PIZZA SALES DATA ANALYSIS PROJECT

USING MYSQL

BY: TANUSH NIMBAWAT

Introduction:

This project focuses on enhancing my data analytics skills by analyzing pizza sales data using MySQL. By working with a dataset comprising orders, pizzas,pizza types, and order details.

I aim to extract meaningful insights and demonstrate proficiency in SQL.The analysis covers key metrics such as total orders, revenue, popular products, customer preferences, and sales trends. The ultimate goal is to showcase my ability to handle real-world data, perform complex queries, and provide actionable business.

Objective:

The objective of this project is to develop my data analytics skills using MySQL by analyzing pizza sales data. Key goals include:

Enhance SQL Proficiency

Perform Data Analysis

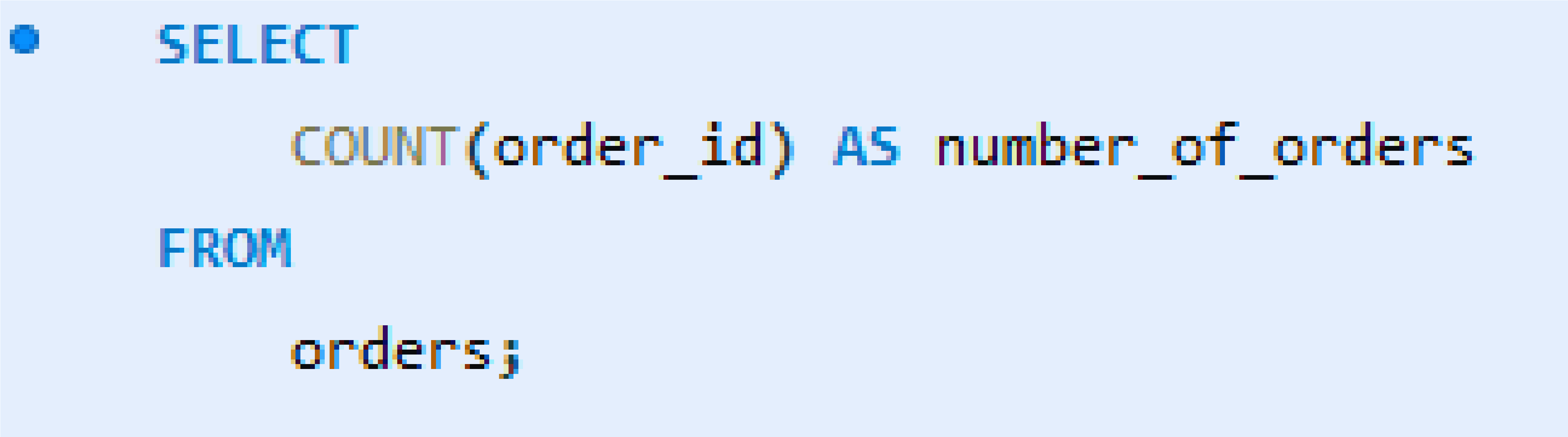
Visualize Data

Improve Decision-Making

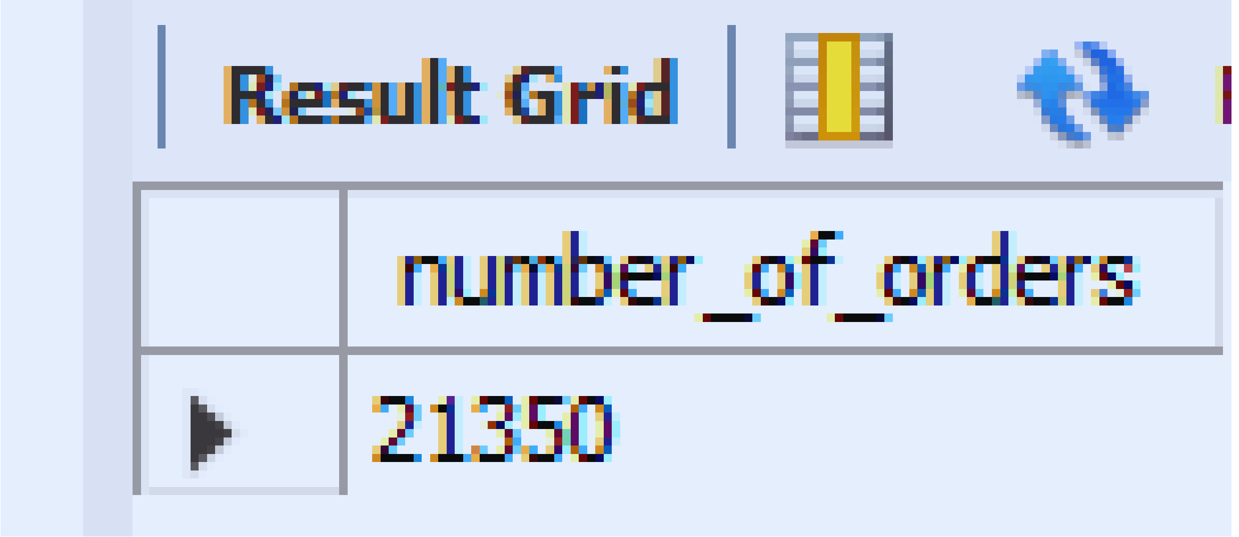
Problem 1

Retrieve the total number of orders placed.

Solution:



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

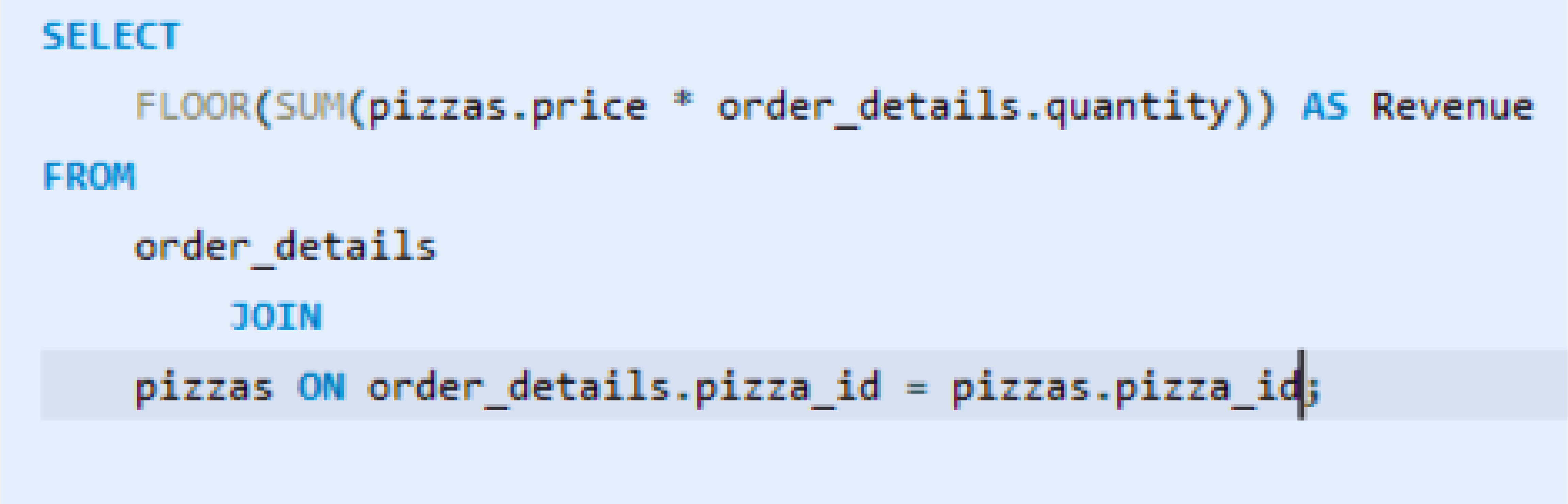


number_of_orders
21350

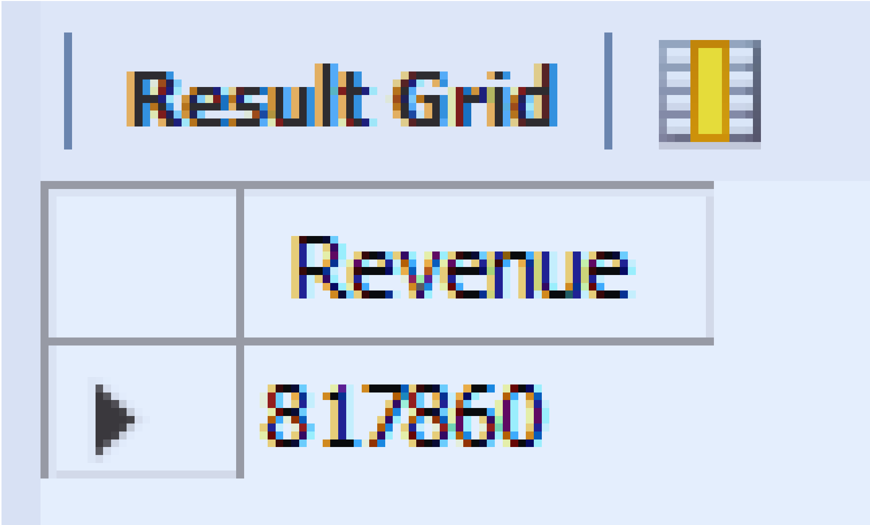
Problem 2

Calculate the total revenue generated from pizza sales.

Solution:



```sql
SELECT
    FLOOR(SUM(pizzas.price * order_details.quantity)) AS Revenue
FROM
    order_details
        JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id;
```

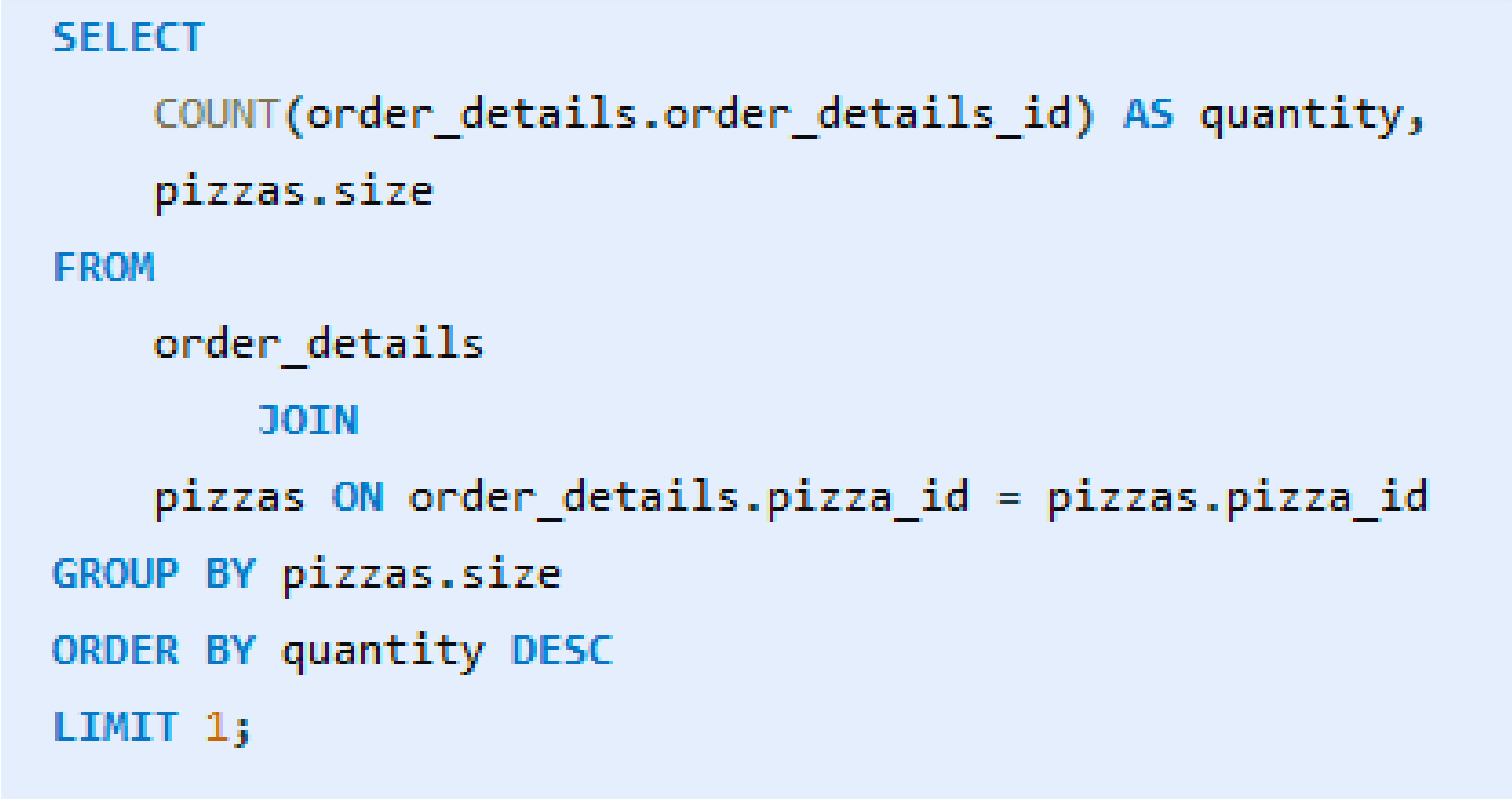


```
   Revenue
▶  817860
```

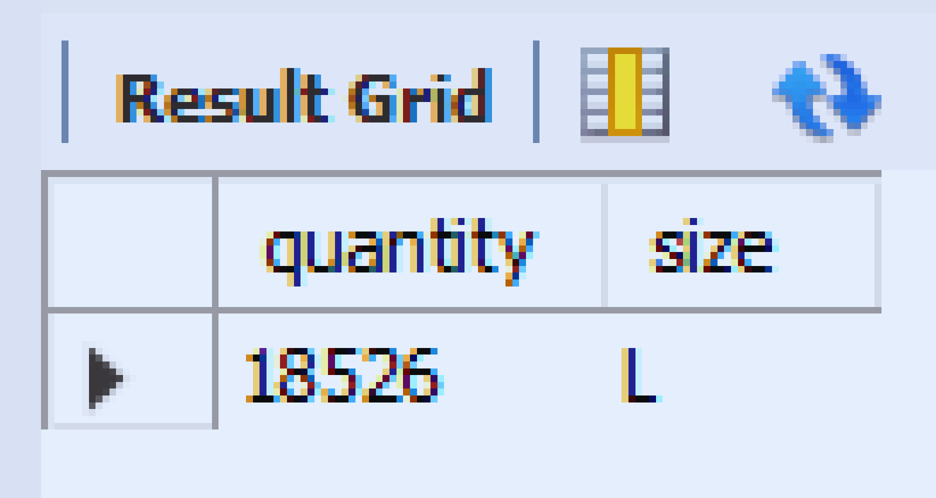
Problem 4

Identify the most common pizza size ordered.

Solution:



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

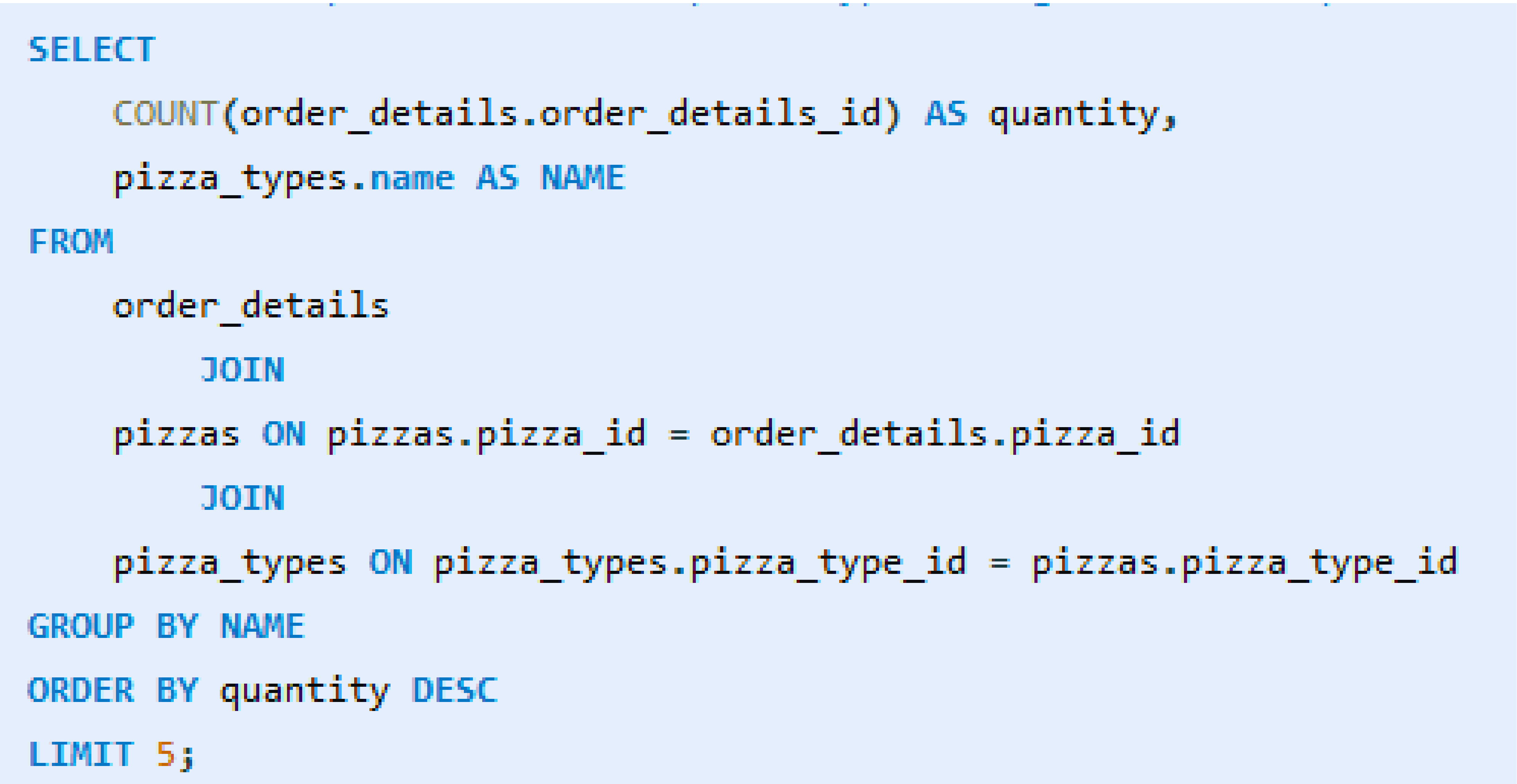


	quantity	size
▶	18526	L

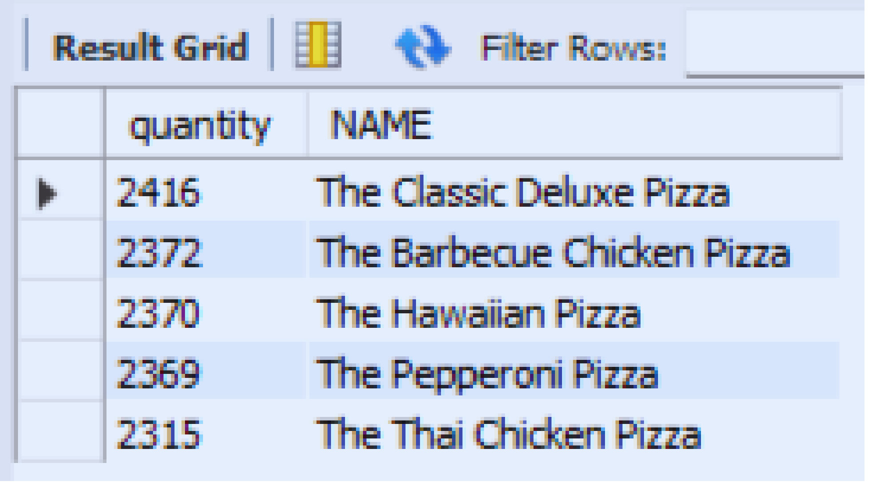
Problem 5

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

Solution:



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

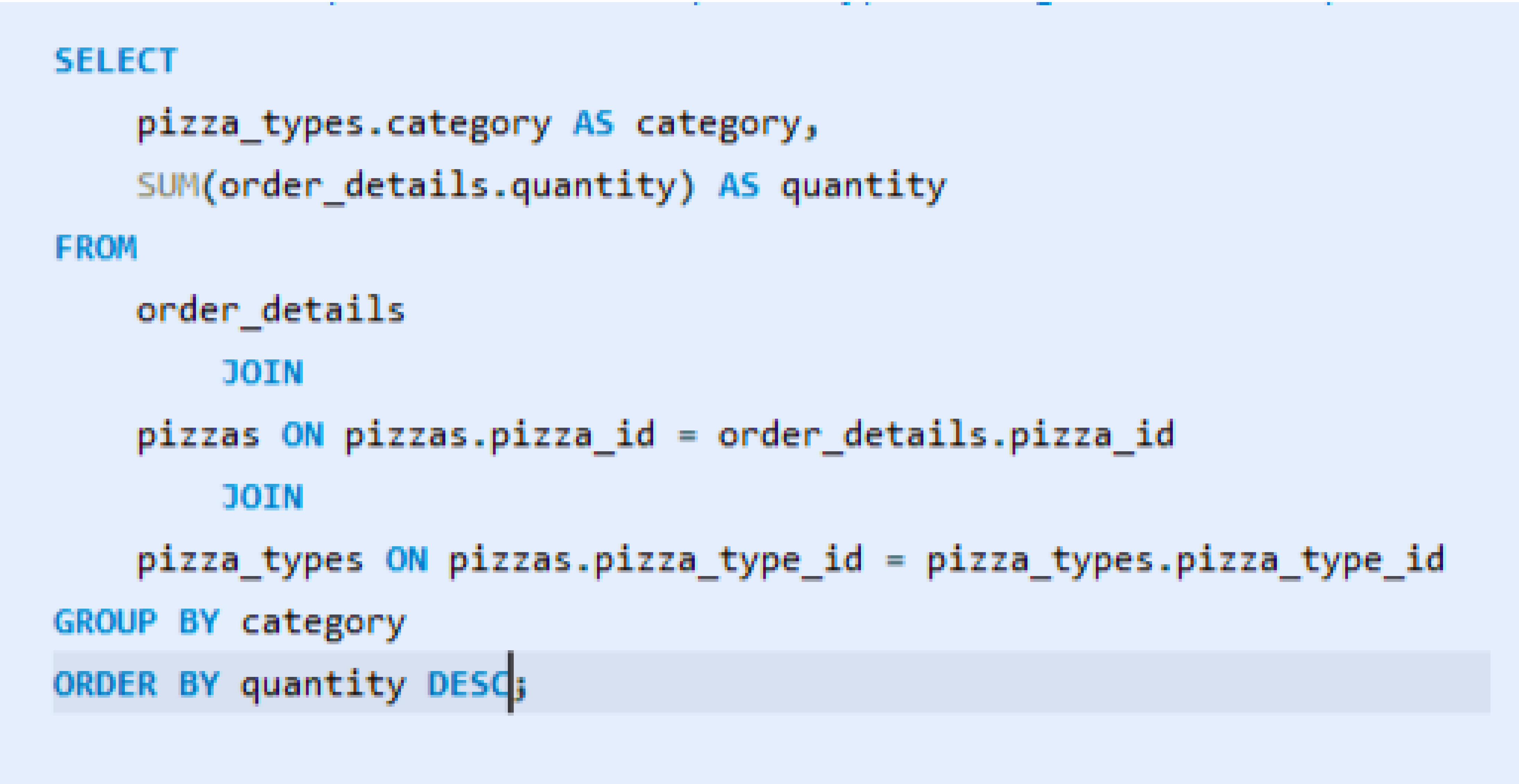


quantity	NAME
2416	The Classic Deluxe Pizza
2372	The Barbecue Chicken Pizza
2370	The Hawaiian Pizza
2369	The Pepperoni Pizza
2315	The Thai Chicken Pizza

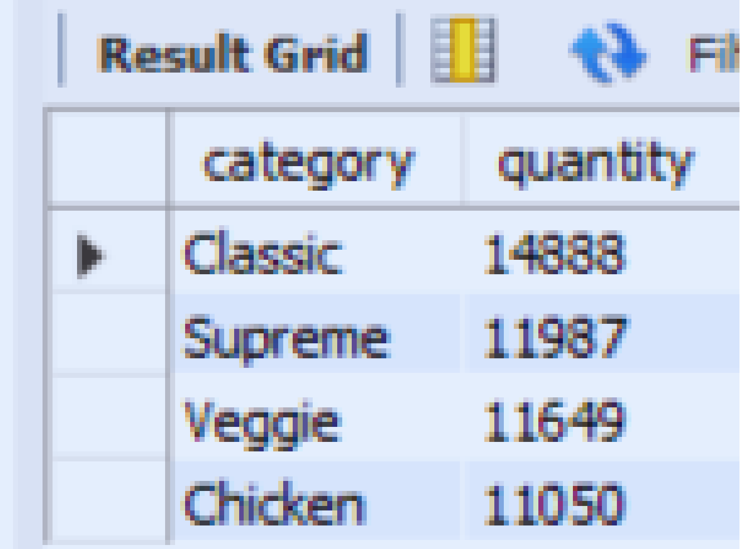
Problem 6

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

Solution:



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

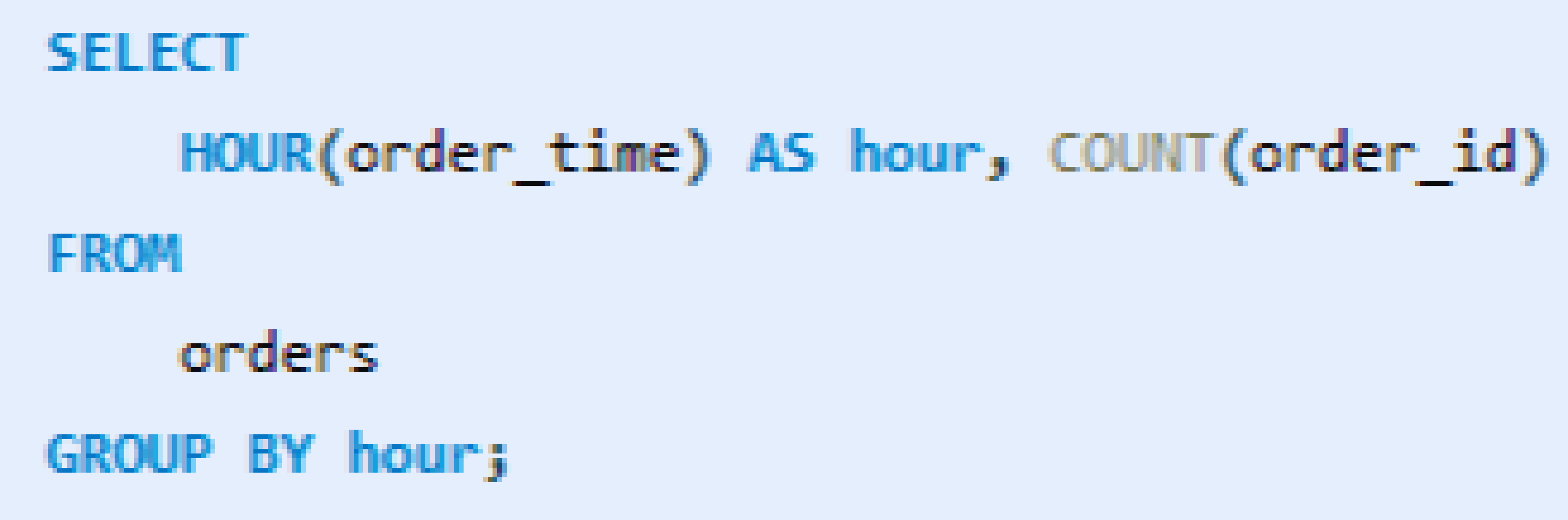


	category	quantity
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050

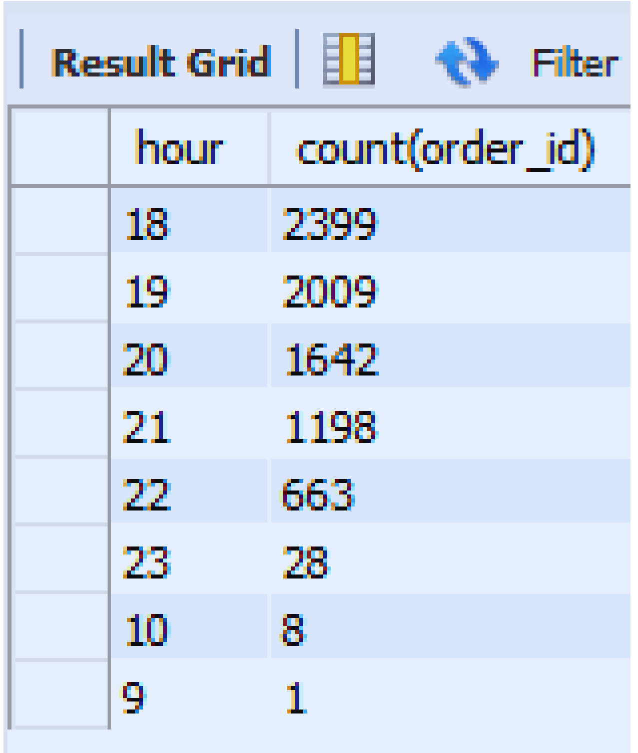
Problem 7

Determine the distribution of orders by hour of the day.

Solution:



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

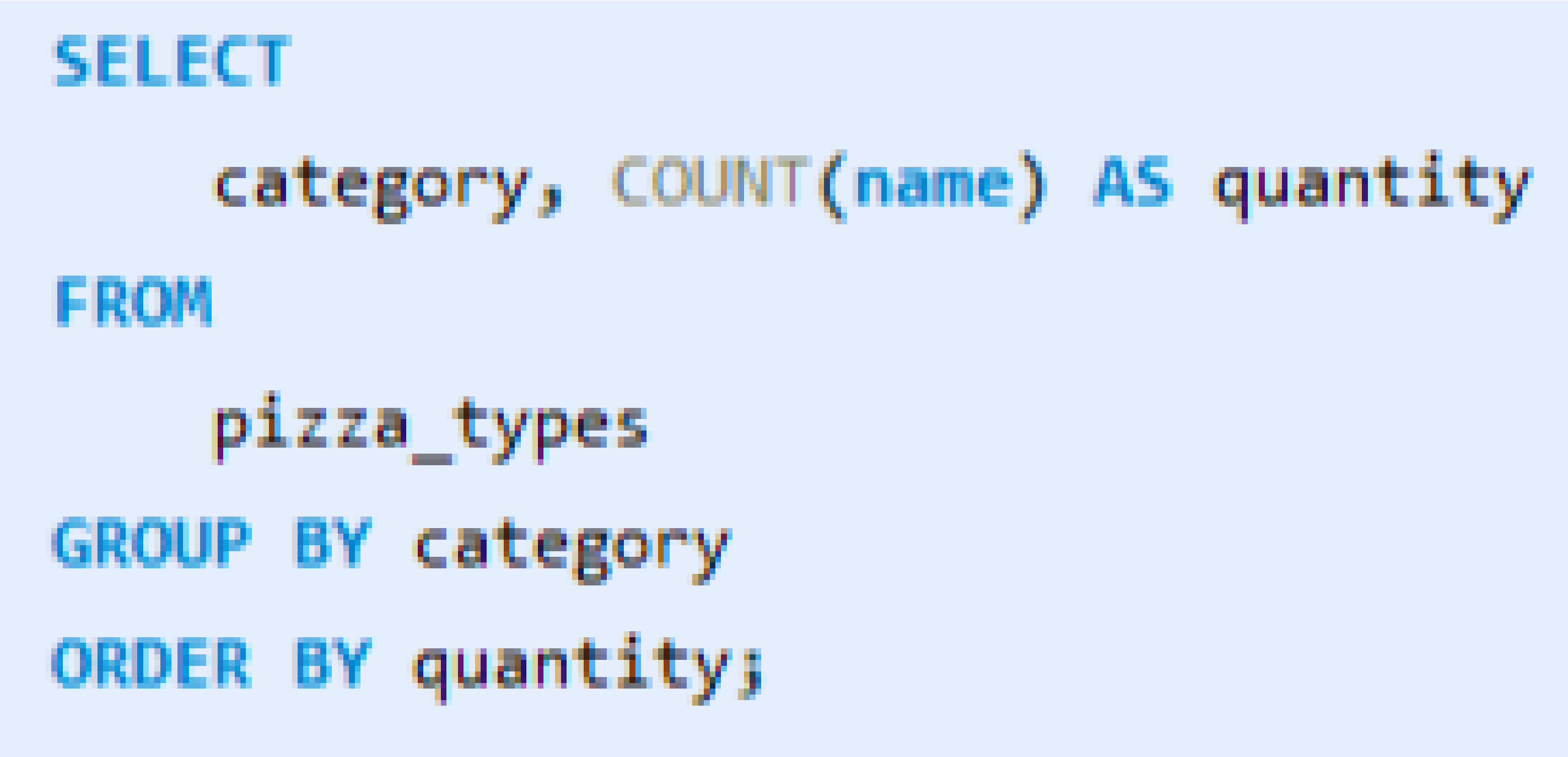


hour	count(order_id)
18	2399
19	2009
20	1642
21	1198
22	663
23	28
10	8
9	1

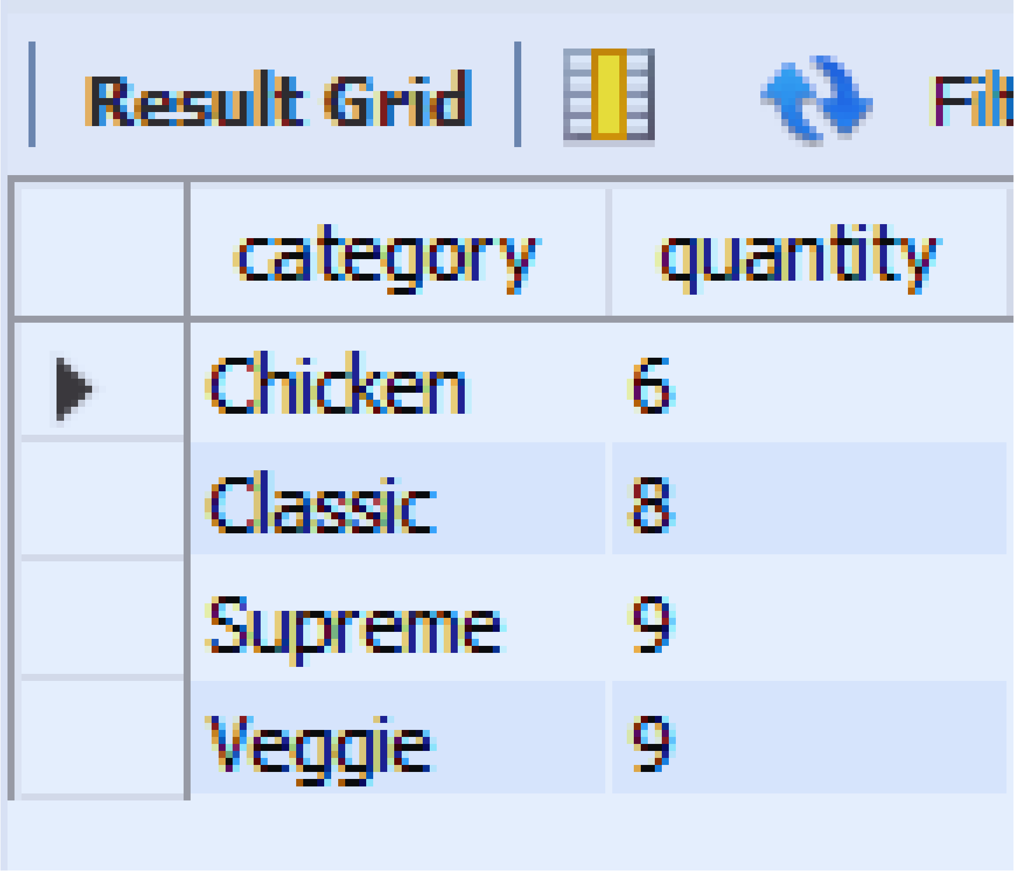
Problem 8

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

Solution:



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

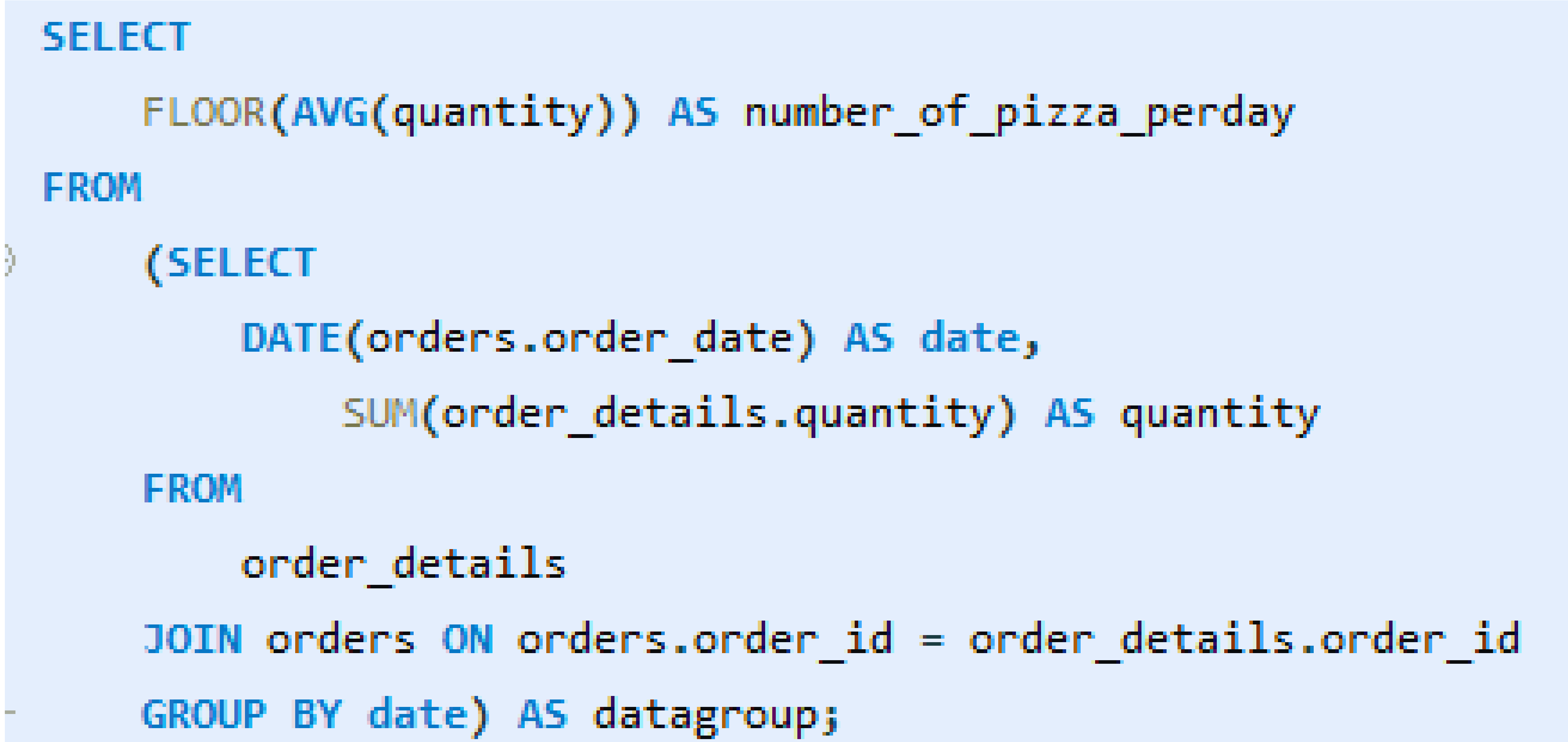


category	quantity
Chicken	6
Classic	8
Supreme	9
Veggie	9

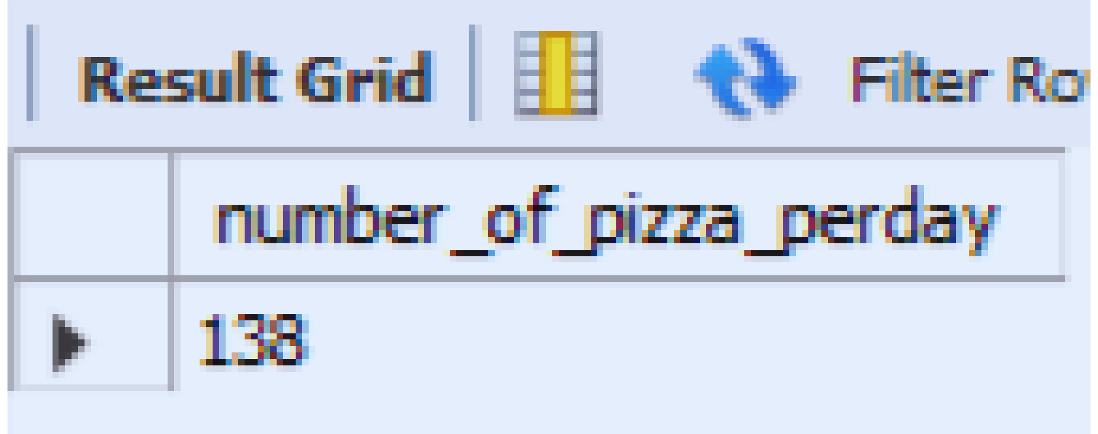
Problem 9

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

Solution:



```
SELECT
    FLOOR(AVG(quantity)) AS number_of_pizza_perday
FROM
    (SELECT
        DATE(orders.order_date) AS date,
            SUM(order_details.quantity) AS quantity
    FROM
        order_details
    JOIN orders ON orders.order_id = order_details.order_id
    GROUP BY date) AS datagroup;
```

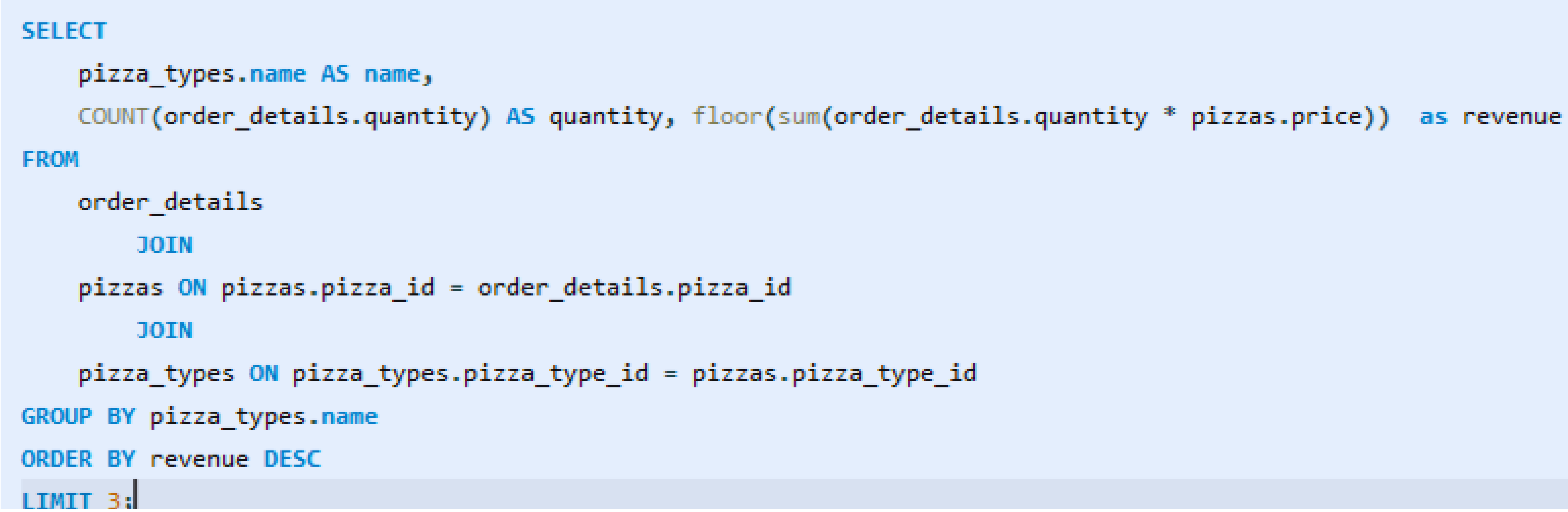


```
number_of_pizza_perday
138
```

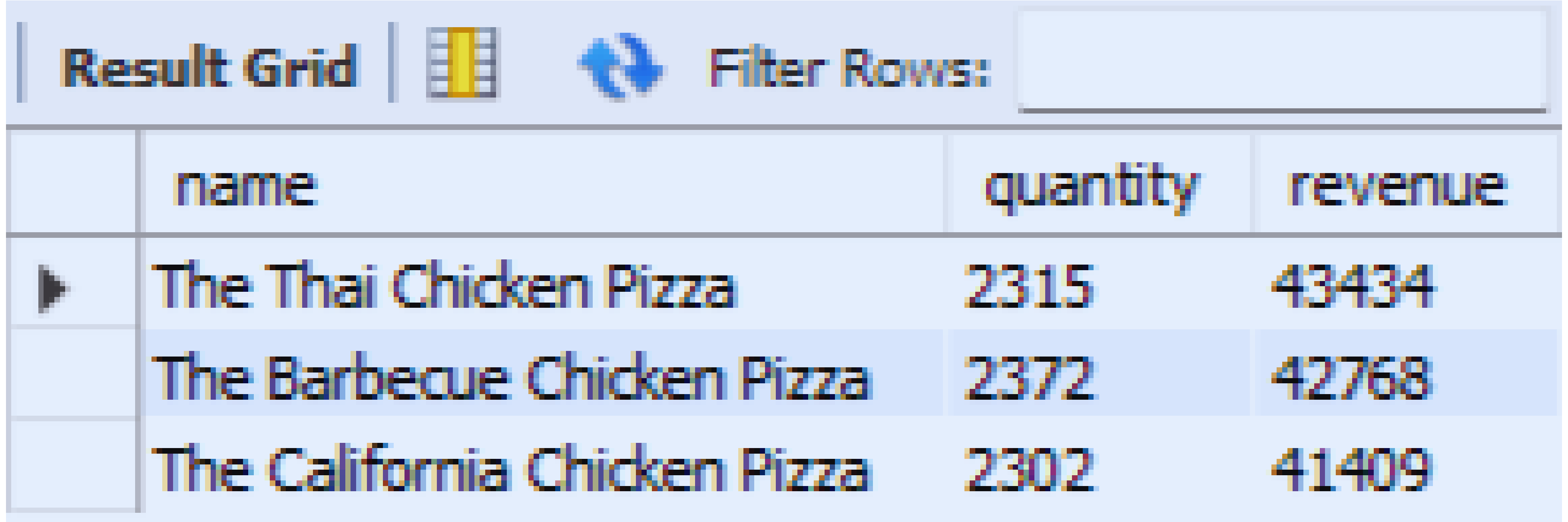
Problem 10

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

Solution:



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

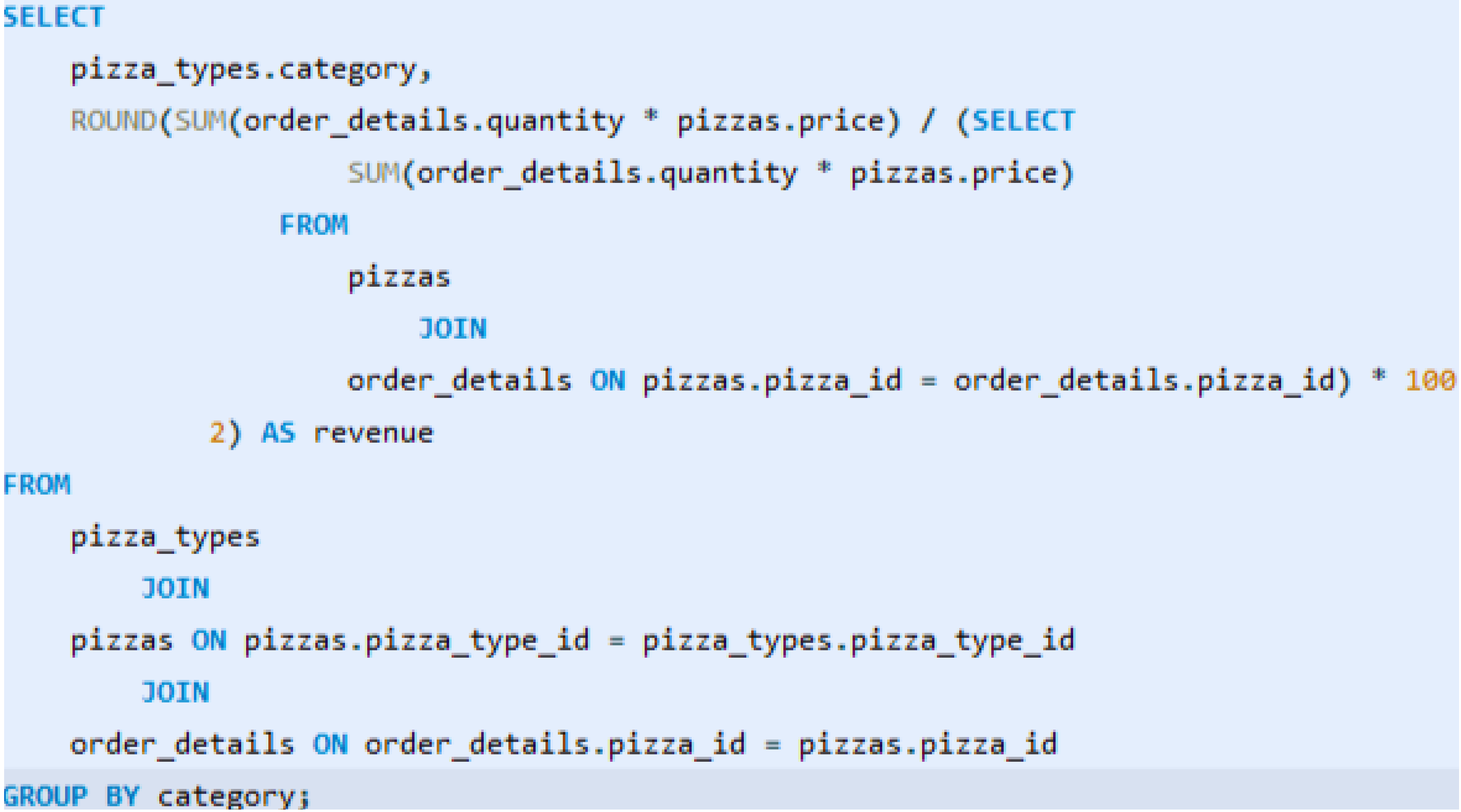


name	quantity	revenue
The Thai Chicken Pizza	2315	43434
The Barbecue Chicken Pizza	2372	42768
The California Chicken Pizza	2302	41409

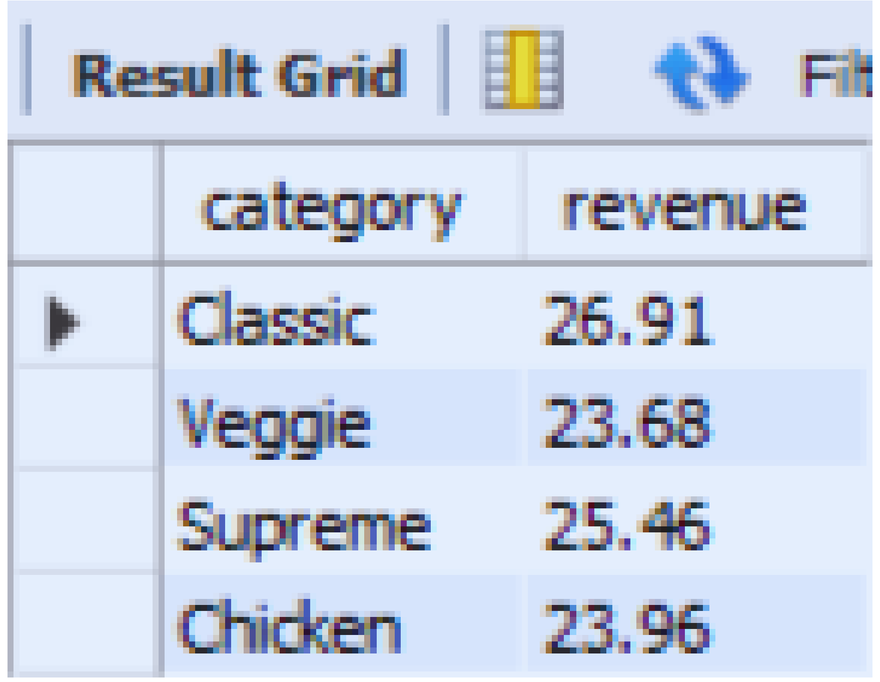
Problem 11

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

Solution:



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



category	revenue
Classic	26.91
Veggie	23.68
Supreme	25.46
Chicken	23.96

CONCLUSION:

This project successfully leveraged MySQL to analyze pizza sales data, enhancing my data analytics skills and demonstrating proficiency in SQL. Key findings include:

Sales Insights

Customer Preferences

Trends and Contributions