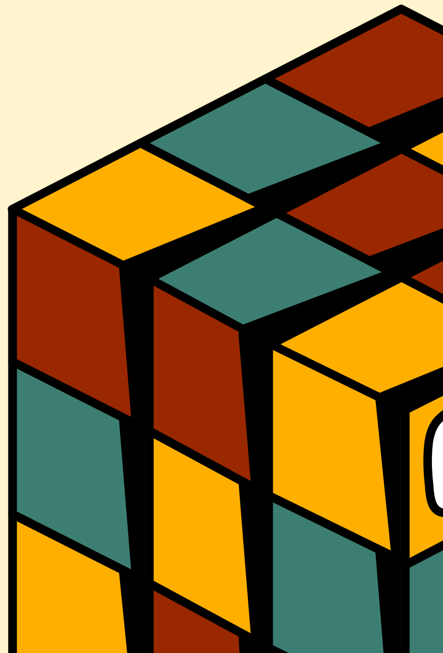
SQL PROJECT

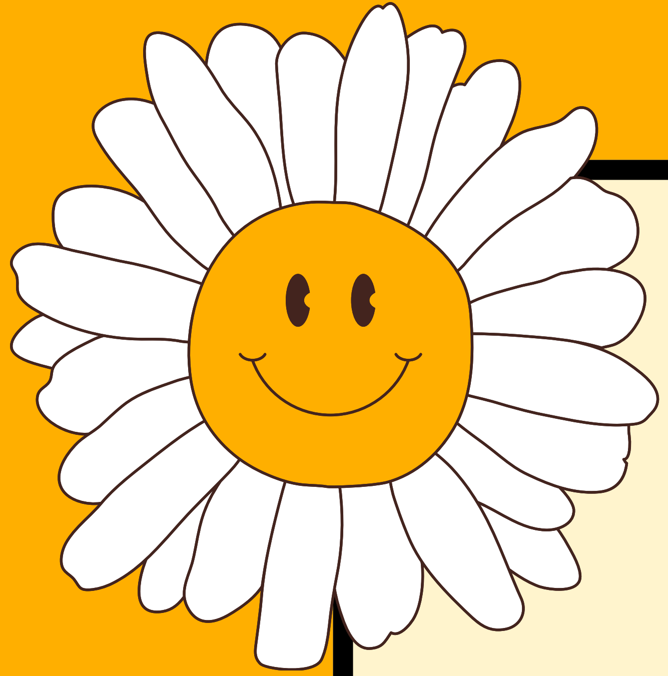
ON PIZZA SALES



Team Member

Atharva Kalyani

LEADER



Introduction

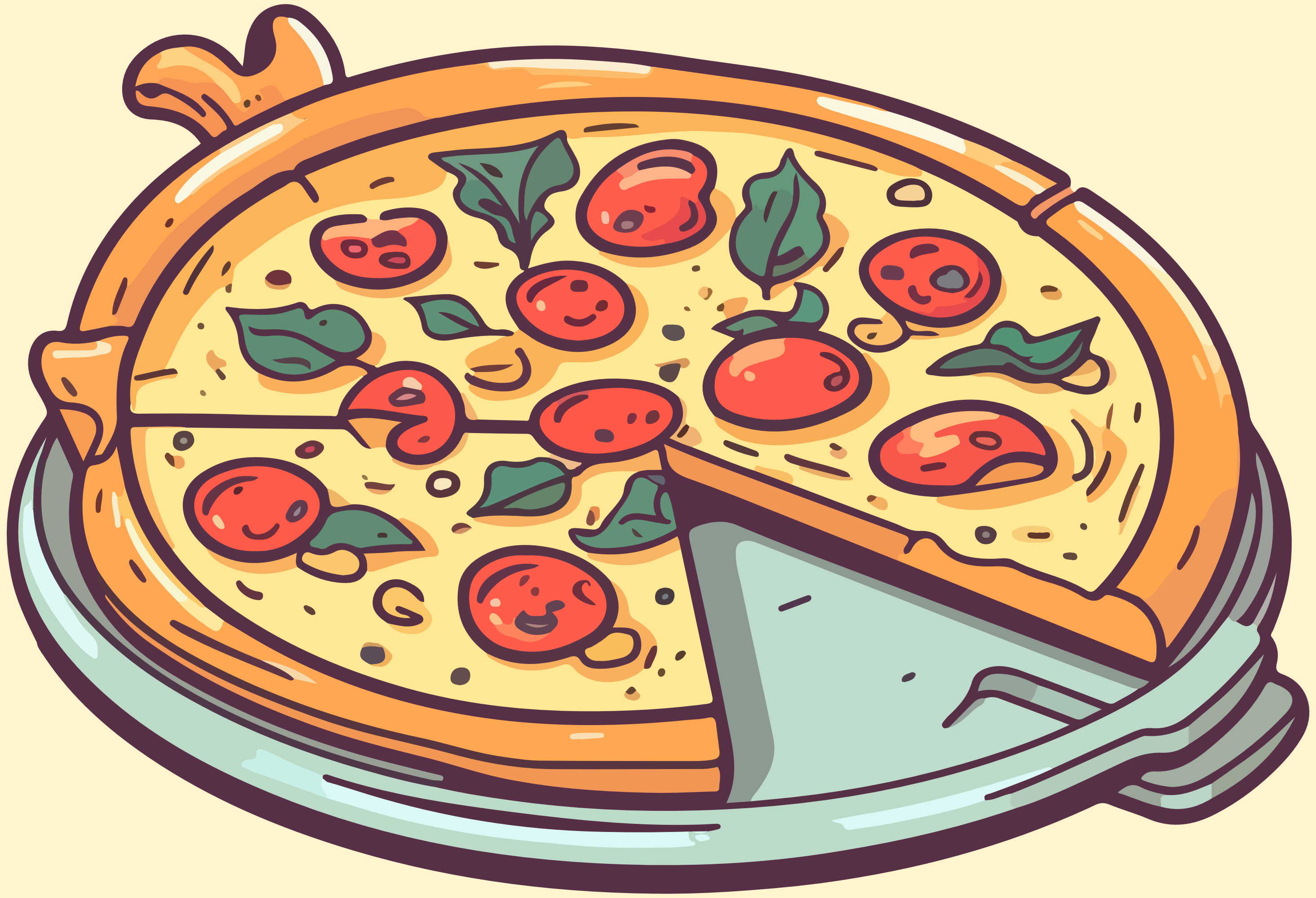
"In my SQL project on pizza sales, I've tackled seven queries to uncover insights like customer preferences, popular toppings, and sales trends. These queries inform strategic decision-making for optimizing sales and enhancing customer satisfaction in the pizza industry."

Background

Background project

Current scenario:

"In response to a desire for business expansion, the pizza company seeks to leverage SQL analysis for informed strategic planning. By harnessing data insights, they aim to identify growth opportunities, refine marketing strategies, and enhance operational efficiency to propel their business forward."



Sales Performance Analysis

Current Sales Trends

over the past years company's sales going down because of some reasons

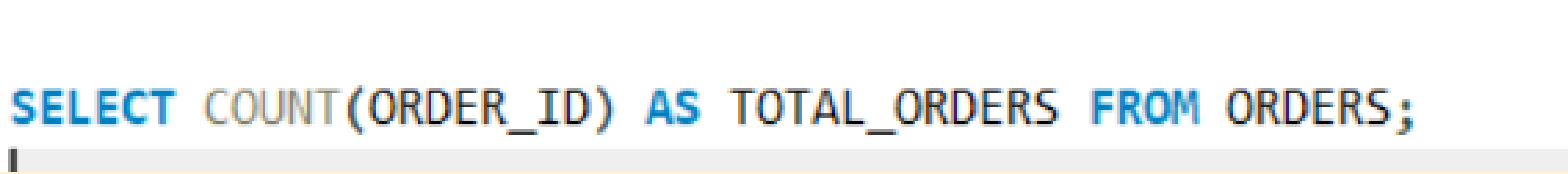
Market Insights

Using SQL analysis, we can identify current sales trends by examining recent sales data, including popular pizza types, peak sales hours, and customer demographics, enabling strategic adaptation to evolving market preferences.

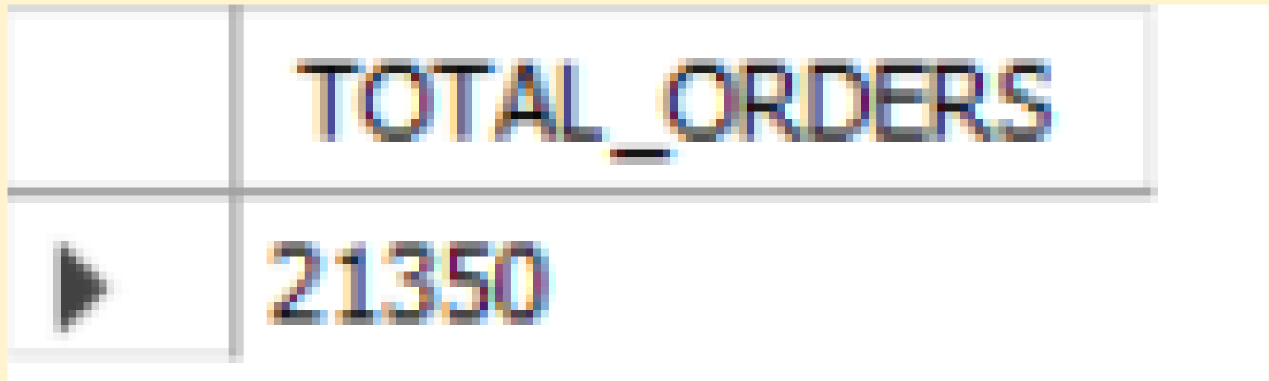
Customer Feedback

Direct feedback from our customers has highlighted areas for improvement

RETRIVE THE TOTAL NO OF ORDERS PLACED

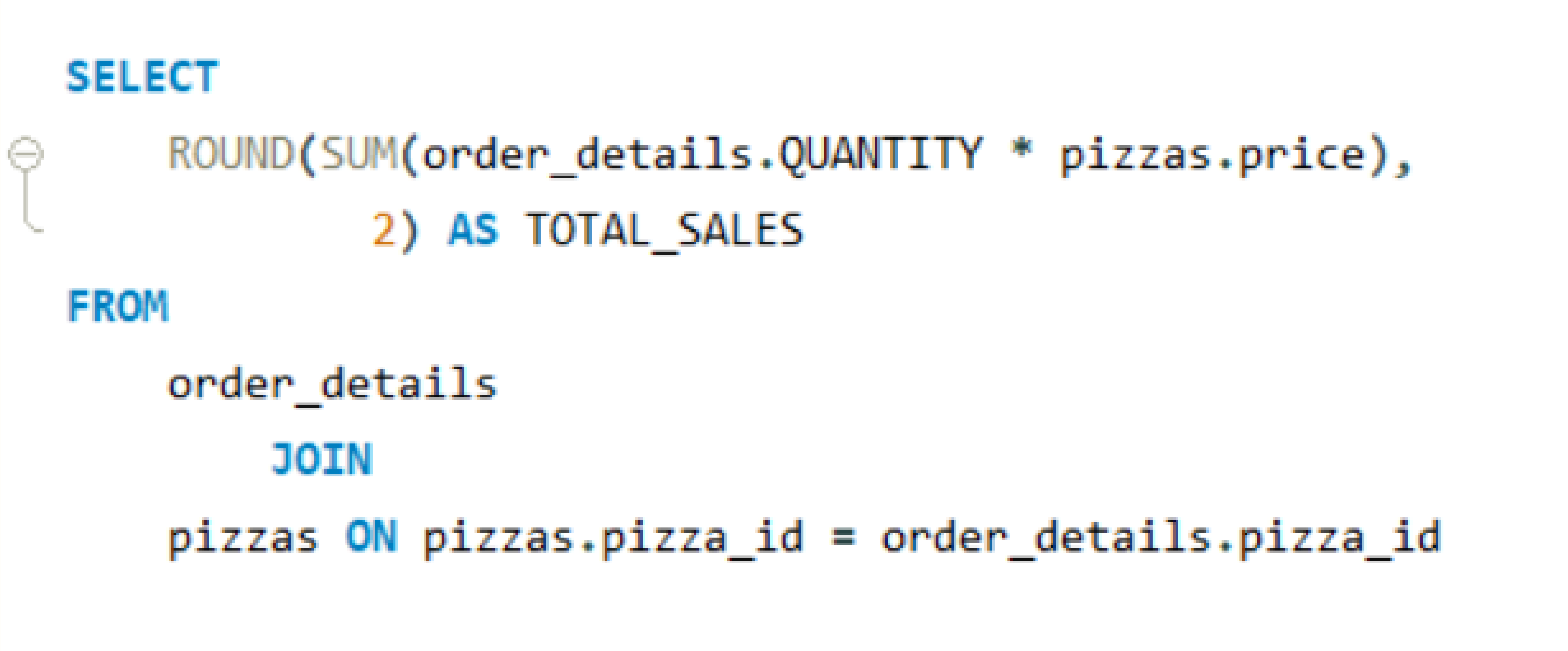


```
SELECT COUNT(ORDER_ID) AS TOTAL_ORDERS FROM ORDERS;
```

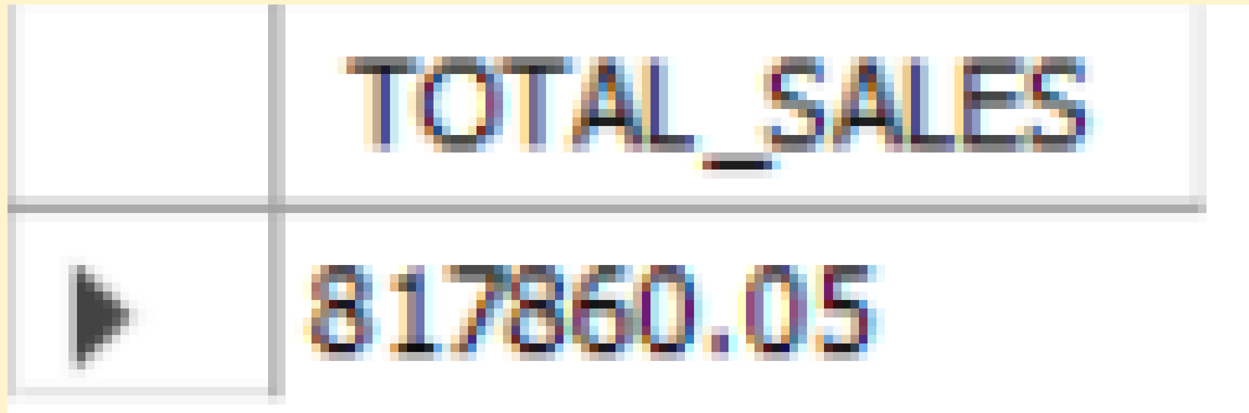


	TOTAL_ORDERS
▶	21350

CALCULATE TOTAL REVENUE GENERATED FROM PIZZA SALES

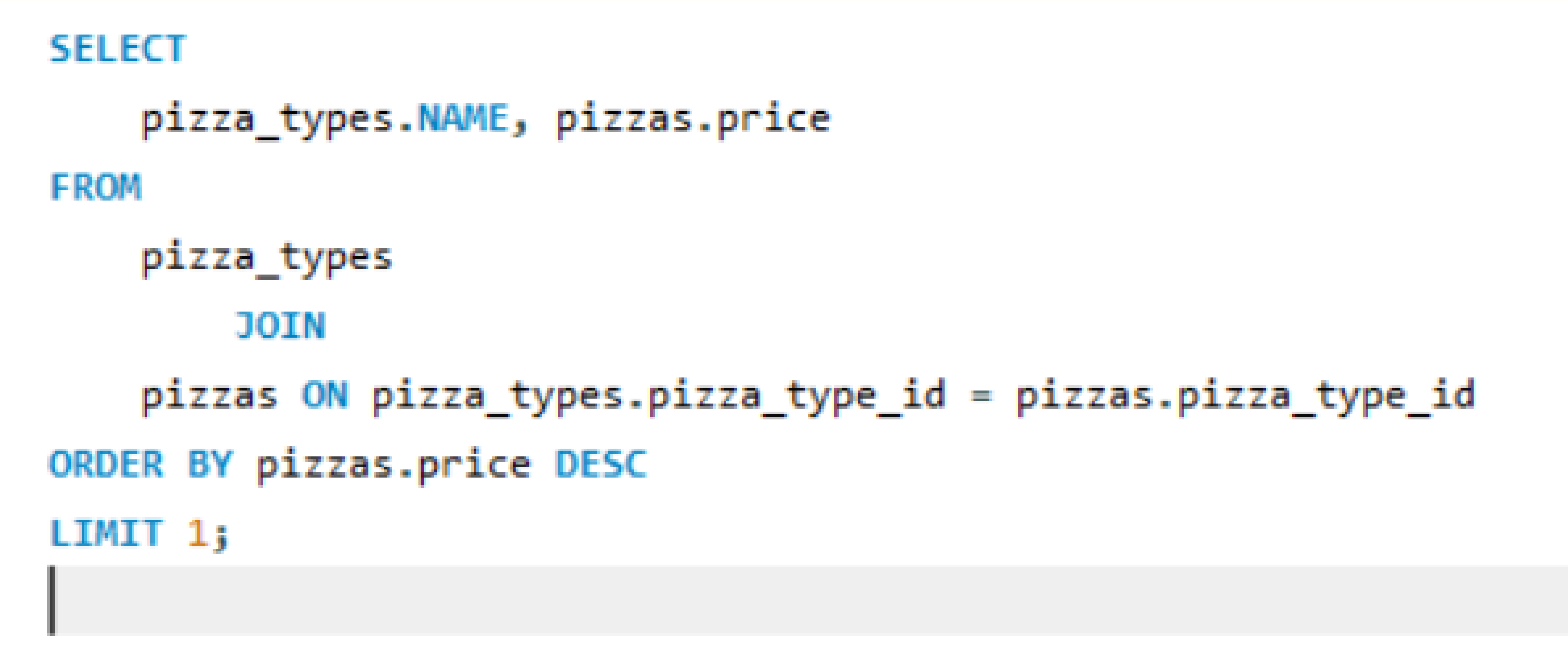


```
SELECT
    ROUND(SUM(order_details.QUANTITY * pizzas.price),
            2) AS TOTAL_SALES
FROM
    order_details
        JOIN
    pizzas ON pizzas.pizza_id = order_details.pizza_id
```

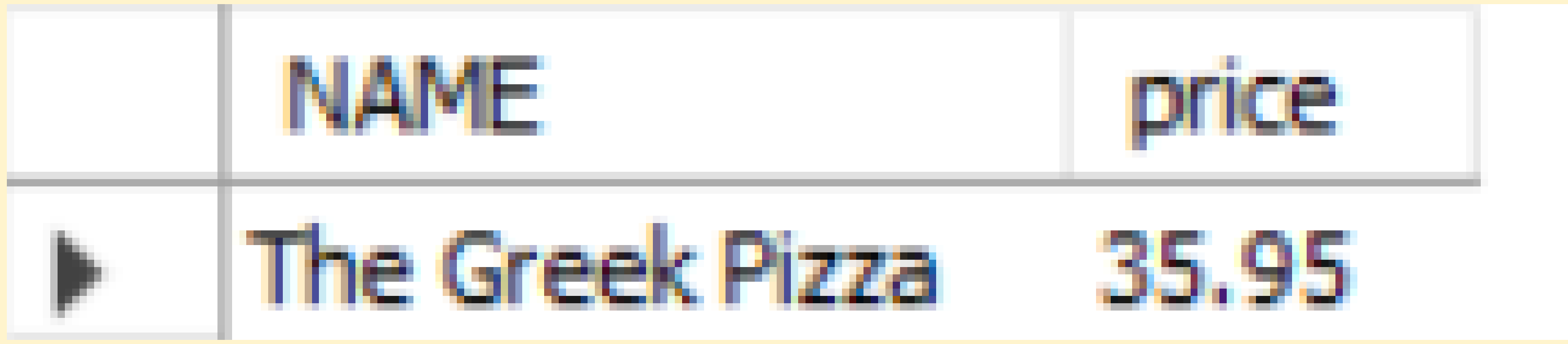


	TOTAL_SALES
▶	817860.05

IDENTIFY THE HIGHEST PRICED PIZZA

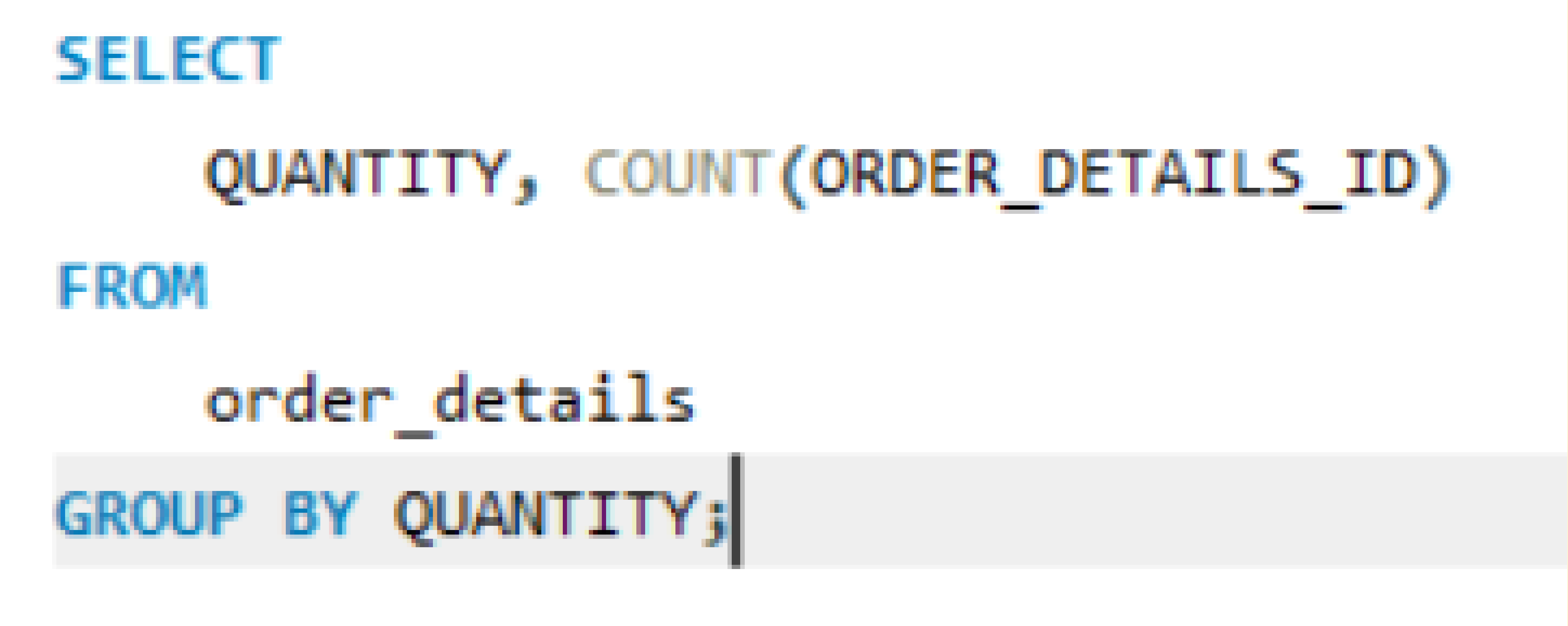


```
SELECT
    pizza_types.NAME, pizzas.price
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
ORDER BY pizzas.price DESC
LIMIT 1;
```

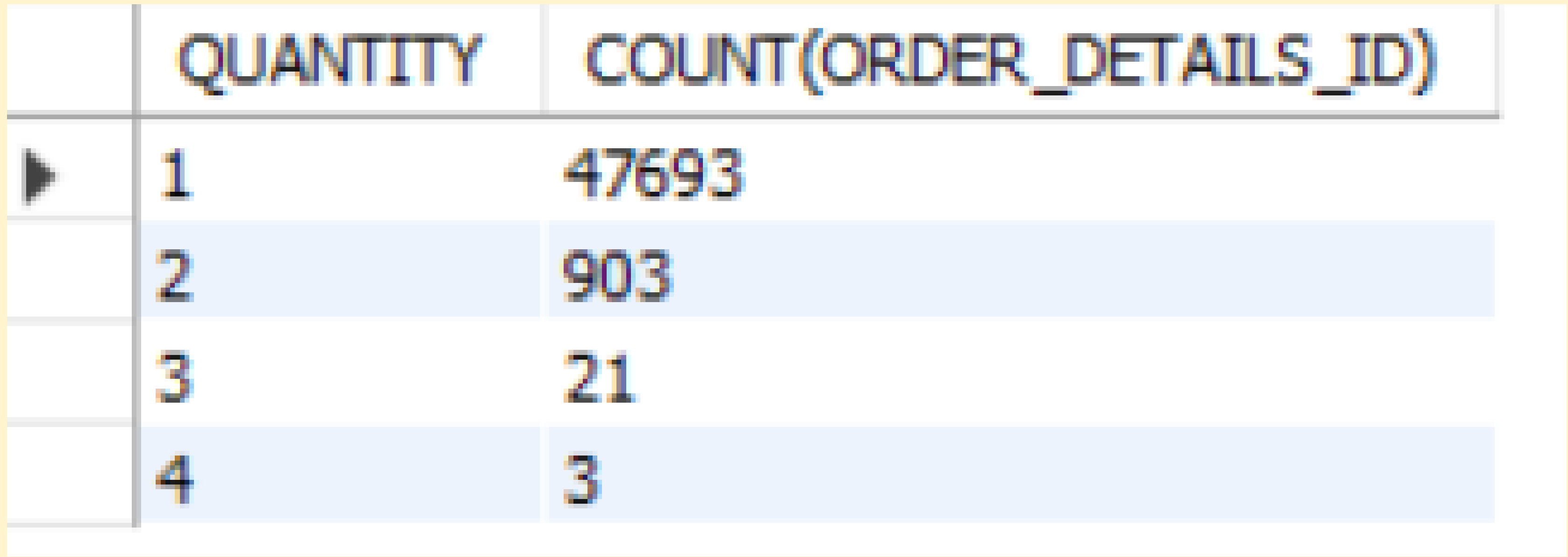


	NAME	price
▶	The Greek Pizza	35.95

IDENTIFY THE MOST COMMAN QUANTITY ORDERED

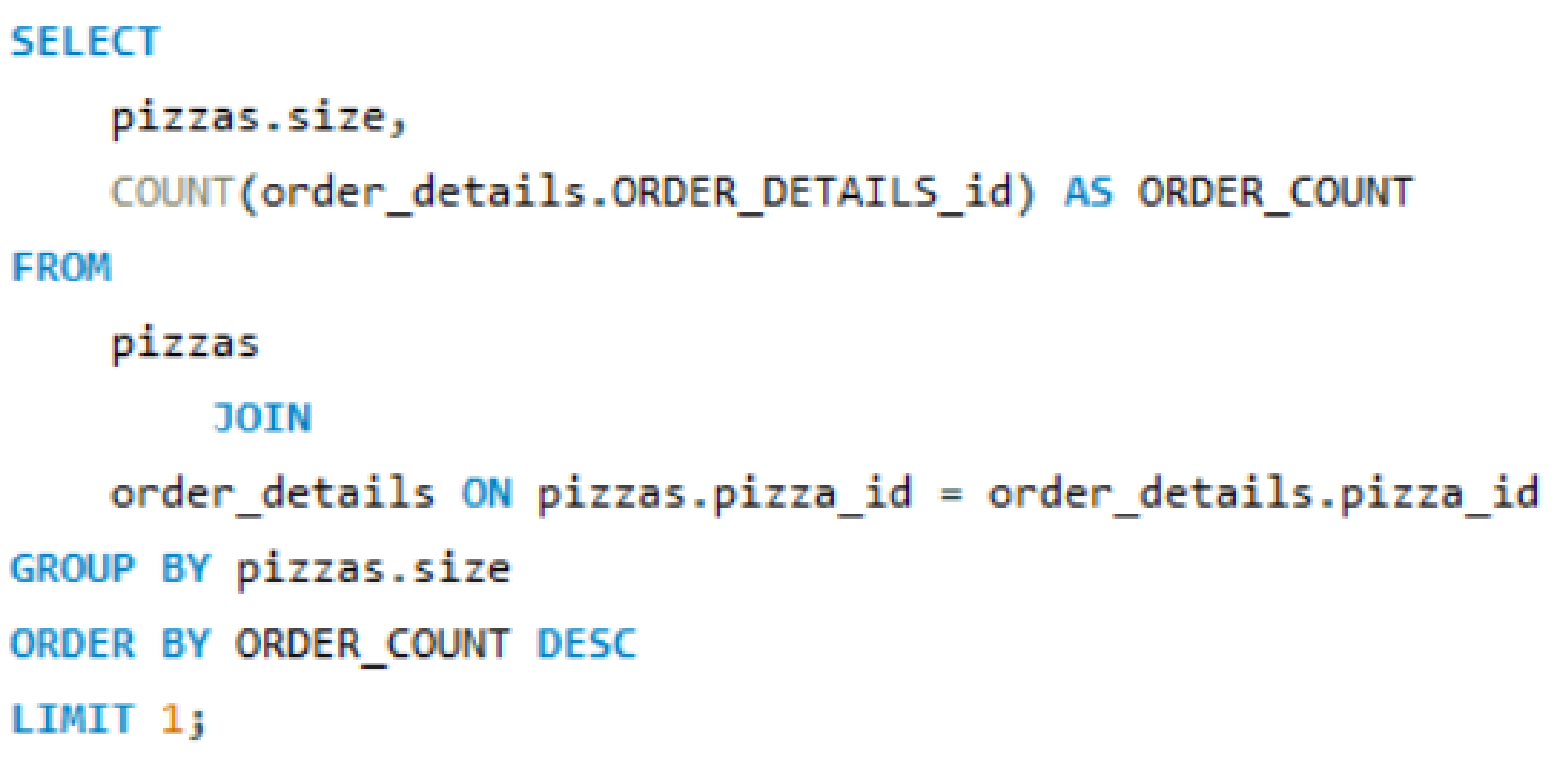


```
SELECT
    QUANTITY, COUNT(ORDER_DETAILS_ID)
FROM
    order_details
GROUP BY QUANTITY;
```

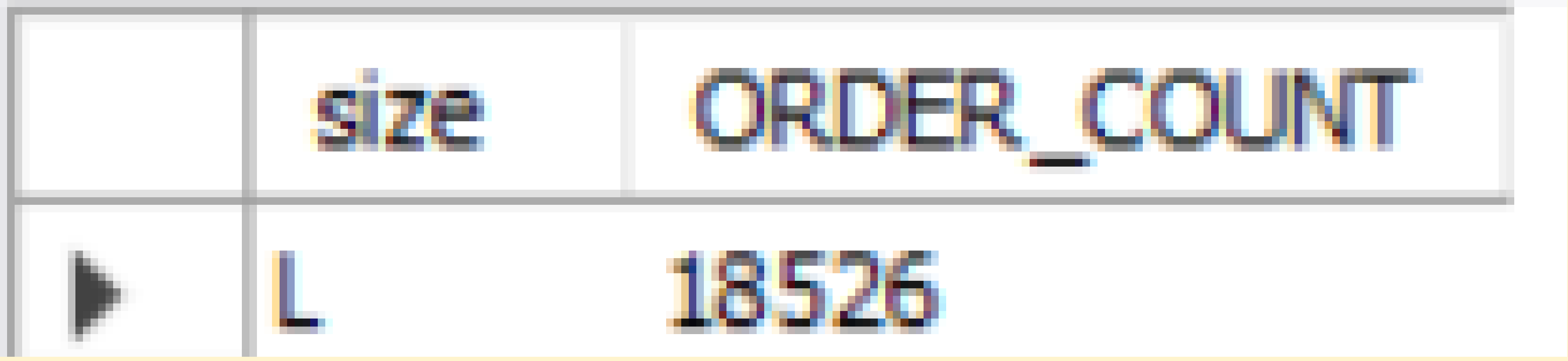


QUANTITY	COUNT(ORDER_DETAILS_ID)
1	47693
2	903
3	21
4	3

IDENTIFY THE MOST COMMAN PIZZA SIZE ORDERED

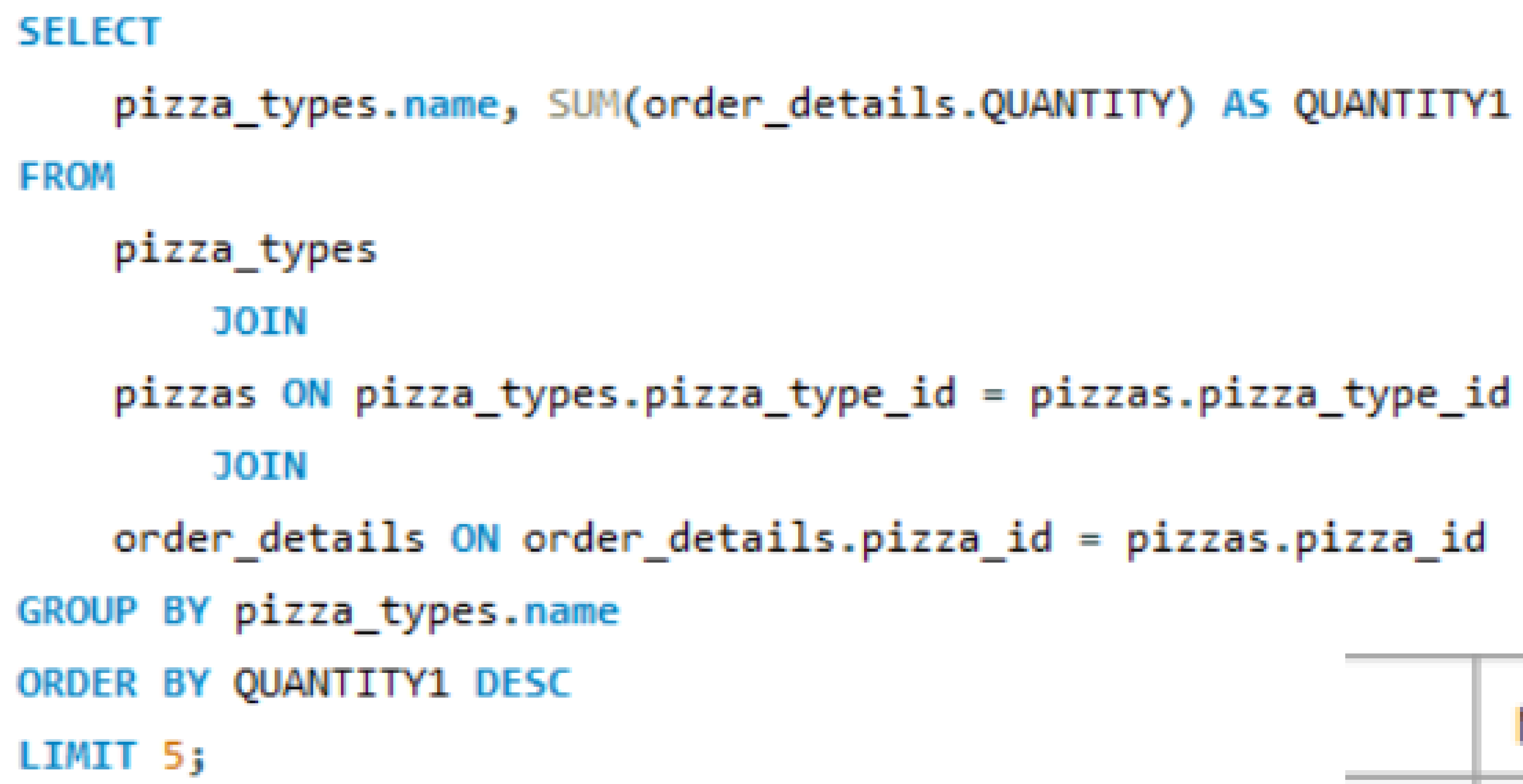


```
SELECT
    pizzas.size,
    COUNT(order_details.ORDER_DETAILS_id) AS ORDER_COUNT
FROM
    pizzas
        JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY pizzas.size
ORDER BY ORDER_COUNT DESC
LIMIT 1;
```

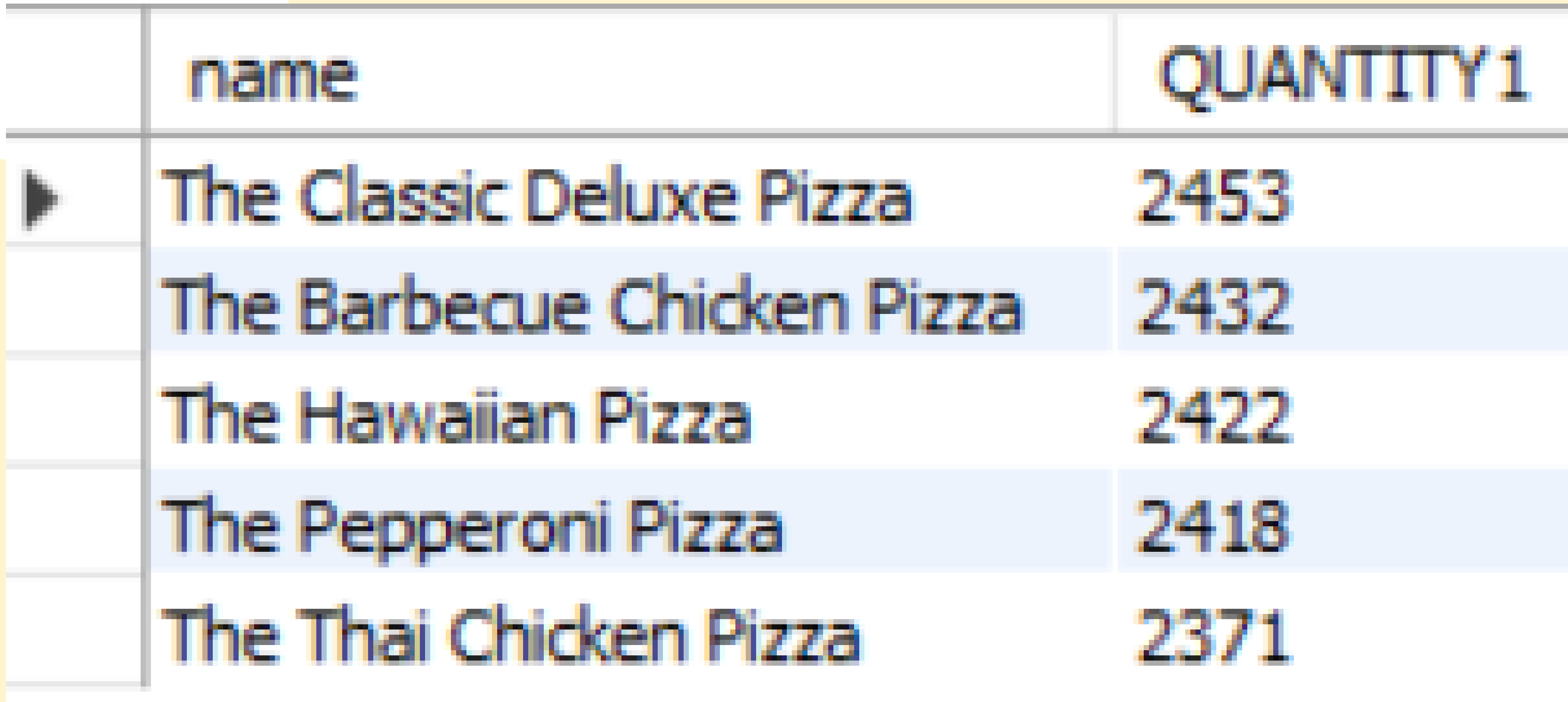


	size	ORDER_COUNT
▶	L	18526

LIST OF TOP 5 MOST ORDERED PIZZAS ALONG WITH THERE QUANTITIES

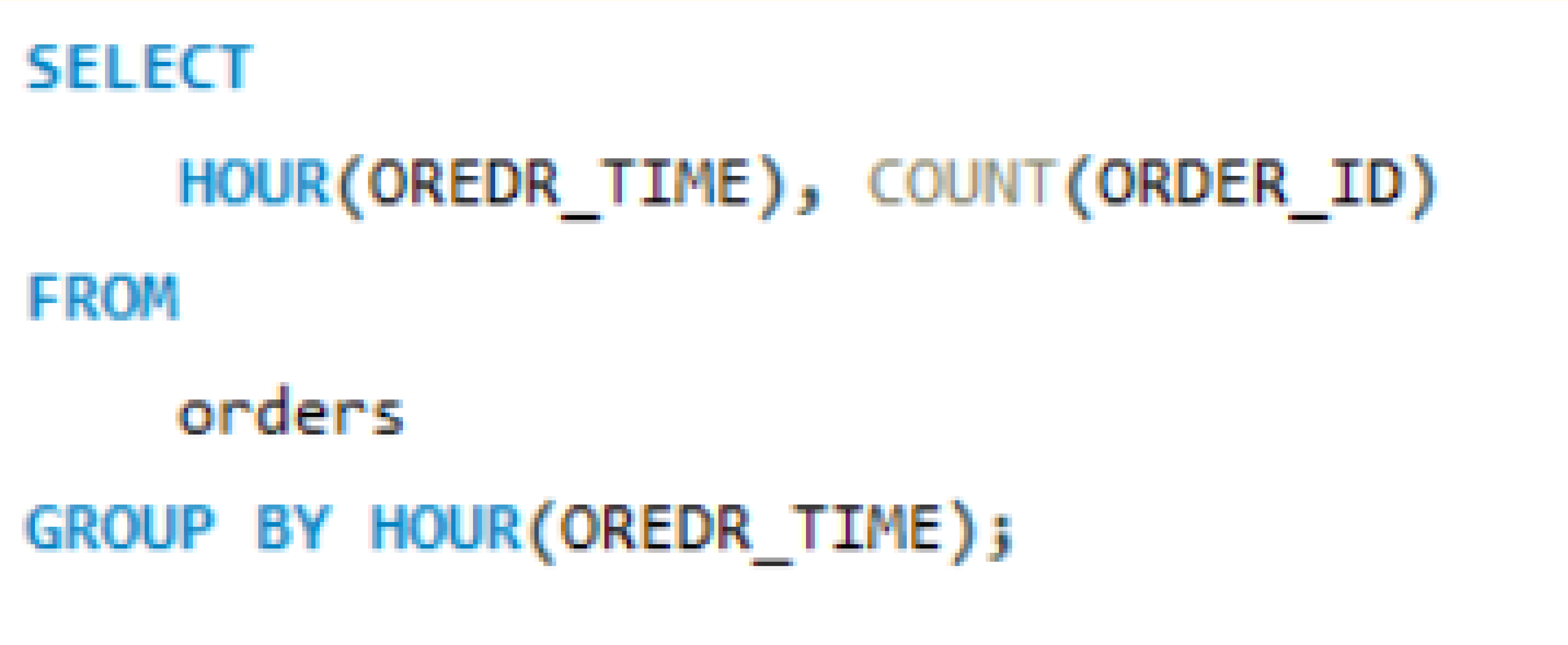


```
SELECT
    pizza_types.name, SUM(order_details.QUANTITY) AS QUANTITY1
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY QUANTITY1 DESC
LIMIT 5;
```

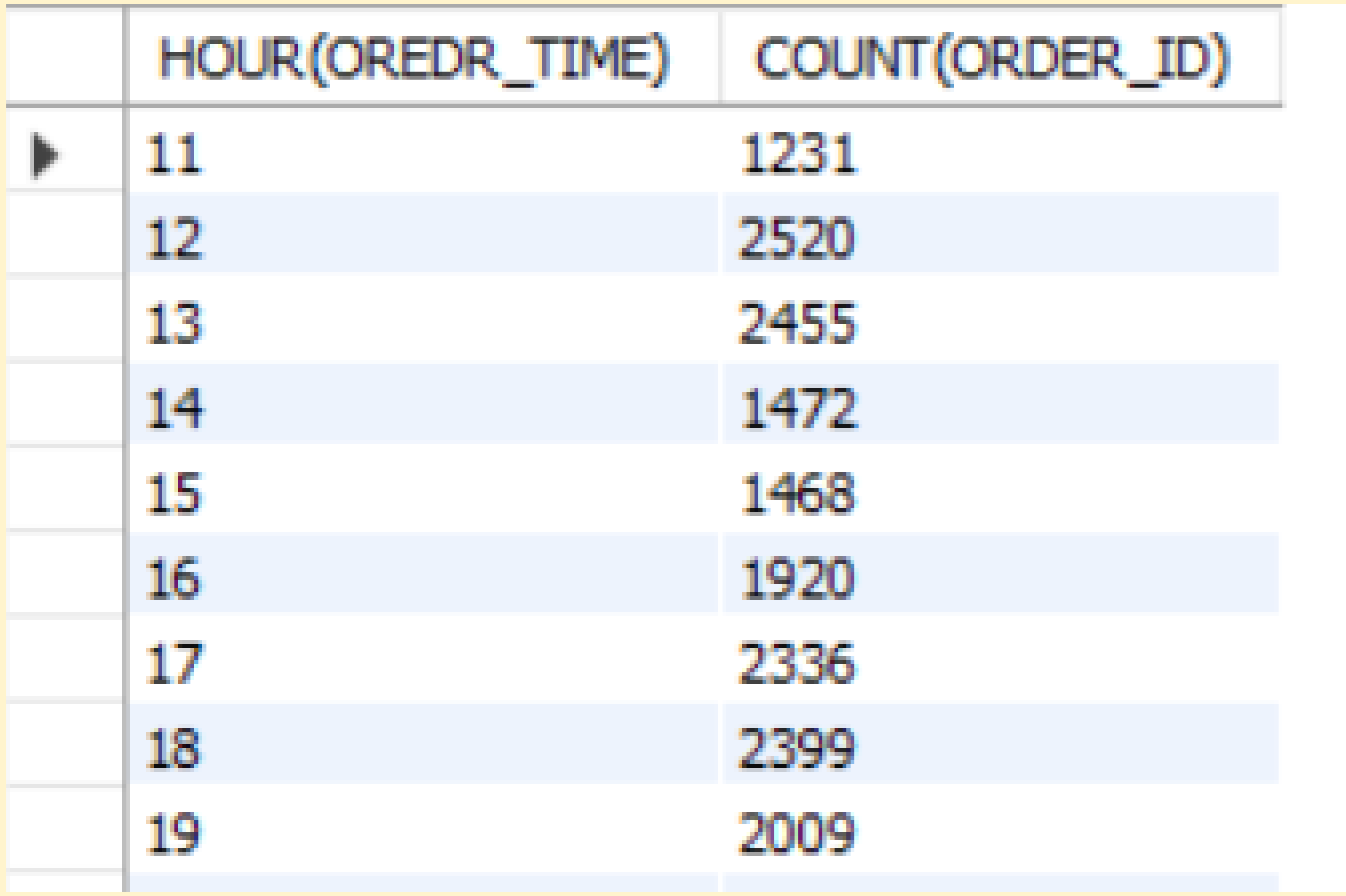


name	QUANTITY1
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	2422
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371

DETERMINE DISTRUBATION OF ORDERS BY HOUR OF THE DAY

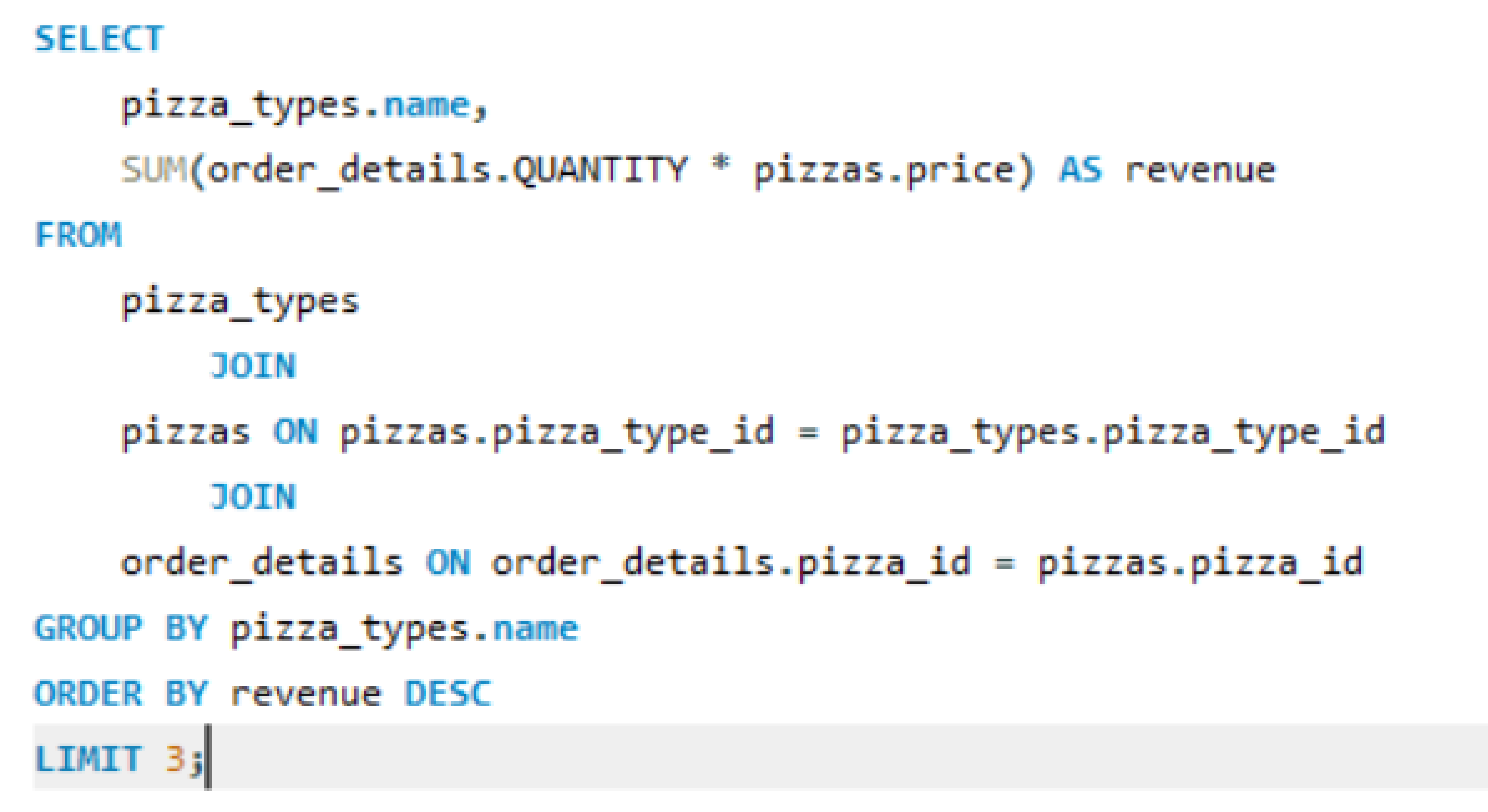


```
SELECT
    HOUR(OREDR_TIME), COUNT(ORDER_ID)
FROM
    orders
GROUP BY HOUR(OREDR_TIME);
```

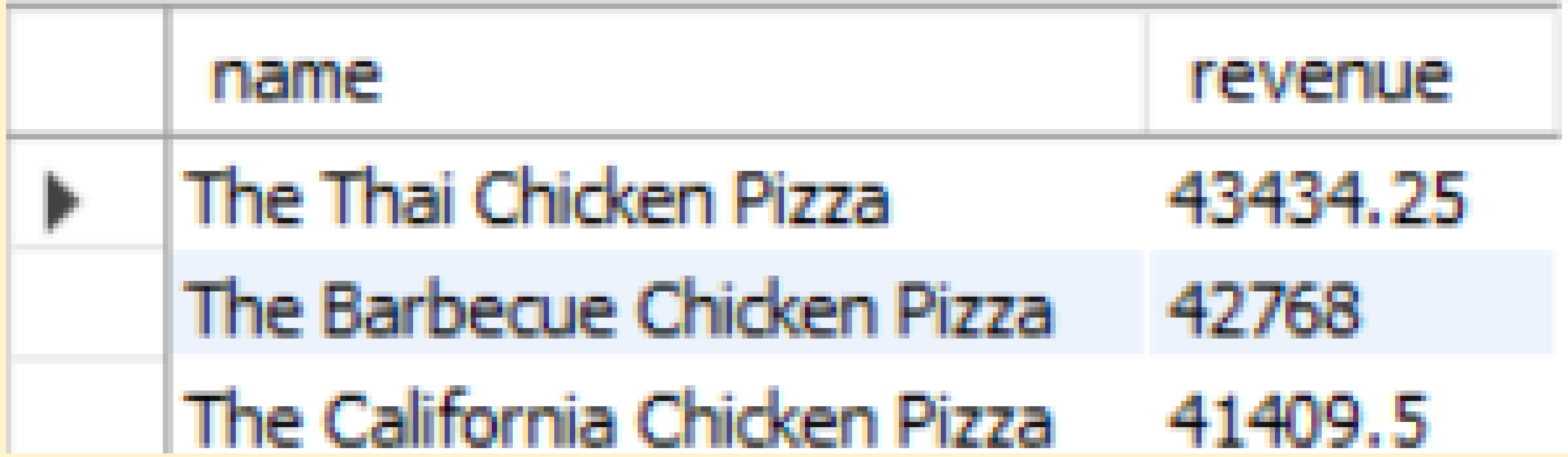


HOUR(OREDR_TIME)	COUNT(ORDER_ID)
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009

DETERMINE THE TOP 3 MOST ORDERED PIZZA BASED ON REVENUE



```
SELECT
    pizza_types.name,
    SUM(order_details.QUANTITY * pizzas.price) 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;
```



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

Thank You