

GOLDEN CRUST PIZZA CO. SALES REPORT



INTRODUCTION

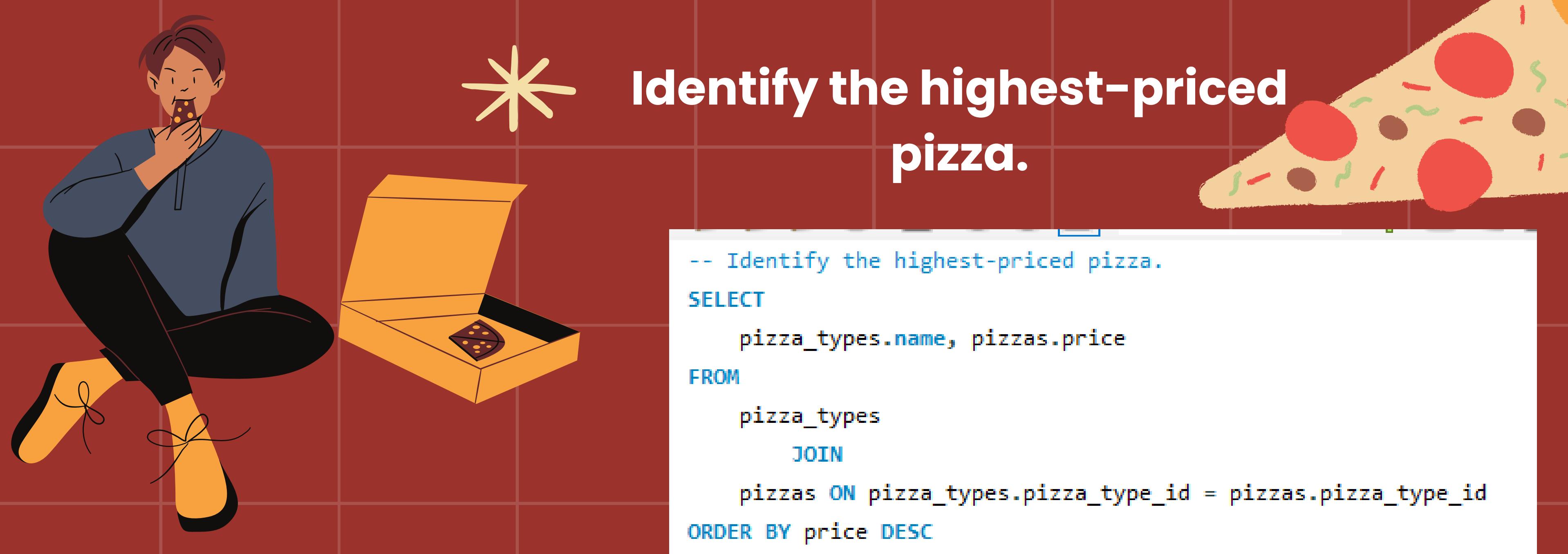
Pizza is a dish that originates from Italy and is one of the favorite foods of many people in various parts of the world.

Let's start our adventure in the world of pizza!

A cartoon illustration of two people eating pizza. On the left, a woman with dark hair and glasses, wearing a yellow jacket, holds a slice of pepperoni pizza. On the right, a man with a green baseball cap and a striped shirt also eats a slice of pepperoni pizza. They are set against a red background with yellow starburst shapes.

OVERVIEW

Presenting the analysis of
Golden Crust Pizza Co. Q2 2024
sales data using SQL



Identify the highest-priced pizza.

```
-- Identify the highest-priced pizza.  
SELECT  
    pizza_types.name, pizzas.price  
FROM  
    pizza_types  
        JOIN  
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id  
ORDER BY price DESC  
LIMIT 1;
```

Result Grid | Filter Rows:

	name	price
▶	The Greek Pizza	35.95

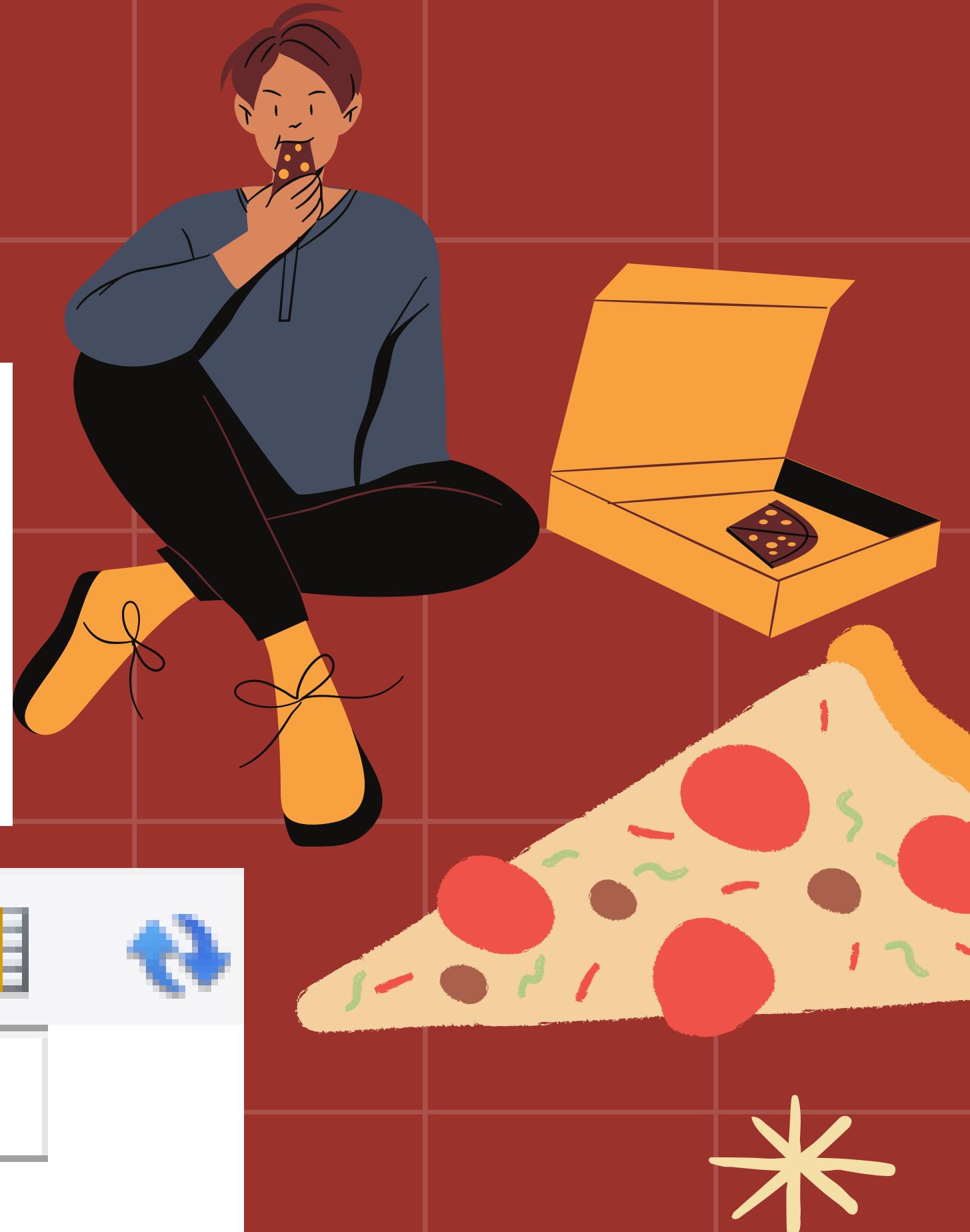


Calculate the total revenue generated from pizza sales.

```
-- Calculate the total revenue generated from pizza sales.
```

```
SELECT  
    round(sum(order_details.quantity * pizzas.price),2) AS total_sales  
FROM  
    order_details  
    JOIN  
    pizzas ON pizzas.pizza_id = order_details.pizza_id;
```

	total_sales
▶	817860.05

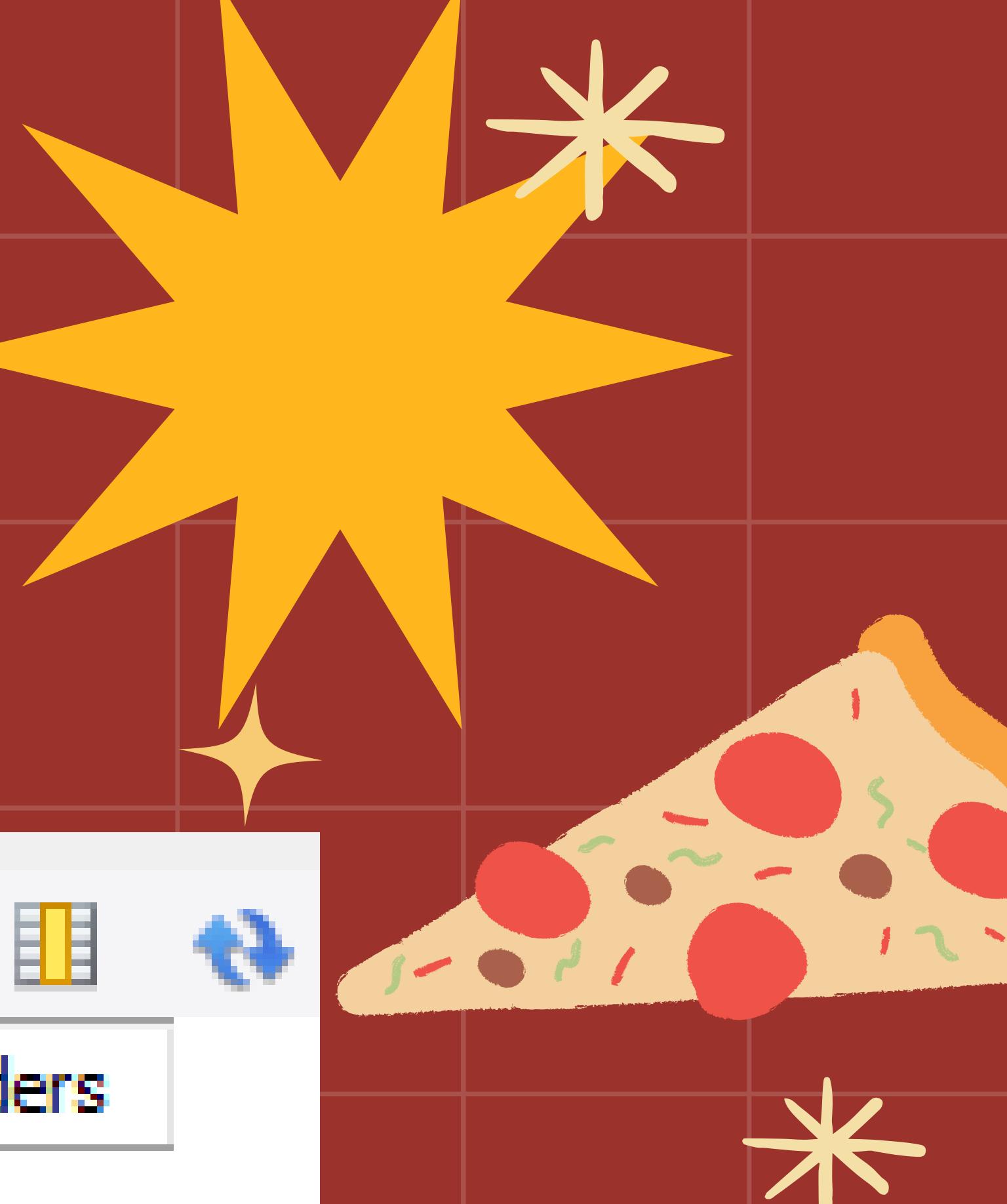


Retrieve the total number of orders placed

```
-- Retrieve the total number of orders placed  
SELECT  
    COUNT(order_id) AS Total_orders  
FROM  
    orders;
```



Result Grid	
	Total_orders
▶	21350



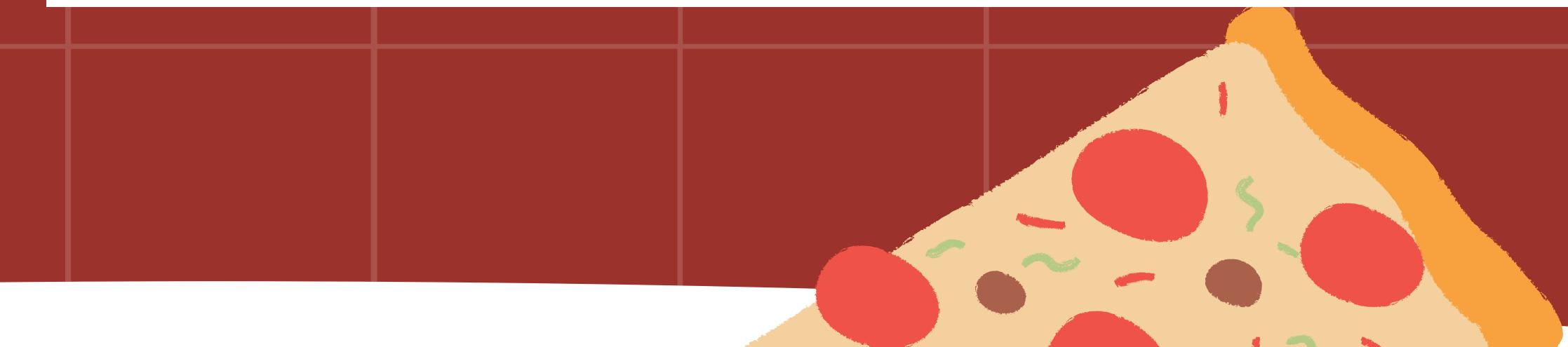


List the top 5 most ordered pizza types along with their quantities

```
-- List the top 5 most ordered pizza types along with their quantities.  
SELECT  
    pizza_types.name,  
    SUM(order_details.quantity) AS Total_Quantities  
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_Quantities DESC  
LIMIT 5;
```

Result Grid | Filter Rows:

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



Identify the most common pizza size ordered

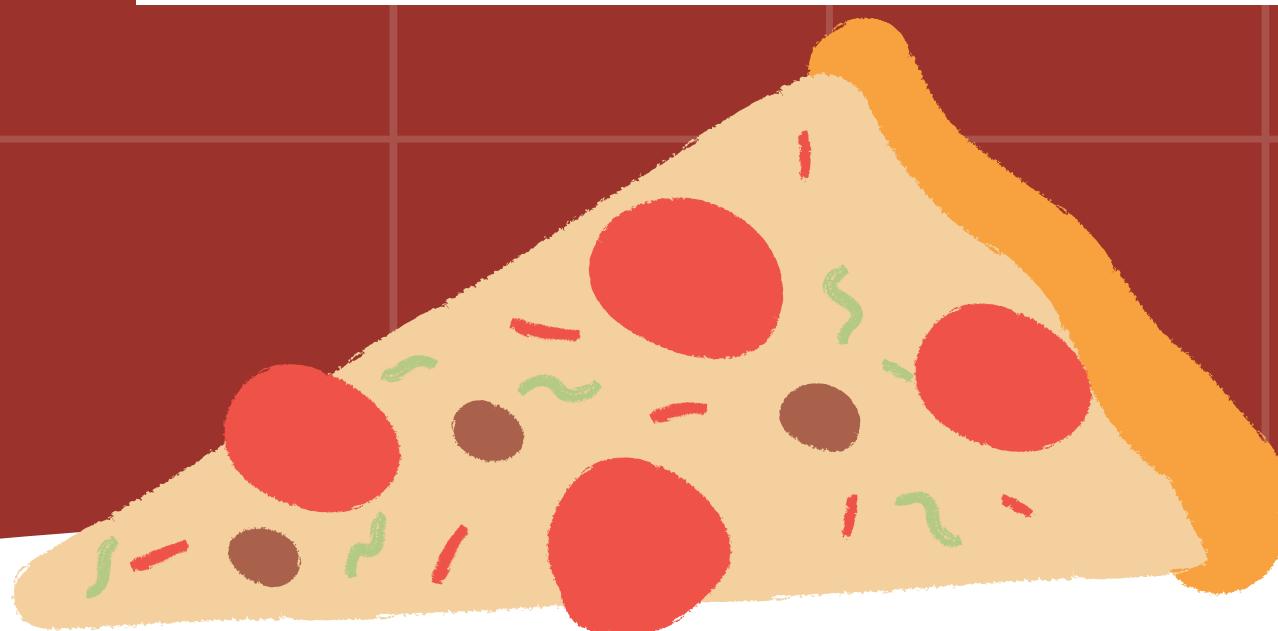
-- Identify the most common pizza size ordered.

```
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;
```



Result Grid | Filter Rows:

	size	order_count
▶	L	18526



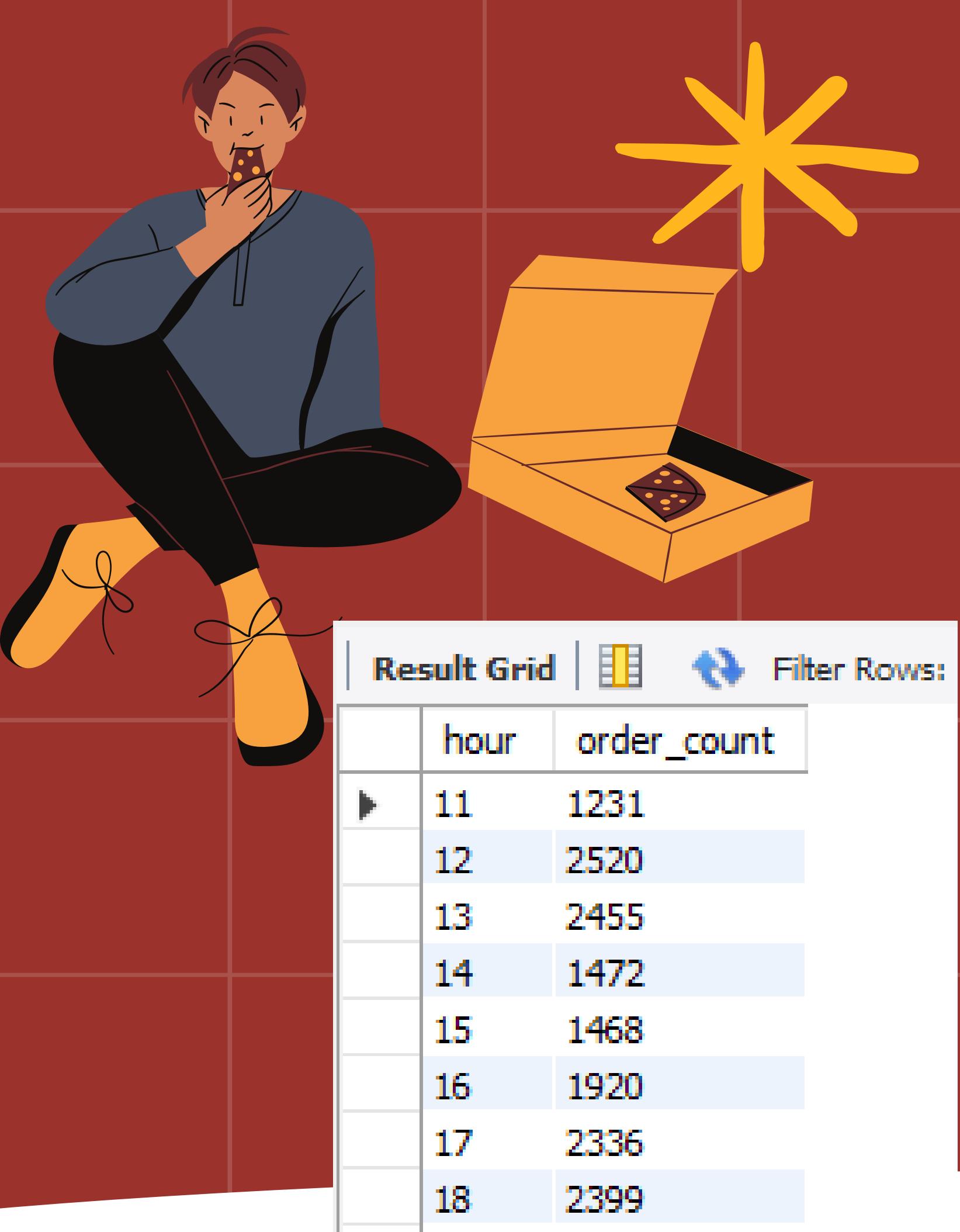
Join the necessary tables to find the total quantity of each pizza category ordered

```
-- Join the necessary tables to find the total quantity of each pizza category ordered
SELECT
    pizza_types.category,
    SUM(order_details.quantity) AS Total_Quantities
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.category
ORDER BY Total_Quantities DESC;
```



Result Grid | Filter Rows:

	category	Total_Quantities
▶	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) AS hour, COUNT(order_id) AS order_count  
FROM  
    orders  
GROUP BY HOUR(order_time);
```

Result Grid | Filter Rows:

	hour	order_count
▶	11	1231
	12	2520
	13	2455
	14	1472
	15	1468
	16	1920
	17	2336
	18	2399



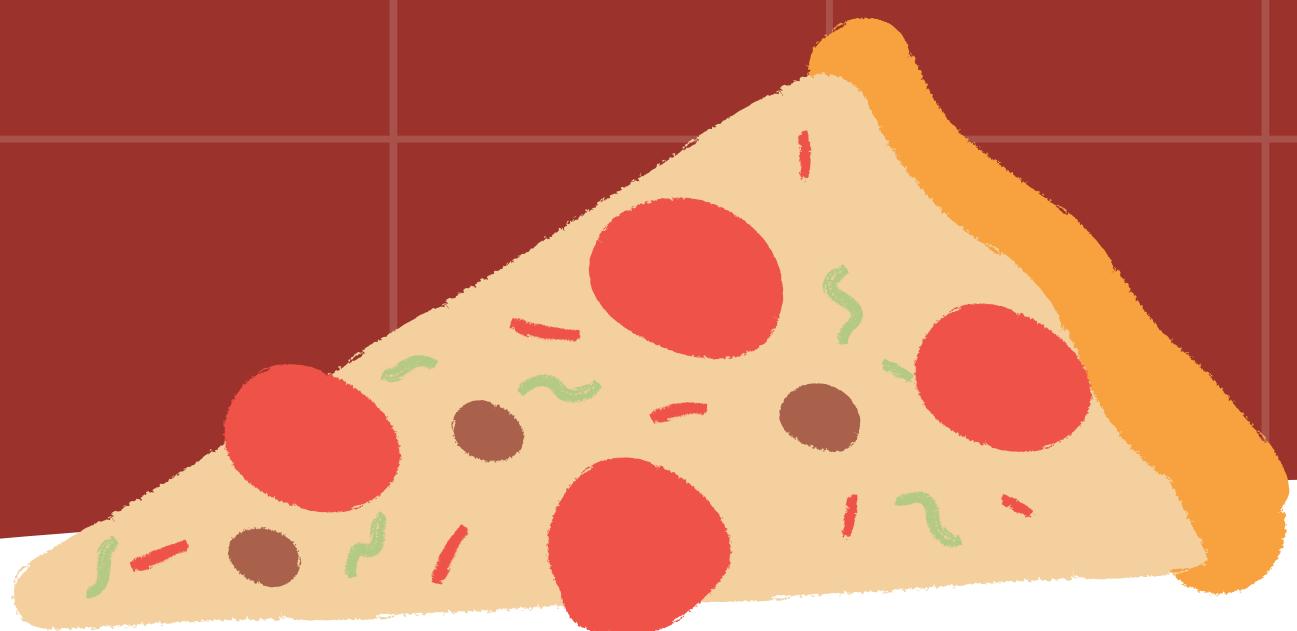
Join relevant tables to find the category-wise distribution of pizzas.

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

```
SELECT  
    category, COUNT(name) AS Total_count  
FROM  
    pizza_types  
GROUP BY category  
ORDER BY total_count DESC;
```

Result Grid | Filter Rows:

	category	Total_count
▶	Supreme	9
	Veggie	9
	Classic	8
	Chicken	6



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

```
-- Group the orders by date and calculate the average number of pizzas ordered per day.  
SELECT  
    ROUND(AVG(total_quantity), 0) AS Average_pizza_per_day  
FROM  
(SELECT  
    orders.order_date AS Dates,  
    SUM(order_details.quantity) AS total_quantity  
FROM  
    orders  
JOIN order_details ON orders.order_id = order_details.order_id  
GROUP BY Dates) AS order_quantity;
```

Result Grid	
	Average_pizza_per_day
▶	138



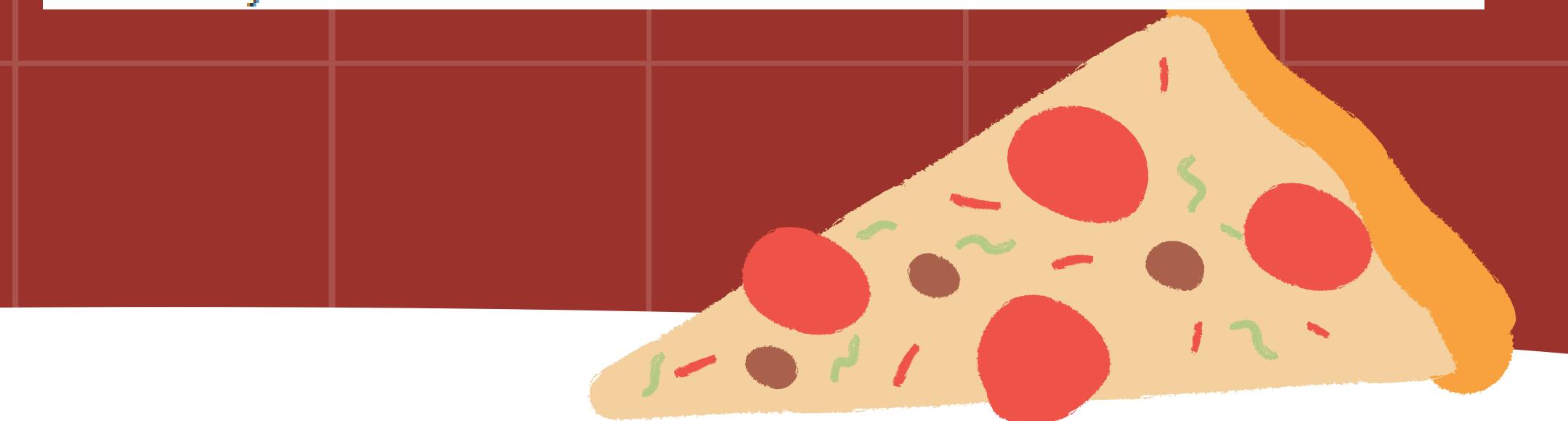


Determine the top 3 most ordered pizza types based on revenue

```
-- 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 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;
```

Result Grid | Filter Rows:

	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

```
-- Calculate the percentage contribution of each pizza type to total revenue.  
  
SELECT  
    pizza_types.category,  
    ROUND(SUM(order_details.quantity * pizzas.price) / (SELECT  
        ROUND(SUM(order_details.quantity * pizzas.price), 2)  
    FROM  
        order_details  
    JOIN  
        pizzas ON pizzas.pizza_id = order_details.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_types.category  
ORDER BY revenue DESC;
```



Result Grid | Filter Rows:

	category	revenue
▶	Classic	26.91
	Supreme	25.46
	Chicken	23.96
	Veggie	23.68



Analyze the cumulative revenue generated over time.

```
-- 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(order_details.quantity * pizzas.price) as revenue  
      from order_details join pizzas  
        on order_details.pizza_id = pizzas.pizza_id  
     join orders  
       on orders.order_id = order_details.order_id  
    group by orders.order_date) as sales;
```



Result Grid | Filter Rows:

	order_date	cum_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

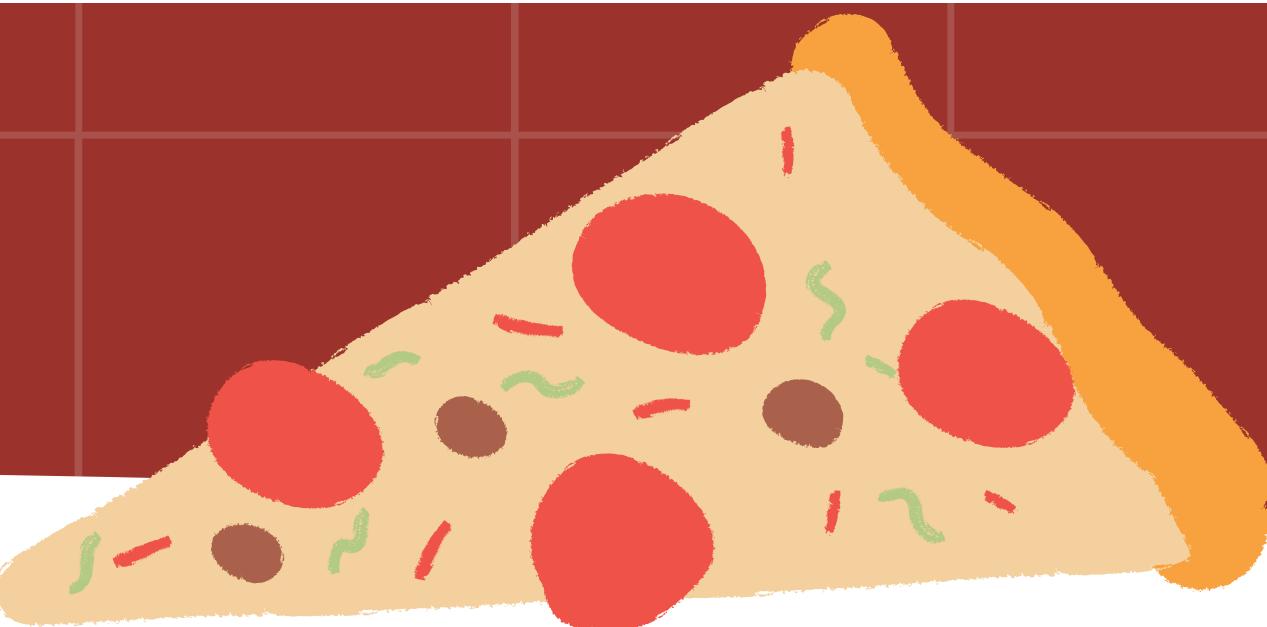


Result Grid | Filter Rows: Export:

	name	category	revenue
▶	The Thai Chicken Pizza	Chicken	43434.25
	The Barbecue Chicken Pizza	Chicken	42768
	The California Chicken Pizza	Chicken	41409.5
	The Classic Deluxe Pizza	Classic	38180.5
	The Hawaiian Pizza	Classic	32273.25
	The Pepperoni Pizza	Classic	30161.75
	The Spicy Italian Pizza	Supreme	34831.25
	The Italian Supreme Pizza	Supreme	33476.75
	The Sicilian Pizza	Supreme	30940.5
	The Four Cheese Pizza	Veggie	32265.70000000065
	The Mexicana Pizza	Veggie	26780.75
	The Five Cheese Pizza	Veggie	26066.5

Determine the top 3 most ordered pizza types based on revenue for each pizza category

```
-- Determine the top 3 most ordered pizza types based on revenue for each pizza category
select name, category, revenue from
(select category, name, revenue,
rank() over(partition by category order by revenue desc) as ranks
from
(select pizza_types.category, 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.category, pizza_types.name)as a) as b
where ranks<= 3;
```



VARIATIONS



Veggie Pizza



Supreme Pizza



Chicken Pizza



Classic Pizza



THANK YOU

“Have fun making your own pizza
and enjoy every bite”