

There is a Hospital in a city. For managing the whole hospital, there needs a database system. The database system is given below:

## Assumed Tables and Columns:

- **patients**

- (patient\_id, name, age, gender, address, phone, email, emergency\_contact, blood\_group, registration\_date)

- **doctors**

- (doctor\_id, name, specialization, phone, email, schedule, room\_number)

- **nurse**

- (nurse\_id, name, phone, email, department\_id, joining\_date, shift)

- **appointments**

- (appointment\_id, patient\_id, doctor\_id, appointment\_date, appointment\_time, status)

- **departments**

- (department\_id, name, head\_of\_department)

- **rooms**

- (room\_id, room\_number, room\_type, status)

- **admissions**

- (admission\_id, patient\_id, admit\_date, discharge\_date, room\_id, diagnosis)

- **bills**

- (bill\_id, patient\_id, admission\_id, amount, payment\_status, payment\_date, payment\_method)

- **medications**

- (med\_id, name, description, dosage\_form, price)

- **prescriptions**

- (prescription\_id, patient\_id, doctor\_id, date\_app, symptoms, diagnosis, medicines, tests\_recommended)

- **tests**

- (test\_id, patient\_id, test\_type, test\_date, result, doctor\_id)

- **staffs**

- (staff\_id, name, role, department, phone, joining\_date)

- **inventory**

- (item\_id, name, quantity, unit, last\_updated)

**All the codes of all tables creation and data insertion is given below for practice:**

```
CREATE TABLE doctors (  
    doctor_id NUMBER PRIMARY KEY,  
    name VARCHAR2(100) NOT NULL,  
    specialization VARCHAR2(100),  
    phone VARCHAR2(20) UNIQUE,  
    email VARCHAR2(100),  
    schedule VARCHAR2(100),  
    room_number VARCHAR2(10)  
)
```

```
CREATE TABLE rooms (  
    room_id NUMBER PRIMARY KEY,
```

```

        room_number VARCHAR2(10) UNIQUE NOT NULL,
        room_type VARCHAR2(50) CHECK (room_type IN ('General', 'ICU',
'Cabin')),
        status VARCHAR2(20) CHECK (status IN ('Available', 'Occupied'))
    )

```

```

CREATE TABLE admissions (
    admission_id NUMBER PRIMARY KEY,
    patient_id NUMBER NOT NULL,
    admit_date DATE NOT NULL,
    discharge_date DATE,
    room_id NUMBER,
    diagnosis VARCHAR2(200),
    FOREIGN KEY (patient_id) REFERENCES patients(patient_id),
    FOREIGN KEY (room_id) REFERENCES rooms(room_id)
)

```

```

CREATE TABLE appointments (
    appointment_id NUMBER PRIMARY KEY,
    patient_id NUMBER NOT NULL,
    doctor_id NUMBER NOT NULL,
    appointment_date DATE NOT NULL,
    appointment_time VARCHAR2(10),
    status VARCHAR2(20) CHECK (status IN ('Pending', 'Completed',
'Cancelled')),
    FOREIGN KEY (patient_id) REFERENCES patients(patient_id),
    FOREIGN KEY (doctor_id) REFERENCES doctors(doctor_id)
)

```

```

CREATE TABLE bills (
    bill_id NUMBER,
    patient_id NUMBER,
    admission_id NUMBER,
    amount NUMBER(10,2) ,
    payment_status VARCHAR2(20),
    payment_date DATE,
    payment_method VARCHAR2(50),
    PRIMARY KEY(bill_id),
    FOREIGN KEY (patient_id) REFERENCES patients(patient_id),
    FOREIGN KEY (admission_id) REFERENCES admissions(admission_id)
)

```

```
CREATE TABLE medications (  
    med_id NUMBER PRIMARY KEY,  
    name VARCHAR2(100) NOT NULL,  
    description VARCHAR2(200),  
    dosage_form VARCHAR2(50),  
    price NUMBER(10,2) CHECK (price >= 0)  
)
```

```
CREATE TABLE tests (  
    test_id NUMBER PRIMARY KEY,  
    patient_id NUMBER NOT NULL,  
    test_type VARCHAR2(100) NOT NULL,  
    test_date DATE DEFAULT SYSDATE,  
    result VARCHAR2(200),  
    doctor_id NUMBER,  
    FOREIGN KEY (patient_id) REFERENCES patients(patient_id),  
    FOREIGN KEY (doctor_id) REFERENCES doctors(doctor_id)  
)
```

```
CREATE TABLE staffs (  
    staff_id NUMBER PRIMARY KEY,  
    name VARCHAR2(100) NOT NULL,  
    role VARCHAR2(50),  
    department VARCHAR2(100),  
    phone VARCHAR2(20),  
    joining_date DATE  
)
```

```
CREATE TABLE departments (  
    department_id NUMBER PRIMARY KEY,  
    name VARCHAR2(100) NOT NULL UNIQUE,  
    head_of_department VARCHAR2(100)
```

)

```
CREATE TABLE inventory (  
    item_id NUMBER PRIMARY KEY,  
    name VARCHAR2(100) NOT NULL,  
    quantity NUMBER CHECK (quantity >= 0),  
    unit VARCHAR2(20),  
    last_updated DATE DEFAULT SYSDATE  
)
```

```
CREATE TABLE nurse (  
    nurse_id NUMBER PRIMARY KEY,  
    name VARCHAR2(100) NOT NULL,  
    phone VARCHAR2(20) UNIQUE,  
    email VARCHAR2(100),  
    department_id NUMBER NOT NULL,  
    joining_date DATE,  
    shift VARCHAR2(50) CHECK (shift IN ('Morning', 'Evening', 'Night')),  
    FOREIGN KEY (department_id) REFERENCES departments(department_id)  
)
```

```
CREATE TABLE patients (  
    patient_id NUMBER PRIMARY KEY,  
    name VARCHAR2(100) NOT NULL,  
    age NUMBER CHECK (age > 0),  
    gender VARCHAR2(10) CHECK (gender IN ('Male', 'Female')),  
    address VARCHAR2(200),  
    phone VARCHAR2(20) UNIQUE,  
    email VARCHAR2(100),  
    emergency_contact VARCHAR2(100),  
    blood_group VARCHAR2(5),  
    registration_date DATE DEFAULT SYSDATE  
)
```

```
CREATE TABLE prescriptions (  
    prescription_id NUMBER PRIMARY KEY,  
    patient_id NUMBER NOT NULL,  
    doctor_id NUMBER NOT NULL,  
    date_app DATE,
```

```
symptoms VARCHAR2(200),  
diagnosis VARCHAR2(200),  
medicines VARCHAR2(300),  
tests_recommended VARCHAR2(300)  
)
```

```
INSERT INTO doctors VALUES (1, 'Dr. Ahsan Ullah', 'Cardiology', '01810000001',  
'ahsan@hospital.com', 'Sun-Thu 9AM-5PM', 'R101');  
INSERT INTO doctors VALUES (2, 'Dr. Nazma Akter', 'Neurology', '01810000002',  
'nazma@hospital.com', 'Sun-Thu 10AM-4PM', 'R102');  
INSERT INTO doctors VALUES (3, 'Dr. Rafiq Hasan', 'Orthopedics', '01810000003',  
'rafiq@hospital.com', 'Sun-Thu 11AM-6PM', 'R103');  
INSERT INTO doctors VALUES (4, 'Dr. Salma Khatun', 'Dermatology', '01810000004',  
'salma@hospital.com', 'Sun-Thu 9AM-3PM', 'R104');  
INSERT INTO doctors VALUES (5, 'Dr. Tanvir Alam', 'Gastroenterology',  
'01810000005', 'tanvir@hospital.com', 'Sun-Thu 10AM-5PM', 'R105');  
INSERT INTO doctors VALUES (6, 'Dr. Laila Noor', 'Pediatrics', '01810000006',  
'laila@hospital.com', 'Sun-Thu 8AM-2PM', 'R106');  
INSERT INTO doctors VALUES (7, 'Dr. Imran Hossain', 'Oncology', '01810000007',  
'imran@hospital.com', 'Sun-Thu 12PM-6PM', 'R107');  
INSERT INTO doctors VALUES (8, 'Dr. Mahbuba Sultana', 'ENT', '01810000008',  
'mahbuba@hospital.com', 'Sun-Thu 9AM-1PM', 'R108');  
INSERT INTO doctors VALUES (9, 'Dr. Khaled Mahmud', 'Ophthalmology',  
'01810000009', 'khaled@hospital.com', 'Sun-Thu 2PM-8PM', 'R109');  
INSERT INTO doctors VALUES (10, 'Dr. Farzana Jahan', 'Psychiatry', '01810000010',  
'farzana@hospital.com', 'Sun-Thu 10AM-3PM', 'R110');
```

```
INSERT INTO rooms VALUES (1, 'RM101', 'General', 'Available');  
INSERT INTO rooms VALUES (2, 'RM102', 'General', 'Occupied');  
INSERT INTO rooms VALUES (3, 'RM103', 'ICU', 'Available');  
INSERT INTO rooms VALUES (4, 'RM104', 'ICU', 'Occupied');  
INSERT INTO rooms VALUES (5, 'RM105', 'Cabin', 'Available');  
INSERT INTO rooms VALUES (6, 'RM106', 'Cabin', 'Occupied');  
INSERT INTO rooms VALUES (7, 'RM107', 'General', 'Available');  
INSERT INTO rooms VALUES (8, 'RM108', 'General', 'Available');  
INSERT INTO rooms VALUES (9, 'RM109', 'ICU', 'Occupied');  
INSERT INTO rooms VALUES (10, 'RM110', 'Cabin', 'Available');
```

```

INSERT INTO departments VALUES (1, 'Cardiology', 'Dr. Ahsan Ullah');
INSERT INTO departments VALUES (2, 'Neurology', 'Dr. Nazma Akter');
INSERT INTO departments VALUES (3, 'Orthopedics', 'Dr. Rafiq Hasan');
INSERT INTO departments VALUES (4, 'Dermatology', 'Dr. Salma Khatun');
INSERT INTO departments VALUES (5, 'Gastroenterology', 'Dr. Tanvir Alam');
INSERT INTO departments VALUES (6, 'Pediatrics', 'Dr. Laila Noor');
INSERT INTO departments VALUES (7, 'Oncology', 'Dr. Imran Hossain');
INSERT INTO departments VALUES (8, 'ENT', 'Dr. Mahbuba Sultana');
INSERT INTO departments VALUES (9, 'Ophthalmology', 'Dr. Khaled Mahmud');
INSERT INTO departments VALUES (10, 'Psychiatry', 'Dr. Farzana Jahan');

```

```

INSERT INTO nurse VALUES (1, 'Nusrat Jahan', '01910000001', 'nusrat@hospital.com',
1, TO_DATE('2022-01-01', 'YYYY-MM-DD'), 'Morning');
INSERT INTO nurse VALUES (2, 'Shirin Akter', '01910000002', 'shirin@hospital.com',
2, TO_DATE('2021-12-15', 'YYYY-MM-DD'), 'Evening');
INSERT INTO nurse VALUES (3, 'Maliha Khatun', '01910000003',
'maliha@hospital.com', 3, TO_DATE('2020-11-20', 'YYYY-MM-DD'), 'Night');
INSERT INTO nurse VALUES (4, 'Rafiq Sultana', '01910000004',
'rafiqa@hospital.com', 4, TO_DATE('2023-03-01', 'YYYY-MM-DD'), 'Morning');
INSERT INTO nurse VALUES (5, 'Farzana Ahmed', '01910000005',
'farzana@hospital.com', 5, TO_DATE('2019-06-10', 'YYYY-MM-DD'), 'Night');
INSERT INTO nurse VALUES (6, 'Kazi Munni', '01910000006', 'munni@hospital.com',
1, TO_DATE('2022-07-05', 'YYYY-MM-DD'), 'Evening');
INSERT INTO nurse VALUES (7, 'Shamima Nasrin', '01910000007',
'shamima@hospital.com', 2, TO_DATE('2020-10-12', 'YYYY-MM-DD'), 'Night');
INSERT INTO nurse VALUES (8, 'Jui Sultana', '01910000008', 'jui@hospital.com', 3,
TO_DATE('2021-01-18', 'YYYY-MM-DD'), 'Morning');
INSERT INTO nurse VALUES (9, 'Rumana Akter', '01910000009',
'rumana@hospital.com', 4, TO_DATE('2022-04-22', 'YYYY-MM-DD'), 'Evening');
INSERT INTO nurse VALUES (10, 'Mahmuda Khatun', '01910000010',
'mahmuda@hospital.com', 5, TO_DATE('2023-08-30', 'YYYY-MM-DD'), 'Night');

```

```

INSERT INTO appointments VALUES (1, 1, 1, TO_DATE('2024-05-01', 'YYYY-MM-DD'),
'10:00 AM', 'Completed');

```

```
INSERT INTO appointments VALUES (2, 2, 2, TO_DATE('2024-05-02','YYYY-MM-DD'),
'11:30 AM', 'Completed');
INSERT INTO appointments VALUES (3, 3, 3, TO_DATE('2024-05-03','YYYY-MM-DD'),
'12:00 PM', 'Pending');
INSERT INTO appointments VALUES (4, 4, 4, TO_DATE('2024-05-04','YYYY-MM-DD'),
'02:00 PM', 'Cancelled');
INSERT INTO appointments VALUES (5, 5, 5, TO_DATE('2024-05-05','YYYY-MM-DD'),
'09:30 AM', 'Completed');
INSERT INTO appointments VALUES (6, 6, 1, TO_DATE('2024-05-06','YYYY-MM-DD'),
'11:00 AM', 'Completed');
INSERT INTO appointments VALUES (7, 7, 2, TO_DATE('2024-05-07','YYYY-MM-DD'),
'10:30 AM', 'Pending');
INSERT INTO appointments VALUES (8, 8, 3, TO_DATE('2024-05-08','YYYY-MM-DD'),
'01:00 PM', 'Completed');
INSERT INTO appointments VALUES (9, 9, 4, TO_DATE('2024-05-09','YYYY-MM-DD'),
'03:00 PM', 'Cancelled');
INSERT INTO appointments VALUES (10, 10, 5,
TO_DATE('2024-05-10','YYYY-MM-DD'), '09:00 AM', 'Completed');
```

```
INSERT INTO patients VALUES (1, 'Rahim Uddin', 'Male', TO_DATE('1985-06-15',
'YYYY-MM-DD'), '01711112222', 'Dhanmondi, Dhaka');
INSERT INTO patients VALUES (2, 'Shila Akter', 'Female', TO_DATE('1992-11-03',
'YYYY-MM-DD'), '01855556666', 'Uttara, Dhaka');
INSERT INTO patients VALUES (3, 'Karim Hossain', 'Male', TO_DATE('1978-02-21',
'YYYY-MM-DD'), '01688889999', 'Mohakhali, Dhaka');
INSERT INTO patients VALUES (4, 'Mita Sultana', 'Female', TO_DATE('1989-07-12',
'YYYY-MM-DD'), '01933334444', 'Mirpur, Dhaka');
INSERT INTO patients VALUES (5, 'Nasir Khan', 'Male', TO_DATE('1995-01-05',
'YYYY-MM-DD'), '01777778888', 'Banani, Dhaka');
INSERT INTO patients VALUES (6, 'Rina Begum', 'Female', TO_DATE('1983-03-18',
'YYYY-MM-DD'), '01812345678', 'Gulshan, Dhaka');
INSERT INTO patients VALUES (7, 'Sajjad Hossain', 'Male', TO_DATE('1990-09-30',
'YYYY-MM-DD'), '01623456789', 'Badda, Dhaka');
INSERT INTO patients VALUES (8, 'Mousumi Akter', 'Female', TO_DATE('2000-04-25',
'YYYY-MM-DD'), '01987654321', 'Tejgaon, Dhaka');
INSERT INTO patients VALUES (9, 'Jamal Uddin', 'Male', TO_DATE('1965-12-10',
'YYYY-MM-DD'), '01712344321', 'Shyamoli, Dhaka');
INSERT INTO patients VALUES (10, 'Sharmin Jahan', 'Female',
TO_DATE('1998-08-08', 'YYYY-MM-DD'), '01876543210', 'Rampura, Dhaka');
```



```
INSERT INTO inventory VALUES (1, 'Syringe', 100, 'pcs',
TO_DATE('2024-01-01','YYYY-MM-DD'));
INSERT INTO inventory VALUES (2, 'Gloves', 500, 'pairs',
TO_DATE('2024-01-02','YYYY-MM-DD'));
INSERT INTO inventory VALUES (3, 'Saline', 200, 'bottles',
TO_DATE('2024-01-03','YYYY-MM-DD'));
INSERT INTO inventory VALUES (4, 'Face Mask', 1000, 'pcs',
TO_DATE('2024-01-04','YYYY-MM-DD'));
INSERT INTO inventory VALUES (5, 'IV Stand', 20, 'units',
TO_DATE('2024-01-05','YYYY-MM-DD'));
INSERT INTO inventory VALUES (6, 'Wheelchair', 10, 'units',
TO_DATE('2024-01-06','YYYY-MM-DD'));
INSERT INTO inventory VALUES (7, 'Bed Sheet', 150, 'pcs',
TO_DATE('2024-01-07','YYYY-MM-DD'));
INSERT INTO inventory VALUES (8, 'Bandage', 300, 'rolls',
TO_DATE('2024-01-08','YYYY-MM-DD'));
INSERT INTO inventory VALUES (9, 'Stethoscope', 50, 'pcs',
TO_DATE('2024-01-09','YYYY-MM-DD'));
INSERT INTO inventory VALUES (10, 'Oxygen Cylinder', 25, 'units',
TO_DATE('2024-01-10','YYYY-MM-DD'));
```

```
INSERT INTO staffs VALUES (1, 'Rokeya Sultana', 'Receptionist', 'Admin',
'01711111111', TO_DATE('2019-01-01','YYYY-MM-DD'));
INSERT INTO staffs VALUES (2, 'Mizanur Rahman', 'Cleaner', 'Support', '01711111112',
TO_DATE('2020-02-02','YYYY-MM-DD'));
INSERT INTO staffs VALUES (3, 'Shamima Nasrin', 'Accountant', 'Accounts',
'01711111113', TO_DATE('2018-03-03','YYYY-MM-DD'));
INSERT INTO staffs VALUES (4, 'Kamal Uddin', 'Security', 'Security', '01711111114',
TO_DATE('2017-04-04','YYYY-MM-DD'));
INSERT INTO staffs VALUES (5, 'Hasan Ali', 'Driver', 'Transport', '01711111115',
TO_DATE('2021-05-05','YYYY-MM-DD'));
INSERT INTO staffs VALUES (6, 'Fatema Akter', 'Lab Assistant', 'Lab', '01711111116',
TO_DATE('2020-06-06','YYYY-MM-DD'));
```

```
INSERT INTO staffs VALUES (7, 'Rashidul Hasan', 'Technician', 'OT', '01711111117',  
TO_DATE('2019-07-07','YYYY-MM-DD'));  
INSERT INTO staffs VALUES (8, 'Nasima Akter', 'Receptionist', 'Admin', '01711111118',  
TO_DATE('2022-08-08','YYYY-MM-DD'));  
INSERT INTO staffs VALUES (9, 'Mehedi Hasan', 'Accountant', 'Accounts',  
'01711111119', TO_DATE('2016-09-09','YYYY-MM-DD'));  
INSERT INTO staffs VALUES (10, 'Shila Khatun', 'Cleaner', 'Support', '01711111120',  
TO_DATE('2023-10-10','YYYY-MM-DD'));
```

```
INSERT INTO tests VALUES (1, 1, 'Blood Test',  
TO_DATE('2024-04-01','YYYY-MM-DD'), 'Normal', 1);  
INSERT INTO tests VALUES (2, 2, 'X-Ray', TO_DATE('2024-04-02','YYYY-MM-DD'),  
'Clear', 2); INSERT INTO tests VALUES (3, 3, 'MRI',  
TO_DATE('2024-04-03','YYYY-MM-DD'), 'Disc bulge', 3);  
INSERT INTO tests VALUES (4, 4, 'CT Scan', TO_DATE('2024-04-04','YYYY-MM-DD'),  
'Appendix inflamed', 4);  
INSERT INTO tests VALUES (5, 5, 'Ultrasound',  
TO_DATE('2024-04-05','YYYY-MM-DD'), 'Gallstone detected', 5);  
INSERT INTO tests VALUES (6, 6, 'ECG', TO_DATE('2024-04-06','YYYY-MM-DD'),  
'Normal rhythm', 1);  
INSERT INTO tests VALUES (7, 7, 'CBC', TO_DATE('2024-04-07','YYYY-MM-DD'), 'Low  
Hemoglobin', 2);  
INSERT INTO tests VALUES (8, 8, 'Allergy Test',  
TO_DATE('2024-04-08','YYYY-MM-DD'), 'Dust Allergy', 3);  
INSERT INTO tests VALUES (9, 9, 'Blood Sugar',  
TO_DATE('2024-04-09','YYYY-MM-DD'), 'High', 4);  
INSERT INTO tests VALUES (10, 10, 'Urine Test',  
TO_DATE('2024-04-10','YYYY-MM-DD'), 'Normal', 5);
```

```
INSERT INTO medications VALUES (1, 'Paracetamol', 'Used for fever and pain',  
'Tablet', 1.5); INSERT INTO medications VALUES (2, 'Amoxicillin', 'Antibiotic', 'Capsule',  
3.0);  
INSERT INTO medications VALUES (3, 'Ibuprofen', 'Anti-inflammatory', 'Tablet', 2.0);  
INSERT INTO medications VALUES (4, 'Omeprazole', 'For acidity', 'Capsule', 1.8);  
INSERT INTO medications VALUES (5, 'Metformin', 'For diabetes', 'Tablet', 2.5);  
INSERT INTO medications VALUES (6, 'Cough Syrup', 'Used for cough', 'Syrup', 4.0)
```

```
; INSERT INTO medications VALUES (7, 'ORS', 'Rehydration solution', 'Packet', 0.5);  
INSERT INTO medications VALUES (8, 'Vitamin D', 'Supplement', 'Tablet', 2.2);  
INSERT INTO medications VALUES (9, 'Antihistamine', 'For allergies', 'Tablet', 1.0);  
INSERT INTO medications VALUES (10, 'Insulin', 'Hormone for diabetes', 'Injection',  
5.0);
```

```
INSERT INTO prescriptions VALUES (1, 1, 1, TO_DATE('2024-04-01','YYYY-MM-DD'),  
'Fever', 'Viral Infection', 'Paracetamol, ORS', 'Blood Test');  
INSERT INTO prescriptions VALUES (2, 2, 2, TO_DATE('2024-04-02','YYYY-MM-DD'),  
'Cough', 'Bronchitis', 'Cough Syrup, Steam Inhalation', 'X-Ray');  
INSERT INTO prescriptions VALUES (3, 3, 3, TO_DATE('2024-04-03','YYYY-MM-DD'),  
'Headache', 'Migraine', 'Painkiller, Sleep Aid', 'CT Scan');  
INSERT INTO prescriptions VALUES (4, 4, 4, TO_DATE('2024-04-04','YYYY-MM-DD'),  
'Stomach Pain', 'Gastric Ulcer', 'Antacids', 'Ultrasound');  
INSERT INTO prescriptions VALUES (5, 5, 5, TO_DATE('2024-04-05','YYYY-MM-DD'),  
'Joint Pain', 'Arthritis', 'Ibuprofen, Gel', 'Blood Test');  
INSERT INTO prescriptions VALUES (6, 6, 1, TO_DATE('2024-04-06','YYYY-MM-DD'),  
'Back Pain', 'Muscle Strain', 'Painkillers, Heat Pad', "");  
INSERT INTO prescriptions VALUES (7, 7, 2, TO_DATE('2024-04-07','YYYY-MM-DD'),  
'Fatigue', 'Anemia', 'Iron Supplements', 'CBC');  
INSERT INTO prescriptions VALUES (8, 8, 3, TO_DATE('2024-04-08','YYYY-MM-DD'),  
'Skin Rash', 'Allergy', 'Antihistamines', "");  
INSERT INTO prescriptions VALUES (9, 9, 4, TO_DATE('2024-04-09','YYYY-MM-DD'),  
'High Sugar', 'Diabetes', 'Metformin, Insulin', 'Blood Sugar');  
INSERT INTO prescriptions VALUES (10, 10, 5,  
TO_DATE('2024-04-10','YYYY-MM-DD'), 'Nausea', 'Indigestion', 'Antiemetics', "");
```

```
INSERT INTO admissions VALUES (1, 1, TO_DATE('2024-01-01','YYYY-MM-DD'),  
TO_DATE('2024-01-05','YYYY-MM-DD'), 1, 'Fever and dehydration');  
INSERT INTO admissions VALUES (2, 2, TO_DATE('2024-01-06','YYYY-MM-DD'),  
TO_DATE('2024-01-10','YYYY-MM-DD'), 2, 'Fractured arm');  
INSERT INTO admissions VALUES (3, 3, TO_DATE('2024-01-10','YYYY-MM-DD'),  
TO_DATE('2024-01-13','YYYY-MM-DD'), 3, 'Migraine observation');  
INSERT INTO admissions VALUES (4, 4, TO_DATE('2024-01-14','YYYY-MM-DD'),  
TO_DATE('2024-01-17','YYYY-MM-DD'), 4, 'Post-surgery recovery');
```

```
INSERT INTO admissions VALUES (5, 5, TO_DATE('2024-01-18','YYYY-MM-DD'),
TO_DATE('2024-01-21','YYYY-MM-DD'), 5, 'Appendicitis');
INSERT INTO admissions VALUES (6, 6, TO_DATE('2024-01-22','YYYY-MM-DD'),
TO_DATE('2024-01-25','YYYY-MM-DD'), 1, 'Chest infection');
INSERT INTO admissions VALUES (7, 7, TO_DATE('2024-01-26','YYYY-MM-DD'),
TO_DATE('2024-01-28','YYYY-MM-DD'), 2, 'Asthma attack');
INSERT INTO admissions VALUES (8, 8, TO_DATE('2024-01-29','YYYY-MM-DD'),
TO_DATE('2024-02-01','YYYY-MM-DD'), 3, 'Minor burn treatment');
INSERT INTO admissions VALUES (9, 9, TO_DATE('2024-02-02','YYYY-MM-DD'),
TO_DATE('2024-02-05','YYYY-MM-DD'), 4, 'Dengue fever');
INSERT INTO admissions VALUES (10, 10, TO_DATE('2024-02-06','YYYY-MM-DD'),
TO_DATE('2024-02-09','YYYY-MM-DD'), 5, 'High blood pressure monitoring');
```

```
INSERT INTO bills VALUES (1, 1, 1, 1500.00, 'Paid',
TO_DATE('2024-01-06','YYYY-MM-DD'), 'Credit Card');
INSERT INTO bills VALUES (2, 2, 2, 2300.50, 'Paid',
TO_DATE('2024-01-11','YYYY-MM-DD'), 'Cash');
INSERT INTO bills VALUES (3, 3, 3, 1850.00, 'Unpaid',
TO_DATE('2024-01-13','YYYY-MM-DD'), 'Cash');
INSERT INTO bills VALUES (4, 4, 4, 3200.00, 'Partial',
TO_DATE('2024-01-17','YYYY-MM-DD'), 'Bank Transfer');
INSERT INTO bills VALUES (5, 5, 5, 975.75, 'Paid',
TO_DATE('2024-01-21','YYYY-MM-DD'), 'Credit Card');
INSERT INTO bills VALUES (6, 6, 6, 1450.25, 'Unpaid',
TO_DATE('2024-01-25','YYYY-MM-DD'), 'Mobile Payment');
INSERT INTO bills VALUES (7, 7, 7, 1985.00, 'Paid',
TO_DATE('2024-01-28','YYYY-MM-DD'), 'Cash');
INSERT INTO bills VALUES (8, 8, 8, 2100.00, 'Partial',
TO_DATE('2024-02-01','YYYY-MM-DD'), 'Mobile Payment');
INSERT INTO bills VALUES (9, 9, 9, 1600.00, 'Paid',
TO_DATE('2024-02-05','YYYY-MM-DD'), 'Credit Card');
INSERT INTO bills VALUES (10, 10, 10, 2450.00, 'Paid',
TO_DATE('2024-02-09','YYYY-MM-DD'), 'Cash');
```

## SQL Query Questions set:

**1. Retrieve the names of all patients.**

```
SELECT patient_name FROM Patient;
```

**2. Find the names and ages of all doctors.**

```
SELECT doctor_name, age FROM Doctor;
```

**3. List all patients who are older than 40.**

```
SELECT * FROM Patient WHERE age > 40;
```

**4. Show all departments in alphabetical order.**

```
SELECT * FROM Department ORDER BY  
department_name ASC;
```

**5. Count the total number of nurses.**

```
SELECT COUNT(*) FROM Nurse;
```

**6. Find all patients whose names start with 'A'.**

```
SELECT * FROM Patient WHERE patient_name  
LIKE 'A%';
```

**7. Retrieve doctors whose name ends with 'n'.**

```
SELECT * FROM Doctor WHERE doctor_name  
LIKE '%n';
```

**8. Display nurse names in uppercase.**

```
SELECT UPPER(nurse_name) FROM Nurse;
```

**9. Show patient names with 'Mr.' prefix.**

```
SELECT CONCAT('Mr. ', patient_name) FROM  
Patient;
```

**10. Get the length of each doctor's name.**

```
SELECT doctor_name, LENGTH(doctor_name)  
AS name_length FROM Doctor;
```

**11. List all appointments with patient and doctor names.**

```
SELECT A.appointment_id, P.patient_name,  
D.doctor_name  
FROM Appointment A
```

```
JOIN Patient P ON A.patient_id =  
P.patient_id  
JOIN Doctor D ON A.doctor_id =  
D.doctor_id;
```

**12. Get ward details and their respective department names.**

```
SELECT W.*, D.department_name  
FROM Ward W  
JOIN Department D ON W.department_id =  
D.department_id;
```

**13. Display prescriptions with corresponding patient names.**

```
SELECT Pr.prescription_id,  
Pa.patient_name  
FROM Prescription Pr
```



```
JOIN Patient Pa ON Pr.patient_id =  
Pa.patient_id;
```

**14. Get all patients who were admitted and their room numbers.**

```
SELECT Pa.patient_name, Ad.room_number  
FROM Admission Ad  
JOIN Patient Pa ON Ad.patient_id =  
Pa.patient_id;
```

**15. Use NATURAL JOIN to list all patient-prescription details.**

```
SELECT * FROM Patient NATURAL JOIN  
Prescription;
```

**16. Find the average age of patients.**

```
SELECT AVG(age) FROM Patient;
```

**17. Count how many departments there are.**

```
SELECT COUNT(*) FROM Department;
```

**18. Get the maximum age of doctors.**

```
SELECT MAX(age) FROM Doctor;
```

**19. Find the number of prescriptions for each patient.**

```
SELECT patient_id, COUNT(*) AS  
total_prescriptions  
FROM Prescription  
GROUP BY patient_id;
```

**20. List departments having more than 2 wards.**

```
SELECT department_id, COUNT(*) AS  
total_wards  
FROM Ward  
GROUP BY department_id  
HAVING COUNT(*) > 2;
```

**21. Add an email column to the Patient table.**

```
ALTER TABLE Patient ADD email  
VARCHAR(100);
```

**22. Drop the email column from the Patient table.**

```
ALTER TABLE Patient DROP COLUMN email;
```

**23. Rename the Nurse table to NursingStaff.**

```
ALTER TABLE Nurse RENAME TO NursingStaff;
```

**24. Drop the Doctor table.**

```
DROP TABLE Doctor;
```

**25. Change the datatype of age in Patient to SMALLINT.**

```
ALTER TABLE Patient MODIFY age SMALLINT;
```

**26. Find patients who have more prescriptions than the average.**

```
SELECT patient_id
FROM Prescription
GROUP BY patient_id
HAVING COUNT(*) > (
    SELECT AVG(pres_count) FROM (
        SELECT COUNT(*) AS pres_count
FROM Prescription GROUP BY patient_id
    ) AS temp
);
```

**27. Get doctors who have never had an appointment.**

```
SELECT doctor_name FROM Doctor
WHERE doctor_id NOT IN (
    SELECT doctor_id FROM Appointment
);
```

**28. List patients who have been prescribed 'Paracetamol'.**

```
SELECT patient_name FROM Patient
WHERE patient_id IN (
    SELECT patient_id FROM Prescription
WHERE medicine = 'Paracetamol'
);
```

**29. Retrieve wards that belong to departments with more than 3 doctors.**

```
SELECT * FROM Ward
WHERE department_id IN (
    SELECT department_id FROM Doctor
GROUP BY department_id
HAVING COUNT(*) > 3
);
```

**30. Get the names of patients admitted in the same room as patient\_id = 1.**

```
SELECT patient_name FROM Patient
WHERE patient_id IN (
    SELECT patient_id FROM Admission
    WHERE room_number = (
        SELECT room_number FROM Admission
WHERE patient_id = 1
    )
);
```

**31. Get patients who have both appointments and prescriptions.**

```
SELECT patient_id FROM Appointment
INTERSECT
SELECT patient_id FROM Prescription;
```

**32. Find patients who have appointments but no prescriptions.**

```
SELECT patient_id FROM Appointment
```

EXCEPT

SELECT patient\_id FROM Prescription;

**33. List all patients who have appointments or prescriptions.**

SELECT patient\_id FROM Appointment

UNION

SELECT patient\_id FROM Prescription;

**34. Show doctors who are either surgeons or pediatricians.**

SELECT \* FROM Doctor

WHERE specialty = 'Surgeon'

UNION

SELECT \* FROM Doctor

WHERE specialty = 'Pediatrician';

**35. Get departments with no wards.**



```
SELECT department_id FROM Department  
EXCEPT  
SELECT department_id FROM Ward;
```

**36. Find patients who don't have a phone number.**

```
SELECT * FROM Patient WHERE phone IS  
NULL;
```

**37. Retrieve all patients who provided their address.**

```
SELECT * FROM Patient WHERE address IS  
NOT NULL;
```

**38. Update all null phone numbers to 'N/A'.**

```
UPDATE Patient SET phone = 'N/A' WHERE  
phone IS NULL;
```

**39. Insert a patient with some null fields.**

```
INSERT INTO Patient(patient_id,  
patient_name, age)  
VALUES (101, 'Niharika', 30);
```

**40. Check if any doctor has a null specialty.**

```
SELECT * FROM Doctor WHERE specialty IS  
NULL;
```

**41. List doctors who have appointments.**

```
SELECT * FROM Doctor D
```

```
WHERE EXISTS (  
    SELECT 1 FROM Appointment A WHERE  
    A.doctor_id = D.doctor_id  
);
```

**42. Get nurses who are not assigned to any ward.**

```
SELECT * FROM Nurse N  
WHERE NOT EXISTS (  
    SELECT 1 FROM Ward W WHERE W.nurse_id  
    = N.nurse_id  
);
```

**43. Find doctors older than all nurses.**

```
SELECT * FROM Doctor  
WHERE age > ALL (  
    SELECT age FROM Nurse  
);
```

**44. Find patients who are older than some nurses.**

```
SELECT * FROM Patient
WHERE age > SOME (
    SELECT age FROM Nurse
);
```

**45. Get departments that have at least one doctor.**

```
SELECT * FROM Department D
WHERE EXISTS (
    SELECT 1 FROM Doctor Doc WHERE
    Doc.department_id = D.department_id
);
```

**46. Insert a new doctor.**

```
INSERT INTO Doctor (doctor_id,  
doctor_name, age, specialty,  
department_id)  
VALUES (10, 'Dr. Zaman', 45,  
'Neurologist', 3);
```

**47. Delete all appointments before 2024.**

```
DELETE FROM Appointment  
WHERE appointment_date < '2024-01-01';
```

**48. Update all doctor salaries by 10%.**

```
UPDATE Doctor  
SET salary = salary * 1.10;
```

**49. Insert a new department.**

```
INSERT INTO Department (department_id,  
department_name)  
VALUES (6, 'Oncology');
```

**50. Delete patients who were never admitted.**

```
DELETE FROM Patient  
WHERE patient_id NOT IN (SELECT  
patient_id FROM Admission);
```

**51. List the second oldest patient.**

```
SELECT * FROM Patient  
WHERE age = (  
    SELECT MAX(age) FROM Patient  
    WHERE age < (SELECT MAX(age) FROM  
Patient)  
);
```

**52. Get average age of doctors by department.**

```
SELECT department_id, AVG(age) FROM  
Doctor  
GROUP BY department_id;
```

**53. Retrieve doctors whose salary is above department average.**

```
SELECT * FROM Doctor D  
WHERE salary > (  
    SELECT AVG(salary) FROM Doctor  
    WHERE department_id = D.department_id  
);
```

**54. Get patient names who have the most prescriptions.**

```
SELECT patient_name FROM Patient
```

```
WHERE patient_id = (  
    SELECT patient_id FROM Prescription  
    GROUP BY patient_id  
    ORDER BY COUNT(*) DESC  
    LIMIT 1  
);
```

**55. List departments with total doctor salary > 200,000.**

```
SELECT department_id FROM Doctor  
GROUP BY department_id  
HAVING SUM(salary) > 200000;
```

**56. Use WITH to get department doctor counts.**

```
WITH DeptDoctorCount AS (  
    SELECT department_id, COUNT(*) AS  
total
```



```
        FROM Doctor
        GROUP BY department_id
    )
SELECT * FROM DeptDoctorCount WHERE total
> 2;
```

**57. Scalar subquery to show each doctor's department name.**

```
SELECT doctor_name,
       (SELECT department_name FROM
Department
       WHERE department_id =
Doctor.department_id) AS dept_name
FROM Doctor;
```

**58. Use subquery in FROM to get top 3 oldest patients.**

```
SELECT * FROM (
```

```
        SELECT * FROM Patient ORDER BY age  
DESC LIMIT 3  
) AS TopPatients;
```

**59. List doctors and total appointments for each.**

```
SELECT D.doctor_name,  
       (SELECT COUNT(*) FROM Appointment  
A  
       WHERE A.doctor_id = D.doctor_id)  
AS total_appointments  
FROM Doctor D;
```

**60. WITH clause to find average number of prescriptions per patient.**

```
WITH PresCount AS (  
    SELECT patient_id, COUNT(*) AS total  
    FROM Prescription
```

```
        GROUP BY patient_id
    )
    SELECT AVG(total) AS avg_prescriptions
    FROM PresCount;
```

**61. Insert a new patient who has the same age as the oldest patient currently in the database.**

```
INSERT INTO Patient(patient_id,
patient_name, age)
VALUES (102, 'Sabina Yasmin', (SELECT
MAX(age) FROM Patient));
```

**62. Insert into Prescription a new record for the patient who had the latest appointment.**

```
INSERT INTO
Prescription(prescription_id,
patient_id, medicine)
VALUES (201,
```

```
        (SELECT patient_id FROM
Appointment ORDER BY appointment_date
DESC LIMIT 1),
        'Antibiotic' );
```

**63. Insert into Admission a new record for the patient who has the most prescriptions.**

```
INSERT INTO Admission(admission_id,
patient_id, room_number)
VALUES (301,
        (SELECT patient_id
         FROM Prescription
         GROUP BY patient_id
         ORDER BY COUNT(*) DESC LIMIT
1),
        'B202' );
```

**64. Insert a doctor into the department that currently has the least number of doctors.**

```
INSERT INTO Doctor(doctor_id,  
doctor_name, age, specialty,  
department_id)  
VALUES (11, 'Dr. Latif', 40,  
'Orthopedic',  
        (SELECT department_id  
         FROM Doctor  
         GROUP BY department_id  
         ORDER BY COUNT(*) ASC LIMIT  
1));
```

**65. Update salary of all doctors in the department with the highest average salary by 15%.**

```
UPDATE Doctor  
SET salary = salary * 1.15  
WHERE department_id = (  
    SELECT department_id
```

```
FROM Doctor
GROUP BY department_id
ORDER BY AVG(salary) DESC
LIMIT 1
);
```

**66. Set the age of patients who have more than 2 prescriptions to the average patient age.**

```
UPDATE Patient
SET age = (SELECT AVG(age) FROM
Patient)
WHERE patient_id IN (
    SELECT patient_id
    FROM Prescription
    GROUP BY patient_id
    HAVING COUNT(*) > 2
);
```

**67. Update room number to 'A999' for patients admitted to the same room as patient 'Sabrina Rahman'.**

```
UPDATE Admission
SET room_number = 'A999'
WHERE room_number = (
    SELECT room_number
    FROM Admission
    JOIN Patient ON
    Admission.patient_id =
    Patient.patient_id
    WHERE patient_name = 'Sabrina
    Rahman'
);
```

**68. Delete all patients who have never had any appointments or prescriptions.**

```
DELETE FROM Patient
```

```
WHERE patient_id NOT IN (SELECT  
patient_id FROM Appointment)  
AND patient_id NOT IN (SELECT  
patient_id FROM Prescription);
```

**69. Delete all prescriptions for patients younger than the average patient age.**

```
DELETE FROM Prescription  
WHERE patient_id IN (  
    SELECT patient_id FROM Patient  
    WHERE age < (SELECT AVG(age) FROM  
Patient)  
);
```

**70. Delete doctors who are not associated with any department (invalid foreign keys).**

```
DELETE FROM Doctor  
WHERE department_id NOT IN (
```



```
SELECT department_id FROM  
Department  
);
```