PIZZA SALES PROJECT

- Renuka Joshi

DESCRIPTION

This project analyzes pizza sales data to uncover insights into customer preferences, sales performance, and revenue trends. Using SQL, it explores order volumes, popular pizza types and sizes, and peak ordering times. The analysis also highlights top revenue-generating items and category-wise performance. These insights support smarter decisions in marketing, inventory, and menu planning.

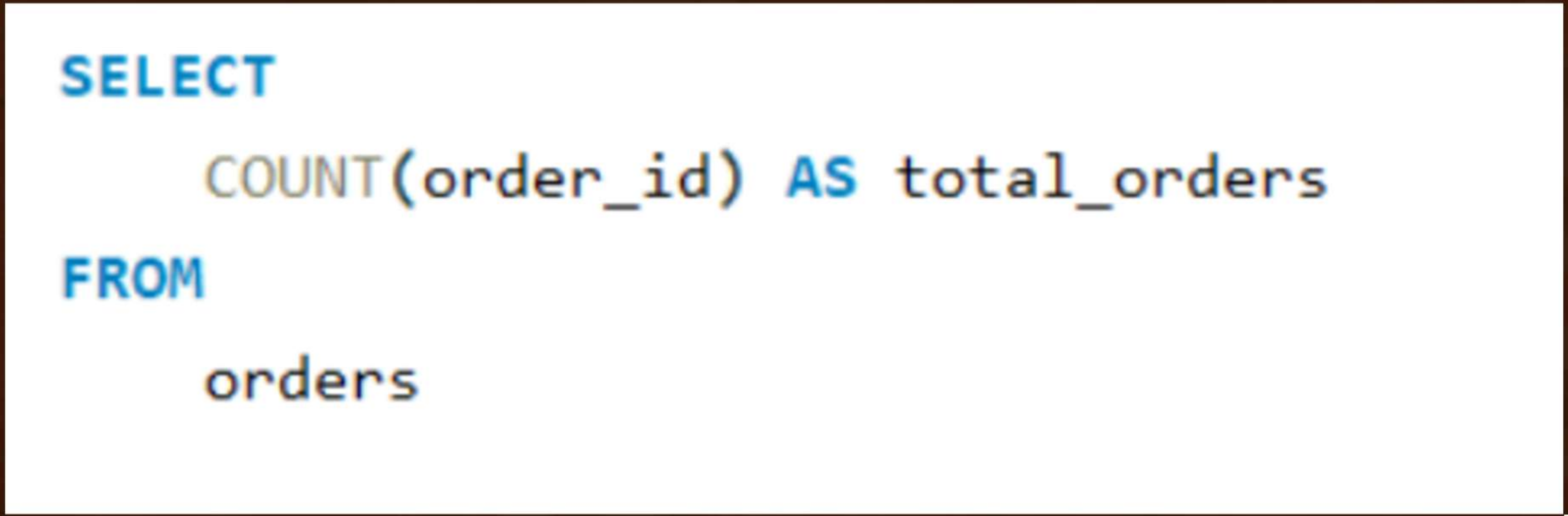
DATASET DETAILS

The analysis is based on four interconnected CSV files:

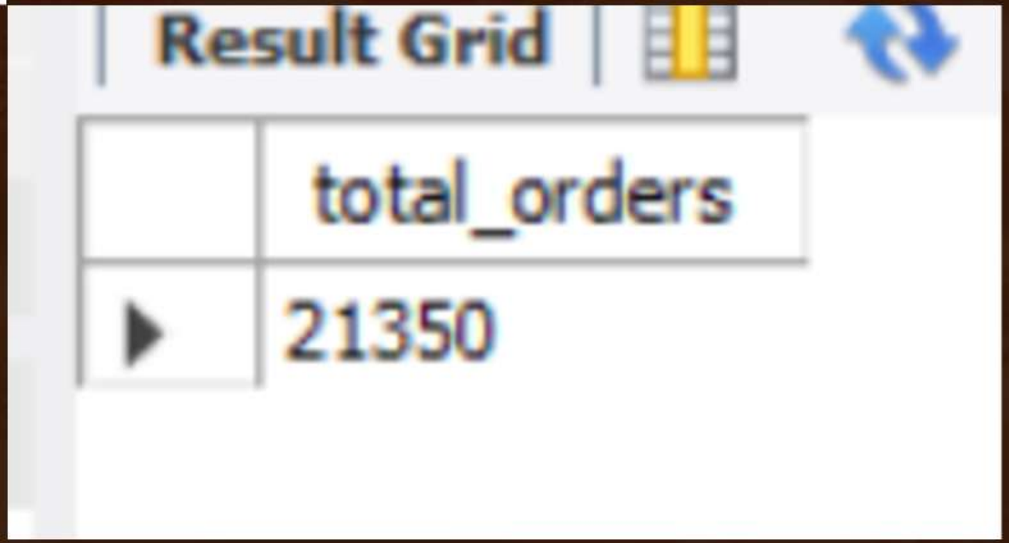
1. orders.csv
   - Contains order-level data including order_id, date, and time of purchase.
2. order_details.csv
   - Line-item details for each order, with order_details_id, order_id, pizza_id, and quantity.
3. pizzas.csv
   - Provides price and size details for each pizza, with fields like pizza_id, pizza_type_id, size, and price.
4. pizza_types.csv
   - Describes the types of pizzas, including pizza_type_id, name, category (e.g., Classic, Veggie, Chicken), and ingredients.

These datasets together offer a comprehensive view of customer orders, product information, and pricing, enabling in-depth sales and performance analysis.

Retrieve the total number of orders placed



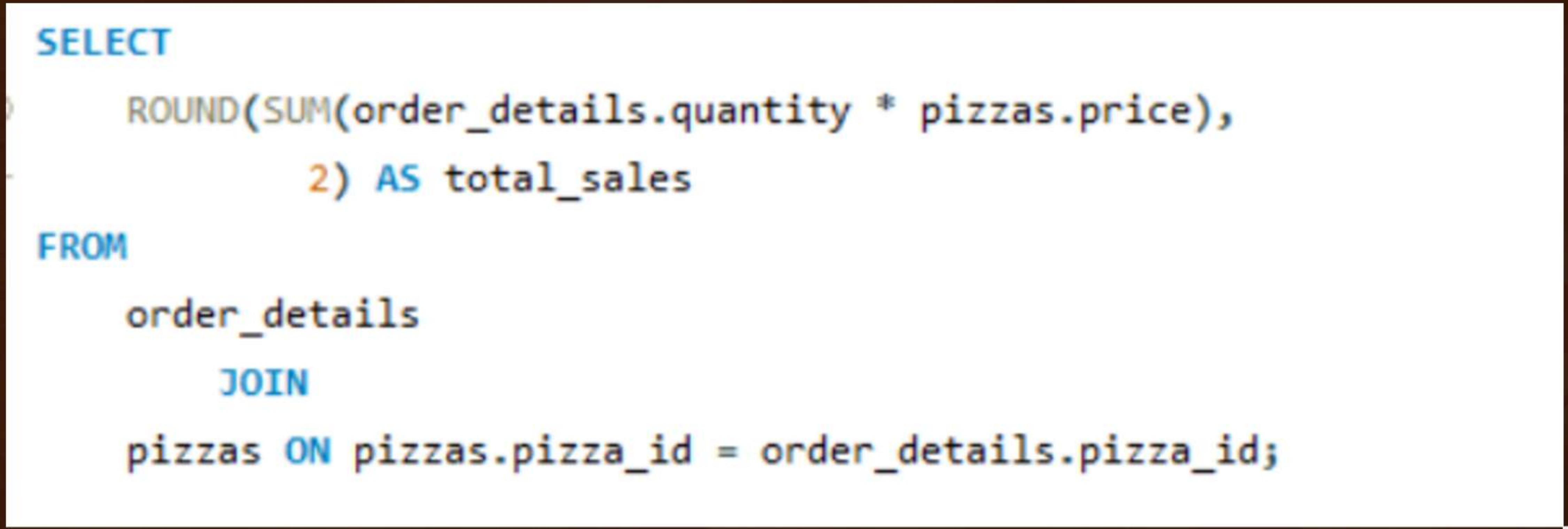
```sql
SELECT
    COUNT(order_id) AS total_orders
FROM
    orders
```



	total_orders
▶	21350

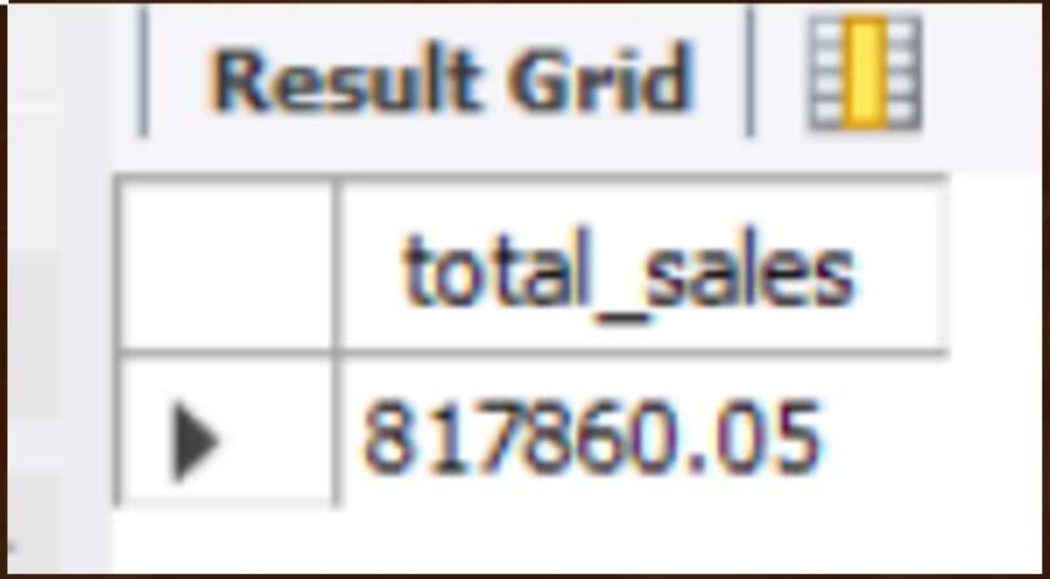
Insight: There are 21,350 unique orders placed.

Calculate the total revenue generated from pizza sales



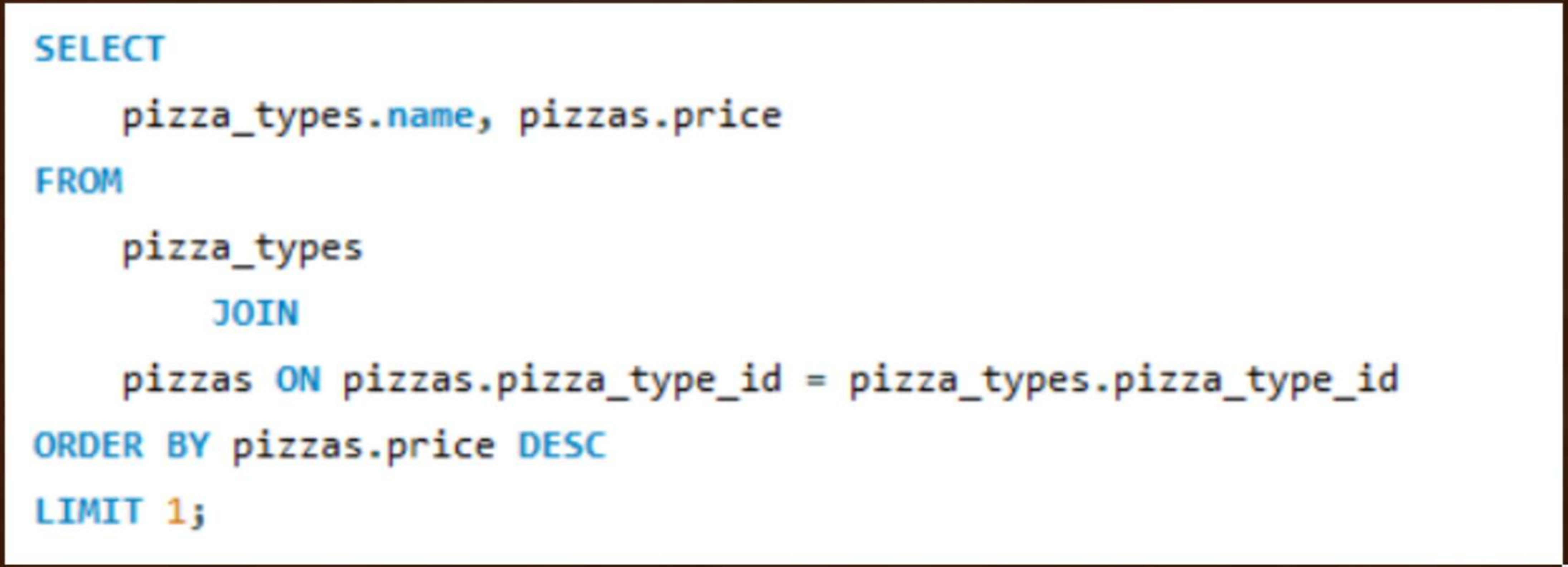
```sql
SELECT
    ROUND(SUM(order_details.quantity * pizzas.price),
            2) AS total_sales
FROM
    order_details
        JOIN
    pizzas ON pizzas.pizza_id = order_details.pizza_id;
```

Insight: Total revenue generated is $817,860.05.

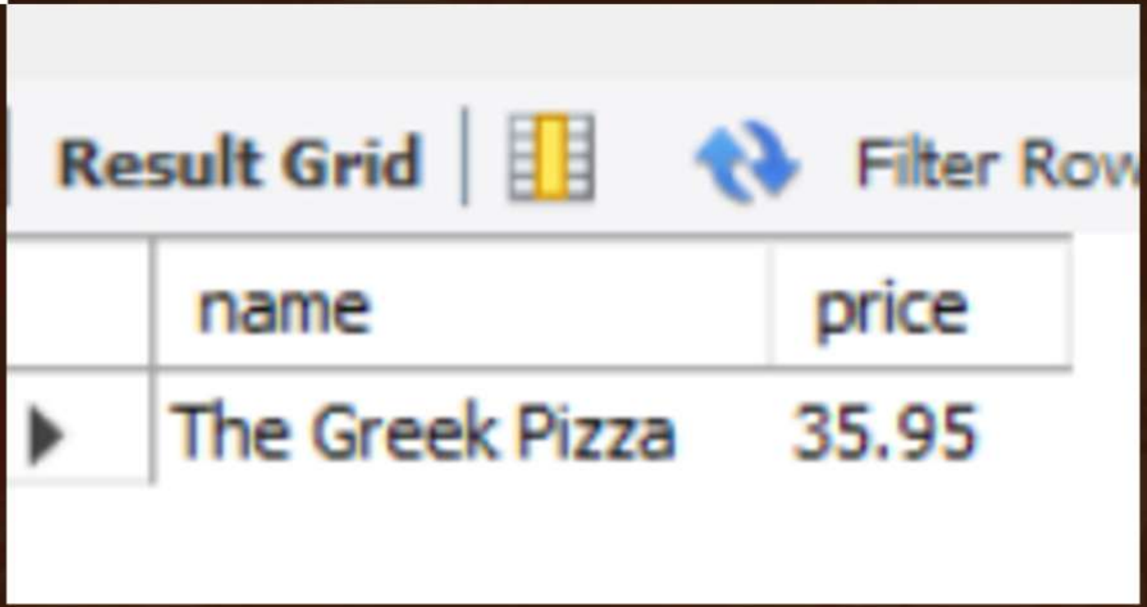


total_sales
817860.05

Identify the highest-priced pizza



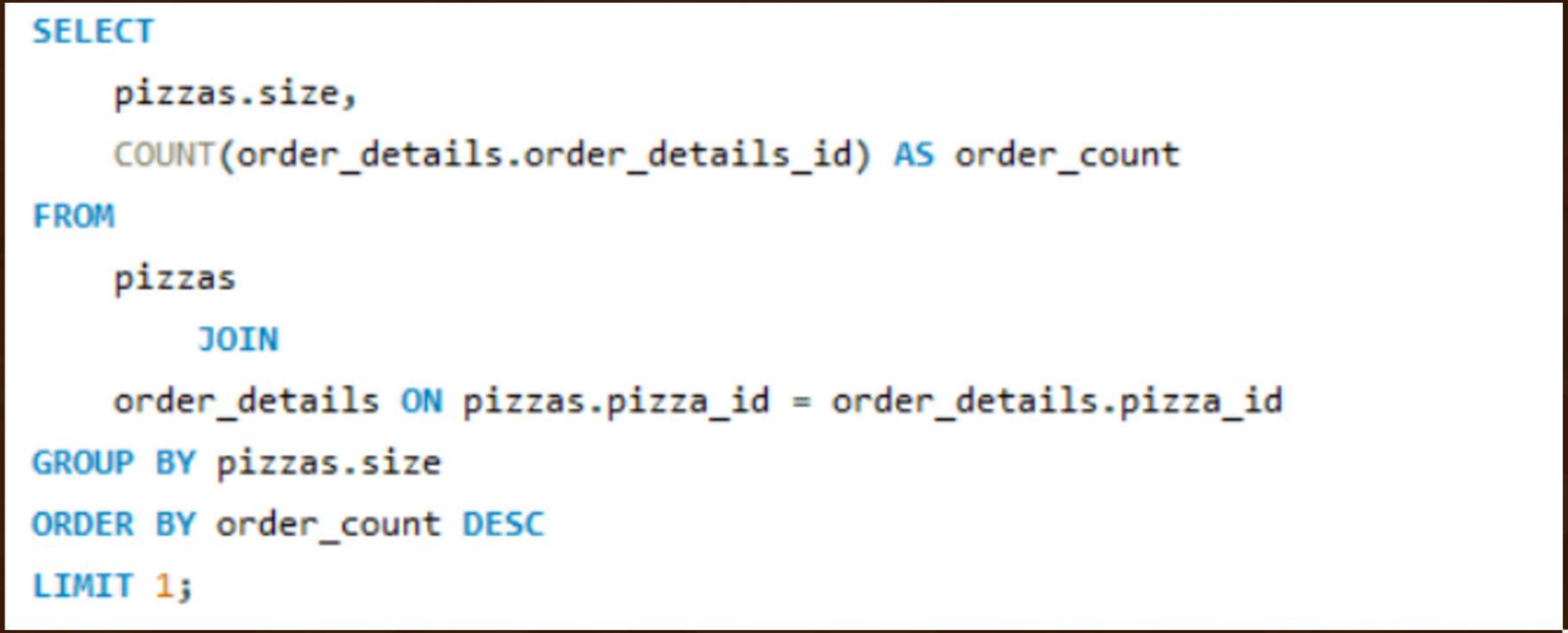
```
SELECT
    pizza_types.name, pizzas.price
FROM
    pizza_types
        JOIN
    pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
ORDER BY pizzas.price DESC
LIMIT 1;
```



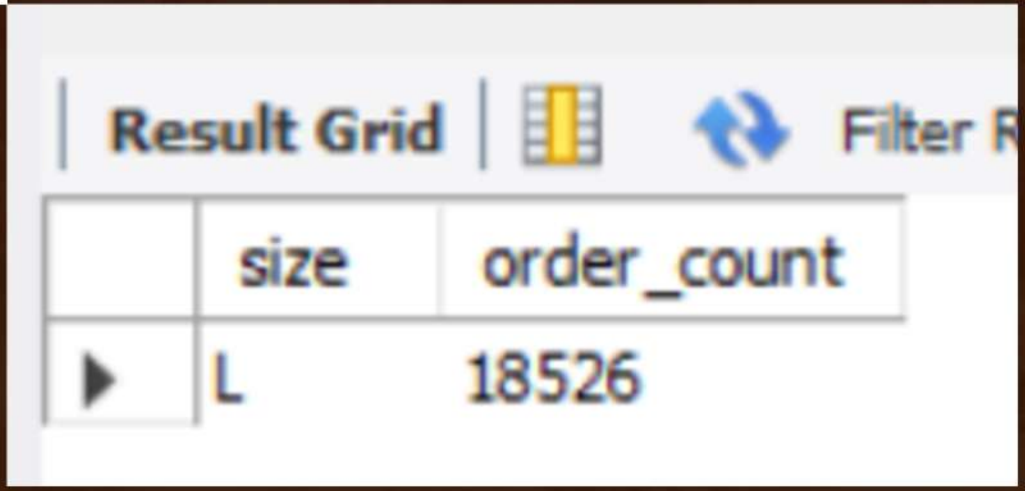
name	price
The Greek Pizza	35.95

Insight: The Greek Pizza (XXL) is the highest-priced at $35.95.

Identify the most common pizza size ordered



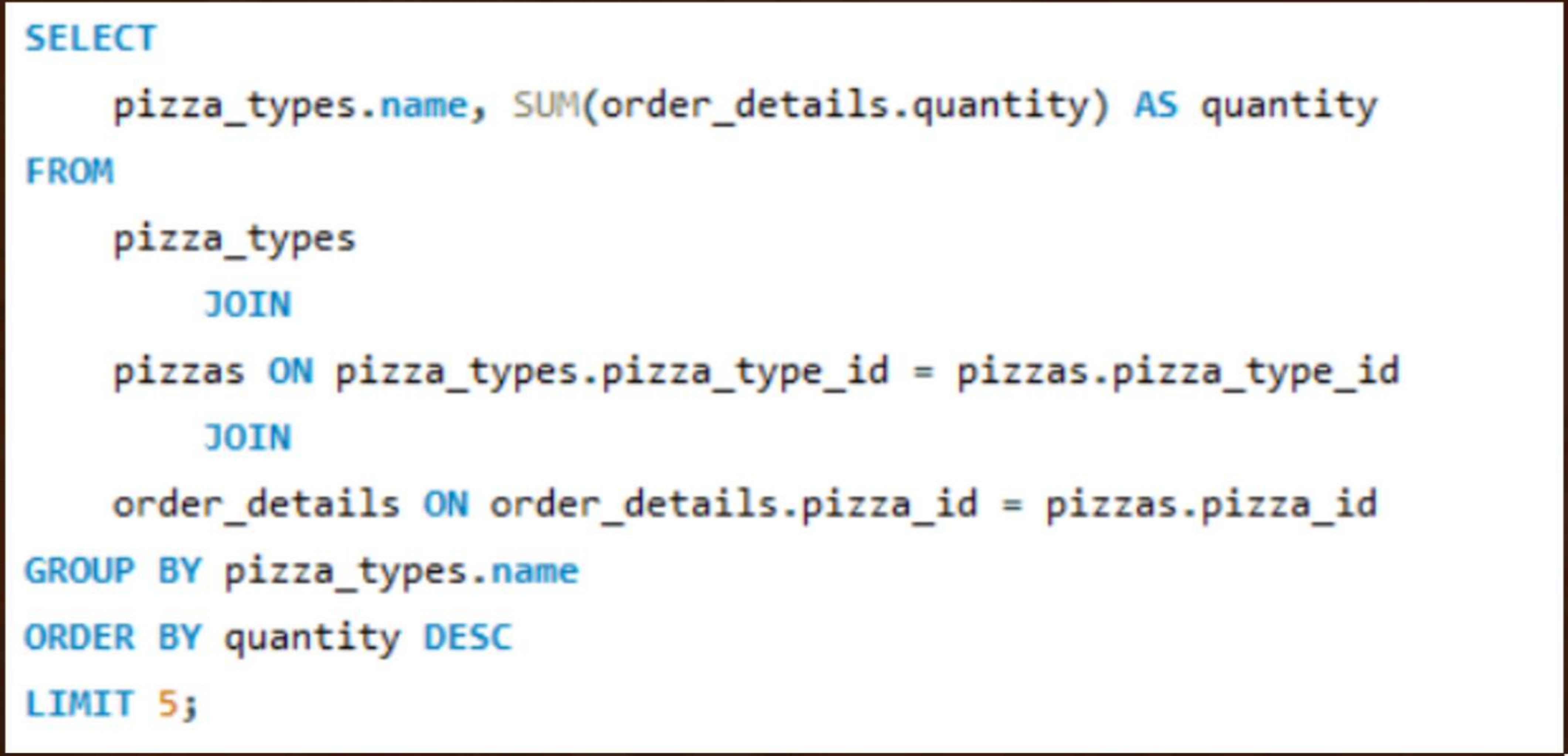
```sql
SELECT
    pizzas.size,
    COUNT(order_details.order_details_id) AS order_count
FROM
    pizzas
        JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY pizzas.size
ORDER BY order_count DESC
LIMIT 1;
```



size	order_count
L	18526

Insight: The Large (L) size is the most commonly ordered.

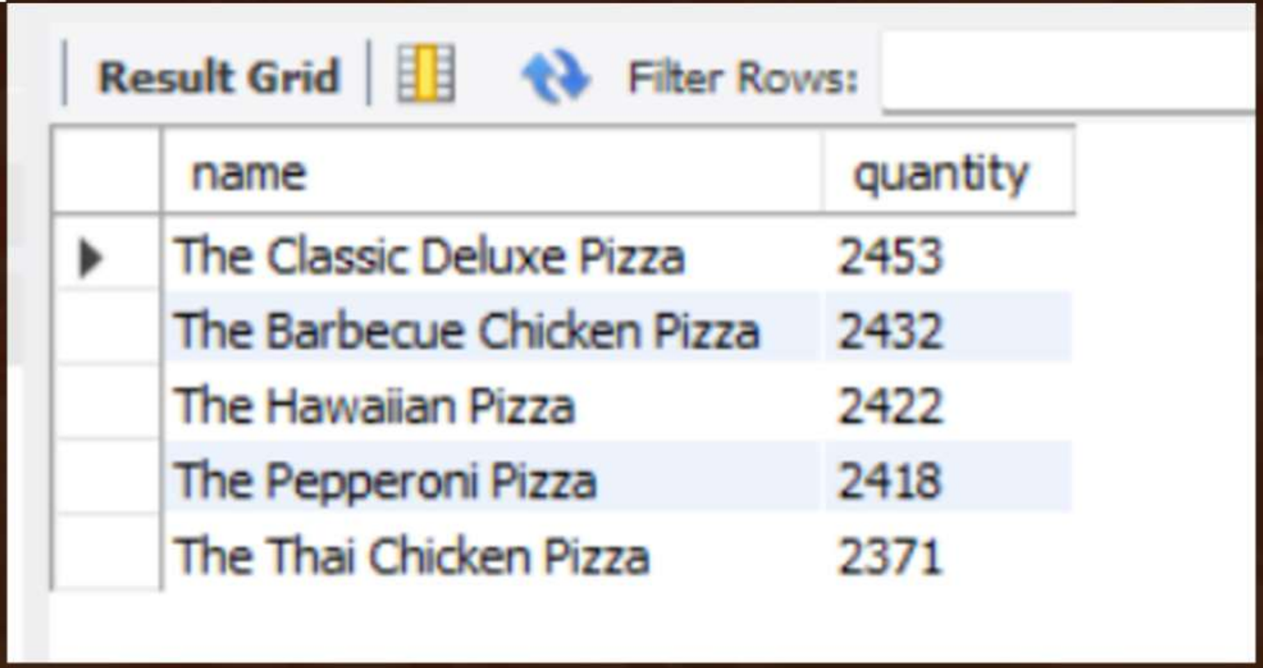
List the top 5 most ordered pizza types along with their quantities



```
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;
```

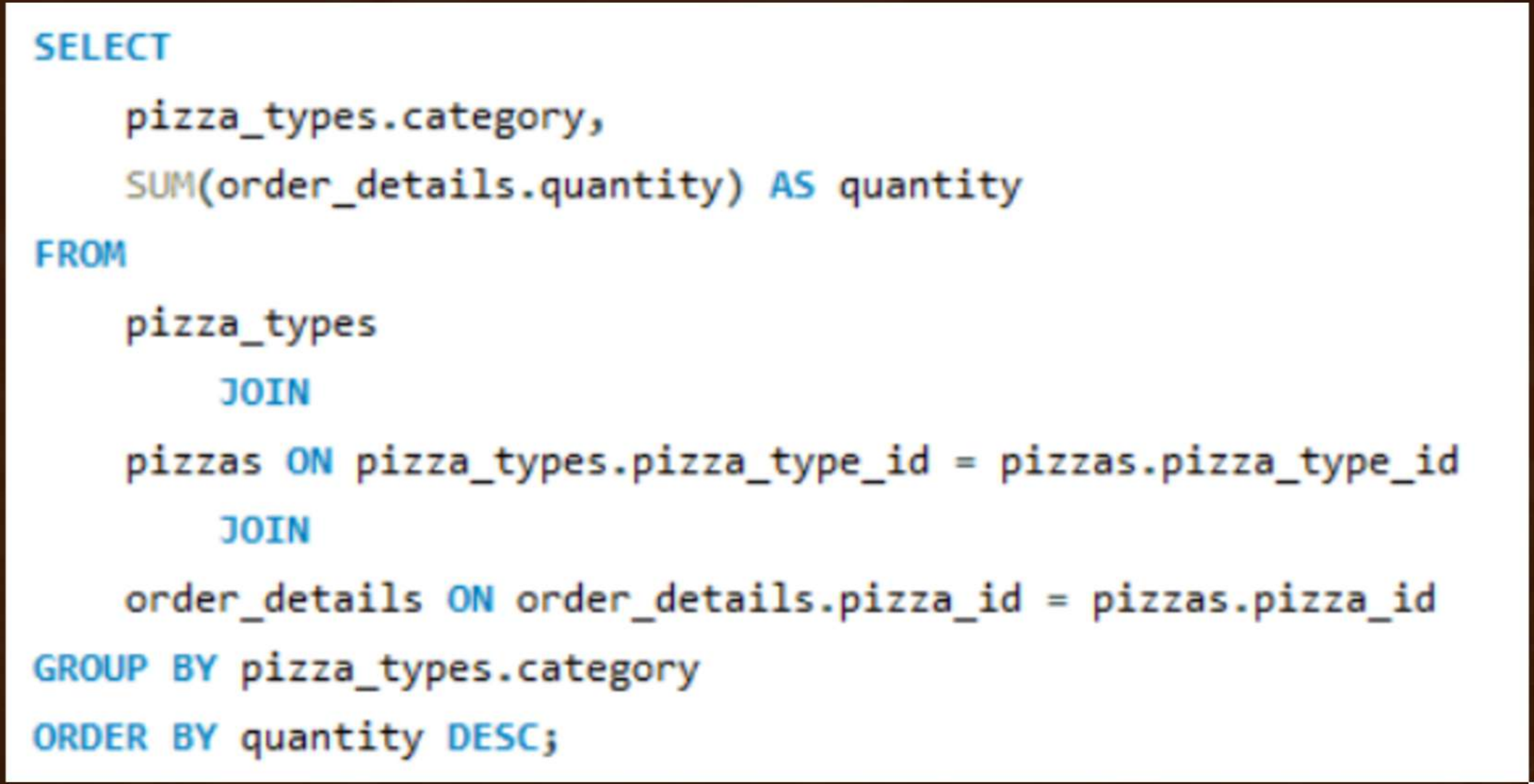
Insight: Top 5 most ordered pizzas include:

- Classic Deluxe
- BBQ Chicken
- Hawaiian
- Pepperoni
- Thai Chicken

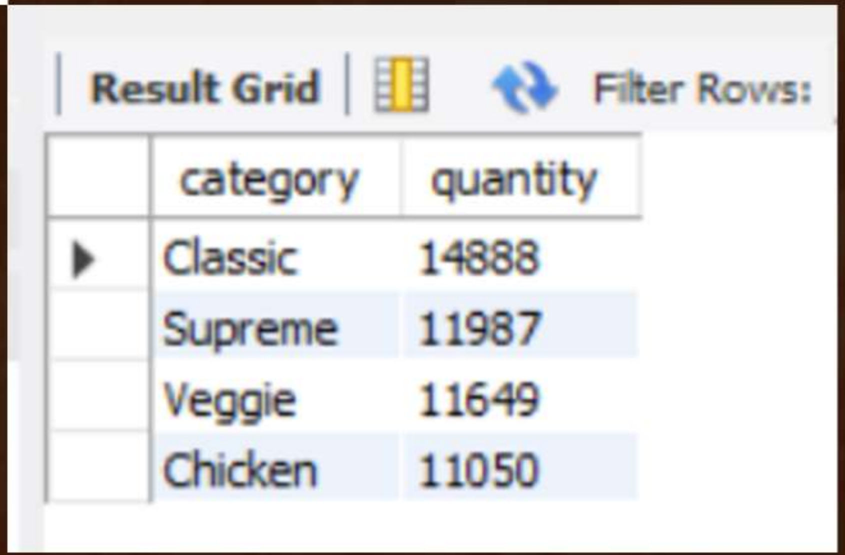


name	quantity
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	2422
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371

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



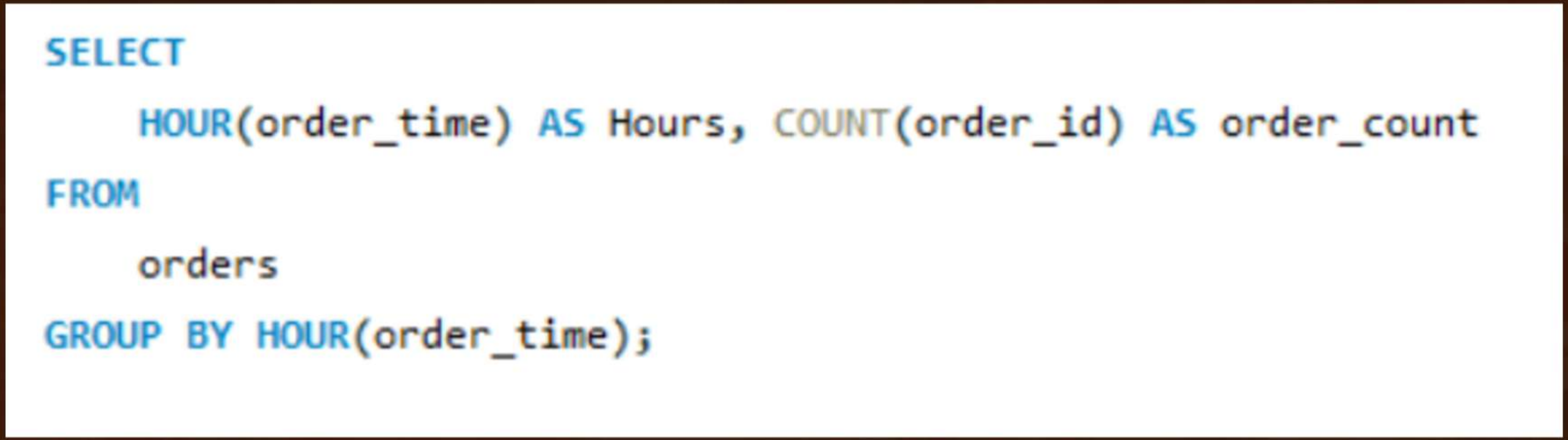
```
SELECT
    pizza_types.category,
    SUM(order_details.quantity) AS quantity
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
ORDER BY quantity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



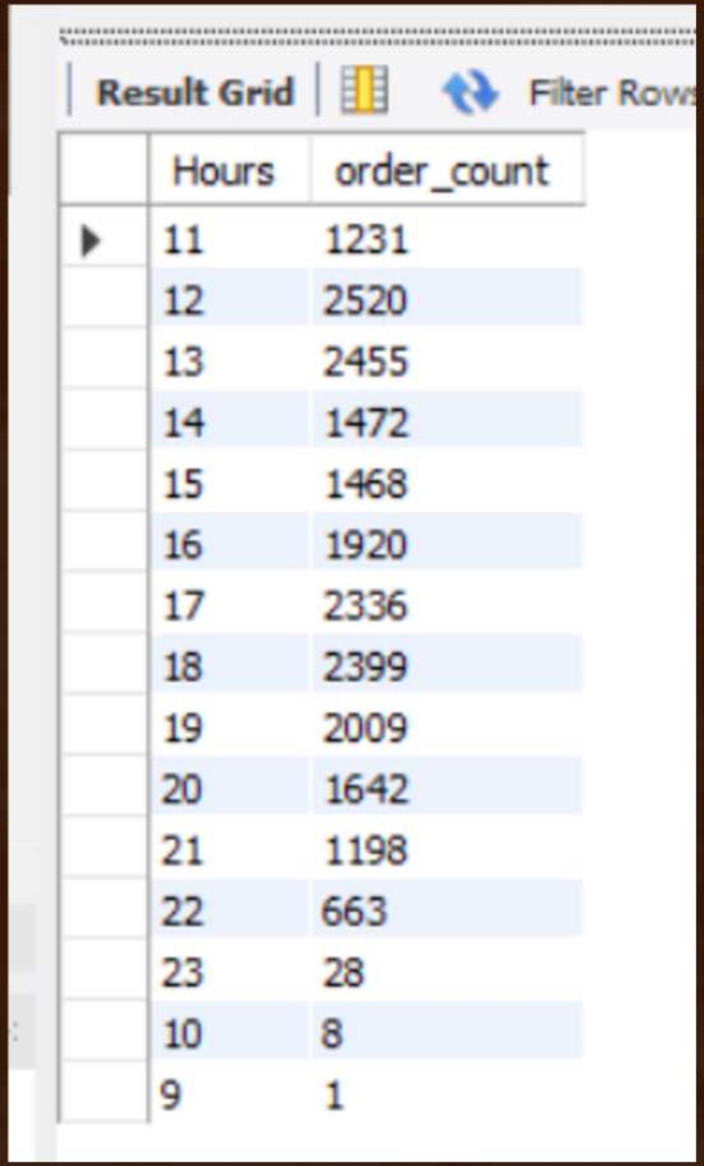
category	quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

Insight: Useful for analyzing which category (e.g. Chicken, Veggie, etc.) sells the most.

Determine the distribution of orders by hour of the day



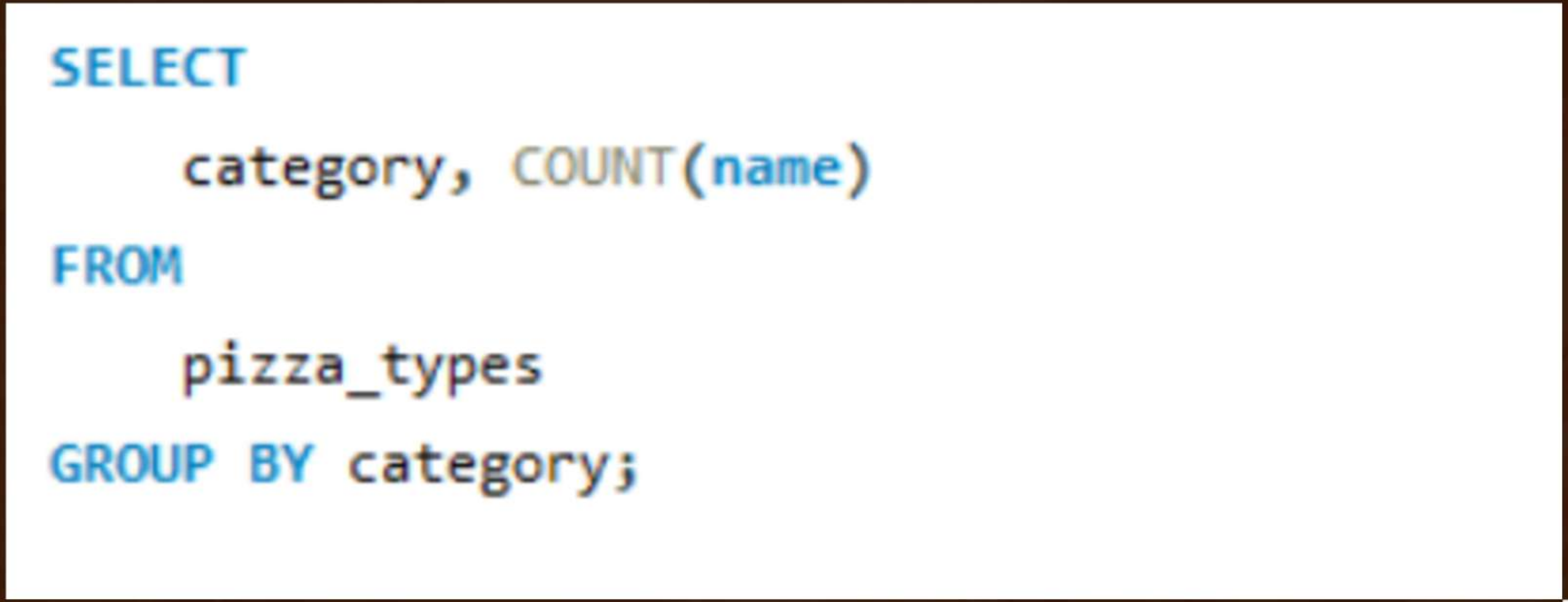
```
SELECT
    HOUR(order_time) AS Hours, COUNT(order_id) AS order_count
FROM
    orders
GROUP BY HOUR(order_time);
```



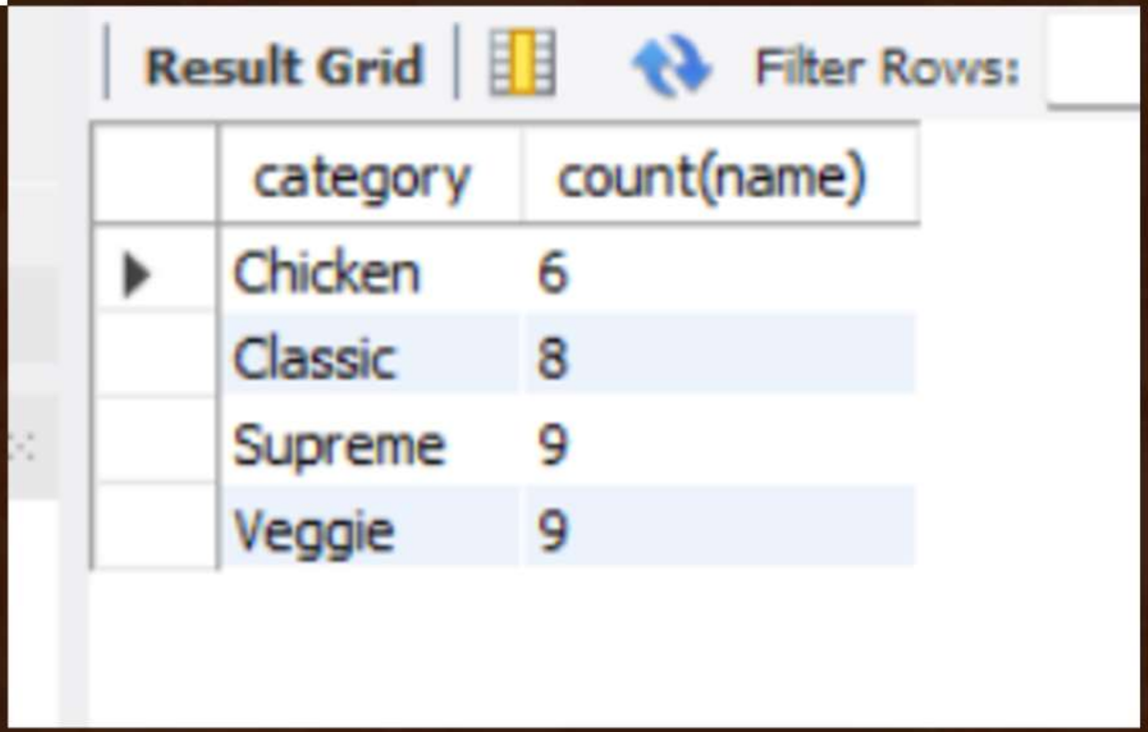
Hours	order_count
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642
21	1198
22	663
23	28
10	8
9	1

Insight: Reveals peak ordering hours, ideal for staffing and marketing.

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



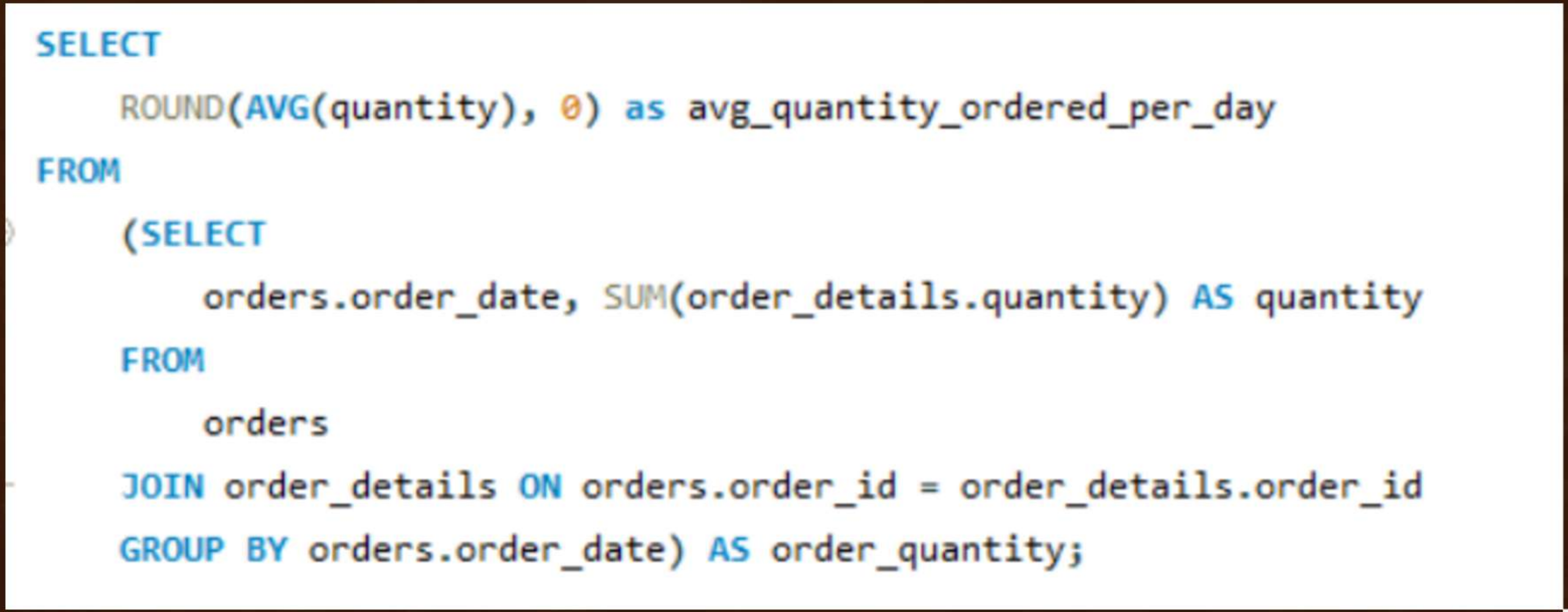
```
SELECT
    category, COUNT(name)
FROM
    pizza_types
GROUP BY category;
```



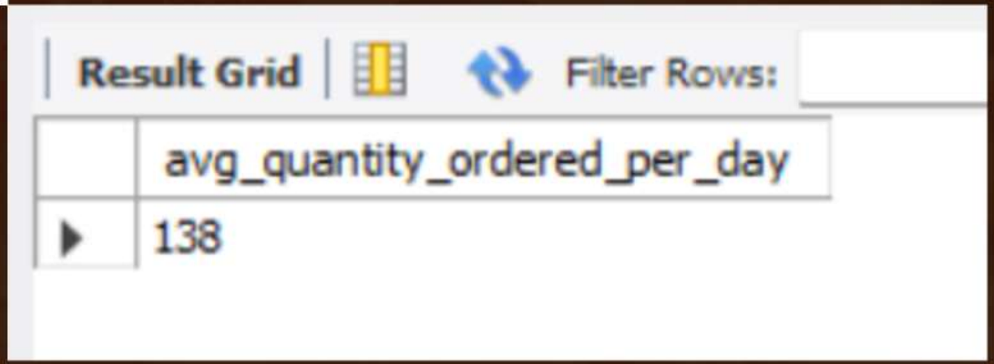
category	count(name)
Chicken	6
Classic	8
Supreme	9
Veggie	9

Insight: Shows the variety in each category (e.g., number of unique pizzas in "Meat" vs "Veggie").

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



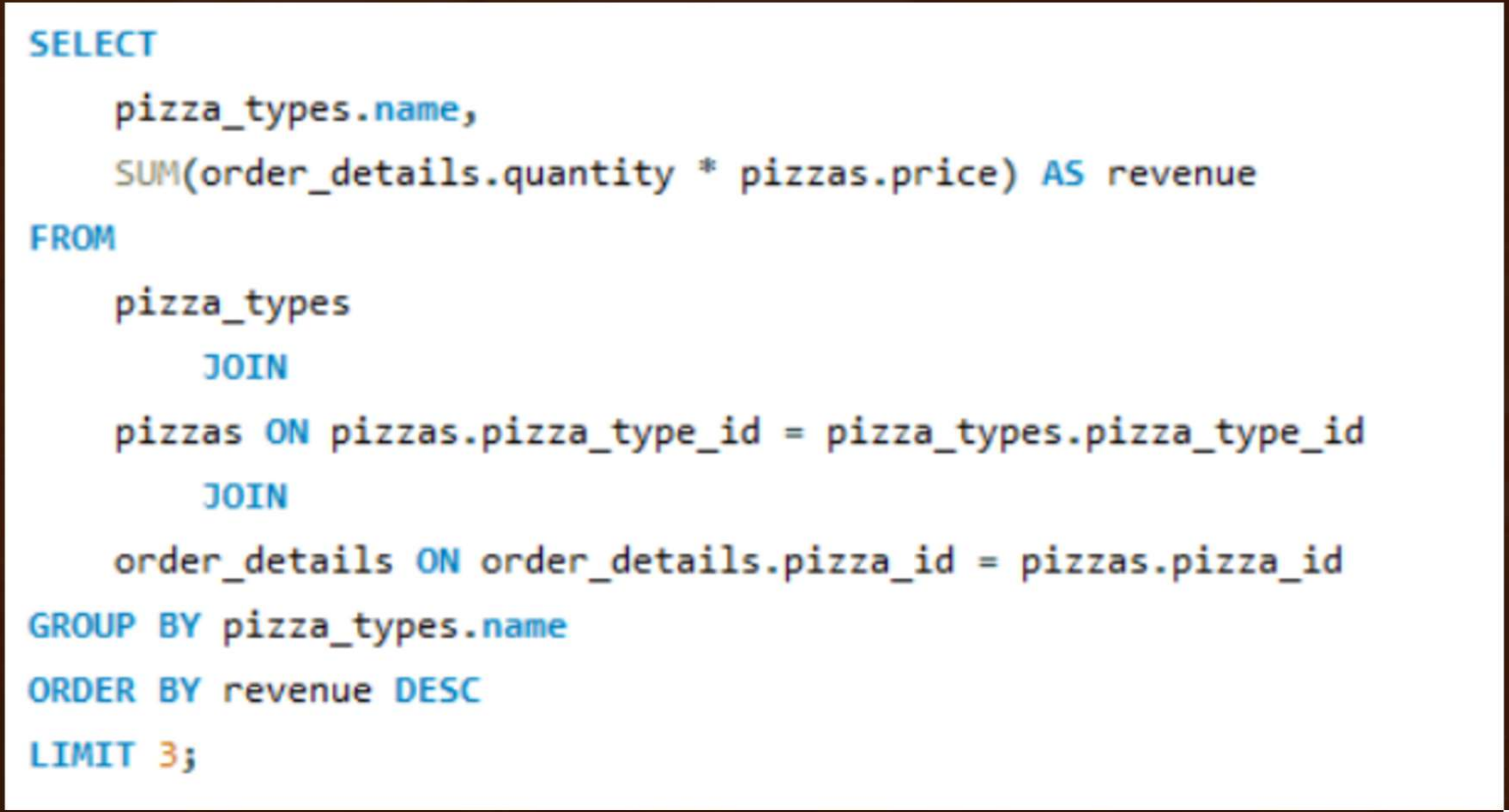
```
SELECT
    ROUND(AVG(quantity), 0) as avg_quantity_ordered_per_day
FROM
    (SELECT
        orders.order_date, SUM(order_details.quantity) AS quantity
    FROM
        orders
    JOIN order_details ON orders.order_id = order_details.order_id
    GROUP BY orders.order_date) AS order_quantity;
```



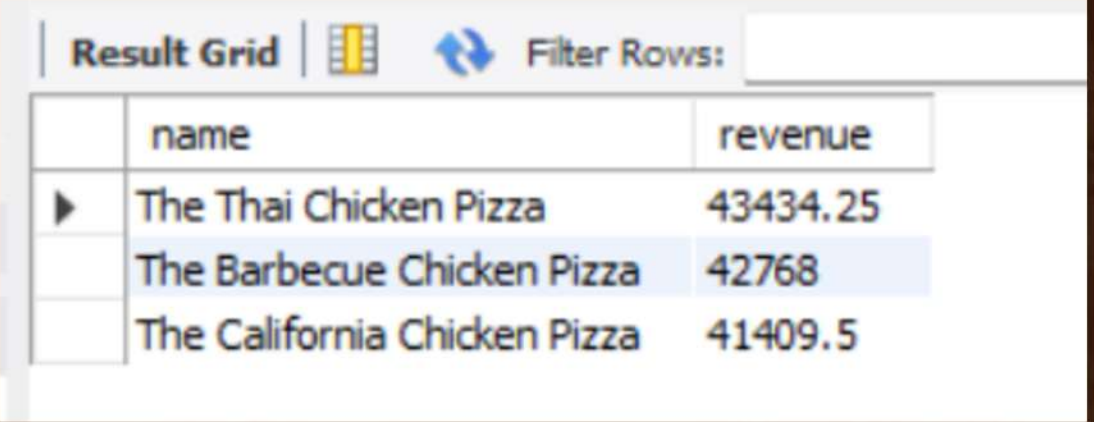
```
  avg_quantity_ordered_per_day
▶ 138
```

Insight: Useful for forecasting daily demand.

Determine the top 3 most ordered pizza types based on revenue



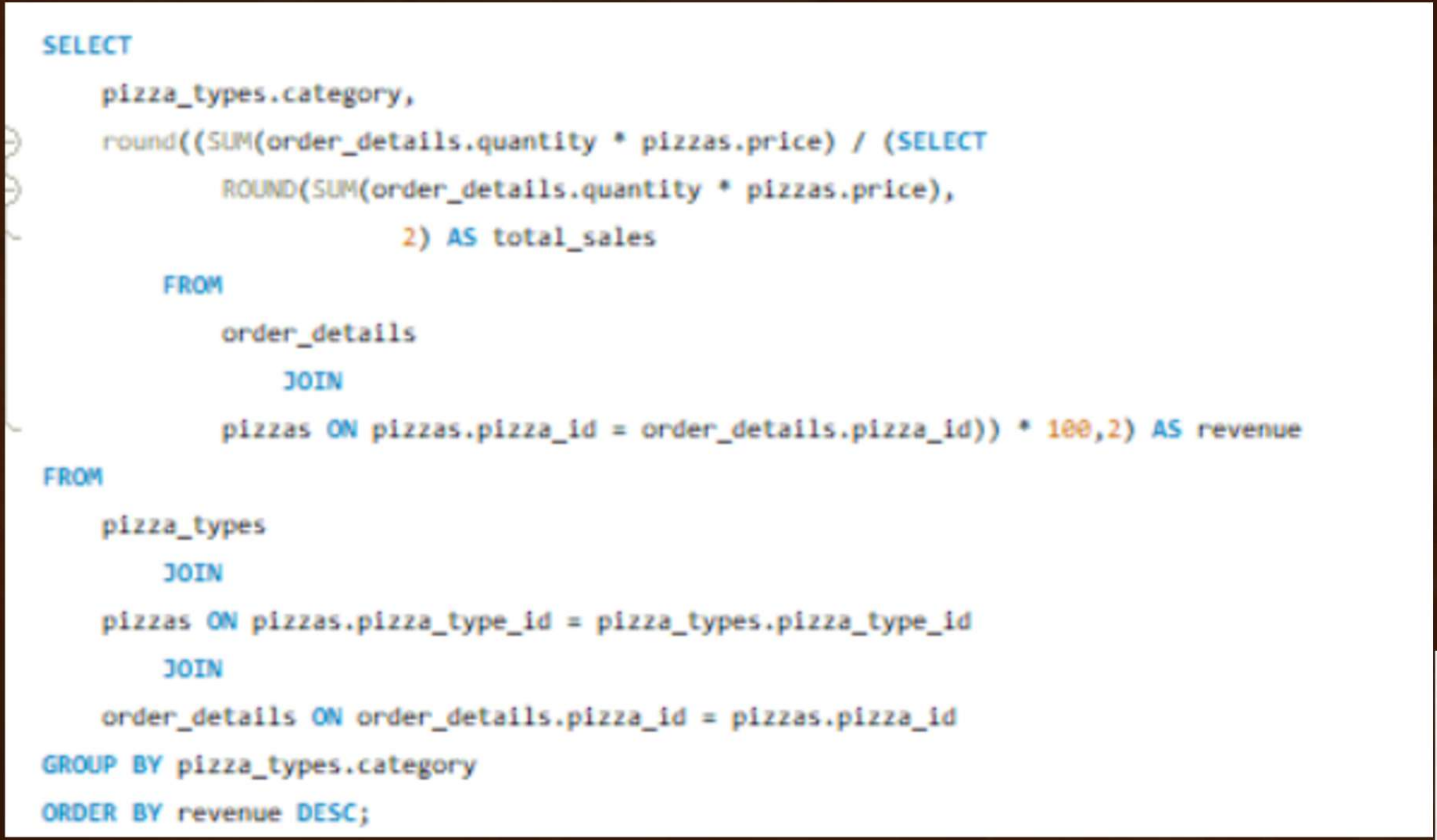
```sql
SELECT
    pizza_types.name,
    SUM(order_details.quantity * pizzas.price) AS revenue
FROM
    pizza_types
        JOIN
    pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY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



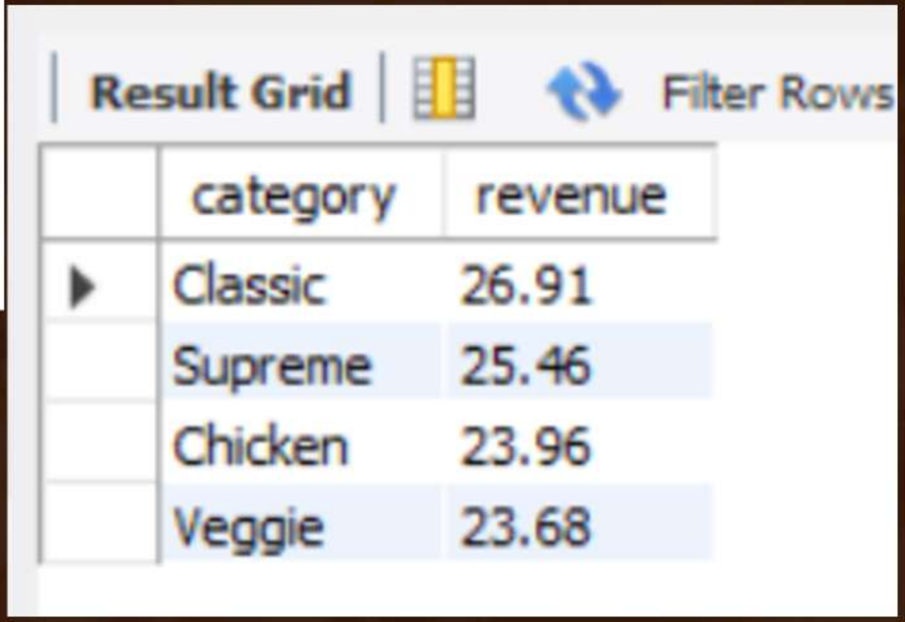
name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5

Insight: Identifies best-sellers not just by volume, but by profitability.

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



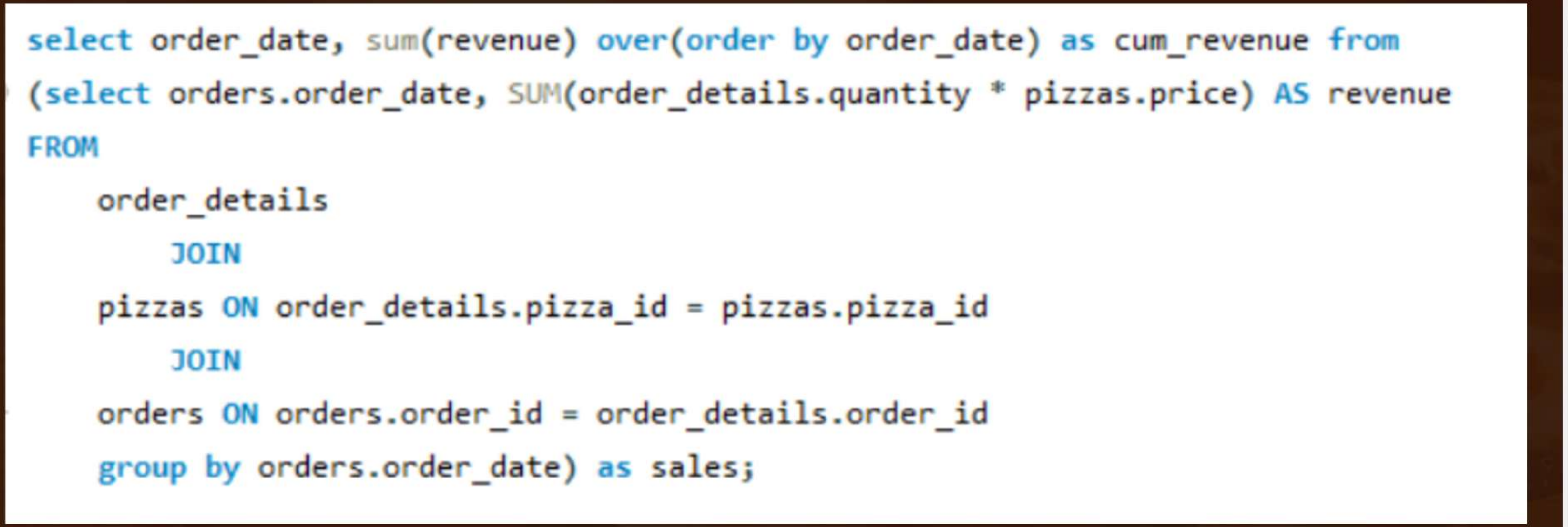
```
SELECT
    pizza_types.category,
    round((SUM(order_details.quantity * pizzas.price) / (SELECT
            ROUND(SUM(order_details.quantity * pizzas.price),
                        2) AS total_sales
        FROM
            order_details
                JOIN
            pizzas ON pizzas.pizza_id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



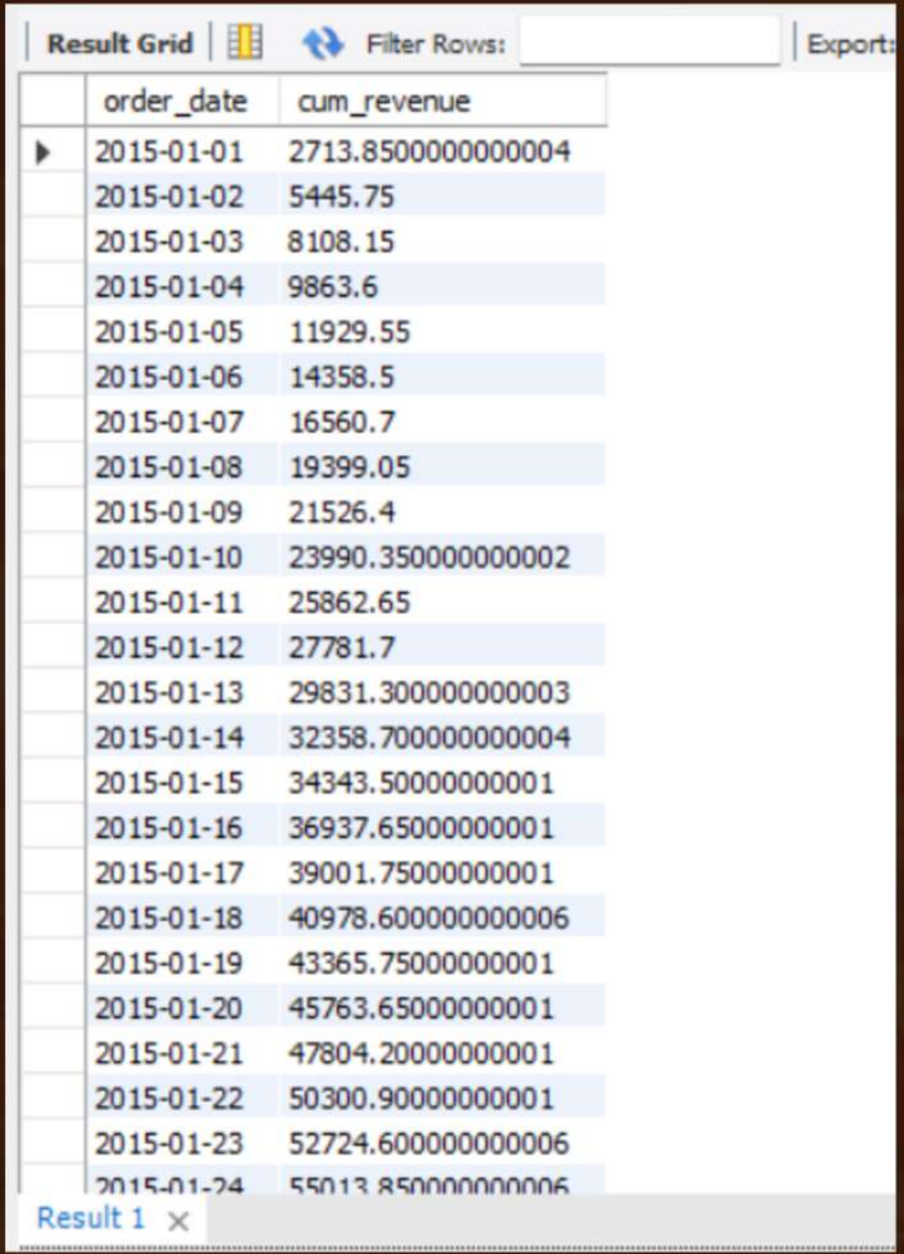
category	revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

Insight: Shows which pizzas are contributing the most to overall revenue.

Analyze the cumulative revenue generated over time.



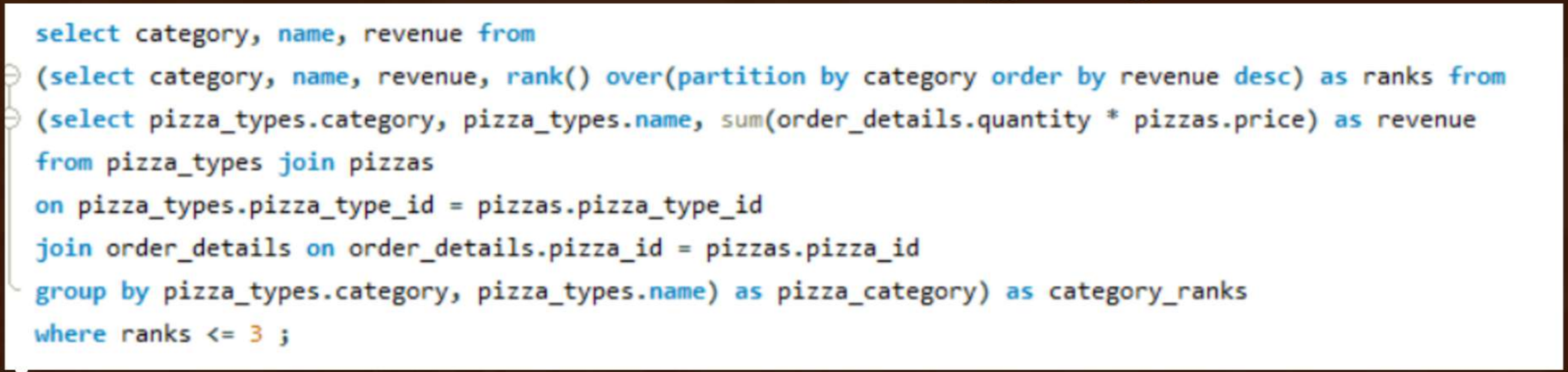
```sql
select order_date, sum(revenue) over(order by order_date) as cum_revenue from
(select orders.order_date, SUM(order_details.quantity * pizzas.price) AS revenue
FROM
    order_details
        JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id
        JOIN
    orders ON orders.order_id =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



order_date	cum_revenue
2015-01-01	2713.850000000004
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
2015-01-08	19399.05
2015-01-09	21526.4
2015-01-10	23990.350000000002
2015-01-11	25862.65
2015-01-12	27781.7
2015-01-13	29831.300000000003
2015-01-14	32358.700000000004
2015-01-15	34343.50000000001
2015-01-16	36937.65000000001
2015-01-17	39001.75000000001
2015-01-18	40978.600000000006
2015-01-19	43365.75000000001
2015-01-20	45763.65000000001
2015-01-21	47804.20000000001
2015-01-22	50300.90000000001
2015-01-23	52724.600000000006
2015-01-24	55013.850000000006

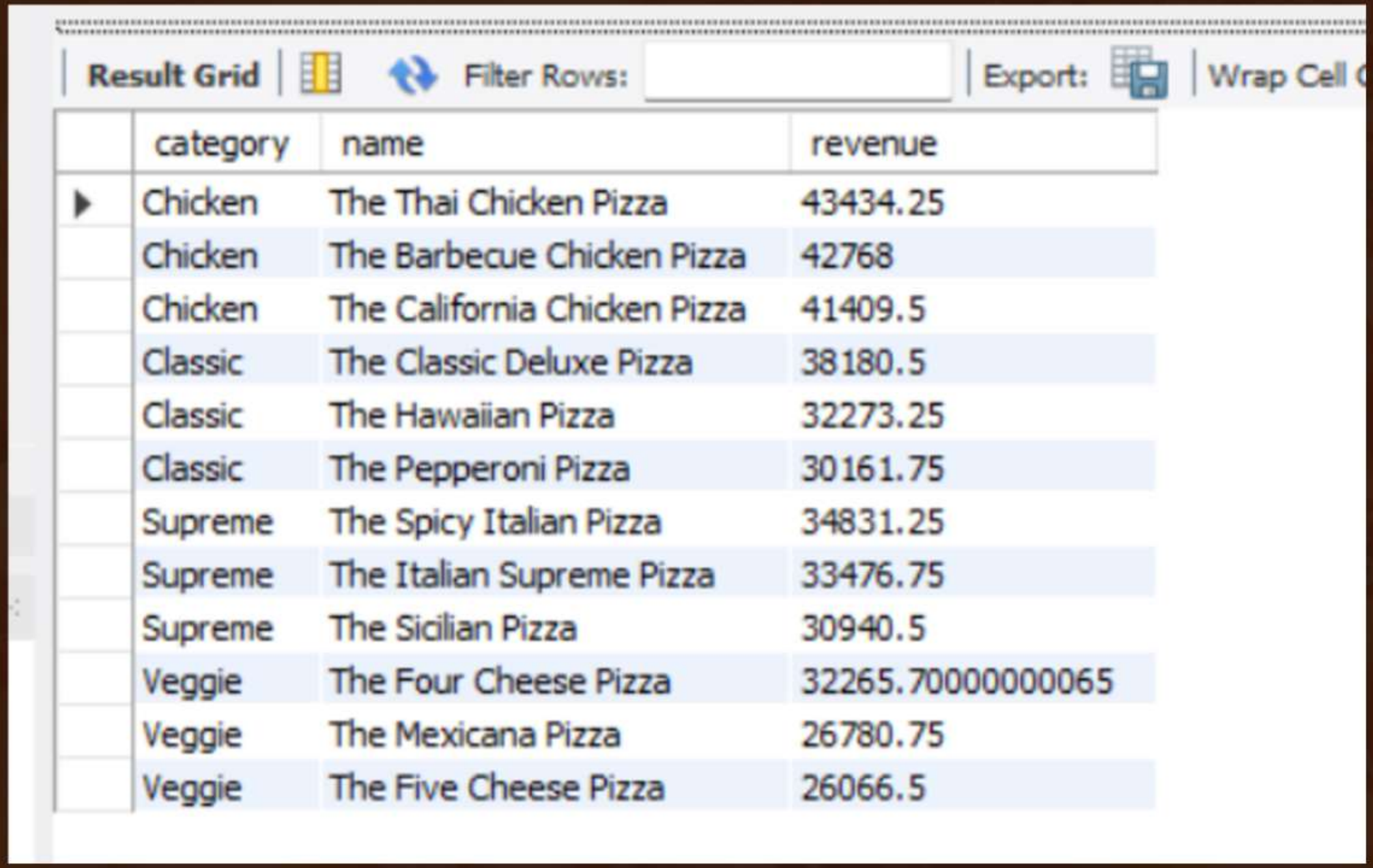
Insight: Great for visualizing business growth and forecasting trends.

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



```
select category, name, revenue from
(select category, name, revenue, rank() over(partition by category order by revenue desc) as ranks 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 pizza_category) as category_ranks
where ranks <= 3 ;
```

Insight: Helps find top performers in each category—useful for curated menus or promotions.



category	name	revenue
Chicken	The Thai Chicken Pizza	43434.25
Chicken	The Barbecue Chicken Pizza	42768
Chicken	The California Chicken Pizza	41409.5
Classic	The Classic Deluxe Pizza	38180.5
Classic	The Hawaiian Pizza	32273.25
Classic	The Pepperoni Pizza	30161.75
Supreme	The Spicy Italian Pizza	34831.25
Supreme	The Italian Supreme Pizza	33476.75
Supreme	The Sicilian Pizza	30940.5
Veggie	The Four Cheese Pizza	32265.70000000065
Veggie	The Mexicana Pizza	26780.75
Veggie	The Five Cheese Pizza	26066.5

CONCLUSION

1. Over $817,000 in revenue was generated from more than 21,000 customer orders.

2. Large-sized pizzas were the most frequently ordered size.

3. The Classic Deluxe and BBQ Chicken pizzas were the top-selling and most profitable items.

4. Most orders were placed during lunch and dinner hours, highlighting peak business times.

5. A few pizzas contributed to the majority of revenue, supporting the 80/20 rule in menu performance.

Pizza Sales Presentation

THANK YOU
FOR ATTENTION