



Pizza Sales Analysis

With SQL | MR. Mehar

Basic | Intermediate | Advanced





Objective

The objective of this pizza sales data analysis project is to utilize Structured Query Language (SQL) to gain insights into various aspects of the pizza sales business, such as popular toppings, busiest days of the week, and highest-selling pizza sizes and crust types. The results can be used to inform marketing and inventory decisions, as well as improve overall business operations.

Tables To Be Served

1. Pizzas
2. Pizza_types
3. Orders
4. Order_details





Basic



1. Retrieve The Total Number Of Orders Placed:

SQL Query : -----



```
SELECT
    COUNT(order_id) AS Total_Order_Placed
FROM
    orders
```

Result : -----

	Total_Order_Placed
▶	21350



2. Calculate The Total Revenue Generated From Pizza Sales:



SQL Query : -----

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

Result : -----

	Total_Revnue
▶	972956

3. Identify The Highest Priced Pizza:

SQL Query : -----

```
SELECT
    pizza_types.name AS highest_priced_Pizza, 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 : -----

	highest_priced_Pizza	price
▶	The Greek Pizza	35.95



4. Identify The Most Common Pizza Size Ordered:



SQL Query :

```
SELECT
    pizzas.size,
    SUM(order_details.quantity) AS
    No_of_Orders
FROM
    pizzas
    JOIN
    order_details ON pizzas.pizza_id =
    order_details.pizza_id
GROUP BY pizzas.size
ORDER BY SUM(order_details.quantity) DESC;
```

Result :

	size	No_of_Orders
▶	L	22560
	M	18622
	S	17020
	XL	688
	XXL	36

5. List The Top 5 Most Ordered Pizza Types Along With Their Quantities:



SQL Query :

```
SELECT
    pizza_types.name AS most_ordered_pizza,
    SUM(order_details.quantity) AS Quantity
FROM
    pizza_types
    JOIN
    pizzas ON pizza_types.pizza_type_id =
pizzas.pizza_type_id
    JOIN
    order_details ON pizzas.pizza_id =
order_details.pizza_id
GROUP BY pizza_types.name
ORDER BY SUM(order_details.quantity) DESC
LIMIT 5;
```

Result :

	most_ordered_pizza	Quantity
▶	The Barbecue Chicken Pizza	2954
	The Pepperoni Pizza	2867
	The Classic Deluxe Pizza	2857
	The Hawaiian Pizza	2824
	The California Chicken Pizza	2821



Intermediate

6. Join The Necessary Tables To Find The Total Quantity Of Each Pizza Category Ordered:



SQL Query :

```
SELECT
    pizza_types.category AS Pizza_Category,
    SUM(order_details.quantity) AS Quantity_Ordered
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 Quantity_Ordered DESC;
```

Result :

	Pizza_Category	Quantity_Ordered
▶	Classic	17614
	Supreme	14291
	Veggie	13968
	Chicken	13053

7. Determine The Distribution Of Orders By Hour Of The Day:



SQL Query :



```
SELECT
    HOUR(orders.time) AS Time_Hour,
    COUNT(orders.order_id) AS Orders
FROM
    orders
GROUP BY Time_Hour
ORDER BY Orders DESC;
```

Result :

	Time_Hour	Orders
▶	12	2520
	13	2455
	18	2399
	17	2336
	19	2009
	16	1920
	20	1642
	14	1472
	15	1468
	11	1231
	21	1198
	22	663

9. Group The Orders By Date And Calculate The Average Number Of Pizzas Ordered Per Day:

SQL Query : -----

```
SELECT
    ROUND(AVG(Quantity), 0) AS Average_Orders_Per_Day
FROM
    (SELECT
        orders.date, SUM(order_details.quantity) AS Quantity
    FROM
        orders
    JOIN order_details ON orders.order_id = order_details.order_id
    GROUP BY date) AS orders_quantity;
```

Result : -----

	Average_Orders_Per_Day
▶	245



10. Determine The Top 5 Most Ordered Pizza Types Based On Revenue:



SQL Query :

```
SELECT
    pizza_types.name,
    (order_details.quantity * pizzas.price) AS Revnue
FROM
    order_details
    JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id
    JOIN
    pizza_types ON pizzas.pizza_type_id =
    pizza_types.pizza_type_id
ORDER BY Revnue DESC
LIMIT 5;
```

Result :

	name	Revnue
▶	The Thai Chicken Pizza	62.25
	The Spicy Italian Pizza	62.25
	The Spicy Italian Pizza	62.25
	The Thai Chicken Pizza	62.25
	The Greek Pizza	51



Advance

11. Calculate The Percentage Contribution Of Each Pizza Type To Total Revenue:



SQL Query :

```
SELECT
    pizza_types.category AS Pizza_Type,
    ROUND((SUM(order_details.quantity * pizzas.price) / (SELECT
        ROUND(SUM(order_details.quantity * pizzas.price),
            0) AS Total_Revenue
    FROM
        order_details
        JOIN
        pizzas ON order_details.pizza_id = pizzas.pizza_id) * 100),
    2) AS Revenue_Percentage
FROM
    pizza_types
    JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
    JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY Pizza_Type
ORDER BY Revenue_Percentage DESC;
```

Result :

	Pizza_Type	Revenue_Percentage
▶	Classic	26.79
	Supreme	25.54
	Veggie	23.88
	Chicken	23.78

12. Analyze The Cumulative Revenue Generated Over Time:



SQL Query :

```
SELECT order_date,  
sum(Revenue) over (order by order_date) as Cum_Revenue  
from (SELECT  
    orders.date AS Order_Date,  
    ROUND(SUM(order_details.quantity * pizzas.price),  
        2) AS Revenue  
FROM  
    orders  
    JOIN  
    order_details ON orders.order_id =  
    order_details.order_id  
    JOIN  
    pizzas ON pizzas.pizza_id = order_details.pizza_id  
GROUP BY Order_Date  
ORDER BY Revenue) as sales;
```

Result :

	order_date	Cum_Revenue
▶	2015-01-01	5427.7
	2015-01-02	10891.5
	2015-01-03	16216.3
	2015-01-04	19727.2
	2015-01-05	23859.1
	2015-01-06	28717
	2015-01-07	33121.4
	2015-01-08	38798.1

13. Determine The Top 3 Most Ordered Pizza Types Based On Revenue For Each Pizza Category:

SQL Query :

```
Select name, Revenue from
(select category, name, Revenue,
rank() over(partition by category order by Revenue desc) as
Category_Rank
from(
select pizza_types.category, pizza_types.name,
round(sum(order_details.quantity * pizzas.price),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, pizza_types.name) as sales)
as Rank_Data
where Category_Rank <= 3;
```

Result :

	category	name	Revenue	Category_Rank
▶	Chicken	The Barbecue Chicken Pizza	51971.5	1
	Chicken	The Thai Chicken Pizza	50165.25	2
	Chicken	The California Chicken Pizza	49359.75	3
	Chicken	The Southwest Chicken Pizza	40392.5	4
	Chicken	The Chicken Alfredo Pizza	19820.5	5
	Chicken	The Chicken Pesto Pizza	19680.25	6
	Classic	The Classic Deluxe Pizza	44408	1
	Classic	The Hawaiian Pizza	37682.25	2
	Classic	The Pepporoni Pizza	35807.25	3

	name	Revenue
▶	The Barbecue Chicken Pizza	51971.5
	The Thai Chicken Pizza	50165.25
	The California Chicken Pizza	49359.75
	The Classic Deluxe Pizza	44408
	The Hawaiian Pizza	37682.25
	The Pepperoni Pizza	35807.25
	The Spicy Italian Pizza	42064
	The Italian Supreme Pizza	40560
	The Sicilian Pizza	36132.25





THANK YOU!

Mr. Mehar |  