EX.No.8 **ARIMA MODEL FOR TIME SERIES FORECASTING**

Date:17.04.25

AIM:

Implement a program for an ARIMA model for time series forecasting

Procedure and Code:

Step1:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from statsmodels.tsa.arima.model import ARIMA
from sklearn.metrics import mean_squared_error
```

Step2:

```
file_path = '/mnt/data/GOOGL.csv'  # Change this path if needed
df = pd.read_csv(file_path)
df['Date'] = pd.to_datetime(df['Date'])
df.set_index('Date', inplace=True)
data = df['Close'].dropna()
```

Step3:

```
# Perform decomposition
result = seasonal_decompose(df['Passengers'], model='multiplicative', period=12)
```

Step4:

```
plt.figure(figsize=(10, 4))
```

```
        plt.plot(data, label='GOOGL Closing Price')
        plt.title('GOOGL Stock Closing Price')
        plt.xlabel('Date')
        plt.ylabel('Price')
        plt.legend()
        plt.show()
```

Step5:

```
    train_size = int(len(data) * 0.8)
    train, test = data[:train_size], data[train_size:]
```

Step6:

```
    order = (5, 1, 0)
    model = ARIMA(train, order=order)
    model_fit = model.fit()
    print(model_fit.summary())
```

Step7:

```
    forecast = model_fit.forecast(steps=len(test))
    forecast.index = test.index
    plt.figure(figsize=(10, 4))
    plt.plot(train, label='Train')
    plt.plot(test, label='Test')
    plt.plot(forecast, label='Forecast', linestyle='--')
    plt.title('ARIMA Forecast vs Actual')
```

```
plt.legend()
plt.show()
```

Step 8:

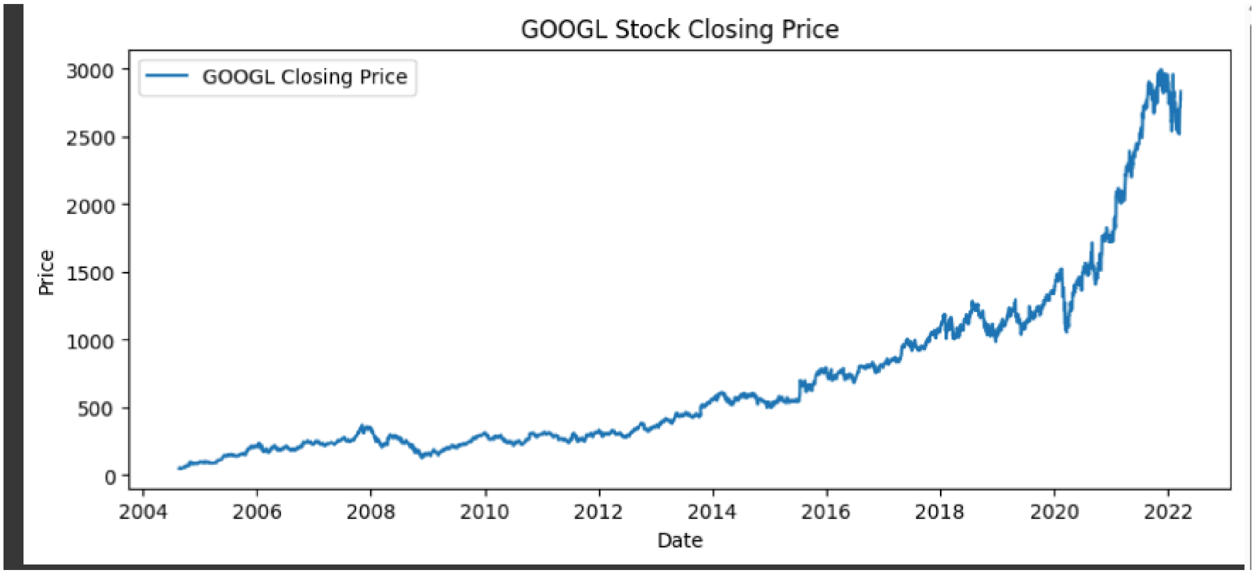
```
rmse = np.sqrt(mean_squared_error(test, forecast))
print(f'Root Mean Squared Error: {rmse:.2f}')
```

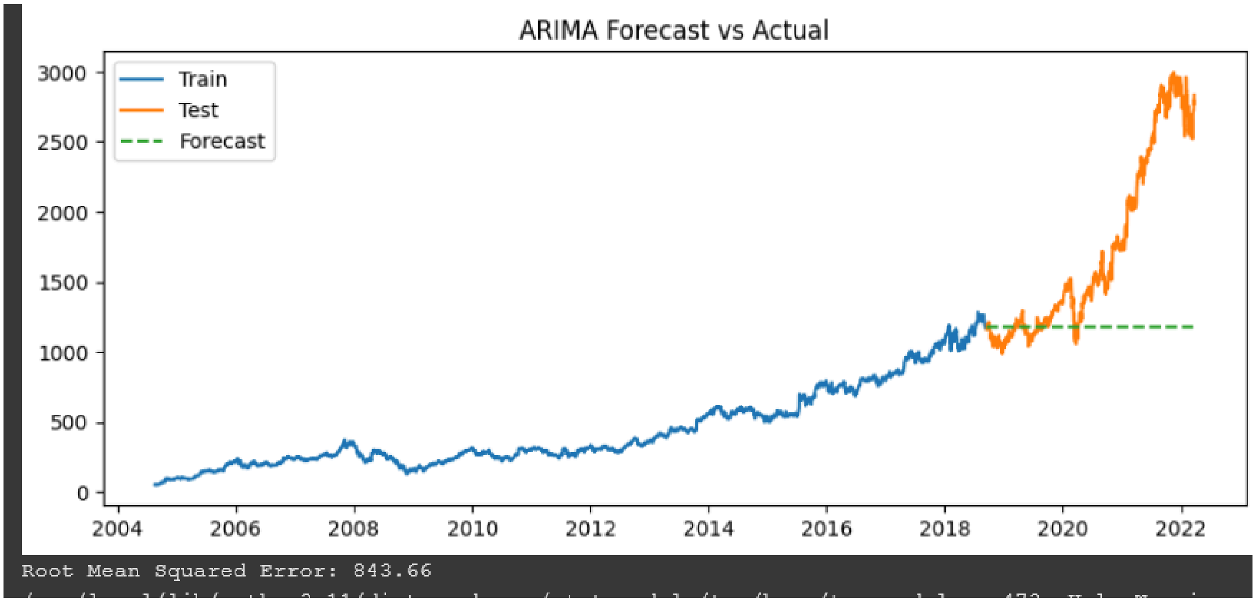
Step 9:

```
final_model = ARIMA(data, order=order).fit()
future_forecast = final_model.forecast(steps=30)
last_date = data.index[-1]
future_dates = pd.date_range(start=last_date + pd.Timedelta(days=1), periods=30, freq='B')
future_forecast.index = future_dates
```

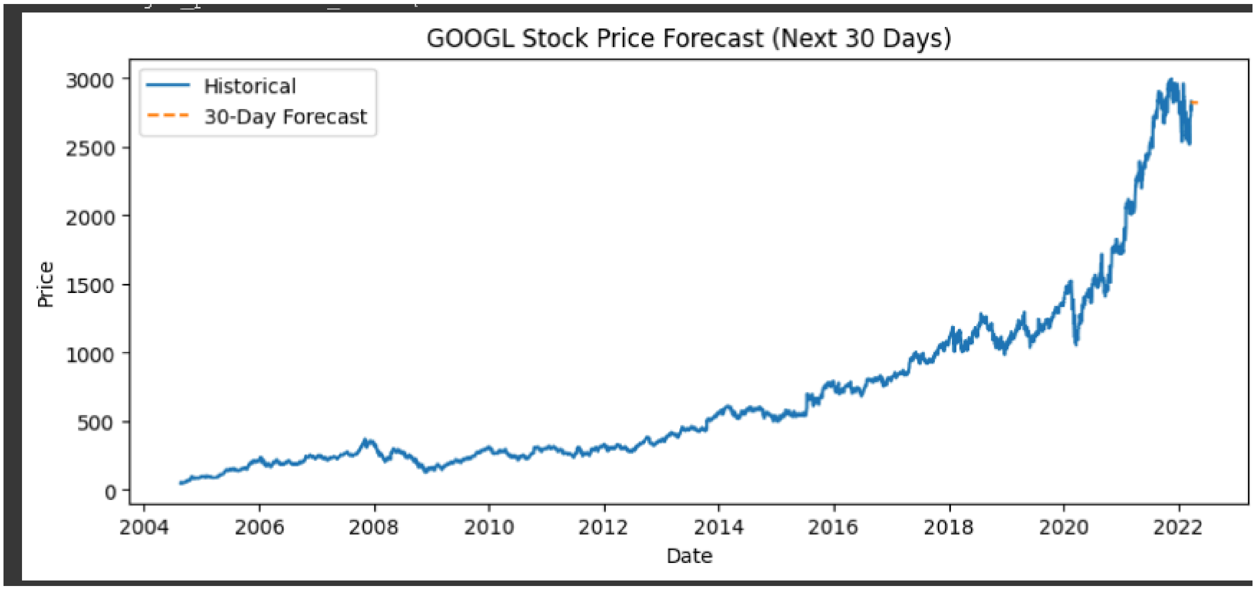
Step 10: Plot the Future Forecast

```
plt.figure(figsize=(10, 4))
plt.plot(data, label='Historical')
plt.plot(future_forecast, label='30-Day Forecast', linestyle='--')
plt.title('GOOGL Stock Price Forecast (Next 30 Days)')
plt.xlabel("Date")
plt.ylabel("Price")
plt.legend()
plt.show()
```





```
Root Mean Squared Error: 843.66
```



RESULT:

The program to execute   ARIMA model for time series forecasting. has been executed successfully.