

```
In [1]: import numpy as np
arr=np.array([[1,2,3],[34,56,78],[22,33,44]])
print("original array",arr)
```

```
original array [[ 1  2  3]
 [34 56 78]
 [22 33 44]]
```

```
In [4]: import numpy as np
na=np.array([1,2,3])
# na
new_arr=np.r_[arr,[na]]
print("new array",new_arr)
```

```
new array [[ 1  2  3]
 [34 56 78]
 [22 33 44]
 [ 1  2  3]]
```

```
In [10]: import pandas as pd
import numpy as np
df=pd.DataFrame(np.random.randn(5,4),
                 ['a','b','c','d','e'],['w','x','y','z'])
df
```

Out[10]:

	w	x	y	z
a	-1.142032	-0.807771	1.218944	0.569139
b	-0.152715	-1.070854	-1.797606	1.499187
c	0.671548	0.594451	-1.756803	0.253431
d	1.959344	1.593450	-0.389332	0.335396
e	0.599975	1.509099	-0.233384	-0.344869

```
In [6]: df["w"]
```

```
Out[6]: a    -0.536477
b    -0.518584
c     2.012798
d    -0.431437
e     1.542469
Name: w, dtype: float64
```

```
In [7]: df.loc['a']
```

```
Out[7]: w    -0.536477  
        x     0.445307  
        y    -0.917868  
        z    -0.061874  
        Name: a, dtype: float64
```

```
In [8]: df.loc['b','y']
```

```
Out[8]: -0.18707395352516518
```

```
In [9]: df.loc[['a','b'],['w','x']]
```

```
Out[9]:
```

	w	x
a	-0.536477	0.445307
b	-0.518584	-0.577105

```
In [10]: df
```

```
Out[10]:
```

	w	x	y	z
a	-0.536477	0.445307	-0.917868	-0.061874
b	-0.518584	-0.577105	-0.187074	1.643717
c	2.012798	-0.316101	-0.682504	0.617219
d	-0.431437	0.655573	0.195862	1.539922
e	1.542469	0.150712	-1.342804	-1.795212

```
In [11]: df>0
```

```
Out[11]:
```

	w	x	y	z
a	False	True	False	False
b	False	False	False	True
c	True	False	False	True
d	False	True	True	True
e	True	True	False	False

```
In [12]: df['x']>0
```

```
Out[12]: a    True  
        b   False  
        c   False  
        d    True  
        e    True  
        Name: x, dtype: bool
```

```
In [13]: df[df["w"]>0]
```

```
Out[13]:
```

	w	x	y	z
c	2.012798	-0.316101	-0.682504	0.617219
e	1.542469	0.150712	-1.342804	-1.795212

```
In [15]: d={"a":[1,2,np.nan], "b":[5,np.nan,np.nan],
           "c":[1,2,3], "d":[np.nan,np.nan,np.nan]}
```

```
In [18]: df1=pd.DataFrame(d)
df1
```

```
Out[18]:
```

	a	b	c	d
0	1.0	5.0	1	NaN
1	2.0	NaN	2	NaN
2	NaN	NaN	3	NaN

```
In [20]: df1.dropna()
```

```
Out[20]:
```

	a	b	c	d
--	---	---	---	---

```
In [21]: df1.dropna(how='any')
```

```
Out[21]:
```

	a	b	c	d
--	---	---	---	---

```
In [24]: df1.dropna(how='all',axis=1)
```

```
Out[24]:
```

	a	b	c
0	1.0	5.0	1
1	2.0	NaN	2
2	NaN	NaN	3

```
In [26]: df1.isna()
```

```
Out[26]:
```

	a	b	c	d
0	False	False	False	True
1	False	True	False	True
2	True	True	False	True

```
In [27]: df1.isna().sum()
```

```
Out[27]: a      1  
        b      2  
        c      0  
        d      3  
        dtype: int64
```

```
In [28]: df1.dropna(thresh=2)
```

```
Out[28]:
```

	a	b	c	d
0	1.0	5.0	1	NaN
1	2.0	NaN	2	NaN

```
In [29]: df1.dropna(thresh=3)
```

```
Out[29]:
```

	a	b	c	d
0	1.0	5.0	1	NaN

```
In [30]: df1.dropna(thresh=1)
```

```
Out[30]:
```

	a	b	c	d
0	1.0	5.0	1	NaN
1	2.0	NaN	2	NaN
2	NaN	NaN	3	NaN

```
In [31]: df1.dropna(thresh=3,axis=1)
```

```
Out[31]:
```

	c
0	1
1	2
2	3

```
In [32]: df1.dropna(thresh=3,axis=1)
```

```
Out[32]:
```

	c
0	1
1	2
2	3

```
In [33]: df1.fillna(value=1)
```

```
Out[33]:
```

	a	b	c	d
0	1.0	5.0	1	1.0
1	2.0	1.0	2	1.0
2	1.0	1.0	3	1.0

```
In [34]: df1['a'].fillna(value=df1['a'].mean())
```

```
Out[34]:
```

0	1.0
1	2.0
2	1.5

Name: a, dtype: float64

```
In [38]: data ={'COMPANY':['FB','GOOGLE','META','META','GOOGLE','FB'],
               "PERSON":["mam",'in','si','srr','maheshbabu','chandu'],
               'SALES':[300,500,100,200,600,20]}
```

```
In [40]: df2=pd.DataFrame(data)
df2
```

```
Out[40]:
```

	COMPANY	PERSON	SALES
0	FB	mam	300
1	GOOGLE	in	500
2	META	si	100
3	META	srr	200
4	GOOGLE	maheshbabu	600
5	FB	chandu	20

```
In [44]: bycomp=df2.groupby("COMPANY")
```

```
In [45]: bycomp
```

```
Out[45]: <pandas.core.groupby.generic.DataFrameGroupBy object at 0x00000211091E3FD0>
```

```
In [46]: bycomp.sum()['SALES']
```

```
Out[46]:
```

COMPANY	
FB	320
GOOGLE	1100
META	300

Name: SALES, dtype: int64

In [48]: `bycomp.max()`

Out[48]:

	PERSON	SALES
COMPANY		
FB	mam	300
GOOGLE	maheshbabu	600
META	srr	200

In [49]: `bycomp.describe()`

Out[49]:

	SALES							
	count	mean	std	min	25%	50%	75%	max
COMPANY								
FB	2.0	160.0	197.989899	20.0	90.0	160.0	230.0	300.0
GOOGLE	2.0	550.0	70.710678	500.0	525.0	550.0	575.0	600.0
META	2.0	150.0	70.710678	100.0	125.0	150.0	175.0	200.0

In [50]: `bycomp['SALES'].max()`

Out[50]:

COMPANY	
FB	300
GOOGLE	600
META	200

Name: SALES, dtype: int64

In [51]: `df2.loc[bycomp['SALES'].idxmax()][['PERSON', 'SALES']]`

Out[51]:

	PERSON	SALES
0	mam	300
4	maheshbabu	600
3	srr	200

In [52]: `bycomp[['PERSON', 'SALES']].max()`

Out[52]:

	PERSON	SALES
COMPANY		
FB	mam	300
GOOGLE	maheshbabu	600
META	srr	200

```
In [53]: df=pd.read_csv(r"C:\Users\vippa\Downloads\samplecsv.csv")
```

```
In [54]: df
```

Out[54]:

	Year	Industry_aggregation_NZSIOC	Industry_code_NZSIOC	Industry_name_NZSIOC	
0	2020	Level 1	99999	All industries	[(m
1	2020	Level 1	99999	All industries	[(m
2	2020	Level 1	99999	All industries	[(m
3	2020	Level 1	99999	All industries	[(m
4	2020	Level 1	99999	All industries	[(m
...	...	...	...	...	
37075	2013	Level 3	ZZ11	Food product manufacturing	Perce
37076	2013	Level 3	ZZ11	Food product manufacturing	Perce
37077	2013	Level 3	ZZ11	Food product manufacturing	Perce
37078	2013	Level 3	ZZ11	Food product manufacturing	Perce
37079	2013	Level 3	ZZ11	Food product manufacturing	Perce

37080 rows × 10 columns



```
In [55]: df.head()
```

Out[55]:

	Year	Industry_aggregation_NZSIOC	Industry_code_NZSIOC	Industry_name_NZSIOC	Units
0	2020	Level 1	99999	All industries	Dollars (millions)
1	2020	Level 1	99999	All industries	Dollars (millions)
2	2020	Level 1	99999	All industries	Dollars (millions)
3	2020	Level 1	99999	All industries	Dollars (millions)
4	2020	Level 1	99999	All industries	Dollars (millions)

```
In [56]: df.tail()
```

Out[56]:

	Year	Industry_aggregation_NZSIOC	Industry_code_NZSIOC	Industry_name_NZSIOC	
37075	2013	Level 3	ZZ11	Food product manufacturing	Perce
37076	2013	Level 3	ZZ11	Food product manufacturing	Perce
37077	2013	Level 3	ZZ11	Food product manufacturing	Perce
37078	2013	Level 3	ZZ11	Food product manufacturing	Perce
37079	2013	Level 3	ZZ11	Food product manufacturing	Perce



In [57]: `df.head(4)`

Out[57]:

	Year	Industry_aggregation_NZSIOC	Industry_code_NZSIOC	Industry_name_NZSIOC	Units
0	2020	Level 1	99999	All industries	Dollars (millions)
1	2020	Level 1	99999	All industries	Dollars (millions)
2	2020	Level 1	99999	All industries	Dollars (millions)
3	2020	Level 1	99999	All industries	Dollars (millions)

In [58]: `df.tail(5)`

Out[58]:

	Year	Industry_aggregation_NZSIOC	Industry_code_NZSIOC	Industry_name_NZSIOC	
37075	2013	Level 3	ZZ11	Food product manufacturing	Perce
37076	2013	Level 3	ZZ11	Food product manufacturing	Perce
37077	2013	Level 3	ZZ11	Food product manufacturing	Perce
37078	2013	Level 3	ZZ11	Food product manufacturing	Perce
37079	2013	Level 3	ZZ11	Food product manufacturing	Perce

In [59]: `df=pd.read_csv(r"C:\Users\vippa\Downloads\delimatercsv.csv")`

In [61]: `name={"NAME":'Sridhar','Age':'all_age','City':"Place"}`

In [62]: `df=df.rename(columns=name)`

```
In [63]: df
```

Out[63]:

	<b>Name;Age;City</b>
<b>0</b>	Alice;25;New York
<b>1</b>	Bob;30;San Francisco
<b>2</b>	Charlie;22;Los Angeles

```
In [11]: df_titanic=pd.read_csv(r"C:\Users\vippa\Downloads\titanic_train.csv")
df_titanic
```

Out[11]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500
...	...	...	...	...	...	...	...	...	...	...
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500

891 rows × 12 columns



```
In [65]: df_titanic.head()
```

```
Out[65]:
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...)	female	38.0	1	0	PC 17599	71.2833	NaN
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN

```
In [72]: df_titanic['Pclass'].value_counts()
```

```
Out[72]: Pclass
3      491
1      216
2      184
Name: count, dtype: int64
```

In [76]: df\_titanic

Out[76]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500
...	...	...	...	...	...	...	...	...	...	...
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500

891 rows × 12 columns



```
In [73]: df_titanic.describe()
```

```
Out[73]:
```

	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
<b>count</b>	891.000000	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
<b>mean</b>	446.000000	0.383838	2.308642	29.699118	0.523008	0.381594	32.204208
<b>std</b>	257.353842	0.486592	0.836071	14.526497	1.102743	0.806057	49.693429
<b>min</b>	1.000000	0.000000	1.000000	0.420000	0.000000	0.000000	0.000000
<b>25%</b>	223.500000	0.000000	2.000000	20.125000	0.000000	0.000000	7.910400
<b>50%</b>	446.000000	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200
<b>75%</b>	668.500000	1.000000	3.000000	38.000000	1.000000	0.000000	31.000000
<b>max</b>	891.000000	1.000000	3.000000	80.000000	8.000000	6.000000	512.329200

```
In [84]: df_titanic.shape[0]
```

```
Out[84]: 891
```

```
In [85]: df_titanic.isnull().sum()
```

```
Out[85]: PassengerId      0
Survived      0
Pclass      0
Name      0
Sex      0
Age      177
SibSp      0
Parch      0
Ticket      0
Fare      0
Cabin      687
Embarked      2
dtype: int64
```

```
In [93]: ans=(df_titanic.isnull().sum()/df_titanic.shape[0])*100
#ans
ans.max()
```

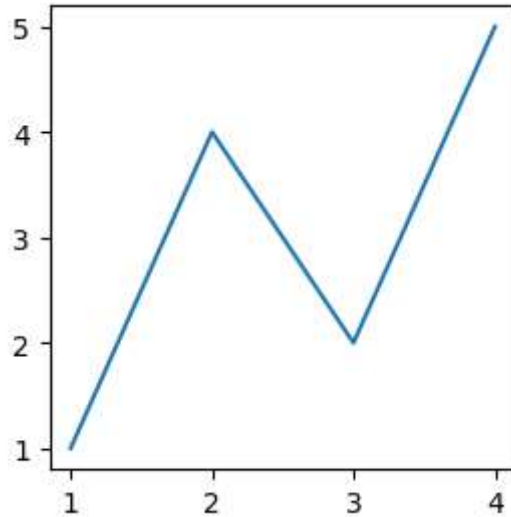
```
Out[93]: 77.10437710437711
```

```
In [111]: import matplotlib.pyplot as plt
```

```
In [112]: x = [1,2,3,4]
y = [1,4,2,5]

plt.figure(figsize=(3,3))
plt.plot(x,y)
```

Out[112]: [<matplotlib.lines.Line2D at 0x2110eab1290>]



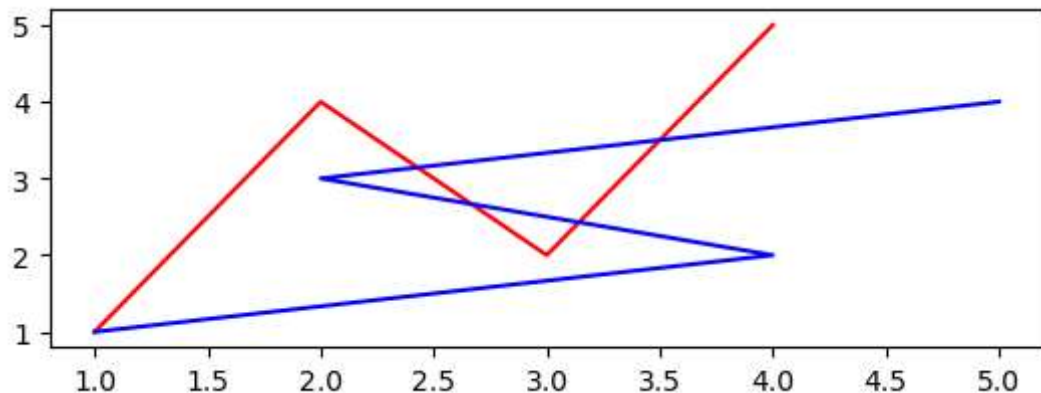
```
In [136]: #plt.subplot(1, 2, 1)
#plt.plot(x, y, 'b')

#plt.subplot(1, 2, 1)
#plt.plot(y, x, 'g')

plt.subplot(2,1,2)
plt.plot(x,y, 'r')

plt.subplot(2,1,2)
plt.plot(y,x, 'b')
```

Out[136]: [<matplotlib.lines.Line2D at 0x2111074c350>]



```
In [114]: years = np.random.randint(0,100)
```

```
In [115]: years
```

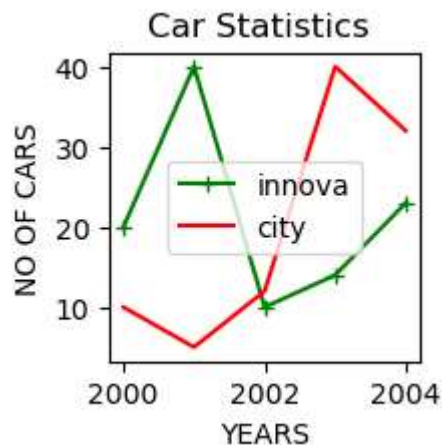
```
Out[115]: 88
```

```
In [4]: import numpy as np
x_years = np.arange(2000,2005, dtype = int)
y_innova = np.array([20,40,10,14,23])
y_honda_city=np.array([10,5,12,40,32])
x_years,y_innova
```

```
Out[4]: (array([2000, 2001, 2002, 2003, 2004]), array([20, 40, 10, 14, 23]))
```

```
In [5]: import matplotlib.pyplot as plt
plt.figure(figsize=(2,2))
plt.plot(x_years,y_innova,'-+g')
plt.plot(x_years,y_honda_city,'r')
plt.xlabel("YEARS")
plt.ylabel("NO OF CARS")
plt.title("Car Statistics ")
plt.legend(['innova',"city"])
```

```
Out[5]: <matplotlib.legend.Legend at 0x2102314ed90>
```

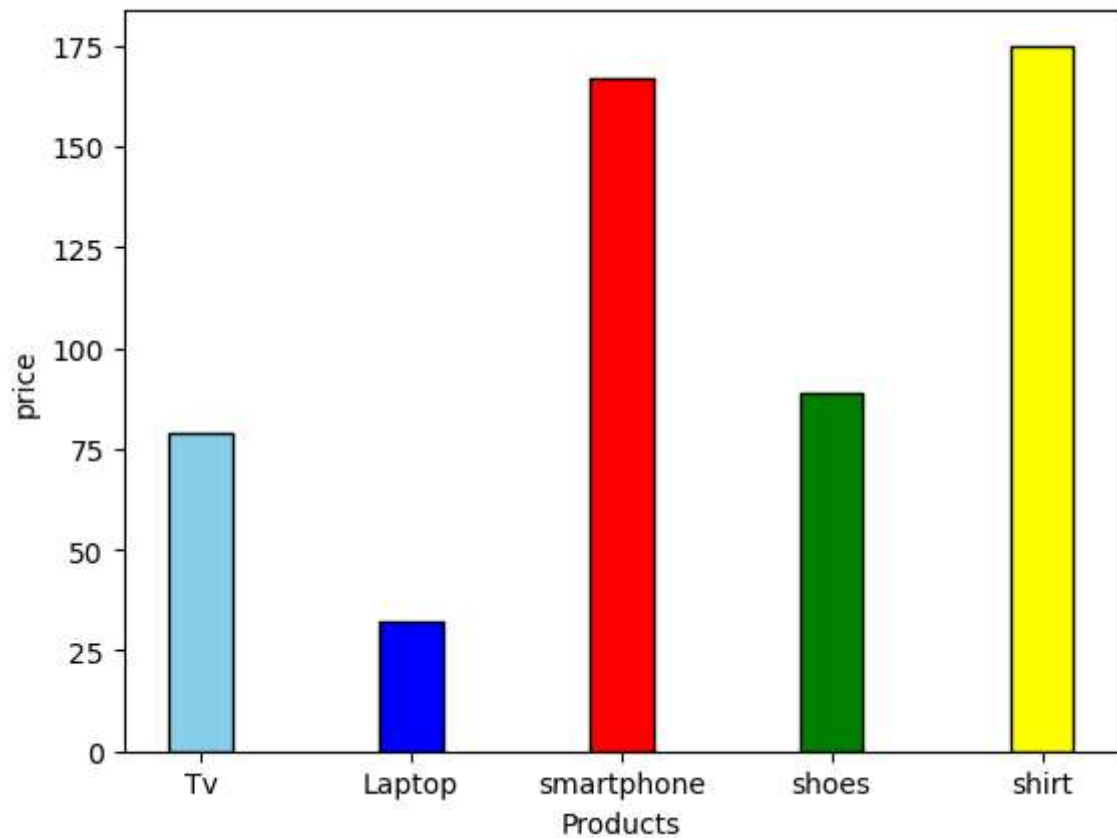




```
In [6]: # barplot
products = ['Tv', 'Laptop', 'smartphone', 'shoes', 'shirt']
sales = np.random.randint(30,200,size=len(products))

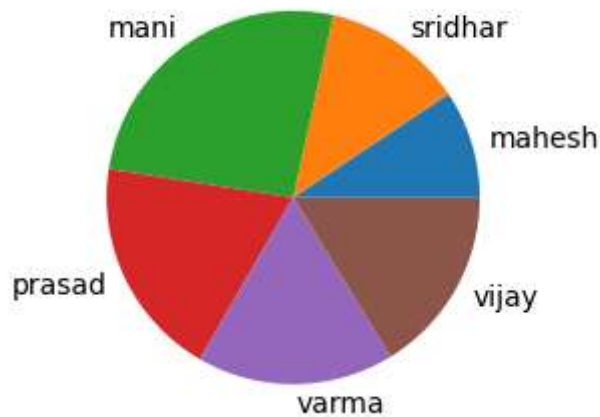
plt.bar(products,sales,color=['skyblue','blue','red','green','yellow'], width=
plt.ylabel("price")
plt.xlabel('Products')
```

Out[6]: Text(0.5, 0, 'Products')



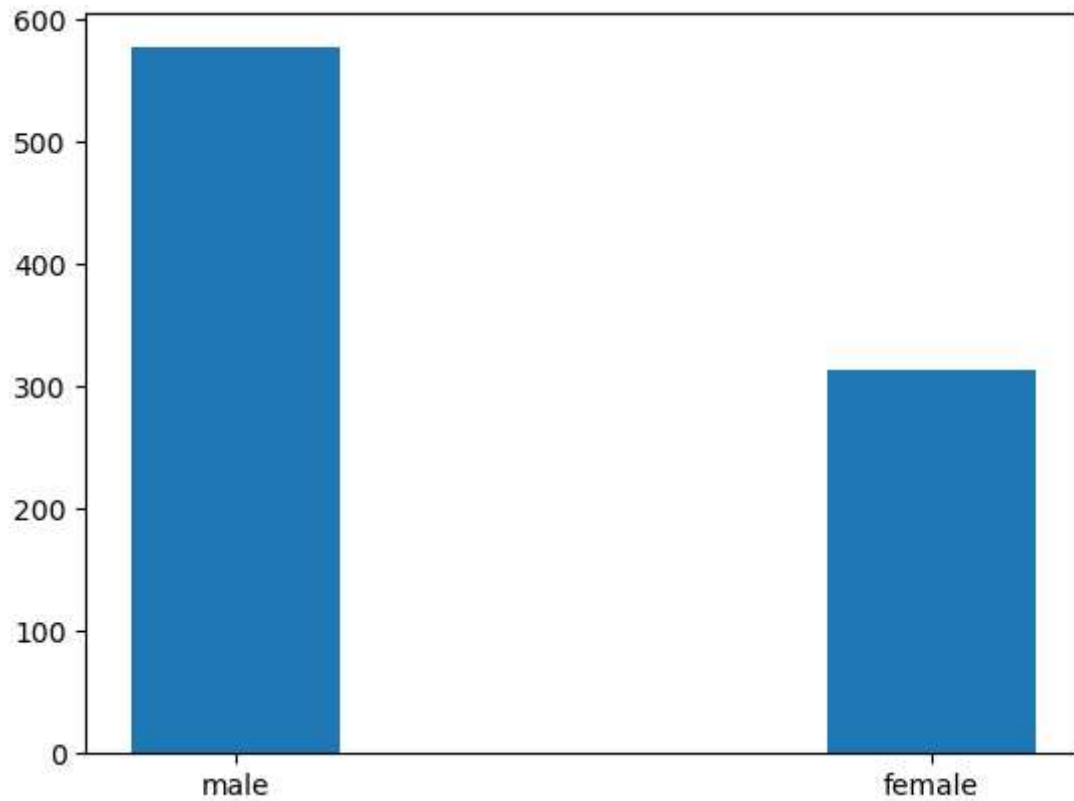
```
In [7]: student_name=['mahesh','sridhar','mani','prasad','varma','vijay']  
scores = [23,30,64,47,42,40]  
plt.figure(figsize=(3,4))  
plt.pie(scores,labels=student_name)
```

```
Out[7]: ([<matplotlib.patches.Wedge at 0x2102496dd10>,  
<matplotlib.patches.Wedge at 0x2102496e9d0>,  
<matplotlib.patches.Wedge at 0x2102496fe10>,  
<matplotlib.patches.Wedge at 0x21024981510>,  
<matplotlib.patches.Wedge at 0x21024982910>,  
<matplotlib.patches.Wedge at 0x21024983f50>],  
[Text(1.0528888995438015, 0.31847286417753534, 'mahesh'),  
Text(0.6213091383344759, 0.9077306619378191, 'sridhar'),  
Text(-0.6213091383344759, 0.9077306619378191, 'mani'),  
Text(-0.9919348580473716, -0.4754631819505492, 'prasad'),  
Text(0.014047325301146971, -1.0999103020937133, 'varma'),  
Text(0.9595739177560261, -0.5377898254544718, 'vijay')])
```



```
In [13]: male=(df_titanic["Sex"]=="male").sum()
female=(df_titanic["Sex"]=="female").sum()
counts=[male,female]
names = ["male","female"]
plt.bar(names,counts,width=0.3)
```

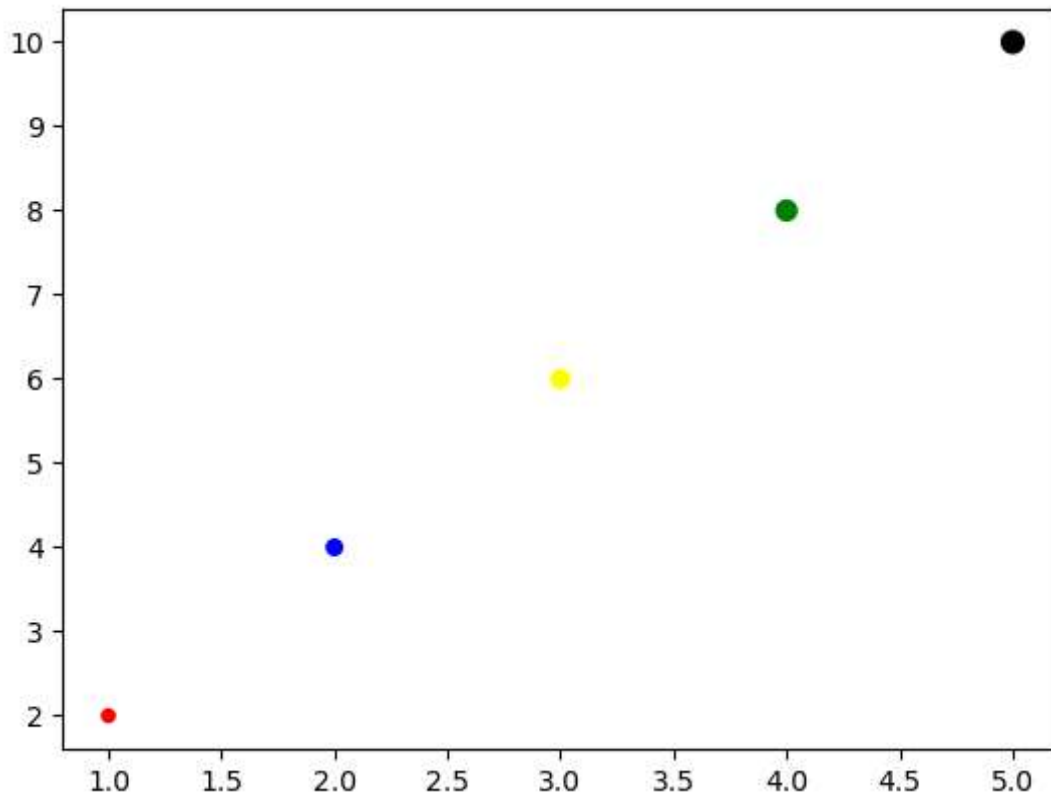
Out[13]: <BarContainer object of 2 artists>



```
In [17]: x_data = [1,2,3,4,5]
y_data = [2,4,6,8,10]

plt.scatter(x_data,y_data,sizes=[20,30,40,50,60],
            c=["red","blue","yellow","green","black",])
```

Out[17]: <matplotlib.collections.PathCollection at 0x2102969d3d0>



In [ ]: