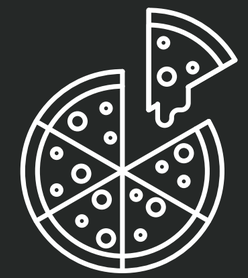
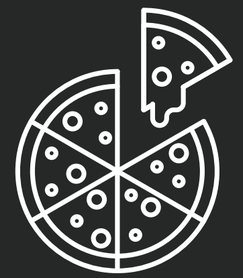
SQL Project on Pizza Sales

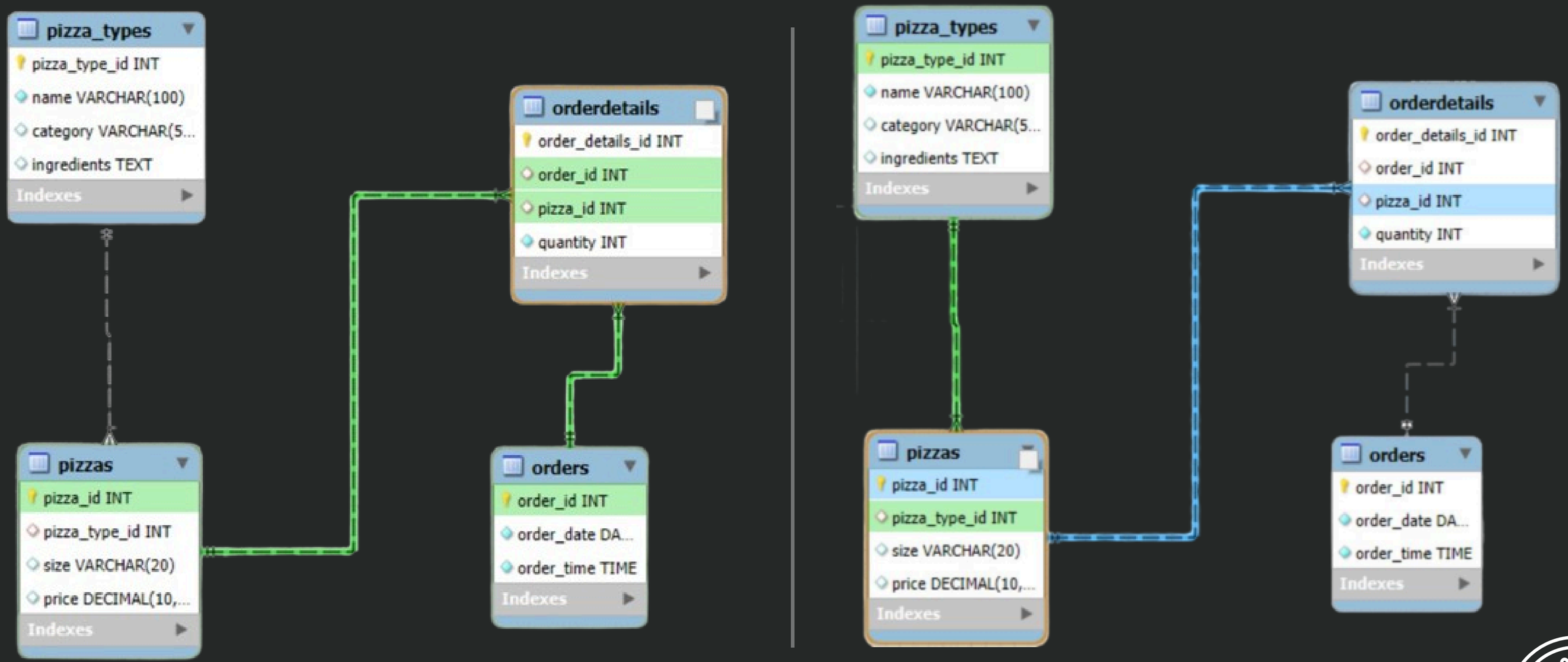


Project – PizzaMania

Hello, I am Shubham Raikwar. In this project, I applied SQL queries to analyze pizza sales data, focusing on trends, customer preferences, and inventory management.



Schema - PizzaMania

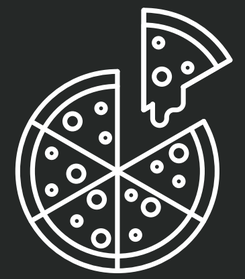


Schema Overview

The PizzaMania database consists of four interconnected tables:

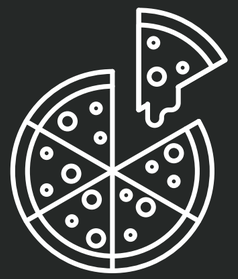
1. **pizza_types**: Stores information about different pizza types, identified by pizza_type_id.
2. **pizzas**: Contains details about each pizza, identified by pizza_id, including its size and price. It references pizza_types via the pizza_type_id foreign key.
3. **orders**: Tracks customer orders, uniquely identified by order_id.
4. **orderdetails**: Links individual pizzas to specific orders, referencing both orders through order_id and pizzas through pizza_id.

This structure allows for efficient querying of pizza sales, types, and order details.



Questions:

- Retrieve the total number of orders placed.
- Calculate the total revenue generated from pizza sales.
- Identify the highest-priced pizza.
- Identify the most common pizza size ordered.
- List the top 5 most ordered pizza types along with their quantities.
- Join the necessary tables to find the total quantity of each pizza category ordered.
- Determine the distribution of orders by hour of the day.
- Join relevant tables to find the category-wise distribution of pizzas.
- Group the orders by date and calculate the average number of pizzas ordered per day.
- Determine the top 3 most ordered pizza types based on revenue.
- Calculate the percentage contribution of each pizza type to total revenue.
- Analyze the cumulative revenue generated over time.
- Determine the top 3 most ordered pizza types based on revenue for each pizza category.



Q1. Retrieve the total number of orders placed.

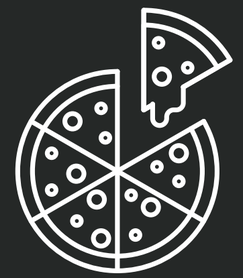
Query:

```
select count(*) as Total_Orders from orders;
```

Result:

Result Grid

	Total_Orders
▶	21350



Q2. Calculate the total revenue generated from pizza sales.

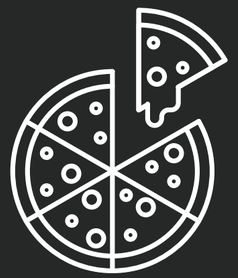
Query:

```
select round(sum(price*quantity), 0) as Total_Revenue from pizzas
inner join orderdetails on pizzas.pizza_id = orderdetails.pizza_id;
```

Result:

Result Grid

	Total_Revenue
▶	817860



Q3. Identify the highest-priced pizza.

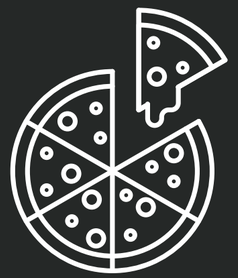
Query:

```
select pizza_types.name, pizzas.price
from pizzas join pizza_types using(pizza_type_id)
order by price desc limit 1;
```

Result:

Result Grid | Filter Ro

	name	price
▶	The Greek Pizza	35.95



Q4. Identify the most common pizza size ordered.

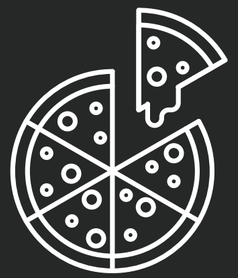
Query:

```
select pizzas.size, count(orderdetails.order_id) as CountOfSize from pizzas
inner join orderdetails on pizzas.pizza_id = orderdetails.pizza_id
group by pizzas.size order by CountOfSize desc;
```

Result:

Result Grid

size	CountOfSize
L	18526
M	15385
S	14137
XL	544
XXL	28



Q5. List the top 5 most ordered pizza types along with their quantities.

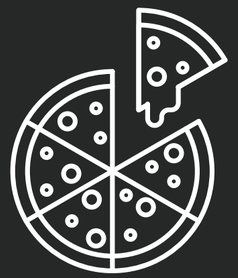
Query:

```
select pizza_types.name, sum(orderdetails.quantity) as Qtys
from pizza_types inner join pizzas on pizza_types.pizza_type_id
= pizzas.pizza_type_id inner join orderdetails on pizzas.pizza_id =
orderdetails.pizza_id group by pizza_types.name order by Qtys desc limit 5;
```

Result:

Result Grid | Filter Rows:

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



Q6. Join the necessary tables to find the total quantity of each pizza category ordered.

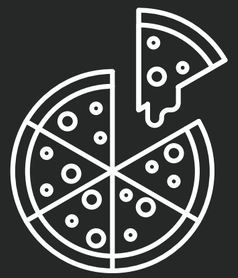
Query:

```
select pizza_types.category, sum(orderdetails.quantity) as Qtys
from pizza_types inner join pizzas on pizza_types.pizza_type_id
= pizzas.pizza_type_id inner join orderdetails on pizzas.pizza_id =
orderdetails.pizza_id group by pizza_types.category order by Qtys desc;
```

Result:

Result Grid

category	Qtys
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050



Q7. Determine the distribution of orders by hour of the day.

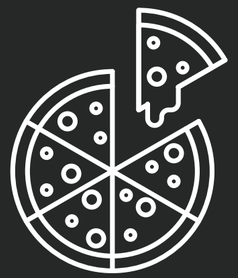
Query:

```
select hour(order_time) as Hours, count(order_id) as CntOfOrders
from orders group by Hours order by CntOfOrders desc;
```

Result:

Result Grid | Filter

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



Q8. Join relevant tables to find the category-wise distribution of pizzas.

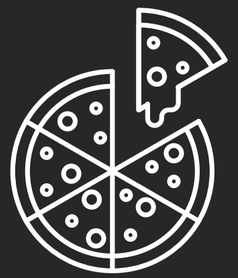
Query:

```
select category, count(pizza_type_id) as Pizza_distribution
from pizza_types group by category order by Pizza_distribution desc;
```

Result:

Result Grid | Filter Rows:

category	Pizza_distribution
Supreme	9
Veggie	9
Classic	8
Chicken	6



Q9. Group the orders by date and calculate the average number of pizzas ordered per day.

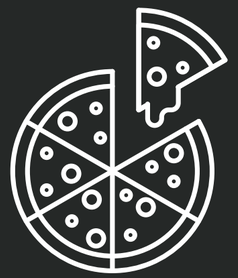
Query:

```
with mydata as(
select order_date, sum(quantity) as Qtys
from orders join orderdetails on orders.order_id = orderdetails.order_id
group by order_date)
select round(avg(Qtys),0) as 'average order per day' from mydata;
```

Result:

Result Grid | Filter Rows:

	average order per day
▶	138



Q10. Determine the top 3 most ordered pizza types based on revenue.

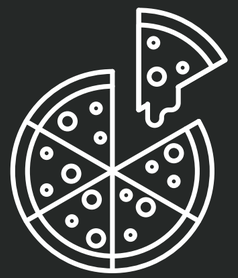
Query:

```
select pizza_types.name, sum(price*quantity) as revenue
from pizzas join pizza_types on pizzas.pizza_type_id = pizza_types.pizza_type_id
join orderdetails on pizzas.pizza_id = orderdetails.pizza_id
group by pizza_types.name order by revenue desc limit 3;
```

Result:

Result Grid | Filter Rows:

name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5



Q11. Calculate the percentage contribution of each pizza type to total revenue.

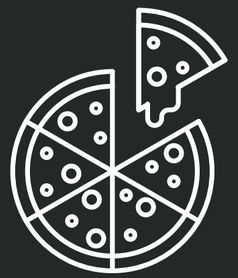
Query:

```
with mydata as(
select category , round(sum(quantity*price),2) as total_revenue
from pizzas join pizza_types
on pizzas.pizza_type_id = pizza_types.pizza_type_id
join orderdetails on pizzas.pizza_id = orderdetails.pizza_id
group by category)
select mydata.category,
concat( round((total_revenue/(select sum(total_revenue) from mydata))*100,2)," %")
as Percentage_Contribution
from mydata;
```

Result:

Result Grid | Filter Rows:

category	Percentage_Contribution
Classic	26.91 %
Veggie	23.68 %
Supreme	25.46 %
Chicken	23.96 %



Q12. Analyze the cumulative revenue generated over time.

Query:

```
with mydata as(
select order_date, round(sum(pizzas.price*orderdetails.quantity),2) as revenue
from orders join orderdetails on orders.order_id = orderdetails.order_id
join pizzas on pizzas.pizza_id = orderdetails.pizza_id group by order_date)
select mydata.order_date,
round(sum(revenue) over(order by order_date),2) as 'Cumulative Revenue' from mydata;
```

Result:

Result Grid | Filter Rows:

order_date	Cumulative 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

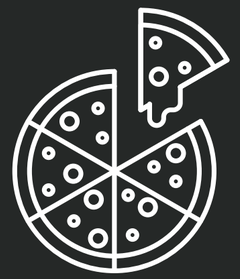
Result Grid | Filter Rows:

order_date	Cumulative Revenue
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

...

Result Grid | Filter Rows:

order_date	Cumulative Revenue
2015-12-27	810615.8
2015-12-28	812253
2015-12-29	813606.25
2015-12-30	814944.05
2015-12-31	817860.05



Q13. Determine the top 3 most ordered pizza types based on revenue for each pizza category.

Query:

```
with mydata1 as(
with mydata as(
select pizza_types.category, pizza_types.name,
sum(price*quantity) as revenue from pizza_types join pizzas on
pizza_types.pizza_type_id = pizzas.pizza_type_id join orderdetails
on orderdetails.pizza_id = pizzas.pizza_id group by
pizza_types.category, pizza_types.name)
select mydata.*, rank() over(partition by category order by revenue desc) as "rank"
from mydata) select * from mydata1 where mydata1.rank<=3 ;
```

Result:

Result Grid | Filter Rows: | Export: | Wrap Cell

category	name	revenue	rank
Chicken	The Thai Chicken Pizza	43434.25	1
Chicken	The Barbecue Chicken Pizza	42768	2
Chicken	The California Chicken Pizza	41409.5	3
Classic	The Classic Deluxe Pizza	38180.5	1
Classic	The Hawaiian Pizza	32273.25	2
Classic	The Pepperoni Pizza	30161.75	3
Supreme	The Spicy Italian Pizza	34831.25	1
Supreme	The Italian Supreme Pizza	33476.75	2
Supreme	The Sicilian Pizza	30940.5	3
Veggie	The Four Cheese Pizza	32265.70000000065	1
Veggie	The Mexicana Pizza	26780.75	2
Veggie	The Five Cheese Pizza	26066.5	3

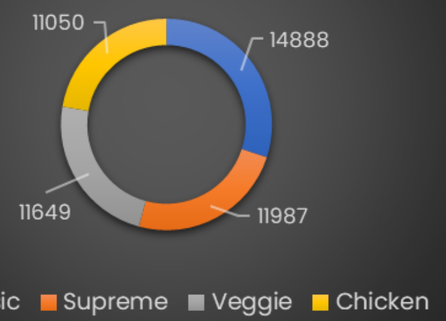
Total Orders
21350

Total Revenue
$ 817,860

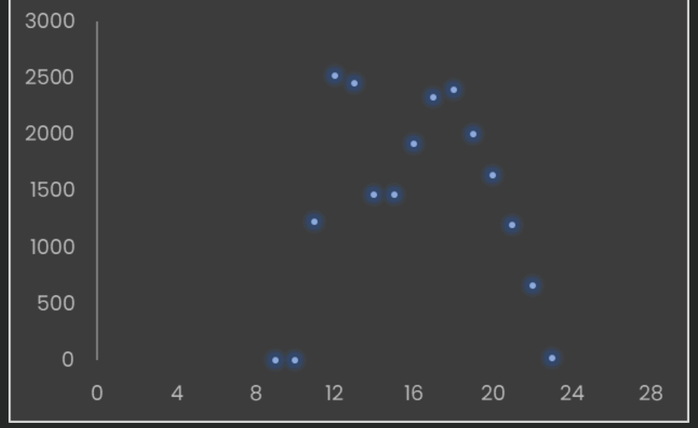
Highest Priced Pizza
The Greek Pizza - $ 35.95

Average Order Per Day
138

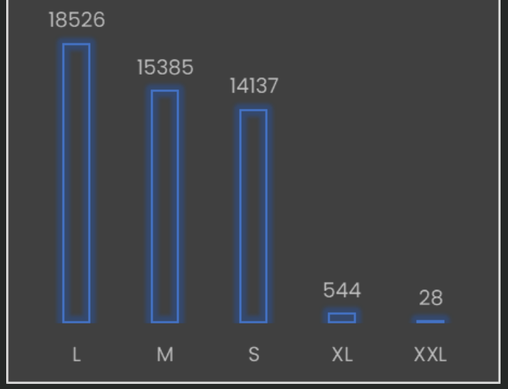
Pizza Category vs Quantity Ordered



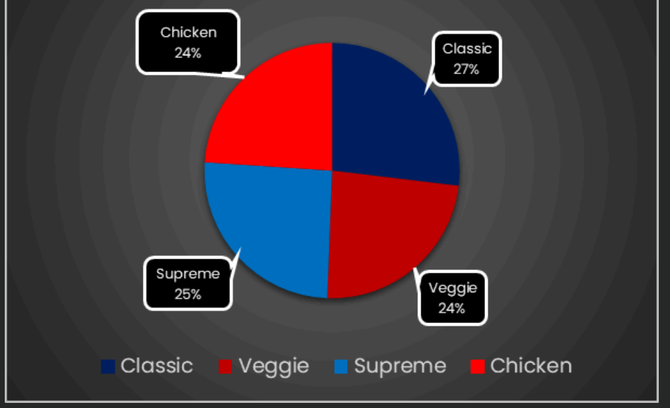
Distribution of Orders by Hour



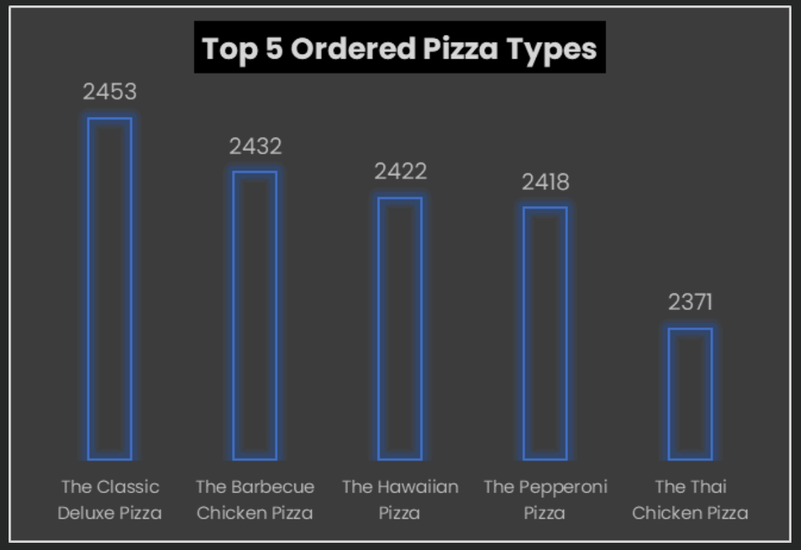
Count of size of Pizza Ordered



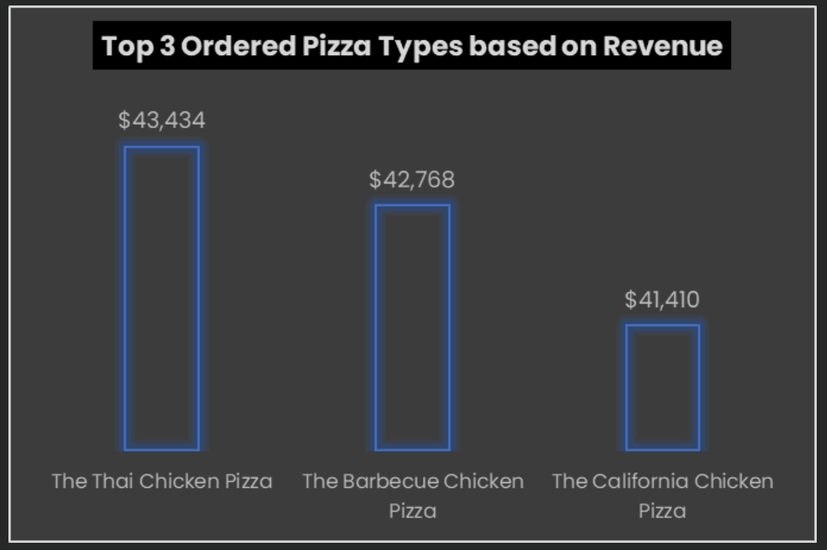
Percentage Contribution of Pizza Types based on Total Revenue



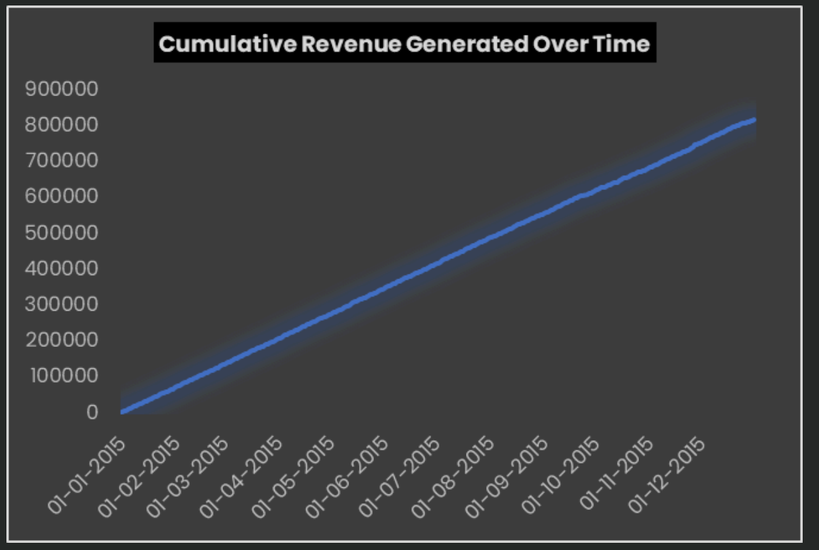
Top 5 Ordered Pizza Types

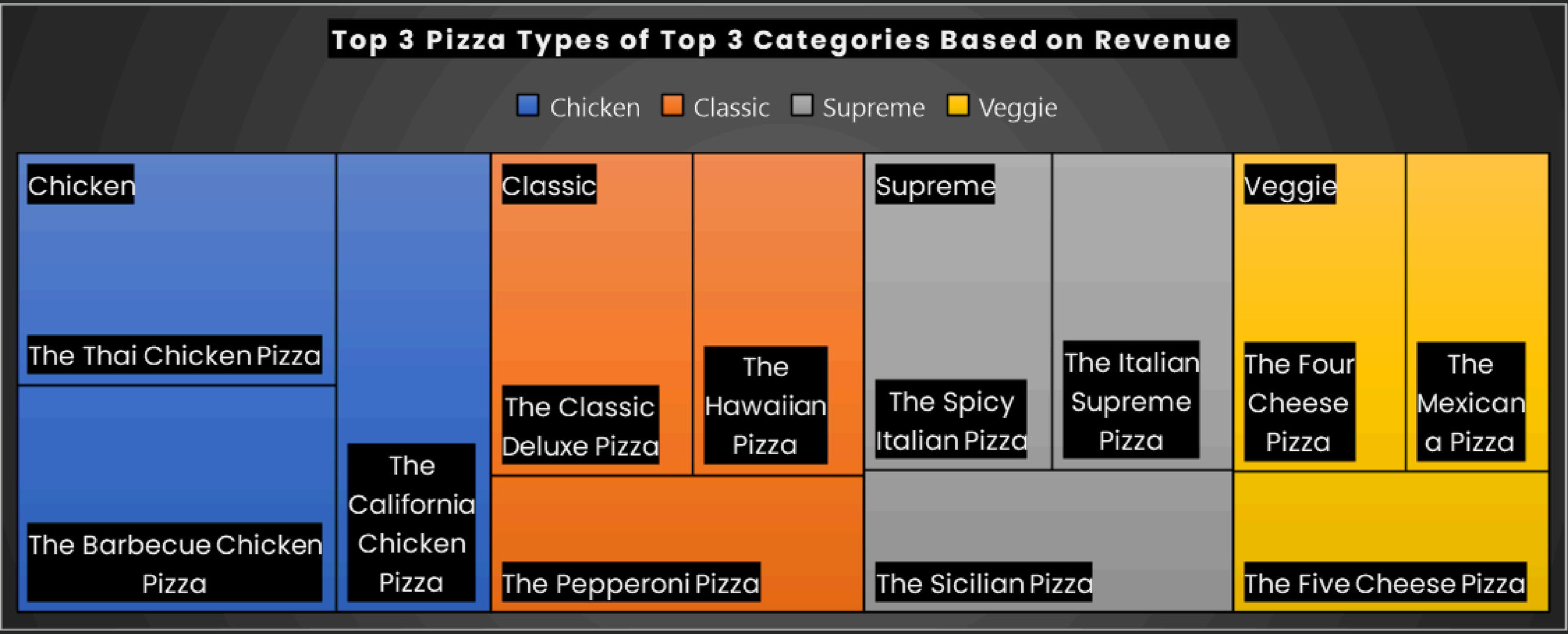


Top 3 Ordered Pizza Types based on Revenue



Cumulative Revenue Generated Over Time




Top 3 Pizza Types of Top 3 Categories Based on Revenue
Chicken
Classic
Supreme
Veggie
Chicken
The Thai Chicken Pizza
The Barbecue Chicken Pizza
The California Chicken Pizza
Classic
The Classic Deluxe Pizza
The Hawaiian Pizza
The Pepperoni Pizza
Supreme
The Spicy Italian Pizza
The Italian Supreme Pizza
The Sicilian Pizza
Veggie
The Four Cheese Pizza
The Mexican a Pizza
The Five Cheese Pizza

Thank You

Contact number : +91 6263349734
Email : shubhamraikwar276@gmail.com
LinkedIn : https://www.linkedin.com/in/shubham-raikwar/