SQL PIZZA SALES ANALYSIS

Objectives :

The objective of this SQL project is to analyze pizza sales data to extract meaningful business insights that can help optimize sales strategies, inventory management, and customer preferences. The project aims to: text

Objectives :

- **Analyze Order Trends** – Retrieve the total number of orders placed and determine order distribution by hour and date.
- **Revenue Insights** – Calculate total revenue, identify top-performing pizza types by revenue, and analyze cumulative revenue over time.
- **Product Performance Evaluation** – Identify the highest-priced pizza, most ordered pizza sizes, and top-selling pizza types.
- **Category-wise Analysis** – Determine the total quantity of pizzas ordered per category and analyze category-wise distribution.
- **Performance Metrics** – Calculate the percentage contribution of each pizza type to total revenue and assess average orders per day.
- **Data Integration** – Utilize SQL joins to combine relevant tables and extract comprehensive insights.



Questions :

1.Retrieve the total number of orders placed.
2.Calculate the total revenue generated from pizza sales.
3.Identify the highest-priced pizza.
4.Identify the most common pizza size ordered.
5.List the top 5 most ordered pizza types along with their quantities.

$6

6.Join the necessary tables to find the total quantity of each pizza category ordered.
7.Determine the distribution of orders by hour of the day.
8.Join relevant tables to find the category-wise distribution of pizzas.
9.Group the orders by date and calculate the average number of pizzas ordered per day.
10.Determine the top 3 most ordered pizza types based on revenue.

11.Calculate the percentage contribution of each pizza type to total revenue.
12.Analyze the cumulative revenue generated over time.
13.Determine the top 3 most ordered pizza types based on revenue for each pizza category.

1.Retrieve the total number of orders placed.

```
1    #Retrieve the total number of orders placed
2
3 •  select count(order_id) as total_orders from orders;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

	total_orders
▶	21350

2.Calculate the total revenue generated from pizza sales.

```sql
#Calculate the total revenue generated from 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
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

total_sales
817860.05

3.Identify the highest-priced pizza.

```sql
#Identify the highest-priced 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;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

name	price
The Greek Pizza	35.95

4.Identify the most common pizza size ordered.

```sql
#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;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

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

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

```sql
#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;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

name	quantity
The Classic Deluxe Pizza	2453
The Barbecue Chicken Pizza	2432
The Hawaiian Pizza	2422
The Pepperoni Pizza	2418
The Thai Chicken Pizza	2371

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

```sql
#Join the necessary tables to find the total quantity of each pizza category 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;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

category	quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

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

```sql
#Determine 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) ;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

hour	order_count
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642

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

```
1     #Join relevant tables to find the category-wise distribution of pizzas.
2
3 •   select pizza_types.category , count(pizza_types.name) from pizza_types
4     group by category
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

category	count(pizza_types.name)
Chicken	6
Classic	8
Supreme	9
Veggie	9

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

```sql
#Group the orders by date and calculate the average number of pizzas ordered per day.

SELECT
    ROUND(AVG(quantity), 0) AS avg_pizza_ordered_per_day
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 ;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

avg_pizza_ordered_per_day
138

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

```sql
#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 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 ;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5

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

```sql
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_percentage
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_percentage DESC;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

category	revenue_percentage
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

12.Analyze the cumulative revenue generated over time.

```sql
#Analyze the cumulative 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 ;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

order_date	cum_revenue
2015-01-01	2713.8500000000004
2015-01-02	5445.75
2015-01-03	8108.15
2015-01-04	9863.6
2015-01-05	11929.55
2015-01-06	14358.5
2015-01-07	16560.7

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

```sql
#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;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5

KEY INSIGHTS OF PIZZA_SALES :

1.Strong Sales Performance: A total of 21,350 orders generated $817,860 in sales.

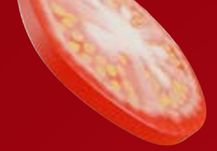
2.The average order value is approximately $38.31 ($817,860 ÷ 21,350).

3.Popular Pizza Trends: Greek Pizza is the most expensive at $35.95, indicating a premium offering. The most ordered pizza size is "L", showing customer preference for larger portions.

4.The top three most ordered pizzas are
1.Chicken Pizza (43,434),
2.Barbecue Pizza (42,768),
3.California Pizza (41,409),

5.Order Patterns & Peak Hours: The busiest hours are 12 PM (2,520 orders) and 11 AM (1,231 orders), indicating strong lunchtime demand. The average pizzas ordered per day is 138, showing a steady flow of sales.

6.Category & Distribution Insights: The Classic category dominates with 14,888 pizzas ordered. The pizza category count is relatively balanced: Chicken (6 types), Classic (8 types), Supreme (9 types), Veggie (9 types)—suggesting a diverse menu.

RECOMMENDATIONS:

- Promotions during peak hours (11 AM - 12 PM) to maximize revenue
- Upselling premium pizzas like Greek Pizza through combo deals.
- Stock management for classic pizzas, given their high demand.
- Exploring variations of Chicken, Barbecue, and California Pizzas, as they are the top choices.