



# SQL pizza sales project

By Umaima Abdul Qadir

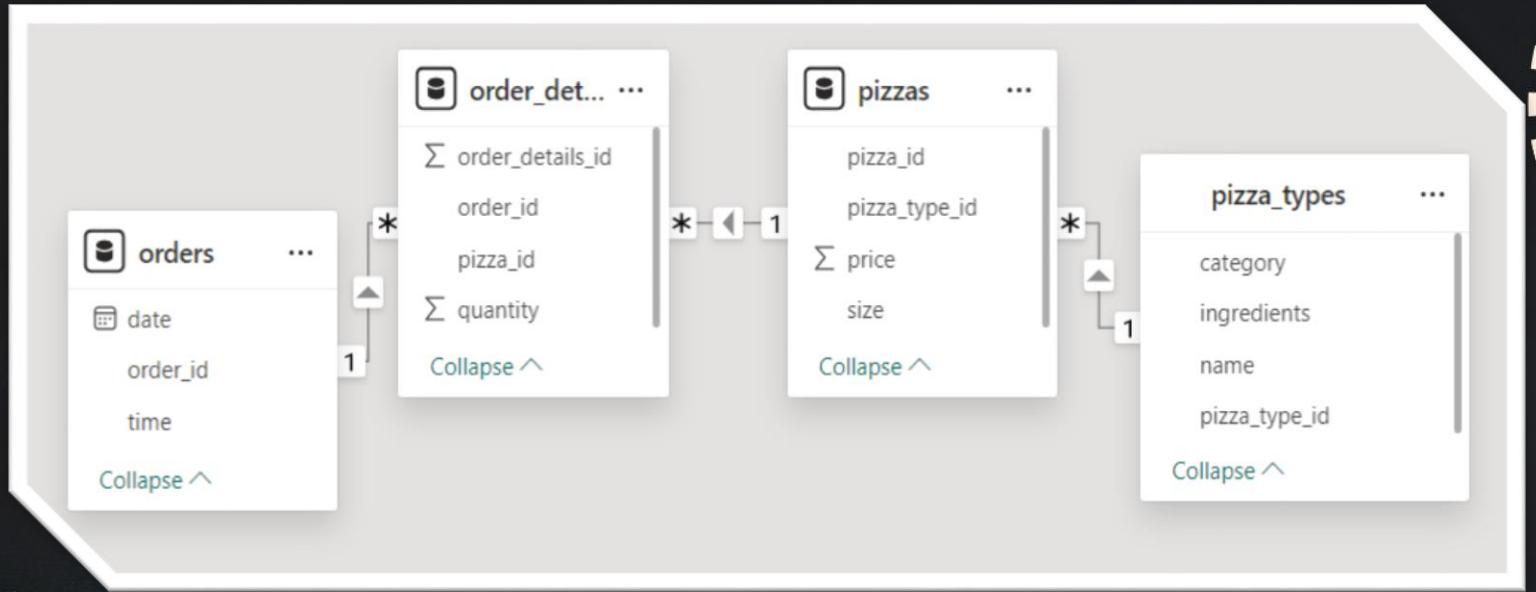




# Insights

- 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.
- 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.
- 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.

# Model



# 1. Retrieve the total number of orders placed.

```
-- Retreive Total orders

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

110 %

Results Messages

Total_orders
1 21350



## 2. Calculate the total revenue generated from pizza sales.

```
-- retreive total_revenue  
--  
  
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;
```

110 %

	Total_revenue
1	817860.05



### 3. Identify the highest-priced pizza.

```
-- Identify the highest pizza price
```

```
SELECT TOP 1 pizza_id, round(MAX(price),2) AS highest_price  
FROM pizzas  
GROUP BY pizza_id  
ORDER BY highest_price DESC;
```

110 %

Results Messages

pizza_id	highest_price
the_greek_xxl	35.95



#### 4. Identify the most common pizza size ordered.

----Identify the most common pizza size ordered.

```
select top 1 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;
```

110 %

Results Messages

size	order_count
L	18526

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

```
-- list the top 5 most ordered pizza types along with their quantity

SELECT
    top 5 pt.name,
    SUM(od.quantity) AS total_quantity
FROM order_details od
JOIN pizzas p ON od.pizza_id = p.pizza_id
JOIN pizza_types pt ON p.pizza_type_id = pt.pizza_type_id
GROUP BY pt.name
ORDER BY total_quantity DESC;
```

121 %

Results Messages

	name	total_quantity
1	The Classic Deluxe Pizza	2453
2	The Barbecue Chicken Pizza	2432
3	The Hawaiian Pizza	2422
4	The Pepperoni Pizza	2418
5	The Thai Chicken Pizza	2371

6. 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 pizza_types.category ,  
sum(order_details.quantity) as Total_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.category  
order by Total_quantity desc;
```



A decorative image of a pizza slice with various toppings like pepperoni, olives, and cheese is positioned in the top right corner of the slide.

	category	Total_quantity
1	Classic	14888
2	Supreme	11987
3	Veggie	11649
4	Chicken	11050

## 7. 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 order_hour ,  
count(order_id) as orders  
from orders  
group by DATEPART(hour, time)  
order by orders desc ;
```



A screenshot of a SQL query results window titled "Results". The table has two columns: "order\_hour" and "orders". The data shows the distribution of orders by hour of the day, with the highest volume at hour 12 (2520 orders) and the lowest at hour 9 (1 order).

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

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

Wyn text

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

```
select category , count(category) as cat_count
from pizza_types
group by category
order by cat_count desc;
```

146 %

Results Messages

	category	cat_count
1	Supreme	9
2	Veggie	9
3	Classic	8
4	Chicken	6

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

*Replaced with your own text*

```
-- Group the orders by date and calculate the average number of pizzas ordered per day.  
  
select avg(no_pizzas_order) as avg_pizzas_order_per_day  
from  
(select orders.date , sum(order_details.quantity) as no_pizzas_order  
from orders  
join order_details  
on orders.order_id = order_details.order_id  
group by date) as order_quantity;
```

146 %

	avg_pizzas_order_per_day
1	138



## 10. 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 pizza_types.name ,  
sum(order_details.quantity * pizzas.price) as revenue  
from order_details  
join pizzas  
on pizzas.pizza_id = order_details.pizza_id  
join pizza_types  
on pizzas.pizza_type_id = pizza_types.pizza_type_id  
group by pizza_types.name  
order by revenue desc;
```

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



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

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

select pizza_types.category,
round( sum(order_details.quantity * pizzas.price) / (
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) *100) ,2 )as revenue
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.category
order by revenue desc;
```

121 %

Results Messages

	category	revenue
1	Classic	26.91
2	Supreme	25.46
3	Chicken	23.96
4	Veggie	23.68



## 12. Analyze the cumulative revenue generated over time.

--Analyze the cumulative revenue generated over time.

```
select date ,round(sum(revenue) over(order by date),2) as cumulative_revenue
from
(select orders.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 order_details.order_id = orders.order_id
group by orders.date) as daily_sales;
```

	date	cumulative_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



# 13. 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 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 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.category , pizza_types.name ) as a) as b  
where rn<= 3;
```



	name	revenue
1	The Thai Chicken Pizza	43434.25
2	The Barbecue Chicken Pizza	42768
3	The California Chicken Pizza	41409.5
4	The Classic Deluxe Pizza	38180.5
5	The Hawaiian Pizza	32273.25
6	The Pepperoni Pizza	30161.75
7	The Spicy Italian Pizza	34831.25
8	The Italian Supreme Pizza	33476.75
9	The Sicilian Pizza	30940.5
10	The Four Cheese Pizza	32265.7010040283
11	The Mexicana Pizza	26780.75
12	The Five Cheese Pizza	26066.5

➤ *For more content like this , visit my Github.*

➤ *For collaboration purposes , email me at*

*umaimaaqadir@gmail.com*





THANK  
YOU