

**Kingdom of Saudi Arabia**

**Ministry of Higher Education**

**King Faisal University**

**College of Computer Sciences & Information Technology**

Hospital Management System

**Database Concepts and Design Course Project**

Year: 2023-2024 Semester: 2nd

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Student Name** | **Student ID** | **Contribution** |
| **S1** | Ahmed Osamah Al-Nuaim | 221426288 | * List of Main Reports * Database Conceptual Design (ERD) * Logical Design (Relational Schema) * Populate Database (DML SQL Commands) * Queries of Reports * Application * Helped in (DDL SQL Commands) |
| **S2** | Ali Habib Al-Ahmed | 221423022 | * List of main functions * Database Conceptual Design (ERD) * Queries for Reports * Helped in Application * Helped in (DDL SQL Commands) |
| **S3** | Essa Mohammed Boobaid | 223050179 | * Application Screenshots * List of users |
| **S4** | Khaled Salem Abbad | 222453448 | * Introduction * DDL SQL Command * Application Credentials |
| **S5** | Osama Sadiq Ahmed | 222453501 | * List of users * Helped in (DML SQL Commands) * Helped in (DDL SQL Commands) |

**Instructor:** Mr. Mohammed AlZahrani

## Table of Contents

[Table of Contents 2](#_Toc166423841)

[1. Introduction (case description) 3](#_Toc166423842)

[2. System Analysis 4](#_Toc166423843)

[A. List of users 4](#_Toc166423844)

[B. List of main functions (at least three) 4](#_Toc166423845)

[C. List of main reports (at least three) 4](#_Toc166423846)

[3. Database Conceptual Design (ERD) 5](#_Toc166423847)

[4. Logical Design (Relational Schema) 6](#_Toc166423848)

[5. Physical Design (DDL SQL Commands) 7](#_Toc166423849)

[6. Populate Database (DML SQL Commands) 8](#_Toc166423850)

[7. Queries for Reports 9](#_Toc166423851)

[8. Application Screenshots 10](#_Toc166423852)

[9. Application Credentials 11](#_Toc166423853)

# Introduction (case description)

In today's busy world, efficient management of healthcare facilities is crucial to ensuring patient safety and the smooth operation of hospitals. Hospital Management Systems (HMS) have revolutionized hospital workflows by simplifying processes, enhancing patient care, and optimizing resource utilization. This project aims to develop a user-friendly Hospital Management System tailored to the specific requirements of modern healthcare facilities. Therefore, the outlined requirements are as follows:

* Each doctor is identified by a unique doctorID and has a full name, gender, date of birth, qualifications, experience, email, specialty, rank, phone number, department, contact address, salary, and employment date.
* Each nurse is identified by a unique nurseID, and has a full name, qualifications, a unique doctorID, department, address, ER, rank, specialty, salary, contact address, gender, and date of birth.
* Each patient is identified by a patientID, medical history, contact, gender, insurance, age, patient name, date of birth, address, and blood type.
* Each medication is identified by a unique medicineID, medType, medName, expiration date, price, quantity, and duration.
* Each appointment is identified by a unique aptID, unique doctorID, unique patientID, appointment date, and appointment time, appointment status.
* Each lab test is identified by a unique testID, unique doctorID, unique patientID, test result, test date, and test type.
* Each payment is identified by a unique paymentID, unique invoiceID, datePaid, details, payment method, buyer name, buyer contact, confirmation number, amountPaid.
* Each department is identified by a unique depID, unique doctorID, unique nurseID, department name, contact, and description.
* Each invoice is identified by unique invoiceID, unique patientID, status, information, amount, date.

# System Analysis

## A. List of users

Doctor:

* Diagnosing patients and determining appropriate treatment.
* Scheduling appointments.
* Accessing all patient records.

Nurse:

Assisting doctors in patient care, monitoring patient vital signs.

Recording patient observations.

Accessing patient records.

Patients:

* Register for appointments.
* Get a consulted by doctors.
* Receive medical care from nurses.
* Pay for their invoices.

Lab Technician:

* Conducting laboratory tests.
* Preparing samples and performing necessary tests for patients.
* Updating laboratory test records.

Pharmacist:

* Managing medication inventory, dispensing prescriptions, and determining dosage.
* Accessing medication inventory.

Each person in this system has specific access and function that helps in building and developing the system.

## B. List of main functions (at least three)

1. **Patient Management:**
   1. The system should allow registration of new patients.

1. **Appointment Scheduling:**
   1. The system should facilitate the scheduling of appointments for patients with doctors.
2. **Doctor Management:**
   1. The system should support the addition and management of doctors.
   2. It should allow assigning doctors to specific departments within the hospital.

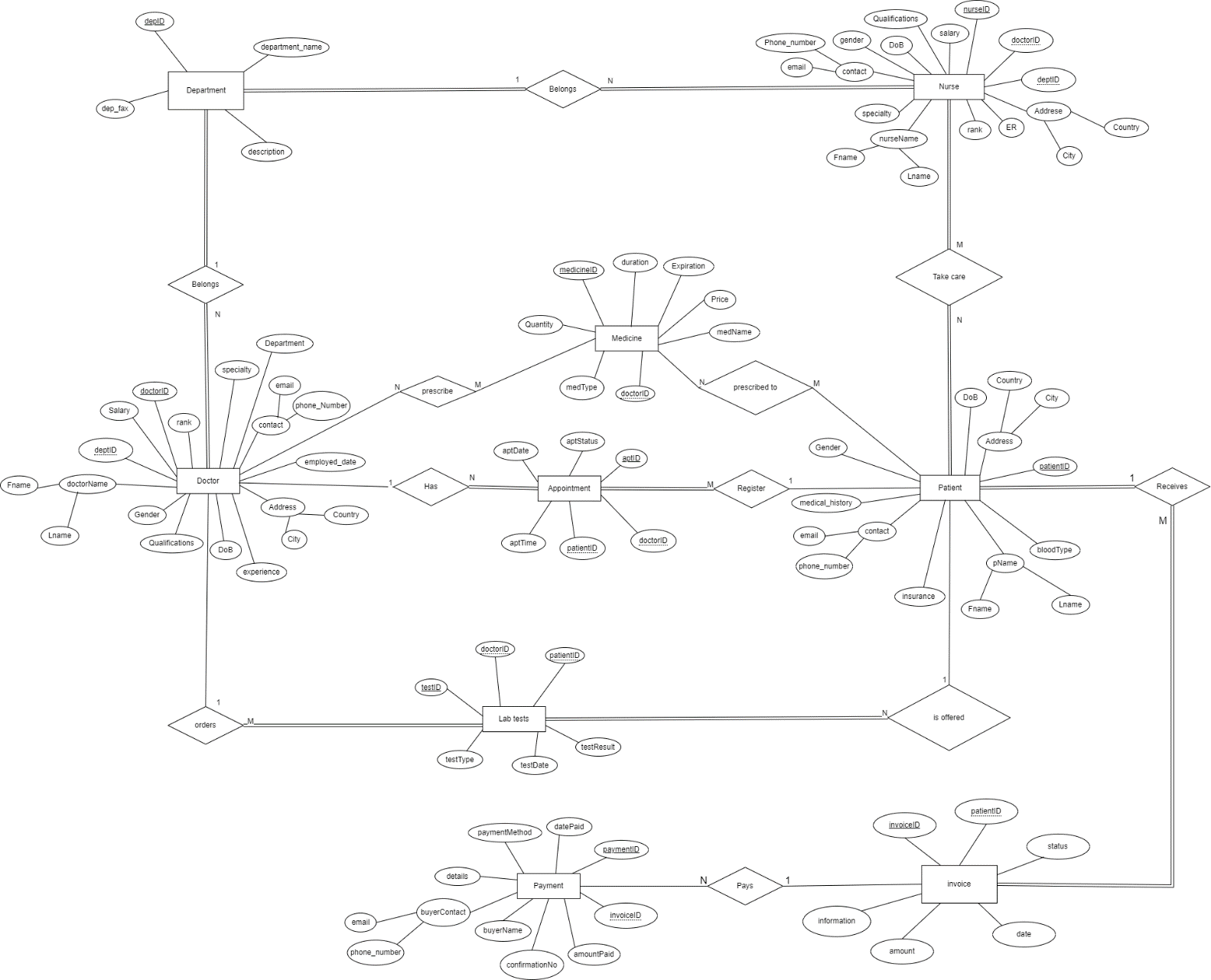
1. **Nurse Management:**
   1. The system should support the addition and management of nurses.
   2. It should allow assigning nurses to specific departments within the hospital.
2. **Payment Management:**
   1. The system should handle payment processing for services rendered to patients.
3. **Invoice Management:**
   1. The system should generate and manage invoices for services provided to patients.
4. **Medicine Management:**
   1. The system should maintain a database of medicines, including their names, duration, and other relevant information.
   2. It should allow healthcare providers to prescribe medicines to patients and record the details of the prescription.

1. **Lab Test Management:**
   1. Ordering lab tests and diagnostic procedures.
   2. Integrating lab results with patient records.

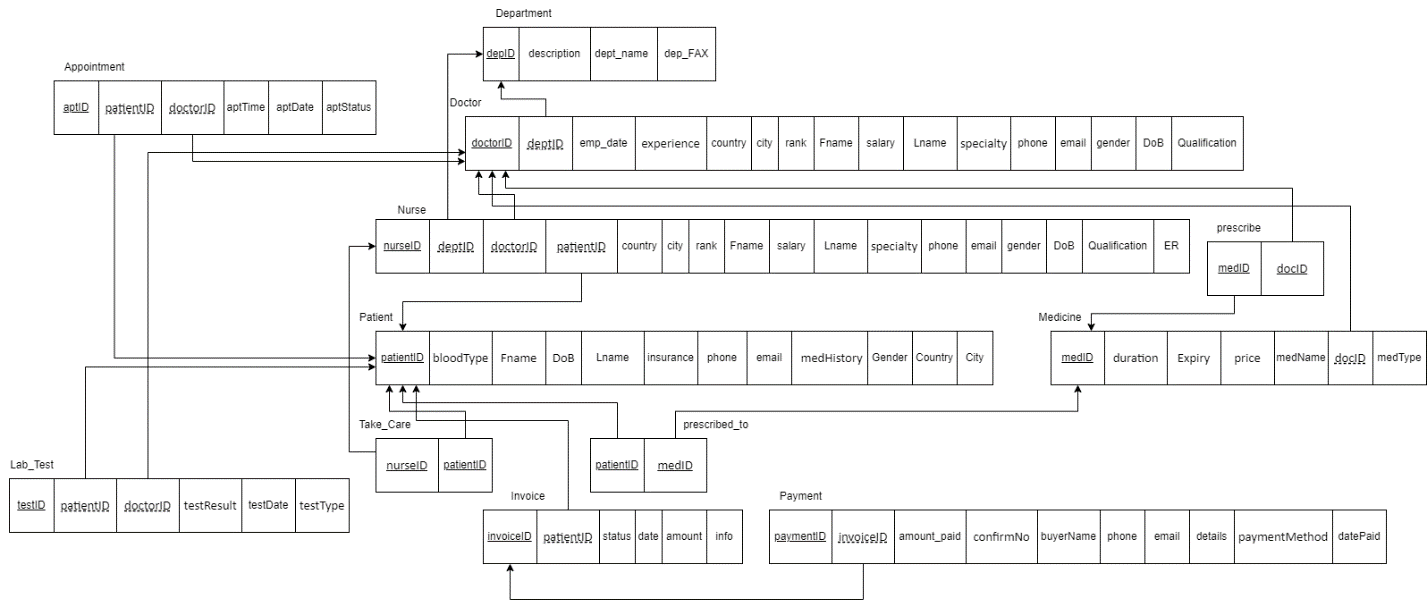
## C. List of main reports (at least three)

1. Doctor Schedule Report - Lists available times and dates for each doctor.
2. Nurse Assignment Report - Displays the nurses' assignments for the various departments or patients they are responsible for.
3. Patient Medical History Report - Provides a brief overview of the patient's medical history.
4. Patient Medication Report - tracks medications prescribed to patients by the specialist.
5. Daily Appointments Report - Summarizes all patient appointments scheduled for the day.
6. Medicine Stock Report - Shows the number of medicines in stock and their availability.
7. Laboratory Test Results Report - Displays the results of all laboratory tests performed.
8. Financial Transaction Report – Summarizes all financial transactions, including invoices, payments, and insurance.

# Database Conceptual Design (ERD)



# Logical Design (Relational Schema)



|  |  |  |  |
| --- | --- | --- | --- |
| Department | | | |
| **Name** | **Datatype** | **Key type** | **Constraint** |
| depID | Number | PK |  |
| Department\_name | Varchar2(100) |  | Not Null |
| Dep\_fax | Number |  |  |
| Description | Varchar2(200) |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Doctor | | | |
| Name | Datatype | Key type | Constraint |
| DoctorID | Number | PK |  |
| depID | Number | FK | CONSTRAINT doctor\_dep\_Id\_FK FOREIGN KEY (depID) REFERENCES department(depID) ON DELETE SET NULL; |
| Fname | Varchar2(50) |  |  |
| Mname | Varchar2(50) |  |  |
| Gender | Varchar2(10) |  |  |
| Date\_of\_birth | Date |  |  |
| Qualifications | Varchar2(100) |  |  |
| Experience | Number |  |  |
| Email | Varchar2(100) |  |  |
| Speciality | Varchar2(100) |  |  |
| Rank | Varchar2(50) |  |  |
| Phone\_Number | Varchar2(20) |  |  |
| Country | Varchar2(50) |  |  |
| City | Varchar2(50) |  |  |
| Salary | Number |  | CHECK (salary > 0) |
| Employment\_date | Date |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Nurse | | | |
| **Name** | **Datatype** | **Key type** | **Constraint** |
| nurseID | Number | PK |  |
| doctorID | Number | FK | CONSTRAINT nurse\_doctor\_Id\_FK REFERENCES doctor(doctorID) ON DELETE SET NULL |
| DepID | Number | FK | depID NUMBER CONSTRAINT nurse\_dep\_Id\_FK REFERENCES department(depID) ON DELETE SET NULL |
| patientID | Number | FK | patientID NUMBER CONSTRAINT nurse\_patient\_Id\_FK REFERENCES patient(patientID) ON DELETE SET NULL |
| email | Varchar2(100) |  |  |
| phone | Varchar2(20) |  |  |
| gender | Varchar2(10) |  |  |
| Date\_of\_Birth | Date |  |  |
| qualifications | Varchar2(100) |  |  |
| salary | Number |  | CHECK (salary > 0) |
| country | Varchar2(50) |  |  |
| city | Varchar2(50) |  |  |
| ER | Number |  | CHECK (ER IN ("YES","NO") |
| rank | Varchar2(50) |  |  |
| Fname | Varchar2(50) |  |  |
| Lname | Varchar2(50) |  |  |
| specialty | Varchar2(100) |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Patient | | | |
| **Name** | **Datatype** | **Key type** | **Constraint** |
| PatientID | Number | PK |  |
| Fname | Varchar2(50) |  |  |
| Lname | Varchar2(50) |  |  |
| Medical\_History | Varchar2(300) |  |  |
| email | Varchar2(100) |  |  |
| phone | Varchar2(20) |  |  |
| gender | Varchar2(10) |  |  |
| Date\_of\_Birth | Date |  |  |
| Insurance | Number |  |  |
| Blood\_Type | Varchar2(10) |  |  |
| country | Varchar2(50) |  |  |
| city | Varchar2(50) |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Medication | | | |
| **Name** | **Datatype** | **Key type** | **Constraint** |
| medicineID | Number | PK |  |
| medType | Varchar2(50) |  |  |
| medName | Varchar2(100) |  |  |
| Expiration\_date | Date |  |  |
| price | Number |  |  |
| Quantity | Number |  |  |
| Duration | Varchar2(50) |  |  |
| DoctorID | Number | FK | doctorID NUMBER CONSTRAINT medication\_doctor\_Id\_FK REFERENCES doctor(doctorID) ON DELETE SET NULL |

|  |  |  |  |
| --- | --- | --- | --- |
| Appointment | | | |
| **Name** | **Datatype** | **Key type** | **Constraint** |
| aptID | Number | PK |  |
| doctorID | Number | FK | doctorID NUMBER CONSTRAINT appointment\_doctor\_Id\_FK REFERENCES doctor(doctorID) ON DELETE SET NULL |
| patientID | Number | FK | patientID NUMBER CONSTRAINT appointment\_patient\_Id\_FK REFERENCES patient(patientID) ON DELETE SET NULL |
| apTime | Varchar2(50) |  |  |
| aptDate | Date |  |  |
| aptStatus | Varchar2(50) |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Lab\_Test | | | |
| **Name** | **Datatype** | **Key type** | **Constraint** |
| testID | Number | PK |  |
| doctorID | Number | FK | CONSTRAINT lab\_test\_doctor\_fk FOREIGN KEY (doctorID) REFERENCES doctor(doctorID) ON DELETE SET NULL |
| patientID | Number | FK | CONSTRAINT lab\_test\_patient\_fk FOREIGN KEY (patientID) REFERENCES patient(patientID) ON DELETE SET NULL |
| Test\_Date | Date |  |  |
| Test\_Type | Varchar2(50) |  |  |
| Test\_Result | Varchar2(200) |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Invoice | | | |
| **Name** | **Datatype** | **Key type** | **Constraint** |
| invoiceID | Number | PK |  |
| patientID | Number | FK | CONSTRAINT invoice\_patient\_fk FOREIGN KEY (patientID) REFERENCES patient(patientID) ON DELETE SET NULL |
| status | Varchar2(50) |  |  |
| Invoice\_Date | Date |  |  |
| Amount | Number |  |  |
| Information | Varchar2(200) |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Payment | | | |
| **Name** | **Datatype** | **Key type** | **Constraint** |
| paymentID | Number | PK |  |
| Payer\_Name | Varchar2(100) |  |  |
| Payer\_Phone | Varchar2(20) |  |  |
| invoiceID | Number | FK | CONSTRAINT payment\_invoice\_fk FOREIGN KEY (invoiceID) REFERENCES invoice(invoiceID) ON DELETE SET NULL |
| datePaid | Date |  |  |
| details | Varchar2(200) |  |  |
| Payment\_method | Varchar2(50) |  |  |
| Confirmation\_Number | Varchar2(50) |  |  |
| AmountPaid | Number |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Take\_care | | | |
| **Name** | **Datatype** | **Key type** | **Constraint** |
| nurseID | Number | PK | CONSTRAINT take\_care\_nurse\_fk FOREIGN KEY (nurseID) REFERENCES nurse(nurseID) ON DELETE SET NULL |
| patientID | Number | PK | CONSTRAINT take\_care\_patient\_fk FOREIGN KEY (patientID) REFERENCES patient(patientID) ON DELETE SET NULL |

|  |  |  |  |
| --- | --- | --- | --- |
| Prescribe\_to | | | |
| **Name** | **Datatype** | **Key type** | **Constraint** |
| medicineID | Number | PK | CONSTRAINT prescribed\_to\_medicineID\_fk FOREIGN KEY (medicineID) REFERENCES medication(medicineID) ON DELETE SET NULL |
| patientID | Number | PK | CONSTRAINT prescribed\_to\_patient\_fk FOREIGN KEY (patientID) REFERENCES patient(patientID) ON DELETE SET NULL |

|  |  |  |  |
| --- | --- | --- | --- |
| Prescribe | | | |
| **Name** | **Datatype** | **Key type** | **Constraint** |
| doctorID | Number | PK | ONSTRAINT prescribe\_medicineID\_fk FOREIGN KEY (doctorID) REFERENCES doctor(doctorID) ON DELETE SET NULL |
| patientID | Number | PK | CONSTRAINT prescribe\_patient\_fk FOREIGN KEY (patientID) REFERENCES patient(patientID) ON DELETE SET NULL |

# Physical Design (DDL SQL Commands)

## Depatment Table:

CREATE TABLE department (

DepID Number Primary Key,

department\_name VARCHAR2(100) not null,

dep\_Fax NUMBER,

description VARCHAR2(200)

);

## Doctor Table:

CREATE TABLE doctor (

doctorID NUMBER PRIMARY KEY,

depID NUMBER,

fname VARCHAR2(50),

lname VARCHAR2(50),

gender VARCHAR2(10),

date\_of\_birth DATE,

qualifications VARCHAR2(100),

experience NUMBER,

email VARCHAR2(100),

specialty VARCHAR2(100),

rank VARCHAR2(50),

phone\_number VARCHAR2(20),

department VARCHAR2(100),

country VARCHAR2(50),

city VARCHAR2(50),

salary NUMBER CHECK (salary > 0),

employment\_date DATE

CONSTRAINT doctor\_dep\_Id\_FK FOREIGN KEY (depID) REFERENCES department(depID) ON DELETE SET NULL;

);

## Patient Table:

CREATE TABLE patient (

patientID NUMBER PRIMARY KEY,

fname VARCHAR2(50),

lname VARCHAR2(50),

medical\_history VARCHAR2(300),

email VARCHAR2(100),

phone\_number VARCHAR2(20),

gender VARCHAR2(10),

insurance NUMBER,

date\_of\_birth DATE,

country VARCHAR2(50),

city VARCHAR2(50),

blood\_type VARCHAR2(10)

);

## Nurse Table:

CREATE TABLE nurse (

nurseID NUMBER PRIMARY KEY,

fname VARCHAR2(50),

lname VARCHAR2(50),

gender VARCHAR2(10),

date\_of\_birth DATE,

qualifications VARCHAR2(100),

doctorID NUMBER CONSTRAINT nurse\_doctor\_Id\_FK REFERENCES doctor(doctorID) ON DELETE SET NULL

depID NUMBER CONSTRAINT nurse\_dep\_Id\_FK REFERENCES department(depID) ON DELETE SET NULL

patientID NUMBER CONSTRAINT nurse\_patient\_Id\_FK REFERENCES patient(patientID) ON DELETE SET NULL

department VARCHAR2(100),

country VARCHAR2(50),

city VARCHAR2(50),

ER NUMBER (5) CHECK (ER IN (‘YES’,’NO’)),

rank VARCHAR2(50),

specialty VARCHAR2(100),

salary NUMBER CHECK (salary > 0),

phone\_number VARCHAR2(20),

email VARCHAR2(100)

);

## Medication Table:

CREATE TABLE medication (

medicineID NUMBER PRIMARY KEY,

medType VARCHAR2(50),

medName VARCHAR2(100),

expiration\_date DATE,

price NUMBER,

quantity NUMBER check (quantity >-1),

duration VARCHAR2(50),

doctorID NUMBER CONSTRAINT medication\_doctor\_Id\_FK REFERENCES doctor(doctorID) ON DELETE SET NULL

);

## Appointment Table:

CREATE TABLE appointment (

aptID NUMBER PRIMARY KEY,

doctorID NUMBER CONSTRAINT appointment\_doctor\_Id\_FK REFERENCES doctor(doctorID) ON DELETE SET NULL,

patientID NUMBER CONSTRAINT appointment\_patient\_Id\_FK REFERENCES patient(patientID) ON DELETE SET NULL,

aptDate DATE,

aptTime VARCHAR2(50),

aptStatus VARCHAR2(50)

);

## Lab Test Table:

CREATE TABLE lab\_test (

testID NUMBER PRIMARY KEY,

doctorID NUMBER,

patientID NUMBER,

test\_date DATE,

test\_type VARCHAR2(50),

test\_result VARCHAR2(200),

CONSTRAINT lab\_test\_doctor\_fk FOREIGN KEY (doctorID) REFERENCES doctor(doctorID) ON DELETE SET NULL ,

CONSTRAINT lab\_test\_patient\_fk FOREIGN KEY (patientID) REFERENCES patient(patientID) ON DELETE SET NULL

);

## Invoice Table:

CREATE TABLE invoice (

invoiceID NUMBER PRIMARY KEY,

patientID NUMBER,

status VARCHAR2(50),

information VARCHAR2(200),

amount NUMBER,

invoice\_date DATE,

CONSTRAINT invoice\_patient\_fk FOREIGN KEY (patientID) REFERENCES patient(patientID) ON DELETE SET NULL

);

## Payment Table:

CREATE TABLE payment (

paymentID NUMBER PRIMARY KEY,

payer\_name VARCHAR2(100),

payer\_phone VARCHAR2(20),

invoiceID NUMBER,

datePaid DATE,

details VARCHAR2(200),

payment\_method VARCHAR2(50),

confirmation\_number VARCHAR2(50),

amountPaid NUMBER,

CONSTRAINT payment\_invoice\_fk FOREIGN KEY (invoiceID) REFERENCES invoice(invoiceID) ON DELETE SET NULL);

## Additional tables for many-to-many relationship:

create table Take\_care (

nurseID NUMBER,

patientID NUMBER,

CONSTRAINT take\_care\_nurse\_fk FOREIGN KEY (nurseID) REFERENCES nurse(nurseID) ON DELETE SET NULL,

CONSTRAINT take\_care\_patient\_fk FOREIGN KEY (patientID) REFERENCES patient(patientID) ON DELETE SET NULL

);

create table prescribed\_to (

medicineID NUMBER Primary Key,

patientID NUMBER Primary Key,

CONSTRAINT prescribed\_to\_medicineID\_fk FOREIGN KEY (medicineID) REFERENCES medication(medicineID) ON DELETE SET NULL,

CONSTRAINT prescribed\_to\_patient\_fk FOREIGN KEY (patientID) REFERENCES patient(patientID) ON DELETE SET NULL

);

create table prescribe (

doctorID NUMBER,

patientID NUMBER,

primary key (doctorID, patientID),

CONSTRAINT prescribe\_medicineID\_fk FOREIGN KEY (doctorID) REFERENCES doctor(doctorID) ON DELETE SET NULL,

CONSTRAINT prescribe\_patient\_fk FOREIGN KEY (patientID) REFERENCES patient(patientID) ON DELETE SET NULL

);

# Populate Database (DML SQL Commands)

Department table:  
INSERT INTO department VALUES (101, 'Cardiology', 123456789, 'Heart related diseases')

INSERT INTO department VALUES (102, 'Neurology', 987654321, 'Brain and nervous system disorders')

INSERT INTO department VALUES (103, 'Pediatrics', 234567891, 'Child care and diseases')

INSERT INTO department VALUES (104, 'Orthopedics', 345678912, 'Bone and joint treatment')

INSERT INTO department VALUES (105, 'Dermatology', 456789123, 'Skin care')

INSERT INTO department VALUES (106, 'Oncology', 567891234, 'Cancer treatment')

INSERT INTO department VALUES (107, 'Emergency', 678912345, 'Emergency and acute care')

INSERT INTO department VALUES (108, 'Gastroenterology', 789123456, 'Digestive system disorders')

INSERT INTO department VALUES (109, 'Urology', 891234567, 'Urinary system disorders')

INSERT INTO department VALUES (110, 'Ophthalmology', 912345678, 'Eye diseases')

## Doctor Table:

INSERT INTO doctor VALUES (1, 101, 'Ahmed', 'Osamah', 'Al-Nuaim', 'Male', DATE '1980-06-15', 'MD Cardiology', 15, 'ABC@email.com', 'Cardiology', 'Senior', '555-1234', 'SA', 'Al-Ahsa', 10001, 200000, DATE '2008-05-10')

INSERT INTO doctor VALUES (2, 102, 'Khalid', 'Ahmad', 'Abbad', 'Male', DATE '1982-07-20', 'MD Neurology', 12, 'DEF@email.com', 'Neurology', 'Consultant', '555-2345', 'SA', 'Almadinah', 90001, 180000, DATE '2010-03-15')

INSERT INTO doctor VALUES (3, 103, 'Osama', 'Ibrahim', 'Ahmed', 'Male', DATE '1975-05-11', 'MD Pediatrics', 20, 'GHI@email.com', 'Pediatrics', 'Head', '555-3456', 'SA', 'Makkah', 60007, 220000, DATE '2000-04-20')

INSERT INTO doctor VALUES (4, 104, 'Eisa', 'Sultan', 'Boobaid', 'Male', DATE '1985-08-25', 'MD Orthopedics', 10, 'JKL@email.com', 'Orthopedics', 'Junior', '555-4567', 'SA', 'Jazan', 77001, 150000, DATE '2013-09-10')

INSERT INTO doctor VALUES (5, 105, 'Mohammed', 'Saleh', 'Al-Zahrani', 'Male', DATE '1983-09-30', 'MD Dermatology', 14, 'MNO@email.com', 'Dermatology', 'Senior', '555-5678', 'SA', 'Hafr Albatin', 85001, 10000000000, DATE '2009-01-25')

INSERT INTO doctor VALUES (6, 106, 'Abdullah', 'Sami', 'Almarry', 'Male', DATE '1979-12-15', 'MD Oncology', 18, 'PQY@email.com', 'Oncology', 'Consultant', '555-6789', 'SA', 'Dammam', 19101, 210000, DATE '2005-07-30')

INSERT INTO doctor VALUES (7, 107, 'Hamad', 'Abdulrahim', 'Aldosary', 'Male', DATE '1988-10-05', 'MD Emergency', 8, 'STU@email.com', 'Emergency', 'Junior', '555-7890', 'SA', 'Al-Ahsa', 78201, 140000, DATE '2015-06-15')

INSERT INTO doctor VALUES (8, 108, 'Salman', 'Adel', 'Alshammary', 'Male', DATE '1978-03-20', 'MD Gastroenterology', 19, 'VWX@email.com', 'Gastroenterology', 'Head', '555-8901', 'SA', 'Jeddah', 75201, 230000, DATE '2004-08-10')

INSERT INTO doctor VALUES (9, 109, 'Zaid', 'Ibrahim', 'Aldawood', 'Male', DATE '1984-01-15', 'MD Urology', 13, 'YZZ@email.com', 'Urology', 'Senior', '555-9012', 'SA', 'Arar', 95101, 160000, DATE '2011-02-20')

INSERT INTO doctor VALUES (10, 110, 'Azzam', 'Salman', 'Al-Abbad', 'Male', DATE '1981-04-22', 'MD Ophthalmology', 11, 'ABCDEF@email.com', 'Ophthalmology', 'Consultant', '555-0123', 'SA', 'Riyadh', 73301, 175000, DATE '2012-11-25')

Patient Table:  
INSERT INTO patient VALUES (10, 'Amira', 'Fahad', 'Al-Ghamdi', 'No significant medical history.', 'amira.alghamdi@email.com', '666-1234', 'Female', 1, DATE '1990-10-10', 30, 'Saudi Arabia', 'Riyadh', 'A+');

INSERT INTO patient VALUES (11, 'Khalid', 'Yousef', 'Al-Farsi', 'Allergic to penicillin.', 'khalid.alfarsi@email.com', '666-2345', 'Male', 2, DATE '1985-08-15', 35, 'Oman', 'Muscat', 'B+');

INSERT INTO patient VALUES (12, 'Layla', 'Mohammad', 'Al-Khouri', 'Diabetic.', 'layla.alkhouri@email.com', '666-3456', 'Female', 3, DATE '1982-03-20', 39, 'United Arab Emirates', 'Dubai', 'O-');

INSERT INTO patient VALUES (13, 'Omar', 'Tariq', 'Ibrahim', 'Hypertensive, chronic kidney disease.', 'omar.ibrahim@email.com', '666-4567', 'Male', 4, DATE '1975-07-25', 45, 'Qatar', 'Doha', 'AB+');

INSERT INTO patient VALUES (14, 'Fatima', 'Ali', 'Bakr', 'Asthmatic.', 'fatima.bakr@email.com', '666-5678', 'Female', 5, DATE '1995-12-30', 25, 'Bahrain', 'Manama', 'A-');

INSERT INTO patient VALUES (15, 'Nasser', 'Sami', 'Al-Masri', 'History of cardiac arrhythmias.', 'nasser.almasri@email.com', '666-6789', 'Male', 6, DATE '1970-02-10', 50, 'Egypt', 'Cairo', 'B-');

INSERT INTO patient VALUES (16, 'Noura', 'Hussein', 'Al-Saad', 'Migraines, epilepsy.', 'noura.alsaad@email.com', '666-7890', 'Female', 7, DATE '1988-05-15', 32, 'Kuwait', 'Kuwait City', 'O+');

INSERT INTO patient VALUES (17, 'Ibrahim', 'Mahmoud', 'Al-Rashid', 'Chronic liver disease.', 'ibrahim.alrashid@email.com', '666-8901', 'Male', 8, DATE '1979-01-25', 41, 'Jordan', 'Amman', 'AB-');

INSERT INTO patient VALUES (18, 'Mariam', 'Faisal', 'Najjar', 'Thyroid disorders.', 'mariam.najjar@email.com', '666-9012', 'Female', 9, DATE '1983-11-30', 37, 'Lebanon', 'Beirut', 'A+');

INSERT INTO patient VALUES (19, 'Saud', 'Abdul', 'Al-Fayad', 'History of peptic ulcer disease.', 'saud.alfayad@email.com', '666-0123', 'Male', 10, DATE '1978-04-20', 42, 'Iraq', 'Baghdad', 'B+');

Nurse Table:  
INSERT INTO nurse VALUES (20, 'Nawal', 'Al-Sayed', 'Female', DATE '1987-04-10', 'Registered Nurse', 1, 101, 10, 'Egypt', 'Cairo', ‘YES’, 'Senior Nurse', 'Cardiology Nursing', 75000, '777-1234', 'nawal.alsayed@email.com');

INSERT INTO nurse VALUES (21, 'Sami', 'Nasser', 'Male', DATE '1990-08-22', 'Nurse Practitioner', 2, 102, 11, 'Saudi Arabia', 'Riyadh', ‘NO’, 'Lead Nurse', 'Neurology Nursing', 85000, '777-2345', 'sami.nasser@email.com');

INSERT INTO nurse VALUES (22, 'Layla', 'Fahmi', 'Female', DATE '1992-01-30', 'Clinical Nurse Specialist', 3, 103, 12, 'Jordan', 'Amman', ‘YES’, 'Staff Nurse', 'Pediatric Nursing', 65000, '777-3456', 'layla.fahmi@email.com');

INSERT INTO nurse VALUES (23, 'Karim', 'El-Baz', 'Male', DATE '1983-09-17', 'Registered Nurse', 4, 104, 13, 'UAE', 'Dubai', ‘YES’, 'Senior Nurse', 'Orthopedic Nursing', 78000, '777-4567', 'karim.elbaz@email.com');

INSERT INTO nurse VALUES (24, 'Rania', 'Moussa', 'Female', DATE '1985-11-05', 'Nurse Practitioner', 5, 105, 14, 'Qatar', 'Doha', ‘YES’, 'Lead Nurse', 'Dermatology Nursing', 82000, '777-5678', 'rania.moussa@email.com');

INSERT INTO nurse VALUES (25, 'Faisal', 'Al-Ahmad', 'Male', DATE '1988-06-20', 'Clinical Nurse Specialist', 6, 106, 15, 'Kuwait', 'Kuwait City', ‘NO’, 'Staff Nurse', 'Oncology Nursing', 71000, '777-6789', 'faisal.alahmad@email.com');

INSERT INTO nurse VALUES (26, 'Mona', 'Habib', 'Female', DATE '1981-12-15', 'Registered Nurse', 7, 107, 16, 'Oman', 'Muscat', ‘YES’, 'Senior Nurse', 'Emergency Nursing', 79000, '777-7890', 'mona.habib@email.com');

INSERT INTO nurse VALUES (27, 'Ibrahim', 'Al-Sulaiman', 'Male', DATE '1994-03-25', 'Nurse Practitioner', 8, 108, 17, 'Bahrain', 'Manama', ‘NO’, 'Lead Nurse', 'Gastroenterology Nursing', 86000, '777-8901', 'ibrahim.alsulaiman@email.com');

INSERT INTO nurse VALUES (28, 'Hala', 'Al-Mansoori', 'Female', DATE '1993-07-18', 'Clinical Nurse Specialist', 9, 109, 18, 'Syria', 'Damascus', ‘YES’, 'Staff Nurse', 'Urology Nursing', 73000, '777-9012', 'hala.almansoori@email.com');

INSERT INTO nurse VALUES (29, 'Yousef', 'Khaled', 'Male', DATE '1986-11-03', 'Registered Nurse', 10, 110, 19, 'Lebanon', 'Beirut', ‘YES’, 'Senior Nurse', 'Ophthalmology Nursing', 77000, '777-0123', '[yousef.khaled@email.com](mailto:yousef.khaled@email.com)');

## Medication Table:

INSERT INTO medication VALUES (1, 'Tablet', 'Paracetamol', DATE '2023-12-31', 5, 100, '7 days', 1);

INSERT INTO medication VALUES (2, 'Capsule', 'Amoxicillin', DATE '2024-01-31', 8, 150, '14 days', 2);

INSERT INTO medication VALUES (3, 'Syrup', 'Cough Syrup', DATE '2024-03-15', 10, 120, '5 days', 3);

INSERT INTO medication VALUES (4, 'Injection', 'Insulin', DATE '2023-11-20', 45, 90, '30 days', 4);

INSERT INTO medication VALUES (5, 'Ointment', 'Hydrocortisone', DATE '2024-02-28', 22, 80, '10 days', 5);

INSERT INTO medication VALUES (6, 'Tablet', 'Atorvastatin', DATE '2024-05-30', 15, 200, '30 days', 6);

INSERT INTO medication VALUES (7, 'Capsule', 'Omeprazole', DATE '2023-10-25', 12, 180, '15 days', 7);

INSERT INTO medication VALUES (8, 'Injection', 'Vaccine', DATE '2024-04-10', 60, 70, 'Single use', 8);

INSERT INTO medication VALUES (9, 'Tablet', 'Metformin', DATE '2023-09-15', 7, 110, '30 days', 9);

INSERT INTO medication VALUES (10, 'Capsule', 'Gabapentin', DATE '2024-06-07', 18, 95, '21 days', 10);

## Appointment Table:

INSERT INTO appointment VALUES (1, 1, 10, DATE '2024-06-10', '09:00', 'Scheduled')

INSERT INTO appointment VALUES (2, 2, 11, DATE '2024-06-11', '10:00', 'Completed')

INSERT INTO appointment VALUES (3, 3, 12, DATE '2024-06-12', '11:00', 'In progress')

INSERT INTO appointment VALUES (4, 4, 13, DATE '2024-06-13', '12:00', 'Resecheduled')

INSERT INTO appointment VALUES (5, 5, 14, DATE '2024-06-14', '13:00', 'Awaiting payment')

INSERT INTO appointment VALUES (6, 6, 15, DATE '2024-06-15', '14:00', 'Cancelled')

INSERT INTO appointment VALUES (7, 7, 16, DATE '2024-06-16', '15:00', 'Checked in')

INSERT INTO appointment VALUES (8, 8, 17, DATE '2024-06-17', '16:00', 'Completed')

INSERT INTO appointment VALUES (9, 9, 18, DATE '2024-06-18', '17:00', 'Scheduled')

INSERT INTO appointment VALUES (10, 10, 19, DATE '2024-06-19', '18:00', 'Completed')

## Lab\_Test Table:

INSERT INTO lab\_test VALUES (1, 1, 10, DATE '2024-01-05', 'Blood Test', 'Normal');

INSERT INTO lab\_test VALUES (2, 2, 11, DATE '2024-01-06', 'X-Ray', 'Normal');

INSERT INTO lab\_test VALUES (3, 3, 12, DATE '2024-01-07', 'MRI', 'Abnormal');

INSERT INTO lab\_test VALUES (4, 4, 13, DATE '2024-01-08', 'CT Scan (Computed Tomography)', 'Normal');

INSERT INTO lab\_test VALUES (5, 5, 14, DATE '2024-01-09', 'Ultrasound', 'Abnormal');

INSERT INTO lab\_test VALUES (6, 6, 15, DATE '2024-01-10', 'Electrocardiogram (Heart Activity)', 'Normal');

INSERT INTO lab\_test VALUES (7, 7, 16, DATE '2024-01-11', 'Blood Test', 'High Cholesterol');

INSERT INTO lab\_test VALUES (8, 8, 17, DATE '2024-01-12', 'Blood Test', 'Normal');

INSERT INTO lab\_test VALUES (9, 9, 18, DATE '2024-01-13', 'X-Ray', 'Broken Bone');

INSERT INTO lab\_test VALUES (10, 10, 19, DATE '2024-01-14', 'MRI', 'Normal');

## Invoice Table:

INSERT INTO invoice VALUES (1, 10, 'Paid', 'General Consultation', 100, DATE '2024-01-15');

INSERT INTO invoice VALUES (2, 11, 'Unpaid', 'Surgery', 2000, DATE '2024-01-16');

INSERT INTO invoice VALUES (3, 12, 'Paid', 'Routine Check-up', 150, DATE '2024-01-17');

INSERT INTO invoice VALUES (4, 13, 'Partially Paid', 'Emergency Treatment', 750, DATE '2024-01-18');

INSERT INTO invoice VALUES (5, 14, 'Paid', 'Skin Treatment', 300, DATE '2024-01-19');

INSERT INTO invoice VALUES (6, 15, 'Unpaid', 'Cancer Treatment', 2200, DATE '2024-01-20');

INSERT INTO invoice VALUES (7, 16, 'Paid', 'Accident Care', 900, DATE '2024-01-21');

INSERT INTO invoice VALUES (8, 17, 'Unpaid', 'Digestive Issue Treatment', 400, DATE '2024-01-22');

INSERT INTO invoice VALUES (9, 18, 'Paid', 'Urinary System Check', 250, DATE '2024-01-23');

INSERT INTO invoice VALUES (10, 19, 'Partially Paid', 'Eye Examination', 100, DATE '2024-01-24');

## Payment Table:

INSERT INTO payment VALUES (1, 'Layla Abdullah', '666-1234', 5, DATE '2024-01-15', 'Consultation fee (Skin Treatment)', 'Credit Card', 'CN12345', 100);

INSERT INTO payment VALUES (2, 'Bader Zaid', '666-2345', 1, DATE '2024-01-17', 'Routine check-up fee (General Consultation)', 'Cash', 'CN23456', 150);

INSERT INTO payment VALUES (3, 'Aljawharah Mohammed', '666-3456', 3, DATE '2024-01-19', 'Skin treatment fee (Routine Check-up)', 'Debit Card', 'CN34567', 300);

INSERT INTO payment VALUES (4, 'Rami Malik', '666-4567', 7, DATE '2024-01-21', 'Accident care fee', 'Credit Card', 'CN45678', 900);

INSERT INTO payment VALUES (5, 'Abdulrahman Khaled', '666-5678', 9, DATE '2024-01-23', 'Urinary system check fee', 'Cash', 'CN56789', 250);

## Take\_Care Table:

INSERT INTO Take\_care (nurseID, patientID) VALUES (20, 10);

INSERT INTO Take\_care (nurseID, patientID) VALUES (21, 11);

INSERT INTO Take\_care (nurseID, patientID) VALUES (22, 12);

INSERT INTO Take\_care (nurseID, patientID) VALUES (23, 13);

INSERT INTO Take\_care (nurseID, patientID) VALUES (24, 14);

INSERT INTO Take\_care (nurseID, patientID) VALUES (25, 15);

INSERT INTO Take\_care (nurseID, patientID) VALUES (26, 16);

INSERT INTO Take\_care (nurseID, patientID) VALUES (27, 17);

INSERT INTO Take\_care (nurseID, patientID) VALUES (28, 18);

INSERT INTO Take\_care (nurseID, patientID) VALUES (29, 19);

## Prescribed\_To Table:

INSERT INTO prescribed\_to (medicineID, patientID) VALUES (1, 10);

INSERT INTO prescribed\_to (medicineID, patientID) VALUES (2, 11);

INSERT INTO prescribed\_to (medicineID, patientID) VALUES (3, 12);

INSERT INTO prescribed\_to (medicineID, patientID) VALUES (4, 13);

INSERT INTO prescribed\_to (medicineID, patientID) VALUES (5, 14);

INSERT INTO prescribed\_to (medicineID, patientID) VALUES (6, 15);

INSERT INTO prescribed\_to (medicineID, patientID) VALUES (7, 16);

INSERT INTO prescribed\_to (medicineID, patientID) VALUES (8, 17);

INSERT INTO prescribed\_to (medicineID, patientID) VALUES (9, 18);

INSERT INTO prescribed\_to (medicineID, patientID) VALUES (10, 19);

## Prescribed Table:

INSERT INTO prescribe (doctorID, patientID) VALUES (1, 10);

INSERT INTO prescribe (doctorID, patientID) VALUES (2, 11);

INSERT INTO prescribe (doctorID, patientID) VALUES (3, 12);

INSERT INTO prescribe (doctorID, patientID) VALUES (4, 13);

INSERT INTO prescribe (doctorID, patientID) VALUES (5, 14);

INSERT INTO prescribe (doctorID, patientID) VALUES (6, 15);

INSERT INTO prescribe (doctorID, patientID) VALUES (7, 16);

INSERT INTO prescribe (doctorID, patientID) VALUES (8, 17);

INSERT INTO prescribe (doctorID, patientID) VALUES (9, 18);

INSERT INTO prescribe (doctorID, patientID) VALUES (10, 19);

# Queries for Reports

## Doctor Schedule Report - Lists available times and dates for each doctor.

1. SELECT CONCAT(CONCAT( d.fname, ' ' ),d.lname) AS "Doctor Name", a.aptDate, a.aptTime AS "Appointment Time"
2. FROM doctor d
3. JOIN appointment a ON d.doctorID = a.doctorID
4. ORDER BY d.fname, d.lname, a.aptDate, a.aptTime;

## Nurse Assignment Report - Displays the nurses' assignments for the various departments or patients they are responsible for.

1. select n.nurseID, concat(concat(n.fname, ' '), n.lname) as "Nurse Name", concat(concat(p.fname, ' '), p.lname) as "Patient Name"
2. from nurse n join patient p
3. on n.patientID = p.patientID

## Patient Medical History Report - Provides a brief overview of the patient's medical history.

1. SELECT CONCAT(CONCAT( fname, ' ' ),lname) AS "Patient Name", medical\_history AS " Medical History"
2. FROM patient;

## Patient Medication Report - tracks medications prescribed to patients by the specialist.

1. SELECT CONCAT(CONCAT( p.fname, ' ' ), p.lname) AS "Patient Name", m.medName AS "Medication Name", m.duration
2. FROM patient p
3. JOIN prescribed\_to pt ON p.patientID = pt.patientID
4. JOIN medication m ON pt.medicineID = m.medicineID

## Daily Appointments Report - Summarizes all patient appointments scheduled for the day.

1. SELECT a.aptDate as "Appointment Date", d.fname || ' ' || d.lname AS "Doctor Name", p.fname || ' ' || p.lname AS "Patient Name", a.aptTime as "Appointment Time"
2. FROM appointment a
3. JOIN doctor d ON a.doctorID = d.doctorID
4. JOIN patient p ON a.patientID = p.patientID
5. WHERE a.aptDate = '06-10-2024' -- WHERE a.aptDate = CURRENT\_DATE

## Medicine Stock Report - Shows the number of medicines in stock and their availability.

1. SELECT medName AS "Medication Name", quantity, expiration\_date AS "Expiration Date"
2. FROM medication

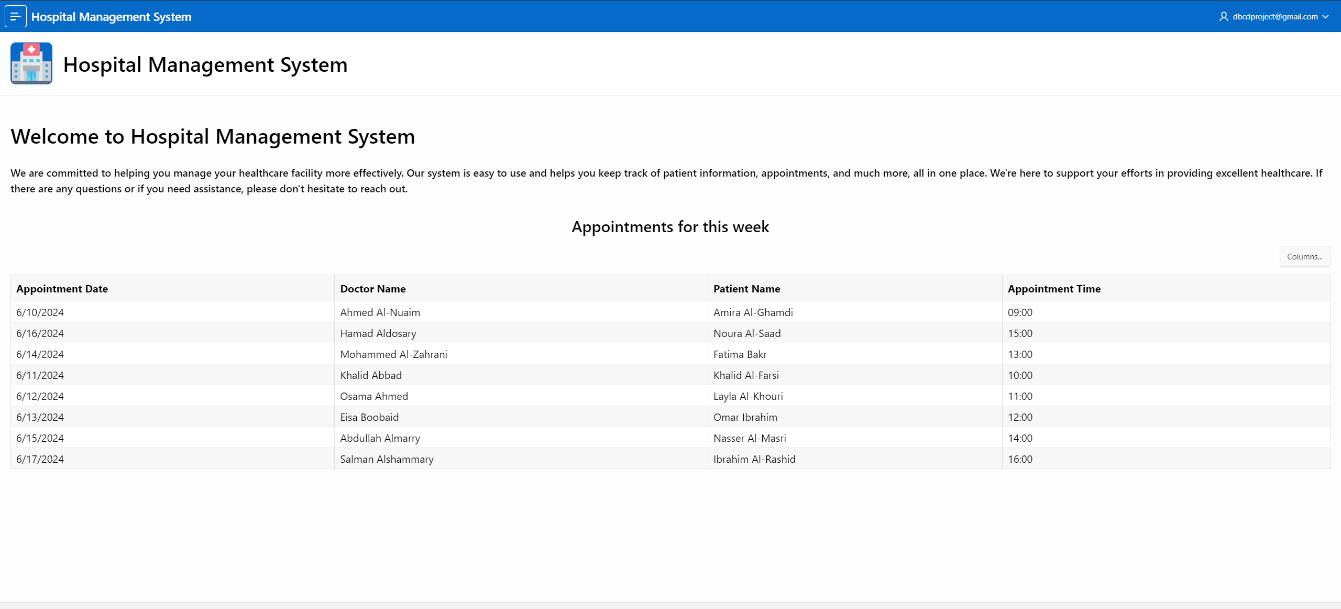
## Laboratory Test Results Report - Displays the results of all laboratory tests performed.

1. SELECT CONCAT(CONCAT( p.fname, ' ' ), p.lname) AS "Patient Name", l.test\_date, l.test\_type, l.test\_result
2. FROM lab\_test l
3. JOIN patient p ON l.patientID = p.patientID

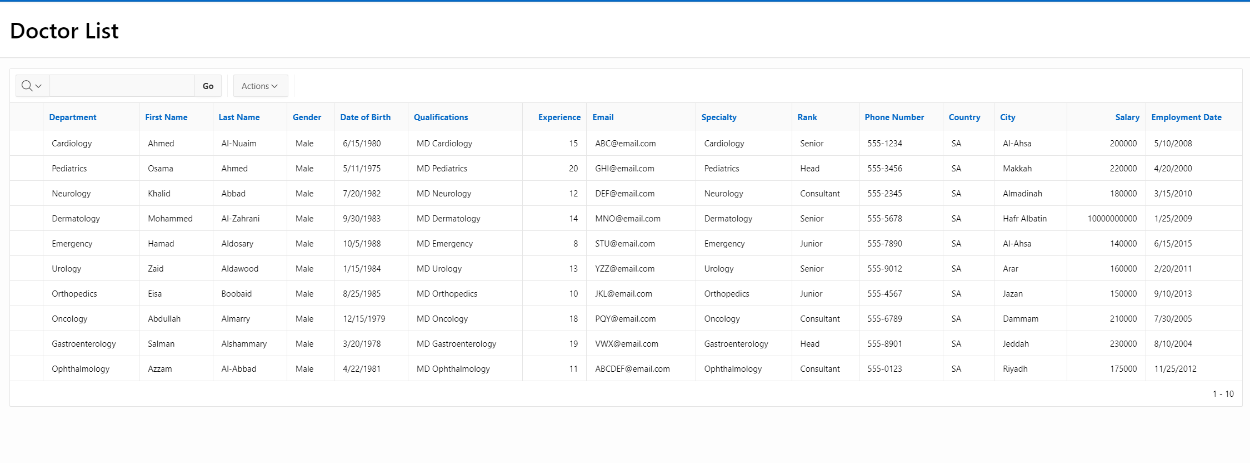
## Financial Transaction Report – Summarizes all financial transactions, including invoices, payments, and insurance.

1. SELECT p.buyer\_name, p.buyer\_phone, i.status, i.information, i.amount, i.invoice\_date, p.datePaid AS "Payment Date", p.payment\_method, p.amountPaid AS "Amount", p.confirmation\_number
2. FROM invoice i
3. LEFT JOIN payment p ON i.invoiceID = p.invoiceID

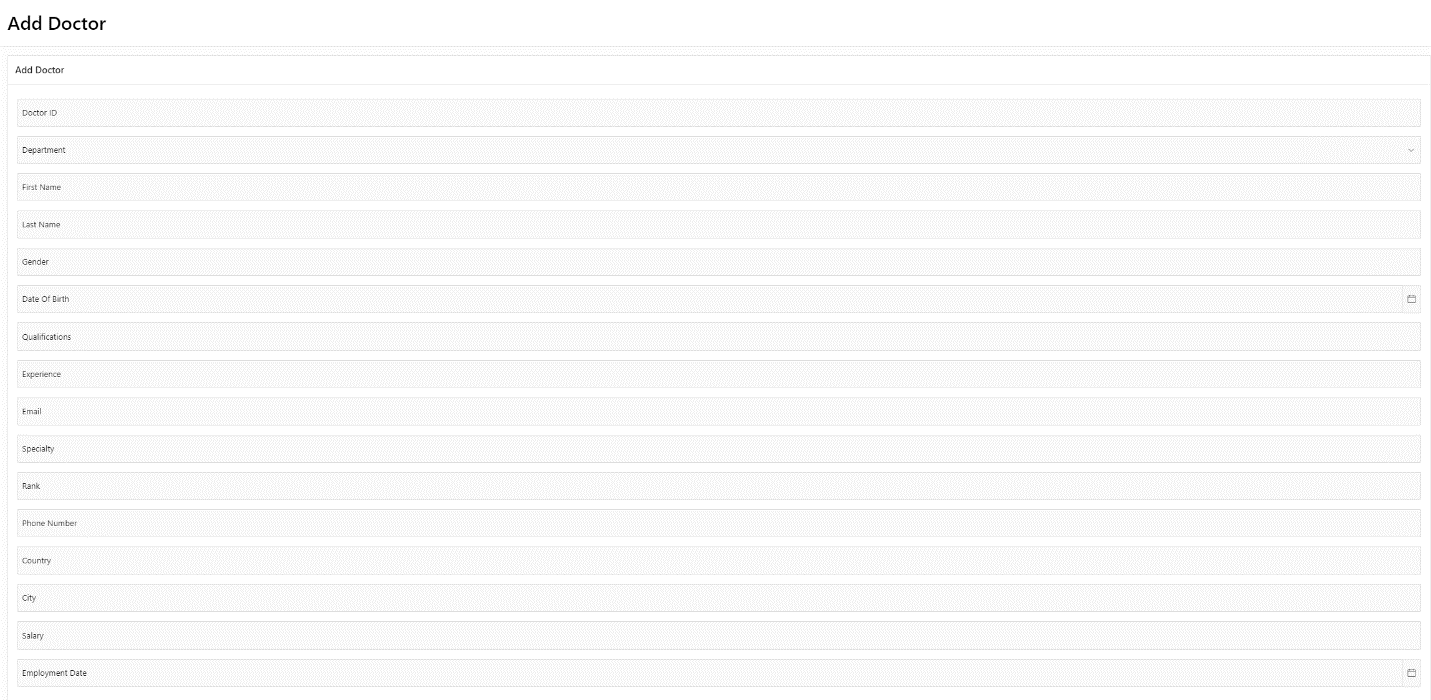
# Application Screenshots:

Home page:   


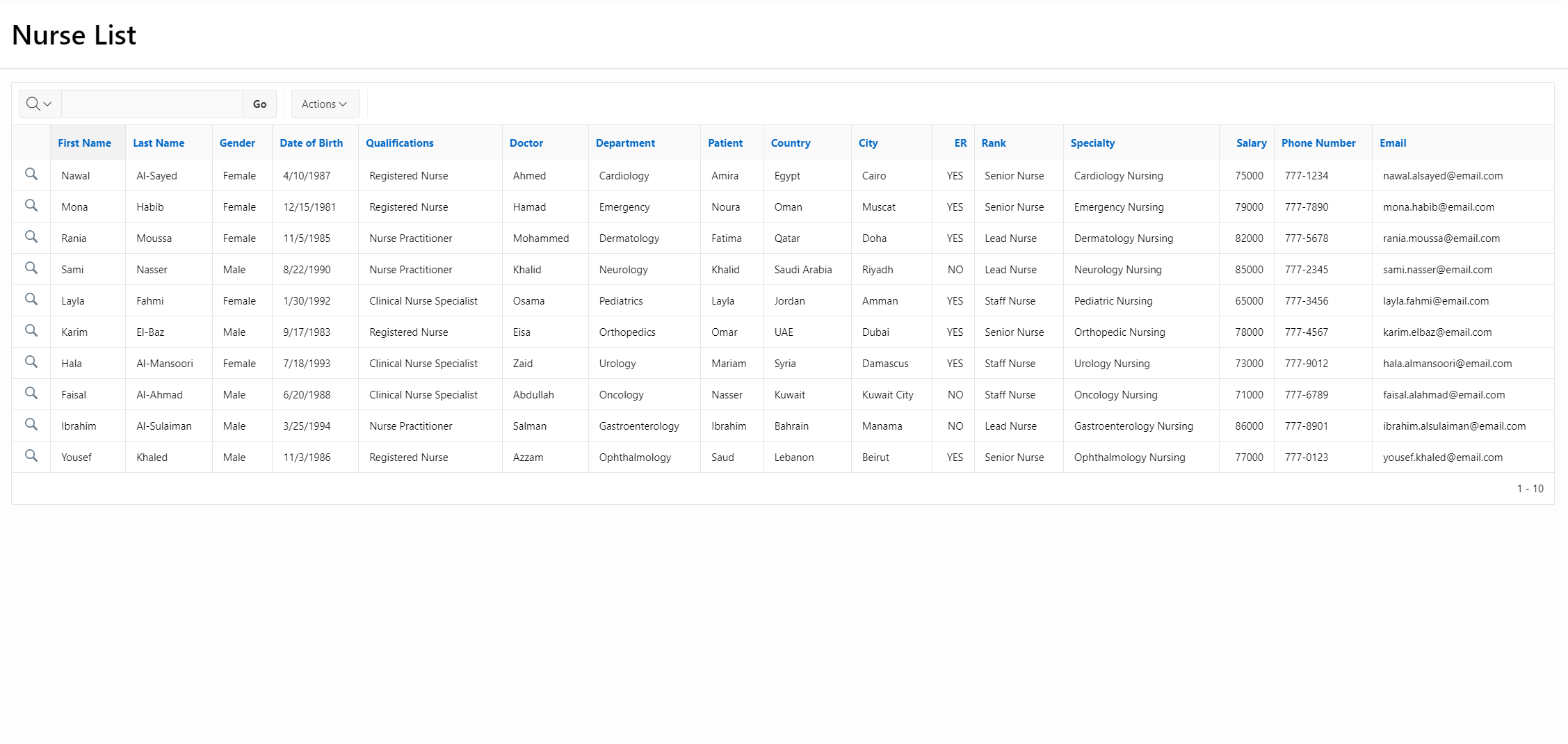
Doctor List:



Add Doctor:



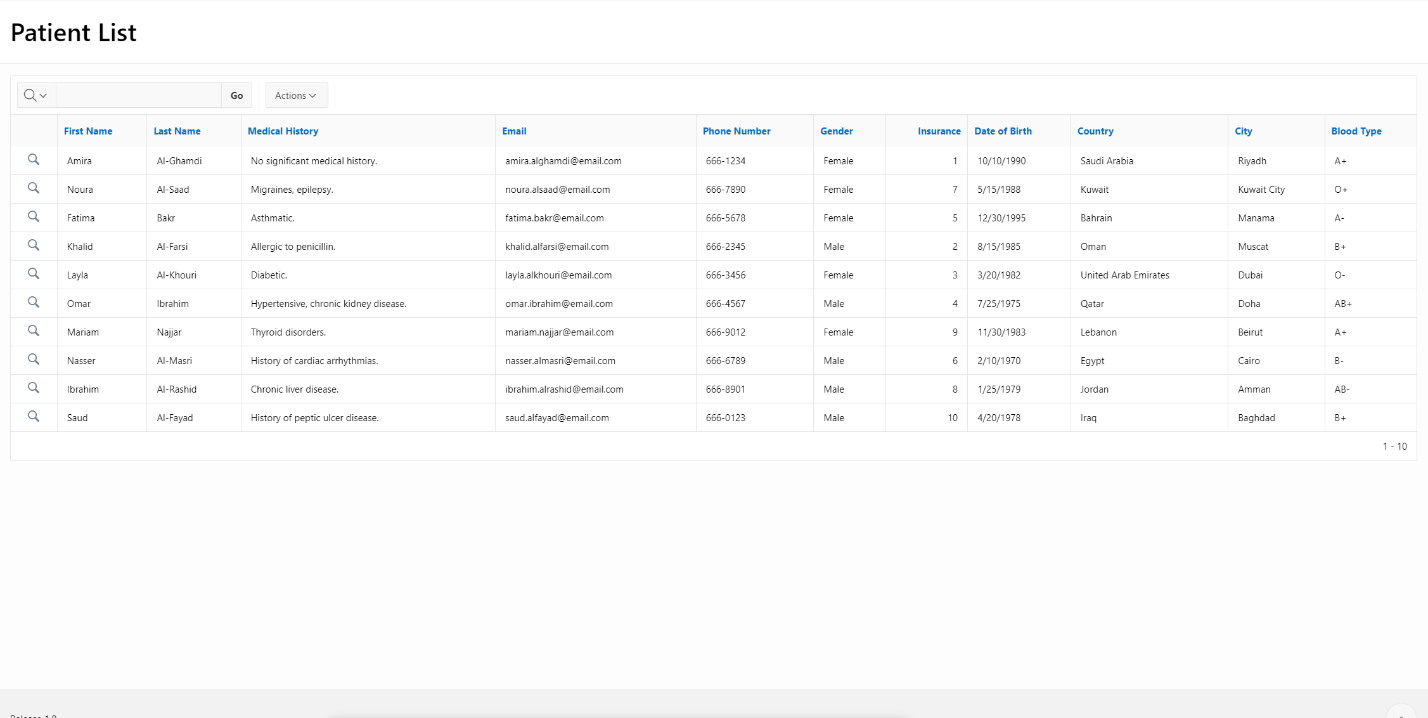
Nurse List:



Add Nurse:



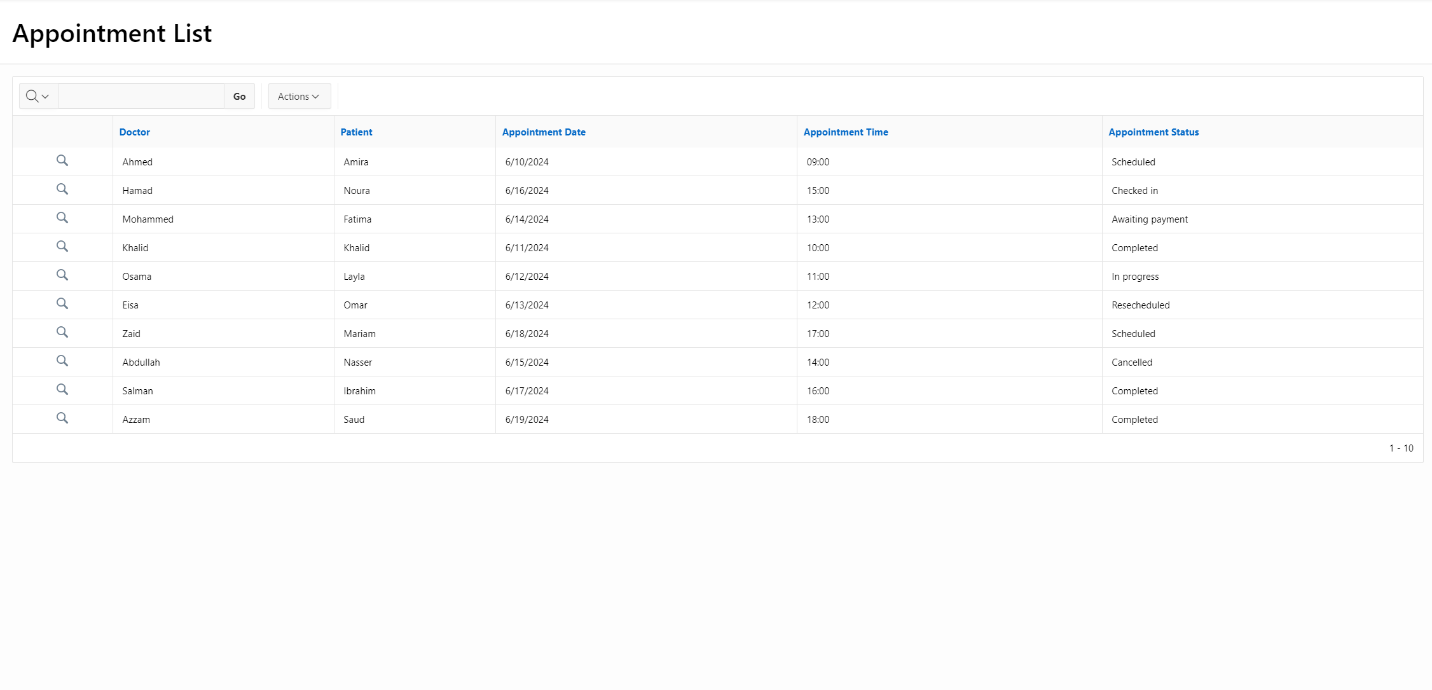
Patient List:



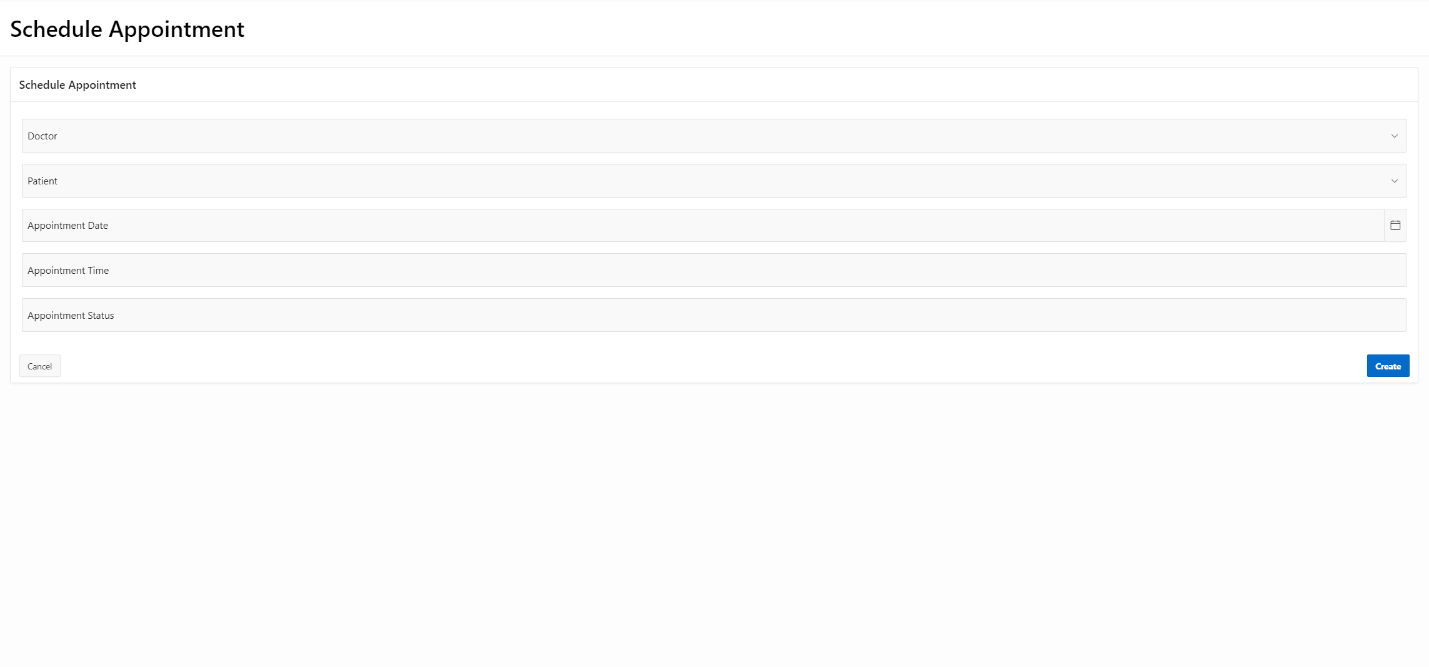
Add Patient:



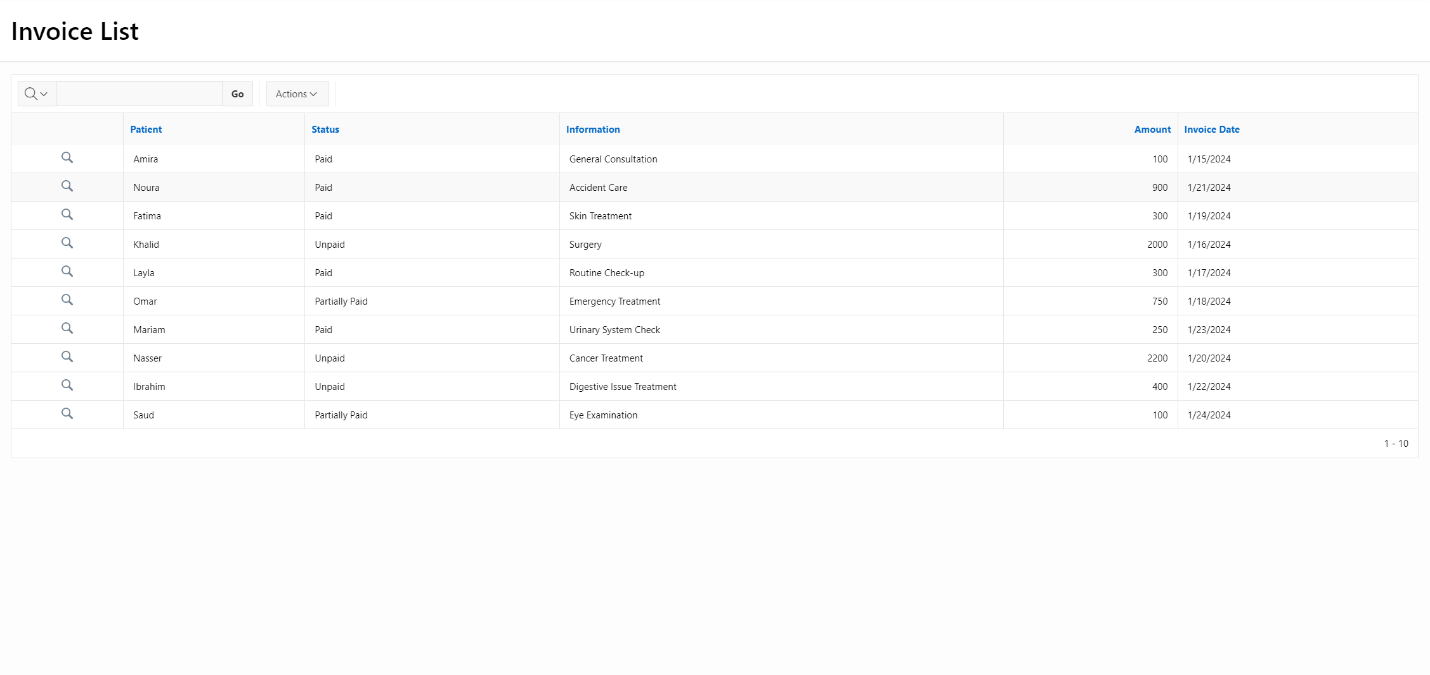
Appointment List:



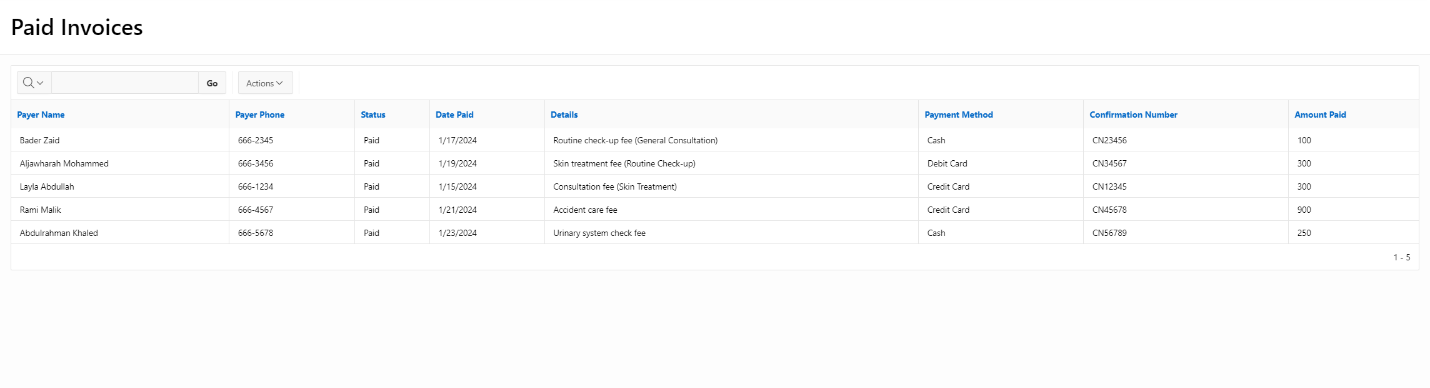
Schedule appointment:



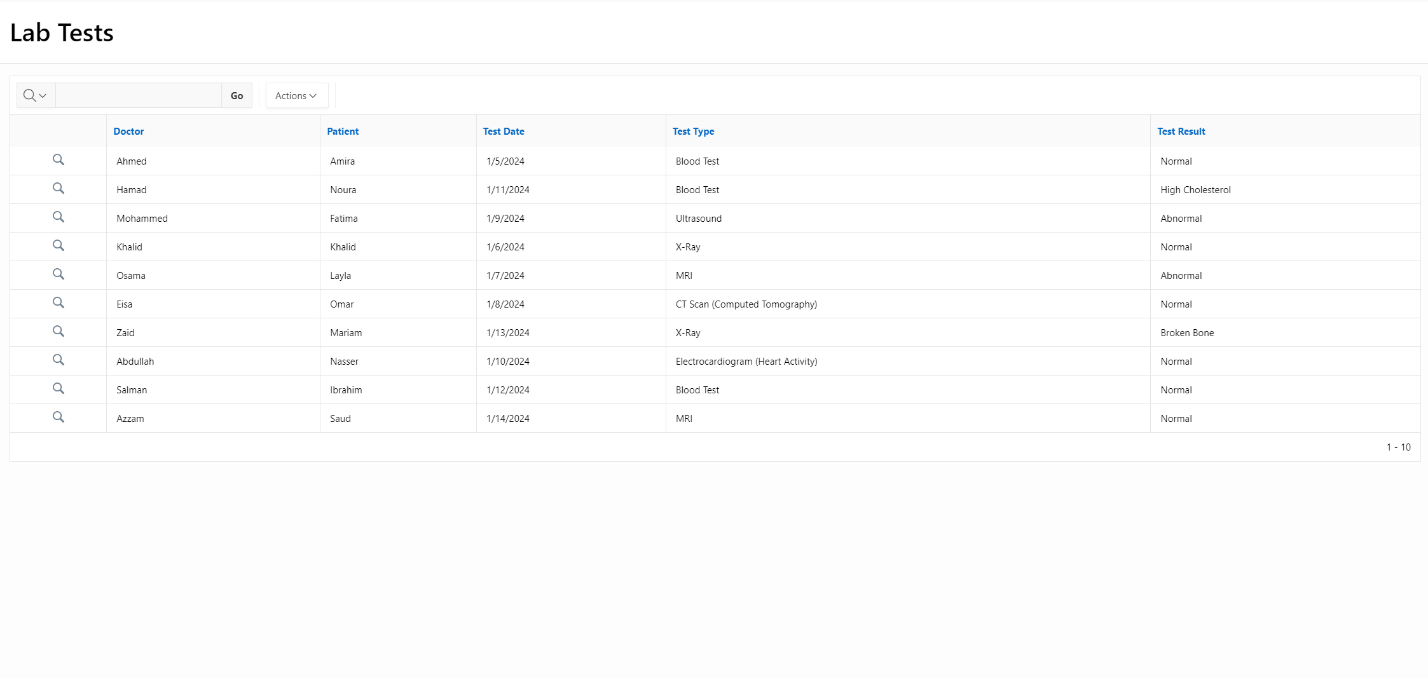
Invoice List:



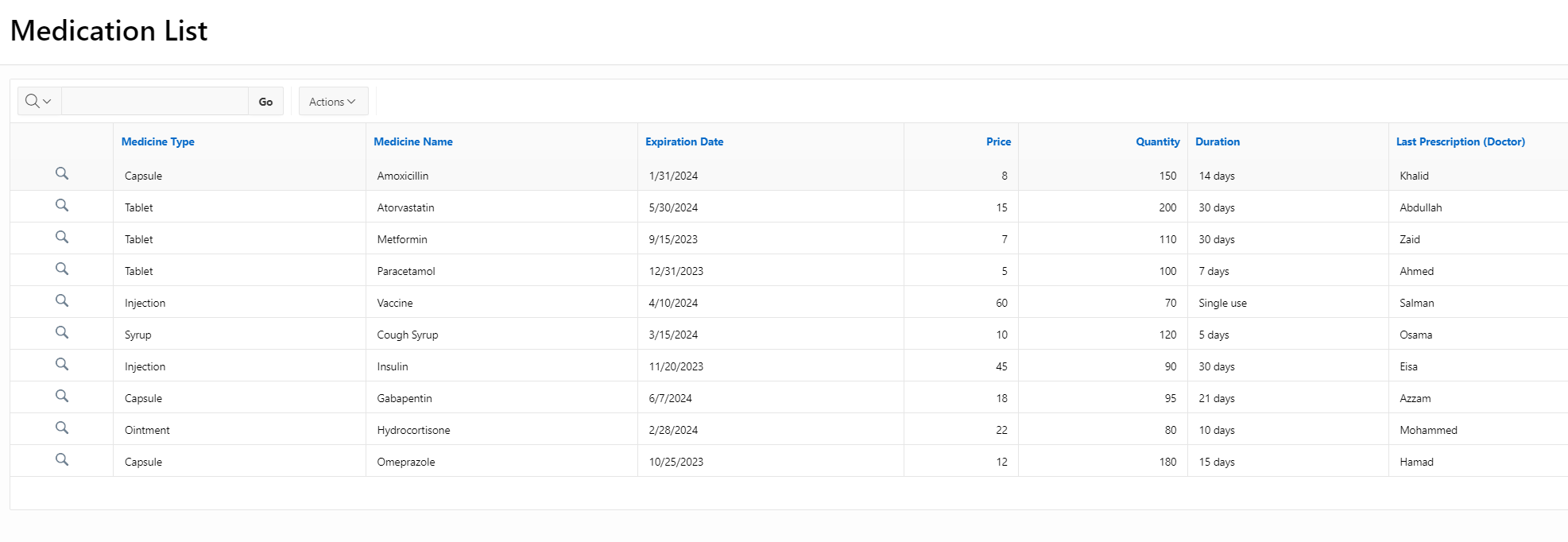
Paid Invoices:



Lap Tests:



Medication List:



# Application Credentials

Login Link: [Oracle APEX - Sign In](https://apex.oracle.com/pls/apex/f?p=4550:1:128319193353222:::::)

Workspace: Hospital\_Management\_System1

Username: DBCDPROJECT@GMAIL.COM

Password: DBCD2214\*\*\*