By Bhagya M

Task no2-Predict the optimum number of clusters and represent it visually

In [3]:

```
#importing libraries

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn import datasets
```

In [4]:

```
#importing dataset

iris=datasets.load_iris()
iris_df=pd.DataFrame(iris.data,columns=iris.feature_names)
iris_df.head()
```

Out[4]:

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)
0	5.1	3.5	1.4	0.2
1	4.9	3.0	1.4	0.2
2	4.7	3.2	1.3	0.2
3	4.6	3.1	1.5	0.2
4	5.0	3.6	1.4	0.2

In [5]:

```
x=iris_df.iloc[:,[0,1,2,3]].values
x[0:5]
```

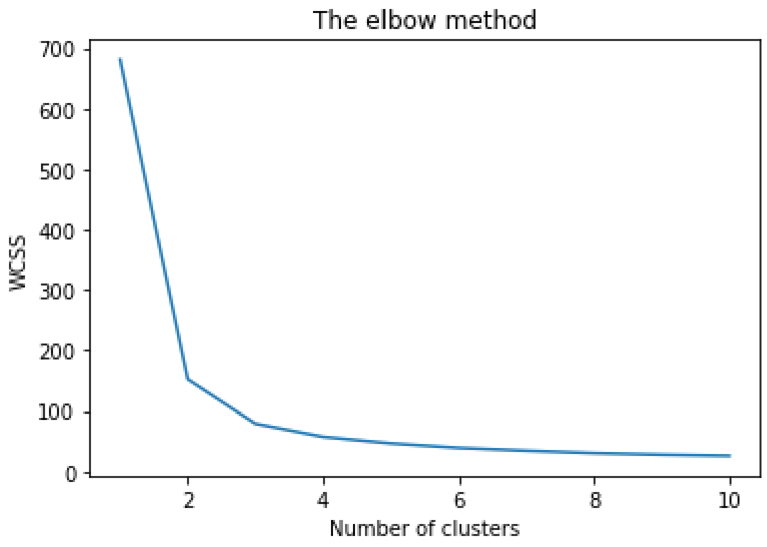
Out[5]:

```
array([[5.1, 3.5, 1.4, 0.2],
       [4.9, 3. , 1.4, 0.2],
       [4.7, 3.2, 1.3, 0.2],
       [4.6, 3.1, 1.5, 0.2],
       [5. , 3.6, 1.4, 0.2]])
```

In [7]:

```
#visualization for numbers of clusters

from sklearn.cluster import KMeans
wcss=[]
for i in range(1,11):
    kmeans=KMeans(n_clusters=i)
    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()
```



In [8]:

```
#creating model with 3 clusters

kmeans=KMeans(n_clusters=3,random_state=0)
model=kmeans.fit_predict(x)
```

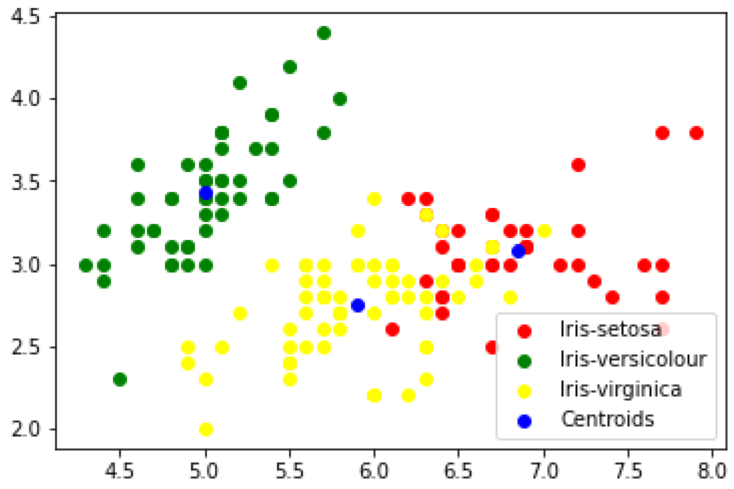
In [10]:

```
#visualization for classification and cluster centroids

plt.scatter(x[model==0,0],x[model==0,1],color='red',label='Iris-setosa')
plt.scatter(x[model==1,0],x[model==1,1],color='green',label='Iris-versicolour')
plt.scatter(x[model==2,0],x[model==2,1],color='yellow',label='Iris-virginica')
plt.scatter(kmeans.cluster_centers_[:,0],kmeans.cluster_centers_[:,1],c='blue',label='Centro
ids')
plt.legend()
```

Out[10]:

```
<matplotlib.legend.Legend at 0x3b981d3970>
```



In [ ]: