Coding Challenge - Car Rental System – SQL

By SHREYASI REJA

Create Database

CREATE DATABASE CarRentalSystem;

USE CarRentalSystem;

```
mysql> CREATE DATABASE  CarRentalSystem;
Query OK, 1 row affected (0.05 sec)

mysql> USE CarRentalSystem;
Database changed
```

SQL Schema:

CREATE TABLE

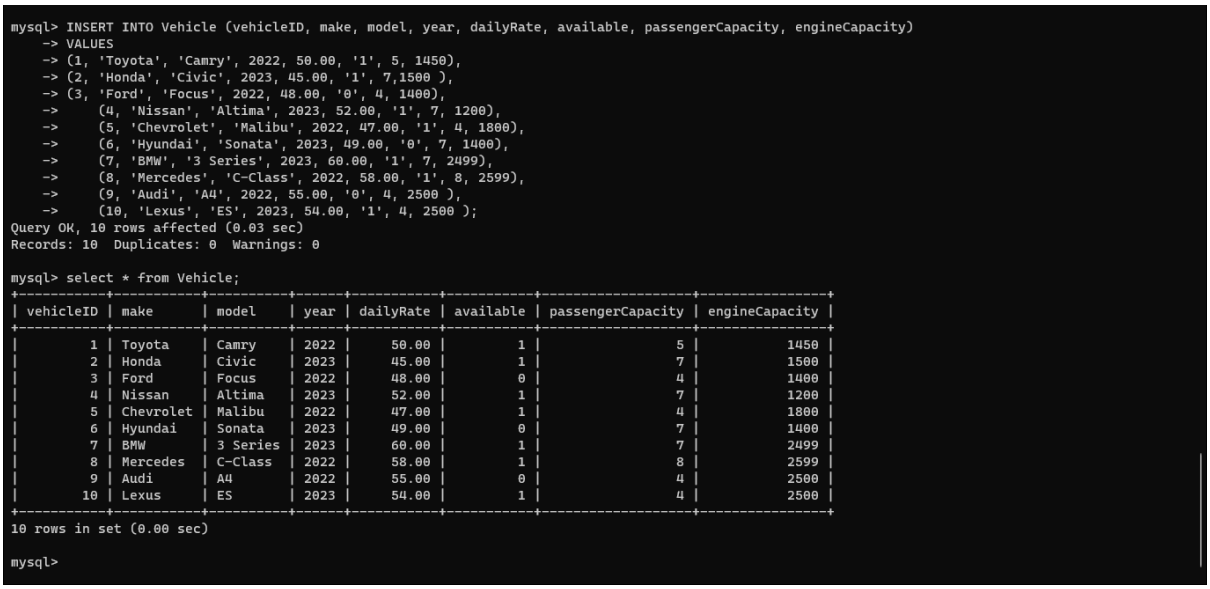
1. Vehicle Table:

```
CREATE TABLE Vehicle (
    ->     vehicleID INT PRIMARY KEY,
    ->     make VARCHAR(50),
    ->     model VARCHAR(50),
    ->     year INT,
    ->     dailyRate DECIMAL(10, 2),
    ->  available BOOLEAN,
    ->     passengerCapacity INT,
    -> engineCapacity INT
    -> );
```

```
mysql> CREATE TABLE Vehicle (
    ->     vehicleID INT PRIMARY KEY,
    ->     make VARCHAR(50),
    ->     model VARCHAR(50),
    ->     year INT,
    ->     dailyRate DECIMAL(10, 2),
    ->  available BOOLEAN,
    ->     passengerCapacity INT,
    -> engineCapacity INT
    -> );
Query OK, 0 rows affected (0.09 sec)

mysql> DESC Vehicle;
+-------------------+---------------+------+-----+---------+-------+
| Field             | Type          | Null | Key | Default | Extra |
+-------------------+---------------+------+-----+---------+-------+
| vehicleID         | int           | NO   | PRI | NULL    |       |
| make              | varchar(50)   | YES  |     | NULL    |       |
| model             | varchar(50)   | YES  |     | NULL    |       |
| year              | int           | YES  |     | NULL    |       |
| dailyRate         | decimal(10,2) | YES  |     | NULL    |       |
| available         | tinyint(1)    | YES  |     | NULL    |       |
| passengerCapacity | int           | YES  |     | NULL    |       |
| engineCapacity    | int           | YES  |     | NULL    |       |
+-------------------+---------------+------+-----+---------+-------+
8 rows in set (0.01 sec)

mysql>
```

DATA TABLE:-

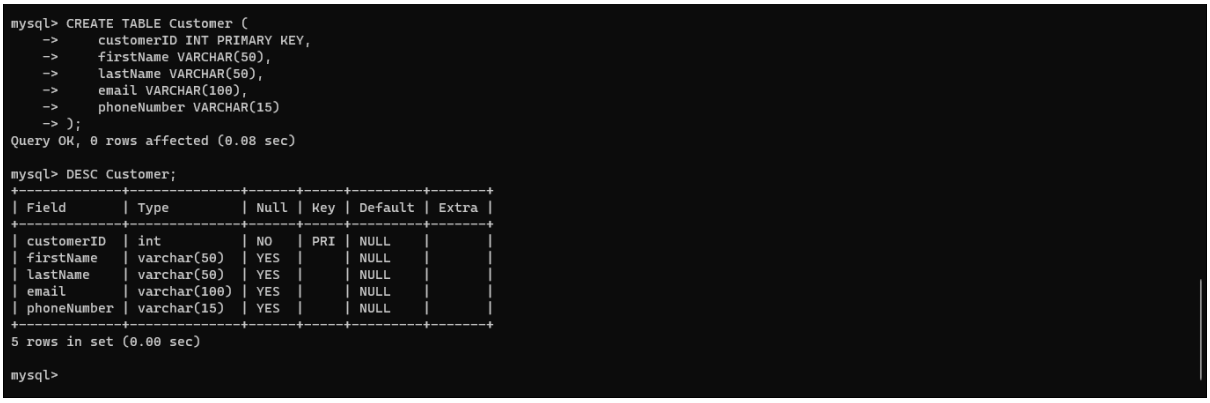
```
INSERT INTO Vehicle (vehicleID, make, model, year, dailyRate,
available, passengerCapacity, engineCapacity)
    -> VALUES
    -> (1, 'Toyota', 'Camry', 2022, 50.00, '1', 5, 1450),
    -> (2, 'Honda', 'Civic', 2023, 45.00, '1', 7,1500 ),
    -> (3, 'Ford', 'Focus', 2022, 48.00, '0', 4, 1400),
    ->     (4, 'Nissan', 'Altima', 2023, 52.00, '1', 7, 1200),
    ->     (5, 'Chevrolet', 'Malibu', 2022, 47.00, '1', 4, 1800),
    ->     (6, 'Hyundai', 'Sonata', 2023, 49.00, '0', 7, 1400),
    ->     (7, 'BMW', '3 Series', 2023, 60.00, '1', 7, 2499),
    ->     (8, 'Mercedes', 'C-Class', 2022, 58.00, '1', 8, 2599),
    ->     (9, 'Audi', 'A4', 2022, 55.00, '0', 4, 2500 ),
    ->     (10, 'Lexus', 'ES', 2023, 54.00, '1', 4, 2500 );
```



```
mysql> INSERT INTO Vehicle (vehicleID, make, model, year, dailyRate, available, passengerCapacity, engineCapacity)
    -> VALUES
    -> (1, 'Toyota', 'Camry', 2022, 50.00, '1', 5, 1450),
    -> (2, 'Honda', 'Civic', 2023, 45.00, '1', 7,1500 ),
    -> (3, 'Ford', 'Focus', 2022, 48.00, '0', 4, 1400),
    ->     (4, 'Nissan', 'Altima', 2023, 52.00, '1', 7, 1200),
    ->     (5, 'Chevrolet', 'Malibu', 2022, 47.00, '1', 4, 1800),
    ->     (6, 'Hyundai', 'Sonata', 2023, 49.00, '0', 7, 1400),
    ->     (7, 'BMW', '3 Series', 2023, 60.00, '1', 7, 2499),
    ->     (8, 'Mercedes', 'C-Class', 2022, 58.00, '1', 8, 2599),
    ->     (9, 'Audi', 'A4', 2022, 55.00, '0', 4, 2500 ),
    ->     (10, 'Lexus', 'ES', 2023, 54.00, '1', 4, 2500 );
Query OK, 10 rows affected (0.03 sec)
Records: 10  Duplicates: 0  Warnings: 0

mysql> select * from Vehicle;
+-----------+-----------+----------+------+-----------+-----------+-------------------+----------------+
| vehicleID | make      | model    | year | dailyRate | available | passengerCapacity | engineCapacity |
+-----------+-----------+----------+------+-----------+-----------+-------------------+----------------+
|         1 | Toyota    | Camry    | 2022 |     50.00 |         1 |                 5 |           1450 |
|         2 | Honda     | Civic    | 2023 |     45.00 |         1 |                 7 |           1500 |
|         3 | Ford      | Focus    | 2022 |     48.00 |         0 |                 4 |           1400 |
|         4 | Nissan    | Altima   | 2023 |     52.00 |         1 |                 7 |           1200 |
|         5 | Chevrolet | Malibu   | 2022 |     47.00 |         1 |                 4 |           1800 |
|         6 | Hyundai   | Sonata   | 2023 |     49.00 |         0 |                 7 |           1400 |
|         7 | BMW       | 3 Series | 2023 |     60.00 |         1 |                 7 |           2499 |
|         8 | Mercedes  | C-Class  | 2022 |     58.00 |         1 |                 8 |           2599 |
|         9 | Audi      | A4       | 2022 |     55.00 |         0 |                 4 |           2500 |
|        10 | Lexus     | ES       | 2023 |     54.00 |         1 |                 4 |           2500 |
+-----------+-----------+----------+------+-----------+-----------+-------------------+----------------+
10 rows in set (0.00 sec)

mysql>
```

2. Customer Table:

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



```
mysql> CREATE TABLE Customer (
    ->     customerID INT PRIMARY KEY,
    ->     firstName VARCHAR(50),
    ->     lastName VARCHAR(50),
    ->     email VARCHAR(100),
    ->     phoneNumber VARCHAR(15)
    -> );
Query OK, 0 rows affected (0.08 sec)

mysql> DESC Customer;
+-------------+--------------+------+-----+---------+-------+
| Field       | Type         | Null | Key | Default | Extra |
+-------------+--------------+------+-----+---------+-------+
| customerID  | int          | NO   | PRI | NULL    |       |
| firstName   | varchar(50)  | YES  |     | NULL    |       |
| lastName    | varchar(50)  | YES  |     | NULL    |       |
| email       | varchar(100) | YES  |     | NULL    |       |
| phoneNumber | varchar(15)  | YES  |     | NULL    |       |
+-------------+--------------+------+-----+---------+-------+
5 rows in set (0.00 sec)

mysql>
```

DATA 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');

```
mysql> 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');
Query OK, 10 rows affected (0.02 sec)
Records: 10  Duplicates: 0  Warnings: 0

mysql> SELECT * FROM Customer;
+------------+-----------+----------+----------------------+--------------+
| customerID | firstName | lastName | email                | phoneNumber  |
+------------+-----------+----------+----------------------+--------------+
|          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 |
+------------+-----------+----------+----------------------+--------------+
10 rows in set (0.00 sec)

mysql>
```

3. Lease Table:

CREATE TABLE Lease (

-> leaseID INT PRIMARY KEY,

-> vehicleID INT,

-> customerID INT,

-> startDate DATE,

-> endDate DATE,

-> leaseType VARCHAR(20),

-> FOREIGN KEY (vehicleID) REFERENCES Vehicle(vehicleID),

-> FOREIGN KEY (customerID) REFERENCES Customer(customerID)

->);

```
mysql> CREATE TABLE Lease (
    ->     leaseID INT PRIMARY KEY,
    ->     vehicleID INT,
    ->     customerID INT,
    ->     startDate DATE,
    ->     endDate DATE,
    ->     leaseType VARCHAR(20),
    ->     FOREIGN KEY (vehicleID) REFERENCES Vehicle(vehicleID),
    ->     FOREIGN KEY (customerID) REFERENCES Customer(customerID)
    -> );
Query OK, 0 rows affected (0.13 sec)

mysql> DESC  Lease;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| leaseID    | int         | NO   | PRI | NULL    |       |
| vehicleID  | int         | YES  | MUL | NULL    |       |
| customerID | int         | YES  | MUL | NULL    |       |
| startDate  | date        | YES  |     | NULL    |       |
| endDate    | date        | YES  |     | NULL    |       |
| leaseType  | varchar(20) | YES  |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
6 rows in set (0.00 sec)

mysql>
```

DATA TABLE;

INSERT INTO Lease (leaseID, vehicleID, customerID, startDate, endDate, leaseType)

-> 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');

```
mysql> SELECT * FROM  Lease;
+---------+-----------+------------+------------+------------+-----------+
| leaseID | vehicleID | customerID | startDate  | endDate    | leaseType |
+---------+-----------+------------+------------+------------+-----------+
|       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   |
+---------+-----------+------------+------------+------------+-----------+
10 rows in set (0.00 sec)

mysql>
```

4. Payment Table:

CREATE TABLE Payment (

-> paymentID INT PRIMARY KEY,

-> leaseID INT,

-> paymentDate DATE,

-> amount DECIMAL(10, 2),

-> FOREIGN KEY (leaseID) REFERENCES Lease(leaseID)

->);

```
mysql> CREATE TABLE Payment (
    ->     paymentID INT PRIMARY KEY,
    ->     leaseID INT,
    ->     paymentDate DATE,
    ->     amount DECIMAL(10, 2),
    ->     FOREIGN KEY (leaseID) REFERENCES Lease(leaseID)
    -> );
Query OK, 0 rows affected (0.11 sec)

mysql> DESC Payment;
+-------------+---------------+------+-----+---------+-------+
| Field       | Type          | Null | Key | Default | Extra |
+-------------+---------------+------+-----+---------+-------+
| paymentID   | int           | NO   | PRI | NULL    |       |
| leaseID     | int           | YES  | MUL | NULL    |       |
| paymentDate | date          | YES  |     | NULL    |       |
| amount      | decimal(10,2) | YES  |     | NULL    |       |
+-------------+---------------+------+-----+---------+-------+
4 rows in set (0.00 sec)

mysql>
```

DATA 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),

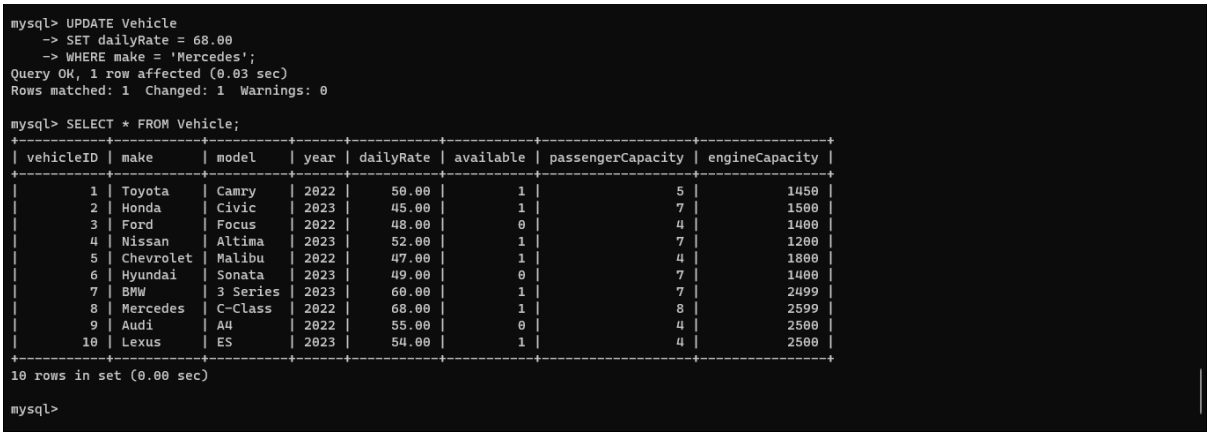
-> (10, 10, '2023-10-25', 1500.00);

```
mysql> 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);
Query OK, 10 rows affected (0.07 sec)
Records: 10  Duplicates: 0  Warnings: 0

mysql> SELECT * FROM Payment;
+-----------+---------+-------------+---------+
| paymentID | leaseID | paymentDate | 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 |
+-----------+---------+-------------+---------+
10 rows in set (0.00 sec)

mysql>
```

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

QUERY:-

```
UPDATE Vehicle
    -> SET dailyRate = 68.00
    -> WHERE make = 'Mercedes';
```



```
mysql> UPDATE Vehicle
    -> SET dailyRate = 68.00
    -> WHERE make = 'Mercedes';
Query OK, 1 row affected (0.03 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> SELECT * FROM Vehicle;
+-----------+-----------+----------+------+-----------+-----------+-------------------+----------------+
| vehicleID | make      | model    | year | dailyRate | available | passengerCapacity | engineCapacity |
+-----------+-----------+----------+------+-----------+-----------+-------------------+----------------+
|         1 | Toyota    | Camry    | 2022 |     50.00 |         1 |                 5 |           1450 |
|         2 | Honda     | Civic    | 2023 |     45.00 |         1 |                 7 |           1500 |
|         3 | Ford      | Focus    | 2022 |     48.00 |         0 |                 4 |           1400 |
|         4 | Nissan    | Altima   | 2023 |     52.00 |         1 |                 7 |           1200 |
|         5 | Chevrolet | Malibu   | 2022 |     47.00 |         1 |                 4 |           1800 |
|         6 | Hyundai   | Sonata   | 2023 |     49.00 |         0 |                 7 |           1400 |
|         7 | BMW       | 3 Series | 2023 |     60.00 |         1 |                 7 |           2499 |
|         8 | Mercedes  | C-Class  | 2022 |     68.00 |         1 |                 8 |           2599 |
|         9 | Audi      | A4       | 2022 |     55.00 |         0 |                 4 |           2500 |
|        10 | Lexus     | ES       | 2023 |     54.00 |         1 |                 4 |           2500 |
+-----------+-----------+----------+------+-----------+-----------+-------------------+----------------+
10 rows in set (0.00 sec)

mysql>
```

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

QUERY:-

```
DELETE FROM Payment
    -> WHERE leaseID IN (SELECT leaseID FROM Lease WHERE
customerID = (SELECT customerID FROM Customer WHERE
firstName = 'David' AND lastName = 'Lee
'));
```

```
mysql> select * from  Payment;
+-----------+---------+-----------------+---------+
| 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 |
|         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 |
+-----------+---------+-----------------+---------+
9 rows in set (0.00 sec)

mysql>
```

```
DELETE FROM Lease
    -> WHERE customerID = (SELECT customerID FROM
Customer WHERE firstName = 'David' AND lastName = 'Lee');
```

```
mysql> select * from Lease;
+---------+-----------+------------+------------+------------+-----------+
| leaseID | vehicleID | customerID | startDate  | endDate    | leaseType |
+---------+-----------+------------+------------+------------+-----------+
|       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   |
|       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   |
+---------+-----------+------------+------------+------------+-----------+
9 rows in set (0.00 sec)
```

DELETE FROM Customer

-> WHERE firstName = 'David' AND lastName = 'Lee';

```
mysql> select * from Customer;
+------------+-----------+----------+-----------------------+--------------+
| customerID | firstName | lastName | email                 | phoneNumber  |
+------------+-----------+----------+-----------------------+--------------+
|          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 |
|          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 |
+------------+-----------+----------+-----------------------+--------------+
9 rows in set (0.00 sec)
```

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

QUERY:-

ALTER TABLE Payment

-> CHANGE COLUMN paymentDate transactionDate DATE;

```
mysql> ALTER TABLE Payment
    -> CHANGE COLUMN paymentDate transactionDate DATE;
Query OK, 0 rows affected (0.07 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> DESC Payment;
+-----------------+---------------+------+-----+---------+-------+
| 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 rows in set (0.00 sec)

mysql>
```

4.Find a specific customer by email.

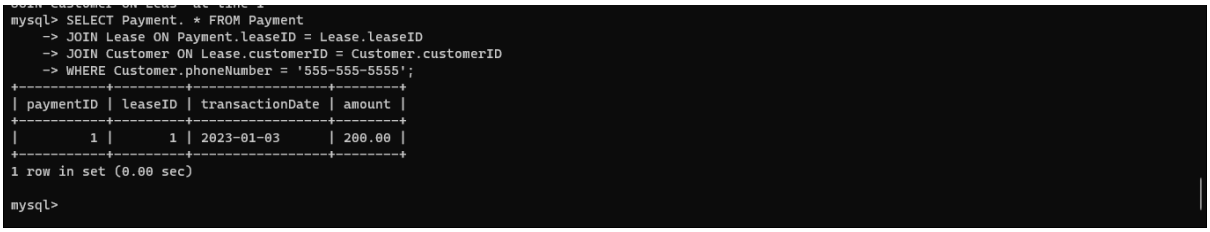
SELECT * FROM Customer

-> WHERE email = 'johndoe@example.com';

```
mysql> SELECT * FROM Customer
    -> WHERE email = 'johndoe@example.com';
+------------+-----------+----------+---------------------+--------------+
| customerID | firstName | lastName | email               | phoneNumber  |
+------------+-----------+----------+---------------------+--------------+
|          1 | John      | Doe      | johndoe@example.com | 555-555-5555 |
+------------+-----------+----------+---------------------+--------------+
1 row in set (0.00 sec)

mysql>
```

5. Get active leases for a specific customer.
   QUERY:-
   SELECT Lease.* FROM Lease
   -> JOIN Customer ON Lease.customerID = Customer.customerID
   -> WHERE Customer.email = 'johndoe@example.com'
   -> AND '2023-01-03' BETWEEN Lease.startDate AND Lease.endDate;

6. 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 = '555-555-5555';



7. Calculate the average daily rate of all available cars.
   QUERY:-
   SELECT AVG(Vehicle.dailyRate) AS averageDailyRate
   -> FROM Vehicle
   -> LEFT JOIN Lease ON Vehicle.vehicleID = Lease.vehicleID
   -> WHERE Vehicle.available = '1' OR Lease.vehicleID IS NULL;

```
mysql> SELECT AVG(Vehicle.dailyRate) AS averageDailyRate
    -> FROM Vehicle
    -> LEFT JOIN Lease ON Vehicle.vehicleID = Lease.vehicleID
    -> WHERE Vehicle.available = '1' OR Lease.vehicleID IS NULL;
+------------------+
| averageDailyRate |
+------------------+
|        53.200000 |
+------------------+
1 row in set (0.00 sec)

mysql>
```

8. Find the car with the highest daily rate.

   QUERY:-

   SELECT * FROM Vehicle

   -> ORDER BY dailyRate DESC

   -> LIMIT 1;

```
mysql> SELECT * FROM Vehicle
    -> ORDER BY dailyRate DESC
    -> LIMIT 1;
+-----------+----------+---------+------+-----------+-----------+-------------------+----------------+
| vehicleID | make     | model   | year | dailyRate | available | passengerCapacity | engineCapacity |
+-----------+----------+---------+------+-----------+-----------+-------------------+----------------+
|         8 | Mercedes | C-Class | 2022 |     68.00 |         1 |                 8 |           2599 |
+-----------+----------+---------+------+-----------+-----------+-------------------+----------------+
1 row in set (0.00 sec)

mysql>
```

9. Retrieve all cars leased by a specific customer.

   QUERY:-

   SELECT Vehicle.* FROM Vehicle

   -> JOIN Lease ON Vehicle.vehicleID = Lease.vehicleID

   -> JOIN Customer ON Lease.customerID = Customer.customerID

   -> WHERE Customer.email = 'johndoe@example.com';

```
mysql> SELECT Vehicle.* FROM Vehicle
    -> JOIN Lease ON Vehicle.vehicleID = Lease.vehicleID
    -> JOIN Customer ON Lease.customerID = Customer.customerID
    -> WHERE Customer.email = 'johndoe@example.com';
+-----------+--------+-------+------+-----------+-----------+-------------------+----------------+
| vehicleID | make   | model | year | dailyRate | available | passengerCapacity | engineCapacity |
+-----------+--------+-------+------+-----------+-----------+-------------------+----------------+
|         1 | Toyota | Camry | 2022 |     50.00 |         1 |                 5 |           1450 |
+-----------+--------+-------+------+-----------+-----------+-------------------+----------------+
1 row in set (0.00 sec)

mysql>
```

10. Find the details of the most recent lease.

    QUERY:-

    SELECT * FROM Lease

    -> ORDER BY startDate DESC

    -> LIMIT 1;

```
mysql> SELECT * FROM Lease
    -> ORDER BY startDate DESC
    -> LIMIT 1;
+---------+-----------+------------+------------+------------+-----------+
| leaseID | vehicleID | customerID | startDate  | endDate    | leaseType |
+---------+-----------+------------+------------+------------+-----------+
|      10 |        10 |         10 | 2023-10-10 | 2023-10-31 | Monthly   |
+---------+-----------+------------+------------+------------+-----------+
1 row in set (0.00 sec)

mysql>
```

11. List all payments made in the year 2023.

QUERY:-

```
SELECT Payment.* FROM Payment
    -> JOIN Lease ON Payment.leaseID = Lease.leaseID
    -> JOIN Customer ON Lease.customerID =
Customer.customerID
    -> WHERE YEAR(Payment.transactionDate) = 2023;
```

```
mysql> SELECT Payment.* FROM Payment
    -> JOIN Lease ON Payment.leaseID = Lease.leaseID
    -> JOIN Customer ON Lease.customerID = Customer.customerID
    -> WHERE YEAR(Payment.transactionDate) = 2023;
+-----------+---------+-----------------+---------+
| 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 |
|         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 |
+-----------+---------+-----------------+---------+
9 rows in set (0.00 sec)
```

12. Retrieve customers who have not made any payments.

QUERY:-

```
SELECT * FROM Customer
    -> WHERE customerID NOT IN (
    -> SELECT DISTINCT customerID
    -> FROM Payment
    -> );
```

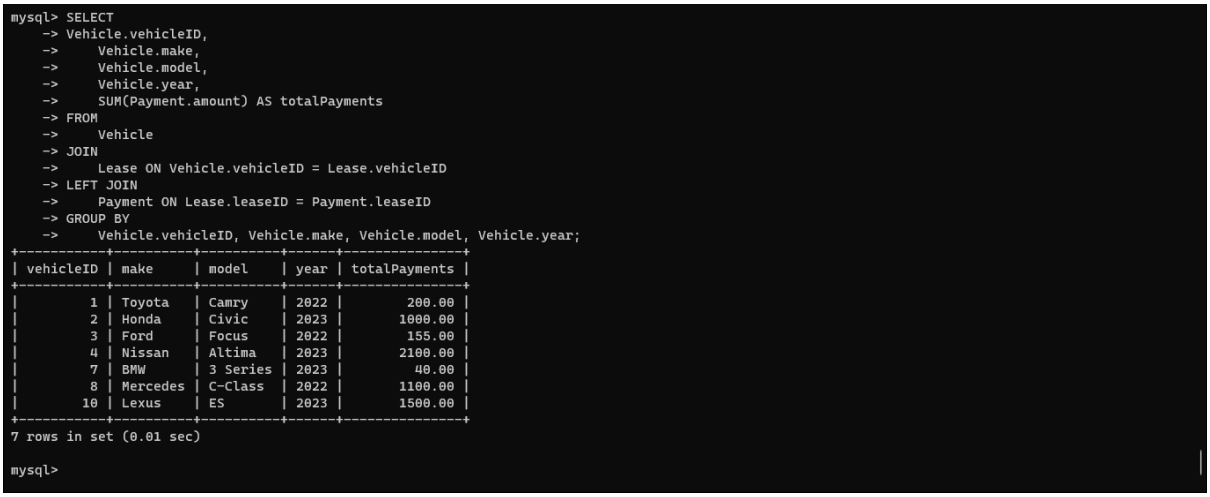
```
mysql> SELECT * FROM Customer
    -> WHERE customerID NOT IN (
    -> SELECT DISTINCT customerID
    -> FROM Payment
    -> );
Empty set (0.00 sec)

mysql>
```

13. Retrieve Car Details and Their Total Payments.

QUERY:-

```
SELECT
    -> Vehicle.vehicleID,
    ->     Vehicle.make,
    ->     Vehicle.model,
    ->     Vehicle.year,
    ->     SUM(Payment.amount) AS totalPayments
    -> FROM
```

-> Vehicle
-> JOIN
-> Lease ON Vehicle.vehicleID = Lease.vehicleID
-> LEFT JOIN
-> Payment ON Lease.leaseID = Payment.leaseID
-> GROUP BY
-> Vehicle.vehicleID, Vehicle.make, Vehicle.model, Vehicle.year;



```
mysql> SELECT
    -> Vehicle.vehicleID,
    ->     Vehicle.make,
    ->     Vehicle.model,
    ->     Vehicle.year,
    ->     SUM(Payment.amount) AS totalPayments
    -> FROM
    ->     Vehicle
    -> JOIN
    ->     Lease ON Vehicle.vehicleID = Lease.vehicleID
    -> LEFT JOIN
    ->     Payment ON Lease.leaseID = Payment.leaseID
    -> GROUP BY
    ->     Vehicle.vehicleID, Vehicle.make, Vehicle.model, Vehicle.year;
+-----------+----------+----------+------+---------------+
| vehicleID | make     | model    | year | totalPayments |
+-----------+----------+----------+------+---------------+
|         1 | Toyota   | Camry    | 2022 |        200.00 |
|         2 | Honda    | Civic    | 2023 |       1000.00 |
|         3 | Ford     | Focus    | 2022 |        155.00 |
|         4 | Nissan   | Altima   | 2023 |       2100.00 |
|         7 | BMW      | 3 Series | 2023 |         40.00 |
|         8 | Mercedes | C-Class  | 2022 |       1100.00 |
|        10 | Lexus    | ES       | 2023 |       1500.00 |
+-----------+----------+----------+------+---------------+
7 rows in set (0.01 sec)

mysql>
```

14. Calculate Total Payments for Each Customer.

QUERY:-

SELECT
-> Customer.customerID,
-> Customer.firstName,
-> Customer.lastName,
-> SUM(Payment.amount) AS totalPayments
-> FROM
-> Customer
-> LEFT JOIN
-> Lease ON Customer.customerID = Lease.customerID
-> LEFT JOIN
-> Payment ON Lease.leaseID = Payment.leaseID
-> GROUP BY
-> Customer.customerID, Customer.firstName, Customer.lastName;

```
mysql> SELECT
    ->     Customer.customerID,
    ->     Customer.firstName,
    ->     Customer.lastName,
    ->     SUM(Payment.amount) AS totalPayments
    -> FROM
    ->     Customer
    -> LEFT JOIN
    ->     Lease ON Customer.customerID = Lease.customerID
    -> LEFT JOIN
    ->     Payment ON Lease.leaseID = Payment.leaseID
    -> GROUP BY
    ->     Customer.customerID, Customer.firstName, Customer.lastName;
+------------+-----------+----------+---------------+
| customerID | firstName | lastName | totalPayments |
+------------+-----------+----------+---------------+
|          1 | John      | Doe      |        200.00 |
|          2 | Jane      | Smith    |       1000.00 |
|          3 | Robert    | Johnson  |       1355.00 |
|          4 | Sarah     | Brown    |        900.00 |
|          6 | Laura     | Hall     |          NULL |
|          7 | Michael   | Davis    |         40.00 |
|          8 | Emma      | Wilson   |       1100.00 |
|          9 | William   | Taylor   |          NULL |
|         10 | Olivia    | Adams    |       1500.00 |
+------------+-----------+----------+---------------+
9 rows in set (0.00 sec)

mysql>
```

15. List Car Details for Each Lease.

QUERY:-

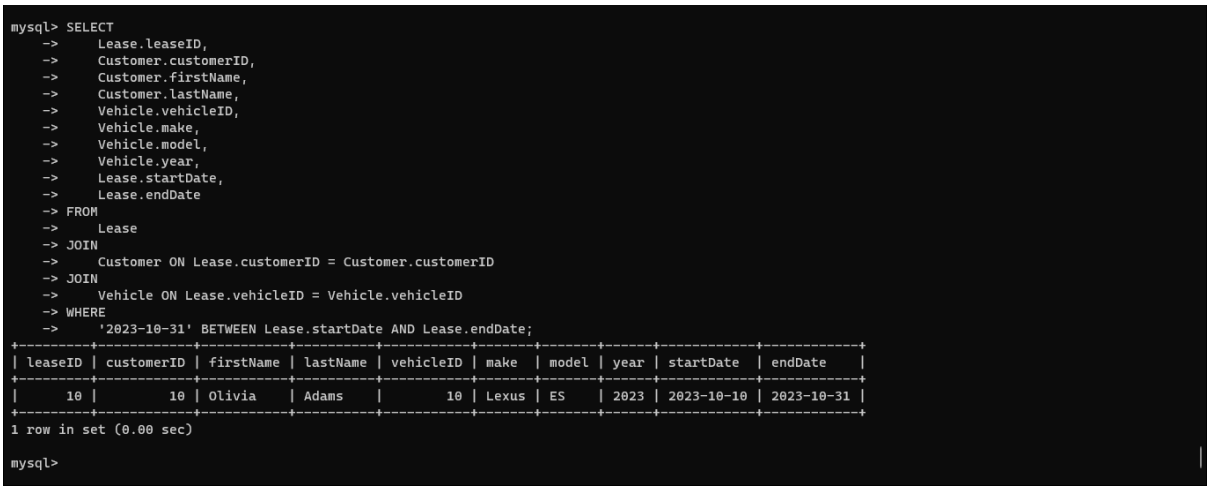
```
SELECT
    ->     Lease.leaseID,
    ->     Vehicle.vehicleID,
    ->     Vehicle.make,
    ->     Vehicle.model,
    ->     Vehicle.year
    -> FROM
    ->     Lease
    -> JOIN
    ->     Vehicle ON Lease.vehicleID = Vehicle.vehicleID;
```

```
mysql> SELECT
    ->     Lease.leaseID,
    ->     Vehicle.vehicleID,
    ->     Vehicle.make,
    ->     Vehicle.model,
    ->     Vehicle.year
    -> FROM
    ->     Lease
    -> JOIN
    ->     Vehicle ON Lease.vehicleID = Vehicle.vehicleID;
+---------+-----------+----------+----------+------+
| leaseID | vehicleID | make     | model    | year |
+---------+-----------+----------+----------+------+
|       1 |         1 | Toyota   | Camry    | 2022 |
|       2 |         2 | Honda    | Civic    | 2023 |
|       3 |         3 | Ford     | Focus    | 2022 |
|       9 |         3 | Ford     | Focus    | 2022 |
|       4 |         4 | Nissan   | Altima   | 2023 |
|       6 |         4 | Nissan   | Altima   | 2023 |
|       7 |         7 | BMW      | 3 Series | 2023 |
|       8 |         8 | Mercedes | C-Class  | 2022 |
|      10 |        10 | Lexus    | ES       | 2023 |
+---------+-----------+----------+----------+------+
9 rows in set (0.00 sec)

mysql>
```

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

QUERY:-

```
SELECT
    ->     Lease.leaseID,
    ->     Customer.customerID,
    ->     Customer.firstName,
    ->     Customer.lastName,
    ->     Vehicle.vehicleID,
    ->     Vehicle.make,
    ->     Vehicle.model,
    ->     Vehicle.year,
    ->     Lease.startDate,
    ->     Lease.endDate
    -> FROM
    ->     Lease
    -> JOIN
    ->     Customer ON Lease.customerID = Customer.customerID
    -> JOIN
    ->     Vehicle ON Lease.vehicleID = Vehicle.vehicleID
    -> WHERE
    ->     '2023-10-31' BETWEEN Lease.startDate AND
Lease.endDate;
```



```
mysql> SELECT
    ->     Lease.leaseID,
    ->     Customer.customerID,
    ->     Customer.firstName,
    ->     Customer.lastName,
    ->     Vehicle.vehicleID,
    ->     Vehicle.make,
    ->     Vehicle.model,
    ->     Vehicle.year,
    ->     Lease.startDate,
    ->     Lease.endDate
    -> FROM
    ->     Lease
    -> JOIN
    ->     Customer ON Lease.customerID = Customer.customerID
    -> JOIN
    ->     Vehicle ON Lease.vehicleID = Vehicle.vehicleID
    -> WHERE
    ->     '2023-10-31' BETWEEN Lease.startDate AND Lease.endDate;
+---------+------------+-----------+----------+-----------+-------+-------+------+------------+------------+
| leaseID | customerID | firstName | lastName | vehicleID | make  | model | year | startDate  | endDate    |
+---------+------------+-----------+----------+-----------+-------+-------+------+------------+------------+
|      10 |         10 | Olivia    | Adams    |        10 | Lexus | ES    | 2023 | 2023-10-10 | 2023-10-31 |
+---------+------------+-----------+----------+-----------+-------+-------+------+------------+------------+
1 row in set (0.00 sec)

mysql>
```

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

QUERY:-

```
SELECT
    ->     Customer.customerID,
```

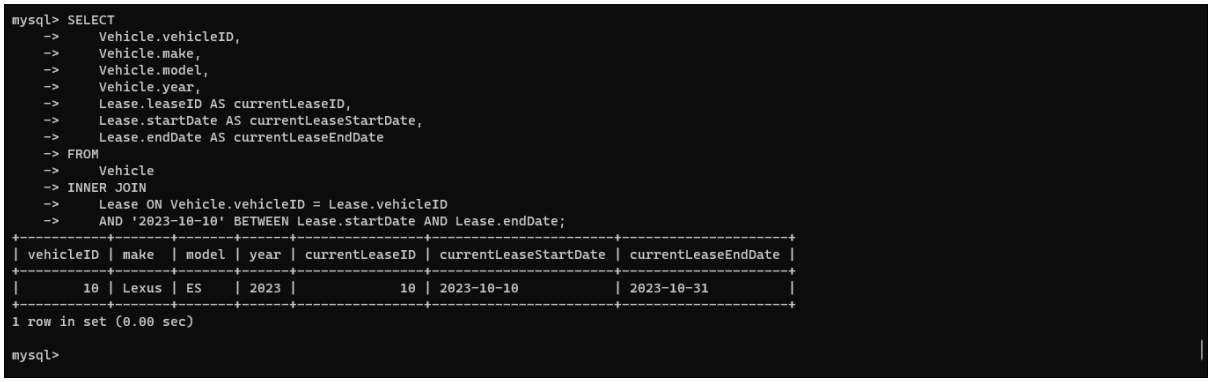
```
    ->     Customer.firstName,
    ->     Customer.lastName,
    ->     SUM(Payment.amount) AS totalPayments
    -> FROM
    ->     Customer
    -> JOIN
    ->     Lease ON Customer.customerID = Lease.customerID
    -> JOIN
    ->     Payment ON Lease.leaseID = Payment.leaseID
    -> GROUP BY
    ->     Customer.customerID, Customer.firstName,
Customer.lastName
    -> ORDER BY
    ->     totalPayments DESC
    -> LIMIT 1;
```

18. List All Cars with Their Current Lease Information.

QUERY:-

```
SELECT
    ->     Vehicle.vehicleID,
    ->     Vehicle.make,
    ->     Vehicle.model,
    ->     Vehicle.year,
    ->     Lease.leaseID AS currentLeaseID,
    ->     Lease.startDate AS currentLeaseStartDate,
    ->     Lease.endDate AS currentLeaseEndDate
```

```
    -> FROM
    ->     Vehicle
    -> INNER JOIN
    ->     Lease ON Vehicle.vehicleID = Lease.vehicleID
    ->     AND '2023-10-10' BETWEEN Lease.startDate AND
Lease.endDate;
```



```
mysql> SELECT
    ->     Vehicle.vehicleID,
    ->     Vehicle.make,
    ->     Vehicle.model,
    ->     Vehicle.year,
    ->     Lease.leaseID AS currentLeaseID,
    ->     Lease.startDate AS currentLeaseStartDate,
    ->     Lease.endDate AS currentLeaseEndDate
    -> FROM
    ->     Vehicle
    -> INNER JOIN
    ->     Lease ON Vehicle.vehicleID = Lease.vehicleID
    ->     AND '2023-10-10' BETWEEN Lease.startDate AND Lease.endDate;
+-----------+-------+-------+------+----------------+----------------------+--------------------+
| vehicleID | make  | model | year | currentLeaseID | currentLeaseStartDate | currentLeaseEndDate |
+-----------+-------+-------+------+----------------+----------------------+--------------------+
|        10 | Lexus | ES    | 2023 |             10 | 2023-10-10           | 2023-10-31         |
+-----------+-------+-------+------+----------------+----------------------+--------------------+
1 row in set (0.00 sec)

mysql>
```