|  |  |
| --- | --- |
| **EX.N0:7** | **Implement program for decomposingtime series data into trend and seasonality** |
| **DATE:07/04/2025** |

**AIM:**

To Implement program for decomposing time series data into trend and seasonality.

# PROGRAM:

# import pandas as pd

# import matplotlib.pyplot as plt

# from statsmodels.tsa.seasonal import seasonal\_decompose

# 

# # Load the dataset

# file\_path = r"/content/MSFT.csv"

# df = pd.read\_csv(file\_path, parse\_dates=['Date'], index\_col='Date')

# df.columns = df.columns.str.strip()

# print("\nAvailable columns:", df.columns)

# # Specify the target column for decomposition

# target\_col = "Adj Close"

# if target\_col not in df.columns:

# raise ValueError(f"'{target\_col}' column not found in the dataset.")

# # Extract the target time series and remove any missing values

# ts = df[target\_col].dropna()

# # Perform seasonal decomposition

# result = seasonal\_decompose(ts, model='additive', period=30)

# # Plot the decomposed components

# plt.figure(figsize=(12, 10))

# plt.subplot(411)

# plt.plot(ts, label="Original", color='blue')

# plt.title("Original Time Series")

# plt.legend()

# plt.subplot(412)

# plt.plot(result.trend, label="Trend", color='orange')

# plt.title("Trend Component")

# plt.legend()

# plt.subplot(413)

# plt.plot(result.seasonal, label="Seasonality", color='green')

# plt.title("Seasonal Component")

# plt.legend()

# plt.subplot(414)

# plt.plot(result.resid, label="Residuals", color='red')

# plt.title("Residual Component")

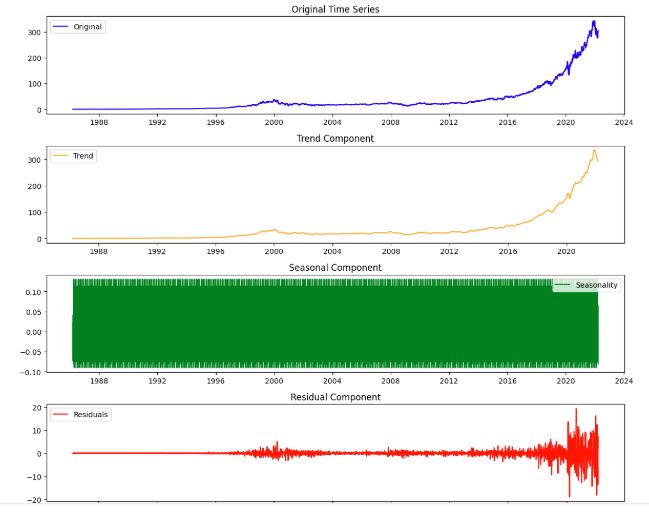
# plt.legend()

# # Adjust layout and display the plot

# plt.tight\_layout()

# plt.show()

# OUTPUT:

****

**RESULT:**

Thus, the program for Implement program for decomposing time series data into trend and seasonality is executed successfully.