Task 2: Write Complex SQL Queries for Reporting

Objective:

To generate sales reports using complex SQL queries such as daily totals, average transaction value, top-selling products, and city-wise sales summaries. The aim was to simulate real-world reporting using SQL.

What I Built:

I created a Sales Reporting System using MySQL Workbench with tables for customers, products, orders, and order details. The database supports generation of key reports such as daily totals, average transactions, top products, and city-wise sales insights.

Why I Built It:

This project helps demonstrate my ability to design relational databases, use JOINs and aggregate functions, and extract meaningful business insights using SQL queries.

How I Built It:

1. Designed database schema with 4 tables (customers, products, orders, order_details).
2. Inserted realistic sample data to simulate transactions.
3. Wrote SQL queries using JOIN, GROUP BY, ORDER BY, and aggregate functions.
4. Generated reports and verified output in MySQL Workbench.

Reports Generated:

Report Type	Description
Daily Sales Totals	Shows total revenue generated per day.
Average Transaction Value	Displays average order amount across all transactions.
Top 3 Products by Revenue	Ranks the highest revenue-generating products.
City-wise Total Sales	Shows revenue contribution from each customer city.

Key Learnings:

- Designed relational schema and applied normalization principles.
- Learned advanced SQL features like GROUP BY, JOIN, and aggregate functions.
- Practiced transforming raw transactional data into analytical insights.
- Strengthened understanding of SQL-based reporting for data analytics.

Conclusion:

This project provided practical exposure to SQL data analysis and report generation. It demonstrated how structured queries can transform stored data into actionable business insights.