Hello!

hello my name is Saurabh Das
in this project i have utilize SQL queries to solve questions that were related to pizza sales

1:- Retrieve the total number of orders placed.

2:-Calculate the total revenue generated from pizza sales.

3:-Identify the highest-priced pizza.

4:-Identify the most common pizza size ordered.

5:-list the top 5 most ordered pizza types along with their quantities

6:-Join the necessary tables to find the total quantity of each pizza category ordered

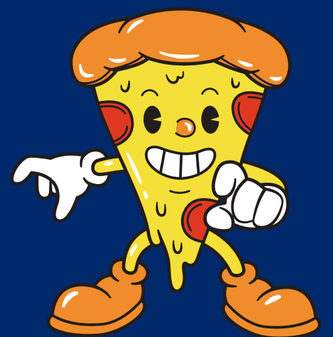
7:-Determine the distribution of orders by hour of the day.

8:-Join relevant tables to find the category-wise distribution of pizzas

9:-Group the orders by date and calculate the average number of pizzas ordered

10:-Determine the top 3 most ordered pizza types based on revenue.

11:-Calculate the percentage contribution of each pizza type to total revenue.

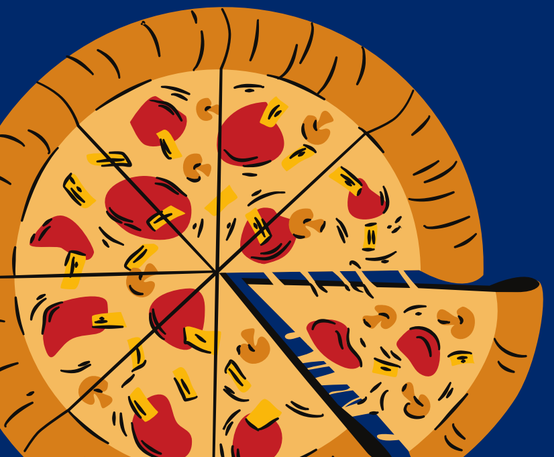


Retrieve the total number of orders placed.

```sql
SELECT
    COUNT(order_id)
FROM
    orders;
```

Result Grid

	count(order_id)
▶	21350



Calculate the total revenue generated from pizza sales

```sql
SELECT
    ROUND(SUM(pizzas.price * orders_details.quantity),
            2) AS total_sale
FROM
    orders_details
        JOIN
    pizzas ON pizzas.pizza_id = orders_details.pizza_id;
```

Result Grid

	total_sale
▶	817860.05

Identify the highest-priced pizza

```sql
SELECT
    pizza_types.name, pizzas.price
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
ORDER BY price DESC
LIMIT 1;
```

Result Grid | Filter Rows:

	name	price
▶	The Greek Pizza	35.95

Identify the most common pizza size ordered.

```sql
SELECT
    pizzas.size,
    COUNT(orders_details.order_details_id) AS order_count
FROM
    pizzas
        JOIN
    orders_details ON pizzas.pizza_id = orders_details.pizza_id
GROUP BY size
ORDER BY order_count DESC
LIMIT 1;
```

Result Grid | Filter Rows:

size	order_count
L	18526

list the top 5 most ordered pizza types along with their quantities

```
    pizza_types.name, SUM(orders_details.quantity) AS total
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    orders_details ON orders_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY total DESC
LIMIT 5;
```

Result Grid | Filter Rows:

name	total
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	2422
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371

Join the necessary tables to find the total quantity of each pizza category ordered.

```sql
SELECT
    pizza_types.category,
    SUM(orders_details.quantity) AS total_sale
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    orders_details ON orders_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
ORDER BY total_sale DESC;
```

Result Grid | Filter Rows:

category	total_sale
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

Determine the distribution of orders by hour of the day.

```sql
SELECT
    HOUR(order_time), COUNT(order_id)
FROM
    orders
GROUP BY HOUR(order_time);
```

Result Grid | Filter Rows:

HOUR(order_time)	COUNT(order_id)
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642
21	1198
22	663
23	28
10	8
9	1

Join relevant tables to find the category-wise distribution of pizzas

```sql
SELECT
    category, COUNT(name) AS count_
FROM
    pizza_types
GROUP BY category;
```

Result Grid

category	count_
Chicken	6
Classic	8
Supreme	9
Veggie	9

Group the orders by date and calculate the average number of pizzas ordered

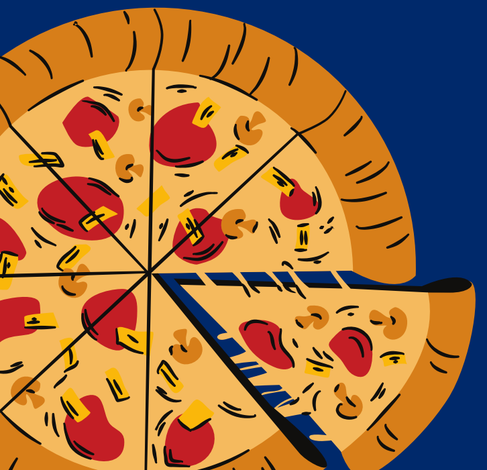
```sql
SELECT
    ROUND(AVG(all_orders))
FROM
    (SELECT
        orders.order_date,
            SUM(orders_details.quantity) AS all_orders
    FROM
        orders
    JOIN orders_details ON orders.order_id = orders_details.order_id
    GROUP BY orders.order_date) AS order_all;
```

Result Grid | Filter Rows

	ROUND(AVG(all_orders))
▶	138

Determine the top 3 most ordered pizza types based on revenue.

```sql
SELECT
    pizza_types.name,
    SUM(orders_details.quantity * pizzas.price) AS revenue
FROM
    pizza_types
        JOIN
    pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
        JOIN
    orders_details ON pizzas.pizza_id = orders_details.pizza_id
GROUP BY pizza_types.name
ORDER BY revenue DESC
LIMIT 3;
```



Result Grid | Filter Rows:

name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5

Calculate the percentage contribution of each pizza type to total revenue.

```
SELECT
    pizza_types.category,
    ROUND(SUM(orders_details.quantity * pizzas.price) / (SELECT
                    SUM(orders_details.quantity * pizzas.price)
                FROM
                    orders_details
                        JOIN
                    pizzas ON pizzas.pizza_id = orders_details.pizza_id) *
            2) AS revenue_percentage
FROM
    pizza_types
        JOIN
    pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
        JOIN
    orders_details ON pizzas.pizza_id = orders_details.pizza_id
GROUP BY pizza_types.category
ORDER BY revenue_percentage DESC
LIMIT 3;
```

Result Grid | Filter Rows:

category	revenue_percentage
Classic	26.91
Supreme	25.46
Chicken	23.96

Insights

- ***Sales Trends:*** The project identified which pizza types and sizes were most popular, indicating customer preferences.
- ***Revenue Analysis:*** It highlighted the pizza types contributing the most to overall revenue, which can guide future marketing and sales strategies.
- ***Order Patterns:*** The analysis of order distribution by time of day can help optimize operations during peak hours.

THANK YOU