**Historical weekly stock closed prices analysis**

**Objective:**

Stock option expired at last trade date of each week. Normally each Friday, but if Friday is a holiday, last market trade date will be on Thursday and weekly option will be expired on.

Compare weekly closed price of last trade date with prior week and calculate percentage change. Sum up each tracking ticker of win and lost running total within specified years.

**Environment setup:**

Database: PostgreSQL

2 tables: market\_holiday and stockprice

Run sql script : Create\_Tables\_script.sql

Load sample holidays data from 2016 to 2020:

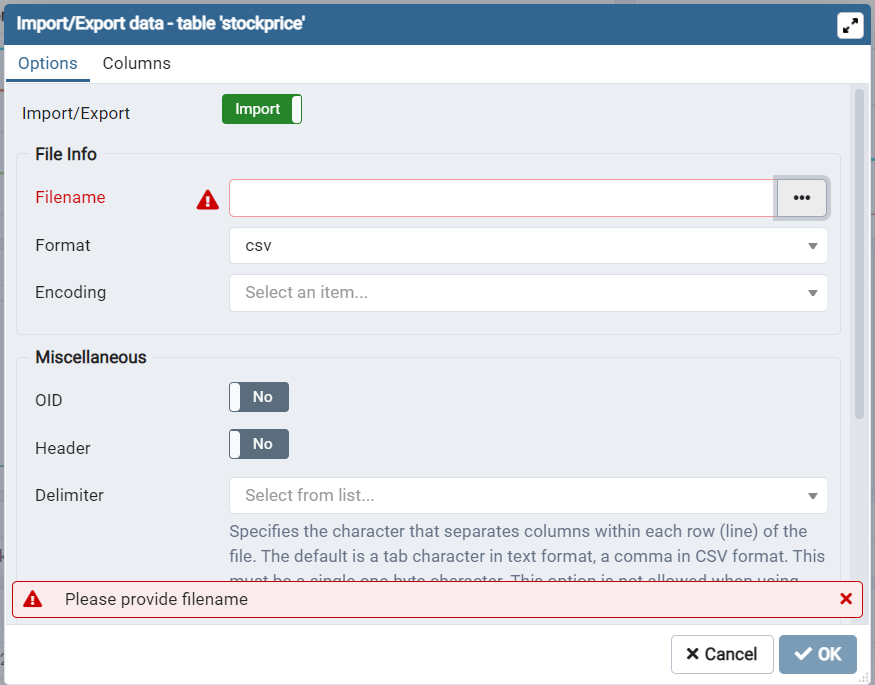
Run sql script : market\_holiday\_2016\_to\_2021.sql

Upload historical daily stock price, csv file for each ticker to PostgreSQL:

\* Download all .csv data file to local drive

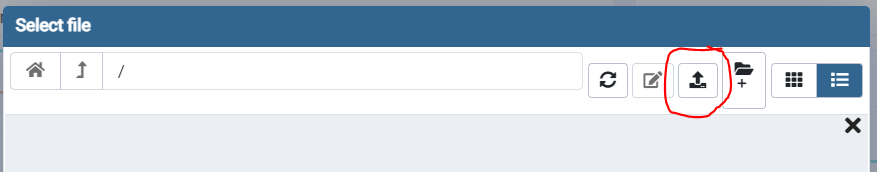
Use pgAdmin 4 to import each ticker csv file into stockprice table

\* right click at table name stockprice and select import/export

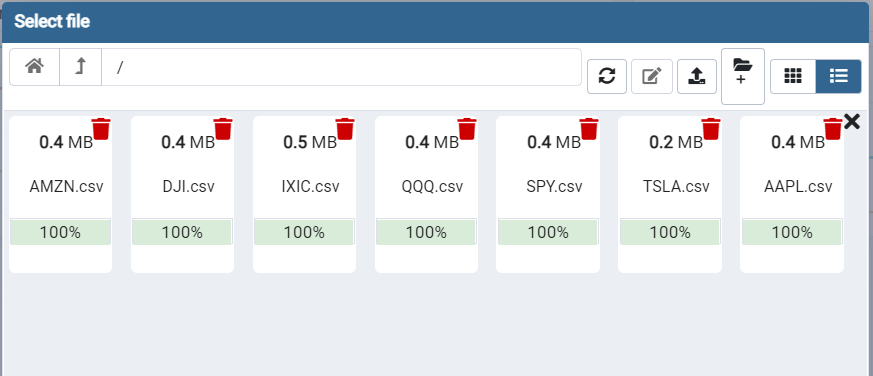


\* Click on button ‘…’ right of Filename field:

\* Click on upload button top right area with up arrow



\* Locate historical .csv files and drag into upload popup window ‘select file’.



\* Click cancel button back to import/export data window.

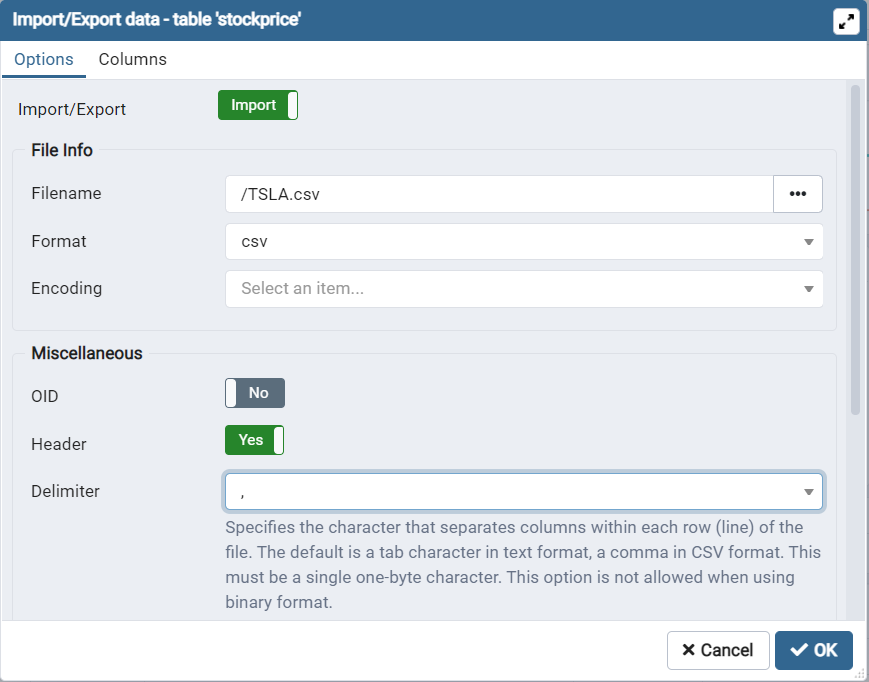
\* import/export: select import

Filename: click ‘…’ button and select a .csv file

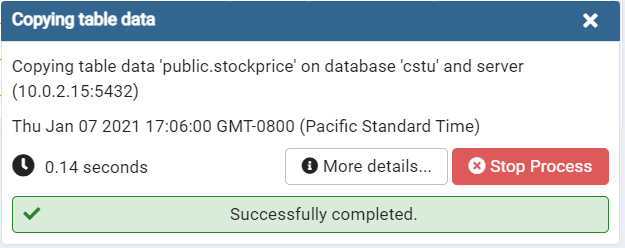
Format: csv

Header: Yes

Delimiter: ,



\* Click OK button to import .csv file to stockprice table. Follow screen will show if import successful.



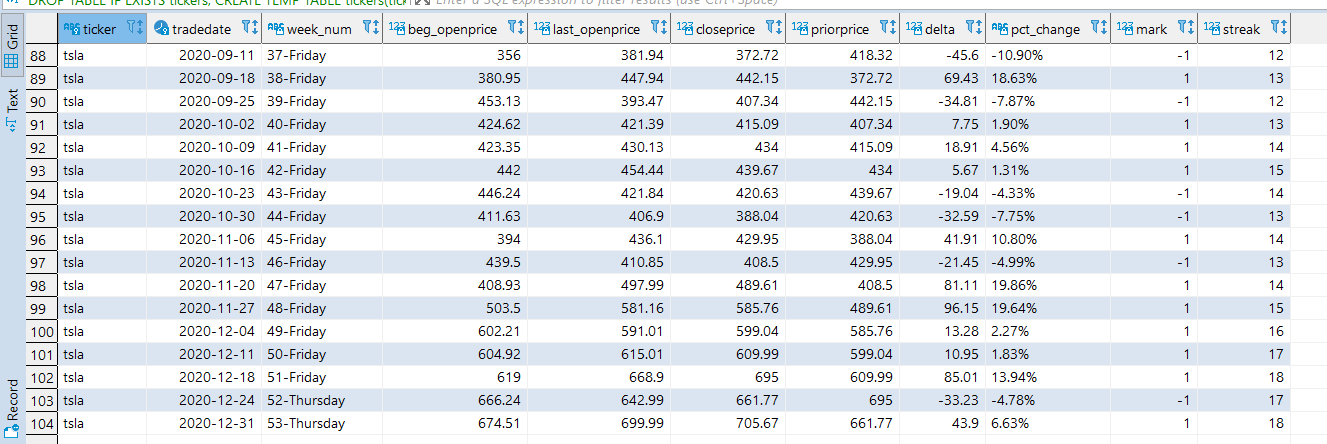
\* Repeat each ticker .csv files to import into stockprice table.

**Historical stock price data analysis:**

**\*** Cut and Paste query from: Weekly\_Stock\_Closed\_Price\_Analysis\_Script.sql

\* Use any query browser tool, my example DBeaver, and run the sql script.

Following is sample query result:



**Next step and tasks for improvement the process:**

Write a Python code with rest API to continue retrieve history data to build up the stock price data bank for specify tickers. Also possible setup schedule job and load historical data automatically.