Project Report: Online Retail Sales Database Design

Introduction

This project focuses on designing and implementing a normalized database schema for an online retail sales platform. The aim is to handle core business operations such as managing customers, products, orders, payments, and shipments effectively using SQL.

Abstract

The Online Retail Sales Database is designed to store and retrieve data for an e-commerce application. It supports essential operations including tracking customer orders, managing product inventory, recording payments, and ensuring delivery through shippers. The schema is normalized and optimized for querying, reporting, and scalability. Data integrity and relationships are enforced using primary and foreign keys.

Tools Used

- **MySQL (phpMyAdmin / Workbench)** – for writing and executing SQL queries.
- **dbdiagram.io** – for designing the Entity Relationship Diagram (ERD).
- **VS Code / Notepad** – for maintaining .sql scripts and documentation.
- **GitHub** – for version control and submission.

Steps Involved in Building the Project

1. **Schema Design**: Identified six core entities: `Customers`, `Products`, `Orders`, `Order_Items`, `Payments`, and `Shippers`.
2. **ER Diagram**: Designed ERD showing one-to-many and many-to-many relationships.
3. **SQL Schema**: Wrote `CREATE TABLE` scripts with constraints (PK, FK).
4. **Data Population**: Populated tables using `INSERT INTO` statements from `.sql` files.
5. **Queries**:
    - Basic SELECT queries for retrieval.
    - Aggregate queries using `SUM`, `AVG`, `COUNT`, `GROUP BY`.
    - Join queries to connect related tables.
    - Views and subqueries for reusable logic and cleaner reports.
6. **Reporting**: Generated sales reports using JOINs and views.
7. **Stored Procedures and Functions**: Created reusable logic blocks to calculate revenue or process order summaries.

Conclusion

The EcommerceDB project provided hands-on experience in designing, normalizing, and managing relational databases. It improved skills in SQL querying, reporting, data integrity, and abstraction using views and procedures. This project can be extended further with analytics, user management, or front-end integration for real-world usage.