



SQL PROJECT ON PIZZA

PRESENTED BY HARSHIT GUPTA



INTRODUCTION

IN THE PIZZA HUT SQL PROJECT, I ANALYZED SALES AND OPERATIONAL DATA USING A RANGE OF SQL TECHNIQUES. THIS INCLUDED JOINS FOR COMBINING DATA FROM MULTIPLE TABLES, SUBQUERIES FOR NESTED DATA RETRIEVAL, AND WINDOW FUNCTIONS TO PERFORM CALCULATIONS ACROSS SPECIFIC DATA PARTITIONS, ENHANCING INSIGHTS INTO SALES TRENDS AND CUSTOMER BEHAVIOR. I ALSO USED COMMON TABLE EXPRESSIONS (CTES) TO SIMPLIFY COMPLEX QUERIES, MAKING THEM MORE READABLE AND MANAGEABLE. AGGREGATION FUNCTIONS PLAYED A KEY ROLE IN SUMMARIZING DATA, SUCH AS CALCULATING TOTAL SALES AND AVERAGE ORDER VALUES, HELPING TO UNCOVER VALUABLE BUSINESS INSIGHTS FROM THE PIZZA HUT DATASET.



SHOW TABLES

```
+-----+  
| Tables_in_pizzahut |  
+-----+  
| order_details  
| orders  
| pizza_types  
| pizzas  
+-----+
```



ORDERS_DETAILS

```
select * from order_details limit 5;
```

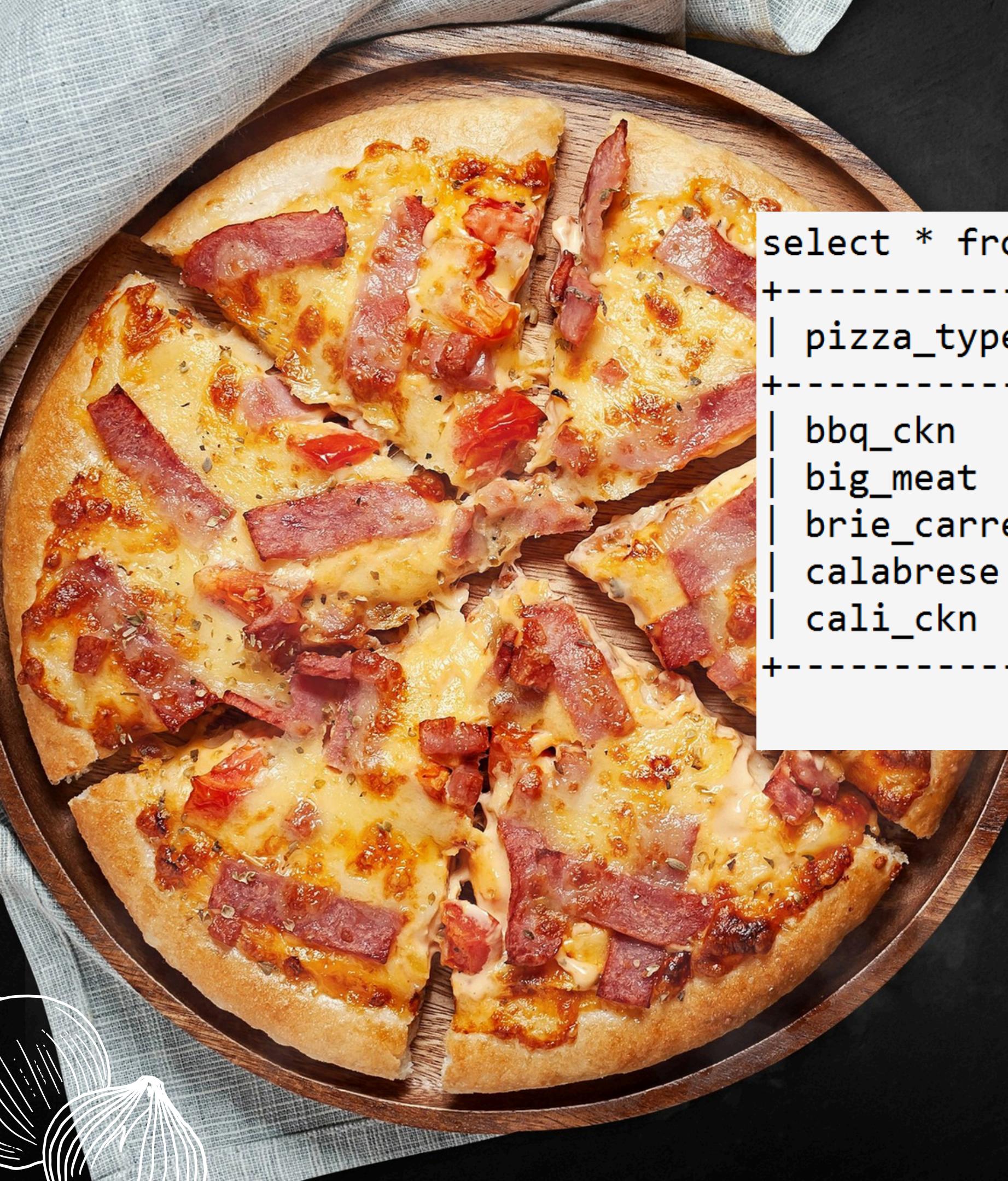
order_details_id	order_id	pizza_id	quantity
1	1	hawaiian_m	1
2	2	classic_dlx_m	1
3	2	five_cheese_1	1
4	2	ital_supr_1	1
5	2	mexicana_m	1

A photograph of a pizza on a round wooden board. The pizza is cut into eight slices and topped with melted cheese, ham, and small red pepper pieces. It sits on a dark surface with a light-colored cloth napkin visible at the top left. In the bottom left corner, there's a white line-art illustration of a whisk.

ORDERS

```
select * from orders limit 5;
```

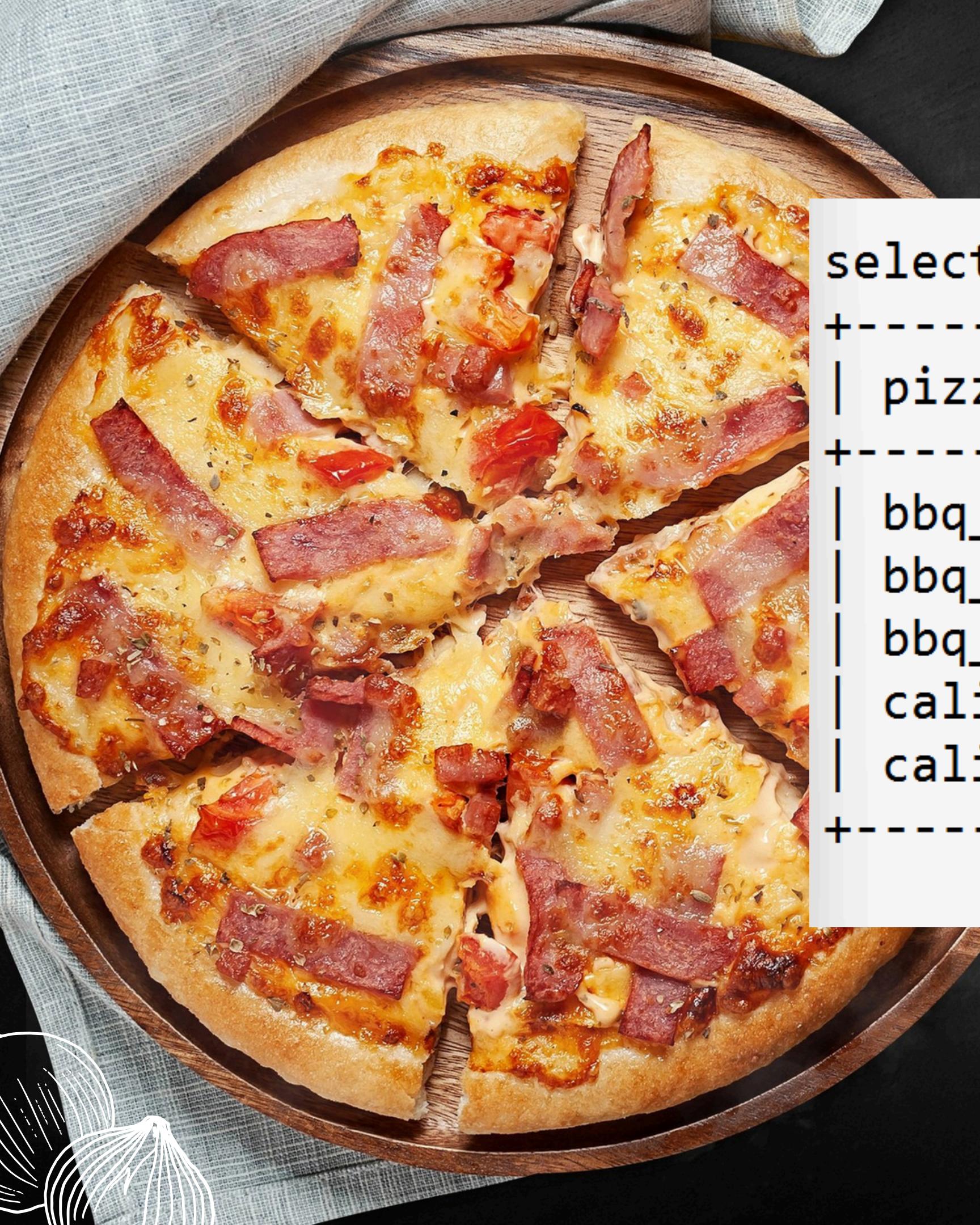
+-----+-----+-----+	id order_date order_time	+-----+-----+-----+
	1 2015-01-01 11:38:36	
	2 2015-01-01 11:57:40	
	3 2015-01-01 12:12:28	
	4 2015-01-01 12:16:31	
	5 2015-01-01 12:21:30	
	+-----+-----+-----+	



PIZZA_TYPES

```
select * from pizza_types limit 5;
```

pizza_type_id	name	category
bbq_ckn	The Barbecue Chicken Pizza	Chicken
big_meat	The Big Meat Pizza	Classic
brie_carre	The Brie Carre Pizza	Supreme
calabrese	The Calabrese Pizza	Supreme
cali_ckn	The California Chicken Pizza	Chicken



PIZZAS

```
select * from pizzas limit 5;
```

-----+-----+-----+-----+	-----+-----+-----+-----+	-----+-----+-----+-----+	-----+-----+-----+-----+
-----+-----+-----+-----+	-----+-----+-----+-----+	-----+-----+-----+-----+	-----+-----+-----+-----+
-----+-----+-----+-----+	-----+-----+-----+-----+	-----+-----+-----+-----+	-----+-----+-----+-----+
-----+-----+-----+-----+	-----+-----+-----+-----+	-----+-----+-----+-----+	-----+-----+-----+-----+
-----+-----+-----+-----+	-----+-----+-----+-----+	-----+-----+-----+-----+	-----+-----+-----+-----+
bbq_ckn_s	bbq_ckn	S	12.75
bbq_ckn_m	bbq_ckn	M	16.75
bbq_ckn_l	bbq_ckn	L	20.75
cali_ckn_s	cali_ckn	S	12.75
cali_ckn_m	cali_ckn	M	16.75
-----+-----+-----+-----+	-----+-----+-----+-----+	-----+-----+-----+-----+	-----+-----+-----+-----+

QUESTIONS

BASIC:

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:

1. JOIN THE NECESSARY TABLES TO FIND THE TOTAL QUANTITY OF EACH PIZZA CATEGORY ORDERED.
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:

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.

1.RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.

```
select count(id) as Total_order from orders;  
+-----+  
| Total_order |  
+-----+  
|      200   |  
+-----+
```



2.CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

```
select
sum(p.price*o.quantity) as total_sales
from pizzas p
inner join
order_details o
on
o.pizza_id=p.pizza_id;
+-----+
| total_sales |
+-----+
| 3755.55 |
+-----+
```



3. IDENTIFY THE HIGHEST-PRICED PIZZA.

```
select * from pizzas
where price = (select max(price) from pizzas);
+-----+-----+-----+-----+
| pizza_id      | pizza_type_id | size    | price   |
+-----+-----+-----+-----+
| the_greek_xx1 | the_greek     | XXL    | 35.95  |
+-----+-----+-----+-----+
```



4.IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.

```
select size, count(size) as count  
from pizzas  
group by size  
order by count(size) desc;
```

size	count
S	35
M	34
L	34
XL	1
XXL	1



5. LIST THE TOP 5 MOST ORDERED PIZZA TYPES ALONG WITH THEIR QUANTITIES.

```
select p.name,sum(o.quantity) as Total  
from pizza_types as p  
inner join  
pizzas as s on p.pizza_type_id=s.pizza_type_id  
inner join  
order_details as o on o.pizza_id=s.pizza_id  
group by p.name  
order by Total desc limit 5;
```

name	Total
The Barbecue Chicken Pizza	24
The Italian Supreme Pizza	17
The Thai Chicken Pizza	14
The Classic Deluxe Pizza	13
The Mexicana Pizza	10



1.JOIN THE NECESSARY TABLES TO FIND THE TOTAL QUANTITY OF EACH PIZZA CATEGORY ORDERED.

```
select
p.category,sum(o.quantity) as Total_quantity
from
pizza_types as p inner join pizzas as s
on
p.pizza_type_id = s.pizza_type_id
inner join
order_details as o
on
o.pizza_id = s.pizza_id
group by p.category;
+-----+-----+
| category | Total_quantity |
+-----+-----+
| Classic  |       60  |
| Veggie   |       50  |
| Supreme  |       48  |
| Chicken  |      64  |
+-----+-----+
```



2.DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.

```
select hour(o.order_time) as Hours,  
count(d.quantity) as Total_quanitiy  
from orders as o inner join order_details as d  
on  
o.id=d.order_details_id group by Hours;
```

Hours	Total_quanitiy
11	6
12	17
13	13
14	10
15	11
16	8
17	13
18	22
19	14
20	19
21	16
22	21
23	30



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 SUM OF PIZZAS ORDERED PER MONTH.

```
select monthname(o.order_date) as Month,  
sum(d.quantity) as Total_Quantity from orders as o  
inner join order_details as d  
group by monthname(o.order_date);  
+-----+-----+  
| Month | Total_Quantity |  
+-----+-----+  
| February | 26593 |  
| January | 14007 |  
+-----+-----+
```



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

```
select pizza_types.name as Name,  
sum(p.price*o.quantity) as Revenue  
from order_details o inner join pizzas p  
on p.pizza_id = o.pizza_id inner join pizza_types  
on pizza_types.pizza_type_id = p.pizza_type_id  
group by Name  
order by Revenue desc  
limit 3;  
+-----+-----+  
| Name | Revenue |  
+-----+-----+  
| The Barbecue Chicken Pizza | 458.00 |  
| The Italian Supreme Pizza | 302.25 |  
| The Thai Chicken Pizza | 262.50 |  
+-----+-----+
```



1.CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA CATEGORY TO TOTAL REVENUE.ORDERED PIZZA TYPES BASED ON REVENUE.

```
select pizza_types.category as Category,  
sum(p.price*o.quantity*100)/(select sum(p.price*o.quantity))  
from pizzas p inner join order_details o on p.pizza_id=o.pizza_id)  
as Percentage from order_details o inner join  
pizzas p on p.pizza_id = o.pizza_id inner join  
pizza_types on pizza_types.pizza_type_id = p.pizza_type_id  
group by category ;  
+-----+-----+  
| Category | Percentage |  
+-----+-----+  
| Classic  | 23.505212 |  
| Veggie   | 22.488051 |  
| Supreme  | 22.799590 |  
| Chicken  | 31.207147 |  
+-----+-----+
```



2.ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME.

```
select orders.* , p.price * o.quantity as Total , sum(p.price * o.quantity)
over (rows between unbounded preceding and current row) as Cumulative_Tot
from pizzas p inner join order_details o on p.pizza_id = o.pizza_id
inner join orders on orders.id = o.order_id limit 5;
```

id	order_date	order_time	Total	Cumulative_Total
1	2015-01-01	11:38:36	13.25	13.25
2	2015-01-01	11:57:40	16.00	29.25
2	2015-01-01	11:57:40	18.50	47.75
2	2015-01-01	11:57:40	20.75	68.50
2	2015-01-01	11:57:40	16.00	84.50

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

```
select
name as NAME,category as CATEGORY,Total as REVENUE,rnk from
(select name,category,Total,row_number()
over(partition by category order by Total desc) as rnk from
(select t.name,t.category,sum(p.price*o.quantity) as Total from
order_details as o inner join pizzas p on p.pizza_id=o.pizza_id
inner join
pizza_types as t on p.pizza_type_id=t.pizza_type_id
group by t.category,t.name)
as a ) as b where rnk<4;
```

NAME	CATEGORY	REVENUE	rnk
The Barbecue Chicken Pizza	Chicken	458.00	1
The Thai Chicken Pizza	Chicken	262.50	2
The California Chicken Pizza	Chicken	175.50	3
The Classic Deluxe Pizza	Classic	196.50	1
The Italian Capocollo Pizza	Classic	187.00	2
The Pepperoni Pizza	Classic	102.75	3
The Italian Supreme Pizza	Supreme	302.25	1
The Spicy Italian Pizza	Supreme	199.25	2
The Spinach Supreme Pizza	Supreme	95.25	3
The Mexicana Pizza	Veggie	185.75	1
The Five Cheese Pizza	Veggie	148.00	2
The Four Cheese Pizza	Veggie	116.05	3





THANK YOU!

