MAI172: Advance Database Technologies

Register Number: 2448513

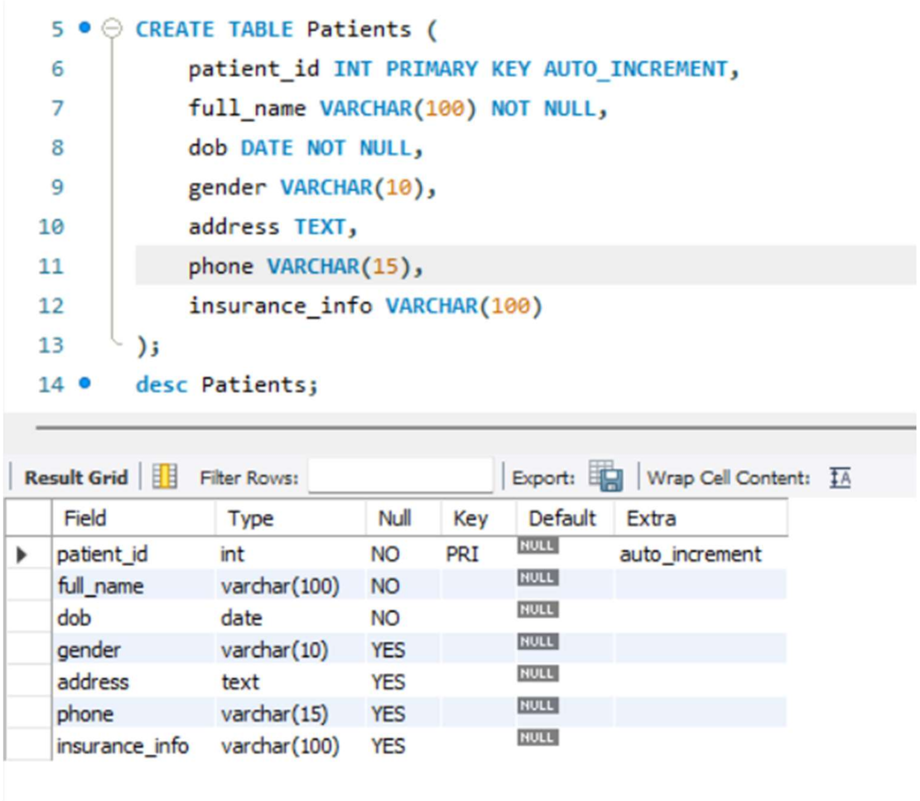
Name: Deshmukh Pratik Bhushanrao

Experiment Number and Name: 4: Sub-queries and correlated queries

Date: 01/08/2024 Time: 9.45 to 11.45

Creating Tables:

Creating Patients table:



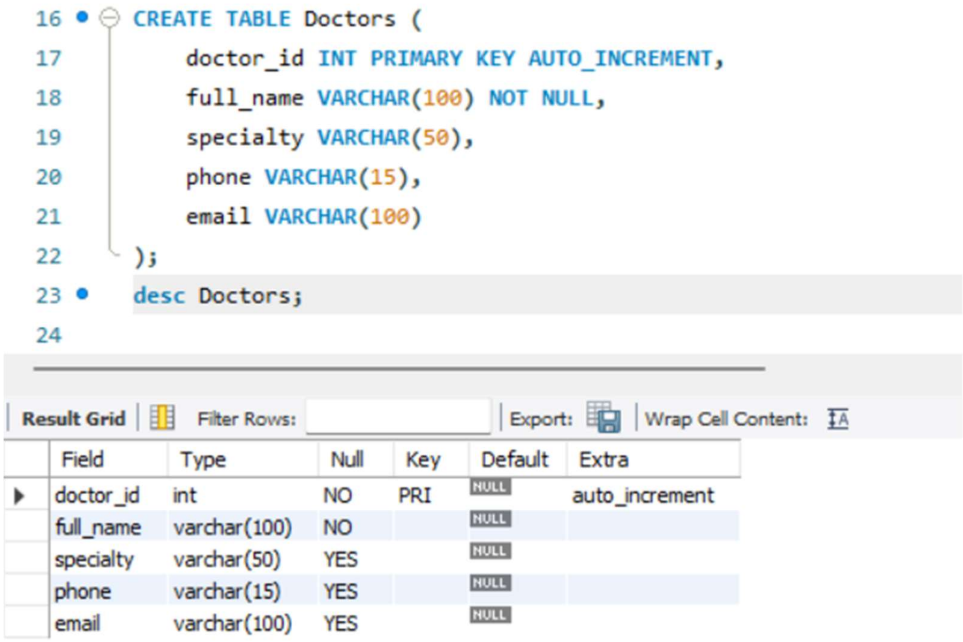
```
CREATE TABLE Patients (
    patient_id INT PRIMARY KEY AUTO_INCREMENT,
    full_name VARCHAR(100) NOT NULL,
    dob DATE NOT NULL,
    gender VARCHAR(10),
    address TEXT,
    phone VARCHAR(15),
    insurance_info VARCHAR(100)
);
desc Patients;
```

Field	Type	Null	Key	Default	Extra
patient_id	int	NO	PRI	NULL	auto_increment
full_name	varchar(100)	NO		NULL	
dob	date	NO		NULL	
gender	varchar(10)	YES		NULL	
address	text	YES		NULL	
phone	varchar(15)	YES		NULL	
insurance_info	varchar(100)	YES		NULL	

Inference:

The Patients table is created to store information about patients, including their ID, full name, date of birth, gender, address, phone number, and insurance information. This table serves as the primary repository for patient-related data, enabling efficient management and retrieval of patient details.

Creating Doctors table:



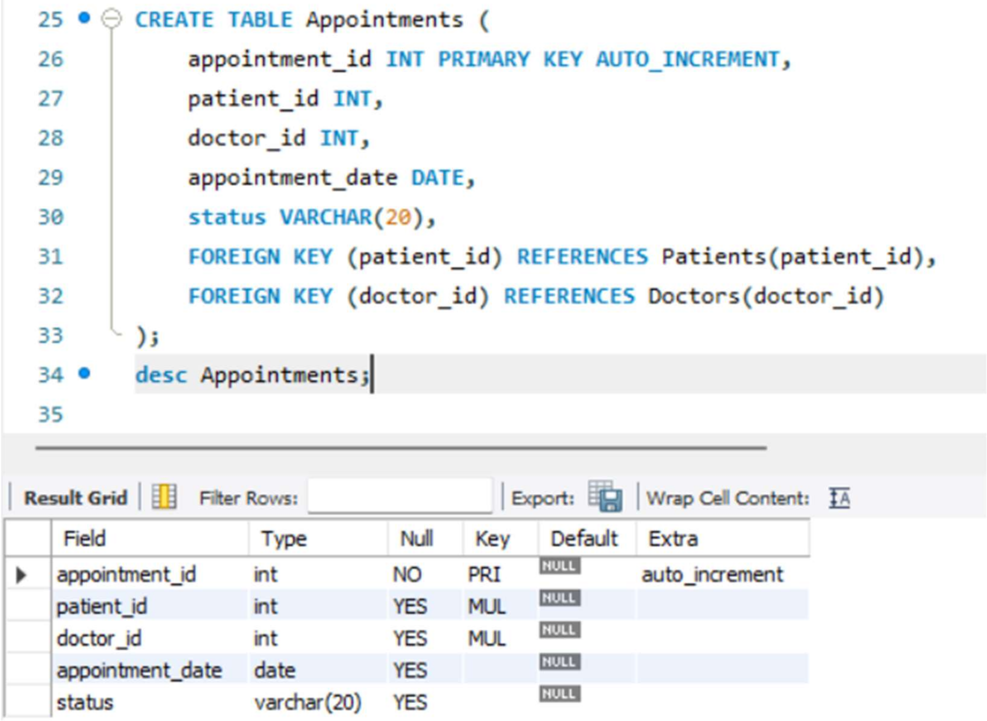
```
CREATE TABLE Doctors (
    doctor_id INT PRIMARY KEY AUTO_INCREMENT,
    full_name VARCHAR(100) NOT NULL,
    specialty VARCHAR(50),
    phone VARCHAR(15),
    email VARCHAR(100)
);
desc Doctors;
```

Field	Type	Null	Key	Default	Extra
doctor_id	int	NO	PRI	NULL	auto_increment
full_name	varchar(100)	NO		NULL	
specialty	varchar(50)	YES		NULL	
phone	varchar(15)	YES		NULL	
email	varchar(100)	YES		NULL	

Inference:

The Doctors table is created to store information about doctors, including their ID, full name, specialty, phone number, and email. This table helps manage doctor-related data, allowing for easy access to information about the medical professionals working in the hospital.

Creating Appointments table:



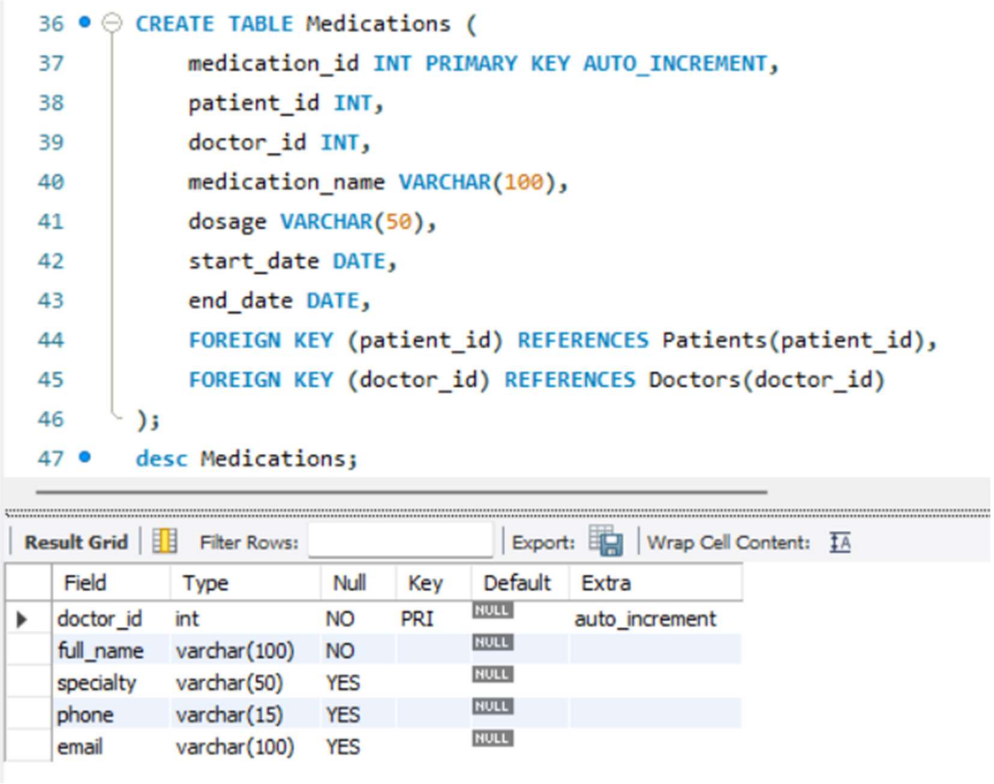
```
CREATE TABLE Appointments (
    appointment_id INT PRIMARY KEY AUTO_INCREMENT,
    patient_id INT,
    doctor_id INT,
    appointment_date DATE,
    status VARCHAR(20),
    FOREIGN KEY (patient_id) REFERENCES Patients(patient_id),
    FOREIGN KEY (doctor_id) REFERENCES Doctors(doctor_id)
);
desc Appointments;
```

Field	Type	Null	Key	Default	Extra
appointment_id	int	NO	PRI	NULL	auto_increment
patient_id	int	YES	MUL	NULL	
doctor_id	int	YES	MUL	NULL	
appointment_date	date	YES		NULL	
status	varchar(20)	YES		NULL	

Inference:

The Appointments table is created to store information about appointments, including the appointment ID, patient ID, doctor ID, appointment date, and status. This table links patients and doctors through appointments, facilitating the scheduling and tracking of patient visits.

Creating Medications table:



```
CREATE TABLE Medications (
    medication_id INT PRIMARY KEY AUTO_INCREMENT,
    patient_id INT,
    doctor_id INT,
    medication_name VARCHAR(100),
    dosage VARCHAR(50),
    start_date DATE,
    end_date DATE,
    FOREIGN KEY (patient_id) REFERENCES Patients(patient_id),
    FOREIGN KEY (doctor_id) REFERENCES Doctors(doctor_id)
);
desc Medications;
```

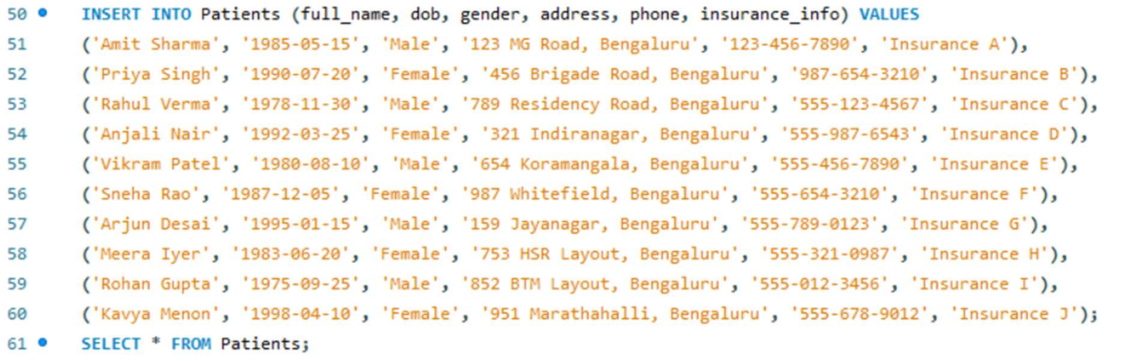
Field	Type	Null	Key	Default	Extra
doctor_id	int	NO	PRI	NULL	auto_increment
full_name	varchar(100)	NO		NULL	
specialty	varchar(50)	YES		NULL	
phone	varchar(15)	YES		NULL	
email	varchar(100)	YES		NULL	

Inference:

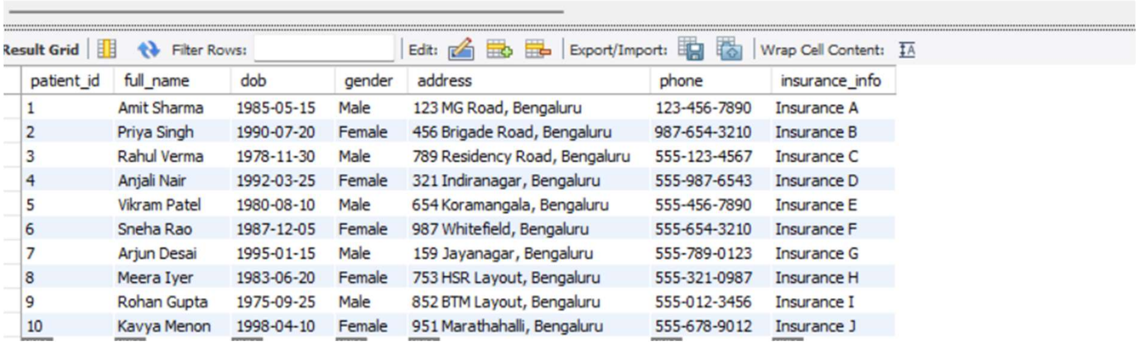
The Medications table is created to store information about medications prescribed to patients, including the medication ID, patient ID, doctor ID, medication name, dosage, start date, and end date. This table helps track the medications prescribed to patients and the doctors responsible for the prescriptions.

Inserting Records into the created tables:

1. Inserting records in the Patients table:



```sql
INSERT INTO Patients (full_name, dob, gender, address, phone, insurance_info) VALUES
('Amit Sharma', '1985-05-15', 'Male', '123 MG Road, Bengaluru', '123-456-7890', 'Insurance A'),
('Priya Singh', '1990-07-20', 'Female', '456 Brigade Road, Bengaluru', '987-654-3210', 'Insurance B'),
('Rahul Verma', '1978-11-30', 'Male', '789 Residency Road, Bengaluru', '555-123-4567', 'Insurance C'),
('Anjali Nair', '1992-03-25', 'Female', '321 Indiranagar, Bengaluru', '555-987-6543', 'Insurance D'),
('Vikram Patel', '1980-08-10', 'Male', '654 Koramangala, Bengaluru', '555-456-7890', 'Insurance E'),
('Sneha Rao', '1987-12-05', 'Female', '987 Whitefield, Bengaluru', '555-654-3210', 'Insurance F'),
('Arjun Desai', '1995-01-15', 'Male', '159 Jayanagar, Bengaluru', '555-789-0123', 'Insurance G'),
('Meera Iyer', '1983-06-20', 'Female', '753 HSR Layout, Bengaluru', '555-321-0987', 'Insurance H'),
('Rohan Gupta', '1975-09-25', 'Male', '852 BTM Layout, Bengaluru', '555-012-3456', 'Insurance I'),
('Kavya Menon', '1998-04-10', 'Female', '951 Marathahalli, Bengaluru', '555-678-9012', 'Insurance J');
SELECT * FROM Patients;
```

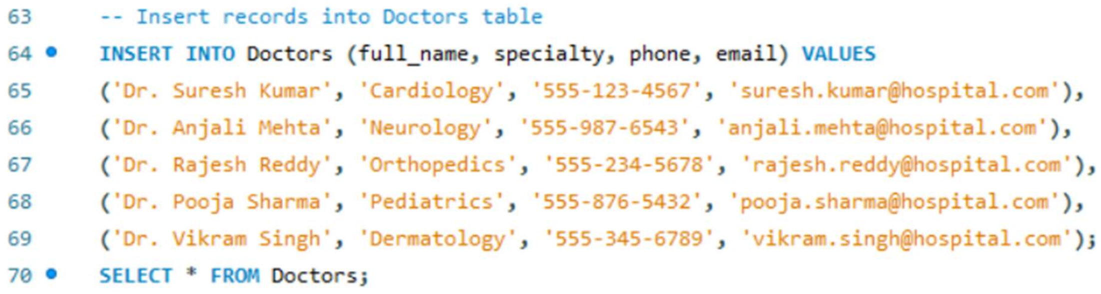


patient_id	full_name	dob	gender	address	phone	insurance_info
1	Amit Sharma	1985-05-15	Male	123 MG Road, Bengaluru	123-456-7890	Insurance A
2	Priya Singh	1990-07-20	Female	456 Brigade Road, Bengaluru	987-654-3210	Insurance B
3	Rahul Verma	1978-11-30	Male	789 Residency Road, Bengaluru	555-123-4567	Insurance C
4	Anjali Nair	1992-03-25	Female	321 Indiranagar, Bengaluru	555-987-6543	Insurance D
5	Vikram Patel	1980-08-10	Male	654 Koramangala, Bengaluru	555-456-7890	Insurance E
6	Sneha Rao	1987-12-05	Female	987 Whitefield, Bengaluru	555-654-3210	Insurance F
7	Arjun Desai	1995-01-15	Male	159 Jayanagar, Bengaluru	555-789-0123	Insurance G
8	Meera Iyer	1983-06-20	Female	753 HSR Layout, Bengaluru	555-321-0987	Insurance H
9	Rohan Gupta	1975-09-25	Male	852 BTM Layout, Bengaluru	555-012-3456	Insurance I
10	Kavya Menon	1998-04-10	Female	951 Marathahalli, Bengaluru	555-678-9012	Insurance J

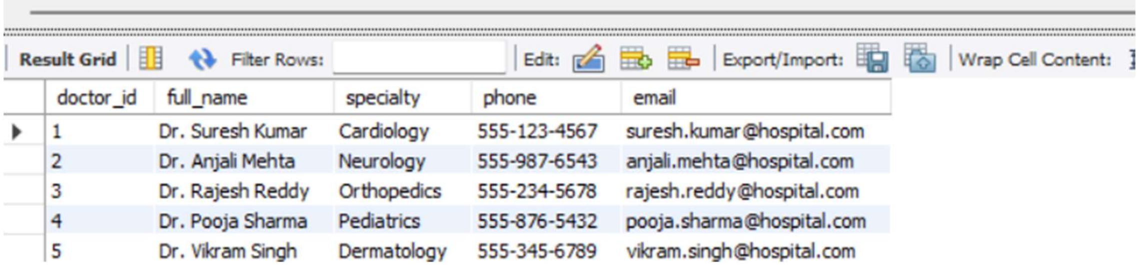
Inference:

Sample records are inserted into the Patients table to populate it with patient information. This data includes various patients with different names, dates of birth, genders, addresses, phone numbers, and insurance information. These records provide a realistic dataset for testing and querying patient-related information

2. Inserting records in the Doctors table:



```sql
-- Insert records into Doctors table
INSERT INTO Doctors (full_name, specialty, phone, email) VALUES
('Dr. Suresh Kumar', 'Cardiology', '555-123-4567', 'suresh.kumar@hospital.com'),
('Dr. Anjali Mehta', 'Neurology', '555-987-6543', 'anjali.mehta@hospital.com'),
('Dr. Rajesh Reddy', 'Orthopedics', '555-234-5678', 'rajesh.reddy@hospital.com'),
('Dr. Pooja Sharma', 'Pediatrics', '555-876-5432', 'pooja.sharma@hospital.com'),
('Dr. Vikram Singh', 'Dermatology', '555-345-6789', 'vikram.singh@hospital.com');
SELECT * FROM Doctors;
```

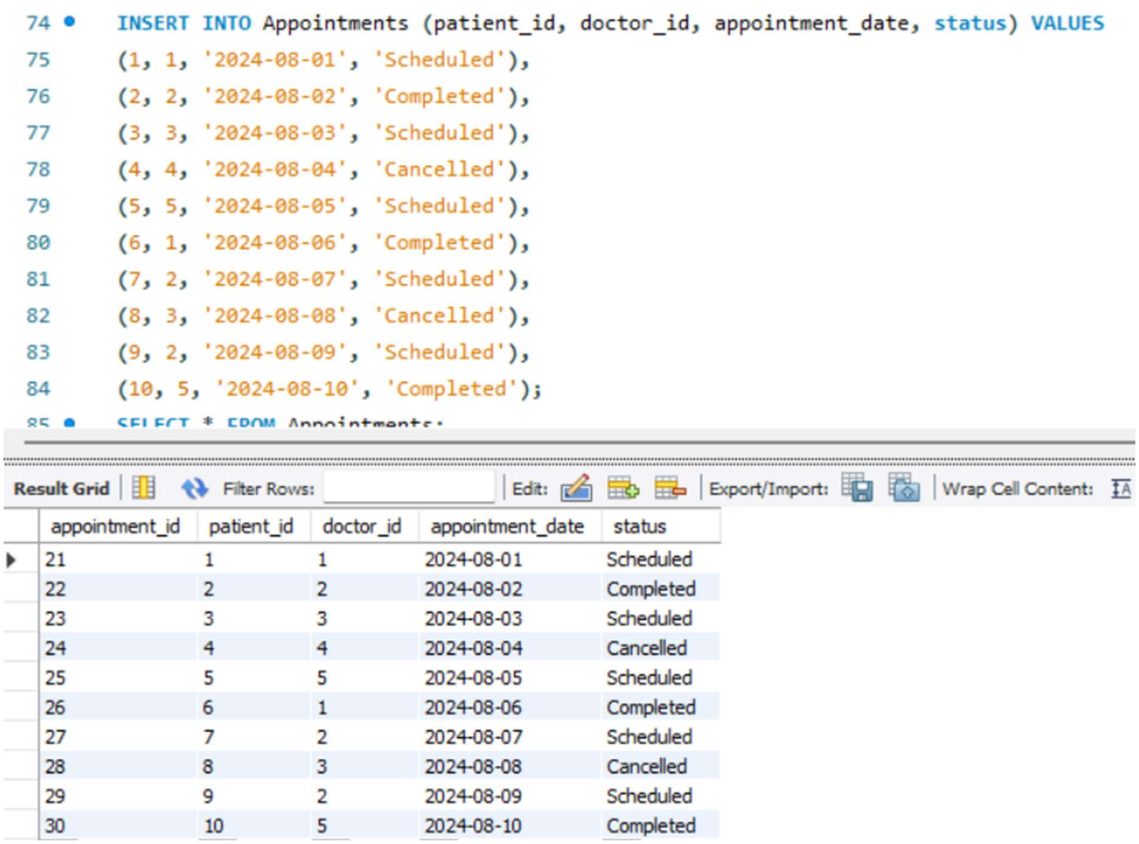


doctor_id	full_name	specialty	phone	email
1	Dr. Suresh Kumar	Cardiology	555-123-4567	suresh.kumar@hospital.com
2	Dr. Anjali Mehta	Neurology	555-987-6543	anjali.mehta@hospital.com
3	Dr. Rajesh Reddy	Orthopedics	555-234-5678	rajesh.reddy@hospital.com
4	Dr. Pooja Sharma	Pediatrics	555-876-5432	pooja.sharma@hospital.com
5	Dr. Vikram Singh	Dermatology	555-345-6789	vikram.singh@hospital.com

Inference:

Sample records are inserted into the Doctors table to populate it with doctor information. This data includes various doctors with different names, specialties, phone numbers, and email addresses. These records provide a realistic dataset for testing and querying doctor-related information.

3. Inserting records in the Appointments table:



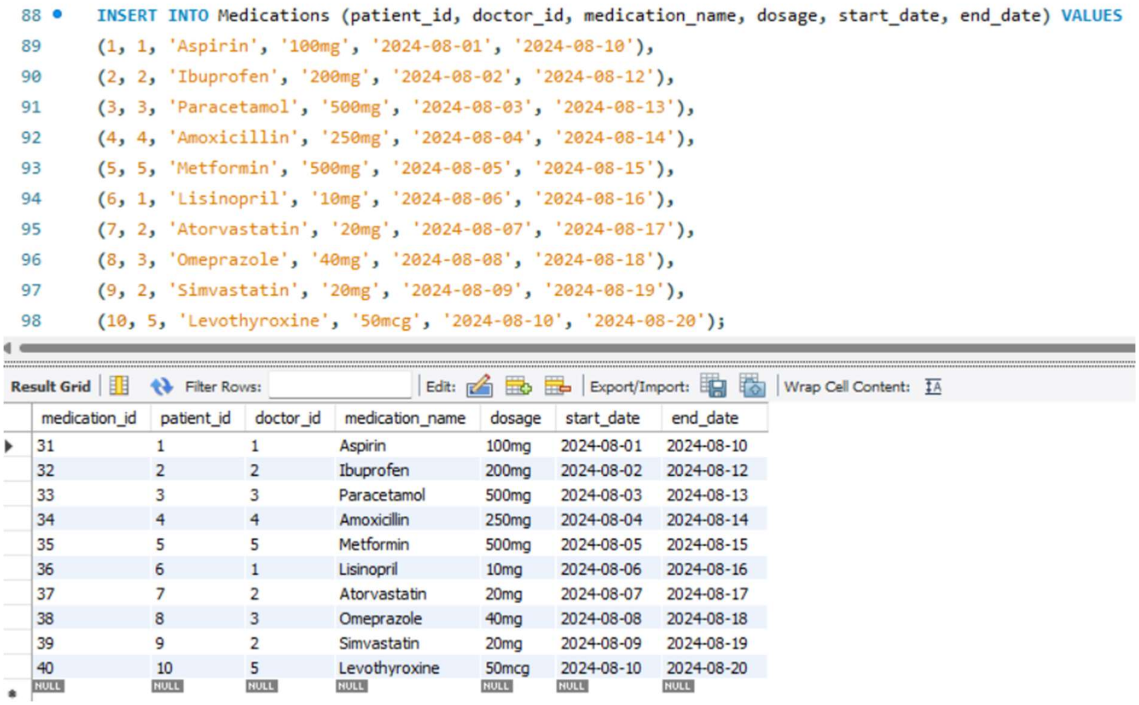
```sql
INSERT INTO Appointments (patient_id, doctor_id, appointment_date, status) VALUES
(1, 1, '2024-08-01', 'Scheduled'),
(2, 2, '2024-08-02', 'Completed'),
(3, 3, '2024-08-03', 'Scheduled'),
(4, 4, '2024-08-04', 'Cancelled'),
(5, 5, '2024-08-05', 'Scheduled'),
(6, 1, '2024-08-06', 'Completed'),
(7, 2, '2024-08-07', 'Scheduled'),
(8, 3, '2024-08-08', 'Cancelled'),
(9, 2, '2024-08-09', 'Scheduled'),
(10, 5, '2024-08-10', 'Completed');
SELECT * FROM Appointments;
```

appointment_id	patient_id	doctor_id	appointment_date	status
21	1	1	2024-08-01	Scheduled
22	2	2	2024-08-02	Completed
23	3	3	2024-08-03	Scheduled
24	4	4	2024-08-04	Cancelled
25	5	5	2024-08-05	Scheduled
26	6	1	2024-08-06	Completed
27	7	2	2024-08-07	Scheduled
28	8	3	2024-08-08	Cancelled
29	9	2	2024-08-09	Scheduled
30	10	5	2024-08-10	Completed

Inference:

Sample records are inserted into the Appointments table to populate it with appointment information. This data includes various appointments linking patients and doctors, with different appointment dates and statuses. These records provide a realistic dataset for testing and querying appointment-related information.

4. Inserting records in the Medications table:



```sql
INSERT INTO Medications (patient_id, doctor_id, medication_name, dosage, start_date, end_date) VALUES
(1, 1, 'Aspirin', '100mg', '2024-08-01', '2024-08-10'),
(2, 2, 'Ibuprofen', '200mg', '2024-08-02', '2024-08-12'),
(3, 3, 'Paracetamol', '500mg', '2024-08-03', '2024-08-13'),
(4, 4, 'Amoxicillin', '250mg', '2024-08-04', '2024-08-14'),
(5, 5, 'Metformin', '500mg', '2024-08-05', '2024-08-15'),
(6, 1, 'Lisinopril', '10mg', '2024-08-06', '2024-08-16'),
(7, 2, 'Atorvastatin', '20mg', '2024-08-07', '2024-08-17'),
(8, 3, 'Omeprazole', '40mg', '2024-08-08', '2024-08-18'),
(9, 2, 'Simvastatin', '20mg', '2024-08-09', '2024-08-19'),
(10, 5, 'Levothyroxine', '50mcg', '2024-08-10', '2024-08-20');
```

medication_id	patient_id	doctor_id	medication_name	dosage	start_date	end_date
31	1	1	Aspirin	100mg	2024-08-01	2024-08-10
32	2	2	Ibuprofen	200mg	2024-08-02	2024-08-12
33	3	3	Paracetamol	500mg	2024-08-03	2024-08-13
34	4	4	Amoxicillin	250mg	2024-08-04	2024-08-14
35	5	5	Metformin	500mg	2024-08-05	2024-08-15
36	6	1	Lisinopril	10mg	2024-08-06	2024-08-16
37	7	2	Atorvastatin	20mg	2024-08-07	2024-08-17
38	8	3	Omeprazole	40mg	2024-08-08	2024-08-18
39	9	2	Simvastatin	20mg	2024-08-09	2024-08-19
40	10	5	Levothyroxine	50mcg	2024-08-10	2024-08-20
NULL	NULL	NULL	NULL	NULL	NULL	NULL

Inference:

Sample records are inserted into the Medications table to populate it with medication information. This data includes various medications prescribed to patients by doctors, with different medication names, dosages, start dates, and end dates. These records provide a realistic dataset for testing and querying medication-related information.

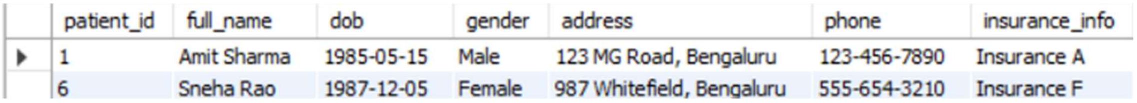
A. Nested Queries

1. Find the names of patients with appointments scheduled with doctor Suresh Kumar.

Query:

SELECT * FROM Patients where patient_id in (select patient_id from Appointments where doctor_id in (select doctor_id from Doctors where full_name="Dr. Suresh Kumar"));

Output:



	patient_id	full_name	dob	gender	address	phone	insurance_info
▶	1	Amit Sharma	1985-05-15	Male	123 MG Road, Bengaluru	123-456-7890	Insurance A
	6	Sneha Rao	1987-12-05	Female	987 Whitefield, Bengaluru	555-654-3210	Insurance F

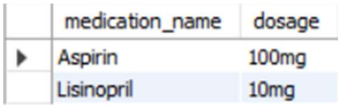
Inference:

This query retrieves the names of patients who have appointments with ‘Dr. Suresh Kumar’. It helps identify which patients are under the care of this specific doctor.

2. List the medication name and dosage prescribed by doctors specializing in ‘Cardiology’.

Query:

Output:



	medication_name	dosage
▶	Aspirin	100mg
	Lisinopril	10mg

Inference:

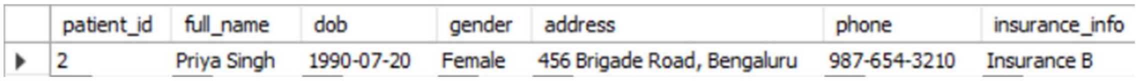
This query lists all medications prescribed by doctors specializing in ‘Cardiology’. It helps understand the treatment patterns and medications commonly prescribed by cardiologists.

3. Find the details of appointments for patients who are prescribed ‘Aspirin’.

Query:

Select * from Patients where patient_id in (select patient_id from Medications where medication_name = "Ibuprofen");

Output:



	patient_id	full_name	dob	gender	address	phone	insurance_info
▶	2	Priya Singh	1990-07-20	Female	456 Brigade Road, Bengaluru	987-654-3210	Insurance B

Inference:

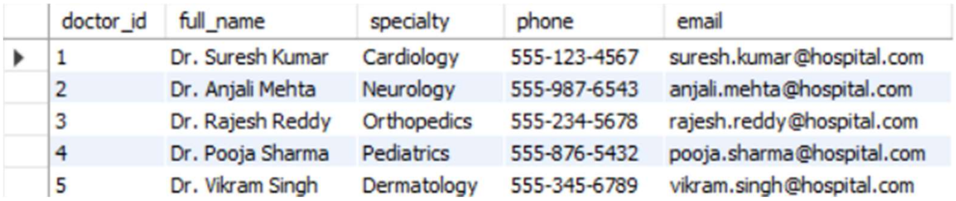
This query retrieves appointment details for patients who have been prescribed ‘Aspirin’. It helps track the appointments of patients on this specific medication.

4. Get the names of doctors who have appointments with patients living in ‘Bengaluru’.

Query:

select * from Doctors where doctor_id in (select doctor_id from Appointments where patient_id in (select patient_id from Patients where address like "%Bengaluru"));

Output:



doctor_id	full_name	specialty	phone	email
1	Dr. Suresh Kumar	Cardiology	555-123-4567	suresh.kumar@hospital.com
2	Dr. Anjali Mehta	Neurology	555-987-6543	anjali.mehta@hospital.com
3	Dr. Rajesh Reddy	Orthopedics	555-234-5678	rajesh.reddy@hospital.com
4	Dr. Pooja Sharma	Pediatrics	555-876-5432	pooja.sharma@hospital.com
5	Dr. Vikram Singh	Dermatology	555-345-6789	vikram.singh@hospital.com

Inference:

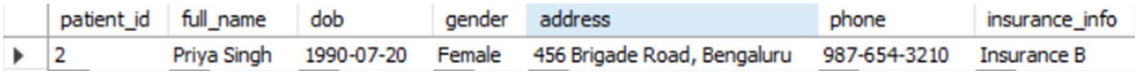
This query lists the names of doctors who have appointments with patients residing in ‘Bengaluru’. It helps identify which doctors are treating patients from this specific location.

5. Find the names of patients who have completed appointments and are prescribed ‘Ibuprofen’.

Query:

select * from Patients where patient_id in (select patient_id from Medications where medication_name= "Ibuprofen");

Output:



patient_id	full_name	dob	gender	address	phone	insurance_info
2	Priya Singh	1990-07-20	Female	456 Brigade Road, Bengaluru	987-654-3210	Insurance B

Inference:

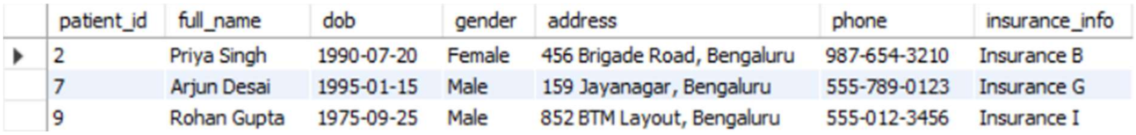
This query retrieves the names of patients who have completed their appointments and are prescribed ‘Ibuprofen’. It helps track patients who have finished their consultations and are on this medication.

6. List the names of patients who have appointments with doctors specializing in ‘Neurology’.

Query:

select * from Patients where patient_id in (select patient_id from Appointments where doctor_id in (select doctor_id from Doctors where specialty="Neurology"));

Output:



patient_id	full_name	dob	gender	address	phone	insurance_info
2	Priya Singh	1990-07-20	Female	456 Brigade Road, Bengaluru	987-654-3210	Insurance B
7	Arjun Desai	1995-01-15	Male	159 Jayanagar, Bengaluru	555-789-0123	Insurance G
9	Rohan Gupta	1975-09-25	Male	852 BTM Layout, Bengaluru	555-012-3456	Insurance I

Inference:

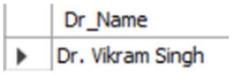
This query lists the names of patients who have appointments with neurologists. It helps identify patients receiving neurological care.

7. Find the names of doctors who have prescribed ‘Metformin’ to any patient.

Query:

select full_name as Dr_Name from Doctors where doctor_id in (select doctor_id from Medications where medication_name= "Metformin");

Output:



	Dr_Name
▶	Dr. Vikram Singh

Inference:

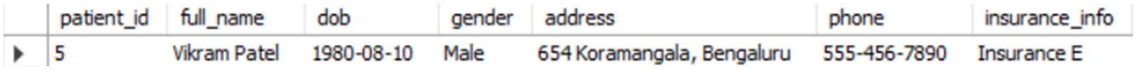
This query retrieves the names of doctors who have prescribed ‘Metformin’. It helps identify which doctors are prescribing this medication

8. Get the details of patients who have appointments scheduled after ‘2024-08-05’.

Query:

select * from Patients where patient_id in (select patient_id from Appointments where appointment_date = "2024-08-05");

Output:



	patient_id	full_name	dob	gender	address	phone	insurance_info
▶	5	Vikram Patel	1980-08-10	Male	654 Koramangala, Bengaluru	555-456-7890	Insurance E

Inference:

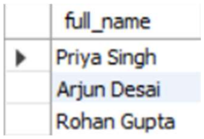
This query lists the details of patients with appointments scheduled after ‘2024-08-05’. It helps track upcoming appointments beyond this date.

9. List the names of patients who have been prescribed medications by ‘Dr. Anjali Mehta’.

Query:

select full_name from Patients where patient_id in (select patient_id from Medications where doctor_id in (select doctor_id from Doctors where full_name= "Dr. Anjali Mehta"));

Output:



	full_name
▶	Priya Singh
	Arjun Desai
	Rohan Gupta

Inference:

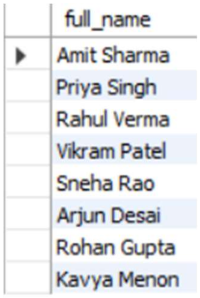
This query retrieves the names of patients who have been prescribed medications by ‘Dr. Anjali Mehta’. It helps identify patients under her care and the medications she has prescribed

10. Find the names of patients who have both ‘Scheduled’ and ‘Completed’ appointments.

Query:

select full_name from Patients where patient_id in (select patient_id from Appointments where status="Scheduled" or status="Completed");

Output:



```
full_name
Amit Sharma
Priya Singh
Rahul Verma
Vikram Patel
Sneha Rao
Arjun Desai
Rohan Gupta
Kavya Menon
```

Inference:

This query lists the names of patients who have both ‘Scheduled’ and ‘Completed’ appointments. It helps track patients who have ongoing and past appointments.

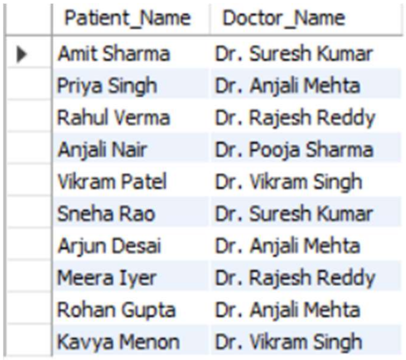
B. Correlated Queries

1. Find the names of patients along with the names of doctors they have appointments with.

Query:

select p.full_name AS Patient_Name, d.full_name AS Doctor_Name from Patients p join Appointments a on a.patient_id = p.patient_id join Doctors d on a.doctor_id = d.doctor_id;

Output:



	Patient_Name	Doctor_Name
▶	Amit Sharma	Dr. Suresh Kumar
	Priya Singh	Dr. Anjali Mehta
	Rahul Verma	Dr. Rajesh Reddy
	Anjali Nair	Dr. Pooja Sharma
	Vikram Patel	Dr. Vikram Singh
	Sneha Rao	Dr. Suresh Kumar
	Arjun Desai	Dr. Anjali Mehta
	Meera Iyer	Dr. Rajesh Reddy
	Rohan Gupta	Dr. Anjali Mehta
	Kavya Menon	Dr. Vikram Singh

Inference:

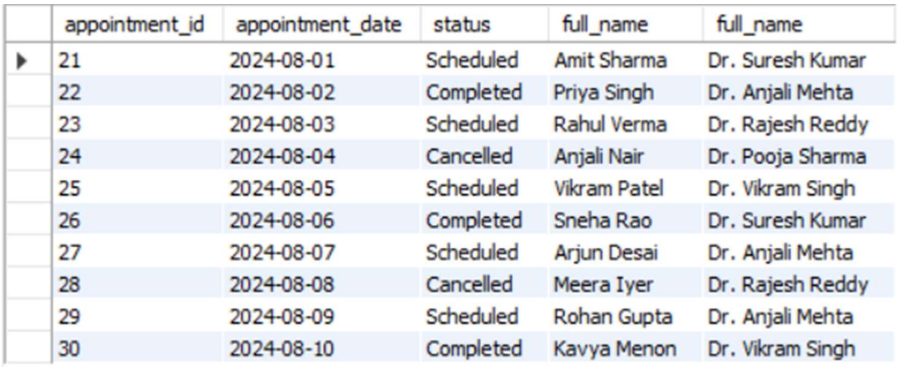
This query retrieves the names of patients and the corresponding names of doctors they have appointments with. It helps identify the doctor-patient relationships and provides a clear view of which patients are being treated by which doctors.

2. List the details of appointments along with patient and doctor names.

Query:

select appointment_id, appointment_date, status, p.full_name, d.full_name from Appointments a join Patients p on a.patient_id = p.patient_id join Doctors d on a.doctor_id = d.doctor_id;

Output:



	appointment_id	appointment_date	status	full_name	full_name
▶	21	2024-08-01	Scheduled	Amit Sharma	Dr. Suresh Kumar
	22	2024-08-02	Completed	Priya Singh	Dr. Anjali Mehta
	23	2024-08-03	Scheduled	Rahul Verma	Dr. Rajesh Reddy
	24	2024-08-04	Cancelled	Anjali Nair	Dr. Pooja Sharma
	25	2024-08-05	Scheduled	Vikram Patel	Dr. Vikram Singh
	26	2024-08-06	Completed	Sneha Rao	Dr. Suresh Kumar
	27	2024-08-07	Scheduled	Arjun Desai	Dr. Anjali Mehta
	28	2024-08-08	Cancelled	Meera Iyer	Dr. Rajesh Reddy
	29	2024-08-09	Scheduled	Rohan Gupta	Dr. Anjali Mehta
	30	2024-08-10	Completed	Kavya Menon	Dr. Vikram Singh

Inference:

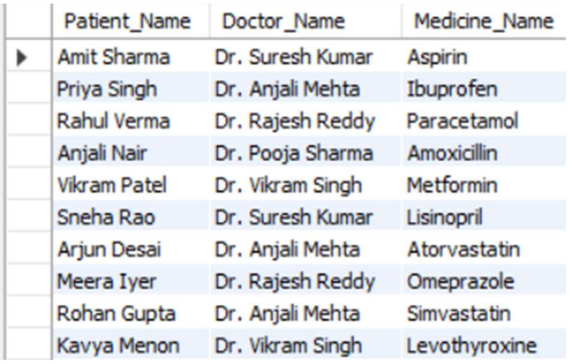
This query lists all the details of appointments, including the appointment ID, patient name, doctor name, appointment date, and status. It provides a comprehensive overview of all scheduled appointments, making it easier to manage and track appointments.

3. Get the names of patients who have been prescribed medications along with the prescribing doctor’s name.

Query:

select p.full_name as Patient_Name, d.full_name as Doctor_Name, m.medication_name as Medicine_Name from Patients p join Medications m on m.patient_id = p.patient_id join Doctors d on m.doctor_id = d.doctor_id;

Output:



Patient_Name	Doctor_Name	Medicine_Name
Amit Sharma	Dr. Suresh Kumar	Aspirin
Priya Singh	Dr. Anjali Mehta	Ibuprofen
Rahul Verma	Dr. Rajesh Reddy	Paracetamol
Anjali Nair	Dr. Pooja Sharma	Amoxicillin
Vikram Patel	Dr. Vikram Singh	Metformin
Sneha Rao	Dr. Suresh Kumar	Lisinopril
Arjun Desai	Dr. Anjali Mehta	Atorvastatin
Meera Iyer	Dr. Rajesh Reddy	Omeprazole
Rohan Gupta	Dr. Anjali Mehta	Simvastatin
Kavya Menon	Dr. Vikram Singh	Levothyroxine

Inference:

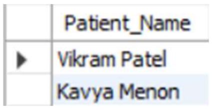
This query retrieves the names of patients who have been prescribed medications, along with the names of the doctors who prescribed them. It helps track medication prescriptions and the responsible doctors, providing insights into treatment plans.

4. Find the names of patients who have appointments with doctors specializing in ‘Dermatology’.

Query:

select p.full_name as Patient_Name from Patients p join Appointments a on a.patient_id= p.patient_id join Doctors d on a.doctor_id = d.doctor_id where d.specialty = "Dermatology";

Output:



Patient_Name
Vikram Patel
Kavya Menon

Inference:

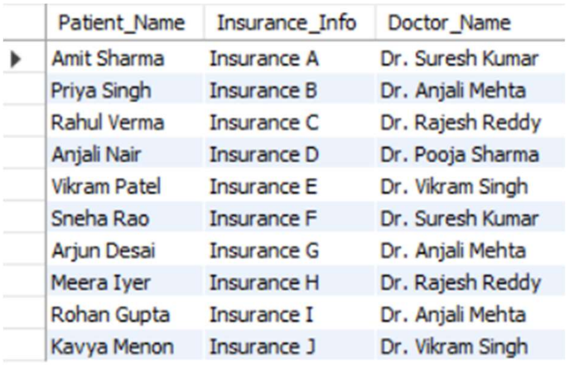
This query lists the names of patients who have appointments with dermatologists. It helps identify patients receiving dermatological care and provides insights into the patient load for this specialty.

5. List the names of patients along with their insurance information and the names of doctors they have appointments with.

Query:

SELECT p.full_name AS Patient_Name, p.insurance_info as Insurance_Info, d.full_name AS Doctor_Name FROM Appointments a JOIN Patients p ON a.patient_id = p.patient_id JOIN Doctors d ON a.doctor_id = d.doctor_id;

Output:



Patient_Name	Insurance_Info	Doctor_Name
Amit Sharma	Insurance A	Dr. Suresh Kumar
Priya Singh	Insurance B	Dr. Anjali Mehta
Rahul Verma	Insurance C	Dr. Rajesh Reddy
Anjali Nair	Insurance D	Dr. Pooja Sharma
Vikram Patel	Insurance E	Dr. Vikram Singh
Sneha Rao	Insurance F	Dr. Suresh Kumar
Arjun Desai	Insurance G	Dr. Anjali Mehta
Meera Iyer	Insurance H	Dr. Rajesh Reddy
Rohan Gupta	Insurance I	Dr. Anjali Mehta
Kavya Menon	Insurance J	Dr. Vikram Singh

Inference:

This query retrieves the names of patients, their insurance information, and the names of doctors they have appointments with. It helps track patient insurance details and the corresponding doctors, providing a comprehensive view of patient-doctor-insurance relationships.