In [1]:

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

```
C:\Users\admin\anaconda3\lib\site-packages\scipy\__init__.py:146: UserWarn
ing: A NumPy version >=1.16.5 and <1.23.0 is required for this version of
SciPy (detected version 1.23.5
  warnings.warn(f"A NumPy version >={np_minversion} and <{np_maxversion}"
```

In [5]:

```
data = pd.read_csv('Mall_Customers.csv')
data.head()
```

Out[5]:

	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

In [4]:

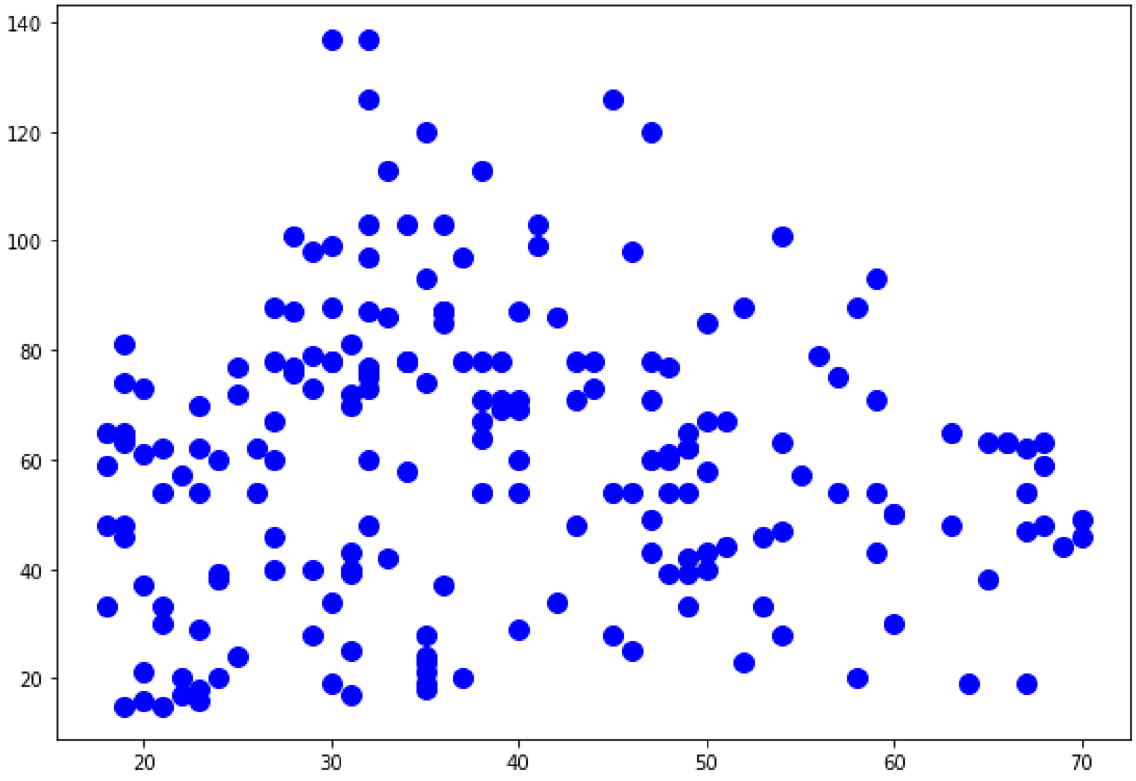
```
datasubset = data.loc[:,["Age","Annual Income (k$)"]]
```

In [6]:

```
#Scatter diagram
plt.figure(figsize = (10,7))
plt.scatter(datasubset[['Age']], datasubset[['Annual Income (k$)']], s
```

Out[6]:

```
<matplotlib.collections.PathCollection at 0x1d6704e7820>
```



In [9]:

```
import scipy.cluster.hierarchy as sch
plt.figure(figsize = (10,7))
```

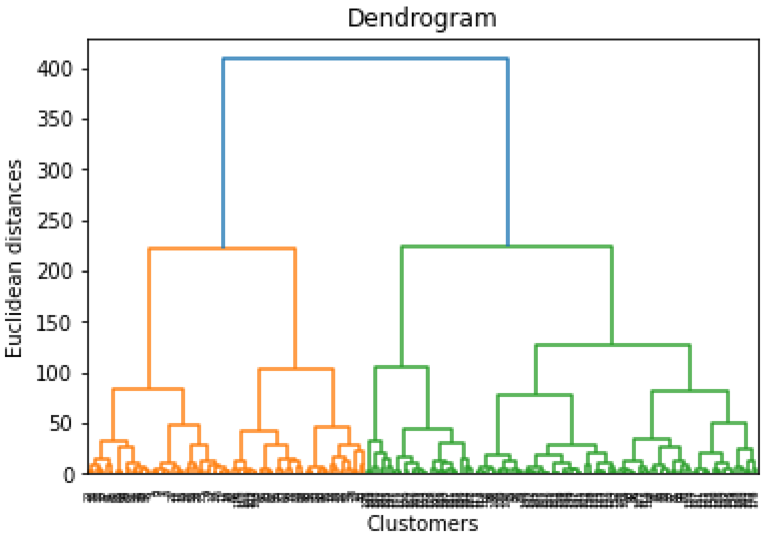
Out[9]:

```
<Figure size 720x504 with 0 Axes>

<Figure size 720x504 with 0 Axes>
```

In [11]:

```
dendogram = sch.dendrogram(sch.linkage(datasubset, method = 'ward'))
plt.title('Dendrogram')
plt.xlabel('Clustomers')
plt.ylabel('Euclidean distances')
plt.show()
```



In [12]:

```
#check for largest distance vertically without crossing any horizontal
from sklearn.cluster import AgglomerativeClustering
cluster = AgglomerativeClustering(n_clusters = 2, affinity = 'euclidean
cluster.fit_predict(datasubset)
```

Out[12]:

```
array([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,
       0, 1, 1, 1, 1, 0, 1, 1, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0,
       0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 1,
       1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0], dtype=int64)
```

In [13]:

```
cl = cluster.fit_predict(datasubset)
```

In [14]:

```
from sklearn.metrics import silhouette_score
silhouette_score(datasubset,cl)
```

Out[14]:

```
0.4104652474372429
```

In [16]:

```
from sklearn.cluster import AgglomerativeClustering
import scipy.cluster.hierarchy as sch
from sklearn.datasets import make_blobs
```

In [17]:

```
#Generate a ranodm dataset with 100 samples and 3 features
x, y = make_blobs(n_samples = 100, centers = 4, n_features = 3, random_
```

In [19]:

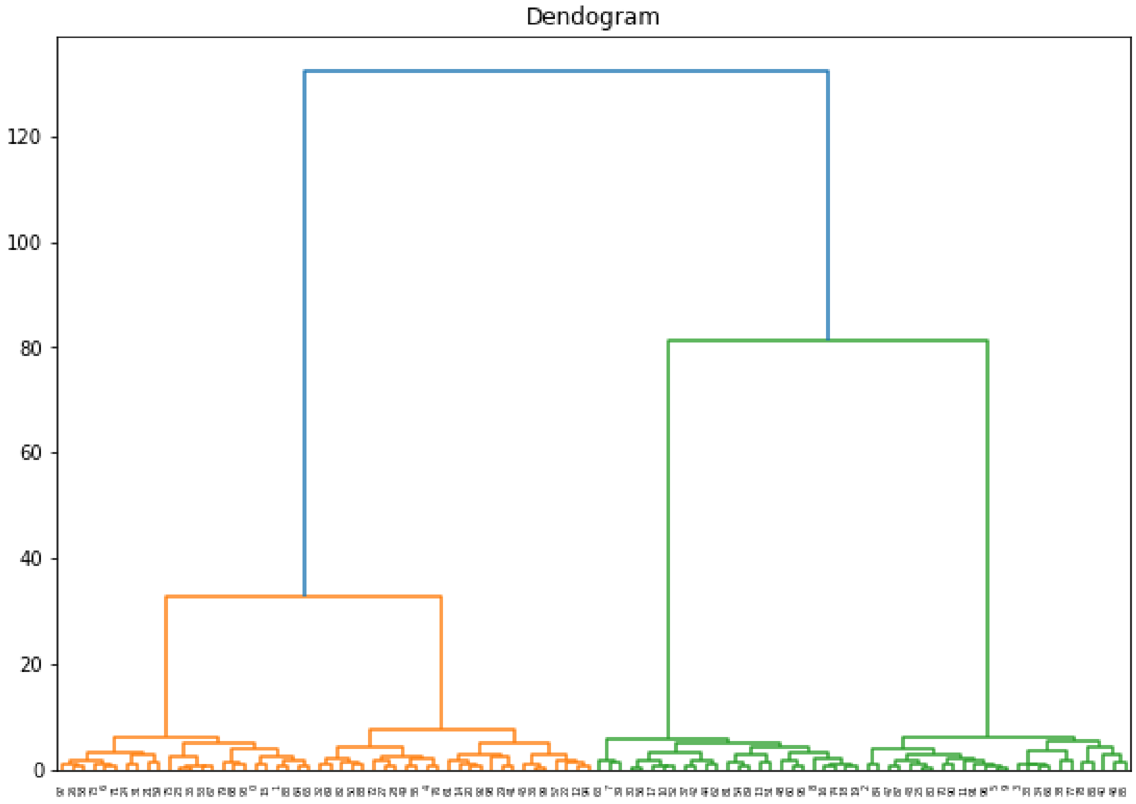
```
#Perform the dendogram
clustering = AgglomerativeClustering(n_clusters = 3, affinity = 'euclid
clustering.fit(x)
```

Out[19]:

```
AgglomerativeClustering(n_clusters=3)
```

In [21]:

```
#plot the dendogram
plt.figure(figsize = (10,7))
plt.title('Dendogram')
dendogram = sch.dendrogram(sch.linkage(x, method = 'ward'))
plt.show()
```



In [ ]:

```

```