```python
In [1]:  # Importing the libraries
         import numpy as np
         import matplotlib.pyplot as plt
         import pandas as pd
```
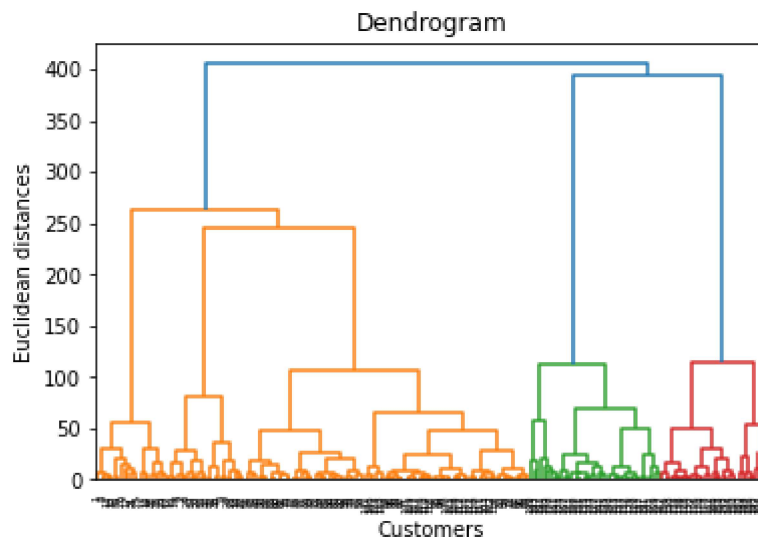
```python
In [2]:  # Importing the dataset
         dataset = pd.read_csv('Mall_Customers.csv')
         X = dataset.iloc[:, [3, 4]].values
```

Using the dendrogram to find the optimal number of clusters

```python
In [3]:  import scipy.cluster.hierarchy as sch

         dendrogram = sch.dendrogram(sch.linkage(X, method = 'ward'))

         plt.title('Dendrogram')
         plt.xlabel('Customers')
         plt.ylabel('Euclidean distances')
         plt.show()
```


Hierarchical Clustering model

```python
In [4]:  from sklearn.cluster import AgglomerativeClustering

         hc = AgglomerativeClustering(n_clusters = 5, affinity = 'euclidean', linkage = 'w

         y_hc = hc.fit_predict(X)
```

Visualising the clusters

```
In [5]:  plt.scatter(X[y_hc == 0, 0], X[y_hc == 0, 1], s = 100, c = 'red', label = 'Cluste
         plt.scatter(X[y_hc == 1, 0], X[y_hc == 1, 1], s = 100, c = 'blue', label = 'Clust
         plt.scatter(X[y_hc == 2, 0], X[y_hc == 2, 1], s = 100, c = 'green', label = 'Clus
         plt.scatter(X[y_hc == 3, 0], X[y_hc == 3, 1], s = 100, c = 'cyan', label = 'Clust
         plt.scatter(X[y_hc == 4, 0], X[y_hc == 4, 1], s = 100, c = 'magenta', label = 'Cl
         plt.title('Clusters of customers')
         plt.xlabel('Annual Income (k$)')
         plt.ylabel('Spending Score (1-100)')
         plt.legend()
         plt.show()
```