Queries On Whole Database (Healthcare Management System)

Easy Level Queries

1. List all patients and their insurance information.

```
SELECT first_name, last_name, insurance_info FROM Patient;
```

2. Count the total number of doctors.

```
SELECT COUNT(*) AS total_doctors FROM Doctor;
```

3. List all inventory items with quantities greater than 50.

```
SELECT item_name, quantity FROM Inventory WHERE quantity > 50;
```

4. Display appointments scheduled for a specific doctor.

```
SELECT appointment_id, patient_id, appointment_date
FROM Appointment
WHERE doctor_id = 'D01';
```

5. Count the number of appointments scheduled for today.

```
SELECT COUNT(*) AS appointments_today
FROM Appointment
WHERE DATE(appointment_date) = CURRENT_DATE();
```

6. Find all patients with a gender of 'Male.'

```
SELECT first_name, last_name
FROM Patient
WHERE gender = 'Male';
```

7. List all unpaid bills.

```
SELECT billing_id, amount, billing_date
FROM Billing
WHERE payment_status = 'Pending';
```

8. Retrieve the names of all doctors with "Cardiology" as their specialization.

```
SELECT first_name, last_name
FROM Doctor
WHERE specialisation = 'Cardiology';
```

9. Display the names of all items in the inventory that have already expired.

```
SELECT item_name, expiration_date
FROM Inventory
WHERE expiration_date < CURRENT_DATE();
```

10. Find the total amount billed for a specific patient.

```
SELECT SUM(amount) AS total_billed
FROM Billing
WHERE patient_id = 'P01';
```

Medium Level Queries

1. Find the total number of appointments per doctor.

```
SELECT doctor_id, COUNT(*) AS total_appointments
FROM Appointment
GROUP BY doctor_id;
```

2. Retrieve the total quantity of items in the inventory.

```
SELECT SUM(quantity) AS total_items
FROM Inventory;
```

3. List all patients who have more than one billing record.

```
SELECT patient_id, COUNT(*) AS bill_count
FROM Billing
GROUP BY patient_id
HAVING bill_count > 1;
```

4. Find the average billing amount for all patients.

```
SELECT AVG(amount) AS average_billing
FROM Billing;
```

5. List all appointments for patients born after 1990.

```
SELECT a.appointment_id, a.patient_id, a.appointment_date
FROM Appointment a
JOIN Patient p ON a.patient_id = p.patient_id
WHERE p.dob > '1990-01-01';
```

6. Find the doctor with the most appointments scheduled.

```
SELECT doctor_id, COUNT(*) AS total_appointments
FROM Appointment
GROUP BY doctor_id
ORDER BY total_appointments DESC
LIMIT 1;
```

7. Retrieve the inventory items with less than 10 days to expiration.

```
SELECT item_name, expiration_date
FROM Inventory
WHERE expiration_date BETWEEN CURRENT_DATE() AND DATE_ADD(CURRENT_DATE(),
INTERVAL 10 DAY);
```

8. Display all appointments for patients who have an unpaid bill.

```
SELECT a.appointment_id, a.patient_id, a.doctor_id
FROM Appointment a
JOIN Billing b ON a.patient_id = b.patient_id
WHERE b.payment_status = 'Pending';
```

9. List all patients who have no insurance information.

```
SELECT first_name, last_name
FROM Patient
WHERE insurance_info IS NULL OR insurance_info = '';
```

10. Find the total number of doctors with "Surgery" in their specialization.

```
SELECT COUNT(*) AS total_surgeons
```

```
FROM Doctor
WHERE specialisation LIKE '%Surgery%';
```

Hard Level Queries

1. Find the patient with the highest total billing amount.

```
SELECT patient_id, SUM(amount) AS total_billed
FROM Billing
GROUP BY patient_id
ORDER BY total_billed DESC
LIMIT 1;
```

2. Retrieve the top 3 inventory items with the highest quantities.

```
SELECT item_name, quantity
FROM Inventory
ORDER BY quantity DESC
LIMIT 3;
```

3. Find the top 5 patients who have the most appointments scheduled.

```
SELECT patient_id, COUNT(*) AS appointment_count
FROM Appointment
GROUP BY patient_id
ORDER BY appointment_count DESC
LIMIT 5;
```

4. Identify patients who have appointments with more than 2 different doctors.

```
SELECT patient_id, COUNT(DISTINCT doctor_id) AS doctor_count
FROM Appointment
GROUP BY patient_id
HAVING doctor_count > 2;
```

5. List all appointments scheduled with doctors who specialize in "Orthopedics."

```
SELECT a.appointment_id, a.patient_id, a.appointment_date
FROM Appointment a
JOIN Doctor d ON a.doctor_id = d.doctor_id
WHERE d.specialisation = 'Orthopedics';
```

6. Retrieve the total billing amount for each patient born before 1980.

```
SELECT b.patient_id, SUM(b.amount) AS total_billed
FROM Billing b
JOIN Patient p ON b.patient_id = p.patient_id
WHERE p.dob < '1980-01-01'
GROUP BY b.patient_id;
```

7. Find all doctors who are not associated with any appointments.

```
SELECT d.first_name, d.last_name
FROM Doctor d
LEFT JOIN Appointment a ON d.doctor_id = a.doctor_id
WHERE a.doctor_id IS NULL;
```

8. List all patients whose total billing amount exceeds $1,000.

```
SELECT patient_id, SUM(amount) AS total_billed
FROM Billing
GROUP BY patient_id
HAVING total_billed > 1000;
```

9. Find the total number of male and female patients who have an unpaid bill.

```
SELECT p.gender, COUNT(DISTINCT b.patient_id) AS total_patients
FROM Patient p
JOIN Billing b ON p.patient_id = b.patient_id
WHERE b.payment_status = 'Pending'
GROUP BY p.gender;
```

10. Retrieve the average quantity of items grouped by their expiration year.

```
SELECT YEAR(expiration_date) AS expiration_year, AVG(quantity) AS average_quantity
FROM Inventory
GROUP BY expiration_year;
```

Queries On Individual Tables (Healthcare Management System)

Doctor Table Questions:

1. What are the most common specializations among the doctors?

```
SELECT specialisation, COUNT(*) AS count
FROM Doctor
GROUP BY specialisation
ORDER BY count DESC;
```

2. How many doctors have a specific schedule pattern?

```
SELECT COUNT(*) AS doctor_count
FROM Doctor
WHERE schedule LIKE '%Morning%';
```

3. Which doctor has the longest schedule text?

```
SELECT first_name, last_name, schedule
FROM Doctor
ORDER BY LENGTH(schedule) DESC
LIMIT 1;
```

4. How many doctors are assigned a specific specialization?

```
SELECT COUNT(*) AS doctor_count
FROM Doctor
WHERE specialisation = 'Cardiology';
```

5. What percentage of doctors are assigned to a particular specialization?

```
SELECT (COUNT(*) * 100.0 / (SELECT COUNT(*) FROM Doctor)) AS percentage
FROM Doctor
WHERE specialisation = 'Neurology';
```

6. How many unique specializations exist in the doctor table?

```
SELECT COUNT(DISTINCT specialisation) AS unique_specializations
FROM Doctor;
```

7. What is the frequency distribution of specializations among doctors?

```
SELECT specialisation, COUNT(*) AS count
```

```
FROM Doctor
GROUP BY specialisation
ORDER BY count DESC;
```

8. How many doctors have the same first or last name?

```
SELECT first_name, last_name, COUNT(*) AS name_count
FROM Doctor
GROUP BY first_name, last_name
HAVING COUNT(*) > 1;
```

9. What is the total count of doctors available in the table?

```
SELECT COUNT(*) AS total_doctors
FROM Doctor;
```

10. Which specializations have fewer than 2 doctors assigned?

```
SELECT specialisation
FROM Doctor
GROUP BY specialisation
HAVING COUNT(*) < 2;
```

Patient Table Questions:

1. How many patients are registered for each gender?

```
SELECT gender, COUNT(*) AS patient_count
FROM Patient
GROUP BY gender;
```

2. What is the distribution of patients by their birth years?

```
SELECT YEAR(dob) AS birth_year, COUNT(*) AS patient_count
FROM Patient
GROUP BY YEAR(dob)
ORDER BY birth_year;
```

3. How many patients share the same first name?

```
SELECT first_name, COUNT(*) AS name_count
FROM Patient
GROUP BY first_name
```

```
HAVING COUNT(*) > 1;
```

4. What percentage of patients have insurance information recorded?

```
SELECT (COUNT(insurance_info) * 100.0 / COUNT(*)) AS percentage
FROM Patient;
```

5. What is the most common city mentioned in the address field?

```
SELECT SUBSTRING_INDEX(address, ',', 1) AS city, COUNT(*) AS city_count
FROM Patient
GROUP BY city
ORDER BY city_count DESC
LIMIT 1;
```

6. How many patients are above the age of 50?

```
SELECT COUNT(*) AS patient_count
FROM Patient
WHERE YEAR(CURDATE()) - YEAR(dob) > 50;
```

7. What is the total number of patients without phone numbers?

```
SELECT COUNT(*) AS patient_count
FROM Patient
WHERE phone IS NULL;
```

8. Which age group has the highest number of patients?

```
SELECT CASE
         WHEN YEAR(CURDATE()) - YEAR(dob) BETWEEN 0 AND 18 THEN '0-18'
         WHEN YEAR(CURDATE()) - YEAR(dob) BETWEEN 19 AND 35 THEN '19-35'
         WHEN YEAR(CURDATE()) - YEAR(dob) BETWEEN 36 AND 50 THEN '36-50'
         ELSE '50+'
      END AS age_group, COUNT(*) AS patient_count
FROM Patient
GROUP BY age_group
ORDER BY patient_count DESC
LIMIT 1;
```

9. How many patients share the same last name?

```
SELECT last_name, COUNT(*) AS name_count
FROM Patient
GROUP BY last_name
HAVING COUNT(*) > 1;
```

10. What is the average number of characters in patient addresses?

```
SELECT AVG(CHAR_LENGTH(address)) AS avg_address_length
FROM Patient;
```

Inventory Table Questions:

1. Which inventory items have a quantity below a certain threshold (e.g., 10)?

```
SELECT item_name, quantity
FROM Inventory
WHERE quantity < 10;
```

2. How many items in the inventory are expiring in the next 30 days?

```
SELECT COUNT(*) AS expiring_soon_count
FROM Inventory
WHERE expiration_date BETWEEN CURDATE() AND DATE_ADD(CURDATE(), INTERVAL 30 DAY);
```

3. What is the most common inventory item name?

```
SELECT item_name, COUNT(*) AS item_count
FROM Inventory
GROUP BY item_name
ORDER BY item_count DESC
LIMIT 1;
```

4. What is the total quantity of all items in the inventory?

```
SELECT SUM(quantity) AS total_quantity
FROM Inventory;
```

5. How many items have expired to date?

```
SELECT COUNT(*) AS expired_items_count
FROM Inventory
WHERE expiration_date < CURDATE();
```

6. What is the maximum and minimum quantity recorded in the table?

```
SELECT MAX(quantity) AS max_quantity, MIN(quantity) AS min_quantity
FROM Inventory;
```

7. How many unique inventory item names exist?

```
SELECT COUNT(DISTINCT item_name) AS unique_item_count
FROM Inventory;
```

8. What is the average quantity of items in the inventory?

```
SELECT AVG(quantity) AS avg_quantity
FROM Inventory;
```

9. Which inventory items have an expiration date beyond 1 year?

```
SELECT item_name, expiration_date
FROM Inventory
WHERE expiration_date > DATE_ADD(CURDATE(), INTERVAL 1 YEAR);
```

10. How many items are marked as zero quantity?

```
SELECT COUNT(*) AS zero_quantity_count
FROM Inventory
WHERE quantity = 0;
```

Appointment Table Questions:

1. How many appointments are marked as "completed"?

```
SELECT COUNT(*) AS completed_appointments
FROM Appointment
WHERE status = 'completed';
```

2. What is the most common status among appointments?

```
SELECT status, COUNT(*) AS status_count
FROM Appointment
GROUP BY status
ORDER BY status_count DESC
LIMIT 1;
```

3. What is the total number of appointments scheduled for a specific date?

```
SELECT COUNT(*) AS appointments_on_specific_date
FROM Appointment
WHERE appointment_date = '2024-11-01';  -- Replace with the specific date
```

4. How many unique patient IDs are listed in the appointment table?

```
SELECT COUNT(DISTINCT patient_id) AS unique_patient_count
FROM Appointment;
```

5. How many appointments were scheduled in the past month?

```
SELECT COUNT(*) AS appointments_last_month
FROM Appointment
WHERE appointment_date BETWEEN DATE_SUB(CURDATE(), INTERVAL 1 MONTH) AND
CURDATE();
```

6. What is the frequency of appointments per doctor ID?

```
SELECT doctor_id, COUNT(*) AS appointment_count
FROM Appointment
GROUP BY doctor_id;
```

7. How many appointments have been scheduled but are yet to be completed?

```
SELECT COUNT(*) AS pending_appointments
FROM Appointment
WHERE status != 'completed';
```

8. What is the average time gap between consecutive appointments?

```
SELECT AVG(TIMESTAMPDIFF(MINUTE, LAG(appointment_date) OVER (ORDER BY
appointment_date), appointment_date)) AS avg_time_gap
FROM Appointment;
```

9. How many appointments have been canceled to date?

```
SELECT COUNT(*) AS canceled_appointments
FROM Appointment
WHERE status = 'canceled';
```

10. What is the total number of appointments for patients over the age of 50?

```
SELECT COUNT(*) AS appointments_for_patients_over_50
FROM Appointment a
JOIN Patient p ON a.patient_id = p.patient_id
WHERE TIMESTAMPDIFF(YEAR, p.dob, CURDATE()) > 50;
```

Billing Table Questions:

1. How many bills have been marked as "paid"?

```
SELECT COUNT(*) AS paid_bills
FROM Billing
WHERE payment_status = 'paid';
```

2. What is the total amount of revenue generated?

```
SELECT SUM(amount) AS total_revenue
FROM Billing;
```

3. What is the average billing amount per patient?

```
SELECT AVG(amount) AS average_billing_amount
FROM Billing;
```

4. How many bills have been generated in the last 30 days?

```
SELECT COUNT(*) AS bills_last_30_days
FROM Billing
WHERE billing_date BETWEEN DATE_SUB(CURDATE(), INTERVAL 30 DAY) AND
CURDATE();
```

5. What is the maximum billing amount recorded?

```
SELECT MAX(amount) AS max_billing_amount
FROM Billing;
```

6. How many patients have outstanding payment statuses?

```
SELECT COUNT(DISTINCT patient_id) AS patients_with_outstanding_payments
FROM Billing
WHERE payment_status != 'paid';
```

7. What is the frequency of billing per patient?

```
SELECT patient_id, COUNT(*) AS billing_frequency
FROM Billing
GROUP BY patient_id;
```

8. How many billing records are associated with patients above the age of 60?

```
SELECT COUNT(*) AS billing_records_above_60
FROM Billing b
JOIN Patient p ON b.patient_id = p.patient_id
WHERE TIMESTAMPDIFF(YEAR, p.dob, CURDATE()) > 60;
```

9. What is the distribution of payment statuses (e.g., "paid," "pending")?

```
SELECT payment_status, COUNT(*) AS status_count
FROM Billing
GROUP BY payment_status;
```

10. How many patients have multiple billing records?

```
SELECT patient_id, COUNT(*) AS billing_count
FROM Billing
GROUP BY patient_id
```

```
HAVING billing_count > 1;
```