Car Service & Maintenance System - Case Study

Description: A car service center keeps track of customers, their vehicles, service appointments, mechanics, and maintenance tasks. The system records vehicle details, service history, mechanics assigned, parts used, and costs. This helps in managing appointments, billing, and customer satisfaction efficiently.

Database Schema:

1. Customers(CustomerID, Name, Phone, Email, City)
2. Vehicles(VehicleID, CustomerID, Make, Model, Year, LicensePlate)
3. Mechanics(MechanicID, Name, Specialty, ExperienceYears)
4. ServiceAppointments(AppointmentID, VehicleID, MechanicID, ServiceDate, ServiceType, Cost)
5. PartsUsed(PartID, AppointmentID, PartName, Quantity, PartCost)

SQL Table Creation (DDL):

```
-- Customers Table CREATE TABLE Customers ( CustomerID INT PRIMARY KEY, Name
VARCHAR(100), Phone VARCHAR(15), Email VARCHAR(100), City VARCHAR(50) ); -- Vehicles
Table CREATE TABLE Vehicles ( VehicleID INT PRIMARY KEY, CustomerID INT, Make
VARCHAR(50), Model VARCHAR(50), Year INT, LicensePlate VARCHAR(20), FOREIGN KEY
(CustomerID) REFERENCES Customers(CustomerID) ); -- Mechanics Table CREATE TABLE
Mechanics ( MechanicID INT PRIMARY KEY, Name VARCHAR(100), Specialty VARCHAR(50),
ExperienceYears INT ); -- ServiceAppointments Table CREATE TABLE ServiceAppointments
( AppointmentID INT PRIMARY KEY, VehicleID INT, MechanicID INT, ServiceDate DATE,
ServiceType VARCHAR(50), Cost DECIMAL(10,2), FOREIGN KEY (VehicleID) REFERENCES
Vehicles(VehicleID), FOREIGN KEY (MechanicID) REFERENCES Mechanics(MechanicID) ); --
PartsUsed Table CREATE TABLE PartsUsed ( PartID INT PRIMARY KEY, AppointmentID INT,
PartName VARCHAR(100), Quantity INT, PartCost DECIMAL(10,2), FOREIGN KEY
(AppointmentID) REFERENCES ServiceAppointments(AppointmentID) );
```

Sample Insertions (DML):

```
-- Customers INSERT INTO Customers VALUES (1, 'Rahul Sharma', '9876543210',
'rahul@gmail.com', 'Delhi'), (2, 'Priya Verma', '9123456780', 'priya@gmail.com',
'Mumbai'), (3, 'Amit Patel', '9988776655', 'amit@gmail.com', 'Ahmedabad'); --
Vehicles INSERT INTO Vehicles VALUES (101, 1, 'Toyota', 'Corolla', 2018,
'DL10AB1234'), (102, 1, 'Honda', 'City', 2020, 'DL11XY9876'), (103, 2, 'Hyundai',
'i20', 2019, 'MH12CD4567'), (104, 3, 'Ford', 'EcoSport', 2021, 'GJ01EF7890'); --
Mechanics INSERT INTO Mechanics VALUES (201, 'Suresh Kumar', 'Engine Repair', 10),
(202, 'Anil Mehta', 'Electrical', 7), (203, 'Ravi Singh', 'General Service', 5); --
ServiceAppointments INSERT INTO ServiceAppointments VALUES (301, 101, 201,
'2025-09-20', 'Engine Repair', 8000.00), (302, 102, 202, '2025-09-22', 'Electrical
Repair', 2500.00), (303, 103, 203, '2025-09-23', 'General Service', 1500.00), (304,
104, 201, '2025-08-15', 'Engine Repair', 9000.00), (305, 101, 203, '2025-09-25',
'General Service', 2000.00); -- PartsUsed INSERT INTO PartsUsed VALUES (401, 301,
'Engine Oil', 2, 1200.00), (402, 301, 'Air Filter', 1, 500.00), (403, 302,
'Battery', 1, 4000.00), (404, 303, 'Brake Pads', 2, 1500.00), (405, 304, 'Piston
Kit', 1, 7000.00), (406, 305, 'Coolant', 1, 600.00);
```

Queries & Answers:

1. List all customers
SQL: SELECT * FROM Customers;
Answer: Rahul Sharma, Priya Verma, Amit Patel

2. Find all vehicles of CustomerID=1
SQL: SELECT * FROM Vehicles WHERE CustomerID=1;
Answer: Toyota Corolla, Honda City

3. Mechanics specializing in Engine Repair
SQL: SELECT * FROM Mechanics WHERE Specialty='Engine Repair';
Answer: Suresh Kumar

4. Upcoming appointments (next 7 days)
SQL: SELECT * FROM ServiceAppointments WHERE ServiceDate BETWEEN CURRENT_DATE AND CURRENT_DATE + INTERVAL 7 DAY;
Answer: Appointment 305 (Sep 25, 2025)

5. Revenue per mechanic
SQL: SELECT M.Name, SUM(S.Cost) FROM Mechanics M JOIN ServiceAppointments S ON M.MechanicID=S.MechanicID GROUP BY M.Name;
Answer: Suresh Kumar=17000, Anil Mehta=2500, Ravi Singh=3500

6. Vehicles serviced last month
SQL: SELECT V.Make, V.Model FROM Vehicles V JOIN ServiceAppointments S ON V.VehicleID=S.VehicleID WHERE MONTH(S.ServiceDate)=MONTH(CURRENT_DATE - INTERVAL 1 MONTH);
Answer: Ford EcoSport

7. Parts used in appointment 301
SQL: SELECT PartName, Quantity, PartCost FROM PartsUsed WHERE AppointmentID=301;
Answer: Engine Oil, Air Filter

8. Customers with more than one vehicle
SQL: SELECT C.Name, COUNT(V.VehicleID) FROM Customers C JOIN Vehicles V ON C.CustomerID=V.CustomerID GROUP BY C.Name HAVING COUNT(V.VehicleID)>1;
Answer: Rahul Sharma (2 vehicles)

9. Appointments with customer and vehicle
SQL: SELECT S.AppointmentID, C.Name, V.Model, S.ServiceDate FROM ServiceAppointments S JOIN Vehicles V ON S.VehicleID=V.VehicleID JOIN Customers C ON V.CustomerID=C.CustomerID;
Answer: Appointments with details

10. Vehicles not serviced in last year
SQL: SELECT V.Make, V.Model FROM Vehicles V WHERE V.VehicleID NOT IN (SELECT VehicleID FROM ServiceAppointments WHERE ServiceDate >= CURRENT_DATE - INTERVAL 1 YEAR);
Answer: None

11. Total cost per appointment
SQL: SELECT S.AppointmentID, S.Cost+IFNULL(SUM(P.PartCost*P.Quantity),0) FROM ServiceAppointments S LEFT JOIN PartsUsed P ON S.AppointmentID=P.AppointmentID GROUP BY S.AppointmentID, S.Cost;
Answer: 301=10700, 304=16000, etc.

12. Mechanics with more than 10 services
SQL: SELECT M.Name, COUNT(S.AppointmentID) FROM Mechanics M JOIN ServiceAppointments S ON M.MechanicID=S.MechanicID GROUP BY M.Name HAVING COUNT(S.AppointmentID)>10;
Answer: None

13. Most frequently used parts
SQL: SELECT PartName, COUNT(*) FROM PartsUsed GROUP BY PartName ORDER BY COUNT(*) DESC LIMIT 1;
Answer: Brake Pads

14. Appointments costing more than 1000
SQL: SELECT * FROM ServiceAppointments WHERE Cost>1000;

Answer: All except appointment 303

15. Customers with number of services
SQL: SELECT C.Name, COUNT(S.AppointmentID) FROM Customers C JOIN Vehicles V ON C.CustomerID=V.CustomerID JOIN ServiceAppointments S ON V.VehicleID=S.VehicleID GROUP BY C.Name;
Answer: Rahul Sharma=3, Priya Verma=1, Amit Patel=1