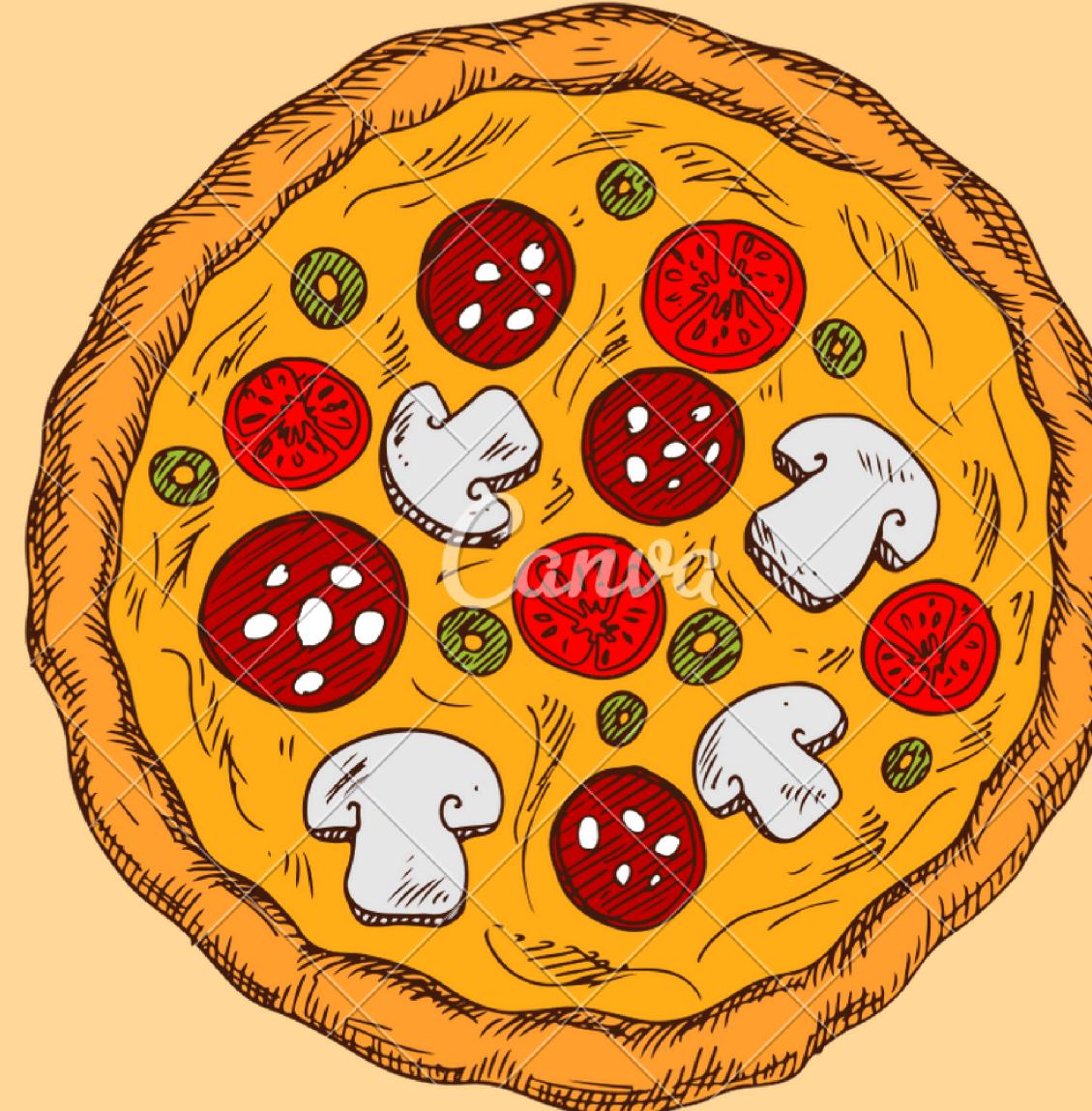


Pizza SALES SQL PROJECT

Delicious Pizza for Everyone!





Hello!

My name is Hrithik Gogna
In this project I have utilised SQL Query's to answer
some questions related to pizza sales

Introduction to the Pizza Sales Database

The "pizza_sales.zip" database provides comprehensive insights into pizza sales, including order details, customer information, menu items, and geographic data. This dataset enables analysis of customer preferences, market trends, and operational efficiencies in the pizza industry, empowering businesses to optimize strategies and enhance profitability.

Dataset can be accessed via following link-

<https://github.com/HrithikGogna/SQL-PROJECT-pizza-sales.git>

Calculate the total revenue generated from pizza sales

```
SELECT  
    ROUND(SUM(od.quantity * p.price), 3) AS total_revenue  
FROM  
    ORDERS_details AS od  
    JOIN  
    pizzas AS p ON od.pizza_id = p.pizza_id;
```

total_revenue
817860.05

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

SELECT

```
    pizza_types.name as Pizza_name,  
    SUM(orders_details.quantity) AS Quantity_Ordered
```

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 Quantity_Ordered **DESC**

LIMIT 5;

Pizza_name	Quantity_Ordered
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	2422
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371

Determine the distribution of orders by hour of the day

```
select HOUR(order_time) AS Hour, COUNT(order_id) As Orders_recieved  
from orders GROUP BY HOUR(order_time) order by Hour;
```

Hour	Orders_recieved
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

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

```
SELECT  
    ROUND(AVG(quantity), 0) AS Avg_pizza_order  
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_pizza_order
138

Determine the top 3 most ordered pizza types based on revenue

```
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 orders_details.pizza_id = pizzas.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

Calculate the percentage contribution of each pizza type to total revenue

```
SELECT
```

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

Analyze the cumulative revenue generated over time

```
select order_date,  
round(sum(revenue) over (order by order_date),2)as cum_revenue  
from  
(select orders.order_date,  
sum(orders_details.quantity * pizzas.price) as revenue  
from orders_details join pizzas  
on orders_details.pizza_id = pizzas.pizza_id  
join orders  
on orders.order_id = orders_details.order_id  
group by orders.order_date) as sales;
```

order_date	cum_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

2015-12-26	809196.8
2015-12-27	810615.8
2015-12-28	812253
2015-12-29	813606.25
2015-12-30	814944.05
2015-12-31	817860.05

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

```
select category, 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((orders_details.quantity) * pizzas.price) as revenue
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.category, pizza_types.name) as a)as b
where rn<=2;
```

category	revenue
Chicken	43434.25
Chicken	42768
Classic	38180.5
Classic	32273.25
Supreme	34831.25
Supreme	33476.75
Veggie	32265.7000
Veggie	26780.75

THANK YOU