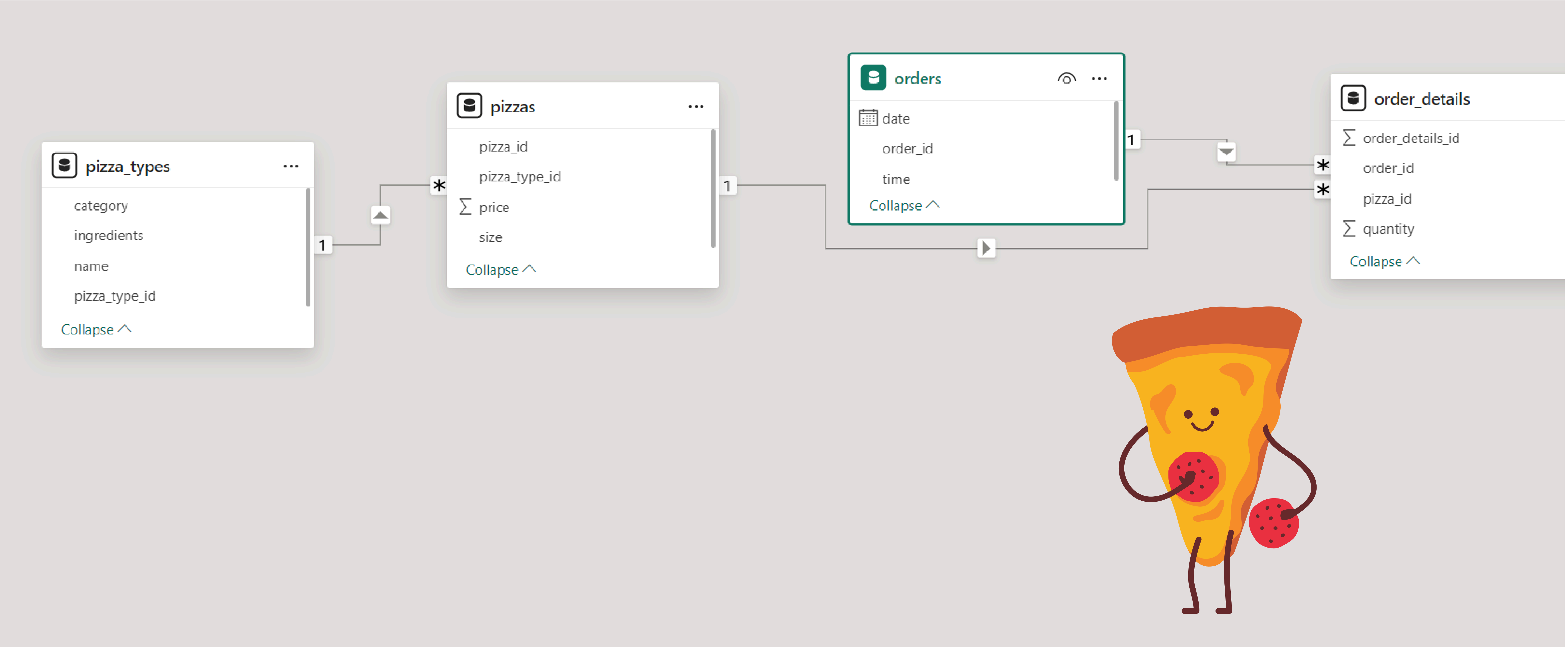
PIZZA SALES ANALYSIS USING SQL

Understand Schema



pizza_types
- category
- ingredients
- name
- pizza_type_id

Collapse

pizzas
- pizza_id
- pizza_type_id
- price
- size

Collapse

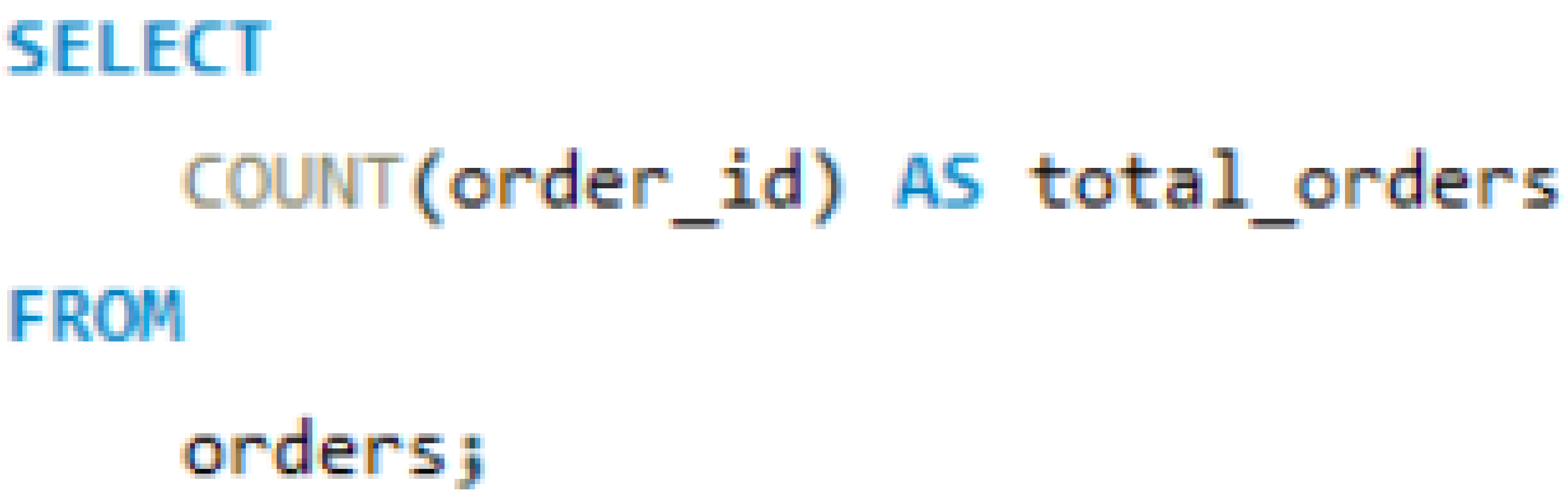
orders
- date
- order_id
- time

Collapse

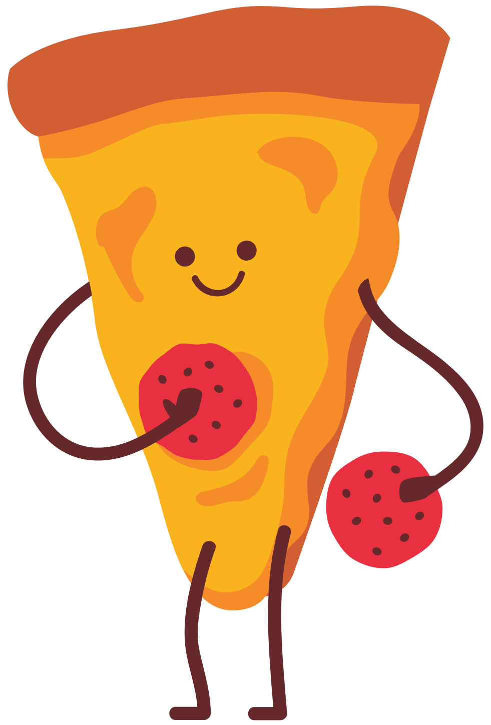
order_details
- order_details_id
- order_id
- pizza_id
- quantity

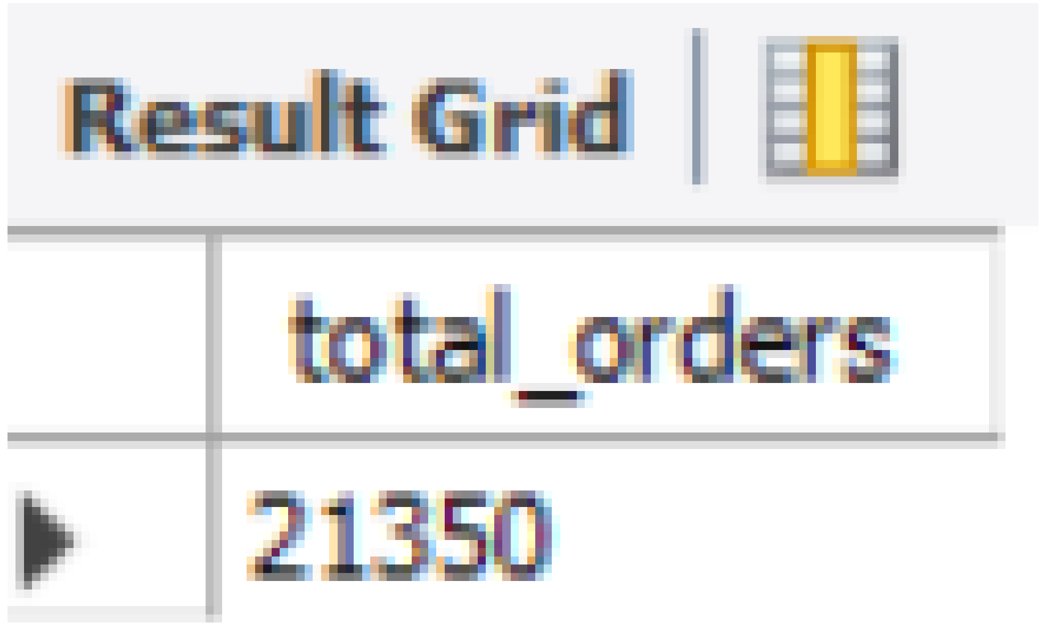
Collapse

Retrieve the total number of orders placed.



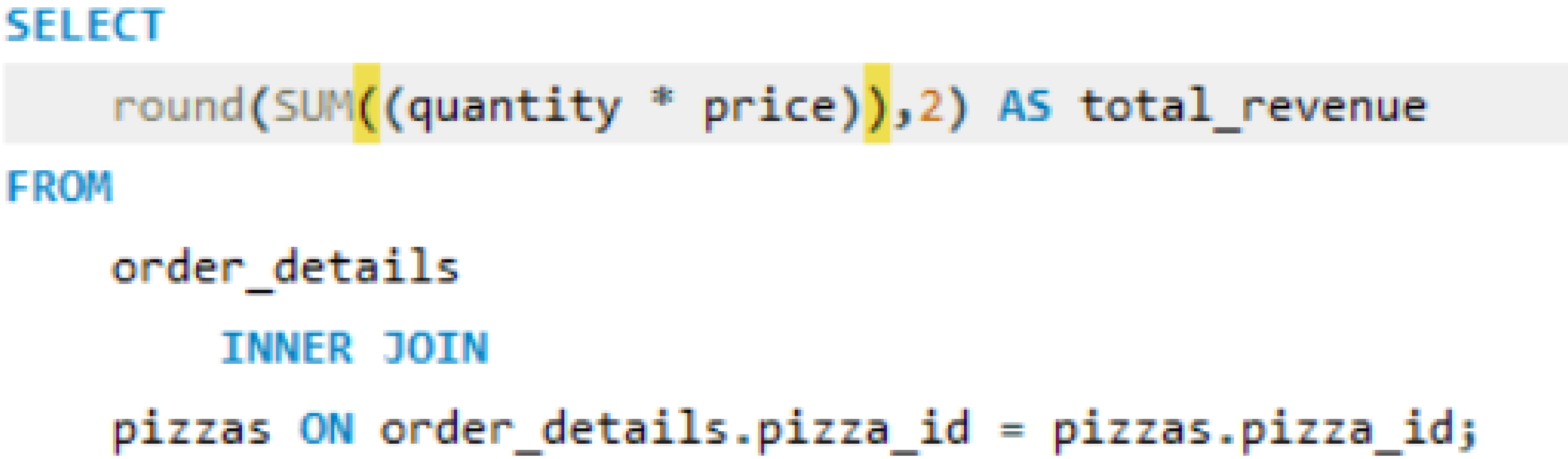
```sql
SELECT
    COUNT(order_id) AS total_orders
FROM
    orders;
```



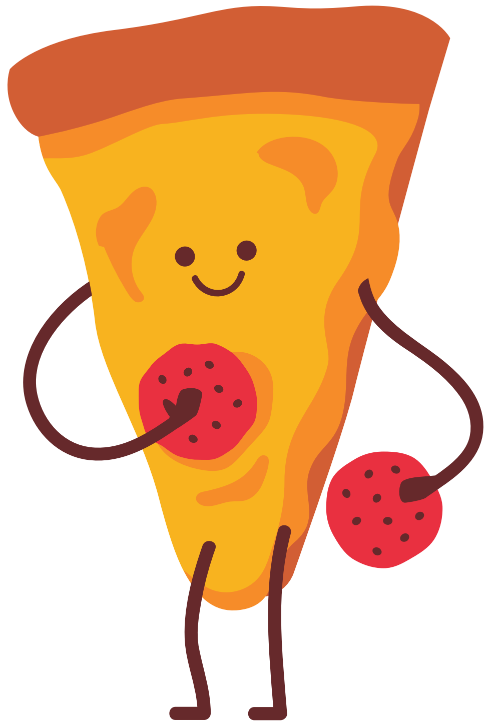


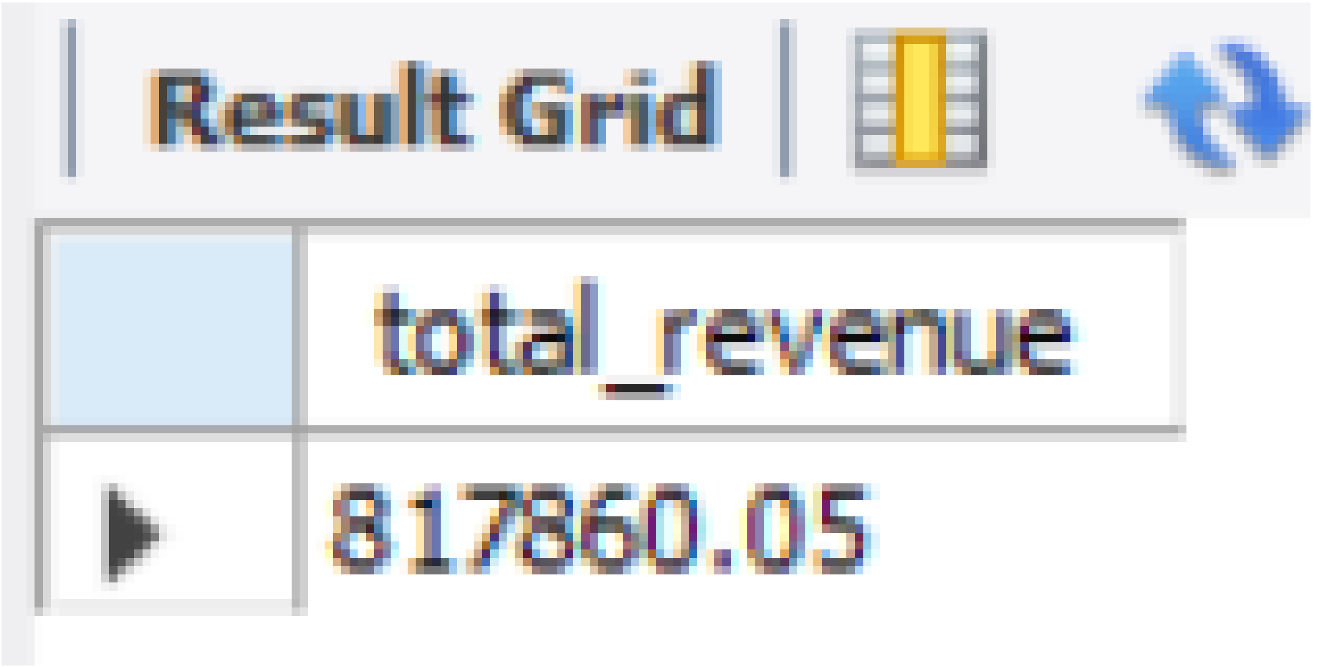
```
total_orders
21350
```

Calculate the total revenue generated from pizza sales.



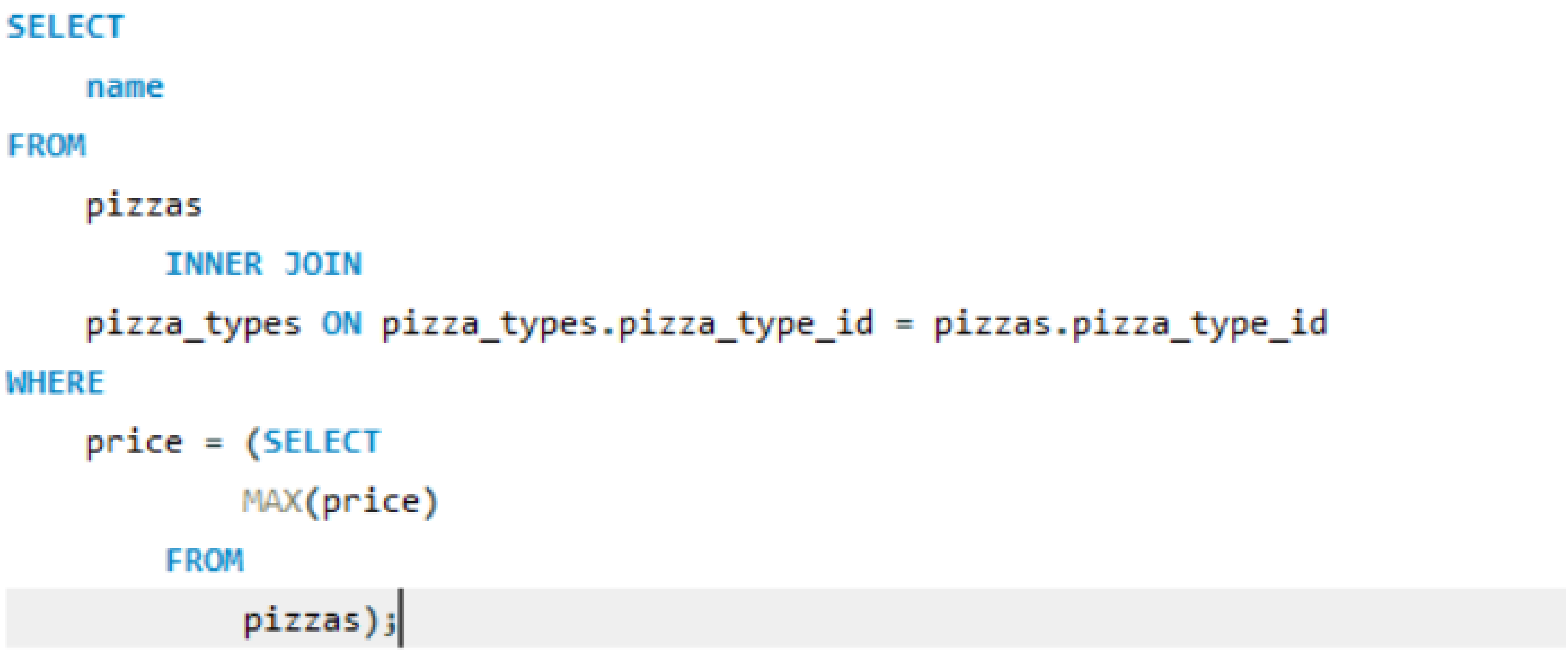
```sql
SELECT
    round(SUM((quantity * price)),2) AS total_revenue
FROM
    order_details
        INNER JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id;
```



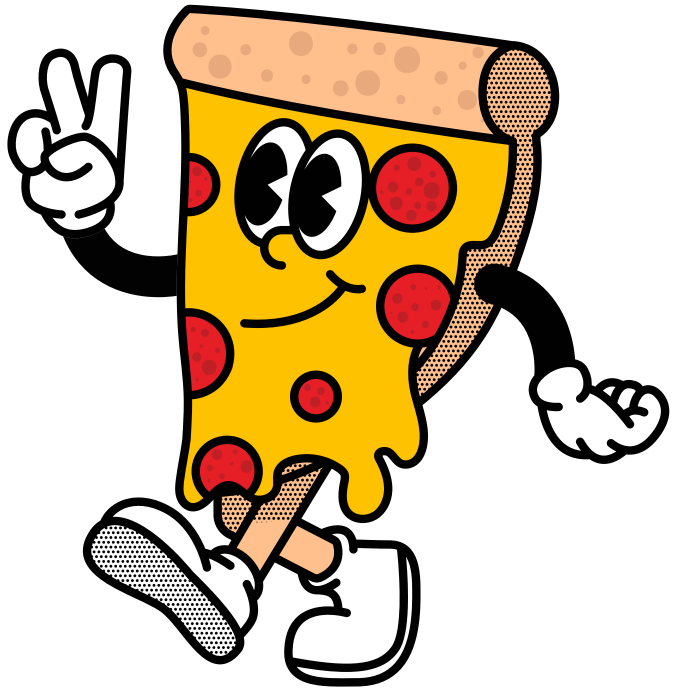


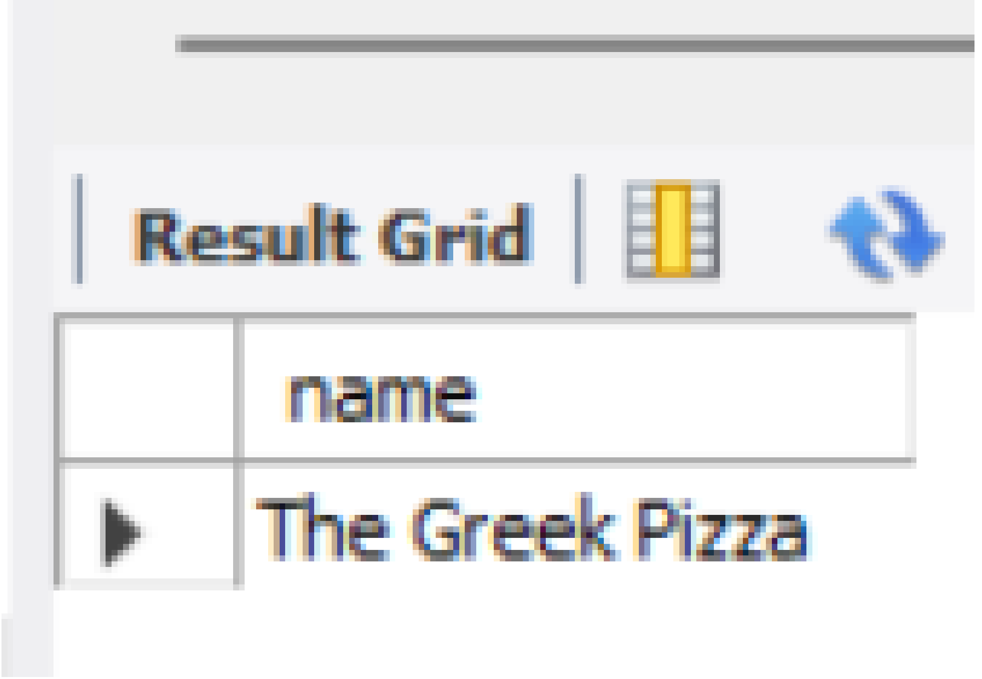
```
total_revenue
817860.05
```

Identify the highest-priced pizza.



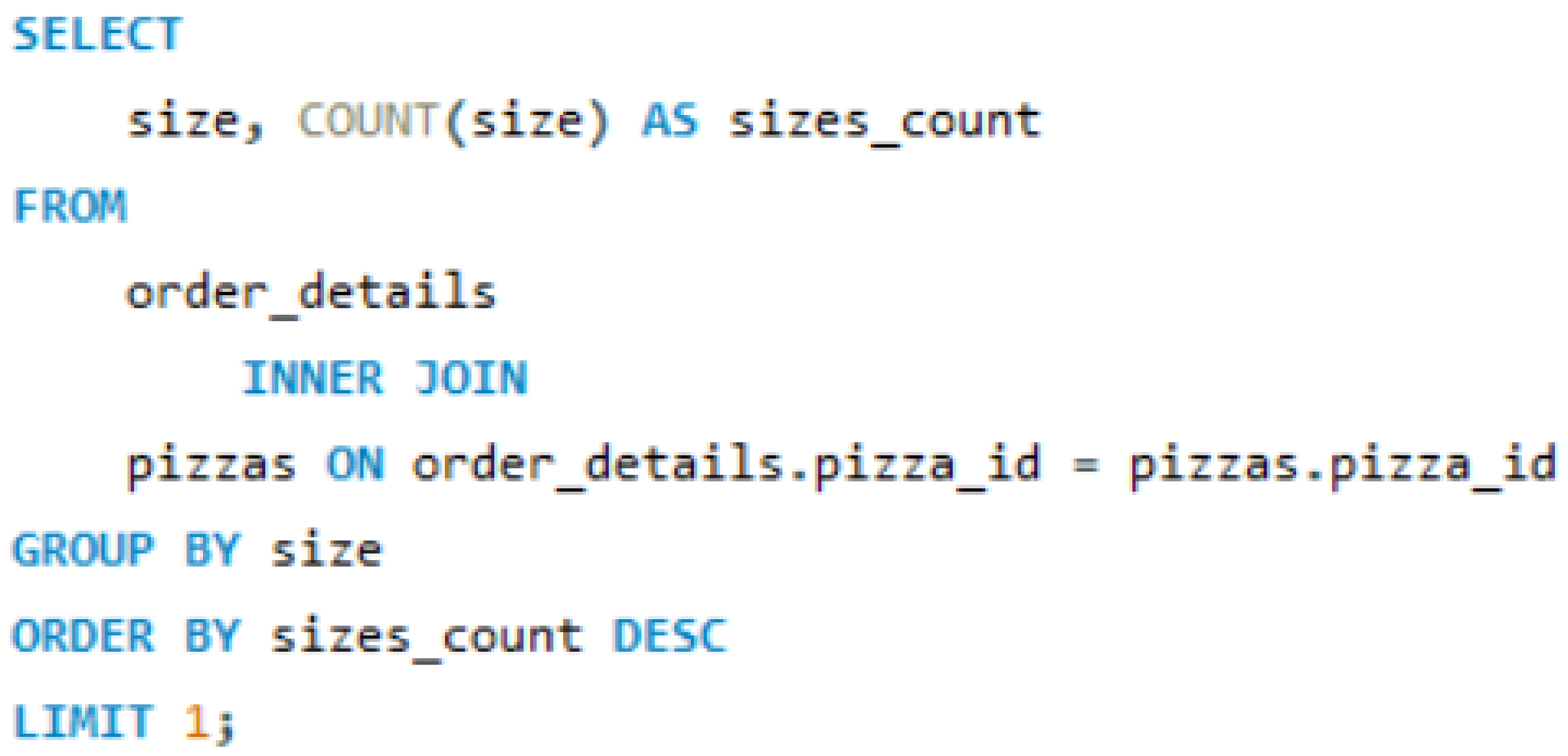
```
SELECT
    name
FROM
    pizzas
        INNER JOIN
    pizza_types ON pizza_types.pizza_type_id = pizzas.pizza_type_id
WHERE
    price = (SELECT
            MAX(price)
        FROM
            pizzas);
```



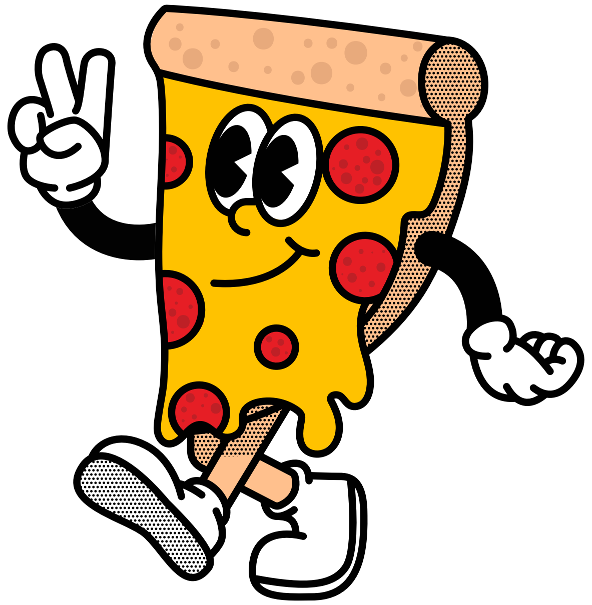


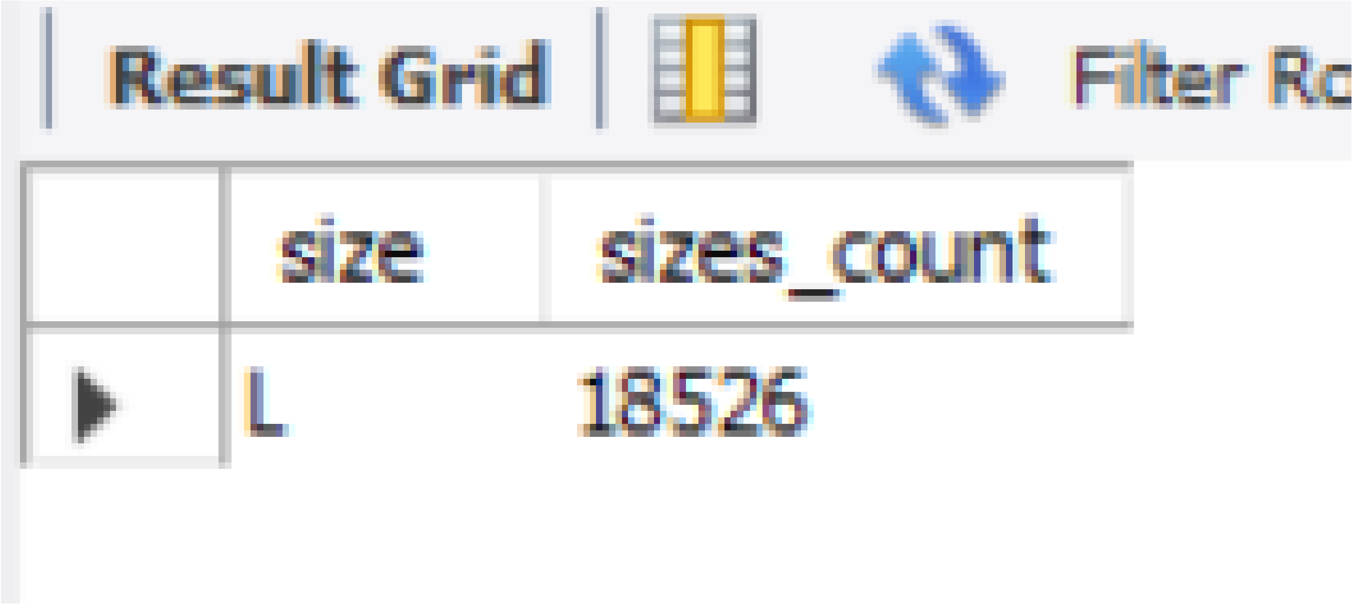
	name
▶	The Greek Pizza

Identify the most common pizza size ordered.



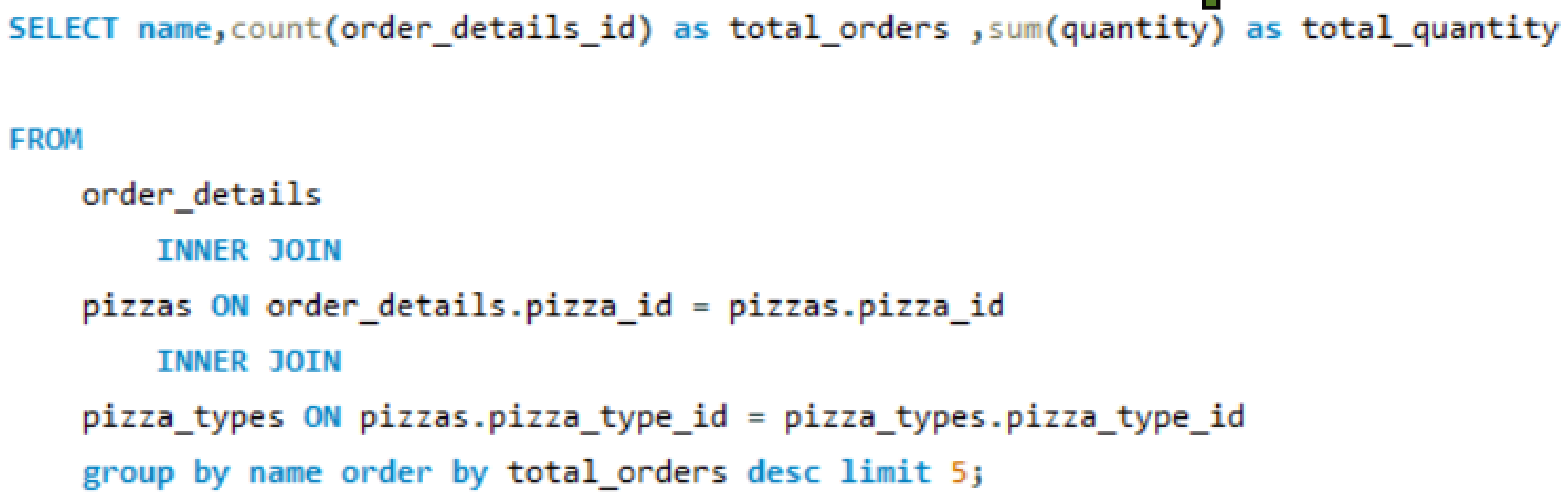
```
SELECT
    size, COUNT(size) AS sizes_count
FROM
    order_details
        INNER JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id
GROUP BY size
ORDER BY sizes_count DESC
LIMIT 1;
```



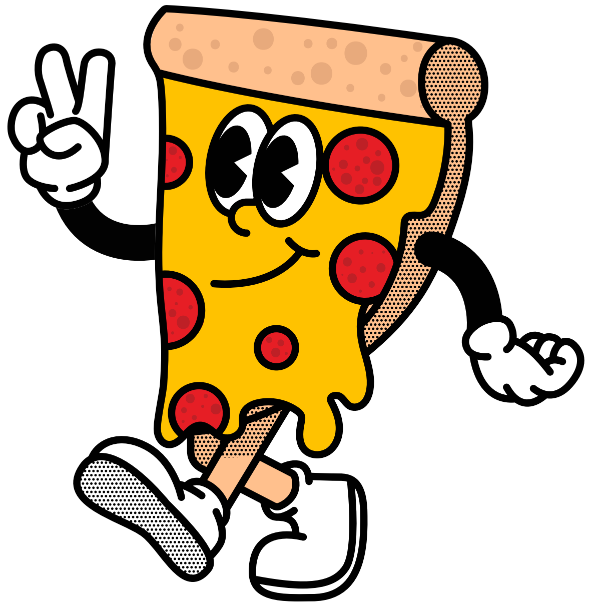


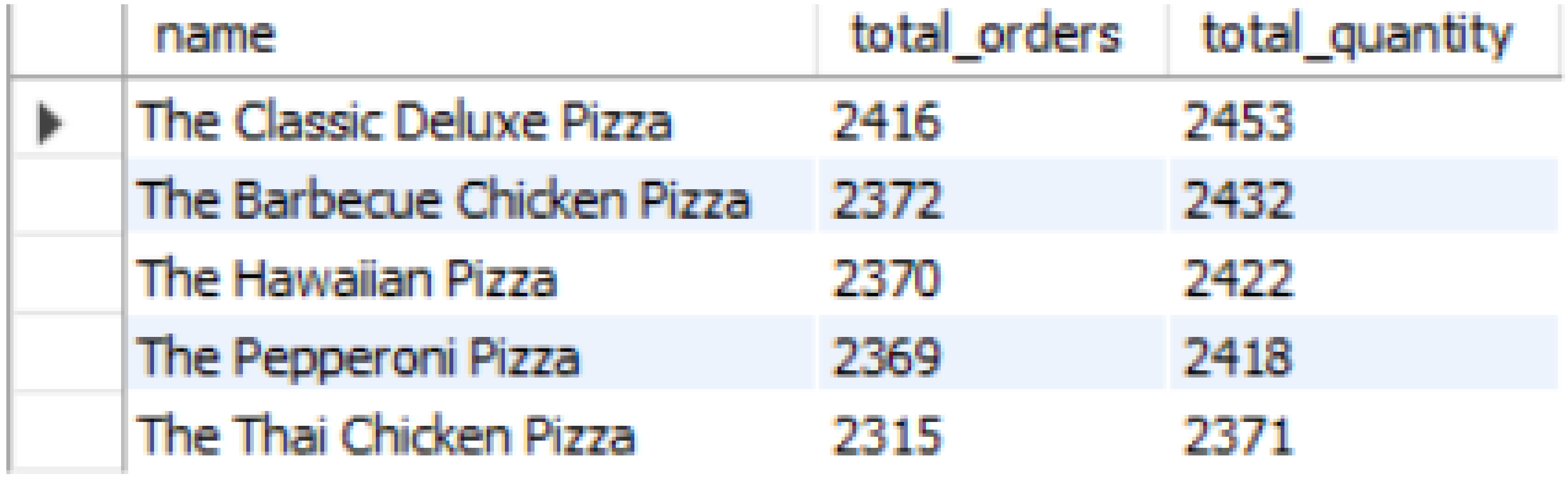
	size	sizes_count
▶	L	18526

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



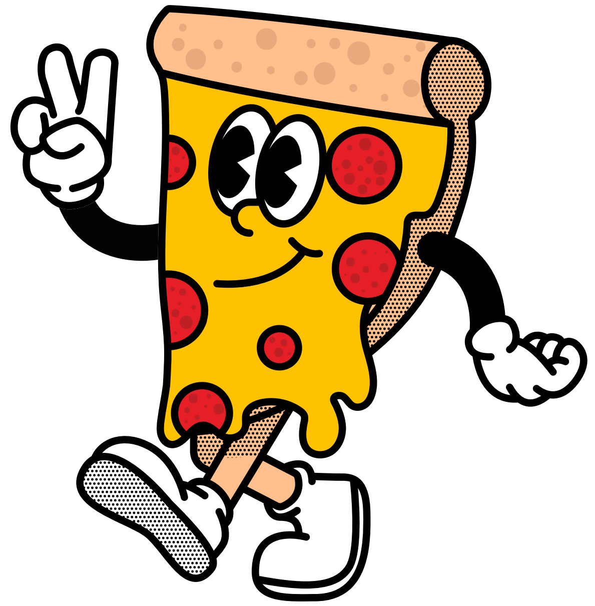
```
SELECT name,count(order_details_id) as total_orders ,sum(quantity) as total_quantity

FROM
    order_details
        INNER JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id
        INNER JOIN
    pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
    group by name order by total_orders desc limit 5;
```

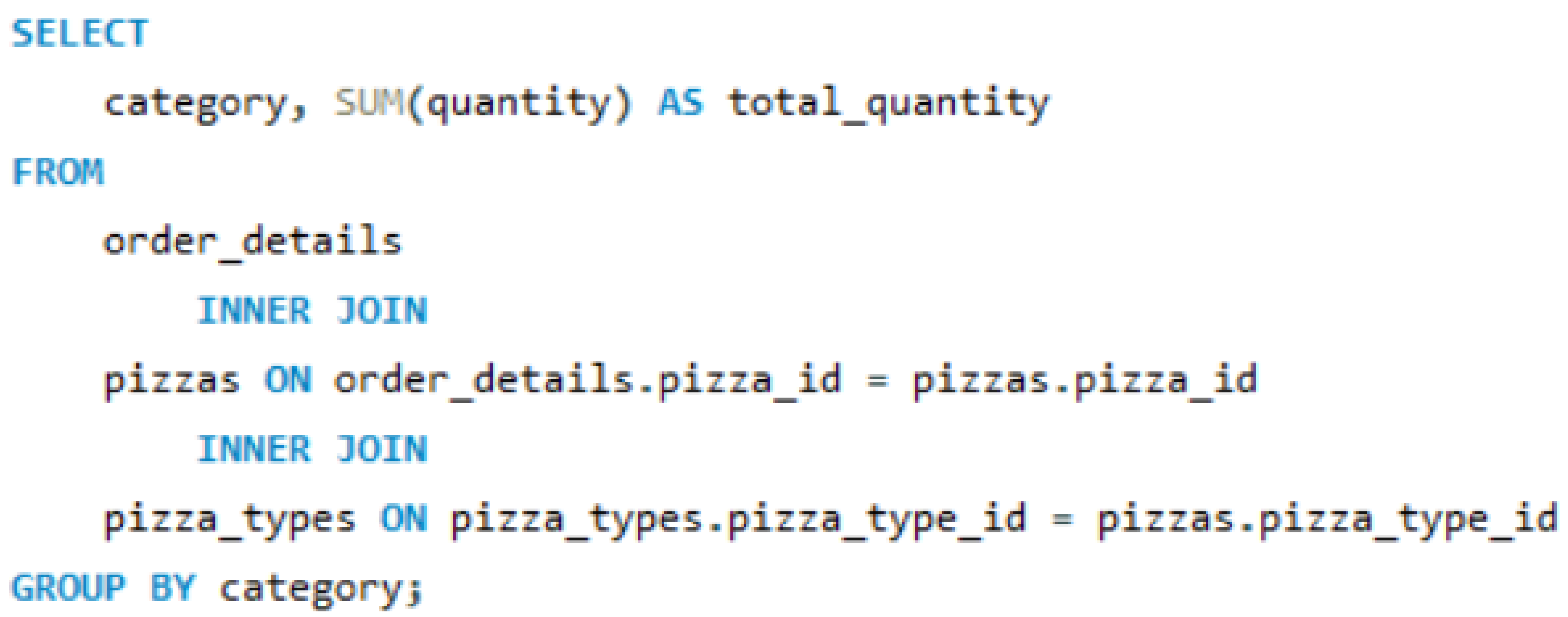




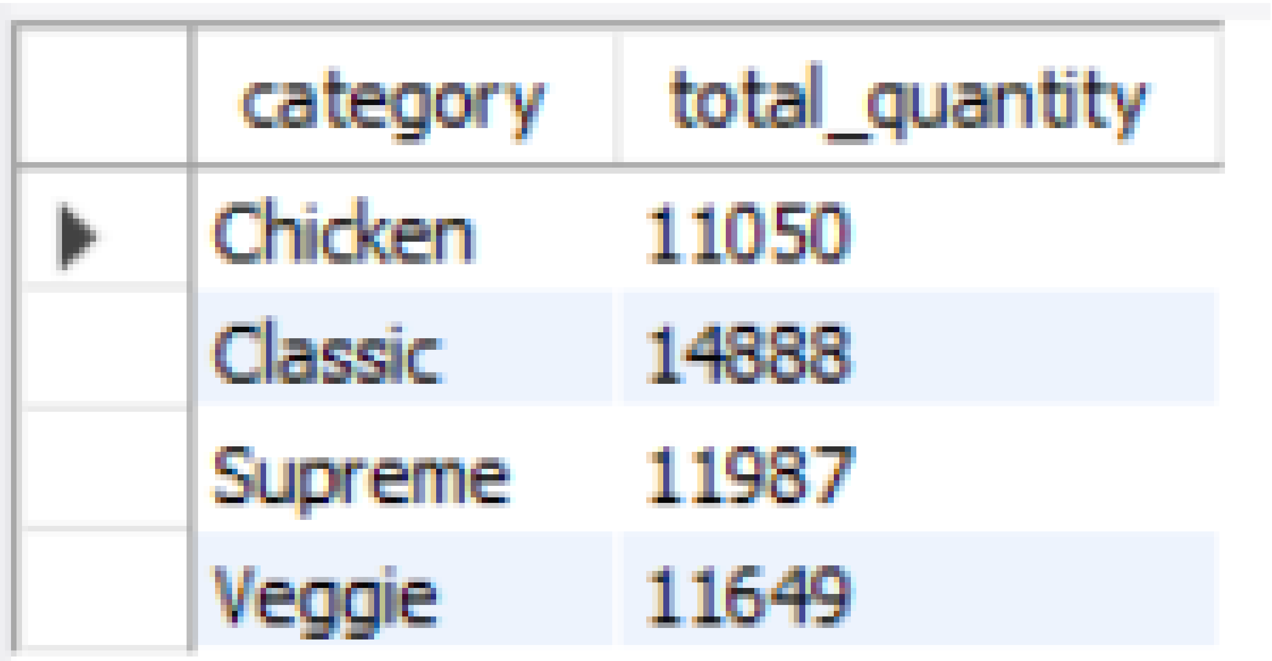
name	total_orders	total_quantity
The Classic Deluxe Pizza	2416	2453
The Barbecue Chicken Pizza	2372	2432
The Hawaiian Pizza	2370	2422
The Pepperoni Pizza	2369	2418
The Thai Chicken Pizza	2315	2371

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



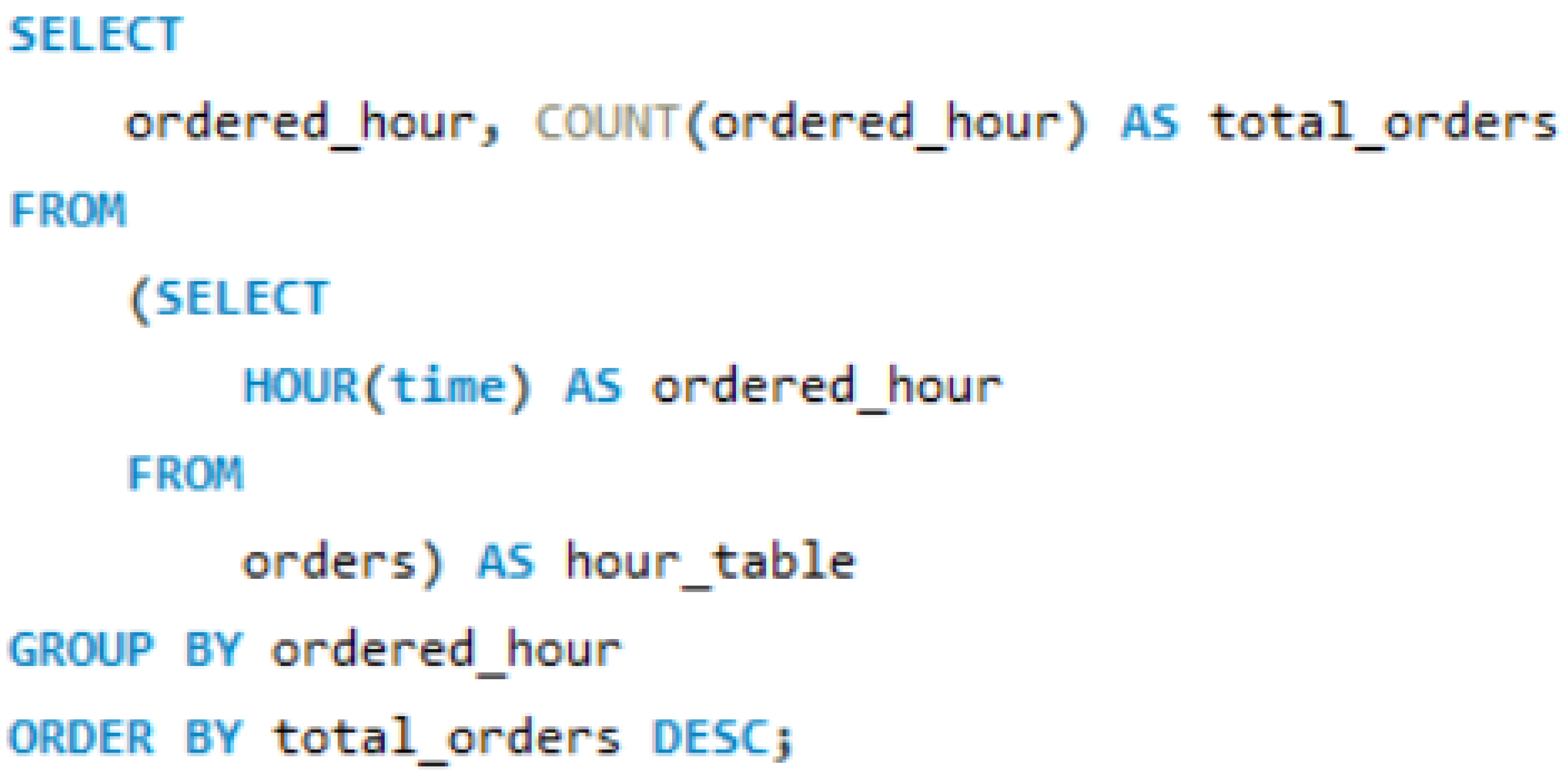


```
SELECT
    category, SUM(quantity) AS total_quantity
FROM
    order_details
        INNER JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id
        INNER JOIN
    pizza_types ON pizza_types.pizza_type_id = pizzas.pizza_type_id
GROUP BY category;
```

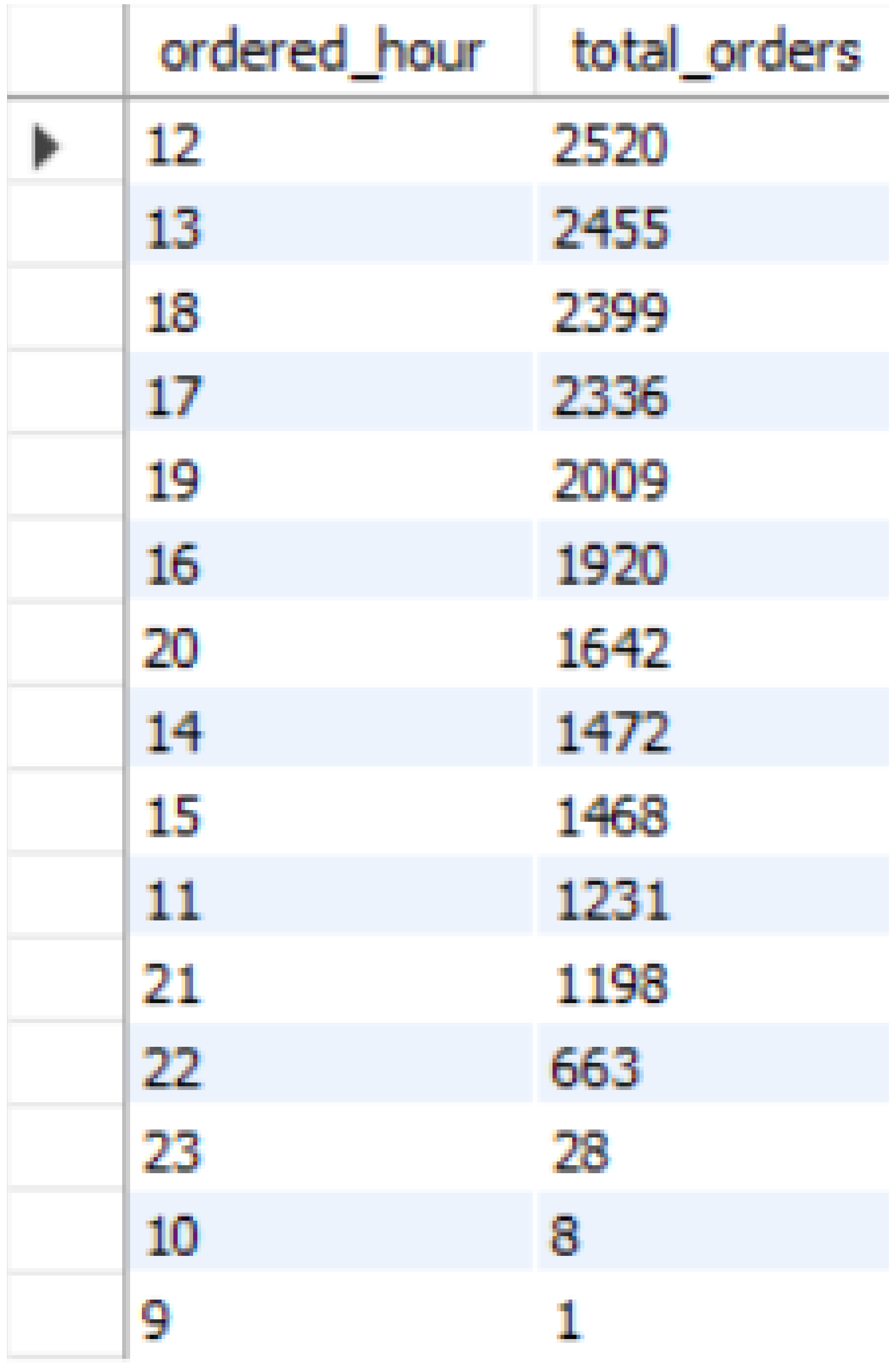


	category	total_quantity
▶	Chicken	11050
	Classic	14888
	Supreme	11987
	Veggie	11649

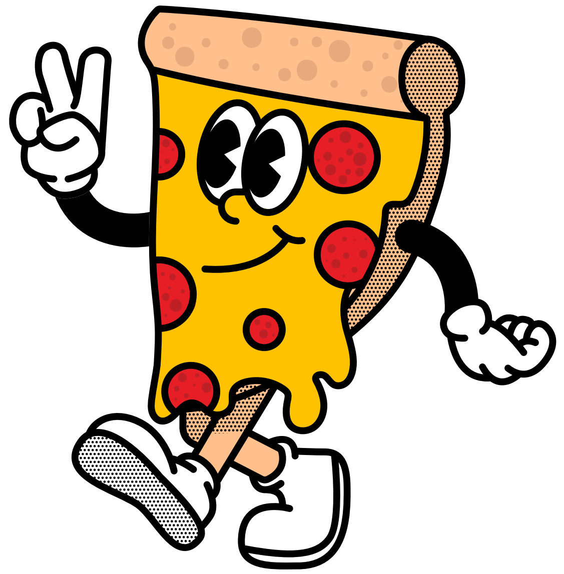
Determine the distribution of orders by hour of the day.



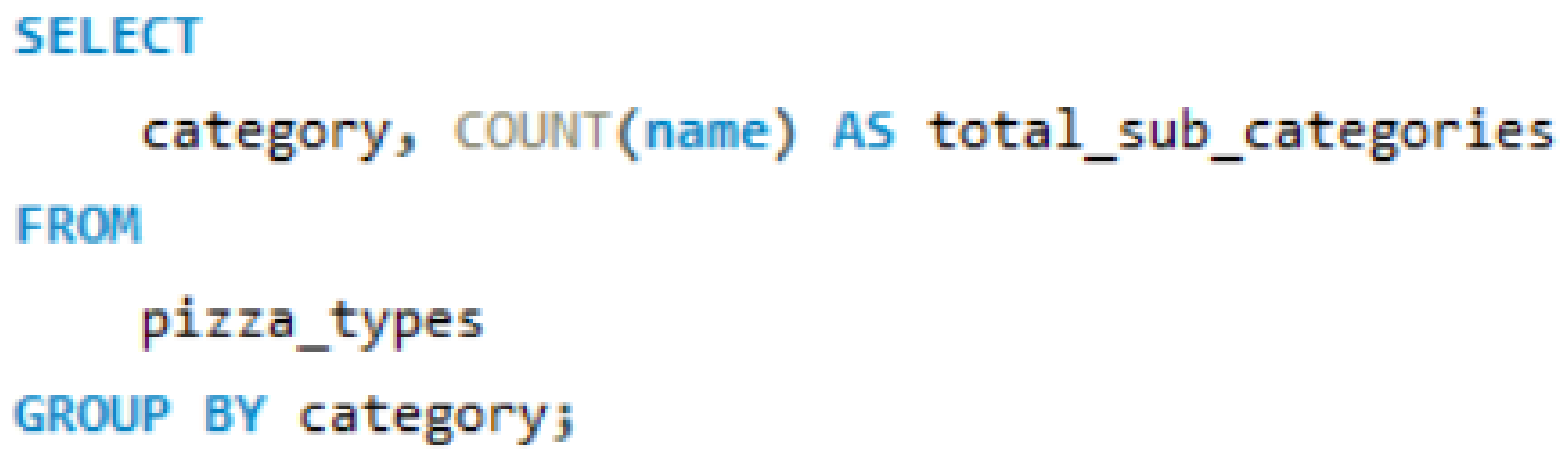
```
SELECT
    ordered_hour, COUNT(ordered_hour) AS total_orders
FROM
    (SELECT
        HOUR(time) AS ordered_hour
    FROM
        orders) AS hour_table
GROUP BY ordered_hour
ORDER BY total_orders DESC;
```



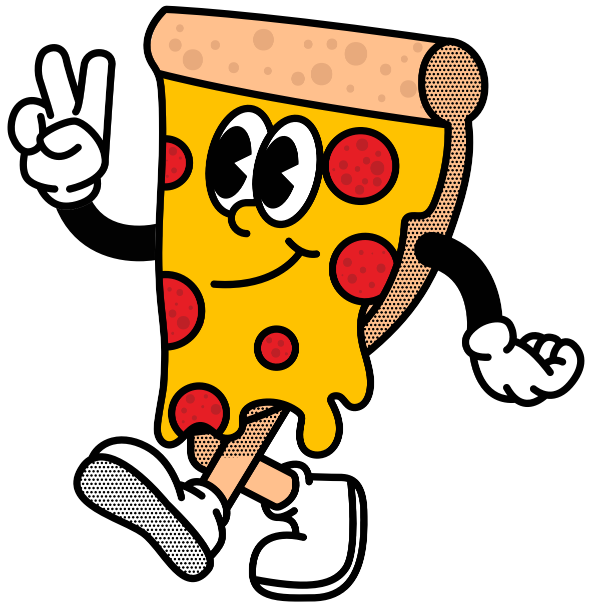
ordered_hour	total_orders
12	2520
13	2455
18	2399
17	2336
19	2009
16	1920
20	1642
14	1472
15	1468
11	1231
21	1198
22	663
23	28
10	8
9	1

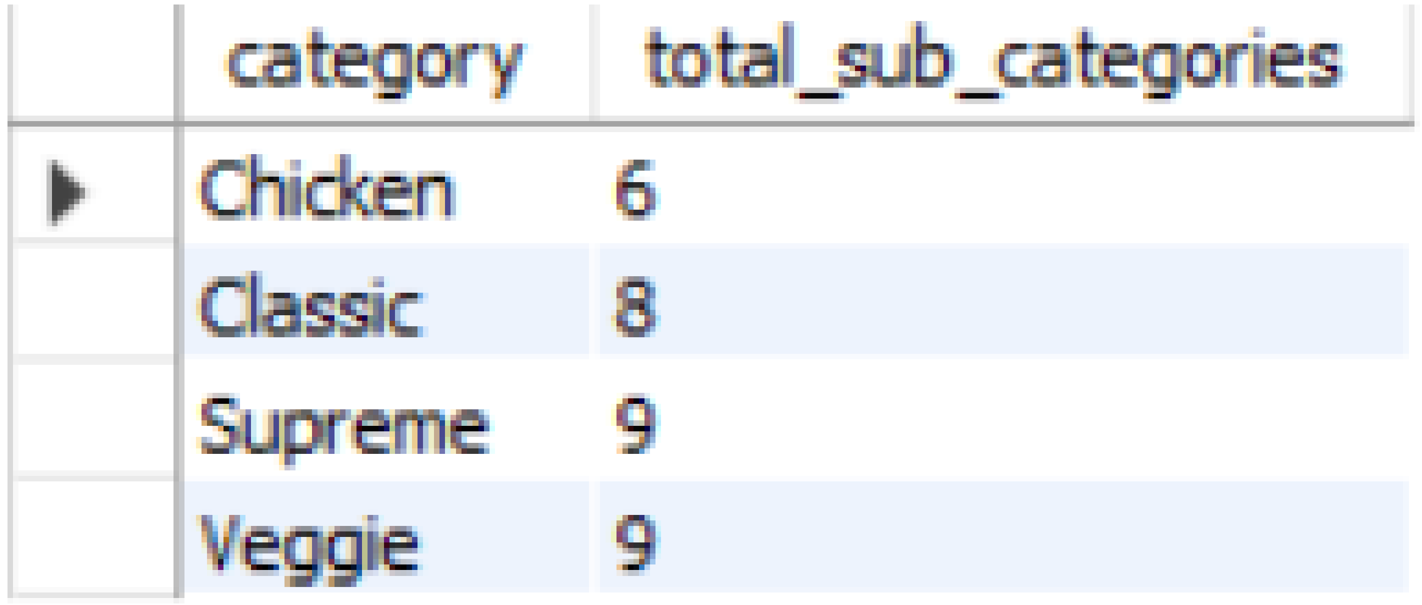


Join relevant tables to find the category-wise distribution of pizzas.



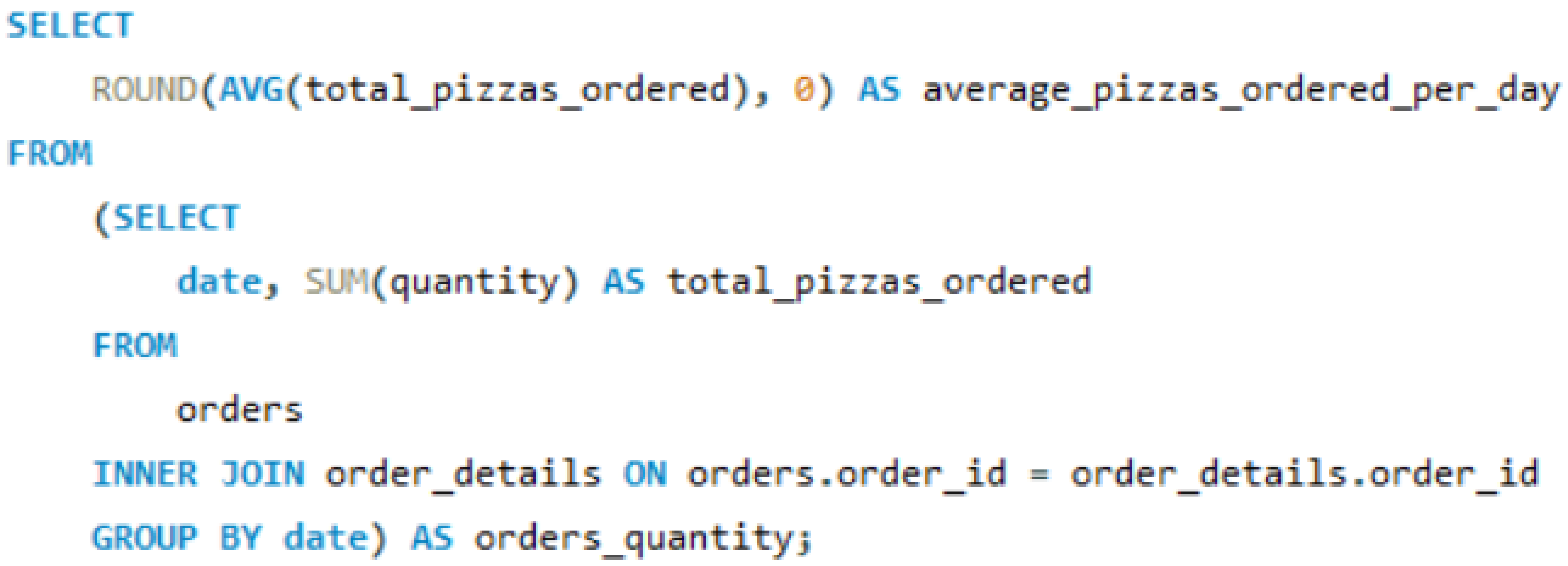
```
SELECT
    category, COUNT(name) AS total_sub_categories
FROM
    pizza_types
GROUP BY category;
```



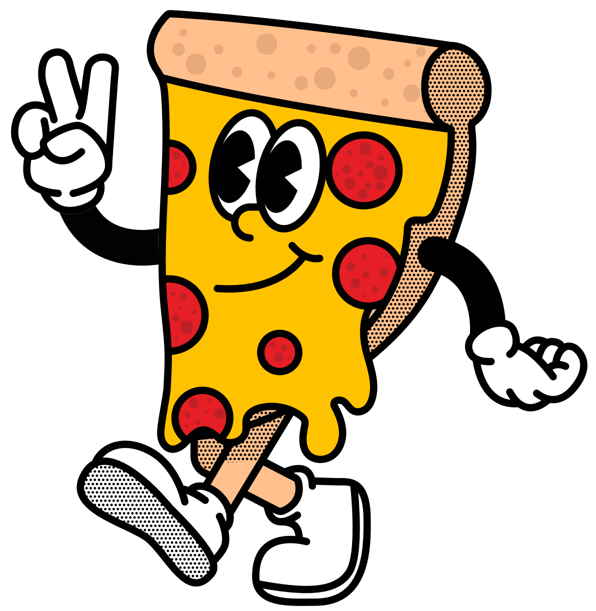


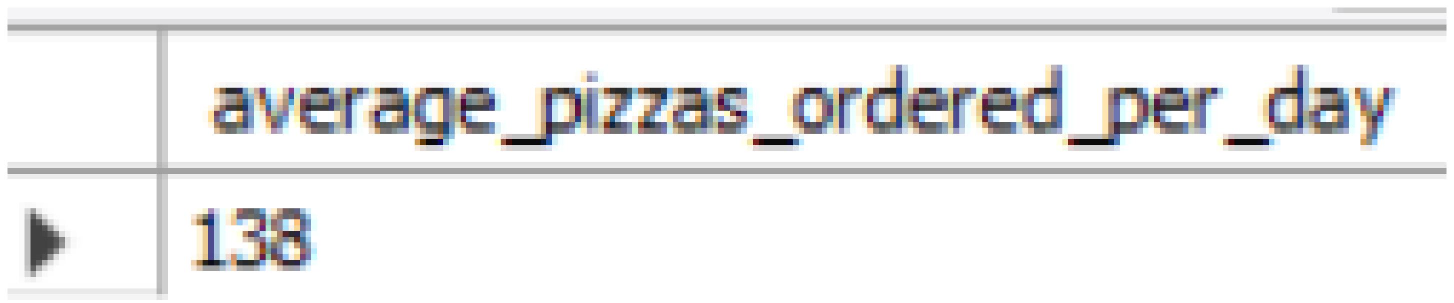
category	total_sub_categories
Chicken	6
Classic	8
Supreme	9
Veggie	9

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



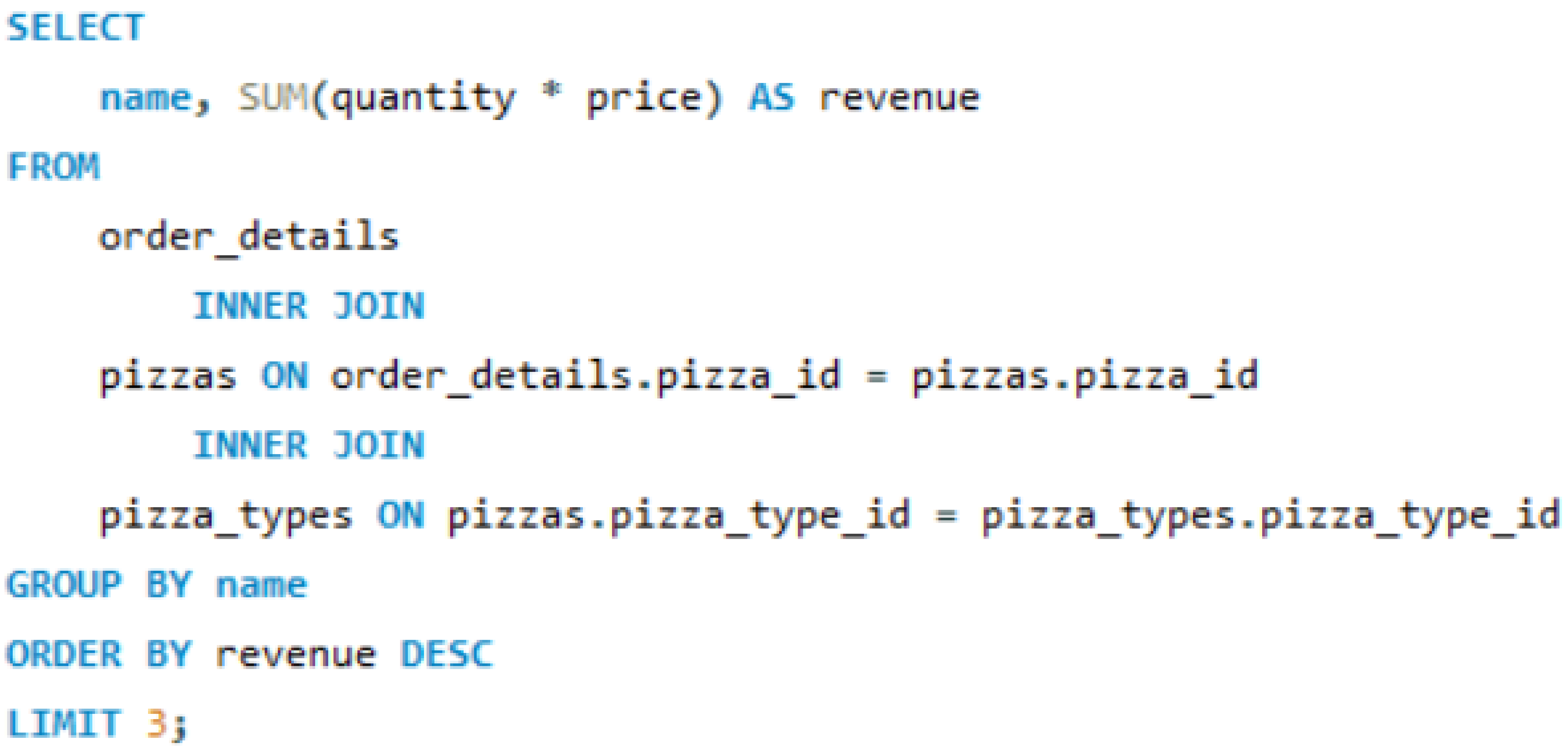
```
SELECT
    ROUND(AVG(total_pizzas_ordered), 0) AS average_pizzas_ordered_per_day
FROM
    (SELECT
        date, SUM(quantity) AS total_pizzas_ordered
    FROM
        orders
    INNER JOIN order_details ON orders.order_id = order_details.order_id
    GROUP BY date) AS orders_quantity;
```



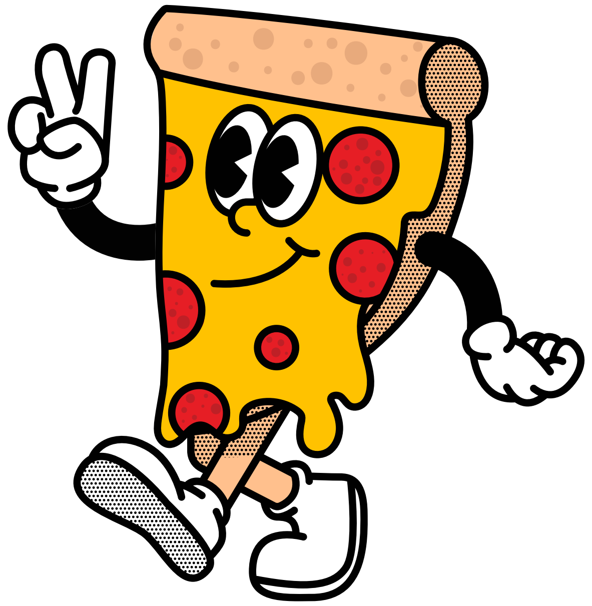


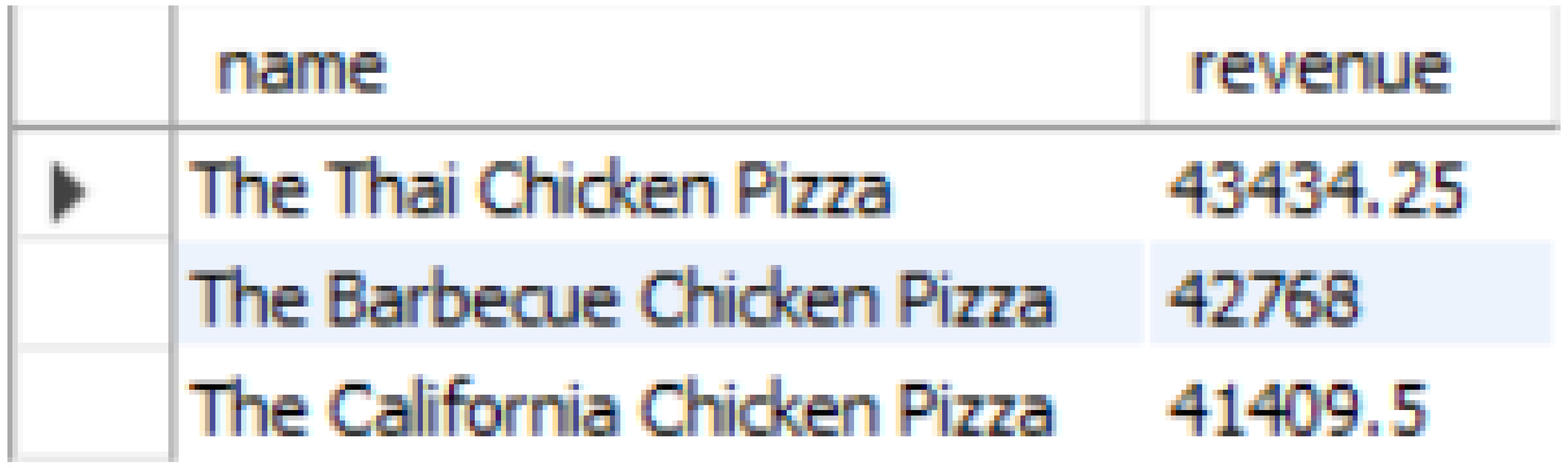
	average_pizzas_ordered_per_day
▶	138

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



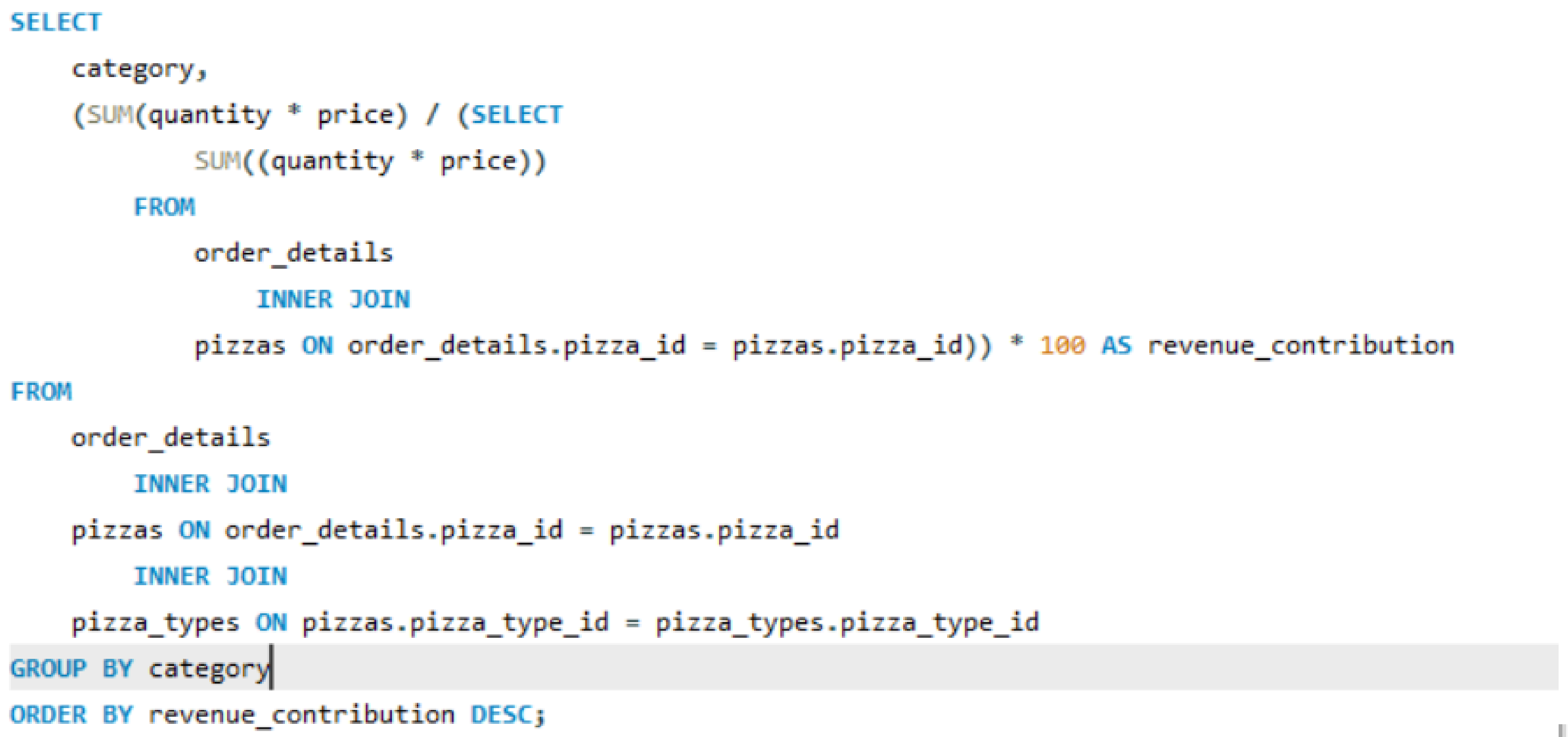
```
SELECT
    name, SUM(quantity * price) AS revenue
FROM
    order_details
        INNER JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id
        INNER JOIN
    pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
GROUP BY name
ORDER BY revenue DESC
LIMIT 3;
```



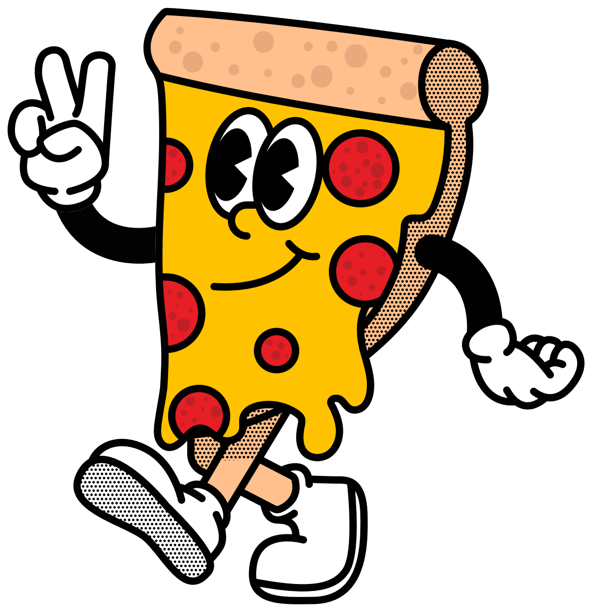


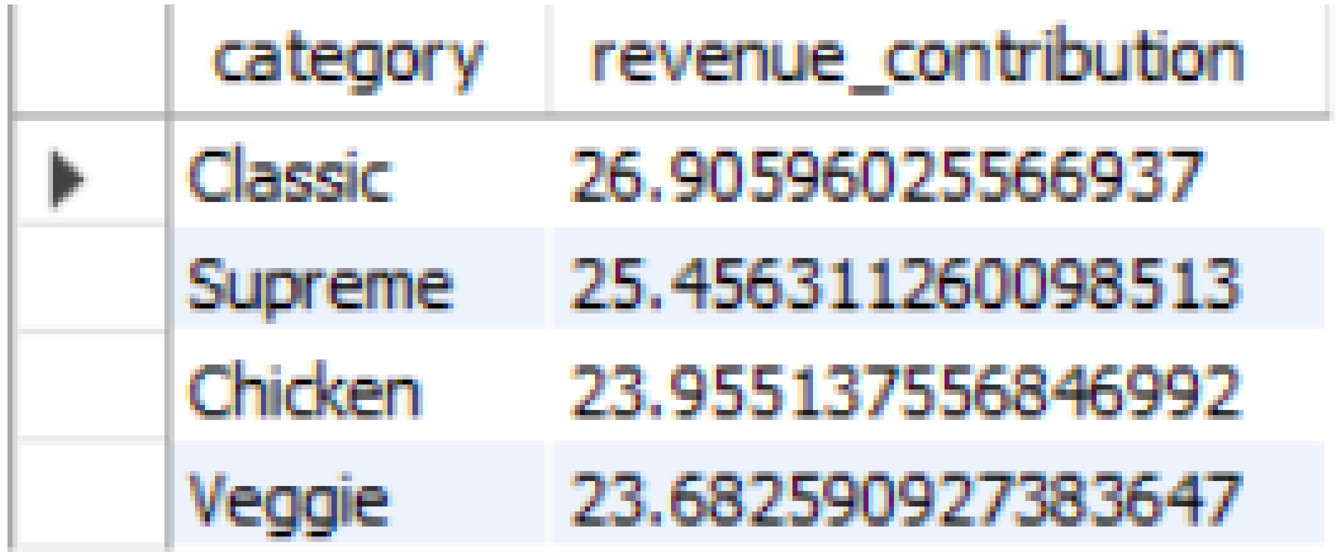
	name	revenue
▶	The Thai Chicken Pizza	43434.25
	The Barbecue Chicken Pizza	42768
	The California Chicken Pizza	41409.5

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



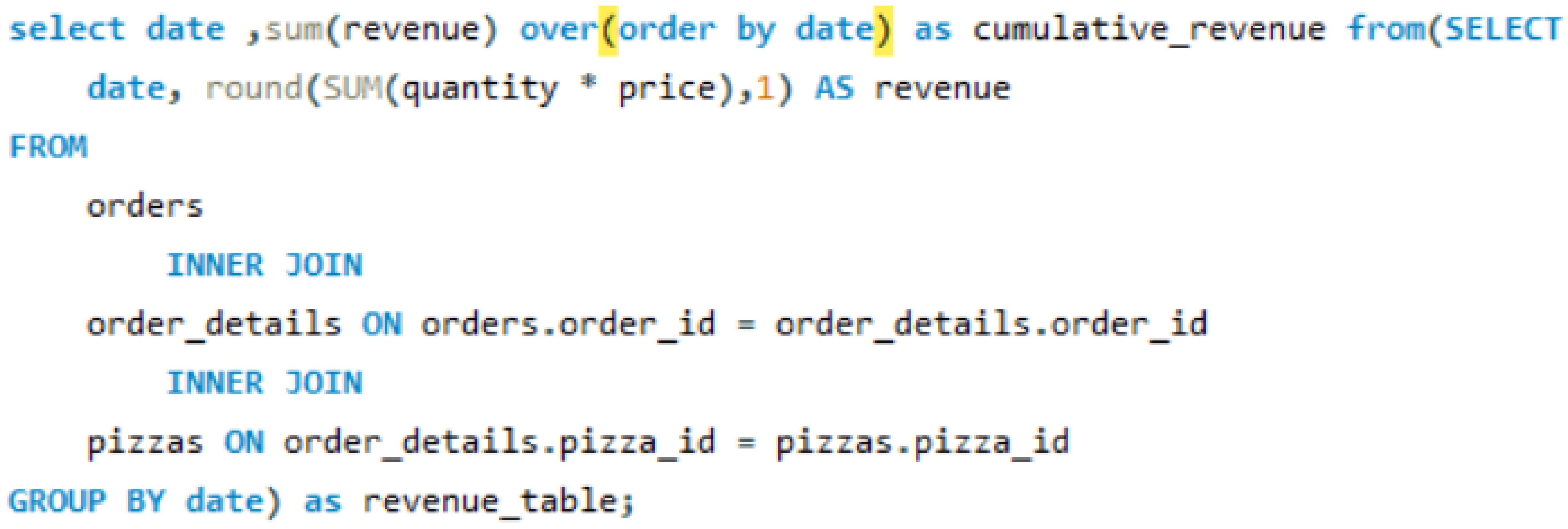
```sql
SELECT
    category,
    (SUM(quantity * price) / (SELECT
            SUM((quantity * price))
        FROM
            order_details
                INNER JOIN
            pizzas ON order_details.pizza_id = pizzas.pizza_id)) * 100 AS revenue_contribution
FROM
    order_details
        INNER JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id
        INNER JOIN
    pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
GROUP BY category
ORDER BY revenue_contribution DESC;
```



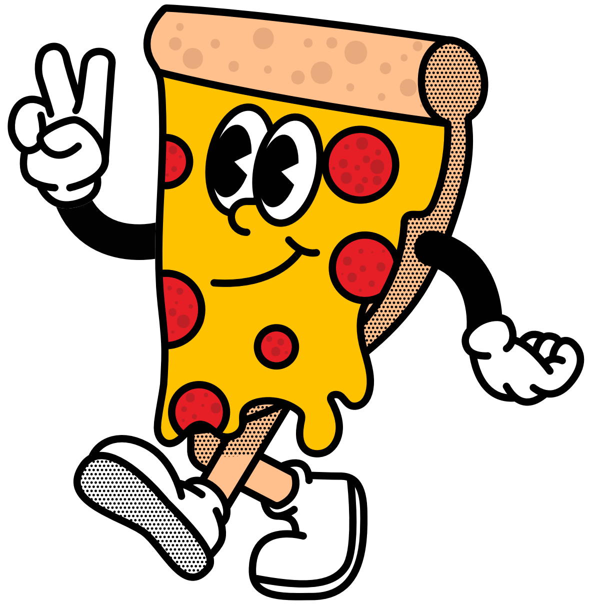


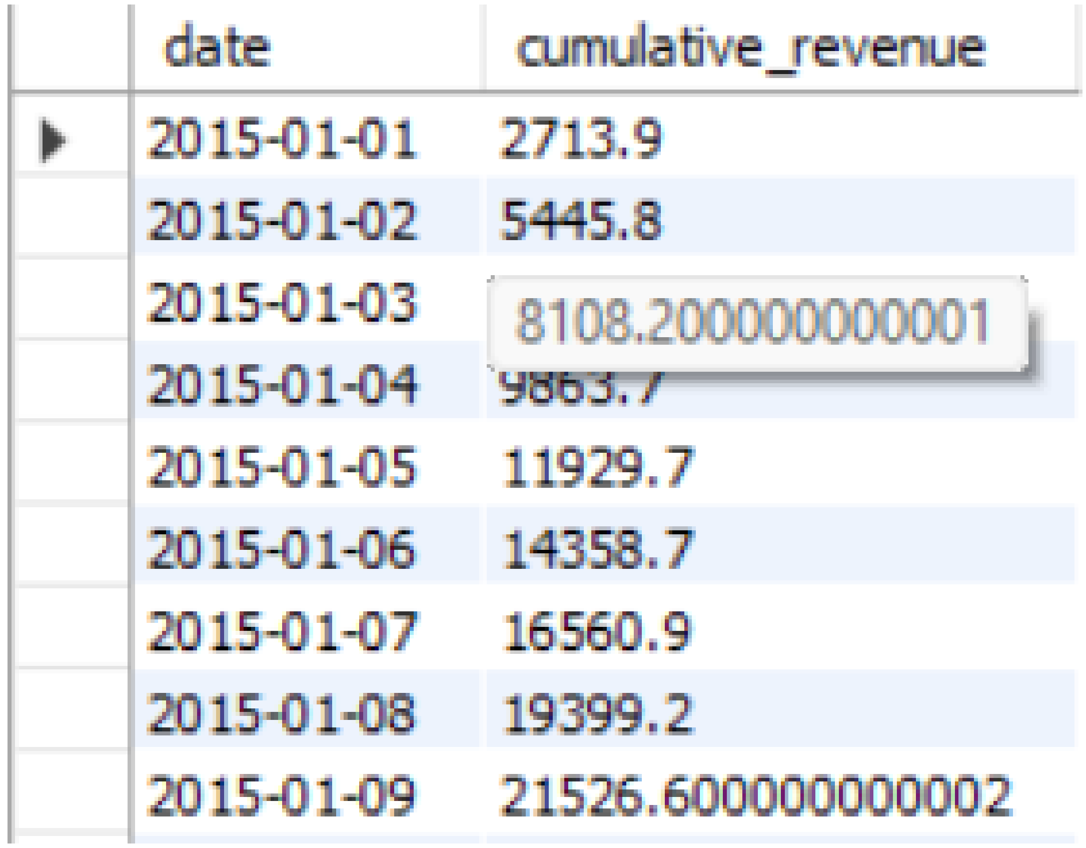
category	revenue_contribution
Classic	26.9059602556937
Supreme	25.4563112600985 13
Chicken	23.9551375568469 92
Veggie	23.6825909273836 47

Analyze the cumulative revenue generated over time.



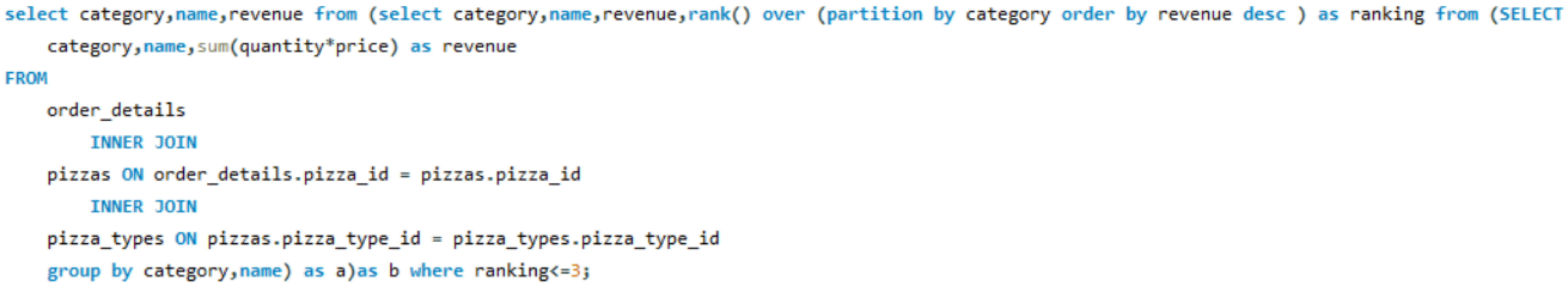
```sql
select date ,sum(revenue) over(order by date) as cumulative_revenue from(SELECT
    date, round(SUM(quantity * price),1) AS revenue
FROM
    orders
        INNER JOIN
    order_details ON orders.order_id = order_details.order_id
        INNER JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id
GROUP BY date) as revenue_table;
```



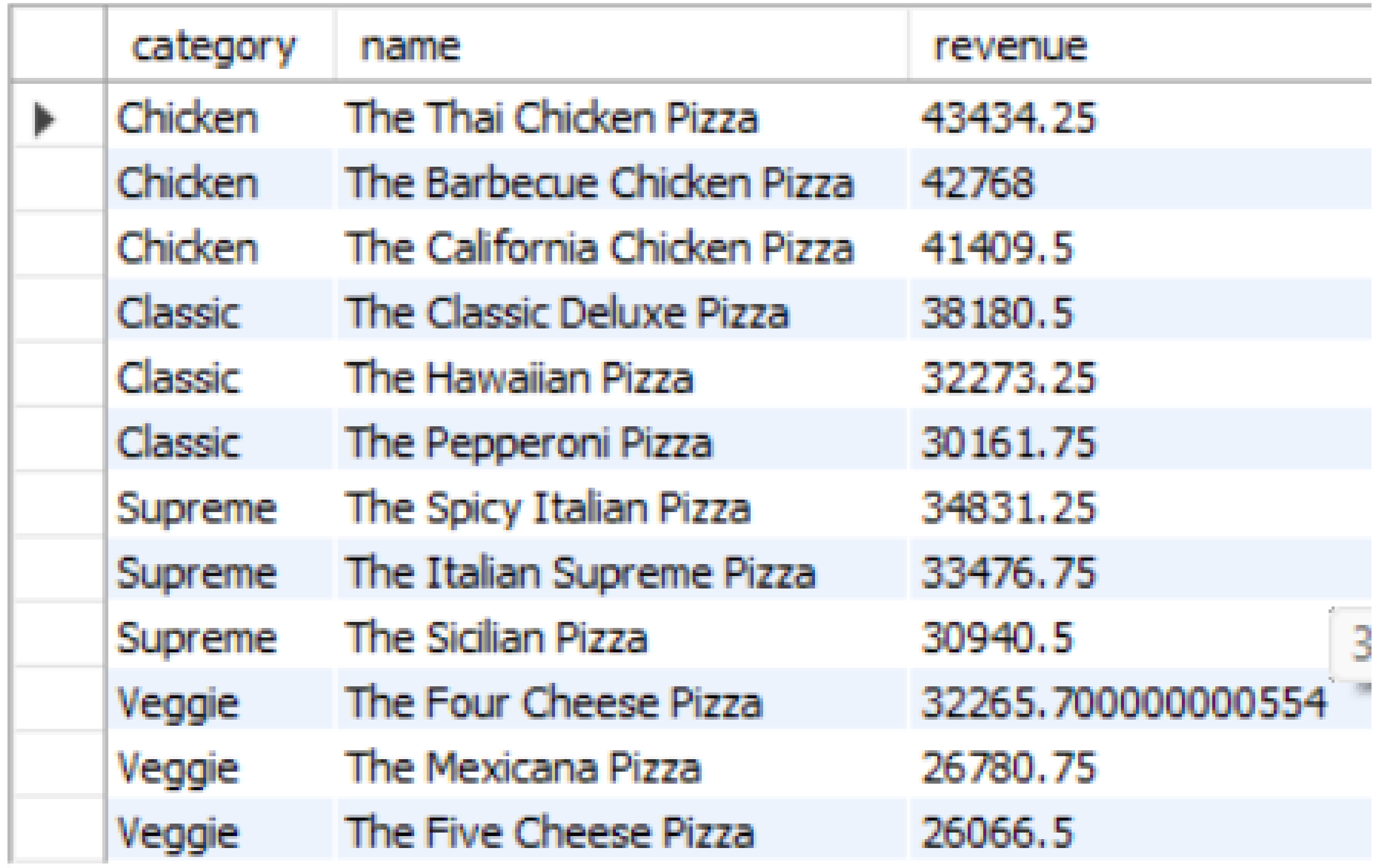


date	cumulative_revenue
2015-01-01	2713.9
2015-01-02	5445.8
2015-01-03	8108.200000000001
2015-01-04	9863.7
2015-01-05	11929.7
2015-01-06	14358.7
2015-01-07	16560.9
2015-01-08	19399.2
2015-01-09	21526.600000000002

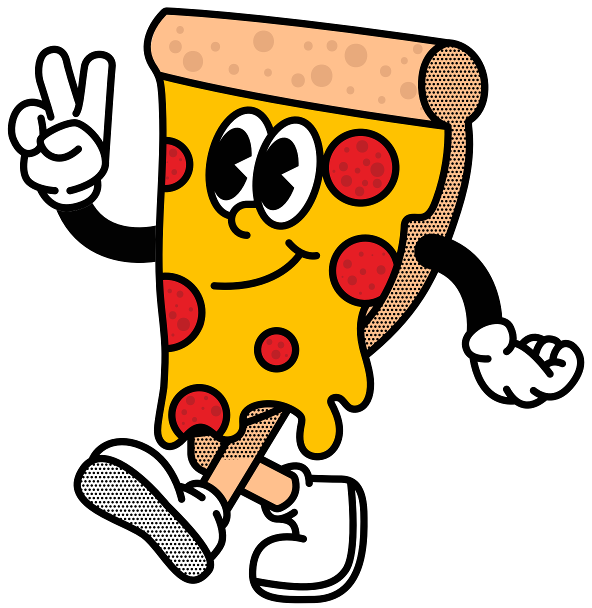
Determine the top 3 most ordered pizza types based on revenue for each pizza category.



```
select category,name,revenue from (select category,name,revenue,rank() over (partition by category order by revenue desc ) as ranking from (SELECT
    category,name,sum(quantity*price) as revenue
FROM
    order_details
        INNER JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id
        INNER JOIN
    pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
    group by category,name) as a)as b where ranking<=3;
```



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	32265.70000000554
Veggie	The Mexicana Pizza	26780.75
Veggie	The Five Cheese Pizza	26066.5



PIZZA PARTY!