King Saud University
College of Computer and Information Sciences
Department of Information Technology

IT222: Database Principles
1st Semester 1445 H

THE CHEFZ

Phase # 3

Section #	NAME	ID
View Name: Customer view		
#56390	Sarah Alsaleh	443201066
#56390	Ghada Binoun	443200646
#56390	Raghad Alzkeri	443201025
#56390	Mashael Al-Jaad	443203034
#56390	Batool Alkhuraim	443200604

Supervised By: TA.Noura Alsaud
TA.Abeer Aladrees.

Project Description:

An application for meal delivery is called The Chefz. You may explore a variety of eateries, go through their menus, and order your favorite foods with just a few taps on your phone. With the app's help, you can track your order in real-time and find out when it will reach your door. The Chefz application's database development is the goal of this project. The CHEFZ Database's goal is to maintain the data utilized and generated to enable.

View Description:

Our database view is targeted toward the customer where they can view restaurants, bakeries and more. They can also place home delivery or store pickup orders and make reservations for a variety of fine dining restaurants.

Data Requirements:

- Order: An order is a request from the customer to order from a variety of options. The order has a Date, Status, Time, Payment method, total price, and a unique OrderID. Each order has at least one product.

- Product: A product could be a variety of things such as (Food, Drinks, Flowers,…,etc.) that gets delivered to the customer. Each Product has a name, price, rating, description (preparation time, serve), restaurant name, and a unique productID.

- Customer: A Customer is the person who places an order. The customer has a unique PhoneNumber, Name, one or many Addresses. Each customer has zero or many orders.

- Delivery carrier: A delivery carrier is the person responsible for delivering the orders placed by the customer. The carrier has a FName, LName, and a unique phoneNumber. Each delivery carrier can deliver 1 to many orders.
-
- Restaurant: A restaurant is a business that prepares and serves food and drinks to customers . The restaurant has a location, type, rating and a unique Name. Each restaurant has one to many products.

Transaction Requirements:

Data Entry:

1. Customer enters their Phone number.

2. Customer enters their Address details.

Data update/deletion:

1. Update personal info (name, number).

2. Update / delete address.

3. Update / delete order.

Data Queries:

Lists all past orders.

Display the current state of an order.

List all registered addresses in the application.

List all past orders in a month.

List products with price range.

List restaurant based on ratings.

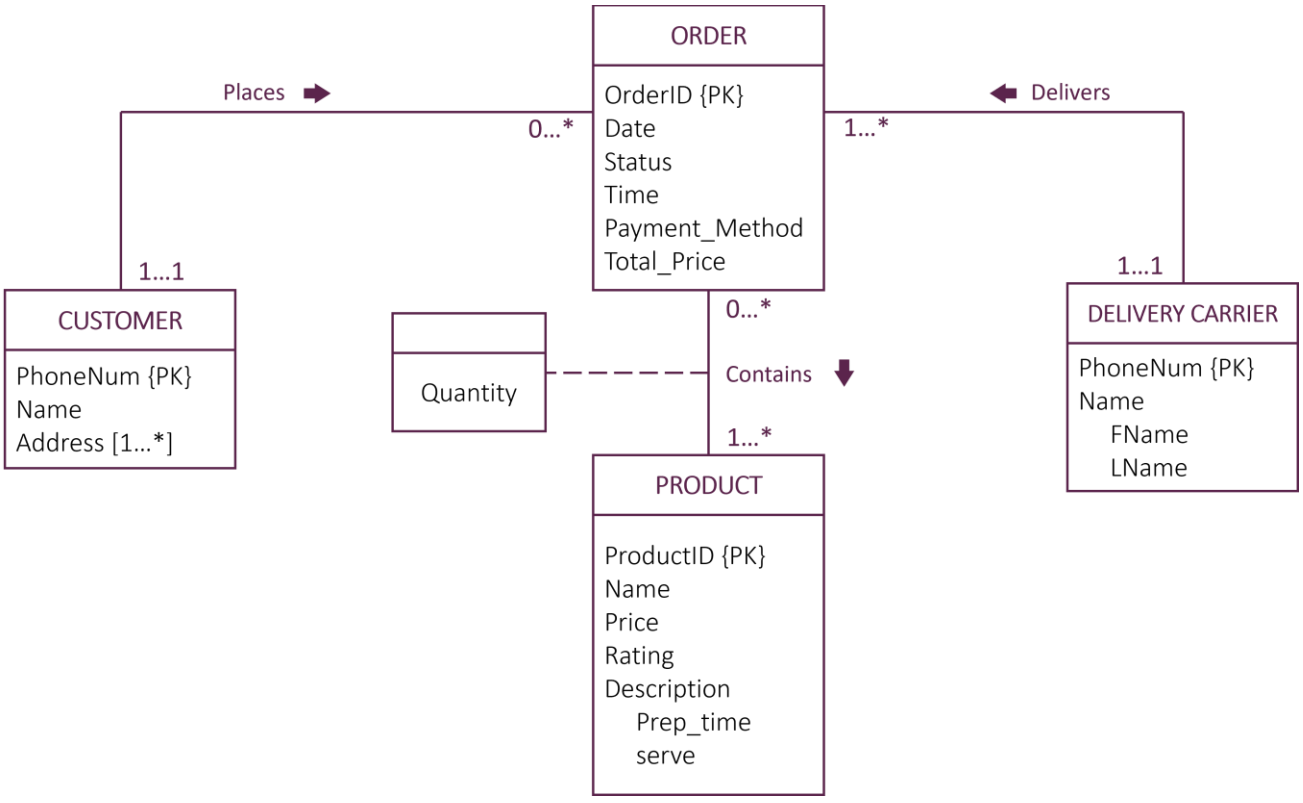
Display Items that are available now.

Finding restaurants around 20 km away from the place being delivered to.

Display restaurant based on type.

List the allowed payment methods.

Global enhanced entity relationship diagram (EER):



Relational Schema:

PRODUCT (ProductID, Name, Price, Rating, Prep_time, serve, NameR)
Primary key ProductID
Foreign key NameR references Restaurant (Name)

ORDER_ (OrderID, Date, Status, Time, Payment_Method, Total_Price, PhoneD, PhoneC)
Primary key OrderID
Foreign key PhoneD references Delivery Carrier (PhoneNum)
Foreign key PhoneC references Customer (PhoneNum)

Contains_ (ProductID, OrderID, Quantity)
Primary key ProductID, OrderID
Foreign key OrderID references Order (OrderID)
Foreign key ProductID references Product (ProductID)

DELIVERY CARRIER (PhoneNum, FName, LName)
Primary key PhoneNum

RESTAUANT (Name, Location, Type, Rating)
Primary key Name

CUSTOMER (PhoneNum, Name)
Primary key PhoneNum

Customer_Address (Address, PhoneNum)
Primary key PhoneNum, Address

Data Dictionary showing description of all entities:

Entity Name	Description	Occurrence
Order	An order is a request from the customer to order from a variety of options.	Each order has at least one product.
Product	A product could be a variety of things such as (Food, Drinks, Flowers,…,etc.) that gets delivered to the customer.	None or many products can be in an order.
Customer	A Customer is the person who places an order.	A customer has zero or many orders.
Delivery carrier	A delivery carrier is the person responsible for delivering the orders placed by the customer.	Each delivery carrier can deliver one to many orders.
Restaurant	A Restaurant is a business that prepares and serves food and drinks to customers.	Each restaurant has one to many products.

Data Dictionary showing description of all relationships:

Entity Name	Multiplicity	Relationship	Entity Name	Multiplicity
Order	0…*	Contains	Product	1…*
Customer	1..1	Places	Order	0..*
Restaurant	1…1	Has	Product	1…*
Delivery carrier	1…1	Delivers	Order	1…*

Data Dictionary showing description of all attributes:

Entity Name	Attribute	Description	Data Type	Length	Nulls	Multi-Valued	Default Value	Range	PK
Order	Date	Date of the order	Varchar	20					
	Time	Time of the order	Varchar	5					
	Status	Order's status	Varchar	30					
	Payment method	Order's payment method	Varchar	15				Apple Pay	
								Mada	
								Credit card	
								tamara	
	Total_price	Order's total price	Decimal	6.5					
	OrderID	Order's ID	Varchar	10					Y
Product	Price	The price of product	Decimal	5					
	Rating	Product's rating	Varchar	5					
	Description								
	Prep time	How long will it take	Varchar	5					
	serve	How many people it serves	Varchar	5					

	Name	Name of the product	Varchar	16					
	ProductID	ID uniquely identifies the product	Varchar	10					Y
Customer	PhoneNum	Customer phone number	Varchar	10					Y
	Name	Customer's name	Varchar	16					
	Addresses	Customers address	Varchar	25		Y			
Delivery carrier	Name								
	FName	First name of the carrier	Varchar	16					
	LName	Last name of the carrier	Varchar	16					
	PhoneNum	Carrier's phone number	Varchar	10					Y
Restaurant	Name	Restaurant's name	Varchar	20					Y
	Location	Restaurant's location	Varchar	25					
	Type	The type of cuisine	Varchar	16					
	Rating	Restaurant's rating	Varchar	5					
	Quantity	Product quantity	Integer				0		

DB tables creation commands:

```
CREATE TABLE RESTAURANT(
Name VARCHAR(20) NOT NULL,
Location VARCHAR(25) NOT NULL,
Type VARCHAR(16) NOT NULL,
Rating VARCHAR(5) NOT NULL,
PRIMARY KEY(Name)
);

CREATE TABLE CUSTOMER(
PhoneNum VARCHAR(10) NOT NULL,
Name VARCHAR(16) NOT NULL,
PRIMARY KEY(PhoneNum)
);

CREATE TABLE Customer_Address(
Address VARCHAR(25) NOT NULL,
PhoneNum VARCHAR(10) NOT NULL,
PRIMARY KEY(PhoneNum, Address)
);

CREATE TABLE DELIVERY_CARRIER(
PhoneNum VARCHAR(10) NOT NULL,
FName VARCHAR(16) NOT NULL,
LName VARCHAR(16) NOT NULL,
PRIMARY KEY(PhoneNum)
);

CREATE TABLE PRODUCT(
ProudctID VARCHAR(10) NOT NULL,
Name VARCHAR(20) NOT NULL,
Price DECIMAL(5) NOT NULL,
Rating VARCHAR(5) NOT NULL,
Prep_time VARCHAR(5) NOT NULL,
serve VARCHAR(5) NOT NULL,
NameR VARCHAR(20) NOT NULL,
PRIMARY KEY(ProudctID),
FOREIGN KEY(NameR) references RESTAURANT(Name)
);
```

```
CREATE TABLE ORDER_(
OrderID VARCHAR(10) NOT NULL,
Date VARCHAR(20) NOT NULL,
Status VARCHAR(30) NOT NULL,
Time VARCHAR(5) NOT NULL,
Payment_Method VARCHAR(15) NOT NULL,
Total_Price DECIMAL(6.5) NOT NULL,
PhoneD VARCHAR(10) NOT NULL,
PhoneC VARCHAR(10) NOT NULL,
PRIMARY KEY(OrderID),
FOREIGN KEY(PhoneD) references DELIVERY_CARRIER(PhoneNum),
FOREIGN KEY(PhoneC) references CUSTOMER(PhoneNum)
);

CREATE TABLE Contains_(
ProudctID VARCHAR(10) NOT NULL,
OrderID VARCHAR(10) NOT NULL,
Quantity int DEFAULT 0 NOT NULL,
PRIMARY KEY(ProudctID, OrderID),
FOREIGN KEY(ProudctID) references PRODUCT(ProudctID),
FOREIGN KEY(OrderID) references ORDER_(OrderID)
);
```

Data insertion commands:

```
INSERT INTO RESTAURANT VALUES("Burger King", "Al nada", "Italian", "4");
INSERT INTO RESTAURANT VALUES("Sloppy", "Al Nakheel", "Fast food", "5");

INSERT INTO PRODUCT VALUES ("T001", "FRENCH TOAST", 25.5, "5", "1Hour", "Two", "Burger
King");
INSERT INTO PRODUCT VALUES ("T002", "Burger", 27.0, "4", "35Min", "One", "Sloppy");

INSERT INTO CUSTOMER VALUES("0592797675", "Sara");
INSERT INTO CUSTOMER VALUES("0592797078", "Masheal");

INSERT INTO Customer_Address VALUES("Al falah 28", "0592797675");
INSERT INTO Customer_Address VALUES("Hitten 7", "0592797078");

INSERT INTO DELIVERY_CARRIER VALUES("0592797078", "Batool", "Nasser");
INSERT INTO DELIVERY_CARRIER VALUES("0539000234", "Raghad", "Mohammed");

INSERT INTO ORDER_ VALUES ("8600787", "2023-09-05", "Delivered", "7:45", "Apple pay", 320,
"0592797078", "0592797078");
INSERT INTO ORDER_ VALUES ("8600654", "2023-11-09", "ACCEPTED", "4:00", "MADA", 150,
"0539000234", "0592797078");

INSERT INTO Contains_ VALUES("T001", "8600787", 5);
INSERT INTO Contains_ VALUES("T002", "8600654", 0);
```

Data Queries commands and outputs:

1- Display RESTAURANT based on type.

*SELECT **
FROM RESTAURANT
WHERE Type = "Italian";

Schema SQL

```
CREATE TABLE RESTAURANT(
Name VARCHAR(20) NOT NULL,
Location VARCHAR(25) NOT NULL,
Type VARCHAR(16) NOT NULL,
Rating VARCHAR(5) NOT NULL,
PRIMARY KEY(Name)
);

CREATE TABLE CUSTOMER(
PhoneNum VARCHAR(10) NOT NULL,
Name VARCHAR(16) NOT NULL,
PRIMARY KEY(PhoneNum)
);
```

Text to DDL

Query SQL

```
-- 1- Display RESTAURANT based on type.
SELECT *
FROM RESTAURANT
WHERE Type = "Italian";
```

Results

Copy as Markdown

Query #1 Execution time: 1ms

Name	Location	Type	Rating
Burger King	Al nada	Italian	4

2- List all past Orders.

*SELECT **
*FROM ORDER_*
WHERE Status = "Delivered" AND Date < CURRENT_DATE;

Schema SQL

```
CREATE TABLE RESTAURANT(
Name VARCHAR(20) NOT NULL,
Location VARCHAR(25) NOT NULL,
Type VARCHAR(16) NOT NULL,
Rating VARCHAR(5) NOT NULL,
PRIMARY KEY(Name)
);

CREATE TABLE CUSTOMER(
PhoneNum VARCHAR(10) NOT NULL,
Name VARCHAR(16) NOT NULL,
PRIMARY KEY(PhoneNum)
);
```

Text to DDL

Query SQL

```
-- 2- List all past Orders.
SELECT *
FROM ORDER_
WHERE Status = "Delivered" AND Date < CURRENT_DATE;
```

Results

Copy as Markdown

Query #1 Execution time: 0ms

OrderID	Date	Status	Time	Payment_Method	Total_Price	PhoneD	PhoneC
8600787	2023-09-05	Delivered	7:45	Apple pay	320	0592797078	0592797078

3- List restaurant based on ratings.

*SELECT **
FROM RESTAURANT
WHERE Rating = "5";

Schema SQL

```
CREATE TABLE RESTAURANT(
Name VARCHAR(20) NOT NULL,
Location VARCHAR(25) NOT NULL,
Type VARCHAR(16) NOT NULL,
Rating VARCHAR(5) NOT NULL,
PRIMARY KEY(Name)
);

CREATE TABLE CUSTOMER(
PhoneNum VARCHAR(10) NOT NULL,
Name VARCHAR(16) NOT NULL,
PRIMARY KEY(PhoneNum)
);
```

Text to DDL

Query SQL

```
-- 3- List restaurant based on ratings.
SELECT *
FROM RESTAURANT
WHERE Rating = "5";

```

Results

Copy as Markdown

Query #1 Execution time: 1ms

Name	Location	Type	Rating
Sloppy	Al Nakheel	Fast food	5

4- List the allowed payment methods.

SELECT DISTINCT Payment_Method
FROM ORDER_;

Schema SQL

```
CREATE TABLE RESTAURANT(
Name VARCHAR(20) NOT NULL,
Location VARCHAR(25) NOT NULL,
Type VARCHAR(16) NOT NULL,
Rating VARCHAR(5) NOT NULL,
PRIMARY KEY(Name)
);

CREATE TABLE CUSTOMER(
PhoneNum VARCHAR(10) NOT NULL,
Name VARCHAR(16) NOT NULL,
PRIMARY KEY(PhoneNum)
);
```

Text to DDL

Query SQL

```
-- 4- List the allowed payment methods.
SELECT DISTINCT Payment_Method
FROM ORDER_;

```

Results

Copy as Markdown

Query #1 Execution time: 1ms

Payment_Method
MADA
Apple pay

5- Display the current state of an order.

SELECT OrderID, Status
*FROM ORDER_*
WHERE OrderID = "8600654";

Schema SQL

```
CREATE TABLE RESTAURANT(
Name VARCHAR(20) NOT NULL,
Location VARCHAR(25) NOT NULL,
Type VARCHAR(16) NOT NULL,
Rating VARCHAR(5) NOT NULL,
PRIMARY KEY(Name)
);

CREATE TABLE CUSTOMER(
PhoneNum VARCHAR(10) NOT NULL,
Name VARCHAR(16) NOT NULL,
PRIMARY KEY(PhoneNum)
);
```

Text to DDL

Query SQL

```
-- 5- Display the current state of an order.
SELECT OrderID, Status
FROM ORDER_
WHERE OrderID = "8600654";

```

Results

Copy as Markdown

Query #1 Execution time: 0ms

OrderID	Status
8600654	ACCEPTED

6- List all registered addresses in the application.

SELECT DISTINCT Address
FROM Customer_Address;

Schema SQL

```
CREATE TABLE RESTAURANT(
Name VARCHAR(20) NOT NULL,
Location VARCHAR(25) NOT NULL,
Type VARCHAR(16) NOT NULL,
Rating VARCHAR(5) NOT NULL,
PRIMARY KEY(Name)
);

CREATE TABLE CUSTOMER(
PhoneNum VARCHAR(10) NOT NULL,
Name VARCHAR(16) NOT NULL,
PRIMARY KEY(PhoneNum)
);
```

Text to DDL

Query SQL

```
-- 6- List all registered addresses in the application.
SELECT DISTINCT Address
FROM Customer_Address;
```

Results

Copy as Markdown

Query #1 Execution time: 1ms

Address
Hitten 7
Al falah 28

7- List all past orders in a month.

*SELECT **
*FROM ORDER_*
WHERE MONTH(Date) = 9;

Schema SQL

```
CREATE TABLE RESTAURANT(
Name VARCHAR(20) NOT NULL,
Location VARCHAR(25) NOT NULL,
Type VARCHAR(16) NOT NULL,
Rating VARCHAR(5) NOT NULL,
PRIMARY KEY(Name)
);

CREATE TABLE CUSTOMER(
PhoneNum VARCHAR(10) NOT NULL,
Name VARCHAR(16) NOT NULL,
PRIMARY KEY(PhoneNum)
);
```

Text to DDL

Query SQL

```
-- 7- List all past orders in a month.
SELECT *
FROM ORDER_
WHERE MONTH(Date) = 9;
```

Results

Copy as Markdown

Query #1 Execution time: 0ms

OrderID	Date	Status	Time	Payment_Method	Total_Price	PhoneD	PhoneC
8600787	2023-09-05	Delivered	7:45	Apple pay	320	0592797078	0592797078

Work Distribution:

NAME	ID	Percentage	WORK
Sarah Alsaleh	443201066	20%	Project description View description Data requirement Transaction requirement EER model
Ghada Binoun	443200646	20%	Project description View description Data requirement Transaction requirement EER model
Raghad Alzkeri	443201025	20%	Project description View description Data requirement Transaction requirement EER model
Mashael Al-Jaad	443203034	20%	Project description View description Data requirement Transaction requirement EER model
Batool Alkhuraim	443200604	20%	Project description View description Data requirement Transaction requirement EER model