

# SQL PROJECT ON PIZZA SALE

This project analyzes a comprehensive pizza sales dataset using SQL to uncover valuable insights into customer preferences, revenue patterns, and product performance. The dataset includes details on orders, pizza sizes, prices, and categories. Key objectives included calculating total orders, total revenue, identifying best-selling products, and finding the most popular pizza sizes and categories. SQL concepts like JOIN, GROUP BY, ORDER BY, and aggregate functions were used to answer business questions and generate actionable insights.

CHECK NOW



# PROJECT OVERVIEW

## CONTENT:

- Objective: Analyze pizza sales to answer business questions and identify patterns.
- Dataset Tables:
  - orders – Order date, time, and IDs
  - order\_details – Pizza quantities per order
  - pizzas – Pizza size, price, ID
  - pizza\_types – Category, name, ingredients
- Scope: Basic and intermediate-level SQL queries.

## Visual Idea:

- Dataset schema diagram showing table relationships.



# TOOLS & SKILLS USED

## CONTENT:

Tools: SQL (MySQL )

## Skills Applied:

Aggregate functions (SUM, COUNT, MAX)

Joins (INNER JOIN)

Filtering & sorting (WHERE, ORDER BY)

Grouping (GROUP BY)

Data analysis for business decision-making.



# BUSINESS QUESTIONS ADDRESSED IN THE ANALYSIS

## CONTENT:

Retrieve the total number of orders placed.

Calculate the total revenue from pizza sales.

Identify the highest-priced pizza.

Determine the most common pizza size ordered.

List the top 5 most ordered pizza types by quantity.

Find the total quantity of each pizza category ordered by joining multiple tables.

- 



Q.1

# RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.

- `select count(order_id) as Total_Orders from orders;`

Result Grid	
	Total_Orders
▶	21350



Q.2

# CALCULATE THE TOTAL REVENUE FROM PIZZA SALES.

- **SELECT**

```
ROUND(SUM(orders_details.quantity * pizzas.price),  
     2) AS Total_Sales  
  
FROM  
  
    orders_details  
  
    JOIN  
  
    pizzas ON pizzas.pizza_id = orders_details.pizza_id
```

Result Grid	
	Total_Sales
▶	817860.05



Q.3

# IDENTIFY THE HIGHEST-PRICED PIZZA.

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;

	name	price
▶	The Greek Pizza	35.95



Q.4

# DETERMINE THE MOST COMMON PIZZA SIZE ORDERED.

SELECT

```
pizzas.size,  
COUNT(orders_details.order_details_id) AS order_count  
FROM  
pizzas  
JOIN  
orders_details ON pizzas.pizza_id = orders_details.pizza_id  
GROUP BY pizzas.size  
ORDER BY order_count DESC;
```

size	order_count
L	18526
M	15385
S	14137
XL	544
XXL	0



Q.5

# LIST THE TOP 5 MOST ORDERED PIZZA TYPES BY QUANTITY.

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

	name	quantity
▶	The Classic Deluxe Pizza	2453
	The Barbecue Chicken Pizza	2432
	The Hawaiian Pizza	2422
	The Pepperoni Pizza	2418
	The Thai Chicken Pizza	2371



**Q.6**

FIND THE TOTAL QUANTITY OF EACH PIZZA CATEGORY ORDERED BY JOINING MULTIPLE TABLES.

**SELECT**

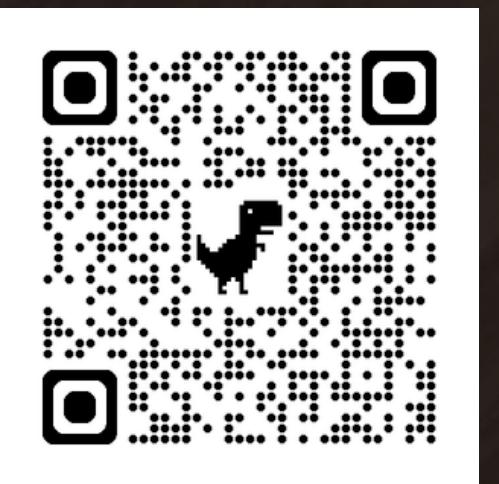
```
 pizza_types.category,  
 SUM(orders_details.quantity) AS quantity  
FROM  
 pizza_types  
 JOIN  
 pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id  
 JOIN  
 orders_details ON orders_details.pizza_id = pizzas.pizza_id  
GROUP BY pizza_types.category  
ORDER BY quantity desc;
```

	category	quantity
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050

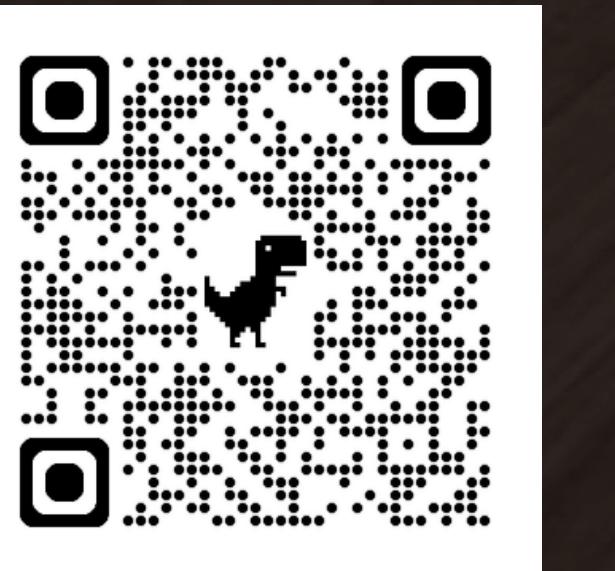


# THANK YOU - LET'S CONNECT!

LINKEDIN



PORTFOLIO



GITHUB

