

# SQL PROJECT



## PIZZA SALES



*In this project , I examined the dataset with SQL and help the store understand its business growth by answering simple questions.*



# 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.



# QUESTIONS

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.



# QUESTIONS

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.



*Now , coming to solutions part  
of each questions one by one.*





1. Retrieve the total number of orders placed.

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

	total_orders
▶	21350





## 2. Calculate the total revenue generated from pizza sales.

```
select round(sum(order_details.quantity * pizzas.price),2) as revenue  
from order_details join pizzas  
on pizzas.pizza_id = order_details.pizza_id;
```



	Result Grid	grid icon
revenue		
▶	817860.05	





### 3. 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;
```

Result Grid | Filter Rows:

	name	price
▶	The Greek Pizza	35.95





4. Identify the most common pizza size ordered..

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;



Result Grid | Filter

	size	order_count
▶	L	18526
	M	15385
	S	14137
	XL	544
	XXL	28



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

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
1	The Classic Deluxe Pizza	2453
2	The Barbecue Chicken Pizza	2432
3	The Hawaiian Pizza	2422
4	The Pepperoni Pizza	2418
5	The Thai Chicken Pizza	2371



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 order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
ORDER BY quantity DESC;
```



	category	quantity
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050



# 7. Determine the distribution of orders by hour of the day.

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



	hour	order_count
	11	1231
	12	2520
	13	2455
	14	1472
	15	1468
	16	1920
	17	2336
	18	2399
	19	2009



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

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



Result Grid | Filter R

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



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

```
SELECT  
    ROUND(AVG(quantity), 0) as avg_pizza_orderd_per_day  
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_quantity;
```



avg_pizza_orderd_per_day
138



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

```
select pizza_types.name,  
sum(order_details.quantity * pizzas.price) as revenue  
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.name order by revenue desc limit 3;
```



Result Grid		Filter Rows:
	name	revenue
1	The Thai Chicken Pizza	43434.25
2	The Barbecue Chicken Pizza	42768
3	The California Chicken Pizza	41409.5



# THANK YOU

