# Code 1:

# pip install pandas numpy plotly scikit-learn

# import pandas as pd

# import numpy as np

# from sklearn.cluster import KMeans

# import matplotlib.pyplot as plt

# # Provided student data

# data = {

# 'Roll No': list(range(1, 59)),

# 'Unit\_Test\_I\_Grades': [12, 6, 9, 2, 14, 'ABSENT', 10, 'ABSENT', 12, 'ABSENT', 'ABSENT', 14, 8, 'ABSENT', 14, 12, 'ABSENT', 14, 'ABSENT',

# 6, 15, 8, 14, 'ABSENT', 13, 8, 14, 15, 5, 'ABSENT', 11, 'ABSENT', 'ABSENT', 6, 9, 13, 'ABSENT', 7, 'ABSENT',

# 12, 13, 9, 10, 2, 9, 7, 8, 0, 10, 'ABSENT', 8, 8, 12, 13, 16, 11, 'ABSENT', 'ABSENT']

# }

# # Create a DataFrame from the provided data

# df = pd.DataFrame(data)

# # Handle missing or non-numeric values (in this case, replacing 'ABSENT' with -1)

# df['Unit\_Test\_I\_Grades'] = pd.to\_numeric(df['Unit\_Test\_I\_Grades'], errors='coerce').fillna(-1)

# # Perform K-Means clustering

# kmeans = KMeans(n\_clusters=3, random\_state=42)

# df['Cluster'] = kmeans.fit\_predict(df['Unit\_Test\_I\_Grades'].values.reshape(-1, 1))

# # Visualize the clusters

# plt.scatter(df['Roll No'], df['Unit\_Test\_I\_Grades'], c=df['Cluster'], cmap='viridis', s=50, alpha=0.7)

# plt.xlabel('Roll No')

# plt.ylabel('Unit Test I Grades')

# plt.title('K-Means Clustering of Students')

# plt.show()

# import pandas as pd

# import numpy as np

# import matplotlib.pyplot as plt

# import skfuzzy as fuzz

# # Provided student data

# data = {

# 'Roll No': list(range(1, 59)),

# 'Unit\_Test\_I\_Grades': [12, 6, 9, 2, 14, 'ABSENT', 10, 'ABSENT', 12, 'ABSENT', 'ABSENT', 14, 8, 'ABSENT', 14, 12, 'ABSENT', 14, 'ABSENT',6, 15, 8, 14, 'ABSENT', 13, 8, 14, 15, 5, 'ABSENT', 11, 'ABSENT', 'ABSENT', 6, 9, 13, 'ABSENT', 7, 'ABSENT',12, 13, 9, 10, 2, 9, 7, 8, 0, 10, 'ABSENT', 8, 8, 12, 13, 16, 11, 'ABSENT', 'ABSENT']

# }

# # Create a DataFrame from the provided data

# df = pd.DataFrame(data)

# # Handle missing or non-numeric values (replacing 'ABSENT' with -1)

# df['Unit\_Test\_I\_Grades'] = pd.to\_numeric(df['Unit\_Test\_I\_Grades'], errors='coerce').fillna(-1)

# # Prepare the data for clustering

# grades = df['Unit\_Test\_I\_Grades'].values.reshape(-1, 1)

# # Perform Fuzzy C-Means clustering

# cntr, u, u0, d, jm, p, fpc = fuzz.cluster.cmeans(

# grades.T, 3, 2, error=0.005, maxiter=1000, init=None, seed=42)

# # Get the cluster membership for each student

# df['Cluster'] = np.argmax(u, axis=0)

# # Visualize the clusters

# plt.scatter(df['Roll No'], df['Unit\_Test\_I\_Grades'], c=df['Cluster'], cmap='viridis', s=50, alpha=0.7)

# plt.xlabel('Roll No')

# plt.ylabel('Unit Test I Grades')

# plt.title('Fuzzy C-Means Clustering of Students')

# plt.show()