```
#Name:Chaitanya Santosh Giram
#Roll no:9037
#TE IT
```

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

```
#Importing libraries
import pandas as pd
import numpy as np
```

```
data_set=pd.read_csv("/content/drive/MyDrive/Colab Notebooks/sample/Mall_Customers.csv")
data_set
```

	CustomerID	Genre	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
...	...	...	...	...	...
195	196	Female	35	120	79
196	197	Female	45	126	28
197	198	Male	32	126	74
198	199	Male	32	137	18
199	200	Male	30	137	83

200 rows × 5 columns

```
data_set.isnull().sum()
```

```
CustomerID              0
Genre                   0
Age                     0
Annual Income (k$)      0
Spending Score (1-100)  0
dtype: int64
```

```
X=data_set[['Annual Income (k$)','Spending Score (1-100)']].to_numpy()
X
```

```
       [ 76,  40],
       [ 76,  87],
       [ 77,  12],
       [ 77,  97],
       [ 77,  36],
       [ 77,  74],
       [ 78,  22],
       [ 78,  90],
       [ 78,  17],
       [ 78,  88],
       [ 78,  20],
       [ 78,  76],
       [ 78,  16],
       [ 78,  89],
       [ 78,   1],
       [ 78,  78],
       [ 78,   1],
       [ 78,  73],
       [ 79,  35],
       [ 79,  83],
       [ 81,   5],
       [ 81,  93],
       [ 85,  26],
       [ 85,  75],
       [ 86,  20],
       [ 86,  95],
       [ 87,  27],
       [ 87,  63],
       [ 87,  13],
       [ 87,  75],
       [ 87,  10],
```

```
[ 87,  10],
[ 87,  92],
[ 88,  13],
[ 88,  86],
[ 88,  15],
[ 88,  69],
[ 93,  14],
[ 93,  90],
[ 97,  32],
[ 97,  86],
[ 98,  15],
[ 98,  88],
[ 99,  39],
[ 99,  97],
[101,  24],
[101,  68],
[103,  17],
[103,  85],
[103,  23],
[103,  69],
[113,   8],
[113,  91],
[120,  16],
[120,  79],
[126,  28],
[126,  74],
[137,  18],
[137,  83]])
```

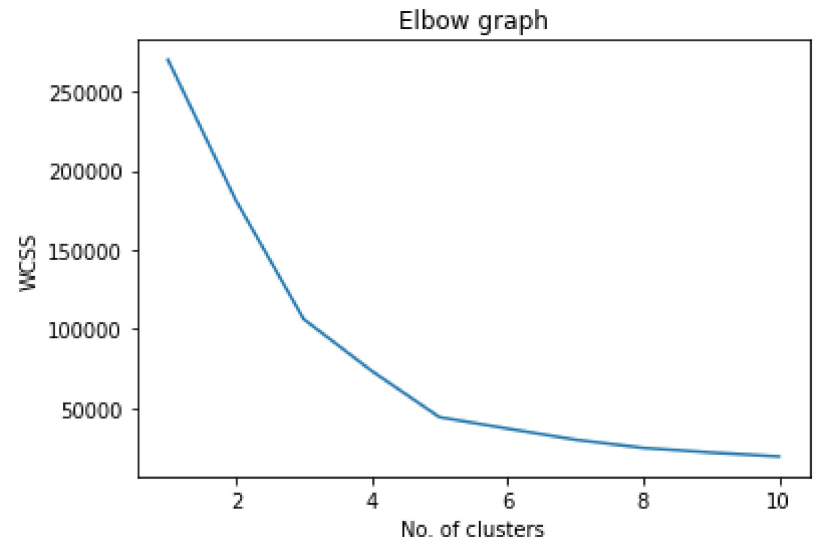
Part A

K-Means Clustering

```
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt

#wcss
wcss_list=[]
for i in range(1,11):
  km=KMeans(n_clusters=i)
  y_predict=km.fit_predict(X)
  wcss_list.append(km.inertia_)
plt.plot(range(1,11),wcss_list)
plt.title("Elbow graph")
plt.xlabel("No. of clusters")
plt.ylabel("WCSS")
```

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



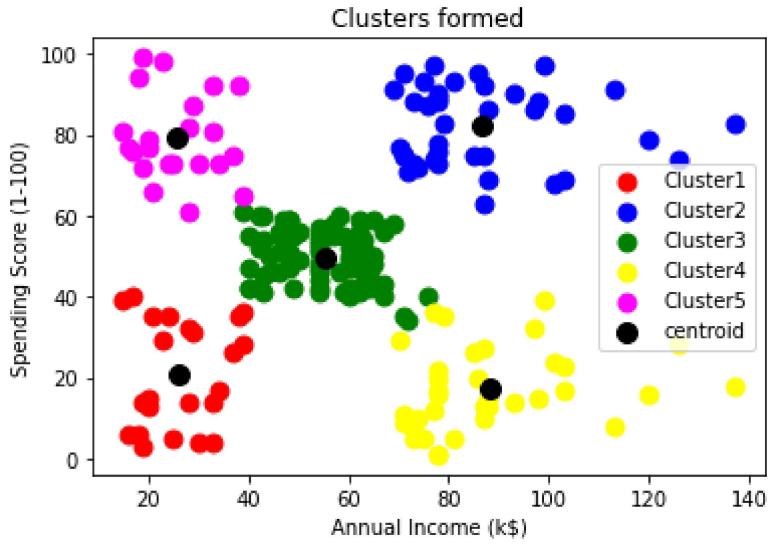

```
km=KMeans(n_clusters=5)
y_predict=km.fit_predict(X)
y_predict
```

```
array([0, 4, 0, 4, 0, 4, 0, 4, 0, 4, 0, 4, 0, 4, 0, 4, 0, 4, 0, 4,
       0, 4, 0, 4, 0, 4, 0, 4, 0, 4, 0, 4, 0, 4, 0, 4, 0, 4, 0, 2,
       0, 4, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
       2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
       2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
       2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 3, 1, 2, 1, 3, 1, 3, 1,
       2, 1, 3, 1, 3, 1, 3, 1, 3, 1, 2, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1,
       3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1,
       3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1, 3, 1,
       3, 1], dtype=int32)
```

```
plt.scatter(X[y_predict == 0,0] , X[y_predict==0,1],s=80, c='red', label='Cluster1')
plt.scatter(X[y_predict==1,0] , X[y_predict==1,1],s=80, c="blue", label="Cluster2")
```

```
plt.scatter(X[y_predict==1,0], X[y_predict==1,1],s=80, c="blue", label="Cluster2")
plt.scatter(X[y_predict==2,0], X[y_predict==2,1],s=80, c="green", label="Cluster3")
plt.scatter(X[y_predict==3,0], X[y_predict==3,1],s=80, c="yellow", label="Cluster4")
plt.scatter(X[y_predict==4,0], X[y_predict==4,1],s=80, c="magenta", label="Cluster5")
plt.scatter(km.cluster_centers_[:,0],km.cluster_centers_[:,1],s=100,c="black",label="centroid")
plt.title("Clusters formed")
plt.xlabel("Annual Income (k$)")
plt.ylabel("Spending Score (1-100)")
plt.legend()
```

```
<matplotlib.legend.Legend at 0x7f4cc20fead0>
```

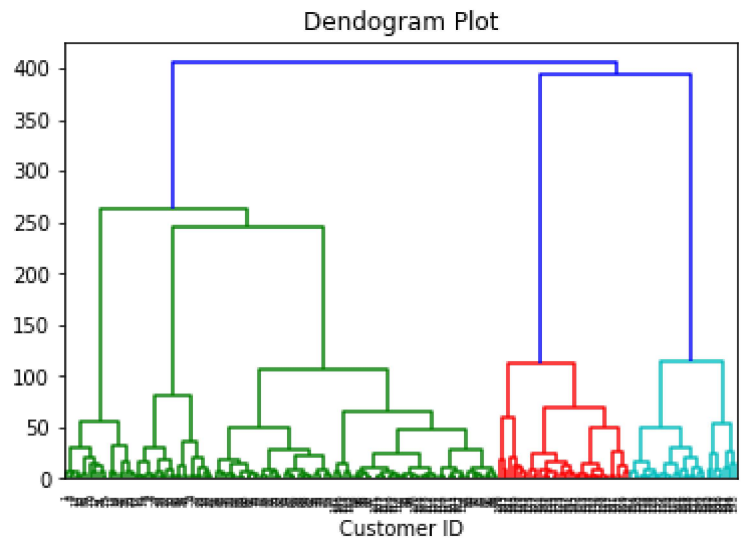


Part B

Agglomerative Clustering

```
from sklearn.cluster import AgglomerativeClustering
agc=AgglomerativeClustering(n_clusters=3)
y_predict=agc.fit_predict(X)
```

```
import scipy.cluster.hierarchy as shc
dendro = shc.dendrogram(shc.linkage(X,method="ward"))
plt.title("Dendogram Plot")
plt.xlabel("Euclidean Distances")
plt.xlabel("Customer ID")
plt.show()
```




```
plt.scatter(X[y_predict ==0,0], X[y_predict ==0,1], s=80, c="red",label="Cluster1")
plt.scatter(X[y_predict ==1,0],X[y_predict == 1,1],s=80, c="yellow",label="Cluster2")
plt.scatter(X[y_predict ==2,0],X[y_predict == 2,1],s=80, c="green",label="Cluster3")
plt.title("Clusters formed")
plt.xlabel("Annual Income (k$)")
plt.ylabel("Spending Score (1-100)")
plt.legend()
```

```
<matplotlib.legend.Legend at 0x7f5f92b7ec10>
```