ECMM445 Learning from Data:

Fortune 500 Data Analysis

Introduction:

A. The main goal of the analysis states that the model will be primarily utilized for predicting the profitability of Fortune 500 companies. I'll be identifying the variables that have a strong impact on companies' profits using two distinct regression models.

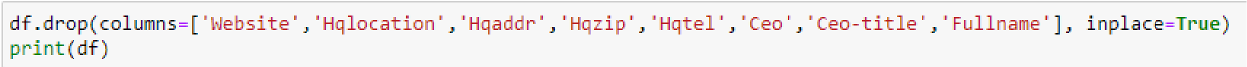
B. This report analyses the performance of companies in each industry as well as the performance of various sectors. This will help the companies to understand their shortcomings and then build strategies to increase their profits which will eventually strengthen their position among the fortune 500 companies. We will predict the profits of the companies so that assets and equity investments can be planned accordingly

Methodology and Dataset:

C. The dataset I selected was the profits data of Fortune 500 companies from 2017. There are 23 columns that include all the information. The dataset contained 23 columns which were narrowed down to 8 after analysis. I was able to understand company's revenue generation, amount invested in assets and equity, and profit made from the revenue thanks to the concepts of Revenues, Profits, Assets, and Revchange, Prftchange, and Totshequity columns. In addition to these columns, rank and the total number of employees aided me in determining whether or not larger firms are always more profitable due to a surplus of manpower.

D. Data Cleaning:

- There are no duplicate entries and null columns in this dataset.
- The single outlier in the dataset is for "Walmart," which has higher values for some indexes, including revenue and workers. Despite being an outlier, this is still useful information. In some future situations, it could be necessary to temporarily disregard this entry in order to avoid significantly affecting the results or visualization.
- A new dataframe is created which removes the unwanted columns.

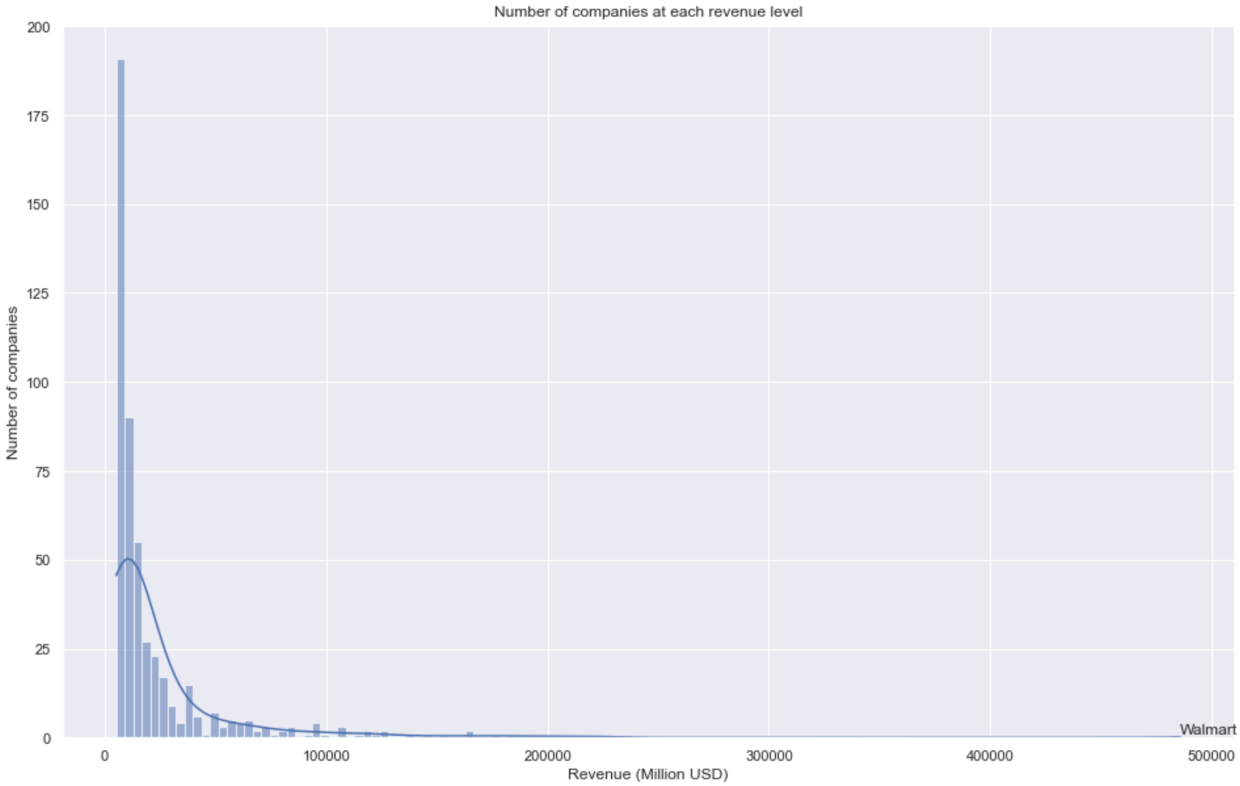


```
df.drop(columns=['Website','Hqlocation','Hqaddr','Hqzip','Hqtel','Ceo','Ceo-title','Fullname'], inplace=True)
print(df)
```

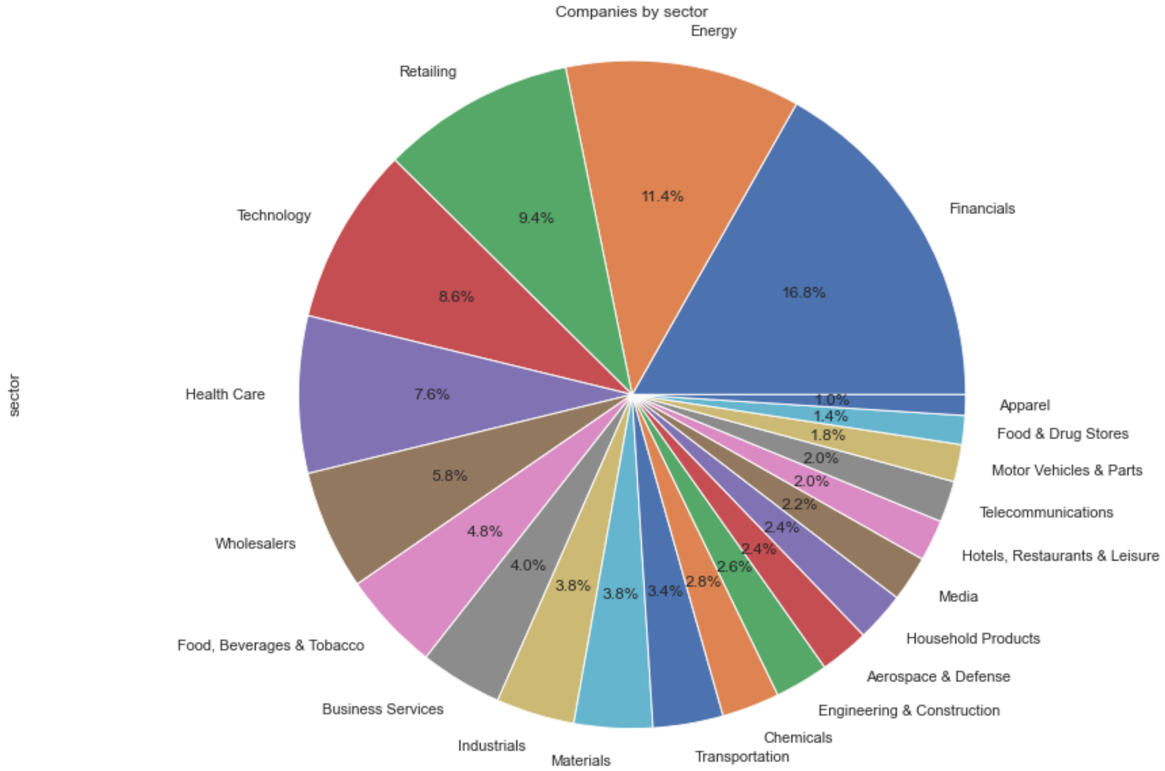
- I started scaling my data after eliminating 18 columns which is a crucial step before data pre-processing. It enables us to quickly and effectively train and test machine learning models by bringing the values of all the dataset's features closer together in terms of their values. The values of my columns needed to be normalized.

Data Exploration:

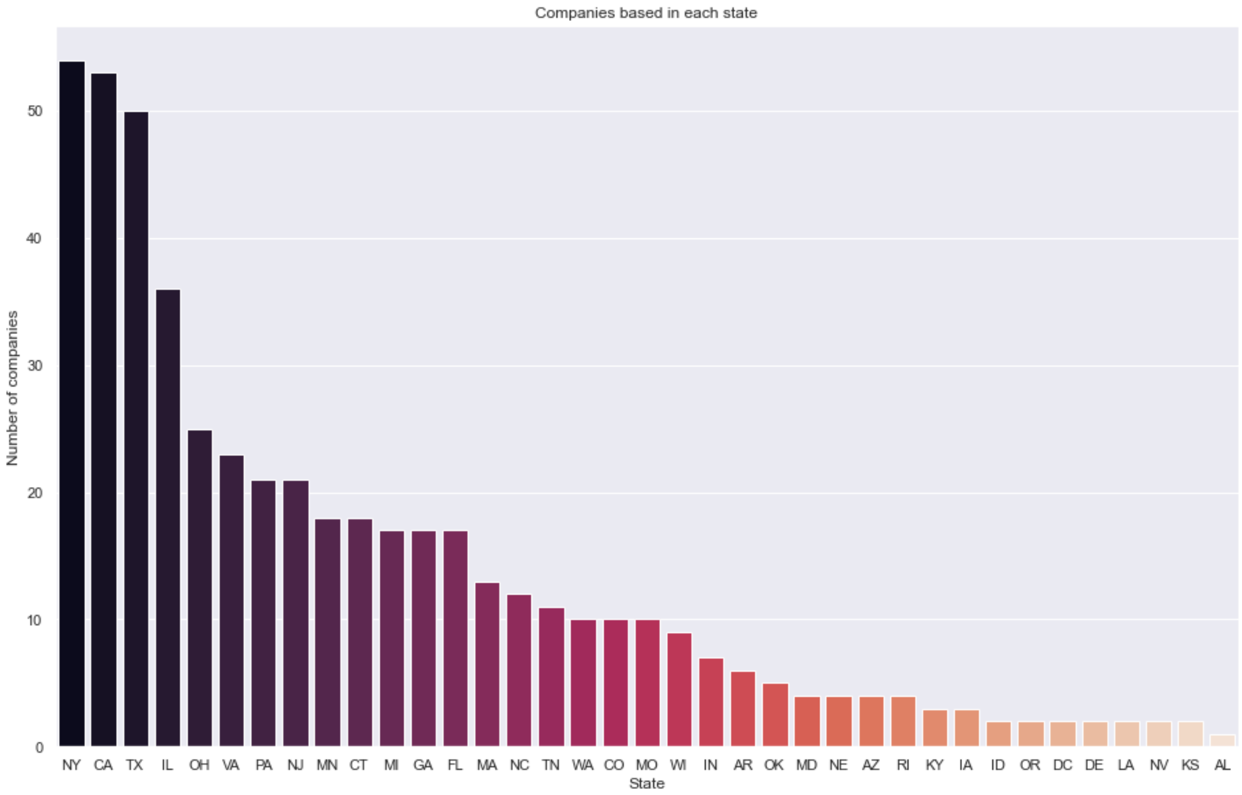
- The below histogram depicts the revenue data with bars split by sector the company is involved with. The first-ranked company, Walmart, had a yearly revenue of 485,873 million USD, compared to second-place Berkshire Hathaway's 223,604 million USD, a difference of 117.3% rounded to four significant numbers.



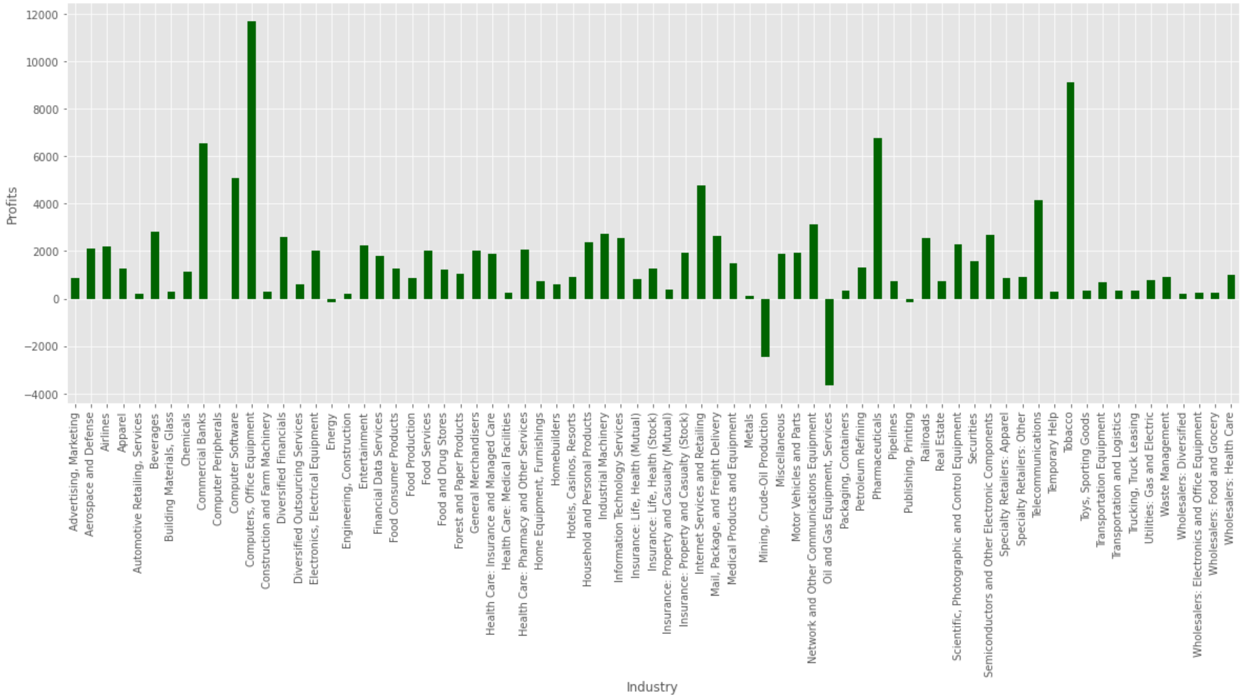
- The pie chart shows companies by sector. Financials, Energy, Retailing, Technology, Health Care, and Wholesalers are the most prevalent sectors, with only those exceeding 5% of the dataset.



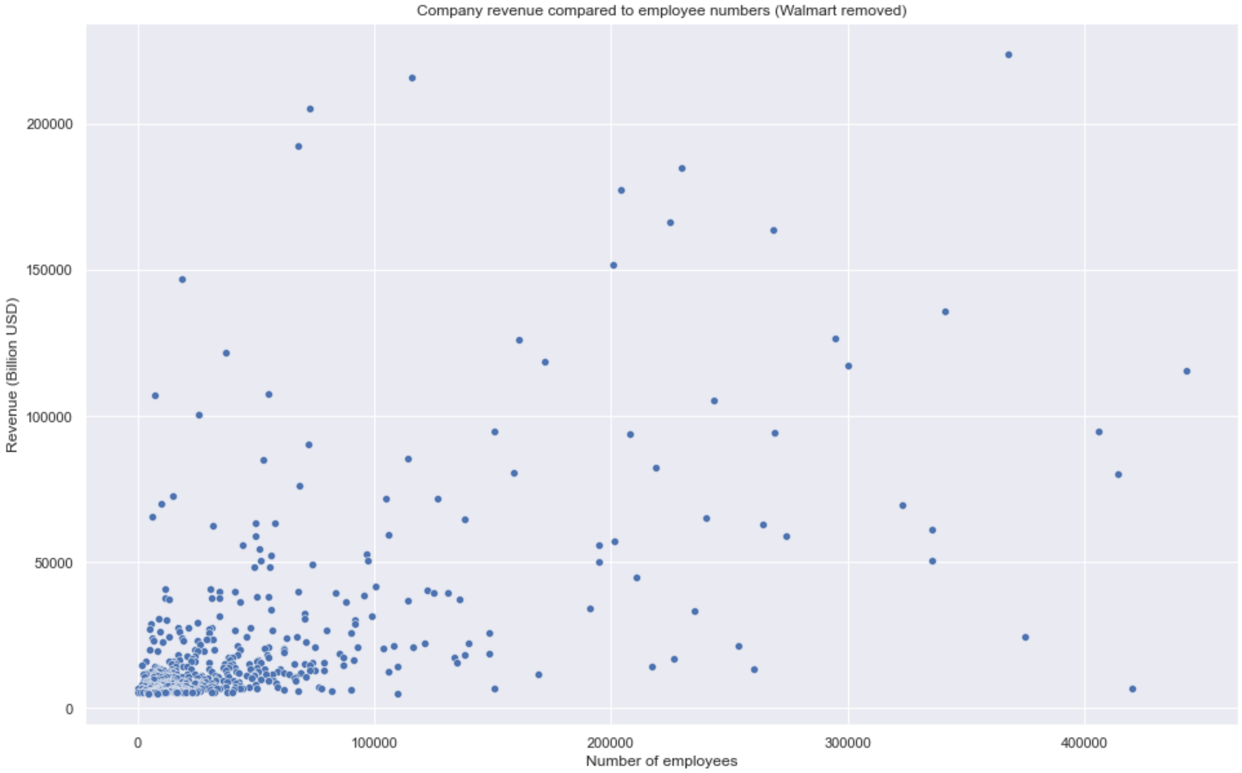
- The upcoming exploration is comparing the number of companies located in each state using seaborn countplot bar graph. New York is the most popular state for corporate headquarters, followed by California with Alabama being the least popular.



- From the graph we can see that the computer's, office equipment industry has generated maximum profit while mining, crude-oil production and oil and gas equipment, services industry were in losses. Majority of the industries generated profits around 2000 million USD.

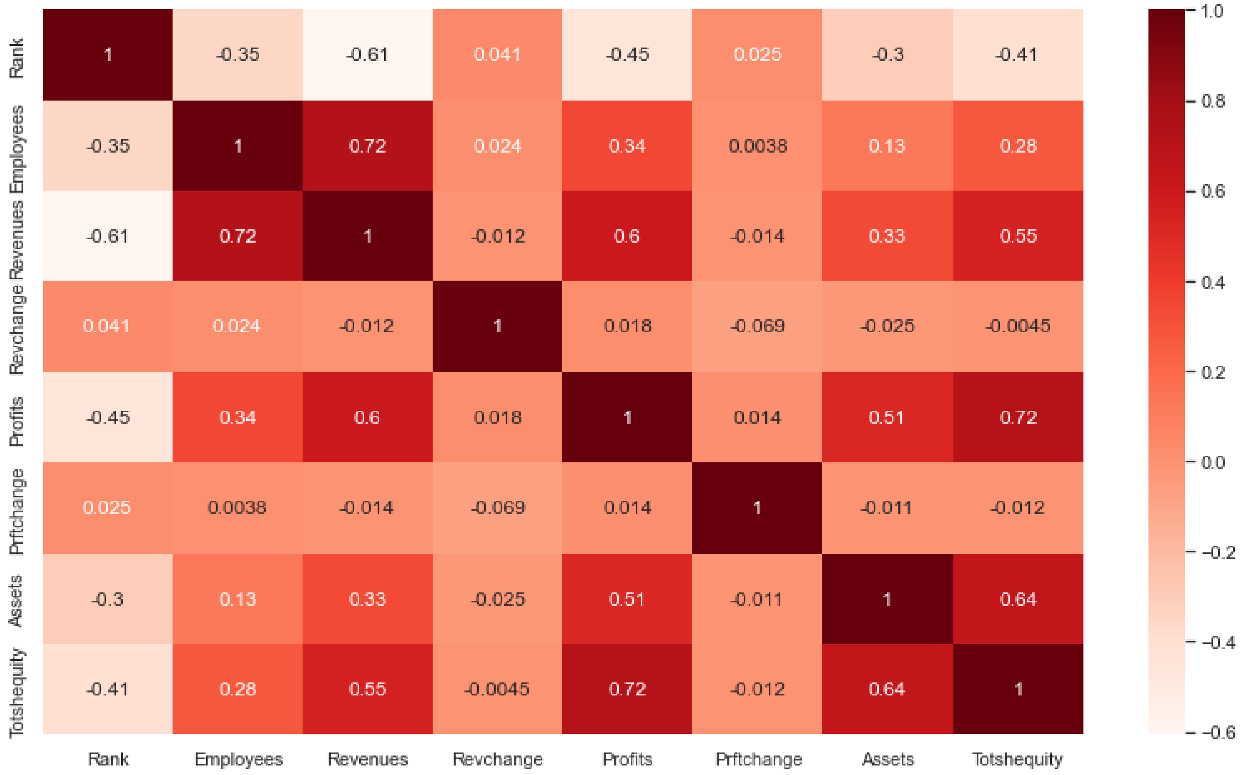


- The following is a scatter plot that examined the relationship between a company's employee count and its revenue.



Results:

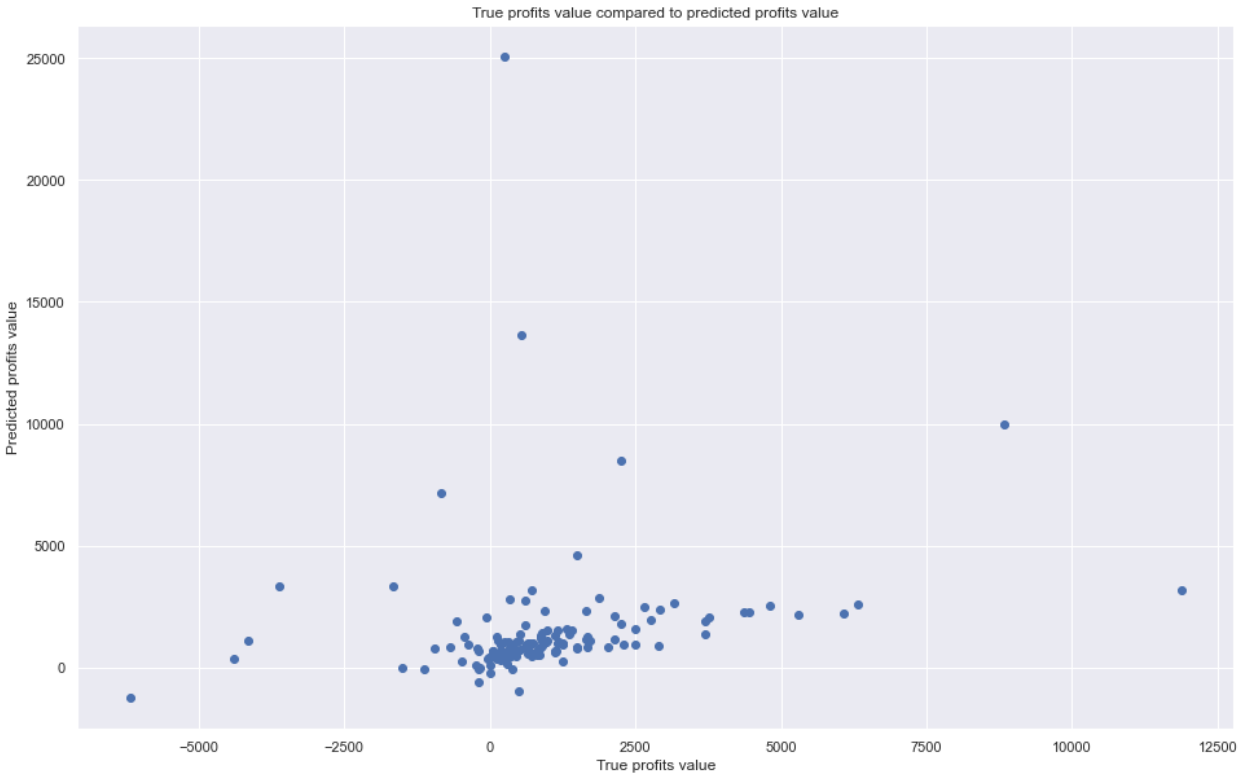
E. I decided to run regression on this dataset to use as a prediction; hence, the variables I was going to use is 'x' and the variable I was going to attempt and forecast as 'y'. For 'y', I decided on profits and for ‘x’, the heatmap's numerical values—employees, revenues, profits, price change, assets, revchange, and totshequity will be used.



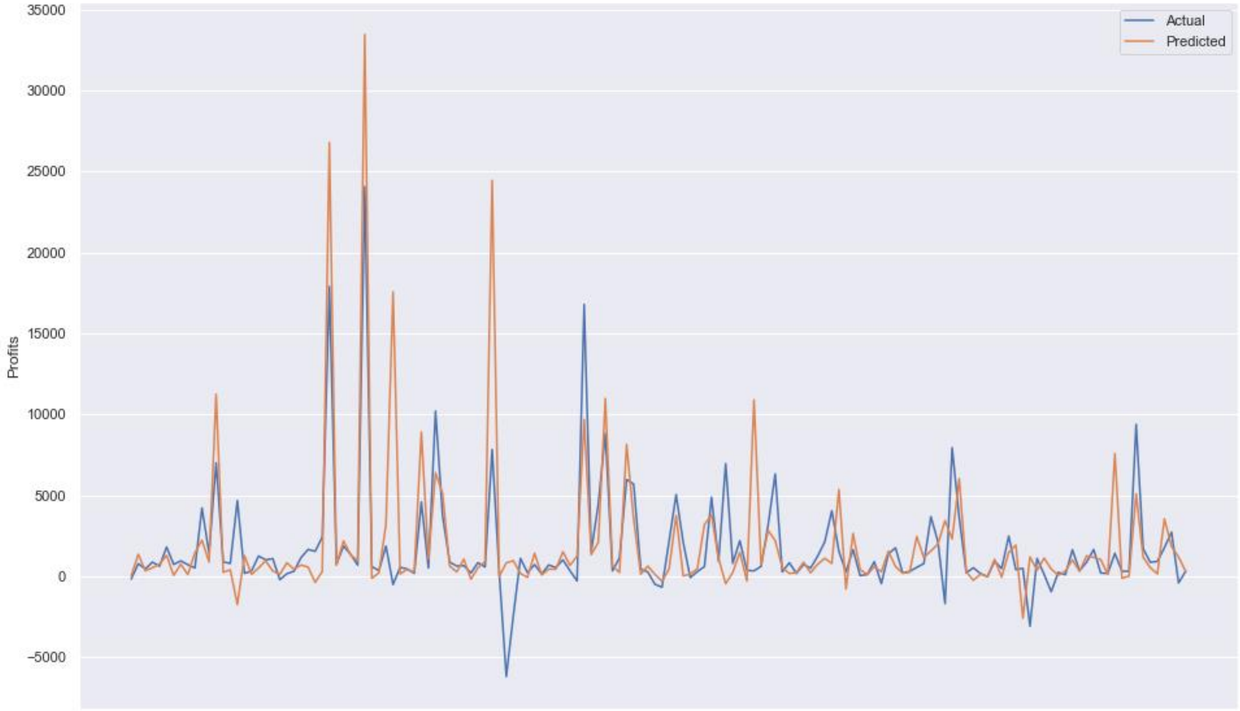
- In linear regression models, a straight line is used to represent the relationship between variables. Multiple linear regression tries to describe the relationship between multiple explanatory variables and a response variable. Profits, in this scenario, would be an example of a continuous dependent variable.

Multiple Linear Regression:

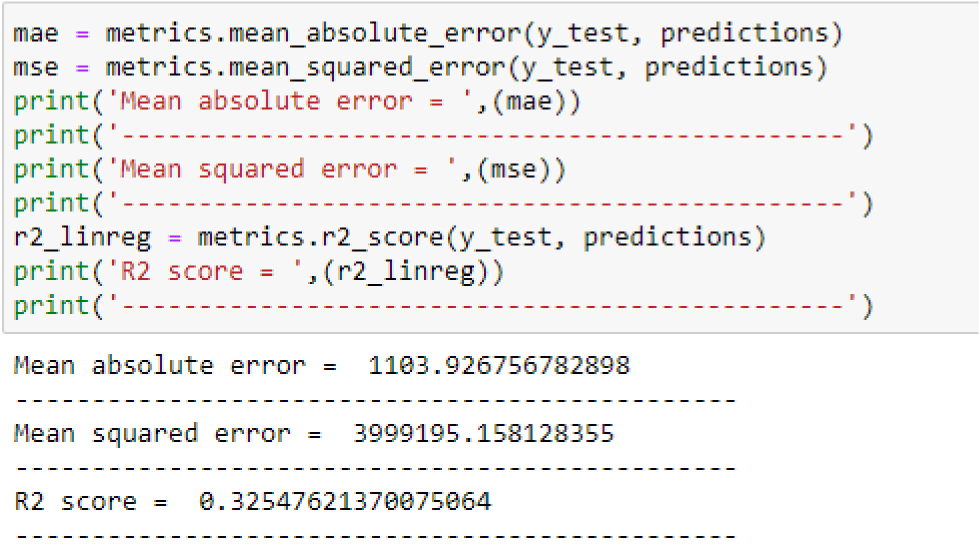
- For the regression model, data must be divided into two sets: training and test data. The former is used to fit the regression model, and the latter is used to test the model's fit. To accomplish this, I utilized the scikit-learn library ,train test split with a test size of 0.3, which designates that 30% of the data would be used for testing and the remaining 70% for refining the regression's fit. Utilizing the fit() and linear regression() functions, the fit is then implemented. The predict() method can then be used with the test data for the independent variables to forecast the values of the dependent variable.



- The results of the predictions compared to the true values are depicted in the figure below, which demonstrates that while the predicted value is frequently close to the actual value, there are some values with less accurate predictions and a few with extremely inaccurate predictions.



- The mean absolute error (MAE) and the mean squared error (MSE) are two methods to calculate the prediction error and determine the accuracy of the predictions of regression fit. MAE is non-differentiable and is not very sensitive to outliers unlike MSE. MAE measures the absolute difference between the true and predicted values while MSE measures the variation in predictions to true values.

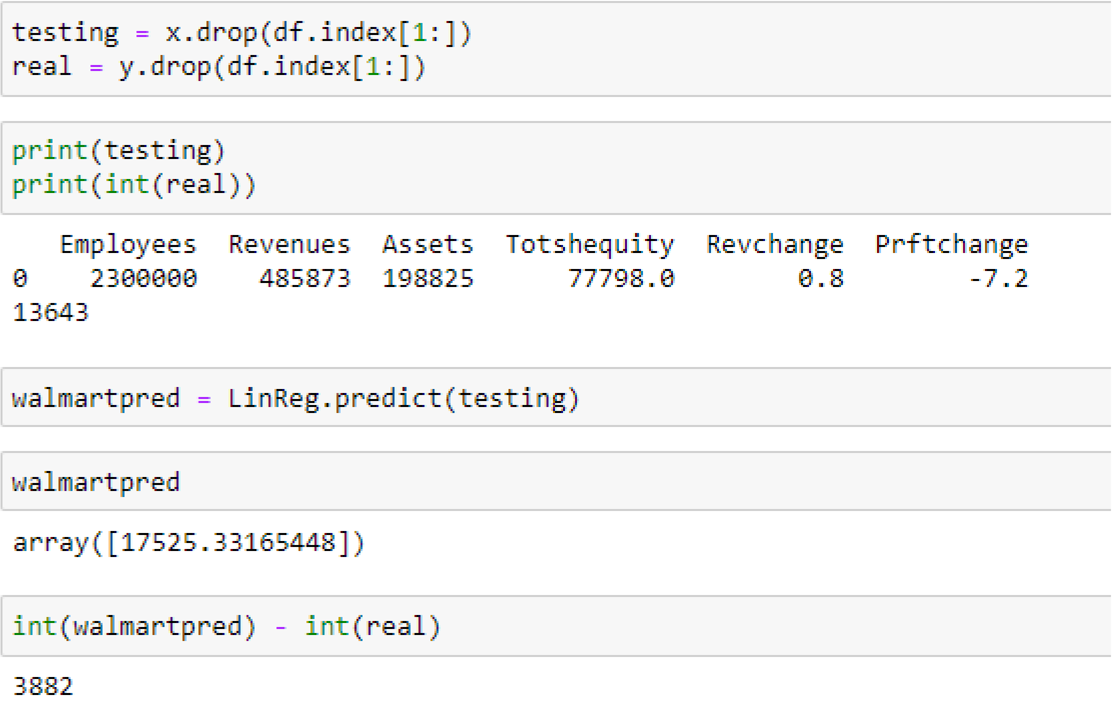


```
mae = metrics.mean_absolute_error(y_test, predictions)
mse = metrics.mean_squared_error(y_test, predictions)
print('Mean absolute error = ',(mae))
print('-----------------------------------------------')
print('Mean squared error = ',(mse))
print('-----------------------------------------------')
r2_linreg = metrics.r2_score(y_test, predictions)
print('R2 score = ',(r2_linreg))
print('-----------------------------------------------')
```

```
Mean absolute error =  1103.926756782898
-----------------------------------------------
Mean squared error =  3999195.158128355
-----------------------------------------------
R2 score =  0.32547621370075064
-----------------------------------------------
```

- The average difference between the predicted and actual profit values, as determined by the MAE value, fluctuated between 1000 and 1600 (Million USD) with each execution. The large MSE indicates a large variance in the findings which could be the result of the few incorrect predictions. The R2 score was 0.3336, which indicates that 33.37% of the data matched the regression. After several executions, the lowest and greatest R2 scores observed were 0.3136 and 0.7998. The randomly chosen division of the training and testing data is to blame for the variation in measures.

- The dataset's first item, Walmart, was used to illustrate how the multiple linear regression fit could be used to predict the profit values. The model predicted a profit of 17525 million USD, but the actual value was 13643 million USD, resulting in a prediction error of 3882 million USD.



```
testing = x.drop(df.index[1:])
real = y.drop(df.index[1:])
```

```
print(testing)
print(int(real))
```

```
   Employees  Revenues  Assets  Totshequity  Revchange  Prftchange
0    2300000    485873  198825      77798.0        0.8        -7.2
13643
```

```
walmartpred = LinReg.predict(testing)
```

```
walmartpred
```

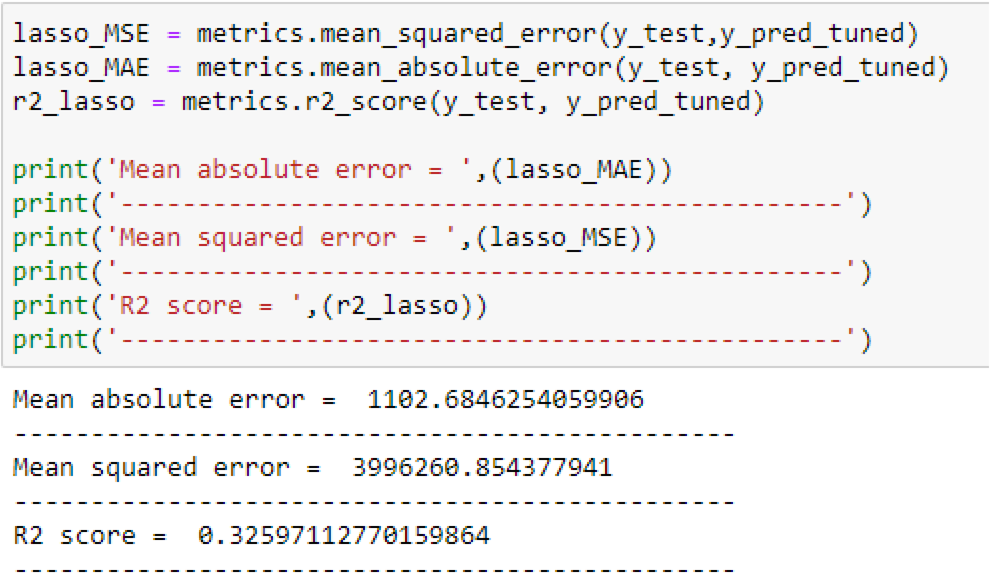
```
array([17525.33165448])
```

```
int(walmartpred) - int(real)
```

```
3882
```

<u>LASSO Regression:</u>

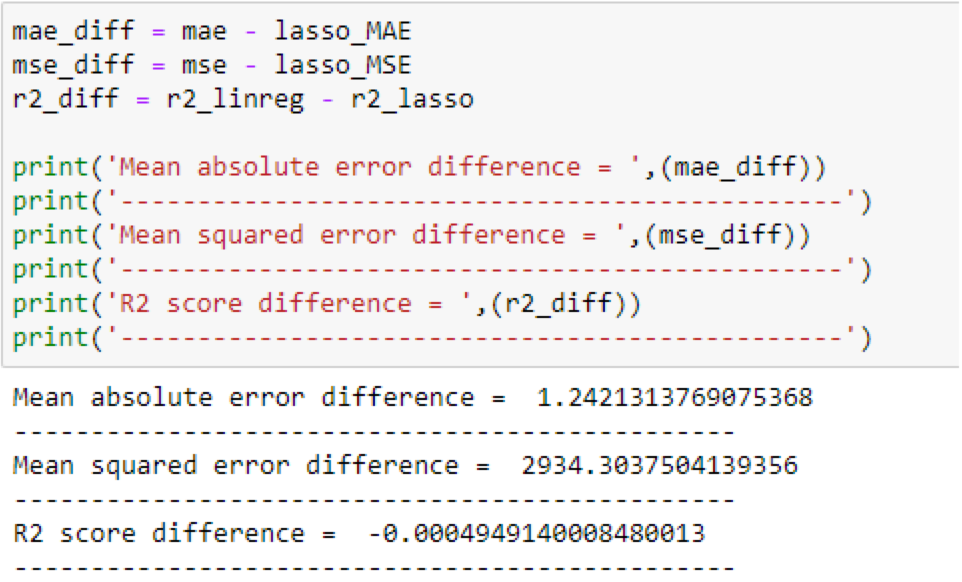
- Least Absolute Shrinkage and Selection Operator, or LASSO, is a statistical formula for data regularisation and variable selection. By using L1 regularisation, Lasso imposes a penalty equivalent to the magnitude of the coefficients in absolute terms. This regularisation may lead to models with few coefficients as some may become zero throughout the process and then removed from the model.
- To tune this model, I used the scikit-learn LassoCV function to get the ideal lambda value. This function uses cross validation to apply iterative fitting along a regularisation path. The lambda value changes because random data is split into training and test data, but I discovered that the value is close to 1021. The lasso model is used to predict the values of test data using this optimal value.



```
lasso_MSE = metrics.mean_squared_error(y_test,y_pred_tuned)
lasso_MAE = metrics.mean_absolute_error(y_test, y_pred_tuned)
r2_lasso = metrics.r2_score(y_test, y_pred_tuned)

print('Mean absolute error = ',(lasso_MAE))
print('-----------------------------------------------')
print('Mean squared error = ',(lasso_MSE))
print('-----------------------------------------------')
print('R2 score = ',(r2_lasso))
print('-----------------------------------------------')
```

```
Mean absolute error =  1102.6846254059906
-----------------------------------------------
Mean squared error =  3996260.854377941
-----------------------------------------------
R2 score =  0.32597112770159864
-----------------------------------------------
```

F. I advise using the LASSO regression model as the ultimate model. The bulk of the iterations of the code offered lasso regression to have slightly lower MAE and MSE and slightly higher R2 score, which are all positives. In contrast to multiple linear regression, this higher accuracy represents a relatively tiny improvement.



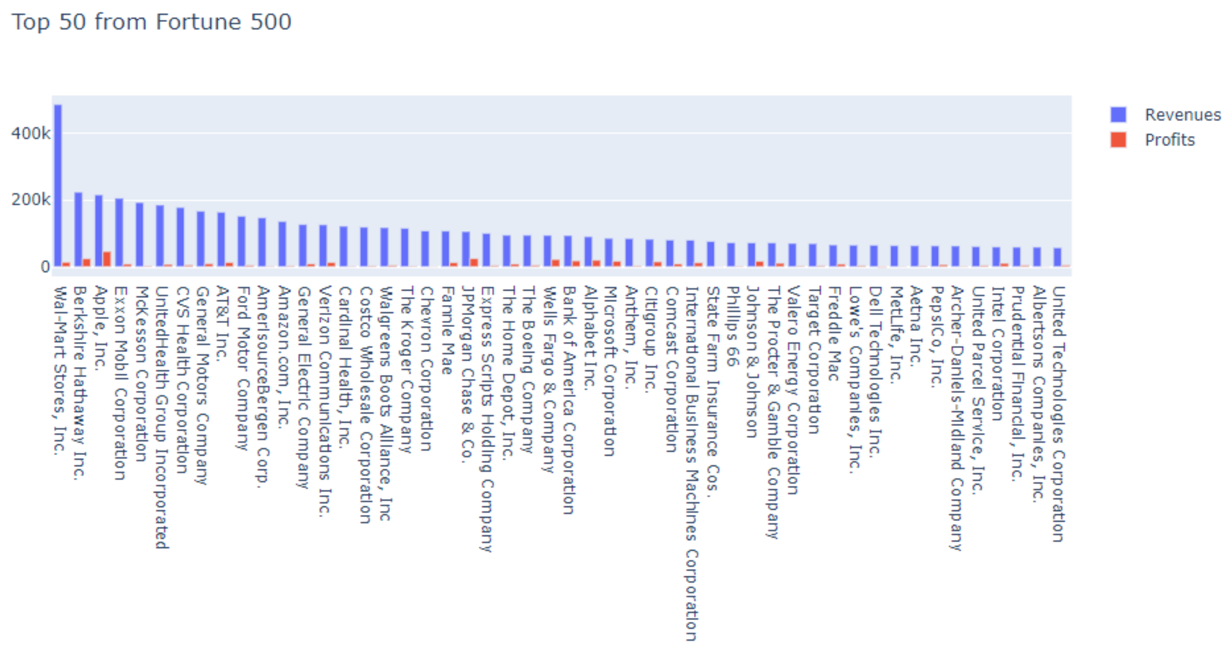
```
mae_diff = mae - lasso_MAE
mse_diff = mse - lasso_MSE
r2_diff = r2_linreg - r2_lasso

print('Mean absolute error difference = ',(mae_diff))
print('-----------------------------------------------')
print('Mean squared error difference = ',(mse_diff))
print('-----------------------------------------------')
print('R2 score difference = ',(r2_diff))
print('-----------------------------------------------')
```

```
Mean absolute error difference =  1.2421313769075368
-----------------------------------------------
Mean squared error difference =  2934.3037504139356
-----------------------------------------------
R2 score difference =  -0.0004949140008480013
-----------------------------------------------
```

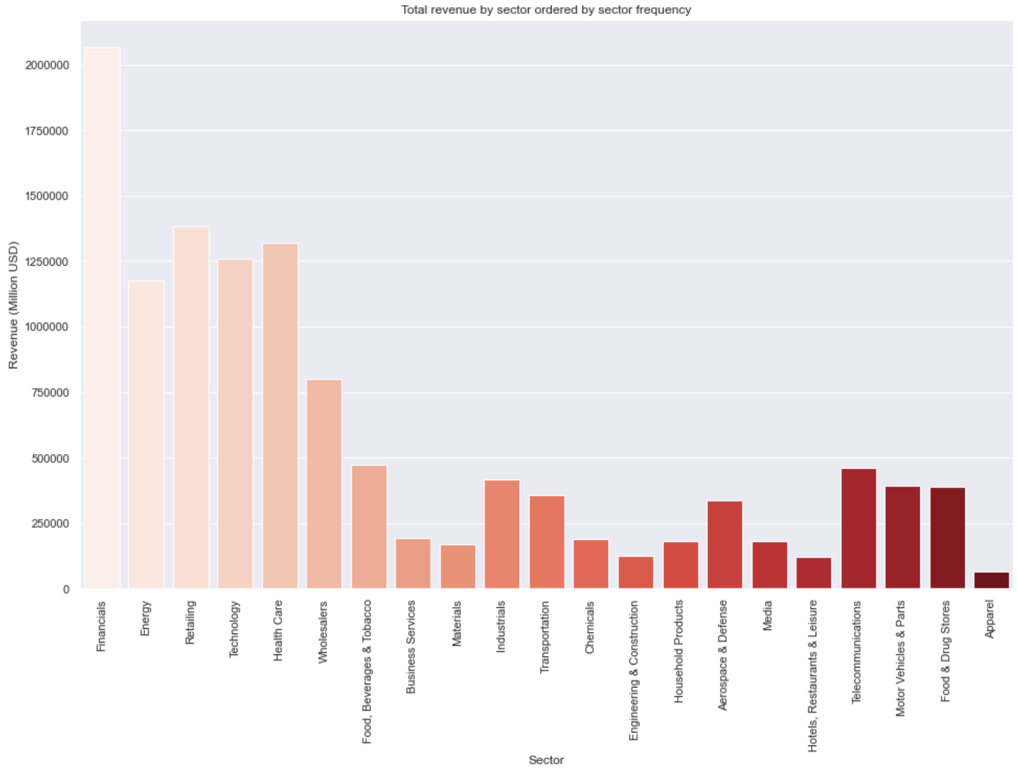
Discussion:

G. Key Findings and Data Insights:

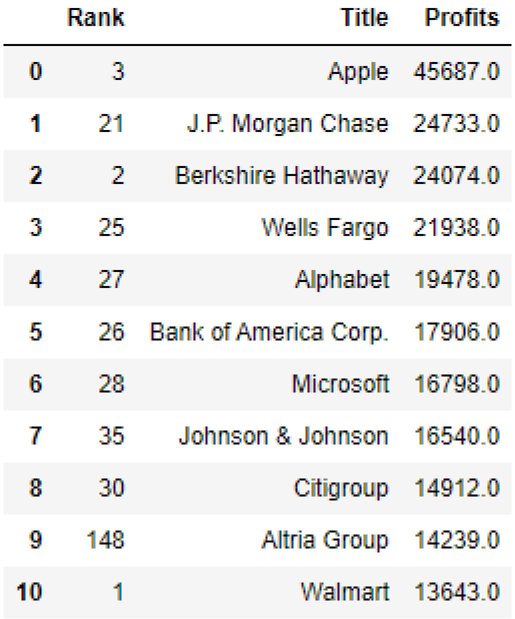
- Due to Walmart's extensive expansion and lower prices for high-quality products, their stores have consistently been in the top 5 companies since 2010.



- The financial sector has maintained its strong revenue after recovering from the 2009 recession.



- The insights from this report made it easier to comprehend that the assets, revenues, and employee are independent variables, while the other factors are reliant on each other.
- A company's profits are positively and negatively impacted by revenue and personnel count respectively. The company’s annual profits are impacted by amount of equity company possesses which are in turn influenced by assets.
- The top 10 companies with the largest earnings in 2017 were analysed, and while some of them had good profits, their rankings have fallen overall. Examples include Berkshire Hathaway, Apple, Wells Fargo, Citigroup etc.



	Rank	Title	Profits
0	3	Apple	45687.0
1	21	J.P. Morgan Chase	24733.0
2	2	Berkshire Hathaway	24074.0
3	25	Wells Fargo	21938.0
4	27	Alphabet	19478.0
5	26	Bank of America Corp.	17906.0
6	28	Microsoft	16798.0
7	35	Johnson & Johnson	16540.0
8	30	Citigroup	14912.0
9	148	Altria Group	14239.0
10	1	Walmart	13643.0

- Despite holding the #1 spot on the Fortune 500 list, Walmart is ranked 10th in terms of earnings and was unable to crack the top 10 Profit Change Companies.

H. Analysing the Data:

- Both the models had flaws while analysing the result. Since the relationship between the independent (profits) and the dependent variable (assets) was not linear, multiple linear regression, while being correct, was unable to determine the nature of the link.
- If two or more variables in LASSO have a collinear relationship then one of them is randomly chosen, which is an inaccurate strategy for data exploration.
- In my opinion, the method of stacking two or more machine models on top of one another could be the next step when analysing this data. To identify the relationship between the dependent and independent variables, we might start by doing linear regression followed by developing the primary model using multiple linear regression and assess the precision of our predictions. Finally, LASSO is a useful technique that may be used to enhance the functionality of regression models.
- Further, a dataset with additional data attributes might make it easier to analyse the positive correlation between the values. Adding company_foundation_date, rank_company_state, company_initial _profit columns would help us in analyzing the difference between the profits and the growth of the company.

References:

1. (January 2018) A Complete Tutorial on Ridge and Lasso Regression in Python by Aarshay Jain.
   https://www.analyticsvidhya.com/blog/2016/01/ridge-lasso-regression-python-complete-tutorial/

2. (May 7, 2021) Multiple Linear Regression Implementation in Python by Harshita Yadav.
   https://medium.com/machine-learning-with-python/multiple-linear-regression-implementation-in-python-2de9b303fc0c

3. Visualization with Seaborn.
   https://jakevdp.github.io/PythonDataScienceHandbook/04.14-visualization-with-seaborn.html

4. (Aug 2, 2017) Ensemble Methods in Machine Learning: What are They and Why Use Them? By Evan Lutins.
   https://towardsdatascience.com/ensemble-methods-in-machine-learning-what-are-they-and-why-use-them-68ec3f9fef5f

5. Fortune Global 500
   https://en.wikipedia.org/wiki/Fortune_Global_500