

## **Database Tables**

This document contains the DDL Schema for a Hospital EMR system. It lists the SQL database tables and schema used by the system's MySQL database.

### **Departments**

```
CREATE TABLE Departments (
    dept_id INT AUTO_INCREMENT PRIMARY KEY,
    dept_name VARCHAR(100) NOT NULL,
    building_location VARCHAR(50),
    head_of_dept_id INT,
    phone_extension VARCHAR(10)
);
```

### **Staff**

```
CREATE TABLE Staff (
    staff_id INT AUTO_INCREMENT PRIMARY KEY,
    first_name VARCHAR(50) NOT NULL,
    last_name VARCHAR(50) NOT NULL,
    role ENUM('Doctor', 'Nurse', 'Admin', 'Technician') NOT NULL,
    specialty VARCHAR(100),
    dept_id INT,
    license_number VARCHAR(50) UNIQUE,
    is_active BOOLEAN DEFAULT TRUE,
    FOREIGN KEY (dept_id) REFERENCES Departments(dept_id),
    INDEX idx_staff_role (role)
);
```

### **Insurance Providers**

```
CREATE TABLE Insurance_Providers (
    provider_id INT AUTO_INCREMENT PRIMARY KEY,
    provider_name VARCHAR(150) NOT NULL,
    claims_address TEXT,
    support_phone VARCHAR(20),
    contract_renewal_date DATE
);
```

## **Patients**

```
CREATE TABLE Patients (
    patient_id INT AUTO_INCREMENT PRIMARY KEY,
    first_name VARCHAR(50) NOT NULL,
    last_name VARCHAR(50) NOT NULL,
    date_of_birth DATE NOT NULL,
    gender ENUM('Male', 'Female', 'Other', 'Unknown'),
    primary_insurance_id INT,
    insurance_policy_number VARCHAR(50),
    emergency_contact_name VARCHAR(100),
    emergency_contact_phone VARCHAR(20),
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (primary_insurance_id) REFERENCES Insurance_Providers(provider_id),
    INDEX idx_patient_dob (date_of_birth),
    INDEX idx_patient_name (last_name, first_name)
);
```

## **Encounters**

```
CREATE TABLE Encounters (
    encounter_id INT AUTO_INCREMENT PRIMARY KEY,
    patient_id INT NOT NULL,
    staff_id INT NOT NULL, -- The primary provider seen
    dept_id INT, -- Which dept the visit happened in
    encounter_date DATETIME NOT NULL,
    encounter_type ENUM('Inpatient', 'Outpatient', 'Emergency', 'Telehealth'),
    visit_reason TEXT, -- Chief Complaint
    discharge_disposition VARCHAR(100), -- E.g., 'Home', 'Transfer', 'Deceased'
    FOREIGN KEY (patient_id) REFERENCES Patients(patient_id),
    FOREIGN KEY (staff_id) REFERENCES Staff(staff_id),
    FOREIGN KEY (dept_id) REFERENCES Departments(dept_id),
    INDEX idx_encounter_date (encounter_date)
);
```

## **Diagnoses**

```
CREATE TABLE Diagnoses (
    diagnosis_id INT AUTO_INCREMENT PRIMARY KEY,
    encounter_id INT NOT NULL,
    icd_code VARCHAR(10) NOT NULL,
    description VARCHAR(255) NOT NULL,
    is_primary BOOLEAN DEFAULT FALSE,
    severity ENUM('Mild', 'Moderate', 'Severe'),
```

```
FOREIGN KEY (encounter_id) REFERENCES Encounters(encounter_id) ON DELETE  
CASCADE  
);
```

## Medications

```
CREATE TABLE Medications (  
    med_id INT AUTO_INCREMENT PRIMARY KEY,  
    med_name VARCHAR(150) NOT NULL,  
    generic_name VARCHAR(150),  
    brand_name VARCHAR(150),  
    form ENUM('Tablet', 'Liquid', 'Injection', 'Capsule', 'Cream'),  
    standard dosage VARCHAR(50), -- e.g. "500mg"  
    requires_authorization BOOLEAN DEFAULT FALSE  
);
```

## Prescriptions

```
CREATE TABLE Prescriptions (  
    prescription_id INT AUTO_INCREMENT PRIMARY KEY,  
    encounter_id INT NOT NULL,  
    med_id INT NOT NULL,  
    dosage_instructions TEXT, -- e.g. "Take 1 pill twice daily"  
    quantity INT DEFAULT 30,  
    refills_remaining INT DEFAULT 0,  
    prescribed_date DATE,  
    FOREIGN KEY (encounter_id) REFERENCES Encounters(encounter_id),  
    FOREIGN KEY (med_id) REFERENCES Medications(med_id)  
);
```

## Orders

```
CREATE TABLE Orders (  
    test_code_id INT AUTO_INCREMENT PRIMARY KEY,  
    test_name VARCHAR(100) NOT NULL, -- e.g. "Complete Blood Count"  
    test_category VARCHAR(50), -- e.g. "Hematology"  
    base_cost DECIMAL(10, 2),  
    standard_range_min DECIMAL(10, 2),  
    standard_range_max DECIMAL(10, 2),  
    unit_of_measure VARCHAR(20) -- e.g. "mg/dL"  
);
```

## Results

```
CREATE TABLE Results (
    result_id INT AUTO_INCREMENT PRIMARY KEY,
    encounter_id INT NOT NULL,
    test_code_id INT NOT NULL,
    result_value DECIMAL(10, 2),
    result_text VARCHAR(50), -- For non-numeric results like "Positive/Negative"
    is_abnormal BOOLEAN DEFAULT FALSE,
    performed_at DATETIME DEFAULT CURRENT_TIMESTAMP,
    technician_notes TEXT,
    FOREIGN KEY (encounter_id) REFERENCES Encounters(encounter_id),
    FOREIGN KEY (test_code_id) REFERENCES Lab_Tests(test_code_id)
);
```