

## **INTRODUCTION**

The **Online Retail Sales Database Project** is designed to simulate the backend data management of a real-world e-commerce platform. This system models core retail operations such as product listings, customer details, order management, and payment tracking. The goal of this project was to apply advanced SQL skills including schema design, data population, query optimization and stored routines to create a professional and scalable database.

## **ABSTRACT**

This project involved designing a relational database from scratch using **MySQL**. It focused on creating normalized tables, establishing relationships through foreign keys and building complex SQL logic to analyse and manage retail data. Additionally, business-level insights were generated using JOINS, GROUP BYs, aggregate functions and views. Realistic sample data was created to test the schema, allowing for queries related to top-selling products, customer spending, pending payments, and category-wise sales trends.

## **Tools Used->**

- **MySQL Workbench 8.0** – SQL development and schema creation
- **dbdiagram.io** – ER diagram modelling and visualization
- **GitHub** – Version control and project documentation

## **Steps Involved in Building the Project->**

### **1. Requirement Analysis & Planning**

Identified key entities: Customers, Categories, Products, Orders, Order Items, Payments.

### **2. Schema Design & ER Diagram**

Built a fully normalized schema with primary and foreign key constraints. Designed a clear ER diagram to represent table relationships.

### **3. Database Creation (DDL)**

Used CREATE TABLE statements with constraints like AUTO\_INCREMENT, NOT NULL, UNIQUE, and FOREIGN KEY.

### **4. Sample Data Insertion (DML)**

Inserted realistic test data into all tables to mimic real business scenarios (e.g., orders with multiple items, pending payments).

### **5. SQL Querying**

Wrote SELECT queries using JOIN, GROUP BY, ORDER BY, LIMIT, and HAVING to extract insights.

### **6. Views & Abstractions**

Created views for product sales, customer order history, and incomplete payments to simplify recurring business logic.

### **7. Function** GetTotalSpentByCustomer() function to compute total spend

## **CONCLUSION**

This project provided a hands-on understanding of database design, SQL logic, and analytics from a retail perspective. It strengthened key skills in database normalization, relationship modelling, DDL/DML writing, and the use of advanced SQL features like views and stored routines. This project is a strong portfolio piece that demonstrates the practical application of SQL in solving business challenges and preparing data for analysis.

**Submitted by:**

**Dikshant Kumar Singh**

SQL Developer Internship – Final Project

July 2025

**Submitted to:**

**Elevate Labs**