



# Data-Driven Insights for Pizza Sales :

From Analysis to  
Actionable Business  
Strategies



# PROJECT OVERVIEW

In this project, I conducted a comprehensive analysis of pizza sales data to uncover actionable insights and drive strategic business decisions. Leveraging a robust dataset comprising orders, order details, pizzas, and pizza types, I employed advanced SQL techniques to transform raw data into meaningful intelligence.

## Approach:

- 1. Advanced SQL Analytics:** Leveraged complex JOIN operations, subqueries, Common Table Expressions (CTEs) for multi-dimensional data integration and window functions for sophisticated time-based analyses and ranking evaluations.
- 2. Sales and Trend Analysis:** Analyzed temporal sales patterns to identify peak periods and seasonal trends.
- 3. Customer Segmentation and Behavior:** Segmented customers based on purchasing behavior and order frequency. Derived insights into customer preferences and loyalty trends.
- 4. Product Performance Evaluation:** Evaluated sales volume, revenue, and profitability metrics for each pizza category to identify top performers.



# DATASET OVERVIEW

This pizza sales dataset offers a comprehensive overview of transactional data from a pizza restaurant over the course of a year. It provides detailed insights into customer orders, including pizza specifications, order times, and pricing information.

Dataset Composition includes four tables:

- 1.Orders: Contains order\_id, date, time, and customer\_id
- 2.Order\_details: Includes order\_id, pizza\_id, and quantity
- 3.Pizzas: Stores pizza\_id, pizza\_type\_id, size, and price
- 4.Pizza\_types: Contains pizza\_type\_id, name, category, and ingredients



# KEY PROBLEMS

I leveraged SQL to extract meaningful insights across various complexity levels, addressing key business questions:

1. Retrieve the total number of orders placed.
2. Calculate the total revenue generated from pizza sales.
3. Identify the highest-priced pizza.
4. Identify the most common pizza size ordered.
5. List the top 5 most ordered pizza types along with their quantities.
6. Join the necessary tables to find the total quantity of each pizza category ordered.
7. Determine the distribution of orders by hour of the day.



The background of the slide features a close-up, slightly blurred image of several pizzas. In the foreground, a pizza is topped with fresh green leafy vegetables and sliced strawberries. Behind it, another pizza with a tomato and pepperoni topping is visible. The pizzas are resting on metal trays. The overall lighting is warm and the image has a soft, artistic feel.

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

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

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

11. Calculate the percentage contribution of each pizza type to total revenue.

12. Analyze the cumulative revenue generated over time.

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



# Total number of orders placed

## QUERY

```
SELECT  
  COUNT(order_id) AS TOTAL_ORDERS  
FROM order_details;
```

## OUTPUT

Result Grid  	
	TOTAL_ORDERS
	48620





# Total revenue generated from pizza sales

## QUERY

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

## OUTPUT

Result Grid   Filter Rows:	
TOTAL_REVENUE_GENERATED	
817860.05	





# The Highest-priced pizza

## QUERY

```
SELECT
pizza_types.pizza_type_id,
pizzas.price
FROM pizza_types
JOIN pizzas
ON pizza_types.pizza_type_id = pizzas.pizza_type_id
ORDER BY pizzas.price DESC LIMIT 1;
```

## OUTPUT

Result Grid   Filter Row	
pizza_type_id	price
the_greek	35.95



# Most common pizza size ordered

## QUERY

```
SELECT
pizzas.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
LIMIT 1;
```

## OUTPUT

Result Grid				Filter
size	ORDER_COUNT			
L	18526			




# Top 5 most ordered pizza types along with their quantities

## QUERY

```
SELECT
pizza_types.name,
SUM(order_details.quantity) AS QUANTITY
FROM pizzas
JOIN order_details
ON pizzas.pizza_id = order_details.pizza_id
JOIN pizza_types
ON pizzas.pizza_type_id = pizza_types.pizza_type_id
GROUP BY 1
ORDER BY 2 DESC
LIMIT 5
```

## OUTPUT

Result Grid   Filter Rows:

	name	QUANTITY
	The Classic Deluxe Pizza	2453
<input type="checkbox"/>	The Barbecue Chicken Pizza	2432
	The Hawaiian Pizza	2422
<input type="checkbox"/>	The Pepperoni Pizza	2418
	The Thai Chicken Pizza	2371



# Total quantity of each pizza category ordered

## QUERY

```
SELECT
pizza_types.category,
SUM(order_details.quantity) AS QUANTITY
FROM pizzas
JOIN pizza_types
ON pizzas.pizza_type_id = pizza_types.pizza_type_id
JOIN order_details
ON pizzas.pizza_id = order_details.pizza_id
GROUP BY 1
```

## OUTPUT

Result Grid   Filter Rows:		
	category	QUANTITY
	Classic	14888
	Veggie	11649
	Supreme	11987
	Chicken	11050





# Distribution of orders by hour of the day

## QUERY

```
SELECT  
  HOUR(orders.time) AS HOUR,  
  COUNT(orders.order_id) AS TOTAL_ORDERS  
FROM orders  
GROUP BY 1  
ORDER BY 1
```

## OUTPUT

Result Grid   Filter		
	HOUR	TOTAL_ORDERS
	9	1
	10	8
	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





# Category-wise distribution of pizzas

## QUERY

```
SELECT  
pizza_types.category,  
COUNT(*) AS DISTRIBUTION_OF_PIZZA  
FROM pizza_types  
GROUP BY 1
```

## OUTPUT

Result Grid   Filter Rows: <input type="text" value="Search"/>		
	category	DISTRIBUTION_OF_PIZZA
	Chicken	6
	Classic	8
	Supreme	9
	Veggie	9



# Average number of pizzas ordered per day

## QUERY

```
SELECT  
ROUND(AVG(QUANTITY),0) AS AVG_PIZZA_PER_DAY  
FROM  
(  
  SELECT  
    orders.date,  
    SUM(order_details.quantity) AS QUANTITY  
  FROM orders  
  JOIN order_details  
  ON orders.order_id = order_details.order_id  
  GROUP BY 1  
)A
```

## OUTPUT

Result Grid				Filter
AVG_PIZZA_PER_DAY				
138				



# Top 3 most ordered pizza types based on revenue

## QUERY

```
SELECT
pizza_types.name,
SUM(order_details.quantity * pizzas.price) AS REVENUE
FROM pizzas
JOIN pizza_types
ON pizzas.pizza_type_id = pizza_types.pizza_type_id
JOIN order_details
ON pizzas.pizza_id = order_details.pizza_id
GROUP BY 1
ORDER BY 2 DESC
LIMIT 3
```

## OUTPUT

Result Grid		Filter Rows:	Q	S
	name	REVENUE		
	The Thai Chicken Pizza	43434.25		
	The Barbecue Chicken Pizza	42768		
	The California Chicken Pizza	41409.5		



# Percentage contribution of each pizza type to total revenue


## QUERY

```
WITH CTE AS(
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
)
SELECT
pizza_types.category AS PIZZA,
ROUND((SUM(pizzas.price *
order_details.quantity)/CTE.TOTAL_REVENUE)*100,2) AS
REVENUE_PERCENTAGE
FROM pizzas
JOIN pizza_types
ON pizzas.pizza_type_id = pizza_types.pizza_type_id
JOIN order_details
ON pizzas.pizza_id = order_details.pizza_id
CROSS JOIN CTE
GROUP BY 1,CTE.TOTAL_REVENUE
ORDER BY 2 DESC
```



# Percentage contribution of each pizza type to total revenue

## OUTPUT

Result Grid				Filter Rows:	<input type="text" value="Search"/>
	PIZZA	REVENUE_PERCENTAGE			
	Classic	26.91			
<input type="checkbox"/>	Supreme	25.46			
	Chicken	23.96			
<input type="checkbox"/>	Veggie	23.68			



# Cumulative revenue generated over time

## QUERY

```
SELECT
  DATE,
  ROUND(SUM(REVENUE) OVER(ORDER BY DATE),2) AS
  CUMULATIVE_REVENUE
FROM
  (
    SELECT
    orders.date AS 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 1
  )A
```



# Cumulative revenue generated over time

## OUTPUT

Result Grid   Filter Rows: <input type="text"/>		
	DATE	CUMULATIVE_REVENUE
	2015-01-01	2713.85
<input type="checkbox"/>	2015-01-02	5445.75
	2015-01-03	8108.15
<input type="checkbox"/>	2015-01-04	9863.6
	2015-01-05	11929.55
<input type="checkbox"/>	2015-01-06	14358.5
	2015-01-07	16560.7
<input type="checkbox"/>	2015-01-08	19399.05
	2015-01-09	21526.4
<input type="checkbox"/>	2015-01-10	23990.35
	2015-01-11	25862.65
<input type="checkbox"/>	2015-01-12	27781.7
	2015-01-13	29831.3
<input type="checkbox"/>	2015-01-14	32358.7
	2015-01-15	34343.5
<input type="checkbox"/>	2015-01-16	36937.65
	2015-01-17	39001.75
<input type="checkbox"/>	2015-01-18	40978.6
	2015-01-19	43365.75
<input type="checkbox"/>	2015-01-20	45763.65



# Top 3 most ordered pizza types based on revenue for each pizza category

## QUERY

```
SELECT
PIZZA_NAME,
REVENUE,
ORDER_RANKING
FROM
(SELECT
PIZZA_CATEGORY,
PIZZA_NAME,
REVENUE,
RANK() OVER(PARTITION BY PIZZA_CATEGORY ORDER BY
REVENUE DESC) AS ORDER_RANKING
FROM
(
SELECT
pizza_types.category AS PIZZA_CATEGORY,
pizza_types.name AS PIZZA_NAME,
SUM(pizzas.price * order_details.quantity) AS REVENUE
FROM pizzas
JOIN pizza_types
ON pizzas.pizza_type_id = pizza_types.pizza_type_id
JOIN order_details
ON pizzas.pizza_id = order_details.pizza_id
GROUP BY 1,2
)A
)B
WHERE ORDER_RANKING <= 3
```



# Cumulative revenue generated over time

## OUTPUT

	PIZZA_NAME	REVENUE	ORDER_RANKING
	The Thai Chicken Pizza	43434.25	1
	The Barbecue Chicken Pizza	42768	2
	The California Chicken Pizza	41409.5	3
	The Classic Deluxe Pizza	38180.5	1
	The Hawaiian Pizza	32273.25	2
	The Pepperoni Pizza	30161.75	3
	The Spicy Italian Pizza	34831.25	1
	The Italian Supreme Pizza	33476.75	2
	The Sicilian Pizza	30940.5	3
	The Four Cheese Pizza	32265.700000000065	1
	The Mexicana Pizza	26780.75	2



The image shows three pizzas on metal trays. The pizza in the foreground is topped with a fresh salad of green leaves, sliced strawberries, and orange segments. The pizza in the background is a pepperoni pizza. The pizza on the right is partially visible and also appears to be a pepperoni pizza. The text "Thank you!" is written in a white, handwritten-style font across the middle of the image.

Thank you!