

Pizza_Sales Related Questions

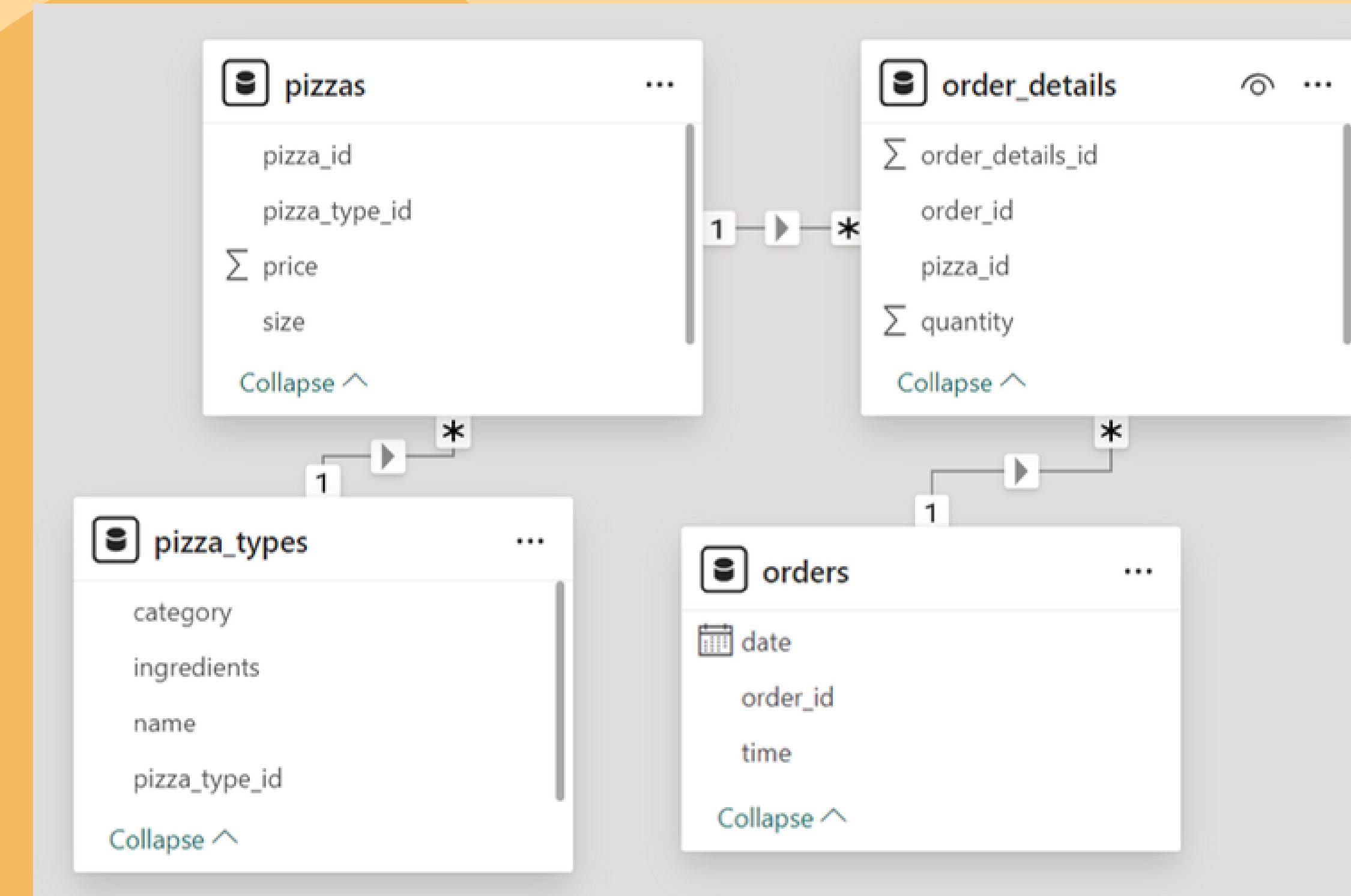
PIZZA SALES

Hello!

My name is Vedant Saraf. This is a SQL project , in this project, I have utilized the SQL query to solve the quetions that is related to pizza Sales.

Schema of the Project

There are four dataset related to the pizza and order. I have imported the data in table from these datasets.



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(order_details.quantity * pizzas.price),  
        2) AS Total_sales  
  
FROM  
    order_details  
    JOIN  
    pizzas ON order_details.pizza_id = pizzas.pizza_id;
```

Result Grid	
	Total_sales
▶	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 Rows:

	name	price
▶	The Greek Pizza	35.95

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

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 pizzas.pizza_id = order_details.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(order_details.quantity) AS Quantity
```

FROM

```
    pizza_types
```

JOIN

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

JOIN

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

GROUP BY pizza_types.category

ORDER BY Quantity ASC;

	category	Quantity
▶	Chicken	11050
	Veggie	11649
	Supreme	11987
	Classic	14888

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);
```

	Hour	Order_Count
▶	11	1231
	12	2520
	13	2455
	14	1472
	15	1468
	16	1920
	17	2336

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_pizzas_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;
```

Result Grid |  Filter Rows: 

Avg_pizzas_ordered_per_day
138

138



Determine the top 3 most 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 pizzas.pizza_id = order_details.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

Determine the top 3 most ordered pizza types based on revenue

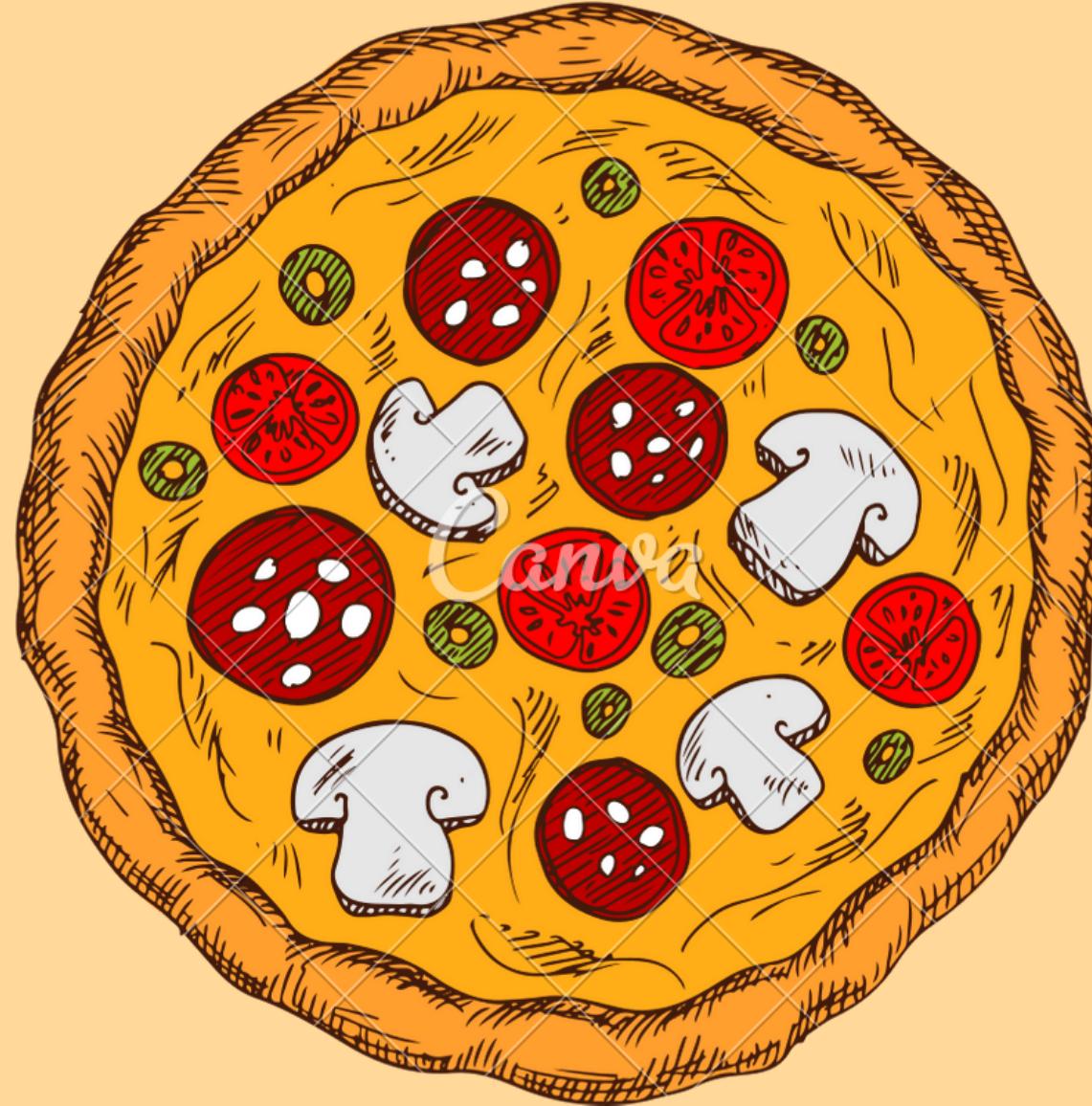
```
SELECT
    pizza_types.category,
    ROUND(SUM(order_details.quantity * pizzas.price) / (SELECT
        ROUND(SUM(order_details.quantity * pizzas.price),
        2) AS Total_sales
    )
    FROM
        order_details
        JOIN
            pizzas ON order_details.pizza_id = pizzas.pizza_id) * 100,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

	category	Revenue
▶	Classic	27
	Supreme	25
	Veggie	24
	Chicken	24

Analyze the cumulative revenue generated over time

```
select order_date, sum(Revenue) over(order by order_date) as cum_rev
from
  (select orders.order_date, sum(order_details.quantity*pizzas.price) as Revenue
  from order_details join pizzas
  on order_details.pizza_id = pizzas.pizza_id
  join orders
  on order_details.order_id = orders.order_id
  group by orders.order_date) as Sales;
```

	order_date	cum_rev
▶	2015-01-01	2713.8500000000004
	2015-01-02	5445.75
	2015-01-03	8108.15
	2015-01-04	9863.6
	2015-01-05	11929.55



Pizza_Sales Present

**THANK
YOU**