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

**EX:No.7**

**DATE:29/03/25**

**AIM:**

To implement a program that decomposes time series data into its trend, seasonal, and residual components..

**ALGORITHM:**

1. Import the required libraries and load the time series dataset.
2. Convert the 'Date' column to datetime and set it as the index.
3. Ensure the time series has a consistent frequency (e.g., daily or monthly).
4. Use the seasonal\_decompose() function from statsmodels.
5. Specify the model type as either 'additive' or 'multiplicative' based on data behavior.
6. Decompose the series into trend, seasonal, and residual components.
7. Plot the original series along with its decomposed components for visualization.

**CODE:**

import pandas as pd

import matplotlib.pyplot as plt

from statsmodels.tsa.seasonal import seasonal\_decompose

# Load dataset

file\_path = '/content/AAPL.csv'  # Replace with your dataset path

data = pd.read\_csv(file\_path)

# Convert 'Date' to datetime and set as index

data['Date'] = pd.to\_datetime(data['Date'])

data.set\_index('Date', inplace=True)

# Use 'Close' prices

ts = data['Close']

# Decompose the time series

# period=30 assumes monthly seasonality for daily data (~1 month)

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

# Plot the decomposed components

plt.figure(figsize=(14, 10))

plt.subplot(411)

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

plt.legend(loc='upper left')

plt.subplot(412)

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

plt.legend(loc='upper left')

plt.subplot(413)

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

plt.legend(loc='upper left')

plt.subplot(414)

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

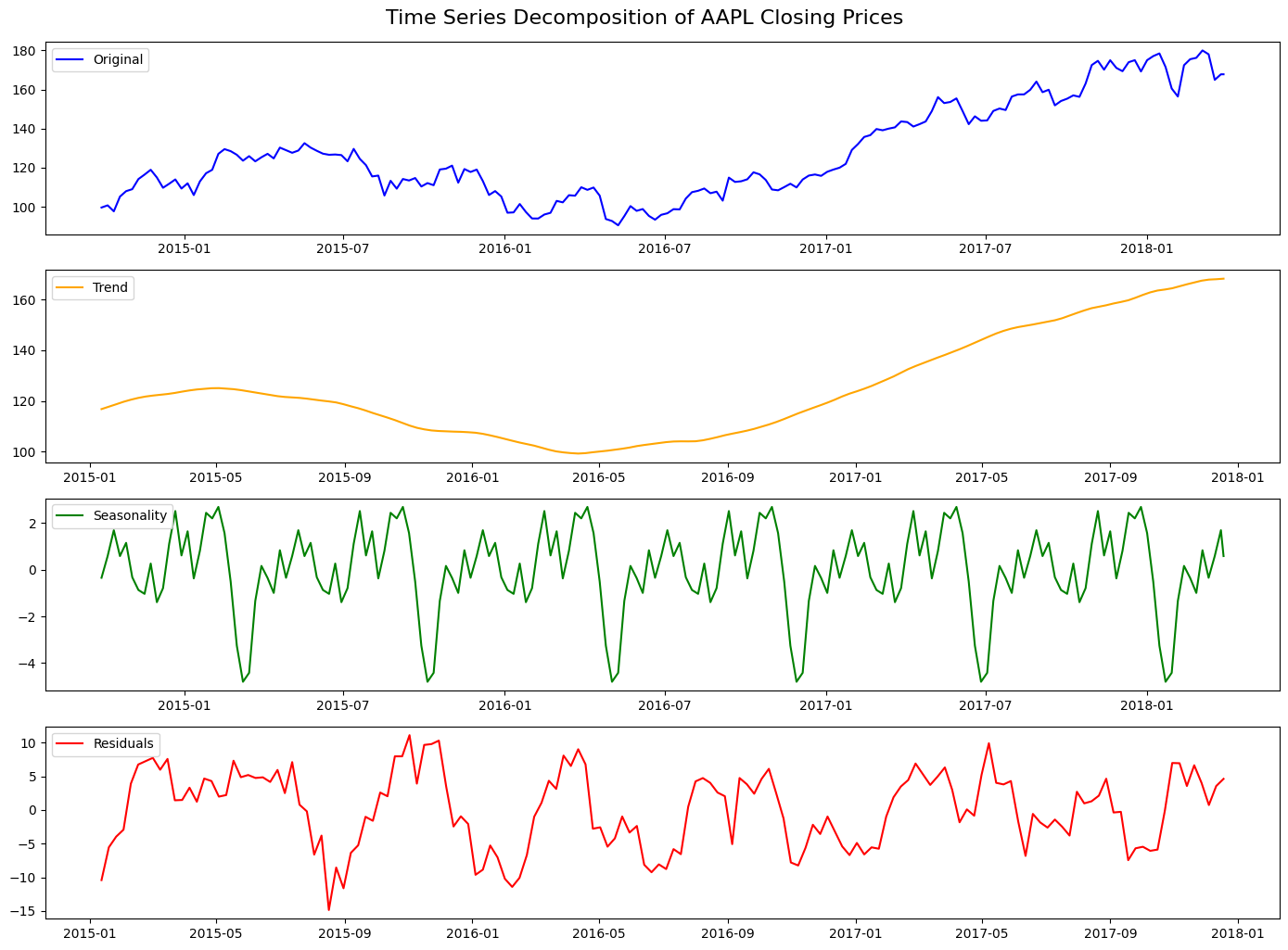
plt.legend(loc='upper left')

plt.tight\_layout()

plt.suptitle('Time Series Decomposition of AAPL Closing Prices', fontsize=16, y=1.02)

plt.show()

**OUTPUT:**



**RESULT:**

Thus the program has been completed and verified successfully.