SHUBHI TARAN

CODING CHALLENGE NO 4:

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

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

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

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

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

QUERIES:

CREATING DATABASE + TABLES AND INSERTING VALUES:

-- coding challenge file 4 shubhi taran

```
CREATE DATABASE CarRental;
Use CarRental;
create table vehicle(
vehicleID int primary key,
make varchar(50),
model varchar(50),
year int,
dailyrate decimal(10,2),
status varchar(50),
passengercapacity int,
enginecapacity int
);

create table customer (
customerID int primary key,
firstname varchar(50),
lastname varchar (50),
email varchar(50),
phonenumber varchar(50)
);
```

```
create table lease (
leaseID int primary key,
carID int,
customerID int,
startdate date,
enddate date,
leasetype varchar(20),
foreign key (carID) references vehicle(vehicleID),
foreign key (customerID) references customer(customerID)
);

create table payment (
paymentID int primary key,
leaseID int,
paymentdate date,
amount decimal(10,2),
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),

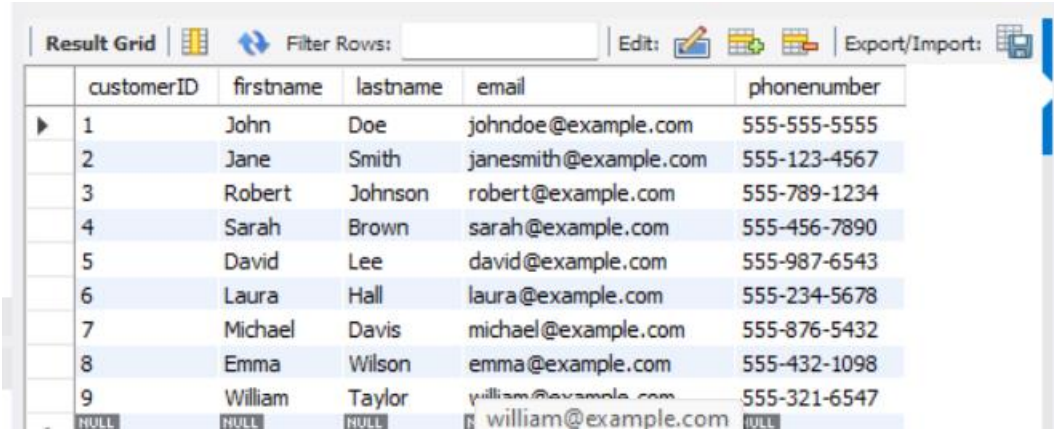
(10, 10, '2023-10-25', 1500.00);

DISPLAYING TABLES: VEHICLE TABLE

Result Grid | Filter Rows: | Edit: | Export/Import:

vehicleID	make	model	year	dailyrate	status	passengercapacity	en
1	Toyota	Camry	2022	50.00	available	4	145
2	Honda	Civic	2023	45.00	available	7	150
3	Ford	Focus	2022	48.00	notAvailable	4	140
4	Nissan	Altima	2023	52.00	available	7	120
5	Chevrolet	Malibu	2022	47.00	available	4	180
6	Hyundai	Sonata	2023	49.00	notAvailable	7	140
7	BMW	3 Series	2023	60.00	available	7	249
8	Mercedes	C-Class	2022	68.00	available	8	259
9	Audi	A4	2022	55.00	notAvailable	4	250
10	Lexus	ES	2023	54.00	available	4	250
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

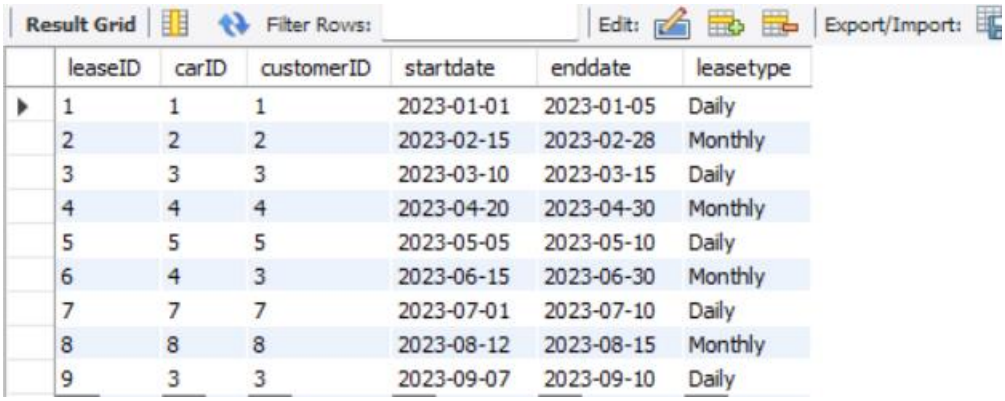
CUSTOMER TABLE:



Result Grid | Filter Rows: | Edit: | Export/Import:

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
NULL	NULL	NULL	NULL	NULL

LEASE TABLE:



Result Grid | Filter Rows: | Edit: | Export/Import:

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

PAYMENT TABLE:

```
95 •  select * from payment;
96 •  update vehicle
```

Result Grid | Filter Rows: | Edit: | Export/Import:

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

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

```
update vehicle
set dailyrate = 68
where make = 'Mercedes';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

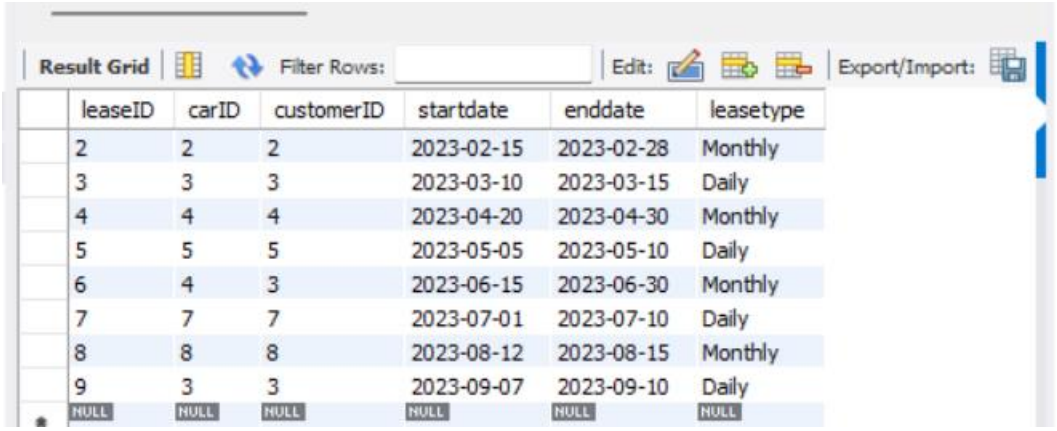
dailyrate
68.00

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

```
103     -- 2 deleting a customer n all assiciated leases and payments
104 •   delete from payment
105     where leaseID in
106     (select leaseID from lease where customerID=10);
107 •   DELETE FROM lease
108     WHERE customerID = 10;
109 •   DELETE FROM customer
110     WHERE customerID = 10;
```

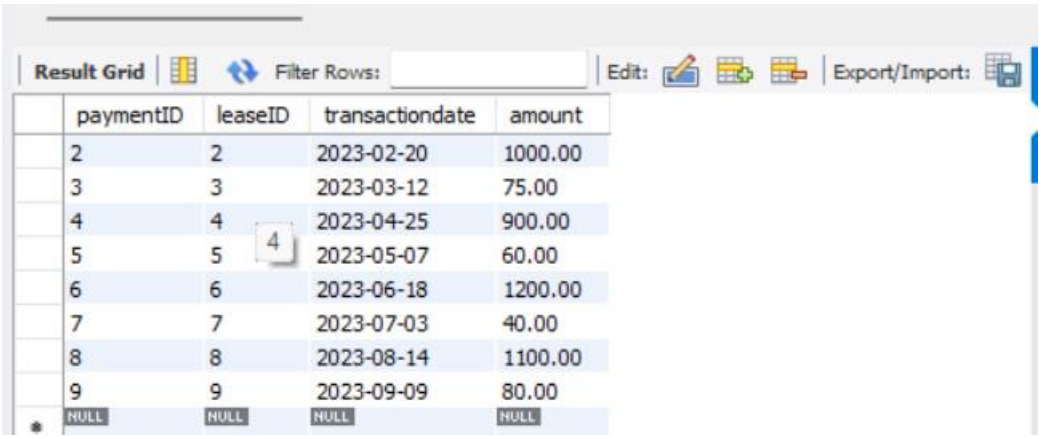
Result Grid | Filter Rows: | Edit: | Export/Import:

customerID	firstname	lastname	email	phonenumber
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
NULL	NULL	NULL	NULL	NULL



Result Grid | Filter Rows: | Edit: | Export/Import:

leaseID	carID	customerID	startdate	enddate	leasetype
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
NULL	NULL	NULL	NULL	NULL	NULL



Result Grid | Filter Rows: | Edit: | Export/Import:

paymentID	leaseID	transactiondate	amount
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
NULL	NULL	NULL	NULL

USING PROCEDURE:

```
DELIMITER //
create procedure DeleteCustNleaseNpaymentsData(in inputCustomerID INT)
BEGIN
delete from payment
where leaseID IN (
select leaseID from lease where customerID=inputCustomerID
);
delete from lease
where customerID = inputCustomerID;

delete from customer
where customerID=inputCustomerID;
END //
DELIMITER ;
```

rid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

```
274    END //
275    DELIMITER ;
276
277 •  CALL DeleteCustNleaseNpaymentsData(3);
278 •  select * from payment;
```

Result Grid | Filter Rows: | Edit:

paymentID	leaseID	txnDate	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

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

```
116 •  alter table payment
117    rename column paymentdate to transactiondate;
```

Result Grid | Filter Rows: | Ed

paymentID	leaseID	transactiondate	amount

. 4. Find a specific customer by email.

```
121    -- 4 finding a customer by email
122 •  select * from customer
123    where email = 'robert@example.com';
124
```

Result Grid | Filter Rows: | Edit: | Export/Import:

customerID	firstname	lastname	email	phonenumber
3	Robert	Johnson	robert@example.com	555-789-1234
NULL	NULL	NULL	NULL	NULL

USING PROCEDURE:

```
247 •  select * from payment;
248
249    -- QUERY 4 PROCEDURE FN
250    DELIMITER //
251 •  CREATE PROCEDURE GetcustomerByEmail(in inputEmail varchar(50))
252    BEGIN
253    select * from customer
254    where email = inputEmail;
255    END //
256    DELIMITER ;
257
258 •  CALL GetcustomerByEmail('sarah@example.com');
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

customerID	firstname	lastname	email	phonenumber
4	Sarah	Brown	sarah@example.com	555-456-7890

5. Get active leases for a specific customer.

```
125    -- 5 getting active lease for a customer
126 •  select * from lease
127    where customerID = 3
128    and '2025-04-15' between startdate and enddate;
129    -- output shows null bcz no customer has an active lease
130
```

Result Grid | Filter Rows: | Edit: | Export/Import:

leaseID	carID	customerID	startdate	enddate	leasetype
NULL	NULL	NULL	NULL	NULL	NULL

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

```
131     -- 6 all payments made by customer with spcfc phonenumber
132 •   select * from payment p
133     join lease l on p.leaseID=l.leaseID
134     join customer c on l.customerID = c.customerID
135     where c.phonenumber = '555-555-5555';
136
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

paymentID	leaseID	transactiondate	amount	leaseID	carID	customerID	startdat
1	1	2023-01-03	200.00	1	1	1	2023-01-

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

leasetype	customerID	firstname	lastname	email	phonenumber
Daily	1	John	Doe	johndoe@example.com	555-555-5555

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

```
136 •   select avg(dailyrate)
137     from vehicle
138     where lower(status) = 'available';
139
140     -- using alias
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

avg(dailyrate)
53.714286

8. Find the car with the highest daily rate.

```
145     -- 8 car with highest dailyrate
146 •   select * from vehicle
147     where dailyrate = (select max(dailyrate) from vehicle);
148
```

Result Grid | Filter Rows: | Edit: | Export/Import:

vehicleID	make	model	year	dailyrate	status	passengercapacity	engine
8	Mercedes	C-Class	2022	68.00	available	8	2599
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

9. Retrieve all cars leased by a specific customer.

```
149     -- 9 retrieve all cars leased by spcfc customer here we r using
150 •   select * from vehicle v
151     join lease l on v.vehicleID = l.carID
152     where l.customerID = 7;
153
154     -- 10 details fo most recent lease
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

vehicleID	make	model	year	dailyrate	status	passengercapacity	enginecapa
7	BMW	3 Series	2023	60.00	available	7	2499

10. Find the details of the most recent lease.

```
155 •   select * from lease
156     order by startdate desc
157     limit 1;
158
159     -- 11 all payments made in year 2023
160 •   select * from payment
```

Result Grid | Filter Rows: | Edit: | Export/Import:

leaseID	carID	customerID	startdate	enddate	leasetype
9	3	3	2023-09-07	2023-09-10	Daily
NULL	NULL	NULL	NULL	NULL	NULL

11. List all payments made in the year 2023.

```
159     -- 11 all payments made in year 2023
160 •   select * from payment
161     where year(transactiondate)=2023;
162     -- if we try 2022 we get op as null as no year is 2022
163 •   select * from payment
164     where year(transactiondate)=2022;
165
166     -- 12 customer who have not made any payments
```

Result Grid | Filter Rows: | Edit: | Export/Import:

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
NULL	NULL	NULL	NULL

```
159     -- 11 all payments made in year 2023
160 •   select * from payment
161     where year(transactiondate)=2023;
162     -- if we try 2022 we get op as null as no year is 2022
163 •   select * from payment
164     where year(transactiondate)=2022;
165
166     -- 12 customer who have not made any payments
```

Result Grid | Filter Rows: | Edit: | Export/Import:

paymentID	leaseID	transactiondate	amount
NULL	NULL	NULL	NULL

12. Retrieve customers who have not made any payments.

```
171 •   select * from customer
172     where customerID not in (
173     select lease.customerID
174     from lease
175     join payment on lease.leaseID=payment.leaseID
176     );
177
178     -- 13 retrieve car details and thier total payments
```

Result Grid | Filter Rows: | Edit: | Export/Import:

customerID	firstname	lastname	email	phonenumber
6	Laura	Hall	laura@example.com	555-234-5678
9	William	Taylor	william@example.com	555-321-6547
NULL	NULL	NULL	NULL	NULL

13. Retrieve Car Details and Their Total Payments.

```sql
select v.vehicleID,v.make,v.model,v.year,
sum(p.amount) as totalpayment
from vehicle v
join lease l on v.vehicleID = l.carID
join payment p on l.leaseID = p.leaseID
group by v.vehicleID, v.make,v.model,v.year;

-- 14 total payment for each cust
select * from customer;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

vehicleID	make	model	year	totalpayment
1	Toyota	Camry	2022	200.00
2	Honda	Civic	2023	1000.00
3	Ford	Focus	2022	155.00
4	Nissan	Altima	2023	2100.00
5	Chevrolet	Malibu	2022	60.00
7	BMW	3 Series	2023	40.00
8	Mercedes	C-Class	2022	1100.00

14. Calculate Total Payments for Each Customer.

```sql
select c.customerID,c.firstname,c.lastname,sum(p.amount)
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;

-- 15  car dets for each lease

select l.leaseID,l.startdate,l.enddate,
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

customerID	firstname	lastname	sum(p.amount)
1	John	Doe	200.00
2	Jane	Smith	1000.00
3	Robert	Johnson	1355.00
4	Sarah	Brown	900.00
5	David	Lee	60.00
7	Michael	Davis	40.00
8	Emma	Wilson	1100.00

15. List Car Details for Each Lease.

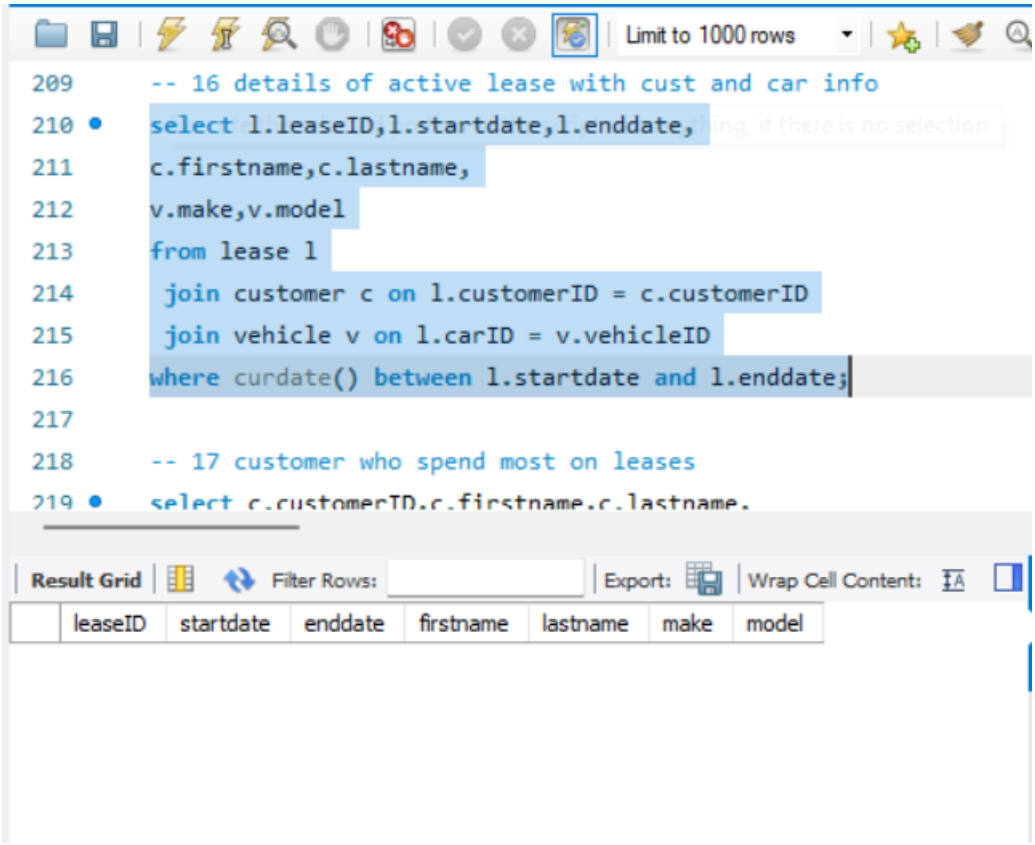
```
204 •  select l.leaseID,l.startdate,l.enddate,
205    v.vehicleID,v.make,v.year,v.model
206    from lease l
207    join vehicle v on l.carID=v.vehicleID;
208
209    -- 16 details of active lease with cust and car info
210 •  select l.leaseID,l.startdate,l.enddate,
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

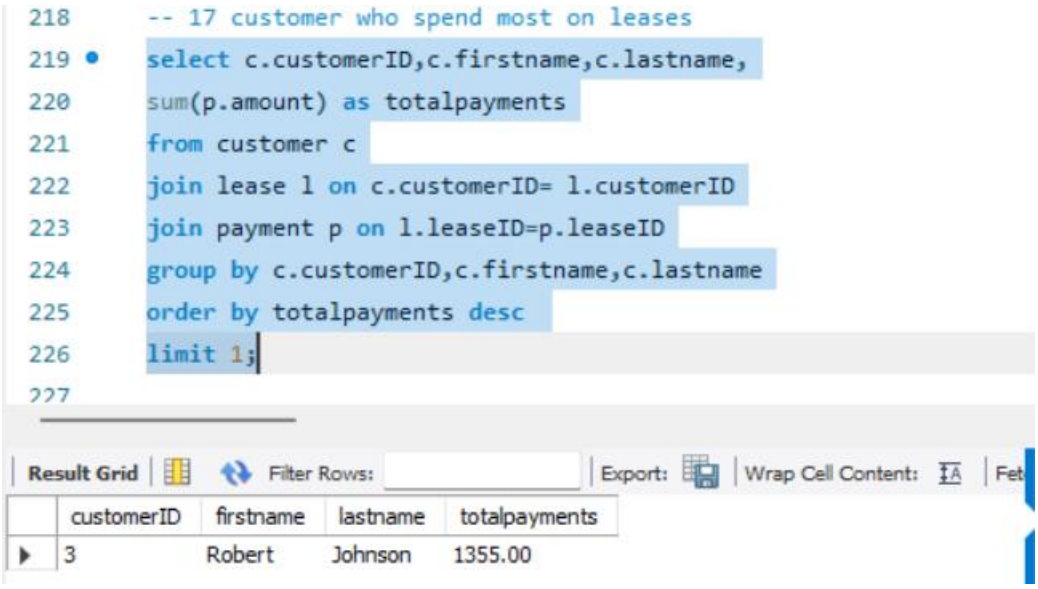
leaseID	startdate	enddate	vehicleID	make	year	model
1	2023-01-01	2023-01-05	1	Toyota	2022	Camry
2	2023-02-15	2023-02-28	2	Honda	2023	Civic
3	2023-03-10	2023-03-15	3	Ford	2022	Focus
4	2023-04-20	2023-04-30	4	Nissan	2023	Altima
5	2023-05-05	2023-05-10	5	Chevrolet	2022	Malibu
6	2023-06-15	2023-06-30	4	Nissan	2023	Altima
7	2023-07-01	2023-07-10	7	BMW	2023	3 Series
8	2023-08-12	2023-08-15	8	Mercedes	2022	C-Class
9	2023-09-07	2023-09-10	3	Ford	2022	Focus

\

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.

```
227
228     -- 18 all cars with current lease info
229 •   select v.vehicleID,v.make,v.model,v.year,
230     l.leaseID,l.customerID,l.startdate,l.enddate
231     from
232      vehicle v
233     left join lease l on v.vehicleID = l.carID
234     and curdate() between l.startdate and l.enddate;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

vehicleID	make	model	year	leaseID	customerID	startdate	enddate
1	Toyota	Camry	2022	NULL	NULL	NULL	NULL
2	Honda	Civic	2023	NULL	NULL	NULL	NULL
3	Ford	Focus	2022	NULL	NULL	NULL	NULL
4	Nissan	Altima	2023	NULL	NULL	NULL	NULL
5	Chevrolet	Malibu	2022	NULL	NULL	NULL	NULL
6	Hyundai	Sonata	2023	NULL	NULL	NULL	NULL
7	BMW	3 Series	2023	NULL	NULL	NULL	NULL
8	Mercedes	C-Class	2022	NULL	NULL	NULL	NULL
9	Audi	A4	2022	NULL	NULL	NULL	NULL