## Comprehensive Sales Analysis of Pizza Hut Orders

Retrieve the total number of orders placed.

```
SELECT

COUNT(order_id) AS total_orders

FROM

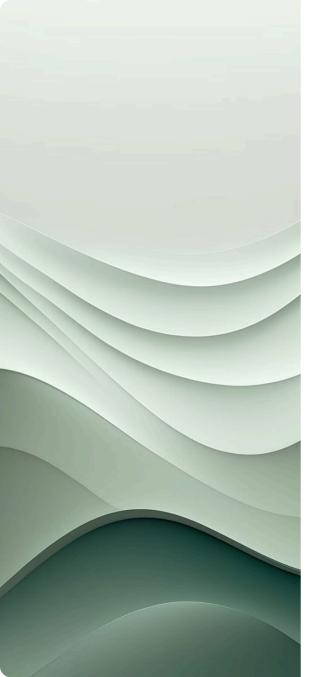
orders;

Result Grid | | | | | | | | |
```

total\_orders

21350





Calculate the total revenue generated from pizza sales.

Identify the top 3 highest-priced pizza.

R	esult Grid	Filter Rows:
	name	price
•	The Greek Pizza	35.95
	The Greek Pizza	25.5
	The Brie Carre Pizza	23.65





Identify the most common pizza size ordered.

	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.

```
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;
```







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

```
SELECT
     pizza types.category, SUM(order details.quantity) AS qty
 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 qty DESC;
Kesult Grid | H
   category
   Classic
           14888
   Supreme
         11987
           11649
   Chicken
          11050
```

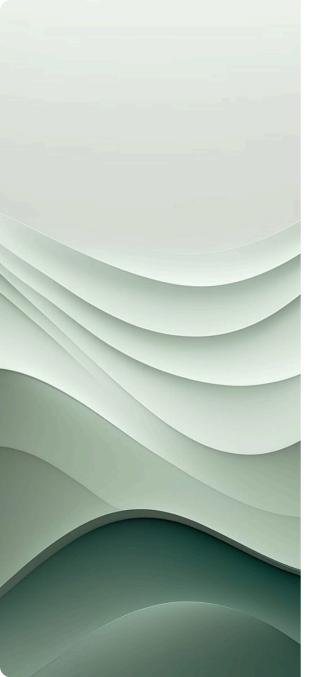


Determine the distribution of orders by hour of the day.

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

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





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



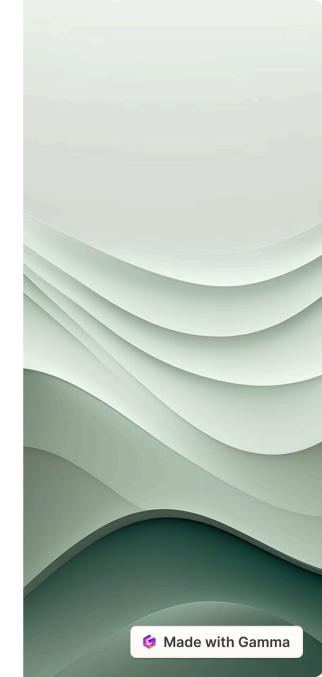
Group the orders by date and calculate the average number of pizzas ordered per day.

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

```
avg_pizza_order_per_day

138
```





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

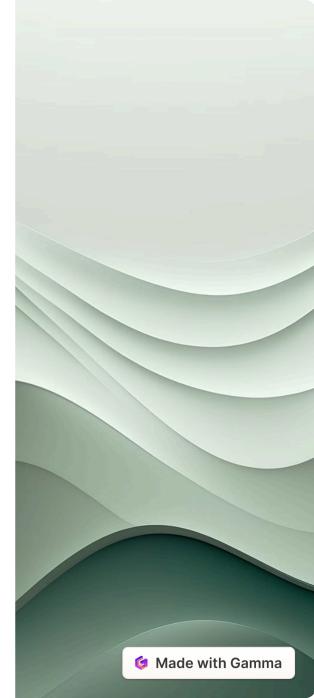
```
SELECT
    pizza_types.name,
    SUM(order_details.quantity * pizzas.price) AS revenue
FROM
    pizza_types
        JOIN
    pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY revenue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



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) A5 total revenue
                 FROM
                     order_details
                         JOIN
                     pizzas ON pizzas.pizza_id = order_details.pizza_id) * 100,
             2) 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
ORDER BY revenue DESC;
 category revenue
         26.91
  Classic
         25.46
   Supreme
         23.96
   Chicken
```

23.68





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;
```

	order_date	cum_revenue
•	2015-01-01	2713.85000000000004
	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

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

```
select name, revenue from
(select category, name, revenue,
rank() over(partition by category order by revenue desc) as rn
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 rn<=3;</pre>
```

	name	revenue
٠	The Thai Chicken Pizza	43434.25
	The Barbecue Chicken Pizza	42768
	The California Chicken Pizza	41409.5
	The Classic Deluxe Pizza	38180.5
	The Hawaiian Pizza	32273.25
	The Pepperoni Pizza	30161.75
	The Spicy Italian Pizza	34831.25
	The Italian Supreme Pizza	33476.75

