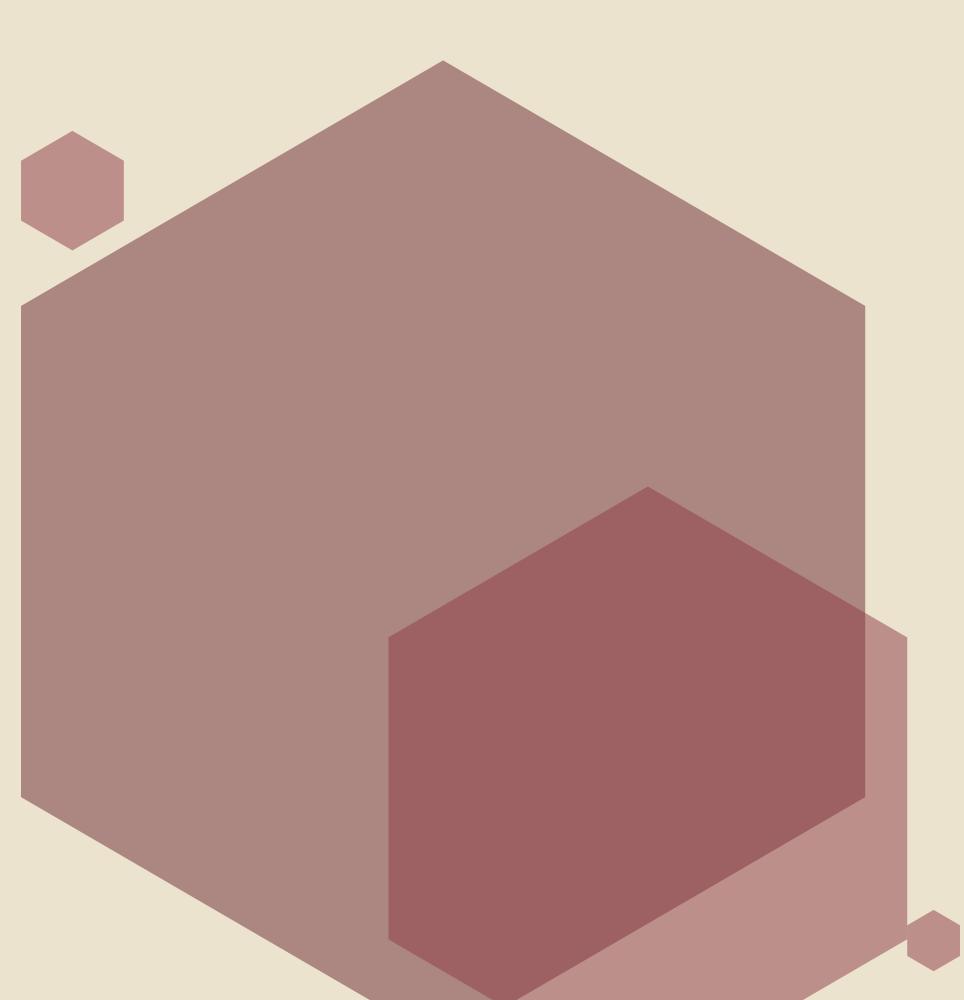
Pizza SALES REPORT



INTRODUCTION

Hello, my name is Vipul Saini, and I am excited to present my project titled "Pizza Sales using SQL." This project is a comprehensive analysis of pizza sales data, leveraging the power of SQL to derive meaningful insights and trends from the dataset..

CHALLENGES:

- Retrieve the total number of orders placed.
- Calculate the total revenue generated from pizza sales.
- Identify the highest-priced pizza.
- Identify the most common pizza size ordered.
- List the top 5 most ordered pizza types along with their quantities.
- Join the necessary tables to find the total quantity of each pizza category ordered.
- Determine the distribution of orders by hour of the day.
- Join relevant tables to find the category-wise distribution of pizzas.
- Group the orders by date and calculate the average number of pizzas ordered per day.
- Determine the top 3 most ordered pizza types based on revenue.
- Calculate the percentage contribution of each pizza type to total revenue.
- Analyze the cumulative revenue generated over time.
- Determine the top 3 most ordered pizza types based on revenue for each pizza category.

RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED



count(order_id)

21350

CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

```
SELECT

ROUND(SUM(order_details.quantity * pizzas.price),

2) AS total_sales

FROM

order_details

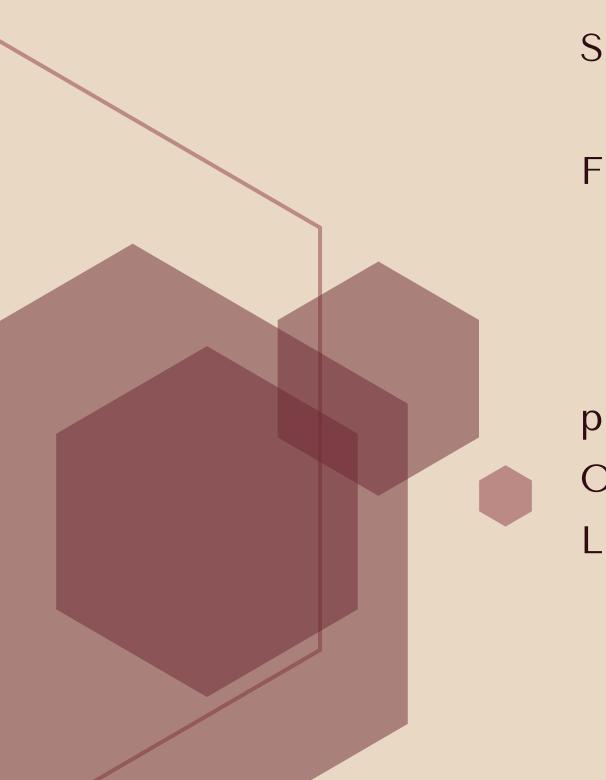
JOIN

pizzas ON pizzas.pizza_id =

order_details.pizza_id;

817860.05
```

Identify the highest-priced pizza.



```
SELECT
  pizza_types.name, pizzas.price
FROM
  pizza_types
     JOIN
    pizzas ON pizzas.pizza_type_id =
pizza_types.pizza_type_id
ORDER BY pizzas.price DESC
LIMIT 1;
                               name
```

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;
```

size	order_count
L	18526
М	15385
S	14137
XL	544
XXL	28

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

Total_quantity

2453

2432

2422

2418

2371

The Thai Chicken Pizza

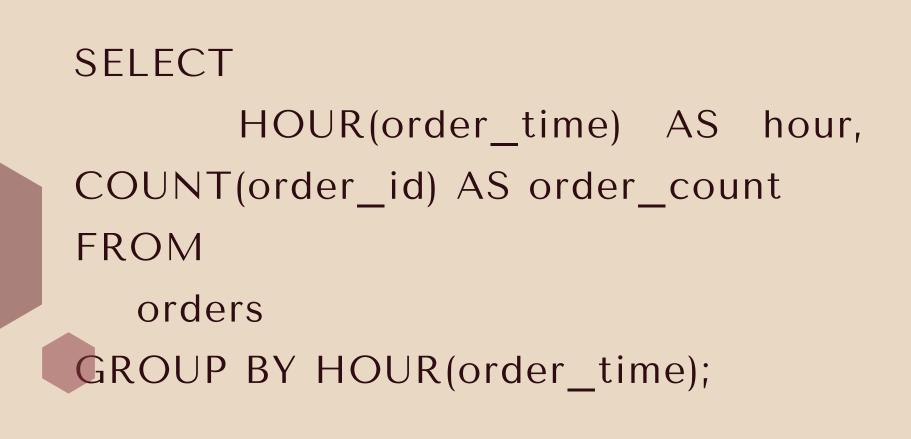
```
SELECT
  pizza_types.name,
  SUM(order details.quantity) AS Total quantity
FROM
   pizza_types
      JOIN
   pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
     JOIN
  order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
                                          name
ORDER BY Total_quantity DESC
                                          The Classic Deluxe Pizza
LIMIT 5;
                                          The Barbecue Chicken Pizza
                                          The Hawaiian Pizza
                                          The Pepperoni Pizza
```

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 pizzas.pizza_type_id = pizza_types.pizza_type_id
     JOIN
  order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
ORDER BY Total_quantity DESC;
```

category	Total_quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

Determine the distribution of orders by hour of the day.



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
10	8
9	1

JOIN RELEVANT TABLES TO FIND THE CATEGORY-WISE DISTRIBUTION OF PIZZAS.

select category, count(name) from pizza_types

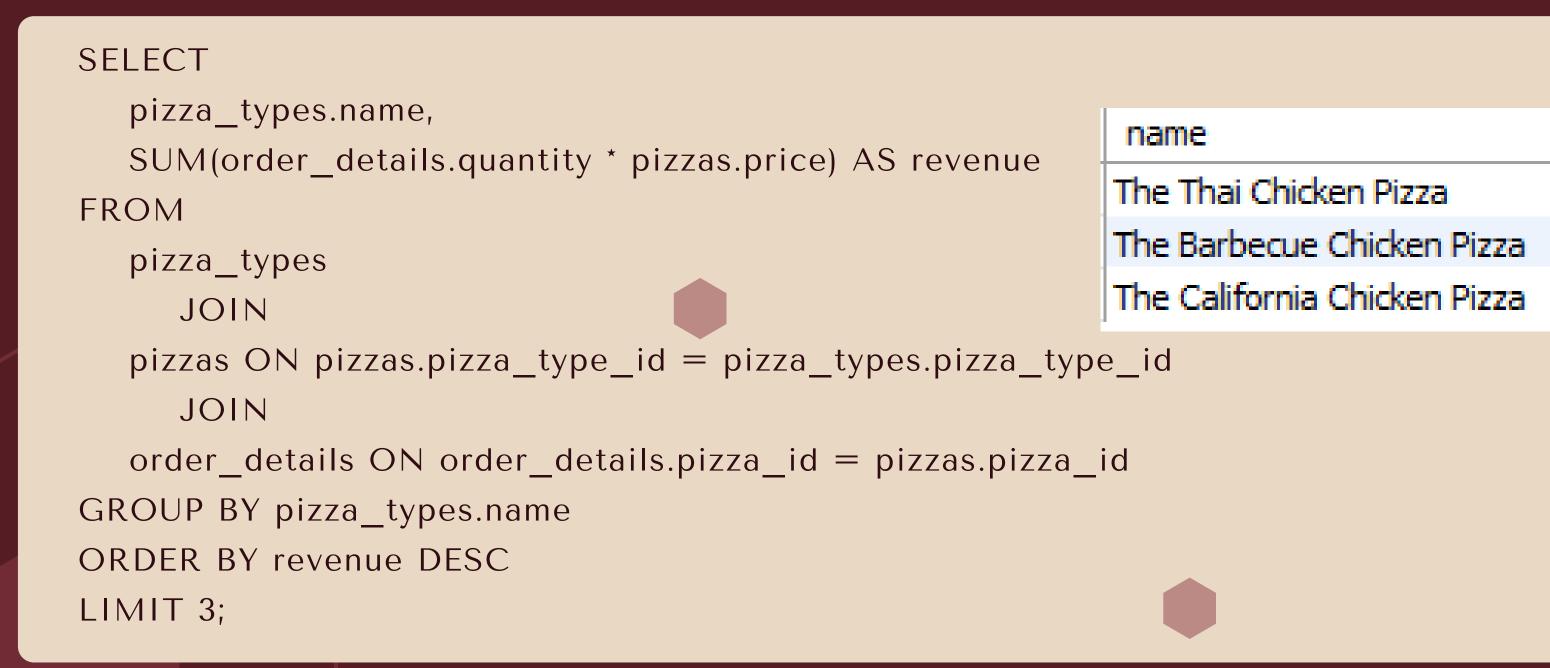
group by category;

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.

Average_pizza_per_day 138 SELECT ROUND(AVG(quantity), 0) as Average_pizza_per_day FROM (SELECT orders.order date, SUM(order details.quantity) AS quantity FROM orders JOIN order details ON order details.order id = orders.order id GROUP BY orders.order_date) AS order_quantity;

DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE.





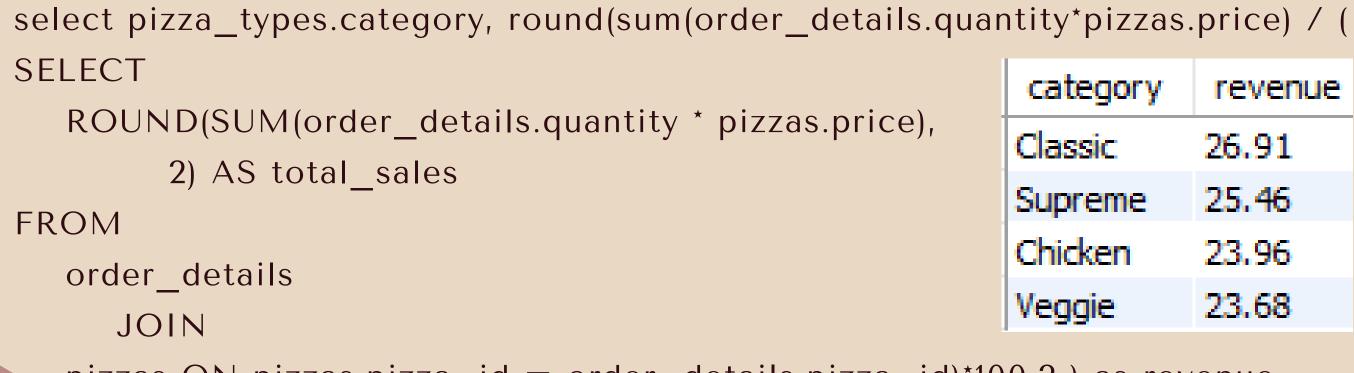
revenue

43434,25

42768

41409.5

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



pizzas ON pizzas.pizza_id = order_details.pizza_id)*100,2) 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.category
order by revenue desc;

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	2713.8500000000
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
2015-01-10	23990.350000000
2015-01-11	25862.65

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
The Four Cheese Pizza	32265.7000000006
The Mexicana Pizza	26780.75

The Five Cheese Pizza

26066.5

THANK YOU