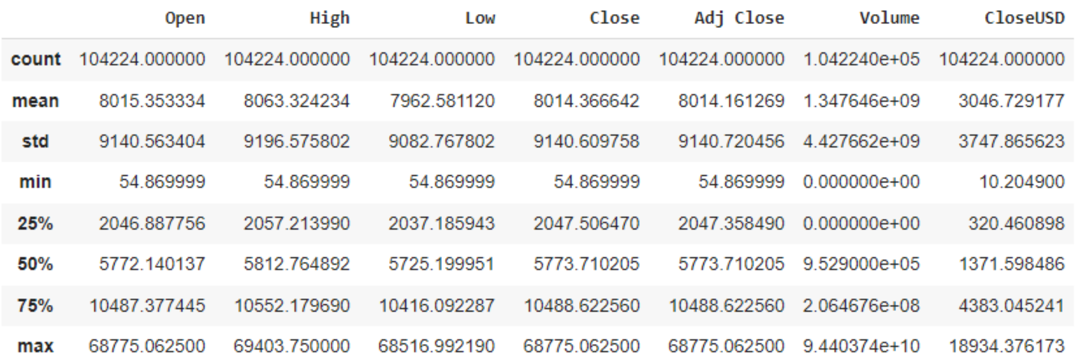
CIS4500 - Final Project Proposal

1. **Team Members:**
    a. Ashish Pothireddy:
        i. Email: ashishpr@wharton.upenn.edu
        ii. Github username: @ashishpreddy
    b. Austin Wang:
        i. Email: austinwa@seas.upenn.edu
        ii. Github username: @austinw1995
    c. Alex Huang:
        i. Email: ahuang21@seas.upenn.edu
        ii. Github username: @AlexHuang21
    d. Noah Erdogan:
        i. Email: noahe@sas.upenn.edu
        ii. Github username: @noah-erdogan
2. **Application Description:**
    a. Our application aims to provide users with a comprehensive understanding of stocks' and indices' performances over time, with a specific focus on the 2013-2018 period. The 2013-2018 period holds several key similarities to the present, which makes it an interesting project for us to undertake. Namely, the 2013-2018 period was part of the global economic recovery from the 2008 Financial Crisis, and we similarly are in the recovery period from the COVID recession. There were also contrasting market dynamics and governmental policies in the 2013-2018 period with significant bull and bear runs and ever-shifting monetary/fiscal policies, which is very similar to the economic market we are currently in. As such, we believe the analysis that can be produced from this project has significant implications for our present.
    b. With our two datasets, we plan to provide users with the ability to analyze not only individual stock performances and indices but also compare and contrast selected stock performances against chosen indices and against other stocks. The vast amount of data available to us allows us to produce very interesting queries and analysis, ranging from stock/index volatility analysis, visualizations of

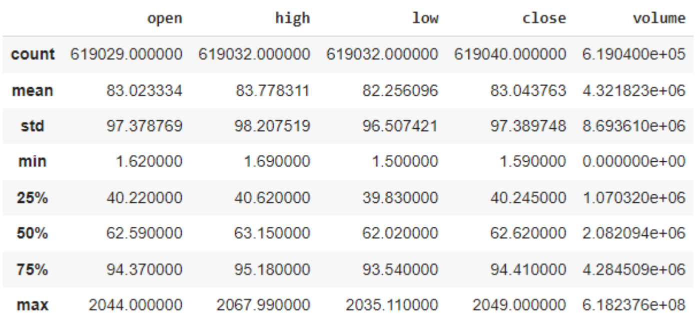
time-series performance data, best/worst performers, and much more. The ultimate goal is to provide an immensely flexible application that provides a user-friendly interface, allowing all users to explore/analyze the extent of the data available to them and draw their own conclusions for the looming recessionary period.

3. **Chosen Datasets:**
   a. Stock Exchange Data (https://www.kaggle.com/datasets/mattiuzc/stock-exchange-data?select=indexData.csv), the dataset contains performance data on 14 different major stock indices inducing the NYSE and NASDAQ, from 1965 to 2021. This dataset notably does not provide the S&P500 index, allowing us to further the complexity of our project by artificially creating the S&P500 index with our other dataset. The dataset contains 3 separate files, but we'll be focusing on the index_processed.csv file, which is 10 mB. The file contains 9 attributes: a ticker for the index, date of observation, the open, the high, the low, the close, the close adjusted for dividends and stock splits, the total trading volume, and the closeUSD. With a total 104224 rows, the mean and standard deviation of the dataset is as follows:



	Open	High	Low	Close	Adj Close	Volume	CloseUSD
count	104224.000000	104224.000000	104224.000000	104224.000000	104224.000000	1.042240e+05	104224.000000
mean	8015.353334	8063.324234	7962.581120	8014.366642	8014.161269	1.347646e+09	3046.729177
std	9140.563404	9196.575802	9082.767802	9140.609758	9140.720456	4.427662e+09	3747.865623
min	54.869999	54.869999	54.869999	54.869999	54.869999	0.000000e+00	10.204900
25%	2046.887756	2057.213990	2037.185943	2047.506470	2047.358490	0.000000e+00	320.460898
50%	5772.140137	5812.764892	5725.199951	5773.710205	5773.710205	9.529000e+05	1371.598486
75%	10487.377445	10552.179690	10416.092287	10488.622560	10488.622560	2.064676e+08	4383.045241
max	68775.062500	69403.750000	68516.992190	68775.062500	68775.062500	9.440374e+10	18934.376173

   b. S&P 500 Stock Data (https://www.kaggle.com/datasets/camnugent/sandp500), the data set contains 2013-2018 historical stock prices for all companies in the S&P 500 index. The table is 29.58 MB, contains 7 attributes, namely date, open, high, low, close, volume, and ticker name, and has 619040 rows. The mean and standard deviation of the dataset is as follows:



	open	high	low	close	volume
count	619029.000000	619032.000000	619032.000000	619040.000000	6.190400e+05
mean	83.023334	83.778311	82.256096	83.043763	4.321823e+06
std	97.378769	98.207519	96.507421	97.389748	8.693610e+06
min	1.620000	1.690000	1.500000	1.590000	0.000000e+00
25%	40.220000	40.620000	39.830000	40.245000	1.070320e+06
50%	62.590000	63.150000	62.020000	62.620000	2.082094e+06
75%	94.370000	95.180000	93.540000	94.410000	4.284509e+06
max	2044.000000	2067.990000	2035.110000	2049.000000	6.182376e+08

4. **Potential queries:**
    a. S&P 500 Stock Data Queries:
        i. Query to pull individual stock performance data based on user input. This could be accomplished with a search functionality or a page that contains all the different stocks, and upon a user selecting a stock, they are provided with the stock's performance over the 5 year time period.
        ii. Query to compare and contrast the performance of selected stocks. The user will be able to select multiple stocks and then we will overlay the data in a line graph that provides the performances of the stocks over time.
        iii. Identify the top 5 stocks by percentage of the total S&P index. As we do not have an S&P500 index in either dataset, we will have to use aggregation functions to calculate the total index value and produce the stocks that are the highest percentage of the total index value.
            1. Note that we technically do not have the market capitalization of the stocks, so here we will define the percentage of the total S&P index using a price-weighted portfolio assumption to account for that fact.
        iv. Identify the top 10 stocks by percentage change in price in a selected period of time. This would allow users to identify the top 10 percentage gainers and losers over a specific period. We will calculate percentage change and sort by that metric within the selected time range.
        v. Identify the top 10 stocks by volatility in a selected period of time. This would allow users to identify the top 10 most and least volatile stocks over

a specific period. We will calculate volatility and sort by that metric within the selected time range.

b. Stock Exchange Data Queries:

   i. Query to pull individual index performance data based on user input. This could be accomplished with a search functionality or a page that contains all the different indices, and upon a user selecting an index, they are provided with the index's performance over the time period from 1965 to 2021. If the index did not begin reporting in 1965, we will simply go as far back as possible.

   ii. Query to compare and contrast the performance of selected indices. The user will be able to select multiple indices and then we will overlay the data in a line graph that provides the performances of the indices over time (remaining again cognizant of the fact that some indices may not have begun reporting at the same time).

c. Queries that make use of both datasets:

   i. Query to compare and contrast the performance of selected S&P 500 stocks with selected indices. The user will be able to select multiple stocks and indices and overlay the data in a line graph that provides the performances of them over time. The join will be performed based on the same date. We would also make use of subqueries to isolate the specific time period of the index.

   ii. Query to take the mean of selected stocks and compare with selected indices to see how a batch of stocks have performed in comparison to the overall market performance of a particular country. The aggregation will be performed on the selected stocks and join based on the same date.

   iii. Similarly, in producing several KPIs for stocks and indices (for example stock/index volatility, percentage change), we will be making use of complex mathematics that require aggregation functions and pull data from both datasets.

   iv. Query to get the beta of a particular stock relative to an index in a given time period in one of two ways: (1) dividing the covariance of stock returns and index returns by the variance of stock returns and index returns or (2) determining the slope of the returns data of the stock vs the index. The second method will be simpler to implement. This will involve joining stock data with index data based on dates, calculating percent change of close and open prices for each date in the period we are analyzing, as well as performing the slope calculation.