

### 3.Handling missing values

```
import pandas as pd
from sklearn.impute import KNNImputer
df = pd.read_csv("bank.csv")
print("Initial Missing Values:")
print(df.isnull().sum())
df_drop_columns = df.dropna(axis=1)
print("\nAfter Dropping Columns with ANY Missing Value:")
print(df_drop_columns.isnull().sum())

df_impute = df.copy() # Work on a separate copy

# Mean Imputation for Numeric Columns
numeric_cols = df_impute.select_dtypes(include=['int64', 'float64']).columns
for col in numeric_cols:
    df_impute[col] = df_impute[col].fillna(df_impute[col].mean())

# Median Imputation (example: age)
if "age" in df_impute.columns:
    df_impute["age"] = df_impute["age"].fillna(df_impute["age"].median())

# Mode Imputation for Categorical Columns
categorical_cols = df_impute.select_dtypes(include=['object']).columns
for col in categorical_cols:
    df_impute[col] = df_impute[col].fillna(df_impute[col].mode()[0])

print("\nAfter Simple Imputation (Mean/Median/Mode):")
print(df_impute.isnull().sum())
# Convert categorical → numeric using One-Hot Encoding
df_encoded = pd.get_dummies(df)
```

```
# Apply KNN Imputation
imputer = KNNImputer(n_neighbors=5)

df_knn = pd.DataFrame(
    imputer.fit_transform(df_encoded),
    columns=df_encoded.columns
)

print("\nAfter KNN Imputation:")
print(df_knn.isnull().sum())

df_drop_columns.to_excel("1_Dropped_Columns.xlsx", index=False)
df_impute.to_excel("2_Basic_Imputation.xlsx", index=False)
df_knn.to_excel("3_KNN_Imputed.xlsx", index=False)

print("\nALL THREE OUTPUT FILES SAVED:")
print(" - 1_Dropped_Columns.xlsx")
print(" - 2_Basic_Imputation.xlsx")
print(" - 3_KNN_Imputed.xlsx")
```