

EX:No.2

DATE:1/02/25

Implement programs for visualizing time series data.

AIM:

To Implement programs for visualizing time series data.

OBJECTIVE:

To analyze and visualize air pollution trends from 2012 to 2021 using multiple time-series plots.

BACKGROUND:

- Load, clean, and analyze **gold price data** from 2012 to 2021.
- Handle **missing values and outliers** for better accuracy.
- Identify **trends** and fluctuations in gold prices over time.
- Visualize gold price changes using **time-series plots**.
- Provide insights for **investors, analysts, and policymakers**.

SCOPE OF THE PROGRAM:

- Load and clean gold price (2012-2021).
- Handle missing values and remove outliers.
- Visualize trends using line, scatter, area, bar, and box plots.
- Identify seasonal and yearly variations.
- Analyze and compare pollution levels over time.

CODE:

```
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib.pyplot as plt
import seaborn as sns

df = pd.read_csv("/content/gold_data.csv")

print("Column names in dataset:", df.columns)

date_col = "Date" # Change if needed
gold_price_col = "Price" # Update based on actual column name

df[date_col] = pd.to_datetime(df[date_col], errors='coerce')
```

```

df = df[(df[date_col].dt.year >= 2012) & (df[date_col].dt.year <= 2021)]

df = df.sort_values(by=date_col)

fig, axes = plt.subplots(3, 2, figsize=(15, 12))
fig.suptitle("Gold Price Visualizations (2012-2021)", fontsize=16, fontweight="bold")

# Line Plot
axes[0, 0].plot(df[date_col], df[gold_price_col], color='gold', label="Gold Price")
axes[0, 0].set_title("Line Plot - Gold Price Trend")
axes[0, 0].set_xlabel("Date")
axes[0, 0].set_ylabel("Gold Price (INR)")
axes[0, 0].legend()
axes[0, 0].grid()

# Scatter Plot
axes[0, 1].scatter(df[date_col], df[gold_price_col], color='red', alpha=0.5)
axes[0, 1].set_title("Scatter Plot - Gold Price Distribution")
axes[0, 1].set_xlabel("Date")
axes[0, 1].set_ylabel("Gold Price (INR)")
axes[0, 1].grid()

# Area Plot
axes[1, 0].fill_between(df[date_col], df[gold_price_col], color="gold", alpha=0.5)
axes[1, 0].set_title("Area Plot - Cumulative Gold Price Trend")
axes[1, 0].set_xlabel("Date")
axes[1, 0].set_ylabel("Gold Price (INR)")
axes[1, 0].grid()

# Bar Chart (Monthly Average Prices)
df["Year-Month"] = df[date_col].dt.to_period("M")
monthly_avg = df.groupby("Year-Month")[gold_price_col].mean()
axes[1, 1].bar(monthly_avg.index.astype(str), monthly_avg, color='orange')
axes[1, 1].set_title("Bar Chart - Monthly Avg Gold Prices")
axes[1, 1].set_xlabel("Year-Month")
axes[1, 1].set_ylabel("Avg Gold Price (INR)")
axes[1, 1].tick_params(axis='x', rotation=90)

# Box Plot (Detect Outliers)
sns.boxplot(y=df[gold_price_col], ax=axes[2, 0], color='gold')
axes[2, 0].set_title("Box Plot - Gold Price Distribution")
axes[2, 0].set_ylabel("Gold Price (INR)")

# Hide extra subplot
axes[2, 1].axis("off")

```

```

# Adjust layout and show plot
plt.tight_layout(rect=[0, 0, 1, 0.97])
plt.show()

df = pd.read_csv("/content/gold_data.csv")

# Print column names to verify
print("Column names in dataset:", df.columns)

# Convert Date column to datetime (update the column name based on your dataset)
date_col = "Date" # Change if your date column has a different name
df[date_col] = pd.to_datetime(df[date_col], errors='coerce')

# Check if 'Gold Price' exists, otherwise update with the correct column name
gold_price_col = "Price" # Change based on actual column name from print(df.columns)

# Filter data for years (2012-2021)
df = df[(df[date_col].dt.year >= 2012) & (df[date_col].dt.year <= 2021)]

# Remove Outliers (Using IQR Method)
Q1 = df[gold_price_col].quantile(0.25)
Q3 = df[gold_price_col].quantile(0.75)
IQR = Q3 - Q1
df = df[(df[gold_price_col] >= (Q1 - 1.5 * IQR)) & (df[gold_price_col] <= (Q3 + 1.5 * IQR))]

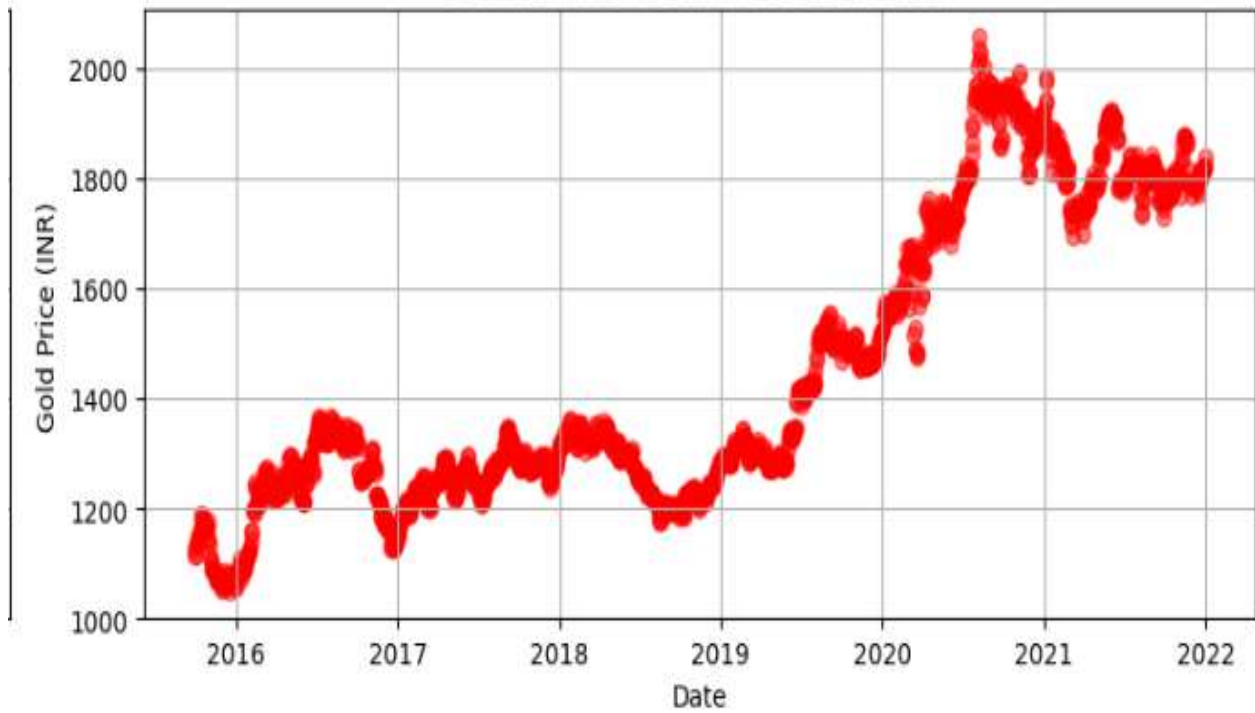
# Plot Gold Prices Over Time
plt.figure(figsize=(10, 5))
plt.plot(df[date_col], df[gold_price_col], color='gold', label="Gold Price")
plt.xlabel("Date")
plt.ylabel("Gold Price (INR)")
plt.title("Gold Price Trends in Indian Market (2012-2021)")
plt.legend()
plt.grid()
plt.show()

```

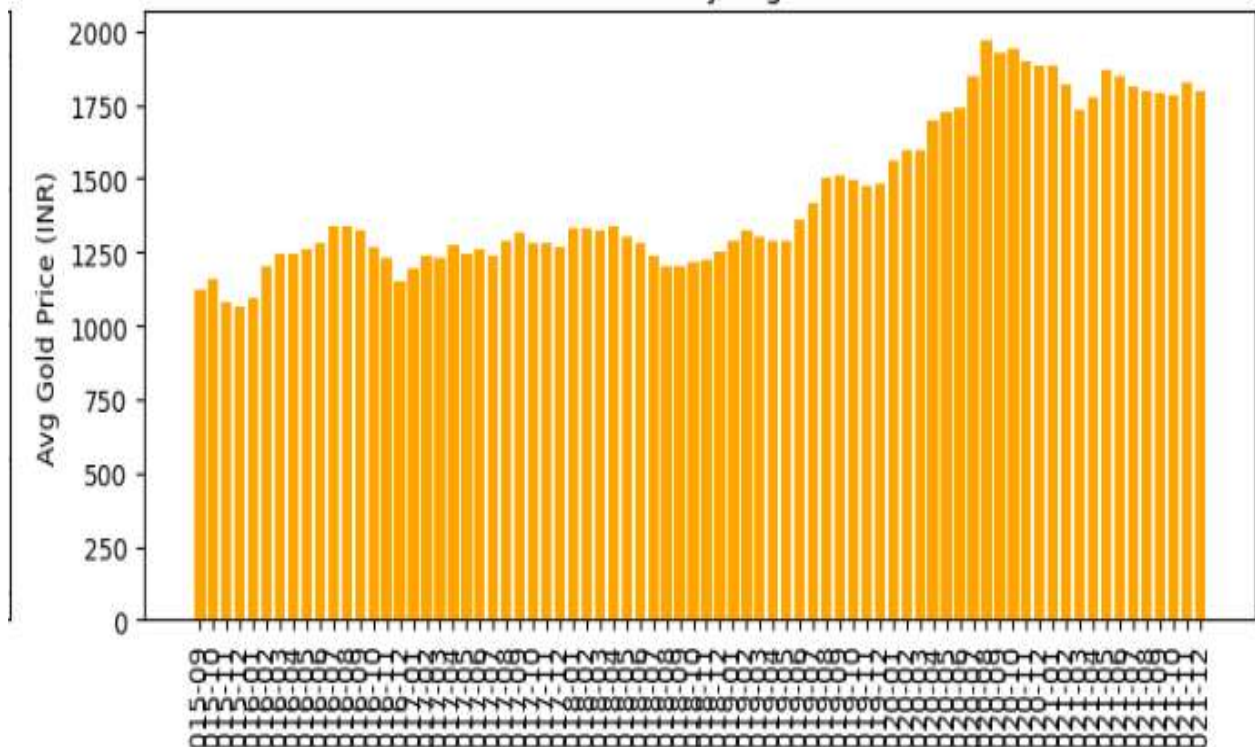
OUTPUT:

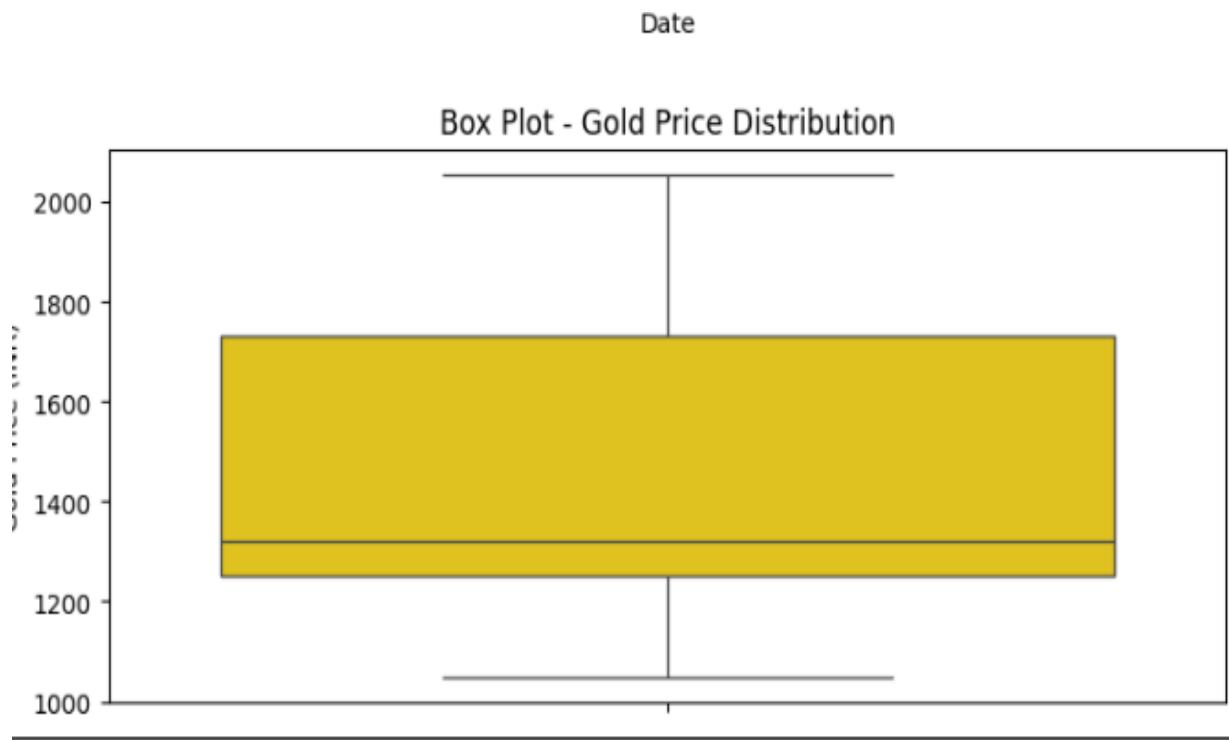


Scatter Plot - Gold Price Distribution



Bar Chart - Monthly Avg Gold Prices





RESULT:

Thus, the program using the time series data implementation has been done successfully.