Queries for Hospital Management System

1. List all patients currently admitted to the ICU

```
SELECT
   p.patient_id,
   p.name,
   p.age,
   p.disease,
   i.admission_date,
   DATEDIFF(CURRENT_DATE, i.admission_date) AS days_admitted
FROM
   PATIENT p
JOIN
   INPATIENT i ON p.patient_id = i.patient_id
JOIN
   WARD w ON i.ward_id = w.ward_id
WHERE
   w.ward_name = 'ICU'
   AND i.discharge_date IS NULL
ORDER BY
   i.admission_date;
```

2. Find the total revenue generated by each doctor

```
SELECT
   d.doctor_id,
   d.name AS doctor_name,
   d.specialization,
   COUNT(b.bill_id) AS total_bills,
   SUM(b.doctor_fee) AS total_doctor_fees,
   SUM(b.total_amount) AS total_revenue
FROM
   DOCTOR d
LEFT JOIN
   BILL b ON d.doctor_id = b.doctor_id
GROUP BY
   d.doctor_id, d.name, d.specialization
```

```
ORDER BY
    total_revenue DESC;
```

3. Find all patients who have been prescribed antibiotics

```
SELECT
    p.patient_id,
    p.name,
    pr.prescription_id,
    pr.medicine_name,
    pr.dosage,
    pr.prescribed_date,
    d.name AS doctor_name
FROM
    PATIENT p
JOIN
    PRESCRIPTION pr ON p.patient_id = pr.patient_id
JOIN
    DOCTOR d ON pr.doctor_id = d.doctor_id
WHERE
    pr.medicine_name LIKE '%amoxicillin%'
    OR pr.medicine_name LIKE '%antibiotic%'
    OR pr.medicine_name LIKE '%penicillin%'
    OR pr.medicine_name LIKE '%azithromycin%'
ORDER BY
    pr.prescribed_date DESC;
```

4. Calculate the average length of stay for discharged inpatients by ward type

```
SELECT
    w.ward_type,
    COUNT(i.patient_id) AS patient_count,
    ROUND(AVG(DATEDIFF(i.discharge_date, i.admission_date)), 1) AS
avg_stay_days,
    MIN(DATEDIFF(i.discharge_date, i.admission_date)) AS min_stay_days,
    MAX(DATEDIFF(i.discharge_date, i.admission_date)) AS max_stay_days
FROM
    INPATIENT i
```

```
JOIN
   WARD w ON i.ward_id = w.ward_id
WHERE
   i.discharge_date IS NOT NULL
GROUP BY
   w.ward_type
ORDER BY
   avg_stay_days DESC;
```

5. Find nurses who are working night shifts in critical care wards

```
SELECT
   n.nurse_id,
   n.name,
   n.qualification,
   n.shift,
   w.ward_name,
   w.ward_type
FROM
   NURSE n
JOIN
   WARD w ON n.ward_id = w.ward_id
WHERE
   n.shift = 'Night'
   AND (w.ward_type = 'Intensive Care' OR w.ward_type = 'Critical Care')
ORDER BY
   n.name;
```

6. Find all patients who have visited the hospital more than once

```
SELECT
   p.patient_id,
   p.name,
   p.disease,
   COUNT(pd.visit_date) AS visit_count
FROM
   PATIENT p
JOIN
   PATIENT_DOCTOR pd ON p.patient_id = pd.patient_id
GROUP BY
```

```
  p.patient_id, p.name, p.disease
HAVING
  COUNT(pd.visit_date) > 1
ORDER BY
  visit_count DESC;
```

7. Calculate the total cost of medical tests for each patient

```
SELECT
  p.patient_id,
  p.name,
  COUNT(mt.test_id) AS test_count,
  SUM(mt.cost) AS total_test_cost
FROM
  PATIENT p
LEFT JOIN
  MEDICAL_TEST mt ON p.patient_id = mt.patient_id
GROUP BY
  p.patient_id, p.name
ORDER BY
  total_test_cost DESC;
```

8. Find the occupancy rate of each ward

```
SELECT
  w.ward_id,
  w.ward_name,
  w.capacity AS total_beds,
  COUNT(i.patient_id) AS occupied_beds,
  w.capacity - COUNT(i.patient_id) AS available_beds,
  ROUND((COUNT(i.patient_id) * 100.0 / w.capacity), 2) AS
occupancy_percentage
FROM
  WARD w
LEFT JOIN
  INPATIENT i ON w.ward_id = i.ward_id AND i.discharge_date IS NULL
GROUP BY
  w.ward_id, w.ward_name, w.capacity
ORDER BY
  occupancy_percentage DESC;
```

9. Find doctors who have treated both inpatients and outpatients

```
SELECT
   d.doctor_id,
   d.name,
   d.specialization,
   COUNT(DISTINCT pd.patient_id) AS total_patients,
   COUNT(DISTINCT CASE WHEN i.patient_id IS NOT NULL THEN pd.patient_id
END) AS inpatient_count,
   COUNT(DISTINCT CASE WHEN o.patient_id IS NOT NULL THEN pd.patient_id
END) AS outpatient_count
FROM
   DOCTOR d
JOIN
   PATIENT_DOCTOR pd ON d.doctor_id = pd.doctor_id
LEFT JOIN
   INPATIENT i ON pd.patient_id = i.patient_id
LEFT JOIN
   OUTPATIENT o ON pd.patient_id = o.patient_id
GROUP BY
   d.doctor_id, d.name, d.specialization
HAVING
   COUNT(DISTINCT CASE WHEN i.patient_id IS NOT NULL THEN pd.patient_id
END) > 0
   AND COUNT(DISTINCT CASE WHEN o.patient_id IS NOT NULL THEN
pd.patient_id END) > 0
ORDER BY
   total_patients DESC;
```

10. Calculate average bill amount by patient type (inpatient vs outpatient)

```
SELECT
   CASE
      WHEN i.patient_id IS NOT NULL THEN 'Inpatient'
      ELSE 'Outpatient'
   END AS patient_type,
   COUNT(b.bill_id) AS bill_count,
   ROUND(AVG(b.doctor_fee), 2) AS avg_doctor_fee,
   ROUND(AVG(b.room_charge), 2) AS avg_room_charge,
   ROUND(AVG(b.test_charge), 2) AS avg_test_charge,
   ROUND(AVG(b.total_amount), 2) AS avg_total_amount
FROM
   BILL b
JOIN
   PATIENT p ON b.patient_id = p.patient_id
LEFT JOIN
   INPATIENT i ON p.patient_id = i.patient_id
LEFT JOIN
   OUTPATIENT o ON p.patient_id = o.patient_id
GROUP BY
   patient_type
ORDER BY
   avg_total_amount DESC;
```