

# PIZZA SALES REPORT

20 JUNE, 2024



# INTRODUCTION

In this presentation, we'll explore pizza sales data using SQL. We'll see SQL code and result for many queries. The sales data is taken from Kaggle and query questions are taken from WsCube tech.

# PROBLEMS

01

Retrieve total number of orders placed

02

Calculate total revenue generated from pizza sales

03

Identify the highest-priced pizza

04

Identify the most common pizza size ordered

05

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

06

Find the total quantity of each pizza category ordered

07

Determine the distribution of orders by hour of the day

08

Category wise distribution of pizzas

09

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

10

Find top 3 pizzas based on revenue

11

Calculate the percentage contribution of each pizza category to total revenue

12

Analyze the cumulative revenue generated over time


# RETRIEVE TOTAL NUMBER OF ORDERS PLACED

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

	total_orders
▶	21350

# CALCULATE TOTAL REVENUE GENERATED FROM PIZZA SALES

**SELECT**

 ROUND(SUM(order\_details.quantity \* pizzas.price),  
2) AS Revenue

**FROM**

order\_details

**JOIN**

pizzas ON order\_details.pizza\_id = pizzas.pizza\_id;

	Revenue
▶	817860.05



# IDENTIFY THE HIGHEST-PRICED PIZZA

```
SELECT
    pizza_id AS Highest_priced_pizza, price
FROM
    pizzas
ORDER BY price DESC
LIMIT 1;
```

	Highest_priced_pizza	price
▶	the_greek_xxl	35.95

# IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED

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

	size	Total_Orders
▶	L	18956
	M	15635
	S	14403
	XL	552
	XXL	28

# LIST THE TOP 5 MOST ORDERED PIZZA TYPES ALONG WITH THEIR QUANTITIES

```
SELECT
    pizza_types.name AS Pizza_Type,
    SUM(order_details.quantity) AS Total_Orders
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 Total_Orders DESC
LIMIT 5;
```

	Pizza_Type	Total_Orders
►	The Classic Deluxe Pizza	2453
	The Barbecue Chicken Pizza	2432
	The Hawaiian Pizza	2422
	The Pepperoni Pizza	2418
	The Thai Chicken Pizza	2371



# FIND THE TOTAL QUANTITY OF EACH PIZZA CATEGORY ORDERED

```
SELECT
    pizza_types.category,
    SUM(order_details.quantity) AS Total_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 Total_quantity DESC;
```

	category	Total_quantity
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050

# DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY

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

	hour_of_day	total_orders
▶	11	1231
	12	2520
	13	2455
	14	1472
	15	1468
	16	1920
	17	2336
	18	2399

# CATEGORY WISE DISTRIBUTION OF PIZZAS

**SELECT**

category, COUNT(name) Number\_of\_pizzas

**FROM**

pizza\_types

**GROUP BY** category;

	category	Number_of_pizzas
▶	Chicken	6
	Classic	8
	Supreme	9
	Veggie	9

# GROUP THE ORDERS BY DATE AND CALCULATE THE AVERAGE NUMBER OF PIZZAS ORDER PER DAY

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

	Avg_quantity_per_day
▶	138

# FIND TOP 3 PIZZAS BASED ON REVENUE

```
SELECT
    pizza_types.name,
    ROUND(SUM(order_details.quantity * pizzas.price),
          2) AS Revenue
FROM
    order_details
    JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id
    JOIN
    pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_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

# CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA CATEGORY TO TOTAL REVENUE

```
SELECT
    pizza_types.category,
    ROUND((SUM(order_details.quantity * pizzas.price) / (SELECT
        SUM(order_details.quantity * pizzas.price) AS Total_sales
    FROM
        order_details
        JOIN
        pizzas ON order_details.pizza_id = pizzas.pizza_id))) * 100,
    2) AS revenue
FROM
    order_details
    JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id
    JOIN
    pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
GROUP BY pizza_types.category
ORDER BY revenue DESC;
```

	category	revenue
▶	Classic	26.91
	Supreme	25.46
	Chicken	23.96
	Veggie	23.68

# ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME

```
SELECT order_date,  
sum(revenue) OVER(ORDER BY order_date) AS Cumulative_revenue  
FROM  
(SELECT orders.order_date,  
ROUND(SUM(order_details.quantity * pizzas.price),2) AS Revenue  
FROM order_details  
JOIN pizzas  
ON order_details.pizza_id = pizzas.pizza_id  
JOIN orders  
ON order_details.order_id=orders.order_id  
GROUP BY order_date) AS sales;
```

	order_date	Cumulative_revenue
▶	2015-01-01	2713.85
	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



# DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPE BASED ON REVENUE FOR EACH PIZZA CATEGORY

```
select name,category, 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,
ROUND(SUM(order_details.quantity * pizzas.price),2) AS Revenue
from order_details
join pizzas
on order_details.pizza_id = pizzas.pizza_id
join pizza_types
on pizzas.pizza_type_id=pizza_types.pizza_type_id
group by category,pizza_types.name
order by revenue desc) as A) as b
where rn<4;
```

	name	category	revenue
▶	The Thai Chicken Pizza	Chicken	43434.25
	The Barbecue Chicken Pizza	Chicken	42768
	The California Chicken Pizza	Chicken	41409.5
	The Classic Deluxe Pizza	Classic	38180.5
	The Hawaiian Pizza	Classic	32273.25
	The Pepperoni Pizza	Classic	30161.75
	The Spicy Italian Pizza	Supreme	34831.25
	The Italian Supreme Pizza	Supreme	33476.75



**THANK YOU**

20 JUNE, 2024