



A Delicious Dive into Pizza Sales



HOLA!

*I am Khushi Gupta and in this presentation
I am presenting the various queries and
analysis I did for a Pizza Sales dataset.*

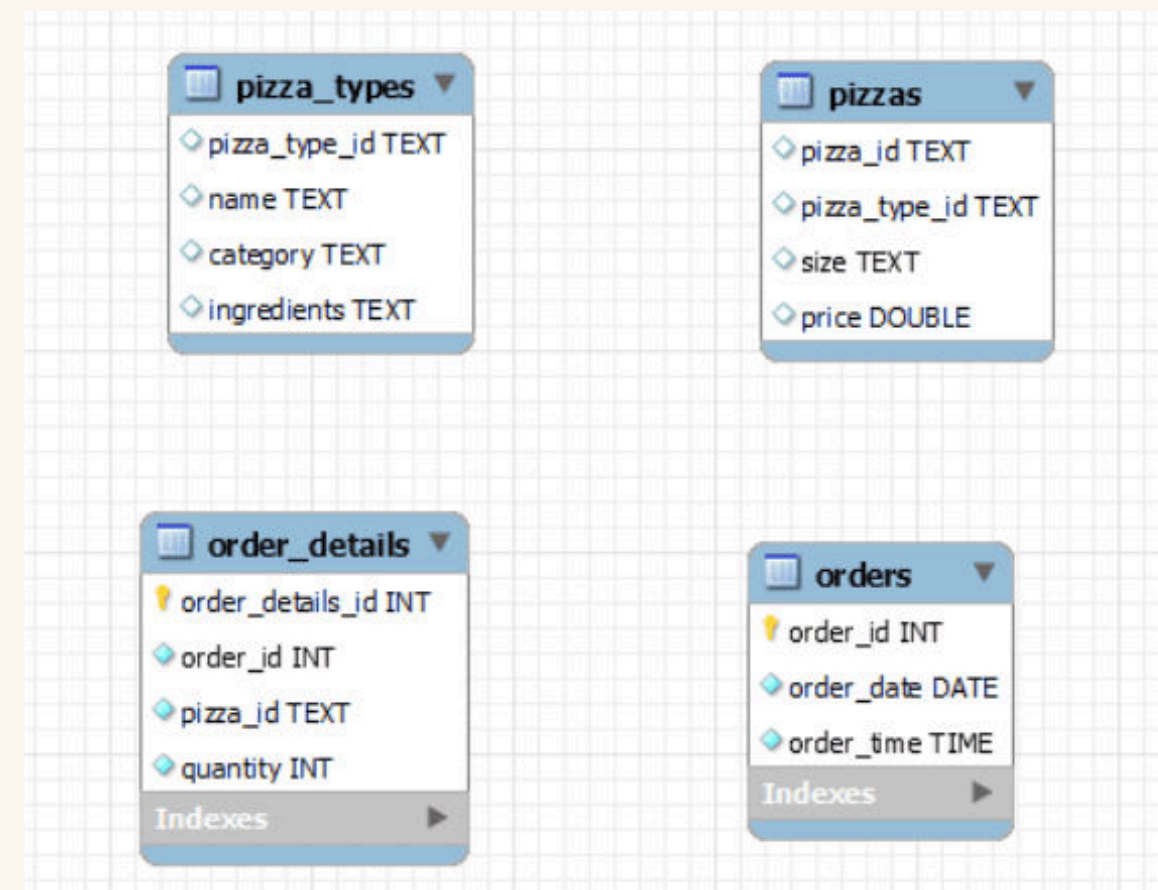
Let's begin...

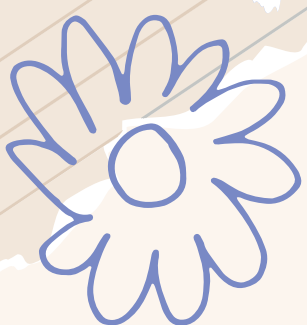




Know Your Crust

Let's first visualize the schema of the dataset.



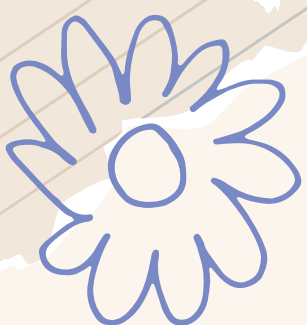


Retrieve the total number of orders placed.

```
select count(*) from orders;
```

	count(*)
▶	21350





Calculate the total revenue generated from pizza sales.

```
SELECT
    ROUND(SUM(order_details.quantity * pizzas.price),
          2) AS TotalRevenue
FROM
    order_details
    JOIN
    pizzas ON pizzas.pizza_id = order_details.pizza_id;
```

	TotalRevenue
▶	817860.05



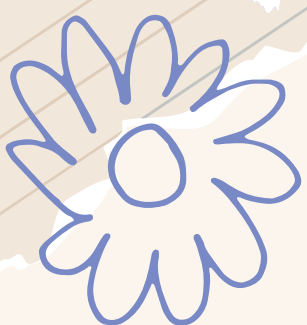


Identify the highest-priced pizza.

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

	name	price
▶	The Greek Pizza	35.95

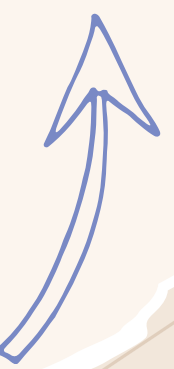


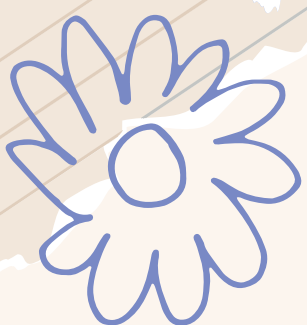


Identify the most common pizza size ordered

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

	size	cnt
▶	L	18526

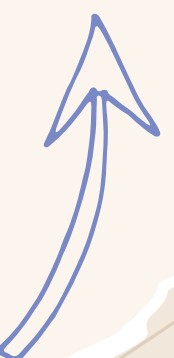


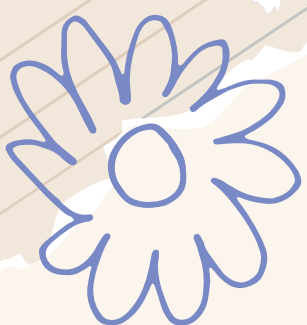


List the top 5 most ordered pizza types along with their quantities.

```
select pizza_types.name, sum(order_details.quantity) as quantityOrdered
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 quantityOrdered desc limit 5;
```

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



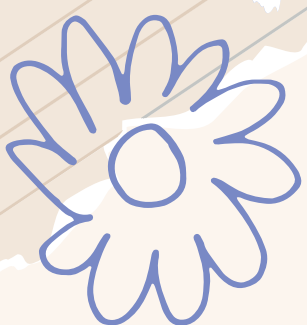


To find the total quantity of each pizza category ordered.

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

	category	QuantityOrdered
▶	Classic	14888
	Veggie	11649
	Supreme	11987
	Chicken	11050



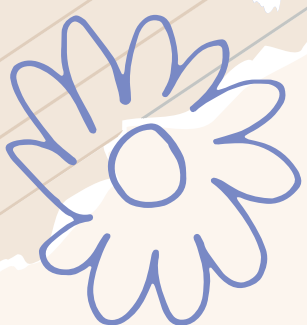


Determine the distribution of orders by hour of the day.

```
SELECT
    HOUR(order_time) AS hour, COUNT(order_id) AS order_count
FROM
    orders
GROUP BY HOUR(order_time)
ORDER BY order_count DESC;
```

	hour	order_count
▶	12	2520
	13	2455
	18	2399
	17	2336
	19	2009
	16	1920
	20	1642
	14	1472
	15	1468
	11	1231
	21	1198
	22	663
	23	28
	10	8
	9	1





To find the category-wise distribution of pizzas.

```
SELECT  
    category, COUNT(*)  
FROM  
    pizza_types  
GROUP BY category;
```

	category	COUNT(*)
▶	Chicken	6
	Classic	8
	Supreme	9
	Veggie	9





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) order_quantity_by_date;
```

	round(avg(quantity),0)
▶	138

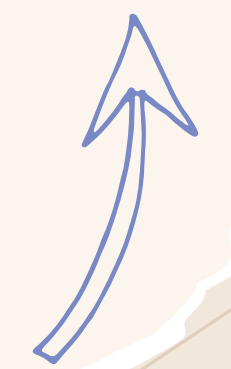


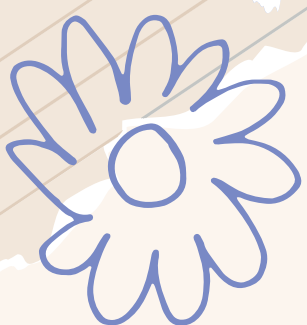


Determine the top 3 most ordered pizza types based on revenue.

```
select pizza_types.name, sum(pizzas.price*order_details.quantity) as Revenue
from pizzas
join pizza_types
on pizzas.pizza_type_id = pizza_types.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





Calculate the percentage contribution of each pizza type to total revenue.

```
set @totalRevenue = (SELECT
    ROUND(SUM(order_details.quantity * pizzas.price),2) AS TotalRevenue
FROM
    order_details
    JOIN
    pizzas ON pizzas.pizza_id = order_details.pizza_id);

select pizza_types.category, round((sum(order_details.quantity*pizzas.price)/@totalRevenue)*100,2) as revenuePercent
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 revenuePercent desc;
```

category	revenuePercent
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68





Determine the top 3 most ordered pizza types based on revenue for each pizza category.

```
select category, name, revenue
from
(
select category, name, revenue,
rank() over (partition by category order by revenue desc) as rn
from
(select pizza_types.category, 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 order_details.pizza_id = pizzas.pizza_id
group by pizza_types.category, pizza_types.name) t
) a
where rn<=3;
```

	category	name	revenue
▶	Chicken	The Thai Chicken Pizza	43434.25
	Chicken	The Barbecue Chicken Pizza	42768
	Chicken	The California Chicken Pizza	41409.5
	Classic	The Classic Deluxe Pizza	38180.5
	Classic	The Hawaiian Pizza	32273.25
	Classic	The Pepperoni Pizza	30161.75
	Supreme	The Spicy Italian Pizza	34831.25
	Supreme	The Italian Supreme Pizza	33476.75
	Supreme	The Sicilian Pizza	30940.5
	Veggie	The Four Cheese Pizza	32265.700000000065
	Veggie	The Mexicana Pizza	26780.75
	Veggie	The Five Cheese Pizza	26066.5





Analyze the cumulative revenue generated over time.

```
select order_date, revenue,  
sum(revenue) over(order by order_date) as cum_revenue  
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 orders.order_id = order_details.order_id  
group by orders.order_date) as sales;
```

order_date	revenue	cum_revenue
2015-01-01	2713.85000000000004	2713.85000000000004
2015-01-02	2731.89999999999996	5445.75
2015-01-03	2662.39999999999996	8108.15
2015-01-04	1755.45000000000003	9863.6
2015-01-05	2065.95	11929.55
2015-01-06	2428.95	14358.5
2015-01-07	2202.20000000000003	16560.7
2015-01-08	2838.34999999999995	19399.05
2015-01-09	2127.35000000000004	21526.4
2015-01-10	2463.95	23990.3500000000002
2015-01-11	1872.30000000000002	25862.65
2015-01-12	1919.05000000000002	27781.7
2015-01-13	2049.60000000000004	29831.3000000000003
2015-01-14	2527.39999999999996	32358.7000000000004
2015-01-15	1984.80000000000002	34343.5000000000001
2015-01-16	2594.15	36937.6500000000001
2015-01-17	2064.10000000000004	39001.7500000000001
2015-01-18	1976.85000000000001	40978.6000000000006

