PROGRAMMING ASSESSMENT

1. Using pandas, write a Python function to clean and preprocess a given DataFrame, which involves handling missing values, normalizing numerical columns, and encoding categorical columns.

import pandas as pd

from sklearn.preprocessing import StandardScaler, OneHotEncoder

from sklearn.compose import ColumnTransformer

from sklearn.pipeline import Pipeline

def clean\_and\_preprocess(df: pd.DataFrame) -> pd.DataFrame:

# Handle missing values

df = df.fillna(df.median(numeric\_only=True)).fillna('Unknown')

# Separate numerical and categorical columns

numerical\_cols = df.select\_dtypes(include=['int64', 'float64']).columns

categorical\_cols = df.select\_dtypes(include=['object', 'category']).columns

# Preprocessing for numerical data: scaling

numerical\_transformer = StandardScaler()

# Preprocessing for categorical data: one-hot encoding

categorical\_transformer = OneHotEncoder(handle\_unknown='ignore')

# Bundle preprocessing for numerical and categorical data

preprocessor = ColumnTransformer(

transformers=[

('num', numerical\_transformer, numerical\_cols),

('cat', categorical\_transformer, categorical\_cols)

])

# Apply the transformations to the DataFrame

df\_transformed = preprocessor.fit\_transform(df)

# Create DataFrame with transformed data

df\_transformed = pd.DataFrame(df\_transformed, columns=[

\*numerical\_cols,

\*preprocessor.named\_transformers\_['cat'].get\_feature\_names\_out(categorical\_cols)

])

return df\_transformed

# Example usage

data = {

'age': [25, 30, None, 40],

'salary': [50000, 54000, 52000, None],

'department': ['HR', 'Engineering', 'Finance', None]

}

df = pd.DataFrame(data)

cleaned\_df = clean\_and\_preprocess(df)

print(cleaned\_df)

