

PIZZAS SALE

Delicious Pizza For Everyone



HELLO !

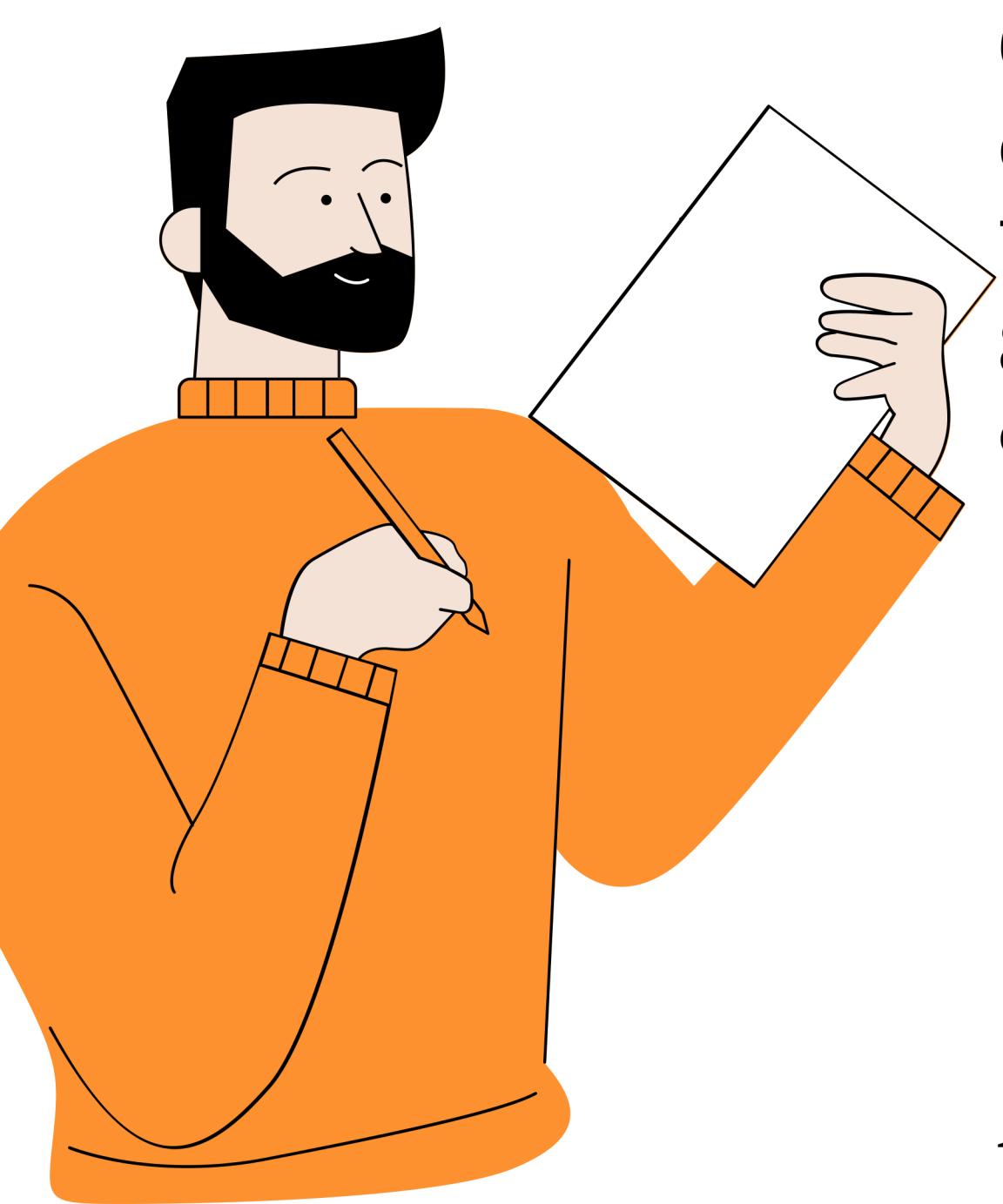
My Name Is Md Sahil Gaddi

**This project i have utilize sql query to
solve questions to related pizzas sales**





PIZZA SALES QUESTIONS

- 
1. Retrieve the total number of orders placed
 2. Calculate the total revenue generated from pizza sales
 3. Identify the most common pizza size ordered
 4. Identify the highest price pizza
 5. List the top 5 most ordered pizza types along with their quantities
 6. Join the necessary tables to find the total quantity of each pizza category ordered
 7. Determine the distribution of orders by hour of the day
 8. Join relevant tables to find the category wise distribution of pizza
 9. GROUP THE ORDERS BY DATE AND CALCULATE THE AVERAGE NUMBER OF PIZZAS ORDERED PER DAY
 10. Determine the top 3 most ordered pizza types based on revenue
 11. JOIN REVELANT TABLES TO FIND THE CATEGORY WISE DISTRIBUTION OF PIZZAS.
 12. analyze the cumulative revenue generated over time .
 13. Determine the top 3 most ordered pizza types based on revenue for each pizza category

1. Retrieve the total number of orders placed

```
SELECT count(order_id) AS TOTAL_ORDERS  
FROM orders;
```

	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
```

	TOTAL_SALES
817860.05	

3. 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

LIMIT 1;

	size	order_count
	L	18526

4. 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 ;
```

	name	price
▶	The Greek Pizza	35.95

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

6.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 pizzas.pizza_id=order_details.pizza_id  
 GROUP BY pizza_types.category ORDER BY quantity DESC;
```

	category	quantity
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050

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

```
SELECT HOUR(order_time) as hour ,  
       count(order_id) as order_count  
  from orders  
 GROUP BY HOUR(order_time);
```

	hour	order_count
▶	11	1231
	12	2520
	13	2455
	14	1472
	15	1468
	16	1920
	17	2336
	18	2399
	19	2009

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

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

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

9.GROUP THE ORDERS BY DATE AND CALCULATE THE AVERAGE NUMBER OF PIZZAS ORDERED PER DAY

```
SELECT round(avg(QUANTITY),0) AS AVERAGE_PIZZA_ORDER  
FROM  
> (SELECT orderss.order_date, sum(order_details.quantity) AS QUANTITY  
FROM  
    orderss JOIN order_details  
    ON orderss.order_id=order_details.order_id  
    GROUP BY orderss.order_date) AS ORDER_QUANTITY;
```

	AVERAGE_PIZZA_ORDER
▶	138

10.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 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 revenue DESC LIMIT 3;
```

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

11.JOIN REVELANT TABLES TO FIND THE CATEGORY WISE DISTRIBUTION OF PIZZAS.

```
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 pizzas.pizza_id = order_details.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

12.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 ordersss
        ON ordersss.order_id=order_details.order_id
    GROUP BY ordersss.order_date) AS sales;
```

	order_date	cum_revenue
▶	2015-01-01	2713.8500000000004
	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

13.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;
```

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
The Italian Supreme Pizza	33476.75
The Sicilian Pizza	30940.5

Thank You

My Sql Query Presentation

END

