# **Coding Challenge Car Rental System**

**SUBMITED BY: E.R HARISH** 

DATE OF SUBMISSION: 26/09/2024

**Coding Challenge: Car Rental System** 

# **GITHUB link:**

https://github.com/Harish0562/E.R-Harish-Coding-Challenge-Car-Rental-System-Hexaware.git

### **Creating Database named Car Rental System:**

```
Commands completed successfully.

Completion time: 2024-09-23T14:11:08.4520340+05:30
```

#### **Creating Tables:**

#### **Vehicle Table:**

```
CREATE TABLE Vehicle (
    vehicleID INT PRIMARY KEY,
    make VARCHAR(50),
    model VARCHAR(50),
    year INT,
    dailyRate DECIMAL(5, 2),
    status TINYINT, -- 1 = available, 0 = notAvailable
    passengerCapacity INT,
    engineCapacity DECIMAL(6, 2)
);
```

#### **Customer Table:**

```
customerID INT PRIMARY KEY,
firstName VARCHAR(50),
lastName VARCHAR(50),
email VARCHAR(100),
phoneNumber VARCHAR(20)
);
```

#### Lease Table:

```
☐ CREATE TABLE Lease (
       leaseID INT PRIMARY KEY,
       vehicleID INT,
       customerID INT,
       startDate DATE.
       endDate DATE,
       leaseType VARCHAR(20), -- 'Daily' or 'Monthly'
       FOREIGN KEY (vehicleID) REFERENCES Vehicle(vehicleID),
       FOREIGN KEY (customerID) REFERENCES Customer(customerID)
  );
Payment Table:
□ CREATE TABLE Payment (
       paymentID INT PRIMARY KEY,
       leaseID INT,
       transactionDate DATE,
       amount DECIMAL(10, 2),
       FOREIGN KEY (leaseID) REFERENCES Lease(leaseID)
  );
Inserting records into each tables Vehicle and Customer Table:
```

```
☐ INSERT INTO Vehicle (vehicleID, make, model, year, dailyRate, status, passengerCapacity, 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),
 (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);

☐INSERT INTO Customer (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');
```

#### **Inserting into Lease table:**

```
□INSERT INTO Lease (leaseID, vehicleID, customerID, startDate, endDate, leaseType)

VALUES

(1, 1, 1, '2023-01-01', '2023-02-28', 'Monthly'),

(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');
```

#### **Inserting into Payment table:**

```
□ INSERT INTO Payment (paymentID, leaseID, transactionDate, 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.

#### Before updating the daily rate of Mercedes:

	vehicleID	make	model	year	dailyRate	status	passengerCapacity	engineCapacity
1	1	Toyota	Camry	2022	50.00	1	4	1450.00
2	2	Honda	Civic	2023	45.00	1	7	1500.00
3	3	Ford	Focus	2022	48.00	0	4	1400.00
4	4	Nissan	Altima	2023	52.00	1	7	1200.00
5	5	Chevrolet	Malibu	2022	47.00	1	4	1800.00
6	6	Hyundai	Sonata	2023	49.00	0	7	1400.00
7	7	BMW	3 Series	2023	60.00	1	7	2499.00
8	8	Mercedes	C-Class	2022	58.00	1	8	2599.00
9	9	Audi	A4	2022	55.00	0	4	2500.00
10	10	Lexus	ES	2023	54.00	1	4	2500.00

## After Updating the daily rate for a Mercedes car to 68:

```
--1)
UPDATE Vehicle SET dailyRate = 68 WHERE make = 'Mercedes';
```

	vehicleID	make	model	year	dailyRate	status	passengerCapacity	engineCapacity
1	1	Toyota	Camry	2022	50.00	1	4	1450.00
2	2	Honda	Civic	2023	45.00	1	7	1500.00
3	3	Ford	Focus	2022	48.00	0	4	1400.00
4	4	Nissan	Altima	2023	52.00	1	7	1200.00
5	5	Chevrolet	Malibu	2022	47.00	1	4	1800.00
6	6	Hyundai	Sonata	2023	49.00	0	7	1400.00
7	7	BMW	3 Series	2023	60.00	1	7	2499.00
8	8	Mercedes	C-Class	2022	68.00	1	8	2599.00
9	9	Audi	A4	2022	55.00	0	4	2500.00
10	10	Lexus	ES	2023	54.00	1	4	2500.00

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

```
--2)
DELETE FROM Payment WHERE leaseID IN (SELECT leaseID FROM Lease WHERE customerID = 1);
DELETE FROM Lease WHERE customerID = 1;
DELETE FROM Customer WHERE customerID = 1;
```

Deleting customer details from payment table based on their customer and lease ID

	paymentID	leaseID	transactionDate	amount
1	2	2	2023-02-20	1000.00
2	3	3	2023-03-12	75.00
3	4	4	2023-04-25	900.00
4	5	5	2023-05-07	60.00
5	6	6	2023-06-18	1200.00
6	7	7	2023-07-03	40.00
7	8	8	2023-08-14	1100.00
8	9	9	2023-09-09	80.00
9	10	10	2023-10-25	1500.00

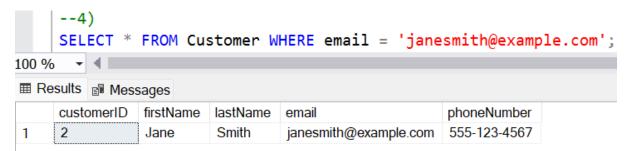
	leaseID	vehicleID	customerID	startDate	endDate	leaseType
1	2	2	2	2023-02-15	2023-02-28	Monthly
2	3	3	3	2023-03-10	2023-03-15	Daily
3	4	4	4	2023-04-20	2023-04-30	Monthly
4	5	5	5	2023-05-05	2023-05-10	Daily
5	6	4	3	2023-06-15	2023-06-30	Monthly
6	7	7	7	2023-07-01	2023-07-10	Daily
7	8	8	8	2023-08-12	2023-08-15	Monthly
8	9	3	3	2023-09-07	2023-09-10	Daily
9	10	10	10	2023-10-10	2023-10-31	Monthly

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

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

	paymentID	leaseID	transactionDate	amount
1	2	2	2023-02-20	1000.00
2	3	3	2023-03-12	75.00
3	4	4	2023-04-25	900.00
4	5	5	2023-05-07	60.00
5	6	6	2023-06-18	1200.00
6	7	7	2023-07-03	40.00
7	8	8	2023-08-14	1100.00
8	9	9	2023-09-09	80.00
9	10	10	2023-10-25	1500.00

# 4. Find a specific customer by email.



#### 5. Get active leases for a specific customer.

```
SELECT * FROM Lease JOIN Vehicle ON Lease.vehicleID=Vehicle.vehicleID

WHERE Lease.customerID = 2 AND Vehicle.status=1;

100 % 

■ Results 
■ Messages

| leaseID vehicleID customerID startDate endDate leaseType vehicleID make model year dailyRate status passengerCapacity engineCapacity
1 2 2 2 2023-02-15 2023-02-28 Monthly 2 Honda Civic 2023 45.00 1 7 1500.00
```

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

```
--6)
SELECT p.* FROM Payment p JOIN Lease 1 ON p.leaseID = 1.leaseID JOIN Customer c ON 1.customerID = c.customerID
WHERE c.phoneNumber = '555-123-4567';

100 % 
Results Messages

paymentID leaseID transactionDate amount
1 2 2 2023-02-20 1000.00
```

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

```
--7)
SELECT AVG(dailyRate) AS avgDailyRate FROM Vehicle WHERE status = 1;

100 % 
Results Messages

avgDailyRate
1 53.714285
```

#### 8. Find the car with the highest daily rate.

```
--8)
SELECT TOP(1)* FROM Vehicle ORDER BY dailyRate DESC;

100 % 
Results Messages

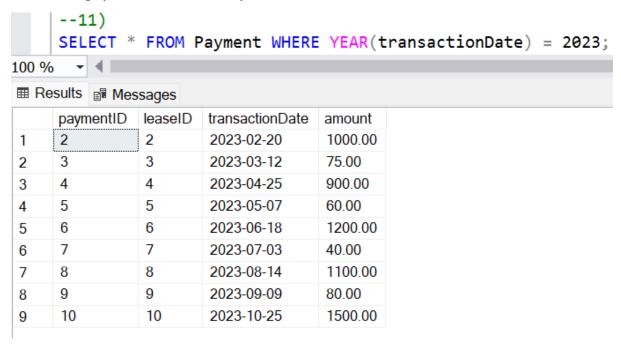
vehicleID make model year dailyRate status passengerCapacity engineCapacity

Mercedes C-Class 2022 68.00 1 8 2599.00
```

#### 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.



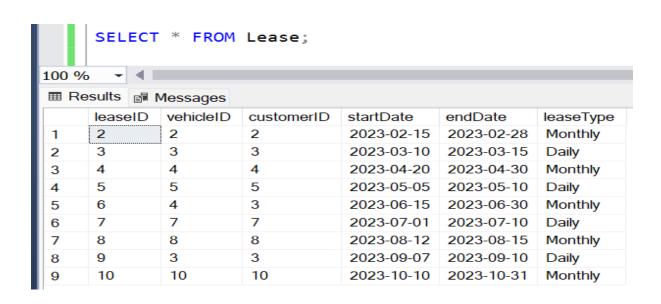
#### 12. Retrieve customers who have not made any payments.

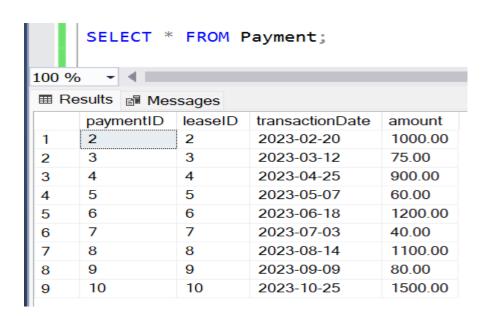
```
--12)
SELECT c.* FROM Customer c JOIN Lease 1 ON 1.customerID=c.customerID WHERE leaseID NOT IN(
SELECT leaseID FROM Payment);

100 %
Results Messages

customerID firstName lastName email phoneNumber
```

The result is empty set because every customer who have taken lease had mad their payments as seeing the following tables:





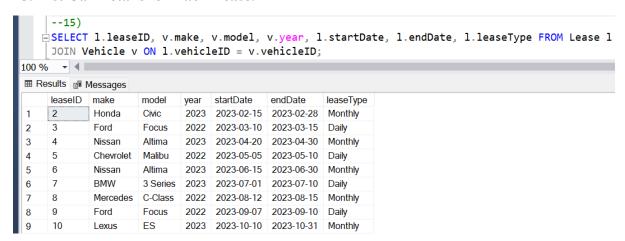
#### 13. Retrieve Car Details and Their Total Payments.

--13) ⊟SELECT v.vehicleID, v.make, v.model, COALESCE(SUM(p.amount), 0) AS TotalPayments FROM Vehicle v LEFT JOIN Lease 1 ON v.vehicleID = 1.vehicleID LEFT JOIN Payment p ON 1.leaseID = p.leaseID GROUP BY v.vehicleID, v.make, v.model; - 4 ■ 100 % vehicleID make model TotalPayments Toyota Camry 0.00 1000.00 Honda Civic 155.00 3 Ford Focus 3 4 4 Nissan Altima 2100.00 5 5 Chevrolet Malibu 60.00 Hyundai Sonata 0.00 6 6 7 BMW 3 Series 40 00 7 8 8 Mercedes C-Class 1100.00 9 Audi A4 0.00 1500.00 10 10 Lexus ES

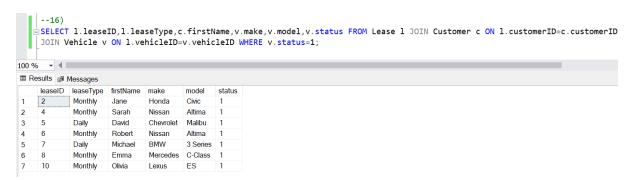
#### 14. Calculate Total Payments for Each Customer.

```
--14)
   SELECT c.customerID, c.firstName, c.lastName, SUM(p.amount) AS TotalPayments FROM Customer c
     LEFT JOIN Lease 1 ON c.customerID = 1.customerID LEFT JOIN Payment p ON 1.leaseID = p.leaseID
     GROUP BY c.customerID, c.firstName, c.lastName;
100 % - 4
customerID firstName lastName TotalPayments
               Jane
                        Smith
                                1000.00
2
               Robert
                        Johnson
                                1355.00
                                900.00
3
     4
               Sarah
                        Brown
     5
               David
                                60.00
4
                        Lee
5
     6
               Laura
                        Hall
                                NULL
6
     7
               Michael
                        Davis
                                40.00
7
               Emma
                        Wilson
                                1100.00
     8
               William
8
                        Taylor
                                NULL
9
     10
               Olivia
                        Adams
                                1500.00
```

#### 15. List Car Details for Each Lease.



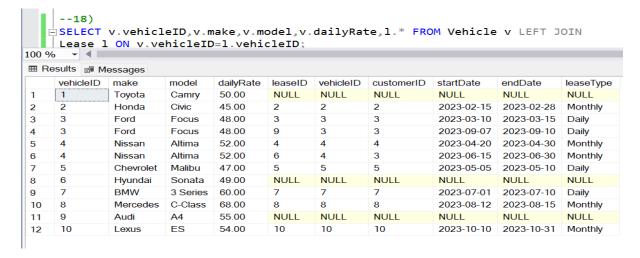
#### 16. Retrieve Details of Active Leases with Customer and Car Information.



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

```
SELECT TOP 1 c.customerID, c.firstName, c.lastName, SUM(p.amount) AS TotalSpent FROM Customer c
    JOIN Lease 1 ON c.customerID = 1.customerID JOIN Payment p ON 1.leaseID = p.leaseID
    GROUP BY c.customerID, c.firstName, c.lastName ORDER BY TotalSpent DESC;
customerID
             firstName lastName
                             TotalSpent
             Olivia
                     Adams
                             1500.00
```

#### 18. List All Cars with Their Current Lease Information.



#### CODE:

```
use coding challenge car rental system;
CREATE TABLE Vehicle (
  vehicleID INT PRIMARY KEY,
  make VARCHAR(50),
  model VARCHAR(50),
  year INT,
  dailyRate DECIMAL(5, 2),
  status TINYINT, -- 1 = available, 0 = notAvailable
  passengerCapacity INT,
  engineCapacity DECIMAL(6, 2)
);
```

```
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,
  leaseType VARCHAR(20), -- 'Daily' or 'Monthly'
  FOREIGN KEY (vehicleID) REFERENCES Vehicle(vehicleID),
  FOREIGN KEY (customerID) REFERENCES Customer(customerID)
);
CREATE TABLE Payment (
  paymentID INT PRIMARY KEY,
  leaseID INT,
  transactionDate DATE,
  amount DECIMAL(10, 2),
  FOREIGN KEY (leaseID) REFERENCES Lease(leaseID)
);
```

INSERT INTO Vehicle (vehicleID, make, model, year, dailyRate, status, passengerCapacity, engineCapacity)

#### **VALUES**

- (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),
- (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);

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');

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

#### **VALUES**

- (1, 1, 1, '2023-01-01', '2023-01-05', 'Daily'),
- (2, 2, 2, '2023-02-15', '2023-02-28', 'Monthly'),

```
(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 (paymentID, leaseID, transactionDate, 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 Vehicle SET dailyRate = 68 WHERE make = 'Mercedes';
--2)
DELETE FROM Payment WHERE leaseID IN (SELECT leaseID FROM Lease WHERE
customerID = 1);
DELETE FROM Lease WHERE customerID = 1;
DELETE FROM Customer WHERE customerID = 1;
```

(3, 3, 3, '2023-03-10', '2023-03-15', 'Daily'),

```
select * from Payment;
--3)
EXEC sp rename 'paymentDate', 'transactionDate', 'COLUMN';
--4)
SELECT * FROM Customer WHERE email = 'janesmith@example.com';
--5)
SELECT * FROM Lease JOIN Vehicle ON Lease.vehicleID=Vehicle.vehicleID
WHERE Lease.customerID = 2 AND Vehicle.status=1;
--6)
SELECT p.* FROM Payment p JOIN Lease 1 ON p.leaseID = 1.leaseID JOIN Customer c ON
1.customerID = c.customerID
WHERE c.phoneNumber = '555-123-4567';
--7)
SELECT AVG(dailyRate) AS avgDailyRate FROM Vehicle WHERE status = 1;
--8)
SELECT TOP(1)* FROM Vehicle ORDER BY dailyRate DESC;
--9)
SELECT * FROM Vehicle WHERE vehicleID IN( SELECT vehicleID FROM Lease
WHERE customerID=(SELECT customerID
FROM Customer WHERE firstName='Sarah'));
--10)
SELECT * FROM Lease WHERE endDate = (SELECT MAX(endDate) FROM Lease);
```

**--11**)

SELECT \* FROM Payment WHERE YEAR(transactionDate) = 2023;

--12)

SELECT c.\* FROM Customer c JOIN Lease 1 ON 1.customerID=c.customerID WHERE leaseID NOT IN(

SELECT leaseID FROM Payment);

SELECT \* FROM Payment;

--13)

SELECT v.vehicleID, v.make, v.model, COALESCE(SUM(p.amount), 0) AS TotalPayments FROM Vehicle v

LEFT JOIN Lease 1 ON v.vehicleID = 1.vehicleID LEFT JOIN Payment p ON 1.leaseID = p.leaseID

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

--14)

SELECT c.customerID, c.firstName, c.lastName, SUM(p.amount) AS TotalPayments FROM Customer c

LEFT JOIN Lease 1 ON c.customerID = 1.customerID LEFT JOIN Payment p ON 1.leaseID = p.leaseID

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

--15)

SELECT l.leaseID, v.make, v.model, v.year, l.startDate, l.endDate, l.leaseType FROM Lease l JOIN Vehicle v ON l.vehicleID = v.vehicleID;

--16)

SELECT l.leaseID,l.leaseType,c.firstName,v.make,v.model,v.status FROM Lease l JOIN Customer c ON l.customerID=c.customerID

JOIN Vehicle v ON l.vehicleID=v.vehicleID WHERE v.status=1;

--17)

SELECT TOP 1 c.customerID, c.firstName, c.lastName, SUM(p.amount) AS TotalSpent FROM Customer c

JOIN Lease I ON c.customerID = l.customerID JOIN Payment p ON l.leaseID = p.leaseID GROUP BY c.customerID, c.firstName, c.lastName ORDER BY TotalSpent DESC;

--18)

SELECT v.vehicleID,v.make,v.model,v.dailyRate,l.\* FROM Vehicle v LEFT JOIN Lease 1 ON v.vehicleID=l.vehicleID;