Creating Database named Car Rental System:

```
mysql> CREATE DATABASE Car_Rental_System;
Query OK, 1 row affected (0.02 sec)

mysql> USE Car_Rental_System;
Database changed
```

Creating Tables:

Vehicle Table:

```
mysql> 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
    -> );
Query OK, 0 rows affected (0.06 sec)
```

Customer Table:

```
mysql> CREATE TABLE Customer (
    -> customerID INT PRIMARY KEY,
    -> firstName VARCHAR(255),
    -> lastName VARCHAR(255),
    -> email VARCHAR(255),
    -> phoneNumber VARCHAR(20)
    -> );
Query OK, 0 rows affected (0.04 sec)
```

Lease Table:

```
mysql> 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)
    -> );
Query OK, 0 rows affected (0.10 sec)
```

Payment Table:

```
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)
```

Inserting records into each tables Vehicle and Customer Table:

```
mysql> 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);
Query OK, 7 rows affected (0.01 sec)
Records: 7  Duplicates: 0  Warnings: 0
```

```
mysql> INSERT INTO Vehicle (vehicleID, make, model, year, dailyRate, status, passengerCapacity, engineCapacity)
    -> VALUES
    ->     (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);
Query OK, 3 rows affected (0.02 sec)
Records: 3  Duplicates: 0  Warnings: 0
```

```
mysql> select * from vehicle;
+-----------+-----------+----------+------+-----------+--------------+-------------------+----------------+
| 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 |     58.00 | available    |                 8 |           2599 |
|         9 | Audi      | A4       | 2022 |     55.00 | notAvailable |                 4 |           2500 |
|        10 | Lexus     | ES       | 2023 |     54.00 | available    |                 4 |           2500 |
+-----------+-----------+----------+------+-----------+--------------+-------------------+----------------+
10 rows in set (0.00 sec)
```

```
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)
```

Inserting into Lease table:

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

```
mysql> SELECT * FROM Lease;
+---------+-----------+------------+------------+------------+---------+
| 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 |         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)
```

Inserting into Payment table:

```
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 (0.02 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)
```

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

Before updating the daily rate of Mercedes:

```
mysql> select * from vehicle;
+-----------+-----------+----------+------+-----------+--------------+-------------------+----------------+
| 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 |     58.00 | available    |                 8 |           2599 |
|         9 | Audi      | A4       | 2022 |     55.00 | notAvailable |                 4 |           2500 |
|        10 | Lexus     | ES       | 2023 |     54.00 | available    |                 4 |           2500 |
+-----------+-----------+----------+------+-----------+--------------+-------------------+----------------+
10 rows in set (0.00 sec)
```

After Updating the daily rate for a Mercedes car to 68:

```
mysql> UPDATE Vehicle SET dailyRate=68 WHERE make LIKE 'Mercedes';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> SELECT * FROM Vehicle;
+-----------+-----------+----------+------+-----------+--------------+-------------------+----------------+
| 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 |
+-----------+-----------+----------+------+-----------+--------------+-------------------+----------------+
10 rows in set (0.00 sec)
```

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

```
mysql> DELETE FROM Payment
    -> WHERE leaseID IN (SELECT leaseID FROM Lease WHERE customerID = 3);
Query OK, 3 rows affected (0.01 sec)
```

Deleting customer details from payment table based on their customer and lease ID.

```
mysql> SELECT * FROM Payment;
+-----------+---------+-------------+---------+
| paymentID | leaseID | paymentDate | amount  |
+-----------+---------+-------------+---------+
|         1 |       1 | 2023-01-03  |  200.00 |
|         2 |       2 | 2023-02-20  | 1000.00 |
|         4 |       4 | 2023-04-25  |  900.00 |
|         5 |       5 | 2023-05-07  |   60.00 |
|         7 |       7 | 2023-07-03  |   40.00 |
|         8 |       8 | 2023-08-14  | 1100.00 |
|        10 |      10 | 2023-10-25  | 1500.00 |
+-----------+---------+-------------+---------+
7 rows in set (0.00 sec)
```

Deleting the customer from the lease table based on a specific customer:

```
mysql> DELETE FROM Lease
    -> WHERE customerID = 3;
Query OK, 3 rows affected (0.00 sec)
```

```
mysql> SELECT * FROM Lease;
+---------+-----------+------------+------------+------------+---------+
| 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 |
|       4 |         4 |          4 | 2023-04-20 | 2023-04-30 | Monthly |
|       5 |         5 |          5 | 2023-05-05 | 2023-05-10 | Daily   |
|       7 |         7 |          7 | 2023-07-01 | 2023-07-10 | Daily   |
|       8 |         8 |          8 | 2023-08-12 | 2023-08-15 | Monthly |
|      10 |        10 |         10 | 2023-10-10 | 2023-10-31 | Monthly |
+---------+-----------+------------+------------+------------+---------+
7 rows in set (0.00 sec)
```

Deleting the customer from the customer table based on customer id:

```
mysql> DELETE FROM Customer
    -> WHERE customerID = 3;
Query OK, 1 row affected (0.01 sec)
```

```
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 |
|          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 |
+------------+-----------+----------+----------------------+--------------+
9 rows in set (0.00 sec)
```

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

```
mysql> ALTER TABLE Payment CHANGE paymentDate transactionDate DATE;
Query OK, 0 rows affected (0.03 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

After Renamed:

```
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)
```

4. Find a specific customer by email.

```
mysql> SELECT * FROM Customer WHERE email LIKE 'emma@example.com';
+------------+-----------+----------+------------------+--------------+
| customerID | firstName | lastName | email            | phoneNumber  |
+------------+-----------+----------+------------------+--------------+
|          8 | Emma      | Wilson   | emma@example.com | 555-432-1098 |
+------------+-----------+----------+------------------+--------------+
1 row in set (0.00 sec)
```

5. Get active leases for a specific customer.

```
mysql> SELECT l.leaseID,v.vehicleID,v.make,v.model,v.year,l.startDate,l.endDate,l.type
    -> FROM Lease l
    -> JOIN Vehicle v ON l.vehicleID = v.vehicleID
    -> WHERE l.customerID = 5 AND v.status = 'Available';
+---------+-----------+-----------+--------+------+-----------+-----------+-------+
| leaseID | vehicleID | make      | model  | year | startDate | endDate   | type  |
+---------+-----------+-----------+--------+------+-----------+-----------+-------+
|       5 |         5 | Chevrolet | Malibu | 2022 | 2023-05-05 | 2023-05-10 | Daily |
+---------+-----------+-----------+--------+------+-----------+-----------+-------+
1 row in set (0.00 sec)
```

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

```
mysql> SELECT c.customerid, l.leaseid, p.*
    -> FROM lease l
    -> JOIN customer c ON l.customerid = c.customerid
    -> JOIN payment p ON l.leaseid = p.leaseid
    -> WHERE c.phoneNumber LIKE '555-555-5555';
+------------+---------+-----------+---------+-----------------+--------+
| customerid | leaseid | paymentID | leaseID | transactionDate | amount |
+------------+---------+-----------+---------+-----------------+--------+
|          1 |       1 |         1 |       1 | 2023-01-03      | 200.00 |
+------------+---------+-----------+---------+-----------------+--------+
1 row in set (0.01 sec)
```

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

```
mysql> SELECT AVG(dailyRate) AS Average_daily_rate_For_Avl_Cars
    -> FROM Vehicle WHERE Status='Available';
+---------------------------------+
| Average_daily_rate_For_Avl_Cars |
+---------------------------------+
|                       53.714286 |
+---------------------------------+
1 row in set (0.00 sec)
```

8. Find the car with the highest daily rate.

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

9. Retrieve all cars leased by a specific customer.

```
mysql> SELECT * FROM vehicle
    -> WHERE vehicleID IN (
    -> SELECT vehicleID FROM lease
    -> WHERE customerID = (
    -> SELECT customerID FROM customer
    -> WHERE firstName = 'Sarah')
    -> );
+-----------+--------+--------+------+-----------+-----------+-------------------+----------------+
| vehicleID | make   | model  | year | dailyRate | status    | passengerCapacity | engineCapacity |
+-----------+--------+--------+------+-----------+-----------+-------------------+----------------+
|         4 | Nissan | Altima | 2023 |     52.00 | available |                 7 |           1200 |
+-----------+--------+--------+------+-----------+-----------+-------------------+----------------+
1 row in set (0.00 sec)
```

10. Find the details of the most recent lease.

```
mysql> SELECT c.firstName,v.make,l.*
    -> FROM Customer c
    -> JOIN Lease l ON l.CustomerID=c.CustomerID
    -> JOIN Vehicle v ON l.VehicleID=v.VehicleID
    -> WHERE startDate<CURDATE()
    -> ORDER BY startDate DESC LIMIT 1;
+-----------+-------+---------+-----------+------------+------------+------------+---------+
| firstName | make  | leaseID | vehicleID | customerID | startDate  | endDate    | type    |
+-----------+-------+---------+-----------+------------+------------+------------+---------+
| Olivia    | Lexus |      10 |        10 |         10 | 2023-10-10 | 2023-10-31 | Monthly |
+-----------+-------+---------+-----------+------------+------------+------------+---------+
1 row in set (0.01 sec)
```

11. List all payments made in the year 2023.

```
mysql> SELECT paymentID,leaseID,amount,YEAR(transactionDate) AS Current_Year
    -> FROM Payment
    -> HAVING Current_Year=2023;
+-----------+---------+---------+--------------+
| paymentID | leaseID | amount  | Current_Year |
+-----------+---------+---------+--------------+
|         1 |       1 |  200.00 |         2023 |
|         2 |       2 | 1000.00 |         2023 |
|         4 |       4 |  900.00 |         2023 |
|         5 |       5 |   60.00 |         2023 |
|         7 |       7 |   40.00 |         2023 |
|         8 |       8 | 1100.00 |         2023 |
|        10 |      10 | 1500.00 |         2023 |
+-----------+---------+---------+--------------+
7 rows in set (0.00 sec)
```

12. Retrieve customers who have not made any payments.

```
mysql> SELECT c.*
    -> FROM customer c
    -> JOIN lease l ON l.customerID = c.customerID
    -> WHERE leaseID NOT IN (
    -> SELECT leaseID from payment);
Empty set (0.00 sec)
```

The result is empty set because every customers who have took lease had mad their payments as seeing the following tables :

```
mysql> select * from lease;
+---------+-----------+------------+------------+------------+---------+
| 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 |
|       4 |         4 |          4 | 2023-04-20 | 2023-04-30 | Monthly |
|       5 |         5 |          5 | 2023-05-05 | 2023-05-10 | Daily   |
|       7 |         7 |          7 | 2023-07-01 | 2023-07-10 | Daily   |
|       8 |         8 |          8 | 2023-08-12 | 2023-08-15 | Monthly |
|      10 |        10 |         10 | 2023-10-10 | 2023-10-31 | Monthly |
+---------+-----------+------------+------------+------------+---------+
7 rows in set (0.00 sec)

mysql> select * from payment;
+-----------+---------+-----------------+---------+
| paymentID | leaseID | transactionDate | amount  |
+-----------+---------+-----------------+---------+
|         1 |       1 | 2023-01-03      |  200.00 |
|         2 |       2 | 2023-02-20      | 1000.00 |
|         4 |       4 | 2023-04-25      |  900.00 |
|         5 |       5 | 2023-05-07      |   60.00 |
|         7 |       7 | 2023-07-03      |   40.00 |
|         8 |       8 | 2023-08-14      | 1100.00 |
|        10 |      10 | 2023-10-25      | 1500.00 |
+-----------+---------+-----------------+---------+
7 rows in set (0.00 sec)
```

13. Retrieve Car Details and Their Total Payments.

```
mysql> SELECT v.*,SUM(p.amount) AS totalPayments
    -> FROM Vehicle v
    -> LEFT JOIN
    -> Lease l ON v.vehicleID = l.vehicleID
    -> LEFT JOIN
    -> Payment p ON l.leaseID = p.leaseID
    -> GROUP BY v.vehicleID;
+-----------+----------+----------+------+-----------+-------------+-------------------+----------------+---------------+
| vehicleID | make     | model    | year | dailyRate | status      | passengerCapacity | engineCapacity | totalPayments |
+-----------+----------+----------+------+-----------+-------------+-------------------+----------------+---------------+
|         1 | Toyota   | Camry    | 2022 |     50.00 | available   |                 4 |           1450 |        200.00 |
|         2 | Honda    | Civic    | 2023 |     45.00 | available   |                 7 |           1500 |       1000.00 |
|         3 | Ford     | Focus    | 2022 |     48.00 | notAvailable |                4 |           1400 |          NULL |
|         4 | Nissan   | Altima   | 2023 |     52.00 | available   |                 7 |           1200 |        900.00 |
|         5 | Chevrolet | Malibu  | 2022 |     47.00 | available   |                 4 |           1800 |         60.00 |
|         6 | Hyundai  | Sonata   | 2023 |     49.00 | notAvailable |                7 |           1400 |          NULL |
|         7 | BMW      | 3 Series | 2023 |     60.00 | available   |                 7 |           2499 |         40.00 |
|         8 | Mercedes | C-Class  | 2022 |     68.00 | available   |                 8 |           2599 |       1100.00 |
|         9 | Audi     | A4       | 2022 |     55.00 | notAvailable |                4 |           2500 |          NULL |
|        10 | Lexus    | ES       | 2023 |     54.00 | available   |                 4 |           2500 |       1500.00 |
+-----------+----------+----------+------+-----------+-------------+-------------------+----------------+---------------+
10 rows in set (0.00 sec)
```

14. Calculate Total Payments for Each Customer.

```
mysql> SELECT c.*,SUM(p.amount) AS totalPayments
    -> FROM Customer c
    -> LEFT JOIN
    -> Lease l ON c.customerID = l.customerID
    -> LEFT JOIN
    -> Payment p ON l.leaseID = p.leaseID
    -> GROUP BY c.customerID;
+------------+-----------+----------+----------------------+-------------+---------------+
| customerID | firstName | lastName | email                | phoneNumber | totalPayments |
+------------+-----------+----------+----------------------+-------------+---------------+
|          1 | John      | Doe      | johndoe@example.com  | 555-555-5555 |        200.00 |
|          2 | Jane      | Smith    | janesmith@example.com | 555-123-4567 |       1000.00 |
|          4 | Sarah     | Brown    | sarah@example.com    | 555-456-7890 |        900.00 |
|          5 | David     | Lee      | david@example.com    | 555-987-6543 |         60.00 |
|          6 | Laura     | Hall     | laura@example.com    | 555-234-5678 |          NULL |
|          7 | Michael   | Davis    | michael@example.com  | 555-876-5432 |         40.00 |
|          8 | Emma      | Wilson   | emma@example.com     | 555-432-1098 |       1100.00 |
|          9 | William   | Taylor   | william@example.com  | 555-321-6547 |          NULL |
|         10 | Olivia    | Adams    | olivia@example.com   | 555-765-4321 |       1500.00 |
+------------+-----------+----------+----------------------+-------------+---------------+
9 rows in set (0.00 sec)
```

15. List Car Details for Each Lease.

```
mysql> SELECT v.vehicleID,v.make,v.model,v.dailyRate,v.passengerCapacity,
    -> l.leaseID,l.startDate,l.endDate,l.Type
    -> FROM Lease l
    -> LEFT JOIN Vehicle v ON l.vehicleID = v.vehicleID;
+-----------+-----------+----------+-----------+------------------+---------+------------+------------+---------+
| vehicleID | make      | model    | dailyRate | passengerCapacity | leaseID | startDate  | endDate    | Type    |
+-----------+-----------+----------+-----------+------------------+---------+------------+------------+---------+
|         1 | Toyota    | Camry    |     50.00 |                4 |       1 | 2023-01-01 | 2023-01-05 | Daily   |
|         2 | Honda     | Civic    |     45.00 |                7 |       2 | 2023-02-15 | 2023-02-28 | Monthly |
|         4 | Nissan    | Altima   |     52.00 |                7 |       4 | 2023-04-20 | 2023-04-30 | Monthly |
|         5 | Chevrolet | Malibu   |     47.00 |                4 |       5 | 2023-05-05 | 2023-05-10 | Daily   |
|         7 | BMW       | 3 Series |     60.00 |                7 |       7 | 2023-07-01 | 2023-07-10 | Daily   |
|         8 | Mercedes  | C-Class  |     68.00 |                8 |       8 | 2023-08-12 | 2023-08-15 | Monthly |
|        10 | Lexus     | ES       |     54.00 |                4 |      10 | 2023-10-10 | 2023-10-31 | Monthly |
+-----------+-----------+----------+-----------+------------------+---------+------------+------------+---------+
7 rows in set (0.00 sec)
```

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

```
mysql> SELECT l.leaseID,l.type,c.firstName,v.make,v.model,v.status
    -> FROM Lease l
    -> JOIN Customer c ON l.customerID = c.customerID
    -> JOIN Vehicle v ON l.vehicleID = v.vehicleID
    -> WHERE v.status = 'available';
+---------+---------+-----------+-----------+----------+-----------+
| leaseID | type    | firstName | make      | model    | status    |
+---------+---------+-----------+-----------+----------+-----------+
|       1 | Daily   | John      | Toyota    | Camry    | available |
|       2 | Monthly | Jane      | Honda     | Civic    | available |
|       4 | Monthly | Sarah     | Nissan    | Altima   | available |
|       5 | Daily   | David     | Chevrolet | Malibu   | available |
|       7 | Daily   | Michael   | BMW       | 3 Series | available |
|       8 | Monthly | Emma      | Mercedes  | C-Class  | available |
|      10 | Monthly | Olivia    | Lexus     | ES       | available |
+---------+---------+-----------+-----------+----------+-----------+
7 rows in set (0.00 sec)
```

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

```
mysql> SELECT c.*,SUM(p.amount) AS totalPayments
    -> 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 totalPayments DESC
    -> LIMIT 1;
+------------+-----------+----------+--------------------+--------------+---------------+
| customerID | firstName | lastName | email              | phoneNumber  | totalPayments |
+------------+-----------+----------+--------------------+--------------+---------------+
|         10 | Olivia    | Adams    | olivia@example.com | 555-765-4321 |       1500.00 |
+------------+-----------+----------+--------------------+--------------+---------------+
1 row in set (0.01 sec)
```

18. List All Cars with Their Current Lease Information

```
mysql> SELECT v.vehicleID,v.make,v.model,v.dailyRate,l.*
    -> FROM Vehicle v
    -> LEFT JOIN
    -> Lease l ON v.vehicleID = l.vehicleID;
+-----------+-----------+----------+-----------+---------+-----------+------------+------------+------------+---------+
| vehicleID | make      | model    | dailyRate | leaseID | vehicleID | customerID | startDate  | endDate    | type    |
+-----------+-----------+----------+-----------+---------+-----------+------------+------------+------------+---------+
|         1 | Toyota    | Camry    |     50.00 |       1 |         1 |          1 | 2023-01-01 | 2023-01-05 | Daily   |
|         2 | Honda     | Civic    |     45.00 |       2 |         2 |          2 | 2023-02-15 | 2023-02-28 | Monthly |
|         3 | Ford      | Focus    |     48.00 |    NULL |      NULL |       NULL | NULL       | NULL       | NULL    |
|         4 | Nissan    | Altima   |     52.00 |       4 |         4 |          4 | 2023-04-20 | 2023-04-30 | Monthly |
|         5 | Chevrolet | Malibu   |     47.00 |       5 |         5 |          5 | 2023-05-05 | 2023-05-10 | Daily   |
|         6 | Hyundai   | Sonata   |     49.00 |    NULL |      NULL |       NULL | NULL       | NULL       | NULL    |
|         7 | BMW       | 3 Series |     60.00 |       7 |         7 |          7 | 2023-07-01 | 2023-07-10 | Daily   |
|         8 | Mercedes  | C-Class  |     68.00 |       8 |         8 |          8 | 2023-08-12 | 2023-08-15 | Monthly |
|         9 | Audi      | A4       |     55.00 |    NULL |      NULL |       NULL | NULL       | NULL       | NULL    |
|        10 | Lexus     | ES       |     54.00 |      10 |        10 |         10 | 2023-10-10 | 2023-10-31 | Monthly |
+-----------+-----------+----------+-----------+---------+-----------+------------+------------+------------+---------+
10 rows in set (0.00 sec)
```