SQL-Project

Jayasri's Pizza

A project on pizza house sales analysis

www.linkedin.com/in/jayasri-kolli-6781222a9

NEXT >

PROJECT SUMMARY

01 OBJECTIVE

To analyze pizza sales data to provide actionable insights on sales performance , customer preference and revenue generation

02 ABOUT THE DATASET

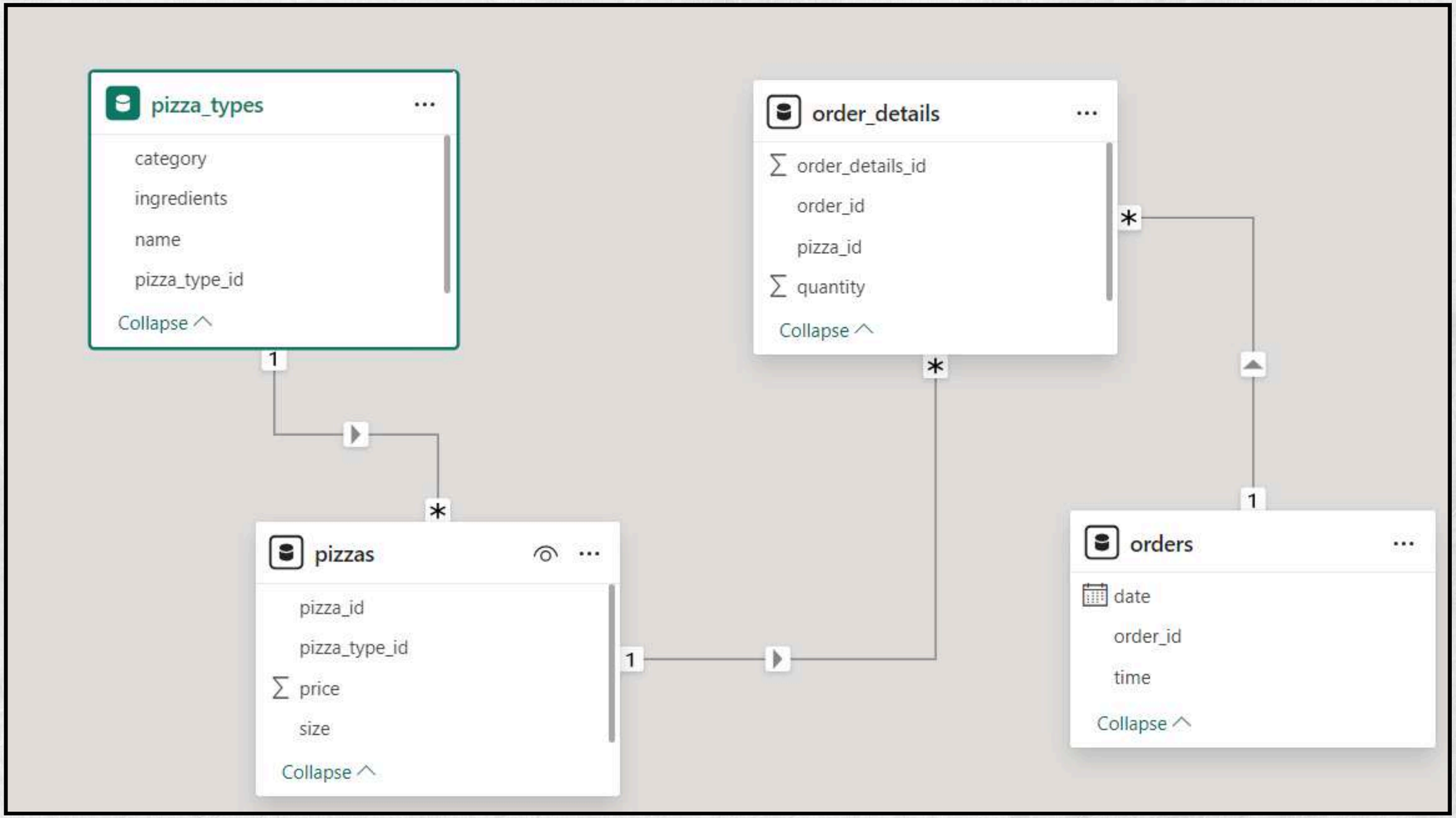
Analyzing the sales records from a pizza restaurant spanning one year.
Dataset has 48620 sales transactions to analyze.

03 TOOLS USED

Mysql for data querying and manipulation
PowerBi Desktop for plots

NEXT >

DATA SCHEMA



NEXT >

BASIC QUESTIONS

NEXT >

Retrieve the total number of orders placed

```sql
SELECT
    COUNT(order_id) AS Total_number_of_orders
FROM
    orders;
```

Total_number_of_orders
21350

NEXT >

Calculate the total revenue generated from pizza sales

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

Result Grid

total_revenue
817860.05

NEXT >

Identify the highest-priced pizza

```sql
SELECT
    pizza_types.name, pizzas.price AS highest_priced_pizza
FROM
    pizzas
        JOIN
    pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
ORDER BY pizzas.price DESC
LIMIT 1;
```

	name	highest_priced_pizza
	The Greek Pizza	35.95

NEXT >

Identify the most common pizza size ordered

```sql
SELECT
    pizzas.size, SUM(orders_details.quantity) as Times_orderded
FROM
    pizzas
        JOIN
    orders_details ON pizzas.pizza_id = orders_details.pizza_id
GROUP BY pizzas.size order by Times_orderded desc limit 1;
```

size	Times_orderded
L	18956

NEXT >

List the top 5 most ordered pizza types along with their quantities

```sql
SELECT
    pizza_types.name, SUM(orders_details.quantity) as total_quantity
FROM
    pizza_types
        JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
        JOIN
    orders_details ON orders_details.pizza_id = pizzas.pizza_id
GROUP BY pizza_types.name
ORDER BY total_quantity DESC
LIMIT 5;
```

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

NEXT >

ADVANCED QUESTIONS

NEXT >

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

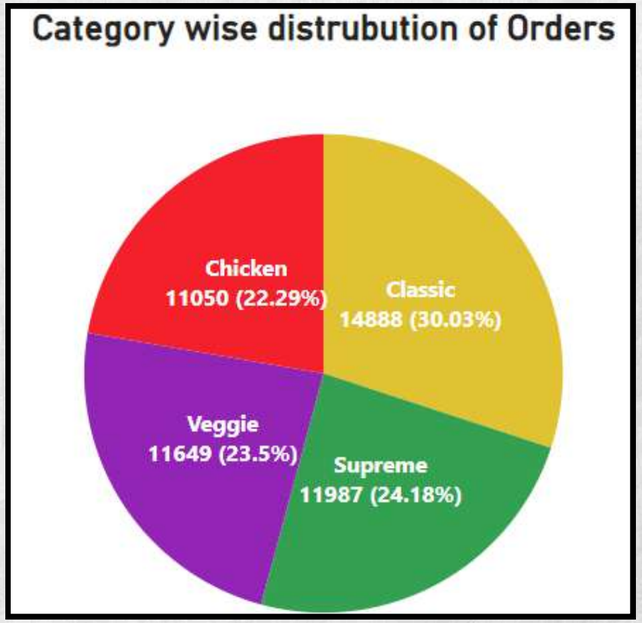
```sql
SELECT
    pizza_types.category,
    SUM(orders_details.quantity) AS total_quantity
FROM
    pizza_types
        JOIN
    pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
        JOIN
    orders_details ON pizzas.pizza_id = orders_details.pizza_id
        JOIN
    orders ON orders_details.order_id = orders.order_id
GROUP BY pizza_types.category;
```

category	total_quantity
Classic	14888
Veggie	11649
Supreme	11987
Chicken	11050

NEXT >

Determine the distribution of orders by hour of the day

```sql
SELECT
    pizza_types.category,
    SUM(orders_details.quantity) AS total_quantity
FROM
    pizza_types
        JOIN
    pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
        JOIN
    orders_details ON pizzas.pizza_id = orders_details.pizza_id
        JOIN
    orders ON orders_details.order_id = orders.order_id
GROUP BY pizza_types.category;
```



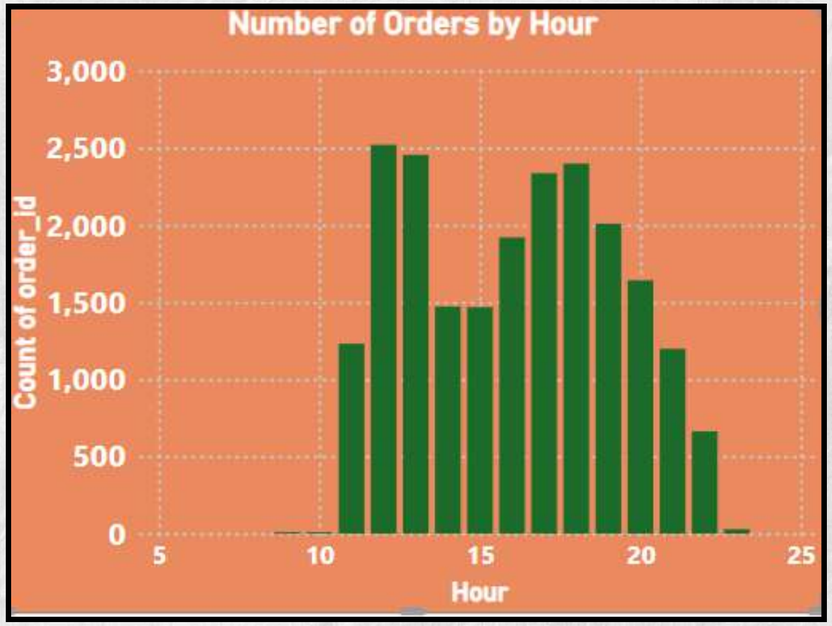
Result Grid | Filter R

	category	total_quantity
▶	Classic	14888
	Veggie	11649
	Supreme	11987
	Chicken	11050

NEXT >

Determine the distribution of orders by hour of the day

```sql
SELECT
    HOUR(orders.order_time) AS hour,
    COUNT(order_id) AS order_count
FROM
    orders
GROUP BY hour
order by hour;
```



esult Grid

hour	order_count
9	1
10	8
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642
21	1198
22	663
23	28

NEXT >

Join relevant tables to find the category-wise distribution of pizzas

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

category	COUNT(name)
Chicken	6
Classic	8
Supreme	9
Veggie	9

category	name
Chicken	The Barbecue Chicken Pizza
Chicken	The California Chicken Pizza
Chicken	The Chicken Alfredo Pizza
Chicken	The Chicken Pesto Pizza
Chicken	The Southwest Chicken Pizz
Chicken	The Thai Chicken Pizza
Classic	The Big Meat Pizza
Classic	The Classic Deluxe Pizza
Classic	The Hawaiian Pizza
Classic	The Italian Capocollo Pizza
Classic	The Napolitana Pizza
Classic	The Pepperoni, Mushroom,
Classic	The Pepperoni Pizza
Classic	The Greek Pizza

category	name
Supreme	The Brie Carre Pizza
Supreme	The Calabrese Pizza
Supreme	The Italian Supreme Pizza
Supreme	The Pepper Salami Pizza
Supreme	The Prosciutto and Arugula Pizza
Supreme	The Sicilian Pizza
Supreme	The Soppressata Pizza
Supreme	The Spicy Italian Pizza
Supreme	The Spinach Supreme Pizza
Veggie	The Five Cheese Pizza
Veggie	The Four Cheese Pizza
Veggie	The Green Garden Pizza
Veggie	The Italian Vegetables Pizza
Veggie	The Mediterranean Pizza
Veggie	The Mexicana Pizza
Veggie	The Spinach Pesto Pizza
Veggie	The Spinach and Feta Pizza
Veggie	The Vegetables + Vegetables Pi.

NEXT >

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

```sql
SELECT
    ROUND(AVG(quantity)) AS avg_number_of_pizzas_per_day
FROM
    (SELECT
        orders.order_date, SUM(orders_details.quantity) AS quantity
    FROM
        orders
    JOIN orders_details ON orders.order_id = orders_details.order_id
    GROUP BY orders.order_date) AS order_quantity;
```

	avg_pizzas_per_day
▶	138

NEXT >

Determine the top 3 most ordered pizza types based on revenue

```sql
SELECT
    pizza_types.name,
    SUM(orders_details.quantity * pizzas.price) AS revenue
FROM
    pizza_types
        JOIN
    pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
        JOIN
    orders_details ON pizzas.pizza_id = orders_details.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

NEXT >

Calculate the percentage contribution of each pizza type to total revenue

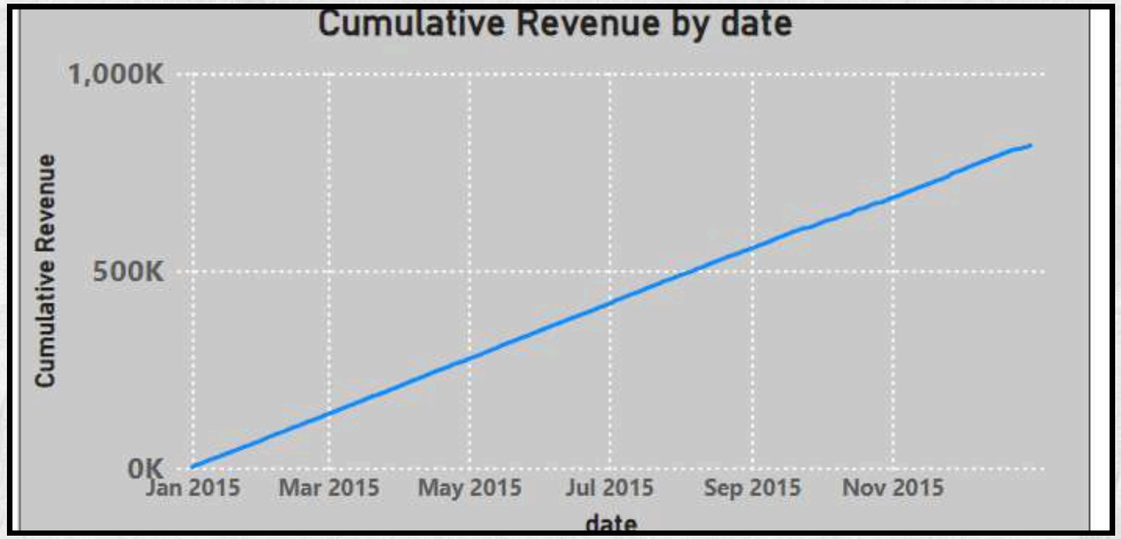
```sql
SELECT
    pizza_types.category,
    ROUND((SUM(orders_details.quantity * pizzas.price) * 100 / (SELECT
                    ROUND(SUM(orders_details.quantity * pizzas.price),
                                2) AS total_revenue_generated
                FROM
                    orders_details
                        JOIN
                    pizzas ON orders_details.pizza_id = pizzas.pizza_id)),
            2) AS revenue
FROM
    pizza_types
        JOIN
    pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
        JOIN
    orders_details ON pizzas.pizza_id = orders_details.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

NEXT >

Analyze the cumulative revenue generated over time

```sql
select order_date, sum(revenue) over(order by order_date)as cumulative_revenue
from (select orders.order_date , sum(pizzas.price * orders_details.quantity) as revenue
from
orders_details join orders
on orders_details.order_id =orders.order_id
join
pizzas
on pizzas.pizza_id =orders_details.pizza_id group by orders.order_date ) as sales;
```



order_date	cumulative_revenu
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
2015-01-10	23990.35
2015-01-11	25862.65
2015-01-12	27781.7
2015-01-13	29831.3
2015-01-14	32358.7
2015-01-15	34343.5
2015-01-16	36937.65
2015-01-17	39001.75
2015-01-18	40978.6
2015-01-19	43365.75
2015-01-20	45763.65
2015-01-21	47804.2
2015-01-22	50300.9

NEXT >

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

```
select name, revenue,rn from
(select category,name, revenue, rank() over (partition by category
order by revenue desc)as rn from
(select pizza_types.category,pizza_types.name,
round(sum(orders_details.quantity*pizzas.price),2)as revenue
from pizza_types join pizzas
on pizza_types.pizza_type_id =pizzas.pizza_type_id
join orders_details
on pizzas.pizza_id=orders_details.pizza_id
group by pizza_types.category,pizza_types.name) as a) as b where rn<4;
```

name	revenue	rn
The Thai Chicken Pizza	43434.25	1
The Barbecue Chicken Pizza	42768	2
The California Chicken Pizza	41409.5	3
The Classic Deluxe Pizza	38180.5	1
The Hawaiian Pizza	32273.25	2
The Pepperoni Pizza	30161.75	3
The Spicy Italian Pizza	34831.25	1
The Italian Supreme Pizza	33476.75	2
The Sicilian Pizza	30940.5	3
The Four Cheese Pizza	32265.7	1
The Mexicana Pizza	26780.75	2
The Five Cheese Pizza	26066.5	3

NEXT >

THANKS !

Do you have any questions ?

jayasri15121985@gmail.com

www.linkedin.com/in/jayasri-kolli-6781222a9

https://www.hackerrank.com/profile/jayasri15121985

NEXT >