College of Engineering

Northeastern University

Project Report

DrivePro - Automobile Dealer Management System

DAMG 6210 Data Management and Database Design

TEAM 1

Name	NUID
Siddharth Bahekar	002417718
Chitra Periya	002893640
Rachita Shah	002482615

TABLE OF CONTENTS

Problem Statement

Challenges and identified gaps in Automotive Industry

The automotive industry is experiencing significant challenges in effectively managing dealership operations due to outdated systems and processes. Current software solutions struggle to efficiently handle essential tasks such as sales, inventory management, parts tracking, and customer information management within dealerships. This inefficiency leads to various operational hurdles, including difficulty in accurately measuring sales performance, inadequate inventory control, and challenges in tracking customer satisfaction.

Furthermore, despite the wealth of data generated within dealership ecosystems, there is a notable gap in leveraging this data to drive informed decision-making. Existing systems lack the capability to perform comprehensive data analysis and complex queries, limiting their ability to provide valuable insights for improving sales performance, enhancing customer satisfaction, optimizing revenue growth, and managing inventory effectively.

In essence, the automotive industry faces a critical need for a modern Automotive Dealer Management System (DMS) that can address these challenges. Such a system should streamline dealership operations, adapt to the increasing volume of data generated, and provide actionable insights for strategic decision-making. To bridge this gap, it is imperative to develop an advanced DMS solution that seamlessly integrates with existing dealership workflows, harnesses the power of relational databases for efficient data management, and empowers stakeholders with the insights required to thrive in the competitive automotive market.

Designing Process

Requirement Gathering

To kickstart the development of our automobile dealership system, we went through an extensive process of online research. Through rigorous exploration of diverse sources, we gained insights into dealership operations at their core. This involved studying materials from various articles and sources to comprehend the detailed steps involved in running a dealership and how they interrelate.

We constantly monitored the latest industry news to gain a real-world perspective on the challenges encountered by users with existing systems. By the insights gathered from online sources and current affairs updates, we ensured a complete understanding of both theoretical frameworks and practical challenges. This approach is pivotal in ensuring that our system not only meets conventional requirements but also effectively addresses the genuine challenges faced by dealership personnel in their daily operations. By leveraging insights from both theoretical knowledge and real-world experiences, we aim to develop a system that optimizes dealership operations and resolves practical issues faced by users.

Vision

Our vision for the automation dealership system is to revolutionize the way dealerships operate, ushering in a new era of efficiency, transparency, and profitability. We envision a system that seamlessly integrates cutting-edge technology with intuitive design, empowering dealership personnel to streamline their workflows and enhance customer satisfaction.

At the core of our vision lies a commitment to modernize dealership operations by offering a dynamic system that adapts to the evolving needs of the automotive industry. We envision a platform that not only meets the fundamental requirements of dealership management but also anticipates and proactively addresses emerging challenges. By leveraging RDBMS concepts, our system will empower dealerships to make data-driven decisions, optimize inventory management, and deliver personalized customer experiences that set them apart in a competitive market.

Business Problems Addressed

Customer Management:

- Efficiently store and manage customer contact details, addresses, and service history, enabling personalized communication and better customer service
- Analyze customer data to identify trends and preferences, allowing targeted marketing campaigns and proactive engagement to enhance customer satisfaction and retention

Inventory Management:

- Track vehicle inventory details such as make, model, year, and price, ensuring optimal inventory levels and reducing stockouts to improve sales efficiency
- Utilize historical sales data and demand forecasting to make informed purchasing decisions, minimizing overstocking and understocking risks, and maximizing profitability

Sales Management:

- Enable comprehensive visibility into the sales pipeline and key performance indicators (KPIs) such as conversion rates and deal size, facilitating better sales forecasting and performance analysis
- Provide real-time access to customer and inventory data for sales staff, streamlining the sales process and enhancing the customer experience

Service Management:

- Automate service appointment scheduling and reminders based on vehicle maintenance schedules, driving repeat business and increasing service revenue
- Allocate resources efficiently based on workload and skillset, minimizing wait times and maximizing throughput to enhance customer satisfaction and loyalty

Supplier Management:

- Proactively manage supplier relationships through automated communication and performance monitoring, ensuring timely procurement of parts and effective cost management
- Streamline procurement processes through integration with inventory and purchasing modules, reducing lead times and administrative overhead

Payment Management:

- Ensure compliance with payment industry standards for secure handling of payment information, supporting multiple payment methods to cater to diverse customer preferences

- Automate reconciliation of payments with accounting systems, streamlining financial reporting and reducing errors associated with manual data entry

Feedback Management:

- Aggregate feedback from various channels and analyze sentiment to identify trends and areas for improvement, enhancing customer satisfaction and loyalty
- Integrate feedback with customer records for personalized follow-up and resolution of complaints, demonstrating responsiveness and commitment to customer satisfaction

Employee Management:

- Automate routine HR tasks such as onboarding and payroll processing, freeing up HR staff to focus on strategic initiatives and employee development
- Provide visibility into employee performance metrics and facilitate training and development programs to improve employee morale, productivity, and retention

Business Rules

- Each dealership within the system is required to possess a distinct combination of a unique name and location, ensuring that no two dealerships share identical identifiers
- Feedback ratings submitted within the system must fall within the predefined range of values, specifically ranging from 1 to 5, to maintain consistency and reliability in the evaluation of services
- Employee records within the system necessitate an age requirement of no less than 18 years, ensuring compliance with legal regulations and safeguarding against the employment of minors
- Salespersons registered within the system are mandated to possess unique email addresses, thereby preventing duplication and ensuring clear communication channels
- The pricing information associated with parts stored within the system must be non-negative, prohibiting the input of negative values to maintain accurate financial records
- Transactions recorded within the system must include a valid date, with the transaction date field being mandatory to prevent the omission of crucial temporal information

- Employees recorded within the system are to be affiliated with a specific dealership, establishing clear organizational hierarchies and facilitating managerial oversight
- Vehicles registered within the system are to be exclusively linked to a designated dealership, ensuring accurate inventory management and dealership attribution
- Each transaction logged within the system must involve essential entities, including a vehicle, a customer, and a salesperson, to ensure completeness and accuracy in transactional records
- Service appointments scheduled within the system are required to encompass essential components, namely a customer, a vehicle, and a service type, ensuring that all pertinent information is captured for service-related activities
- Feedback submissions within the system must be linked to corresponding service appointments, allowing for direct correlation between customer feedback and service experiences
- Parts cataloged within the system must be associated with both a specific vehicle and a designated supplier, facilitating effective inventory tracking and procurement management
- Inventory quantities attributed to vehicles stored within the system are mandated to maintain non-negative values, preventing erroneous data entries and ensuring inventory accuracy
- Feedback comments submitted within the system are restricted to a maximum length of 1000 characters, promoting concise and focused feedback submissions
- Each customer profile within the system is permitted to engage in multiple transactions, allowing for the recording of recurring interactions and transactions with the dealership
- Employees affiliated with dealerships are permitted to be employed across various roles within the organization, with each employee being exclusively associated with a single dealership, ensuring clarity in organizational structure and employee responsibilities
- Phone numbers recorded within the system are required to adhere to a standardized format of 10 digits, ensuring consistency and uniformity in contact information records

Relationships/Associations

Dealership to Employee:

- One-to-Many relationship (One dealership can have multiple employees)

Dealership to Vehicle:

- One-to-Many relationship (One dealership can have multiple vehicle)

Customer to Transaction:

- One-to-Many relationship (One Customer can have multiple transactions)

Vehicle to Part:

- One-to-Many relationship (One vehicle can have multiple employees)

Supplier to Part:

- One-to-Many relationship (One supplier can have multiple parts)

Dealership to Employee:

- One-to-Many relationship (One dealership can have multiple employees)

ServiceAppointment to ServiceType:

- One-to-Many relationship (One service appointment can have multiple service types)

Transaction to PaymentMethod:

- One-to-Many relationship (One transaction can have multiple payment methods)

Customer to ServiceAppointment:

- One-to-Many relationship (One customer can have multiple service appointments)

SalesPerson to SalesCommission:

- One-to-Many relationship (One sales person can have multiple commissions).

Vehicle to VehicleInventory:

- One-to-one relationship (One vehicle will always be in one vehicle inventory)

ServiceAppointment to Feedback:

- One-to-one relationship (One service appointment can have one feedback)

Table Description

Dealership:

- Each dealership operates independently, managing its inventory, staff, and services
- They serve as the central hubs for vehicle sales, maintenance, and customer interaction

Vehicle:

- Each vehicle is associated with a specific dealership, indicating where it is available for purchase
- The inventory of vehicles is tracked to monitor stock levels and availability
- Vehicles are involved in sales transactions, where customers purchase or lease them
- They can also be scheduled for service appointments for maintenance and repairs

Customer:

- Customers engage in transactions to buy or lease vehicles from dealerships
- They schedule service appointments for vehicle maintenance and repairs
- After service appointments, customers provide feedback, which helps improve service quality

Transaction:

- Transactions record the details of vehicle sales, including the vehicle involved, the customer purchasing it, the transaction date, and the amount paid
- They facilitate the transfer of ownership from the dealership to the customer

Inventory:

- Inventory management tracks the quantity of each vehicle available at a dealership
- It ensures that dealerships maintain adequate stock levels to meet customer demand

Employee:

- Employees work at dealerships, fulfilling various roles such as sales representatives, service technicians, and administrative staff
- They contribute to the day-to-day operations of the dealership, ensuring smooth functioning and customer satisfaction

ServiceType:

- Dealerships offer various services such as vehicle maintenance, repairs, and inspections
- These services aim to keep customers' vehicles in optimal condition and ensure their safety and performance

Part:

- Dealerships stock various parts and accessories necessary for vehicle maintenance and repairs
- These parts are used during service appointments to replace worn-out or damaged components

Supplier:

- Suppliers provide dealerships with parts and components necessary for vehicle maintenance and repairs
- They play a crucial role in ensuring dealerships have access to high-quality parts to serve their customers effectively

SalesPerson:

- SalesPerson assist customers in selecting and purchasing vehicles
- They earn commissions based on the sales they facilitate, incentivizing them to provide excellent customer service

PaymentMethod:

- Payment methods offer customers various options to pay for their vehicle purchases, including cash, credit/debit cards, financing, or leasing

ServiceAppointment:

- Service appointments allow customers to schedule maintenance or repairs for their vehicles
- They ensure timely servicing to maintain vehicle performance and safety

Feedback:

- Customer feedback provides valuable insights into the quality of service provided by the dealership
- Dealerships use this feedback to identify areas for improvement and enhance the overall customer experience

SalesCommission:

- Sales commissions reward salespeople for their efforts in facilitating vehicle sales.
- They are typically calculated as a percentage of the sale amount, motivating sales staff to achieve higher sales volumes.

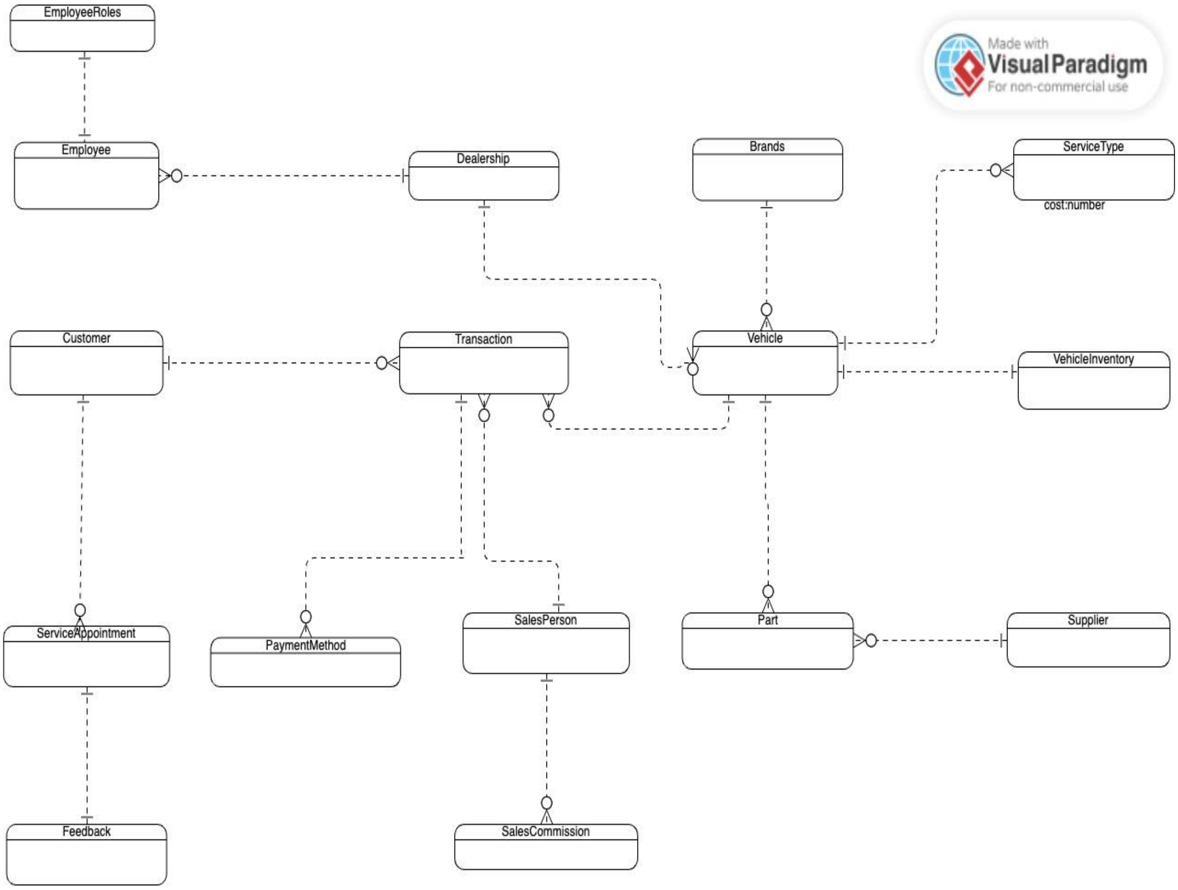
Tools and Techniques

Database Lifecycle(DBLC)

We have incorporated the different steps in the DBLC lifecycle like database initial study, requirement collection and analysis, normalization, implementation, testing, and furthermore into our Dealer Management System. Below are the steps followed:

- Database Initial Study : Extensive research, defined problems, proposed solutions, and the scope of the project
- Database Requirements Analysis: Gathering and analyzing requirements for the database, including data types, volumes, performance requirements, and user needs
- Entity-Relationship Diagrams (ERD): Creating ERDs to visualize the relationships between different data entities and understand the database structure
- Normalization: Applying normalization techniques to ensure data integrity, referential integrity, and reduce redundancy in the database schema
- Database Modeling Tools: Utilizing tools like Visual Paradigm and Oracle SQL for designing and visualizing the database schema
- Database Design Review: Conducting peer reviews and walkthroughs to validate the database design against requirements and best practices
- SQL Scripting: Writing SQL scripts to create database tables, define relationships, and implement constraints based on the finalized database design
- Data Quality Assessment: Evaluating data quality and consistency within the database through data profiling and validation checks
- Backup and Recovery Procedures: Establishing backup and recovery procedures to ensure data integrity and minimize downtime in case of failures
- Database Administration (DBA) Tasks: Performing routine maintenance tasks such as index optimization, database tuning, and space management

ER Diagram - Conceptual



Key Design Decisions

Appropriate use of foreign keys:

- Foreign keys added to establish relationships between tables like transaction_id in the result table linking to Vehicle table
- This ensures referential integrity is maintained

Normalisation of tables:

- Tables are normalised to 2NF to reduce redundancy and maintain integrity

Auto-incremented primary keys:

- Primary keys like customer_id, employee_id use auto-increment to generate unique IDs

One-to-Many Relationships:

- The ERD illustrates one-to-many relationships between entities using crow-foot notation. For example, one dealership can have multiple vehicles, employees, and transactions associated with it

Appropriate data types:

- Varchar, number and date used sensibly as per attribute requirements

Indexes on foreign keys:

- Indexes likely added on foreign keys for fast joins and queries involving relationships

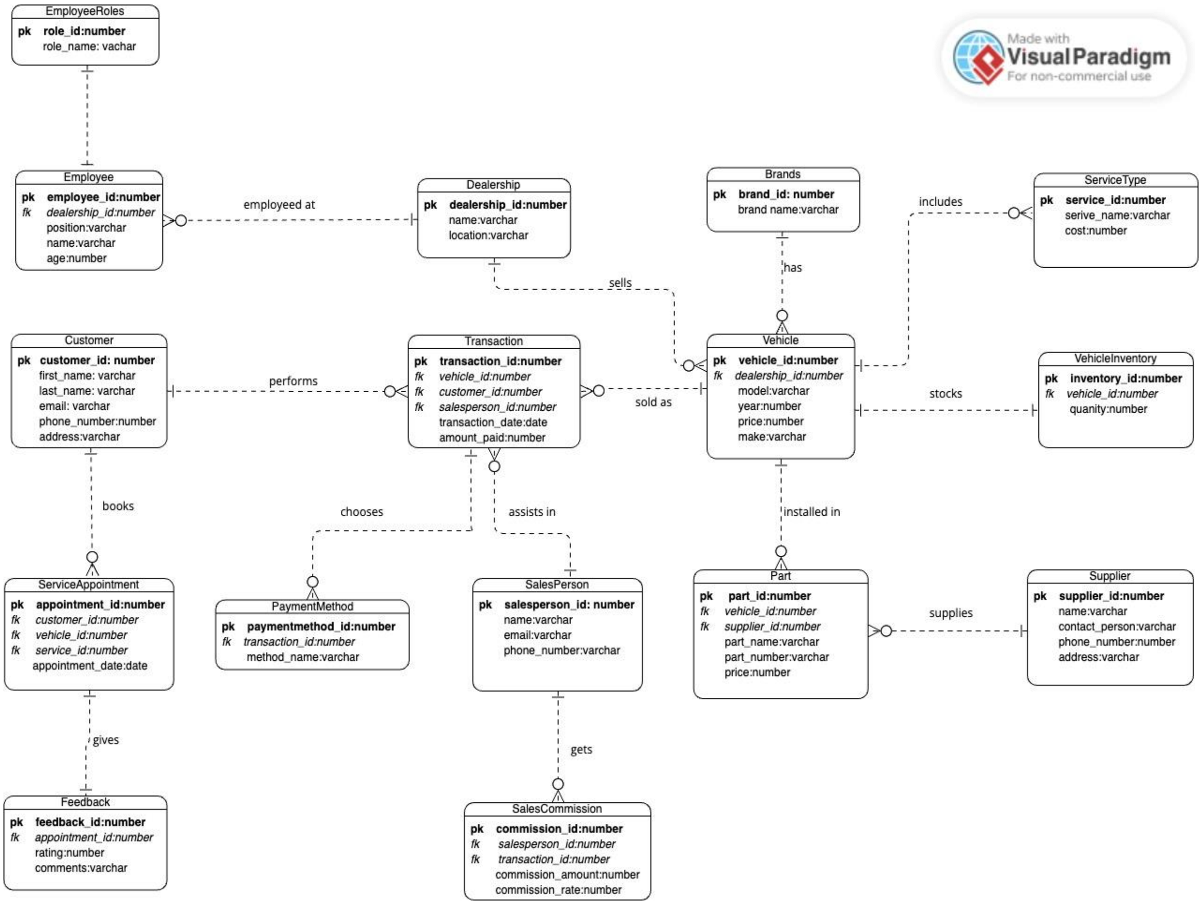
Constraints:

- Constraints such as NOT NULL, UNIQUE, and PRIMARY KEY are applied to ensure data consistency and integrity

Entity Relationships:

- Relationships between entities are carefully defined to accurately model the interactions and dependencies between different aspects of the dealership management system. This allows for efficient data retrieval and manipulation while maintaining data integrity

ER Diagram - Initial



Normalization

In the provided database schema for the Automobile Dealership Management System (DMS), several normalization techniques have been applied to ensure data integrity, minimize redundancy, and optimize data storage. Let's discuss the normalization levels achieved in this database:

- **First Normal Form (1NF):**

All tables have atomic values in each column, meaning that each column contains only single, indivisible values. There are no repeating groups or arrays within any of the tables

- **Second Normal Form (2NF):**

Each non-key attribute is fully functionally dependent on the primary key.

For example: In the Transactions table, attributes like transaction_date and amount_paid are functionally dependent on the transaction_id, which is the primary key. In the ServiceAppointments table, attributes such as appointment_date are fully dependent on the appointment_id, which serves as the primary key

Normalization Decisions and Achievements

- **Decomposition into Smaller Tables:**

Various tables such as Transactions, Inventory, ServiceAppointments, Feedback, SalesCommission, etc., have been created to store specific types of data separately. This decomposition reduces redundancy and improves data management.

- **Removal of Partial Dependencies:**

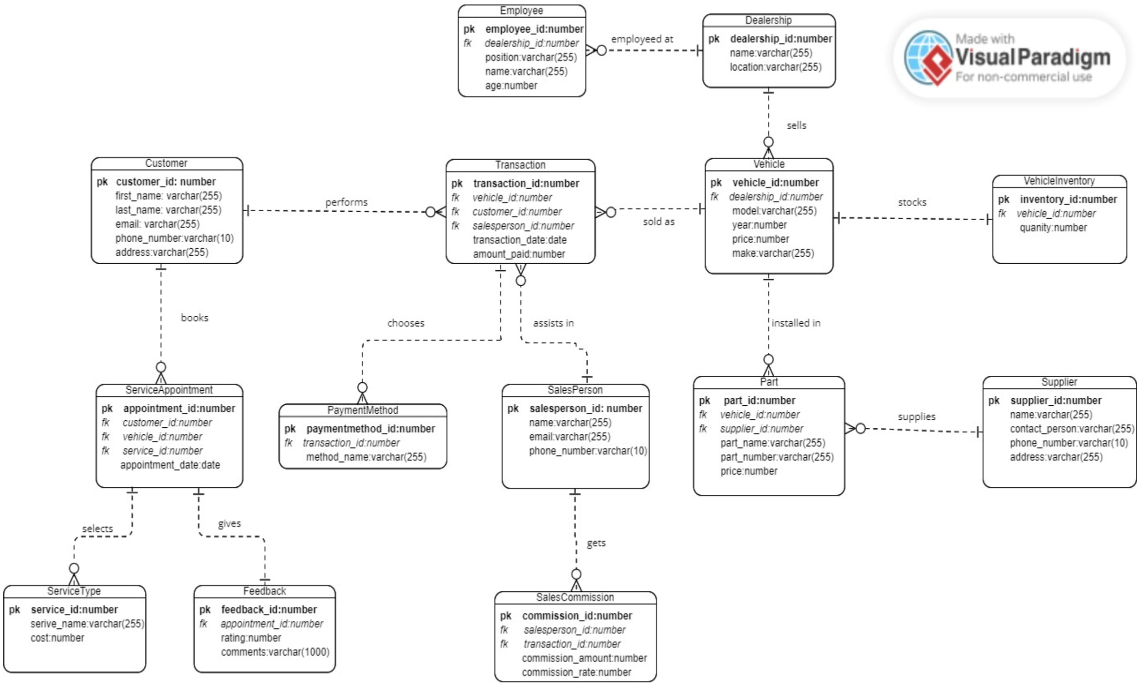
Non-key attributes in each table are fully functionally dependent on the primary key, ensuring that there are no partial dependencies.

- **Elimination of Redundancy:**

Redundant data has been minimized by breaking tables into smaller, more atomic units and establishing relationships between them. For example, rather than storing all information about a transaction in a single table, it is split between Transactions and Inventory tables, reducing redundancy and improving data integrity.

Overall, normalization in the provided database schema ensures that the data is well-structured, efficient, and maintains integrity, which is crucial for the effective functioning of the Automobile Dealership Management System.

ER Diagram - Final



Physical Data Store Organization

Table Name	Column Name	Data Type	Constraints
Dealership	dealership_id	NUMBER	PRIMARY KEY
	name	VARCHAR(255)	NOT NULL
	location	VARCHAR(255)	NOT NULL
Customer	customer_id	NUMBER	PRIMARY KEY
	first_name	VARCHAR(255)	NOT NULL
	last_name	VARCHAR(255)	NOT NULL
	email	VARCHAR(255)	UNIQUE
	phone_number	VARCHAR(10)	UNIQUE
	address	VARCHAR(255)	
Employee	employee_id	NUMBER	PRIMARY KEY
	dealership_id	NUMBER	NOT NULL
	position	VARCHAR(255)	NOT NULL
	name	VARCHAR(255)	NOT NULL
	age	NUMBER	
Vehicle	vehicle_id	NUMBER	PRIMARY KEY
	dealership_id	NUMBER	NOT NULL
	model	VARCHAR(255)	NOT NULL
	year	NUMBER	
	price	NUMBER	NOT NULL
	make	VARCHAR(255)	NOT NULL
SalesPerson	salesperson_id	NUMBER	PRIMARY KEY

	name	VARCHAR(255)	NOT NULL
	email	VARCHAR(255)	
	phone_number	VARCHAR(10)	
Transaction	transaction_id	NUMBER	PRIMARY KEY
	vehicle_id	NUMBER	NOT NULL
	customer_id	NUMBER	NOT NULL
	salesperson_id	NUMBER	NOT NULL
	transaction_date	DATE	
	amount_paid	NUMBER	
PaymentMethod	paymentmethod_id	NUMBER	PRIMARY KEY
	transaction_id	NUMBER	NOT NULL
	method_name	VARCHAR(255)	NOT NULL
ServiceType	service_id	NUMBER	PRIMARY KEY
	service_name	VARCHAR(255)	NOT NULL
	cost	NUMBER	CHECK (cost >= 0)
Supplier	supplier_id	NUMBER	PRIMARY KEY
	name	VARCHAR(255)	NOT NULL
	contact_person	VARCHAR(255)	NOT NULL
	phone_number	VARCHAR(10)	UNIQUE
	address	VARCHAR(255)	
VehicleInventory	inventory_id	NUMBER	PRIMARY KEY
	vehicle_id	NUMBER	NOT NULL
	quantity	NUMBER	NOT NULL CHECK (quantity >= 0)
ServiceAppointment	appointment_id	NUMBER	PRIMARY KEY
	customer_id	NUMBER	NOT NULL
	vehicle_id	NUMBER	NOT NULL
	service_id	NUMBER	NOT NULL
	appointment_date	DATE	
Feedback	feedback_id	NUMBER	PRIMARY KEY
	appointment_id	NUMBER	NOT NULL
	rating	NUMBER	
	comments	VARCHAR(1000)	
SalesCommission	commission_id	NUMBER	PRIMARY KEY
	salesperson_id	NUMBER	NOT NULL
	transaction_id	NUMBER	NOT NULL
	commission_amount	NUMBER	NOT NULL
	commission_rate	NUMBER	NOT NULL CHECK (commission_rate >= 0 AND commission_rate <= 1)
Part	part_id	NUMBER	PRIMARY KEY
	vehicle_id	NUMBER	NOT NULL
	supplier_id	NUMBER	NOT NULL
	part_name	VARCHAR(255)	NOT NULL

	part_number	VARCHAR(255)	UNIQUE
	price	NUMBER	NOT NULL

DDL Commands

```
--Creation of Dealership table
CREATE TABLE Dealership (
    dealership_id NUMBER PRIMARY KEY,
    name VARCHAR(255) NOT NULL,
    location VARCHAR(255) NOT NULL
);


--Creation of Customer table
CREATE TABLE Customer (
    customer_id NUMBER PRIMARY KEY,
    first_name VARCHAR(255) NOT NULL,
    last_name VARCHAR(255) NOT NULL,
    email VARCHAR(255) UNIQUE,
    phone_number VARCHAR(10) UNIQUE,
    address VARCHAR(255)
);


-- Creation of Employee table
CREATE TABLE Employee (
    employee_id NUMBER PRIMARY KEY,
    dealership_id NUMBER NOT NULL,
    position VARCHAR(255) NOT NULL,
    name VARCHAR(255) NOT NULL,
    age NUMBER,
    CONSTRAINT fk_dealership_id_emp FOREIGN KEY (dealership_id)
REFERENCES Dealership(dealership_id)
);


-- Creation of Vehicle table
CREATE TABLE Vehicle (
    vehicle_id NUMBER PRIMARY KEY,
    dealership_id NUMBER NOT NULL,
    model VARCHAR(255) NOT NULL,
    year NUMBER,
    price NUMBER NOT NULL,
    make VARCHAR(255) NOT NULL,
    CONSTRAINT fk_dealership_id_vehicle FOREIGN KEY (dealership_id)
REFERENCES Dealership(dealership_id)
);
```

```sql
-- Creation of SalesPerson table
CREATE TABLE SalesPerson (
   salesperson_id NUMBER PRIMARY KEY,
   name VARCHAR(255) NOT NULL,
   email VARCHAR(255),
   phone_number VARCHAR(10)
);


-- Creation of Transaction table
CREATE TABLE Transaction (
   transaction_id NUMBER PRIMARY KEY,
   vehicle_id NUMBER NOT NULL,
   customer_id NUMBER NOT NULL,
   salesperson_id NUMBER NOT NULL,
   transaction_date DATE,
   amount_paid NUMBER,
   CONSTRAINT fk_vehicle_id_trans FOREIGN KEY (vehicle_id) REFERENCES
Vehicle(vehicle_id),
   CONSTRAINT fk_customer_id_trans FOREIGN KEY (customer_id)
REFERENCES Customer(customer_id),
   CONSTRAINT fk_salesperson_id_trans FOREIGN KEY (salesperson_id)
REFERENCES SalesPerson(salesperson_id)
);


-- Creation of PaymentMethod table
CREATE TABLE PaymentMethod (
   paymentmethod_id NUMBER PRIMARY KEY,
   transaction_id NUMBER NOT NULL,
   method_name VARCHAR(255) NOT NULL,
   CONSTRAINT fk_transaction_id_pm FOREIGN KEY (transaction_id)
REFERENCES Transaction(transaction_id)
);


-- Creation of ServiceType table
CREATE TABLE ServiceType (
   service_id NUMBER PRIMARY KEY,
   service_name VARCHAR(255) NOT NULL,
   cost NUMBER CHECK (cost >= 0)
);


-- Creation of Supplier table
CREATE TABLE Supplier (
   supplier_id NUMBER PRIMARY KEY,
   name VARCHAR(255) NOT NULL,
   contact_person VARCHAR(255) NOT NULL,
   phone_number VARCHAR(10) UNIQUE,
   address VARCHAR(255));
```

```sql
-- Creation of VehicleInventory table
CREATE TABLE VehicleInventory (
   inventory_id NUMBER PRIMARY KEY,
   vehicle_id NUMBER NOT NULL,
   quantity NUMBER NOT NULL CHECK (quantity >= 0),
   CONSTRAINT fk_vehicle_id_inventory FOREIGN KEY (vehicle_id)
REFERENCES Vehicle(vehicle_id)
);


-- Creation of ServiceAppointment table
CREATE TABLE ServiceAppointment (
   appointment_id NUMBER PRIMARY KEY,
   customer_id NUMBER NOT NULL,
   vehicle_id NUMBER NOT NULL,
   service_id NUMBER NOT NULL,
   appointment_date DATE,
   CONSTRAINT fk_customer_id_appointment FOREIGN KEY (customer_id)
REFERENCES Customer(customer_id),
   CONSTRAINT fk_vehicle_id_appointment FOREIGN KEY (vehicle_id)
REFERENCES Vehicle(vehicle_id),
   CONSTRAINT fk_service_id_appointment FOREIGN KEY (service_id)
REFERENCES ServiceType(service_id)
);


-- Creation of Feedback table
CREATE TABLE Feedback (
   feedback_id NUMBER PRIMARY KEY,
   appointment_id NUMBER NOT NULL,
   rating NUMBER,
   comments VARCHAR(1000),
   CONSTRAINT fk_appointment_id_feedback FOREIGN KEY (appointment_id)
REFERENCES ServiceAppointment(appointment_id)
);


-- Creation of SalesCommission table
CREATE TABLE SalesCommission (
   commission_id NUMBER PRIMARY KEY,
   salesperson_id NUMBER NOT NULL,
   transaction_id NUMBER NOT NULL,
   commission_amount NUMBER NOT NULL,
   commission_rate NUMBER NOT NULL CHECK (commission_rate >= 0 AND
commission_rate <= 1),
   CONSTRAINT fk_salesperson_id_commission FOREIGN KEY (salesperson_id)
REFERENCES SalesPerson(salesperson_id),
   CONSTRAINT fk_transaction_id_commission FOREIGN KEY (transaction_id)
REFERENCES Transaction(transaction_id)
);
```

```
-- Creation of Part table
CREATE TABLE Part (
   part_id NUMBER PRIMARY KEY,
   vehicle_id NUMBER NOT NULL,
   supplier_id NUMBER NOT NULL,
   part_name VARCHAR(255) NOT NULL,
   part_number VARCHAR(255) UNIQUE,
   price NUMBER NOT NULL,
   CONSTRAINT fk_vehicle_id_part FOREIGN KEY (vehicle_id) REFERENCES
Vehicle(vehicle_id),
   CONSTRAINT fk_supplier_id_part FOREIGN KEY (supplier_id) REFERENCES
Supplier(supplier_id));
```

DML Commands

```
--Insertion of dealership data in table
INSERT INTO Dealership (dealership_id, name, location)
VALUES (1, 'Maple Leaf Motors', '123 Queen St W, Toronto, ON, M5V 1C2,
Canada');

INSERT INTO Dealership (dealership_id, name, location)
VALUES (2, 'Great White North Autos', '456 Granville St, Vancouver, BC, V6B 1V2,
Canada');

INSERT INTO Dealership (dealership_id, name, location)
VALUES (3, 'Northern Lights Motors', '789 Rue Sainte-Catherine O, Montreal, QC,
H3C 2N6, Canada');

INSERT INTO Dealership (dealership_id, name, location)
VALUES (4, 'True North Cars', '987 Rideau St, Ottawa, ON, K1N 1E5, Canada');

INSERT INTO Dealership (dealership_id, name, location)
VALUES (5, 'Canuck Motors', '654 4 Ave SW, Calgary, AB, T2P 0L2, Canada');

INSERT INTO Dealership (dealership_id, name, location)
VALUES (6, 'Polaris Autos', '321 Jasper Ave, Edmonton, AB, T5S 0H5, Canada');

INSERT INTO Dealership (dealership_id, name, location)
VALUES (7, 'North Star Motors', '741 Main St, Winnipeg, MB, R3E 0L7, Canada');

INSERT INTO Dealership (dealership_id, name, location)
VALUES (8, 'Saskatchewan Motors', '852 22nd St W, Saskatoon, SK, S7H 0W6,
Canada');

INSERT INTO Dealership (dealership_id, name, location)
VALUES (9, 'Frostbite Autos', '963 Hollis St, Halifax, NS, B3H 2G7, Canada');

INSERT INTO Dealership (dealership_id, name, location)
VALUES (10, 'Mountie Motors', '159 Government St, Victoria, BC, V8W 1P1,
Canada');


--Insertion of customer data in table
INSERT INTO Customer (customer_id, first_name, last_name, email,
phone_number, address)
VALUES (201, 'Aarav', 'Patel', 'aarav@gmail.com', '1234567890', '520 King St,
Toronto, ON, Canada');

INSERT INTO Customer (customer_id, first_name, last_name, email,
phone_number, address)
VALUES (202, 'Neha', 'Sharma', 'neha@yahoo.com', '2345678901', '123 Queen St,
Vancouver, BC, Canada');

INSERT INTO Customer (customer_id, first_name, last_name, email,
phone_number, address)
VALUES (203, 'Arjun', 'Singh', 'arjun@hotmail.com', '3456789012', '564 Bay St,
Montreal, QC, Canada');

INSERT INTO Customer (customer_id, first_name, last_name, email,
phone_number, address)
VALUES (204, 'Aisha', 'Kumar', 'aisha@gmail.com', '4567890123', '238 Yonge St,
Ottawa, ON, Canada');

INSERT INTO Customer (customer_id, first_name, last_name, email,
phone_number, address)
VALUES (205, 'Vikram', 'Gupta', 'vikram@gmail.com', '5678901234', '656 Dundas St,
Calgary, AB, Canada');

INSERT INTO Customer (customer_id, first_name, last_name, email,
phone_number, address)
VALUES (206, 'Jasmine', 'Malhotra', 'jasmine@yahoo.com', '6789012345', '879 Bloor
St, Edmonton, AB, Canada');

INSERT INTO Customer (customer_id, first_name, last_name, email,
phone_number, address)
VALUES (207, 'Aditya', 'Kaur', 'aditya@gmail.com', '7890123456', '546 Wellington St,
Winnipeg, MB, Canada');

INSERT INTO Customer (customer_id, first_name, last_name, email,
phone_number, address)
VALUES (208, 'Meera', 'Rajput', 'meera@gmail.com', '8901234567', '876 Portage
Ave, Saskatoon, SK, Canada');

INSERT INTO Customer (customer_id, first_name, last_name, email,
phone_number, address)
VALUES (209, 'Rohan', 'Das', 'rohan@hotmail.com', '9012345678', '244 Jasper Ave,
Halifax, NS, Canada');

INSERT INTO Customer (customer_id, first_name, last_name, email,
phone_number, address)
```

```
VALUES (210, 'Priyanka', 'Chopra', 'priyanka@gmail.com', '0123456789', '589
Granville St, Victoria, BC, Canada');


--Insertion of Employee data in table
INSERT INTO Employee (employee_id, dealership_id, position, name, age)
VALUES (101, 1, 'Manager', 'John Smith', 35);

INSERT INTO Employee (employee_id, dealership_id, position, name, age)
VALUES (102, 2, 'Salesperson', 'Alice Johnson', 28);

INSERT INTO Employee (employee_id, dealership_id, position, name, age)
VALUES (103, 3, 'Technician', 'Rajesh Patel', 40);

INSERT INTO Employee (employee_id, dealership_id, position, name, age)
VALUES (104, 4, 'Manager', 'Emily Smith', 37);

INSERT INTO Employee (employee_id, dealership_id, position, name, age)
VALUES (105, 5, 'Salesperson', 'Amit Sharma', 32);

INSERT INTO Employee (employee_id, dealership_id, position, name, age)
VALUES (106, 6, 'Technician', 'Sophia Williams', 45);

INSERT INTO Employee (employee_id, dealership_id, position, name, age)
VALUES (107, 7, 'Manager', 'Priya Singh', 38);

INSERT INTO Employee (employee_id, dealership_id, position, name, age)
VALUES (108, 8, 'Salesperson', 'Michael Brown', 30);

INSERT INTO Employee (employee_id, dealership_id, position, name, age)
VALUES (109, 9, 'Technician', 'Anusha Reddy', 42);

INSERT INTO Employee (employee_id, dealership_id, position, name, age)
VALUES (110, 10, 'Manager', 'David Lee', 33);

INSERT INTO Employee (employee_id, dealership_id, position, name, age)
VALUES (111, 5, 'Consultanat', 'Siddharth Bahekar', 25);

INSERT INTO Employee (employee_id, dealership_id, position, name, age)
VALUES (112, 3, 'Receptionist', 'Chitra Periya', 35);

INSERT INTO Employee (employee_id, dealership_id, position, name, age)
VALUES (113, 1, 'IT Helpdesk', 'Rachita Shah', 26);

INSERT INTO Employee (employee_id, dealership_id, position, name, age)
VALUES (114, 8, 'Finance Manager', 'Kush Parmar', 38);

INSERT INTO Employee (employee_id, dealership_id, position, name, age)
VALUES (115, 4, 'Auto detailer', 'Tapasvi Patel', 27);
```

--Insertion of Vehicle data in table

```
INSERT INTO Vehicle (vehicle_id, dealership_id, model, year, price, make)
VALUES (601, 1, 'Toyota Camry', 2022, 30000, 'Toyota');

INSERT INTO Vehicle (vehicle_id, dealership_id, model, year, price, make)
VALUES (602, 2, 'Honda Civic', 2021, 25000, 'Honda');

INSERT INTO Vehicle (vehicle_id, dealership_id, model, year, price, make)
VALUES (603, 3, 'Maruti Swift', 2023, 20000, 'Maruti');

INSERT INTO Vehicle (vehicle_id, dealership_id, model, year, price, make)
VALUES (604, 4, 'Ford Mustang', 2020, 40000, 'Ford');

INSERT INTO Vehicle (vehicle_id, dealership_id, model, year, price, make)
VALUES (605, 5, 'Hyundai Tucson', 2022, 35000, 'Hyundai');

INSERT INTO Vehicle (vehicle_id, dealership_id, model, year, price, make)
VALUES (606, 6, 'BMW 3 Series', 2023, 45000, 'BMW');

INSERT INTO Vehicle (vehicle_id, dealership_id, model, year, price, make)
VALUES (607, 7, 'Mercedes-Benz E-Class', 2021, 55000, 'Mercedes-Benz');

INSERT INTO Vehicle (vehicle_id, dealership_id, model, year, price, make)
VALUES (608, 8, 'Audi Q5', 2022, 48000, 'Audi');

INSERT INTO Vehicle (vehicle_id, dealership_id, model, year, price, make)
VALUES (609, 9, 'Toyota Highlander', 2023, 52000, 'Toyota');

INSERT INTO Vehicle (vehicle_id, dealership_id, model, year, price, make)
VALUES (610, 10, 'Chevrolet Silverado', 2020, 60000, 'Chevrolet');

INSERT INTO Vehicle (vehicle_id, dealership_id, model, year, price, make)
VALUES (611, 3, 'Honda City', 2019, 65000, 'Honda');

INSERT INTO Vehicle (vehicle_id, dealership_id, model, year, price, make)
VALUES (612, 5, 'BMW 328i', 2022, 80000, 'BMW');

INSERT INTO Vehicle (vehicle_id, dealership_id, model, year, price, make)
VALUES (613, 9, 'Fiat Punto', 2020, 60000, 'Fiat');

INSERT INTO Vehicle (vehicle_id, dealership_id, model, year, price, make)
VALUES (614, 1, 'Ford F150', 2018, 55000, 'FORD');

INSERT INTO Vehicle (vehicle_id, dealership_id, model, year, price, make)
VALUES (615, 4, 'Ford Fusion', 2021, 75000, 'FORD');
```

--Insertion of SalesPerson data in table

```
INSERT INTO SalesPerson (salesperson_id, name, email, phone_number)
VALUES (701, 'Daniel Johnson', 'daniel@gmail.com', '1234567890');
```

```
INSERT INTO SalesPerson (salesperson_id, name, email, phone_number)
VALUES (702, 'Emma Patel', 'emma@gmail.com', '2345678901');

INSERT INTO SalesPerson (salesperson_id, name, email, phone_number)
VALUES (703, 'Liam Brown', 'liam@gmail.com', '3456789012');

INSERT INTO SalesPerson (salesperson_id, name, email, phone_number)
VALUES (704, 'Olivia Smith', 'olivia@hotmail.com', '4567890123');

INSERT INTO SalesPerson (salesperson_id, name, email, phone_number)
VALUES (705, 'Noah Sharma', 'noah@outlook.com', '5678901234');

INSERT INTO SalesPerson (salesperson_id, name, email, phone_number)
VALUES (706, 'Sophia Lee', 'sophia@hotmail.com', '6789012345');

INSERT INTO SalesPerson (salesperson_id, name, email, phone_number)
VALUES (707, 'Ethan Singh', 'ethan@gmail.com', '7890123456');

INSERT INTO SalesPerson (salesperson_id, name, email, phone_number)
VALUES (708, 'Isabella Williams', 'isabella@gmail.com', '8901234567');

INSERT INTO SalesPerson (salesperson_id, name, email, phone_number)
VALUES (709, 'Lucas Reddy', 'lucas@gmail.com', '9012345678');

INSERT INTO SalesPerson (salesperson_id, name, email, phone_number)
VALUES (710, 'Ava Johnson', 'ava@yahoo.com', '0123456789');

INSERT INTO SalesPerson (salesperson_id, name, email, phone_number)
VALUES (711, 'William Patel', 'william@gmail.com', '1234567890');

INSERT INTO SalesPerson (salesperson_id, name, email, phone_number)
VALUES (712, 'Mia Brown', 'mia@gmail.com', '2345678901');

INSERT INTO SalesPerson (salesperson_id, name, email, phone_number)
VALUES (713, 'James Smith', 'james@yahoo.com', '3456789012');

INSERT INTO SalesPerson (salesperson_id, name, email, phone_number)
VALUES (714, 'Charlotte Sharma', 'charlotte@outlook.com', '4567890123');

INSERT INTO SalesPerson (salesperson_id, name, email, phone_number)
VALUES (715, 'Benjamin Williams', 'benjamin@gmail.com', '5678901234');

INSERT INTO SalesPerson (salesperson_id, name, email, phone_number)
VALUES (716, 'Amelia Lee', 'amelia@gmail.com', '6789012345');

INSERT INTO SalesPerson (salesperson_id, name, email, phone_number)
VALUES (717, 'Henry Singh', 'henry@gmail.com', '7890123456');

INSERT INTO SalesPerson (salesperson_id, name, email, phone_number)
VALUES (718, 'Ella Johnson', 'ella@gmail.com', '8901234567');
```

```
INSERT INTO SalesPerson (salesperson_id, name, email, phone_number)
VALUES (719, 'Alexander Reddy', 'alexander@hotmail.com', '9012345678');

INSERT INTO SalesPerson (salesperson_id, name, email, phone_number)
VALUES (720, 'Grace Patel', 'grace@gmail.com', '0123456789');


--Insertion of Transaction data in table
INSERT INTO Transaction (transaction_id, vehicle_id, customer_id, salesperson_id,
transaction_date, amount_paid)
VALUES (20240, 601, 201, 701, TO_DATE('2018-05-15', 'YYYY-MM-DD'), 55000);

INSERT INTO Transaction (transaction_id, vehicle_id, customer_id, salesperson_id,
transaction_date, amount_paid)
VALUES (20241, 602, 202, 702, TO_DATE('2019-07-20', 'YYYY-MM-DD'), 40000);

INSERT INTO Transaction (transaction_id, vehicle_id, customer_id, salesperson_id,
transaction_date, amount_paid)
VALUES (20242, 603, 203, 703, TO_DATE('2020-09-10', 'YYYY-MM-DD'), 62000);

INSERT INTO Transaction (transaction_id, vehicle_id, customer_id, salesperson_id,
transaction_date, amount_paid)
VALUES (20243, 604, 204, 704, TO_DATE('2021-03-25', 'YYYY-MM-DD'), 48000);

INSERT INTO Transaction (transaction_id, vehicle_id, customer_id, salesperson_id,
transaction_date, amount_paid)
VALUES (20244, 605, 205, 705, TO_DATE('2022-08-18', 'YYYY-MM-DD'), 51000);

INSERT INTO Transaction (transaction_id, vehicle_id, customer_id, salesperson_id,
transaction_date, amount_paid)
VALUES (20245, 606, 206, 706, TO_DATE('2023-02-12', 'YYYY-MM-DD'), 58000);

INSERT INTO Transaction (transaction_id, vehicle_id, customer_id, salesperson_id,
transaction_date, amount_paid)
VALUES (20246, 607, 207, 707, TO_DATE('2018-12-05', 'YYYY-MM-DD'), 70000);

INSERT INTO Transaction (transaction_id, vehicle_id, customer_id, salesperson_id,
transaction_date, amount_paid)
VALUES (20247, 608, 208, 708, TO_DATE('2019-10-30', 'YYYY-MM-DD'), 45000);

INSERT INTO Transaction (transaction_id, vehicle_id, customer_id, salesperson_id,
transaction_date, amount_paid)
VALUES (20248, 609, 209, 709, TO_DATE('2020-06-08', 'YYYY-MM-DD'), 53000);

INSERT INTO Transaction (transaction_id, vehicle_id, customer_id, salesperson_id,
transaction_date, amount_paid)
VALUES (20249, 610, 210, 710, TO_DATE('2023-11-28', 'YYYY-MM-DD'), 66000);

INSERT INTO Transaction (transaction_id, vehicle_id, customer_id, salesperson_id,
transaction_date, amount_paid)
VALUES (20250, 611, 205, 709, TO_DATE('2023-11-28', 'YYYY-MM-DD'), 66000);
```

```
INSERT INTO Transaction (transaction_id, vehicle_id, customer_id, salesperson_id,
transaction_date, amount_paid)
VALUES (20251, 613, 207, 704, TO_DATE('2019-05-28', 'YYYY-MM-DD'), 65000);

INSERT INTO Transaction (transaction_id, vehicle_id, customer_id, salesperson_id,
transaction_date, amount_paid)
VALUES (20252, 612, 202, 702, TO_DATE('2022-03-15', 'YYYY-MM-DD'), 80000);

INSERT INTO Transaction (transaction_id, vehicle_id, customer_id, salesperson_id,
transaction_date, amount_paid)
VALUES (20253, 605, 209, 703, TO_DATE('2021-11-10', 'YYYY-MM-DD'), 60000);


INSERT INTO Transaction (transaction_id, vehicle_id, customer_id, salesperson_id,
transaction_date, amount_paid)
VALUES (20254, 602, 204, 710, TO_DATE('2018-09-20', 'YYYY-MM-DD'), 55000);

INSERT INTO Transaction (transaction_id, vehicle_id, customer_id, salesperson_id,
transaction_date, amount_paid)
VALUES (20255, 603, 202, 704, TO_DATE('2024-01-12', 'YYYY-MM-DD'), 65000);


--Insertion of PaymentMethod data in table
INSERT INTO PaymentMethod (paymentmethod_id, transaction_id, method_name)
VALUES (301, 20241, 'Credit Card');

INSERT INTO PaymentMethod (paymentmethod_id, transaction_id, method_name)
VALUES (302, 20242, 'Cash');

INSERT INTO PaymentMethod (paymentmethod_id, transaction_id, method_name)
VALUES (303, 20243, 'Cheque');

INSERT INTO PaymentMethod (paymentmethod_id, transaction_id, method_name)
VALUES (304, 20244, 'Credit Card');

INSERT INTO PaymentMethod (paymentmethod_id, transaction_id, method_name)
VALUES (305, 20245, 'Cash');

INSERT INTO PaymentMethod (paymentmethod_id, transaction_id, method_name)
VALUES (306, 20246, 'Cheque');

INSERT INTO PaymentMethod (paymentmethod_id, transaction_id, method_name)
VALUES (307, 20247, 'Credit Card');

INSERT INTO PaymentMethod (paymentmethod_id, transaction_id, method_name)
VALUES (308, 20248, 'Cash');

INSERT INTO PaymentMethod (paymentmethod_id, transaction_id, method_name)
VALUES (309, 20249, 'Cheque');

INSERT INTO PaymentMethod (paymentmethod_id, transaction_id, method_name)
```

```
VALUES (310, 20240, 'Credit Card');


--Insertion of ServiceType data in table
INSERT INTO ServiceType (service_id, service_name, cost)
VALUES (401, 'Oil Change', 50);

INSERT INTO ServiceType (service_id, service_name, cost)
VALUES (402, 'Brake Replacement', 150);

INSERT INTO ServiceType (service_id, service_name, cost)
VALUES (403, 'Tire Rotation', 75);

INSERT INTO ServiceType (service_id, service_name, cost)
VALUES (404, 'Engine Tune-Up', 200);

INSERT INTO ServiceType (service_id, service_name, cost)
VALUES (405, 'Wheel Alignment', 100);

INSERT INTO ServiceType (service_id, service_name, cost)
VALUES (406, 'Battery Replacement', 120);

INSERT INTO ServiceType (service_id, service_name, cost)
VALUES (407, 'Transmission Flush', 250);

INSERT INTO ServiceType (service_id, service_name, cost)
VALUES (408, 'AC Repair', 180);

INSERT INTO ServiceType (service_id, service_name, cost)
VALUES (409, 'Diagnostic Check', 80);

INSERT INTO ServiceType (service_id, service_name, cost)
VALUES (410, 'Electrical System Check', 90);


--Insertion of Supplier data in table
INSERT INTO Supplier (supplier_id, name, contact_person, phone_number,
address)
VALUES (501, 'Red Rock Auto Parts', 'Alex', '1234567890', '123 King St, Toronto,
ON, Canada');

INSERT INTO Supplier (supplier_id, name, contact_person, phone_number,
address)
VALUES (502, 'Sunrise Supplies', 'Emma', '2345678901', '456 Queen St, Vancouver,
BC, Canada');

INSERT INTO Supplier (supplier_id, name, contact_person, phone_number,
address)
VALUES (503, 'Golden Harvest Parts', 'Charlie', '3456789012', '789 Bay St, Montreal,
QC, Canada');
```

```
INSERT INTO Supplier (supplier_id, name, contact_person, phone_number,
address)
VALUES (504, 'Eagle Eye Suppliers', 'Lily', '4567890123', '987 Yonge St, Ottawa,
ON, Canada');

INSERT INTO Supplier (supplier_id, name, contact_person, phone_number,
address)
VALUES (505, 'Thunderbird Auto Parts', 'Noah', '5678901234', '654 Dundas St,
Calgary, AB, Canada');

INSERT INTO Supplier (supplier_id, name, contact_person, phone_number,
address)
VALUES (506, 'Rainbow Supplies', 'Sophia', '6789012345', '321 Bloor St, Edmonton,
AB, Canada');

INSERT INTO Supplier (supplier_id, name, contact_person, phone_number,
address)
VALUES (507, 'Silver Arrow Parts', 'Jack', '7890123456', '741 Wellington St,
Winnipeg, MB, Canada');

INSERT INTO Supplier (supplier_id, name, contact_person, phone_number,
address)
VALUES (508, 'Mountain Peak Supplies', 'Olivia', '8901234567', '852 Portage Ave,
Saskatoon, SK, Canada');

INSERT INTO Supplier (supplier_id, name, contact_person, phone_number,
address)
VALUES (509, 'Prairie Wind Parts', 'Lucas', '9012345678', '963 Jasper Ave, Halifax,
NS, Canada');

INSERT INTO Supplier (supplier_id, name, contact_person, phone_number,
address)
VALUES (510, 'Lakeside Suppliers', 'Grace', '0123456789', '159 Granville St,
Victoria, BC, Canada');
```

--Insertion of VehicleInventory data in table

```
INSERT INTO VehicleInventory (inventory_id, vehicle_id, quantity)
VALUES (801, 601, 5);

INSERT INTO VehicleInventory (inventory_id, vehicle_id, quantity)
VALUES (802, 602, 3);

INSERT INTO VehicleInventory (inventory_id, vehicle_id, quantity)
VALUES (803, 603, 7);

INSERT INTO VehicleInventory (inventory_id, vehicle_id, quantity)
VALUES (804, 604, 2);

INSERT INTO VehicleInventory (inventory_id, vehicle_id, quantity)
VALUES (805, 605, 4);
```

```
INSERT INTO VehicleInventory (inventory_id, vehicle_id, quantity)
VALUES (806, 606, 6);

INSERT INTO VehicleInventory (inventory_id, vehicle_id, quantity)
VALUES (807, 607, 3);

INSERT INTO VehicleInventory (inventory_id, vehicle_id, quantity)
VALUES (808, 608, 5);

INSERT INTO VehicleInventory (inventory_id, vehicle_id, quantity)
VALUES (809, 609, 4);

INSERT INTO VehicleInventory (inventory_id, vehicle_id, quantity)
VALUES (810, 610, 2);


--Insertion of Service Appointment data in table
INSERT INTO ServiceAppointment (appointment_id, customer_id, vehicle_id,
service_id, appointment_date)
VALUES (901, 201, 601, 401, TO_DATE('2024-04-15', 'YYYY-MM-DD'));

INSERT INTO ServiceAppointment (appointment_id, customer_id, vehicle_id,
service_id, appointment_date)
VALUES (902, 202, 602, 402, TO_DATE('2024-04-16', 'YYYY-MM-DD'));

INSERT INTO ServiceAppointment (appointment_id, customer_id, vehicle_id,
service_id, appointment_date)
VALUES (903, 203, 603, 403, TO_DATE('2024-04-17', 'YYYY-MM-DD'));

INSERT INTO ServiceAppointment (appointment_id, customer_id, vehicle_id,
service_id, appointment_date)
VALUES (904, 204, 604, 404, TO_DATE('2024-04-18', 'YYYY-MM-DD'));

INSERT INTO ServiceAppointment (appointment_id, customer_id, vehicle_id,
service_id, appointment_date)
VALUES (905, 205, 605, 405, TO_DATE('2024-04-19', 'YYYY-MM-DD'));

INSERT INTO ServiceAppointment (appointment_id, customer_id, vehicle_id,
service_id, appointment_date)
VALUES (906, 206, 606, 406, TO_DATE('2024-04-20', 'YYYY-MM-DD'));

INSERT INTO ServiceAppointment (appointment_id, customer_id, vehicle_id,
service_id, appointment_date)
VALUES (907, 207, 607, 407, TO_DATE('2024-04-21', 'YYYY-MM-DD'));

INSERT INTO ServiceAppointment (appointment_id, customer_id, vehicle_id,
service_id, appointment_date)
VALUES (908, 208, 608, 408, TO_DATE('2024-04-22', 'YYYY-MM-DD'));

INSERT INTO ServiceAppointment (appointment_id, customer_id, vehicle_id,
service_id, appointment_date)
```

```
VALUES (909, 209, 609, 409, TO_DATE('2024-04-23', 'YYYY-MM-DD'));

INSERT INTO ServiceAppointment (appointment_id, customer_id, vehicle_id,
service_id, appointment_date)
VALUES (910, 210, 610, 410, TO_DATE('2024-04-24', 'YYYY-MM-DD'));

INSERT INTO ServiceAppointment (appointment_id, customer_id, vehicle_id,
service_id, appointment_date)
VALUES (911, 203, 613, 401, TO_DATE('2024-04-25', 'YYYY-MM-DD'));

INSERT INTO ServiceAppointment (appointment_id, customer_id, vehicle_id,
service_id, appointment_date)
VALUES (912, 207, 607, 402, TO_DATE('2024-04-26', 'YYYY-MM-DD'));

INSERT INTO ServiceAppointment (appointment_id, customer_id, vehicle_id,
service_id, appointment_date)
VALUES (913, 210, 610, 403, TO_DATE('2024-04-27', 'YYYY-MM-DD'));

INSERT INTO ServiceAppointment (appointment_id, customer_id, vehicle_id,
service_id, appointment_date)
VALUES (914, 204, 604, 404, TO_DATE('2024-04-28', 'YYYY-MM-DD'));

INSERT INTO ServiceAppointment (appointment_id, customer_id, vehicle_id,
service_id, appointment_date)
VALUES (915, 206, 606, 405, TO_DATE('2024-04-29', 'YYYY-MM-DD'));

INSERT INTO ServiceAppointment (appointment_id, customer_id, vehicle_id,
service_id, appointment_date)
VALUES (916, 202, 602, 406, TO_DATE('2024-04-30', 'YYYY-MM-DD'));

INSERT INTO ServiceAppointment (appointment_id, customer_id, vehicle_id,
service_id, appointment_date)
VALUES (917, 205, 605, 407, TO_DATE('2024-05-01', 'YYYY-MM-DD'));


--Insertion of Feedback data in table
INSERT INTO Feedback (feedback_id, appointment_id, rating, comments)
VALUES (1001, 901, 5, 'Excellent service!');

INSERT INTO Feedback (feedback_id, appointment_id, rating, comments)
VALUES (1002, 902, 4, 'Good experience overall.');

INSERT INTO Feedback (feedback_id, appointment_id, rating, comments)
VALUES (1003, 903, 3, 'Average service, could be better.');

INSERT INTO Feedback (feedback_id, appointment_id, rating, comments)
VALUES (1004, 904, 2, 'Disappointed with the service.');

INSERT INTO Feedback (feedback_id, appointment_id, rating, comments)
VALUES (1005, 905, 5, 'Highly recommended!');
```

```
INSERT INTO Feedback (feedback_id, appointment_id, rating, comments)
VALUES (1006, 906, 4, 'Satisfied with the work done.');

INSERT INTO Feedback (feedback_id, appointment_id, rating, comments)
VALUES (1007, 907, 3, 'Service was okay.');

INSERT INTO Feedback (feedback_id, appointment_id, rating, comments)
VALUES (1008, 908, 4, 'Would visit again.');

INSERT INTO Feedback (feedback_id, appointment_id, rating, comments)
VALUES (1009, 909, 5, 'Great service and friendly staff.');

INSERT INTO Feedback (feedback_id, appointment_id, rating, comments)
VALUES (1010, 910, 4, 'Happy with the outcome.');


--Insertion of Sales Commission data in table
INSERT INTO SalesCommission (commission_id, salesperson_id, transaction_id,
commission_amount, commission_rate)
VALUES (2001, 701, 20241, 5000, 0.7);

INSERT INTO SalesCommission (commission_id, salesperson_id, transaction_id,
commission_amount, commission_rate)
VALUES (2002, 702, 20242, 3500, 0.2);

INSERT INTO SalesCommission (commission_id, salesperson_id, transaction_id,
commission_amount, commission_rate)
VALUES (2003, 703, 20243, 4500, 0.9);

INSERT INTO SalesCommission (commission_id, salesperson_id, transaction_id,
commission_amount, commission_rate)
VALUES (2004, 704, 20244, 6000, 0.5);

INSERT INTO SalesCommission (commission_id, salesperson_id, transaction_id,
commission_amount, commission_rate)
VALUES (2005, 705, 20245, 4000, 0.8);

INSERT INTO SalesCommission (commission_id, salesperson_id, transaction_id,
commission_amount, commission_rate)
VALUES (2006, 706, 20246, 5500, 0.4);

INSERT INTO SalesCommission (commission_id, salesperson_id, transaction_id,
commission_amount, commission_rate)
VALUES (2007, 707, 20247, 7000, 0.1);

INSERT INTO SalesCommission (commission_id, salesperson_id, transaction_id,
commission_amount, commission_rate)
VALUES (2008, 708, 20248, 4800, 0.2);

INSERT INTO SalesCommission (commission_id, salesperson_id, transaction_id,
commission_amount, commission_rate)
```

```
VALUES (2009, 709, 20249, 5100, 0.6);

INSERT INTO SalesCommission (commission_id, salesperson_id, transaction_id,
commission_amount, commission_rate)
VALUES (2010, 710, 20240, 6300, 0.3);

INSERT INTO SalesCommission (commission_id, salesperson_id, transaction_id,
commission_amount, commission_rate)
VALUES (2011, 702, 20241, 5500, 0.6);

INSERT INTO SalesCommission (commission_id, salesperson_id, transaction_id,
commission_amount, commission_rate)
VALUES (2012, 708, 20242, 3700, 0.3);

INSERT INTO SalesCommission (commission_id, salesperson_id, transaction_id,
commission_amount, commission_rate)
VALUES (2013, 703, 20243, 4800, 0.9);

INSERT INTO SalesCommission (commission_id, salesperson_id, transaction_id,
commission_amount, commission_rate)
VALUES (2014, 707, 20244, 6200, 0.1);

INSERT INTO SalesCommission (commission_id, salesperson_id, transaction_id,
commission_amount, commission_rate)
VALUES (2015, 706, 20245, 4200, 0.6);

INSERT INTO SalesCommission (commission_id, salesperson_id, transaction_id,
commission_amount, commission_rate)
VALUES (2016, 701, 20246, 5700, 0.1);

INSERT INTO SalesCommission (commission_id, salesperson_id, transaction_id,
commission_amount, commission_rate)
VALUES (2017, 710, 20247, 7200, 0.5);


--Insertion of Part data in table
INSERT INTO Part (part_id, vehicle_id, supplier_id, part_name, part_number, price)
VALUES (3001, 601, 501, 'Oil Filter', 'OF123', 10);

INSERT INTO Part (part_id, vehicle_id, supplier_id, part_name, part_number, price)
VALUES (3002, 602, 502, 'Brake Pads', 'BP456', 30);

INSERT INTO Part (part_id, vehicle_id, supplier_id, part_name, part_number, price)
VALUES (3003, 603, 503, 'Tire', 'T789', 50);

INSERT INTO Part (part_id, vehicle_id, supplier_id, part_name, part_number, price)
VALUES (3004, 604, 504, 'Spark Plugs', 'SP101', 5);

INSERT INTO Part (part_id, vehicle_id, supplier_id, part_name, part_number, price)
VALUES (3005, 605, 505, 'Wheel', 'W202', 80);
```

```
INSERT INTO Part (part_id, vehicle_id, supplier_id, part_name, part_number, price)
VALUES (3006, 606, 506, 'Battery', 'B303', 70);

INSERT INTO Part (part_id, vehicle_id, supplier_id, part_name, part_number, price)
VALUES (3007, 607, 507, 'Transmission Fluid', 'TF404', 20);

INSERT INTO Part (part_id, vehicle_id, supplier_id, part_name, part_number, price)
VALUES (3008, 608, 508, 'AC Compressor', 'ACC505', 100);

INSERT INTO Part (part_id, vehicle_id, supplier_id, part_name, part_number, price)
VALUES (3009, 609, 509, 'Diagnostic Tool', 'DT606', 150);

INSERT INTO Part (part_id, vehicle_id, supplier_id, part_name, part_number, price)
VALUES (3010, 610, 510, 'Fuse Box', 'FB707', 15);

INSERT INTO Part (part_id, vehicle_id, supplier_id, part_name, part_number, price)
VALUES (3011, 602, 504, 'Air Filter', 'AF234', 20);

INSERT INTO Part (part_id, vehicle_id, supplier_id, part_name, part_number, price)
VALUES (3012, 605, 502, 'Headlight Bulb', 'HB567', 25);

INSERT INTO Part (part_id, vehicle_id, supplier_id, part_name, part_number, price)
VALUES (3013, 603, 509, 'Shock Absorber', 'SA890', 60);

INSERT INTO Part (part_id, vehicle_id, supplier_id, part_name, part_number, price)
VALUES (3014, 607, 503, 'Brake Caliper', 'BC112', 40);

INSERT INTO Part (part_id, vehicle_id, supplier_id, part_name, part_number, price)
VALUES (3015, 609, 505, 'Radiator', 'RD213', 90);

INSERT INTO Part (part_id, vehicle_id, supplier_id, part_name, part_number, price)
VALUES (3016, 604, 501, 'Alternator', 'ALT314', 75);

INSERT INTO Part (part_id, vehicle_id, supplier_id, part_name, part_number, price)
VALUES (3017, 608, 507, 'Power Steering Pump', 'PSP415', 110);
```

Sequences

```
--Sequence for dealership
CREATE SEQUENCE dealership_id_seq
START WITH 11
INCREMENT BY 1
NOCACHE;


--Sequence for Customer
CREATE SEQUENCE customer_id_seq
START WITH 211
INCREMENT BY 1
NOCACHE;
```

--Sequence for Employee

```
CREATE SEQUENCE employee_id_seq
START WITH 116
INCREMENT BY 1
NOCACHE;
```

Index

--Index on customer id in transaction table

```
CREATE INDEX idx_transaction_customer_id ON Transaction(customer_id);
```

--Index on salesperson id in SalesCommission table

```
CREATE INDEX idx_salescommission_salesperson_id ON
SalesCommission(salesperson_id);
```

Aggregate function

--Number of transactions done by customer

```
SELECT customer_id, COUNT(*) AS transaction_count
FROM Transaction
GROUP BY customer_id;
```

--Minimum commission rate charged by a SalesPerson

```
SELECT MIN(commission_rate) AS min_commission_rate
FROM SalesCommission;
```

--Average price of a Vehicle as per make

```
SELECT make, AVG(price) AS average_price
FROM Vehicle
GROUP BY make;
```

Complex Queries and Joins

--Transaction Details with Customer Information for Transactions above $50,000:

```
SELECT t.transaction_id, t.amount_paid, c.first_name, c.last_name
FROM Transaction t
JOIN Customer c ON t.customer_id = c.customer_id
WHERE t.amount_paid > 50000;
```

--Vehicle Inventory Details with Part Information for Parts with Price above $50:

```
SELECT vi.inventory_id, v.model, v.year, p.part_name, p.price
FROM VehicleInventory vi
```

```
JOIN Vehicle v ON vi.vehicle_id = v.vehicle_id
JOIN Part p ON v.vehicle_id = p.vehicle_id
WHERE p.price > 50;
```

--Service Appointment Details with Service Type Information for Oil Change Services:

```
SELECT sa.appointment_id, st.service_name, st.cost, sa.appointment_date
FROM ServiceAppointment sa
JOIN ServiceType st ON sa.service_id = st.service_id
WHERE st.service_name = 'Oil Change';
```

--selecting the first name and last name of customers, the name of the dealership where the transaction occurred, and the transaction date.

```
SELECT
   c.first_name, c.last_name, d.name AS dealership_name, t.transaction_date,
   c.customer_id
FROM Customer c
INNER JOIN
   Transaction t ON c.customer_id = t.customer_id
INNER JOIN
   Vehicle v ON v.vehicle_id = t.vehicle_id
LEFT join  Dealership d ON v.dealership_id = d.dealership_id
RIGHT JOIN
   SalesPerson s ON t.salesperson_id = s.salesperson_id
WHERE c.last_name LIKE 'S%';
```

--selecting the dealership name, the total number of transactions, and the total amount paid for each dealership.

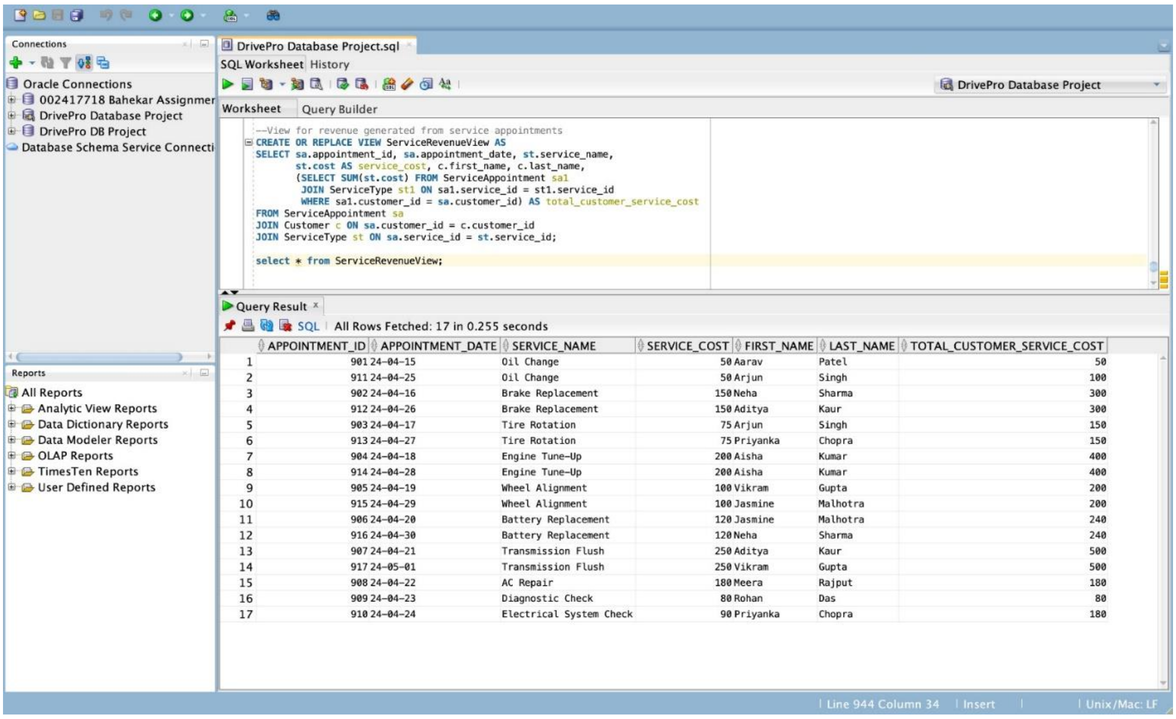
```
SELECT
   d.name AS dealership_name,
   COUNT(t.transaction_id) AS total_transactions,
   SUM(t.amount_paid) AS total_amount_paid
FROM  Dealership d
INNER JOIN
   Vehicle v ON v.dealership_id = d.dealership_id
LEFT JOIN
   Transaction t ON v.vehicle_id = t.vehicle_id
INNER JOIN
   Customer c ON t.customer_id = c.customer_id
WHERE
   c.address LIKE '%St%'
GROUP BY
   d.name
HAVING
   COUNT(t.transaction_id) > 0
ORDER BY
   total_transactions DESC;
```

VIEWS

--View for revenue generated from service appointments

```
CREATE OR REPLACE VIEW ServiceRevenueView AS
SELECT sa.appointment_id, sa.appointment_date, st.service_name,
       st.cost AS service_cost, c.first_name, c.last_name,
       (SELECT SUM(st.cost) FROM ServiceAppointment sa1
        JOIN ServiceType st1 ON sa1.service_id = st1.service_id
        WHERE sa1.customer_id = sa.customer_id) AS total_customer_service_cost
FROM ServiceAppointment sa
JOIN Customer c ON sa.customer_id = c.customer_id
JOIN ServiceType st ON sa.service_id = st.service_id;

select * from ServiceRevenueView;
```



```
--View for revenue generated from service appointments
CREATE OR REPLACE VIEW ServiceRevenueView AS
SELECT sa.appointment_id, sa.appointment_date, st.service_name,
       st.cost AS service_cost, c.first_name, c.last_name,
       (SELECT SUM(st.cost) FROM ServiceAppointment sa1
        JOIN ServiceType st1 ON sa1.service_id = st1.service_id
        WHERE sa1.customer_id = sa.customer_id) AS total_customer_service_cost
FROM ServiceAppointment sa
JOIN Customer c ON sa.customer_id = c.customer_id
JOIN ServiceType st ON sa.service_id = st.service_id;

select * from ServiceRevenueView;
```

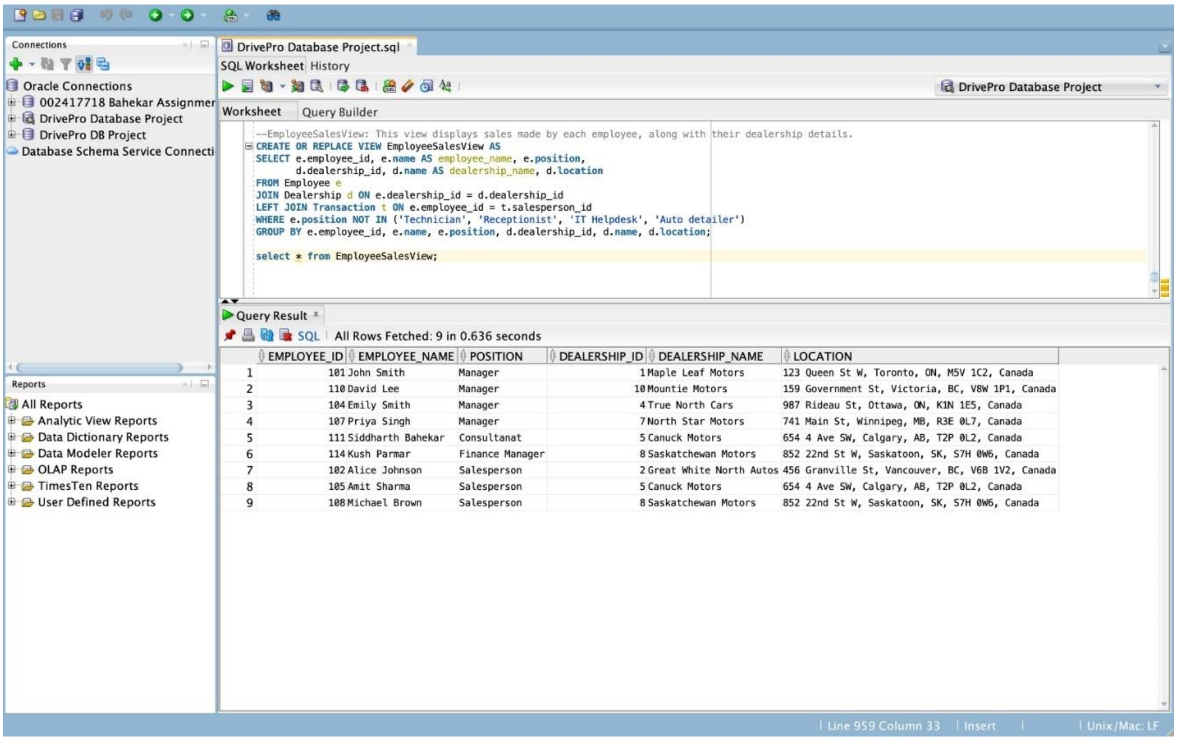
All Rows Fetched: 17 in 0.255 seconds

	APPOINTMENT_ID	APPOINTMENT_DATE	SERVICE_NAME	SERVICE_COST	FIRST_NAME	LAST_NAME	TOTAL_CUSTOMER_SERVICE_COST
1	901	24-04-15	Oil Change	50	Aarav	Patel	50
2	911	24-04-25	Oil Change	50	Arjun	Singh	100
3	902	24-04-16	Brake Replacement	150	Neha	Sharma	300
4	912	24-04-26	Brake Replacement	150	Aditya	Kaur	300
5	903	24-04-17	Tire Rotation	75	Arjun	Singh	150
6	913	24-04-27	Tire Rotation	75	Priyanka	Chopra	150
7	904	24-04-18	Engine Tune-Up	200	Aisha	Kumar	400
8	914	24-04-28	Engine Tune-Up	200	Aisha	Kumar	400
9	905	24-04-19	Wheel Alignment	100	Vikram	Gupta	200
10	915	24-04-29	Wheel Alignment	100	Jasmine	Malhotra	200
11	906	24-04-20	Battery Replacement	120	Jasmine	Malhotra	240
12	916	24-04-30	Battery Replacement	120	Neha	Sharma	240
13	907	24-04-21	Transmission Flush	250	Aditya	Kaur	500
14	917	24-05-01	Transmission Flush	250	Vikram	Gupta	500
15	908	24-04-22	AC Repair	180	Meera	Rajput	180
16	909	24-04-23	Diagnostic Check	80	Rohan	Das	80
17	910	24-04-24	Electrical System Check	90	Priyanka	Chopra	180

--EmployeeSalesView: This view displays sales made by each employee, along with their dealership details.

```
CREATE OR REPLACE VIEW EmployeeSalesView AS
SELECT e.employee_id, e.name AS employee_name, e.position,
       d.dealership_id, d.name AS dealership_name, d.location
FROM Employee e
JOIN Dealership d ON e.dealership_id = d.dealership_id
LEFT JOIN Transaction t ON e.employee_id = t.salesperson_id
WHERE e.position NOT IN ('Technician', 'Receptionist', 'IT Helpdesk', 'Auto detailer')
GROUP BY e.employee_id, e.name, e.position, d.dealership_id, d.name, d.location;
```

select * from EmployeeSalesView;



```
--EmployeeSalesView: This view displays sales made by each employee, along with their dealership details.
CREATE OR REPLACE VIEW EmployeeSalesView AS
SELECT e.employee_id, e.name AS employee_name, e.position,
       d.dealership_id, d.name AS dealership_name, d.location
FROM Employee e
JOIN Dealership d ON e.dealership_id = d.dealership_id
LEFT JOIN Transaction t ON e.employee_id = t.salesperson_id
WHERE e.position NOT IN ('Technician', 'Receptionist', 'IT Helpdesk', 'Auto detailer')
GROUP BY e.employee_id, e.name, e.position, d.dealership_id, d.name, d.location;

select * from EmployeeSalesView;
```

Query Result

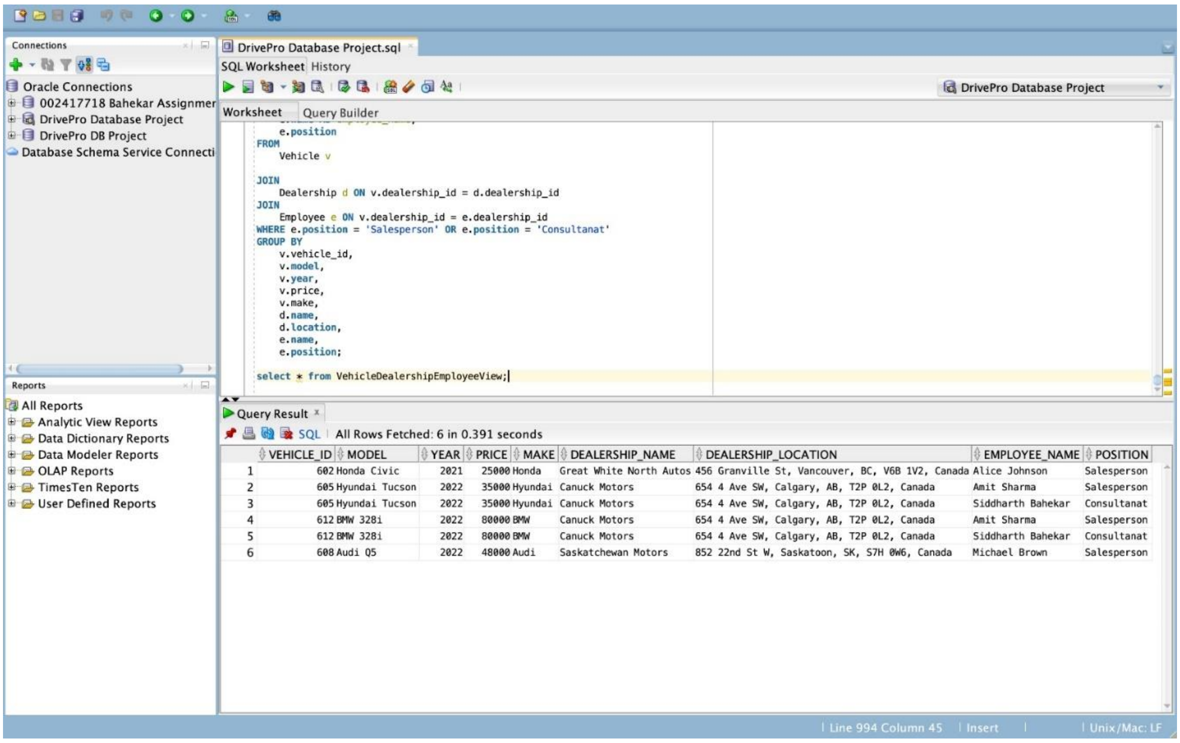
All Rows Fetched: 9 in 0.636 seconds

	EMPLOYEE_ID	EMPLOYEE_NAME	POSITION	DEALERSHIP_ID	DEALERSHIP_NAME	LOCATION
1	101	John Smith	Manager	1	Maple Leaf Motors	123 Queen St W, Toronto, ON, M5V 1C2, Canada
2	110	David Lee	Manager	10	Mountie Motors	159 Government St, Victoria, BC, V8W 1P1, Canada
3	104	Emily Smith	Manager	4	True North Cars	987 Rideau St, Ottawa, ON, K1N 1E5, Canada
4	107	Priya Singh	Manager	7	North Star Motors	741 Main St, Winnipeg, MB, R3E 0L7, Canada
5	111	Siddharth Bahekar	Consultanat	5	Canuck Motors	654 4 Ave SW, Calgary, AB, T2P 0L2, Canada
6	114	Kush Parmar	Finance Manager	8	Saskatchewan Motors	852 22nd St W, Saskatoon, SK, S7H 0W6, Canada
7	102	Alice Johnson	Salesperson	2	Great White North Autos	456 Granville St, Vancouver, BC, V6B 1V2, Canada
8	105	Amit Sharma	Salesperson	5	Canuck Motors	654 4 Ave SW, Calgary, AB, T2P 0L2, Canada
9	108	Michael Brown	Salesperson	8	Saskatchewan Motors	852 22nd St W, Saskatoon, SK, S7H 0W6, Canada

--View to combine vehicle,dealership and employee table

```
CREATE VIEW VehicleDealershipEmployeeView AS
SELECT
    v.vehicle_id,
    v.model,
    v.year,
    v.price,
    v.make,
    d.name AS dealership_name,
    d.location AS dealership_location,
    e.name AS employee_name,
    e.position
FROM Vehicle v
JOIN Dealership d ON v.dealership_id = d.dealership_id
JOIN Employee e ON v.dealership_id = e.dealership_id
WHERE e.position = 'Salesperson' OR e.position = 'Consultanat'
GROUP BY
    v.vehicle_id,v.model,v.year,v.price,v.make,d.name,d.location, e.name,e.position;

select * from VehicleDealershipEmployeeView;
```



```
        e.position
    FROM
        Vehicle v

    JOIN
        Dealership d ON v.dealership_id = d.dealership_id
    JOIN
        Employee e ON v.dealership_id = e.dealership_id
    WHERE e.position = 'Salesperson' OR e.position = 'Consultanat'
    GROUP BY
        v.vehicle_id,
        v.model,
        v.year,
        v.price,
        v.make,
        d.name,
        d.location,
        e.name,
        e.position;

    select * from VehicleDealershipEmployeeView;
```

All Rows Fetched: 6 in 0.391 seconds

	VEHICLE_ID	MODEL	YEAR	PRICE	MAKE	DEALERSHIP_NAME	DEALERSHIP_LOCATION	EMPLOYEE_NAME	POSITION
1	602	Honda Civic	2021	25000	Honda	Great White North Autos	456 Granville St, Vancouver, BC, V6B 1V2, Canada	Alice Johnson	Salesperson
2	605	Hyundai Tucson	2022	35000	Hyundai	Canuck Motors	654 4 Ave SW, Calgary, AB, T2P 0L2, Canada	Amit Sharma	Salesperson
3	605	Hyundai Tucson	2022	35000	Hyundai	Canuck Motors	654 4 Ave SW, Calgary, AB, T2P 0L2, Canada	Siddharth Bahekar	Consultanat
4	612	BMW 328i	2022	80000	BMW	Canuck Motors	654 4 Ave SW, Calgary, AB, T2P 0L2, Canada	Amit Sharma	Salesperson
5	612	BMW 328i	2022	80000	BMW	Canuck Motors	654 4 Ave SW, Calgary, AB, T2P 0L2, Canada	Siddharth Bahekar	Consultanat
6	608	Audi Q5	2022	48000	Audi	Saskatchewan Motors	852 22nd St W, Saskatoon, SK, S7H 0W6, Canada	Michael Brown	Salesperson

Applications of the Dealer Management System

- **Sales and Inventory Management:** DMS facilitates efficient tracking and management of vehicle sales, inventory levels, and dealership finances. It enables dealerships to maintain accurate records of vehicle stock, monitor sales performance, and streamline the sales process from initial inquiry to final sale

- **Customer Relationship Management (CRM):** DMS platforms often incorporate CRM functionalities, allowing dealerships to maintain comprehensive customer profiles, track interactions, and manage marketing campaigns. This enables personalized customer engagement, enhances customer satisfaction, and fosters long-term customer relationships

- **Service and Maintenance Tracking:** DMS systems assist in scheduling and tracking vehicle service appointments, managing maintenance tasks, and tracking service history. By centralizing service records and automating service reminders, dealerships can optimize service operations, improve customer service, and increase service revenue

- **Finance and Accounting:** DMS platforms include tools for managing dealership finances, such as invoicing, billing, and financial reporting. These systems integrate with accounting software to streamline financial processes, ensure compliance with regulatory requirements, and provide insights into dealership performance

- **Reporting and Analytics:** DMS solutions offer robust reporting and analytics capabilities, allowing dealerships to analyze sales trends, monitor inventory turnover, and assess profitability. By leveraging data-driven insights, dealerships can make informed business decisions, identify areas for improvement, and optimize dealership operations

- **Integration with OEMs and Third-Party Partners:** DMS platforms often integrate with original equipment manufacturers (OEMs) and third-party vendors, facilitating seamless data exchange and collaboration. Integration with OEM systems enables dealerships to access manufacturer incentives, vehicle configuration data, and promotional offers, while integration with third-party providers enhances functionality and expands service offerings

Learning Outcomes

- **Problem Analysis**: The report demonstrates a comprehensive understanding of the Automobile Dealer Management System, showcasing the capacity to identify and analyze real-world challenges within the domain effectively

- **Requirement Gathering**: Proficiency in eliciting project requirements is evident through thorough research, incorporating real-world insights, and forward-thinking considerations for future system integrations

- **Database Design**: The adept application of design principles is apparent, encompassing conceptual and logical Entity-Relationship (ER) diagrams, physical data store implementations, and adept skills in structured and normalized database design

- **Business Rules**: Clear articulation of business rules and relationships ensures alignment of the system with business objectives while upholding data integrity

- **Design Decisions**: Informed decision-making in database design, with a focus on foreign key establishment, normalization techniques, and mapping table utilization, contributes to the development of a robust and scalable system architecture

- **Data Modeling**: Effective communication of relational structures through ER diagrams facilitates the translation of complex business requirements into visual representations, facilitating comprehension for both technical and non-technical stakeholders

- **Application Impact**: Discussion on potential applications underscores an awareness of the broader implications and ramifications of the proposed Dealer Management System.

Conclusion

The automotive industry faces substantial challenges in dealership operations due to outdated systems and processes, hindering efficiency and effectiveness. These challenges include difficulties in sales management, inventory control, and leveraging data for decision-making. To overcome these hurdles, there's a pressing need for a modern Automotive Dealer Management System (DMS) that streamlines operations, adapts to evolving needs, and provides actionable insights. This system should seamlessly integrate with existing workflows, harness data effectively, and empower stakeholders.

Our approach involves extensive requirement gathering, blending theoretical knowledge with real-world insights. Our vision emphasizes revolutionizing dealership operations through advanced technology and intuitive design, aiming to modernize processes and enhance customer satisfaction. By addressing these challenges with a robust DMS solution, the automotive industry can optimize operations, improve customer experiences, and thrive in the competitive market landscape.

References

- National Automotive Dealership Association. (2018). State of the Automotive Dealership Industry Report: Trends and Challenges in Dealer Management. Retrieved from https://www.nada.org/state-of-the-industry-report

- Miller, R. (2022, June 15). "The Importance of Dealer Management Systems in Today's Automotive Landscape." AutoTech Blog. Retrieved from https://www.autotechblog.com/importance-dealer-management-systems

- Automotive Insights. (2021, December 10). "How Dealer Management Systems Are Shaping the Future of Automotive Retail." Automotive Insights Blog. Retrieved from https://www.automotiveinsightsblog.com/dealer-management-systems-future-automotive-retail

- Dealer Solutions Magazine. (2020). "The Evolution of Dealer Management Systems: From Legacy to Modern Solutions." Dealer Solutions Blog. Retrieved from https://www.dealersolutionsmagazine.com/evolution-dms-legacy-to-modern

- Automotive News. (2019, August 25). "Digital Transformation in Dealership Operations: Harnessing the Power of Modern DMS." Automotive News Blog. Retrieved from https://www.automotivenews.com/digital-transformation-dealership-operations-dms

Appendix

1. Team Charter

Team Charter – Team 1.pdf

2. Database Topic and Project Objective

Dealer Management System.pdf

3. Design and Initial ERD

Data Requirements Phase.pdf

4. DDL Scripts

DDL.sql

5. DML Scripts

DML.sql

6. Sequences and Indexes

Indexes and Sequences.sql

7. Aggregate Functions

Aggregate functions.sql

8. Complex Queries and Joins

Complex Queries and Joins.sql

9. Views

Views.sql