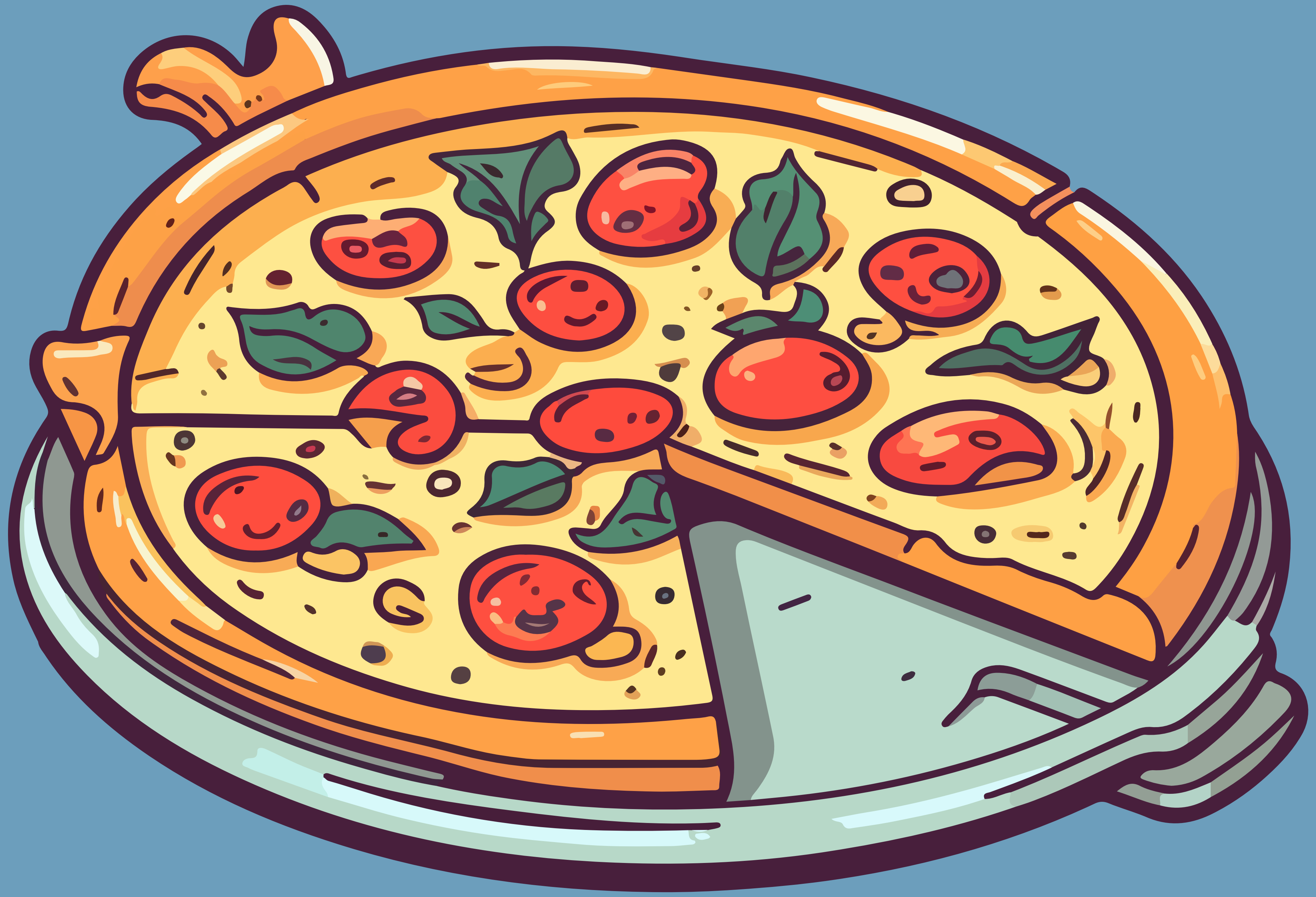


PIZZA SALES PROJECT

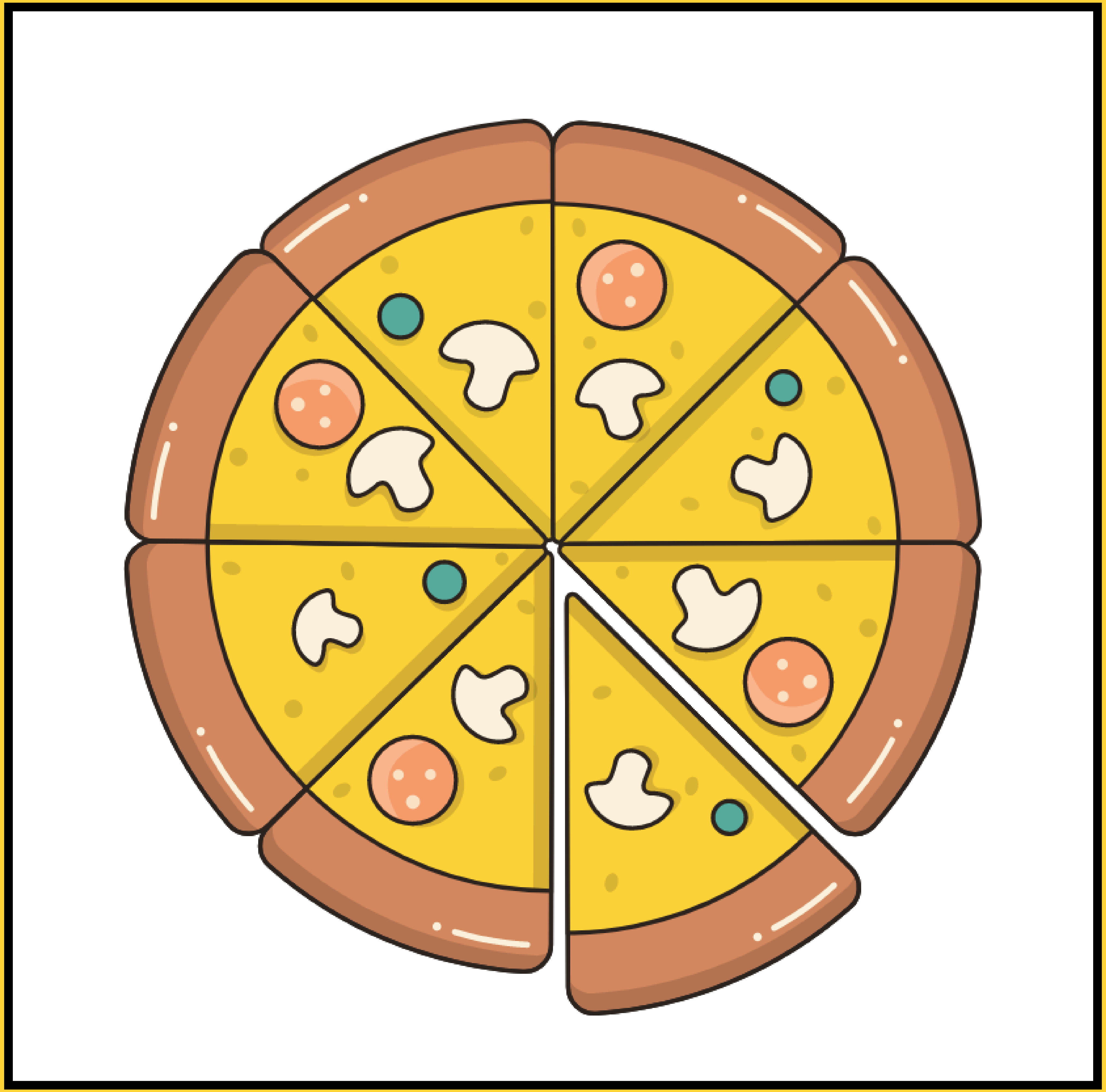
MUNAWER JABEEN

INTRODUCTION

- THIS PROJECT ANALYZES PIZZA SALES DATA TO UNCOVER TRENDS AND INSIGHTS ABOUT CUSTOMER ORDERING BEHAVIOR.
- THE DATASET INCLUDES DETAILS ON ORDERS, ORDER DETAILS, PIZZA TYPES, AND PIZZAS.
- SQL WAS USED TO CLEAN, JOIN, AND QUERY THE DATA FOR MEANINGFUL BUSINESS INSIGHTS.
- THE OBJECTIVE IS TO IDENTIFY TOP-SELLING PIZZAS, CALCULATE REVENUE, AND HIGHLIGHT KEY PATTERNS IN SALES PERFORMANCE.
- THESE INSIGHTS CAN SUPPORT BETTER DECISION-MAKING IN INVENTORY MANAGEMENT, MARKETING, AND MENU PLANNING.

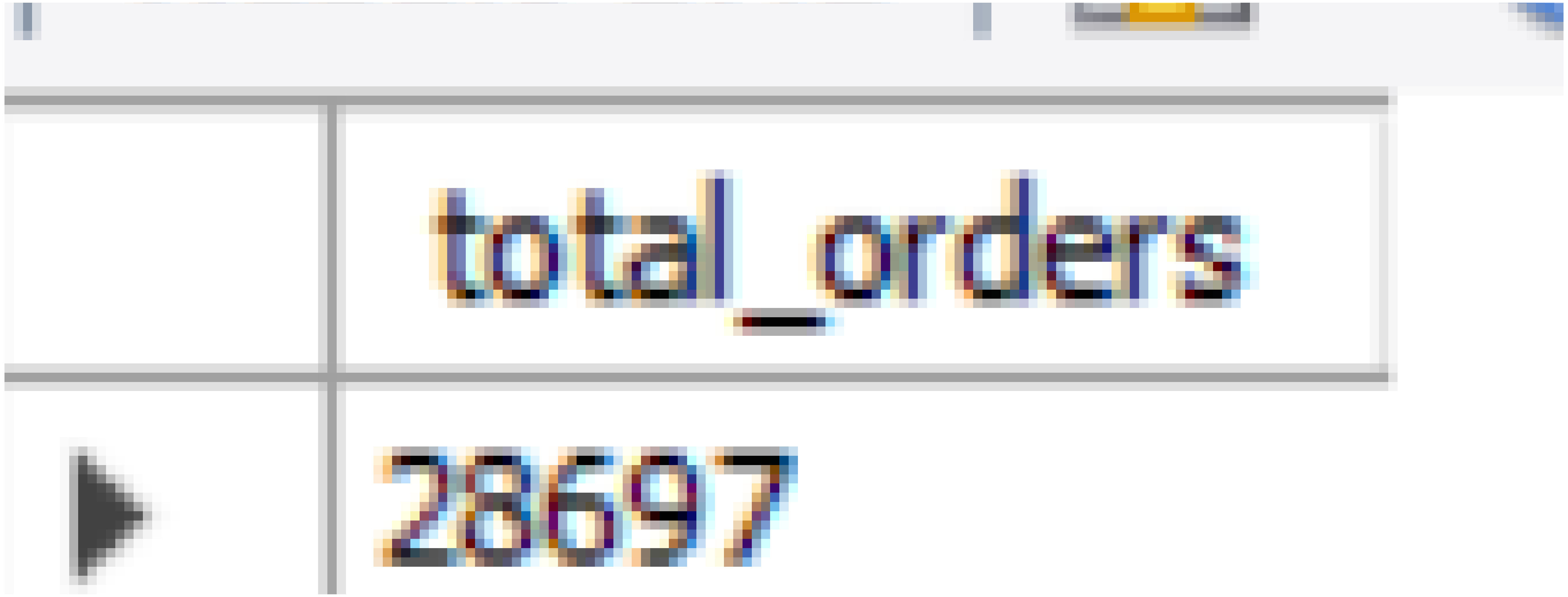
1

Retrieve the total number of orders placed.

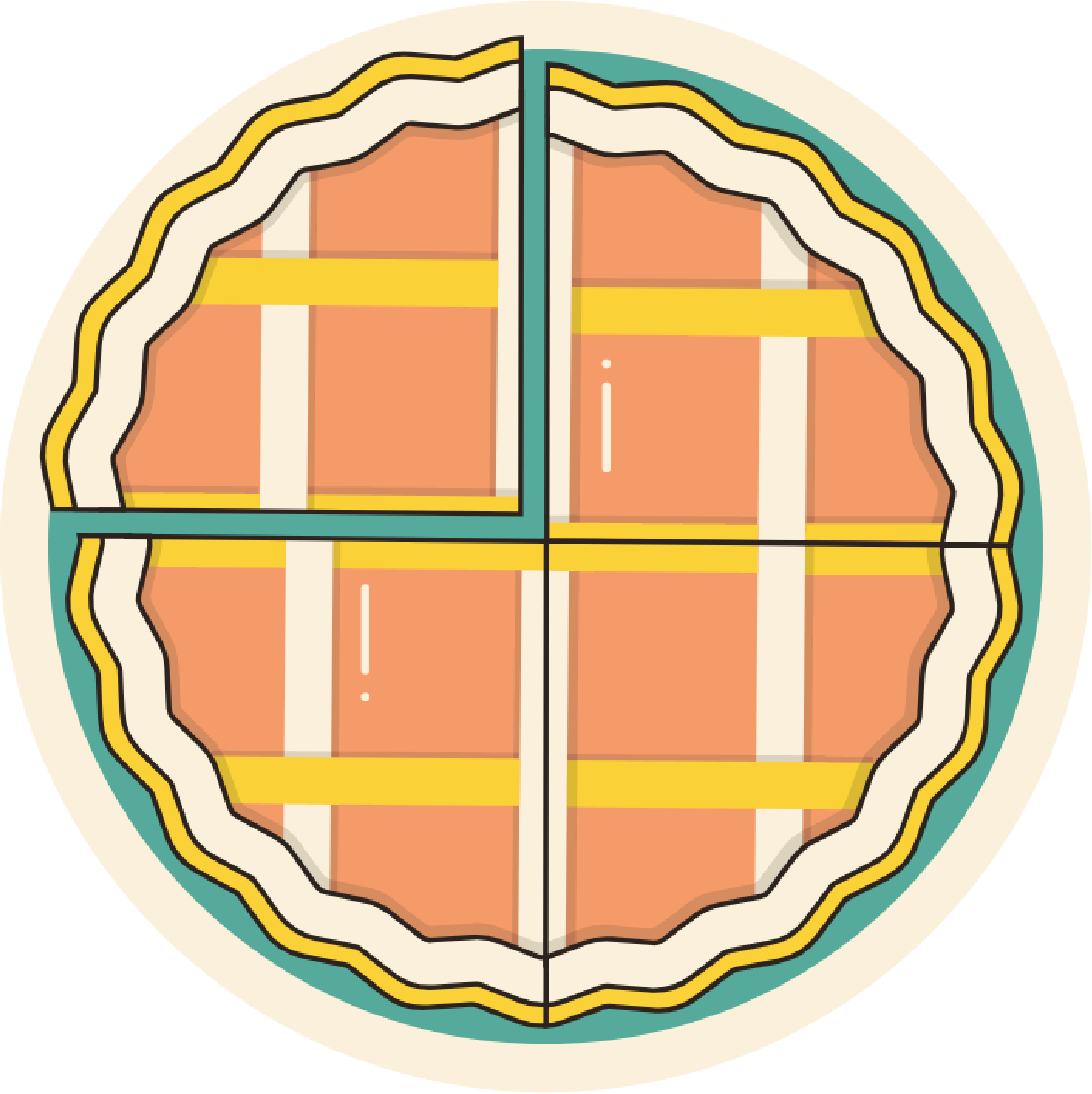


ANSWER

```
select
count(order_id) as total_orders from orders;
```



total_orders
28697

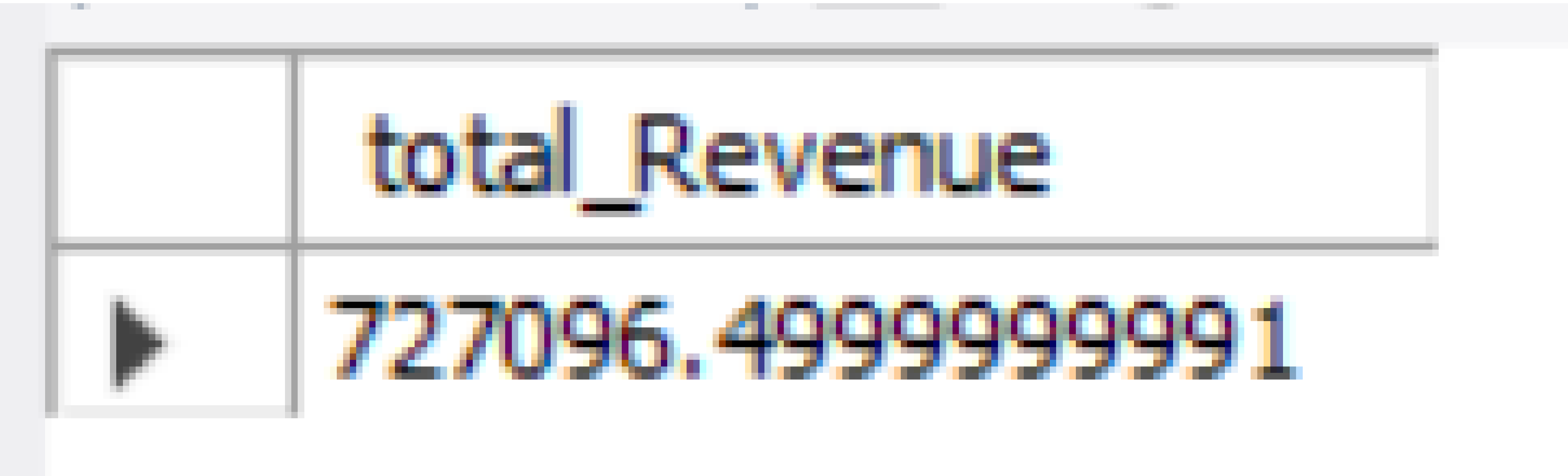


2

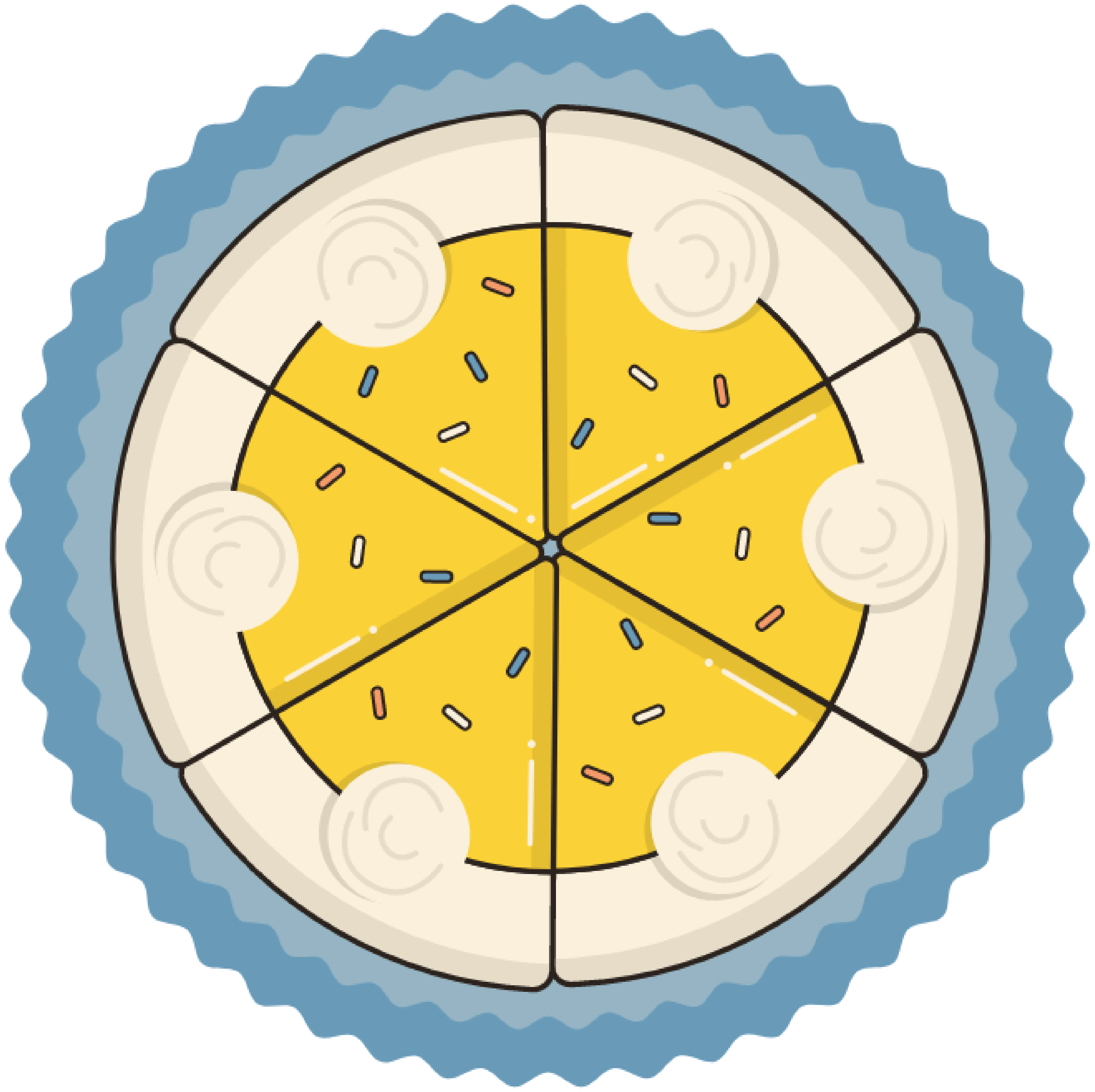
Calculate the total revenue generated from pizza sales

ANSWER

```
Select SUM(order_details.quantity*pizzas.price) as total_Revenue
from order_details join pizzas
ON order_details.pizza_id = pizzas.pizza_id
```



	total_Revenue
▶	727096.4999999991

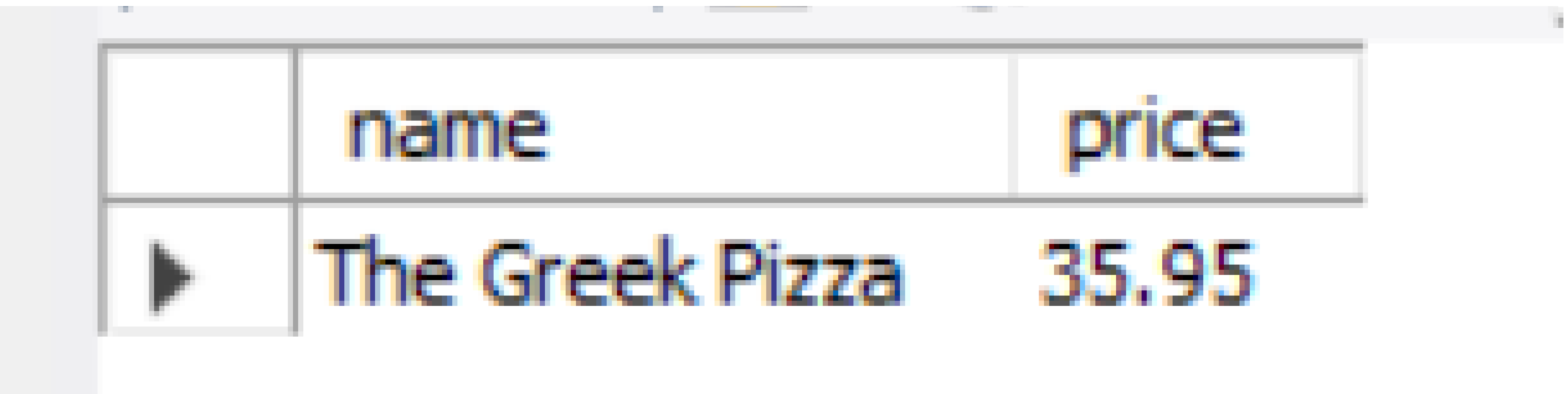


3

Identify the highest-priced pizza.

ANSWER

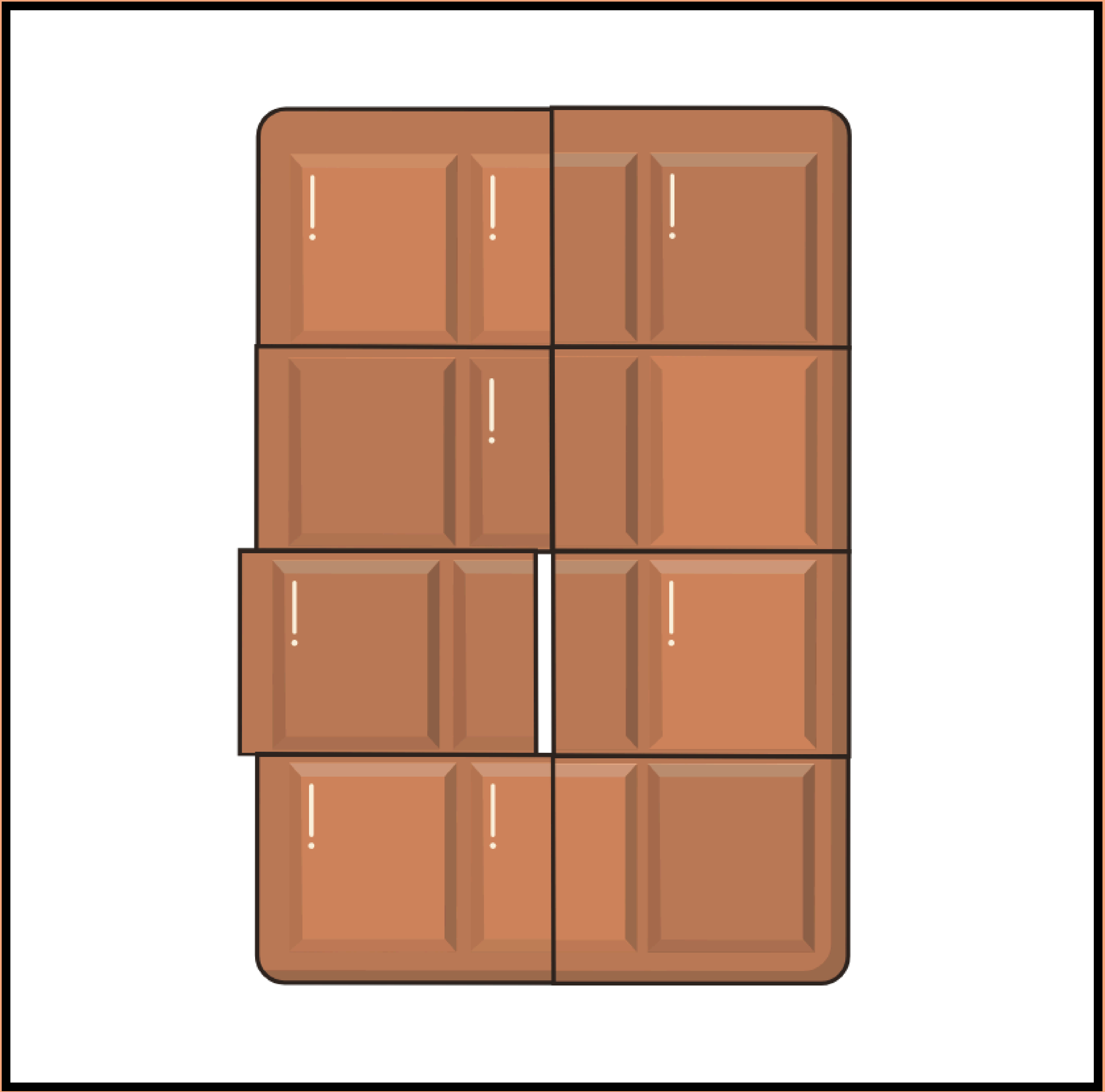
```
SELECT  pizza_types.name, pizzas.price  FROM   pizza_types JOIN
   pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
         ORDER BY pizzas.price DESC    LIMIT 1;
```



	name	price
▶	The Greek Pizza	35.95

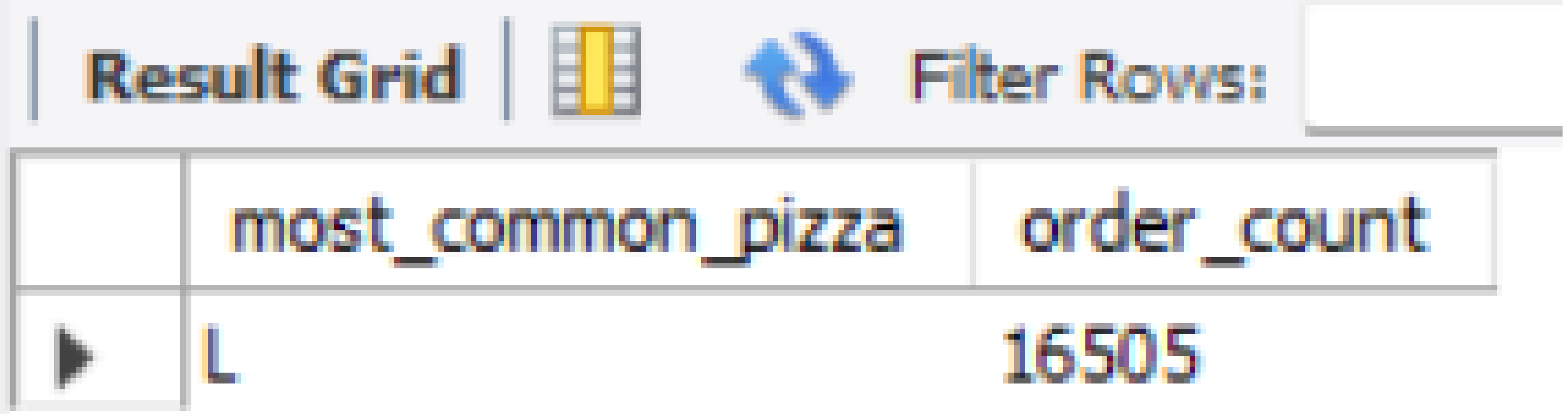
4

Identify the most common pizza size ordered.



ANSWER

```
select order_id,size, count(size) as pop_size
from  pizza.order_details od left join pizza.pizzas m
on od.pizza_id = m.pizza_id   group by size,order_id
order by pop_size desc limit 5
```



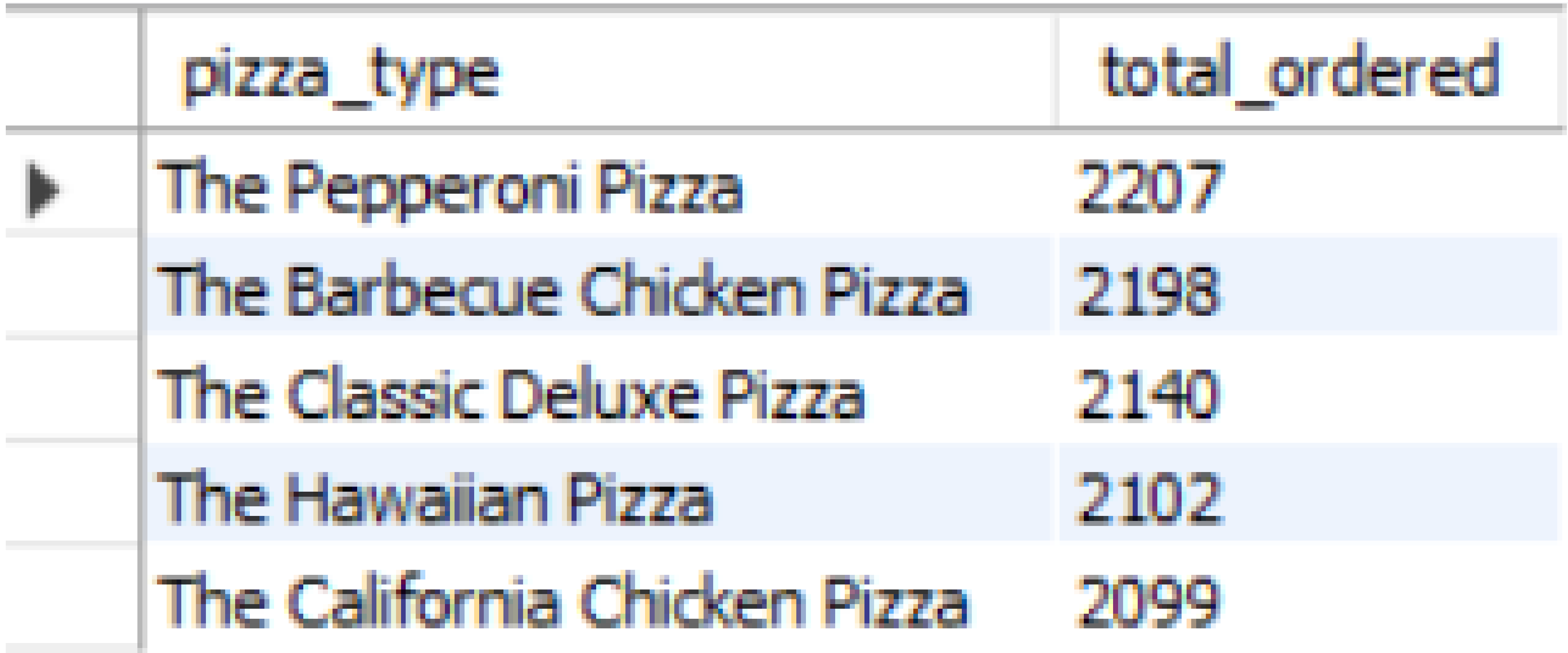
most_common_pizza	order_count
L	16505

5

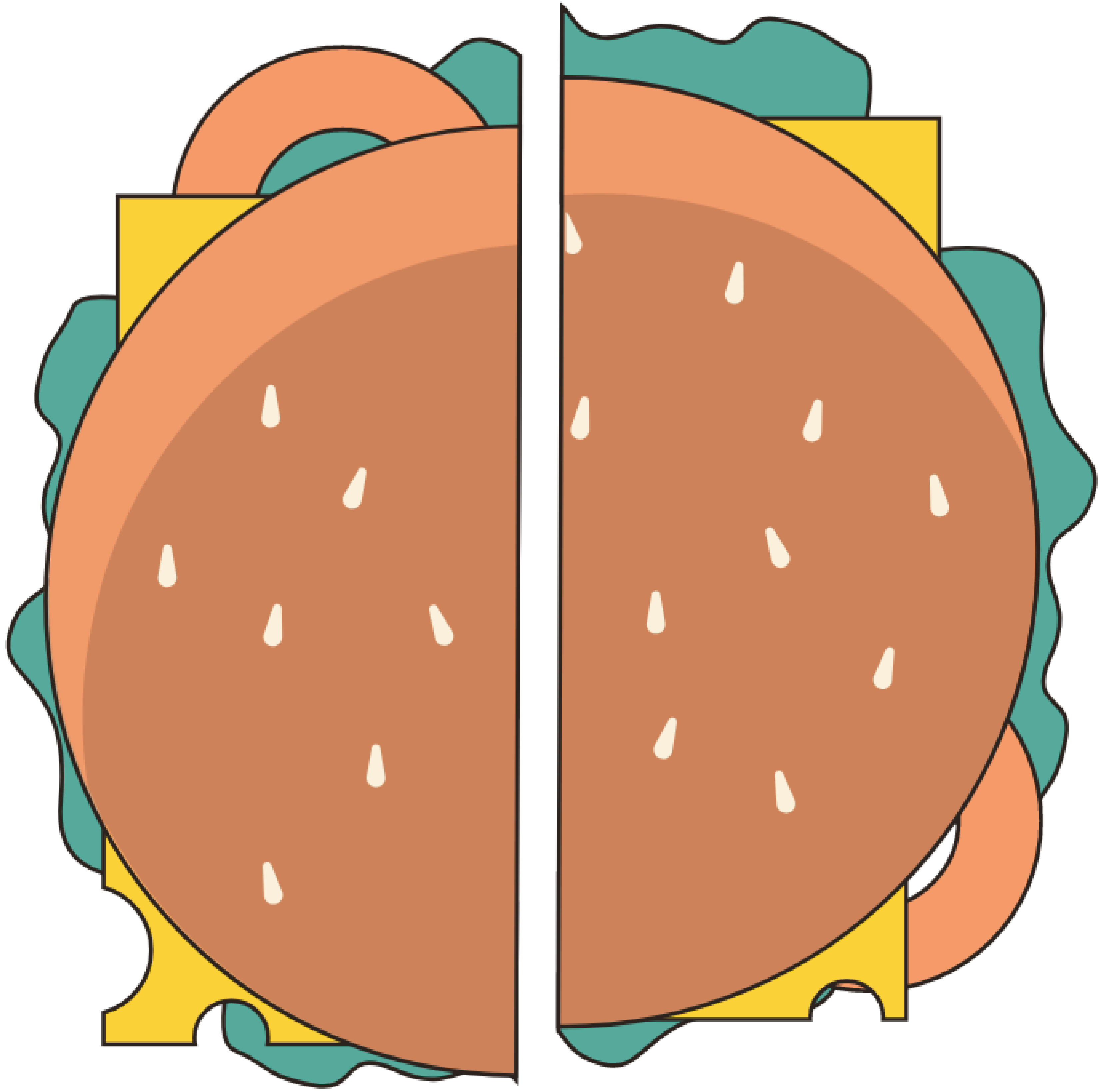
List the top 5 most ordered pizza types along with their quantities.

ANSWER

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



pizza_type	total_ordered
The Pepperoni Pizza	2207
The Barbecue Chicken Pizza	2198
The Classic Deluxe Pizza	2140
The Hawaiian Pizza	2102
The California Chicken Pizza	2099

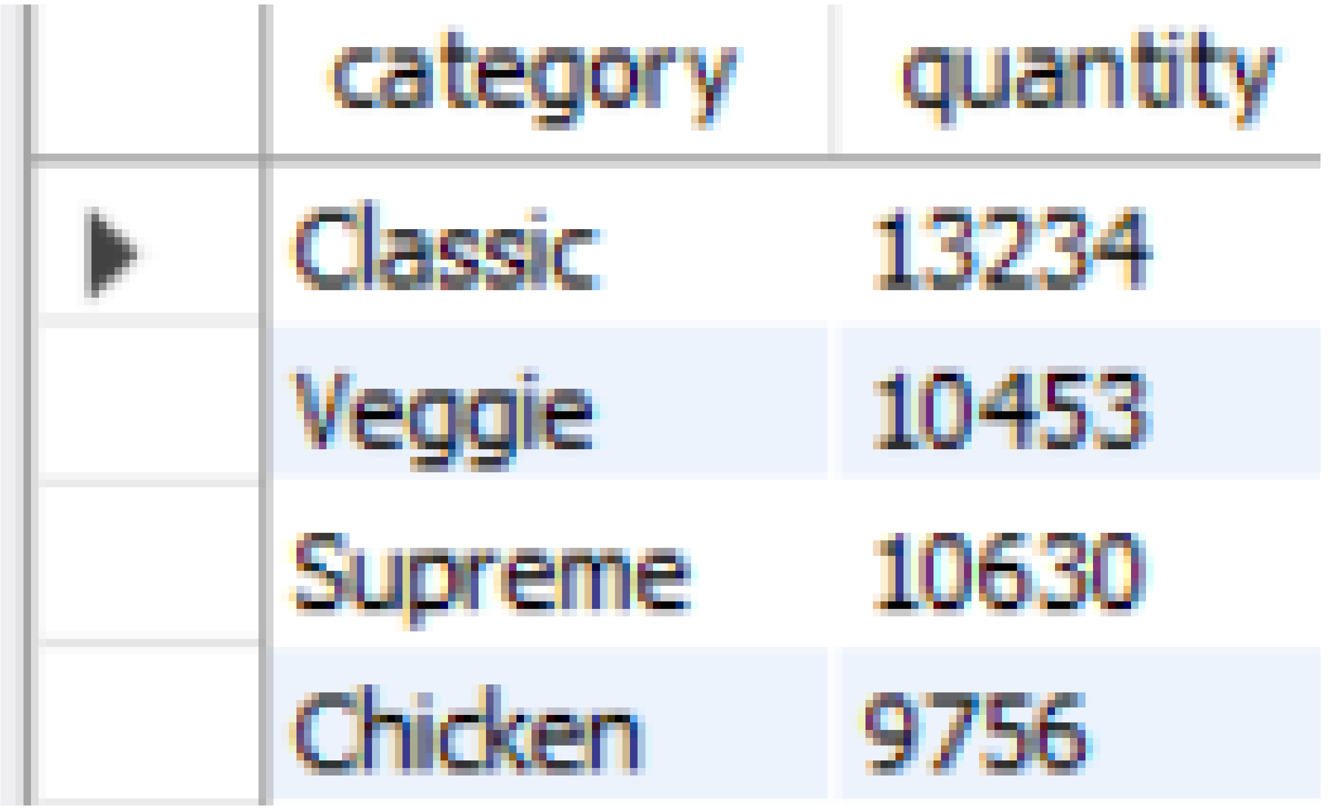


6

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

ANSWER

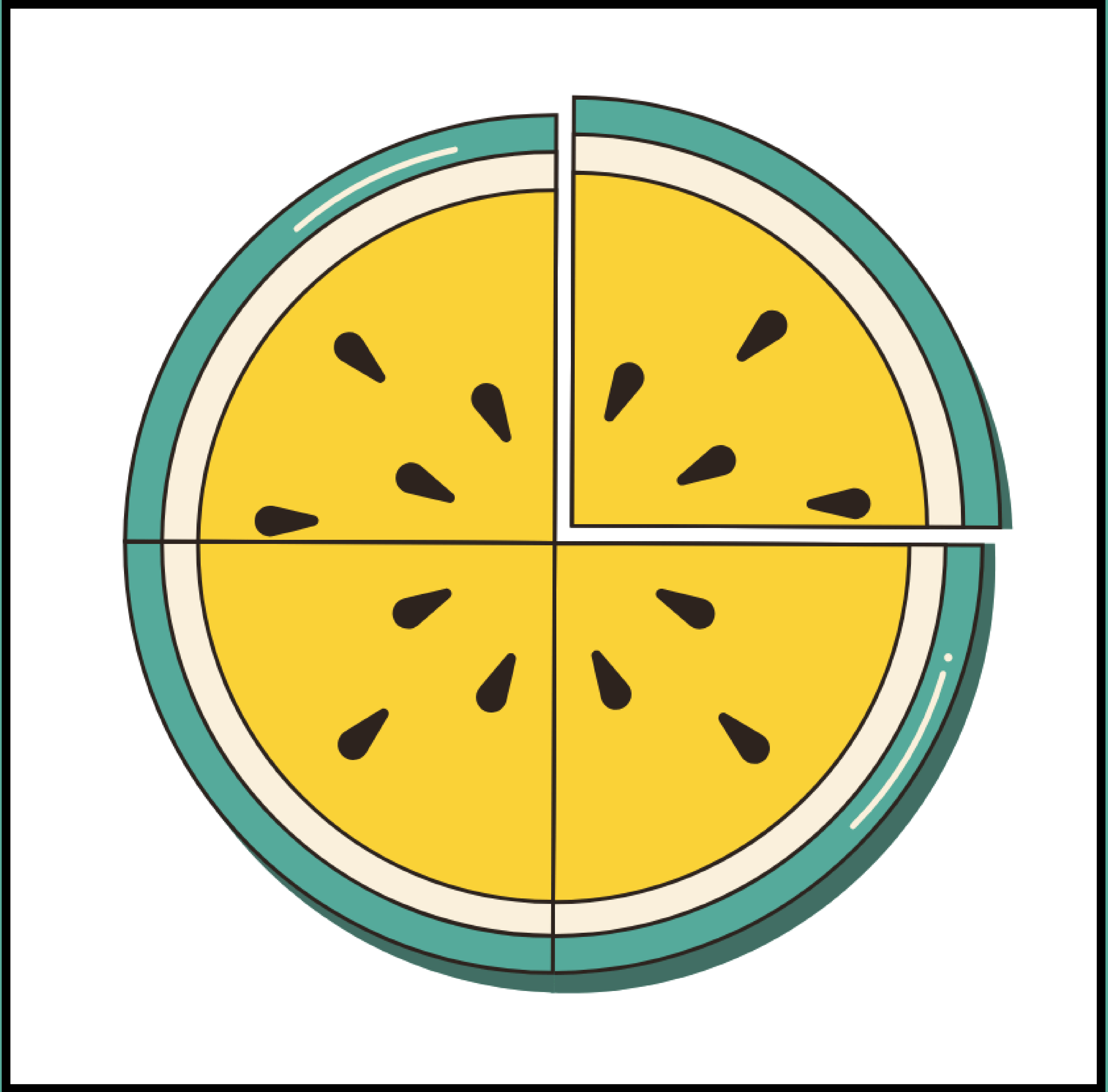
```
SELECT pt.category,  sum(od.quantity) as total_quantity
FROM pizza.order_details AS od  LEFT JOIN pizza.pizzas as p
ON od.pizza_id = p.pizza_id LEFT JOIN pizza.pizza_types AS pt  ON
p.pizza_type_id = pt.pizza_type_id group by pt.category
order by total_quantity desc
```



	category	quantity
▶	Classic	13234
	Veggie	10453
	Supreme	10630
	Chicken	9756

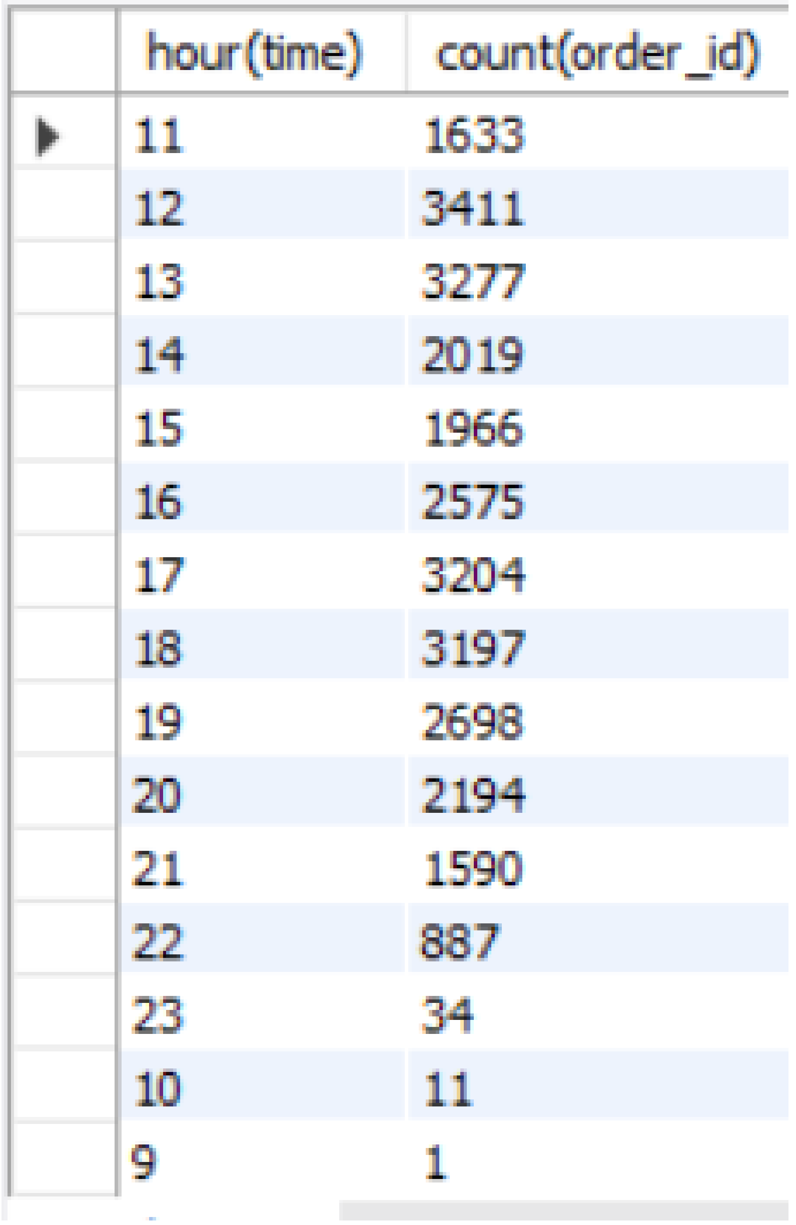
7

Determine the distribution of orders by hour of the day.

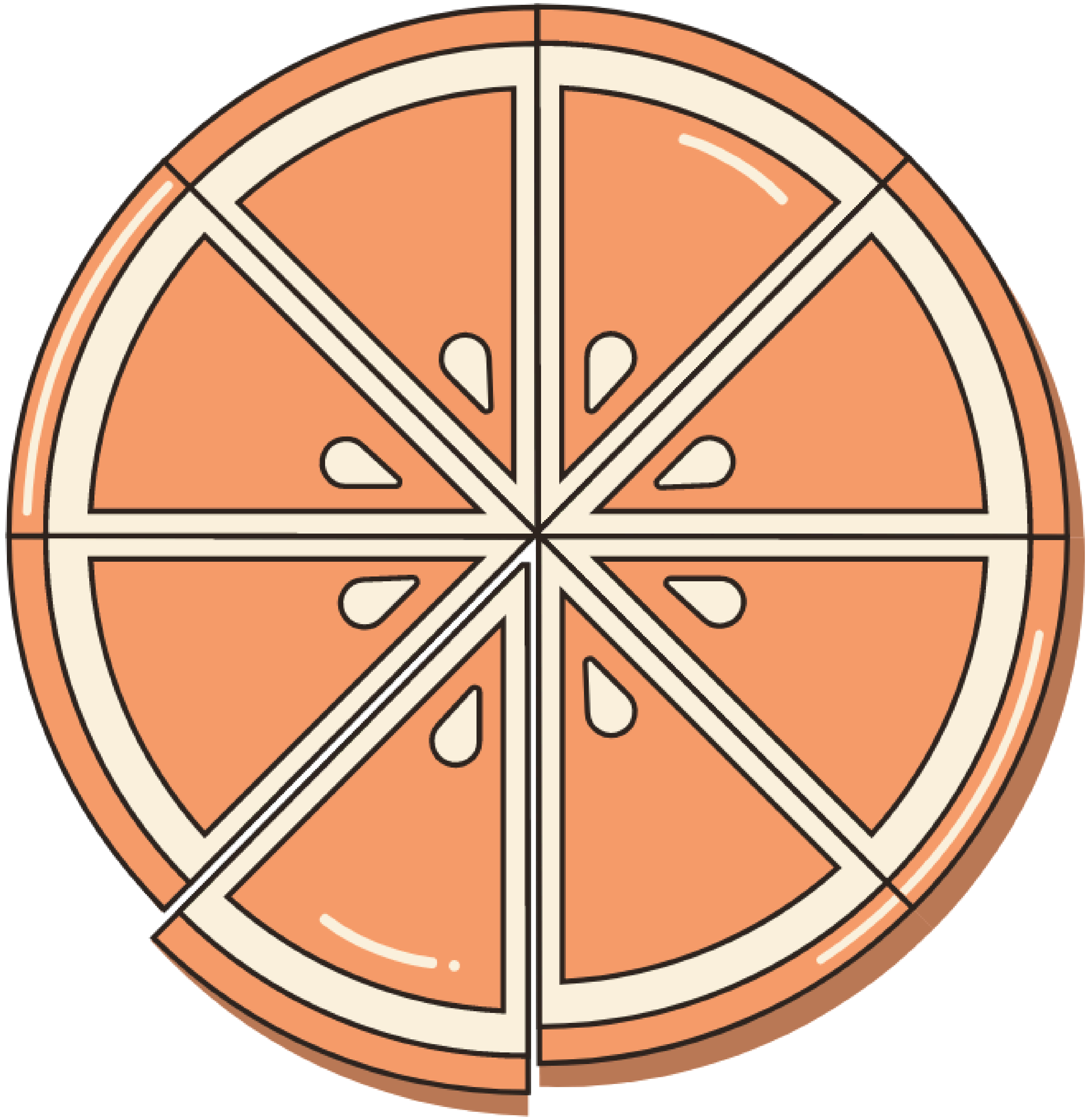


ANSWER

```
SELECT HOUR(TIME), COUNT(ORDER_ID) FROM
ORDERS    GROUP BY HOUR(TIME);
```



hour(time)	count(order_id)
11	1633
12	3411
13	3277
14	2019
15	1966
16	2575
17	3204
18	3197
19	2698
20	2194
21	1590
22	887
23	34
10	11
9	1

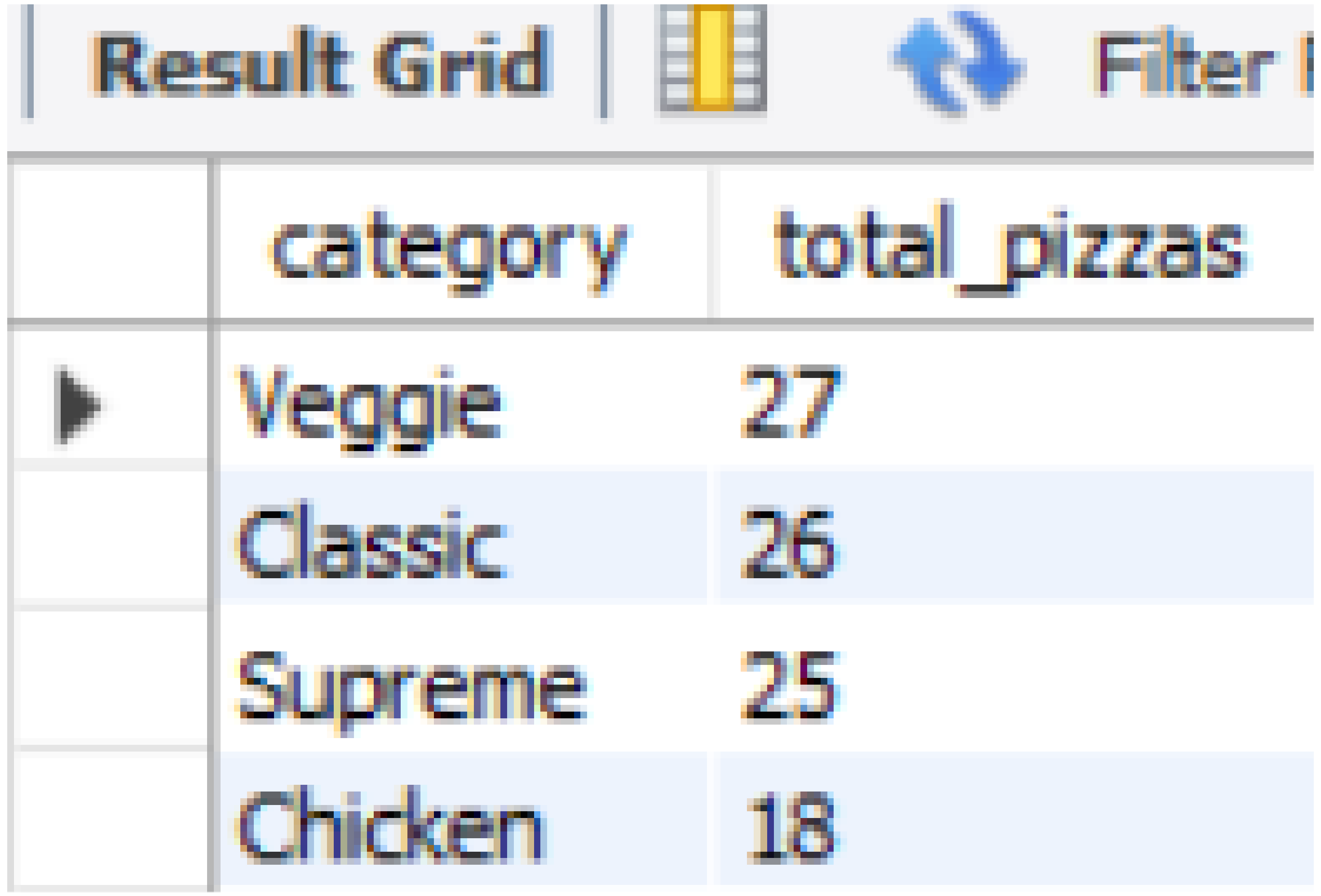


8

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

ANSWER

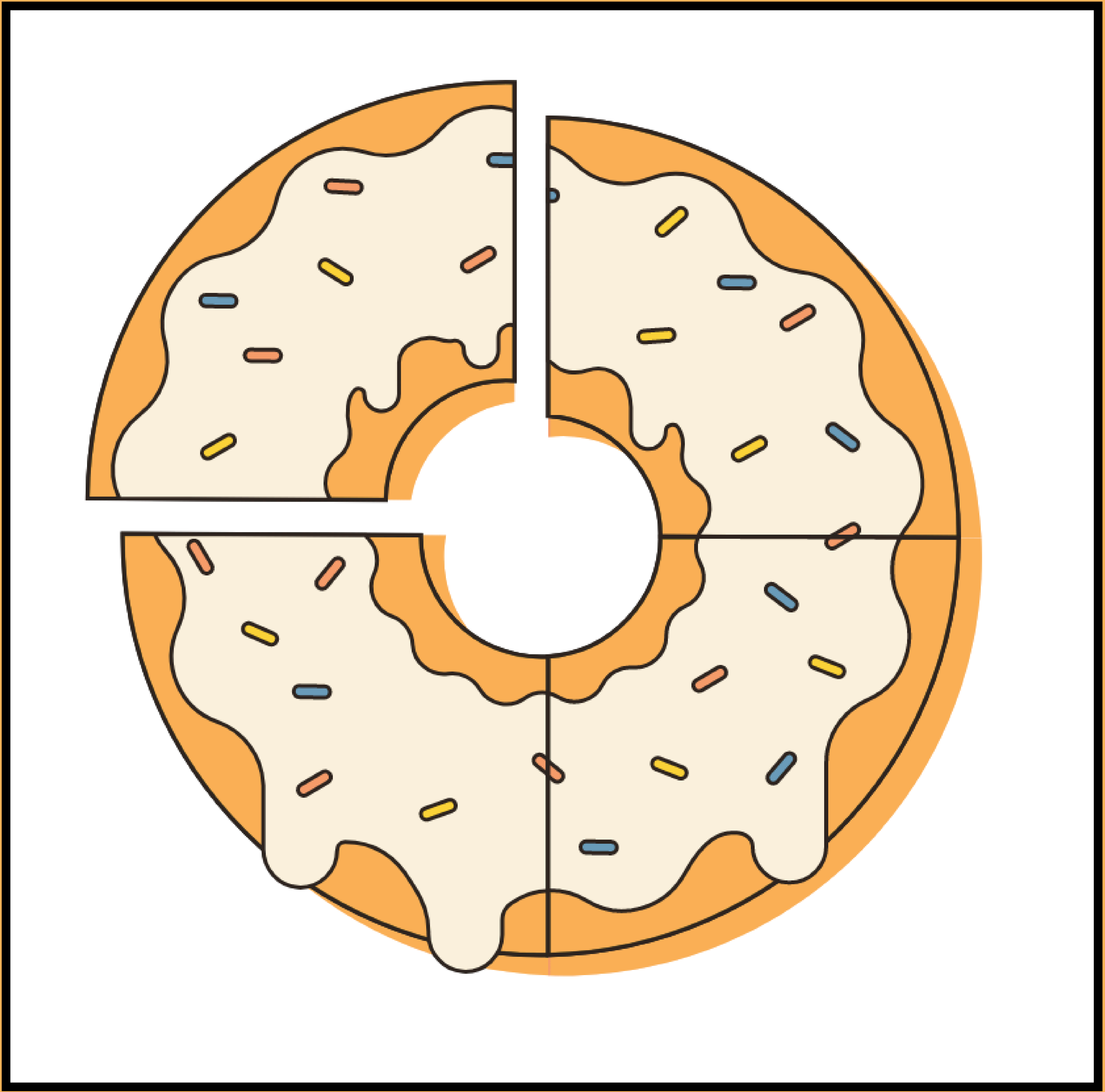
```
SELECT  pt.category, COUNT(p.pizza_id) AS total_pizzas
FROM   pizzas p JOIN   pizza_types pt ON p.pizza_type_id =
pt.pizza_type_id GROUP BY pt.category  ORDER BY
total_pizzas DESC;
```



category	total_pizzas
Veggie	27
Classic	26
Supreme	25
Chicken	18

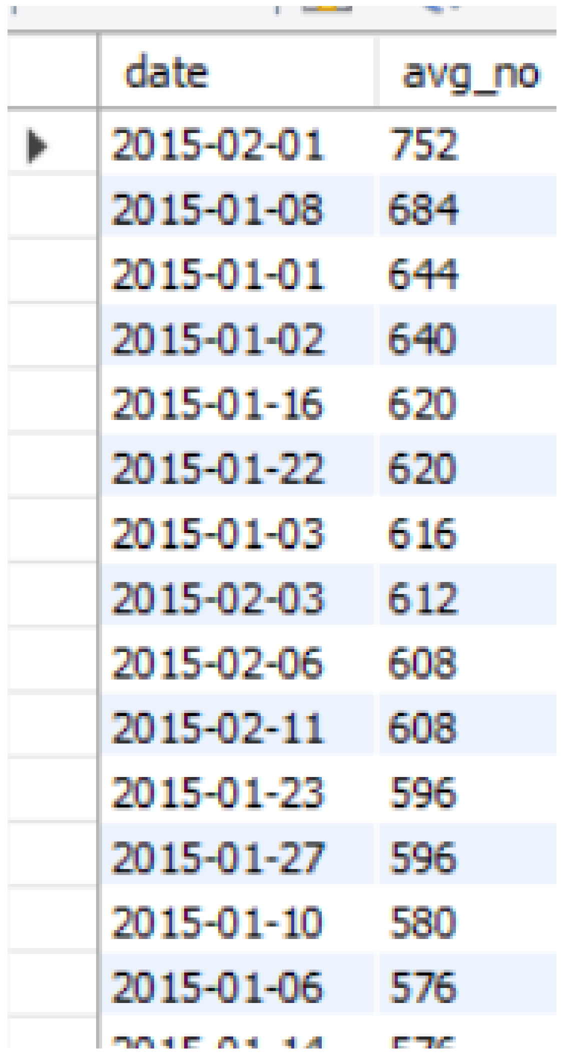
9

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

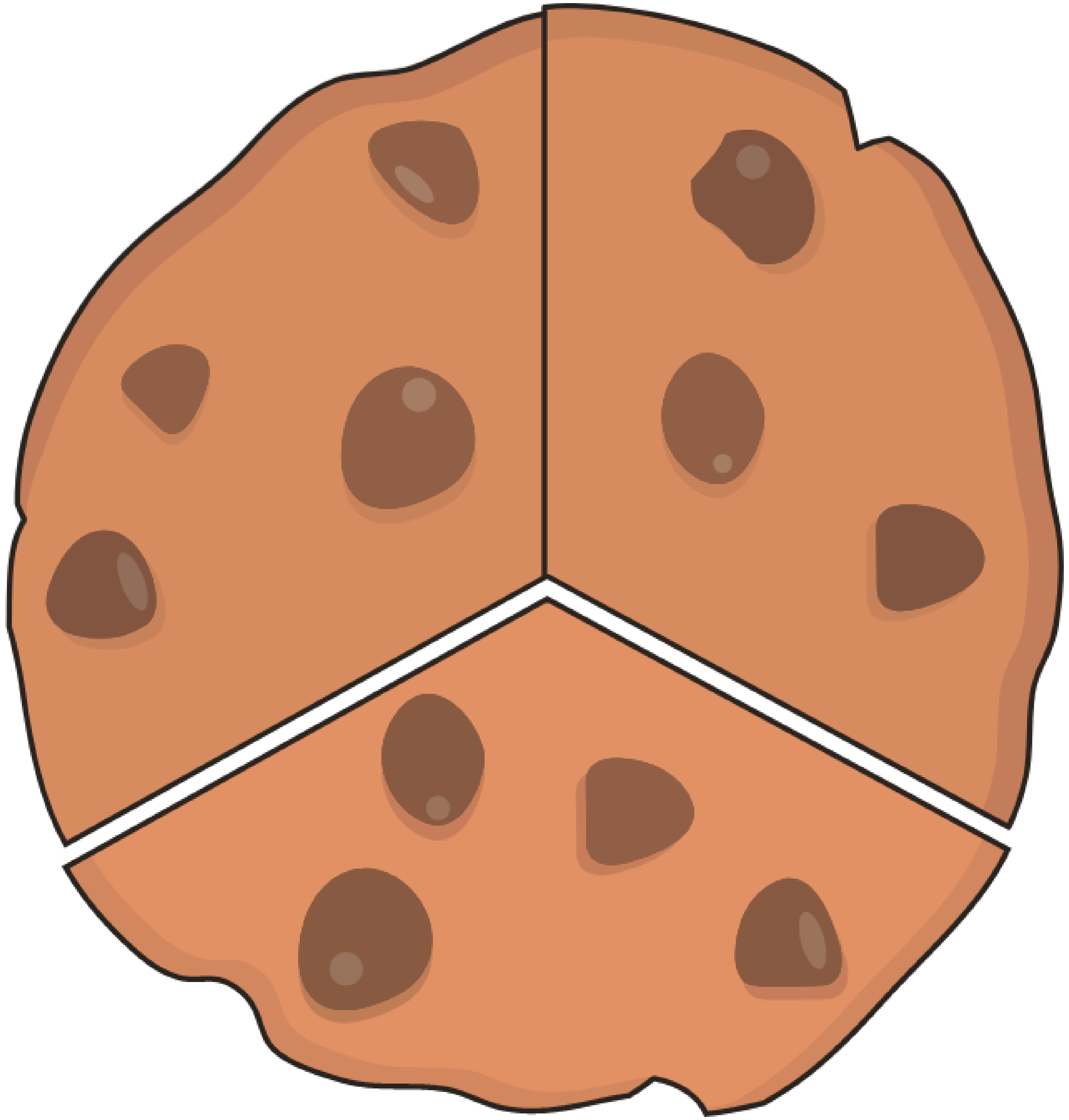


ANSWER

```
select  o.date, count(date) as avg_no from  pizza.order_details
od left join pizza.orders as oon od.order_id = o.order_id
group by date  order by avg_no desc
```



date	avg_no
2015-02-01	752
2015-01-08	684
2015-01-01	644
2015-01-02	640
2015-01-16	620
2015-01-22	620
2015-01-03	616
2015-02-03	612
2015-02-06	608
2015-02-11	608
2015-01-23	596
2015-01-27	596
2015-01-10	580
2015-01-06	576
2015-01-14	576

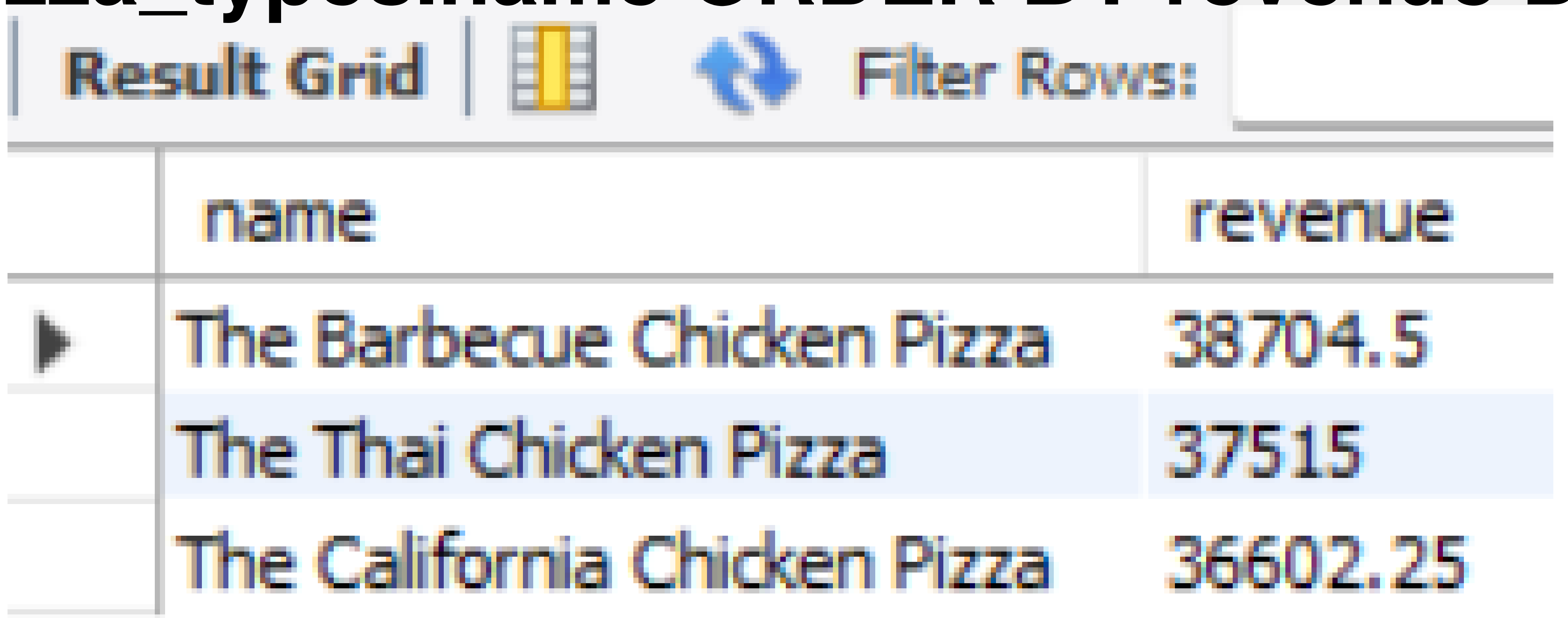


13

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

ANSWER

```
SELECT  pizza_types.name,  SUM(order_details.quantity *
pizzas.price) AS revenue 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 revenue DESC LIMIT 3;
```



name	revenue
The Barbecue Chicken Pizza	38704.5
The Thai Chicken Pizza	37515
The California Chicken Pizza	36602.25