DBMS

SEMESTER PROJECT

Prepared By :

Muhammad Waleed(110839)

Aakif Saleem(110840)

Submitted to:

Mam Sehrish Khan

SEMESTER PROJECT

DATABASE MANAGEMENT SYSTEM

SUBMITTED BY:

Muhammad Waleed
Aakif Saleem

SUBMITTED TO:

Mam Sehrish Khan

PROJECT

Table of Content:

1-Project Overview

1.1 Introduction:

This project is for a fashion clothing store chain that launches seasonal collections like Spring, Fall, and Wedding Collections. Each product is available in different sizes, colors, and fabric types. The store’s inventory is connected and synced with its online website.
Customers can return products both in-store and online, and they can give reasons like defective item, wrong size, or product doesn't match the online description. The system keeps track of all returns, and if any product is returned too many times, it is flagged for review.
The system also saves customer preferences so the company can send personalized marketing offers.

1.2 Objective of the Project

- To design a database system for a clothing store with seasonal product collections.
- To manage product details like size, color, and fabric.
- To sync inventory between physical stores and online portal.
- To create a return system where customers can give a reason.
- To identify high-return items for quality checks.
- To store customer preferences for personalized promotions.
- To apply database concepts like ERD, normalization, and SQL queries**.**

1.3 Scope of the System

The system includes the following features:

- Managing product details (size, color, fabric).
- Linking products to seasonal collections.
- Keeping inventory updated across store and online.
- Handling product returns with proper reasons.
- Flagging high-return items for further review.
- Saving customer preferences for better marketing.
- The system is for use by store managers and online staff.
- This project focuses only on the backend (database), not the frontend or website design.

1.4 Tools & Technologies Used

- **Database Software:** MySQL

- **ERD Tool:** dbdiagram.io / Draw.io
- **SQL Editor:** MySQL Workbench
- **Documentation:** Microsoft Word / Google Docs

2-Entity Relationship Diagram (ERD)

2.1 ERD Description:

The Entity Relationship Diagram (ERD) of the Clothing Store Chain system shows how the key entities are connected in the database. The main entities are:

- **Product**, which includes details like size, color, fabric type, season, and description.
- **Customer**, who can place multiple orders.
- **Order**, which contains multiple **OrderItems**, and each OrderItem is linked to a specific product.
- **Inventory**, which tracks how many items of each product are available at different store locations.
- **Return**, which is related to an OrderItem and includes the reason and return date.
- **Review Flag**, which is used to mark items with high return rates.
- **Preferences**, which stores customer interests for personalized marketing.

The relationships include:

- A **Customer places Orders**
- An **Order contains multiple OrderItems**
- Each **OrderItem refers to one Product**
- **Products are listed in Inventory** at store locations

- **OrderItems can be returned**, and these returns can be **flagged**
- A **Customer has Preferences**

This ERD ensures that every return is traceable, every product is properly managed in inventory, and customer data helps in marketing personalization.

3.2 Explanation of Entities and Relationships

Entities:

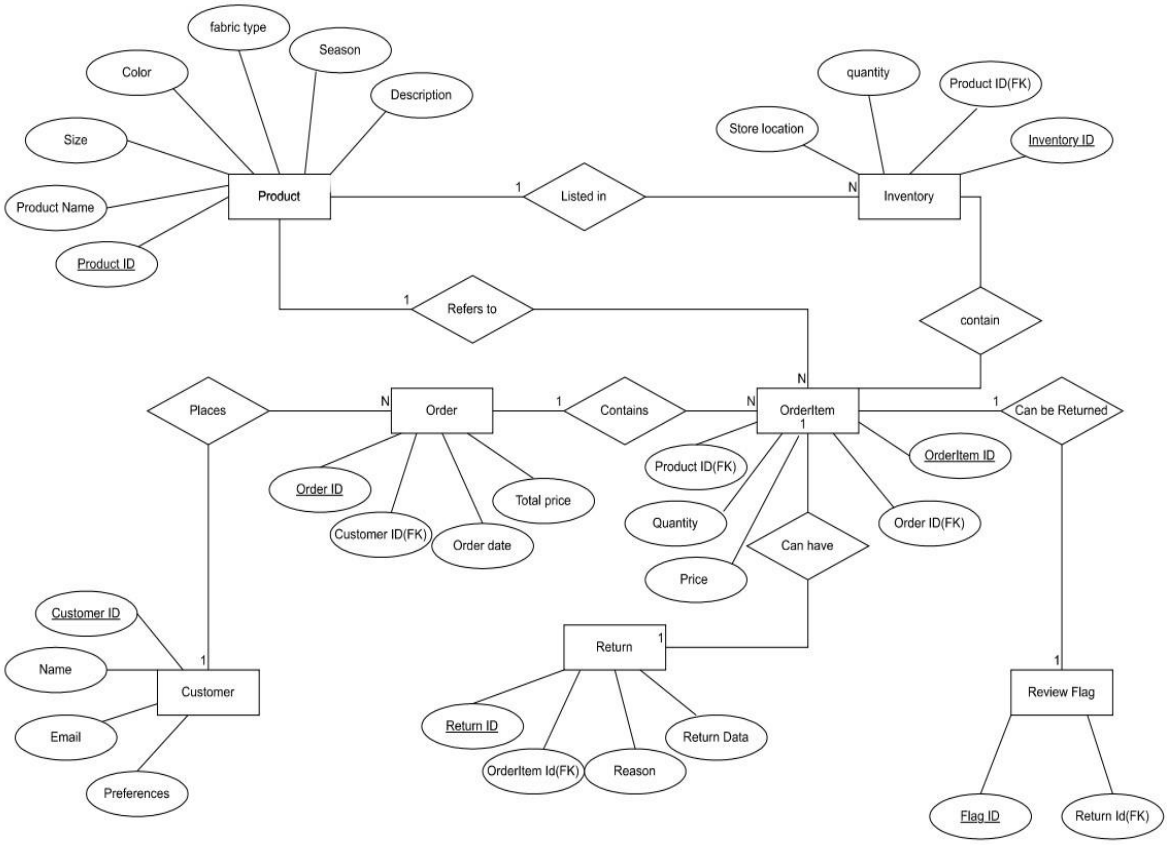
1. **Product**
    - Attributes: ProductID, ProductName, Size, Color, FabricType, Season, Description
    - Each product can belong to multiple seasonal lines and is listed in the inventory.
2. **Customer**
    - Attributes: CustomerID, Name, Email, Preferences
    - Customers place orders and have preferences used for personalized marketing.
3. **Inventory**
    - Attributes: InventoryID, StoreLocation, Quantity, ProductID (FK)
    - Tracks stock of each product at a specific location.
4. **Order**
    - Attributes: OrderID, OrderDate, TotalPrice, CustomerID (FK)
    - Represents a transaction by a customer.
5. **OrderItem**
    - Attributes: OrderItemID, OrderID (FK), ProductID (FK), Quantity, Price
    - A bridge entity between Order and Product to handle multiple products per order.

6. **Return**
    - Attributes: ReturnID, OrderItemID (FK), Reason, ReturnDate
    - Customers can return order items with specific reasons.
7. **ReviewFlag**
    - Attributes: FlagID, ReturnID (FK)
    - Flags items with high return rates for review and quality checks.

Relationships:

- **Customer → Order**:
  One customer can place many orders (1:N relationship).
- **Order → OrderItem**:
  One order contains multiple order items (1:N relationship).
- **Product → OrderItem**:
  One product can appear in many order items (1:N relationship).
- **OrderItem → Return**:
  One order item can have one return (1:1 relationship).
- **Return → ReviewFlag**:
  One return can be flagged for review (1:1 relationship).
- **Product → Inventory**:
  One product can be listed in many inventory locations (1:N relationship).

ERD Diagram:



3- Normalization

3.1 Normalization (1NF, 2NF, 3NF with Examples)

First Normal Form (1NF):

- No repeating groups or arrays.
- Each cell contains atomic (indivisible) values.

Example (before 1NF):

ProductID	ProductName	Sizes	Colors
P01	Shirt	S, M, L	Red, Blue

After 1NF:

ProductID	ProductName	Size	Color
P01	Shirt	S	Red
P01	Shirt	M	Blue

Second Normal Form (2NF):

- Be in 1NF
- Remove partial dependencies (i.e., non-key attributes should depend on the whole primary key)

Example (before 2NF):

OrderItemID	OrderID	ProductID	Quantity	ProductName

Issue: ProductName depends only on ProductID, not the full key (OrderItemID).

After 2NF:

- Separate Product table:
  - ProductID, ProductName
- OrderItem only keeps FK reference to ProductID

- Be in 2NF
- No transitive dependencies (non-key attributes shouldn't depend on other non-key attributes)

Example (before 3NF):

CustomerID	Name	Email	Domain
C01	Ali	ali@gmail.com	gmail.com

Issue: Domain depends on Email (not directly on CustomerID).
After 3NF:

- Remove Domain attribute
- Keep: CustomerID, Name, Email

3.3 Final Table Structures after Normalization

1. **Product** (ProductID, ProductName, Size, Color, FabricType, Season, Description)
2. **Inventory** (InventoryID, StoreLocation, Quantity, ProductID FK)
3. **Customer** (CustomerID, Name, Email, Preferences)
4. **Order** (OrderID, OrderDate, TotalPrice, CustomerID FK)
5. **OrderItem** (OrderItemID, OrderID FK, ProductID FK, Quantity, Price)
6. **Return** (ReturnID, OrderItemID FK, Reason, ReturnDate)
7. **ReviewFlag** (FlagID, ReturnID FK)

4-SQL Schema and Table Creation Queries

4.1 SQL CREATE TABLE Statements

First we have to create a database which is shown in below image:

```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| college            |
| dept               |
| employee           |
| home               |
| information_schema |
| mysql              |
| performance_schema |
| sys                |
| waleed             |
+--------------------+
9 rows in set (0.00 sec)

mysql> create database clothing_store_chain;
Query OK, 1 row affected (0.11 sec)

mysql> show databases;
+----------------------+
| Database             |
+----------------------+
| clothing_store_chain |
| college              |
| dept                 |
| employee             |
| home                 |
| information_schema   |
| mysql                |
| performance_schema   |
| sys                  |
| waleed               |
+----------------------+
10 rows in set (0.00 sec)

mysql> use clothing_store_chain;
Database changed
mysql> show tables;
Empty set (0.00 sec)

mysql>
```

Now we have to create tables according to our ERD diagram

```
mysql> use clothing_store_chain;
Database changed
mysql> show tables;
Empty set (0.00 sec)

mysql> create table Product(ProductID INT(10) PRIMARY KEY,ProductName VARCHAR(100),Color VARCHAR(50),Size VARCHAR(20),FabricType VARCHAR(50),Season VARCHAR(50),Descript
ion TEXT);
Query OK, 0 rows affected, 1 warning (0.64 sec)

mysql> CREATE TABLE Inventory (InventoryID INT(10) PRIMARY KEY,ProductID INT(10),StoreLocation VARCHAR(100),Quantity INT(10));
Query OK, 0 rows affected, 3 warnings (0.29 sec)

mysql> create table Customer (CustomerID INT(10) PRIMARY KEY,Name VARCHAR(100),Email VARCHAR(100),Preferences VARCHAR(255));
Query OK, 0 rows affected, 1 warning (0.73 sec)

mysql> create table Order(OrderID INT(10) PRIMARY KEY,CustomerID INT(10),OrderDate DATE,TotalPrice INT(10));
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'Order(OrderID
 INT(10) PRIMARY KEY,CustomerID INT(10),OrderDate DATE,TotalPrice I' at line 1
mysql> create table Orders(OrderID INT(10) PRIMARY KEY,CustomerID INT(10),OrderDate DATE,TotalPrice INT(10));
Query OK, 0 rows affected, 3 warnings (0.26 sec)

mysql> create table OrderItem (OrderItemID INT()10 PRIMARY KEY,OrderID INT(10),ProductID INT(10),Quantity INT(10),Price INT(100));
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near ')10 PRIMARY K
EY,OrderID INT(10),ProductID INT(10),Quantity INT(10),Price INT(100' at line 1
mysql> create table OrderItem (OrderItemID INT PRIMARY KEY,OrderID INT,ProductID INT,Quantity INT,Price INT);
Query OK, 0 rows affected (0.28 sec)

mysql> create table ProductReturn(ReturnID INT PRIMARY KEY,OrderItemID INT,ReturnDate DATE,Reason TEXT);
Query OK, 0 rows affected (0.26 sec)

mysql> create table ReviewFlag (FlagID INT PRIMARY KEY,ReturnID INT);
Query OK, 0 rows affected (0.69 sec)
```

```
mysql> show tables;
+--------------------------------+
| Tables_in_clothing_store_chain |
+--------------------------------+
| customer                       |
| inventory                      |
| orderitem                      |
| orders                         |
| product                        |
| productreturn                  |
| reviewflag                     |
+--------------------------------+
7 rows in set (0.00 sec)

mysql> desc customer;
+-------------+--------------+------+-----+---------+-------+
| Field       | Type         | Null | Key | Default | Extra |
+-------------+--------------+------+-----+---------+-------+
| CustomerID  | int          | NO   | PRI | NULL    |       |
| Name        | varchar(100) | YES  |     | NULL    |       |
| Email       | varchar(100) | YES  |     | NULL    |       |
| Preferences | varchar(255) | YES  |     | NULL    |       |
+-------------+--------------+------+-----+---------+-------+
4 rows in set (0.00 sec)

mysql> desc inventory;
+---------------+--------------+------+-----+---------+-------+
| Field         | Type         | Null | Key | Default | Extra |
+---------------+--------------+------+-----+---------+-------+
| InventoryID   | int          | NO   | PRI | NULL    |       |
| ProductID     | int          | YES  |     | NULL    |       |
| StoreLocation | varchar(100) | YES  |     | NULL    |       |
| Quantity      | int          | YES  |     | NULL    |       |
+---------------+--------------+------+-----+---------+-------+
4 rows in set (0.00 sec)

mysql> desc orderitem;
+-------------+------+------+-----+---------+-------+
| Field       | Type | Null | Key | Default | Extra |
+-------------+------+------+-----+---------+-------+
| OrderItemID | int  | NO   | PRI | NULL    |       |
| OrderID     | int  | YES  |     | NULL    |       |
| ProductID   | int  | YES  |     | NULL    |       |
| Quantity    | int  | YES  |     | NULL    |       |
| Price       | int  | YES  |     | NULL    |       |
+-------------+------+------+-----+---------+-------+
5 rows in set (0.00 sec)

mysql> desc order;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'order' at lin
e 1
mysql> desc orders;
+------------+------+------+-----+---------+-------+
| Field      | Type | Null | Key | Default | Extra |
+------------+------+------+-----+---------+-------+
| OrderID    | int  | NO   | PRI | NULL    |       |
| CustomerID | int  | YES  |     | NULL    |       |
| OrderDate  | date | YES  |     | NULL    |       |
| TotalPrice | int  | YES  |     | NULL    |       |
+------------+------+------+-----+---------+-------+
4 rows in set (0.00 sec)

mysql> desc product;
+-------------+--------------+------+-----+---------+-------+
| Field       | Type         | Null | Key | Default | Extra |
+-------------+--------------+------+-----+---------+-------+
| ProductID   | int          | NO   | PRI | NULL    |       |
| ProductName | varchar(100) | YES  |     | NULL    |       |
| Color       | varchar(50)  | YES  |     | NULL    |       |
| Size        | varchar(20)  | YES  |     | NULL    |       |
| FabricType  | varchar(50)  | YES  |     | NULL    |       |
| Season      | varchar(50)  | YES  |     | NULL    |       |
| Description | text         | YES  |     | NULL    |       |
+-------------+--------------+------+-----+---------+-------+
7 rows in set (0.00 sec)

mysql> desc productreturn;
+--------------+------+------+-----+---------+-------+
| Field        | Type | Null | Key | Default | Extra |
+--------------+------+------+-----+---------+-------+
| ReturnID     | int  | NO   | PRI | NULL    |       |
| OrderItemID  | int  | YES  |     | NULL    |       |
| ReturnDate   | date | YES  |     | NULL    |       |
| Reason       | text | YES  |     | NULL    |       |
+--------------+------+------+-----+---------+-------+
4 rows in set (0.00 sec)

mysql> desc reviewFlag;
+----------+------+------+-----+---------+-------+
| Field    | Type | Null | Key | Default | Extra |
+----------+------+------+-----+---------+-------+
| FlagID   | int  | NO   | PRI | NULL    |       |
| ReturnID | int  | YES  |     | NULL    |       |
+----------+------+------+-----+---------+-------+
2 rows in set (0.00 sec)

mysql>
```

Inserting values:

```
mysql> insert into product values(1, 'Cotton T-Shirt', 'Blue', 'M', 'Cotton', 'Summer', 'Comfortable summer t-shirt');
Query OK, 1 row affected (0.14 sec)

mysql> insert into product values(2, 'Jeans', 'Black', 'L', 'leather', 'Winter', 'Stylish black jeans');
Query OK, 1 row affected (0.09 sec)

mysql> insert into product values(3, 'Sweater', 'Grey', 'M', 'Wool', 'Winter', 'Warm woolen sweater');
Query OK, 1 row affected (0.05 sec)

mysql> insert into product values(4, 'Shirt', 'White', 'S', 'cotton', 'Summer', 'Lightweight shirt');
Query OK, 1 row affected (0.06 sec)

mysql> insert into product values(5, 'Jacket', 'Brown', 'L', 'Leather', 'Winter', 'Leather winter jacket');
Query OK, 1 row affected (0.05 sec)

mysql> select* from product;
+-----------+----------------+-------+------+------------+--------+----------------------------+
| ProductID | ProductName    | Color | Size | FabricType | Season | Description                |
+-----------+----------------+-------+------+------------+--------+----------------------------+
|         1 | Cotton T-Shirt | Blue  | M    | Cotton     | Summer | Comfortable summer t-shirt |
|         2 | Jeans          | Black | L    | leather    | Winter | Stylish black jeans        |
|         3 | Sweater        | Grey  | M    | Wool       | Winter | Warm woolen sweater        |
|         4 | Shirt          | White | S    | cotton     | Summer | Lightweight shirt          |
|         5 | Jacket         | Brown | L    | Leather    | Winter | Leather winter jacket      |
+-----------+----------------+-------+------+------------+--------+----------------------------+
5 rows in set (0.00 sec)
```

```
mysql> insert into inventory vlaues(101, 1, 'Lahore', 20);
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'vlaues(101, 1
, 'Lahore', 20)' at line 1
mysql> insert into inventory vlaues(101, 1, 'Lahore', 20);
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'vlaues(101, 1
, 'Lahore', 20)' at line 1
mysql> insert into inventory vlaues(101, 1, 'Lahore', 20);
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'vlaues(101, 1
, 'Lahore', 20)' at line 1
mysql> insert into inventory values(101, 1, 'Lahore', 20);
Query OK, 1 row affected (0.08 sec)

mysql> insert into inventory values(102, 2, 'Karachi', 15);
Query OK, 1 row affected (0.08 sec)

mysql> insert into inventory values(103, 3, 'Islamabad', 10);
Query OK, 1 row affected (0.07 sec)

mysql> insert into inventory values(104, 4, 'Multan', 25);
Query OK, 1 row affected (0.10 sec)

mysql> insert into inventory values(105, 5, 'Peshawar', 12);
Query OK, 1 row affected (0.15 sec)

mysql> select* from inventory;
+-------------+-----------+---------------+----------+
| InventoryID | ProductID | StoreLocation | Quantity |
+-------------+-----------+---------------+----------+
|         101 |         1 | Lahore        |       20 |
|         102 |         2 | Karachi       |       15 |
|         103 |         3 | Islamabad     |       10 |
|         104 |         4 | Multan        |       25 |
|         105 |         5 | Peshawar      |       12 |
+-------------+-----------+---------------+----------+
5 rows in set (0.00 sec)
```

```
mysql> insert into customer values(1001, 'Ali', 'ali@gmail.com', 'Summer,formal');
Query OK, 1 row affected (0.08 sec)

mysql> insert into customer values(1002, 'Ahmad', 'ahmad@gmail.com', 'Winter, Formal');
Query OK, 1 row affected (0.08 sec)

mysql> insert into customer values(1003, 'Usman', 'usman@gmail.com', 'Casual, T-Shirts');
Query OK, 1 row affected (0.08 sec)

mysql> insert into customer values(1004, 'Waleed', 'waleed@gmail.com', 'Jeans, Jackets');
Query OK, 1 row affected (0.08 sec)

mysql> insert into customer values(1005, 'Umer', 'umer@gmail.com', 'Shirts, Winter');
Query OK, 1 row affected (0.08 sec)

mysql> select* from customer;
+------------+--------+------------------+------------------+
| CustomerID | Name   | Email            | Preferences      |
+------------+--------+------------------+------------------+
|       1001 | Ali    | ali@gmail.com    | Summer,formal    |
|       1002 | Ahmad  | ahmad@gmail.com  | Winter, Formal   |
|       1003 | Usman  | usman@gmail.com  | Casual, T-Shirts |
|       1004 | Waleed | waleed@gmail.com | Jeans, Jackets   |
|       1005 | Umer   | umer@gmail.com   | Shirts, Winter   |
+------------+--------+------------------+------------------+
5 rows in set (0.00 sec)
```

```
mysql> insert into orders values((202, 1002, '2025-05-02', 3500),(203, 1003, '2025-05-03', 1500),(204, 1004, '2025-05-04', 4500),(205, 1005, '2025-05-05', 2800));
ERROR 1241 (21000): Operand should contain 1 column(s)
mysql> insert into orders values((202, 1002, '2025-05-02', 3500);
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '' at line 1
mysql> insert into orders values((202, 1002, '2025-05-02', 3500);
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '' at line 1
mysql> insert into orders values(202, 1002, '2025-05-02', 3500);
Query OK, 1 row affected (0.04 sec)

mysql> insert into orders values(203, 1003, '2025-05-03', 1500);
Query OK, 1 row affected (0.06 sec)

mysql> insert into orders values(204, 1004, '2025-05-04', 4500);
Query OK, 1 row affected (0.10 sec)

mysql> insert into orders values(205, 1005, '2025-05-05', 2800);
Query OK, 1 row affected (0.10 sec)

mysql> select* from orders;
+---------+------------+------------+------------+
| OrderID | CustomerID | OrderDate  | TotalPrice |
+---------+------------+------------+------------+
|     201 |       1001 | 2025-05-01 |       2000 |
|     202 |       1002 | 2025-05-02 |       3500 |
|     203 |       1003 | 2025-05-03 |       1500 |
|     204 |       1004 | 2025-05-04 |       4500 |
|     205 |       1005 | 2025-05-05 |       2800 |
+---------+------------+------------+------------+
5 rows in set (0.00 sec)

mysql> insert into orderitem values(301, 201, 1, 2, 1000);
Query OK, 1 row affected (0.08 sec)

mysql> insert into orderitem values(302, 202, 2, 1, 3500);
Query OK, 1 row affected (0.08 sec)

mysql> insert into orderitem values(303, 203, 4, 3,1500);
Query OK, 1 row affected (0.08 sec)

mysql> insert into orderitem values(304, 204, 5, 1, 4500);
Query OK, 1 row affected (0.06 sec)

mysql> insert into orderitem values(305, 205, 3, 2, 1400);
Query OK, 1 row affected (0.09 sec)

mysql> select* from orderitem;
+-------------+---------+-----------+----------+-------+
| OrderItemID | OrderID | ProductID | Quantity | Price |
+-------------+---------+-----------+----------+-------+
|         301 |     201 |         1 |        2 |  1000 |
|         302 |     202 |         2 |        1 |  3500 |
|         303 |     203 |         4 |        3 |  1500 |
|         304 |     204 |         5 |        1 |  4500 |
|         305 |     205 |         3 |        2 |  1400 |
+-------------+---------+-----------+----------+-------+
5 rows in set (0.00 sec)
```

```
mysql> insert into productreturn values(401, 301, '2025-05-06', 'Wrong size');
Query OK, 1 row affected (0.07 sec)

mysql> insert into productreturn values(402, 302, '2025-05-07', 'Defective item');
Query OK, 1 row affected (0.10 sec)

mysql> insert into productreturn values(403, 303, '2025-05-08', 'Color mismatch');
Query OK, 1 row affected (0.08 sec)

mysql> insert into productreturn values(404, 304, '2025-05-09', 'Not satisfied');
Query OK, 1 row affected (0.20 sec)

mysql> insert into productreturn values(405, 305, '2025-05-10', 'Late delivery');
Query OK, 1 row affected (0.07 sec)

mysql> select* from productreturn;
+----------+-------------+------------+----------------+
| ReturnID | OrderItemID | ReturnDate | Reason         |
+----------+-------------+------------+----------------+
|      401 |         301 | 2025-05-06 | Wrong size     |
|      402 |         302 | 2025-05-07 | Defective item |
|      403 |         303 | 2025-05-08 | Color mismatch |
|      404 |         304 | 2025-05-09 | Not satisfied  |
|      405 |         305 | 2025-05-10 | Late delivery  |
+----------+-------------+------------+----------------+
5 rows in set (0.00 sec)
```

```
mysql> insert into reviewflag values(501, 401);
Query OK, 1 row affected (0.09 sec)

mysql> insert into reviewflag values(502, 402);
Query OK, 1 row affected (0.06 sec)

mysql> insert into reviewflag values(503,403);
Query OK, 1 row affected (0.06 sec)

mysql> insert into reviewflag values(504,404);
Query OK, 1 row affected (0.09 sec)

mysql> insert into reviewflag values(505,405);
Query OK, 1 row affected (0.16 sec)

mysql> select* from reviewflag;
+--------+----------+
| FlagID | ReturnID |
+--------+----------+
|    501 |      401 |
|    502 |      402 |
|    503 |      403 |
|    504 |      404 |
|    505 |      405 |
+--------+----------+
5 rows in set (0.00 sec)

mysql>
```

4.2 Primary and Foreign Key Constraints

we have to add foreign keys in it using the queries which have shown below:

```
mysql> ALTER TABLE Inventory ADD CONSTRAINT fk_inventory_product FOREIGN KEY (ProductID) REFERENCES Product(ProductID);
Query OK, 0 rows affected (1.91 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc inventory;
+--------------+--------------+------+-----+---------+-------+
| Field        | Type         | Null | Key | Default | Extra |
+--------------+--------------+------+-----+---------+-------+
| InventoryID  | int          | NO   | PRI | NULL    |       |
| ProductID    | int          | YES  | MUL | NULL    |       |
| StoreLocation | varchar(100) | YES  |     | NULL    |       |
| Quantity     | int          | YES  |     | NULL    |       |
+--------------+--------------+------+-----+---------+-------+
4 rows in set (0.00 sec)

mysql> ALTER TABLE Orders ADD CONSTRAINT fk_orders_customer FOREIGN KEY (CustomerID) REFERENCES Customer(CustomerID);
Query OK, 0 rows affected (0.56 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc customer;
+-------------+--------------+------+-----+---------+-------+
| Field       | Type         | Null | Key | Default | Extra |
+-------------+--------------+------+-----+---------+-------+
| CustomerID  | int          | NO   | PRI | NULL    |       |
| Name        | varchar(100) | YES  |     | NULL    |       |
| Email       | varchar(100) | YES  |     | NULL    |       |
| Preferences | varchar(255) | YES  |     | NULL    |       |
+-------------+--------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
mysql> desc orders;
+------------+------+------+-----+---------+-------+
| Field      | Type | Null | Key | Default | Extra |
+------------+------+------+-----+---------+-------+
| OrderID    | int  | NO   | PRI | NULL    |       |
| CustomerID | int  | YES  | MUL | NULL    |       |
| OrderDate  | date | YES  |     | NULL    |       |
| TotalPrice | int  | YES  |     | NULL    |       |
+------------+------+------+-----+---------+-------+
4 rows in set (0.00 sec)

mysql> ALTER TABLE OrderItem ADD CONSTRAINT fk_orderitem_order FOREIGN KEY (OrderID) REFERENCES Orders(OrderID);
Query OK, 0 rows affected (0.59 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc orderitem;
+-------------+------+------+-----+---------+-------+
| Field       | Type | Null | Key | Default | Extra |
+-------------+------+------+-----+---------+-------+
| OrderItemID | int  | NO   | PRI | NULL    |       |
| OrderID     | int  | YES  | MUL | NULL    |       |
| ProductID   | int  | YES  |     | NULL    |       |
| Quantity    | int  | YES  |     | NULL    |       |
| Price       | int  | YES  |     | NULL    |       |
+-------------+------+------+-----+---------+-------+
5 rows in set (0.00 sec)

mysql> ALTER TABLE OrderItem ADD CONSTRAINT fk_orderitem_product FOREIGN KEY (ProductID) REFERENCES Product(ProductID);
Query OK, 0 rows affected (1.28 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc orderitem;
+-------------+------+------+-----+---------+-------+
| Field       | Type | Null | Key | Default | Extra |
+-------------+------+------+-----+---------+-------+
| OrderItemID | int  | NO   | PRI | NULL    |       |
| OrderID     | int  | YES  | MUL | NULL    |       |
| ProductID   | int  | YES  | MUL | NULL    |       |
| Quantity    | int  | YES  |     | NULL    |       |
| Price       | int  | YES  |     | NULL    |       |
+-------------+------+------+-----+---------+-------+
5 rows in set (0.00 sec)
```

```
mysql> ALTER TABLE ReturnTable ADD CONSTRAINT fk_return_orderitem FOREIGN KEY (OrderItemID) REFERENCES OrderItem(OrderItemID);
ERROR 1146 (42S02): Table 'clothing_store_chain.returntable' doesn't exist
mysql> ALTER TABLE ProductReturn ADD CONSTRAINT fk_return_orderitem FOREIGN KEY (OrderItemID) REFERENCES OrderItem(OrderItemID);
Query OK, 0 rows affected (1.31 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc productreturn;
+-------------+------+------+-----+---------+-------+
| Field       | Type | Null | Key | Default | Extra |
+-------------+------+------+-----+---------+-------+
| ReturnID    | int  | NO   | PRI | NULL    |       |
| OrderItemID | int  | YES  | MUL | NULL    |       |
| ReturnDate  | date | YES  |     | NULL    |       |
| Reason      | text | YES  |     | NULL    |       |
+-------------+------+------+-----+---------+-------+
4 rows in set (0.00 sec)

mysql> ALTER TABLE ReviewFlag ADD CONSTRAINT fk_reviewflag_returnm FOREIGN KEY (ReturnID) REFERENCES ReturnTable(ReturnID);
ERROR 1824 (HY000): Failed to open the referenced table 'returntable'
mysql> ALTER TABLE ReviewFlag ADD CONSTRAINT fk_reviewflag_returnm FOREIGN KEY (ReturnID) REFERENCES ProductReturn(ReturnID);
Query OK, 0 rows affected (0.88 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc reviewflag;
+----------+------+------+-----+---------+-------+
| Field    | Type | Null | Key | Default | Extra |
+----------+------+------+-----+---------+-------+
| FlagID   | int  | NO   | PRI | NULL    |       |
| ReturnID | int  | YES  | MUL | NULL    |       |
+----------+------+------+-----+---------+-------+
2 rows in set (0.00 sec)

mysql>
```

6-SQL Other Queries

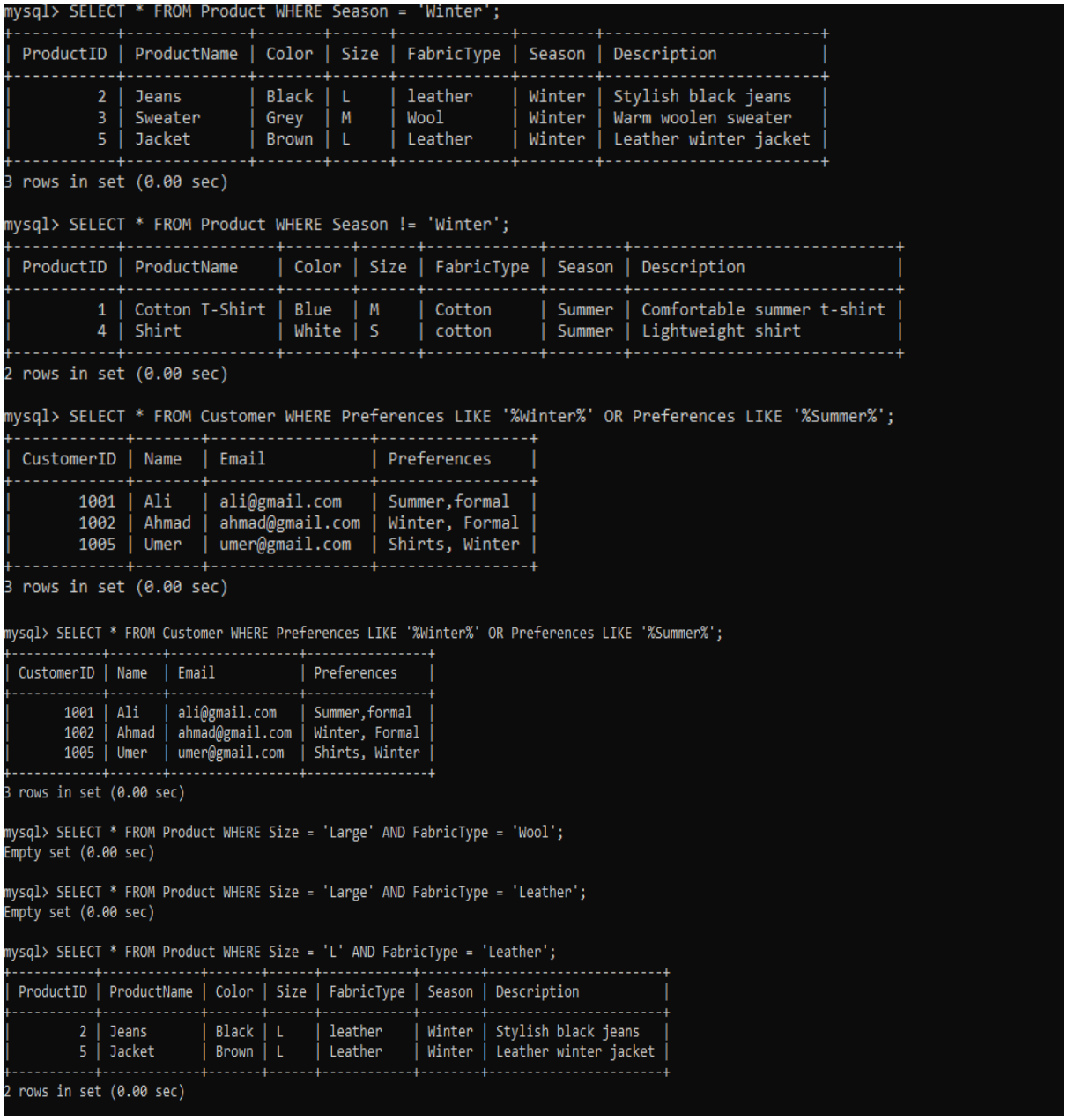
6.1 Display the use of Select Query

```
mysql> SELECT * FROM Inventory WHERE StoreLocation = 'Lahore';
+-------------+-----------+---------------+----------+
| InventoryID | ProductID | StoreLocation | Quantity |
+-------------+-----------+---------------+----------+
|         101 |         1 | Lahore        |       20 |
+-------------+-----------+---------------+----------+
1 row in set (0.05 sec)

mysql> SELECT * FROM product WHERE Color = 'Blue';
+-----------+----------------+-------+------+------------+--------+---------------------------+
| ProductID | ProductName    | Color | Size | FabricType | Season | Description               |
+-----------+----------------+-------+------+------------+--------+---------------------------+
|         1 | Cotton T-Shirt | Blue  | M    | Cotton     | Summer | Comfortable summer t-shirt |
+-----------+----------------+-------+------+------------+--------+---------------------------+
1 row in set (0.00 sec)

mysql> SELECT * FROM customer WHERE Name = 'Waleed';
+------------+--------+------------------+----------------+
| CustomerID | Name   | Email            | Preferences    |
+------------+--------+------------------+----------------+
|       1004 | Waleed | waleed@gmail.com | Jeans, Jackets |
+------------+--------+------------------+----------------+
1 row in set (0.00 sec)
```

```
mysql> SELECT * FROM Customer WHERE Name LIKE 'A%';
+------------+-------+-----------------+----------------+
| CustomerID | Name  | Email           | Preferences    |
+------------+-------+-----------------+----------------+
|       1001 | Ali   | ali@gmail.com   | Summer,formal  |
|       1002 | Ahmad | ahmad@gmail.com | Winter, Formal |
+------------+-------+-----------------+----------------+
2 rows in set (0.00 sec)

mysql> SELECT * FROM Customer WHERE Name LIKE '-A%';
Empty set (0.00 sec)

mysql> SELECT * FROM Customer WHERE Name LIKE '-a%';
Empty set (0.00 sec)

mysql> SELECT * FROM Customer WHERE Name LIKE '-%a';
Empty set (0.00 sec)

mysql> SELECT * FROM Customer WHERE Name LIKE 'W%';
+------------+--------+------------------+----------------+
| CustomerID | Name   | Email            | Preferences    |
+------------+--------+------------------+----------------+
|       1004 | Waleed | waleed@gmail.com | Jeans, Jackets |
+------------+--------+------------------+----------------+
1 row in set (0.00 sec)
```

6.2 Display the use of Where Query

In first two images we use AND, OR, Not operators.



```
mysql> SELECT * FROM Product WHERE Season = 'Winter';
+-----------+-------------+-------+------+------------+--------+----------------------+
| ProductID | ProductName | Color | Size | FabricType | Season | Description          |
+-----------+-------------+-------+------+------------+--------+----------------------+
|         2 | Jeans       | Black | L    | leather    | Winter | Stylish black jeans  |
|         3 | Sweater     | Grey  | M    | Wool       | Winter | Warm woolen sweater  |
|         5 | Jacket      | Brown | L    | Leather    | Winter | Leather winter jacket |
+-----------+-------------+-------+------+------------+--------+----------------------+
3 rows in set (0.00 sec)

mysql> SELECT * FROM Product WHERE Season != 'Winter';
+-----------+----------------+-------+------+------------+--------+---------------------------+
| ProductID | ProductName    | Color | Size | FabricType | Season | Description               |
+-----------+----------------+-------+------+------------+--------+---------------------------+
|         1 | Cotton T-Shirt | Blue  | M    | Cotton     | Summer | Comfortable summer t-shirt |
|         4 | Shirt          | White | S    | cotton     | Summer | Lightweight shirt         |
+-----------+----------------+-------+------+------------+--------+---------------------------+
2 rows in set (0.00 sec)

mysql> SELECT * FROM Customer WHERE Preferences LIKE '%Winter%' OR Preferences LIKE '%Summer%';
+------------+-------+-----------------+----------------+
| CustomerID | Name  | Email           | Preferences    |
+------------+-------+-----------------+----------------+
|       1001 | Ali   | ali@gmail.com   | Summer,formal  |
|       1002 | Ahmad | ahmad@gmail.com | Winter, Formal |
|       1005 | Umer  | umer@gmail.com  | Shirts, Winter |
+------------+-------+-----------------+----------------+
3 rows in set (0.00 sec)

mysql> SELECT * FROM Customer WHERE Preferences LIKE '%Winter%' OR Preferences LIKE '%Summer%';
+------------+-------+-----------------+----------------+
| CustomerID | Name  | Email           | Preferences    |
+------------+-------+-----------------+----------------+
|       1001 | Ali   | ali@gmail.com   | Summer,formal  |
|       1002 | Ahmad | ahmad@gmail.com | Winter, Formal |
|       1005 | Umer  | umer@gmail.com  | Shirts, Winter |
+------------+-------+-----------------+----------------+
3 rows in set (0.00 sec)

mysql> SELECT * FROM Product WHERE Size = 'Large' AND FabricType = 'Wool';
Empty set (0.00 sec)

mysql> SELECT * FROM Product WHERE Size = 'Large' AND FabricType = 'Leather';
Empty set (0.00 sec)

mysql> SELECT * FROM Product WHERE Size = 'L' AND FabricType = 'Leather';
+-----------+-------------+-------+------+------------+--------+----------------------+
| ProductID | ProductName | Color | Size | FabricType | Season | Description          |
+-----------+-------------+-------+------+------------+--------+----------------------+
|         2 | Jeans       | Black | L    | leather    | Winter | Stylish black jeans  |
|         5 | Jacket      | Brown | L    | Leather    | Winter | Leather winter jacket |
+-----------+-------------+-------+------+------------+--------+----------------------+
2 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM OrderItem WHERE Price > 1000;
+-------------+---------+-----------+----------+-------+
| OrderItemID | OrderID | ProductID | Quantity | Price |
+-------------+---------+-----------+----------+-------+
|         302 |     202 |         2 |        1 |  3500 |
|         303 |     203 |         4 |        3 |  1500 |
|         304 |     204 |         5 |        1 |  4500 |
|         305 |     205 |         3 |        2 |  1400 |
+-------------+---------+-----------+----------+-------+
4 rows in set (0.03 sec)

mysql> SELECT * FROM OrderItem WHERE Price > 3000;
+-------------+---------+-----------+----------+-------+
| OrderItemID | OrderID | ProductID | Quantity | Price |
+-------------+---------+-----------+----------+-------+
|         302 |     202 |         2 |        1 |  3500 |
|         304 |     204 |         5 |        1 |  4500 |
+-------------+---------+-----------+----------+-------+
2 rows in set (0.00 sec)

mysql> SELECT * FROM ReturnTable WHERE Reason LIKE '%damaged%';
ERROR 1146 (42S02): Table 'clothing_store_chain.returntable' doesn't exist
mysql> SELECT * FROM productreturn WHERE Reason LIKE '%damaged%';
Empty set (0.00 sec)

mysql> SELECT * FROM productreturn WHERE Reason LIKE '%size mismatch%';
Empty set (0.00 sec)

mysql> SELECT * FROM productreturn WHERE Reason LIKE '%Wrong size%';
+----------+-------------+------------+------------+
| ReturnID | OrderItemID | ReturnDate | Reason     |
+----------+-------------+------------+------------+
|      401 |         301 | 2025-05-06 | Wrong size |
+----------+-------------+------------+------------+
1 row in set (0.00 sec)

mysql> SELECT * FROM Inventory WHERE Quantity < 50;
+-------------+-----------+---------------+----------+
| InventoryID | ProductID | StoreLocation | Quantity |
+-------------+-----------+---------------+----------+
|         101 |         1 | Lahore        |       20 |
|         102 |         2 | Karachi       |       15 |
|         103 |         3 | Islamabad     |       10 |
|         104 |         4 | Multan        |       25 |
|         105 |         5 | Peshawar      |       12 |
+-------------+-----------+---------------+----------+
5 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM Inventory WHERE Quantity < 10;
Empty set (0.00 sec)

mysql> SELECT * FROM Inventory WHERE Quantity < 20;
+-------------+-----------+---------------+----------+
| InventoryID | ProductID | StoreLocation | Quantity |
+-------------+-----------+---------------+----------+
|         102 |         2 | Karachi       |       15 |
|         103 |         3 | Islamabad     |       10 |
|         105 |         5 | Peshawar      |       12 |
+-------------+-----------+---------------+----------+
3 rows in set (0.00 sec)

mysql> SELECT * FROM Orders WHERE OrderDate > '2025-05-01';
+---------+------------+------------+------------+
| OrderID | CustomerID | OrderDate  | TotalPrice |
+---------+------------+------------+------------+
|     202 |       1002 | 2025-05-02 |       3500 |
|     203 |       1003 | 2025-05-03 |       1500 |
|     204 |       1004 | 2025-05-04 |       4500 |
|     205 |       1005 | 2025-05-05 |       2800 |
+---------+------------+------------+------------+
4 rows in set (0.03 sec)

mysql> SELECT * FROM Orders WHERE OrderDate > '2025-05-05';
Empty set (0.00 sec)

mysql>
```

6.3 Display the use of AS Query

```
mysql> SELECT ProductName AS Name, FabricType AS Material FROM Product;
+---------------+----------+
| Name          | Material |
+---------------+----------+
| Cotton T-Shirt | Cotton   |
| Jeans         | leather  |
| Sweater       | Wool     |
| Shirt         | cotton   |
| Jacket        | Leather  |
+---------------+----------+
5 rows in set (0.00 sec)

mysql> SELECT ProductName AS Name FROM Product WHERE Season = 'Winter';
+---------+
| Name    |
+---------+
| Jeans   |
| Sweater |
| Jacket  |
+---------+
3 rows in set (0.00 sec)

mysql> SELECT ProductID, Quantity * Price AS TotalItemCost FROM OrderItem;
+-----------+---------------+
| ProductID | TotalItemCost |
+-----------+---------------+
|         1 |          2000 |
|         2 |          3500 |
|         4 |          4500 |
|         5 |          4500 |
|         3 |          2800 |
+-----------+---------------+
5 rows in set (0.00 sec)
```

6.4 Display the use of Distinct Query

```
mysql> SELECT DISTINCT FabricType FROM Product;
+------------+
| FabricType |
+------------+
| Cotton     |
| leather    |
| Wool       |
+------------+
3 rows in set (0.02 sec)

mysql> SELECT DISTINCT Season, Size FROM Product;
+--------+------+
| Season | Size |
+--------+------+
| Summer | M    |
| Winter | L    |
| Winter | M    |
| Summer | S    |
+--------+------+
4 rows in set (0.00 sec)

mysql> SELECT DISTINCT Size FROM Product;
+------+
| Size |
+------+
| M    |
| L    |
| S    |
+------+
3 rows in set (0.00 sec)
```

6.5 Display the use of Order BY Query

```
mysql> SELECT * FROM Product ORDER BY ProductName ASC;
+-----------+---------------+-------+------+------------+--------+---------------------------+
| ProductID | ProductName   | Color | Size | FabricType | Season | Description               |
+-----------+---------------+-------+------+------------+--------+---------------------------+
|         1 | Cotton T-Shirt | Blue  | M    | Cotton     | Summer | Comfortable summer t-shirt |
|         5 | Jacket        | Brown | L    | Leather    | Winter | Leather winter jacket     |
|         2 | Jeans         | Black | L    | leather    | Winter | Stylish black jeans       |
|         4 | Shirt         | White | S    | cotton     | Summer | Lightweight shirt         |
|         3 | Sweater       | Grey  | M    | Wool       | Winter | Warm woolen sweater       |
+-----------+---------------+-------+------+------------+--------+---------------------------+
5 rows in set (0.05 sec)

mysql> SELECT * FROM OrderItem ORDER BY Quantity DESC, Price ASC;
+-------------+---------+-----------+----------+-------+
| OrderItemID | OrderID | ProductID | Quantity | Price |
+-------------+---------+-----------+----------+-------+
|         303 |     203 |         4 |        3 |  1500 |
|         301 |     201 |         1 |        2 |  1000 |
|         305 |     205 |         3 |        2 |  1400 |
|         302 |     202 |         2 |        1 |  3500 |
|         304 |     204 |         5 |        1 |  4500 |
+-------------+---------+-----------+----------+-------+
5 rows in set (0.00 sec)

mysql> SELECT * FROM Inventory ORDER BY Quantity ASC;
+-------------+-----------+---------------+----------+
| InventoryID | ProductID | StoreLocation | Quantity |
+-------------+-----------+---------------+----------+
|         103 |         3 | Islamabad     |       10 |
|         105 |         5 | Peshawar      |       12 |
|         102 |         2 | Karachi       |       15 |
|         101 |         1 | Lahore        |       20 |
|         104 |         4 | Multan        |       25 |
+-------------+-----------+---------------+----------+
5 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM Product ORDER BY Season DESC;
+-----------+---------------+-------+------+------------+--------+---------------------------+
| ProductID | ProductName   | Color | Size | FabricType | Season | Description               |
+-----------+---------------+-------+------+------------+--------+---------------------------+
|         2 | Jeans         | Black | L    | leather    | Winter | Stylish black jeans       |
|         3 | Sweater       | Grey  | M    | Wool       | Winter | Warm woolen sweater       |
|         5 | Jacket        | Brown | L    | Leather    | Winter | Leather winter jacket     |
|         1 | Cotton T-Shirt| Blue  | M    | Cotton     | Summer | Comfortable summer t-shirt|
|         4 | Shirt         | White | S    | cotton     | Summer | Lightweight shirt         |
+-----------+---------------+-------+------+------------+--------+---------------------------+
5 rows in set (0.00 sec)

mysql> SELECT * FROM Product ORDER BY Size ASC, ProductName ASC;
+-----------+---------------+-------+------+------------+--------+---------------------------+
| ProductID | ProductName   | Color | Size | FabricType | Season | Description               |
+-----------+---------------+-------+------+------------+--------+---------------------------+
|         5 | Jacket        | Brown | L    | Leather    | Winter | Leather winter jacket     |
|         2 | Jeans         | Black | L    | leather    | Winter | Stylish black jeans       |
|         1 | Cotton T-Shirt| Blue  | M    | Cotton     | Summer | Comfortable summer t-shirt|
|         3 | Sweater       | Grey  | M    | Wool       | Winter | Warm woolen sweater       |
|         4 | Shirt         | White | S    | cotton     | Summer | Lightweight shirt         |
+-----------+---------------+-------+------+------------+--------+---------------------------+
5 rows in set (0.00 sec)

mysql>
```

6.6 Display the use of Arithmetic Query

```
mysql> SELECT OrderItemID, Quantity + Price AS Total FROM OrderItem;
+-------------+-------+
| OrderItemID | Total |
+-------------+-------+
|         301 |  1002 |
|         302 |  3501 |
|         303 |  1503 |
|         304 |  4501 |
|         305 |  1402 |
+-------------+-------+
5 rows in set (0.00 sec)

mysql> SELECT OrderItemID, Price - Quantity AS Difference FROM OrderItem;
+-------------+------------+
| OrderItemID | Difference |
+-------------+------------+
|         301 |        998 |
|         302 |       3499 |
|         303 |       1497 |
|         304 |       4499 |
|         305 |       1398 |
+-------------+------------+
5 rows in set (0.09 sec)

mysql> SELECT OrderItemID, Quantity * Price AS TotalCost FROM OrderItem;
+-------------+-----------+
| OrderItemID | TotalCost |
+-------------+-----------+
|         301 |      2000 |
|         302 |      3500 |
|         303 |      4500 |
|         304 |      4500 |
|         305 |      2800 |
+-------------+-----------+
5 rows in set (0.00 sec)

mysql> SELECT OrderItemID, Price / Quantity AS UnitPrice FROM OrderItem
    -> WHERE Quantity != 0;
+-------------+-----------+
| OrderItemID | UnitPrice |
+-------------+-----------+
|         301 |  500.0000 |
|         302 | 3500.0000 |
|         303 |  500.0000 |
|         304 | 4500.0000 |
|         305 |  700.0000 |
+-------------+-----------+
5 rows in set (0.00 sec)

mysql>
```

6.7 Display the use of Relational Query

```
mysql> SELECT * FROM OrderItem WHERE Quantity = 5;
Empty set (0.00 sec)

mysql> SELECT * FROM OrderItem WHERE Quantity = 12;
Empty set (0.00 sec)

mysql> SELECT * FROM OrderItem WHERE Quantity = 15;
Empty set (0.00 sec)

mysql> SELECT * FROM OrderItem WHERE Quantity = 25;
Empty set (0.00 sec)

mysql> SELECT * FROM inventory WHERE Quantity = 25;
+-------------+-----------+---------------+----------+
| InventoryID | ProductID | StoreLocation | Quantity |
+-------------+-----------+---------------+----------+
|         104 |         4 | Multan        |       25 |
+-------------+-----------+---------------+----------+
1 row in set (0.00 sec)

mysql> SELECT * FROM inventory WHERE Quantity = 15;
+-------------+-----------+---------------+----------+
| InventoryID | ProductID | StoreLocation | Quantity |
+-------------+-----------+---------------+----------+
|         102 |         2 | Karachi       |       15 |
+-------------+-----------+---------------+----------+
1 row in set (0.00 sec)

mysql> SELECT * FROM inventory WHERE Quantity = 15;
+-------------+-----------+---------------+----------+
| InventoryID | ProductID | StoreLocation | Quantity |
+-------------+-----------+---------------+----------+
|         102 |         2 | Karachi       |       15 |
+-------------+-----------+---------------+----------+
1 row in set (0.00 sec)

mysql> SELECT * FROM OrderItem WHERE Price <= 500;
Empty set (0.00 sec)

mysql> SELECT * FROM OrderItem WHERE Price <= 1500;
+-------------+---------+-----------+----------+-------+
| OrderItemID | OrderID | ProductID | Quantity | Price |
+-------------+---------+-----------+----------+-------+
|         301 |     201 |         1 |        2 |  1000 |
|         303 |     203 |         4 |        3 |  1500 |
|         305 |     205 |         3 |        2 |  1400 |
+-------------+---------+-----------+----------+-------+
3 rows in set (0.00 sec)

mysql> SELECT * FROM OrderItem WHERE Price > 1000;
+-------------+---------+-----------+----------+-------+
| OrderItemID | OrderID | ProductID | Quantity | Price |
+-------------+---------+-----------+----------+-------+
|         302 |     202 |         2 |        1 |  3500 |
|         303 |     203 |         4 |        3 |  1500 |
|         304 |     204 |         5 |        1 |  4500 |
|         305 |     205 |         3 |        2 |  1400 |
+-------------+---------+-----------+----------+-------+
4 rows in set (0.00 sec)
```

6.8 Display the use of Between, And, In and Like Queries

```
mysql> SELECT * FROM OrderItem WHERE Price BETWEEN 500 AND 1500;
+-------------+---------+-----------+----------+-------+
| OrderItemID | OrderID | ProductID | Quantity | Price |
+-------------+---------+-----------+----------+-------+
|         301 |     201 |         1 |        2 |  1000 |
|         303 |     203 |         4 |        3 |  1500 |
|         305 |     205 |         3 |        2 |  1400 |
+-------------+---------+-----------+----------+-------+
3 rows in set (0.05 sec)

mysql> SELECT * FROM Inventory WHERE Quantity BETWEEN 10 AND 30;
+-------------+-----------+---------------+----------+
| InventoryID | ProductID | StoreLocation | Quantity |
+-------------+-----------+---------------+----------+
|         101 |         1 | Lahore        |       20 |
|         102 |         2 | Karachi       |       15 |
|         103 |         3 | Islamabad     |       10 |
|         104 |         4 | Multan        |       25 |
|         105 |         5 | Peshawar      |       12 |
+-------------+-----------+---------------+----------+
5 rows in set (0.00 sec)

mysql> SELECT * FROM OrderItem WHERE Quantity > 2 AND Price < 1000;
Empty set (0.00 sec)

mysql> SELECT * FROM invetory WHERE Quantity > 2 AND Price < 1000;
ERROR 1146 (42S02): Table 'clothing_store_chain.invetory' doesn't exist
mysql> SELECT * FROM inventory WHERE Quantity > 2 AND Price < 1000;
ERROR 1054 (42S22): Unknown column 'Price' in 'where clause'
mysql> SELECT * FROM inventory WHERE Quantity > 2 AND ProductID > 3;
+-------------+-----------+---------------+----------+
| InventoryID | ProductID | StoreLocation | Quantity |
+-------------+-----------+---------------+----------+
|         104 |         4 | Multan        |       25 |
|         105 |         5 | Peshawar      |       12 |
+-------------+-----------+---------------+----------+
2 rows in set (0.06 sec)

mysql> SELECT * FROM Product WHERE Season = 'Summer' AND Color = 'Red';
Empty set (0.00 sec)

mysql> SELECT * FROM Product WHERE Season = 'Summer' AND Color = 'Blue';
+-----------+---------------+-------+------+------------+--------+---------------------------+
| ProductID | ProductName   | Color | Size | FabricType | Season | Description               |
+-----------+---------------+-------+------+------------+--------+---------------------------+
|         1 | Cotton T-Shirt | Blue  | M    | Cotton     | Summer | Comfortable summer t-shirt |
+-----------+---------------+-------+------+------------+--------+---------------------------+
1 row in set (0.00 sec)

mysql> SELECT * FROM Customer WHERE Email LIKE '%@gmail.com';
+------------+--------+------------------+------------------+
| CustomerID | Name   | Email            | Preferences      |
+------------+--------+------------------+------------------+
|       1001 | Ali    | ali@gmail.com    | Summer,formal    |
|       1002 | Ahmad  | ahmad@gmail.com  | Winter, Formal   |
|       1003 | Usman  | usman@gmail.com  | Casual, T-Shirts |
|       1004 | Waleed | waleed@gmail.com | Jeans, Jackets   |
|       1005 | Umer   | umer@gmail.com   | Shirts, Winter   |
+------------+--------+------------------+------------------+
5 rows in set (0.00 sec)

mysql> SELECT * FROM Product WHERE ProductName LIKE 'S%';
+-----------+-------------+-------+------+------------+--------+---------------------+
| ProductID | ProductName | Color | Size | FabricType | Season | Description         |
+-----------+-------------+-------+------+------------+--------+---------------------+
|         3 | Sweater     | Grey  | M    | Wool       | Winter | Warm woolen sweater |
|         4 | Shirt       | White | S    | cotton     | Summer | Lightweight shirt   |
+-----------+-------------+-------+------+------------+--------+---------------------+
2 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM ReviewFlag WHERE ReturnID IS NULL;
Empty set (0.00 sec)

mysql> SELECT * FROM ReturnTable WHERE Reason IS NULL;
ERROR 1146 (42S02): Table 'clothing_store_chain.returntable' doesn't exist
mysql> SELECT * FROM productreturn WHERE Reason IS NULL;
Empty set (0.00 sec)

mysql> SELECT * FROM Customer WHERE Preferences IS NOT NULL;
+------------+--------+-----------------+-----------------+
| CustomerID | Name   | Email           | Preferences     |
+------------+--------+-----------------+-----------------+
|       1001 | Ali    | ali@gmail.com   | Summer,formal   |
|       1002 | Ahmad  | ahmad@gmail.com | Winter, Formal  |
|       1003 | Usman  | usman@gmail.com | Casual, T-Shirts |
|       1004 | Waleed | waleed@gmail.com | Jeans, Jackets  |
|       1005 | Umer   | umer@gmail.com  | Shirts, Winter  |
+------------+--------+-----------------+-----------------+
5 rows in set (0.00 sec)

mysql> SELECT * FROM productreturn WHERE ReturnDate IS NOT NULL;
+----------+-------------+------------+---------------+
| ReturnID | OrderItemID | ReturnDate | Reason        |
+----------+-------------+------------+---------------+
|      401 |         301 | 2025-05-06 | Wrong size    |
|      402 |         302 | 2025-05-07 | Defective item |
|      403 |         303 | 2025-05-08 | Color mismatch |
|      404 |         304 | 2025-05-09 | Not satisfied  |
|      405 |         305 | 2025-05-10 | Late delivery  |
+----------+-------------+------------+---------------+
5 rows in set (0.00 sec)

mysql>
```

6.9 Display the use of Function Aggregate Queries

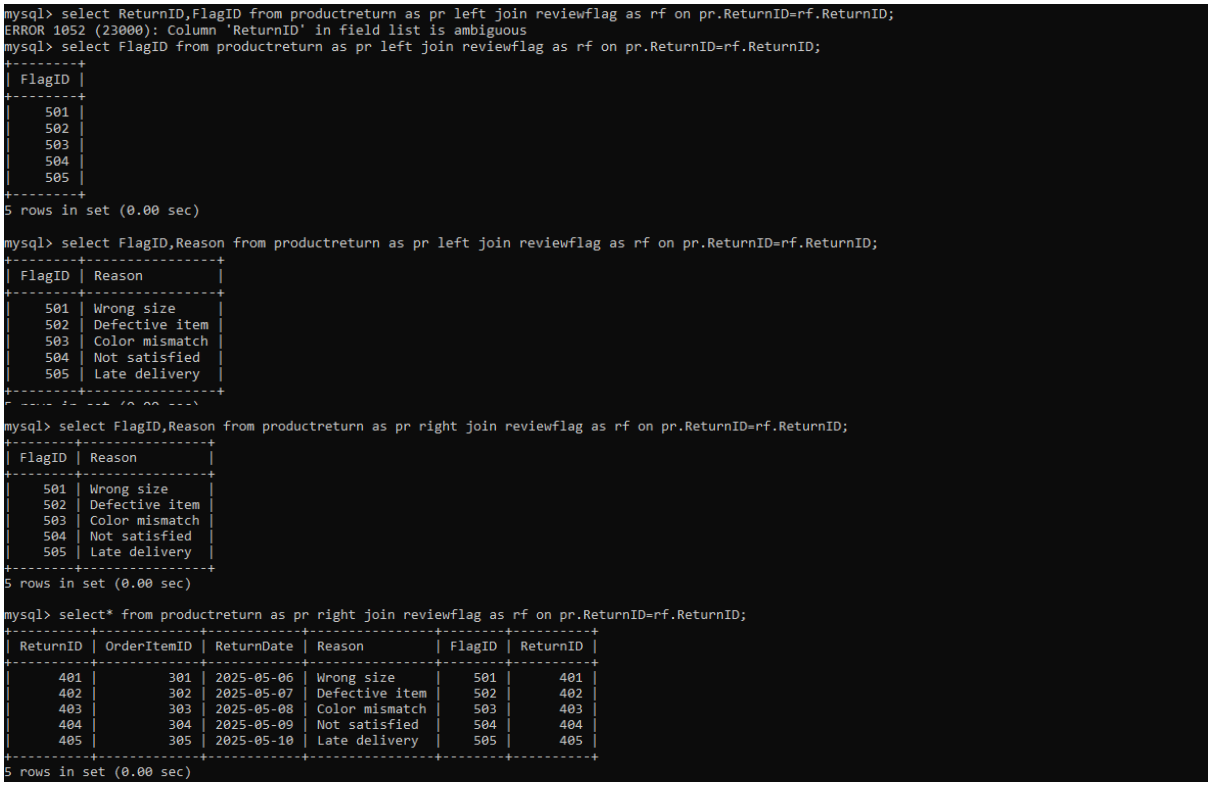
```
mysql> SELECT COUNT(*) AS TotalCustomers FROM Customer;
+----------------+
| TotalCustomers |
+----------------+
|              5 |
+----------------+
1 row in set (0.03 sec)

mysql> SELECT COUNT(ProductID) AS TotalProducts FROM Product;
+---------------+
| TotalProducts |
+---------------+
|             5 |
+---------------+
1 row in set (0.00 sec)

mysql> SELECT SUM(Price) AS TotalOrderAmount FROM OrderItem;
+------------------+
| TotalOrderAmount |
+------------------+
|            11900 |
+------------------+
1 row in set (0.00 sec)

mysql> SELECT SUM(Quantity) AS TotalInventory FROM Inventory;
+----------------+
| TotalInventory |
+----------------+
|             82 |
+----------------+
1 row in set (0.00 sec)
```

```
mysql> SELECT AVG(Price) AS AveragePrice FROM OrderItem;
+--------------+
| AveragePrice |
+--------------+
|    2380.0000 |
+--------------+
1 row in set (0.00 sec)

mysql> SELECT AVG(Quantity) AS AverageInventory FROM Inventory;
+------------------+
| AverageInventory |
+------------------+
|          16.4000 |
+------------------+
1 row in set (0.00 sec)
```

```
mysql> SELECT MIN(Price) AS LowestPrice FROM OrderItem;
+-------------+
| LowestPrice |
+-------------+
|        1000 |
+-------------+
1 row in set (0.03 sec)

mysql> SELECT MIN(OrderDate) AS FirstOrderDate FROM `Order`;
ERROR 1146 (42S02): Table 'clothing_store_chain.order' doesn't exist
mysql> SELECT MIN(OrderDate) AS FirstOrderDate FROM Orders
    -> SELECT MIN(OrderDate) AS FirstOrderDate FROM Orders;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'SELECT MIN(Or
derDate) AS FirstOrderDate FROM Orders' at line 2
mysql> SELECT MIN(OrderDate) AS FirstOrderDate FROM Orders;
+----------------+
| FirstOrderDate |
+----------------+
| 2025-05-01     |
+----------------+
1 row in set (0.00 sec)

mysql> SELECT MAX(Price) AS HighestPrice FROM OrderItem;
+-------------+
| HighestPrice |
+-------------+
|        4500 |
+-------------+
1 row in set (0.00 sec)

mysql> SELECT MAX(OrderDate) AS LatestOrderDate FROM `Order`;
ERROR 1146 (42S02): Table 'clothing_store_chain.order' doesn't exist
mysql> SELECT MAX(OrderDate) AS LatestOrderDate FROM Orders;
+-----------------+
| LatestOrderDate |
+-----------------+
| 2025-05-05      |
+-----------------+
1 row in set (0.00 sec)
```

6.10 Display the use of Join Queries

```
mysql> select* from productreturn as pr inner join reviewflag as rf on pr.ReturnID=rf.ReturnID;
+----------+-------------+------------+----------------+--------+----------+
| ReturnID | OrderItemID | ReturnDate | Reason         | FlagID | ReturnID |
+----------+-------------+------------+----------------+--------+----------+
|      401 |         301 | 2025-05-06 | Wrong size     |    501 |      401 |
|      402 |         302 | 2025-05-07 | Defective item |    502 |      402 |
|      403 |         303 | 2025-05-08 | Color mismatch |    503 |      403 |
|      404 |         304 | 2025-05-09 | Not satisfied  |    504 |      404 |
|      405 |         305 | 2025-05-10 | Late delivery  |    505 |      405 |
+----------+-------------+------------+----------------+--------+----------+
5 rows in set (0.00 sec)

mysql> select* from productreturn as pr left join reviewflag as rf on pr.ReturnID=rf.ReturnID;
+----------+-------------+------------+----------------+--------+----------+
| ReturnID | OrderItemID | ReturnDate | Reason         | FlagID | ReturnID |
+----------+-------------+------------+----------------+--------+----------+
|      401 |         301 | 2025-05-06 | Wrong size     |    501 |      401 |
|      402 |         302 | 2025-05-07 | Defective item |    502 |      402 |
|      403 |         303 | 2025-05-08 | Color mismatch |    503 |      403 |
|      404 |         304 | 2025-05-09 | Not satisfied  |    504 |      404 |
|      405 |         305 | 2025-05-10 | Late delivery  |    505 |      405 |
+----------+-------------+------------+----------------+--------+----------+
5 rows in set (0.00 sec)
```



```
mysql> select ReturnID,FlagID from productreturn as pr left join reviewflag as rf on pr.ReturnID=rf.ReturnID;
ERROR 1052 (23000): Column 'ReturnID' in field list is ambiguous
mysql> select FlagID from productreturn as pr left join reviewflag as rf on pr.ReturnID=rf.ReturnID;
+--------+
| FlagID |
+--------+
|    501 |
|    502 |
|    503 |
|    504 |
|    505 |
+--------+
5 rows in set (0.00 sec)

mysql> select FlagID,Reason from productreturn as pr left join reviewflag as rf on pr.ReturnID=rf.ReturnID;
+--------+----------------+
| FlagID | Reason         |
+--------+----------------+
|    501 | Wrong size     |
|    502 | Defective item |
|    503 | Color mismatch |
|    504 | Not satisfied  |
|    505 | Late delivery  |
+--------+----------------+

mysql> select FlagID,Reason from productreturn as pr right join reviewflag as rf on pr.ReturnID=rf.ReturnID;
+--------+----------------+
| FlagID | Reason         |
+--------+----------------+
|    501 | Wrong size     |
|    502 | Defective item |
|    503 | Color mismatch |
|    504 | Not satisfied  |
|    505 | Late delivery  |
+--------+----------------+
5 rows in set (0.00 sec)

mysql> select* from productreturn as pr right join reviewflag as rf on pr.ReturnID=rf.ReturnID;
+----------+-------------+------------+----------------+--------+----------+
| ReturnID | OrderItemID | ReturnDate | Reason         | FlagID | ReturnID |
+----------+-------------+------------+----------------+--------+----------+
|      401 |         301 | 2025-05-06 | Wrong size     |    501 |      401 |
|      402 |         302 | 2025-05-07 | Defective item |    502 |      402 |
|      403 |         303 | 2025-05-08 | Color mismatch |    503 |      403 |
|      404 |         304 | 2025-05-09 | Not satisfied  |    504 |      404 |
|      405 |         305 | 2025-05-10 | Late delivery  |    505 |      405 |
+----------+-------------+------------+----------------+--------+----------+
5 rows in set (0.00 sec)
```

7-Conclusion

7.1 Summary of the Project

This project aimed to design and normalize a database for a Clothing Store Chain that handles seasonal product lines and return policies. The Entity Relationship Diagram (ERD) helped visualize the relationships between entities such as Product, Inventory, Customer, Order, and Return. Through normalization, we ensured the database is optimized by eliminating redundancy and improving data integrity. The final schema supports core business operations like managing product stock, placing orders, handling returns, and reviewing return flags.