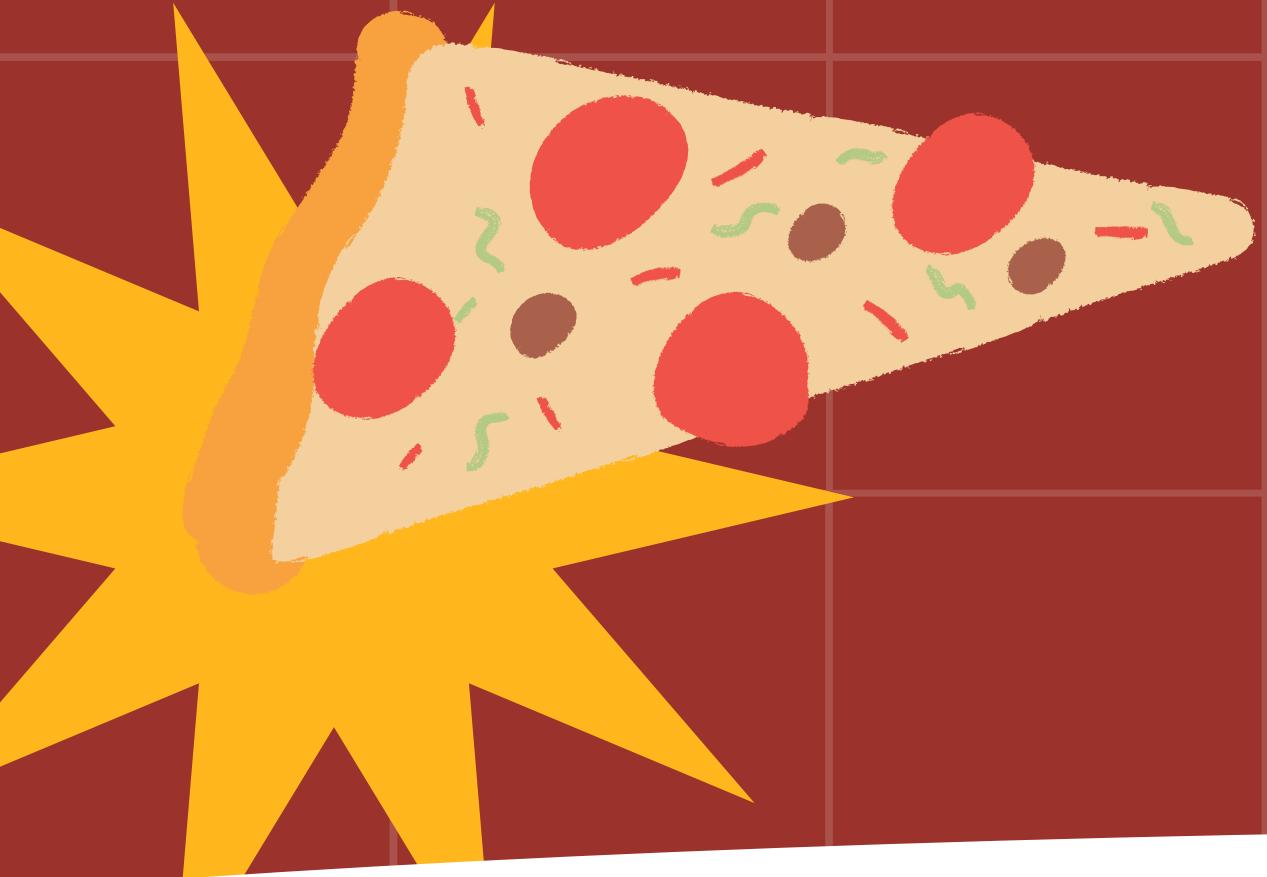


PIZZA SALES



HELLO,
I'M GAÚRAV SONKAR

I HAVE UTILIZE SQL QUERIES TO SOLVE QUESTION
THAT WERE RELATED TO PIZZA SALES



INTRODUCTION

Welcome!

This presentation will explore pizza sales data using SQL.
We'll uncover insights like total orders, revenue, popular
pizzas, and customer preferences.
about your pizza business.

Get ready!

We'll dive into the data and reveal valuable information

QUESTIONS

To make valueable insights from datasets or tables :



- Retrieve the total number of orders placed.
- Calculate the total revenue generated from pizza sales.
- Identify the highest-priced pizza.
- Identify the most common pizza size ordered.
- List the top 5 most ordered pizza types along with their quantities.
- Join the necessary tables to find the total quantity of each pizza category ordered.
- Determine the distribution of orders by hour of the day.
- Join relevant tables to find the category-wise distribution of pizzas.
- Group the orders by date and calculate the average number of pizzas ordered per day.
- Determine the top 3 most ordered pizza types based on revenue.
- Calculate the percentage contribution of each pizza type to total revenue.
- Analyze the cumulative revenue generated over time.
- Determine the top 3 most ordered pizza types based on revenue for each pizza category.

QUESTION 1

Retrieve the total number of orders placed.

ANS

```
SELECT  
COUNT(order_id) AS total_orders  
FROM  
orders;
```

Result Grid	
	total_orders
▶	21350

QUESTION 2

Calculate the total revenue generated from pizza sales.

ANS

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

Result Grid	
	total_sales
▶	817860.05

QUESTION 3

Identify the highest-priced pizza.

ANS

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

Result Grid | Filter Rows

	name	price
▶	The Greek Pizza	35.95

QUESTION 4

Identify the most common pizza size ordered.

ANS

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

Result Grid		Filter
	size	order_count
▶	L	18526

QUESTION 5

List the top 5 most ordered pizza types along with their quantities.

ANS

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

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

QUESTION 6

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

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

	category	quantity
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050

QUESTION 7

Determine the distribution of orders by hour of the day.

ANS

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

	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

QUESTION 8

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

ANS

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

Result Grid | Filter Rows:

	category	COUNT(name)
▶	Chicken	6
	Classic	8
	Supreme	9
	Veggie	9

QUESTION 9

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

ANS

```
SELECT ROUND(AVG(quantity), 0) as  
avgerage_pizza_order_per_day  
FROM (SELECT  
orders.order_date,  
SUM(order_details.quantity) AS quantity  
FROM order_details  
JOIN orders ON orders.order_id =  
order_details.order_id  
GROUP BY orders.order_date)  
AS order_quantity;
```

	Result Grid	Filter Rows:
	avgerage_pizza_order_per_day	138

QUESTION 10

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

ANS

```
select pizza_types.name ,  
sum(order_details.quantity*pizzas.price) as  
revenue  
from pizza_types join pizzas  
on pizza_types.pizza_type_id =  
pizzas.pizza_type_id  
join order_details  
on order_details.pizza_id = pizzas.pizza_id  
group by pizza_types.name  
order by revenue desc limit 3;
```

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

QUESTION 11

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

ANS

```
select pizza_types.category ,  
round((sum(order_details.quantity*pizzas.price))/  
     (selectround(sum(  
          order_details.quantity*pizzas.price),2) as  
           total_sales from order_details join pizzas on  
           pizzas.pizza_id = order_details.pizza_id ))*100,2) as  
      revenue from pizza_types join pizzas  
      on pizza_types.pizza_type_id =  
           pizzas.pizza_type_id  
      join order_details on order_details.pizza_id =  
           pizzas.pizza_id group by pizza_types.category  
      order by revenue desc;
```

Result Grid | Filter

	category	revenue
▶	Classic	26.91
	Supreme	25.46
	Chicken	23.96
	Veggie	23.68

QUESTION 12

Analyze the cumulative revenue generated over time.

ANS

```
select order_date ,  
sum(revenue) over(order by order_date) as  
cum_revenue from  
(select orders.order_date ,  
sum(order_details.quantity*pizzas.price) as  
revenue  
from order_details join pizzas  
on order_details.pizza_id = pizzas.pizza_id  
join orders  
on orders.order_id = order_details.order_id  
group by orders.order_date) as sales;
```

order_date	cum_revenue
2015-01-01	2713.850000000004
2015-01-02	5445.75
2015-01-03	8108.15
2015-01-04	9863.6
2015-01-05	11929.55
2015-01-06	14358.5
2015-01-07	16560.7
2015-01-08	19399.05
2015-01-09	21526.4
2015-01-10	23990.35000000002
2015-01-11	25862.65
2015-01-12	27781.7
2015-01-13	29831.30000000003
2015-01-14	32358.70000000004
2015-01-15	34343.50000000001
2015-01-16	36937.65000000001
2015-01-17	39001.75000000001
2015-01-18	40078.60000000005

QUESTION 13

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

ANS

```
select name , revenue from
(select category , name ,revenue ,
rank() over(partition by category order by revenue
desc ) as rn from
(select pizza_types.category , pizza_types.name ,
sum(order_details.quantity*pizzas.price) as
revenue from pizza_types join pizzas
on pizza_types.pizza_type_id =
pizzas.pizza_type_id
join order_details
on pizzas.pizza_id = order_details.pizza_id
group by pizza_types.category , pizza_types.name)
as a) as b
where rn<=3;
```

name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5
The Classic Deluxe Pizza	38180.5
The Hawaiian Pizza	32273.25
The Pepperoni Pizza	30161.75
The Spicy Italian Pizza	34831.25
The Italian Supreme Pizza	33476.75
The Sicilian Pizza	30940.5
The Four Cheese Pizza	32265.70000000065
The Mexicana Pizza	26780.75
The Five Cheese Pizza	26066.5

INSIGHTS

Making Data-Driven Decisions for Your
Pizza Business



- Peak Sales Periods: 12 O'Clock
- Popular Pizza Categories : Classic
- Best-Selling Pizzas : The Classic Deluxe Pizza
- Customer Preferences : large size
- Revenue Analysis : 81786.05
- Ordered Pizza Based on Revenue : The thai chicken pizza



THANK YOU

“Have fun making your own pizza
and enjoy every bite”