

# SQL PROJECT

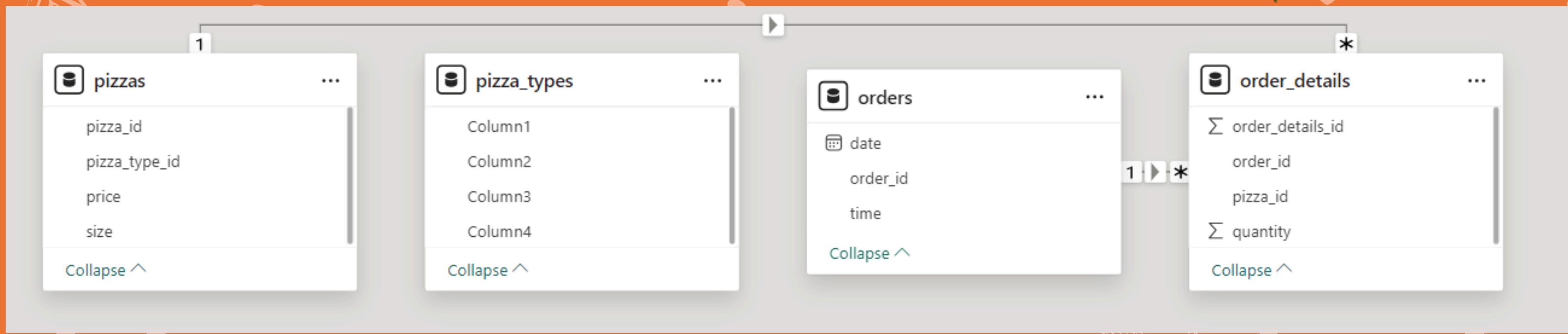
## PIZZA SALES



# *SUMMARY*

Hello! My Name Is Rahul Nehra in this project i have utilize SQL Query to solve SQL Question to solve that are related to pizza sales.

# Data Model



# Database And Tables

```
create database pizzahut;

SELECT * FROM pizzahut.order_details;

use pizzahut;

create table orders (
    order_id int not null,
    order_date date not null,
    order_time time not null,
    primary key(order_id) );

create table order_details (
    order_details_id int not null,
    order_id int not null,
    pizza_id text not null,
    quantity int not null,
    primary key(order_details_id) );
```

*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(order_details.quantity * pizzas.price),  
        2) AS total_sales  
FROM  
    order_details  
    JOIN  
    pizzas ON pizzas.pizza_id = order_details.pizza_id
```

Result Grid	
	total_sales
▶	817860.05



# *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;**

<b>Result Grid</b>		   Filter Rows:
	<b>name</b>	<b>price</b>
▶	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	
M	15385	
S	14137	
XL	544	
XXL	28	



# *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

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

Result Grid | Filter

	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 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



*join relevant tables to find the category wise  
distribution of pizzas*

```
SELECT category, COUNT(*) AS count
FROM pizza_types
GROUP BY category;
```

Result Grid		
	category	count
▶	Chicken	6
	Classic	8
	Supreme	9
	Veggie	9

*group the orders by date and calculate the average number of pizzas order per day*

```
SELECT
    ROUND(AVG(quantity), 0) as avg_pizza_ordered_per_day
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 Rows:	
	avg_pizza_ordered_per_day
▶	138

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

```
SELECT pizza_types.name,  
       sum(order_details.quantity * pizzas.price) 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.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 total percentage contribution of each pizza type to total revenue*

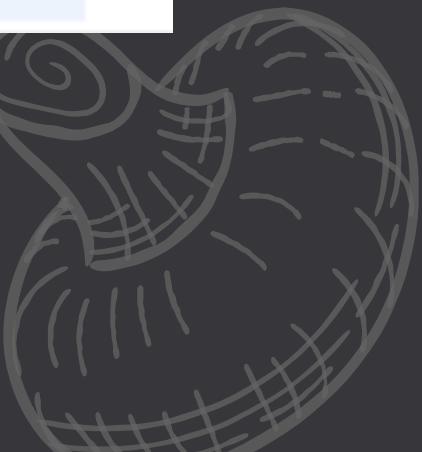
```
SELECT
    pizza_types.category,
    (SUM(order_details.quantity * pizzas.price) / (SELECT
        ROUND(SUM(order_details.quantity * pizzas.price),
        2) AS total_sales
    )
    FROM
        order_details
        JOIN
            pizzas ON pizzas.pizza_id = order_details.pizza_id)) * 100 AS 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;
```

	category	revenue
▶	Classic	26.90596025566967
	Supreme	25.45631126009862
	Chicken	23.955137556847287
	Veggie	23.682590927384577

# *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;
```

Result Grid		
	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



# 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;
```

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

# Thank You

