

Database Management System Project (Review-1)

Topic : Online Food Ordering System

Submitted By-

Dhiraj Zambare (19BIT0095)

Pranav Mahalpure (19BIT0109)

Slotno: L33+34

Project Name: Online Food Ordering System

Introduction

Our project is based on Online food ordering System. It facilitates customers to buy food from a restaurant through online. This project performs various functions such as providing data regarding the customers who make use of the online portal for ordering food. It is also used to store their account username and password. The information regarding the available food is also stored in this database, which can be used by the customer to order food. The food ordered by the user is stored in a separate table. After which the total cost is calculated and the payment information is stored. The delivery details of the food is also stored.

DATA REQUIREMENT:

Entities:

- 1) **USER** : It is an entity type which stores various information regarding the customer such as User_id, Name, Address, Phone Number, Date of Birth, Email. The data regarding the user is accessed through this entity type.
- 2) **LOGIN** : This entity type stores value of the username and password of the user accounts. This Entity type is used for verifying the identity of the user.
- 3) **FOOD** : This Entity type stores the information regarding the various types of food which the restaurant provides. It stores the values of the food_id, name, description, price of all the food provided and the type of food provided.
- 4) **Category** : It tells the category in which the food fall under (eg: breakfast, lunch, dinner).It also stores information regarding the category id and the name of category.

- 5) **Food Image** : This Entity type stores the information regarding Food name its images and the id associated with it.
- 6) **RESTAURANT** : This Entity type stores information about different restaurants from which user can order food. It contains data of restaurant name, id, address, rating, description.
- 7) **ORDER** : This Entity type stores values regarding the orders made by the user. It has Attributes to store information such as Quantity, Price, Order_no.
- 8) **PAYMENT** : This Entity type stores the payment information of the users for the items they bought. It stores values of the Total amount need to be paid , Pay_id, status, bank, Card holder name,Expiry date.

RELATIONSHIPS

- 1) **USER logs in by LOGIN (1:N)** : The user will login to their account by entering their username and password. A user might have multiple accounts with different usernames. But an account is accessed by a single user. An account should have atleast one user. Like wise a user should have an account before ordering food.
- 2) **USER issues ORDER (1:N)** : User issues orders for the food available. A user can have multiple orders .But a order can belong to only one user. A user does not need to order anything. But a order should have one user.
- 3) **ORDER delivered_to USER (N:1)** : The order is delivered to the user. A user can have multiple orders delivered while 1 order can be delivered to only one user. A order is always delivered if the payment is made. If an order is delivered , it can only be delivered to the user who ordered it. It has the relationship attribute of time_taken and distance
- 4) **ORDER contain FOOD (N:1)** : Orders contain the food ordered by the user . A order can have only a single food. But a food might be ordered by multiple users with different order_id's. A order should contain atleast one food ,but a food is not necessary to be ordered
- 5) **FOOD belongs to RESTAURANT (M:N)** : The food selected by the user belongs to a restaurant. A restaurant can contain many foods. Similary a food can belong to many restaurants. A restaurant may not have all the food. But a food will always belong to a restaurant.
- 6) **FOOD has Category (N:1)** : Food has different category. A specific food might have single category. But there may be different category for different types of food.
- 7) **FOOD has Food Images (1:N)** : For a specific food there may be different images but a image belongs to a single food only.

8) **ORDER has PAYMENT (N:1)** : An order needs to be paid in online before delivering the food. An order can be paid by only one user while the payment by user can contain multiple orders. An order should be paid by the user or else the order is not valid. Similarly the payment should be made to atleast one user.

FUNCTIONAL REQUIREMENTS

- 1) **VIEWER** : The System must allow the users to login into their account if their username and password is correct. If they don't have an account they must register first. After logging in the user is allowed to view their profile and can also make changes to it. The system then displays the restaurant from which the user wishes to order. Then the user selects the food. The food is categorized into breakfast, lunch, dinner. After choosing the category of food, the system displays the food assigned under that category. The food name along with price ,type and description is displayed. The user should select his/her preferred food and specify the quantity. This will be added to his/her cart which contains the orders placed by the user.

After placing his/her orders. The system shows the orders placed by the user. Then, the user can add more orders or delete orders. After which the user proceeds to payment in which he enters his payment details. After payment the system provides the option to see delivery details.

BASIC ANOLOGY:

- View the website with the browser.
- Login to the website.
- View all Restaurants.
- View all categories of food.
- View all food items.
- Selecting the required order.
- View all orders placed by the user.
- View the total price along with payment details.
- View delivery details.

View all available food:

- a) Name of the food

- b) Price of the food
- c) Description of the food
- d) Type of food

View orders placed:

- a) Name of the food
- b) Quantity
- c) Price

View delivery details:

- a) User_id
- b) Date
- c) Distance
- d) Estimated time

2) ADMINISTRATOR :

Administrator is responsible for creating the website which is used to access the database and also modifying it . The administrator has all the privileges of the user but also the authority to add and remove data from the database which the user cannot do.

Administrator is responsible for creating different user accounts and assigning the id and password. Administrators are the one who generate the fixtures and update them in the database. Further the administrator can delete an user account if the user wishes to delete their account or if the account has been inactive for a very long time. They should have access to FOOD and RESTAURANT table. They can add, delete or modify the information stored in the tables. The administer can change the details of the restaurant.

If the user orders a particular food , then the administrator can decrease the availability of the particular dish or if the restaurant has received a new batch of food, they can increase the availability. The administrator is also responsible for changing payment status.

BASIC ANALOGY:

- Create Website
- Generate login ID for user
- Design website
- Display different restaurants
- Display different food menus
- Display user orders
- Display user payment details
- Display user delivery details

3) REMOVAL OF OLD DATA :

I. If the user wishes to delete his user account ,then the data regarding the user in USER table should be deleted in the user table.

II. If the delivery is completed for a particular order then the data from delivery table and order table is deleted.

III. If the user wishes to cancel their payment , then the data regarding the order made by user should be deleted from PAYMENT table.

IV. If the restaurant wishes to remove a particular food from their menu, then data regarding that particular food should be removed from FOOD table.

V. If a restaurant is closed permanently or if they decline online service. Then the data regarding it will be deleted from RESTAURENT table.

4) MODIFICATION OF DATA :

I. If the user wishes to modify his/her profile, then the data in USER table needs to be modifies.

II. If the user wishes to change his/her account password ,then the data in LOGIN table should be modified.

III. If the user wishes to modify the quantity of food he ordered, then the data in ORDER table needs to be modified.

IV. If the restaurant decides to change the name or price of a food, then the data in FOOD table should be modified.

V. If the details of a restaurant needs to be changed. Then the data from restaurant table is modified.

VI. If the user orders a specific food , the availability of the food is decreased by the quantity of food the user ordered.

VII. If the user has finished his/her payment, the payment status is changed from false to true.

5) RETRIVAL OF DATA :

I. For the user to view his/her account details, we have to retrieve

- a) Username
- b) Name
- c) User_id
- d) Date of Birth
- e) Email
- f) Address

II. View username and password during password modification, we need to retrieve

- a) Username
- b) Password

III. View different restaurants under a specified distance from the user address.

- a) Name
- b) Address
- c) Description
- d) rating

IV. View all foods under a specific category

- a) Name
- b) Price
- c) Description
- d) Type

V. View all foods under a specific type

- a) Name
- b) Price
- c) Description

VI. When the user wishes to view the items he/she ordered, we retrieve from ORDER

- a) Order_id
- b) Food Name
- c) Price
- d) Quantity

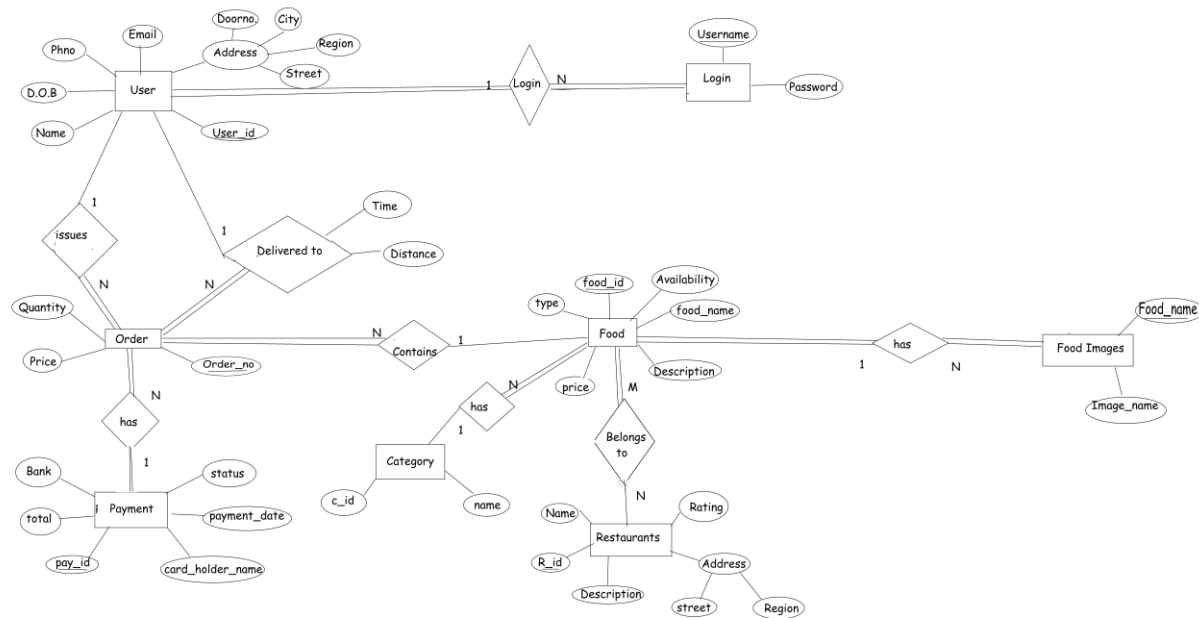
VII. When the user proceed to payment, we need to display

- a) User_id
- b) Total cost

VIII. When the user wishes to delivery details

- a) Estimated time
- b) Distance
- c) Date

ER MODEL along with key constraints, participation constraints and cardinality constraints:



Thank You!

Database Management System Project (Review-2)

Topic : Online Food Ordering System

Submitted By-

Dhiraj Zambare (19BIT0095)

Pranav Mahalpure (19BIT0109)

Slotno: L33+34

Project Name: Online Food Ordering System

Relational Model

USER INFO

<u>user_id</u>	name	email	dob	phone	doorno.	street	region	city
----------------	------	-------	-----	-------	---------	--------	--------	------

LOGIN

<u>user_id</u>	<u>username</u>	password
----------------	-----------------	----------

DELIVERY

<u>order_no</u>	user_id	time	distance
-----------------	---------	------	----------

ORDERS

user_id	<u>order_no</u>	food_id	pay_id	quantity	price
---------	-----------------	---------	--------	----------	-------

PAYMENT

<u>Pay_id</u>	Total	Bank	Card_Holder_Name	Exp.date	Status
---------------	-------	------	------------------	----------	--------

RESTAURANT

<u>R_id</u>	name	description	rating	street	region
-------------	------	-------------	--------	--------	--------

BELONGS

<u>R_id</u>	<u>Food_id</u>
-------------	----------------

FOOD

<u>Food_id</u>	Types	F_name	price	description	availability
----------------	-------	--------	-------	-------------	--------------

FOOD IMAGES

<u>Food_id</u>	Food_name	Image_name
----------------	-----------	------------

TABLES:

- USER_INFO
- LOGIN
- ORDERS
- FOOD
- BELONGS
- RESTAURANT
- PAYMENT
- DELIVERY
- FOOD_IMAGES

CREATE TABLES

CODE:

```
CREATE or REPLACE TYPE categ as VARRAY(3) of VARCHAR(10)
```

```
/
```

```
create table user_info(
```

```
    user_id number(6) constraint user_pk primary key,
```

```
    name varchar(20) not null,
```

```
    dob DATE not null,
```

```
    email varchar(30),
```

```
    phone number(10) not null,
```

```
    door_no varchar(10) not null,
```

```
    street varchar(20) not null,
```

```
    region varchar(20)not null,
```

```
    city varchar(20) not null
```

```
);
```

```
create table LOGIN(
```

```
    user_id constraint login_fk references user_info,
```

```
    username varchar(30) constraint login_pk primary key,
```

```
    password varchar(30) not null);
```

```
create table FOOD(  
    food_id number(5) constraint food_pk primary key,  
    type varchar(10),  
    f_name varchar(30) not null,  
    price number(10) not null,  
    description varchar(40),  
    category categ,  
    availability varchar(20)  
);
```

```
create table RESTAURANT(  
    r_id number(6) constraint restaurant_pk primary key,  
    name varchar(30) not null,  
    description varchar(60),  
    rating number(5,1),  
    street varchar(30) not null,  
    region varchar(30) not null  
);
```

```
create table BELONGS(  
    r_id constraint restaurant_fk references RESTAURANT,  
    food_id constraint rest_food_fk references FOOD,  
    constraint belongs_pk primary key(r_id,food_id)  
);
```

```
create table PAYMENT(  
    pay_id number(5) constraint payment_pk primary key, total number(10) not null,  
    bank varchar(30), card_holder_name varchar(20), exp_date DATE,  
    status varchar(20) not null  
);
```

```
create table ORDERS(  
  user_id constraint order_user_fk references USER_INFO,  
  order_no number(6) constraint order_pk primary key,  
  food_id constraint order_food_fk references FOOD,  
  pay_id constraint order_pay_fk references PAYMENT,  
  quantity number(10) not null,  
  t_price number(10) not null  
);  
  
create table DELIVERY(  
  order_no constraint deliver_order_fk primary key references ORDERS,  
  user_id constraint deliver_user_fk references USER_INFO,  
  time timestamp(0) not null,  
  distance number(10) not null  
);  
  
create table food_image(  
  food_id constraint food_images_fk references food,  
  f_name varchar(30) not null,  
  image_name varchar(70) not null,  
  constraint fimage primary key( food_id) );
```

SCREENSHOTS

USER_INFO:

```
SQL> create table user_info(  
 2  user_id number(6) constraint user_pk primary key,  
 3  name varchar(20) not null,  
 4  dob DATE not null,  
 5  email varchar(30),  
 6  phone number(10) not null,  
 7  door_no varchar(10) not null,  
 8  street varchar(20) not null,  
 9  region varchar(20)not null,  
10  city varchar(20) not null  
11 );
```

Table created.

LOGIN:

```
SQL> create table LOGIN(  
 2  user_id constraint login_fk references user_info,  
 3  username varchar(30) constraint login_pk primary key,  
 4  password varchar(30) not null);
```

Table created.

FOOD:

```
SQL> create table FOOD(  
 2  food_id number(5) constraint food_pk primary key,  
 3  type varchar(10),  
 4  f_name varchar(30) not null,  
 5  price number(10) not null,  
 6  description varchar(40),  
 7  category categ,  
 8  availability varchar(20)  
 9 );
```

Table created.

RESTAURANT:

```
SQL> create table RESTAURANT(  
  2  r_id number(6) constraint restaurant_pk primary key,  
  3  name varchar(30) not null,  
  4  description varchar(60),  
  5  rating number(5,1),  
  6  street varchar(30)not null,  
  7  region varchar(30) not null  
  8  );
```

Table created.

BELONGS:

```
SQL> create table BELONGS(  
  2  r_id constraint restaurant_fk references RESTAURANT,  
  3  food_id constraint rest_food_fk references FOOD,  
  4  constraint belongs_pk primary key(r_id,food_id)  
  5  );
```

Table created.

PAYMENT:

```
SQL> create table PAYMENT(  
  2  pay_id number(5) constraint payment_pk primary key, total number(10) not null,  
  3  bank varchar(30), card_holder_name varchar(20), exp_date DATE,  
  4  status varchar(20) not null  
  5  );
```

Table created.

ORDERS:

```
SQL> create table ORDERS(  
  2  user_id constraint order_user_fk references USER_INFO,  
  3  order_no number(6) constraint order_pk primary key,  
  4  food_id constraint order_food_fk references FOOD,  
  5  pay_id constraint order_pay_fk references PAYMENT,  
  6  quantity number(10) not null,  
  7  t_price number(10) not null);
```

Table created.

DELIVERY:

```
SQL> create table DELIVERY(  
  2  order_no constraint deliver_order_fk primary key references ORDERS,  
  3  user_id constraint deliver_user_fk references USER_INFO,  
  4  time timestamp(0) not null,  
  5  distance number(10) not null  
  6 );
```

Table created.

FOOD_IMAGE:

```
SQL> create table food_image(  
  2  food_id constraint food_images_fk references food,  
  3  f_name varchar(30) not null,  
  4  image_name varchar(70) not null,  
  5  constraint fimage primary key( food_id)  
  6 );
```

Table created.

INSERT VALUES INTO TABLES

USER_INFO:

insert into user_info values(67534,'Liam',DATE '1996-05-20','liam123@gamil.com',8976835689, '2','EveningBazzard','ParkTown','Chennai')

insert into user_info values(95674,'Noah',DATE '1989-02-12','Noah_678@gamil.com',9845298640,'17','BPC Plaza','T nagar','Chennai')

insert into user_info values(86386,'Isabella',DATE '1999-09-04','Isa_bella@gamil.com',7849378597, '167/2 F12','Eldams Square','Alwarpet','Chennai')

insert into user_info values(78493,'William',DATE '2000-11-15','Will5050@gamil.com',8674903980,'1','BabaTowers Sterling','Nungambakkam','Chennai')

insert into user_info values(32794,'Olivia',DATE '1997-01-19','Olivia20@gamil.com',9674536896,'84','Audiappa Naikan st','Sowcarpet','Chennai')

```
SQL> insert into user_info values(67534,'Liam',DATE '1996-05- 20','liam123@gamil.com',8976835689, '2','EveningBazzard','ParkTown','Chennai')
2 ;
```

```
1 row created.
```

```
SQL> insert into user_info values(95674,'Noah',DATE '1989-02- 12','Noah_678@gamil.com',9845298640,'17','BPC Plaza','T nagar','Chennai');
```

```
1 row created.
```

```
SQL> insert into user_info values(86386,'Isabella',DATE '1999-09- 04','Isa_bella@gamil.com',7849378597, '167/2 F12','Eldams Square','Alwarpet','Chennai');
```

```
1 row created.
```

```
SQL> insert into user_info values(78493,'William',DATE '2000-11- 15','Will5050@gamil.com',8674903980,'1','BabaTowers Sterling','Nungambakkam','Chennai');
```

```
1 row created.
```

```
SQL> insert into user_info values(32794,'Olivia',DATE '1997-01- 19','Olivia20@gamil.com',9674536896,'84','Audiappa Naikan st','Sowcarpet','Chennai');
```

```
1 row created.
```

```
SQL>
```


LOGIN:

insert into login values(67534,'Liam123','iamliam')

insert into login values(95674,'Noah678','xyznoah')

insert into login values(86386,'Isabella12','isa12949')

insert into login values(78493,'William50','\$will\$')

insert into login values(32794,'Olivia20','greeneearth')

```
SQL> insert into login values(67534,'Liam123','iamliam') ;  
1 row created.  
SQL> insert into login values(95674,'Noah678','xyznoah') ;  
1 row created.  
SQL> insert into login values(86386,'Isabella12','isa12949') ;  
1 row created.  
SQL> insert into login values(78493,'William50','$will$') ;  
1 row created.  
SQL> insert into login values(32794,'Olivia20','greeneearth');  
1 row created.  
SQL>
```

FOOD:

insert into food values(11111,'Veg','Idly',15,'Rice,Urad dal',categ('breakfast','lunch','dinner'),50);

insert into food values(12993,'Veg','Aapam',15,'Rice,Batter',categ('breakfast','lunch','dinner'),30);

insert into food values(12847,'Veg','Chola Poori',50,'Maida,Chick peas,Garlic,Tomato',categ('breakfast',null,'dinner'),20);

insert into food values(12789,'Veg','Utthapam',30,'Onion,Rice,Urad dal,White Lentils',categ('breakfast',null,'dinner'),25);

insert into food values(11993,'Veg','Onion Pakoda',20,'Chana dal flour,rice flour,ginger',categ('breakfast','lunch',null),30);

insert into food values(11283,'Veg','Paneer fry',60,'Paneer,Garam masala,chilli,tomato',categ(null,'lunch','dinner'),15);

insert into food values(12132,'Veg','chappati',20,'wheat flour,water',categ('breakfast','lunch','dinner'),60);

insert into food values(12243,'Veg','Veg Meals',90,'Combo of various veg foods',categ(null,'lunch',null),50);

insert into food values(12446,'Veg','Veg Briyani',70,'Basmati rice,Garam masala,chili,Peas',categ(null,'lunch',null),40);

insert into food values(21838,'Non-veg','Chicken idly',25,'Chicken,Rice,Urad dal',categ('breakfast','lunch','dinner'),45);

insert into food values(21934,'Non-veg','Egg Dosa',20,'egg,Rice flour',categ('breakfast',null,'dinner'),30);

insert into food values(22893,'Non-veg','fish curry',45,'Fish,curry leaves,chilli,Curry powder',categ('breakfast','lunch','dinner'),20);

insert into food values(29834,'Non-veg','Chicken Pakoda',30,'chicken,cheda dal flour,rice flour',categ('breakfast','lunch',null),25);

insert into food values(22444,'Non-veg','Chicken pepper fry',60,'chicken,pepper',categ(null,null,'dinner'),20);

insert into food values(27373,'Non-veg','Non-veg Meals',120,'combo of various non-veg foods',categ(null,'lunch','dinner'),50);

insert into food values(23783,'Non-veg','Chicken Briyani',90,'Chicken,Basmati rice,Garam masala,chili',categ(null,'lunch',null),40);

insert into food values(29323,'Non-veg','Egg briyani',80,'Egg,Basmati rice,Garam

```
masala,chili',categ(null,'lunch',null),43);
```

```
SQL> insert into food values(11111,'Veg','Idly',15,'Rice,Urad dal',  
2 categ('breakfast','lunch','dinner'),50);
```

```
1 row created.
```

```
SQL> insert into food values(12993,'Veg','Aapam',15,'Rice,Batter',  
2 categ('breakfast','lunch','dinner'),30);
```

```
1 row created.
```

```
SQL> insert into food values(12847,'Veg','CholaPoori',50,'Maida,Chick  
2 peas,Garlic,Tomato',categ('breakfast',null,'dinner'),20);
```

```
1 row created.
```

```
SQL> insert into food values(12789,'Veg','Utthapam',30,'Onion,Rice,Uraddal,White Lentils',  
2 categ('breakfast',null,'dinner'),25);
```

```
1 row created.
```

```
SQL> insert into food values(12789,'Veg','Utthapam',30,'Onion,Rice,Uraddal,White Lentils',
  2  categ('breakfast',null,'dinner'),25);

1 row created.

SQL> insert into food values(11993,'Veg','OnionPakoda',20,'ChanadalfLOUR,rice
  2  flour,ginger',categ('breakfast','lunch',null),30);

1 row created.

SQL> insert into food values(11283,'Veg','Paneerfry',60,'Paneer,Garam
  2  masala,chilli,tomato',categ(null,'lunch','dinner'),15);

1 row created.

SQL> insert into food values(12132,'Veg','chappati',20,'wheat
  2  flour,water',categ('breakfast','lunch','dinner'),60);

1 row created.

SQL> insert into food values(12243,'Veg','VegMeals',90,'Comboofvariousveg
  2  foods',categ(null,'lunch',null),50);

1 row created.

SQL> insert into food values(12446,'Veg','VegBriyani',70,'Basmatirice,Garam
  2  masala,chilli,Peas',categ(null,'lunch',null),40);

1 row created.

SQL> insert into food values(21838,'Non-veg','Chickenidly',25,'Chicken,Rice,Urad
  2  dal',categ('breakfast','lunch','dinner'),45);

1 row created.

SQL> insert into food values(21934,'Non-veg','EggDosa',20,'egg,Rice
  2  flour',categ('breakfast',null,'dinner'),30);

1 row created.
```

```
SQL> insert into food values(22893,'Non-veg','fishcurry',45,'Fish,curryleaves,chilli,Curry
  2  powder',categ('breakfast','lunch','dinner'),20);

1 row created.

SQL> insert into food values(29834,'Non-veg','Chicken Pakoda',30,'chicken,cheda dal flour,rice
  2  flour',categ('breakfast','lunch',null),25);

1 row created.

SQL> insert into food values(22444,'Non-veg','Chickenpepper
  2  fry',60,'chicken,pepper',categ(null,null,'dinner'),20);

1 row created.

SQL> insert into food values(27373,'Non-veg','Non-veg Meals',120,'combo of various non-veg
  2  foods',categ(null,'lunch','dinner'),50);

1 row created.

SQL>
```

```
SQL> insert into food values(23783,'Non-veg','Chicken Briyani',90,'Chicken,Basmati rice,Garam
  2 masala,chili',categ(null,'lunch',null),40);

1 row created.

SQL> insert into food values(29323,'Non-veg','Eggbriyani',80,'Egg,Basmatirice,Garam
  2 masala,chili',categ(null,'lunch',null),43);

1 row created.
```

RESTAURANT:

insert into restaurant values(100000,'South king Spice','Good food and great Vibes',4.1,'Spencer Plaza','Anna salai');

insert into restaurant values(123892,'Indian Curry','A taste of Home',4.5,'N S C Bose road','Parrys');

insert into restaurant values(398113,'Meal Cage','Every meal is a Happy meal',3.9,'Rythams Arcade','Eldams Road');

insert into restaurant values(742921,'Pattero Kitchen','Eat.Dring.Love',4.3,'Vannier Street','Choolaimedu');

insert into restaurant values(234894,'Indian Accent','Nurture through nature',4.6, 'Post Office Street','Parrys');

```
SQL> insert into restaurant values(100000,'South king Spice','Good food and great Vibes',4.1,'Spencer
  2 Plaza','Anna salai');

1 row created.

SQL> insert into restaurant values(123892,'Indian Curry','A taste of Home',4.5,'N S C Bose road','Parrys');

1 row created.

SQL> insert into restaurant values(398113,'Meal Cage','Every meal is a Happy meal',3.9,'Rythams
  2 Arcade','Eldams Road');

1 row created.

SQL> insert into restaurant values(742921,'Pattero Kitchen','Eat.Dring.Love',4.3,'Vannier Street','Choolaimedu');

1 row created.

SQL> insert into restaurant values(234894,'Indian Accent','Nurture through nature',4.6, 'Post Office Street','Parrys');

1 row created.
```

BELONGS:

insert into belongs values(100000,11111);

insert into belongs values(100000,12993);

insert into belongs values(100000,12789);

insert into belongs values(100000,11283);

insert into belongs values(100000,12132);

insert into belongs values(100000,12243);

insert into belongs values(100000,21838);

insert into belongs values(100000,22893);

insert into belongs values(100000,22444);

insert into belongs values(100000,27373);

insert into belongs values(123892,11111);

insert into belongs values(123892,12993);

insert into belongs values(123892,12847);

insert into belongs values(123892,12789);

insert into belongs values(123892,11993);

insert into belongs values(123892,12243);

insert into belongs values(123892,12446);

insert into belongs values(398113,11111);

insert into belongs values(398113,12993);

insert into belongs values(398113,12789);

insert into belongs values(398113,12243);

insert into belongs values(398113,12446);

insert into belongs values(398113,27373);

insert into belongs values(398113,23783);

insert into belongs values(742921,11111);

insert into belongs values(742921,12789);

insert into belongs values(742921,12132);
insert into belongs values(742921,21934);
insert into belongs values(742921,29834);
insert into belongs values(742921,22444);
insert into belongs values(742921,29323);
insert into belongs values(234894,11111);
insert into belongs values(234894,12993);
insert into belongs values(234894,12789);
insert into belongs values(234894,11283);
insert into belongs values(234894,12132);
insert into belongs values(234894,12243);
insert into belongs values(234894,12446);
insert into belongs values(234894,27373);
insert into belongs values(234894,23783);

```
SQL> insert into belongs values(100000,11111);  
1 row created.  
SQL> insert into belongs values(100000,12993);  
1 row created.  
SQL> insert into belongs values(100000,12789);  
1 row created.  
SQL> insert into belongs values(100000,11283);  
1 row created.  
SQL> insert into belongs values(100000,12132);  
1 row created.  
SQL> insert into belongs values(100000,12243);  
1 row created.  
SQL> insert into belongs values(100000,21838);  
1 row created.  
SQL> insert into belongs values(100000,22893);  
1 row created.  
SQL> insert into belongs values(100000,22444);  
1 row created.
```

```
SQL> insert into belongs values(100000,27373);  
1 row created.  
  
SQL> insert into belongs values(123892,11111);  
1 row created.  
  
SQL> insert into belongs values(123892,12993);  
1 row created.  
  
SQL> insert into belongs values(123892,12847);  
1 row created.  
  
SQL> insert into belongs values(123892,12789);  
1 row created.  
  
SQL> insert into belongs values(123892,11993);  
1 row created.  
  
SQL> insert into belongs values(123892,12243);  
1 row created.  
  
SQL> insert into belongs values(123892,12446);  
1 row created.  
  
SQL> insert into belongs values(398113,11111);  
1 row created.  
  
SQL> insert into belongs values(398113,12993);  
1 row created.
```



```
SQL> insert into belongs values(398113,12789);
```

```
1 row created.
```

```
SQL> insert into belongs values(398113,12243);
```

```
1 row created.
```

```
SQL> insert into belongs values(398113,12446);
```

```
1 row created.
```

```
SQL> insert into belongs values(398113,27373);
```

```
1 row created.
```

```
SQL> insert into belongs values(398113,23783);
```

```
1 row created.
```

```
SQL> insert into belongs values(742921,11111);
```

```
1 row created.
```

```
SQL> insert into belongs values(742921,12789);
```

```
1 row created.
```

```
SQL> insert into belongs values(742921,12132);
```

```
1 row created.
```

```
SQL> insert into belongs values(742921,21934);
```

```
1 row created.
```

```
SQL> insert into belongs values(742921,29834);
```

```
1 row created.
```

```
SQL> insert into belongs values(742921,22444);
```

```
1 row created.
```

```
SQL> insert into belongs values(234894,11111);
```

```
1 row created.
```

```
SQL> insert into belongs values(234894,12993);
```

```
1 row created.
```

```
SQL> insert into belongs values(234894,12789);
```

```
1 row created.
```

```
SQL> insert into belongs values(234894,11283);
```

```
1 row created.
```

```
SQL> insert into belongs values(234894,12132);
```

```
1 row created.
```

```
SQL> insert into belongs values(234894,12243);
```

```
1 row created.
```

```
SQL> insert into belongs values(234894,12446);
```

```
1 row created.
```

```
SQL> insert into belongs values(234894,27373);
```

```
1 row created.
```

```
SQL> insert into belongs values(234894,23783);
```

```
1 row created.
```

```
SQL>
```

PAYMENT:

insert into payment values(13839,155,'HDFC','LIAM V',DATE '2020-12-12','Confirmed');

insert into payment values(17284,120,'ICICI','ISABELLA H',DATE '2022-11-24','Confirmed');

insert into payment values(19274,220,'SBI','OLIVIA P',DATE '2021-06-07','Confirmed');

insert into payment values(19428,60,'SBI','OLIVIA P',DATE '2021-06-07','Pending');

```
SQL> insert into payment values(13839,155,'HDFC','LIAM V',DATE '2020-12-12','Confirmed');
1 row created.

SQL> insert into payment values(17284,120,'ICICI','ISABELLA H',DATE '2022-11-24','Confirmed');
1 row created.

SQL> insert into payment values(19274,220,'SBI','OLIVIA P',DATE '2021-06-07','Confirmed');
1 row created.

SQL> insert into payment values(19428,60,'SBI','OLIVIA P',DATE '2021-06-07','Pending');
1 row created.
```

ORDERS:

insert into orders values(67534,1,11111,13839,3,45);

insert into orders values(67534,2,12789,13839,1,30);

insert into orders values(67534,3,12132,13839,2,40);

insert into orders values(32794,4,12132,19274,4,80);

insert into orders values(32794,5,29834,19274,2,60);

insert into orders values(32794,6,22444,19274,1,80);

insert into orders values(86386,7,12993,17284,4,60);

insert into orders values(86386,8,12789,17284,2,60);

insert into orders values(32794,9,29323,19428,1,60);

```
SQL> insert into orders values(67534,1,11111,13839,3,45);
1 row created.

SQL> insert into orders values(67534,2,12789,13839,1,30);
1 row created.

SQL> insert into orders values(67534,3,12132,13839,2,40);
1 row created.

SQL> insert into orders values(32794,4,12132,19274,4,80);
1 row created.

SQL> insert into orders values(32794,5,29834,19274,2,60);
1 row created.

SQL> insert into orders values(32794,6,22444,19274,1,80);
1 row created.

SQL> insert into orders values(86386,7,12993,17284,4,60);
1 row created.

SQL> insert into orders values(86386,8,12789,17284,2,60);
1 row created.

SQL> insert into orders values(32794,9,29323,19428,1,60);
1 row created.
```

DELIVERY:

```
insert into delivery values(1,67534,to_timestamp('07:30','hh24:mi'),16);
insert into delivery values(2,67534,to_timestamp('07:30','hh24:mi'),16);
insert into delivery values(3,67534,to_timestamp('07:30','hh24:mi'),16);
insert into delivery values(4,32794,to_timestamp('14:00','hh24:mi'),23);
insert into delivery values(5,32794,to_timestamp('14:00','hh24:mi'),23);
insert into delivery values(6,32794,to_timestamp('14:00','hh24:mi'),23);
insert into delivery values(7,86386,to_timestamp('14:00','hh24:mi'),14);
insert into delivery values(8,86386,to_timestamp('14:00','hh24:mi'),14);
```

```
SQL> insert into delivery values(1,67534,to_timestamp('07:30','hh24:mi'),16);
1 row created.

SQL> insert into delivery values(2,67534,to_timestamp('07:30','hh24:mi'),16);
1 row created.

SQL> insert into delivery values(3,67534,to_timestamp('07:30','hh24:mi'),16);
1 row created.

SQL> insert into delivery values(4,32794,to_timestamp('14:00','hh24:mi'),23);
1 row created.

SQL> insert into delivery values(5,32794,to_timestamp('14:00','hh24:mi'),23);
1 row created.

SQL> insert into delivery values(6,32794,to_timestamp('14:00','hh24:mi'),23);
1 row created.

SQL> insert into delivery values(7,86386,to_timestamp('14:00','hh24:mi'),14);
1 row created.

SQL> insert into delivery values(8,86386,to_timestamp('14:00','hh24:mi'),14);
1 row created.

SQL>
```

FOOD_IMAGE:

```
insert into food_image values(11111,'Idly','idly.jpg');

insert into food_image values(12993,'Aapam','Aapam.jpg');

insert into food_image values(12847,'Chola Poori','Chola Poori.jpg');

insert into food_image values(12789,'Utthapam','Utthapam.jpg');

insert into food_image values(11993,'Onion Pakode','Onion Pakode.jpg');

insert into food_image values(11283,'Paneer fry','Paneer fry.jpg');

insert into food_image values(12132,'Chappati','Chappati.jpg');

insert into food_image values(12243,'Veg Meals','Veg Meals.jpg');

insert into food_image values(12446,'Veg Biryani','Veg Biryani.jpg');
```

```
insert into food_image values(21838,'Chicken idly','Chicken idly.jpg');

insert into food_image values(21934,'Egg Dosa','Egg Dosa.jpg');

insert into food_image values(22893,'fish curry','fish curry.jpg');

insert into food_image values(29834,'Chicken pakode','Chicken pakode.jpg');

insert into food_image values(22444,'Chicken pepper fry','Chicken pepper fry.jpg');

insert into food_image values(27373,'Non-Veg Meals','Non-Veg Meals.jpg');

insert into food_image values(23783,'Chicken Briyani','Chicken Briyani.jpg');

insert into food_image values(29323,'Egg Biryani','Egg Biryani.jpg');
```

```
SQL> insert into food_image values(11111,'Idly','idly.jpg');
1 row created.

SQL> insert into food_image values(12993,'Aapam','Aapam.jpg');
1 row created.
```

```
SQL> insert into food_image values(12132,'Chappati','Chappati.jpg');
1 row created.

SQL> insert into food_image values(12243,'Veg Meals','Veg Meals.jpg');
1 row created.
```

```
SQL> insert into food_image values(12446,'Veg Biryani','Veg Biryani.jpg');
1 row created.

SQL> insert into food_image values(21838,'Chicken idly','Chicken idly.jpg');
1 row created.

SQL> insert into food_image values(21934,'Egg Dosa','Egg Dosa.jpg');
1 row created.

SQL> insert into food_image values(22893,'fish curry','fish curry.jpg');
1 row created.
```

```
SQL> insert into food_image values(12847,'Chola Poori','Chola Poori.jpg');
1 row created.

SQL> insert into food_image values(12789,'Utthapam','Utthapam.jpg');
1 row created.

SQL> insert into food_image values(11993,'Onion Pakode','Onion Pakode.jpg');
1 row created.

SQL> insert into food_image values(11283,'Paneer fry','Paneer fry.jpg');
1 row created.
```

```
SQL> insert into food_image values(29834,'Chicken pakode','Chicken pakode.jpg');
1 row created.

SQL> insert into food_image values(22444,'Chicken pepper fry','Chicken pepper fry.jpg');
1 row created.

SQL> insert into food_image values(27373,'Non-Veg Meals','Non-Veg Meals.jpg');
1 row created.

SQL> insert into food_image values(23783,'Chicken Briyani','Chicken Briyani.jpg');
1 row created.

SQL> insert into food_image values(29323,'Egg Biryani','Egg Biryani.jpg');
1 row created.
```

DISPLAY TABLES:

user_info:

```
SQL> select *from user_info;
```

USER_ID	NAME	DOB	EMAIL
PHONE	DOOR_NO	STREET	REGION

67534	Liam	20-MAY-96	liam123@gamil.com
8976835689	2	EveningBazzard	ParkTown
Chennai			

95674	Noah	12-FEB-89	Noah_678@gamil.com
9845298640	17	BPC Plaza	T nagar
Chennai			

USER_ID	NAME	DOB	EMAIL
PHONE	DOOR_NO	STREET	REGION

86386	Isabella	04-SEP-99	Isa_bella@gamil.com
7849378597	167/2 F12	Eldams Square	Alwarpet
Chennai			

78493	William	15-NOV-00	Will15050@gamil.com
8674903980	1	BabaTowers Sterling	Nungambakkam

USER_ID	NAME	DOB	EMAIL
PHONE	DOOR_NO	STREET	REGION

32794	Olivia	19-JAN-97	Olivia20@gamil.com
9674536896	84	Audiappa Naikan st	Sowcarpet
Chennai			

Login:

```
SQL> select *from login;
```

USER_ID	USERNAME	PASSWORD
67534	Liam123	iamliam
95674	Noah678	xyznoah
86386	Isabella12	isa12949
78493	William50	\$will\$
32794	Olivia20	greenearth

```
SQL>
```

Orders:

```
SQL> select *from orders;
```

USER_ID	ORDER_NO	FOOD_ID	PAY_ID	QUANTITY	T_PRICE
67534	1	11111	13839	3	45
67534	2	12789	13839	1	30
67534	3	12132	13839	2	40
32794	4	12132	19274	4	80
32794	5	29834	19274	2	60
32794	6	22444	19274	1	80
86386	7	12993	17284	4	60
86386	8	12789	17284	2	60
32794	9	29323	19428	1	60

```
9 rows selected.
```

Food:

```
SQL> select *from food;
```

FOOD_ID	TYPE	F_NAME	PRICE

DESCRIPTION			

CATEGORY			

AVAILABILITY			

11111	Veg	Idly	15
Rice,Urad dal			
CATEG('breakfast', 'lunch', 'dinner')			
50			

FOOD_ID	TYPE	F_NAME	PRICE

DESCRIPTION			

CATEGORY			

AVAILABILITY			

12993	Veg	Aapam	15
Rice,Batter			
CATEG('breakfast', 'lunch', 'dinner')			
30			

FOOD_ID	TYPE	F_NAME	PRICE

DESCRIPTION			

CATEGORY			

AVAILABILITY			

12847	Veg	CholaPoori	50
Maida,Chick			
peas,Garlic,Tomato			
CATEG('breakfast', NULL, 'dinner')			
20			

FOOD_ID	TYPE	F_NAME	PRICE

DESCRIPTION			

CATEGORY			

AVAILABILITY

12789 Veg Uttchapam 30
Onion,Rice,Uraddal,White Lentils
CATEG('breakfast', NULL, 'dinner')
25

FOOD_ID	TYPE	F_NAME	PRICE
---------	------	--------	-------

DESCRIPTION

CATEGORY

AVAILABILITY

11993 Veg OnionPakoda 20
Chanadalflour,rice
flour,ginger
CATEG('breakfast', 'lunch', NULL)

FOOD_ID	TYPE	F_NAME	PRICE
---------	------	--------	-------

DESCRIPTION

CATEGORY

AVAILABILITY

30

11283 Veg Paneerfry 60
Paneer,Garam
masala,chilli,tomato

FOOD_ID	TYPE	F_NAME	PRICE
---------	------	--------	-------

DESCRIPTION

CATEGORY

AVAILABILITY

CATEG(NULL, 'lunch', 'dinner')
15

12132 Veg chappati 20
wheat

FOOD_ID	TYPE	F_NAME	PRICE

DESCRIPTION			

CATEGORY			

AVAILABILITY			

flour,water			
CATEG('breakfast', 'lunch', 'dinner')			
60			
12243	Veg	VegMeals	90
FOOD_ID	TYPE	F_NAME	PRICE

DESCRIPTION			

CATEGORY			

AVAILABILITY			

Comboofvariousveg			
foods			
CATEG(NULL, 'lunch', NULL)			
50			
FOOD_ID	TYPE	F_NAME	PRICE

DESCRIPTION			

CATEGORY			

AVAILABILITY			

12446	Veg	VegBriyani	70
Basmatirice,Garam			
masala,chili,Peas			
CATEG(NULL, 'lunch', NULL)			
40			
FOOD_ID	TYPE	F_NAME	PRICE

DESCRIPTION			

CATEGORY			

AVAILABILITY			

```

-----
      21838 Non-veg    Chickenidly          25
Chicken,Rice,Urad
dal
CATEG('breakfast', 'lunch', 'dinner')

      FOOD_ID TYPE      F_NAME          PRICE
-----
DESCRIPTION
-----
CATEGORY
-----
AVAILABILITY
-----
45

      21934 Non-veg    EggDosa              20
egg,Rice
flour

      FOOD_ID TYPE      F_NAME          PRICE
-----
DESCRIPTION
-----
CATEGORY
-----
AVAILABILITY
-----
CATEG('breakfast', NULL, 'dinner')
30

      22893 Non-veg    fishcurry            45
Fish,curryleaves,chilli,Curry

      FOOD_ID TYPE      F_NAME          PRICE
-----
DESCRIPTION
-----
CATEGORY
-----
AVAILABILITY
-----
powder
CATEG('breakfast', 'lunch', 'dinner')
20

      29834 Non-veg    Chicken Pakoda       30

      FOOD_ID TYPE      F_NAME          PRICE

```

FOOD_ID	TYPE	F_NAME	PRICE

DESCRIPTION			

CATEGORY			

AVAILABILITY			

chicken, chana dal flour, rice			
flour			
CATEG('breakfast', 'lunch', NULL)			
25			
FOOD_ID	TYPE	F_NAME	PRICE

DESCRIPTION			

CATEGORY			

AVAILABILITY			

2244	Non-veg	Chickenpepper fry	60
chicken, pepper			
CATEG(NULL, NULL, 'dinner')			
20			
FOOD_ID	TYPE	F_NAME	PRICE

DESCRIPTION			

CATEGORY			

AVAILABILITY			

2737	Non-veg	Non-veg Meals	120
combo of various non-veg			
foods			
CATEG(NULL, 'lunch', 'dinner')			
FOOD_ID	TYPE	F_NAME	PRICE

DESCRIPTION			

CATEGORY			

AVAILABILITY			

```
-----
50
      23783 Non-veg    Chicken Briyani          90
Chicken,Basmati rice,Garam
masala,chili

      FOOD_ID TYPE      F_NAME          PRICE
-----
DESCRIPTION
-----
CATEGORY
-----
AVAILABILITY
-----
CATEG(NULL, 'lunch', NULL)
40

      29323 Non-veg    Eggbriyani          80
Egg,Basmatirice,Garam

      FOOD_ID TYPE      F_NAME          PRICE
-----
DESCRIPTION
-----
CATEGORY
-----
AVAILABILITY
-----
masala,chili
CATEG(NULL, 'lunch', NULL)
43

17 rows selected.

SQL>
```

Restaurant:

```
SQL> select *from restaurant;
```

R_ID	NAME	DESCRIPTION	RATING
100000	South king Spice	Good food and great Vibes	4.1
	Spencer Plaza	Anna salai	

123892	Indian Curry	A taste of Home	4.5
--------	--------------	-----------------	-----

R_ID	NAME	DESCRIPTION	RATING
	N S C Bose road	Parrys	
398113	Meal Cage	Every meal is a Happy meal	3.9
	Rythams Arcade	Eldams Road	

R_ID	NAME	DESCRIPTION	RATING
742921	Pattero Kitchen	Eat.Dring.Love	4.3
	Vannier Street	Choolaimedu	
234894	Indian Accent	Nurture through nature	4.6
	Post Office Street	Parrys	

Belongs:

```
SQL> select *from belongs;
```

R_ID	FOOD_ID
100000	11111
100000	12993
100000	12789
100000	11283
100000	12132
100000	12243
100000	21838
100000	22893
100000	22444
100000	27373
123892	11111

R_ID	FOOD_ID
123892	12993
123892	12847
123892	12789
123892	11993
123892	12243
123892	12446
398113	11111
398113	12993
398113	12789
398113	12243
398113	12446

R_ID	FOOD_ID
398113	27373
398113	23783
742921	11111
742921	12789
742921	12132
742921	21934
742921	29834
742921	22444
234894	11111
234894	12993
234894	12789

R_ID	FOOD_ID
234894	11283
234894	12132
234894	12243
234894	12446
234894	27373
234894	23783

```
39 rows selected.
```

```
SQL>
```

Payment:

```
SQL> select *from payment;
```

PAY_ID	TOTAL	BANK	CARD_HOLDER_NAME
13839	155	HDFC	LIAM V
12-DEC-20	Confirmed		

17284	120	ICICI	ISABELLA H
24-NOV-22	Confirmed		

19274	220	SBI	OLIVIA P
07-JUN-21	Confirmed		

PAY_ID	TOTAL	BANK	CARD_HOLDER_NAME
19428	60	SBI	OLIVIA P
07-JUN-21	Pending		

Delivery:

```
SQL> select *from delivery;
```

ORDER_NO	USER_ID
1	67534
01-MAY-21 07.30.00 AM	
16	

2	67534
01-MAY-21 07.30.00 AM	
16	

ORDER_NO	USER_ID
3	67534
01-MAY-21 07.30.00 AM	
16	

4	32794
01-MAY-21 02.00.00 PM	

ORDER_NO	USER_ID
5	32794
01-MAY-21 02.00.00 PM	
23	
6	32794

ORDER_NO	USER_ID
7	86386
01-MAY-21 02.00.00 PM	
14	

ORDER_NO	USER_ID
8	86386
01-MAY-21 02.00.00 PM	
14	

```
8 rows selected.
```

```
SQL>
```

Food_Image:

```
SQL> select *from food_image;
```

FOOD_ID	F_NAME	IMAGE_NAME
---------	--------	------------

11111	Idly	idly.jpg
-------	------	----------

12993	Aapam	Aapam.jpg
-------	-------	-----------

12847	Chola Poori	Chola Poori.jpg
-------	-------------	-----------------

FOOD_ID	F_NAME	IMAGE_NAME
---------	--------	------------

12789	Utthapam	Utthapam.jpg
-------	----------	--------------

11993	Onion Pakode	Onion Pakode.jpg
-------	--------------	------------------

11283	Paneer fry	Paneer fry.jpg
-------	------------	----------------

FOOD_ID	F_NAME	IMAGE_NAME
---------	--------	------------

12132	Chappati	Chappati.jpg
-------	----------	--------------

12243	Veg Meals	Veg Meals.jpg
-------	-----------	---------------

12446	Veg Biryani	Veg Biryani.jpg
-------	-------------	-----------------

FOOD_ID	F_NAME	IMAGE_NAME
---------	--------	------------

21838	Chicken idly	
-------	--------------	--

```
Chicken idly.jpg

      21934 Egg Dosa
Egg Dosa.jpg

      22893 fish curry
fish curry.jpg

      FOOD_ID F_NAME
-----
IMAGE_NAME
-----

      29834 Chicken pakode
Chicken pakode.jpg

      22444 Chicken pepper fry
Chicken pepper fry.jpg

      27373 Non-Veg Meals
Non-Veg Meals.jpg

      FOOD_ID F_NAME
-----
IMAGE_NAME
-----

      23783 Chicken Briyani
Chicken Briyani.jpg

      29323 Egg Biryani
Egg Biryani.jpg

17 rows selected.

SQL>
```

Thank You!