

```
In [169]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

```
In [170]: #Exercise 1: Create a line plot using matplotlib pyplot that displays the population of four different cities over
#time. Each city should have its own line, and the x-axis should represent years (e.g. 2010, 2011,2012, etc.)
#while the y-axis should represent the population.
```

```
# Create a DataFrame with population data for four cities over time
data = {'Year': range(2010, 2017),
        'City A': [500000, 550000, 600000, 650000, 700000, 750000, 800000],
        'City B': [800000, 850000, 900000, 950000, 1000000, 1050000, 1100000],
        'City C': [1000000, 1050000, 1100000, 1150000, 1200000, 1250000, 1300000],
        'City D': [1200000, 1250000, 1300000, 1350000, 1400000, 1450000, 1500000]}

df = pd.DataFrame(data)

# Set the 'Year' column as the index for the DataFrame
df.set_index('Year', inplace=True)

# Create a line chart for the population of four cities over time
plt.figure(figsize=(10, 6)) # Set the figure size

# Plot City A with custom marker and Label
plt.plot(df.index, df['City A'], label='City A', marker='*', markersize=9)

# Plot City B with custom marker and Label
plt.plot(df.index, df['City B'], label='City B', marker='*', markersize=9)

# Plot City C with custom marker and Label
plt.plot(df.index, df['City C'], label='City C', marker='*', markersize=9)

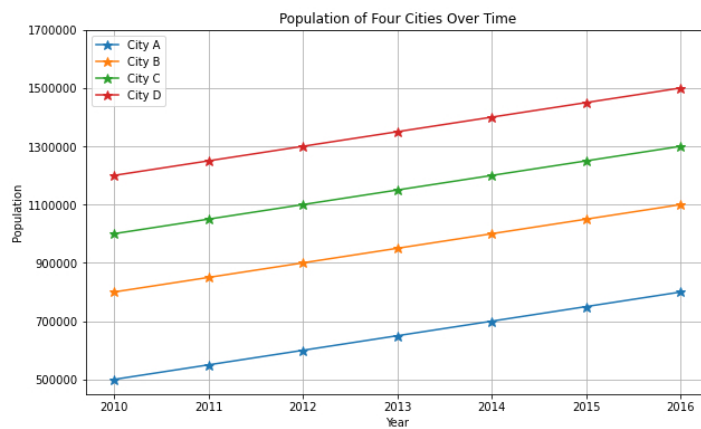
# Plot City D with custom marker and Label
plt.plot(df.index, df['City D'], label='City D', marker='*', markersize=9)

# Set custom y-axis tick values
ticks = np.arange(500000, 1800000, 200000)
plt.yticks(ticks, ticks)

# Set Labels and title
plt.xlabel('Year') # Set the x-axis Label
plt.ylabel('Population') # Set the y-axis Label
plt.title('Population of Four Cities Over Time') # Set the plot title

plt.legend() # Add a Legend for city names
plt.grid(True) # Add a grid to the plot

# Show the plot
plt.show()
```



```
In [171]: #Exercise 2: Create a scatter plot using matplotlib pyplot that shows the relationship between the number of
#hours studied and the test scores obtained by a group of students. Use the following
#data: Hours Studied: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10], Test Scores: [93, 57, 61, 54, 51, 53, 87, 81, 83, 85]
```

```
Hours_Studied = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10],
Test_Scores = [93, 57, 61, 54, 51, 53, 87, 81, 83, 85]

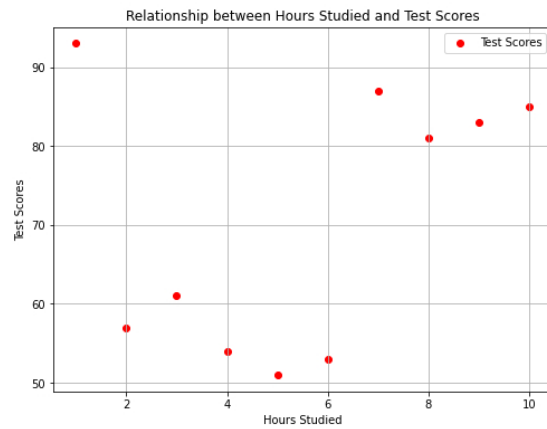
# Create a scatter plot
plt.figure(figsize=(8, 6)) # Optional: Adjust the figure size
plt.scatter(Hours_Studied, Test_Scores, c='red', marker='o', label='Test Scores')

# Add Labels and title
plt.xlabel('Hours Studied')
plt.ylabel('Test Scores')
plt.title('Relationship between Hours Studied and Test Scores')

# Add a grid (optional)
plt.grid(True)

# Show the plot
plt.legend()
```

```
plt.show()
```



In [172]: *#Exercise 3: Create a bar chart using matplotlib pyplot that shows the total sales for each month of the year.*

*#Use the following data:*

*#Month: ["Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"]*

*#Sales: [11860, 10480, 4997, 5523, 13965, 6011, 13158, 9533, 5158, 9058, 11346, 6675]*

*# Data*

*months = ["Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"]*

*sales = [11860, 10480, 4997, 5523, 13965, 6011, 13158, 9533, 5158, 9058, 11346, 6675]*

*# Create a bar chart*

*plt.figure(figsize=(8,6))*

*plt.bar(months, sales, color='green', alpha=0.8)*

*# Add labels and title*

*plt.xlabel('Month')*

*plt.ylabel('Total Sales')*

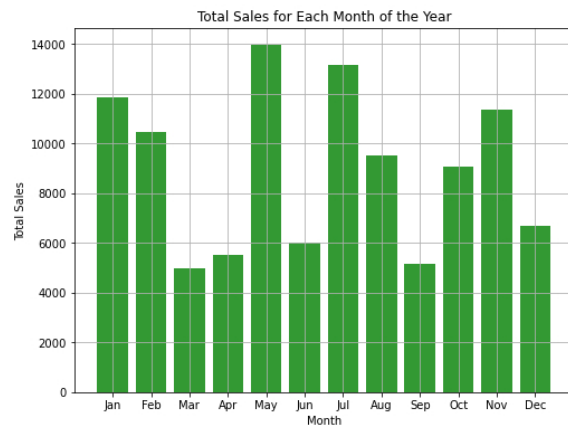
*plt.title('Total Sales for Each Month of the Year')*

*# Add a grid (optional)*

*plt.grid(True)*

*# Show the plot*

*plt.show()*



In [ ]: