

# SQL PROJECT ON PIZZA SALES





# HELLO !

I'm Chetan yadav and in this project  
I have utilise SQL Queries to solve question that were  
related to pizza sales.



# QUESTIONS

## Basic:

- 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.



## Intermediate:

- 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.



## Advanced:

- 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.



```
SELECT  
    COUNT(order_id) AS total_orders  
FROM  
    pizzahut.orders;
```

Result Grid

|   | total_orders |
|---|--------------|
| ▶ | 21350        |

# CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

SELECT

```
ROUND(SUM(orders_details.quantity * pizzas.price),  
      2) AS total_sales
```

FROM

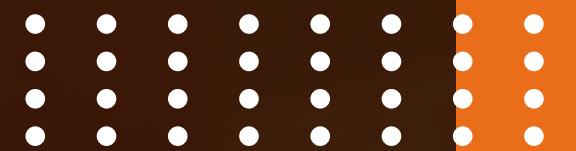
```
pizzahut.orders_details
```

JOIN

```
pizzahut.pizzas ON pizzas.pizza_id = orders_details.pizza_id
```

| Result Grid |             |
|-------------|-------------|
|             | total_sales |
| ▶           | 817860.05   |

# IDENTIFY THE HIGHEST-PRICED PIZZA.



```
SELECT
    pizza_types.name, pizzas.price
FROM
    pizzahut.pizza_types
    JOIN
    pizzahut.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.



```
SELECT
    pizzas.size,
    COUNT(orders_details.order_details_id) AS order_count
FROM
    pizzahut.pizzas
        JOIN
    pizzahut.orders_details ON pizzas.pizza_id = orders_details.pizza_id
GROUP BY pizzas.size
ORDER BY order_count DESC
LIMIT 1;
```

Result Grid | Filter Row

|   | size | order_count |
|---|------|-------------|
| ▶ | L    | 18526       |



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



```
SELECT
    pizza_types.name,
    SUM(orders_details.quantity) AS total_quantity
FROM
    pizzahut.pizza_types
        JOIN
    pizzahut.pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
        JOIN
    pizzahut.orders_details ON orders_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY total_quantity DESC
LIMIT 5;
```

|   | name                       | total_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(orders_details.quantity)
FROM pizzahut.pizzas
    JOIN pizzahut.pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
    JOIN pizzahut.orders_details ON orders_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category;
```

|   | category | SUM(orders_details.quantity) |
|---|----------|------------------------------|
| ▶ | Classic  | 14888                        |
|   | Veggie   | 11649                        |
|   | Supreme  | 11987                        |
|   | Chicken  | 11050                        |

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

```
SELECT  
    HOUR(order_time), COUNT(order_id)  
FROM  
    pizzahut.orders  
GROUP BY HOUR(order_time);
```

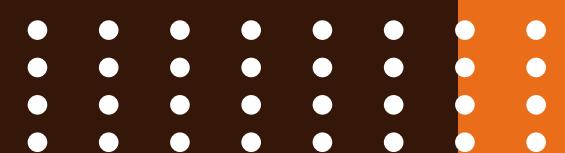
|   | HOUR(order_time) | COUNT(order_id) |
|---|------------------|-----------------|
| ▶ | 11               | 1231            |
|   | 12               | 2520            |
|   | 13               | 2455            |
|   | 14               | 1472            |
|   | 15               | 1468            |
|   | 16               | 1920            |
|   | 17               | 2336            |
|   | 18               | 2399            |
|   | 19               | 2009            |

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

```
SELECT
    pizza_types.name,
    SUM(orders_details.quantity * pizzas.price) AS revenue
FROM
    pizzahut.pizza_types
        JOIN
    pizzahut.pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    pizzahut.orders_details ON orders_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.



```
select pizza_types.category,
round(sum(orders_details.quantity * pizzas.price) / (SELECT
    ROUND(SUM(orders_details.quantity * pizzas.price),
    2) AS total_sales
FROM
    pizzahut.orders_details
    JOIN pizzahut.pizzas
        ON pizzas.pizza_id = orders_details.pizza_id) * 100,2 ) as revenue
from pizzahut.pizza_types
join pizzahut.pizzas
ON pizza_types.pizza_type_id = pizzas.pizza_type_id
join pizzahut.orders_details
ON orders_details.pizza_id = pizzas.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   |

# 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(orders_details.quantity * pizzas.price) as revenue  
from pizzahut.orders  
join pizzahut.orders_details  
ON orders.order_id = orders_details.order_id  
join pizzahut.pizzas  
On pizzas.pizza_id = orders_details.pizza_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            |

# 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(orders_details.quantity * pizzas.price) as revenue
from pizzahut.pizza_types
join pizzahut.pizzas
ON pizza_types.pizza_type_id = pizzas.pizza_type_id
join pizzahut.orders_details
ON orders_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        | 32265.70000000065 |
|   | The Mexicana Pizza           | 26780.75          |
|   | The Five Cheese Pizza        | 26066.5           |

Pizza Sales Presentation

THANK YOU  
FOR ATTENTION

See You Next