

# PIZZAHUT SALES



By Annie Evangelene





# LIST OF CONTENTS

Introduction

Ingredients

Queries &  
Solutions

# INTRODUCTION

Hola! My name is Annie Evangelene, In this project I've used SQL to solve few PIZZA sales related queries.

**Let's start our adventure in the world of pizza!**

# INGREDIENTS

To make delicious pizza sales, you will need the following queries:



- 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.

# SQL QUERY 1

Retrieve the total number of orders placed.

SQL

```
select count(order_id) as TOTAL_ORDERS from pizzahut.dbo.orders;
```

Output

	Results	Messages
	TOTAL_ORDERS	
1	21350	

# SQL QUERY 2

Calculate the total revenue generated from pizza sales.

SQL

```
- select
  round(sum(pizzahut.dbo.order_details.quantity * pizzahut.dbo.pizzas.price),2)
  as TOTAL_SALES
  from pizzahut.dbo.order_details join pizzahut.dbo.pizzas
  on pizzahut.dbo.pizzas.pizza_id = pizzahut.dbo.order_details.pizza_id;
```

Output

	Results	Messages
1	TOTAL_SALES 817860.05	

# SQL QUERY 3

Identify the highest-priced pizza.

SQL

```
select top 1 pizzahut.dbo.pizza_types.name, pizzahut.dbo.pizzas.price  
from pizzahut.dbo.pizza_types join pizzahut.dbo.pizzas  
on pizzahut.dbo.pizza_types.pizza_type_id = pizzahut.dbo.pizzas.pizza_type_id  
order by pizzahut.dbo.pizzas.price desc;
```

Output

Results    Messages

	name	price
1	The Greek Pizza	35.95

# SQL QUERY 4

Identify the most common pizza size ordered.

## SQL

```
- select pizzahut.dbo.pizzas.size, count(pizzahut.dbo.pizzas.size) as PIZZA_Count  
from pizzahut.dbo.pizzas  
group by pizzahut.dbo.pizzas.size  
order by PIZZA_Count desc;
```

## Output

	size	PIZZA_Count
1	S	32
2	L	31
3	M	31
4	XL	1
5	XXL	1

# SQL QUERY 5

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

SQL

```
- select top 5 pizzahut.dbo.pizza_types.name,
sum(pizzahut.dbo.order_details.quantity) as Quantity
from pizzahut.dbo.pizza_types join pizzahut.dbo.pizzas
on pizzahut.dbo.pizza_types.pizza_type_id = pizzahut.dbo.pizzas.pizza_type_id
join pizzahut.dbo.order_details
on pizzahut.dbo.order_details.pizza_id = pizzahut.dbo.pizzas.pizza_id
group by pizzahut.dbo.pizza_types.name
order by Quantity desc;
```

Output

	name	Quantity
1	The Classic Deluxe Pizza	2453
2	The Barbecue Chicken Pizza	2432
3	The Hawaiian Pizza	2422
4	The Pepperoni Pizza	2418
5	The Thai Chicken Pizza	2371

# SQL QUERY 6

Join the necessary tables to find the total quantity of each pizza category ordered.

SQL

```
select pizzahut.dbo.pizza_types.category, sum(pizzahut.dbo.order_details.quantity) as Quantity
from pizzahut.dbo.pizza_types
join pizzahut.dbo.pizzas
on pizzahut.dbo.pizza_types.pizza_type_id = pizzahut.dbo.pizzas.pizza_type_id
join pizzahut.dbo.order_details
on pizzahut.dbo.order_details.pizza_id = pizzahut.dbo.pizzas.pizza_id
group by pizzahut.dbo.pizza_types.category
order by Quantity desc;
```

Output

	category	Quantity
1	Classic	13529
2	Supreme	11987
3	Veggie	11649
4	Chicken	11050
5	Mushroom	1359

# SQL QUERY 7

Determine the distribution of orders by hour of the day.

## SQL

```
SELECT datepart(hour,pizzahut.dbo.orders.time) as Time,  
count(pizzahut.dbo.orders.order_id) as No_of_Orders  
FROM pizzahut.dbo.orders  
group by datepart(hour,pizzahut.dbo.orders.time)  
order by Time;
```

## Output

	Time	No_of_Orders
1	9	1
2	10	8
3	11	1231
4	12	2520
5	13	2455
6	14	1472
7	15	1468
8	16	1920
9	17	2336
10	18	2399
11	19	2009
12	20	1642
13	21	1198
14	22	663
15	23	28

# SQL QUERY 8

Join relevant tables to find the category-wise distribution of pizzas.

SQL

```
- select pizzahut.dbo.pizza_types.category as Category,  
      count(pizzahut.dbo.pizza_types.name)  
   from pizzahut.dbo.pizza_types  
group by category;
```

Output

	Category	(No column name)
1	Mushroom	1
2	Chicken	6
3	Classic	7
4	Supreme	9
5	Veggie	9

# SQL QUERY 9

Group the orders by date and calculate the average number of pizzas ordered per day.

SQL

```
select avg(Quantity) as Avg_Quantity  
from  
(select pizzahut.dbo.orders.date,  
sum(pizzahut.dbo.order_details.quantity) as Quantity  
from pizzahut.dbo.orders  
join pizzahut.dbo.order_details  
on pizzahut.dbo.orders.order_id = pizzahut.dbo.order_details.order_id  
group by pizzahut.dbo.orders.date) as Order_quantity;
```

Output

	Avg_Quantity
1	138

# SQL QUERY 10

Determine the top 3 most ordered pizza types based on revenue.

SQL

```
select top 3 pizzahut.dbo.pizza_types.name ,  
sum(pizzahut.dbo.order_details.quantity * pizzahut.dbo.pizzas.price) as Revenue  
from pizzahut.dbo.pizza_types  
join pizzahut.dbo.pizzas on  
pizzahut.dbo.pizza_types.pizza_type_id = pizzahut.dbo.pizzas.pizza_type_id  
join pizzahut.dbo.order_details on  
pizzahut.dbo.pizzas.pizza_id = pizzahut.dbo.order_details.pizza_id  
group by pizzahut.dbo.pizza_types.name  
order by Revenue desc;
```

Output

	name	Revenue
1	The Thai Chicken Pizza	43434.25
2	The Barbecue Chicken Pizza	42768
3	The California Chicken Pizza	41409.5

# SQL QUERY 11

Calculate the percentage contribution of each pizza type to total revenue. (top 5)

SQL

```
SELECT TOP (5) pizzahut.dbo.pizza_types.name,
    ROUND(COUNT(pizzahut.dbo.order_details.quantity * pizzahut.dbo.pizzas.price) * 100 /
    SUM(pizzahut.dbo.order_details.quantity * pizzahut.dbo.pizzas.price), 2) AS Revenue_Percent
FROM      pizzahut.dbo.pizza_types INNER JOIN
          pizzahut.dbo.pizzas ON pizzahut.dbo.pizza_types.pizza_type_id = pizzahut.dbo.pizzas.pizza_type_id INNER JOIN
          pizzahut.dbo.order_details ON pizzahut.dbo.pizzas.pizza_id = pizzahut.dbo.order_details.pizza_id
GROUP BY pizzahut.dbo.pizza_types.name
ORDER BY Revenue_Percent DESC
```

Output

	name	Revenue_Percent
1	The Big Meat Pizza	7.88
2	The Pepperoni Pizza	7.85
3	The Hawaiian Pizza	7.34
4	"The Pepperoni	7.13
5	The Green Garden Pizza	7.07

# SQL QUERY 12

Analyze the cumulative revenue generated over time.  
(top 10 displayed in output)

SQL

```
select date ,
       sum(Revenue) over(order by date) as Cumm_Rev
  from
  (select pizzahut.dbo.orders.date,
         sum(pizzahut.dbo.order_details.quantity * pizzahut.dbo.pizzas.price) as Revenue
    from pizzahut.dbo.order_details
   join pizzahut.dbo.pizzas on
     pizzahut.dbo.order_details.pizza_id = pizzahut.dbo.pizzas.pizza_id
   join pizzahut.dbo.orders on
     pizzahut.dbo.orders.order_id = pizzahut.dbo.order_details.order_id
  group by pizzahut.dbo.orders.date) as sales;
```

Output

	date	Cumm_Rev
1	2015-01-01	2713.85000228882
2	2015-01-02	5445.7500038147
3	2015-01-03	8108.15000724792
4	2015-01-04	9863.60000801086
5	2015-01-05	11929.5500087738
6	2015-01-06	14358.5000114441
7	2015-01-07	16560.700012207
8	2015-01-08	19399.0500183105
9	2015-01-09	21526.4000225067
10	2015-01-10	23990.350025177

# SQL QUERY 13

Determine the top 3 most ordered pizza types based on revenue for each pizza category.

SQL

```
select top 3 name, Revenue
from
(select name, Category, Revenue,
rank() over(partition by Category order by Revenue desc) as rn
from
(select pizzahut.dbo.pizza_types.name , pizzahut.dbo.pizza_types.category as Category,
sum(pizzahut.dbo.order_details.quantity * pizzahut.dbo.pizzas.price) as Revenue
from pizzahut.dbo.pizza_types
join pizzahut.dbo.pizzas on
pizzahut.dbo.pizza_types.pizza_type_id = pizzahut.dbo.pizzas.pizza_type_id
join pizzahut.dbo.order_details on
pizzahut.dbo.order_details.pizza_id = pizzahut.dbo.pizzas.pizza_id
group by pizzahut.dbo.pizza_types.name , Category) as a) as b
where rn<=3
order by Revenue desc;
```

Output

	name	Revenue
1	The Thai Chicken Pizza	43434.25
2	The Barbecue Chicken Pizza	42768
3	The California Chicken Pizza	41409.5

A festive illustration set against a red background with a white grid. In the center, the words "THANK YOU" are written in large, bold, white capital letters. To the left, a person with dark curly hair and a green shirt holds a slice of pizza. To the right, another person with glasses and a green shirt holds a piece of pie. Above them, a reindeer with a yellow and orange patterned collar and a bell hangs from its neck looks down. The reindeer's collar features a dollar sign (\$) symbol. The scene is decorated with yellow stars and a large yellow starburst at the bottom.

THANK YOU