**CODING CHALLENGE - CAR RENTAL SYSTEM – SQL**

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.**

**SOLUTION:**

DROP DATABASE IF EXISTS CarRental;

CREATE DATABASE CarRental;

USE CarRental;

CREATE TABLE Vehicle (

vehicleID INT PRIMARY KEY,

make VARCHAR(50),

model VARCHAR(50),

year YEAR,

dailyRate DECIMAL(6,2),

status VARCHAR(20),

passengerCapacity INT,

engineCapacity INT

);

CREATE TABLE Customer (

customerID INT PRIMARY KEY,

firstName VARCHAR(50),

lastName VARCHAR(50),

email VARCHAR(100),

phoneNumber VARCHAR(20)

);

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)

);

CREATE TABLE Payment (

paymentID INT PRIMARY KEY,

leaseID INT,

paymentDate DATE,

amount DECIMAL(8,2),

FOREIGN KEY (leaseID) REFERENCES Lease(leaseID)

);

INSERT INTO Vehicle 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),

(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);

INSERT INTO Customer 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');

INSERT INTO Lease 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');

INSERT INTO Payment 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);

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

UPDATE Vehicle

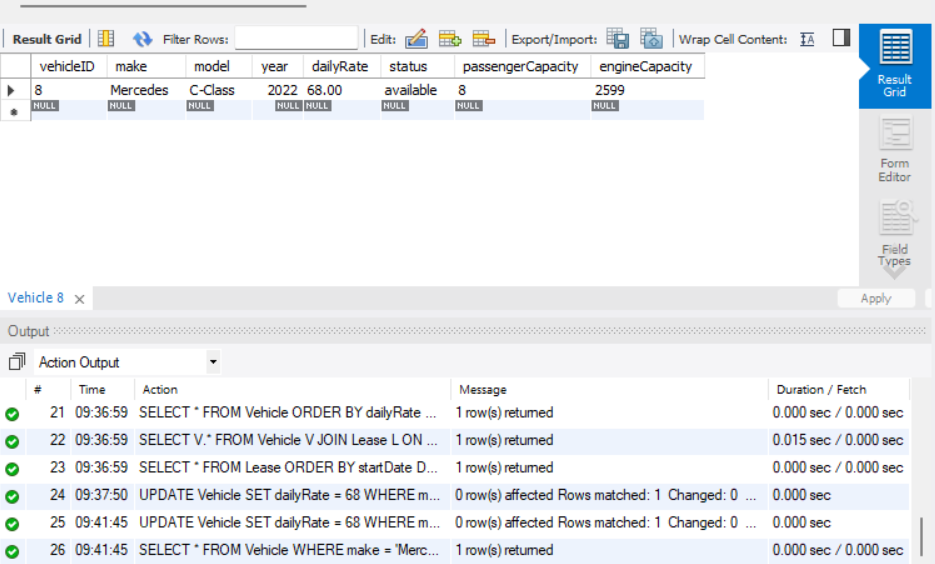
SET dailyRate = 68

WHERE make = 'Mercedes' AND model = 'C-Class';

SELECT \* FROM Vehicle

WHERE make = 'Mercedes' AND model = 'C-Class';

**OUTPUT**



1. **Delete a specific customer and all associated leases and payments.**

DELETE FROM Payment

WHERE leaseID IN (

SELECT leaseID FROM Lease WHERE customerID = 3

);

DELETE FROM Lease

WHERE customerID = 3;

DELETE FROM Customer

WHERE customerID = 3;

SELECT \* FROM Customer WHERE customerID = 3;

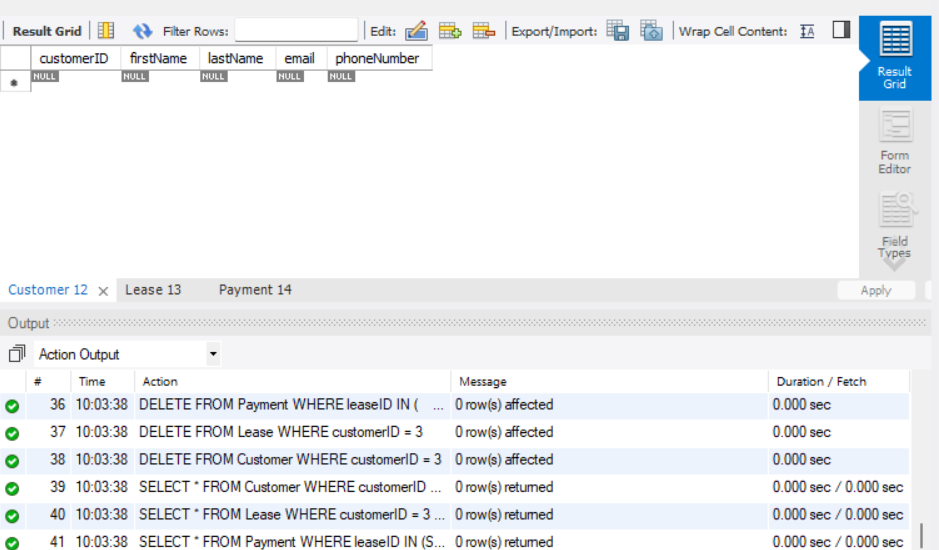
SELECT \* FROM Lease WHERE customerID = 3;

SELECT \* FROM Payment

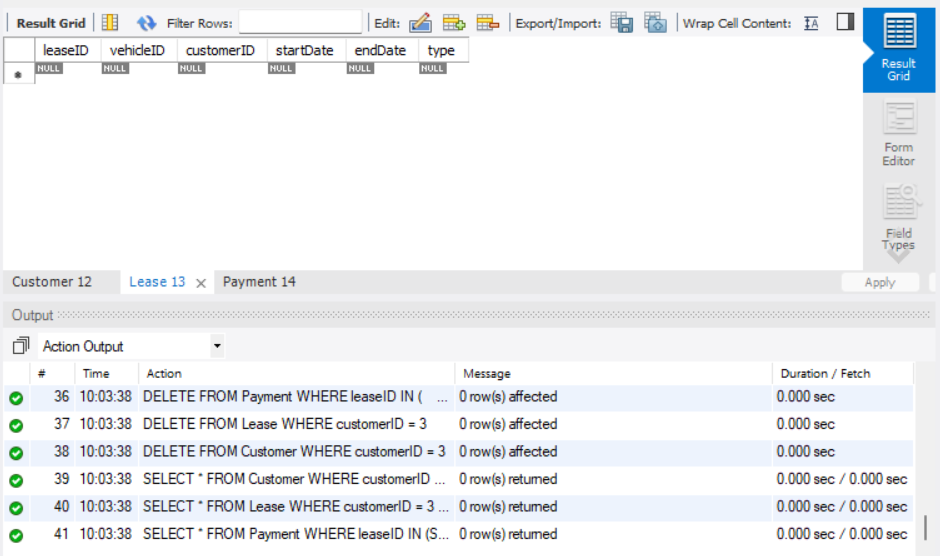
WHERE leaseID IN (SELECT leaseID FROM Lease WHERE customerID = 3);

**OUTPUT**

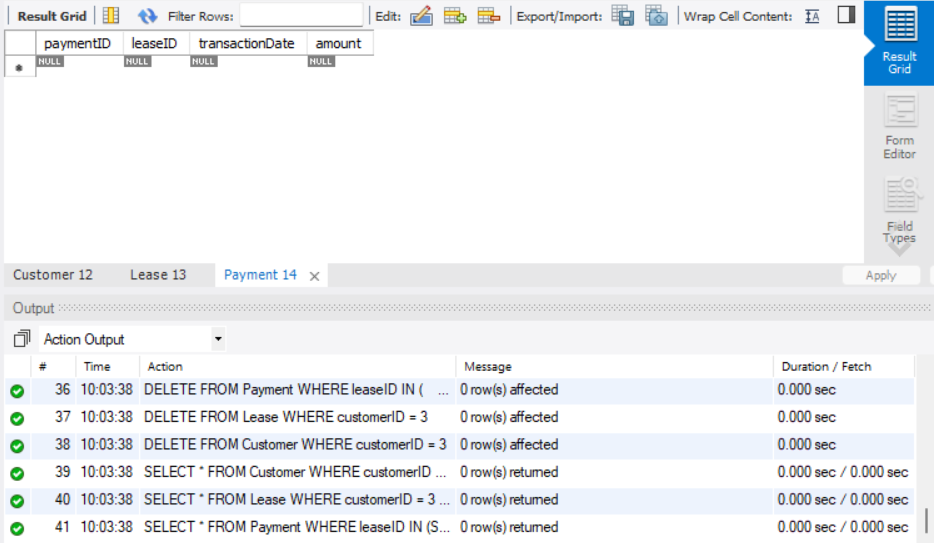
**DELETE CUSTOMER ID**

****

**DELETE LEASE ID**

****

**DELETE PAYMENT ID**

****

1. **Rename the "paymentDate" column in the Payment table to "transactionDate".**

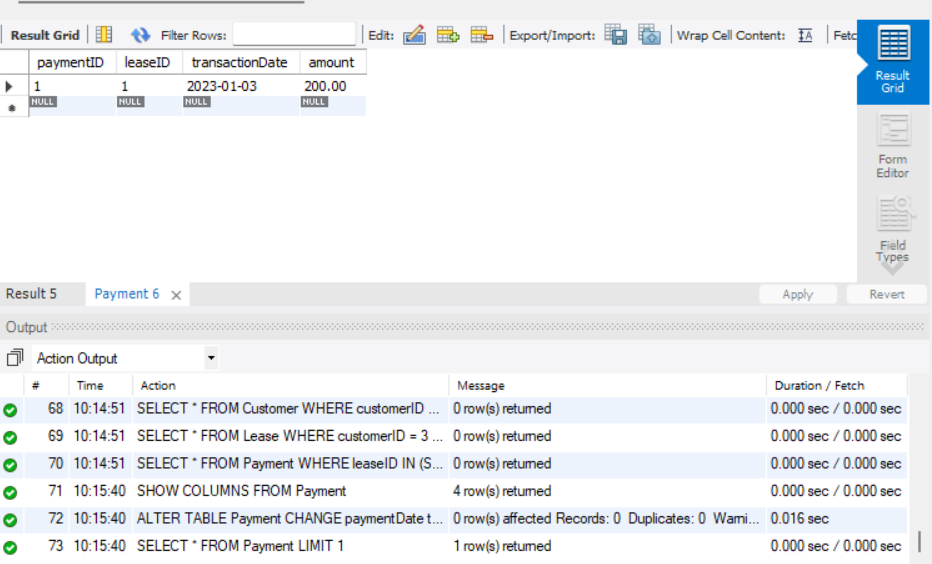
SHOW COLUMNS FROM Payment;

ALTER TABLE Payment

CHANGE paymentDate transactionDate DATE;

SELECT \* FROM Payment LIMIT 1;

**OUTPUT**

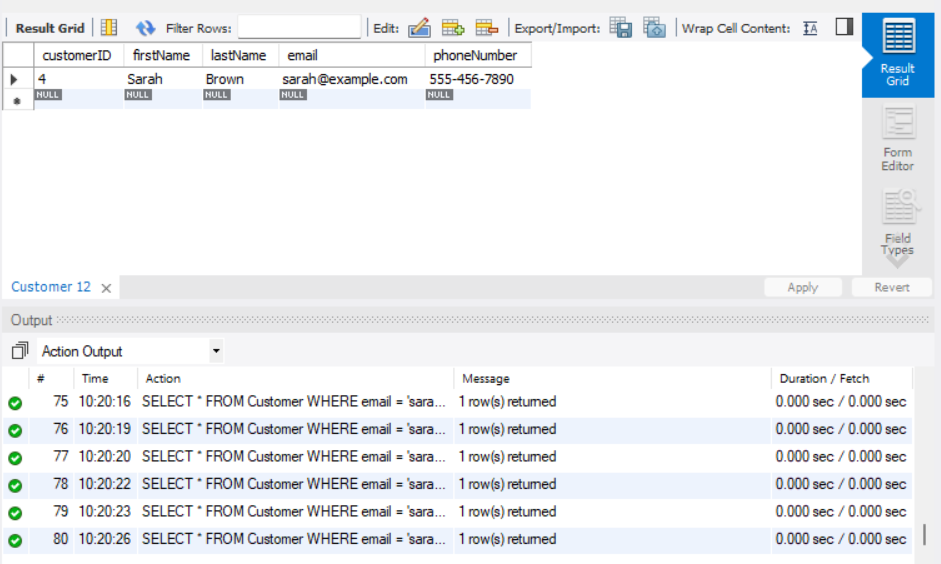
****

1. **Find a specific customer by email.**

SELECT \* FROM Customer

WHERE email = 'sarah@example.com';

**OUTPUT**

****

1. **Get active leases for a specific customer.**

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

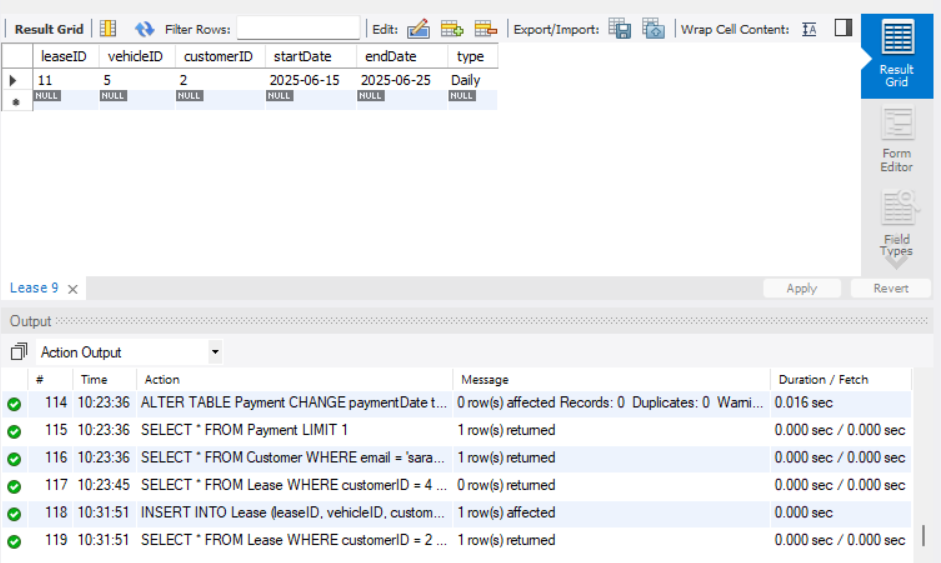
VALUES (11, 5, 2, '2025-06-15', '2025-06-25', 'Daily');

SELECT \* FROM Lease

WHERE customerID = 2

AND CURDATE() BETWEEN startDate AND endDate;

**OUTPUT**

****

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

SELECT P.\*

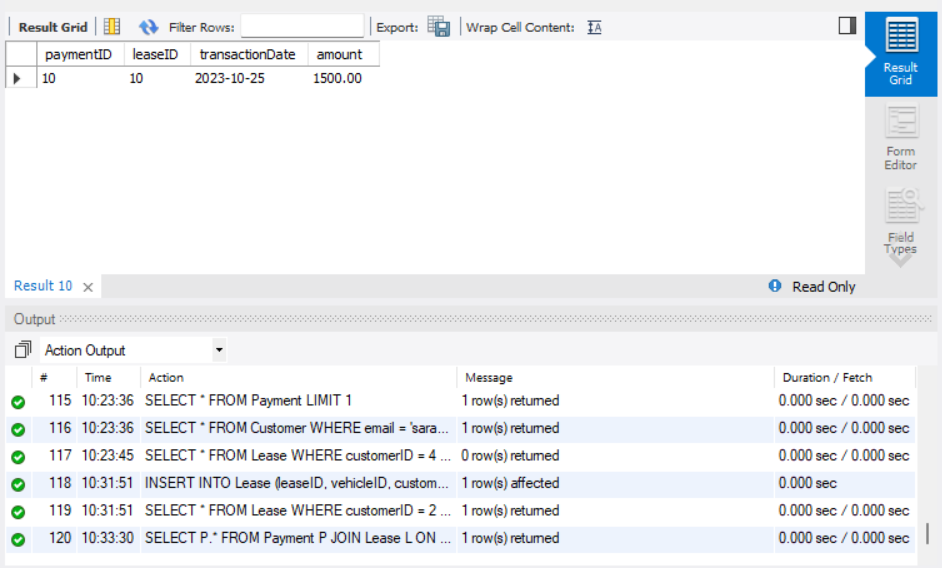
FROM Payment P

JOIN Lease L ON P.leaseID = L.leaseID

JOIN Customer C ON L.customerID = C.customerID

WHERE C.phoneNumber = '555-765-4321';

**OUTPUT**

****

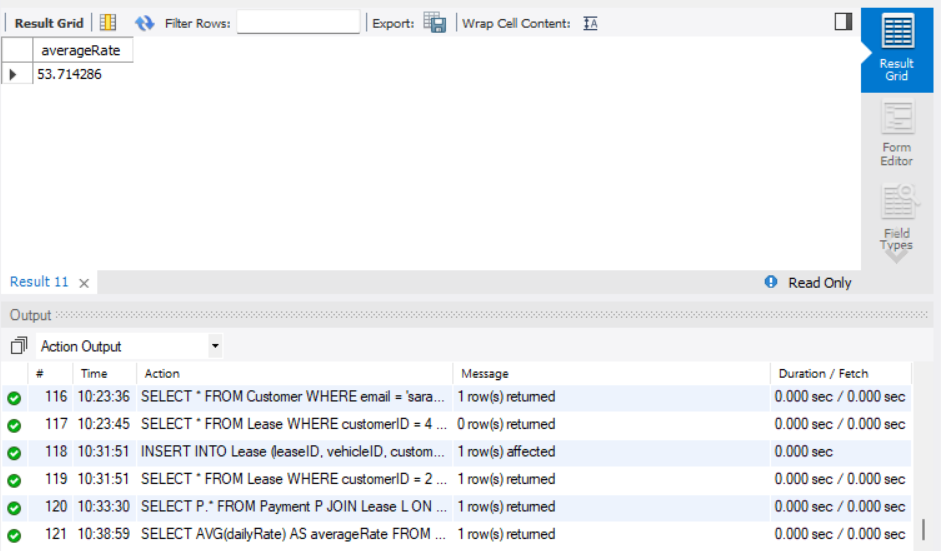
1. **Calculate the average daily rate of all available cars.**

SELECT AVG(dailyRate) AS averageRate

FROM Vehicle

WHERE status = 'available';

**OUTPUT**

****

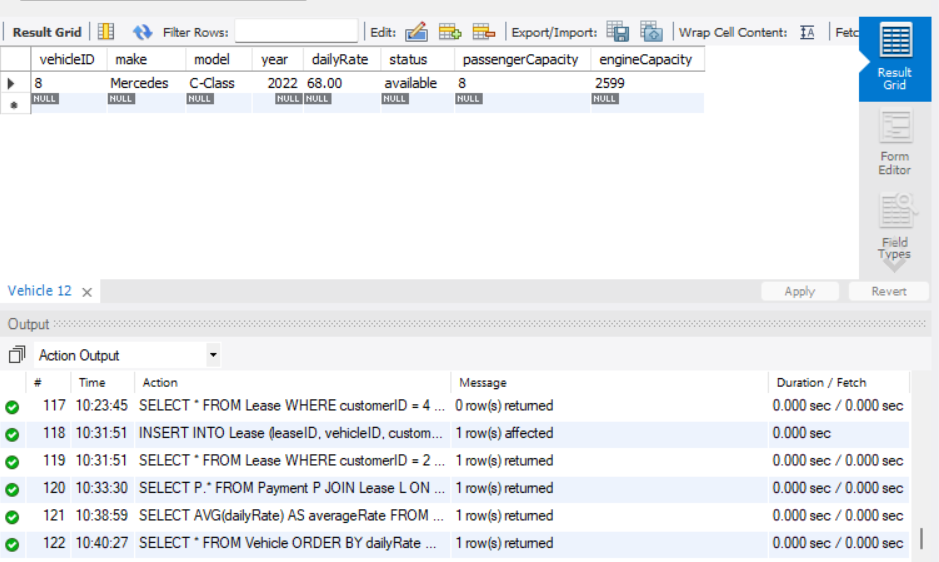
1. **Find the car with the highest daily rate.**

SELECT \* FROM Vehicle

ORDER BY dailyRate DESC

LIMIT 1;

**OUTPUT:**

****

1. **Retrieve all cars leased by a specific customer.**

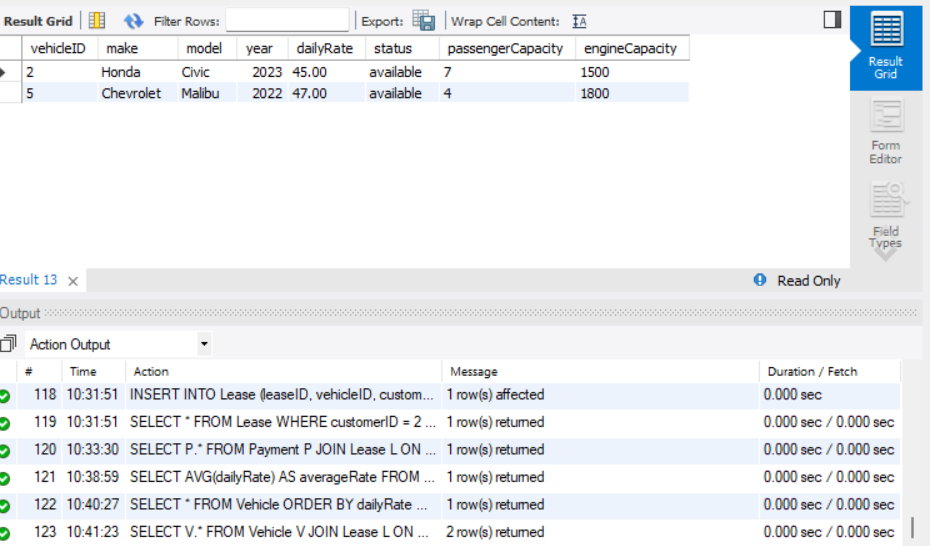
SELECT V.\*

FROM Vehicle V

JOIN Lease L ON V.vehicleID = L.vehicleID

WHERE L.customerID = 2;

**OUTPUT**

****

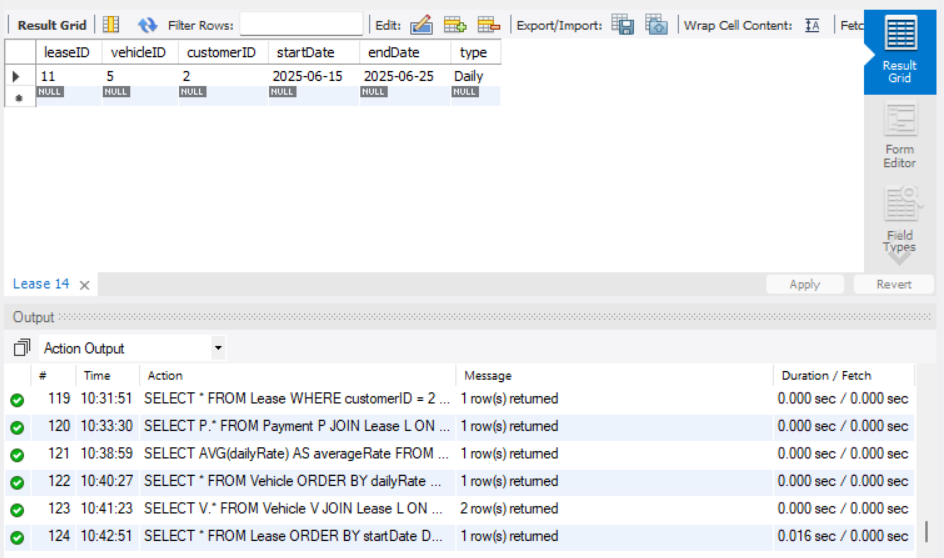
1. **Find the details of the most recent lease.**

SELECT \* FROM Lease

ORDER BY startDate DESC

LIMIT 1;

**OUTPUT**

****

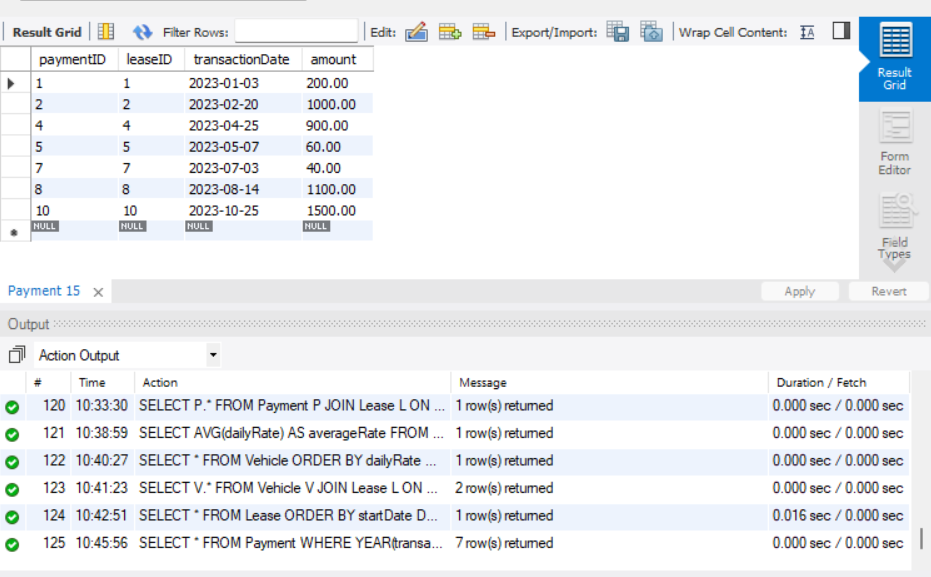
1. **List all payments made in the year 2023.**

SELECT \*

FROM Payment

WHERE YEAR(transactionDate) = 2023;

**OUTPUT**

****

1. **Retrieve customers who have not made any payments.**

SELECT \*

FROM Customer C

WHERE C.customerID NOT IN (

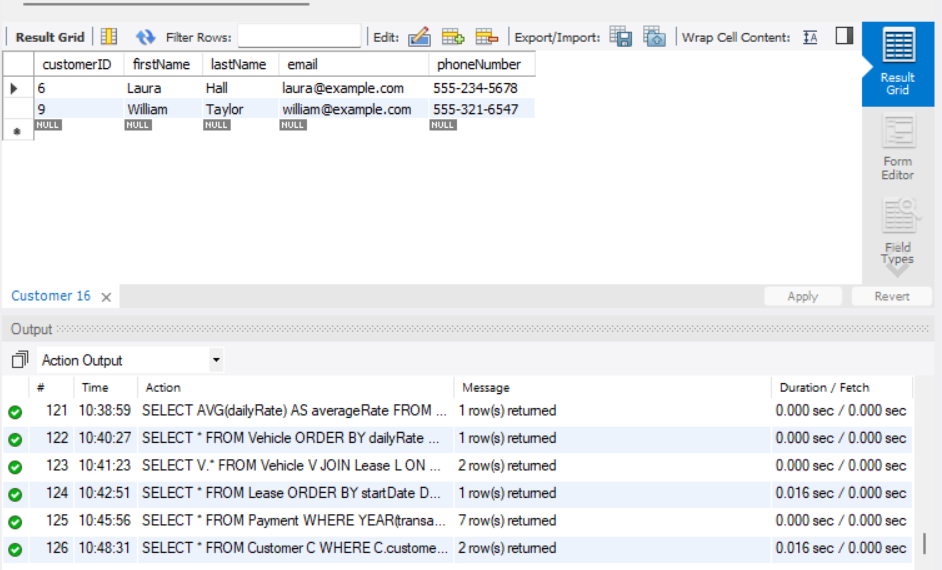
SELECT DISTINCT L.customerID

FROM Lease L

JOIN Payment P ON L.leaseID = P.leaseID

);

**OUTPUT**

****

1. **Retrieve Car Details and Their Total Payments.**

SELECT V.vehicleID, V.make, V.model, SUM(P.amount) AS total\_payment

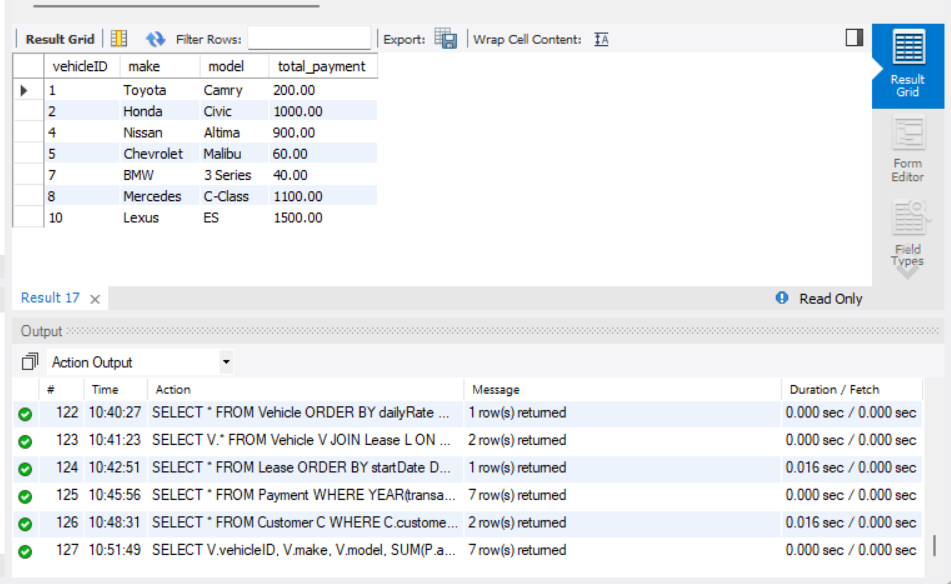
FROM Vehicle V

JOIN Lease L ON V.vehicleID = L.vehicleID

JOIN Payment P ON L.leaseID = P.leaseID

GROUP BY V.vehicleID, V.make, V.model;

**OUTPUT**

****

1. **Calculate Total Payments for Each Customer.**

SELECT C.customerID, C.firstName, C.lastName, SUM(P.amount) AS total\_paid

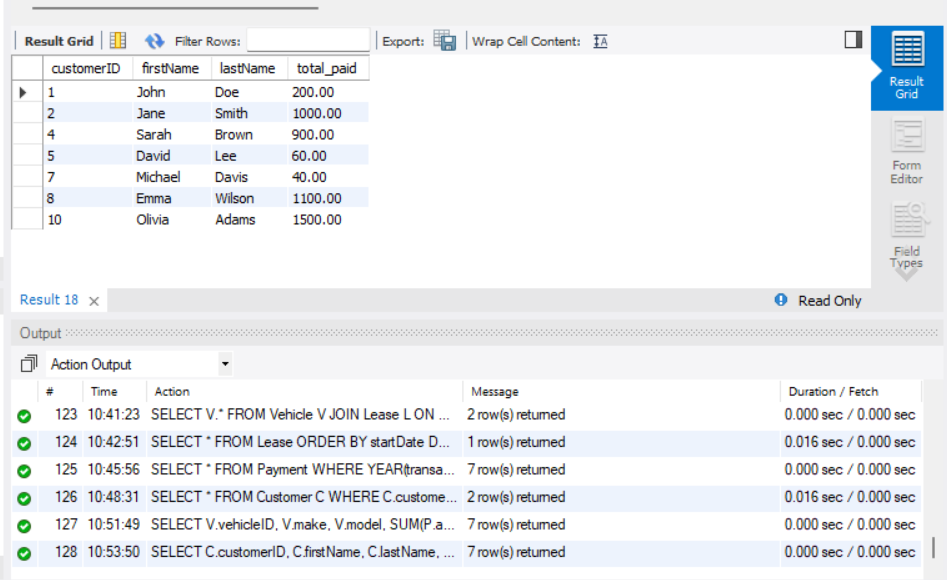
FROM Customer C

JOIN Lease L ON C.customerID = L.customerID

JOIN Payment P ON L.leaseID = P.leaseID

GROUP BY C.customerID, C.firstName, C.lastName;

**OUTPUT**

****

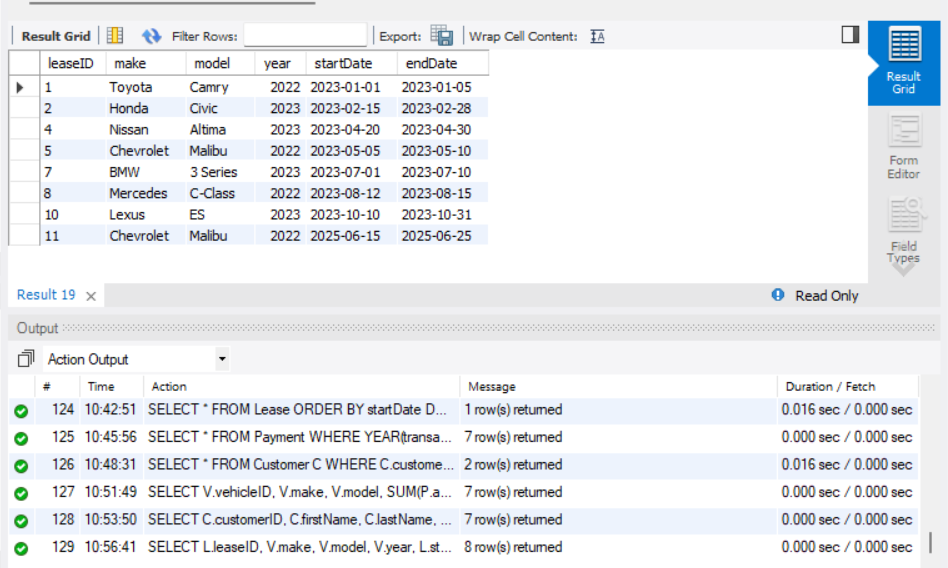
1. **List Car Details for Each Lease.**

SELECT L.leaseID, V.make, V.model, V.year, L.startDate, L.endDate

FROM Lease L

JOIN Vehicle V ON L.vehicleID = V.vehicleID;

**OUTPUT**

****

1. **Retrieve Details of Active Leases with Customer and Car Information.**

SELECT L.\*, C.firstName, C.lastName, V.make, V.model

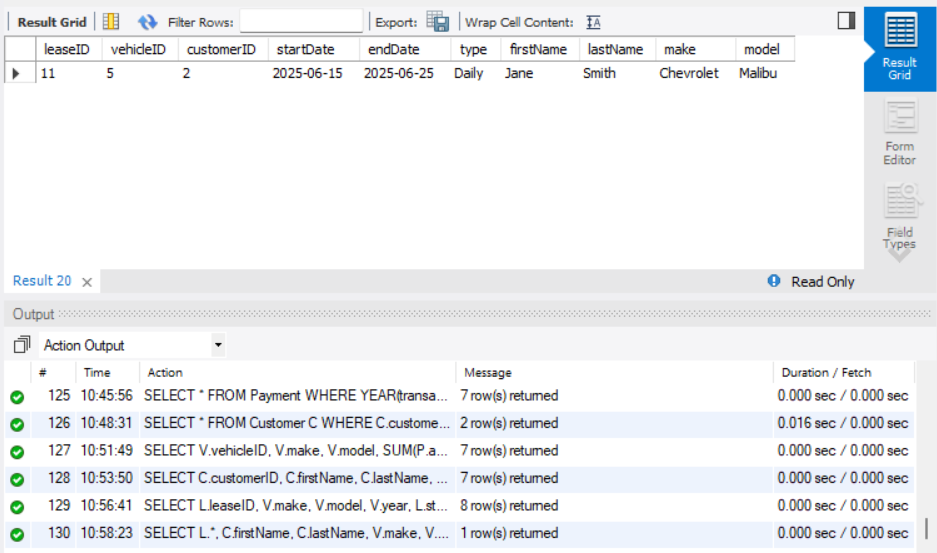
FROM Lease L

JOIN Customer C ON L.customerID = C.customerID

JOIN Vehicle V ON L.vehicleID = V.vehicleID

WHERE CURDATE() BETWEEN L.startDate AND L.endDate;

**OUTPUT**



1. **Find the Customer Who Has Spent the Most on Leases.**

SELECT C.customerID, C.firstName, C.lastName, SUM(P.amount) AS total\_spent

FROM Customer C

JOIN Lease L ON C.customerID = L.customerID

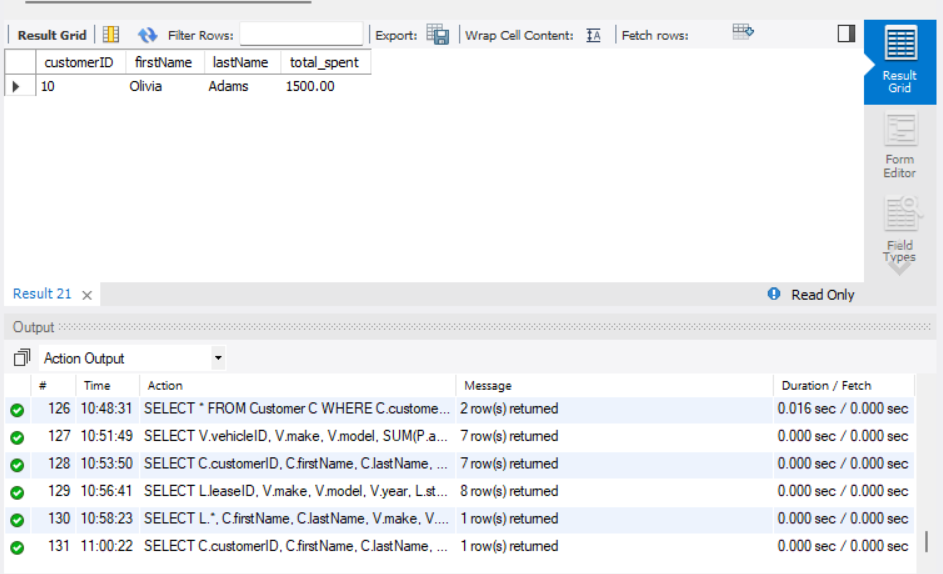
JOIN Payment P ON L.leaseID = P.leaseID

GROUP BY C.customerID, C.firstName, C.lastName

ORDER BY total\_spent DESC

LIMIT 1;

**OUTPUT**

****

1. **List All Cars with Their Current Lease Information.**

SELECT V.vehicleID, V.make, V.model, L.startDate, L.endDate, C.firstName, C.lastName

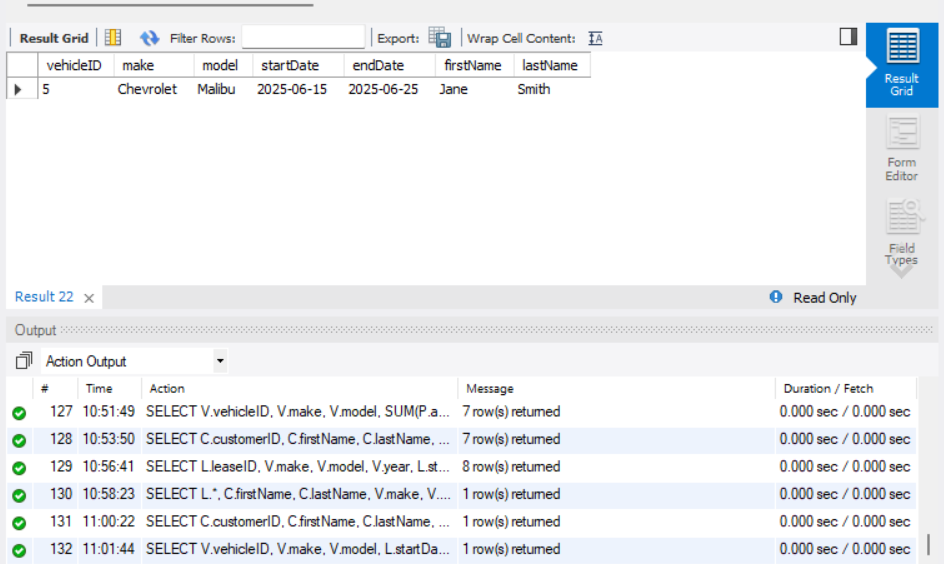
FROM Vehicle V

LEFT JOIN Lease L ON V.vehicleID = L.vehicleID

LEFT JOIN Customer C ON L.customerID = C.customerID

WHERE CURDATE() BETWEEN L.startDate AND L.endDate;

**OUTPUT**

****