**ETL Project**

**Project Team:**

Jordan Gilmartin, Lynell Robinson, Min Xe, Nick Mangarella, Sushama Kunnath

**Introduction**

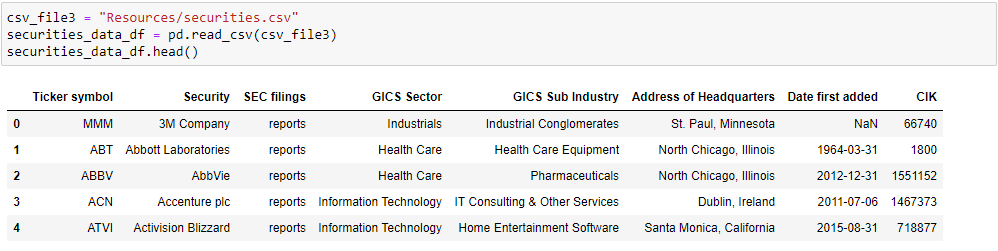
For this ETL project, we used Kaggle as our public source data platform. The datasets are based on the New York Stock Exchange S&P 500 company’s historical prices with fundamental data ranging over various years from 2010 to 2016. The datasets include are:

* Fundamental Data - <https://www.kaggle.com/dgawlik/nyse?select=fundamentals.csv>
* Security Data - <https://www.kaggle.com/dgawlik/nyse?select=securities.csv>
* Prices Data - <https://www.kaggle.com/dgawlik/nyse?select=prices.csv>

**Extraction**

* Identified the dataset file types
  + All of our datasets originate from CSV files
* Extracted the data from each of the CSV files and load the data into separate DataFrames.

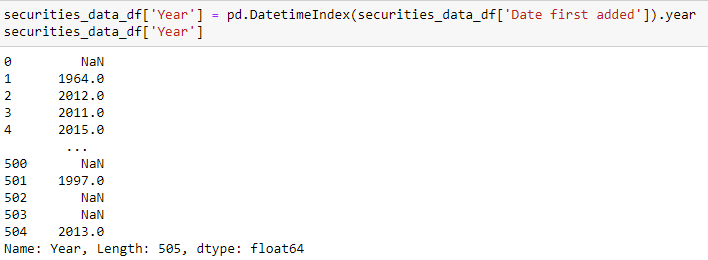
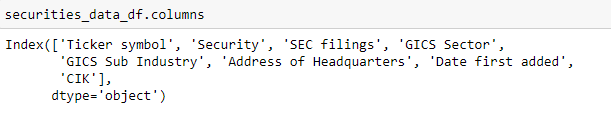
Figure 1: One of the converted CSV files to a DataFrame

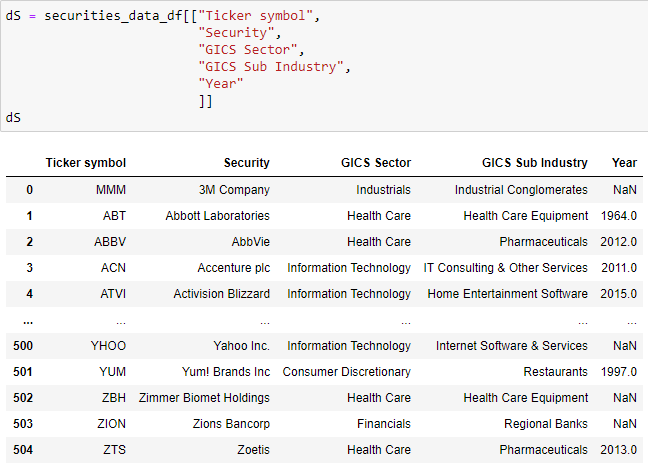


**Transformation**

* For each of the datasets:
  + Identified data columns relevant to the proposed inquiry
  + Created or used a year column from an existing date column
  + Formatted the year column to the same data type in each DataFrame
  + Created new DataFrames with the columns relevant to the proposed inquiry

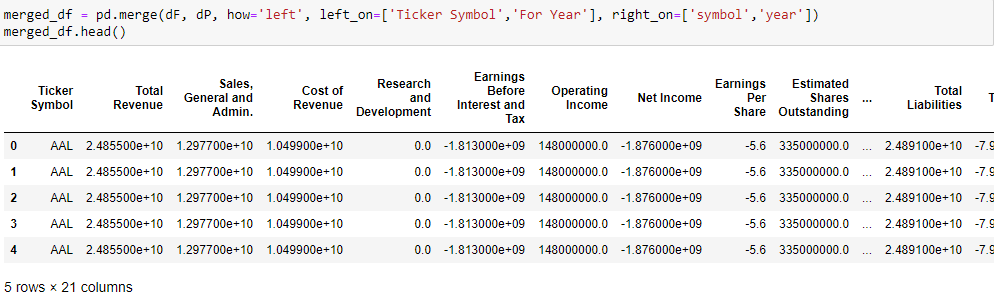
Figure 2: The above-described process for one of the DataFrames





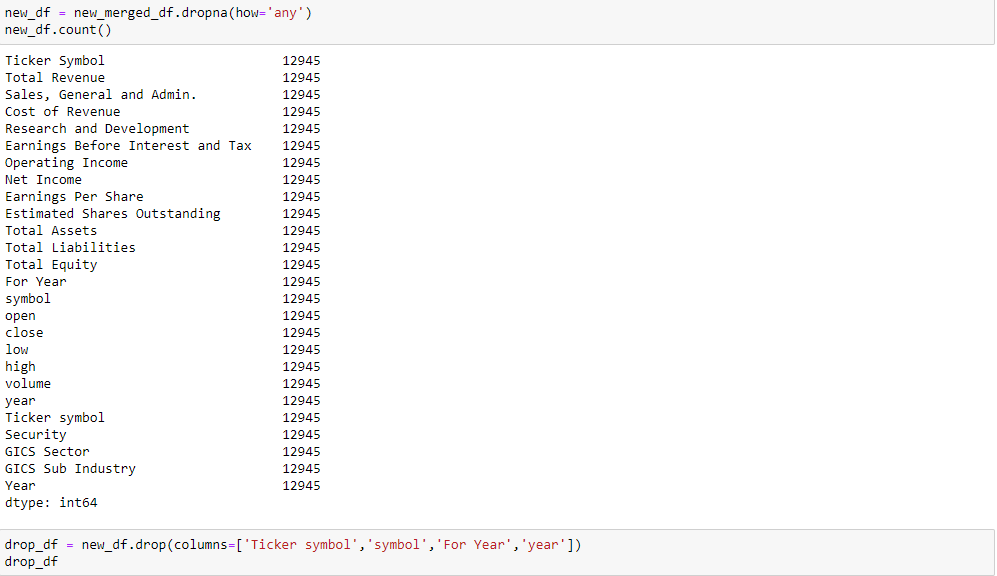
* Merged the three datasets on the Ticker Symbol and Year

Figure 3: One of the merged DataFrames



* Cleaned the DataFrame
  + Dropped the null values within the new dataset
  + Dropped the duplicate ticker symbol and year columns

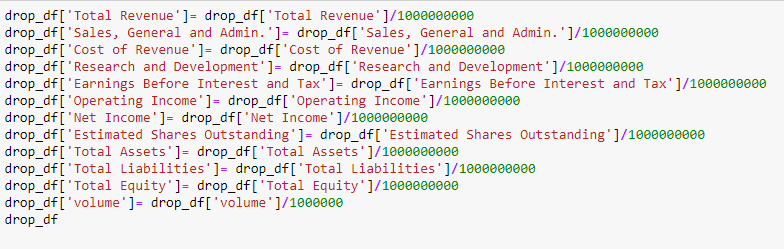
Figure 4: Cleaning of the DataFrame

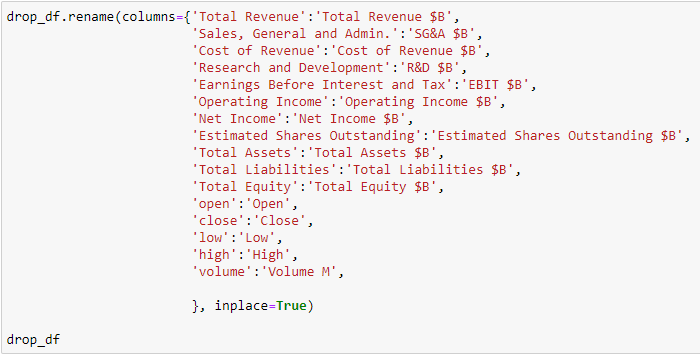


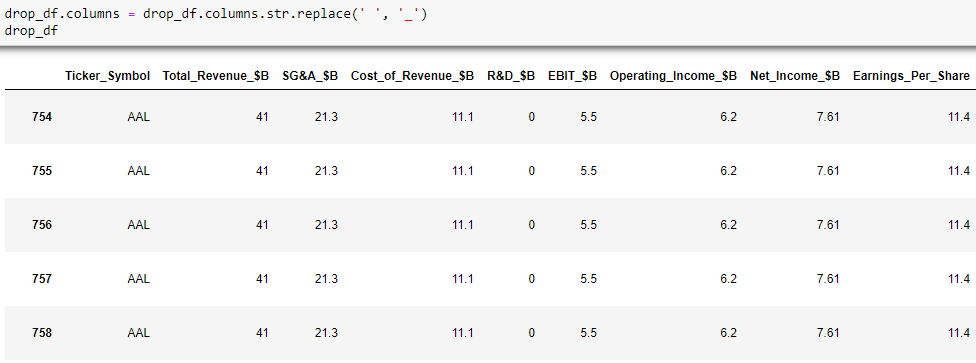
* Formatted the DateFrame
  + Formatted the year column
  + To make the exponential column data more readable
    - Divided by a number (billion or million)
    - Formatted the decimal places
    - Renamed the column to represent the exponent (B or M)
  + Replaced all spaces within the column names to underscores for simpler table name conversion

Figure 5: Formatting of the DataFrame and the Final DateFrame result







**Load**

* Connected to the PostgreSQL database
* Use pandas to load the merged converted DataFrame into the database

Figure 6: The load process into a PostgreSQL Database

