# TASK 1 DATA PIPELINE DEVLOPMENT

### CREATE A PIPELINE FOR DATA PREPROCESSING, TRANSFORMATION, AND LOADING USING TOOLS LIKE PANDAS AND SCIKIT-LEARN

## ETL Pipeline using Pandas and Scikit-learn

```
# Import Libraries
import pandas as pd
import numpy as np
from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.impute import SimpleImputer
import os


# Step 1: Extract (Load the data)
def extract_data(file_path):
    df = pd.read_csv(file_path)
    print("Data Loaded Successfully")
    print("Shape of data:", df.shape)
    return df


# Step 2: Transform (Data Cleaning and Preprocessing)
def transform_data(df):
    # Handling Missing Values
    print("Handling Missing Values...")
    imputer = SimpleImputer(strategy='mean')
    numeric_cols = df.select_dtypes(include=['int64', 'float64']).columns
    df[numeric_cols] = imputer.fit_transform(df[numeric_cols])

    # Encoding Categorical Variables
    print("Encoding Categorical Variables...")
    categorical_cols = df.select_dtypes(include=['object']).columns
    le = LabelEncoder()
    for col in categorical_cols:
        df[col] = le.fit_transform(df[col].astype(str))

    # Feature Scaling
    print("Scaling Numerical Features...")
    scaler = StandardScaler()
    df[numeric_cols] = scaler.fit_transform(df[numeric_cols])

    return df


# Step 3: Load (Save the transformed data)
def load_data(df, output_path):
    df.to_csv(output_path, index=False)
    print(f"Transformed data saved to {output_path}")


# Main Pipeline
def main():
    input_file = '/content/Titanic-Dataset.csv'
    output_file = 'transformed_data.csv'

    # ETL Process
    df = extract_data(input_file)
    df_transformed = transform_data(df)
    load_data(df_transformed, output_file)

if __name__ == "__main__":
    main()
```

```
Data Loaded Successfully
Shape of data: (891, 12)
Handling Missing Values...
Encoding Categorical Variables...
Scaling Numerical Features...
Transformed data saved to transformed_data.csv
```