CIS 353 DataBase Project:

Car Dealership Project

Section 01 Fall 2018

Cole Sellers

Hai Duong

Eli Anderson

Lauren DeFrancesco

Table of Contents

**Summary:**

**Big picture:**

Our project will be based on a car dealership company. This particular company may have several dealership locations, each with differing vehicle types, customers, and employees. The point of this project is to allow for a way to track which customers are buying which cars so that the dealership can see which cars are selling. It also creates a way for the customer to keep a record of which car they bought in case anything came up needing the proof of purchase in the future. It also creates a way for employees to track which one is selling the most cars because in every company a little competition is always healthy. And for vehicle type, it will show which cars are selling so the company knows which cars they should have more in a stock of.

**Entities:**

**Model:**

A vehicle model consists of a string model name and a model year. Since a specific model can be made multiple years, the primary key of the table is the name.

**Vehicle:**

A vehicle is from a dealership and each vehicle is identified by its VIN number, this will be its primary key. All vehicles will be of the same make, but these will be split between potential dozens of different models. Each vehicle will have a specific cost which could be different from any other vehicle as well as a color which also could be different from any other vehicle.

**Dealership:**

A dealership is a place where it has vehicles to be bought and sold. In a given dealership business there could be anywhere from 1 to many dealership locations. Each dealership will be identified by its dID which will be its primary key. A dealership will also have a name which could be repeated throughout the other dealerships. A dealership also has an address. Finally, a dealership can have a phone number but dealerships can have multiple phone numbers, so this would be a multivalued attribute.

**Customer:**

A customer is who buys or sells a vehicle. This customer will come into a dealership to buy a car of a specific model. Each customer will be identified by their social security number which will act as a primary key. Each customer also has a name as well as a phone number. After the customer has chosen a vehicle they will have a payment amount and the bill/receipt will be sent to them via email so we will need an email.

**Employee:**

An employee, like a customer, will be identified by their social security number. Each employee will have a phone number in which they can be reached as well as an hourly rate. On top of an hourly rate because this is a car salesperson they will receive commision which is based person by person and is a percent value on how much an employee will be paid based on the car they sell. Each employee will assist many customers.

**Sales:**

Each vehicle sale will have a receipt that holds data of the employee, customer, and vehicle. The receipt will have its own ID number (saleID), the employee’s ID (eSSN), the customer’s ID (cSSN), the vehicle’s ID (VIN), and the date.

**Relational schema:**

employee(ssn, phoneNum, commish, rate, dID)

customer(ssn, phoneNum, name, email, eSSn, contactDate)

vehicle(VIN, cost, color, dID, mNum)

model(mNum, Year)

dealership(dID, name, address)

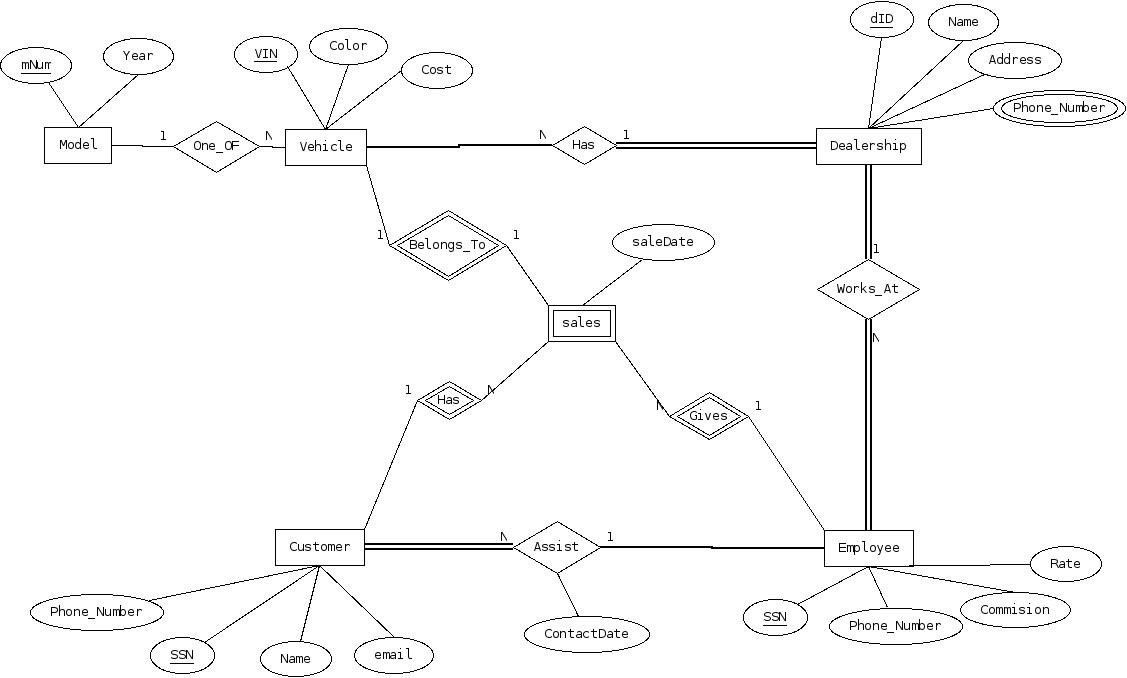
sale( SSN, eSSN, VIN, saleDate)

dealerPhoneNum(phoneNum, dId)

**Integrity constraint:**

|  |  |  |
| --- | --- | --- |
| IC names and table | IC type | English statement |
| dId in Dealership | Key | A dealership will be identified by it dealership ID number. |
| dId in employee and dId in dealership | Foreign key | An employee can work for only a single dealership, but a dealership can have multiple employees working for it. |
| cost in vehicle | 1-attribute | A vehicles price cannot be below $0 |
| Employee table using attributes commision and rate. | 2-attribute, 1-row | If an employee is making more than $20.00/hour their commision cannot be greater than 10% |

**ER Diagram:**

****