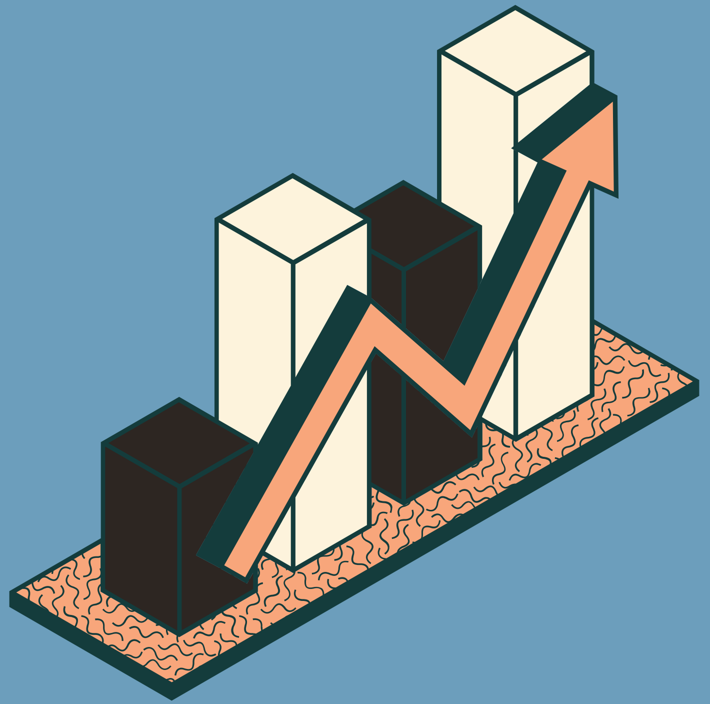
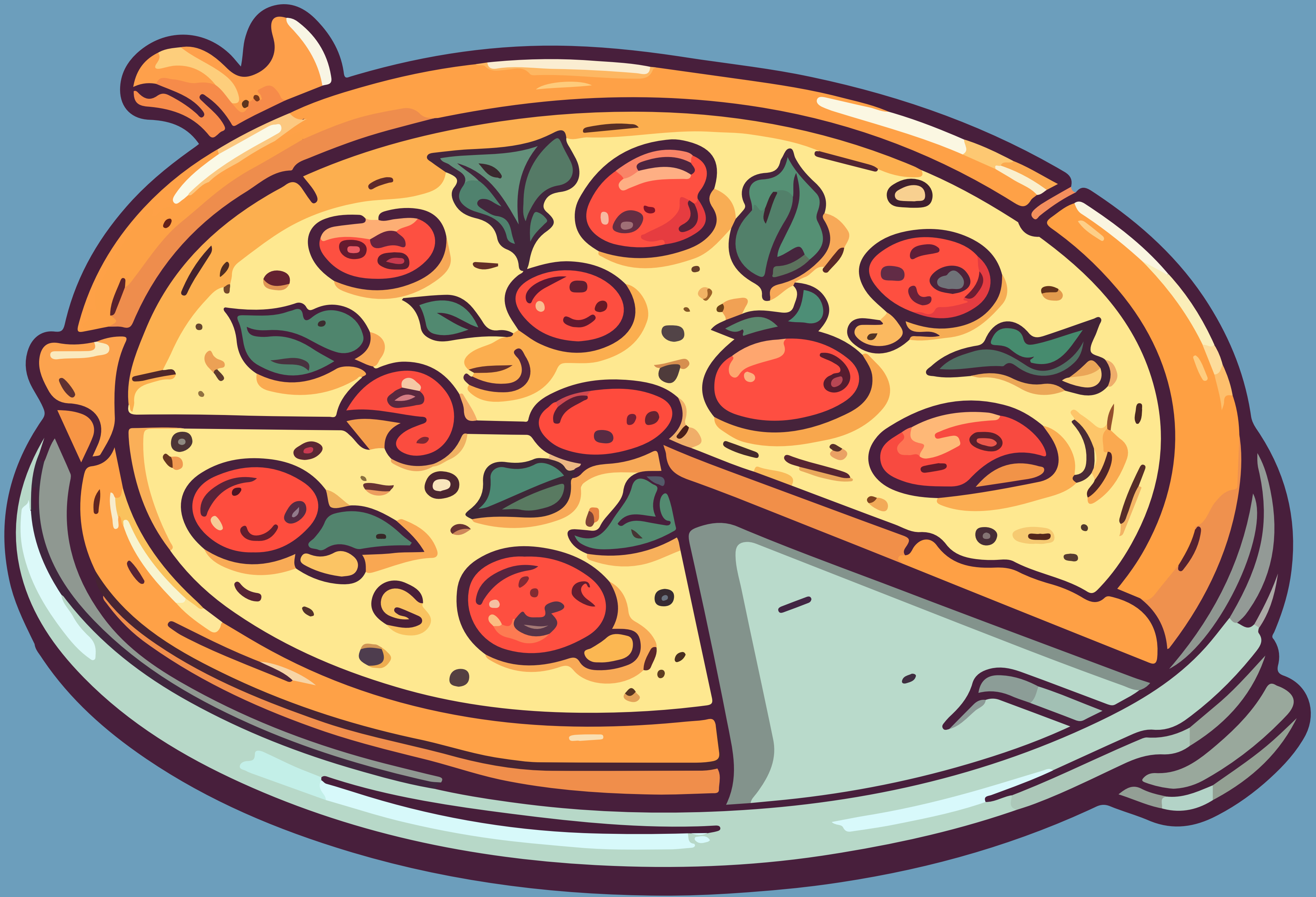


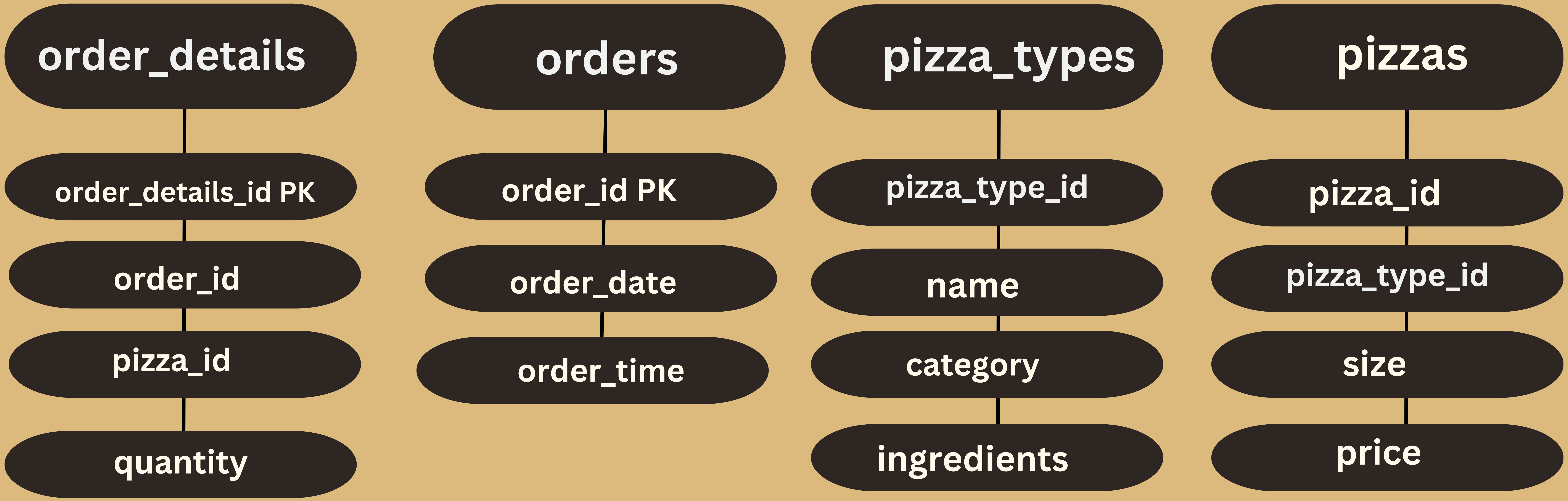
SQL PROJECT ON SALES - PIZZERIA



By-Sameeha Shaikh

SCHEMA DESIGN

This database contains four tables: ORDERS_DETAILS, ORDERS, PIZZA_TYPES, and PIZZAS.



order_details

order_details_id PK

order_id

pizza_id

quantity

orders

order_id PK

order_date

order_time

pizza_types

pizza_type_id

name

category

ingredients

pizzas

pizza_id

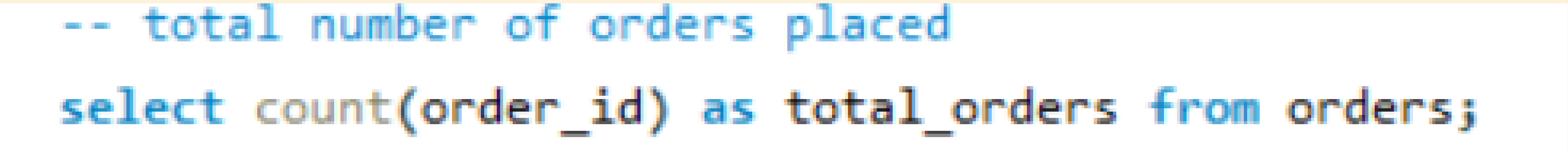
pizza_type_id

size

price

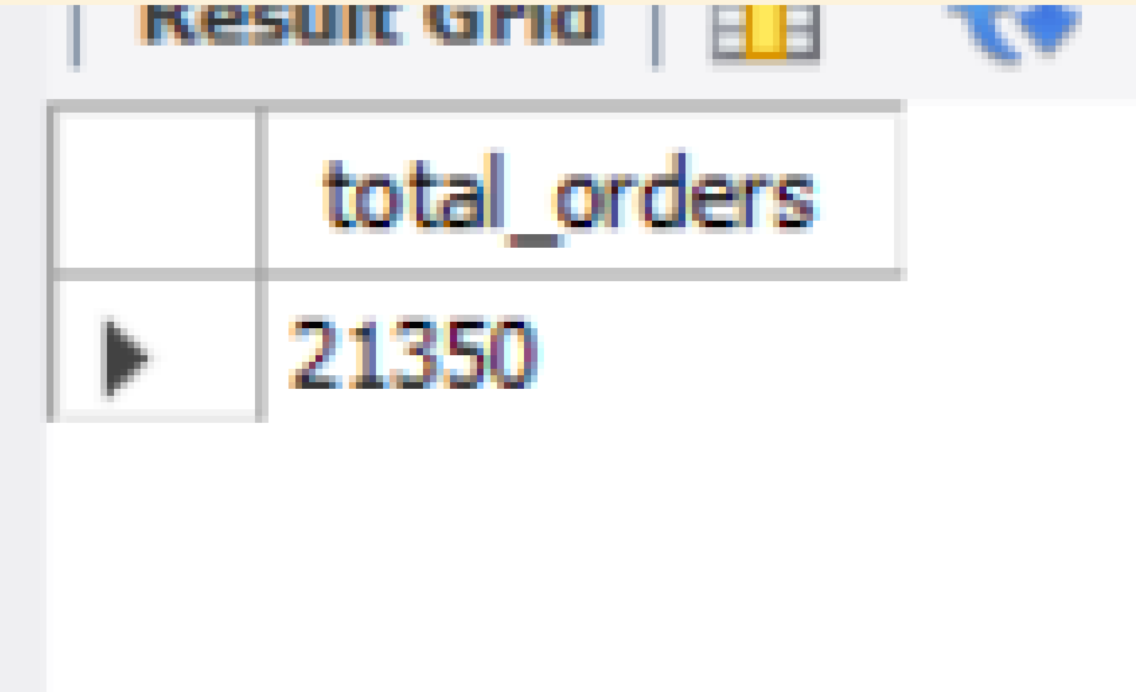
Retrieve the total number of orders placed.

Query:



```sql
-- total number of orders placed
select count(order_id) as total_orders from orders;
```

Output:

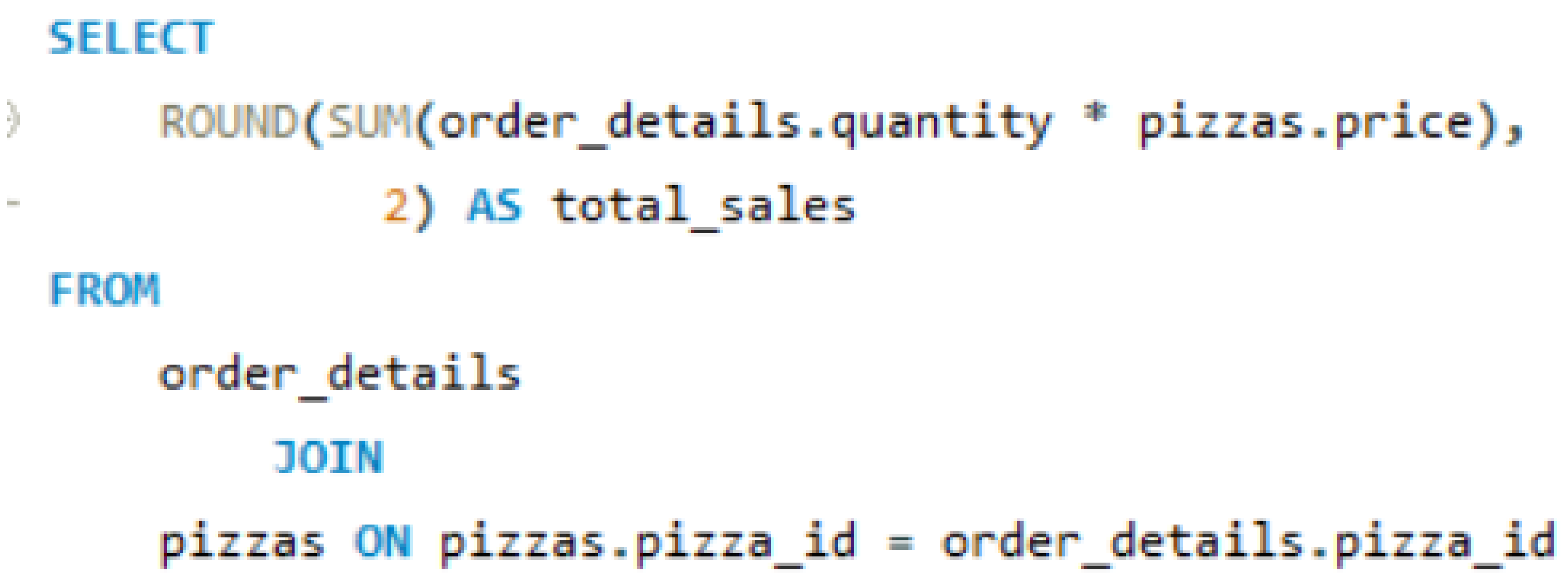


```
total_orders
21350
```

This query counts the total number of orders placed in the database. By using COUNT(order_id), it goes through the orders table and gives us the total number of unique orders recorded. The result is returned with a label, total_orders

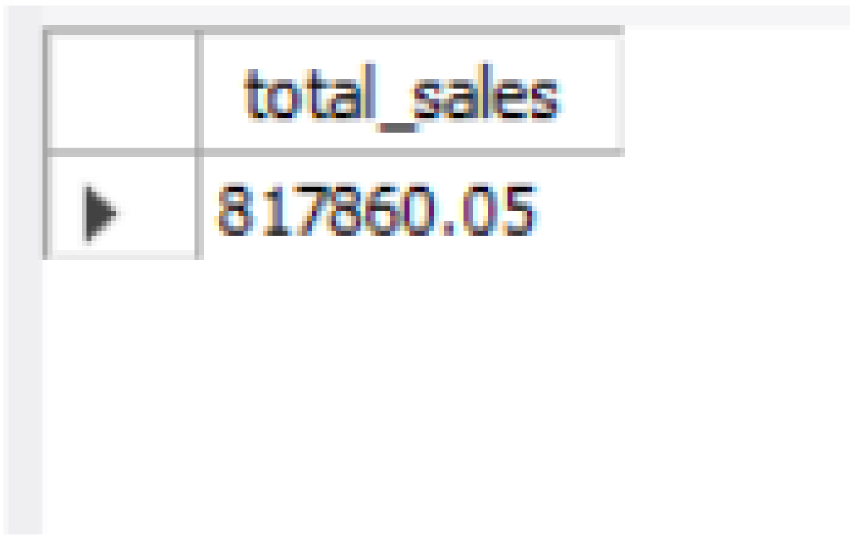
Calculate the total revenue generated from pizza sales.

Query:



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

Output:

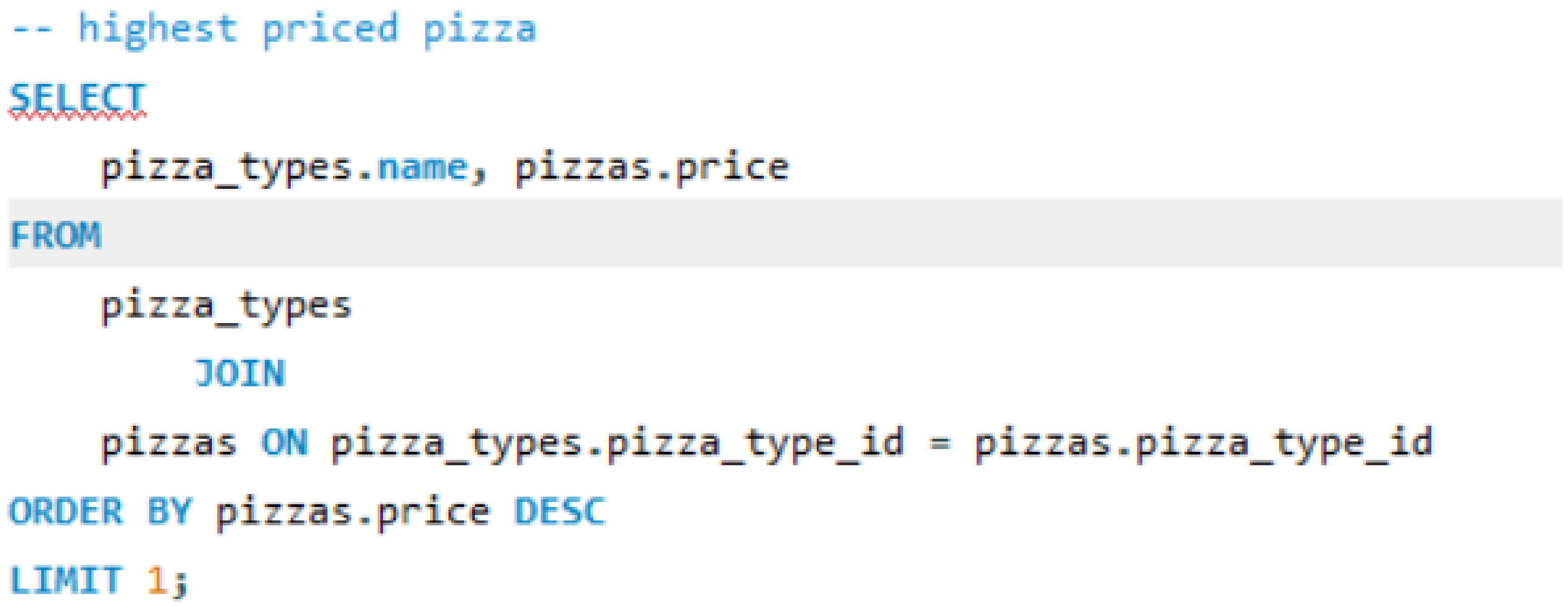


	total_sales
▶	817860.05

***This query calculates the total sales by multiplying the quantity of pizzas ordered with their price, then summing it all up. We round the result to two decimal places to get a clean number. The JOIN ensures we're matching the pizza details with their prices.**

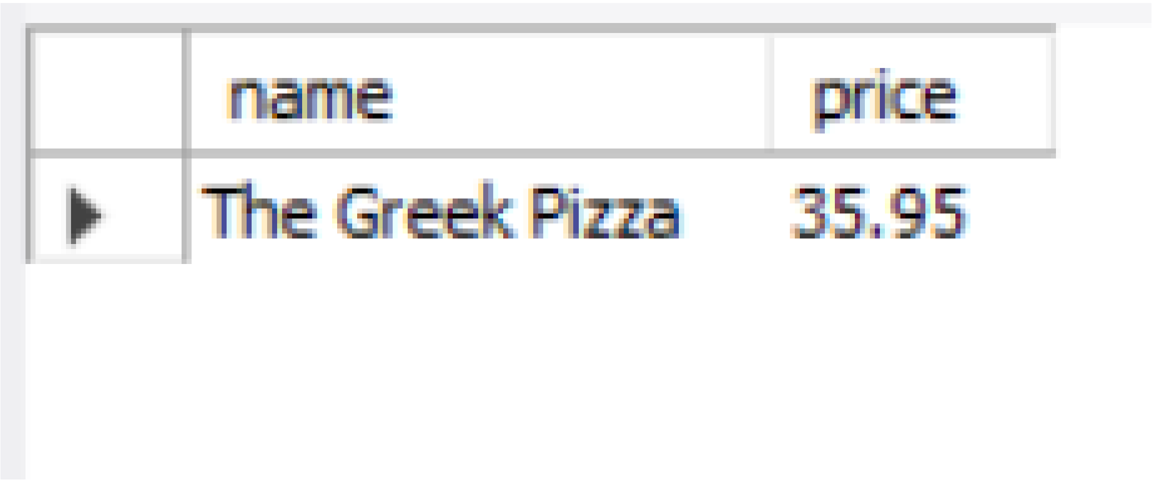
Identify the highest-priced pizza.

Query:



```
-- 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;
```

Output:

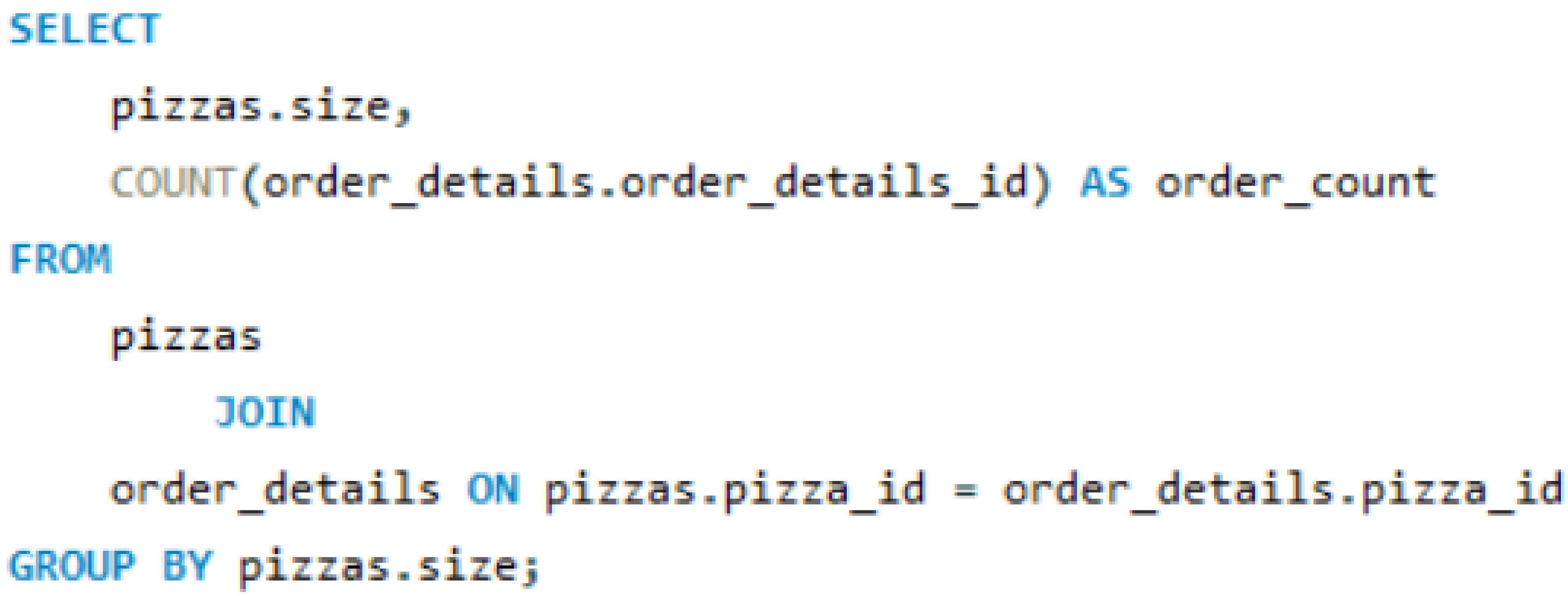


	name	price
▶	The Greek Pizza	35.95

***This query finds the most expensive pizza by joining the pizza_types and pizzas tables. It selects the pizza name and price, sorts them in descending order by price, and limits the result to just one, giving us the highest-priced pizza - The Greek Pizza**

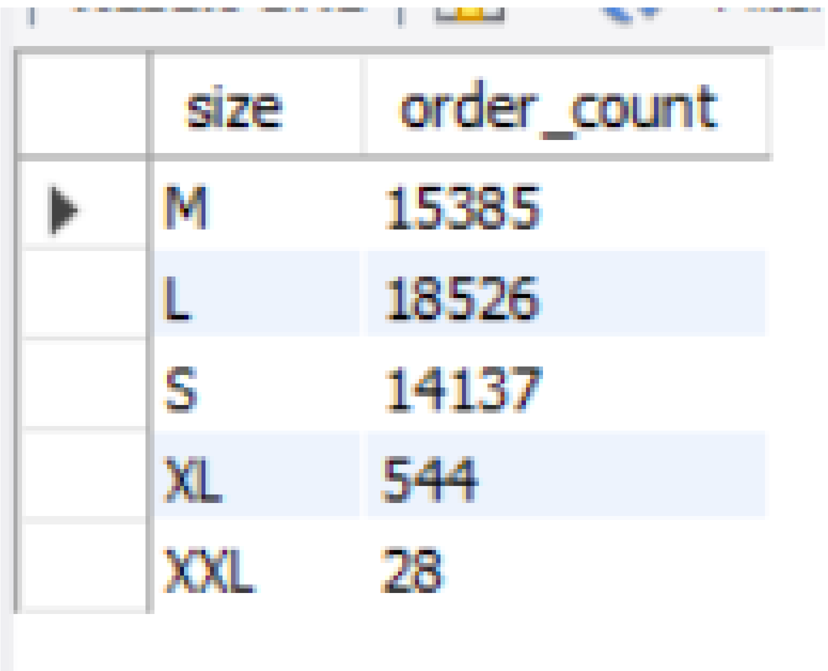
Identify the most common pizza size ordered.

Query:



```
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;
```

Output:

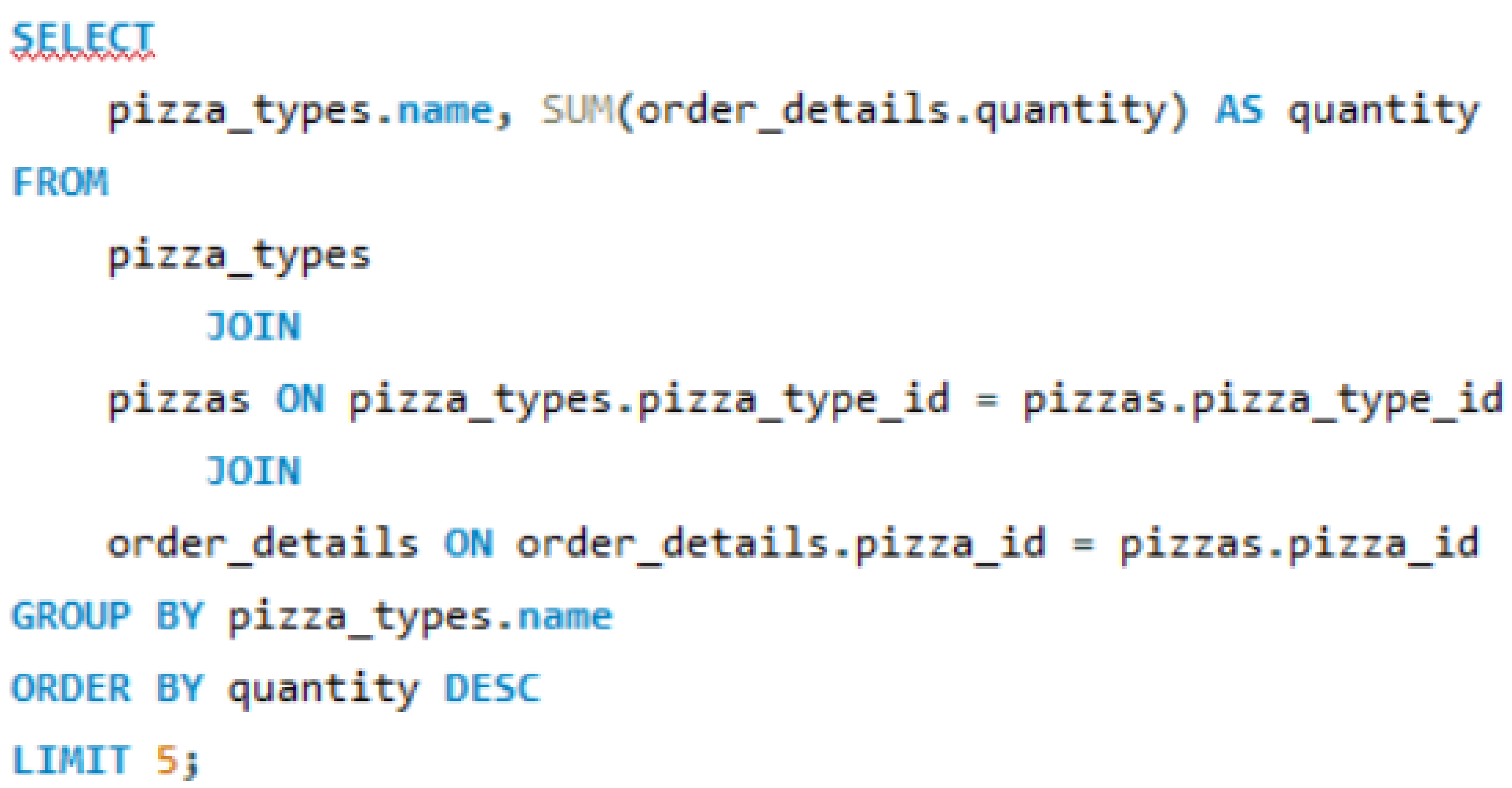


size	order_count
M	15385
L	18526
S	14137
XL	544
XXL	28

***This query counts the number of orders for each pizza size. It joins the pizzas and order_details tables and groups the results by size to show how many times each size has been ordered, with the output indicating that the medium-sized pizza has the highest order count, making it the most popular size among customers.**

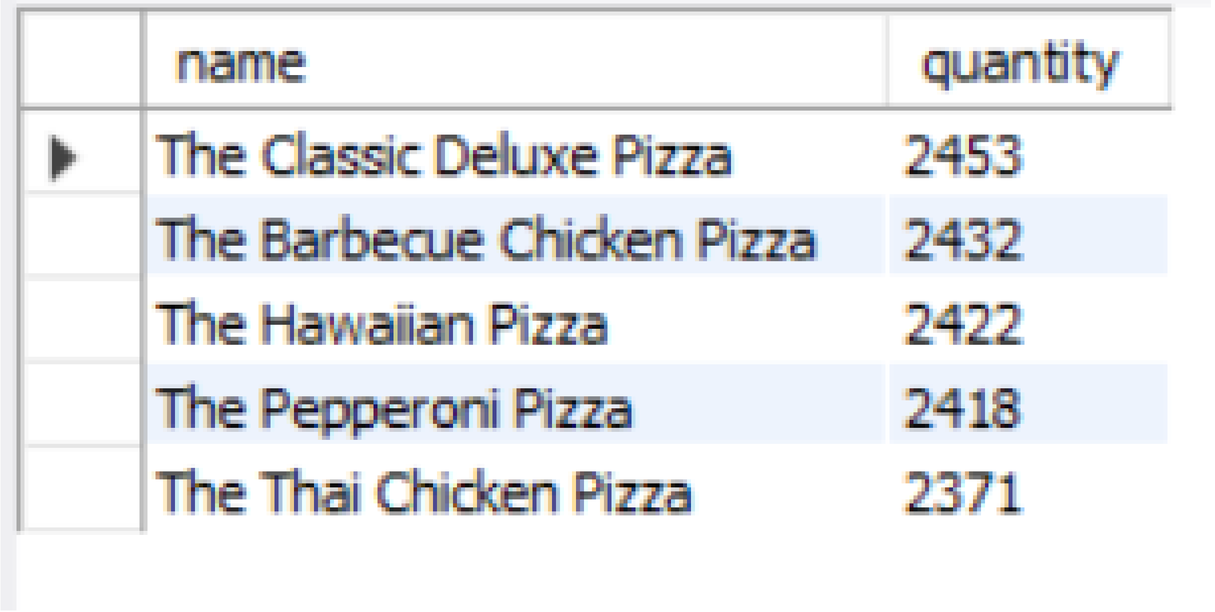
List the top 5 most ordered pizza types along with their quantities.

Query:



```
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;
```

Output:

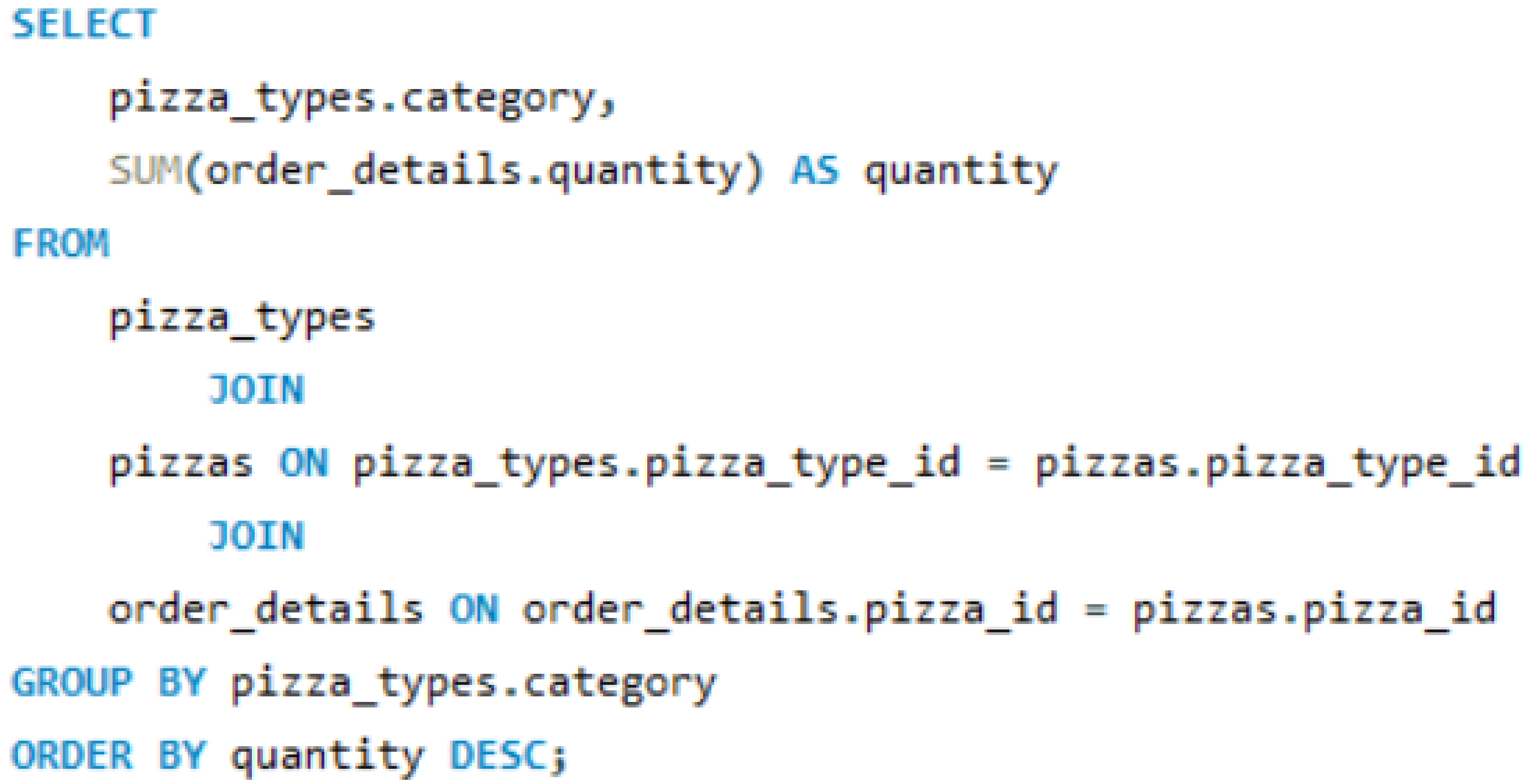


	name	quantity
▶	The Classic Deluxe Pizza	2453
	The Barbecue Chicken Pizza	2432
	The Hawaiian Pizza	2422
	The Pepperoni Pizza	2418
	The Thai Chicken Pizza	2371

***This query retrieves the top five pizza types based on total quantity sold. It joins the pizza_types, pizzas, and order_details tables to sum the quantities ordered for each type, grouping the results by pizza type name and sorting them in descending order.**

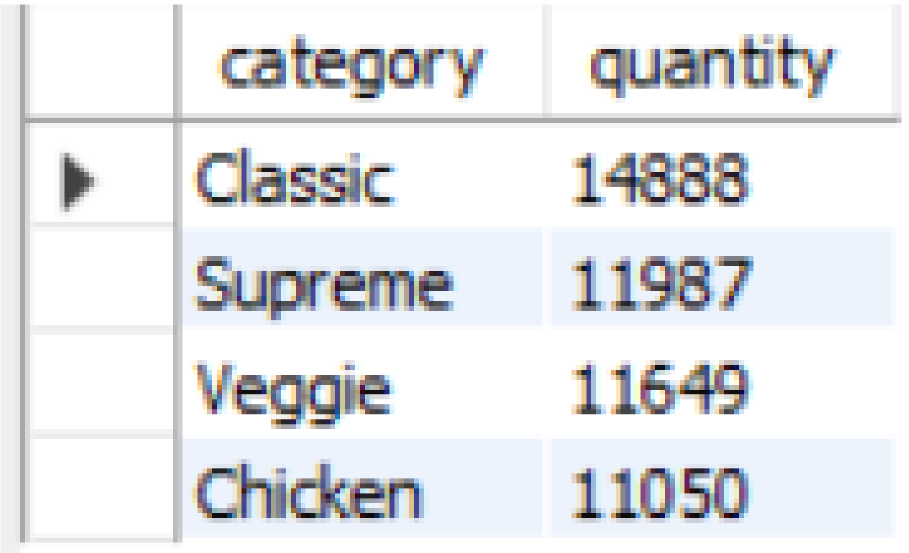
Join the necessary tables to find the total quantity of each pizza category ordered.

Query:



```
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;
```

Output:

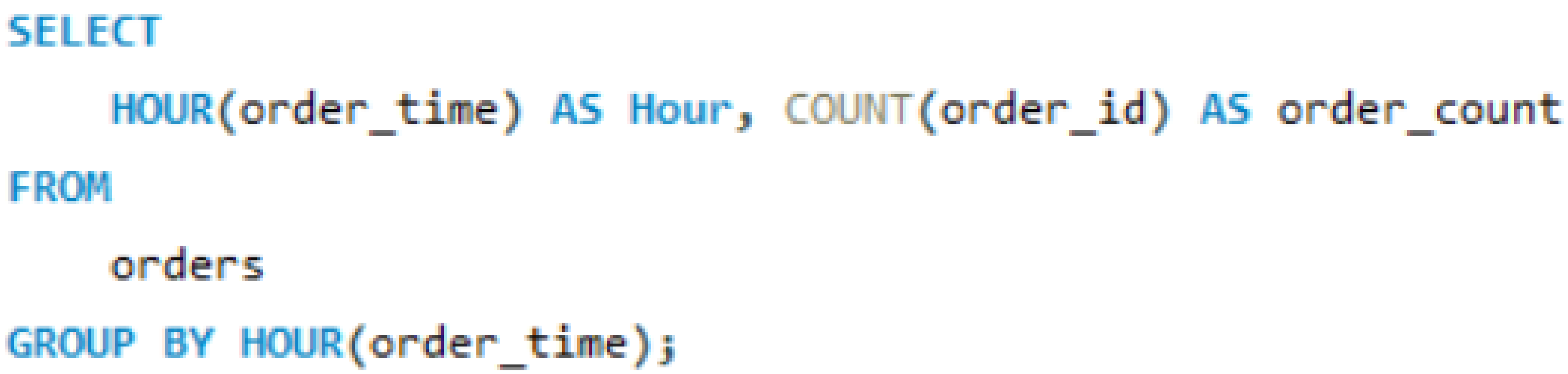


category	quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

This query totals the quantity sold for each pizza category. It joins the pizza_types, pizzas, and order_details tables, sums the quantities ordered, and groups the results by pizza category. The output is sorted in descending order, showing the most popular categories first, that is the CLASSIC

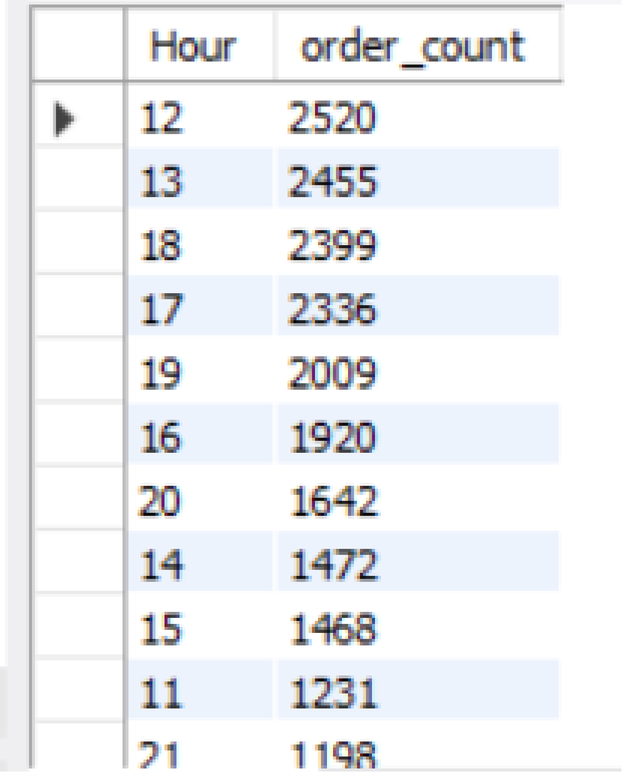
Determine the distribution of orders by hour of the day.

Query:



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

Output:

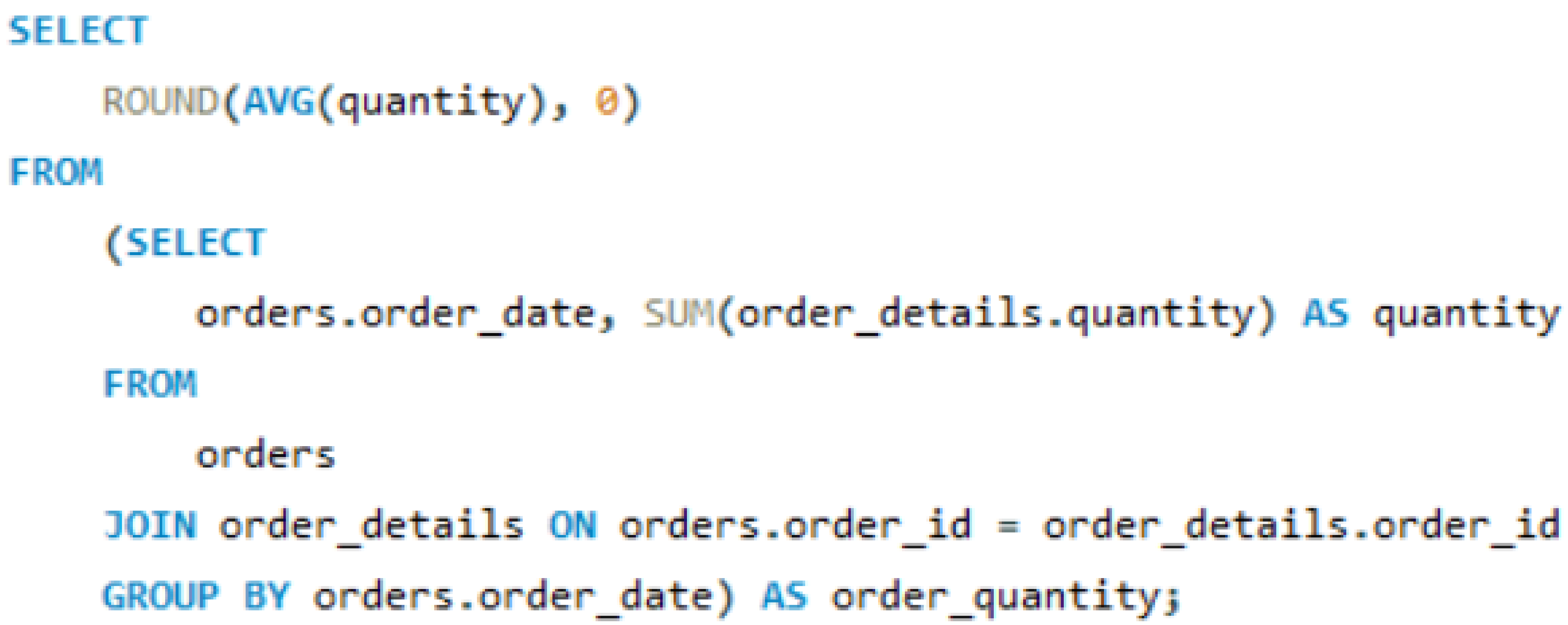


Hour	order_count
12	2520
13	2455
18	2399
17	2336
19	2009
16	1920
20	1642
14	1472
15	1468
11	1231
21	1198

This query counts the number of orders per hour, showing that the highest number of orders occurred during the 12th hour (noon), followed by the 13th, 18th, and 17th hours, with the 19th hour having a significant count.

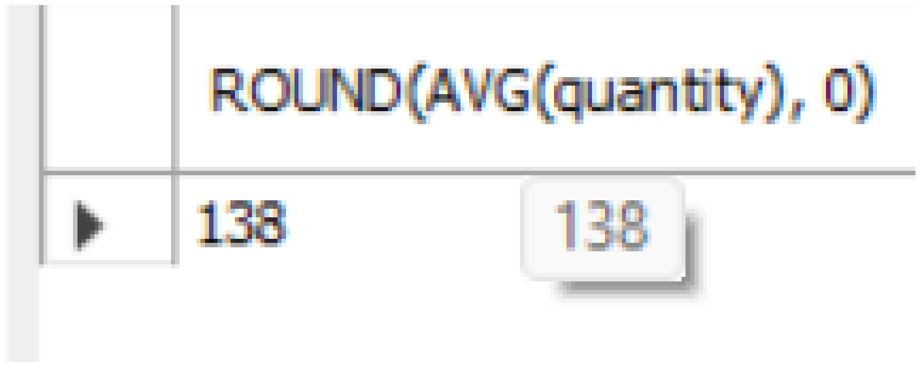
Group the orders by date and calculate the average number of pizzas ordered per day.

Query:



```
SELECT
    ROUND(AVG(quantity), 0)
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;
```

Output:

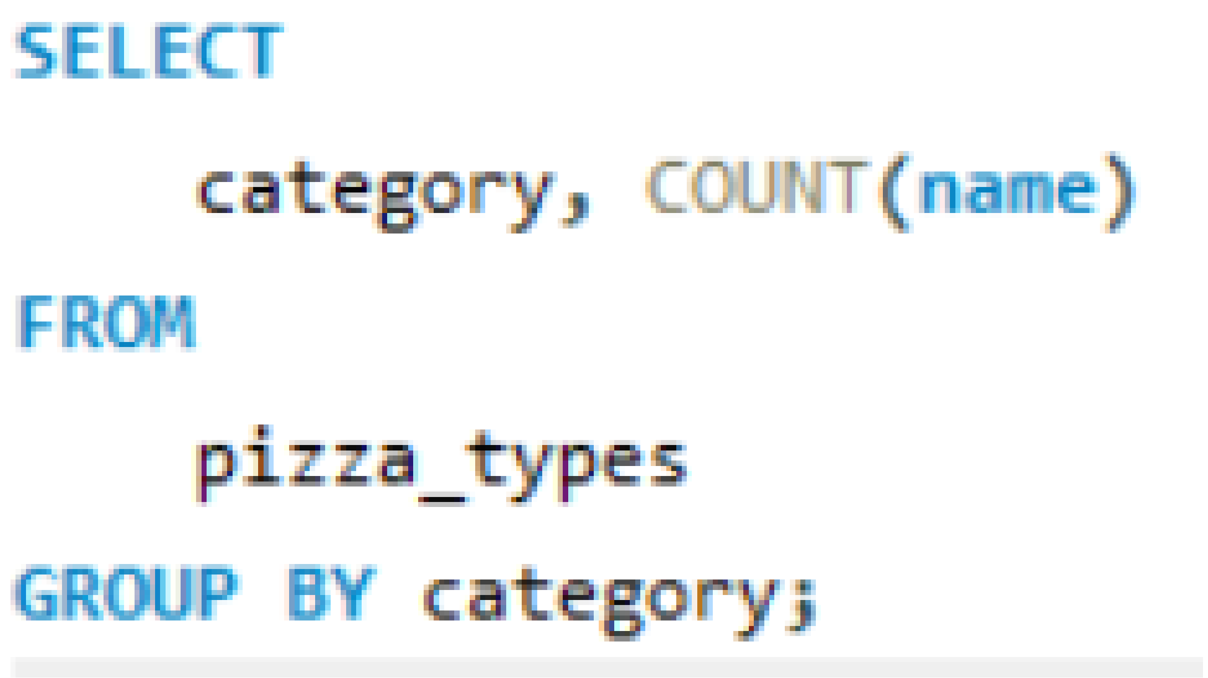


```
   ROUND(AVG(quantity), 0)
▶  138          138
```

This query calculates the average quantity of pizzas ordered per day. It first sums the total quantity of pizzas for each order date and then takes the average of these daily totals. The final output shows that the average quantity is approximately 138 pizzas per day.

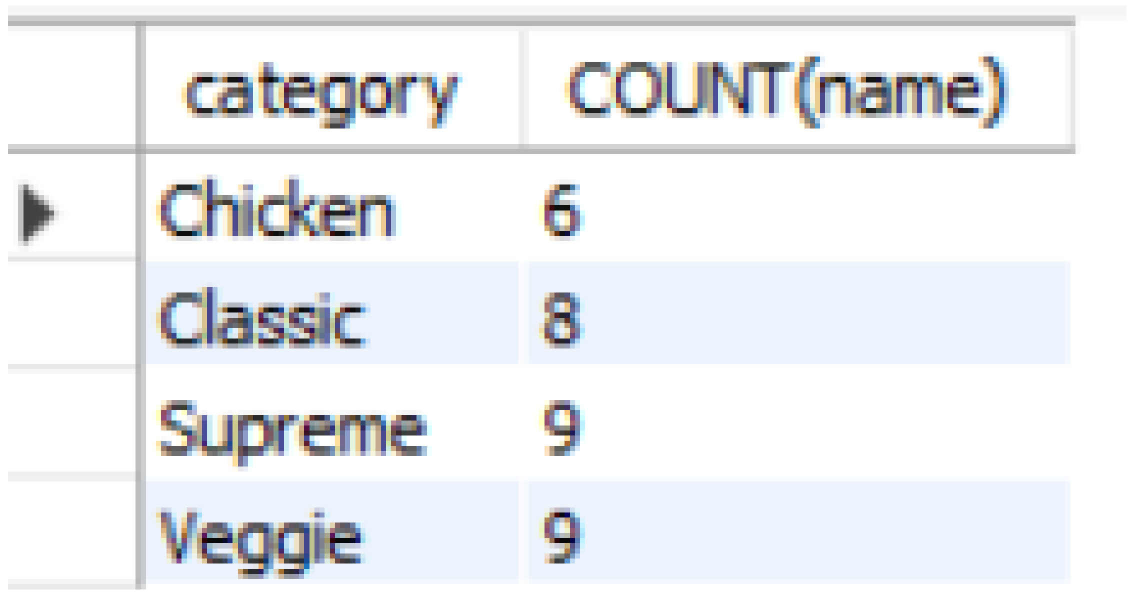
Join relevant tables to find the category-wise distribution of pizzas.

Query:



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

Output:

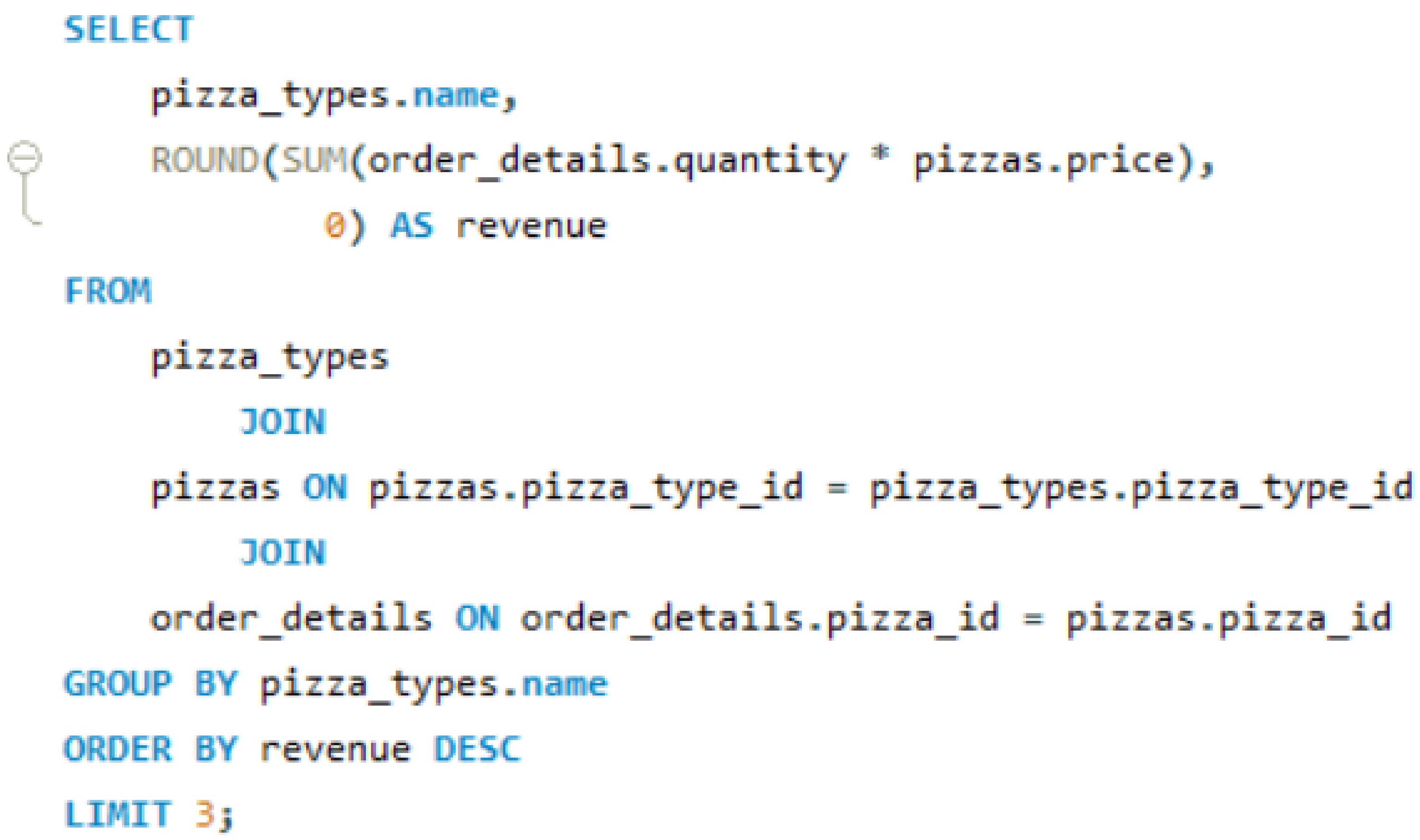


category	COUNT(name)
Chicken	6
Classic	8
Supreme	9
Veggie	9

This query counts the number of pizza types in each category. It groups the results by category, showing how many different pizza types fall under each category.

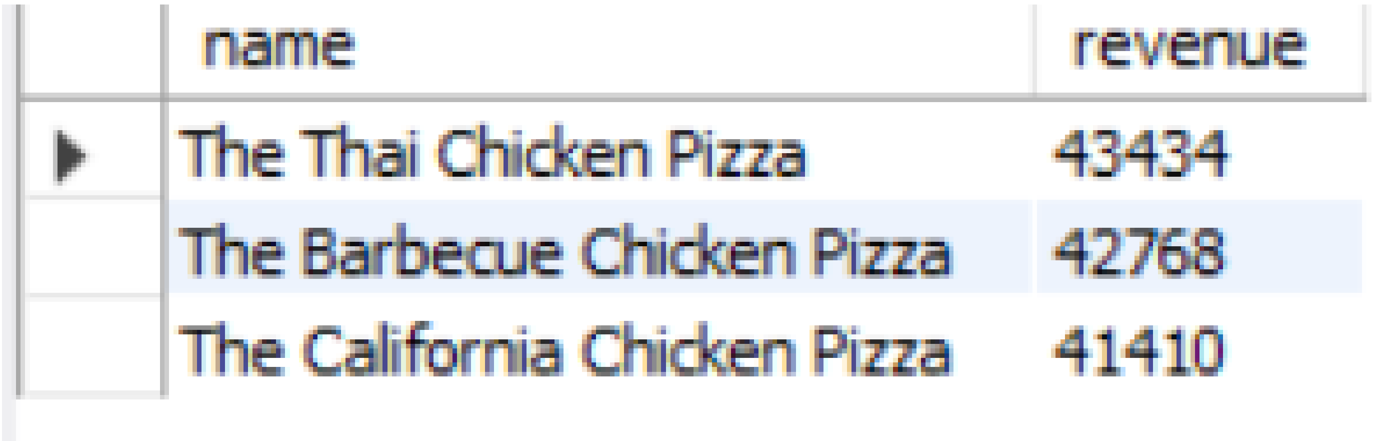
Determine the top 3 most ordered pizza types based on revenue.

Query:



```
SELECT
    pizza_types.name,
    ROUND(SUM(order_details.quantity * pizzas.price),
            0) 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;
```

Output:

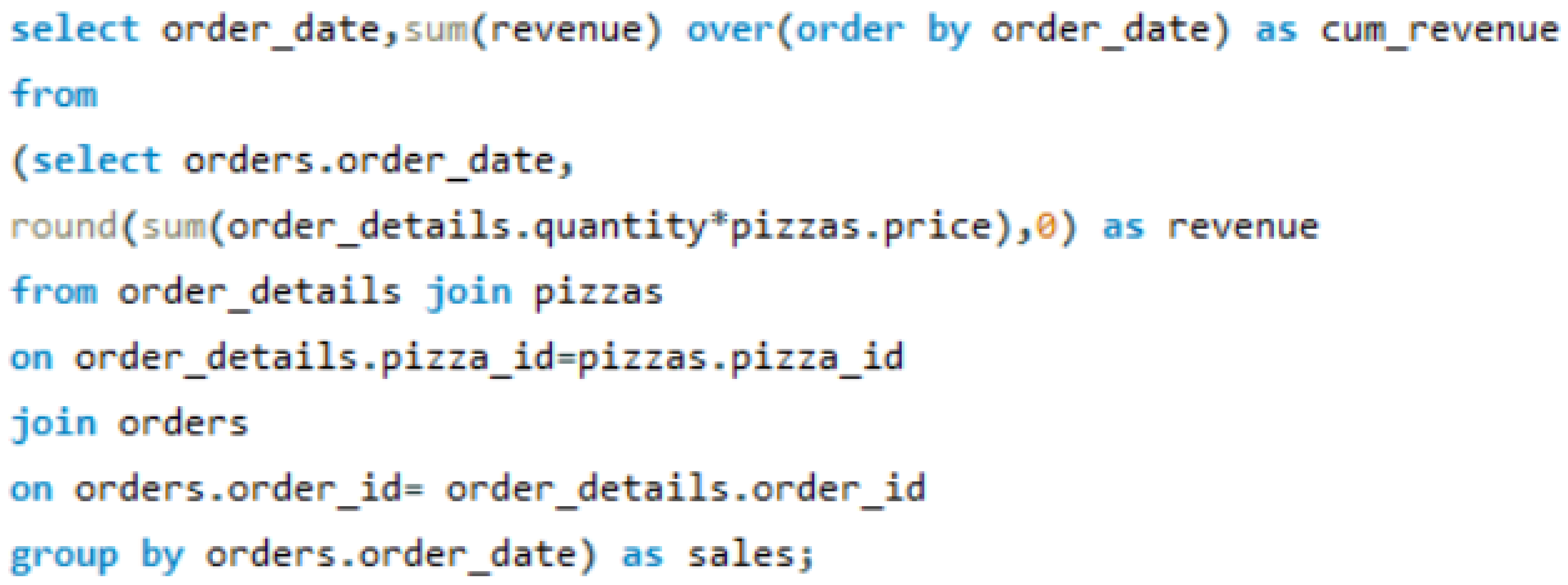


name	revenue
The Thai Chicken Pizza	43434
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41410

This query gets the top three pizza types by revenue, calculating total earnings by multiplying quantity ordered by price. The results are grouped by pizza type and sorted in descending order of revenue.

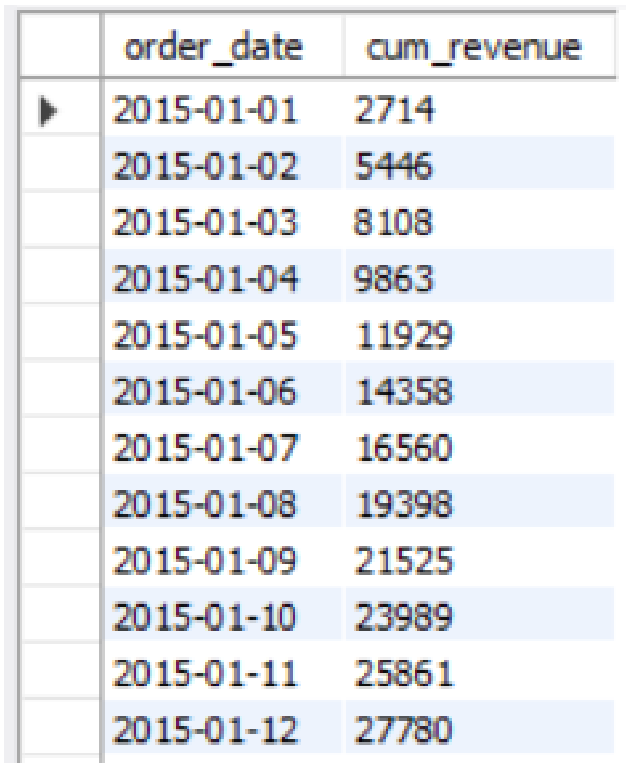
Analyze the cumulative revenue generated over time.

Query:



```
select order_date,sum(revenue) over(order by order_date) as cum_revenue
from
(select orders.order_date,
round(sum(order_details.quantity*pizzas.price),0) 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.order_date) as sales;
```

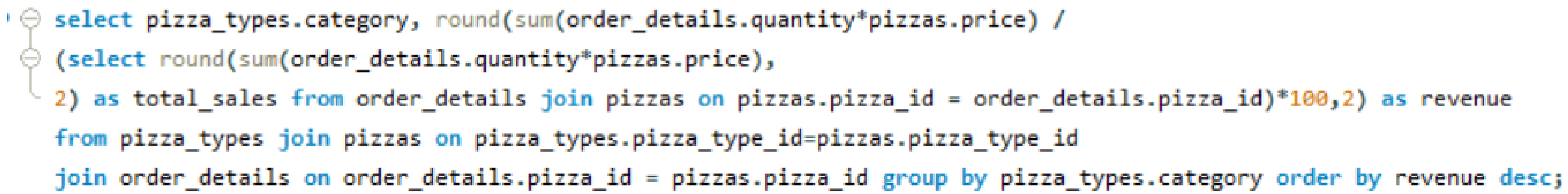
Output:



order_date	cum_revenue
2015-01-01	2714
2015-01-02	5446
2015-01-03	8108
2015-01-04	9863
2015-01-05	11929
2015-01-06	14358
2015-01-07	16560
2015-01-08	19398
2015-01-09	21525
2015-01-10	23989
2015-01-11	25861
2015-01-12	27780

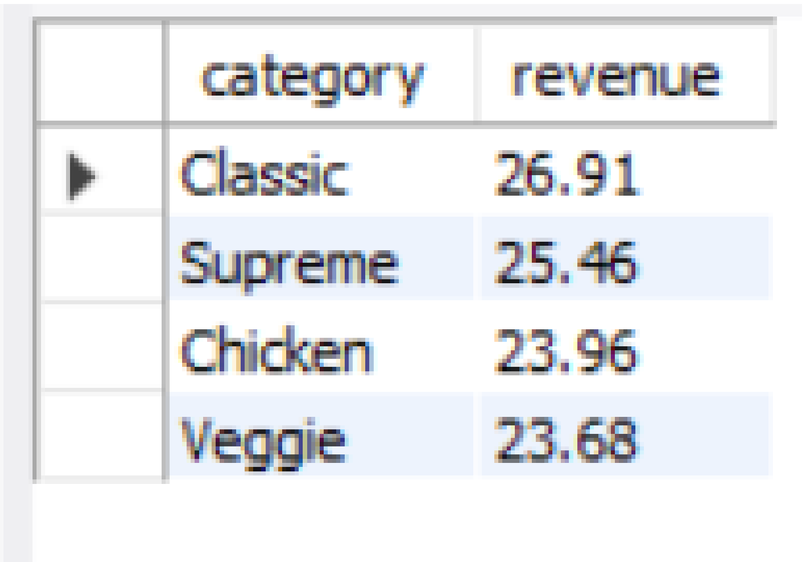
Calculate the percentage contribution of each pizza type to total revenue.

Query:



```
select pizza_types.category, round(sum(order_details.quantity*pizzas.price) /
(select round(sum(order_details.quantity*pizzas.price),
2) as total_sales 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;
```

Output:

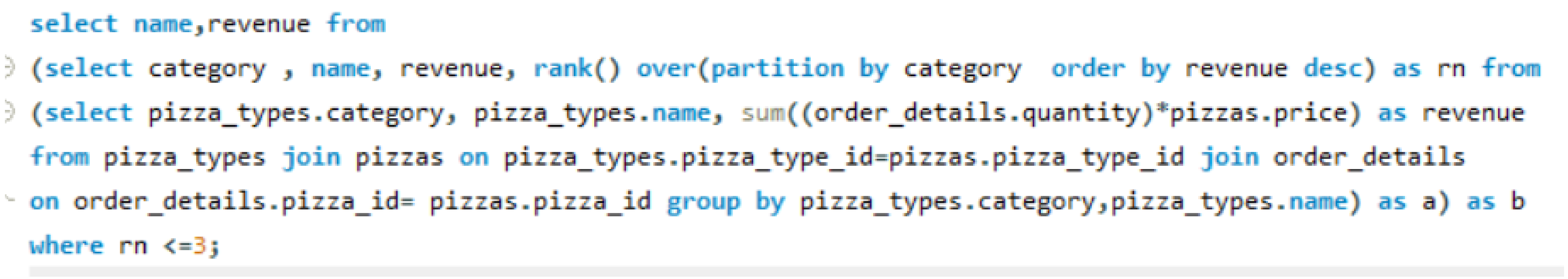


	category	revenue
▶	Classic	26.91
	Supreme	25.46
	Chicken	23.96
	Veggie	23.68

This query calculates the percentage of total revenue each pizza category generates. It multiplies the quantity sold by the price, sums that up for each category, and then divides it by the total sales of all pizzas. The result is sorted, so we can see which category contributes the most to overall sales.

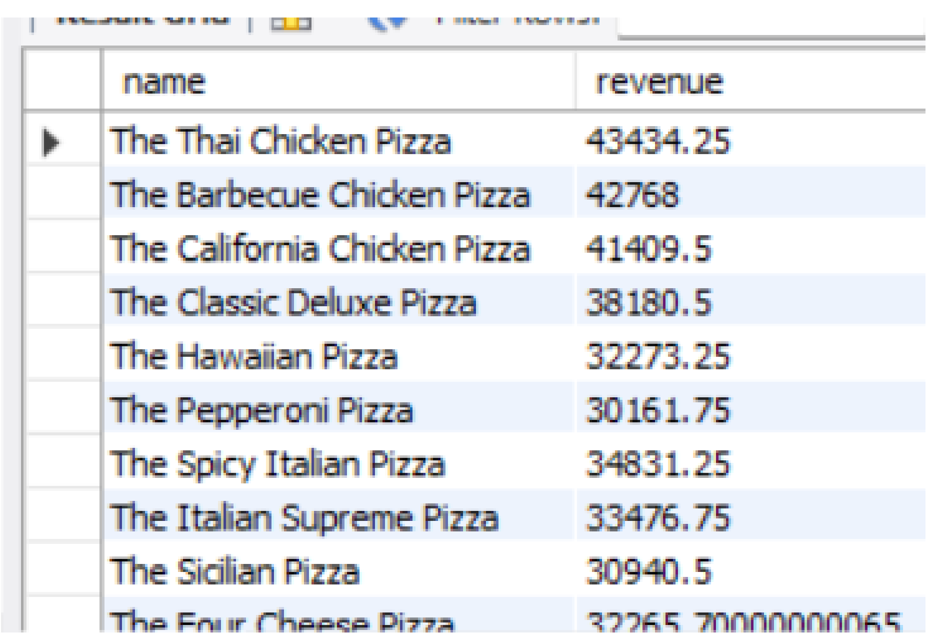
Determine the top 3 most ordered pizza types based on revenue for each pizza category.

Query:



```
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;
```

Output:



```
  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
```

This query finds the top 3 pizzas by revenue within each category. First, it calculates the revenue for each pizza by multiplying the quantity sold by the price. Then, it ranks the pizzas within their categories based on revenue. Finally, it selects only the top 3 pizzas per category and displays their names along with the revenue they generated.

INSIGHTS GAINED:-

1.Total revenue: $817,860.05

2.Highest priced pizza: Greek pizza ($35.95)

3.Most common size ordered: Medium

4.Busiest hours: 12th, 13th, 18th, 17th, 19th

5.Avg pizzas ordered: 138

6.Top 5 pizzas ordered by quantity: Classic Deluxe, BBQ Chicken, Hawaiian, Pepperoni, Thai Chicken

7.Top 3 pizzas by revenue: Thai Chicken, BBQ Chicken, California

8.Revenue contribution: Classic: 26.91%, Supreme: 25.46%, Chicken: 23.96%, Veggie: 23.68%