Pizza Sales Analyze Report Armed with -SQL

-Yogesh Kumar Saini

Pizza!

Project Overview:

- This project focuses on analyzing data from Pizza Hut's sales and orders.
- We will explore insights related to order trends, revenue, and pizza preferences.

- **Key Areas of Analysis**:

- Total Orders
- Total Revenue
- Popular Pizza Types & Sizes
- Hourly Distribution of Orders
- Category-wise Distribution of Pizzas
- Cumulative Revenue Trends

Database Schema

- **Tables Involved**:
  - orders: Contains order details such as order_id, order_date, and order_time.
  - orders_details: Contains order details for each pizza in the order.
  - pizza_types: Contains different pizza categories and types.
  - pizzas: Contains pizza details like size, price, and pizza type.
- **Relationships**:
  - orders <–> orders_details
  - pizza_types <–> pizzas
  - orders_details <–> pizzas

Total Orders Placed

```sql
SELECT
    COUNT(order_id) AS total_orders
FROM
    orders;
```

Total Revenue Generated

```
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
```

Highest-Priced Pizza

```sql
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;
```

Most Common Pizza Size Ordered

```sql
SELECT
    pizzas.size,
    COUNT(orders_details.order_details_id) AS order_count
FROM
    pizzas
        JOIN
    orders_details ON pizzas.pizza_id = orders_details.pizza_id
GROUP BY pizzas.size
ORDER BY order_count DESC;
```

Top 5 Most Ordered Pizza Types

```sql
SELECT
    pizza_types.name,
    SUM(orders_details.quantity) AS total_orders
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 total_orders DESC
LIMIT 5;
```

Quantity of Each Pizza Category Ordered

```sql
SELECT
    pizza_types.category,
    SUM(orders_details.quantity) AS quantity
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 quantity DESC;
```

Distribution of Orders by Hour

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

Category-wise Distribution of Pizzas

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

Average Number of Pizzas Ordered Per Day

```sql
SELECT
    ROUND(AVG(quantity), 0)
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;
```

Top 3 Most Ordered Pizza Types Based on Revenue

```sql
SELECT
    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.name
ORDER BY revenue DESC
LIMIT 3;
```

Percentage Contribution of Each Pizza Type to Revenue

```sql
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 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
ORDER BY revenue DESC;
```

Cumulative Revenue Over Time

```sql
SELECT
    order_date,
    SUM(revenue) OVER (ORDER BY order_date) AS cum_revenue
FROM (
    SELECT
        o.order_date,
        SUM(od.quantity * p.price) AS revenue
    FROM
        orders_details od
    JOIN pizzas p ON od.pizza_id = p.pizza_id
    JOIN orders o ON o.order_id = od.order_id
    GROUP BY
        o.order_date
) AS Sales;
```

Top 3 Pizza Types by Revenue in Each Category

```sql
SELECT
    name,
    revenue
FROM (
    SELECT
        pt.category,
        pt.name,
        SUM(od.quantity * p.price) AS revenue,
        RANK() OVER (PARTITION BY pt.category ORDER BY SUM(od.quantity * p.price) DESC) AS rn
    FROM
        pizza_types pt
    JOIN pizzas p ON pt.pizza_type_id = p.pizza_type_id
    JOIN orders_details od ON od.pizza_id = p.pizza_id
    GROUP BY
        pt.category,
        pt.name
) AS ranked_pizzas
WHERE rn <= 3;
```

Conclusion

Key Insights:

Popular pizza types and sizes.
Revenue distribution and the contribution of different pizza categories.

Peak hours for orders and overall trends.

Recommendations:

- Focus on popular pizzas and sizes.
- Optimize inventory based on peak order times.
- Consider promotional strategies for high-revenue pizzas.


THANK
YOU!