



**Data analysis of orders data of Borcelle  
Pizza's by MS SQL Server**

# Introduction

- **Project focus:** Analyzing the order data to find the answers to some key metrics which will help to take better business decisions
- **Primary purpose of the project**
  - ❖ Find the best performing pizzas
  - ❖ Find the hours where more pizzas being sold
  - ❖ Find the pizzas that are contribution to the company revenue
  - ❖ Take better business decisions based on the answered questions



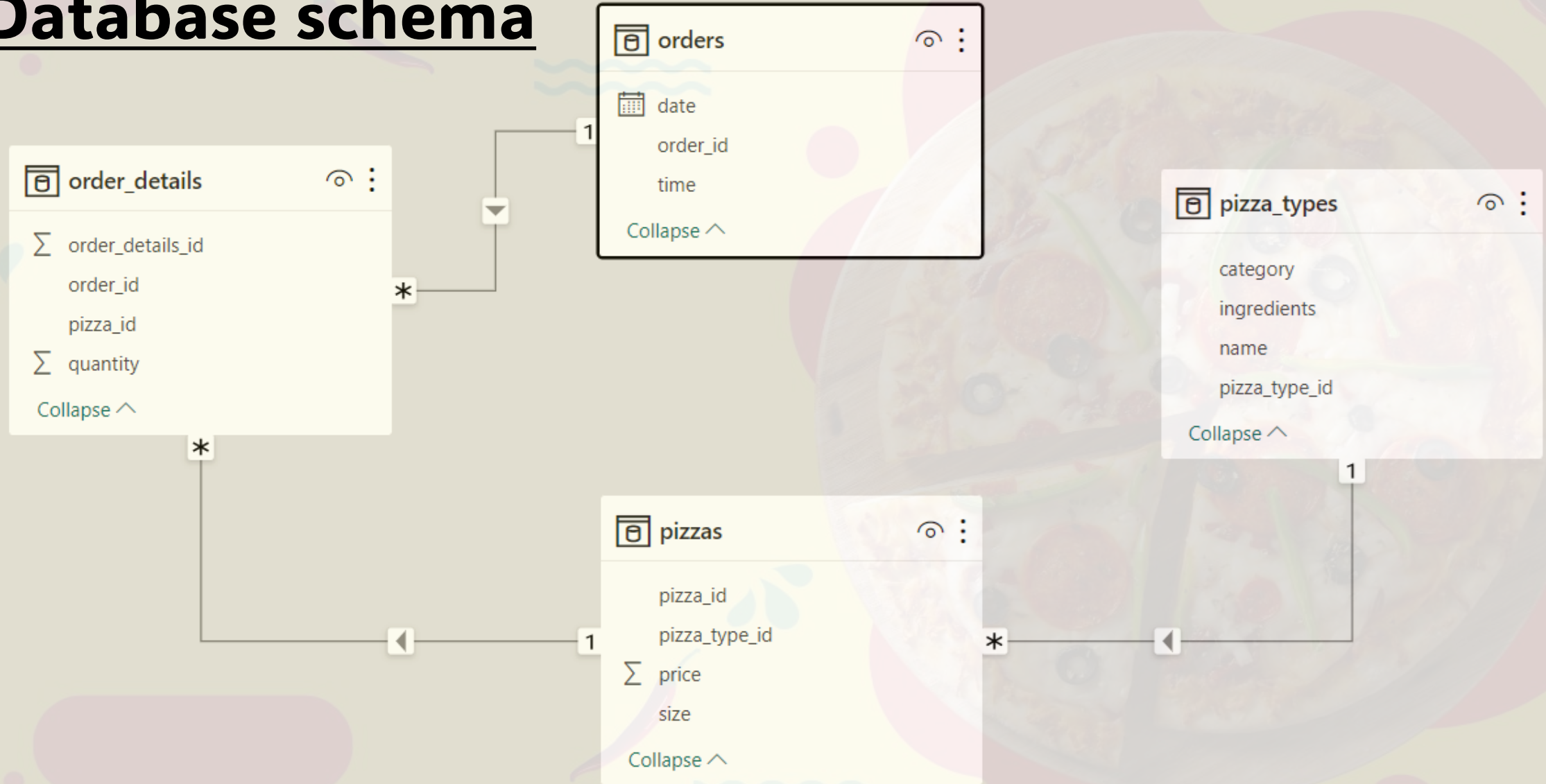
# DATABASE OVERVIEW

- Pizzas Table – (🔑 pizza\_id, pizza\_type\_id, size, price)
- Orders\_details Table – (🔑 order\_id, orders\_details\_id, pizza\_id, quantity)
- Orders Table – (🔑 order\_id, date, time)
- Pizza\_types Table – (🔑 pizza\_type\_id, name, category, ingredients)





# Database schema



# Retrieve the total number of orders placed

```
/*Retrieve the total number of orders placed*/  
  
SELECT count([order_id]) as Total_number_of_orders  
FROM [orders]
```

Results		Messages	
		Total_number_of_orders	
1	21350		

# Calculate the total revenue generated from pizza sales

```
/*Calculate the total revenue generated from pizza sales*/  
  
SELECT  
    round(sum([order_details].[quantity] * [pizzas].[price]), 2) as total_price  
FROM  
    [pizzas] JOIN [order_details]  
ON  
    [pizzas].[pizza_id] = [order_details].[pizza_id]  
JOIN  
    [orders]  
ON  
    [order_details].[order_id] = [orders].[order_id]
```

Results		Messages
	total_price	
1	817860.05	

# Identify the highest-priced pizza

```
/*Identify the highest-priced pizza*/
```

```
SELECT
    TOP 1 [pizzas].[price] as Max_price, [pizza_types].[name] as Pizza_name
FROM
    [pizzas]
JOIN
    [pizza_types]
ON
    [pizzas].[pizza_type_id] = [pizza_types].[pizza_type_id]
ORDER BY
    Max_price DESC
```

Results			Messages	
	Max_price	Pizza_name		
1	35.95	The Greek Pizza		



# Identify the most common pizza size ordered

```
/*Identify the most common pizza size ordered*/

SELECT
    TOP 1 count([order_details].[quantity]) as quantity_ordered
    ,[pizzas].[size]
FROM
    [orders]
JOIN
    [order_details]
ON
    [orders].[order_id] = [order_details].[order_id]
JOIN
    [pizzas]
ON
    [order_details].[pizza_id] = [pizzas].[pizza_id]
GROUP BY
    [pizzas].[size]
ORDER BY
    quantity_ordered DESC
```

Results			Messages	
	quantity_ordered	size		
1	18526	L		



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

```
/*List the top 5 most ordered pizza types along with their quantities*/
```

```
SELECT
    TOP 5 count([order_details].[quantity]) as quantity_ordered
    , [pizza_types].[name]
FROM
    [orders]
JOIN
    [order_details]
ON
    [orders].[order_id] = [order_details].[order_id]
JOIN
    [pizzas]
ON
    [order_details].[pizza_id] = [pizzas].[pizza_id]
JOIN
    [pizza_types]
ON
    [pizzas].[pizza_type_id] = [pizza_types].[pizza_type_id]
GROUP BY
    [pizza_types].[name]
ORDER BY
    quantity_ordered DESC
```

Results Messages		
	quantity_ordered	name
1	2416	The Classic Deluxe Pizza
2	2372	The Barbecue Chicken Pizza
3	2370	The Hawaiian Pizza
4	2369	The Pepperoni Pizza
5	2315	The Thai Chicken Pizza

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

```
/*Join the necessary tables to  
find the total quantity of each pizza category ordered*/  
  
SELECT  
    sum([order_details].[quantity]) as quantity  
    , [pizza_types].[category]  
FROM  
    [order_details]  
JOIN  
    [orders]  
ON  
    [order_details].[order_id] = [orders].[order_id]  
JOIN  
    [pizzas]  
ON  
    [order_details].[pizza_id] = [pizzas].[pizza_id]  
JOIN  
    [pizza_types]  
ON  
    [pizzas].[pizza_type_id] = [pizza_types].[pizza_type_id]  
GROUP BY  
    [pizza_types].[category]  
ORDER BY  
    quantity DESC
```

Results		Messages
	quantity	category
1	446166	Supreme
2	446166	Veggie
3	396592	Classic
4	297444	Chicken

# Determine the distribution of orders by hour of the day

```
/*Determine the distribution of orders by hour of the day*/
```

```
SELECT
    DATEPART(hour, time) as hour_of_day
    ,count([order_id]) as No_of_orders
FROM
    [orders]
GROUP BY
    DATEPART(hour, time)
ORDER BY
    hour_of_day
```

	hour_of_day	No_of_orders
1	9	1
2	10	8
3	11	1231
4	12	2520
5	13	2455
6	14	1472
7	15	1468
8	16	1920
9	17	2336
10	18	2399
11	19	2009
12	20	1642
13	21	1198
14	22	663
15	23	28



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

```
/*Join relevant tables to find the category-wise distribution of pizzas*/
```

```
SELECT
    count([orders].[order_id]) as pizza_count
    , [pizza_types].[category]
FROM
    [orders]
JOIN
    [order_details]
ON
    [orders].[order_id] = [order_details].[order_id]
JOIN
    [pizzas]
ON
    [order_details].[pizza_id] = [pizzas].[pizza_id]
JOIN
    [pizza_types]
ON
    [pizzas].[pizza_type_id] = [pizza_types].[pizza_type_id]
GROUP BY
    [pizza_types].[category]
```

Results			Messages		
		pizza_count			category
1		10815			Chicken
2		14579			Classic
3		11777			Supreme
4		11449			Veggie

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

```
/*Group the orders by date and calculate the average number of pizzas
ordered per day*/

SELECT
    sum(orders_per_day)/count(orders_per_day) as average_number_of_orders
FROM(
    SELECT
        [orders].[date]
        ,count([order_details].[quantity]) as orders_per_day
    FROM
        [orders]
    JOIN
        [order_details]
    ON
        [orders].[order_id] = [order_details].[order_id]
    GROUP BY
        [orders].[date]) as daily_orders
```

Results		Messages
average_number_of_orders		
1	135	

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

```
-- Determine the top 3 most ordered pizza types based on revenue

SELECT
    TOP 3 SUM([order_details].[quantity] * [pizzas].[price]) as revenue
    , [pizza_types].[name]
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]
GROUP BY
    [pizza_types].[name]
ORDER BY
    [revenue] DESC
```

Results			Messages		
	revenue	name			
1	43434.25	The Thai Chicken Pizza			
2	42768	The Barbecue Chicken Pizza			
3	41409.5	The California Chicken Pizza			



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

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

```
SELECT (ROUND(SUM([order_details].[quantity] * [pizzas].[price]), 2) / (SELECT ROUND(SUM([order_details].[quantity] * [pizzas].[price]), 2) as Revenue
FROM [order_details]
JOIN [pizzas]
ON [order_details].[pizza_id] = [pizzas].[pizza_id]*100) as Revenue_contribution
, [pizza_types].[name]
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]
GROUP BY
[pizza_types].[name]
ORDER BY
Revenue_contribution DESC
```

Results Messages		
	Revenue_contribution	name
1	5.31071911386306	The Thai Chicken Pizza
2	5.22925652133271	The Barbecue Chicken Pizza
3	5.06315230827083	The California Chicken Pizza
4	4.66834148458529	The Classic Deluxe Pizza
5	4.25882765639427	The Spicy Italian Pizza
6	4.24348273277317	The Southwest Chicken Pizza
7	4.09321252456334	The Italian Supreme Pizza
8	3.94606020920083	The Hawaiian Pizza
9	3.94513706837741	The Four Cheese Pizza
10	3.7831044565632	The Sicilian Pizza
11	3.68788645441234	The Pepperoni Pizza
12	3.47909156340379	The Greek Pizza
13	3.27449054395064	The Mexicana Pizza
14	3.18715897664888	The Five Cheese Pizza
15	3.12143868623978	The Pepper Salami Pizza
16	3.06825110237381	The Italian Capocollo Pizza
17	2.98030818353336	The Vegetables + Vegetable...
18	2.95811612267894	The Prosciutto and Arugula ...
19	2.94512490248179	The Napolitana Pizza
20	2.84538289894463	The Spinach and Feta Pizza
21	2.80830442812312	The Big Meat Pizza
22	2.30290011108869	The Pepperoni, Mushroom, ...

# Analyze the cumulative revenue generated over time

-- Analyze the cumulative revenue generated over time

```
SELECT
    [orders].[date] as date,
    SUM(SUM([order_details].[quantity] * [pizzas].[price])) OVER (ORDER BY date) as cummulative_revenue
FROM
    [orders]
JOIN
    [order_details]
ON
    [orders].[order_id] = [order_details].[order_id]
JOIN
    [pizzas]
ON
    [order_details].[pizza_id] = [pizzas].[pizza_id]
GROUP BY
    [orders].[date]
ORDER BY
    [orders].[date]
```

Results			Messages	
	date	cummulative_revenue		
1	2015-01-01	2713.85		
2	2015-01-02	5445.75		
3	2015-01-03	8108.15		
4	2015-01-04	9863.6		
5	2015-01-05	11929.55		
6	2015-01-06	14358.5		
7	2015-01-07	16560.7		
8	2015-01-08	19399.05		
9	2015-01-09	21526.4		
10	2015-01-10	23990.35		
11	2015-01-11	25862.65		
12	2015-01-12	27781.7		
13	2015-01-13	29831.3		
14	2015-01-14	32358.7		
15	2015-01-15	34343.5		
16	2015-01-16	36937.65		
17	2015-01-17	39001.75		
18	2015-01-18	40978.6		
19	2015-01-19	43365.75		
20	2015-01-20	45763.65		
21	2015-01-21	47804.2		
22	2015-01-22	50300.9		
23	2015-01-23	52724.6		

# Determine the top 3 most ordered pizza types based on revenue for each pizza category

-- Determine the top 3 most ordered pizza types based on revenue for each pizza category

```
SELECT
    [Revenue]
    , [Name]
    , [Category]
    , [category_rank]
FROM
    (SELECT
        [Revenue]
        , [Name]
        , [Category]
        , RANK() OVER (PARTITION BY Category ORDER BY Name) as [category_rank]
    FROM
        (SELECT
            SUM([order_details].[quantity] * [pizzas].[price]) as revenue
            , [pizza_types].[name]
            , [pizza_types].[category]
        FROM
            [order_details]
        JOIN
            [pizzas]
        ON
            [order_details].[pizza_id] = [pizzas].[pizza_id]
        JOIN
            [pizza_types]
        ON
            [pizza_types].[pizza_type_id] = [pizzas].[pizza_type_id]
        GROUP BY
            [pizza_types].[name]
            , [pizza_types].[category]) AS Base_table) AS Ranked_table
WHERE
    [category_rank] <= 3
```

Results Messages				
	Revenue	Name	Category	category_rank
1	42768	The Barbecue Chicken Pizza	Chicken	1
2	41409.5	The California Chicken Pizza	Chicken	2
3	16900.25	The Chicken Alfredo Pizza	Chicken	3
4	22968	The Big Meat Pizza	Classic	1
5	38180.5	The Classic Deluxe Pizza	Classic	2
6	28454.1	The Greek Pizza	Classic	3
7	11588.4...	The Brie Carre Pizza	Supre...	1
8	15934.25	The Calabrese Pizza	Supre...	2
9	33476.75	The Italian Supreme Pizza	Supre...	3
10	26066.5	The Five Cheese Pizza	Veggie	1
11	32265.7...	The Four Cheese Pizza	Veggie	2
12	13955.75	The Green Garden Pizza	Veggie	3



# Conclusion

We have completed the project on Pizza Sales here, we have answered the questions which will help the company take better decisions

We have written easier queries in the beginning and those became more and more advanced as we moved forward. We have used multiple JOINS, advanced SQL functions, etc. to complete this project