create database car\_rental;

use car\_rental;

CREATE TABLE Vehicle (

vehicleID INT PRIMARY KEY,

make VARCHAR(50),

model VARCHAR(50),

year INT,

dailyRate DECIMAL(10,2),

status VARCHAR(20), -- 'available' or 'notAvailable'

passengerCapacity INT,

engineCapacity INT

);

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);

CREATE TABLE Customer (

customerID INT PRIMARY KEY,

firstName VARCHAR(50),

lastName VARCHAR(50),

email VARCHAR(100),

phoneNumber VARCHAR(20)

);

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, 'Olivia', 'Adams', 'olivia@example.com', '555-765-4321');

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)

);

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

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');

CREATE TABLE Payment (

paymentID INT PRIMARY KEY,

leaseID INT,

paymentDate DATE,

amount DECIMAL(10,2),

FOREIGN KEY (leaseID) REFERENCES Lease(leaseID)

);

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),

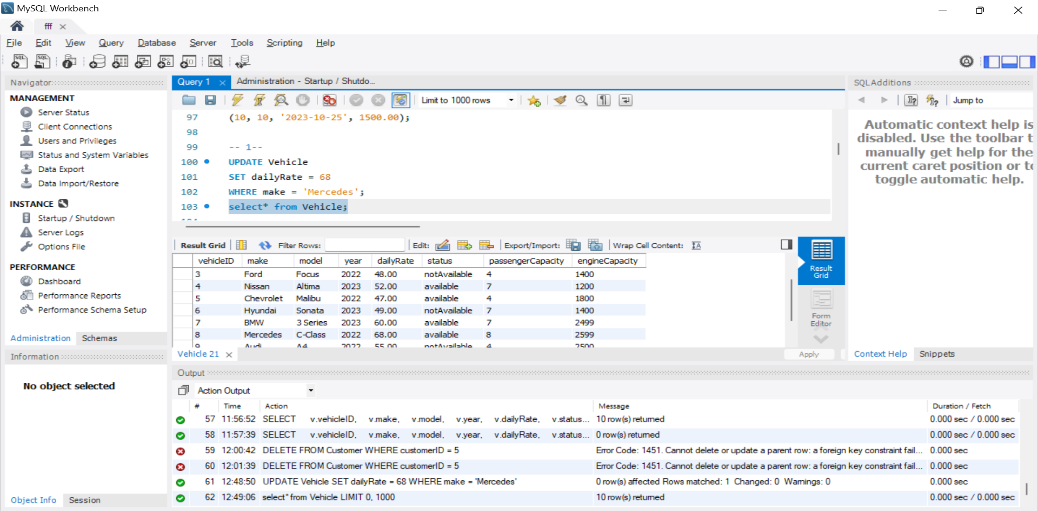
(10, 10, '2023-10-25', 1500.00);

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

UPDATE Vehicle

SET dailyRate = 68

WHERE make = 'Mercedes';



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

ALTER TABLE Payment

RENAME COLUMN paymentDate TO transactionDate;

SHOW COLUMNS FROM Payment;

A screenshot of a computer

AI-generated content may be incorrect.

(4)Find a specific customer by email.

SELECT \*

FROM Customer

WHERE email = 'johndoe@example.com';

A screenshot of a computer

AI-generated content may be incorrect.

(5)Get active leases for a specific customer.

SELECT

l.leaseID,

v.make,

v.model,

l.startDate,

l.endDate

FROM Lease l

JOIN Customer c ON l.customerID = c.customerID

JOIN Vehicle v ON l.vehicleID = v.vehicleID

WHERE c.email = 'robert@example.com';

A screenshot of a computer

AI-generated content may be incorrect.

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

SELECT

p.paymentID,

p.transactionDate, -- column name changed:

p.amount,

l.leaseID,

v.make,

v.model

FROM Payment p

JOIN Lease l ON p.leaseID = l.leaseID

JOIN Customer c ON l.customerID = c.customerID

JOIN Vehicle v ON l.vehicleID = v.vehicleID

WHERE c.phoneNumber = '555-789-1234';A screenshot of a computer

AI-generated content may be incorrect.

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

SELECT AVG(dailyRate) AS averageDailyRate

FROM Vehicle

WHERE status = 'available';

A screenshot of a computer

AI-generated content may be incorrect.

(8)Find the car with the highest daily rate.

select \* FROM Vehicle

ORDER BY dailyRate DESC

LIMIT 1;A screenshot of a computer

AI-generated content may be incorrect.

(9)Retrieve all cars leased by a specific customer.

SELECT

v.make,

v.model,

v.year,

v.dailyRate,

l.startDate,

l.endDate

FROM Lease l

JOIN Vehicle v ON l.vehicleID = v.vehicleID

JOIN Customer c ON l.customerID = c.customerID

WHERE c.email = 'robert@example.com';

A screenshot of a computer

AI-generated content may be incorrect.

(10). Find the details of the most recent lease.

SELECT

l.leaseID,

v.make,

v.model,

c.firstName,

c.lastName,

l.startDate,

l.endDate,

l.type

FROM Lease l

JOIN Vehicle v ON l.vehicleID = v.vehicleID

JOIN Customer c ON l.customerID = c.customerID

ORDER BY l.startDate DESC

LIMIT 1;

A screenshot of a computer

AI-generated content may be incorrect.

(11)List all payments made in the year 2023.

SELECT

p.paymentID,

p.leaseID,

p.transactionDate,

p.amount

FROM Payment p

WHERE YEAR(p.transactionDate) = 2023;

A screenshot of a computer

AI-generated content may be incorrect.

(12)Retrieve customers who have not made any payments.

SELECT

c.customerID,

c.firstName,

c.lastName,

c.email,

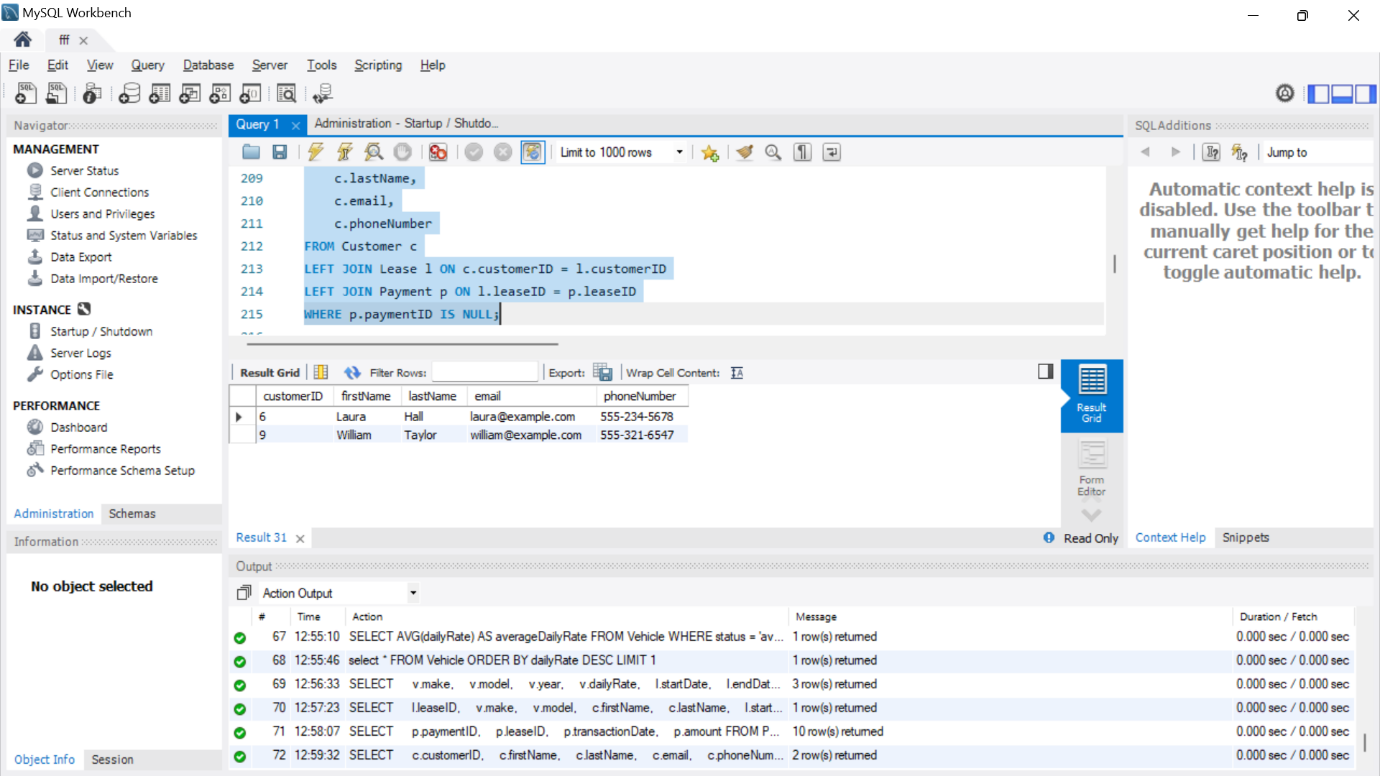
c.phoneNumber

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,

v.model,

v.year,

v.dailyRate,

SUM(p.amount) AS totalPayments

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, v.model, v.year, v.dailyRate;

A screenshot of a computer

AI-generated content may be incorrect.

(14) Calculate Total Payments for Each Customer.

SELECT

c.customerID,

c.firstName,

c.lastName,

SUM(p.amount) AS totalPayments

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;

A screenshot of a computer

AI-generated content may be incorrect.

(15)List Car Details for Each Lease.

SELECT

l.leaseID,

v.vehicleID,

v.make,

v.model,

v.year,

v.dailyRate,

v.status,

v.passengerCapacity,

v.engineCapacity,

l.startDate,

l.endDate,

l.type AS leaseType

FROM Lease l

JOIN Vehicle v ON l.vehicleID = v.vehicleID;

A screenshot of a computer

AI-generated content may be incorrect.

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

SELECT

l.leaseID,

l.startDate,

l.endDate,

l.type AS leaseType,

c.customerID,

c.firstName,

c.lastName,

c.email,

c.phoneNumber,

v.vehicleID,

v.make,

v.model,

v.year,

v.dailyRate,

v.status,

v.passengerCapacity,

v.engineCapacity

FROM Lease l

JOIN Vehicle v ON l.vehicleID = v.vehicleID

JOIN Customer c ON l.customerID = c.customerID;

A screenshot of a computer

AI-generated content may be incorrect.

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

SELECT

c.customerID,

c.firstName,

c.lastName,

c.email,

c.phoneNumber,

SUM(p.amount) AS totalSpent

FROM Payment p

JOIN Lease l ON p.leaseID = l.leaseID

JOIN Customer c ON l.customerID = c.customerID

GROUP BY c.customerID

ORDER BY totalSpent DESC

LIMIT 1;

A screenshot of a computer

AI-generated content may be incorrect.

(18)List All Cars with Their Current Lease Information.

SELECT

v.vehicleID,

v.make,

v.model,

v.year,

v.dailyRate,

v.status AS carStatus,

l.leaseID,

l.startDate,

l.endDate,

l.type AS leaseType,

c.customerID,

c.firstName,

c.lastName,

c.email,

c.phoneNumber

FROM Vehicle v

JOIN Lease l ON v.vehicleID = l.vehicleID

JOIN Customer c ON l.customerID = c.customerID

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

A screenshot of a computer

AI-generated content may be incorrect.