**Coding Challenge - Car Rental System – SQL**

Create database car;

Use car;

Vehicle:

Create table vehicle(vehicleID integer(4) Primary Key ,

make varchar(20),

model varchar(20),

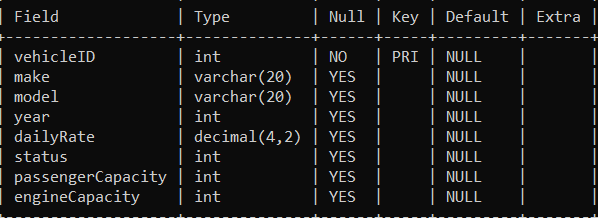
year integer(5),

dailyRate decimal (2,2),

status integer(2),

passengerCapacity integer(4),

engineCapacity integer(5));



insert into vehicle values(1, ‘Toyota’, ‘Camry’, 2022, 50.00,1, 4, 1450);

insert into vehicle values(2, ‘Honda’, ‘Civic’, 2023, 45.00,1, 7, 1500);

insert into vehicle values(3, ‘Ford’, ‘Focus’, 2022, 48.00,0, 4, 1400);

insert into vehicle values(4, ‘Nissan’, ‘Altima’, 2023, 52.00,1, 7, 1200);

insert into vehicle values(5, ‘Chevrolet’, ‘Malibu’, 2022, 47.00,1, 4, 1800);

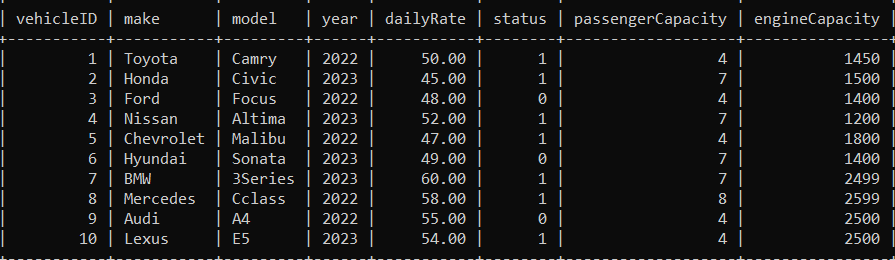
insert into vehicle values(6, ‘Hyundai’, ‘Sonata’, 2023, 49.00,0, 7, 1400);

insert into vehicle values(7, ‘BMW’, ‘ 3Series’, 2023, 60.00,1, 7, 2499);

insert into vehicle values(8, ‘Mercedes’, ‘ Cclass’, 2022, 58.00,8, 7, 2599);

insert into vehicle values(9, ‘Audi’, ‘ A4’, 2022, 55.00,0, 4, 2500);

insert into vehicle values(10, ‘Lexus’, ‘ E5’, 2023, 54.00,1, 4, 2500);



Customer:

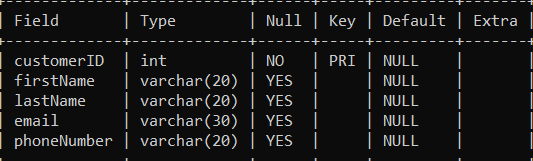
Create table customer(customerID integer(2) Primary Key ,

firstName varchar(20),

lastName varchar(20),

email varchar(30),

phoneNumber varchar(20));



insert into customer values(1, ‘John’, ‘Doe’, ‘ johndoe@example.com’, ‘555-555-5555’);

insert into customer values(2, ‘Jane’, ‘Smith’, ‘janesmith@example.com ’, ‘555-123-4567’);

insert into customer values(3, ‘Robert’, ‘Johnson’, ‘ robert@example.com’, ‘555-789-1234’);

insert into customer values(4, ‘Sarah’, ‘Brown’, ‘ sarah@example.com’, ‘555-456-7890’);

insert into customer values(5, ‘David’, ‘Lee’, ‘ david@example.com’, ‘555-987-6543’);

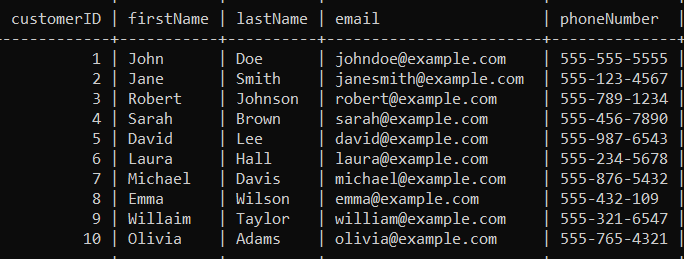
insert into customer values(6, ‘Laura’, ‘Hall’, ‘ laura@example.com’, ‘555-234-5678’);

insert into customer values(7, ‘Michael’, ‘Davis’, ‘ michael@example.com’, ‘555-876-5432’);

insert into customer values(8, ‘Emma’, ‘Wilson’, ‘ emma@example.com’, ‘555-432-1098’);

insert into customer values(9, ‘Willaim’, ‘Taylor’, ‘ william@example.com’, ‘555-321-6547’);

insert into customer values(10, ‘Olivia’, ‘Adams’, ‘ olivia@example.com’, ‘555-765-4321’);



Lease:

create table lease (

leaseID integer(4) PRIMARY KEY,

vehicleID integer(4),

customerID integer(2),

startDate DATE,

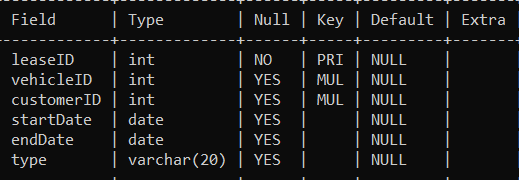
endDate DATE,

type VARCHAR(20),

FOREIGN KEY (vehicleID) REFERENCES Vehicle(vehicleID),

FOREIGN KEY (customerID) REFERENCES Customer(customerID)

);



INSERT INTO lease VALUES (1, 1, 1, '2023-01-01', '2023-01-05', 'Daily');

INSERT INTO lease VALUES (2, 2, 2, '2023-02-15', '2023-02-28', 'Monthly');

INSERT INTO lease VALUES (3, 3, 3, '2023-03-10', '2023-03-15', 'Daily');

INSERT INTO lease VALUES (4, 4, 4, '2023-04-20', '2023-04-30', 'Monthly');

INSERT INTO lease VALUES (5, 5, 5, '2023-05-05', '2023-05-10', 'Daily');

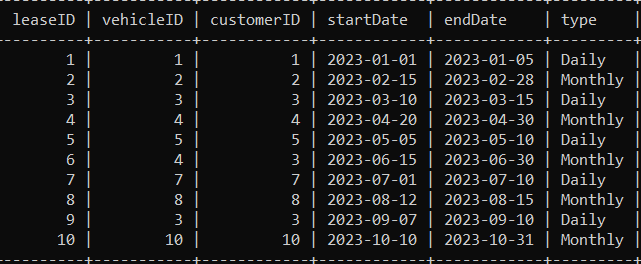
INSERT INTO lease VALUES (6, 4, 3, '2023-06-15', '2023-06-30', 'Monthly');

INSERT INTO lease VALUES (7, 7, 7, '2023-07-01', '2023-07-10', 'Daily');

INSERT INTO lease VALUES (8, 8, 8, '2023-08-12', '2023-08-15', 'Monthly');

INSERT INTO lease VALUES (9, 3, 3, '2023-09-07', '2023-09-10', 'Daily');

INSERT INTO lease VALUES (10, 10, 10, '2023-10-10', '2023-10-31', 'Monthly');



Payment:

CREATE TABLE payment (

paymentID INTEGER(4) PRIMARY KEY,

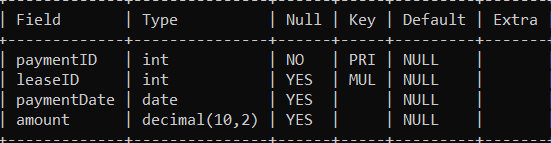
leaseID INTEGER(4),

paymentDate DATE,

amount DECIMAL(10, 2),

FOREIGN KEY (leaseID) REFERENCES lease(leaseID)

);



INSERT INTO payment VALUES (1, 1, '2023-01-03', 200.00);

INSERT INTO payment VALUES (2, 2, '2023-02-20', 1000.00);

INSERT INTO payment VALUES (3, 3, '2023-03-12', 75.00);

INSERT INTO payment VALUES (4, 4, '2023-04-25', 900.00);

INSERT INTO payment VALUES (5, 5, '2023-05-07', 60.00);

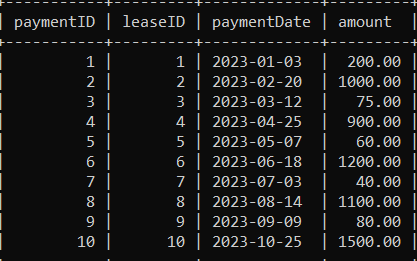
INSERT INTO payment VALUES (6, 6, '2023-06-18', 1200.00);

INSERT INTO payment VALUES (7, 7, '2023-07-03', 40.00);

INSERT INTO payment VALUES (8, 8, '2023-08-14', 1100.00);

INSERT INTO payment VALUES (9, 9, '2023-09-09', 80.00);

INSERT INTO payment VALUES (10, 10, '2023-10-25', 1500.00);



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

Update vehicle set dailyRate=68.00 where make= ‘Mercedes’;

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

delete from payment where leaseID IN (

select leaseID from lease

where customerID = 5);

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

alter table payment

rename column paymentDate to transactionDate;

4. Find a specific customer by email.

Select customer\_id, first\_name, last\_name from

customer where email= ‘robert@example.com’;

5. Get active leases for a specific customer.

select leaseID, customerID, startDate, enddate

from lease where customerID = 5 and

CURRENT\_DATE between startDate and endDate;

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

select p.paymentID, p.paymentDate, p.amount, c.customerID, c.phoneNumber

from payment p join customer c

ON p.leaseID in (select leaseID from lease where customerID = c.customerID)

where c.phoneNumber = ' 555-234-5678;

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

Select status, avg(dailyRate) from vehicle group by status having status=1;

8. Find the car with the highest daily rate.

Select vehicleID, make from vehicle

where dailyRate = (select MAX(dailyRate) from vehicle);

9. Retrieve all cars leased by a specific customer.

Select v.vehicleID, v.make, v.model, l.leaseID

from vehicle v join lease l

on v.vehicleID = l.vehicleID where customerID=3;

10. Find the details of the most recent lease.

Select leaseID, startDate, endDate from lease where

endDate=(select max(endDate) from lease);

11. List all payments made in the year 2023.

select paymentID, paymentDate, amount

from payment

where paymentDate LIKE ‘2023-%-%’;

12. Retrieve customers who have not made any payments.

Select c.customerID, c.firstName, c.lastName from customer c

left join lease l on c.customerID = l.customerID

left join payment p on l.leaseID = p.leaseID

where p.paymentID is null;

13. Retrieve Car Details and Their Total Payments.

select v.vehicleID, v.make, p.paymentID, sum(p.amount)

from vehicle v

LEFT JOIN lease l on v.vehicleID = l.vehicleID

LEFT JOIN payment p on l.leaseID = p.leaseID

Group by v.vehicleID, v.make, p.paymentID;

14. Calculate Total Payments for Each Customer.

Select c.customerID, c.firstName, c.lastName, sum(p.amount)

from customer c

left join lease l on c.customerID = l.customerID

left join payment p on l.leaseID = p.leaseID

group by c.customerID, c.firstName, c.lastName;

15. List Car Details for Each Lease.

Select v.vehicleID, v.make, v.model, v.year, l.leaseID

from vehicle v left join lease l

on v.vehicleID = l.vehicleID;

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

Select l.leaseID, c.customerID, c.firstName, c.lastName, v.vehicleID, v.make,

v.model, l.startDate, l.endDate

from lease join Customer c

on l.customerID = c.customerID

join Vehicle v on l.vehicleID = v.vehicleID

where CURRENT\_DATE BETWEEN l.startDate AND l.endDate;

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

select c.customerID, c.firstName, c.lastName,

sum(p.amount) as mostSpent

from customer as c join lease as l

on c.customerID = l.customerID

join payment p on l.leaseID = p.leaseID

group by c.customerID, c.firstName, c.lastName

order by mostSpent desc

LIMIT 1;

18. List All Cars with Their Current Lease Information.

Select v.vehicleID, v.make, v.model, v.year, l.leaseID, l.startDate, l.endDate

from vehicle v join lease l

on v.vehicleID = l.vehicleID;