



# SQL PROJECT

Presenting the Pizza Sales Analysis
Report, which I've generated using SQL.
Goal with this report is to gain insights
into the sales performance across various
types of pizzas, customer preferences,
and trends over time. By analyzing the
data, I was able to identify key patterns
and opportunities for optimizing the sales
strategy. Let's dive into the findings and
see how they can help us drive better
business decisions."





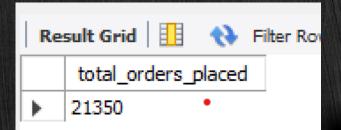
Retrieve the total number of orders placed.

## SELECT

COUNT(order\_id) AS total\_orders\_placed

#### **FROM**

orders;







**Calculate the total** revenue generated from pizza sales.

## SELECT ROUND(SUM(o.quantity \* p.price), 2) AS total\_sales FROM orders\_details o JOIN pizzas p ON o.pizza\_id = p.pizza\_id;





total\_sales

817860.05



## Identify the highestpriced pizza

Result Grid

name

The Greek Pizza

Filter Rows

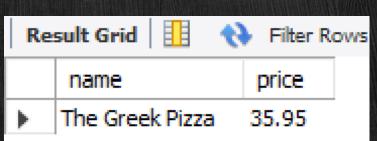
price

35.95





## Identify the highestpriced pizza





Identify the most common pizza size ordered.

```
SELECT
    size, COUNT(size) AS total_order_count
FROM
    orders_details
        JOIN
    pizzas ON orders_details.pizza_id = pizzas.pizza_id
GROUP BY size
ORDER BY size;
```

R	esult Gri	i <b>d</b> 🔢 🙌 Filter Ro
	size	total_order_count
•	L	18526
	M	15385
	S	14137
	XL	544
	VVI	28



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

```
SELECT
    pizza_types.name, SUM(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;
```





Join the necessary tables to find the total quantity of each pizza category ordered

```
SELECT
category, SUM(quantity) AS total_quantity

FROM

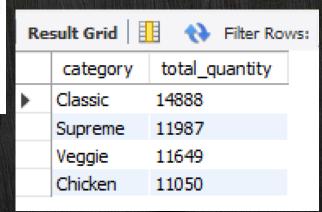
pizzas
JOIN

pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
JOIN

orders_details ON orders_details.pizza_id = pizzas.pizza_id

GROUP BY category

ORDER BY total_quantity DESC;
```





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
Þ	11	123 Resets all
	12	252
	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;

Re	sult Grid	Filter Rows
	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(qty), 0) AS average_quantity_per_day

FROM

(SELECT

order_date, SUM(quantity) AS qty

FROM

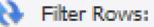
orders

JOIN orders_details ON orders.order_id = orders_details.order_id

GROUP BY order_date) AS order_quantity;
```







average\_quantity\_per\_day



138



LIMIT 3;

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

```
SELECT

name,

SUM(orders_details.quantity * pizzas.price) AS total_sales

FROM

pizzas

JOIN

pizza_types ON pizza_types.pizza_type_id = pizzas.pizza_type_id

JOIN

orders_details ON orders_details.pizza_id = pizzas.pizza_id

GROUP BY name

ORDER BY total_sales DESC
```



	name	total_sales
•	The Thai Chicken Pizza	43434.25
	The Barbecue Chicken Pizza	42768
	The California Chicken Pizza	41409.5



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

```
SELECT

name,

SUM(orders_details.quantity * pizzas.price) AS total_sales

FROM

pizzas

JOIN

pizza_types ON pizza_types.pizza_type_id = pizzas.pizza_type_id

JOIN

orders_details ON orders_details.pizza_id = pizzas.pizza_id

GROUP BY name

ORDER BY total_sales DESC

LIMIT 3;

Result Grid 

Filter Rows:
```

	name	total_sales	
<b>•</b>	The Thai Chicken Pizza	43434.25	
	The Barbeque Chicken Pizza	42768	

41409.5

The California Chicken Pizza



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

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

Re	sult Grid	H 🙌 Filt
	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 order\_date, sum(orders\_details.quantity \* pizzas.price) as revenue from
 orders join
 orders\_details
 on orders.order\_id = orders\_details.order\_id
 join
 pizzas
 on
 pizzas.pizza\_id= orders\_details.pizza\_id
 group by order\_date) as sales;

		NAME OF TAXABLE PARTY OF TAXABLE PARTY.
Re	esult Grid	N Filter Rows:
	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
	2015-01-12	27781.7
	2015-01-13	29831.300000000003
	2015-01-14	32358.700000000004
	2015-01-15	34343.50000000001
	2015-01-16	36937.65000000001
	2015-01-17	39001.75000000001
	2015-01-18	40978.6000000000006
	2015-01-19	43365.75000000001
	2015-01-20	45763.65000000001
	2015-01-21	47804.20000000001
	2015 21 22	F0000 0000000000





where rn <=3;

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

group by category, name order by total\_sales) as a) as b

	name	total_sales
٠	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