Coding Challenge - Car Rental System – SQL

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

```
mysql> UPDATE Vehicle
    -> SET dailyRate = 68.00
    -> WHERE make = 'Mercedes';
Query OK, 1 row affected (0.00 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         |                 4 |           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)
```

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

```
mysql> DELETE FROM Payment
    -> WHERE leaseID IN (SELECT leaseID FROM Lease WHERE customerID = 5);
Query OK, 1 row affected (0.01 sec)

mysql> DELETE FROM Lease
    -> WHERE customerID = 5;
Query OK, 1 row affected (0.00 sec)

mysql> DELETE FROM Customer
    -> WHERE customerID = 5;
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 |
|          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".

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

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)
```

4. Find a specific customer by email.

```
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)
```

5. Get active leases for a specific customer.

```
1 row in set (0.00 sec)

mysql> SELECT Lease.*, Customer.firstName, Customer.lastName
    -> FROM Lease
    -> JOIN Customer ON Lease.customerID = Customer.customerID
    -> WHERE Lease.customerID = 5
    -> AND Lease.endDate > CURDATE();
Empty set (0.00 sec)
```

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

```
mysql> SELECT Payment.*, Customer.firstName, Customer.lastName
    -> FROM Payment
    -> JOIN Lease ON Payment.leaseID = Lease.leaseID
    -> JOIN Customer ON Lease.customerID = Customer.customerID
    -> WHERE Customer.phoneNumber = '555-123-4567';
+-----------+---------+-----------------+---------+-----------+----------+
| paymentID | leaseID | transactionDate | amount  | firstName | lastName |
+-----------+---------+-----------------+---------+-----------+----------+
|         2 |       2 | 2023-02-20      | 1000.00 | Jane      | Smith    |
+-----------+---------+-----------------+---------+-----------+----------+
1 row in set (0.00 sec)
```

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

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

8. Find the car with the highest daily rate.

```
mysql> SELECT *
    -> FROM Vehicle
    -> WHERE dailyRate = (SELECT MAX(dailyRate) FROM Vehicle);
+-----------+----------+---------+------+-----------+-----------+-------------------+----------------+
| 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)
```

9. Retrieve all cars leased by a specific customer.

```
mysql> SELECT Vehicle.*
    -> FROM Vehicle
    -> JOIN Lease ON Vehicle.vehicleID = Lease.vehicleID
    -> JOIN Customer ON Lease.customerID = Customer.customerID
    -> WHERE Customer.customerID = 1;
+-----------+--------+-------+------+-----------+-----------+-------------------+----------------+
| vehicleID | make   | model | year | dailyRate | available | passengerCapacity | engineCapacity |
+-----------+--------+-------+------+-----------+-----------+-------------------+----------------+
|         1 | Toyota | Camry | 2022 |     50.00 | 1         |                 4 |           1450 |
+-----------+--------+-------+------+-----------+-----------+-------------------+----------------+
1 row in set (0.00 sec)
```

10. Find the details of the most recent lease.

```
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)
```

11. List all payments made in the year 2023.

```
mysql> SELECT *
    -> FROM Payment
    -> WHERE YEAR(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.

```
mysql> SELECT Customer.*
   -> FROM Customer
   -> LEFT JOIN Lease ON Customer.customerID = Lease.customerID
   -> LEFT JOIN Payment ON Lease.leaseID = Payment.leaseID
   -> WHERE Payment.paymentID IS NULL;
+------------+-----------+----------+--------------------+--------------+
| customerID | firstName | lastName | email              | phoneNumber  |
+------------+-----------+----------+--------------------+--------------+
|          6 | Laura     | Hall     | laura@example.com  | 555-234-5678 |
|          9 | William   | Taylor   | william@example.com | 555-321-6547 |
+------------+-----------+----------+--------------------+--------------+
2 rows in set (0.00 sec)

mysql>
```

13. Retrieve Car Details and Their Total Payments.

```
mysql> SELECT V.*, COALESCE(SUM(P.amount), 0) 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 | available | passengerCapacity | engineCapacity | totalPayments |
+-----------+-----------+----------+------+-----------+-----------+-----------------+----------------+---------------+
|         1 | Toyota    | Camry    | 2022 |     50.00 | 1         |                 4 |           1450 |        200.00 |
|         2 | Honda     | Civic    | 2023 |     45.00 | 1         |                 7 |           1500 |       1000.00 |
|         3 | Ford      | Focus    | 2022 |     48.00 | 0         |                 4 |           1400 |        155.00 |
|         4 | Nissan    | Altima   | 2023 |     52.00 | 1         |                 7 |           1200 |       2100.00 |
|         5 | Chevrolet | Malibu   | 2022 |     47.00 | 1         |                 4 |           1800 |          0.00 |
|         6 | Hyundai   | Sonata   | 2023 |     49.00 | 0         |                 7 |           1400 |          0.00 |
|         7 | BMW       | 3 Series | 2023 |     60.00 | 1         |                 7 |           2499 |         40.00 |
|         8 | Mercedes  | C-Class  | 2022 |     68.00 | 1         |                 8 |           2599 |       1100.00 |
|         9 | Audi      | A4       | 2022 |     55.00 | 0         |                 4 |           2500 |          0.00 |
|        10 | Lexus     | ES       | 2023 |     54.00 | 1         |                 4 |           2500 |       1500.00 |
+-----------+-----------+----------+------+-----------+-----------+-----------------+----------------+---------------+
10 rows in set (0.00 sec)
```

14. Calculate Total Payments for Each Customer.

```
mysql> SELECT C.customerID, C.firstName, C.lastName, COALESCE(SUM(P.amount), 0) 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 | totalPayments |
+------------+-----------+----------+---------------+
|          1 | John      | Doe      |        200.00 |
|          2 | Jane      | Smith    |       1000.00 |
|          3 | Robert    | Johnson  |       1355.00 |
|          4 | Sarah     | Brown    |        900.00 |
|          6 | Laura     | Hall     |          0.00 |
|          7 | Michael   | Davis    |         40.00 |
|          8 | Emma      | Wilson   |       1100.00 |
|          9 | William   | Taylor   |          0.00 |
|         10 | Olivia    | Adams    |       1500.00 |
+------------+-----------+----------+---------------+
9 rows in set (0.00 sec)
```

15. List Car Details for Each Lease.

```
mysql> SELECT L.*, V.make, V.model, V.year, V.dailyRate
    -> FROM Lease L
    -> JOIN Vehicle V ON L.vehicleID = V.vehicleID;
+---------+-----------+------------+------------+------------+-----------+----------+----------+------+-----------+
| leaseID | vehicleID | customerID | startDate  | endDate    | Leasetype | make     | model    | year | dailyRate |
+---------+-----------+------------+------------+------------+-----------+----------+----------+------+-----------+
|       1 |         1 |          1 | 2023-01-01 | 2023-01-05 | Daily     | Toyota   | Camry    | 2022 |     50.00 |
|       2 |         2 |          2 | 2023-02-15 | 2023-02-28 | Monthly   | Honda    | Civic    | 2023 |     45.00 |
|       3 |         3 |          3 | 2023-03-10 | 2023-03-15 | Daily     | Ford     | Focus    | 2022 |     48.00 |
|       4 |         4 |          4 | 2023-04-20 | 2023-04-30 | Monthly   | Nissan   | Altima   | 2023 |     52.00 |
|       6 |         4 |          3 | 2023-06-15 | 2023-06-30 | Monthly   | Nissan   | Altima   | 2023 |     52.00 |
|       7 |         7 |          7 | 2023-07-01 | 2023-07-10 | Daily     | BMW      | 3 Series | 2023 |     60.00 |
|       8 |         8 |          8 | 2023-08-12 | 2023-08-15 | Monthly   | Mercedes | C-Class  | 2022 |     68.00 |
|       9 |         3 |          3 | 2023-09-07 | 2023-09-10 | Daily     | Ford     | Focus    | 2022 |     48.00 |
|      10 |        10 |         10 | 2023-10-10 | 2023-10-31 | Monthly   | Lexus    | ES       | 2023 |     54.00 |
+---------+-----------+------------+------------+------------+-----------+----------+----------+------+-----------+
9 rows in set (0.00 sec)
```

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

```
9 rows in set (0.00 sec)

mysql> SELECT L.*, C.firstName, C.lastName, V.make, V.model, V.year, V.dailyRate
    -> FROM Lease L
    -> JOIN Customer C ON L.customerID = C.customerID
    -> JOIN Vehicle V ON L.vehicleID = V.vehicleID
    -> WHERE L.endDate > CURDATE();
Empty set (0.00 sec)
```

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

```
Empty set (0.00 sec)

mysql> SELECT C.customerID, C.firstName, C.lastName, COALESCE(SUM(P.amount), 0) AS totalAmountSpent
    -> 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
    -> ORDER BY totalAmountSpent DESC
    -> LIMIT 1;
+------------+-----------+----------+-------------------+
| customerID | firstName | lastName | totalAmountSpent |
+------------+-----------+----------+-------------------+
|         10 | Olivia    | Adams    |          1500.00 |
+------------+-----------+----------+-------------------+
1 row in set (0.00 sec)
```

18. List All Cars with Their Current Lease Information.

```
ysql> SELECT V.*, L.*
  -> FROM Vehicle V
  -> LEFT JOIN Lease L ON V.vehicleID = L.vehicleID
  -> ;
+-----------+----------+----------+------+-----------+-----------+---------------------+----------------+---------+-----------+------------+------------+------------+-----------+
| vehicleID | make     | model    | year | dailyRate | available | passengerCapacity | engineCapacity | leaseID | vehicleID | customerID | startDate  | endDate    | Leasetype |
+-----------+----------+----------+------+-----------+-----------+---------------------+----------------+---------+-----------+------------+------------+------------+-----------+
|         1 | Toyota   | Camry    | 2022 |     50.00 | 1         |                   4 |           1450 |       1 |         1 |          1 | 2023-01-01 | 2023-01-05 | Daily     |
|         2 | Honda    | Civic    | 2023 |     45.00 | 1         |                   7 |           1500 |       2 |         2 |          2 | 2023-02-15 | 2023-02-28 | Monthly   |
|         3 | Ford     | Focus    | 2022 |     48.00 | 0         |                   4 |           1400 |       3 |         3 |          3 | 2023-03-10 | 2023-03-15 | Daily     |
|         3 | Ford     | Focus    | 2022 |     48.00 | 0         |                   4 |           1400 |       9 |         3 |          3 | 2023-09-07 | 2023-09-10 | Daily     |
|         4 | Nissan   | Altima   | 2023 |     52.00 | 1         |                   7 |           1200 |       4 |         4 |          4 | 2023-04-20 | 2023-04-30 | Monthly   |
|         4 | Nissan   | Altima   | 2023 |     52.00 | 1         |                   7 |           1200 |       6 |         4 |          3 | 2023-06-15 | 2023-06-30 | Monthly   |
|         5 | Chevrolet | Malibu  | 2022 |     47.00 | 1         |                   4 |           1800 |    NULL |      NULL |       NULL | NULL       | NULL       | NULL      |
|         6 | Hyundai  | Sonata   | 2023 |     49.00 | 0         |                   7 |           1400 |    NULL |      NULL |       NULL | NULL       | NULL       | NULL      |
|         7 | BMW      | 3 Series | 2023 |     60.00 | 1         |                   7 |           2499 |       7 |         7 |          7 | 2023-07-01 | 2023-07-10 | Daily     |
|         8 | Mercedes | C-Class  | 2022 |     68.00 | 1         |                   8 |           2599 |       8 |         8 |          8 | 2023-08-12 | 2023-08-15 | Monthly   |
|         9 | Audi     | A4       | 2022 |     55.00 | 0         |                   4 |           2500 |    NULL |      NULL |       NULL | NULL       | NULL       | NULL      |
|        10 | Lexus    | ES       | 2023 |     54.00 | 1         |                   4 |           2500 |      10 |        10 |         10 | 2023-10-10 | 2023-10-31 | Monthly   |
+-----------+----------+----------+------+-----------+-----------+---------------------+----------------+---------+-----------+------------+------------+------------+-----------+
2 rows in set (0.00 sec)
```