**A logo of university of gujrat

Description automatically generated**

**Database Project**

**Submitted by:**

Saria Rukhsar 22024119-029

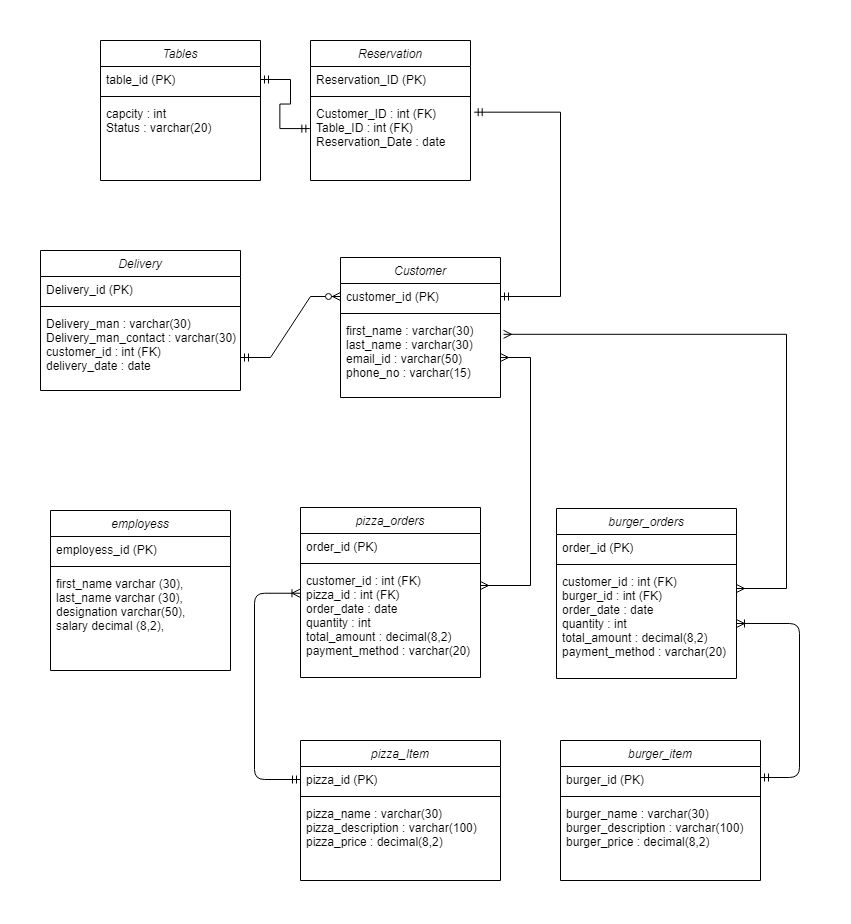
**Submission Date:**

27-06-2024

**Submitted To:**

Sir Naveed Anwar But

**ERD:**

****

**Create Database:**

create database fast\_food\_deleivery

**Create Tables and Insert Values:**

create table customer(

customer\_id int primary key,

first\_name varchar (30),

last\_name varchar (30),

email\_id varchar (50),

phone\_no varchar (15)

);

insert into customer values

(1,'Saria','Rukhsar','sariarukhsar5@gmail.com','0341-6451972'),

(2,'Umer','Ilyas','ui14@gmail.com','0304-403253'),

(3,'Ahmed','Shahbaz','ams9@gmail.com','0312-3467113'),

(4,'Huzaifa','Adil','hza12@gmail.com','0315-4067113'),

(5,'Haroon','Ali','hra19@gmail.com','0300-4573113');

create table pizza\_item(

pizza\_id int primary key,

pizza\_name varchar (30),

pizza\_description varchar (100),

pizza\_type varchar (10),

pizza\_price decimal (8,2),

);

insert into pizza\_item

values(1,'royal crust pizza','classic chicken chesse pizza','L', 2000.00);

insert into pizza\_item

values(2,'royal cheese pizza','classic chicken chesse pizza','L', 2500.00);

insert into pizza\_item

values(3,'crown crust pizza','classic chicken chesse pizza','M', 1500.00);

insert into pizza\_item

values(4,'malai crust pizza','classic chicken chesse pizza','L', 2000.00);

insert into pizza\_item

values(5,'fajita pizza','classic chicken chesse pizza','S', 1000.00);

create table burger\_item(

burger\_id int primary key,

burger\_name varchar (30),

burger\_description varchar (100),

burger\_price decimal (8,2)

);

insert into burger\_item

values(1,'royal crust burger','classic chicken chesse burger', 500.00);

insert into burger\_item

values(2,'royal cheese burger','classic chicken chesse burger', 700.00);

insert into burger\_item

values(3,'crown crust burger','classic chicken chesse burger', 300.00);

insert into burger\_item

values(4,'malai crust burger','classic chicken chesse burger', 400.00);

insert into burger\_item

values(5,'fajita burger','classic chicken chesse burger', 500.00);

create table pizza\_orders(

order\_id int primary key,

customer\_id int,

pizza\_id int,

order\_date date,

quantity int,

total\_amount decimal (8,2),

payment\_method varchar(20),

foreign key (customer\_id) references customer(customer\_id),

foreign key (pizza\_id) references pizza\_item (pizza\_id),

);

insert into pizza\_orders values

(1,1,2,'2023-09-12',1,2500.00 , 'Cash'),

(2,1,3,'2023-09-14',3,4500.00 , 'Credit Card'),

(3,3,2,'2023-09-13',2,5000.00 , 'Cash'),

(4,4,1,'2023-09-15',1,2000.00 , 'Cash'),

(5,2,5,'2023-09-12',1,1000.00 , 'Debit Card'),

(6,4,5,'2023-09-11',2,2000.00 , 'Debit Card');

create table burger\_orders(

order\_id int primary key,

customer\_id int,

burger\_id int,

order\_date date,

quantity int,

total\_amount decimal (8,2),

payment\_method varchar(20),

foreign key (customer\_id) references customer(customer\_id),

foreign key (burger\_id) references burger\_item (burger\_id),

);

insert into burger\_orders values

(1,1,2,'2023-09-12',1,700.00 , 'Cash'),

(2,6,3,'2023-09-12',2,900.00 , 'Cash'),

(3,7,2,'2023-09-12',6,4200.00 , 'Debit Card'),

(4,5,1,'2023-09-15',4,2000.00 , 'Credit Card'),

(5,2,4,'2023-09-12',1,400.00 , 'Cash'),

(6,4,5,'2023-09-11',2,1000.00 , 'Cash');

Create table Tables (

table\_id int primary key,

capcity int,

Status varchar(20),

);

Insert into Tables Values

(101,6,'Reserved'),

(102,4,'Reserved'),

(103,6,'Available'),

(104,2,'Available'),

(105,4,'Reserved');

Create Table Reservation (

Reservation\_ID int primary key,

Customer\_ID int,

Table\_ID int,

Reservation\_Date date,

Foreign key (Customer\_ID) references customer(customer\_id),

Foreign key (Table\_ID) references Tables (table\_id),

);

Insert into Reservation Values

(1,5,101,'2023-07-09'),

(2,3,102,'2023-07-09'),

(3,9,105,'2023-07-10');

create table employees(

employees\_id int primary key,

first\_name varchar (30),

last\_name varchar (30),

designation varchar(50),

salary decimal (8,2),

);

insert into employees

values(1,'saad','ali','Cashier',5000.00);

insert into employees

values(2,'asad','ali','Cook',20000.00);

insert into employees

values(3,'saud','ali','Cook',50000.00);

insert into employees

values(4,'zaid','ali','Manager',45500.00);

insert into employees

values(5,'shakeel','ali','Waiter',500.00);

Create Table Delivery (

Delivery\_id int primary key,

Delivery\_man varchar (30),

Delivery\_man\_contact varchar(30),

customer\_id int,

delivery\_date date,

foreign key (customer\_id) references customer(customer\_id)

);

insert into Delivery values

(1, 'Ali' , '0332-4582701' , 2 , '2021-08-14'),

(2, 'Ahsan', '0341-5509011' , 4 , '2021-01-19'),

(3, 'Saad', '0341-5509341' , 6 , '2021-01-29');

**Procedures and Query’s:**

--Select all records from the customer table:

SELECT \* FROM customer;

--Select all records from the pizza\_item table:

SELECT \* FROM pizza\_item;

--Select all records from the burger\_item table:

SELECT \* FROM burger\_item;

--Select all records from the pizza\_orders table:

SELECT \* FROM pizza\_orders;

--Select all records from the burger\_orders table:

SELECT \* FROM burger\_orders;

--Select all records from the Tables table:

SELECT \* FROM Tables;

--Select all records from the Reservation table:

SELECT \* FROM Reservation;

--Select all records from the employees table:

SELECT \* FROM employees;

--Select the first name and last name of all customers:

SELECT first\_name, last\_name FROM customer;

--Select the pizza names and their prices:

SELECT pizza\_name, pizza\_price FROM pizza\_item;

--Select the burger names and their prices:

SELECT burger\_name, burger\_price FROM burger\_item;

--Select the order\_id, customer\_id, and total\_amount from the pizza\_orders table:

SELECT order\_id, customer\_id, total\_amount FROM pizza\_orders;

--Select the order\_id, customer\_id, and total\_amount from the burger\_orders table:

SELECT order\_id, customer\_id, total\_amount FROM burger\_orders;

--Select the table\_id and capacity of all tables:

SELECT table\_id, capcity FROM Tables;

--Select the reservation\_id, customer\_id, and reservation\_date from the Reservation table:

SELECT Reservation\_ID, Customer\_ID, Reservation\_Date FROM Reservation;

--Select the first name, last name, and salary of all employees:

SELECT first\_name, last\_name, salary FROM employees;

--Count the total number of customers in the customer table:

SELECT COUNT(customer\_id) AS total\_customers FROM customer;

--Calculate the average pizza price:

SELECT AVG(pizza\_price) AS average\_pizza\_price FROM pizza\_item;

--Calculate the total amount spent by each customer in pizza orders:

SELECT customer\_id, SUM(total\_amount) AS total\_spent FROM pizza\_orders GROUP BY customer\_id;

--Find the highest pizza price:

SELECT MAX(pizza\_price) AS highest\_pizza\_price FROM pizza\_item;

--Find the lowest pizza price:

SELECT MIN(pizza\_price) AS lowest\_pizza\_price FROM pizza\_item;

--Find the customer who spent the most on burger orders:

SELECT top 1 customer\_id, MAX(total\_amount) AS max\_spent FROM burger\_orders GROUP BY customer\_id ORDER BY max\_spent DESC;

--Find the customer who placed the first pizza order:

SELECT top 1 customer\_id, MIN(order\_date) AS first\_pizza\_order FROM pizza\_orders GROUP BY customer\_id ORDER BY first\_pizza\_order ASC;

--Calculate the total earnings from pizza orders:

SELECT SUM(total\_amount) AS total\_earnings FROM pizza\_orders;

--Calculate the total earnings from burger orders:

SELECT SUM(total\_amount) AS total\_earnings FROM burger\_orders;

--Find the busiest day with the most pizza orders:

SELECT top 1 order\_date, COUNT(order\_id) AS pizza\_orders\_count FROM pizza\_orders GROUP BY order\_date ORDER BY pizza\_orders\_count DESC;

--Find the number of available tables:

SELECT COUNT(table\_id) AS available\_tables FROM Tables WHERE Status = 'Available';

--FIND NUMBER OF RESEVERD TABLE

SELECT COUNT(table\_id) AS reserved\_tables FROM Tables WHERE Status = 'Reserved';

--Retrieve all delivery records:

SELECT \* FROM Delivery;

--Retrieve delivery details for a specific delivery ID:

SELECT \* FROM Delivery WHERE Delivery\_id = 1;

--Retrieve delivery details for a specific customer ID:

SELECT \* FROM Delivery WHERE customer\_id = 2;

--Retrieve delivery records for a specific delivery date:

SELECT \* FROM Delivery WHERE delivery\_date = '2021-08-09';

--Retrieve deliveries along with customer information for a specific delivery man

SELECT Delivery.Delivery\_id,Delivery.Delivery\_man,Delivery.Delivery\_man\_contact,Delivery.delivery\_date,Delivery.customer\_id, customer.first\_name, customer.last\_name, customer.phone\_no

FROM Delivery

inner JOIN customer ON Delivery.customer\_id = customer.customer\_id

WHERE Delivery\_man = 'Ali';

--Retrieve the count of deliveries for each delivery man:

SELECT Delivery\_man, COUNT(\*) AS delivery\_count

FROM Delivery

GROUP BY Delivery\_man;

--Update the delivery date for a specific delivery ID:

select \* from Delivery UPDATE Delivery

SET delivery\_date = '2021-08-15'

WHERE Delivery\_id = 2;

--Retrieve all pizza orders with customer information:

CREATE PROCEDURE PIZZA\_ORDER\_INFO

AS

BEGIN

SELECT pizza\_orders.order\_id, pizza\_orders.order\_date, pizza\_orders.total\_amount, pizza\_orders.payment\_method, customer.first\_name, customer.last\_name, customer.email\_id,

customer.phone\_no

FROM pizza\_orders

inner JOIN customer ON pizza\_orders.customer\_id = customer.customer\_id;

END;

exec PIZZA\_ORDER\_INFO

--Retrieve all burger orders with customer information:

SELECT burger\_orders.order\_id, burger\_orders.order\_date, burger\_orders.total\_amount, burger\_orders.payment\_method, customer.first\_name, customer.last\_name, customer.email\_id,

customer.phone\_no

FROM burger\_orders

inner JOIN customer ON burger\_orders.customer\_id = customer.customer\_id;

--Retrieve all pizza orders along with the pizza details:

SELECT pizza\_orders.order\_id, pizza\_item.pizza\_name, pizza\_item.pizza\_type, pizza\_item.pizza\_price, pizza\_orders.quantity, pizza\_orders.total\_amount

FROM pizza\_orders

inner JOIN pizza\_item ON pizza\_orders.pizza\_id = pizza\_item.pizza\_id;

--Retrieve all burger orders along with the burger details:

SELECT burger\_orders.order\_id, burger\_item.burger\_name, burger\_item.burger\_price, burger\_orders.quantity, burger\_orders.total\_amount

FROM burger\_orders

inner JOIN burger\_item ON burger\_orders.burger\_id = burger\_item.burger\_id;

--Retrieve all reserved tables along with the customer information who made the reservation:

SELECT Reservation.Reservation\_ID, Tables.table\_id, Tables.Status, customer.first\_name, customer.last\_name, customer.email\_id, customer.phone\_no

FROM Reservation

inner JOIN Tables ON Reservation.Table\_ID = Tables.table\_id

inner JOIN customer ON Reservation.Customer\_ID = customer.customer\_id;

--Retrieve all pizza orders with customer names and pizza details:

SELECT pizza\_orders.order\_id, customer.first\_name, customer.last\_name, pizza\_item.pizza\_name, pizza\_item.pizza\_type, pizza\_orders.quantity, pizza\_orders.total\_amount

FROM pizza\_orders

inner JOIN customer ON pizza\_orders.customer\_id = customer.customer\_id

inner JOIN pizza\_item ON pizza\_orders.pizza\_id = pizza\_item.pizza\_id;

--Retrieve all burger orders with customer names and burger details:

SELECT burger\_orders.order\_id, customer.first\_name, customer.last\_name, burger\_item.burger\_name, burger\_item.burger\_price, burger\_orders.quantity, burger\_orders.total\_amount

FROM burger\_orders

inner JOIN customer ON burger\_orders.customer\_id = customer.customer\_id

inner JOIN burger\_item ON burger\_orders.burger\_id = burger\_item.burger\_id;

--Retrieve the total number of pizza and burger orders placed by each customer:

SELECT customer.first\_name, customer.last\_name, COUNT(pizza\_orders.order\_id) AS pizza\_orders, COUNT(burger\_orders.order\_id) AS burger\_orders

FROM customer

LEFT JOIN pizza\_orders ON customer.customer\_id = pizza\_orders.customer\_id

LEFT JOIN burger\_orders ON customer.customer\_id = burger\_orders.customer\_id

GROUP BY customer.customer\_id, customer.first\_name, customer.last\_name;

--Retrieve the total sales amount for each pizza type:

SELECT pizza\_item.pizza\_type, SUM(pizza\_orders.total\_amount) AS total\_sales

FROM pizza\_orders

inner JOIN pizza\_item ON pizza\_item.pizza\_id = pizza\_item.pizza\_id

GROUP BY pizza\_item.pizza\_type;

--Retrieve the average salary of employees for each designation:

SELECT designation, AVG(salary) AS average\_salary

FROM employees

GROUP BY designation;

--Retrieve the count of available and reserved tables:

SELECT Status, COUNT(\*) AS count\_of\_tables

FROM Tables

GROUP BY Status;

--Retrieve all pizza orders along with the corresponding customer and table information (if reserved):

CREATE PROCEDURE PROC5

AS

BEGIN

SELECT pizza\_orders.order\_id, customer.first\_name, customer.last\_name, customer.email\_id, customer.phone\_no, Tables.table\_id, Tables.Status

FROM pizza\_orders

inner JOIN customer ON pizza\_orders.customer\_id = customer.customer\_id

LEFT JOIN Reservation ON pizza\_orders.customer\_id = Reservation.Customer\_ID

LEFT JOIN Tables ON Reservation.Table\_ID = Tables.table\_id;

END;

EXEC PROC5

--Retrieve all burger orders along with the corresponding customer and table information (if reserved):

CREATE PROCEDURE PROC4

AS

BEGIN

SELECT burger\_orders.order\_id, customer.first\_name, customer.last\_name, customer.email\_id, customer.phone\_no, Tables.table\_id, Tables.Status

FROM burger\_orders

inner JOIN customer ON burger\_orders.customer\_id = customer.customer\_id

LEFT JOIN Reservation ON burger\_orders.customer\_id = Reservation.Customer\_ID

LEFT JOIN Tables ON Reservation.Table\_ID = Tables.table\_id

END;

EXEC PROC4

--Retrieve the total number of pizza and burger orders served by each employee:

CREATE PROCEDURE PROC3

AS

BEGIN

SELECT employees.employees\_id, employees.first\_name, employees.last\_name, employees.designation,

COUNT(pizza\_orders.order\_id) AS pizza\_orders\_served, COUNT(burger\_orders.order\_id) AS burger\_orders\_served

FROM employees

LEFT JOIN pizza\_orders ON employees.employees\_id = pizza\_orders.customer\_id

LEFT JOIN burger\_orders ON employees.employees\_id = burger\_orders.customer\_id

GROUP BY employees.employees\_id, employees.first\_name, employees.last\_name, employees.designation

END;

EXEC PROC3

--Retrieve all pizza orders with delivery details (if delivered):

create procedure PROC2

AS

BEGIN

SELECT pizza\_orders.order\_id, customer.first\_name, customer.last\_name, customer.email\_id, customer.phone\_no, Delivery.Delivery\_man, Delivery.Delivery\_man\_contact,

Delivery.delivery\_date

FROM pizza\_orders

inner JOIN customer ON pizza\_orders.customer\_id = customer.customer\_id

LEFT JOIN Delivery ON pizza\_orders.customer\_id = Delivery.customer\_id;

END;

EXEC PROC2

--Retrieve all burger orders with delivery details (if delivered):

create procedure proc1

as

Begin

SELECT burger\_orders.order\_id, customer.first\_name, customer.last\_name, customer.email\_id, customer.phone\_no, Delivery.Delivery\_man, Delivery.Delivery\_man\_contact,

Delivery.delivery\_date

FROM burger\_orders

inner JOIN customer ON burger\_orders.customer\_id = customer.customer\_id

LEFT JOIN Delivery ON burger\_orders.customer\_id = Delivery.customer\_id ;

end;

exec proc1

select customer.first\_name,customer.last\_name,(count(pizza\_orders.quantity)+

count(burger\_orders.quantity)) as TOTAl from customer

left join pizza\_orders on customer.customer\_id= pizza\_orders.customer\_id

left join burger\_orders on burger\_orders.customer\_id = customer.customer\_id

where pizza\_orders.quantity>0 or burger\_orders.quantity>0

GROUP BY

customer.first\_name,customer.last\_name

;