CS23334-FUNDAMENTALS OF DATA SCIENCE

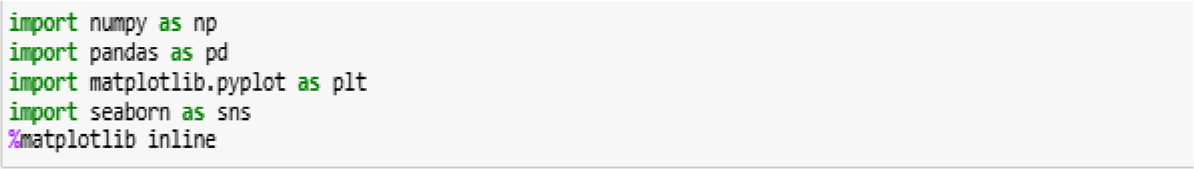
DEVA DHARSHINI P (240701107)

10.) K MEANS CLUSTERING

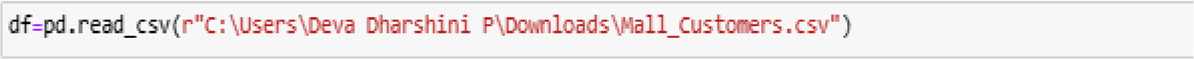
Aim:

To implement K-Means Clustering to group similar data points into clusters based on their features.

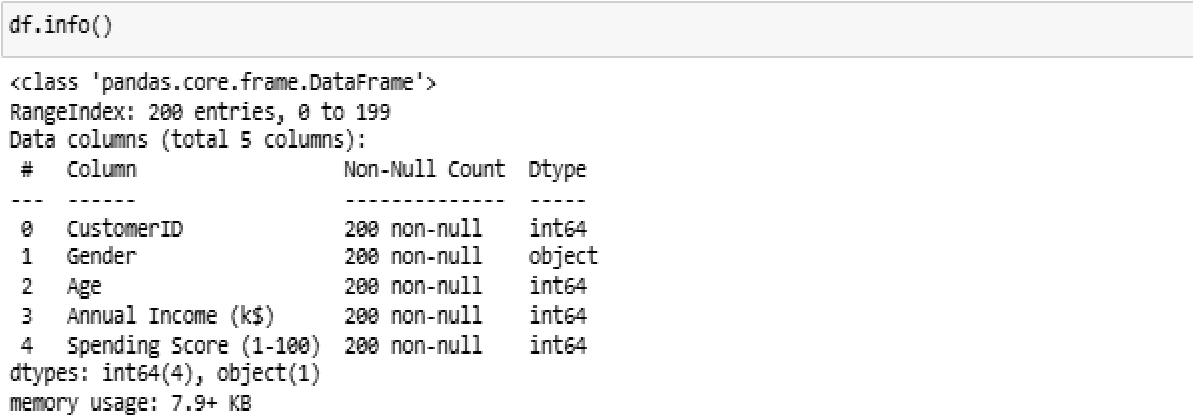
Code:



```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

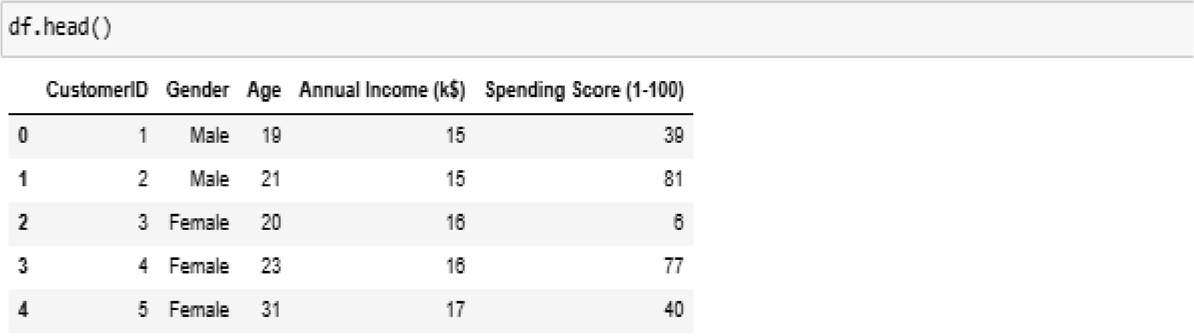


```
df=pd.read_csv(r"C:\Users\Deva Dharshini P\Downloads\Mall_Customers.csv")
```



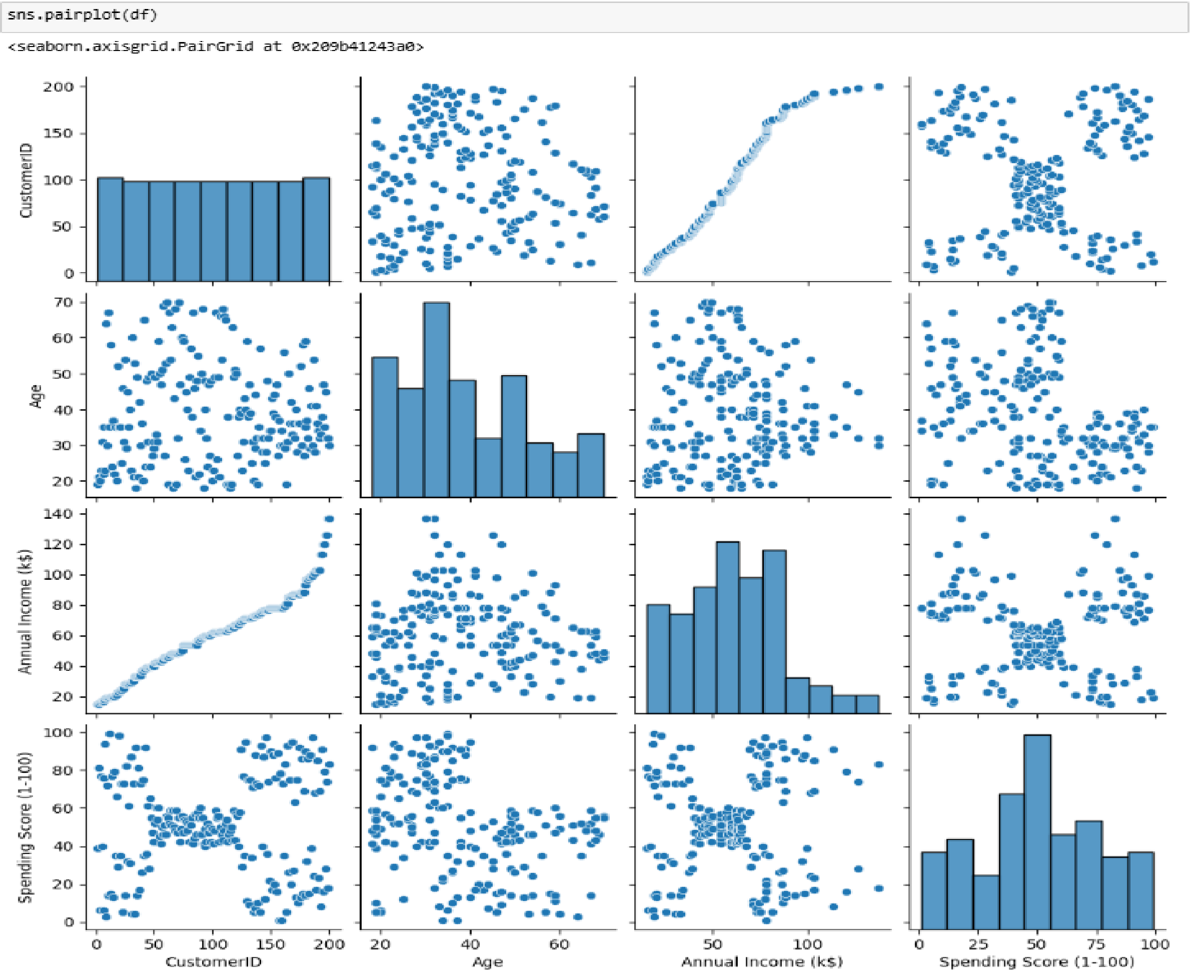
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 5 columns):
 #   Column                  Non-Null Count  Dtype 
---  ------                  --------------  ----- 
 0   CustomerID              200 non-null    int64 
 1   Gender                  200 non-null    object
 2   Age                     200 non-null    int64 
 3   Annual Income (k$)      200 non-null    int64 
 4   Spending Score (1-100)  200 non-null    int64 
dtypes: int64(4), object(1)
memory usage: 7.9+ KB
```



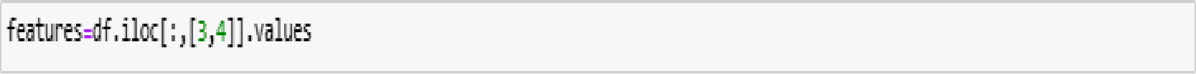
```
df.head()
```

	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)
0	1	Male	19	15	39
1	2	Male	21	15	81
2	3	Female	20	16	6
3	4	Female	23	16	77
4	5	Female	31	17	40

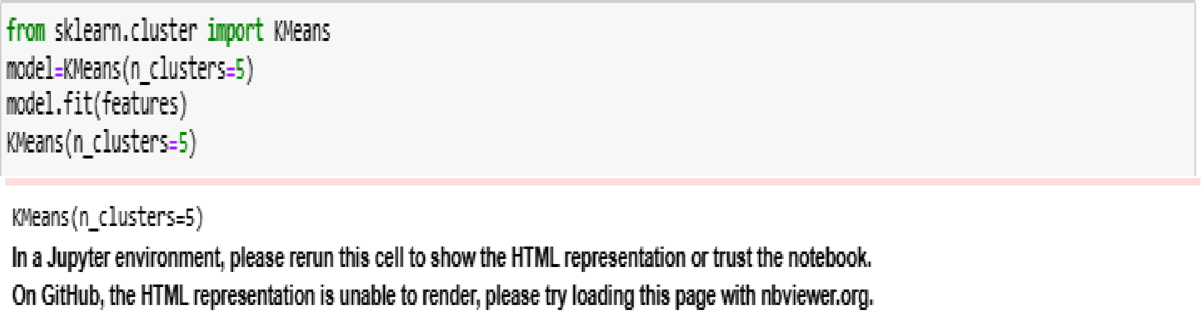


```
sns.pairplot(df)
```

```
<seaborn.axisgrid.PairGrid at 0x209b41243a0>
```



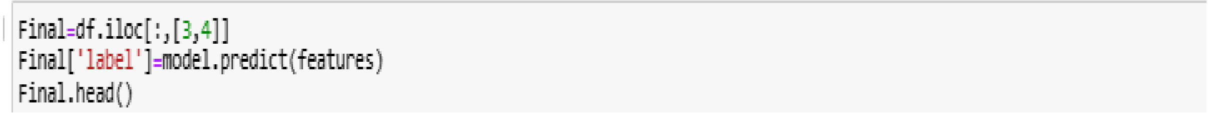
```
features=df.iloc[:,[3,4]].values
```



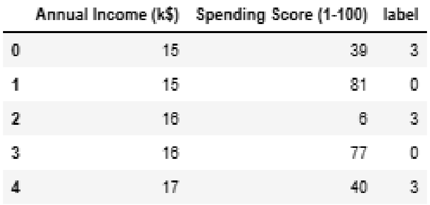
```
from sklearn.cluster import KMeans
model=KMeans(n_clusters=5)
model.fit(features)
KMeans(n_clusters=5)
```

```
KMeans(n_clusters=5)
```

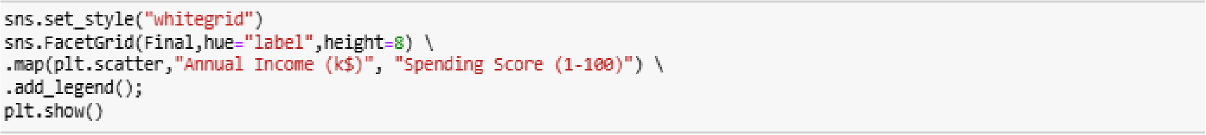
In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.
On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.



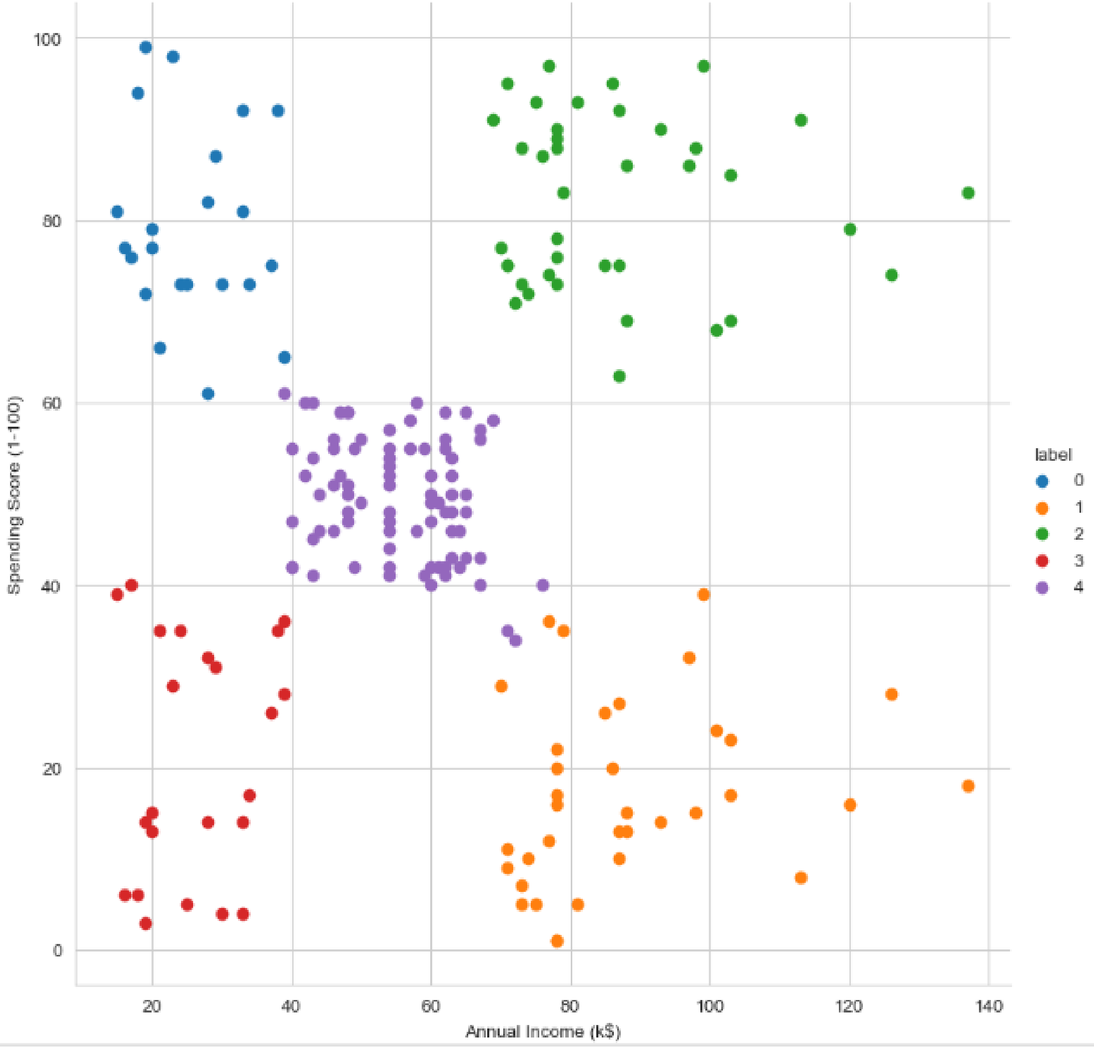
```
Final=df.iloc[:,[3,4]]
Final['label']=model.predict(features)
Final.head()
```

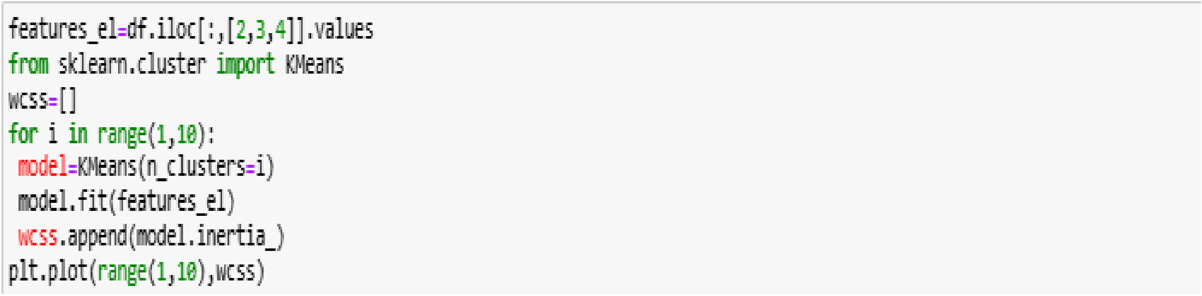


	Annual Income (k$)	Spending Score (1-100)	label
0	15	39	3
1	15	81	0
2	16	6	3
3	16	77	0
4	17	40	3



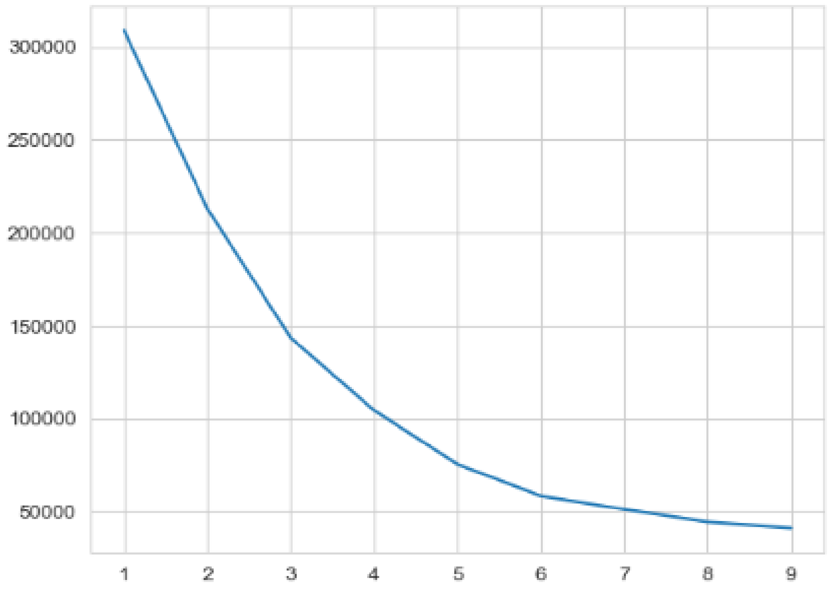
```
sns.set_style("whitegrid")
sns.FacetGrid(Final,hue="label",height=8) \
.map(plt.scatter,"Annual Income (k$)", "Spending Score (1-100)") \
.add_legend();
plt.show()
```





```
features_el=df.iloc[:,[2,3,4]].values
from sklearn.cluster import KMeans
wcss=[]
for i in range(1,10):
 model=KMeans(n_clusters=i)
 model.fit(features_el)
 wcss.append(model.inertia_)
plt.plot(range(1,10),wcss)
```

```
[<matplotlib.lines.Line2D at 0x209b7bd08e0>]
```



Result:
The K-Means algorithm successfully clustered the dataset into distinct groups, showing clear separation between different clusters.