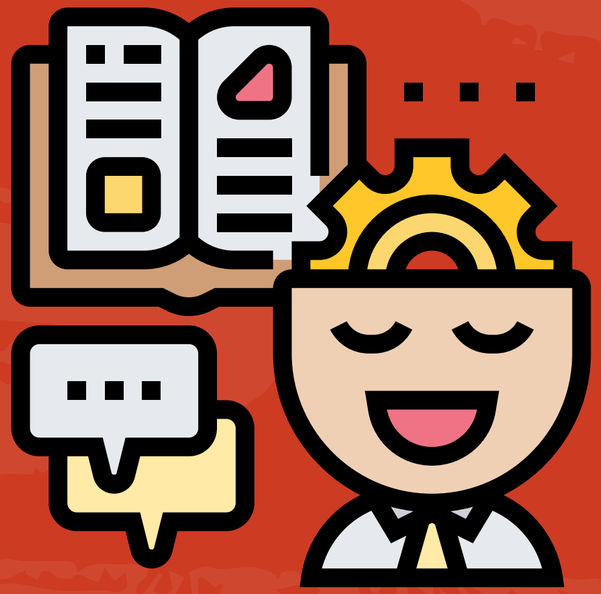
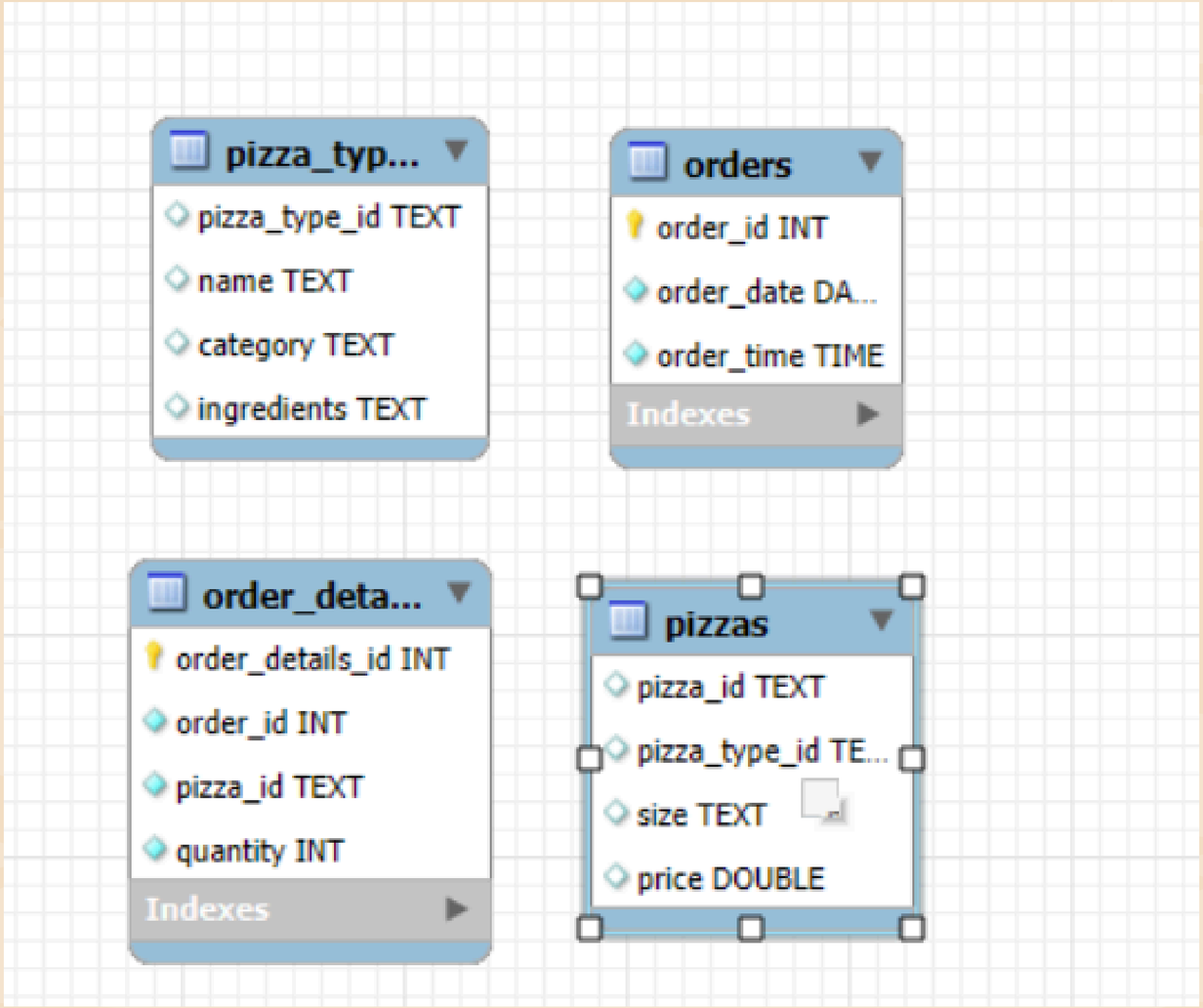
MySQL Project On Pizza Sales



HELLO

My name is Mahi and I have created this project to analyze pizza sales data using SQL queries. Through this analysis, I aim to uncover key trends in customer preferences ,revenue generation and popular pizza types to help businesses make data driven decision. I hope you find the insight helpful.

Schema of pizza sales



Retrieve the total number of order placed

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

Result Grid

	total_orders
▶	21350

Calculate the total revenue generated from pizza sales

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

Result Grid | Filter Rows:

total_sales
817860.05

Identify the highest-priced pizza

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

Result Grid | Filter Rows:

	name	price
▶	The Greek Pizza	35.95

Identify the most common pizza size ordered.

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

Result Grid | Filter Rows:

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

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

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

Result Grid | Filter Rows:

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

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

```sql
SELECT pizza_types.category,SUM(order_details.quantity) AS total_quantity
FROM pizzas
JOIN order_details
ON pizzas.pizza_id=order_details.pizza_id
JOIN pizza_types
ON pizzas.pizza_type_id=pizza_types.pizza_type_id
GROUP BY pizza_types.category
ORDER BY total_quantity DESC;
```

Result Grid | Filter Rows:

category	total_quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

Determine the distribution of orders by hour of the day.

```sql
SELECT HOUR(order_time) AS Hour,COUNT(order_id) AS order_count
FROM orders
GROUP BY HOUR(order_time);
```

Result Grid | Filter Rows:

Hour	order_count
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642
21	1198

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

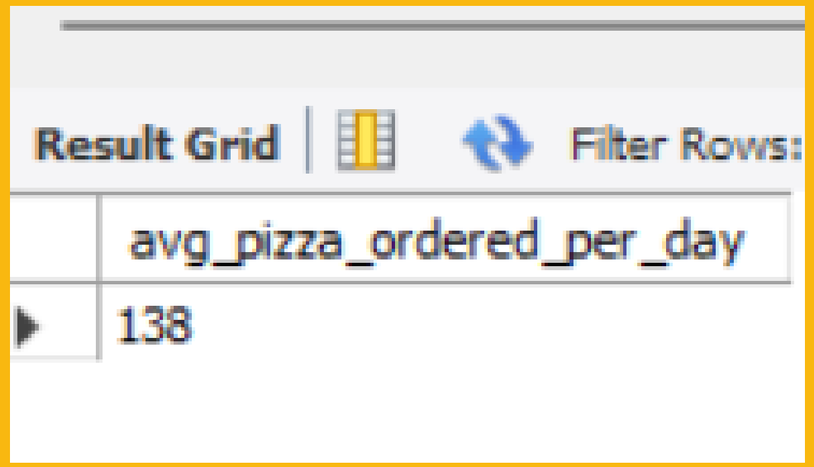
```sql
SELECT category,COUNT(name)
FROM pizza_types
GROUP BY category;
```

Result Grid | Filter Rows:

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

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

```
SELECT ROUND(AVG(quantity),0) AS avg_pizza_ordered_per_day FROM (SELECT orders.order_date,SUM(order_details.quantity) AS quantity
FROM orders
JOIN order_details
ON orders.order_id=order_details.order_id
GROUP BY orders.order_date) AS order_quantity;
```



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

```sql
SELECT pizza_types.name,SUM(pizzas.price*order_details.quantity) AS total_revenue
FROM pizzas
JOIN order_details
ON pizzas.pizza_id=order_details.pizza_id
JOIN pizza_types
ON pizza_types.pizza_type_id=pizzas.pizza_type_id
GROUP BY pizza_types.name
ORDER BY total_revenue DESC LIMIT 3;
```

Result Grid | Filter Rows:

name	total_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.

```
ROUND(SUM(order_details.quantity*pizzas.price) / (SELECT ROUND(SUM(pizzas.price*order_details.quantity),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 pizzas.pizza_type_id= pizza_types.pizza_type_id
JOIN order_details
ON order_details.pizza_id=pizzas.pizza_id
GROUP BY category
ORDER BY revenue DESC;
```

Result Grid | Filter Rows:

category	revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

Analyze the cumulative revenue generated over time.

```sql
SELECT order_date,SUM(revenue) OVER(ORDER BY order_date) AS cum_revenue
FROM
(SELECT orders.order_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 order_date) AS sales;
```

Result Grid | Filter Rows:

order_date	cum_revenue
2015-01-01	2713.8500000000004
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	23990.350000000002
2015-01-11	25862.65

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

```sql
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;
```

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

name	revenue
The California Chicken Pizza	41409.5
The Southwest Chicken Pizza	34705.75
The Chicken Alfredo Piz	
The Chicken Pesto Pizza	16701.75
The Pepperoni Pizza	30161.75
The Greek Pizza	28454.100000000013
The Italian Capocollo Pizza	25094
The Napolitana Pizza	24087
The Big Meat Pizza	22968
The Pepperoni, Mushroom, and Peppers Pizza	18834.5
The Sicilian Pizza	30940.5

The Southwest Chicken Pizza

Thank You!

I hope this analysis provided valuable insights into pizza sales trends and customer behavior. Data can be a powerful tool for drivind useful decisions and I’m passionate about exploring new opportunities for analysis.