

SQL PROJECT ON PIZZA'S SALES

Home

About

Contact





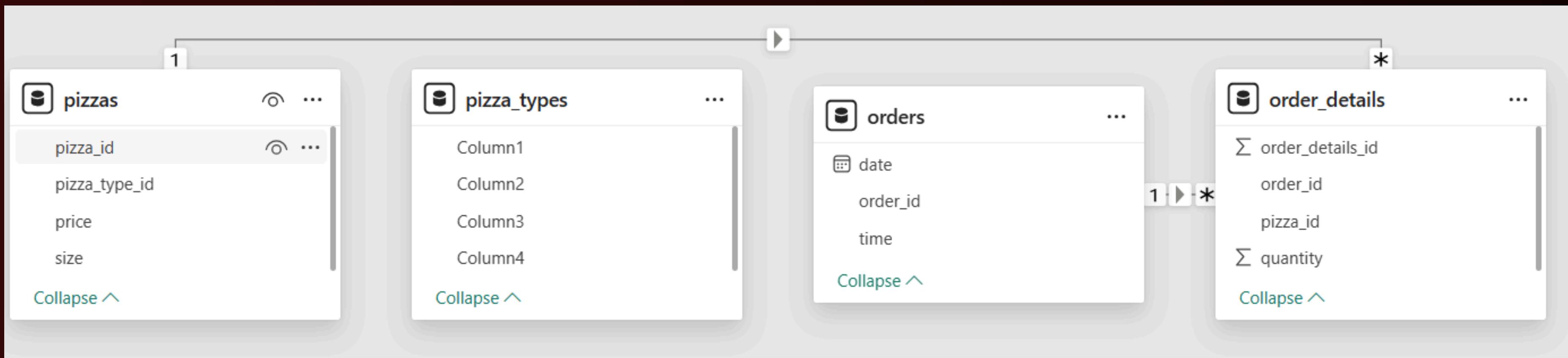
Contact



ABOUT ME
I AM SONU KUMAR SINGH. IN THIS
PROJECT, I HAVE USED SQL QUERIES TO
ANALYSE AND SOLVE QUESTIONS
RELATED TO PIZZA SALES.

[Home](#)[About](#)[Contact](#)

TABLE SCHEMA





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 pizzas.pizza_id = order_details.pizza_id
```



Result Grid	
	total_sales
▶	817036.05



IDENTIFY THE HIGHEST-PRICED PIZZA.

[Home](#)[Contact](#)

```
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.

Contact

```
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
LIMIT 1;
```

Result Grid | Filter

	size	order_count
▶	L	18526



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	2429
	The Hawaiian Pizza	2420
	The Pepperoni Pizza	2414
	The Thai Chicken Pizza	2369



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

Contact

```
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	14874
	Supreme	11977
	Veggie	11640
	Chicken	11034



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
	17	2226



JOIN RELEVANT TABLES TO FIND THE CATEGORY-WISE DISTRIBUTION OF PIZZAS.

Contact

```
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.

[About](#)[Contact](#)

```
SELECT
    ROUND(AVG(quantity), 0) as avg_pizza_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	
	avg_pizza_ordered_per_day
▶	138



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

[Home](#)[About](#)[Contact](#)

```
select pizza_types.name,sum(order_details.quantity*pizzas.price) as revenue 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
```

Result Grid | Filter Rows:

	name	revenue
▶	The Thai Chicken Pizza	43392.75
	The Barbecue Chicken Pizza	42713.75
	The California Chicken Pizza	41329.75



Home

About

Contact

**THANK YOU
FOR ATTENTION.**