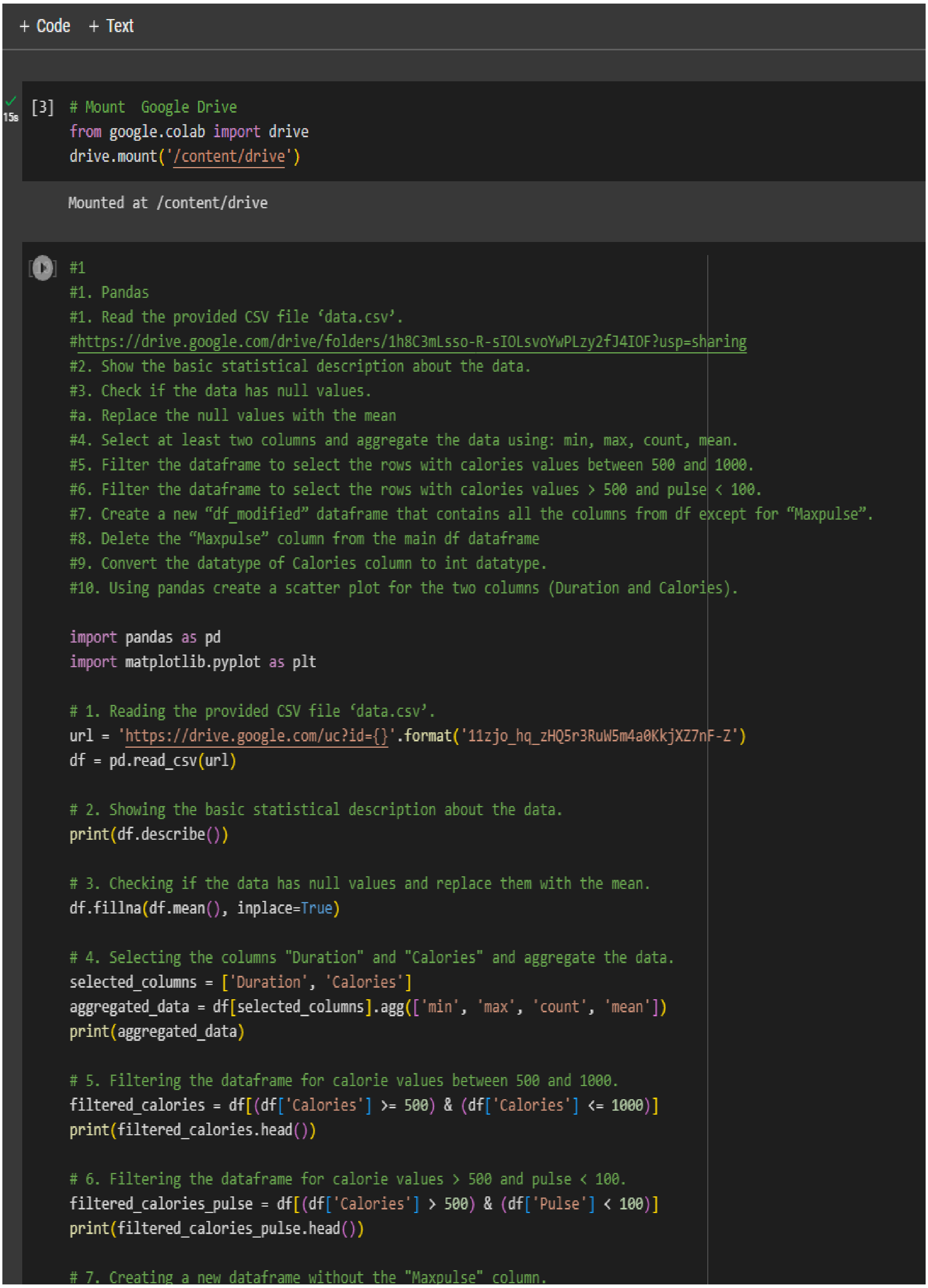
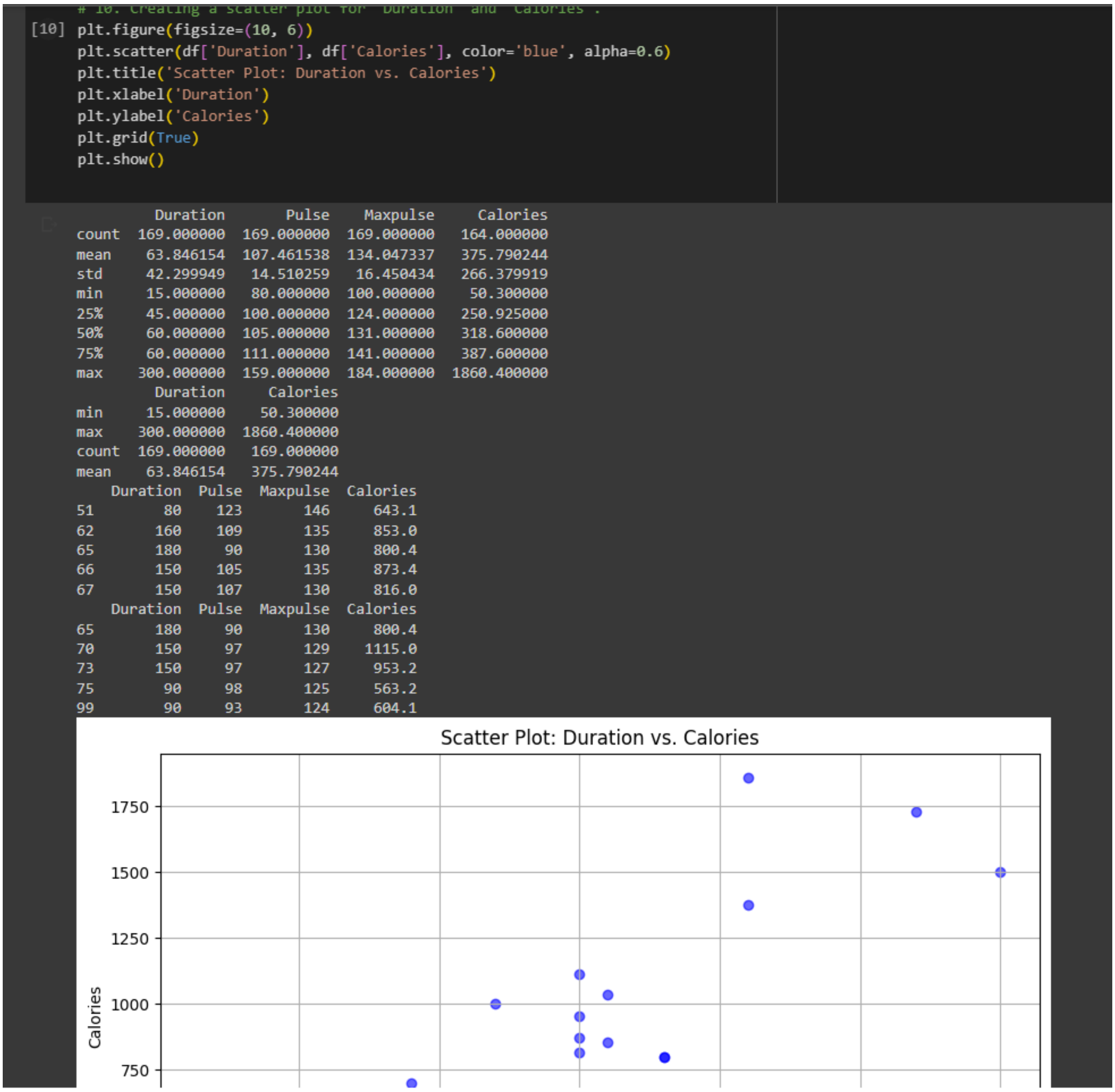


```
# Mount  Google Drive
from google.colab import drive
drive.mount('/content/drive')
```

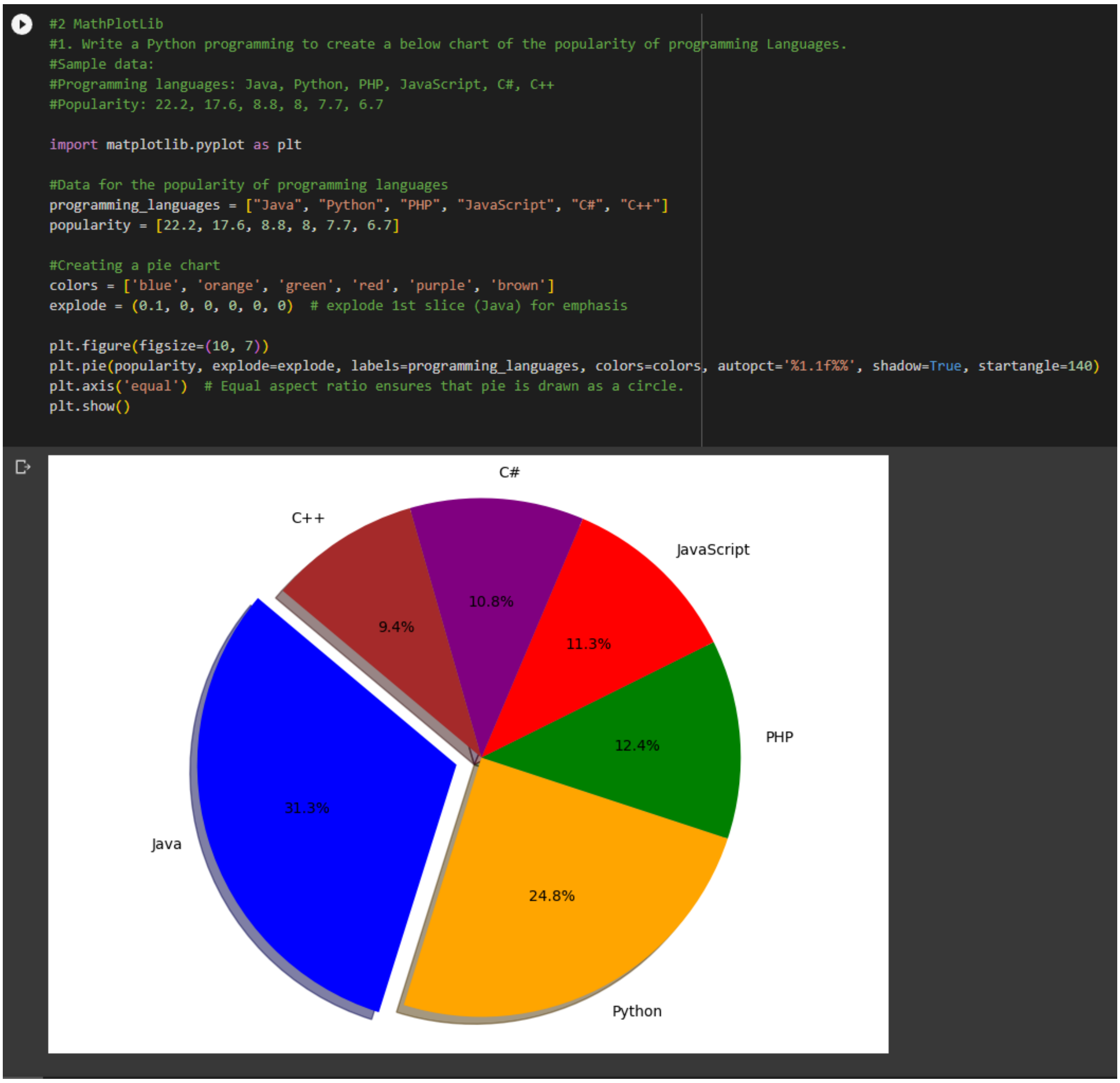
```
Mounted at /content/drive
```

```
#1
#1. Pandas
#1. Read the provided CSV file ‘data.csv’.
#https://drive.google.com/drive/folders/1h8C3mLsso-R-sIOLsvoYwPLzy2fJ4IOF?usp=sharing
#2. Show the basic statistical description about the data.
#3. Check if the data has null values.
#a. Replace the null values with the mean
#4. Select at least two columns and aggregate the data using: min, max, count, mean.
#5. Filter the dataframe to select the rows with calories values between 500 and 1000.
#6. Filter the dataframe to select the rows with calories values > 500 and pulse < 100.
#7. Create a new “df_modified” dataframe that contains all the columns from df except for “Maxpulse”.
#8. Delete the “Maxpulse” column from the main df dataframe
#9. Convert the datatype of Calories column to int datatype.
#10. Using pandas create a scatter plot for the two columns (Duration and Calories).

import pandas as pd
import matplotlib.pyplot as plt

# 1. Reading the provided CSV file ‘data.csv’.
url = 'https://drive.google.com/uc?id={}'.format('11zjo_hq_zHQ5r3RuW5m4a0KkjXZ7nF-Z')
df = pd.read_csv(url)

# 2. Showing the basic statistical description about the data.
print(df.describe())

# 3. Checking if the data has null values and replace them with the mean.
df.fillna(df.mean(), inplace=True)

# 4. Selecting the columns "Duration" and "Calories" and aggregate the data.
selected_columns = ['Duration', 'Calories']
aggregated_data = df[selected_columns].agg(['min', 'max', 'count', 'mean'])
print(aggregated_data)

# 5. Filtering the dataframe for calorie values between 500 and 1000.
filtered_calories = df[(df['Calories'] >= 500) & (df['Calories'] <= 1000)]
print(filtered_calories.head())

# 6. Filtering the dataframe for calorie values > 500 and pulse < 100.
filtered_calories_pulse = df[(df['Calories'] > 500) & (df['Pulse'] < 100)]
print(filtered_calories_pulse.head())

# 7. Creating a new dataframe without the "Maxpulse" column.
```

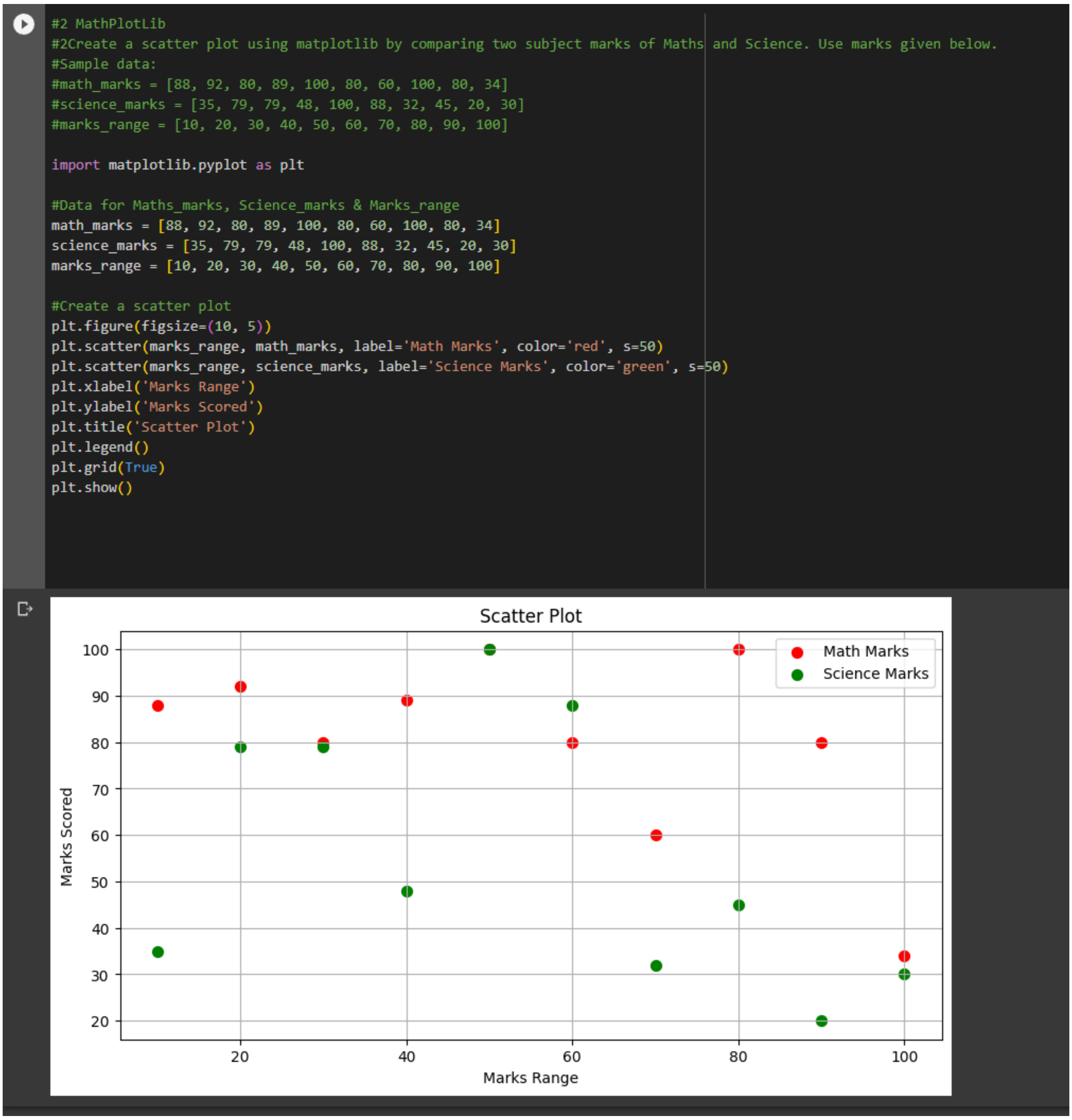


```
# 10. Creating a scatter plot for 'Duration' and 'Calories'.
[10] plt.figure(figsize=(10, 6))
     plt.scatter(df['Duration'], df['Calories'], color='blue', alpha=0.6)
     plt.title('Scatter Plot: Duration vs. Calories')
     plt.xlabel('Duration')
     plt.ylabel('Calories')
     plt.grid(True)
     plt.show()
```

```
         Duration       Pulse    Maxpulse     Calories
count  169.000000  169.000000  169.000000   164.000000
mean    63.846154  107.461538  134.047337   375.790244
std     42.299949   14.510259   16.450434   266.379919
min     15.000000   80.000000  100.000000    50.300000
25%     45.000000  100.000000  124.000000   250.925000
50%     60.000000  105.000000  131.000000   318.600000
75%     60.000000  111.000000  141.000000   387.600000
max    300.000000  159.000000  184.000000  1860.400000
         Duration     Calories
min     15.000000    50.300000
max    300.000000  1860.400000
count  169.000000   169.000000
mean    63.846154   375.790244
    Duration  Pulse  Maxpulse  Calories
51        80    123       146     643.1
62       160    109       135     853.0
65       180     90       130     800.4
66       150    105       135     873.4
67       150    107       130     816.0
    Duration  Pulse  Maxpulse  Calories
65       180     90       130     800.4
70       150     97       129    1115.0
73       150     97       127     953.2
75        90     98       125     563.2
99        90     93       124     604.1
```



```
#2 MathPlotLib
#1. Write a Python programming to create a below chart of the popularity of programming Languages.
#Sample data:
#Programming languages: Java, Python, PHP, JavaScript, C#, C++
#Popularity: 22.2, 17.6, 8.8, 8, 7.7, 6.7

import matplotlib.pyplot as plt

#Data for the popularity of programming languages
programming_languages = ["Java", "Python", "PHP", "JavaScript", "C#", "C++"]
popularity = [22.2, 17.6, 8.8, 8, 7.7, 6.7]

#Creating a pie chart
colors = ['blue', 'orange', 'green', 'red', 'purple', 'brown']
explode = (0.1, 0, 0, 0, 0, 0)  # explode 1st slice (Java) for emphasis

plt.figure(figsize=(10, 7))
plt.pie(popularity, explode=explode, labels=programming_languages, colors=colors, autopct='%1.1f%%', shadow=True, startangle=140)
plt.axis('equal')  # Equal aspect ratio ensures that pie is drawn as a circle.
plt.show()
```



```python
#2 MathPlotLib
#2Create a scatter plot using matplotlib by comparing two subject marks of Maths and Science. Use marks given below.
#Sample data:
#math_marks = [88, 92, 80, 89, 100, 80, 60, 100, 80, 34]
#science_marks = [35, 79, 79, 48, 100, 88, 32, 45, 20, 30]
#marks_range = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100]

import matplotlib.pyplot as plt

#Data for Maths_marks, Science_marks & Marks_range
math_marks = [88, 92, 80, 89, 100, 80, 60, 100, 80, 34]
science_marks = [35, 79, 79, 48, 100, 88, 32, 45, 20, 30]
marks_range = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100]

#Create a scatter plot
plt.figure(figsize=(10, 5))
plt.scatter(marks_range, math_marks, label='Math Marks', color='red', s=50)
plt.scatter(marks_range, science_marks, label='Science Marks', color='green', s=50)
plt.xlabel('Marks Range')
plt.ylabel('Marks Scored')
plt.title('Scatter Plot')
plt.legend()
plt.grid(True)
plt.show()
```

Github Repo Link: https://github.com/Krypton0626/Bigdata/tree/main/ICP%204

YouTube Video Link: https://youtu.be/i2ojsa2xqMw