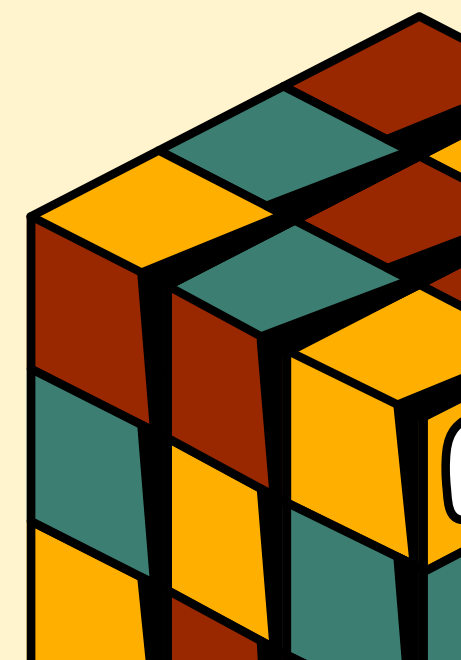
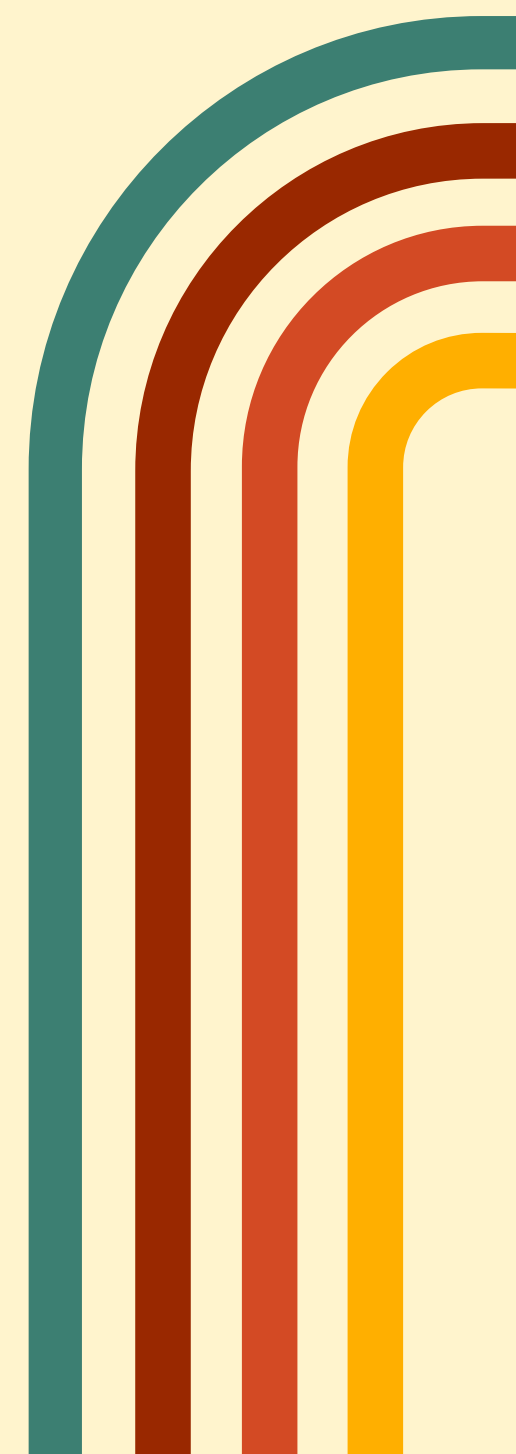
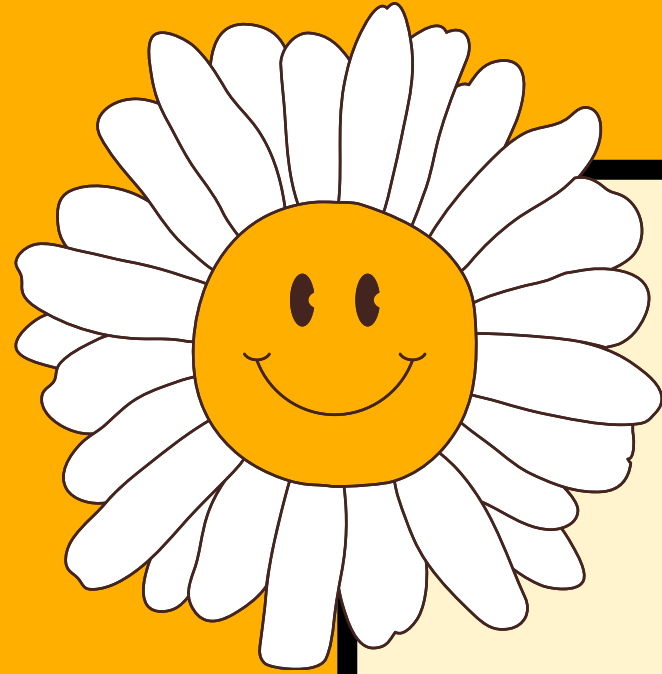


# Pizza Sales Report





# Hello!

**My Name is Srishti jaiswal.I have done  
BCA. In this project I have utilize sql  
queries to solve questions related to  
pizza sales.we use joins,group by,order  
by,subqueries,aggreate functions and  
window functions .**

# Basic Questions



1. Retrieve the total number of orders placed.
2. Calculate the total revenue generated from pizza sales.

3. Identify the highest-priced pizza.
4. Identify the most common pizza size ordered.

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

# Intermediate Questions



1. Join the necessary tables to find the total quantity of each pizza category.
2. Determine the distribution of orders by hour of the day.

3. Join relevant tables to find the category-wise distribution of pizzas.
4. Group the orders by date and calculate the average number of pizzas ordered per day.

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

# Advanced Questions



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

**2. Analyze the cumulative revenue generated over time.**

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



...

# Level 1

Basic questions



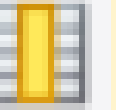
# 1. Retrieve the total number of orders placed.

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

Result Grid	
	total_orders
▶	21350

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

```
select round(sum(order_details.quantity * pizzas.price),2 )  
as total_sales  
from order_details join pizzas  
on pizzas.pizza_id = order_details.pizza_id
```

Result Grid		
	total_sales	
▶	817860.05	



### 3. 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;
```

Result Grid			Filter Rows
	name	price	
▶	The Greek Pizza	35.95	

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

```
select pizzas.size, count(order_details.order_details_id)
as order_count
from order_details join pizzas
on order_details.pizza_id = pizzas.pizza_id
group by pizzas.size
order by order_count desc;
```

	size	order_count
►	L	18526
	M	15385
	S	14137
	XL	544
	XXL	28

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

```
select pizza_types.name,  
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 order_details.pizza_id = pizzas.pizza_id  
group by pizza_types.name  
order by quantity desc limit 5;
```

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



...

# Level 2

Intermediate questions

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

```
select pizza_types.category,  
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 order_details.pizza_id = pizzas.pizza_id  
group by pizza_types.category order by quantity desc;
```

	category	quantity
►	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050

## 2. Determine the distribution of orders by hour of the day.

```
select hour(order_time) as hourly_order,  
count(order_id) as order_count  
from orders  
group by hourly_order;
```

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

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

```
select category ,count(name) from pizza_types  
group by category
```

	category	count(name)
▶	Chicken	6
	Classic	8
	Supreme	9
	Veggie	9

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

```
select round(avg(quantity),0) as avg_pizza 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;
```

	avg_pizza
▶	138



## 5. 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 pizzas.pizza_type_id = pizza_types.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



...

# Level 3

Advanced questions

# 1. 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_sales  
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

## 2. 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	2713.8500000000000
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

### 3. 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;
```

The Italian Supreme Pizza	33476.75
The Sicilian Pizza	30940.5
The Four Cheese Pizza	32265.7000
The Mexicana Pizza	26780.75
The Five Cheese Pizza	26066.5

	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

Thank  
You

