PIZZA HUT

SQL - PROJECT

Prepared by
Smruti Pragyan Rath

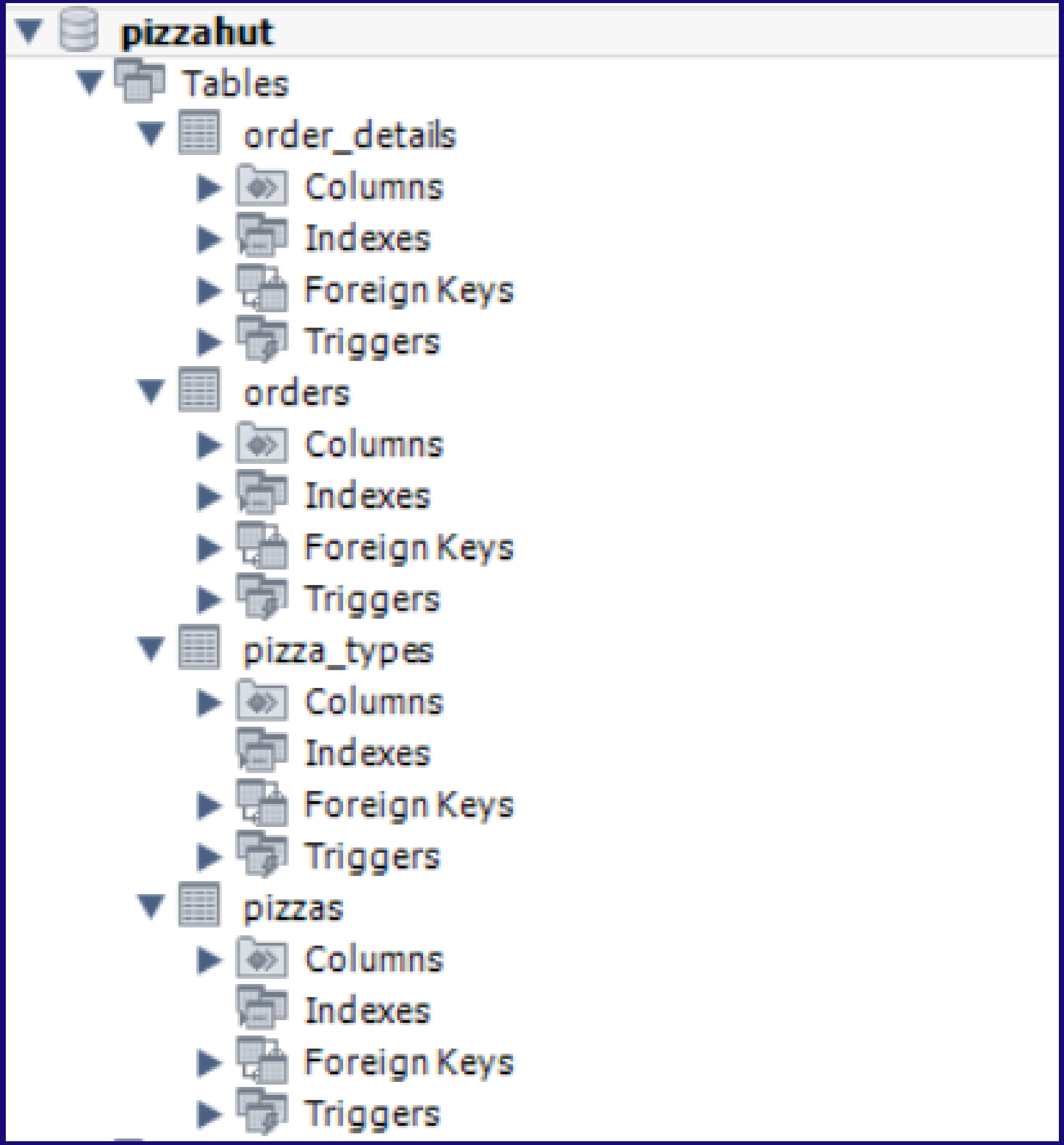
INTRODUCTION

I AM SMRUTI PRAGYAN RATH, I HAVE USED-

- pizzahut database schema
- 4 CSV type tables

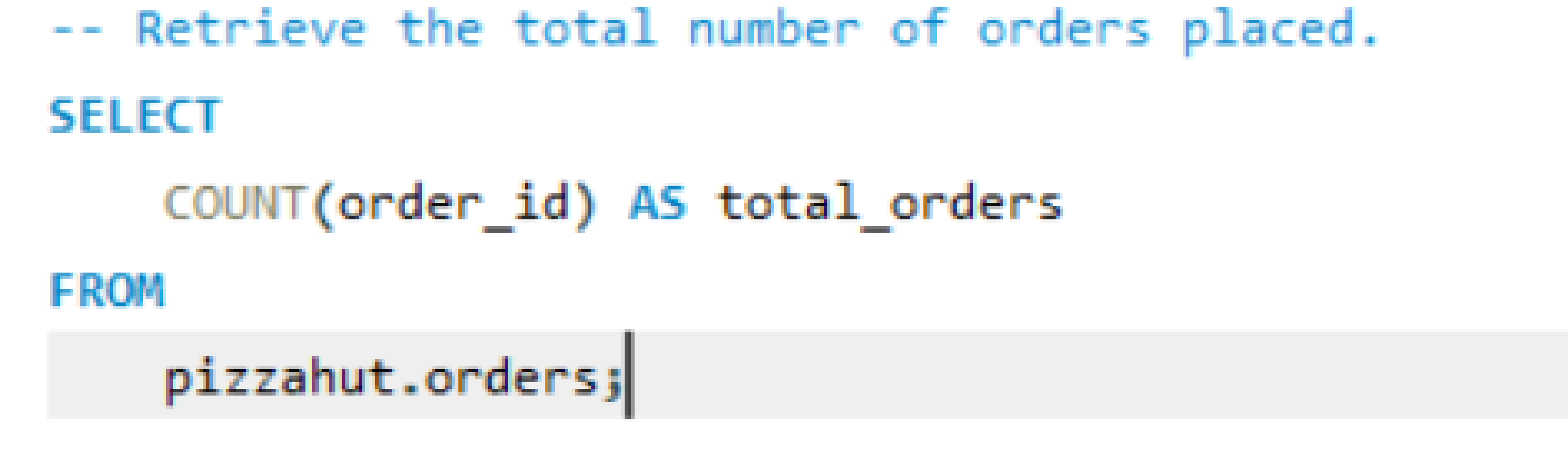
SCHEMA OVERVIEW

Here is how my schema looks on MySQL WorkBench

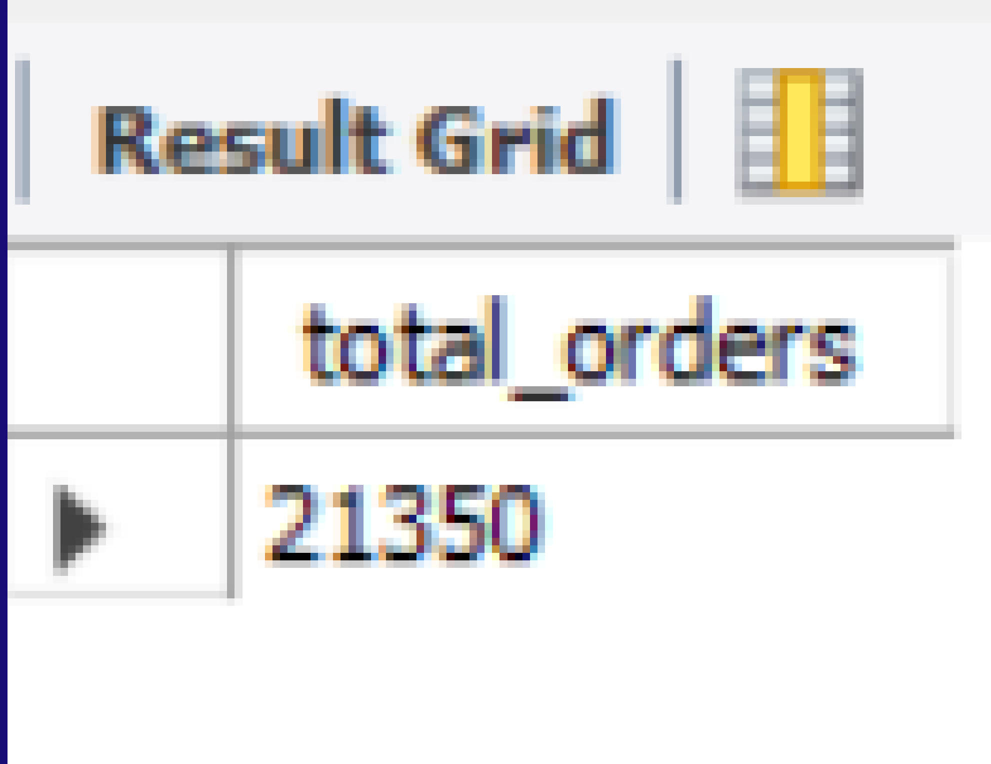


- pizzahut
  - Tables
    - order_details
      - Columns
      - Indexes
      - Foreign Keys
      - Triggers
    - orders
      - Columns
      - Indexes
      - Foreign Keys
      - Triggers
    - pizza_types
      - Columns
      - Indexes
      - Foreign Keys
      - Triggers
    - pizzas
      - Columns
      - Indexes
      - Foreign Keys
      - Triggers

RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.

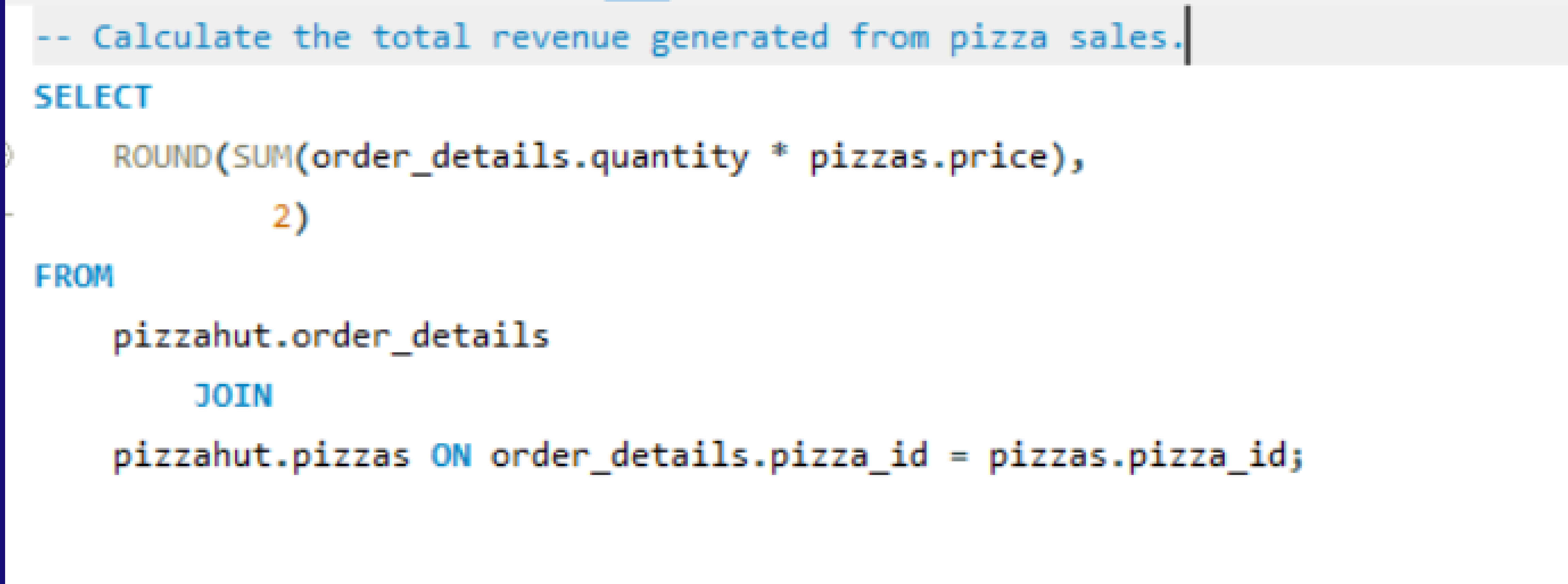


```sql
-- Retrieve the total number of orders placed.
SELECT
    COUNT(order_id) AS total_orders
FROM
    pizzahut.orders;
```

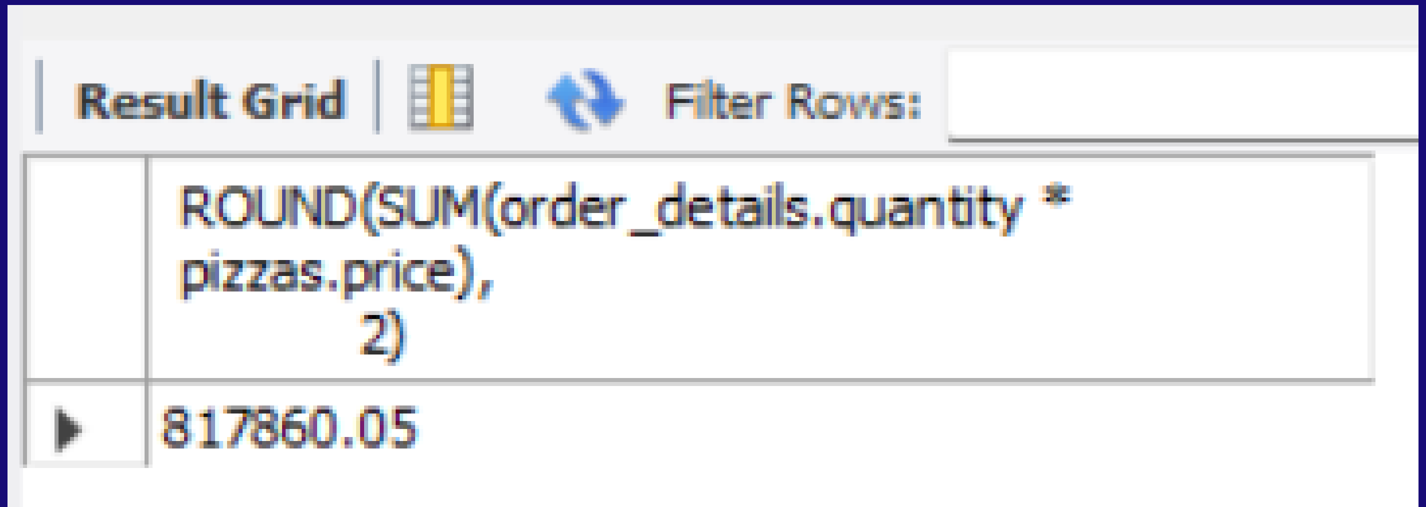


```
total_orders
21350
```

CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

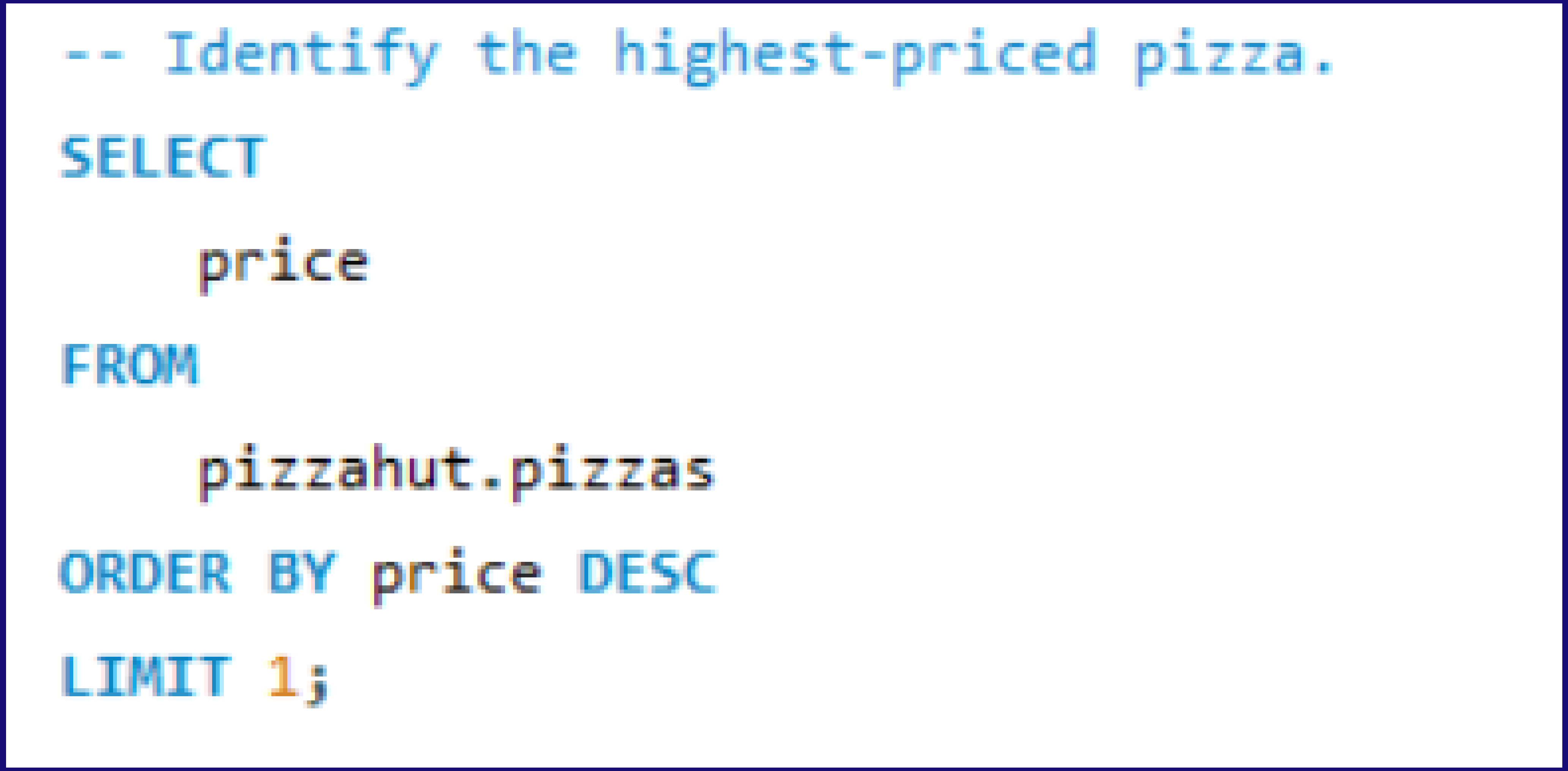


```
-- Calculate the total revenue generated from pizza sales.
SELECT
    ROUND(SUM(order_details.quantity * pizzas.price),
            2)
FROM
    pizzahut.order_details
        JOIN
    pizzahut.pizzas ON order_details.pizza_id = pizzas.pizza_id;
```

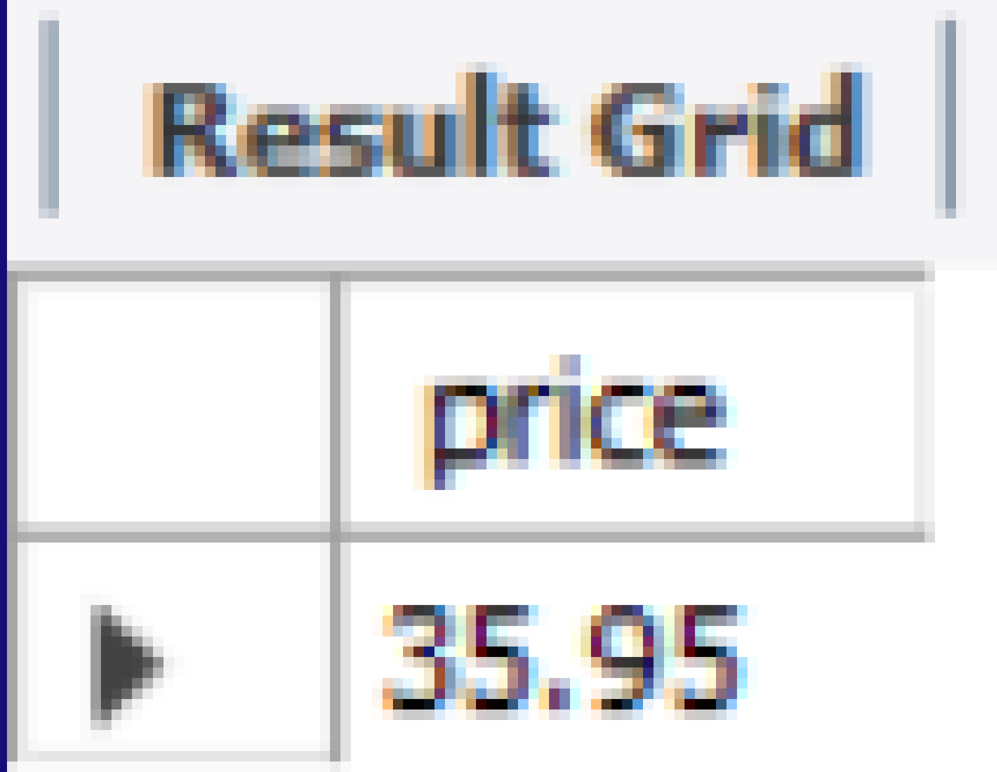


```
ROUND(SUM(order_details.quantity *
pizzas.price),
        2)
817860.05
```

IDENTIFY THE HIGHEST-PRICED PIZZA.



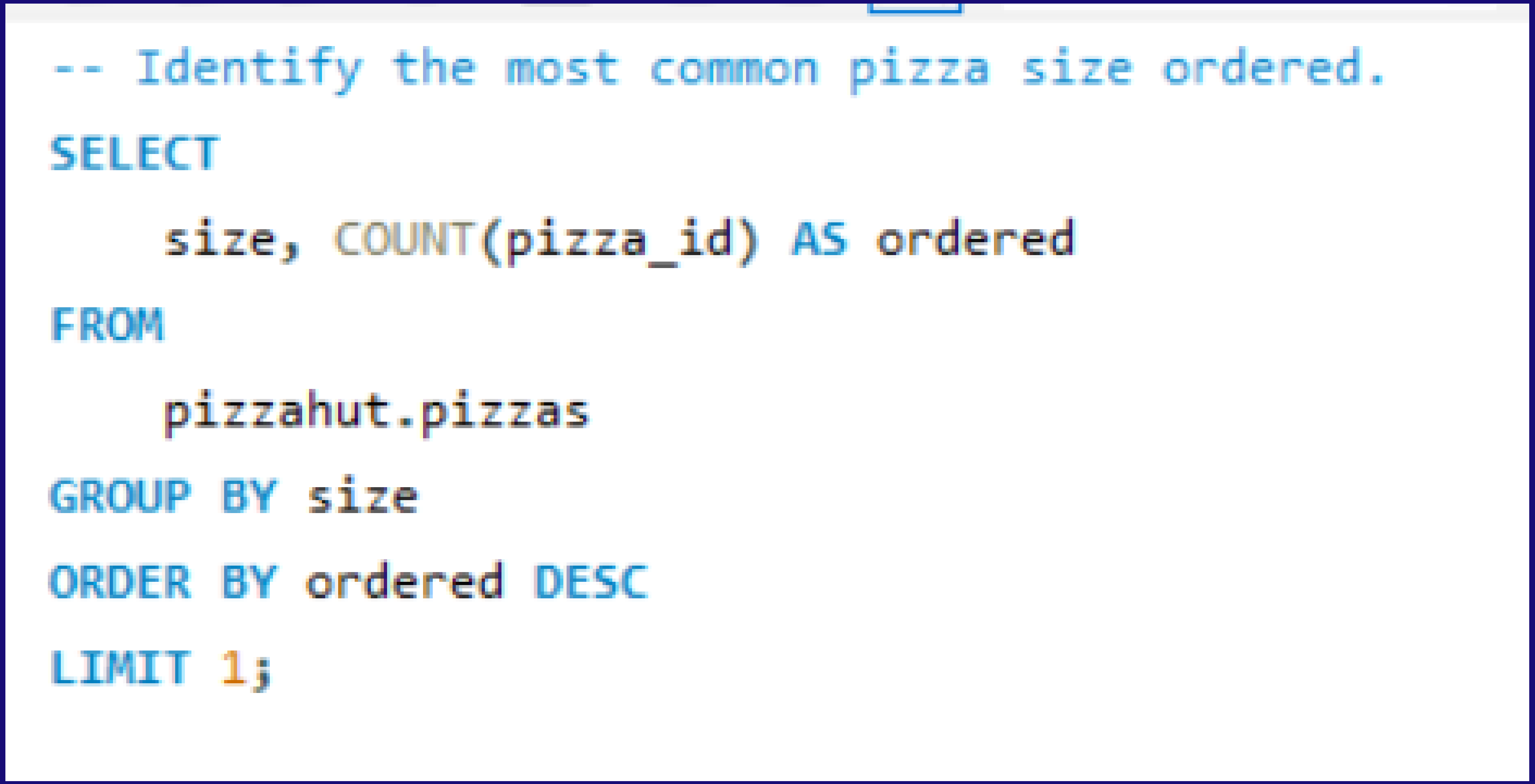
```sql
-- Identify the highest-priced pizza.
SELECT
    price
FROM
    pizzahut.pizzas
ORDER BY price DESC
LIMIT 1;
```



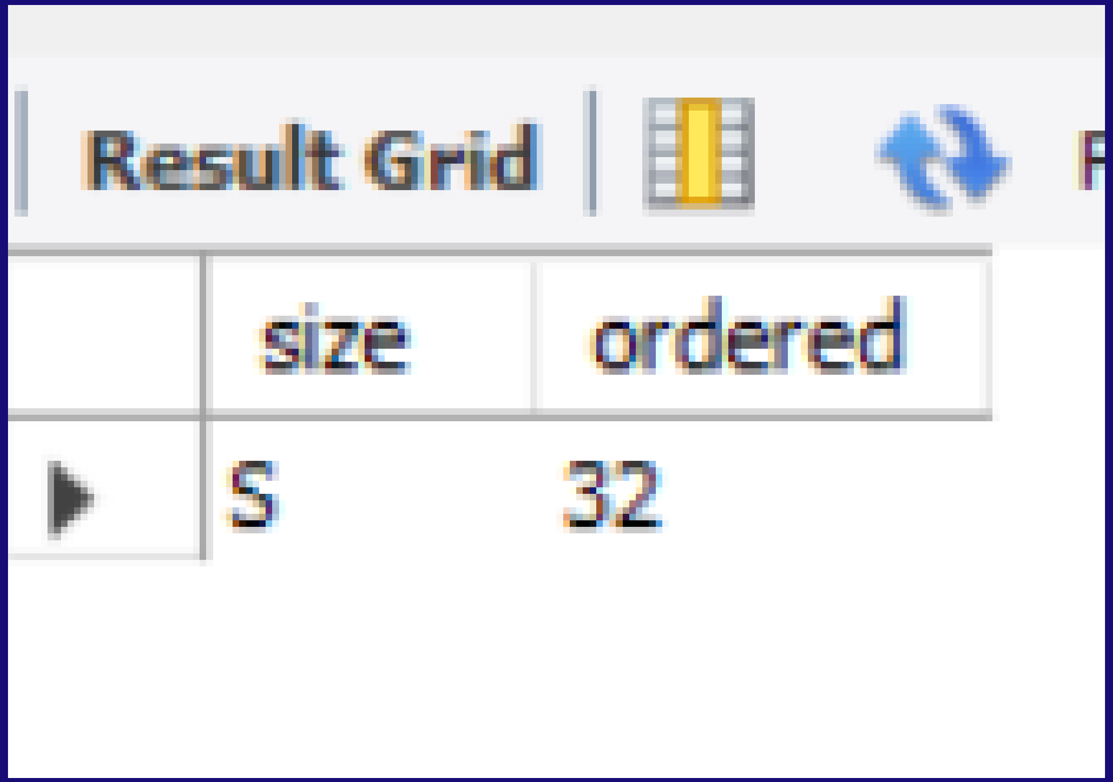
Result Grid

	price
▶	35.95

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.

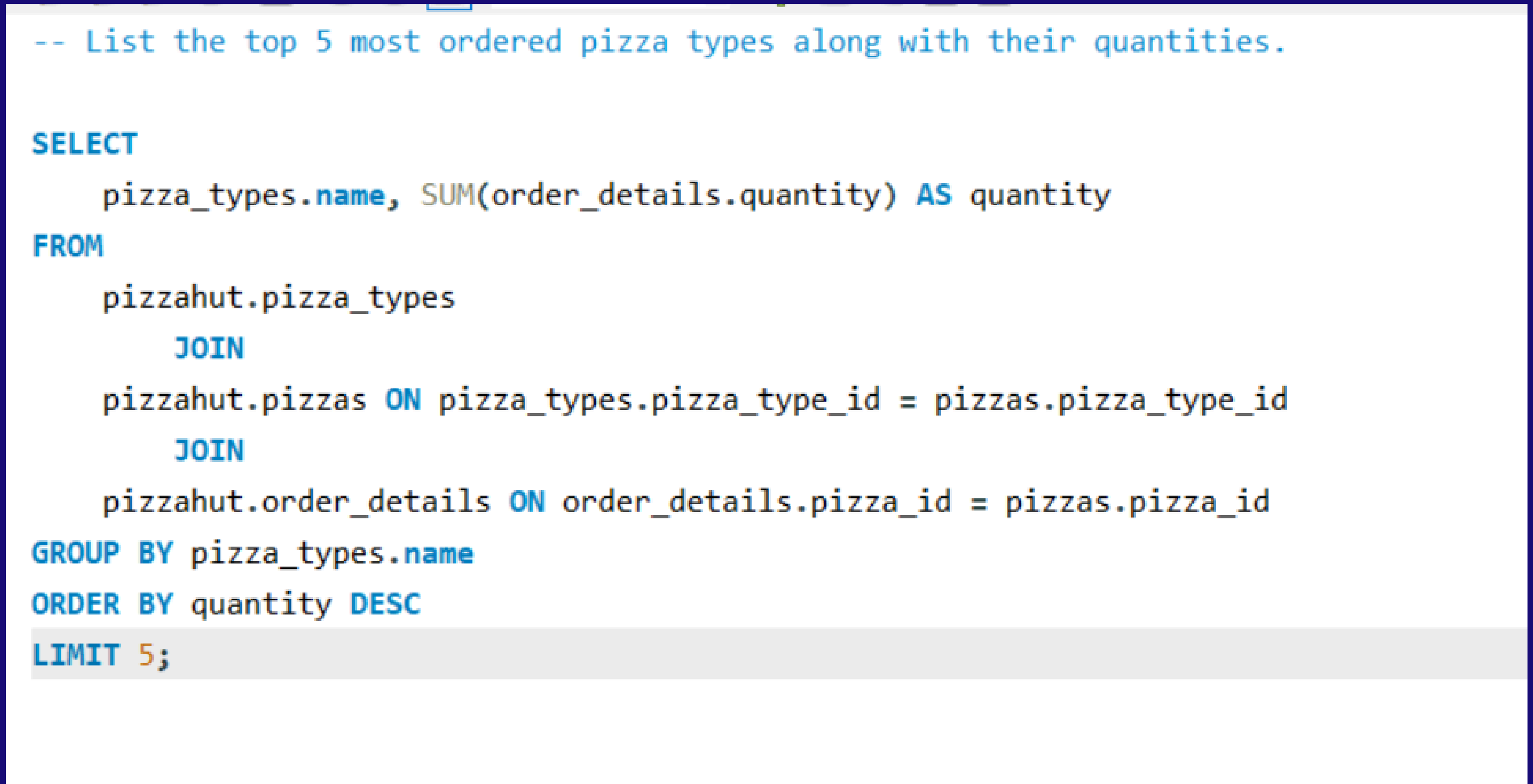


```
-- Identify the most common pizza size ordered.
SELECT
    size, COUNT(pizza_id) AS ordered
FROM
    pizzahut.pizzas
GROUP BY size
ORDER BY ordered DESC
LIMIT 1;
```



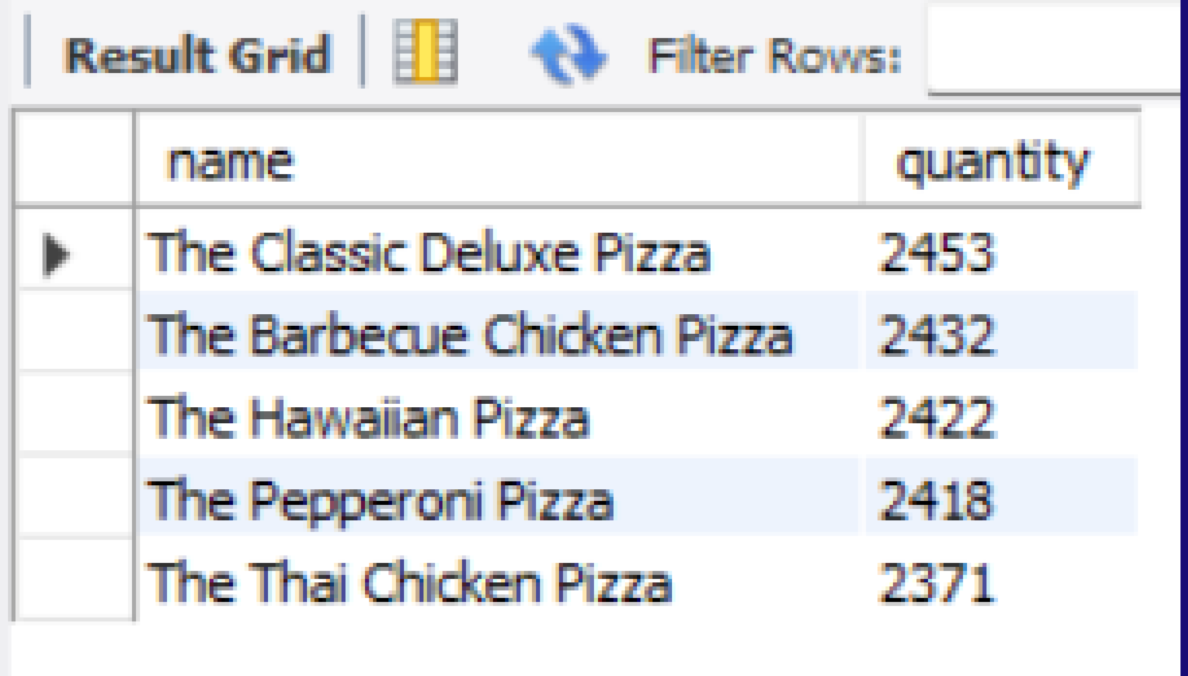
size	ordered
S	32

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



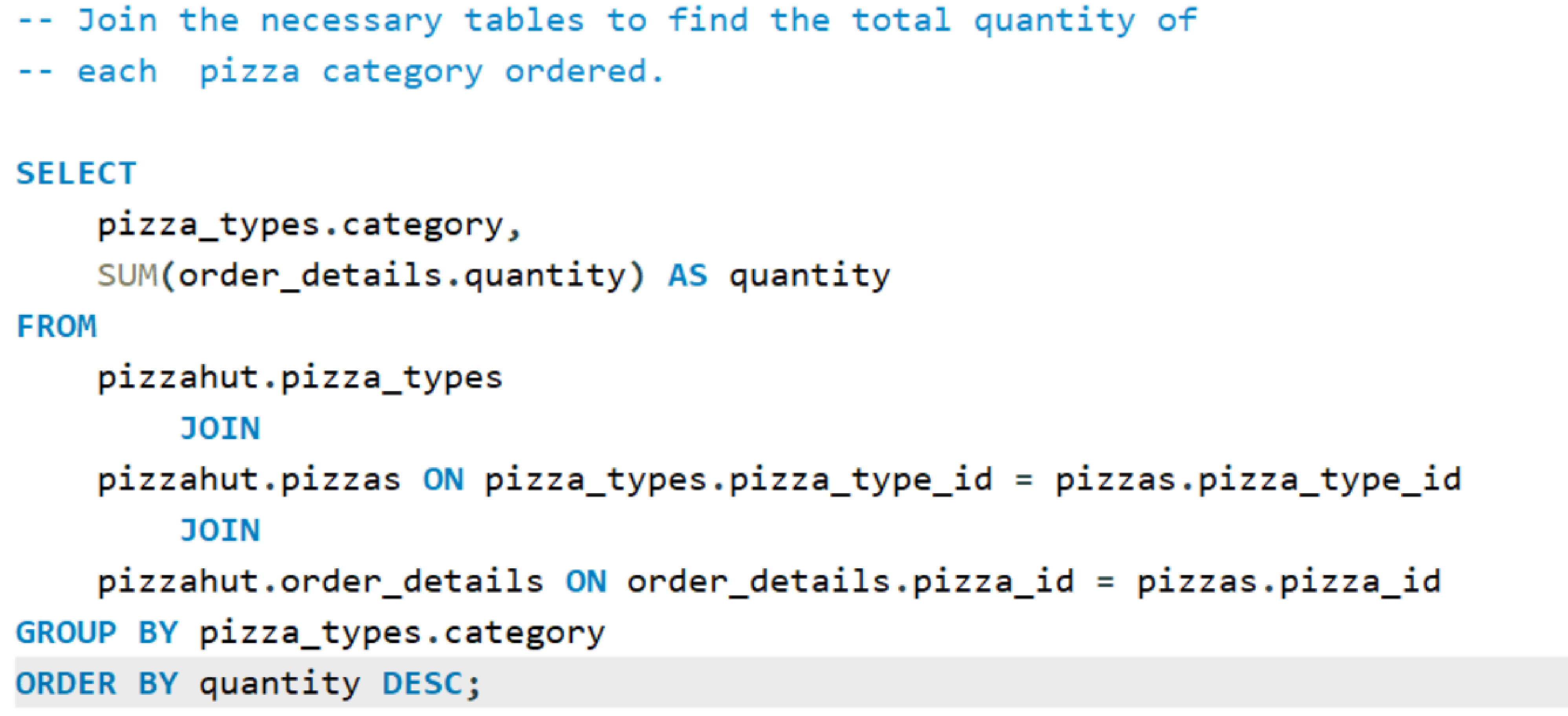
```
-- List the top 5 most ordered pizza types along with their quantities.

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



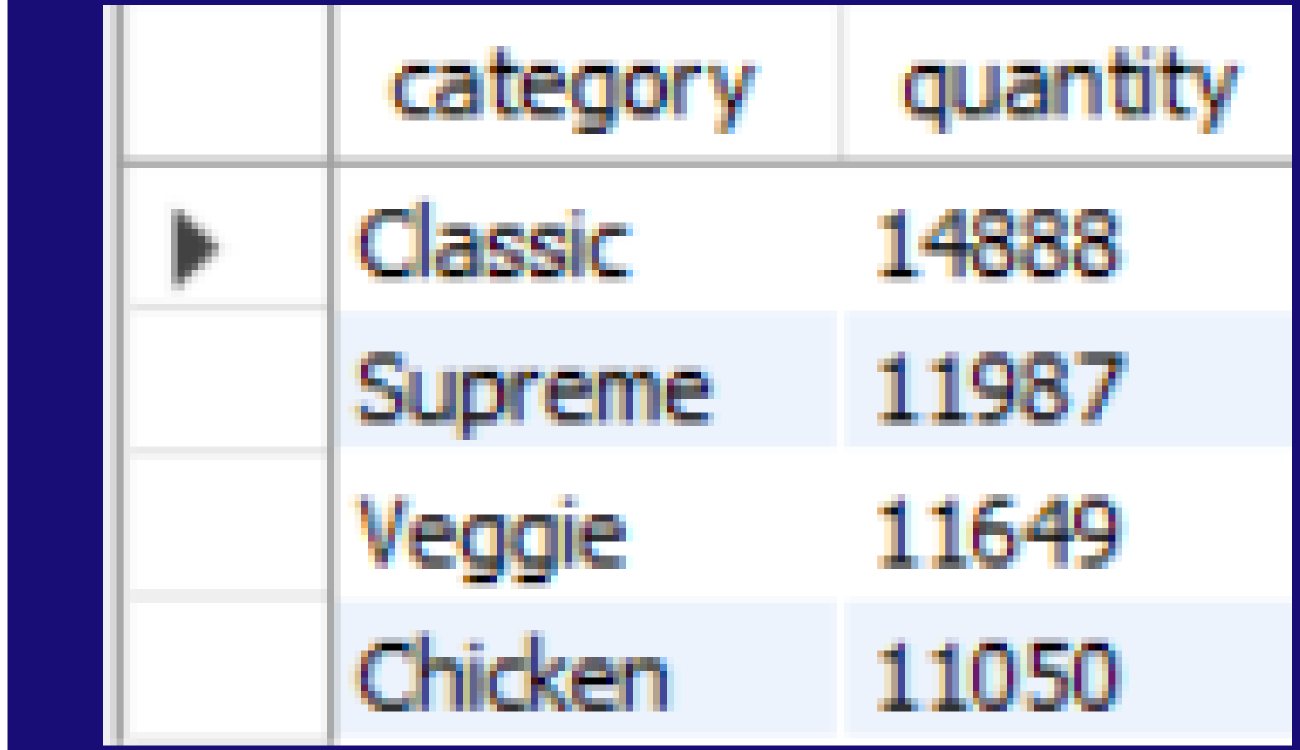
name	quantity
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	2422
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371

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



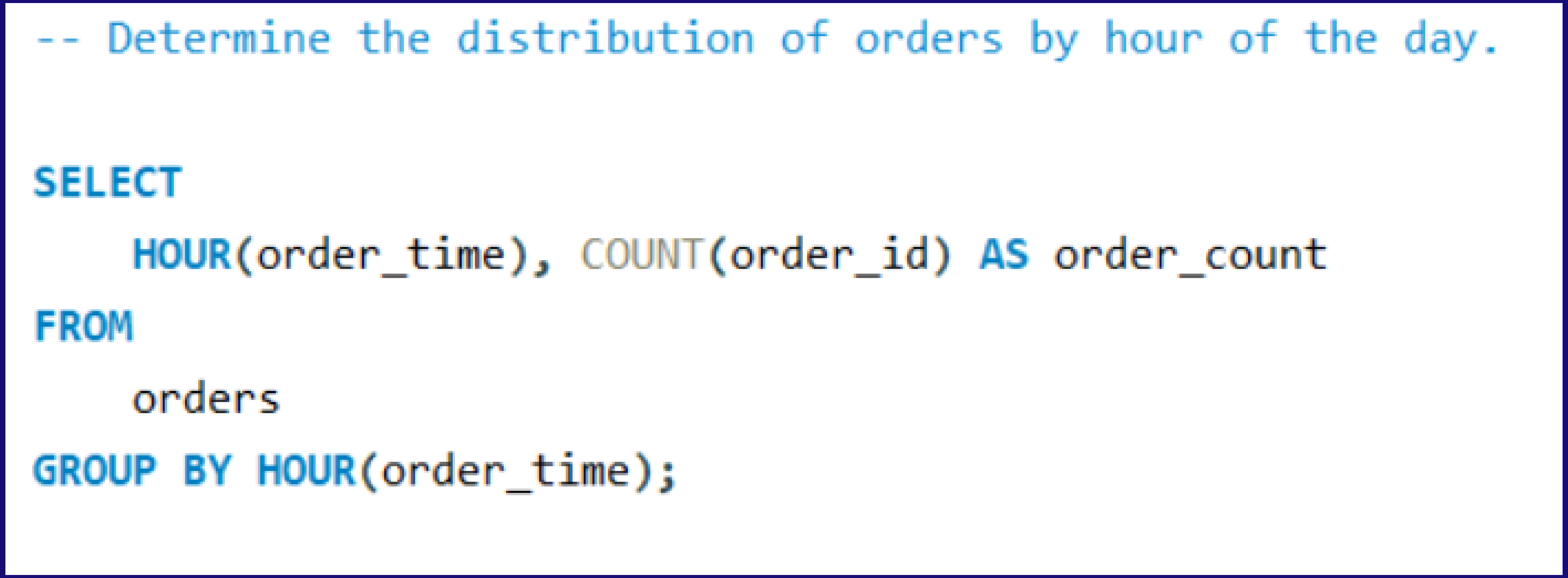
```
-- Join the necessary tables to find the total quantity of
-- each  pizza category ordered.

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



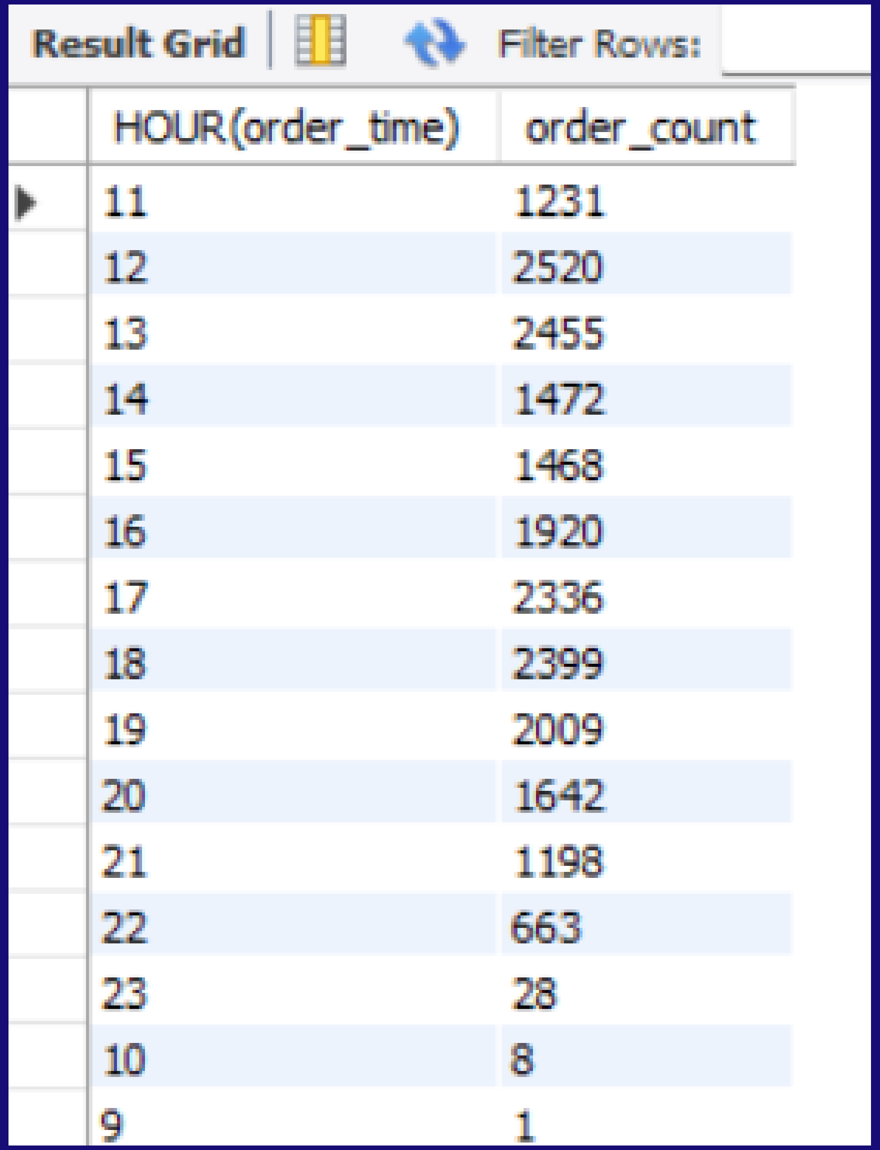
	category	quantity
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.



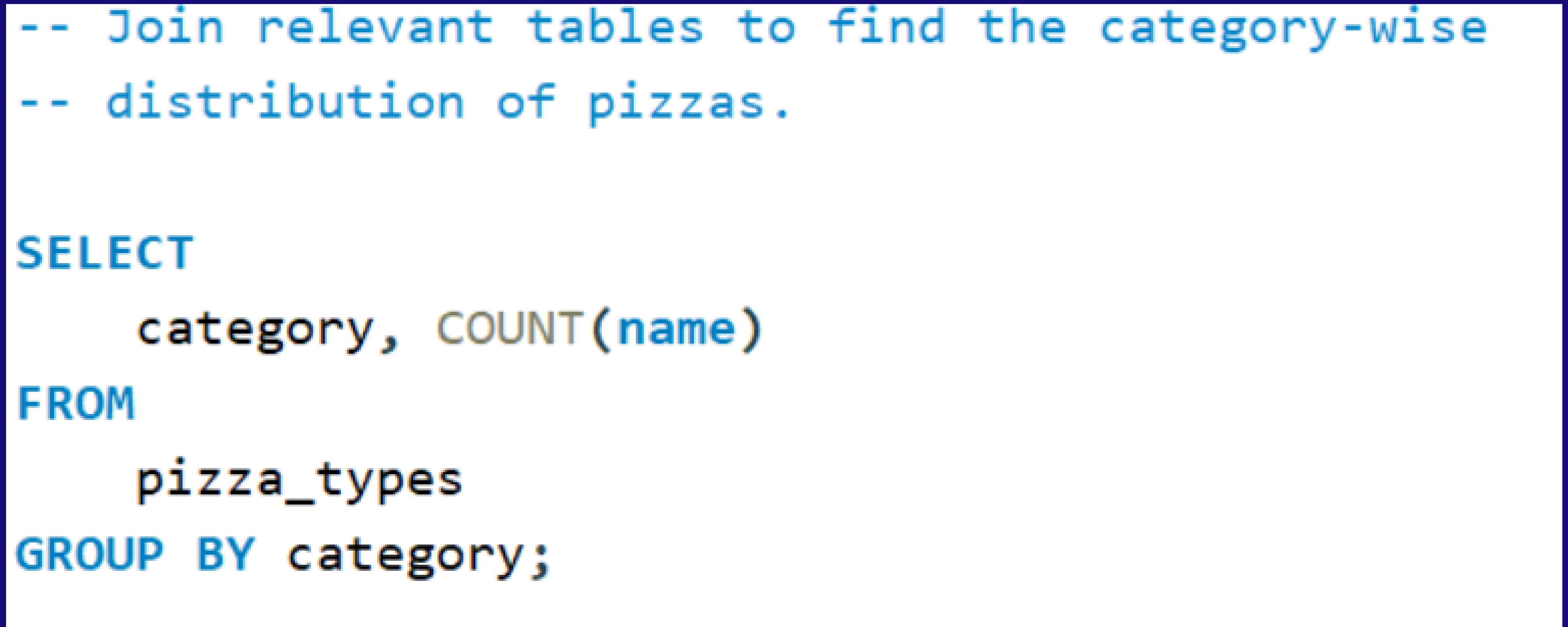
```
-- Determine the distribution of orders by hour of the day.

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



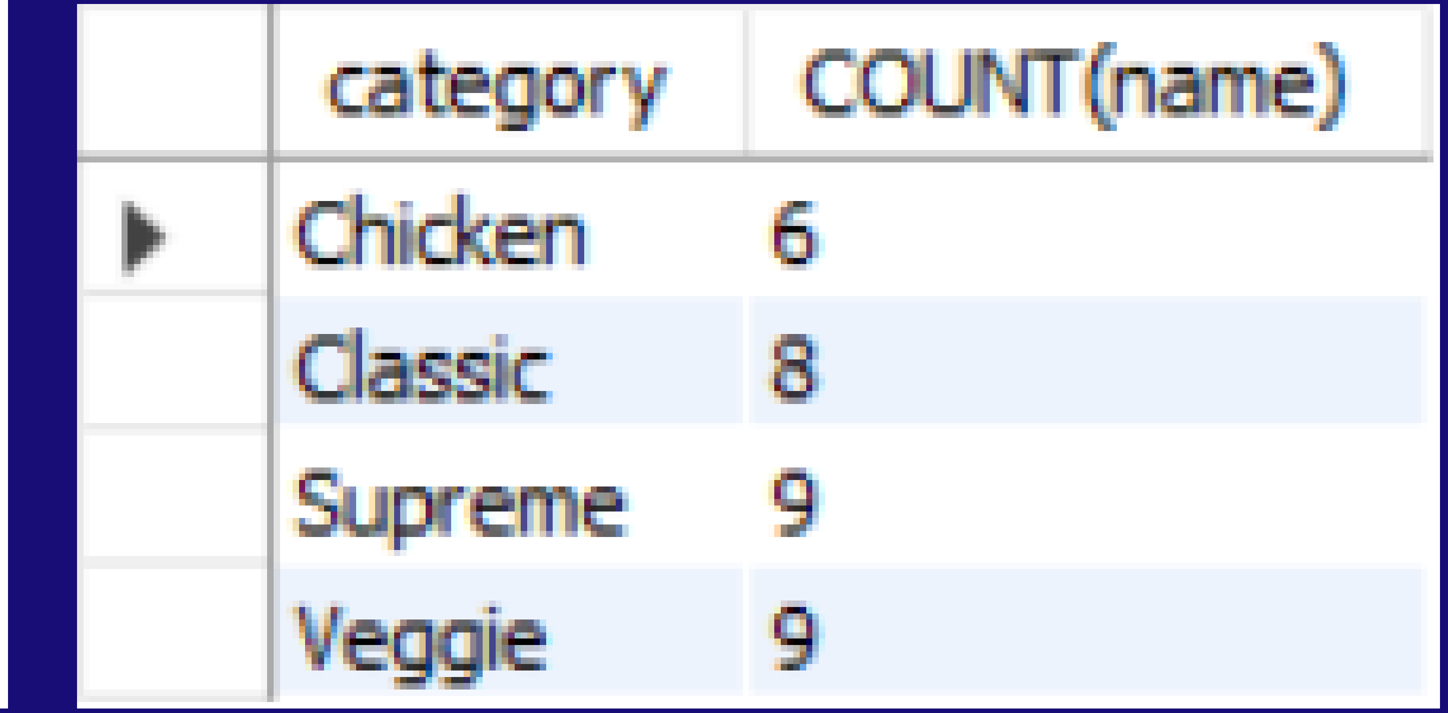
HOUR(order_time)	order_count
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642
21	1198
22	663
23	28
10	8
9	1

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



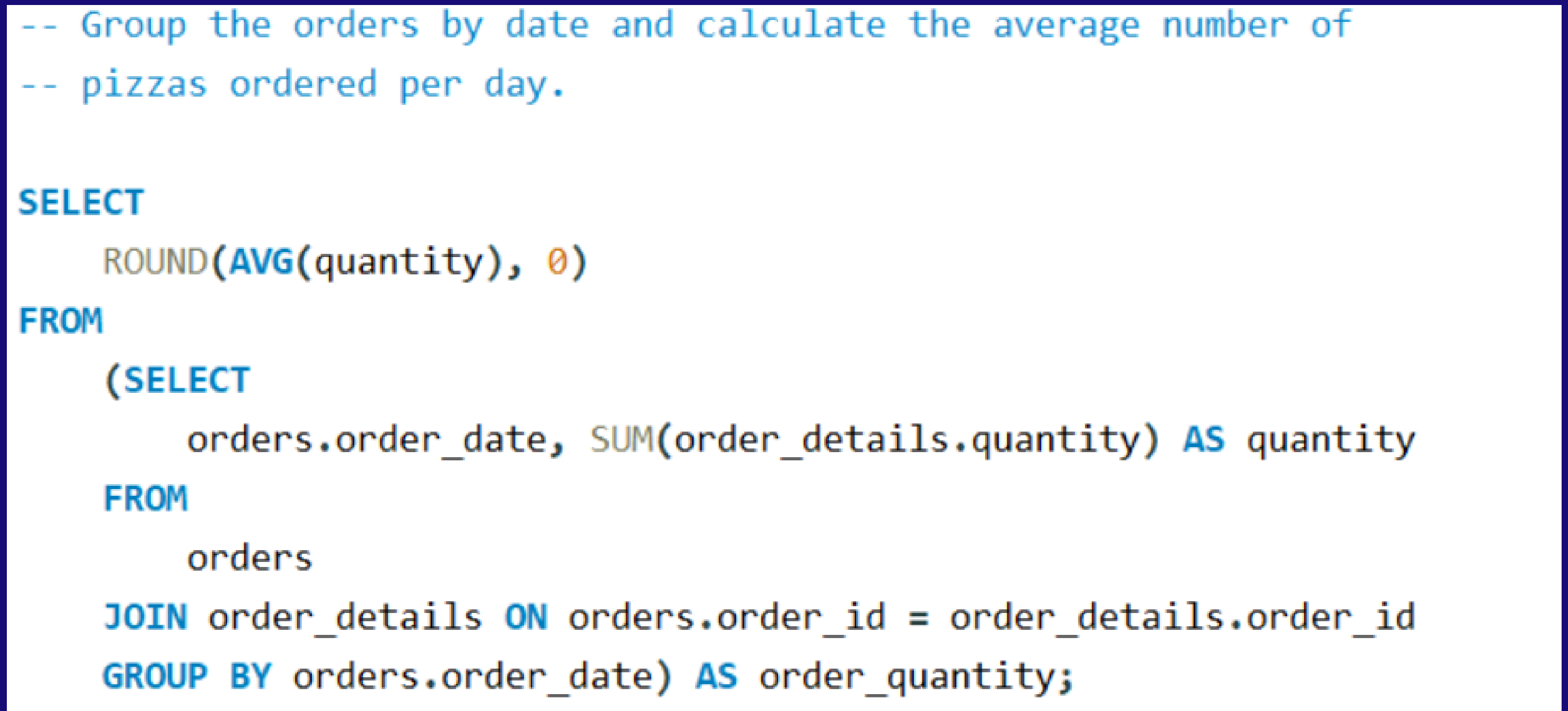
```sql
-- Join relevant tables to find the category-wise
-- distribution of pizzas.

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



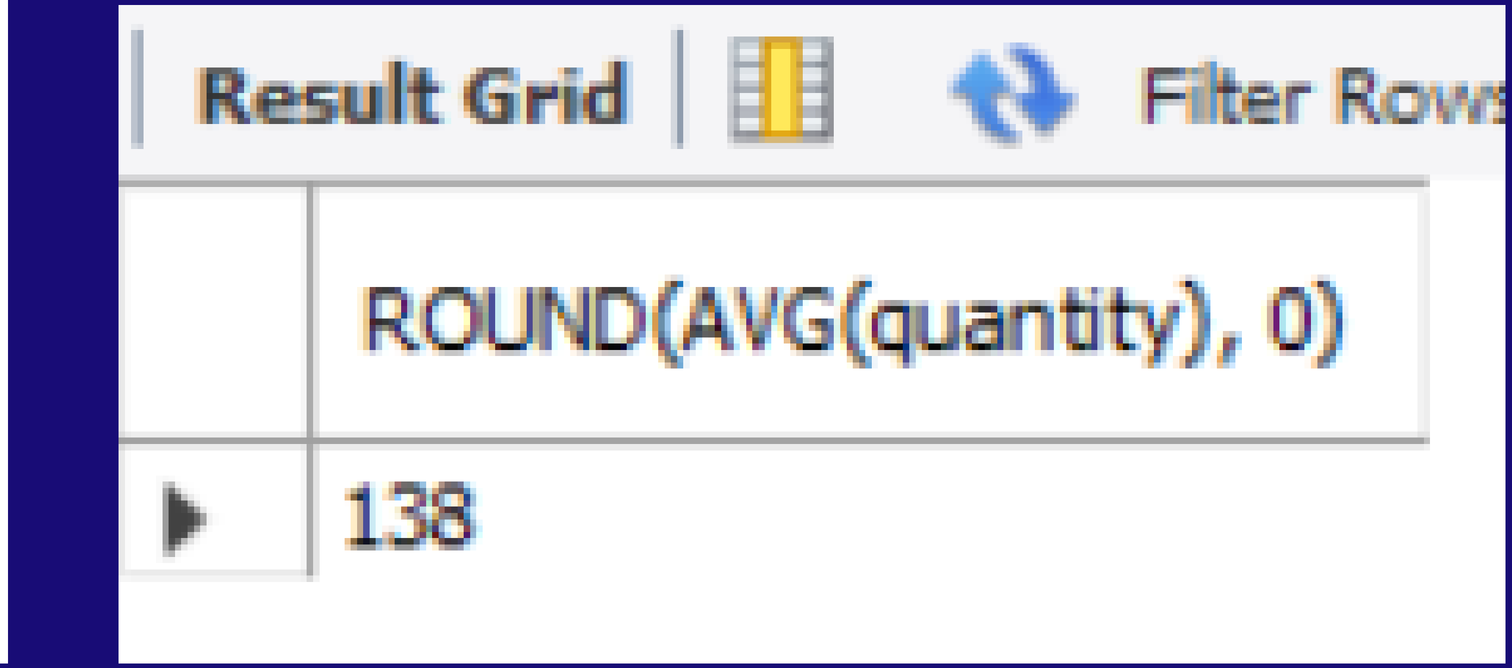
category	COUNT(name)
Chicken	6
Classic	8
Supreme	9
Veggie	9

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



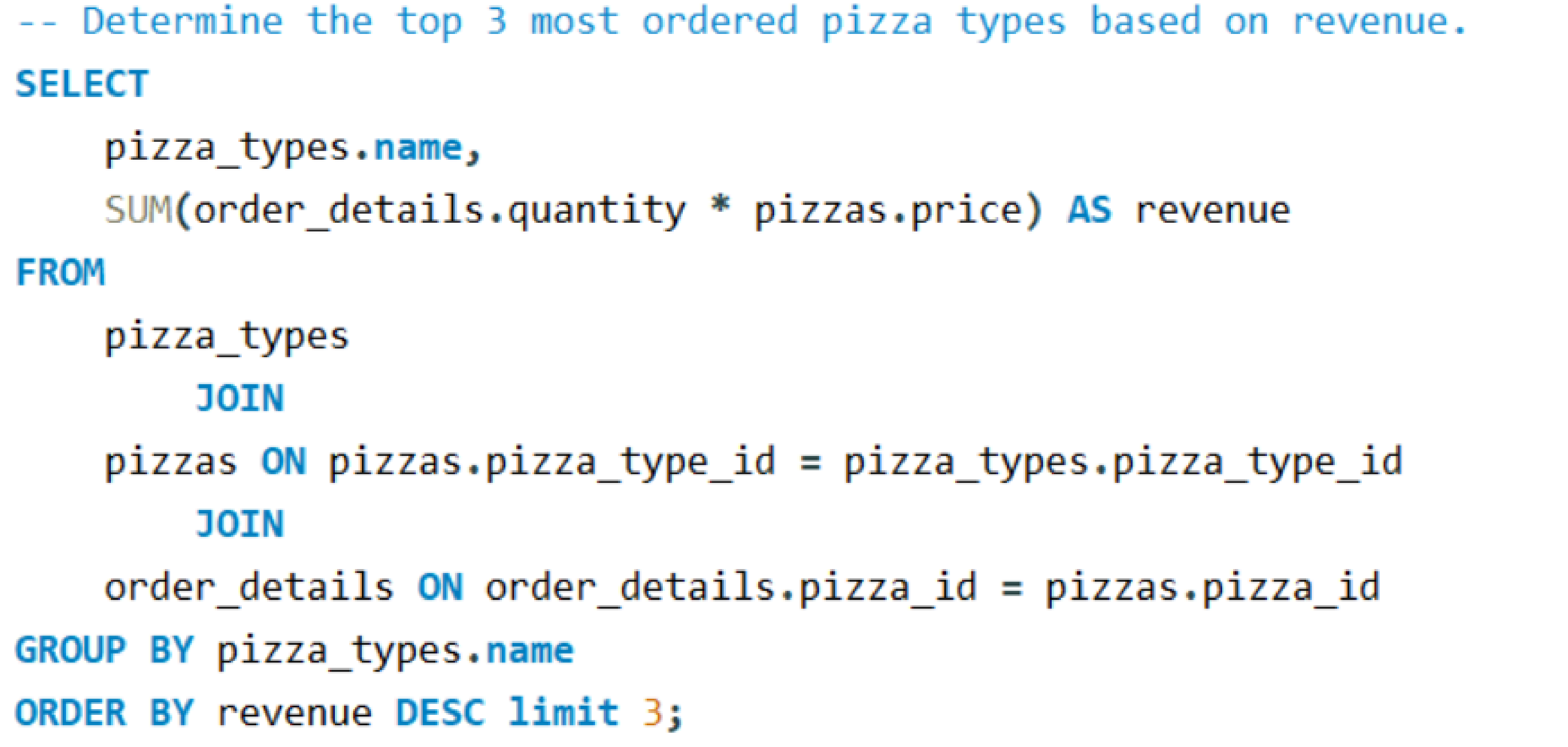
```sql
-- Group the orders by date and calculate the average number of
-- pizzas ordered per day.

SELECT
    ROUND(AVG(quantity), 0)
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;
```

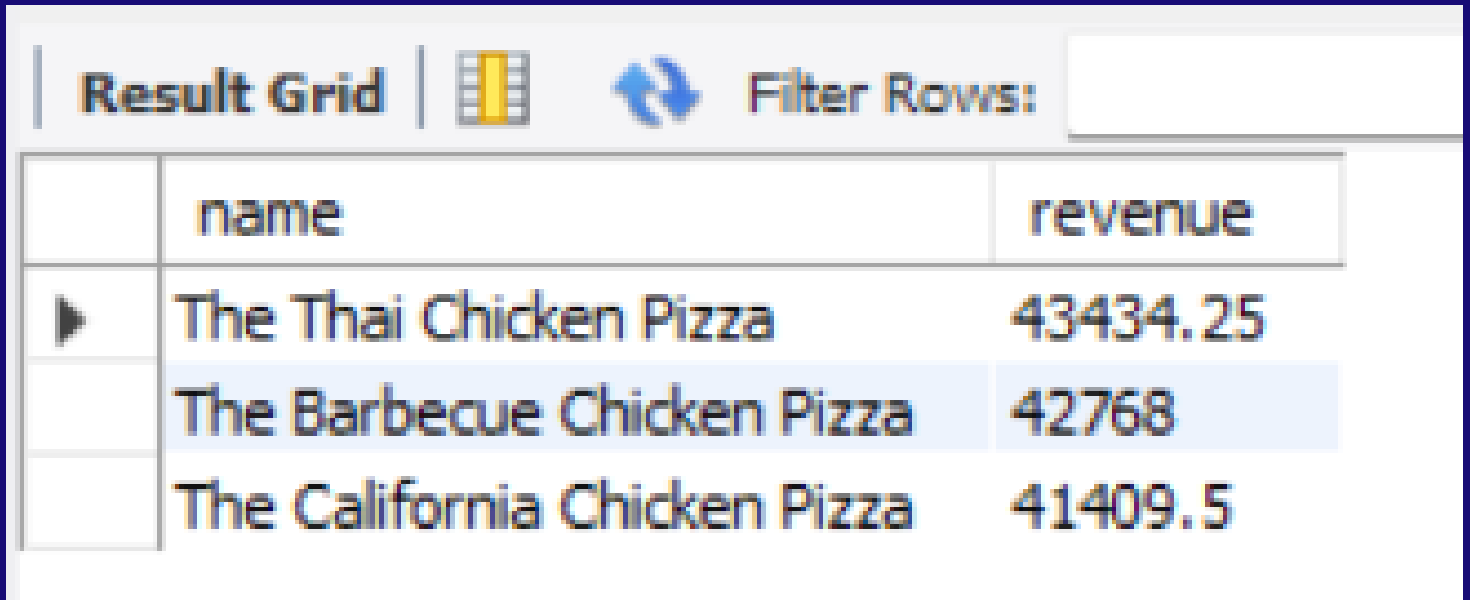


ROUND(AVG(quantity), 0)
138

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



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

THANK

YOU!