

1)a. Write a python program to find the best of two test average marks out of three test's marks accepted from the user.

Code:-

```
m1 = int(input("Enter marks for test1 : "))
m2 = int(input("Enter marks for test2 : "))
m3 = int(input("Enter marks for test3 : "))
if m1 <= m2 and m1 <= m3:
    avgMarks = (m2+m3)/2
elif m2 <= m1 and m2 <= m3:
    avgMarks = (m1+m3)/2
elif m3 <= m1 and m2 <= m2:
    avgMarks = (m1+m2)/2
best_of_two = sorted([m1, m2, m3], reverse=True)[:2]
average_best_of_two = sum(best_of_two)/2
print("Average of best two test marks out of three test's marks is", average_best_of_two);
```

Output:-

Enter marks for test1: 85

Enter marks for test2: 92

Enter marks for test3: 78

Average of best two test marks out of three test's marks is 90.0

1)b. Develop a Python program to check whether a given number is palindrome or not and also count the number of occurrences of each digit in the input number.

Code:-

```
from collections import Counter
value = input("Enter a value : ")
if value == value[::-1]:
    print("Palindrome")
else:
    print("Not Palindrome")
counted_dict = Counter(value)
for key in sorted(counted_dict.keys()):
    print(f'{key} appears {counted_dict[key]} times');
```

Output:-

Enter a value: 987654

Not Palindrome

4 appears 1 times

5 appears 1 times

6 appears 1 times

7 appears 1 times

8 appears 1 times

9 appears 1 times

Enter a value: 123321

Palindrome

1 appears 2 times

2 appears 2 times

3 appears 2 times

3)a. Write a Python program that accepts a sentence and find the number of words, digits, uppercase letters and lowercase letters.

Code:-

```
import string
sentence = input("Enter a sentence : ")
wordList = sentence.strip().split(" ")
print(f'This sentence has {len(wordList)} words', end='\n\n')
digit_count = uppercase_count = lowercase_count = 0
for character in sentence:
    if character in string.digits:
        digit_count += 1
    elif character in string.ascii_uppercase:
        uppercase_count += 1
    elif character in string.ascii_lowercase:
        lowercase_count += 1

print(f'This sentence has {digit_count} digits',
      f' {uppercase_count} upper case letters',
      f' {lowercase_count} lower case letters', sep='\n')
```

Output:-

Enter a sentence : Rama went to Devaraja market to pick 2 kgs of vegetable
This sentence has 11 words

This sentence has 1 digits
2 upper case letters
42 lower case letters
Enter a sentence: Python is Fun!
This sentence has 3 words

This sentence has 0 digits
3 uppercase letters
9 lowercase letters
Enter a sentence: Hello, World! 123
This sentence has 3 words

This sentence has 3 digits
1 uppercase letters
12 lowercase letters

3)b. Write a Python program to find the string similarity between two given strings.

Code:-

```
from difflib import SequenceMatcher
```

```
str1 = input("Enter String 1 : ")
```

```
str2 = input("Enter String 2 : ")
```

```
sim = SequenceMatcher(None, str1, str2).ratio()
```

```
print("Similarity between strings \"" + str1 + "\" and \"" + str2 + "\" is : ",sim)
```

Output:-

```
Enter String 1 : Python Exercises
```

```
Enter String 2 : Python Exercise
```

```
Similarity between strings "Python Exercises" and "Python Exercise" is : 0.967741935483871
```

```
Enter String 1 : Python Exercises
```

```
Enter String 2 : Python Exercises
```

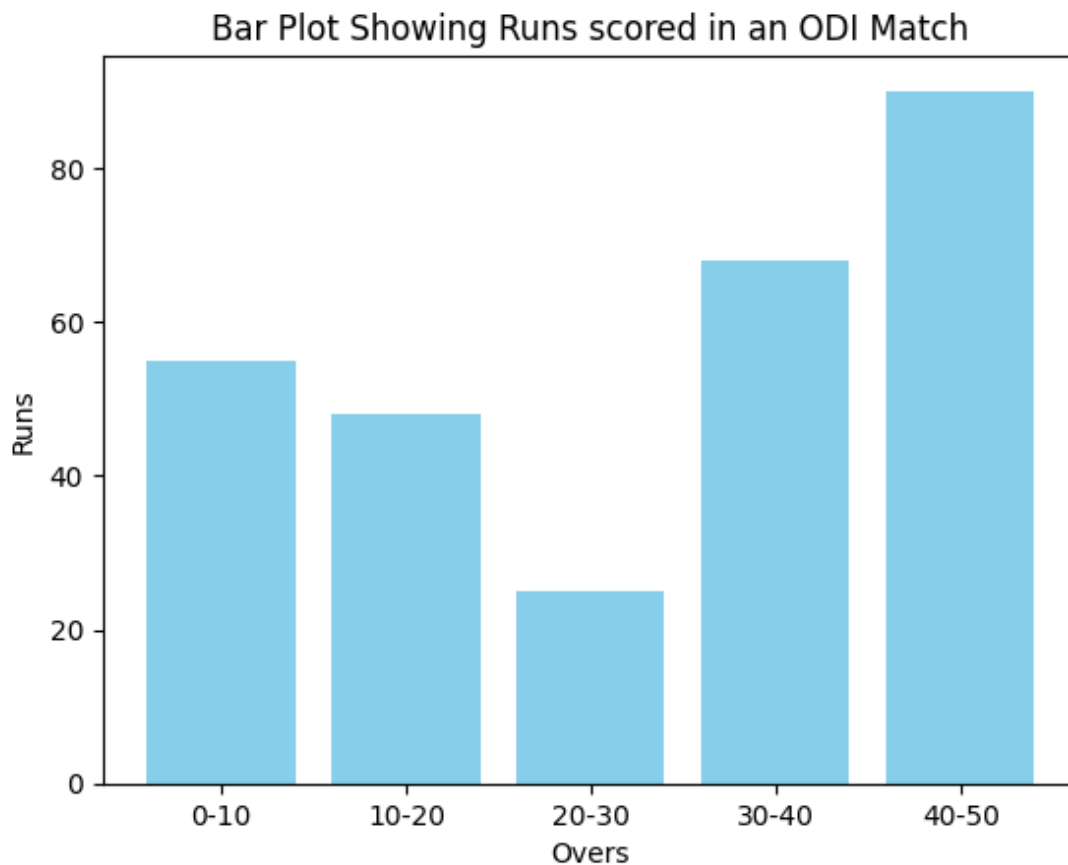
```
Similarity between strings "Python Exercises" and "Python Exercises" is : 1.0
```

4)a. Write a Python program to Demonstrate how to Draw a Bar Plot using Matplotlib.

Code:-

```
import matplotlib.pyplot as plt
categories = ['0-10', '10-20', '20-30', '30-40', '40-50']
values = [55, 48, 25, 68, 90]
plt.bar(categories, values, color='skyblue')
plt.xlabel('Overs')
plt.ylabel('Runs')
plt.title('Bar Plot Showing Runs scored in an ODI Match')
plt.show()
```

Output:-

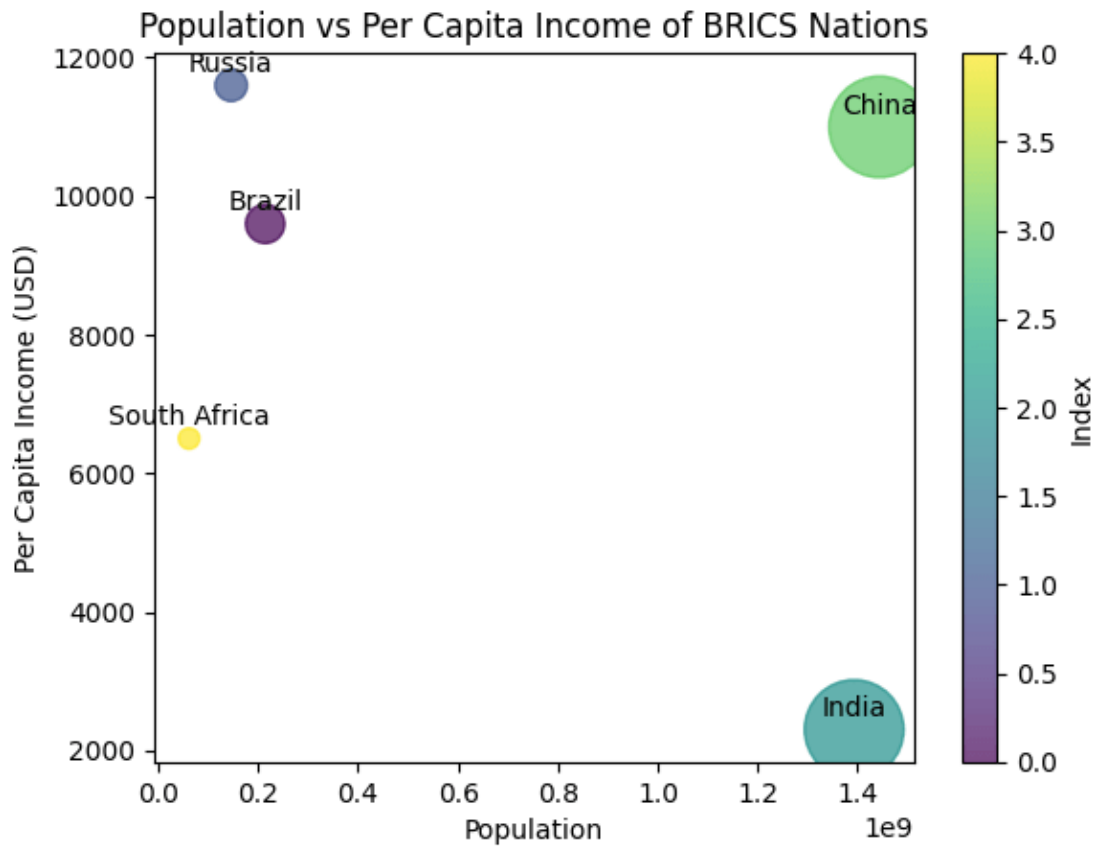


4)b. Write a Python program to Demonstrate how to Draw a Scatter Plot using Matplotlib.

Code:-

```
import matplotlib.pyplot as plt
import numpy as np
countries = ['Brazil', 'Russia', 'India', 'China', 'South Africa']
population = [213993437, 145912025, 1393409038, 1444216107, 61608912] # Population in 2021
per_capita_income = [9600, 11600, 2300, 11000, 6500] # Per capita income in USD
circle_size = [pop / 1000000 for pop in population] # Scaling down for better visualization
colors = np.arange(len(countries))
scatter = plt.scatter(population, per_capita_income, s=circle_size, c=colors, cmap='viridis',
alpha=0.7, label='BRICS Nations')
for i, country in enumerate(countries):
    plt.annotate(country, (population[i], per_capita_income[i]), textcoords="offset points",
xytext=(0,5), ha='center')
plt.colorbar(scatter, label='Index')
plt.xlabel('Population')
plt.ylabel('Per Capita Income (USD)')
plt.title('Population vs Per Capita Income of BRICS Nations')
plt.show()
```

Output:-

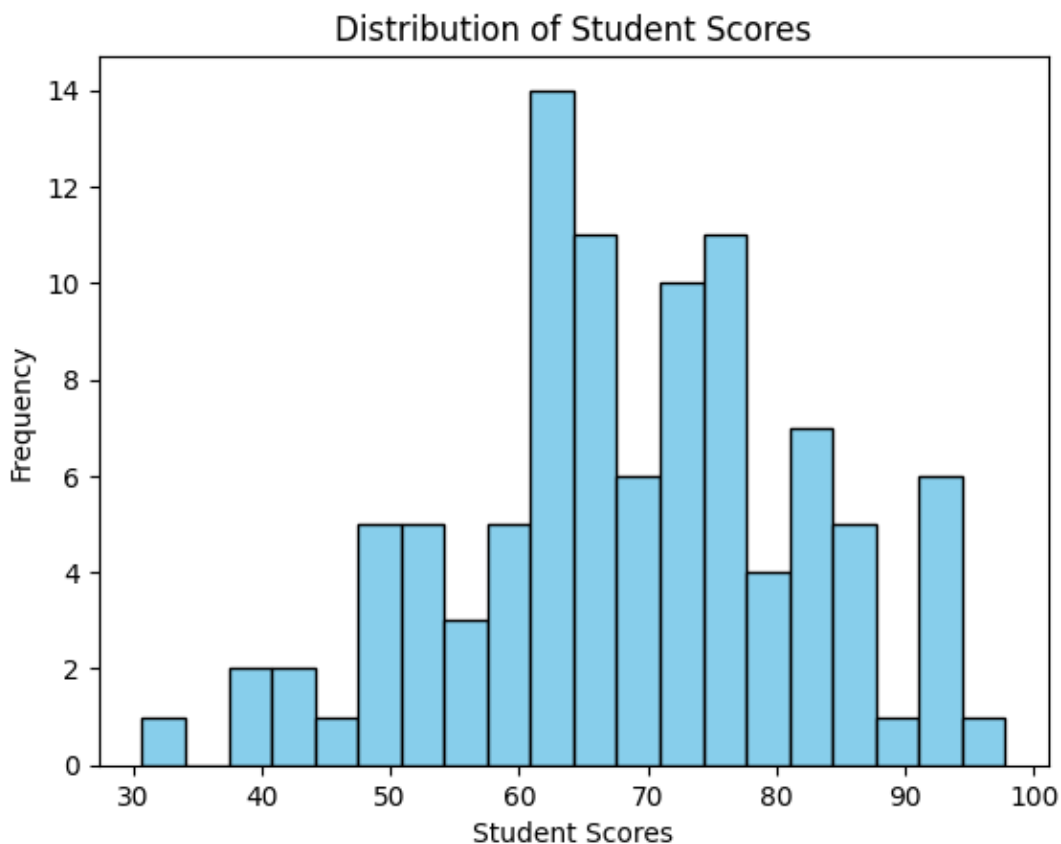


5)a. Write a Python program to Demonstrate how to Draw a Histogram Plot using Matplotlib.

Code:-

```
import matplotlib.pyplot as plt
import numpy as np
np.random.seed(42)
student_scores = np.random.normal(loc=70, scale=15, size=100)
plt.hist(student_scores, bins=20, color='skyblue', edgecolor='black')
plt.xlabel('Student Scores')
plt.ylabel('Frequency')
plt.title('Distribution of Student Scores')
plt.show()
```

Output:-

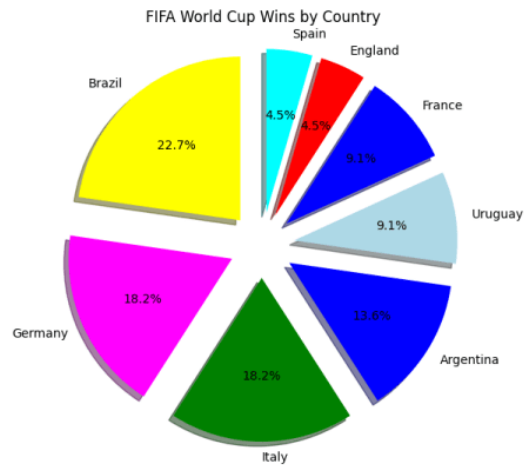


5)b. Write a Python program to Demonstrate how to Draw a Pie Chart using Matplotlib.

Code:-

```
import matplotlib.pyplot as plt
countries = ['Brazil', 'Germany', 'Italy', 'Argentina', 'Uruguay', 'France', 'England', 'Spain']
wins = [5, 4, 4, 3, 2, 2, 1, 1] # Replace with actual data
colors = ['yellow', 'magenta', 'green', 'blue', 'lightblue', 'blue', 'red', 'cyan']
plt.pie(wins, labels=countries, autopct='%1.1f%%', colors=colors, startangle=90, explode=[0.2,
0.2, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2], shadow=True)
plt.title('FIFA World Cup Wins by Country')
plt.axis('equal') # Equal aspect ratio ensures that the pie chart is circular.
plt.show()
```

Output:-

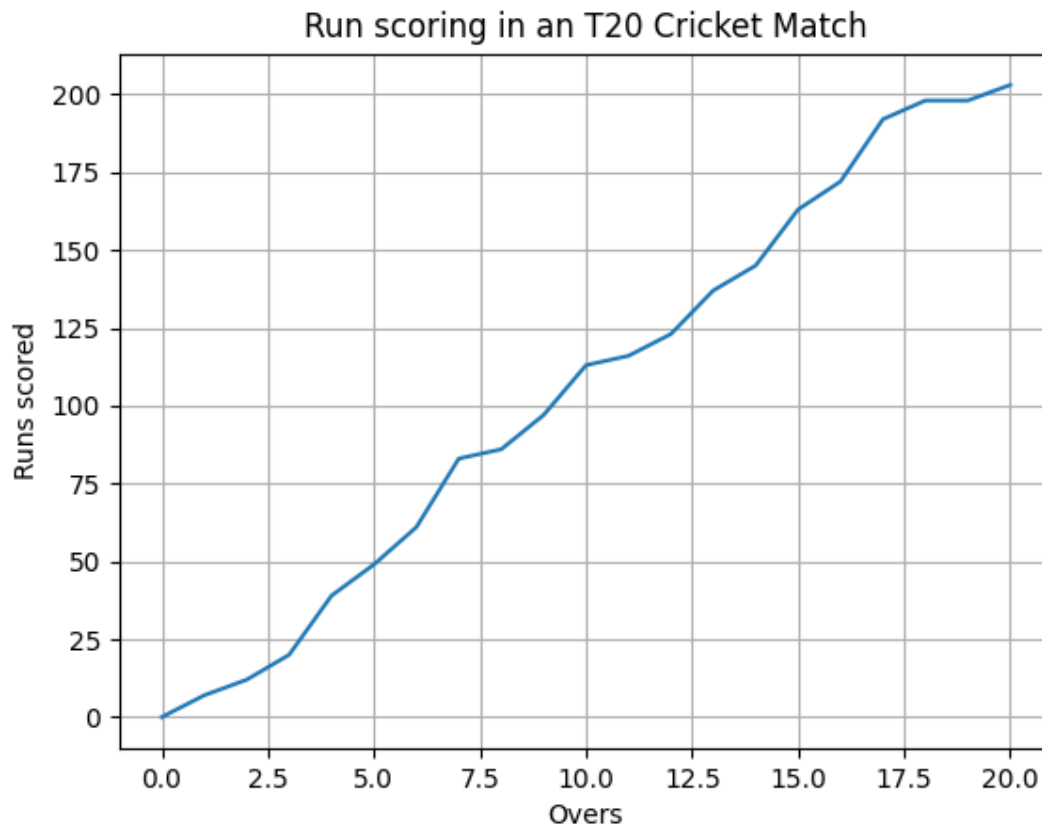


6)a. Write a Python program to illustrate Linear Plotting using Matplotlib.

Code:-

```
import matplotlib.pyplot as plt
overs = [0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20]
runs_scored = [0,7,12,20,39,49,61,83,86,97,113,116,123,137,145,163,172,192,198,198,203]
plt.plot(overs, runs_scored)
plt.xlabel('Overs')
plt.ylabel('Runs scored')
plt.title('Run scoring in an T20 Cricket Match')
plt.grid(True)
plt.show()
```

Output:-

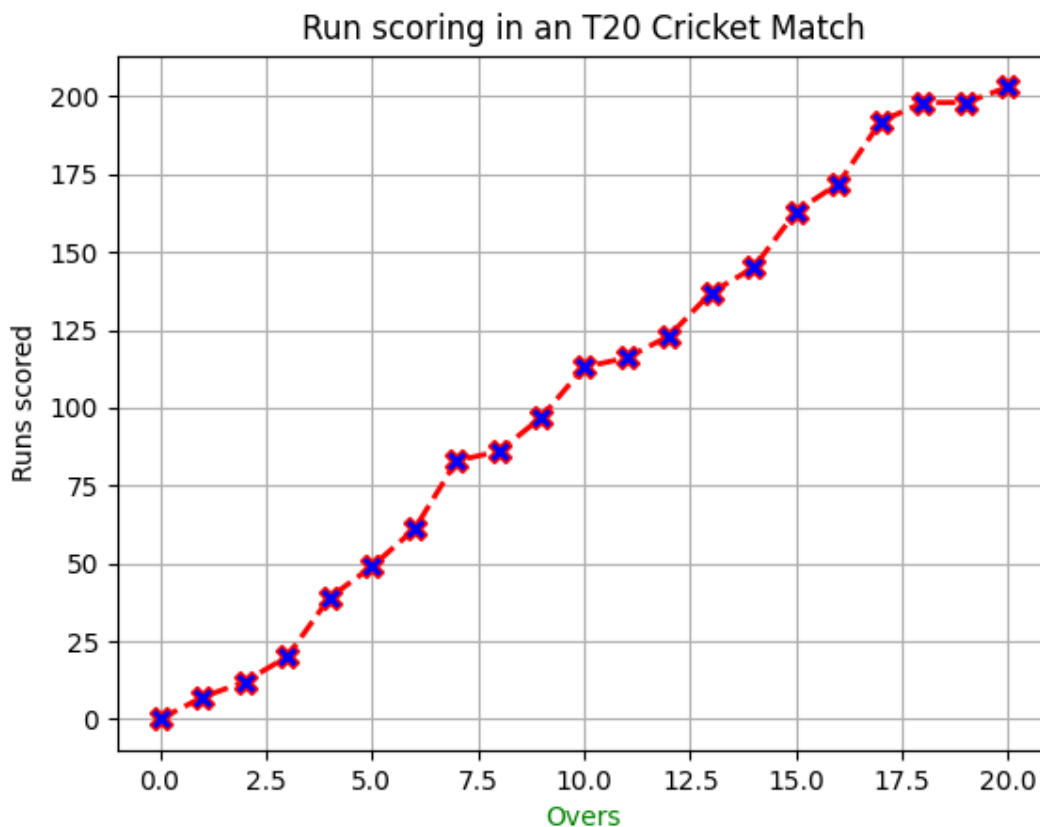


6)b. Write a Python program to illustrate liner plotting with line formatting using Matplotlib.

Code:-

```
import matplotlib.pyplot as plt
overs = [0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20]
runs_scored = [0,7,12,20,39,49,61,83,86,97,113,116,123,137,145,163,172,192,198,198,203]
plt.plot(overs, runs_scored, marker='X', linestyle='dashed',color='red', linewidth=2,
markerfacecolor='blue', markersize=8)
plt.xlabel('Overs', color = 'green')
plt.ylabel('Runs scored')
plt.title('Run scoring in an T20 Cricket Match')
plt.grid(True)
plt.show()
```

Output:-

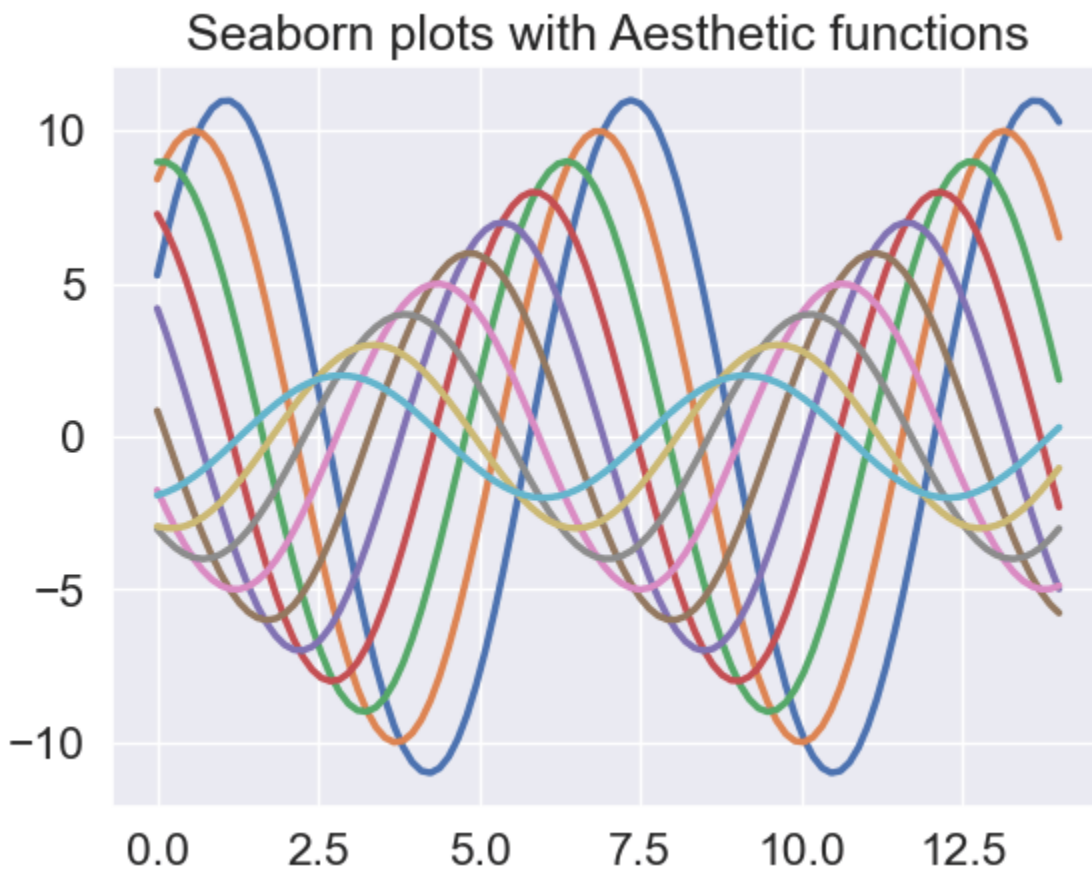


7)a. Write a Python program which explains uses of customizing seaborn plots with Aesthetic functions.

Code:-

```
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
def sinplot(n=10):
    x = np.linspace(0, 14, 100)
    for i in range(1, n + 1):
        plt.plot(x, np.sin(x + i * .5) * (n + 2 - i))
sns.set_theme()
sns.set_context("notebook", font_scale=1.5, rc={"lines.linewidth": 2.5})
sinplot()
plt.title('Seaborn plots with Aesthetic functions')
plt.show()
```

Output:-

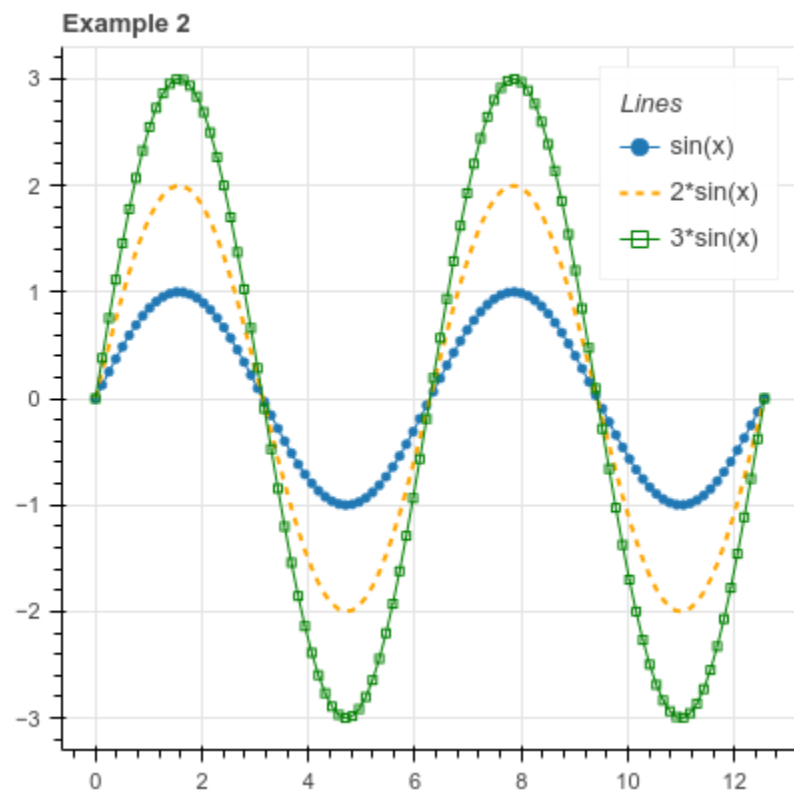
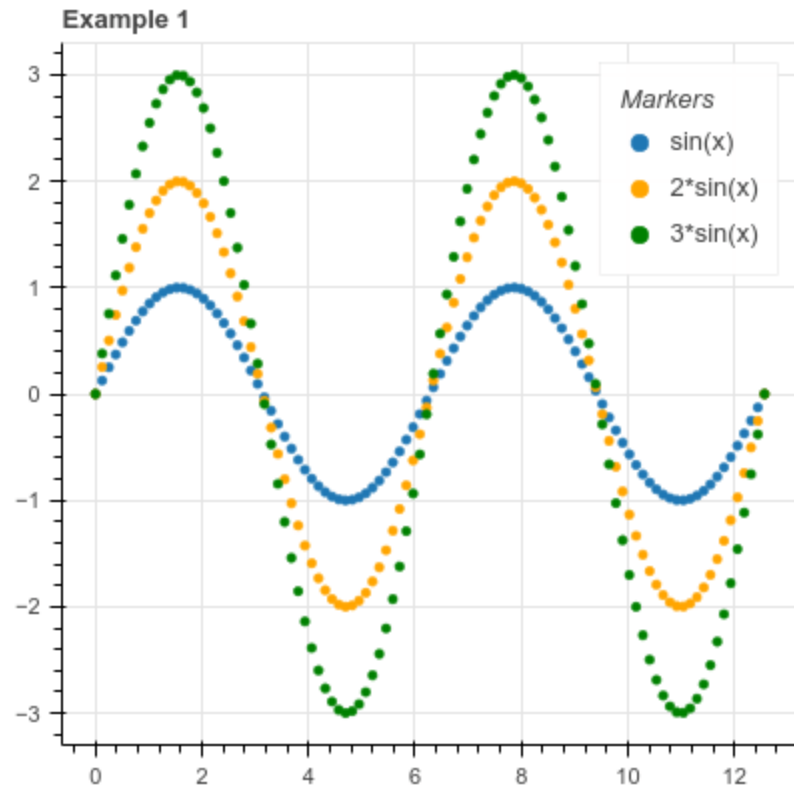


8)a. Write a Python program to explain working with Bokeh line graph using Annotations and Legends.

a) Write a Python program for plotting different types of plots using Bokeh.

Code:-

```
import numpy as np
from bokeh.layouts import gridplot
from bokeh.plotting import figure, show
x = np.linspace(0, 4*np.pi, 100)
y = np.sin(x)
TOOLS = "pan,wheel_zoom,box_zoom,reset,save,box_select"
p1 = figure(title="Example 1", tools=TOOLS)
p1.circle(x, y, legend_label="sin(x)")
p1.circle(x, 2*y, legend_label="2*sin(x)", color="orange")
p1.circle(x, 3*y, legend_label="3*sin(x)", color="green")
p1.legend.title = 'Markers'
p2 = figure(title="Example 2", tools=TOOLS)
p2.circle(x, y, legend_label="sin(x)")
p2.line(x, y, legend_label="sin(x)")
p2.line(x, 2*y, legend_label="2*sin(x)",
        line_dash=(4, 4), line_color="orange", line_width=2)
p2.square(x, 3*y, legend_label="3*sin(x)", fill_color=None, line_color="green")
p2.line(x, 3*y, legend_label="3*sin(x)", line_color="green")
p2.legend.title = 'Lines'
show(gridplot([p1, p2], ncols=2, width=400, height=400))
```



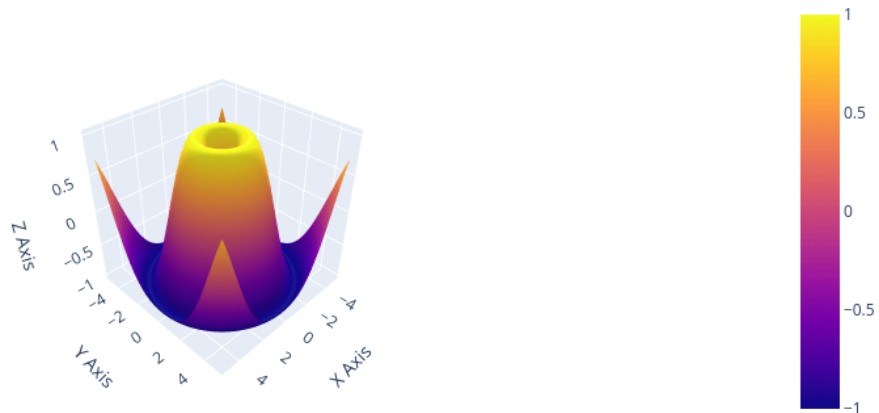
9)a. Write a Python program to draw 3D Plots using Plotly Libraries.

Code:-

```
import plotly.graph_objects as go
import numpy as np
x = np.linspace(-5, 5, 100)
y = np.linspace(-5, 5, 100)
x, y = np.meshgrid(x, y)
z = np.sin(np.sqrt(x**2 + y**2))
fig = go.Figure(data=[go.Surface(z=z, x=x, y=y)])
fig.update_layout(scene=dict(
    xaxis_title='X Axis',
    yaxis_title='Y Axis',
    zaxis_title='Z Axis'),
    margin=dict(l=0, r=0, b=0, t=40),
    title='3D Surface Plot of sin(sqrt(x^2 + y^2))')
fig.show()
```

Output:-

3D Surface Plot of sin(sqrt(x^2 + y^2))



10)b. Write a Python program for creating Maps using Plotly Libraries.

Code:-

```
import plotly.express as px
import pandas as pd
data=
pd.read_csv('https://raw.githubusercontent.com/plotly/datasets/master/gapminder_with_codes.csv')
fig = px.choropleth(data, locations='iso_alpha', color='gdpPercap', hover_name='country',
                    projection='natural earth', title='GDP per Capita by Country')
fig.show()
```

Output:-

GDP per Capita by Country

