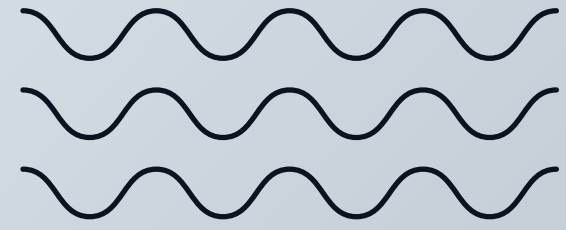


DATA DOUGHNUT: EXPLORING PIZZA SALES TRENDS WITH SQL

BY- N I T E S H K U M A R



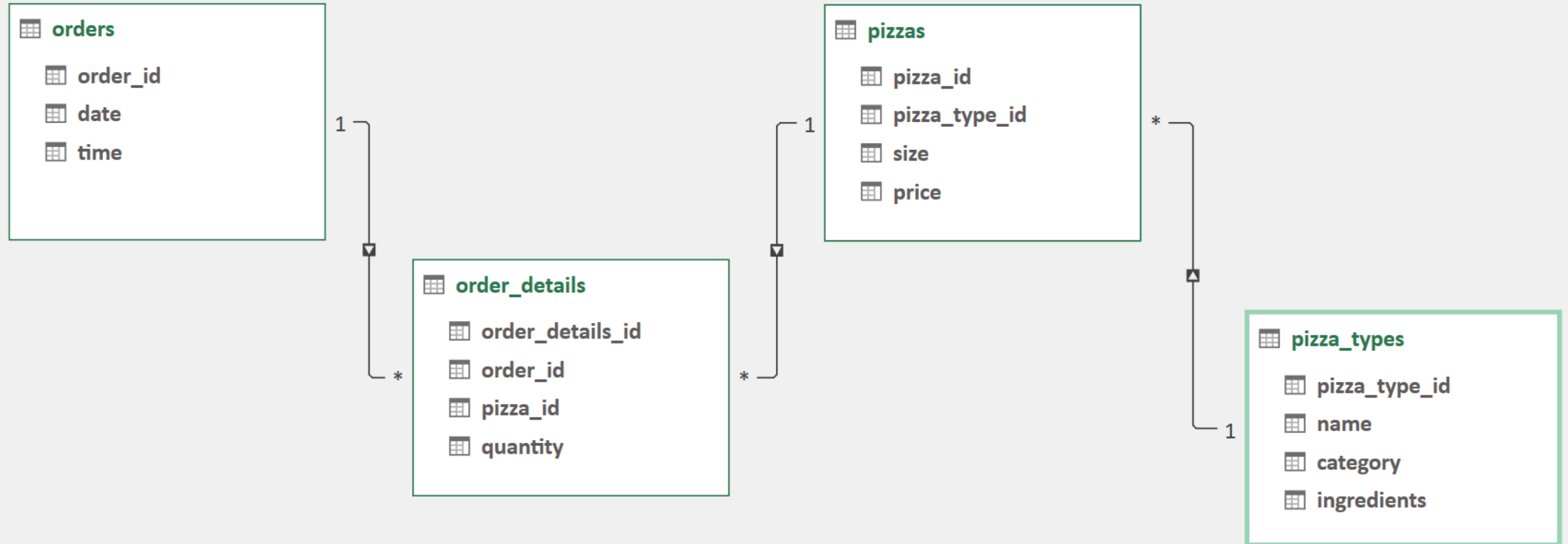
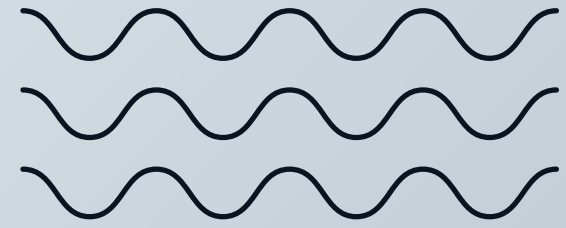


Project Title: Pizza Sales Analysis using SQL

Dive into our Pizza Sales project, where SQL queries unveil intricate insights into customer preferences, popular toppings, and peak ordering periods. Through meticulous analysis, optimize operations, streamline inventory management, and elevate customer satisfaction. Join us in savoring the flavors of success through data-driven decisions in pizza sales.



DATA MODEL



RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.

```
SELECT  
    COUNT(order_id) AS order_placed  
  
FROM  
    orders
```

Result Grid	
	order_placed
▶	21350

CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

```
SELECT
```

```
    ROUND(SUM(order_details.quantity * pizzas.price),  
          2) AS total_revenue
```

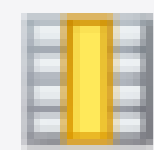
```
FROM
```

```
    order_details
```

```
    JOIN
```

```
    pizzas ON order_details.pizza_id = pizzas.pizza_id
```

Result Grid



	total_revenue
▶	817860.05

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.

```
SELECT
    pizzas.size,
    COUNT(order_details.order_details_id) AS most_common_pizza
FROM
    pizzas
    JOIN
        order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY pizzas.size
ORDER BY 2 DESC
```

	size	most_common_pizza
▶	L	18526

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

```
SELECT
    pizza_types.name, SUM(order_details.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 1
ORDER BY 2 DESC
LIMIT 5;
```

name	SUM(order_details.quantity)
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	2422
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371

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 pizza_types.pizza_type_id = pizzas.pizza_type_id
    JOIN
    order_details ON order_Details.pizza_id = pizzas.pizza_id
GROUP BY 1
ORDER BY 2 DESC;
```

category	Quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY

```
SELECT  
    HOUR(time) AS hour, COUNT(order_id) AS order_count  
FROM  
    orders  
GROUP BY 1
```

hour	order_count
11	1231
12	2520
13	2455
14	1472
15	1468

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

```
SELECT
    ROUND(AVG(quantity), 0) as avg_daily_ordered_pizza
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 1) AS a
```

avg_daily_ordered_pizza

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
    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 1
ORDER BY 2 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

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

category	revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME

```
select date,  
sum(revenue) over(order by date) as cum_revenue  
from  
(select orders.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.date) as sales
```

date	cum_revenue
2015-01-01	2713.8500000000000004
2015-01-02	5445.75
2015-01-03	8108.15
2015-01-04	9863.6
2015-01-05	11929.55

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

```
select 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) as a) as b
where rn <= 3;
```

name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5
The Classic Deluxe Pizza	38180.5
The Hawaiian Pizza	32273.25



**ENDING OUR PIZZA SALES PROJECT, SQL
QUERIES HAVE BEEN THE KEY INGREDIENT
IN UNRAVELING CUSTOMER PREFERENCES.
WITH THESE INSIGHTS, WE'RE READY TO
SPICE UP OPERATIONS, INCREASE PROFITS,
AND KEEP OUR PIZZA FANS SATISFIED.
THANKS FOR JOINING US ON THIS
DELICIOUS JOURNEY**

