



Ingoude Company

# PIZZA SALES REPORT

02 May, 2024

# INTRODUCTION

Welcome to our Sales Report Presentation. Today, we delve into a comprehensive overview of sales of the pizza store exploring the . This project represent more than just numbers, it's a narrative the different scenarios about the best and the most pizzas sold. This will help in focusing more on least profitable pizzas and maintaining and improving the most profitable pizzas.



# AGENDA

- 01 Retrieve the total number of orders placed.
- 02 Calculate the 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 Join the necessary tables to find the total quantity of each pizza category ordered.
- 07 Determine the distribution of orders by hour of the day.
- 08 Join relevant tables to find the category-wise distribution of pizzas..
- 09 Group the orders by date and calculate the average number of pizzas ordered per day.
- 10 Determine the top 3 most ordered pizza types based on revenue.
- 11 Calculate the percentage contribution of each pizza type to total revenue.
- 12 Analyze the cumulative revenue generated over time.
- 13 Determine the top 3 most ordered pizza types based on revenue for each pizza category.

Retrieve the total number of orders placed.

**SELECT**

COUNT(order\_id) AS 'total orders places'

**FROM**

orders;

**output**

Result Grid	
	total orders places
▶	21350

Calculate the total revenue generated from pizza sales.

**SELECT**

```
    ROUND(SUM(orders_details.quantity * pizzas.price),  
          2) AS 'total revenue'
```

**FROM**

```
orders_details
```

**JOIN**

```
pizzas ON pizzas.pizza_id = orders_details.pizza_id;
```

**output**

Result Grid	
	total revenue
▶	817860.05

# Identify the highest-priced pizza.

```
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;
```

output

Result Grid | Filter Row

	name	price
▶	The Greek Pizza	35.95

Identify the most common pizza size ordered.

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

output

Result Grid

	size	total_orders
▶	L	18526

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

```
SELECT
    pizza_types.name, SUM(orders_details.quantity) AS quantity
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 quantity DESC
LIMIT 5;
```

output

Result Grid		Filter Rows:
	name	quantity
▶	The Classic Deli	Refresh data re-execute
	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.

```
select pizza_types.category,  
sum(orders_details.quantity) as quantity  
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 quantity desc;
```

output

Result Grid | Filter

	category	quantity
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050

Determine the distribution of orders by hour of the day.

```
SELECT  
    HOUR(order_time), COUNT(order_id) AS 'order count'  
FROM  
    orders  
GROUP BY HOUR(order_time);
```

output

Result Grid	
HOUR(order_time)	order count
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.

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

output

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

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

```
SELECT  
    ROUND(AVG(quantity), 2) AS 'average orders per day'  
FROM  
    (SELECT  
        orders.order_date, SUM(orders_details.quantity) AS quantity  
    FROM  
        orders  
    JOIN orders_details ON orders.order_id = orders_details.order_id  
    GROUP BY orders.order_date) AS order_quantity;
```

## output

Result Grid	
	average orders per day
▶	138.47

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

```
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 orders_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY revenue DESC
LIMIT 3;
```

## output

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
        ROUND(SUM(orders_details.quantity * pizzas.price),
        2) AS total_sale
    )
FROM
    orders_details
    JOIN
        pizzas ON pizzas.pizza_id = orders_details.pizza_id) * 100,2) AS revenue
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 category
ORDER BY revenue DESC;
```

## output

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

output

result Grid	
order_date	cum_revenue
2015-01-01	2713.8500000000004
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	23990.350000000002
2015-01-11	25862.65

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

```
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((orders_details.quantity)*pizzas.PRICE) AS REVENUE
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,pizza_types.name) as A) as B
where rn<=3;
```

## output

Result Grid		
	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

# THANK YOU

02 May, 2024