# Database for the Pharmacy Management System

A Database for The Pharmacy Management System could theoretically look like the following:

## 1. Database Structure

The system consists of the following main tables, ensuring data integrity and normalization:  
• Persons: Stores shared attributes for doctors and patients.  
• Doctors: Stores doctor-specific information, linked to Persons.  
• Patients: Stores patient-specific information, linked to Persons.  
• Medications: Stores details about available drugs.  
• Prescriptions: Links doctors, patients, and medications.  
• MedicationStock: Tracks inventory of medications in the pharmacy.

## 2. Table Design

### Persons Table

Stores shared attributes for doctors and patients.  
• id (Primary Key)  
• first\_name (VARCHAR)  
• last\_name (VARCHAR)  
• age (INTEGER)  
• phone\_number (VARCHAR)

### Doctors Table

Stores doctor-specific information, linked to Persons.  
• id (Primary Key)  
• person\_id (Foreign Key)  
• specialization (VARCHAR)

### Patients Table

Stores patient-specific information, linked to Persons.  
• id (Primary Key)  
• person\_id (Foreign Key)

### Medications Table

Stores details about available drugs.  
• id (Primary Key)  
• name (VARCHAR)  
• dose (VARCHAR)  
• quantity (INTEGER)  
• expiry\_date (DATE)

### Prescriptions Table

Links doctors, patients, and medications.  
• id (Primary Key)  
• doctor\_id (Foreign Key)  
• patient\_id (Foreign Key)  
• medication\_id (Foreign Key)  
• prescription\_expiry (DATE)

### MedicationStock Table

Tracks inventory of medications in the pharmacy.  
• id (Primary Key)  
• medication\_id (Foreign Key)  
• quantity (INTEGER)

## 3. Relationships

• Persons → Doctors (One-to-One)  
• Persons → Patients (One-to-One)  
• Doctors → Prescriptions (One-to-Many)  
• Patients → Prescriptions (One-to-Many)  
• Medications → Prescriptions (One-to-Many)  
• Medications → MedicationStock (One-to-One)

## 4. ERD

A screenshot of a computer

AI-generated content may be incorrect.