Coding Challenge - Car Rental System – SQL

```sql
CREATE TABLE Vehicle ( vehicleID INT PRIMARY KEY,
    make VARCHAR(255),
    model VARCHAR(255),
    year INT,
    dailyRate DECIMAL(10, 2),
    status VARCHAR(20),
    passengerCapacity INT,
    engineCapacity INT );
CREATE TABLE Customer ( customerID INT PRIMARY KEY,
    firstName VARCHAR(255),
    lastName VARCHAR(255),
    email VARCHAR(255),
    phoneNumber VARCHAR(15) );
CREATE TABLE Lease ( leaseID INT PRIMARY KEY,
    vehicleID INT,
    customerID INT,
    startDate DATE,
    endDate DATE,
    type VARCHAR(20),
    FOREIGN KEY (vehicleID) REFERENCES Vehicle(vehicleID),
    FOREIGN KEY (customerID) REFERENCES Customer(customerID) );
```

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

```

✓	98	14:58:05	INSERT INTO Vehicle VALUES (1, 'Toyota', 'Camry', 2022, 50.00, 'available', 4, 1450), (2, 'Honda', 'Civic', 202...
✓	99	14:58:05	INSERT INTO Customer VALUES (1, 'John', 'Doe', 'johndoe@example.com', '555-555-5555'), (2, 'Jane', 'Smith'...
✓	100	14:58:05	INSERT INTO Lease VALUES (1, 1, 1, '2023-01-01', '2023-01-05', 'Daily'), (2, 2, 2, '2023-02-15', '2023-02-28', '...
✓	101	14:58:05	INSERT INTO Payment VALUES (1, 1, '2023-01-03', 200.00), (2, 2, '2023-02-20', 1000.00), (3, 3, '2023-03-12',...

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

```
1  UPDATE Vehicle
2  SET dailyRate = 68
3  WHERE make = 'Mercedes';
4
5  SELECT * FROM Vehicle;
6
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Ce

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	8	2599
9	Audi	A4	2022	55.00	notAvailable	4	2500
10	Lexus	ES	2023	54.00	available	4	2500

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

```
1  DELETE FROM Payment
2  WHERE leaseID IN (SELECT leaseID FROM Lease WHERE customerID = 1);
3  DELETE FROM Lease
4  WHERE customerID = 1;
5  DELETE FROM Customer
6  WHERE customerID = 1;
```

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

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

```sql
ALTER TABLE Payment
RENAME COLUMN paymentDate TO transactionDate;
DESC Payment;
```

Result Grid | Filter Rows: | Export: | Wrap Cell

Field	Type	Null	Key	Default	Extra
paymentID	int	NO	PRI	NULL	
leaseID	int	YES	MUL	NULL	
transactionDate	date	YES		NULL	
amount	decimal(10,2)	YES		NULL	

4. Find a specific customer by email.

```sql
SELECT * FROM Customer WHERE email = "robert@example.com";
```

Result Grid | Filter Rows: | Edit: | Export/Import:

customerID	firstName	lastName	email	phoneNumber
3	Robert	Johnson	robert@example.com	555-789-1234

5. Get active leases for a specific customer.

```sql
SELECT * FROM Customer JOIN
Lease ON Customer.customerID = Lease.customerID
WHERE CURDATE() BETWEEN Lease.startDate AND Lease.endDate;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

customerID	firstName	lastName	email	phoneNumber	leaseID	vehicleID	customerID	startDate	endDate	type

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

```sql
SELECT Customer.customerID, Payment.* FROM Customer JOIN
Lease ON Customer.customerID = Lease.customerID
JOIN Payment ON Payment.LeaseID = Lease.LeaseID
WHERE Customer.phoneNumber = '555-123-4567';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

customerID	paymentID	leaseID	transactionDate	amount
2	2	2	2023-02-20	1000.00

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

```sql
SELECT AVG(dailyRate) AS 'Average_Daily_Rate' FROM Vehicle
WHERE status = 'available';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Average_Daily_Rate
53.714286

8. Find the car with the highest daily rate.

```sql
SELECT *  FROM Vehicle ORDER BY dailyRate DESC LIMIT 1;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap

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

9. Retrieve all cars leased by a specific customer.

```sql
SELECT c.customerID, v.vehicleID, v.make FROM Customer c
JOIN Lease l ON c.customerID = l.customerID
JOIN Vehicle v ON v.vehicleID = l.vehicleID
WHERE c.customerID = 2;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

customerID	vehicleID	make
2	2	Honda

10. Find the details of the most recent lease.

```sql
SELECT * FROM Lease ORDER BY startdate DESC LIMIT 1;
```

Result Grid | Filter Rows: | Edit: | Expo

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

11. List all payments made in the year 2023.

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

Result Grid | Filter Rows: | Edit: | Export/Import

paymentID	leaseID	transactionDate	amount
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

12. Retrieve customers who have not made any payments.

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

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell C

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

13. Retrieve Car Details and Their Total Payments.

```sql
SELECT v.vehicleID, COALESCE(SUM(p.amount), 0) AS 'Total Payment' FROM Vehicle v
LEFT JOIN Lease l ON v.vehicleID = l.vehicleID
LEFT JOIN Payment p ON p.leaseID = l.leaseID
GROUP BY v.vehicleID;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

vehicleID	Total Payment
1	200.00
2	1000.00
3	155.00
4	2100.00
5	60.00
6	0.00
7	40.00
8	1100.00
9	0.00
10	1500.00

14. Calculate Total Payments for Each Customer.

```sql
SELECT c.customerID, COALESCE(SUM(amount), 0) AS "Total_Payments" FROM Customer c
LEFT JOIN Lease l ON c.customerID = l.customerID
LEFT JOIN Payment p ON p.leaseID = l.leaseID
GROUP BY c.customerID;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

customerID	Total_Payments
1	200.00
2	1000.00
3	1355.00
4	900.00
5	60.00
6	0.00
7	40.00
8	1100.00
9	0.00
10	1500.00

15. List Car Details for Each Lease.

```sql
SELECT v.*, l.leaseID FROM Vehicle v
JOIN Lease l ON v.vehicleID = l.vehicleID;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

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

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

```sql
SELECT Customer.*, Vehicle.* FROM Customer JOIN
Lease ON Customer.customerID = Lease.customerID
JOIN Vehicle ON Vehicle.vehicleID = Lease.vehicleID
WHERE CURDATE() BETWEEN startDate AND endDate;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

customerID	firstName	lastName	email	phoneNumber	vehicleID	make	model	year	dailyRate	status	passengerCapacity	engineCapacity

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

```sql
SELECT c.customerID, c.firstName, c.lastName, c.email, c.phoneNumber,
    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
ORDER BY totalSpent DESC LIMIT 1;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

customerID	firstName	lastName	email	phoneNumber	totalSpent
10	Olivia	Adams	olivia@example.com	555-765-4321	1500.00

18. List All cars with their current lease information.

```sql
SELECT * FROM Lease
JOIN Vehicle ON Vehicle.vehicleID = Lease.vehicleID
WHERE CURDATE() BETWEEN startDate AND endDate;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

leaseID	vehicleID	customerID	startDate	endDate	type	vehicleID	make	model	year	dailyRate	status	passengerCapacity	engineCapacity