



LIST OF CONTENTS

Introduction

Schema

Query

Insights

Thank You!!!

PIZZA SALES REPORT

Dataset contains details of pizza orders (i.e date, time, quantity and type of pizza ordered) and information about pizza (i.e size, price, category and ingredients)

The objective of the project is to gain valuable insights into the sales operations, customer interest and pizza category in demand

Let's start our adventure in the world of pizza!

SCHEMA

order_details		pizzas	
order_details_id	int	pizza_id	text
order_id	int	pizza_type_id	text
pizza_id	text	size	text
quantity	int	price	double
Pizza_types		Orders	
pizza_type_id	text	order_id	int
name	text	order_date	date
category	text	order_time	time

Retrieve the total number of orders placed.

```
#Retrieve the total number of orders placed.
```

```
Select count(distinct order_id) as total_no_of_orders from orders;
```

	total_no_of_orders
▶	21350



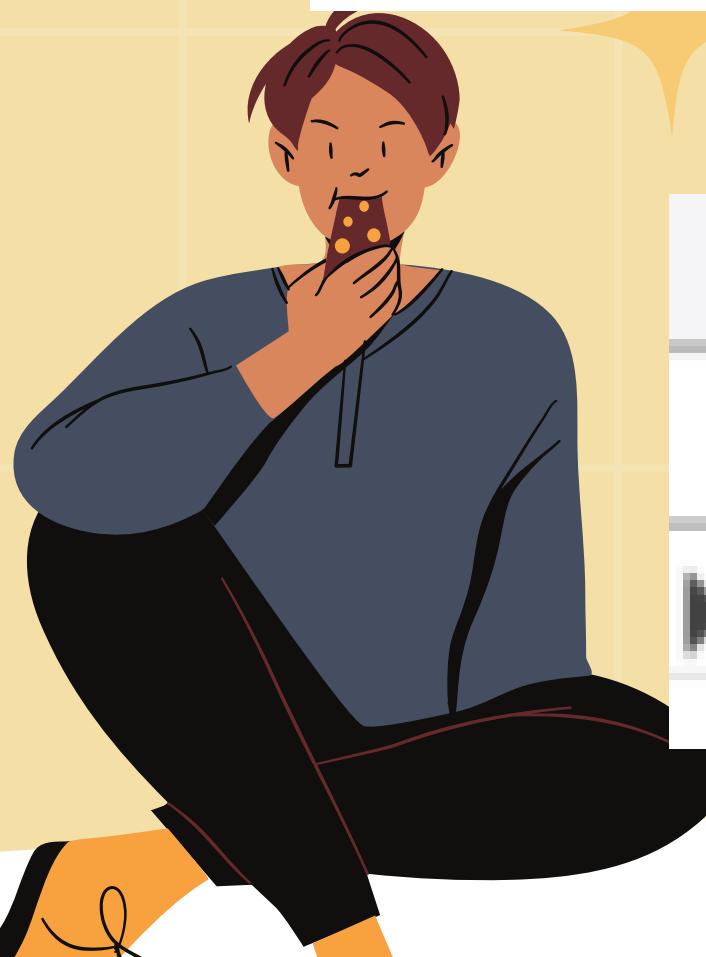
Calculate the total revenue generated from pizza sales

```
#Calculate the total revenue generated from pizza sales.  
SELECT  
    ROUND(SUM(x.revenue), 2) AS total_revenue  
FROM  
    (SELECT  
        (order_details.quantity * pizzas.price) AS revenue  
    FROM  
        order_details  
    JOIN pizzas ON pizzas.pizza_id = order_details.pizza_id) x;  
  
SELECT  
    sum(order_details.quantity * pizzas.price) AS revenue  
FROM  
    order_details  
JOIN pizzas ON pizzas.pizza_id = order_details.pizza_id;
```

total_revenue
► 817860.05

Identify the highest priced pizza

```
#Identify the highest-priced pizza.  
  
SELECT  
    pizza_types.name, pizzas.price  
FROM  
    pizzas  
    JOIN  
        pizza_types ON pizza_types.pizza_type_id = pizzas.pizza_type_id  
ORDER BY price DESC  
LIMIT 1;
```



name	price
The Greek Pizza	35.95



Identify the most common pizza size ordered

```
-- Identify the most common pizza size order
SELECT
    SUM(quantity) AS total_quantity, size
FROM
    pizzas
        JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id
GROUP BY (size)
ORDER BY total_quantity DESC
LIMIT 1;
```

	total_quantity	size
▶	18956	L



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

```
-- List the top 5 most ordered pizza types along with their quantities.  
SELECT  
    x.name  
FROM  
    (SELECT  
        name, SUM(quantity) AS total_qu  
    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 name  
    ORDER BY total_qu DESC  
    LIMIT 5) x;
```

name
The Classic Deluxe Pizza
The Barbecue Chicken Pizza
The Hawaiian Pizza
The Pepperoni Pizza
The Thai Chicken Pizza

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

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

```
SELECT
```

```
    category, SUM(quantity) AS total_qu
```

```
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 category
```

```
ORDER BY total_qu DESC;
```

category	total_qu
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

Determine the distribution of orders by hour of the day

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

```
SELECT  
    COUNT(order_id) AS order_count,  
    HOUR(order_time) AS order_hour  
FROM  
    orders  
GROUP BY HOUR(order_time)  
ORDER BY order_count DESC;
```

	order_count	order_hour
▶	2520	12
	2455	13
	2399	18
	2336	17
	2009	19
	1920	16
	1642	20
	1472	14
	1468	15
	1231	11
	1198	21
	663	22



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

```
-- Join relevant tables to find the category-wise distribution of pizzas.  
•  
SELECT  
    category, COUNT(name) AS total_qu  
FROM  
    pizzas  
    JOIN  
    pizza_types ON pizza_types.pizza_type_id = pizzas.pizza_type_id  
GROUP BY category  
ORDER BY total_qu DESC
```



	category	total_qu
▶	Veggie	27
	Classic	26
	Supreme	25
	Chicken	18

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

```
-- Group the orders by date and calculate the average number of pizzas ordered per day.  
SELECT  
    ROUND(AVG(x.total_quantity), 2)  
FROM  
(SELECT  
    order_date, (SUM(quantity)) AS total_quantity  
FROM  
    orders  
JOIN order_details ON order_details.order_id = orders.order_id  
GROUP BY order_date) x;
```

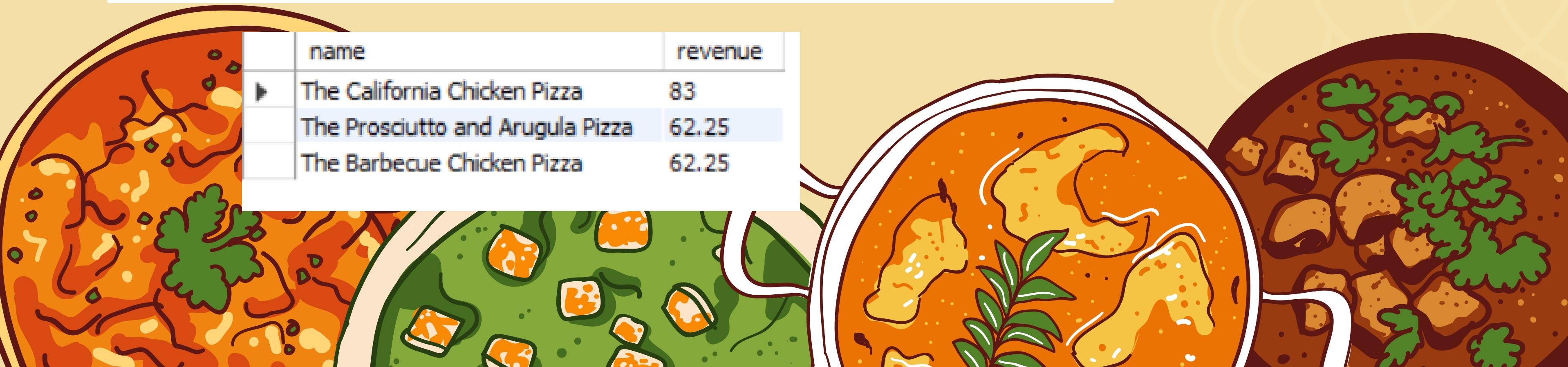
▶ 138.47



Determine the top 3 most ordered pizza types based on revenue

```
-- Determine the top 3 most ordered pizza types based on revenue.  
SELECT pizza_types.name,  
       (order_details.quantity * pizzas.price) AS revenue  
  FROM  
    order_details  
   JOIN pizzas ON pizzas.pizza_id = order_details.pizza_id  
   JOIN  
pizza_types ON pizza_types.pizza_type_id = pizzas.pizza_type_id order by revenue desc limit 3;
```

name	revenue
The California Chicken Pizza	83
The Prosciutto and Arugula Pizza	62.25
The Barbecue Chicken Pizza	62.25



Calculate the percentage contribution of each pizza type to total revenue

```
-- Calculate the percentage contribution of each pizza type to total revenue.  
SELECT pizza_types.category,  
       sum(order_details.quantity * pizzas.price)/(SELECT  
              sum(order_details.quantity * pizzas.price) AS revenue  
           FROM  
              order_details  
           JOIN pizzas ON pizzas.pizza_id = order_details.pizza_id) AS percent_revenue  
        FROM  
          order_details  
        JOIN pizzas ON pizzas.pizza_id = order_details.pizza_id  
        JOIN  
          pizza_types ON pizza_types.pizza_type_id = pizzas.pizza_type_id group by category order by percent_revenue desc;
```



	category	percent_revenue
▶	Classic	0.26905960255669903
	Supreme	0.2545631126009884
	Chicken	0.23955137556847494
	Veggie	0.23682590927384783

Analyze the cumulative revenue generated over time

-- Analyze the cumulative revenue generated over time.

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

	order_date	revenue	cum_rev
▶	2015-01-01	2713.8500000000004	2713.8500000000004
	2015-01-02	2731.899999999996	5445.75
	2015-01-03	2662.399999999996	8108.15
	2015-01-04	1755.4500000000003	9863.6
	2015-01-05	2065.95	11929.55
	2015-01-06	2428.95	14358.5
	2015-01-07	2202.200000000003	16560.7
	2015-01-08	2838.349999999995	19399.05
	2015-01-09	2127.350000000004	21526.4
	2015-01-10	2463.95	23990.350000000002
	2015-01-11	1872.300000000002	25862.65
	2015-01-12	1919.050000000002	27781.7



Conclusion



- Total 21350 orders amounting to \$817860.05 have been placed
- Greek Pizza is the highest priced pizza
- Large sized pizza are most commonly ordered
- Rush hours are between 12 noon to 1 pm
- On average 138 pizzas are ordered per day
- Classic category pizza generates maximum revenue



THANK YOU

“Have fun making your own pizza
and enjoy every bite”

