

A Structured Query Language (SQL)-Based Data Exploration

PIZZA SALES - SQL ANALYSIS REPORT



BY
MAYANK

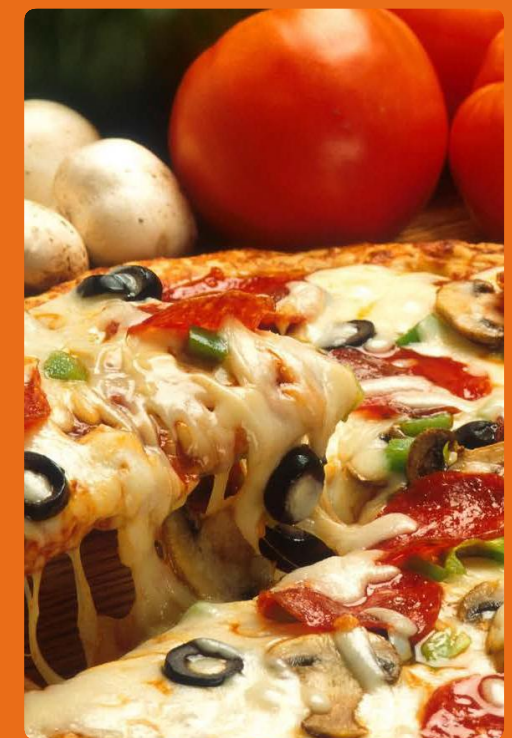
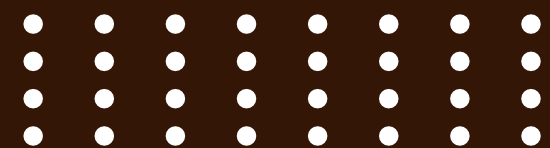
Aspiring Data Analyst

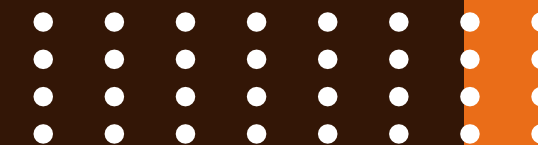


INTRODUCTION

PROJECT OVERVIEW

This project focuses on analyzing pizza sales using SQL to uncover patterns, trends, and actionable insights. It involves writing and executing queries on multiple related tables to answer real-world business questions.





DATASET OVERVIEW

PIZZAS: CONTAINS PIZZA SIZE AND PRICE DETAILS

PIZZA_TYPES : DESCRIBES THE TYPE AND CATEGORY OF PIZZAS

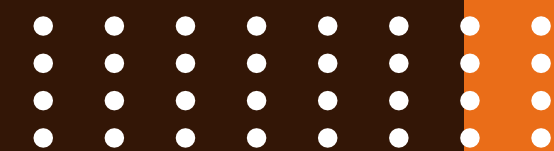


ORDER_DETAILS : RECORDS QUANTITY AND PIZZA ORDERED
PER ORDER

ORDERS : PROVIDES ORDER DATE AND TIME

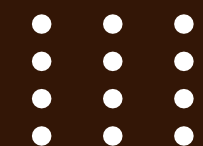


RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED



```
1 SELECT
2     COUNT(order_id) AS TOTAL_ORDERS
3 FROM
4     orders;
```

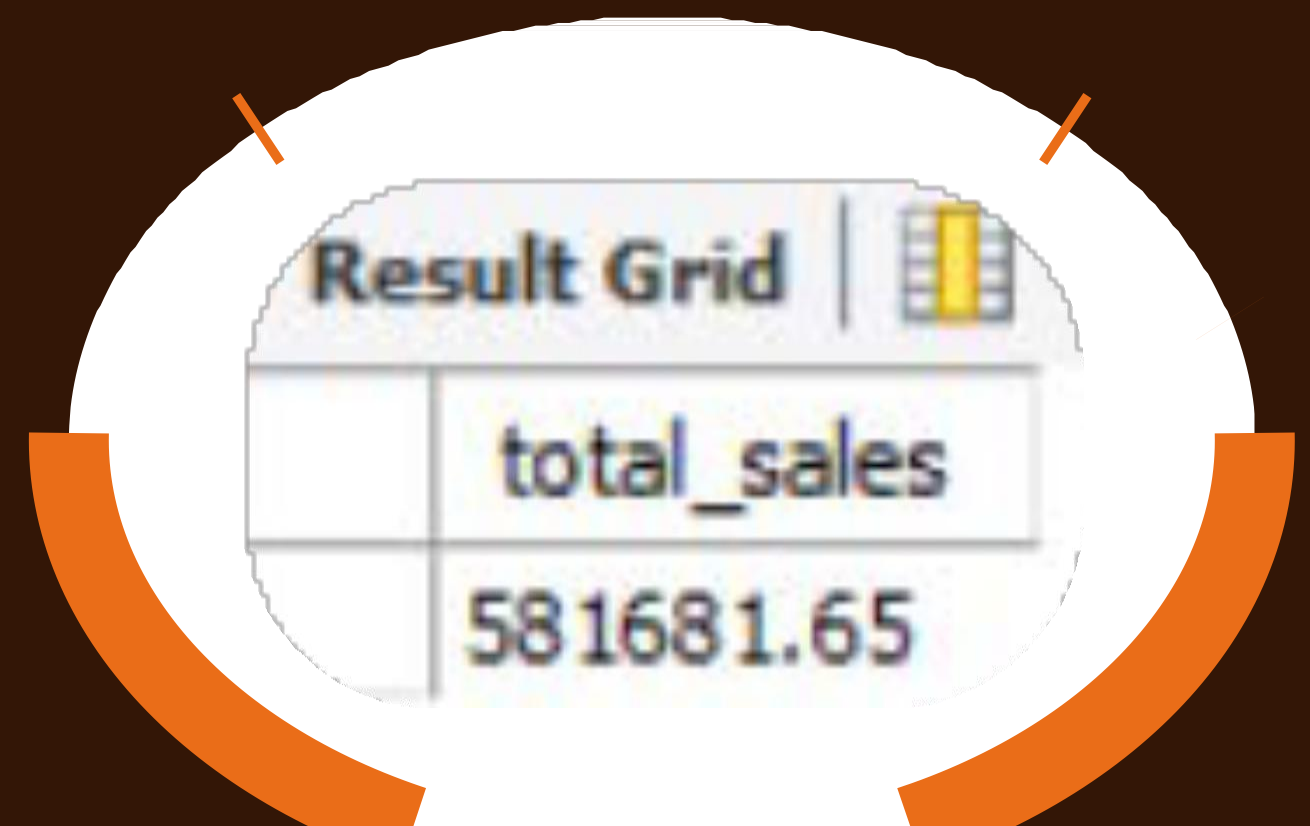
Result Grid	
	TOTAL_ORDERS
▶	4314





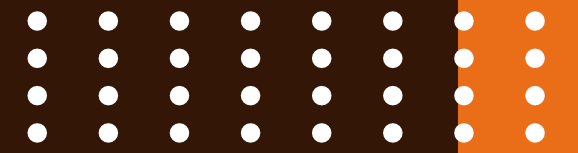
CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

```
SELECT
2  ROUND(SUM(order_details.quantity * pizzas.price),
3         2) AS total_sales
4  FROM
5  order_details
6  JOIN
7  pizzas ON pizzas.pizza_id = order_details.pizza_id;
```



	total_sales
	581681.65

IDENTIFY THE HIGHEST PRICES PIZZA.



```
1 select pizza_types.name , pizzas.price as expensive_pizza
2 from pizza_types join pizzas on pizza_types.pizza_type_id
3 = pizzas.pizza_type_id
4 order by expensive_pizza desc limit 1;
```

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



IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.

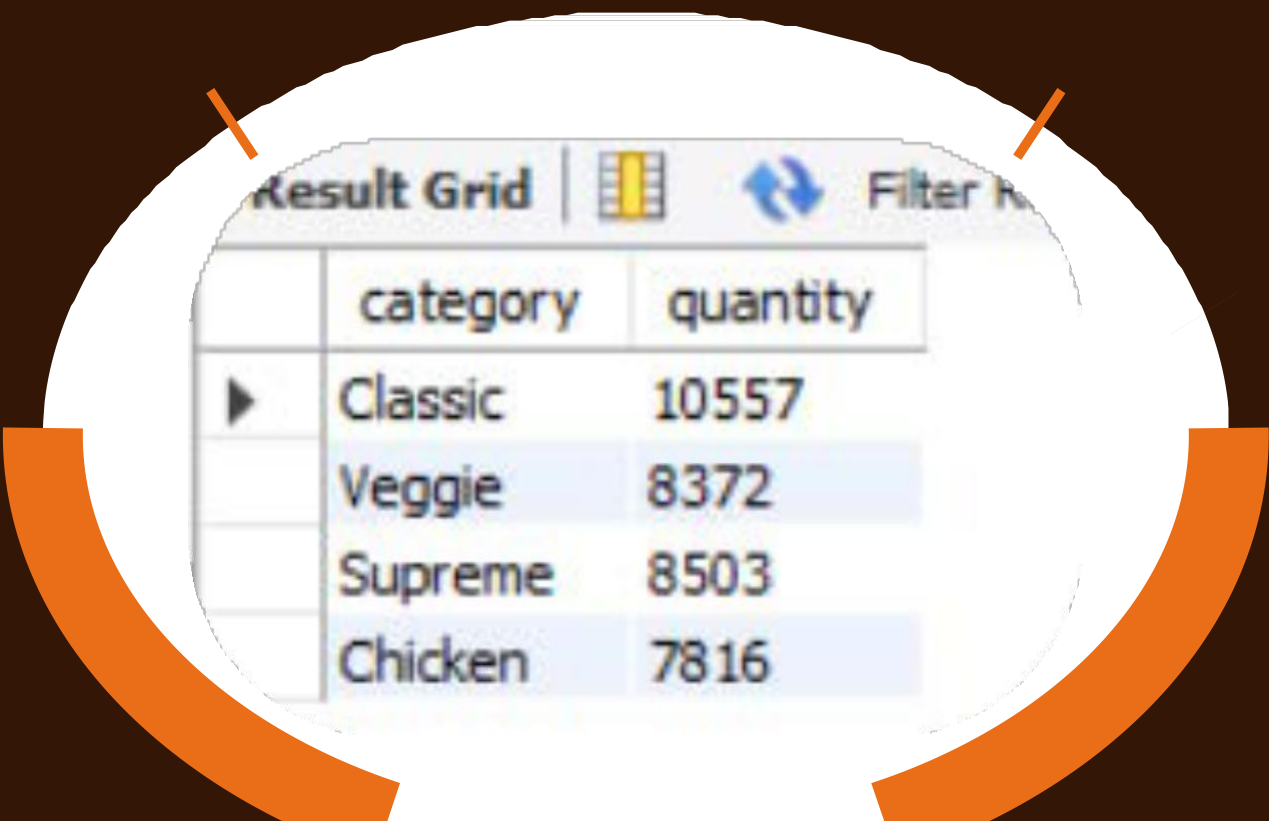
```
1 • SELECT
2     pizzas.size,
3     COUNT(order_details.order_details_id) AS total_order_detail_id
4 FROM
5     pizzas
6     JOIN
7     order_details ON pizzas.pizza_id = order_details.pizza_id
8 GROUP BY pizzas.size;
```

size	total_order_detail_id
M	10957
L	13180
S	10028
XL	400



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

```
1 • SELECT
2     pizza_types.category,
3     SUM(order_details.quantity) AS quantity
4 FROM
5     pizza_types
6     JOIN
7     pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
8     JOIN
9     order_details ON pizzas.pizza_id = order_details.pizza_id
10 GROUP BY pizza_types.category;
```



	category	quantity
▶	Classic	10557
	Veggie	8372
	Supreme	8503
	Chicken	7816



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

```
1 • SELECT
2     pizza_types.category AS category,
3     SUM(order_details.quantity) AS quantity
4 FROM
5     pizza_types
6     JOIN
7     pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
8     JOIN
9     order_details ON order_details.pizza_id = pizzas.pizza_id
10 GROUP BY category
11 ORDER BY quantity DESC;
```

	category	quantity
▶	Classic	10557
	Supreme	8503
	Veggie	8372
	Chicken	7816



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

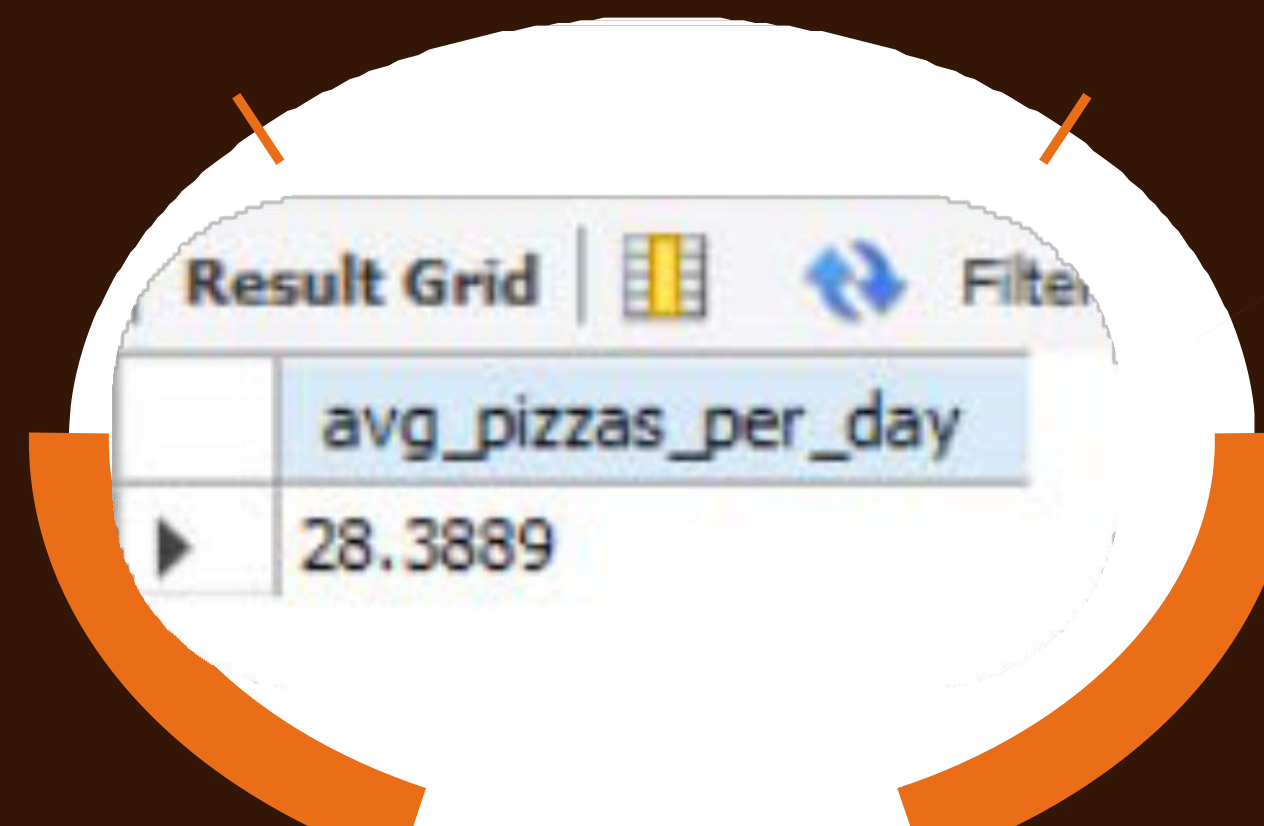
```
1 SELECT
2     category, COUNT(name) AS name
3 FROM
4     pizza_types
5 GROUP BY category;
```

	category	name
▶	Chicken	6
	Classic	8
	Supreme	9
	Veggie	9



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

```
SELECT AVG(quantity_per_day) AS avg_pizzas_per_day
FROM (
  SELECT
    orders.date,
    SUM(order_details.quantity) AS quantity_per_day
  FROM orders
  JOIN order_details
    ON orders.order_id = order_details.order_id
  GROUP BY orders.date
) AS daily_totals;
```



Result Grid | Filter

	avg_pizzas_per_day
▶	28.3889

DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE

```
1 • SELECT
2     pizza_types.name,
3     ROUND(SUM(order_details.quantity * pizzas.price),
4           0) AS revenue
5 FROM
6     pizza_types
7     JOIN
8     pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
9     JOIN
10    order_details ON pizzas.pizza_id = order_details.pizza_id
11 GROUP BY pizza_types.name
12 ORDER BY revenue DESC
13 LIMIT 3;
```

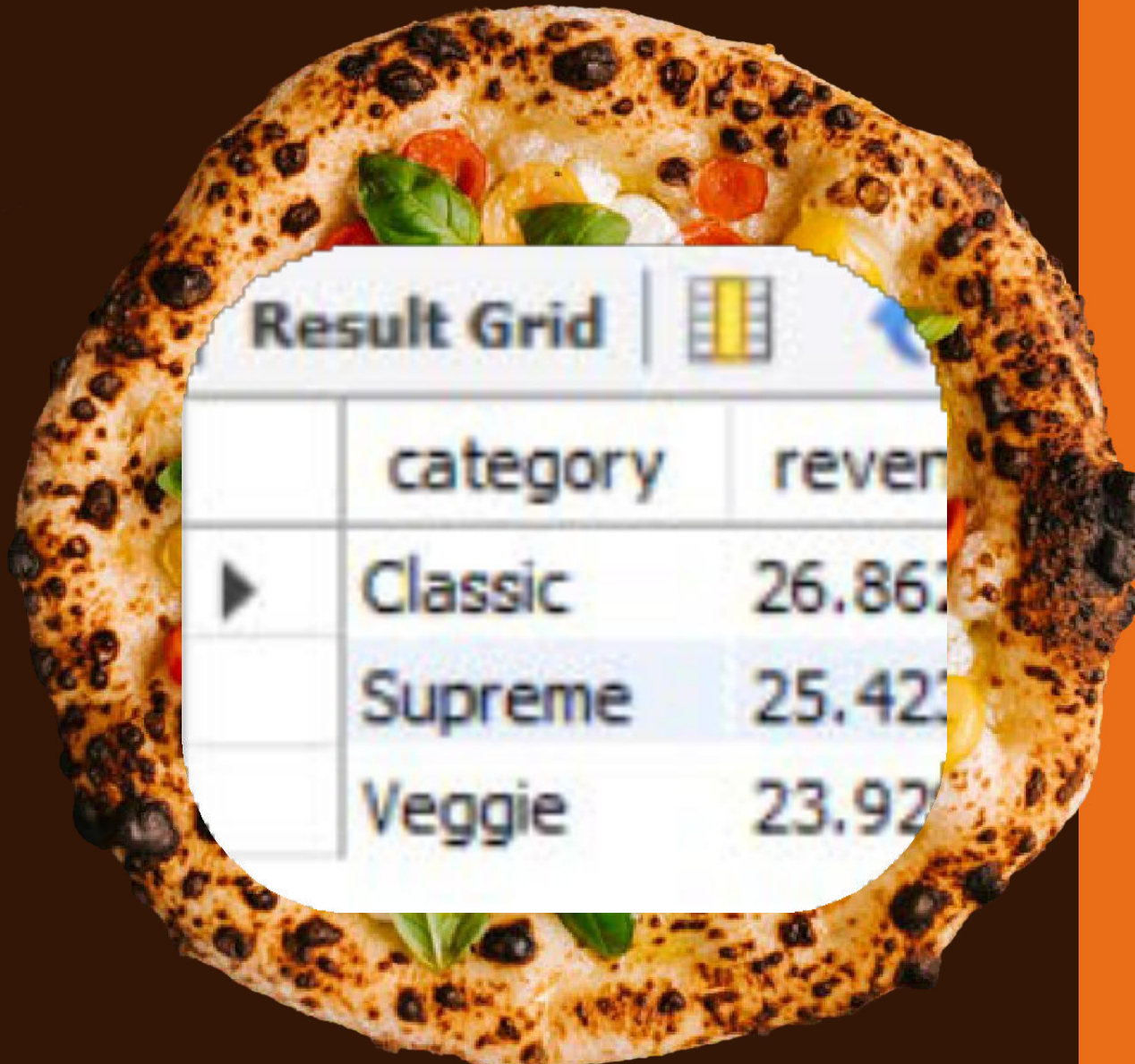


name	revenue
The Barbecue Chicken Pizza	30842
The Thai Chicken Pizza	30241
The California Chicken Pizza	29390

CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPE TO TOTAL REVENUE



```
1 • SELECT pizza_types.category ,  
2      |(SUM(order_details.quantity * pizzas.price)/(select  
3      round(SUM(order_details.quantity*pizzas.price),2) as total_sales  
4      from order_details join pizzas  
5      on pizzas.pizza_id = order_details.pizza_id))*100 as revenue  
6      FROM pizza_types join  
7      pizzas on pizza_types.pizza_type_id = pizzas.pizza_type_id jo  
8      order_details  
9      on pizzas.pizza_id = order_details.pizza_id  
0      group by pizza_types.category order by revenue desc limit 3 ;  
1
```



Result Grid

	category	revenue
▶	Classic	26.86%
	Supreme	25.42%
	Veggie	23.92%



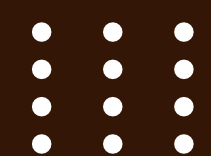
ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME



```
1 • select date , sum(revenue) over (order by date) as cum_revenue
2 from
3 (select orders.date,
4     ROUND(SUM(order_details.quantity * pizzas.price)) as revenue
5 from order_details join
6     pizzas on order_details.pizza_id = pizzas.pizza_id join
7     orders on orders.order_id = order_details.order_details_id
8 group by orders.date) as sales;
9 |
```



2015-02-12	9518
2015-02-13	9899
2015-02-14	10199
2015-02-15	10289
2015-02-16	10364
2015-02-17	10560
2015-02-18	10814
2015-02-19	10936
2015-02-20	11162
2015-02-21	11357
2015-02-22	11451
2015-02-23	11547
2015-02-24	11637
2015-02-25	11773
2015-02-26	12007



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

```
1 SELECT
2     category,
3     name,
4     revenue,
5     rn
6 FROM (
7     SELECT
8         pizza_types.category,
9         pizza_types.name,
10        SUM(order_details.quantity * pizzas.price) AS revenue,
11        RANK() OVER (PARTITION BY pizza_types.category ORDER BY SUM(order_details.quantity *
12 FROM pizza_types
13 JOIN pizzas
14     ON pizza_types.pizza_type_id = pizzas.pizza_type_id
15 JOIN order_details
16     ON order_details.pizza_id = pizzas.pizza_id
17     GROUP BY pizza_types.category, pizza_types.name
18 ) AS ranked_pizzas
19 WHERE rn <= 3;
```

category	name	revenue	rn
Chicken	The Barbecue Chicken Pizza	30841.75	1
Chicken	The Thai Chicken Pizza	30241.25	2
Chicken	The California Chicken Pizza	29398.25	3
Classic	The Classic Deluxe Pizza	26653.5	1
Classic	The Hawaiian Pizza	22562.25	2
Classic	The Pepperoni Pizza	21575	3
Supreme	The Spicy Italian Pizza	25034.25	1
Supreme	The Italian Supreme Pizza	24057.75	2
Supreme	The Sicilian Pizza	21550.5	3
Veggie	The Four Cheese Pizza	22805.150000000365	1
Veggie	The Five Cheese Pizza	19110.5	2
Veggie	The Mexicana Pizza	18787.75	3

THANK YOU

