PIZZA SALES (SQL)

By Krishna Mohan Yadav m

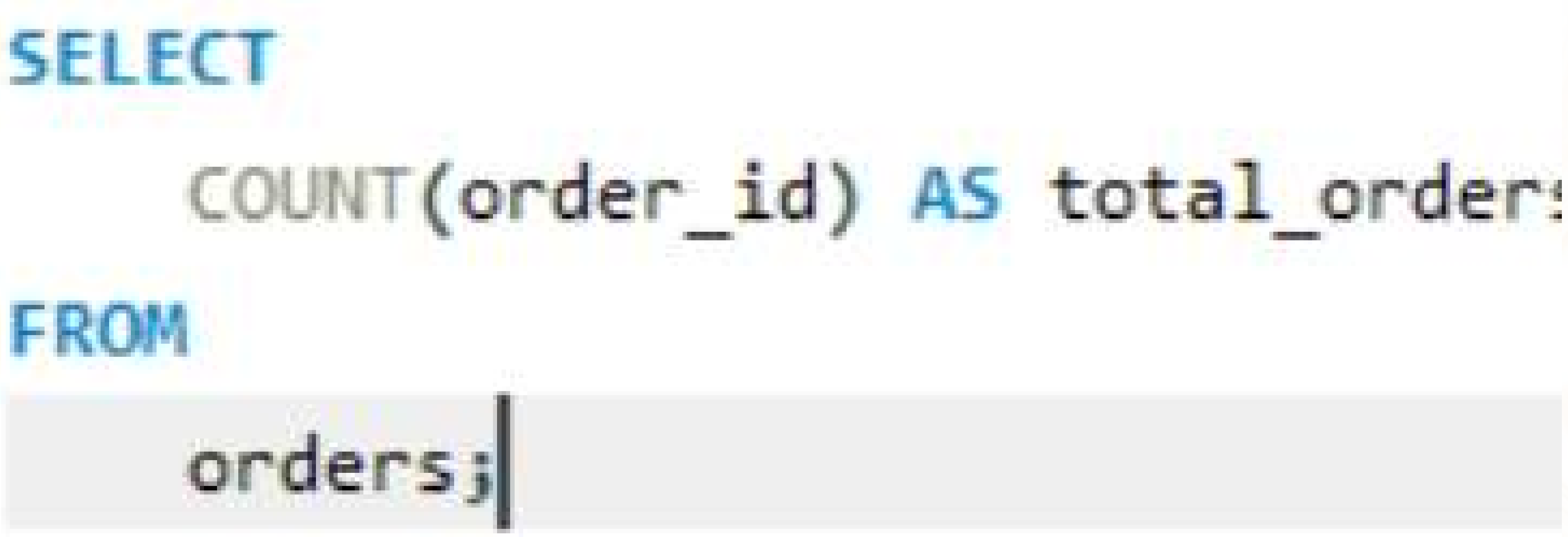
HELLO!

Hello, my name is Krishna Mohan Yadav. In this project, I have resolved several questions related to pizza sales by using multiple SQL queries.

PROJECT GOALS

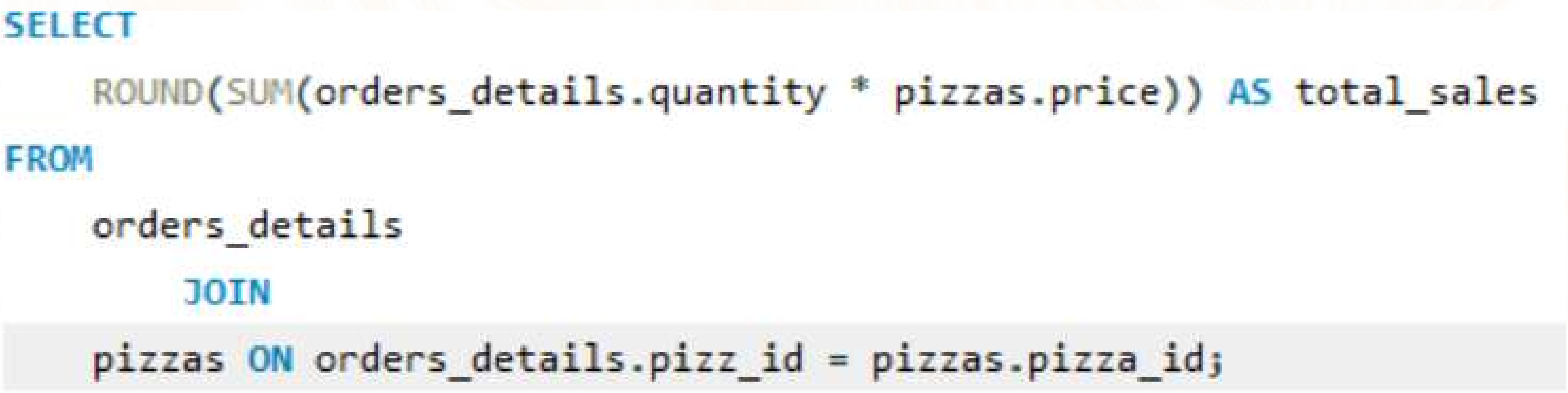
- The goal of this project is to analyze pizza sales data using SQL. We aim to identify sales trends, customer preferences, and areas for improvement. The insights gained will help optimize sales strategies and improve business performance.

1.RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.



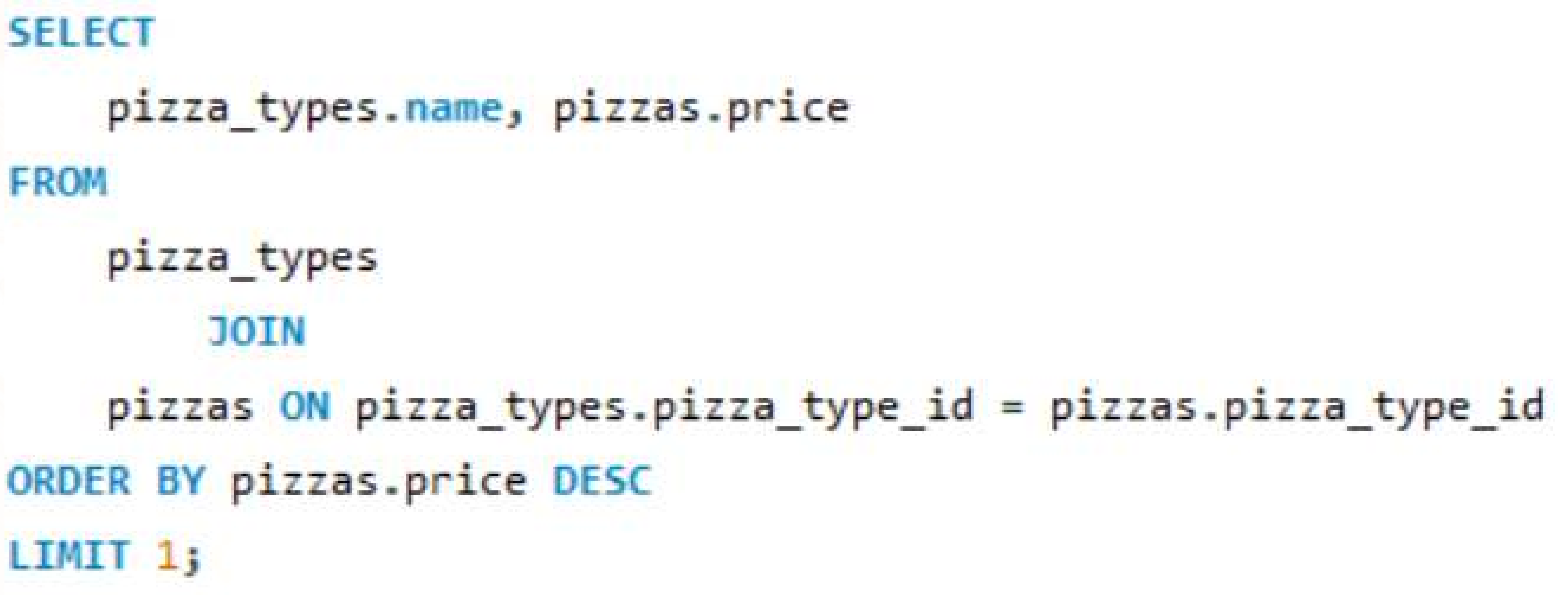
```
SELECT
    COUNT(order_id) AS total_orders
FROM
    orders;
```

2 .CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.



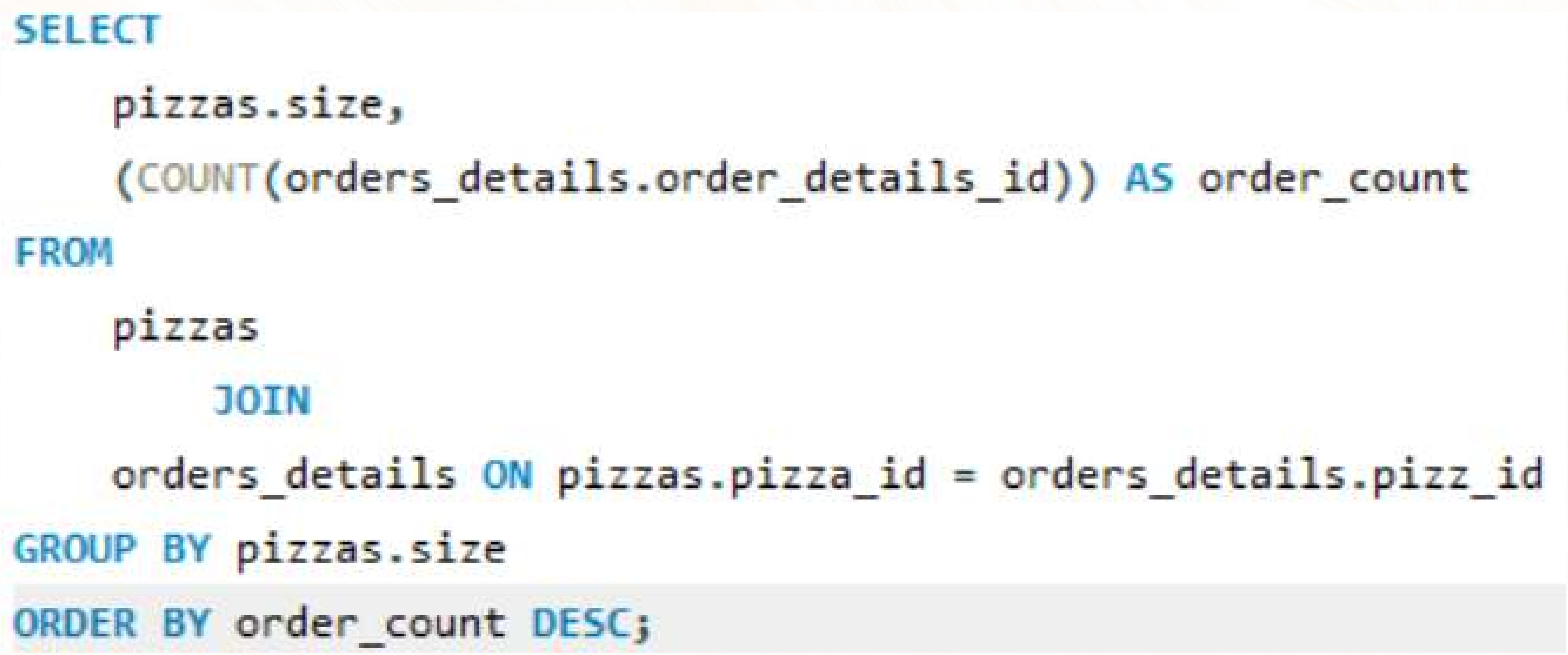
```sql
SELECT
    ROUND(SUM(orders_details.quantity * pizzas.price)) AS total_sales
FROM
    orders_details
        JOIN
    pizzas ON orders_details.pizz_id = pizzas.pizza_id;
```

3.IDENTIFY THE HIGHEST-PRICED PIZZA.



```sql
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;
```

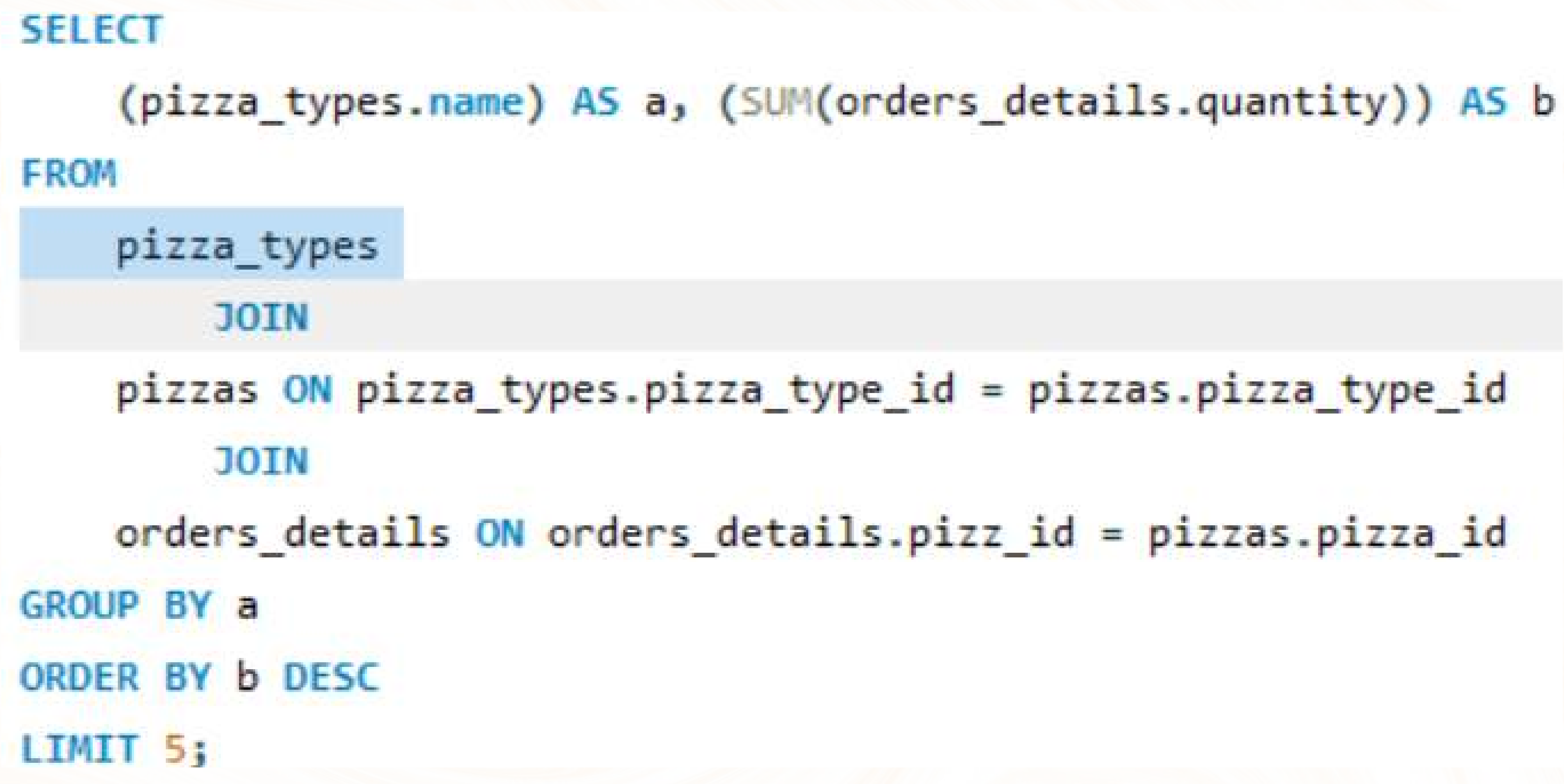
4.IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.



```
SELECT
    pizzas.size,
    (COUNT(orders_details.order_details_id)) AS order_count
FROM
    pizzas
        JOIN
    orders_details ON pizzas.pizza_id = orders_details.pizz_id
GROUP BY pizzas.size
ORDER BY order_count DESC;
```

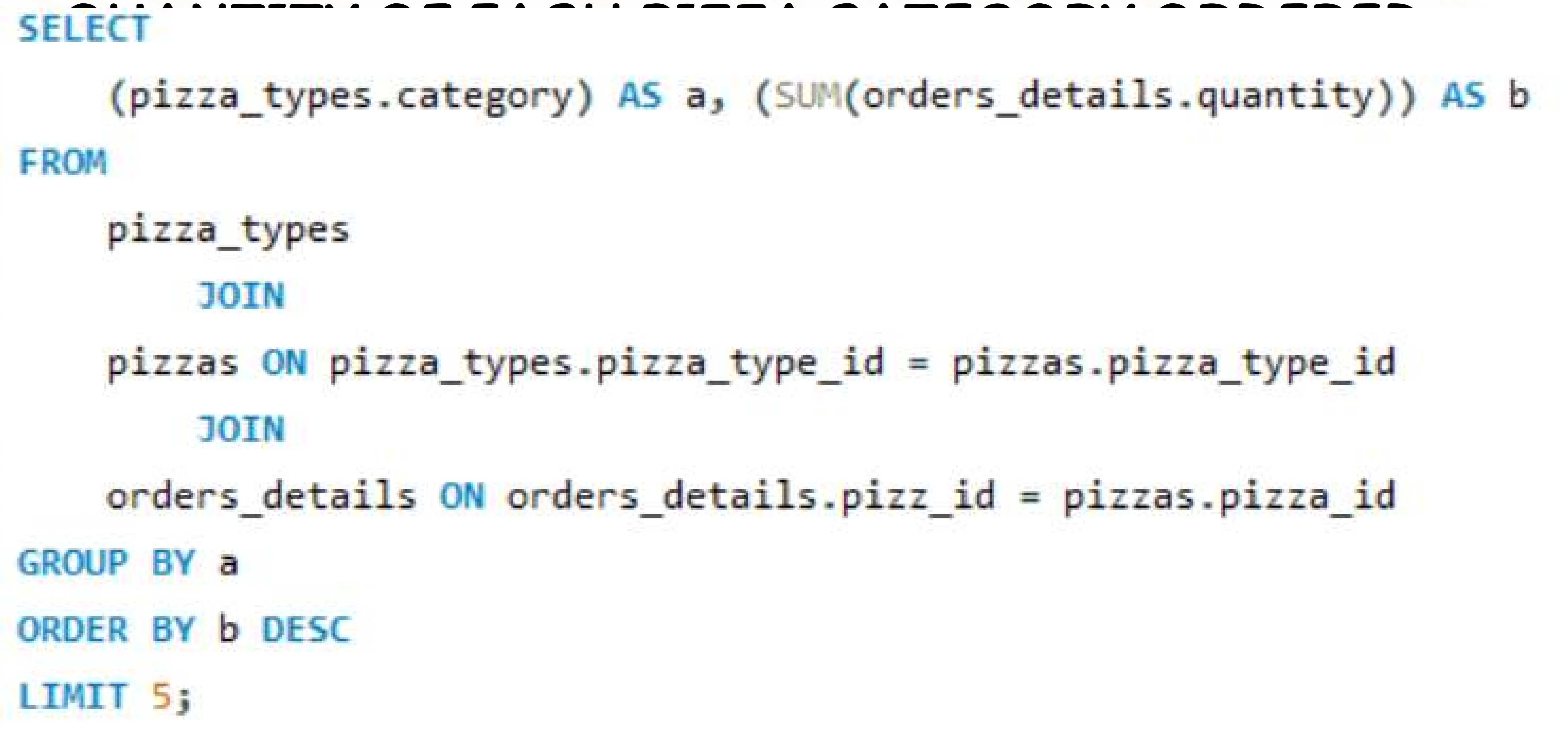
Identify the most common pizza size ordered.

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



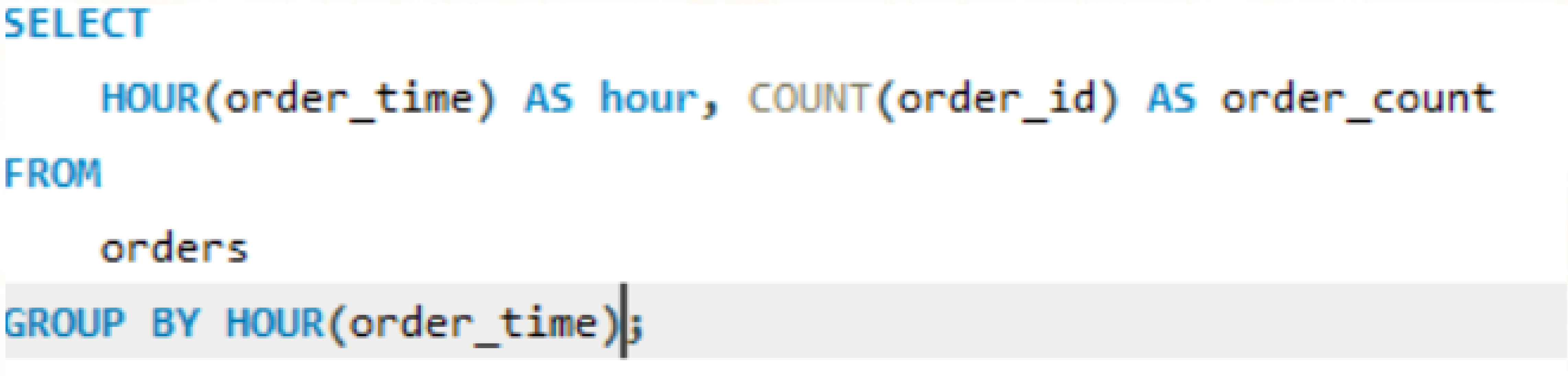
```sql
SELECT
    (pizza_types.name) AS a, (SUM(orders_details.quantity)) AS b
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    orders_details ON orders_details.pizz_id = pizzas.pizza_id
GROUP BY a
ORDER BY b DESC
LIMIT 5;
```

6.JOIN THE NECESSARY TABLES TO FIND THE TOTAL QUANTITY OF EACH PIZZA CATEGORY ORDERED.



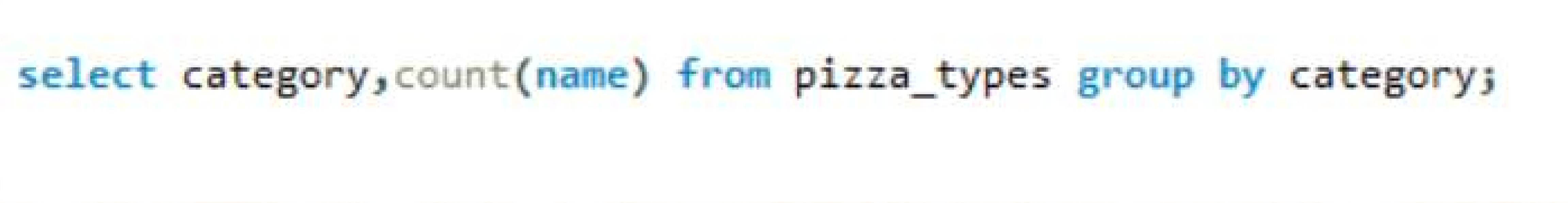
```
SELECT
    (pizza_types.category) AS a, (SUM(orders_details.quantity)) AS b
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    orders_details ON orders_details.pizz_id = pizzas.pizza_id
GROUP BY a
ORDER BY b DESC
LIMIT 5;
```

7.DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.



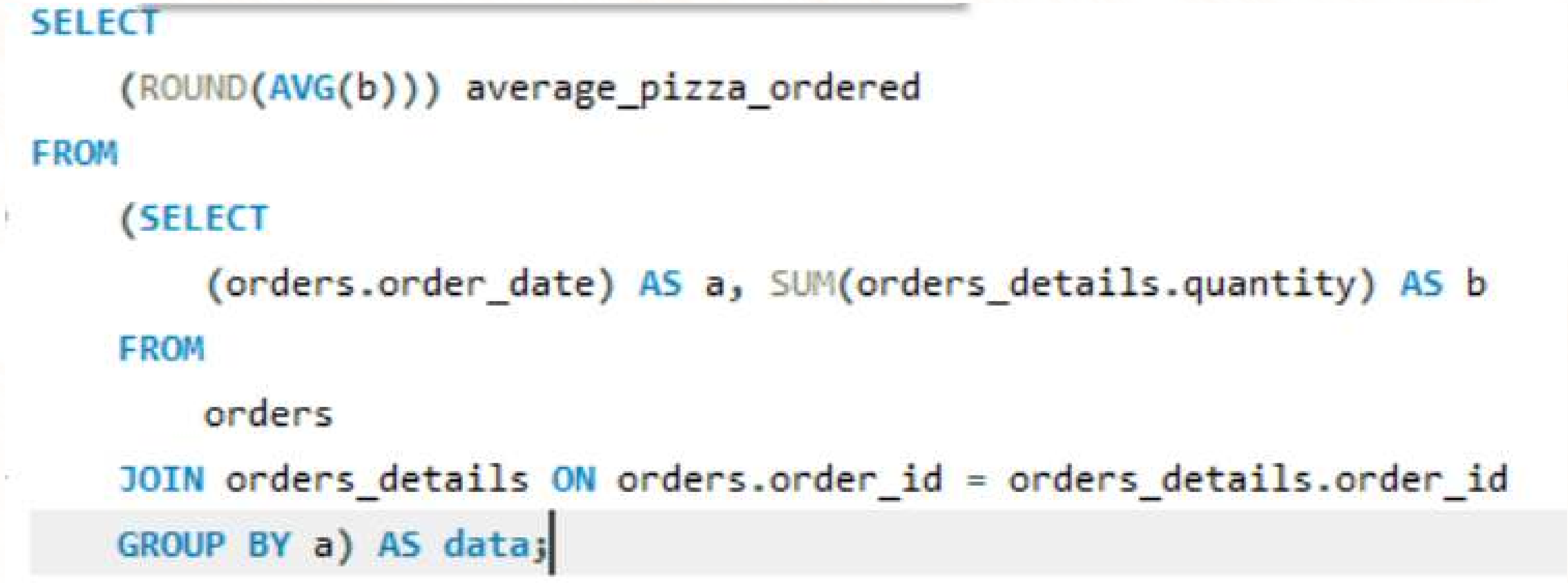
```sql
SELECT
    HOUR(order_time) AS hour, COUNT(order_id) AS order_count
FROM
    orders
GROUP BY HOUR(order_time);
```

8.JOIN RELEVANT TABLES TO FIND THE CATEGORY-WISE DISTRIBUTION OF PIZZAS.



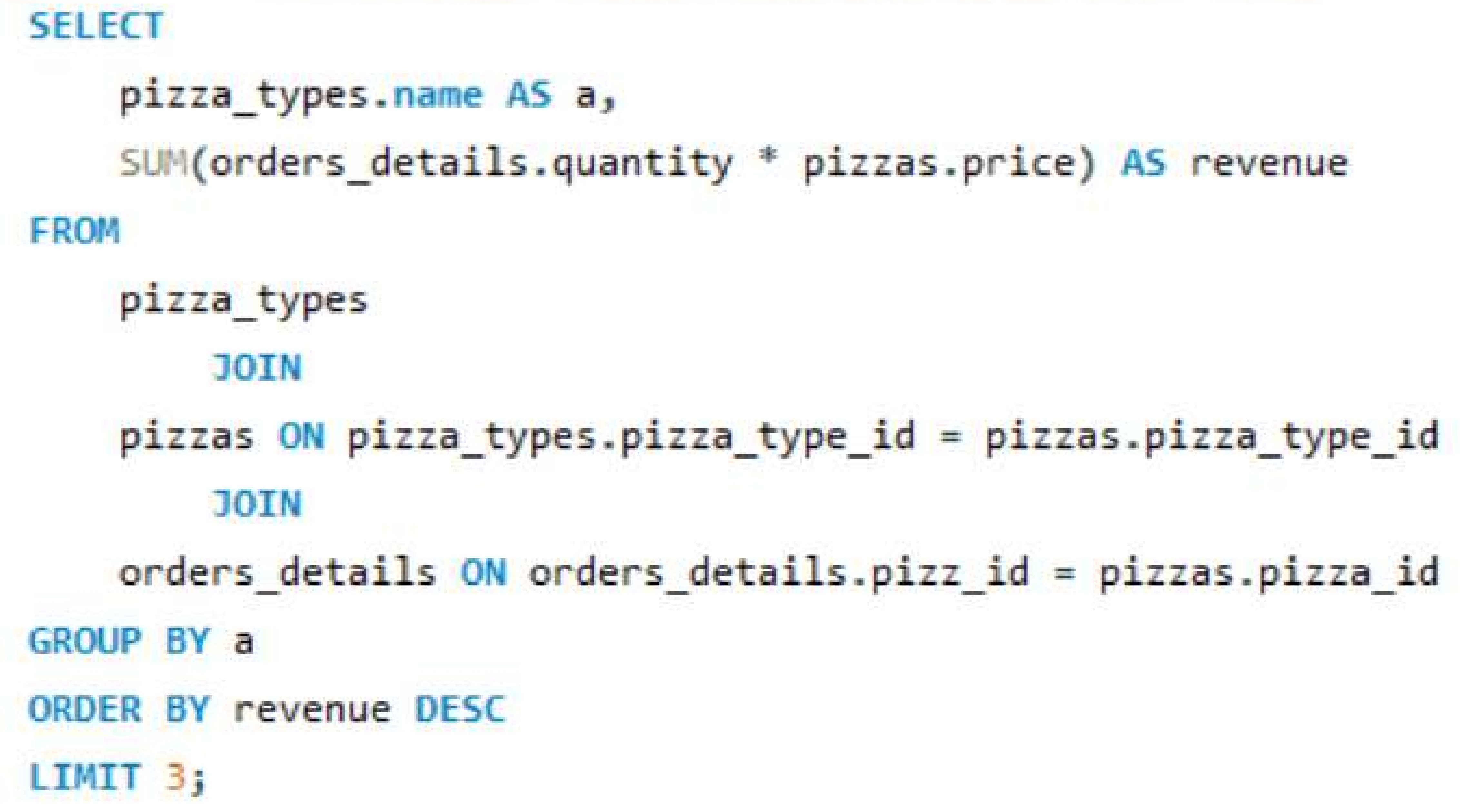
```
select category,count(name) from pizza_types group by category;
```

9.GROUP THE ORDERS BY DATE AND CALCULATE THE AVERAGE NUMBER OF PIZZAS ORDERED PER DAY.



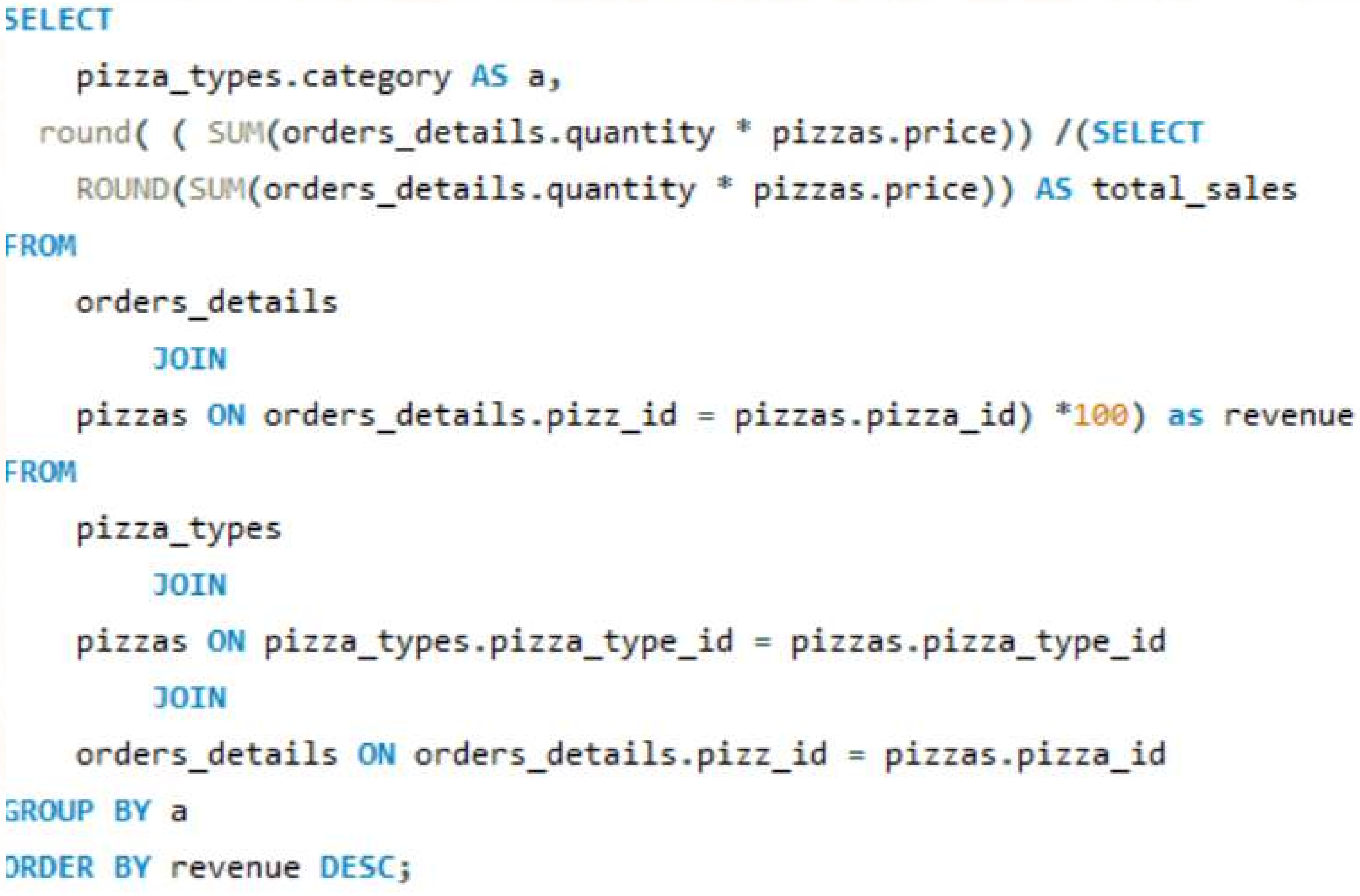
```sql
SELECT
    (ROUND(AVG(b))) average_pizza_ordered
FROM
    (SELECT
        (orders.order_date) AS a, SUM(orders_details.quantity) AS b
    FROM
        orders
    JOIN orders_details ON orders.order_id = orders_details.order_id
    GROUP BY a) AS data;
```

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



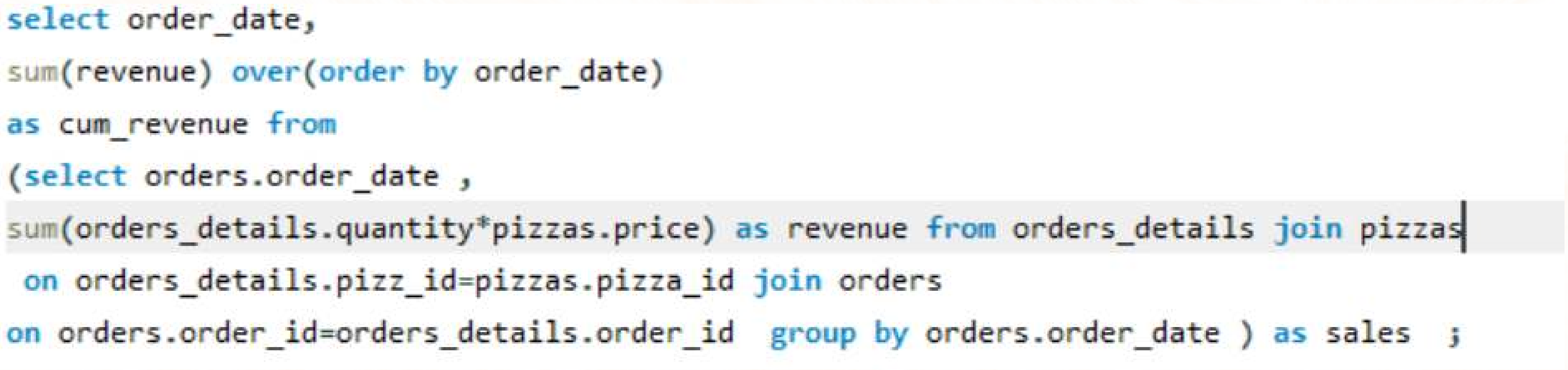
```
SELECT
    pizza_types.name AS a,
    SUM(orders_details.quantity * pizzas.price) AS revenue
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    orders_details ON orders_details.pizz_id = pizzas.pizza_id
GROUP BY a
ORDER BY revenue DESC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

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



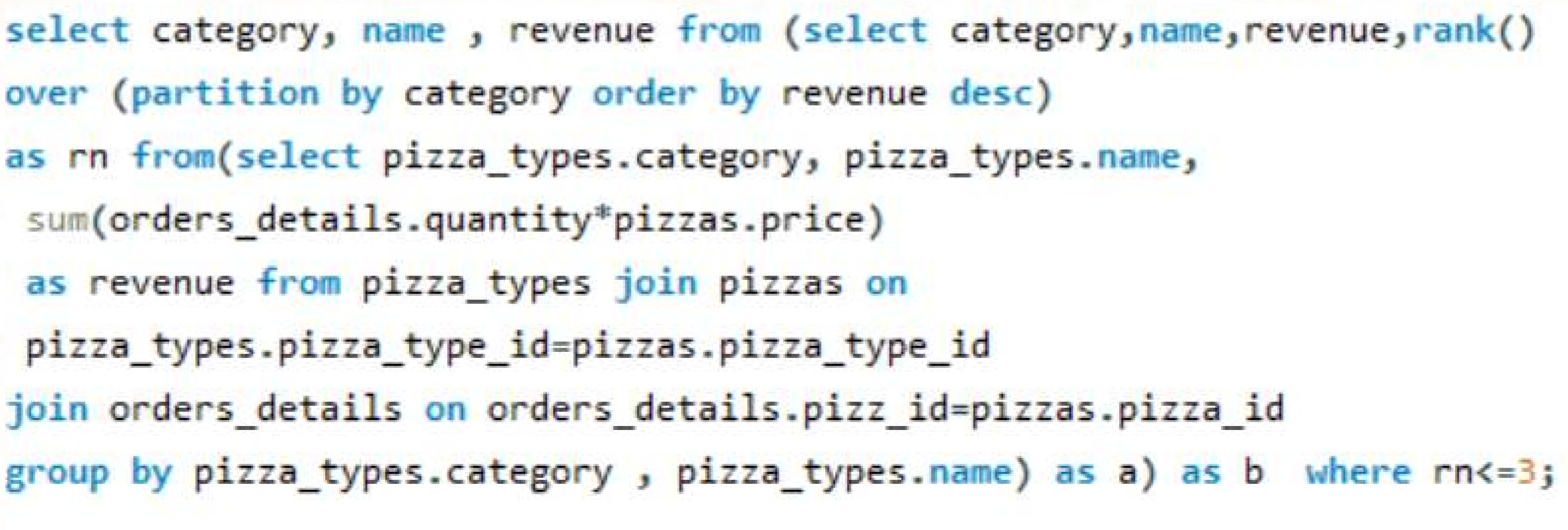
```
SELECT
    pizza_types.category AS a,
  round( ( SUM(orders_details.quantity * pizzas.price)) /(SELECT
    ROUND(SUM(orders_details.quantity * pizzas.price)) AS total_sales
FROM
    orders_details
        JOIN
    pizzas ON orders_details.pizz_id = pizzas.pizza_id) *100) as revenue
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    orders_details ON orders_details.pizz_id = pizzas.pizza_id
GROUP BY a
ORDER BY revenue DESC;
```

12.ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME.



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

13.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 rn from(select pizza_types.category, pizza_types.name,
 sum(orders_details.quantity*pizzas.price)
 as revenue from pizza_types join pizzas on
 pizza_types.pizza_type_id=pizzas.pizza_type_id
join orders_details on orders_details.pizz_id=pizzas.pizza_id
group by pizza_types.category , pizza_types.name) as a) as b  where rn<=3;
```

THANK YOU