



SQL PROJECT ON PIZZA SALES

Presented by Vijay Kumar

DOCUMENTATION



HELLO !

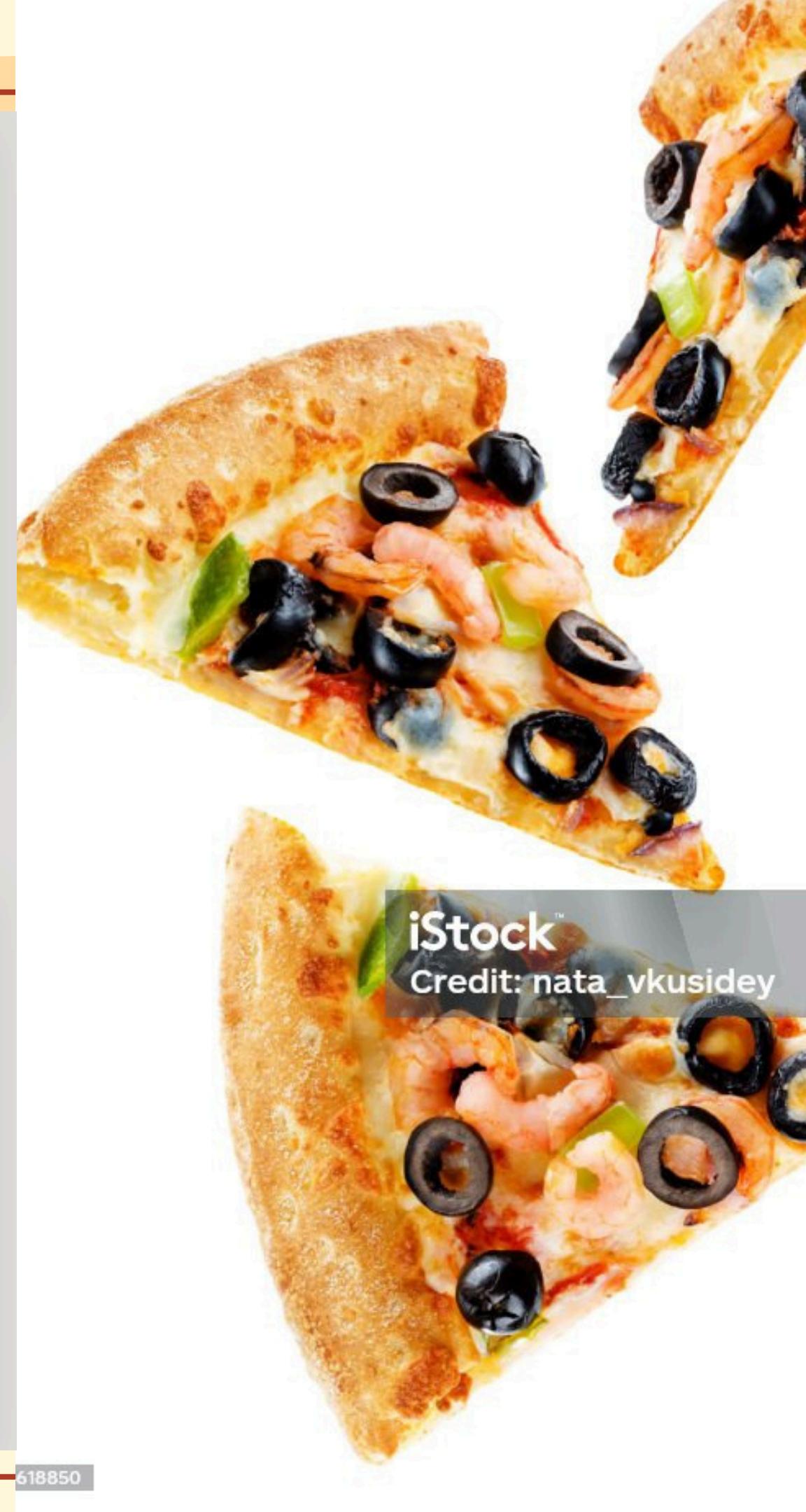
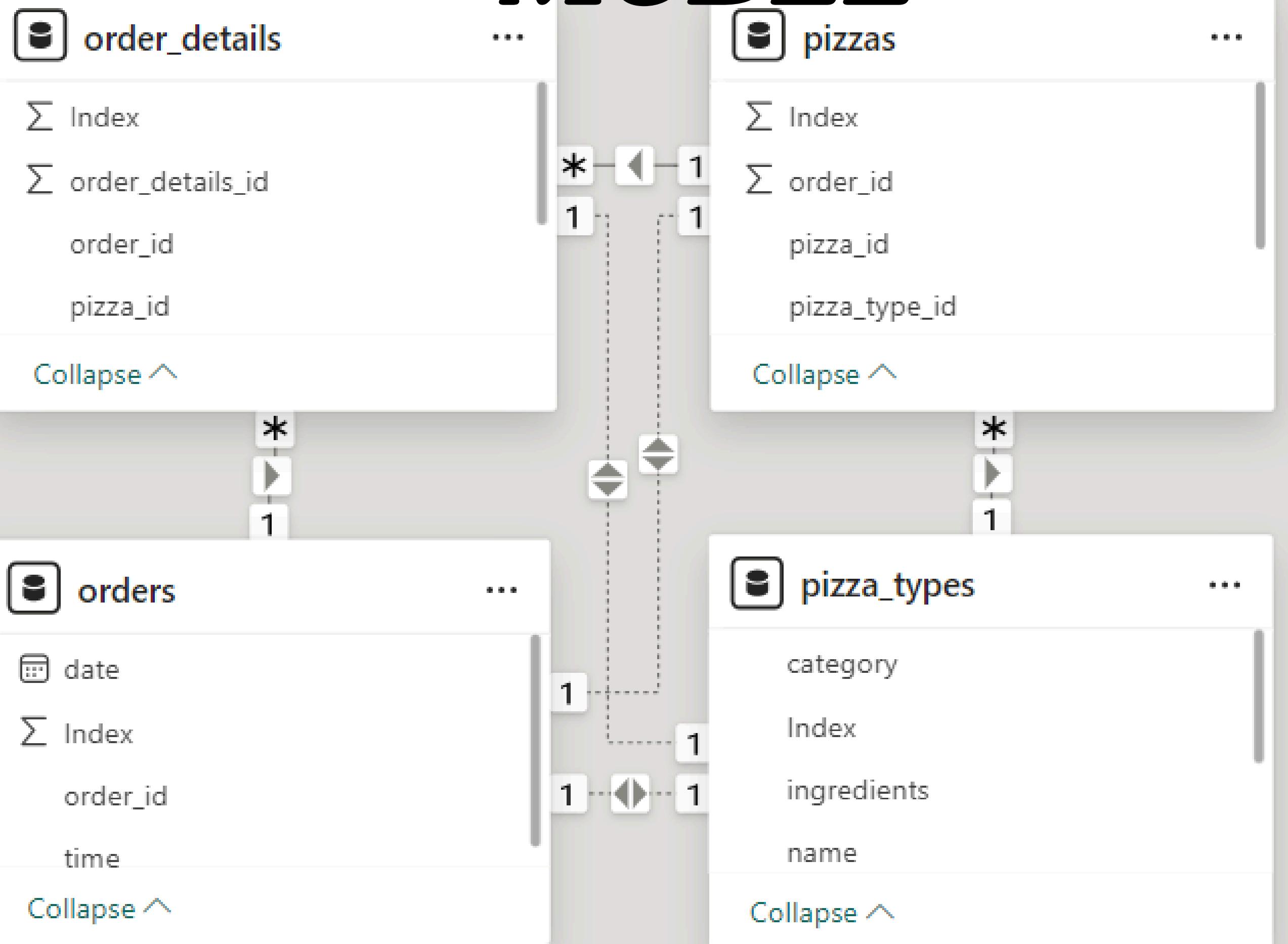
My name is Vijay Kumar and in this project, I have utilize SQL queries to solve queries that is related to Pizza sales.

PROJECT GOALS

1. Retrieve the total number of orders placed.
2. Calculate the total revenue generated from pizza sales.
3. Identify the highest-priced pizza.
4. Identify the most common pizza size ordered.
5. List the top 5 most ordered pizza types along with their quantities.
6. Join the necessary tables to find the total quantity of each pizza category ordered.
7. Determine the distribution of orders by hour of the day.
8. Join relevant tables to find the category-wise distribution of pizzas.
9. Group the orders by date and calculate the average number of pizzas ordered per day.
10. Determine the top 3 most ordered pizza types based on revenue.



MODEL





RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED

```
SELECT  
    COUNT(ORDER_ID)  
FROM  
    ORDERS;
```

Result Grid	
	COUNT(ORDER_ID)
▶	21350



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;

Result Grid	
	Total_Revenue
▶	817860.05



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 Row

	name	price
▶	The Greek Pizza	35.95



IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.

```
SELECT
    pizzas.size,
    MAX(order_details.Order_details_id) AS order_count
FROM
    pizzas
    JOIN
    order_details ON pizzas.pizza_id = order_details.Pizza_
GROUP BY pizzas.size
ORDER BY order_count;
```

Result Grid

	size	order_count
▶	XXL	47490
	XL	48541
	M	48616
	L	48619
	S	48620

Result 1



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;
```

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



JOIN THE NECESSARY TABLES TO FIND THE TOTAL QUANTITY OF EACH PIZZA CATEGORY ORDERED.

SELECT

`pizza_types.category, SUM(order_details.Quantity) AS qty`

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.category`

ORDER BY `qty DESC;`

Result Grid

	<code>category</code>	<code>qty</code>
1	Supreme	446166
2	Veggie	446166
3	Classic	396592
4	Chicken	297444



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

	HOUR	ORDER_COUNT
▶	9	1
	10	8
	11	1231
	12	2520
	13	2455
	14	1472
	15	1468
	16	1970

Result 5 ×



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)  
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 QTY;
```

[Result Grid](#) |   [Filter Rows](#)

ROUND(AVG(QUANTITY), 0)

138



DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE.

SELECT

```
    pizza_types.name,  
    SUM(pizzas.price * order_details.Quantity) AS revenue
```

FROM

```
    pizza_types
```

JOIN

```
    pizzas ON pizza_types.pizza_type_id = pizza_types.pizza_type_id
```

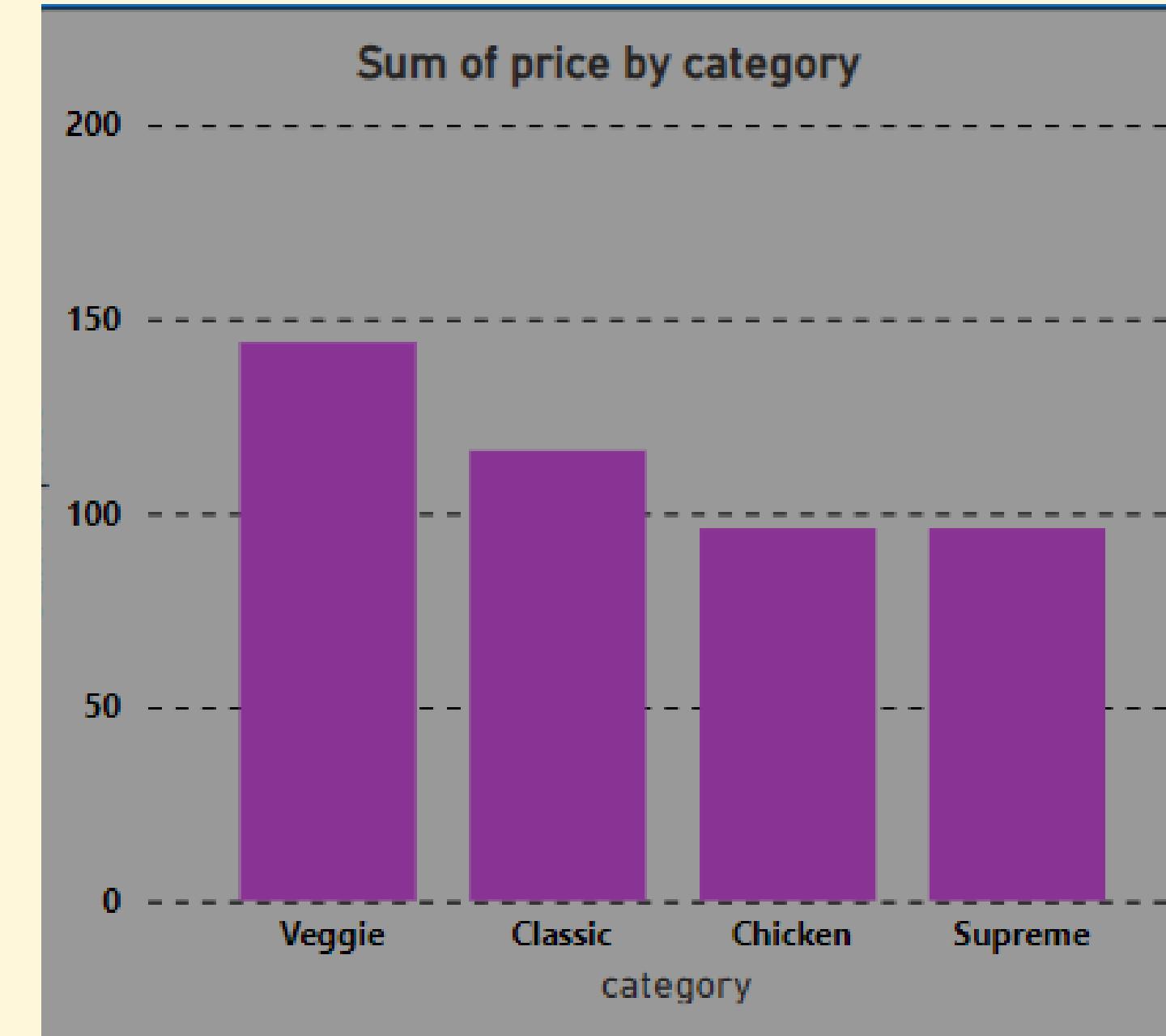
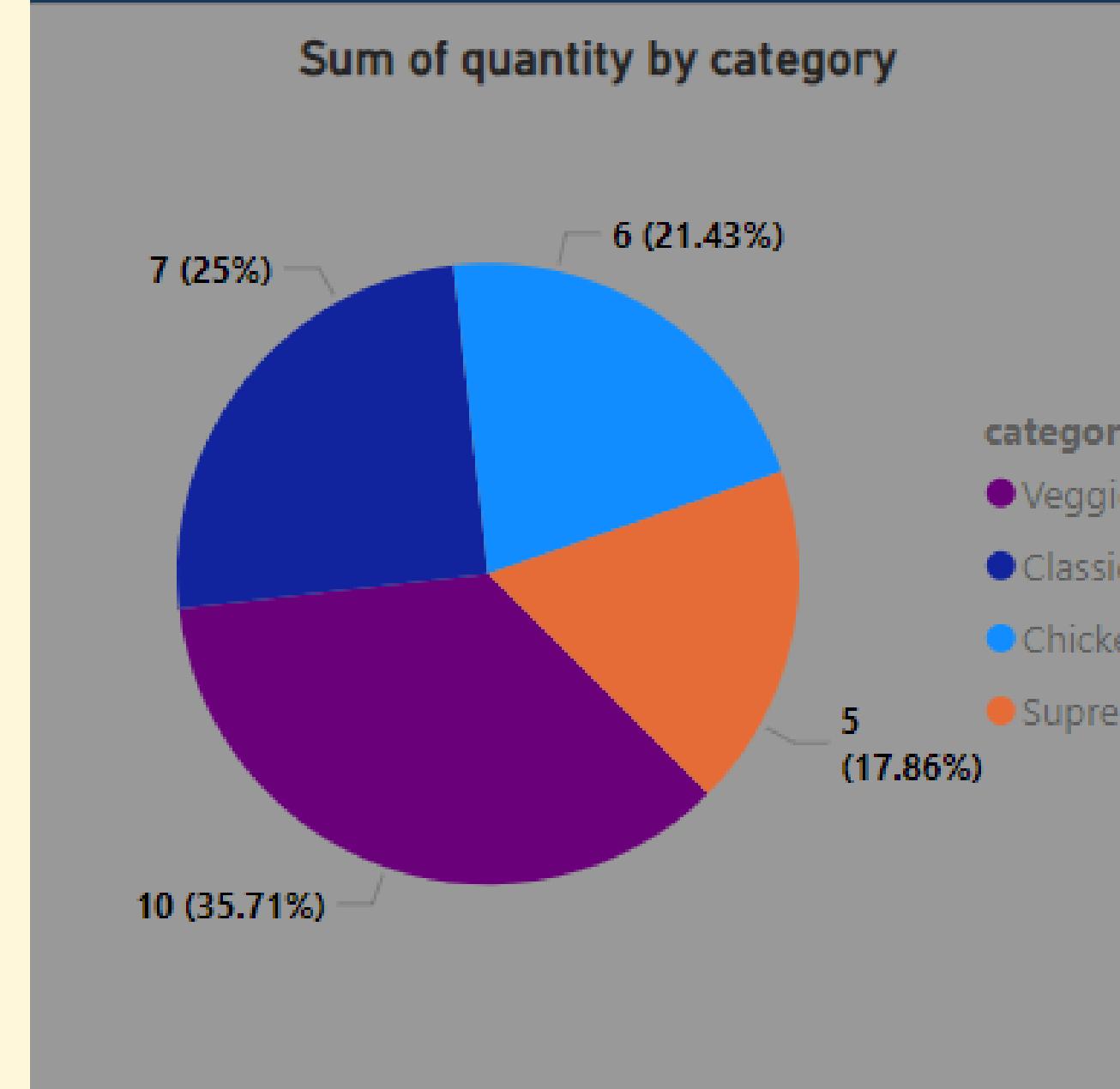
JOIN

```
    order_details ON order_details.Pizza_id = pizzas.pizza_id
```

Result Grid |  Filter Rows:

	name	revenue
1	The Barbecue Chicken Pizza	817860.049999993
2	The California Chicken Pizza	817860.049999993
3	The Chicken Alfredo Pizza	817860.049999993

CHART



**THANK YOU
EVERYONE**