Name - Saurabh Rajesh Pande

Roll no - 2447006 , batch - A

problem statement - Implement K-Means clustering on Iris.csv dataset. Determine the number of clustersusing the elbow method.

In [5]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
from sklearn.preprocessing import StandardScaler
```

In [ ]:
```
```

In [7]:
```
df = pd.read_csv('Iris.csv')
```

In [ ]:
```
```

In [8]:
```
df
```

Out[8]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa
...	...	...	...	...	...	...
145	146	6.7	3.0	5.2	2.3	Iris-virginica
146	147	6.3	2.5	5.0	1.9	Iris-virginica
147	148	6.5	3.0	5.2	2.0	Iris-virginica
148	149	6.2	3.4	5.4	2.3	Iris-virginica
149	150	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 6 columns

In [9]:
```
df.drop('Id',axis=1,inplace=True)
```

In [10]:
```
df
```

Out[10]:

	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa
...	...	...	...	...	...
145	6.7	3.0	5.2	2.3	Iris-virginica
146	6.3	2.5	5.0	1.9	Iris-virginica
147	6.5	3.0	5.2	2.0	Iris-virginica
148	6.2	3.4	5.4	2.3	Iris-virginica
149	5.9	3.0	5.1	1.8	Iris-virginica

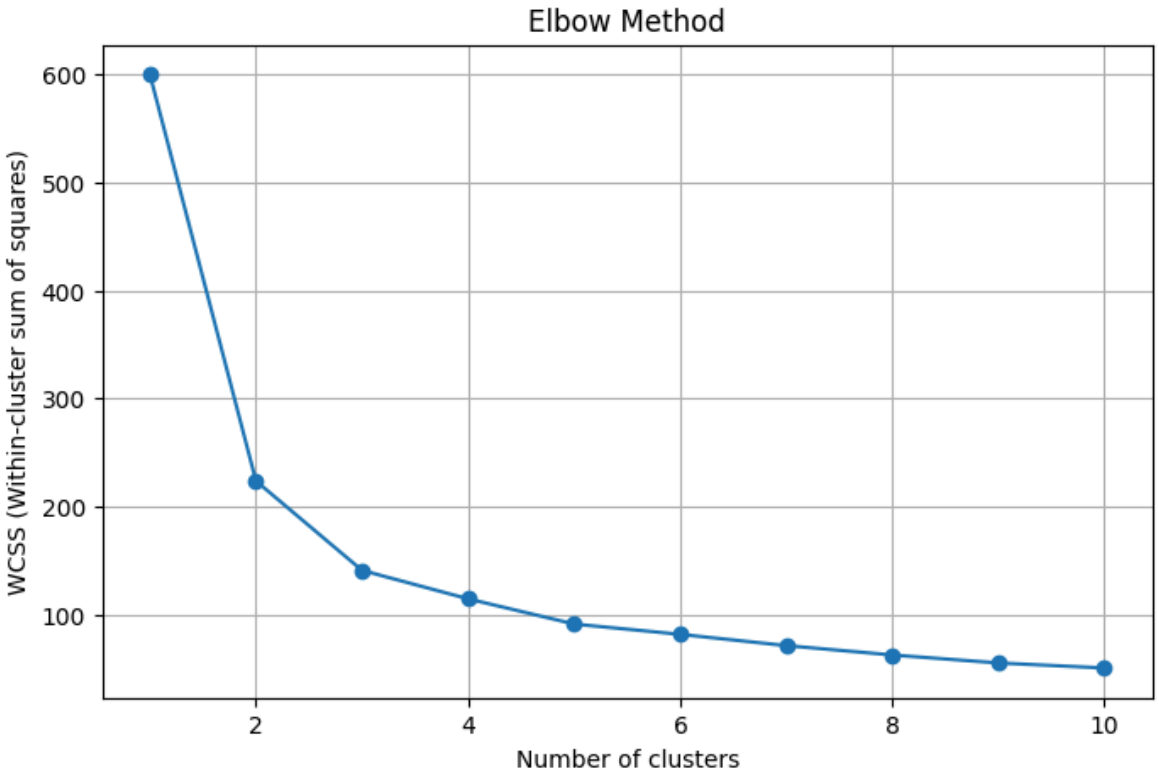
150 rows × 5 columns

In [11]:
```
X = df[['SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', 'PetalWidthCm']]
```

In [12]:
```
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

In [13]:
```
wcss = []
for i in range(1, 11):
    kmeans = KMeans(n_clusters=i, init='k-means++', max_iter=300, n_init=10, ran
    kmeans.fit(X_scaled)
    wcss.append(kmeans.inertia_)
```

In [14]:
```
plt.figure(figsize=(8,5))
plt.plot(range(1, 11), wcss, marker='o')
plt.title('Elbow Method')
plt.xlabel('Number of clusters')
plt.ylabel('WCSS (Within-cluster sum of squares)')
plt.grid(True)
plt.show()
```

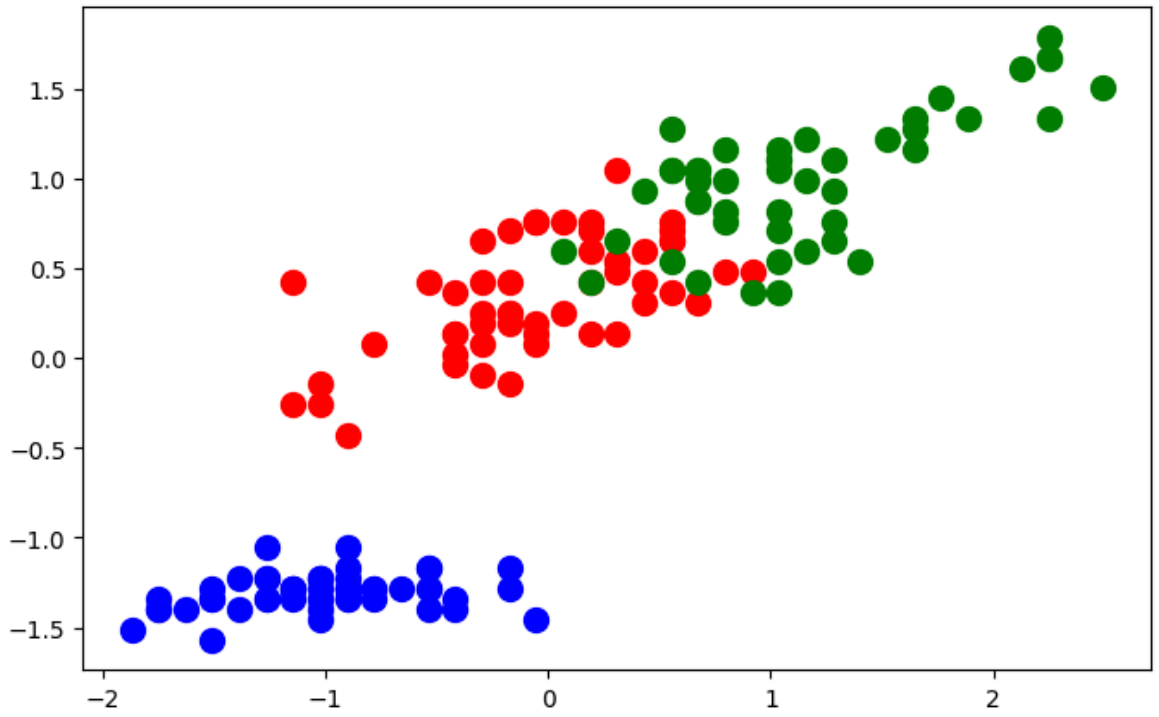


In [15]:
```
kmeans = KMeans(n_clusters=3, init='k-means++', max_iter=300, n_init=10, random_
y_kmeans = kmeans.fit_predict(X_scaled)
```

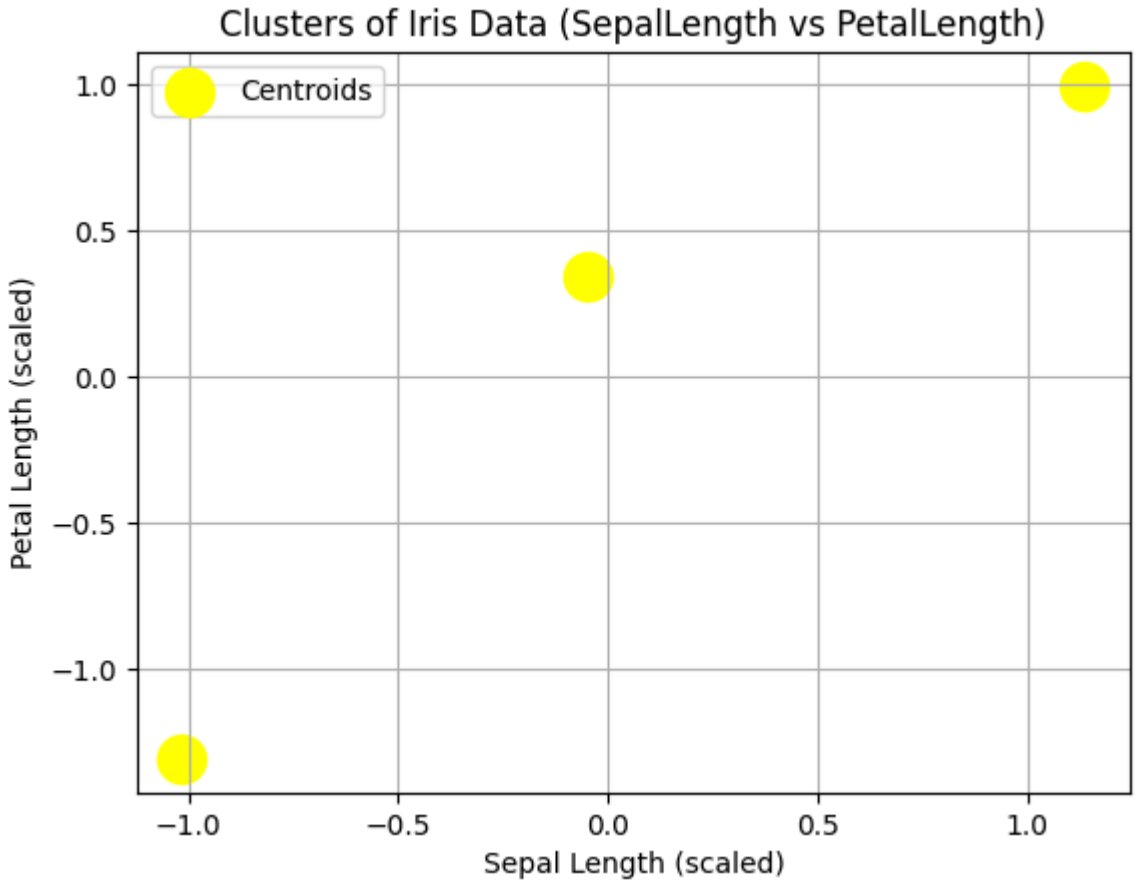
In [16]:
```
df['Cluster'] = y_kmeans
```

In [17]:
```
plt.figure(figsize=(8,5))
plt.scatter(X_scaled[y_kmeans == 0, 0], X_scaled[y_kmeans == 0, 2], s=100, c='re
plt.scatter(X_scaled[y_kmeans == 1, 0], X_scaled[y_kmeans == 1, 2], s=100, c='bl
plt.scatter(X_scaled[y_kmeans == 2, 0], X_scaled[y_kmeans == 2, 2], s=100, c='gr
```

Out[17]:
```
<matplotlib.collections.PathCollection at 0x262d35952d0>
```



In [18]:
```
plt.scatter(kmeans.cluster_centers_[:, 0], kmeans.cluster_centers_[:, 2], s=300,
plt.title('Clusters of Iris Data (SepalLength vs PetalLength)')
plt.xlabel('Sepal Length (scaled)')
plt.ylabel('Petal Length (scaled)')
plt.legend()
plt.grid(True)
plt.show()
```



In [ ]: