


SQL Data Analysis Project

Analysis of pizza sales project
By Isha Gupta











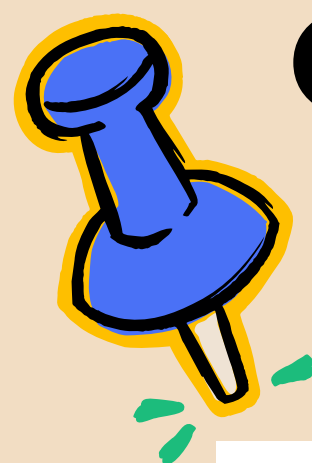
Retrieve the total number of orders placed.



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



Result Grid			
	total_orders		
	21350		



Calculate the total revenue generated from pizza sales.



SELECT

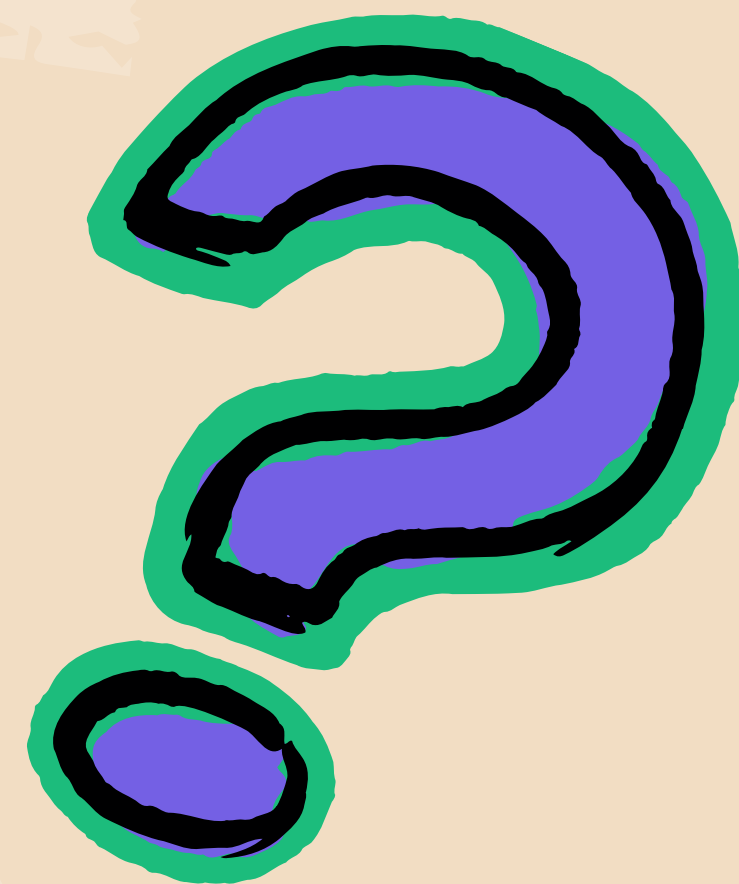
```
ROUND(SUM(orders_details.quantity * pizzas.price),  
      2) AS total_sales
```

FROM

```
orders_details
```

JOIN

```
pizzas ON pizzas.pizza_id = Orders_details.pizza_id;
```



Result Grid		
	total_orders	
▶	21350	

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

	name	price
▶	The Greek Pizza	35.95

Identify the most common pizza size ordered.

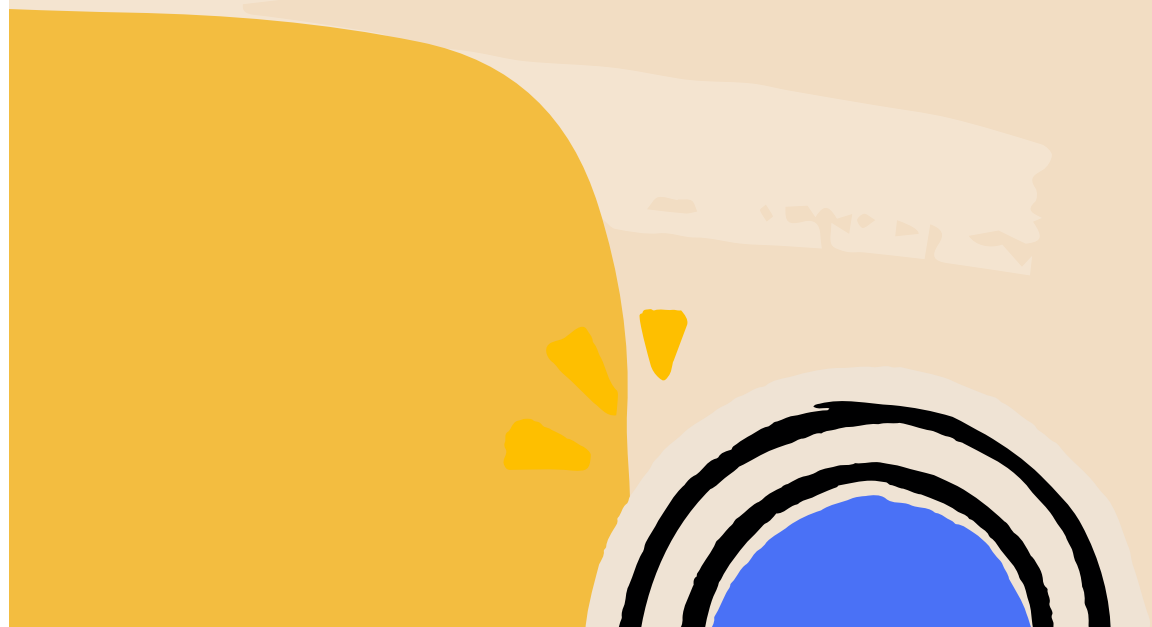
```
SELECT
    pizzas.size,
    COUNT(orders_details.order_details_id) AS order_count
FROM
    pizzas
    JOIN
    orders_details ON pizzas.pizza_id = orders_details.pizza_id
GROUP BY pizzas.size
ORDER BY order_count DESC;
```

Result Grid			Filter R
	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.

```
SELECT
    pizza_types.name, SUM(orders_details.quantity) AS quantity
FROM
    pizza_types
    JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
    JOIN
    orders_details ON orders_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY quantity DESC
LIMIT 5;
```

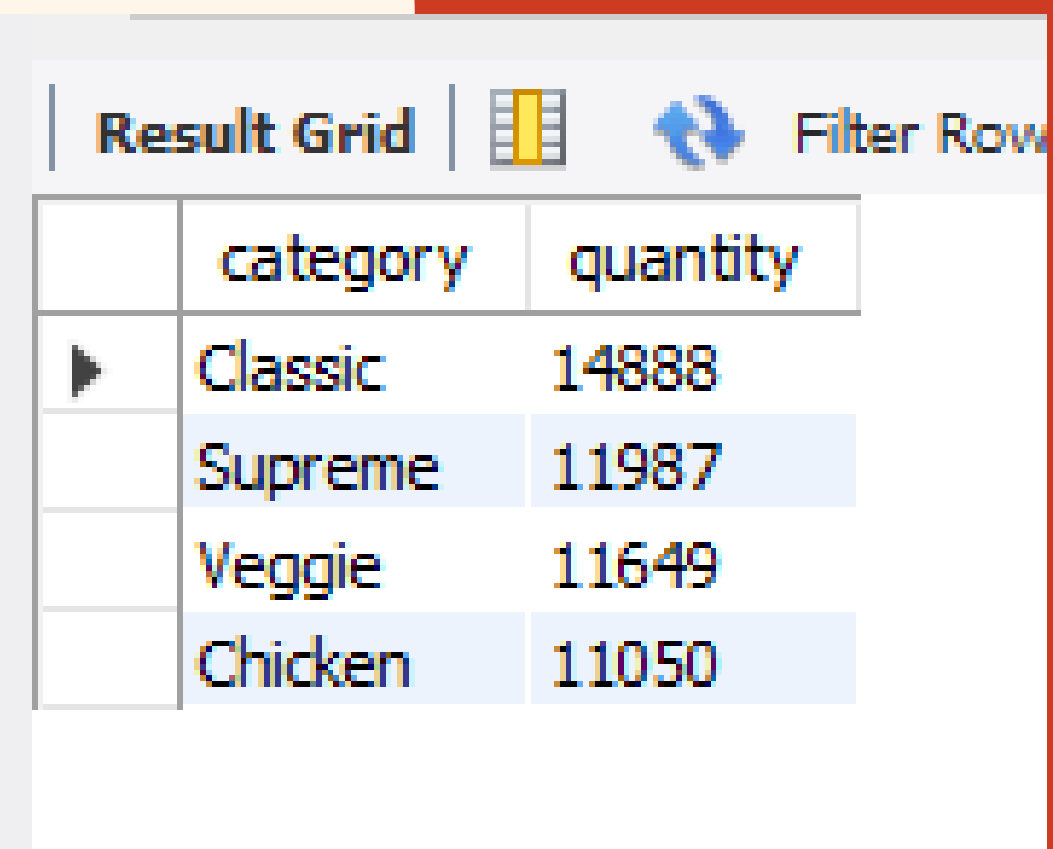


	name	quantity
▶	The Classic Deluxe Pizza	2453
	The Barbecue Chicken Pizza	2432
	The Hawaiian Pizza	2422
	The Pepperoni Pizza	2418
	The Thai Chicken Pizza	2371

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



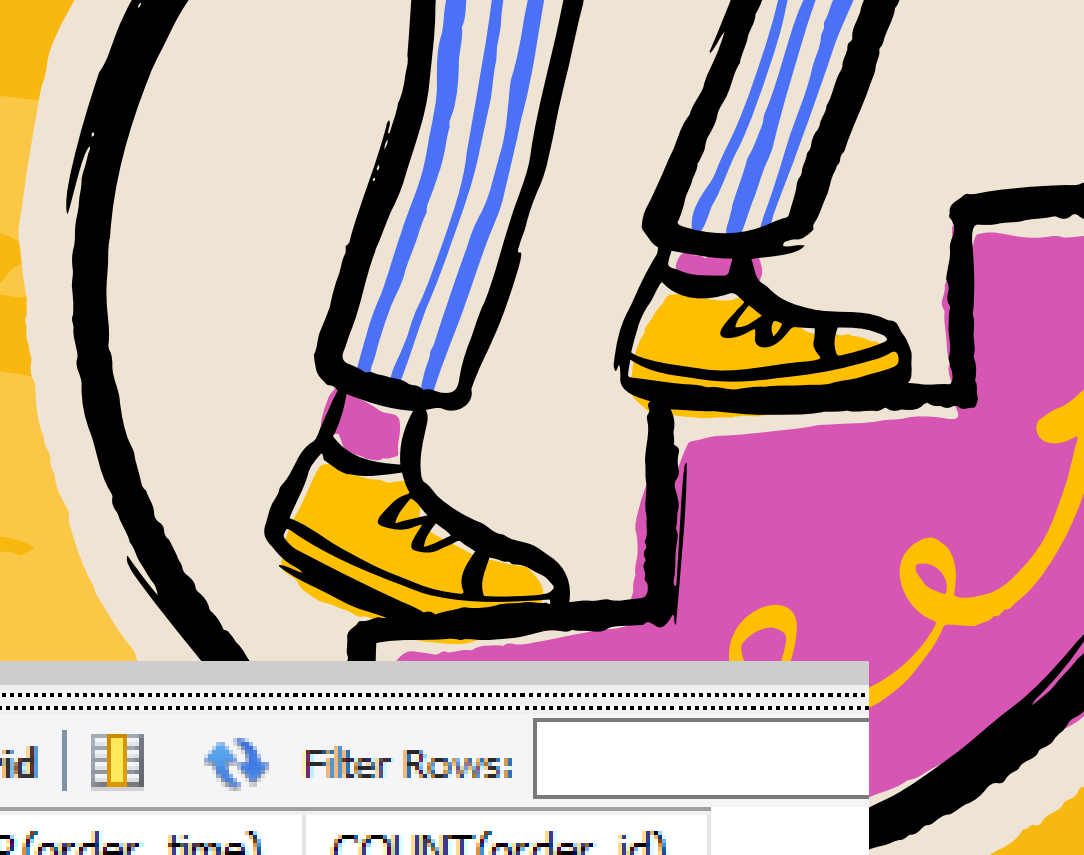
```
SELECT
    pizza_types.category,
    SUM(orders_details.quantity) AS quantity
FROM
    pizza_types
    JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
    JOIN
    orders_details ON orders_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
ORDER BY quantity DESC;
```



	category	quantity
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050



Determine the distribution of orders by hour of the day.



```
SELECT
    HOUR(order_time), COUNT(order_id)
FROM
    orders
GROUP BY HOUR(order_time);
```

Result Grid			Filter Rows:
	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	

Result 17 x





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	

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

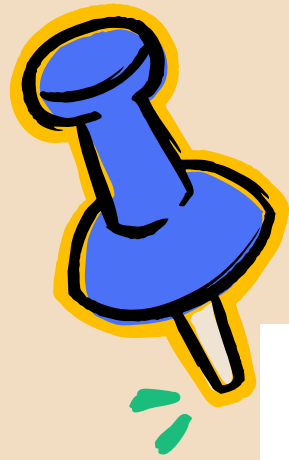
```
SELECT
    ROUND(AVG(quantity), 0) as avg_pizza_per_day
FROM
    (SELECT
        orders.order_date, SUM(orders_details.quantity) AS quantity
    FROM
        orders
    JOIN orders_details ON orders.order_id = orders_details.order_id
    GROUP BY orders.order_date) AS order_quantity;
```

Result Grid |   Filter R

	avg_pizza_per_day
--	-------------------

▶	138
---	-----

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



```
SELECT
  pizza_types.name,
  SUM(orders_details.quantity * pizzas.price) AS revenue
FROM
  pizza_types
  JOIN
  pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
  JOIN
  orders_details ON orders_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY revenue DESC
LIMIT 3;
```

Result Grid			Filter Rows:
	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.

```
SELECT
    pizza_types.category,
    ROUND(SUM(orders_details.quantity * pizzas.price) / (SELECT
        ROUND(SUM(orders_details.quantity * pizzas.price),
            2) AS total_sales
        FROM
            orders_details
            JOIN
                pizzas ON pizzas.pizza_id = orders_details.pizza_id) * 100,
        2) AS revenue
FROM
    pizza_types
    JOIN
        pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
    JOIN
        orders_details ON orders_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category
ORDER BY revenue DESC;
```

Result Grid			Filter Rows
	category	revenue	
▶	Classic	26.91	
	Supreme	25.46	
	Chicken	23.96	
	Veggie	23.68	

Thank You!

