PIZZAHUT SALES INSIGHTS

DESCRIBE TABLES

desc orders

Field	Type	Null	Key	Default	Extra
order_id	int	NO	PRI	NULL	
order_date	date	YES		NULL	
order_time	time	YES		NULL	

desc order_details

Field	Type	Null	Key	Default	Extra
order_details_id	int	NO	PRI	NULL	
order_id	int	YES	MUL	NULL	
pizza_id	varchar(25)	YES	MUL	NULL	
quantity	int	YES		NULL	

DESCRIBE TABLES

desc pizzas

Field	Type	Null	Key	Default	Extra
pizza_id	varchar(20)	NO	PRI	NULL	
pizza_type_id	varchar(20)	YES	MUL	NULL	
size	text	YES		NULL	
price	double	YES		NULL	

desc pizza_types

Field	Type	Null	Key	Default	Extra
pizza_type_id	varchar(20)	NO	PRI	NULL	
name	text	YES		NULL	
category	text	YES		NULL	
ingredients	text	YES		NULL	

Retrieve the total number of orders placed.

```sql
SELECT
    COUNT(order_id) AS 'Total orders'
FROM
    orders
```

Calculate the total revenue generated from pizza sales.

```sql
SELECT
    ROUND(SUM(o.quantity * p.price), 2) AS 'Total revenue generated'
FROM
    pizzas AS p
        RIGHT JOIN
    order_details AS o ON o.pizza_id = p.pizza_id;
```

Identify the highest-priced pizza.

```sql
SELECT
    pizza_types.name, pizzas.price
FROM
    pizza_types
        INNER JOIN
    pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
WHERE
    pizzas.price IN (SELECT
            MAX(price)
        FROM
            pizzas);
```

Identify the most common pizza size ordered.

```sql
SELECT
    p.size, COUNT(o.order_details_id) AS 'Total orders'
FROM
    order_details AS o
        INNER JOIN
    pizzas AS p ON p.pizza_id = o.pizza_id
GROUP BY p.size
ORDER BY 'Total orders' DESC
LIMIT 1;
```

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

```sql
SELECT
    pt.name, o.quantity
FROM
    order_details AS o
        INNER JOIN
    pizzas AS p ON p.pizza_id = o.pizza_id
        INNER JOIN
    pizza_types AS pt ON pt.pizza_type_id = p.pizza_type_id
GROUP BY p.pizza_type_id
ORDER BY COUNT(o.order_id) DESC
LIMIT 5;
```

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

```
SELECT
    pt.category, SUM(o.quantity) AS Quantityy
FROM
    order_details AS o
        INNER JOIN
    pizzas AS p ON p.pizza_id = o.pizza_id
        INNER JOIN
    pizza_types AS pt ON pt.pizza_type_id = p.pizza_type_id
GROUP BY pt.category order by Quantityy desc;
```

Determine the distribution of orders by hour of the day.

```sql
SELECT
    HOUR(order_time) AS Hourr,
    (COUNT(order_id) / (SELECT
            COUNT(*)
        FROM
            orders))*100 AS distribution
FROM
    orders
GROUP BY Hourr
ORDER BY distribution DESC;
```

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

```sql
SELECT
    category,
    (COUNT(pizza_type_id) / (SELECT
            COUNT(*)
        FROM
            pizza_types)) * 100 AS Distribution
FROM
    pizza_types
GROUP BY category
ORDER BY Distribution DESC;
```

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

```sql
SELECT
    ROUND(AVG(Total_Orders_per_day), 0) AS Average_Orders
FROM
    (SELECT
        orders.order_date,
            SUM(order_details.quantity) AS Total_Orders_per_day
    FROM
        orders
    INNER JOIN order_details ON orders.order_id = order_details.order_id
    GROUP BY orders.order_date) AS Order_per_day;
```

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

```sql
SELECT
    pt.name, SUM(p.price * o.quantity) AS revenue
FROM
    pizza_types AS pt
        INNER JOIN
    pizzas AS p ON p.pizza_type_id = pt.pizza_type_id
        INNER JOIN
    order_details AS o ON o.pizza_id = p.pizza_id
GROUP BY pt.name
ORDER BY revenue DESC
LIMIT 3;
```

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

```sql
SELECT
    pt.category,
    ROUND((SUM(p.price * o.quantity) / (SELECT
                    SUM(o.quantity * p.price) AS 'Total revenue generated'
                FROM
                    pizzas AS p
                        RIGHT JOIN
                    order_details AS o ON o.pizza_id = p.pizza_id)) * 100,
            2) AS revenue
FROM
    pizza_types AS pt
        INNER JOIN
    pizzas AS p ON p.pizza_type_id = pt.pizza_type_id
        INNER JOIN
    order_details AS o ON o.pizza_id = p.pizza_id
GROUP BY pt.category
ORDER BY revenue DESC;
```

Analyze the cumulative revenue generated over time.

```
select
order_date,sum(revenue) over(order by order_date) as cum_sum
from
(select o.order_date,sum(p.price*od.quantity) as revenue
from
orders as o inner join order_details as od on o.order_id=od.order_id
inner join pizzas as p on p.pizza_id=od.pizza_id
group by order_date) as a;
```

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

```sql
select Category,Name,Revenue
from
(select *,rank() over(partition by category order by revenue desc) as Ranking
from
(select pt.name,sum(p.price*o.quantity) as Revenue,pt.category
from pizza_types as pt inner join pizzas as p on p.pizza_type_id=pt.pizza_type_id
inner join order_details as o on o.pizza_id=p.pizza_id
group by pt.category,pt.name)
as a )
as b
where ranking<=3;
```

THANK YOU