

**Entities**

* 1. Customer - Description : This entity represents customer information.
* 2.Order - Description: This entity represents an instance of an order and its details.
* 3. Store Location - Description: Store locations to pick up and contact information.
* 4. Warehouse - Description: This represents the warehouses we own and its contact information.
* 5. Shipment - Description: This entity represents the order shipment details
* 6. Delivery Vehicles - Description: This entity represents the delivery vehicles utilized for delivery and its handler.
* 7. Employee - Description: This entity represents the employees responsible for the shipments and their vehicle.
* 8. Item - Description: This entity represents the item descriptions and its relationship to the order.

**Business Rules**

Each Customer can have one or more order.

Each order must be from one customer.

Order must contain one or more Item.

Item must belongs to one Order.

Item must be retrieved from one or many Store Location.

Store Location may have one Item.

Store Location must be supplied by one Warehouse.

Warehouse can supply one or many Location.

Warehouse must have one employee.

Employee must work for one Warehouse.

Employee must handle one or more shipment.

Shipment must be handled by one Employee.

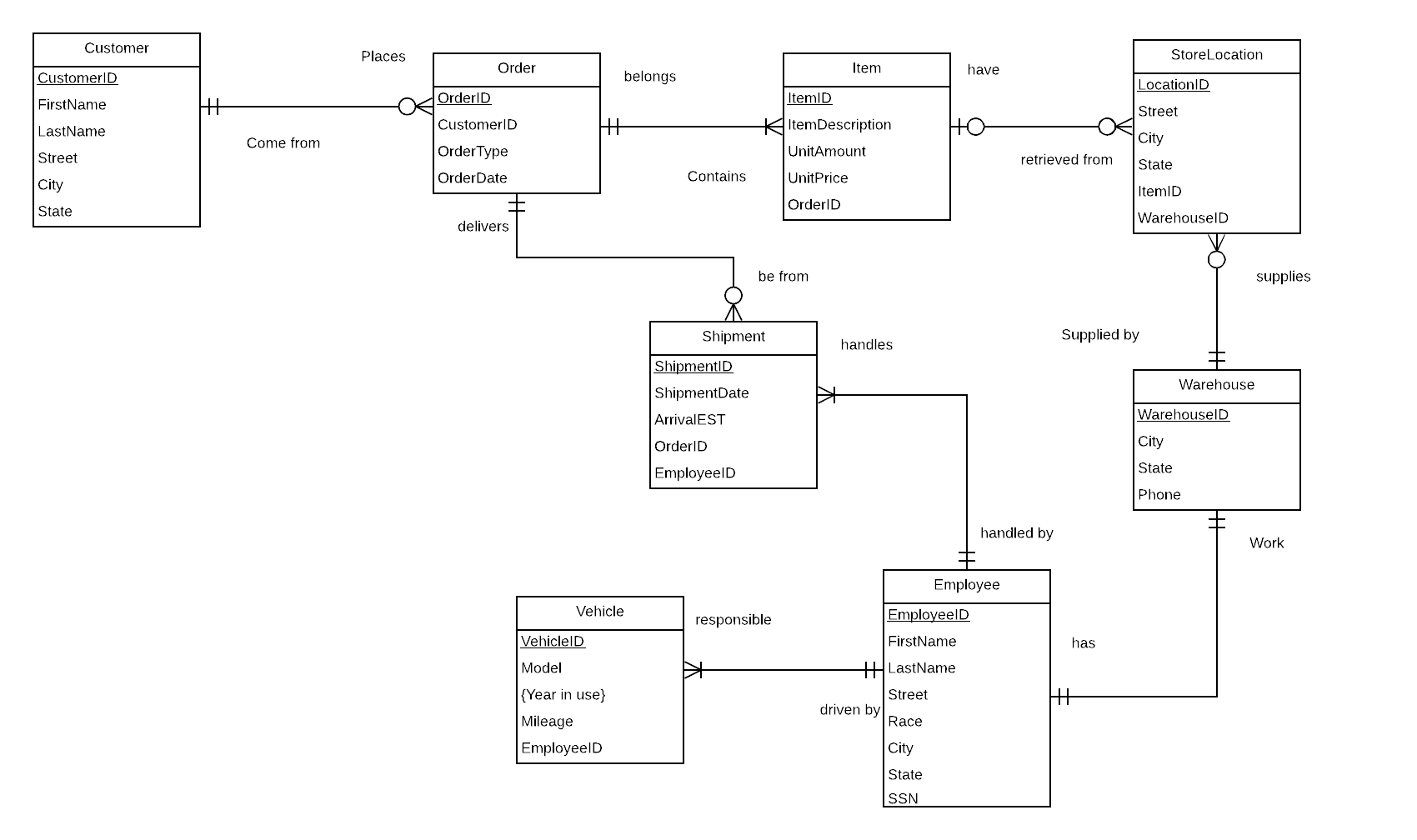
Employee must be responsible for one or more vehicle.

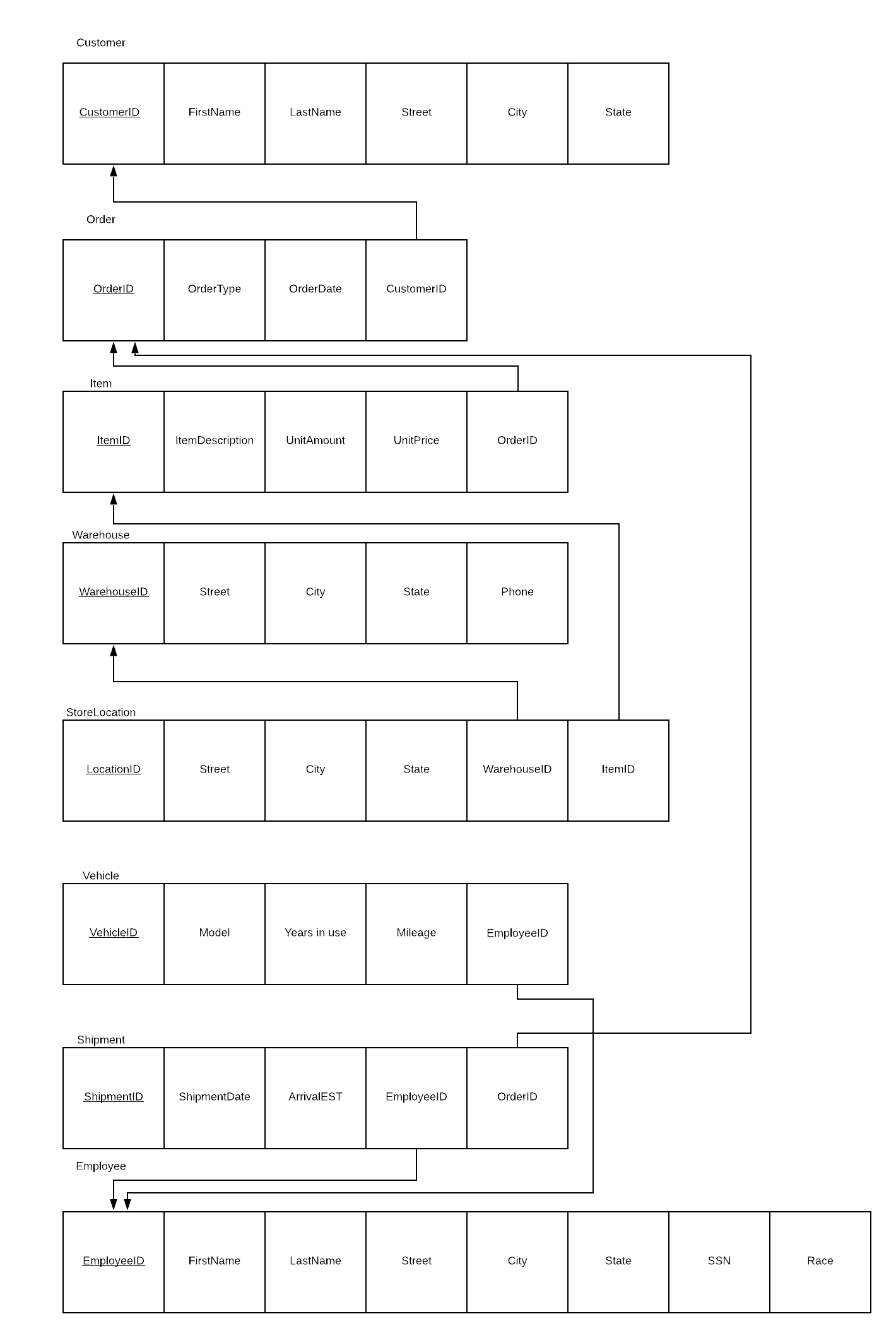
Vehicle must be driven by one employee.

Shipment must deliver one order.

Order can be from one or more shipment.

**Entity Relationship Diagram**



**Relational Model** 

**SQL Statements**

***a. Create all your tables***

CREATE TABLE Customer

**(**

CustomerID VARCHAR(15) NOT NULL,

FirstName VARCHAR(15) NOT NULL,

LastName VARCHAR(15) NOT NULL,

Street VARCHAR(25),

City VARCHAR(25),

State CHAR(2),

CONSTRAINT Customer\_PK PRIMARY KEY (CustomerID)

**);**

CREATE TABLE Employee

**(**

EmployeeID VARCHAR(15) NOT NULL,

FirstName VARCHAR(15) NOT NULL,

LastName VARCHAR(15) NOT NULL,

Race VARCHAR(45),

Street VARCHAR(45),

City VARCHAR(45),

State CHAR(2),

SSN INT(9) NOT NULL,

CONSTRAINT Employee\_PK PRIMARY KEY (EmployeeID)

**);**

CREATE TABLE Orders

**(**

OrderID VARCHAR(15) NOT NULL,

CustomerID VARCHAR(15) NOT NULL,

OrderType VARCHAR(50),

OrderDate DATE,

CONSTRAINT Order\_PK PRIMARY KEY (OrderID),

CONSTRAINT Order\_FK FOREIGN KEY (CustomerID) REFERENCES Customer (CustomerID)

**);**

CREATE TABLE Item

**(**

ItemID VARCHAR(15) NOT NULL,

OrderID VARCHAR(15) NOT NULL,

ItemDescription VARCHAR(50),

UnitAmount INT(5),

UnitPrice INT(50),

CONSTRAINT Item\_PK PRIMARY KEY (ItemID),

CONSTRAINT Item\_FK FOREIGN KEY (ItemID) REFERENCES Orders (OrderID)

**);**

CREATE TABLE Warehouse

**(**

WarehouseID VARCHAR(50) NOT NULL,

Street VARCHAR(25),

City VARCHAR(25),

State CHAR(2),

CONSTRAINT Warehouse\_PK PRIMARY KEY (WarehouseID)

**);**

CREATE TABLE StoreLocation

**(**

LocationID VARCHAR(50) NOT NULL,

ItemID VARCHAR(15) NOT NULL,

WarehouseID VARCHAR(50) NOT NULL,

Street VARCHAR(25),

City VARCHAR(25),

State CHAR(2),

CONSTRAINT StoreLocation\_PK PRIMARY KEY (LocationID),

CONSTRAINT StoreLocation\_FK1 FOREIGN KEY (ItemID) REFERENCES Item (ItemID),

CONSTRAINT StoreLocation\_FK2 FOREIGN KEY (WarehouseID) REFERENCES Warehouse (WarehouseID)

**);**

CREATE TABLE Vehicle

(

VehicleID VARCHAR(15) NOT NULL,

EmployeeID VARCHAR(15) NOT NULL,

Model VARCHAR(50),

Years\_In\_Use INT,

Mileage INT,

CONSTRAINT Vehicle\_PK PRIMARY KEY (VehicleID),

CONSTRAINT Vehicle\_FK FOREIGN KEY (EmployeeID) REFERENCES Employee (EmployeeID)

);

CREATE TABLE Shipment

(

ShipmentID VARCHAR(15) NOT NULL,

EmployeeID VARCHAR(15) NOT NULL,

OrderID VARCHAR(15) NOT NULL,

ShipmentDate DATE,

ArrivalEST DATE,

CONSTRAINT Shipment\_PK PRIMARY KEY (ShipmentID),

CONSTRAINT Shipment\_FK1 FOREIGN KEY (EmployeeID) REFERENCES Employee (EmployeeID),

CONSTRAINT Shipment\_FK2 FOREIGN KEY (OrderID) REFERENCES Orders (OrderID)

);

***b. Load data in your tables***

**Customer**

INSERT INTO Customer VALUES

(1, 'Dionne', 'MacKessock', '3326 Jenifer Crossing', 'Charleston', 'WV');

INSERT INTO Customer VALUES

(2, 'Ephraim', 'Casone', '977 Mariners Cove Court', 'New Bedford', 'MA');

INSERT INTO Customer VALUES

(3, 'Maren', 'Quayle', '7371 Beilfuss Junction', 'Hialeah', 'FL');

INSERT INTO Customer VALUES

(4, 'Gusti', 'Maletratt', '8 Parkside Park', 'Joliet', 'IL');

INSERT INTO Customer VALUES

(5, 'Kevyn', 'Anning', '632 Nova Trail', 'Albuquerque', 'NM');

INSERT INTO Customer VALUES

(6, 'Almeria', 'Korneichik', '5246 Crescent Oaks Hill', 'Silver Spring', 'MD');

**Orders**

INSERT INTO Orders VALUES

(1, '1', 'delivery', ' 2019/03/30 ');

INSERT INTO Orders VALUES

(2, '2', 'delivery', ' 2018/11/01 ');

INSERT INTO Orders VALUES

( 3, '3', 'delivery', ' 2018/07/11 ' );

INSERT INTO Orders VALUES

( 4, '4', 'pick-up', ' 2018/07/27 ' );

INSERT INTO Orders VALUES

( 5, '5', 'delivery', ' 2018/08/13 ' );

INSERT INTO Orders VALUES

( 6, '6', 'delivery', ' 2018/08/30 ' );

**Item**

INSERT INTO Item VALUES

(1, 1, 'Shrimp - 150 - 250', '144', '305.03');

INSERT INTO Item VALUES

(2, 2, 'Wine - Fume Blanc Fetzer', '3', '192.72');

INSERT INTO Item VALUES

(3, 3, 'Icecream - Dibs', '146', '95.3');

INSERT INTO Item VALUES

(4, 4, 'Sauce - Chili', '42', '251.87');

INSERT INTO Item VALUES

(5, 5, 'Kellogs Cereal In A Cup', '87', '4.2');

INSERT INTO Item VALUES

(6, 6, 'Miso Paste White', '44', '430.59');

INSERT INTO Item VALUES

(7, 7, 'Cookie Dough - Double', '85', '493.29');

INSERT INTO Item VALUES

(8, 8, 'Wine - Red, Gallo, Merlot', '149', '292.73');

INSERT INTO Item VALUES

(9, 9, 'Straw - Regular', '148', '169.29');

INSERT INTO Item VALUES

(10, 10, 'Mushroom - White Button', '28', '406.25');

INSERT INTO Item VALUES

(11, 11, 'Tea - Orange Pekoe', '10', '136.92');

**Warehouse**

INSERT INTO Warehouse VALUES

(1, '4 Armistice Hill', 'San Francisco', 'CA');

INSERT INTO Warehouse VALUES

(2, '31 Basil Trail', 'Houston', 'TX');

**StoreLocation**

INSERT INTO StoreLocation VALUES

(1, 1, 1, '0 Nelson Drive', 'Boca Raton', 'FL');

INSERT INTO StoreLocation VALUES

(2, 11, 2, '31737 Fisk Road', 'Asheville', 'NC');

INSERT INTO StoreLocation VALUES

(3, 13, 1, '530 Toban Way', 'Troy', 'MI');

INSERT INTO StoreLocation VALUES

(4, 14, 1, '73 Canary Park', 'Houston', 'TX');

**Employees**

INSERT INTO Employee VALUES

(827187902, 'Elden', 'Diter', 'Japanese', '400 Arrowood Lane', 'Laascaanood', 'WV', 168-24-7268);

INSERT INTO Employee VALUES

(922690952, 'Bertie', 'Annwyl', 'Central American', '52 Banding Park', 'Trakai', 'MA', 783-57-5057);

INSERT INTO Employee VALUES

(512312750, 'Sharia', 'Ruggles', 'Polynesian', '5640 Spaight Street', 'Shatura', 'FL', 316-16-0059);

INSERT INTO Employee VALUES

(241409828, 'Fredi', 'Gilhooly', 'Guamanian', '50 Moose Drive', 'Kadam', 'IL', 298-82-8202);

INSERT INTO Employee VALUES

(143790874, 'Natividad', 'Jennins', 'Tongan', '54 Longview Pass', 'Polen', 'NM', 286-52-7641);

INSERT INTO Employee VALUES

(399610794, 'Weidar', 'Hawarden', 'Guamanian', '3315 Namekagon Court', 'RÃ½maÅ™ov', 'MD', 324-33-8449);

INSERT INTO Employee VALUES

(603094362, 'Mufi', 'Ambroix', 'Taiwanese', '0000 Anthes Avenue', 'Bakar', 'DE', 568-20-5301);

INSERT INTO Employee VALUES

(908584570, 'Ula', 'Maffucci', 'Yuman', '4567 Thackeray Lane', 'Tamansari', 'FL', 459-91-7139);

INSERT INTO Employee VALUES

(832292701, 'Margarete', 'Cowlin', 'Pakistani', '18493 Parkside Park', 'Oslo', 'WI', 140-87-8396);

***c. Retrieve information from your table.***

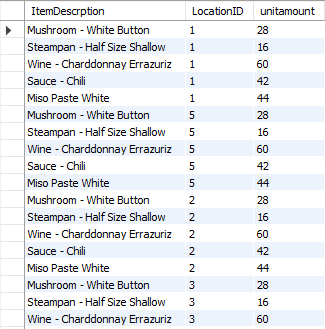
**1.Find store locations that carry items that are stocked between 10 and 80 and cost more than $20.**

Select ItemDescrption,LocationID, unitamount

From Item, StoreLocation

Where unitamount > 10 AND unitamount < 80.00 AND

unitprice > 20.00**;**

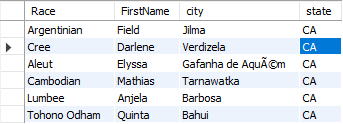


**2.Display the race, name and address of the employees in CA.**

Select Race, FirstName, city,state

from employee

where State = 'CA'**;**



**3. display the order number, customer number, order date and items description for pick up orders**

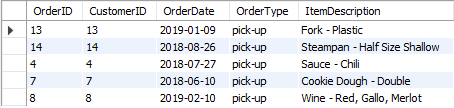
SELECT Orders.OrderID, Orders.CustomerID, Orders.OrderDate,Orders.OrderType,

Item.ItemDescription

FROM orders, item

WHERE orders.OrderID=Item.OrderID AND

Orders.ordertype = 'pick-up';



**4. Display item description and stock, make stock appear from lowest to highest for items that cost more than 50 dollars.**

Select ItemDescription, UnitAmount AS Stock

from item

WHERE UnitPrice > 50.00

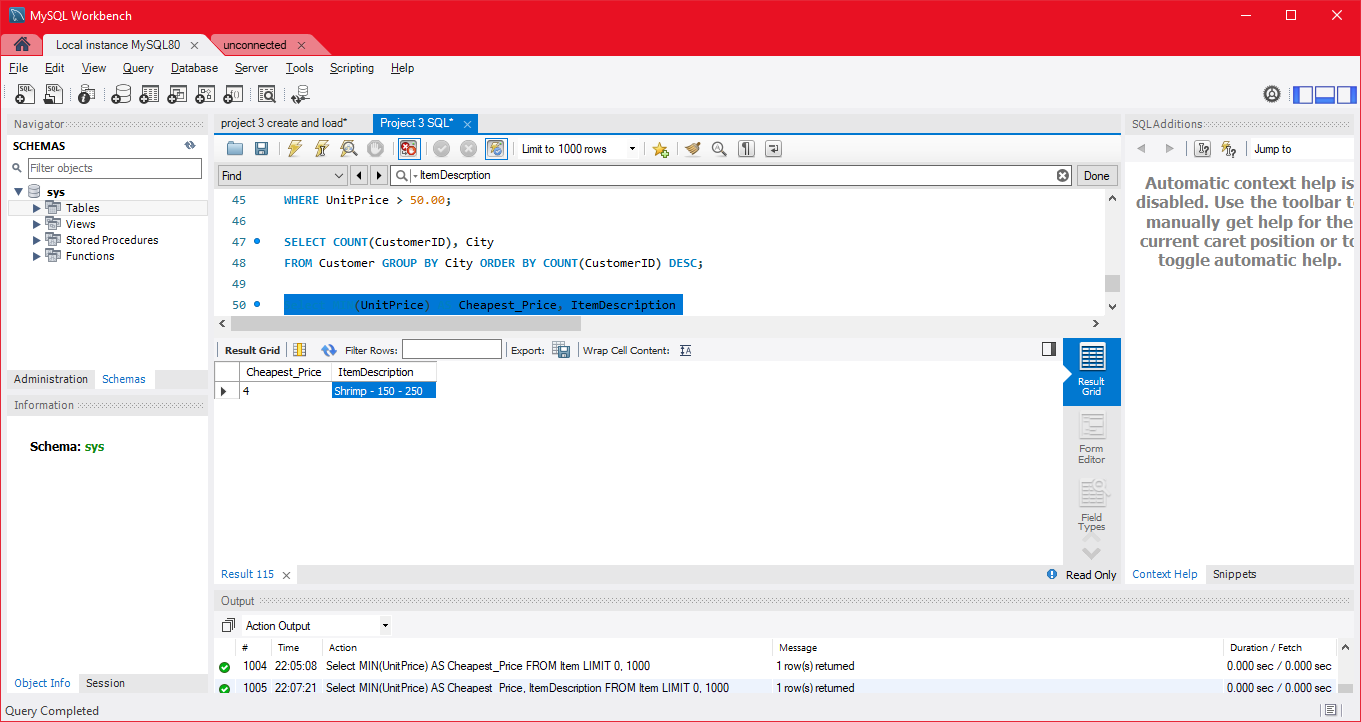
Order by UnitAmount ASC;



**5.Display the cheapest item and its description**

Select MIN(UnitPrice) AS Cheapest\_Price, ItemDescription

FROM Item;



**Conclusion**

During this project, we have an deeper understanding of databases. While were constructing the SQL codes, the relation model, and ER diagram, we understood the process of uploading a database and improving it. Initially, we had were planning to upload an database for Walmart entire store. However the idea seemed too big of a task so we narrowed it down to the delivery department of Walmart. Another challenge we had was that after we loaded the data, we found some minor mistakes around the ER diagram and Relation model. We had a tough decision between starting over or adjusting. Ultimately we chose to adjust and change the model. This project really help us understand more about database and database design.