

**Group Members:** Johnson Absolu, Koffi Gnamien Aristide, Ramon N. Nguema

**Class:** Database Development I

**Professor:** Frank Ravanshad

**Project:** Car Dealership Database Phase 2

**Date:** Tuesday, November, 8th, 2022

## Relational Schema

For this phase II, we converted the E-R Diagram of the exclusive Wolkswagen car dealership into a relational model. For that, we worked with every entity inside the er diagram and put through all their possible forms (1NF, 2NF, and 3NF) in order to get well behaved tables. Overall, we've been able to normalize every table and we ended with a pretty straightforward output.

To accomplish our goal, we went through a process of elimination of all multivalued attributes first, then we removed any partial dependencies within each table, and finally we took all transitive dependencies so we can keep moving forward with a clean relational model.

## Original Tables

Inventory	<u><b>Inventory ID</b></u> , Inventory total
Order	<u><b>Order ID</b></u> , Item, (order_date), (shipped_date), (delivery_date), order_total, payment_method
Car	<u><b>Car ID</b></u> , Body_type, Year, Model, (Trim), {color}, Price, [Total], [Status]
Employee	<u><b>Employee ID</b></u> , (Employee_name), Emp_address, Title

Customer	<b>Customer ID</b> , (Customer_name), Customer_address), Phone_number, Email_address, Order_id
----------	--

### Sub-Entities

ISA	Sales_number, Commission
Driver	License_ID, License_class, Truck_num, Delivery_number

**Inventory Entity** is an entity with an ID of inventory\_ID and a simple attribute of inventory\_total, that keeps track of the number of vehicles.

**inventory\_ID** is a unique identifier that identifies the inventory.

**Inventory\_total** is a simple attribute that tracks the total number of inventory.

**Car Entity** is an entity with an ID of Car\_ID, it has nine attributes. Its attributes are, Car\_ID, Body\_type, Year, Model, Trim, Color, Price, Total\_units and Status.

**Car\_ID** is a unique identifier that identifies a car.

**Body\_type** is a simple attribute that identifies the body type of the car.

**Year** is a simple attribute that identifies the age of the car.

**Trim** is a composite attribute that is composed of Engine\_size, Transmission and

### Drivetrain.

**Color** is a multi-value attribute, it allows a car to be more than one color.

**Price** is a simple attribute that defines the currency value of a car.

**Total Units** is a derived attribute. Its value is derived from Model and Trim.

**Status** is a derived attribute. Its value is derived off of

**Order Entity** is an entity with an id of Order\_id, it tracks customer orders and has seven attributes. The attributes are Order\_id, Item, Order\_date, Shipped\_date, Delivery\_date, Order\_total, and Payment\_total.

**Order\_id** is a unique identifier that identifies an order.

**Item** is a simple attribute that defines an object/entity that is included in an order.

**Order\_date** is a simple attribute that states the date an order is placed.

**Shipped\_date** is a simple attribute that states the data that an order is shipped.

**Delivery\_date** is a simple attribute that states the date an order is delivered.

**Order\_total** is a simple attribute that states the total amount of an order.

**Payment\_method** is a simple attribute that calculates the method used to place an order.

**Customer Entity** is an entity with an id of **Customer\_ID**. It identifies a customer who places an order, and has six attributes. Its attributes are **Customer\_ID**, **customer\_name**, **customer\_address**, **phone\_number**, **email\_address** and **order\_id**.

**Customer\_ID** is a unique identifier that identifies a customer.

**Customer\_name** is a composite attribute that captures a customer's name, it is composed of First Name and Last Name.

**Customer\_address** is a composite attribute that captures a customer's address. It is composed of a house number, Street, City, State and Zip Code.

**Phone\_number** is a simple attribute that captures a customer's phone number who places an order.

**Email\_Address** is a simple attribute that captures a customer's email address who places an order.

**Order\_ID** is a foreign key in the customer entity that matches a customer to an order.

**Employee** is an entity with an id of **Emp\_ID**, It is a super entity with total generalization of sub entities **ISA** and **Driver**. It has seven attributes, which are **Emp\_ID**, **Emp\_name**, **Emp\_address**, **Title**.

**Emp\_id** is a unique identifier that identifies an employee.

**Emp\_Name** is a composite attribute that captures an employee's name. It is composed of First name, and Last name.

**Emp Address** is a simple attribute that captures an employee's address.

**Title** is a simple attribute that states the title of an employee.

**ISA** is a Sub-entity of **Employee** with total specialization and disjoint constraint of the **Employee** entity. It has two attributes, **sales\_number** and **commission**.

**Sales number** is a simple attribute that connects an order with an ISA.

**Commission** is a simple attribute that states a commission amount from a sales.

**Driver** is a Sub-entity of Employee with total specialization and disjoint constraint of the employee entity. Its ID is License\_ID and It has 4 attributes, License\_id, truck\_num and delivery\_number.

**License ID** is a simple attribute that captures the license number.

**Truck Num** is a simple attribute that states a truck number, this number is assigned to a Driver.

**Delivery number** is a simple attribute that states a number for a delivery.

## ***Integrity constraints***

### **- Domain constraints**

- Later, we'll define the components of each table and their data such as domain name, meaning, data type, size (or length), and allowable values or allowable range (if applicable). For this job, we just focused on normalization.

### **- Entity Integrity**

- For entity integrity constraints, all primary keys will be set to NOT NULL since primary key values would be used to identify individual rows in a relation

### **- Referential Integrity**

- Through normalization, we made sure that every foreign key matches with the primary key. Also, we removed duplicate and unnecessary data through the process of decomposition.

## Normalized Tables

### *Candidate keys*

TABLE NAME	ATTRIBUTES
Inventory	<u>Inventory ID</u> , Inventory_total
Car	<u>Car ID</u> , Drive_train, Transmission, Price, Total_units, Status
Car Color	<u>Color ID</u> , <u>Car_ID</u> , Color_name
Car Model	<u>Car ID</u> , <u>Model</u> , Engine_size
Order	<u>Order ID</u> , Item, order_date, shipped_date, delivery_date, order_total, payment_method
Customer	<u>Customer Id</u> , First_name, Last_name, House_number, Street, Phone_number, Email_address, <u>Zip code</u>
Customer Zip Code	<u>Zip Code</u> , State, City, <u>Customer Id</u>
Employee	<u>Employee ID</u> , First_name, Last_name, Employee_address, Title
E Driver	<u>D Employee ID</u> , License_id, Truck_number, Delivery_number
E ISA	<u>I Employee ID</u> , Sales_number, Commission

### *Foreign Keys*

<u>Car_ID</u>	References the relationship between the Car entity(parent), Car Color, and Car Body Type
<u>Custome ID</u>	References the relationship between the Customer entity(parent) and Customer Zip Code
<u>Order_id</u>	References the relationship between the Customer entity(parent) and Order
<u>Car ID</u>	References the relationship between the Car entity(parent), Model the child

## DEPENDENCIES

### *Inventory relation*

- **Partial Functional Dependencies:** N/A
- **Full Functional Dependencies:**
  - **Inventory ID** => Inventory Total
- **Transitive dependencies:** N/A

### *Car relation*

- **Partial Functional Dependencies:** N/A
- **Full Functional Dependencies:**
  - **Car ID** => Drive Train, Transmission, Price, Total Units, Status
- **Transitive dependencies**
  - Body type => Model, Engine size

### *Color relation*

- **Partial Functional Dependencies:** N/A
- **Full Functional Dependencies:**
  - **Color ID**, **Car ID** => Color name
- **Transitive dependencies:** N/A

### *Model relation*

- **Partial Functional Dependencies:** N/A
- **Full Functional Dependencies:**
  - **Model**, **Car ID** => Engine Size
- **Transitive dependencies:** N/A

## *Order relation*

- **Partial Functional Dependencies:** N/A
- **Full Functional Dependencies:**
  - Order\_ID => Item, Order\_Date, Shipped\_date, Delivery\_date, Order\_total, Payment\_Method
- **Transitive dependencies:** N/A

## *Customer relation*

- **Partial Functional Dependencies:** N/A
- **Full Functional Dependencies:**
  - Customer\_id => Full Name, Phone number, Email address, Street, House Number, Order ID
- **Transitive dependencies:**
  - Zip Code => State, City

## *Customer Zip Relation*

- **Partial Functional Dependencies:** N/A
- **Full Functional Dependencies:**
  - Zip Code, Customer\_id => State, City
- **Transitive dependencies:** N/A

## *Employee relation*

- **Partial Functional Dependencies:** N/A
- **Full Functional Dependencies:**
  - Emp\_ID => FirstName, LastName, EmpAddress, Title
- **Transitive dependencies:** N/A

## ***Driver relation***

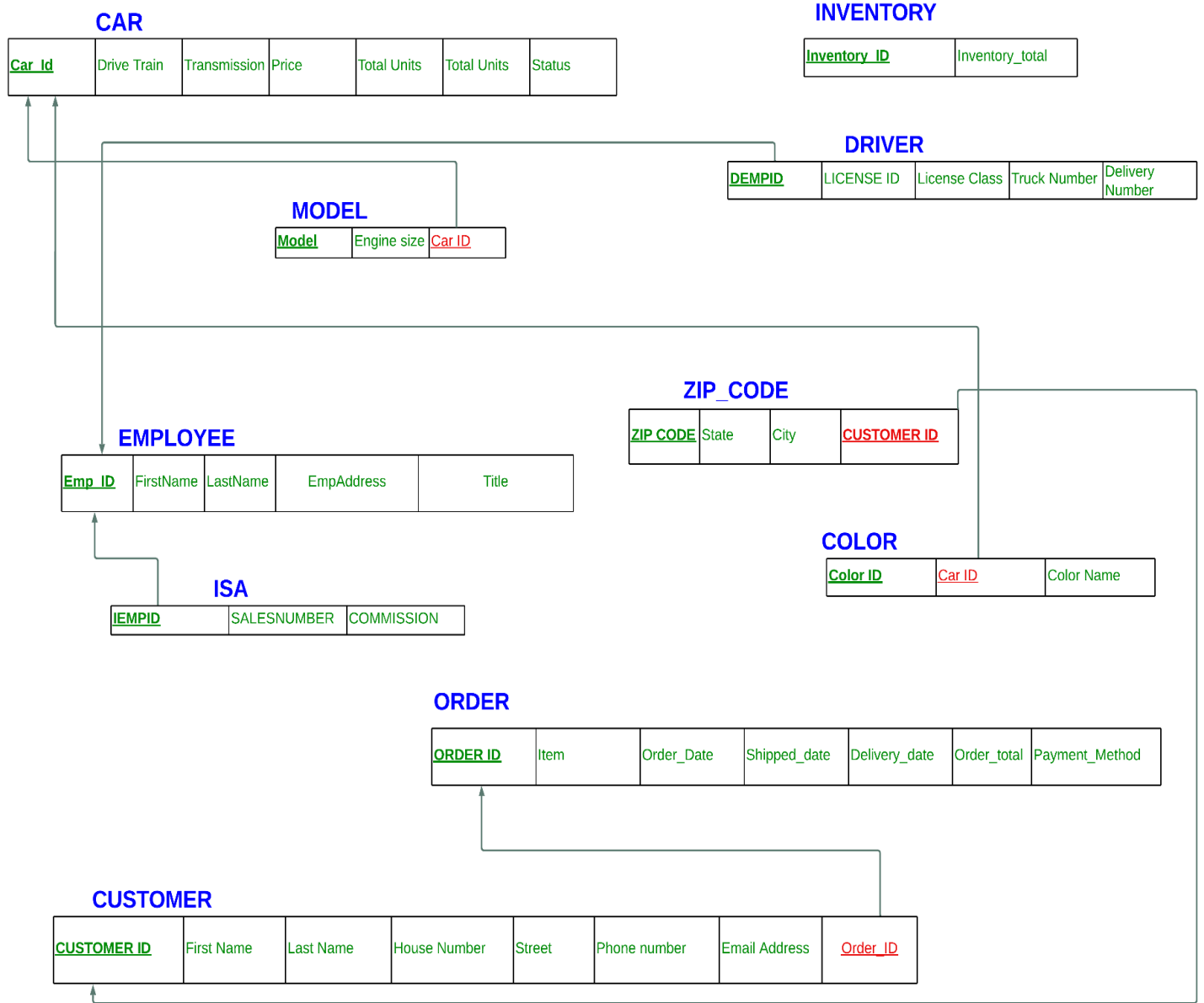
- Partial Functional Dependencies: N/A
- Full Functional Dependencies:
  - **Dempid** => License\_ID, License Class, Truck Number, Delivery Number
- **Transitive dependencies:** N/A

## ***ISA relation***

- **Partial Functional Dependencies:** N/A
- **Full Functional Dependencies:**
  - **IEMPID** => Sales Number, Commission
- **Transitive dependencies:** N/A

**\*\* The red color represents foreign keys\*\***



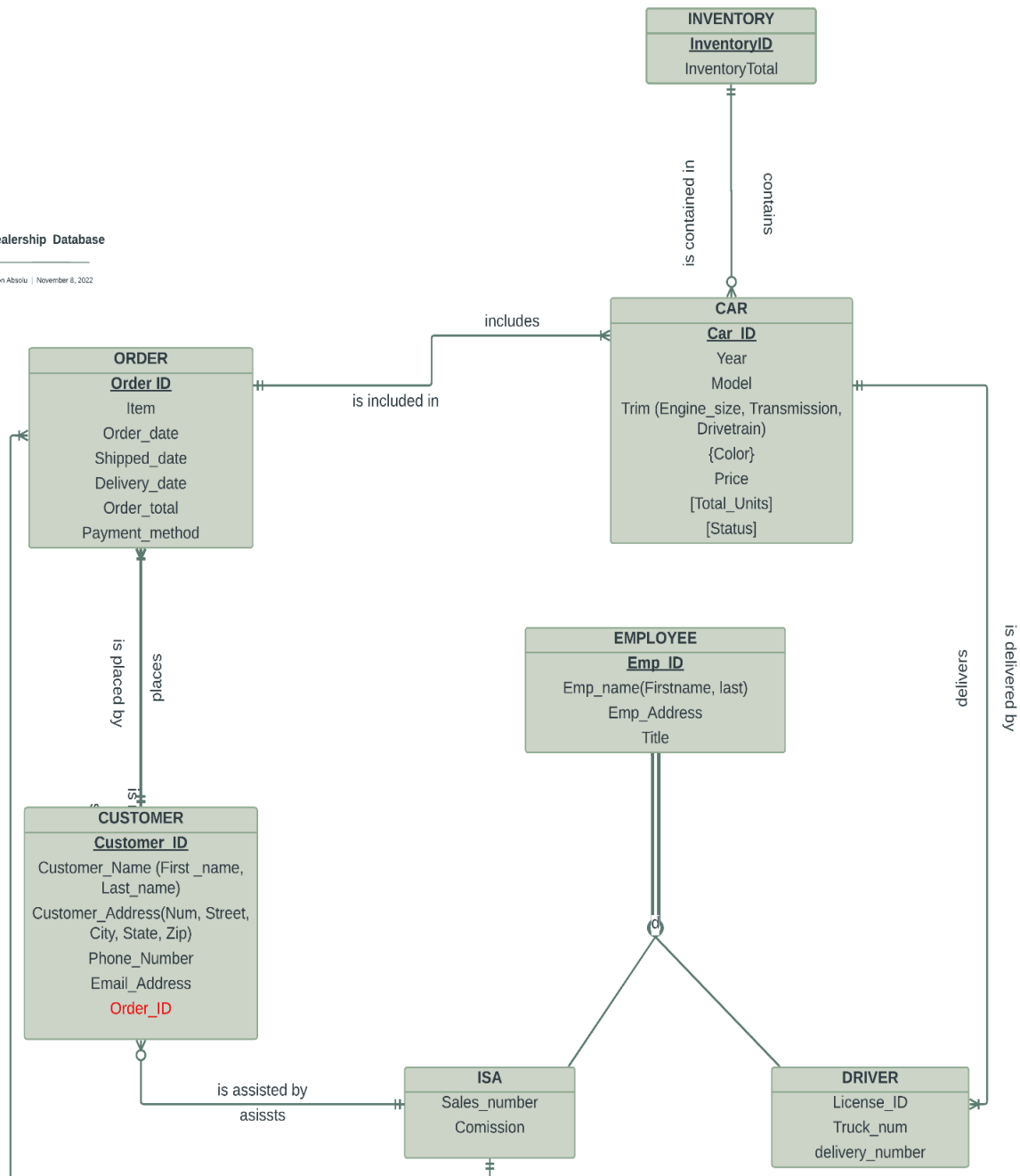
**Diagram Key (Beta)**

● FOREIGN KEY

● ENTITY NAME

## Car Dealership Database

Johnson Absolu | November 8, 2022



## Diagram Key (Beta)

● FOREIGN KEY