Name : Sarthak Pakhare

PRN : 202201090082

Roll no : 646

Div : F

Batch : F3

```
[ ] from google.colab import drive
    drive.mount('/content/drive')
```

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

```
[ ] import pandas as pd
    import numpy as np
    import matplotlib.pyplot as plt
    from pandas import Series, DataFrame
```

```
[ ] # Reading the tips.csv file
    df1=pd.read_csv('/content/drive/MyDrive/Colab Notebooks/tips.csv')

    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

```
[ ] df1.tail()
```

	total_bill	tip	sex	smoker	day	time	size
239	29.03	5.92	Male	No	Sat	Dinner	3
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
243	18.78	3.00	Female	No	Thur	Dinner	2

```
[ ] df1.columns
```

```
Index(['total_bill', 'tip', 'sex', 'smoker', 'day', 'time', 'size'], dtype='object')
```

```
[ ] df1.info()
```

```
<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
```

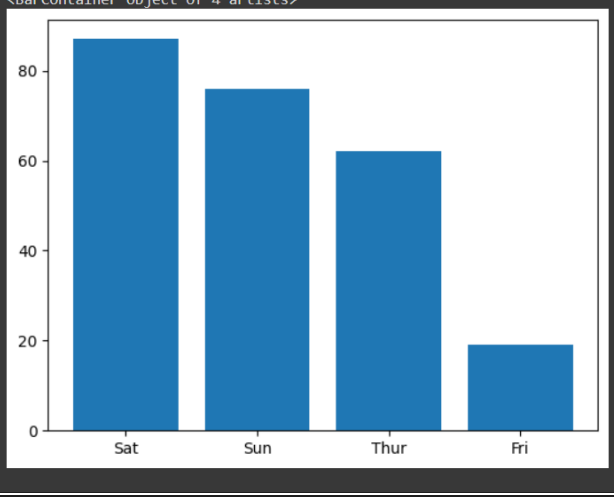
```
[ ] df1.describe()
```

	total_bill	tip	size
count	244.000000	244.000000	244.000000
mean	19.785943	2.998279	2.569672
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

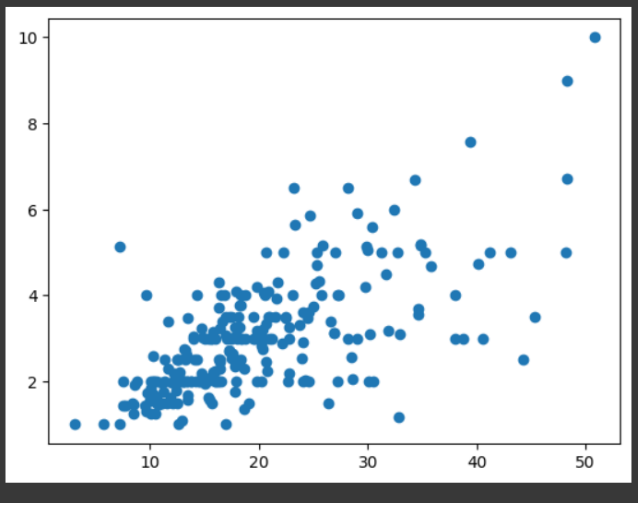
```
[ ] a=pd.DataFrame(df1['day'].value_counts())
    a.reset_index(inplace=True)
    plt.bar(a['index'],a['day'])
```

```
[ ] a=pd.DataFrame(df1['day'].value_counts())
    a.reset_index(inplace=True)
    plt.bar(a['index'],a['day'])
```

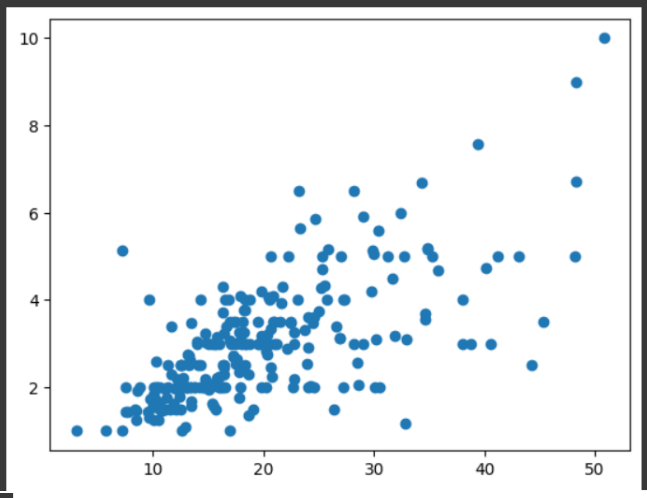
```
<BarContainer object of 4 artists>
```

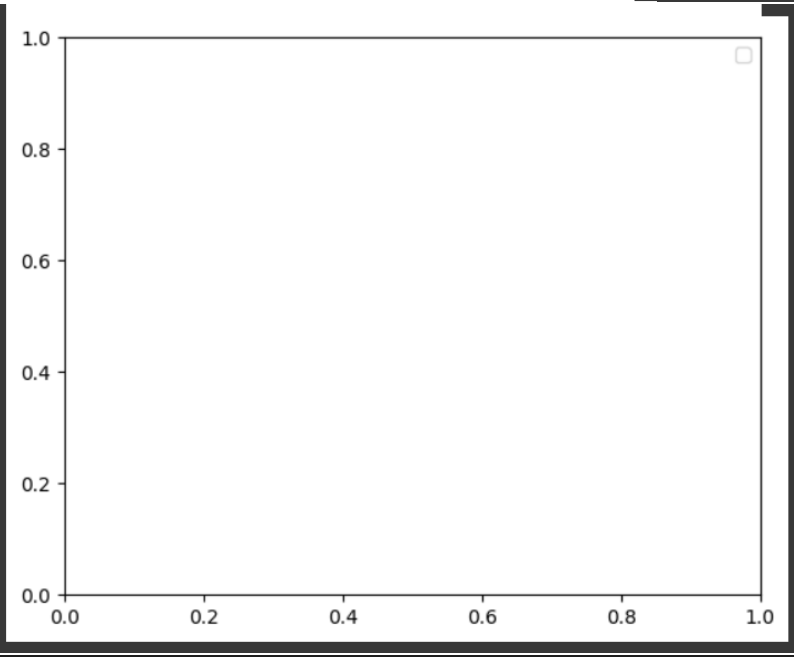



```
plt.scatter(df1['total_bill'],df1['tip'])
plt.show()
```

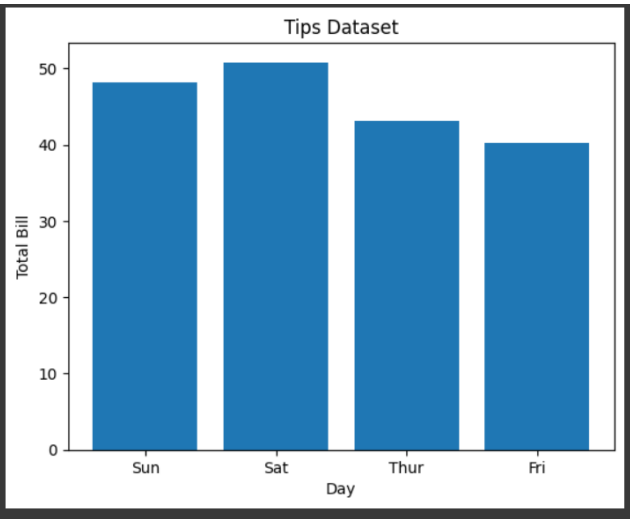


```
plt.scatter(x='total_bill',y='tip',data=df1)
fig=plt.figure(figsize=(5,4))
ax=fig.add_axes([1,1,1,1])
ax.legend(labels=('sun','mon','tue'))
plt.show()
```

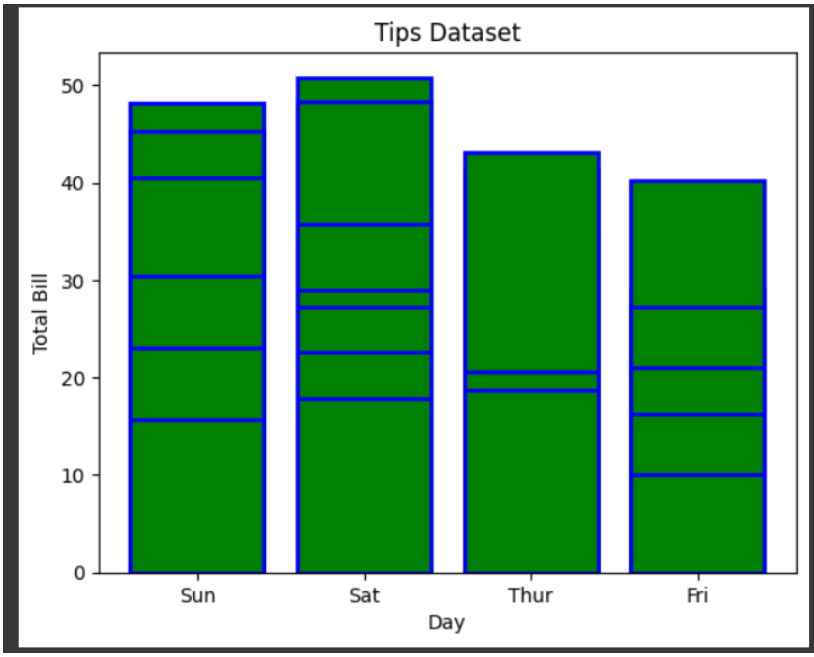




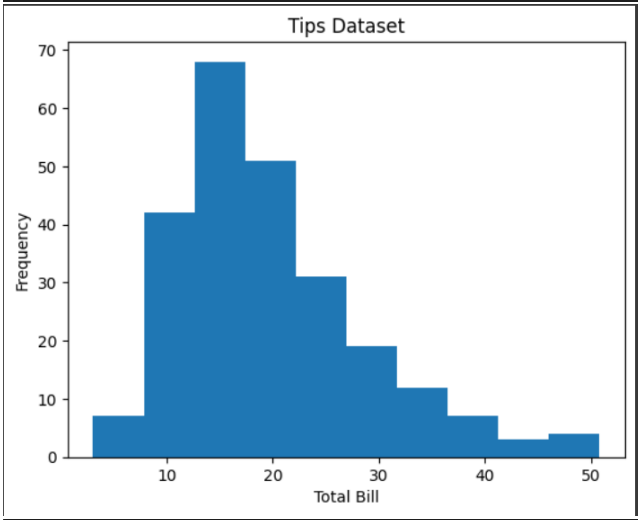

```
#Different types of Matplotlib Plots
#bar chart
import matplotlib.pyplot as plt
import pandas as pd

# Reading the tips.csv file
data = pd.read_csv('/content/drive/MyDrive/Colab Notebooks/tips.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()
```



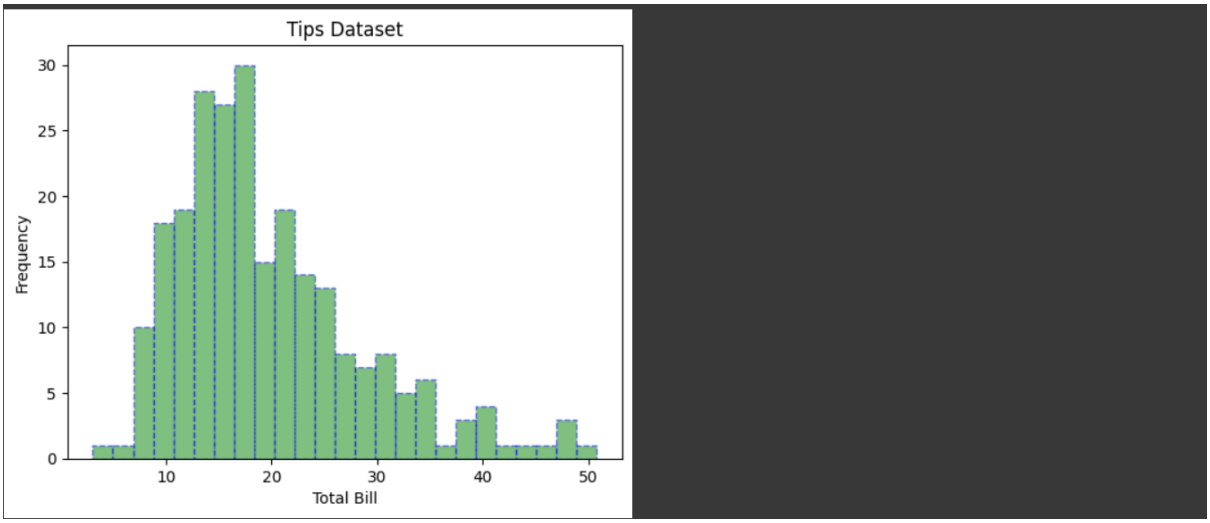

```
[ ]  import matplotlib.pyplot as plt
     import pandas as pd



     # initializing the data
     x = data['day']
     y = data['total_bill']

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

     # 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()
```



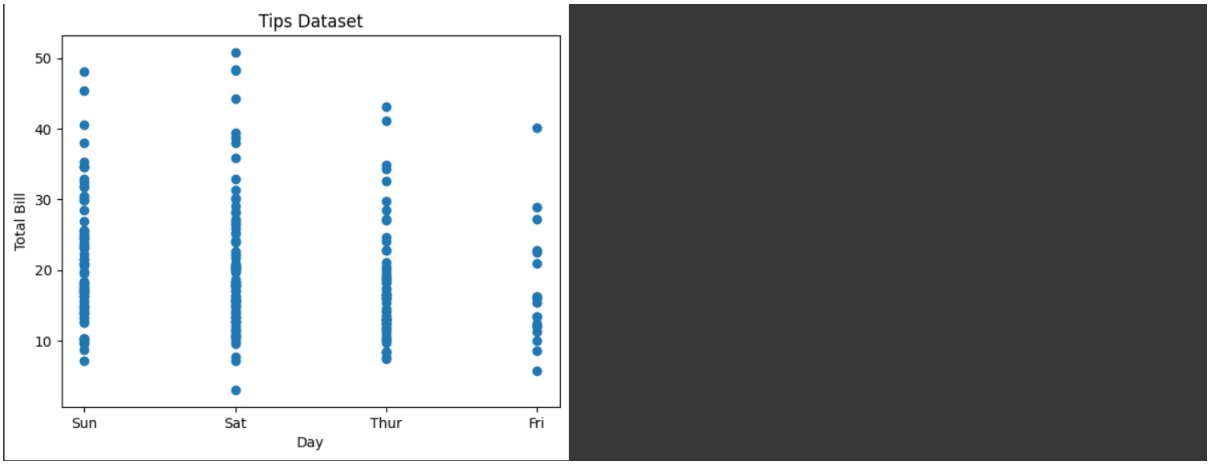
```python
import matplotlib.pyplot as plt
import pandas as pd


# initializing the data
x = data['total_bill']

# plotting the data
plt.hist(x)

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

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

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

plt.show()
```



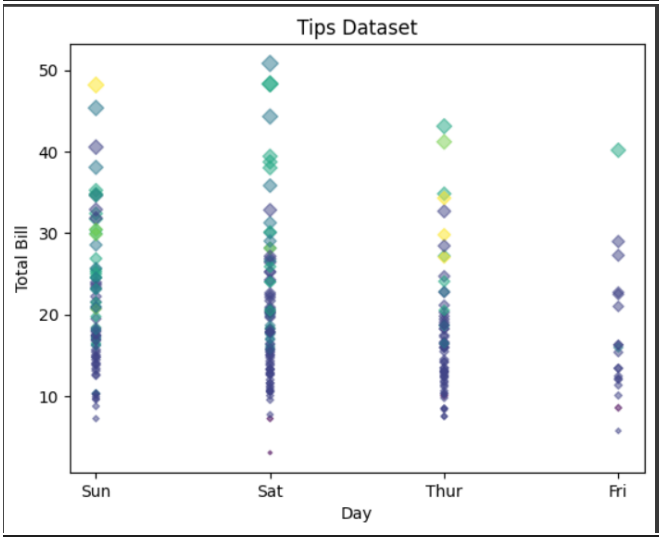

```python
import matplotlib.pyplot as plt
import pandas as pd


# initializing the data
x = data['total_bill']

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

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

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

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

plt.show()
```



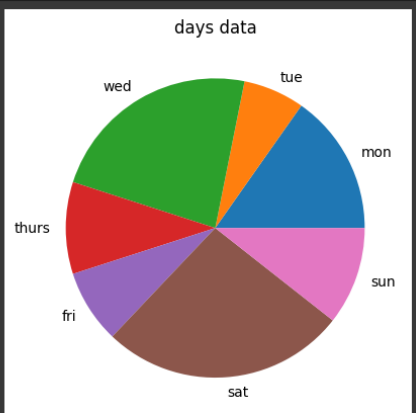

```
import matplotlib.pyplot as plt
import pandas as pd



# initializing the data
x = data['day']
y = data['total_bill']

# plotting the data
plt.scatter(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()
```



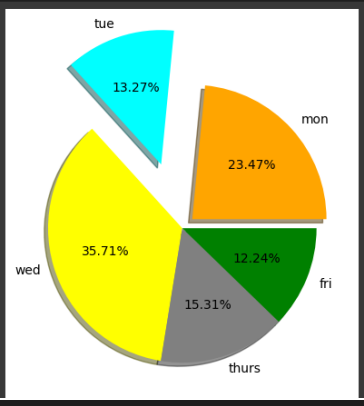
```python
import matplotlib.pyplot as plt
import pandas as pd


# initializing the data
x = data['day']
y = data['total_bill']

# plotting the data
plt.scatter(x, y, c=data['size'], s=data['total_bill'],
      marker='D', alpha=0.5)

# 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()
```



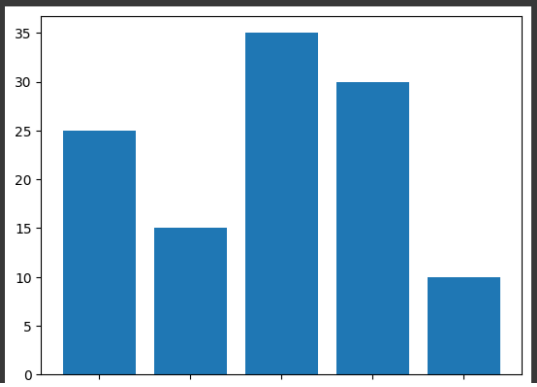

```python
import matplotlib.pyplot as plt
import pandas as pd


# initializing the data
day = ['mon', 'tue', 'wed',
    'thurs', 'fri', 'sat', 'sun']
data = [23, 10, 35, 15, 12, 40, 16]

# plotting the data
plt.pie(data, labels=day)

# Adding title to the plot
plt.title("days data")

plt.show()
```



```
# initializing the data
days = ['mon', 'tue', 'wed',
    'thurs', 'fri',]
data = [23, 13, 35, 15, 12]

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

colors = ( "orange", "cyan", "yellow",
    "grey", "green",)

# plotting the data
plt.pie(data, labels=days, explode=explode, autopct='%1.2f%%',
    colors=colors, shadow=True)

plt.show()
```




```
import matplotlib.pyplot as plt

# Creating data
year = ['sat', 'sun', 'thurs', 'mon', 'tue']
production = [25, 15, 35, 30, 10]

# Plotting barchart
plt.bar(year, production)

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

# Saving figure by changing parameter values
plt.savefig("output1", facecolor='y', bbox_inches="tight",
      pad_inches=0.3, transparent=True)
```



```
x = np.linspace(0, 10, 100)
fig = plt.figure()
plt.plot(x, np.sin(x))
plt.plot(x, np.cos(x))
fig.savefig('graph1.png')
```