

Delicious Pizza for Everyone!

SQL PROJECT

PIZZA SALES

indranilsarkar098@gmail.com

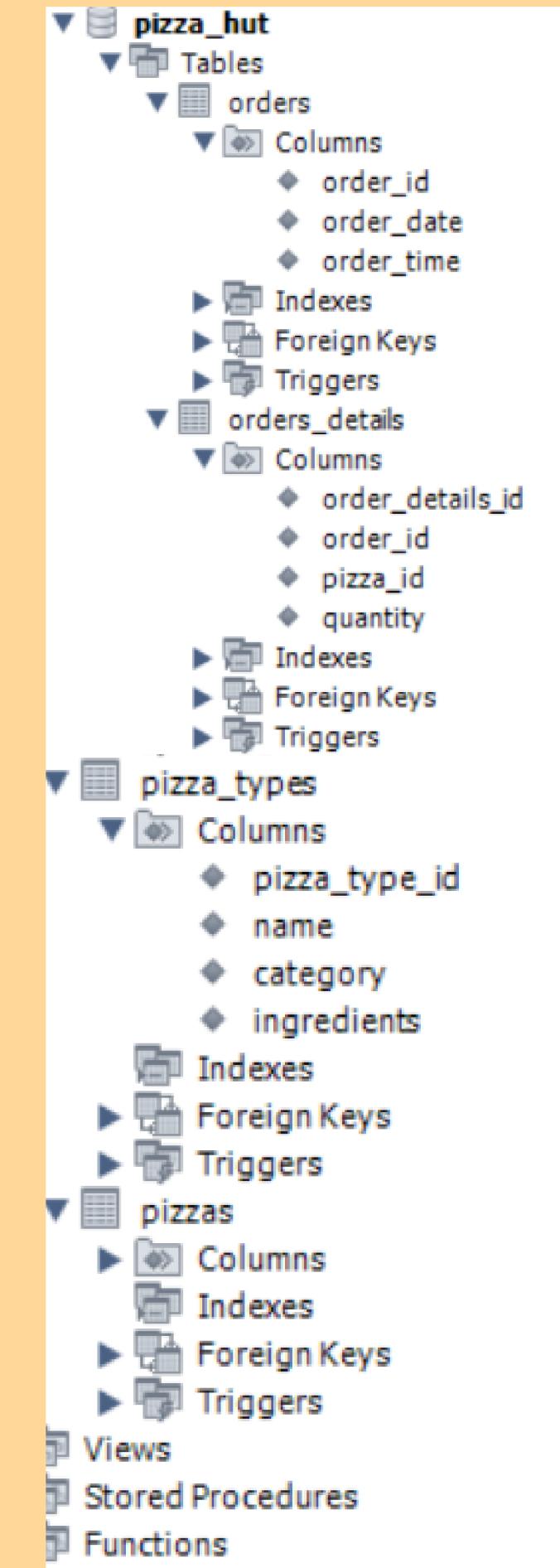


Hello!

I'm Indranil Sarkar and in this project I have utilized SQL queries to solve questions that were related to pizza sales.

About

SQL is a powerful tool for managing and analyzing large datasets in relational databases. By executing specific **SQL** queries, businesses can gain valuable insights into pizza sales, enabling data-driven decision-making to enhance profitability and customer satisfaction.



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(orders_details.quantity * pizzas.price),  
          2) AS Total_sales
```

```
FROM
```

```
    orders_details
```

```
JOIN
```

```
    pizzas
```

```
on pizzas.pizza_id = orders_details.pizza_id;
```

	Total_sales
▶	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(orders_details.order_details_id) AS Total_orders
```

FROM

```
pizzas
```

JOIN

```
orders_details ON orders_details.pizza_id = pizzas.pizza_id
```

GROUP BY pizzas.size

ORDER BY Total_orders **DESC;**

	size	Total_orders
▶	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(orders_details.quantity) AS quantity  
FROM  
    pizza_types  
        JOIN  
    pizzas ON pizzas.pizza_type_id = pizza_types.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;
```

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.

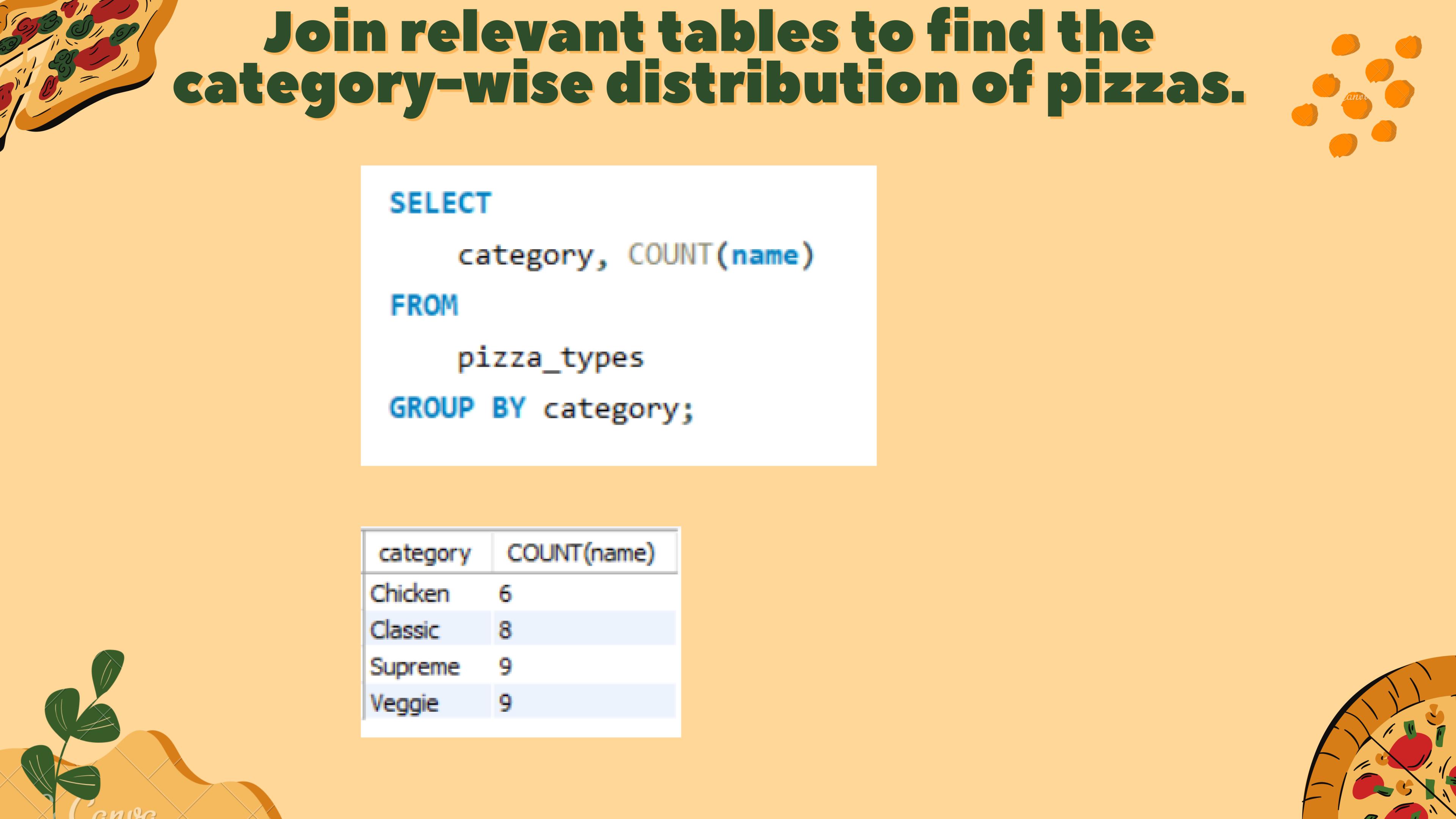
```
SELECT  
    pizza_types.category,  
    SUM(orders_details.quantity) AS quantity  
FROM  
    pizzas  
        JOIN  
    pizza_types ON pizza_types.pizza_type_id = pizzas.pizza_type_id  
        JOIN  
    orders_details ON orders_details.pizza_id = pizzas.pizza_id  
GROUP BY pizza_types.category  
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(order_time) AS Hours, COUNT(order_id) AS Order_counts  
FROM  
    orders  
GROUP BY HOUR(order_time);
```

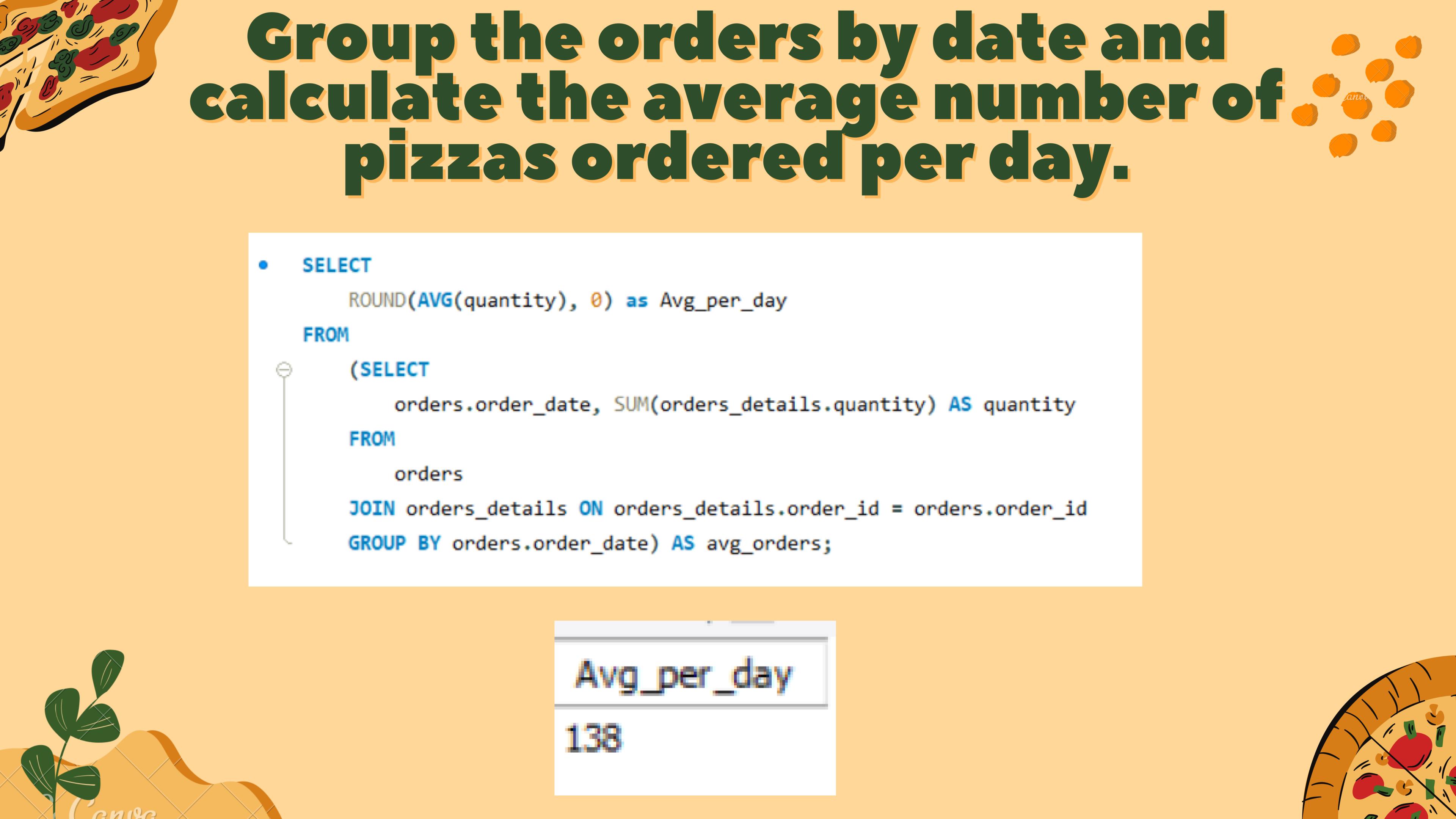
Hours	Order_counts
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642
21	1198
22	663
23	28
10	8
9	1



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), 0) as Avg_per_day
```

```
FROM
```

```
(SELECT
```

```
    orders.order_date, SUM(orders_details.quantity) AS quantity
```

```
FROM
```

```
    orders
```

```
JOIN orders_details ON orders_details.order_id = orders.order_id
```

```
GROUP BY orders.order_date) AS avg_orders;
```

Avg_per_day

138

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

- **SELECT**

```
    pizza_types.name,  
    SUM(orders_details.quantity * pizzas.price) AS revenue  
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 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  
        ROUND(SUM(orders_details.quantity * pizzas.price),  
        2) AS Total_sales  
    FROM  
        orders_details  
        JOIN  
        pizzas ON pizzas.pizza_id = orders_details.pizza_id)) * 100,  
    2) AS revenue  
  
FROM  
    orders_details  
    JOIN  
    pizzas ON pizzas.pizza_id = orders_details.pizza_id  
    JOIN  
    pizza_types ON pizza_types.pizza_type_id = pizzas.pizza_type_id  
GROUP BY pizza_types.category  
ORDER BY revenue;
```

category	revenue
Veggie	23.68
Chicken	23.96
Supreme	25.46
Classic	26.91

Analyze the cumulative revenue generated over time.

```
• select order_date, sum(revenue) over (order by order_date) as cumulative_revenue  
from (select orders.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 orders.order_date ) as sales ;
```

order_date	cumulative_revenue
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
2015-01-08	19399 19399.05
2015-01-09	21526.4
2015-01-10	23990.35000000002
2015-01-11	25862.65
2015-01-12	27781.7
2015-01-13	29831.30000000003
2015-01-14	32358.70000000004
2015-01-15	34343.50000000001
2015-01-16	36937.65000000001
2015-01-17	39001.75000000001
2015-01-18	40978.60000000006

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(orders_details.quantity * pizzas.price) AS revenue
  from pizza_types join pizzas
  on pizzas.pizza_type_id = pizza_types.pizza_type_id
  join orders_details
  on orders_details.pizza_id = pizzas.pizza_id
  group by pizza_types.category,pizza_types.name) as a) as b
  where rn <=3;
```

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	32265.70000000065
The Mexicana Pizza	26780.75
The Five Cheese Pizza	26066.5

Thank You