PIZZA SALES REPORT

22 june , 2024

INTRODUCTION

Welcome to our Sales Report of pizzhut presentention. Today, we delve into a comprehensive overview of our sales performance, exploring the highs, challenges, and strategic insights that have shaped our journey. This presentation is more than just numbers, it's a narrative of our collective efforts, showcasing the impact of our sales strategies and the pathways to future success.

RETRIVE THE TOTAL NO OF ORDER PLACE .

```
select count(order_id) as
total_orders from orders;
```

total_orders
21350

IDENTIFY THE HIGHEST PRICE PIZZA.

```
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;
```

THE GREEK PIZZA
35.95

CALCULATE THE TOTAL NUMBER OF PIZZA SALES.

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

TOTAL_SALES
146557.5

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED

```
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 limit 1;
```

SIZE,	ORDER_COUNT
L	3374

LIST THE TOP 5 MOST ORDERED PIZZA TYPES ALONG WITH THEIR QUANTITIES .

```
SELECT PIZZA_TYPES.NAME,
SUM(ORDER_DETAILS.QUANTITY) AS QUANTITY
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.NAME
ORDER BY QUANTITY DESC LIMIT 5;
```

TOTAL QUANTITY OF EACH PIZZA ORDERED.

```
select pizza_types.category,
sum(order_details.quantity) as quantity
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
order by quantity desc;
```

Chicken
1955
Classic
2648
Supreme
2165
Veggie
2124

DETERMINES THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY

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

JOIN RELEVANT TABLES TO FIND THE CATEGORY WISE DISTRIBUTION OF PIZZAS

```
select category , count(name) from pizza_types
group by category;
```

GROUPS THE ORDERS BY DATE AND CALCULATE THE AVG NUMBER OF PIZZAS PER DAY

```
SELECT
    ROUND(AVG(quantity), 0)
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 THE REVENUE

```
SELECT
    pizza_types.name,
    SUM(order_details.quantity * pizzas.price) 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 pizza_types.name
ORDER BY revenue DESC
LIMIT 3;
```

CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPE OF TOTAL REVENUE .

```
select pizza_types.category,
round(sum(order_details.quantity * pizzas.price) / (select
round(sum(order_details.quantity * pizzas.price),
2) as total_sales
from order_details
join
pizzas on pizzas.pizza_id = order_details.pizza_id) *100,2) 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
order by revenue desc;
```

ANALYZE THE CUMULATICE REVENUE GENERATED OVER TIME

```
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 orders.order_date) as sales;
```

DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE FOR EACH PIZZA CATEGORY.

```
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;
```