New York Stock Exchange Trade and Quote Database Access

General Information

- Offered through the Applied Mathematics and Statistics Department at Stony Brook
- Over 20TB of high-frequency equity tick data is accessible, spanning the last decade
- Hands-on empirical data analysis for coursework and research projects

Requirements for Access

- Graduate Students within the AMS department
- Undergraduate Students enrolled in a quantitative finance course in the AMS department

Contact Information

- Server & Database Access and Maintenance- Victor Poon victor.poon@stonybrook.edu
- Research Supervisor- Professor Pawel Polak pawel.polak@stonybrook.edu
- General Questions - Jason Bohne jason.bohne@stonybrook.edu
- TAQ Database Group in Slack for general questions; Invite Link

Technology Required

- Bash (Shell- Scripting)
- Python Scripting
- Slack (Access to the TAQ Database Group)

Server Access

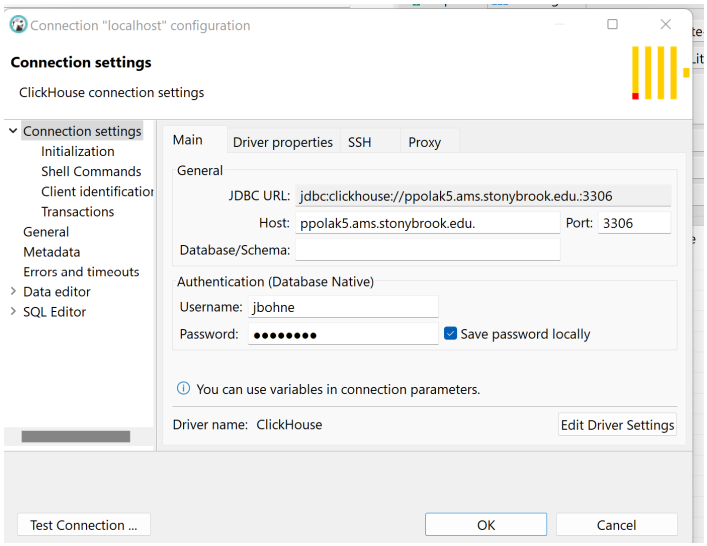
- Email Victor Poon victor.poon@stonybrook.edu with the Subject “AMS TAQ Database Access” and your **student ID** and **status (undergraduate vs. graduate)**
- CC Professor Pawel Polak pawel.polak@stonybrook.edu and Jason Bohne jason.bohne@stonybrook.edu on the above email

Database Access

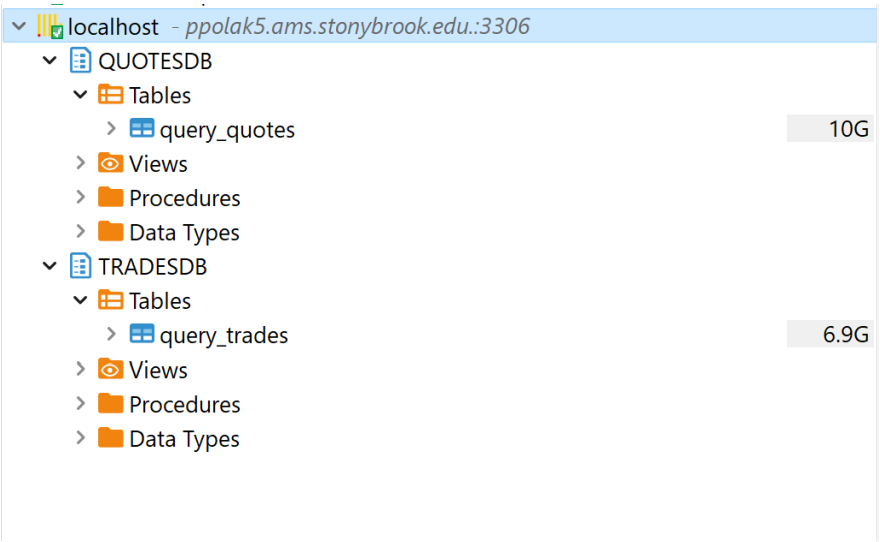
When requesting server access, share your desired method of access (DBeaver or Advanced). Either approach allows for access with minimal interaction with logging into the server.

- **Python Access (Graduate Students and Faculty)**
  - Remote access with Python client in TAQ-Query-Scripts-Public
  - **Requires familiarity** with GitHub, Conda, and Linux, if not see below
  - Best-suited for users that require frequent queries/large amounts of data
  - For example, working with big data architectures/ machine learning models
  - Contact jason.bohne@stonybrook.edu for more information

- **DBeaver (Undergraduate)**
  - Download DBeaver
  - Create New Database Connection with host: `ppolak5.ams.stonybrook.edu.` on port `3306` with **your** username and password



  - You should see query_trades and query_quotes tables respectively within the trade and quote databases



- Click on these tables to see a snapshot. You can query via SQL expressions in the filter bar or via SQL script. See here for SQL tutorials

Notes

- You can query trades and quotes directly from the filter bar
- Find the Database Schema within the table Properties -> DDL
- You can export data via CSV for a more thorough statistical analysis via File -> Export

Example Trade Queries

Query by Symbol:

SELECT x. FROM TRADESDB.query_trades x WHERE Symbol='AAPL'*

Query by Symbol and Single Date

SELECT x. FROM TRADESDB.query_trades x WHERE Symbol='AAPL' AND Date ='2020-01-07'*

Query by Symbol and Multiple Dates

SELECT x. FROM TRADESDB.query_trades x WHERE Symbol='AAPL' AND Date >='2020-01-07' and `Date` <'2020-01-11'*

Example Quote Queries

Query by Symbol:

SELECT x. FROM QUOTESDB.query_quotes x WHERE Symbol='AAPL'*

Query by Symbol and Single Date

SELECT x. FROM QUOTESDB.query_quotes x WHERE Symbol='AAPL' AND Effective_Date ='2020-01-07'*

Query by Symbol and Multiple Dates

SELECT x. FROM QUOTESDB.query_quotes x WHERE Symbol='AAPL' AND Effective_Date >='2020-01-07' AND Effective_Date <'2020-01-11'*

Full Trade Table Schema

```
   `Time` DateTime64(9,
'America/New_York'),

   `Exchange` Nullable(String),

   `Symbol` Nullable(String),

   `Sale_Condition` Nullable(String),

   `Trade_Volume` Nullable(UInt64),

   `Trade_Price` Nullable(Float64),

   `Trade_Stop_Stock_Indicator` Nullable(String),

   `Trade_Correction_Indicator` Nullable(UInt8),

   `Sequence_Number` Nullable(UInt64),

   `Trade_Id` Nullable(UInt64),

   `Source_of_Trade` Nullable(String),

   `Trade_Reporting_Facility` Nullable(String),

   `Participant_Timestamp` Nullable(UInt64),

   `Trade_Reporting_Facility_TRF_Timestamp` Nullable(UInt64),

   `Trade_Through_Exempt_Indicator` Nullable(UInt64),

   `Date` Date,

   `YearMonth` String
```

Full Quote Table Schema

```
`Time` String,
   `Exchange` String,
   `Symbol` String,
   `Bid_Price` Float64,
   `Bid_Size` Float64,
   `Offer_Price` Float64,
   `Offer_Size` Float64,
   `Quote_Condition` String,
   `Sequence_Number` UInt8,
   `FINRA_BBO_Indicator` String,
   `FINRA_ADF_MPID_Indicator` String,
   `Source_Of_Quote` String,
   `Best Bid Quote Condition` String,
   `Best_Bid_Exchange` String,
   `Best_Bid_Price` Float64,
   `Best_Bid_Size` Float64,
   `Best_Bid_FINRA_Market_Maker_ID` String,
   `Best_Offer_Quote_Condition` String,
   `Best_Offer_Exchange` String,
   `Best_Offer_Price` Float64,
   `Best_Offer_Size` Float64,
   `Best_Offer_FINRA_Market_Maker_ID` String,
   `LULD_Indicator` String,
   `LULD_NBBO_Indicator` String,
   `SIP_Generated_Message_Identifier` String,
   `Participant_Timestamp` UInt32,
   `FINRA_ADF_Timestamp` UInt32,
   `Security_Status_Indicator` String,
   `National_BBO_Ind` String,
   `Quote_Cancel_Correction` String,
   `Effective_Date` DateTime
```