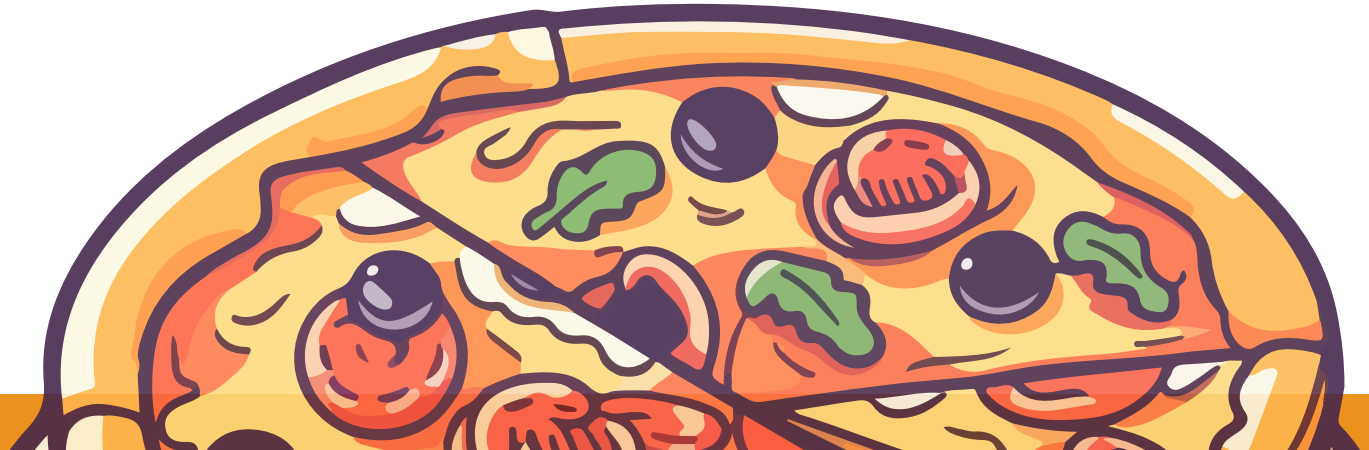
PEPPERONI PIZZA SALES ANALYSER ON SQL

A simple yet elegant pizza sales project

A comprehensive SQL-based project designed to analyze and visualize the sales performance of a pizza business. This project aims to leverage SQL to manage, query, and extract insights from a database containing detailed information about pizza sales. The primary objectives are to understand sales trends, customer preferences, and overall business performance to inform decision-making and strategic planning.



Query: List the top 5 ordered pizza types along with their quantities

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

SELECT
    pizza_types.name, SUM(order_details.quantity) AS 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
ORDER BY quantity DESC
LIMIT 5;
```

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

Query: Join the necessary tables to find the total quantity of each pizza category ordered

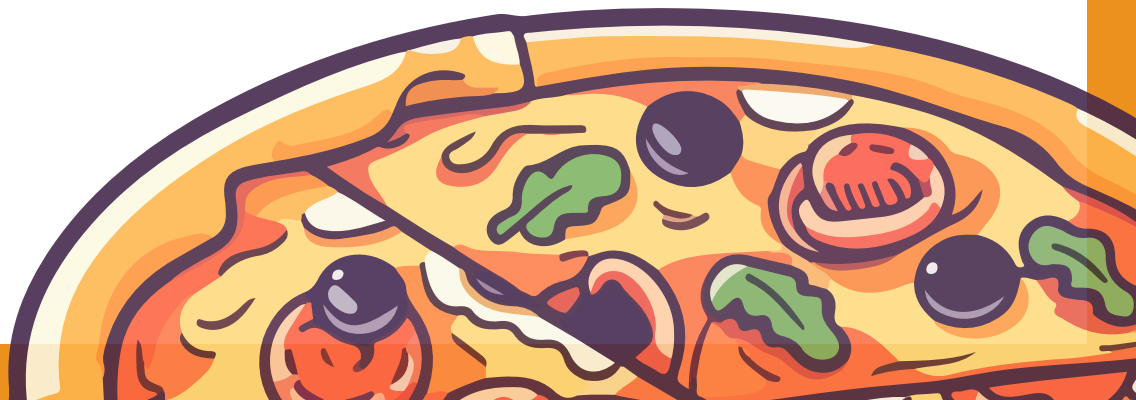
```sql
-- Join the necessary tables to find the total quantity of each pizza category ordered.
SELECT
    pizza_types.category, COUNT(order_details.quantity) AS total
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.category;
```

	category	total
▶	Classic	14579
	Veggie	11449
	Supreme	11777
	Chicken	10815

Query: Determine the distribution of orders by hour of the day

```
1      -- Determine the distribution of orders by hour of the day.
2 •    SELECT hour(time) AS hour, count(order_id) AS count
3       FROM orders
4       GROUP BY hour;
```

	hour	count
▶	11	1231
	12	2520
	13	2455
	14	1472
	15	1468
	16	1920
	17	2336
	18	2399
	19	2009



Query: Group the orders by date and calculate the average number of pizzas ordered per day

```sql
-- Group the orders by date and calculate the average number of pizzas ordered
SELECT round(avg(quantity),0) AS average_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 total;
```

	average_quantity
▶	138

Query: Determine the top 3 most ordered pizza types based on revenue

```sql
-- Determine the top 3 most ordered pizza types based on revenue.
SELECT 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.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

Query: Calculate the percentage contribution of each pizza type to total revenue

```sql
-- Calculate the percentage contribution of each pizza catagory 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 order_details.pizza_id = pizzas.pizza_id ) * 100,2),'%') 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 ORDER BY revenue DESC;
```

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

Query: Analyze the cumulative revenue generated over time

```sql
-- Analyze the cumulative revenue generated over time.
SELECT date ,round(sum(sales) OVER(ORDER BY date),2) AS cum_revenue
FROM
(SELECT orders.date , round(sum(order_details.quantity * pizzas.price ),2) AS sales
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.date) as cum_revenue;
```

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	21526.4

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

```sql
-- Determine the top 3 most ordered pizza types based on revenue for each pizza category
SELECT category , name , sales
FROM
(SELECT category , name , sales,
Rank() OVER( partition by category order by sales desc) AS rn
FROM
(SELECT pizza_types.category , pizza_types.name , sum(order_details.quantity * pizzas.price) AS sales
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 a) AS d
WHERE rn<=3;
```

category	name	sales
Chicken	The Thai Chicken Pizza	43434.2
Chicken	The Barbecue Chicken Pizza	42768
Chicken	The California Chicken Pizza	41409.5
Classic	The Classic Deluxe Pizza	38180.5
Classic	The Hawaiian Pizza	32273.2
Classic	The Pepperoni Pizza	30161.8
Supreme	The Spicy Italian Pizza	34831.2
Supreme	The Italian Supreme Pizza	33476.8
Supreme	The Sicilian Pizza	30940.5
Veggie	The Four Cheese Pizza	32265.7
Veggie	The Mexicana Pizza	26780.8
Veggie	The Five Cheese Pizza	26066.5