Online Retail Sales Database Project

Introduction The project focuses on designing a normalized SQL database for an e-commerce platform that manages online retail sales. The database handles critical entities such as Customers, Products, Orders, Payments, and Order Items, allowing efficient management of sales, inventory, and transactions. The goal is to create a robust, scalable, and normalized database that supports querying, reporting, and analysis for business insights.

Abstract This project demonstrates the complete process of building a relational database for an online retail system. Key objectives include: - Identifying main entities and their relationships. - Designing an ER diagram to visually represent the database structure. - Normalizing the database to Third Normal Form (3NF) to eliminate redundancy. - Creating SQL tables with constraints for data integrity. - Populating the tables with sample data. - Executing JOIN queries and views to generate sales and payment reports.

The database ensures accurate tracking of customer orders, product inventory, and payment transactions, providing a foundation for reporting and business analytics.

Tools Used - Database Management System: MySQL / PostgreSQL - ER Diagram Tool: dbdiagram.io - Querying: SQL (DDL, DML, JOINs, aggregation functions) - Development Environment: MySQL Workbench / pgAdmin / Terminal

Steps Involved in Building the Project 1. Identifying Key Entities - Customers, Products, Orders, OrderItems, Payments. 2. Designing the ER Diagram - Created using dbdiagram.io. - Defined primary keys, foreign keys, and relationships. 3. Normalizing Schema to 3NF - Ensured atomic data (1NF), removed partial dependencies (2NF), and eliminated transitive dependencies (3NF). 4. Creating SQL Schema - Wrote CREATE TABLE statements with constraints like PRIMARY KEY, UNIQUE, and FOREIGN KEY. 5. Populating Tables with Sample Data - Inserted 10 customers, 5 products, 10 orders, and corresponding order items and payments. 6. Generating Queries and Reports - Sales by customer - Best-selling products - Payment summary

Conclusion The project successfully demonstrates the creation of a normalized e-commerce database capable of managing customer orders, product inventory, and payments efficiently. By following database normalization principles, the schema reduces redundancy and improves data integrity. The resulting database supports complex queries, making it suitable for sales analysis and reporting. This project serves as a foundation for building scalable, real-world online retail applications.