Project 1: Supply Chain & Inventory Optimization System

Tools

MySQL (core) | Python (data wrangling) | Power BI (visualizations)

Business Problem

A global supply chain and logistics company wants to optimize its inventory management and delivery performance. The operations team is struggling with stock-outs, delayed shipments, and regional demand mismatches. As a data analyst, you are tasked to build SQL-driven insights to optimize restocking, identify delay patterns, and analyze supplier and warehouse efficiency.

Project Goals

- • Identify slow-moving and fast-moving products based on sales trends
- • Detect late shipment patterns by product, supplier, and region
- • Monitor stock availability and demand forecasting for warehouses
- • Rank products/suppliers based on fulfillment efficiency and delivery SLAs
- • Track rolling sales and inventory turnover using window functions

Core SQL Requirements

Must use:

- • Window functions (ROW_NUMBER(), RANK(), LAG(), LEAD(), SUM() OVER)
- • CTEs (recursive for date sequence generation, supplier tree mapping)
- • Correlated and non-correlated subqueries
- • Nested aggregation queries
- • Date-time functions for delay analysis

Key Business Queries to Solve

1. Which suppliers consistently miss delivery deadlines? Show % on-time delivery.
2. Identify top 5 slowest-moving SKUs in each category over the last 90 days.
3. Calculate inventory turnover ratio per warehouse monthly using window functions.
4. Which products are at risk of stock-out in the next 7 days, based on historical demand?
5. Provide a rolling 30-day sales and returns report for each product.
6. Which customers frequently cancel orders or return products? Segment them by behavior.
7. Create a monthly restocking recommendation table using a CTE that calculates needed stock = (avg daily sales * 30) – current stock.

Kaggle Dataset (Modified/Relational Ready)

https://www.kaggle.com/datasets/prachi13/customer-analytics
(Note: It contains granular order, product, fulfillment, and shipping details and can be transformed into normalized relational tables.)

Expected Student Workflow

- • Ingest the dataset and normalize it into relational tables
- • Clean the data in Python (missing dates, invalid deliveries)
- • Create MySQL schema, populate with INSERT scripts
- • Write SQL scripts answering the above business queries
- • Build Power BI dashboard for:
  o KPI cards: Stockout Risk, On-Time Delivery Rate, Inventory Turnover
  o Product-level performance
  o Supplier comparison
  o Geographic demand heatmap

Bonus Challenges

- • Create a recursive CTE to track supplier hierarchy if multiple layers are added (supplier → sub-supplier).
- • Use window functions to generate product sales ranks within regions and time windows.
- • Generate alert system SQL logic: Flag any product whose daily sales spike >150% above 7-day average.

Final Comparison Summary

Feature	Previous Version	Upgraded Version
Domain	E-Commerce Sales	Supply Chain & Logistics
Dataset	Superstore	Supply Chain Analytics
SQL Complexity	Joins, Aggregation	CTE, Window Functions, Subqueries
Core Focus	Descriptive BI	Optimization + Predictive Planning
Project Level	Medium	Medium to Hard