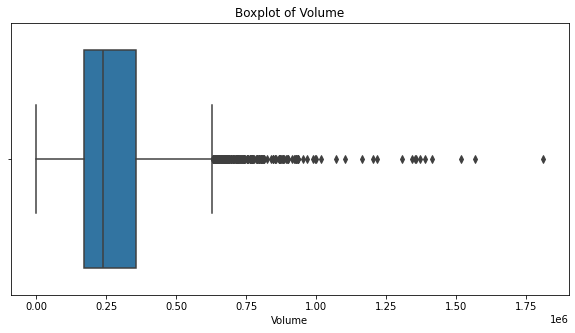
**Comprehensive Stock Analysis and Prediction Task**

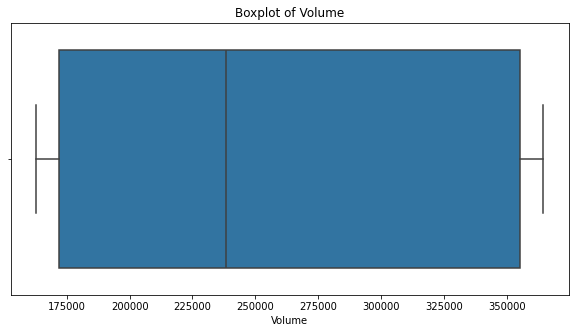
Objective: As a Data Scientist, your task is to develop a Python program that takes a CSV file containing historical stock data as input, conducts a comprehensive analysis of the stock, and predicts its future performance

**Deliverables:**

**1. Data Validation and Cleaning**

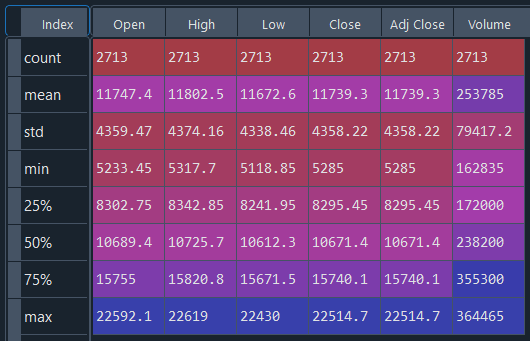
* Missing Values:
  + Data contain 14 Null values in "Open", "High", "Low", "Close", "Adj Close", "Volume" Columns.
  + Using SimpleImputer, Imputed the Null values.
* Outliers:
  + Column Volume contains Outliers.
  + Winsorizer (capping methos: IQR), Outliers get treated

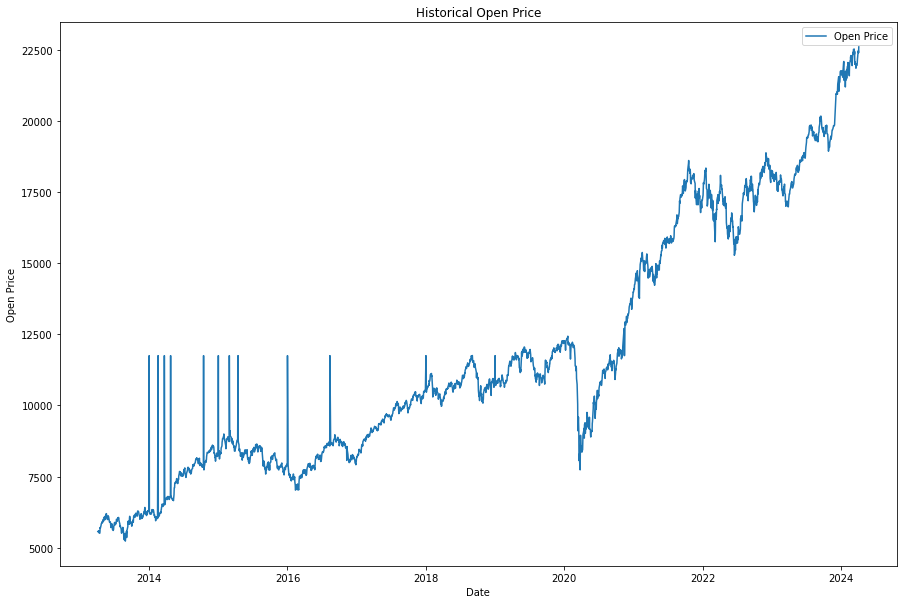
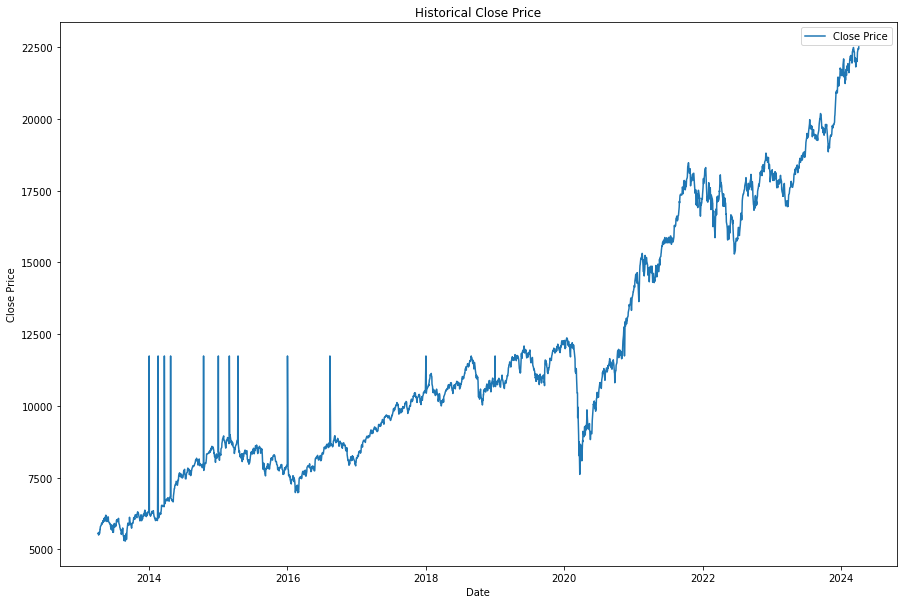


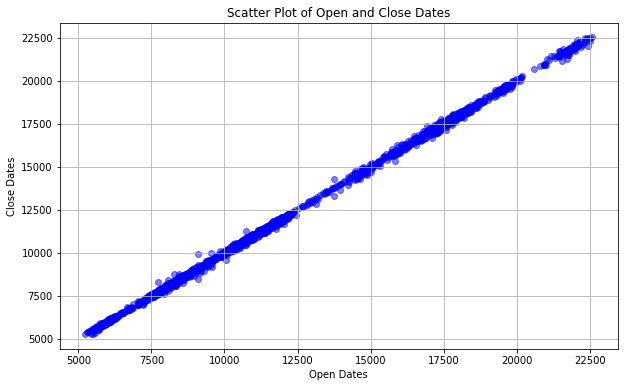
* After Removal of Outliers

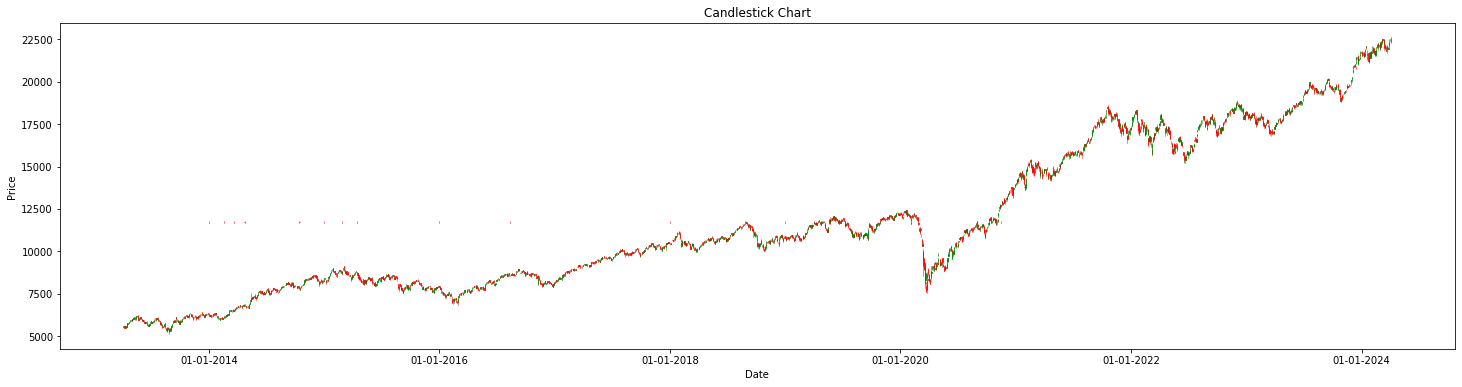
2. Stock Analysis:

* Mean, std, min, max of dataset.

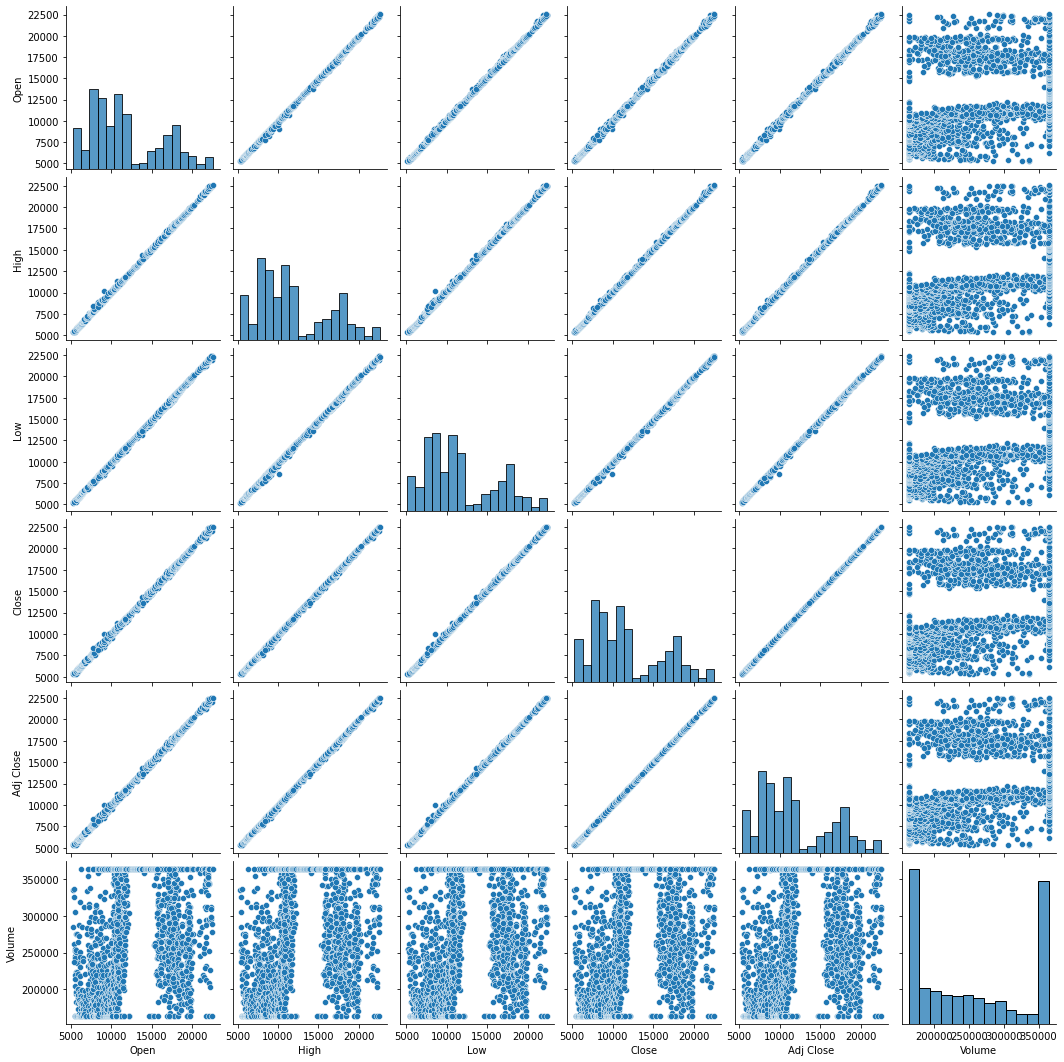


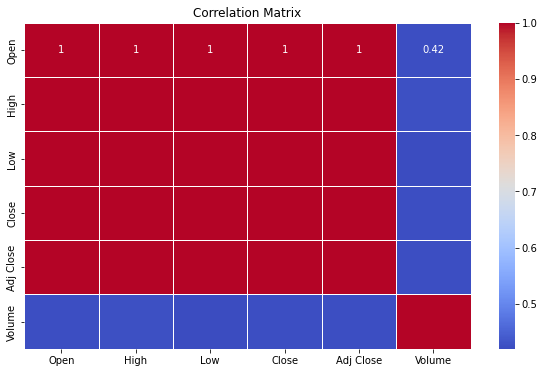
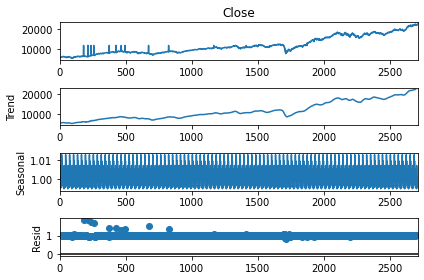
* Visualization of Historical trend of Open Price.
* There is **Sharp divergence** from 2014 to 2017.
* Visualization of Historical trend of Close Price.
* There is **Sharp divergence** from 2014 to 2017

* Visualization of correlation between Open and Close price.
* There is a **Strong Positive Correlation** between Open and Close price.
* Candlestick plot



* Pair plot: From Pair plot we can see there is a **high Correlation** among all values expect Volume.



* Heatmap for correlation: From Heatmap we can see there is a correlation of 1 from every column.
* Seasonal Decompose:
  + There are some **seasonal trends** are present in the stock and **there Up trend in stock.**

RMSE and MAE Values:

1. LinearRegression:

RMSE: 3033.270891403247

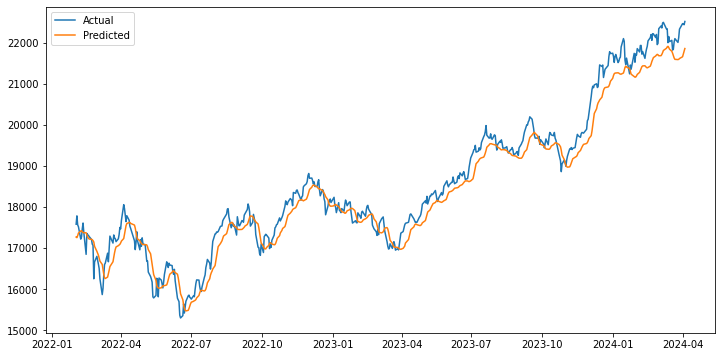
MAE: 2784.8473675297

2. ARIMA model

RMSE: 2540.1653955259903

MSE: 6452440.236627712

3. LSTM Model:

 RMSE: 357.49761651411075

Deployment and Monitoring:

1. Deploy the model as a web service using a framework such as Flask or Django.

2. Use Docker to containerize the application for consistent deployment across environments.

3. Implement a CI/CD pipeline for automated testing and deployment.

4. Use cloud services (e.g., AWS, GCP, Azure) to host the application.

5. Set up logging and monitoring tools (e.g., Prometheus, Grafana) to track model performance and usage.

1. Monitor model predictions against actual stock prices to assess accuracy.

2. Track key performance metrics (e.g., RMSE, MAE) over time.

3. Set up alerts for performance degradation or anomalies in predictions.

4. Implement feedback loops to retrain the model with new data periodically.

Prepared By:  
  
Akshaya Acharya

Junior Data Scientist

AiSPRY. Bangalore.

Mob: 8088858002

Email: [akshaya0500@gmail.com](mailto:akshaya0500@gmail.com)