

Where Every Slice is a Taste of Perfection

WELCOME TO PIZZA RESTO

ORDER
NOW

Start Your Slide



ABOUT THE PROJECT

Hey! The name of my project is Pizza hut wherein I've used a dataset and performed the queries on MySQL.

By using DML, DDL, DCL and joins I've learnt how to retrieve all the information from one schema to other.

Database: Pizza hut

Tables used:

- **order_details**
- **orders**
- **pizza_types**
- **pizzas**



OUR HISTORY

Hey! Here I've created a project on MySQL using the following questions:

- Retrieve the total number of orders placed.
- Calculate the total revenue generated from the 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 on hour basis of the day.
- Join relevant tables to find category-wise distribution of pizzas.
- Group the orders by date and calculate the average number pizzas ordered per day.
- Determine the 3 topmost ordered pizza types based on revenue.



QUESTION 1



Retrieve the total number of orders placed.

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

	total_orders
▶	21350



QUESTION 2



Calculate the total revenue generated from the 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;
```

total_sales	817860.05
-------------	-----------

QUESTION 3

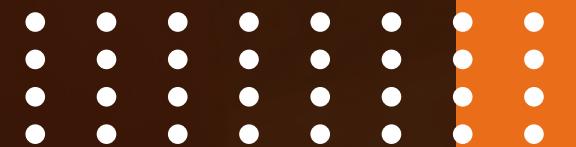
Identify the highest-priced pizza.

```
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;
```

	name	price
▶	The Greek Pizza	35.95

QUESTION 4

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

QUESTION 5



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

QUESTION 6

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;



	category	quantity
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050

QUESTION 7

Determine the distribution of orders on hour basis of the day.

SELECT

HOUR(order_time), COUNT(order_id) AS order_count

FROM

orders

GROUP BY HOUR(order_time);

	HOUR(order_time)	order_count
▶	11	1231
	12	2520
	13	2455
	14	1472
	15	1468
	16	1920
	17	2336
	18	2399
	19	2009
	20	1642
	21	1198

QUESTION 8

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

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

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

QUESTION 9

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

```
SELECT  
    ROUND(AVG(quantity), 0) AS avg_pizza_order_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;
```

	avg_pizza_order_per_day
▶	138

QUESTION 10

Determine the 3 topmost ordered pizza types based on revenue.

```
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 order_details.pizza_id = pizzas.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