Delicious

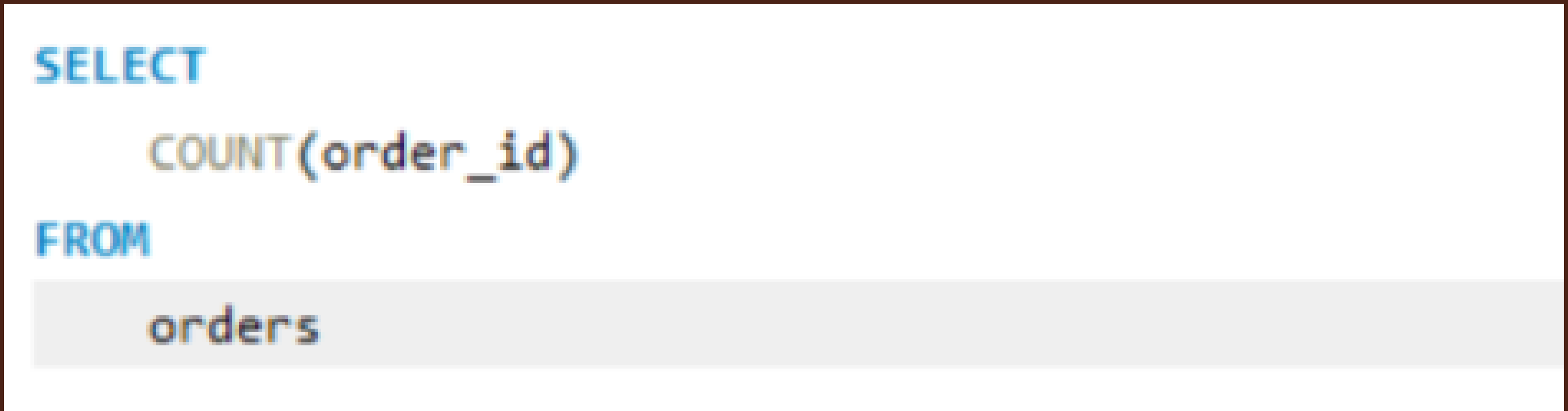
PIZZA

Presented by

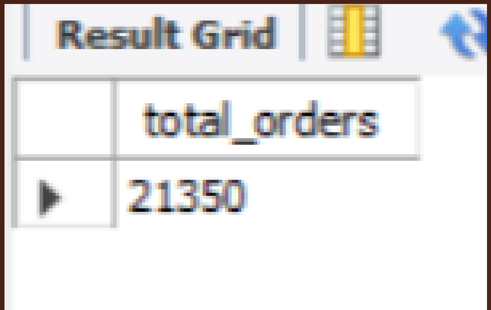
Ayush Dhasmana

This report provides an in-depth analysis of pizza orders, types, and details based on four datasets: orders_table_1.csv, pizza_types_table_1.csv, pizzas_table_1.csv, and order_details_table_1.csv.

1. Retrieve the total number of orders placed.

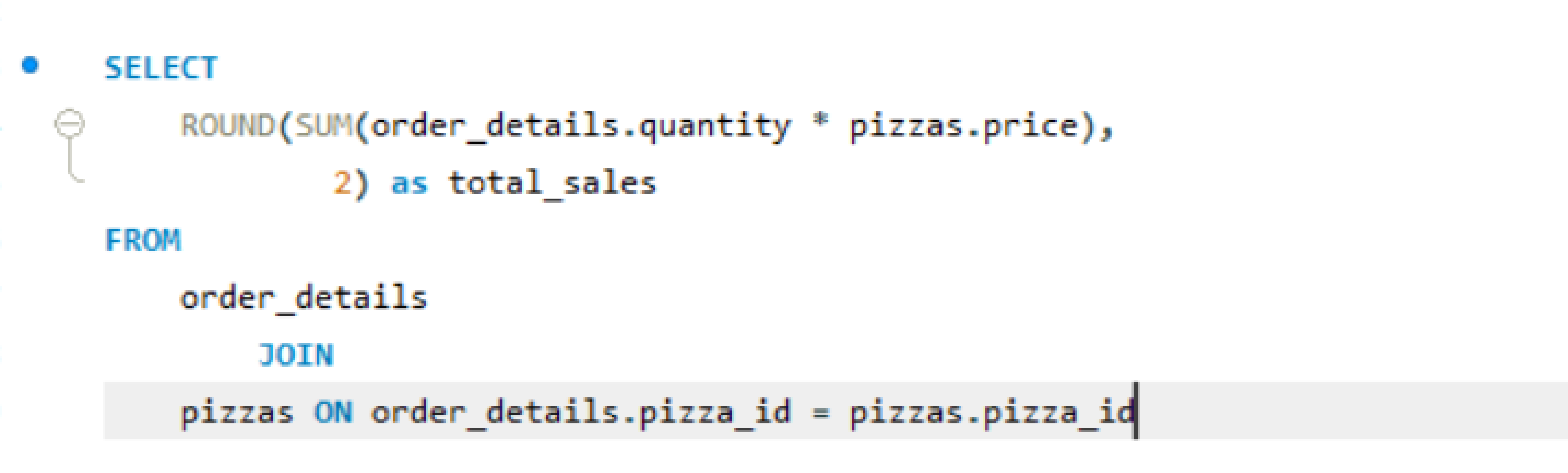


```
SELECT
    COUNT(order_id)
FROM
    orders
```

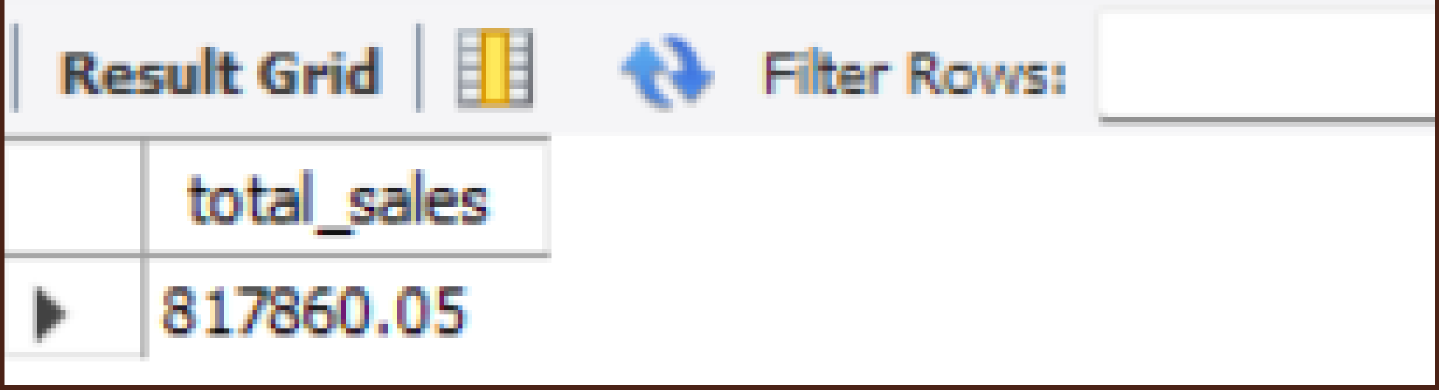


total_orders
21350

2. Calculate the total revenue generated from pizza sales.

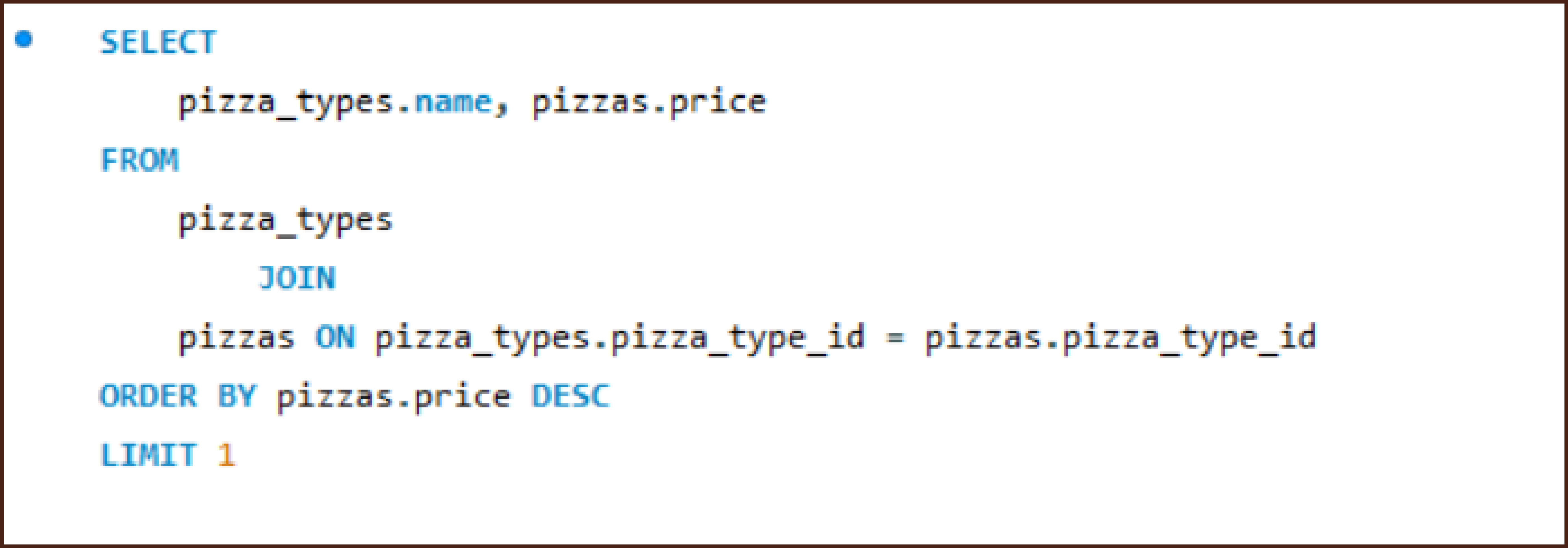


```
SELECT
    ROUND(SUM(order_details.quantity * pizzas.price),
            2) as total_sales
FROM
    order_details
        JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id
```

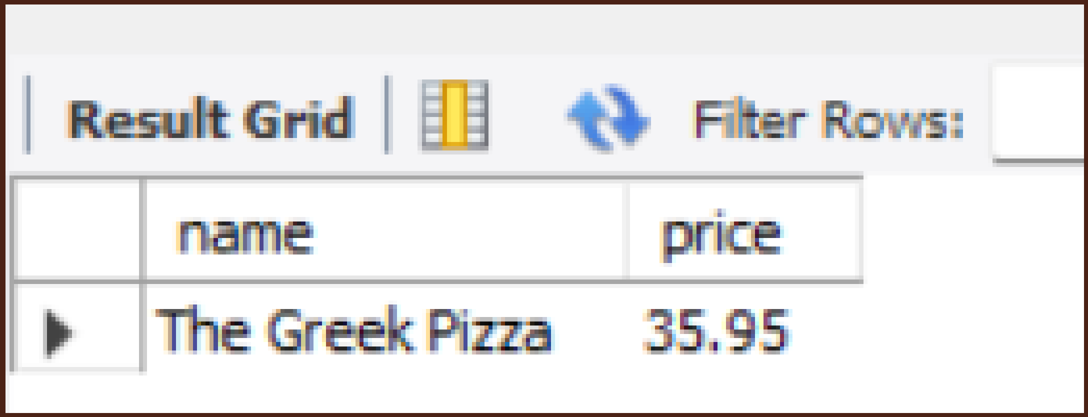


total_sales
817860.05

3. Identify the highest-priced pizza.

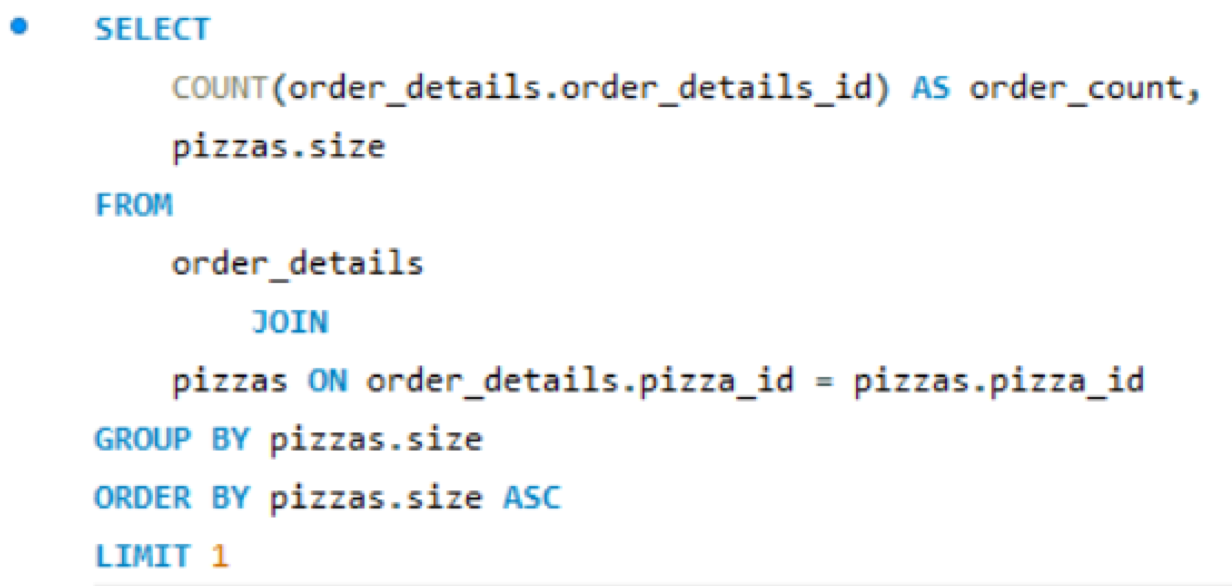


```sql
SELECT
    pizza_types.name, pizzas.price
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
ORDER BY pizzas.price DESC
LIMIT 1
```

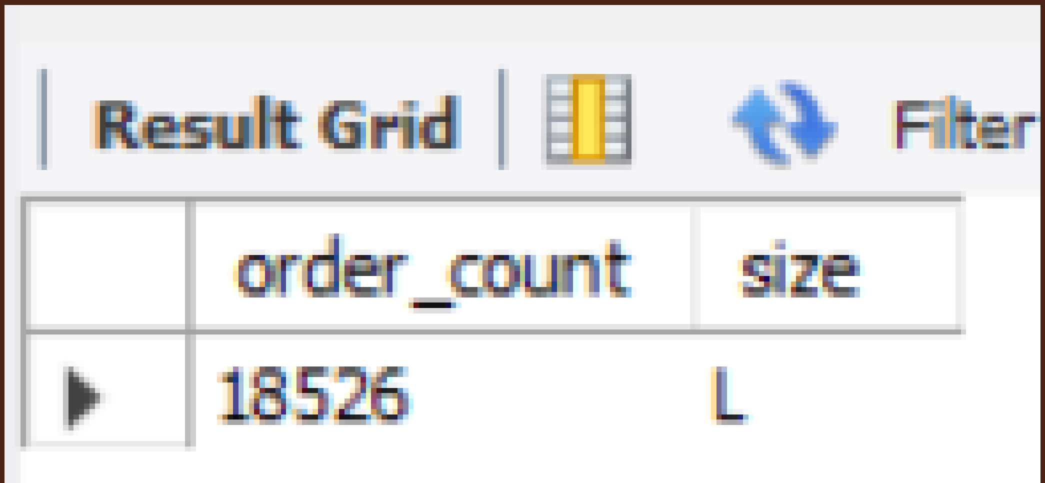


name	price
The Greek Pizza	35.95

4. Identify the most common pizza size ordered.

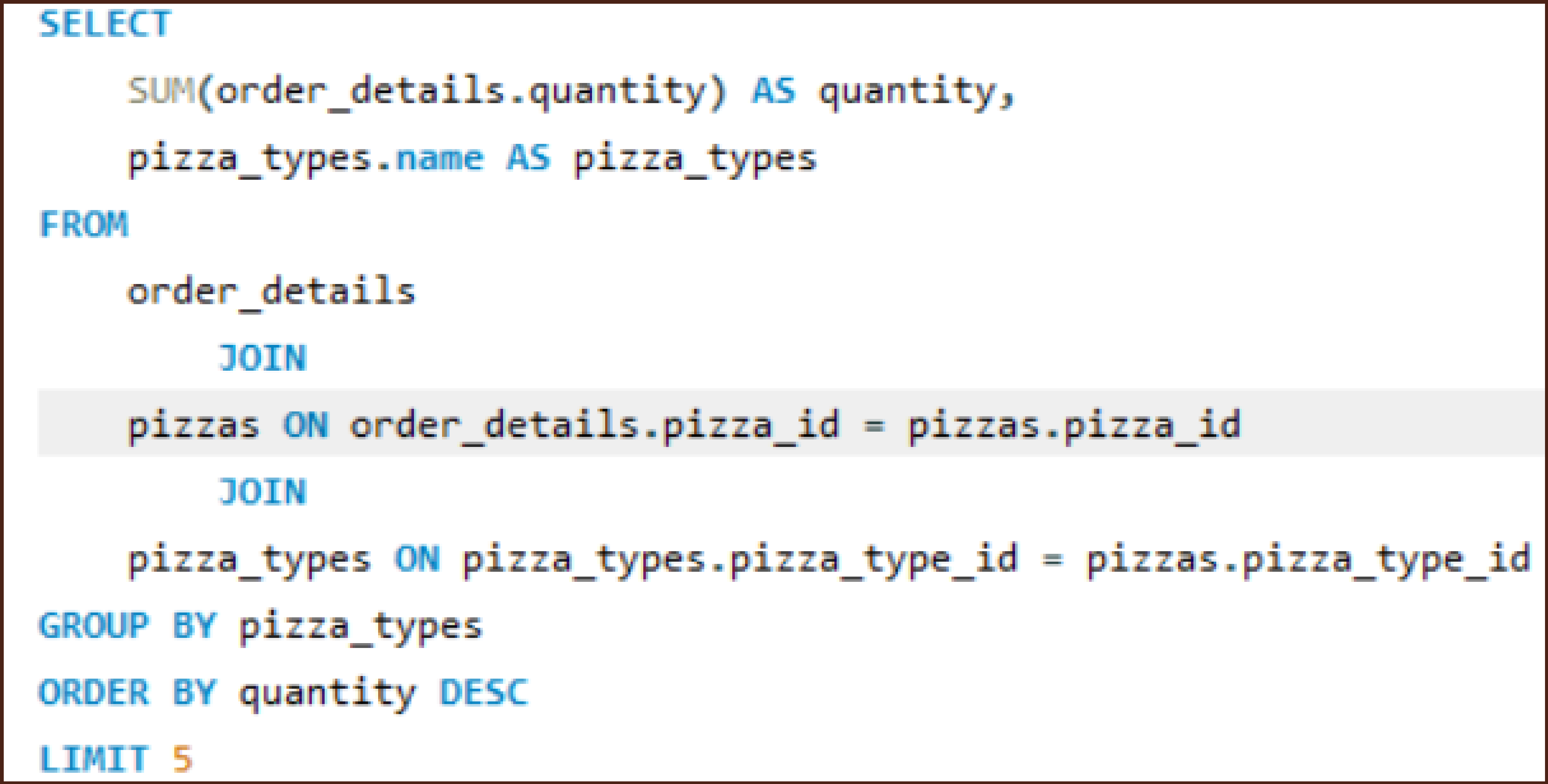


```
SELECT
    COUNT(order_details.order_details_id) AS order_count,
    pizzas.size
FROM
    order_details
        JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizzas.size
ORDER BY pizzas.size ASC
LIMIT 1
```

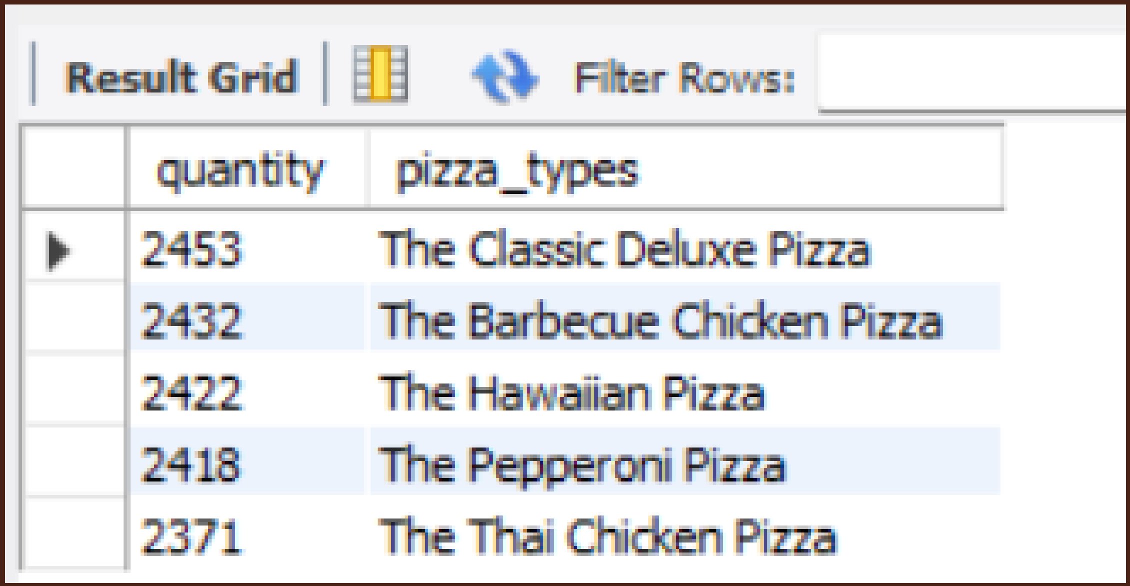


```
order_count   size
18526         L
```

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

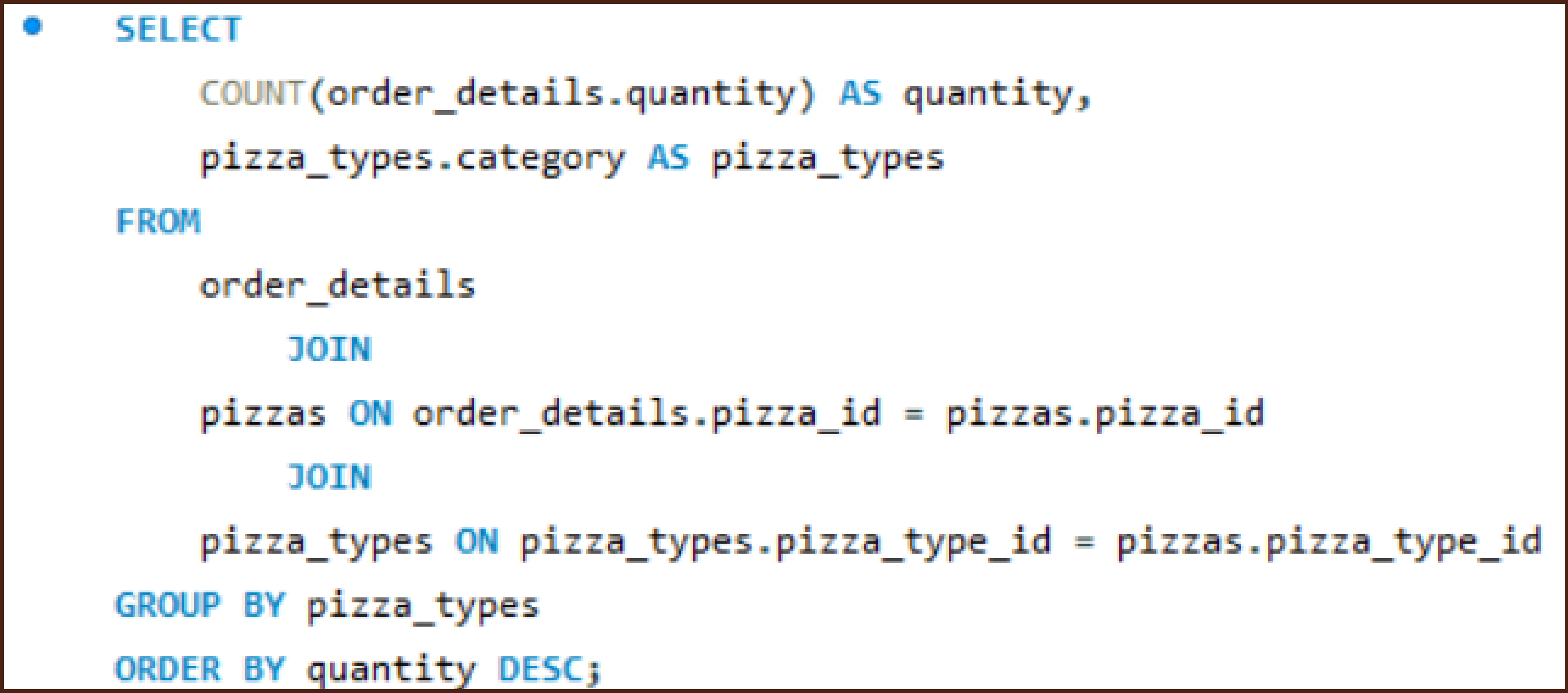


```
SELECT
    SUM(order_details.quantity) AS quantity,
    pizza_types.name AS pizza_types
FROM
    order_details
        JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id
        JOIN
    pizza_types ON pizza_types.pizza_type_id = pizzas.pizza_type_id
GROUP BY pizza_types
ORDER BY quantity DESC
LIMIT 5
```

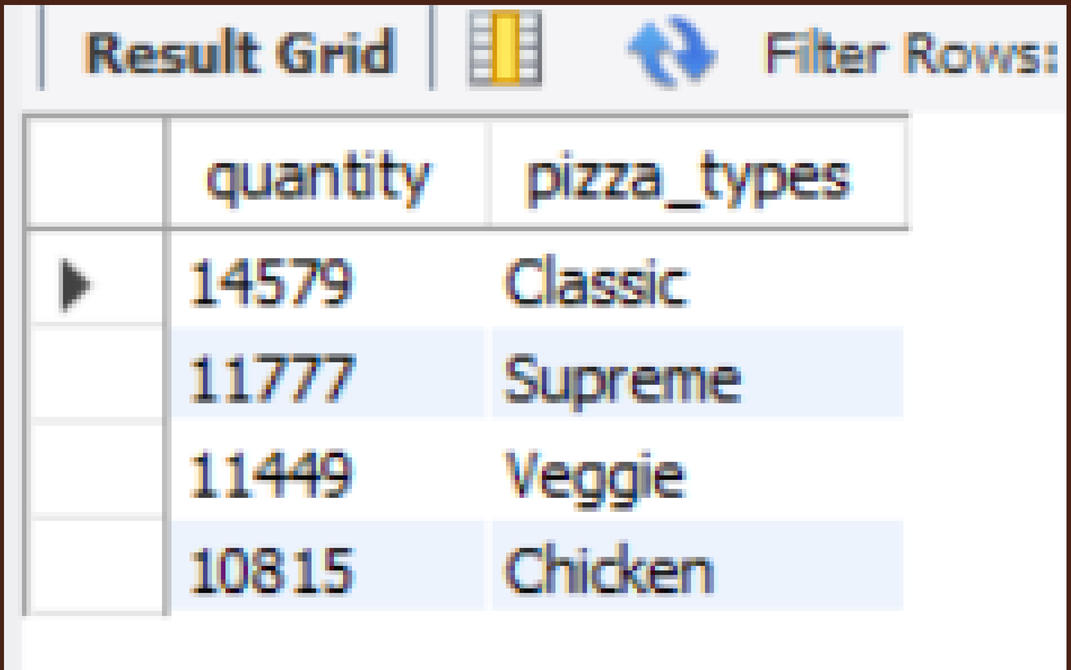


	quantity	pizza_types
▶	2453	The Classic Deluxe Pizza
	2432	The Barbecue Chicken Pizza
	2422	The Hawaiian Pizza
	2418	The Pepperoni Pizza
	2371	The Thai Chicken Pizza

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

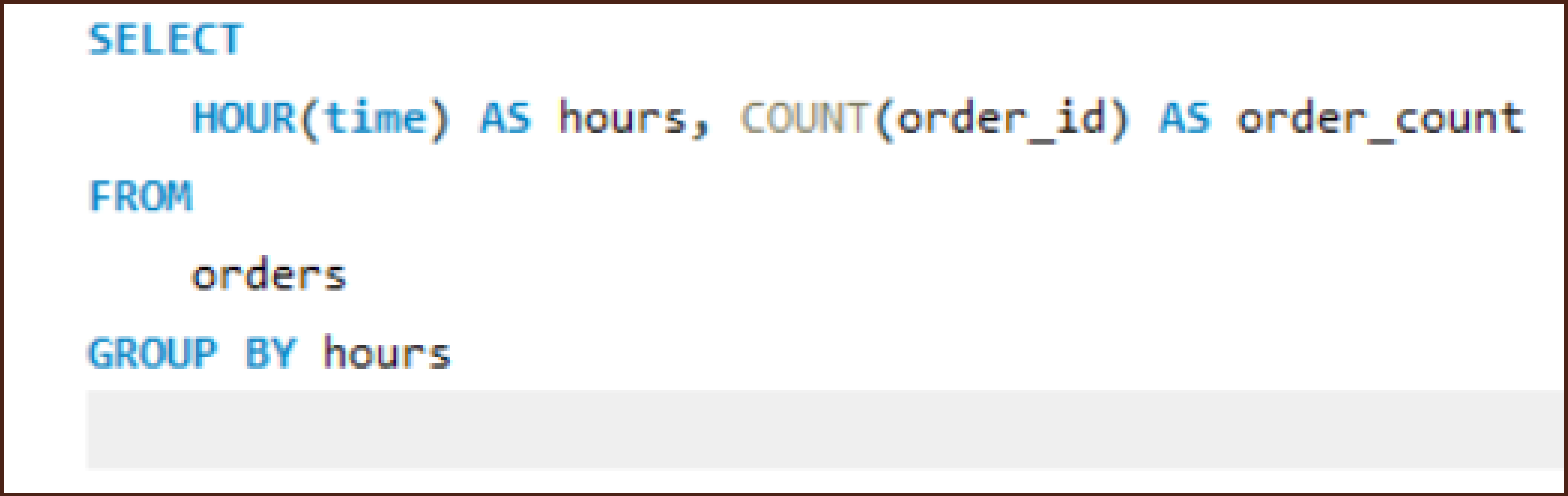


```sql
SELECT
    COUNT(order_details.quantity) AS quantity,
    pizza_types.category AS pizza_types
FROM
    order_details
        JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id
        JOIN
    pizza_types ON pizza_types.pizza_type_id = pizzas.pizza_type_id
GROUP BY pizza_types
ORDER BY quantity DESC;
```

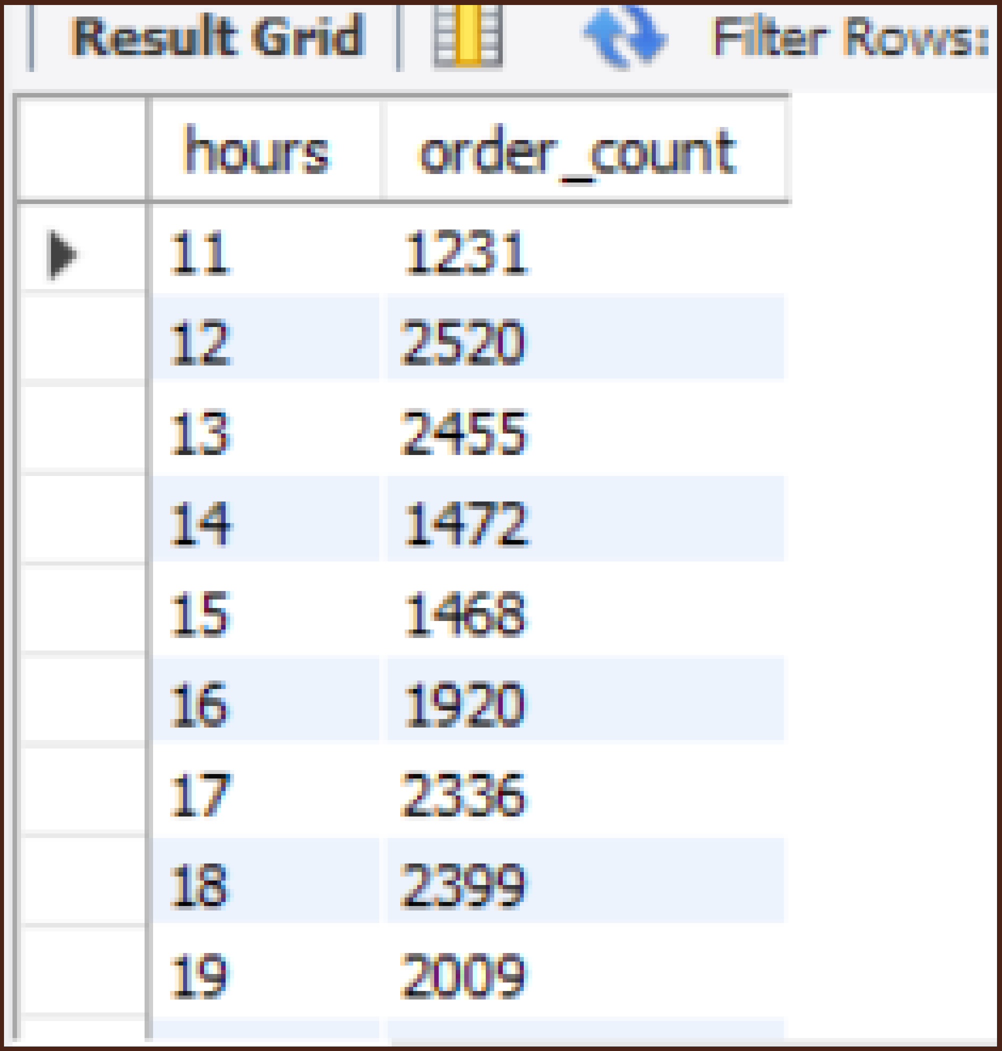


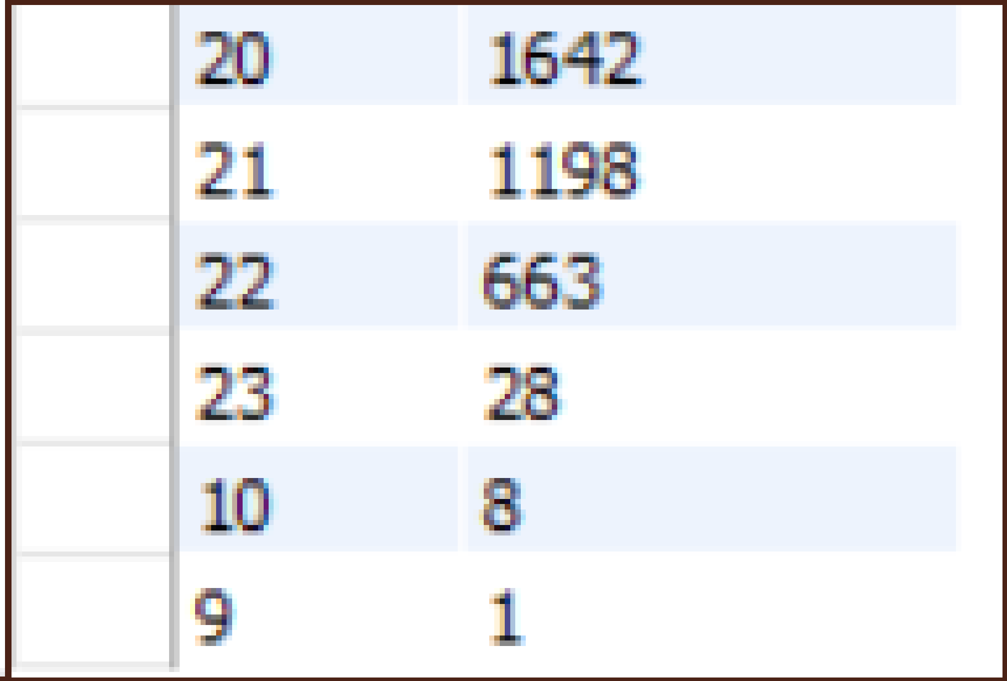
	quantity	pizza_types
▶	14579	Classic
	11777	Supreme
	11449	Veggie
	10815	Chicken

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



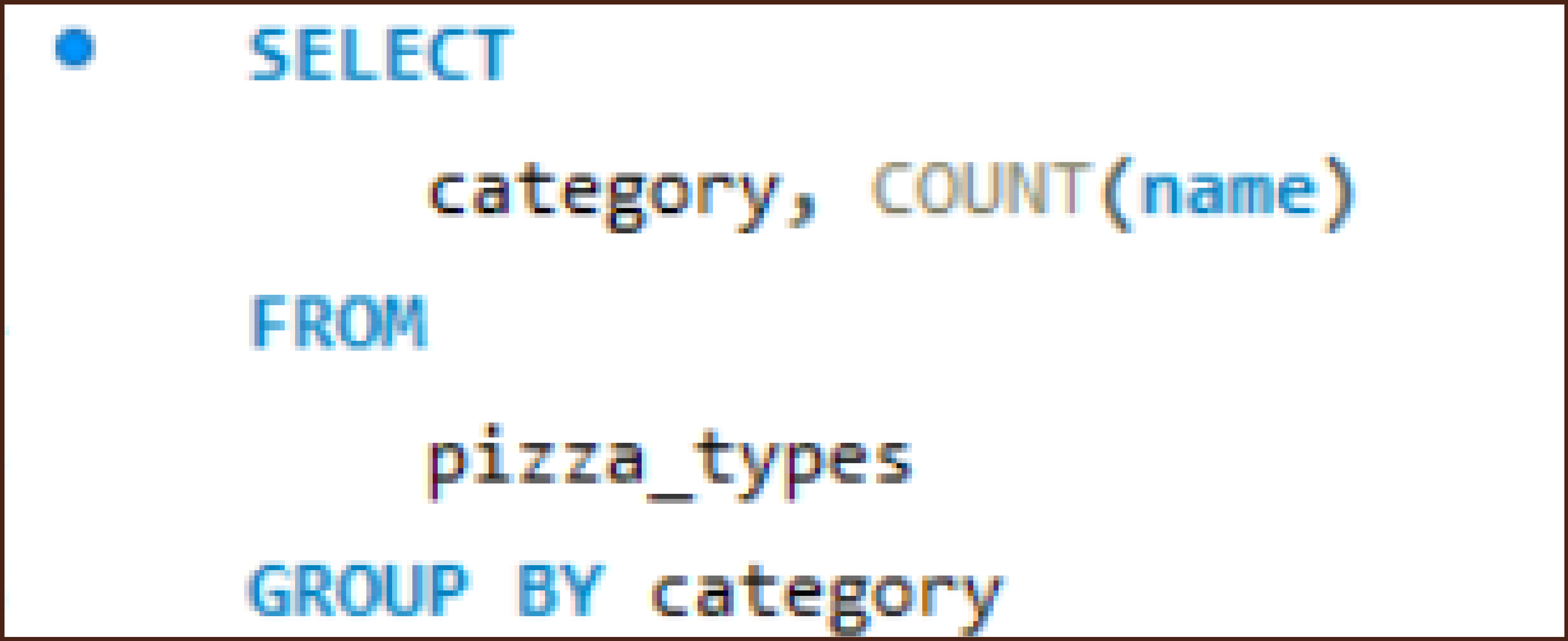
```
SELECT
    HOUR(time) AS hours, COUNT(order_id) AS order_count
FROM
    orders
GROUP BY hours
```



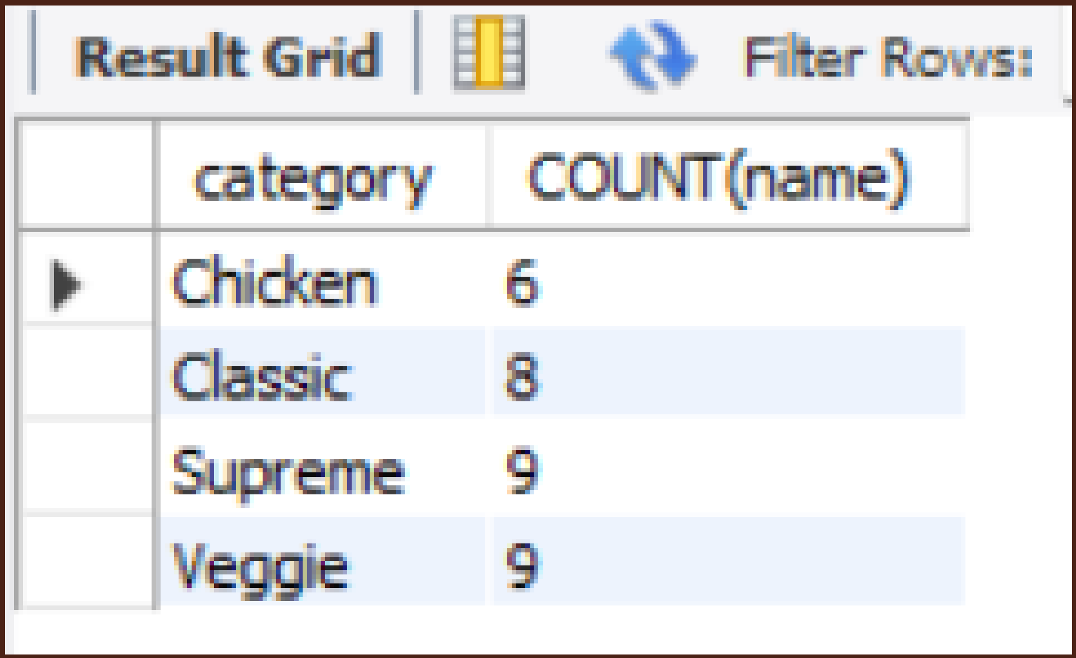


hours	order_count
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642
21	1198
22	663
23	28
10	8
9	1

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

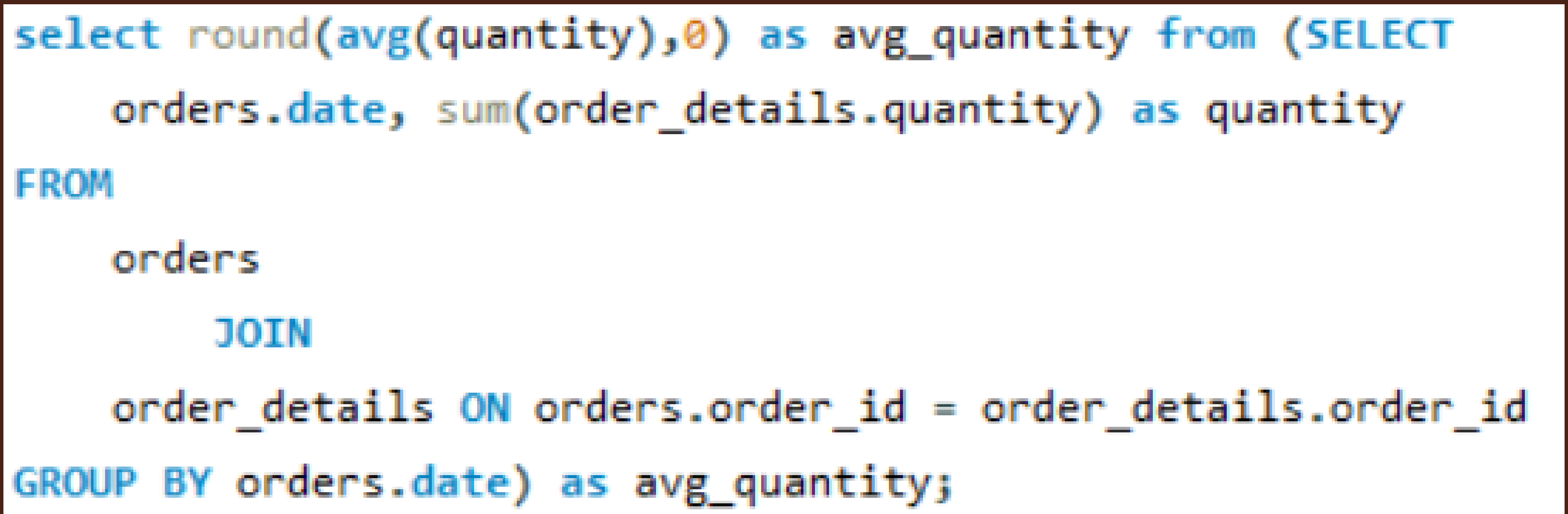


```
SELECT
    category, COUNT(name)
FROM
    pizza_types
GROUP BY category
```

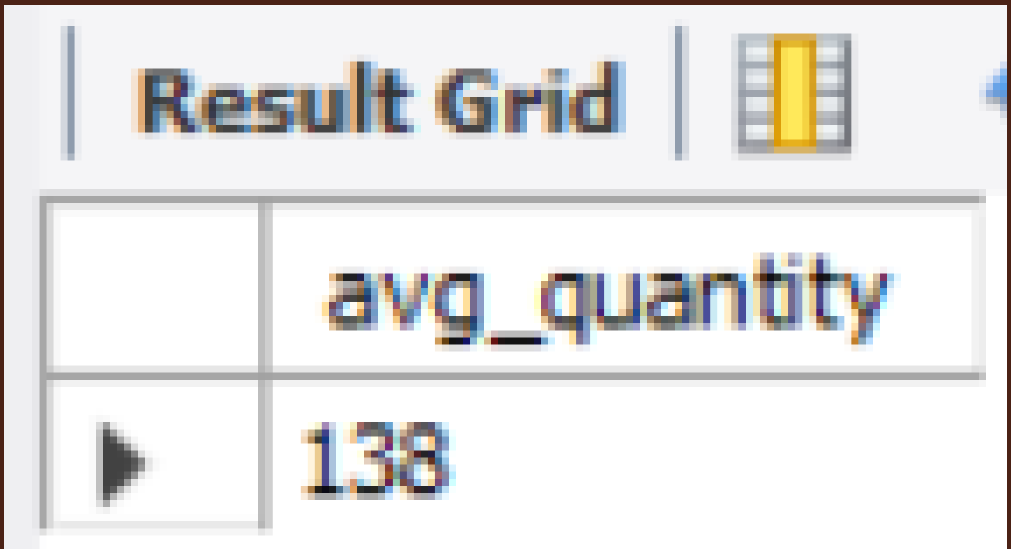


category	COUNT(name)
Chicken	6
Classic	8
Supreme	9
Veggie	9

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

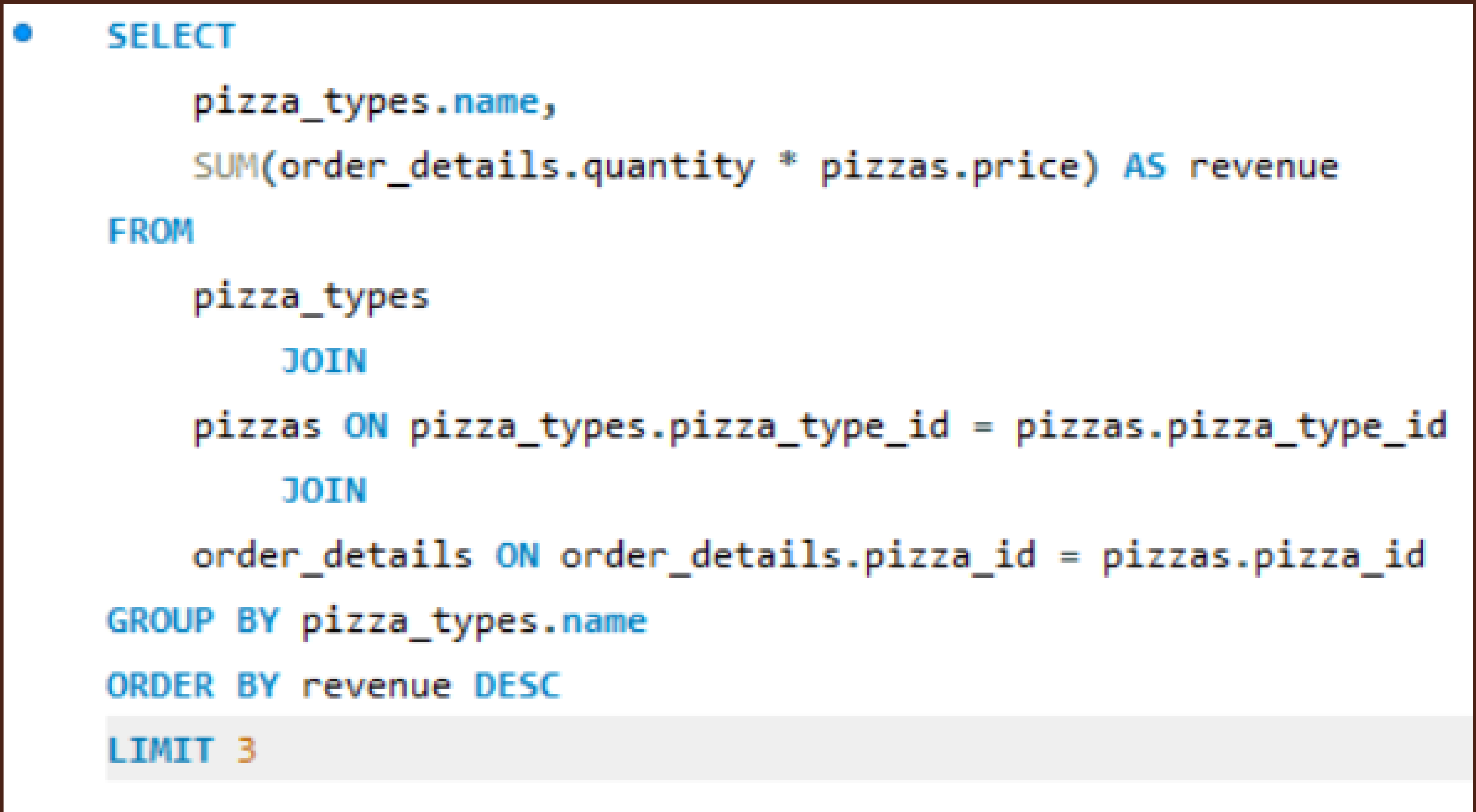


```
select round(avg(quantity),0) as avg_quantity from (SELECT
    orders.date, sum(order_details.quantity) as quantity
FROM
    orders
        JOIN
    order_details ON orders.order_id = order_details.order_id
GROUP BY orders.date) as avg_quantity;
```

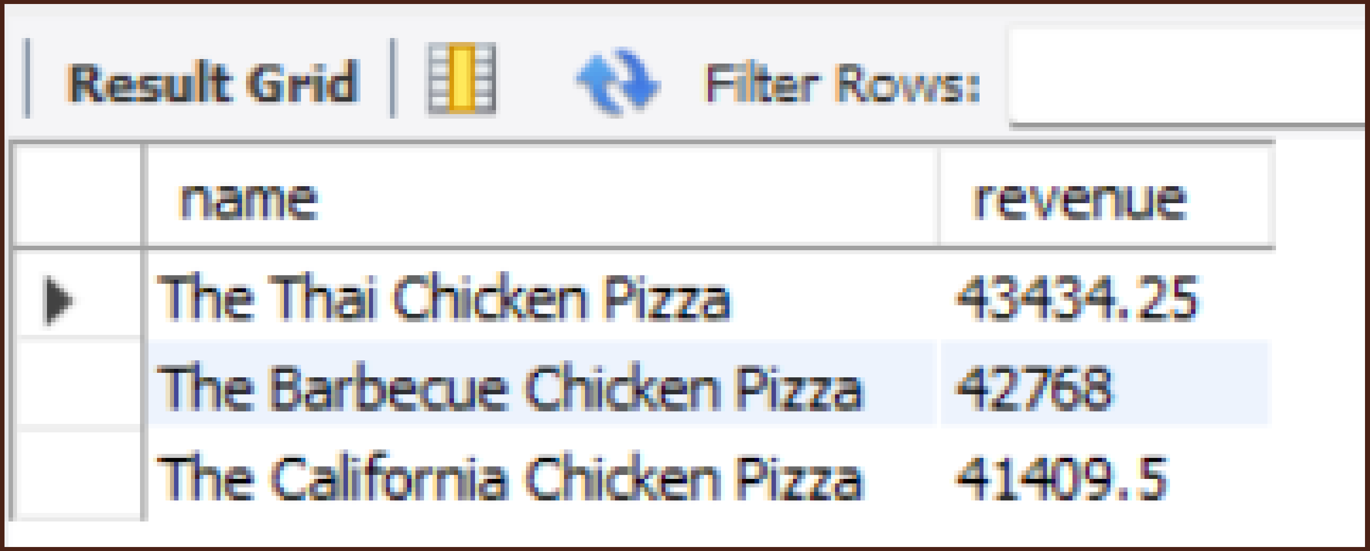


```
   avg_quantity
▶  138
```

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

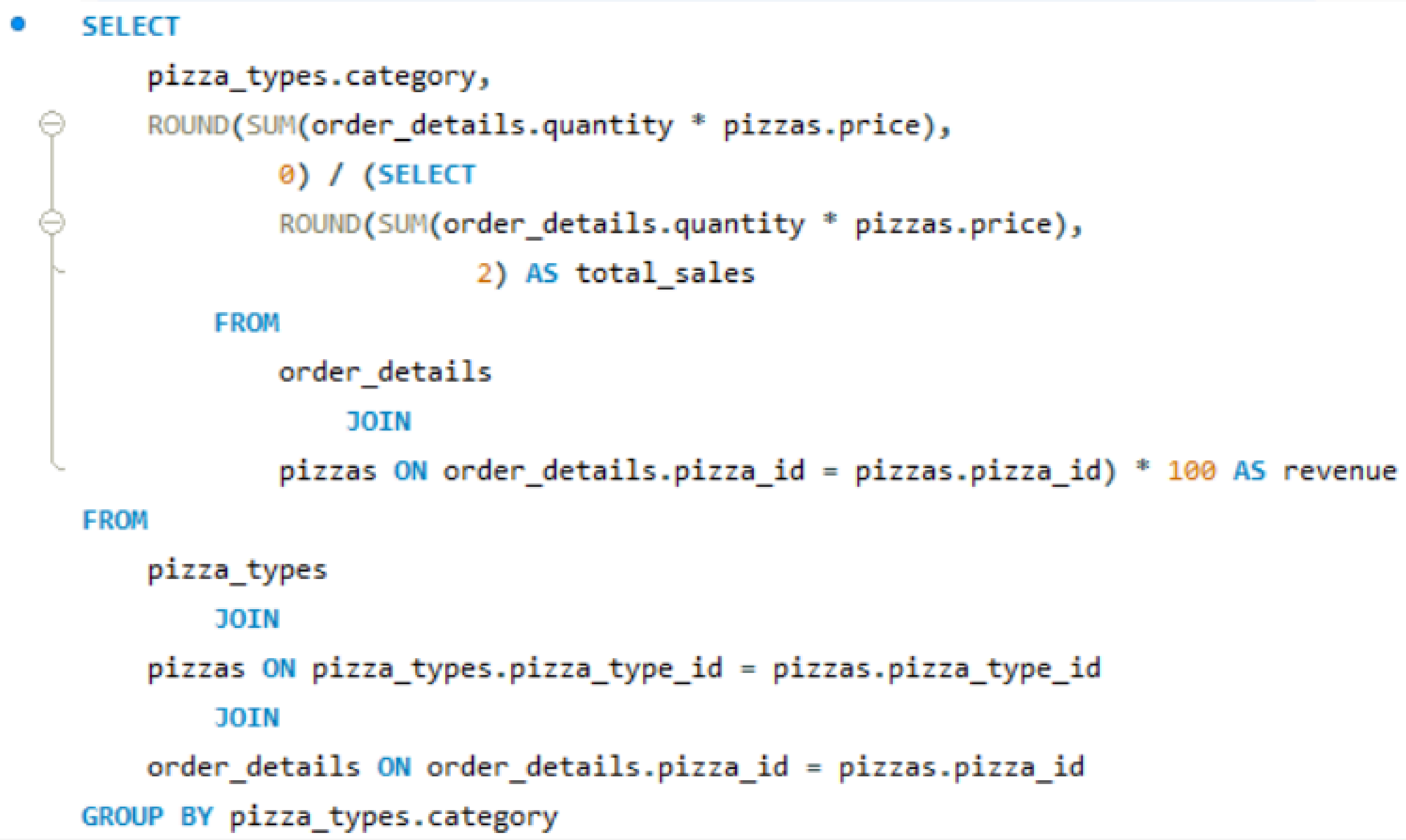


```
SELECT
    pizza_types.name,
    SUM(order_details.quantity * pizzas.price) AS revenue
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY revenue DESC
LIMIT 3
```

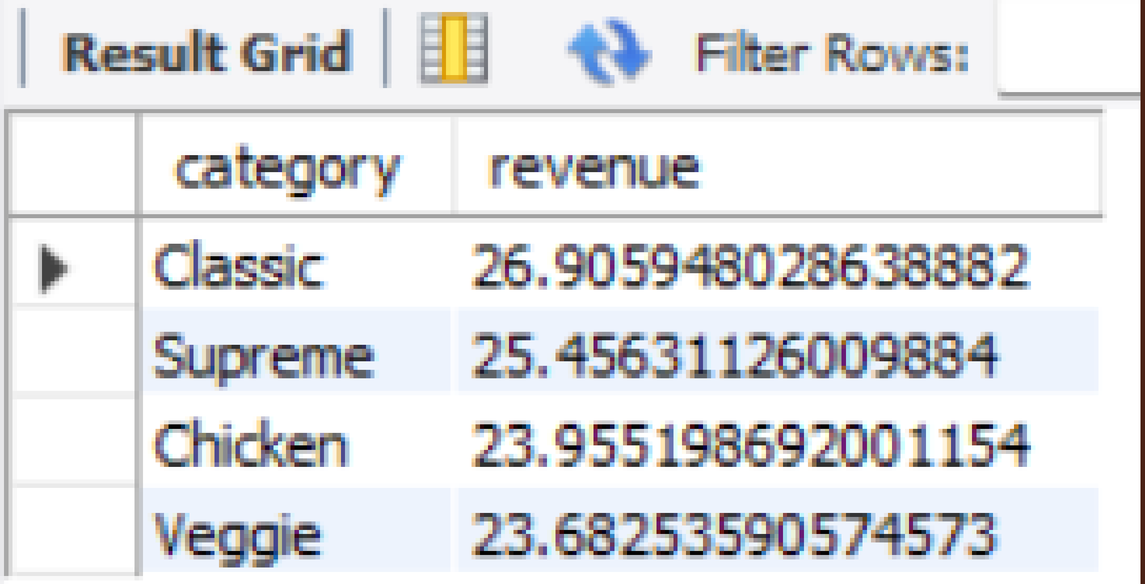


name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5

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

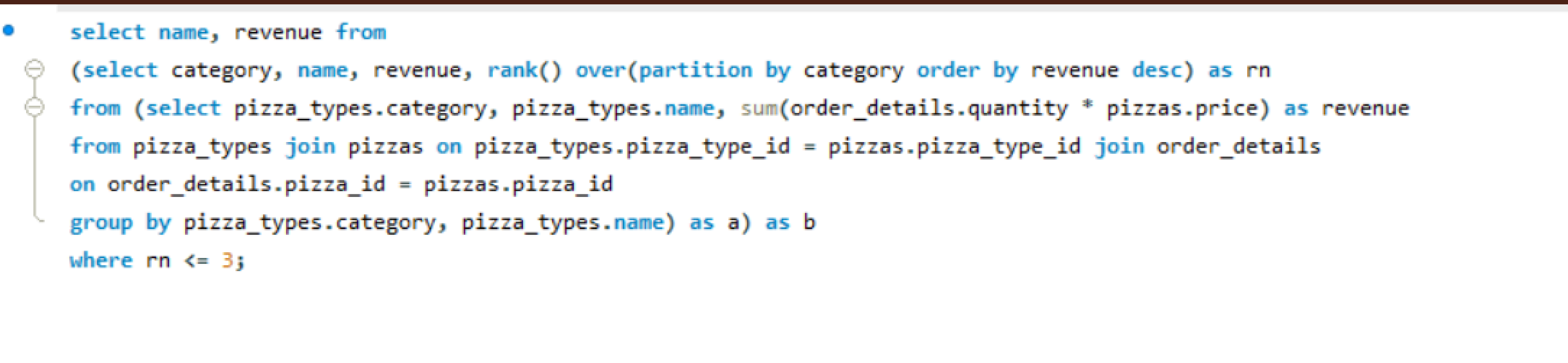


```
SELECT
    pizza_types.category,
    ROUND(SUM(order_details.quantity * pizzas.price),
            0) / (SELECT
            ROUND(SUM(order_details.quantity * pizzas.price),
                        2) AS total_sales
        FROM
            order_details
                JOIN
            pizzas ON order_details.pizza_id = pizzas.pizza_id) * 100 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
```

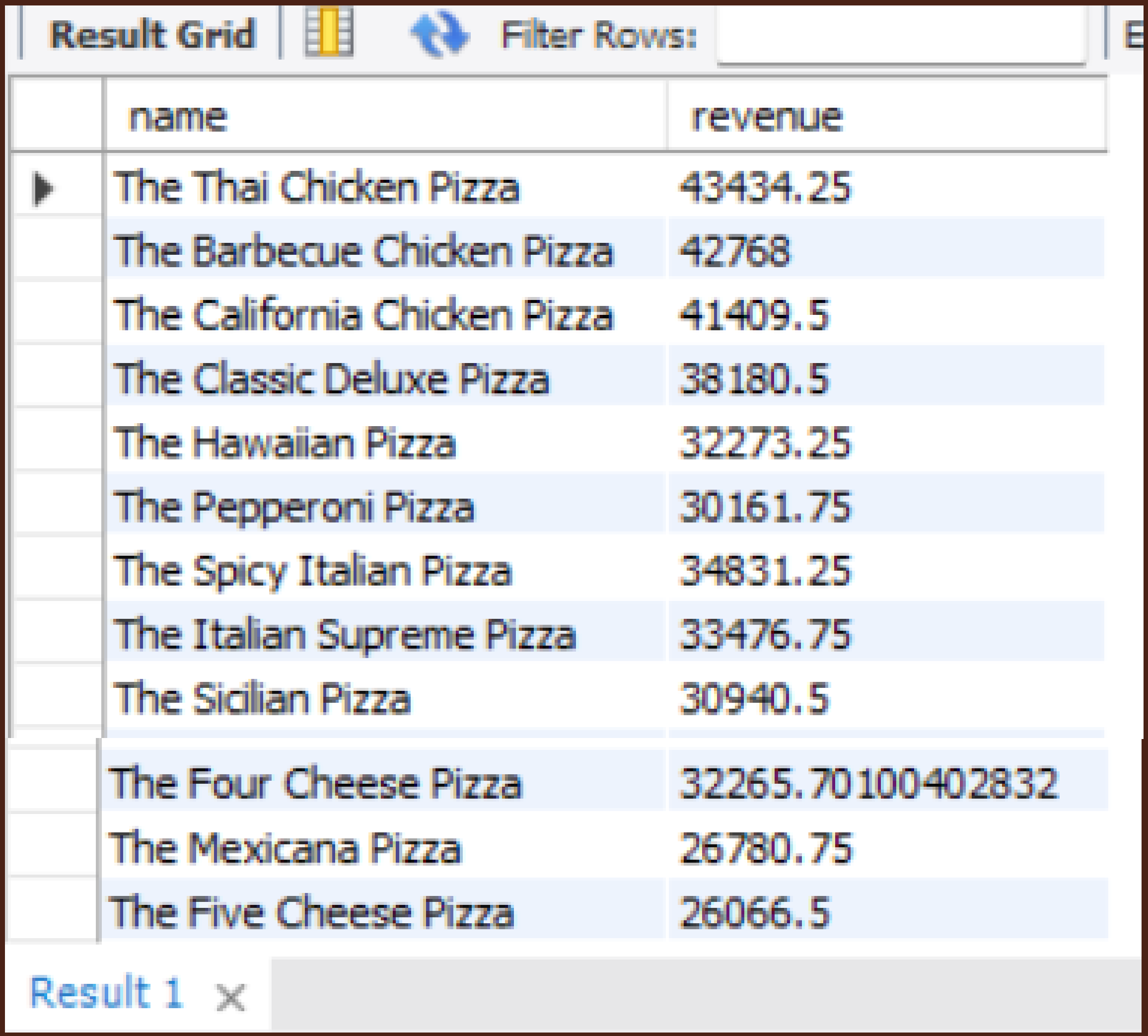


category	revenue
Classic	26.905948028638882
Supreme	25.4563112600984
Chicken	23.955198692001154
Veggie	23.68253590574573

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



```sql
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;
```



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	32265.70100402832
The Mexicana Pizza	26780.75
The Five Cheese Pizza	26066.5