This database is a loose recreation of my CS340 final’s database. The obvious difference now and then is that this database only contains index data such as the S&P500 as well as various macroeconomic indices. The original was purely single stock data, but I no longer have access. Structurally, the new database is less complex, and I would say streamlined, containing only rows with three points of data, the date, value, and what units that value is in.

The SQL in this demonstrates the CRUD functions you would expect out of a database interface. I implemented the handler as a class, to should a future extension want to run multiple connections. The extension would only need to create another SeriesDB object. I would hope the program serves as a good showcase and replicate on a small scale a common use of a SQL database. The connections and retrieves real world data from Federal Reserve, caches it locally with code to keep the data up-to-date, and finally passes that data onto the chart generation code.

The caching improvement increased the responsiveness of the app, and also reduced the number of calls to the FRED api.

I had some confusion with series missing blocks of data when I viewed them in the chart. DB Browser helped me see what was going wrong, and after eyeballing the data, I realized that the missing data corresponded to already existing dates in other series. I dropped the DISTINCT keyword from the obs\_date properties, because in a flat table with multiple series the same date can exist. I made the table flat so I didn’t have to dynamically generate tables, to help with the SQL injection attacks, but I might go back and create carefully escaped code to give each series a table.

SQL injections attack were mitigated through utilizing question mark substitution from Python’s sqlite library. SQL injections insert code in an input field and then becomes unwittingly processed by the database. Question mark substitution prevents injection attacks by marking input to be accepted for values only. Also, this program possesses a smaller attack surface as it doesn’t have any fields accepting arbitrary data. We’re I to expand the available time series, I would probably enumerate the available sources by creating a new index table listing all the available sources. This doesn’t have to be limited to FRED data. But no matter the data source, the possible inputs are known ahead of time.