```
--Coding Challenge - Car Rental System - SQL
CREATE DATABASE VehicleLeasingDB;
USE VehicleLeasingDB;
-- Vehicle Table
CREATE TABLE Vehicle (
    carID INT PRIMARY KEY IDENTITY(1,1),
    make VARCHAR(50) NOT NULL,
    model VARCHAR(50) NOT NULL,
    vear INT NOT NULL,
    dailyRate DECIMAL(10,2) NOT NULL,
    status VARCHAR(20) CHECK (status IN ('available',
'notAvailable')),
    passengerCapacity INT NOT NULL,
    engineCapacity INT NOT NULL
);
-- Customer Table
CREATE TABLE Customer (
    customerID INT PRIMARY KEY IDENTITY(1,1),
    firstName VARCHAR(50) NOT NULL,
    lastName VARCHAR(50) NOT NULL,
    email VARCHAR(100) UNIQUE NOT NULL,
    phoneNumber VARCHAR(15) UNIQUE NOT NULL
);
-- Lease Table
CREATE TABLE Lease (
    leaseID INT PRIMARY KEY IDENTITY(1,1),
    carID INT NOT NULL,
    customerID INT NOT NULL,
    startDate DATE NOT NULL,
    endDate DATE NOT NULL,
    leaseType VARCHAR(20) CHECK (leaseType IN ('Daily',
'Monthly')),
    FOREIGN KEY (carID) REFERENCES Vehicle(carID) ,
    FOREIGN KEY (customerID) REFERENCES Customer(customerID)
);
-- Payment Table
CREATE TABLE Payment (
    paymentID INT PRIMARY KEY IDENTITY(1,1),
    leaseID INT NOT NULL,
    paymentDate DATE NOT NULL,
    amount DECIMAL(10,2) NOT NULL,
    FOREIGN KEY (leaseID) REFERENCES Lease(leaseID)
);
```

```
-- Inserting into Vehicle Table
```

```
INSERT INTO Vehicle (make, model, year, dailyRate, status,
passengerCapacity, engineCapacity)
VALUES
('Toyota', 'Camry', 2022, 50.00, 'available', 4, 1450), ('Honda', 'Civic', 2023, 45.00, 'available', 7, 1500), ('Ford', 'Focus', 2022, 48.00, 'notAvailable', 4, 1400),
('Nissan', 'Altima', 2023, 52.00, 'available', 7, 1200),
('Chevrolet', 'Malibu', 2022, 47.00, 'available', 4, 1800), ('Hyundai', 'Sonata', 2023, 49.00, 'notAvailable', 7, 1400),
('BMW', '3 Series', 2023, 60.00, 'available', 7, 2499),
('Mercedes', 'C-Class', 2022, 58.00, 'available', 8, 2599), ('Audi', 'A4', 2022, 55.00, 'notAvailable', 4, 2500), ('Lexus', 'ES', 2023, 54.00, 'available', 4, 2500);
-- Inserting into Customer Table
INSERT INTO Customer (firstName, lastName, email, phoneNumber)
VALUES
('John', 'Doe', 'johndoe@example.com', '555-555-555'), ('Jane', 'Smith', 'janesmith@example.com', '555-123-4567'), ('Robert', 'Johnson', 'robert@example.com', '555-789-1234'),
('Sarah', 'Brown', 'sarah@example.com', '555-456-7890'), ('David', 'Lee', 'david@example.com', '555-987-6543'), ('Laura', 'Hall', 'laura@example.com', '555-234-5678'),
('Michael', 'Davis', 'michael@example.com', '555-876-5432'), ('Emma', 'Wilson', 'emma@example.com', '555-432-1098'),
('William', 'Taylor', 'william@example.com', '555-321-6547'), ('Olivia', 'Adams', 'olivia@example.com', '555-765-4321');
SELECT * FROM Vehicle;
-- Inserting into Lease Table
INSERT INTO Lease (carID, customerID, startDate, endDate,
leaseType)
VALUES
(1, 1, '2023-01-01',
                                 '2023-01-05', 'Daily'),
                                   12023-02-281,
           '2023-02-15',
(2, 2,
                                                          'Monthly'),
(3, 3, '2023-03-10',
(4, 4, '2023-04-20',
                                   '2023-03-15',
                                                          'Daily'),
                                   '2023-04-30',
                                                         'Monthly'),
                                  '2023-05-10',
'2023-06-30',
(5, 5,
           '2023-05-05'
                                                          'Daily'),
           '2023-06-15',
(4, 3,
                                                          'Monthly'),
(7, 7, '2023-07-01', (8, 8, '2023-08-12',
(7, 7, '2023-07-01', '2023-07-10', 'Daily'), (8, 8, '2023-08-12', '2023-08-15', 'Monthly' (3, 3, '2023-09-07', '2023-09-10', 'Daily'),
                                                          'Monthly'),
(10, 10, '2023-10-10', '2023-10-31', 'Monthly');
```

-- Inserting into Payment Table

```
INSERT INTO Payment (leaseID, paymentDate, amount)
VALUES
(1, '2023-01-03', 200.00),
(2, '2023-02-20', 1000.00),
(3, '2023-03-12', 75.00),
(4, '2023-04-25', 900.00),
(5, '2023-05-07', 60.00),
(6, '2023-06-18', 1200.00),
(7, '2023-07-03', 40.00),
(8, '2023-08-14', 1100.00),
(9, '2023-09-09', 80.00),
(10, '2023-10-25', 1500.00);
```

--1. Update the daily rate for a Mercedes car to 68.

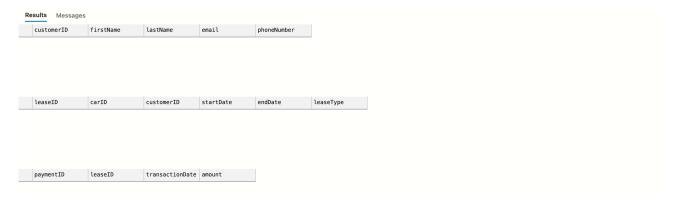
UPDATE Vehicle SET dailyRate=68 WHERE make='Mercedes';
SELECT * FROM Vehicle;

	carID 🗸	make 🗸	model V	year 🗸	dailyRate ∨	status 🗸	passengerCapacity ∨	engineCapacity 🗸
1	1	Toyota	Camry	2022	50.00	available	4	1450
2	2	Honda	Civic	2023	45.00	available	7	1500
3	3	Ford	Focus	2022	48.00	notAvailable	4	1400
4	4	Nissan	Altima	2023	52.00	available	7	1200
5	5	Chevrolet	Malibu	2022	47.00	available	4	1800
6	6	Hyundai	Sonata	2023	49.00	notAvailable	7	1400
7	7	BMW	3 Series	2023	60.00	available	7	2499
8	8	Mercedes	C-Class	2022	68.00	available	8	2599
9	9	Audi	A4	2022	55.00	notAvailable	4	2500
10	10	Lexus	ES	2023	54.00	available	4	2500

--2. Delete a specific customer and all associated leases and payments.

```
DELETE FROM Payment
WHERE leaseID IN (SELECT leaseID FROM Lease WHERE customerID = 3);
DELETE FROM Lease
WHERE customerID = 3
DELETE FROM Customer
WHERE customerID = 3;

SELECT * FROM Customer WHERE customerID = 3;
SELECT * FROM Lease WHERE customerID = 3;
SELECT * FROM Payment WHERE leaseID NOT IN (SELECT leaseID FROM Lease);
```



--3. Rename the "paymentDate" column in the Payment table to "transactionDate".

EXEC sp_rename 'Payment.paymentDate', 'transactionDate', 'COLUMN';
SELECT * FROM Payment;

	paymentID \vee	leaseID 🗸	transactionDate 🗸	amount \vee
1	1	1	2023-01-03	200.00
2	2	2	2023-02-20	1000.00
3	4	4	2023-04-25	900.00
4	5	5	2023-05-07	60.00
5	7	7	2023-07-03	40.00
6	8	8	2023-08-14	1100.00
7	10	10	2023-10-25	1500.00
6	8	8	2023-08-14	1100.00

--4. Find a specific customer by email.

SELECT * FROM Customer WHERE email='johndoe@example.com';



--5. Get active leases for a specific customer.

SELECT * FROM Lease
WHERE customerID = 3 AND endDate >= GETDATE();



--6. Find all payments made by a customer with a specific phone number.



--7. Calculate the average daily rate of all available cars.

SELECT make, AVG(dailyRate) AS [Average Daily Rate]
FROM Vehicle
WHERE status = 'available'
GROUP BY make;



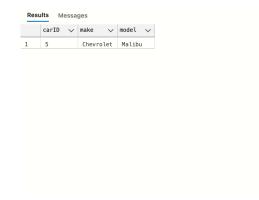
--8. Find the car with the highest daily rate.

SELECT make, model, dailyRate
FROM Vehicle
WHERE dailyRate = (SELECT MAX(dailyRate) FROM Vehicle);



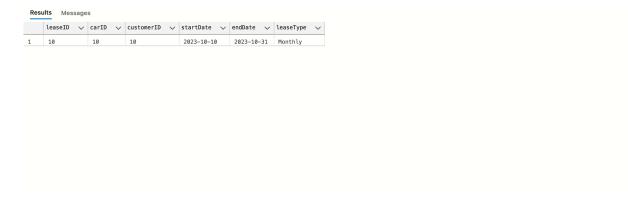
--9. Retrieve all cars leased by a specific customer.

```
SELECT V.carID, V.make, V.model
FROM Vehicle V
JOIN Lease L ON V.carID = L.carID
JOIN Customer C ON C.customerID = L.customerID
WHERE C.customerID = 5;
```



-- 10. Find the details of the most recent lease.

SELECT *
FROM Lease
WHERE startDate = (SELECT MAX(startDate) FROM Lease);



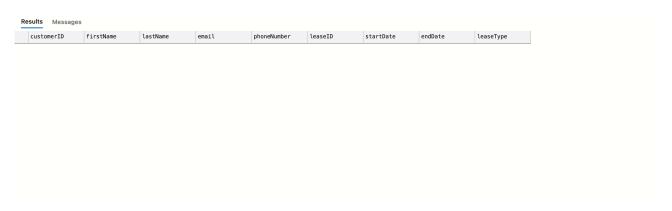
--11. List all payments made in the year 2023.

SELECT * FROM Payment WHERE YEAR(transactionDate)=2023;

Results Messages											
	paymentID	∨ leas	seID ∨	transactionDate	~	amount	~				
1	1	1		2023-01-03		200.00					
2	2	2		2023-02-20		1000.00					
3	4	4		2023-04-25		900.00					
4	5	5		2023-05-07		60.00					
5	7	7		2023-07-03		40.00					
6	8	8		2023-08-14		1100.00					
7	10	10		2023-10-25		1500.00					

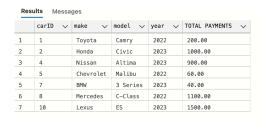
--12. Retrieve customers who have not made any payments.

SELECT C.*, L.leaseID, L.startDate, L.endDate, L.leaseType
FROM Customer C
JOIN Lease L ON C.customerID = L.customerID
LEFT JOIN Payment P ON L.leaseID = P.leaseID
WHERE P.leaseID IS NULL;



--13. Retrieve Car Details and Their Total Payments.

SELECT V.carID, V.make, V.model, V.year, SUM(P.amount) AS [TOTAL
PAYMENTS]
FROM Vehicle V
JOIN Lease L ON V.carID = L.carID
JOIN Payment P ON L.leaseID = P.leaseID
GROUP BY V.carID, V.make, V.model, V.year;



--14. Calculate Total Payments for Each Customer.

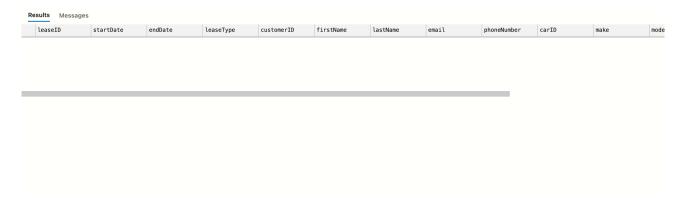
SELECT C.customerID, C.FirstName, C.LastName, C.Email,
SUM(P.amount) AS [TOTAL PAYMENTS]
FROM Customer C
JOIN Lease L ON C.customerID = L.customerID
JOIN Payment P ON L.leaseID = P.leaseID
GROUP BY C.customerID, C.FirstName, C.LastName, C.Email;

	customerID V	FirstName V	LastName V	Email	TOTAL PAYMENTS		
	Customer 25	T 21 5 CTGING V	Eus citamo V	Lind 2 C	TOTAL TATALETTS		
1	1	John	Doe	johndoe@example.com	200.00		
2	2	Jane	Smith	janesmith@example.com	1000.00		
3	4	Sarah	Brown	sarah@example.com	900.00		
4	5	David	Lee	david@example.com	60.00		
5	7	Michael	Davis	michael@example.com	40.00		
6	8	Emma	Wilson	emma@example.com	1100.00		
7	10	Olivia	Adams	olivia@example.com	1500.00		

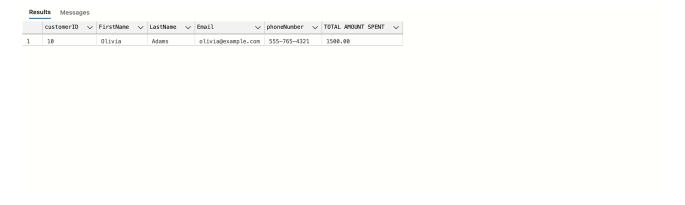
--15. List Car Details for Each Lease.

Res	Results Messages											
	leaseID 🗸	startDate 🗸	endDate 🗸	leaseType 🗸	carID 🗸	make 🗸	model 🗸	year 🗸	dailyRate ∨	status 🗸		
1	1	2023-01-01	2023-01-05	Daily	1	Toyota	Camry	2022	50.00	available		
2	2	2023-02-15	2023-02-28	Monthly	2	Honda	Civic	2023	45.00	available		
3	4	2023-04-20	2023-04-30	Monthly	4	Nissan	Altima	2023	52.00	available		
4	5	2023-05-05	2023-05-10	Daily	5	Chevrolet	Malibu	2022	47.00	available		
5	7	2023-07-01	2023-07-10	Daily	7	BMW	3 Series	2023	60.00	available		
6	8	2023-08-12	2023-08-15	Monthly	8	Mercedes	C-Class	2022	68.00	available		
7	10	2023-10-10	2023-10-31	Monthly	10	Lexus	ES	2023	54.00	available		

--16. Retrieve Details of Active Leases with Customer and Car Information.



--17. Find the Customer Who Has Spent the Most on Leases.



-- 18. List All Cars with Their Current Lease Information.

	carID 🗸	make 🗸	model 🗸	year 🗸	dailyRate ∨	status 🗸	leaseID \vee	startDate 🗸	endDate \vee	leaseType \vee	customerID \vee	firstName \vee	last
1	1	Toyota	Camry	2022	50.00	available	NULL	NULL	NULL	NULL	NULL	NULL	NUL
2	2	Honda	Civic	2023	45.00	available	NULL	NULL	NULL	NULL	NULL	NULL	NUL
3	3	Ford	Focus	2022	48.00	notAvailable	NULL	NULL	NULL	NULL	NULL	NULL	NUL
4	4	Nissan	Altima	2023	52.00	available	NULL	NULL	NULL	NULL	NULL	NULL	NUL
5	5	Chevrolet	Malibu	2022	47.00	available	NULL	NULL	NULL	NULL	NULL	NULL	NUL
6	6	Hyundai	Sonata	2023	49.00	notAvailable	NULL	NULL	NULL	NULL	NULL	NULL	NUL
7	7	BMW	3 Series	2023	60.00	available	NULL	NULL	NULL	NULL	NULL	NULL	NUL
8	8	Mercedes	C-Class	2022	68.00	available	NULL	NULL	NULL	NULL	NULL	NULL	NUL
9	9	Audi	A4	2022	55.00	notAvailable	NULL	NULL	NULL	NULL	NULL	NULL	NUL
10	10	Lexus	ES	2023	54.00	available	NULL	NULL	NULL	NULL	NULL	NULL	NUL