EX:No.7 DATE: 08/03/25	**Implement program for decomposing time series data into trend and seasonality**

AIM:
To implement a program for decomposing time series data into its trend, seasonal, and residual components.

OBJECTIVE:
To analyze and understand the underlying structure of the Office Supply Sales time series data by breaking it down into trend and seasonality components using decomposition techniques.

BACKGROUND:
- Time series data often contains patterns that repeat over time.
- Decomposition helps in separating these components for better analysis and forecasting.
- Trend shows long-term progression.
- Seasonality captures periodic fluctuations.
- Residual reveals random noise not explained by trend or seasonality.
- Understanding these components helps improve model accuracy and insights.

SCOPEOFTHEPROGRAM:

- Load and prepare the office supply sales dataset.

- Aggregate data monthly for decomposition.

- Apply time series decomposition using additive model.

- Visualize and interpret the trend, seasonal, and residual components.

ALGORITHM:

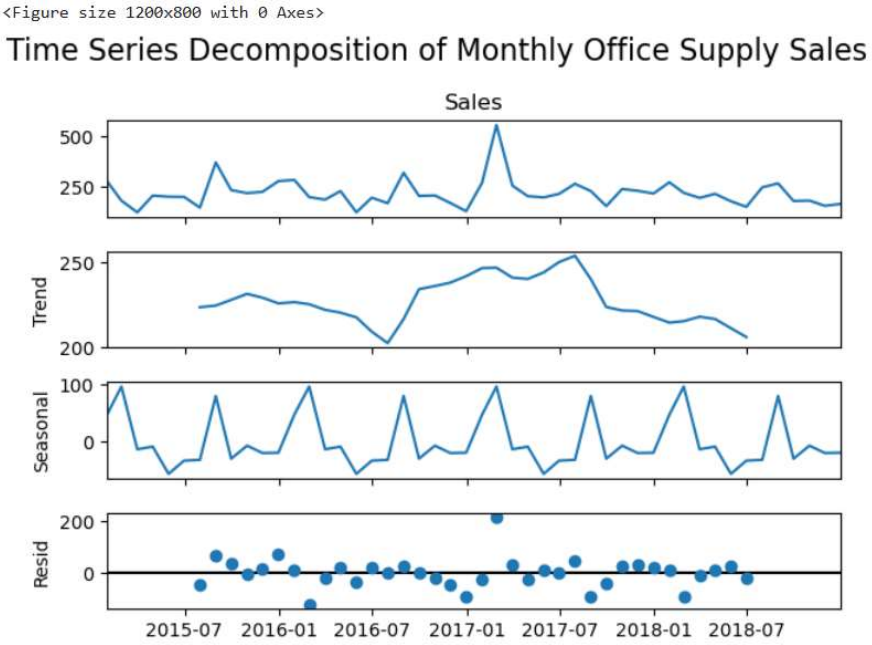
- Import required libraries.
- Load the cleaned sales dataset.
- Resample the data monthly to make it suitable for decomposition.
- Use the seasonal_decompose() function to perform additive decomposition.
- Plot the original series, trend, seasonality, and residual components

CODE:

```
import pandas as pd
import matplotlib.pyplot as plt
from statsmodels.tsa.seasonal import seasonal_decompose

# Load the CSV file
df = pd.read_csv("supermarket_sales - Sheet1.csv")

# Convert 'Date' column to datetime format
df['Date'] = pd.to_datetime(df['Date'])

# Group by Date and sum the 'Total' sales
daily_sales = df.groupby('Date')['Total'].sum().sort_index()

# Resample to ensure uniform daily frequency
daily_sales = daily_sales.asfreq('D').fillna(method='ffill')  # Forward-fill missing dates

# Decompose the time series (Additive model)
decomposition = seasonal_decompose(daily_sales, model='additive', period=7)  # Weekly seasonality

# Plot the original, trend, seasonal, and residual components
plt.figure(figsize=(14, 10))

plt.subplot(411)
plt.plot(daily_sales, label='Original', color='blue')
plt.legend(loc='upper left')

plt.subplot(412)
plt.plot(decomposition.trend, label='Trend', color='orange')
plt.legend(loc='upper left')

plt.subplot(413)
plt.plot(decomposition.seasonal, label='Seasonality', color='green')
plt.legend(loc='upper left')

plt.subplot(414)
plt.plot(decomposition.resid, label='Residual', color='red')
plt.legend(loc='upper left')

plt.tight_layout()
plt.show()
```

OUTPUT:

<Figure size 1200x800 with 0 Axes>



RESULT:

Thus,theprogramfor decomposing time series data into its trend, seasonal, and residual componentshasbeendone successfully.