



# INTRODUCTION

Hello my name Md Tariq and in this project i have utilized SQL queries to solve questions. That were related to in pizza sales.





Retrieve the total number of orders placed.

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



Result Grid	
	total_orders
	21350



# Calculate the total revenue generated from pizza sales

SELECT

ROUND(SUM(orders\_details.quantity \* pizzas.price),  
2) AS total\_revenue

FROM

orders\_details

JOIN

pizzas ON pizzas.pizza\_id = orders\_details.pizza\_id;



Result Grid

total\_revenue

817860.05



Identify the highest-priced pizza.

```
SELECT
```

```
*
```

```
FROM
```

```
pizzas
```

```
ORDER BY price DESC
```

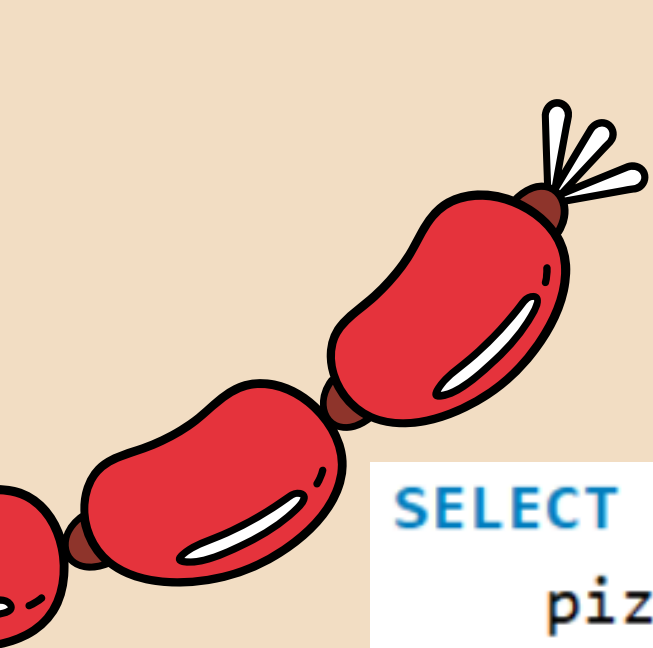
```
LIMIT 1;
```

Result Grid					Filter Rows:				
	pizza_id	pizza_type_id	size	price					
▶	the_greek_xxl	the_greek	XXL	35.95					

# Identify the most common pizza size ordered.

```
SELECT
  pizzas.size,
  COUNT(orders_details.order_details_id) AS order_count
FROM
  pizzas
  JOIN
  orders_details ON pizzas.pizza_id = orders_details.pizza_id
GROUP BY pizzas.size
ORDER BY order count DESC
```

Result Grid		Filter
	size	order_count
	L	18526



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

```
SELECT
    pizza_types.name, SUM(orders_details.quantity) AS quantity
FROM
    pizza_types
    JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
    JOIN
    orders_details ON orders_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY quantity DESC
LIMIT 5;
```

Result Grid		Filter Rows:
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	









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

```
SELECT
    pizza_types.category,
    SUM(orders_details.quantity) AS quantity
FROM
    pizza_types
    JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
    JOIN
    orders_details ON orders_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
ORDER BY quantity DESC;
```

Result Grid |   Filter Rows:

	category	quantity
	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050



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

**SELECT**

**HOUR**(order\_time) **AS** order\_hour,  
**COUNT**(order\_id) **AS** order\_count

**FROM**

orders

**GROUP BY HOUR**(order\_time);

Result Grid |   Filter Rows


	order_hour	order_count
▶	11	1231
	12	2520
	13	2455
	14	1472
	15	1468
	16	1920






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

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



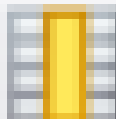

	category	count(name)
	Chicken	6
	Classic	8
	Supreme	9
	Veggie	9





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

```
SELECT
    ROUND(AVG(quantity), 0) as avg_pizzas_ordered_per_day
FROM
    (SELECT
        orders.order_date, SUM(orders_details.quantity) AS quantity
    FROM
        orders
    JOIN orders_details ON orders.order_id = orders_details.order_id
    GROUP BY orders.order_date) AS order_quantity;
```

Result Grid |   Filter Rows:


	avg_pizzas_ordered_per_day
--	----------------------------



	138
--	-----



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

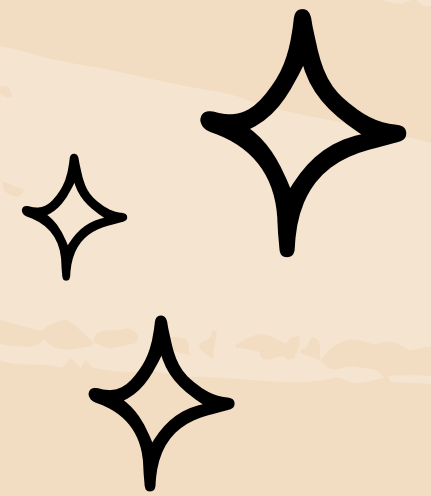
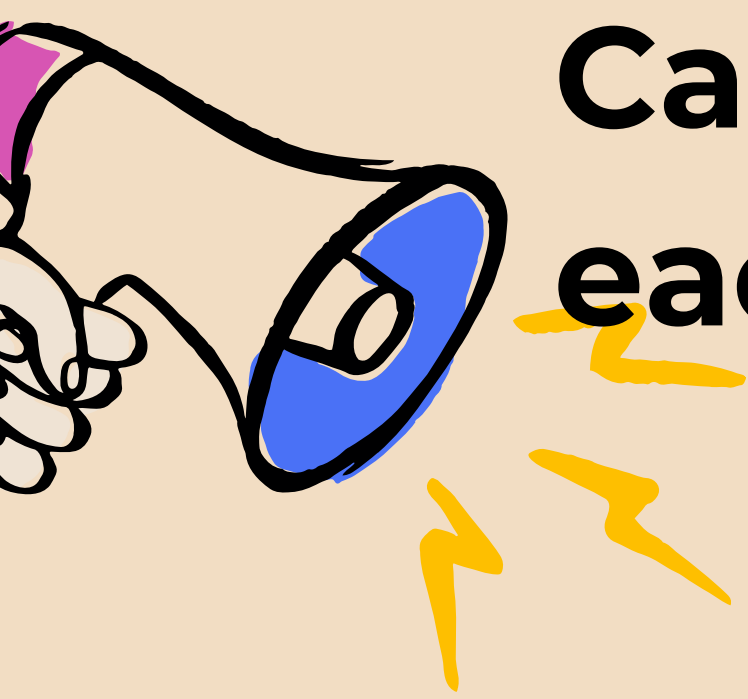
```
SELECT
    pizza_types.name,
    SUM(orders_details.quantity * pizzas.price) AS revenue
FROM
    pizza_types
    JOIN
    pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
    JOIN
    orders_details ON orders_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY revenue DESC
LIMIT 3;
```



Result Grid |   Filter Rows:

	name	revenue
	The Thai Chicken Pizza	43434.25
	The Barbecue Chicken Pizza	42768
	The California Chicken Pizza	41409.5

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

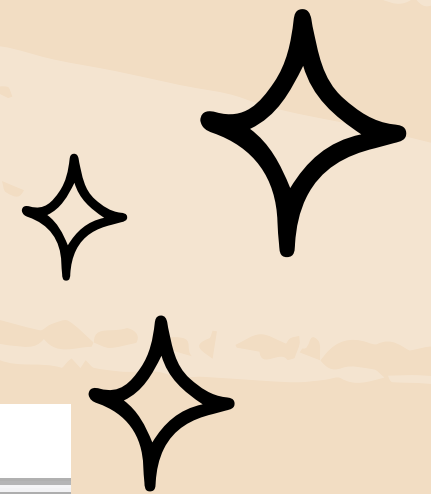
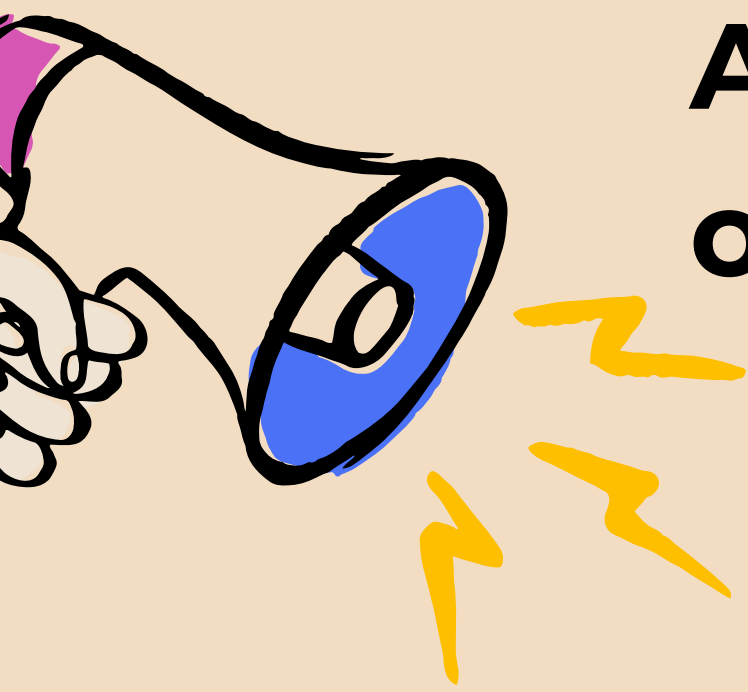


```
SELECT
    pizza_types.category,
    round( (SUM(orders_details.quantity * pizzas.price) / (SELECT
        ROUND(SUM(orders_details.quantity * pizzas.price),
            2) AS total_revenue
    FROM
        orders_details
        JOIN
        pizzas ON pizzas.pizza_id = orders_details.pizza_id) ) * 100,2) as revenue
FROM
    pizza_types
    JOIN
    pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
    JOIN
    orders_details ON orders_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
```


Result Grid			Filter Rows:
	category	revenue	
▶	Classic	26.91	
	Supreme	25.46	
	Chicken	23.96	
	Veggie	23.68	

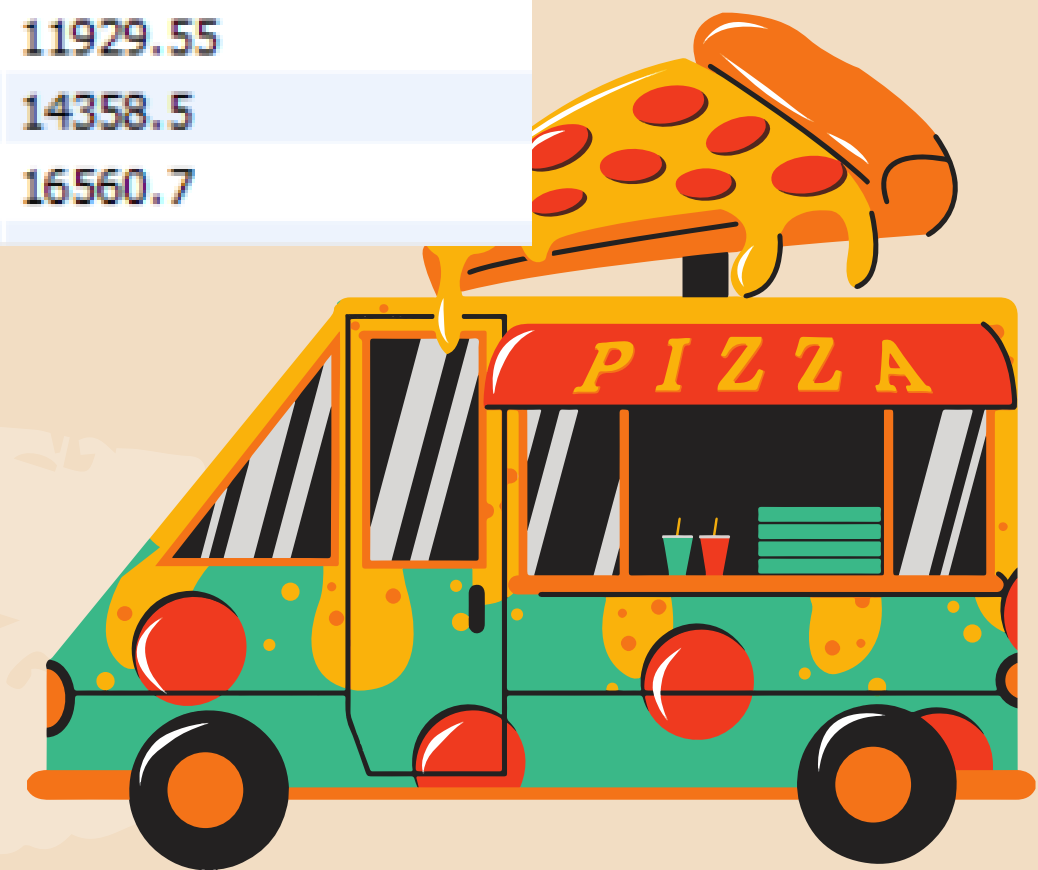


# Analyze the cumulative revenue generated over time.



```
select order_date,  
sum(revenue) over(order by order_date) as cum_revenue  
from  
(select orders.order_date,  
sum(orders_details.quantity * pizzas.price) as revenue  
from orders_details join pizzas  
on orders_details.pizza_id = pizzas.pizza_id  
join orders  
on orders.order_id = orders_details.order_id  
group by orders.order_date) as sales;
```

Result Grid    Filter Rows: <input type="text"/>		
	order_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





# Thank You!



**Email**

m.tarique95@gmail.com



**LinkedIn**

[www.reallygreatsite.com](http://www.reallygreatsite.com)



**Phone Number**

+919971425611

