SQL sales analyzation by Ardhendu

PIZZA + PARTY = FUN

Objective:

The goal of this project is to analyze and generate insights from pizza sales data. By performing a series of queries, we aim to understand key metrics such as total orders, revenue, and pizza popularity, as well as more detailed insights like the distribution of orders over time and category-wise performance.

Total Orders and Revenue:

We calculated the total number of orders placed and the total revenue generated from pizza sales. This provides a high-level overview of the business performance.

Highest-Priced Pizza and Common Pizza Size:

Identified the most expensive pizza and the most frequently ordered pizza size, giving insight into customer preferences and pricing.

Top 5 Most Ordered Pizza Types:

Listed the top 5 pizza types based on order quantities to understand popular choices among customers.

Pizza Category Quantity and Distribution by Hour:

Determined the total quantity of each pizza category ordered and analyzed the distribution of orders by hour to optimize staffing and inventory.

Category-wise Pizza Distribution:

Examined the distribution of pizzas across different categories to assess category performance.

Average Daily Pizza Orders:

Calculated the average number of pizzas ordered per day to identify trends and adjust operational strategies.

Top 3 Most Ordered Pizza Types by Revenue:

Identified the top 3 pizza types contributing the most to revenue, aiding in targeted marketing and menu adjustments.

Revenue Contribution by Pizza Type:

Calculated the percentage contribution of each pizza type to total revenue to prioritize high-revenue items.

Cumulative Revenue Analysis:

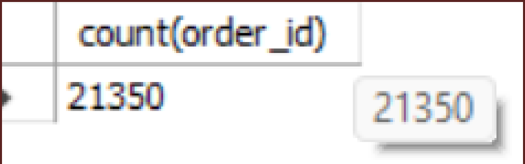
Analyzed cumulative revenue over time to understand growth patterns and forecast future performance.

Top 3 Pizza Types by Revenue for Each Category:

Determined the top 3 pizza types based on revenue within each category, enabling more precise inventory and promotional strategies.

1.Retrieve the total number of orders placed ?

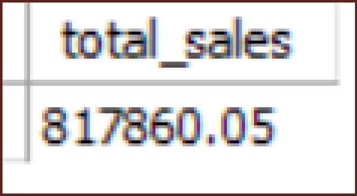
```
SELECT count(order_id)
FROM pizza.orders;
```



```
count(order_id)
21350
```

2.Calculate the total revenue generated from pizza sales ?

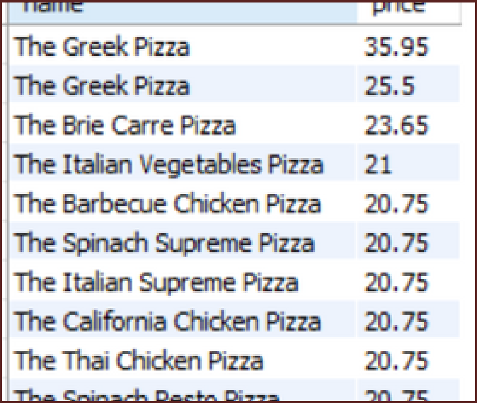
```
SELECT

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



total_sales
817860.05

3.Identify Top 10 highest-priced pizza ?

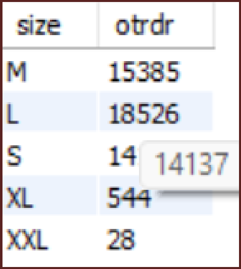
```
SELECT
pizza_types.name, pizzas.price
FROM
pizza_types
JOIN
pizzas ON pizzas.pizza_type_id
= pizza_types.pizza_type_id
ORDER BY pizzas.price DESC
LIMIT 10
```



name	price
The Greek Pizza	35.95
The Greek Pizza	25.5
The Brie Carre Pizza	23.65
The Italian Vegetables Pizza	21
The Barbecue Chicken Pizza	20.75
The Spinach Supreme Pizza	20.75
The Italian Supreme Pizza	20.75
The California Chicken Pizza	20.75
The Thai Chicken Pizza	20.75
The Spinach Pesto Pizza	20.75

4.Identify the most common pizza size ordered?

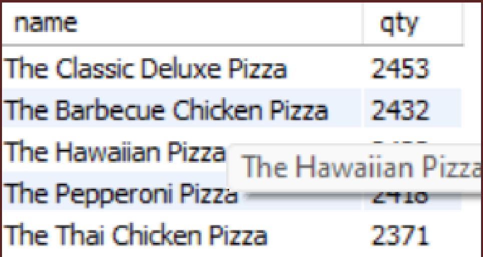
```
SELECT
pizzas.size,
COUNT(order_details.order_details_id) AS otrdr
FROM
pizzas
JOIN
order_details ON
pizzas.pizza_id =
order_details.pizza_id
GROUP BY pizzas.size;
```



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

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

```
SELECT
pt.name, SUM(o.quantity) AS qty
FROM
pizza_types pt,
pizzas p,
order_details o
WHERE
p.pizza_id = o.pizza_id
 AND pt.pizza_type_id =
p.pizza_type_id
GROUP BY pt.name
ORDER BY qty DESC
LIMIT 5;
```



name	qty
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	[illegible]
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371

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

```
select
pt.name, SUM(o.quantity) AS
qty from pizza_types pt
join pizzas p on
pt.pizza_type_id =
p.pizza_type_id
join order_details o on
o.pizza_id = p.pizza_id
GROUP BY pt.name
ORDER BY qty DESC ;
```

7.Determine the distribution of orders by hour of the day ?

```
SELECT
pt.category, SUM(o.quantity)
AS qty
FROM
pizza_types pt,
pizzas p,
order_details o
WHERE
p.pizza_id = o.pizza_id
    AND pt.pizza_type_id =
p.pizza_type_id
GROUP BY pt.category
ORDER BY qty DESC;
```

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

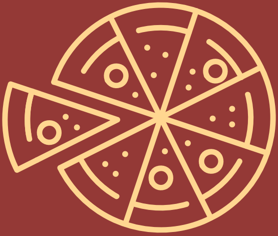
```
SELECT
  category,
COUNT(name)
 FROM
  pizza.pizza_types
GROUP BY category;
```

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

```
SELECT
ROUND(AVG(avg_qty), 0)
avg_order_per_day
FROM
(SELECT
o.date, SUM(od.quantity)
avg_qty
FROM
orders o, order_details od
WHERE
o.order_id = od.order_id
GROUP BY o.date) AS oq
```

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

```
SELECT
pt.name, SUM(od.quantity *
p.price) AS amount
FROM
pizza_types pt,
order_details od,
pizzas p
WHERE
pt.pizza_type_id =
p.pizza_type_id
AND p.pizza_id = od.pizza_id
GROUP BY pt.name
ORDER BY amount DESC
LIMIT 3;
```



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

```
SELECT pt.name ,round(
(sum(od.quantity * p.price) /(SELECT

ROUND(SUM(order_details.quantity *
pizzas.price),
2) AS total_sales
FROM
order_details
JOIN
pizzas ON order_details.pizza_id =
pizzas.pizza_id)) * 100,2 )as
persentage FROM pizza_types
pt,order_details od,pizzas p
where pt.pizza_type_id =
p.pizza_type_id and p.pizza_id =
od.pizza_id
group by pt.name
order by persentage desc
```

12.Analyze the cumulative revenue generated over time?

```
SELECT date,sum(rev)over
(order by date) as c_rev from
(SELECT
o.date,round(sum(p.price *
od.quantity),0) as rev FROM
orders o, pizzas p,
order_details od where
od.pizza_id= p.pizza_id and
od.order_id= o.order_id
group by o.date order by rev)
as sale;
```

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

```
select name, rev, category from
(select category,
name,rev,rank()over(partition by
category order by rev desc) as rn
from
(SELECT
pt.category,pt.name,round(sum(p
.price * od.quantity),0) as rev
FROM pizza_types pt, pizzas p,
order_details od where
pt.pizza_type_id= p.pizza_type_id
and od.pizza_id= p.pizza_id
group by pt.category, pt. name )
as a) as b where rn <=3;
```

Conclusion:

This project provides a comprehensive analysis of pizza sales data, offering valuable insights into customer behavior, product performance, and operational efficiency. The findings can inform strategic decisions, enhance customer satisfaction, and drive revenue growth.

thankyou

MADE BY ARDHENDU