PIZZA_SALES _ANALYSI

Objective: The objective of this project is to a data from a pizza_hut restaurant chain to gain customer preferences, sales trends, popular me hours, and other relevant metrics.

DATA SOURCES:

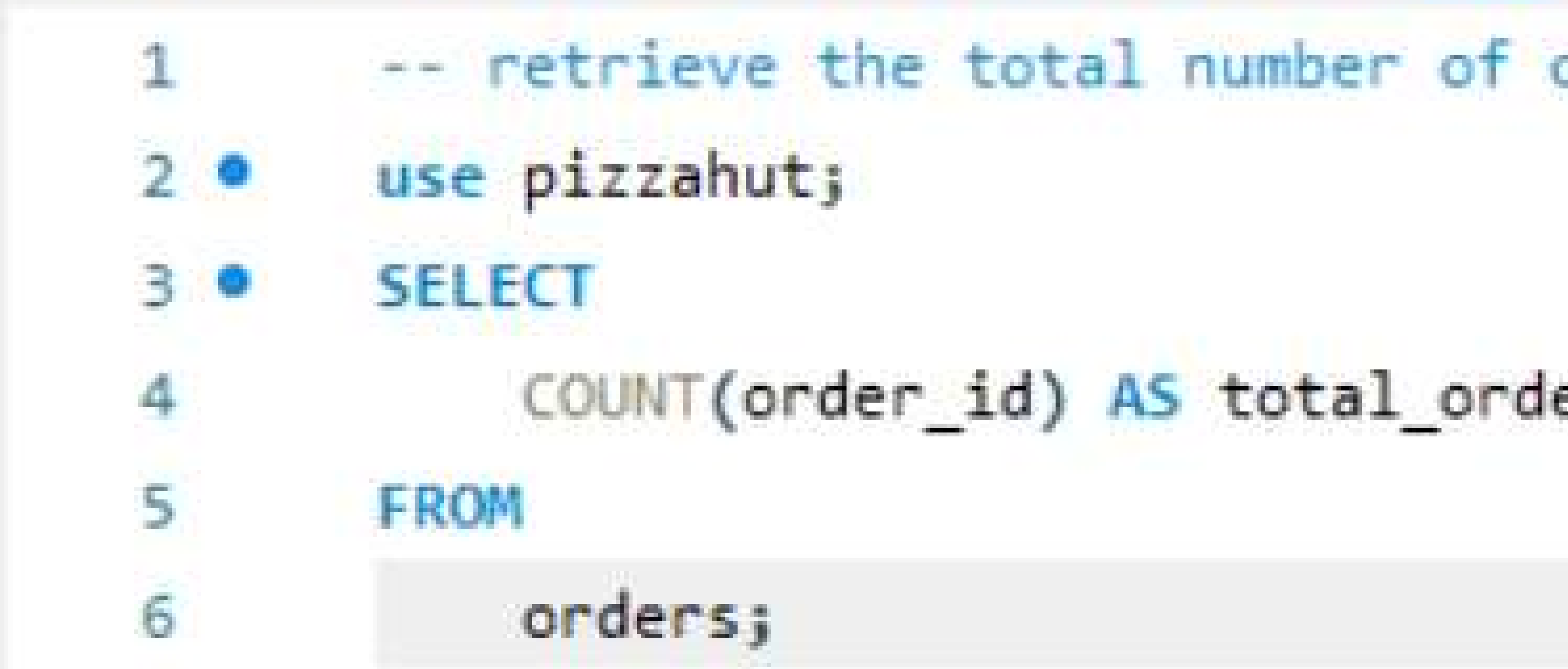
1.Orders Table: The ordes table contains orders_id, D the time of order placed

2.Orders_details Table: The table contain details of four column having order_details_id,order_id, Pizza_id

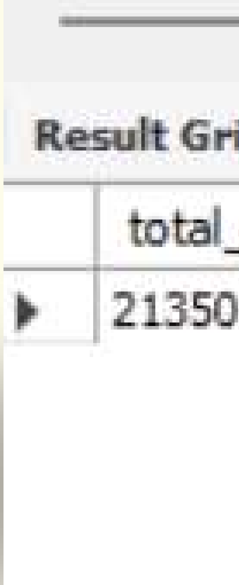
3.Pizza_types Table: the table contains varieties in p specifies the type and categories of pizza.The coloum pizza_type_id,name, categories and pizza_id.

4.pizzas Table: The table contains the coloumn pizza and size.

1.Retrieve the total number of ord



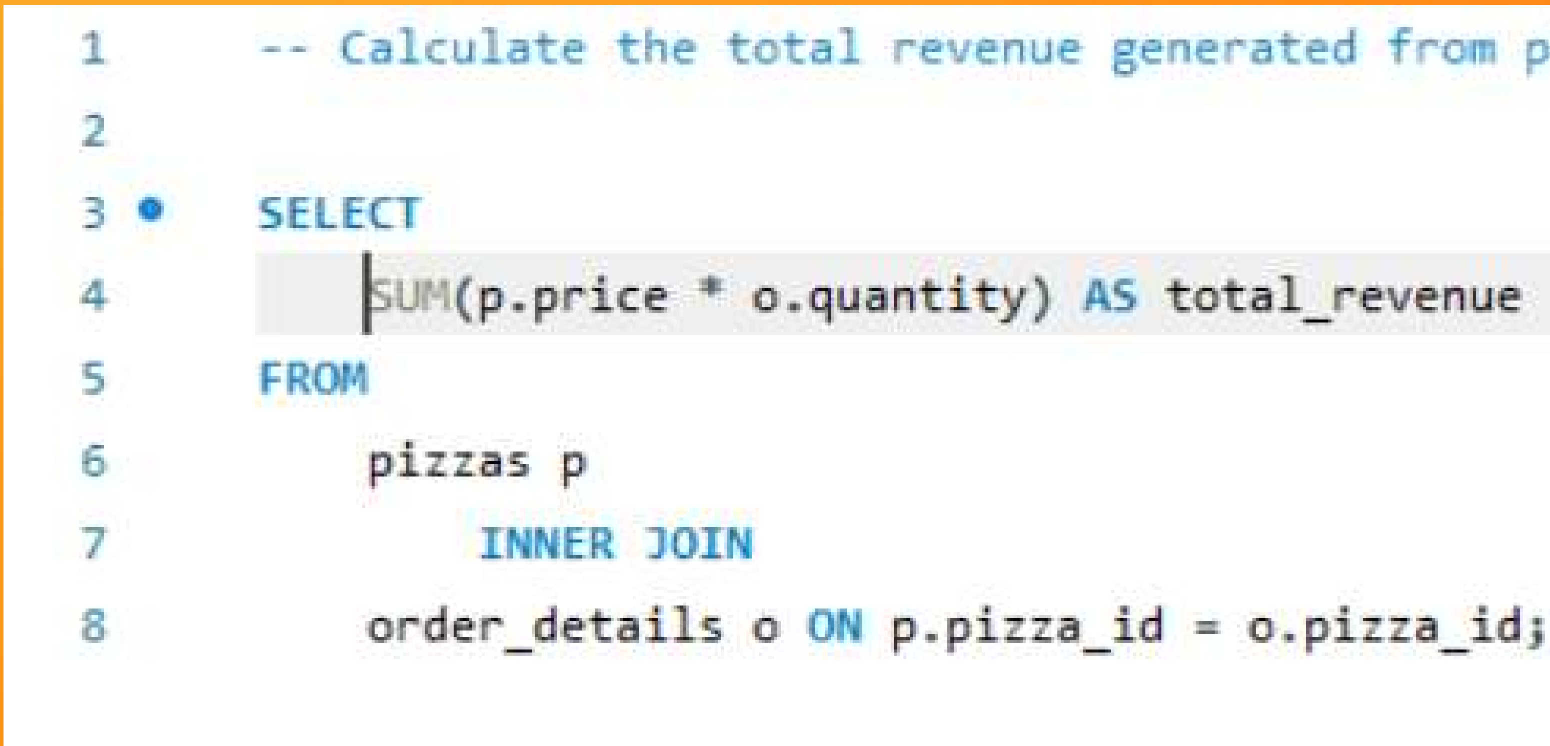
```
-- retrieve the total number of c
use pizzahut;
SELECT
    COUNT(order_id) AS total_orde
FROM
    orders;
```



Result Gri

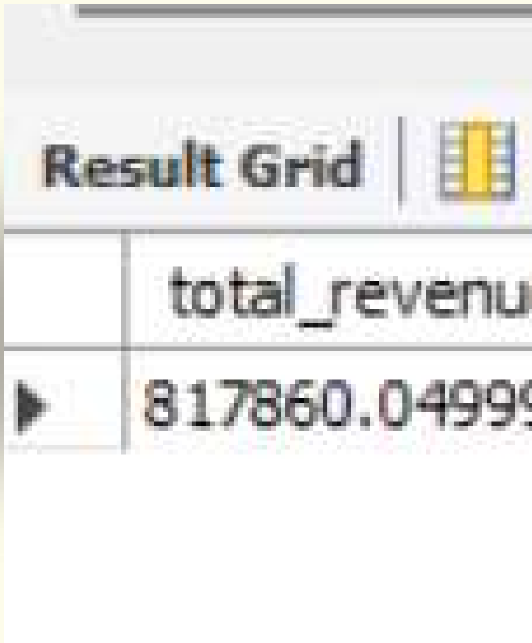
total_
21350

2.Calculate the total re generated from pizza



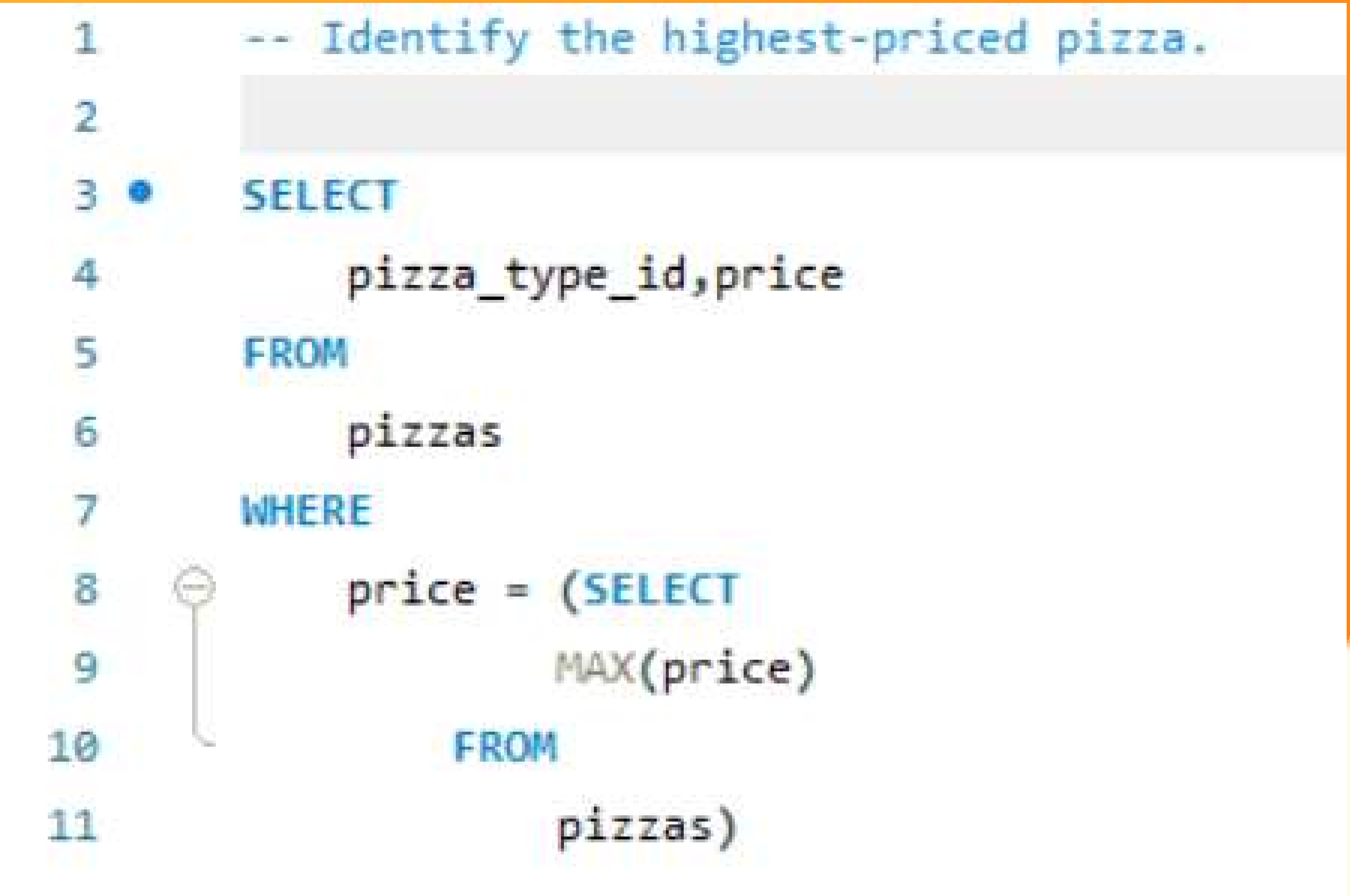
```
-- Calculate the total revenue generated from p

SELECT
    SUM(p.price * o.quantity) AS total_revenue
FROM
    pizzas p
        INNER JOIN
    order_details o ON p.pizza_id = o.pizza_id;
```



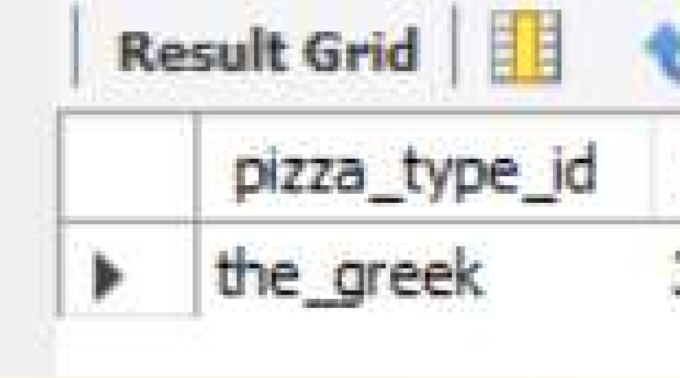
```
total_revenu
817860.0499
```

3.Identify the highest-price



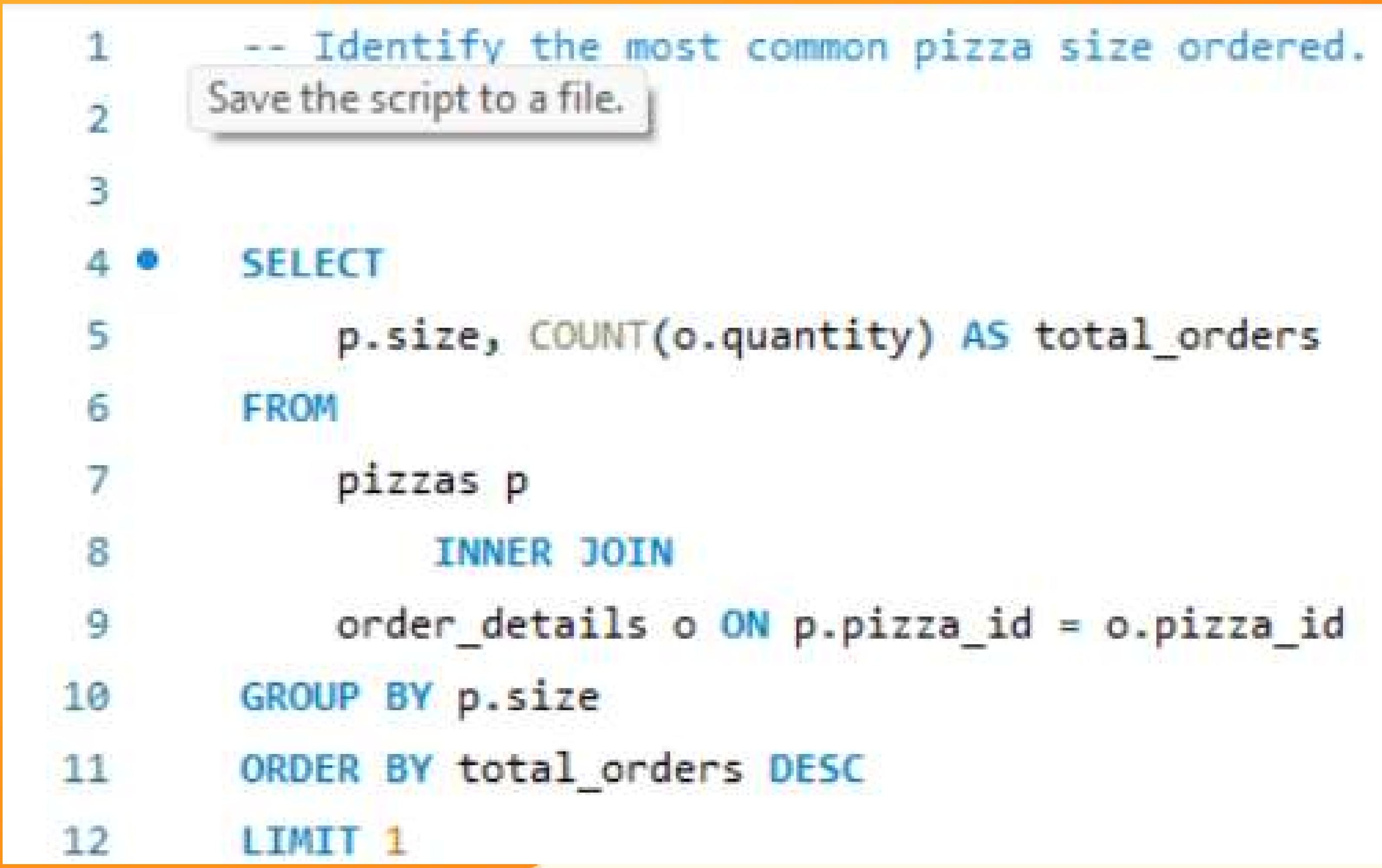
```sql
-- Identify the highest-priced pizza.

SELECT
    pizza_type_id,price
FROM
    pizzas
WHERE
    price = (SELECT
            MAX(price)
        FROM
            pizzas)
```



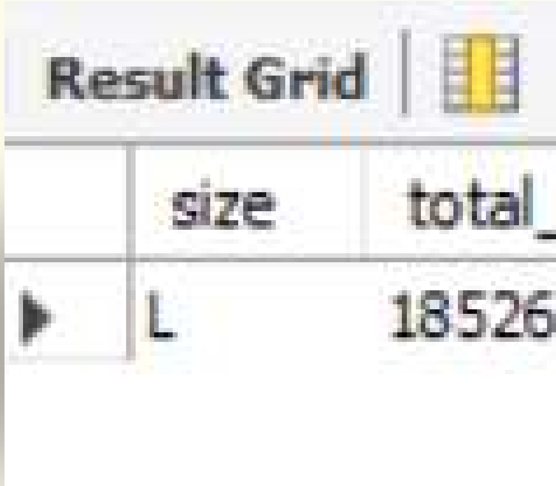
	pizza_type_id
▶	the_greek

4.Identify the most commo size ordered.



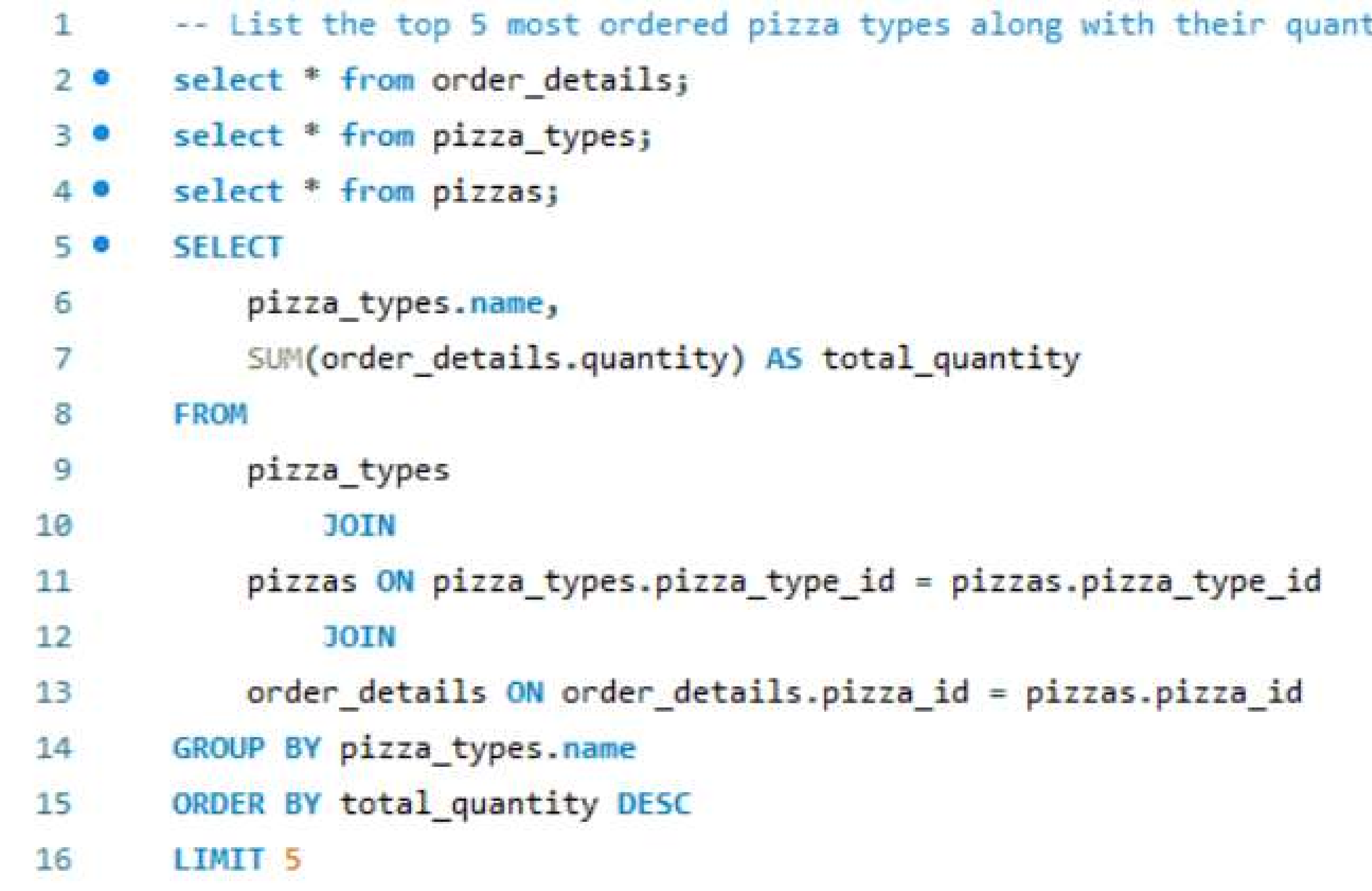
```
-- Identify the most common pizza size ordered.


SELECT
    p.size, COUNT(o.quantity) AS total_orders
FROM
    pizzas p
        INNER JOIN
    order_details o ON p.pizza_id = o.pizza_id
GROUP BY p.size
ORDER BY total_orders DESC
LIMIT 1
```

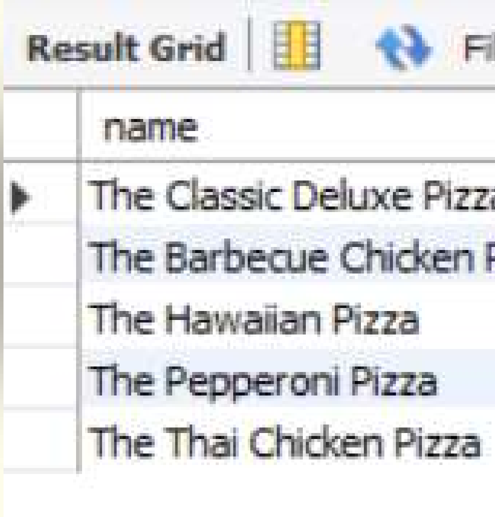


```
   size   total_
▶  L      18526
```

5.List the top 5 most order types along with their qua

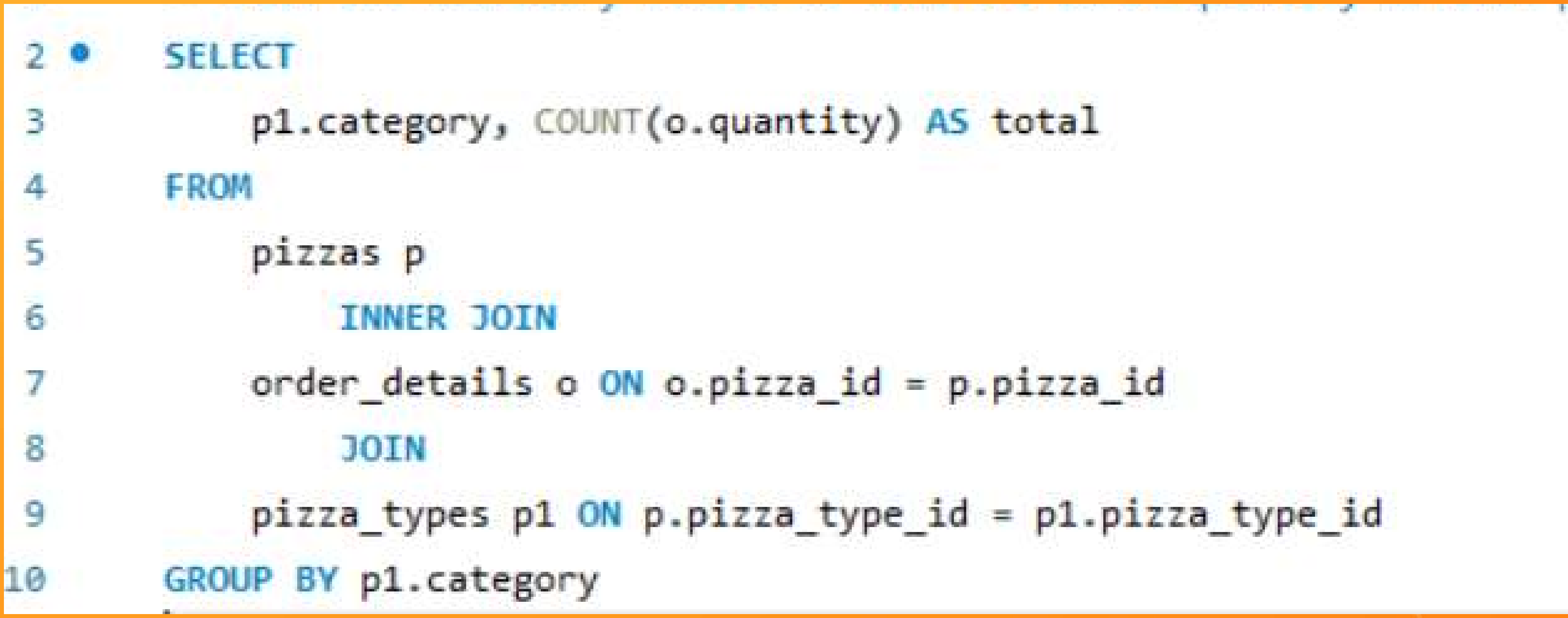


```
-- List the top 5 most ordered pizza types along with their quant
select * from order_details;
select * from pizza_types;
select * from pizzas;
SELECT
    pizza_types.name,
    SUM(order_details.quantity) AS total_quantity
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY total_quantity DESC
LIMIT 5
```

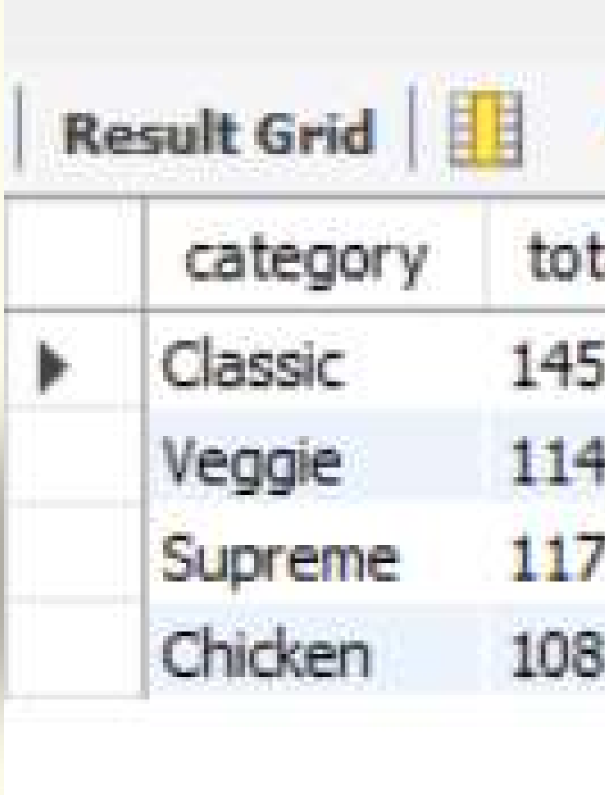


name
The Classic Deluxe Pizz
The Barbecue Chicken P
The Hawaiian Pizza
The Pepperoni Pizza
The Thai Chicken Pizza

6.Join the necessary tables to find t quantity of each pizza category o

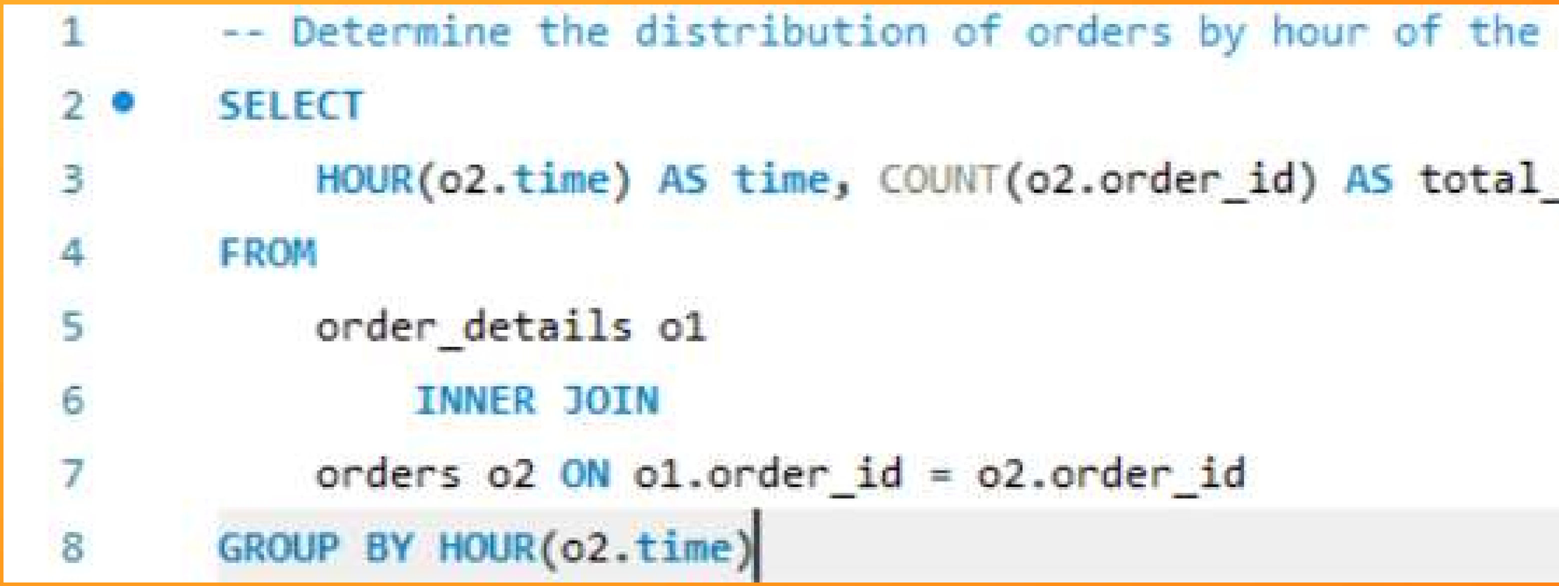


```
SELECT
    p1.category, COUNT(o.quantity) AS total
FROM
    pizzas p
        INNER JOIN
    order_details o ON o.pizza_id = p.pizza_id
        JOIN
    pizza_types p1 ON p.pizza_type_id = p1.pizza_type_id
GROUP BY p1.category
```

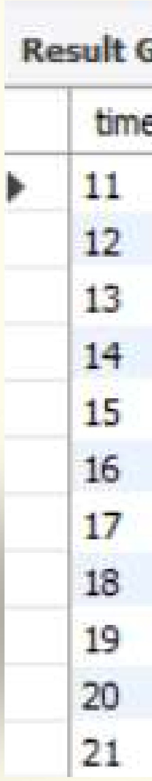


category	tot
Classic	145
Veggie	114
Supreme	117
Chicken	108

7.Determine the distribution by hour of the day.



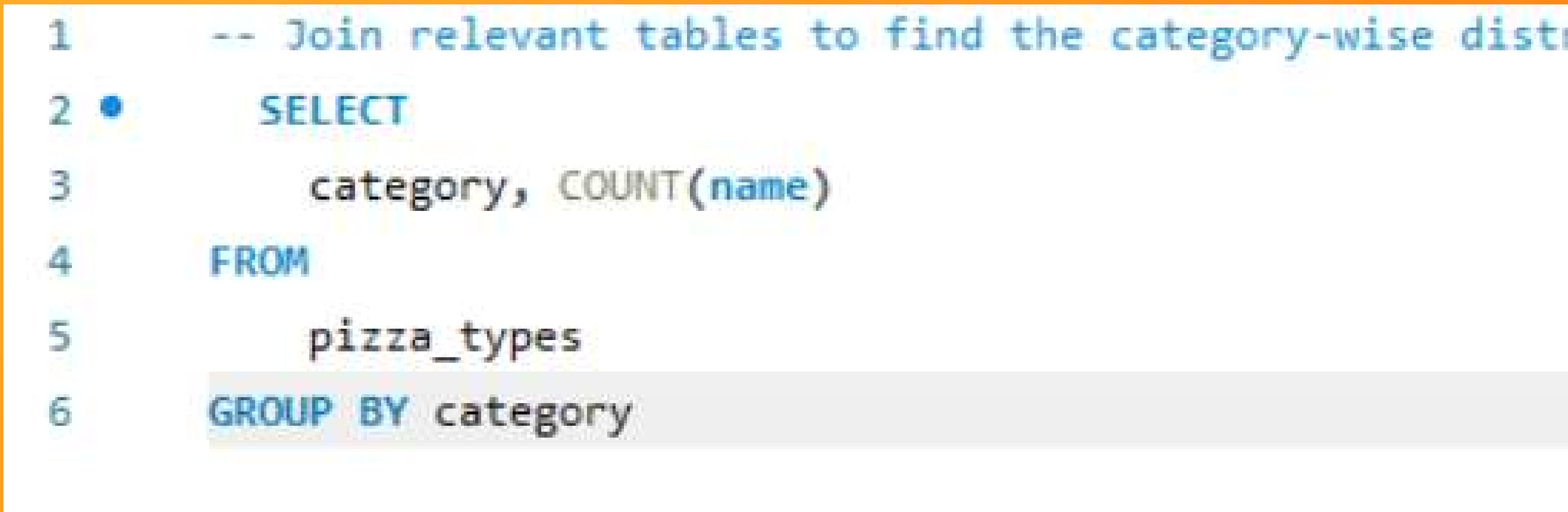
```sql
-- Determine the distribution of orders by hour of the
SELECT
    HOUR(o2.time) AS time, COUNT(o2.order_id) AS total_
FROM
    order_details o1
        INNER JOIN
    orders o2 ON o1.order_id = o2.order_id
GROUP BY HOUR(o2.time)
```



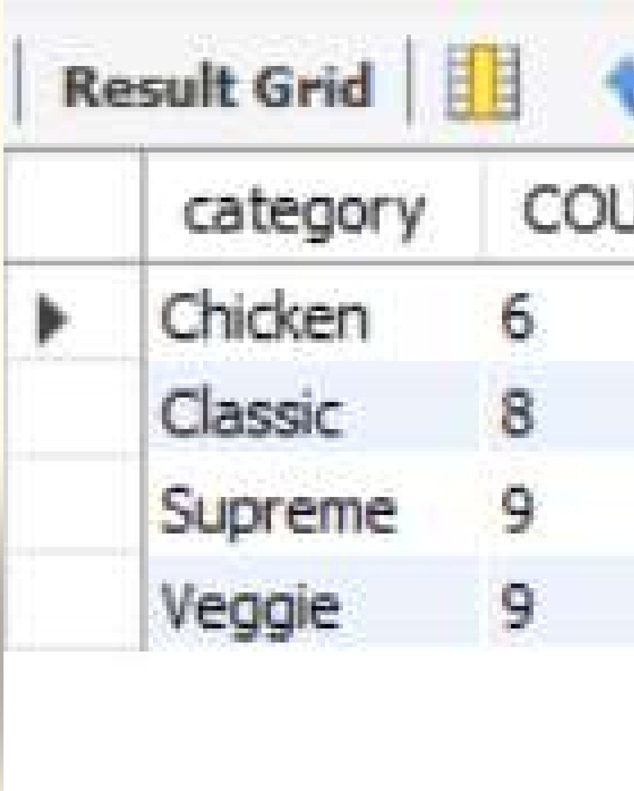
Result G

time
11
12
13
14
15
16
17
18
19
20
21

8.Join relevant tables to find the wise distribution of pizz

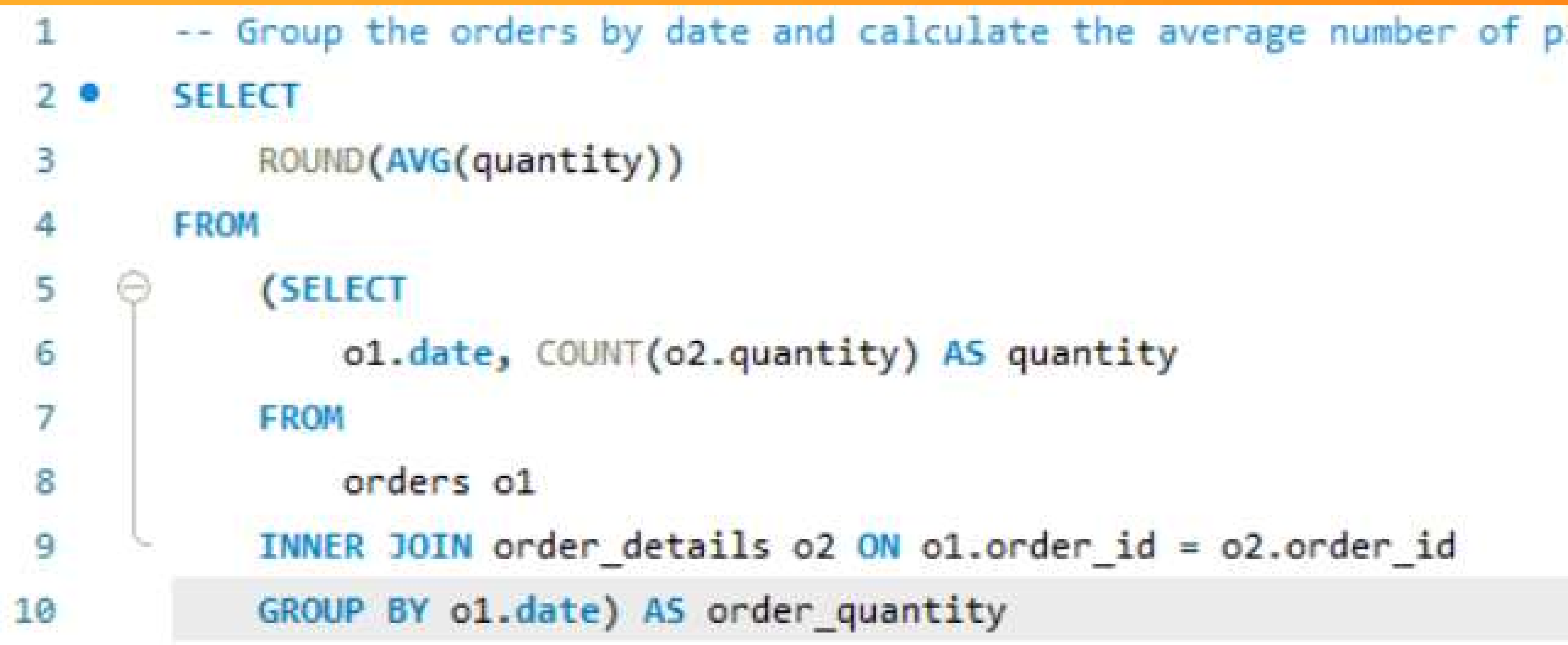


```
-- Join relevant tables to find the category-wise dist
  SELECT
    category, COUNT(name)
FROM
    pizza_types
GROUP BY category
```

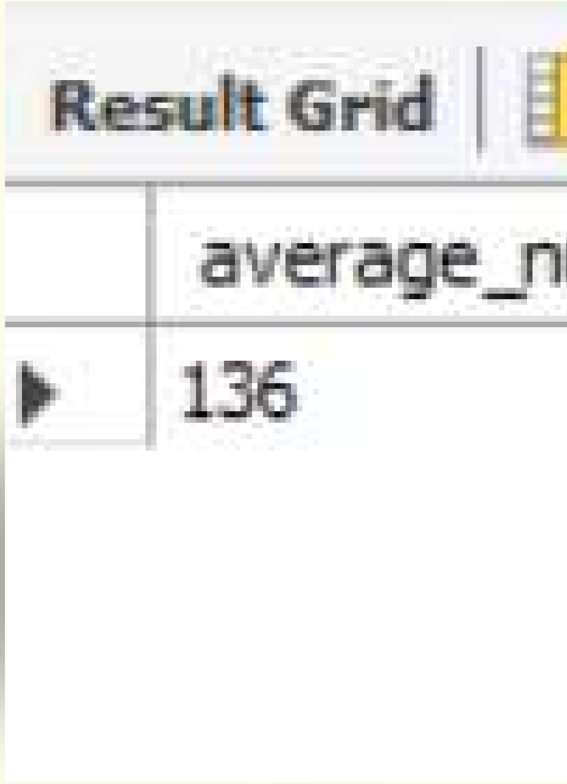


category	COU
Chicken	6
Classic	8
Supreme	9
Veggie	9

9.Group the orders by date and calc average number of pizzas ordered

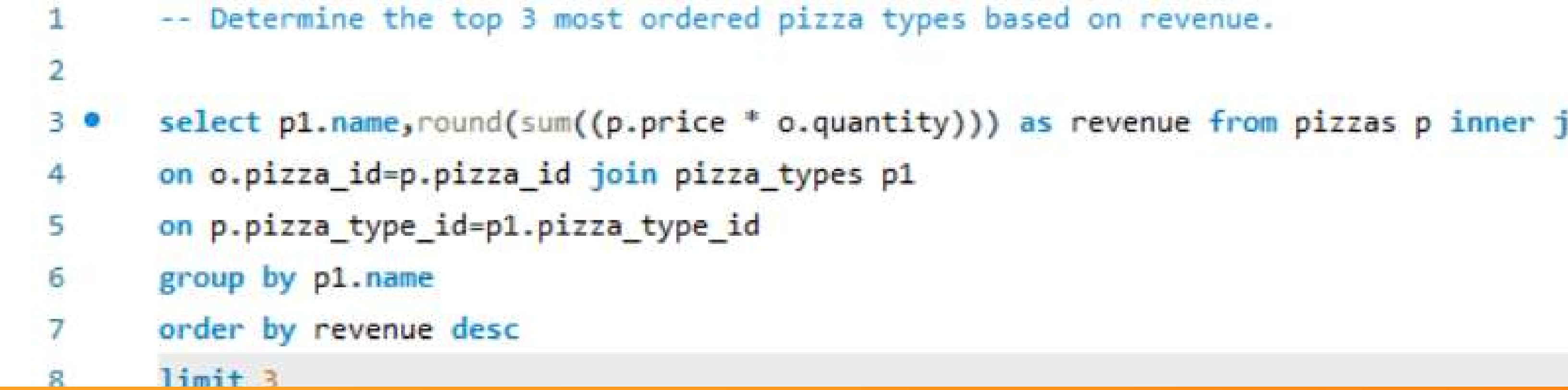


```
-- Group the orders by date and calculate the average number of p
SELECT
    ROUND(AVG(quantity))
FROM
    (SELECT
        o1.date, COUNT(o2.quantity) AS quantity
    FROM
        orders o1
    INNER JOIN order_details o2 ON o1.order_id = o2.order_id
    GROUP BY o1.date) AS order_quantity
```



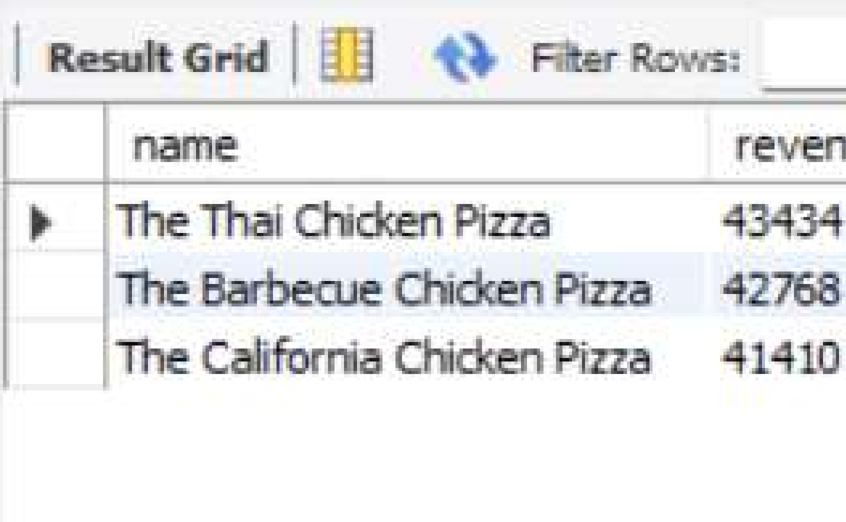
```
average_n
136
```

10.Determine the top 3 most pizza types based on reve



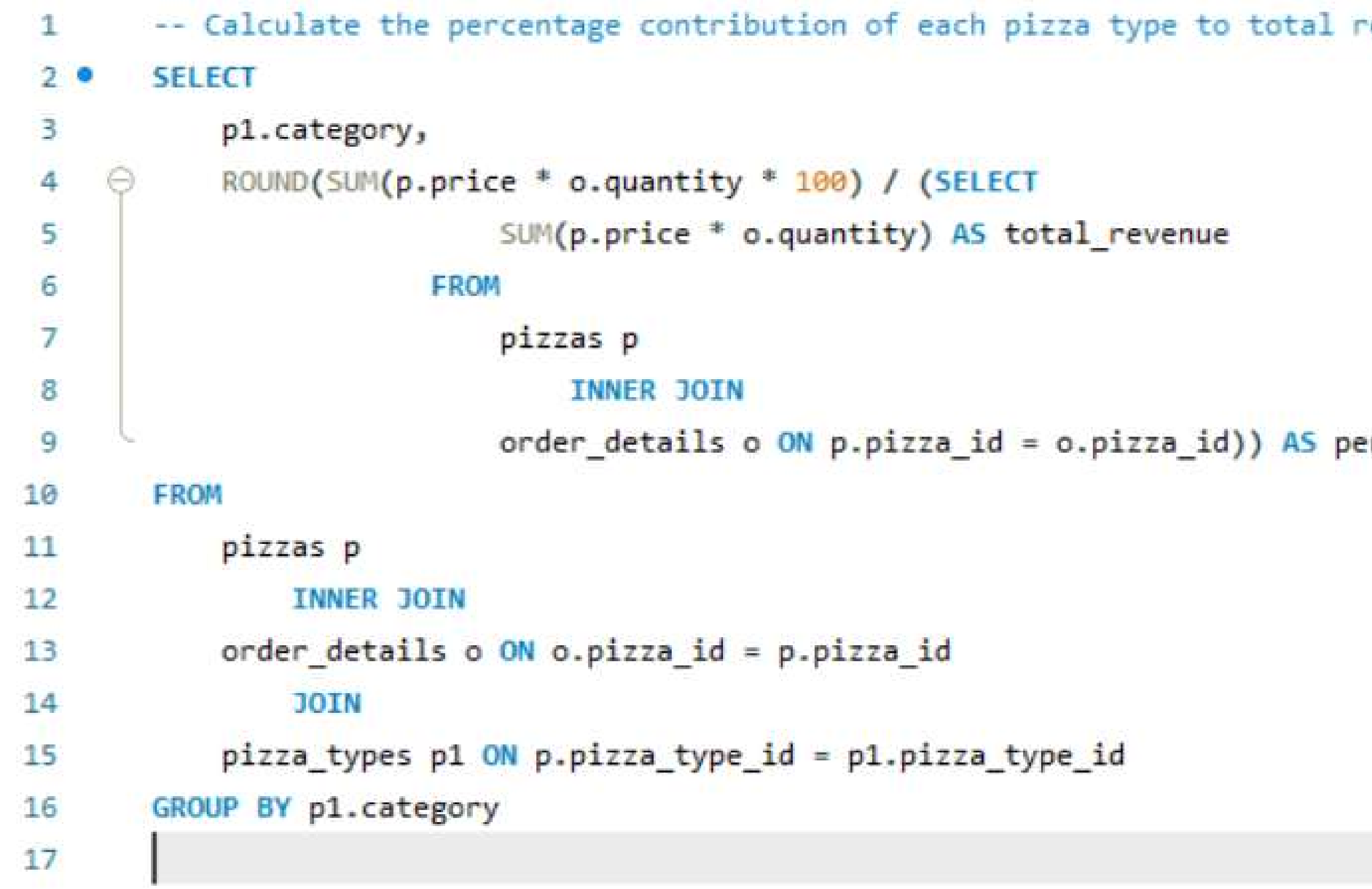
```
-- Determine the top 3 most ordered pizza types based on revenue.

select p1.name,round(sum((p.price * o.quantity))) as revenue from pizzas p inner j
on o.pizza_id=p.pizza_id join pizza_types p1
on p.pizza_type_id=p1.pizza_type_id
group by p1.name
order by revenue desc
limit 3
```

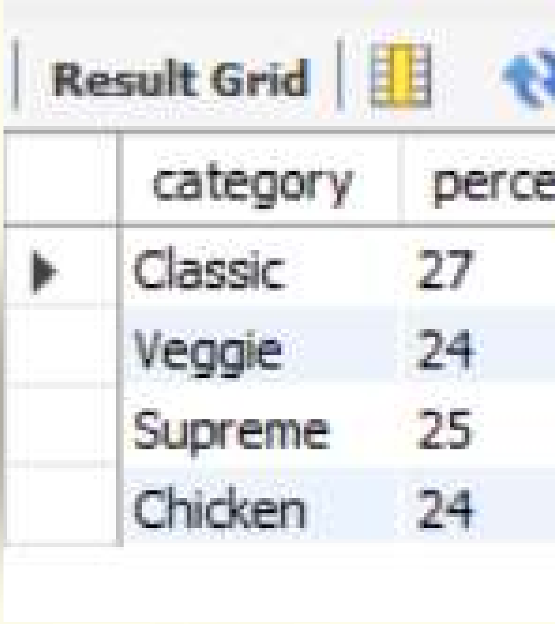


name	reven
The Thai Chicken Pizza	43434
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41410

11.Calculate the percentage contri each pizza type to total reve

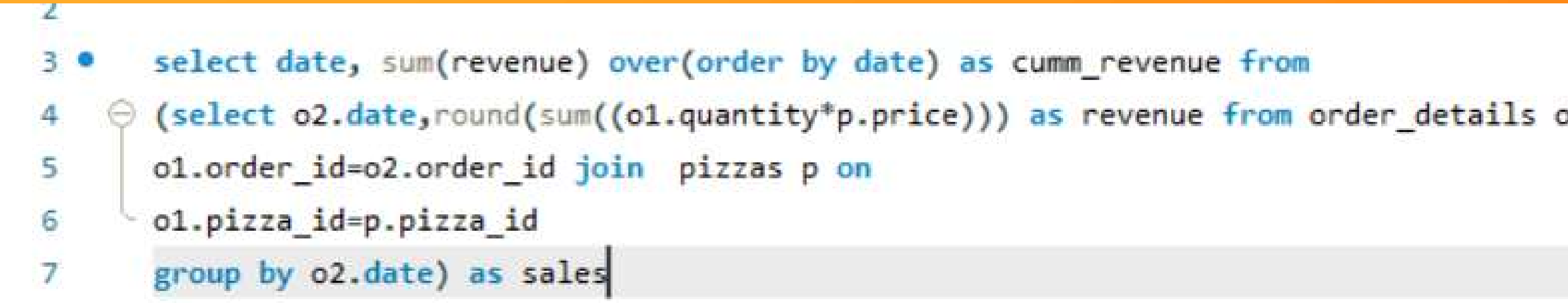


```sql
-- Calculate the percentage contribution of each pizza type to total r
SELECT
    p1.category,
    ROUND(SUM(p.price * o.quantity * 100) / (SELECT
                    SUM(p.price * o.quantity) AS total_revenue
                FROM
                    pizzas p
                        INNER JOIN
                    order_details o ON p.pizza_id = o.pizza_id)) AS pe
FROM
    pizzas p
        INNER JOIN
    order_details o ON o.pizza_id = p.pizza_id
        JOIN
    pizza_types p1 ON p.pizza_type_id = p1.pizza_type_id
GROUP BY p1.category
```

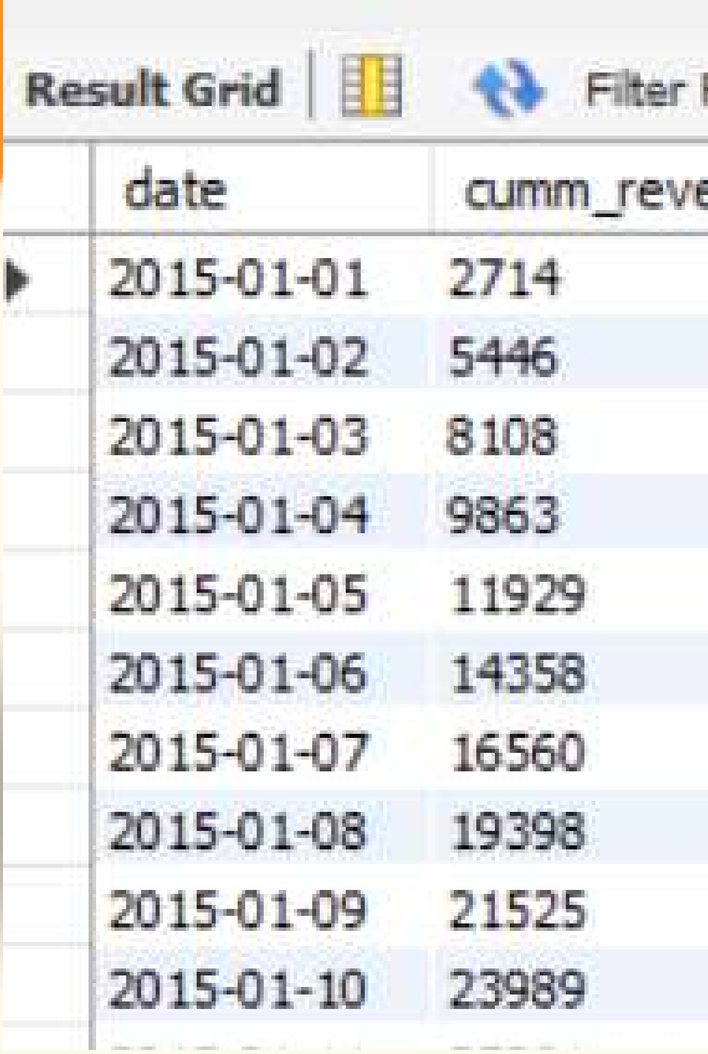


category	perce
Classic	27
Veggie	24
Supreme	25
Chicken	24

12.Analyze the cumulative generated over time

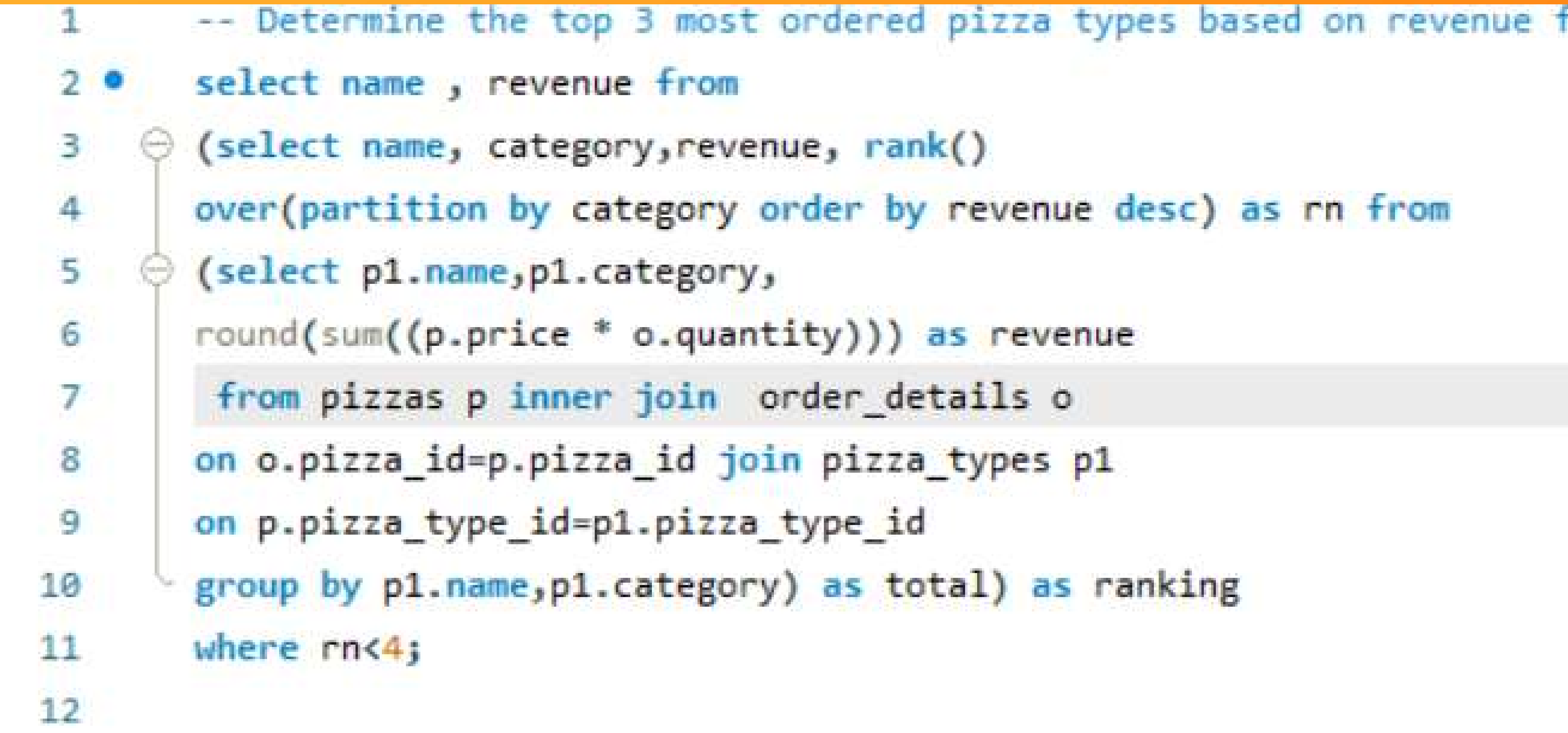


```
select date, sum(revenue) over(order by date) as cumm_revenue from
(select o2.date,round(sum((o1.quantity*p.price))) as revenue from order_details o
o1.order_id=o2.order_id join  pizzas p on
o1.pizza_id=p.pizza_id
group by o2.date) as sales
```

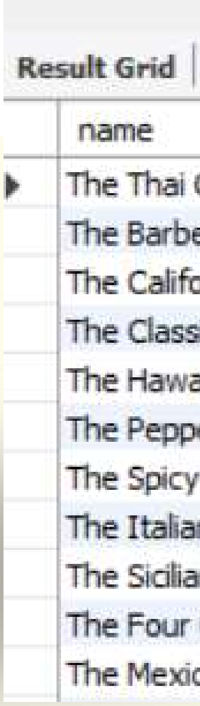


date	cumm_reve
2015-01-01	2714
2015-01-02	5446
2015-01-03	8108
2015-01-04	9863
2015-01-05	11929
2015-01-06	14358
2015-01-07	16560
2015-01-08	19398
2015-01-09	21525
2015-01-10	23989

13.Determine the top 3 most orde types based on revenue for each pizz



```
-- Determine the top 3 most ordered pizza types based on revenue f
select name , revenue from
(select name, category,revenue, rank()
over(partition by category order by revenue desc) as rn from
(select p1.name,p1.category,
round(sum((p.price * o.quantity))) as revenue
 from pizzas p inner join  order_details o
on o.pizza_id=p.pizza_id join pizza_types p1
on p.pizza_type_id=p1.pizza_type_id
group by p1.name,p1.category) as total) as ranking
where rn<4;
```



Result Grid

name
The Thai (
The Barbe
The Califo
The Class
The Hawa
The Pepp
The Spicy
The Italia
The Sicilia
The Four
The Mexic