

Delicious Pizza for Everyone!

# PIZZASTORE SALES ANALYSIS

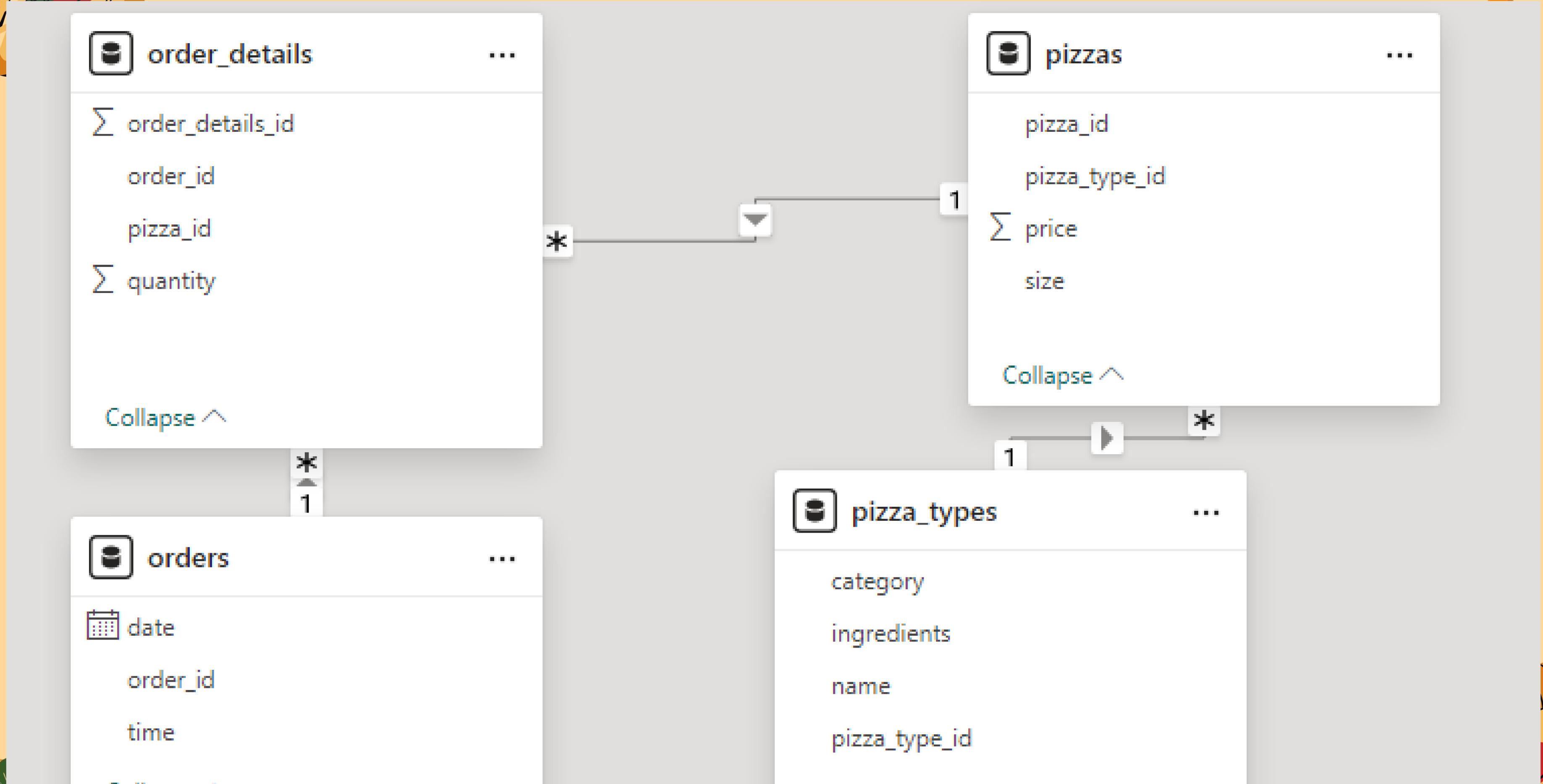


# Hello!



My name is Firoz, and i prepared pizza sales analysis project examine into understanding customer behavior and optimizing business strategies. By analyzing order data, uncovered insights such as the total number of orders placed and the total revenue generated from pizza sales. Also identified the highest-priced pizza and the most common pizza size ordered. Additionally, highlighted the top 5 most ordered pizza types along with their quantities to inform menu optimization. Through advanced analysis, examine deeper into category-wise pizza distribution, hourly order distribution, and revenue contribution by pizza type. This project equips stakeholders with actionable insights to enhance customer experience and drive business growth

In this project i have used SQL queries to solve questions related to pizza sales. and question and their answers are mention from next page.



# Retrieve the total number of orders placed.

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

	Total_orders
▶	21350

# Calculate the total revenue generated from pizza sales:

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

	Total_revenue
▶	817860.05

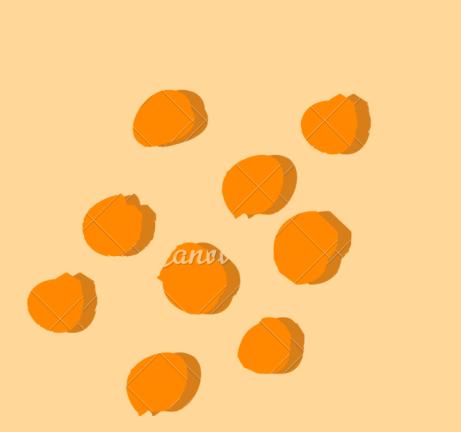
# Identify the highest-priced pizza.

```
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;
```

	name	price
▶	The Greek Pizza	35.95



# 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;
```

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



# 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
▶	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(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



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



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



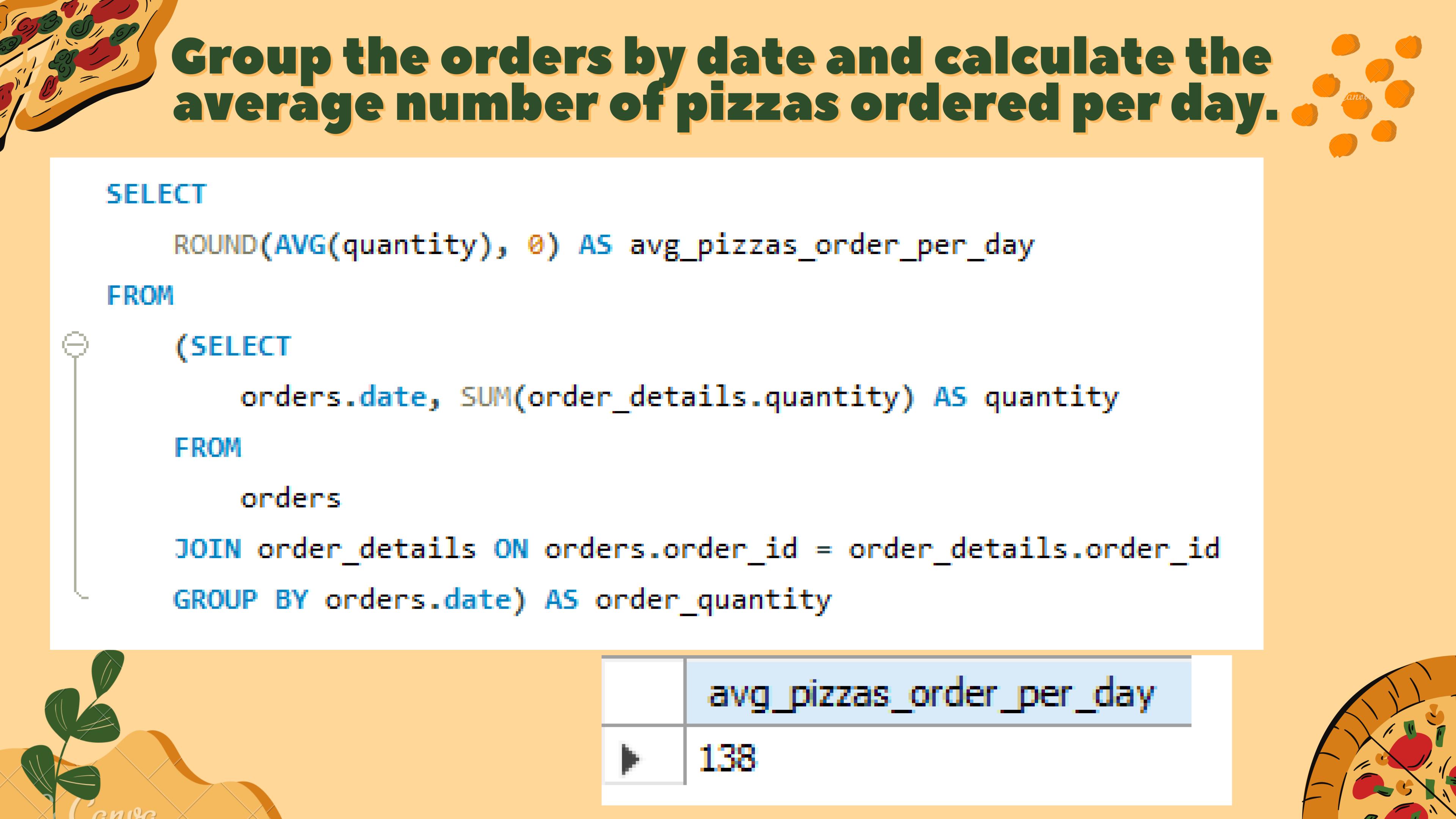
# Determine the distribution of orders by hour of the day.

	hour	order_count
▶	12	2520
	13	2455
	18	2399
	17	2336
	19	2009
	16	1920
	20	1642
	14	1472
	15	1468
	11	1231
	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), 0) AS avg_pizzas_order_per_day  
FROM  
    (SELECT  
        orders.date, SUM(order_details.quantity) AS quantity  
    FROM  
        orders  
    JOIN order_details ON orders.order_id = order_details.order_id  
    GROUP BY orders.date) AS order_quantity
```

	avg_pizzas_order_per_day
▶	138

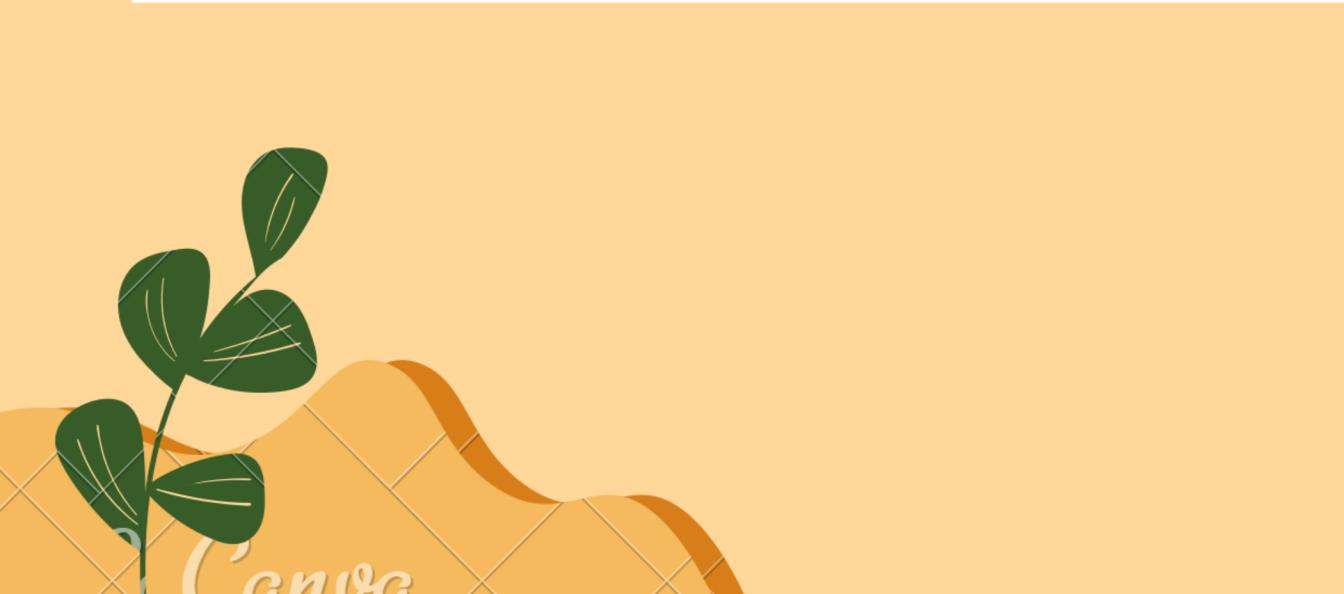


# Determine the top 3 most ordered pizza types based on revenue.



```
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;
```

	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,  
    concat(ROUND((SUM(order_details.quantity * pizzas.price) /  
        (SELECT ROUND(SUM(order_details.quantity * pizzas.price), 2) AS Total_revenue  
         FROM order_details  
         JOIN pizzas ON pizzas.pizza_id = order_details.pizza_id)  
        ) * 100, 2), '%') AS revenue
```

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

```
    revenue DESC;
```

	category	revenue
▶	Classic	26.91%
	Supreme	25.46%
	Chicken	23.96%
	Veggie	23.68%

# Analyze the cumulative revenue generated over time.

```
select date,  
       sum(revenue) over(order by date) as cum_revenue  
  from  
(select orders.date,  
           sum(order_details.quantity* pizzas.price) as revenue  
      from order_details join pizzas on order_details.pizza_id= pizzas.pizza_id  
      join orders  
        on orders.order_id = order_details.order_id  
   group by orders.date) as sales;
```

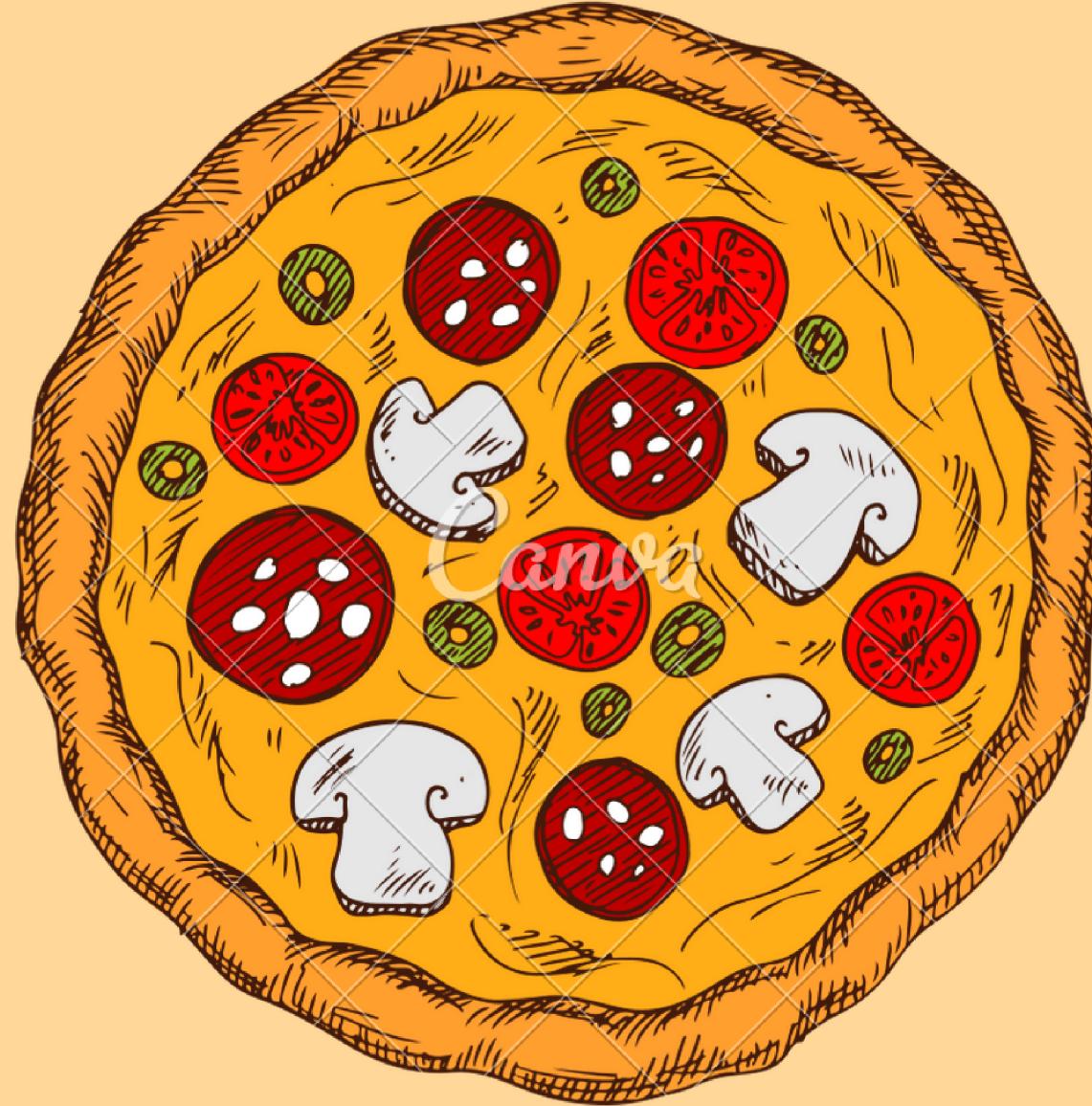
	date	cum_revenue
▶	2015-01-01	2713.8500000000004
	2015-01-02	5445.75
	2015-01-03	8108.15
	2015-01-04	9863.6
	2015-01-05	11929.55
	2015-01-06	14358.5
	2015-01-07	16560.7

# 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((order_details.quantity)* pizzas.price) as revenue  
      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) as a) as b  
  where rn <=3 ;
```

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

	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	32265.7000000065
	The Mexicana Pizza	26780.75
	The Five Cheese Pizza	26066.5



Pizzastore Sales Analysis

**THANK  
YOU**