

Where Every Slice is a Taste of Perfection

WELCOME TO PIZZA RESTO

ORDER
NOW

Start Your Slide



HELLO



My name is Mansur Ahamad. I have done MCA from Asian International University and BSc(Hons) Mathematics from Aligarh Muslim University. In this project i have utilized sql query to solve questions related to Pizza Sales.

Pizza Sales Questions

- 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 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  
FROM  
    orders
```

Result Grid	
	Total_Orders
▶	21350

Calculate the total revenue generated from pizza sales.

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

Result Grid |

	Toatal_Revenue
▶	817860.05

Identify the highest-priced pizza.

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

Result Grid | Filter Rows:

	price	name
→	35.95	The Greek Pizza

In 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 pizzas.pizza_type_id = pizza_types.pizza_type_id  
        JOIN  
    order_details ON order_details.pizza_id = pizzas.pizza_id  
GROUP BY category  
ORDER BY quantity DESC
```

Result Grid | Filter R

	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) 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

Result 48 ×

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

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

Result Grid

Avg(quantity)
138.4749

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

```
SELECT pizza_types.category,  
       ROUND(  
           (SUM(order_details.quantity * pizzas.price) /  
            (SELECT ROUND(SUM(order_details.quantity * pizzas.price), 2)  
             FROM order_details  
             JOIN pizzas ON pizzas.pizza_id = order_details.pizza_id) * 100), 2  
        ) AS revenue_percentage  
  FROM pizza_types  
  JOIN pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id  
  JOIN order_details ON order_details.pizza_id = pizzas.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

Result 50 ×

Analyze the cumulative revenue generated over time.

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

Result Grid | Filter Rows:

	order_date	Cum_rev
▶	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

Result 52 ×

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

```
SELECT
    pizza_types.category,
    pizza_types.name,
    SUM(order_details.quantity * pizzas.price) AS revenu,
    RANK() OVER (PARTITION BY pizza_types.category ORDER BY SUM(order_details.quantity * pizzas.price) DESC) AS rn
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, pizza_types.name
```

Result Grid | Filter Rows: Export:

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

THANKS