# CODING CHALLENGE

# Car Rental System:

Sql Schemas:

1. Creating table
2. **Vehicle**

CREATE TABLE Vehicle (

vehicleID INT PRIMARY KEY,

make VARCHAR(50) NOT NULL,

model VARCHAR(50) NOT NULL,

year INT NOT NULL,

dailyRate DECIMAL(10,2) NOT NULL,

status ENUM('available', 'notAvailable') NOT NULL,

passengerCapacity INT NOT NULL,

engineCapacity int NOT NULL );

1. **Customer**

CREATE TABLE Customer (

customerID INT PRIMARY KEY,

firstName VARCHAR(50) NOT NULL,

lastName VARCHAR(50) NOT NULL,

email VARCHAR(100) NOT NULL UNIQUE,

phoneNumber VARCHAR(15) NOT NULL );

1. **Lease**

CREATE TABLE Lease (

leaseID INT PRIMARY KEY,

vehicleID INT,

customerID INT,

startDate DATE NOT NULL,

endDate DATE NOT NULL,

type ENUM('DailyLease', 'MonthlyLease') NOT NULL,

Constraint Lease\_VehicleId\_fk FOREIGN KEY (vehicleID) REFERENCES Vehicle(vehicleID) ON DELETE CASCADE,

Constraint Lease\_CustomeId\_fk FOREIGN KEY (customerID) REFERENCES Customer(customerID) ON DELETE CASCADE);

1. **Payment**

CREATE TABLE Payment (

paymentID INT PRIMARY KEY,

leaseID INT,

paymentDate DATE NOT NULL,

amount DECIMAL(10,2) NOT NULL,

Constraint Payment\_leaseId\_fk FOREIGN KEY (leaseID) REFERENCES Lease(leaseID) ON DELETE CASCADE;

1. Inserting values
2. **Vehicle**

INSERT INTO Vehicle VALUES

(1, 'Toyota', 'Camry', 2022, 50.00, 'available', 4, 1450),

(2, 'Honda', 'Civic', 2023, 45.00, 'available', 7, 1500),

(3, 'Ford', 'Focus', 2022, 48.00, 'notAvailable', 4, 1400),

(4, 'Nissan', 'Altima', 2023, 52.00, 'available', 7, 1200),

(5, 'Chevrolet', 'Malibu', 2022, 47.00, 'available', 4, 1800),

(6, 'Hyundai', 'Sonata', 2023, 49.00, 'notAvailable', 7, 1400),

(7, 'BMW', '3 Series', 2023, 60.00, 'available', 7, 2499),

(8, 'Mercedes', 'C-Class', 2022, 58.00, 'available', 8, 2599),

(9, 'Audi', 'A4', 2022, 55.00, 'notAvailable', 4, 2500),

(10, 'Lexus', 'ES', 2023, 54.00, 'available', 4, 2500);

1. **Customer**

INSERT INTO Customer values

(1, 'John', 'Doe', 'johndoe@example.com', '555-555-5555'),

(2, 'Jane', 'Smith', 'janesmith@example.com', '555-123-4567'),

(3, 'Robert', 'Johnson', 'robert@example.com', '555-789-1234'),

(4, 'Sarah', 'Brown', 'sarah@example.com', '555-456-7890'),

(5, 'David', 'Lee', 'david@example.com', '555-987-6543'),

(6, 'Laura', 'Hall', 'laura@example.com', '555-234-5678'),

(7, 'Michael', 'Davis', 'michael@example.com', '555-876-5432'),

(8, 'Emma', 'Wilson', 'emma@example.com', '555-432-1098'),

(9, 'William', 'Taylor', 'william@example.com', '555-321-6547'),

(10, 'Olivia', 'Adams', 'olivia@example.com', '555-765-4321');

1. **Lease**

INSERT INTO Lease values

(1, 1, 1, '2023-01-01', '2023-01-05', 'DailyLease'),

(2, 2, 2, '2023-02-15', '2023-02-28', 'MonthlyLease'),

(3, 3, 3, '2023-03-10', '2023-03-15', 'DailyLease'),

(4, 4, 4, '2023-04-20', '2023-04-30', 'MonthlyLease'),

(5, 5, 5, '2023-05-05', '2023-05-10', 'DailyLease'),

(6, 4, 3, '2023-06-15', '2023-06-30', 'MonthlyLease'),

(7, 7, 7, '2023-07-01', '2023-07-10', 'DailyLease'),

(8, 8, 8, '2023-08-12', '2023-08-15', 'MonthlyLease'),

(9, 3, 3, '2023-09-07', '2023-09-10', 'DailyLease'),

(10, 10, 10, '2023-10-10', '2023-10-31', 'MonthlyLease');

1. **Payment**

INSERT INTO Payment values

(1, 1, '2023-01-03', 200.00),

(2, 2, '2023-02-20', 1000.00),

(3, 3, '2023-03-12', 75.00),

(4, 4, '2023-04-25', 900.00),

(5, 5, '2023-05-07', 60.00),

(6, 6, '2023-06-18', 1200.00),

(7, 7, '2023-07-03', 40.00),

(8, 8, '2023-08-14', 1100.00),

(9, 9, '2023-09-09', 80.00),

(10, 10, '2023-10-25', 1500.00);

1. Quries:
2. Update the daily rate for a Mercedes car to 68.

**Query:** Update Vehicle Set dailyRate =68 WHERE make = ‘Mercedes’;

1. Delete a specific customer and all associated leases and payments

**Query:** Delete from Customer WHERE CustomerID =1;

1. Rename the "paymentDate" column in the Payment table to "transactionDate".

**Query:** Alter TABLE Payment CHANGE paymentDate transactionDate DATE;

1. Find a specific customer by email.

**Query:** Select \* from customer WHERE email=’emma@example.com’;

1. Get active leases for a specific customer.

**Query:** SELECT \* FROM Lease

WHERE customerID = specific\_customer\_id AND endDate >= CURRENT\_DATE();

1. Find all payments made by a customer with a specific phone number.

**Query:** Select Payment.\* from Payment

JOIN Lease ON Payment.leaseID = Lease.leaseID

JOIN Customer ON Lease.customerID = Customer.customerID

WHERE Customer.phoneNumber = ' 555-432-1098 ';

1. Calculate the average daily rate of all available cars.

**Query:** Select AVG(dailyRate) FROM Vehicle WHERE status = 'available';

1. Find the car with the highest daily rate.

**Query:** Select \* from Vehicle ORDER BY dailyRate DESC LIMIT 1;

1. Retrieve all cars leased by a specific customer.

**Query:** Select\* from Vehicle JOIN Lease ON Vehicle.vehicleID = Lease.vehicleID WHERE Lease.customerID = 2;

1. Find the details of the most recent lease.

**Query:** Select \* from Lease ORDER BY endDate DESC LIMIT 1;

1. List all payments made in the year 2023.

**Query:** Select \* from Payment WHERE year(transactionDate) = 2023;

1. Retrieve customers who have not made any payments.

**Query:** Select\* from Customer WHERE customerID NOT IN ( Select Lease.customerID FROM Lease JOIN Payment ON Lease.leaseID = Payment.leaseID );

1. Retrieve Car Details and Their Total Payments.

**Query:** Select Vehicle.\*, Sum(Payment.amount) AS totalPayments from Vehicle

JOIN Lease ON Vehicle.vehicleID = Lease.vehicleID

JOIN Payment ON Lease.leaseID = Payment.leaseID

GROUP BY Vehicle.vehicleID;

1. Calculate Total Payments for Each Customer.

**Query:** Select Customer.\*, SUM(Payment.amount) AS totalPayments FROM Customer

JOIN Lease ON Customer.customerID = Lease.customerID

JOIN Payment ON Lease.leaseID = Payment.leaseID

GROUP BY Customer.customerID;

1. List Car Details for Each Lease.

**Query:** Select Lease.\*, Vehicle.\* FROM Lease JOIN Vehicle ON Lease.vehicleID = Vehicle.vehicleID;

1. Retrieve Details of Active Leases with Customer and Car Information.

**Query:** Select Lease.\*, Customer.\*, Vehicle.\* FROM Lease

JOIN Customer ON Lease.customerID = Customer.customerID

JOIN Vehicle ON Lease.vehicleID = Vehicle.vehicleID WHERE Lease.endDate >= CURRENT\_DATE();

1. Find the Customer Who Has Spent the Most on Leases.

**Query:** Select Customer.\*, Sum(Payment.amount) as totalspent FROM Customer

JOIN Lease ON Customer.customerID = Lease.customerID

JOIN Payment ON Lease.leaseID = Payment.leaseID

GROUP BY Customer.customerID ORDER BY totalspent

DESC LIMIT 1;

1. List All Cars with Their Current Lease Information.

**Query:** Select Vehicle.\*, Lease.\* FROM Vehicle LEFT JOIN Lease ON Vehicle.vehicleID = Lease.vehicleID AND Lease.endDate >= CURRENT\_DATE();