

SQL server code challenge

-- Create database

```
CREATE DATABASE Car_RentalSystem  
  
USE Car_RentalSystem
```

-- Vehicle Table

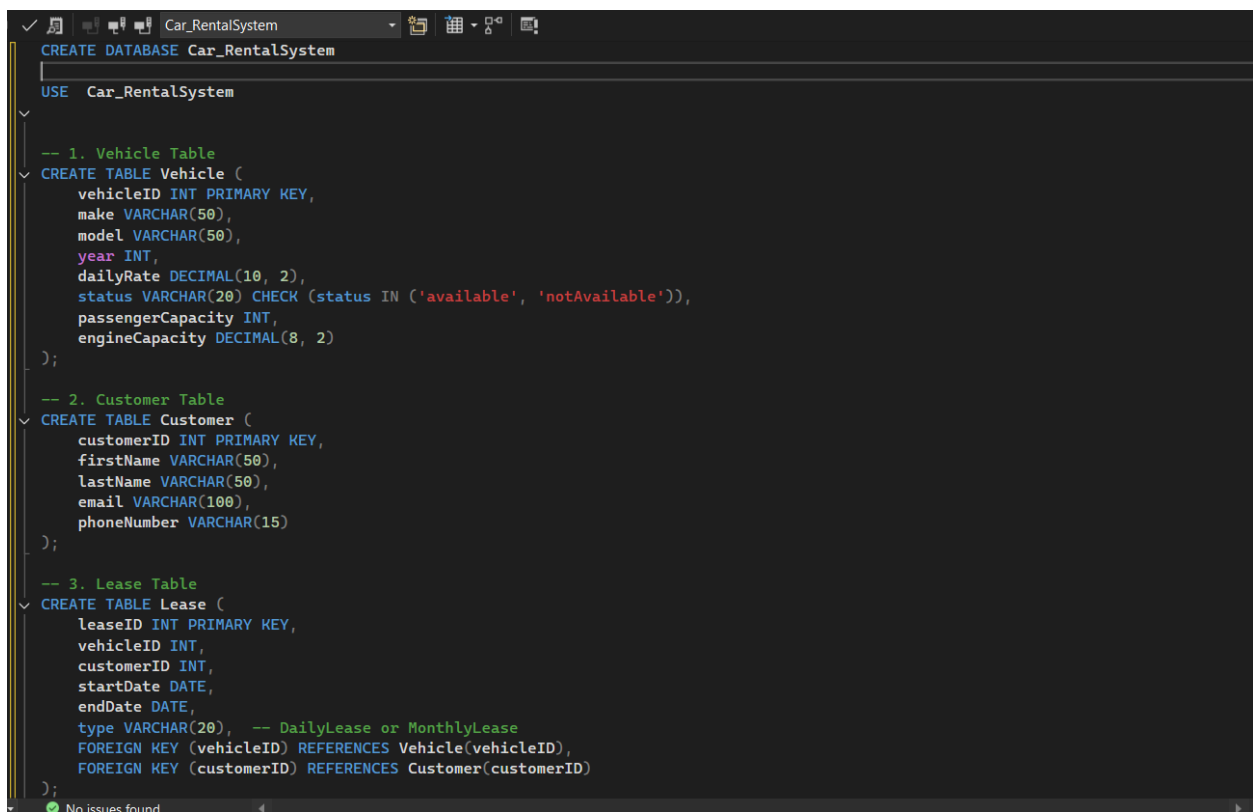
```
CREATE TABLE Vehicle (  
    vehicleID INT PRIMARY KEY,  
    make VARCHAR(50),  
    model VARCHAR(50),  
    year INT,  
    dailyRate DECIMAL(10, 2),  
    status VARCHAR(20) CHECK (status IN ('available', 'notAvailable')),  
    passengerCapacity INT,  
    engineCapacity DECIMAL(8, 2)  
);
```

-- Customer Table

```
CREATE TABLE Customer (  
    customerID INT PRIMARY KEY,  
    firstName VARCHAR(50),  
    lastName VARCHAR(50),  
    email VARCHAR(100),  
    phoneNumber VARCHAR(15)  
);
```

-- Lease Table

```
CREATE TABLE Lease (  
    leaseID INT PRIMARY KEY,  
    vehicleID INT,  
    customerID INT,  
    startDate DATE,  
    endDate DATE,  
    type VARCHAR(20), -- DailyLease or MonthlyLease  
    FOREIGN KEY (vehicleID) REFERENCES Vehicle(vehicleID),  
    FOREIGN KEY (customerID) REFERENCES Customer(customerID)  
);
```



The screenshot shows a SQL IDE window titled 'Car_RentalSystem'. The main editor area contains the following SQL code:

```
CREATE DATABASE Car_RentalSystem  
  
USE Car_RentalSystem  
  
-- 1. Vehicle Table  
CREATE TABLE Vehicle (  
    vehicleID INT PRIMARY KEY,  
    make VARCHAR(50),  
    model VARCHAR(50),  
    year INT,  
    dailyRate DECIMAL(10, 2),  
    status VARCHAR(20) CHECK (status IN ('available', 'notAvailable')),  
    passengerCapacity INT,  
    engineCapacity DECIMAL(8, 2)  
);  
  
-- 2. Customer Table  
CREATE TABLE Customer (  
    customerID INT PRIMARY KEY,  
    firstName VARCHAR(50),  
    lastName VARCHAR(50),  
    email VARCHAR(100),  
    phoneNumber VARCHAR(15)  
);  
  
-- 3. Lease Table  
CREATE TABLE Lease (  
    leaseID INT PRIMARY KEY,  
    vehicleID INT,  
    customerID INT,  
    startDate DATE,  
    endDate DATE,  
    type VARCHAR(20), -- DailyLease or MonthlyLease  
    FOREIGN KEY (vehicleID) REFERENCES Vehicle(vehicleID),  
    FOREIGN KEY (customerID) REFERENCES Customer(customerID)  
);
```

The status bar at the bottom indicates 'No issues found'.

-- Payment Table

```
CREATE TABLE Payment (  
    paymentID INT PRIMARY KEY,  
    leaseID INT,  
    amount DECIMAL(10, 2),  
    paymentDate DATE,  
    FOREIGN KEY (leaseID) REFERENCES Lease(leaseID)
```

```

paymentID INT PRIMARY KEY,

leaseID INT,

paymentDate DATE,

amount DECIMAL(10, 2),

FOREIGN KEY (leaseID) REFERENCES Lease(leaseID)

);

```

SELECT * FROM Vehicle

SELECT * FROM Customer

SELECT * FROM Lease

SELECT * FROM Payment

The screenshot displays a SQL IDE with the following content:

SQL Script (Left Pane):

```

-- 4. Payment Table
CREATE TABLE Payment (
    paymentID INT PRIMARY KEY,
    leaseID INT,
    paymentDate DATE,
    amount DECIMAL(10, 2),
    FOREIGN KEY (leaseID) REFERENCES Lease(leaseID)
);

SELECT * FROM Vehicle;

SELECT * FROM Customer;

SELECT * FROM Lease;

SELECT * FROM Payment;

-- Insert into Vehicle Table
INSERT INTO Vehicle (vehicleID, make, model, year, dailyRate, status, passengerCapacity, engineCapacity) VALUES
(1, 'Toyota', 'Camry', 2022, 50.00, 'available', 4, 1450),
(2, 'Honda', 'Civic', 2023, 45.00, 'available', 7, 1500),
(3, 'Ford', 'Focus', 2022, 43.00, '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);

-- Insert into Customer Table
INSERT INTO Customer (customerID, firstName, lastName, email, phoneNumber) VALUES
(1, 'John', 'Doe', 'john.doe@example.com', '555-555-5555'),
(2, 'Jane', 'Smith', 'jane.smith@example.com', '555-123-4567'),
(3, 'Sarah', 'Brown', 'sarah.brown@example.com', '555-456-7890'),
(4, 'David', 'Lee', 'david.lee@example.com', '555-987-6543'),
(5, 'Laura', 'Hall', 'laura.hall@example.com', '555-234-5678'),
(6, 'Michael', 'Davis', 'michael.davis@example.com', '555-876-5432'),
(7, 'Emma', 'Wilson', 'emma.wilson@example.com', '555-432-1098'),
(8, 'William', 'Taylor', 'william.taylor@example.com', '555-321-0987'),
(9, 'Olivia', 'Adams', 'olivia.adams@example.com', '555-765-4321'),
(10, 'Robert', 'Johnson', 'robert.johnson@example.com', '555-789-1234');

```

Data Tables (Right Pane):

vehicleID	make	model	year	dailyRate	status	passengerCapacity	engineCapacity
1	Toyota	Camry	2022	50.00	available	4	1450
2	Honda	Civic	2023	45.00	available	7	1500
3	Ford	Focus	2022	43.00	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

customerID	firstName	lastName	email	phoneNumber
1	John	Doe	john.doe@example.com	555-555-5555
2	Jane	Smith	jane.smith@example.com	555-123-4567
3	Sarah	Brown	sarah.brown@example.com	555-456-7890
4	David	Lee	david.lee@example.com	555-987-6543
5	Laura	Hall	laura.hall@example.com	555-234-5678
6	Michael	Davis	michael.davis@example.com	555-876-5432
7	Emma	Wilson	emma.wilson@example.com	555-432-1098
8	William	Taylor	william.taylor@example.com	555-321-0987
9	Olivia	Adams	olivia.adams@example.com	555-765-4321
10	Robert	Johnson	robert.johnson@example.com	555-789-1234

leaseID	vehicleID	customerID	startDate	endDate	type
1	1	1	2023-01-01	2023-01-05	Daily
2	2	2	2023-02-15	2023-02-20	Monthly
3	4	4	2023-04-20	2023-04-30	Monthly
4	5	5	2023-05-05	2023-05-10	Daily
5	7	7	2023-07-01	2023-07-10	Daily
6	8	8	2023-06-12	2023-06-15	Monthly
7	10	10	2023-10-10	2023-10-31	Monthly

paymentID	leaseID	transactionDate	amount
1	1	2023-01-03	200.00
2	2	2023-02-20	1000.00
3	4	2023-04-25	900.00
4	5	2023-05-07	60.00
5	7	2023-07-03	40.00
6	8	2023-06-14	1100.00
7	10	2023-10-25	1500.00

-- Insert into Vehicle Table

```

INSERT INTO Vehicle (vehicleID, make, model, year, dailyRate, status, passengerCapacity, engineCapacity)
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);

-- Insert into Customer Table

INSERT INTO Customer (customerID, firstName, lastName, email, phoneNumber) 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');

-- Insert into Lease Table

INSERT INTO Lease (leaseID, vehicleID, customerID, startDate, endDate, type) VALUES

(1, 1, 1, '2023-01-01', '2023-01-05', 'Daily'),
(2, 2, 2, '2023-02-15', '2023-02-28', 'Monthly'),
(3, 3, 3, '2023-03-10', '2023-03-15', 'Daily'),

```
(4, 4, 4, '2023-04-20', '2023-04-30', 'Monthly'),  
(5, 5, 5, '2023-05-05', '2023-05-10', 'Daily'),  
(6, 4, 3, '2023-06-15', '2023-06-30', 'Monthly'),  
(7, 7, 7, '2023-07-01', '2023-07-10', 'Daily'),  
(8, 8, 8, '2023-08-12', '2023-08-15', 'Monthly'),  
(9, 3, 3, '2023-09-07', '2023-09-10', 'Daily'),  
(10, 10, 10, '2023-10-10', '2023-10-31', 'Monthly');
```

--Insert into Payment Table

```
INSERT INTO Payment (paymentID, leaseID, paymentDate, amount) 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);
```

```
Car_RentalSystem
INSERT INTO Customer (customerID, firstName, lastName, email, phoneNumber) 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');

-- Insert into Lease Table
INSERT INTO Lease (leaseID, vehicleID, customerID, startDate, endDate, type) VALUES
(1, 1, 1, '2023-01-01', '2023-01-05', 'Daily'),
(2, 2, 2, '2023-02-15', '2023-02-28', 'Monthly'),
(3, 3, 3, '2023-03-10', '2023-03-15', 'Daily'),
(4, 4, 4, '2023-04-20', '2023-04-30', 'Monthly'),
(5, 5, 5, '2023-05-05', '2023-05-10', 'Daily'),
(6, 4, 3, '2023-06-15', '2023-06-30', 'Monthly'),
(7, 7, 7, '2023-07-01', '2023-07-10', 'Daily'),
(8, 8, 8, '2023-08-12', '2023-08-15', 'Monthly'),
(9, 3, 3, '2023-09-07', '2023-09-10', 'Daily'),
(10, 10, 10, '2023-10-10', '2023-10-31', 'Monthly');

--Insert into Payment Table
INSERT INTO Payment (paymentID, leaseID, paymentDate, amount) 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: Update the daily rate for a Mercedes car to 68

UPDATE Vehicle

SET dailyRate = 68

WHERE make = 'Mercedes';

-- 2: Delete a specific customer (e.g., customerID = 3) 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;

```

(10, 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

-- 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 Payment

SELECT * FROM Lease

SELECT * FROM Customer

-- 3: Rename the "paymentDate" column in the Payment table to "transactionDate".
EXEC sp_rename 'Payment.paymentDate', 'transactionDate', 'COLUMN';

-- 4: Find a specific customer by email
SELECT * FROM Customer
WHERE email = 'janesmith@example.com';

-- 5: Get active leases for a specific customer
SELECT * FROM Lease
WHERE customerID = 1 AND endDate >= GETDATE();

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

```

vehicleID	make	model	year	dailyRate	status	passengerCapacity	engineCapacity
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	68.00	available	3	2599
9	Audi	A4	2022	55.00	notAvailable	4	2500

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

leaseID	vehicleID	customerID	startDate	endDate	type
1	1	1	2023-01-01	2023-01-05	Daily
2	2	2	2023-02-15	2023-02-28	Monthly
3	4	4	2023-04-20	2023-04-30	Monthly
4	5	5	2023-05-05	2023-05-10	Daily
5	7	7	2023-07-01	2023-07-10	Daily
6	8	8	2023-08-12	2023-08-15	Monthly
7	10	10	2023-10-10	2023-10-31	Monthly

customerID	firstName	lastName	email	phoneNumber
1	John	Doe	johndoe@example.com	555-555-5555
2	Jane	Smith	janesmith@example.com	555-123-4567
3	Sarah	Brown	sarah@example.com	555-456-7890
4	David	Lee	david@example.com	555-987-6543
5	Laura	Hall	laura@example.com	555-234-5678
6	Michael	Davis	michael@example.com	555-876-5432
7	Emma	Wilson	emma@example.com	555-432-1098
8	William	Taylor	william@example.com	555-321-0987
9	Olivia	Adams	olivia@example.com	555-765-4321

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

EXEC sp_rename 'Payment.paymentDate', 'transactionDate', 'COLUMN';

-- 4: Find a specific customer by email

SELECT * FROM Customer

WHERE email = 'janesmith@example.com';

-- 5: Get active leases for a specific customer

SELECT * FROM Lease

WHERE customerID = 1 AND endDate >= GETDATE();

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

SELECT p.* FROM Payment p

JOIN Lease l ON p.leaseID = l.leaseID

JOIN Customer c ON l.customerID = c.customerID

WHERE c.phoneNumber = '555-876-5432';

The screenshot shows a SQL IDE with a script containing 7 queries. The results are displayed in a table viewer on the right. The queries and their results are as follows:

```

WHERE customerID = 3;
SELECT * FROM Payment

SELECT * FROM Lease

SELECT * FROM Customer

--3: Rename the "paymentDate" column in the Payment table to "transactionDate".
EXEC sp_rename 'Payment.paymentDate', 'transactionDate', 'COLUMN';

-- 4: Find a specific customer by email
SELECT * FROM Customer
WHERE email = 'janesmith@example.com';

-- 5: Get active leases for a specific customer
SELECT * FROM Lease
WHERE customerID = 1 AND endDate >= GETDATE();

-- 6: Find all payments made by a customer with a specific phone number
SELECT p.* FROM Payment p
JOIN Lease l ON p.leaseID = l.leaseID
JOIN Customer c ON l.customerID = c.customerID
WHERE c.phoneNumber = '555-876-5432';

-- 7: Calculate the average daily rate of all available cars
SELECT AVG(dailyRate) AS AvgDailyRate
FROM Vehicle
WHERE status = 'available';

```

The table viewer shows the following results:

customerID	firstName	lastName	email	phoneNumber	
1	2	Jane	Smith	janesmith@example.com	555-123-4567

leaseID	vehicleID	customerID	startDate	endDate	type
1	7	7	2023-07-03	40.00	

paymentID	leaseID	transactiondate	amount	
1	7	7	2023-07-03	40.00

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

```

SELECT AVG(dailyRate) AS AvgDailyRate

FROM Vehicle

WHERE status = 'available';

```

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

```

SELECT TOP 1 *

FROM Vehicle

ORDER BY dailyRate DESC;

```

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

```

SELECT v.*

FROM Vehicle v

JOIN Lease l ON v.vehicleID = l.vehicleID

WHERE l.customerID = 2;

```


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

```
SELECT TOP 1 *  
  
FROM Lease  
  
ORDER BY startDate DESC;
```

The screenshot shows a SQL IDE with a query editor on the left and a results pane on the right. The query editor contains the following SQL code:

```
JOIN Customer c ON l.customerID = c.customerID  
WHERE c.phoneNumber = '555-876-5432';  
  
-- 7: Calculate the average daily rate of all available cars  
SELECT AVG(dailyRate) AS AvgDailyRate  
FROM Vehicle  
WHERE status = 'available';  
  
-- 8: Find the car with the highest daily rate  
SELECT TOP 1 *  
FROM Vehicle  
ORDER BY dailyRate DESC;  
  
-- 9: Retrieve all cars leased by a specific customer  
SELECT v.*  
FROM Vehicle v  
JOIN Lease l ON v.vehicleID = l.vehicleID  
WHERE l.customerID = 2;  
  
-- 10: Find the details of the most recent lease  
SELECT TOP 1 *  
FROM Lease  
ORDER BY startDate DESC;  
  
-- 11: List all payments made in the year 2023  
SELECT *  
FROM Payment  
WHERE YEAR(transactionDate) = 2023;  
  
-- 12: Retrieve customers who have not made any payments  
SELECT *  
FROM Customer  
WHERE customerID NOT IN (
```

The results pane on the right displays the results of the query. It shows a table with the following data:

vehicleID	make	model	year	dailyRate	status	passengerCapacity	engineCapacity	
1	8	Mercedes	C-Class	2022	68.00	available	8	2599

Below this table, there is another table with the following data:

vehicleID	make	model	year	dailyRate	status	passengerCapacity	engineCapacity	
1	2	Honda	Civic	2023	45.00	available	7	1500

At the bottom of the results pane, there is a table with the following data:

leaseID	vehicleID	customerID	startDate	endDate	type	
1	10	10	10	2023-10-10	2023-10-31	Monthly

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

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

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

```
SELECT *  
  
FROM Customer  
  
WHERE customerID NOT IN (  
  
    SELECT DISTINCT l.customerID  
  
    FROM Lease l  
  
    JOIN Payment p ON l.leaseID = p.leaseID  
  
);
```

-- 13: Retrieve Car Details and Their Total Payments

```
SELECT v.vehicleID, v.make, v.model, SUM(p.amount) AS TotalPayments
FROM Vehicle v
JOIN Lease l ON v.vehicleID = l.vehicleID
JOIN Payment p ON l.leaseID = p.leaseID
GROUP BY v.vehicleID, v.make, v.model;
```

-- 14: Calculate Total Payments for Each Customer

```
SELECT c.customerID, c.firstName, c.lastName, SUM(p.amount) AS TotalPaid
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;
```

The screenshot displays a SQL IDE interface with a query editor on the left and a results pane on the right. The query editor contains five SQL queries, each preceded by a comment. The results pane shows the output of these queries as tables.

Query 1: `FROM Lease ORDER BY startDate DESC;`

Query 2: `-- 11: List all payments made in the year 2023`
`SELECT *`
`FROM Payment`
`WHERE YEAR(transactionDate) = 2023;`

Query 3: `-- 12: Retrieve customers who have not made any payments`
`SELECT *`
`FROM Customer`
`WHERE customerID NOT IN (`
`SELECT DISTINCT l.customerID`
`FROM Lease l`
`JOIN Payment p ON l.leaseID = p.leaseID`
`);`

Query 4: `-- 13: Retrieve Car Details and Their Total Payments`
`SELECT v.vehicleID, v.make, v.model, SUM(p.amount) AS TotalPayments`
`FROM Vehicle v`
`JOIN Lease l ON v.vehicleID = l.vehicleID`
`JOIN Payment p ON l.leaseID = p.leaseID`
`GROUP BY v.vehicleID, v.make, v.model;`

Query 5: `-- 14: Calculate Total Payments for Each Customer`
`SELECT c.customerID, c.firstName, c.lastName, SUM(p.amount) AS TotalPaid`
`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;`

Query 6: `-- 15: List Car Details for Each Lease`
`SELECT l.leaseID, v.make, v.model, v.year, l.startDate, l.endDate`
`FROM Lease l`
`JOIN Vehicle v ON l.vehicleID = v.vehicleID;`

Results:

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

customerID	firstName	lastName	email	phoneNumber
6	Laura	Hall	laura@example.com	555-234-5678
9	William	Taylor	william@example.com	555-321-6547

vehicleID	make	model	TotalPayments
1	Toyota	Camry	200.00
2	Honda	Civic	1000.00
3	Nissan	Altima	900.00
4	Chevrolet	Malibu	60.00
5	BMW	3 Series	40.00
6	Mercedes	C-Class	1100.00
7	Lexus	ES	1500.00

customerID	firstName	lastName	TotalPaid
1	John	Doe	200.00
2	Jane	Smith	1000.00
3	Sarah	Brown	900.00
4	David	Lee	60.00
5	Michael	Davis	40.00
6	Emma	Wilson	1100.00
7	Olivia	Adams	1500.00

-- 15: List Car Details for Each Lease

```
SELECT l.leaseID, v.make, v.model, v.year, l.startDate, l.endDate
FROM Lease l
JOIN Vehicle v ON l.vehicleID = v.vehicleID;
```

```
JOIN Vehicle v ON l.vehicleID = v.vehicleID;
```

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

```
SELECT l.leaseID, c.firstName, c.lastName, v.make, v.model, l.startDate, l.endDate  
FROM Lease l  
JOIN Customer c ON l.customerID = c.customerID  
JOIN Vehicle v ON l.vehicleID = v.vehicleID  
WHERE l.endDate >= GETDATE();
```

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

```
SELECT TOP 1 c.customerID, c.firstName, c.lastName, SUM(p.amount) AS TotalSpent  
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  
ORDER BY TotalSpent DESC;
```

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

```
SELECT v.vehicleID, v.make, v.model, l.leaseID, l.startDate, l.endDate  
FROM Vehicle v  
LEFT JOIN Lease l ON v.vehicleID = l.vehicleID AND l.endDate >= GETDATE();
```

-- 14: Calculate Total Payments for Each Customer
SELECT c.customerID, c.firstName, c.lastName, SUM(p.amount) AS TotalPaid
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;

-- 15: List Car Details for Each Lease
SELECT l.leaseID, v.make, v.model, v.year, l.startDate, l.endDate
FROM Lease l
JOIN Vehicle v ON l.vehicleID = v.vehicleID;

-- 16: Retrieve Details of Active Leases with Customer and Car Information
SELECT l.leaseID, c.firstName, c.lastName, v.make, v.model, l.startDate, l.endDate
FROM Lease l
JOIN Customer c ON l.customerID = c.customerID
JOIN Vehicle v ON l.vehicleID = v.vehicleID
WHERE l.endDate >= GETDATE();

-- 17: Find the Customer Who Has Spent the Most on Leases
SELECT TOP 1 c.customerID, c.firstName, c.lastName, SUM(p.amount) AS TotalSpent
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
ORDER BY TotalSpent DESC;

-- 18: List All Cars with Their Current Lease Information
SELECT v.vehicleID, v.make, v.model, l.leaseID, l.startDate, l.endDate
FROM Vehicle v
LEFT JOIN Lease l ON v.vehicleID = l.vehicleID AND l.endDate >= GETDATE();

Ln: 222 Chr: 1 SPC CRLF

No issues found

leaseID	make	model	year	startDate	endDate
1	Toyota	Camry	2022	2023-01-01	2023-01-05
2	Honda	Civic	2023	2023-02-15	2023-02-28
3	Nissan	Altima	2023	2023-04-20	2023-04-30
4	Chevrolet	Malibu	2022	2023-05-05	2023-05-10
5	BMW	3 Series	2023	2023-07-01	2023-07-10
6	Mercedes	C-Class	2022	2023-08-12	2023-08-15
7	Lexus	ES	2023	2023-10-10	2023-10-31

leaseID	firstName	lastName	make	model	startDate	endDate
---------	-----------	----------	------	-------	-----------	---------

customerID	firstName	lastName	TotalSpent
1	Olivia	Adams	1500.00

vehicleID	make	model	leaseID	startDate	endDate
1	Toyota	Camry	NULL	NULL	NULL
2	Honda	Civic	NULL	NULL	NULL
3	Ford	Focus	NULL	NULL	NULL
4	Nissan	Altima	NULL	NULL	NULL
5	Chevrolet	Malibu	NULL	NULL	NULL
6	Hyundai	Sonata	NULL	NULL	NULL
7	BMW	3 Series	NULL	NULL	NULL
8	Mercedes	C-Class	NULL	NULL	NULL
9	Audi	A4	NULL	NULL	NULL
10	Lexus	ES	NULL	NULL	NULL