 # Assignment 2- Preprocess and clean datasets for Generative AI applications using Python libraries such as Pandas and NumPy. Handle missing data, normalize features, and encode categorical variables.

```
# Import necessary libraries
import pandas as pd                #For data manipulation and analysis.
import numpy as np                 #For numerical operations.
from sklearn.preprocessing import MinMaxScaler, StandardScaler,
LabelEncoder                       # To scale and encode data.

# Q1: Create sample dataset
data = {
    'Name': ['Alice', 'Bob', 'Charlie', 'David', np.nan],
    'Age': [25, 30, np.nan, 40, 35],
    'Salary': [50000, 60000, 55000, np.nan, 70000],
    'Department': ['HR', 'IT', 'HR', 'Finance', 'IT']
}                              # np.nan stands for "Not a Number".

df = pd.DataFrame(data)
print("Original Dataset:\n", df)

# Q2: Handling missing values
## a) Drop rows with any missing value
df_dropna = df.dropna()        #dropna() removes rows containing any
                                    missing(NaN) value. Useful when dataset
                                    is large and removing a few rows won't
                                    harm the model.
print("\nDataset after dropping missing values:\n", df_dropna)

## b) Fill missing values
df_filled = df.copy()
df_filled['Age'] = df_filled['Age'].fillna(df_filled['Age'].mean())
df_filled['Salary'] =
df_filled['Salary'].fillna(df_filled['Salary'].mean())
df_filled['Name'] = df_filled['Name'].fillna(df_filled['Name'].mode()[0])
print("\nDataset after filling missing values:\n", df_filled)

# Q3: Normalize numeric features
## Min-Max Scaling
scaler_minmax = MinMaxScaler()
df_minmax = df_filled.copy()
df_minmax[['Age', 'Salary']] =
scaler_minmax.fit_transform(df_minmax[['Age', 'Salary']])
print("\nMin-Max Scaled Data:\n", df_minmax)


# Q4: Encoding Categorical Variables
## a) Label Encoding
df_label = df_filled.copy()
le = LabelEncoder()
df_label['Department_Label'] = le.fit_transform(df_label['Department'])
```

```
print("\nLabel Encoded Data:\n", df_label)

## b) One-Hot Encoding
df_onehot = pd.get_dummies(df_filled, columns=['Department'])
print("\nOne-Hot Encoded Data:\n", df_onehot)
```

Output:-

```
Original Dataset:
       Name   Age   Salary Department
0    Alice  25.0  50000.0         HR
1      Bob  30.0  60000.0         IT
2  Charlie   NaN  55000.0         HR
3    David  40.0      NaN    Finance
4      NaN  35.0  70000.0         IT

Dataset after dropping missing values:
     Name   Age   Salary Department
0  Alice  25.0  50000.0         HR
1    Bob  30.0  60000.0         IT

Dataset after filling missing values:
       Name   Age   Salary Department
0    Alice  25.0  50000.0         HR
1      Bob  30.0  60000.0         IT
2  Charlie  32.5  55000.0         HR
3    David  40.0  58750.0    Finance
4    Alice  35.0  70000.0         IT

Min-Max Scaled Data:
       Name       Age  Salary Department
0    Alice  0.000000  0.0000         HR
1      Bob  0.333333  0.5000         IT
2  Charlie  0.500000  0.2500         HR
3    David  1.000000  0.4375    Finance
4    Alice  0.666667  1.0000         IT


Label Encoded Data:
       Name   Age   Salary Department  Department_Label
0    Alice  25.0  50000.0         HR                 1
1      Bob  30.0  60000.0         IT                 2
2  Charlie  32.5  55000.0         HR                 1
3    David  40.0  58750.0    Finance                 0
4    Alice  35.0  70000.0         IT                 2

One-Hot Encoded Data:
       Name   Age   Salary  Department_Finance  Department_HR
Department_IT
0    Alice  25.0  50000.0               False           True
False
1      Bob  30.0  60000.0               False          False
True
2  Charlie  32.5  55000.0               False           True
False
```

```
3    David  40.0  58750.0               True          False
False
4    Alice  35.0  70000.0              False          False
True
```