

PIZZA SALES ANALYSIS (USING SQL)

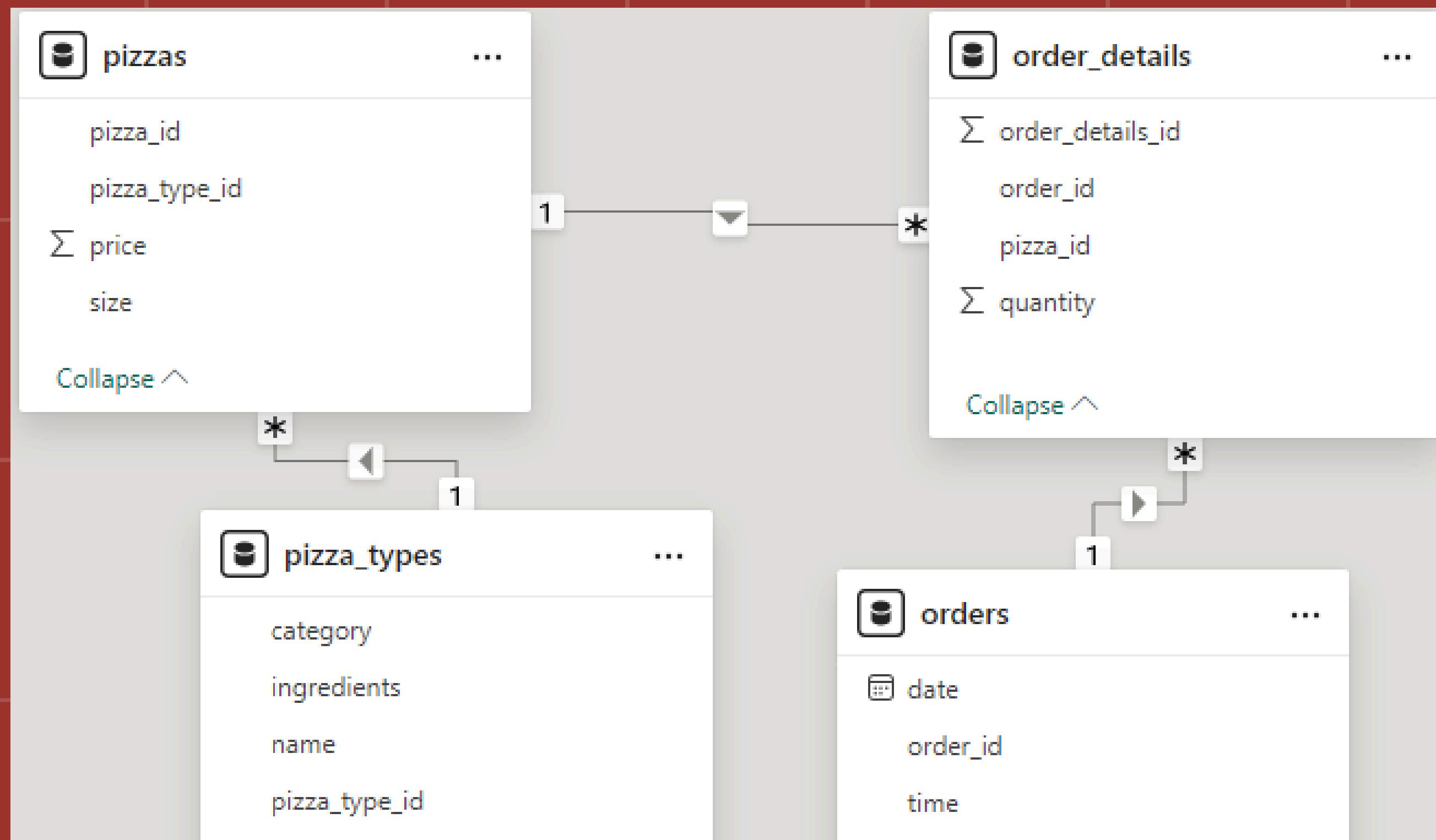
BY - MEHAK NARANG



OBJECTIVE

The objective of this analysis is to provide a comprehensive overview of sales performance of various pizza types and categories. The insights derived will guide strategic decision-making to optimize inventory management, enhance customer satisfaction, and drive revenue growth.

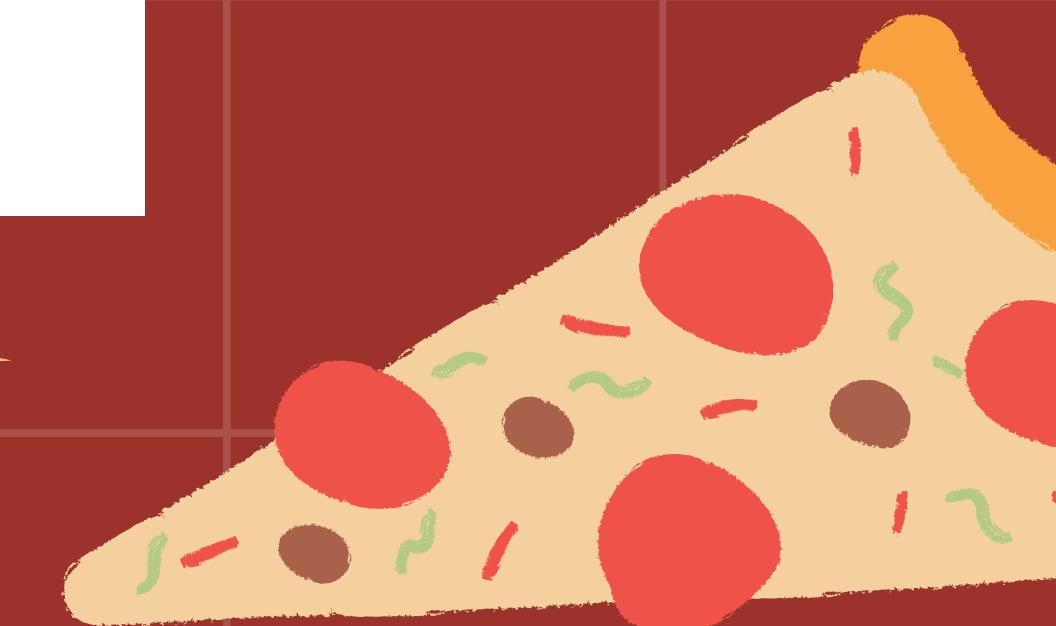
DATABASE SCHEMA



RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED

```
SELECT COUNT(order_id) as Total_Orders  
FROM orders;
```

Result Grid	
	Total_Orders
▶	21350



IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED

```
SELECT pizzas.size,COUNT(order_details.quantity) as order_count  
FROM order_details JOIN pizzas  
ON order_details.pizza_id=pizzas.pizza_id  
GROUP BY pizzas.size ORDER BY order_count DESC  
LIMIT 1;
```

Result Grid |

	size	order_count
▶	L	14516

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;
```

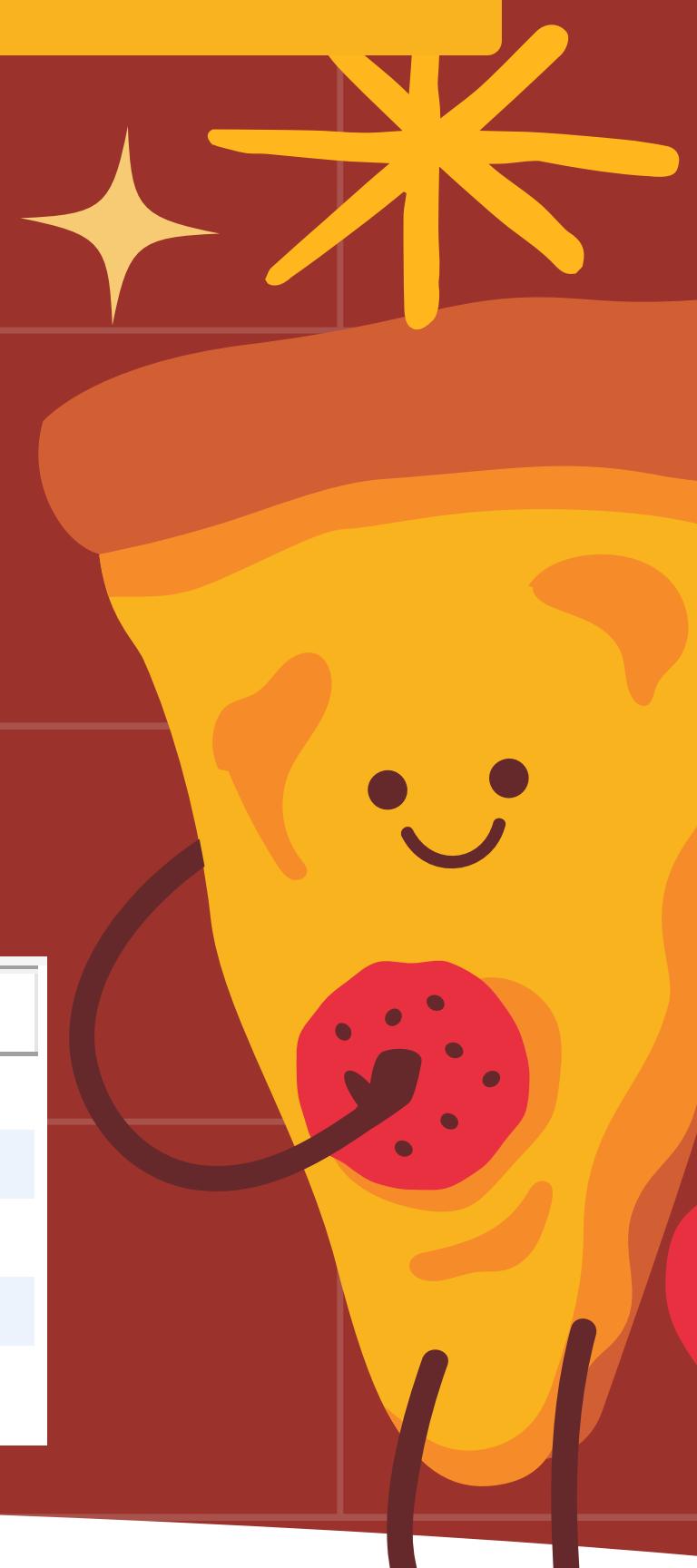
	name	price
▶	The Greek Pizza	35.95



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

```
SELECT pizza_types.name,COUNT(order_details.quantity) as order_count  
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 order_count DESC  
LIMIT 5;
```

	name	order_count
▶	The Barbecue Chicken Pizza	1887
	The Pepperoni Pizza	1873
	The Classic Deluxe Pizza	1870
	The Hawaiian Pizza	1820
	The California Chicken Pizza	1792



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 pizzas.pizza_type_id=pizza_types.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	11645
	Supreme	9333
	Veggie	9154
	Chicken	8627

CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES

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

	total_revenue
→	639651.05

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 pizzas.pizza_type_id=pizza_types.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	11645
	Supreme	9333
	Veggie	9154
	Chicken	8627

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY

```
SELECT TIME_FORMAT(time, "%H") as hour, COUNT(order_id) as orders  
FROM orders  
group by hour;
```

	hour	orders
▶	11	1231
	12	2520
	13	2455
	14	1472
	15	1468
	16	1920
	17	2336
	18	2399

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

```
SELECT category,COUNT(name) AS count  
FROM pizza_types  
GROUP BY category;
```

	category	count
▶	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.date, SUM(order_details.quantity) as quantity  
FROM orders JOIN order_details  
ON orders.order_id=order_details.order_id  
GROUP BY orders.date) AS orders_per_day;
```

	AVG(quantity)
▶	138.4250



DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE

```
SELECT pizza_types.name,SUM(pizzas.price *order_details.quantity) 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  
ORDER BY revenue DESC LIMIT 3;
```

	name	revenue
▶	The Barbecue Chicken Pizza	38885.25
	The Thai Chicken Pizza	33446.25
	The California Chicken Pizza	32205.5

CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPE TO TOTAL REVENUE

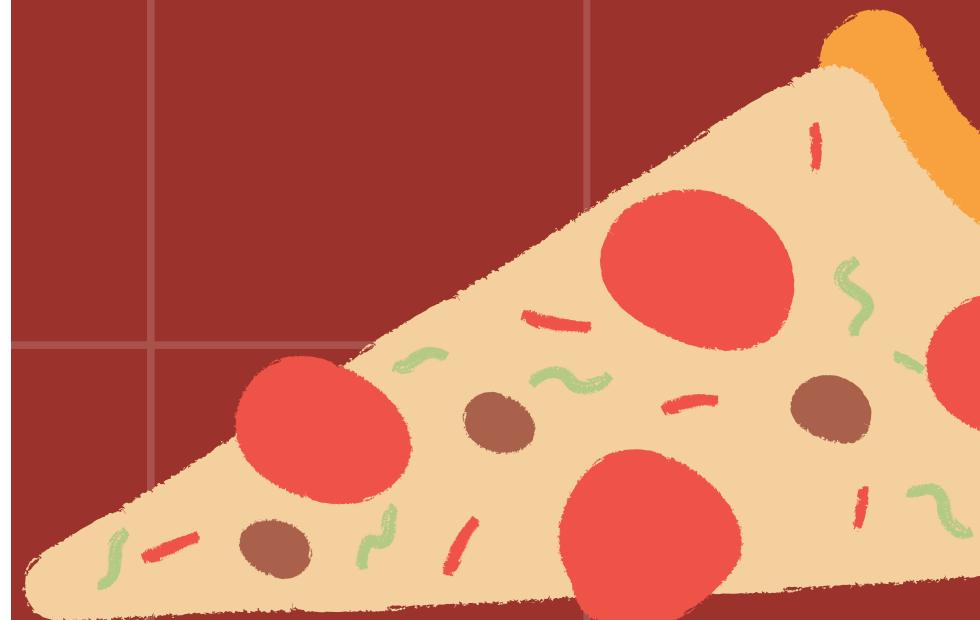
```
SELECT pizza_types.category,  
ROUND((SUM(pizzas.price *order_details.quantity) / (SELECT  
ROUND(SUM(order_details.quantity * pizzas.price),2) AS total_revenue  
FROM order_details JOIN pizzas  
ON order_details.pizza_id = pizzas.pizza_id))*100,2) AS percentage  
FROM pizza_types JOIN pizzas  
ON 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 percentage DESC;
```

	category	percentage
▶	Classic	26.93
	Supreme	25.37
	Chicken	23.91
	Veggie	23.79

ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME

```
SELECT date,  
       SUM(revenue) OVER( ORDER BY date) AS cumm_revenue  
  FROM  
(SELECT orders.date , SUM(pizzas.price*order_details.quantity) AS revenue  
    FROM orders JOIN order_details  
  ON orders.order_id=order_details.order_id  
JOIN pizzas  
ON pizzas.pizza_id=order_details.pizza_id  
GROUP BY orders.date) as sales;
```

date	cumm_revenue
2015-01-01	2713.850000000004
2015-01-02	5445.75
2015-01-03	8108.15



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 r
FROM
(SELECT pizza_types.category , pizza_types.name , SUM(pizzas.price *order_details.quantity) 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) AS p) AS z
WHERE r<=3;
```

THANK YOU

