# Lab 10 > Part A

Name: Debatreya Das Roll No. 12212070 CS A4 Data Mining Lab 10

### Create A data warehouse from different .csv files using PostgreSQL tool

## Dataset

I have gathered 3 CSV files.

* [products.csv](./Data/products.csv)
* [sales.csv](./Data/sales.csv)
* [customers.csv](./Data/customer.csv)

Now I will use Python to Put these Tables in Postgres and then Show Applying a Query

## Import Libraries and Connect to PostgreSQL

import psycopg2  
import pandas as pd

# Connect with local Postgres  
conn = psycopg2.connect(  
 dbname="warehouse",   
 user="postgres",  
 password="postgres",   
 host="localhost"  
)  
cursor = conn.cursor()

### Create Tables

# Drop tables if they exist  
cursor.execute("DROP TABLE IF EXISTS sales;")  
cursor.execute("DROP TABLE IF EXISTS customers;")  
cursor.execute("DROP TABLE IF EXISTS products;")  
  
# Create 'products' table  
cursor.execute("""  
CREATE TABLE products (  
 product\_id SERIAL PRIMARY KEY,  
 product\_name VARCHAR(255),  
 category VARCHAR(255),  
 price NUMERIC  
);  
""")  
  
# Create 'customers' table  
cursor.execute("""  
CREATE TABLE customers (  
 customer\_id SERIAL PRIMARY KEY,  
 first\_name VARCHAR(255),  
 last\_name VARCHAR(255),  
 city VARCHAR(255)  
);  
""")  
  
# Create 'sales' table  
cursor.execute("""  
CREATE TABLE sales (  
 sale\_id SERIAL PRIMARY KEY,  
 product\_id INT REFERENCES products(product\_id),  
 customer\_id INT REFERENCES customers(customer\_id),  
 sale\_date DATE,  
 quantity INT  
);  
""")  
  
conn.commit()

### Load Data from CSV into Respective Files

# Load each .csv file into the respective table  
with open('./Data/products.csv', 'r') as f:  
 next(f) # Skip the header row  
 cursor.copy\_from(f, 'products', sep=',')  
  
with open('./Data/customer.csv', 'r') as f:  
 next(f)  
 cursor.copy\_from(f, 'customers', sep=',')  
  
with open('./Data/sales.csv', 'r') as f:  
 next(f)  
 cursor.copy\_from(f, 'sales', sep=',')  
  
  
conn.commit()

Now the Tables are ready.

## Quering the Data Warehouse

# Example query to retrieve sales with customer details  
query = """  
SELECT s.sale\_id, s.sale\_date, p.product\_name, c.first\_name, c.last\_name, s.quantity, p.price  
FROM sales s  
JOIN products p ON s.product\_id = p.product\_id  
JOIN customers c ON s.customer\_id = c.customer\_id;  
"""  
  
# Run the query and display the results  
df = pd.read\_sql(query, conn)  
df.head()

C:\Users\debat\AppData\Local\Temp\ipykernel\_24728\2773805579.py:10: UserWarning: pandas only supports SQLAlchemy connectable (engine/connection) or database string URI or sqlite3 DBAPI2 connection. Other DBAPI2 objects are not tested. Please consider using SQLAlchemy.  
 df = pd.read\_sql(query, conn)

sale\_id sale\_date product\_name first\_name last\_name quantity price  
0 1 2024-01-01 Smartphone John Doe 2 500.0  
1 2 2024-01-03 Blender Jane Smith 1 80.0  
2 3 2024-01-05 Laptop Mike Brown 1 1200.0  
3 4 2024-01-08 Watch Sara Johnson 3 150.0  
4 5 2024-01-09 Shoes John Doe 1 60.0

### Close connection

# Close cursor and connection  
cursor.close()  
conn.close()