**Task 6 – Doctor Patient Appointment System**

CREATE DATABASE Appointment\_System;

USE Appointment\_System;

**Create the Tables -**

**Departments Table –**

CREATE TABLE Departments (

department\_id INT PRIMARY KEY AUTO\_INCREMENT,

department\_name VARCHAR(100)not null

);

**Specialties Table –**

CREATE TABLE Specialties ( specialty\_id INT PRIMARY KEY, specialty\_name VARCHAR(100) );

INSER DATA IN SPECIALITIES

**Doctors Table –**

CREATE TABLE Doctors (

doctor\_id INT PRIMARY KEY AUTO\_INCREMENT,

first\_name VARCHAR(100) not null,

last\_name VARCHAR(100) not null,

email VARCHAR(100) UNIQUE,

phone VARCHAR(20) ,

department\_id INT,

specialty\_id INT,

joining\_date DATE,

FOREIGN KEY (department\_id) REFERENCES Departments(department\_id),

FOREIGN KEY (specialty\_id) REFERENCES Specialties(specialty\_id)

);

**Patients Table –**

CREATE TABLE Patients (

patient\_id INT PRIMARY KEY AUTO\_INCREMENT,

first\_name VARCHAR(100) NOT NULL,

last\_name VARCHAR(100) NOT NULL,

email VARCHAR(100) UNIQUE,

phone VARCHAR(20),

date\_of\_birth DATE,

gender VARCHAR(10),

address TEXT

);

**Appointments Table –**

CREATE TABLE Appointments (

appointment\_id INT PRIMARY KEY AUTO\_INCREMENT,

doctor\_id INT ,

patient\_id INT,

appointment\_date DATETIME,

reason VARCHAR (300),

status VARCHAR(20),

FOREIGN KEY (doctor\_id) REFERENCES Doctors(doctor\_id),

FOREIGN KEY (patient\_id) REFERENCES Patients(patient\_id)

);

**Payments Table –**

CREATE TABLE Payments (

payment\_id INT PRIMARY KEY AUTO\_INCREMENT,

appointment\_id INT ,

payment\_date DATE,

payment\_amount DECIMAL(10, 2),

payment\_method VARCHAR(20),

FOREIGN KEY (appointment\_id) REFERENCES Appointments(appointment\_id)

);

**Insert data in the Tables –**

**Departments Table –**

1. INSERT INTO Departments (department\_name)

VALUES

('Cardiology'),

('Neurology'),

('Orthopedics'),

('Pediatrics'),

('General Medicine');

**Specialties Table –**

INSERT INTO Specialties (specialty\_id, specialty\_name)

VALUES

(1, 'Cardiac Surgery'),

(2, 'Neurophysiology'),

(3, 'Joint Replacement'),

(4, 'Child Health'),

(5, 'General Practice');

**Doctors Table –**

INSERT INTO Doctors (first\_name, last\_name, email, phone, department\_id, specialty\_id, joining\_date)

VALUES

('Amit', 'Sharma', 'amit.sharma@example.in', '9876543210', 1, 1, '2023-03-15'),

('Priya', 'Raj', 'priya.raj@example.in', '9876543211', 2, 2, '2022-08-20'),

('Ravi', 'Patel', 'ravi.patel@example.in', '9876543212', 3, 3, '2021-05-10'),

('Neha', 'Gupta', 'neha.gupta@example.in', '9876543213', 4, 4, '2020-07-25'),

('Arun', 'Mehta', 'arun.mehta@example.in', '9876543214', 5, 5, '2019-12-01');

**Patients Table –**

INSERT INTO Patients (first\_name, last\_name, email, phone, date\_of\_birth, gender, address)

VALUES

('Rohan', 'Singh', 'rohan.singh@example.in', '9988776655', '1995-06-15', 'Male', 'Mumbai, Maharashtra'),

('Sneha', 'Kumar', 'sneha.kumar@example.in', '9988776656', '1988-11-20', 'Female', 'Delhi, Delhi'),

('Anil', 'Reddy', 'anil.reddy@example.in', '9988776657', '1978-03-10', 'Male', 'Hyderabad, Telangana'),

('Pooja', 'Iyer', 'pooja.iyer@example.in', '9988776658', '1992-09-25', 'Female', 'Bengaluru, Karnataka'),

('Vikram', 'Chowdhury', 'vikram.chowdhury@example.in', '9988776659', '1985-01-05', 'Male', 'Kolkata, West Bengal');

**Appointments Table –**

INSERT DATA IN APPOINTMENTS TABLE

INSERT INTO Appointments (doctor\_id, patient\_id, appointment\_date, reason, status)

VALUES

(1, 1, '2025-01-12 10:00:00', 'Heart checkup', 'Scheduled'),

(2, 2, '2025-01-12 11:00:00', 'Migraine treatment', 'Scheduled'),

(3, 3, '2025-01-13 09:30:00', 'Knee pain', 'Scheduled'),

(4, 4, '2025-01-13 10:30:00', 'Routine checkup for child', 'Scheduled'),

(5, 5, '2025-01-14 12:00:00', 'General health checkup', 'Scheduled');

**Payments Table –**

INSERT DATA IN PAYMENTS TABLE

INSERT INTO Payments (appointment\_id, payment\_date, payment\_amount, payment\_method)

VALUES

(1, '2025-01-12', 1500.00, 'Credit Card'),

(2, '2025-01-12', 2000.00, 'UPI'),

(3, '2025-01-13', 2500.00, 'Debit Card'),

(4, '2025-01-13', 1200.00, 'Cash'),

(5, '2025-01-14', 1000.00, 'Net Banking');

**SQL Queries for the Case Study**

1. **Find the Total Number of Appointments for Each Doctor**

SELECT doctors.first\_name,doctors.last\_name,COUNT(appointments.doctor\_id)AS Appointments FROM doctors JOIN appointments on appointments.doctor\_id = doctors.doctor\_id

GROUP by doctors.doctor\_id;

1. **List All Patients Who Have an Appointment with a Specific Doctor (e.g., Dr. John Smith)**

SELECT patients.first\_name,patients.last\_name , concat(doctors.first\_name ," ",doctors.last\_name )as Appointment\_Doctor FROM patients

JOIN appointments on appointments.patient\_id = patients.patient\_id

JOIN doctors on doctors.doctor\_id = appointments.doctor\_id

WHERE doctors.first\_name ='Amit'

GROUP by patients.patient\_id;

1. **Find the Number of Appointments Scheduled in a Specific Department**

SELECT departments.department\_name,COUNT(appointments.appointment\_id) as Schedule\_Appointmnet FROM departments

JOIN doctors on doctors.department\_id = departments.department\_id

JOIN appointments on appointments.doctor\_id = doctors.doctor\_id

WHERE appointments.status = "Scheduled"

GROUP by departments.department\_name;

1. **Find the Most Popular Specialty Based on Number of Appointments**

SELECT specialties.specialty\_name , COUNT(appointments.appointment\_id) FROM specialties

JOIN doctors on doctors.specialty\_id = specialties.specialty\_id

JOIN appointments on appointments.doctor\_id = doctors.doctor\_id

GROUP by specialties.specialty\_name;

1. **Get the Total Payment Amount for All Completed Appointments**

SELECT appointments.reason,SUM(payments.payment\_amount)as total\_payment FROM payments

JOIN appointments on appointments.appointment\_id = payments.appointment\_id

WHERE appointments.status = "Compelete";

1. **Find the Number of Patients Seen by Each Doctor**

SELECT doctors.first\_name,doctors.last\_name ,COUNT(doctors.doctor\_id) FROM

doctors JOIN appointments on appointments.doctor\_id = doctors.doctor\_id

GROUP by doctors.doctor\_id;

1. **List All Patients Who Have Missed Their Appointments (Status 'Cancelled')**

SELECT CONCAT( patients.first\_name,patients.last\_name)AS Patients\_Name ,appointments.status FROM patients

JOIN appointments on appointments.patient\_id = patients.patient\_id

WHERE appointments.status = "Cancelled";

1. **Find the Total Number of Appointments for Each Status (Scheduled, Completed, Cancelled)**

SELECT appointments.status, COUNT(appointments.appointment\_id) FROM appointments

GROUP by appointments.status;

1. **Get the Average Payment Amount for Completed Appointments**

SELECT AVG(payments.payment\_amount)as avg\_payment FROM payments

JOIN appointments on payments.appointment\_id = appointments.appointment\_id

WHERE appointments.status = "Completed";

1. **Find the Doctor with the Highest Number of Appointments**

SELECT doctors.first\_name,doctors.last\_name,COUNT(\*) as total\_appointment FROM appointments

JOIN doctors on doctors.doctor\_id = appointments.doctor\_id

GROUP by doctors.doctor\_id

ORDER by total\_appointment DESC;