**PIZZA LAND SALES ANALYSIS**

1) Calculate the total number of orders placed

Select Count(order\_id) as Total\_Orders From orders;

2) Calculate the total revenue generated from pizza sales

SELECT floor(SUM(order\_details.quantity \* pizzas.price)) AS Total\_Sales

FROM order\_details INNER JOIN pizzas

ON pizzas.pizza\_id=order\_details.pizza\_id;

3) Identify Total Pizzas Sold

Select SUM(order\_details.quantity) as Total\_Pizzas\_Sold

FROM order\_details;

4) Identify pizzas ordered by size.

Select pizzas.size, count(order\_details.order\_details\_id) AS Total\_quantity

From pizzas Inner join order\_details

ON order\_details.pizza\_id= pizzas.pizza\_id

Group by pizzas.size

ORDER BY Total\_quantity DESC;

5) Calculate average order value

SELECT

ROUND(SUM(p.price \* od.quantity) / COUNT(DISTINCT o.order\_id), 2) AS avg\_order\_value

FROM

order\_details od

JOIN

pizzas p ON od.pizza\_id = p.pizza\_id

JOIN

orders o ON od.order\_id = o.order\_id;

6) Find the total quantity of each pizza category ordered.

Select pizza\_types.category, Count(order\_details.quantity) as Total\_Quantity

FROM pizza\_types INNER JOIN pizzas

ON pizzas.pizza\_type\_id=pizza\_types.pizza\_type\_id

Inner Join order\_details

ON order\_details.pizza\_id=pizzas.pizza\_id

Group by pizza\_types.category;

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

SELECT HOUR(time) AS Hour, COUNT(order\_id) AS Total\_orders

FROM orders GROUP BY HOUR(time) ORDER BY Hour ASC;

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

Select pizza\_types.name, SUM(order\_details.quantity \*pizzas.price) as revenue

from pizza\_types INNER JOIN pizzas

ON pizzas.pizza\_type\_id=pizza\_types.pizza\_type\_id INNER JOIN order\_details

ON order\_details.pizza\_id=pizzas.pizza\_id

Group by pizza\_types.name ORDER BY revenue DESC LIMIT 3;

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

Select pizza\_types.category, CONCAT(ROUND(SUM(order\_details.quantity \* pizzas.price)\*100 / (Select

SUM(order\_details.quantity \* pizzas.price) AS Total\_Sales

FROM order\_details INNER JOIN pizzas

ON pizzas.pizza\_id=order\_details.pizza\_id),2),'%') AS revenue\_percentage

from pizza\_types INNER JOIN pizzas

ON pizzas.pizza\_type\_id=pizza\_types.pizza\_type\_id INNER JOIN order\_details

ON order\_details.pizza\_id=pizzas.pizza\_id

Group by pizza\_types.category order by revenue\_percentage desc;

10) Analyze the cumulative revenue generated over time.

SELECT month,

Round(SUM(revenue) OVER (ORDER BY month),2) AS cum\_revenue

FROM (

SELECT date\_format(orders.date,'%Y-%m') as month,

SUM(order\_details.quantity \* pizzas.price) AS revenue

FROM order\_details

INNER JOIN pizzas ON order\_details.pizza\_id = pizzas.pizza\_id

INNER JOIN orders ON orders.order\_id = order\_details.order\_id

GROUP BY date\_format(orders.date,'%Y-%m')

) AS sales;

11) Find pizza size revenue contribution by each category

WITH size\_revenue AS (

SELECT pt.category, p.size,

SUM(od.quantity \* p.price) AS revenue

FROM

order\_details od

JOIN pizzas p ON od.pizza\_id = p.pizza\_id

JOIN pizza\_types pt ON p.pizza\_type\_id = pt.pizza\_type\_id

GROUP BY

pt.category, p.size

),

total\_category\_revenue AS (

SELECT category, SUM(revenue) AS total\_revenue

FROM

size\_revenue

GROUP BY

category

)

SELECT

sr.category,

sr.size,

ROUND(sr.revenue, 2) AS revenue,

CONCAT(ROUND((sr.revenue / tcr.total\_revenue) \* 100, 2),'%') AS category\_percetage

FROM

size\_revenue sr

JOIN

total\_category\_revenue tcr ON sr.category = tcr.category

ORDER BY

sr.category, category\_percetage DESC;