```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.preprocessing import StandardScaler
from sklearn.datasets import load_iris
from sklearn.cluster import KMeans
from sklearn.decomposition import PCA
df = load_iris(as_frame=True)
df = df.frame
df.head()
df.drop('target', axis =1, inplace=True)


df.info()
scaler =StandardScaler()

features =scaler.fit(df)
features =features.transform(df)


scaled_df =pd.DataFrame(features,columns=df.columns)

X=scaled_df.values
wcss = {}
for i in range(1, 11):
  kmeans = KMeans(n_clusters = i, init = 'k-means++', random_state = 42)
  kmeans.fit(X)
  wcss[i] = kmeans.inertia_

plt.plot(wcss.keys(), wcss.values(), 'gs-')
plt.xlabel("Values of 'k'")
plt.ylabel('WCSS')
plt.title('Plot 1')
plt.show()
```

```
RangeIndex: 150 entries, 0 to 149
Data columns (total 4 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   sepal length (cm)  150 non-null    float64
 1   sepal width (cm)   150 non-null    float64
 2   petal length (cm)  150 non-null    float64
 3   petal width (cm)   150 non-null    float64
dtypes: float64(4)
memory usage: 4.8 KB
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureW
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureW
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureW
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureW
```

```
kmeans=KMeans(n_clusters=3)
kmeans.fit(X)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWa
  warnings.warn(
```

```
KMeans(n_clusters=3)
```

+ Code + Text

```
kmeans.cluster_centers_
```

```
array([[-1.01457897,  0.85326268, -1.30498732, -1.25489349],
       [ 1.13597027,  0.08842168,  0.99615451,  1.01752612],
       [-0.05021989, -0.88337647,  0.34773781,  0.2815273 ]])
```

```
kmeans.labels_
```

```
array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 1, 1, 1, 2, 2, 2, 1, 2, 2, 2, 2, 2, 2, 2, 2, 1,
       2, 2, 2, 2, 1, 2, 2, 2, 2, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 1, 1, 2,
       2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 2, 1, 1, 1, 1, 2, 1, 1, 1,
       1, 1, 1, 2, 2, 1, 1, 1, 1, 2, 1, 2, 1, 2, 1, 1, 2, 1, 1, 1, 1, 1,
       1, 2, 2, 1, 1, 1, 2, 1, 1, 1, 2, 1, 1, 1, 2, 1, 1, 2], dtype=int32)
```

```
pca=PCA(n_components=3)

reduced_X=pd.DataFrame(data=pca.fit_transform(X),columns=['PCA1','PCA2','PCA3'])
```

```
reduced_X.head()
```

	PCA1	PCA2	PCA3
0	-2.264703	0.480027	-0.127706
1	-2.080961	-0.674134	-0.234609
2	-2.364229	-0.341908	0.044201
3	-2.299384	-0.597395	0.091290
4	-2.389842	0.646835	0.015738

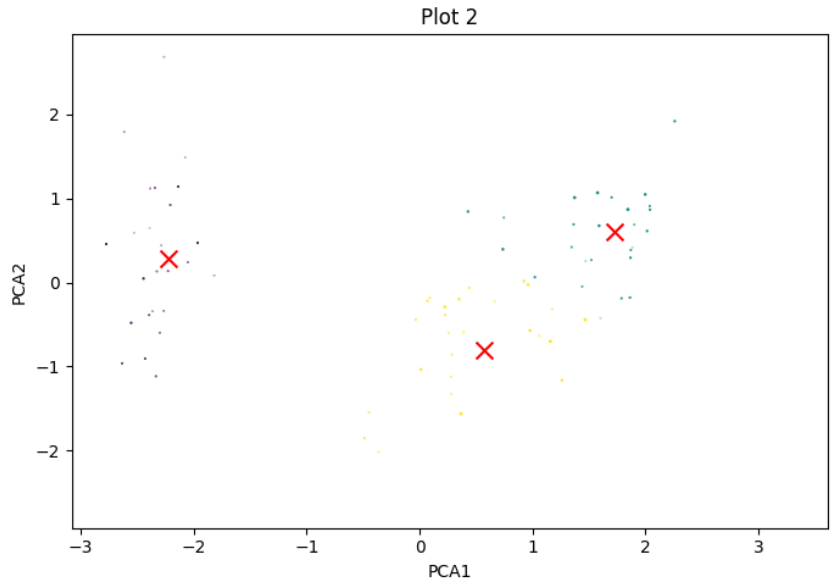
```
centers=pca.transform(kmeans.cluster_centers_)
```

```
centers
```

```
array([[-2.22475316,  0.28892745],
       [ 1.72103664,  0.60288719],
       [ 0.57262144, -0.80720888]])
```

```
plt.figure(figsize=(7,5))
```

```
plt.scatter(reduced_X['PCA1'],reduced_X['PCA2'],reduced_X['PCA3'],c=kmeans.labels_)
plt.scatter(centers[:,0],centers[:,1],marker='x',s=100,c='red')
plt.xlabel('PCA1')
plt.ylabel('PCA2')
plt.title('Plot 2')
plt.tight_layout()
```




```
pca.components_
```

```
array([[ 0.52106591, -0.26934744,  0.5804131 ,  0.56485654],
       [ 0.37741762,  0.92329566,  0.02449161,  0.06694199]])
```

```
component_df=pd.DataFrame(pca.components_,index=['PCA1',"PCA2","PCA3"],columns=df.columns)

sns.heatmap(component_df)
plt.show()
```