



PIZZA SALES STORES DATA

PROJECT

# PIZZA SALES STORE SQL PROJECT





# PIZZA SALES STORES

SQL Project

Hello

Santosh Ambule here..

Hello,

In this project, I have created a Pizza Sales Store Analysis using SQL queries. The project is based on a real-world scenario where I analyzed sales data to extract valuable business insights.

I used structured SQL queries to answer key business questions such as identifying the best-selling pizza types, tracking monthly revenue, analyzing peak order times, and understanding customer preferences.

This hands-on project enhanced my skills in SQL, data analysis, and problem-solving—helping me understand how data can drive strategic business decisions.





# SQL QUESTION BASED ON PIZZAS SALES STORES

01

**Retrieve the total number of orders placed ?**

02

**Calculate the total revenue generated from pizza sales ?**

03

**Identify the most common pizza size ordered ?**

04

**Identify the most common pizza size ordered ?**

05

**Determine the distribution of orders by hour of the day ?**

06

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

07

**Analyze the cumulative revenue generated over time ?**

Q1.

# RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED ?

```
#Retrieve the total number of orders placed.  
  
select count(order_id) as total_orders from orders
```

Result Grid   Filter Rows:

total_orders
21350

## Q2. CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES ?

```
1 # Q2 Calculate the total revenue generated from pizza sales.  
2  
3 • Select  
4 sum(order_details.quantity*pizzas.price) as total_sales  
5 from order_details join pizzas  
6 on pizzas.pizza_id = order_details.pizza_id  
7
```

Result Grid	
	 Filter Rows: <input type="button" value="C"/>
total_sales	
817860.05	

Q3.

## IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED ?

```
1      # Identify the most common pizza size ordered.  
2  
3 •   SELECT  
4       Quantity, COUNT(order_details_id)  
5   FROM  
6       Order_details  
7   GROUP BY quantity;  
8
```

Result Grid   Filter Rows:

Quantity	COUNT(order_details_...
1	47693
2	903
3	21
4	3

Q4.

# IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED ?

```
# Identify the most common pizza size ordered.

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;
```

Result Grid   Filter Rows:

size	order_count
L	18526
M	15385
S	14137
XL	544
XXL	28

## Q5. DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY ?

```
1  #Determine the distribution of orders by hour of the day.  
2  
3  
4 • SELECT  
5      HOUR(order_time) AS hour, COUNT(order_id) AS order_count  
6  FROM  
7      orders  
8  GROUP BY HOUR(order_time);  
9
```

Result Grid   Filter Rows:  

hour	order_count
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399

Q6.

# GROUP THE ORDERS BY DATE AND CALCULATE THE AVERAGE NUMBER OF PIZZAS ORDERED PER DAY ?

```
1  # Group the orders by date and calculate the average number of pizzas ordered per day.  
2  
3 • SELECT  
4      ROUND(AVG(quantity), 0) as avg_pizzas_orders_per_Day  
5  FROM  
6  (SELECT  
7      orders.order_date, SUM(order_details.quantity) AS quantity  
8  FROM  
9      orders  
10     JOIN order_details ON orders.order_id = order_details.order_id  
11     GROUP BY orders.order_date) AS order_quantity;
```

Result Grid	
	Filter Rows:
avg_pizzas_orders_per_Day	
138	

**Q7.**

# ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME ?

```

1  # Analyze the cumulative revenue generated over time.
2
3 • Select order_date,
4   sum(revenue) over (order by order_date) as cum_revenue
5   from
6   (SELECT
7     orders.order_date,
8     ROUND(SUM(order_details.quantity * Pizzas.price),
9           2) AS revenue
10  FROM
11    order_details
12    JOIN
13    pizzas ON order_details.pizza_id = pizzas.pizza_id
14    JOIN
15    orders ON orders.order_id = order_details.order_id
16  GROUP BY orders.order_date) as sales;

```

**Result Grid**   Filter Rows:  Search

order_date	cum_revenue
2015-01-01	2713.85
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	18888.95



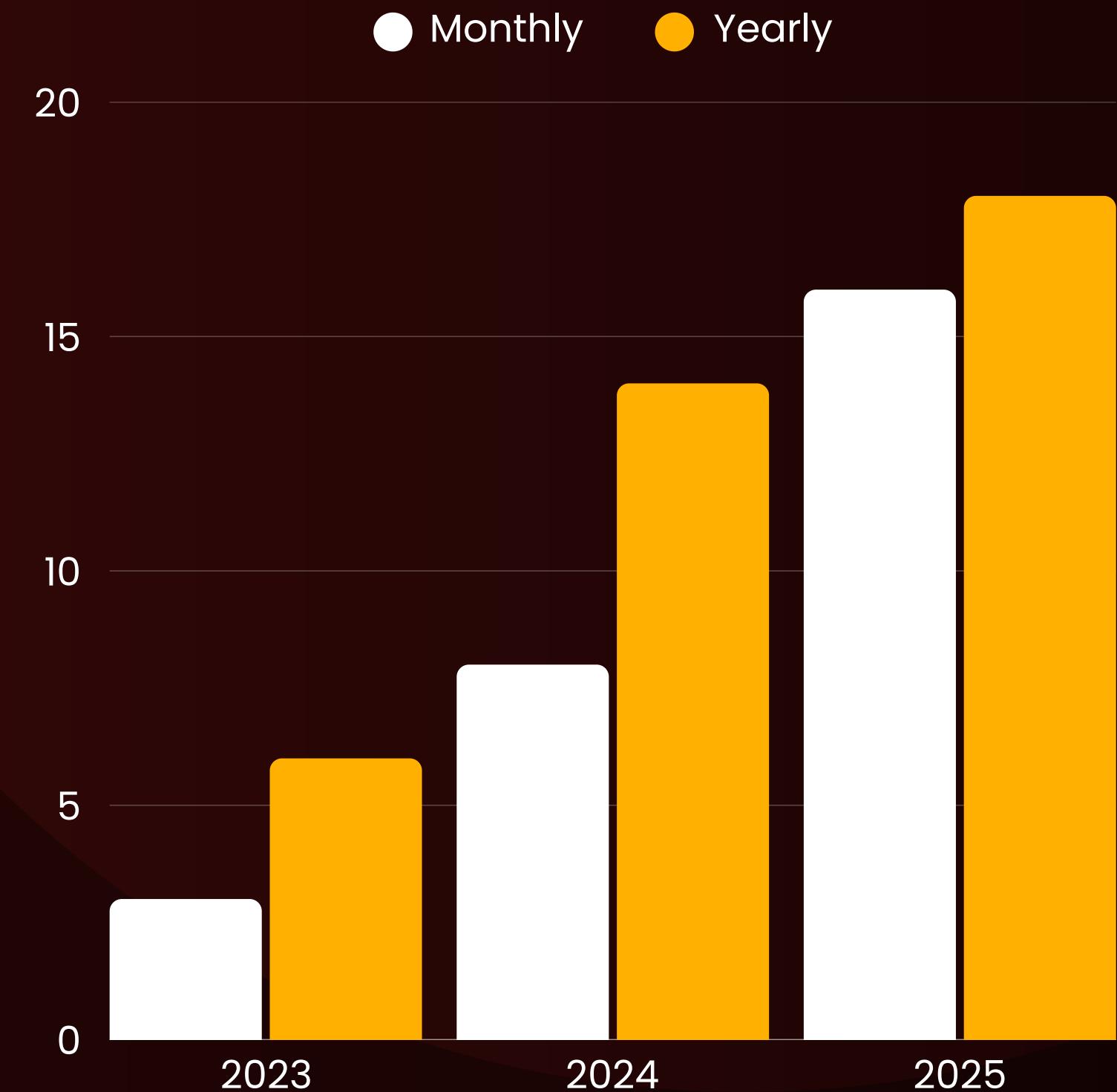
# PIZZA SALES STORE

Project

# SALES REPORT

## TOTAL SALES

This bar chart represents the sales report of various pizza types based on total revenue. It helps identify the top-performing pizzas and supports data-driven decisions for inventory and marketing strategies.





**PIZZA SALES STORE**

Project

Home

About

Contact

# THANK YOU

## FOR ATTENTION

- PIZZA SALES STORE SQL PROJECT !...