In [ ]:
```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
from sklearn import datasets
from sklearn.preprocessing import StandardScaler
```

In [ ]:
```python
iris = datasets.load_iris()
data = pd.DataFrame(data = iris.data,columns = iris.feature_names)
```

In [ ]:
```python
data.head()
```

Out[ ]:

	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 [ ]:
```python
scaler = StandardScaler()

data = scaler.fit_transform(data)
```
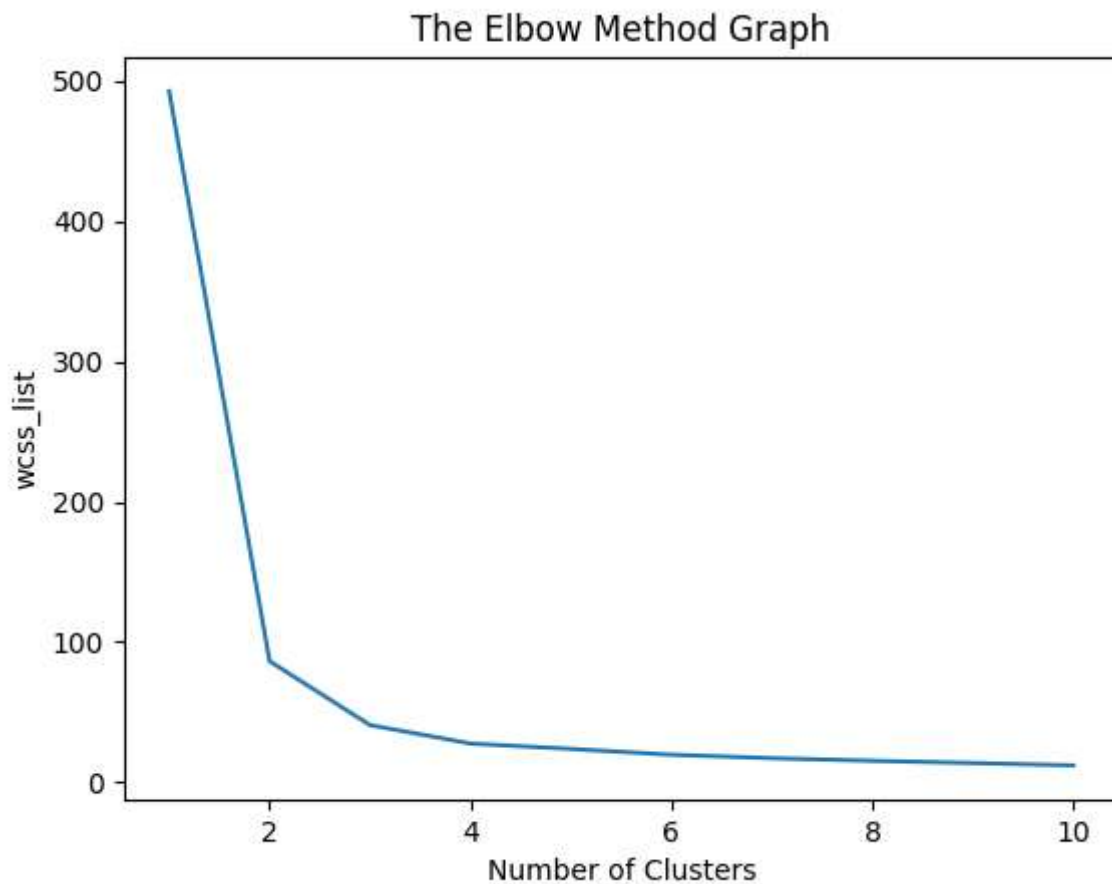
In [ ]:
```python
x = data.iloc[:,[1,2]].values
```

In [ ]:
```python
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_)
```

In [ ]:
```python
plt.plot(range(1,11),wcss)
plt.title("The Elbow Method Graph")
plt.xlabel("Number of Clusters")
plt.ylabel("wcss_list")
plt.show()
```

The Elbow Method Graph

In [ ]:
```python
kmeans = KMeans(n_clusters = 2,init = 'k-means++', random_state = 42)
y_pred = kmeans.fit_predict(x)
```

In [ ]:
```python
plt.scatter(x[y_pred == 0,0],x[y_pred == 0,1], s = 100, c = 'red',label = 'Cluster
plt.scatter(x[y_pred == 1,0],x[y_pred == 1,1], s = 100, c = 'green',label='Cluster

plt.title("Clusters of Iris")
plt.xlabel("sepal_length")
plt.ylabel("sepal_width")
plt.legend()
plt.show()
```

Clusters of Iris