SQL PROJECT

Analyzing Pizza Sales

ABOUT THIS PROJECT

Introduction

IN TODAY'S DYNAMIC BUSINESS LANDSCAPE, DATA PLAYS A PIVOTAL ROLE IN DECISION-MAKING PROCESSES ACROSS INDUSTRIES.

FOR THE FOOD SERVICE SECTOR, UNDERSTANDING CONSUMER PREFERENCES AND MARKET TRENDS IS ESSENTIAL FOR DRIVING GROWTH AND MAINTAINING A COMPETITIVE EDGE.

IN THIS PROJECT, I DELVE INTO THE WORLD OF PIZZA SALES DATA TO EXTRACT VALUABLE INSIGHTS USING SQL QUERIES AND SUB QUERIES.

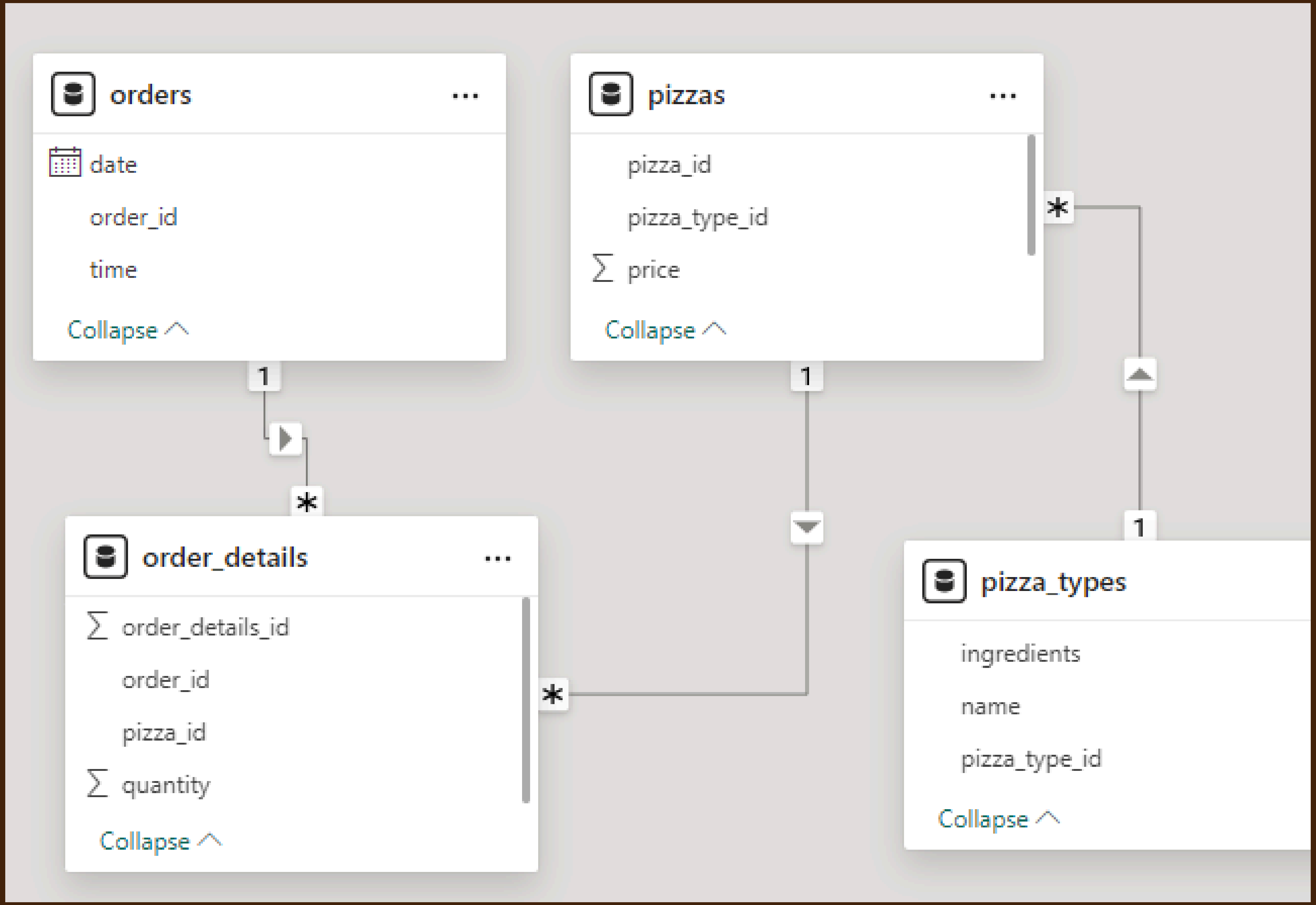
OBJECTIVE

- THE OBJECTIVE OF THIS PROJECT IS TO UTILIZE SQL QUERIES AND SUB QUERIES TO ANALYSE A DATASET CONTAINING INFORMATION ABOUT PIZZA SALES.
- THE DATASET PROVIDES A COMPREHENSIVE OVERVIEW OF VARIOUS ASPECTS OF PIZZA, INCLUDING CUSTOMER ORDERS, PIZZA TYPES, PRICES, DATES, AND ORDER DETAILS
- THROUGH THIS PROJECT, I AIM TO EXPLORE THE DATASET, ANSWER SPECIFIC QUESTIONS RELATED TO PIZZA SALES, AND GAIN INSIGHTS INTO POPULAR PIZZA CATEGORY, BEST-SELLING ITEMS, PEAK SALES TIME, CUSTOMER PREFERENCES, AND MORE.

PROJECT STRUCTURE

THE PROJECT IS STRUCTURED INTO SECTIONS, EACH FOCUSING ON SPECIFIC ASPECTS OF PIZZA SALES ANALYSIS. WITHIN EACH SECTION, QUESTIONS OF VARYING DIFFICULTY LEVELS ARE PRESENTED, ALONG WITH CORRESPONDING SQL QUERIES AND RESULTS.

DATABASE SCHEMA

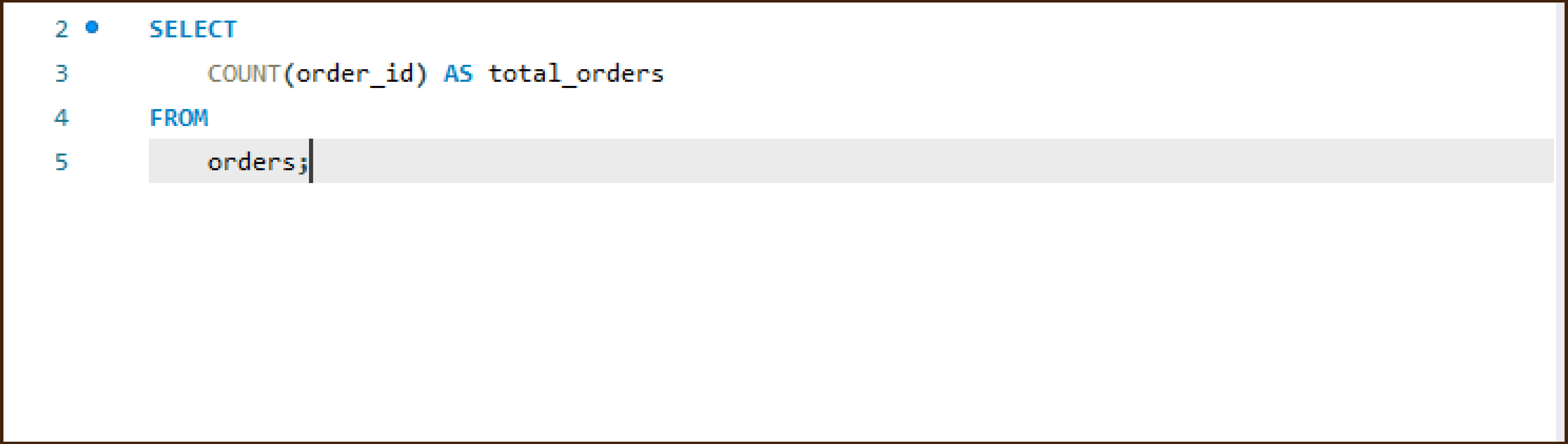


THE DATASET USED IN THIS PROJECT COMPRISES MULTIPLE TABLES, EACH CONTAINING SPECIFIC INFORMATION ABOUT DIFFERENT ASPECTS OF PIZZA SALES.

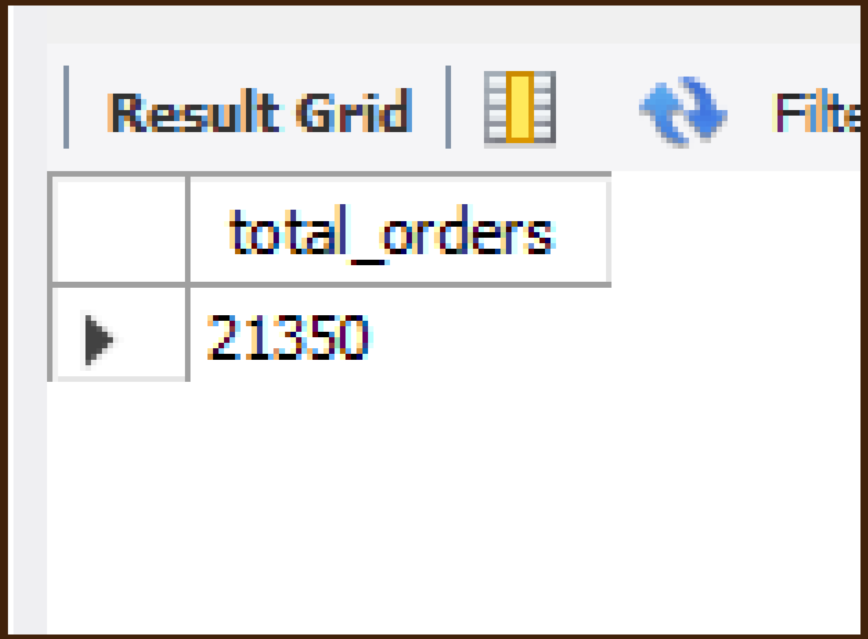
THESE TABLES INCLUDE DETAILS SUCH AS PIZZAS, ORDER DETAILS, PIZZA TYPES, PIZZA ORDERS.

THE DATASET IS RICH IN INFORMATION, PROVIDING US WITH AMPLE OPPORTUNITIES TO EXTRACT MEANINGFUL INSIGHTS THROUGH SQL QUERIES.

RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.

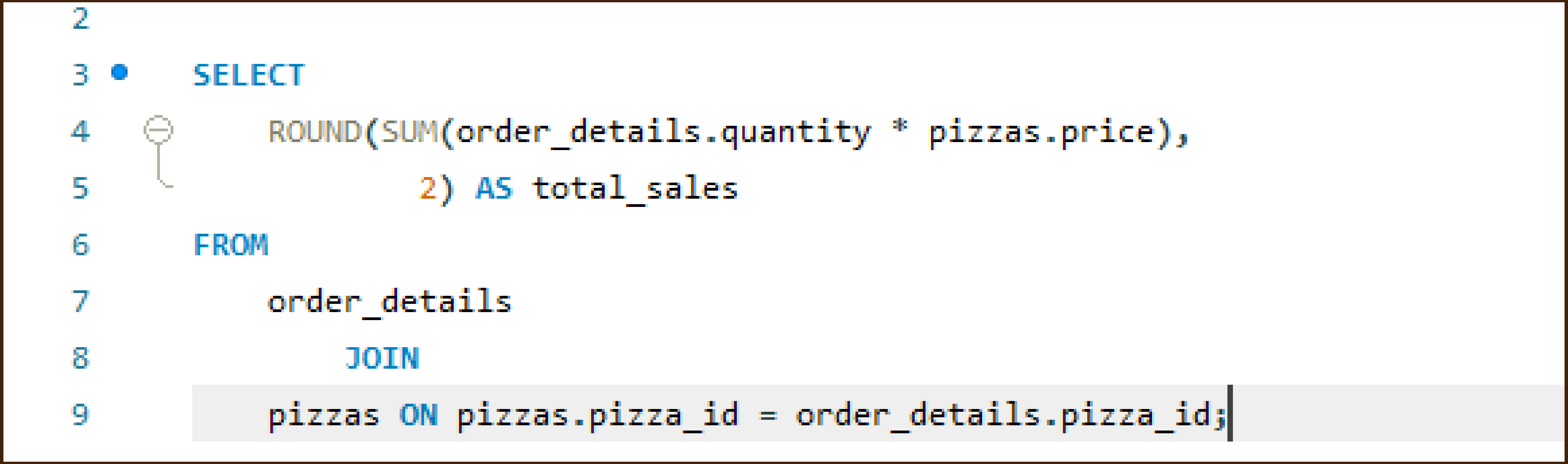


```sql
SELECT
    COUNT(order_id) AS total_orders
FROM
    orders;
```

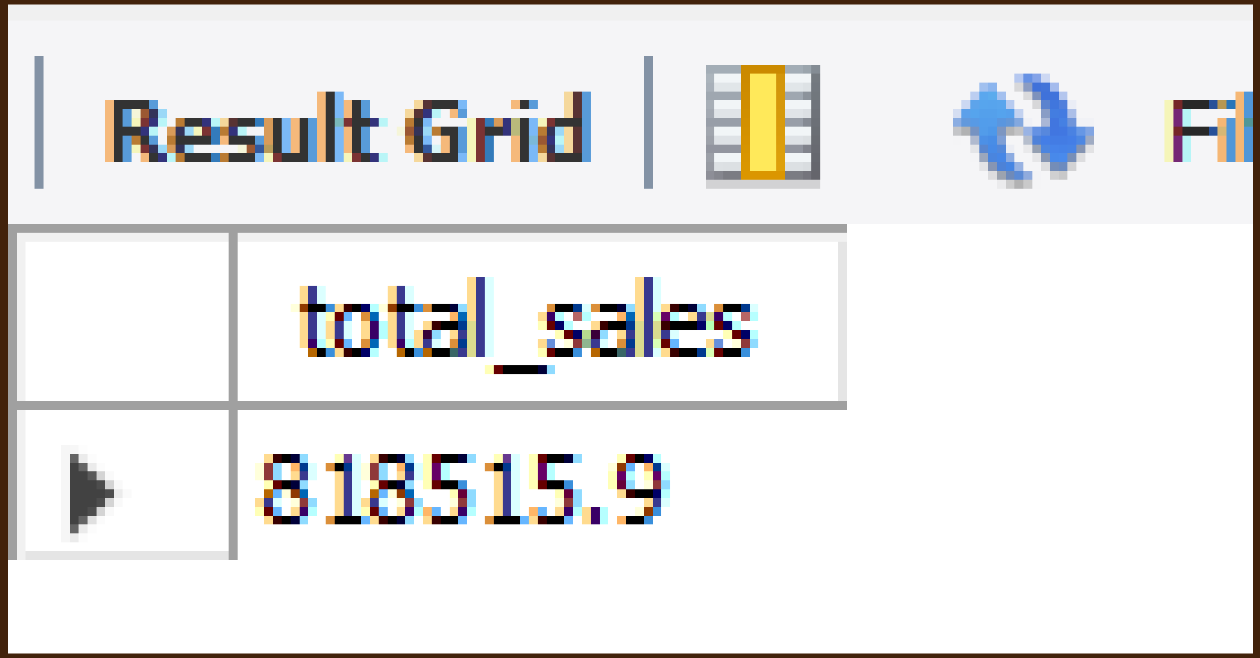


total_orders
21350

CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

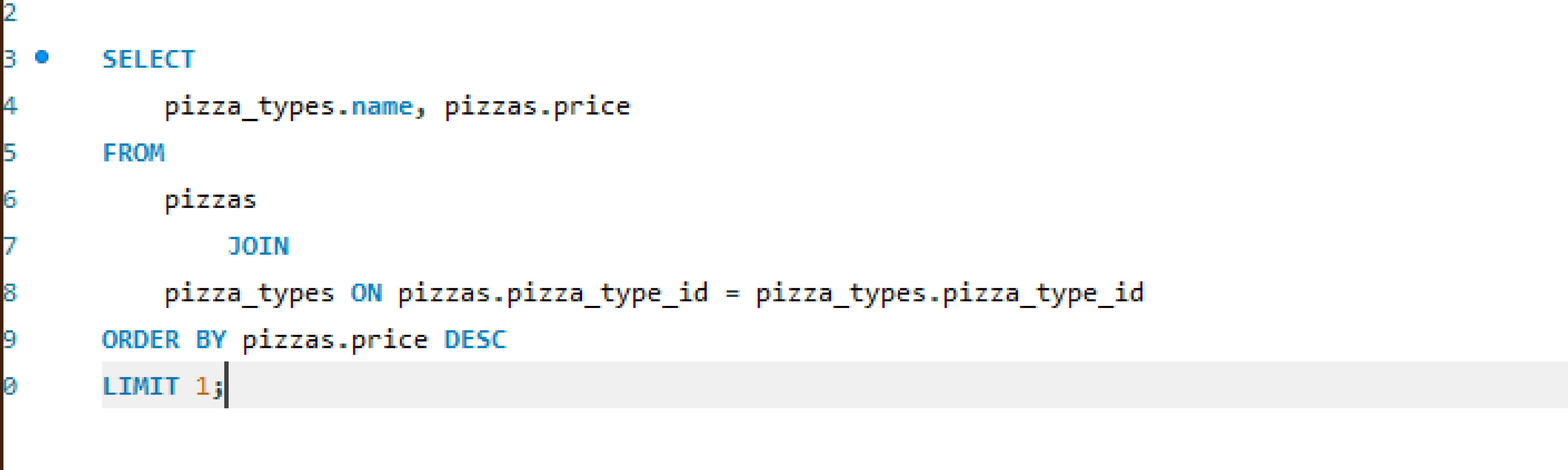


```
SELECT
    ROUND(SUM(order_details.quantity * pizzas.price),
            2) AS total_sales
FROM
    order_details
        JOIN
    pizzas ON pizzas.pizza_id = order_details.pizza_id;
```

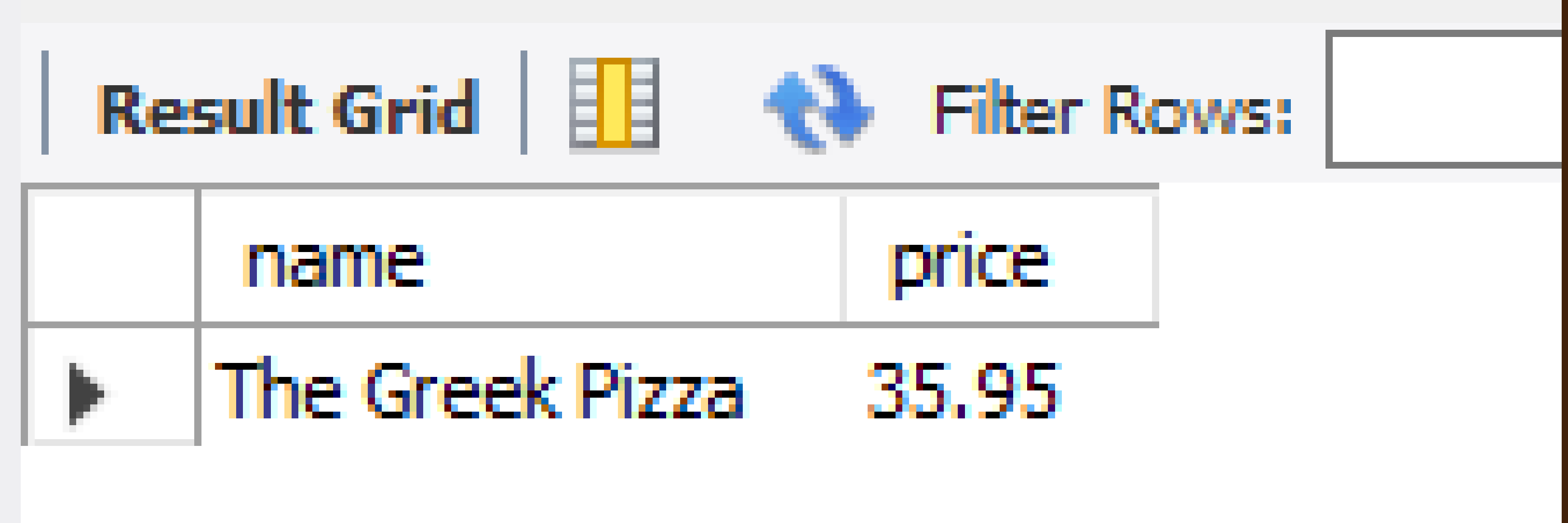


```
total_sales
818515.9
```

IDENTIFY THE HIGHEST-PRICED PIZZA.

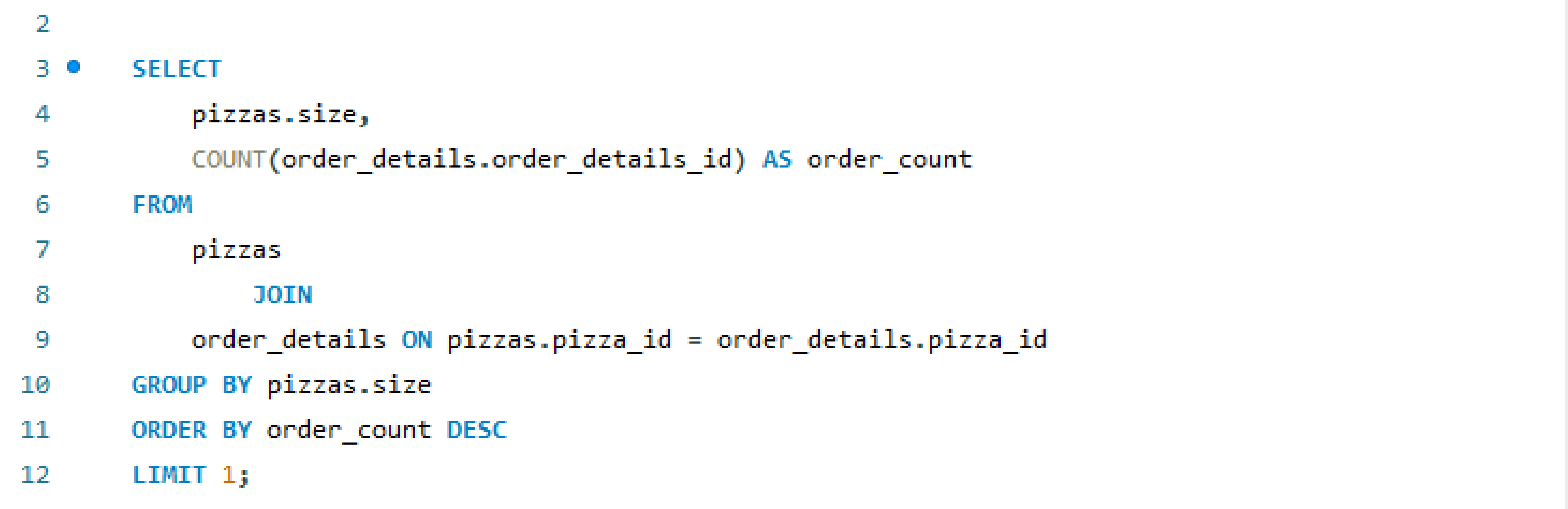


```sql
SELECT
    pizza_types.name, pizzas.price
FROM
    pizzas
        JOIN
    pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
ORDER BY pizzas.price DESC
LIMIT 1;
```



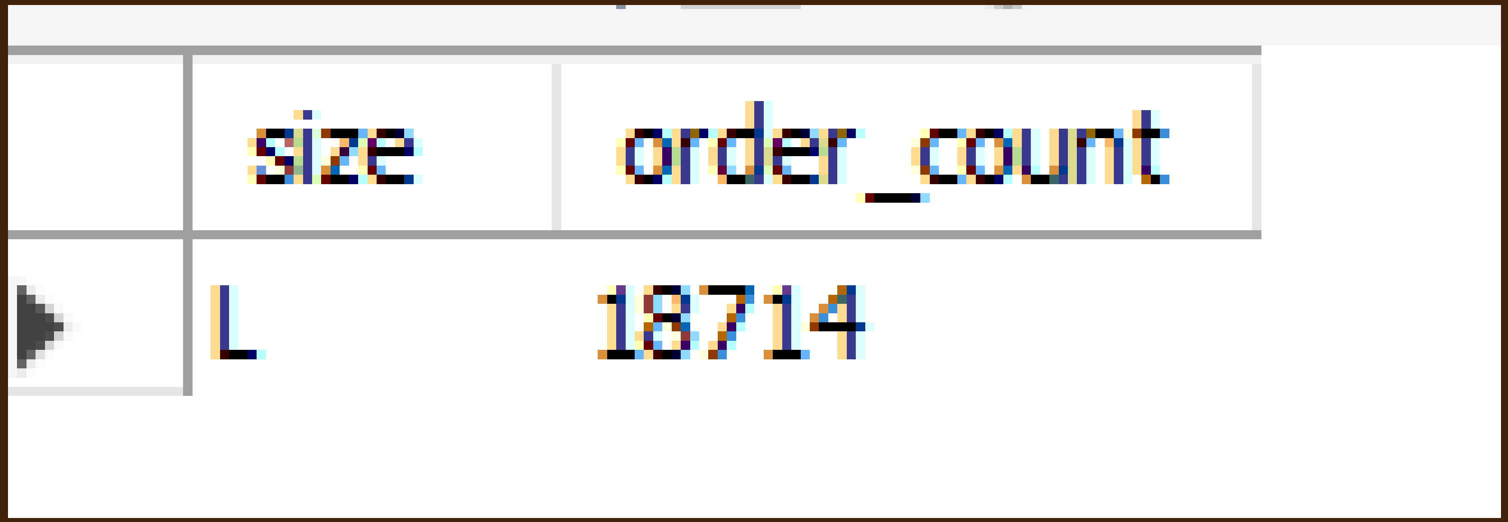
name	price
The Greek Pizza	35.95

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.



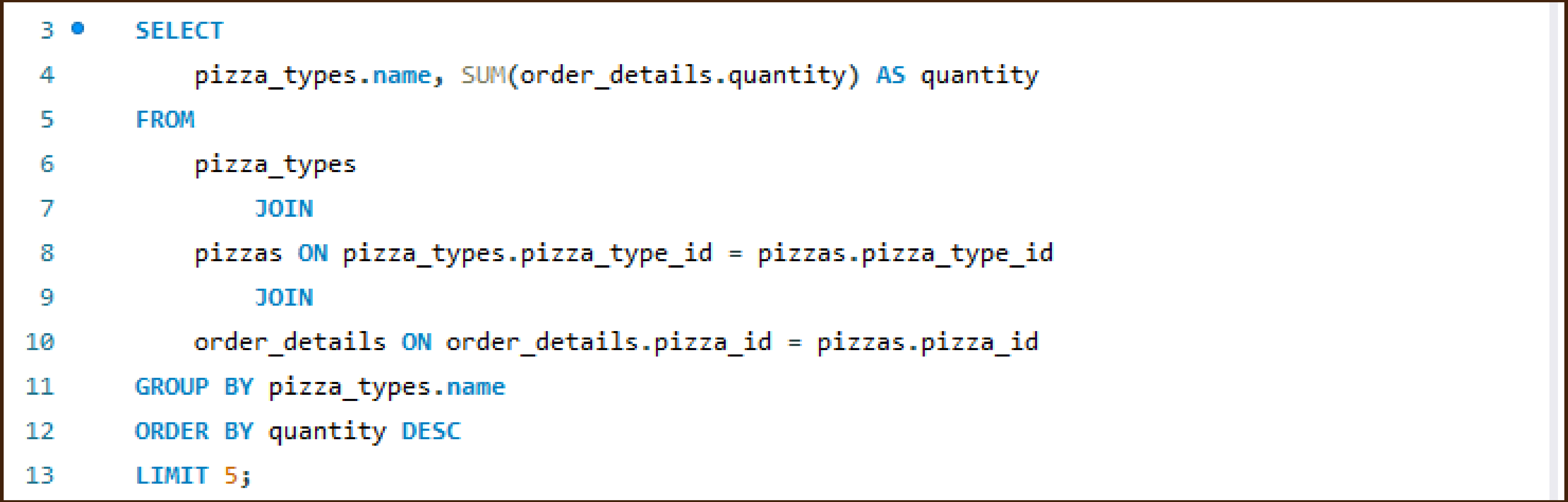
```sql

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;
```

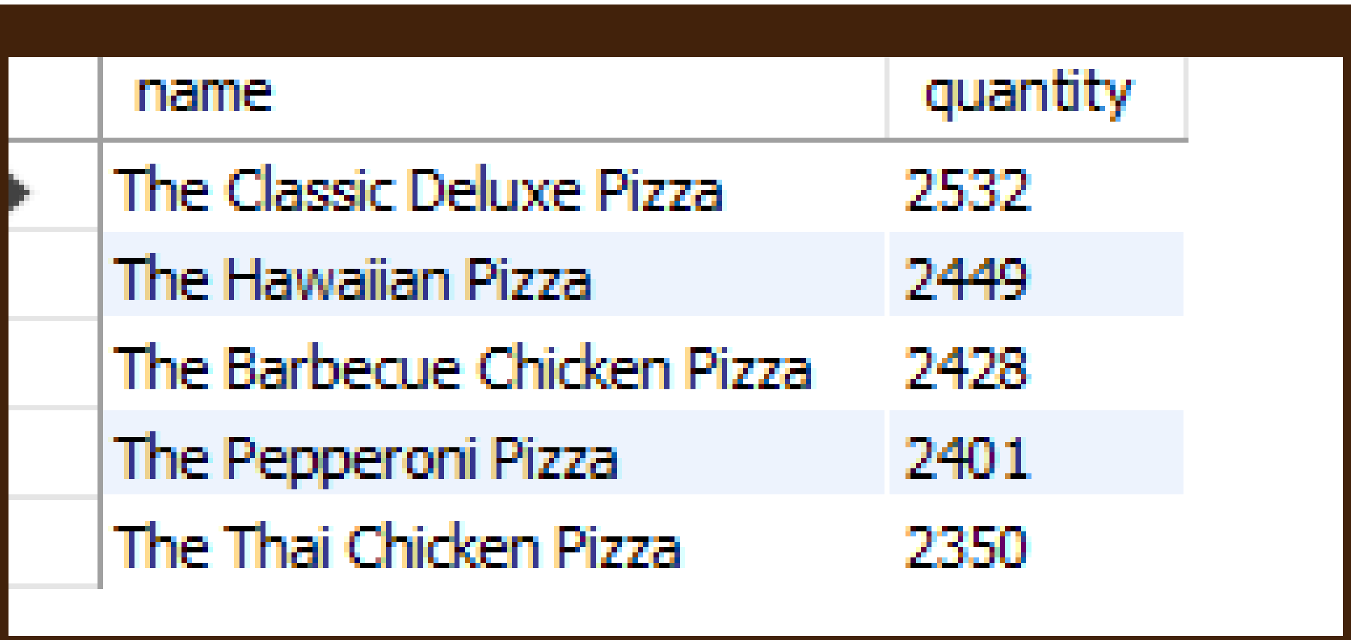


size	order_count
L	18714

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

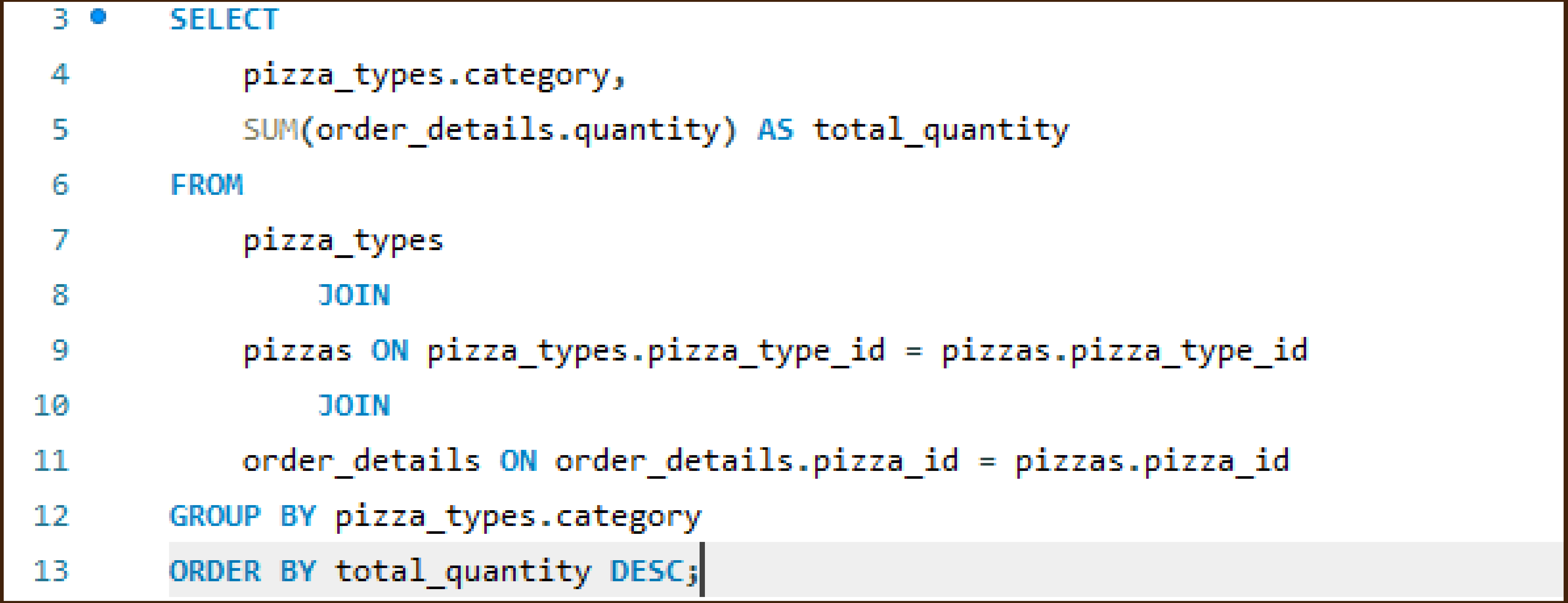


```sql
SELECT
    pizza_types.name, SUM(order_details.quantity) AS quantity
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY quantity DESC
LIMIT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

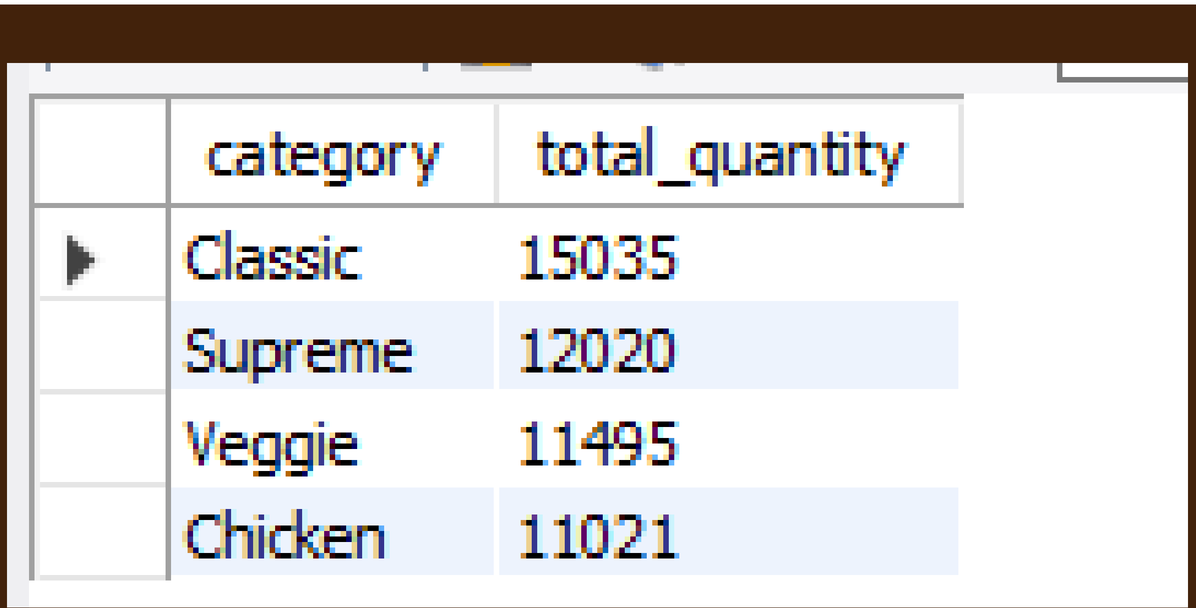


name	quantity
The Classic Deluxe Pizza	2532
The Hawaiian Pizza	2449
The Barbecue Chicken Pizza	2428
The Pepperoni Pizza	2401
The Thai Chicken Pizza	2350

FIND THE TOTAL QUANTITY OF EACH PIZZA CATEGORY ORDERED.

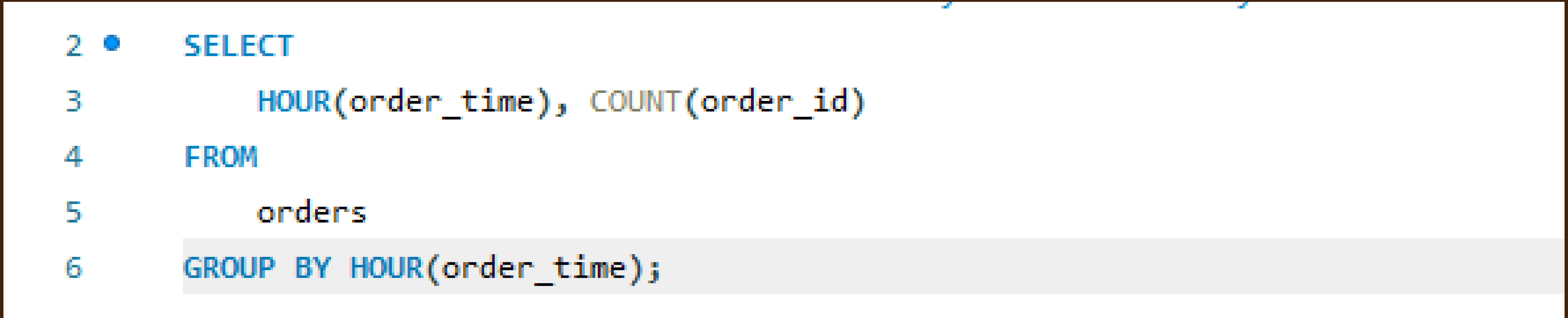


```sql
SELECT
    pizza_types.category,
    SUM(order_details.quantity) AS total_quantity
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
ORDER BY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

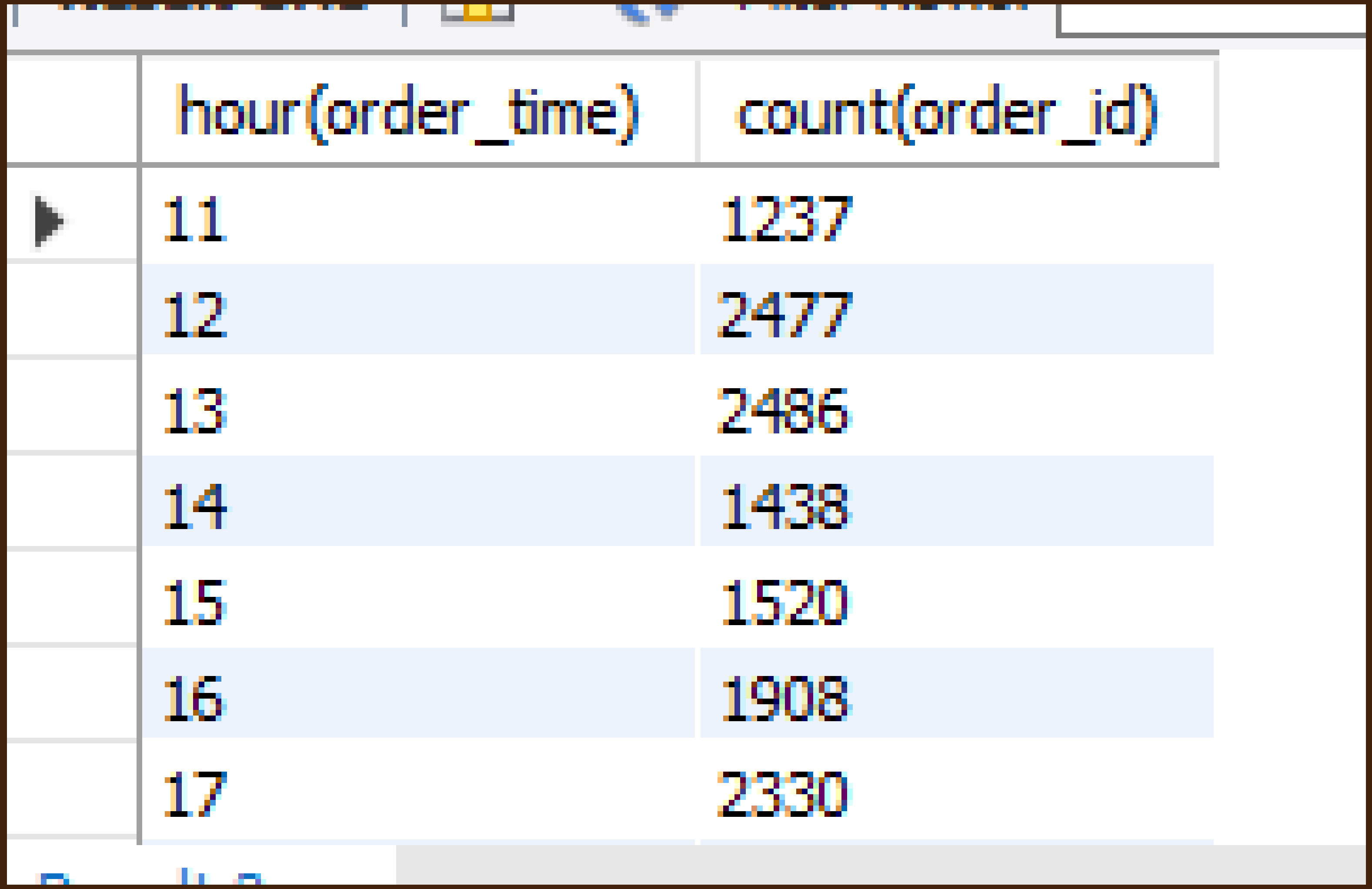


category	total_quantity
Classic	15035
Supreme	12020
Veggie	11495
Chicken	11021

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.

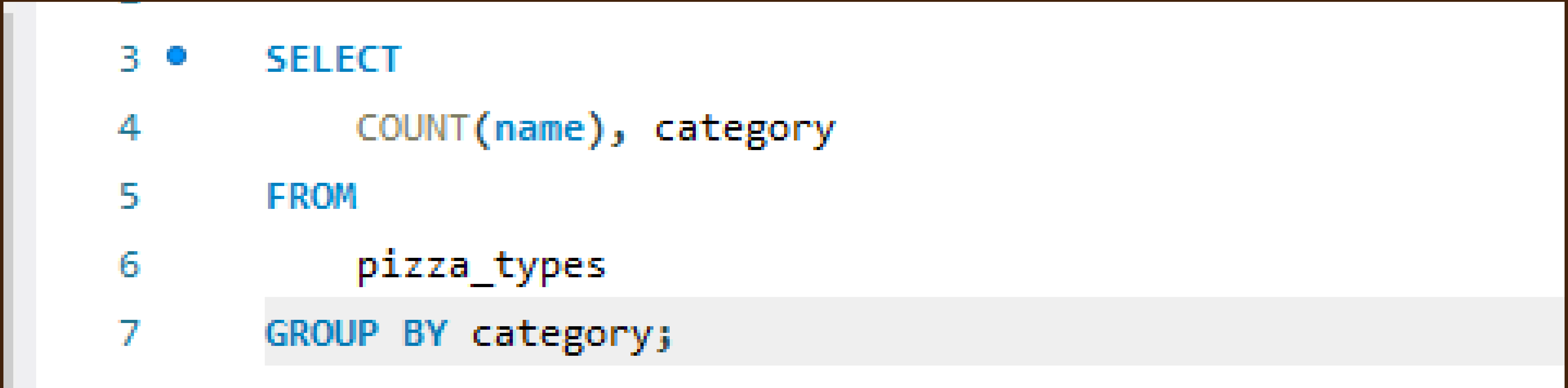


```
SELECT
    HOUR(order_time), COUNT(order_id)
FROM
    orders
GROUP BY HOUR(order_time);
```

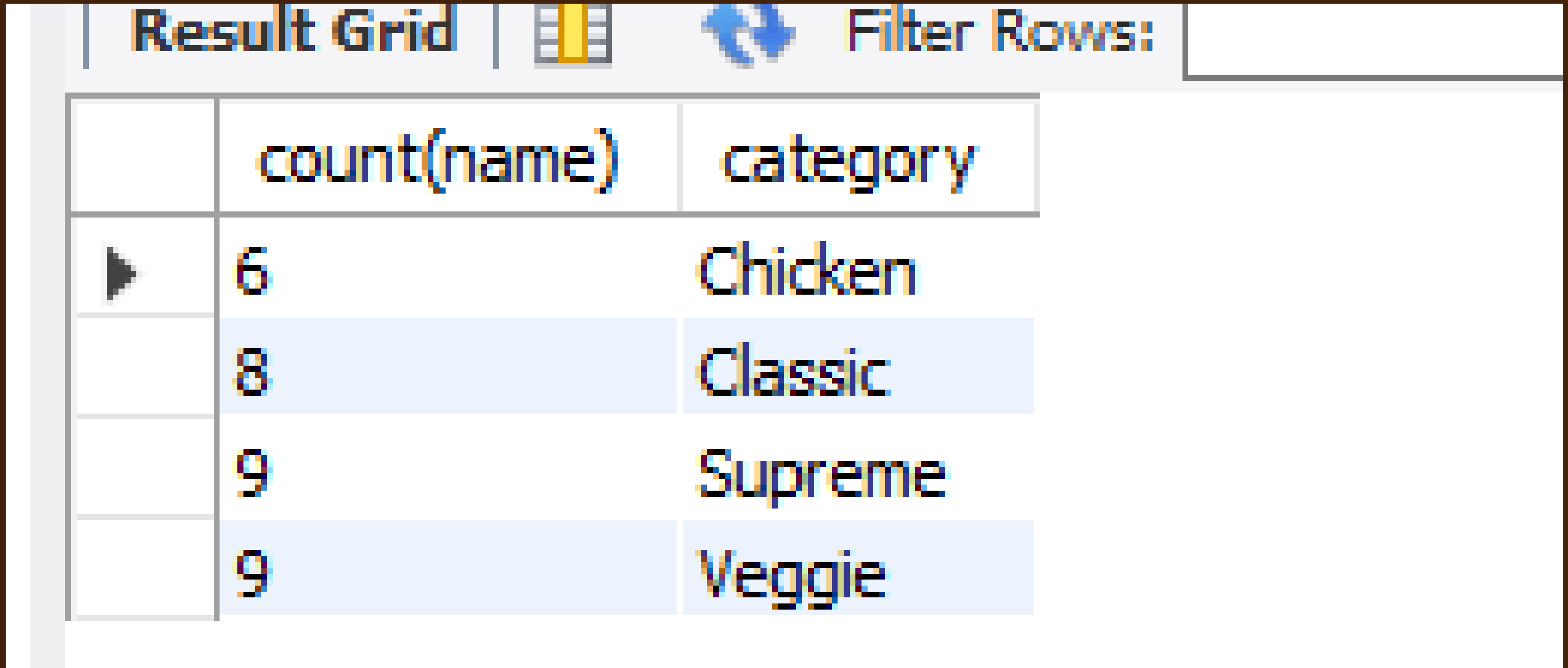


hour(order_time)	count(order_id)
11	1237
12	2477
13	2486
14	1438
15	1520
16	1908
17	2330

FIND THE CATEGORY-WISE DISTRIBUTION OF PIZZAS.

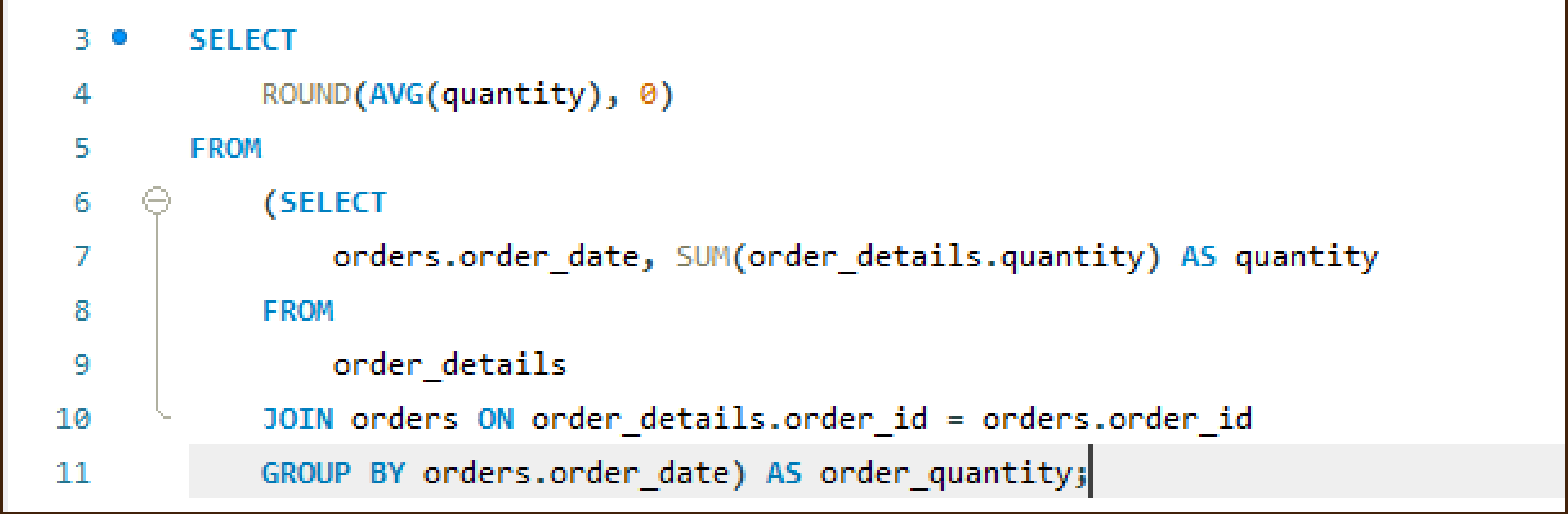


```sql
SELECT
    COUNT(name), category
FROM
    pizza_types
GROUP BY category;
```

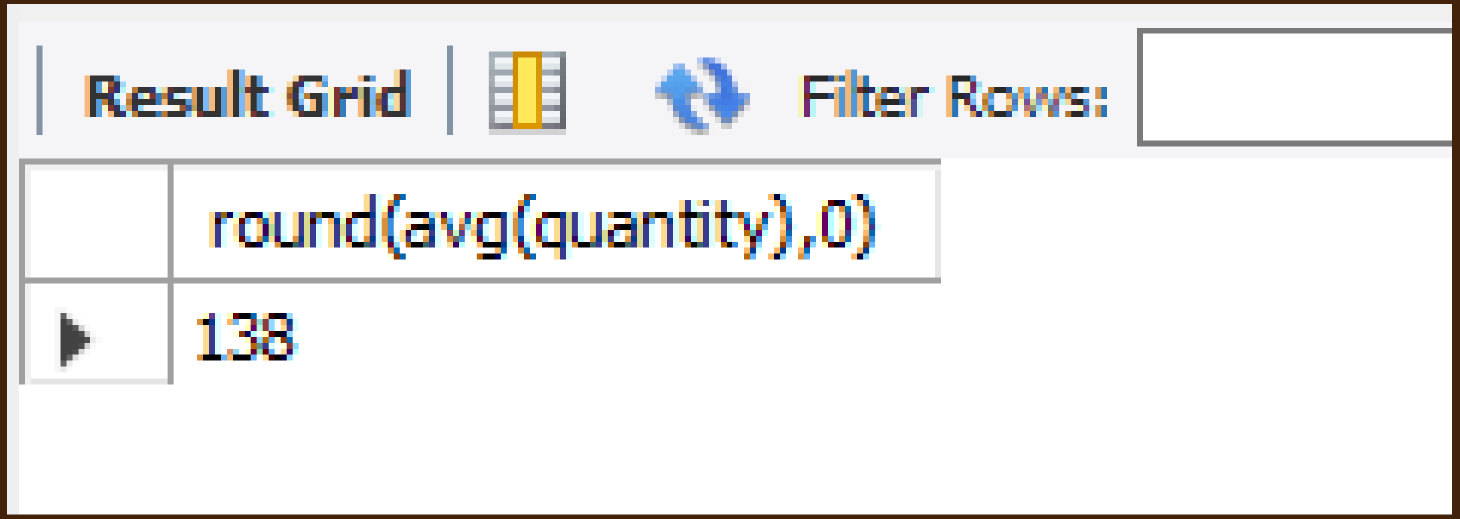


count(name)	category
6	Chicken
8	Classic
9	Supreme
9	Veggie

CALCULATE THE AVERAGE NUMBER OF PIZZAS ORDERED PER DAY.

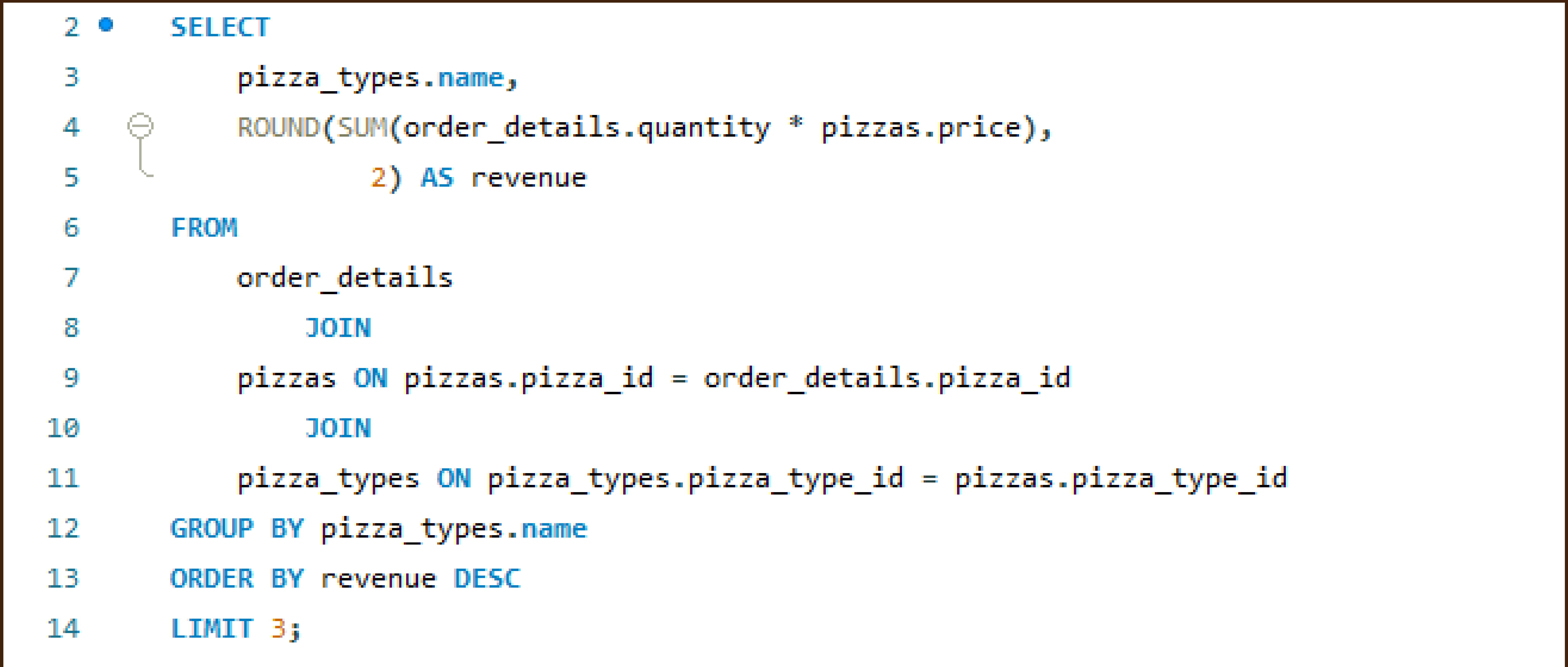


```sql
SELECT
    ROUND(AVG(quantity), 0)
FROM
    (SELECT
        orders.order_date, SUM(order_details.quantity) AS quantity
    FROM
        order_details
    JOIN orders ON order_details.order_id = orders.order_id
    GROUP BY orders.order_date) AS order_quantity;
```

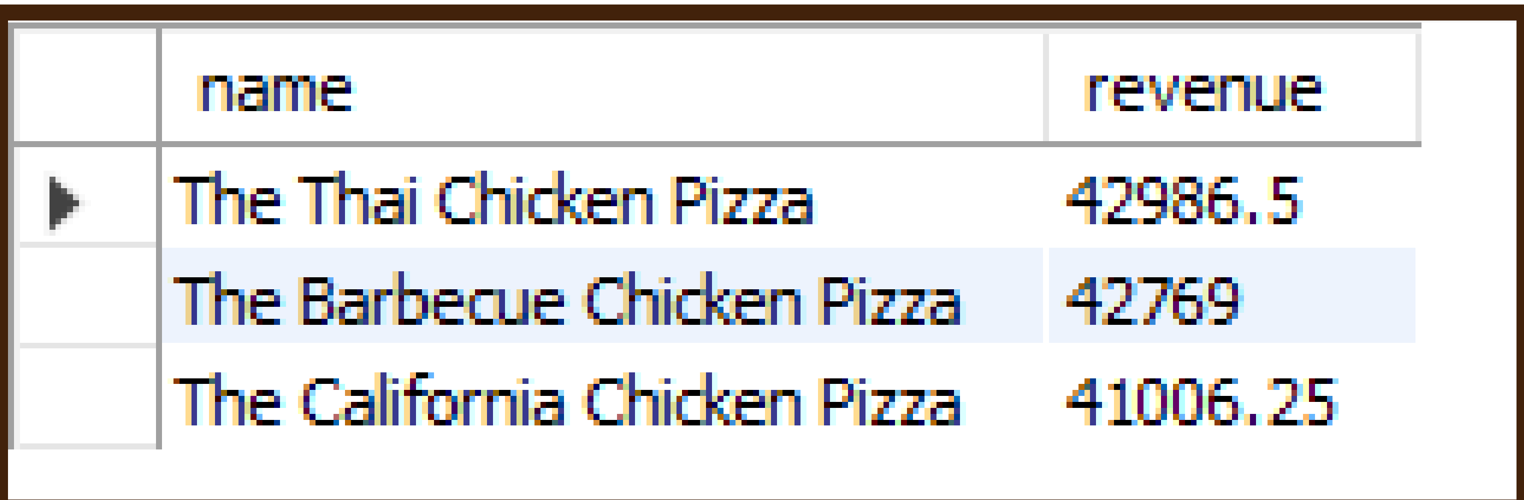


```
round(avg(quantity),0)
138
```

DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE.

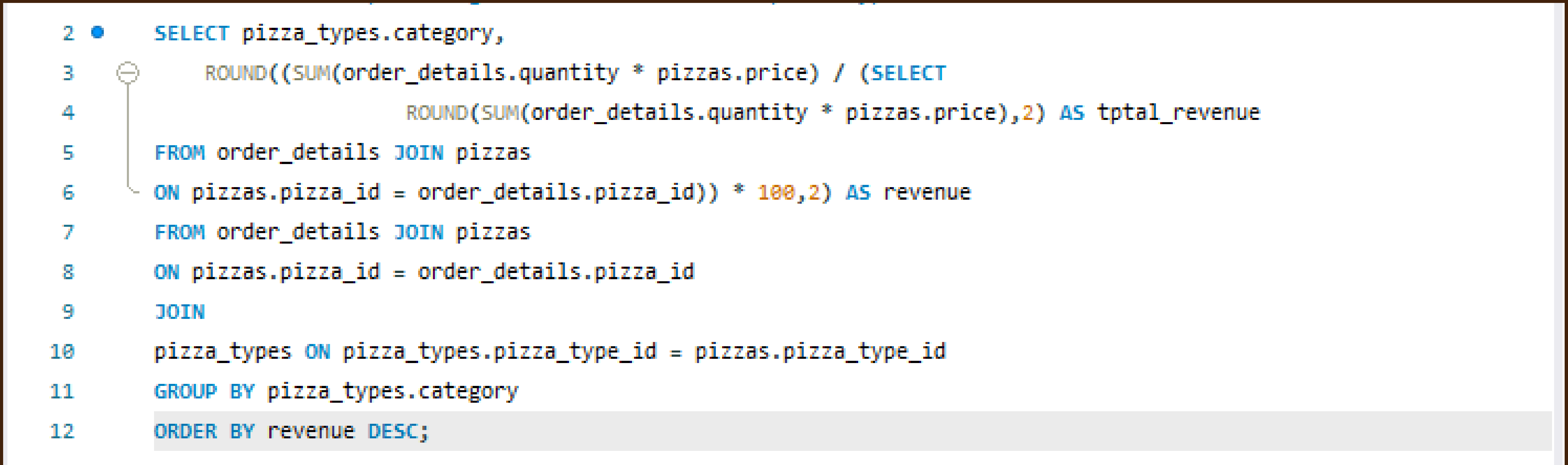


```sql
SELECT
    pizza_types.name,
    ROUND(SUM(order_details.quantity * pizzas.price),
            2) AS revenue
FROM
    order_details
        JOIN
    pizzas ON pizzas.pizza_id = order_details.pizza_id
        JOIN
    pizza_types ON pizza_types.pizza_type_id = pizzas.pizza_type_id
GROUP BY pizza_types.name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

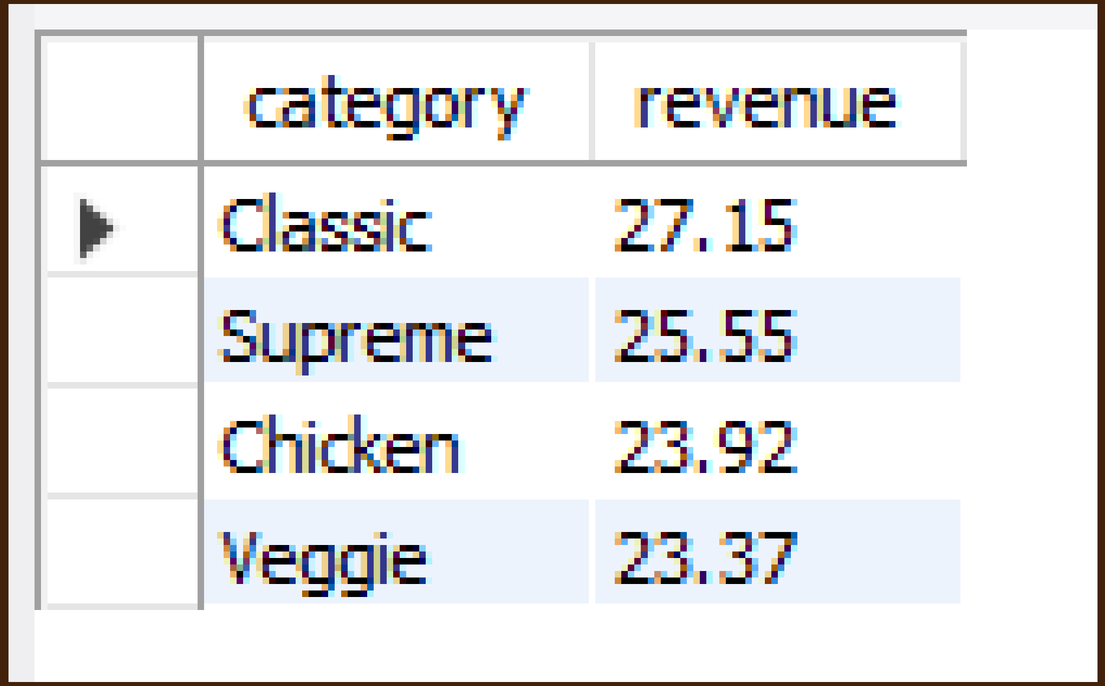


name	revenue
The Thai Chicken Pizza	42986.5
The Barbecue Chicken Pizza	42769
The California Chicken Pizza	41006.25

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

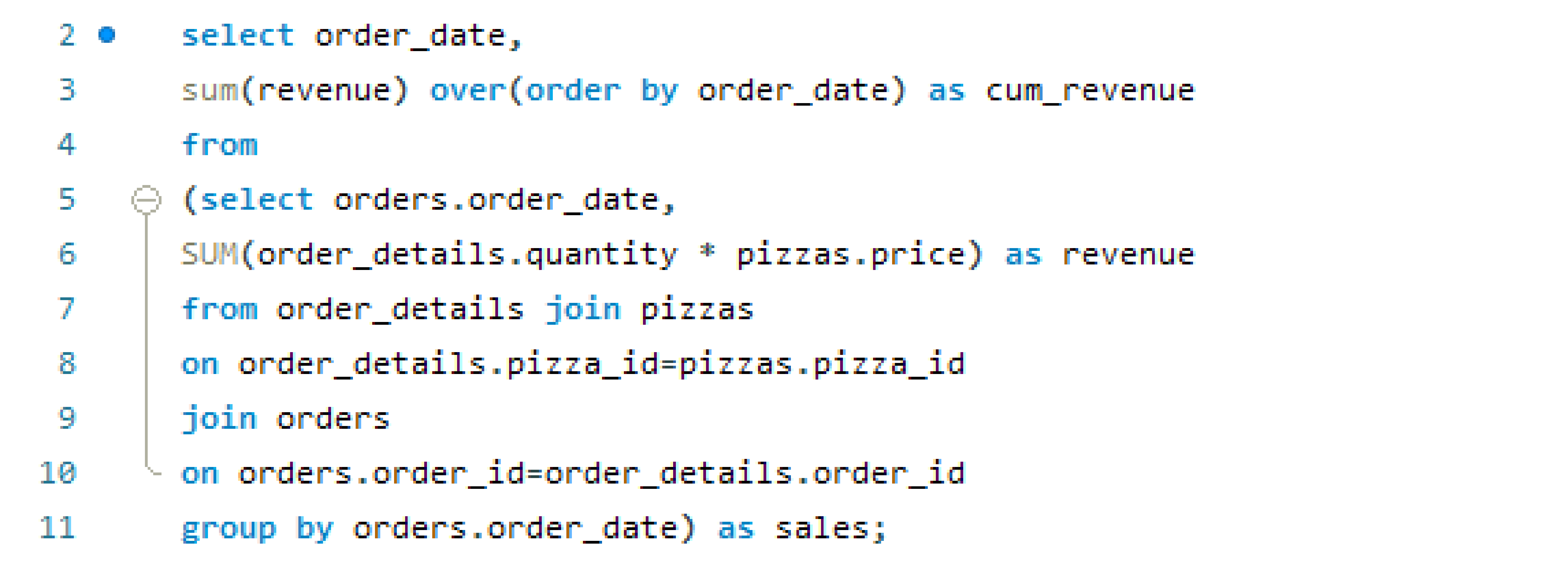


```sql
SELECT pizza_types.category,
    ROUND((SUM(order_details.quantity * pizzas.price) / (SELECT
                    ROUND(SUM(order_details.quantity * pizzas.price),2) AS tptal_revenue
FROM order_details JOIN pizzas
ON pizzas.pizza_id = order_details.pizza_id)) * 100,2) AS revenue
FROM order_details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

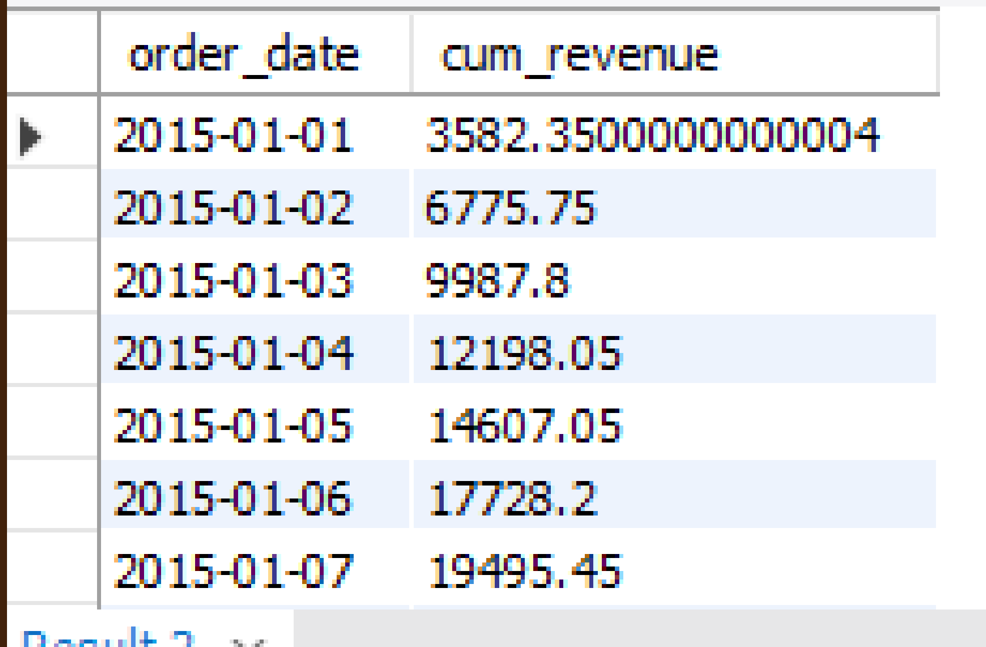


	category	revenue
▶	Classic	27.15
	Supreme	25.55
	Chicken	23.92
	Veggie	23.37

ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME.

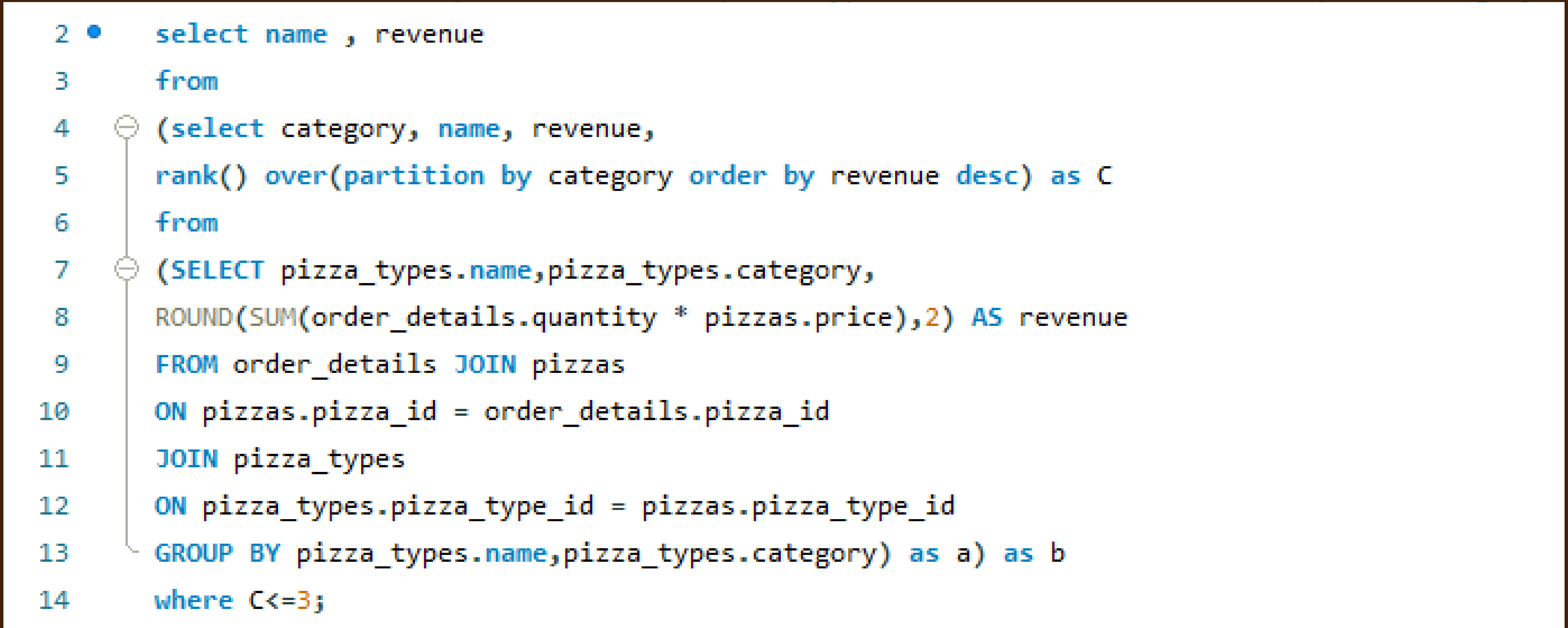


```sql
select order_date,
sum(revenue) over(order by order_date) as cum_revenue
from
(select orders.order_date,
SUM(order_details.quantity * pizzas.price) as revenue
from order_details join pizzas
on order_details.pizza_id=pizzas.pizza_id
join orders
on orders.order_id=order_details.order_id
group by orders.order_date) as sales;
```

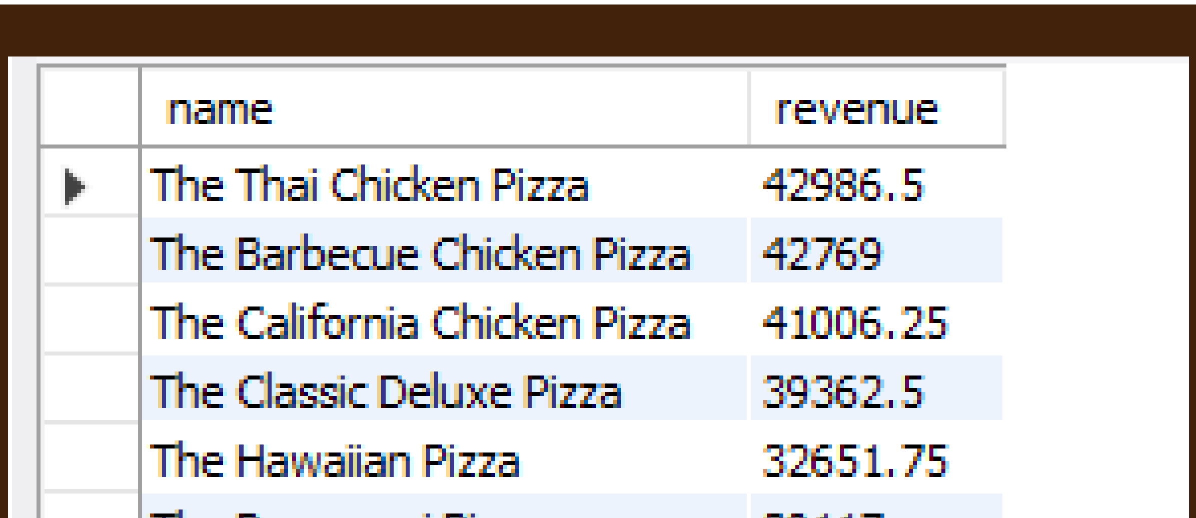


order_date	cum_revenue
2015-01-01	3582.3500000000004
2015-01-02	6775.75
2015-01-03	9987.8
2015-01-04	12198.05
2015-01-05	14607.05
2015-01-06	17728.2
2015-01-07	19495.45

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 C
from
(SELECT pizza_types.name,pizza_types.category,
ROUND(SUM(order_details.quantity * pizzas.price),2) 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,pizza_types.category) as a) as b
where C<=3;
```



name	revenue
The Thai Chicken Pizza	42986.5
The Barbecue Chicken Pizza	42769
The California Chicken Pizza	41006.25
The Classic Deluxe Pizza	39362.5
The Hawaiian Pizza	32651.75

SUMMARY

THE DATASET COMPRISED A TOTAL OF 21,350 ORDERS PLACED FOR PIZZA, REFLECTING THE SUBSTANTIAL DEMAND FOR THIS POPULAR FOOD ITEM.

THE REVENUE GENERATED FROM THESE ORDERS AMOUNTED TO $818,515, INDICATING THE SIGNIFICANT ECONOMIC IMPACT OF PIZZA SALES.

THE CLASSIC DELUXE PIZZA EMERGED AS THE MOST ORDERED PIZZA VARIETY, SUGG ESTING ITS POPULARITY AMONG CUSTOMERS.

ANALYSIS OF ORDER TIMESTAMPS REVEALED THAT THE HIGHEST NUMBER OF ORDERS WAS PLACED AT 1:00 PM, HIGHLIGHTING A PEAK IN PIZZA CONSUMPTION DURING LUNCH HOURS.

ON AVERAGE, 138 PIZZAS WERE SOLD PER DAY, PROVIDING INSIGHTS INTO THE DAILY CONSUMPTION PATTERNS AND OVERALL MARKET DEMAND.

THE THAI CHICKEN PIZZA EMERGED AS THE TOP REVENUE GENERATOR, INDICATING ITS PROFITABILITY AND POPULARITY AMONG CUSTOMERS.

Thank You