



PIZZA





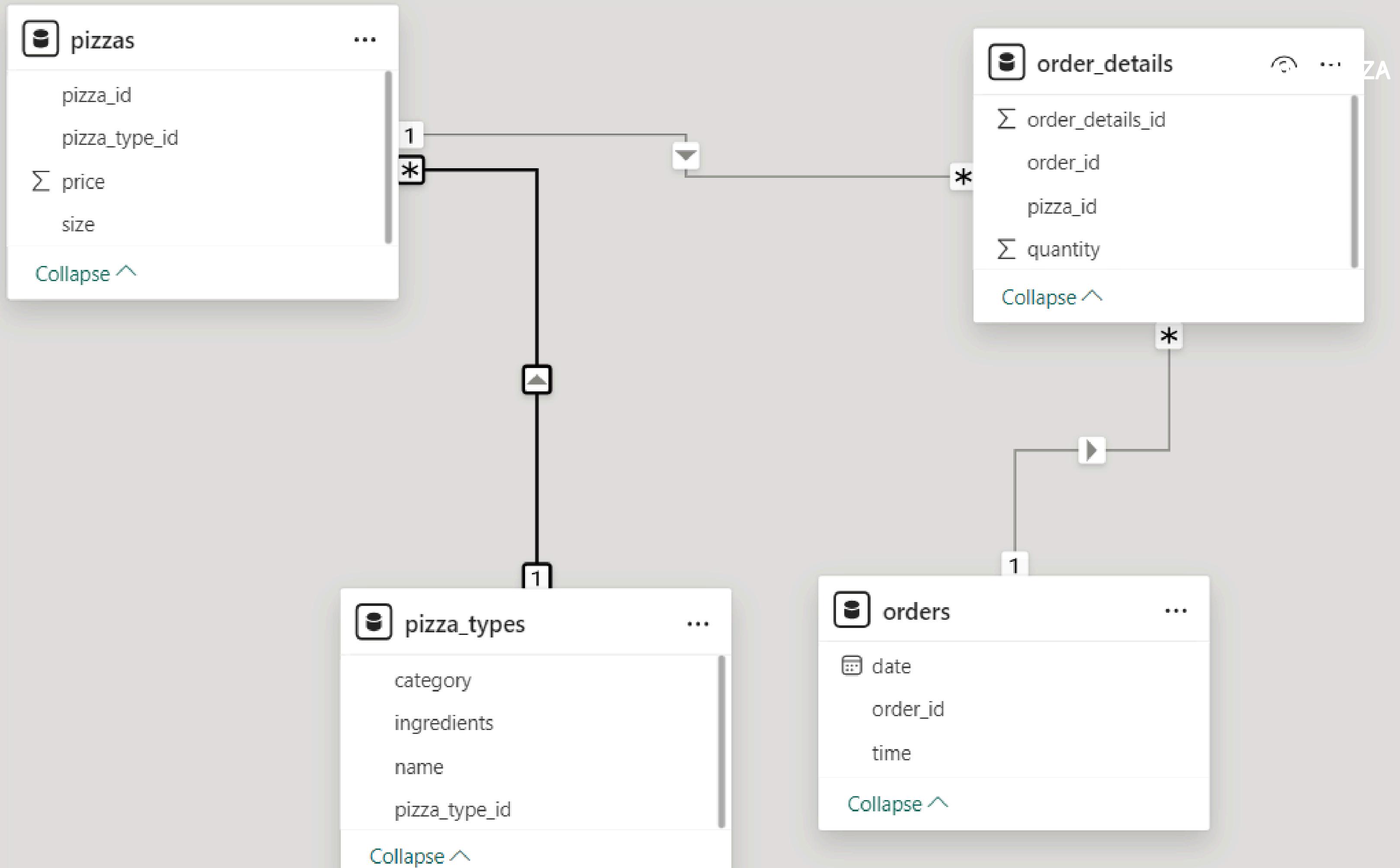
HELLO !

MY NAME IS KANCHAN AND IN THIS PROJECT I HAVE
UTILIZED SQL QUERIES TO SOLVE THE QUESTION THAT
WERE RELATED TO PIZZA SALES.

INTRODUCTION

THE PIZZA SALES SQL PROJECT IS A DATA ANALYSIS PROJECT THAT FOCUSES ON EXTRACTING BUSINESS INSIGHTS FROM A FICTIONAL PIZZA RESTAURANT'S SALES DATABASE. THE GOAL IS TO USE SQL QUERIES TO ANALYZE SALES TRENDS, CUSTOMER BEHAVIOR, AND OPERATIONAL PERFORMANCE TO HELP THE BUSINESS MAKE DATA-DRIVEN DECISIONS.

THE OBJECTIVES OF THE PROJECT IS ANALYSE SALES PERFORMANCE BASED ON DATE, TIME, CATEGORY. ALSO IDENTIFY TOP SELLING PIZZA TYPES AND SIZES. UNDERSTAND CUSTOMER PREFERENCE AND PEAK ORDERING TIME



RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED

SELECT

COUNT(order_id) AS total_orders

FROM

orders;

	total_orders
▶	21350

CALCULATE 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
```

total_sales
817860.05

IDENTIFY THE HIGHEST PRICE 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;

	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 order_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(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	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);

	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

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

SELECT

category, COUNT(name)

FROM

pizza_types

GROUP BY category;

	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 avg(quantity) 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;
```

	avg(quantity)
▶	138.4749

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 pizzas.pizza_type_id = pizza_types.pizza_type_id  
join order_details  
on order_details.pizza_id=pizzas.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

THANK YOU