



PIZZA SALES PROJECT





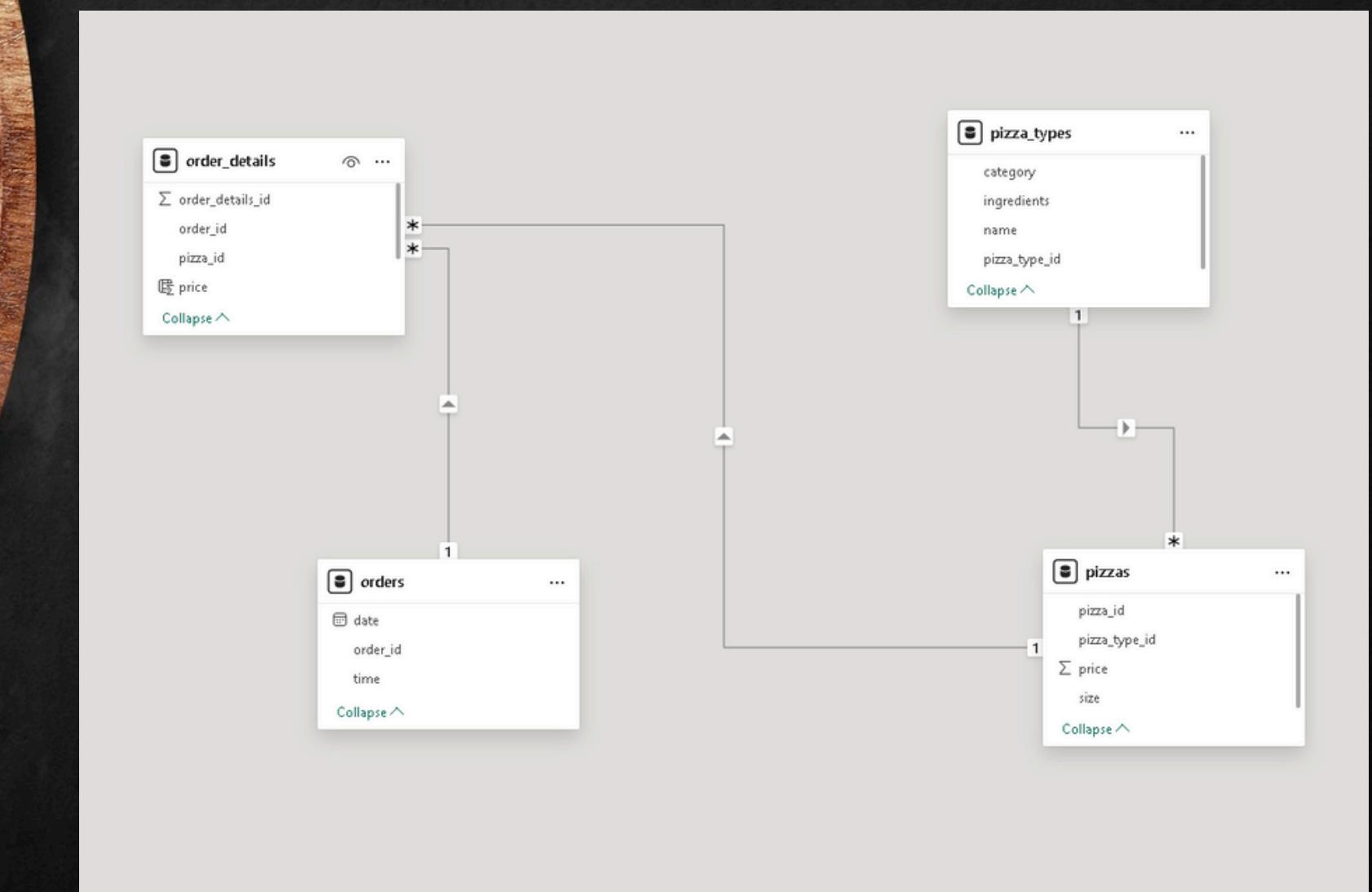
LARANA PIZZA

HELLO TO ALL THE DATA ANALYST ENTHUSIASTS OUT THERE:

In this project, we used MySQL to analyze and answer business questions related to pizza sales for a fictional pizza restaurant. Our goal was to leverage data to provide insights into customer preferences, sales trends, and operational efficiencies. By creating a relational database, we organized data on orders, order_details, pizzas and pizza types.



SCHEMA VIEW OF THE DATABASE





RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED

```
3  
4 • SELECT  
5     COUNT(order_details_id)  
6 FROM  
7     order_details;  
8
```

Result Grid	
	COUNT(order_id)
▶	21350



CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

SELECT

```
ROUND(SUM(o.quantity * p.price), 2) AS total_revenue_generated
```

FROM

```
order_details o
```

JOIN

```
pizzas p ON o.pizza_id = p.pizza_id;
```

Result Grid	
	total_revenue_generated
▶	817860.05



IDENTIFY THE HIGHEST-PRICED PIZZA.

```
SELECT
    pt.name, p.price
FROM
    pizza_types pt
        JOIN
    pizzas p ON pt.pizza_type_id = p.pizza_type_id
ORDER BY p.price DESC
LIMIT 1;
```

Result Grid | Filter Rows

	name	price
▶	The Greek Pizza	35.95



IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.

```
select p.size, count(od.quantity) as quantity_ordered from pizzas p  
JOIN order_details od on  
p.pizza_id=od.pizza_id  
group by 1  
order by 2 desc ;
```

Result Grid		Filter Rows
	size	quantity_ordered
▶	L	18526
	M	15385
	S	14137
	XL	544
	XXL	28



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

```
SELECT
    pt.name, COUNT(od.quantity) quantities_ordered
FROM
    pizza_types pt
        JOIN
    pizzas p ON pt.pizza_type_id = p.pizza_type_id
        JOIN
    order_details od ON p.pizza_id = od.pizza_id
GROUP BY 1
ORDER BY 2 DESC
LIMIT 5;
```

Result Grid		
	name	quantities_ordered
▶	The Classic Deluxe Pizza	2416
	The Barbecue Chicken Pizza	2372
	The Hawaiian Pizza	2370
	The Pepperoni Pizza	2369
	The Thai Chicken Pizza	2315



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

```
SELECT
    pt.category AS category,
    COUNT(od.quantity) AS total_quantity
FROM
    pizza_types pt
        JOIN
    pizzas p ON pt.pizza_type_id = p.pizza_type_id
        JOIN
    order_details od ON p.pizza_id = od.pizza_id
GROUP BY 1
ORDER BY 2 DESC;
```

	category	total_quantity
▶	Classic	14579
	Supreme	11777
	Veggie	11449
	Chicken	10815



DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.

```
select hour(orders.time) as hours, count(order_id) as orders from orders  
group by 1  
order by 2 desc;
```

	hours	orders
▶	12	2520
	13	2455
	18	2399
	17	2336
	19	2009
	16	1920
	20	1642
	14	1472
	15	1468
	11	1231



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

SELECT

```
pt.category, COUNT(pt.pizza_type_id) AS count_of_pizzas
```

FROM

```
pizza_types pt
```

GROUP BY 1**ORDER BY 2 DESC;**

Result Grid | Filter Rows:

	category	count_of_pizzas
▶	Supreme	9
	Veggie	9
	Classic	8
	Chicken	6



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

```
SELECT  
    ROUND(AVG(q.total_quantity), 2) AS average_per_day  
FROM  
    (SELECT  
        o.order_date, SUM(od.quantity) AS total_quantity  
    FROM  
        orders o  
    JOIN order_details od ON o.order_id = od.order_id  
    GROUP BY 1  
    ORDER BY 2 DESC) AS q;
```

Result Grid	
	average_per_day
▶	138.47



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

```
SELECT
    pt.name, SUM(p.price * od.quantity) AS revenue
FROM
    pizza_types pt
        JOIN
    pizzas p ON pt.pizza_type_id = p.pizza_type_id
        JOIN
    order_details od ON p.pizza_id = od.pizza_id
GROUP BY 1
ORDER BY 2 DESC
LIMIT 3
```

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



PIZZA SALES PROJECT

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

```
SELECT
    pt.name,
    ROUND(SUM(p.price * od.quantity) / (SELECT
                                            ROUND(SUM(od.quantity * p.price), 2) AS total_revenue
                                         FROM
                                            pizzas p
                                         JOIN
                                            order_details od ON p.pizza_id = od.pizza_id) * 100,
    2) AS percentage_contribution
FROM
    pizza_types pt
    JOIN
        pizzas p ON pt.pizza_type_id = p.pizza_type_id
    JOIN
        order_details od ON p.pizza_id = od.pizza_id
GROUP BY 1
ORDER BY 2 DESC;
```

Result Grid		
	name	percentage_contribution
▶	The Thai Chicken Pizza	5.31
	The Barbecue Chicken Pizza	5.23
	The California Chicken Pizza	5.06
	The Classic Deluxe Pizza	4.67
	The Spicy Italian Pizza	4.26
	The Southwest Chicken Pizza	4.24
	The Italian Supreme Pizza	4.09
	The Hawaiian Pizza	3.95
	The Four Cheese Pizza	3.95
	The Sicilian Pizza	3.78



ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME.

```
select ov.order_date,  
sum(ov.revenue) over (order by ov.order_date) as cum_revenue from  
(select date(o.order_date) as order_date,round(sum(od.quantity*p.price),2) as revenue from orders o  
JOIN order_details od on o.order_id=od.order_id  
JOIN pizzas p on od.pizza_id=p.pizza_id  
group by 1  
order by 1) as ov;
```

	order_date	cum_revenue
▶	2015-01-01	2713.85
	2015-01-02	5445.75
	2015-01-03	8108.15
	2015-01-04	9863.6
	2015-01-05	11929.55
	2015-01-06	14358.5
	2015-01-07	16560.7
	2015-01-08	19399.05
	2015-01-09	21526.399999999998
	2015-01-10	23990.35



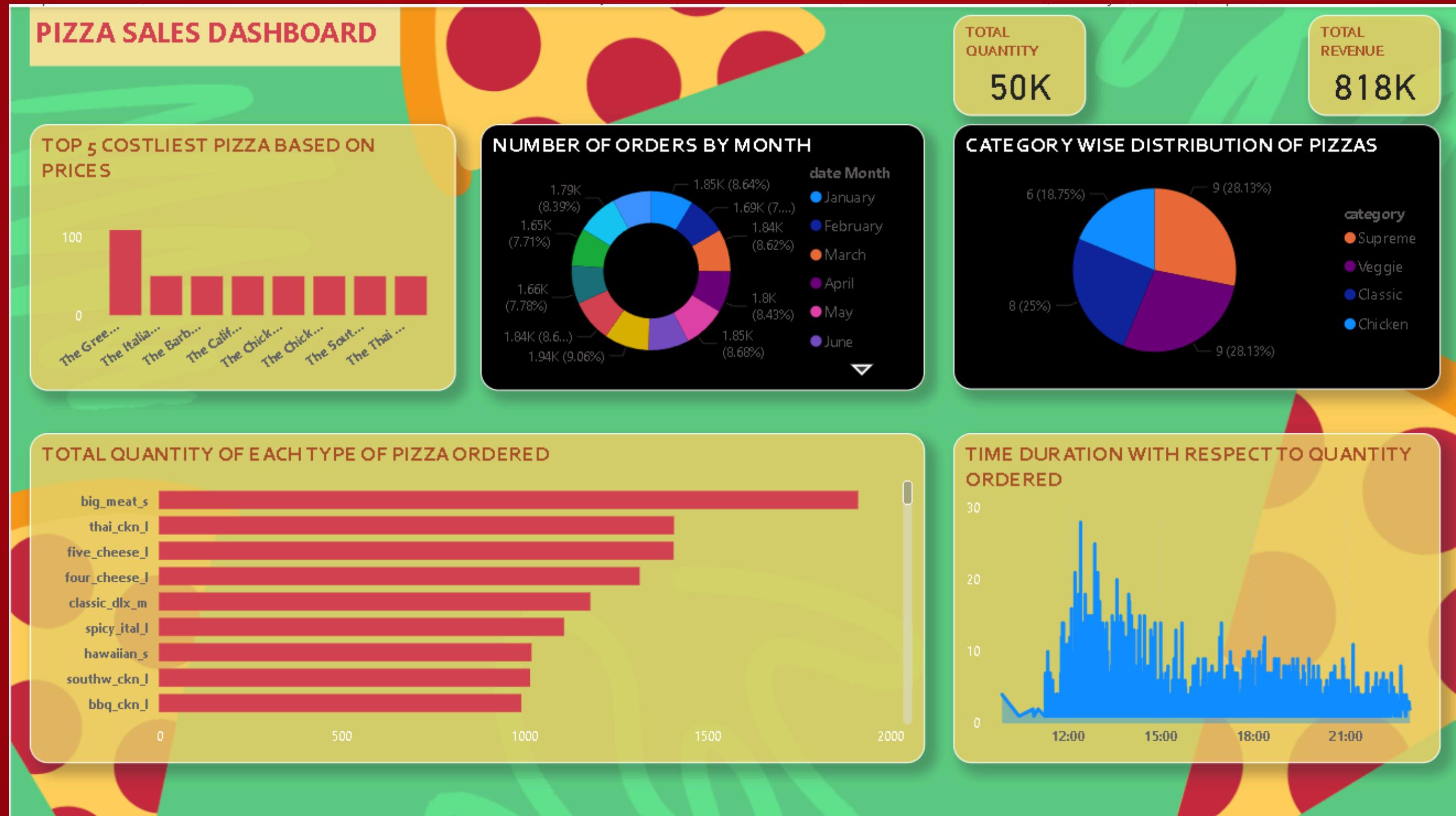
DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE FOR EACH PIZZA CATEGORY.

```
select category, name, revenue
from
(select category, name, revenue,
rank() over(partition by category order by revenue desc) as rn
from
(select pt.category, pt.name, sum(p.price*od.quantity) as revenue from pizza_types pt
JOIN pizzas p on pt.pizza_type_id=p.pizza_type_id
JOIN order_details od on p.pizza_id=od.pizza_id
group by 1,2) as p) as q
where rn<=3;
```

Result Grid			Export:
	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	32265.70000000065

PIZZA SALES PROJECT

PIZZA SALES DASHBOARD





LARANA PIZZA

CONTACT DETAILS

📞 8638888390

🔗 <https://github.com/AnkuranKh/Pizza-Sales-SQL.git>

🔗 <https://www.linkedin.com/in/ankuran-khanikar-38a352247/>



PIZZA SALES PROJECT

THANK YOU! 🍕

WE APPRECIATE YOUR ATTENTION AND INTEREST IN OUR PIZZA
SALES DATA ANALYSIS!

OUR FINDINGS HIGHLIGHT OPPORTUNITIES TO ENHANCE SALES,
OPTIMIZE INVENTORY. WE LOOK FORWARD TO SEEING HOW THESE
INSIGHTS CAN DRIVE DELICIOUS RESULTS!
QUESTIONS? LET'S DISCUSS!

