



SQL Project on : Pizza Sales





Hello !

My name is Bishal Budhakshetri. In this project i have utilized SQL queries to solve questions that were related to pizza sales.





Questions:

Basic:

Retrieve the total number of orders placed.
Calculate the total revenue generated from pizza sales.
Identify the highest-priced pizza.
Identify the most common pizza size ordered.
List the top 5 most ordered pizza types along with their quantities.

Intermediate:

Join the necessary tables to find the total quantity of each pizza category ordered.
Determine the distribution of orders by hour of the day.
Join relevant tables to find the category-wise distribution of pizzas.
Group the orders by date and calculate the average number of pizzas ordered per day.
Determine the top 3 most ordered pizza types based on revenue.

Advanced:

Calculate the percentage contribution of each pizza type to total revenue.
Analyze the cumulative revenue generated over time.
Determine the top 3 most ordered pizza types based on revenue for each pizza category.

|





Retrieve the total number of order placed.

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

total_orders
21350





Calculate the total revenue generated from pizza sales.

```
SELECT  
    ROUND(SUM(order_details.quantity * pizzas.price),  
          2) AS total_revenue  
FROM  
    order_details  
    JOIN  
    pizzas ON order_details.pizza_id = pizzas.pizza_id;
```

total_revenue
817860.05







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







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
LIMIT 1;
```

size	order_count
L	18526



List the top 5 most ordered pizza types along with their quantities.

```
SELECT
    pizza_types.name, SUM(order_details.quantity) AS Total_count
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 Total_count DESC
LIMIT 5;
```

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





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

```
SELECT
    pizza_types.category,
    SUM(order_details.quantity) AS total_count
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.category
ORDER BY total_count DESC;
```

category	total_count
Supreme	446166
Veggie	446166
Classic	396592
Chicken	297444







Determine the distribution of orders by hour of the day.

```
SELECT
    HOUR(order_time) AS Hour, COUNT(order_id) AS order_count
FROM
    orders
GROUP BY Hour
ORDER BY order_count DESC;
```

Hour	order_count
12	2520
13	2455
18	2399
17	2336
19	2009
16	1920
20	1642
14	1472
15	1468
11	1231
21	1198
22	663
23	28
10	8
9	1





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

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

category	total_count
Chicken	6
Classic	8
Supreme	9
Veggie	9





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

```
SELECT
    ROUND(AVG(quantity), 0) AS Average_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;
```

Average_per_day

138





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

```
SELECT
    pizza_types.name,
    SUM(order_details.quantity * pizzas.price) 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;
```

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



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

```
SELECT
    pizza_types.category,
    ROUND(
        (SUM(order_details.quantity * pizzas.price) /
         (SELECT
             SUM(order_details.quantity * pizzas.price)
           FROM order_details
           JOIN pizzas
             ON pizzas.pizza_id = order_details.pizza_id) * 100),
        2
    ) 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.category
ORDER BY revenue DESC;
```

category	revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

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(order_details.quantity * pizzas.price) AS revenue
    FROM
        order_details
    JOIN
        pizzas
        ON order_details.pizza_id = pizzas.pizza_id
    JOIN
        orders
        ON orders.order_id = order_details.order_id
    GROUP BY
        orders.order_date
) AS sales;
```

order_date	cum_revenue
2015-01-01	2713.85000000000004
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
2015-01-09	21526.4
2015-01-10	23990.3500000000002
2015-01-11	25862.65
2015-01-12	27781.7
2015-01-13	29831.3000000000003
2015-01-14	32358.7000000000004
2015-01-15	34343.500000000001
2015-01-16	36937.650000000001

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(order_details.quantity * pizzas.price) 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.category,
            pizza_types.name
    ) AS a
    ) AS b
WHERE
    rn <= 3;
```

name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
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
The Four Cheese Pizza	32265.700000000065
The Mexicana Pizza	26780.75
The Five Cheese Pizza	26066.5



Thank You !

