

**OMIS 452 Project: Kishen's Cars**

John Costuna (Z1922141) and Muhammad Siddiqui (Z1922019)

Northern Illinois University

OMIS 452 - Database Management for Business

Kishen Iyengar

May 1, 2024

## **Project Deliverable 1**

### **Kishen's Cars**

Kishen's Cars is a national car dealership founded in 2021 and based in Long Beach, California. Kishen's Cars partners with multiple manufacturers to provide various makes and models to their customers. Kishen and his company pride themselves on their quality customer service and their honest, down-to-earth selling approach. The company has also recently expanded from originally only selling cars to also offering mechanic services for its customers. The newly hired mechanics are the best in the nation, as they know how to repair multiple different car types without issue.

Kishen's Cars has 20 locations throughout the United States, from California to New York. Kishen also proudly employs 200 employees. Each dealership houses 10 employees: 1 manager, 4 salesmen, 4 mechanics, and 1 mechanic team lead. Kishen's Cars utilizes 1 payroll table, with specifications for each employee based on their ID. This helps account for the separate salaries and bonuses paid to employees, which allows for more accurate revenue tracking.

Each location also provides 20 cars, which are ordered by the manager of the location. Every car is identified through its Vehicle Identification Number (VIN), alongside the Customer ID, Car Purchase ID, make, model, color, and dealership location. Customers are also free to speak to the manager to request certain models to be ordered, which the manager is always happy to fulfill! Car payments can be paid in full or through payment plans. Payment plans are tracked with a table and each separate payment is given a unique ID alongside the date the payment was made. This helps Kishen and his company keep track of both outstanding payments and monthly revenue.

Customers of Kishen's Cars are also able to give optional reviews of their vehicle after purchasing. The customer would list their car's VIN, their customer name and ID, and give a score out of 5. They can also leave further descriptions about the car, which is also optional

The last feature of Kishen's Cars is its highly regarded mechanic service. The service was introduced last year and has been a big driving force in the company's sales. Customers are given free maintenance for the car's lifetime, not including body repairs. Repairs included are oil changes, battery recharges, battery replacements, and headlights. For a repair to be completed, customers must first make an appointment with the dealership. Once the appointment is made, a mechanic is assigned to the car. From there, the mechanic will let the Team Lead purchase the necessary parts to work on the vehicle.

Kishen's Cars needs to store a lot of information such as the various payrolls, purchases, employees, customers, and repairs to not only keep the company running efficiently but also so the company can provide accurate financial statements to check on their growth, as well as to give out to budding investors. This is why the company is moving to update its database system to maintain its market position.

**Entities List** (\*: primary key):

**Payroll** (payroll\_id\*, sales\_id[null], mechanic\_id[null], mechanic\_teamlead[null], location\_manager\_id[null], fname, lname, salary, bonuses[null])

**Salesmen** (salesmen\_id\*, salesmen\_name, email, phone\_number, location\_id)

**Cars** (VIN\*, customer\_id, cpurchase\_id, car\_make, car\_model, car\_color, location\_id)

**Customers** (customer\_id\*, customer\_name, phone\_number, email)

**Car Purchases** (cpurchase\_id\*, salesmen\_id, customer\_id, location\_id, VIN, price)

**Manufacturer Purchases** (mpurchase\_id\*, lmanager\_id, VIN, price)

**Car Payments** (payment\_id\*, customer\_id, VIN, amount\_paid)

**Locations** (location\_id\*, state, city, zipcode)

**Mechanics** (mechanic\_id\*, mechanic\_name, phone\_number, email, location\_id)

**Appointments** (appointment\_id\*, mechanic\_id, customer\_id, VIN, date)

**Mechanic Team Lead** (mteamlead\_id\*, teamlead\_name, phone\_number, email, location\_id)

**Repair Details** (repairdetail\_id\*, appointment\_id, repair\_name, repair\_notes)

**Location Manager** (lmanager\_id\*, lmanager\_name, phone\_number, email, location\_id)

**Car Part Purchases** (ppurchases\_id\*, mteamlead\_id, part\_name, part\_price)

**Car Review** (review\_id\*, VIN\*, customer\_id, car\_name, score\_out\_of\_five, review\_desc)

### **Assumptions**

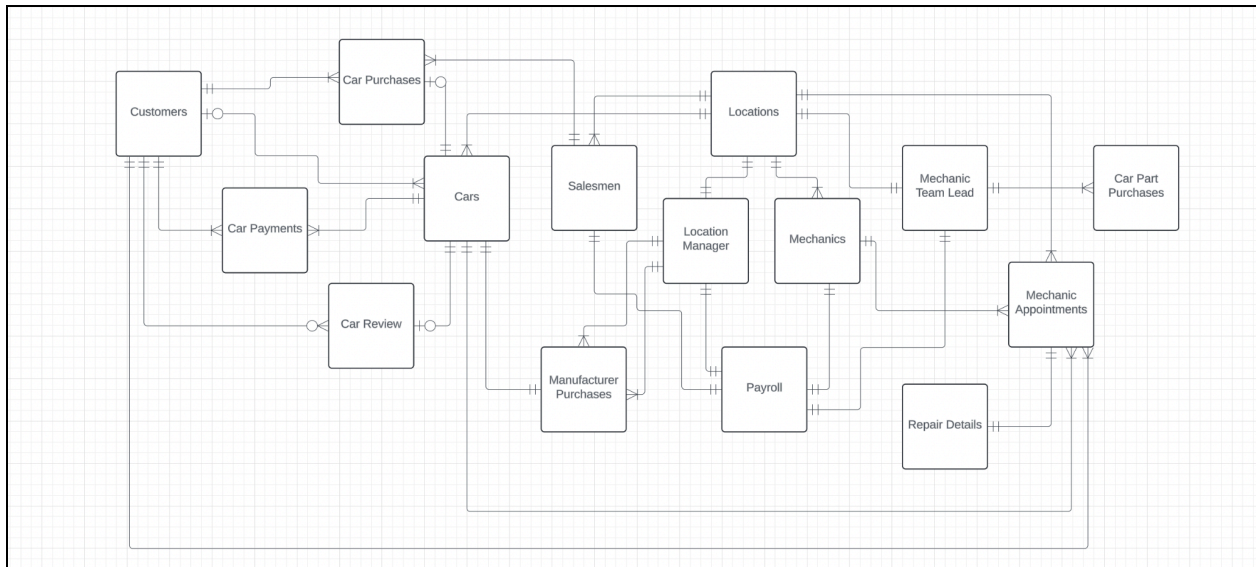
To help further explain our fictitious company, we will list our assumptions below. This will help readers understand our company's context further, and bring to light any questions that may be raised.

1. We assume that the mechanic team lead will buy the parts, as due to our work experience it is not the overall manager of the dealership who handles repairs and purchasing parts, it is someone who works in the repair unit, usually the team lead.
2. We assume that reviews are optional for all cars, and are put on the specific VIN
3. We assume that all employees work at one location.
4. We assume that every employee has one payroll, that can be updated if needed
5. We assume that to be a customer you must own a car, but that each car does not need to be owned by a customer.
6. We assume that location managers are in charge of purchasing from suppliers.
7. We assume that the payroll is taken care of by a CFO, but that is not included in the diagram. We did not want location managers to take care of their own payrolls as it can lead to falsified financial statements.

## Project Deliverable 2

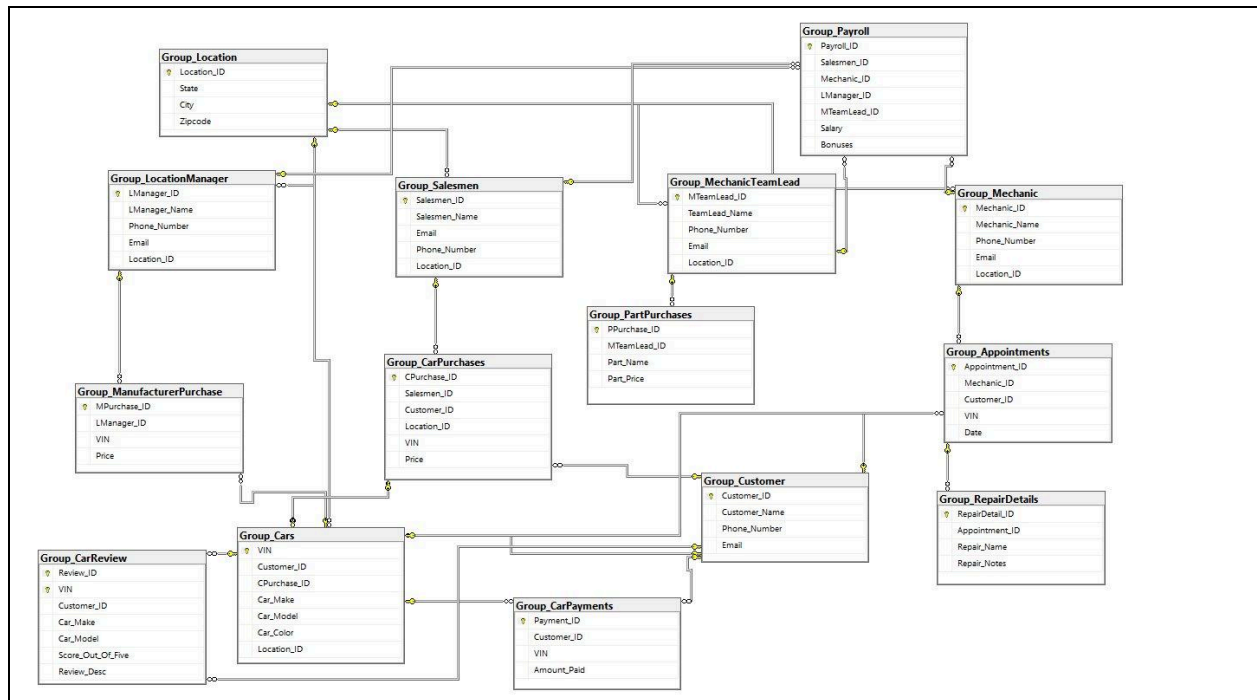
### ER-Diagram

Listed below is the ER Diagram for Kishen's Cars, with all the entities and their connections to each other.



## Project Deliverable 3

### Database Diagram + Queries



**(Query responses will be in SQL Database)**

Query 1: Return mechanic names who worked on the car with the VIN "1J8GR48K38C188128"

```
SELECT Mechanic_Name FROM Group_Mechanic WHERE Mechanic_ID IN (SELECT
Mechanic_ID FROM Group_Appointments WHERE VIN = '1J8GR48K38C188128')
```

Query 2: Display Cars Not Sold

```
SELECT VIN, Car_Make, Car_Model FROM Group_Cars WHERE VIN NOT IN (SELECT
VIN FROM Group_CarPurchases)
```

Query 3: Display Managers making over 5000 dollars

```
SELECT LManager_Name FROM Group_LocationManager WHERE LManager_ID IN
(SELECT LManager_ID FROM Group_Payroll WHERE Salary > 5000)
```

Query 4: Display income details for Salesmen

```
SELECT s.Salesmen_Name, p.Salary, p.Bonuses, (p.Salary + p.Bonuses) AS Full_Income
FROM Group_Payroll p, Group_Salesmen s WHERE s.Salesmen_ID = p.Salesmen_ID
```

Query 5: Displays all cars with a score greater than 3.

```
SELECT c.VIN, c.Car_Make, c.Car_Model, r.Score_Out_Of_Five FROM Group_Cars c,
Group_CarReview r WHERE c.VIN = r.VIN AND r.Score_Out_Of_Five > 3
```

Query 6: Display the State and Zipcode of the Dealership that Anthony Edwards works in.

```
SELECT s.Salesmen_Name, l.State, l.Zipcode FROM Group_Salesmen s, Group_Location l
WHERE s.Location_ID = l.Location_ID AND Salesmen_Name = 'Anthony Edwards'
```

Query 7: Display all mechanic repairs that were tire changes

```
SELECT a.VIN, r.Repair_Name FROM Group_Appointments a, Group_RepairDetails r
WHERE a.Appointment_ID = r.Appointment_ID AND r.Repair_Name = 'Tire Repair'
```

Query 8: Display the amount each manager has spent

```
SELECT l.LManager_Name, SUM(p.Price) AS TotalPurchases FROM Group_LocationManager
l, Group_ManufacturerPurchase p WHERE l.LManager_ID = p.LManager_ID GROUP BY
l.LManager_Name
```

Query 9: Display manufacturer purchases where the car costs more than 2000 dollars

```
SELECT m.LManager_ID, p.VIN, p.Price FROM Group_LocationManager m,
Group_ManufacturerPurchase p WHERE m.LManager_ID = p.LManager_ID AND p.Price >
2000
```

Query 10: Display mechanics having more than one appointment

```
SELECT m.Mechanic_ID, m.Mechanic_Name, COUNT(a.Appointment_ID) AS NumAppts
FROM Group_Mechanic m, Group_Appointments a WHERE m.Mechanic_ID = a.Mechanic_ID
GROUP BY m.Mechanic_ID, m.Mechanic_Name HAVING COUNT(a.Appointment_ID) > 1
```

Query 11: Display all customers owning only 1 car

```
SELECT c.Customer_Name, COUNT(p.VIN) AS NumPayments FROM Group_Customer c,
Group_CarPurchases p WHERE c.Customer_ID = p.Customer_ID GROUP BY
c.Customer_Name HAVING COUNT(p.VIN) = 1
```

Query 12: Display all customers who left a review description, and their respective review

description

```
SELECT c.Customer_Name, r.Review_Desc FROM Group_Customer c, Group_CarReview r
WHERE c.Customer_ID = r.Customer_ID AND r.Review_Desc IS NOT NULL
```

Query 13: Display the number of payments from each customer made

```
SELECT c.Customer_Name, COUNT(p.VIN) AS NumPayments FROM Group_Customer c,
Group_CarPayments p WHERE c.Customer_ID = p.Customer_ID GROUP BY
c.Customer_Name
```

Query 14: Display all customers who left a review

```
SELECT Customer_Name FROM Group_Customer WHERE Customer_ID IN (SELECT
Customer_ID FROM Group_CarReview)
```

Query 15: Display all team leads who have purchased parts

```
SELECT TeamLead_Name FROM Group_MechanicTeamLead WHERE MTeamLead_ID IN
(SELECT MTeamLead_ID FROM Group_PartPurchases)
```

Query 16: Display all mechanics working in Chicago

```
SELECT m.Mechanic_Name FROM Group_Mechanic m, Group_Location l WHERE
m.Location_ID = l.Location_ID AND l.City = 'Chicago'
```

Query 17: Display all manufacturer purchases between 1,000 and 10,000 dollars

```
SELECT p.VIN, Car_Make, Car_Model ,p.Price FROM Group_ManufacturerPurchase p,
Group_Cars c WHERE c.VIN = p.VIN AND p.Price BETWEEN 1000 AND 10000
```

Query 18: Display all cars in Illinois

```
SELECT c.VIN, c.Car_Make, c.Car_Model FROM Group_Cars c, Group_Location l WHERE
c.Location_ID = l.Location_ID AND l.State = 'Illinois'
```

Query 19: Show all purchased cars from Dekalb

```
SELECT c.VIN, c.Car_Make, c.Car_Model FROM Group_Cars c, Group_Location l,
Group_CarPurchases p WHERE p.VIN = c.VIN AND c.Location_ID = l.Location_ID AND
l.City = 'Dekalb' AND p.Customer_ID IS NOT NULL
```

Query 20: Display number of mechanics at each city

```
SELECT l.City, COUNT(m.Mechanic_ID) AS Num_Mechanics FROM Group_Location l,
Group_Mechanic m WHERE l.Location_ID = m.Location_ID GROUP BY l.City
```