

Task 4: SQL for Data Analysis Summary

The primary objective of this task was to demonstrate mastery of core SQL concepts by analyzing an E-commerce dataset. This was achieved by normalizing the initial flat data into a relational structure and executing complex queries.

1. Key Accomplishments

- **Database Setup:** Overcame a common technical hurdle by successfully **resetting the forgotten MySQL root password** using the configuration file method, allowing access to the database environment.
- **Data Normalization:** Transformed the initial flat `Ecommerce.csv` file into a relational model consisting of three linked tables (`Orders`, `Customer_Demographics`, `Shipment_Details`) to enable complex analysis using JOINS.
- **Query Execution:** Executed a comprehensive SQL script (`Ecommerce_SQL_Database 2.sql`) that covered all analytical requirements outlined in the mini-guide.

2. Analytical Concepts Covered

The final SQL script successfully utilized all required concepts, preparing the data for the final deliverables:

SQL Concept	Example Use in Script
JOINS (INNER/LEFT)	Joining <code>Orders</code> and <code>Shipment_Details</code> to calculate aggregate metrics by <code>Mode_of_Shipment</code> .
Aggregate Functions	Used AVG() on <code>Cost_of_the_Product</code> and SUM() on <code>Discount_offered</code> .
GROUP BY & HAVING	Grouping results by <code>Warehouse_Block</code> and filtering the aggregated results with a HAVING clause (e.g., <code>AVG(Cost_of_the_Product) > 200</code>).
Subqueries	Writing a nested <code>SELECT</code> statement to filter orders where the <code>Discount_offered</code> was above the global average.
Views	Creating a VIEW (<code>High_Value_Shipments</code>) to simplify access to frequently analyzed, filtered data sets.