***TASK#06(ETL)***

 **Introduction:**

You'll be given a csv dataset file and you'll be extracting data from it through pandas or pyspark. Then there will be the transformation phase and after performing that you'll try to load the final data to a database.

There are some problems with the data and you need to remove them during the transformation phase to make data useful for everyone:

- An order id should always exist as an integer

- A product id cannot be 0

- We never had a product priced more than 1500 Rs. so any item with amount greater than 1500 Rs is an anomaly and it should be treated as 1500 Rs

- A status of an item can never be null or None, if it is then its an anomaly and item rows to be considered as fake orders and should not be kept in final data

- There must be duplication in final data Directions to solve the problem:

- You'll be learning how to CREATE a table in database with pre-defined schema

- Will be figuring out how in the best way you can remove anomalies from the data - How to read the data with pandas or pyspark I'm open for discussions and help during the process.

 **Code Explanation:**

import pandas as pd  
from sqlalchemy import create\_engine  
  
# Read the CSV file  
dataframe = pd.read\_csv("C:\\Users\\MOON KHAN\\Downloads\\Documents\\dataset.csv")  
  
# Transformation phase  
# order\_id is an integer and has no missing values  
dataframe['order\_id'] = dataframe['order\_id'].astype('Int64').dropna()  
  
# Remove rows where product\_id is 0  
dataframe = dataframe[dataframe['product\_id'] != 0]  
  
# Cap the amount at 1500 Rs  
dataframe['amount'] = dataframe['amount'].apply(lambda x: min(x, 1500))  
  
# Remove rows where status is null  
dataframe = dataframe[dataframe['status'].notna()]  
  
# Remove duplicates  
dataframe = dataframe.drop\_duplicates()  
  
# Saving the cleaned data to a new CSV file  
dataframe.to\_csv('C:\\Users\\MOON KHAN\\Downloads\\Documents\\cleaned\_dataset.csv', index=False)  
  
print(dataframe)  
  
# Loading the data into PostgreSQL  
db\_connection\_string = 'postgresql://postgres:SpeakO13.@localhost:5432/Taskno4'  
  
# Create an SQLAlchemy engine  
engine = create\_engine(db\_connection\_string)  
  
# Load the DataFrame into the PostgreSQL table  
dataframe.to\_sql('cleaned\_data', con=engine, if\_exists='replace', index=False)  
  
print("ETL process is completed successfully and the data is loaded into PostgreSQL!")



