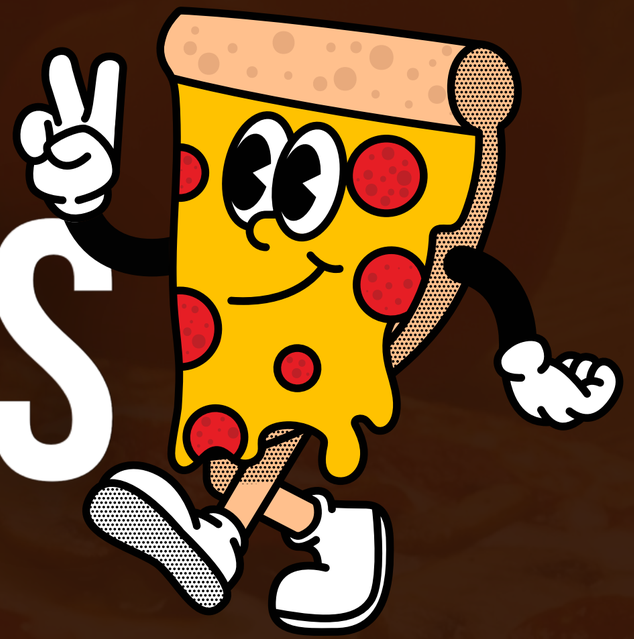
# PIZZAS SALES ANALYSIS




We have taken a pizza dataset as an example, where we will analyze the data under different conditions and situations using various SQL queries.

# Retrieve the total number of orders placed.

```
select count(order_id) as total_orders from orders;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| total_orders |
| --- |
| 21350 |

# Calculate the total revenue generated from pizza sales.

```sql
select * from orders;
select*from  pizzas;
select * from order_details;
 select (order_details.quantity* pizzas.price)
 from order_details join pizzas
 on pizzas.pizza_id=order_details.pizza_id;
 select round(sum(order_details.quantity* pizzas.price),2)
 as total_revenue
 from order_details join pizzas
 on pizzas.pizza_id=order_details.pizza_id;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| total_revenue |
| --- |
| 78242644.20 |

# Identify the highest-priced pizza.

```sql
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
LIMIT 1;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

| name | price |
|---|---|
| The Pepperoni Pizza | 9.75 |

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

```sql
select order_date,
sum(revenue) over(order by order_date) as cum_revenue
from
(select orders.order_date, sum(order_details.quantity*pizzas.pri
as revenue from order_details join pizzas
 on order_details.pizza_id = pizzas.pizza_id
 join orders
 on orders.order_id=order_details_id
 group by orders.order_date) as sales;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| order_date | cum_revenue |
|---|---|
| 2015-01-01 | 110480.99999999962 |
| 2015-01-02 | 216227.09999999928 |
| 2015-01-03 | 323551.49999999895 |
| 2015-01-04 | 410357.9999999988 |
| 2015-01-05 | 495586.19999999856 |

# Identify the most common pizza size ordered.

```sql
SELECT
    pizzas.size,
    COUNT(order_details.order_details_id) as order_counts
FROM
    pizzas
        JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY pizzas.size
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| size | order_counts |
|---|---|
| S | 1555840 |
| M | 1507220 |
| L | 1507220 |
| XL | 48620 |
| XXL | 48620 |

# Determine the distribution of orders by hour of the day.

```
select hour(order_time), count(order_id) from orders
group by hour(order_time);
```

Result Grid | Filter Rows: | Export: | Wrap Cell C

| hour(order_time) | count(order_id) |
|---|---|
| 9 | 1 |
| 10 | 8 |
| 11 | 1231 |
| 12 | 2520 |
| 13 | 2455 |

# Join the necessary tables to find the total quantity of each pizza category ordered.

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

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| quantity | category |
|---|---|
| 892332 | Chicken |
| 1239350 | Supreme |
| 1288924 | Classic |
| 1338498 | Veggie |

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

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

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| | cayegory | name |
|---|---|---|
| | 6 | The Barbecue Chicken Pizza |
| | 8 | The Big Meat Pizza |
| ▶ | 9 | The Brie Carre Pizza |
| | 9 | The Five Cheese Pizza |

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

```sql
SELECT
    ROUND(AVG(quantity), 0) as average_pizza_per_day
FROM
    (SELECT
        orders.order_date, SUM(order_details.quantity) as quantity
    FROM
        orders
    JOIN order_details ON orders.order_id = order_details.order_id
    GROUP BY orders.order_date) AS order_quantity;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

| average_pizza_per_day |
| --- |
| 138 |

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

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

sult Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

| name | revenue |
|---|---|
| The Greek Pizza | 5450661.3000048855 |
| The Italian Vegetables Pizza | 2503487 |
| The Barbecue Chicken Pizza | 2491093.5 |