Online Retail Sales Database Design

Introduction

The *Online Retail Sales Database Design* project focuses on developing a normalized SQL database schema that replicates the backend of a retail e-commerce platform. As digital commerce continues to grow, scalable and efficient databases are essential for managing products, customers, orders, and transactions. This project helps simulate real-world tasks such as inventory tracking, customer management, and sales reporting in a relational database system.

The system is designed to handle key operations of an e-commerce platform, including order placement, payment tracking, and product management using structured data.

Abstract

This project involves designing a relational schema and implementing it using **MySQL**. It follows **Third Normal Form (3NF)** to ensure minimum redundancy and maximum data integrity. Six key entities—Customers, Products, Categories, Orders, OrderItems, and Payments—were defined, linked through primary and foreign keys, and visualized using an ER diagram.

Sample data was inserted into all tables, and SQL queries were written to perform data analysis, such as identifying top customers, tracking product sales, and summarizing payments. Views were also created to allow simplified, reusable reporting on key business metrics.

Tools Used

Tool	Purpose
MySQL Workbench	Database creation, SQL scripting, ERD design
MySQL Server	Execution of SQL scripts
ER Diagram (EER)	Visual representation of table relationships

Steps Involved in Building the Project

1. **Entity Identification**: Selected main entities – Products, Customers, Orders, Payments, Categories, and OrderItems.
2. **Schema Design & Normalization**: Created a 3NF schema to eliminate redundancy and support scalability.
3. **Table Creation (DDL)**: Defined SQL tables with constraints (primary keys, foreign keys, data types).
4. **Sample Data Insertion**: Inserted realistic values into each table for testing and demonstration.
5. **ER Diagram Creation**: Used MySQL Workbench to create an Enhanced Entity-Relationship (EER) diagram.
6. **Analytical Queries**: Executed JOINs and aggregate functions to generate insights.
7. **Views Creation**: Built reusable views like CustomerOrderSummary and ProductSalesReport.
8. **Export of Deliverables**: Saved the ER diagram and all SQL scripts for submission.

Sample Views and Queries

- **CustomerOrderSummary** – Total orders and spend per customer
- **ProductSalesReport** – Quantity sold and revenue per product
- **PaymentsByMethod** – Aggregated payments by method
- **HighValueCustomers** – Filters customers exceeding a purchase threshold

These help provide business insights into sales, product performance, and customer trends.

Conclusion

Through this project, I developed core database design and query-writing skills relevant to real-world e-commerce systems. It helped solidify my understanding of relational models, data integrity, and analytical reporting using SQL. This project aligns directly with the responsibilities of data analysts, backend developers, and SQL-based roles—making it highly resume-worthy and practical for placement.