

Data Visualization Example 1

Question: Data Visualization.

Please refer your own data.

1. Draw a pie chart.
2. Draw a Bar graph.
3. Draw a horizontal bar graph.
4. Draw scatter plot diagram.
5. Draw Histogram.

Code :

```
import numpy as np
```

```
import pandas as pd
```

```
import matplotlib.pyplot as plt
```

```
data = [33,52,12,17,42,48]
```

```
color = ['lightblue','darkblue','cyan','wheat','brown','red']
```

```
labels = ['Python','C','C++','PHP','Java','Ruby']
```

```
plt.pie(data,labels=labels,autopct='%.2f%%',colors = color)
```

```
plt.show()
```

```
filmns = ['Tanhaji','Avatar 1','Avatar 2','Avenger','Spiderman','Superman']
```

```
box_office = [350,1400,650,1800,400,500]
```

```
plt.bar(filmns,box_office)
```

```
plt.ylabel('Box Office (in crore)')
```

```
plt.xlabel('Movie Title')
```

```
plt.title('World wide box office collection')
```

```
plt.show()
```

```
plt.barh(filmns,box_office)
```

```
plt.xlabel('Box Office (in crore)')
```

```
plt.ylabel('Movie Title')
```

```
plt.title('World wide box office collection')
```

```
plt.show()
```

```
x = [1,2,3,4,5,6,7,8]
```

```
y = [5,2,4,2,1,4,5,2]
```

```
plt.scatter(x,y,label='Sample lable',marker='o',s=25)
```

```
plt.xlabel('x')
```

```
plt.ylabel('y')
```

```
plt.title('Scatter Plot Diagram')
```

```
plt.legend()
```

```
plt.show()
```

```
data = [163,163,160,170,180,172,173,190,175,180]
```

```
plt.hist(data,color='purple')
```

```
plt.xlabel('Height in cm')
```

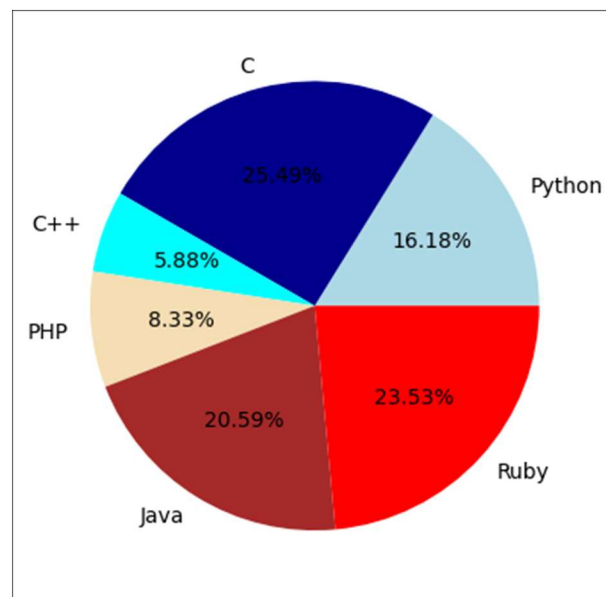
```
plt.ylabel('Number of people')
```

```
plt.title('Height of Peoples')
```

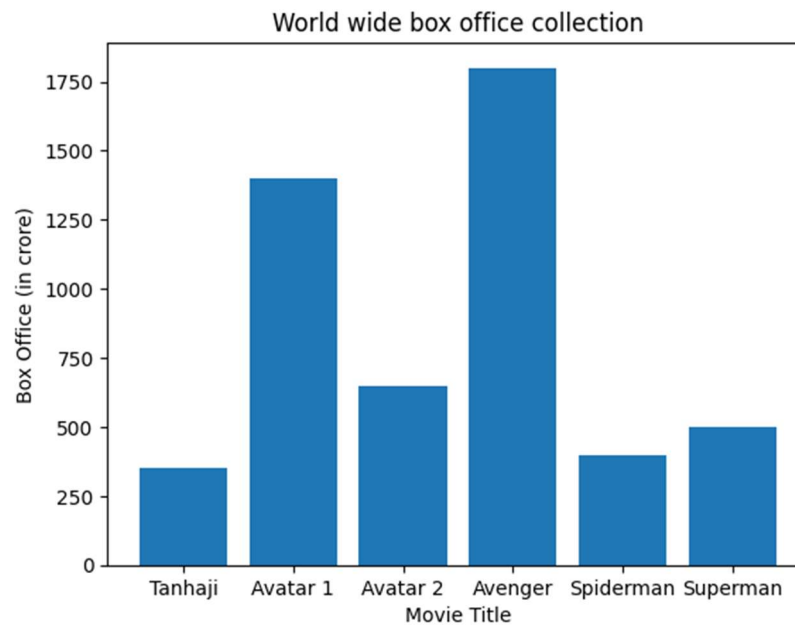
```
plt.show()
```

Output :

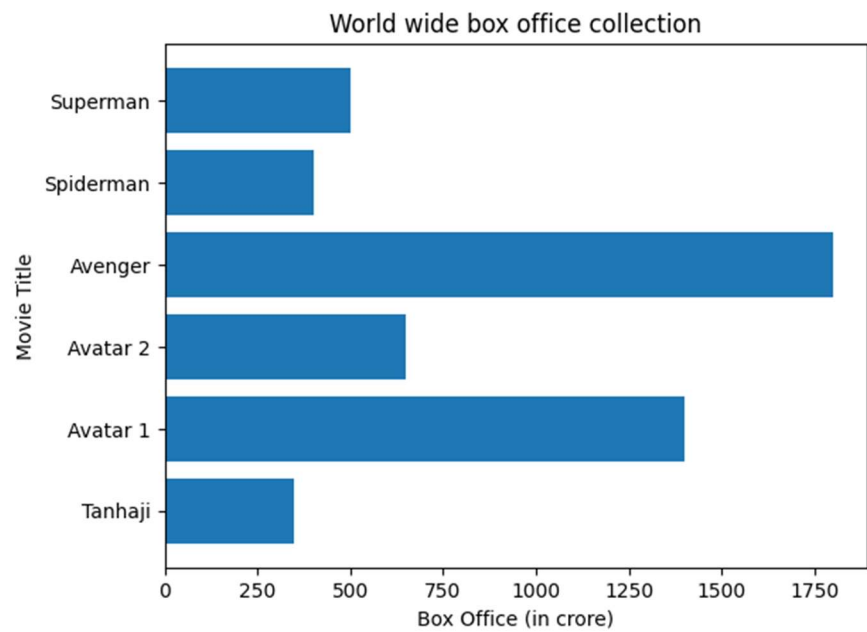
Pie Chart



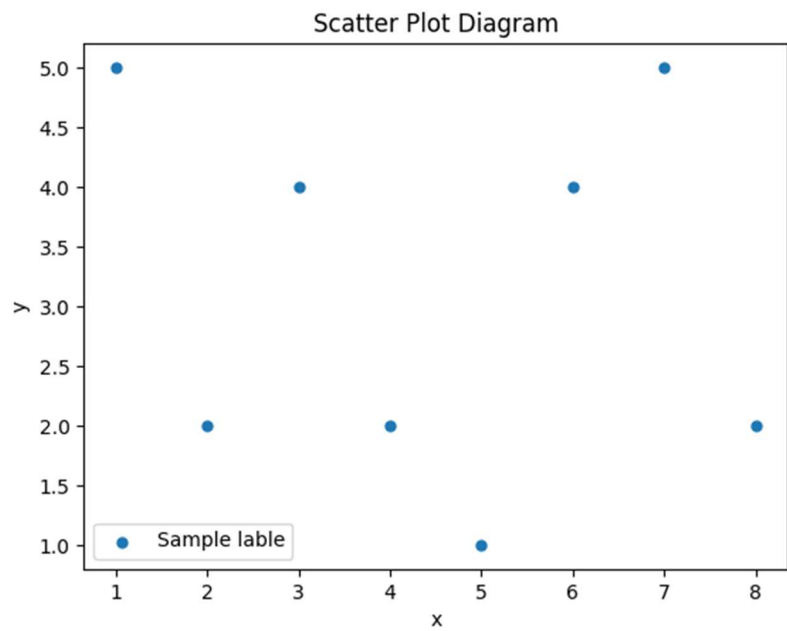
Bar Graph



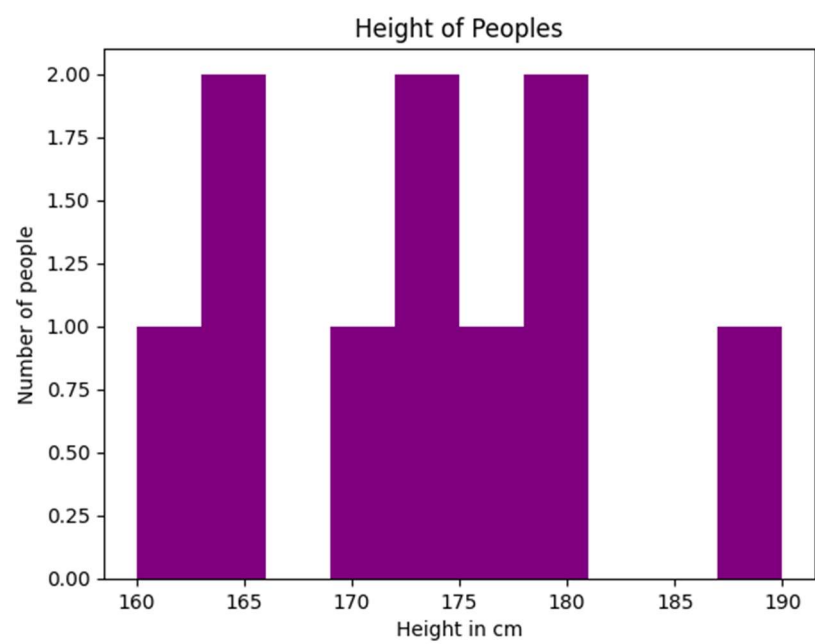
Horizontal Bar Graph



Scatter Graph



Histogram



Data Visualization Example 2

Question : Data Visualization

Please refer “income.csv” dataset.

1. Draw a scatter plot for Age and Income.
2. Draw a bar graph for Age and Income.
3. Draw histogram for Income.
4. Draw a pie chart for Income

Code :

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

dat = pd.read_csv("income.csv")

plt.scatter(dat['Age'],dat['Income($)'])
plt.xlabel('Age')
plt.ylabel('Income ($)')
plt.title('Scatter Plot')
plt.show()

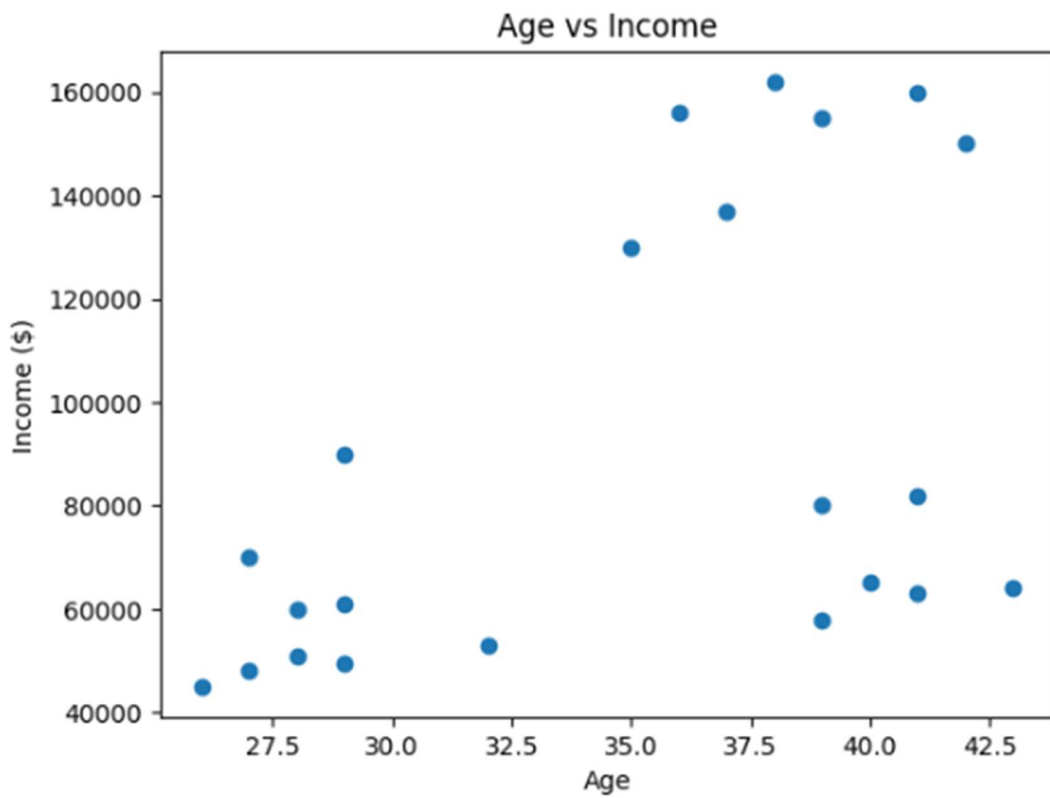
plt.bar(dat['Age'],dat['Income($)'])
plt.title('Bar Chart')
plt.xlabel('Age')
plt.ylabel('Income')
plt.show()

plt.hist(dat['Income($)'])
plt.title('Histogram')
plt.show()

data_name['Age']
```

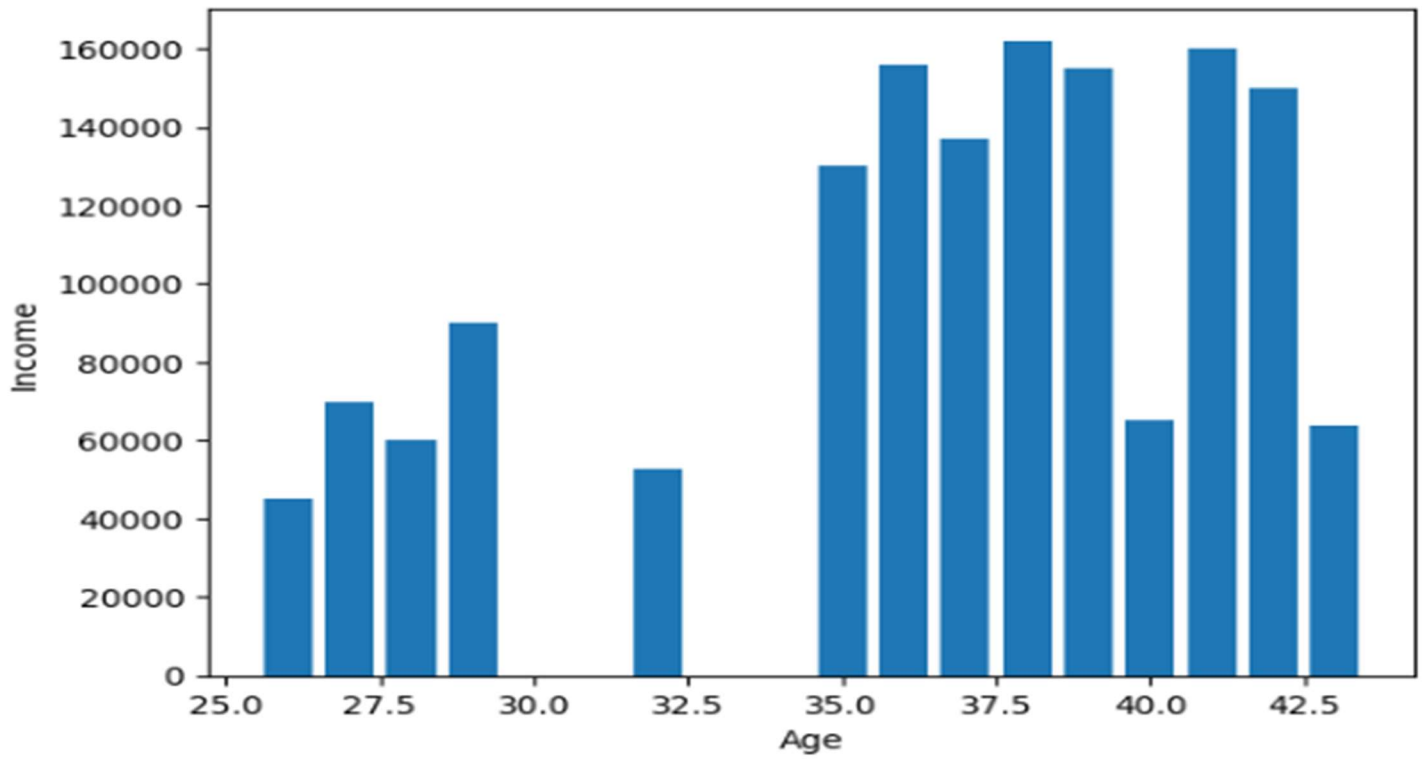
```
names = list(data_name['Name'])  
print(names)  
  
plt.pie(data_name['Income($)',labels=names, autopct='%0.2f%%')  
plt.show()
```

Output :



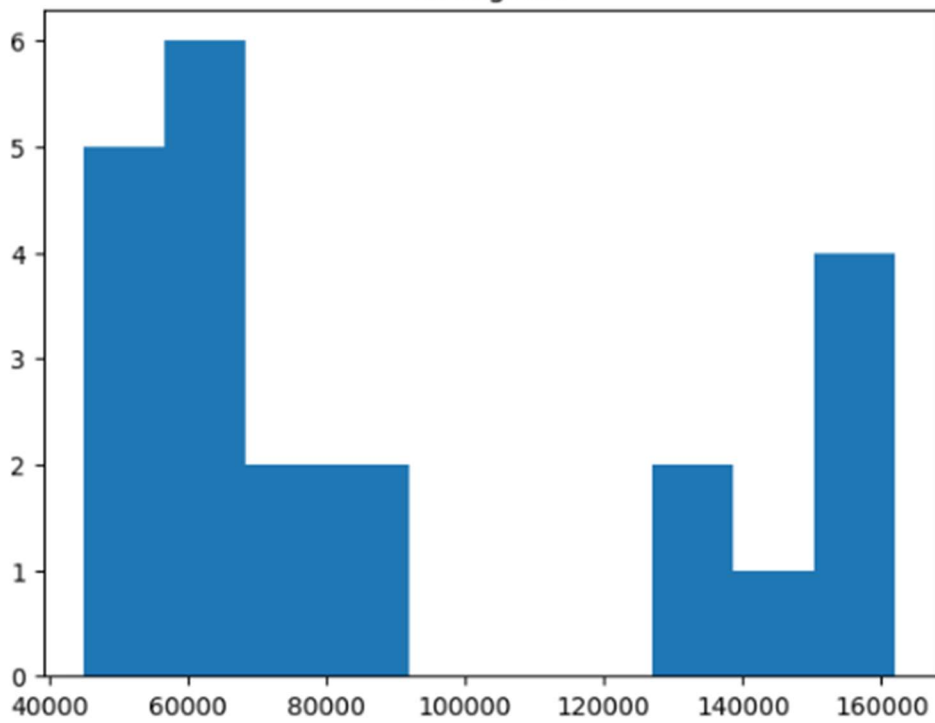
Scatter Plot

Bar Chart

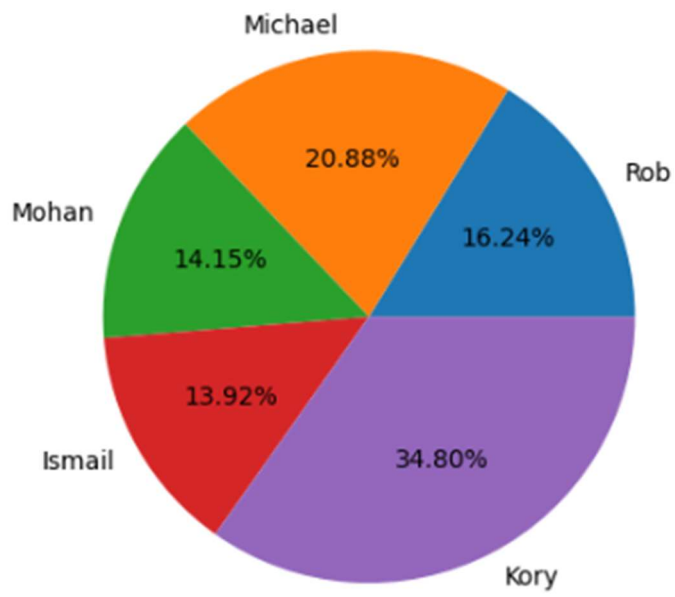


Bar Graph

Histogram



Histogram



Pie Chart

Data Visualization Example 3

Question : Data Visualization

Please refer “iris.csv” dataset.

1. Draw a Box plot for ‘variety’ and ‘petal.length’.
2. Draw a Scatter plot for ‘sepal.length’ and ‘sepal.width’.

Code :

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

iris = pd.read_csv('iris.csv')

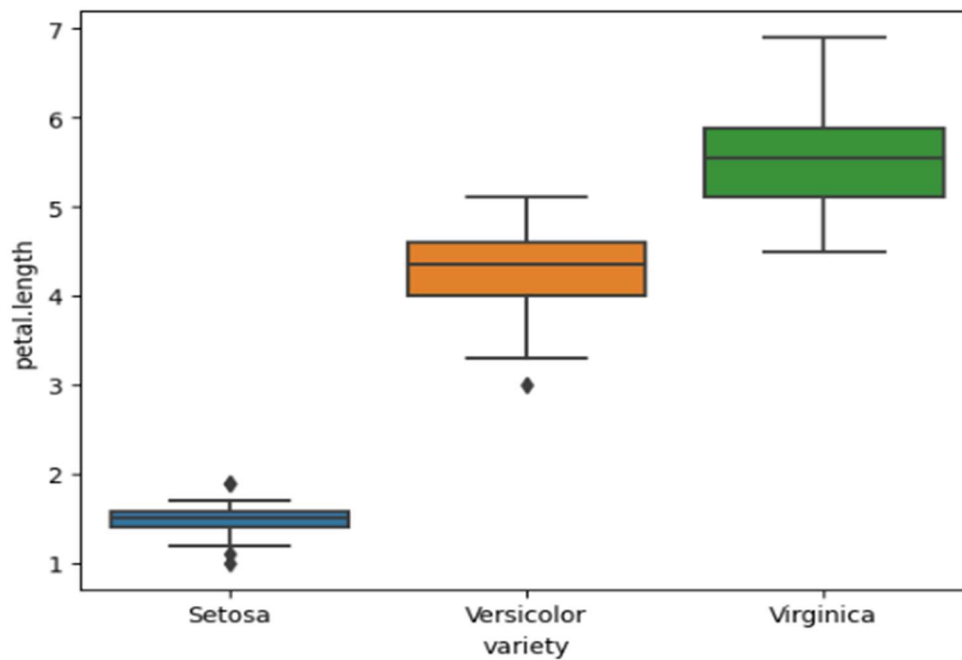
sb.boxplot(x='variety', y='petal.length', data=iris)

iris.plot(kind="scatter", x='sepal.length', y='sepal.width')
```

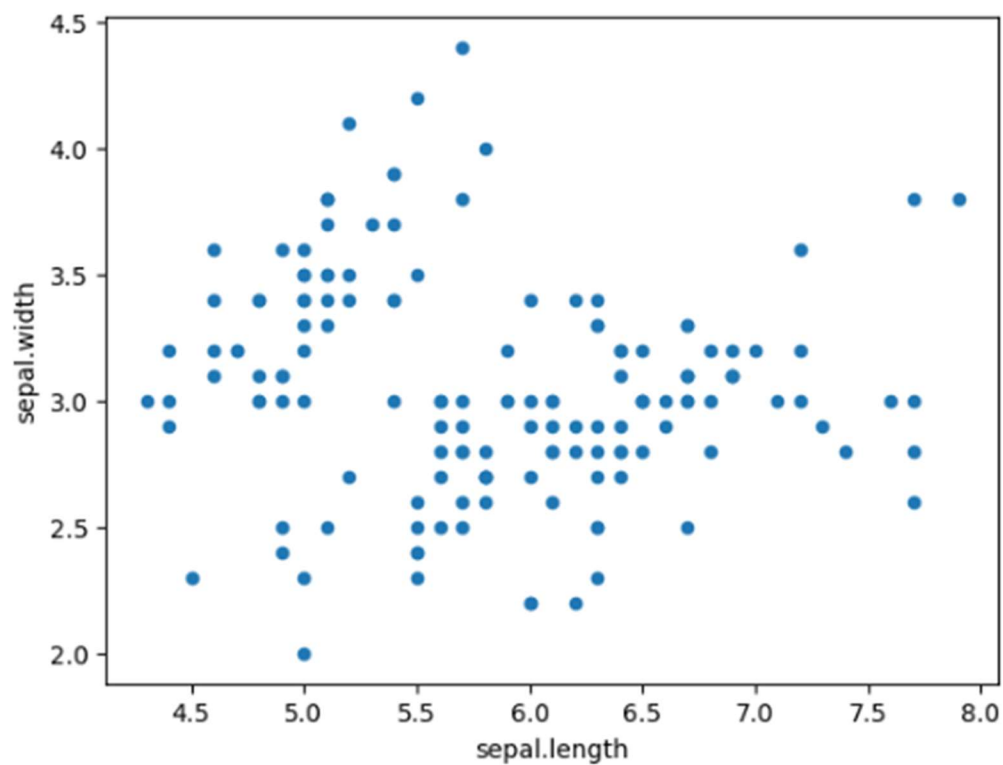
Output :

Box Plot

```
<AxesSubplot: xlabel='variety', ylabel='petal.length'>
```



```
<AxesSubplot: xlabel='sepal.length', ylabel='sepal.width'>
```



Scatter Plot

Visualization Example 4

Question : Data Visualization

Please refer “Salary_Data.csv” dataset.

1. Draw a bar graph for Age and Salary.
2. Draw a line chart for YearsExperience and Salary.
3. Draw a Scatter plot for Age and Salary.
4. Draw Histogram for Salary.

Code:

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

sal = pd.read_csv('Salary_Data.csv')

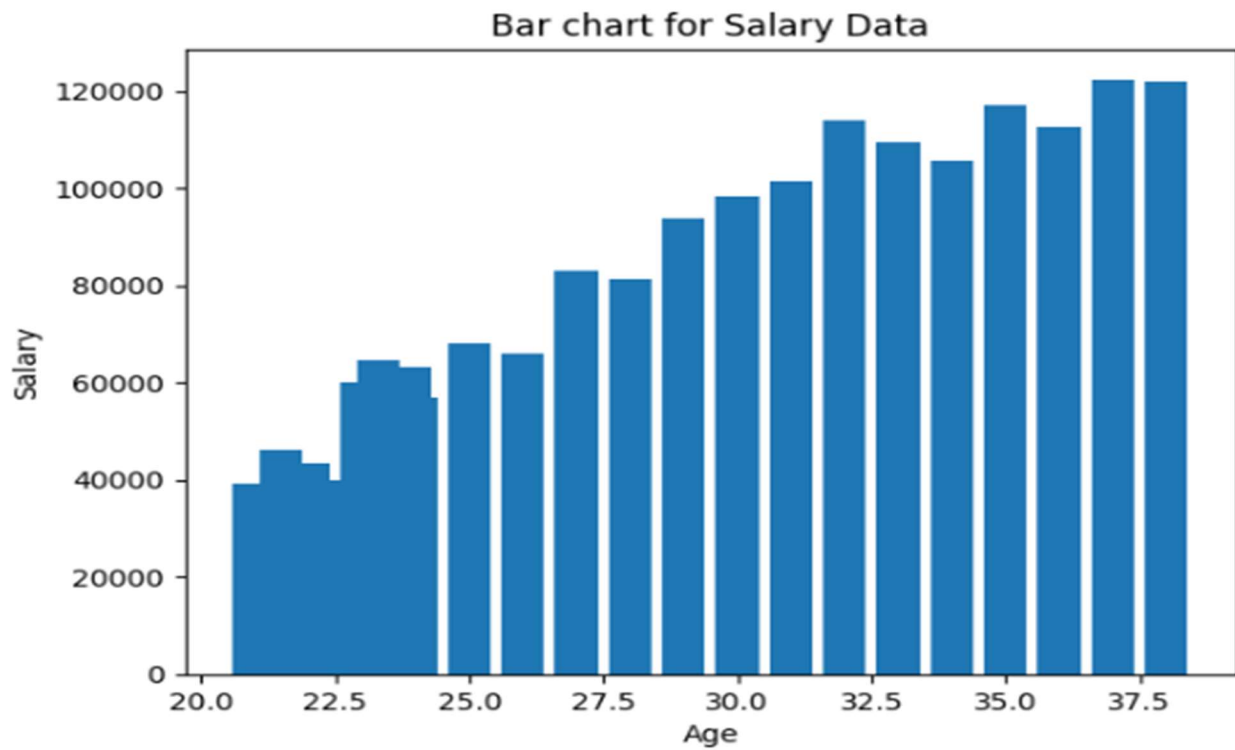
plt.bar(sal['Age'],sal['Salary'])
plt.xlabel('Age')
plt.ylabel('Salary')
plt.title('Bar chart for Salary Data')
plt.show()

plt.plot(sal['YearsExperience'],sal['Salary'])
plt.xlabel('Year of Experience')
plt.ylabel('Salary')
plt.show()

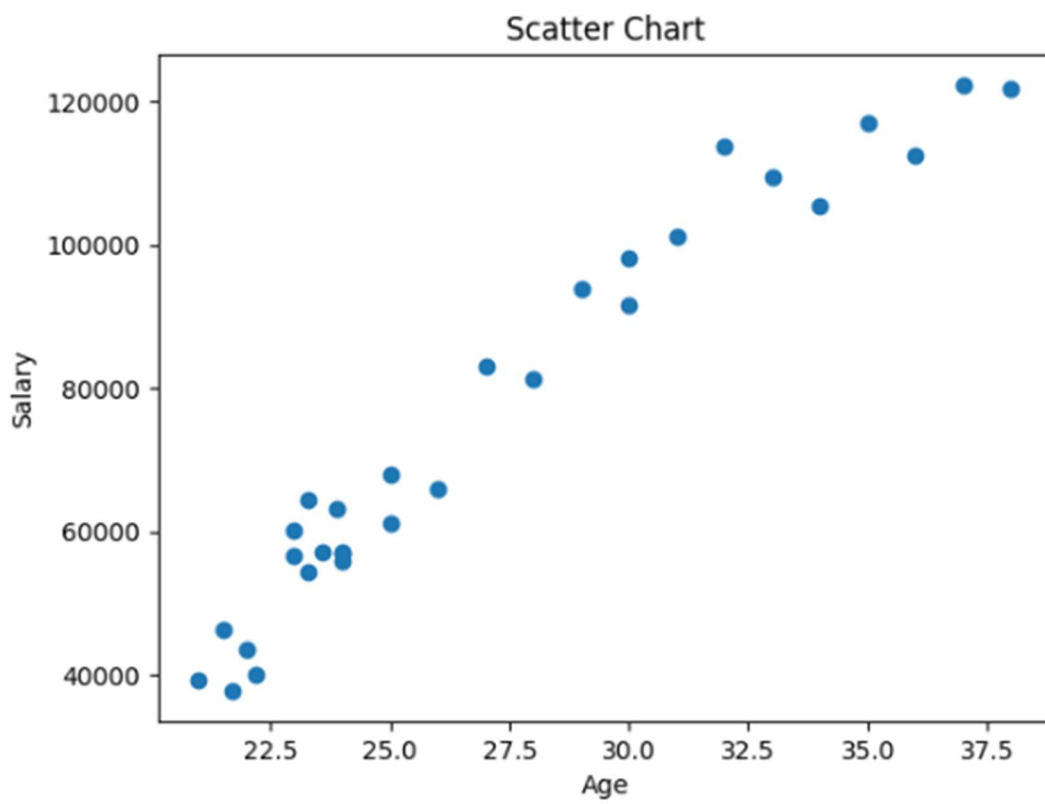
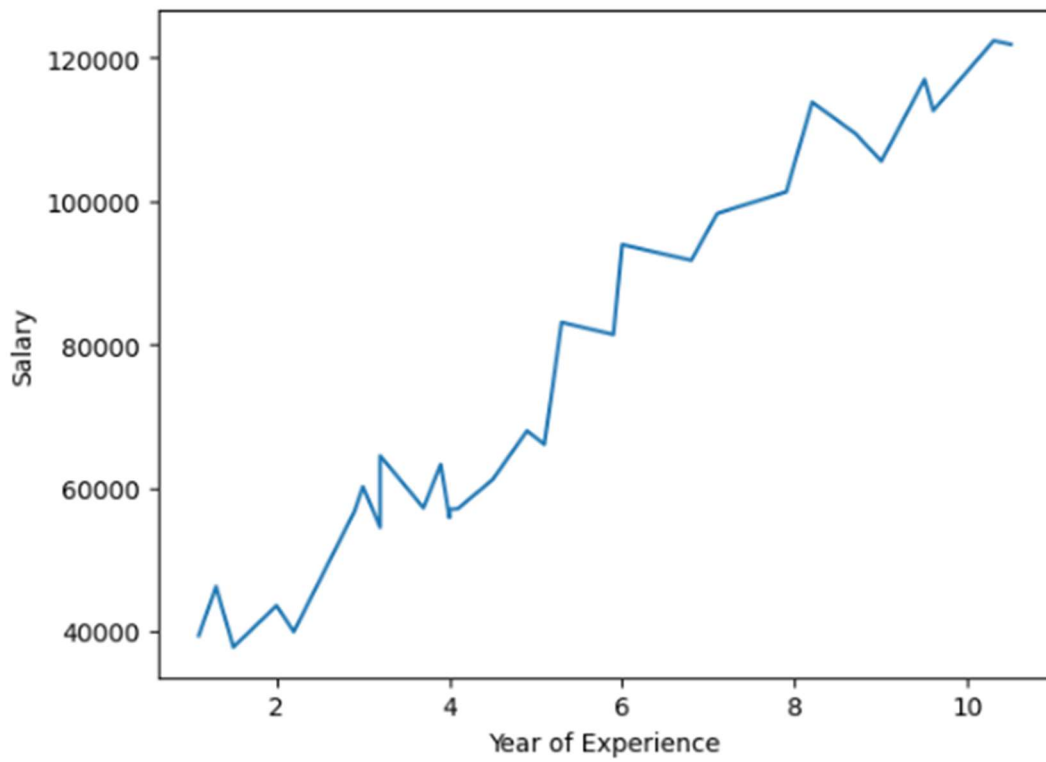
plt.scatter(sal['Age'],sal['Salary'])
plt.xlabel('Age')
plt.ylabel('Salary')
plt.title('Scatter Chart')
plt.show()
```

```
plt.hist(sal['Salary'])  
plt.xlabel('Salary')  
# plt.ylabel('Experience')  
plt.title('Histogram For Salary Data')  
plt.show()
```

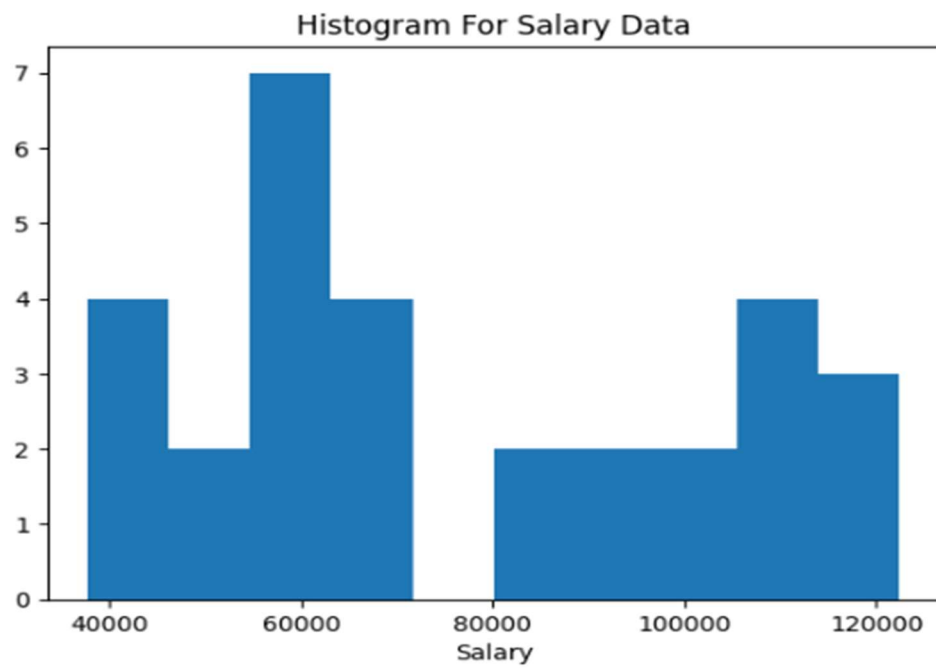
Output :



Bar Chart



Scatter Plot



Histogram

Classification Example 1

Question: Data Analysis

1. Implement SVM Classifier on iris dataset.
2. Implement KNN classifier on iris dataset.
3. Implement Decision Tree Classifier on iris dataset.

Code :

```
import numpy as np
import pandas as pd
from sklearn.datasets import load_iris
import matplotlib.pyplot as plt
%matplotlib inline

iris = load_iris()
iris.feature_names
iris.target_names
df = pd.DataFrame(iris.data,columns=iris.feature_names)
df.head()

df['target'] = iris.target
df.head()

df[df.target==1].head()

df0 = df[:50]
df1 = df[50:100]
df2 = df[100:]

plt.xlabel('Sepal Length')
plt.ylabel('Sepal Width')
plt.scatter(df0['sepal length (cm)'],df0['sepal width (cm)'],color="green",marker='+')
```



```
plt.scatter(df1['sepal length (cm)'],df1['sepal width (cm)'],color="blue",marker='.')

plt.xlabel('petal length')
plt.ylabel('petal width')
plt.scatter(df0['petal length (cm)'],df0['petal width (cm)'],color="green",marker='+')
plt.scatter(df1['petal length (cm)'],df1['petal width (cm)'],color="blue",marker='.')
```

```
from sklearn.model_selection import train_test_split
x = df.drop('target', axis='columns')
y = df.target
X_train, X_test, y_train, y_test = train_test_split(x, y, test_size=0.2)
print(len(X_train))
print(len(X_test))
```

Support Vector Machine Classification Algorithm

```
from sklearn.svm import SVC
model = SVC()
model.fit(X_train,y_train)
model.score(X_test,y_test)
model.predict([[4.8,3.0,1.5,0.3]])
```

K-Nearest Neighbor Classification Algorithm

```
from sklearn.neighbors import KNeighborsClassifier
knn = KNeighborsClassifier(n_neighbors=10)
knn.fit(X_train,y_train)
knn.score(X_test,y_test)
knn.predict([[4.8,3.0,1.5,0.3]])
```

Decision Tree Classification Algorithm

```
from sklearn.tree import DecisionTreeClassifier
```

```
t = DecisionTreeClassifier()
t.fit(X_train,y_train)
t.score(X_test,y_test)
t.predict([[4.8,3.0,1.5,0.5]])
```

Output :

```
['sepal length (cm)',
'sepal width (cm)',
'petal length (cm)',
'petal width (cm)']
```

```
array(['setosa', 'versicolor', 'virginica'], dtype='<U10')
```

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)
0	5.1	3.5	1.4	0.2
1	4.9	3.0	1.4	0.2
2	4.7	3.2	1.3	0.2
3	4.6	3.1	1.5	0.2
4	5.0	3.6	1.4	0.2

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	target
0	5.1	3.5	1.4	0.2	0
1	4.9	3.0	1.4	0.2	0
2	4.7	3.2	1.3	0.2	0
3	4.6	3.1	1.5	0.2	0
4	5.0	3.6	1.4	0.2	0

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	target
50	7.0	3.2	4.7	1.4	1
51	6.4	3.2	4.5	1.5	1
52	6.9	3.1	4.9	1.5	1

53	5.5	2.3	4.0	1.3	1
----	-----	-----	-----	-----	---

54	6.5	2.8	4.6	1.5	1
----	-----	-----	-----	-----	---

120

30

SVC()

0.9666666666666667

KNeighborsClassifier(n_neighbors=10)

0.9

array([0])

DecisionTreeClassifier()

0.9333333333333333

array([0])

Clustering Example 1

Question: Machine Learning

1. Implement clustering algorithm on iris dataset and display it using scatter plot.

Code :

```
import pandas as pd
from sklearn.preprocessing import MinMaxScaler
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt
from sklearn.datasets import load_iris
%matplotlib inline

iris = load_iris()
df = pd.DataFrame(iris.data,columns=iris.feature_names)
df.head()

df['flower'] = iris.target
df.head()
df.drop(['sepal length (cm)','sepal width (cm)','flower'],axis='columns',inplace=True)
df.head(3)
km = KMeans(n_clusters=3)
yp = km.fit_predict(df)
yp
df['cluster'] = yp
df.head(2)
df.cluster.unique()
df0 = df[df.cluster==0]
df1 = df[df.cluster==1]
df2 = df[df.cluster==2]
plt.scatter(df0['petal length (cm)'],df0['petal width (cm)'], color='blue')
plt.scatter(df1['petal length (cm)'],df1['petal width (cm)'], color='green')
plt.scatter(df2['petal length (cm)'],df2['petal width (cm)'], color='yellow')
```

Output :

sepal length (cm)			sepal width (cm)		petal length (cm)	petal width (cm)
0	5.1	3.5	1.4	0.2		
1	4.9	3.0	1.4	0.2		
2	4.7	3.2	1.3	0.2		
3	4.6	3.1	1.5	0.2		
4	5.0	3.6	1.4	0.2		

sepal length (cm)			sepal width (cm)			petal length (cm)	petal width (cm)	flower
0	5.1	3.5	1.4	0.2	0			
1	4.9	3.0	1.4	0.2	0			
2	4.7	3.2	1.3	0.2	0			
3	4.6	3.1	1.5	0.2	0			
4	5.0	3.6	1.4	0.2	0			

petal length (cm)		petal width (cm)
0	1.4	0.2
1	1.4	0.2
2	1.3	0.2

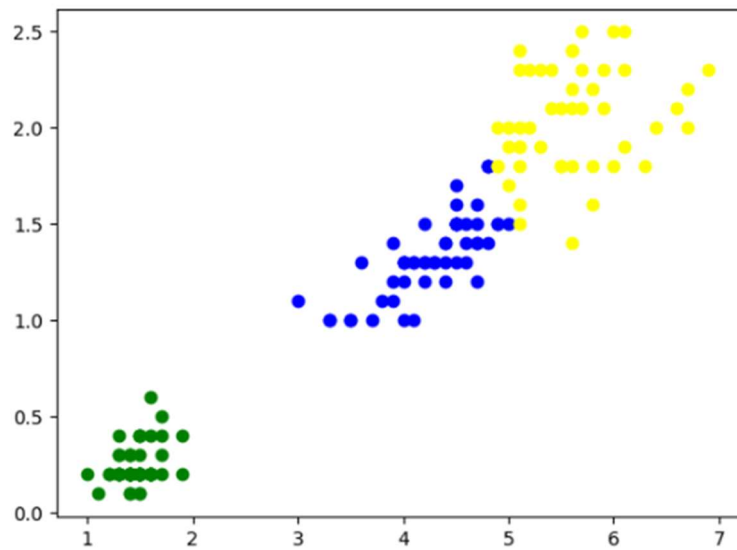
```
array([1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,  
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,  
       1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,  
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2, 0, 0, 0, 0, 0, 2, 0, 0, 0,  
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2, 2, 2, 2, 2, 2, 0, 2, 2, 2,  
       2, 2, 2, 2, 2, 2, 2, 2, 2, 0, 2, 2, 2, 2, 2, 2, 0, 2, 2, 2, 2,  
       2, 2, 2, 2, 2, 2, 0, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2], dtype=int32)
```

petal length (cm) petal width (cm) cluster

```
0    1.4    0.2    1
1    1.4    0.2    1
```

```
array([1, 0, 2], dtype=int32)
```

```
Out[12]: <matplotlib.collections.PathCollection at 0x7fd8f88b6380>
```



```
In [13]: ccc = f1
```

Clustering Example 2

Question : Data Analysis

1. Implement clustering algorithm on “income.csv” dataset and display it using scatter plot.

Code :

```
import matplotlib.pyplot as plt
from sklearn.preprocessing import MinMaxScaler
from sklearn.cluster import KMeans
import pandas as pd
%matplotlib inline

df = pd.read_csv("income.csv")
df.head()
plt.scatter(df.Age,df['Income($)'])
plt.xlabel('Age')
plt.ylabel('Income($)')

km = KMeans(n_clusters=3)
y_predicted = km.fit_predict(df[['Age','Income($)']])
y_predicted

df['cluster'] = y_predicted
df.head()
km.cluster_centers_

df0 = df[df.cluster==0]
df1 = df[df.cluster==1]
df2 = df[df.cluster==2]
plt.scatter(df0.Age,df0['Income($)'],color = 'green')
plt.scatter(df1.Age,df1['Income($)'],color = 'red')
plt.scatter(df2.Age,df2['Income($)'],color = 'black')
plt.xlabel('Age')
plt.ylabel('Income($)')
```

```
plt.legend()
```

```
scalar = MinMaxScaler()
```

```
scalar.fit(df[['Income($)']])
```

```
df['Income($)'] = scalar.transform(df[['Income($)']])
```

```
scalar.fit(df[['Age']])
```

```
df['Age'] = scalar.transform(df[['Age']])
```

```
df.head()
```

```
plt.scatter(df.Age,df['Income($)'])
```

```
km = KMeans(n_clusters=3)
```

```
y_predicted = km.fit_predict(df[['Age','Income($)']])
```

```
y_predicted
```

```
df['cluster'] = y_predicted
```

```
df.head()
```

```
df0 = df[df.cluster==0]
```

```
df1 = df[df.cluster==1]
```

```
df2 = df[df.cluster==2]
```

```
plt.scatter(df0.Age,df0['Income($)'],color = 'green')
```

```
plt.scatter(df1.Age,df1['Income($)'],color = 'red')
```

```
plt.scatter(df2.Age,df2['Income($)'],color = 'black')
```

```
plt.xlabel('Age')
```

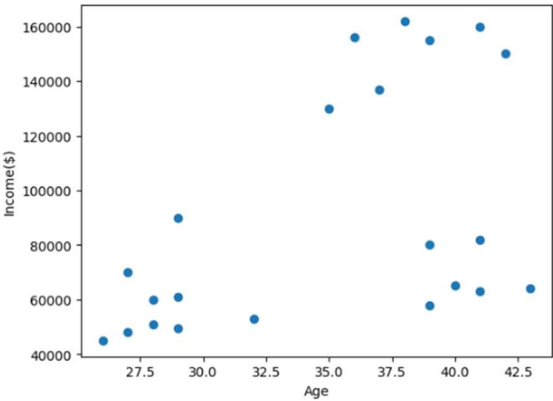
```
plt.ylabel('Income($)')
```

```
plt.legend()
```


Output :

	Name	Age	Income(\$)
0	Rob	27	70000
1	Michael	29	90000
2	Mohan	29	61000
3	Ismail	28	60000
4	Kory	42	150000

```
In [3]: plt.scatter(df.Age,df['Income($)'])
plt.xlabel('Age')
plt.ylabel('Income($)')
Out[3]: Text(0, 0.5, 'Income($)')
```



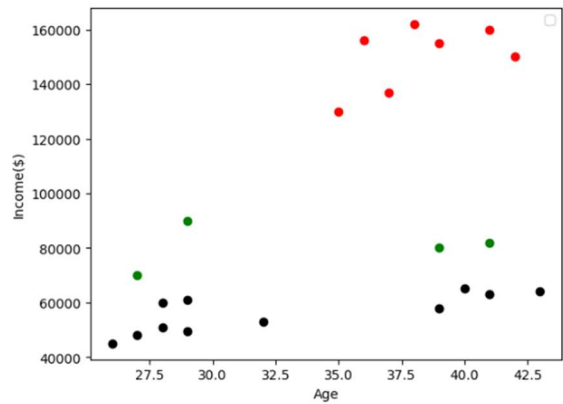
array([0, 0, 2, 2, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 0, 0, 2],
dtype=int32)

	Name	Age	Income(\$)	cluster
0	Rob	27	70000	0
1	Michael	29	90000	0
2	Mohan	29	61000	2
3	Ismail	28	60000	2
4	Kory	42	150000	1

array([[3.40000000e+01, 8.05000000e+04],

[3.82857143e+01, 1.50000000e+05],
[3.29090909e+01, 5.61363636e+04]])

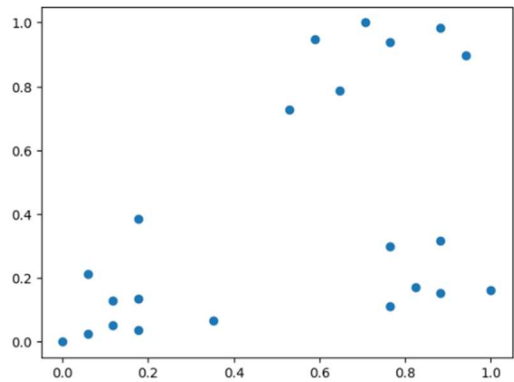
Out[7]: <matplotlib.legend.Legend at 0x7f67ad605660>



	Name	Age	Income(\$)	cluster
0	Rob	0.058824	0.213675	0
1	Michael	0.176471	0.384615	0
2	Mohan	0.176471	0.136752	2
3	Ismail	0.117647	0.128205	2
4	Kory	0.941176	0.897436	1

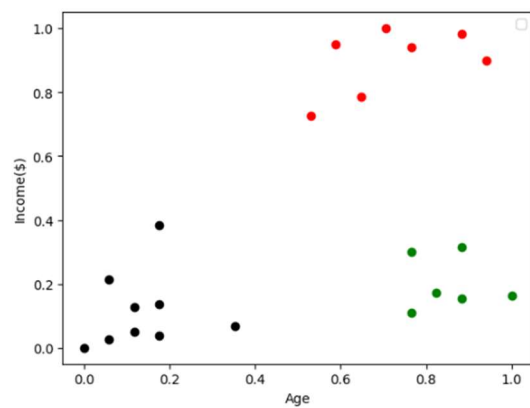
In [10]: plt.scatter(df.Age,df['Income(\$)'])

Out[10]: <matplotlib.collections.PathCollection at 0x7f67a834d5a0>



Name	Age	Income(\$)	cluster
0	Rob	0.058824	0.213675
1	Michael	0.176471	0.384615
2	Mohan	0.176471	0.136752
3	Ismail	0.117647	0.128205
4	Kory	0.941176	0.897436

Out[14]: <matplotlib.legend.Legend at 0x7f67a8219420>



Python Program List

1. **Write a python program to accept input string from user and display number of vowels and consonant in string.**

Input : Beautiful

Output :

Total Vowels : 5

Total Consonant: 4

2. **Write a program to find the quadrants in which coordinates lies Get the value of x and y coordinates as input from the user and check in which quadrants the point lies and print it.**

Input:

10 20

Output:

This point lies in first quadrant

Input:

-10 20

Output:

This point lies in second quadrant number

3. **Write a python program to accept input from user and check whether number is Armstrong or not.**

Input: 151

Output: 151 is Armstrong number

Input: 159

Output: 159 is not Armstrong number

4. **Write a python program to count the occurrence of each word in a given sentence**

Input:

Enter a string: python is a good python program

Output:

python=2

is=1

a=2

good=1

program=1

5. **Write python program to check whether given number strong or not.(accept user input)**

Input:

145

Output:

It is Strong Number

Explanation:

$145 = 1! + 4! + 5!$

6. **Write a python program to print prime number between 1 to 100**

Output:

2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97

7. Write program to check whether number is Harshad number or Not.

Input:

Enter number : 21

Output:

It is Harshad Number

Explanation:

The sum of digit of 21 is $2+1=3$, as number 21 is divisible by 3 so it is Harshad Number

8. Write python program to accept number and check whether number is Automorphic or not.

Input:

Enter number: 5

Output:

It's an Automorphic number.

Explanation:

Number=5 and square of number=25, As square of number ends with number itself,

It's an Automorphic number.

9. Write a Python program to print factorial of number using Recursion.

Input:

Enter number: 5

Output:

Factorial of 5 is 120

10. Write a program to print length of String using Recursion.

Input:

Enter number: Trinity

Output:

Length of Given String is 7