Lab: Adding Databases and Tables

1. create database cms;

==========

```
CREATE TABLE doctor (
    id INT AUTO_INCREMENT PRIMARY KEY,
    email VARCHAR(100) NOT NULL UNIQUE,
    name VARCHAR(100) NOT NULL,
    password VARCHAR(255) NOT NULL, -- consider hashing in application layer
    phone VARCHAR(20),
    specialty VARCHAR(50) NOT NULL,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
CREATE TABLE doctor_available_times (
    id INT AUTO_INCREMENT PRIMARY KEY,
    doctor_id INT NOT NULL,
    available_times VARCHAR(20) NOT NULL,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (doctor_id) REFERENCES doctor(id) ON DELETE CASCADE
);
CREATE TABLE patient (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    email VARCHAR(100) NOT NULL UNIQUE,
    password VARCHAR(255) NOT NULL,  -- Store hashed passwords in production
    phone VARCHAR(20),
    address VARCHAR(255),
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

```
CREATE TABLE appointment (
   id INT AUTO_INCREMENT PRIMARY KEY,
   doctor_id INT NOT NULL,
   patient_id INT NOT NULL,
   appointment_time DATETIME NOT NULL,
   status TINYINT NOT NULL DEFAULT 0, -- 0 = pending, 1 = completed, etc.
   created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
   FOREIGN KEY (doctor_id) REFERENCES doctor(id) ON DELETE CASCADE,
   FOREIGN KEY (patient_id) REFERENCES patient(id) ON DELETE CASCADE
);
CREATE TABLE admin (
   id INT AUTO_INCREMENT PRIMARY KEY,
   username VARCHAR(50) NOT NULL UNIQUE,
   password VARCHAR(255) NOT NULL,
   created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

Question 19

Submit the output of the SQL statement that lists all the tables using the “show tables” command. (3 points)

```
SHOW TABLES;
+----------------------+
| Tables_in_cms        |
+----------------------+
| admin                |
| appointment          |
| doctor               |
| doctor_available_times |
| patient              |
+----------------------+
```

Question 20

Submit the output of the SQL query that displays exactly 5 records from the Patient table. (3 points)

SELECT * FROM patient LIMIT 5;

+----+---------------+----------------------+------------+-------------+---------------------------+----------
----------+

| id | name | email | password | phone | address | created_at |

+----+---------------+----------------------+------------+-------------+---------------------------+----------
----------+

| 1 | Jane Doe | jane.doe@example.com | passJane1 | 888-111-1111 | 101 Oak St, Cityville | 2025-06-23 07:02:50 |

| 2 | John Smith | john.smith@example.com | smithSecure | 888-222-2222 | 202 Maple Rd, Townsville | 2025-06-23 07:02:50 |

| 3 | Emily Rose | emily.rose@example.com | emilyPass99 | 888-333-3333 | 303 Pine Ave, Villageton | 2025-06-23 07:02:50 |

| 4 | Michael Jordan | michael.j@example.com | airmj23 | 888-444-4444 | 404 Birch Ln, Metropolis | 2025-06-23 07:02:50 |

| 5 | Olivia Moon | olivia.m@example.com | moonshine12 | 888-555-5555 | 505 Cedar Blvd, Springfield | 2025-06-23 07:02:50 |

+----+---------------+----------------------+------------+-------------+---------------------------+----------
----------+

Question 21

Submit the output of the SQL statement that runs the *GetDailyAppointmentReportByDoctor* stored procedure. (3 points)

```
DELIMITER $$


CREATE PROCEDURE GetDailyAppointmentReportByDoctor(
   IN report_date DATE
)
BEGIN
   SELECT
```

```
    d.name AS doctor_name,
    a.appointment_time,
    a.status,
    p.name AS patient_name,
    p.phone AS patient_phone
  FROM
    appointment a
  JOIN
    doctor d ON a.doctor_id = d.id
  JOIN
    patient p ON a.patient_id = p.id
  WHERE
    DATE(a.appointment_time) = report_date
  ORDER BY
    d.name, a.appointment_time;
END$$

DELIMITER ;
CALL GetDailyAppointmentReportByDoctor('2025-04-15');
+-----------------+---------------------+--------+---------------+---------------+
| doctor_name     | appointment_time    | status | patient_name  | patient_phone |
+-----------------+---------------------+--------+---------------+---------------+
| Dr. Ava Hall    | 2025-04-15 11:00:00 |      1 | Lucas Turner  | 889-666-6666  |
| Dr. Mark Johnson | 2025-04-15 12:00:00 |     1 | Michael Jordan | 888-444-4444  |
| Dr. Mark Johnson | 2025-04-15 13:00:00 |     1 | Olivia Moon   | 888-555-5555  |
+-----------------+---------------------+--------+---------------+---------------+
```

Question 22

Submit the output of the SQL statement that runs the *GetDoctorWithMostPatientsByMonth* stored procedure. (3 points)

```
DELIMITER $$

CREATE PROCEDURE GetDoctorWithMostPatientsByMonth(
    IN input_month INT,
    IN input_year INT
)
BEGIN
    SELECT
        doctor_id,
        COUNT(patient_id) AS patients_seen
    FROM
        appointment
    WHERE
        MONTH(appointment_time) = input_month
        AND YEAR(appointment_time) = input_year
    GROUP BY
        doctor_id
    ORDER BY
        patients_seen DESC
    LIMIT 1;
END $$

DELIMITER ;

CALL GetDoctorWithMostPatientsByMonth(4, 2025);
```

```
+-----------+---------------+
| doctor_id | patients_seen |
+-----------+---------------+
|         2 |            31 |
+-----------+---------------+
```

Question 23

Submit the output of the SQL statement that runs the *GetDoctorWithMostPatientsByYear* stored procedure. (3 points)

```
DELIMITER $$

CREATE PROCEDURE GetDoctorWithMostPatientsByYear(
    IN input_year INT
)
BEGIN
    SELECT
        doctor_id,
        COUNT(patient_id) AS patients_seen
    FROM
        appointment
    WHERE
        YEAR(appointment_time) = input_year
    GROUP BY
        doctor_id
    ORDER BY
        patients_seen DESC
    LIMIT 1;
END $$

DELIMITER ;
```

```
CALL GetDoctorWithMostPatientsByYear(2025);
```

```
+-----------+--------------+
| doctor_id | patients_seen |
+-----------+--------------+
|         1 |           34 |
+-----------+--------------+
```