SQL PROJECT ON PIZZA SALES

HELLO!

My name is Aarchi Nagpal.

In this project, I have used SQL queries to address questions related to pizza sales.

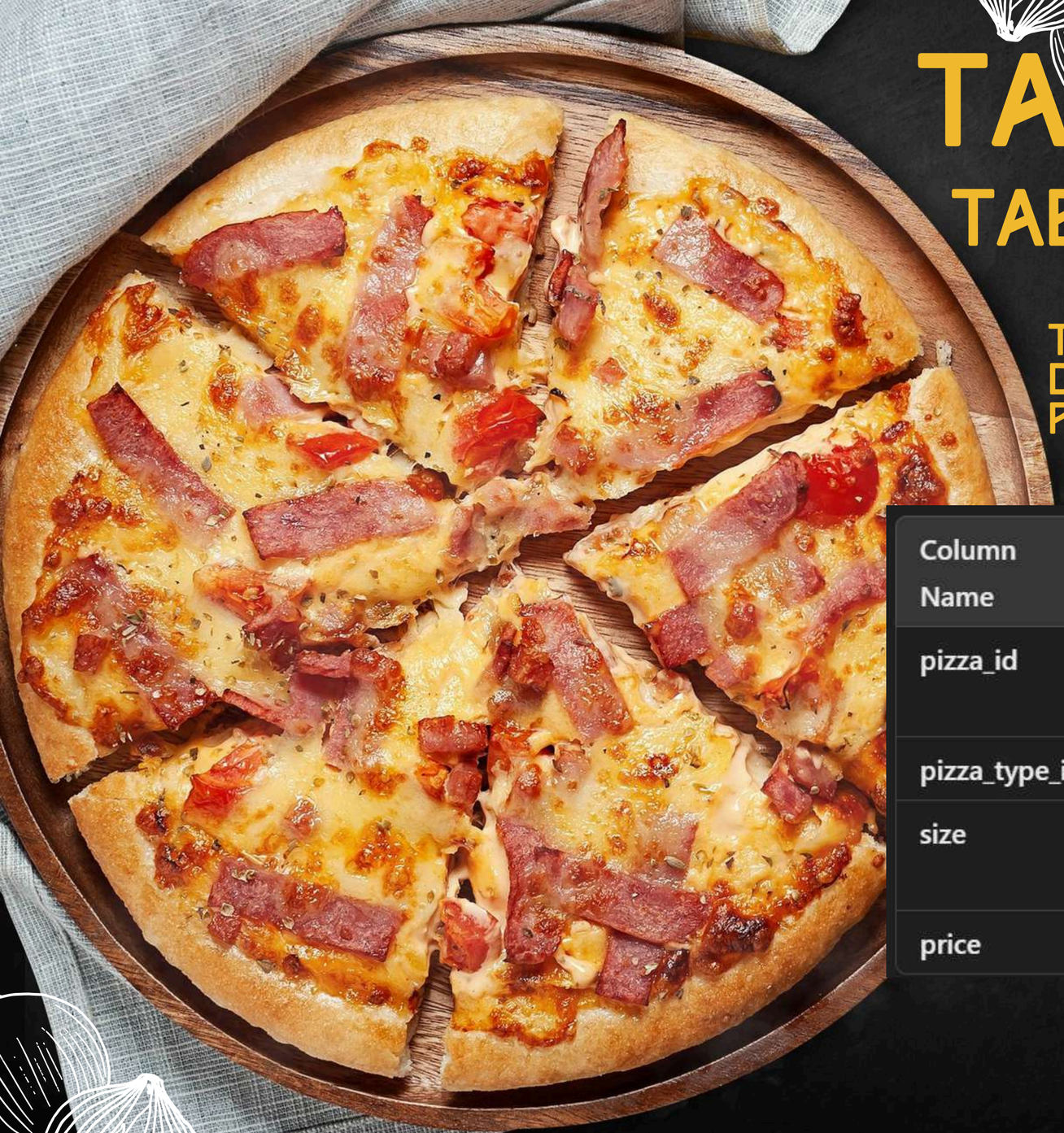


TABLE SCHEMAS

TABLE NAME: PIZZAS

THIS TABLE STORES DETAILS ABOUT DIFFERENT PIZZA ITEMS, INCLUDING THE PIZZA'S UNIQUE ID, TYPE, SIZE, AND PRICE.

Column Name	Data Type	Description
pizza_id	VARCHAR(50)	Unique identifier for each pizza item (e.g., 'bbq_ckn_s')
pizza_type_id	VARCHAR(50)	Type of the pizza (e.g., 'bbq_ckn', 'cali_ckn')
size	VARCHAR(1)	Size of the pizza (e.g., 'S' for Small, 'M' for Medium, 'L' for Large)
price	DECIMAL(5,2)	Price of the pizza item (e.g., 12.75)

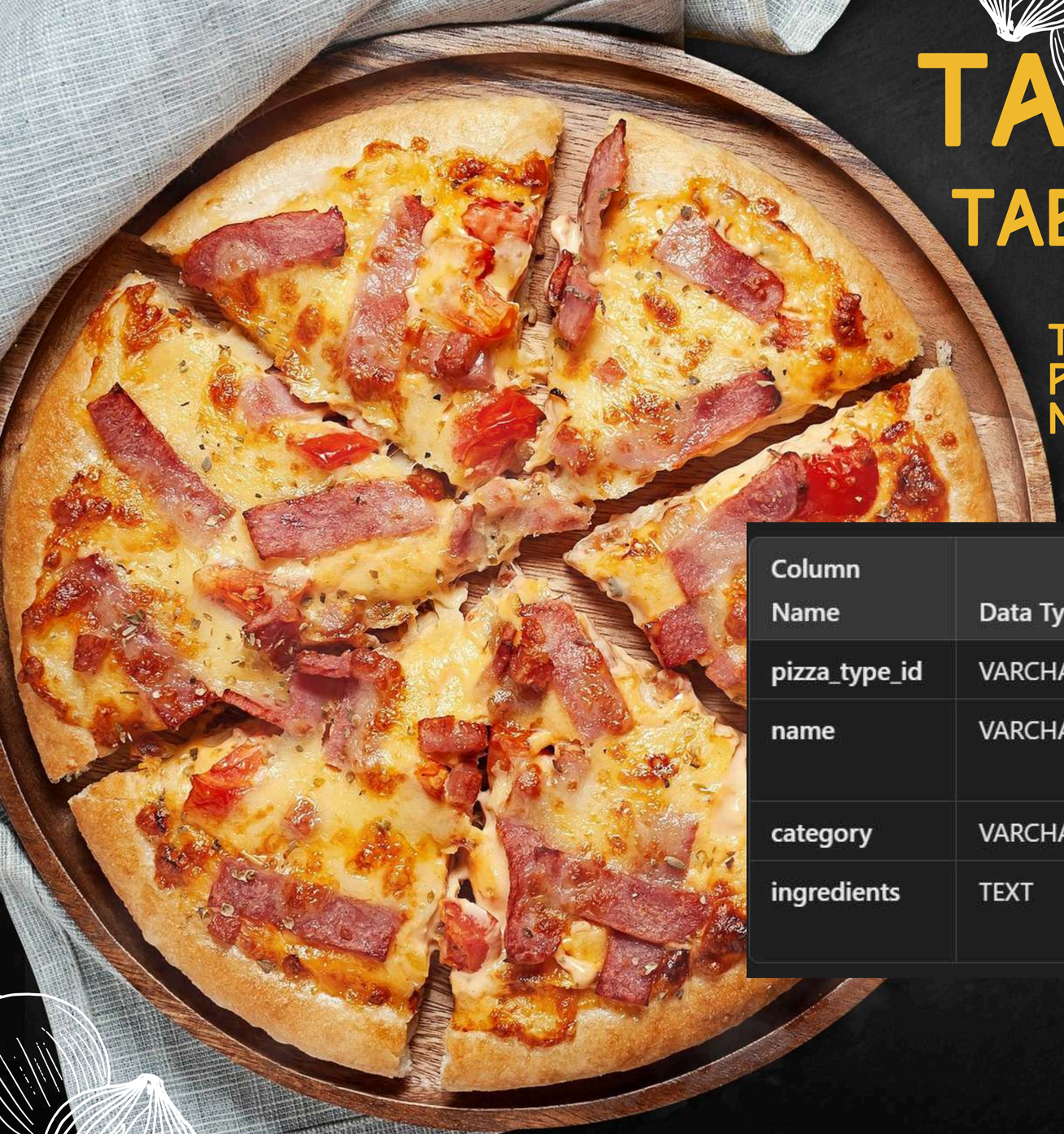


TABLE SCHEMAS

TABLE NAME: PIZZAS_TYPES

THIS TABLE STORES DETAILS ABOUT THE PIZZA TYPES, INCLUDING THEIR UNIQUE ID, NAME, CATEGORY, AND INGREDIENTS.

Column Name	Data Type	Description
pizza_type_id	VARCHAR(50)	Unique identifier for each type of pizza (e.g., 'bbq_ckn', 'classic_dlx')
name	VARCHAR(100)	The name of the pizza (e.g., 'The Barbecue Chicken Pizza', 'The Hawaiian Pizza')
category	VARCHAR(50)	Category of the pizza (e.g., 'Chicken', 'Classic', 'Supreme', 'Veggie')
ingredients	TEXT	List of ingredients used in the pizza (e.g., 'Barbecued Chicken, Red Peppers, ...')

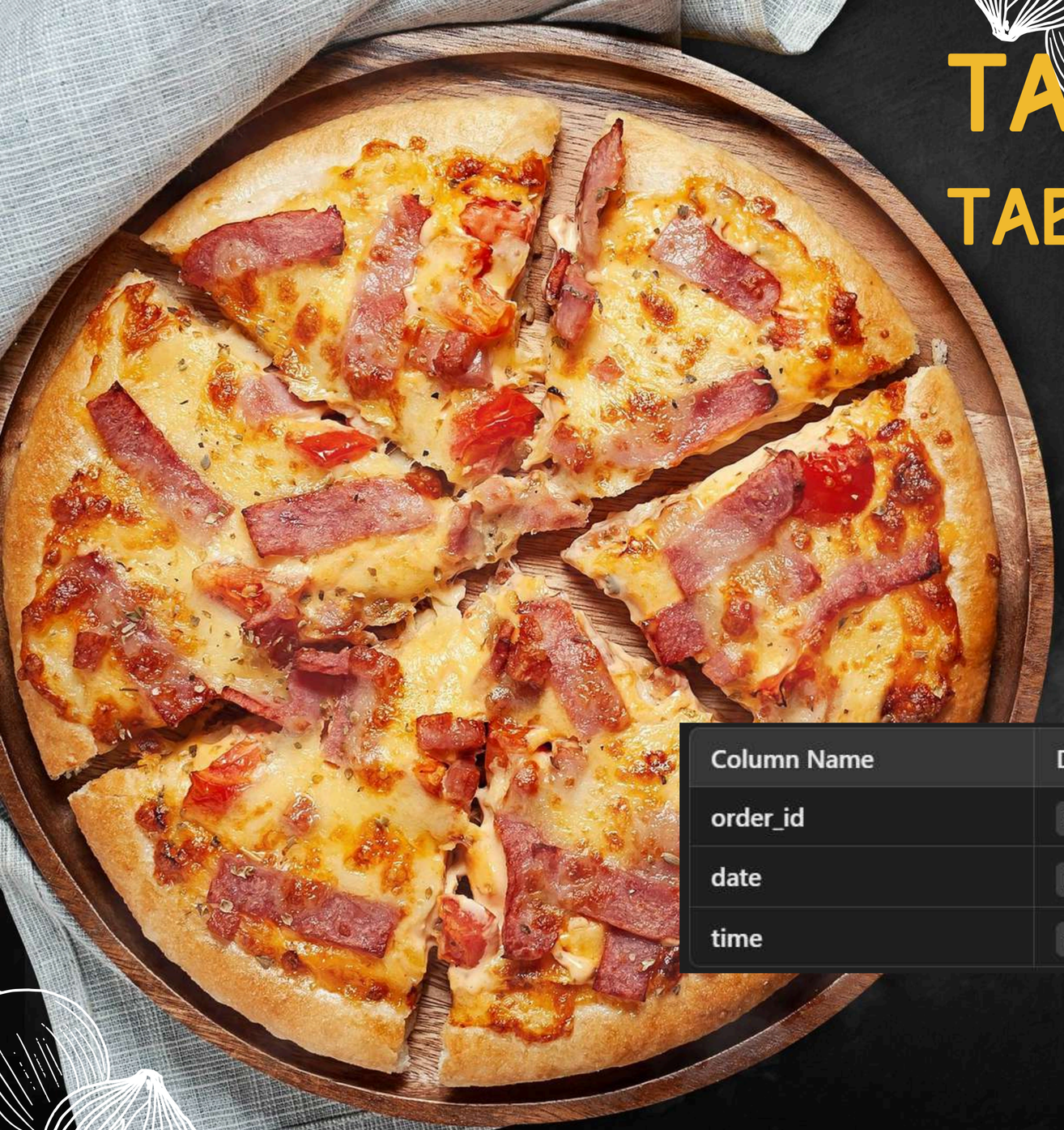


TABLE SCHEMAS

TABLE NAME: ORDERS

THIS TABLE STORES DETAILS ABOUT ORDERS PLACED BY CUSTOMERS. SPECIFICALLY, IT TRACKS THE UNIQUE IDENTIFIER OF EACH ORDER ALONG WITH THE DATE AND TIME WHEN THE ORDER WAS MADE.

Column Name	Data Type	Description
order_id	`INT`	Unique identifier for each order.
date	`DATE`	The date the order was placed.
time	`TIME`	The time the order was placed.

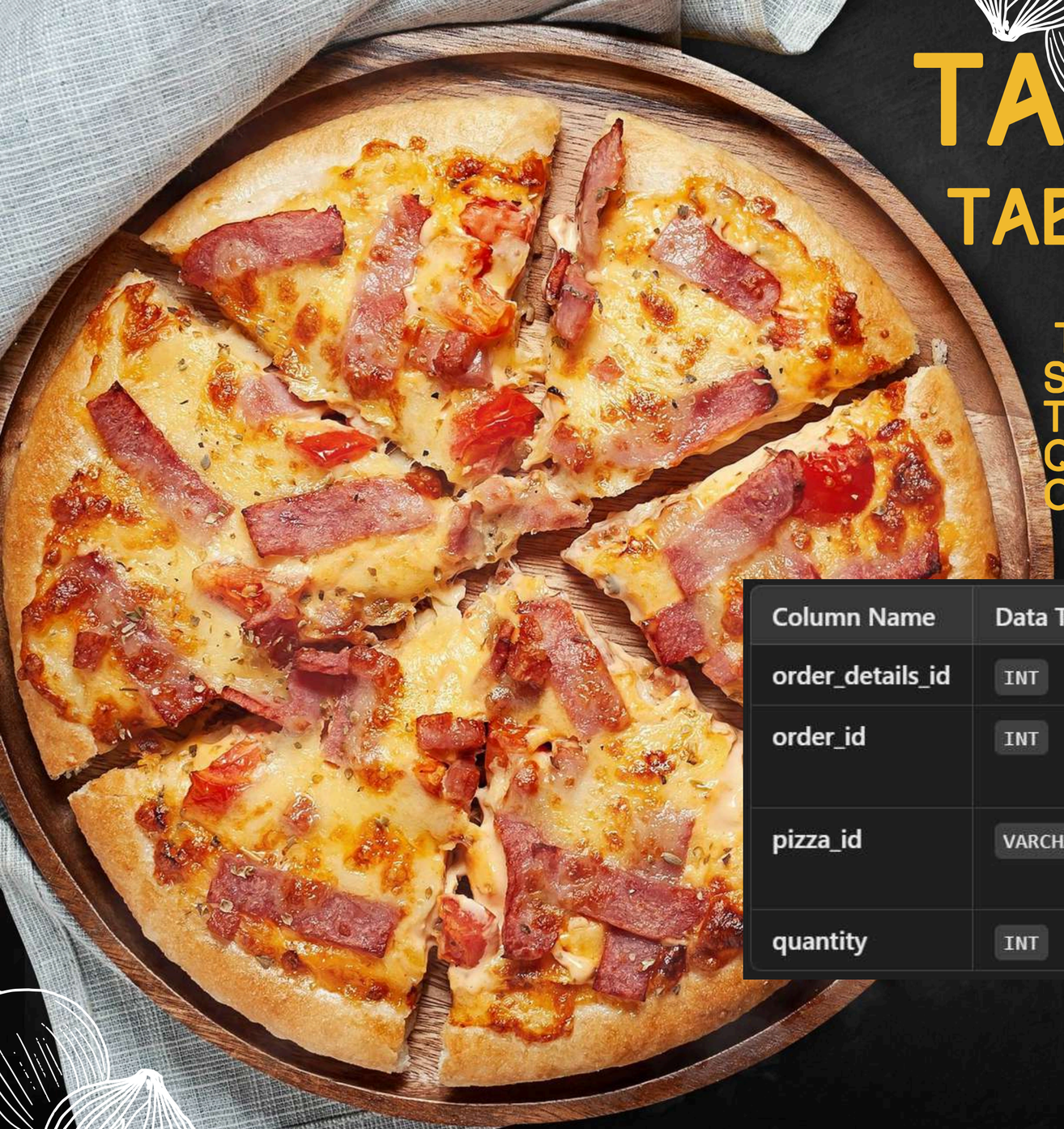


TABLE SCHEMAS

TABLE NAME:ORDER_DETAILS

THE INDIVIDUAL ITEMS IN EACH ORDER. SPECIFICALLY, IT TRACKS THE SPECIFIC TYPES OF PIZZAS ORDERED, THE QUANTITY OF EACH PIZZA, AND WHICH ORDER EACH PIZZA BELONGS TO.

Column Name	Data Type	Description
order_details_id	`INT`	Unique identifier for each order item.
order_id	`INT`	The ID of the order this detail is associated with. References the `orders` table.
pizza_id	`VARCHAR(255)`	Identifier for the type of pizza ordered (e.g., 'hawaiian_m', 'classic_dlx_m').
quantity	`INT`	The quantity of the specific pizza ordered.

RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.

```sql
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(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_revenue
▶	817860.05

IDENTIFY THE HIGHEST-PRICED PIZZA.

```sql
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..

```sql
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 | Filte

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

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

```sql
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 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.

Result Grid | Filter

category	quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

```sql
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;
```

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.

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

Result Grid

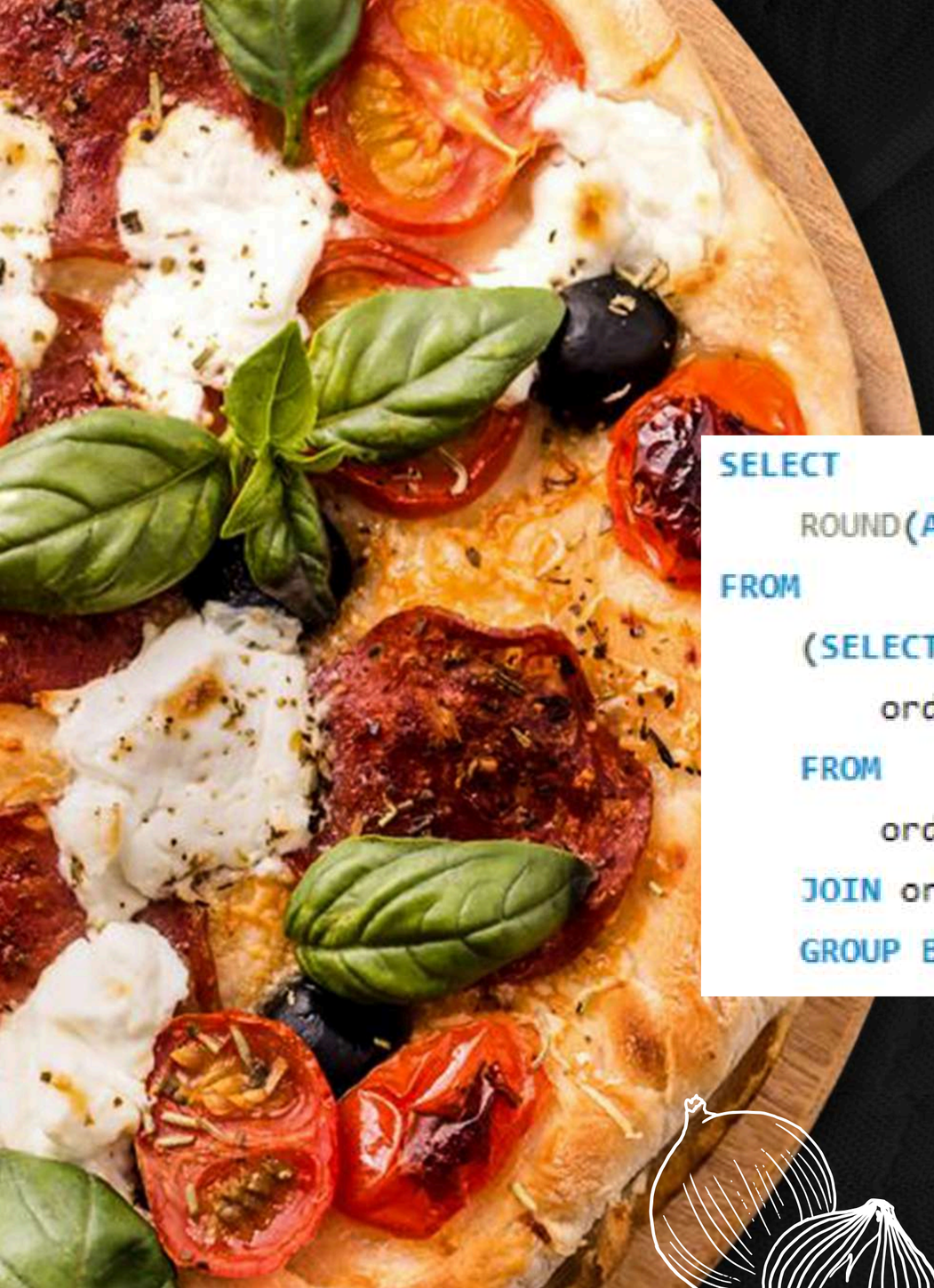
hour	order_count
11	1231
12	2520
13	12
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642
21	1198
22	663
23	28
10	8
9	1

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

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

Result Grid | Filter Ro

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.

```sql
SELECT
    ROUND(AVG(quantity), 0)
FROM
    (SELECT
        orders.order_date, SUM(order_details.quantity) AS quantity
    FROM
        orders
    JOIN order_details ON orders.order_id = order_details.order_id
    GROUP BY orders.order_date) AS order_quantity
```

Result Grid | Filter R

	round(avg(quantity),0)
▶	138

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

```sql
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;
```

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.

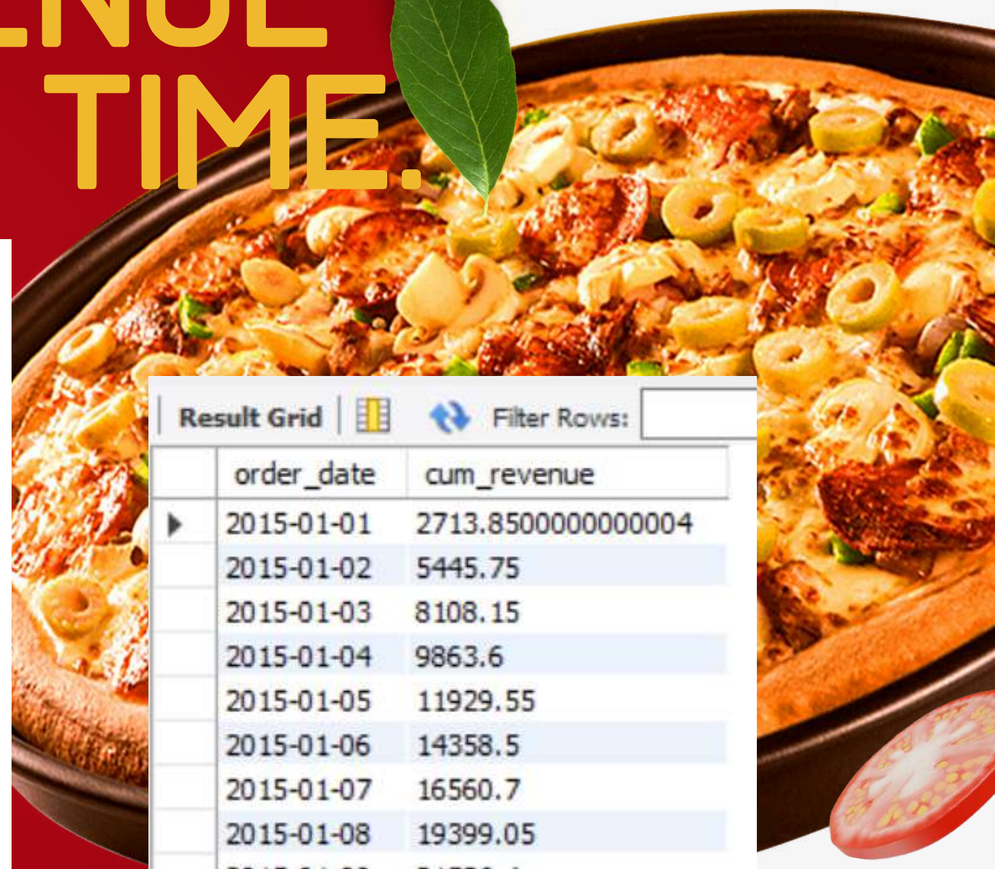
```sql
SELECT
    pizza_types.category,
    ROUND(SUM(order_details.quantity * pizzas.price) / (SELECT
                    ROUND(SUM(order_details.quantity * pizzas.price),
                                2) AS total_revenue
                FROM
                    order_details
                        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;
```

Result Grid | Filte

	category	revenue
▶	Classic	26.91
	Supreme	25.46
	Chicken	23.96
	Veggie	23.68

ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME.

```sql
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;
```



Result Grid | Filter Rows:

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
2015-01-10	23990.350000000002
2015-01-11	25862.65
2015-01-12	27781.7
2015-01-13	29831.300000000003
2015-01-14	32358.700000000004
2015-01-15	34343.50000000001

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

```sql
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 ;
```

Result Grid | Filter Rows:

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	32265.70000000065
The Mexicana Pizza	26780.75
The Five Cheese Pizza	26066.5

LARANA PIZZA

THANK YOU!