I. Storyline

The project's initial goal was to build an extensive database for a Mercedes dealership. The objective was to effectively handle and arrange a large volume of data about car models, their options, and customer orders.

The creation of the database schema was the initial stage of the project. Since it established the framework for the entire database system, this was an important step. Selecting the tables, their fields, and the connections between them took meticulous preparation. The dealership's policies and the type of data that required to be kept in storage served as the process' compass.

The next step was to add data to the database when the schema was in place. Insert data into the tables, required for writing SQL queries. The relationships between the tables and their structure determined how difficult these searches were.

After that, the project entered the retrieval phase, during which SQL queries were created to obtain information from the database. These searches were meant to provide specific answers to inquiries regarding the data, like the specifications of a specific car type or the total cost of all orders placed.

Numerous obstacles were faced during the undertaking. These included maintaining data integrity, comprehending intricate SQL queries, and performance-enhancing database optimization. All the same, every obstacle was a great way to learn and advance my comprehension of database management.

II. Components of Database Design

A) All entities along with their attributes and primary keys for each entity:

1) Vehicle_Model:

Primary Key: model_id

Attributes: model_name, release_year, base_price

2) Seats:

Primary Key: seat_id

Attributes: material, heating_option, price

3) Steering_Wheel:

Primary Key: wheel_id

Attributes: material, heating_option, price

4) Rims:

Primary Key: rim_id

Attributes: style, size, price

5) Vehicle_Colour:

Primary Key: colour_id

Attributes: Colour_name, hex_code, price

6) Exterior_Feature:

Primary Key: E_feature_id

Attributes: feature_name, description, price

7) Interior_Feature:

Primary Key: I_feature_id

Attributes: feature_name, description, price

8) Customer:

Primary Key: customer_id

Attributes: first_name, last_name, email

9) Configuration:

Primary Key: config_id

Attributes: model_id, seat_id, wheel_id, rim_id, colour_id

10) Order:

Primary Key: order_id

Attributes: config_id, customer_id, order_date, total_price

B) Describe all relationships among various entities:

Entity 1	Association	Entity 2	Cardinality
Vehicle_Model	Must_Have	Configuration	One to many
Configuration	Belongs_to	Seats	One to many
Configuration	Belongs_to	Steering_Wheel	One to many
Configuration	Belongs_to	Rims	One to many
Configuration	Belongs_to	Vehicle_Colour	One to many
Configuration	Belongs_to	Interior_Feature	One to many
Configuration	Belongs_to	Exterior_Feature	One to many
Customer	gives	Order	One to many
Configuration	HAS	Order	One to many

C) Participation for all relationships:

Entity1	Association	Entity2	Participation
Vehicle_Model	Must_Have	Configuration	TOTAL
Customer	gives	Order	TOTAL
Configuration	HAS	Order	TOTAL

III. Entity Relationship Diagram
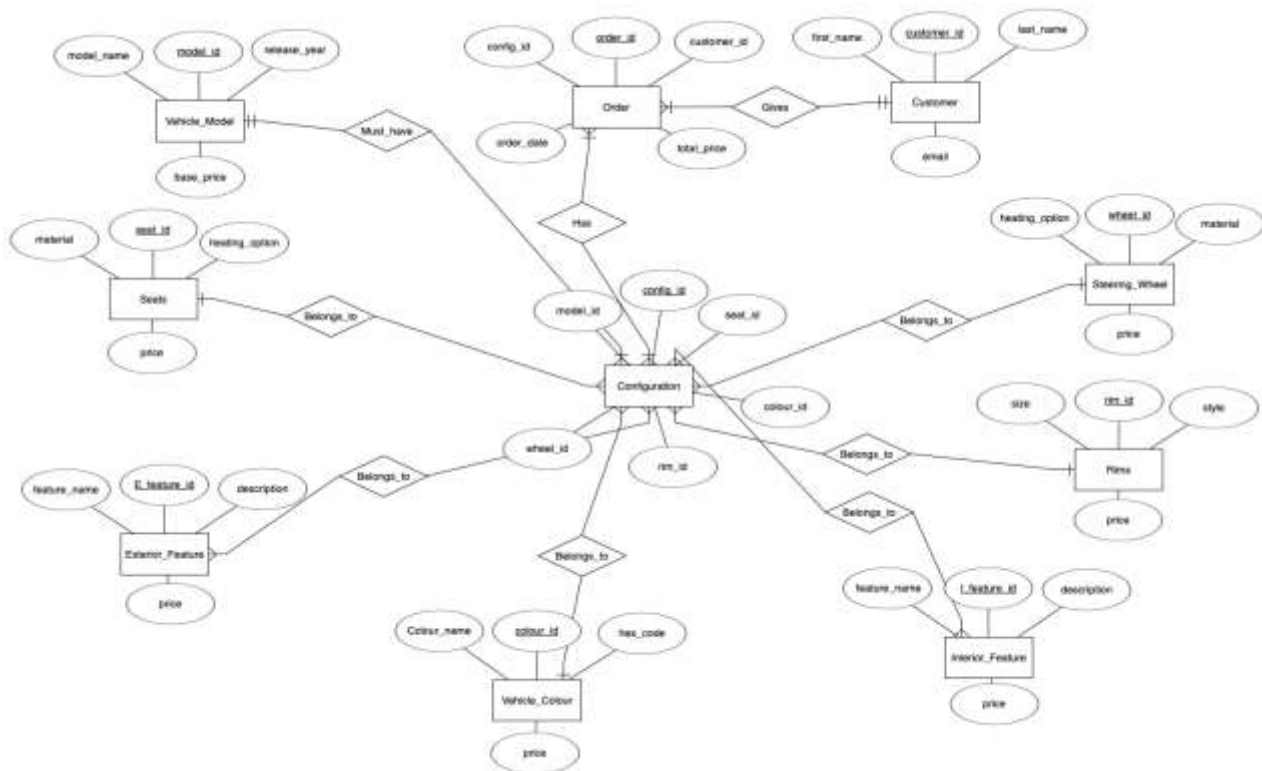
IV. Relational Model
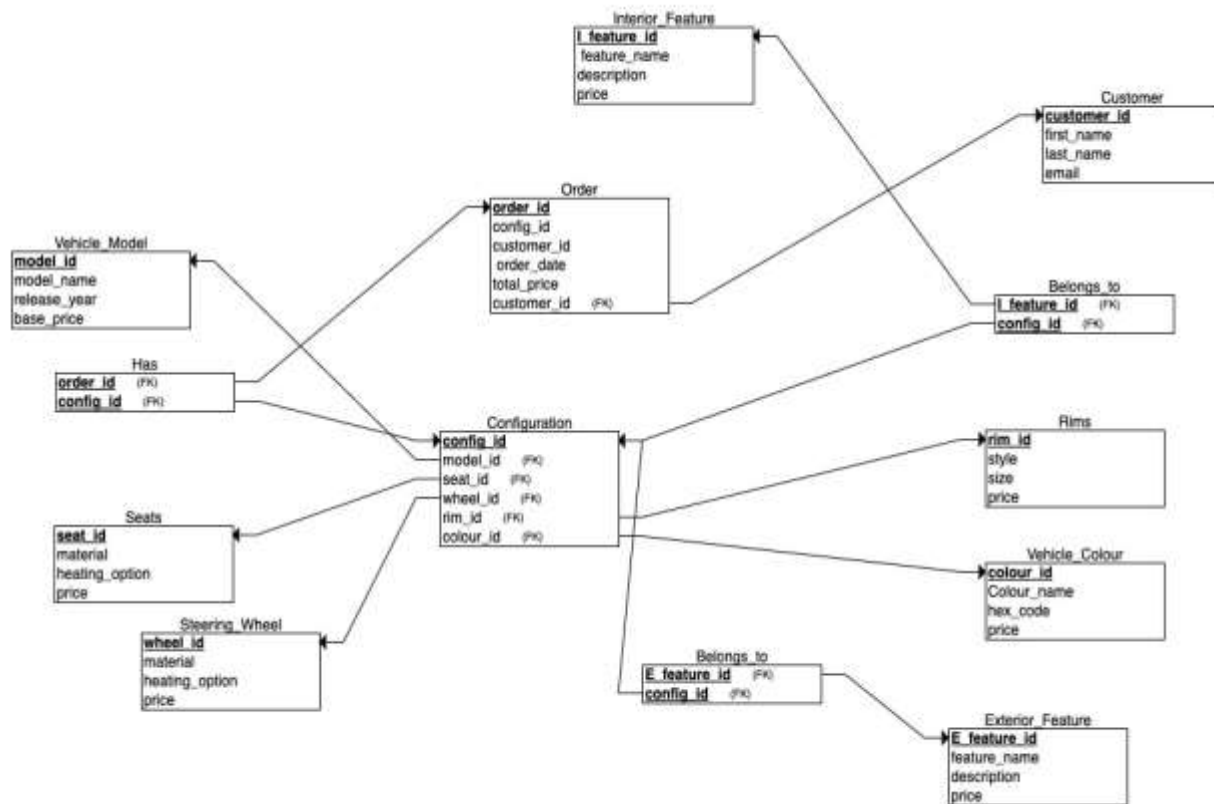
TABLES OBTAINED:

Vehicle_Model(model_id,model_name,release_year_base_price)
Seats(seat_id,material,heating_option)
Steering_Wheel(wheel_id,material, heating_option, price)
Interior_Feature(I_feature_id,feature_name,description,price)
Exterior_Feature(E_feature_id,feature_name, description, price)
Rims(rim_id,style, size, price)
Vehicle_Colour(colour_id,Colour_name, hex_code, price)
Has(order_id,config_id)
Belongs_to(E_feature_id,config_id)
Belongs_to(I_feature_id,config_id)
Order(order_id,config_id, customer_id, order_date, total_price)
Customer(customer_id,first_name,last_name,email)
Configuration(config_id, model_id, seat_id, wheel_id, rim_id, colour_id)

V. Normalization

(1) Vehicle _ Model (model-id, model-name, release-year, base-price)

Relations:
a) model-id → model-name, release-year, base-price
b) model-name → release-year, base-price

→ 1NF — yes — [Atomicity]
→ 2NF — yes — [no partial dependency]
→ 3NF — yes — [no transitivity]
→ BCNF — yes — [both the superkey derives release-year and base-price]

(2) Seats (Seat-id, material, heating-option, price)

Relations:
a) seat-id → material, heating-option, price
b) material → price
c) heating-option → price

→ 1NF — yes — [Atomicity]
→ 2MF — yes — [No partial dependency]
→ 3NF — yes — [No transitivity]
→ BCNF — yes — [LHS are superkeys which derives price attribute]

(3) Steering_wheel (wheel-id, material, heating-option, price)

Relations:
a) wheel-id → material, heating-option, price
b) material → price
c) heating-option → price

→ 1NF — yes — [Atomicity]
→ 2NF — yes — [No partial dependency]
→ 3NF — yes — [No transitivity]
→ BCNF — yes — [Superkeys on LHS determines price]

(4) Rims (rim_id, style, size, price)
   Relations —
   a) rim_id → style, size, price
   b) Style, size → price

   → 1NF - yes - [Atomicity]
   → 2NF - yes - [No partial dependency]
   → 3NF - yes - [No transitivity]
   → BCNF - yes - [rim-id is the candidate Key on LHS]
   2nd relation does not violate BCNF because combination style, size serves as a superkey.

(5) Vehicle_Colour (colour-id, colour_name, hex-code, price)
   Relations —
   a) colour_id → colour_name, hex-code, price
   b) colour_name → price
   c) hex-code → price

   → 1NF - yes - [Atomicity]
   → 2NF - No - [Partial dependencies]
   decompose —
       R₁ (colour-id, colour_name, hex-code)
       R₂ (colour-id, price)
   → 3NF (using decompose table)
       yes - [No transitivity]
   → BCNF - yes - [Colour-id is the candidate key and is a superkey in LHS]

6) Braking_Feature (I_feature_id, feature_name, description, price)

   Relations:
   i) I_feature_id → feature_name, description, price
   ii) feature_name → price
   iii) description → price

   1NF - yes - [Atomicity]
   2NF - yes - [No partial dependencies]
   3NF - yes - [No transitivity]

   BCNF - yes -

   Since all our FDs are based on the primary key (I_feature_id), the table satisfies BCNF.

b) Interior_feature (I_feature_id, feature_name, description, price)

Relations:

1) I_feature_id → (feature_name, description, price)

2) feature_name → price

3) description → price

INF - yes - [atomicity]

2NF - No Yes - [No partial dependencies]

3NF - yes - [No Transitivity]

BCNF - yes -

    [requires that every non-trivial FD is a superke

• Since all our FDs are based on the primary key (I_feature_id), the table satisfies BCNF.

8) Customer (customer_id, first_name, last_name, email)

Relations:

i) customer_id → first_name, last_name, email)

ii) email → customer_id, first_name, last_name)

INF - yes - [Atomicity]
2NF - Yes - [No partial dependencies]
3NF - No Yes - [No Transitivity]
BCNF - NO
Decompose-

    R1 (customer_id, first_name, last_name)
    R2 (email, customer_id, email)

9) Configuration(config_id , model_id , seat_id, wheel_id, aim_id, colour_id.

Relations:

i) config_id → (model_id, seat_id, wheel_id, aim_id, colour_id)

ii) model_id → (seat_id, wheel_id, aim_id, colour_id)

iii) seat_id, wheel_id, aim_id, colour_id → config_id

1NF - yes - [Atomicity]
2NF - yes - [Partial Dependency]
3NF - yes - [No Transitivity]
BCNF - NO

Decompose:

R1 (config_id , model_id, seat_id, wheel_id, aim_id, colour_id)

R2 (model_id, seat_id, wheel_id, aim_id, colour_id)

---

10) Order (order_id, config_id, customer_id, order_date, total_price).

Relations:

i) order_id → {config_id, customer_id, order_date, total_price.

2) config_id → total_price

3) customer_id → order_date.

1NF: Yes - [Atomicity].
2NF - No - [Partial Dependency]

Decompose-
R1 (order_id, customer_id, order_date, config_id)
R2 (config_id, total_price).

R 3NF? — No —

Decompose:

R1 (order_id, config_id)
R2 (config_id, total_price).

BCNF: - NO -

R1 (order_id, customer_id, order_date, config_id)
R2 (config_id, total_price)

VI. SQL Queries

- Create the tables

```sql
CREATE DATABASE Mercedes;
USE Mercedes;

CREATE TABLE Vehicle_Model (
    model_id INT PRIMARY KEY,
    model_name VARCHAR(255),
    release_year INT,
    base_price DECIMAL(10, 2)
);

CREATE TABLE Seats (
    seat_id INT PRIMARY KEY,
    material VARCHAR(255),
    heating_option BOOLEAN,
    price DECIMAL(10, 2)
);

CREATE TABLE Steering_Wheel (
    wheel_id INT PRIMARY KEY,
    material VARCHAR(255),
    heating_option BOOLEAN,
    price DECIMAL(10, 2)
);

CREATE TABLE Rims (
    rim_id INT PRIMARY KEY,
    style VARCHAR(255),
    size INT,
    price DECIMAL(10, 2)
);

CREATE TABLE Vehicle_Colour (
    colour_id INT PRIMARY KEY,
    Colour_name VARCHAR(255),
    hex_code VARCHAR(7),
    price DECIMAL(10, 2)
);
```

```sql
CREATE TABLE Exterior_Feature (
    E_feature_id INT PRIMARY KEY,
    feature_name VARCHAR(255),
    description TEXT,
    price DECIMAL(10, 2)
);

CREATE TABLE Interior_Feature (
    I_feature_id INT PRIMARY KEY,
    feature_name VARCHAR(255),
    description TEXT,
    price DECIMAL(10, 2)
);

CREATE TABLE Customer (
    customer_id INT PRIMARY KEY,
    first_name VARCHAR(255),
    last_name VARCHAR(255),
    email VARCHAR(255)
);

CREATE TABLE Configuration (
    config_id INT PRIMARY KEY,
    model_id INT,
    seat_id INT,
    wheel_id INT,
    rim_id INT,
    colour_id INT,
    FOREIGN KEY (model_id) REFERENCES Vehicle_Model(model_id),
    FOREIGN KEY (seat_id) REFERENCES Seats(seat_id),
    FOREIGN KEY (wheel_id) REFERENCES Steering_Wheel(wheel_id),
    FOREIGN KEY (rim_id) REFERENCES Rims(rim_id),
    FOREIGN KEY (colour_id) REFERENCES Vehicle_Colour(colour_id)
);

CREATE TABLE `Order` (
    order_id INT PRIMARY KEY,
    config_id INT,
    customer_id INT,
    order_date DATE,
```

```
    total_price DECIMAL(10, 2),
    FOREIGN KEY (config_id) REFERENCES Configuration(config_id),
    FOREIGN KEY (customer_id) REFERENCES Customer(customer_id)
);
```

- Populate the tables (20 tuples for each relation)

```
INSERT INTO Vehicle_Model (model_id, model_name, release_year, base_price)
VALUES
    (1, 'A-Class', 2018, 33000),
    (2, 'B-Class', 2019, 35000),
    (3, 'C-Class', 2020, 41000),
    (4, 'CLA', 2021, 37000),
    (5, 'CLS', 2022, 70000),
    (6, 'E-Class', 2023, 54000),
    (7, 'G-Class', 2024, 130000),
    (8, 'GLA', 2025, 36000),
    (9, 'GLB', 2026, 38000),
    (10, 'GLC', 2027, 42000),
    (11, 'GLE', 2028, 54000),
    (12, 'GLS', 2029, 76000),
    (13, 'S-Class', 2030, 94000),
    (14, 'SL', 2031, 91000),
    (15, 'SLC', 2032, 49000),
    (16, 'AMG GT', 2033, 115000),
    (17, 'AMG GT 4-Door', 2034, 89000),
    (18, 'EQC', 2035, 67000),
    (19, 'EQS', 2036, 102000),
    (20, 'EQV', 2037, 71000);

-- Inserting 20 records into Seats
INSERT INTO Seats (seat_id, material, heating_option, price)
VALUES
    (1, 'Leather', TRUE, 5000),
    (2, 'Alcantara', TRUE, 6000),
    (3, 'Carbon Fiber', FALSE, 8000),
    (4, 'Fabric', FALSE, 3000),
    (5, 'Synthetic Leather', TRUE, 4000),
    (6, 'Velour', FALSE, 3500),
    (7, 'Vinyl', FALSE, 3000),
    (8, 'Wool', TRUE, 4500),
```

```sql
    (9, 'Microfiber', TRUE, 5500),
    (10, 'Suede', FALSE, 7000),
    (11, 'Nappa Leather', TRUE, 6000),
    (12, 'Perforated Leather', TRUE, 6500),
    (13, 'Pigmented Leather', FALSE, 5500),
    (14, 'Semi-Aniline Leather', TRUE, 7000),
    (15, 'Top-Grain Leather', TRUE, 7500),
    (16, 'Full-Grain Leather', FALSE, 8000),
    (17, 'Corrected-Grain Leather', TRUE, 5500),
    (18, 'Split Leather', FALSE, 5000),
    (19, 'Bonded Leather', TRUE, 4500),
    (20, 'Faux Leather', FALSE, 4000);

-- Inserting 20 records into Steering_Wheel
INSERT INTO Steering_Wheel (wheel_id, material, heating_option, price)
VALUES
    (1, 'Leather', TRUE, 3000),
    (2, 'Alcantara', TRUE, 3500),
    (3, 'Carbon Fiber', FALSE, 5000),
    (4, 'Wood', FALSE, 4000),
    (5, 'Plastic', FALSE, 2000),
    (6, 'Metal', FALSE, 2500),
    (7, 'Rubber', TRUE, 3000),
    (8, 'Synthetic', TRUE, 3500),
    (9, 'Vinyl', FALSE, 3000),
    (10, 'Cloth', FALSE, 2500),
    (11, 'Polyurethane', TRUE, 4000),
    (12, 'Silicone', TRUE, 4500),
    (13, 'Gel', FALSE, 4000),
    (14, 'Neoprene', TRUE, 4500),
    (15, 'Microfiber', TRUE, 5000),
    (16, 'Suede', FALSE, 5500),
    (17, 'Velvet', TRUE, 6000),
    (18, 'Cotton', FALSE, 3500),
    (19, 'Linen', TRUE, 4000),
    (20, 'Hemp', FALSE, 3500);

-- Inserting 20 records into Rims
INSERT INTO Rims (rim_id, style, size, price)
VALUES
```

```
  (1, 'Sport', 19, 6000),
  (2, 'Track', 20, 8000),
  (3, 'Forged', 21, 10000),
  (4, 'Alloy', 18, 5000),
  (5, 'Steel', 17, 4000),
  (6, 'Chrome', 20, 7000),
  (7, 'Wire', 19, 6000),
  (8, 'Spoke', 18, 5000),
  (9, 'Multi-Spoke', 20, 7000),
  (10, 'Mesh', 19, 6000),
  (11, 'Fan Blade', 18, 5000),
  (12, 'Turbine', 20, 7000),
  (13, 'Rally', 19, 6000),
  (14, 'Off-Road', 18, 5000),
  (15, 'Performance', 20, 7000),
  (16, 'Luxury', 19, 6000),
  (17, 'Vintage', 18, 5000),
  (18, 'Classic', 20, 7000),
  (19, 'Modern', 19, 6000),
  (20, 'Custom', 18, 5000);

-- Inserting 20 records into Vehicle_Colour
INSERT INTO Vehicle_Colour (colour_id, Colour_name, hex_code, price)
VALUES
  (1, 'Black', '#000000', 0),
  (2, 'White', '#FFFFFF', 0),
  (3, 'Red', '#FF0000', 2000),
  (4, 'Blue', '#0000FF', 2000),
  (5, 'Green', '#008000', 2000),
  (6, 'Yellow', '#FFFF00', 2000),
  (7, 'Purple', '#800080', 2000),
  (8, 'Orange', '#FFA500', 2000),
  (9, 'Pink', '#FFC0CB', 2000),
  (10, 'Brown', '#A52A2A', 2000),
  (11, 'Gray', '#808080', 0),
  (12, 'Silver', '#C0C0C0', 0),
  (13, 'Gold', '#FFD700', 3000),
  (14, 'Beige', '#F5F5DC', 2000),
  (15, 'Ivory', '#FFFFF0', 2000),
  (16, 'Teal', '#008080', 2000),
```

```sql
    (17, 'Maroon', '#800000', 2000),
    (18, 'Lime', '#00FF00', 2000),
    (19, 'Indigo', '#4B0082', 2000),
    (20, 'Violet', '#EE82EE', 2000);

-- Inserting 20 records into Exterior_Feature
INSERT INTO Exterior_Feature (E_feature_id, feature_name, description, price)
VALUES
    (1, 'Sunroof', 'A sunroof is a movable panel that opens to uncover a window in an automobile roof, allowing light and/or fresh air to enter the passenger compartment.', 2000),
    (2, 'Spoiler', 'A spoiler is an automotive aerodynamic device whose intended design function is to "spoil" unfavorable air movement across a body of a vehicle in motion.', 1500),
    (3, 'Roof Rack', 'A roof rack is a set of bars secured to the roof of a motor car. It is used to carry bulky items such as luggage, bicycles, canoes, kayaks, skis, or various carriers and containers.', 1000),
    (4, 'Running Boards', 'Running boards are narrow steps fitted under the side doors of a car or truck. They aid entry into the vehicle and give better access to the roof.', 1200),
    (5, 'Tow Hitch', 'A tow hitch is a device attached to the chassis of a vehicle for towing, or a towbar to an aircraft nose gear.', 1300),
    (6, 'Fog Lights', 'Fog lights are positioned low on the vehicle and have a wide beam angled toward the ground, which reduces glare in fog and improves visibility.', 1400),
    (7, 'Rain Sensing Wipers', 'Rain sensing wipers automatically activate and adjust their speed based on the amount of water on the windshield.', 1500),
    (8, 'Rear Window Defroster', 'A rear window defroster is a thin electrical filament, line or grid on the rear window of a vehicle, which when heated, eliminates mist.', 1600),
    (9, 'LED Headlights', 'LED headlights provide a brighter, crisper and whiter light output, closer to the color temperature of daylight, dramatically improving light projection distance and overall visibility.', 1700),
    (10, 'Power Mirrors', 'Power mirrors are outside mirrors that can be adjusted remotely with controls inside the vehicle.', 1800),
    (11, 'Alloy Wheels', 'Alloy wheels are wheels that are made from an alloy of aluminium or magnesium. They are typically lighter for the same strength and provide better heat conduction.', 1900),
    (12, 'Tinted Windows', 'Tinted windows have a thin film that reduces the transmission of light through the glass.', 2000),
    (13, 'Heated Mirrors', 'Heated mirrors use a heating element to remove fog or frost from the mirror.', 2100),
    (14, 'Chrome Door Handles', 'Chrome door handles are door handles made of chrome for aesthetic appeal.', 2200),
    (15, 'Mud Flaps', 'Mud flaps are used in combination with the vehicle fender to protect the vehicle, passengers, other vehicles, and pedestrians from mud and other flying debris.', 2300),
```

(16, 'Roof Spoiler', 'A roof spoiler is an aerodynamic device that is usually attached to the rear of the car's roof. It can be functional and/or decorative.', 2400),
    (17, 'Side Skirts', 'Side skirts are used to reduce the amount of high pressure area on the side of the car to go under the car from the sides.', 2500),
    (18, 'Front Bumper Lip', 'A front bumper lip is installed at the bottom of the front bumper, usually made of rubber or ABS plastic, and serves both decorative and practical purposes.', 2600),
    (19, 'Rear Diffuser', 'A rear diffuser is a shaped section of the car underbody which improves the car"s aerodynamic properties by enhancing the transition between the high-velocity airflow underneath the car and the much slower freestream airflow of the ambient atmosphere.', 2700),
(20, 'Hood Scoop', 'A hood scoop is an upraised component on the hood of a vehicle that either allows a flow of air to directly enter the engine compartment, or appears to do so.', 2800);

-- Inserting 20 records into Interior_Feature
INSERT INTO Interior_Feature (I_feature_id, feature_name, description, price)
VALUES
    (1, 'Air Conditioning', 'Air conditioning is a system for controlling the humidity, ventilation, and temperature in a vehicle.', 2000),
    (2, 'Heated Seats', 'Heated seats are a standard feature on many vehicles theses days. They provide comfort and warmth.', 1500),
    (3, 'Leather Seats', 'Leather seats are a luxurious, comfortable, and durable option for your car.', 1000),
    (4, 'Power Seats', 'Power seats are a feature in cars where the seats can be adjusted by using a switch or joystick and a set of small electric motors.', 1200),
    (5, 'Navigation System', 'A navigation system is a computer that is capable of combining information from a variety of sources and presenting it in a useful format.', 1300),
    (6, 'Bluetooth', 'Bluetooth is a wireless technology that allows two devices to communicate with each other. In cars, it is commonly used to allow a mobile phone to communicate with the car's audio system.', 1400),
    (7, 'Backup Camera', 'A backup camera is a special type of video camera that is produced specifically for the purpose of being attached to the rear of a vehicle to aid in backing up.', 1500),
    (8, 'Cruise Control', 'Cruise control is a system that automatically controls the speed of a motor vehicle.', 1600),
    (9, 'Keyless Entry', 'Keyless entry allows you to lock, unlock, and start your vehicle without ever having to take your key out of your pocket.', 1700),
    (10, 'USB Ports', 'USB ports in a car can be used to charge a phone, connect a music device, or even connect to the car's entertainment system.', 1800),
    (11, 'Wireless Charging Pad', 'A wireless charging pad allows you to charge your phone without the need for wires.', 1900),
    (12, 'Ambient Lighting', 'Ambient lighting can be found on the dashboard, center console, door handles, footwells, and other areas of the car.', 2000),

(13, 'Heated Steering Wheel', 'A heated steering wheel is a great feature for those cold winter days.', 2100),
    (14, 'Power Windows', 'Power windows or electric windows are automobile windows which can be raised and lowered by pressing a button or switch.', 2200),
    (15, 'Auto-Dimming Rearview Mirror', 'An auto-dimming rearview mirror is a type of rear-view mirror that automatically dims when it is dark outside and bright lights are shone into it.', 2300),
    (16, 'Satellite Radio', 'Satellite radio is a service that delivers radio programming via a direct broadcast satellite.', 2400),
    (17, 'Premium Audio System', 'A premium audio system includes multiple speakers and a powerful amplifier. It delivers a superior audio experience.', 2500),
    (18, 'WiFi Hotspot', 'A WiFi hotspot is a physical location where people can access the Internet, typically using Wi-Fi, via a wireless local area network with a router connected to an internet service provider.', 2600),
    (19, 'Rain Sensing Wipers', 'Rain sensing wipers automatically activate and adjust their speed based on the amount of water on the windshield.', 2700),
    (20, 'Dual-Zone Climate Control', 'Dual-zone climate control allows the driver and front passenger to set their own individual temperature settings.', 2800);

-- Inserting 20 records into Customer
INSERT INTO Customer (customer_id, first_name, last_name, email)
VALUES
    (1, 'John', 'Doe', 'john.doe@example.com'),
    (2, 'Jane', 'Doe', 'jane.doe@example.com'),
    (3, 'Alice', 'Smith', 'alice.smith@example.com'),
    (4, 'Bob', 'Johnson', 'bob.johnson@example.com'),
    (5, 'Charlie', 'Williams', 'charlie.williams@example.com'),
    (6, 'David', 'Brown', 'david.brown@example.com'),
    (7, 'Eve', 'Jones', 'eve.jones@example.com'),
    (8, 'Frank', 'Miller', 'frank.miller@example.com'),
    (9, 'Grace', 'Davis', 'grace.davis@example.com'),
    (10, 'Harry', 'Garcia', 'harry.garcia@example.com'),
    (11, 'Ivy', 'Rodriguez', 'ivy.rodriguez@example.com'),
    (12, 'Jack', 'Wilson', 'jack.wilson@example.com'),
    (13, 'Kate', 'Martinez', 'kate.martinez@example.com'),
    (14, 'Luke', 'Anderson', 'luke.anderson@example.com'),
    (15, 'Mia', 'Taylor', 'mia.taylor@example.com'),
    (16, 'Noah', 'Thomas', 'noah.thomas@example.com'),
    (17, 'Olivia', 'Hernandez', 'olivia.hernandez@example.com'),
    (18, 'Peter', 'Moore', 'peter.moore@example.com'),
    (19, 'Queen', 'Martin', 'queen.martin@example.com'),

```sql
   (20, 'Robert', 'Jackson', 'robert.jackson@example.com');


-- Inserting 20 records into Configuration
INSERT INTO Configuration (config_id, model_id, seat_id, wheel_id, rim_id, colour_id)
VALUES
   (1, 1, 1, 1, 1, 1),
   (2, 2, 2, 2, 2, 2),
   (3, 3, 3, 3, 3, 3),
   (4, 4, 4, 4, 4, 4),
   (5, 5, 5, 5, 5, 5),
   (6, 6, 6, 6, 6, 6),
   (7, 7, 7, 7, 7, 7),
   (8, 8, 8, 8, 8, 8),
   (9, 9, 9, 9, 9, 9),
   (10, 10, 10, 10, 10, 10),
   (11, 11, 11, 11, 11, 11),
   (12, 12, 12, 12, 12, 12),
   (13, 13, 13, 13, 13, 13),
   (14, 14, 14, 14, 14, 14),
   (15, 15, 15, 15, 15, 15),
   (16, 16, 16, 16, 16, 16),
   (17, 17, 17, 17, 17, 17),
   (18, 18, 18, 18, 18, 18),
   (19, 19, 19, 19, 19, 19),
   (20, 20, 20, 20, 20, 20);

-- Inserting 20 records into Order
INSERT INTO `Order` (order_id, config_id, customer_id, order_date, total_price)
VALUES
   (1, 1, 1, '2024-01-01', 35000),
   (2, 2, 2, '2024-01-02', 37000),
   (3, 3, 3, '2024-01-03', 43000),
   (4, 4, 4, '2024-01-04', 39000),
   (5, 5, 5, '2024-01-05', 72000),
   (6, 6, 6, '2024-01-06', 56000),
   (7, 7, 7, '2024-01-07', 132000),
   (8, 8, 8, '2024-01-08', 38000),
   (9, 9, 9, '2024-01-09', 40000),
   (10, 10, 10, '2024-01-10', 44000),
   (11, 11, 11, '2024-01-11', 56000),
```
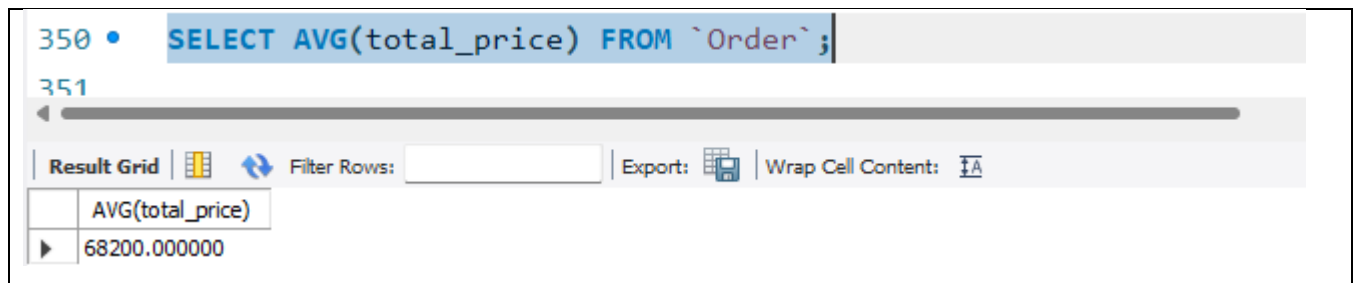
```
    (12, 12, 12, '2024-01-12', 78000),
    (13, 13, 13, '2024-01-13', 96000),
    (14, 14, 14, '2024-01-14', 93000),
    (15, 15, 15, '2024-01-15', 51000),
    (16, 16, 16, '2024-01-16', 117000),
    (17, 17, 17, '2024-01-17', 91000),
    (18, 18, 18, '2024-01-18', 69000),
    (19, 19, 19, '2024-01-19', 104000),
    (20, 20, 20, '2024-01-20', 73000);
```

- Run SQL queries (minimum 20) covering **all concepts** learned in the class

Query1: What is the average total price of the orders placed?
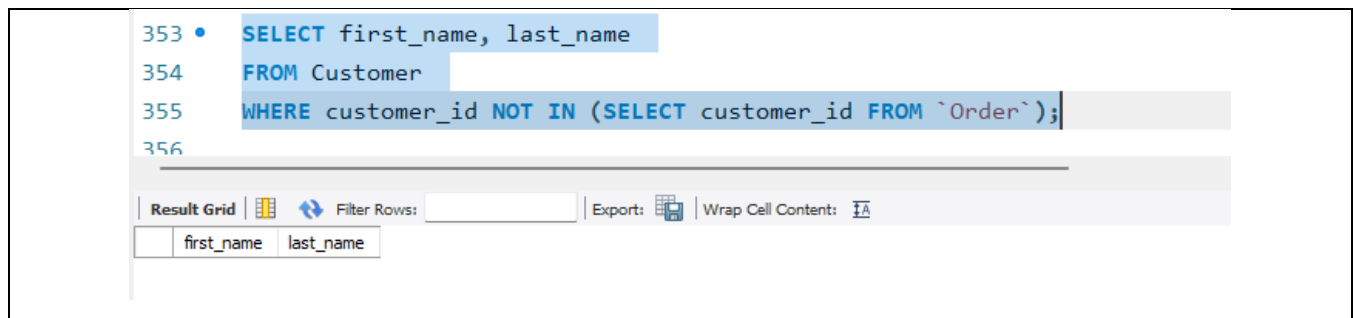
```
350 •    SELECT AVG(total_price) FROM `Order`;
351
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

AVG(total_price)
68200.000000

Query2: Which customers have not placed any orders?

```
353 •    SELECT first_name, last_name
354      FROM Customer
355      WHERE customer_id NOT IN (SELECT customer_id FROM `Order`);
356
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

first_name	last_name

Query3: What are the different materials used for steering wheels and how many configurations use each material?

```
358 •  SELECT Steering_Wheel.material, COUNT(Configuration.config_id)
359    FROM Steering_Wheel
360    JOIN Configuration ON Steering_Wheel.wheel_id = Configuration.wheel_id
361    GROUP BY Steering_Wheel.material;
362
```

material	COUNT(Configuration.config_id)
Leather	1
Alcantara	1
Carbon Fiber	1
Wood	1
Plastic	1
Metal	1
Rubber	1
Synthetic	1
Vinyl	1
Cloth	1
Polyurethane	1
Silicone	1
Gel	1
Neoprene	1
Microfiber	1
Suede	1
Velvet	1
Cotton	1
Linen	1
Hemp	1

Query 4: Which vehicle models have been ordered with a 'Leather' steering wheel?

```
364 •  SELECT Vehicle_Model.model_name
365    FROM Vehicle_Model
366    JOIN Configuration ON Vehicle_Model.model_id = Configuration.model_id
367    JOIN Steering_Wheel ON Configuration.wheel_id = Steering_Wheel.wheel_id
368    JOIN `Order` ON Configuration.config_id = `Order`.config_id
369    WHERE Steering_Wheel.material = 'Leather';
370
```

model_name
A-Class

Query5: How many configurations use rims of size 19?

```
371    -- How many configurations use rims of size 19?
372 •  SELECT COUNT(config_id)
373    FROM Configuration
374    WHERE rim_id IN (SELECT rim_id FROM Rims WHERE size = 19);
```

COUNT(config_id)
6

Query 6: Which customers have ordered a configuration with the 'Red' vehicle color?

```
377 •   SELECT Customer.first_name, Customer.last_name
378      FROM Customer
379      JOIN `Order` ON Customer.customer_id = `Order`.customer_id
380      JOIN Configuration ON `Order`.config_id = Configuration.config_id
381      JOIN Vehicle_Colour ON Configuration.colour_id = Vehicle_Colour.colour_id
382      WHERE Vehicle_Colour.Colour_name = 'Red';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content

first_name	last_name
Alice	Smith

Query7: What is the total price of all configurations that have 'Leather' seats and a 'Leather' steering wheel?

```
385 •   SELECT SUM(total_price)
386      FROM `Order`
387      JOIN Configuration ON `Order`.config_id = Configuration.config_id
388      JOIN Seats ON Configuration.seat_id = Seats.seat_id
389      JOIN Steering_Wheel ON Configuration.wheel_id = Steering_Wheel.wheel_id
390      WHERE Seats.material = 'Leather' AND Steering_Wheel.material = 'Leather';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

first_name	last_name
Alice	Smith

Query8: What is the total price of the orders placed in the year 2024?

```
393 •   SELECT SUM(total_price)
394      FROM `Order`
395      WHERE YEAR(order_date) = 2024;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

SUM(total_price)
1364000.00

Query 9: How many orders have been placed for each type of seat material?

```
398 •    SELECT Seats.material, COUNT(`Order`.order_id)
399      FROM Seats
400      JOIN Configuration ON Seats.seat_id = Configuration.seat_id
401      JOIN `Order` ON Configuration.config_id = `Order`.config_id
402      GROUP BY Seats.material;
403
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

material	COUNT(`Order`.order_id)
Leather	1
Alcantara	1
Carbon Fiber	1
Fabric	1
Synthetic Leather	1
Velour	1
Vinyl	1
Wool	1
Microfiber	1
Suede	1
Nappa Leather	1
Perforated Leather	1
Pigmented Leather	1
Semi-Aniline Leat...	1
Top-Grain Leather	1
Full-Grain Leather	1
Corrected-Grain ...	1
Split Leather	1
Bonded Leather	1
Faux Leather	1

Query10: Which vehicle color has been chosen the most in the configurations?

```
405 •    SELECT Vehicle_Colour.Colour_name, COUNT(Configuration.config_id)
406      FROM Vehicle_Colour
407      JOIN Configuration ON Vehicle_Colour.colour_id = Configuration.colour_id
408      GROUP BY Vehicle_Colour.Colour_name
409      ORDER BY COUNT(Configuration.config_id) DESC
410      LIMIT 1;
411
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

Colour_name	COUNT(Configuration.config_id)
Black	1

Query11: What is the total price of all orders placed by a customer named 'John Doe'?

```
413 •    SELECT SUM(total_price)
414      FROM `Order`
415      JOIN Customer ON `Order`.customer_id = Customer.customer_id
416      WHERE Customer.first_name = 'John' AND Customer.last_name = 'Doe';
417
418
419
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

SUM(total_price)
39000.00

Query12: Which customers have ordered a configuration with the 'Alcantara' seat material?

```
421 •   SELECT Customer.first_name, Customer.last_name
422     FROM Customer
423     JOIN `Order` ON Customer.customer_id = `Order`.customer_id
424     JOIN Configuration ON `Order`.config_id = Configuration.config_id
425     JOIN Seats ON Configuration.seat_id = Seats.seat_id
426     WHERE Seats.material = 'Alcantara';
427
```

first_name	last_name
Jane	Doe

Query13: What are the different styles of rims and how many configurations use each style?

```
429 •   SELECT Rims.style, COUNT(Configuration.config_id)
430     FROM Rims
431     JOIN Configuration ON Rims.rim_id = Configuration.rim_id
432     GROUP BY Rims.style;
433
434
435
```

style	COUNT(Configuration.config_id)
Sport	1
Track	1
Forged	1
Alloy	1
Steel	1
Chrome	1
Wire	1
Spoke	1
Multi-Spoke	1
Mesh	1
Fan Blade	1
Turbine	1
Rally	1
Off-Road	1
Performance	1
Luxury	1
Vintage	1
Classic	1
Modern	1
Custom	1

Query14: How many orders have been placed for each type of steering wheel material?

```
437 •  SELECT Steering_Wheel.material, COUNT("Order".order_id)
438    FROM Steering_Wheel
439    JOIN Configuration ON Steering_Wheel.wheel_id = Configuration.wheel_id
440    JOIN `Order` ON Configuration.config_id = `Order`.config_id
441    GROUP BY Steering_Wheel.material;
442
443
```

material	COUNT("Order".order_id)
Leather	1
Alcantara	1
Carbon Fiber	1
Wood	1
Plastic	1
Metal	1
Rubber	1
Synthetic	1
Vinyl	1
Cloth	1
Polyurethane	1
Silicone	1
Gel	1
Neoprene	1
Microfiber	1
Suede	1
Velvet	1
Cotton	1
Linen	1
Hemp	1

Query15: What is the most expensive order placed?

```
445 •  SELECT MAX(total_price)
446    FROM `Order`;
447
```

MAX(total_price)
132000.00

Query16: How many different vehicle models have been ordered?

```
449 •  SELECT COUNT(DISTINCT model_id)
450    FROM Configuration
451    JOIN `Order` ON Configuration.config_id = `Order`.config_id;
```

COUNT(DISTINCT model_id)
20

Query17: Which customer has placed the most orders?

```
454 •    SELECT Customer.first_name, Customer.last_name
455      FROM Customer
456   ⊖ WHERE customer_id = (
457          SELECT customer_id
458          FROM `Order`
459          GROUP BY customer_id
460          ORDER BY COUNT(order_id) DESC
461          LIMIT 1
462      );
463
```

first_name	last_name
John	Doe

Query18: What is the average total price of orders for each vehicle model?

```
466 •    SELECT Vehicle_Model.model_name, AVG(`Order`.total_price)
467      FROM Vehicle_Model
468      JOIN Configuration ON Vehicle_Model.model_id = Configuration.model_id
469      JOIN `Order` ON Configuration.config_id = `Order`.config_id
470      GROUP BY Vehicle_Model.model_name;
471
472
```

model_name	AVG('Order'.total_price)
A-Class	35000.000000
B-Class	37000.000000
C-Class	43000.000000
CLA	39000.000000
CLS	72000.000000
E-Class	56000.000000
G-Class	132000.000000
GLA	38000.000000
GLB	40000.000000
GLC	44000.000000
GLE	56000.000000
GLS	78000.000000
S-Class	96000.000000
SL	93000.000000
SLC	51000.000000
AMG GT	117000.000000
AMG GT 4-…	91000.000000
EQC	69000.000000
EQS	104000.000000
EQV	73000.000000

Query19: Which customer has spent the most on their orders?

```
473 •    SELECT Customer.first_name, Customer.last_name
474      FROM Customer
475      JOIN `Order` ON Customer.customer_id = `Order`.customer_id
476      GROUP BY Customer.customer_id
477      ORDER BY SUM(`Order`.total_price) DESC
478      LIMIT 1;
```

first_name	last_name
Eve	Jones

Query 20: How many orders have been placed for each type of vehicle color?

```
481 •    SELECT Vehicle_Colour.Colour_name, COUNT(`Order`.order_id)
482      FROM Vehicle_Colour
483      JOIN Configuration ON Vehicle_Colour.colour_id = Configuration.colour_id
484      JOIN `Order` ON Configuration.config_id = `Order`.config_id
485      GROUP BY Vehicle_Colour.Colour_name;
486
```

Colour_name	COUNT(`Order`.order_id)
Black	1
White	1
Red	1
Blue	1
Green	1
Yellow	1
Purple	1
Orange	1
Pink	1
Brown	1
Gray	1
Silver	1
Gold	1
Beige	1
Ivory	1
Teal	1
Maroon	1
Lime	1
Indigo	1
Violet	1

VII. Project demonstration

- ● Tools/software/ libraries used: CSS,HTML,MYSQL
- ● Screenshot and Description of the Demonstration of project (If GUI is made):
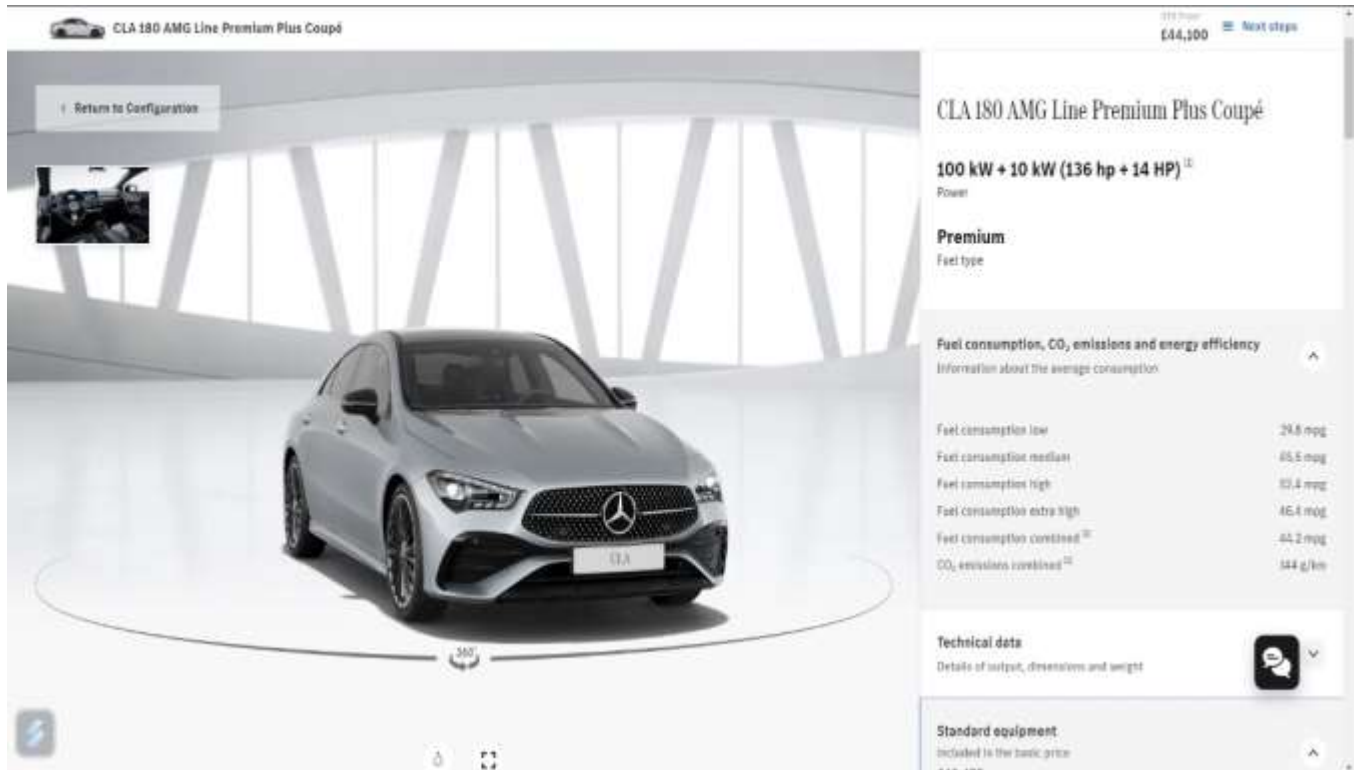
VIII. Self -Learning beyond classroom

This initiative was an attempt to learn outside of the classroom by self-study. It gave me the chance to learn more about the useful applications of SQL and database administration, which are not usually taught in a classroom. During this project, self-taught skills such as building a database schema, writing sophisticated SQL queries, and troubleshooting were learned through practical experience. Additionally, the project provided an opportunity to learn about the practical uses of databases in large-scale data management and organization, such as in auto dealerships.

IX. Learning from the Project

Through the initiative, theoretical knowledge was put into practice. Understanding databases, data structures, and SQL's role in data manipulation were all made easier by it. The initiative also shed light on the practical applications of databases, such as running an automobile dealership. Observing the relationships between various tables, the preservation of data integrity, and the performance optimization of queries was educational.

X. Challenges Faced

There were difficulties with the project. Among the difficulties encountered were knowing how several tables relate to one another, making sure that data was accurate, and performance-enhancing query optimization. Nevertheless, these difficulties were insightful learning opportunities that shed light on the complexities of database administration. Critical thinking, problem-solving abilities, and in-depth subject-matter knowledge were all necessary for each task.

XI. Conclusion

To sum up, this endeavor was an important learning experience that combined self-taught abilities with real-world application of classroom knowledge. It emphasized how crucial self-learning is to advancement both personally and professionally. The obstacles encountered throughout the project served as teaching opportunities that enhanced knowledge of SQL and database administration. The initiative demonstrated how important practical, hands-on experience is for increasing learning and supporting theoretical knowledge.