PIZZAHUT

MYSQL PROJECT

PIZZAHUT

MYSQL PROJECT

ABOUT PROJECT

PIZZA SALES ANALYSIS

Basic and Intermediate Insights

By- Roshni

AGENDA

PIZZAHUT

MYSQL PROJECT

Basic Analysis

- Total Orders and Revenue
- Highest-Priced Pizza
- Most Common Pizza Size
- Top 5 Most Ordered Pizzas

Intermediate Analysis

- Pizza Category Quantities
- Order Distribution by Hour
- Category-Wise Pizza Distribution
- Average Pizzas Ordered Per Day
- Top 3 Pizzas by Revenue

BASIC ANALYSIS - TOTAL ORDERS PLACED

Objective: **Find the total number of orders placed.**

Query:

```
-- Retrieve the total number of orders placed.
SELECT
    COUNT(order_id) AS total_order
FROM
    orders;
```

Insight:

Result Grid | Filter Rows:

	total_order
▶	21350

BASIC ANALYSIS - TOTAL REVENUE FROM PIZZA SALES

Objective: Calculate the total revenue generated from pizza sales.

Query:

```
-- 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;
```

Insight:

Result Grid

	total_revenue
▶	817860.05

BASIC ANALYSIS - HIGHEST-PRICED PIZZA

Objective: **Identify the highest-priced pizza.**

Query:

```sql
-- Identify the highest-priced pizza.
SELECT
    pizza_types.name, pizzas.price AS highest_priced_pizza
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
ORDER BY pizzas.price DESC
LIMIT 1;
```

Insight:

Result Grid | Filter Rows:

	name	highest_priced_pizza
▶	The Greek Pizza	35.95

BASIC ANALYSIS - MOST COMMON PIZZA SIZE ORDERED

Objective: Identify the most common pizza size ordered.

Query:

```sql
-- 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;
```

Insight:

Result Grid | Filter Rov

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

BASIC ANALYSIS - TOP 5 MOST ORDERED PIZZAS

Objective: List the top 5 most ordered pizzas along with their quantities

Query:

```sql
-- 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;
```

Insight:

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

INTERMEDIATE ANALYSIS - PIZZA CATEGORY QUANTITIES

Objective: Find the total quantity of pizzas ordered by category.

Query:

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

SELECT
    pizza_types.category,
    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.category
ORDER BY total_quantity DESC;
```

Insight:

Result Grid | Filter Rows:

category	total_quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

INTERMEDIATE ANALYSIS -ORDER DISTRIBUTION BY HOUR

Objective: Determine the distribution of orders by hour of the day.

Query:

```
-- 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);
```

Insight:

Result Grid

hour	order_count
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

INTERMEDIATE ANALYSIS - CATEGORY-WISE PIZZA DISTRIBUTION

Objective: Find the category-wise distribution of pizzas ordered.

Query:

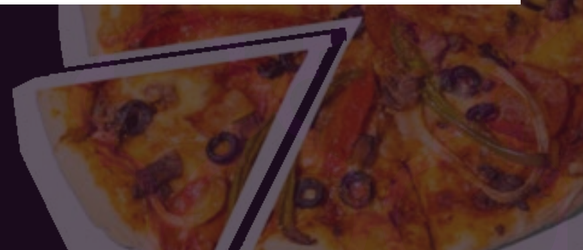
```
-- Join relevant tables to find the category-wise distribution of pizzas.

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

Insight:

Result Grid | Filter Rows:

	category	count(name)
▶	Chicken	6
	Classic	8
	Supreme	9
	Veggie	9



INTERMEDIATE ANALYSIS - AVERAGE PIZZAS ORDERED PER DAY

Objective: Calculate the average number of pizzas ordered per day.

Query:

```sql
-- Group the orders by date and calculate the average number of pizzas ordered 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;
```

Insight:

Result Grid | Filter Rows:

	avg_pizza_ordered_per_day
▶	138

INTERMEDIATE ANALYSIS - TOP 3 PIZZAS BY REVENUE

Objective: Identify the top 3 most ordered pizzas based on revenue.

Query:

```
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;
```

Insight:

Result Grid | Filter Rows:

name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5

CONCLUSION

This project analyzed pizza sales data using SQL to extract key business insights. We explored total orders, revenue, popular pizza types and sizes, peak order hours, and category-wise demand. The analysis shows how SQL can help make informed decisions in a pizza business like Pizza Hut.


PIZZAHUT
MYSQL
PROJECT
THANK YOU