Ride Zen Car Rental

Car Rental Database Management System

- M.W.K.S.R.Padmakumara

TABLE OF CONTENTS

➢ Case Study

"Ride Zen" is a car rental company which consists of several branches across the country. Each branch has a unique ID, Name, Location and one or two contact numbers. Every branch has cars related to various categories. Each category has its own ID, Category Name, and a small Description. Each car has a unique ID, Model, and Rental Rate. All branches have drivers for each car and any driver can drive many cars. Every driver is registered in a branch with a unique ID, and they have a name and a contact number. Every customer who comes to the reservation can choose a car whether they want it with a driver or without the driver. Customers can make several reservations. Each reservation has a unique ID, Pickup Date, Return Date and Rental Amount. Each reservation may include a car. And each customer has a unique ID, Name, NIC, Address, and contact number. The customer can make the payment to the reservation at once or in several times. All payments have an ID, Date, Time, Cost and Remaining Amount. All cars have damage reports. Each car has multiple damage reports, and it has its own report ID, Date, Time, and Description. Also, each damage report has multiple maintenance logs. It has ID, Maintenance Date, Time, Description, and Cost. Also, all cars are covered by an insurance plan, and it has their ID, Type, and Description.

Assumptions

1) Customer can choose whether they want with driver or without driver,
   I assumed that with driver and without driver as reservation type.

2) customer can make the payment to the reservation at once or in serval times,

   I assumed,

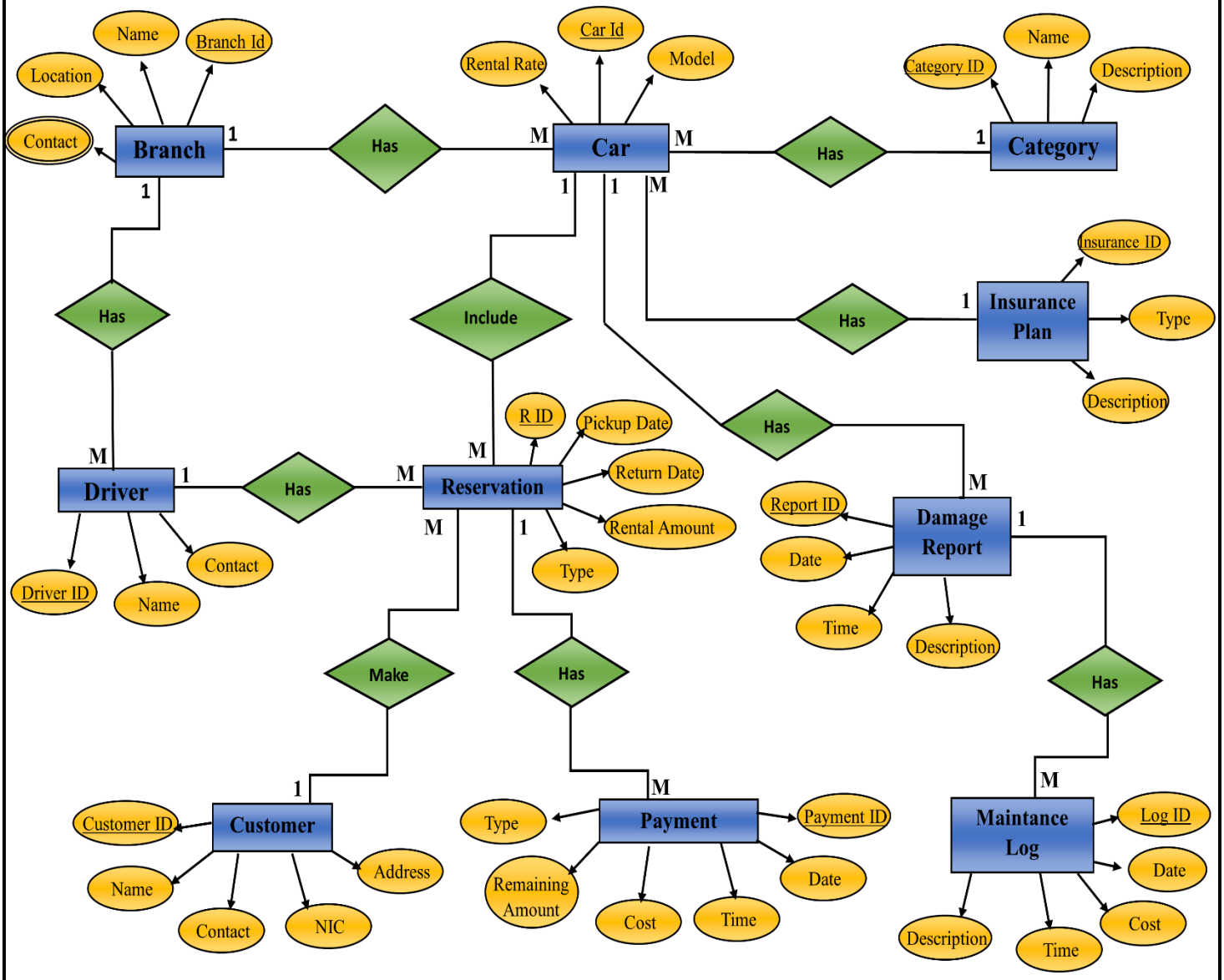
   - payment to the reservation at once: - Full Payment
   - payment to the reservation at serval times: - Down Payments

   I assumed Full Payment and Down Payment as Payment type.

We ourselves have created a YouTube Video about the SQL project of Ride Zen Car Rental Company , by entering the following link you can review it.

Click here

➢ Entity Relationship (ER) diagram



➢ Logical Design

(Convert ER into the Logical Schema)

1) Branch (Branch_ID, Name, Location)

2) Branch_Contact (Branch ID, Contact)

3) Category (Categroy_ID, Name, Description)

4) Insurance_Plan (Insurance_ID, Type, Description)

5) Car (Car_ID, Model, Rental_Rate, Branch_ID, Category_ID, Insurance_ID)

6) Driver (Driver_ID, Name, Contact, Branch_ID)

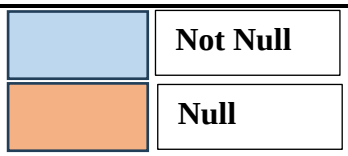
7) Customer (Customer_ID, Name, Address, Contact)

8) Reservation (R_ID, Pickup_Date, Return_Date, Rental Amount, Type, Driver_ID, Customer_ID, Car_ID)

9) Payment (Payment_ID, Type, Cost, Date, Time, Remaining Amount, Reservation_ID)

10) Damage_Report (DReport_ID, Date, Time, Description, Car_ID)

11) Maintance_Log (Log_ID, Date, Time, Description, Cost, Report_ID)



➢ Column Names & Data Types

01) Branch Table

Column Name	Data Type		
Branch_ID	Varchar (10)	PK	Check 'B%'
Name	Varchar (100)		
Location	Varchar (100)		

02) Branch_Contact Table

Column Name	Data Type			
Branch_ID	Varchar (10)	PK	FK (Branch)	Check 'B%'
Contact	Int			

03) Category Table

Column Name	Data Type		
Category_ID	Varchar (10)	PK	Check 'Cat%'
Name	Varchar (50)		
Description	Varchar (500)		

04) Insurance_Plan Table

Column Name	Data Type		
Insurance_ID	Varchar (10)	PK	Check 'I%'
Type	Varchar (100)		
Description	Varchar (500)		

05) Car Table

Column Name	Data Type		
Car_ID	Varchar (10)	PK	Check 'CAR%'
Model	Varchar (50)		
Rental Rate	Decimal (10,2)		
Branch_ID	Varchar (10)	FK (Branch)	Check 'B%'
Category_ID	Varchar (10)	FK (Category)	Check 'Cat%'
Insurance_ID	Varchar (10)	FK (Insurance_Plane)	Check 'I%'

06) Driver Table

Column Name	Data Type		
Driver_ID	Varchar (10)	PK	Check 'D%'
Name	Varchar (100)		
Contact	Int		

07) Customer Table

Column Name	Data Type		
Customer_ID	Varchar (10)	PK	Check 'CU%'
Name	Varchar (100)		
Contact	Int		
NIC	Varchar (12)	Unique	
Address	Varchar(200)		

08) Reservation Table

Column Name	Data Type		
Reservation_ID	Varchar (10)	PK	Check 'R%'
Pickup_Date	Date		
Return_Date	Date		
Rental Amount	Decimal (10,2)		
Type	Varchar (20)		With Driver/Without Driver
Driver_ID	Varchar (10)	FK (Driver)	Check 'D%'
Customer_ID	Varchar (10)	FK (Customer)	Check 'C%'
Car_ID	Varchar (10)	FK (Car)	Check 'CAR%'

09) Payment Table

Column Name	Data Type		
Payment_ID	Varchar (10)	PK	Check 'P%'
Cost	Decimal (10,2)		
Date/Time	Datetime		Get Date ()
Type	Varchar (20)		Full Payment/Down Payment
Remaining_Amount	Decimal (10,2)		
Reservation_ID	Varchar (10)	FK (Reservation)	Check 'R%'

10) Damage_Report Table

Column Name	Data Type		
Report_ID	Varchar (10)	PK	Check 'DR%'
Date	Date		
Time	Time		
Description	Varchar (500)		
Car_ID	Varchar (10)	FK (Car)	Check 'CAR%'

11) Maintance_Log Table

Column Name	Data Type		
Log_ID	Varchar (10)	PK	Check 'ML%'
Date	Date		
Time	Time		
Cost	Decimal(10,2)		
Description	Varchar (500)		
Report_ID	Varchar (10)	FK (Damage_Report)	Check 'DR%'

➢ Physical Design (In MySQL)

❖ CREATE TABLES

1) Create Ride_Zen Database

Create Ride_Zen Database — SQL

```sql
Create Database Ride_Zen
```

2) Create Branch Table

Create Branch Table — SQL

```sql
Create Table Branch(
	Branch_ID Varchar(10) Not Null Primary Key Check(Branch_ID Like 'B%'),
	Name varchar(100) Not Null,
	Location Varchar(100) Not Null
);
```

3) Create Branch_Contact Table

Create Branch_Contact Table — SQL

```sql
Create Table Branch_Contact (
   Branch_ID Varchar(10) Not Null Foreign Key References Branch(Branch_ID) Check(Branch_ID
Like 'B%'),
   Contact int not null Unique,
   Primary Key(Branch_ID,Contact)
);
```

4) Create Category Table

Create Category Table — SQL

```sql
Create Table Category (
	Category_ID Varchar(10) Primary Key Not Null Check (Car_ID Like 'Cat%'),
	Name Varchar(50) Not Null,
	Description Varchar(500) Not Null
);
```

5) Create Insurance_Plan Table

Create Insurance_Plan Table — SQL

```sql
Create Table Insurance_Plan(
	Insurance_ID Varchar(10) Not Null Primary Key Check(Insurance_ID Like 'I%'),
	Type Varchar(100) Not Null,
	Description Varchar(500) Not Null
);
```

6) Create Car Table

Create Car Table — SQL

```
Create Table Car(
 Car_ID Varchar(10) Not Null Primary Key Check(Car_ID Like 'CAR%'),
 Model Varchar(50) Not Null,
 Rental_Rate Decimal(10,2) Not Null,
 Branch_ID Varchar(10) Not Null Foreign Key References Branch(Branch_ID) Check(Branch_ID Like 'B%'),
 Category_ID Varchar(10) Not Null Foreign Key References Category (Category _ID)
       Check(Category_ID Like 'Cat%'),
 Insurance_ID Varchar(10) Not Null Foreign Key References Insurance (Insurance_ID)
       Check(Insurance_ID Like'I%')
);
```

7) Create Driver Table

Create Driver Table — SQL

```
Create Table Driver(
       Driver_ID Varchar(10) Not Null Primary Key Check(Driver_ID Like 'D%'),
       Name Varchar(100) Not Null,
       Contact Int Not Null
);
```

8) Create Customer Table

Create Customer Table — SQL

```
Create Table Customer(
       Customer_ID Varchar(10) Not Null Primary Key Check(Customer_ID Like 'C%'),
       Name Varchar(100) Not Null,
       Contact Int Not Null,
       NIC Varchar(12) Not Null Unique,
       Address Varchar(200) Not Null
);
```

9) Create Reservation Table

Create Reservation Table — SQL

```
Create Table Reservation(
  Reservation_Id Varchar(10) Not Null Primary Key Check(Reservation_ID Like 'R%'),
  Pickup_Date Date Not Null,
  Returan_Date Date Not Null,
  Rental_Amount Decimal(10,2) Not Null,
  Type Varchar(20) Not Null Check(Type IN('With Driver','Without Driver')),
  Driver_ID Varchar(10) Foreign Key References Driver(Driver_ID) Check(Driver_ID Like 'D%'),
  Customer_ID Varchar(10) Not Null Foreign Key ReferencesCustomer(Customer_ID)
       Check(Customer_ID Like 'C%'),
  Car_ID Varchar(10) Not Null Foreign Key References Car(Car_ID) Check(Car_ID Like '%C')
);
```

10) Create Payment Table

Create Payment Table SQL

```
Create Table Payment(
  Payment_ID Varchar(10) Not Null Primary Key Check(Payment_ID Like 'P%'),
  Cost Decimal(10,2) Not Null Check(Cost > 1000),
  Type Varchar(20) Not Null Check(Type IN ('Full Payment','Down Payment')),
  Remaining_Amoun  Decimal(10,2) Not Null,
  Reservation_Id Varchar(10) Not Null Foreign Key References Reservation(Reservation_)
       Check(Reservation_ID Like 'R%'),
  Date Datetime Default Getdate()
);
```

11) Create Damage_Report Table

Create Damage_Report Table SQL

```
Create Table Damage_Report(
       Report_ID Varchar(10) Not Null Primary Key Check(Report_ID Like 'DR%'),
       Date Date Not Null,
       Time Time Not Null,
       Description Varchar(500) Not Null,
       Car_ID Varchar(10) Not Null Foreign Key References Car(Car_ID) Check(Car_ID Like '%C')
);
```

12) Create Maintance_Log Table

Create Maintance_Log Table SQL

```
Create Table Maintance_Log(
   Log_ID Varchar(10) Not Null Primary Key Check(Log_ID Like 'ML%'),
   Date Date Not Null,
   Time Time Not Null,
   Cost Decimal(12,2) Not Null,
   Description Varchar(500) Not Null,
   Report_ID Varchar(10) Not Null Foreign Key References Damage_Report(Report_ID) Check(Report_ID
      Like 'DR%')
);
```

❖ INSERT DATA

01) Insert Data into Branch Table

Insert Data into Branch Table — SQL

```
Insert into Branch(Branch_ID,Name,Location)
      Values ('B001','Colombo Central Branch','Colombo'),
             ('B002','Kandy Main Branch','Kandy'),
             ('B003','Galle South Branch','Galle'),
             ('B004','Jaffna North Branch','Jaffna'),
             ('B005','Mathara Coastal Branch','Mathara'),
             ('B006','Negombo Beach Branch','Negombo'),
             ('B007','Ella Hill Branch','Ella'),
             ('B008','Trincomalee Port Branch','Trinkomalee'),
             ('B009','Anuradhapura Ancient Branch','Anuradhapura'),
             ('B010','Batticaloa East Branch','Batticaloa'),
             ('B011','Polonnaruwa Heritage Branch','Polonnaruwa'),
             ('B012','Kurunegala Western Branch','Kurunegala'),
             ('B013','Hambantota Southern Branch','Hambantota'),
             ('B014','Mannar Lagoon Branch','Mannar'),
             ('B015','Rathnapura Main Branch','Rathnapura'),
             ('B016','Kaluthara Coast Branch','Kaluthara'),
             ('B017','Badulla Uva Branch','Badulla'),
             ('B018','Ampara Beach Branch','Ampara'),
             ('B019','Nuwara Eliya Hill Branch','Nuwara Eliya'),
             ('B020','Vavuniya Northern Branch','Vavuniya')
      ;
```

02) Insert Data into Branch_Contact Table

Insert Data into Branch Table — SQL

```
Insert Into Branch_Contact(Branch_ID,Contact)
      Values ('B001',0914025748),
             ('B001',0774025749),
             ('B002',0914596123),
             ('B002',0774596124),
             ('B003',0912581967),
             ('B003',0772581968),
             ('B004',0914578912),
             ('B004',0774578913),
             ('B005',0911231444),
             ('B005',0771231444),
             ('B006',0915555667),
             ('B006',0775555668),
             ('B007',0914545777),
             ('B007',0774545778),
             ('B008',0918833123),
             ('B008',0778833124),
             ('B009',0914564567),
             ('B009',0774564568),
             ('B010',0911111222),
             ('B010',0771111223),
             ('B011',0777821235),
             ('B012',0914874789),
             ('B012',0774874788),
             ('B013',0914564561),
             ('B013',0774564561),
             ('B014',0914217811),
             ('B014',0774217822),
             ('B015',0911281287),
             ('B015',0771281288),
             ('B016',0919191999),
```

```
        ('B016',0779192000),
        ('B017',0917777888),
        ('B017',0777777889),
        ('B018',0911567899),
        ('B018',0771567900),
        ('B019',0911591590),
        ('B019',0771591591),
        ('B020',0913573577),
        ('B020',0773573588)
    ;
```

03) Insert Data into Category Table

Insert Data into Category Table — SQL

```
Insert Into Category(Category_ID,Name,Description)
    Values ('Cat001', 'Sedan', 'Sedans are passenger cars with a three-box
        configuration, comprising an engine, passenger, and cargo areas.'),
        ('Cat002', 'SUV', 'Sport Utility Vehicles (SUVs) are versatile vehicles with
        off-road capabilities and ample space for passengers and cargo.'),
        ('Cat003', 'Hatchback', 'Hatchbacks are compact cars with a rear door that
        swings upward, offering easy access to the cargo area.'),
        ('Cat004', 'Convertible', 'Convertibles have a retractable roof, providing
        an open-air driving experience.'),
        ('Cat005', 'Coupe', 'Coupes are two-door cars, often sporty in design, with
        a sloping rear roofline.'),
        ('Cat006', 'Minivan', 'Minivans are family-oriented vehicles with ample
        space for passengers and cargo.'),
        ('Cat007', 'Truck', 'Trucks are vehicles designed for hauling cargo and
        often have an open cargo area.'),
        ('Cat008', 'Electric', 'Electric cars are powered by electric motors and
        batteries, reducing reliance on traditional fuel.'),
        ('Cat009', 'Hybrid', 'Hybrid cars combine traditional combustion engines
        with electric motors for increased fuel efficiency.'),
        ('Cat010', 'Luxury', 'Luxury cars offer high-end features, comfort, and
        superior performance.')
    ;
```

04) Insert Data into Insurance_Plan Table

Insert Data into Insurance_Plan Table — SQL

```
Insert Into Insurance_Plan(Insurance_ID,Type,Description)
    VALUES ('I001', 'Comprehensive Coverage', 'Covers damage to your car from incidents
        other than collisions.'),
        ('I002', 'Liability Coverage', 'Covers damage to others' property and medical
        expenses.'),
        ('I003', 'Collision Coverage', 'Covers damage to your car caused by
        collisions with other vehicles or objects.'),
        ('I004', 'Personal Injury Protection', 'Covers medical expenses for you and
        your passengers in case of an accident.')
    ;
```

05) Insert Data to Car Table

Insert Data into Car Table — SQL

```
Insert Into Car(Car_ID,Model,Rental_Rate,Branch_ID,Category_ID,Insurance_ID)
      values ('CAR001','Toyota Camry',1500.00,'B001','Cat001','I001'),
             ('CAR002','Ford Explorer',2000.00,'B001''Cat002''I001'),
             ('CAR003','Volkswagen Golf',4000.00,'B002','Cat003','I003'),
             ('CAR004','Mazda MX-5 Miata',5000.00,'B002','Cat004','I001'),
             ('CAR005','Audi A5',6800.00,'B003','Cat005','I003'),
             ('CAR006','Chrysler Pacifica',3000.00,'B004','Cat006','I004'),
             ('CAR007','Ford F-150',3000.00,'B005','Cat007','I003'),
             ('CAR008','Tesla Model S',9000.00,'B006','Cat008','I001'),
             ('CAR009','Toyata Prius',2000.00,'B007','Cat009','I003'),
             ('CAR010','Mercedes-Benz S-Class',7000.00,'B008','Cat010','I004'),
             ('CAR010', 'Mercedes-Benz S-Class', 7000.00, 'B008', 'Cat010', 'I004'),
             ('CAR012', 'Honda Accord', 4000.00, 'B009', 'Cat001', 'I003'),
             ('CAR013', 'Jeep Grand', 2500.00, 'B010', 'Cat002', 'I004'),
             ('CAR014', 'BMW 4 Sierra', 9500.00, 'B011', 'Cat003', 'I003'),
             ('CAR015', 'Chevrolet Camaro', 4000.00, 'B012', 'Cat004', 'I004'),
             ('CAR016', 'Kia Sedona', 2000.00, 'B013', 'Cat005', 'I001'),
             ('CAR017', 'GMC Sierra', 5000.00, 'B014', 'Cat006', 'I003'),
             ('CAR018', 'Nissan Leaf', 4000.00, 'B015', 'Cat007', 'I002'),
             ('CAR019', 'Lexus ES Hybrid', 3000.00, 'B016', 'Cat008', 'I003'),
             ('CAR020', 'Rolls-Royce Phantom', 1800.00, 'B017', 'Cat009', 'I004'),
             ('CAR021', 'Nissan Altima', 4000.00, 'B018', 'Cat010', 'I003'),
             ('CAR022', 'Subaru Outback', 4000.00, 'B019', 'Cat001', 'I001'),
             ('CAR023', 'Mini Cooper', 4800.00, 'B020', 'Cat002', 'I003'),
             ('CAR024', 'Porsche 911', 2800.00, 'B011', 'Cat003', 'I001'),
             ('CAR025', 'Ford Mustang', 4000.00, 'B002', 'Cat003', 'I003'),
             ('CAR026', 'Chrysler Voyager', 2000.00, 'B003', 'Cat004', 'I004'),
             ('CAR027', 'Chevrolet Silver', 4000.00, 'B004', 'Cat005', 'I003'),
             ('CAR028', 'Tesla Model X', 2600.00, 'B005', 'Cat004', 'I002'),
             ('CAR029', 'Toyota Prius Prime', 6000.00, 'B006', 'Cat005', 'I003'),
             ('CAR030', 'BMW 7 Series', 4000.00, 'B007', 'Cat006', 'I001')
      ;
```

06) Insert Data into Driver Table

Insert Data into Driver Table — SQL

```
Insert into Driver(Driver_ID, Name, Contact)
      Values ('D001', 'Kamal Perera',0711234567),
             ('D002', 'Samantha Silva',0772345678),
             ('D003', 'Nuwan Fernando',0763456789),
             ('D004', 'Chathurika Rajapakse',0704567890),
             ('D005', 'Ranil Jayawardena',0755678901),
             ('D006', 'Malini de Silva',0786789012),
             ('D007', 'Lakmal Gunasekara',0727890123),
             ('D008', 'Tharindu Bandara',0798901234),
             ('D009', 'Anusha Ratnayake',0769012345),
             ('D010', 'Dilshan Perera',0701234567),
             ('D011', 'Chamari Fernando',0712345678),
             ('D012', 'Nishantha Silva',0773456789),
             ('D013', 'Aruna Jayasuriya',0754567890),
             ('D014', 'Madhavi de Mel',0784567891),
             ('D015', 'Kasun Gunawardena',0726789012),
             ('D016', 'Ishara Bandara',0777890123),
             ('D017', 'Nimal Perera',0768901234),
             ('D018', 'Chandani Fernando',0709012345),
             ('D019', 'Roshan de Silva',0711234567),
             ('D020', 'Sanduni Jayawardena',0772345678)
      ;
```

07) Insert Data into Customer Table

Insert Data into Customer Table — SQL

```sql
Insert into Customer (Customer_ID, Name, Contact, NIC, Address)
      Values ('C001', 'Aruna Perera',071123456, '951234567V', '123 Main Street, Colombo'),
             ('C002', 'Kamala Silva',0772345678, '871234567V', '456 Park Avenue, Kandy'),
             ('C003', 'Nimal Rajapakse',0763456789, '921234567V', '789 Oak Lane, Galle'),
             ('C004', 'Chathuri Gunaratne',0704567890, '841234567V', '101 Pine Road, Negombo'),
             ('C005', 'Rajitha Fernando',0755670901, '781234567V', '202 Cedar Street, Jaffna'),
             ('C006', 'Lakshmi de Silva',0786789012, '631234567V', '303 Elm Avenue, Matara'),
             ('C007', 'Tharindu Bandara',0727890123, '541444567V', '404 Birch Lane, Trincomalee'),
             ('C008', 'Anusha Ratnayake',0798901234, '471234567V', '505 Maple Road, Batticaloa'),
             ('C009', 'Dilshan Perera',0769012345, '311234567V', '606 Pine Lane, Anuradhapura'),
             ('C010', 'Chamari Fernando',0701234567, '291234567V', '707 Oak Street, Ratnapura'),
             ('C011', 'Nishantha Silva',0712345678, '381234567V', '808 Cedar Avenue, Badulla'),
             ('C012', 'Aruna Jayasuriya',0773456789, '261234567V','909 Birch Road, Nuwara Eliya'),
             ('C013', 'Madhavi de Mel',0754557890, '181234567V', '1010 Pine Lane, Polonnaruwa'),
             ('C014', 'Kasun Gunawardena',0785678901, '925234567V', '1111 Elm Street, Kalutara'),
             ('C015', 'Ishara Bandara',0797890123, '751234567V', '1212 Oak Avenue, Dambulla'),
             ('C016', 'Nimal Perera',0768901234, '651234567V', '1313 Cedar Road, Hambantota'),
             ('C017', 'Chandani Fernando',0709012345, '541234567V', '1414 Maple Lane, Vavuniya'),
             ('C018', 'Roshan de Silva',0711234567, '981234567V', '1515 Pine Street, Ampara'),
             ('C019', 'Sanduni Jayawardena',0772345678, '881234567V','1616 Elm Road, Kurunegala'),
             ('C020', 'Asanka Rajapakse',0763456789, '761234567V', '1717 Oak Lane, Gampaha')
       ;
```

08) Insert Data into Reservation Table

Insert Data into Reservation Table — SQL

```sql
Insert into Reservation(Reservation_Id,Pickup_Date,Returan_Date,Rental_Amount,Type,Driver_ID,
Customer_ID,Car_ID)
      Values ('R001','2023-12-01','2023-12-05',2500.00,'With Driver',D001','C001','CAR001'),
             ('R002','2023-12-03','2023-12-08',1800.00,'Without Driver',NULL,'C002','CAR002'),
             ('R003','2023-12-10','2023-12-10',3200.00,'With Driver','D002,'C003','CAR003'),
             ('R004','2023-12-10','2023-12-12',4200.00,'Without Driver',NULL,'C004','CAR004'),
             ('R005','2023-12-10','2023-12-15',6000.00,'With Driver','D003','C005','CAR005'),
             ('R006','2023-12-12','2023-12-18',2800.00,'Without Driver',NULL,'C006','CAR006'),
             ('R007','2023-12-13','2023-12-20',3000.00,'With Driver','D004','C007','CAR007'),
             ('R008','2023-12-13','2023-12-24',3200.00,'Without Driver',NULL,'C008','CAR008'),
             ('R009','2023-12-13','2023-12-28',2000.00,'With Driver','D005','C009','CAR009'),
             ('R010','2023-12-22','2023-12-30',4800.00,'Without Driver',NULL,'C010','CAR010')
       ;
```

09) Insert Data into Payment Table

Insert Data into Payment Table — SQL

```sql
Insert into Payment (Payment_ID, Cost, Type, Remaining_Amount, Reservation_Id)
      Values ('P001', 1500.00, 'Full Payment', 0.00, 'R001'),
             ('P002', 1100.00, 'Down Payment', 500.00, 'R002'),
             ('P003', 2500.00, 'Full Payment', 0.00, 'R003'),
             ('P004', 4200.00, 'Full Payment', 0.00, 'R004'),
             ('P005', 6000.00, 'Down Payment', 3000.00, 'R005'),
             ('P006', 2800.00, 'Full Payment', 0.00, 'R006'),
             ('P007', 3000.00, 'Down Payment', 2000.00, 'R007'),
             ('P008', 3200.00, 'Full Payment', 0.00, 'R008'),
             ('P009', 2000.00, 'Down Payment', 1000.00, 'R009'),
             ('P010', 4800.00, 'Full Payment', 0.00, 'R010')
       ;
```

10) Insert Data into Damage Report Table

Insert Data into Damage Report Table — SQL

```sql
Insert into Damage_Report (Report_ID, Date, Time, Description, Car_ID)
      Values ('DR001', '2023-12-02', '08:30:00', 'Scratch on the rear bumper', 'CAR001'),
             ('DR002', '2023-12-05', '10:15:00', 'Dent on the driver-side door', 'CAR002'),
             ('DR003', '2023-12-08', '13:45:00', 'Cracked windshield', 'CAR003'),
             ('DR004', '2023-12-12', '15:20:00', 'Paint chipping on the hood', 'CAR004'),
             ('DR005', '2023-12-15', '11:10:00', 'Broken side mirror', 'CAR005'),
             ('DR006', '2023-12-18', '09:30:00', 'Scuff marks on the front bumper', 'CAR006'),
             ('DR007', '2023-12-22', '14:00:00', 'Dent on the passenger-side door', 'CAR007'),
             ('DR008', '2023-12-25', '12:45:00', 'Scratch on the rear quarter panel', 'CAR008'),
             ('DR009', '2023-12-28', '16:20:00', 'Broken taillight', 'CAR009'),
             ('DR010', '2024-01-02', '09:15:00', 'Dent on the roof', 'CAR010')
      ;
```

11) Insert Data into Maintance Log Table

Insert Data into Maintance Log Table — SQL

```sql
Insert into Maintance_Log (Log_ID, Date, Time, Cost, Description, Report_ID)
      Values ('ML001','2023-12-03','09:00:00',350.00,'Replace rear bumper', 'DR001'),
             ('ML002','2023-12-07','11:30:00',500.00,'Repair driver-side door dent', 'DR002'),
             ('ML003','2023-12-10','14:45:00',700.00,'Replace windshield', 'DR003'),
             ('ML004','2023-12-14','16:20:00',600.00,'Repaint hood', 'DR004'),
             ('ML005','2023-12-17','12:10:00',350.00,'Replace side mirror', 'DR005'),
             ('ML006','2023-12-21','10:30:00',450.00,'Repair front bumper scuff marks','DR006'),
             ('ML007','2023-12-24','15:00:00',600.00,'Repair passenger-side door dent','DR007'),
             ('ML008','2023-12-28','13:45:00',300.00,'Fix rear quarter panel scratch', 'DR008'),
             ('ML009','2024-01-02','17:20:00',400.00,'Replace broken taillight', 'DR009'),
             ('ML010','2024-01-06','09:15:00',550.00,'Repair roof dent', 'DR010')
      ;
```

❖ SIMPLE SELECT QUERIES

Simple SELECT Queries

```
--(1)  Display all data from the Car table
select * from Car;
```

Results | Messages

	Car_ID	Model	Rental_Rate	Branch_ID	Category_ID	Insurance_ID
1	CAR001	Toyota Camry	1500.00	B001	Cat001	I001
2	CAR002	Ford Explorer	2000.00	B001	Cat002	I001
3	CAR003	Volkswagen Golf	4000.00	B002	Cat003	I003
4	CAR004	Mazda MX-5 Miata	5000.00	B002	Cat004	I001
5	CAR005	Audi A5	6800.00	B003	Cat005	I003
6	CAR006	Chrysler Pacifica	3000.00	B004	Cat006	I004
7	CAR007	Ford F-150	3000.00	B005	Cat007	I003
8	CAR008	Tesla Model S	9000.00	B006	Cat008	I001
9	CAR009	Toyota Prius	2000.00	B007	Cat009	I003
10	CAR010	Mercedes-Benz S-Class	7000.00	B008	Cat010	I004
11	CAR012	Honda Accord	4000.00	B009	Cat001	I003
12	CAR013	Jeep Grand	2500.00	B010	Cat002	I004
13	CAR014	BMW 4 Sierra	9500.00	B011	Cat003	I003
14	CAR015	Chevrolet Camaro	4000.00	B012	Cat004	I004
15	CAR016	Kia Sedona	2000.00	B013	Cat005	I001
16	CAR017	GMC Sierra	5000.00	B014	Cat006	I003
17	CAR018	Nissan Leaf	4000.00	B015	Cat007	I002
18	CAR019	Lexus ES Hybrid	3000.00	B016	Cat008	I003
19	CAR020	Rolls-Royce Phantom	1800.00	B017	Cat009	I004
20	CAR021	Nissan Altima	4000.00	B018	Cat010	I003
21	CAR022	Subaru Outback	4000.00	B019	Cat001	I001
22	CAR023	Mini Cooper	4800.00	B020	Cat002	I003
23	CAR024	Porsche 911	2800.00	B011	Cat003	I001
24	CAR025	Ford Mustang	4000.00	B002	Cat003	I003
25	CAR026	Chrysler Voyager	2000.00	B003	Cat004	I004
26	CAR027	Chevrolet Silver	4000.00	B004	Cat005	I003
27	CAR028	Tesla Model X	2600.00	B005	Cat004	I002
28	CAR029	Toyota Prius Prime	6000.00	B006	Cat005	I003
29	CAR030	BMW 7 Series	4000.00	B007	Cat006	I001

Simple SELECT Queries

SQL

```sql
--(2)  Display all data from the Driver Table
select * from Driver;
```

Results | Messages

	Driver_ID	Name	Contact
1	D001	Kamal Perera	711234567
2	D002	Samantha Silva	772345678
3	D003	Nuwan Fernando	763456789
4	D004	Chathurika Raj...	704567890
5	D005	Ranil Jayaward...	755678901
6	D006	Malini de Silva	786789012
7	D007	Lakmal Gunase...	727890123
8	D008	Tharindu Band...	798901234
9	D009	Anusha Ratnay...	769012345
10	D010	Dilshan Perera	701234567
11	D011	Chamari Fernan...	712345678
12	D012	Nishantha Silva	773456789
13	D013	Aruna Jayasuriya	754567890
14	D014	Madhavi de Mel	7845678...
15	D015	Kasun Gunawa...	726789012
16	D016	Ishara Bandara	777890123
17	D017	Nimal Perera	768901234
18	D018	Chandani Fern...	709012345
19	D019	Roshan de Silva	711234567
20	D020	Sanduni Jayaw...	772345678

Simple SELECT Queries

SQL

```sql
--(3)  Display Name, Contact from the Customer Table
select Name,Contact from Customer ;
```

Results | Messages

	Name	Contact
1	Aruna Perera	711234567
2	Kamala Silva	772345678
3	Nimal Raja...	763456789
4	Chathuri G...	704567890
5	Rajitha Fer...	755670901
6	Lakshmi de...	786789012
7	Tharindu B...	727890123
8	Anusha Ra...	798901234
9	Dilshan Per...	769012345
10	Chamari Fe...	701234567
11	Nishantha ...	712345678
12	Aruna Jaya...	773456789

12	Aruna Jaya...	773456789
13	Madhavi d...	754557890
14	Kasun Gun...	785678901
15	Ishara Ban...	797890123
16	Nimal Perera	768901234
17	Chandani F...	709012345
18	Roshan de ...	711234567
19	Sanduni Ja...	772345678
20	Asanka Raj...	763456789

Simple SELECT Queries — SQL

```sql
--(4)  Display Rental Rate less than 2500.00 Car Model and Rental Rate
select Model,Rental_Rate from Car
where Rental_Rate<2500.00 ;
```

Results | Messages

	Model	Rental_Rate
1	Toyota Camry	1500.00
2	Ford Explorer	2000.00
3	Toyota Prius	2000.00
4	Kia Sedona	2000.00
5	Rolls-Royce...	1800.00
6	Chrysler Voy...	2000.00

❖ SELECT QUERIES USING GROUP BY AND HAVING CLAUSE.

SELECT Queries using GROUP BY and HAVING clause — SQL

```sql
--(1)  Display Number of Reservation ,Sum of Rental Amount Each Reservation Type from Reservation
Table
select Type,count(Reservation_ID) AS Reservation,sum(Rental_Amount) AS Income from Reservation
group by  Type;
```

Results | Messages

	Type	Reservation	Income
1	With Driver	5	16700.00
2	Without Driver	5	16800.00

SELECT Queries using GROUP BY and HAVING clause — SQL

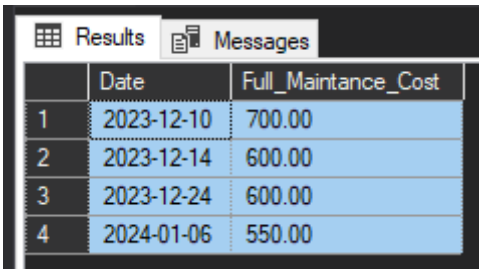
```sql
--(2)  Display number of Reservations ,Sum of Rental amount Each Pickup Date and Sum of rental rate
grater than 3000.00
select  Pickup_Date,count(Reservation_ID) AS Reservation,sum(Rental_Amount) AS Income
from Reservation
group by Pickup_Date
having Sum(Rental_Amount)>3000;
```

Results | Messages

	Pickup_Date	Reservation	Income
1	2023-12-10	3	13400.00
2	2023-12-13	3	8200.00
3	2023-12-22	1	4800.00

SELECT Queries using GROUP BY and HAVING clause

```sql
--(3)  Display Full Maintance Cost for Each Date and cost Grater than 500
select Date,Sum(Cost) AS Full_Maintance_Cost
from Maintance_Log
Group By Date
Having Sum(Cost)>500;
```



Results | Messages

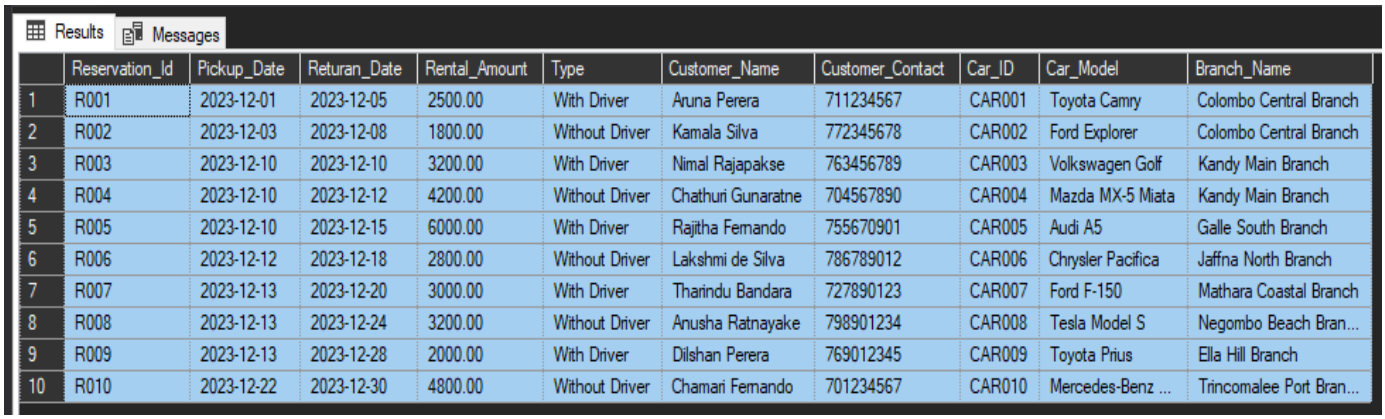
	Date	Full_Maintance_Cost
1	2023-12-10	700.00
2	2023-12-14	600.00
3	2023-12-24	600.00
4	2024-01-06	550.00

- **JOIN RELEVANT TABLES AND DISPLAY DIFFERENT DATA.**

Join relevant Tables and Display Different Data — SQL

```sql
--(1)  Display Reservation ID,Pickup & Return Dates,Rental Amount, Reservation Type, Customer Name
and Contact ,Car Id and Model And Branch Name

SELECT R.Reservation_Id, R.Pickup_Date, R.Returan_Date, R.Rental_Amount, R.Type,
       C.Name AS Customer_Name, C.Contact AS Customer_Contact,
       Ca.Car_ID, Ca.Model AS Car_Model,
       B.Name AS Branch_Name
FROM Reservation R
       INNER JOIN Customer C ON R.Customer_ID = C.Customer_ID
       INNER JOIN Car Ca ON R.Car_ID = Ca.Car_ID
       INNER JOIN Branch B ON Ca.Branch_ID = B.Branch_ID;
```



Results | Messages

	Reservation_Id	Pickup_Date	Returan_Date	Rental_Amount	Type	Customer_Name	Customer_Contact	Car_ID	Car_Model	Branch_Name
1	R001	2023-12-01	2023-12-05	2500.00	With Driver	Aruna Perera	711234567	CAR001	Toyota Camry	Colombo Central Branch
2	R002	2023-12-03	2023-12-08	1800.00	Without Driver	Kamala Silva	772345678	CAR002	Ford Explorer	Colombo Central Branch
3	R003	2023-12-10	2023-12-10	3200.00	With Driver	Nimal Rajapakse	763456789	CAR003	Volkswagen Golf	Kandy Main Branch
4	R004	2023-12-10	2023-12-12	4200.00	Without Driver	Chathuri Gunaratne	704567890	CAR004	Mazda MX-5 Miata	Kandy Main Branch
5	R005	2023-12-10	2023-12-15	6000.00	With Driver	Rajitha Fernando	755670901	CAR005	Audi A5	Galle South Branch
6	R006	2023-12-12	2023-12-18	2800.00	Without Driver	Lakshmi de Silva	786789012	CAR006	Chrysler Pacifica	Jaffna North Branch
7	R007	2023-12-13	2023-12-20	3000.00	With Driver	Tharindu Bandara	727890123	CAR007	Ford F-150	Mathara Coastal Branch
8	R008	2023-12-13	2023-12-24	3200.00	Without Driver	Anusha Ratnayake	798901234	CAR008	Tesla Model S	Negombo Beach Bran...
9	R009	2023-12-13	2023-12-28	2000.00	With Driver	Dilshan Perera	769012345	CAR009	Toyota Prius	Ella Hill Branch
10	R010	2023-12-22	2023-12-30	4800.00	Without Driver	Chamari Fernando	701234567	CAR010	Mercedes-Benz ...	Trincomalee Port Bran...

Join relevant Tables and Display Different Data

SQL

```sql
--(2) Display All payments Cost, Remaining Amount, Payment Date, Reservation Id, Rental Amount,
Customer Name And Customer Contact Number

SELECT P.Payment_ID, P.Cost,P.Remaining_Amount, P.Date AS Payment_Date,
       R.Reservation_Id, R.Rental_Amount,
       C.Name AS Customer_Name, C.Contact AS Customer_Contact
FROM Payment P
       INNER JOIN Reservation R ON P.Reservation_Id = R.Reservation_Id
       INNER JOIN Customer C ON R.Customer_ID = C.Customer_ID;
```

Results | Messages

	Payment_ID	Cost	Remaining_Amount	Payment_Date	Reservation_Id	Rental_Amount	Customer_Name	Customer_Contact
1	P001	1500.00	0.00	2023-12-10 21:21:26.117	R001	2500.00	Aruna Perera	711234567
2	P002	1100.00	500.00	2023-12-10 21:21:26.117	R002	1800.00	Kamala Silva	772345678
3	P003	2500.00	0.00	2023-12-10 21:21:26.117	R003	3200.00	Nimal Rajapakse	763456789
4	P004	4200.00	0.00	2023-12-10 21:21:26.117	R004	4200.00	Chathuri Gunaratne	704567890
5	P005	6000.00	3000.00	2023-12-10 21:21:26.117	R005	6000.00	Rajitha Fernando	755670901
6	P006	2800.00	0.00	2023-12-10 21:21:26.117	R006	2800.00	Lakshmi de Silva	786789012
7	P007	3000.00	2000.00	2023-12-10 21:21:26.117	R007	3000.00	Tharindu Bandara	727890123
8	P008	3200.00	0.00	2023-12-10 21:21:26.117	R008	3200.00	Anusha Ratnayake	798901234
9	P009	2000.00	1000.00	2023-12-10 21:21:26.117	R009	2000.00	Dilshan Perera	769012345
10	P010	4800.00	0.00	2023-12-10 21:21:26.117	R010	4800.00	Chamari Fernando	701234567

Join relevant Tables and Display Different Data

```sql
--(3)  Display All Car ID,Damage Report Id,Damage Date,Damge Description,Maintance Log Id,Maintance
Date and Cost

SELECT C.Car_ID,
       DR.Report_ID, DR.Date AS Damage_Date, DR.Description AS Damage_Description,
       ML.Log_ID, ML.Date AS Maintenance_Date, ML.Cost AS Maintenance_Cost
FROM Damage_Report DR
       INNER JOIN Car C ON DR.Car_ID = C.Car_ID
       INNER JOIN Maintance_Log ML ON DR.Report_ID = ML.Report_ID;
```

Results | Messages

	Car_ID	Report_ID	Damage_Date	Damage_Description	Log_ID	Maintenance_Date	Maintenance_Cost
1	CAR001	DR001	2023-12-02	Scratch on the rear bumper	ML001	2023-12-03	350.00
2	CAR002	DR002	2023-12-05	Dent on the driver-side door	ML002	2023-12-07	500.00
3	CAR003	DR003	2023-12-08	Cracked windshield	ML003	2023-12-10	700.00
4	CAR004	DR004	2023-12-12	Paint chipping on the hood	ML004	2023-12-14	600.00
5	CAR005	DR005	2023-12-15	Broken side mirror	ML005	2023-12-17	350.00
6	CAR006	DR006	2023-12-18	Scuff marks on the front bumper	ML006	2023-12-21	450.00
7	CAR007	DR007	2023-12-22	Dent on the passenger-side door	ML007	2023-12-24	600.00
8	CAR008	DR008	2023-12-25	Scratch on the rear quarter panel	ML008	2023-12-28	300.00
9	CAR009	DR009	2023-12-28	Broken taillight	ML009	2024-01-02	400.00
10	CAR010	DR010	2024-01-02	Dent on the roof	ML010	2024-01-06	550.00

❖ SUB QUERIES

Sub Queries — SQL

```sql
--(1) Display Payment Id,Payment Cost,Remaining Amount Payment Date,Reservation id,Rental
Rate,Customer Name and Contaact in reservations where a driver drives

SELECT P.Payment_ID, P.Cost,P.Remaining_Amount, P.Date AS Payment_Date,
       R.Reservation_Id, R.Rental_Amount,
       C.Name AS Customer_Name, C.Contact AS Customer_Contact
FROM Payment P
       INNER JOIN Reservation R ON P.Reservation_Id = R.Reservation_Id
       INNER JOIN Customer C ON R.Customer_ID = C.Customer_ID
       where Driver_ID in
              (select Driver_ID from Car);
```

Results | Messages

	Payment_ID	Cost	Remaining_Amount	Payment_Date	Reservation_Id	Rental_Amount	Customer_Name	Customer_Contact
1	P001	1500.00	0.00	2023-12-10 21:21:26.117	R001	2500.00	Aruna Perera	711234567
2	P003	2500.00	0.00	2023-12-10 21:21:26.117	R003	3200.00	Nimal Rajapakse	763456789
3	P005	6000.00	3000.00	2023-12-10 21:21:26.117	R005	6000.00	Rajitha Fernando	755670901
4	P007	3000.00	2000.00	2023-12-10 21:21:26.117	R007	3000.00	Tharindu Bandara	727890123
5	P009	2000.00	1000.00	2023-12-10 21:21:26.117	R009	2000.00	Dilshan Perera	769012345

Sub Queries — SQL

```sql
--(2)  Display All Car ID,Damage Report Id,Damage Date,Damge Description,Maintance Log Id,Maintance
Date and Cost in Maintence Cost Grater than 500

SELECT C.Car_ID,
       DR.Report_ID, DR.Date AS Damage_Date, DR.Description AS Damage_Description,
       ML.Log_ID, ML.Date AS Maintenance_Date, ML.Cost AS Maintenance_Cost
FROM Damage_Report DR
       INNER JOIN Car C ON DR.Car_ID = C.Car_ID
       INNER JOIN Maintance_Log ML ON DR.Report_ID = ML.Report_ID
       where Cost in
                     (select Cost from Maintance_Log
                     where Cost>500);
```

Results | Messages

	Car_ID	Report_ID	Damage_Date	Damage_Description	Log_ID	Maintenance_Date	Maintenance_Cost
1	CAR003	DR003	2023-12-08	Cracked windshield	ML003	2023-12-10	700.00
2	CAR004	DR004	2023-12-12	Paint chipping on the hood	ML004	2023-12-14	600.00
3	CAR007	DR007	2023-12-22	Dent on the passenger-side door	ML007	2023-12-24	600.00
4	CAR010	DR010	2024-01-02	Dent on the roof	ML010	2024-01-06	550.00

❖ VIEWS

VIEWS SQL

```sql
--(1)  View to display reservation details with customer information

Create View ReservationDetails AS
	SELECT R.Reservation_Id, R.Pickup_Date, R.Returan_Date, R.Rental_Amount, R.Type,
		C.Customer_ID, C.Name AS Customer_Name, C.Contact AS Customer_Contact,
		Ca.Car_ID, Ca.Model AS Car_Model
	FROM Reservation R
		INNER JOIN Customer C ON R.Customer_ID = C.Customer_ID
		INNER JOIN Car Ca ON R.Car_ID = Ca.Car_ID;

			--view to get reservation details
			SELECT * FROM ReservationDetails;
```

Results Messages

	Reservation_Id	Pickup_Date	Returan_Date	Rental_Amount	Type	Customer_ID	Customer_Name	Customer_Contact	Car_ID	Car_Model
1	R001	2023-12-01	2023-12-05	2500.00	With Driver	C001	Aruna Perera	711234567	CAR001	Toyota Camry
2	R002	2023-12-03	2023-12-08	1800.00	Without Driver	C002	Kamala Silva	772345678	CAR002	Ford Explorer
3	R003	2023-12-10	2023-12-10	3200.00	With Driver	C003	Nimal Rajapakse	763456789	CAR003	Volkswagen Golf
4	R004	2023-12-10	2023-12-12	4200.00	Without Driver	C004	Chathuri Gunaratne	704567890	CAR004	Mazda MX-5 Miata
5	R005	2023-12-10	2023-12-15	6000.00	With Driver	C005	Rajitha Fernando	755670901	CAR005	Audi A5
6	R006	2023-12-12	2023-12-18	2800.00	Without Driver	C006	Lakshmi de Silva	786789012	CAR006	Chrysler Pacifica
7	R007	2023-12-13	2023-12-20	3000.00	With Driver	C007	Tharindu Bandara	727890123	CAR007	Ford F-150
8	R008	2023-12-13	2023-12-24	3200.00	Without Driver	C008	Anusha Ratnayake	798901234	CAR008	Tesla Model S
9	R009	2023-12-13	2023-12-28	2000.00	With Driver	C009	Dilshan Perera	769012345	CAR009	Toyota Prius
10	R010	2023-12-22	2023-12-30	4800.00	Without Driver	C010	Chamari Fernando	701234567	CAR010	Mercedes-Benz S-Class

VIEWS SQL

```sql
--(2)  View to display the total cost of maintenance for each car

CREATE VIEW TotalMaintenanceCostEachCar AS
	SELECT D.Car_ID,C.Model ,SUM(M.Cost) AS TotalCost
	FROM Damage_Report D
		INNER JOIN Maintance_Log M ON D.Report_ID = M.Report_ID
		INNER join Car C ON D.Car_ID=C.Car_ID
	GROUP BY D.Car_ID,C.Model;


			-- view to get the total maintenance cost for each car
			SELECT * FROM TotalMaintenanceCostEachCar;
```

Results Messages

	Car_ID	Model	TotalCost
1	CAR001	Toyota Camry	350.00
2	CAR002	Ford Explorer	500.00
3	CAR003	Volkswagen Golf	700.00
4	CAR004	Mazda MX-5 Miata	600.00
5	CAR005	Audi A5	350.00
6	CAR006	Chrysler Pacifica	450.00
7	CAR007	Ford F-150	600.00
8	CAR008	Tesla Model S	300.00
9	CAR009	Toyota Prius	400.00
10	CAR010	Mercedes-Benz S-Class	550.00

❖ STORED PROCEDURES

Stored Procedures — SQL

```sql
--(1)  Get Today All Reservation ----------------------------------------------------------------

            Create Procedure Today_Reservations
            AS
            Begin
                  Select R.Reservation_Id,R.Car_ID,R.Customer_ID,R.Pickup_Date,R.Type
                  from Reservation R
                  where CONVERT(Date, R.Pickup_Date)=CONVERT(Date,Getdate());
            End;

            --Execute Reservation Procedure
            exec Today_Reservations -- (This Result For This Date :- 2023-12-10)
```

Results | Messages

	Reservation_Id	Car_ID	Customer_ID	Pickup_Date	Type
1	R003	CAR003	C003	2023-12-10	With Driver
2	R004	CAR004	C004	2023-12-10	Without Driver
3	R005	CAR005	C005	2023-12-10	With Driver

12/10/2023

Stored Procedures — SQL

```sql
--(2)  Get Today Full Payments----------------------------------------------------------------

      Create Procedure Today_Full_Payments
      AS
      Begin
            Select P.Payment_ID,P.Cost,P.Reservation_Id
            from Payment P
            Where P.Type='Full Payment' and CONVERT(Date,P.Date) = CONVERT(Date, GETDATE());
      End;

      --Execute Today_Full_Payments Procedure
      exec Today_Full_Payments
```

Results | Messages

	Payment_ID	Cost	Reservation_Id
1	P001	1500.00	R001
2	P003	2500.00	R003
3	P004	4200.00	R004
4	P006	2800.00	R006
5	P008	3200.00	R008
6	P010	4800.00	R010

```sql
--All Payment Info
select * from Payment; -- Upload date(2023-12-10)
```

83 %

Results | Messages

	Payment_ID	Cost	Type	Remaining_Amount	Reservation_Id	Date
1	P001	1500.00	Full Payment	0.00	R001	2023-12-10 22:15:40.800
2	P002	1100.00	Down Payment	500.00	R002	2023-12-10 22:15:40.800
3	P003	2500.00	Full Payment	0.00	R003	2023-12-10 22:15:40.800
4	P004	4200.00	Full Payment	0.00	R004	2023-12-10 22:15:40.800
5	P005	6000.00	Down Payment	3000.00	R005	2023-12-10 22:15:40.800
6	P006	2800.00	Full Payment	0.00	R006	2023-12-10 22:15:40.800
7	P007	3000.00	Down Payment	2000.00	R007	2023-12-10 22:15:40.800
8	P008	3200.00	Full Payment	0.00	R008	2023-12-10 22:15:40.800
9	P009	2000.00	Down Payment	1000.00	R009	2023-12-10 22:15:40.800
10	P010	4800.00	Full Payment	0.00	R010	2023-12-10 22:15:40.800

Stored Procedures

SQL

```sql
--(3)  Get Reservations Made By a specific Customer-----------------------------------------------------


            Create Procedure Get_Customer_Reservations @ID varchar(10)
            AS
            Begin
                  Select R.Reservation_Id,R.Pickup_Date,R.Returan_Date,
                        C.Customer_ID,C.Name AS Customer_Name,
                        Car.Car ID,Car.Model AS Car_Model
                  From Reservation R
                        INNER Join Customer C ON R.Customer_ID = C.Customer_ID
                        INNER Join Car ON R.Car_ID = Car.Car_ID
                  Where C.Customer_ID = @ID;
            End;


--Get all reservations Made BY C002 Customer (Using Procedure)
      exec Get_Customer_Reservations @ID='C002'
```

Results | Messages

	Reservation_Id	Pickup_Date	Returan_Date	Customer_ID	Customer_Name	Car_ID	Car_Model
1	R002	2023-12-03	2023-12-08	C002	Kamala Silva	CAR002	Ford Explorer

Stored Procedures

SQL

```sql
--(4)  Update the rental rate of a specific car-----------------------------------------------------

      --View Currently Rental  Rate CAR001
      select Rental_Rate  From Car
      where Car_ID='CAR001';

      --Procedure
      Create Procedure Update_Car_Rental_Rate @Car_ID varchar(10),@New_Rental_Rate Decimal(10,2)
      AS
      Begin
            Update Car
            set Rental_Rate=@New_Rental_Rate
            Where Car_ID=@Car_ID
      End;

-- Update CAR001 Rantal Rate Is 1600.00 (Using Procedure)
exec Update_Car_Rental_Rate @Car_ID='CAR001',@New_Rental_Rate=1600.00


      --View After Updating Rental  Rate CAR001
      select Rental_Rate  From Car
      where Car_ID='CAR001';
```

Results | Messa

	Rental_Rate
1	1500.00

Query

Results | Mess

	Rental_Rate
1	1600.00

Before **After**

Stored Procedures

SQL

```sql
--(5) Get Payment Infomations by a specific Reservation----------------------------------------


            Create Procedure Payment_Info @Reservation_ID varchar(10)
            AS
            Begin
                  SELECT R.Reservation_Id, R.Rental_Amount,
                        P.Payment_ID, P.Cost,P.Remaining_Amount, P.Date AS Payment_Date,
                        C.Name AS Customer_Name, C.Contact AS Customer_Contact
                  FROM Payment P
                        INNER JOIN Reservation R ON P.Reservation_Id = R.Reservation_Id
                        INNER JOIN Customer C ON R.Customer_ID = C.Customer_ID
                  where R.Reservation_Id=@Reservation_ID;
            End;


      --View R001 Reservation Payment Infomations
      exec Payment_Info @Reservation_ID='R001'
```

Results | Messages

	Reservation_Id	Rental_Amount	Payment_ID	Cost	Remaining_Amount	Payment_Date	Customer_Name	Customer_Contact
1	R001	2500.00	P001	1500.00	0.00	2023-12-10 22:15:40.800	Aruna Perera	711234567