

#### **4. Develop a linear regression model for forecasting time series data.**

<b>EX.N0 : 4</b>	<b>DEVELOP A LINEAR REGRESSION MODEL FOR FORECASTING TIME SERIES DATA</b>
<b><u>DATE : 29/03/2025</u></b>	

#### **AIM:**

To Develop a linear regression model for forecasting time series data.

#### **PROGRAM:**

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error

df = pd.read_csv(r"C:\Users\heman\OneDrive\Desktop\abdul\TSA\EX 6\Crude Oil Prices
Daily.xlsx", parse_dates=["Date"]) # Replace with actual file

df = df.sort_values(by="Date")
df["Time_Index"] = np.arange(len(df))
X = df[["Time_Index"]] # Features
y = df["Close"] # Target (gold price)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, shuffle=False)

model = LinearRegression()
model.fit(X_train, y_train)

y_pred = model.predict(X_test)

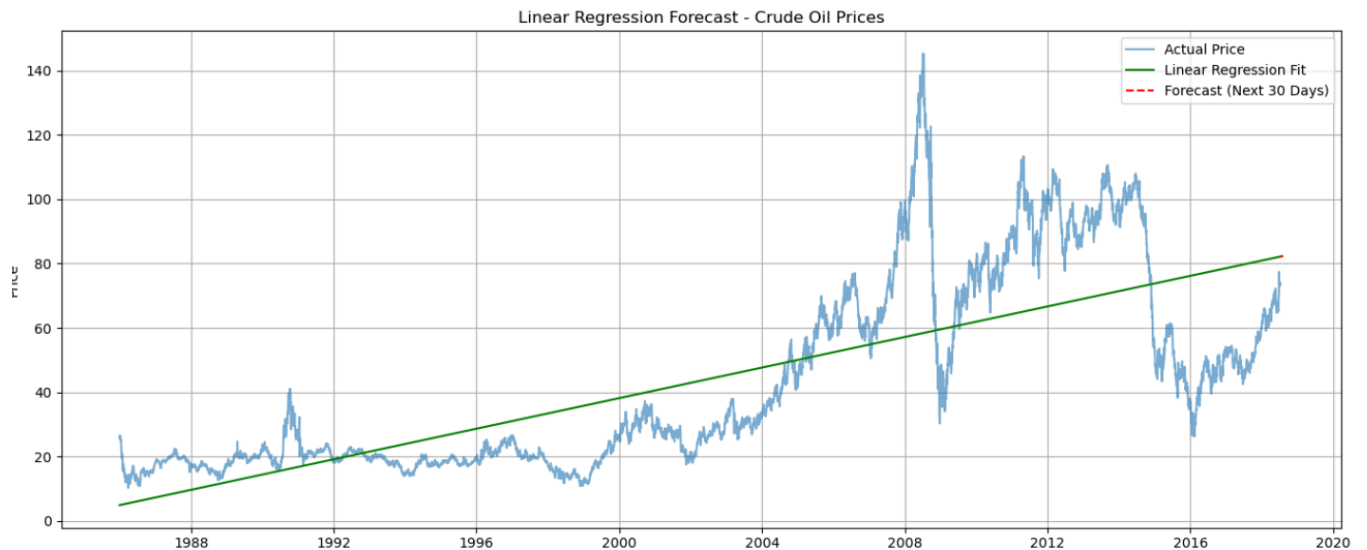
future_X = np.arange(len(df), len(df) + 30).reshape(-1, 1) # Forecast next 30 days
future_y = model.predict(future_X)

mse = mean_squared_error(y_test, y_pred)
print(f"Mean Squared Error: {mse:.2f}")

plt.figure(figsize=(12, 6))
plt.plot(df["Date"], df["Close"], label="Actual Gold Price", color="blue")
plt.plot(df.iloc[len(X_train):]["Date"], y_pred, label="Predicted Gold Price",
color="red") plt.plot(pd.date_range(df["Date"].max(), periods=30, freq='D'), future_y,
label="Forecasted Price", color="green", linestyle="dashed")
plt.xlabel("Date")
plt.ylabel("Gold Price")
plt.legend()
```

```
plt.title("Gold Price Forecasting using Linear Regression")
plt.xticks(rotation=45)
plt.show()
```

### **OUTPUT:**



### **RESULT:**

Thus, the program for Develop a linear regression model for forecasting time series data is executed successfully.