



# Pizza Revenue Generation Findings in SQL

PREPARED BY :-

NILIMA PANDYA ( DATA ANALYST)

PROJECT IN SQL



# Project overview

- ▶ Analyzing sales revenue trends as per category and dates
- ▶ Optimizing sales strategies as per the given data
- ▶ Get detailed analysis on the revenue generated to determine sales optimization strategies

# Data Collection

Data Source: I used Kaggle to fetch the dataset

4 Tables :

1. Pizzas,

2. Pizza\_types ,

3. Orders and

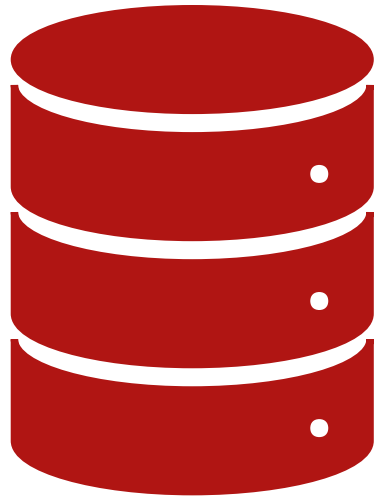
4. Order\_details

Tools Used : MySQL



# Table columns details

1. Pizzas Table	2. Order_details Table	3. Pizza_types	4.orders
Columns	Columns	Columns	Columns
Pizza_id,	order_details_id	Pizza_type_id	Order_id
Pizza_type_id,	Order_id	Name	Date
Size	Pizza_id	Category	Time
Price	Quantity	Ingredients	



# SQL Queries

# 1 : Analyze the cumulative revenue generate over time

```
select order_date, sum(revenue) over(order by order_date) as cumm_rev
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;
```

## Result Grid:

order_date	cumm_rev
2015-01-09	21526.4
2015-01-10	23990.350000000002
2015-01-11	25862.65
2015-01-12	27781.7
2015-01-13	29831.300 29831.300000000003
2015-01-14	32358.700000000004
2015-01-15	34343.500000000001
2015-01-16	36937.650000000001
2015-01-17	39001.750000000001
2015-01-18	40978.600000000006
2015-01-19	43365.750000000001
2015-01-20	45763.650000000001
2015-01-21	47804.200000000001

### 3 : Determine 3 most ordered pizza based on the revenue

```
Select pizza_types.name, sum(order_details.quantity * pizzas.price) as total_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 1
order by total_revenue desc
limit 3;
```

#### Result Grid:

Result Grid			Filter Rows:	Export:	Wrap Cell Content:
	name	total_revenue			
▶	The Thai Chicken Pizza	43434.25			
	The Barbecue Chicken Pizza	42768			
	The California Chicken Pizza	41409.5			

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

```
select pizza_types.category,  
       (sum(order_details.quantity*pizzas.price) /  
       (select sum(order_details.quantity*pizzas.price)  
        from order_details join pizzas  
        on order_details.pizza_id = pizzas.pizza_id) * 100 ) 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;
```

### Result Grid:

Result Grid	Filter Rows:
category	revenue
Classic	26.905960255669903
Supreme	25.45631126009884
Chicken	23.955137556847493
Veggie	23.682590927384783



4: group the orders by date and calculate avg number of pizzas ordered per day

```
• SELECT
    ROUND(AVG(quantity), 0)
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 pizza_quantity;
```

**Result Grid:**

Result Grid		Filter Rows:
	AVG_Pizza_ordered	
▶	138	

## 5: join the relevant tables for category wise distribution of pizzas

```
• SELECT
    category, COUNT(name)
FROM
    pizza_types
GROUP BY 1;
```

### Result Grid:

Result Grid			Filter Rows:
	category	COUNT(name)	
▶	Chicken	6	
	Classic	8	
	Supreme	9	
	Veggie	9	

## 6: Determine the distribution of orders by the hour of the day

```
SELECT
    HOUR(order_time) as hour , COUNT(order_id) as order_count
FROM
    orders
GROUP BY HOUR(order_time);
```

### Result Grid:

Result Grid		
Filter Rows:		
	hour	order_count
▶	11	1231
	12	2520
	13	2455
	14	1472
	15	1468
	16	1920
	17	2336
	18	2399
	19	2009
	20	1642
	21	1198
	22	663
	23	28

7: 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 order_details.pizza_id = pizzas.pizza_id
group by pizza_types.category;
```

Result Grid:

Result Grid			Filter Rows:
	category	Total_quantity	
▶	Classic	14888	
	Veggie	11649	
	Supreme	11987	
	Chicken	11050	

8: list top 5 most common pizza types along with their quantities

```
SELECT
    pizza_types.name,
    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 order_details.pizza_id = pizzas.pizza_id
GROUP BY 1
ORDER BY total_quantity DESC
LIMIT 5;
```

## Result Grid:

Result Grid			Filter Rows:
	name	total_quantity	
▶	The Classic Deluxe Pizza	2453	
	The Barbecue Chicken Pizza	2432	
	The Hawaiian Pizza	2422	
	The Pepperoni Pizza	2418	
	The Thai Chicken Pizza	2371	



## 9: Identify the most commonly ordered pizza size

```
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 1
ORDER BY order_count DESC
LIMIT 1;
```

### Result Grid:

Result Grid		
	size	order_count
▶	L	18526

## 10: Identify the highest price 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:

Result Grid		
	name	price
▶	The Greek Pizza	35.95

## 11: Calculate total revenue generated from pizza sales

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

### Result Grid:

Result Grid		Filter Rows:
	Total_sales	
▶	817860.05	

12: Retrieve the total number of orders placed

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

Result Grid:

Result Grid		Filter
	Total_orders	
▶	21350	

# Conclusion and key findings



- ▶ 1.) Five common most pizzas ordered are
  - Classic Deluxe pizza, Barbecue chicken pizza, The Hawaiian Pizza, The Pepperoni pizza, The Thai Chicken Pizza
- 2) Classic pizza has been ordered the most
- 3) Average of 138 pizzas have been ordered per day
- 4) The Thai Chicken pizza has generated highest revenue amongst all pizzas, which is 43434.25
- 5) Total Revenue generated from the Pizza sales is : 817860.05
- 6) Highest Priced Pizza is Greek pizza with the prize of 35.95
- 7) Most commonly ordered pizza is “L” sized pizza