# Coding Challenge

Creating Database:

```
mysql> create database crs1;
Query OK, 1 row affected (0.01 sec)

mysql> use crs1
Database changed
```

Creating tables:

```
mysql> CREATE TABLE Vehicle (
    ->     vehicleID INT PRIMARY KEY,
    ->     make VARCHAR(255),
    ->     model VARCHAR(255),
    ->     year INT,
    ->     dailyRate DECIMAL(10, 2),
    ->     status BIT,
    ->     passengerCapacity INT,
    ->     engineCapacity INT
    -> );
Query OK, 0 rows affected (0.04 sec)

mysql>
mysql> -- Customer Table
mysql> CREATE TABLE Customer (
    ->     customerID INT PRIMARY KEY,
    ->     firstName VARCHAR(255),
    ->     lastName VARCHAR(255),
    ->     email VARCHAR(255),
    ->     phoneNumber VARCHAR(20)
    -> );
ERROR 1050 (42S01): Table 'customer' already exists
mysql>
mysql> -- Lease Table
mysql> CREATE TABLE Lease (
    ->     leaseID INT PRIMARY KEY,
    ->     vehicleID INT,
    ->     customerID INT,
    ->     startDate DATE,
    ->     endDate DATE,
    ->     type VARCHAR(20) CHECK (type IN ('DailyLease', 'MonthlyLease')),
    ->     FOREIGN KEY (vehicleID) REFERENCES Vehicle(vehicleID),
    ->     FOREIGN KEY (customerID) REFERENCES Customer(customerID)
    -> );
Query OK, 0 rows affected (0.07 sec)
```

```
mysql> -- 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 (0.02 sec)
```

Inserting values:

```
mysql> INSERT INTO Vehicle (vehicleID, make, model, year, dailyRate, status, passengerC
apacity, engineCapacity)
    -> VALUES
    -> (1, 'Toyota', 'Camry', 2022, 50, 1, 4, 1450),
    -> (2, 'Honda', 'Civic', 2023, 45, 1, 7, 1500),
    -> (3, 'Ford', 'Focus', 2022, 48, 0, 4, 1400),
    -> (4, 'Nissan', 'Altima', 2023, 52, 1, 7, 1200),
    -> (5, 'Chevrolet', 'Malibu', 2022, 47, 1, 4, 1800),
    -> (6, 'Hyundai', 'Sonata', 2023, 49, 0, 7, 1400),
    -> (7, 'BMW', '3 Series', 2023, 60, 1, 7, 2499),
    -> (8, 'Mercedes', 'C - Class', 2022, 58, 1, 8, 2599),
    -> (9, 'Audi', 'A4', 2022, 55, 0, 4, 2500),
    -> (10, 'Lexus', 'ES', 2023, 54, 1, 4, 2500);
Query OK, 10 rows affected (0.01 sec)
Records: 10  Duplicates: 0  Warnings: 0

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 (0.00 sec)
Records: 10  Duplicates: 0  Warnings: 0

mysql> INSERT INTO Lease (leaseID, vehicleID, customerID, startDate, endDate, type)
    -> VALUES
    -> (1, 1, 1, '2023-01-01', '2023-05-01', '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> 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 (0.01 sec)
Records: 10  Duplicates: 0  Warnings: 0
```

Tasks :

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
```

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, 2 rows affected (0.01 sec)
```

```
mysql>
mysql> DELETE FROM Lease
    -> WHERE customerID = 3;
Query OK, 2 rows affected (0.00 sec)
```

```
mysql>
mysql> DELETE FROM Customer
    -> WHERE customerID = 3;
Query OK, 1 row affected (0.00 sec)
```

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

```
mysql> ALTER TABLE Payment
    -> RENAME COLUMN paymentDate TO transactionDate;
Query OK, 0 rows affected (0.02 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

4. Find a specific customer by email.

```
mysql> SELECT *
    -> FROM Customer
    -> WHERE email = 'janesmith@example.com';
+------------+-----------+----------+----------------------+--------------+
| customerID | firstName | lastName | email                | phoneNumber  |
+------------+-----------+----------+----------------------+--------------+
|          2 | Jane      | Smith    | janesmith@example.com | 555-123-4567 |
+------------+-----------+----------+----------------------+--------------+
1 row in set (0.00 sec)
```

5. Get active leases for a specific customer.

```
mysql> SELECT *
    -> FROM Lease
    -> WHERE customerID = 1
    ->   AND endDate <= CURDATE();
+---------+-----------+------------+------------+------------+------------+
| leaseID | vehicleID | customerID | startDate  | endDate    | type       |
+---------+-----------+------------+------------+------------+------------+
|       1 |         1 |          1 | 2023-01-01 | 2023-05-01 | DailyLease |
+---------+-----------+------------+------------+------------+------------+
1 row in 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 = '555-555-5555';
+-----------+---------+-----------------+--------+-----------+----------+
| paymentID | leaseID | transactionDate | amount | firstName | lastName |
+-----------+---------+-----------------+--------+-----------+----------+
|         1 |       1 | 2023-01-03      | 200.00 | John      | Doe      |
+-----------+---------+-----------------+--------+-----------+----------+
1 row in set (0.00 sec)
```

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

```
mysql> SELECT AVG(dailyRate) AS avgDailyRate
    -> FROM Vehicle
    -> WHERE status = 'available';
+--------------+
| avgDailyRate |
+--------------+
|    50.666667 |
+--------------+
```

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 |
+-----------+------+----------+------+-----------+---------------+-------------------+
----------------+
|         7 | BMW  | 3 Series | 2023 |     60.00 | 0x01          |                 7 |
          2499 |
+-----------+------+----------+------+-----------+---------------+-------------------+
----------------+
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
    -> WHERE Lease.customerID = 2;
+-----------+-------+-------+------+-----------+---------------+-------------------+--
--------------+
| vehicleID | make  | model | year | dailyRate | status        | passengerCapacity | e
ngineCapacity |
+-----------+-------+-------+------+-----------+---------------+-------------------+--
--------------+
|         2 | Honda | Civic | 2023 |     45.00 | 0x01          |                 7 |
         1500 |
+-----------+-------+-------+------+-----------+---------------+-------------------+--
--------------+
1 row in set (0.00 sec)
```

10. Find the details of the most recent lease:

```
mysql> SELECT Lease.*, Vehicle.make, Vehicle.model
    -> FROM Lease
    -> JOIN Vehicle ON Lease.vehicleID = Vehicle.vehicleID
    -> ORDER BY startDate DESC
    -> LIMIT 1;
+---------+-----------+------------+-----------+-----------+-------------+-------+--
------+
| leaseID | vehicleID | customerID | startDate  | endDate    | type         | make  | m
odel |
+---------+-----------+------------+-----------+-----------+-------------+-------+--
------+
|      10 |        10 |         10 | 2023-10-10 | 2023-10-31 | MonthlyLease | Lexus | E
S    |
+---------+-----------+------------+-----------+-----------+-------------+-------+--
------+
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 |
|         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 |
|         9 |       9 | 2023-09-09      |   80.00 |
|        10 |      10 | 2023-10-25      | 1500.00 |
+-----------+---------+-----------------+---------+
8 rows in set (0.00 sec)
```

12. Retrieve customers who have not made any payments:

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

13. Retrieve Car Details and Their Total Payments:

```
mysql> SELECT Vehicle.*, COALESCE(SUM(Payment.amount), 0) AS totalPayments
    -> FROM Vehicle
    -> LEFT JOIN Lease ON Vehicle.vehicleID = Lease.vehicleID
    -> LEFT JOIN Payment ON Lease.leaseID = Payment.leaseID
    -> GROUP BY Vehicle.vehicleID;
+-----------+-----------+-----------+------+-----------+----------------+--------------
-----+---------------+--------------+
| vehicleID | make      | model     | year | dailyRate | status         | passengerCapa
city | engineCapacity | totalPayments |
+-----------+-----------+-----------+------+-----------+----------------+--------------
-----+---------------+--------------+
|         1 | Toyota    | Camry     | 2022 |     50.00 | 0x01           |
   4 |           1450 |        200.00 |
|         2 | Honda     | Civic     | 2023 |     45.00 | 0x01           |
   7 |           1500 |       1000.00 |
|         3 | Ford      | Focus     | 2022 |     48.00 | 0x00           |
   4 |           1400 |          0.00 |
|         4 | Nissan    | Altima    | 2023 |     52.00 | 0x01           |
   7 |           1200 |        900.00 |
|         5 | Chevrolet | Malibu    | 2022 |     47.00 | 0x01           |
   4 |           1800 |         60.00 |
|         6 | Hyundai   | Sonata    | 2023 |     49.00 | 0x00           |
   7 |           1400 |          0.00 |
|         7 | BMW       | 3 Series  | 2023 |     60.00 | 0x01           |
   7 |           2499 |         40.00 |
|         8 | Mercedes  | C - Class | 2022 |     58.00 | 0x01           |
   8 |           2599 |       1100.00 |
|         9 | Audi      | A4        | 2022 |     55.00 | 0x00           |
   4 |           2500 |         80.00 |
|        10 | Lexus     | ES        | 2023 |     54.00 | 0x01           |
   4 |           2500 |       1500.00 |
+-----------+-----------+-----------+------+-----------+----------------+--------------
-----+---------------+--------------+
10 rows in set (0.00 sec)
```

14. Calculate Total Payments for Each Customer:

```
mysql> SELECT Customer.*, COALESCE(SUM(Payment.amount), 0) AS totalPayments
    -> FROM Customer
    -> LEFT JOIN Lease ON Customer.customerID = Lease.customerID
    -> LEFT JOIN Payment ON Lease.leaseID = Payment.leaseID
    -> GROUP BY Customer.customerID;
+------------+-----------+----------+----------------------+-------------+-----------
----+
| customerID | firstName | lastName | email                | phoneNumber  | totalPayme
nts |
+------------+-----------+----------+----------------------+-------------+-----------
----+
|          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 |          0
.00 |
|          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 |         80
.00 |
|         10 | Olivia    | Adams    | olivia@example.com    | 555-765-4321 |       1500
.00 |
+------------+-----------+----------+----------------------+-------------+-----------
----+
```

15. List Car Details for Each Lease

```
mysql> SELECT Lease.*, Vehicle.make, Vehicle.model
    -> FROM Lease
    -> JOIN Vehicle ON Lease.vehicleID = Vehicle.vehicleID;
+---------+-----------+------------+------------+------------+--------------+----------
-+------------+
| leaseID | vehicleID | customerID | startDate  | endDate    | type         | make
 | model      |
+---------+-----------+------------+------------+------------+--------------+----------
-+------------+
|       1 |         1 |          1 | 2023-01-01 | 2023-05-01 | DailyLease   | Toyota
 | Camry      |
|       2 |         2 |          2 | 2023-02-15 | 2023-02-28 | MonthlyLease | Honda
 | Civic      |
|       4 |         4 |          4 | 2023-04-20 | 2023-04-30 | MonthlyLease | Nissan
 | Altima     |
|       5 |         5 |          5 | 2023-05-05 | 2023-05-10 | DailyLease   | Chevrolet
 | Malibu     |
|       7 |         7 |          7 | 2023-07-01 | 2023-07-10 | DailyLease   | BMW
 | 3 Series   |
|       8 |         8 |          8 | 2023-08-12 | 2023-08-15 | MonthlyLease | Mercedes
 | C - Class  |
|       9 |         9 |          9 | 2023-09-07 | 2023-09-10 | DailyLease   | Audi
 | A4         |
|      10 |        10 |         10 | 2023-10-10 | 2023-10-31 | MonthlyLease | Lexus
 | ES         |
+---------+-----------+------------+------------+------------+--------------+----------
-+------------+
8 rows in set (0.00 sec)
```

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

```
mysql> SELECT Lease.*, Customer.firstName, Customer.lastName, Vehicle.make, Vehicle.mod
el
    -> FROM Lease
    -> JOIN Customer ON Lease.customerID = Customer.customerID
    -> JOIN Vehicle ON Lease.vehicleID = Vehicle.vehicleID
    -> WHERE endDate >= CURDATE();
Empty set (0.00 sec)
```

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

```
mysql> SELECT Customer.*, COALESCE(SUM(Payment.amount), 0) AS totalPayments
    -> FROM Customer
    -> LEFT JOIN Lease ON Customer.customerID = Lease.customerID
    -> LEFT JOIN Payment ON Lease.leaseID = Payment.leaseID
    -> GROUP BY Customer.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 (0.00 sec)
```

18. List All Cars with Their Current Lease Information:

```
mysql> SELECT Vehicle.*, Lease.startDate, Lease.endDate, Customer.firstName, Customer.lastName
    -> FROM Vehicle
    -> LEFT JOIN Lease ON Vehicle.vehicleID = Lease.vehicleID
    -> LEFT JOIN Customer ON Lease.customerID = Customer.customerID;
+-----------+----------+----------+------+-------------+--------+-------------------+----------------+------------+------------+-----------+----------+
| vehicleID | make     | model    | year | dailyRate   | status | passengerCapacity | engineCapacity | startDate  | endDate    | firstName | lastName |
+-----------+----------+----------+------+-------------+--------+-------------------+----------------+------------+------------+-----------+----------+
|         1 | Toyota   | Camry    | 2022 |     50.00 | 0x01   |                 4 |           1450 | 2023-01-01 | 2023-05-01 | John      | Doe      |
|         2 | Honda    | Civic    | 2023 |     45.00 | 0x01   |                 7 |           1500 | 2023-02-15 | 2023-02-28 | Jane      | Smith    |
|         3 | Ford     | Focus    | 2022 |     48.00 | 0x00   |                 4 |           1400 | NULL       | NULL       | NULL      | NULL     |
|         4 | Nissan   | Altima   | 2023 |     52.00 | 0x01   |                 7 |           1200 | 2023-04-20 | 2023-04-30 | Sarah     | Brown    |
|         5 | Chevrolet | Malibu   | 2022 |     47.00 | 0x01   |                 4 |           1800 | 2023-05-05 | 2023-05-10 | David     | Lee      |
|         6 | Hyundai  | Sonata   | 2023 |     49.00 | 0x00   |                 7 |           1400 | NULL       | NULL       | NULL      | NULL     |
|         7 | BMW      | 3 Series | 2023 |     60.00 | 0x01   |                 7 |           2499 | 2023-07-01 | 2023-07-10 | Michael   | Davis    |
|         8 | Mercedes | C - Class | 2022 |     68.00 | 0x01   |                 8 |           2599 | 2023-08-12 | 2023-08-15 | Emma      | Wilson   |
|         9 | Audi     | A4       | 2022 |     55.00 | 0x00   |                 4 |           2500 | 2023-09-07 | 2023-09-10 | William   | Taylor   |
|        10 | Lexus    | ES       | 2023 |     54.00 | 0x01   |                 4 |           2500 | 2023-10-10 | 2023-10-31 | Olivia    | Adams    |
+-----------+----------+----------+------+-------------+--------+-------------------+----------------+------------+------------+-----------+----------+
10 rows in set (0.01 sec)
```

```
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
```