CREATE TABLE Customer

(

CustomerID INTEGER NOT NULL IDENTITY PRIMARY KEY,

FirstName VARCHAR(30) NOT NULL,

MiddleName VARCHAR (30) NULL,

LastName VARCHAR(30) NOT NULL,

PhoneNumber VARCHAR(20) NOT NULL,

Address1 VARCHAR (30) NOT NULL,

Address2 VARCHAR (30) NOT NULL,

City VARCHAR (30) NOT NULL,

CState CHAR (2) NOT NULL,

ZipCode VARCHAR (10) NOT NULL,

)

CREATE TABLE Restaurant

(

RestaurantID INTEGER NOT NULL IDENTITY PRIMARY KEY,

RestaurantName VARCHAR(40) NOT NULL,

PhoneNumber VARCHAR(20) NOT NULL,

Address1 VARCHAR (30) NOT NULL,

Address2 VARCHAR (30) NOT NULL,

City VARCHAR (30) NOT NULL,

CState CHAR (2) NOT NULL,

ZipCode VARCHAR (10) NOT NULL,

RestaurantStatus VARCHAR(7) DEFAULT 'Unknown'

CONSTRAINT check\_Status CHECK (RestaurantStatus='Unknown' or RestaurantStatus='Open' or RestaurantStatus='Closed')

)

CREATE TABLE Food

(

FoodID INTEGER NOT NULL IDENTITY PRIMARY KEY,

RestaurantID INTEGER NOT NULL FOREIGN KEY REFERENCES Restaurant(RestaurantID),

FoodName VARCHAR (40) NOT NULL,

FoodDescription TEXT,

Price DECIMAL(5,2) NOT NULL

)

CREATE TABLE Car

(

PlateNumber VARCHAR(20) NOT NULL PRIMARY KEY,

Maker VARCHAR(20) NOT NULL,

Model VARCHAR(20) NOT NULL,

Color VARCHAR(20) NOT NULL,

)

CREATE TABLE Driver

(

DriverID INTEGER NOT NULL IDENTITY PRIMARY KEY,

FirstName VARCHAR(30) NOT NULL,

LastName VARCHAR(30) NOT NULL,

PhoneNumber VARCHAR(20) NOT NULL,

PlateNumber VARCHAR (20) NOT NULL FOREIGN KEY REFERENCES Car(PlateNumber)

)

CREATE TABLE Orders

(

OrderID INTEGER NOT NULL IDENTITY PRIMARY KEY,

CustomerID INTEGER NOT NULL FOREIGN KEY REFERENCES Customer(CustomerID),

OrderDate DATE NOT NULL DEFAULT GETDATE(),

PlacedTime TIME NOT NULL DEFAULT GETDATE(),

OrderStatus VARCHAR(10) NOT NULL,

CONSTRAINT check\_OrderStatus CHECK( OrderStatus='Prepared' or OrderStatus='Preparing' or OrderStatus='Delivering' or OrderStatus='Delivered')

)

SELECT \* FROM Orderline

CREATE TABLE Orderline

(

FoodID INTEGER NOT NULL FOREIGN KEY REFERENCES Food(FoodID),

OrderID INTEGER NOT NULL FOREIGN KEY REFERENCES Orders(OrderID),

Quantity INTEGER NOT NULL,

CONSTRAINT PK\_Orderline PRIMARY KEY(FoodID, OrderID),

)

CREATE TABLE Payment

(

PaymentID INTEGER NOT NULL IDENTITY PRIMARY KEY,

OrderID INTEGER NOT NULL FOREIGN KEY REFERENCES Orders(OrderID),

PaymentAmount DECIMAL(5,2) NOT NULL,

PaymentDate DATETIME NOT NULL DEFAULT GETDATE(),

)

CREATE TABLE Delivery

(

DeliveryID INTEGER NOT NULL IDENTITY PRIMARY KEY,

OrderId INTEGER NOT NULL FOREIGN KEY REFERENCES Orders(OrderID),

DriverID INTEGER NOT NULL FOREIGN KEY REFERENCES Driver(DriverID),

DeliveryStatus VARCHAR(10) NOT NULL,

CONSTRAINT check\_DeliveryStatus CHECK(DeliveryStatus='On the way' or DeliveryStatus='Arrived')

)

SELECT \* FROM Food

SELECT \* FROM Customer

SELECT \* FROM Restaurant

SELECT \* FROM Driver

SELECT \* FROM Car

SELECT \* FROM Orders

SELECT \* FROM Orderline

SELECT \* FROM Payment

SELECT \* FROM Delivery

INSERT INTO Customer (FirstName,MiddleName,LastName,PhoneNumber,Address1,Address2,City,CState,ZipCode)

VALUES('Daenerys','E','Targaryen','315-455-3928','505','Walnut Ave.','Syracuse','NY','13210')

INSERT INTO Customer (FirstName,MiddleName,LastName,PhoneNumber,Address1,Address2,City,CState,ZipCode)

VALUES('Jon','K','Snow','315-455-5465','325','E Genesee St.','Syracuse','NY','13244')

INSERT INTO Customer (FirstName,LastName,PhoneNumber,Address1,Address2,City,CState,ZipCode)

VALUES('Yuanliang','Song','315-455-0530','505','Walnut Ave.','Syracuse','NY','13210')

INSERT INTO Customer (FirstName,MiddleName,LastName,PhoneNumber,Address1,Address2,City,CState,ZipCode)

VALUES('Lily','M','Turner','315-455-3928','3','Ostrom Ave','Syracuse','NY','13225')

INSERT INTO Customer (FirstName,MiddleName,LastName,PhoneNumber,Address1,Address2,City,CState,ZipCode)

VALUES('Olivia','A','Wallace','315-555-8888','4248','Nottingham Rd','Syracuse','NY','13244')

INSERT INTO Customer (FirstName,MiddleName,LastName,PhoneNumber,Address1,Address2,City,CState,ZipCode)

VALUES('Heather','S','Cameron','315-455-9999','410','Comstock Ave.','Syracuse','NY','13210')

INSERT INTO Customer (FirstName,MiddleName,LastName,PhoneNumber,Address1,Address2,City,CState,ZipCode)

VALUES('Sebastian','','Chapman','315-455-7777','17','James St.','Syracuse','NY','13210')

INSERT INTO Customer (FirstName,MiddleName,LastName,PhoneNumber,Address1,Address2,City,CState,ZipCode)

VALUES('Nathan','N','Kerr','315-555-5555','112','Lafayette Rd.','Syracuse','NY','13205')

INSERT INTO Restaurant (RestaurantName,PhoneNumber,Address1,Address2,City,CState,ZipCode,RestaurantStatus)

VALUES('Bleu Monkey Cafe','315-741-3342','163','Marshall St','Syracuse','NY','13210','Open')

INSERT INTO Restaurant (RestaurantName,PhoneNumber,Address1,Address2,City,CState,ZipCode,RestaurantStatus)

VALUES('Johnny Rockets','315-474-7406','1','Destiny USA','Syracuse','NY','13204','Open')

INSERT INTO Restaurant (RestaurantName,PhoneNumber,Address1,Address2,City,CState,ZipCode,RestaurantStatus)

VALUES('Antonios Liverpool Pizzeria','315-123-1111','7608','Oswego','Liverpool','NY','13090','Open')

INSERT INTO Restaurant (RestaurantName,PhoneNumber,Address1,Address2,City,CState,ZipCode,RestaurantStatus)

VALUES('Taco Bell','315-123-3333','8095','Oswego Rd','Liverpool','NY','13090','Open')

INSERT INTO Restaurant (RestaurantName,PhoneNumber,Address1,Address2,City,CState,ZipCode,RestaurantStatus)

VALUES('Number 1 Kitchen','315-123-2222','317','Nottingham Rd','Syracuse','NY','13210','Open')

INSERT INTO Restaurant (RestaurantName,PhoneNumber,Address1,Address2,City,CState,ZipCode)

VALUES('Oishi Sushi','315-123-4444','710','S Crouse Ave','Syracuse','NY','13210')

INSERT INTO Restaurant (RestaurantName,PhoneNumber,Address1,Address2,City,CState,ZipCode,RestaurantStatus)

VALUES('New York Style Diner','315-123-5555','549','E Brighton Ave','Syracuse','NY','13212','Open')

INSERT INTO Food(RestaurantID,FoodName,FoodDescription)

VALUES(1,'Beef Bowl','Served with corn,seaweed and scallions.')

INSERT INTO Food(RestaurantID,FoodName,FoodDescription)

VALUES(1,'Grilled Chicken with Teriyaki Sauce','Served with rice.')

INSERT INTO Food(RestaurantID,FoodName,FoodDescription)

VALUES(1,'Lizard Roll','Tempura shrimp and avocado topped with salmon, avocado, teriyaki and spicy sauce.')

INSERT INTO Food(RestaurantID,FoodName)

VALUES(2,'Street Fries')

INSERT INTO Food(RestaurantID,FoodName,FoodDescription)

VALUES(2,'Garden Salad','Crisp seasonal greens topped with fresh diced tomatoes, shredded Wisconsin cheddar cheese and choice of dressing.')

INSERT INTO Food(RestaurantID,FoodName,FoodDescription)

VALUES(2,'The Original Burger','A certified Angus beff patty with crisp shredded lettuce, fresh tomato, chopped onion')

INSERT INTO Food(RestaurantID,FoodName,Price)

VALUES(3,'10 Traditional Wings',9.99)

INSERT INTO Food(RestaurantID,FoodName,FoodDescription,Price)

VALUES(3,'10 Boneless Wings','Served with 1 bleu cheese',8.49)

INSERT INTO Food(RestaurantID,FoodName,FoodDescription,Price)

VALUES(4,'General Tsos Chicken','Spicy.',8.5)

INSERT INTO Food(RestaurantID,FoodName,Price)

VALUES(4,'Fried Dumplings',6.8)

INSERT INTO Food(RestaurantID,FoodName,Price)

VALUES(4,'Sweet and Sour Chicken Combo',8.5)

INSERT INTO Food(RestaurantID,FoodName,FoodDescription,Price)

VALUES(5,'Chalipa Cravings Box','Includes a Chalupa Supreme, Beefy 5-Layer Burrito, Crunchy Taco, Cinnamon Twists, and a Drink.',5)

INSERT INTO Food(RestaurantID,FoodName,FoodDescription,Price)

VALUES(5,'Fiesta Taco Salad Combo','Served with adrink and 3 tacos',5)

INSERT INTO Food(RestaurantID,FoodName,FoodDescription,Price)

VALUES(5,'XXL Grilled Stuft Burrito Combo','Served with a drink and a crunchy taco.',7.99)

INSERT INTO Food(RestaurantID,FoodName,FoodDescription,Price)

VALUES(6,'Dinner Box','Your choice of entree. Served with rice, gyoza, California roll and your choice of soup or salad.',11.95)

INSERT INTO Food(RestaurantID,FoodName,FoodDescription,Price)

VALUES(6,'Chicken Teriyaki','Served with rice and your choice of miso soup or salad.',11.5)

INSERT INTO Food(RestaurantID,FoodName,Price)

VALUES(6,'Any 3 Roll',15.95)

INSERT INTO Car (PlateNumber,Maker, Model, Color)

VALUES('XYZ-1234','Honda','Civic','Black')

INSERT INTO Car (PlateNumber,Maker, Model, Color)

VALUES('UVW-1234','BMW','M5','Blue')

INSERT INTO Car (PlateNumber,Maker, Model, Color)

VALUES('ABC-1234','Jeep','Cherokee','Black')

INSERT INTO Car (PlateNumber,Maker, Model, Color)

VALUES('DEF-1234','Honda','Accord','Black')

INSERT INTO Driver (FirstName,LastName,PhoneNumber,PlateNumber)

VALUES('Dorothy','Paige','315-555-0126','XYZ-1234')

INSERT INTO Driver (FirstName,LastName,PhoneNumber,PlateNumber)

VALUES('Amy','May','315-555-5555','UVW-1234')

INSERT INTO Driver (FirstName,LastName,PhoneNumber,PlateNumber)

VALUES('Charles','Duncan','315-444-5555','ABC-1234')

INSERT INTO Driver (FirstName,LastName,PhoneNumber,PlateNumber)

VALUES('Victor','Miller','315-333-5565','DEF-1234')

INSERT INTO Orders(CustomerID,OrderDate,PlacedTime,OrderStatus)

VALUES(1,'2019-04-16','12:15:24.000','Delivered')

INSERT INTO Orders(CustomerID,OrderStatus)

VALUES(2,'Delivering')

INSERT INTO Orders(CustomerID,OrderDate,PlacedTime,OrderStatus)

VALUES(3,'2019-04-15','17:15:00.000','Delivered')

INSERT INTO Orders(CustomerID,OrderDate,PlacedTime,OrderStatus)

VALUES(4,'2019-04-14','18:20:00.000','Delivered')

INSERT INTO Orders(CustomerID,OrderDate,PlacedTime,OrderStatus)

VALUES(5,'2019-04-13','11:00:00.000','Delivered')

INSERT INTO Orders(CustomerID,OrderDate,PlacedTime,OrderStatus)

VALUES(6,'2019-04-14','15:10:00.000','Delivered')

INSERT INTO Orders(CustomerID,OrderDate,PlacedTime,OrderStatus)

VALUES(7,'2019-04-14','10:21:00.000','Delivered')

INSERT INTO Orders(CustomerID,OrderDate,PlacedTime,OrderStatus)

VALUES(8,'2019-04-14','11:31:00.000','Delivered')

SELECT \* FROM Orderline

INSERT INTO Orderline(FoodID,OrderID,Quantity)

VALUES(1,1,1)

INSERT INTO Orderline(FoodID,OrderID,Quantity)

VALUES(2,1,2)

INSERT INTO Orderline(FoodID,OrderID,Quantity)

VALUES(15,1,1)

INSERT INTO Orderline(FoodID,OrderID,Quantity)

VALUES(4,1,1)

INSERT INTO Orderline(FoodID,OrderID,Quantity)

VALUES(10,1,1)

INSERT INTO Orderline(FoodID,OrderID,Quantity)

VALUES(3,2,1)

INSERT INTO Orderline(FoodID,OrderID,Quantity)

VALUES(6,2,1)

INSERT INTO Orderline(FoodID,OrderID,Quantity)

VALUES(12,2,1)

INSERT INTO Orderline(FoodID,OrderID,Quantity)

VALUES(10,3,1)

INSERT INTO Orderline(FoodID,OrderID,Quantity)

VALUES(10,4,1)

INSERT INTO Orderline(FoodID,OrderID,Quantity)

VALUES(11,5,1)

INSERT INTO Orderline(FoodID,OrderID,Quantity)

VALUES(7,5,1)

INSERT INTO Orderline(FoodID,OrderID,Quantity)

VALUES(6,5,1)

INSERT INTO Orderline(FoodID,OrderID,Quantity)

VALUES(8,6,1)

INSERT INTO Orderline(FoodID,OrderID,Quantity)

VALUES(9,6,1)

INSERT INTO Orderline(FoodID,OrderID,Quantity)

VALUES(4,6,1)

INSERT INTO Orderline(FoodID,OrderID,Quantity)

VALUES(10,7,1)

INSERT INTO Orderline(FoodID,OrderID,Quantity)

VALUES(5,7,1)

INSERT INTO Orderline(FoodID,OrderID,Quantity)

VALUES(3,8,1)

INSERT INTO Orderline(FoodID,OrderID,Quantity)

VALUES(8,8,1)

INSERT INTO Orderline(FoodID,OrderID,Quantity)

VALUES(9,8,1)

SELECT \* FROM Delivery

INSERT INTO Delivery(OrderId, DriverID,DeliveryStatus)

VALUES (1,1,'Arrived')

INSERT INTO Delivery(OrderId, DriverID,DeliveryStatus)

VALUES (2,2,'On the way')

INSERT INTO Delivery(OrderId, DriverID,DeliveryStatus)

VALUES (3,3,'Arrived')

INSERT INTO Delivery(OrderId, DriverID,DeliveryStatus)

VALUES (4,4,'Arrived')

INSERT INTO Delivery(OrderId, DriverID,DeliveryStatus)

VALUES (5,1,'Arrived')

INSERT INTO Delivery(OrderId, DriverID,DeliveryStatus)

VALUES (6,2,'Arrived')

INSERT INTO Delivery(OrderId, DriverID,DeliveryStatus)

VALUES (7,3,'Arrived')

INSERT INTO Delivery(OrderId, DriverID,DeliveryStatus)

VALUES (8,4,'Arrived')

SELECT \* FROM Payment

INSERT INTO Payment (OrderID,PaymentAmount )

VALUES(1,(select Sum(Ol.Quantity\* F.Price)\*1.2 From Orders O join Orderline OL on O.OrderID=OL.OrderID join Food F on OL.FoodID=F.FoodID

where O.OrderID=1))

INSERT INTO Payment (OrderID,PaymentAmount )

VALUES(2,(select Sum(Ol.Quantity\* F.Price)\*1.2 From Orders O join Orderline OL on O.OrderID=OL.OrderID join Food F on OL.FoodID=F.FoodID

where O.OrderID=2))

INSERT INTO Payment (OrderID,PaymentAmount )

VALUES(3,(select Sum(Ol.Quantity\* F.Price)\*1.2 From Orders O join Orderline OL on O.OrderID=OL.OrderID join Food F on OL.FoodID=F.FoodID

where O.OrderID=3))

INSERT INTO Payment (OrderID,PaymentAmount )

VALUES(4,(select Sum(Ol.Quantity\* F.Price)\*1.2 From Orders O join Orderline OL on O.OrderID=OL.OrderID join Food F on OL.FoodID=F.FoodID

where O.OrderID=4))

INSERT INTO Payment (OrderID,PaymentAmount )

VALUES(5,(select Sum(Ol.Quantity\* F.Price)\*1.2 From Orders O join Orderline OL on O.OrderID=OL.OrderID join Food F on OL.FoodID=F.FoodID

where O.OrderID=5))

INSERT INTO Payment (OrderID,PaymentAmount )

VALUES(6,(select Sum(Ol.Quantity\* F.Price)\*1.2 From Orders O join Orderline OL on O.OrderID=OL.OrderID join Food F on OL.FoodID=F.FoodID

where O.OrderID=6))

INSERT INTO Payment (OrderID,PaymentAmount )

VALUES(7,(select Sum(Ol.Quantity\* F.Price)\*1.2 From Orders O join Orderline OL on O.OrderID=OL.OrderID join Food F on OL.FoodID=F.FoodID

where O.OrderID=7))

INSERT INTO Payment (OrderID,PaymentAmount )

VALUES(8,(select Sum(Ol.Quantity\* F.Price)\*1.2 From Orders O join Orderline OL on O.OrderID=OL.OrderID join Food F on OL.FoodID=F.FoodID

where O.OrderID=8))

-- How many orders does each restaurant have?

CREATE VIEW NumOfOrdersForRest AS

SELECT R.RestaurantID, R.RestaurantName,count(DISTINCT O.OrderID ) NumOfOrders

FROM Orders O join Orderline OL

on O.OrderID= OL.OrderID join Food F

on OL.FoodID=F.FoodID join Restaurant R

on F.RestaurantID=R.RestaurantID

Group by RestaurantName,R.RestaurantID

SELECT \* FROM NumOfOrdersForRest

-- Which customer ordered food from more than one restaurant in a single order

CREATE VIEW CusOrderedinManyRest AS

SELECT C.CustomerID,C.FirstName,C.LastName,O.OrderID,count(DISTINCT R.RestaurantID) NumberOfRest FROM Customer C join Orders O

on C.CustomerID=O.CustomerID join Orderline OL

on O.OrderID= OL.OrderID join Food F

on OL.FoodID=F.FoodID join Restaurant R

on F.RestaurantID=R.RestaurantID

group by C.CustomerID,C.FirstName,C.LastName,O.OrderID

Having count(DISTINCT R.RestaurantID)!=1

SELECT \* FROM CusOrderedinManyRest

--Which restaurants are top 3 popular

SELECT TOP 3 R.RestaurantID, R.RestaurantName, count(OrderID) TimesOrdered

FROM Orderline OL join Food F on OL.FoodID=F.FoodID

join Restaurant R on F.RestaurantID=R.RestaurantID

group by R.RestaurantID,R.RestaurantName

Order by TimesOrdered DESC

--Which food is most popular food

SELECT TOP 1 OL.FoodID, FoodName, Count(OrderID) TimesOrdered,Sum(Quantity) QuantityOrderd

FROM Orderline OL join Food F on OL.FoodID = F.FoodID

Group by OL.FoodID,FoodName

Order by QuantityOrderd DESC

-- What is the average payment of orders for restaurant

CREATE VIEW AvePay AS

SELECT R.RestaurantID,RestaurantName,F.FoodID,O.OrderID,Quantity,Price FROM Payment P join Orders O on P.OrderID= O.OrderID

join Orderline OL on O.OrderID=OL.OrderID

join Food F on OL.FoodID = F.FoodID

join Restaurant R on F.RestaurantID=R.RestaurantID

SELECT \* FROM AvePay

SELECT RestaurantID,RestaurantName, sum(Quantity\*Price)/Count(DISTINCT OrderID) AverangePayment FROM AvePay

group by RestaurantID,RestaurantName

-- What is the delivery status of orders placed by customers

CREATE VIEW DelStatForCus AS

SELECT C.CustomerID,C.FirstName,C.LastName, Dr.DriverID,Dr.FirstName as DriverFirstName,Dr.PlateNumber,D.DeliveryStatus FROM Customer C join Orders O on C.CustomerID=O.CustomerID

join Delivery D on O.OrderID=D.DeliveryID

join Driver Dr on D.DriverID=Dr.DriverID

SELECT \* FROM DelStatForCus

Order by CustomerID

-- What is the order status of orders placed by customers

CREATE VIEW OrderStatus AS

SELECT C.CustomerID, C.FirstName,C.LastName,O.OrderID,O.OrderStatus FROM Customer C join Orders O on C.CustomerID= O.CustomerID

SELECT \* FROM OrderStatus