K-Means Clustering

Importing the libraries

```
In [2]:  import numpy as np
         import matplotlib.pyplot as plt
         import pandas as pd
```
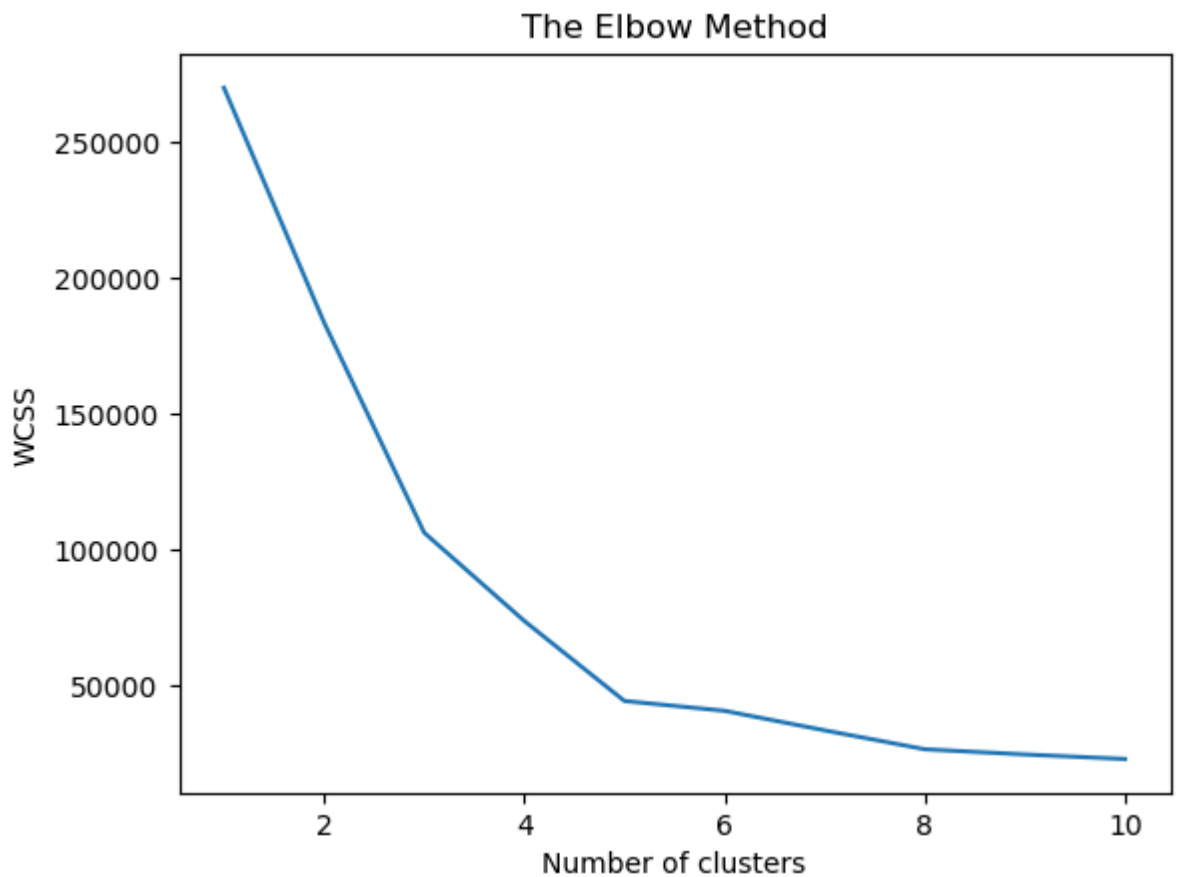
Importing the dataset

```
In [3]:  dataset = pd.read_csv(r"C:\Users\Prachi\Documents\VS Code Files\Machine Learning\CL
```

```
In [4]:  X = dataset.iloc[:, [3,4]].values
```

Using the elbow method to find the optimal number of clusters

```
In [5]:  from sklearn.cluster import KMeans
         wcss = []
         for i in range(1, 11):
             kmeans = KMeans(n_clusters = i, init = 'k-means++', random_state = 42)
             kmeans.fit(X)
             wcss.append(kmeans.inertia_)
         plt.plot(range(1, 11), wcss)
         plt.title('The Elbow Method')
         plt.xlabel('Number of clusters')
         plt.ylabel('WCSS')
         plt.show()
```

The Elbow Method

Training the K-means model on the dataset

```
In [6]: kmeans = KMeans(n_clusters = 5, init = 'k-means++', random_state = 42)
        y_kmeans = kmeans.fit_predict(X)
```

Visualising the clusters

```
In [7]: plt.scatter(X[y_kmeans == 0, 0], X[y_kmeans == 0, 1], s = 100, c = 'red', label = '
        plt.scatter(X[y_kmeans == 1, 0], X[y_kmeans == 1, 1], s = 100, c = 'blue', label =
        plt.scatter(X[y_kmeans == 2, 0], X[y_kmeans == 2, 1], s = 100, c = 'green', label =
        plt.scatter(X[y_kmeans == 3, 0], X[y_kmeans == 3, 1], s = 100, c = 'cyan', label =
        plt.scatter(X[y_kmeans == 4, 0], X[y_kmeans == 4, 1], s = 100, c = 'magenta', label
        plt.scatter(kmeans.cluster_centers_[:, 0], kmeans.cluster_centers_[:, 1], s = 300,
        plt.title('Clusters of customers')
        plt.xlabel('Annual Income (k$)')
        plt.ylabel('Spending Score (1-100)')
        plt.legend()
        plt.show()
```