Assignment-2 (Data Preprocessing & Similarity)

```
import pandas as pd
import numpy as np
from sklearn.preprocessing import StandardScaler, MinMaxScaler, KBinsDiscretizer, OneHotEncoder

# Load dataset (from Kaggle link, assumed downloaded locally)
# df = pd.read_csv('AdventureWorks.csv')

# Part I - Feature Selection (example)
#   Selected   features:   ['Age',   'Gender',   'YearlyIncome',   'CommuteDistance',   'Occupation',
'MaritalStatus', 'BikeBuyer']
#                                          df_selected                                                =
df[['Age','Gender','YearlyIncome','CommuteDistance','Occupation','MaritalStatus','BikeBuyer']]

# Data types identified manually (Nominal, Ordinal, Continuous)

# Part II - Preprocessing
# Handling Nulls
# df_selected = df_selected.dropna()

# Normalization
# scaler = MinMaxScaler()
# df_selected[['Age','YearlyIncome']] = scaler.fit_transform(df_selected[['Age','YearlyIncome']])

# Discretization
# kb = KBinsDiscretizer(n_bins=4, encode='ordinal', strategy='uniform')
# df_selected['Income_bin'] = kb.fit_transform(df_selected[['YearlyIncome']])

# One-Hot Encoding
# df_selected = pd.get_dummies(df_selected, columns=['Gender','Occupation','MaritalStatus'])

# Standardization
# std_scaler = StandardScaler()
#                                df_selected[['Age','YearlyIncome']]                                    =
std_scaler.fit_transform(df_selected[['Age','YearlyIncome']])

# Part III - Similarity & Correlation
# Example similarity
# obj1 = df_selected.iloc[0]
# obj2 = df_selected.iloc[1]

# Simple Matching, Jaccard, Cosine
# from sklearn.metrics.pairwise import cosine_similarity
# cosine = cosine_similarity([obj1],[obj2])

# Correlation between Commute Distance and Yearly Income
# corr = df_selected['CommuteDistance'].corr(df_selected['YearlyIncome'])

# ------------------------------------------------
# Name: Aaditya Khanna
```

```
# Roll No: 102483002
# Sub Group: 3C53
# ------------------------------------------------
```