Sales Analysis

PIZZA RESTAURANT

20% OFF

# Sql Project on

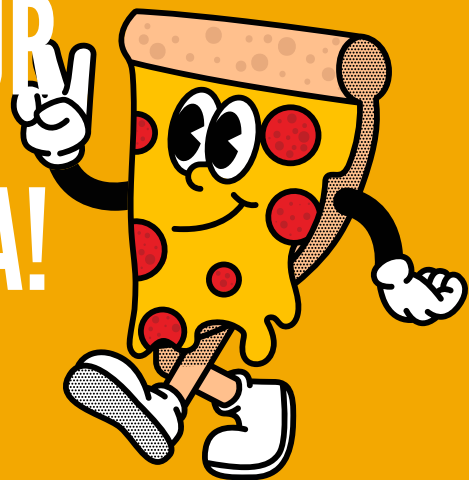
# PIZZA SALES

WATCH NOW

PIZZA RESTAURANT

WELCOME EVERYONE! MY NAME IS TEJAS SARODE

TODAY, WE'LL USE SQL TO ANALYZE PIZZA ORDER DATA AND DISCOVER WHAT MAKES OUR PIZZAS FLY OFF THE SHELVES! WE'LL EXPLORE CUSTOMER PREFERENCES, IDENTIFY TRENDS, AND GAIN INSIGHTS TO BOOST OUR PIZZA BUSINESS. LET'S GET SAUCY WITH SOME DATA!



# Q1 RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.

```sql
SELECT
    COUNT(order_id) AS total_orders
FROM
    orders;
```

Result Grid

| | total_orders |
|---|---|
| ▶ | 21350 |

# Q2 CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

```
SELECT
    ROUND(SUM(orders_details.quntity * 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 |

# Q3 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;
```

Result Grid | Filter Ro

| | name | price |
|---|---|---|
| ▶ | The Greek Pizza | 35.95 |

# Q4 IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.

```sql
SELECT
    pizzas.size,
    COUNT(orders_details.orders_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 limit 1;
```

Result Grid

| size | order_count |
|---|---|
| XXL | 28 |

# Q5 LIST THE TOP 5 MOST ORDERED PIZZA TYPES ALONG WITH THEIR QUANTITIES..

```sql
SELECT
    pizza_types.name, SUM(orders_details.quntity) 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 limit 5;
```

Result Grid | Filter Rows:

| name | quantity |
|---|---|
| The Brie Carre Pizza | 490 |
| The Mediterranean Pizza | 934 |
| The Calabrese Pizza | 937 |
| The Spinach Supreme Pizza | 950 |
| The Soppressata Pizza | 961 |

# Q6 JOIN THE NECESSARY TABLES TO FIND THE TOTAL QUANTITY OF EACH PIZZA CATEGORY ORDERED.

```
SELECT
    pizza_types.category,
    SUM(orders_details.quntity) 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;
```

Result Grid

| category | Quantity |
|---|---|
| Classic | 14888 |
| Supreme | 11987 |
| Veggie | 11649 |
| Chicken | 11050 |

# Q7 DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.

```sql
SELECT
    HOUR(order_time) AS hour, COUNT(order_id) AS Order_count
FROM
    orders
GROUP BY HOUR(order_time);
```

Result Grid | Filter

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

Result 1

# Q8 JOIN RELEVANT TABLES TO FIND THE CATEGORY-WISE DISTRIBUTION OF PIZZAS.

```
SELECT
    category, COUNT(name)
FROM
    pizza_types
GROUP BY category;
```

Result Grid | Filter Rows

| category | COUNT(name) |
|---|---|
| Chicken | 6 |
| Classic | 8 |
| Supreme | 9 |
| Veggie | 9 |

# Q9 GROUP THE ORDERS BY DATE AND CALCULATE THE AVERAGE NUMBER OF PIZZAS ORDERED PER DAY.

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

Result Grid | Filter Ro

| | ROUND(AVG(quantity), 0) |
|---|---|
| ▶ | 61 |

# Q10 DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE.

```
SELECT
    pizza_types.name, SUM(quntity * pizzas.price) AS revenue
FROM
    pizza_types
        JOIN
    pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
        JOIN
    orders_details ON orders_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY revenue desc limit 3;
```

Result Grid | Filter Rows:

| name | revenue |
|---|---|
| The Thai Chicken Pizza | 43434.25 |
| The Barbecue Chicken Pizza | 42768 |
| The California Chicken Pizza | 41409.5 |

# Q11 CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPE TO TOTAL REVENUE.

```
sELECT pizza_types.category,
    round(sum(orders_details.quntity * pizzas.price) / (SELECT
    ROUND(SUM(orders_details.quntity * pizzas.price),
            2) AS total_sales
FROM orders_details
        JOIN
    pizzas ON pizzas.pizza_id = orders_details.pizza_id)*100,2) as revenue
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 revenue desc ;
```

Result Grid

| category | revenue |
|---|---|
| Classic | 26.91 |
| Supreme | 25.46 |
| Chicken | 23.96 |
| Veggie | 23.68 |

# Q12 ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME.

```
select order_date ,
sum(revenue) over (order by order_date) as cum_revenue
from
(select orders.order_date,
sum(orders_details.quntity * pizzas.price) as revenue
from orders_details join pizzas
on orders_details.pizza_id = pizzas.pizza_id
join orders
on orders.order_id = orders_details.order_id
group by orders.order_date) as sales;
```

Result Grid  Filter Rows:

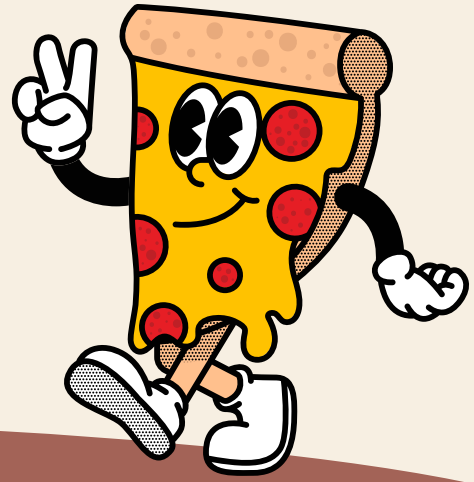
| order_date | cum_revenue |
|---|---|
| 2015-01-01 | 2713.8500000000004 |
| 2015-01-02 | 5445.75 |
| 2015-01-03 | 8108.15 |
| 2015-01-04 | 9863.6 |
| 2015-01-05 | 11929.55 |
| 2015-01-06 | 14358.5 |
| 2015-01-07 | 16560.7 |
| 2015-01-08 | 19399.05 |

Result 1

# Q13 DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE FOR EACH PIZZA CATEGORY.

```
select name , revenue  from
(select category,name, revenue,
rank() over (partition by category order by revenue desc ) as rn
from (select pizza_types.category, pizza_types.name,
sum(orders_details.quntity * pizzas.price)  as revenue
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,pizza_types.name) as a) as b
where rn <=3 ;
```

Result Grid | Filter Rows:

| name | revenue |
|---|---|
| The California Chicken Pizza | 41409.5 |
| The Classic Deluxe Pizza | 38180.5 |
| The Hawaiian Pizza | 32273.25 |
| The Pepperoni Pizza | 30161.75 |
| The Spicy Italian Pizza | 34831.25 |
| The Italian Supreme Pizza | 33476.75 |
| The Sicilian Pizza | 30940.5 |

# THANK YOU