EX:No.7 221501039

Implement program for decomposing time series data into trend and seasonality

Aim:

Write a program for decomposing time series data into trend and seasonality

Algorithm:

Step 1: Import Libraries

- **Import necessary Python libraries:**
  - `pandas` **for handling time series data.**
  - `numpy` **for numeric operations.**
  - `matplotlib.pyplot` **for plotting.**
  - `seasonal_decompose` **from** `statsmodels.tsa.seasonal` **for decomposition.**

Step 2: Generate or Load Time Series Data

- **You can either:**
  - **Load a real dataset (e.g., monthly sales, temperature, etc.)**
  - **Or generate synthetic data with trend, seasonality, and noise.**
- **Ensure the data is in time series format with a DateTime index.**

Step 3: Create a Pandas Series

- **Convert the data into a** `pandas.Series` **object with a** `DateTimeIndex`**.**
- **This is required for the decomposition function to work correctly.**

Step 4: Decompose the Time Series

- **Use the** `seasonal_decompose()` **function.**
- **Specify:**
  - `model='additive'` **(or** `'multiplicative'` **depending on your data pattern).**

- `period=12` **if you're working with monthly data (i.e., one seasonal cycle per year).**

- **This function separates the time series into:**
  - **Trend: Long-term progression of the series.**
  - **Seasonal: Repeating short-term cycle.**
  - **Residual: Random noise or irregular component.**

Step 5: Plot the Components

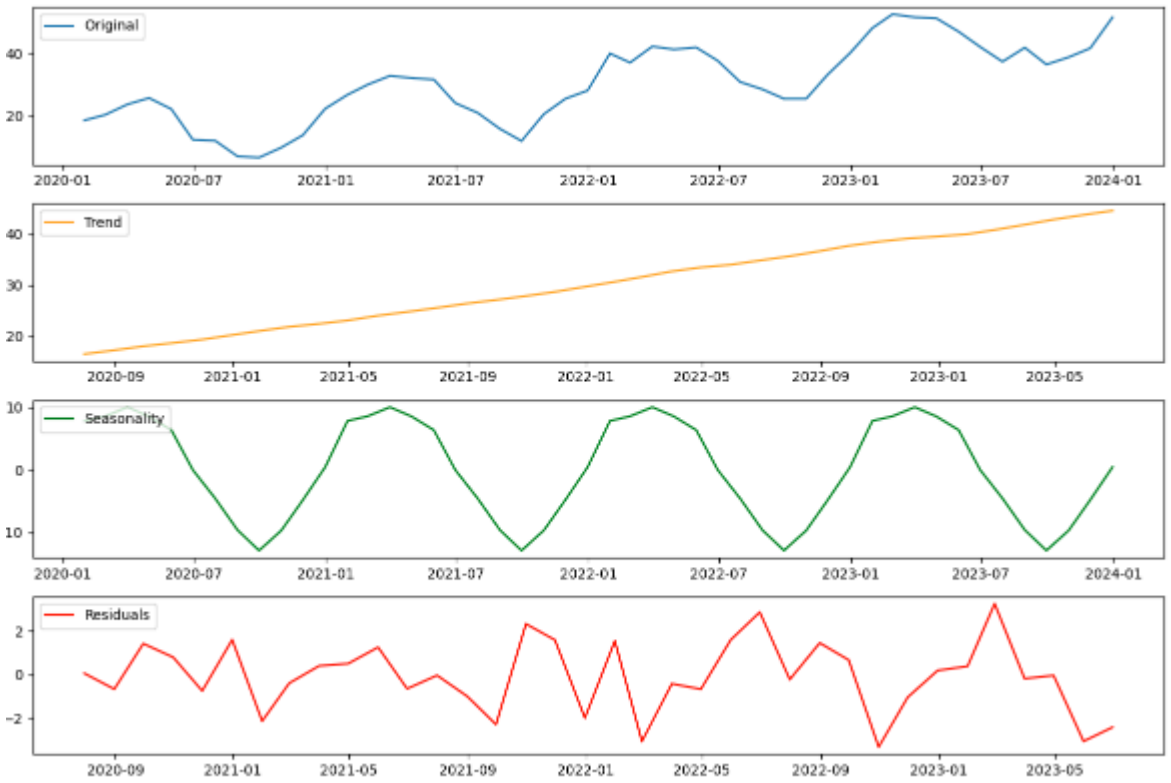
- **Plot each of the four components:**
  1. **Original Time Series**
  2. **Trend Component**
  3. **Seasonal Component**
  4. **Residual Component**

Code:

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

# Step 1: Generate a sample time series data with trend and seasonality
np.random.seed(0)
date_range = pd.date_range(start='2020-01-01', periods=48, freq='M')  # Monthly data for 4 years
trend = np.linspace(10, 50, 48)
seasonality = 10 * np.sin(2 * np.pi * date_range.month / 12)
noise = np.random.normal(0, 2, 48)
data = trend + seasonality + noise

# Step 2: Create a pandas Series
ts = pd.Series(data, index=date_range)
```

```
# Step 3: Decompose the time series
decomposition = seasonal_decompose(ts, model='additive', period=12)  # 12 months = 1 year

# Step 4: Plot the decomposition
plt.figure(figsize=(12, 8))
plt.subplot(411)
plt.plot(ts, label='Original')
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='Residuals', color='red')
plt.legend(loc='upper left')

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

Output:



Result:

Thus, the program for decomposing time series data into trend and seasonality was done.