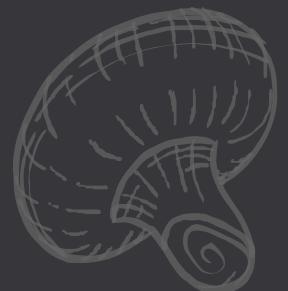


PIZZA SALES PROJECT





Hello!

My name is Riya and I am a recent graduate seeking a career in Data Analytics.

In this project I have solved queries based in pizza sales and to showcase the same this presentation has been created.

Questions for this Project

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.*

Questions for this Project

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.*

Questions for this Project

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.

```
use pizzahut ;  
select count(order_id) as Total_Orders from orders;
```

Result Grid	
	Total_Orders
▶	21350

Calculate the total revenue generated from pizza sales.

```
use pizzahut;

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-priced pizza.

```
use pizzahut;  
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.

```
use pizzas hut;  
  
SELECT  
    pizzas.size,  
    COUNT(order_details.order_details_id) AS total_count  
FROM  
    pizzas  
        JOIN  
            order_details ON pizzas.pizza_id = order_details.pizza_id  
GROUP BY pizzas.size  
ORDER BY total_count DESC  
LIMIT 1;
```

Result Grid | Filter

	size	total_count
▶	L	18526

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

```
use pizzahut;
SELECT
    pizza_types.name,
    SUM(order_details.quantity) AS total_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 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.

```
use pizzahut;
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;
```

	category	quantity
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050

Determine the distribution of orders by hour of the day.

```
use pizzahut;  
SELECT  
    HOUR(order_time), COUNT(order_id)  
FROM  
    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
	20	1642
	21	1198
	22	663
	23	28
	10	8
	9	1

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

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

Result Grid | Filter Rows

	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.

```
use pizzahut;
SELECT
    ROUND(AVG(quantity), 0) AS average_quantity
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;
```

average_quantity
138

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

```
use pizzahut;  
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 pizzas.pizza_id = order_details.pizza_id  
GROUP BY pizza_types.name  
ORDER BY revenue DESC  
LIMIT 3;
```

	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.

```
use pizzahut;
SELECT
    pizza_types.category,
    ROUND(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,
    2) AS revenue
FROM
    pizza_types
    JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
    JOIN
    order_details ON pizzas.pizza_id = order_details.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.

```
use pizzahut;  
  
select order_date,  
sum(revenue) over(order by order_date) as cum_rev  
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;
```

	order_date	cum_rev
▶	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	23990.35000000002
	2015-01-11	25862.65
	2015-01-12	27781.7
	2015-01-13	29831.30000000003
	2015-01-14	32358.70000000004
	2015-01-15	34343.50000000001
	2015-01-16	36937.65000000001

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

```
use pizzahut;
select category, 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 ;
```

	category	name	revenue
▶	Chicken	The Thai Chicken Pizza	43434.25
	Chicken	The Barbecue Chicken Pizza	42768
	Chicken	The California Chicken Pizza	41409.5
	Classic	The Classic Deluxe Pizza	38180.5
	Classic	The Hawaiian Pizza	32273.25
	Classic	The Pepperoni Pizza	30161.75
	Supreme	The Spicy Italian Pizza	34831.25
	Supreme	The Italian Supreme Pizza	33476.75
	Supreme	The Sicilian Pizza	30940.5
	Veggie	The Four Cheese Pizza	32265.70000000065
	Veggie	The Mexicana Pizza	26780.75
	Veggie	The Five Cheese Pizza	26066.5

Conclusion

After completion of this project the conclusions drawn at the end are:

- *Total orders are 21350 and Total sales are 817860.05*
- *Highest priced pizza is The Freek Pizza and the highest ordered pizza size is L.*
- *Top most ordered pizza type is The classic Deluxe Pizza and highest ordered pizza category is Classic.*
- *Average number of pizzas ordered per day is 138.*

Thankyouu !

Through this project I solved 13 questions which included select, from, where, group by, limit, order by statements, joins, subqueries and many more sql commands.

I hope you found this presentation informative and insightful.

Thankyou for your attention and time.