**🧾 Online Retail Sales Database Design Report**

**📘 1. Introduction.**

In today's technological era, e-commerce platforms require efficient and reliable database systems to manage large amounts of customers, products, and transactions data. This project aims to design a normalized SQL schema for an “Online Retail Sales System”. The database will support core business operations such as customer management, order processing, payment tracking, and inventory management while ensuring data consistency and integrity.

**📄 2. Abstract.**

This project involves designing a normalized relational database schema for an Online Retail Store using MySQL Workbench. The goal is to create a well-structured schema that eliminates data redundancy and supports efficient queries. The schema is normalized to Third Normal Form (3NF) and includes five core entities: Customers, Products, Orders, Order Items, and Payments. The design includes SQL scripts for table creation, sample data insertion, and example queries for generating sales reports. The final output includes an ER diagram, SQL schema, sample data, and reporting views.

**🛠 3. Tools Used.**

**Tool/Technology Purpose**

MySQL Workbench Database design, ER diagrams, SQL scripting.

SQL Language Creating tables, constraints, queries.

**🧱 4. Steps Involved in Building the Project.**

**✅ Step 1:** Identify Key Entities and Attributes

Customers: customer id, name, email, phone, address

Products: product id, name, description, price, stock qty

Orders: order id, customer id, order date, total amount

Order Items: order item id, order id, product id, quantity, unit price

Payments: payment id, order id, payment date, amount, payment method

**✅ Step 2**: Normalize the Schema to 3NF

1NF: All columns contain atomic values

2NF: All non-key attributes depend on the full primary key

3NF: No transitive dependencies exist

**✅ Step 3:** Create SQL Schema (DDL)

Tables were created using SQL CREATE TABLE with PRIMARY KEY and FOREIGN KEY constraints to enforce referential integrity.

**✅ Step 4:** Populate Tables with Sample Data

Used INSERT INTO queries to insert mock data for customers, products, orders, and payments.

**✅ Step 5:** Reverse Engineer to ER Diagram

Used MySQL Workbench’s Reverse Engineer tool to generate the ER diagram from the SQL schema.

**✅ Step 6**: Write JOIN Queries and Views

Created analytical SQL queries and views to:

Display customer order summaries.

Show total sales per product.

Track payment methods and amounts.

**✅ 5. Conclusion.**

This project successfully demonstrates how to design a normalized SQL database for an online retail platform using MySQL Workbench. By applying 3NF normalization, the database avoids data redundancy and improves consistency. The ER diagram clearly shows the relationships among entities. The generated SQL schema, sample data, and sales queries help simulate real-world e-commerce operations. This design set a solid foundation for building a scalable and efficient backend for any online retail application.