1 #Name-pawar vaishnavi

2 #PRN NO-202201090095

1. #Roll no-649
2. #Batch-F3

**Assignment5(a)**

import matplotlib.pyplot as plt

# initializing the data

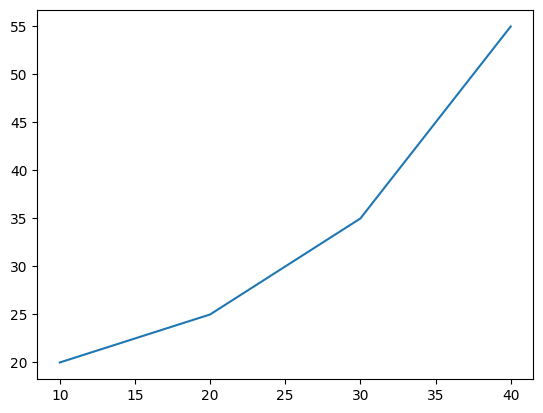
x = [10, 20, 30, 40]

y = [20, 25, 35, 55]

# plotting the data

plt.plot(x, y)

plt.show()



#Adding Title

#initilizing the data

x=[10,20,30,40]

y=[20,25,35,55]

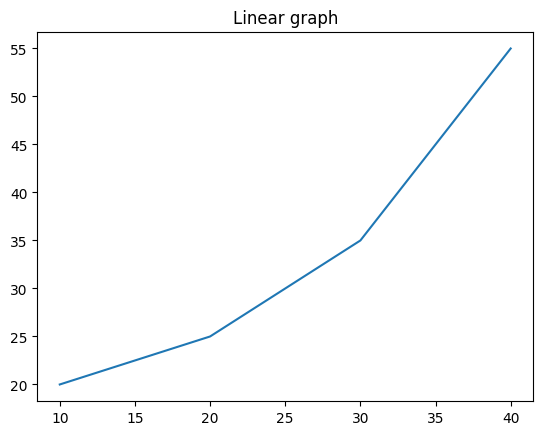
#plotting the data

plt.plot(x,y)

#Adding title to the plot

plt.title("Linear graph")

plt.show()



#initilizing the data

x=[10,20,30,40]

y=[20,25,35,55]

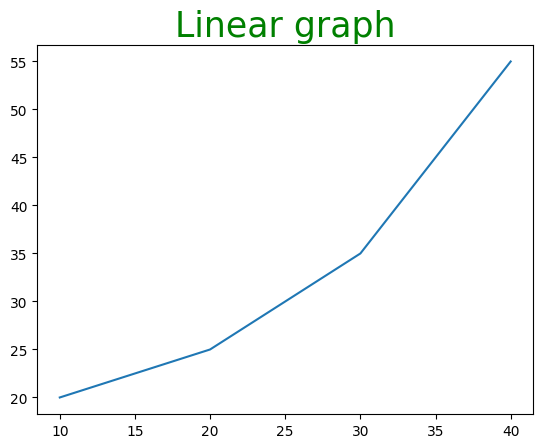
#plotting the data

plt.plot(x,y)

#Adding Title to plot

plt.title("Linear graph",fontsize=25,color="green")

plt.show()



#adding x lable and y lable

import matplotlib.pyplot as plt

#initilizing the data

x=[10,20,30,40]

y=[20,25,35,55]

#plotting the data

plt.plot(x,y)

#Adding Title to plot

plt.title("Linear graph",fontsize=25,color="green")

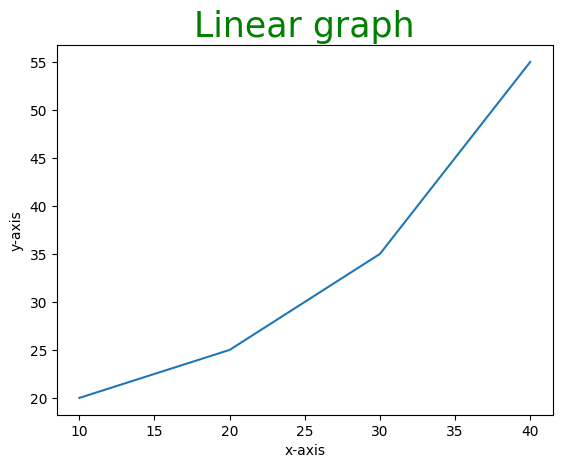
#adding label on the y-axis

plt.ylabel('y-axis')

#adding label on the x-axis

plt.xlabel('x-axis')

plt.show()

#plotting the data

plt.plot(x,y)

#Adding Title to plot

plt.title("Linear graph",fontsize=25,color="green")

#adding label on the y-axis

plt.ylabel('y-axis')

#adding label on the x-axis

plt.xlabel('x-axis')

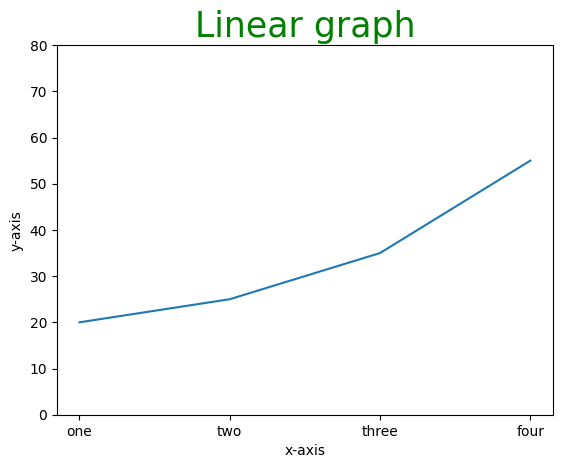
#setting the limit of y-axis

plt.ylim(0,80)

#setting the labels of x-axis

plt.xticks(x, labels=["one","two","three","four"])

plt.show()



#adding legends

import matplotlib.pyplot as plt

#initilizing the data

x=[10,20,30,40]

y=[20,25,35,55]

#plotting the data

plt.plot(x,y)

#Adding label on the x-axis

plt.title("Linear graph",fontsize=25,color="green")

#adding label on the x-axis

plt.xlabel('x-axis')

#setting the limit of y-axis

plt.ylim(0,80)

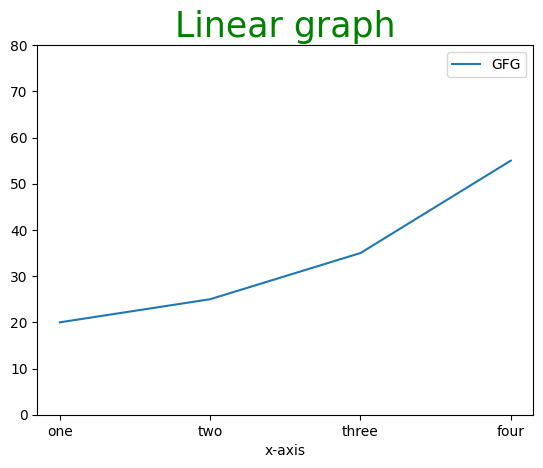
#setting the labels of x-axis

plt.xticks(x, labels=["one","two","three","four"])

#aDDING LEGENDS

plt.legend(["GFG"])

plt.show()



#figure calss

#python programm to show pyplot module

import matplotlib.pyplot as plt

from matplotlib.figure import Figure

#initilizing the data

x=[10,20,30,40]

y=[20,25,35,55]

fig=plt.figure(figsize=(7,5),facecolor='g',edgecolor='b',linewidth=7)

ax=fig.add\_axes([1,1,1,1])

ax.plot(x,y)

plt.title("Linear graph",fontsize=25,color="yellow")

#adding label on the x-axis

plt.xlabel('x-axis')

#setting the limit of y-axis

plt.ylim(0,80)

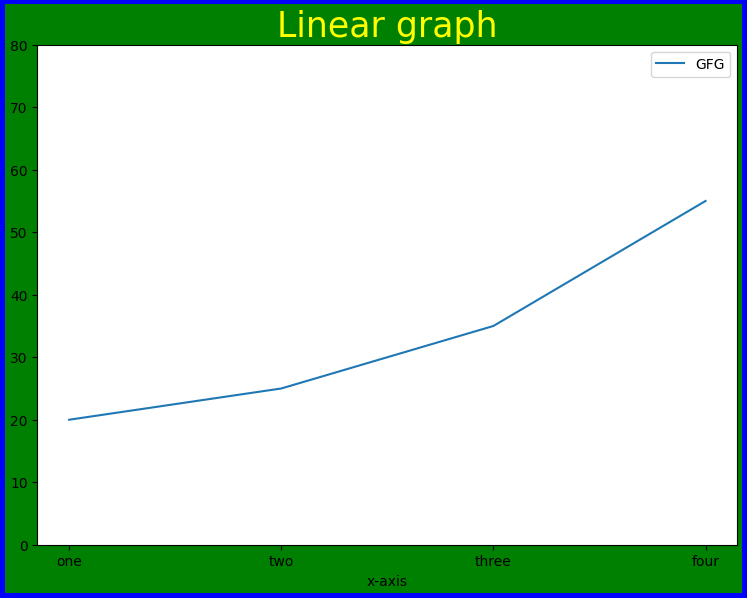
#setting the labels of x-axis

plt.xticks(x, labels=["one","two","three","four"])

#aDDING LEGENDS

plt.legend(["GFG"])

plt.show()



#Figure class

# Python program to show pyplot module

import matplotlib.pyplot as plt

from matplotlib.figure import Figure

# initializing the data

x = [10, 20, 30, 40]

y = [20, 25, 35, 55]

# Creating a new figure with width = 7 inches

# and height = 5 inches with face color as

# green, edgecolor as red and the line width

# of the edge as 7

fig = plt.figure(figsize =(7, 5), facecolor='g',

        edgecolor='b', linewidth=7)

# Creating a new axes for the figure

ax = fig.add\_axes([1, 1, 1, 1])

# Adding the data to be plotted

ax.plot(x, y)

# Adding title to the plot

plt.title("Linear graph", fontsize=25, color="yellow")

# Adding label on the y-axis

plt.ylabel('Y-Axis')

# Adding label on the x-axis

plt.xlabel('X-Axis')

# Setting the limit of y-axis

plt.ylim(0, 80)

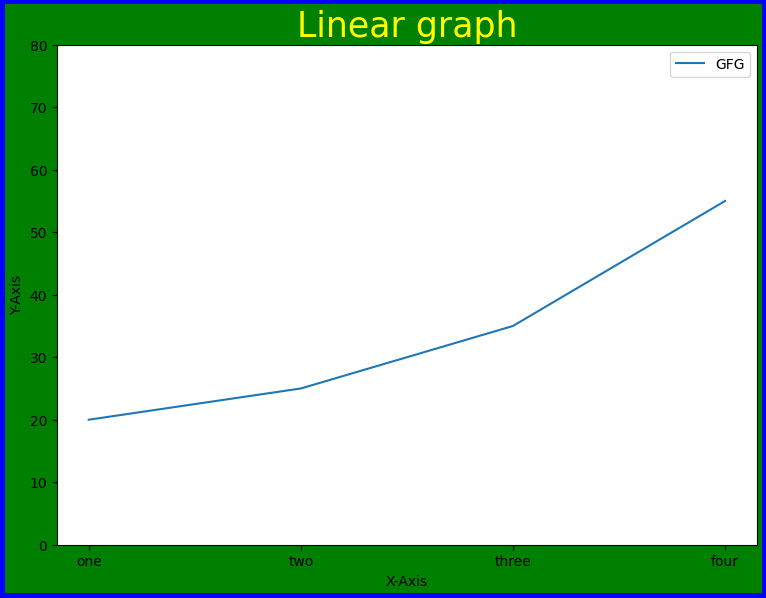
# setting the labels of x-axis

plt.xticks(x, labels=["one", "two", "three", "four"])

# Adding legends

plt.legend(["GFG"])

plt.show()



# Python program to show pyplot module

import matplotlib.pyplot as plt

from matplotlib.figure import Figure

# initializing the data

x = [10, 20, 30, 40]

y = [20, 25, 35, 55]

fig = plt.figure(figsize = (5, 4))

# Adding the axes to the figure

ax = fig.add\_axes([1, 1, 1, 1])

# plotting 1st dataset to the figure

ax1 = ax.plot(x, y)

# plotting 2nd dataset to the figure

ax2 = ax.plot(y, x)

# Setting Title

ax.set\_title("Linear Graph")

# Setting Label

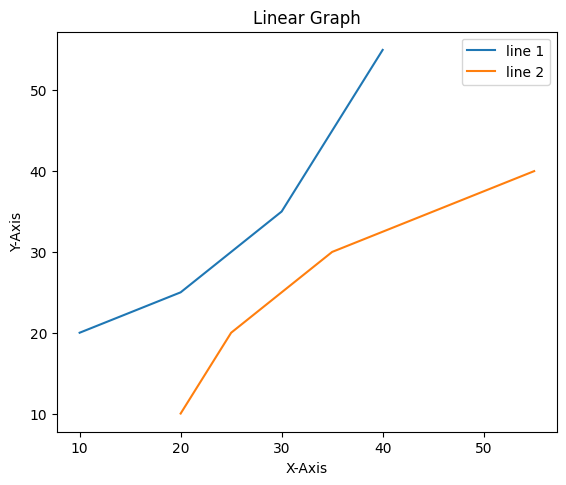
ax.set\_xlabel("X-Axis")

ax.set\_ylabel("Y-Axis")

# Adding Legend

ax.legend(labels = ('line 1', 'line 2'))

plt.show()



#Different line styles

import matplotlib.pyplot as plt

# initializing the data

x = [10, 20, 30, 40]

y = [20, 25, 35, 55]

# plotting the data

plt.plot(x, y, color='green', linewidth=3, marker='o',

    markersize=15, linestyle='--')

# Adding title to the plot

plt.title("Line Chart")

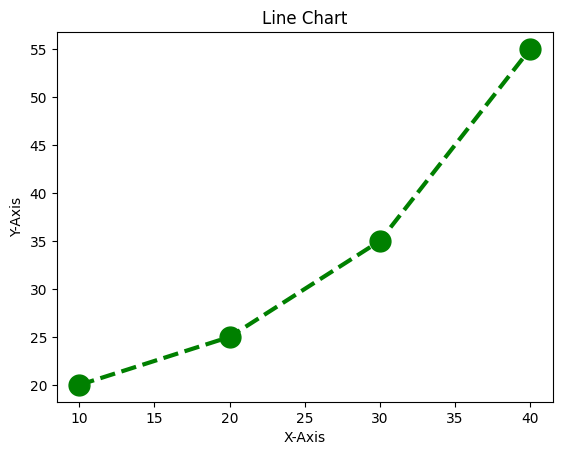
# Adding label on the y-axis

plt.ylabel('Y-Axis')

# Adding label on the x-axis

plt.xlabel('X-Axis')

plt.show()



#Multiple Plots

# Python program to show pyplot module

import matplotlib.pyplot as plt

from matplotlib.figure import Figure

# initializing the data

x = [10, 20, 30, 40]

y = [20, 25, 35, 55]

# Creating a new figure with width = 5 inches

# and height = 4 inches

fig = plt.figure(figsize =(5, 4))

# Creating first axes for the figure

ax1 = fig.add\_axes([0.1, 0.1, 0.8, 0.8])

# Creating second axes for the figure

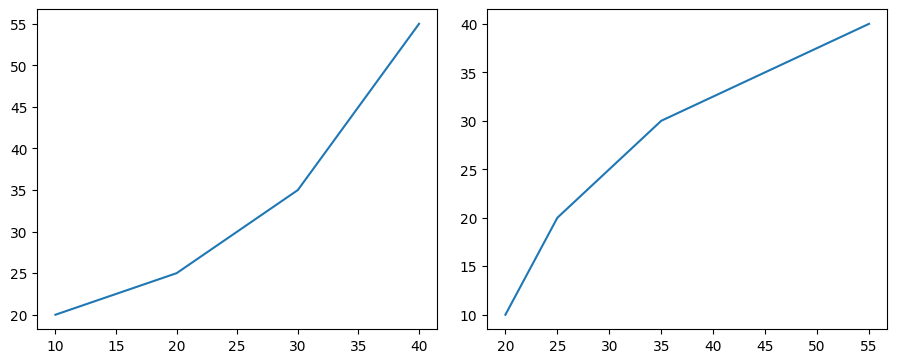
ax2 = fig.add\_axes([1, 0.1, 0.8, 0.8])

# Adding the data to be plotted

ax1.plot(x, y)

ax2.plot(y, x)

plt.show()



#Using subplot() method.

import matplotlib.pyplot as plt

# initializing the data

x = [10, 20, 30, 40]

y = [20, 25, 35, 55]

# Creating figure object

plt.figure()

# adding first subplot

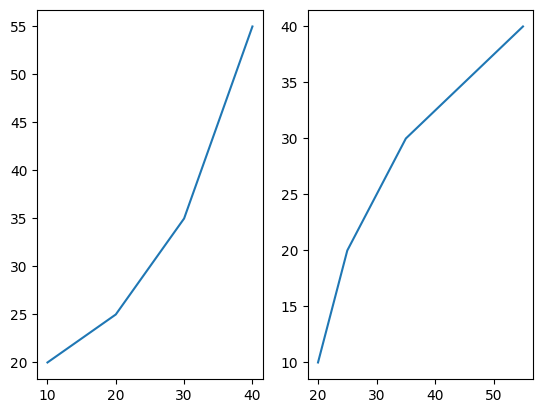
plt.subplot(121)

plt.plot(x, y)

# adding second subplot

plt.subplot(122)

plt.plot(y, x)



#bar chart

import matplotlib.pyplot as plt

import pandas as pd

# Reading the tips.csv file

data = pd.read\_csv('/content/tips1.csv')

# initializing the data

x = data['day']

y = data['total\_bill']

# plotting the data

plt.bar(x, y)

# Adding title to the plot

plt.title("Tips Dataset")

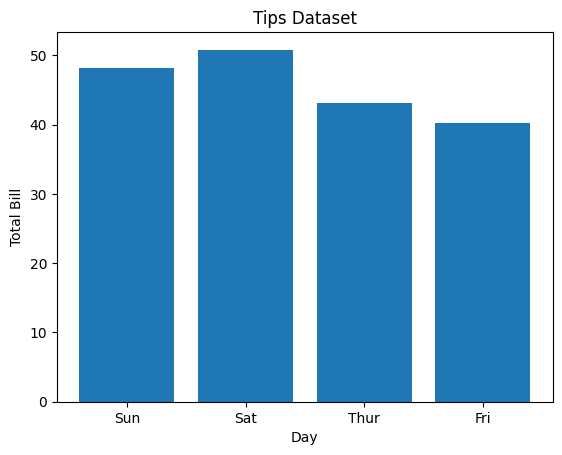
# Adding label on the y-axis

plt.ylabel('Total Bill')

# Adding label on the x-axis

plt.xlabel('Day')

plt.show()



**Assignment5(b)**

5

6 import pandas as pd 7 import numpy as np

1. import matplotlib.pyplot as plt
2. from pandas import Series, DataFrame 10

11

1. # Reading the tips.csv file
2. df1=pd.read\_csv('/content/tips.csv')

14

15 df1.head()



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **total\_bill** | **tip** | **sex** | **smoker** | **day** | **time** | **size** |
| **0** 16.99 | 1.01 | Female | No | Sun | Dinner | 2 |
| **1** 10.34 | 1.66 | Male | No | Sun | Dinner | 3 |
| **2** 21.01 | 3.50 | Male | No | Sun | Dinner | 3 |
| **3** 23.68 | 3.31 | Male | No | Sun | Dinner | 2 |
| **4** 24.59 | 3.61 | Female | No | Sun | Dinner | 4 |

1 df1.tail()



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **total\_bill tip** | **sex** | **smoker** | **day time** | **size** |
| **239** 29.03 5.92  1 df1.columns | Male | No | Sat Dinner | 3 |

Index(['total\_bill', 'tip', 'sex', 'smoker', 'day', 'time', 'size'], dtype='object')

**240** 27.18 2.00 Female Yes Sat Dinner 2

**241** 22.67 2.00 Male Yes Sat Dinner 2

**242** 17.82 1.75 Male No Sat Dinner 2

1 df1.info()

**243** 18.78 3.00 Female No Thur Dinner 2

<class 'pandas.core.frame.DataFrame'> RangeIndex: 244 entries, 0 to 243

Data columns (total 7 columns):

# Column Non-Null Count Dtype

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 0 |  | total\_bill | 244 | non-null |  | float64 |
| 1 |  | tip | 244 | non-null |  | float64 |
| 2 |  | sex | 244 | non-null |  | object |
| 3 |  | smoker | 244 | non-null |  | object |
| 4 |  | day | 244 | non-null |  | object |
| 5 |  | time | 244 | non-null |  | object |
| 6 |  | size | 244 | non-null |  | int64 |

dtypes: float64(2), int64(1), object(4) memory usage: 13.5+ KB

1 df1.describe()



|  |  |  |  |
| --- | --- | --- | --- |
|  | **total\_bill** | **tip** | **size** |
| **count** | 244.000000 | 244.000000 | 244.000000 |
| **mean** | 19.785943 | 2.998279 | 2.569672 |

1. a=pd.DataFrame(df1['day'].value\_counts())

**std** 8.902412 1.383638 0.951100

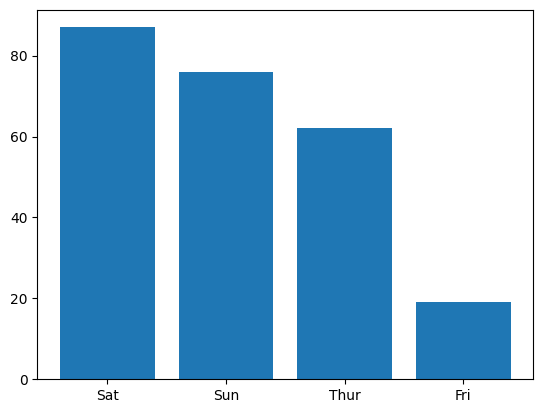
**min** 3.070000 1.000000 1.000000

**25%** 13.347500 2.000000 2.000000

**50%** 17.795000 2.900000 2.000000

**75%** 24.127500 3.562500 3.000000

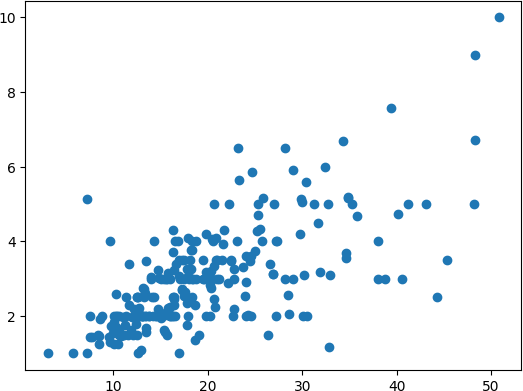
**max** 50.810000 10.000000 6.000000



1. a.reset\_index(inplace=True)
2. plt.bar(a['index'],a['day'])

<BarContainer object of 4 artists>

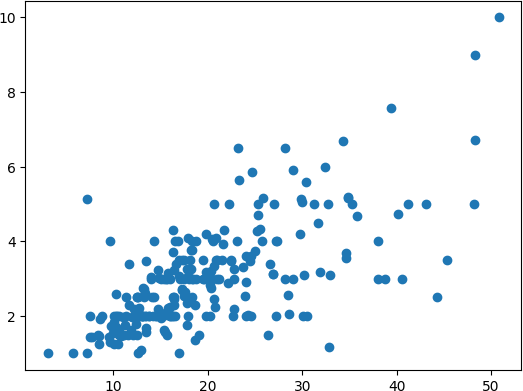
1. plt.scatter(df1['total\_bill'],df1['tip'])
2. plt.show()

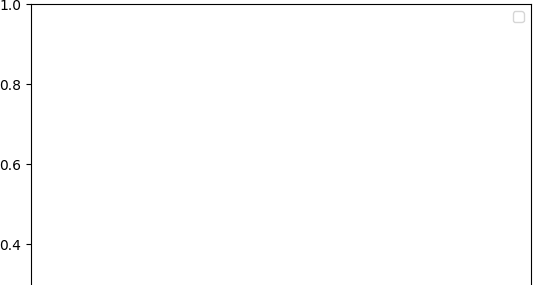


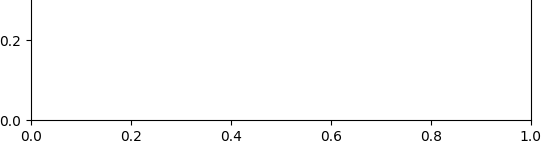
1. plt.scatter(x='total\_bill',y='tip',data=df1)
2. fig=plt.figure(figsize=(5,4))

3 ax=fig.add\_axes([1,1,1,1])

1. ax.legend(labels=('sun','mon','tue'))
2. plt.show()







1 #Different types of Matplotlib Plots 2 #bar chart

3 import matplotlib.pyplot as plt 4 import pandas as pd

5

1. # Reading the tips.csv file
2. data = pd.read\_csv('/content/tips.csv') 8

9 # initializing the data 10 x = data['day']

11 y = data['total\_bill'] 12

13 # plotting the data 14 plt.bar(x, y)

15

16 # Adding title to the plot 17 plt.title("Tips Dataset")

18

19 # Adding label on the y-axis 20 plt.ylabel('Total Bill')

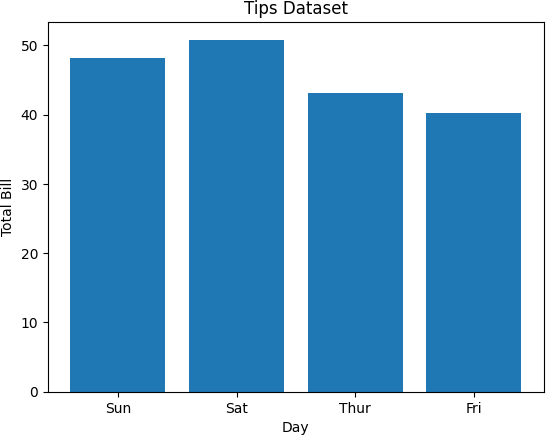
21

22 # Adding label on the x-axis 23 plt.xlabel('Day')

24

25 plt.show()

26



1 import matplotlib.pyplot as plt 2 import pandas as pd

3

4

5

6 # initializing the data 7 x = data['day']

8 y = data['total\_bill'] 9

1. # plotting the data
2. plt.bar(x, y, color='green', edgecolor='blue',
3. linewidth=2) 13

14 # Adding title to the plot 15 plt.title("Tips Dataset")

16

17 # Adding label on the y-axis 18 plt.ylabel('Total Bill')

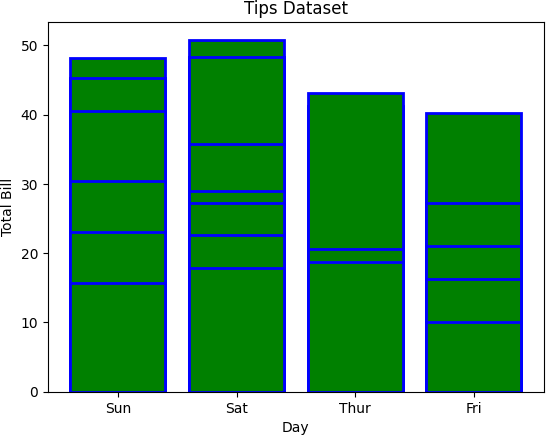
19

20 # Adding label on the x-axis 21 plt.xlabel('Day')

22

23 plt.show()

24



1 import matplotlib.pyplot as plt 2 import pandas as pd

3

4

5

6 # initializing the data 7 x = data['total\_bill'] 8

9 # plotting the data 10 plt.hist(x)

11

12 # Adding title to the plot 13 plt.title("Tips Dataset")

14

15 # Adding label on the y-axis 16 plt.ylabel('Frequency')

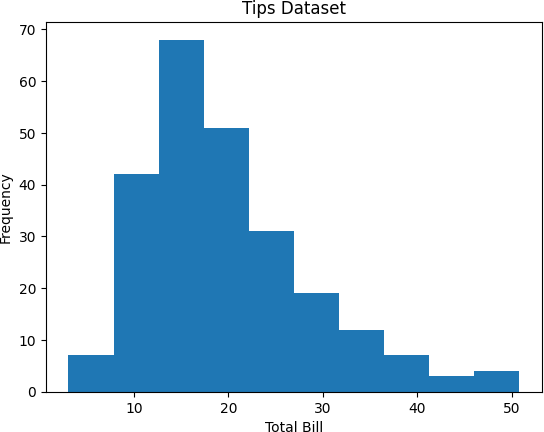
17

18 # Adding label on the x-axis 19 plt.xlabel('Total Bill')

20

21 plt.show()

22



1 import matplotlib.pyplot as plt 2 import pandas as pd

3

4

5

6 # initializing the data 7 x = data['total\_bill'] 8

1. # plotting the data
2. plt.hist(x, bins=25, color='green', edgecolor='blue',
3. linestyle='--', alpha=0.5) 12

13 # Adding title to the plot 14 plt.title("Tips Dataset")

15

16 # Adding label on the y-axis 17 plt.ylabel('Frequency')

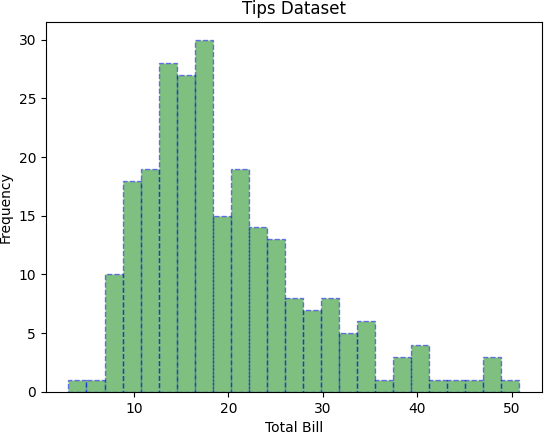
18

19 # Adding label on the x-axis 20 plt.xlabel('Total Bill')

21

22 plt.show()

23



1 import matplotlib.pyplot as plt 2 import pandas as pd

3

4

5

6 # initializing the data 7 x = data['day']

8 y = data['total\_bill'] 9

10 # plotting the data 11 plt.scatter(x, y)

12

13 # Adding title to the plot 14 plt.title("Tips Dataset")

15

16 # Adding label on the y-axis 17 plt.ylabel('Total Bill')

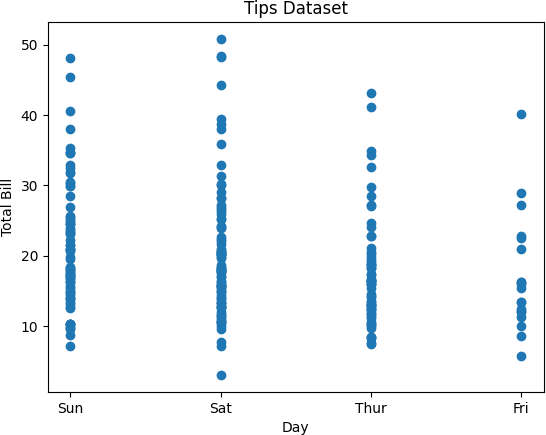
18

19 # Adding label on the x-axis 20 plt.xlabel('Day')

21

22 plt.show()

23



1 import matplotlib.pyplot as plt 2 import pandas as pd

3

4

5 # initializing the data 6 x = data['day']

7 y = data['total\_bill'] 8

1. # plotting the data
2. plt.scatter(x, y, c=data['size'], s=data['total\_bill'],
3. marker='D', alpha=0.5) 12

13 # Adding title to the plot 14 plt.title("Tips Dataset")

15

16 # Adding label on the y-axis 17 plt.ylabel('Total Bill')

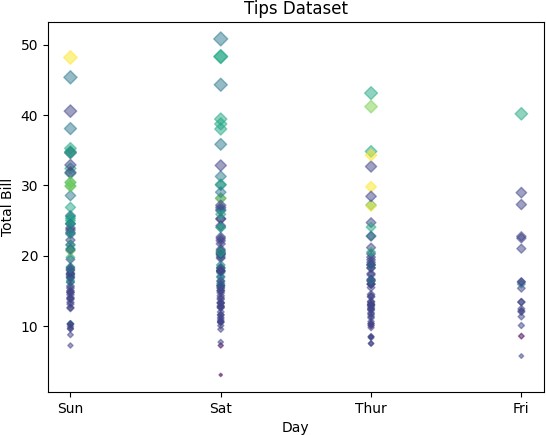
18

19 # Adding label on the x-axis 20 plt.xlabel('Day')

21

22 plt.show()

23



1 import matplotlib.pyplot as plt 2 import pandas as pd

3

4

1. # initializing the data
2. cars = ['AUDI', 'BMW', 'FORD',
3. 'TESLA', 'JAGUAR',]

8 data = [23, 10, 35, 15, 12]

9

1. # plotting the data
2. plt.pie(data, labels=cars)

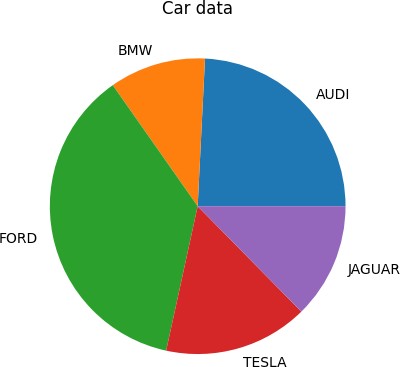
12

13 # Adding title to the plot 14 plt.title("Car data")

15

16 plt.show()

17



1 import matplotlib.pyplot as plt 2 import pandas as pd

3

1. # initializing the data
2. cars = ['AUDI', 'BMW', 'FORD',
3. 'TESLA', 'JAGUAR',]

7 data = [23, 13, 35, 15, 12]

8

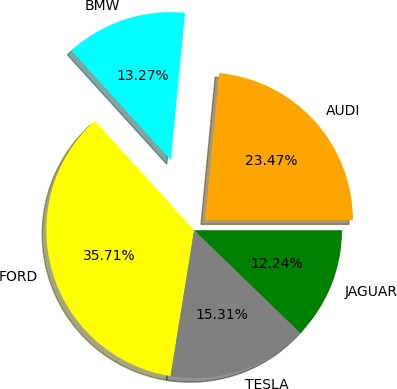
9 explode = [0.1, 0.5, 0, 0, 0]

10

1. colors = ( "orange", "cyan", "yellow",
2. "grey", "green",) 13
3. # plotting the data
4. plt.pie(data, labels=cars, explode=explode, autopct='%1.2f%%',
5. colors=colors, shadow=True) 17

18 plt.show()

19



1 import matplotlib.pyplot as plt 2

3 # Creating data

4 year = ['2010', '2002', '2004', '2006', '2008']

5 production = [25, 15, 35, 30, 10]

6

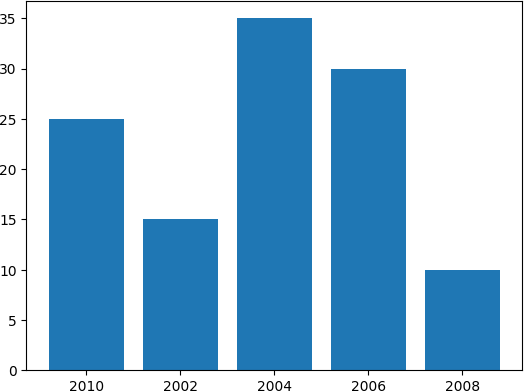
1. # Plotting barchart
2. plt.bar(year, production)

9

1. # Saving the figure.
2. plt.savefig("output.jpg")

12

1. # Saving figure by changing parameter values
2. plt.savefig("output1", facecolor='y', bbox\_inches="tight",
3. pad\_inches=0.3, transparent=True) 16



[Colab paid products](https://colab.research.google.com/signup?utm_source=footer&utm_medium=link&utm_campaign=footer_links) - [Cancel contracts here](https://colab.research.google.com/cancel-subscription) 0s completed at 2:50 PM