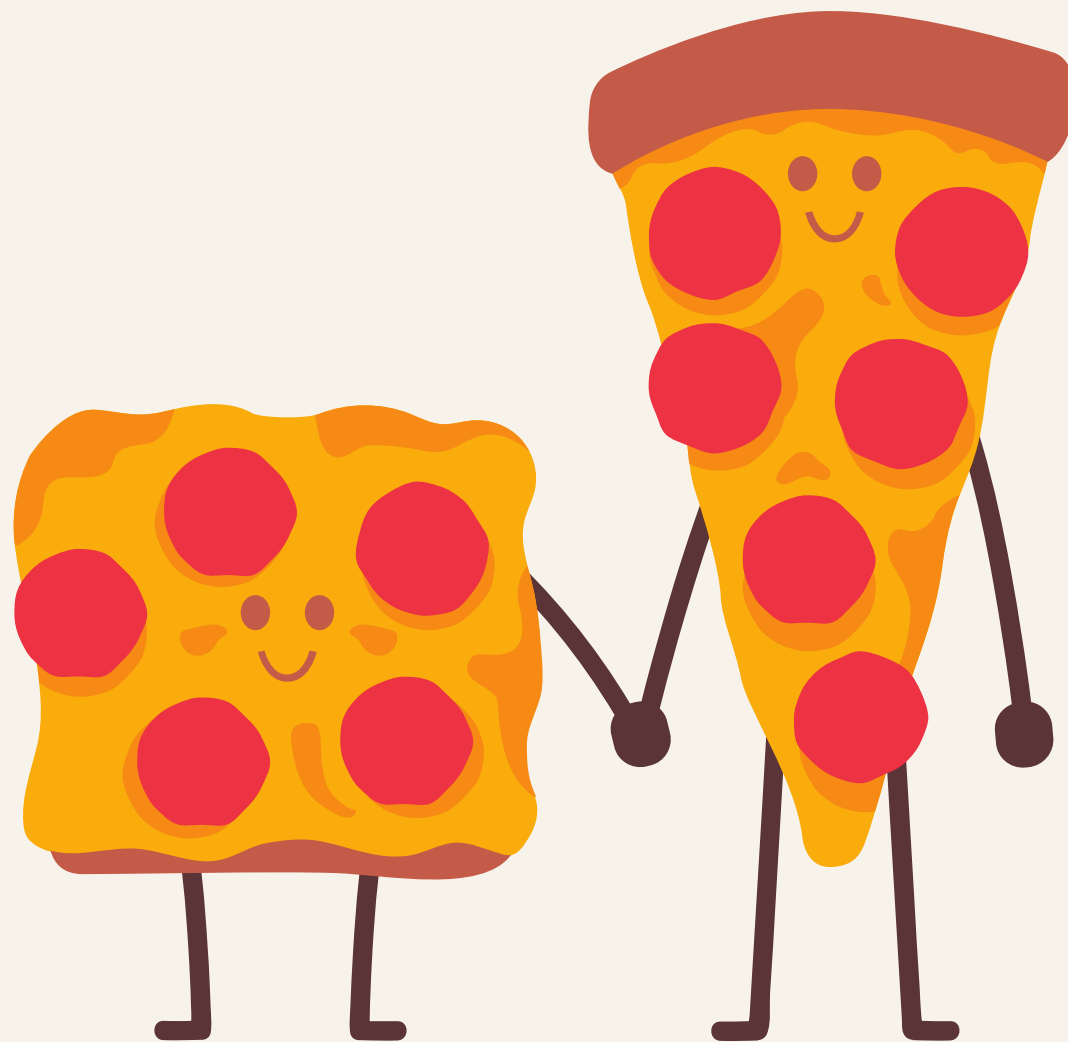# PIZZA SALES SQL PROJECT

ONLY
$25.99

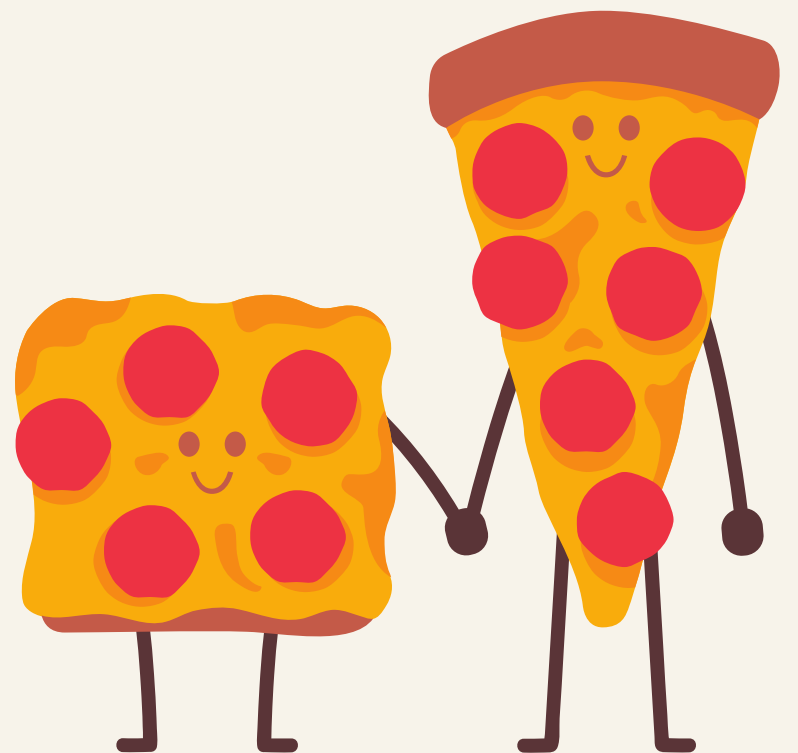+123-456-7890 | www.reallygreatsite.com ● ● ● **BUY NOW**

Hii, My Name is Ila Singh.
In this project I've utilised MySQL Query to solve pizza sales questions

Retrieve the total number of orders placed.

```sql
SELECT
    COUNT(order_id) as Total_orders
FROM
    orders;
```
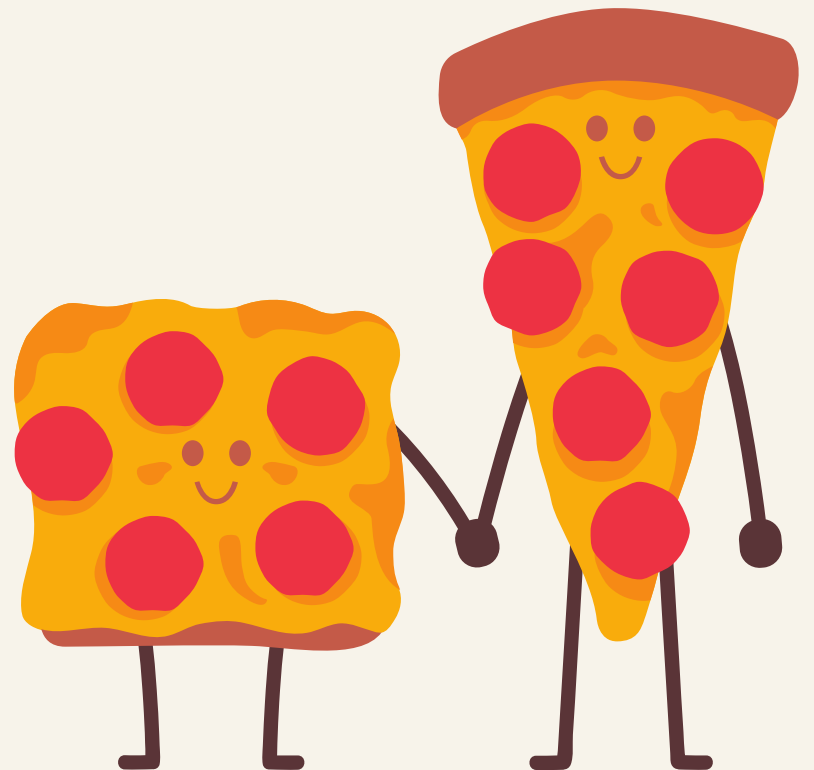
Result Grid

	Total_orders
▶	21350

Calculate the total revenue generated from pizza sales

```sql
SELECT
    sum(order_details.quantity * pizzas.price) as Total_Revenue
FROM
    order_details
        JOIN
    pizzas ON pizzas.pizza_id=order_details.pizza_id;
```
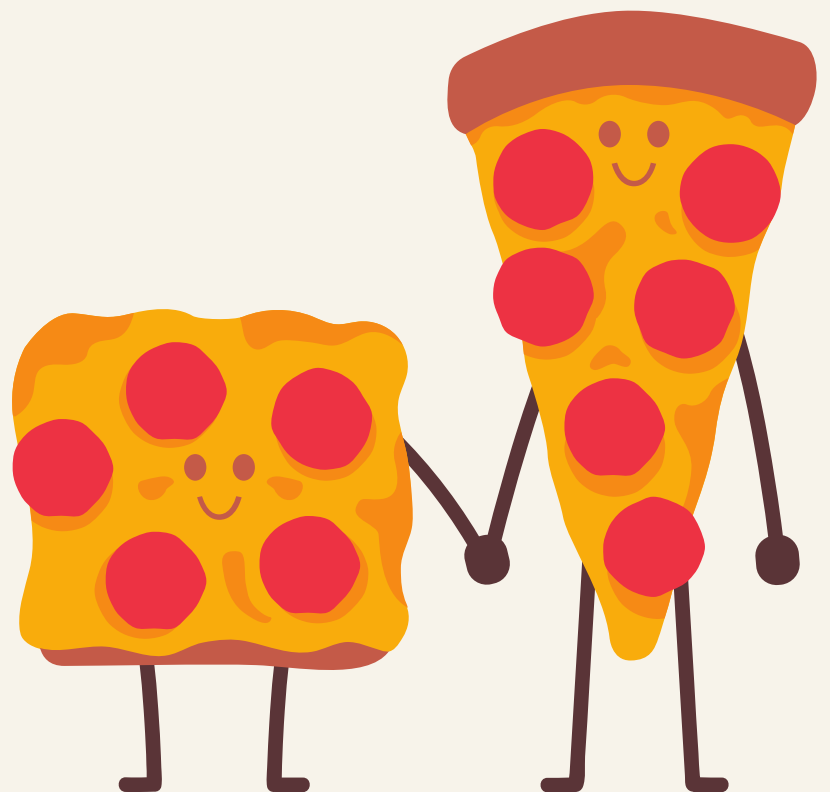
Result Grid

Total_Revenue
817860

Identify the highest-priced pizza.

```sql
SELECT
    name, price
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = Pizzas.pizza_type_id
ORDER BY pizzas.price DESC
LIMIT 1;
```
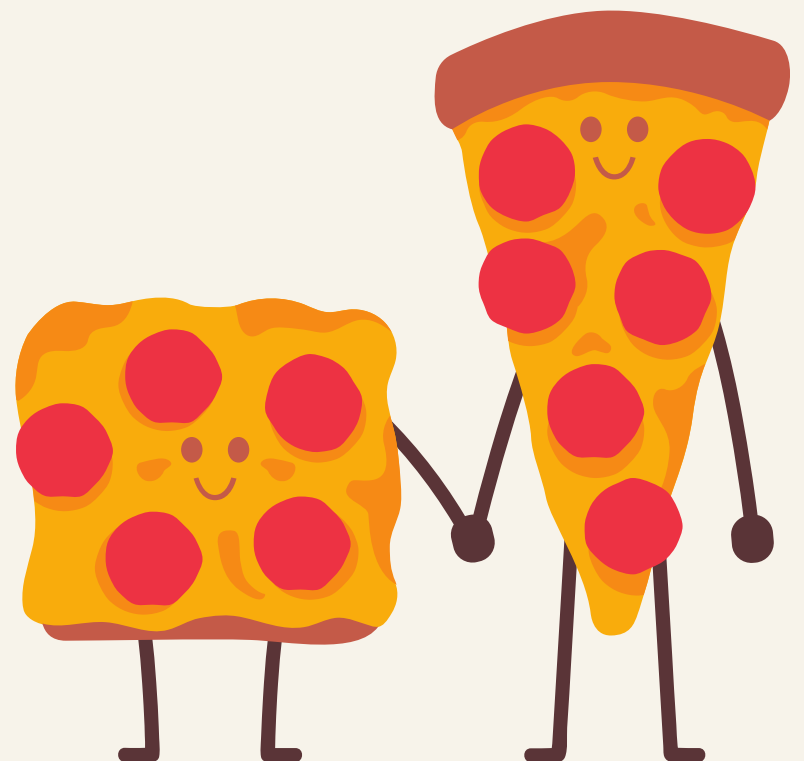
Result Grid | Filter Rows:

name	price
The Greek Pizza	35.95

Identify the most common pizza size ordered.

```sql
SELECT
    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;
```
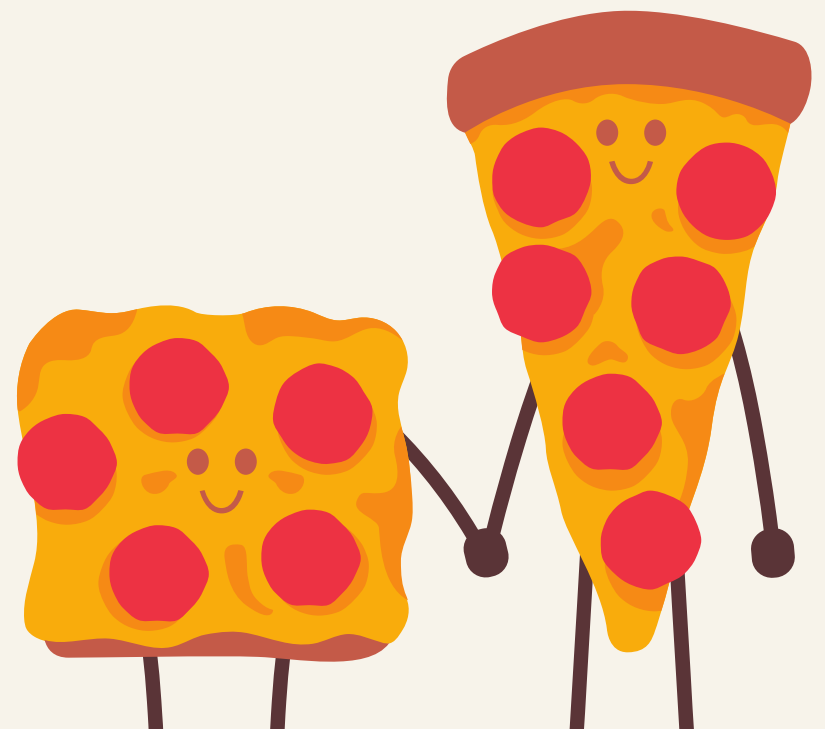
Result Grid

size	order_count
L	18526
M	15385
S	14137
XL	544
XXL	28

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

```sql
SELECT
    pizza_types.name, SUM(order_details.quantity) AS 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 quantity DESC
LIMIT 5;
```
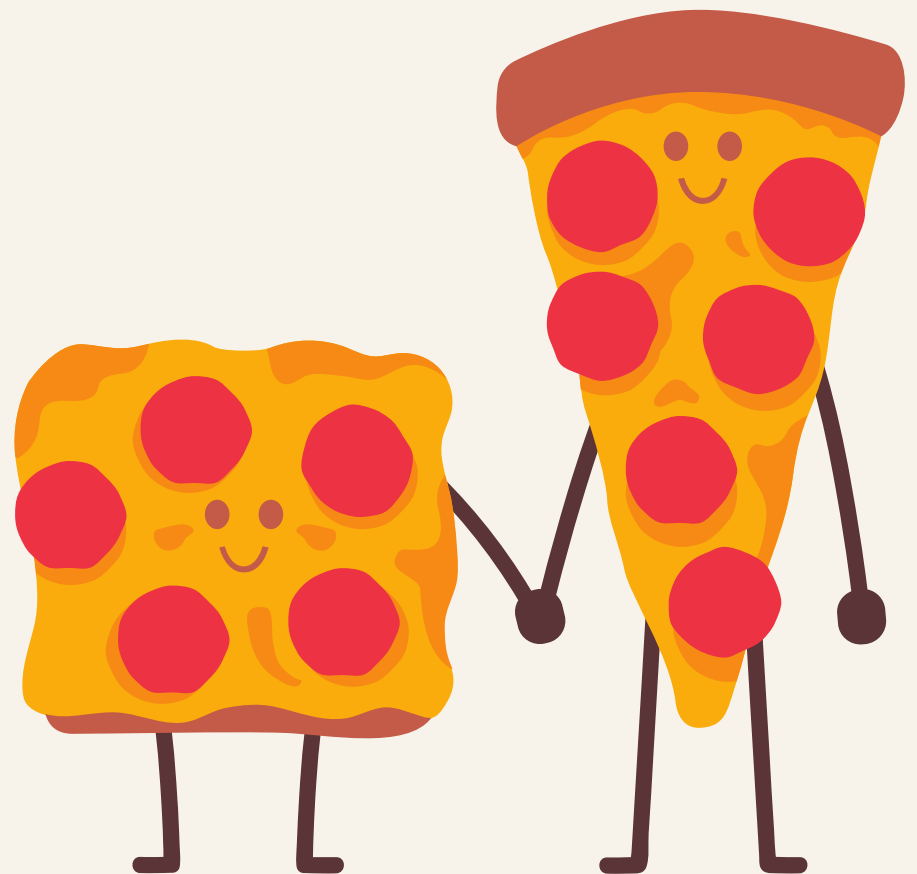
Result Grid | Filter Rows:

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.

```sql
SELECT
    pizza_types.category, SUM(order_details.quantity) as 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.category
ORDER BY quantity desc;
```

Result Grid | Filter

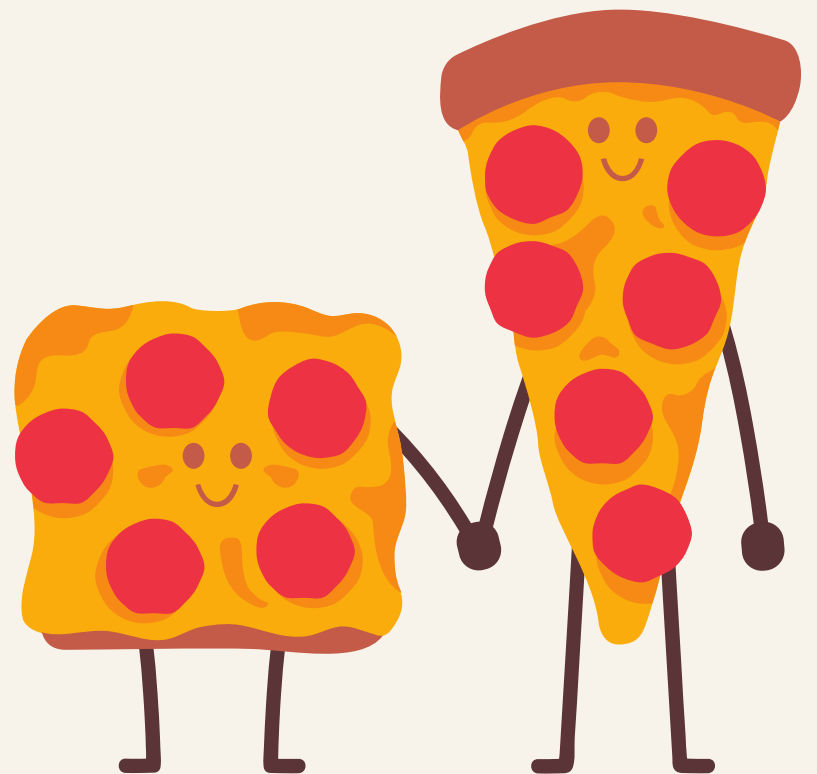| category	quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

Determine the distribution of orders by hour of the day.

```sql
SELECT
    HOUR(order_time) as hour, COUNT(order_id) as order_count
FROM
    orders
GROUP BY HOUR(order_time);
```
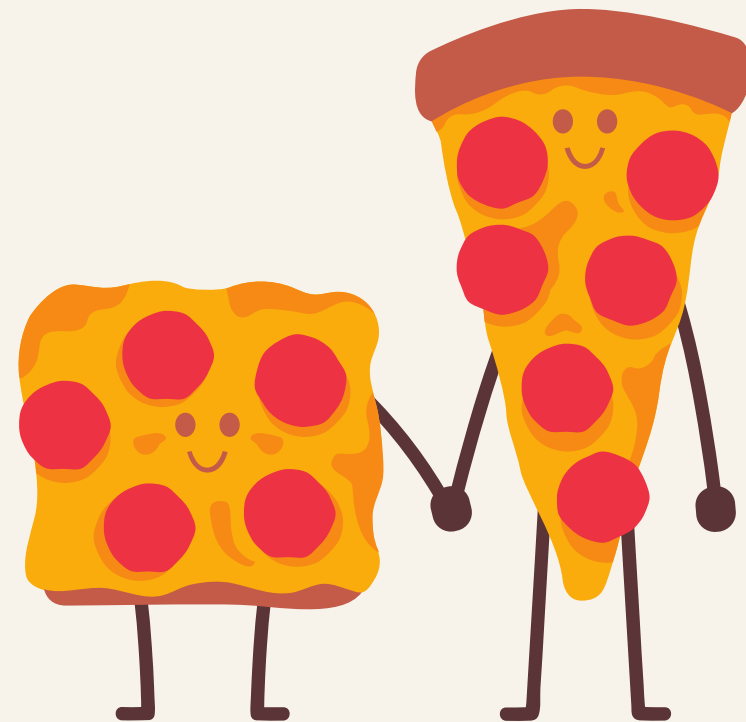
Result Grid | Filter Rows:

hour	order_count
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642

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

```sql
SELECT *
FROM
    pizza_types;


SELECT
    category, COUNT(name)
FROM
    pizza_types
GROUP BY pizza_types.category;
```
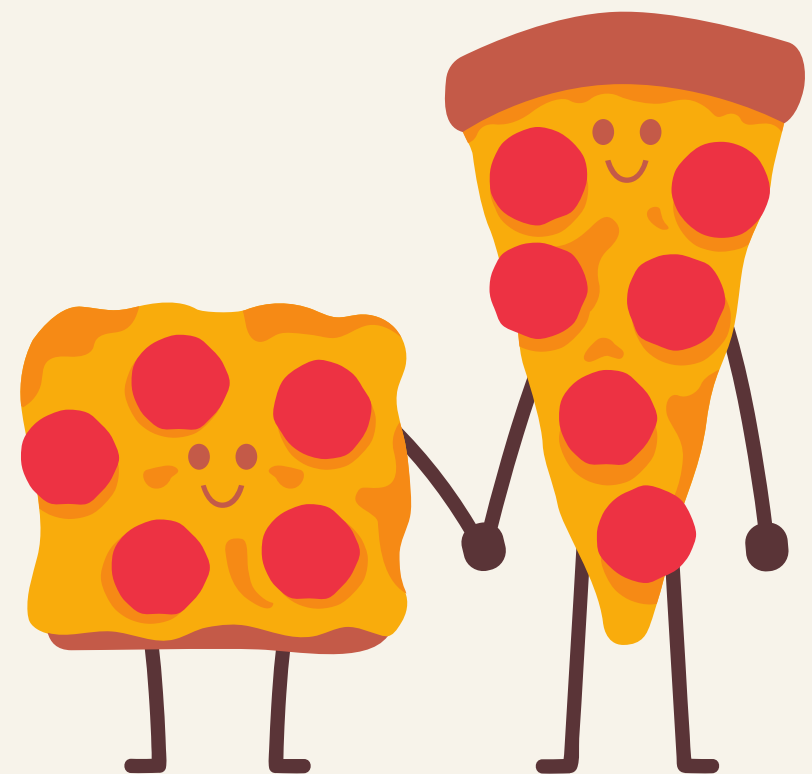
	category	count(name)
▶	Chicken	6
	Classic	8
	Supreme	9
	Veggie	9

Result Grid | ▮ | ↻ Filter Row

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

```sql
SELECT
    ROUND(AVG(quantity), 0) as avg_pizza_ordered_per_day
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;
```

Result Grid | 🔢 | 🔄 Filter Rows:

avg_pizza_ordered_per_day
138

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

```sql
SELECT
    pizza_types.name,
    ROUND(SUM(order_details.quantity* pizzas.price),0) 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;
```

Result Grid | Filter Rows:

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