Coding Challenge - Car Rental System – SQL

Instructions

- Coding Challenge submissions should be done through the partcipants' Github repository, and the link should be shared with trainers and Hexavarsity.

SQL Schema:

1. **Vehicle Table:**
   - vehicleID (Primary Key)
   - make
   - model
   - year
   - dailyRate
   - status (available, notAvailable)
   - passengerCapacity
   - engineCapacity
2. **Customer Table:**
   - customerID (Primary Key)
   - firstName
   - lastName
   - email
   - phoneNumber
3. **Lease Table:**
   - leaseID (Primary Key)
   - vehicleID (Foreign Key referencing Vehicle Table)
   - customerID (Foreign Key referencing Customer Table)
   - startDate
   - endDate
   - type (to distinguish between DailyLease and MonthlyLease)
4. **Payment Table:**
   - paymentID (Primary Key)
   - leaseID (Foreign Key referencing Lease Table)
   - paymentDate
   - amount

Vehicle Table

- type (to distinguish between DailyLease and MonthlyLease)

4. **Payment Table:**
   - paymentID (Primary Key)
   - leaseID (Foreign Key referencing Lease Table)
   - paymentDate
   - amount

Vehicle Table

carID	make	model	Year	dailyRate	available	passenger Capacity	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

© Hexaware Technologies Limited. All rights

carID	make	model	Year	dailyRate	available	passenger Capacity	engineCapacity
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

Customer Table

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

1. Update the daily rate for a Mercedes car to 68.
2. Delete a specific customer and all associated leases and payments.
3. Rename the "paymentDate" column in the Payment table to "transactionDate".
4. Find a specific customer by email.
5. Get active leases for a specific customer.
6. Find all payments made by a customer with a specific phone number.
7. Calculate the average daily rate of all available cars.
8. Find the car with the highest daily rate.
9. Retrieve all cars leased by a specific customer.
10. Find the details of the most recent lease.
11. List all payments made in the year 2023.
12. Retrieve customers who have not made any payments.
13. Retrieve Car Details and Their Total Payments.
14. Calculate Total Payments for Each Customer.
15. List Car Details for Each Lease.
16. Retrieve Details of Active Leases with Customer and Car Information.
17. Find the Customer Who Has Spent the Most on Leases.
18. List All Cars with Their Current Lease Information.

Lease Table

leaseID	carID	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

Payment Table

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

© Hexaware Technologies Limited. All rights

paymentID	leaseID	paymentDate	amount
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 8.0 Command Line Client

```
Query OK, 0 rows affected (3.17 sec)

mysql> INSERT INTO Vehicle (vehicleID, make, model, year, dailyRate, available, passengerCapacity, engineCapacity)
    -> VALUES
    ->     (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, 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 (0.56 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 |                 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 |     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 (0.04 sec)

mysql>
```

Type here to search
22°C Haze
ENG
20:51
10-12-2023

MySQL 8.0 Command Line Client

```
    -> WHERE make = 'Mercedes';
Query OK, 1 row affected (0.32 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 (0.02 sec)

mysql> ALTER TABLE Payment
    -> RENAME COLUMN paymentDate TO transactionDate;
Query OK, 0 rows affected (2.09 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 |
|         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>
```

Type here to search
26°C Mostly cloudy
ENG
21:42
10-12-2023

MySQL 8.0 Command Line Client

```
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 (2.07 sec)

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 (0.28 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 (0.03 sec)

mysql>
```

Type here to search
26°C Mostly cloudy
ENG
21:20
10-12-2023

MySQL 8.0 Command Line Client

```
10 rows in set (0.00 sec)

mysql> CREATE TABLE Lease (
    ->     leaseID INT PRIMARY KEY,
    ->     vehicleID INT,
    ->     customerID INT,
    ->     startDate DATE,
    ->     endDate DATE,
    ->     leaseType VARCHAR(20) CHECK (leaseType IN ('Daily', 'Monthly')),
    ->     FOREIGN KEY (vehicleID) REFERENCES Vehicle(vehicleID),
    ->     FOREIGN KEY (customerID) REFERENCES Customer(customerID)
    -> );
Query OK, 0 rows affected (2.18 sec)

mysql> 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');
Query OK, 10 rows affected (0.28 sec)
Records: 10  Duplicates: 0  Warnings: 0

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   |
+---------+-----------+------------+------------+------------+-----------+
```

Type here to search
26°C Mostly cloudy
ENG
21:13
10-12-2023

MySQL 8.0 Command Line Client

```
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 (1.65 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, 'Oliva', 'Adams', 'oliva@example.com', '555-765-4321');
Query OK, 10 rows affected (0.43 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 | Oliva     | Adams    | oliva@example.com    | 555-765-4321 |
+------------+-----------+----------+----------------------+-------------+
10 rows in set (0.00 sec)

mysql>
```

Type here to search
22°C Haze
ENG
21:03
10-12-2023

MySQL 8.0 Command Line Client

```
|         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> SELECT AVG(dailyRate) AS averageDailyRate
    -> FROM Vehicle
    -> WHERE status = 1;
ERROR 1054 (42S22): Unknown column 'status' in 'where clause'
mysql> ^C
mysql> SELECT AVG(dailyRate) AS averageDailyRate
    -> FROM Vehicle
    -> WHERE available = 1;
+------------------+
| averageDailyRate |
+------------------+
|        53.714286 |
+------------------+
1 row in set (0.07 sec)

mysql> SELECT max(dailyRate)
    -> FROM Vehicle
    -> ORDER BY dailyRate DESC;
+-----------------+
| max(dailyRate) |
+-----------------+
|          68.00 |
+-----------------+
1 row in set (0.24 sec)

mysql> SELECT TOP 1 *
    -> FROM Lease
    -> ^C
mysql> SELECT max(endDate)
    -> FROM Lease
    -> ORDER BY endDate DESC;
+--------------+
| max(endDate) |
+--------------+
| 2023-10-31   |
+--------------+
1 row in set (0.76 sec)
```

Type here to search
26°C Mostly cloudy
ENG
21:57
10-12-2023