**IMPLEMENT PROGRAMS FOR ESTIMATING & ELIMINATING TREND IN TIME SERIES DATA- AGGREGATION, SMOOTHING**

**AIM:** To Implement programs for estimating & eliminating trend in time series data- aggregation, smoothing

**PROCEDURE:**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

# Generate synthetic daily website visitor data

dates = pd.date\_range('2023-01-01', periods=365, freq='D')

visitors = np.random.poisson(lam=1000, size=365) + 50 \* np.sin(np.linspace(0, 10 \* np.pi, 365)) # Synthetic data with trend and seasonality

# Create a DataFrame

df = pd.DataFrame({'Date': dates, 'Visitors': visitors})

# Aggregate by week

df['Week'] = df['Date'].dt.to\_period('W')

# Group by 'Week' and sum only the 'Visitors' column

weekly\_aggregated = df.groupby('Week')['Visitors'].sum().reset\_index()

# Plot aggregated data

plt.figure(figsize=(10, 6))

plt.plot(weekly\_aggregated['Week'].astype(str), weekly\_aggregated['Visitors'], label='Weekly Aggregated Visitors', marker='o')

plt.title('Weekly Aggregated Website Visitors')

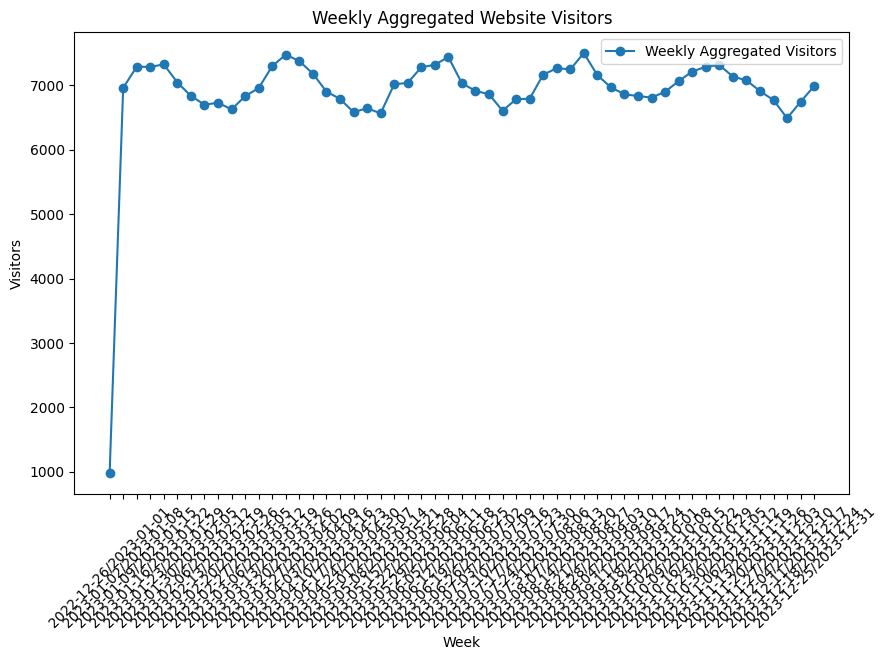
plt.xlabel('Week')

plt.ylabel('Visitors')

plt.xticks(rotation=45)

plt.legend()

plt.show()



# Smoothing with a 7-day moving average

window\_size = 7 # Weekly moving average

df['Moving\_Avg'] = df['Visitors'].rolling(window=window\_size).mean()

# Plot original vs smoothed data

plt.figure(figsize=(10, 6))

plt.plot(df['Date'], df['Visitors'], label='Original Daily Visitors', alpha=0.5)

plt.plot(df['Date'], df['Moving\_Avg'], label=f'{window\_size}-Day Moving Average', color='red')

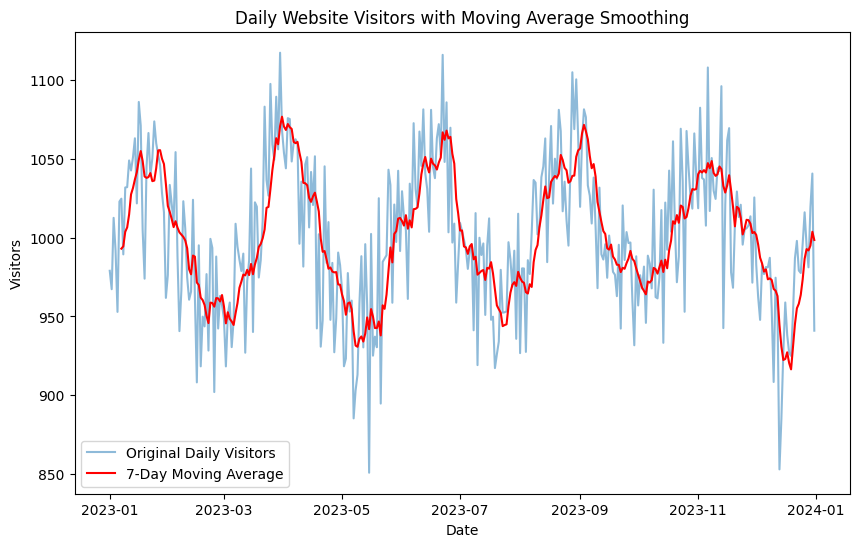
plt.title('Daily Website Visitors with Moving Average Smoothing')

plt.xlabel('Date')

plt.ylabel('Visitors')

plt.legend()

plt.show()



# Differencing to remove trend

df['Differenced'] = df['Visitors'].diff()

# Plot detrended data

plt.figure(figsize=(10, 6))

plt.plot(df['Date'], df['Differenced'], label='Detrended Data (Differencing)', color='green')

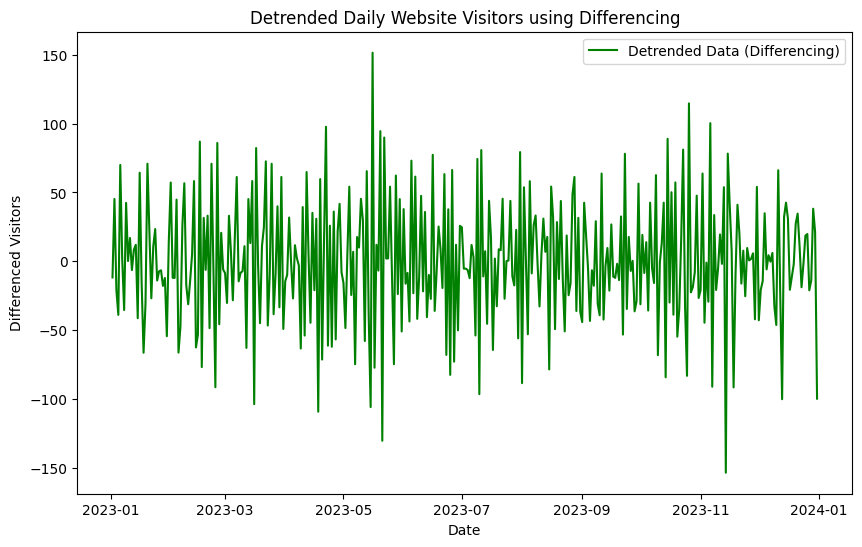
plt.title('Detrended Daily Website Visitors using Differencing')

plt.xlabel('Date')

plt.ylabel('Differenced Visitors')

plt.legend()

plt.show()



# Subtracting the moving average from the original data

df['Detrended'] = df['Visitors'] - df['Moving\_Avg']

# Plot detrended data

plt.figure(figsize=(10, 6))

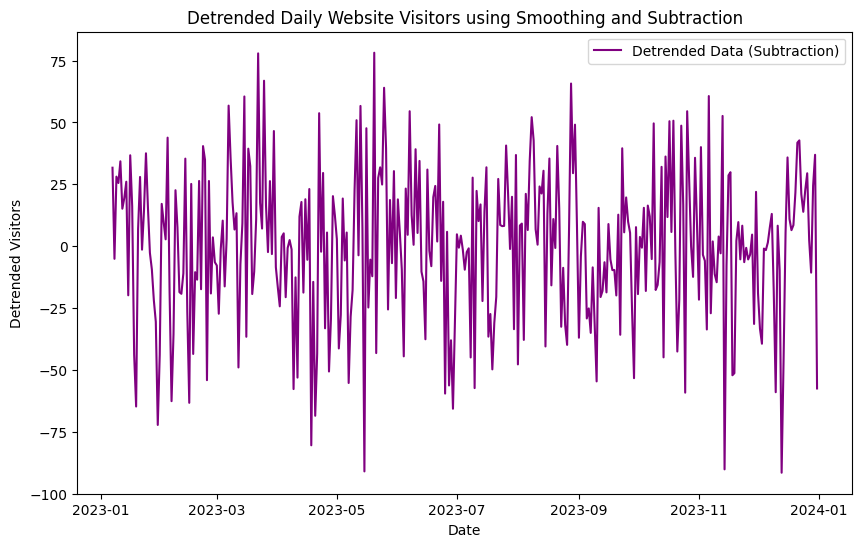
plt.plot(df['Date'], df['Detrended'], label='Detrended Data (Subtraction)', color='purple')

plt.title('Detrended Daily Website Visitors using Smoothing and Subtraction')

plt.xlabel('Date')

plt.ylabel('Detrended Visitors')

plt.legend()

plt.show()

**RESULT:** Thus the expected output has been excecuted successfully.