```
# Importing Libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from kmodes.kmodes import KModes

# Create a Dataset
X1 = np.array(["A", "C", "B", "D", "C", "D", "B", "D"])
X2 = np.array(["M", "Y", "N", "Z", "M", "Y", "N", "Z"])
X3 = np.array(["R", "W", "W"," W", "R", "W", "R", "R"])
ID = np.array(["ID-1", "ID-2", "ID-3", "ID-4", "ID-5", "ID-6", "ID-7", "ID-8"])
df = np.column_stack((ID, X1, X2, X3))
DT = pd.DataFrame(df, columns = ["ID", "X1", "X2", "X3"], index = np.arange(1, 9))
DT

# Looping over a range of cluster numbers to compute Cost
np.random.seed(201681)
cost = []
for i in range(1, 5):
   KM = KModes(n_clusters = i, init = "random")
   KM.fit(DT)
   cost.append(KM.cost_)
print(cost)

# Plotting the Elbow Method
number_clusters = range(1, 5)
plt.figure(figsize = (6, 4))
plt.plot(number_clusters, cost, color = "red", linestyle = "solid", linewidth = 2, marker = "o",
markerfacecolor = "blue", markersize = 10)
```

```
plt.axvline(x = 3, color = "blue", ls = "--", linewidth = 2)
plt.xlabel("Number of clusters")
plt.ylabel("Cost")
plt.title("Elbow Method for Optimal value of k")
plt.grid(True)
plt.show()

# Fitting KMeans with Optimal k
K = KModes(n_clusters = 3, init = "random")
K.fit(DT)
K.fit_predict(DT)

# Accessing KMeans Attributes and Methods
K.cluster_centroids_
K.labels_
K.cost_
K.epoch_costs_
DT.insert(0, "Cluster", K.labels_, True)
DT
```