```
from sklearn.cluster import KMeans

import pandas as pd


from sklearn.preprocessing import MinMaxScaler
```

New Section

```
from matplotlib import pyplot as plt

%matplotlib inline

df=pd.read_csv("/content/KMeans Dataset.csv")
```

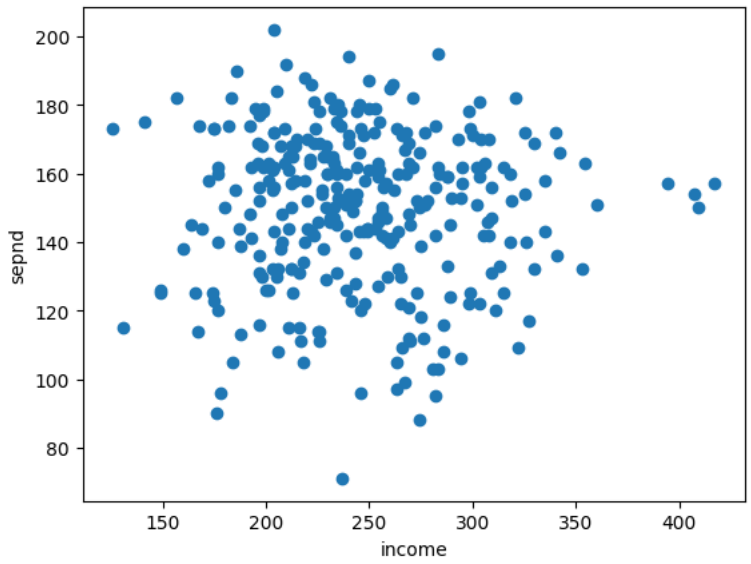
+ Code + Text

```
df.head()
```

	INCOME	SPEND
0	233	150
1	250	187
2	204	172
3	236	178
4	354	163

```
plt.scatter(df.INCOME,df['SPEND'])
plt.xlabel('income')
plt.ylabel('sepnd')
```

```
Text(0, 0.5, 'sepnd')
```




```
km = KMeans(n_clusters=3)
y_predicted = km.fit_predict(df[['INCOME','SPEND']])
y_predicted
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of `n_init` will change fr
  warnings.warn(
array([2, 2, 0, 2, 1, 0, 1, 2, 0, 0, 2, 2, 2, 0, 2, 0, 1, 2, 2, 2, 2, 2,
       2, 2, 0, 1, 0, 0, 1, 0, 0, 0, 0, 2, 0, 0, 1, 2, 2, 1, 1, 2, 0, 2,
       1, 1, 2, 2, 0, 2, 2, 1, 2, 0, 2, 0, 0, 2, 0, 1, 2, 1, 0, 0, 0, 0,
       0, 2, 0, 0, 2, 2, 0, 2, 0, 2, 2, 0, 0, 2, 2, 1, 1, 1, 2, 2, 2, 0,
       0, 2, 2, 0, 0, 1, 0, 2, 1, 2, 1, 2, 2, 2, 0, 2, 0, 0, 2, 2, 2, 2,
       1, 0, 1, 0, 2, 0, 0, 0, 0, 2, 1, 2, 2, 2, 0, 0, 0, 2, 0, 2, 0, 2,
       1, 2, 1, 2, 0, 0, 0, 2, 1, 1, 0, 0, 0, 2, 2, 2, 2, 0, 2, 0, 2, 2,
       0, 0, 2, 0, 0, 0, 2, 1, 0, 0, 0, 1, 2, 2, 2, 0, 2, 2, 2, 0, 0, 0,
       2, 1, 0, 2, 1, 2, 1, 2, 2, 1, 2, 2, 2, 0, 1, 0, 0, 2, 0, 1, 2, 2,
       2, 2, 0, 2, 2, 2, 0, 2, 2, 2, 0, 0, 2, 2, 0, 1, 2, 1, 2, 1, 2, 2,
       1, 0, 2, 1, 2, 0, 2, 0, 1, 1, 2, 1, 1, 2, 1, 1, 1, 1, 1, 2, 2, 2,
       0, 2, 0, 2, 1, 2, 2, 2, 1, 2, 1, 1, 2, 1, 2, 0, 2, 2, 2, 2, 2, 2,
```

```
       0, 0, 1, 0, 1, 2, 2, 2, 2, 2, 2, 0, 0, 2, 1, 0, 1, 0, 0, 0, 0, 1,
       0, 2, 1, 0, 0, 1, 2, 0, 0, 0, 0, 0, 2, 2, 0, 0, 2], dtype=int32)
```

```
df['cluster']=y_predicted
df.head()
```

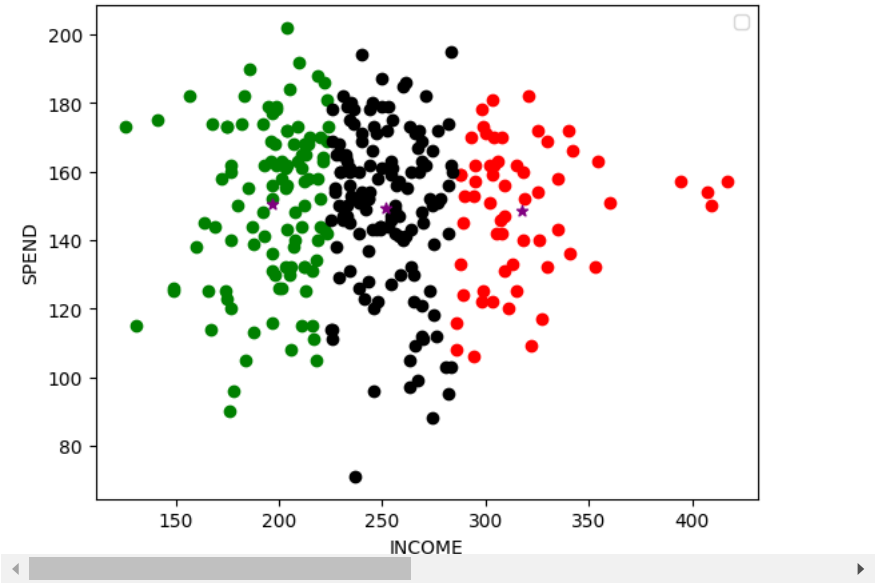
	INCOME	SPEND	cluster
0	233	150	2
1	250	187	2
2	204	172	0
3	236	178	2
4	354	163	1

```
km.cluster_centers_
```

```
array([[196.68224299, 150.57943925],
       [317.55      , 148.73333333],
       [251.61764706, 149.31617647]])
```

```
df1 = df[df.cluster==0]
df2 = df[df.cluster==1]
df3 = df[df.cluster==2]
plt.scatter(df1.INCOME,df1['SPEND'],color='green')
plt.scatter(df2.INCOME,df2['SPEND'],color='red')
plt.scatter(df3.INCOME,df3['SPEND'],color='black')
plt.scatter(km.cluster_centers_[:,0],km.cluster_centers_[:,1],color='purple',marker='*')
plt.xlabel('INCOME')
plt.ylabel('SPEND')
plt.legend()
```

```
WARNING:matplotlib.legend:No artists with labels found to put in legend.  Note that a
<matplotlib.legend.Legend at 0x7ae1b44fece0>
```

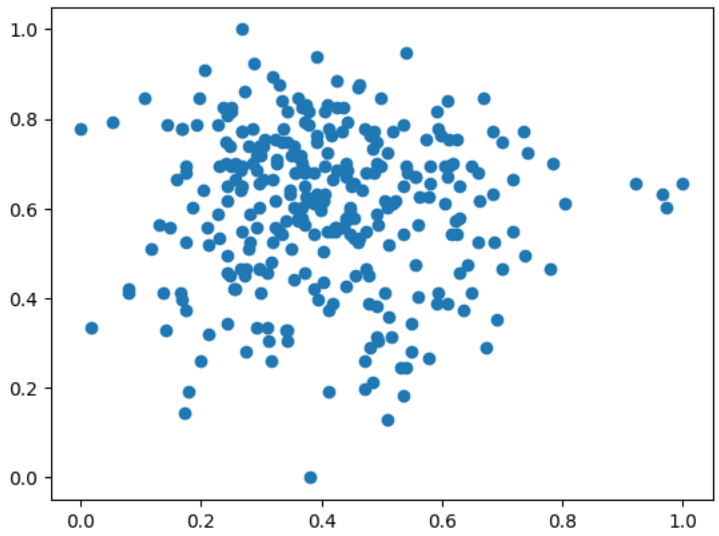



```
scaler = MinMaxScaler()
scaler.fit(df[['SPEND']])
df['SPEND'] = scaler.transform(df[['SPEND']])
scaler.fit(df[['INCOME']])
df['INCOME'] = scaler.transform(df[['INCOME']])
df.head()
```

	INCOME	SPEND	cluster
0	0.367698	0.603053	2
1	0.426117	0.885496	2
2	0.268041	0.770992	0
3	0.378007	0.816794	2
4	0.783505	0.702290	1

```
plt.scatter(df.INCOME,df['SPEND'])
```

```
<matplotlib.collections.PathCollection at 0x7ae1b1a10400>
```




```
km = KMeans(n_clusters=3)
y_predicted = km.fit_predict(df[['INCOME','SPEND']])
y_predicted
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of `n_init` will change fr
  warnings.warn(
array([2, 2, 2, 2, 1, 2, 1, 2, 2, 2, 2, 1, 2, 2, 1, 2, 1, 0, 2, 2, 2, 2,
       2, 0, 2, 1, 2, 0, 1, 2, 2, 0, 2, 1, 0, 2, 1, 2, 1, 1, 1, 2, 2, 1,
       1, 1, 2, 1, 0, 2, 1, 1, 2, 2, 2, 2, 2, 2, 2, 1, 0, 1, 2, 0, 2, 2,
       0, 2, 2, 2, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 1, 1, 0, 1, 1, 2,
       2, 0, 2, 2, 2, 1, 0, 0, 1, 2, 1, 2, 2, 1, 2, 2, 2, 0, 0, 2, 2, 2,
       1, 2, 1, 2, 1, 2, 2, 2, 2, 2, 1, 2, 2, 1, 2, 2, 0, 1, 2, 0, 2, 1,
       1, 2, 1, 1, 0, 0, 0, 0, 1, 1, 2, 0, 0, 0, 2, 2, 2, 2, 0, 0, 2, 1,
       2, 0, 2, 2, 2, 2, 2, 1, 2, 2, 2, 0, 0, 1, 1, 2, 1, 2, 1, 2, 0, 0,
       2, 1, 0, 0, 1, 0, 1, 2, 0, 1, 1, 0, 2, 2, 1, 0, 0, 1, 2, 1, 2, 2,
       0, 2, 2, 1, 0, 1, 2, 2, 0, 1, 0, 2, 2, 1, 0, 1, 1, 1, 0, 1, 0, 1,
       1, 0, 1, 1, 0, 0, 0, 0, 1, 1, 2, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1,
       0, 0, 0, 1, 1, 0, 2, 1, 1, 0, 0, 1, 0, 1, 0, 0, 2, 2, 2, 2, 0, 1,
       0, 0, 1, 0, 0, 0, 1, 2, 0, 2, 0, 2, 0, 1, 1, 0, 1, 2, 0, 2, 0, 1,
       2, 2, 1, 0, 2, 1, 2, 2, 2, 2, 0, 0, 0, 0, 0, 0, 2], dtype=int32)
```

```
df['cluster']=y_predicted
df.head()
```

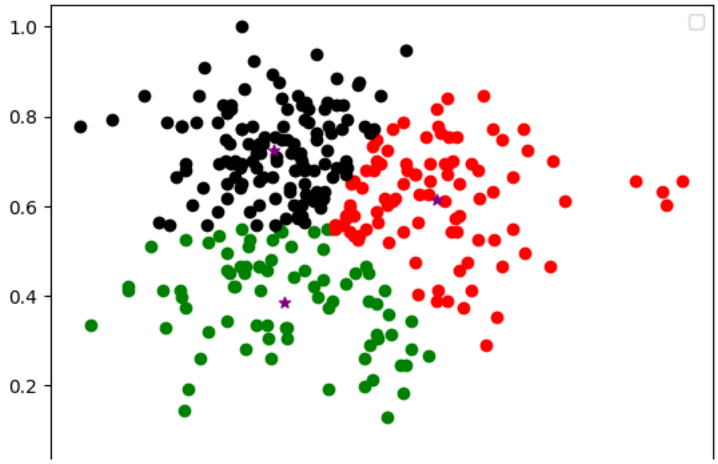
	INCOME	SPEND	cluster
0	0.367698	0.603053	2
1	0.426117	0.885496	2
2	0.268041	0.770992	2
3	0.378007	0.816794	2
4	0.783505	0.702290	1

```
km.cluster_centers_
```

```
array([[0.3395189 , 0.38406489],
       [0.59168677, 0.61385415],
       [0.32102613, 0.72465826]])
```

```
df1 = df[df.cluster==0]
df2 = df[df.cluster==1]
df3 = df[df.cluster==2]
plt.scatter(df1.INCOME,df1['SPEND'],color='green')
plt.scatter(df2.INCOME,df2['SPEND'],color='red')
plt.scatter(df3.INCOME,df3['SPEND'],color='black')
plt.scatter(km.cluster_centers_[:,0],km.cluster_centers_[:,1],color='purple',marker='*')
plt.legend()
```

```
WARNING:matplotlib.legend:No artists with labels found to put in legend.  Note that a
<matplotlib.legend.Legend at 0x7ae1b44fd3f0>
```




```
sse = []
k_rng = range(1,10)
for k in k_rng:
  km = KMeans(n_clusters=k)
  km.fit(df[['INCOME','SPEND']])
  sse.append(km.inertia_) # inertia will calculate SSE

plt.xlabel('K')
plt.ylabel('Sum of squared error')
plt.plot(k_rng,sse)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning
  warnings.warn(
[<matplotlib.lines.Line2D at 0x7ae1b1838ca0>]
```