

# Inventory Database Management System

**CSC 675-03**

**Name: Ahmad Afghan**

**Student ID: 918786406**

**Github: AhmadCr7**

**“Milestone 2”**

**November 3, 2020**

| <b>Milestone</b> | <b>Due Date</b>         | <b>Version</b> |
|------------------|-------------------------|----------------|
| <b>1</b>         | <b>October 6, 2020</b>  | <b>V1</b>      |
| <b>2</b>         | <b>November 3, 2020</b> | <b>V1</b>      |

# Table of Contents

**Section I: Project Description**

**Section II: Use Cases**

**Section III: Database Requirements (Business Rules)**

**Section IV: Main Entities, Attributes and Keys**

**Section V: Entity Relationship Diagram (ERD)**

**Section VI: Testing Table**

**Section VII: Database Model/EER**

**Section VIII: Forward Engineering**

**Section IX: Inserting Data**

**Section X: Testing**

**Section XI: Testing Table**

---

## Section I: Project Description

Having a successful business always requires good inventory management. To maximize the company benefits having detailed inventory reports helps you spot trends, maintain adequate supplies and spot potential problems. Maintaining a good stock should be part of regular planning tools for a company. An inventory database provides a wide variety of information about the inventory that goes beyond what it is. As a part of this class we are creating a car dealership inventory database management system that provides accurate information regarding cars in stock, cost of each car, incoming cars, and sold cars. In this database employees and buyers will be able to check the specific car information and if it's available for sale. Every employee will have access to the inventory and information regarding car make, models, years made, and engine type. Old buyers will be able to bring their car for service.

## Section II: Use Cases

|                 |   |
|-----------------|---|
| Use Case Title: | <b>Desperate Buyer</b>  |
| Actors:         | Customer, Car, Employee, Dealership, Car Type,  |
| Description:    | Edgar Villanueva had a very only honda civic and it recently broke down while he was on his way to work. He drives 50 miles each way to work everyday. He needs a car as soon as possible. Since Edgar had good experiences with his old Honda he is contacting X dealership to see if they have newer model of Honda Civic in stock, after few minutes talking on the phone with car dealership he founds out that the dealership is getting new Honda models this weeked, so he makes appointment to purchase his new vehicle this weeked. He's very excited! |

|                 |  |
|-----------------|--|
| Use Case Title: | <b>Young Boy is Getting New Car</b>  |
| Actors:         | Customer, Account, Price, Address, Credit card, Branch, Salesperson, Sold cars, Discount,  |
| Description:    | Abdul's youngest son Mohammad is turning 16 this month, he's in a market to buy a car for his son. While his son is very happy Abdul himself is not very happy since his budget for the car is very low and he does not have a good credit to finance a car. He contacted a dealership X branch and asked them regarding their available cars. After a quick conversation on the phone with the sales person Abdul shares his fear regarding his bad credit and not getting approved for the car. The salesperson assured him that they have some cars on discount and they have sold many cars in the last couple of days. The sales person prompted Abdul to make an online account through the dealership website and fill out a quick form regarding his credit and his personal information before coming to dealership next week so they can match his information with a budget price car and speed up the process. |

|                 |   |
|-----------------|---|
| Use Case Title: | <b>New Branch Manger</b>  |
| Actors:         | Car feature, Dealership, Branch Manager, Branch Location, Account, Account Type, Incoming Cars, Sold cars, Car Type, Car Year,  |
| Description:    | Bob is a branch manager at dealership X. It's been only two weeks since his move from branch Y to branch Z, so he wants to know about his new branch stock and see what cars are available to be sold. Since now he's working in a different location he needs to create a new account and log in as a new store manager. After logging to his account Bob finds out that they are sold out in a Honda Accord Sports model and he has to put in a new request to stock up some new Honda Accord 2020 Sport model. He ordered a few newer model of the car and he's looking forward to bring some good profits to the company. |

|                 |  |
|-----------------|--|
| Use Case Title: | <b>Old Customer Complaining About His Car</b>  |
| Actors:         | Customer, Car VIN, Dealership, Mechanic, Warranty, Coupon, Contact   |
| Description:    | Hector bought a Car three months ago from dealership X. He has only driven the car only a few miles and his check engine light came on and he's very pissed about that. He decides to contact the dealership and make them fix his car. When he contacts the dealership they ask him for the car vin number so they can look up the make and model of the vehicle in their database. The Mechanic guy tells him that his car is out of warranty and he has to pay out of pocket. However, since it's been only a few days that his warranty has expired they give him a coupon code to apply and they will fix his car for free. Now Hector is not pissed anymore since he's getting his car fixed for free. All he has to do is to create an account and log in to his account and apply the coupon code and show up to the dealership and they will fix his car. |

|                 |   |
|-----------------|---|
| Use Case Title: | <b>New Hired Salesperson</b>  |
| Actors:         | Branch Manager, Account, Account type, Available in stock, Salesperson, Car Type  |
| Description:    | Naweed is a new hire salesperson at dealership X. Since he's very new to the dealership and he wants to make a good reputation of himself by being on top of his work. He wants to educate himself about types of cars that his dealership has available for sale. He logs into his account to see the inventory and what kind of car has a good sales market, so he can prepare himself for customers. However, since he's new to the company the database does not allow him to see the inventory or just shows him few cars that anybody can see. He had to contact the manager and then manually give him permission to the dealership database and now he can access all the cars. |

### Section III: Database Requirements (Business Rules)

#### 1. Customer

- A customer shall be able create only one online account using unique email.
- A customer shall buy many cars.
- A registered user shall be able to apply at least one coupon.
- A customer shall have at least one car.
- A customer shall have a unique id.
- A customer shall have a name.
- A customer shall have one and only one phone number.
- A customer shall have one address.
- A customer shall have an email address.
- A customer shall receive zero or many discounts.
- A customer shall have many credit cards.

#### 2. Customer Address

- A customer address shall have a unique id.
- A customer address shall belong to many customers.
- A customer address shall have a street number.
- A customer address shall have a city.

- A customer address shall have a state.
- A customer address shall have a zip code.
- A customer address shall have country.

### 3. **Account**

- An account shall be created by one and only one customer.
- An account shall have at least one account type.
- An account shall have one unique id.
- An account shall belong to only one customer.

### 4. **Account Type**

- 
- An account type shall have one unique account id.
- An account shall have a description
- An account type shall belong to many accounts.

### 5. **Service**

- A service has a unique service id.
- A service has a description

### 6. **Car**

- A car shall be purchased at least by one customer.
- A car shall be owned by one dealership.
- A car shall have many types.
- A car shall have one and only one engine type.
- A car shall have a year.
- A car shall have make
- A car shall have model
- A car shall have a unique id
- A car shall have a VIN number.
- A car shall have engine type.
- A car shall have at least a color.
- A car shall have a price.

- A car shall have many discounts.
- A car shall come with many warranties.
- A car shall have one or many types.

## **7. Warranty**

- A warranty shall have a unique id.
- A warranty shall be applied to many cars.
- A warranty shall have starting dates.
- A warranty shall have ending dates.

## **8. Car Engine**

- A car engine shall have a unique id.
- A car engine shall have an engine type.

## **9. Car Type**

- A car type shall have a unique id.
- A car type shall belong to many cars.

## **10. Dealership**

- A dealership shall be managed by one manger.
- A dealership shall have many cars.
- A dealership shall have a name.
- A dealership shall have a unique id.
- A dealership shall have a phone number.
- A dealership shall be located for many customers.
- A dealership shall have many incoming cars.
- A dealership shall have at least employees working in it.
- A dealership shall have at least one address.
- A dealership shall have many sections.
- A dealership shall have many cars to be sold.
- A dealership shall have one and only one mechanic shop.
- A dealership shall have a unique address.
- A dealership shall be supplied by at least one supplier.
- A dealership shall offer zero or many financing plan



## **11. Finance Plan**

- A finance plan shall have a unique id
- A finance plan shall have a term.
- A finance plan shall have amount
- A finance plan shall be offered by many dealership

## **12. Dealership Address**

- A dealership shall have a unique id.
- A dealership shall have a street.
- A dealership shall have a city.
- A dealership shall have a state.
- A dealership shall have a country.
- A dealership shall have a zip code.

## **13. Employee**

- An employee shall have a unique employee id.
- An employee shall have many rules.
- An employee shall have a name.
- An employee shall have a last name.
- An employee shall have one and only one phone number.
- An employee shall have a birthday.
- An employee shall have a starting date.

## **14. Branch Manager**

- A branch manager shall supervise many salespersons .
- A branch manager shall have a salary.

## **15. Discount**

- A Discount shall have a unique id
- A discount shall be applied to many customers.
- A discount shall have starting period

- A discount shall have an ending period.

## **16. Dealership Supplies**

- A supplier shall have a unique id.
- A supplier shall supply many dealership
- A supplier shall have a name
- A supplier shall have a phone number

## **17. Credit card**

- A credit card shall have a unique number
- A credit card shall have a expiration date
- A credit card shall be owned by at least one customer
- A credit card shall have CVV
- A credit card shall have a type.
- A credit card shall have one and only one payment type.

## **18. Payment Type**

- A payment type shall have a unique number
- A payment type shall have a billing address
- A payment type shall belong to at least one credit card.

## **Section IV: Main Entities, Attributes and Keys**

### **1. Customer (strong)**

- A. Customer\_id: key, numeric
- B. name : multivalued, alphanumeric
- C. Dob : composite, timestamp
- D. last\_name: composite, alphanumeric
- E. email : key, alphanumeric
- F. phone\_number : composite, alphanumeric
- G. gender : multivalued, alphanumeric

### **2. Customer Address (strong)**

- A. address\_id : key, numeric
- B. street : multivalued, alphanumeric
- C. city : alphanumeric
- D. state : alphanumeric
- E. country : alphanumeric
- F. zip\_code : alphanumeric

### 3. Employee (weak)

- A. employee\_id : key, numeric
- B. name : multivalued, alphanumeric
- C. last\_name: multivalued, alphanumeric
- D. email : key, alphanumeric
- E. phone\_number : multivalued, alphanumeric
- F. gender : multivalued, alphanumeric
- G. Dob : multivalued, timestamp
- H. Employee\_starting\_date : multivalued, timestamp

### 4. Dealership (strong)

- A. dealership\_id : key, numeric
- B. dealership\_name: multivalued, alphanumeric
- C. phone\_number : multivalued, alphanumeric
- D. Email: multivalued, alphanumeric

### 5. Dealership Address (strong)

- A. address\_id : key, numeric
- B. street : multivalued, alphanumeric
- C. city : alphanumeric
- D. state : alphanumeric
- E. country : alphanumeric
- F. zip\_code : alphanumeric

### 6. Dealership Service shop (strong)

- A. Service\_id: key, numeric
- B. Dealership\_id: weak key, numeric
- C. phone\_number : multivalued, alphanumeric

**7. Car parts (strong)**

- A. Part\_id : key, numeric
- B. Car\_vin: weak, key, numeric
- C. Part\_Price : numeric

**8. Car (strong)**

- A. Car\_VIN : key, numeric
- B. Car\_make : multivalued, alphanumeric
- C. Car\_model : multivalued, alphanumeric
- D. car\_year : multivalued, timestamp
- E. car\_price : numeric
- F. Car\_color : multivalued, alphanumeric

**9. Car Type (strong)**

- A. Type\_id: key, numeric
- B. Car\_vin : weak key, numeric
- C. Truck: multivalued, alphanumeric
- D. sedan : multivalued, alphanumeric

**10. Car Engine (strong)**

- A. engine\_id: key, numeric
- B. Car\_id: weak key, numeric
- C. engine\_type: numeric, multivalued, alphanumeric

**11. Account (weak)**

- A. Account\_id: key, numeric
- B. customer\_email: weak key, numeric
- C. Type: weak key, numeric
- D. password : multivalued, alphanumeric

**12. Account Type (strong)**

- A. account\_typeId: key, numeric
- B. Description : alphanumeric

### **13. Discount (strong)**

- A. Discount\_id: key, numeric
- B. Car\_vin : weak key, numeric
- C. Custome\_idr: weak key, numeric
- D. Starting\_date: multivalue, timestamp
- E. Expiration\_date: multivalue, timestamp

### **14. Warranty (strong)**

- A. Warranty\_id: key, numeric
- B. Car\_vin: weak key numeric,
- C. Starting\_date: multivalue, timestamp
- D. Expiration\_date: multivalue, timestamp

### **15. Salesperson (weak)**

- A. Salesperson\_id: key, numeric
- B. Employee\_id: weak key, numeric
- C. Base\_salary: multivalue, numeric
- D. Starting\_date : multivalue, timestamp

### **16. Branch Manager (weak)**

- A. Manbger\_id: key, numeric
- B. Employee\_id: weak key, numeric
- C. Salary: multivalue, numeric
- D. Starting\_date : multivalue, timestamp

### **17. Mechanic (weak)**

- A. Mechanic\_id: key, numeric
- B. Employee\_id: weak key, numeric

- C. Salary: multivalue, numeric
- D. Starting\_date : multivalue, timestamp

**18. Payment Type (strong)**

- A. Type\_id: key, numeric
- B. Billing\_address: alphanumeric
- C. Billlig\_city: alphanumeric
- D. Billlig\_state: alphanumeric
- E. Billlig\_country: alphanumeric
- F. Billlig\_zip: alphanumeric, numeric

**19. Credit card (weak)**

- A. Card\_numer: key, numeric
- B. Bank\_code: alphanumeric
- C. CVV\_number: numeric
- D. Expiration data\_composite\_date
- E. Payment\_type: weak key, numeric

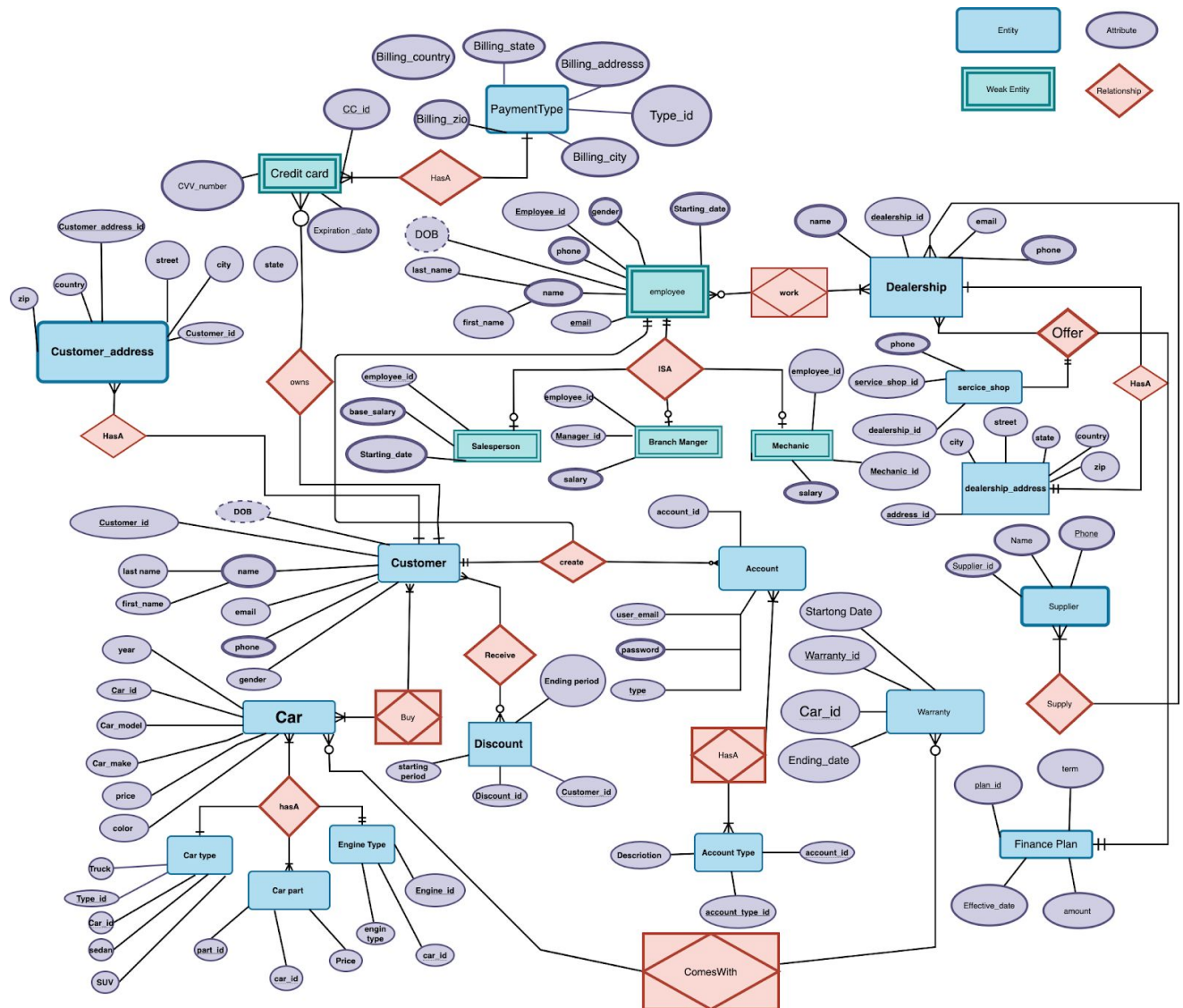
**20. Dealership Supplier (strong)**

- A. Supplier\_id: key, numeric
- B. Name: multivalue, alphanumeric
- C. Phone: alphanumeric

**21. Financing plan (strong)**

- A. Finance\_id: key, numeric
- B. Customer\_id: weak alphanumeric
- C. term : composite, data
- D. Effective\_date: multivalue, timestamp

## Section V: Entity Relationship Diagram (ERD) (Updated)



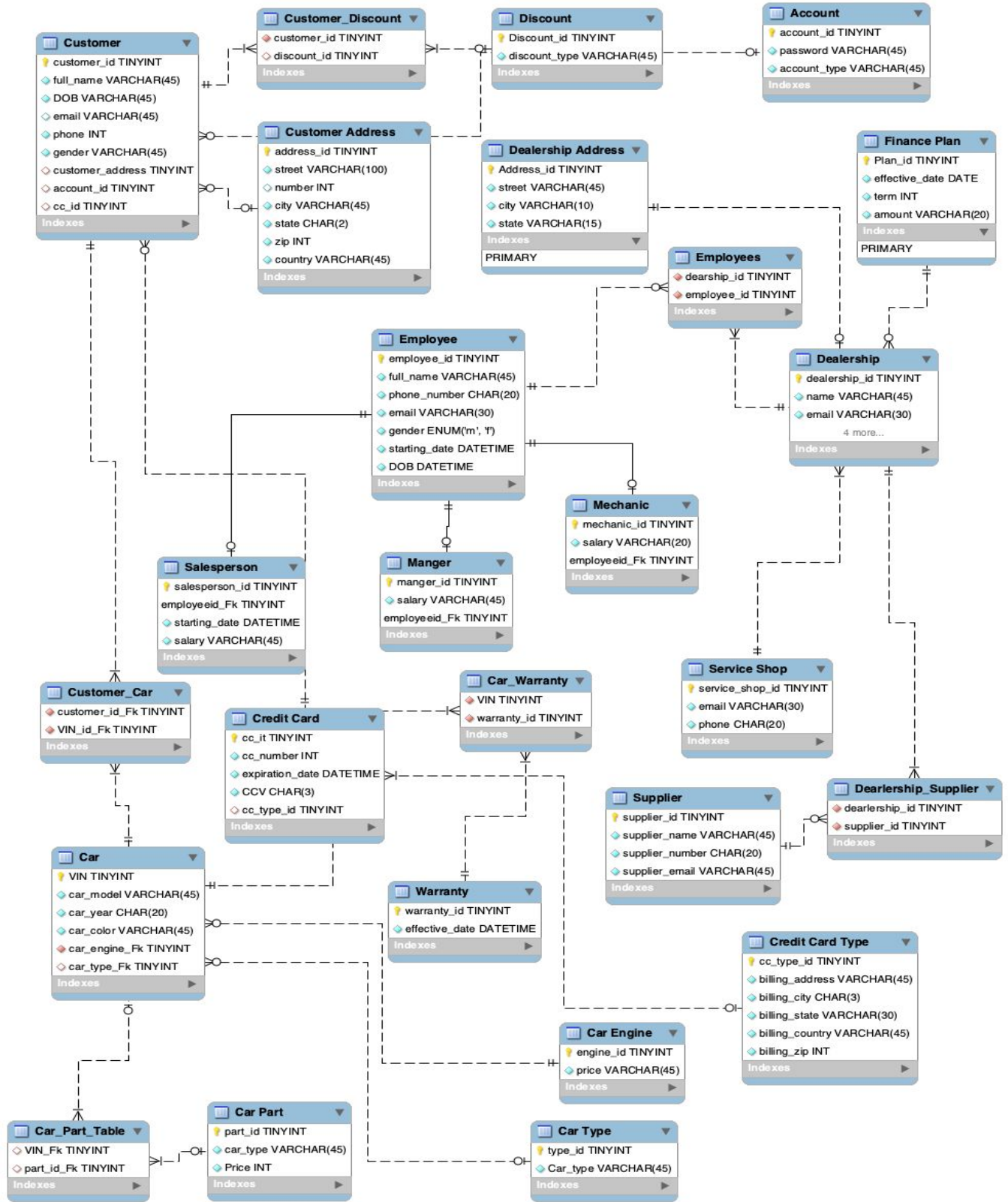
## Section VI: Testing Table

| Rule | Entity A   | Relation   | Entity B       | Cardinality | Pass/Fail | Error Description                              |
|------|------------|------------|----------------|-------------|-----------|--|
| 1    | Car        | HasA       | Car Type       | M to M      | Pass      |  |
| 2    | Car        | HasA       | Engine Type    | M to 1      | Pass      |  |
| 3    | Car        | HasA       | Car Parts      | 1 to 1      | Failed    | A car must have many parts                     |
| 4    | Customer   | Buy        | Car            | M to M      | pass      |  |
| 5    | Customer   | Receive    | Discount       | M to 1      | failed    | A customer shall receive many discount 1- M    |
| 6    | Employee   | ISA        | Salesperson    | 1 to 1      | pass      |  |
| 7    | Employee   | ISA        | Mechanic       | 1 to 1      | pass      |  |
| 8    | Employee   | ISA        | Branch Manager | 1 to 1      | pass      |  |
| 9    | Employee   | workds     | Dealership     | M to 1      | failed    | An employee can work in one or many dealership |
| 10   | Dealership | Offer      | Service shop   | 1 to 1      | pass      |  |
| 11   | Dealership | Offer      | Finance Plan   | 1 to M      | pass      |  |
| 12   | Dealership | supplyby   | Supplier       | 1 to M      | pass      |  |
| 13   | Customer   | Create     | Account        | 1 to 1      | pass      |  |
| 14   | Account    | HasA       | Account        | M to M      | failed    | A account can have one or many type            |
| 15   | Car        | Comes with | Warranty       | 1 to M      | pass      |  |
| 16   | Customer   | HasA       | address        | 1 to M      | failed    | A customer can have zero address               |
| 17   | Dealership | HasA       | address        | 1 to 1      | pass      |  |



| Rule | Entity A     | Relation | Entity B     | Cardinality | Pass/Fail | Error Description                               |
|------|--------------|----------|--------------|-------------|-----------|---|
| 18   | Customer     | owns     | Credit Card  | 1-M         | pass      |   |
| 19   | Supplier     | Supply   | Dealership   | 1-M         | Failed    | A dealership can be supplied by many suppliers  |
| 20   | Finance Plan | offer    | Dealership   | M-N         | Pass      |   |
| 21   | Credit card  | hasA     | Payment Type | 1-1         | pass      |   |
| 21   | Payment Type | HasA     | Credit card  | 1-1         | failed    | A payment type shall belong to many credit card |
|      |              |          |              |             |           |   |
|      |              |          |              |             |           |   |

## Section VII: Database Model/EER



### 3. itemized description table.

Note: the rest of the table are no actions only below entities are having actions as of now:

| Entity      | Action      | Reason                |
|-------------|-------------|-----------------------|
| Salesperson | On Cascade  | It's ISA relationship |
| Mechanic    | ON Cascade  | It's ISA relationship |
| Manger      | On Casecate | It's ISA relationship |

## Section VIII: Forward Engineering

### 5. databasemodel.sql

The screenshot displays the MySQL Workbench interface for a local instance (3306). The 'Schemas' tab is active, showing a script named 'databasemodel.sql'. The script content is as follows:

```
1  -- MySQL Script generated by MySQL Workbench
2  -- Thu Oct 29 12:41:10 2020
3  -- Model: New Model    Version: 1.0
4  -- MySQL Workbench Forward Engineering
5  -- Ahmad Afghan
6  -- CSC 675 SFSU FALL 2020
7
8  SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
9  SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
10 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_S
11
12 -- Schema mydb
13
14
15 DROP DATABASE IF EXISTS `my_dealership_database`;
16 CREATE DATABASE IF NOT EXISTS `my_dealership_database`;
17 USE `my_dealership_database`;
18
19 -- Table `Customer Address`
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

The 'Action Output' pane at the bottom shows the execution results for lines 35 through 39 of the script:

| Time     | Action  | Response   | Duration / Fetch Time |
|----------|---|--|-----------------------|
| 18:10:42 | DROP TABLE IF EXISTS `Customer_Discount`  | 0 row(s) affected, 1 warning(s): 1051 Unknown table '... | 0.00079 sec           |
| 18:10:42 | CREATE TABLE IF NOT EXISTS `Customer_Discount` ( `customer_id` TINYINT NOT NULL, `discount_id` TINYINT NULL, ...  | 0 row(s) affected  | 0.0098 sec            |
| 18:10:42 | DROP TABLE IF EXISTS `Salesperson`  | 0 row(s) affected, 1 warning(s): 1051 Unknown table '... | 0.00072 sec           |
| 18:10:42 | CREATE TABLE IF NOT EXISTS `Salesperson` ( `salesperson_id` TINYINT NOT NULL AUTO_INCREMENT, `employeeid_fk`...   | 0 row(s) affected  | 0.0070 sec            |
| 18:10:42 | DROP TABLE IF EXISTS `Mechanic`   | 0 row(s) affected, 1 warning(s): 1051 Unknown table '... | 0.00098 sec           |
| 18:10:42 | CREATE TABLE IF NOT EXISTS `Mechanic` ( `mechanic_id` TINYINT NOT NULL AUTO_INCREMENT, `salary` VARCHAR(2...      | 0 row(s) affected  | 0.011 sec             |
| 18:10:42 | DROP TABLE IF EXISTS `Manger`   | 0 row(s) affected, 1 warning(s): 1051 Unknown table '... | 0.00097 sec           |
| 18:10:42 | CREATE TABLE IF NOT EXISTS `Manger` ( `manger_id` TINYINT NOT NULL AUTO_INCREMENT, `salary` VARCHAR(45) N...      | 0 row(s) affected  | 0.012 sec             |
| 18:10:42 | DROP TABLE IF EXISTS `Car Part`   | 0 row(s) affected, 1 warning(s): 1051 Unknown table '... | 0.00019 sec           |
| 18:10:42 | CREATE TABLE IF NOT EXISTS `Car Part` ( `part_id` TINYINT NOT NULL, `car_type` VARCHAR(45) NOT NULL, `Price` L... | 0 row(s) affected  | 0.0088 sec            |
| 18:10:42 | DROP TABLE IF EXISTS `Car_Part_Table`   | 0 row(s) affected, 1 warning(s): 1051 Unknown table '... | 0.00013 sec           |
| 18:10:42 | CREATE TABLE IF NOT EXISTS `Car_Part_Table` ( `VIN_FK` TINYINT NULL, `part_id_fk` TINYINT NULL, INDEX `VIN_FK_... | 0 row(s) affected  | 0.014 sec             |
| 18:10:42 | SET SQL_MODE=@OLD_SQL_MODE  | 0 row(s) affected  | 0.00016 sec           |
| 18:10:42 | SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS  | 0 row(s) affected  | 0.00015 sec           |
| 18:10:42 | SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS  | 0 row(s) affected  | 0.00015 sec           |

At the bottom, a status message indicates: 'Active schema was cleared'.

## Section IX: Inserting Data

### 6. inserts.sql

The screenshot shows a database management interface with a sidebar on the left containing sections: MANAGEMENT (Server Status, Client Connections, Users and Privileges, Status and System Variables, Data Export, Data Import/Restore), INSTANCE (Startup / Shutdown, Server Logs, Options File), and PERFORMANCE (Dashboard, Performance Reports, Performance Schema Setup). The main window is titled 'Local instance 3306' and has tabs for 'databasemodel', 'insert', and 'tests'. The 'insert' tab is active, showing a script named 'inserts.sql' with the following content:

```
-- Script name: inserts.sql
-- Author: Ahmad Afghan
-- Purpose: insert sample data to test the integrity of this database system

USE 'my_dealership_database';
select * from `Customer Address`;

INSERT INTO `Customer Address` (address_id, number, street, city, state, zip, country)
VALUE (1, 2629, 'Alderwood Dr', 'San Jose', 'CA', 95132, 'United States'),
```

Below the script, a 'Result Grid' shows the data inserted into the 'Customer Address' table:

| address_id | street       | number | city       | state | zip   | country       |
|------------|--------------|--------|------------|-------|-------|---------------|
| 1          | Alderwood Dr | 2629   | San Jose   | CA    | 95132 | United States |
| 2          | Shona court  | 4064   | San Jose   | CA    | 95124 | United States |
| 3          | Darcot Court | 4801   | Union City | CA    | 94587 | Canada        |
| 4          | Shona drive  | 434    | Frederick  | CA    | 95124 | United States |

Below the result grid, the 'Action Output' section shows the execution log of the script. The log includes the time, action, response, and duration for each statement executed.

| Time     | Action   | Response   | Duration / Fetch Time |
|----------|--|--|-----------------------|
| 18:05:28 | INSERT INTO 'Supplier' (supplier_id, supplier_name, supplier_number, supplier_email) VALUE (32, 'HONDA', '543-453-3345')   | 3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0 | 0.0021 sec            |
| 18:05:28 | INSERT INTO 'Dealership_Supplier' (dealership_id, supplier_id) VALUE (11, 32), (11, 34), (11, 33)  | 3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0 | 0.0024 sec            |
| 18:05:28 | INSERT INTO 'Car Engine' (engine_id, price) VALUE (77, '\$3000'), (23, '\$3423'), (99, '\$2342')   | 3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0 | 0.0013 sec            |
| 18:05:28 | INSERT INTO 'Car Type' (type_id, car_type) VALUE (01, 'sedan'), (02, 'Truck-V8'), (03, 'SUV')  | 3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0 | 0.0013 sec            |
| 18:05:28 | INSERT INTO CAR (VIN, car_model, car_year, car_color, car_engine_Fk, car_type_Fk) VALUE (22, 'Honda Accord', '2017', 'Whit...', (2, '12-05-2019'), (3, '12-09-2015'))        | 3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0 | 0.0016 sec            |
| 18:05:28 | INSERT INTO 'Warranty' (warranty_id, effective_date) VALUE (1, '10-22-2020'), (2, '12-05-2019'), (3, '12-09-2015')   | 3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0 | 0.0012 sec            |
| 18:05:28 | INSERT INTO 'Car_Warranty' (VIN, warranty_id) VALUE (22, 1), (44, 3), (33, 2)  | 3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0 | 0.0013 sec            |
| 18:05:28 | INSERT INTO 'Customer_Car' (customer_id_Fk, VIN_id_Fk) VALUE (1, 22), (2, 33), (3, 44)   | 3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0 | 0.0012 sec            |
| 18:05:28 | INSERT INTO 'Discount' (Discount_id, discount_type) VALUE (01, '20% discount'), (02, '30% discount'), (03, '40% discount')   | 3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0 | 0.0013 sec            |
| 18:05:28 | INSERT INTO 'Customer_Discount' (customer_id, discount_id) VALUE (1, 02), (2, 03), (3, 01)   | 3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0 | 0.0012 sec            |
| 18:05:28 | INSERT INTO 'Salesperson' (salesperson_id, employeeid_Fk, starting_date, salary) VALUE (1, 1, '12/2/2018', '\$20'), (2, 3, '12/2/2018', '\$20'), (3, 1, '12/2/2018', '\$20') | 2 row(s) affected Records: 2 Duplicates: 0 Warnings: 0 | 0.0015 sec            |
| 18:05:28 | INSERT INTO 'Mechanic' (mechanic_id, salary, employeeid_Fk) VALUE (1, '\$45 per hour', 2), (2, '\$30 per hour', 3), (3, '\$30 per hour', 1)                                  | 3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0 | 0.0014 sec            |
| 18:05:28 | INSERT INTO 'Manger' (manger_id, salary, employeeid_Fk) VALUE (1, '\$100 per hour', 1), (2, '\$80 per hour', 2), (3, '\$90 per hour', 3)                                     | 3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0 | 0.0015 sec            |
| 18:05:28 | INSERT INTO 'Car Part' (part_id, car_type, Price) VALUE (1, 'water pump', 100), (2, 'tire', 200), (3, 'breaks', 240)   | 3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0 | 0.0012 sec            |
| 18:05:28 | INSERT INTO 'Car_Part_Table' (VIN_Fk, part_id_Fk) VALUE (33, 1), (22, 3), (22, 3)  | 3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0 | 0.0014 sec            |

The interface also shows a 'Limit to 1000 rows' option and a 'Query Completed' status at the bottom.

## Section X: Testing

### 7. Test.sql

Local Instance 3306

Administration Schemas databasemodel insert tests

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

Script name: tests.sql

Author: Ahmad Afghan

Purpose: test the integrity of this database system

-- database to test data into

USE 'my\_dealership\_database';

SET SQL\_SAFE\_UPDATES=0;

select \* from 'Salesperson';

-- 1. testing table Customer Address

Result Grid

| salesperson_id | employeeid_fk | starting_da... | salary |
|----------------|---------------|----------------|--------|
| 2              | 3             | 12/31/2019     | \$18   |
| NULL           | NULL          | NULL           | NULL   |

Salesperson 2

Action Output

|     | Time     | Action  | Response  | Duration / Fetch Time |
|-----|----------|---|---|-----------------------|
| 294 | 18:08:18 | UPDATE 'Customer Address' SET number = 2625 WHERE street = 'Alderwood Dr'             | 1 row(s) affected Rows matched: 1 Changed: 1 Warni... | 0.0011 sec            |
| 295 | 18:08:18 | DELETE FROM 'Account' WHERE password = 'nice'   | 1 row(s) affected                                     | 0.0011 sec            |
| 296 | 18:08:18 | UPDATE 'Account' SET password = 'nice3333' WHERE password = 'nice23'                  | 1 row(s) affected Rows matched: 1 Changed: 1 Warni... | 0.0012 sec            |
| 297 | 18:08:18 | DELETE FROM 'Credit Card Type' WHERE billing_city = 'Fremont'                         | 0 row(s) affected                                     | 0.00028 sec           |
| 298 | 18:08:18 | UPDATE 'Credit Card Type' SET billing_zip = '95132' WHERE billing_country = 'USA'     | 2 row(s) affected Rows matched: 4 Changed: 2 War...   | 0.0013 sec            |
| 299 | 18:08:18 | UPDATE 'Credit Card' SET CCV = 999 WHERE cc_number = 4769204591402517                 | 1 row(s) affected Rows matched: 1 Changed: 1 Warni... | 0.0011 sec            |
| 300 | 18:08:18 | UPDATE 'Customer' SET gender = 'Female' WHERE email = 'aafghan@mail.sfsu.edu'         | 1 row(s) affected Rows matched: 1 Changed: 1 Warni... | 0.0012 sec            |
| 301 | 18:08:18 | UPDATE 'Employee' SET DOB = '10-25-1993' WHERE full_name = 'Carlos Ortiz'             | 1 row(s) affected Rows matched: 1 Changed: 1 Warni... | 0.0017 sec            |
| 302 | 18:08:18 | UPDATE 'Service Shop' SET service_shop_id = 22 WHERE email = 'serviceshop@yahoo.com'  | 1 row(s) affected Rows matched: 1 Changed: 1 Warni... | 0.00095 sec           |
| 303 | 18:08:18 | UPDATE 'Dealership Address' SET State = 'VA' WHERE street = '4599 Stevens Creek Blvd' | 1 row(s) affected Rows matched: 1 Changed: 1 Warni... | 0.0017 sec            |
| 304 | 18:08:18 | UPDATE 'Finance Plan' SET amount = '400' WHERE plan_id = '1'                          | 1 row(s) affected Rows matched: 1 Changed: 1 Warni... | 0.0013 sec            |
| 305 | 18:08:18 | UPDATE 'Car' SET car_year = '2021' WHERE car_model = 'Honda Civic'                    | 1 row(s) affected Rows matched: 1 Changed: 1 Warni... | 0.0012 sec            |
| 306 | 18:08:18 | UPDATE 'Warranty' SET effective_date = '10-31-1992' WHERE warranty_id = 2             | 1 row(s) affected Rows matched: 1 Changed: 1 Warni... | 0.0010 sec            |
| 307 | 18:08:18 | DELETE FROM 'Salesperson' WHERE salary = '\$20'                                       | 1 row(s) affected                                     | 0.0010 sec            |
| 308 | 18:08:18 | UPDATE 'Salesperson' SET salary = '\$18' WHERE salesperson_id = 2                     | 1 row(s) affected Rows matched: 1 Changed: 1 Warni... | 0.0012 sec            |

Query Completed

## Section XI: Testing Table

| Entity           | SQLQuery | Pass//Failed | Error Description  | Possible Solution        |
|------------------|----------|--------------|--|--------------------------|
| Customer Address | Delete   | pass         | None   | None                     |
| Customer Address | Update   | pass         | None   | None                     |
| Account          | Delete   | pass         | None   | None                     |
| Account          | Update   | pass         | None   | None                     |
| Credit Card Type | Delete   | Failed       | -- Error Code: 1451. Cannot delete or update a parent row: a foreign key constraint fails  | Set Action to On Cascade |
| Credit Card Type | Update   | Pass         | None   | None                     |
| Credit Card      | Delete   | Failed       | -- Error Code: 1451. Cannot delete or update a parent row: a foreign key constraint fails<br>-- Error Code: 1054. Unknown column 'country' in 'where clause' | Set Action to On Cascade |
| Credit Card      | Update   | Pass         | None   | None                     |
| Customer         | Delete   | Failed       | -- Error Code: 1451. Cannot delete or update a parent row: a foreign key constraint fails  | Set Action to On Cascade |
| Customer         | Update   | Pass         | None   | None                     |
| Employee         | Delete   | Failed       | -- Error Code: 1451. Cannot delete or update a parent row: a foreign key constraint fails  | Set Action to On Cascade |
| Employee         | Update   | Pass         | None   | None                     |
| Service Shop     | Delete   | Failed       | -- Error Code: 1451. Cannot delete or update a parent row: a foreign key constraint fails  | Set Action to On Cascade |
| Service Shop     | Update   | Pass         | None   | None                     |

|                    |        |        |   |                          |
|--------------------|--------|--------|---|--------------------------|
| Dealership Address | Delete | Failed | -- Error Code: 1451. Cannot delete or update a parent row: a foreign key constraint fails | Set Action to On Cascade |
| Dealership Address | Update | Pass   | None  | None                     |
| Finance Plan       | Delete | Failed | -- Error Code: 1451. Cannot delete or update a parent row: a foreign key constraint fails | Set Action to On Cascade |
| Finance Plan       | Update | Pass   | None  | None                     |
| Car                | Delete | Failed | -- Error Code: 1451. Cannot delete or update a parent row: a foreign key constraint fails | Set Action to On Cascade |
| Car                | Update | Pass   | None  | None                     |
| Warranty           | Delete | Failed | -- Error Code: 1451. Cannot delete or update a parent row: a foreign key constraint fails | Set Action to On Cascade |
| Warranty           | Update | Pass   | None  | None                     |
| Salesperson        | Delete | Pass   | None  | None                     |
| Salesperson        | Update | Pass   | None  | None                     |
| Manger             | Delete | Pass   | None  | None                     |
| Manger             | Update | Pass   | None  | None                     |
| Mechanic           | Delete | Pass   | None  | None                     |
| Mechanic           | Update | Pass   | None  | None                     |
| Car Part           | Delete | Failed | -- Error Code: 1451. Cannot delete or update a parent row: a foreign key constraint fails | Set Action to On Cascade |
| Car Part           | Update | Pass   | None  | None                     |
| Discount           | Delete | Failed | -- Error Code: 1451. Cannot delete or update a parent row: a foreign key constraint fails | Set Action to On Cascade |

|          |        |        |   |                          |
|----------|--------|--------|---|--------------------------|
| Discount | Update | Failed | -- Error Code: 1451. Cannot delete or update a parent row: a foreign key constraint fails | Set Action to On Cascade |
|----------|--------|--------|---|--------------------------|