

PIZZA SALES ANALYSIS

Using MySQL



Pizzeria
House





HELLO !

My name is Tarun Sharma.
In this project, I have
utilised SQL queries to
solve questions that were
related to pizza sales.



LIST OF CONTENTS

Objective

Database Schema

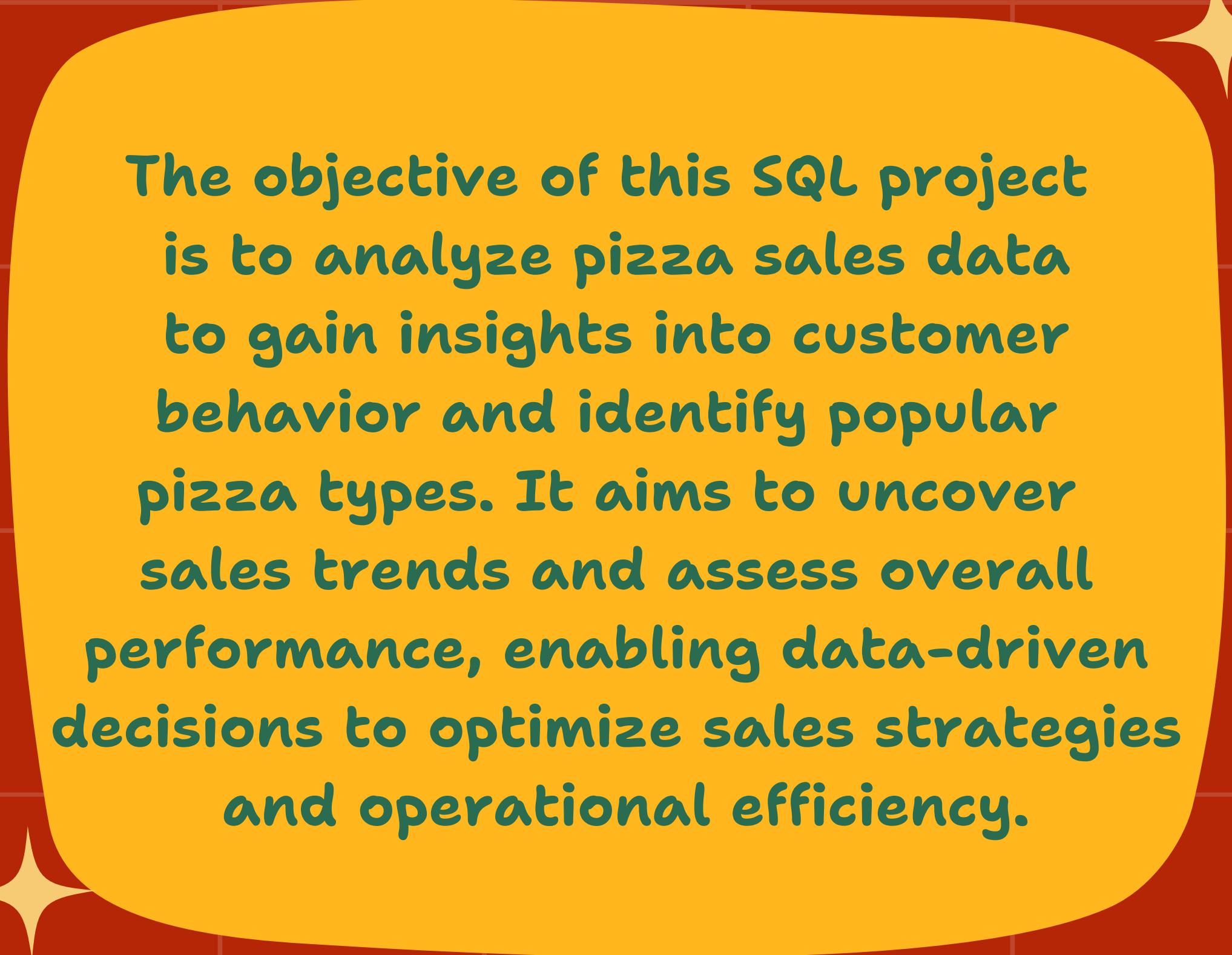
Table Details

Case Study Ques

Insights

Thank you

OBJECTIVE



The objective of this SQL project is to analyze pizza sales data to gain insights into customer behavior and identify popular pizza types. It aims to uncover sales trends and assess overall performance, enabling data-driven decisions to optimize sales strategies and operational efficiency.

DATABASE SCHEMA

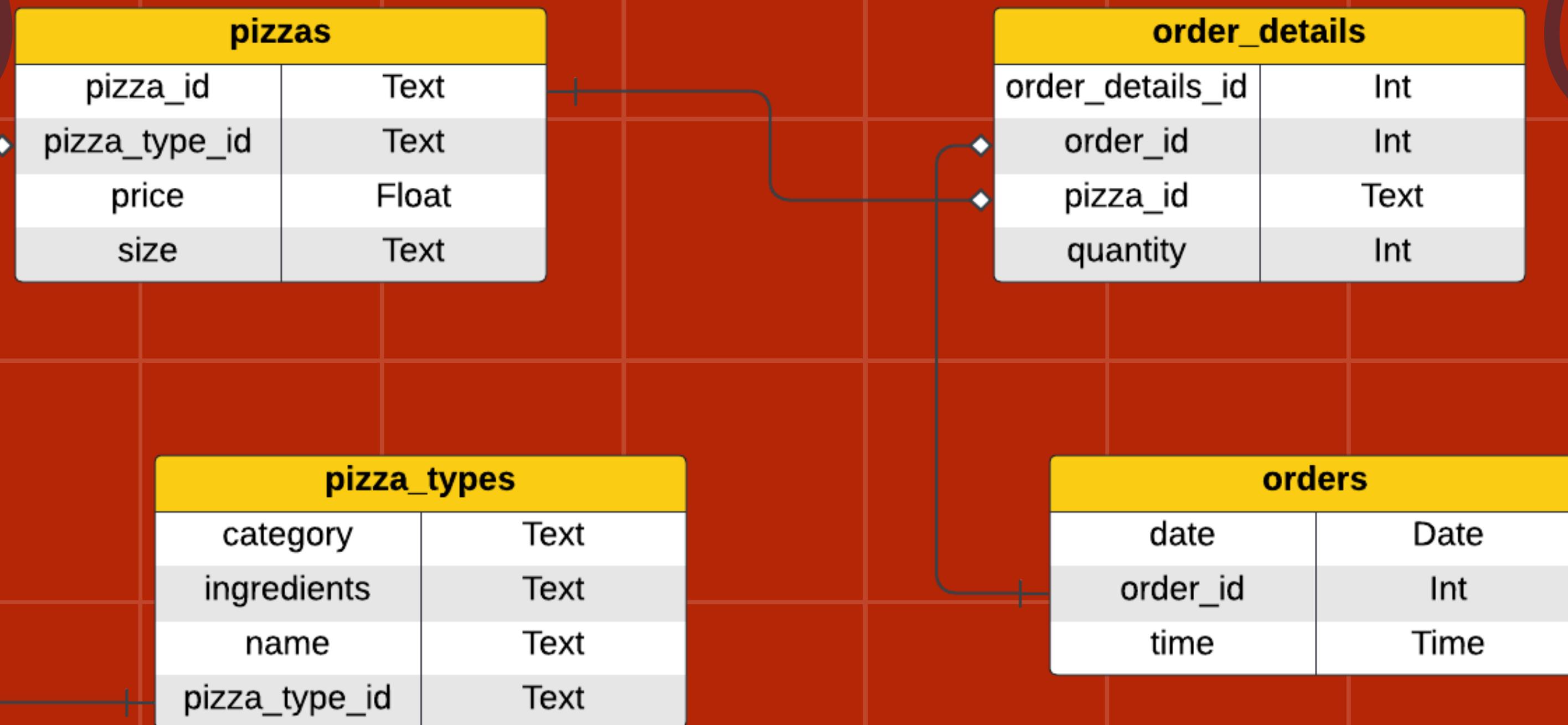


TABLE DETAILS

- **pizzas** : The pizza table contains details about pizzas with columns for pizza_id (unique identifier), pizza_type_id (links to pizza_types table), size (e.g., small, medium, large), and price.
- **pizza_types** : The pizza_type table lists pizza types with columns for pizza_type_id (unique identifier), name, category (e.g., vegetarian, meat), and ingredients (list of ingredients).
- **orders** : The orders table records order information with columns for order_id (unique identifier), date (order date), and time (order time).
- **order_details** : The order_details table tracks individual items in orders with columns for order_details_id, order_id, pizza_id, and quantity, linking orders to specific pizzas and their quantities.



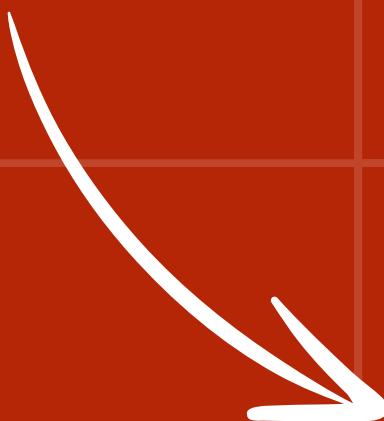
CASE STUDY QUESTION



The total number of orders placed.



```
SELECT  
    COUNT(DISTINCT order_id) AS total_orders_placed  
FROM  
    orders;
```



	total_orders_placed
▶	21350



The total revenue generated from pizza sales.



```
SELECT  
    CONCAT('$ ',  
           ROUND(SUM(p.price * od.quantity), 2)) AS  
    total_revenue  
FROM  
    pizzas p  
    JOIN  
    order_details od ON p.pizza_id = od.pizza_id;
```

A circular illustration of a delivery person wearing a red cap and shirt, holding a stack of pizza boxes.

A white callout box containing a table with two rows. A white arrow points from the word 'total_revenue' in the first row to the value in the second row.

	total_revenue
▶	\$ 817860.05

Identify the highest priced pizza.



```
SELECT
    pt.name, CONCAT('$ ', p.price) AS price
FROM
    pizza_types pt
    JOIN
    pizzas p ON pt.pizza_type_id = p.pizza_type_id
ORDER BY p.price DESC
LIMIT 1;
```



	name	price
▶	The Greek Pizza	\$ 35.95

The most common pizza size ordered.



```
SELECT  
    p.size AS common_size,  
    COUNT(od.order_details_id) AS order_count  
FROM  
    pizzas p  
    JOIN  
    order_details od ON p.pizza_id = od.pizza_id  
GROUP BY p.size  
ORDER BY order_count DESC  
LIMIT 1;
```



	common_size	order_count
▶	L	18526



Top 5 most ordered pizza types along with their quantities.

```
SELECT
    pt.name, SUM(od.quantity) AS total_quantity
FROM
    pizza_types pt
    JOIN
    pizzas p ON pt.pizza_type_id = p.pizza_type_id
    JOIN
    order_details od ON od.pizza_id = p.pizza_id
GROUP BY pt.name
ORDER BY total_quantity DESC
LIMIT 5;
```



	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

The total quantity of each pizza categories ordered.



```
SELECT
    pt.category, SUM(od.quantity) AS total_quantity
FROM
    pizza_types pt
    JOIN
    pizzas p ON pt.pizza_type_id = p.pizza_type_id
    JOIN
    order_details od ON od.pizza_id = p.pizza_id
GROUP BY pt.category
ORDER BY total_quantity DESC;
```

	category	total_quantity
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050



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)  
ORDER BY order_count DESC;
```



	hour	order_count
▶	12	2520
	13	2455
	18	2399
	17	2336
	19	2009
	16	1920
	20	1642
	14	1472
	15	1468
	11	1231
	21	1198
	22	663
	23	28
	10	8
	9	1

Category-wise distribution of pizzas.



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

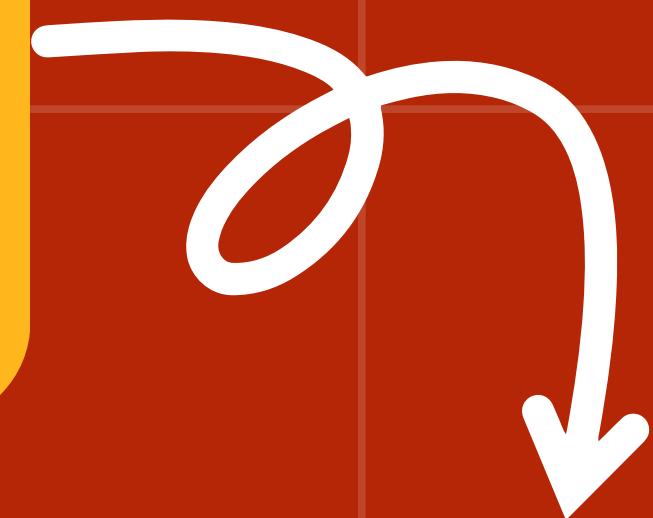


	category	name_count
▶	Supreme	9
	Veggie	9
	Classic	8
	Chicken	6

Average number of pizzas ordered per day.



```
SELECT  
    ROUND(AVG(quantity), 0) AS avg_pizza_ordered_per_day  
FROM  
(SELECT  
    o.order_date, SUM(od.quantity) AS quantity  
FROM  
    orders o  
JOIN order_details od ON o.order_id = od.order_id  
GROUP BY o.order_date) AS daily_orders;
```



	avg_pizza_ordered_per_day
▶	138

Top 3 most ordered pizza types based on revenue.

```
SELECT  
    pt.name,  
    CONCAT('$ ',  
          ROUND(SUM(p.price * od.quantity), 2)) AS 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.name  
ORDER BY Revenue DESC  
LIMIT 3;
```



	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.

```
SELECT
    pt.category,
    CONCAT('$ ', ROUND(SUM(od.quantity * p.price), 2)) AS revenue,
    CONCAT(ROUND((SUM(od.quantity * p.price) / (SELECT
        SUM(od.quantity * p.price)
    FROM
        pizzas p
        JOIN
            order_details od ON od.pizza_id = p.pizza_id)) * 100, 2),
    '%') AS percentage_contribution
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
ORDER BY Revenue DESC;
```

	category	revenue	percentage_contribution
▶	Classic	\$ 220053.1	26.91%
	Supreme	\$ 208197	25.46%
	Chicken	\$ 195919.5	23.96%
	Veggie	\$ 193690.45	23.68%

The cumulative revenue generated over time.



```
SELECT
    sales.order_date,
    CONCAT('$ ', sales.revenue) AS revenue,
    CONCAT('$ ', ROUND(SUM(sales.revenue) OVER (ORDER BY sales.order_date), 2))
AS cum_revenue
FROM
    (SELECT
        o.order_date,
        ROUND(SUM(p.price * od.quantity), 2) AS revenue
    FROM
        pizzas p
    JOIN
        order_details od ON p.pizza_id = od.pizza_id
    JOIN
        orders o ON od.order_id = o.order_id
    GROUP BY
        o.order_date) AS sales;
```

order_date	revenue	cum_revenue
2015-01-01	\$ 2713.85	\$ 2713.85
2015-01-02	\$ 2731.9	\$ 5445.75
2015-01-03	\$ 2662.4	\$ 8108.15
2015-01-04	\$ 1755.45	\$ 9863.6
2015-01-05	\$ 2065.95	\$ 11929.55
2015-01-06	\$ 2428.95	\$ 14358.5
2015-01-07	\$ 2202.2	\$ 16560.7
2015-01-08	\$ 2838.35	\$ 19399.05
2015-01-09	\$ 2127.35	\$ 21526.4
2015-01-10	\$ 2463.95	\$ 23990.35
2015-01-11	\$ 1872.3	\$ 25862.65
2015-01-12	\$ 1919.05	\$ 27781.7
2015-01-13	\$ 2049.6	\$ 29831.3
2015-01-14	\$ 2527.4	\$ 32358.7



Top 3 most ordered pizza type based on revenue for each pizza category.

```
WITH pizza_revenue AS (
    SELECT
        pt.category, pt.name AS pizza_type,
        SUM(od.quantity * p.price) AS revenue
    FROM pizzas p
    JOIN
        pizza_types pt ON p.pizza_type_id = pt.pizza_type_id
    JOIN
        order_details od ON p.pizza_id = od.pizza_id
    GROUP BY pt.category, pt.name),
    ranked_pizza_revenue AS (
        SELECT
            category, pizza_type, revenue, ROW_NUMBER() OVER (PARTITION BY
                category ORDER BY revenue DESC) AS r
            FROM
                pizza_revenue)
    SELECT
        category, pizza_type,
        CONCAT('$ ', ROUND(revenue, 2)) AS revenue
    FROM ranked_pizza_revenue
    WHERE r <= 3
    ORDER BY category, revenue DESC;
```

	category	pizza_type	revenue
▶	Chicken	The Thai Chicken Pizza	\$ 43434.25
	Chicken	The Barbecue Chicken Pizza	\$ 42768
	Chicken	The California Chicken Pizza	\$ 41409.5
	Classic	The Classic Deluxe Pizza	\$ 38180.5
	Classic	The Hawaiian Pizza	\$ 32273.25
	Classic	The Pepperoni Pizza	\$ 30161.75
	Supreme	The Spicy Italian Pizza	\$ 34831.25
	Supreme	The Italian Supreme Pizza	\$ 33476.75
	Supreme	The Sicilian Pizza	\$ 30940.5
	Veggie	The Four Cheese Pizza	\$ 32265.7
	Veggie	The Mexicana Pizza	\$ 26780.75
	Veggie	The Five Cheese Pizza	\$ 26066.5



INSIGHTS

- **Total Orders and Revenue:** The store received a total of 21,350 orders, generating a total revenue of \$ 8,17,860.05.
- **Highest Priced Pizza:** The Greek pizza is the most expensive, priced at \$ 35.95.
- **Most Common Pizza Size:** The large pizza size ("L") is the most popular, with 18,526 orders.
- **Most Ordered Pizza Category:** The classic pizza category leads in popularity with 14,888 orders.
- **Peak Ordering Hours:** The busiest times for orders are between 12 PM and 1 PM.
- **Average Pizzas Ordered Per Day:** On average, 138 pizzas are ordered daily at the store.
- **Top Ordered Pizza by Revenue:** The Thai Chicken pizza generated the highest revenue, totaling \$ 43,434.25.
- **Revenue Contribution:** Classic pizza dominates the total revenue with \$ 2,20,053.10, accounting for a contribution of 26.91%.

Pizzeria
House

THANK YOU

"Slice of Heaven on Every Plate."

~By Tarun Sharma

