	0.00	1.41	0.00	-140,526.48	0.00
10	湘财	2022/7/10	09:33:37	000900	现代投资	买入	4.05	34,400.00	139,320.00	22.29	0.00	1.39	0.00	-139,342.29	0.00





8. Cum 是累计相加：

```
[92]: d = pl.DataFrame({"n": [500, 0, 0, -200, 300, 0, -400]})
      d.with_columns(running_n=pl.col.n.cum_sum())
```

[92]: shape: (7, 2)

n	running_n
i64	i64
500	500
0	500
0	500
-200	300
300	600
0	600
-400	200

9. 日期格式必须是八位的数字：

```
[25]: hq = pro.daily(ts_code="002462.SZ", start_date="20220711", end_date="20231031")
[27]: hq
```

[27]:

	ts_code	trade_date	open	high	low	close	pre_close	change	pct_chg	vol	amount
0	002462.SZ	20231031	14.75	14.90	14.59	14.70	14.75	-0.05	-0.3390	65859.96	96984.271
1	002462.SZ	20231030	13.88	14.89	13.88	14.75	13.93	0.82	5.8866	123932.16	180119.372
2	002462.SZ	20231027	13.70	13.98	13.51	13.93	13.64	0.29	2.1261	35782.00	49386.168
3	002462.SZ	20231026	13.49	13.68	13.40	13.64	13.62	0.02	0.1468	19215.00	26005.866
4	002462.SZ	20231025	13.65	13.77	13.58	13.62	13.67	-0.05	-0.3658	18484.00	25274.163

~可以检查：

```
[118]: d1.join(
           hq, left_on=["交易日期", "证券代码"], right_on=["trade_date", "ts_code"], how="left"
       ).filter(
           ~pl.col("成交价格").is_between(pl.col("low"), pl.col("high")),
       )
```

[118]: shape: (0, 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_chg	vol
u32	str	date	str	str	str	str	f64	f64	f64	f64	f64	f64	f64	f64	f64	f64	f64	f64	f64	f64	f64	f64	f64	f64	f64

```
[152]: PerspectiveWidget(d4)
```

```
[152]:
```