



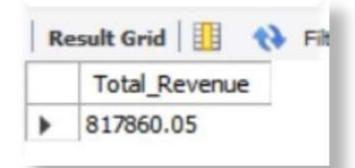
Retrieve the total number of orders placed.



Calculate total revenue generated from pizza sales.

Query:

 SELECT ROUND(SUM(order_details.quantity * pizzas.price),2) AS Total_Revenue FROM order_details
 JOIN pizzas ON order_details.pizza_id = pizzas.pizza_id;



Question 3 Identify the highest-priced pizza.

Query:

```
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;
```



Identify the most common pizza size ordered.

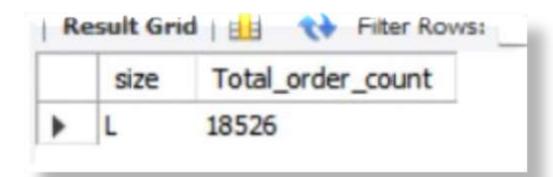
Query:

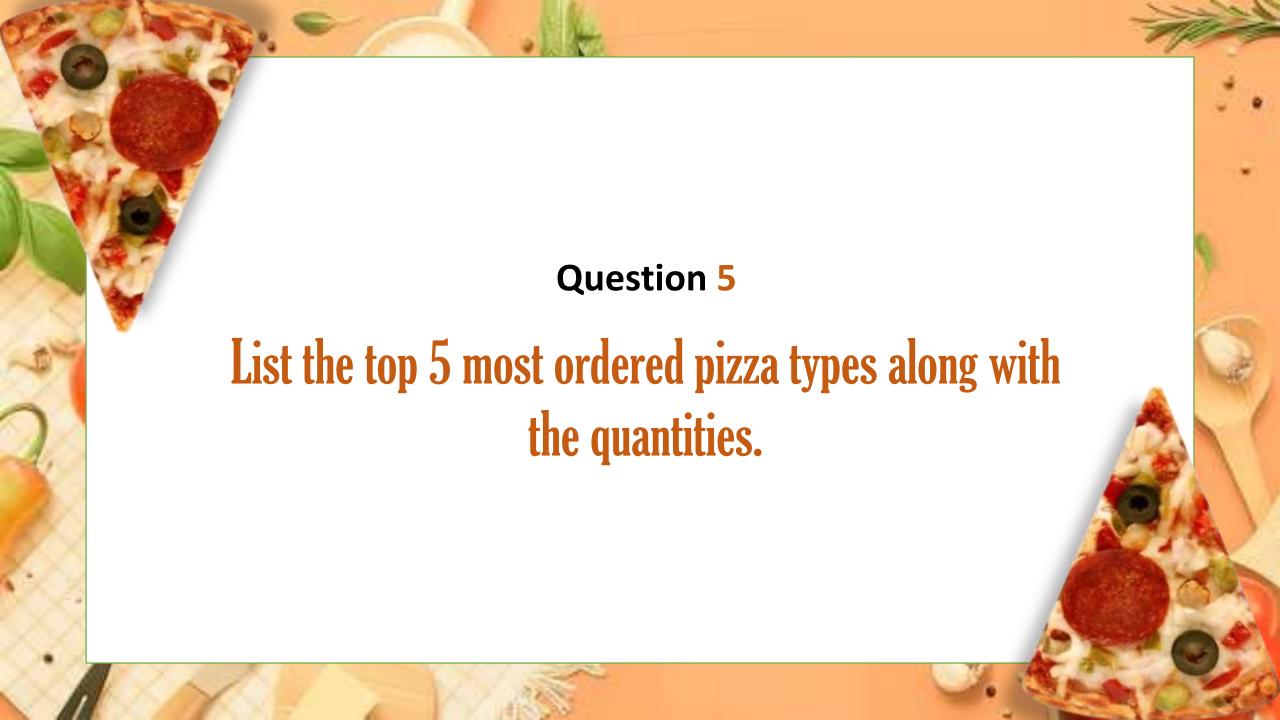
```
COUNT(order_details.order_details_id) AS Total_order_count
FROM pizzas

JOIN order_details ON pizzas.pizza_id = order_details.pizza_id

GROUP BY pizzas.size

ORDER BY Total_order_count DESC LIMIT 1;
```





SELECT pizza_types.name, SUM(order_details.quantity) AS Total_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 Total_Quantity DESC

LIMIT 5;

Result:

Query:

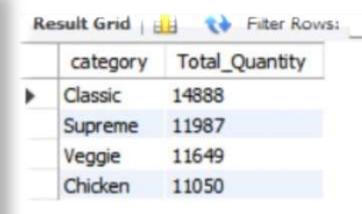


esult Grid 🔠 💎 Filter Ro	WS:	Export: :: Wi
name	Total_Quantity	
The Classic Deluxe Pizza	2453	
The Barbecue Chicken Pizza	2432	
The Hawaiian Pizza	2422	
The Pepperoni Pizza	2418	
The Thai Chicken Pizza	2371	
	name The Classic Deluxe Pizza The Barbecue Chicken Pizza The Hawaiian Pizza The Pepperoni Pizza	The Classic Deluxe Pizza 2453 The Barbecue Chicken Pizza 2432 The Hawaiian Pizza 2422 The Pepperoni Pizza 2418

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

Query: Result:

```
pizza_types.category,
    SUM(order_details.quantity) AS Total_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 Total_Quantity DESC
LIMIT 5;
```



Determine the distribution of orders by hour of the day.

Query:

```
HOUR(orders.order_time) As Order_Hour,
COUNT(orders.order_id) AS Orders
FROM
orders
GROUP BY Order_Hour;
```

Re	esult Grid 🔠	Filter Rovis:	
	Order_Hour	Orders	
٠	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.

Query:

```
pizza_types.category AS Category,
COUNT(pizza_types.name) AS Name
From
pizza_types
group by Category;
```



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

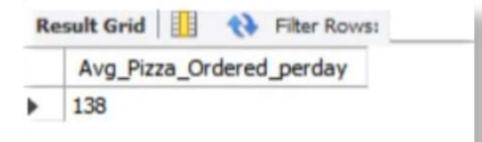
```
Query:
```

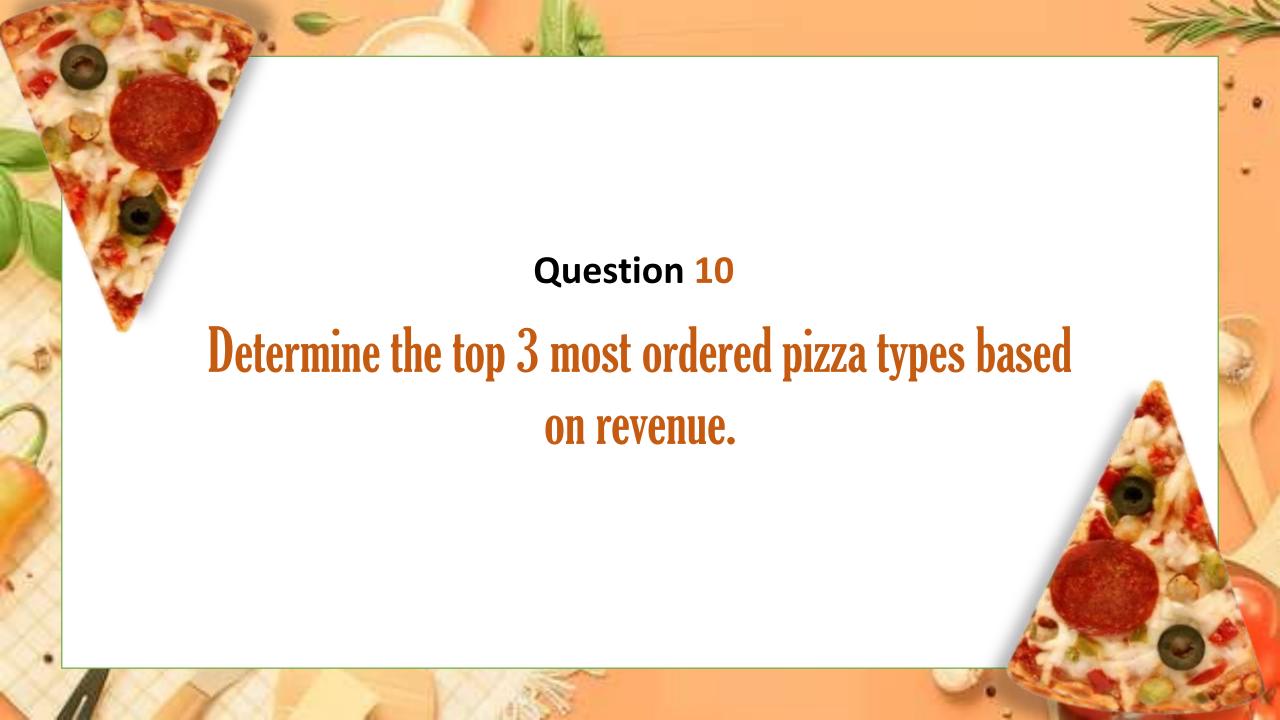
```
FROM

(SELECT orders.order_date,SUM(order_details.quantity) AS Quantity
FROM orders

JOIN order_details ON orders.order_id = order_details.order_id

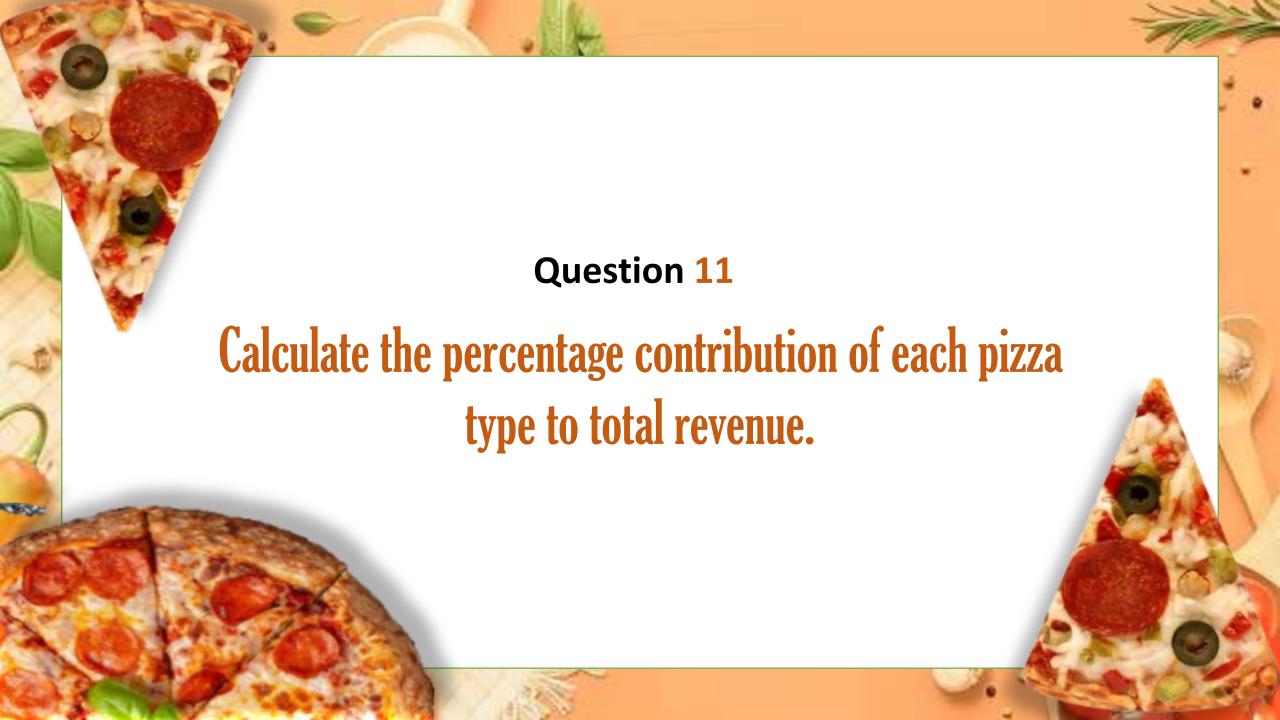
GROUP BY orders.order_date) AS Quantity;
```





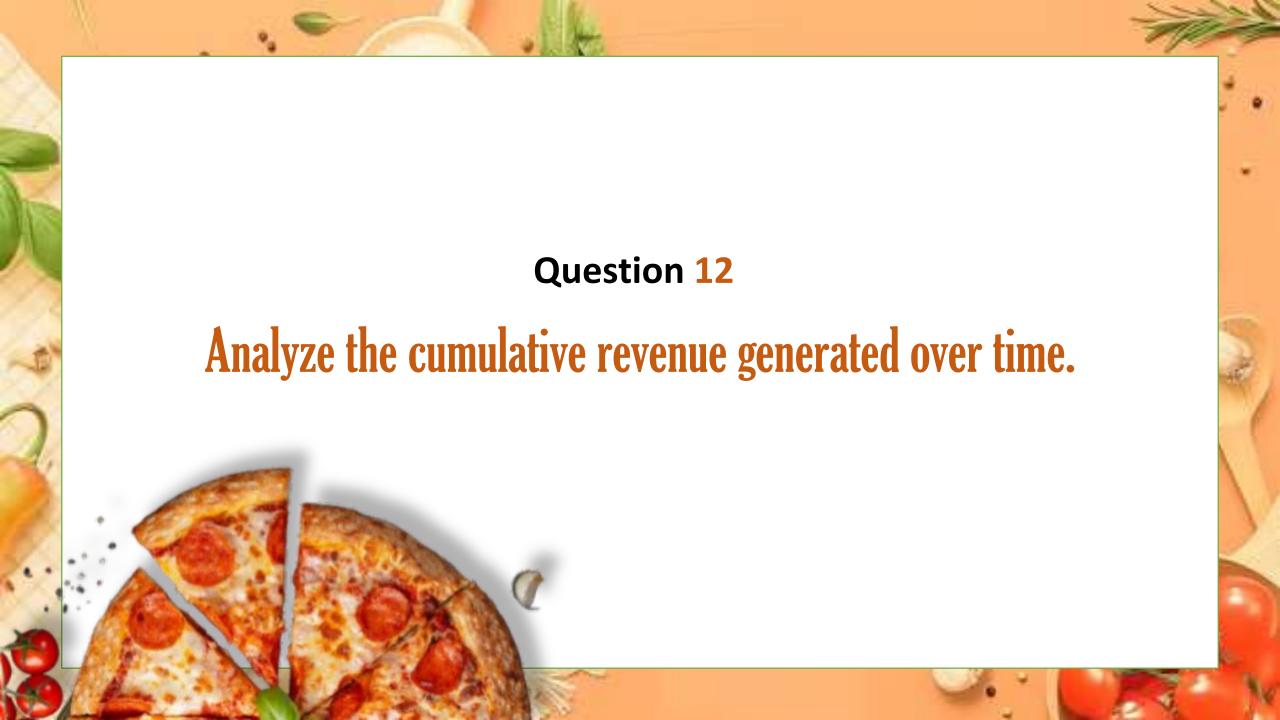
```
pizza_types.name AS Name,
   ROUND(SUM(order_details.quantity * pizzas.price),0) AS Revenue
FROM
   pizza_types
        JOIn
   pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
        order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY Name
ORDER BY Revenue DESC
LIMIT 3;
```

	Name	Revenue
•	The Thai Chicken Pizza	43434
	The Barbecue Chicken Pizza	42768
	The California Chicken Pizza	41410



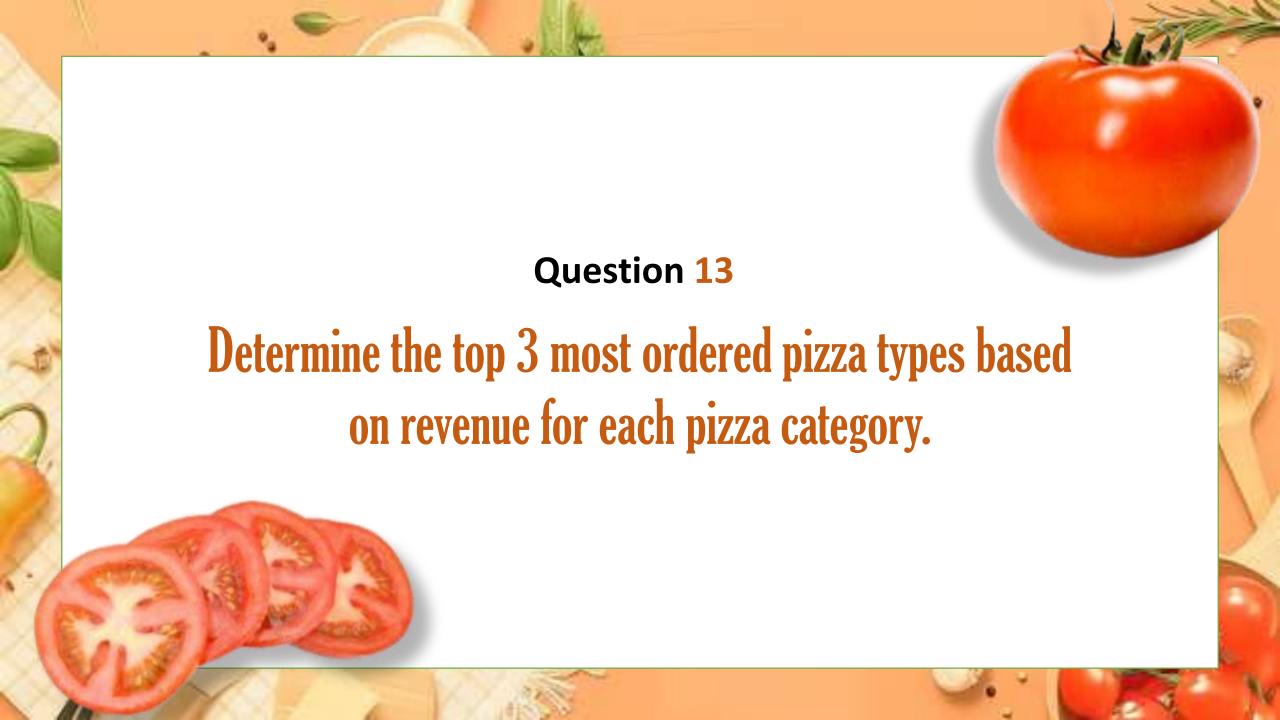
SELECT





```
SELECT
    order_date,
    SUM(Revenue) OVER (ORDER BY order_date) AS cumulative_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 order_details.order_id = orders.order_id
    GROUP BY orders.order_date
) AS Sales;
```

	order_date	cumulative_Revenue	
•	2015-01-01	2713.8500000000004	
	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	29831.30000000003	
	2015-01-14	32358.700000000004	
	2015-01-15	34343.50000000001	



```
SELECT name, revenue
FROM (
   SELECT category, name, revenue,
          RANK() OVER (PARTITION BY category ORDER BY revenue DESC) AS Rank_
   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
```

sult Grid 🔠 📢 Hiter Ko	Export:	
name	revenue	
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	
The Sicilian Pizza	30940.5	
The Four Cheese Pizza	32265.70000000065	
The Mexicana Pizza	26780.75	
The Five Cheese Pizza	26066.5	
	name The Thai Chicken Pizza The Barbecue Chicken Pizza The California Chicken Pizza The Classic Deluxe Pizza The Hawaiian Pizza The Pepperoni Pizza The Spicy Italian Pizza The Italian Supreme Pizza The Sicilian Pizza The Four Cheese Pizza The Mexicana Pizza	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 The Sicilian Pizza 30940.5 The Four Cheese Pizza 32265.700000000065 The Mexicana Pizza 26780.75

