

Where Every Slice is a Taste of Perfection

SQL PROJECT ON PIZZA SALES

Start Your Slide

ORDER
NOW



ABOUT OUR PIZZA SALES PROJECT



This project analyzes pizza sales data using SQL to answer key business questions from a restaurant owner's perspective.

By applying various SQL queries, we uncover insights on top-selling pizzas, peak order times, customer preferences, and revenue trends. These findings help the owner make data-driven decisions to improve sales, optimize inventory, and shape effective business strategies.





ABOUT MYSELF

I am **Vikash Prasad**, a **2024 MSc Mathematics** graduate from **NIT Trichy**, with a **strong analytical and problem-solving background**. I am seeking career opportunities as a **Data Analyst**, **Business Analyst**, or in other analytical roles where I can apply my skills in data interpretation, statistical analysis, and business insights to drive impactful decisions.



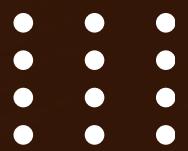
REQUIREMENTS OF PIZZA RESTRO OWNER



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.

1. Join the necessary tables to find the total quantity of each pizza category ordered.
2. Determine the distribution of orders by hour of the day.
3. Join relevant tables to find the category-wise distribution of pizzas.
4. Group the orders by date and calculate the average number of pizzas ordered per day.
5. Determine the top 3 most ordered pizza types based on revenue.

1. Calculate the percentage contribution of each pizza type to total revenue.
2. Analyze the cumulative revenue generated over time.
3. 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(orders_details.quantity * pizzas.price),  
2) AS total_sales  
FROM  
orders_details  
JOIN  
pizzas ON orders_details.pizza_id = pizzas.pizza_id;
```

Result Grid	
	total_sales
▶	817860.05

Identify the highest-priced pizza.

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

Result Grid		Filter Rows
	name	price
▶	The Greek Pizza	35.95

Identify the most common pizza size ordered.

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

Result Grid		Filter Rows
	size	COUNT(orders_details.order_details_id)
▶	L	18526

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

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

	name	SUM(orders_details.quantity)
▶	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.

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

	category	total_quantity
▶	Classic	14888
	Veggie	11649
	Supreme	11987
	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);
```

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.

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

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)  
FROM  
(SELECT  
    orders.Order_date,  
    COUNT(orders_details.order_details_id) AS  
        orders,  
    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;
```

ROUND(AVG(quantity), 2)
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 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 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 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 orders_details.pizza_id = pizzas.pizza_id)) * 100,
    2) AS total_revenue
    FROM
        orders_details
    JOIN
        pizzas ON orders_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 total_revenue DESC;
```

category	total_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_rev 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;
```

order_date	cum_rev
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
2015-01-12	27781.7
2015-01-13	29831.300000000003
2015-01-14	32258.700000000004

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

```
select category, name, revenue, rn 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.name, pizza_types.category) as a)
as b where rn <=3;
```

category	name	revenue	m
Chicken	The Thai Chicken Pizza	43434.25	1
Chicken	The Barbecue Chicken Pizza	42768	2
Chicken	The California Chicken Pizza	41409.5	3
Classic	The Classic Deluxe Pizza	38180.5	1
Classic	The Hawaiian Pizza	32273.25	2
Classic	The Pepperoni Pizza	30161.75	3
Supreme	The Spicy Italian Pizza	34831.25	1
Supreme	The Italian Supreme Pizza	33476.75	2
Supreme	The Sicilian Pizza	30940.5	3
Veggie	The Four Cheese Pizza	32265.70000000065	1
Veggie	The Mexicana Pizza	26780.75	2
Veggie	The Five Cheese Pizza	26066.5	3

Pizza Sales Presentation

THANK YOU
FOR ATTENTION

vikash.nitt01maths@gmail.com

[https://www.linkedin.com/in/vikash
-prasad26/](https://www.linkedin.com/in/vikash-prasad26/)