

ABOUT ME

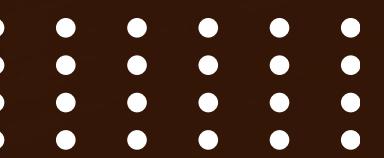
Chowderpally shivakiran

Detail-oriented Business Analyst with a Master's in Management, specializing in Logistics and Supply Chain. Skilled in designing data quality standards and metrics to ensure data accuracy. Proficient in Excel, Powerbi, SQL and Python, with strong analytical and problem-solving skills to drive data quality improvements.



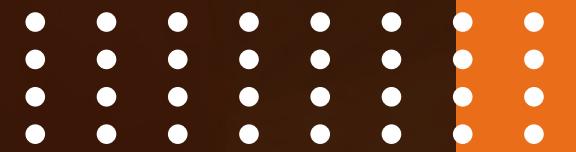
WELCOME TO PIZZA INSIGHTS: SQL-DRIVEN SALES OVERVIEW

From Dough to Data: Understanding Pizza Orders



Retrieve the total number of orders placed

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



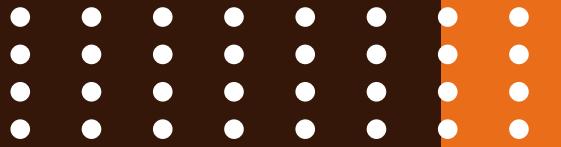
Output



A screenshot of a SQL query results window. The window has a title bar with '100 %' and a zoom control. Below the title bar are two tabs: 'Results' (selected) and 'Messages'. The results table has one row with the following data:

	total_orders
1	21350

Calculate the total revenue generated from pizza sales



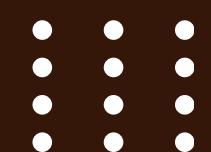
```
select  
round(sum(pizzas.price*order_details.quantity)  
,2) as total_sales from dbo.order_details  
join dbo.pizzas on pizzas.pizza_id =  
order_details.pizza_id;
```

Output

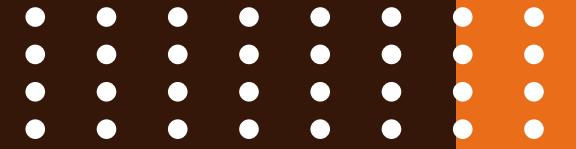
The screenshot shows a SQL query results window with the following details:

- Toolbar: Includes a zoom dropdown set to "100 %", a back arrow, and tabs for "Results" and "Messages".
- Table: A single row with one column named "total_sales". The value is 817860.05, which is highlighted with a blue selection box.

	total_sales
1	817860.05



Identify the highest-priced pizza



```
SELECT pizza_types.name, pizzas.price FROM  
dbo.pizza_types  
JOIN  
dbo.pizzas ON pizza_types.pizza_type_id =  
pizzas.pizza_type_id  
ORDER BY 2 DESC;
```

Output

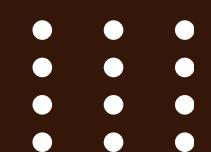
100 %

Results Messages

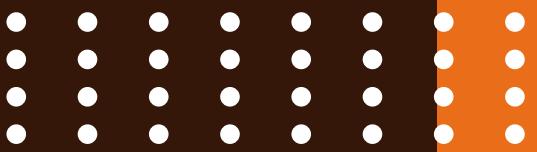
	name	price
1	The Greek Pizza	35.9500007629395
2	The Greek Pizza	25.5
3	The Brie Carre Pizza	23.6499996185303
4	The Italian Vegetables Pizza	21
5	The Spicy Italian Pizza	20.75
6	The Spinach Supreme Pizza	20.75
7	The Spinach Pesto Pizza	20.75
8	The Italian Supreme Pizza	20.75
9	The Pepper Salami Pizza	20.75
10	The Prosciutto and Arugula Pizza	20.75



use Limit function to see highest



Identify the most common pizza size ordered.



```
select
pizzas.size, count(order_details.order_details_id)
) as count
from pizzas join order_details
on pizzas.pizza_id= order_details.pizza_id
group by pizzas.size order by count desc;
```

Output

100 %

Results Messages

	size	count
1	L	18526
2	M	15385
3	S	14137
4	XL	544
5	XXL	28

Percentage Contribution of Each Pizza Type to Total Revenue(using CTE)

```
WITH total_revenue_cte AS (
    SELECT ROUND(SUM(od.quantity * p.price), 2) AS total_revenue FROM
        order_details as od JOIN pizzas p ON p.pizza_id = od.pizza_id),
    category_revenue_cte AS (SELECT pt.category AS pizza_category,
        SUM(od.quantity * p.price) AS category_revenue FROM pizza_types pt
        JOIN pizzas p ON p.pizza_type_id = pt.pizza_type_id
        JOIN order_details od ON od.pizza_id = p.pizza_id GROUP BY pt.category)
    SELECT cr.pizza_category,ROUND((cr.category_revenue / tr.total_revenue)
    * 100, 2) AS revenue FROM category_revenue_cte cr CROSS JOIN
    total_revenue_cte tr ORDER BY revenue DESC
```

Output

The screenshot shows a SQL query results window with the following details:

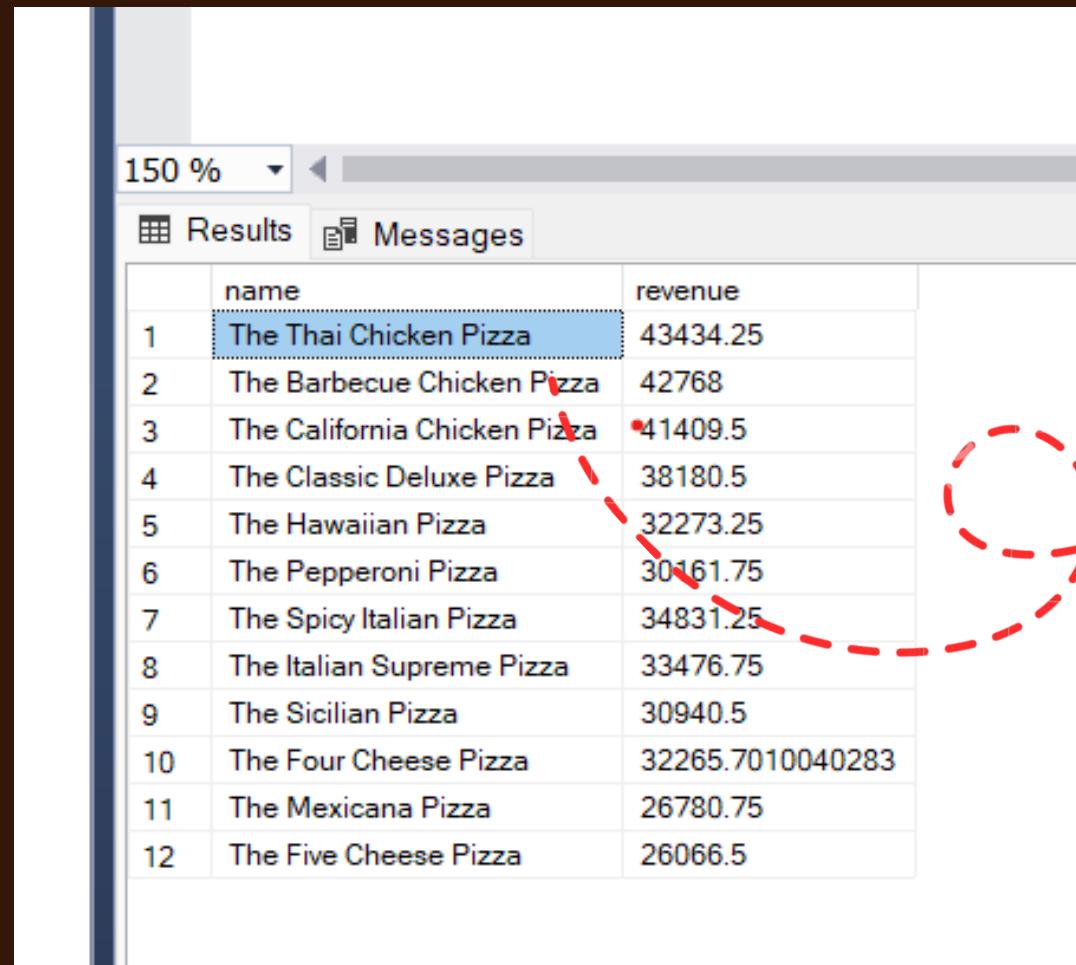
- Zoom Level:** 150%
- Results Tab:** Selected tab.
- Messages Tab:** Unselected tab.
- Table Headers:** pizza_category and revenue.
- Table Data:**

	pizza_category	revenue
1	Classic	26.91
2	Supreme	25.46
3	Chicken	23.96
4	Veggie	23.68

Top 3 Most Ordered Pizza Types by Revenue for Each 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
```

Output



The screenshot shows a SQL query results window with the following details:

- Zoom Level:** 150%
- Results Tab:** Active tab.
- Messages Tab:** Not active.
- Table Headers:** name, revenue
- Data Rows (12 total):**

	name	revenue
1	The Thai Chicken Pizza	43434.25
2	The Barbecue Chicken Pizza	42768
3	The California Chicken Pizza	41409.5
4	The Classic Deluxe Pizza	38180.5
5	The Hawaiian Pizza	32273.25
6	The Pepperoni Pizza	30161.75
7	The Spicy Italian Pizza	34831.25
8	The Italian Supreme Pizza	33476.75
9	The Sicilian Pizza	30940.5
10	The Four Cheese Pizza	32265.7010040283
11	The Mexicana Pizza	26780.75
12	The Five Cheese Pizza	26066.5

use Limit function to find top 3

THANK YOU FOR ATTENTION



Gmail:shivakiranchowderpally.com

[Linkedin](#)

[GitHub](#)