Topic of submission:

DATABASE MANAGEMENT SYSTEM FOR

CAR SHOWROOM

DBMS PROJECT REPORT

Submitted by:

SHIVAM KHURANA	102103754
ROHAN THAKUR	102103762
PRABHMEET KAUR	102103785

Submitted to:
DR. ARCHANA SINGH

INDEX

Introduction to Database Management System For Car Showroom

Car Showroom Management System is a project that is used to manage and control the complete record of Cars. This mini project is to present the record of Cars, Customers and corporations. This Car Showroom Management System has the track of all the cars with every single detail. DBMS can help to organize and manage data related to customers, cars, employees, and transactions.

Benefits of using a DBMS for Car Showroom:

There are several benefits of using a DBMS (database management system) for a car showroom containing attributes such as employee, customer, car, service, manufacturers, and sales:

Data Integrity: A DBMS ensures that data is accurate, consistent, and up-to-date. It enforces data constraints, such as data types and foreign key relationships, preventing data from being corrupted.

Efficient Data Retrieval: A DBMS can retrieve data quickly and efficiently, even when dealing with large amounts of data. It can use indexing, caching, and other optimization techniques to speed up queries and reduce data retrieval times.

Improved Security: A DBMS provides built-in security features, such as access controls, authentication, and encryption, which help to prevent unauthorized access and data breaches.

Simplified Data Management: A DBMS provides a centralized location for storing and managing data, making it easier to maintain, backup, and restore data.

Scalability: A DBMS can handle large amounts of data and can scale up or down as needed to accommodate changes in data volume.

Data Consistency: A DBMS ensures that data is consistent across multiple tables and databases, ensuring that data is not duplicated or inconsistent.

Better Decision Making: A DBMS can provide data analytics and visualization tools, which can help decision-makers gain insights into the business, identify trends, and make informed decisions based on data.

Choosing the right DBMS for Car Showroom:

The type of DBMS that would be suitable for a car showroom with the attributes mentioned (employee, customer, car, service, manufacturers, sales) would be a Relational Database Management System (RDBMS).

An RDBMS provides a structured way to store and manage data in tables with relationships defined between them. This allows for efficient querying, sorting, and filtering of data based on various parameters. Additionally, RDBMSs provide tools for enforcing data integrity and consistency, such as primary and foreign key constraints, which ensure that data is entered correctly and linked properly between tables.

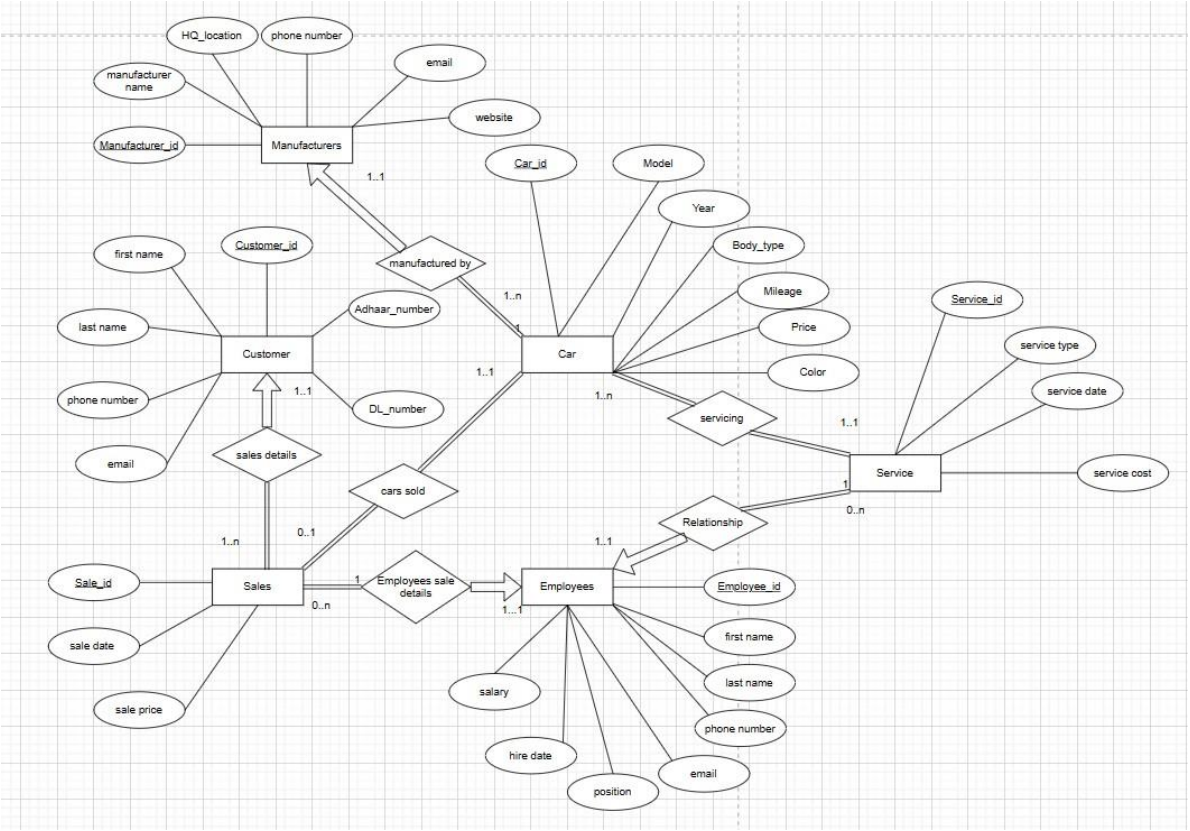
Some popular examples of RDBMSs include Oracle, MySQL, Microsoft SQL Server, and PostgreSQL.

Implementing a DBMS for Car Showroom:

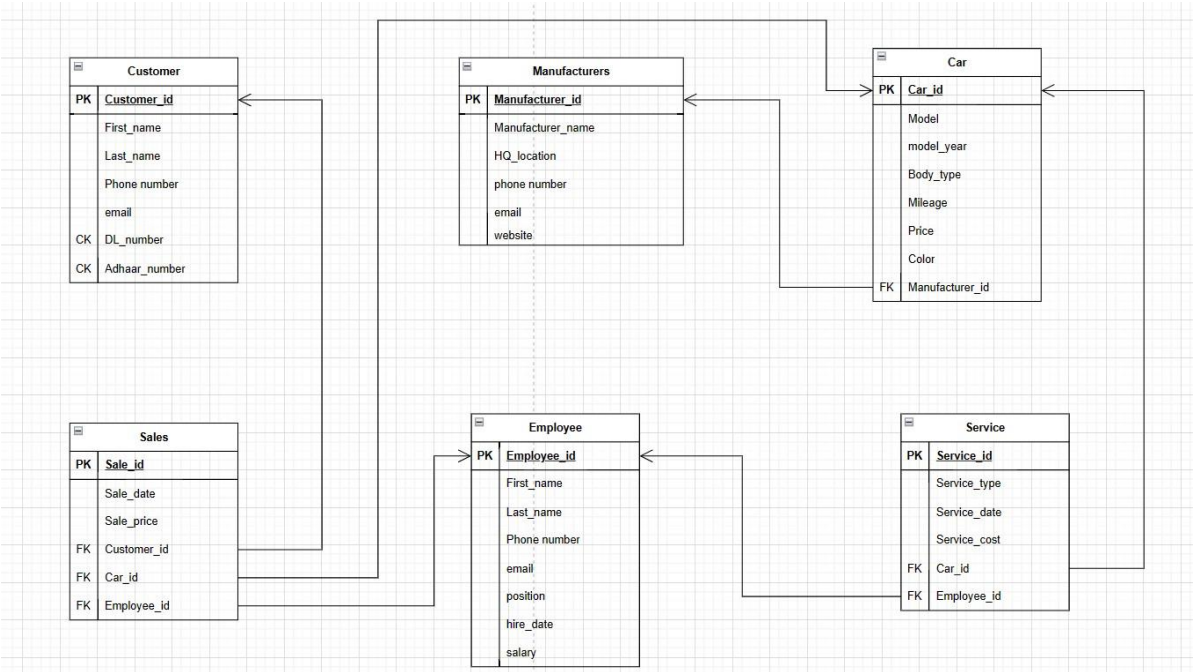
The implementation of a DBMS (database management system) for a car showroom can bring significant benefits to the business in terms of improved data management, increased efficiency, and better decision-making. Here are the key steps involved in implementing a DBMS for a car showroom:

1. **Define Data Requirements**: The first step in implementing a DBMS for a car showroom is to define the data requirements of the business. This involves identifying the types of data that need to be stored and managed, such as customer data, sales data, inventory data, and vehicle information.
2. **Design the Database Schema**: Once the data requirements have been identified, the next step is to design the database schema. This involves creating a blueprint of the database structure, including tables, columns, and relationships between them. The design should be optimized for efficient data storage and retrieval.
3. **Select the DBMS**: There are several DBMS options available, including MySQL, Oracle, and Microsoft SQL Server. The choice of DBMS will depend on the specific needs of the car showroom, such as budget, scalability, and ease of use.
4. I**nstall and Configure the DBMS**: Once the DBMS has been selected, it must be installed and configured according to the requirements of the car showroom. This includes setting up user accounts, access controls, and backup and recovery procedures.

ER DIAGRAM


HQ_location
phone number
email
manufacturer name
website
Manufacturer_id
Manufacturers
1..1
manufactured by
1..n
1
Car_id
Model
Year
Body_type
Mileage
Price
Color
Car
1..1
1..n
Customer_id
first name
Adhaar_number
last name
Customer
phone number
DL_number
email
1..1
sales details
cars sold
1..n
0..1
Sale_id
Sales
1
0..n
Employees sale details
1...1
Employees
Employee_id
first name
last name
phone number
email
position
hire date
salary
Relationship
1..1
servicing
1..1
Service_id
service type
service date
Service
service cost
1
0..n
sale date
sale price

ER TO TABLES


Customer
PK Customer_id
First_name
Last_name
Phone number
email
CK DL_number
CK Adhaar_number
Manufacturers
PK Manufacturer_id
Manufacturer_name
HQ_location
phone number
email
website
Car
PK Car_id
Model
model_year
Body_type
Mileage
Price
Color
FK Manufacturer_id
Sales
PK Sale_id
Sale_date
Sale_price
FK Customer_id
FK Car_id
FK Employee_id
Employee
PK Employee_id
First_name
Last_name
Phone number
email
position
hire_date
salary
Service
PK Service_id
Service_type
Service_date
Service_cost
FK Car_id
FK Employee_id

Normalization

- **Customer table(Customer_id,First_name,last_name,Phone number,email,DL_number,Adhaar_number)**
  Customer_id is number and primary key
  First_name , Last_name , email , Phone number , DL_number , Adhaar_number are varchar

  Already in 3NF

- **Car table(Car_id, Model , model_year , Body_type , Mileage , Price, Color , Manufacturer_id)**
  Car_id is number and primary key
  Manufacturer_id is number and foreign key referenced from Manufactures table(Manufacturer_id)
  Model , Body_type , Color are varchar
  Model_year , Mileage , Price are number

  Already in 3NF

- **Manufactures table(Manufacturer_id ,Manufacturer_name , HQ_location , phone_number, email , website)**
  Manufacturer_id is number and primary key
  Manufacturer_name , HQ_location , email, website ,phone_number are varchar

  Already in 3NF

- **Employee table(employee_id, first_name, Last_name, Phone number, email, position, hire_date, salary)**
  Employee_id is number and primary key

First_name , Last_name , Phone number , email , position are varchar
Hire_date is DATE type
Salary is number
Already in 3NF

- **Sales table(sale_id,sale_date,sale_price)**
  sale_id is primary key
  sale_date is DATE type
  Sale_price are number type

  Already in 3NF

- **Service table(service_id,service_type,service_date,service_cost)**
  Service_id is number primary key
  Service_type is varchar
  Service_date is DATE type
  Service_cost is number

  Already in 3NF

SQL Implementation

- **Creation of tables**

```
CREATE TABLE customer (
  customer_id NUMBER(10) PRIMARY KEY,
  first_name VARCHAR2(50),
  last_name VARCHAR2(50),
  phone_number VARCHAR2(20),
  email VARCHAR2(100),
  aadhar_number VARCHAR2(12),
  driving_license VARCHAR2(20)
);


CREATE TABLE manufacturers (
  manufacturer_id NUMBER(10) PRIMARY KEY,
  manufacturer_name VARCHAR2(100),
  hq_location VARCHAR2(100),
  phone_number VARCHAR2(20),
  email VARCHAR2(100),
  website VARCHAR2(100)
);

CREATE TABLE car (
  car_id NUMBER(10) PRIMARY KEY,
  model VARCHAR2(50),
  model_year NUMBER(4),
  body_type VARCHAR2(50),
  mileage NUMBER(10),
  price NUMBER(12,  2),
  color VARCHAR2(20),
  manufacturer_id NUMBER(10) REFERENCES
manufacturers(manufacturer_id) ON DELETE SET NULL

);

CREATE TABLE employee (
```

```
  employee_id NUMBER(10) PRIMARY KEY,
  first_name VARCHAR2(50),
  last_name VARCHAR2(50),
  phone_number VARCHAR2(20),
  email VARCHAR2(100),
  position VARCHAR2(100),
  hire_date DATE,
  salary NUMBER(10,2)
);




CREATE TABLE sales (
  sale_id NUMBER(10) PRIMARY KEY,
  sale_date DATE,
  sale_price NUMBER(10,2),
  customer_id NUMBER(10),
  car_id NUMBER(10),
  employee_id NUMBER(10),
  CONSTRAINT fk_customer_id FOREIGN KEY (customer_id)
REFERENCES customer(customer_id) ON DELETE SET NULL,
CONSTRAINT fk_car_id FOREIGN KEY (car_id) REFERENCES car(car_id)
ON DELETE SET NULL,
  CONSTRAINT fk_employee_id FOREIGN KEY (employee_id)
REFERENCES employee(employee_id) ON DELETE SET NULL
);


CREATE TABLE service (
  service_id NUMBER(10) PRIMARY KEY,
  service_type VARCHAR2(100),
  service_date DATE,
  service_cost NUMBER(10,2),
  car_id NUMBER(10),
  employee_id NUMBER(10),
  CONSTRAINT fk_carr_id FOREIGN KEY (car_id) REFERENCES
car(car_id) ON DELETE SET NULL,
  CONSTRAINT fk_employeee_id FOREIGN KEY (employee_id)
REFERENCES employee(employee_id) ON DELETE SET NULL
);
```

- **Insertion into tables**

```
INSERT INTO customer VALUES (1, 'John', 'Doe', '1234567890', 'john.doe@email.com', '123456789012', 'DL12345');
INSERT INTO customer VALUES (2, 'Jane', 'Doe', '0987654321', 'jane.doe@email.com', '987654321012', 'DL67890');
INSERT INTO customer VALUES (3, 'Bob', 'Smith', '5551234567', 'bob.smith@email.com', '555123456789', 'DL13579');
INSERT INTO customer VALUES (4, 'Alice', 'Jones', '5559876543', 'alice.jones@email.com', '555987654321', 'DL24680');
INSERT INTO customer VALUES (5, 'Charlie', 'Brown', '5555555555', 'charlie.brown@email.com', '555555555555', 'DL36912');
INSERT INTO customer VALUES (6, 'Lisa', 'Davis', '5554443333', 'lisa.davis@email.com', '555444333222', 'DL97531');
INSERT INTO customer VALUES (7, 'Mike', 'Williams', '5556667777', 'mike.williams@email.com', '555666777888', 'DL86420');
INSERT INTO customer VALUES (8, 'Karen', 'Wilson', '5557778888', 'karen.wilson@email.com', '555777888999', 'DL12367');
INSERT INTO customer VALUES (9, 'David', 'Brown', '5558889999', 'david.brown@email.com', '555888999777', 'DL86429');
INSERT INTO customer VALUES (10, 'Emily', 'Johnson', '5552223333', 'emily.johnson@email.com', '555222333444', 'DL75391');
INSERT INTO customer VALUES (11, 'James', 'Taylor', '5551112222', 'james.taylor@email.com', '555111222333', 'DL97531');
INSERT INTO customer VALUES (12, 'Samantha', 'Harris', '5553334444', 'samantha.harris@email.com', '555333444555', 'DL36912');
INSERT INTO customer VALUES (13, 'Richard', 'Lee', '5554445555', 'richard.lee@email.com', '555444555666', 'DL24680');
INSERT INTO customer VALUES (14, 'Mary', 'Jackson', '5556667777', 'mary.jackson@email.com', '555666777888', 'DL13579');
INSERT INTO customer VALUES (15, 'Tom', 'Miller', '5557778888', 'tom.miller@email.com', '555777888999', 'DL97531');
INSERT INTO customer VALUES (16, 'Kelly', 'Martin', '5558889999', 'kelly.martin@email.com', '555888999777', 'DL12345');
INSERT INTO customer VALUES (17, 'Brian', 'Thompson', '5552223333', 'brian.thompson@email.com', '555222333444', 'DL86420');
INSERT INTO customer VALUES (18, 'Megan', 'Clark', '5551112222', 'megan.clark@email.com', '555111222333', 'DL75391');
```

```
INSERT INTO customer VALUES (19, 'Eric', 'Hall', '5553334444', 'eric.hall@email.com', '555333444555', 'DL24680');
INSERT INTO customer VALUES (20, 'Natalie', 'Young', '5554445555', 'natalie.young@email.com', '555444555666', 'DL36912');
```

```
INSERT INTO manufacturers VALUES (1, 'Toyota', 'Japan', '+81-3-3817-7111', 'info@toyota.com', 'https://www.toyota-global.com/');
INSERT INTO manufacturers VALUES (2, 'Honda', 'Japan', '+81-3-3423-1111', 'info@honda.com', 'https://global.honda/');
INSERT INTO manufacturers VALUES (3, 'Nissan', 'Japan', '+81-45-523-5523', 'info@nissan.com', 'https://www.nissan-global.com/EN/index.html');
INSERT INTO manufacturers VALUES (4, 'Ford', 'United States', '+1-800-392-3673', 'fordteam@ford.com', 'https://www.ford.com/');
INSERT INTO manufacturers VALUES (5, 'Chevrolet', 'United States', '+1-800-222-1020', 'chevrolet@gm.com', 'https://www.chevrolet.com/');
INSERT INTO manufacturers VALUES (6, 'Volkswagen', 'Germany', '+49-5361-9-0', 'info@volkswagen.de', 'https://www.volkswagen.de/');
INSERT INTO manufacturers VALUES (7, 'Mercedes-Benz', 'Germany', '+49-711-17-0', 'dialog@daimler.com', 'https://www.mercedes-benz.com/');
INSERT INTO manufacturers VALUES (8, 'BMW', 'Germany', '+49-89-1250-16000', 'bmw@bmwgroup.com', 'https://www.bmw.com/');
INSERT INTO manufacturers VALUES (9, 'Audi', 'Germany', '+49-841-89-0', 'service@audi.de', 'https://www.audi.com/en.html');
INSERT INTO manufacturers VALUES (10, 'Hyundai', 'South Korea', '+82-2-3464-1114', 'hyundai_motor@hyundai.com', 'https://www.hyundai.com/');
INSERT INTO manufacturers VALUES (11, 'Kia', 'South Korea', '+82-2-3464-1114', 'kia_customer@kia.com', 'https://www.kia.com/');
INSERT INTO manufacturers VALUES (12, 'Mazda', 'Japan', '+81-3-3508-1111', 'info@mazda.com', 'https://www.mazda.com/');
INSERT INTO manufacturers VALUES (13, 'Subaru', 'Japan', '+81-3-3477-8000', 'info@subaru.jp', 'https://www.subaru.jp/');
INSERT INTO manufacturers VALUES (14, 'Lexus', 'Japan', '+81-50-3786-0890', 'contact@lexus-int.com', 'https://www.lexus-int.com/');
INSERT INTO manufacturers VALUES (15, 'Porsche', 'Germany', '+49-711-911-0', 'info@porsche.de', 'https://www.porsche.com/');
INSERT INTO manufacturers VALUES (16, 'Tesla', 'United States', '+1-888-518-3752', 'support@tesla.com', 'https://www.tesla.com/');
```

```
INSERT INTO manufacturers VALUES (17, 'Volvo', 'Sweden', '+46-31-3250000', 'info@volvocars.com', 'https://www.volvocars.com/');
INSERT INTO manufacturers VALUES (18, 'Tesla', 'United States', '+1-650-681-5100', 'info@tesla.com', 'https://www.tesla.com/');
INSERT INTO manufacturers VALUES (19, 'Ferrari', 'Italy', '+39-0536-949111', 'info@ferrari.com', 'https://www.ferrari.com/');
INSERT INTO manufacturers VALUES (20, 'Maserati', 'Italy', '+39-0525-551111', 'info@maserati.com', 'https://www.maserati.com/it/en');
```

```
INSERT INTO car VALUES (1, 'Camry', 2022, 'Sedan', 17.5, 24999.99, 'White', 1);
INSERT INTO car VALUES (2, 'Civic', 2022, 'Sedan', 19.5, 21999.99, 'Black', 2);
INSERT INTO car VALUES (3, 'Altima', 2022, 'Sedan', 18.5, 22999.99, 'Gray', 3);
INSERT INTO car VALUES (4, 'Explorer', 2022, 'SUV', 17.4, 37999.99, 'Blue', 4);
INSERT INTO car VALUES (5, 'Equinox', 2022, 'SUV', 21.5, 31999.99, 'Silver', 5);
INSERT INTO car VALUES (6, 'Tiguan', 2022, 'SUV', 16.5, 30999.99, 'Red', 6);
INSERT INTO car VALUES (7, 'C-Class', 2022, 'Sedan', 14.6, 44999.99, 'Black', 7);
INSERT INTO car VALUES (8, '5 Series', 2022, 'Sedan', 15.4, 53999.99, 'White', 8);
INSERT INTO car VALUES (9, 'A6', 2022, 'Sedan', 16, 51999.99, 'Gray', 9);
INSERT INTO car VALUES (10, 'Sonata', 2022, 'Sedan', 20.5, 23999.99, 'Blue', 10);
INSERT INTO car VALUES (11, 'Optima', 2022, 'Sedan', 19.7, 22999.99, 'Red', 11);
INSERT INTO car VALUES (12, 'Mazda6', 2022, 'Sedan', 18.6, 25999.99, 'White', 12);
INSERT INTO car VALUES (13, 'Legacy', 2022, 'Sedan', 17.4, 27999.99, 'Black', 13);
INSERT INTO car VALUES (14, 'ES', 2022, 'Sedan', 18.6, 49999.99, 'Gray', 14);
INSERT INTO car VALUES (15, 'Panamera', 2022, 'Sedan', 11.3, 85999.99, 'White', 15);
INSERT INTO car VALUES (16, 'Model S', 2022, 'Sedan', 25, 79999.99, 'Red', 16);
```

```
INSERT INTO car VALUES (17, 'XC90', 2022, 'SUV', 14.5, 59999.99, 'Black', 17);
INSERT INTO car VALUES (18, 'Model X', 2022, 'SUV', 26, 99999.99, 'White', 16);
INSERT INTO car VALUES (19, 'Portofino', 2022, 'Convertible', 10.3, 259999.99, 'Red', 19);
INSERT INTO car VALUES (20, 'GranTurismo', 2022, 'Coupe', 10.4, 149999.99, 'Blue', 20);


INSERT INTO employee VALUES (1, 'Amit', 'Sharma', '+91-9876543210', 'amit.sharma@email.com', 'Sales Executive', TO_DATE('2022-01-01', 'YYYY-MM-DD'), 25000);
INSERT INTO employee VALUES(2, 'Rahul', 'Gupta', '+91-9876543211', 'rahul.gupta@email.com', 'Sales Executive', TO_DATE('2022-01-01', 'YYYY-MM-DD'), 25000);
INSERT INTO employee VALUES(3, 'Priya', 'Singh', '+91-9876543212', 'priya.singh@email.com', 'Sales Executive', TO_DATE('2022-01-01', 'YYYY-MM-DD'), 25000);
INSERT INTO employee VALUES(4, 'Suresh', 'Kumar', '+91-9876543213', 'suresh.kumar@email.com', 'Sales Manager', TO_DATE('2022-01-01', 'YYYY-MM-DD'), 40000);
INSERT INTO employee VALUES(5, 'Neha', 'Patil', '+91-9876543214', 'neha.patil@email.com', 'Sales Manager', TO_DATE('2022-01-01', 'YYYY-MM-DD'), 40000);
INSERT INTO employee VALUES(6, 'Sarika', 'Joshi', '+91-9876543215', 'sarika.joshi@email.com', 'Marketing Manager', TO_DATE('2022-01-01', 'YYYY-MM-DD'), 45000);
INSERT INTO employee VALUES(7, 'Alok', 'Mishra', '+91-9876543216', 'alok.mishra@email.com', 'Marketing Executive', TO_DATE('2022-01-01', 'YYYY-MM-DD'), 30000);
INSERT INTO employee VALUES(8, 'Anjali', 'Shah', '+91-9876543217', 'anjali.shah@email.com', 'Marketing Executive', TO_DATE('2022-01-01', 'YYYY-MM-DD'), 30000);
INSERT INTO employee VALUES(9, 'Manish', 'Kapoor', '+91-9876543218', 'manish.kapoor@email.com', 'Finance Manager', TO_DATE('2022-01-01', 'YYYY-MM-DD'), 50000);
INSERT INTO employee VALUES(10, 'Neetu', 'Chopra', '+91-9876543219', 'neetu.chopra@email.com', 'Finance Executive', TO_DATE('2022-01-01', 'YYYY-MM-DD'), 35000);
INSERT INTO employee VALUES(11, 'Vikas', 'Gupta', '+91-9876543220', 'vikas.gupta@email.com', 'Service Manager', TO_DATE('2022-01-01', 'YYYY-MM-DD'), 45000);
```

```
INSERT INTO employee VALUES(12, 'Manoj', 'Singh', '+91-9876543221',
'manoj.singh@email.com', 'Service Executive', TO_DATE('2022-01-01',
'YYYY-MM-DD'), 30000);
INSERT INTO employee VALUES(13, 'Poonam', 'Verma', '+91-9876543222',
'poonam.verma@email.com', 'Service Executive', TO_DATE('2022-01-01',
'YYYY-MM-DD'), 30000);
INSERT INTO employee VALUES(14, 'Rohan', 'Patel', '+91-9876543230',
'rohan.patel@email.com', 'Sales Executive', TO_DATE('2022-01-01',
'YYYY-MM-DD'), 25000);
INSERT INTO employee VALUES(15, 'Rajesh', 'Mishra', '+91-9876543224',
'rajesh.mishra@email.com', 'IT Executive', TO_DATE('2022-01-01',
'YYYY-MM-DD'), 30000);
INSERT INTO employee VALUES(16, 'Kavita', 'Sharma', '+91-9876543225',
'kavita.sharma@email.com', 'HR Manager', TO_DATE('2022-01-01',
'YYYY-MM-DD'), 45000);
INSERT INTO employee VALUES(17, 'Prakash', 'Jha', '+91-9876543226',
'prakash.jha@email.com', 'HR Executive', TO_DATE('2022-01-01',
'YYYY-MM-DD'), 30000);
INSERT INTO employee VALUES(18, 'Aparna', 'Nair', '+91-9876543227',
'aparna.nair@email.com', 'Inventory Manager', TO_DATE('2022-01-01',
'YYYY-MM-DD'), 40000);
INSERT INTO employee VALUES(19, 'Ramesh', 'Menon', '+91-9876543228',
'ramesh.menon@email.com', 'Inventory Executive', TO_DATE('2022-01-01',
'YYYY-MM-DD'), 25000);
INSERT INTO employee VALUES(20, 'Smita', 'Rao', '+91-9876543229',
'smita.rao@email.com', 'Sales Executive', TO_DATE('2022-01-01',
'YYYY-MM-DD'), 25000);
```

```
INSERT INTO sales VALUES (1, TO_DATE('2022-02-01', 'YYYY-MM-DD'),
2480499, 1, 1, 1);
INSERT INTO sales VALUES (2, TO_DATE('2022-02-05', 'YYYY-MM-DD'),
2182849, 2, 2, 2);
INSERT INTO sales VALUES (3, TO_DATE('2022-02-10', 'YYYY-MM-DD'),
2282059, 3, 3, 3);
INSERT INTO sales VALUES (4, TO_DATE('2022-02-15', 'YYYY-MM-DD'),
3770359, 4, 4, 4);
INSERT INTO sales VALUES (5, TO_DATE('2022-02-20', 'YYYY-MM-DD'),
3175039, 5, 5, 5);
INSERT INTO sales VALUES (6, TO_DATE('2022-02-25', 'YYYY-MM-DD'),
3075819, 6, 6, 6);
```

```
INSERT INTO sales VALUES (7, TO_DATE('2022-02-28', 'YYYY-MM-DD'), 4464899, 7, 7, 7);
INSERT INTO sales VALUES (8, TO_DATE('2022-03-03', 'YYYY-MM-DD'), 5357879, 8, 8, 8);
INSERT INTO sales VALUES (9, TO_DATE('2022-03-08', 'YYYY-MM-DD'), 5159439, 9, 9, 9);
INSERT INTO sales VALUES (10, TO_DATE('2022-03-12', 'YYYY-MM-DD'), 2381059, 10, 10, 10);
INSERT INTO sales VALUES (11, TO_DATE('2022-03-18', 'YYYY-MM-DD'), 2282059, 11, 11, 11);
INSERT INTO sales VALUES (12, TO_DATE('2022-03-22', 'YYYY-MM-DD'), 2579719, 12, 12, 12);
INSERT INTO sales VALUES (13, TO_DATE('2022-03-27', 'YYYY-MM-DD'), 2778159, 13, 13, 13);
INSERT INTO sales VALUES (14, TO_DATE('2022-03-31', 'YYYY-MM-DD'), 4960999, 14, 14, 14);
INSERT INTO sales VALUES (15, TO_DATE('2022-04-05', 'YYYY-MM-DD'), 8532919, 15, 15, 15);
INSERT INTO sales VALUES (16, TO_DATE('2022-04-10', 'YYYY-MM-DD'), 7937599, 16, 16, 16);
INSERT INTO sales VALUES (17, TO_DATE('2022-04-15', 'YYYY-MM-DD'), 5953119, 17, 17, 17);
INSERT INTO sales VALUES (18, TO_DATE('2022-04-20', 'YYYY-MM-DD'), 9921999, 18, 18, 18);
INSERT INTO sales VALUES (19, TO_DATE('2022-04-25', 'YYYY-MM-DD'), 25797199, 19, 19, 19);
INSERT INTO sales VALUES (20, TO_DATE('2022-04-30', 'YYYY-MM-DD'), 14882999, 20, 20, 20);
```

```
INSERT INTO service VALUES (1, 'Oil change', TO_DATE('2022-02-03', 'YYYY-MM-DD'), 50, 1, 1);
INSERT INTO service VALUES (2, 'Brake replacement', TO_DATE('2022-02-06', 'YYYY-MM-DD'), 200, 2, 2);
INSERT INTO service VALUES (3, 'Tire rotation', TO_DATE('2022-02-09', 'YYYY-MM-DD'), 75, 3, 3);
INSERT INTO service VALUES (4, 'Transmission flush', TO_DATE('2022-02-12', 'YYYY-MM-DD'), 300, 4, 4);
INSERT INTO service VALUES (5, 'Oil change', TO_DATE('2022-02-15', 'YYYY-MM-DD'), 50, 5, 5);
INSERT INTO service VALUES (6, 'Wheel alignment', TO_DATE('2022-02-18', 'YYYY-MM-DD'), 100, 6, 6);
```

```
INSERT INTO service VALUES (7, 'Brake inspection', TO_DATE('2022-02-21', 'YYYY-MM-DD'), 75, 7, 7);
INSERT INTO service VALUES (8, 'Coolant flush', TO_DATE('2022-02-24', 'YYYY-MM-DD'), 150, 8, 8);
INSERT INTO service VALUES (9, 'Oil change', TO_DATE('2022-02-27', 'YYYY-MM-DD'), 50, 9, 9);
INSERT INTO service VALUES (10, 'Brake replacement', TO_DATE('2022-03-02', 'YYYY-MM-DD'), 200, 10, 10);
INSERT INTO service VALUES (11, 'Tire replacement', TO_DATE('2022-03-05', 'YYYY-MM-DD'), 400, 1, 1);
INSERT INTO service VALUES (12, 'Oil change', TO_DATE('2022-03-08', 'YYYY-MM-DD'), 50, 2, 2);
INSERT INTO service VALUES (13, 'Brake replacement', TO_DATE('2022-03-11', 'YYYY-MM-DD'), 200, 3, 3);
INSERT INTO service VALUES (14, 'Transmission repair', TO_DATE('2022-03-14', 'YYYY-MM-DD'), 500, 4, 4);
INSERT INTO service VALUES (15, 'Oil change', TO_DATE('2022-03-17', 'YYYY-MM-DD'), 50, 5, 5);
INSERT INTO service VALUES (16, 'Wheel alignment', TO_DATE('2022-03-20', 'YYYY-MM-DD'), 100, 6, 6);
INSERT INTO service VALUES (17, 'Brake inspection', TO_DATE('2022-03-23', 'YYYY-MM-DD'), 75, 7, 7);
INSERT INTO service VALUES (18, 'Coolant flush', TO_DATE('2022-03-26', 'YYYY-MM-DD'), 150, 8, 8);
INSERT INTO service VALUES (19, 'Oil change', TO_DATE('2022-03-29', 'YYYY-MM-DD'), 50, 9, 9);
INSERT INTO service VALUES (20, 'Brake replacement', TO_DATE('2022-04-01', 'YYYY-MM-DD'), 200, 10, 10);
```

- **PL SQL**

```
DECLARE
        table_name VARCHAR2(50);
        v_choice VARCHAR2(10);
        emp_id NUMBER(10);
        first_name1 VARCHAR2(50);
        last_name1 VARCHAR2(50);
        phone_number1 VARCHAR2(20);
        email1 VARCHAR2(100);
        position1 VARCHAR2(100);
        hire_date1 DATE;
        salary1 NUMBER(10,2);
        v_customer_id NUMBER(10);
        aadhar_number1 VARCHAR2(12);
        driving_license1 VARCHAR2(20);
        CURSOR c_customers IS SELECT * FROM customer;
        r_customer c_customers%ROWTYPE;
        v_sale_id NUMBER(10);
        v_sale_date  DATE;
        v_sale_price NUMBER(10, 2);
        v_car_id NUMBER(10);
        v_employee_id NUMBER(10);
        v_service_id NUMBER(10);
        v_service_date DATE;
        v_service_type VARCHAR2(50);
        v_service_cost NUMBER(10, 2);
        v_manufacturer_id NUMBER(10);
        v_manufacturer_name VARCHAR2(100);
        v_hq_location VARCHAR2(100);
        v_phone_number VARCHAR2(20);
        v_email VARCHAR2(100);
        v_website VARCHAR2(100);
BEGIN
    DBMS_OUTPUT.PUT_LINE('Which table do you want to access? (employee, customer, sales, service, manufacturer, or car)');
  -- table_name := UPPER(TRIM('&table_name'));
  table_name := UPPER(TRIM('sales'));
  IF table_name = 'EMPLOYEE' THEN
        -- code to access employee table
        -- Get user input for action
        DBMS_OUTPUT.PUT_LINE('Do you want to insert, update, delete or display data in the employee table?');
```

```
        DBMS_OUTPUT.PUT_LINE('Enter "insert", "update", "delete",or
"display":');
        -- v_choice := UPPER(TRIM('&1'));
        v_choice := UPPER(TRIM('display'));

        -- Perform selected action
        IF v_choice = 'INSERT' THEN
        -- Get user input for new employee data
        -- v_employee_id := '&2';
        -- first_name := '&3';
        -- last_name := '&4';
        -- phone_number := '&5';
        -- email := '&6';
        -- position := '&7';
        -- hire_date := TO_DATE('&8', 'YYYY-MM-DD');
        -- salary := '&9';
        v_employee_id := '21';
        first_name1 := 'shivam';
        last_name1 := 'khurana';
        phone_number1 := '9206400037';
        email1 := 'khuranashivam987@gmail.com';
        position1 := 'manager';
        hire_date1 := TO_DATE('2003-12-1', 'YYYY-MM-DD');
        salary1 := '150000';

        -- Insert new employee data into table
        INSERT INTO employee (employee_id, first_name, last_name,
phone_number, email, position, hire_date, salary)
        VALUES (v_employee_id, first_name1, last_name1, phone_number1,
email1, position1, hire_date1, salary1);

        DBMS_OUTPUT.PUT_LINE('New employee data has been inserted
into the table.');

        ELSIF v_choice = 'UPDATE' THEN
        -- Get user input for employee data to update
        -- v_employee_id := '&2';
        -- first_name := '&3';
        -- last_name := '&4';
        -- phone_number := '&5';
        -- email := '&6';
        -- position := '&7';
        -- hire_date := TO_DATE('&8', 'YYYY-MM-DD');
        -- salary := '&9';
```

```
        v_employee_id := '1';
        first_name1 := 'rohan';
        last_name1 := 'thakur';
        phone_number1 := '123456789';
        email1 := 'rohanthakur@gmail.com';
        position1 := 'manager';
        hire_date1 := TO_DATE('2001-1-1', 'YYYY-MM-DD');
        salary1 := '10000';

        -- Update employee data in table
        UPDATE employee
        SET first_name = first_name1, last_name = last_name1,
phone_number = phone_number1,
        email = email1, position = position1, hire_date = hire_date1, salary =
salary1
        WHERE employee_id = v_employee_id;

        DBMS_OUTPUT.PUT_LINE('Employee data has been updated in the
table.');

        ELSIF v_choice = 'DELETE' THEN
        -- Get user input for employee ID to delete
        -- v_employee_id := '&2';
        v_employee_id := '1';

        -- Delete employee data from table
        DELETE FROM employee
        WHERE employee_id = v_employee_id;

        DBMS_OUTPUT.PUT_LINE('Employee data has been deleted from the
table.');
        ELSIF v_choice = 'DISPLAY' THEN
        FOR emp_rec IN (SELECT * FROM employee) LOOP
        -- Assign column values to variables
                v_employee_id := emp_rec.employee_id;
                first_name1 := emp_rec.first_name;
                last_name1 := emp_rec.last_name;
                phone_number1 := emp_rec.phone_number;
                email1 := emp_rec.email;
                position1 := emp_rec.position;
                hire_date1 := emp_rec.hire_date;
                salary1 := emp_rec.salary;

        -- Display variables
```

```
        DBMS_OUTPUT.PUT_LINE(v_employee_id || ', ' || first_name1 || ', ' ||
last_name1 || ', ' || phone_number1 || ', ' || email1 || ', ' || position1 || ', ' ||
hire_date1 || ', ' || salary1);
        END LOOP;
                DBMS_OUTPUT.PUT_LINE('Employee data has been displayed
from the table.');
        ELSE
        DBMS_OUTPUT.PUT_LINE('Invalid choice. Please enter "insert",
"update", "delete",or "display"');
        END IF;
  ELSIF table_name = 'CUSTOMER' THEN
        -- Prompt user for operation choice
        DBMS_OUTPUT.PUT_LINE('Do you want to insert, update, delete or
display data in the customer table?');
        DBMS_OUTPUT.PUT_LINE('Enter "insert", "update", "delete",or
"display":');
        -- v_choice := UPPER(TRIM('&1'));
        v_choice := UPPER(TRIM('display'));
        -- Perform selected operation
        IF v_choice = 'INSERT' THEN
        -- Insert new customer record
        -- v_customer_id := '&Enter customer ID: ';
        -- first_name := '&Enter first name: ';
        -- last_name := '&Enter last name: ';
        -- phone_number := '&Enter phone number: ';
        -- email := '&Enter email address: ';
        -- aadhar_number := '&Enter Aadhar number: ';
        -- driving_license := '&Enter driving license number: ';

        v_customer_id := '21';
        first_name1 := 'Aryan';
        last_name1 := 'Garg';
        phone_number1 := '3434343434';
        email1 := 'aryan@gmail.com';
        aadhar_number1 := '34345656';
        driving_license1 := 'dl08989898';
        INSERT INTO customer VALUES (v_customer_id, first_name1,
last_name1, phone_number1, email1, aadhar_number1, driving_license1);
        DBMS_OUTPUT.PUT_LINE('New record inserted into the customer
table.');

        ELSIF v_choice = 'UPDATE' THEN
        -- Update existing customer record
        --v_customer_id := '&Enter customer ID to update: ';
```

```
        -- phone_number := '&Enter new phone number: ';
        -- email := '&Enter new email address: ';
        v_customer_id := '21';
        phone_number1 := '9090909090';
        email1 := 'rayan@gmail.com';
        UPDATE customer SET phone_number = phone_number1, email = email1 WHERE customer_id = v_customer_id;
        DBMS_OUTPUT.PUT_LINE('Record updated in the customer table.');

        ELSIF v_choice = 'DELETE' THEN
        -- Delete customer record
        -- v_customer_id := '&Enter customer ID to delete: ';
        v_customer_id := '21';
        DELETE FROM customer WHERE customer_id = v_customer_id;
        DBMS_OUTPUT.PUT_LINE('Record deleted from the customer table.');
        ELSIF v_choice='DISPLAY' THEN
        OPEN c_customers;
                LOOP
                FETCH c_customers INTO r_customer;
                EXIT WHEN c_customers%NOTFOUND;
                DBMS_OUTPUT.PUT_LINE(r_customer.customer_id || ', ' || r_customer.first_name || ', ' || r_customer.last_name || ', ' || r_customer.phone_number || ', ' || r_customer.email || ', ' || r_customer.aadhar_number || ', ' || r_customer.driving_license);
                END LOOP;
                CLOSE c_customers;
        ELSE
        DBMS_OUTPUT.PUT_LINE('Invalid choice. Please try again.');
        END IF;
        -- code to access customer table
  ELSIF table_name = 'SALES' THEN
  -- Prompt user for action to perform
        DBMS_OUTPUT.PUT_LINE('Enter the action to perform (insert, update, delete, display): in sales table');
        -- v_choice := UPPER(TRIM('&choice'));
        v_choice := UPPER(TRIM('update'));

  -- Perform action based on user input
  IF v_choice = 'INSERT' THEN
        -- Prompt user for data to insert
        -- v_sale_id := '&sale_id';
        -- v_sale_date := TO_DATE('&sale_date', 'YYYY-MM-DD');
        -- v_sale_price := '&sale_price';
```

```
        -- v_customer_id := '&customer_id';
        -- v_car_id := '&car_id';
        -- v_employee_id := '&employee_id';
        v_sale_id := '21';
        v_sale_date := TO_DATE('2022-02-01', 'YYYY-MM-DD');
        v_sale_price := '454545';
        v_customer_id := '10';
        v_car_id := '5';
        v_employee_id := '4';

        -- Insert data into sales table
        INSERT INTO sales (sale_id, sale_date, sale_price, customer_id,
car_id, employee_id)
        VALUES (v_sale_id, v_sale_date, v_sale_price, v_customer_id,
v_car_id, v_employee_id);

        DBMS_OUTPUT.PUT_LINE('Data inserted successfully.');
  ELSIF v_choice = 'UPDATE' THEN
        -- Prompt user for data to update
        -- v_sale_id := '&sale_id';
        -- v_sale_date := TO_DATE('&sale_date', 'YYYY/MM/DD');
        -- v_sale_price := '&sale_price';
        -- v_customer_id := '&customer_id';
        -- v_car_id := '&car_id';
        -- v_employee_id := '&employee_id';
        v_sale_id := '21';
        v_sale_date := TO_DATE('2022-02-01', 'YYYY-MM-DD');
        v_sale_price := '232323';
        v_customer_id := '5';
        v_car_id := '5';
        v_employee_id := '5';

        -- Update data in sales table
        UPDATE sales SET sale_date = v_sale_date, sale_price =
v_sale_price, customer_id = v_customer_id, car_id = v_car_id, employee_id =
v_employee_id WHERE sale_id = v_sale_id;

        DBMS_OUTPUT.PUT_LINE('Data updated successfully.');
  ELSIF v_choice = 'DELETE' THEN
        -- Prompt user for sale ID to delete
        -- v_sale_id := '&sale_id';
        v_sale_id := '21';

        -- Delete data from sales table
```

```
        DELETE FROM sales WHERE sale_id = v_sale_id;

        DBMS_OUTPUT.PUT_LINE('Data deleted successfully.');
  ELSIF v_choice = 'DISPLAY' THEN
        -- Display data from sales table
        FOR rec IN (SELECT * FROM sales) LOOP
        DBMS_OUTPUT.PUT_LINE(rec.sale_id || ' ' || rec.sale_date  ||  '  '  ||
rec.sale_price || ' ' || rec.customer_id || ' ' || rec.car_id || ' ' || rec.employee_id);
        END LOOP;
  ELSE
        DBMS_OUTPUT.PUT_LINE('Invalid choice.');
  END IF;

        -- code to access sales table
  ELSIF table_name = 'SERVICE' THEN
        -- Prompt user for action to perform
       DBMS_OUTPUT.PUT_LINE('Enter the action to perform (insert, update,
delete, display):');
        -- v_choice := UPPER(TRIM('&choice'));
        v_choice := UPPER(TRIM('display'));

  -- Perform action based on user input
  IF v_choice = 'INSERT' THEN
        -- Prompt user for data to insert
        -- v_service_id := '&service_id';
        -- v_service_type := '&service_type';
        -- v_service_date := TO_DATE('&service_date', 'YYYY/MM/DD');
        -- v_service_cost := '&service_cost';
        -- v_car_id := '&car_id';
        -- v_employee_id:='&employee_id';
        v_service_id := '21';
        v_service_type := 'tyre';
        v_service_date := TO_DATE('2003/10/10', 'YYYY/MM/DD');
        v_service_cost := '5000';
        v_car_id := '3';
        v_employee_id:='3';
        -- Insert data into service table
        INSERT INTO service (service_id, service_type,service_date,
service_cost, car_id, employee_id)
        VALUES (v_service_id, v_service_type,v_service_date,
v_service_cost, v_car_id, v_employee_id);

        DBMS_OUTPUT.PUT_LINE('Data inserted successfully.');
  ELSIF v_choice = 'UPDATE' THEN
```

```
        -- Prompt user for data to update
        -- v_service_id := '&service_id';
        -- v_service_type := '&service_type';
        -- v_service_date := TO_DATE('&service_date', 'YYYY/MM/DD');
        -- v_service_cost := '&service_cost';
        -- v_car_id := '&car_id';
        -- v_employee_id:='&employee_id';
        v_service_id := '21';
        v_service_type := 'dash';
        v_service_date := TO_DATE('&service_date', 'YYYY/MM/DD');
        v_service_cost := '500';
        v_car_id := '3';
        v_employee_id:='3';
        -- Update data in service table
        UPDATE service SET
        service_date = v_service_date,
        service_type = v_service_type,
        service_cost = v_service_cost,
        car_id = v_car_id,
        employee_id=v_employee_id
        WHERE service_id = v_service_id;

        DBMS_OUTPUT.PUT_LINE('Data updated successfully.');
  ELSIF v_choice = 'DELETE' THEN
        -- Prompt user for service ID to delete
        -- v_service_id := '&service_id';
        v_service_id := '21';

        -- Delete data from service table
        DELETE FROM service WHERE service_id = v_service_id;

        DBMS_OUTPUT.PUT_LINE('Data deleted successfully.');

  ELSIF v_choice = 'DISPLAY' THEN
        -- Display data from service table
        FOR rec IN (SELECT * FROM service) LOOP
        DBMS_OUTPUT.PUT_LINE(rec.service_id || ' ' || rec.service_type || ' '||
rec.service_date  ||  '  '    || rec.service_cost || ' ' || rec.car_id || ' ' ||
rec.employee_id);
        END LOOP;
  ELSE
        DBMS_OUTPUT.PUT_LINE('Invalid choice.');
  END IF;
        -- code to access service table
```

```
 ELSIF table_name = 'MANUFACTURERS' THEN
 -- Prompt user for action to perform
   DBMS_OUTPUT.PUT_LINE('Enter the action to perform (insert, update, delete, display): in manufacturers');
 -- v_choice := UPPER(TRIM('&choice'));
 v_choice := UPPER(TRIM('delete'));

 -- Perform action based on user input
 IF v_choice = 'INSERT' THEN
        -- Prompt user for data to insert
        -- v_manufacturer_id := '&manufacturer_id';
        -- v_manufacturer_name := '&manufacturer_name';
        -- v_hq_location := '&hq_location';
        -- v_phone_number := '&phone_number';
        -- v_email := '&email';
        -- v_website := '&website';
        v_manufacturer_id := '21';
        v_manufacturer_name := 'rerererer';
        v_hq_location := 'ffffffff';
        v_phone_number := '9898989898';
        v_email := 'f@gmail.com';
        v_website := 'www.f.com';

        -- Insert data into manufacturers table
        INSERT INTO manufacturers (manufacturer_id, manufacturer_name, hq_location, phone_number, email, website)
        VALUES (v_manufacturer_id, v_manufacturer_name, v_hq_location, v_phone_number, v_email, v_website);

        DBMS_OUTPUT.PUT_LINE('Data inserted successfully.');

 ELSIF v_choice = 'UPDATE' THEN
        -- Prompt user for data to update
        -- v_manufacturer_id := '&manufacturer_id';
        -- v_manufacturer_name := '&manufacturer_name';
        -- v_hq_location := '&hq_location';
        -- v_phone_number := '&phone_number';
        -- v_email := '&email';
        -- v_website := '&website';

        v_manufacturer_id := '21';
        v_manufacturer_name := 'gregrgrgr';
        v_hq_location := 'ggggggg';
        v_phone_number := '66565656566';
```

```
        v_email := 'g@gmail.com';
        v_website := 'www.g.com';

        -- Update data in manufacturers table
        UPDATE manufacturers
        SET manufacturer_name = v_manufacturer_name,
        hq_location = v_hq_location,
        phone_number = v_phone_number,
        email = v_email,
        website = v_website
        WHERE manufacturer_id = v_manufacturer_id;
   DBMS_OUTPUT.PUT_LINE('Data updated successfully.');

  ELSIF v_choice = 'DELETE' THEN
        -- Prompt user for manufacturer ID to delete
        -- v_manufacturer_id := '&manufacturer_id';
        v_manufacturer_id := '21';

        -- Delete data from manufacturers table
        DELETE FROM manufacturers WHERE manufacturer_id =
v_manufacturer_id;

        DBMS_OUTPUT.PUT_LINE('Data deleted successfully.');

  ELSIF v_choice = 'DISPLAY' THEN
        -- Display data from manufacturers table
        FOR rec IN (SELECT * FROM manufacturers) LOOP
        DBMS_OUTPUT.PUT_LINE(rec.manufacturer_id || ' ' ||
rec.manufacturer_name || ' ' || rec.hq_location || ' ' || rec.phone_number || ' ' ||
rec.email || ' ' || rec.website);
        END LOOP;

  ELSE
        DBMS_OUTPUT.PUT_LINE('Invalid choice.');

  END IF;
        -- code to access manufacturer table
  ELSIF table_name = 'CAR' THEN
        DBMS_OUTPUT.PUT_LINE('NOT ALLOWED TO ACCESS CAR ');
        -- code to access car table
  ELSE
        DBMS_OUTPUT.PUT_LINE('Invalid table name');
  END IF;
END;
```

OUTPUT

DISPLAY:

```
250     DBMS_OUTPUT.PUT_LINE('Do you want to insert, update, delete or display data in the employee table?');
251     DBMS_OUTPUT.PUT_LINE('Enter "insert", "update", "delete",or "display":');
252     -- v_choice := UPPER(TRIM('&1'));
253     v_choice := UPPER(TRIM('display'));
```

```
Statement processed.
Which table do you want to access? (employee, customer, sales, service, manufacturer, or car)
Do you want to insert, update, delete or display data in the employee table?
Enter "insert", "update", "delete",or "display":
1, Amit, Sharma, +91-9876543210, amit.sharma@email.com, Sales Executive, 01-JAN-22, 25000
2, Rahul, Gupta, +91-9876543211, rahul.gupta@email.com, Sales Executive, 01-JAN-22, 25000
3, Priya, Singh, +91-9876543212, priya.singh@email.com, Sales Executive, 01-JAN-22, 25000
4, Suresh, Kumar, +91-9876543213, suresh.kumar@email.com, Sales Manager, 01-JAN-22, 40000
5, Neha, Patil, +91-9876543214, neha.patil@email.com, Sales Manager, 01-JAN-22, 40000
6, Sarika, Joshi, +91-9876543215, sarika.joshi@email.com, Marketing Manager, 01-JAN-22, 45000
7, Alok, Mishra, +91-9876543216, alok.mishra@email.com, Marketing Executive, 01-JAN-22, 30000
8, Anjali, Shah, +91-9876543217, anjali.shah@email.com, Marketing Executive, 01-JAN-22, 30000
9, Manish, Kapoor, +91-9876543218, manish.kapoor@email.com, Finance Manager, 01-JAN-22, 50000
10, Neetu, Chopra, +91-9876543219, neetu.chopra@email.com, Finance Executive, 01-JAN-22, 35000
11, Vikas, Gupta, +91-9876543220, vikas.gupta@email.com, Service Manager, 01-JAN-22, 45000
12, Manoj, Singh, +91-9876543221, manoj.singh@email.com, Service Executive, 01-JAN-22, 30000
13, Poonam, Verma, +91-9876543222, poonam.verma@email.com, Service Executive, 01-JAN-22, 30000
14, Rohan, Patel, +91-9876543230, rohan.patel@email.com, Sales Executive, 01-JAN-22, 25000
15, Rajesh, Mishra, +91-9876543224, rajesh.mishra@email.com, IT Executive, 01-JAN-22, 30000
16, Kavita, Sharma, +91-9876543225, kavita.sharma@email.com, HR Manager, 01-JAN-22, 45000
17, Prakash, Jha, +91-9876543226, prakash.jha@email.com, HR Executive, 01-JAN-22, 30000
18, Aparna, Nair, +91-9876543227, aparna.nair@email.com, Inventory Manager, 01-JAN-22, 40000
19, Ramesh, Menon, +91-9876543228, ramesh.menon@email.com, Inventory Executive, 01-JAN-22, 25000
20, Smita, Rao, +91-9876543229, smita.rao@email.com, Sales Executive, 01-JAN-22, 25000
Employee data has been displayed from the table.
```

INSERT:

```
250     DBMS_OUTPUT.PUT_LINE('Do you want to insert, update, delete or display data in the employee table?');
251     DBMS_OUTPUT.PUT_LINE('Enter "insert", "update", "delete",or "display":');
252     -- v_choice := UPPER(TRIM('&1'));
253     v_choice := UPPER(TRIM('insert'));
254
255     -- Perform selected action
256     IF v_choice = 'INSERT' THEN
257         -- Get user input for new employee data
258         -- v_employee_id := '&2';
259         -- first_name := '&3';
260         -- last_name := '&4';
261         -- phone_number := '&5';
262         -- email := '&6';
263         -- position := '&7';
264         -- hire_date := TO_DATE('&8', 'YYYY-MM-DD');
265         -- salary := '&9';
266         v_employee_id := '21';
267         first_name1 := 'shivam';
268         last_name1 := 'khurana';
269         phone_number1 := '9206400037';
270         email1 := 'khuranashivam987@gmail.com';
271         position1 := 'manager';
272         hire_date1 := TO_DATE('2003-12-1', 'YYYY-MM-DD');
273         salary1 := '150000';
```

```
Statement processed.
Which table do you want to access? (employee, customer, sales, service, manufacturer, or car)
Do you want to insert, update, delete or display data in the employee table?
Enter "insert", "update", "delete",or "display":
New employee data has been inserted into the table.
```

UPDATE:

```
    ELSIF v_choice = 'UPDATE' THEN
        -- Get user input for employee data to update
        -- v_employee_id := '&2';
        -- first_name := '&3';
        -- last_name := '&4';
        -- phone_number := '&5';
        -- email := '&6';
        -- position := '&7';
        -- hire_date := TO_DATE('&8', 'YYYY-MM-DD');
        -- salary := '&9';
        v_employee_id := '1';
        first_name1 := 'rohan';
        last_name1 := 'thakur';
        phone_number1 := '123456789';
        email1 := 'rohanthakur@gmail.com';
        position1 := 'manager';
        hire_date1 := TO_DATE('2001-1-1', 'YYYY-MM-DD');
        salary1 := '10000';

        -- Update employee data in table
        UPDATE employee
        SET first_name = first_name1, last_name = last_name1, phone_number = phone_number1,
            email = email1, position = position1, hire_date = hire_date1, salary = salary1
        WHERE employee_id = v_employee_id;
```

```
Statement processed.
Which table do you want to access? (employee, customer, sales, service, manufacturer, or car)
Do you want to insert, update, delete or display data in the employee table?
Enter "insert", "update", "delete",or "display":
Employee data has been updated in the table.
```

DELETE:

```

    ELSIF v_choice = 'DELETE' THEN
        -- Get user input for employee ID to delete
        -- v_employee_id := '&2';
        v_employee_id := '1';

        -- Delete employee data from table
        DELETE FROM employee
        WHERE employee_id = v_employee_id;

        DBMS_OUTPUT.PUT_LINE('Employee data has been deleted from the table.');
    ELSIF v_choice = 'DISPLAY' THEN
        FOR emp_rec IN (SELECT * FROM employee) LOOP
    -- Assign column values to variables
        v_employee_id := emp_rec.employee_id;
        first_name1 := emp_rec.first_name;
        last_name1 := emp_rec.last_name;
```

```
Statement processed.
Which table do you want to access? (employee, customer, sales, service, manufacturer, or car)
Do you want to insert, update, delete or display data in the employee table?
Enter "insert", "update", "delete",or "display":
Employee data has been deleted from the table.
```

CONCLUSION

In conclusion, the implementation of a DBMS for a car showroom can greatly benefit the business in terms of improved data management, increased efficiency, and better decision-making. By organizing data into a well-designed database schema and utilizing appropriate data management techniques, the car showroom can better track sales, inventory, and customer behavior. The DBMS can also enhance data security and facilitate collaboration among employees. Ultimately, investing in a DBMS for a car showroom can lead to improved business operations and increased profitability.

Overall, a well-designed DBMS can be a valuable tool for managing a car showroom and can help drive business success by improving data management, decision-making, and customer experience, among other benefits.

REFERENCES

- https://docs.oracle.com/en-us/iaas/mysql-database/index.html
- https://www.w3schools.com/MySQL/default.asp
- https://www.javatpoint.com/mysql-tutorial