

**Hello my name is Aprajita.**

**In this project I have utilize SQL query to solve the question related to pizza sales.**

# Retrieve the total number of orders placed.

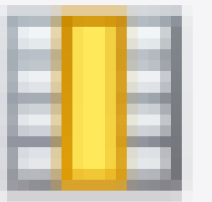
```
SELECT  
    COUNT(order_id) AS total_orders  
FROM  
    orders;
```

Result Grid	
	total_orders
▶	20887

# Calculate the total revenue generated from pizza sales.

```
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
```

Result Grid



	total_sales
▶	51423.15



# 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	

# Identify the most common pizza size ordered.

```
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 pizzas.size
ORDER BY order_count DESC;
```

Result Grid     Filter		
	size	order_count
▶	L	5281
	M	4279
	S	4027
	XL	158
	XXL	9

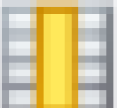

# List the top 5 most ordered pizza types along with their quantity.

```
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;
```

Result Grid			Filter Rows:	
	name	quantity		
▶	The Barbecue Chicken Pizza	852		
	The Hawaiian Pizza	823		
	The Pepperoni Pizza	804		
	The Classic Deluxe Pizza	787		
	The Thai Chicken Pizza	786		



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

```
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;
```

Result Grid					Filter
	category	quantity			
▶	Classic	5215			
	Supreme	4230			
	Veggie	4203			
	Chicken	3942			

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

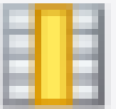

```
SELECT
    HOUR(time) AS hour, COUNT(order_id) AS order_count
FROM
    orders
GROUP BY HOUR(time);
```

Result Grid     Filter		
	hour	order_count
	21	1177
	22	648
	23	27
	10	8
	9	1



Join relevant tables to find the category the average number of pizzas ordered per day.

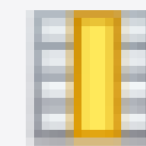
```
SELECT  
    category, COUNT(name)  
FROM  
    pizza_types  
GROUP BY category;
```

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

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

```
SELECT
    ROUND(AVG(quantity), 0) as avg_pizza_ordered_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 orders.date) AS order_quantity;
```

Result Grid



Filter Rows:

	avg_pizza_ordered_per_day
▶	138

# Determine the top 3 most ordererde 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;
```

Result Grid			Filter Rows:
	name	revenue	
▶	The Barbecue Chicken Pizza	19447.25	
	The Thai Chicken Pizza	18063.25	
	The California Chicken Pizza	17733.5	

# 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 order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
ORDER BY revenue DESC
LIMIT 4;
```

Result Grid			Filter
	category	revenue	
▶	Classic	26.69	
	Supreme	25.59	
	Veggie	23.89	
	Chicken	23.83	

# Analyze the cumulative revenue generated over time.

```
select date, sum(revenue) over(order by date) as cum_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 orders.order_id = order_details.order_id
group by orders.date) as sales;
```

Result Grid			Filter Rows:
	date	cum_revenue	
▶	2015-01-01	2713.85000000000004	
	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	
	2015-01-09	21526.4	

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

```
(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;
```

Result Grid			Filter Rows:
	name	revenue	
	The Italian Supreme Pizza	18750	
	The Sicilian Pizza	16706.75	
	The Four Cheese Pizza	17878.0500000000218	
	The Five Cheese Pizza	14781.5	
	The Mexicana Pizza	14373.25	

**Thank you**