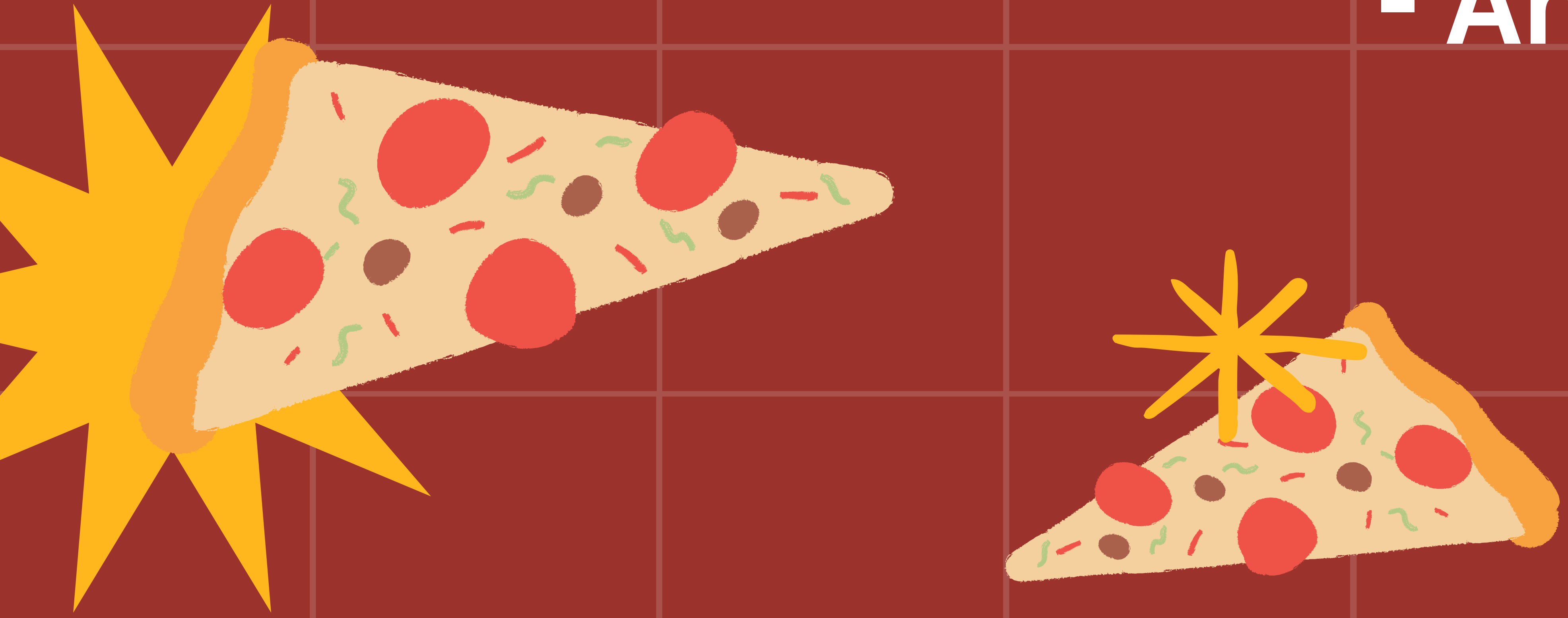
SQL PROJECT ON MARKETING ANALYSIS ON PIZZA SALES

- Ankit Bondre



LIST OF CONTENTS

Introduction

Data Types

Question dataset

Queries

Conclusion

INTRODUCTION

Hello!!! Myself Ankit
I have Conducted a comprehensive marketing analysis on pizza sales data using SQL.

OBJECTIVE:

- **Analyzing customer preferences.**
- **Understanding sales trends.**
- **Evaluating product performance.**

DATA TYPES

Multiple relational database tables used:

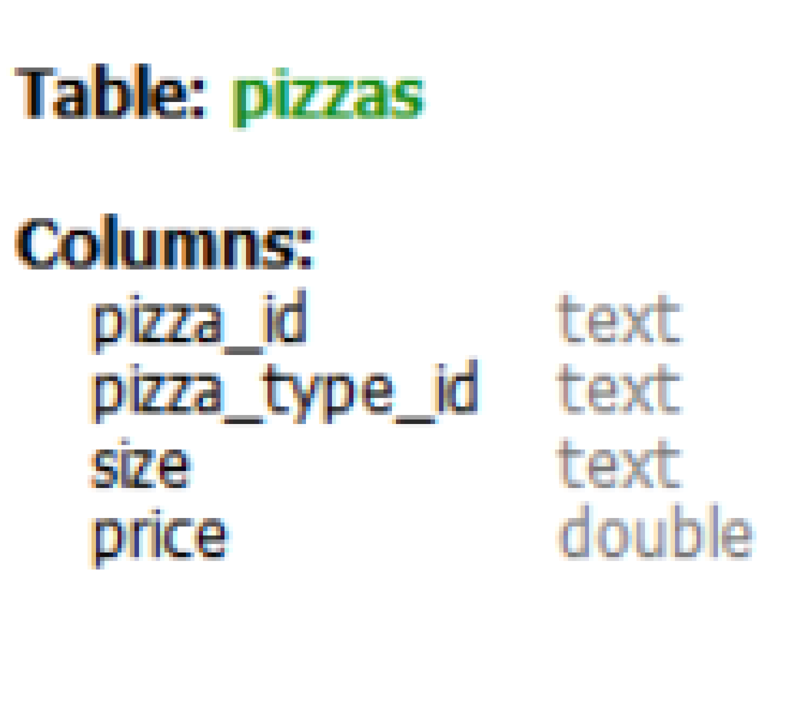


Table: pizzas

Columns:

pizza_id	text
pizza_type_id	text
size	text
price	double

pizzas

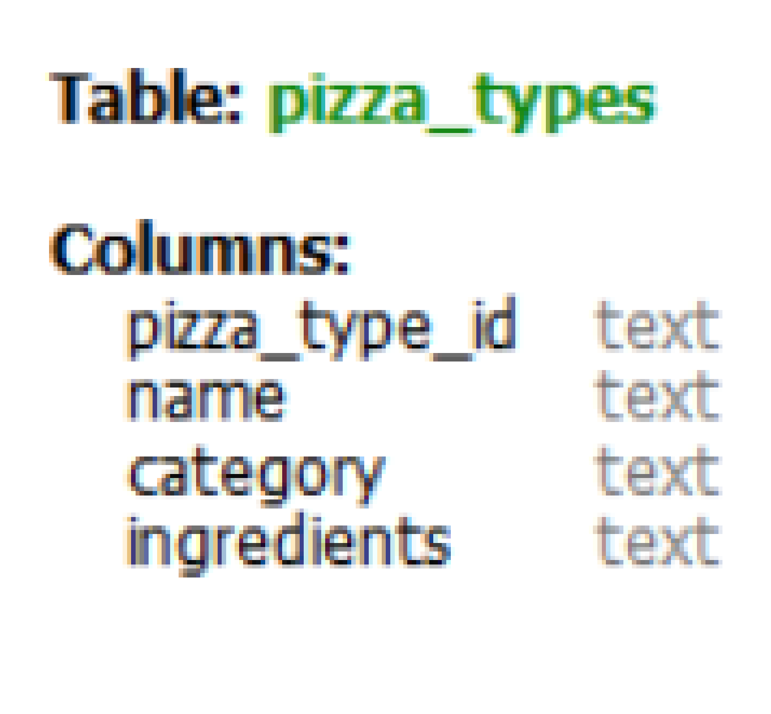


Table: pizza_types

Columns:

pizza_type_id	text
name	text
category	text
ingredients	text

Pizza_types

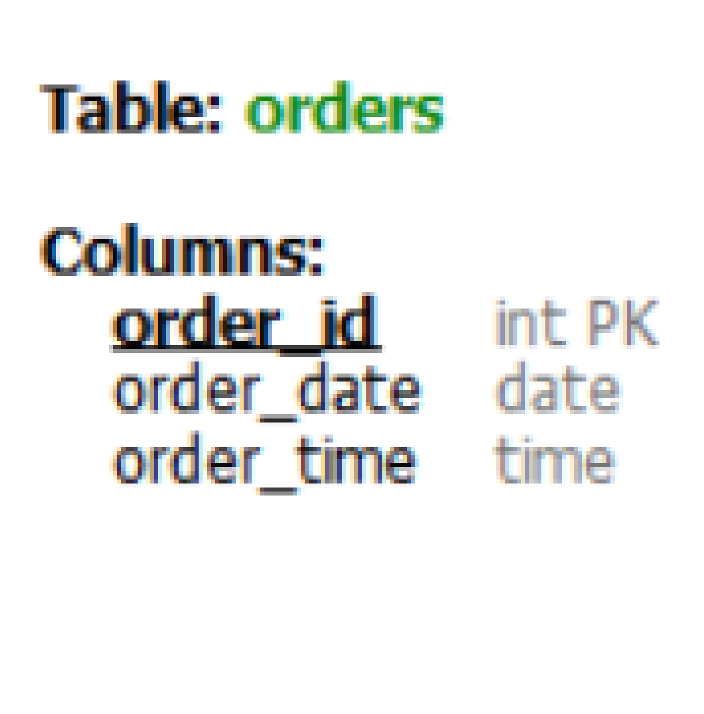


Table: orders

Columns:

order_id	int PK
order_date	date
order_time	time

orders

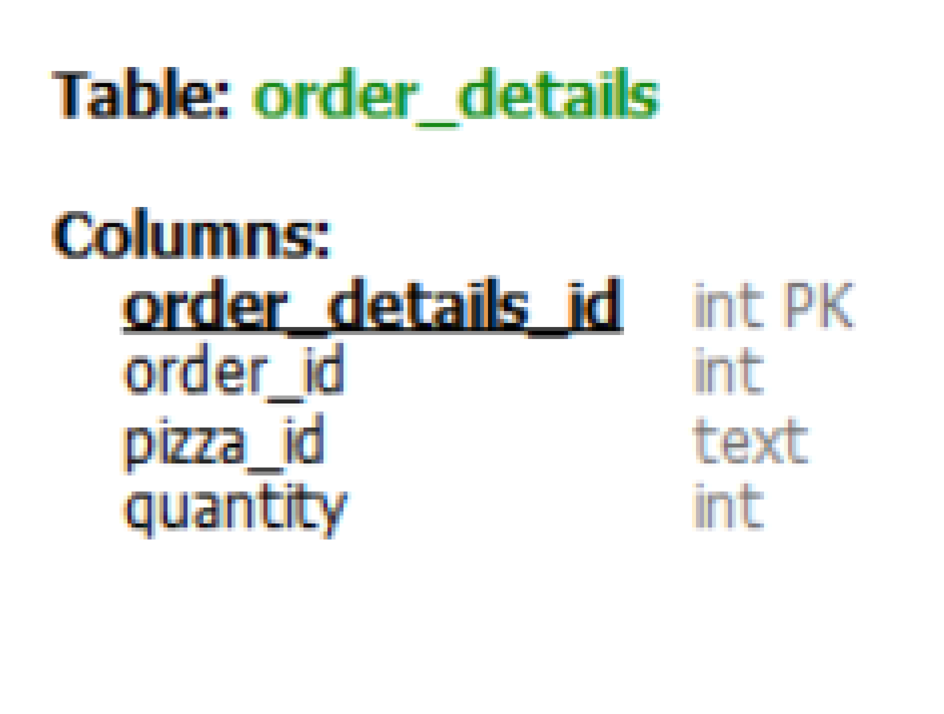


Table: order_details

Columns:

order_details_id	int PK
order_id	int
pizza_id	text
quantity	int

order_details

QUESTIONS DATASET

Basic:

- Retrieve the total number of orders placed.
- Calculate the total revenue generated from pizza sales.
- Identify the highest-priced pizza.
- Identify the most common pizza size ordered.
- List the top 5 most ordered pizza types along with their quantities.

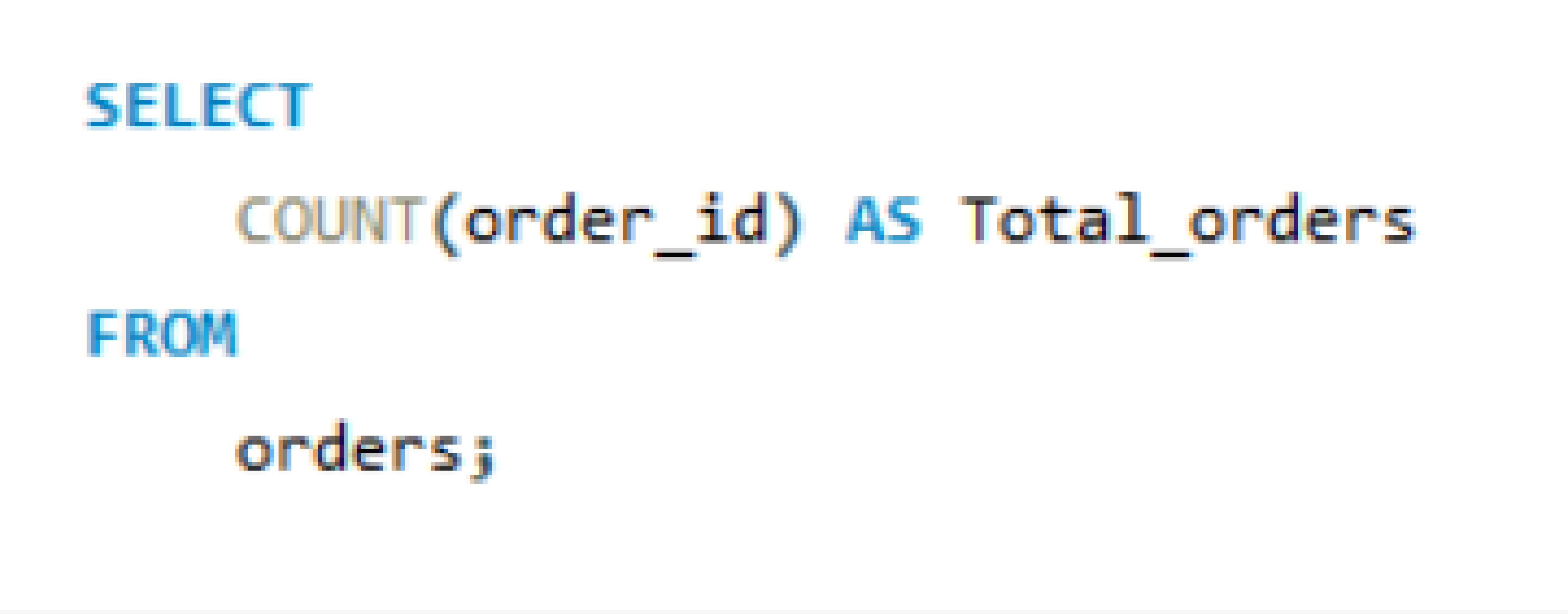
Intermediate:

- Join the necessary tables to find the total quantity of each pizza category ordered.
- Determine the distribution of orders by hour of the day.
- Join relevant tables to find the category-wise distribution of pizzas.
- Group the orders by date and calculate the average number of pizzas ordered per day.
- Determine the top 3 most ordered pizza types based on revenue.

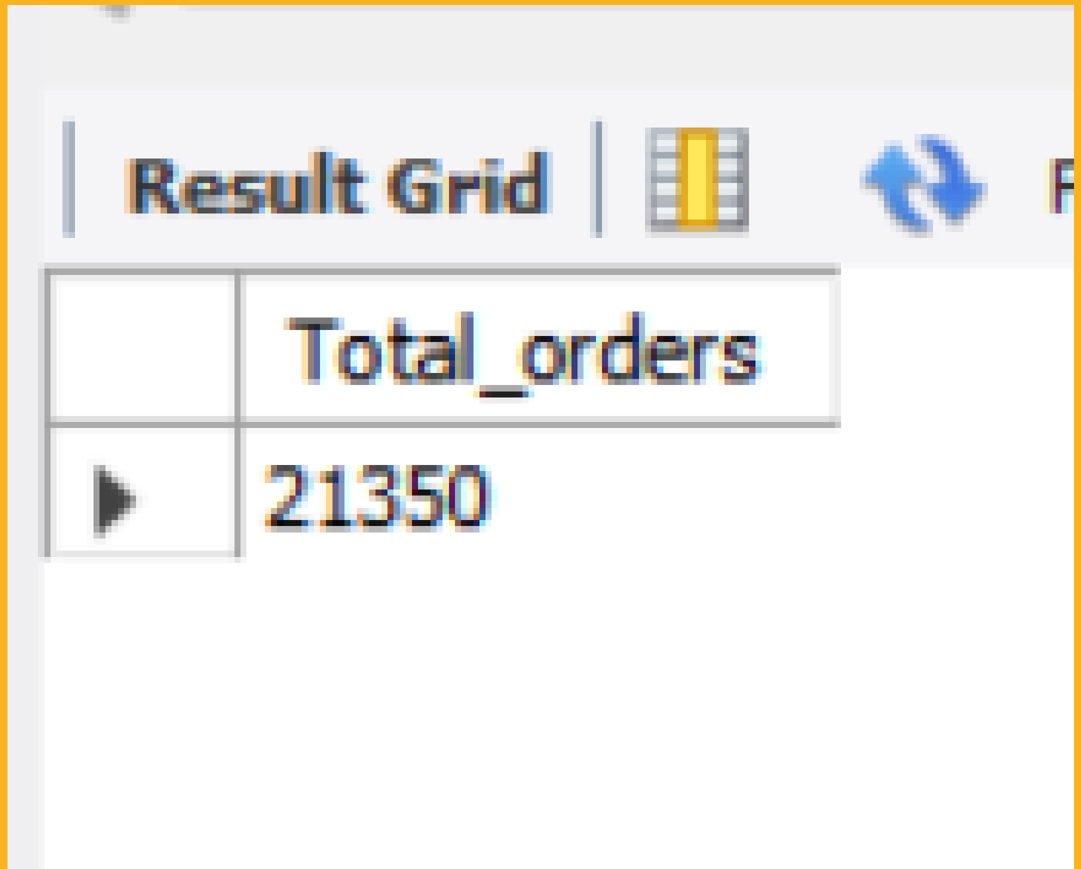
Advanced:

- Calculate the percentage contribution of each pizza type to total revenue.
- Analyze the cumulative revenue generated over time.
- Determine the top 3 most ordered pizza types based on revenue for each pizza category.

RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED

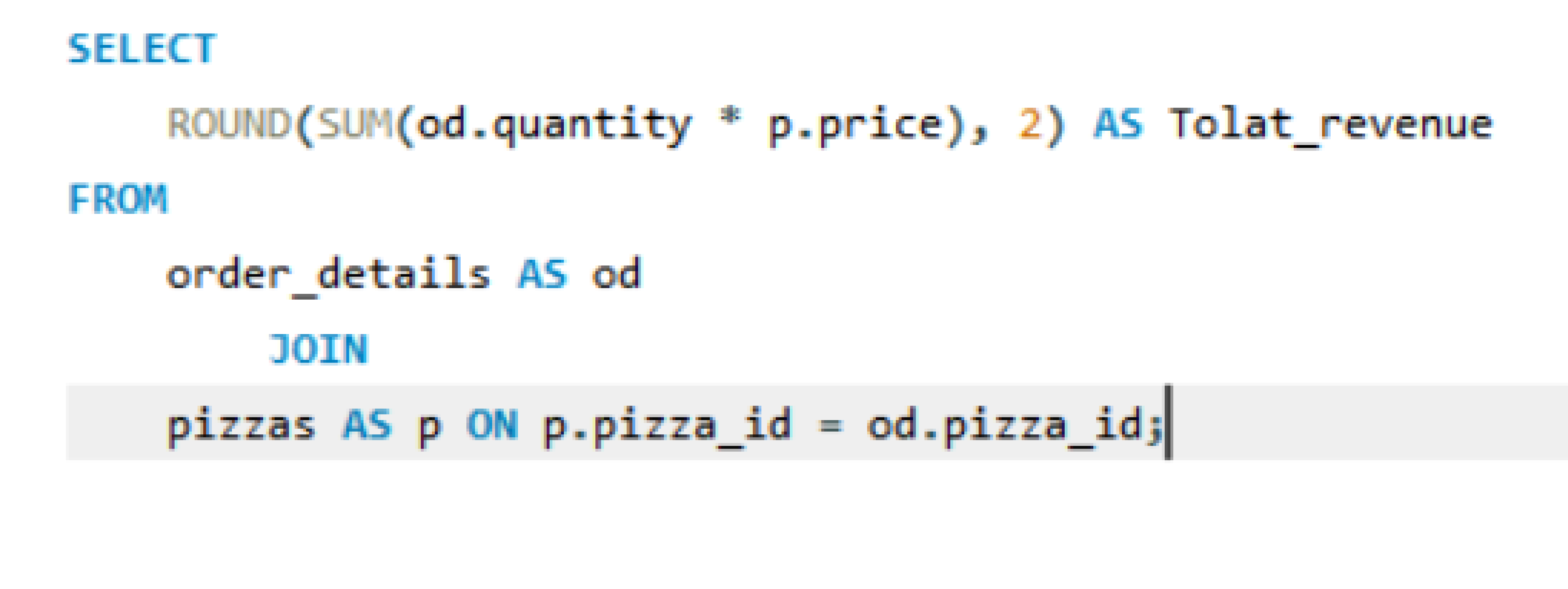


```
SELECT
    COUNT(order_id) AS Total_orders
FROM
    orders;
```

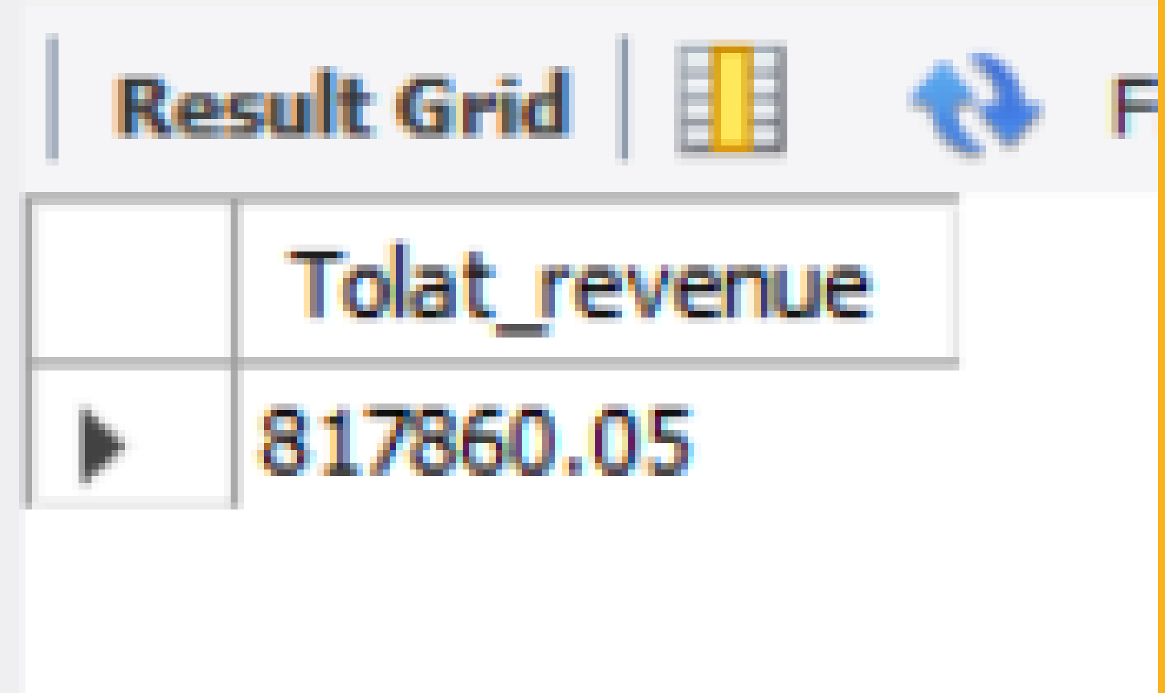


```
Total_orders
21350
```

CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES

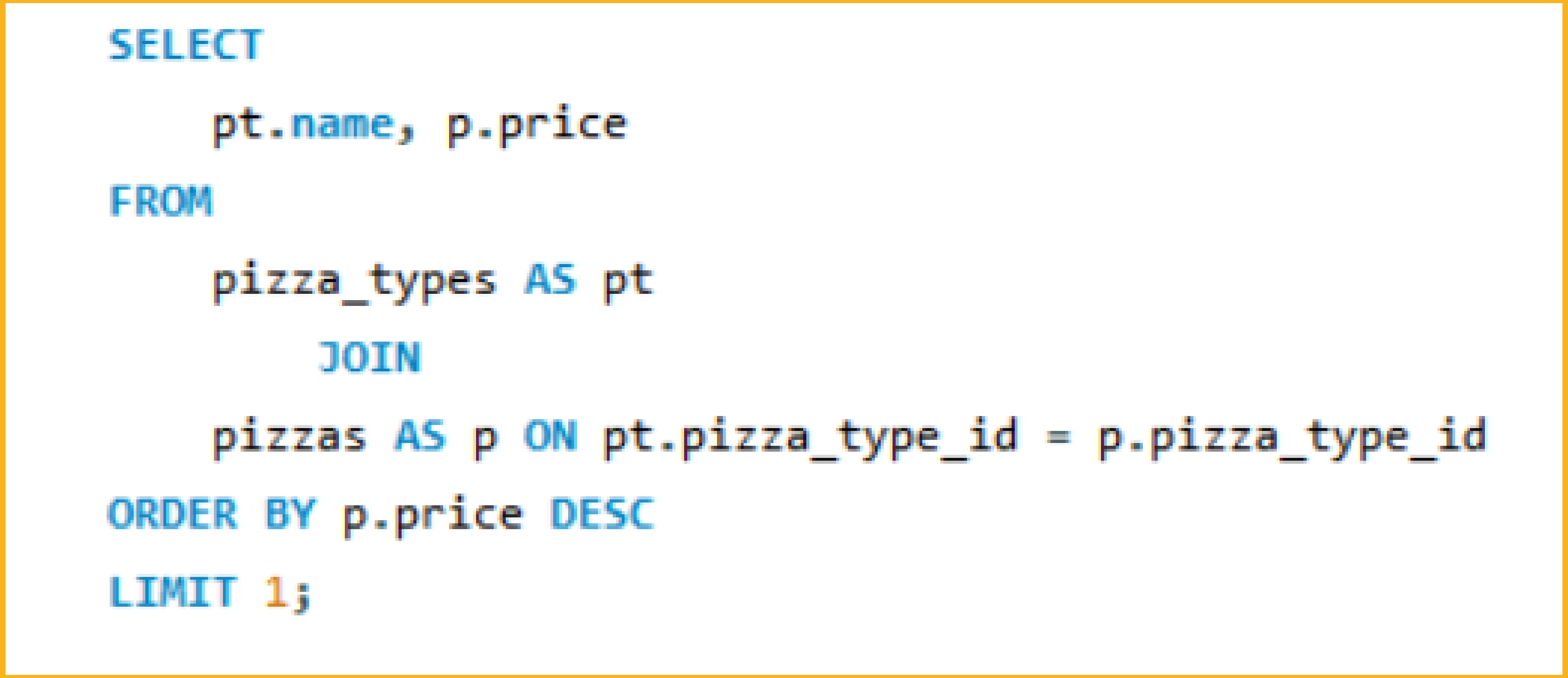


```sql
SELECT
    ROUND(SUM(od.quantity * p.price), 2) AS Tolat_revenue
FROM
    order_details AS od
        JOIN
    pizzas AS p ON p.pizza_id = od.pizza_id;
```

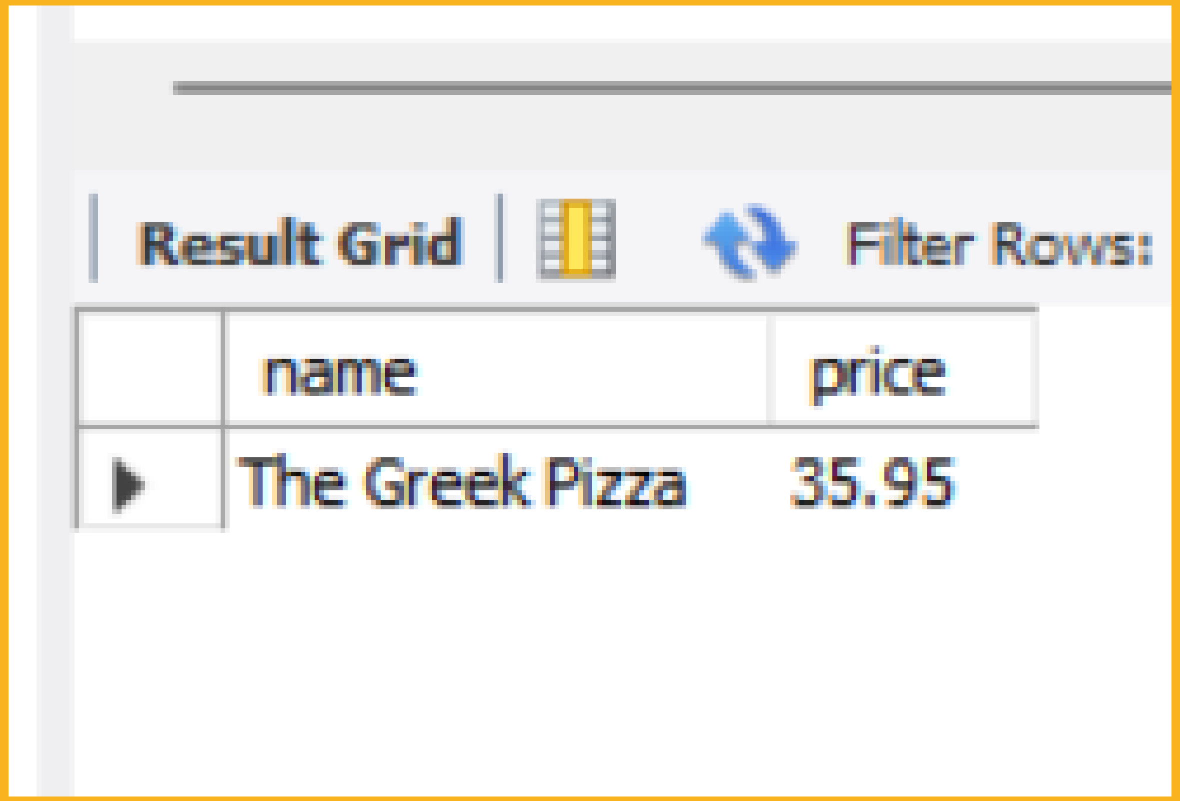


Tolat_revenue
817860.05

IDENTIFY THE HIGHEST-PRICED PIZZA.

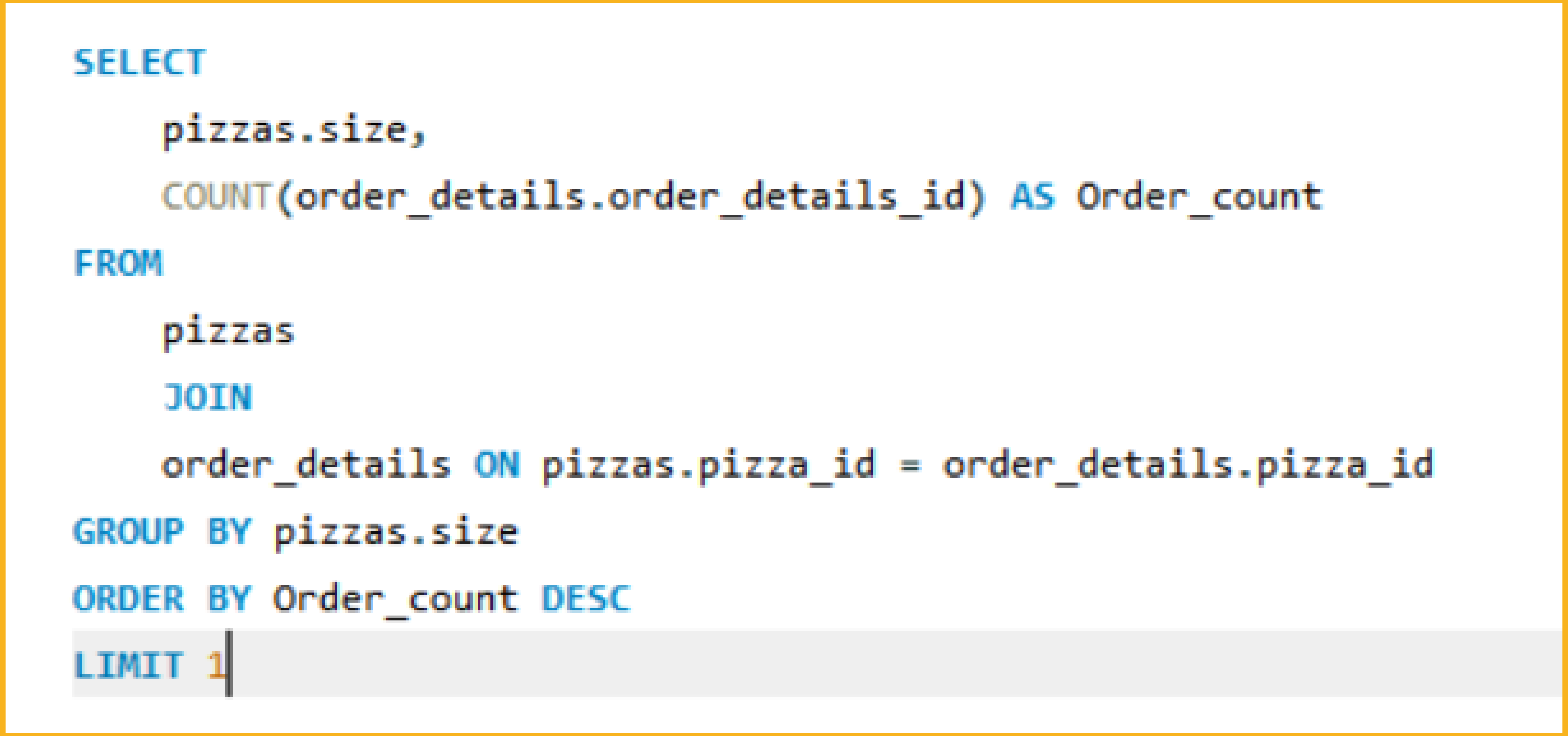


```
SELECT
    pt.name, p.price
FROM
    pizza_types AS pt
        JOIN
    pizzas AS p ON pt.pizza_type_id = p.pizza_type_id
ORDER BY p.price DESC
LIMIT 1;
```

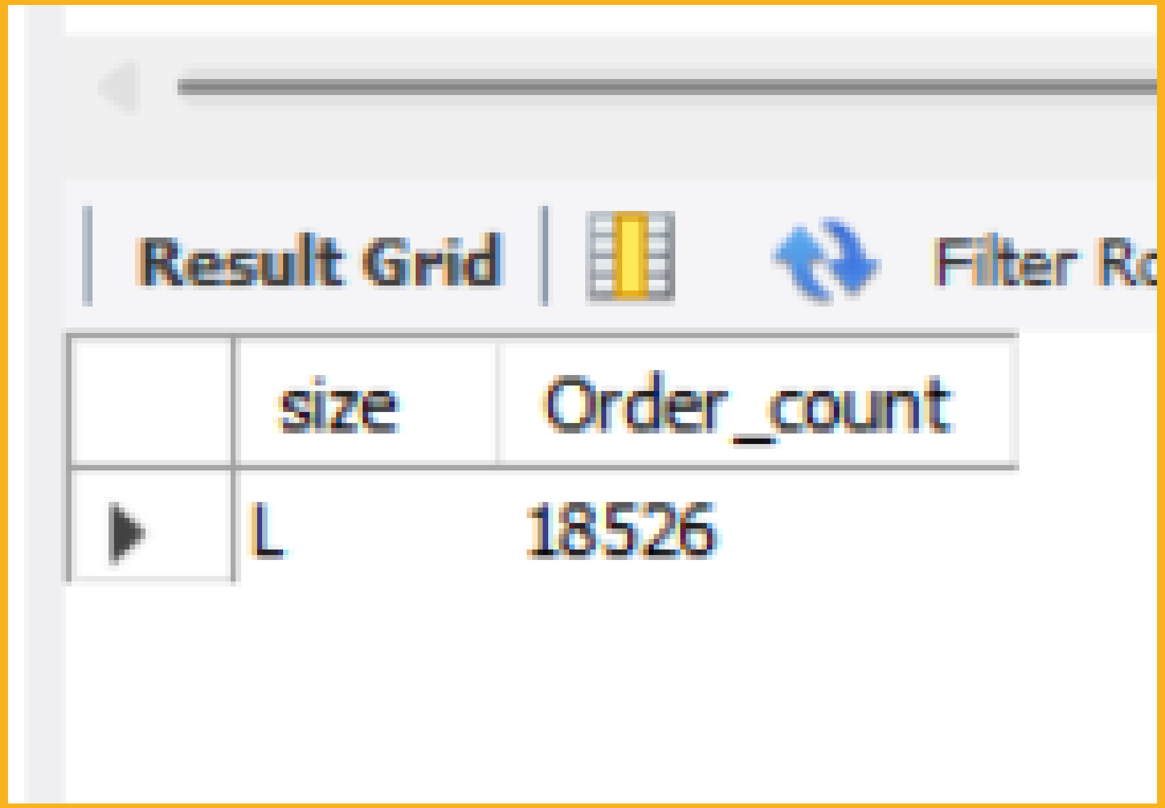


	name	price
▶	The Greek Pizza	35.95

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED

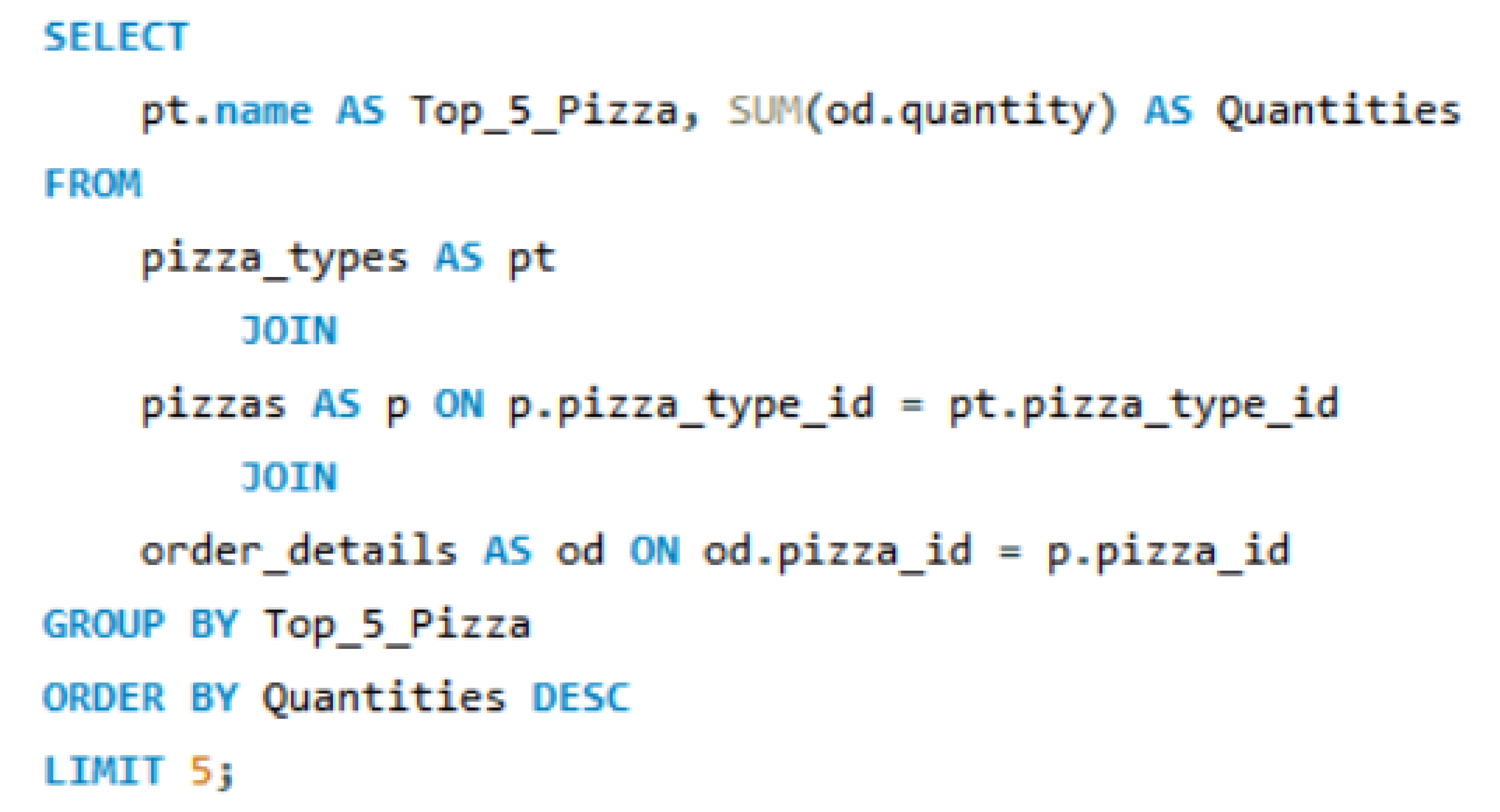


```
SELECT
    pizzas.size,
    COUNT(order_details.order_details_id) AS Order_count
FROM
    pizzas
    JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY pizzas.size
ORDER BY Order_count DESC
LIMIT 1
```

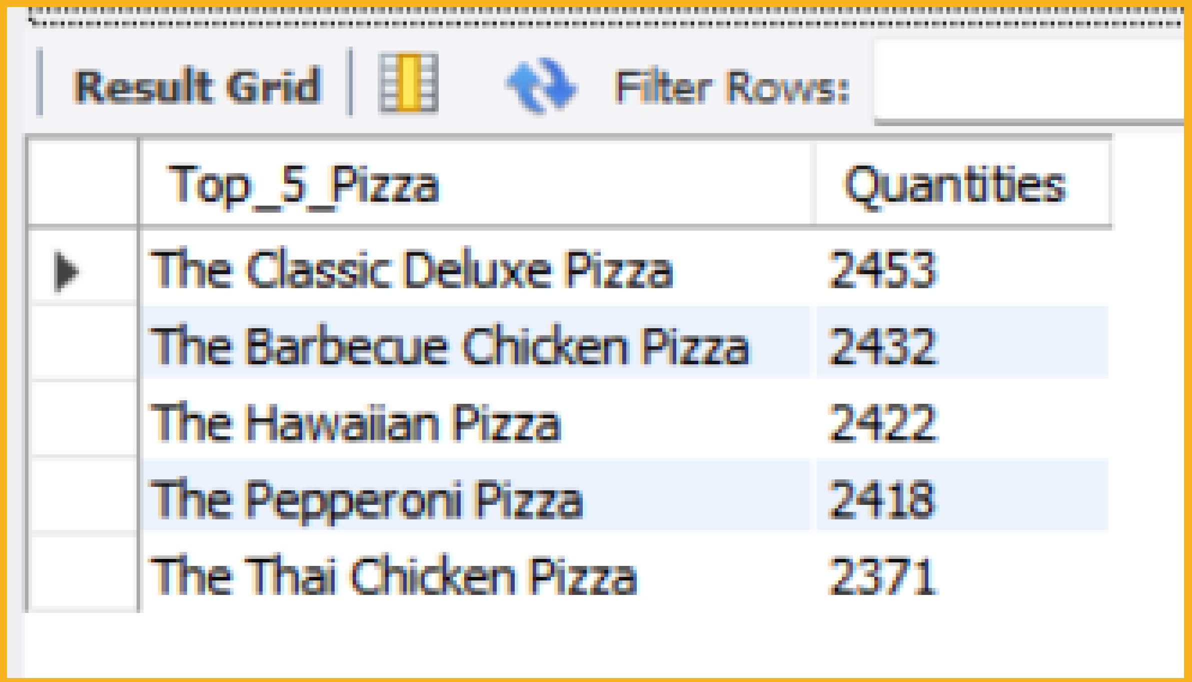


size	Order_count
L	18526

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

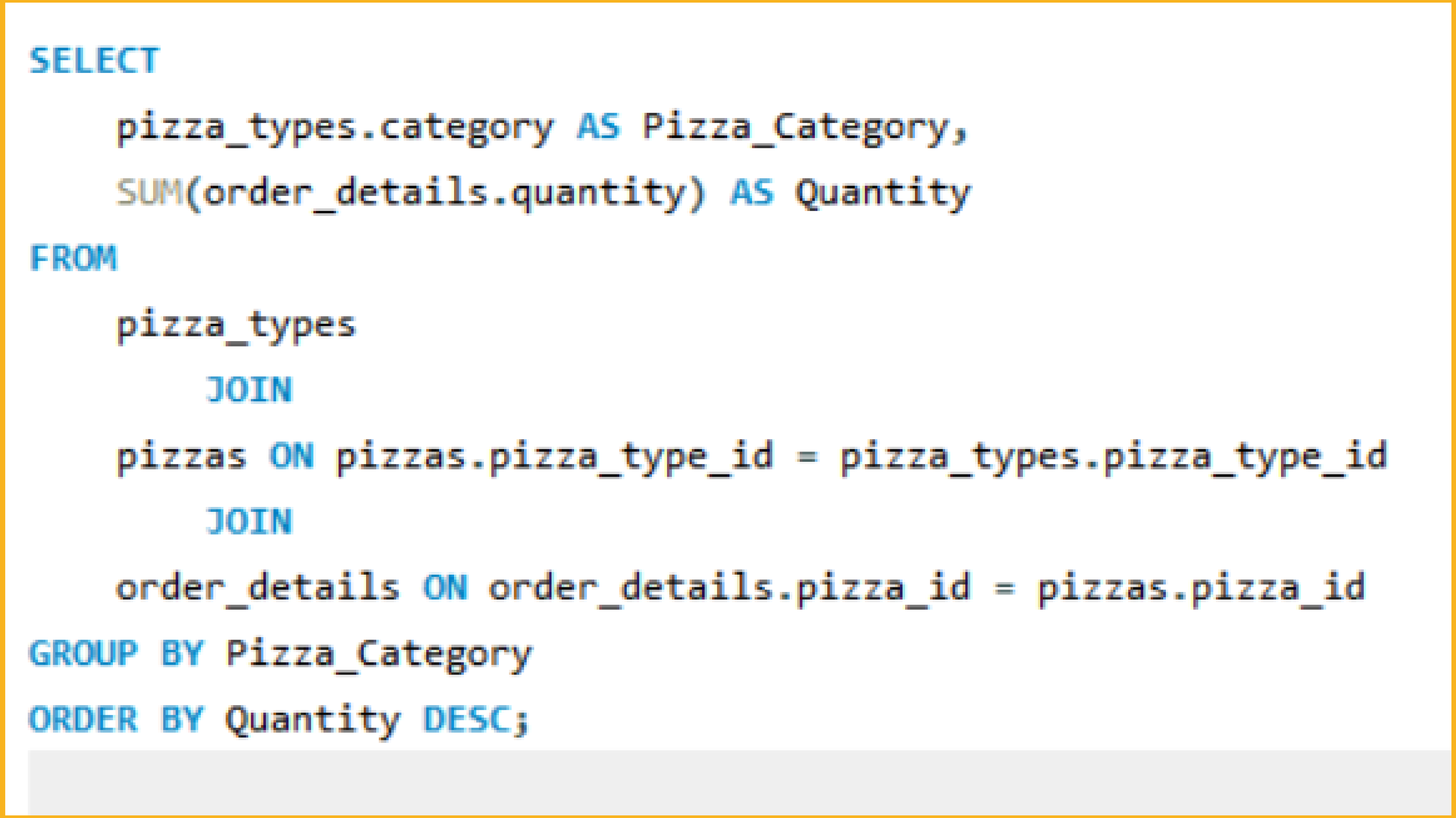


```
SELECT
    pt.name AS Top_5_Pizza, SUM(od.quantity) AS Quantities
FROM
    pizza_types AS pt
        JOIN
    pizzas AS p ON p.pizza_type_id = pt.pizza_type_id
        JOIN
    order_details AS od ON od.pizza_id = p.pizza_id
GROUP BY Top_5_Pizza
ORDER BY Quantities DESC
LIMIT 5;
```

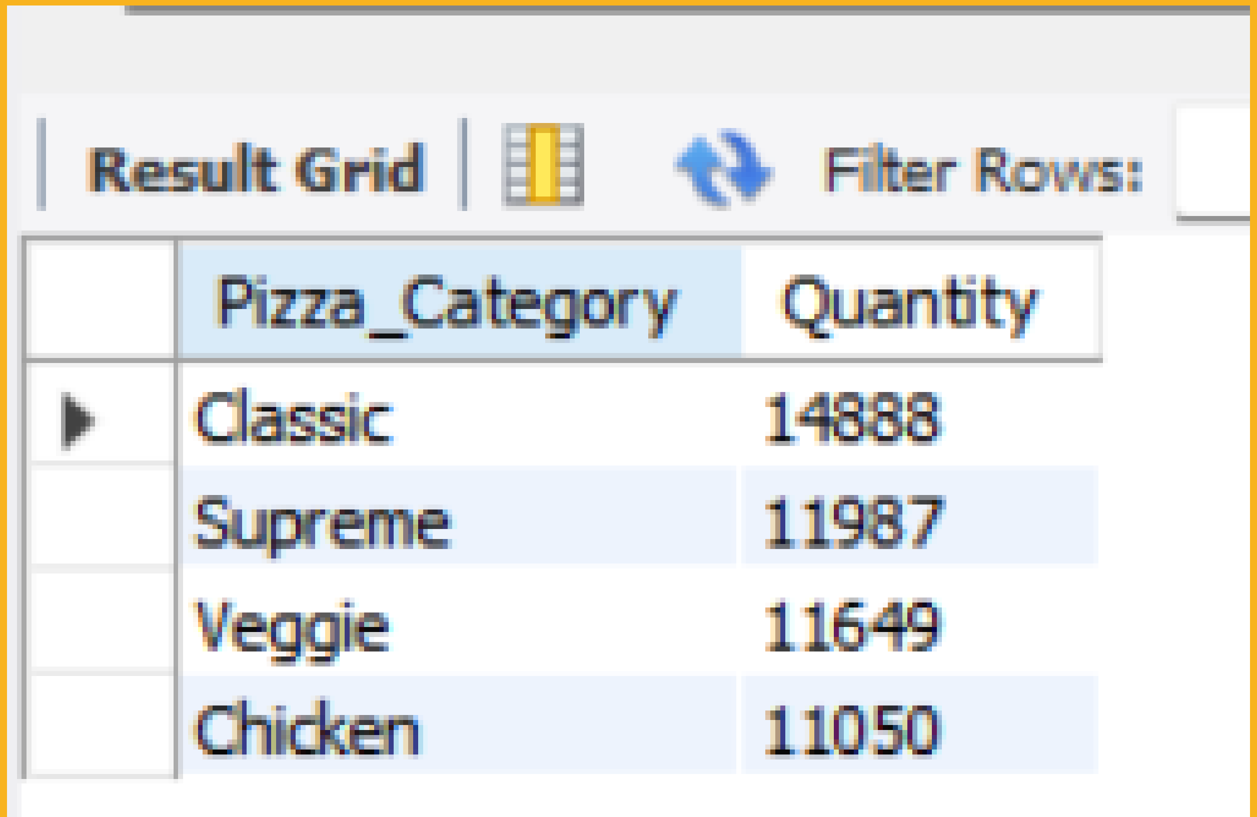


Top_5_Pizza	Quantities
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	2422
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371

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

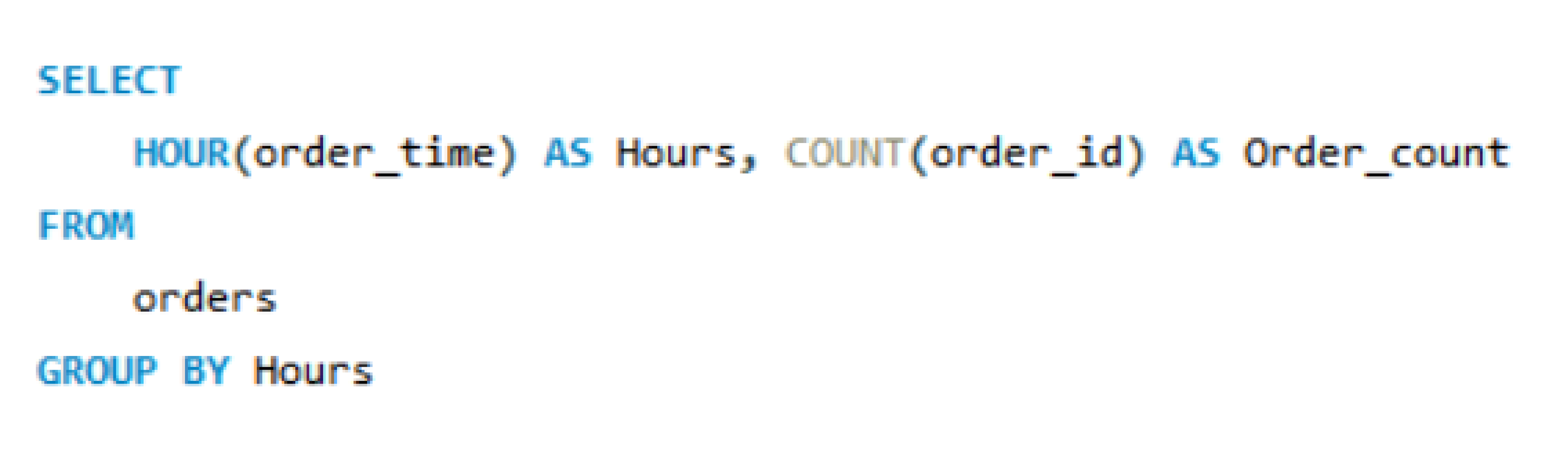


```
SELECT
    pizza_types.category AS Pizza_Category,
    SUM(order_details.quantity) AS Quantity
FROM
    pizza_types
        JOIN
    pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY Pizza_Category
ORDER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

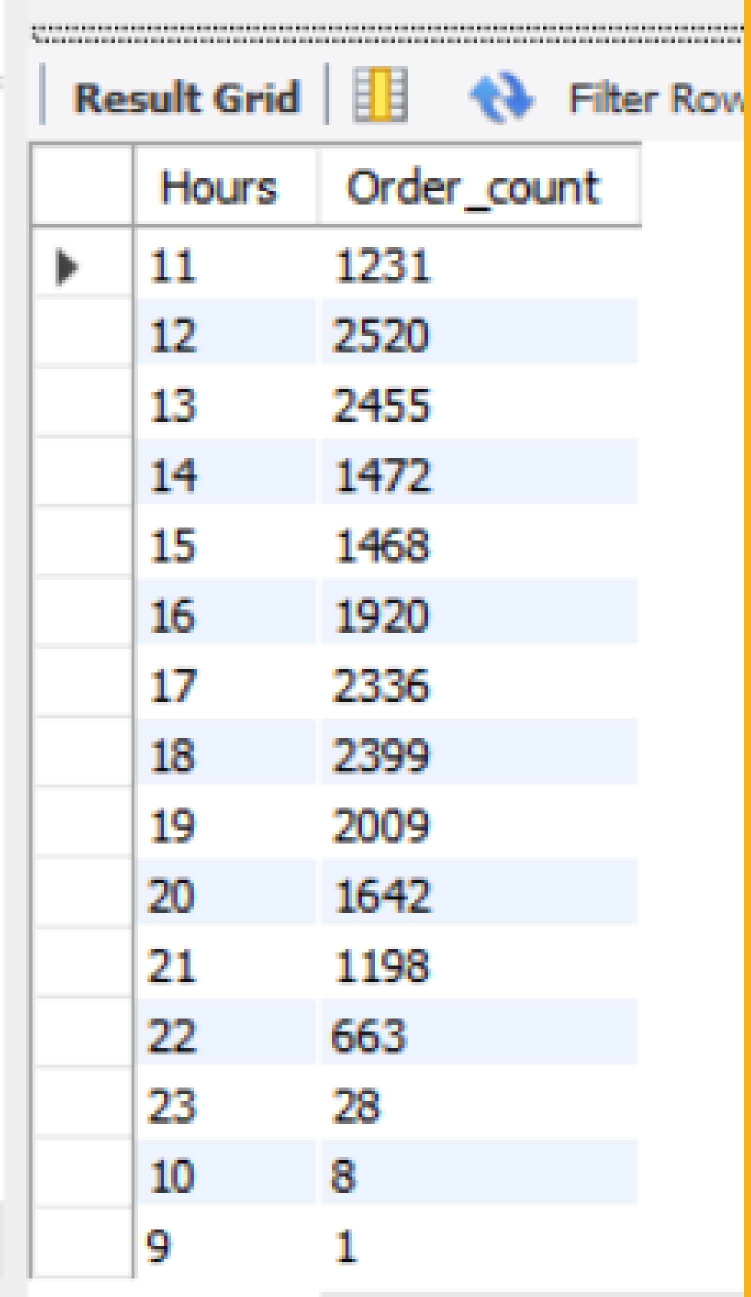


Pizza_Category	Quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.

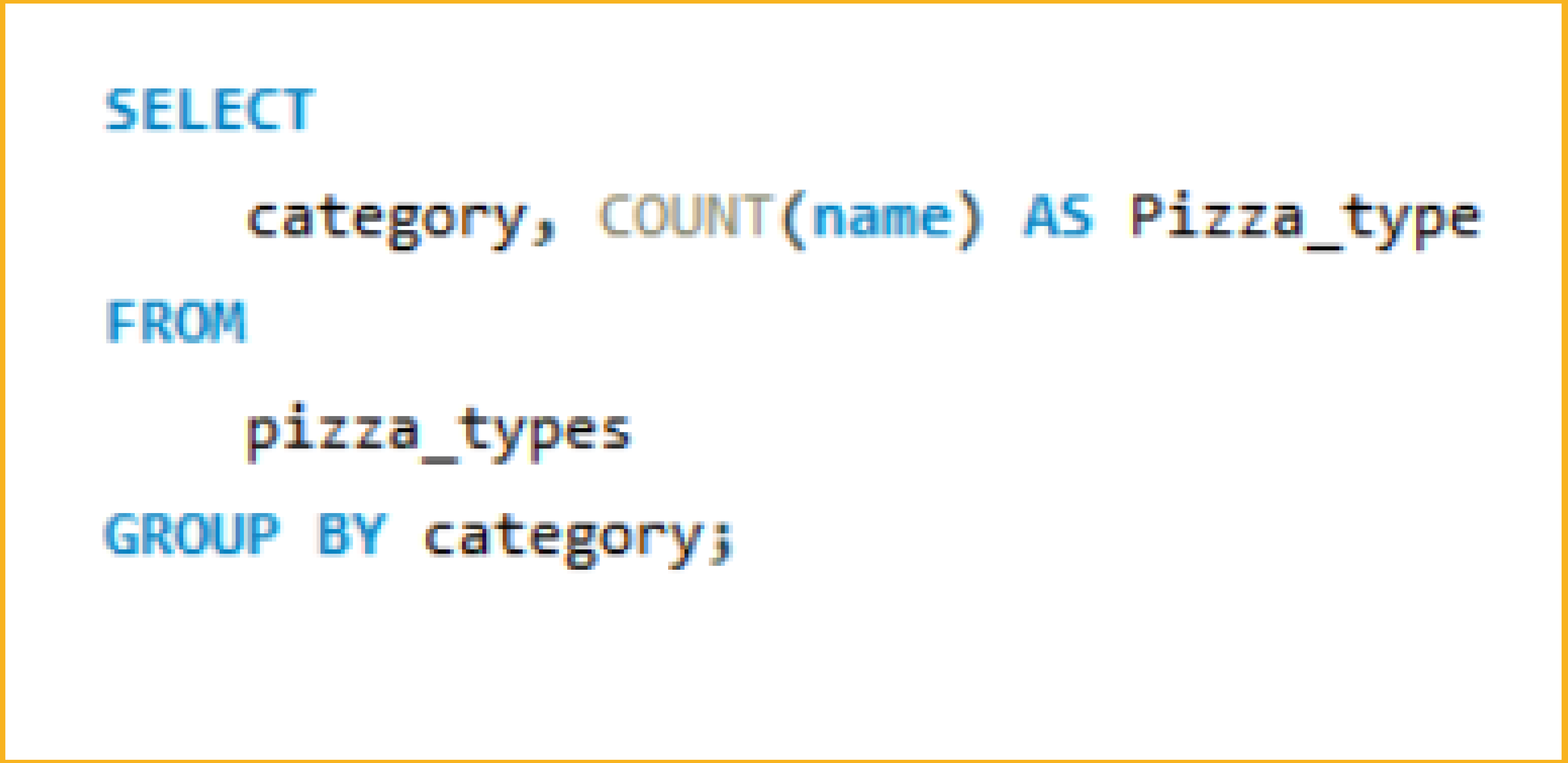


```sql
SELECT
    HOUR(order_time) AS Hours, COUNT(order_id) AS Order_count
FROM
    orders
GROUP BY Hours
```

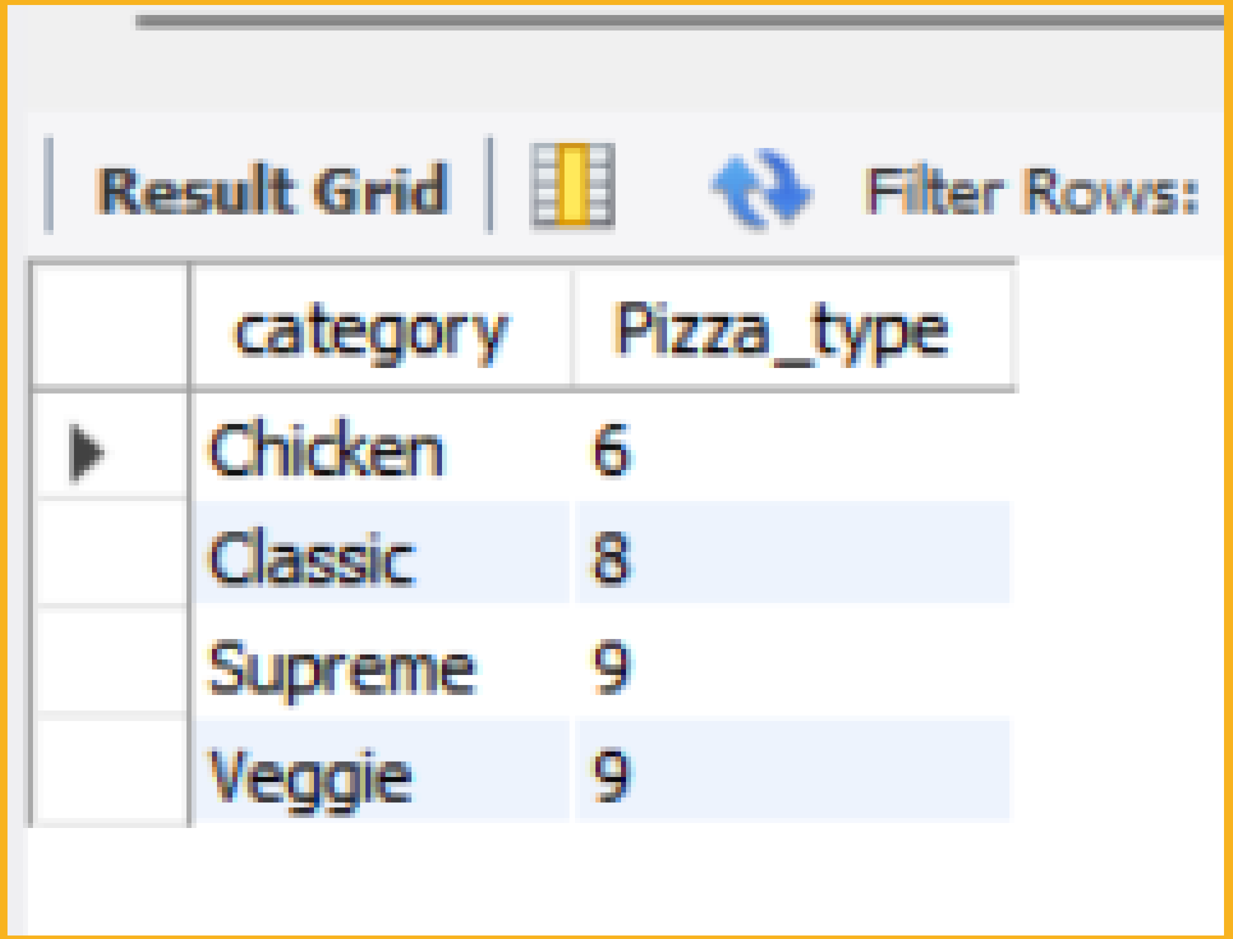


Hours	Order_count
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642
21	1198
22	663
23	28
10	8
9	1

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

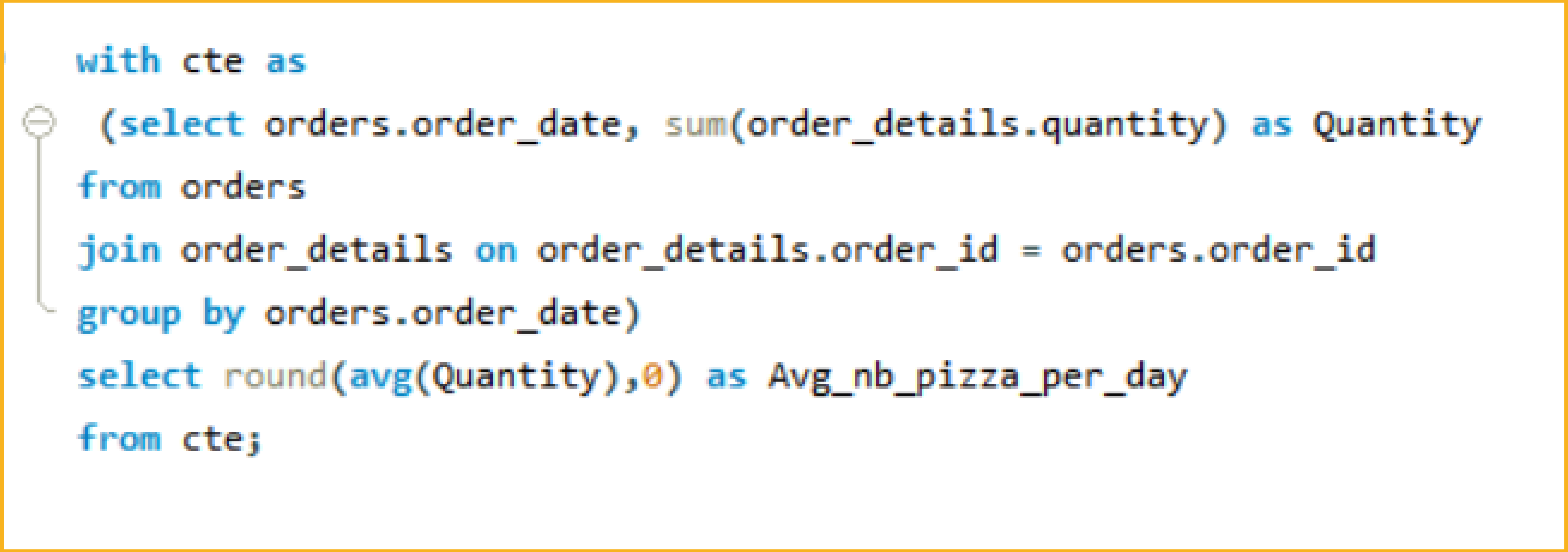


```
SELECT
    category, COUNT(name) AS Pizza_type
FROM
    pizza_types
GROUP BY category;
```

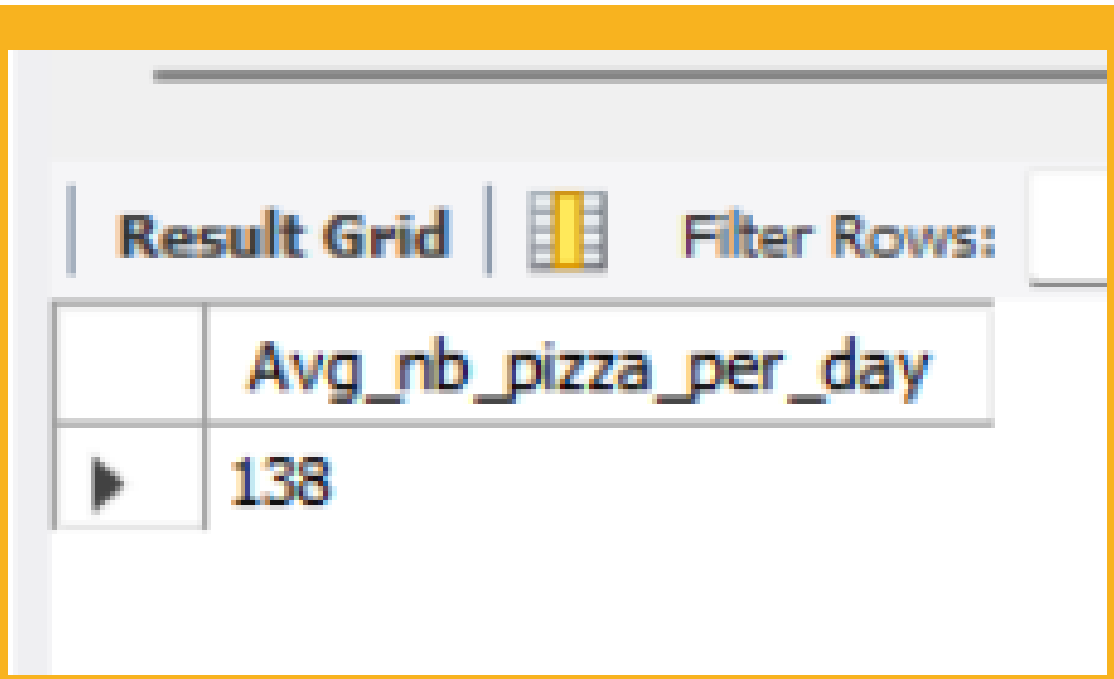


category	Pizza_type
Chicken	6
Classic	8
Supreme	9
Veggie	9

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

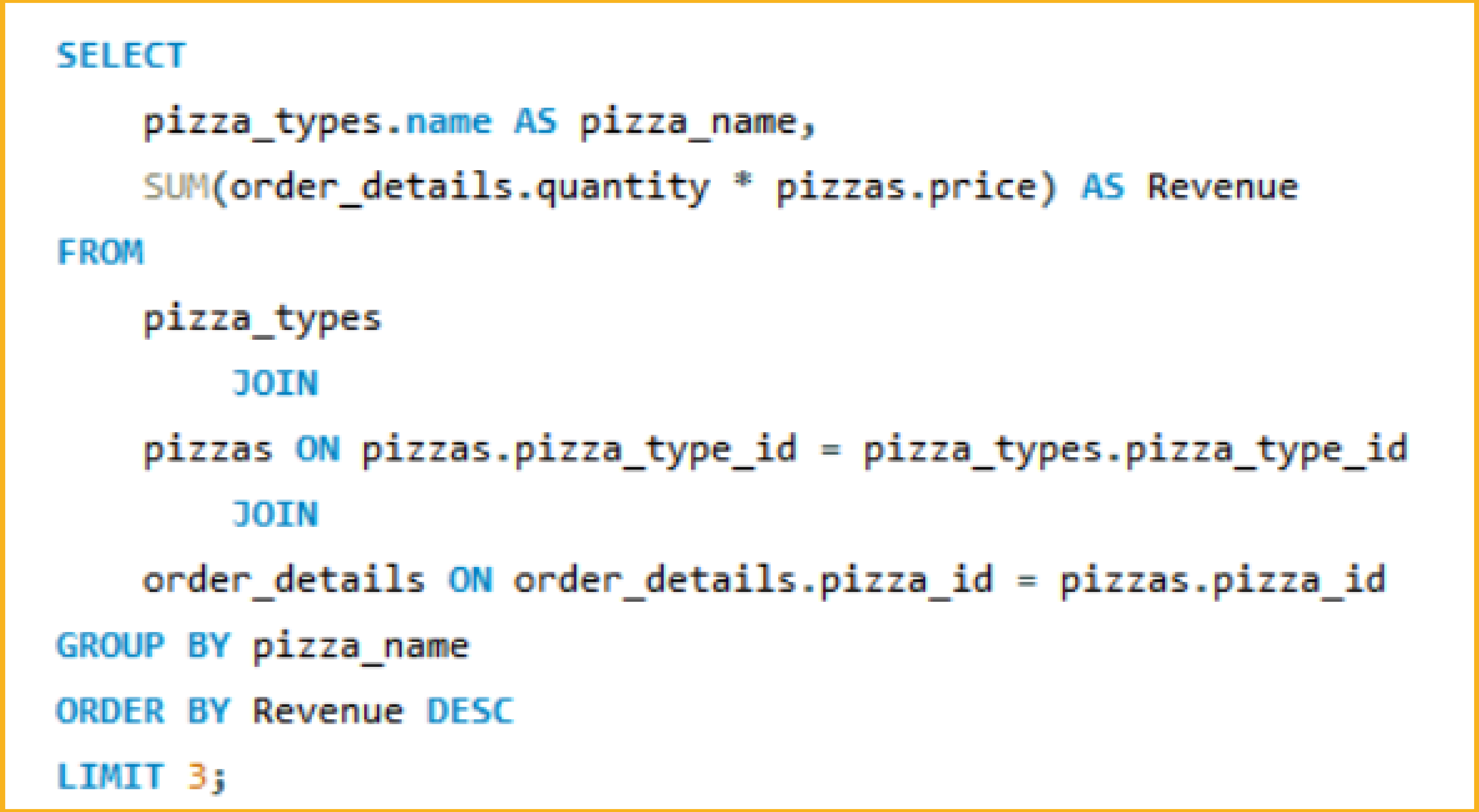


```sql
with cte as
 (select orders.order_date, sum(order_details.quantity) as Quantity
from orders
join order_details on order_details.order_id = orders.order_id
group by orders.order_date)
select round(avg(Quantity),0) as Avg_nb_pizza_per_day
from cte;
```

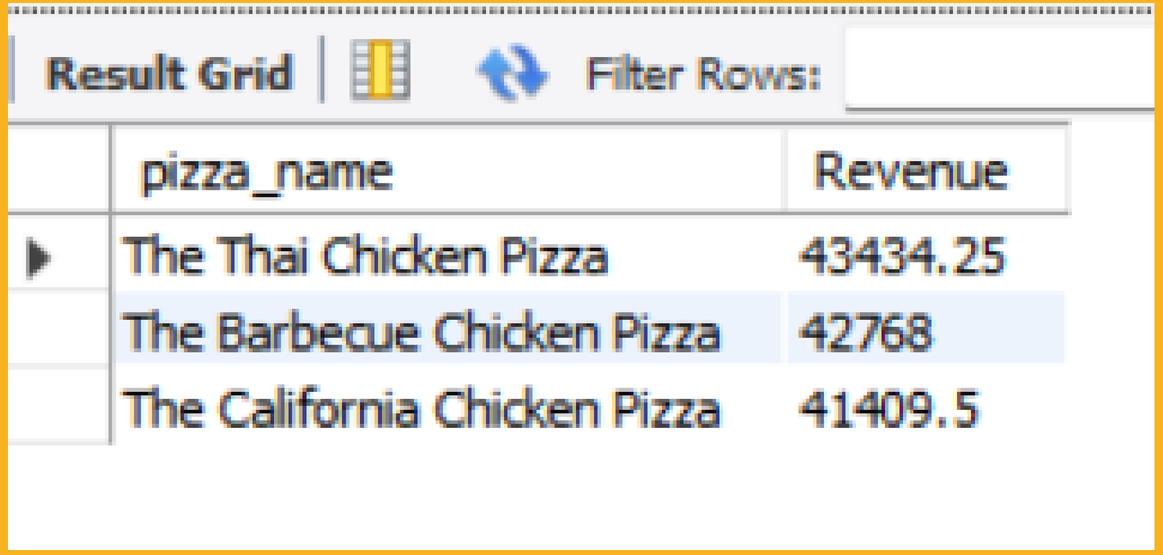


```
Avg_nb_pizza_per_day
138
```

DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE



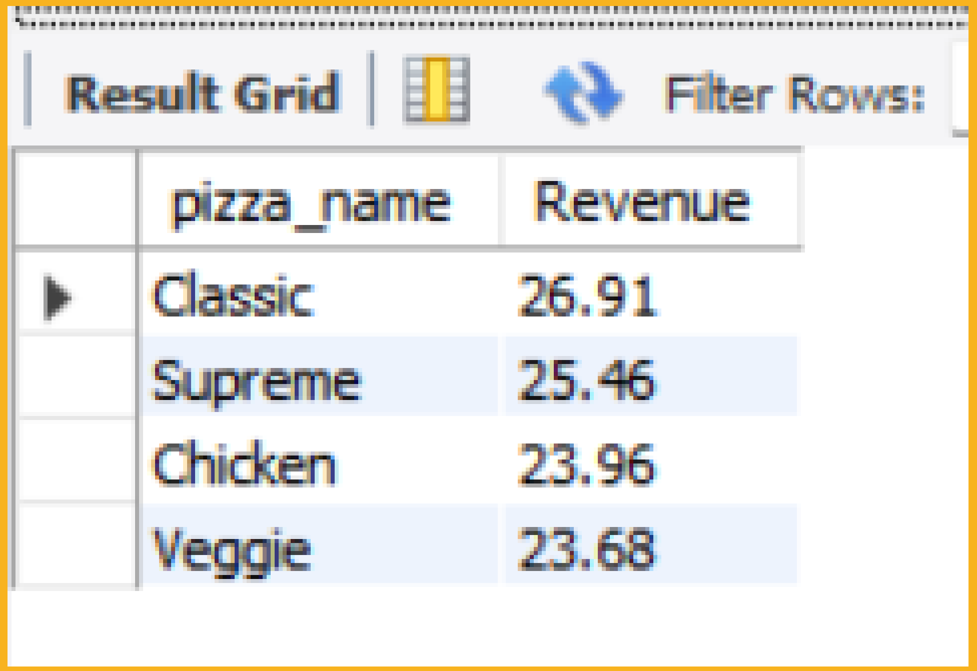
```
SELECT
    pizza_types.name AS pizza_name,
    SUM(order_details.quantity * pizzas.price) AS Revenue
FROM
    pizza_types
        JOIN
    pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_name
ORDER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



pizza_name	Revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5

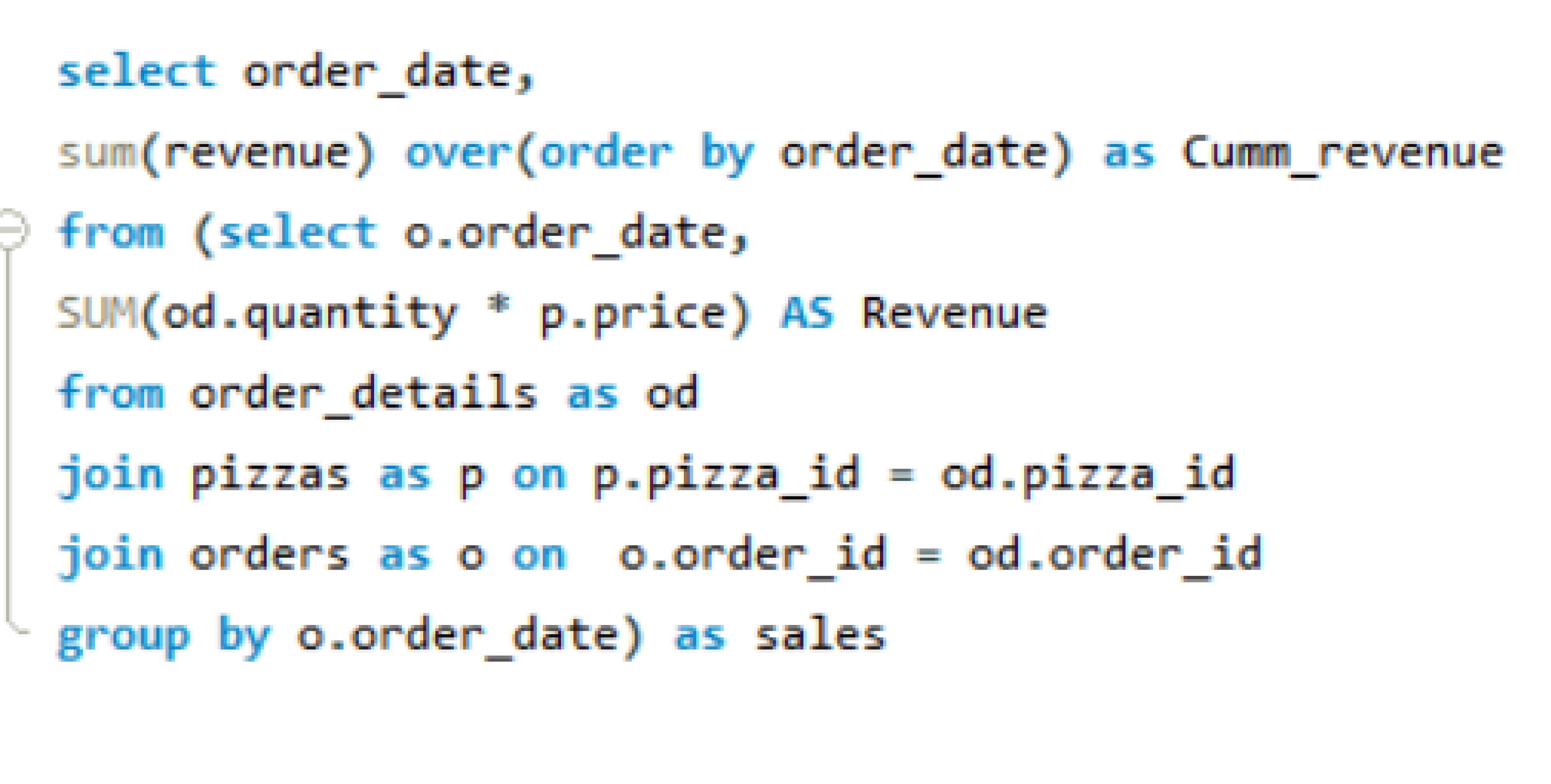
CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPE TO TOTAL REVENUE.

```
SELECT
  pizza_types.category AS pizza_name,
  ROUND(SUM(order_details.quantity * pizzas.price)
       / (SELECT
      ROUND(SUM(od.quantity * p.price), 2) AS Tolat_revenue
    FROM
      order_details AS od
        JOIN
      pizzas AS p ON p.pizza_id = od.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_name
ORDER BY Revenue DESC
```

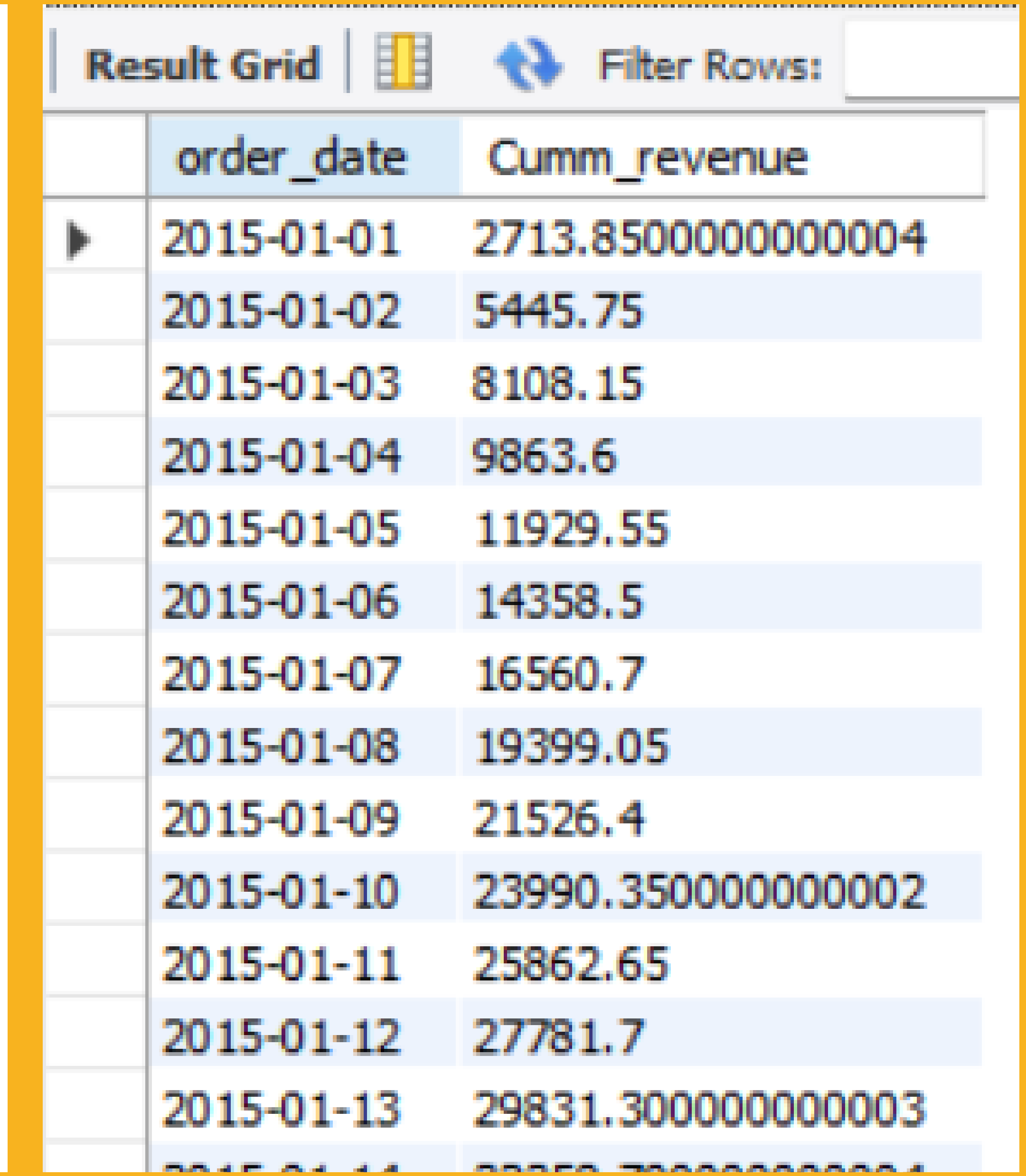


pizza_name	Revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME.



```
select order_date,
sum(revenue) over(order by order_date) as Cumm_revenue
from (select o.order_date,
SUM(od.quantity * p.price) AS Revenue
from order_details as od
join pizzas as p on p.pizza_id = od.pizza_id
join orders as o on  o.order_id = od.order_id
group by o.order_date) as sales
```

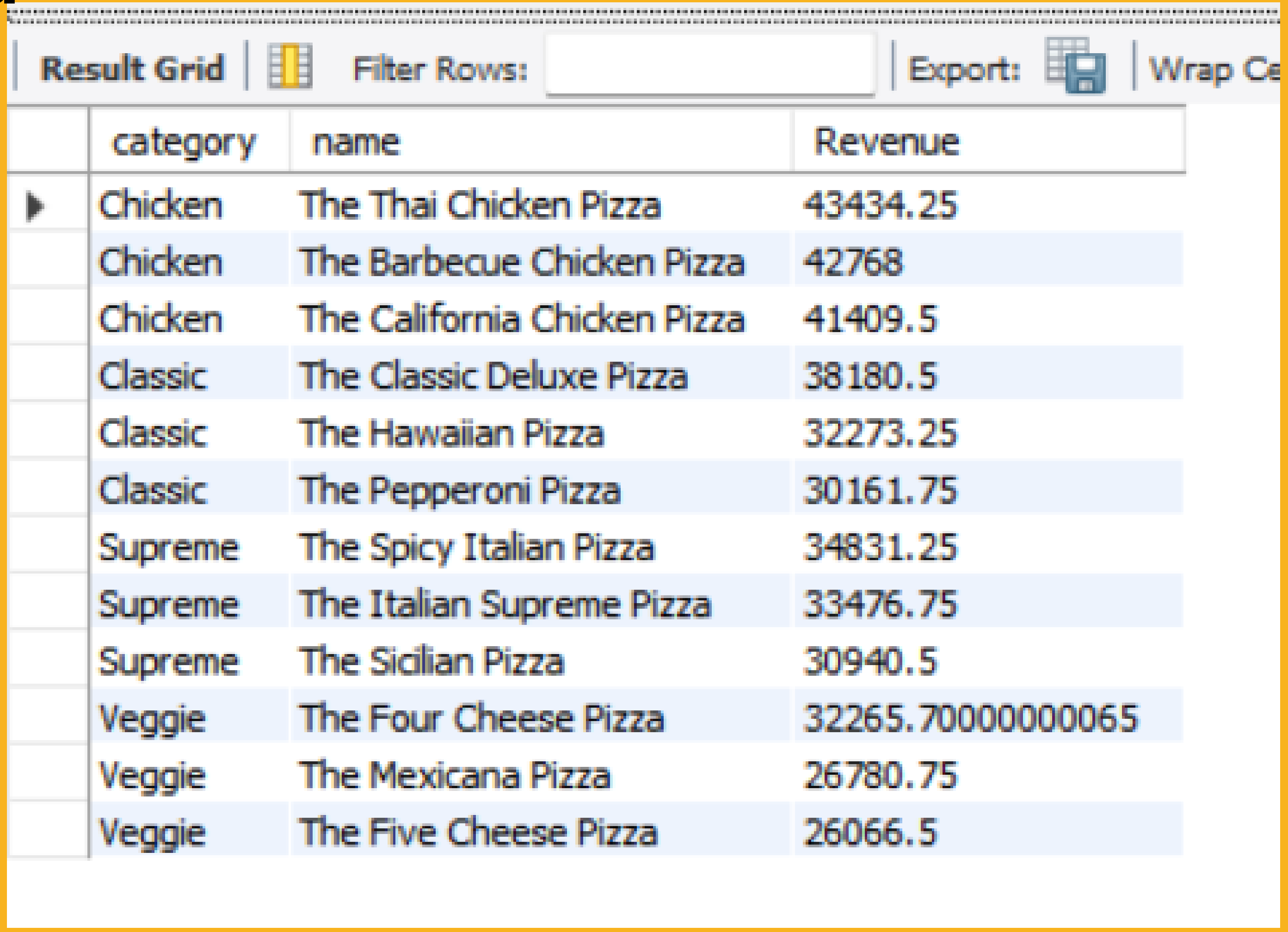


order_date	Cumm_revenue
2015-01-01	2713.850000000004
2015-01-02	5445.75
2015-01-03	8108.15
2015-01-04	9863.6
2015-01-05	11929.55
2015-01-06	14358.5
2015-01-07	16560.7
2015-01-08	19399.05
2015-01-09	21526.4
2015-01-10	23990.350000000002
2015-01-11	25862.65
2015-01-12	27781.7
2015-01-13	29831.300000000003

DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE FOR EACH PIZZA CATEGORY

```
with cte as (
select pizza_types.category, pizza_types.name,
sum(order_details.quantity * pizzas.price) as revenue,
rank() over(partition by pizza_types.category
order by sum(order_details.quantity * pizzas.price) desc) as rnk
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, pizza_types.name)

SELECT category, name, Revenue
FROM cte
WHERE rnk <= 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	32265.70000000065
Veggie	The Mexicana Pizza	26780.75
Veggie	The Five Cheese Pizza	26066.5

CONCLUSION

Summary of key findings:

- Insights into customer preferences.
- Identification of top-performing products.
- Trends in sales and revenue.
- Importance of data-driven decision-making in marketing

THANK YOU

Questions and Guidance will be appreticed