

In [9]:

```
#Kmeans
```

In [10]:

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

In [11]:

```
from sklearn.cluster import KMeans
```

In [12]:

```
MallCostumers = pd.read_csv("C:/Users/jay/Desktop/MallCostumers.csv")
MallCostumers.head()
```

Out[12]:

	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

In [13]:

```
MallCostumers.head()
```

Out[13]:

	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

In [14]:

```
import random
import math
```

In [15]:

```
from sklearn.cluster import KMeans
```

In [16]:

```
X = MallCostumers.iloc[:, [3,4]].values
X
y = MallCostumers.iloc[:, [3,4]].values
```

y

Out[16]:

```
array([[ 15,  39],
       [ 15,  81],
       [ 16,   6],
       [ 16,  77],
       [ 17,  40],
       [ 17,  76],
       [ 18,   6],
       [ 18,  94],
       [ 19,   3],
       [ 19,  72],
       [ 19,  14],
       [ 19,  99],
       [ 20,  15],
       [ 20,  77],
       [ 20,  13],
       [ 20,  79],
       [ 21,  35],
       [ 21,  66],
       [ 23,  29],
       [ 23,  98],
       [ 24,  35],
       [ 24,  73],
       [ 25,   5],
       [ 25,  73],
       [ 28,  14],
       [ 28,  82],
       [ 28,  32],
       [ 28,  61],
       [ 29,  31],
       [ 29,  87],
       [ 30,   4],
       [ 30,  73],
       [ 33,   4],
       [ 33,  92],
       [ 33,  14],
       [ 33,  81],
       [ 34,  17],
       [ 34,  73],
       [ 37,  26],
       [ 37,  75],
       [ 38,  35],
       [ 38,  92],
       [ 39,  36],
       [ 39,  61],
       [ 39,  28],
       [ 39,  65],
       [ 40,  55],
       [ 40,  47],
       [ 40,  42],
       [ 40,  42],
       [ 42,  52],
       [ 42,  60],
       [ 43,  54],
       [ 43,  60],
       [ 43,  45],
       [ 43,  41],
       [ 44,  50],
       [ 44,  46],
       [ 46,  51],
       [ 46,  46],
       [ 46,  56],
       [ 46,  55],
       [ 47,  52],
       [ 47,  59],
       [ 48,  51],
       [ 48,  59],
       [ 48,  50],
       [ 48,  48],
       [ 48,  59],
       [ 48,  47],
       [ 49,  55],
       [ 49,  42],
       [ 50,  49],
```

[50, 56],
[54, 47],
[54, 54],
[54, 53],
[54, 48],
[54, 52],
[54, 42],
[54, 51],
[54, 55],
[54, 41],
[54, 44],
[54, 57],
[54, 46],
[57, 58],
[57, 55],
[58, 60],
[58, 46],
[59, 55],
[59, 41],
[60, 49],
[60, 40],
[60, 42],
[60, 52],
[60, 47],
[60, 50],
[61, 42],
[61, 49],
[62, 41],
[62, 48],
[62, 59],
[62, 55],
[62, 56],
[62, 42],
[63, 50],
[63, 46],
[63, 43],
[63, 48],
[63, 52],
[63, 54],
[64, 42],
[64, 46],
[65, 48],
[65, 50],
[65, 43],
[65, 59],
[67, 43],
[67, 57],
[67, 56],
[67, 40],
[69, 58],
[69, 91],
[70, 29],
[70, 77],
[71, 35],
[71, 95],
[71, 11],
[71, 75],
[71, 9],
[71, 75],
[72, 34],
[72, 71],
[73, 5],
[73, 88],
[73, 7],
[73, 73],
[74, 10],
[74, 72],
[75, 5],
[75, 93],
[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, 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]], dtype=int64)
```

In [17]:

```
#Fitting K-Means to the MallCostumers
```

In [18]:

```
>>> kmeans = KMeans(n_clusters=5, random_state=0).fit(X)
>>> kmeans.labels_
```

Out[18]:

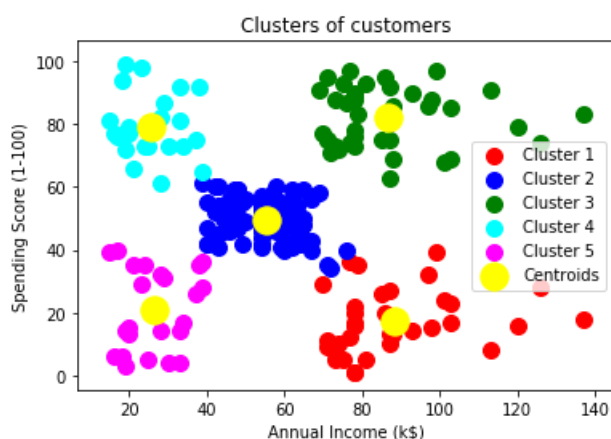
```
array([4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3,
       4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3,
       4, 3, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    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

In [19]:

```
y_kmeans = kmeans.fit_predict(y)
```

In [20]:

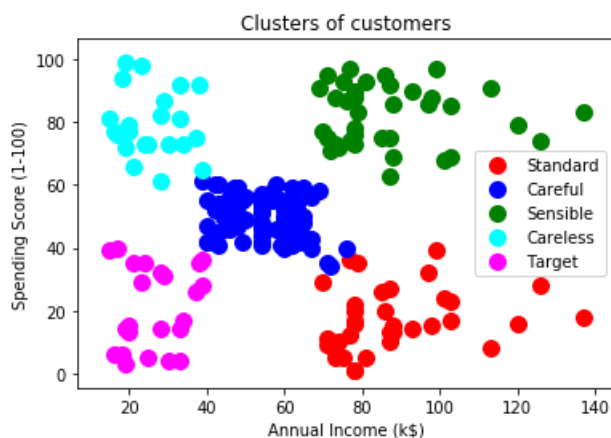
```
# Visualising the clusters
plt.scatter(X[y_kmeans == 0, 0], X[y_kmeans == 0, 1], s = 100, c = 'red', label = 'Cluster 1')
plt.scatter(X[y_kmeans == 1, 0], X[y_kmeans == 1, 1], s = 100, c = 'blue', label = 'Cluster 2')
plt.scatter(X[y_kmeans == 2, 0], X[y_kmeans == 2, 1], s = 100, c = 'green', label = 'Cluster 3')
plt.scatter(X[y_kmeans == 3, 0], X[y_kmeans == 3, 1], s = 100, c = 'cyan', label = 'Cluster 4')
plt.scatter(X[y_kmeans == 4, 0], X[y_kmeans == 4, 1], s = 100, c = 'magenta', label = 'Cluster 5')
plt.scatter(kmeans.cluster_centers[:, 0], kmeans.cluster_centers[:, 1], s = 300, c = 'yellow', label =
'Centroids')
plt.title('Clusters of customers')
plt.xlabel('Annual Income (k$)')
plt.ylabel('Spending Score (1-100)')
plt.legend()
plt.show()
```



In [21]:

```
plt.scatter(X[y_kmeans == 0, 0], X[y_kmeans == 0, 1], s = 100, c = 'red', label = 'Standard')
plt.scatter(X[y_kmeans == 1, 0], X[y_kmeans == 1, 1], s = 100, c = 'blue', label = 'Careful')
plt.scatter(X[y_kmeans == 2, 0], X[y_kmeans == 2, 1], s = 100, c = 'green', label = 'Sensible')
plt.scatter(X[y_kmeans == 3, 0], X[y_kmeans == 3, 1], s = 100, c = 'cyan', label = 'Careless')
plt.scatter(X[y_kmeans == 4, 0], X[y_kmeans == 4, 1], s = 100, c = 'magenta', label = 'Target')
#plt.scatter(kmeans.cluster_centers[:, 0], kmeans.cluster_centers[:, 1], s = 300, c = 'yellow',
label = 'Centroids')

plt.title('Clusters of customers')
plt.xlabel('Annual Income (k$)')
plt.ylabel('Spending Score (1-100)')
plt.legend()
plt.show()
```



In [22]:

```
***  
  
MallCostumers["clusters"] = kmeans.labels_
```

In [23]:

```
MallCostumers.head()
```

Out[23]:

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

In [27]:

```
MallCostumers.sample(5)
```

Out[27]:

	CustomerID	Genre	Age	Annual Income (k\$)	Spending Score (1-100)	clusters
41	42	Male	24	38	92	3
75	76	Male	26	54	54	1
174	175	Female	52	88	13	0
192	193	Male	33	113	8	0
20	21	Male	35	24	35	4

In [29]:

```
centers = pd.DataFrame(kmeans.cluster_centers_)  
centers
```

Out[29]:

	0	1
0	88.200000	17.114286
1	55.296296	49.518519
2	86.538462	82.128205
3	25.727273	79.363636
4	26.304348	20.913043

In [30]:

```
centers["clusters"] = range(5) #n_clusters
```

In [31]:

```
centers
```

Out[31]:

	0	1	clusters
0	88.200000	17.114286	0

1	55.296296	49.518519	clusters
2	86.538462	82.128205	2
3	25.727273	79.363636	3
4	26.304348	20.913043	4

In [64]:

```
MallCostumers["ind"] =MallCostumers.index
MallCostumers.head()
```

Out[64]:

CustomerID	Genre	Age	Annual Income (k\$)	Spending Score (1-100)	clusters	0	1	ind	2
------------	-------	-----	---------------------	------------------------	----------	---	---	-----	---

In [36]:

```
MallCostumers = MallCostumers.merge(centers)
MallCostumers
```

Out[36]:

CustomerID	Genre	Age	Annual Income (k\$)	Spending Score (1-100)	clusters	ind	0	1
0	1	Male	19	15	39	4	0	26.304348 20.913043
1	3	Female	20	16	6	4	2	26.304348 20.913043
2	5	Female	31	17	40	4	4	26.304348 20.913043
3	7	Female	35	18	6	4	6	26.304348 20.913043
4	9	Male	64	19	3	4	8	26.304348 20.913043
5	11	Male	67	19	14	4	10	26.304348 20.913043
6	13	Female	58	20	15	4	12	26.304348 20.913043
7	15	Male	37	20	13	4	14	26.304348 20.913043
8	17	Female	35	21	35	4	16	26.304348 20.913043
9	19	Male	52	23	29	4	18	26.304348 20.913043
10	21	Male	35	24	35	4	20	26.304348 20.913043
11	23	Female	46	25	5	4	22	26.304348 20.913043
12	25	Female	54	28	14	4	24	26.304348 20.913043
13	27	Female	45	28	32	4	26	26.304348 20.913043
14	29	Female	40	29	31	4	28	26.304348 20.913043
15	31	Male	60	30	4	4	30	26.304348 20.913043
16	33	Male	53	33	4	4	32	26.304348 20.913043
17	35	Female	49	33	14	4	34	26.304348 20.913043
18	37	Female	42	34	17	4	36	26.304348 20.913043
19	39	Female	36	37	26	4	38	26.304348 20.913043
20	41	Female	65	38	35	4	40	26.304348 20.913043
21	43	Male	48	39	36	4	42	26.304348 20.913043
22	45	Female	49	39	28	4	44	26.304348 20.913043
23	2	Male	21	15	81	3	1	25.727273 79.363636
24	4	Female	23	16	77	3	3	25.727273 79.363636
25	6	Female	22	17	76	3	5	25.727273 79.363636
26	8	Female	23	18	94	3	7	25.727273 79.363636
27	10	Female	30	19	72	3	9	25.727273 79.363636
28	12	Female	35	19	99	3	11	25.727273 79.363636
29	14	Female	24	20	77	3	13	25.727273 79.363636

...
170	CustomerID	Genre	Age	Annual Income (k\$)	Spending Score (1-100)	clusters	ind	88.200000	17.114286
171	141	Female	57	75	5	0	140	88.200000	17.114286		
172	145	Male	25	77	12	0	144	88.200000	17.114286		
173	147	Male	48	77	36	0	146	88.200000	17.114286		
174	149	Female	34	78	22	0	148	88.200000	17.114286		
175	151	Male	43	78	17	0	150	88.200000	17.114286		
176	153	Female	44	78	20	0	152	88.200000	17.114286		
177	155	Female	47	78	16	0	154	88.200000	17.114286		
178	157	Male	37	78	1	0	156	88.200000	17.114286		
179	159	Male	34	78	1	0	158	88.200000	17.114286		
180	161	Female	56	79	35	0	160	88.200000	17.114286		
181	163	Male	19	81	5	0	162	88.200000	17.114286		
182	165	Male	50	85	26	0	164	88.200000	17.114286		
183	167	Male	42	86	20	0	166	88.200000	17.114286		
184	169	Female	36	87	27	0	168	88.200000	17.114286		
185	171	Male	40	87	13	0	170	88.200000	17.114286		
186	173	Male	36	87	10	0	172	88.200000	17.114286		
187	175	Female	52	88	13	0	174	88.200000	17.114286		
188	177	Male	58	88	15	0	176	88.200000	17.114286		
189	179	Male	59	93	14	0	178	88.200000	17.114286		
190	181	Female	37	97	32	0	180	88.200000	17.114286		
191	183	Male	46	98	15	0	182	88.200000	17.114286		
192	185	Female	41	99	39	0	184	88.200000	17.114286		
193	187	Female	54	101	24	0	186	88.200000	17.114286		
194	189	Female	41	103	17	0	188	88.200000	17.114286		
195	191	Female	34	103	23	0	190	88.200000	17.114286		
196	193	Male	33	113	8	0	192	88.200000	17.114286		
197	195	Female	47	120	16	0	194	88.200000	17.114286		
198	197	Female	45	126	28	0	196	88.200000	17.114286		
199	199	Male	32	137	18	0	198	88.200000	17.114286		

200 rows × 9 columns

In [37]:

```
MallCostumers.sample(20)
```

Out[37]:

	CustomerID	Genre	Age	Annual Income (k\$)	Spending Score (1-100)	clusters	ind	0	1
178	157	Male	37	78	1	0	156	88.200000	17.114286
18	37	Female	42	34	17	4	36	26.304348	20.913043
151	174	Male	36	87	92	2	173	86.538462	82.128205
51	52	Male	33	42	60	1	51	55.296296	49.518519
36	28	Male	35	28	61	3	27	25.727273	79.363636
75	76	Male	26	54	54	1	75	55.296296	49.518519
52	53	Female	31	43	54	1	52	55.296296	49.518519
129	130	Male	38	71	75	2	129	86.538462	82.128205
14	29	Female	40	29	31	4	28	26.304348	20.913043
163	198	Male	32	126	74	2	197	86.538462	82.128205
121	122	Female	38	67	40	1	121	55.296296	49.518519

100	CustomerID	Female	23	Annual Income (k\$)	Spending Score (1-100)	clusters	1	100	55.296296	49.518519
107	108	Male	54	63	46	1	107	55.296296	49.518519	
119	120	Female	50	67	57	1	119	55.296296	49.518519	
37	30	Female	23	29	87	3	29	25.727273	79.363636	
20	41	Female	65	38	35	4	40	26.304348	20.913043	
154	180	Male	35	93	90	2	179	86.538462	82.128205	
114	115	Female	18	65	48	1	114	55.296296	49.518519	
15	31	Male	60	30	4	4	30	26.304348	20.913043	
170	139	Male	19	74	10	0	138	88.200000	17.114286	

In [38]:

```
MallCostumers = MallCostumers.sort_values("ind")
```

In [39]:

```
MallCostumers.head()
```

Out[39]:

	CustomerID	Genre	Age	Annual Income (k\$)	Spending Score (1-100)	clusters	ind	0	1
0	1	Male	19	15	39	4	0	26.304348	20.913043
23	2	Male	21	15	81	3	1	25.727273	79.363636
1	3	Female	20	16	6	4	2	26.304348	20.913043
24	4	Female	23	16	77	3	3	25.727273	79.363636
2	5	Female	31	17	40	4	4	26.304348	20.913043

In [40]:

```
MallCostumers = MallCostumers.drop("ind",1)
MallCostumers.head()
```

Out[40]:

	CustomerID	Genre	Age	Annual Income (k\$)	Spending Score (1-100)	clusters	0	1
0	1	Male	19	15	39	4	26.304348	20.913043
23	2	Male	21	15	81	3	25.727273	79.363636
1	3	Female	20	16	6	4	26.304348	20.913043
24	4	Female	23	16	77	3	25.727273	79.363636
2	5	Female	31	17	40	4	26.304348	20.913043

In [48]:

```
"s1_grps = pd.Series(kmeans.labels_)s2_univs = MallCostumers.iloc[:,0]rslt =
pd.concat([s1_grps,s2_unvs],axis=1)rslt"
```

Out[48]:

```
's1_grps = pd.Series(kmeans.labels_)s2_univs = MallCostumers.iloc[:,0]rslt =
pd.concat([s1_grps,s2_unvs],axis=1)rslt'
```

In [49]:

```
#sns.heatmap("x","y",data = MallCostumers,fit_reg=False,hue="clusters",size=7)
```

In [50]:

```
****
```

In [56]:

```
d = MallCostumers.iloc[:, [2,3,4]]
X = MallCostumers.iloc[:, [2,3,4]].values
X
```

Out[56]:

```
array([[ 19,  15,  39],
       [ 21,  15,  81],
       [ 20,  16,   6],
       [ 23,  16,  77],
       [ 31,  17,  40],
       [ 22,  17,  76],
       [ 35,  18,   6],
       [ 23,  18,  94],
       [ 64,  19,   3],
       [ 30,  19,  72],
       [ 67,  19,  14],
       [ 35,  19,  99],
       [ 58,  20,  15],
       [ 24,  20,  77],
       [ 37,  20,  13],
       [ 22,  20,  79],
       [ 35,  21,  35],
       [ 20,  21,  66],
       [ 52,  23,  29],
       [ 35,  23,  98],
       [ 35,  24,  35],
       [ 25,  24,  73],
       [ 46,  25,   5],
       [ 31,  25,  73],
       [ 54,  28,  14],
       [ 29,  28,  82],
       [ 45,  28,  32],
       [ 35,  28,  61],
       [ 40,  29,  31],
       [ 23,  29,  87],
       [ 60,  30,   4],
       [ 21,  30,  73],
       [ 53,  33,   4],
       [ 18,  33,  92],
       [ 49,  33,  14],
       [ 21,  33,  81],
       [ 42,  34,  17],
       [ 30,  34,  73],
       [ 36,  37,  26],
       [ 20,  37,  75],
       [ 65,  38,  35],
       [ 24,  38,  92],
       [ 48,  39,  36],
       [ 31,  39,  61],
       [ 49,  39,  28],
       [ 24,  39,  65],
       [ 50,  40,  55],
       [ 27,  40,  47],
       [ 29,  40,  42],
       [ 31,  40,  42],
       [ 49,  42,  52],
       [ 33,  42,  60],
       [ 31,  43,  54],
       [ 59,  43,  60],
       [ 50,  43,  45],
       [ 47,  43,  41],
       [ 51,  44,  50],
       [ 69,  44,  46],
       [ 27,  46,  51],
       [ 53,  46,  46],
       [ 70,  46,  56],
       [ 19,  46,  55],
       [ 67,  47,  52],
       [ 54,  47,  59],
       [ 63,  48,  51],
```

[18, 48, 59],
[43, 48, 50],
[68, 48, 48],
[19, 48, 59],
[32, 48, 47],
[70, 49, 55],
[47, 49, 42],
[60, 50, 49],
[60, 50, 56],
[59, 54, 47],
[26, 54, 54],
[45, 54, 53],
[40, 54, 48],
[23, 54, 52],
[49, 54, 42],
[57, 54, 51],
[38, 54, 55],
[67, 54, 41],
[46, 54, 44],
[21, 54, 57],
[48, 54, 46],
[55, 57, 58],
[22, 57, 55],
[34, 58, 60],
[50, 58, 46],
[68, 59, 55],
[18, 59, 41],
[48, 60, 49],
[40, 60, 40],
[32, 60, 42],
[24, 60, 52],
[47, 60, 47],
[27, 60, 50],
[48, 61, 42],
[20, 61, 49],
[23, 62, 41],
[49, 62, 48],
[67, 62, 59],
[26, 62, 55],
[49, 62, 56],
[21, 62, 42],
[66, 63, 50],
[54, 63, 46],
[68, 63, 43],
[66, 63, 48],
[65, 63, 52],
[19, 63, 54],
[38, 64, 42],
[19, 64, 46],
[18, 65, 48],
[19, 65, 50],
[63, 65, 43],
[49, 65, 59],
[51, 67, 43],
[50, 67, 57],
[27, 67, 56],
[38, 67, 40],
[40, 69, 58],
[39, 69, 91],
[23, 70, 29],
[31, 70, 77],
[43, 71, 35],
[40, 71, 95],
[59, 71, 11],
[38, 71, 75],
[47, 71, 9],
[39, 71, 75],
[25, 72, 34],
[31, 72, 71],
[20, 73, 5],
[29, 73, 88],
[44, 73, 7],
[32, 73, 73],
[19, 74, 10],
[35, 74, 72],
[57, 75, 5],
[32, 75, 93],

```
[ 28, 76, 40],
[ 32, 76, 87],
[ 25, 77, 12],
[ 28, 77, 97],
[ 48, 77, 36],
[ 32, 77, 74],
[ 34, 78, 22],
[ 34, 78, 90],
[ 43, 78, 17],
[ 39, 78, 88],
[ 44, 78, 20],
[ 38, 78, 76],
[ 47, 78, 16],
[ 27, 78, 89],
[ 37, 78, 1],
[ 30, 78, 78],
[ 34, 78, 1],
[ 30, 78, 73],
[ 56, 79, 35],
[ 29, 79, 83],
[ 19, 81, 5],
[ 31, 81, 93],
[ 50, 85, 26],
[ 36, 85, 75],
[ 42, 86, 20],
[ 33, 86, 95],
[ 36, 87, 27],
[ 32, 87, 63],
[ 40, 87, 13],
[ 28, 87, 75],
[ 36, 87, 10],
[ 36, 87, 92],
[ 52, 88, 13],
[ 30, 88, 86],
[ 58, 88, 15],
[ 27, 88, 69],
[ 59, 93, 14],
[ 35, 93, 90],
[ 37, 97, 32],
[ 32, 97, 86],
[ 46, 98, 15],
[ 29, 98, 88],
[ 41, 99, 39],
[ 30, 99, 97],
[ 54, 101, 24],
[ 28, 101, 68],
[ 41, 103, 17],
[ 36, 103, 85],
[ 34, 103, 23],
[ 32, 103, 69],
[ 33, 113, 8],
[ 38, 113, 91],
[ 47, 120, 16],
[ 35, 120, 79],
[ 45, 126, 28],
[ 32, 126, 74],
[ 32, 137, 18],
[ 30, 137, 83]], dtype=int64)
```

In [57]:

```
# Fitting KMeans to dataset
```

In [59]:

```
kmeans = KMeans(n_clusters = 5, init = 'k-means++', random_state = 42)
```

In [60]:

```
y_kmeans = kmeans.fit_predict(X)
```

In [61]:

```
centers = pd.DataFrame(kmeans.cluster_centers_)
centers
```

Out[61]:

	0	1	2
0	45.217391	26.304348	20.913043
1	32.692308	86.538462	82.128205
2	40.324324	87.432432	18.189189
3	43.126582	54.822785	49.835443
4	25.272727	25.727273	79.363636

In [66]:

```
centers = pd.DataFrame(kmeans.cluster_centers_)
centers["clusters"] = range(5) #n_clusters
MallCostumers["ind"] = MallCostumers.index
MallCostumers = MallCostumers.merge(centers)
MallCostumers.sample()
```

```
-----
ValueError                                Traceback (most recent call last)
<ipython-input-66-dad8e251610b> in <module>()
      3 MallCostumers["ind"] = MallCostumers.index
      4 MallCostumers = MallCostumers.merge(centers)
----> 5 MallCostumers.sample()

~\.conda\lib\site-packages\pandas\core\generic.py in sample(self, n, frac, replace, weights,
random_state, axis)
    4199             "provide positive value.")
    4200
-> 4201         locs = rs.choice(axis_length, size=n, replace=replace, p=weights)
    4202         return self.take(locs, axis=axis, is_copy=False)
    4203

mtrand.pyx in mtrand.RandomState.choice()

ValueError: a must be greater than 0
```

In [67]:

```
MallCostumers = MallCostumers.sort_values("ind")
MallCostumers = MallCostumers.drop("ind",1)
MallCostumers
```

Out[67]:

CustomerID	Genre	Age	Annual Income (k\$)	Spending Score (1-100)	0	1	2	clusters
------------	-------	-----	---------------------	------------------------	---	---	---	----------