

# PIZZA

Project Using SQL



# INTRODUCTION

This project presents a comprehensive analysis of pizza sales data using SQL. The main objective is to extract meaningful business insights from raw data by writing efficient and insightful queries. The dataset consists of multiple tables, including orders, order details, pizza types, and pricing. We use SQL to explore various aspects such as total orders, revenue generation, and pizza popularity. Through joins, aggregations, and filtering, we analyze customer behavior and sales trends. Key questions include identifying best-selling pizza types, most preferred sizes, and peak order hours. The project also dives into revenue breakdowns and cumulative trends over time. Ultimately, this analysis supports data-driven decision-making for optimizing sales strategies.



A close-up photograph of a person's hand holding a wooden spoon and stirring a pan of pasta. The pasta appears to be fettuccine, mixed with cherry tomatoes and some herbs or meat. The pan is dark-colored and sits on a stove. The background is blurred.

1. RETRIEVE THE TOTAL NUMBER  
OF ORDERS PLACED.

SELECT COUNT(ORDER\_ID) AS  
TOTAL\_ORDERS  
FROM ORDERS;



2. CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

```
SELECT SUM(O.QUANTITY *  
P.PRICE) AS TOTAL_REVENUE  
FROM ORDER_DETAILS O  
JOIN PIZZAS P  
ON O.PIZZA_ID = P.PIZZA_ID;
```



3. IDENTIFY THE HIGHEST-PRICED PIZZA.

```
SELECT P.NAME , PI.PRICE  
FROM PIZZA_TYPES P  
JOIN PIZZAS PI  
ON P.PIZZA_TYPE_ID =  
PI.PIZZA_TYPE_ID  
ORDER BY PI.PRICE DESC LIMIT 1;
```



#### 4. IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.

```
SELECT P.SIZE, COUNT(O.QUANTITY)  
AS ORDER_PIZZA  
FROM PIZZAS P  
JOIN ORDER_DETAILS O  
ON P.PIZZA_ID = O.PIZZA_ID  
GROUP BY P.SIZE  
ORDER BY ORDER_PIZZA DESC  
LIMIT 1;
```



5. LIST THE TOP 5 MOST ORDERED PIZZA TYPES ALONG WITH THEIR QUANTITIES.

```
SELECT P.NAME, SUM(O.QUANTITY) AS ORDER_QUANTITY
FROM PIZZA_TYPES P
JOIN PIZZAS PI
ON P.PIZZA_TYPE_ID = PI.PIZZA_TYPE_ID
JOIN ORDER_DETAILS O
ON PI.PIZZA_ID = O.PIZZA_ID
GROUP BY P.NAME
ORDER BY ORDER_QUANTITY DESC
LIMIT 5;
```



6. JOIN THE NECESSARY TABLES TO FIND THE  
TOTAL QUANTITY OF EACH PIZZA CATEGORY  
ORDERED.

```
SELECT P.CATEGORY, SUM(O.QUANTITY)  
AS TOTAL_QUANTITY  
FROM PIZZA_TYPES P  
JOIN PIZZAS A  
ON P.PIZZA_TYPE_ID = A.PIZZA_TYPE_ID  
JOIN ORDER_DETAILS O  
ON A.PIZZA_ID = O.PIZZA_ID  
GROUP BY P.CATEGORY;
```



7. DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.

```
SELECT EXTRACT(HOUR FROM  
TIME) AS HOURS,  
COUNT(ORDER_ID) AS ORDERS  
FROM ORDERS  
GROUP BY HOURS  
ORDER BY HOURS;
```



8. JOIN RELEVANT TABLES TO FIND THE  
CATEGORY-WISE DISTRIBUTION OF  
PIZZAS.

```
SELECT CATEGORY, COUNT(NAME)  
AS VARIETY_OF_PIZZAS  
FROM PIZZA_TYPES  
GROUP BY CATEGORY  
ORDER BY CATEGORY;
```



9. GROUP THE ORDERS BY DATE  
AND CALCULATE THE AVERAGE NUMBER OF  
PIZZAS ORDERED PER DAY.

```
SELECT
ROUND(AVG(PER_DAY_ORDER),2) AS
PERDAYORDER FROM
(SELECT O.DATE, SUM(OD.QUANTITY)
AS PER_DAY_ORDER
FROM ORDERS O
JOIN ORDER_DETAILS OD
ON O.ORDER_ID = OD.ORDER_ID
GROUP BY O.DATE
ORDER BY O.DATE);
```



## 10. DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE.

```
SELECT PT.NAME, SUM(OD.QUANTITY *  
P.PRICE ) AS REVENUE  
FROM ORDER_DETAILS OD  
JOIN PIZZAS P  
ON OD.PIZZA_ID = P.PIZZA_ID  
JOIN PIZZA_TYPES PT  
ON P.PIZZA_TYPE_ID = PT.PIZZA_TYPE_ID  
GROUP BY PT.NAME  
ORDER BY REVENUE DESC  
LIMIT 3;
```



## 11. CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPE TO TOTAL REVENUE.

```
SELECT PT.NAME, CONCAT(ROUND(SUM(OD.QUANTITY *  
P.PRICE ) * 100/  
          (SELECT  SUM(OD.QUANTITY * P.PRICE ) AS  
TOTAL_REVENUE  
         FROM ORDER_DETAILS OD  
         JOIN PIZZAS P  
        ON OD.PIZZA_ID = P.PIZZA_ID),2), '%') AS  
REVENUE  
FROM ORDER_DETAILS OD  
JOIN PIZZAS P  
ON OD.PIZZA_ID = P.PIZZA_ID  
JOIN PIZZA_TYPES PT  
ON P.PIZZA_TYPE_ID = PT.PIZZA_TYPE_ID  
GROUP BY PT.NAME  
ORDER BY REVENUE DESC;
```



## 12. ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME.

```
SELECT DATE, SUM(REVENUE) OVER(ORDER BY DATE) AS CUMULATIVE_REVENUE
FROM
(SELECT O.DATE, SUM(OD.QUANTITY * P.PRICE) AS REVENUE
FROM ORDERS O
JOIN ORDER_DETAILS OD
ON O.ORDER_ID = OD.ORDER_ID
JOIN PIZZAS P
ON P.PIZZA_ID = OD.PIZZA_ID
GROUP BY O.DATE);
```



## 13. DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE FOR EACH PIZZA CATEGORY

```
SELECT CATEGORY , NAME , REVENUE , RANK  
FROM  
(SELECT CATEGORY , NAME , REVENUE ,  
RANK() OVER(PARTITION BY CATEGORY ORDER BY  
REVENUE)  
FROM  
(SELECT PT.CATEGORY, PT.NAME, SUM(OD.QUANTITY *  
P.PRICE) AS REVENUE  
FROM PIZZA_TYPES PT  
JOIN PIZZAS P  
ON PT.PIZZA_TYPE_ID = P.PIZZA_TYPE_ID  
JOIN ORDER_DETAILS OD  
ON P.PIZZA_ID = OD.PIZZA_ID  
GROUP BY PT.CATEGORY, PT.NAME))  
WHERE RANK <= 3;
```



THANK YOU  
FOR WATCHING