**Project**

**Question 3 marks (10) clo4**

Design the database for Hospital Management System.

Note: It should store patient records, including demographics, medical history, diagnoses, medications, and test results. It should also manage appointments, scheduling, and billing information.

1. Design the Data model by applying all three normal forms.

**Tables**

1. Patient
2. Appointment
3. Doctor
4. Diagnosis
5. Medication
6. TestResult
7. Billing

**Relationships**

1. **Patient-Appointment Relationship**:

* One patient can have multiple appointments.
* One appointment is associated with one patient.
* Foreign Key: PatientID in Appointment table

1. **Appointment-Doctor Relationship:**

* One appointment is associated with one doctor.
* One doctor can have multiple appointments.
* Foreign Key: DoctorID in Appointment table.

1. **Diagnosis-Doctor-Patient Relationship:**

* One diagnosis is associated with one patient and one doctor.
* One patient can have multiple diagnoses.
* One doctor can provide multiple diagnoses.
* Foreign Keys: PatientID and DoctorID in Diagnosis table.

1. **Medication-Doctor-Patient Relationship:**

* One medication record is associated with one patient and one doctor.
* One patient can have multiple medications.
* One doctor can prescribe multiple medications.
* Foreign Keys: PatientID and DoctorID in Medication table.

1. **TestResult-Patient Relationship:**

* One test result is associated with one patient.
* One patient can have multiple test results.
* Foreign Key: PatientID in TestResult table.

1. **Billing-Doctor-Patient-Appointment Relationship:**

* One billing record is associated with one patient, one doctor, and one appointment.
* One patient can have multiple billing records.
* One doctor can generate multiple billing records.
* One appointment can have one billing record.
* Foreign Keys: PatientID, DoctorID, and AppointmentID in Billing table.

**QUERIES**

1. Count the number of completed appointments.
2. Show doctors and their respective specializations.
3. Retrieve patient details for a specific appointment.
4. List all appointments along with corresponding doctor details.
5. Find patients with their diagnosis details.
6. Retrieve medication details for a specific patient.
7. List test results along with patient names.
8. Show billing details for a specific patient.
9. Show billing details for appointments that are unpaid.
10. Find patients with a specific diagnosis.
11. Show billing details for appointments that are unpaid
12. List all medications prescribed by a specific doctor.
13. Find patients with a specific diagnosis.

**IMPLEMENTATION**

-- Create Patient table

DROP TABLE Patient;

CREATE TABLE Patient (

PatientID INT PRIMARY KEY,

FirstName VARCHAR(50),

LastName VARCHAR(50),

DateOfBirth DATE,

Gender VARCHAR(10),

ContactNumber VARCHAR(15),

Address VARCHAR(100)

);

-- Insert data into Patient table

INSERT INTO Patient (PatientID, FirstName, LastName, DateOfBirth, Gender, ContactNumber, Address)

VALUES

(1, 'Muaaz', 'Hussan', '1990-05-15', 'Male', '123-456-7890', '123 Main St'),

(2, 'Mehwish', 'Nasir', '1985-08-20', 'Female', '987-654-3210', '456 Mr St'),

(3, 'Aatiqa', 'Sadiq', '1992-02-10', 'Female', '555-123-4567', '789 Jr St');

-- Create Doctor table

DROP TABLE Doctor;

CREATE TABLE Doctor (

DoctorID INT PRIMARY KEY,

FirstName VARCHAR(50),

LastName VARCHAR(50),

Specialization VARCHAR(50),

ContactNumber VARCHAR(15),

Email VARCHAR(50)

);

INSERT INTO Doctor (DoctorID, FirstName, LastName, Specialization, ContactNumber, Email)

VALUES

(1, 'Dr. Ali', 'Hussan', 'Cardiologist', '555-888-7777', 'ali@example.com'),

(2, 'Dr. Abdul', 'Ahad', 'Orthopedic Surgeon', '555-999-6666', 'ahad@example.com'),

(3, 'Dr. Dua', 'Nasir', 'Pediatrician', '555-777-8888', 'dua@example.com');

-- Create Appointment table

DROP TABLE Appointment

CREATE TABLE Appointment (

AppointmentID INT PRIMARY KEY,

PatientID INT,

DoctorID INT,

AppointmentDate DATETIME,

Reason VARCHAR(255),

Status VARCHAR(20),

FOREIGN KEY (PatientID) REFERENCES Patient(PatientID),

FOREIGN KEY (DoctorID) REFERENCES Doctor(DoctorID)

);

-- Insert data into Appointment table

INSERT INTO Appointment (AppointmentID, PatientID, DoctorID, AppointmentDate, Reason, Status)

VALUES

(101, 1, 1, '2023-01-15 10:00:00', 'Regular Checkup', 'Scheduled'),

(102, 2, 2, '2023-02-20 14:30:00', 'Knee Pain', 'Completed'),

(103, 3, 3, '2023-03-05 09:45:00', 'Child Vaccination', 'Scheduled');

-- Create Diagnosis table

DROP TABLE Diagnosis;

CREATE TABLE Diagnosis (

DiagnosisID INT PRIMARY KEY,

PatientID INT,

DoctorID INT,

DiagnosisDate DATETIME,

Description VARCHAR(255),

FOREIGN KEY (PatientID) REFERENCES Patient(PatientID),

FOREIGN KEY (DoctorID) REFERENCES Doctor(DoctorID)

);

INSERT INTO Diagnosis (DiagnosisID, PatientID, DoctorID, DiagnosisDate, Description)

VALUES

(201, 1, 1, '2023-01-20', 'Healthy'),

(202, 2, 2, '2023-02-25', 'Strained Ligament'),

(203, 3, 3, '2023-03-10', 'Normal');

-- Create Medication table

DROP TABLE Medication;

CREATE TABLE Medication (

MedicationID INT PRIMARY KEY,

PatientID INT,

DoctorID INT,

MedicationName VARCHAR(50),

Dosage VARCHAR(50),

Frequency VARCHAR(50),

StartDate DATE,

EndDate DATE,

FOREIGN KEY (PatientID) REFERENCES Patient(PatientID),

FOREIGN KEY (DoctorID) REFERENCES Doctor(DoctorID)

);

INSERT INTO Medication (MedicationID, PatientID, DoctorID, MedicationName, Dosage, Frequency, StartDate, EndDate)

VALUES

(301, 1, 1, 'Aspirin', '1 tablet', 'Once a day', '2023-01-20', '2023-01-27'),

(302, 2, 2, 'Ibuprofen', '1 tablet', 'Twice a day', '2023-02-25', '2023-03-10'),

(303, 3, 3, 'Multivitamin', '1 tablet', 'Once a day', '2023-03-10', '2023-03-17');

-- Create TestResult table

DROP TABLE TestResult;

CREATE TABLE TestResult (

TestResultID INT PRIMARY KEY,

PatientID INT,

TestDate DATETIME,

TestName VARCHAR(50),

Result VARCHAR(255),

FOREIGN KEY (PatientID) REFERENCES Patient(PatientID)

);

INSERT INTO TestResult (TestResultID, PatientID, TestDate, TestName, Result)

VALUES

(401, 1, '2023-01-25', 'Blood Pressure', '120/80'),

(402, 2, '2023-03-01', 'X-Ray', 'Normal'),

(403, 3, '2023-03-15', 'Blood Test', 'Healthy');

-- Create Billing table

DROP TABLE Billing;

CREATE TABLE Billing (

BillingID INT PRIMARY KEY,

PatientID INT,

DoctorID INT,

AppointmentID INT,

BillingDate DATETIME,

TotalAmount DECIMAL(10, 2),

PaymentStatus VARCHAR(20),

FOREIGN KEY (PatientID) REFERENCES Patient(PatientID),

FOREIGN KEY (DoctorID) REFERENCES Doctor(DoctorID),

FOREIGN KEY (AppointmentID) REFERENCES Appointment(AppointmentID)

);

-- Insert data into Billing table

INSERT INTO Billing (BillingID, PatientID, DoctorID, AppointmentID, BillingDate, TotalAmount, PaymentStatus)

VALUES

(501, 1, 1, 101, '2023-01-15', 150.00, 'Paid'),

(502, 2, 2, 102, '2023-02-20', 200.00, 'Paid'),

(503, 3, 3, 103, '2023-03-05', 100.00, 'Unpaid');

SELECT \* FROM Patient;

SELECT \* FROM Doctor;

SELECT \* FROM Appointment;

SELECT \* FROM Diagnosis;

SELECT \* FROM Medication;

SELECT \* FROM TestResult;

SELECT \* FROM Billing;

-- Count the number of completed appointments

SELECT COUNT(\*) AS NumberOfReturns

FROM Appointment

WHERE Status = 'Completed';

--Show doctors and their respective specializations:

SELECT Doctor.FirstName, Doctor.LastName, Doctor.Specialization

FROM Doctor;

--Retrieve patient details for a specific appointment:

SELECT patient.\*, a.AppointmentDate, a.Reason,a.Status

FROM Patient

JOIN Appointment a ON patient.PatientID = a.PatientID

WHERE a.AppointmentID = 101;

--List all appointments along with corresponding doctor details:

SELECT Appointment.\*, Doctor.FirstName AS DoctorFirstName, Doctor.LastName AS DoctorLastName, Doctor.Specialization

FROM Appointment

JOIN Doctor ON Appointment.DoctorID = Doctor.DoctorID;

--Find patients with their diagnosis details:

SELECT Patient.\*, Diagnosis.DiagnosisDate, Diagnosis.Description

FROM Patient

LEFT JOIN Diagnosis ON Patient.PatientID = Diagnosis.PatientID;

--Retrieve medication details for a specific patient:

SELECT Medication.\*, Doctor.FirstName AS DoctorFirstName, Doctor.LastName AS DoctorLastName

FROM Medication

JOIN Doctor ON Medication.DoctorID = Doctor.DoctorID

WHERE Medication.PatientID = 1;

--List test results along with patient names:

SELECT TestResult.\*, Patient.FirstName AS PatientFirstName, Patient.LastName AS PatientLastName

FROM TestResult

JOIN Patient ON TestResult.PatientID = Patient.PatientID;

--Show billing details for a specific patient:

SELECT Billing.\*, Patient.FirstName AS PatientFirstName, Patient.LastName AS PatientLastName

FROM Billing

JOIN Patient ON Billing.PatientID = Patient.PatientID

WHERE Billing.PatientID = 1;

--Show billing details for appointments that are unpaid:

SELECT Billing.\*, Patient.FirstName AS PatientFirstName, Patient.LastName AS PatientLastName

FROM Billing

JOIN Patient ON Billing.PatientID = Patient.PatientID

WHERE Billing.PaymentStatus = 'Unpaid';

--Find patients with a specific diagnosis:

SELECT Patient.\*, Diagnosis.DiagnosisDate, Diagnosis.Description

FROM Patient

JOIN Diagnosis ON Patient.PatientID = Diagnosis.PatientID

WHERE Diagnosis.Description = 'Strained Ligament';

--Show billing details for appointments that are unpaid:

SELECT Billing.\*, Patient.FirstName AS PatientFirstName, Patient.LastName AS PatientLastName

FROM Billing

JOIN Patient ON Billing.PatientID = Patient.PatientID

WHERE Billing.PaymentStatus = 'Unpaid';

--List all medications prescribed by a specific doctor:

SELECT Medication.\*, Patient.FirstName AS PatientFirstName, Patient.LastName AS PatientLastName

FROM Medication

JOIN Patient ON Medication.PatientID = Patient.PatientID

WHERE Medication.DoctorID = 1;

--Find patients with a specific diagnosis:

SELECT Patient.\*, Diagnosis.DiagnosisDate, Diagnosis.Description

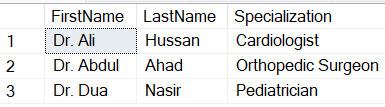
FROM Patient

JOIN Diagnosis ON Patient.PatientID = Diagnosis.PatientID

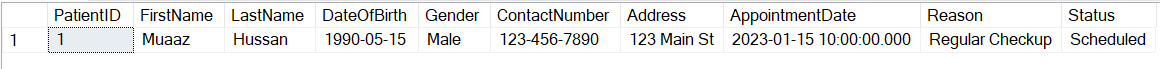
WHERE Diagnosis.Description = 'Strained Ligament';

**OUTPUTS**

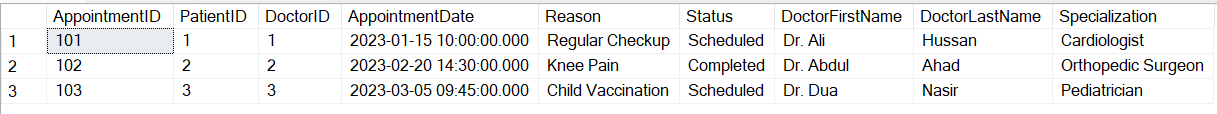
QUERY #1: QUERY #2:

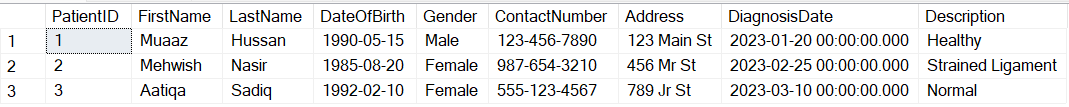
QUERY #3:



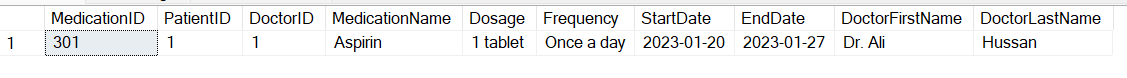
QUERY #4:

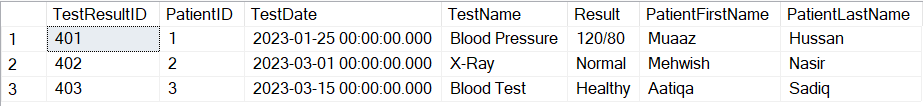


QUERY #5:



QUERY #6:

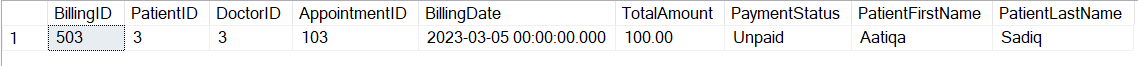
QUERY #7:



QUERY #8:



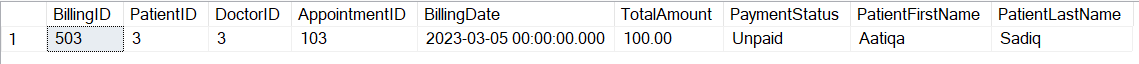
QUERY #9:



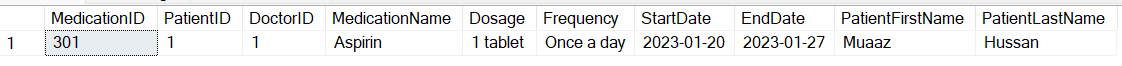
QUERY #10:



QUERY #11:



QUERY #12:



QUERY #13:



**SCHEMA MODEL**

|  |
| --- |
| **Patient** |
| PatientID (PK) |
| FirstName |
| LastName |
| DateOfBirth |
| Gender |
| ContactNumber |
| Address |

|  |
| --- |
| **Doctor** |
| DoctorID (PK) |
| FirstName |
| LastName |
| Specialization |
| ContactNumber |
| Email |

|  |
| --- |
| **Appointment** |
| AppointmentID (PK) |
| PatientID (FK) |
| DoctorID (FK) |
| AppointmentDate |
| Reason |
| Status |

|  |
| --- |
| **Billing** |
| BillingID (PK) |
| PatientID (FK) |
| DoctorID (FK) |
| AppointmentID (FK) |
| TotalAmount |
| PaymentStatus |

|  |
| --- |
| **Diagnosis** |
| DiagnosisID (PK) |
| PatientID (FK) |
| DoctorID (FK) |
| DiagnosisDate |
| Description |

|  |
| --- |
| **TestResult** |
| TestResultID (PK) |
| PatientID (FK) |
| TestDate |
| TestName |
| Result |

|  |
| --- |
| **Medication** |
| MedicationID (PK) |
| PatientID (FK) |
| DoctorID (FK) |
| MedicationName |
| Dosage |
| Frequency |
| StartDate |
| EndDate |