**Course Name: MYSQL**

**Semester: DISM**  
**Faculty Name: Muhammad Toufique**  
**Date: 14-4-2025**

**Instructions:**

This database simulates a small clinic setup. It has tables for **patients, doctors, appointments, prescriptions, medicines, and billing** — all connected logically. You can apply **joins, group by, constraints, and CRUD** operations easily here.

**Database Tables (with Constraints & Relations)**

CREATE TABLE doctors (

  doctor\_id INT PRIMARY KEY,

  name VARCHAR(50) NOT NULL,

  specialization VARCHAR(50),

  fee DECIMAL(6,2) CHECK (fee >= 500)

);

CREATE TABLE patients (

  patient\_id INT PRIMARY KEY,

  name VARCHAR(50) NOT NULL,

  gender VARCHAR(10) CHECK (gender IN ('Male', 'Female')),

  age INT CHECK (age >= 1),

  city VARCHAR(50) DEFAULT 'Lahore'

);

CREATE TABLE appointments (

  appointment\_id INT PRIMARY KEY,

  patient\_id INT,

  doctor\_id INT,

  appointment\_date DATE,

  status VARCHAR(20) CHECK (status IN ('Completed', 'Pending')),

  FOREIGN KEY (patient\_id) REFERENCES patients(patient\_id),

  FOREIGN KEY (doctor\_id) REFERENCES doctors(doctor\_id)

);

CREATE TABLE medicines (

  medicine\_id INT PRIMARY KEY,

  name VARCHAR(50) NOT NULL,

  price DECIMAL(5,2),

  stock INT CHECK (stock >= 0)

);

CREATE TABLE prescriptions (

  prescription\_id INT PRIMARY KEY,

  appointment\_id INT,

  medicine\_id INT,

  quantity INT CHECK (quantity > 0),

  FOREIGN KEY (appointment\_id) REFERENCES appointments(appointment\_id),

  FOREIGN KEY (medicine\_id) REFERENCES medicines(medicine\_id)

);

CREATE TABLE billing (

  bill\_id INT PRIMARY KEY,

  appointment\_id INT,

  amount DECIMAL(10,2),

  payment\_status VARCHAR(10) CHECK (payment\_status IN ('Paid', 'Unpaid')),

  FOREIGN KEY (appointment\_id) REFERENCES appointments(appointment\_id)

);

**20+ Sample Insertions:**

INSERT INTO doctors VALUES

(1, 'Dr. Ahmed', 'Cardiologist', 1000),

(2, 'Dr. Maria', 'Dermatologist', 800),

(3, 'Dr. Bilal', 'General Physician', 700);

INSERT INTO patients VALUES

(1, 'Ali', 'Male', 30, 'Lahore'),

(2, 'Sara', 'Female', 25, 'Karachi'),

(3, 'Zain', 'Male', 40, 'Multan'),

(4, 'Ayesha', 'Female', 35, 'Lahore'),

(5, 'Umar', 'Male', 28, 'Faisalabad');

INSERT INTO appointments VALUES

(101, 1, 1, '2024-02-01', 'Completed'),

(102, 2, 2, '2024-02-02', 'Completed'),

(103, 3, 3, '2024-02-05', 'Pending'),

(104, 4, 1, '2024-02-07', 'Completed'),

(105, 5, 3, '2024-02-10', 'Pending');

INSERT INTO medicines VALUES

(201, 'Panadol', 20.00, 100),

(202, 'Augmentin', 150.00, 50),

(203, 'Zyrtec', 30.00, 60);

INSERT INTO prescriptions VALUES

(301, 101, 201, 2),

(302, 102, 202, 1),

(303, 104, 201, 1),

(304, 104, 203, 2),

(305, 101, 203, 1);

INSERT INTO billing VALUES

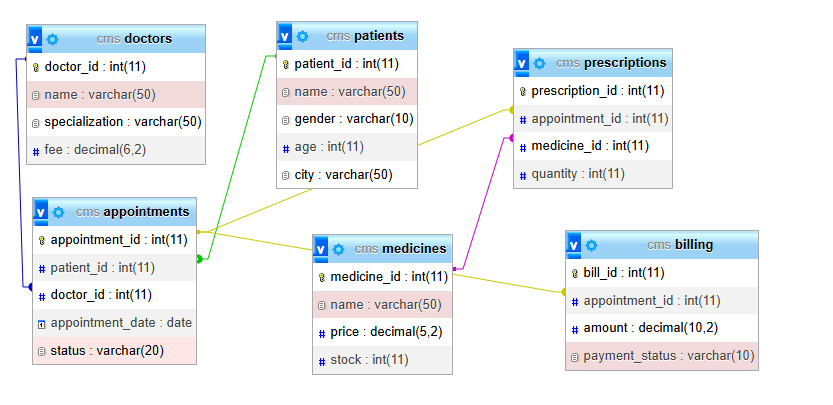
(401, 101, 1040.00, 'Paid'),

(402, 102, 950.00, 'Paid'),

(403, 103, 700.00, 'Unpaid'),

(404, 104, 1050.00, 'Paid'),

(405, 105, 700.00, 'Unpaid');

****

**Note:**

Use the given database and tables to solve the following queries. Write complete SQL statements for each. Read each question carefully.

1. Show the names of all patients who live in Lahore.
2. List all doctors with a fee greater than 800.
3. Show the patient name, doctor name, and appointment date for all appointments.
4. Display the list of all completed appointments along with doctor specialization.
5. List the medicine names prescribed in appointment ID 101.
6. Show each doctor's name with the number of appointments they have.
7. Calculate the total quantity of each medicine prescribed (group by medicine).
8. Find the total billing amount collected for each doctor (only for 'Paid' bills).
9. Insert a new patient named 'Tariq', Male, Age 33, from Karachi.
10. Delete all appointments where status is 'Pending'.