

PIZZA





INGOODE COMPANY

+250

Restaurant

4.9

Good Rate

+100

Pizza menu

Q1: Retrieve the total number of orders placed.

SELECT

FROM

orders;

SELECT

COUNT(order_id) AS total_orders

FROM

orders;

About Our Pizza Place

Result Grid	
	total_orders
▶	21350

Q- Calculate the total revenue generated from pizza sales.

Project Plan

```
• 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

Result Grid	
	total_sales
▶	817860.05



Q- 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;
```



Result Grid | Filter Rows:

	name	price
▶	The Greek Pizza	35.95

Q- 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;
```



Result Grid | Filter Rows:

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

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



✓

```
3 • SELECT
4     pizza_types.name, SUM(order_details.quantity) AS quantity
5     FROM
6         pizza_types
7     JOIN
8         pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
9     JOIN
10        order_details ON order_details.pizza_id = pizzas.pizza_id
11    GROUP BY pizza_types.name
12    ORDER BY quantity DESC
13    LIMIT 5;
14
```

Result Grid | Filter Rows: Export: Wrap Cell Content: Fetch 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

Result 1

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

Mission

```
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;
```

RESULT

Result Grid | Filter Rows:

	category	quantity
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050



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);
```



Result Grid | Filter Rows:

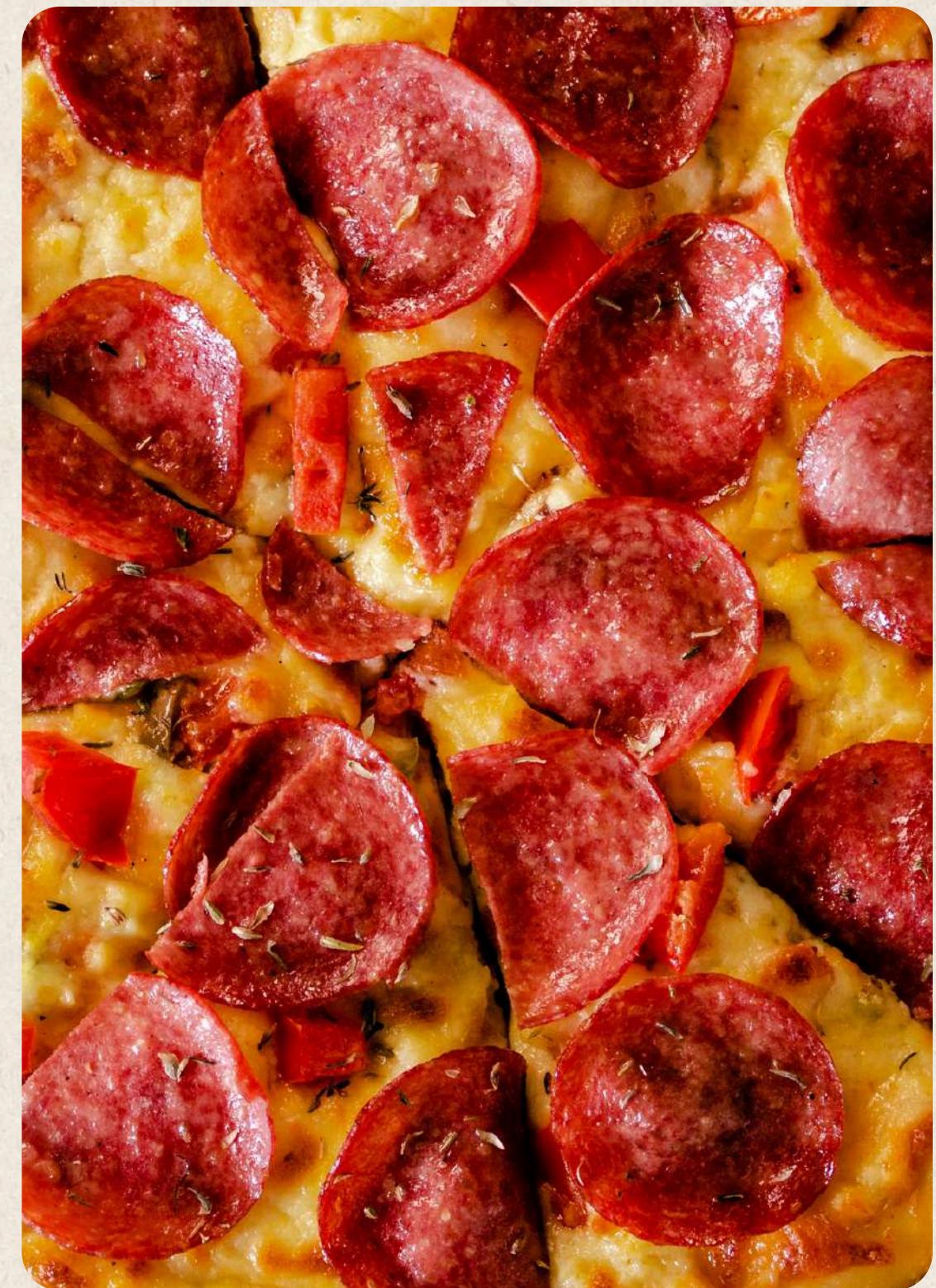
	hour	order_count
▶	11	1231
	12	2520
	13	2455
	14	1472
	15	1468
	16	1920

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

```
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





INGOODE COMPANY

+250

Restaurant

4.9

Good Rate

+100

Pizza menu

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

• **SELECT**

```
ROUND(AVG(quantity), 0) AS avg_pizza_orderd_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;
```

Result Grid



Filter Row

	avg_pizza_orderd_per_day
▶	138

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

```
3 • SELECT
4     pizza_types.name,
5         SUM(order_details.quantity * pizzas.price) AS revenue
6 FROM
7     pizza_types
8     JOIN
9     pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
10    JOIN
11    order_details ON order_details.pizza_id = pizzas.pizza_id
12    GROUP BY pizza_types.name
13    ORDER BY revenue DESC
14    LIMIT 3;
```

Result Grid | Filter Rows: Export: Wrap Cell Content: Fetch rows:

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





```
3 •   select pizza_types.category,
4   round(sum(order_details.quantity*pizzas.price)/ (SELECT
5   ROUND(SUM(order_details.quantity * pizzas.price),
6   2) AS total_sales
7   from order_details join pizzas on pizzas.pizza_id = order_details.pizza_id) * 100,2) as revenue
8   from pizza_types join pizzas
9   on pizza_types.pizza_type_id = pizzas.pizza_type_id
10  join order_details
11  on order_details.pizza_id = pizzas.pizza_id
12  group by pizza_types.category order by revenue desc;
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

category	revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68



```
1   -- Determine the top 3 most ordered pizza types based on revenue for each pizza category.  
2  
3 •  select name, revenue from  
4   (select category, name, revenue,  
5    rank() over(partition by category order by revenue desc) as rn  
6   from  
7   (select pizza_types.category, pizza_types.name,  
8    sum((order_details.quantity) * pizzas.price) as revenue  
9    from pizza_types join pizzas  
10   on pizza_types.pizza_type_id = pizzas.pizza_type_id  
11   join order_details  
12   on order_details.pizza_id = pizzas.pizza_id  
13   group by pizza_types.category, pizza_types.name) as a) as b  
14  where rn <= 3;
```

Result Grid | Filter Rows: Export: Wrap Cell Content:

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



Thank You

Best Pizza

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

End Slide