# SQL Based- E-Commerce Sales Analytics Warehouse

**📌Project Overview**

This project demonstrates a complete data pipeline and analytics workflow using PostgreSQL.  
It follows the Medallion Architecture (Bronze → Silver → Gold) to transform raw data into clean, analytics-ready data, and finally into insights for dashboards.

The goal is to showcase data cleaning, modeling, and business intelligence skills with SQL.

**⚡ Tech Stack**

* **Database:** PostgreSQL (pgAdmin for queries)
* **Architecture:** Medallion (Bronze, Silver, Gold)
* **Languages:** SQL
* **Visualization:** Power BI

**🏗️ Architecture**

1. **Bronze Layer (Raw Data)**
   * Stores raw unprocessed data (customers, products, orders, order\_items, shipments).
   * Minimal cleaning (loading as-is).
2. **Silver Layer (Cleaned & Standardized Data)**
   * Standardizes formats (dates, text casing, null handling).
   * Validates fields (e.g., numeric prices, positive quantities).
   * Removes invalid or missing records.
3. **Gold Layer (Analytics & Facts/Dimensions)**
   * Star schema with fact tables (orders, order\_items, shipments) and dimension tables (customers, products, date).
   * Designed for business intelligence and reporting.

**📂 Data Model**

**🟦 Dimension Tables**

* dim\_customers → Customer info (name, country, signup date)
* dim\_products → Product catalog (product, category, price)
* dim\_date → Calendar table (date, year, month, day)

**🟨 Fact Tables**

* fact\_orders → Orders with total amounts
* fact\_order\_items → Order details with product quantities & revenue
* fact\_shipments → Delivery times & shipment status

**📊 ERD**

**A computer screen shot of a computer

AI-generated content may be incorrect.**

**📊 Analytics Queries**

**1. Sales Trends**

SELECT DATE\_TRUNC('month', order\_date)::DATE AS month,

SUM(total\_amount) AS monthly\_sales

FROM gold.fact\_orders

GROUP BY month

ORDER BY month;

➡️ **Insight:** Monthly sales performance over time

**2. Top Products by Revenue**

SELECT p.product\_name,

SUM(oi.total\_price) AS revenue

FROM gold.fact\_order\_items oi

JOIN gold.dim\_products p ON oi.product\_id = p.product\_id

GROUP BY p.product\_name

ORDER BY revenue DESC

LIMIT 10;

➡️ **Insight:** Best-performing products

**3. Customer Lifetime Value (CLV)**

SELECT c.name,

SUM(o.total\_amount) AS lifetime\_value

FROM gold.fact\_orders o

JOIN gold.dim\_customers c ON o.customer\_id = c.customer\_id

GROUP BY c.name

ORDER BY lifetime\_value DESC;

➡️ **Insight:** Top customers by total spend

**4. Delivery Performance**

SELECT AVG(delivery\_days) AS avg\_delivery\_time,

SUM(CASE WHEN delivery\_days > 7 THEN 1 ELSE 0 END)::FLOAT / COUNT(\*) \* 100 AS pct\_delayed

FROM gold.fact\_shipments;

➡️ **Insight:** Average delivery speed & % delayed shipments

**5. Repeat Customers %**

SELECT COUNT(DISTINCT customer\_id) FILTER (WHERE total\_orders > 1)::FLOAT /

COUNT(DISTINCT customer\_id) \* 100 AS repeat\_customer\_pct

FROM (

SELECT customer\_id, COUNT(order\_id) AS total\_orders

FROM gold.fact\_orders

GROUP BY customer\_id

) AS customer\_orders;

➡️ **Insight:** Share of repeat customers

**📈 Dashboard Ideas**

* **Sales Overview** → Monthly revenue trend
* **Top Customers** → CLV leaderboard
* **Top Products** → Revenue by product
* **Client Retention** → Avg % of client who have repeat purchase
* **Delivery Insights** → Avg delivery time, % delayed shipments

**🚀 How to Run**

1. Clone this repo
2. Load raw data into **Bronze schema**
3. Run transformation SQL scripts for **Silver & Gold layers**
4. Run analytics queries (provided in /queries/)
5. Build dashboard in Power BI, Tableau, or Python

**🎯 Key Learnings**

* Hands-on practice with **Medallion architecture**
* **Data cleaning & transformation** using SQL
* **Data modeling** with facts & dimensions
* Running **business analytics queries**
* Building a portfolio-ready **dashboard**

**👤 Author**

**Neelum Rana** – Aspiring Data Engineer  
📧 Contact: neelumsami@yahoo.com  
🔗 LinkedIn: www.linkedin.com/in/neelum-rana-413318374