ONLINE FOOD DELIVERY SYSTEM

-Radha Mahesh Pacharkar

Batch-T341/DS

ONLINE FOOD DELIVERY SYSTEM

The Online Food Delivery System is a comprehensive SQL-based project designed to simulate the core functionalities of a real-world food ordering platform. It manages various entities such as restaurants, menu items, customers, delivery partners, and orders within a structured relational database. The system allows customers to place orders, restaurants to list and manage their menus, and delivery partners to fulfill deliveries. It also supports order tracking, payment processing, and real-time status updates. This project showcases key SQL concepts including table creation, foreign key relationships, joins, aggregation, and subqueries, making it an ideal example for learning database design and management in an e-commerce or food-tech environment.

Project Aim

To build a relational SQL database for an online food delivery system.

- To manage restaurants, menu items, customers, orders, and delivery partners.
- To handle order placement, delivery tracking, and payments efficiently.
- To apply SQL concepts like joins, constraints, and subqueries.

OBJECTIVES

1. **Restaurant Management:** Store information about restaurants, cuisine types, and ratings.
2. **Menu Management:** Track available food items, prices, and availability.
3. **Customer Management:** Maintain user profiles and order history.
4. **Order Management:** Monitor orders, order items, and assign delivery partners.
5. **Delivery Partner Management:** Track delivery agents, vehicles, and contact details.
6. **Payments:** Manage billing, payment dates, amounts, and modes.

ER Diagram For Online Food Delivery System

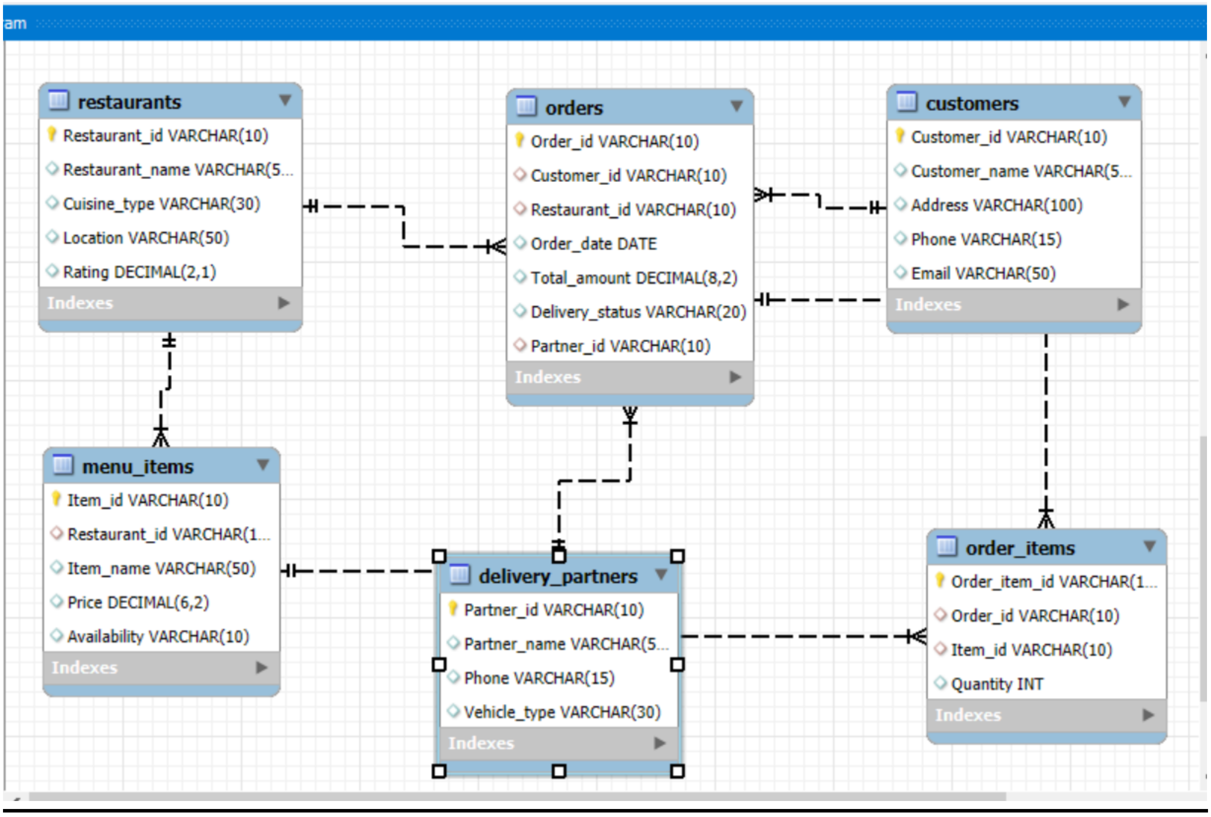
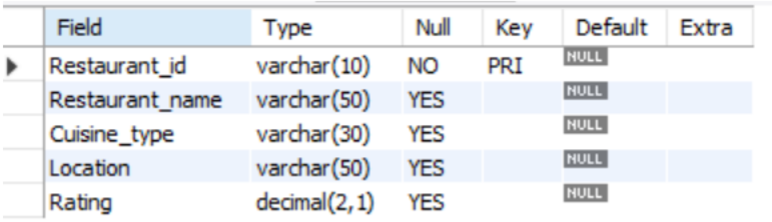


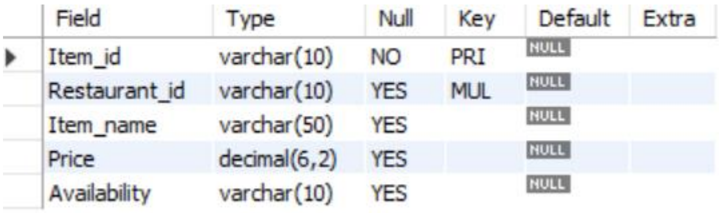
Table Description:

1)Restaurants



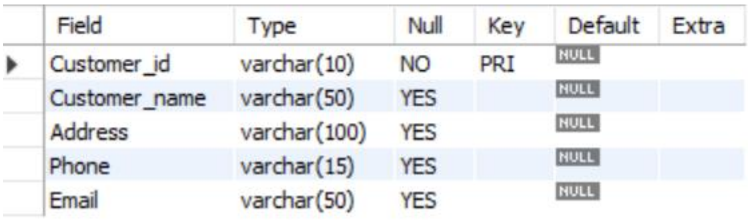
	Field	Type	Null	Key	Default	Extra
▶	Restaurant_id	varchar(10)	NO	PRI	NULL	
	Restaurant_name	varchar(50)	YES		NULL	
	Cuisine_type	varchar(30)	YES		NULL	
	Location	varchar(50)	YES		NULL	
	Rating	decimal(2,1)	YES		NULL	

2)Menu Items



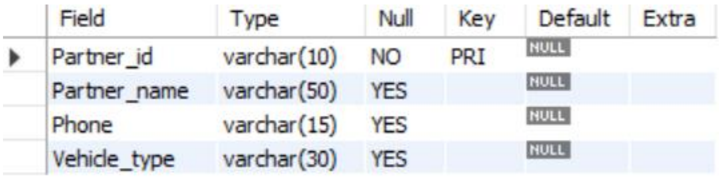
	Field	Type	Null	Key	Default	Extra
▶	Item_id	varchar(10)	NO	PRI	NULL	
	Restaurant_id	varchar(10)	YES	MUL	NULL	
	Item_name	varchar(50)	YES		NULL	
	Price	decimal(6,2)	YES		NULL	
	Availability	varchar(10)	YES		NULL	

3)Customers



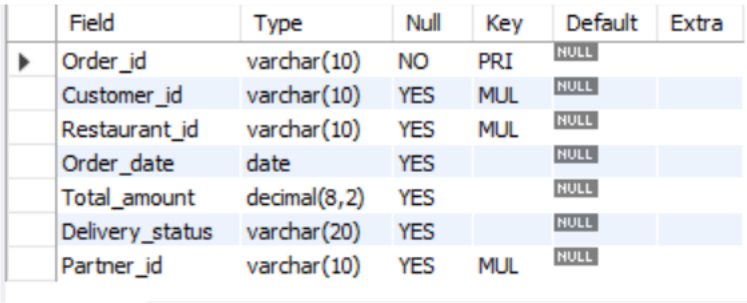
	Field	Type	Null	Key	Default	Extra
▶	Customer_id	varchar(10)	NO	PRI	NULL	
	Customer_name	varchar(50)	YES		NULL	
	Address	varchar(100)	YES		NULL	
	Phone	varchar(15)	YES		NULL	
	Email	varchar(50)	YES		NULL	

4)Delivery Partners



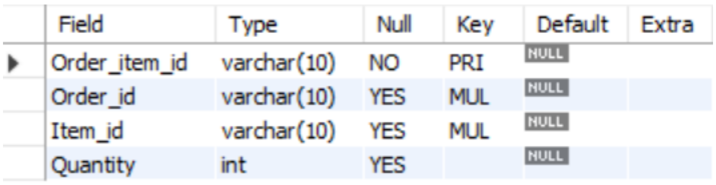
	Field	Type	Null	Key	Default	Extra
▶	Partner_id	varchar(10)	NO	PRI	NULL	
	Partner_name	varchar(50)	YES		NULL	
	Phone	varchar(15)	YES		NULL	
	Vehicle_type	varchar(30)	YES		NULL	

5)Orders



	Field	Type	Null	Key	Default	Extra
▶	Order_id	varchar(10)	NO	PRI	NULL	
	Customer_id	varchar(10)	YES	MUL	NULL	
	Restaurant_id	varchar(10)	YES	MUL	NULL	
	Order_date	date	YES		NULL	
	Total_amount	decimal(8,2)	YES		NULL	
	Delivery_status	varchar(20)	YES		NULL	
	Partner_id	varchar(10)	YES	MUL	NULL	

6)Order Items



Field	Type	Null	Key	Default	Extra
Order_item_id	varchar(10)	NO	PRI	NULL	
Order_id	varchar(10)	YES	MUL	NULL	
Item_id	varchar(10)	YES	MUL	NULL	
Quantity	int	YES		NULL	

Creating Database:

```
Create Database Online_food_delivery_system;

Use online_food_delivery_system;
```

Table Creation And Insertion Commands:

1)Create Table Restaurant

```
CREATE TABLE Restaurants (
    Restaurant_id VARCHAR(10) PRIMARY KEY,
    Restaurant_name VARCHAR(50),
    Cuisine_type VARCHAR(30),
    Location VARCHAR(50),
    Rating DECIMAL(2,1));
```

2)Inserting Values Into Restaurant

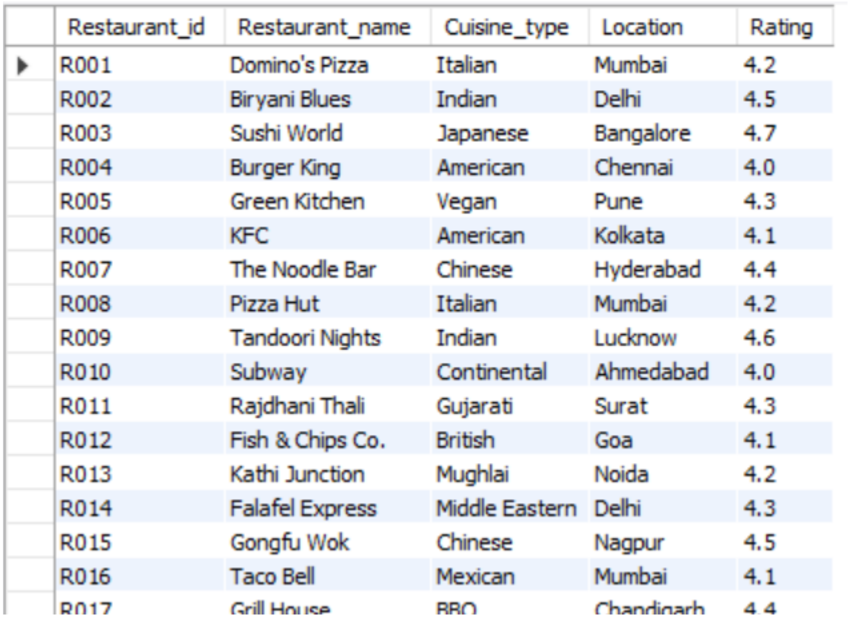
```
INSERT INTO Restaurants (Restaurant_id, Restaurant_name, Cuisine_type,
Location, Rating)
VALUES
('R001', 'Domino''s Pizza', 'Italian', 'Mumbai', 4.2),
('R002', 'Biryani Blues', 'Indian', 'Delhi', 4.5),
('R003', 'Sushi World', 'Japanese', 'Bangalore', 4.7),
('R004', 'Burger King', 'American', 'Chennai', 4.0),
('R005', 'Green Kitchen', 'Vegan', 'Pune', 4.3),
('R006', 'KFC', 'American', 'Kolkata', 4.1),
('R007', 'The Noodle Bar', 'Chinese', 'Hyderabad', 4.4),
('R008', 'Pizza Hut', 'Italian', 'Mumbai', 4.2),
('R009', 'Tandoori Nights', 'Indian', 'Lucknow', 4.6),
('R010', 'Subway', 'Continental', 'Ahmedabad', 4.0),
('R011', 'Rajdhani Thali', 'Gujarati', 'Surat', 4.3),
('R012', 'Fish & Chips Co.', 'British', 'Goa', 4.1),
```

```
SELECT * FROM BRANCH;
```

.

.

OUTPUT:



Restaurant_id	Restaurant_name	Cuisine_type	Location	Rating
R001	Domino's Pizza	Italian	Mumbai	4.2
R002	Biryani Blues	Indian	Delhi	4.5
R003	Sushi World	Japanese	Bangalore	4.7
R004	Burger King	American	Chennai	4.0
R005	Green Kitchen	Vegan	Pune	4.3
R006	KFC	American	Kolkata	4.1
R007	The Noodle Bar	Chinese	Hyderabad	4.4
R008	Pizza Hut	Italian	Mumbai	4.2
R009	Tandoori Nights	Indian	Lucknow	4.6
R010	Subway	Continental	Ahmedabad	4.0
R011	Rajdhani Thali	Gujarati	Surat	4.3
R012	Fish & Chips Co.	British	Goa	4.1
R013	Kathi Junction	Mughlai	Noida	4.2
R014	Falafel Express	Middle Eastern	Delhi	4.3
R015	Gongfu Wok	Chinese	Nagpur	4.5
R016	Taco Bell	Mexican	Mumbai	4.1
R017	Grill House	BBQ	Chandigarh	4.4

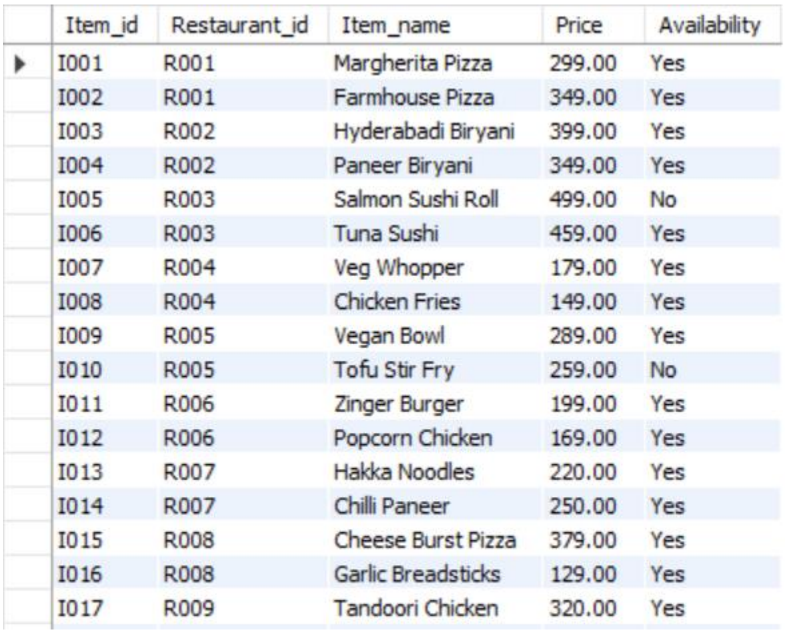
2)Create Table Menu_Items:

```
CREATE TABLE Menu_Items (
   Item_id VARCHAR(10) PRIMARY KEY,
   Restaurant_id VARCHAR(10),
   Item_name VARCHAR(50),
   Price DECIMAL(6,2),
   Availability VARCHAR(10),
   FOREIGN KEY (Restaurant_id) REFERENCES Restaurants(Restaurant_id));
```

Inserting Values Into Menu_Items:

```
INSERT INTO Menu_Items (Item_id, Restaurant_id, Item_name, Price,
Availability) VALUES
('I001', 'R001', 'Margherita Pizza', 299.00, 'Yes'),
('I002', 'R001', 'Farmhouse Pizza', 349.00, 'Yes'),
('I003', 'R002', 'Hyderabadi Biryani', 399.00, 'Yes'),
('I004', 'R002', 'Paneer Biryani', 349.00, 'Yes'),
.
.
SELECT * FROM MENU_ITEMS;
```

OUTPUT:



Item_id	Restaurant_id	Item_name	Price	Availability
I001	R001	Margherita Pizza	299.00	Yes
I002	R001	Farmhouse Pizza	349.00	Yes
I003	R002	Hyderabadi Biryani	399.00	Yes
I004	R002	Paneer Biryani	349.00	Yes
I005	R003	Salmon Sushi Roll	499.00	No
I006	R003	Tuna Sushi	459.00	Yes
I007	R004	Veg Whopper	179.00	Yes
I008	R004	Chicken Fries	149.00	Yes
I009	R005	Vegan Bowl	289.00	Yes
I010	R005	Tofu Stir Fry	259.00	No
I011	R006	Zinger Burger	199.00	Yes
I012	R006	Popcorn Chicken	169.00	Yes
I013	R007	Hakka Noodles	220.00	Yes
I014	R007	Chilli Paneer	250.00	Yes
I015	R008	Cheese Burst Pizza	379.00	Yes
I016	R008	Garlic Breadsticks	129.00	Yes
I017	R009	Tandoori Chicken	320.00	Yes

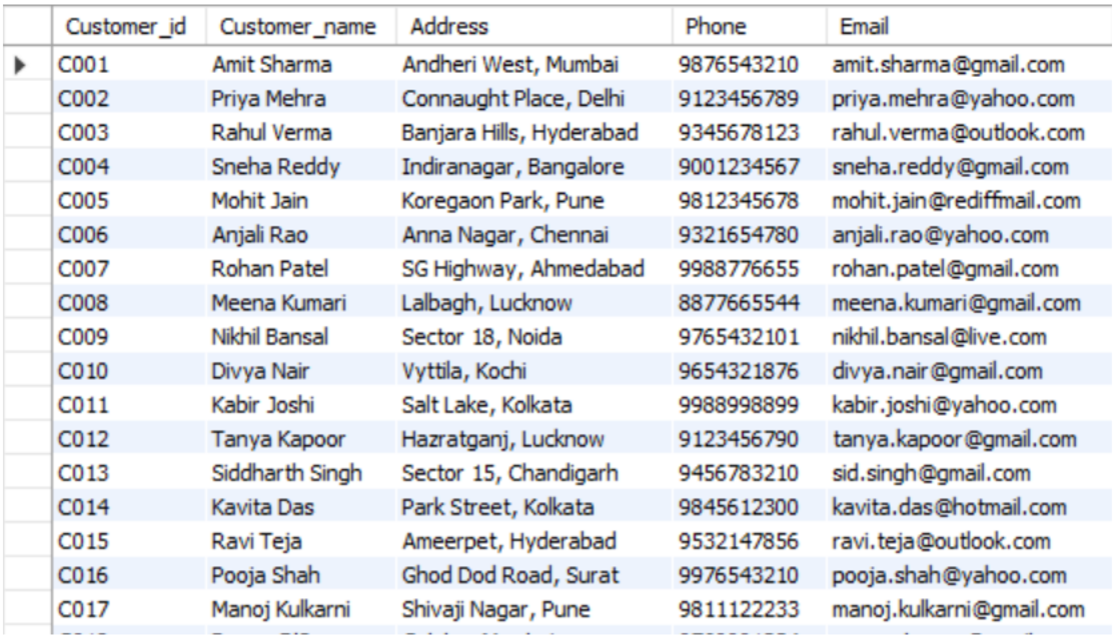
3)Create Table Customers:

```
CREATE TABLE Customers (
   Customer_id VARCHAR(10) PRIMARY KEY,
   Customer_name VARCHAR(50),
  Address VARCHAR(100),
   Phone VARCHAR(15),
   Email VARCHAR(50));
```

Inserting Values Into Customers:

```
INSERT INTO Customers (Customer_id, Customer_name, Address, Phone, Email)
VALUES
('C001', 'Amit Sharma', 'Andheri West, Mumbai', '9876543210',
'amit.sharma@gmail.com'),
('C002', 'Priya Mehra', 'Connaught Place, Delhi', '9123456789',
'priya.mehra@yahoo.com'),
('C003', 'Rahul Verma', 'Banjara Hills, Hyderabad', '9345678123',
'rahul.verma@outlook.com'),
('C004', 'Sneha Reddy', 'Indiranagar, Bangalore', '9001234567',
'sneha.reddy@gmail.com'),
.
SELECT * FROM CUSTOMERS;
```

OUTPUT:



Customer_id	Customer_name	Address	Phone	Email
C001	Amit Sharma	Andheri West, Mumbai	9876543210	amit.sharma@gmail.com
C002	Priya Mehra	Connaught Place, Delhi	9123456789	priya.mehra@yahoo.com
C003	Rahul Verma	Banjara Hills, Hyderabad	9345678123	rahul.verma@outlook.com
C004	Sneha Reddy	Indiranagar, Bangalore	9001234567	sneha.reddy@gmail.com
C005	Mohit Jain	Koregaon Park, Pune	9812345678	mohit.jain@rediffmail.com
C006	Anjali Rao	Anna Nagar, Chennai	9321654780	anjali.rao@yahoo.com
C007	Rohan Patel	SG Highway, Ahmedabad	9988776655	rohan.patel@gmail.com
C008	Meena Kumari	Lalbagh, Lucknow	8877665544	meena.kumari@gmail.com
C009	Nikhil Bansal	Sector 18, Noida	9765432101	nikhil.bansal@live.com
C010	Divya Nair	Vyttila, Kochi	9654321876	divya.nair@gmail.com
C011	Kabir Joshi	Salt Lake, Kolkata	9988998899	kabir.joshi@yahoo.com
C012	Tanya Kapoor	Hazratganj, Lucknow	9123456790	tanya.kapoor@gmail.com
C013	Siddharth Singh	Sector 15, Chandigarh	9456783210	sid.singh@gmail.com
C014	Kavita Das	Park Street, Kolkata	9845612300	kavita.das@hotmail.com
C015	Ravi Teja	Ameerpet, Hyderabad	9532147856	ravi.teja@outlook.com
C016	Pooja Shah	Ghod Dod Road, Surat	9976543210	pooja.shah@yahoo.com
C017	Manoj Kulkarni	Shivaji Nagar, Pune	9811122233	manoj.kulkarni@gmail.com

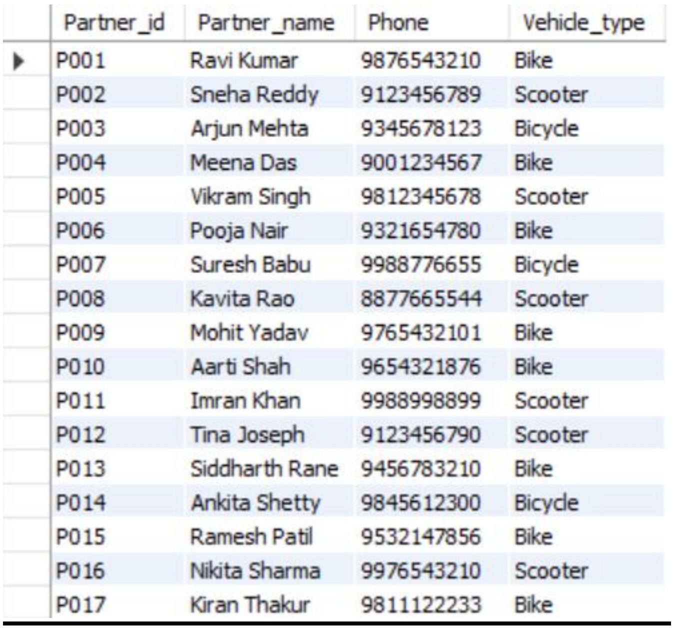
4)Create Table Delivery_Partners:

```
CREATE TABLE Delivery_Partners (
   Partner_id VARCHAR(10) PRIMARY KEY,
   Partner_name VARCHAR(50),
   Phone VARCHAR(15),
   Vehicle_type VARCHAR(30));
```

Inserting Values Into Delivery_Partners:

```
INSERT INTO Delivery_Partners (Partner_id, Partner_name, Phone, Vehicle_type)
VALUES
('P001', 'Ravi Kumar', '9876543210', 'Bike'),
('P002', 'Sneha Reddy', '9123456789', 'Scooter'),
('P003', 'Arjun Mehta', '9345678123', 'Bicycle'),
('P004', 'Meena Das', '9001234567', 'Bike'),
('P005', 'Vikram Singh', '9812345678', 'Scooter'),
('P006', 'Pooja Nair', '9321654780', 'Bike'),
.
.
SELECT * FROM DELIVERY_PARTNERS:
```

OUTPUT:



Partner_id	Partner_name	Phone	Vehicle_type
P001	Ravi Kumar	9876543210	Bike
P002	Sneha Reddy	9123456789	Scooter
P003	Arjun Mehta	9345678123	Bicycle
P004	Meena Das	9001234567	Bike
P005	Vikram Singh	9812345678	Scooter
P006	Pooja Nair	9321654780	Bike
P007	Suresh Babu	9988776655	Bicycle
P008	Kavita Rao	8877665544	Scooter
P009	Mohit Yadav	9765432101	Bike
P010	Aarti Shah	9654321876	Bike
P011	Imran Khan	9988998899	Scooter
P012	Tina Joseph	9123456790	Scooter
P013	Siddharth Rane	9456783210	Bike
P014	Ankita Shetty	9845612300	Bicycle
P015	Ramesh Patil	9532147856	Bike
P016	Nikita Sharma	9976543210	Scooter
P017	Kiran Thakur	9811122233	Bike

5)Create Table Orders:

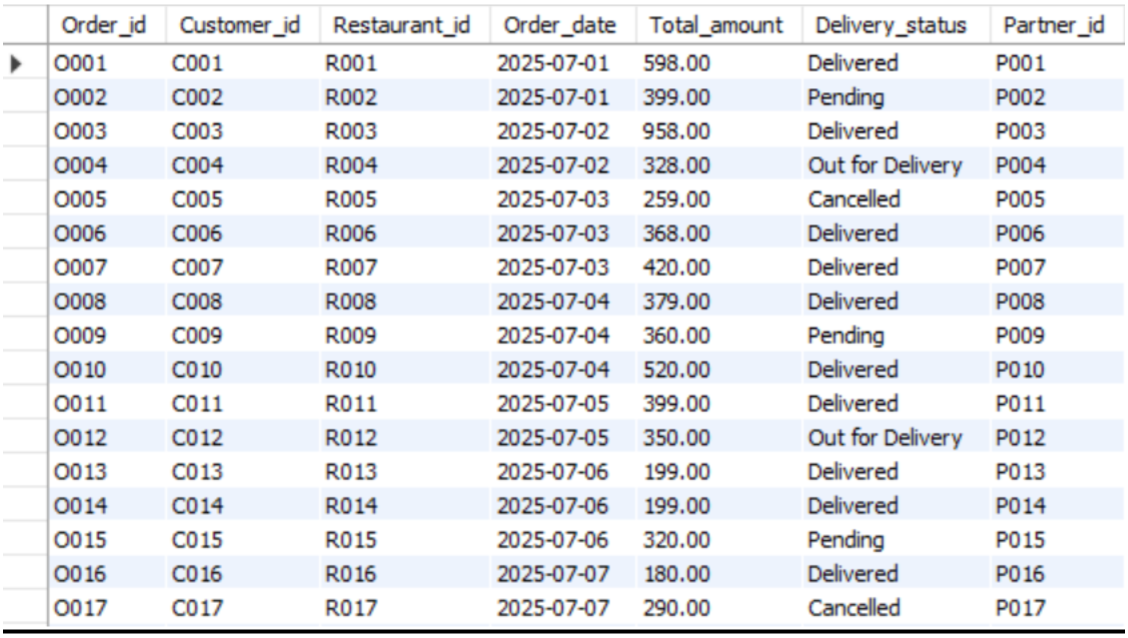
```
CREATE TABLE Orders (
   Order_id VARCHAR(10) PRIMARY KEY,
   Customer_id VARCHAR(10),
   Restaurant_id VARCHAR(10),
   Order_date DATE,
   Total_amount DECIMAL(8,2),
   Delivery_status VARCHAR(20),
   Partner_id VARCHAR(10),
   FOREIGN KEY (Customer_id) REFERENCES Customers(Customer_id),
   FOREIGN KEY (Restaurant_id) REFERENCES Restaurants(Restaurant_id),
   FOREIGN KEY (Partner_id) REFERENCES Delivery_Partners(Partner_id)):
```

Inserting Values Into Orders:

```
INSERT INTO Orders (Order_id, Customer_id, Restaurant_id, Order_date,
Total_amount, Delivery_status, Partner_id) VALUES
('O001', 'C001', 'R001', '2025-07-01', 598.00, 'Delivered', 'P001'),
('O002', 'C002', 'R002', '2025-07-01', 399.00, 'Pending', 'P002'),
('O003', 'C003', 'R003', '2025-07-02', 958.00, 'Delivered', 'P003'),
('O004', 'C004', 'R004', '2025-07-02', 328.00, 'Out for Delivery', 'P004'),
('O005', 'C005', 'R005', '2025-07-03', 259.00, 'Cancelled', 'P005'),
('O006', 'C006', 'R006', '2025-07-03', 368.00, 'Delivered', 'P006'),
('O007', 'C007', 'R007', '2025-07-03', 420.00, 'Delivered', 'P007'),
('O008', 'C008', 'R008', '2025-07-04', 379.00, 'Delivered', 'P008'),
('O009', 'C009', 'R009', '2025-07-04', 360.00, 'Pending', 'P009'),
('O010', 'C010', 'R010', '2025-07-04', 520.00, 'Delivered', 'P010'),
('O011', 'C011', 'R011', '2025-07-05', 399.00, 'Delivered', 'P011'),
('O012', 'C012', 'R012', '2025-07-05', 350.00, 'Out for Delivery', 'P012'),
('O013', 'C013', 'R013', '2025-07-06', 199.00, 'Delivered', 'P013'),
('O014', 'C014', 'R014', '2025-07-06', 199.00, 'Delivered', 'P014'),
.
.
```

SELECT * FROM ORDERS;

OUTPUT:



Order_id	Customer_id	Restaurant_id	Order_date	Total_amount	Delivery_status	Partner_id
O001	C001	R001	2025-07-01	598.00	Delivered	P001
O002	C002	R002	2025-07-01	399.00	Pending	P002
O003	C003	R003	2025-07-02	958.00	Delivered	P003
O004	C004	R004	2025-07-02	328.00	Out for Delivery	P004
O005	C005	R005	2025-07-03	259.00	Cancelled	P005
O006	C006	R006	2025-07-03	368.00	Delivered	P006
O007	C007	R007	2025-07-03	420.00	Delivered	P007
O008	C008	R008	2025-07-04	379.00	Delivered	P008
O009	C009	R009	2025-07-04	360.00	Pending	P009
O010	C010	R010	2025-07-04	520.00	Delivered	P010
O011	C011	R011	2025-07-05	399.00	Delivered	P011
O012	C012	R012	2025-07-05	350.00	Out for Delivery	P012
O013	C013	R013	2025-07-06	199.00	Delivered	P013
O014	C014	R014	2025-07-06	199.00	Delivered	P014
O015	C015	R015	2025-07-06	320.00	Pending	P015
O016	C016	R016	2025-07-07	180.00	Delivered	P016
O017	C017	R017	2025-07-07	290.00	Cancelled	P017

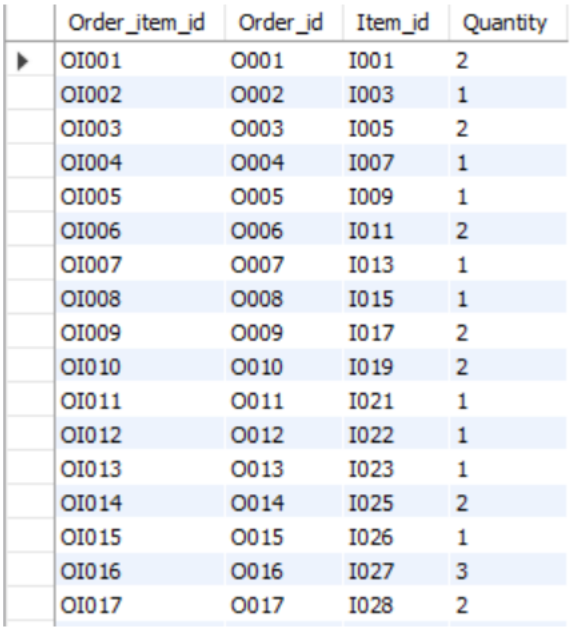
6)Create Table Order_Items:

```
CREATE TABLE Order_Items (
   Order_item_id VARCHAR(10) PRIMARY KEY,
   Order_id VARCHAR(10),
   Item_id VARCHAR(10),
   Quantity INT,
   FOREIGN KEY (Order_id) REFERENCES Orders(Order_id),
   FOREIGN KEY (Item_id) REFERENCES Menu_Items(Item_id));
```

Inserting Values Into Order_Items:

```
INSERT INTO Order_Items (Order_item_id, Order_id, Item_id, Quantity) VALUES
('OI001', 'O001', 'I001', 2),
('OI002', 'O002', 'I003', 1),
('OI003', 'O003', 'I005', 2),
('OI004', 'O004', 'I007', 1),
.
.
SELECT * FROM ORDER_ITEMS;
```

OUTPUT:



Order_item_id	Order_id	Item_id	Quantity
OI001	O001	I001	2
OI002	O002	I003	1
OI003	O003	I005	2
OI004	O004	I007	1
OI005	O005	I009	1
OI006	O006	I011	2
OI007	O007	I013	1
OI008	O008	I015	1
OI009	O009	I017	2
OI010	O010	I019	2
OI011	O011	I021	1
OI012	O012	I022	1
OI013	O013	I023	1
OI014	O014	I025	2
OI015	O015	I026	1
OI016	O016	I027	3
OI017	O017	I028	2

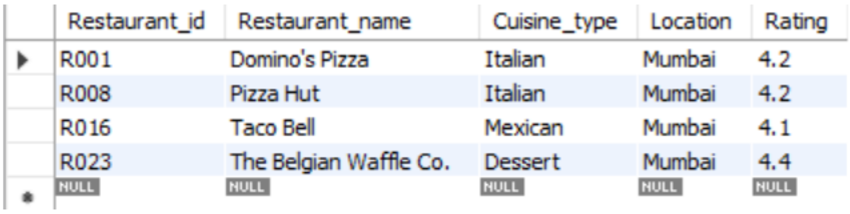
BASIC QUESTIONS

1.View all restaurants in a specific city

SELECT * FROM Restaurants

WHERE Location = 'Mumbai';

OUTPUT:



	Restaurant_id	Restaurant_name	Cuisine_type	Location	Rating
▶	R001	Domino's Pizza	Italian	Mumbai	4.2
	R008	Pizza Hut	Italian	Mumbai	4.2
	R016	Taco Bell	Mexican	Mumbai	4.1
	R023	The Belgian Waffle Co.	Dessert	Mumbai	4.4
*	NULL	NULL	NULL	NULL	NULL

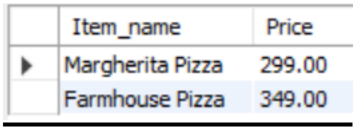
2. List all available menu items from a restaurant

SELECT Item_name, Price

FROM Menu_Items

WHERE Restaurant_id = 'R001' AND Availability = 'Yes';

OUTPUT:



	Item_name	Price
▶	Margherita Pizza	299.00
	Farmhouse Pizza	349.00

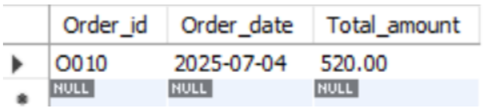
3. Get all orders delivered by a specific delivery partner

SELECT Order_id, Order_date, Total_amount

FROM Orders

WHERE Partner_id = 'P010' AND Delivery_status = 'Delivered';

OUTPUT:

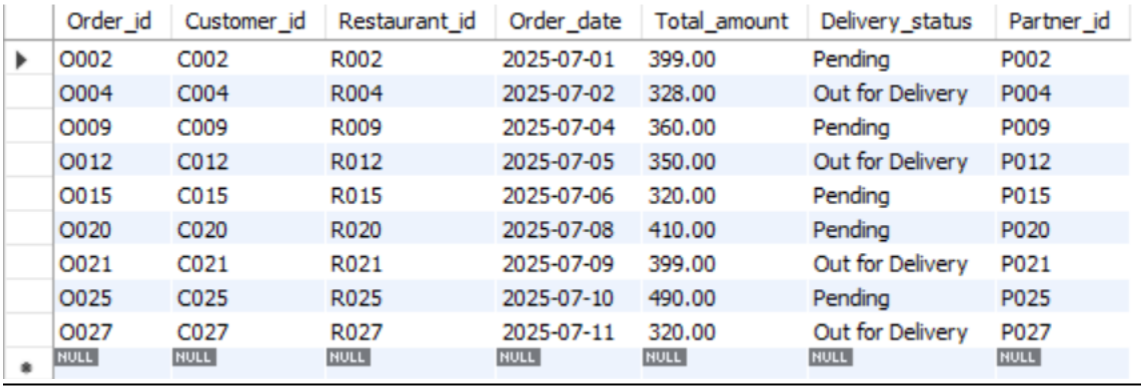


	Order_id	Order_date	Total_amount
▶	O010	2025-07-04	520.00
*	NULL	NULL	NULL

4. List of orders that are still pending or out for delivery

```
SELECT * FROM Orders

WHERE Delivery_status IN ('Pending', 'Out for Delivery');
```

OUTPUT:

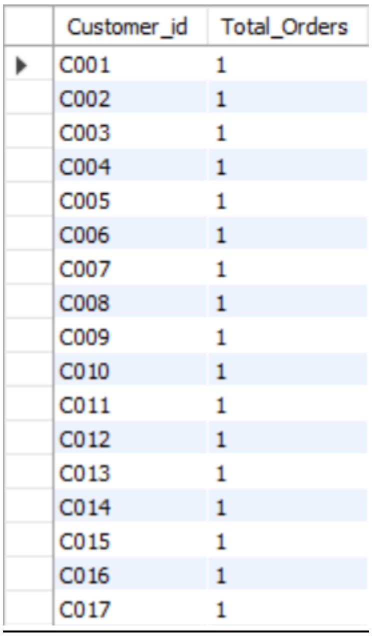


Order_id	Customer_id	Restaurant_id	Order_date	Total_amount	Delivery_status	Partner_id
O002	C002	R002	2025-07-01	399.00	Pending	P002
O004	C004	R004	2025-07-02	328.00	Out for Delivery	P004
O009	C009	R009	2025-07-04	360.00	Pending	P009
O012	C012	R012	2025-07-05	350.00	Out for Delivery	P012
O015	C015	R015	2025-07-06	320.00	Pending	P015
O020	C020	R020	2025-07-08	410.00	Pending	P020
O021	C021	R021	2025-07-09	399.00	Out for Delivery	P021
O025	C025	R025	2025-07-10	490.00	Pending	P025
O027	C027	R027	2025-07-11	320.00	Out for Delivery	P027
NULL	NULL	NULL	NULL	NULL	NULL	NULL

5. Find total number of orders placed by each customer

```
SELECT Customer_id, COUNT(*) AS Total_Orders

FROM Orders

GROUP BY Customer_id;
```

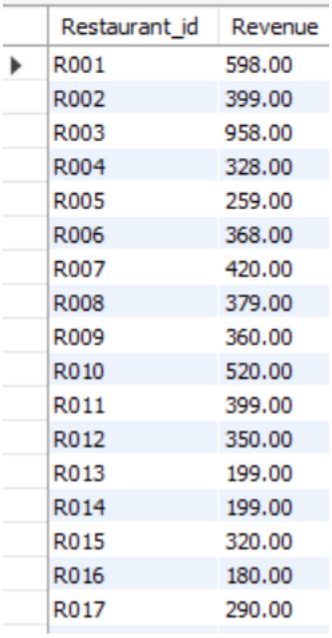
OUTPUT:



Customer_id	Total_Orders
C001	1
C002	1
C003	1
C004	1
C005	1
C006	1
C007	1
C008	1
C009	1
C010	1
C011	1
C012	1
C013	1
C014	1
C015	1
C016	1
C017	1

6. Get the total amount earned by each restaurant

```
SELECT Restaurant_id, SUM(Total_amount) AS Revenue

FROM Orders

GROUP BY Restaurant_id;
```

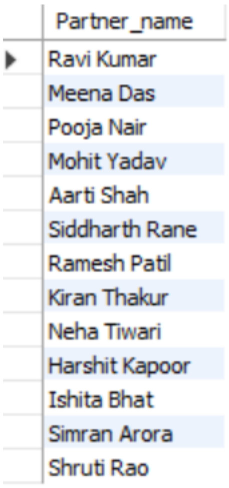


Restaurant_id	Revenue
R001	598.00
R002	399.00
R003	958.00
R004	328.00
R005	259.00
R006	368.00
R007	420.00
R008	379.00
R009	360.00
R010	520.00
R011	399.00
R012	350.00
R013	199.00
R014	199.00
R015	320.00
R016	180.00
R017	290.00

7. Get the names of all delivery partners who use a bike

```
SELECT Partner_name

FROM Delivery_Partners

WHERE Vehicle_type = 'Bike';
```

OUTPUT:

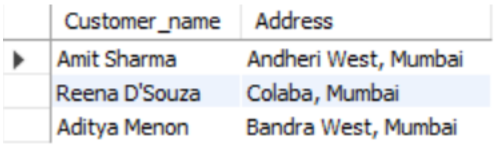


Partner_name
Ravi Kumar
Meena Das
Pooja Nair
Mohit Yadav
Aarti Shah
Siddharth Rane
Ramesh Patil
Kiran Thakur
Neha Tiwari
Harshit Kapoor
Ishita Bhat
Simran Arora
Shruti Rao

8. List all customers from a specific city or address area

```
SELECT Customer_name, Address

FROM Customers

WHERE Address LIKE '%Mumbai%';
```

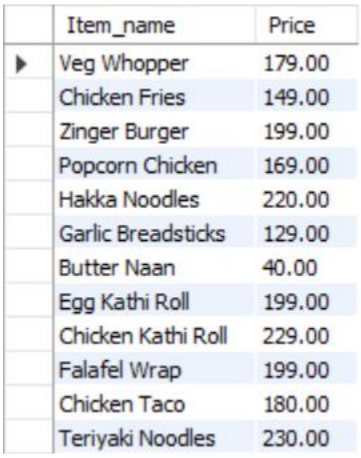
OUTPUT:



Customer_name	Address
Amit Sharma	Andheri West, Mumbai
Reena D'Souza	Colaba, Mumbai
Aditya Menon	Bandra West, Mumbai

9. Get the list of menu items with price less than ₹250

```
SELECT Item_name, Price

FROM Menu_Items

WHERE Price < 250;
```

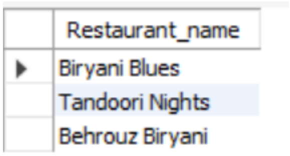


Item_name	Price
Veg Whopper	179.00
Chicken Fries	149.00
Zinger Burger	199.00
Popcorn Chicken	169.00
Hakka Noodles	220.00
Garlic Breadsticks	129.00
Butter Naan	40.00
Egg Kathi Roll	199.00
Chicken Kathi Roll	229.00
Falafel Wrap	199.00
Chicken Taco	180.00
Teriyaki Noodles	230.00

10. Display restaurant names offering a specific cuisine

```
SELECT Restaurant_name

FROM Restaurants

WHERE Cuisine_type = 'Indian';
```

OUTPUT:



Restaurant_name
Biryani Blues
Tandoori Nights
Behrouz Biryani

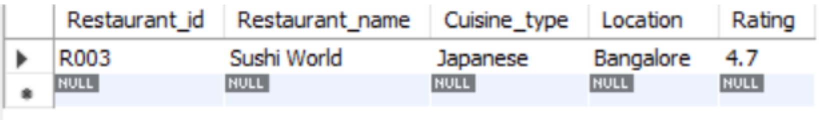
SUB-QUERIES

1.Highest Rated Restaurant

SELECT * FROM Restaurants

WHERE Rating = (SELECT MAX(Rating) FROM Restaurants);

OUTPUT:



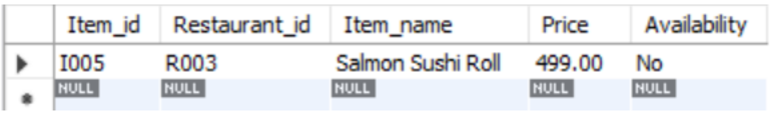
	Restaurant_id	Restaurant_name	Cuisine_type	Location	Rating
▶	R003	Sushi World	Japanese	Bangalore	4.7
*	NULL	NULL	NULL	NULL	NULL

2. Get the highest-priced menu item

SELECT * FROM Menu_Items

WHERE Price = (SELECT MAX(Price) FROM Menu_Items);

OUTPUT:



	Item_id	Restaurant_id	Item_name	Price	Availability
▶	I005	R003	Salmon Sushi Roll	499.00	No
*	NULL	NULL	NULL	NULL	NULL

3. List restaurants that offer the same cuisine as 'Biryani Blues'

SELECT * FROM Restaurants

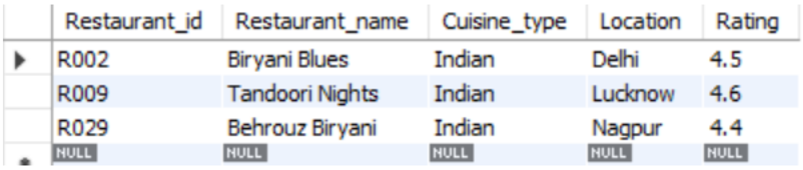
WHERE Cuisine_type = (

SELECT Cuisine_type

FROM Restaurants

WHERE Restaurant_name = 'Biryani Blues');

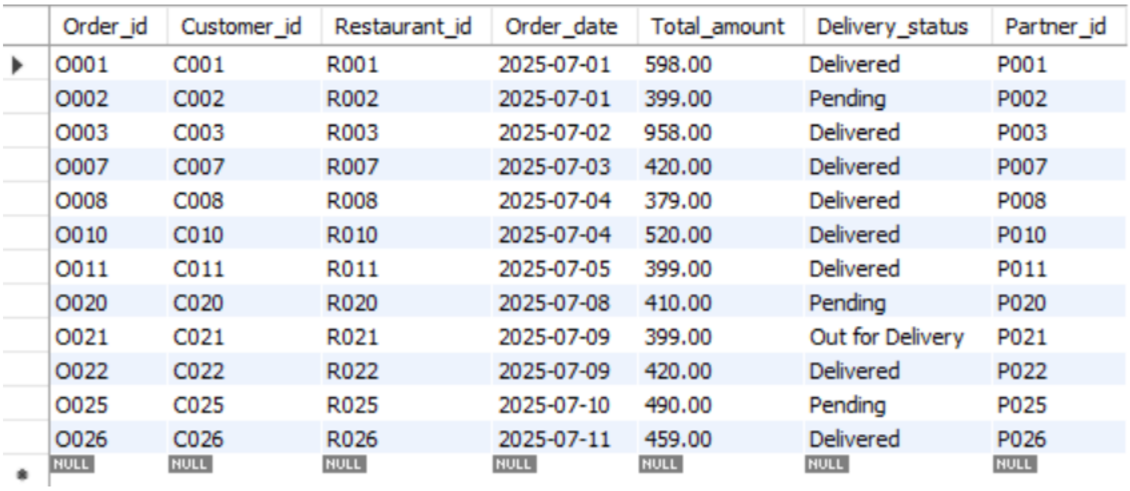
OUTPUT:



	Restaurant_id	Restaurant_name	Cuisine_type	Location	Rating
▶	R002	Biryani Blues	Indian	Delhi	4.5
	R009	Tandoori Nights	Indian	Lucknow	4.6
	R029	Behrouz Biryani	Indian	Nagpur	4.4
*	NULL	NULL	NULL	NULL	NULL

4. Show orders with above-average total amount

```
SELECT * FROM Orders
WHERE Total_amount > (
   SELECT AVG(Total_amount) FROM Orders);
```

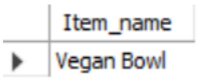


Order_id	Customer_id	Restaurant_id	Order_date	Total_amount	Delivery_status	Partner_id
O001	C001	R001	2025-07-01	598.00	Delivered	P001
O002	C002	R002	2025-07-01	399.00	Pending	P002
O003	C003	R003	2025-07-02	958.00	Delivered	P003
O007	C007	R007	2025-07-03	420.00	Delivered	P007
O008	C008	R008	2025-07-04	379.00	Delivered	P008
O010	C010	R010	2025-07-04	520.00	Delivered	P010
O011	C011	R011	2025-07-05	399.00	Delivered	P011
O020	C020	R020	2025-07-08	410.00	Pending	P020
O021	C021	R021	2025-07-09	399.00	Out for Delivery	P021
O022	C022	R022	2025-07-09	420.00	Delivered	P022
O025	C025	R025	2025-07-10	490.00	Pending	P025
O026	C026	R026	2025-07-11	459.00	Delivered	P026
NULL	NULL	NULL	NULL	NULL	NULL	NULL

5. Find menu items ordered in order 'O005'

```
SELECT Item_name
FROM Menu_Items
WHERE Item_id IN (
   SELECT Item_id
   FROM Order_Items
   WHERE Order_id = 'O005');
```

OUTPUT:

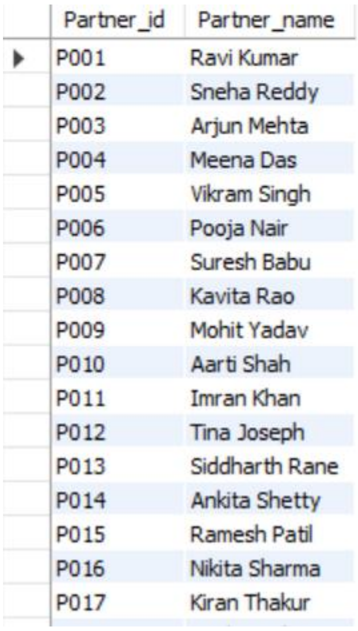


Item_name
Vegan Bowl

6. Find delivery partners who have delivered at least one order

```
SELECT Partner_id, Partner_name
FROM Delivery_Partners
WHERE Partner_id IN (
   SELECT DISTINCT Partner_id
   FROM Orders);
```

OUTPUT:

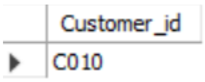


Partner_id	Partner_name
P001	Ravi Kumar
P002	Sneha Reddy
P003	Arjun Mehta
P004	Meena Das
P005	Vikram Singh
P006	Pooja Nair
P007	Suresh Babu
P008	Kavita Rao
P009	Mohit Yadav
P010	Aarti Shah
P011	Imran Khan
P012	Tina Joseph
P013	Siddharth Rane
P014	Ankita Shetty
P015	Ramesh Patil
P016	Nikita Sharma
P017	Kiran Thakur

7) Get customers who ordered from the same restaurant as 'O010'

```
SELECT DISTINCT Customer_id

FROM Orders

WHERE Restaurant_id = (

   SELECT Restaurant_id

   FROM Orders

   WHERE Order_id = 'O010');
```

OUTPUT:



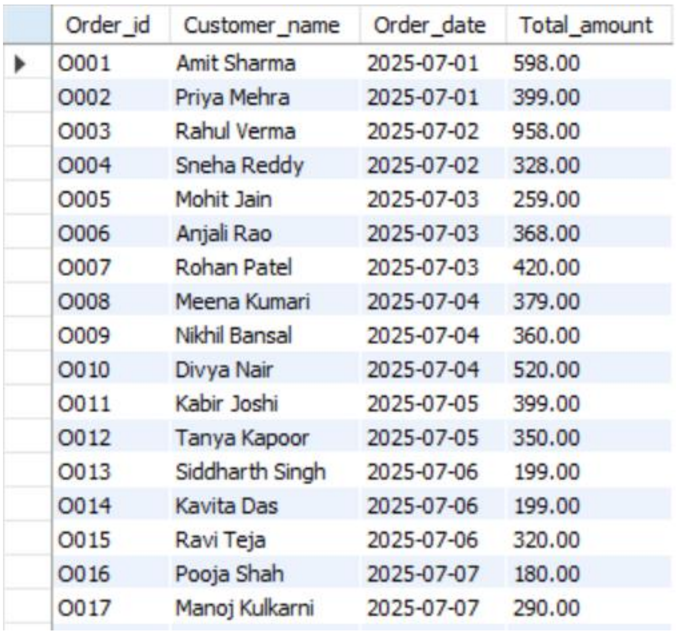
Customer_id
C010

JOINS

1. List orders with customer names and total amount

```
SELECT o.Order_id, c.Customer_name, o.Order_date, o.Total_amount

FROM Orders o

JOIN Customers c ON o.Customer_id = c.Customer_id;
```

OUTPUT:

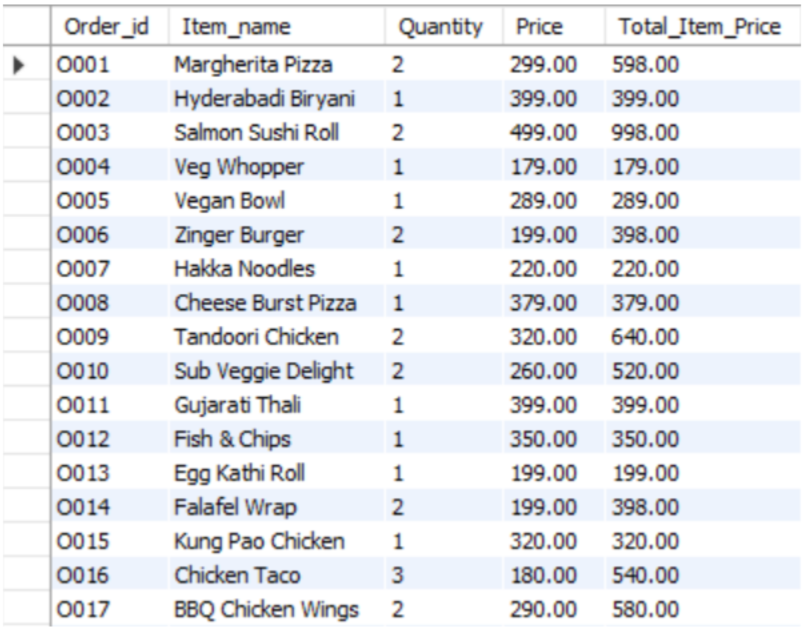


Order_id	Customer_name	Order_date	Total_amount
O001	Amit Sharma	2025-07-01	598.00
O002	Priya Mehra	2025-07-01	399.00
O003	Rahul Verma	2025-07-02	958.00
O004	Sneha Reddy	2025-07-02	328.00
O005	Mohit Jain	2025-07-03	259.00
O006	Anjali Rao	2025-07-03	368.00
O007	Rohan Patel	2025-07-03	420.00
O008	Meena Kumari	2025-07-04	379.00
O009	Nikhil Bansal	2025-07-04	360.00
O010	Divya Nair	2025-07-04	520.00
O011	Kabir Joshi	2025-07-05	399.00
O012	Tanya Kapoor	2025-07-05	350.00
O013	Siddharth Singh	2025-07-06	199.00
O014	Kavita Das	2025-07-06	199.00
O015	Ravi Teja	2025-07-06	320.00
O016	Pooja Shah	2025-07-07	180.00
O017	Manoj Kulkarni	2025-07-07	290.00

2. Show all ordered menu items with their quantity and price

```
SELECT oi.Order_id, m.Item_name, oi.Quantity, m.Price, (oi.Quantity * m.Price) AS Total_Item_Price

FROM Order_Items oi

JOIN Menu_Items m ON oi.Item_id = m.Item_id;
```

OUTPUT:

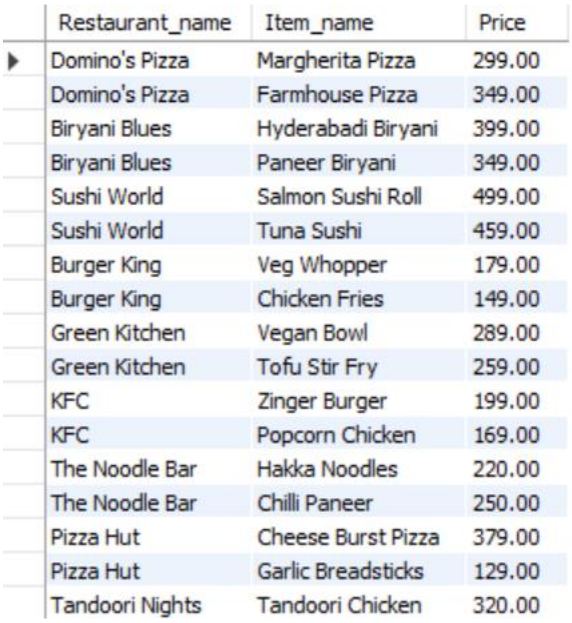


Order_id	Item_name	Quantity	Price	Total_Item_Price
O001	Margherita Pizza	2	299.00	598.00
O002	Hyderabadi Biryani	1	399.00	399.00
O003	Salmon Sushi Roll	2	499.00	998.00
O004	Veg Whopper	1	179.00	179.00
O005	Vegan Bowl	1	289.00	289.00
O006	Zinger Burger	2	199.00	398.00
O007	Hakka Noodles	1	220.00	220.00
O008	Cheese Burst Pizza	1	379.00	379.00
O009	Tandoori Chicken	2	320.00	640.00
O010	Sub Veggie Delight	2	260.00	520.00
O011	Gujarati Thali	1	399.00	399.00
O012	Fish & Chips	1	350.00	350.00
O013	Egg Kathi Roll	1	199.00	199.00
O014	Falafel Wrap	2	199.00	398.00
O015	Kung Pao Chicken	1	320.00	320.00
O016	Chicken Taco	3	180.00	540.00
O017	BBQ Chicken Wings	2	290.00	580.00

3**. Display restaurant name and menu item list**

```
SELECT r.Restaurant_name, m.Item_name, m.Price

FROM Restaurants r

JOIN Menu_Items m ON r.Restaurant_id = m.Restaurant_id;
```

OUTPUT:

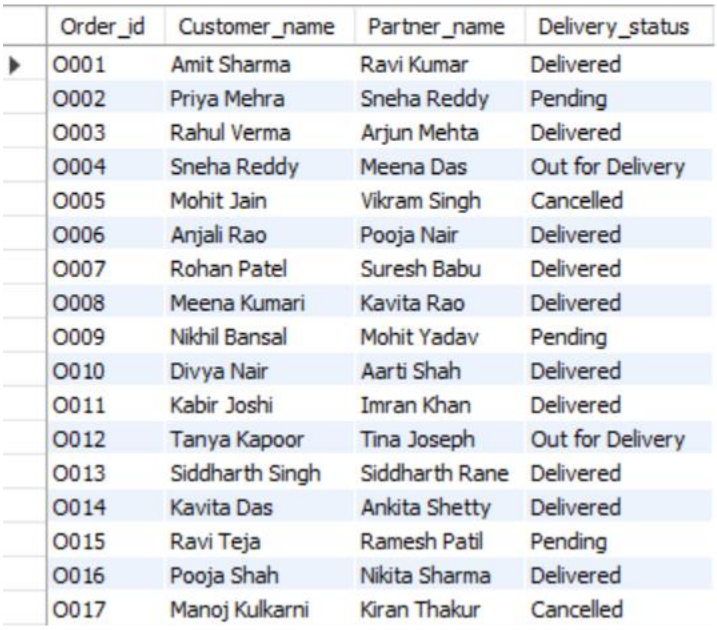


Restaurant_name	Item_name	Price
Domino's Pizza	Margherita Pizza	299.00
Domino's Pizza	Farmhouse Pizza	349.00
Biryani Blues	Hyderabadi Biryani	399.00
Biryani Blues	Paneer Biryani	349.00
Sushi World	Salmon Sushi Roll	499.00
Sushi World	Tuna Sushi	459.00
Burger King	Veg Whopper	179.00
Burger King	Chicken Fries	149.00
Green Kitchen	Vegan Bowl	289.00
Green Kitchen	Tofu Stir Fry	259.00
KFC	Zinger Burger	199.00
KFC	Popcorn Chicken	169.00
The Noodle Bar	Hakka Noodles	220.00
The Noodle Bar	Chilli Paneer	250.00
Pizza Hut	Cheese Burst Pizza	379.00
Pizza Hut	Garlic Breadsticks	129.00
Tandoori Nights	Tandoori Chicken	320.00

4. List customers along with the delivery partner who delivered their order

```
SELECT o.Order_id, c.Customer_name, dp.Partner_name, o.Delivery_status

FROM Orders o

JOIN Customers c ON o.Customer_id = c.Customer_id

JOIN Delivery_Partners dp ON o.Partner_id = dp.Partner_id;
```

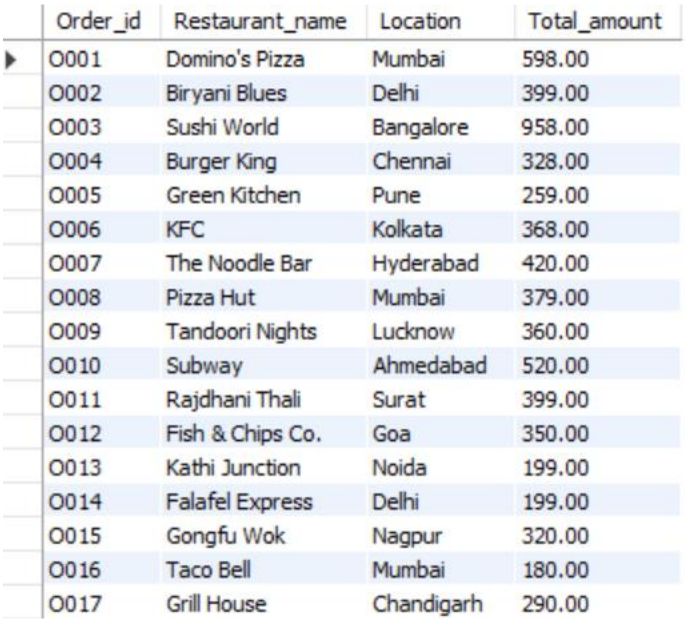
OUTPUT:



	Order_id	Customer_name	Partner_name	Delivery_status
▶	O001	Amit Sharma	Ravi Kumar	Delivered
	O002	Priya Mehra	Sneha Reddy	Pending
	O003	Rahul Verma	Arjun Mehta	Delivered
	O004	Sneha Reddy	Meena Das	Out for Delivery
	O005	Mohit Jain	Vikram Singh	Cancelled
	O006	Anjali Rao	Pooja Nair	Delivered
	O007	Rohan Patel	Suresh Babu	Delivered
	O008	Meena Kumari	Kavita Rao	Delivered
	O009	Nikhil Bansal	Mohit Yadav	Pending
	O010	Divya Nair	Aarti Shah	Delivered
	O011	Kabir Joshi	Imran Khan	Delivered
	O012	Tanya Kapoor	Tina Joseph	Out for Delivery
	O013	Siddharth Singh	Siddharth Rane	Delivered
	O014	Kavita Das	Ankita Shetty	Delivered
	O015	Ravi Teja	Ramesh Patil	Pending
	O016	Pooja Shah	Nikita Sharma	Delivered
	O017	Manoj Kulkarni	Kiran Thakur	Cancelled

5. Show all orders with restaurant name and location

```
SELECT o.Order_id, r.Restaurant_name, r.Location, o.Total_amount

FROM Orders o

JOIN Restaurants r ON o.Restaurant_id = r.Restaurant_id;
```

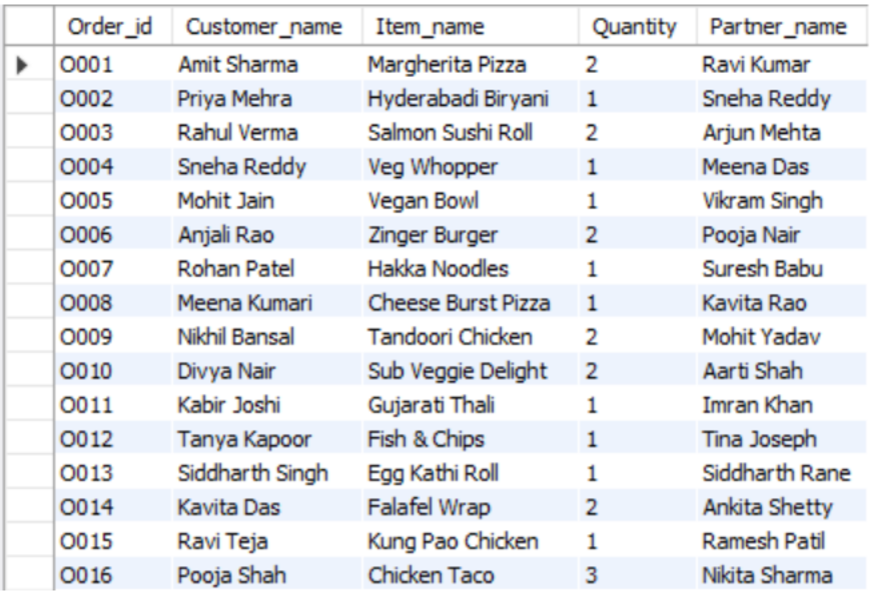


Order_id	Restaurant_name	Location	Total_amount
O001	Domino's Pizza	Mumbai	598.00
O002	Biryani Blues	Delhi	399.00
O003	Sushi World	Bangalore	958.00
O004	Burger King	Chennai	328.00
O005	Green Kitchen	Pune	259.00
O006	KFC	Kolkata	368.00
O007	The Noodle Bar	Hyderabad	420.00
O008	Pizza Hut	Mumbai	379.00
O009	Tandoori Nights	Lucknow	360.00
O010	Subway	Ahmedabad	520.00
O011	Rajdhani Thali	Surat	399.00
O012	Fish & Chips Co.	Goa	350.00
O013	Kathi Junction	Noida	199.00
O014	Falafel Express	Delhi	199.00
O015	Gongfu Wok	Nagpur	320.00
O016	Taco Bell	Mumbai	180.00
O017	Grill House	Chandigarh	290.00

6.Get full order summary: customer name, item name, quantity, delivery partner

```
SELECT o.Order_id, c.Customer_name, m.Item_name, oi.Quantity,
dp.Partner_name

FROM Orders o

JOIN Customers c ON o.Customer_id = c.Customer_id

JOIN Order_Items oi ON o.Order_id = oi.Order_id

JOIN Menu_Items m ON oi.Item_id = m.Item_id

JOIN Delivery_Partners dp ON o.Partner_id = dp.Partner_id;
```

OUTPUT:



	Order_id	Customer_name	Item_name	Quantity	Partner_name
▶	O001	Amit Sharma	Margherita Pizza	2	Ravi Kumar
	O002	Priya Mehra	Hyderabadi Biryani	1	Sneha Reddy
	O003	Rahul Verma	Salmon Sushi Roll	2	Arjun Mehta
	O004	Sneha Reddy	Veg Whopper	1	Meena Das
	O005	Mohit Jain	Vegan Bowl	1	Vikram Singh
	O006	Anjali Rao	Zinger Burger	2	Pooja Nair
	O007	Rohan Patel	Hakka Noodles	1	Suresh Babu
	O008	Meena Kumari	Cheese Burst Pizza	1	Kavita Rao
	O009	Nikhil Bansal	Tandoori Chicken	2	Mohit Yadav
	O010	Divya Nair	Sub Veggie Delight	2	Aarti Shah
	O011	Kabir Joshi	Gujarati Thali	1	Imran Khan
	O012	Tanya Kapoor	Fish & Chips	1	Tina Joseph
	O013	Siddharth Singh	Egg Kathi Roll	1	Siddharth Rane
	O014	Kavita Das	Falafel Wrap	2	Ankita Shetty
	O015	Ravi Teja	Kung Pao Chicken	1	Ramesh Patil
	O016	Pooja Shah	Chicken Taco	3	Nikita Sharma

CONCLUSION

The *Online Food Delivery System* project successfully demonstrates the creation and management of a real-world relational database using SQL. It efficiently handles core components such as restaurants, menu items, customers, orders, delivery partners, and order details through well-designed tables and relationships. With the use of SQL queries—ranging from basic to advanced (including subqueries and joins)—the project enables data retrieval, analysis, and reporting. This system not only streamlines food ordering and delivery operations but also showcases the practical application of database design principles in the e-commerce and food-tech domain. It lays a strong foundation for scalable, data-driven application development.