



MADE BY
Kush Gupta

PIZZA SALES SQL PROJECT



ABOUT

PROJECT TITLE: PIZZA SALES ANALYSIS USING SQL



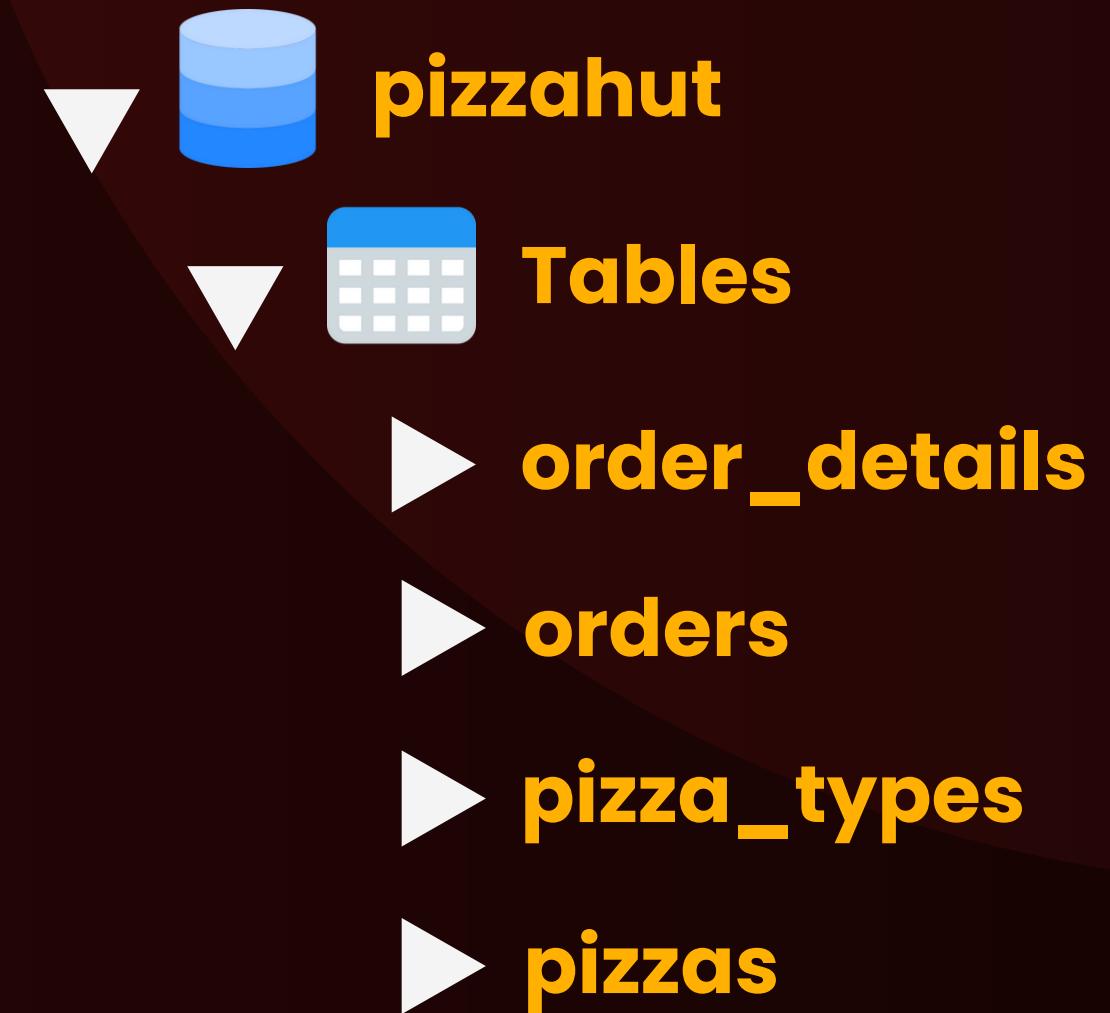
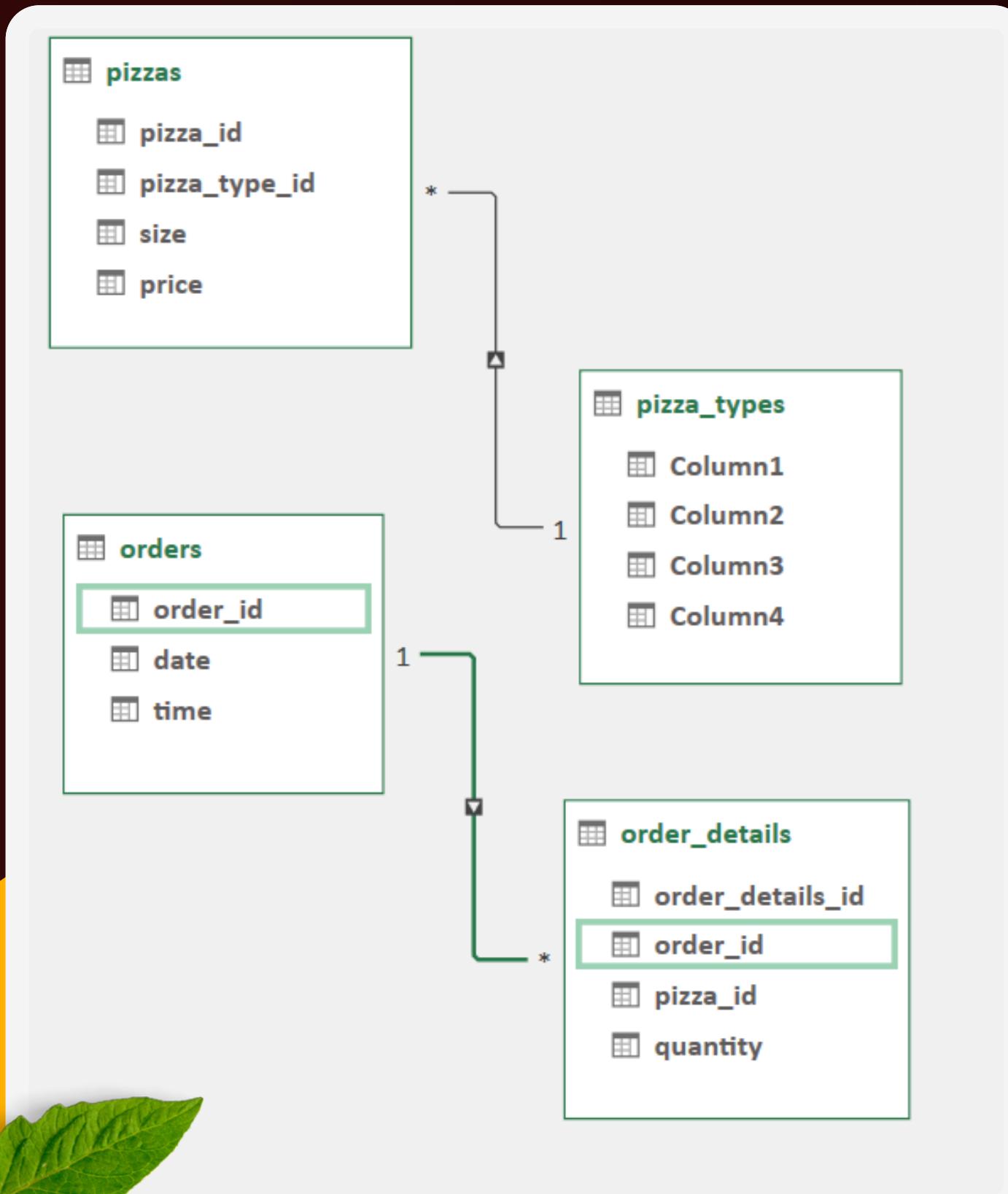
Objective:

To uncover key business insights from a pizza sales dataset using structured queries and analytical functions.

Outcome:

- Extracted meaningful metrics to help the business understand:
 - Best-selling pizza categories
 - Sales performance trends
 - Revenue contribution insights

DATABASE





QUERY 1

- `create database pizzahut;`
- `use pizzahut;`
- `create table orders(`
 `order_id int primary key,`
 `order_date date not null,`
 `order_time time not null`
`);`
- `create table order_details(`
 `order_details_id int primary key,`
 `order_id int not null,`
 `pizza_id text not null,`
 `quantity int not null`
`);`
- `select * from order_details;`

	order_details_id	order_id	pizza_id	quantity
1	1	1	hawaiian_m	1
2	2	2	classic_dlx_m	1
3	2	2	five_cheese_l	1
4	2	2	ital_supr_l	1
5	2	2	mexicana_m	1
6	2	2	thai_ckn_l	1
7	3	3	ital_supr_m	1
8	3	3	prsc_argla_l	1
9	4	4	ital_supr_m	1
10	5	5	ital_supr_m	1
11	6	6	bbq_ckn_s	1
12	6	6	the_greek_s	1
13	7	7	spinach_supr_s	1



TASK 1

RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.

```
select count(order_id) as total_orders  
from orders;
```

Result Grid	
	total_orders
21350	



TASK 2

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
817860.05	



TASK 3

IDENTIFY THE HIGHEST-PRICED PIZZA.

```
select pizza_types.name, pizzas.price  
from  
pizza_types  
join  
pizzas  
on  
pizzas.pizza_type_id = pizza_types.pizza_type_id  
order by pizzas.price desc  
limit 1;
```

Result Grid		Filter Rows:
	name	price
▶	The Greek Pizza	35.95



TASK 4

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.

```
select pizzas.size, count(order_details.quantity) as order_count
from pizzas
join
order_details
on
order_details.pizza_id = pizzas.pizza_id
group by pizzas.size
order by order_count desc
limit 1;
```

	size	order_count
	L	18526



TASK 5

LIST THE TOP 5 MOST ORDERED PIZZA TYPES ALONG WITH THEIR QUANTITIES.

```
select pizza_types.name, pizza_types.category, sum(order_details.quantity) as total_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, pizza_types.category
order by total_quantity desc
limit 5;
```

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



TASK 6

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

```
select pizza_types.category, sum(order_details.quantity) as total_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;
```

Result Grid | Filter Rows:

	category	total_quantity
▶	Classic	14888
	Veggie	11649
	Supreme	11987
	Chicken	11050



TASK 7

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.

```
select count(order_id) as order_count, hour(order_time) as hours  
from orders  
group by hours  
order by hours;
```

Result Grid | Filter Row

	order_count	hours
1	9	
8	10	
1231	11	
2520	12	
2455	13	
1472	14	
1468	15	
1920	16	
2336	17	
2399	18	
2009	19	
1642	20	
1198	21	



TASK 8

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

```
select pizza_types.category, count(pizza_types.name) as total_count
from pizza_types
group by pizza_types.category;
```

	category	total_count
▶	Chicken	6
	Classic	8
	Supreme	9
	Veggie	9



TASK 9

GROUP THE ORDERS BY DATE AND CALCULATE THE AVERAGE NUMBER OF PIZZAS ORDERED PER DAY.

```
select avg(quantity) from
  (select orders.order_date, round(sum(order_details.quantity),0) as quantity
   from orders
   join
   order_details
   on
   orders.order_id = order_details.order_id
   group by orders.order_date) as quantity_ordered;
```

Result Grid	
	avg(quantity)
▶	138.4749



TASK 10

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

```
select pizza_types.name, pizza_types.category, sum(order_details.quantity * pizzas.price) as revenue
from order_details
join
pizzas
on
order_details.pizza_id = pizzas.pizza_id
join
pizza_types
on
pizza_types.pizza_type_id = pizzas.pizza_type_id
group by pizza_types.name, pizza_types.category
order by revenue desc
limit 3;
```

Result Grid | Filter Rows:

	Name	Category	Revenue
▶	The Thai Chicken Pizza	Chicken	43434.25
	The Barbecue Chicken Pizza	Chicken	42768
	The California Chicken Pizza	Chicken	41409.5



TASK 11

CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPE TO TOTAL REVENUE.

```
(select pizza_types.category,  
concat(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  
pizzas.pizza_id = order_details.pizza_id) * 100,2),'%') as revenue  
from order_details  
join  
pizzas  
on  
order_details.pizza_id = pizzas.pizza_id  
join  
pizza_types  
on  
pizza_types.pizza_type_id = pizzas.pizza_type_id  
group by pizza_types.category);
```

Result Grid | Filter Rows:

	category	revenue
▶	Classic	26.91%
	Veggie	23.68%
	Supreme	25.46%
	Chicken	23.96%



TASK 12

ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME.

```
select sales.order_time,
       sales.revenue, SUM(sales.revenue) over (order by sales.order_time) as cum_revenue
  from (select o.order_time, SUM(od.quantity * p.price) AS revenue
        from order_details od
       join
      pizzas p
     on
    od.pizza_id = p.pizza_id
   join
  orders o
  on
  o.order_id = od.order_id
 group by o.order_time) as sales;
```

	order_time	revenue	cum_revenue
▶	09:52:21	83	83
	10:25:19	12.5	95.5
	10:34:34	53.25	148.75
	10:43:04	52.75	201.5
	10:50:46	50.25	251.75
	10:52:26	28.75	280.5
	10:54:03	47.9	328.4
	10:54:15	20.75	349.15
	10:54:55	37.5	386.65
	11:02:20	27	413.65



TASK 13

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

```
select pizza_types.name, pizza_types.category,
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
order_details.pizza_id = pizzas.pizza_id
group by pizza_types.name, pizza_types.category
order by revenue desc
limit 3;
```

Result Grid | Filter Rows: |

	name	category	revenue
▶	The Thai Chicken Pizza	Chicken	43434.25
	The Barbecue Chicken Pizza	Chicken	42768
	The California Chicken Pizza	Chicken	41409.5



MADE BY
Kush Gupta



THANK YOU

FOR ATTENTION

- 2025 PIZZA SALES ANALYSIS USING
MY SQL PRESENTATION