**Online Retail Shop Database Project**

**Introduction**

This project involves designing a secure, normalized MySQL database for an e-commerce platform. It supports the management of customers, sellers, orders, items, feedback, and transactions through efficient relational design and automation using triggers and views.

**Abstract**

The **Online Retail Shop Database** simulates the backend of a shopping site, featuring:

* Users (Buyers/Sellers)
* Orders, Products, Payments, Cart
* Coupons and Feedback

It uses:

* **Triggers** for automation (e.g., setting order status, validating coupons)
* **Views** for analytics (e.g., order summaries, top-selling items)

The database follows 3NF and maintains integrity via foreign key constraints. It’s ready for integration with front-end applications.

**Tools Used**

| **Tool** | **Purpose** |
| --- | --- |
| MySQL Workbench | ER Diagram & Design |
| MySQL Server | Query Execution |
| SQL Files | Schema, Triggers, Views |
| .mwb File | Visual Modeling |
| Text Editor | Script Writing |

**Project Steps**

**1. Schema Design**

* Identified key entities (Users, Orders, Items, etc.)
* Designed ER diagram in MySQL Workbench

**2. Table Creation**

* Created normalized tables with PK–FK relationships

**3. Relationships**

* Linked customers with orders, sellers with items, etc.

**4. Triggers Implemented**

* set\_initial\_order\_status – Inserts initial status for orders
* prevent\_expired\_coupon – Blocks expired coupon usage
* mark\_coupon\_used – Marks a coupon as used post-order
* cleanup\_after\_order\_delete – Removes related records on delete
* update\_order\_total\_after\_order\_item\_insert – Updates total amount
* validate\_password\_length – Validates user password length
* update\_user\_timestamp – Auto-updates timestamps on change

**5. Views Created**

* customer\_order\_summary
* seller\_sales\_summary
* coupon\_usage\_report
* top\_selling\_items, item\_average\_rating
* product\_comments, customer\_cart\_summary, inactive\_customers

**6. Sample Data**

* Populated tables with users, items, orders, feedback, logins

**7. Testing**

* Executed queries to validate logic and correct errors like NULL total\_amount

**Conclusion**

This project demonstrates complete backend development for an online retail platform using MySQL. With well-structured schemas, automated triggers, and insightful views, it forms a scalable, production-ready system that can easily be expanded with front-end or API layers