Retail Store- SQL Assessment Use Case

Objective

You're the Database Developer for a small retail store (brick-and-mortar + online). The goal is to assess SQL skills in creating a structured database schema with proper keys and constraints, and inserting accurate initial data based on given requirements.

Instructions

Step 1: Create Database and Tables

Create a database named `RetailDB` and the following tables:

1. Customers
   - CustomerID (INT, AUTO_INCREMENT, PK)
   - FullName (VARCHAR(100), NOT NULL)
   - Email (VARCHAR(120), UNIQUE)
   - Phone (VARCHAR(20))
   - JoinDate (DATE, NOT NULL)

2. Products
   - ProductID (INT, AUTO_INCREMENT, PK)
   - SKU (VARCHAR(30), UNIQUE, NOT NULL)
   - ProductName (VARCHAR(120), NOT NULL)
   - Category (VARCHAR(50))
   - UnitPrice (DECIMAL(10,2), NOT NULL)
   - Active (TINYINT(1) DEFAULT 1)

3. Stores
   - StoreID (INT, AUTO_INCREMENT, PK)
   - StoreName (VARCHAR(100), NOT NULL)
   - City (VARCHAR(60))
   - State (VARCHAR(60))

4. Inventory
   - InventoryID (INT, AUTO_INCREMENT, PK)
   - StoreID (INT, FK → Stores.StoreID)
   - ProductID (INT, FK → Products.ProductID)
   - QuantityOnHand (INT, NOT NULL)
   - LastRestockDate (DATE)
   - Constraint: (StoreID, ProductID) must be unique

5. Orders
  - OrderID (INT, AUTO_INCREMENT, PK)
  - CustomerID (INT, FK → Customers.CustomerID)
  - StoreID (INT, FK → Stores.StoreID)
  - OrderDate (DATETIME, NOT NULL)
  - Status (ENUM('Placed','Shipped','Delivered','Cancelled') DEFAULT 'Placed')
  - PaymentMethod (ENUM('Card','UPI','Cash','Wallet') NOT NULL)
  - OrderTotal (DECIMAL(12,2))

6. OrderItems
  - OrderItemID (INT, AUTO_INCREMENT, PK)
  - OrderID (INT, FK → Orders.OrderID)
  - ProductID (INT, FK → Products.ProductID)
  - Quantity (INT, NOT NULL)
  - UnitPriceAtSale (DECIMAL(10,2), NOT NULL)

Step 2: Insert Sample Data

Customers:
- ('Aisha Verma','aisha@example.com','+91-900000001','2024-07-10')
- ('Rahul Mehta','rahul@example.com','+91-900000002','2024-08-02')
- ('Priya Shah',NULL,NULL,'2024-09-15')

Products:
- ('SKU-TEE-001','Classic Tee','Apparel',499.00,1)
- ('SKU-MUG-101','Ceramic Mug','Home',299.00,1)
- ('SKU-NB-210','A5 Notebook','Stationery',149.00,1)
- ('SKU-BAG-505','Tote Bag','Apparel',699.00,0)

Stores:
- ('Downtown Outlet','Mumbai','MH')
- ('City Center','Pune','MH')

Inventory:
- (StoreID=1, ProductID=1, 40, '2025-06-01')
- (StoreID=1, ProductID=2, 25, '2025-06-05')
- (StoreID=2, ProductID=1, 10, '2025-05-28')
- (StoreID=2, ProductID=3, 60, '2025-06-10')

Orders:
- (CustomerID=1, StoreID=1, '2025-07-01 10:15:00','Delivered','Card',1247.00)
- (CustomerID=2, StoreID=2, '2025-07-03 16:22:00','Shipped','UPI',149.00)
- (CustomerID=1, StoreID=1, '2025-07-05 12:05:00','Cancelled','Wallet',299.00)

OrderItems:
- (OrderID=1, ProductID=1, 2, 499.00)
- (OrderID=1, ProductID=2, 1, 249.00)
- (OrderID=2, ProductID=3, 1, 149.00)
- (OrderID=3, ProductID=2, 1, 299.00)

Launching MySQL Console:

1. **To open the command terminal the test takers need to go to the Application menu**
2. **(Three horizontal lines at left top) -> Terminal -> New Terminal.**
3. **To login to mysql instance: Open new terminal and use following command:**

   a. sudo systemctl enable mysql
   b. sudo systemctl start mysql

NOTE: After typing any of the above commands you might encounter any warnings.
>> Please note that this warning is expected and can be disregarded. Proceed to the next step.

c. mysql -u root -p
The last command will ask for password which is 'pass@word1'

Running Test Cases:

1. Launch another instance of Terminal
2. Run command: **mvn test**
3. You can run this command any number of times to test the status of your activities.
4. Make sure before final submission, you must run test and then submit