Problem 9: Vehicle Service Center

ER Diagram Description:

Entities: Vehicles, ServiceTypes, Appointments

Relationships: A vehicle can have multiple appointments; each appointment is linked to a service type.

Table Creation:

```
CREATE TABLE Vehicles (
    VehicleID INT PRIMARY KEY,
    OwnerName VARCHAR(100),
    Model VARCHAR(50)
);

CREATE TABLE ServiceTypes (
    ServiceID INT PRIMARY KEY,
    Description VARCHAR(100)
);

CREATE TABLE Appointments (
    AppointmentID INT PRIMARY KEY,
    VehicleID INT,
    ServiceID INT,
    ServiceDate DATE,
    FOREIGN KEY (VehicleID) REFERENCES Vehicles(VehicleID),
    FOREIGN KEY (ServiceID) REFERENCES ServiceTypes(ServiceID)
);
```

Sample Data:

```
INSERT INTO Vehicles VALUES

(1, 'Ramesh Kumar', 'Swift'),

(2, 'Geeta Singh', 'i20'),

(3, 'Akhil Sharma', 'Creta');


INSERT INTO ServiceTypes VALUES

(501, 'Oil Change'),

(502, 'Tyre Replacement'),

(503, 'Battery Check');


INSERT INTO Appointments VALUES

(1001, 1, 501, '2025-03-01'),

(1002, 2, 502, '2025-03-05'),

(1003, 1, 503, '2025-04-02'),

(1004, 3, 501, '2025-04-10');
```

Queries:

```
SELECT * FROM Vehicles WHERE Model = 'Swift';

SELECT * FROM Appointments WHERE ServiceID = 501;

SELECT ServiceID, COUNT(*) AS TotalAppointments FROM Appointments GROUP BY ServiceID;

SELECT VehicleID FROM Appointments WHERE ServiceID = 501

INTERSECT

SELECT VehicleID FROM Appointments WHERE ServiceID = 503;

SELECT Vehicles.OwnerName, ServiceTypes.Description

FROM Appointments
```

JOIN Vehicles ON Appointments.VehicleID = Vehicles.VehicleID

JOIN ServiceTypes ON Appointments.ServiceID = ServiceTypes.ServiceID;

Problem 10: Hotel Reservation System

ER Diagram Description:

Entities: Rooms, Guests, Reservations

Relationships: Guests make reservations for rooms.

Table Creation:

```
CREATE TABLE Rooms (
    RoomID INT PRIMARY KEY,
    Type VARCHAR(50),
    Price DECIMAL(10,2)
);

CREATE TABLE Guests (
    GuestID INT PRIMARY KEY,
    Name VARCHAR(100),
    Phone VARCHAR(15)
);

CREATE TABLE Reservations (
    ReservationID INT PRIMARY KEY,
    RoomID INT,
    GuestID INT,
```

```
    CheckInDate DATE,
    CheckOutDate DATE,
    FOREIGN KEY (RoomID) REFERENCES Rooms(RoomID),
    FOREIGN KEY (GuestID) REFERENCES Guests(GuestID)
);
```

Sample Data:

```
INSERT INTO Rooms VALUES
(1, 'Deluxe', 2500.00),
(2, 'Standard', 1800.00),
(3, 'Suite', 3200.00);

INSERT INTO Guests VALUES
(101, 'Kunal Rao', '9999988888'),
(102, 'Meena Iyer', '8888877777'),
(103, 'Raj Malhotra', '7777766666');

INSERT INTO Reservations VALUES
(201, 1, 101, '2025-03-10', '2025-03-15'),
(202, 2, 102, '2025-03-18', '2025-03-20'),
(203, 3, 103, '2025-04-05', '2025-04-10');
```

Queries:

```
SELECT * FROM Rooms WHERE Price > 2000;
SELECT * FROM Reservations WHERE MONTH(CheckInDate) = 3;
SELECT RoomID, COUNT(*) AS TotalBookings FROM Reservations GROUP BY RoomID;
SELECT GuestID FROM Reservations WHERE RoomID = 1
```

```
UNION

SELECT GuestID FROM Reservations WHERE RoomID = 2;

SELECT Guests.Name, Rooms.Type

FROM Reservations

JOIN Guests ON Reservations.GuestID = Guests.GuestID

JOIN Rooms ON Reservations.RoomID = Rooms.RoomID;
```

Problem 11: Gym Management System

ER Diagram Description:

Entities: Members, Trainers, Subscriptions

Relationships: Members subscribe under trainers for a type of subscription.

Table Creation:

```
CREATE TABLE Members (
    MemberID INT PRIMARY KEY,
    Name VARCHAR(100),
    Age INT
);

CREATE TABLE Trainers (
    TrainerID INT PRIMARY KEY,
    Name VARCHAR(100),
    Expertise VARCHAR(100)
);
```

```
CREATE TABLE Subscriptions (
    MemberID INT,
    TrainerID INT,
    Type VARCHAR(20),
    StartDate DATE,
    PRIMARY KEY (MemberID, TrainerID),
    FOREIGN KEY (MemberID) REFERENCES Members(MemberID),
    FOREIGN KEY (TrainerID) REFERENCES Trainers(TrainerID)
);
```

Sample Data:

```
INSERT INTO Members VALUES
(1, 'Sahil Rana', 35),
(2, 'Priya Desai', 28),
(3, 'Ankit Verma', 40);

INSERT INTO Trainers VALUES
(101, 'Rohit Kumar', 'Cardio'),
(102, 'Anita Shah', 'Weight Training'),
(103, 'Kabir Jain', 'Yoga');

INSERT INTO Subscriptions VALUES
(1, 101, 'Premium', '2025-02-01'),
(2, 102, 'Basic', '2025-03-05'),
(3, 101, 'Basic', '2025-03-15'),
(1, 102, 'Basic', '2025-03-20');
```

Queries:

```
SELECT * FROM Members WHERE Age > 30;

SELECT * FROM Trainers WHERE Expertise = 'Cardio';

SELECT Type, COUNT(*) AS MemberCount FROM Subscriptions GROUP BY Type;

SELECT MemberID FROM Subscriptions WHERE Type = 'Basic'

INTERSECT

SELECT MemberID FROM Subscriptions WHERE Type = 'Premium';

SELECT Members.Name, Trainers.Name

FROM Subscriptions

JOIN Members ON Subscriptions.MemberID = Members.MemberID

JOIN Trainers ON Subscriptions.TrainerID = Trainers.TrainerID;
```