SALES REPORT

2024

presented by
Karan Mehta

→

INTRODUCTION

2024

Hello, I’m Karan Mehta, and I’m excited to present my project today. As a data analyst, I utilized SQL queries to extract meaningful insights from our dataset.

In this project, I focused on addressing key business questions that can drive informed decision-making. By leveraging SQL, I was able to analyze trends, identify patterns, and uncover valuable information that can enhance our strategies.

My objectives included:

- Analyzing customer behavior to understand purchasing trends.
- Identifying potential areas for improvement in our processes.

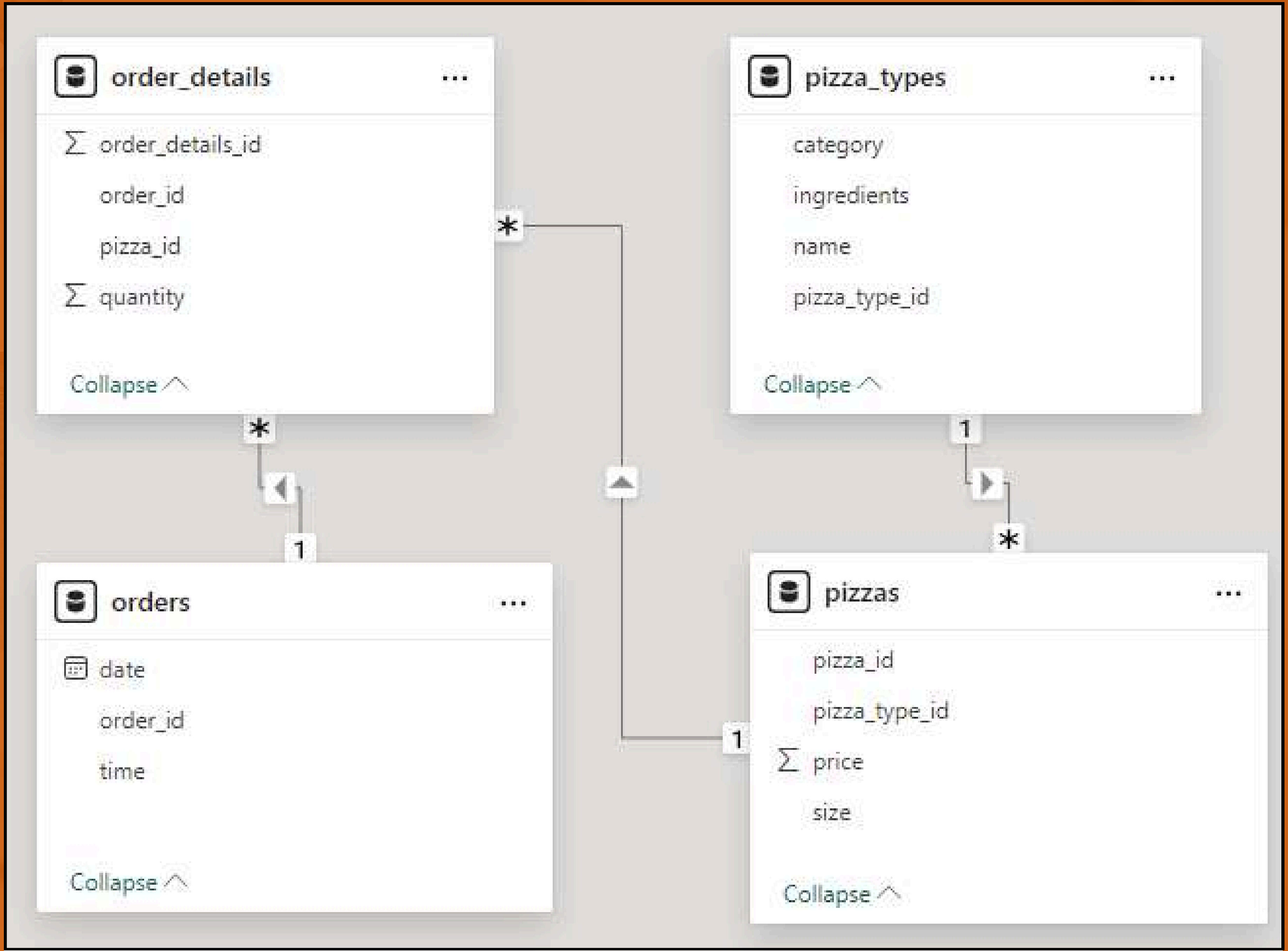
presented by

karan mehta

DATASET

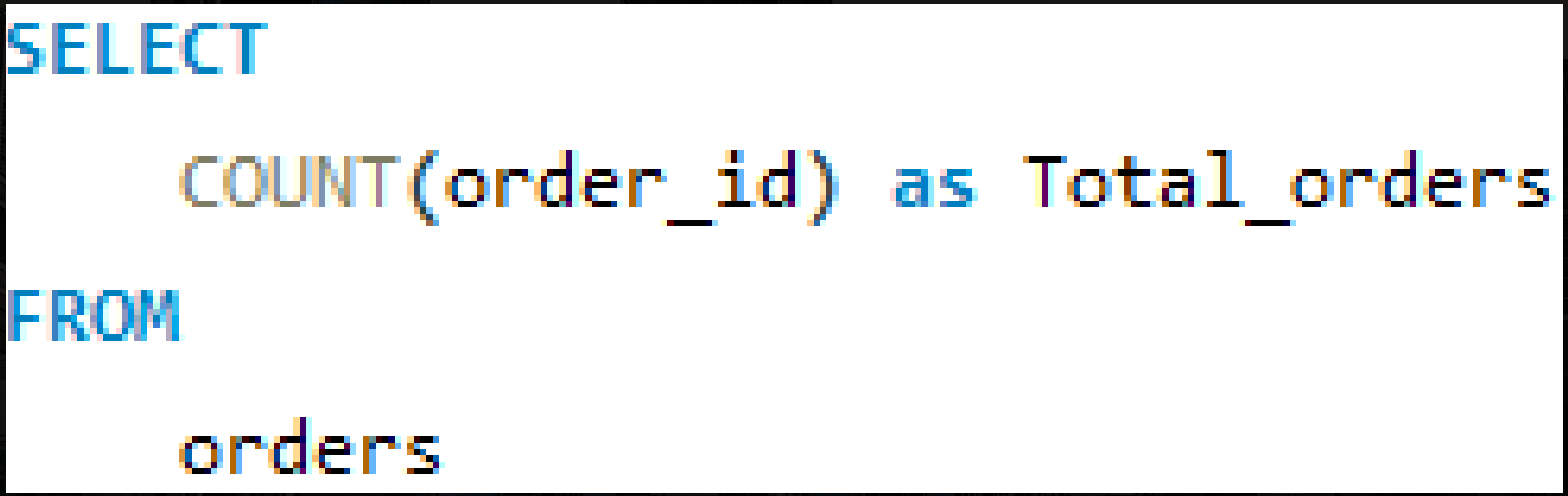
20
24

This dataset encompasses a comprehensive collection of pizza types, categorized by styles and ingredients. It includes detailed order information such as order IDs, customer details, quantities purchased, and pricing.
By analyzing this dataset, we can uncover valuable insights into customer preferences and purchasing behavior, identify trends in pizza consumption, and optimize inventory management. Additionally, this data can support the development of targeted marketing strategies to enhance customer satisfaction and drive sales growth.

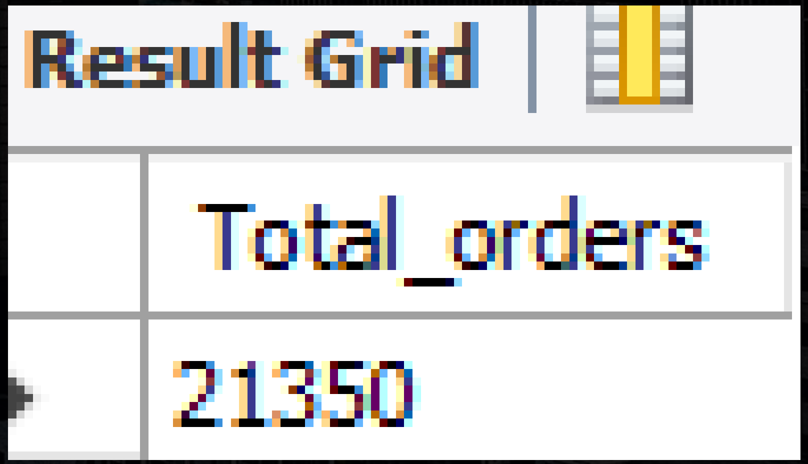


20
24

RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED

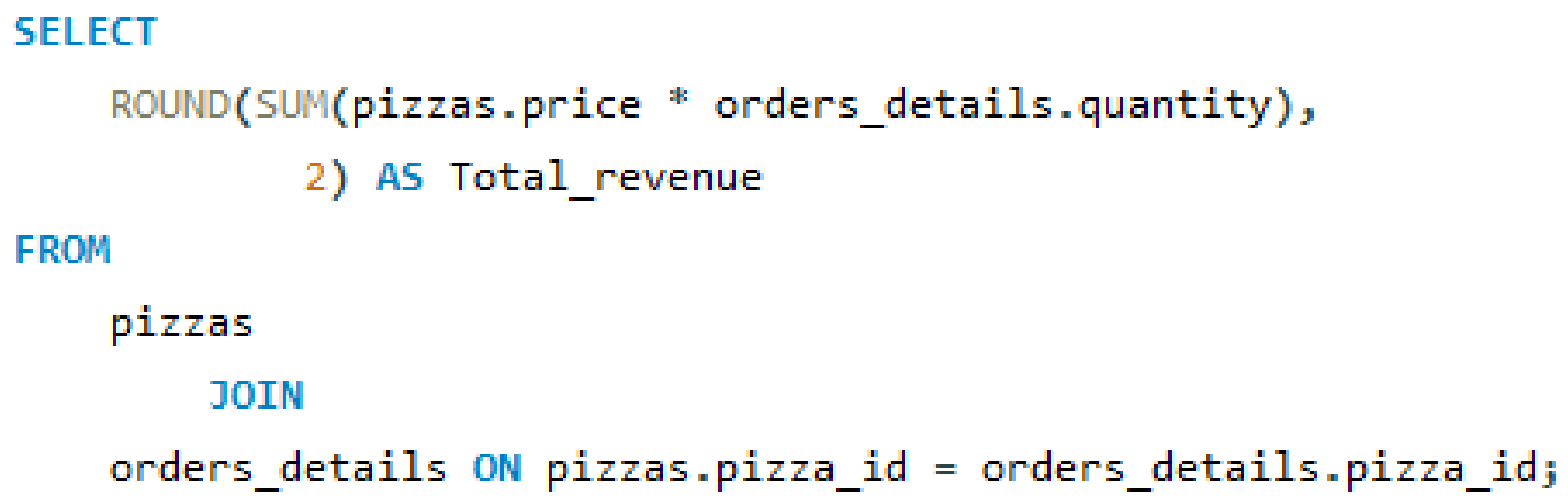


```
SELECT
    COUNT(order_id) as Total_orders
FROM
    orders
```

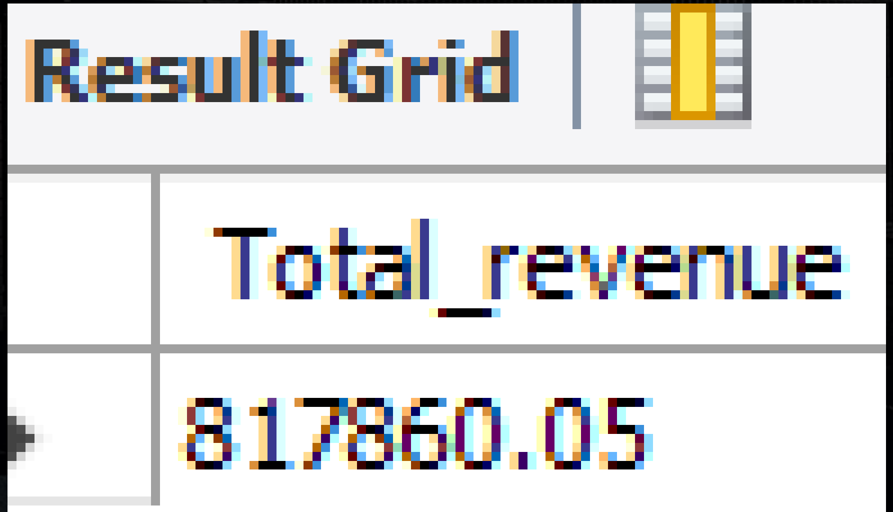


```
Total_orders
21350
```

CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.



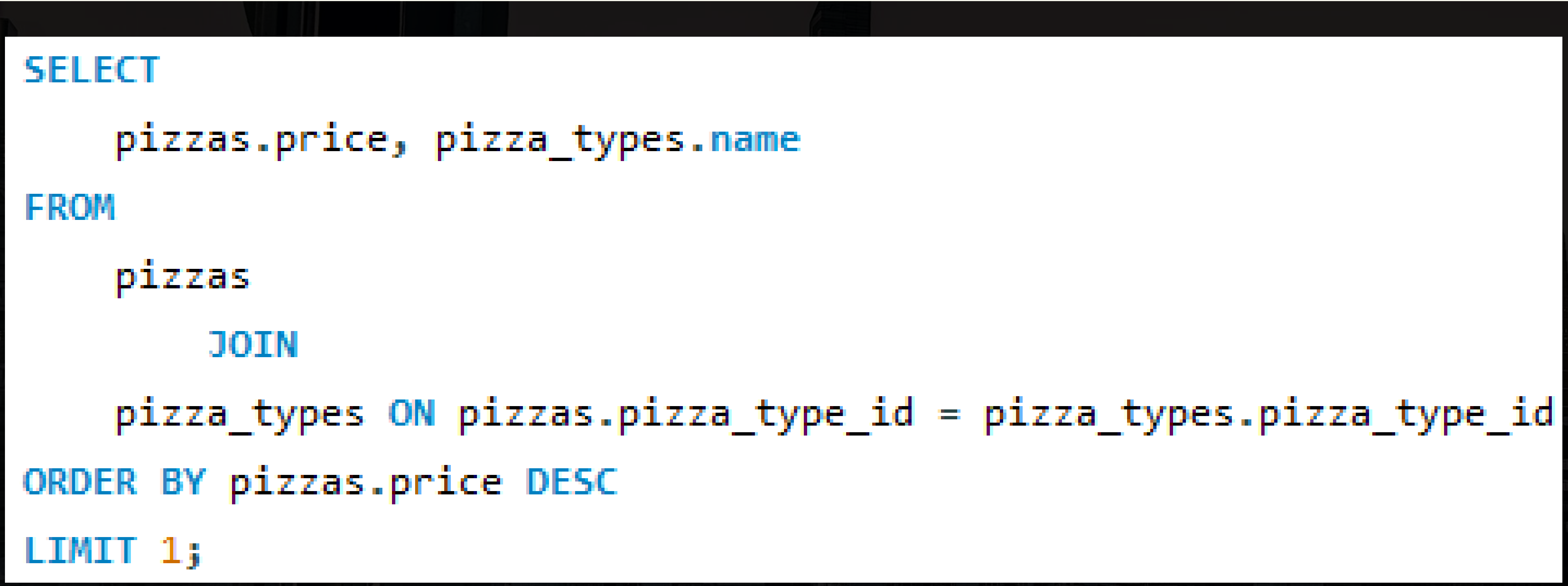
```sql
SELECT
    ROUND(SUM(pizzas.price * orders_details.quantity),
            2) AS Total_revenue
FROM
    pizzas
        JOIN
    orders_details ON pizzas.pizza_id = orders_details.pizza_id;
```



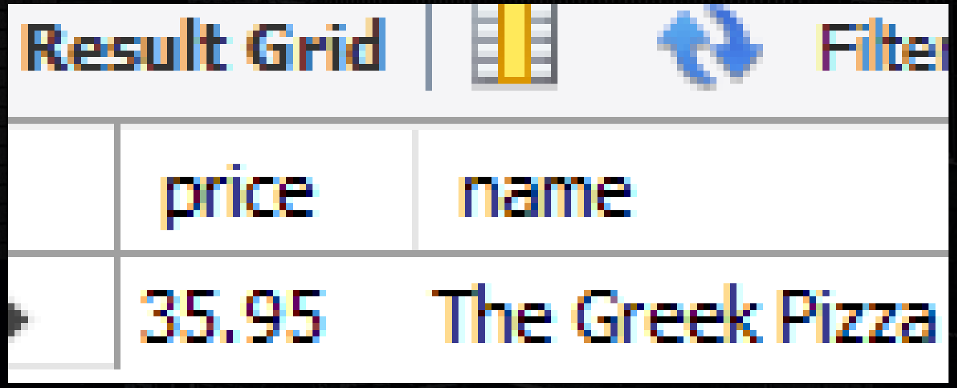
Total_revenue
817860.05

20
24

IDENTIFY THE HIGHEST-PRICED PIZZA

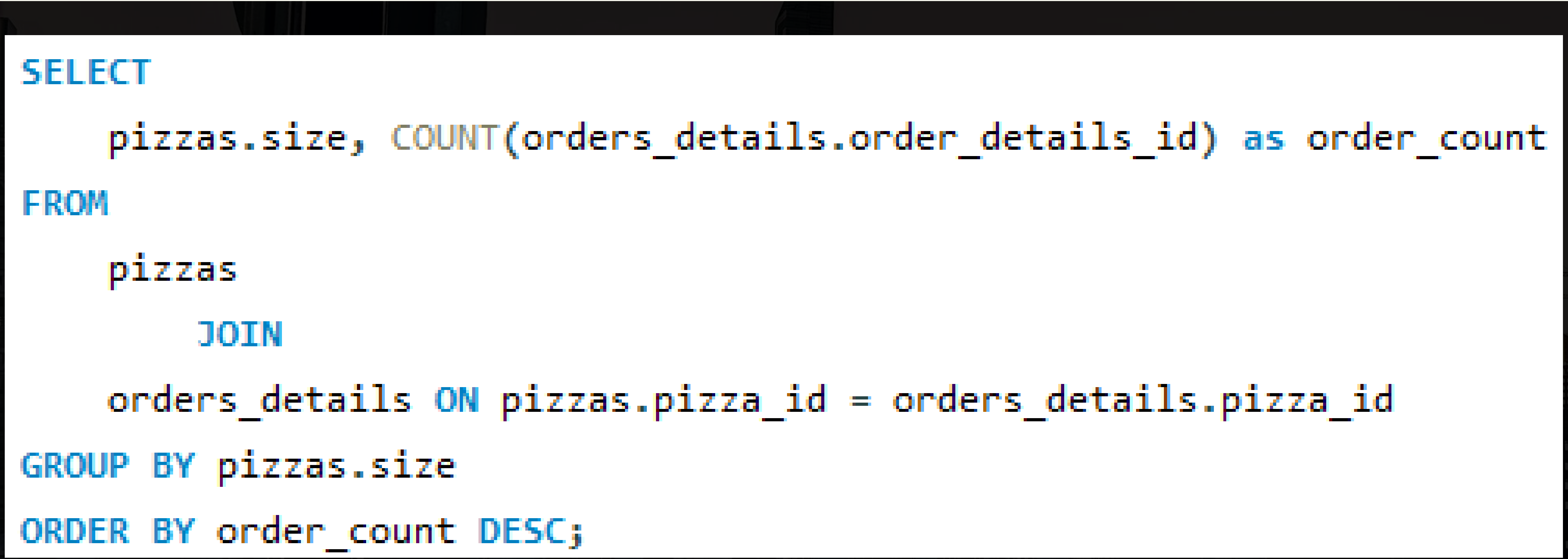


```sql
SELECT
    pizzas.price, pizza_types.name
FROM
    pizzas
        JOIN
    pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
ORDER BY pizzas.price DESC
LIMIT 1;
```

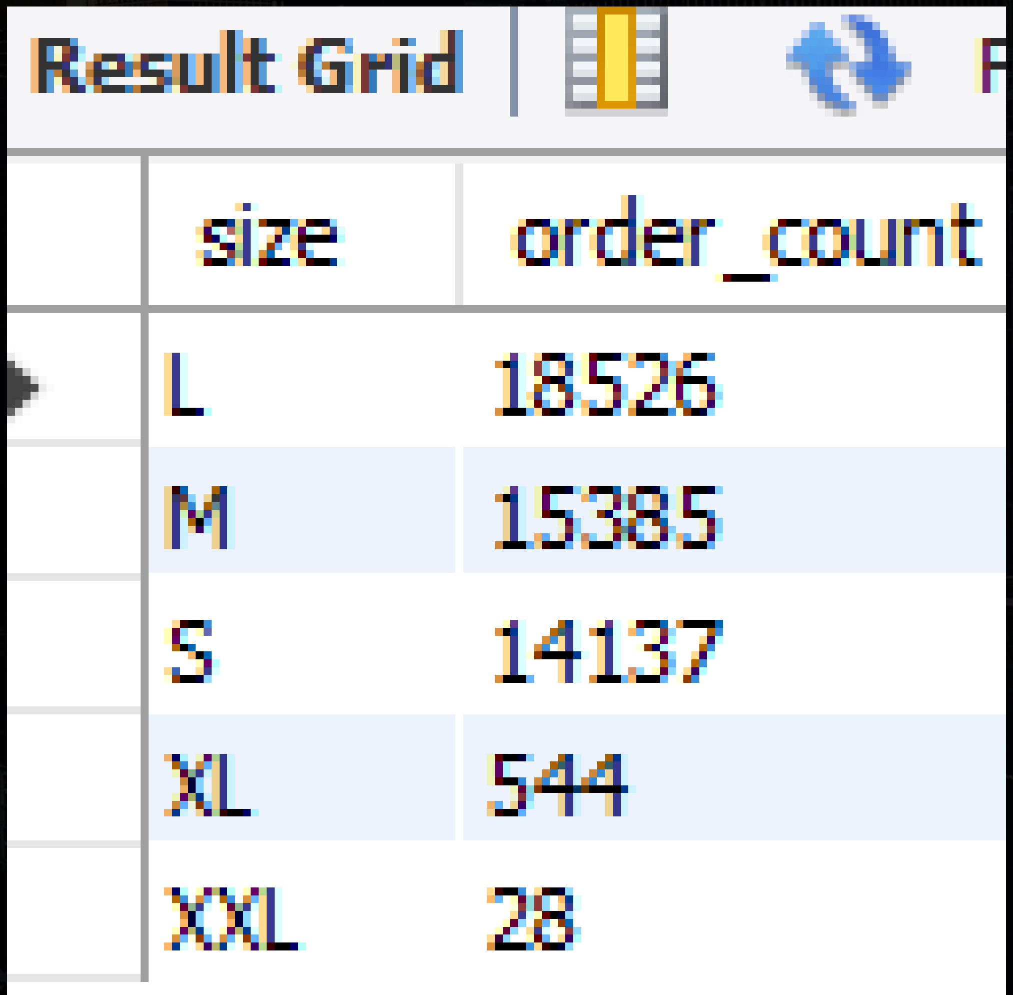


price	name
35.95	The Greek Pizza

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.



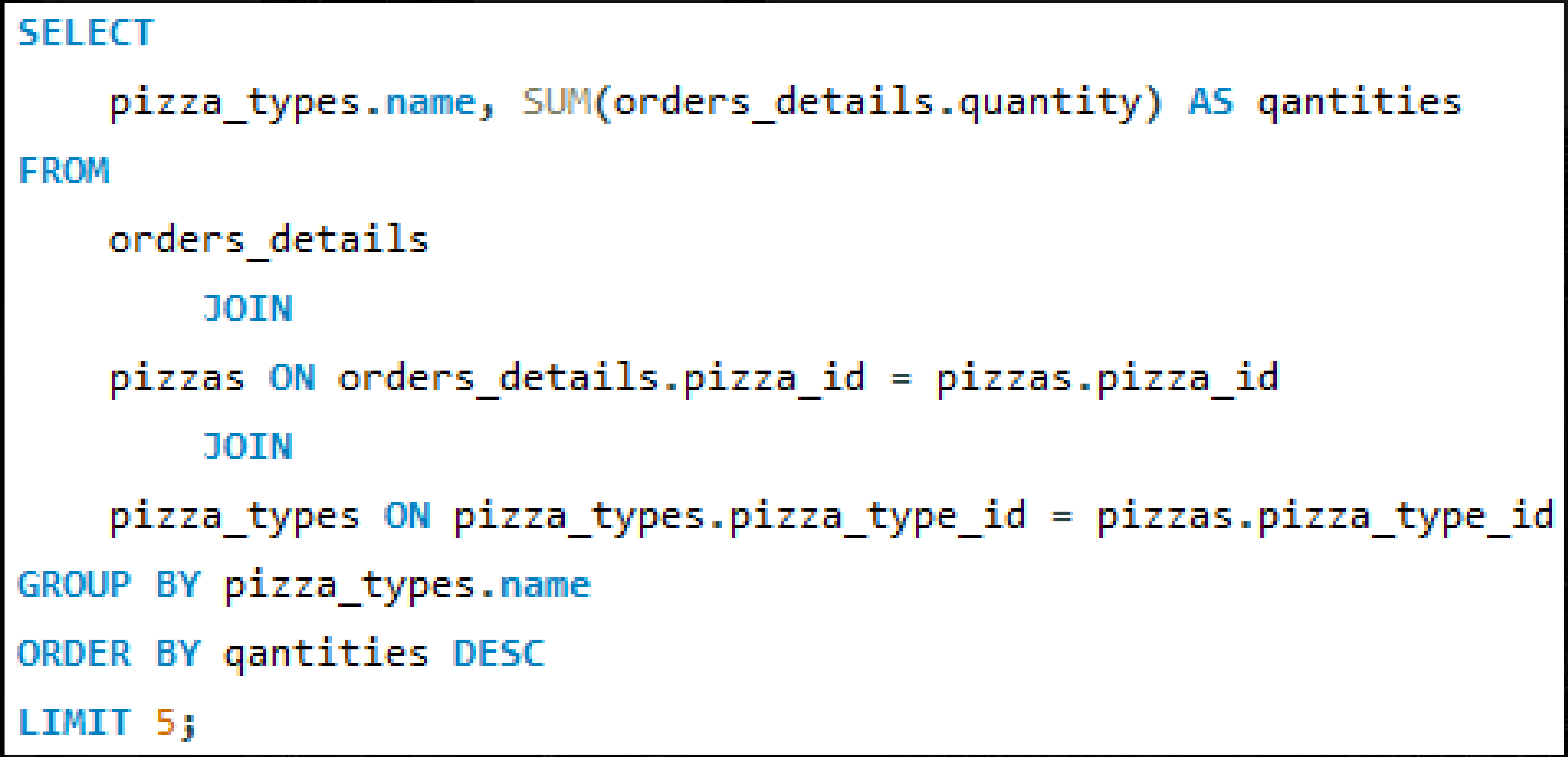
```
SELECT
    pizzas.size, COUNT(orders_details.order_details_id) as order_count
FROM
    pizzas
        JOIN
    orders_details ON pizzas.pizza_id = orders_details.pizza_id
GROUP BY pizzas.size
ORDER BY order_count DESC;
```



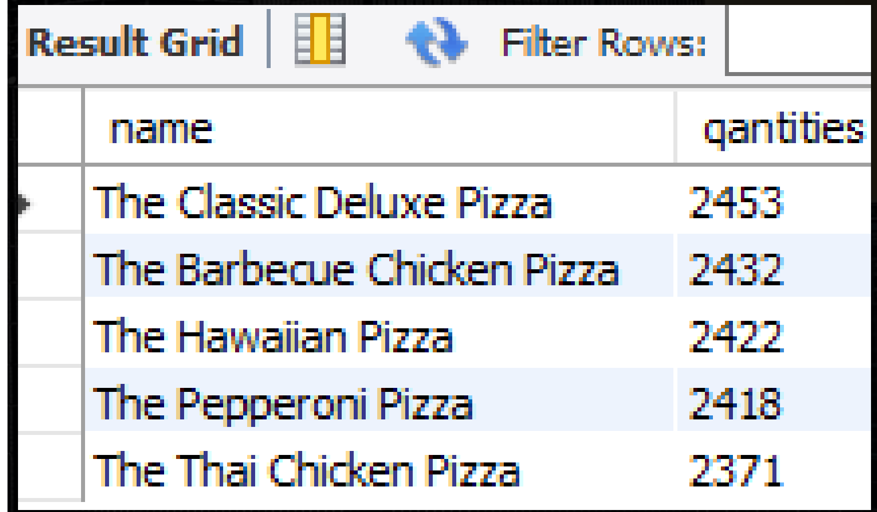
size	order_count
L	18526
M	15385
S	14137
XL	544
XXL	28

20
24

LIST THE TOP 5 MOST ORDERED PIZZA TYPES ALONG WITH THEIR QANTITIES.



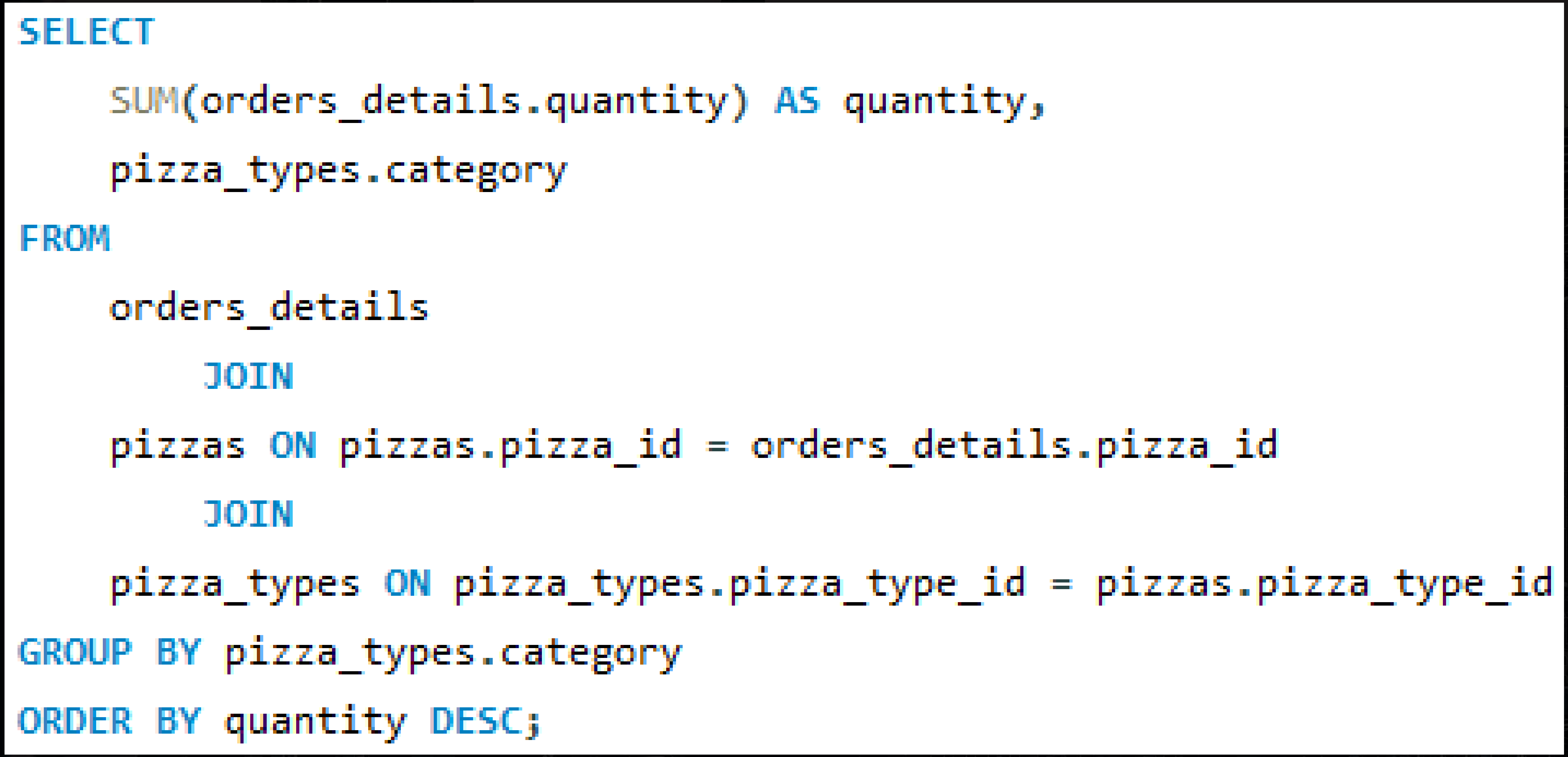
```
SELECT
    pizza_types.name, SUM(orders_details.quantity) AS qantities
FROM
    orders_details
        JOIN
    pizzas ON orders_details.pizza_id = pizzas.pizza_id
        JOIN
    pizza_types ON pizza_types.pizza_type_id = pizzas.pizza_type_id
GROUP BY pizza_types.name
ORDER BY qantities DESC
LIMIT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



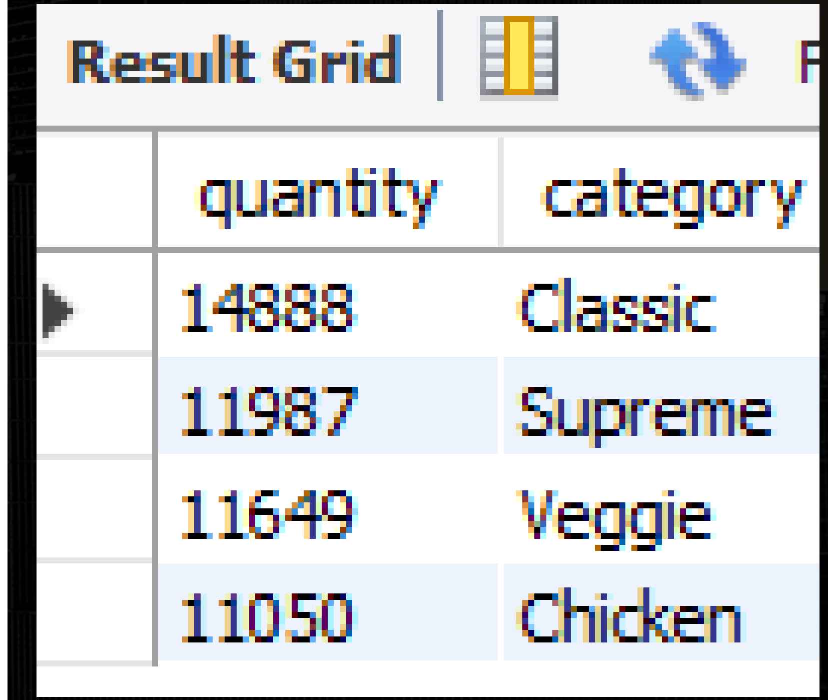
name	qantities
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	2422
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371

20
24

JOIN THE NECESSARY TABLES TO FIND THE TOTAL QUANTITY OF EACH PIZZA CATEGORY ORDERED.



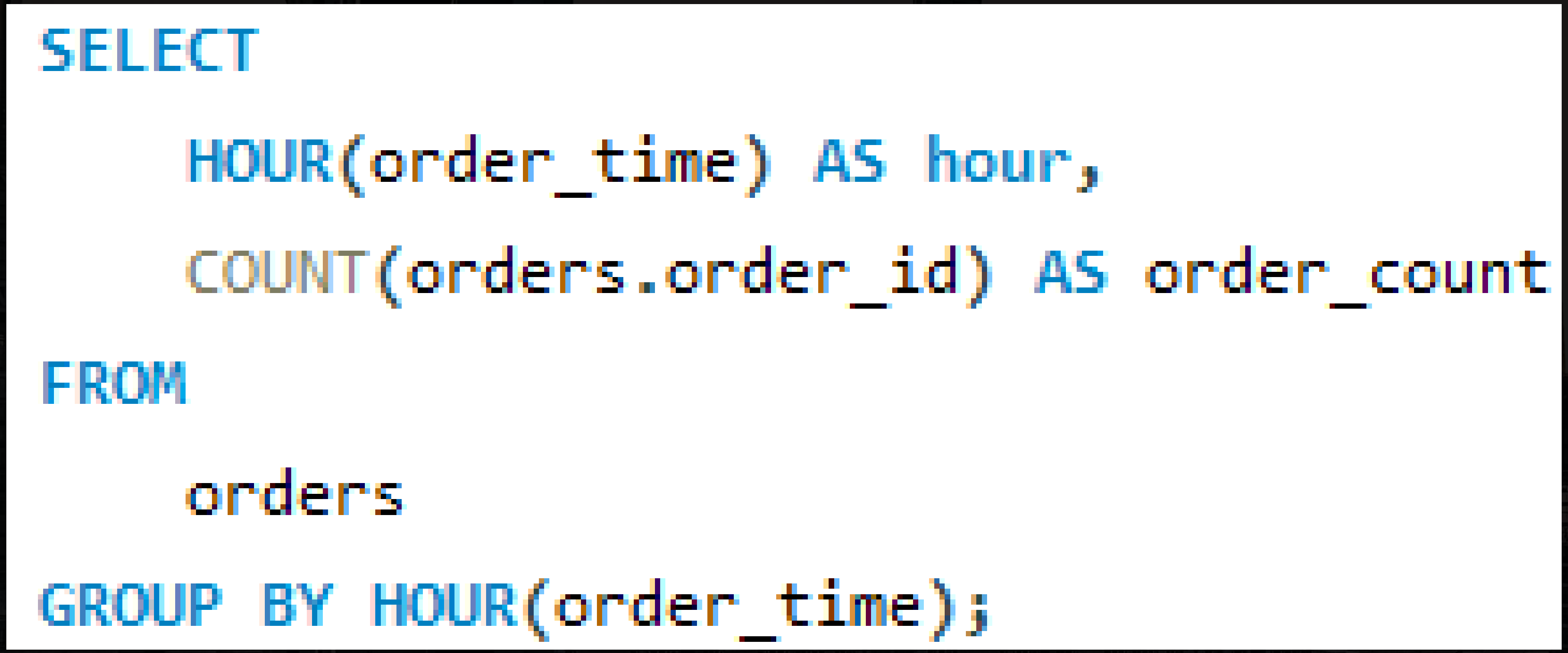
```
SELECT
    SUM(orders_details.quantity) AS quantity,
    pizza_types.category
FROM
    orders_details
        JOIN
    pizzas ON pizzas.pizza_id = orders_details.pizza_id
        JOIN
    pizza_types ON pizza_types.pizza_type_id = pizzas.pizza_type_id
GROUP BY pizza_types.category
ORDER BY quantity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



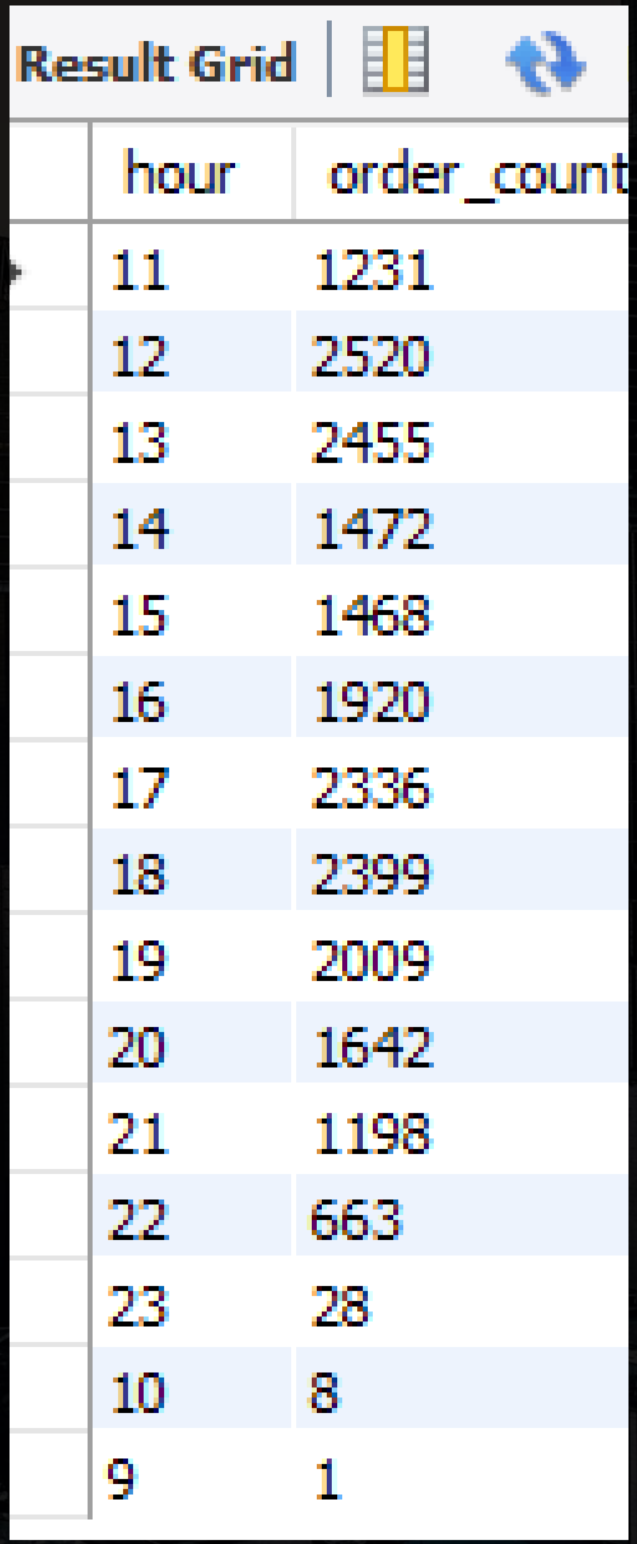
quantity	category
14888	Classic
11987	Supreme
11649	Veggie
11050	Chicken

20
24

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.



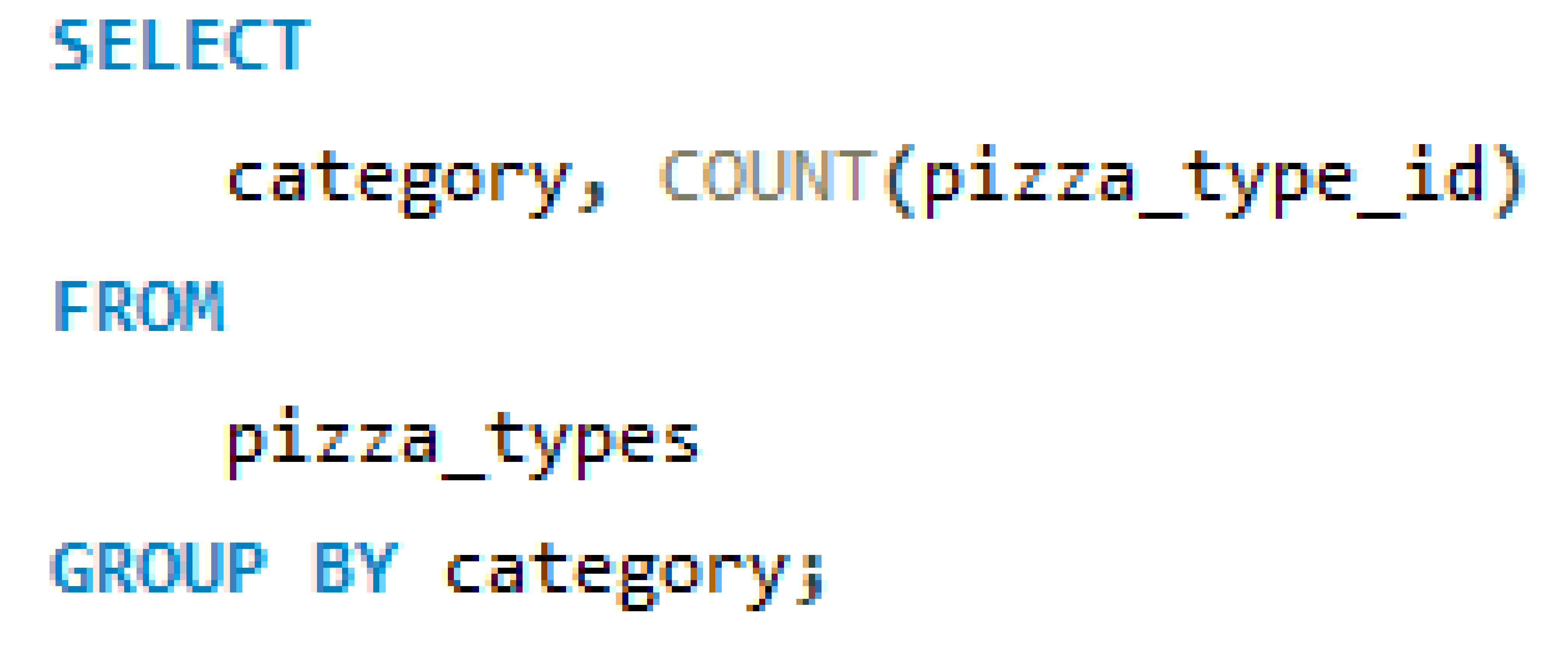
```
SELECT
    HOUR(order_time) AS hour,
    COUNT(orders.order_id) AS order_count
FROM
    orders
GROUP BY HOUR(order_time);
```



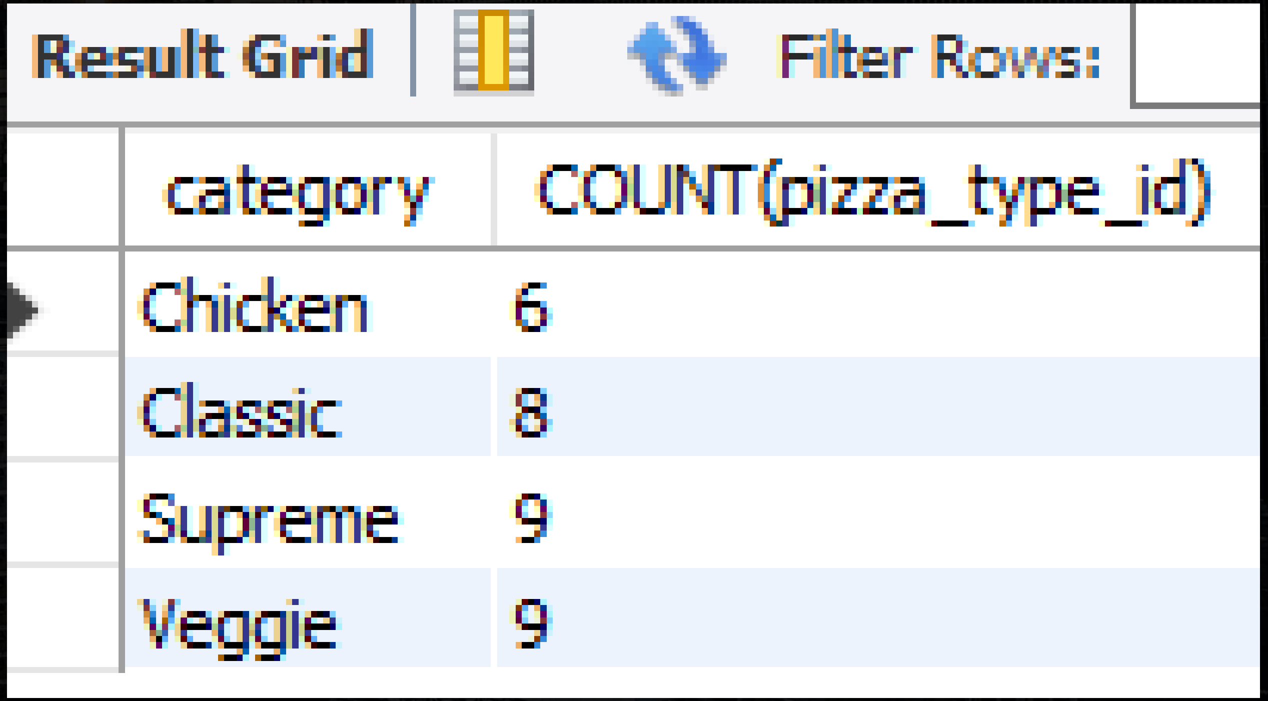
hour	order_count
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642
21	1198
22	663
23	28
10	8
9	1

20
24

JOIN RELEVANT TABLES TO FIND THE CATEGORY WISE DISTRIBUTION OF PIZZAS.



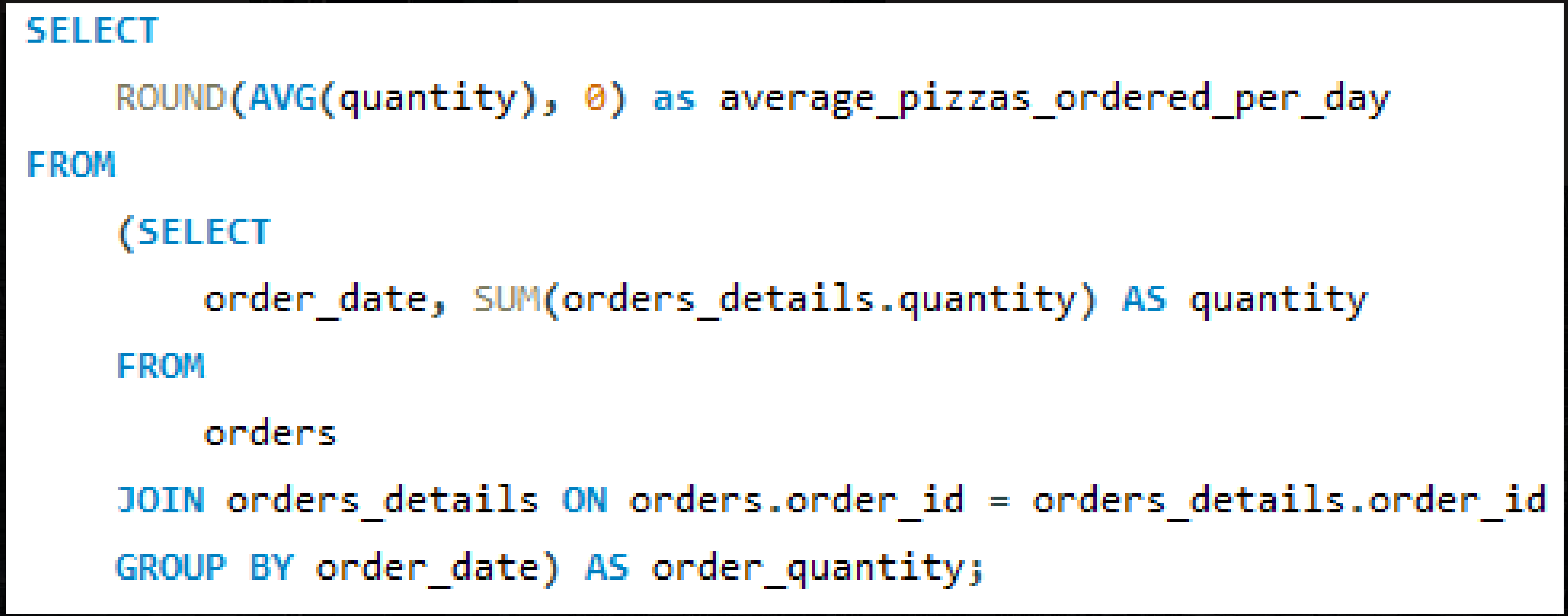
```
SELECT
    category, COUNT(pizza_type_id)
FROM
    pizza_types
GROUP BY category;
```



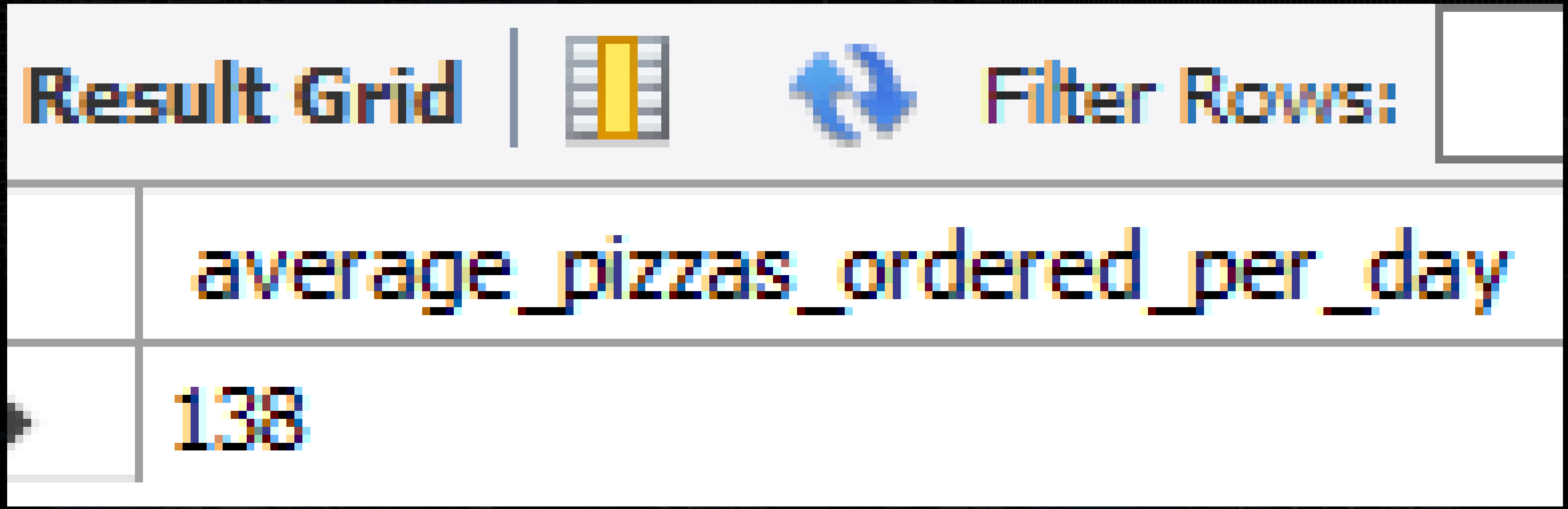
category	COUNT(pizza_type_id)
Chicken	6
Classic	8
Supreme	9
Veggie	9

20
24

GROUP THE ORDERS BY DATE AND CALCULATE THE AVERAGE NUMBER OF PIZZAS ORDERED PER DAY.



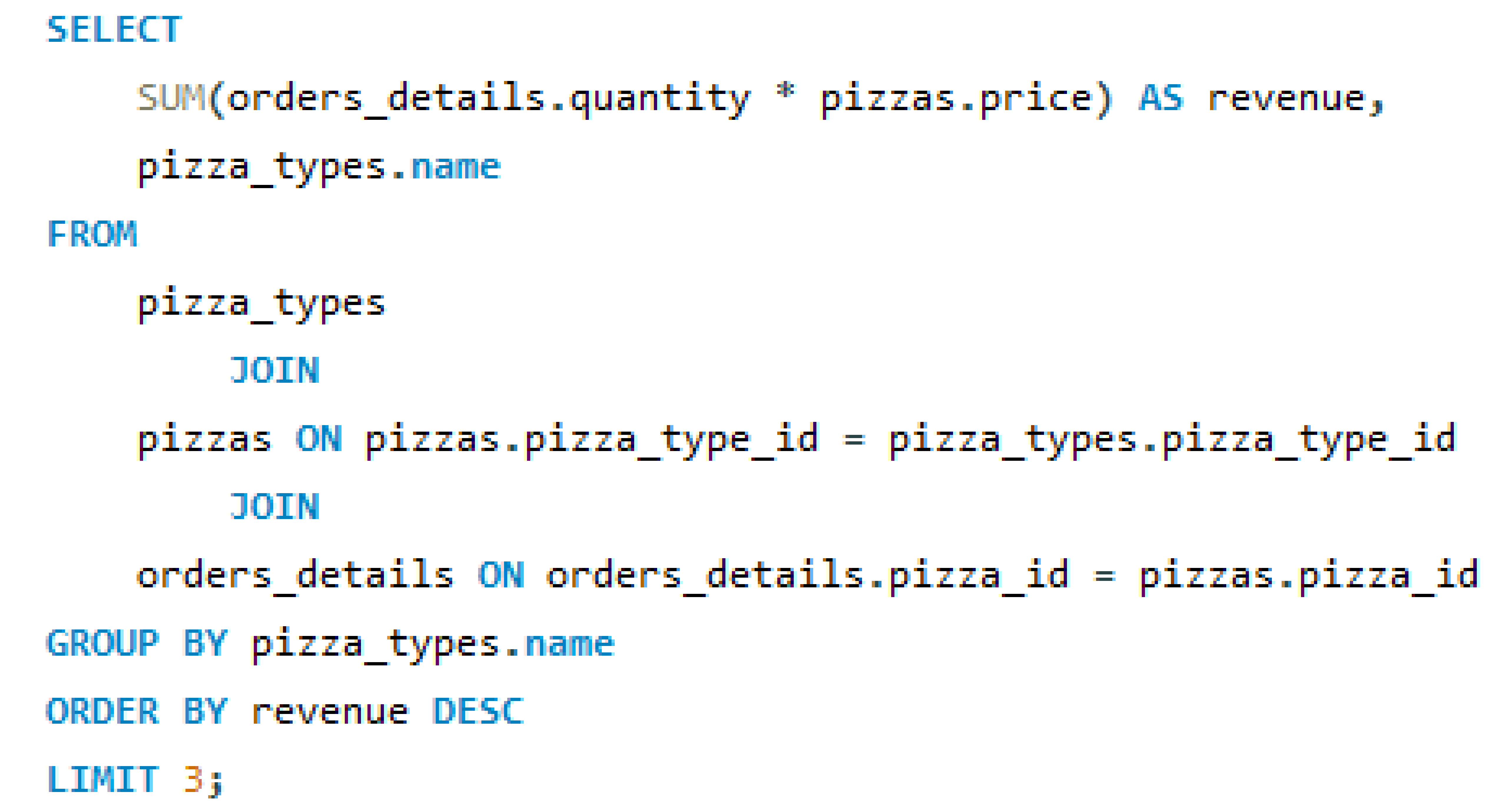
```
SELECT
    ROUND(AVG(quantity), 0) as average_pizzas_ordered_per_day
FROM
    (SELECT
        order_date, SUM(orders_details.quantity) AS quantity
    FROM
        orders
    JOIN orders_details ON orders.order_id = orders_details.order_id
    GROUP BY order_date) AS order_quantity;
```



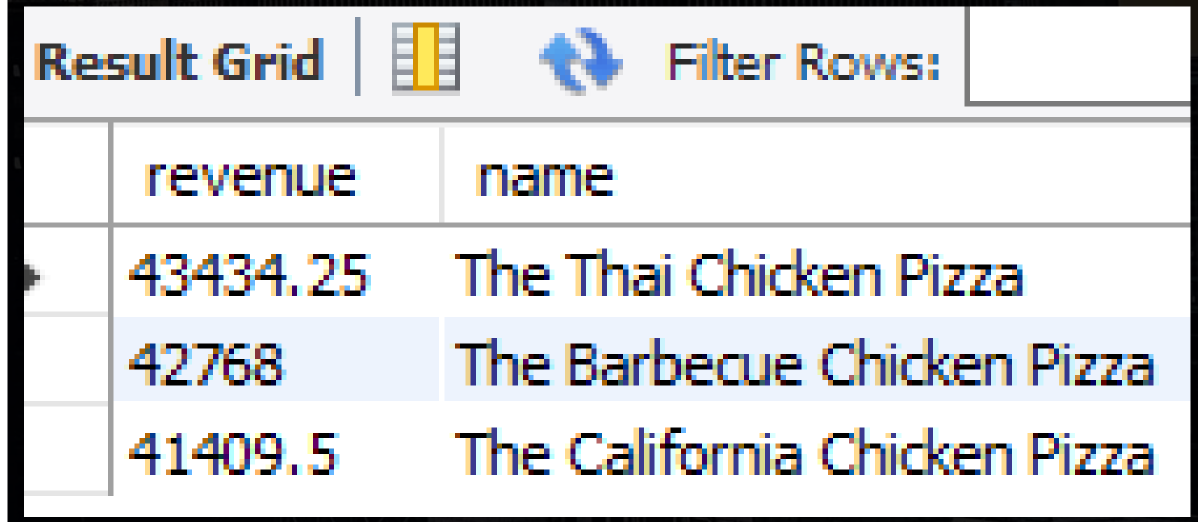
average_pizzas_ordered_per_day
138

20
24

DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE



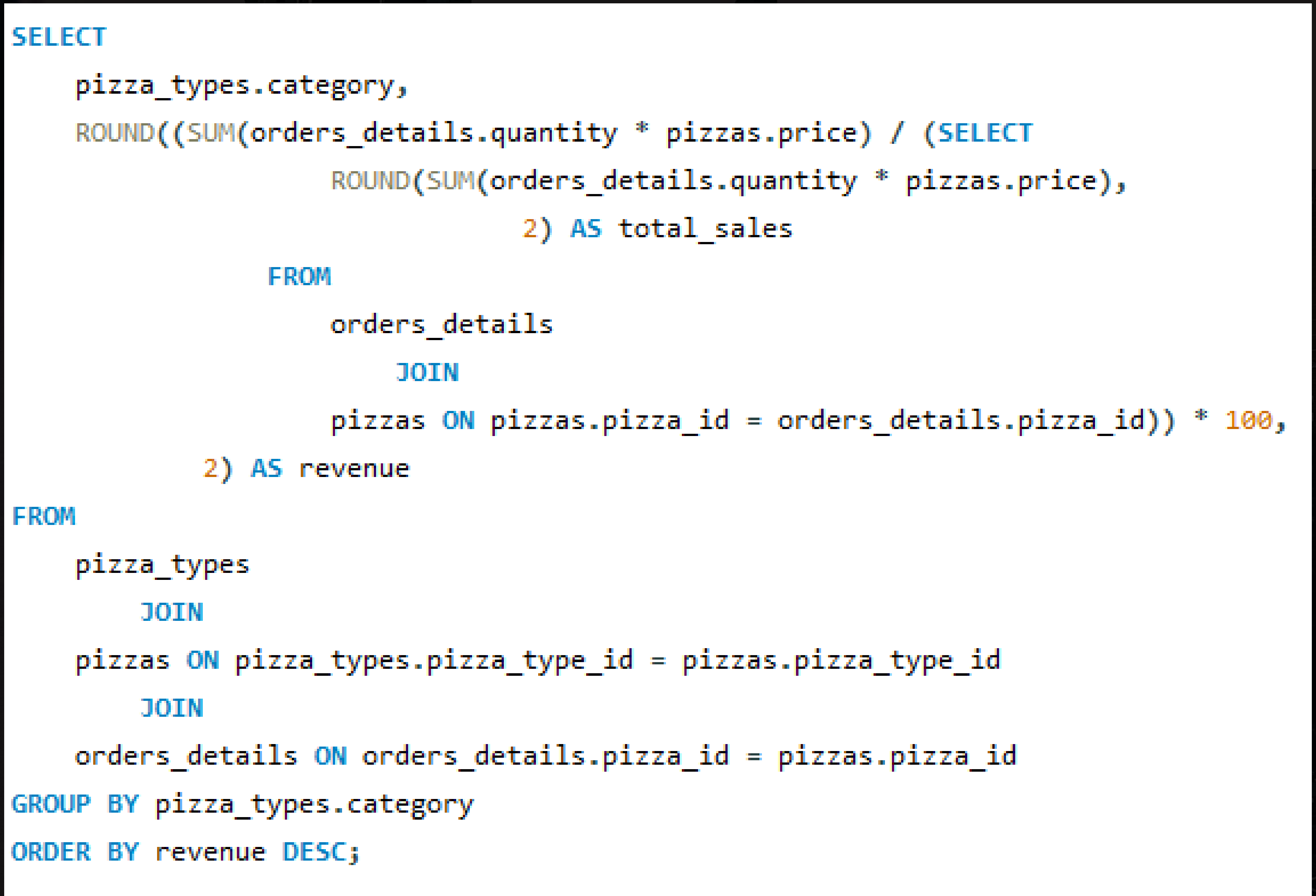
```
SELECT
    SUM(orders_details.quantity * pizzas.price) AS revenue,
    pizza_types.name
FROM
    pizza_types
        JOIN
    pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
        JOIN
    orders_details ON orders_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY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



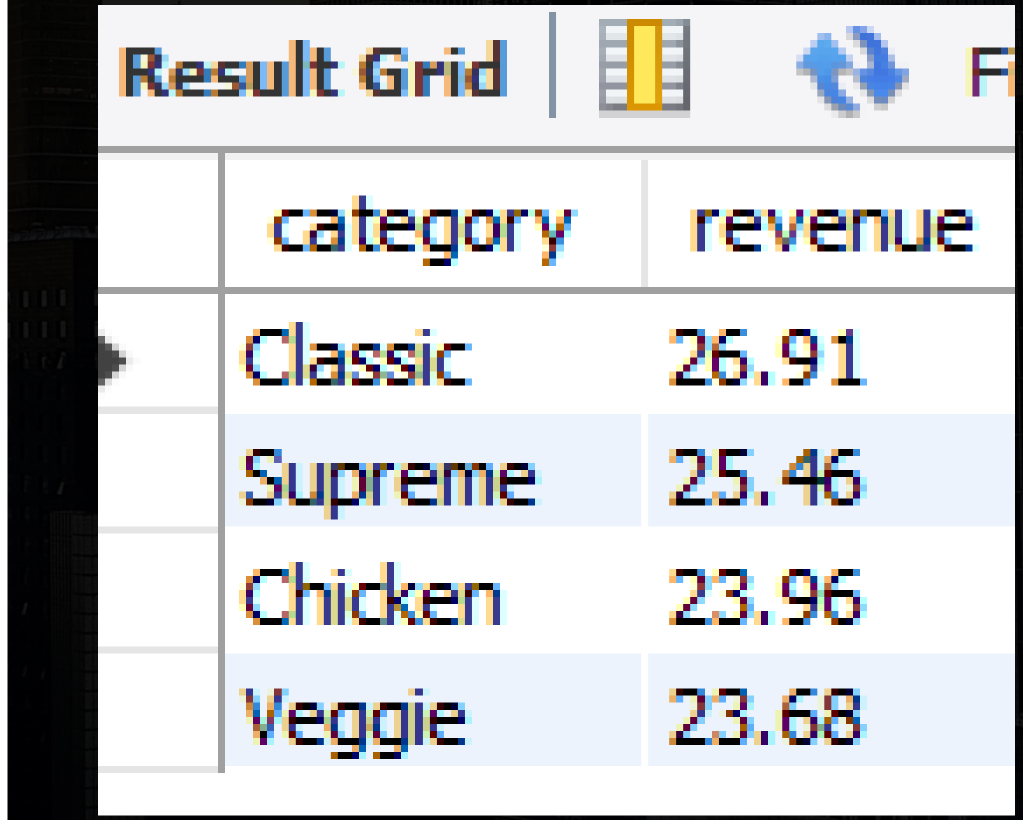
```
revenue     name
43434.25    The Thai Chicken Pizza
42768       The Barbecue Chicken Pizza
41409.5     The California Chicken Pizza
```

20
24

CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPE TO TOTAL REVENUE.



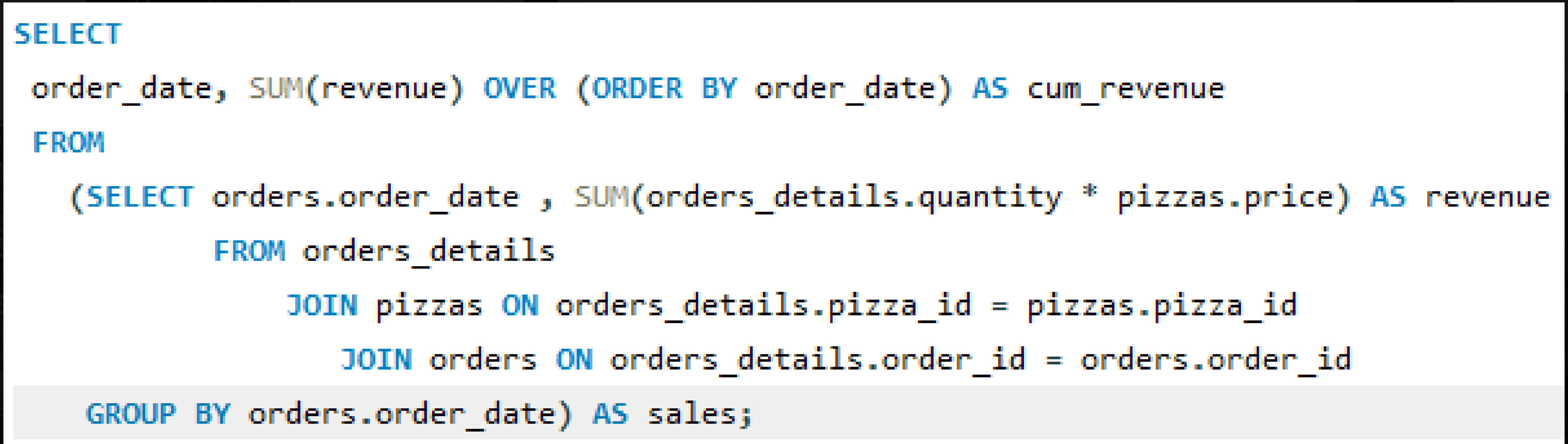
```
SELECT
    pizza_types.category,
    ROUND((SUM(orders_details.quantity * pizzas.price) / (SELECT
                    ROUND(SUM(orders_details.quantity * pizzas.price),
                                2) AS total_sales
                FROM
                    orders_details
                     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



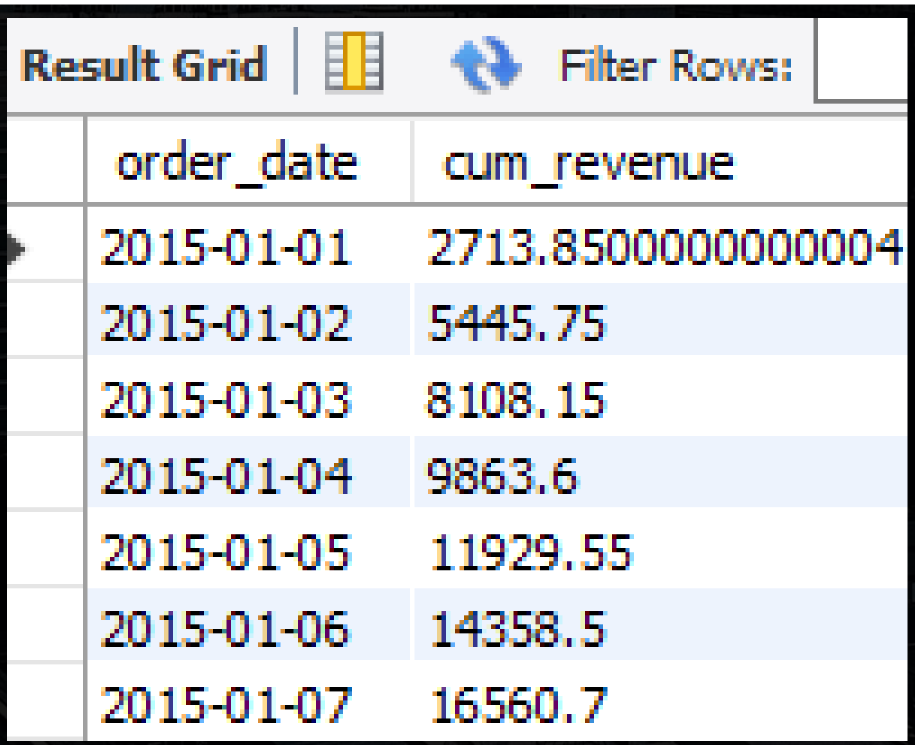
category	revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

20
24

ANALYZE THE CUMULATIVE REVENUE GENRATED OVER TIME

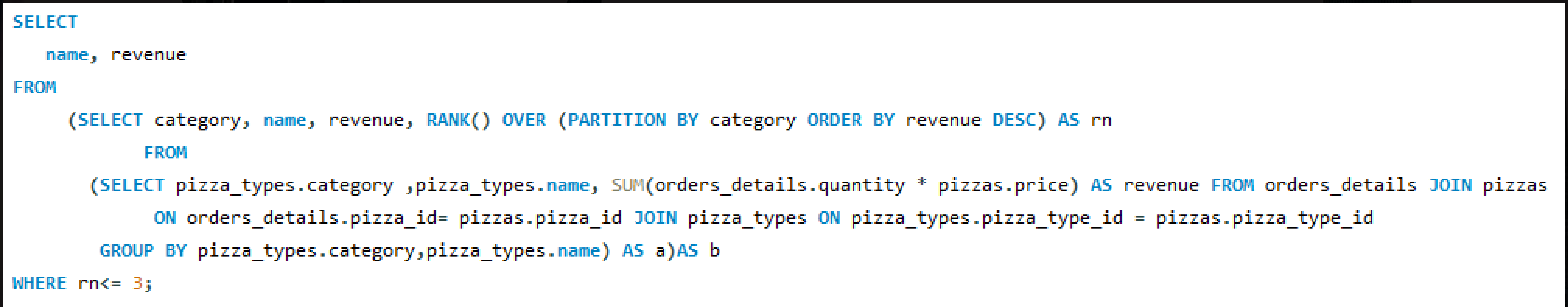


```sql
SELECT
 order_date, SUM(revenue) OVER (ORDER BY order_date) AS cum_revenue
 FROM
   (SELECT orders.order_date , SUM(orders_details.quantity * pizzas.price) AS revenue
           FROM orders_details
               JOIN pizzas ON orders_details.pizza_id = pizzas.pizza_id
                  JOIN orders ON orders_details.order_id = orders.order_id
    GROUP BY orders.order_date) AS sales;
```

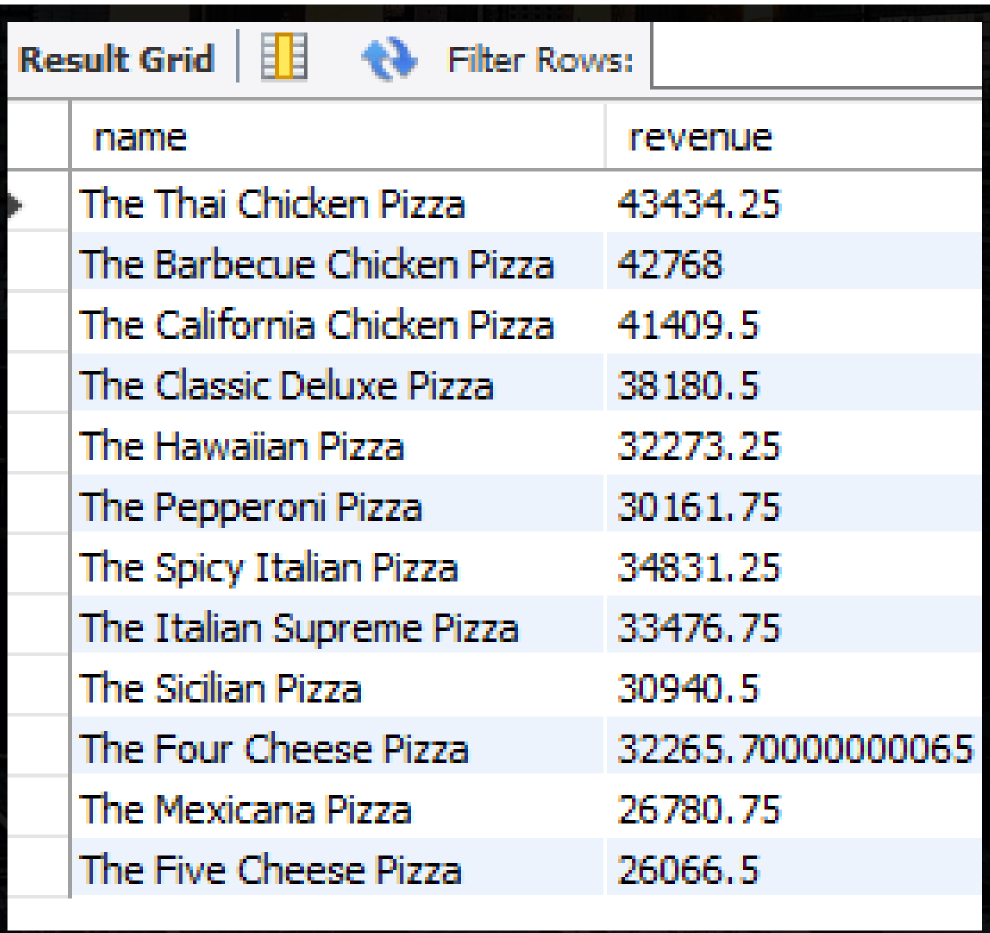


order_date	cum_revenue
2015-01-01	2713.8500000000004
2015-01-02	5445.75
2015-01-03	8108.15
2015-01-04	9863.6
2015-01-05	11929.55
2015-01-06	14358.5
2015-01-07	16560.7

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 orders_details JOIN pizzas
             ON orders_details.pizza_id= pizzas.pizza_id JOIN pizza_types ON pizza_types.pizza_type_id = pizzas.pizza_type_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

20
24

ACTION PLAN

Based on our analysis, we gained valuable insights into customer preferences regarding pizza types, revealing specific trends in popular styles and toppings. Additionally, we identified peak demand times across different time zones, allowing us to understand when customer orders are highest. Our examination of order demand per hour provides a clear picture of sales fluctuations throughout the day. This information not only highlights optimal times for promotions and marketing campaigns but also aids in efficient staffing and inventory management.

By leveraging these insights, businesses can tailor their offerings to align with customer desires and enhance overall operational efficiency, ultimately driving sales and improving customer satisfaction.

THANK YOU!