+ Code + Text

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


df = pd.read_csv(r'https://github.com/YBI-Foundation/Dataset/raw/main/Customer%20Segmentation.csv')


#df = pd.read_csv(r'C:\Users\YBI foundation\Desktop\Customer Segmentation.csv)


#df = pd.read_csv(r'/content/Customer Segmentation.csv')


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

```
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.describe()
```

	CustomerID	Age	Annual Income (k$)	Spending Score (1-100)
count	200.000000	200.000000	200.000000	200.000000
mean	100.500000	38.850000	60.560000	50.200000
std	57.879185	13.969007	26.264721	25.823522
min	1.000000	18.000000	15.000000	1.000000
25%	50.750000	28.750000	41.500000	34.750000
50%	100.500000	36.000000	61.500000	50.000000
75%	150.250000	49.000000	78.000000	73.000000
max	200.000000	70.000000	137.000000	99.000000

```
df.columns
```

```
Index(['CustomerID', 'Gender', 'Age', 'Annual Income (k$)',
       'Spending Score (1-100)'],
      dtype='object')
```

```
df.shape
```

```
(200, 5)
```

```
x = df[['Age','Spending Score (1-100)']].values
```

```
x.shape
```

```
(200, 2)
```

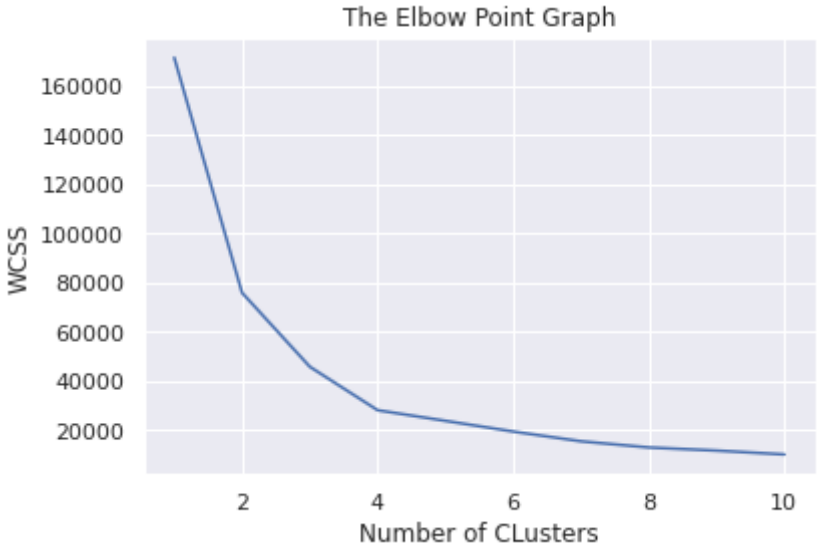
```
x
```

```
array([[19, 39],
       [21, 81],
       [20,  6],
       [23, 77],
       [31, 40],
       [22, 76],
       [35,  6],
       [23, 94],
       [64,  3],
       [30, 72],
       [67, 14],
       [35, 99],
       [58, 15],
       [24, 77],
       [37, 13],
       [22, 79],
       [35, 35],
       [20, 66],
       [52, 29],
       [35, 98],
       [35, 35],
       [25, 73],
       [46,  5],
       [31, 73],
       [54, 14],
       [29, 82],
       [45, 32],
       [35, 61],
       [40, 31],
       [23, 87],
       [60,  4],
       [21, 73],
```

```
               [53,  4],
               [18, 92],
               [49, 14],
               [21, 81],
               [42, 17],
               [30, 73],
               [36, 26],
               [20, 75],
               [65, 35],
               [24, 92],
               [48, 36],
               [31, 61],
               [49, 28],
               [24, 65],
               [50, 55],
               [27, 47],
               [29, 42],
               [31, 42],
               [49, 52],
               [33, 60],
               [31, 54],
               [59, 60],
               [50, 45],
               [47, 41],
               [51, 50],
               [69, 46]
```

```
from sklearn.cluster import KMeans


wcss = []

for i in range(1,11):
  kmeans = KMeans(n_clusters=i, init='k-means++', random_state=122529)
  kmeans.fit(x)

  wcss.append(kmeans.inertia_)


sns.set()
```

```
plt.plot(range(1,11),wcss)
plt.title('The Elbow Point Graph')
plt.xlabel('Number of CLusters')
plt.ylabel('WCSS')
plt.show()
```




```
kmeans = KMeans(n_clusters=5, init='k-means++', random_state=122529)
```

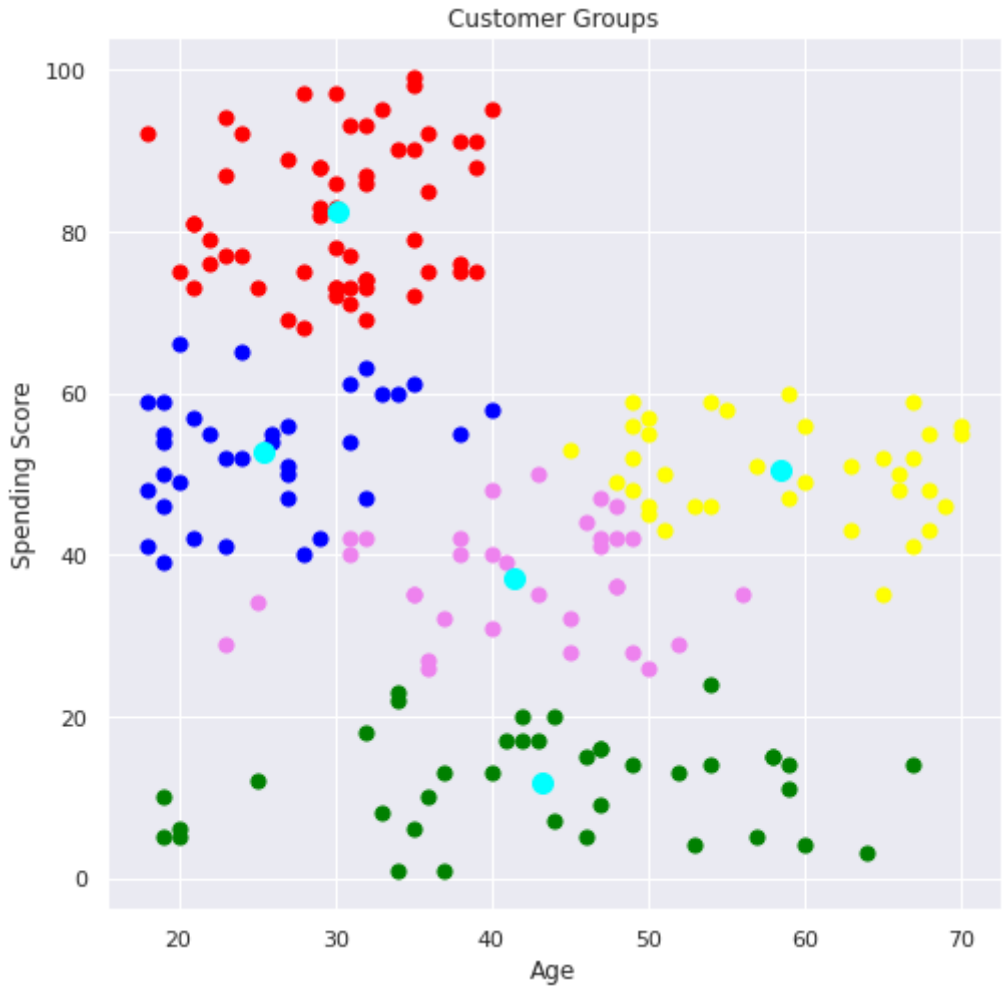
```
#return a label for each point based on their cluster
y = kmeans.fit_predict(x)
```

```
y
```

```
array([4, 1, 0, 1, 3, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 3, 4, 3, 1, 3, 1,
       0, 1, 0, 1, 3, 4, 3, 1, 0, 1, 0, 1, 0, 1, 0, 1, 3, 1, 2, 1, 3, 4,
       3, 4, 2, 4, 4, 3, 2, 4, 4, 2, 2, 3, 2, 2, 4, 2, 2, 4, 2, 2, 2, 4,
       3, 2, 4, 4, 2, 3, 2, 2, 2, 4, 2, 3, 4, 3, 2, 4, 2, 3, 4, 3, 2, 4,
       4, 2, 2, 4, 2, 3, 3, 4, 3, 4, 3, 4, 4, 2, 2, 4, 2, 4, 2, 2, 2, 2,
       2, 4, 3, 4, 4, 4, 2, 2, 2, 2, 4, 3, 4, 1, 3, 1, 3, 1, 0, 1, 0, 1,
       3, 1, 0, 1, 0, 1, 0, 1, 0, 1, 4, 1, 0, 1, 3, 1, 0, 1, 0, 1, 0, 1,
       0, 1, 0, 1, 0, 1, 3, 1, 0, 1, 3, 1, 0, 1, 3, 4, 0, 1, 0, 1, 0, 1,
```

```
       0, 1, 0, 1, 3, 1, 0, 1, 3, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 3, 1,
       0, 1], dtype=int32)
```

```
plt.figure(figsize=(8,8))
plt.scatter(x[y==0,0], x[y==0,1], s=50, c='green', label='Cluster 1')
plt.scatter(x[y==1,0], x[y==1,1], s=50, c='red', label='Cluster 2')
plt.scatter(x[y==2,0], x[y==2,1], s=50, c='yellow',label='Cluster 3')
plt.scatter(x[y==3,0], x[y==3,1], s=50, c='violet', label='Cluster 4')
plt.scatter(x[y==4,0], x[y==4,1], s=50, c='blue', label='Cluster 5')

plt.scatter(kmeans.cluster_centers_[:,0], kmeans.cluster_centers_[:,1], s=100, c='cyan', label='Centroids')

plt.title('Customer Groups')
plt.xlabel('Age')
plt.ylabel('Spending Score')
plt.show()
```



✓ 0s completed at 20:45