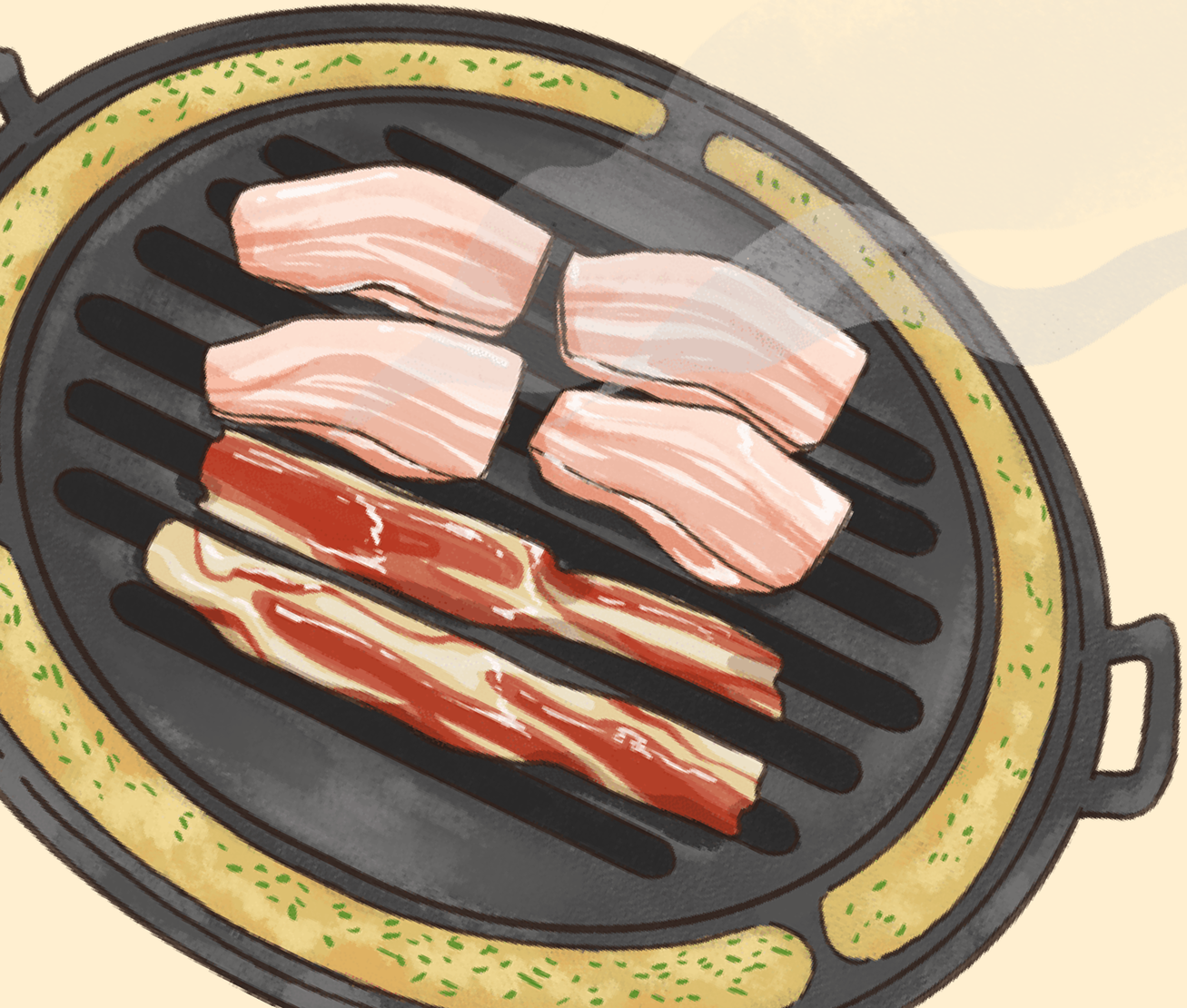
DATABASE PROJICT

GROUP 1

K-MEAL



Description

Our project revolves around a Korean restaurant system with many branches that serve many customers and enables them to order meals in simple ways and add feedback to their order. It also involves the manager adding and modifying employees, supervising orders, and adding or modifying new Ingredients in the menu.

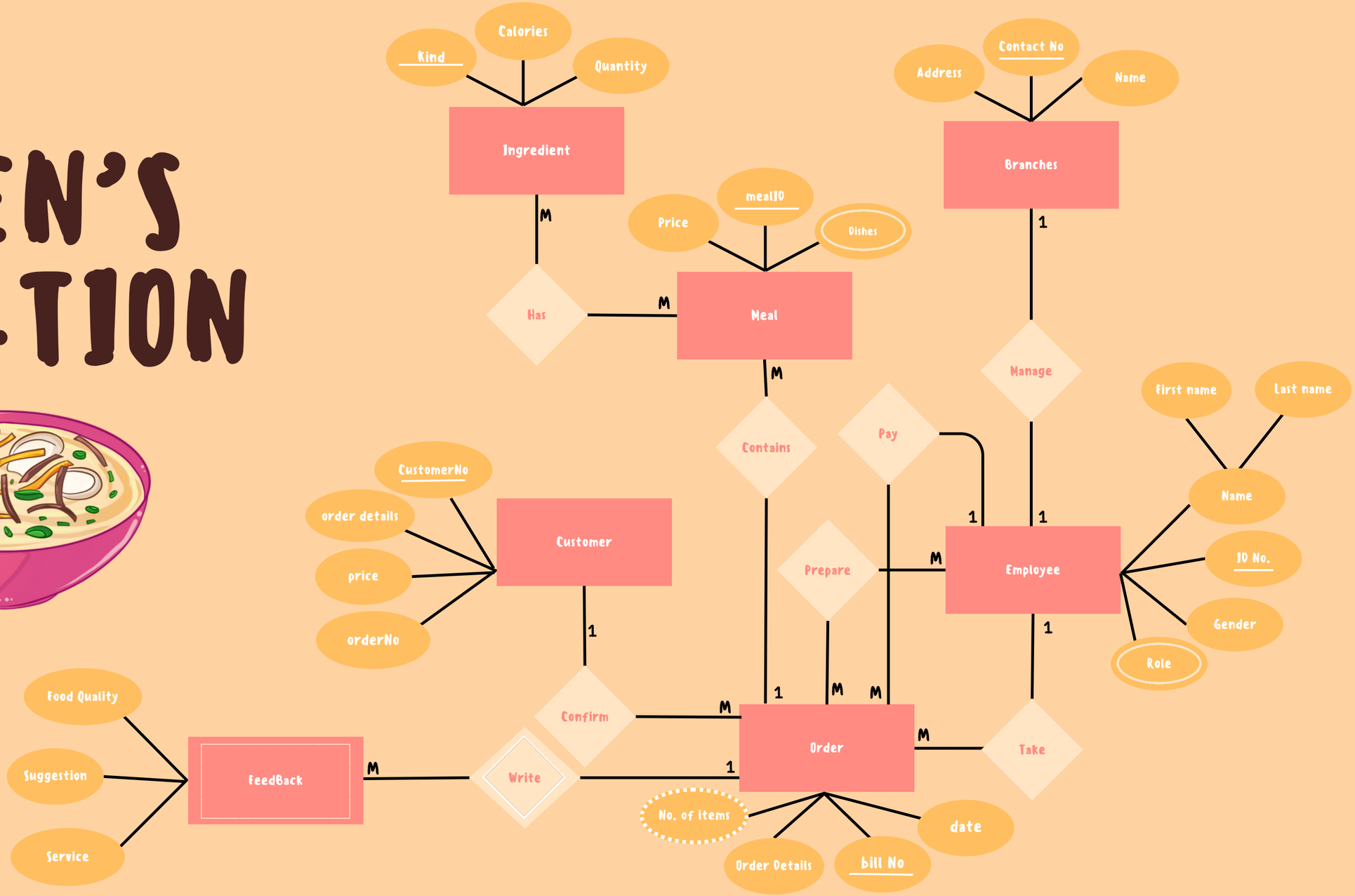
WORK REPORT

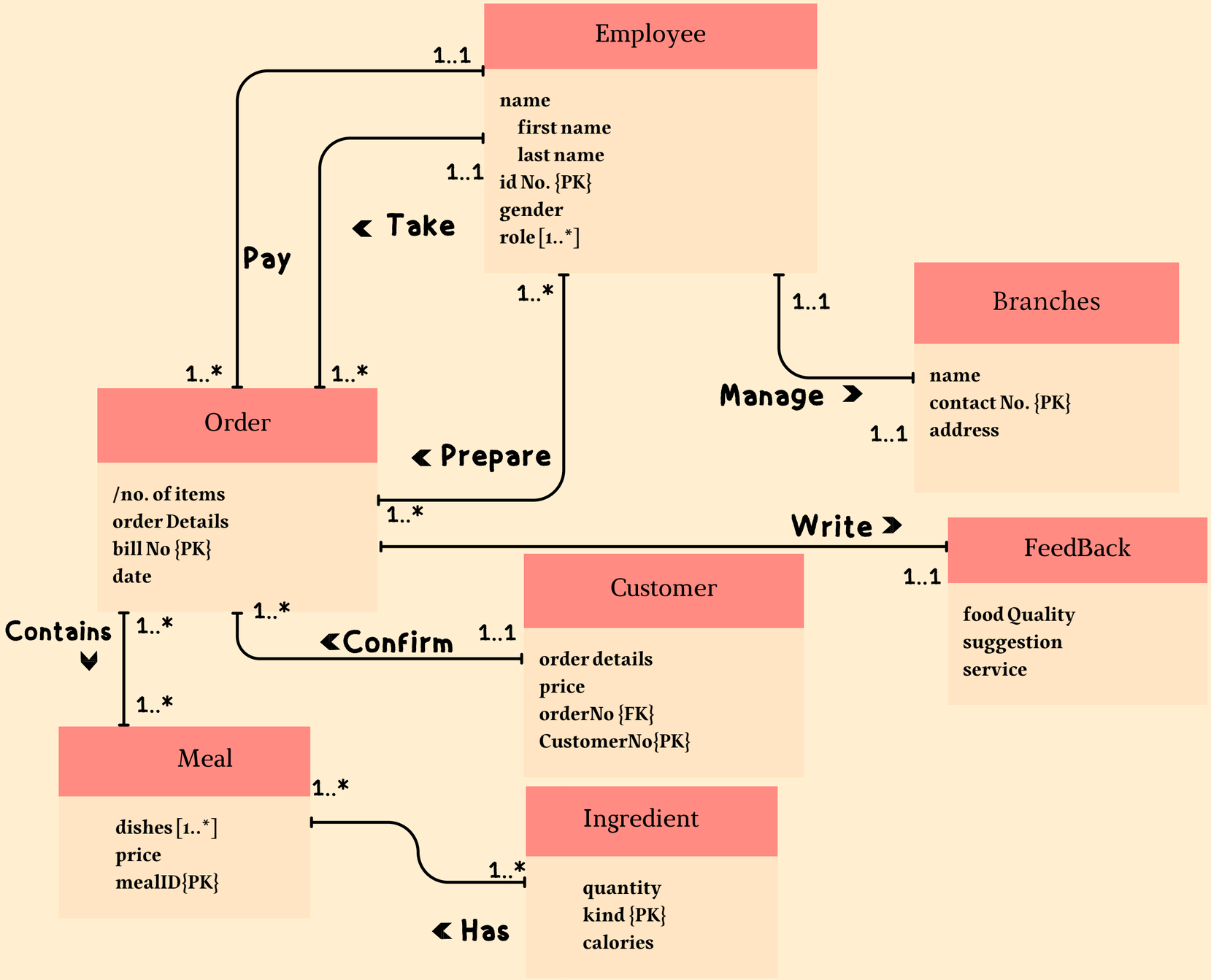
	Jana Al-Musbahie 443007060	Aghid Alansari 443009096	Noura Alsharif 443001417	Batool alsolami 443000985	Jana Mohamed 443000870	Roa'a bukhari 443000939
Business rule	✓		✓	✓	✓	✓
err	✓	✓	✓	✓		
uml		✓		✓	✓	✓
mapping	✓	✓			✓	
Normalizatio	✓		✓			
CREATE TABLE	✓	✓		✓	✓	✓
Insert Data	✓	✓		✓	✓	✓
Select, delete and update	✓	✓		✓	✓	✓
Where, Order and Group by ,Subqueries and join	✓	✓		✓	✓	✓

BUSINESS RULE

- The branch has a name, unique contact number, and address. is managed by one employee that has a unique ID number, a name consisting of a first name, a last name, and the gander.
- Each customer has order details, a unique customer number, order number, and a price.
- The Meal has dishes, prices, and a unique meal ID. can be selected multiple dishes, and the dishes are selected by one or more order.
- Each order has order details, a unique bill number, date ,and the number of dishes.
- Each Ingredient has calories, a unique kind, and the quantity.
- Many Ingredients are has by one or more meals.
- A customer confirms one order.
- An employee takes one or more orders and gets paid for the order.
- Many orders are prepared by one or more employees.
- the customer may write feedback about ordering on the one hand , service, food quality, and suggestions.

CHEN'S NOTATION




Employee
name
first name
last name
id No. {PK}
gender
role [1..*]
1..1
1..1
Take
Pay
1..*
1..*
1..*
Prepare
1..*
1..1
Manage
Branches
name
contact No. {PK}
address
1..1
Order
/no. of items
order Details
bill No {PK}
date
Write
FeedBack
food Quality
suggestion
service
1..1
1..*
1..*
Contains
1..*
Confirm
1..1
Customer
order details
price
orderNo {FK}
CustomerNo{PK}
Meal
dishes [1..*]
price
mealID{PK}
1..*
1..*
Has
Ingredient
quantity
kind {PK}
calories


UML

Mapping STEP 1:

Branches

Contact No	Name	Address

Ingredient

Calories	Quantity	kind

Employee

IDNo	FName	LName	Gender

Meal

mealID	Price

Order

bill No	OrderDetails	date

FeedBack

FoodQuality	Suggestion	Service

Customer

OrderNo	OrderDetails	Price

Mapping

STEP 2:

FeedBack

bill_No	FoodQuality	Suggestion	Service

Fk

STEP 3:

Branches

Contact No	Name	Address	IDNo

Employee

IDNo	FName	LName	Gender	Role

Mapping STEP 4:

Meal

Price	mealID

Ingredient

Calories	Quantity	kind	mealID (Fk)

Order

bill No	date	OrderDetails	IDNo (Fk)

Employee

IDNo	FName	LName	Gender

FeedBack

FoodQuality	Suggestion	Service	bill No (Fk)

Order

bill No	date	OrderDetails

Order

bill No	date	OrderDetails	IDNo	CustomerNo (Fk)

Customer

CustomerNo	OrderDetails	Price

Mapping

STEP 4:

Meal

Price	mealID (Fk)

Order

Bill No	date	OrderDetails	IDNo	OrderNo

STEP 5:

Employee

IDNo (Fk)	FName	LName	Gender

Order

Bill No (Fk)	date	OrderDetails	IDNo	CustomerNo

Prepare

Bill No (Pk)	IDNo (Pk)

STEP 6:

Meal

Price	mealID

meal-Dishes

mealID	Dishes

Meal

Price	mealID

Ingredient

Calories	Quantity	kind	mealID

Has

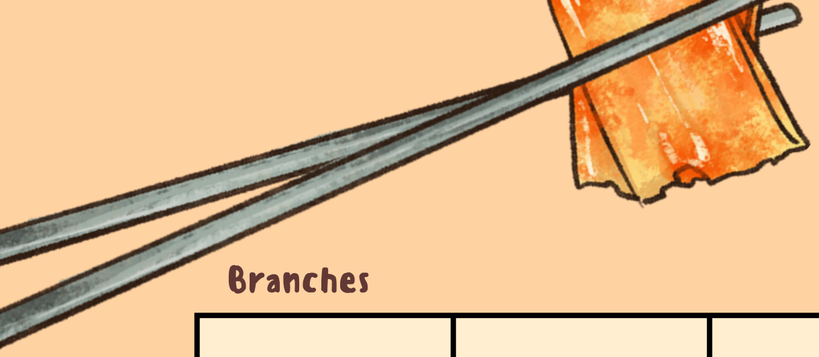
mealID (Pk)	kind (Pk)

STEP 7:

None

Final Mapping

Relational Schema



Branches

Contact No	Name	Address	IDNo

Ingredient

Kind	mealID	Quantity	Calories

Employee

IDNo	FName	LName	Gender

Meal

mealID	Price	Dishes	Bill No

Order

Bill No	date	OrderDetails	CustomerNo	IdNo

FeedBack

Bill No	FoodQuality	Suggestion	Service	IDNo

Customer

CustomerNo	OrderDetails	Price

Prepare

Bill No	IDNo

Has

mealID	kind

NORMALIZATION 1NF

IngredientDetalis

Kind	Quantity	Calories

Ingredient

Kind	mealID

MealPrice

mealID	Price	Bill No

Meal

mealID	Dishes

FeedBack

Bill No	FoodQuality	Suggestion	Service

FeedbackWriter

Bill No	IDNo

EmployeeRole

IDNo.	Role

Employee

IDNo.	FName	LName	Gander

NORMALIZATION

1NF

Branches

Contact No	Address	IDNo

BrancheManger

Name	IDNo

2NF

All relations are 1NF and there is no partial dependency therefore the table is in 2NF

NORMALIZATION 3NF

EmployeeRole

IDNo.	Role

Employee

IDNo.	FName	Gander

EmployeeName

FName	LName

Order

Bill No	OrderDetails	OrderNo	IdNo	data

OrderDetails

OrderDetails	NoItems

Create Tables

Branches

```
create schema 'E-restaurant';

CREATE TABLE Branches (
    ContactNo INT(10) NOT NULL,
    Address VARCHAR(30),
    IDNo INT(10) NOT NULL UNIQUE,
    CONSTRAINT IDNo_PK PRIMARY KEY (ContactNo , IDNo));

create index BM_ID ON Branches (IDNo);
```

BrancheManger

```
CREATE TABLE BrancheManger (
    IDNo INT(10) NOT NULL UNIQUE,
    Name VARCHAR(10) NOT NULL,
    CONSTRAINT DNo_FK FOREIGN KEY (IDNo)
        REFERENCES Branches (IDNo)
        ON DELETE CASCADE ON UPDATE CASCADE);
```

MealPrice

```
CREATE TABLE MealPrice (
    MealID INT(10) NOT NULL,
    Price DECIMAL(4 , 2 ) NOT NULL,
    BillNo INT(10) NOT NULL UNIQUE,
    CONSTRAINT MealID_PK PRIMARY KEY (BillNo , MealID));

create index ME_ID  ON MealPrice (MealID);
```

Meal

```
CREATE TABLE Meal (
    MealID INT(10) NOT NULL UNIQUE,
    Dishes VARCHAR(10) NOT NULL,
    CONSTRAINT MealID_FK FOREIGN KEY (MealID)
        REFERENCES MealPrice (MealID)
        ON DELETE CASCADE);
```

EmployeeRole

```
CREATE TABLE EmployeeRole (
    IDNo INT(10) NOT NULL UNIQUE,
    Role VARCHAR(20),
    CONSTRAINT IDNo_PK PRIMARY KEY (IDNo));
```

Employee

```
CREATE TABLE Employee (
    IDNo INT(10) NOT NULL UNIQUE,
    fName VARCHAR(20) NOT NULL,
    gender CHAR(1) CHECK (gender IN ('F' , 'M')),
    CONSTRAINT IDNo_FK FOREIGN KEY (IDNo)
        REFERENCES EmployeeRole (IDNo)
        ON DELETE CASCADE ON UPDATE CASCADE);

create index EM_ID  ON Employee (fName);
```

EmployeeName

```
CREATE TABLE EmployeeName (
    fName VARCHAR(20) NOT NULL,
    lName VARCHAR(20),
    CONSTRAINT fName_FK FOREIGN KEY (fName)
        REFERENCES Employee (fName)
        ON DELETE CASCADE ON UPDATE CASCADE);
```

Create Tables

Orders

```
CREATE TABLE Orders (
    OrderNo INT(10) NOT NULL UNIQUE,
    OrderDetails VARCHAR(100),
    billNo INT(10) NOT NULL UNIQUE,
    idNo INT(10) NOT NULL UNIQUE,
    NoItems INT(10) NOT NULL,
    dateOforder DATE,
    CONSTRAINT Or_PK PRIMARY KEY (OrderNo , idNo , billNo),
    CONSTRAINT OR_FK FOREIGN KEY (OrderNo)
        REFERENCES CUSTOMER (CustomerNo)
        ON DELETE CASCADE ON UPDATE CASCADE,
    CONSTRAINT OR2_FK FOREIGN KEY (idno)
        REFERENCES Employee (IDNo)
        ON DELETE CASCADE ON UPDATE CASCADE);
```

OrderDetails

```
CREATE TABLE OrderDetails (
    OrderDetails VARCHAR(100),
    NoOfItems INT(10),
    OrderNo INT(10) NOT NULL UNIQUE,
    CONSTRAINT ord_FK FOREIGN KEY (OrderNo)
        REFERENCES Orders (OrderNo)
        ON DELETE CASCADE ON UPDATE CASCADE);
```

Customer

```
CREATE TABLE CUSTOMER (
    CustomerNo INT(10) UNIQUE NOT NULL,
    Price INT(10),
    OrderDetails VARCHAR(100),
    CONSTRAINT CU_PK PRIMARY KEY (CustomerNo));
```

IngredientDetails

```
CREATE TABLE IngredientDetails (
    Kind VARCHAR(20) UNIQUE NOT NULL,
    Quantity INT(100) NOT NULL,
    calories INT(255) NOT NULL,
    CONSTRAINT Kind_PK PRIMARY KEY (Kind));
```

Ingredient

```
CREATE TABLE Ingredient (
    Kind VARCHAR(20) NOT NULL,
    MealID INT(10) NOT NULL,
    CONSTRAINT MealID_PK FOREIGN KEY (MealID)
        REFERENCES MealPrice (MealID)
        ON DELETE CASCADE ON UPDATE CASCADE,
    CONSTRAINT Kind_PK FOREIGN KEY (Kind)
        REFERENCES IngredientDetails (Kind)
        ON DELETE CASCADE ON UPDATE CASCADE);
```

Prepare

```
CREATE TABLE Prepare (
    BillNo INT(10) NOT NULL UNIQUE,
    IDNo INT(10) NOT NULL UNIQUE,
    CONSTRAINT BillNo_PK FOREIGN KEY (BillNo)
        REFERENCES MealPrice (BillNo)
        ON DELETE CASCADE,
    CONSTRAINT IDNo_PK FOREIGN KEY (IDNo)
        REFERENCES employee (IDNo)
        ON DELETE CASCADE);

create index Bill_No ON orders (billNo);
```

```
create index Bill_No ON orders (billNo);

CREATE TABLE FeedBack (
    BillNo INT(10) NOT NULL UNIQUE,
    FoodQuality INT(5),
    Service INT(5),
    Suggestion VARCHAR(500),
    CONSTRAINT BillNo_FK FOREIGN KEY (BillNo)
        REFERENCES orders (BillNo)
        ON DELETE CASCADE);

CREATE TABLE FeedbackWriter (
    BillNo INT(10) NOT NULL UNIQUE,
    IDNo INT(10) NOT NULL UNIQUE,
    CONSTRAINT BillNo_FK2 FOREIGN KEY (BillNo)
        REFERENCES feedback (BillNo)
        ON DELETE CASCADE,
    CONSTRAINT IDNo_FK2 FOREIGN KEY (IDNo)
        REFERENCES employee (IDNo)
        ON DELETE CASCADE);
```

FeedBack

FeedbackWriter

Create Tables

INSERT DATA

```
106 •   insert into branches values(0503320123, 'Makkah - Alzaher', 01),
107                                 (0526404782,'Jeddah - Alnuzha', 02),
108                                 (0576198426,'Jeddah - Alhamra', 03),
109                                 (0599124775,'Makkah - Alaziziyah', 04),
110                                 (0588844236,'Tife - Alryyan', 05);
111 •   SELECT * FROM branches order by IDNo;
```

Result Grid | Filter Rows: | Edit: | Export/Import:

ContactNo	Address	IDNo
503320123	Makkah - Alzaher	1
526404782	Jeddah - Alnuzha	2
576198426	Jeddah - Alhamra	3
599124775	Makkah - Alaziziyah	4
588844236	Tife - Alryyan	5

```
113 •   insert into branchemanger values(01,'Ahmad'),(02, 'Lna'),(03,'Khald'),
114                                      (04, 'Sara'),(05,'Muna');
115 •   SELECT * FROM branchemanger;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

IDNo	Name
1	Ahmad
2	Lna
3	Khald
4	Sara
5	Muna

```
160 •   insert into meal values
161     (1,'Shrimp dumplings'),
162     (2,'Rice noodle rolls'),
163     (3,'Chow mein'),
164     (4,' Sushi'),
165     (5,'Kimchi');
166 •   select*from meal;
167
```

Result Grid | Filter Rows:

MealID	Dishes
1	Shrimp dumplings
2	Rice noodle rolls
3	Chow mein
4	Sushi
5	Kimchi

```
127 •   insert into mealprice values
128     (1,55.5,0),
129     (2,36,1),
130     (3,60.99,2),
131     (4,30.2,3),
132     (5,20.19,4);
133 •   select*from mealprice;
```

Result Grid | Filter Rows:

MealID	Price	BillNo
1	55.50	0
2	36.00	1
3	60.99	2
4	30.20	3
5	20.19	4

```
30 •   insert into employee
31     values
32     (0501346834,'Ahmed','M'),
33     (0533789325,'Ali','M'),
34     (0981265338,'Heba','F'),
35     (0433096119,'Tariq','M'),
36     (0501266452,'Ashraf','M');
37
38 •   select * from employee;
```

Result Grid | Filter Rows:

IDNo	fName	gender
501346834	Ahmed	M
533789325	Ali	M
981265338	Heba	F
433096119	Tariq	M
501266452	Ashraf	M

```
20 •   insert into employeerole
21     values
22     (0501346834,'waiter'),
23     (0533789325,'chef'),
24     (0981265338,'General Manager'),
25     (0433096119,'Executive Chef'),
26     (0501266452,'Barista');
27 •   select * from employeerole;
```

Result Grid | Filter Rows:

IDNo	Role
433096119	Executive Chef
501266452	Barista
501346834	waiter
503378932	chef
981265338	General Manager

INSERT DATA

```
40 •   insert into employeename
41     values
42     ('Ahmed','Yasser'),
43     ('Ali','Basm'),
44     ('Heba','Amir'),
45     ('Tariq','Adam'),
46     ('Ashraf','Khaled');
47 •   select * from employeename;
```

Result Grid | Filter Rows:

fname	lname
Ahmed	Yasser
Ali	Basm
Heba	Amir
Tariq	Adam
Ashraf	Khaled

```
229 •   insert into IngredientDetails values
230     ('Shrimp', 5 , 250 ),
231     ('Rice Noodles', 50 , 250),
232     ('Noodle' , 47 , 241),
233     ('Nori and Rice' , 6 , 197),
234     ('Scallions and Sauce' , 5 , 86);
235 •   SELECT * FROM IngredientDetails;
```

Result Grid | Filter Rows: | Edit:

Kind	Quantity	calories
Noodle	47	241
Nori and Rice	6	197
Rice Noodles	50	250
Scallions and Sauce	5	86
Shrimp	5	250
NULL	NULL	NULL

```
47 •   INSERT INTO Orders
48     values(312,'Tuna Sandwich',10,51,2,'23/12/30'),
49     (311,'Suchi',11,51,3,'23/1/29'),
50     (310,'Ispagity',12,51,1,'23/5/1'),
51     (308,'bulgogi',13,51,4,'23/4/1'),
52     (309,'kimchi',14,51,4,'23/12/25');
53 •   SELECT *
```

Result Grid | Filter Rows: | Edit: | Export/Import:

OrderNo	OrderDetails	billNo	Idno	NoItems	orderDate
308	bulgogi	13	51	4	2023-04-01
309	kimchi	14	51	4	2023-12-25
310	Ispagity	12	51	1	2023-05-01
311	Suchi	11	51	3	2023-01-29
312	Tuna Sandwich	10	51	2	2023-12-30
NULL	NULL	NULL	NULL	NULL	NULL

```
71 •   INSERT INTO OrderDetails
72     values('bulgogi',4,308);
73 •   INSERT INTO OrderDetails
74     values('kimchi',4,309);
75 •    INSERT INTO OrderDetails
76     values('Ispagity',1,310);
77 •     INSERT INTO OrderDetails
78     values('Suchi',3,311);
79 •    INSERT INTO OrderDetails
80     values('Tuna Sandwich',2,312);
81 •   SELECT *
82     FROM OrderDetails;
```

Result Grid | Filter Rows: | Export:

OrdersDetails	NoOfItems	OrderNo
bulgogi	4	308
kimchi	4	309
Ispagity	1	310
Suchi	3	311
Tuna Sandwich	2	312

```
220 •   insert into Ingredient values
221     ('Shrimp',1),
222     ('Rice Noodles',2),
223     ('Noodle',3),
224     ('Nori and Rice',4),
225     ('Scallions and Sauce',5);
226 •   select * from Ingredient;
227
228
```

Result Grid | Filter Rows: | Export:

Kind	MealID
Shrimp	1
Rice Noodles	2
Noodle	3
Nori and Rice	4
Scallions and Sauce	5

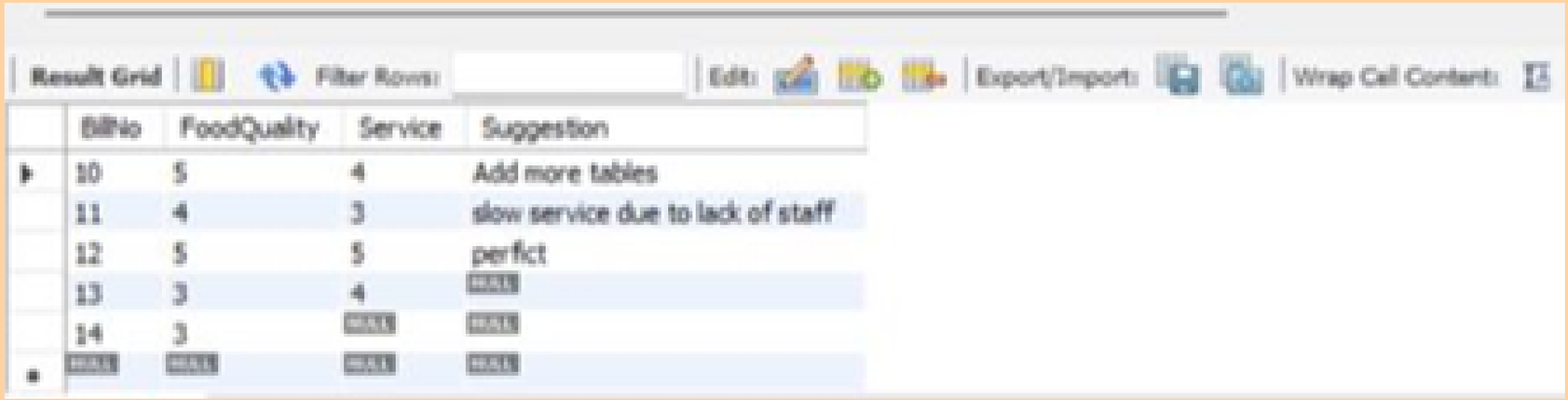
```
118 •   INSERT INTO CUSTOMER
119     VALUES (312,'Tuna Sandwich',15),
120       (311,'Suchi',17),
121       (310,'Ispagity',20),
122       (309,'kimchi',35),
123       (308,'bulgogi',25);
124 •     select *FROM CUSTOMER;
```

Result Grid | Filter Rows: | Edit:

CustomerNo	OrderDetails	Price
308	bulgogi	25
309	kimchi	35
310	Ispagity	20
311	Suchi	17
312	Tuna Sandwich	15
NULL	NULL	NULL

INSERT DATA

```
insert into feedback values(10, 5 , 4, 'Add more tables'),
                           (11,4, 3,'slow service due to lack of staff'),
                           (12,5, 5,'perfict'),
                           (13,3, 4,null),
                           (14,3, null, null);
```



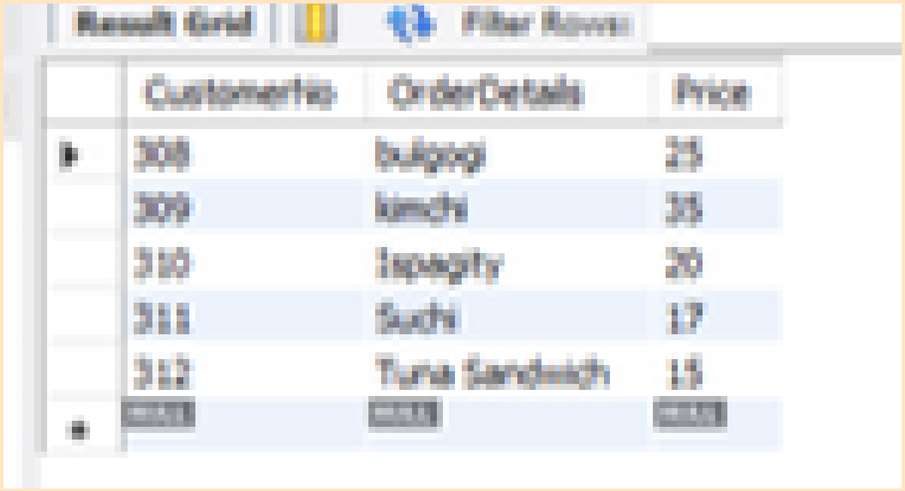
Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

BillNo	FoodQuality	Service	Suggestion
10	5	4	Add more tables
11	4	3	slow service due to lack of staff
12	5	5	perfict
13	3	4	NULL
14	3	NULL	NULL
NULL	NULL	NULL	NULL

```
insert into feedbackwriter
values(10, 0501346834),
(11,0533789325),
(12,0981265338),
(13,0433096119),
(14,0501266452);
```

SELECT

update

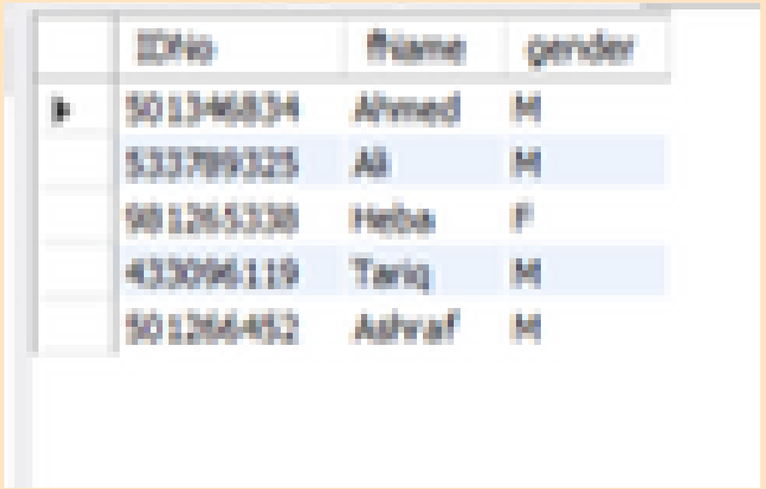


	CustomerNo	OrderDetails	Price
▶	308	bulgogi	25
	309	kimchi	35
	310	Ispagity	20
	311	Suchi	17
	312	Tuna Sandwich	15
*	NULL	NULL	NULL

```
# (update) change orderdetails Tuna Sandwich to 'KOREAN CANDIED SWEET POTATO'
SELECT * FROM customer;
UPDATE customer
SET orderdetails = 'KOREAN CANDIED SWEET POTATO'
WHERE orderdetails='Tuna Sandwich';
```

	CustomerNo	OrderDetails	PRICE
▶	308	bulgogi	25
	309	kimchi	35
	310	Ispagity	20
	311	Suchi	17
	312	KOREAN CANDIED SWEET POTATO	15
*	NULL	NULL	NULL

delete



	IDNo	fname	gender
▶	501346834	Ahmed	M
	533789325	Ali	M
	981265338	Heba	F
	433096119	Tariq	M
	501264452	Ashraf	M

```
# delete IDNo to Ahmed is 0501346834
delete from employee
where IDNo=0501346834;
```

	IDNo	fname	gender
▶	533789325	Ali	M
	981265338	Heba	F
	433096119	Tariq	M
	501264452	Ashraf	M

SELECT

Where

```
53 •  SELECT *
54    FROM Orders
55     where  billNo=11 ;
56 •  INSERT INTO CUSTOMER
```

Result Grid | Filter Rows: | Edit: | Export/Imp

OrderNo	OrderDetails	billNo	Idno	NoItems	orderDate
311	Suchi	11	51	3	2023-01-29
NULL	NULL	NULL	NULL	NULL	NULL

Order by

```
53 •  SELECT *
54    FROM Orders
55     Order by orderDate  ;
56 •  INSERT INTO CUSTOMER
```

Result Grid | Filter Rows: | Edit: | Exp

OrderNo	OrderDetails	billNo	Idno	NoItems	orderDate
311	Suchi	11	51	3	2023-01-29
308	bulgogi	13	51	4	2023-04-01
310	Ispagity	12	51	1	2023-05-01
309	kimchi	14	51	4	2023-12-25
312	Tuna Sandwich	10	51	2	2023-12-30
NULL	NULL	NULL	NULL	NULL	NULL

Orders 5 ×

```
16
17 •  SELECT * FROM IngredientDetails
18    WHERE Quantity= 5;
```

Result Grid | Filter Rows: | Edit

Kind	Quantity	calories
Scallions and Sauce	5	86
Shrimp	5	250
NULL	NULL	NULL

```
16
17 •  SELECT * FROM IngredientDetails
18    ORDER BY calories;
```

Result Grid | Filter Rows:

Kind	Quantity	calories
Scallions and Sauce	5	86
Nori and Rice	6	197
Noodle	47	241
Rice Noodles	50	250
Shrimp	5	250
NULL	NULL	NULL

```
20 •  SELECT *
21    FROM MealPrice
22    ORDER BY Price;
23
```

100% | 1:18

Result Grid | Filter Rows: | Sear

MealID	Price	BillNo
5	20.19	4
4	30.20	3
2	36.00	1
1	55.30	8
1	55.50	0
3	60.99	2

SELECT

Join

```
177 •   select IDNo,fName,gender from employee where gender = 'M'
178     and( IDNo =(select IDNo from employeerole where role = 'waiter') );
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

IDNo	fName	gender
921346834	Ahmed	M

Group by

```
280     #Calculates calories that are less than 243
281 •   SELECT calories, COUNT(*) AS number
282     FROM IngredientDetails
283     WHERE (calories)<243
284     GROUP BY calories;
285
```

Result Grid | Filter Rows: | Export:

calories	number
241	1
197	1
86	1

Subqueries

```
10
11 •  SELECT *
12    FROM Meal
13    WHERE MealID IN (SELECT MealID
14                     FROM MealPrice
15                      WHERE Price<40);
16
```

100% | 2:9

Result Grid | Filter Rows: | Search

MealID	Dishes
2	Rice noodle rolls
4	Sushi
5	Kimchi

```
16 •  SELECT *
17    FROM IngredientDetails
18    WHERE calories= (SELECT MIN(calories)
19    FROM IngredientDetails);
```

Result Grid | Filter Rows: | Edit:

Kind	Quantity	calories
Scallions and Sauce	5	86
NULL	NULL	NULL

```
286     # Calculates prices that are less than 50
287 •   SELECT Price, COUNT(*) AS number
288     FROM mealprice
289     WHERE (Price)<50
290     GROUP BY Price;
291
```

Result Grid | Filter Rows: | Export:

Price	number
36.00	1
30.20	1
30.19	1

THANK YOU!