CAR RENTAL SYSTEM

CREATING TABLE

1. Vehicle table:

```
create table vehicle (

vehicleid int primary key,

make varchar(10),

model varchar(20),

year int,

dailyrate decimal(10,2),

status varchar(20) check (status in ('available', 'notAvailable')),

passengercapacity int,

enginecapacity int);
```

Output:

2. Customer table:

```
mysql> create table customer(customerid int primary key,
-> firstname varchar(20),
-> lastname varchar(20),
-> email varchar(50),
-> phonenumber varchar(12));
Query OK, 0 rows affected (0.09 sec)
```

3. Lease table:

```
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));
```

Output:

4. Payment table:

```
create table payment(
    paymentid int primary key,
    leaseid int,
    paymentdate date,
    amount decimal(10,2),
    foreign key(leaseid) references lease(leaseid));
```

Output:

INSERTING THE DATA

1. Vehicle table:

insert into vehicle (vehicleid, make, model, year, dailyrate, status, passengercapacity, enginecapacity) 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);

```
mysql> insert into vehicle (vehicleid, make, model, year, dailyrate, status, passengercapacity, enginecapacity) 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);
Query OK, 10 rows affected (0.02 sec)
Records: 10 Duplicates: 0 Warnings: 0
```

2. Customer table:

insert into customer (customerid, firstname, lastname, email, phonenumber) values

- (1, 'John', 'Doe', 'johndoe@example.com', '555-555-555'),
- (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');

Output:

```
customer (customerid, firstname,
                                                                                                     email, phonenumber) values
                               'Doe', 'johndoe@example.com',
'Smith', 'janesmith@example.c
                  'John',
                                                                                '555-555-5555')
                                Smith', 'janesmith@example.com', '555-123-4567'),
'Johnson', 'robert@example.com', '555-789-1234'),
'Brown', 'sarah@example.com', '555-456-7890'),
                  'Jane', '
'Robert',
                                'Brown',
                  'Sarah',
                                             david@example.com', '555-987-6543',
'laura@example.com', '555-234-5678'
                                             david@example.com'
                                'Lee',
'Hall'
                                    'Davis'
                                                   'michael@example.com', '555-876-5432'),
                                               , 'michael@example.com', '555-432-1098')
'emma@example.com', '555-432-1098')
                               'Wilson',
                                    'Taylor', 'william@example.com', '555-321-6547'),
'Adams', 'olivia@example.com', '555-765-4321');
                                    'Adams'
Query OK, 10 rows affected (0.02 sec)
                                            Warnings:
```

3. Lease table:

insert into lease (leaseid, vehicleid, customerid, startdate, enddate, type) values (1, 1, 1, '2023-01-01', '2023-01-05', 'DailyLease'),

(2, 2, 2, '2023-02-15', '2023-02-28', 'MonthlyLease'),

```
(3, 3, 3, '2023-03-10', '2023-03-15', 'DailyLease'),

(4, 4, 4, '2023-04-20', '2023-04-30', 'MonthlyLease'),

(5, 5, 5, '2023-05-05', '2023-05-10', 'DailyLease'),

(6, 4, 3, '2023-06-15', '2023-06-30', 'MonthlyLease'),

(7, 7, 7, '2023-07-01', '2023-07-10', 'DailyLease'),

(8, 8, 8, '2023-08-12', '2023-08-15', 'MonthlyLease'),

(9, 3, 3, '2023-09-07', '2023-09-10', 'DailyLease'),

(10, 10, 10, '2023-10-10', '2023-10-31', 'MonthlyLease');
```

```
customerid, :
'DailyLease')
                                                                       startdate, enddate, type) values
                       lease (leaseid,
                                          vehicleid,
                     '2023-01-01',
                                      '2023-01-05',
                                      '2023-02-28'
                                                        'MonthlyLease'),
             2, 2,
                    '2023-02-15'
    -> (3, 3, 3, '2023-03-10'
-> (4, 4, 4, '2023-04-20'
                                      '2023-03-15'
                                                       'DailyLease')
                                      '2023-04-30'
                                                        'MonthlyLease'),
                    '2023-05-05'
                                      '2023-05-10'
                                                       'DailyLease')
        (5, 5, 5,
                    '2023-06-15'
                                      '2023-06-30'
                                                       'MonthlyLease'),
        (6, 4, 3,
                    '2023-07-01'
                                      '2023-07-10'
                                                       'DailyLease')
                                                        'MonthlyLease'),
                    '2023-08-12'
                                      '2023-08-15'
    -> (9, 3, 3, '2023-09-07', '20
-> (10, 10, 10, '2023-10-10',
                                      '2023-09-10'
                                                        'DailyLease'),
                                          '2023-10-31', 'MonthlyLease');
Query OK, 10 rows affected (0.02 sec)
Records: 10 Duplicates: 0 Warnings:
                                  Warnings:
```

4. Payment table:

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);

```
mysql> insert into payment (paymentid, leaseid, paymentdate, amount) values
               '2023-01-03'
                             200.00)
               '2023-02-20'
                             1000.00),
              '2023-03-12'
                             75.00)
               '2023-04-25'
                             900.00),
               '2023-05-07'
               '2023-06-18'
               '2023-07-03'
               '2023-08-14'
                             1100.00),
              '2023-09-09'
                             80.00),
           10, '2023-10-25'
                               1500.00);
Query OK, 10 rows affected (0.02 sec)
Records: 10
             Duplicates: 0
                             Warnings:
```

mysql> select * from vehicle; vehicleid | year | dailyrate | enginecapacity | make model status passengercapacity Camry Civic Toyota 2022 50.00 available 1450 1 2 3 4 45.00 48.00 52.00 47.00 49.00 Honda available 1500 1400 1200 1800 Ford Focus notAvailable 4 7 4 7 7 8 available available Nissan Altima Chevrolet Malibu notAvailable available available Hyundai Sonata 1400 6 7 8 60.00 58.00 BMW 3 Series 2499 Mercedes C-Class 2599 55.00 54.00 9 Audi Α4 notAvailable 2500 available 10 İ Lexus ES

10 rows in set (0.00 sec)

mysql> select * from customer;

customerid	firstname	lastname	email	phonenumber
1 1 2 3 4 4 5 5 6 7 8 8 9 10 10 10	John Jane Robert Sarah David Laura Michael Emma William Olivia	Doe Smith Johnson Brown Lee Hall Davis Wilson Taylor Adams	johndoe@example.com janesmith@example.com robert@example.com sarah@example.com david@example.com laura@example.com michael@example.com emma@example.com william@example.com olivia@example.com	555-555-5555 555-123-4567 555-789-1234 555-456-7890 555-987-6543 555-234-5678 555-876-5432 555-432-1098 555-321-6547 555-765-4321

10 rows in set (0.00 sec)

mysql> select * from lease;

+	Ł	L	L	L	L
leaseid	vehicleid	customerid	startdate	enddate	type
1	1	1	2023-01-01	2023-01-05	DailyLease
2	2	2	2023-02-15	2023-02-28	MonthlyLease
3	3	3	2023-03-10	2023-03-15	DailyLease
4	4	4	2023-04-20	2023-04-30	MonthlyLease
5	5	5	2023-05-05	2023-05-10	DailyLease
6	4	3	2023-06-15	2023-06-30	MonthlyLease
7	7	7	2023-07-01	2023-07-10	DailyLease
8	8	8	2023-08-12	2023-08-15	MonthlyLease
9	3	3	2023-09-07	2023-09-10	DailyLease
10	10	10	2023-10-10	2023-10-31	MonthlyLease

10 rows in set (0.00 sec)

mysql> select * from payment;

paymentid	leaseid	paymentdate	amount
1 2	1 2	2023-01-03 2023-02-20	200.00 1000.00
3 4	3 4	2023-03-12 2023-04-25	75.00 900.00
5	5	2023-05-07	60.00
6	6	2023-06-18	1200.00
7 8	7 8	2023-07-03 2023-08-14	40.00 1100.00
9	9	2023-09-09	80.00
10	10	2023-10-25	1500.00
+		t	

10 rows in set (0.00 sec)

QUERIES:

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

Query: update vehicle set dailyrate = 68.00 where make = 'Mercedes';

Output:

```
mysql> update vehicle set dailyrate = 68.00 where make = 'Mercedes';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnin
                                 Warnings: 0
mysql> select * from vehicle;
  vehicleid |
               make
                              model
                                          year
                                                  dailyrate
                                                                status
                                                                                  passengercapacity
                                                                                                         enginecapacity
                Toyota
                                           2022
                                                        50.00
                                                                 available
                                                                                                    4
7
4
                                                       45.00
48.00
                Honda
                              Civic
                                           2023
                                                                 available
                                                                                                                     1500
           3
                                           2022
                                                                 notAvailable
                                                                                                                     1400
                Ford
                              Focus
                                                                                                    7
4
           4
                                                       52.00
47.00
                              Altima
                                           2023
                                                                 available
                Nissan
                                                                                                                     1200
                Chevrolet
                              Malibu
                                           2022
                                                                 available
           5
                                                                                                                     1800
                                                                                                     7
7
                Hyundai
BMW
                                           2023
                                                       49.00
                                                                 notAvailable
                                                                                                                     1400
                              Sonata
                              3 Series
                                           2023
                                                       60.00
                                                                 available
                                                                                                                     2499
                                           2022
                              C-Class
                                                                                                     8
                Mercedes
                                                        68.00
                                                                 available
                                                                                                                     2599
                                           2022
                                                        55.00
                Audi
                                                                 notAvailable
                                                                                                                     2500
                Lexus
                                           2023
                                                                 available
                                                                                                                     2500
10 rows in set (0.00 sec)
```

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

Query: delete from payment where leaseid in (select leaseid from lease where customerid = <customerid>);

delete from lease where customerid = 10;

delete from customer where customerid = 10;

```
mysql> delete from payment where leaseid in (select leaseid from lease where customerid = 10);
Query OK, 1 row affected (0.01 sec)
mysql> delete from lease where customerid = 10;
Query OK, 1 row affected (0.01 sec)
mysql> delete from customer where customerid = 10;
Query OK, 1 row affected (0.01 sec)
mysgl> select * from customer;
 customerid | firstname
                           lastname
                                                                phonenumber
               John
                           Doe
                                       johndoe@example.com
                                                                555-555-5555
                            Smith
                                       janesmith@example.com
                                                                555-123-4567
           2
               Jane
           3
               Robert
                            Johnson
                                       robert@example.com
                                                                555-789-1234
                                       sarah@example.com
                                                                555-456-7890
               Sarah
                           Brown
                                                                555-987-6543
               David
                                       david@example.com
                           Lee
                           Hall
           6
               Laura
                                       laura@example.com
                                                                555-234-5678
           7
               Michael
                           Davis
                                       michael@example.com
                                                                555-876-5432
                                                                555-432-1098
           8
                           Wilson
               Emma
                                       emma@example.com
               William
                            Taylor
                                       william@example.com
                                                                555-321-6547
 rows in set (0.00 sec)
```

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

Query: alter table payment rename column paymentdate to transactiondate;

Output:

```
mysql> alter table payment rename column paymentdate to transactiondate;
Query OK, 0 rows affected (0.02 sec)
Records: 0 Duplicates: 0 Warnings: 0
mysql> desc payment;
  Field
                                                         Default | Extra
                       Type
                                         Null |
                                                 Key
  paymentid
                                                  PRI
                       int
                                          NO
                                                         NULL
                                          YES
                                                  MUL
  leaseid
                       int
                                                         NULL
  transactiondate
                       date
                                          YES
                                                         NULL
                       decimal(10,2)
  amount
                                         YES
                                                         NULL
  rows in set (0.01 sec)
```

4. Find a specific customer by email.

Query: select * from customer where email = sarah@example.com;

Output:

```
mysql> select * from customer where email = 'sarah@example.com';
+------+
| customerid | firstname | lastname | email | phonenumber |
+-----+
| 4 | Sarah | Brown | sarah@example.com | 555-456-7890 |
+-----+
1 row in set (0.00 sec)
```

5. Get active leases for a specific customer.

Query: select c.customerid, concat(c.firstname,' ',c.lastname) as customername, l.leaseid, l.type

from customer c

join lease 1 on c.customerid = 1.customerid

where c.customerid = 5 and curdate() <= 1.enddate;

Output:

```
mysql> select c.customerid, concat(c.firstname,' ',c.lastname) as customername, l.leaseid, l.type from customer c join lease l on c.customerid = l.customerid d where c.customerid = 5 and curdate() <= l.enddate;
Empty set (0.00 sec)
```

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

Query: select c.customerid, concat(c.firstname, '', c.lastname) as customername, p.paymentid, p.transactiondate from customer c join lease l on c.customerid = l.customerid

join payment p on l.leaseid = p.leaseid

where c.phonenumber = '555-456-7890';

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

Query: select vehicleid, make, model, avg(dailyrate) as avgofdailyrate

from vehicle

where status = 'available'

group by vehicleid, make, model;

Output:

					icce where	scacus -	'available'	group by	venicieia,	make,	model;
vehicleid	make	model	avgofdailyrate								
1	Toyota	Camry	50.000000								
2	Honda	Civic	45.000000								
4	Nissan	Altima	52.000000								
5	Chevrolet	Malibu	47.000000								
7	BMW	3 Series	60.000000								
8	Mercedes	C-Class	68.000000								
10	Lexus	ES	54.000000								
+	+	+									
7 rows in se	t (0.00 sec)										

8. Find the car with the highest daily rate.

Query: select * from vehicle order by dailyrate desc limit 1;

Output:

mysql> select	* from vel	nicle orden	by da:	ilyrate desc	limit 1;		
vehicleid	make	model	year	dailyrate	status	passengercapacity	enginecapacity
8	Mercedes	C-Class	2022	68.00	available	8	2599
1 row in set	(0.00 sec)						

9. Retrieve all cars leased by a specific customer.

Query: select v.*
from vehicle v
join lease l on v.vehicleid = l.vehicleid
where l.customerid = 5;

mysql> select	t v.* from ve	ehicle v	join lea	ase l on v.ve	ehicleid = l	.vehicleid where l.cu	ustomerid = 5;
vehicleid	make	model	year	dailyrate	status	passengercapacity	enginecapacity
5	Chevrolet	Malibu	2022	47.00	available	4	1800
1 row in set	(0.00 sec)	,				•	

10. Find the details of the most recent lease.

Query: select *

from lease

order by startdate desc limit 1;

Output:

```
mysql> select * from lease order by startdate desc limit 1;
+-----+
| leaseid | vehicleid | customerid | startdate | enddate | type |
+-----+
| 9 | 3 | 3 | 2023-09-07 | 2023-09-10 | DailyLease |
+-----+
1 row in set (0.00 sec)
```

11. List all payments made in the year 2023.

Query: select * from payment where transactiondate between '2023-01-01' and '2023-12-31';

Output:

```
mysql> select * from payment where transactiondate between '2023-01-01' and '2023-12-31';
  paymentid | leaseid | transactiondate
                                            amount
                         2023-01-03
                                              200.00
          1
2
3
                     2
                         2023-02-20
                                            1000.00
                     3
                         2023-03-12
                                               75.00
          4
                         2023-04-25
                                              900.00
          5
                         2023-05-07
                                               60.00
          6
                     6
                                             1200.00
                         2023-06-18
          7
                     7
                         2023-07-03
                                               40.00
          8
                     8
                         2023-08-14
                                             1100.00
          9
                         2023-09-09
                                               80.00
 rows in set (0.00 sec)
```

12. Retrieve customers who have not made any payments

Query: select c.*

from customer c

where not exists (select 1

from payment p

join lease 1 on p.leaseid = l.leaseid

where l.customerid = c.customerid);

Output:

mysql> select ->);	c.* from cu	stomer c who	ere not exists (select	1 from payment	p join lease l on p.leaseid = l.leaseid where l.customerid = c.customerid
customerid	firstname	lastname	email	phonenumber	
6 9	Laura William	Hall Taylor	laura@example.com william@example.com	555-234-5678 555-321-6547	
2 rows in set	(0.02 sec)				

13. Retrieve Car Details and Their Total Payments.

Query: select v.vehicleid, v.make, v.model, v.year, v.dailyrate, sum(p.amount) as total_payments

from vehicle v join lease l on v.vehicleid = l.vehicleid join payment p on l.leaseid = p.leaseid group by v.vehicleid;

Output:

-> from v -> join l -> join p		· .vehicleid : l.leaseid :	= l.vehi	icleid	dailyrate, sum(p.:	amount) as total_payments
vehicleid	make	model	year	dailyrate	total_payments	
1	Toyota	Camry	2022	50.00	200.00	Ĭ
2	Honda	Civic	2023	45.00	1000.00	l
3	Ford	Focus	2022	48.00	155.00	l
4	Nissan	Altima	2023	52.00	2100.00	1
5	Chevrolet	Malibu	2022	47.00	60.00	l
7	BMW	3 Series	2023	60.00	40.00	
8	Mercedes	C-Class	2022	68.00	1100.00	l e
+ 7 rows in set	(0.00 sec)	+	+		 	+

14. Calculate Total Payments for Each Customer.

Query: select c.customerid, c.firstname, c.lastname, sum(p.amount) as total_payments from customer c join lease 1 on c.customerid = 1.customerid join payment p on l.leaseid = p.leaseid group by c.customerid, c.firtname, c.lastname;

Output:

p on l.leasei	d = p.lease	id group by		sum(p.amount) as total_payments +rom customer c join lease l on c.customerid = l.customerid join paymen .firstname, c.lastname;
customerid	firstname	lastname	total_payments	
1 . 2 . 3 ! 4 ! 5 !	John Jane Robert Sarah David Michael	Doe Smith Johnson Brown Lee Davis Wilson	200.00 1000.00 1355.00 900.00 60.00 40.00	

15. List Car Details for Each Lease.

Query: select l.leaseid, v.vehicleid, v.make, v.model, v.year, v.dailyrate, l.startdate, l.enddate from lease l join vehicle v on l.vehicleid = v.vehicleid;

-> join	vehicle v	on l.vehicle	id = v.vehi	cleid;			
leaseid	vehicleid	make	model	year	dailyrate	startdate	enddate
1 1	1	Toyota	Camry	2022	50.00	 2023-01-01	2023-01-05
2	2	Honda	Civic	2023	45.00	2023-02-15	2023-02-28
3	3	Ford	Focus	2022	48.00	2023-03-10	2023-03-15
4	4	Nissan	Altima	2023	52.00	2023-04-20	2023-04-30
5	5	Chevrolet	Malibu	2022	47.00	2023-05-05	2023-05-10
6	4	Nissan	Altima	2023	52.00	2023-06-15	2023-06-30
7	7	BMW	3 Series	2023	60.00	2023-07-01	2023-07-10
8	8	Mercedes	C-Class	2022	68.00	2023-08-12	2023-08-15
9	3	Ford	Focus	2022	48.00	2023-09-07	2023-09-10

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

Query: select l.leaseid, l.startdate, l.enddate, c.firstname, c.lastname, v.make, v.model, v.dailyrate from lease l
join customer c on l.customerid = c.customerid

join customer c on l.customerid = c.customerid join vehicle v on l.vehicleid = v.vehicleid where l.enddate >= curdate();

Output:

```
mysql> select l.leaseid, l.startdate, l.enddate, c.firstname, c.lastname, v.make, v.model, 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.

Query: 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
order by total_spent desc
limit 1;

18. List All Cars with Their Current Lease Information.

Query: select v.vehicleid, v.make, v.model, v.year, v.dailyrate, l.leaseid, l.startdate, l.enddate, c.firstname, c.lastname from vehicle v join lease l on v.vehicleid = l.vehicleid join customer c on l.customerid = c.customerid where l.enddate >= curdate();