#PK= Primary Key , FK=Foreign key

Patients :

Column Name	Data Type
patient_id	INT (PK)
name	VARCHAR
dob	DATE
gender	VARCHAR
contact_no	VARCHAR
address	TEXT
blood_group	VARCHAR
email	VARCHAR

Doctors:

Column Name	Data Type
doctor_id	INT (PK)
name	VARCHAR
specialization	VARCHAR
contact_no	VARCHAR
email	VARCHAR
department_id	INT (FK)

Departments :

Column Name	Data Type
department_id	INT (PK)
name	VARCHAR
location	VARCHAR

Appointments:

Column Name	Data Type
apt_id	INT (PK)
patient_id	INT (FK)
doctor_id	INT (FK)
apt_date	DATETIME
reason	TEXT
status	VARCHAR

Medical_Records:

Column Name	Data Type
record_id	INT (PK)
patient_id	INT (FK)
doctor_id	INT (FK)
diagnosis	TEXT
treatment	TEXT
record_date	DATE

Prescriptions:

Column Name	Data Type
pre_id	INT (PK)
record_id	INT (FK)
med_name	VARCHAR
dosage	VARCHAR
duration	VARCHAR

Bills:

Column Name	Data Type
bill_id	INT (PK)
patient_id	INT (FK)
amount	DECIMAL
billing_date	DATE
status	VARCHAR

Rooms:

Column Name	Data Type
room_id	INT (PK)
room_type	VARCHAR
status	VARCHAR
patient_id	INT (FK, nullable)

Relationships:

Patients → Appointments, Medical_Records, Bills, Rooms

Doctors → Appointments, Medical_Records

Departments → Doctors

Medical_Records → Prescriptions

Appointments → (Join between Doctors & Patients)

SQL codes:

```
-- 1. Departments Table
CREATE TABLE Departments (
    department_id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    location VARCHAR(100)
);

-- 2. Patients Table
CREATE TABLE Patients (
    patient_id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    dob DATE,
    gender VARCHAR(10),
    contact_number VARCHAR(15),
    address TEXT,
    blood_group VARCHAR(5),
    email VARCHAR(100)
);

-- 3. Doctors Table
CREATE TABLE Doctors (
    doctor_id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    specialization VARCHAR(100),
    contact_number VARCHAR(15),
    email VARCHAR(100),
    department_id INT,
    FOREIGN KEY (department_id) REFERENCES Departments(department_id)
);

-- 4. Appointments Table
CREATE TABLE Appointments (
    appointment_id INT AUTO_INCREMENT PRIMARY KEY,
    patient_id INT,
    doctor_id INT,
    appointment_date DATETIME,
    reason TEXT,
    status VARCHAR(20),
    FOREIGN KEY (patient_id) REFERENCES Patients(patient_id),
    FOREIGN KEY (doctor_id) REFERENCES Doctors(doctor_id)
);

-- 5. Medical Records Table
CREATE TABLE Medical_Records (
    record_id INT AUTO_INCREMENT PRIMARY KEY,
    patient_id INT,
    doctor_id INT,
```

```
    diagnosis TEXT,
    treatment TEXT,
    record_date DATE,
    FOREIGN KEY (patient_id) REFERENCES Patients(patient_id),
    FOREIGN KEY (doctor_id) REFERENCES Doctors(doctor_id)
);

-- 6. Prescriptions Table
CREATE TABLE Prescriptions (
    prescription_id INT AUTO_INCREMENT PRIMARY KEY,
    record_id INT,
    medicine_name VARCHAR(100),
    dosage VARCHAR(100),
    duration VARCHAR(50),
    FOREIGN KEY (record_id) REFERENCES Medical_Records(record_id)
);

-- 7. Bills Table
CREATE TABLE Bills (
    bill_id INT AUTO_INCREMENT PRIMARY KEY,
    patient_id INT,
    amount DECIMAL(10, 2),
    billing_date DATE,
    status VARCHAR(20),
    FOREIGN KEY (patient_id) REFERENCES Patients(patient_id)
);

-- 8. Rooms Table
CREATE TABLE Rooms (
    room_id INT AUTO_INCREMENT PRIMARY KEY,
    room_type VARCHAR(50),
    status VARCHAR(20),
    patient_id INT,
    FOREIGN KEY (patient_id) REFERENCES Patients(patient_id)
);
```