```
pip install numpy
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: numpy in /usr/local/lib/python3.7/dist-packages (1.21.6)
```

```
import numpy
import numpy as np
from numpy import *
```

```
a=np.array([1,2,3,4,5])
a
```

```
array([1, 2, 3, 4, 5])
```

```
type(a)
```

```
numpy.ndarray
```

```
a=np.array([1,2,3,4,5],dtype=float)
a
```

```
array([1., 2., 3., 4., 5.])
```

```
a.ndim
```

```
1
```

```
a=np.array([1,2,3,4,5],ndmin=2)
a
```

```
array([[1, 2, 3, 4, 5]])
```



```
2
```

```
ar=np.arange(10)
ar
```

```
array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
```

```
ar1=np.arange(2,21,2)
ar1
```

```
array([ 2,  4,  6,  8, 10, 12, 14, 16, 18, 20])
```

```
ar1.ndim
```

```
1
```

```
ar1.shape
```

```
(10,)
```

```
ar1.reshape(2,5)
```

```
array([[ 2,  4,  6,  8, 10],
       [12, 14, 16, 18, 20]])
```

```
arr=ar1.reshape(5,2)
```

```
array([[ 2,  4],
       [ 6,  8],
       [10, 12],
       [14, 16],
       [18, 20]])
```

```
arr=ar1.reshape(5,2)
```

```
arr
```

```
array([[ 2,  4],
       [ 6,  8],
       [10, 12],
       [14, 16],
       [18, 20]])
```

```
arr.min()
```

```
2
```

```
arr.max()
```

```
20
```

```
arr.sum()
```

```
110
```

```
arr.mean()
```

```
11.0
```

Saved successfully!

```
np
```

```
<module 'numpy' from '/usr/local/lib/python3.7/dist-packages/numpy/__init__.py'>
```

```
np.std(arr)
```

```
5.744562646538029
```

```
np.var(arr)
```

```
33.0
```

```
arr1 = np.linspace(0,2,10)
arr1
```

```
array([0.        , 0.22222222, 0.44444444, 0.66666667, 0.88888889,
       1.11111111, 1.33333333, 1.55555556, 1.77777778, 2.        ])
```

```
np.cos(arr1)
```

```
array([ 1.        ,  0.97541009,  0.90284967,  0.78588726,  0.63027505,
        0.44366602,  0.23523757,  0.01524018, -0.20550672, -0.41614684])
```

```
np.sin(arr1)
```

```
array([0.        , 0.22039774, 0.42995636, 0.6183698 , 0.77637192,
       0.8961922 , 0.9719379 , 0.99988386, 0.9786557 , 0.90929743])
```

```
np.tan(arr1)
```

```
array([ 0.        ,  0.22595393,  0.47622143,  0.78684289,  1.23179859,
        2.01997033,  4.13172899, 65.60839711, -4.76215913, -2.18503986])
```

```
a = np.arange(25)
a
```

```
array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
       17, 18, 19, 20, 21, 22, 23, 24])
```

```
arr2 = a.reshape(5,5)
```

```
arr2
```

```
array([[ 0,  1,  2,  3,  4],
       [ 5,  6,  7,  8,  9],
       [10, 11, 12, 13, 14],
       [15, 16, 17, 18, 19],
       [20, 21, 22, 23, 24]])
```

```
np.transpose(arr2)
```

```
array([[ 0,  5, 10, 15, 20],
       [ 1,  6, 11, 16, 21],
       [ 2,  7, 12, 17, 22],
       [ 3,  8, 13, 18, 23],
       [ 4,  9, 14, 19, 24]])
```

```
arr2.T
```

Saved successfully!


```
       [ 3,  8, 13, 18, 23],
       [ 4,  9, 14, 19, 24]])
```


```
arr2.sum(axis=1)
```

```
array([ 10,  35,  60,  85, 110])
```

```
np.zeros(5)
```

```
array([0., 0., 0., 0., 0.])
```

```
arr=np.arange(4).reshape(2,2)
arr
```

```
array([[0, 1],
       [2, 3]])
```

```
arr.sum(axis=0)
```

```
array([2, 4])
```

```
np.zeros(5)
```

```
array([0., 0., 0., 0., 0.])
```

```
np.zeros(shape =(4,3))
```

```
array([[0., 0., 0.],
       [0., 0., 0.],
       [0., 0., 0.],
       [0., 0., 0.]])
```

```
np.ones(5)
```

```
array([1., 1., 1., 1., 1.])
```

```
np.ones(shape =(4,3),dtype =int)
```

```
array([[1, 1, 1],
       [1, 1, 1],
       [1, 1, 1],
       [1, 1, 1]])
```

```
#random
#np.random.rand()
#np.random.randint()
#np.random.randn()
```

```
np.random.rand(4)
```

```
array([0.18492428, 0.32040414, 0.74245543, 0.12159996])
```

```
np.random.rand(4,5)
```

```
array([[0.61057899, 0.28089952, 0.74278482, 0.09924166, 0.34850802],
       [0.45660501, 0.1453472 , 0.27056664, 0.42694231, 0.01533883],
       [0.74877198, 0.61671391, 0.17716643, 0.87721779, 0.45507442],
       [0.67192353, 0.97120731, 0.76756095, 0.70174663, 0.95839242]])
```

```
np.random.randn(4,5)
```

Saved successfully!

```
98, -0.32837305, -0.40465527,  1.25006948],
31, -1.2625048 ,  0.81841582, -0.83332397],
[ 0.00749001, -0.46221233, -1.38495412,  1.91556716, -0.39199923],
[ 1.7047149 ,  0.86239297, -0.69292531, -1.04664816,  0.12680992]])
```

```
np.random.randint(4,40,7)
```

```
array([ 4, 39, 39, 37, 30, 24, 20])
```

```
arr=np.arange(8)
arr
```

```
array([0, 1, 2, 3, 4, 5, 6, 7])
```

```
arr.ndim
```

```
1
```

```
arr.shape
```

```
(8,)
```

```
arr[1]
```

```
1
```

```
arr[-1]
```

```
7
```

```
arr[1:4]
```

```
array([1, 2, 3])
```

```
arr[-5:-1]
```

```
array([3, 4, 5, 6])
```

```
arr2
```

```
array([[ 0,  1,  2,  3,  4],
       [ 5,  6,  7,  8,  9],
       [10, 11, 12, 13, 14],
       [15, 16, 17, 18, 19],
       [20, 21, 22, 23, 24]])
```

```
arr2[3][2]
```

```
17
```

pandas

```
pip install pandas
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: pandas in /usr/local/lib/python3.7/dist-packages (1.3.5)
Requirement already satisfied: python-dateutil>=2.7.3 in /usr/local/lib/python3.7/dist-packages (from panda
Requirement already satisfied: numpy>=1.17.3 in /usr/local/lib/python3.7/dist-packages (from pandas) (1.21.
Requirement already satisfied: pytz>=2017.3 in /usr/local/lib/python3.7/dist-packages (from pandas) (2022.4
six>=1.5 in /usr/local/lib/python3.7/dist-packages (from python-dateutil>=2.
```

Saved successfully!

```
import pandas as pd
```

```
a=pd.Series([1,2,3,4,5])
```

```
a
```

```
0    1
1    2
2    3
3    4
4    5
dtype: int64
```

```
a=pd.Series([1,2,3,4,5],index=[101,102,103,104,105])
```

```
a
```

```
101    1
102    2
103    3
104    4
105    5
dtype: int64
```

```
a.index
```

```
Int64Index([101, 102, 103, 104, 105], dtype='int64')
```

```
a.values
```

```
array([1, 2, 3, 4, 5])
```

```
a=np.arange(6)
b=pd.Series(a)
b
```

```
0    0
1    1
2    2
3    3
4    4
5    5
dtype: int64
```

```
type(b)
```

```
pandas.core.series.Series
```

```
population_dict={"Brampton":234567,
                 "calgary":34567,
                 "Montreal":37687,
                 "vancouver":87654,
                 "toronto":987623}
```

```
type(population_dict)
```

```
dict
```

```
population = pd.Series(population_dict)
```

Saved successfully!

```
Brampton     234567
calgary       34567
Montreal      37687
vancouver     87654
toronto      987623
dtype: int64
```

```
type(population)
```

```
pandas.core.series.Series
```

```
population.index
```

```
Index(['Brampton', 'calgary', 'Montreal', 'vancouver', 'toronto'], dtype='object')
```

```
population["Calgary"]=987654
```

```
population
```

```
Brampton     234567
calgary       34567
Montreal      37687
vancouver     87654
toronto      987623
Calgary      987654
dtype: int64
```

```
area_dict={"Brampton":234567,
"calgary":34567,
"Montreal":37687,
"vancouver":87654,
"toronto":987623,
"Calgary":987654}
```

```
area = pd.Series(area_dict)
area
```

```
Brampton     234567
calgary       34567
Montreal      37687
vancouver     87654
toronto      987623
Calgary      987654
dtype: int64
```

```
a=[1,2,3,4,5,6]
df=pd.DataFrame(a,index=["a","b","c","d","e","f"])
df
```

	0
a	1
b	2
c	3
d	4
e	5
f	6

Saved successfully!

```
thamya",35],["bimla",80],["deepthi",12]]
"Age"],dtype=float)
```

```
/usr/local/lib/python3.7/dist-packages/IPython/core/interactiveshell.py:3326: FutureWarning: Could not cast
  exec(code_obj, self.user_global_ns, self.user_ns)
```

	Name	Age
0	varsha	22.0
1	ragavi	32.0
2	rithamya	35.0
3	bimla	80.0
4	deepthi	12.0

```
population
```

```
Brampton     234567
calgary       34567
Montreal      37687
vancouver     87654
toronto      987623
Calgary      987654
dtype: int64
```

```
area
```

```
Brampton     234567
```

```
calgary       34567
Montreal      37687
vancouver     87654
toronto      987623
Calgary      987654
dtype: int64
```

```
state = pd.DataFrame({"population":population,"Area":area})
state
```

	population	Area
Brampton	234567	234567
calgary	34567	34567
Montreal	37687	37687
vancouver	87654	87654
toronto	987623	987623
Calgary	987654	987654

```
state.count()
```

```
population    6
Area          6
dtype: int64
```

```
state.sum()
```

```
population    2369752
Area          2369752
dtype: int64
```

Saved successfully!

```
Area          394958.666667
dtype: float64
```

```
state.max()
```

```
population    987654
Area          987654
dtype: int64
```

```
state.to_csv("state.csv")
```

```
data=pd.read_csv("state.csv")
```

```
data
```

	Unnamed: 0	population	Area
0	Brampton	234567	234567
1	calgary	34567	34567
2	Montreal	37687	37687
3	vancouver	87654	87654
4	toronto	987623	987623
5	Calgary	987654	987654

```
data.describe()
```

	population	Area
count	6.000000	6.000000
mean	394958.666667	394958.666667
std	464806.543039	464806.543039
min	34567.000000	34567.000000
25%	50178.750000	50178.750000
50%	161110.500000	161110.500000
75%	799359.000000	799359.000000
max	987654.000000	987654.000000

```
data.isnull().any()
```

```
Unnamed: 0    False
population    False
Area          False
dtype: bool
```

```
data.isnull().sum()
```

```
Unnamed: 0    0
population    0
Area          0
dtype: int64
```

```
d=pd.Series(["a","b","c","d"],index=[1,3,5,7])
d
```

Saved successfully!

```
5    c
7    d
dtype: object
```

```
d[1]
```

```
'a'
```

```
d[1:3]
```

```
3    b
5    c
dtype: object
```

```
d.loc[1:3]
```

```
1    a
3    b
dtype: object
```

```
d.iloc[1:3]
```

```
3    b
5    c
dtype: object
```

Matplotlib

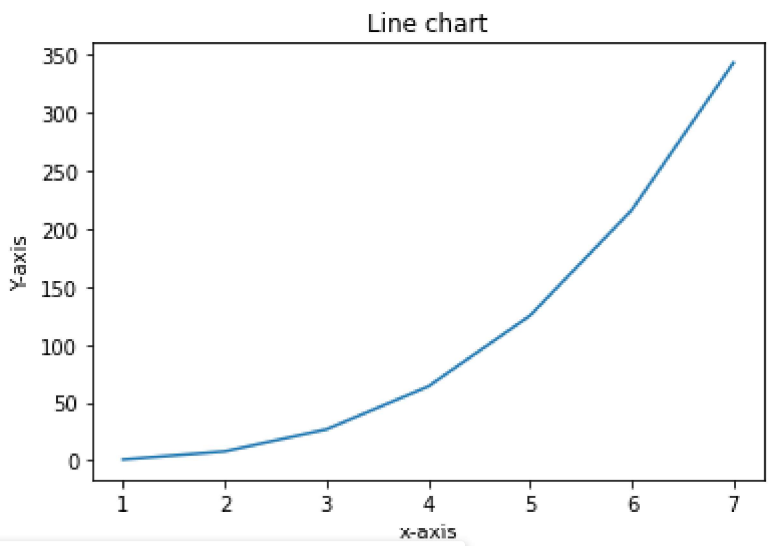
```
from matplotlib import pyplot as plt


x=np.array([1,2,3,4,5,6,7])
y=np.power(x,3)
y
```

```
array([  1,   8,  27,  64, 125, 216, 343])
```

```
x=np.array([1,2,3,4,5,6,7])
y=np.power(x,3)
plt.plot(x,y)
plt.xlabel("x-axis")
plt.ylabel("Y-axis")
plt.title("Line·chart")
```

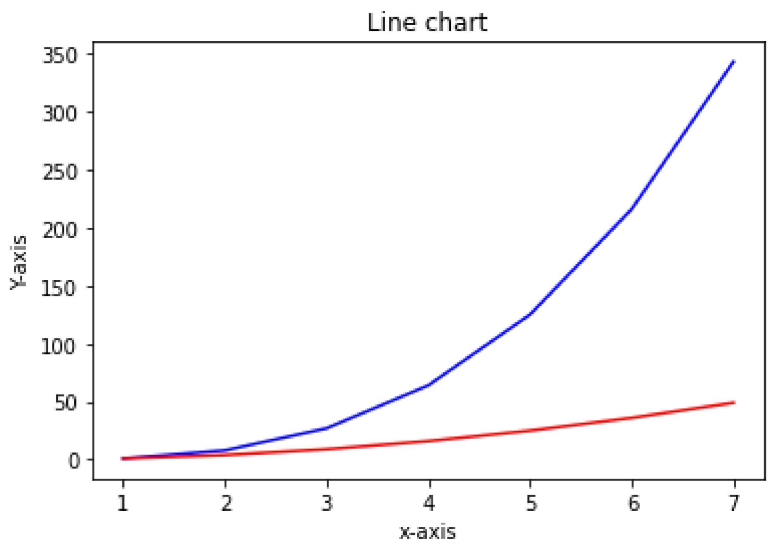
```
Text(0.5, 1.0, 'Line chart')
```



Saved successfully!

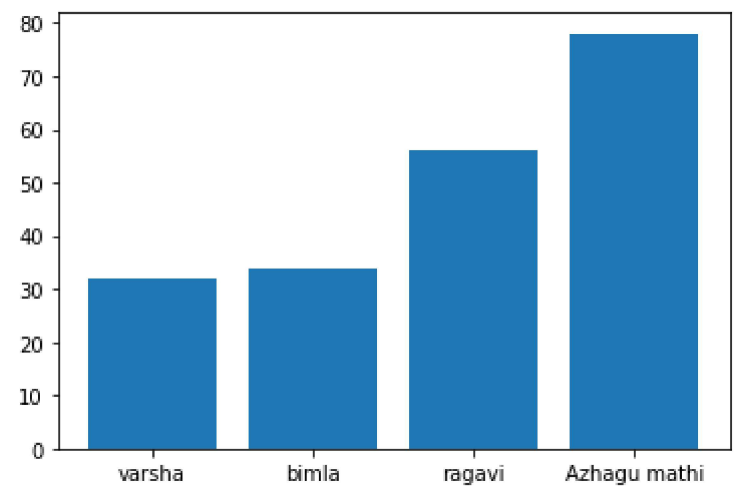
```
x=np.array([1,2,3,4,5,6,7])
y1=np.power(x,3)
y2=np.power(x,2)
plt.plot(x,y1,'b',label="cube")
plt.plot(x,y2,"r",label="Square")
plt.xlabel("x-axis")
plt.ylabel("Y-axis")
plt.title("Line chart")
```

```
Text(0.5, 1.0, 'Line chart')
```



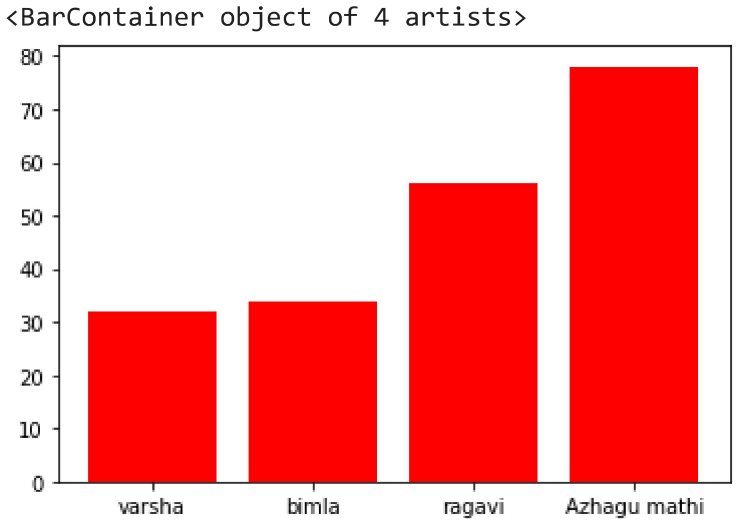

```
age=[32,34,56,78]
name=["varsha","bimla","ragavi","Azhagu mathi"]
plt.bar(name,age)
```

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




```
age=[32,34,56,78]
name=["varsha","bimla","ragavi","Azhagu mathi"]
plt.bar(name,age,color="red")
```

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



Saved successfully!

Seaborn

```
pip install seaborn
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: seaborn in /usr/local/lib/python3.7/dist-packages (0.11.2)
Requirement already satisfied: numpy>=1.15 in /usr/local/lib/python3.7/dist-packages (from seaborn) (1.21.6
Requirement already satisfied: pandas>=0.23 in /usr/local/lib/python3.7/dist-packages (from seaborn) (1.3.5
Requirement already satisfied: scipy>=1.0 in /usr/local/lib/python3.7/dist-packages (from seaborn) (1.7.3)
Requirement already satisfied: matplotlib>=2.2 in /usr/local/lib/python3.7/dist-packages (from seaborn) (3.
Requirement already satisfied: python-dateutil>=2.1 in /usr/local/lib/python3.7/dist-packages (from matplot
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.7/dist-packages (from matplotlib
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in /usr/local/lib/python3.7/dist-pa
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.7/dist-packages (from matplotlib>=2.2
Requirement already satisfied: typing-extensions in /usr/local/lib/python3.7/dist-packages (from kiwisolver
Requirement already satisfied: pytz>=2017.3 in /usr/local/lib/python3.7/dist-packages (from pandas>=0.23->s
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/dist-packages (from python-dateutil>=2.
```

```
import seaborn as sns
```

```
sns.get_dataset_names()
```

```
['anagrams',
 'anscombe',
 'attention',
 'brain_networks',
 'car_crashes',
```

```
        'diamonds',
        'dots',
        'dowjones',
        'exercise',
        'flights',
        'fmri',
        'geyser',
        'glue',
        'healthexp',
        'iris',
        'mpg',
        'penguins',
        'planets',
        'seaice',
        'taxis',
        'tips',
        'titanic']
```

```
tips=sns.load_dataset('tips')
```

```
tips
```

	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
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3
240	27.18	2.00	Female	Yes	Sat	Dinner	2
				Yes	Sat	Dinner	2
				No	Sat	Dinner	2
243	18.78	3.00	Female	No	Thur	Dinner	2

244 rows × 7 columns

Saved successfully!

```
type(tips)
```

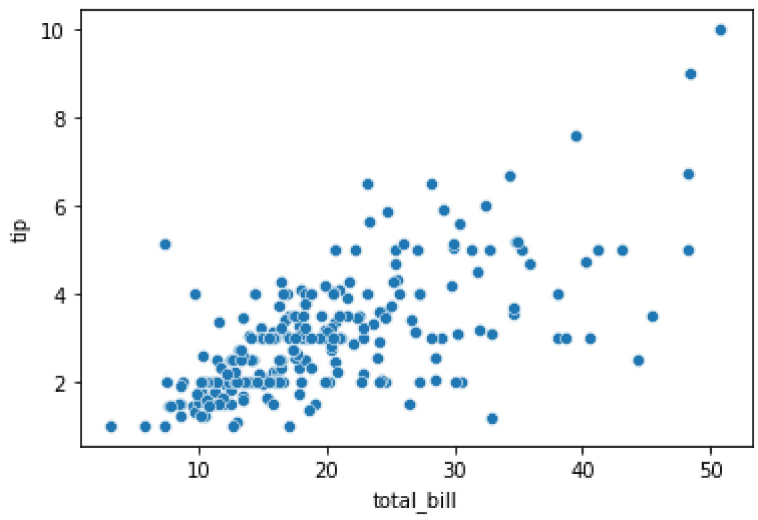
```
pandas.core.frame.DataFrame
```

```
tips.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    category
 3   smoker      244 non-null    category
 4   day         244 non-null    category
 5   time        244 non-null    category
 6   size        244 non-null    int64   
dtypes: category(4), float64(2), int64(1)
memory usage: 7.4 KB
```

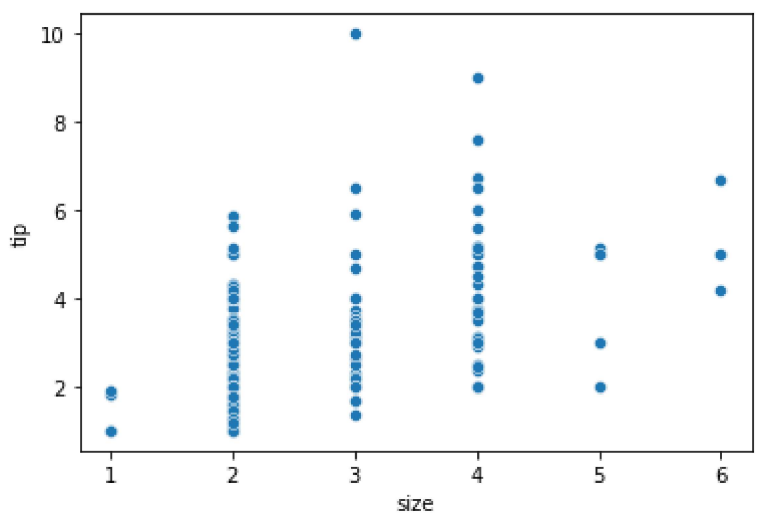
```
sns.scatterplot(x='total_bill',y='tip',data=tips)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f15460ac650>
```




```
sns.scatterplot(x='size',y='tip',data=tips)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f1546057690>
```



Saved successfully!

Colab paid products - Cancel contracts here

✓ 0s completed at 10:38 PM