MedicineInventory Database SQL Code Report

Supervisor:Dr.Mohammed Elsayeh

|  |  |
| --- | --- |
| Student Name | ID |
| Kerolus Magdy | 231001655 |
| Abdelrahman Omar | 222000159 |
| Yousef Tamer | 211001945 |
| Hanin Waleed | 231001812 |

**1. Purpose of the Database**

The MedicineInventory database is designed to manage a medical inventory system for a clinic or pharmacy. Its main objectives are:

* To store detailed information about doctors, including their qualifications and specializations.
* To maintain a comprehensive list of diseases, categorized by type.
* To track medicines, including price, dosage, manufacture and expiry dates.
* To record prescriptions issued to patients and link them with the prescribing doctor and prescribed medicines.
* To automatically calculate billing for prescriptions using triggers, including tax and discount calculations.
* To enable queries for reporting, such as finding top-selling medicines, listing doctors by specialty, and filtering diseases by type.

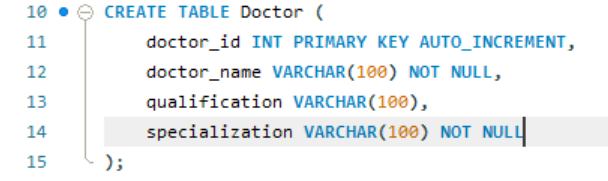
**2. Database Creation**

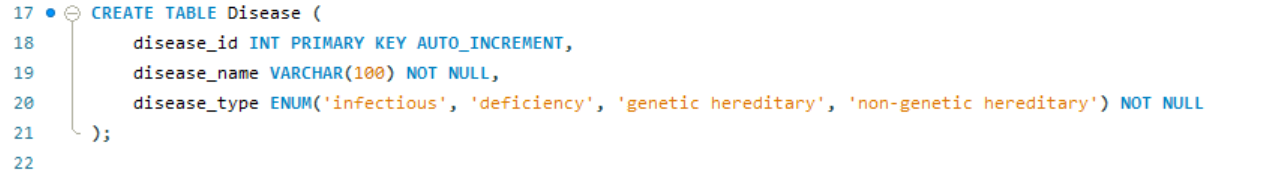
CREATE DATABASE MedicineInventory;

USE MedicineInventory;

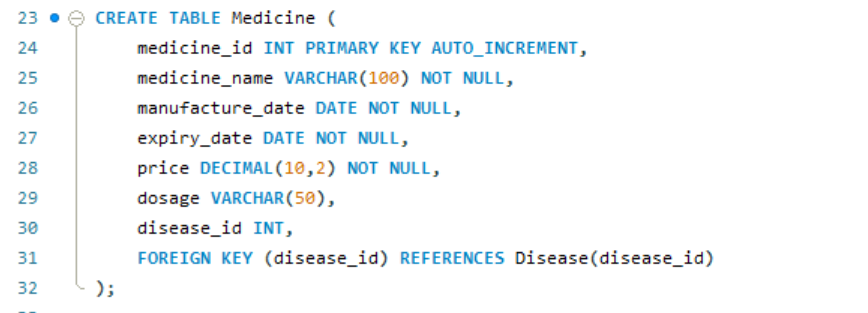
* **Action:** Creates a new database named MedicineInventory and sets it as the active database.
* **Purpose:** Provides a structured storage space for all medical records, ensuring data integrity and efficient access.

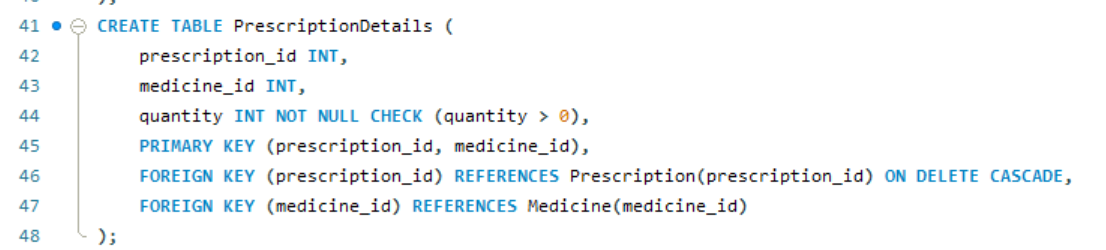
**3. Table Structures**

**a. Doctor Table**

* **Purpose:** Stores information about doctors. Each doctor has a unique ID, a name, qualifications, and a specialization.
* **Use case:** Allows linking prescriptions to specific doctors.  
  **b. Disease Table**
* **Purpose:** Catalogs diseases with type classification.
* **Use case:** Links medicines to the disease they treat.
* **c. Medicine Table**
* **Purpose:** Stores detailed medicine information, including pricing, dosage, and linked disease.
* **Use case:** Enables stock management and billing calculations.

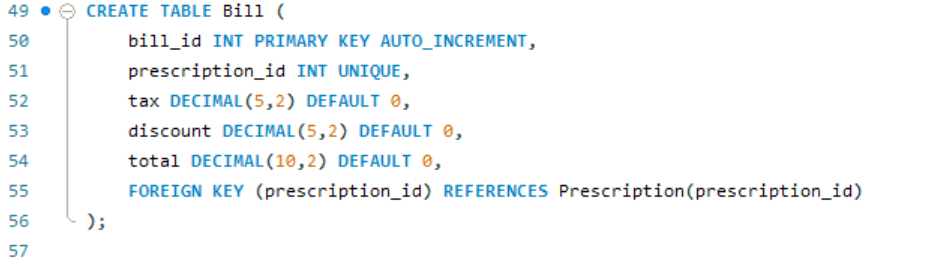
**d. Prescription Table**

* **Purpose:** Records prescriptions, linking patients and doctors.
* **Use case:** Base table for tracking prescription details and generating bills.

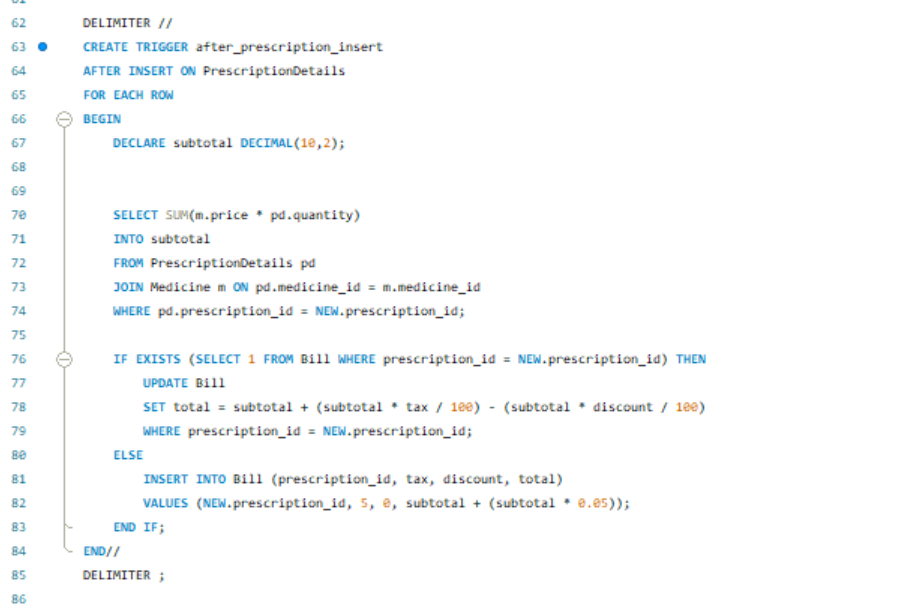
**e. PrescriptionDetails Table**

* **Purpose:** Links each prescription to its prescribed medicines and quantity.
* **Use case:** Supports billing calculations and medicine tracking.

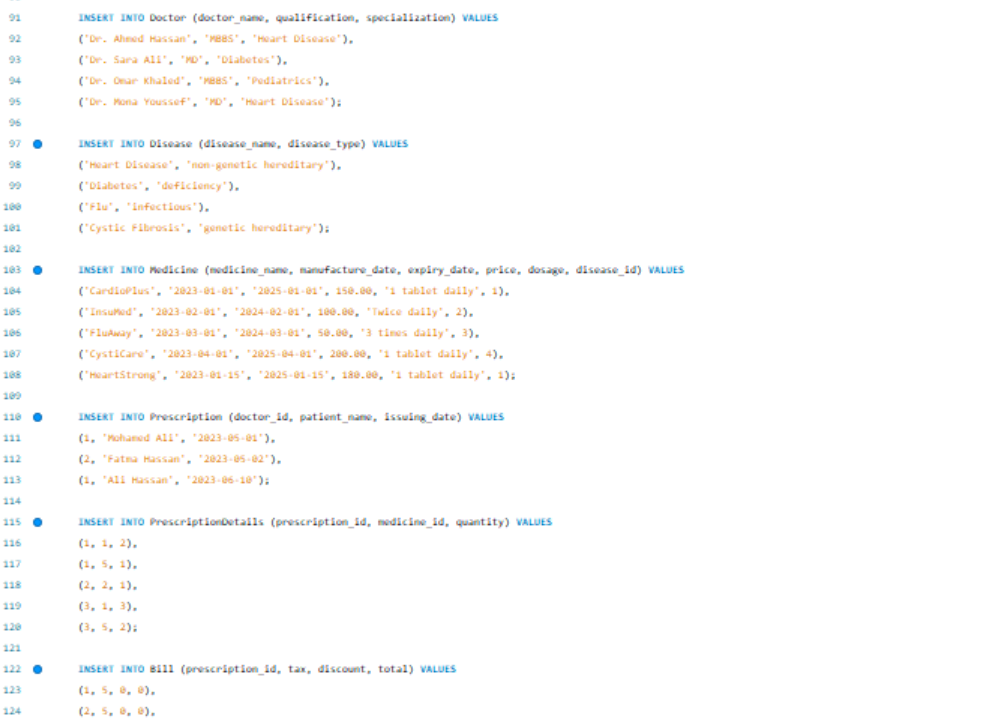
**f. Bill Table**

* **Purpose:** Tracks billing for each prescription, including tax and discount.
* **Use case:** Automated billing ensures accurate financial management.

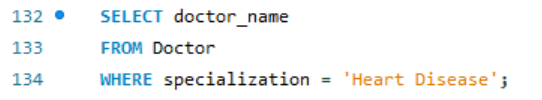
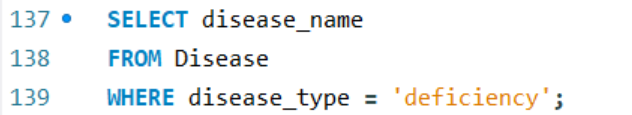
**4. Trigger for Automatic Bill Update**

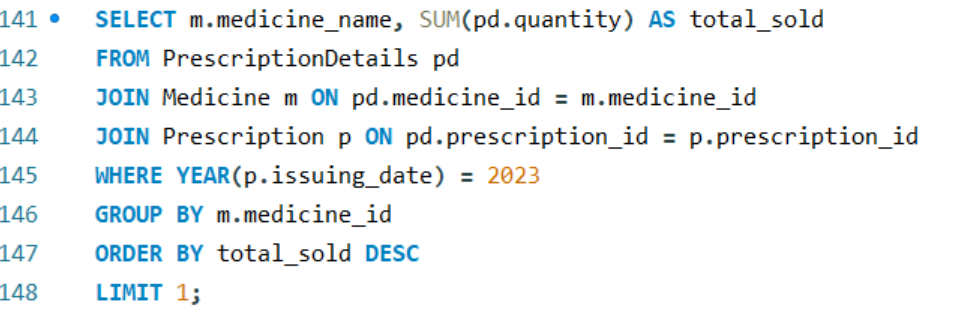
* **Purpose:** Automatically updates or creates a bill when new prescription details are added.
* **Use case:** Ensures billing is always accurate and reduces manual errors.

**5. Sample Data Insertion**

* Adds sample doctors, diseases, medicines, prescriptions, prescription details, and bills.
* Demonstrates relationships between tables and enables testing queries.
* 

**6. Example Queries**

* **List doctors specializing in Heart Disease:**
* **List deficiency diseases:**
* **Top-selling medicine of 2023:**



**7. Summary**

The MedicineInventory database provides a complete system for:

* Managing doctors, diseases, and medicines.
* Tracking prescriptions and linking them with doctors and medicines.
* Automatically generating accurate bills using triggers.
* Performing queries for analytics and reporting.
* This design ensures relational integrity, minimizes manual effort, and supports both medical and financial management tasks.

8. User interface (UI)

The user interface (UI) for the Medicine Inventory Management System is developed in Python using the Tkinter library and is directly connected to the MySQL database through the mysql.connector library. It provides an intuitive and organized layout that allows users to manage doctors, diseases, medicines, prescriptions, and bills without writing SQL queries. The interface includes dedicated forms for adding and editing records, drop-down menus for selecting related data (such as linking medicines to diseases), and searching fields for quickly finding information. A central dashboard offers quick navigation to different modules, while tables display stored records in a clear, sortable format. All actions performed in the UI—such as inserting, updating, or deleting data are instantly reflected in the MySQL database, ensuring real-time synchronization and accuracy.

A screenshot of a computer

AI-generated content may be incorrect.