**Car Rental System Database ERD**

**Introduction:**

The main objective of the application car Rental System require a temporary vehicle, for example those who do not own their own car, or owners of damaged or destroyed vehicles who are awaiting repair or insurance compensation or travelers who are out of town.

In Car Rental System a customer can book a car for trip, marriage, business meeting, office to home, home to office, home to a market where ever they want to travel as simple as that. Car Rental System specializing in renting cars to clients.

Customers can see all the category of vehicles with their prices ,if the user wants it then they can rent it by filling the proceed forms. Their user will write the full name, phone number, email, id number, Gender , location and then submit. After they register,they need to pay the charge of the rental vehicles. The locations of will be stored with their start date and end date.

In Car Rental System I have create three categories were tables undergone.

* Inventory Data
* Business Data
* Customer Data

There are 13 tables in my database design as follows:

**Inventory Data:**

1. Equipment\_type
2. equipment
3. vehicle\_has\_equipment
4. vehicle
5. vehicle\_type
6. rental\_has\_equipment\_type

**Business Data:**

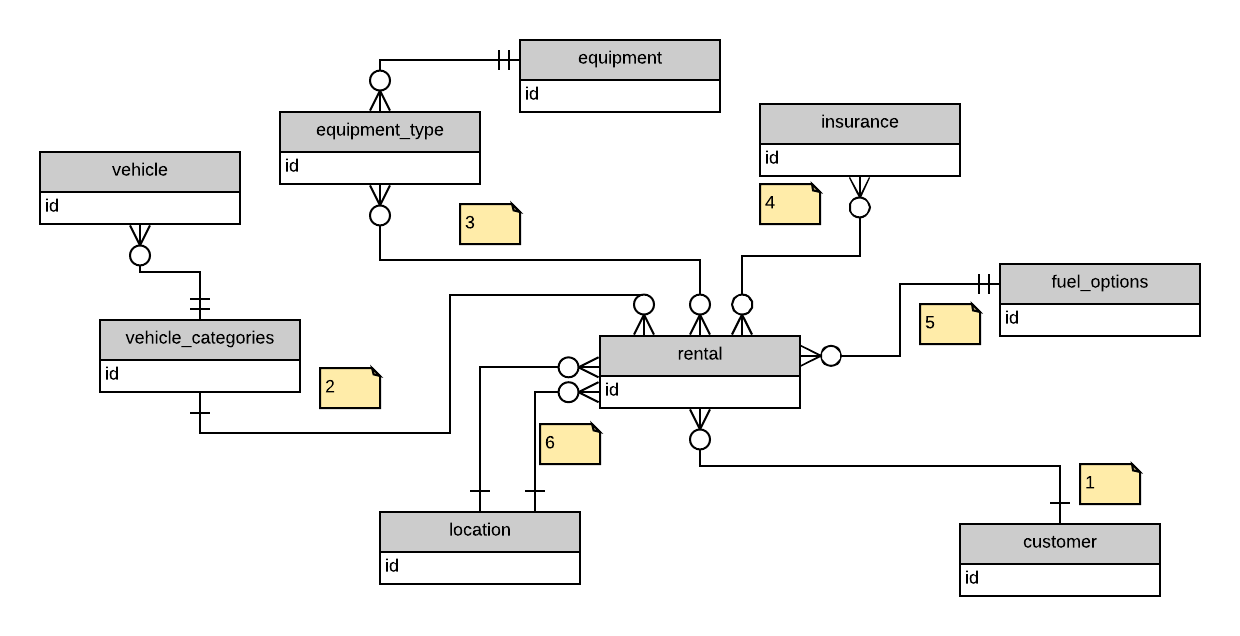
1. rental\_has\_insurance
2. Insurance
3. location
4. rental

**Customer Data:**

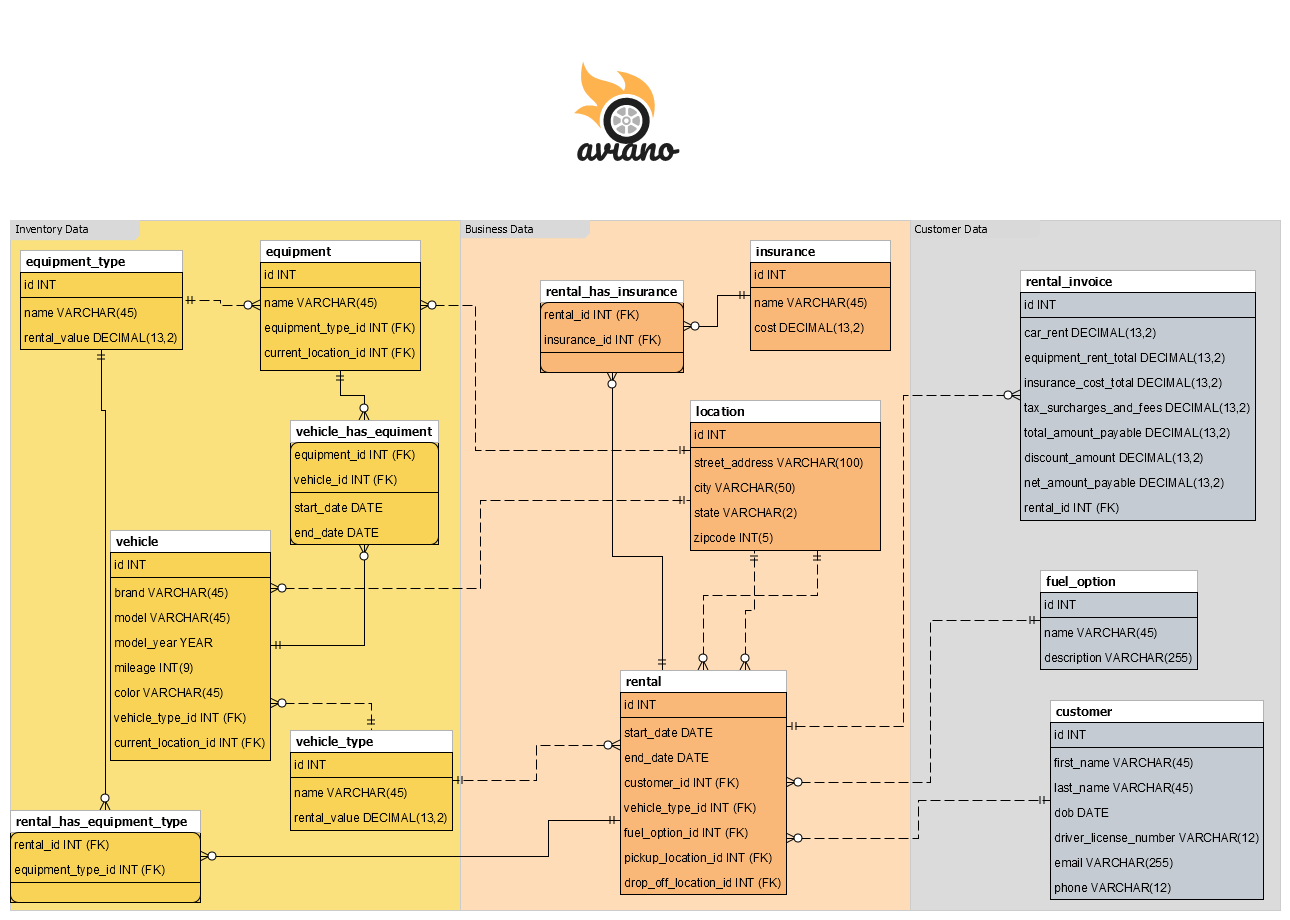
1. rental\_invoice
2. fuel\_option
3. Customer

**ERD Diagrams:**

**Initial ERD of Car Rental System**



**Final ERD Diagram:**

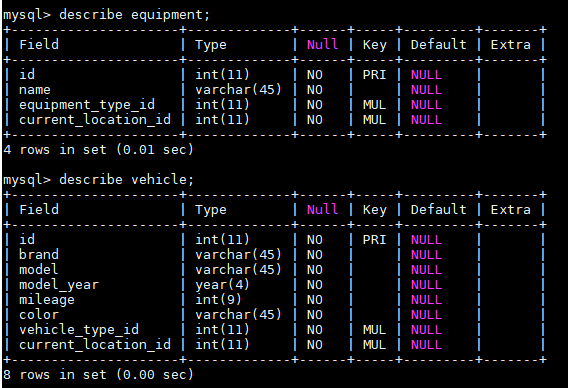


**Relational database design process**

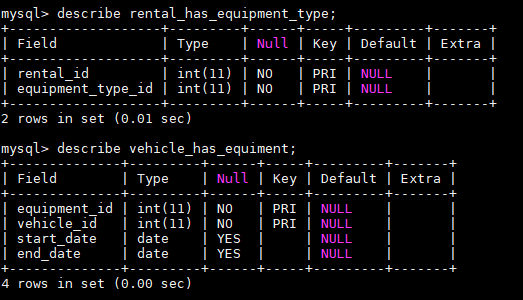
| **Entity Name** | **Why Entity is Included** | **How Entity is Related to Other Entities** |
| --- | --- | --- |
| location | Since there would be various locations in our system, we will set up a location table to represent the geographic information. | The primary key(PK) *ID* relates to *vehicle*, *equipment*, *rental*  entities to tell us which location it belongs to. |
| vehicle | Each vehicle will be assigned a unique ID, and some detailed information such as Name and Contact Number and other basic descriptions. | Vehicle PK *ID* is a foreign key(FK) for  *vehicle\_has\_equipment*  tells us which vehicle has equipment. |
| Vehicle\_has\_equipment | This entity will have all the information about the vehicle’s equipment. | Equipment\_id and vehicle\_id are foreign keys(FK) which has relation with vehicle and equipment. |
| Customers | This stores the details of all the customers who uses the services. | PK *ID* is used as an FK in *rental*. From *rental* we can extract the information of the customer who has made a booking. |
| equipment | This entity stores the details of all equipment name, type and location. | The PK *ID*  is a FK in vehicle\_has\_equipment as equipment\_ID. |
| Equipment\_type | This will have the details about the type of equipment. | The ID(PK) is a foreign key for table equipment and rental\_has\_equipment\_type. |
| Vehicle\_type | This will have details about the restaurant such as address, phone no., city located in etc. | PK *ID*is used as FK in *vehicle as vehicle\_type\_id(FK)* to extract the vehicle information. |
| Rental\_has\_equipment\_type | This entity store the information about which vehicle has equipment type |  |
| Rental\_has\_insurance | It will have all the details of the vehicle insurance. |  |
| insurance | It is used to store all the addresses of customers, hotels, restaurants, and organizers. | PK *ID* is used as FK in *rental\_has\_insurance* which stores all the insurance information. |
| rental | This entity will store all the information of the vehicle, customer and pickup and drop off locations | PK *ID* is a FK in *rental\_has\_equipment\_type, rental\_invoice & rental\_has\_insurance.* |
| Rental\_invoice | This entity gives the total cost for an rent vehicle |  |
| Fuel\_option | This entity is used to extract vehicle description. | Primary Key ID(PK) is a foreign key(Fk) in rental as fuel\_option\_id. |
| Customer | This entity stores the customer information to find the customers with their driving-license. | The Primary key ID(PK) is a foreign key(FK) in rental table as customer\_id(FK). |

**Data Definition Language Scripts**

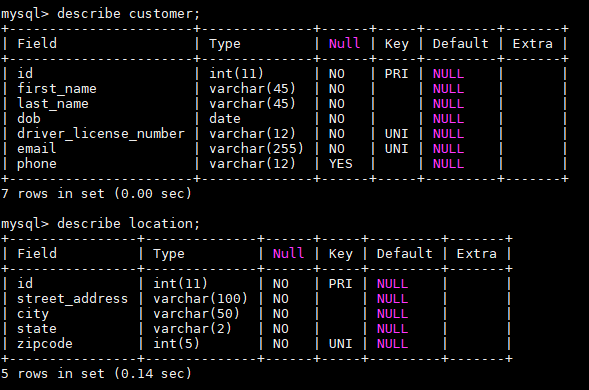
**Equipment & Vehicle Tables:**



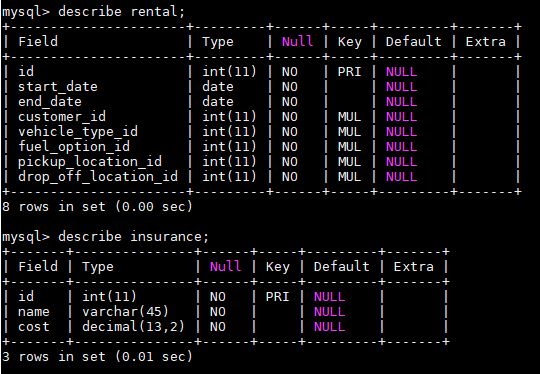
**rental\_has\_equipment & vehicle\_has\_equipment:**



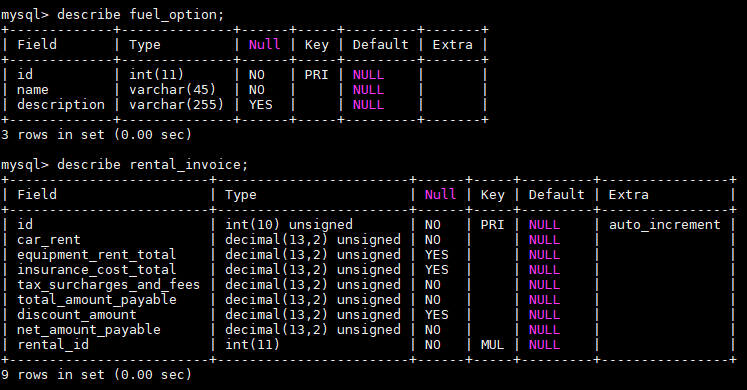
**Customer & Location Tables:**



**Rental & Insurance Tables:**



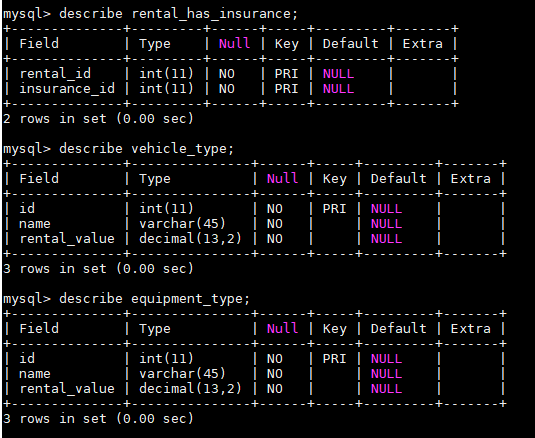
**Rental\_invoice & Fuel\_option Table:**



**Rental\_has\_Insurance Table**

**Vehicle\_type table**

**equipment\_type table:**



**Data Manipulation Language Scripts**

-- ---------------------------------------------------------------------------------------------------------------------------

**-- Data for table `location`**

-- -------------------------------------------------------------------------------------------------------------------------

START TRANSACTION;

INSERT INTO `location` (`id`, `street\_address`, `city`, `state`, `zipcode`) VALUES (1, '1001 Henderson St', 'Fort Worth', 'TX', 76102);

INSERT INTO `location` (`id`, `street\_address`, `city`, `state`, `zipcode`) VALUES (2, '300 Reunion Blvd', 'Dallas', 'TX', 75207);

INSERT INTO `location` (`id`, `street\_address`, `city`, `state`, `zipcode`) VALUES (3, '5911 Blair Rd NW', 'Washington', 'DC', 20011);

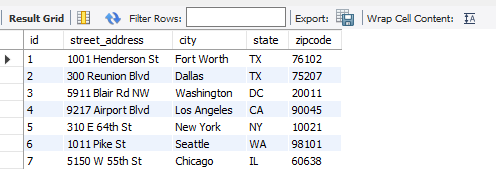
INSERT INTO `location` (`id`, `street\_address`, `city`, `state`, `zipcode`) VALUES (4, '9217 Airport Blvd', 'Los Angeles', 'CA', 90045);

INSERT INTO `location` (`id`, `street\_address`, `city`, `state`, `zipcode`) VALUES (5, '310 E 64th St', 'New York', 'NY', 10021);

INSERT INTO `location` (`id`, `street\_address`, `city`, `state`, `zipcode`) VALUES (6, '1011 Pike St', 'Seattle', 'WA', 98101);

INSERT INTO `location` (`id`, `street\_address`, `city`, `state`, `zipcode`) VALUES (7, '5150 W 55th St', 'Chicago', 'IL', 60638);

COMMIT;



-- --------------------------------------------------------------------------------------------------------------------------

**-- Before Insert Trigger for table `customer`**

-- --------------------------------------------------------------------------------------------------------------------------

DELIMITER $$

DROP TRIGGER IF EXISTS `age\_check` $$

CREATE DEFINER = CURRENT\_USER TRIGGER `age\_check` BEFORE INSERT ON `customer` FOR EACH ROW

BEGIN

DECLARE age INT UNSIGNED;

SELECT TIMESTAMPDIFF(YEAR, new.dob, CURDATE()) INTO age FROM DUAL;

IF (age < 21) THEN

SIGNAL SQLSTATE '45000'

SET MESSAGE\_TEXT = 'customer age\_check constraint on customer.dob failed';

END IF;

END$$

DELIMITER ;

-- -------------------------------------------------------------------------------------------------------------------------------

**-- Data for table `customer`**

-- -------------------------------------------------------------------------------------------------------------------------------

START TRANSACTION;

INSERT INTO `customer` (`id`, `first\_name`, `last\_name`, `dob`, `driver\_license\_number`, `email`, `phone`) VALUES (1, 'Kelby', 'Matterdace', '1974-05-22', 'V435899293', 'kmatterdace0@oracle.com', '181-441-7828');

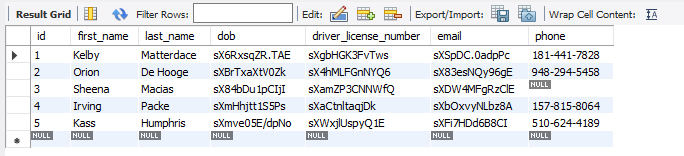
INSERT INTO `customer` (`id`, `first\_name`, `last\_name`, `dob`, `driver\_license\_number`, `email`, `phone`) VALUES (2, 'Orion', 'De Hooge', '1992-08-07', 'Z140530509', 'odehooge1@quantcast.com', '948-294-5458');

INSERT INTO `customer` (`id`, `first\_name`, `last\_name`, `dob`, `driver\_license\_number`, `email`, `phone`) VALUES (3, 'Sheena', 'Macias', '1981-03-10', 'W045654959', 'smacias3@amazonaws.com', NULL);

INSERT INTO `customer` (`id`, `first\_name`, `last\_name`, `dob`, `driver\_license\_number`, `email`, `phone`) VALUES (4, 'Irving', 'Packe', '1994-12-19', 'O232196823', 'ipacke4@cbc.ca', '157-815-8064');

INSERT INTO `customer` (`id`, `first\_name`, `last\_name`, `dob`, `driver\_license\_number`, `email`, `phone`) VALUES (5, 'Kass', 'Humphris', '1993-12-16', 'G055017319', 'khumphris5@xrea.com', '510-624-4189');

COMMIT;



**-- -----------------------------------------------------**

**-- Data for table `fuel\_option`**

**-- -----------------------------------------------------**

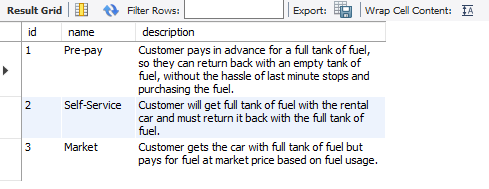
START TRANSACTION;

INSERT INTO `fuel\_option` (`id`, `name`, `description`) VALUES (1, 'Pre-pay', 'Customer pays in advance for a full tank of fuel, so they can return back with an empty tank of fuel, without the hassle of last minute stops and purchasing the fuel.');

INSERT INTO `fuel\_option` (`id`, `name`, `description`) VALUES (2, 'Self-Service', 'Customer will get full tank of fuel with the rental car and must return it back with the full tank of fuel.');

INSERT INTO `fuel\_option` (`id`, `name`, `description`) VALUES (3, 'Market', 'Customer gets the car with full tank of fuel but pays for fuel at market price based on fuel usage. ');

COMMIT;



**-- ------------------------------------------------------------------------------------------------------------------**

**-- Data for table `insurance`**

**-- ------------------------------------------------------------------------------------------------------------------**

START TRANSACTION;

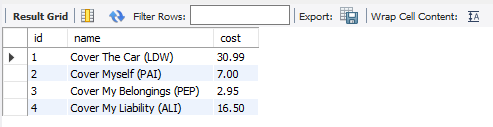
INSERT INTO `insurance` (`id`, `name`, `cost`) VALUES (1, 'Cover The Car (LDW)', 30.99);

INSERT INTO `insurance` (`id`, `name`, `cost`) VALUES (2, 'Cover Myself (PAI)', 7.00);

INSERT INTO `insurance` (`id`, `name`, `cost`) VALUES (3, 'Cover My Belongings (PEP)', 2.95);

INSERT INTO `insurance` (`id`, `name`, `cost`) VALUES (4, 'Cover My Liability (ALI)', 16.50);

COMMIT;



**-- ---------------------------------------------------------------------------------------------------------------------------------**

**-- Data for table `equipment\_type`**

**-- ---------------------------------------------------------------------------------------------------------------------------------**

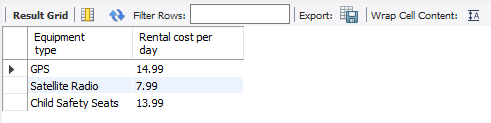
START TRANSACTION;

INSERT INTO `equipment\_type` (`id`, `name`, `rental\_value`) VALUES (1, 'GPS', 14.99);

INSERT INTO `equipment\_type` (`id`, `name`, `rental\_value`) VALUES (2, 'Satellite Radio', 7.99);

INSERT INTO `equipment\_type` (`id`, `name`, `rental\_value`) VALUES (3, 'Child Safety Seats', 13.99);

COMMIT;



**-- ------------------------------------------------------------------------------------------------------------------------**

**-- Data for table `equipment`**

**-- ----------------------------------------------------------------------------------------------------------------------------**

START TRANSACTION;

INSERT INTO `equipment` (`id`, `name`, `equipment\_type\_id`, `current\_location\_id`) VALUES (1, 'Garmin GPS', 1, 5);

INSERT INTO `equipment` (`id`, `name`, `equipment\_type\_id`, `current\_location\_id`) VALUES (2, 'Tomtom GPS', 1, 6);

INSERT INTO `equipment` (`id`, `name`, `equipment\_type\_id`, `current\_location\_id`) VALUES (3, 'Tomtom GPS', 1, 7);

INSERT INTO `equipment` (`id`, `name`, `equipment\_type\_id`, `current\_location\_id`) VALUES (4, 'Infant Child Seat', 3, 1);

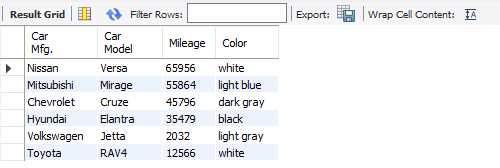
INSERT INTO `equipment` (`id`, `name`, `equipment\_type\_id`, `current\_location\_id`) VALUES (5, 'Child Seat', 3, 7);

INSERT INTO `equipment` (`id`, `name`, `equipment\_type\_id`, `current\_location\_id`) VALUES (6, 'Booster Seat', 3, 1);

INSERT INTO `equipment` (`id`, `name`, `equipment\_type\_id`, `current\_location\_id`) VALUES (7, 'Sirius XM Satellite Radio', 2, 5);

INSERT INTO `equipment` (`id`, `name`, `equipment\_type\_id`, `current\_location\_id`) VALUES (8, 'Sirius XM Satellite Radio', 2, 6);

COMMIT;



**-- --------------------------------------------------------------------------------------------------**

**-- Data for table `vehicle\_type`**

**-- --------------------------------------------------------------------------------------------------------**

START TRANSACTION;

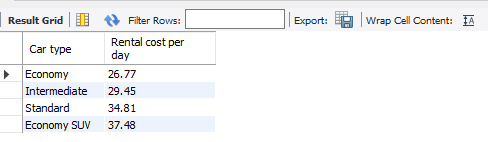
INSERT INTO `vehicle\_type` (`id`, `name`, `rental\_value`) VALUES (1, 'Economy', 26.77);

INSERT INTO `vehicle\_type` (`id`, `name`, `rental\_value`) VALUES (2, 'Intermediate', 29.45);

INSERT INTO `vehicle\_type` (`id`, `name`, `rental\_value`) VALUES (3, 'Standard', 34.81);

INSERT INTO `vehicle\_type` (`id`, `name`, `rental\_value`) VALUES (4, 'Economy SUV', 37.48);

COMMIT;



**-- -----------------------------------------------------------------------------------------------------------------------**

**-- Data for table `vehicle`**

**-- ------------------------------------------------------------------------------------------------------------------------**

START TRANSACTION;

INSERT INTO `vehicle` (`id`, `brand`, `model`, `model\_year`, `mileage`, `color`, `vehicle\_type\_id`, `current\_location\_id`) VALUES (1, 'Nissan', 'Versa', 2016, 65956, 'white', 1, 1);

INSERT INTO `vehicle` (`id`, `brand`, `model`, `model\_year`, `mileage`, `color`, `vehicle\_type\_id`, `current\_location\_id`) VALUES (2, 'Mitsubishi', 'Mirage', 2017, 55864, 'light blue', 1, 6);

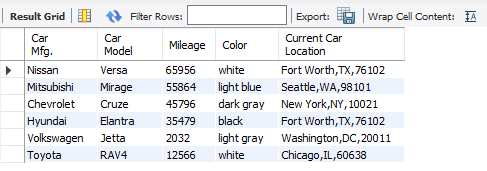
INSERT INTO `vehicle` (`id`, `brand`, `model`, `model\_year`, `mileage`, `color`, `vehicle\_type\_id`, `current\_location\_id`) VALUES (3, 'Chevrolet', 'Cruze', 2017, 45796, 'dark gray', 2, 5);

INSERT INTO `vehicle` (`id`, `brand`, `model`, `model\_year`, `mileage`, `color`, `vehicle\_type\_id`, `current\_location\_id`) VALUES (4, 'Hyundai', 'Elantra', 2018, 35479, 'black', 2, 1);

INSERT INTO `vehicle` (`id`, `brand`, `model`, `model\_year`, `mileage`, `color`, `vehicle\_type\_id`, `current\_location\_id`) VALUES (5, 'Volkswagen', 'Jetta', 2019, 2032, 'light gray', 3, 3);

INSERT INTO `vehicle` (`id`, `brand`, `model`, `model\_year`, `mileage`, `color`, `vehicle\_type\_id`, `current\_location\_id`) VALUES (6, 'Toyota', 'RAV4', 2018, 12566, 'white', 4, 7);

COMMIT;



**-- ---------------------------------------------------------------------------------------------------------------**

**-- Data for table `rental`**

**-- ---------------------------------------------------------------------------------------------------------------**

START TRANSACTION;

INSERT INTO `rental` (`id`, `start\_date`, `end\_date`, `customer\_id`, `vehicle\_type\_id`, `fuel\_option\_id`, `pickup\_location\_id`, `drop\_off\_location\_id`) VALUES (1, '2018-07-14', '2018-07-23', 1, 2, 1, 3, 5);

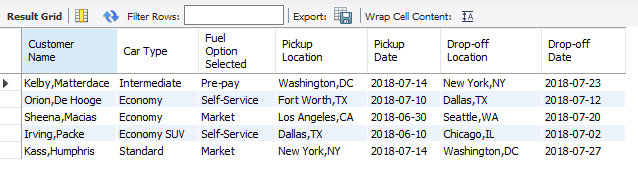
INSERT INTO `rental` (`id`, `start\_date`, `end\_date`, `customer\_id`, `vehicle\_type\_id`, `fuel\_option\_id`, `pickup\_location\_id`, `drop\_off\_location\_id`) VALUES (2, '2018-07-10', '2018-07-12', 2, 1, 2, 1, 2);

INSERT INTO `rental` (`id`, `start\_date`, `end\_date`, `customer\_id`, `vehicle\_type\_id`, `fuel\_option\_id`, `pickup\_location\_id`, `drop\_off\_location\_id`) VALUES (3, '2018-06-30', '2018-07-20', 3, 1, 3, 4, 6);

INSERT INTO `rental` (`id`, `start\_date`, `end\_date`, `customer\_id`, `vehicle\_type\_id`, `fuel\_option\_id`, `pickup\_location\_id`, `drop\_off\_location\_id`) VALUES (4, '2018-06-10', '2018-07-02', 4, 4, 2, 2, 7);

INSERT INTO `rental` (`id`, `start\_date`, `end\_date`, `customer\_id`, `vehicle\_type\_id`, `fuel\_option\_id`, `pickup\_location\_id`, `drop\_off\_location\_id`) VALUES (5, '2018-07-14', '2018-07-27', 5, 3, 3, 5, 3);

COMMIT;



**Queries of my choice:**

SELECT \* FROM location;

SELECT

name AS 'Car type', rental\_value AS 'Rental cost per day'

FROM

vehicle\_type;

SELECT

\*

FROM

fuel\_option;

SELECT

\*

FROM

insurance;

SELECT

name AS 'Equipment type',

rental\_value AS 'Rental cost per day'

FROM

equipment\_type;

SELECT

brand AS 'Car Mfg.',

model AS 'Car Model',

mileage AS 'Mileage',

color AS 'Color'

FROM

vehicle;

SELECT

v.brand AS 'Car Mfg.',

v.model AS 'Car Model',

v.mileage AS 'Mileage',

v.color AS 'Color',

CONCAT\_WS(',', l.city, l.state, l.zipcode) AS 'Current Car Location'

FROM

vehicle AS v

JOIN

location AS l ON v.current\_location\_id = l.id;

SELECT

CONCAT\_WS(',', c.first\_name, c.last\_name) AS 'Customer Name',

vt.name AS 'Car Type',

fo.name AS 'Fuel Option Selected',

CONCAT\_WS(',', l1.city, l1.state) AS 'Pickup Location',

r.start\_date as 'Pickup Date',

CONCAT\_WS(',', l2.city, l2.state) AS 'Drop-off Location',

r.end\_date as 'Drop-off Date'

FROM

rental AS r

JOIN

customer AS c ON r.customer\_id = c.id

JOIN

vehicle\_type AS vt ON r.vehicle\_type\_id = vt.id

JOIN

fuel\_option AS fo ON r.fuel\_option\_id = fo.id

JOIN

location AS l1 ON r.pickup\_location\_id = l1.id

JOIN

location AS l2 ON r.drop\_off\_location\_id = l2.id;

SELECT

c.first\_name,

c.last\_name,

TIMESTAMPDIFF(YEAR, c.dob, CURDATE()) AS age

FROM

customer AS c;

SELECT

CONCAT\_WS(',', c.first\_name, c.last\_name) AS 'Customer Name',

vt.name AS 'Car Type',

r.start\_date,

r.end\_date,

vt.rental\_value AS 'Car Rent per Day',

DATEDIFF(r.end\_date, r.start\_date) AS 'Total Days Car Rented',

vt.rental\_value \* DATEDIFF(r.end\_date, r.start\_date) AS 'Base Rent'

FROM

rental AS r,

vehicle\_type AS vt,

customer as c

WHERE

r.vehicle\_type\_id = vt.id AND r.customer\_id = c.id;

SELECT

CONCAT\_WS(',', c.first\_name, c.last\_name) AS 'Customer Name',

DATEDIFF(r.end\_date, r.start\_date) AS 'Total Days Car Rented',

SUM(i.cost) AS 'Insurance Cost per day',

(SUM(i.cost) \* DATEDIFF(r.end\_date, r.start\_date)) AS 'Total Insurance Cost'

FROM

rental AS r

JOIN

customer AS c ON r.customer\_id = c.id

JOIN

rental\_has\_insurance AS rhi ON r.id = rhi.rental\_id

JOIN

insurance AS i ON rhi.insurance\_id = i.id

GROUP BY r.id;

SELECT

CONCAT\_WS(',', c.first\_name, c.last\_name) AS 'Customer Name',

SUM(et.rental\_value) AS 'Equipment Rental Cost',

DATEDIFF(r.end\_date, r.start\_date) AS 'Total Days Car Rented',

(SUM(et.rental\_value) \* DATEDIFF(r.end\_date, r.start\_date)) AS 'Total Equipment Rental Cost'

FROM

rental AS r

JOIN

customer AS c ON r.customer\_id = c.id

JOIN

rental\_has\_equipment\_type AS rhet ON r.id = rhet.rental\_id

JOIN

equipment\_type AS et ON rhet.equipment\_type\_id = et.id

GROUP BY r.id;

SELECT

CONCAT\_WS(',', c.first\_name, c.last\_name) AS 'Customer Name',

ri.car\_rent AS 'Base Rent',

ri.equipment\_rent\_total AS 'Total Equipment Cost',

ri.insurance\_cost\_total AS 'Total Insurance Cost',

ri.discount\_amount AS 'Discount Amount',

ri.net\_amount\_payable AS 'Net Amount'

FROM

rental\_invoice AS ri

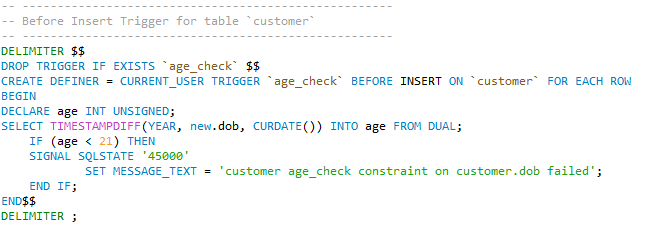
JOIN

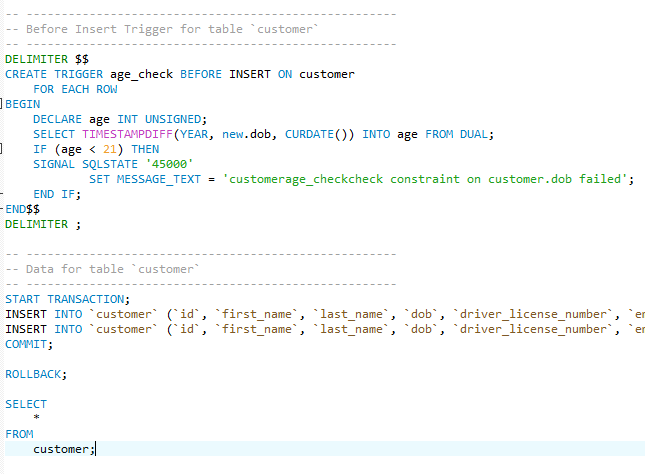
rental AS r ON ri.rental\_id = r.id

JOIN

customer AS c ON r.customer\_id = c.id;

**Triggers:**

****

****

**Python programming:**

import mysql.connector

from mysql.connector import errorcode

# Connecting to the MySQL database

try:

   mydb = mysql.connector.connect(

      user="root",

      password="root",

      host="localhost",

      database="aviano-db")

except mysql.connector.Error as err:

   if err.errno == errorcode.ER\_ACCESS\_DENIED\_ERROR:

      print("Invalid credentials")

   elif err.errno == errorcode.ER\_BAD\_DB\_ERROR:

      print("Database not found")

   else:

      print("Cannot connect to database:", err)

else:

    cursor1 = mydb.cursor()

# Retreive all tables Infomration using the below code

    q1 = ("SELECT \* FROM customer")

    cursor1.execute(q1)

    print("------Customer Table Information--------\n")

    for row in cursor1.fetchall():

        print(row,"\n")

    q2 = ("SELECT \* FROM equipment")

    cursor1.execute(q2)

    print("------Equipment Table Information------------\n")

    for row in cursor1.fetchall():

        print(row,"\n")

    q3 = ("SELECT \* FROM equipment\_type")

    cursor1.execute(q3)

    print("-----Equipment\_type Table Information---------\n")

    for row in cursor1.fetchall():

        print(row,"\n")

    q4 = ("SELECT \* FROM fuel\_option")

    cursor1.execute(q4)

    print("-----fuel\_option Table Information---------\n")

    for row in cursor1.fetchall():

        print(row,"\n")

    q5 = ("SELECT \* FROM insurance")

    cursor1.execute(q5)

    print("-----insurance Table Information---------\n")

    for row in cursor1.fetchall():

        print(row,"\n")

    q6 = ("SELECT \* FROM location")

    cursor1.execute(q6)

    print("-----location Table Information---------\n")

    for row in cursor1.fetchall():

        print(row,"\n")

    q7 = ("SELECT \* FROM rental")

    cursor1.execute(q7)

    print("-----rental Table Information---------\n")

    for row in cursor1.fetchall():

        print(row,"\n")

    q8 = ("SELECT \* FROM rental\_has\_equipment\_type")

    cursor1.execute(q8)

    print("-----rental\_has\_equipment\_type Table Information---------\n")

    for row in cursor1.fetchall():

        print(row,"\n")

    q9 = ("SELECT \* FROM rental\_has\_insurance")

    cursor1.execute(q9)

    print("-----rental\_has\_insurance Table Information---------\n")

    for row in cursor1.fetchall():

        print(row,"\n")

    q10 = ("SELECT \* FROM rental\_invoice")

    cursor1.execute(q10)

    print("-----rental\_invoice Table Information---------\n")

    for row in cursor1.fetchall():

        print(row,"\n")

    q11 = ("SELECT \* FROM vehicle")

    cursor1.execute(q11)

    print("----vehicle Table Information---------\n")

    for row in cursor1.fetchall():

        print(row,"\n")

    q12 = ("SELECT \* FROM vehicle\_has\_equiment")

    cursor1.execute(q12)

    print("----vehicle\_has\_equiment Table Information---------\n")

    for row in cursor1.fetchall():

        print(row,"\n")

    q13 = ("SELECT \* FROM vehicle\_type")

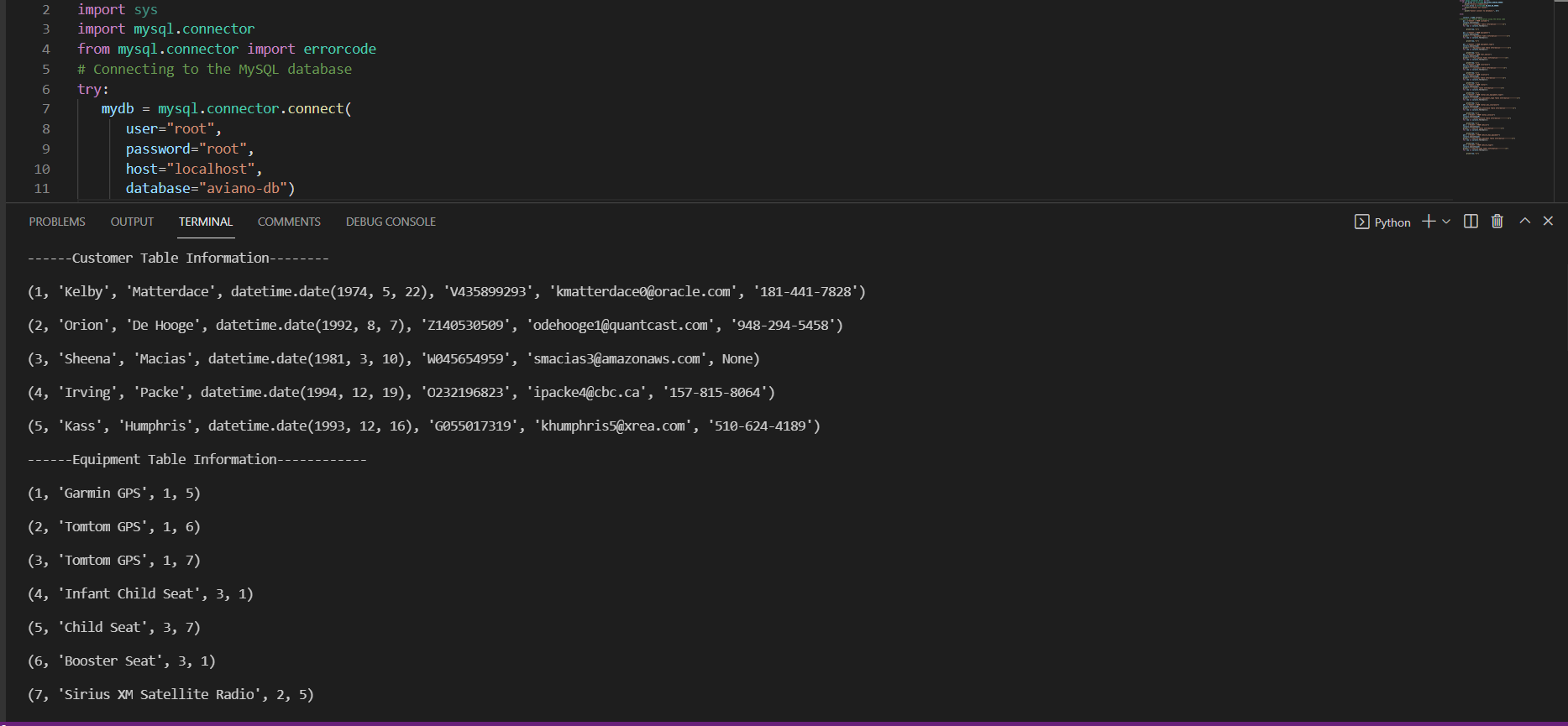
    cursor1.execute(q13)

    print("----vehicle\_type Table Information---------\n")

    for row in cursor1.fetchall():

        print(row,"\n")

**Output :**

****