E-Commerce-Data-Analysis-Project

The Brazilian E-Commerce Public Dataset by Olist is a real-world dataset from the Olist marketplace in Brazil. It records over 100,000 orders placed between 2016 and 2018, including detailed information about customers, sellers, products, payments, deliveries, and reviews.

The dataset consists of multiple interconnected CSV files - such as orders, customers, sellers, order_items, payments, products, and reviews - enabling end-to-end analysis of the order journey: from purchase and payment to delivery and customer feedback.

Dataset :

The dataset used for this project is Olist Brazilian E-Commerce Public Dataset.

Due to size restrictions, the raw CSV files are not included in this repository.
You can download the dataset from the following Google Drive link:

👉 **Google Drive Dataset:**
https://drive.google.com/drive/folders/1KHfSVSZpr5Flyqs3hiq18OID4cTQtS8E?usp=drive_link

Please place the downloaded CSV files into the `data/` folder before running the ingestion script.

Project Overview

This project builds a complete **data engineering and analytics pipeline** using the **Brazilian E-Commerce Public Dataset by Olist**.

It includes:
- A **normalized PostgreSQL OLTP database** (3NF)
- A **Python ingestion pipeline** using Pandas + SQLAlchemy
- An **analytics dashboard built with Streamlit**
- Fully automated deployment using **Docker + Docker Compose**

🟢 *The entire system runs with one command:*
docker-compose up --build

Phase 2 - Advanced Querying & The Analytical Layer (OLAP)
1. Phase2_advanced_queries.sql # All advanced analytical queries
2. performance_tuning_report.md # Before/after EXPLAIN ANALYZE results
3. star_schema_report.md # Star schema explanation & grain definitions
4. Phase2_star_schema.drawio.png # Star schema ERD image
5. dbt/ # dbt data warehouse project

Project Structure

Project/
1. dashboard/
   1. data/
      - README.md (Google drive link)
   2. pages/
      - 02_Product_Details.py
   3. app.py
   4. Dockerfile
   5. ingest_data.py
3. database/
   - schema.sql
   - security.sql
5. docker-compose.yml

How to Run the Entire Project (Docker)

1. Clone the Repository
   ```git clone <your-repo-url>```
   ```cd project```

2. Copy the data(all csv files from Google drive link to below directory -
   ```/project/dashboard/data/```

3. Ensure Docker is installed
   - Start Docker Desktop
   - Ensure Docker is running on your system.

5. Run Everything with One Command
   ```docker-compose up --build```

6. Open the Dashboard
   http://localhost:8501

This will:
1. Start PostgreSQL
2. Execute schema.sql and security.sql
3. Build dashboard container
4. Run ingest_data.py automatically
5. Launch Streamlit dashboard