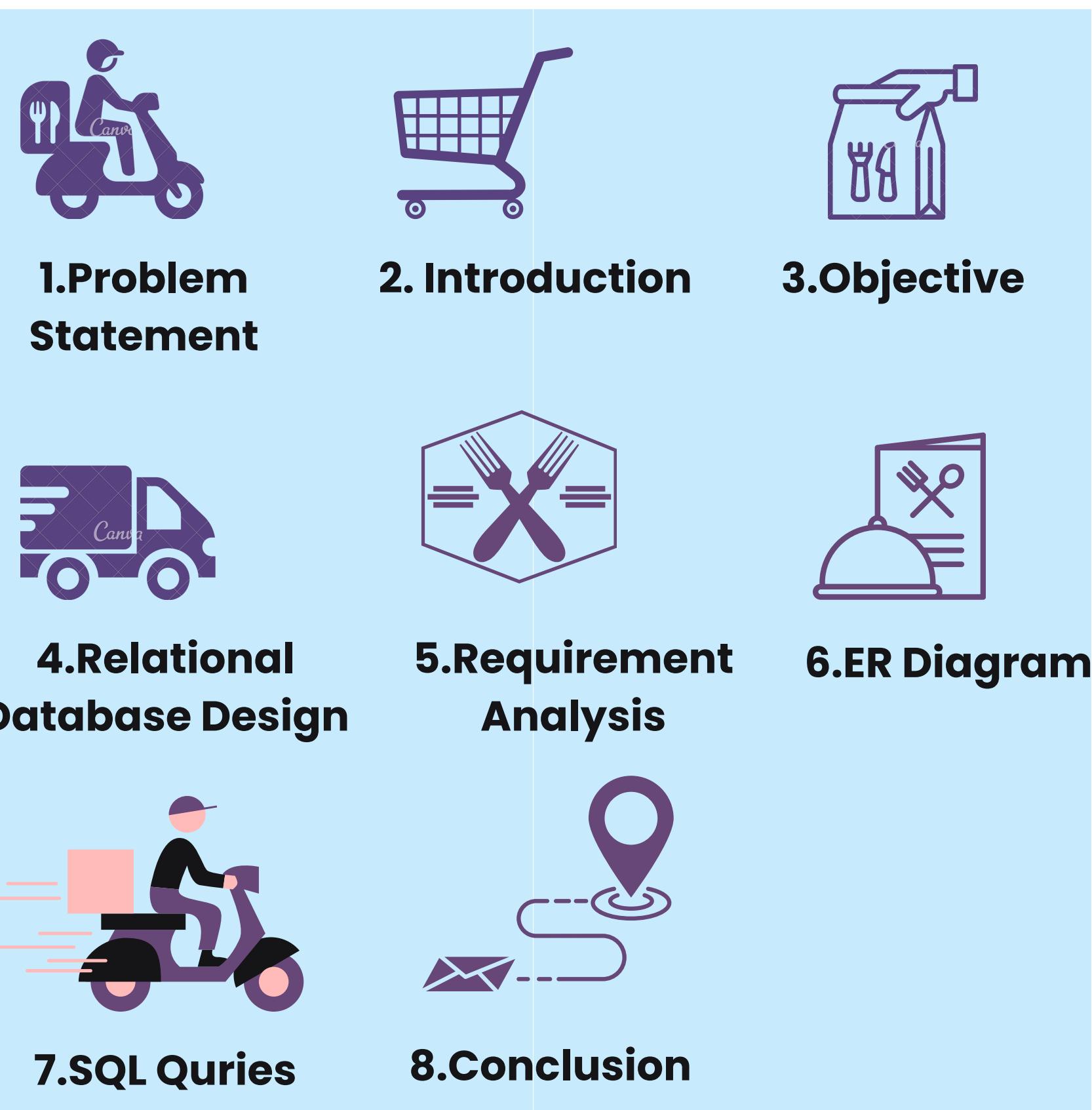
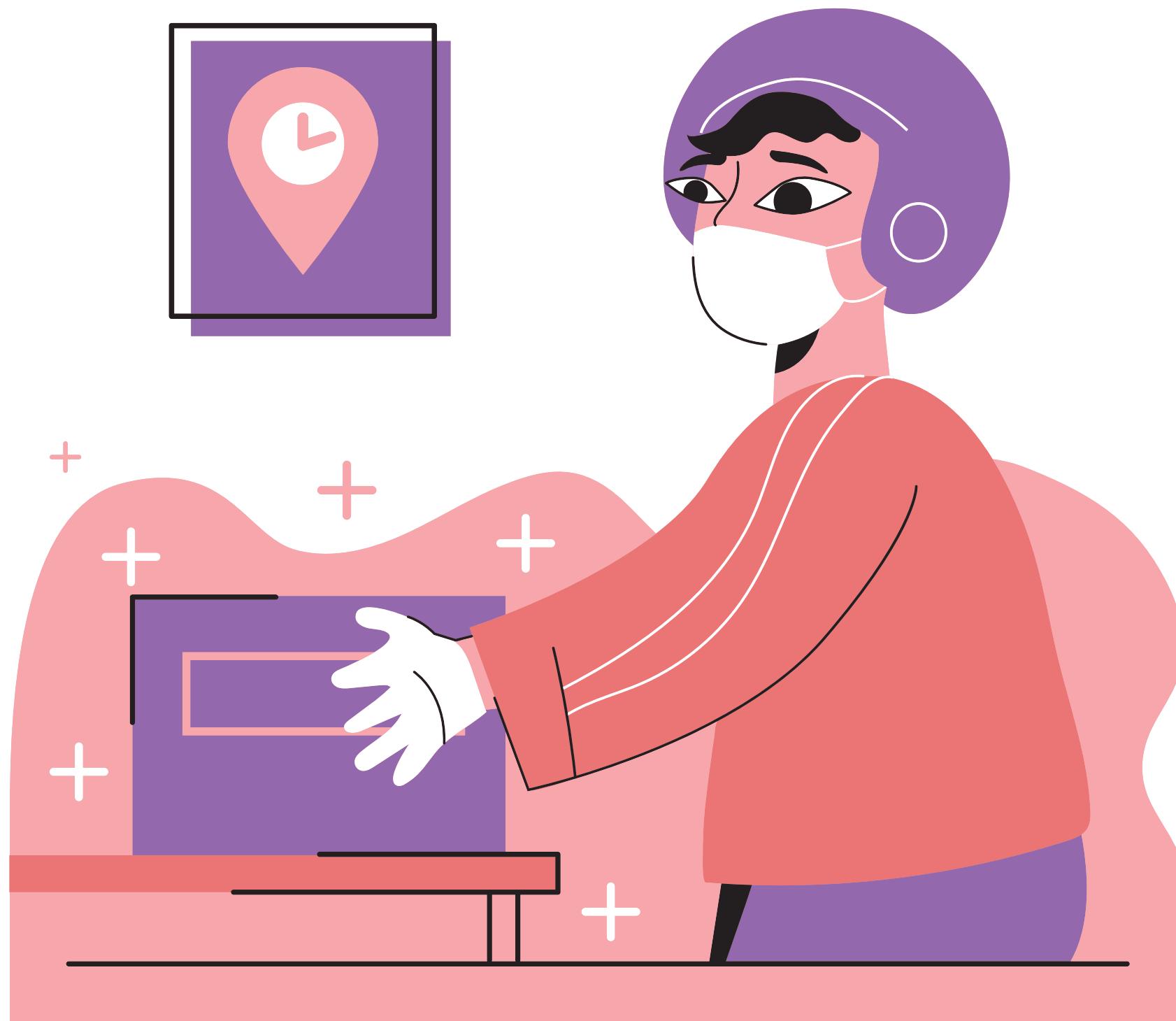


ONLINE FOOD DELIVERY SYSTEM

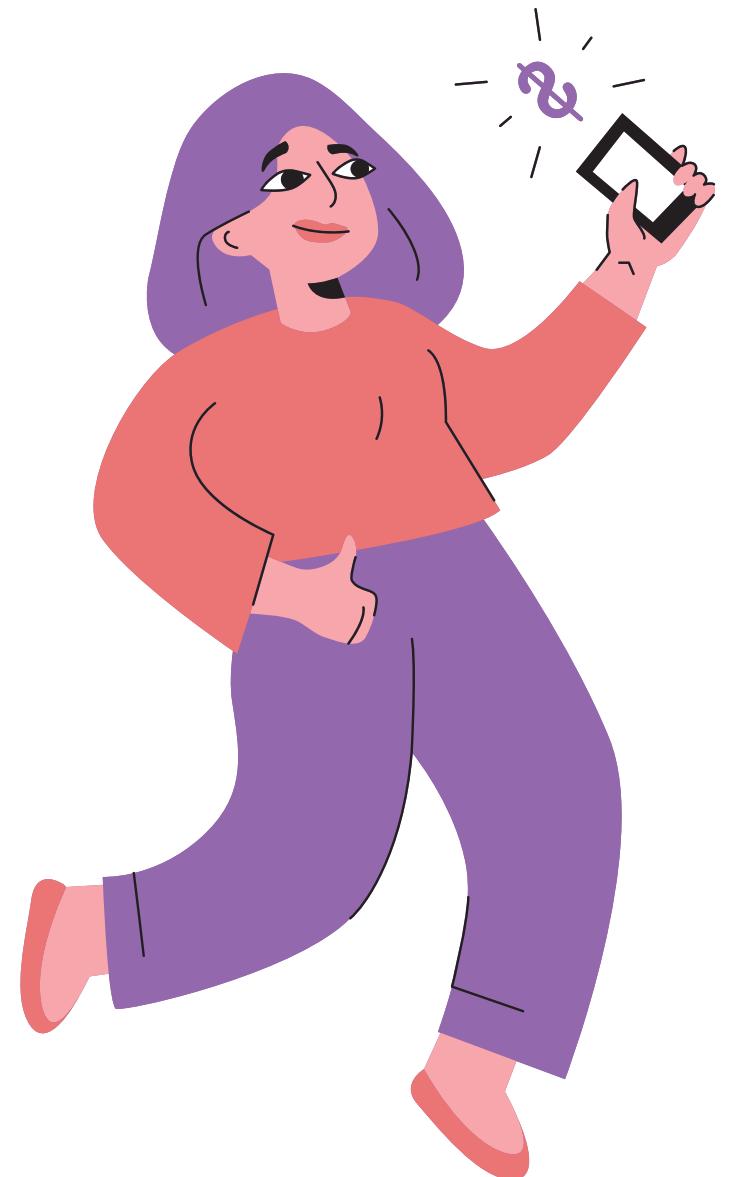
Unnati Pimple -46



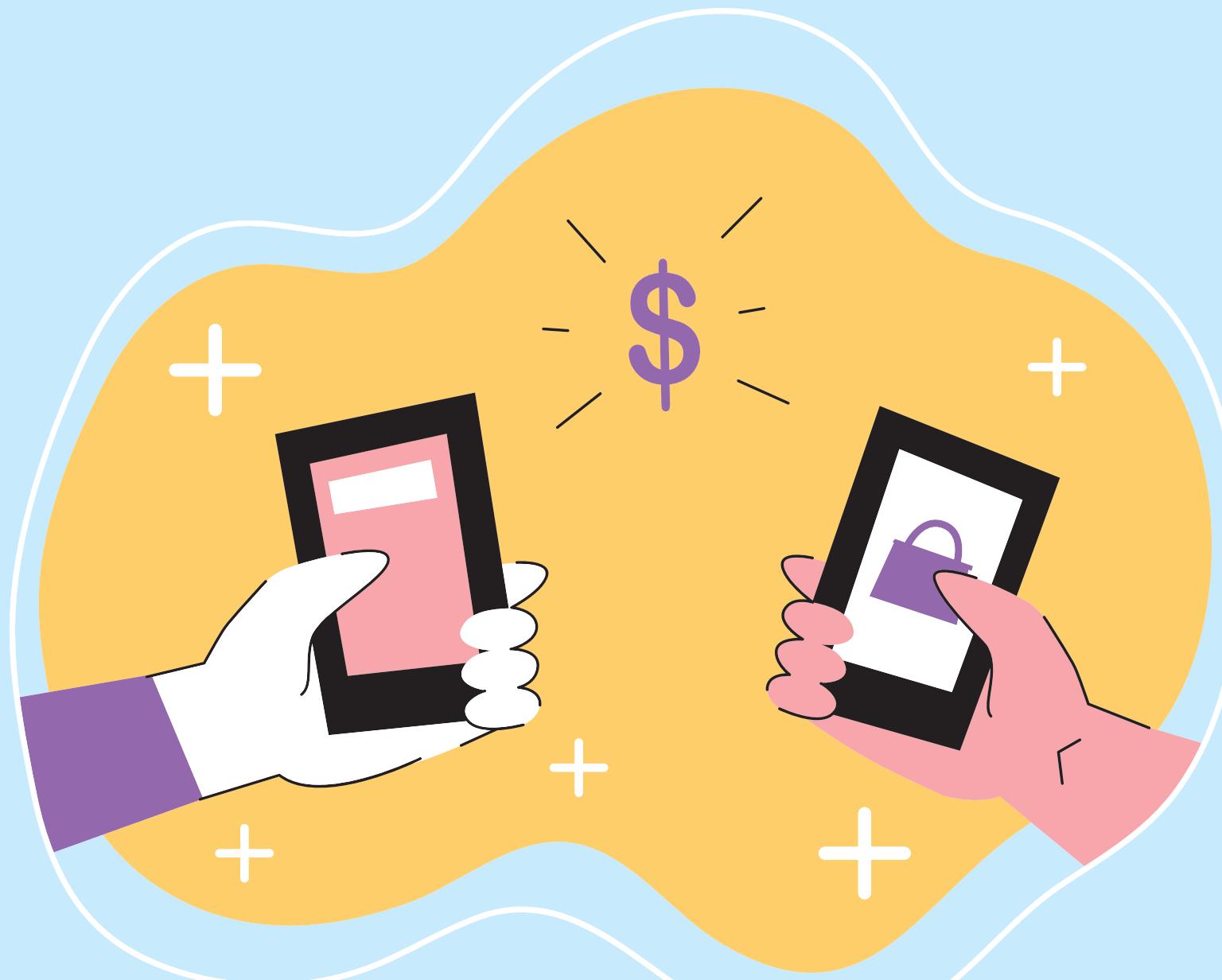


Problem Statement

Develop an efficient and robust database system for an online food delivery platform that enhances the overall user experience, optimizes order management, and ensures seamless communication between customers, restaurants, and delivery drivers.



Objective



Performance

Create a database system that can handle a high volume of concurrent transactions, ensuring fast response times for customers, restaurants, and delivery drivers.



Data Integrity

Implement mechanisms to maintain data consistency, integrity, and accuracy throughout the system, preventing order errors and duplicate entries.



Order Management

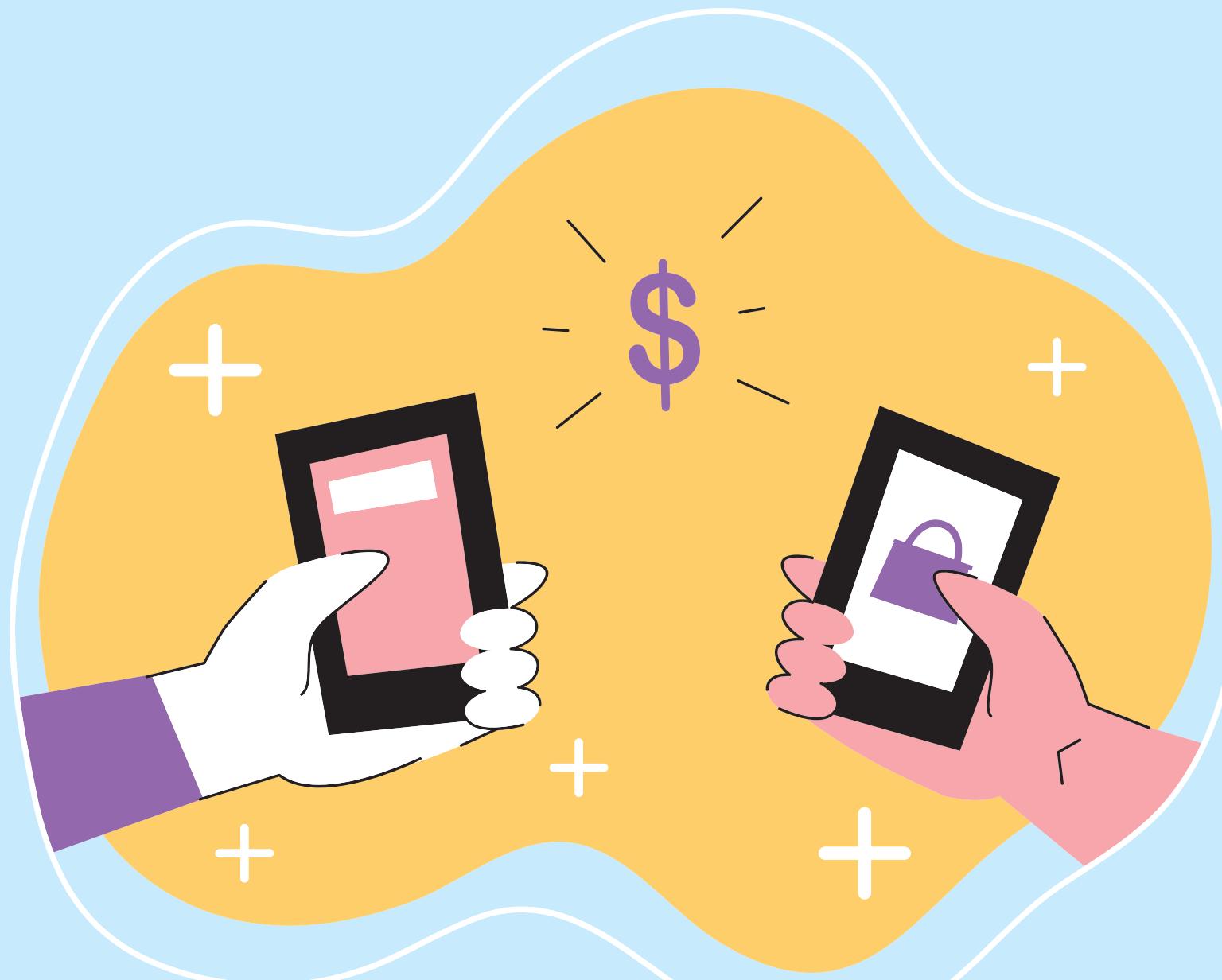
Develop a robust order management system that allows real-time order tracking, order history, and seamless order modification or cancellation.



Delivery Coordination

Facilitate efficient communication between customers, restaurants, and delivery drivers, ensuring timely and accurate order deliveries.

Objective



Restaurant Integration

Enable restaurants to manage their menus, inventory, and order processing efficiently through the database system.



Security

Implement robust security measures to protect user data, prevent unauthorized access, and secure sensitive information such as payment details.



User Experience

Enhance the overall user experience by reducing wait times, providing accurate recommendations, and offering personalized features based on user preferences and order history.



User Management

Streamline user registration, authentication, and profile management processes while ensuring data security and privacy.

Requirement Analysis

User Requirements:

Identify different user roles (e.g., customers, restaurants, delivery drivers, administrators).

Document the specific actions and tasks each user role should be able to perform.

Gather input from stakeholders and potential users to understand their expectations.

Functional Requirements:

Define the core functions of the system, such as user registration, menu browsing, order placement, payment processing, order tracking, and user reviews.

Specify any special features like promo code handling, multiple payment options, and location-based services.

Data Requirements:

Identify the types of data the system will store, including customer profiles, restaurant information, menus, orders, delivery addresses, and payment details.

Determine the relationships between different types of data (e.g., an order is associated with a customer and a restaurant).

Performance Requirements:

Define performance criteria, such as response times for various operations (e.g., searching for restaurants, placing an order).

Consider scalability requirements, including the expected number of users and transactions.

Relational Database Design

1. Customer Table

- Customer_name
- Email
- Phone Number
- Address

2. Restaurant Table

- Restaurant_id
- Name
- Address
- Phone Number
- Cuisine Type

3. Menu Categories

- Category_ID
- restaurant_id
- Description
- Name

4. OrderDetails Table :

- Customer_id
- OrderID
- Order_date

5. Payment Table

- PaymentID
- Order ID
- Payment Date
- Payment Amount
- Payment Method

6. DeliveryTable

- Assignment_id
- Orde_id
- Driver_id
- Assignment_date

Relational Database Design

7.Delivery_drivers

- driver_id
- first_name
- last_name
- phone_no
- vehicle_number

8. Menu Items

- Category_ID
- item_id
- item_name
- Description
- price

9.Order item Table :

- item_id
- OrderID
- quantity
- order_item_id



ER Diagram



SQL Queries and Commands



1. Create Table Customer

```
SQL File 5* x
CREATE TABLE customer (
    customer_id int AUTO_INCREMENT PRIMARY KEY,
    customer_name varchar(255) NOT NULL,
    email varchar(255) NOT NULL,
    phone varchar(10) NOT NULL,
    address text NOT NULL
)
```

2. Create Table Menu_items

```
SQL File 5* x
CREATE TABLE menu_items (
    item_id INT AUTO_INCREMENT PRIMARY KEY,
    category_id INT NOT NULL,
    name VARCHAR(255) NOT NULL,
    description TEXT,
    price DECIMAL(10, 2) NOT NULL,
    FOREIGN KEY (category_id) REFERENCES menu_categories(category_id)
)
```



3. View all contents of customers

```
5 • SELECT * FROM food_delivery_system.customers;
```

	customer_id	customer_name	email	phone	address
•	1	Amit Sharma	amit.sharma@email.com	1234567890	456 Rajput Lane, Delhi
	2	Priya Patel	priya.patel@email.com	9876543210	789 Gandhi Road, Mumbai
	3	Rajesh Kumar	rajesh.kumar@email.com	5555555555	123 Banerjee Street, Kolkata
	4	Meena Yadav	meena.yadav@email.com	7777777777	890 Lajpat Nagar, Delhi
	5	Rajiv Mishra	rajiv.mishra@email.com	8888888888	456 Juhu Beach, Mumbai
	6	Shweta Sinha	shweta.sinha@email.com	7777777777	234 Koramangala, Bangalore
	7	Amit Verma	amit.verma@email.com	8888888888	567 Malleshwaram, Bangalore
	8	Anjali Rajput	anjali.rajput@email.com	9999999999	789 HSR Layout, Bangalore
	9	Sunita Verma	sunita.verma@email.com	7777777777	567 Patel Nagar, Delhi
	10	Rahul Gupta	rahul.gupta@email.com	8888888888	789 Gandhi Road, Mumbai
	11	Deepak Singh	deepak.singh@email.com	9999999999	123 Sarojini Nagar, Bangalore
	NUL	NUL	NUL	NUL	NUL

4. View all contents of restaurants

```
18 • select * from dbmsproject.restaurants;
```

```
19
```

	restaurant_id	name	address	phone_number	cuisine_type
•	1	Chaat Corner	234 Chaat Street, Delhi	91-7777777777	Street Food
	2	Punjab Grill	567 Punjabi Lane, Mumbai	91-8888888888	Punjabi
	3	Dosai Diner	890 Dosa Road, Bangalore	91-9999999999	South Indian
	4	Tandoori Tadka	123 Tandoori Lane, Delhi	91-7777777777	Indian
	5	Spicy Biryani House	456 Biryani Road, Hyderabad	91-8888888888	Biryani



5. View all contents of menu_categories

```
21 • select * from dbmsproject.menu_categories;  
22  
23  
24
```

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

category_id	restaurant_id	name	description
1	1	Street Food	Delicious Indian street food items
2	2	Appetizers	Appetizing starters to tantalize your taste buds
3	3	South Indian	Authentic dishes from South India
4	4	Tandoori Specials	Tandoor-cooked dishes
5	5	Biryani	Flavorful rice dishes with spices

6. View all contents of orders

```
24 • select * from dbmsproject.orders;
```

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

	order_id	customer_id	restaurant_id	order_date	status
▶	1	1	1	2023-10-09 00:49:29	Placed
	2	2	2	2023-10-09 00:49:29	Placed
	3	3	3	2023-10-09 00:49:29	Placed
	4	4	4	2023-10-09 00:49:29	Placed
	5	5	5	2023-10-09 00:49:29	Placed
	6	6	1	2023-10-09 00:49:29	Placed
	7	7	2	2023-10-09 00:49:29	Placed
	8	8	3	2023-10-09 00:49:29	Placed



7. View all contents of order_items

```
24 •    select * from dbmsproject.order_items ;
```

25

Result Grid | Filter Rows: | Edit: |

	order_item_id	order_id	item_id	quantity
▶	1	1	1	2
	2	2	4	1
	3	3	6	3
	4	4	8	2
	5	5	10	1
	6	6	11	2
	7	7	14	2

8. View all contents of delivery_drivers

```
17 •    select * from dbmsproject.delivery_drivers;
```

18

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

	driver_id	first_name	last_name	phone_number	vehicle_plate_number
	1	Ravi	Kumar	91-9999999999	MH-01-AB-1234
	2	Sneha	Verma	91-7777777777	DL-02-CD-5678
	3	Vijay	Sharma	91-8888888888	KA-03-EF-9876



9. View all contents of delivery_assignments

```
24 • select * from dbmsproject.delivery_assignments;
```

```
25
```

```
26
```

assignment_id	order_id	driver_id	assignment_date	status
1	1	1	2023-10-09 00:54:17	Assigned
2	2	2	2023-10-09 00:54:17	Assigned
3	3	3	2023-10-09 00:54:17	Assigned
4	4	1	2023-10-09 00:54:17	Assigned
5	5	2	2023-10-09 00:54:17	Assigned
6	6	3	2023-10-09 00:54:17	Assigned
7	7	1	2023-10-09 00:54:17	Assigned
8	8	2	2023-10-09 00:54:17	Assigned

10. View all contents of payments

```
23 • select * from dbmsproject.payments;
```

```
24
```

payment_id	order_id	payment_date	total_amount	payment_status
1	1	2023-10-09 00:54:41	27.98	Completed
2	2	2023-10-09 00:54:41	11.99	Completed
3	3	2023-10-09 00:54:41	24.97	Completed
4	4	2023-10-09 00:54:41	30.98	Completed
5	5	2023-10-09 00:54:41	19.98	Completed
6	6	2023-10-09 00:54:41	17.99	Completed
7	7	2023-10-09 00:54:41	22.98	Completed
8	8	2023-10-09 00:54:41	16.99	Completed



11. View all contents of menu_items

```
26 • select * from dbmsproject.menu_items;
```

```
27
```

	item_id	category_id	name	description	price
1	1	Pani Puri	Crispy hollow balls filled with spicy tangy water	4.99	
2	1	Samosa Chaat	Samosa topped with chana masala and chutney	6.99	
3	2	Paneer Tikka	Tandoori-grilled cottage cheese with spices	9.99	
4	2	Chicken Seekh Kebab	Minced chicken kebabs cooked to perfection	11.99	
5	3	Masala Dosa	Thin rice crepe filled with spiced potatoes	7.99	
6	3	Idli Sambhar	Steamed rice cakes served with lentil soup	6.99	
7	4	Tandoori Chicken	Marinated chicken cooked in a tandoor	14.99	
8	4	Garlic Naan	Tandoor-baked garlic-flavored flatbread	3.99	
9	5	Chicken Biryani	Fragrant rice with chicken and spices	11.99	
10	5	Veg Biryani	Vegetable biryani with aromatic rice	9.99	
	HULL	HULL	HULL	HULL	HULL



12. Update the payment_status in table payments

```
SQL File 5* x menu_items food_delivery_system.menu_it... menu_items - Ti
 1 • UPDATE payments
 2     SET payment_status = 'Pending'
 3     WHERE order_id IN (5, 6);
 4
```

13. Alter table payments and add a column payment_type

```
SQL File 5* x menu_items food_delivery_system.menu_it... menu_items - Ti
 1 • ALTER TABLE payments
 2     ADD COLUMN payment_type VARCHAR(255) NOT NULL;
 3
 4 • SELECT * FROM food_delivery_system.payments;
```

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

	payment_id	order_id	payment_date	total_amount	payment_status	payment_type
▶	1	1	2023-10-07 02:12:58	27.98	Completed	card
	2	2	2023-10-07 02:12:58	11.99	Completed	UPI
	3	3	2023-10-07 02:12:58	18.98	Completed	card
	4	4	2023-10-08 21:04:40	30.98	Completed	UPI
	5	5	2023-10-08 21:04:40	19.98	Pending	UPI
	6	6	2023-10-08 21:04:40	17.99	Pending	Cash
	7	7	2023-10-08 21:04:40	22.98	Completed	Cash
	8	8	2023-10-08 21:04:40	16.99	Completed	Cash
*	HULL	HULL	HULL	HULL	HULL	HULL



13. Delete an item from menu_items

```
1  DELETE FROM menu_items where item_id = 39;
4
5
6 •  SELECT * FROM food_delivery_system.menu_items;
```

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

item_id	category_id	name	description	price
19	11	Samosa	Fried pastry filled with spiced potatoes	4.99
20	11	Paneer Tikka	Tandoori-grilled cottage cheese	8.99
21	12	Chicken Tikka Masala	Tender chicken in a creamy tomato sauce	12.99
22	12	Dal Makhani	Lentils cooked in buttery tomato sauce	10.99
23	13	Tandoori Chicken	Marinated chicken cooked in a tandoor	14.99
24	14	Naan	Soft and fluffy Indian bread	2.99
25	14	Roti	Whole wheat flatbread	2.49
26	15	Chicken Biryani	Fragrant rice with chicken and spices	11.99
27	15	Veg Biryani	Vegetable biryani with aromatic rice	9.99
28	16	Pani Puri	Crispy hollow balls filled with spicy tangy water	4.99
29	16	Samosa Chaat	Samosa topped with chana masala and chutney	6.99
30	12	Chicken Seekh Kebab	Minced chicken kebabs cooked to perfection	11.99
33	13	Garlic Naan	Tandoor-baked garlic-flavored flatbread	3.99
34	16	Pani Puri	Crispy hollow balls filled with spicy tangy water	4.99
35	16	Samosa Chaat	Samosa topped with chana masala and chutney	6.99
36	12	Chicken Seekh Kebab	Minced chicken kebabs cooked to perfection	11.99



14. Drop table delivery_assignments

```
1 •  DROP TABLE delivery_assignments;
2
```

15. Alter the table and rename a column

```
23 • alter table orders  
24     rename column status to order_status;  
25 • select * from dbmsproject.orders;
```

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

order_id	customer_id	restaurant_id	order_date	order_status
1	1	1	2023-10-09 00:49:29	Placed
2	2	2	2023-10-09 00:49:29	Placed
3	3	3	2023-10-09 00:49:29	Placed
4	4	4	2023-10-09 00:49:29	Placed
5	5	5	2023-10-09 00:49:29	Placed

16. Between condition

```
select order_id, total_amount, payment_status from dbmsproject.payments  
where payment_status='pending' and total_amount between 15.99 and 17.99;
```

Filter Rows: Export: Wrap Cell Content: Fetch rows:

id	total_amount	payment_status
	17.99	Pending
	16.99	Pending



17. OR condition

```
32 •   SELECT restaurant_id, name FROM dbmsproject.menu_categories  
33      where name like 'B%' or name like 'A%';  
34  
35      /*SELECT count(description) FROM dbmsproject.menu_categories where des  
..
```

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

	restaurant_id	name
▶	2	Appetizers
	5	Biryani

18. NOT LIKED condition

```
20 •   select first_name, last_name from dbmsproject.delivery_drivers  
21      where last_name not like '%r';  
22
```

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

	driver_id	first_name	last_name	phone_number	vehidle_plate_number
▶	1	Ravi	Kumar	91-9999999999	MH-01-AB-1234
	2	Sneha	Verma	91-7777777777	DL-02-CD-5678
	3	Vijay	Sharma	91-8888888888	KA-03-EF-9876
	4	Ravi	Kumar	91-9999999999	MH-01-AB-1234
	5	Sneha	Verma	91-7777777777	DL-02-CD-5678



19. NOT Between condition

```
30 • select order_id, item_id, quantity from dbmsproject.order_items  
31      where quantity not between 2 and 3 ;  
32  
33
```

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

	order_id	item_id	quantity
.	2	4	1
.	5	10	1
.	8	16	1

20. SUM

```
31  
32 • select sum(total_amount) from payments  
33      where payment_status='Completed';  
34
```

Result Grid | Filter Rows: Export: Wrap Ce

	sum(total_amount)
.	2110.33



21. Average

```
31 • select name, avg(price) from dbmsproject.menu_items where name='Pani Puri';
```

```
32
```

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

	name	avg(price)
▶	Pani Puri	43.561429

22. MAX

```
32 • select max(total_amount) from dbmsproject.payments where payment_status='Completed';
```

```
33
```

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

	max(total_amount)
▶	30.98

23. MIN

```
31 • select min(total_amount) from dbmsproject.payments where payment_status='Completed';
```

```
32
```

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

	min(total_amount)
▶	11.99



24. ORDER BY

```
7 • SELECT *  
8   FROM payments  
9   ORDER BY total_amount DESC;
```

	payment_id	order_id	payment_date	total_amount	payment_status	payment_type
▶	4	4	2023-10-08 21:04:40	30.98	Completed	UPI
	1	1	2023-10-07 02:12:58	27.98	Completed	card
	7	7	2023-10-08 21:04:40	22.98	Completed	Cash
	5	5	2023-10-08 21:04:40	19.98	Pending	UPI
	3	3	2023-10-07 02:12:58	18.98	Completed	card
	6	6	2023-10-08 21:04:40	17.99	Pending	Cash
	8	8	2023-10-08 21:04:40	16.99	Completed	Cash
	2	2	2023-10-07 02:12:58	11.99	Completed	UPI
●	HULL	HULL	HULL	HULL	HULL	HULL

25. GROUP BY

```
6 • SELECT payment_type, COUNT(*) AS count_of_payments, SUM(total_amount) AS total_amount  
7   FROM payments  
8   GROUP BY payment_type;
```

	payment_type	count_of_payments	total_amount
▶	card	2	46.96
	UPI	3	62.95
	Cash	3	57.96



26.COUNT

```
6 •  SELECT payment_type, COUNT(*) AS count_of_payments, SUM(total_amount) AS total_amount
7   FROM payments
8   GROUP BY payment_type;
```

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

	payment_type	count_of_payments	total_amount
▶	card	2	46.96
	UPI	3	62.95
	Cash	3	57.96

27.Rename the table name

SQL File 5* | menu_items | food_delivery_system.menu_it... | menu_items - Table | menu_items | menu_categories

File Edit View Insert Object Tools Help

1 • RENAME TABLE customer TO customers;

2 • SELECT * FROM customers;

3 |

4

5

6

7

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

	customer_id	customer_name	email	phone	address
▶	1	Amit Sharma	amit.sharma@email.com	1234567890	456 Rajput Lane, Delhi
	2	Priya Patel	priya.patel@email.com	9876543210	789 Gandhi Road, Mumbai
	3	Rajesh Kumar	rajesh.kumar@email.com	5555555555	123 Banerjee Street, Kolkata
	4	Meena Yadav	meena.yadav@email.com	7777777777	890 Lajpat Nagar, Delhi
	5	Rajiv Mishra	rajiv.mishra@email.com	8888888888	456 Juhu Beach, Mumbai
	6	Shweta Sinha	shweta.sinha@email.com	7777777777	234 Koramangala, Bangalore
	7	Amit Verma	amit.verma@email.com	8888888888	567 Malleshwaram, Bangalore
	8	Anjali Rajput	anjali.rajput@email.com	9999999999	789 HSR Layout, Bangalore
	9	Sunita Verma	sunita.verma@email.com	7777777777	567 Patel Nagar, Delhi
	10	Rahul Gupta	rahul.gupta@email.com	8888888888	789 Gandhi Road, Mumbai
	11	Deepak Singh	deepak.singh@email.com	9999999999	123 Sarojini Nagar, Bangalore
*	HULL	HULL	HULL	HULL	HULL



28.Natural Join

```
1 • SELECT orders.customer_id,customers.customer_name,orders.status  
2   FROM orders  
3   NATURAL JOIN Customers;
```

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

	customer_id	customer_name	status
▶	1	Sunita Verma	Placed
	2	Rahul Gupta	Placed
	3	Deepak Singh	Placed
	4	Meena Yadav	Placed
	5	Rajiv Mishra	Placed
	6	Shweta Sinha	Placed

29.Inner Join

SQL File 5* | menu_items menu_items menu_categories order_items payments payments menu_items order_items delivery_assignmen

File Edit View Insert Tools Options Help

1 • SELECT orders.order_id, customers.customer_name AS CustomerName, restaurants.name AS RestaurantName
2 FROM orders
3 INNER JOIN customers ON orders.customer_id = customers.customer_id
4 INNER JOIN restaurants ON orders.restaurant_id = restaurants.restaurant_id;
5

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

	order_id	CustomerName	RestaurantName
▶	1	Amit Sharma	Spice Palace
	2	Priya Patel	Tandoori Delights
	3	Rajesh Kumar	Biryani King
	4	Amit Sharma	Spice Palace
	5	Priya Patel	Tandoori Delights
	6	Rajesh Kumar	Biryani King
	7	Meena Yadav	Chaat Corner
	8	Rajiv Mishra	Punjab Grill
	9	Shweta Sinha	Spice Palace
	10	Amit Verma	Tandoori Delights
	11	Anjali Rajput	Biryani King



30. Right Outer Join

```
3 • SELECT orders.order_id, restaurants.name  
4   FROM orders  
5   RIGHT JOIN restaurants ON orders.restaurant_id = restaurants.restaurant_id;
```

Result Grid	
order_id	name
1	Spice Palace
4	Spice Palace
9	Spice Palace
2	Tandoori Delights
5	Tandoori Delights
10	Tandoori Delights
3	Biryani King
6	Biryani King
11	Biryani King
7	Chaat Corner
8	Punjab Grill
NUL	Dosai Diner
NUL	Tandoori Tadka

31. Left Outer join

```
3 • SELECT restaurants.name, orders.order_id  
4   FROM restaurants  
5   LEFT JOIN orders ON restaurants.restaurant_id = orders.restaurant_id;
```

Result Grid	
name	order_id
Spice Palace	1
Spice Palace	4
Spice Palace	9
Tandoori Delights	2
Tandoori Delights	5
Tandoori Delights	10
Biryani King	3
Biryani King	6
Biryani King	11
Chaat Corner	7
Punjab Grill	8
Dosai Diner	NUL
Tandoori Tadka	NUL



32. Create View

```
1 • CREATE VIEW restaurant_view AS
2   SELECT restaurant_id, name AS restaurant_name, address, phone_number, cuisine_type
3   FROM restaurants;
4
5 • SELECT restaurant_name, address FROM restaurant_view;
```

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

restaurant_name	address
Spice Palace	123 Curry Lane, Delhi
Tandoori Delights	456 Tandoori Road, Mumbai
Biryani King	789 Dum Biryani Street, Hyderabad
Chaat Corner	234 Chaat Street, Delhi
Punjab Grill	567 Punjabi Lane, Mumbai
Dosa Diner	890 Dosa Road, Bangalore
Tandoori Tadka	123 Tandoori Lane, Delhi
Spicy Biryani House	456 Biryani Road, Hyderabad

33. Replace View

```
1 • CREATE OR REPLACE VIEW restaurant_view AS
2   SELECT restaurant_id, name AS restaurant_name, address, phone_number, cuisine_type
3   FROM restaurants
4   WHERE cuisine_type = "indian";
5 • SELECT * FROM restaurant_view;
```

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

restaurant_id	restaurant_name	address	phone_number	cuisine_type
2	Tandoori Delights	456 Tandoori Road, Mumbai	8888888888	Indian
3	Biryani King	789 Dum Biryani Street, Hyderabad	7777777777	Indian
7	Tandoori Tadka	123 Tandoori Lane, Delhi	7777777777	Indian



34. Indexes

```
39 • show index from menu_categories;
40
```

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality
menu_categories	0	PRIMARY	1	category_id	A	15
menu_categories	1	restaurant_id	1	restaurant_id	A	5
menu_categories	1	idx_restaurant_name	1	name	A	8

35. FINDING OUT A RESTAURANT WHERE STREET FOOD IS AVAILABLE

```
1 select * from restaurants where cuisine_type='Street Food';
```

restaurant_id	name	address	phone_number	cuisine_type
1	Chaat Corner	234 Chaat Street, Delhi	91-7777777777	Street Food



36. Total revenue collected by a restaurant

```
1 •  select sum( price) AS totalrevenue  
2      from menu_items  
3      where category_id='3';
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content
totalrevenue	14.98			



CONCLUSION

In conclusion, the online food delivery database we've discussed today plays a pivotal role in the success and efficiency of our online food delivery system. We've explored various aspects of the database, from its design to key features and its importance in delivering a seamless experience to our customers, restaurant partners, and delivery teams.



**THANK
YOU!!**

