SQL PROJECT ON PIZZA SALES

Hello my name is supriya tonde in this project I utilize SQL queries that related to pizza sales

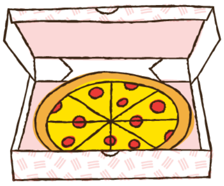
I HAVE USED FOUR TABLES HERE

1.ORDERS

2.ORDER_DETAILS

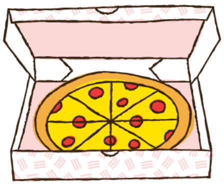
3.PIZZAS

4.PIZZA_TYPES



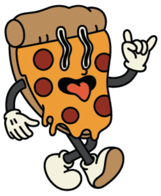
1.retrive the total number of order placed.

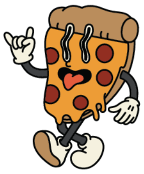
```
select count(order_id) as total_Orders
from orders;
```



2.calculate the total revenue generated from pizza sales.

```
select sum(p.price * o.quantity)  as
total_revenue
from pizzas p join order_details o
on p.pizza_id = o.pizza_id;
```





3.identify heighest pizza price.

```
select pizza_types.name,max(pizzas.price) as
heighest_pizzaprice
from pizza_types join pizzas
group by pizza_types.name
order by heighest_pizzaprice desc limit 1;
```

4.identify most common size pizza orderd.

```
select
pizzas.size,count(order_details.order_details_id)as
most_commonorder
from pizzas join order_details
on pizzas.pizza_id = order_details.pizza_id
group by pizzas.size
order by most_commonorder desc ;
```

5. group the orders by the date calculate the avrage number of pizzas ordered per day.

```
SELECT
ROUND(AVG(quantity), 0) as avrage_pizza_perday
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;
```

6. join the necessary tables to find the total qunatity of each pizza category ordered .

```
select pizza_types.category,
sum(order_details.quantity) as total_quantity
from  pizza_types  join pizzas
on pizza_types.pizza_type_id =pizzas.pizza_type_id
join order_details
on pizzas.pizza_id = order_details.pizza_id
group by pizza_types.category
order by total_quantity desc;
```

7.distribution of order details by the hour of the day.

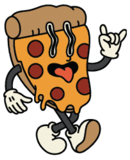
```
select hour(order_time) as hour,
count(order_id)as order_count
from orders
group by hour(order_time);
```

8. Determine the most ordered top 3 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) 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;
```

9.Analyze the cumulative revenue genrated over time.

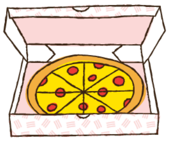
```
select order_date,sum(revenue) over(order by order_date)as
cum_revenue
from
(select orders.order_date,
sum(order_details.quantity*pizzas.price)as revenue
from orders join order_details
on orders.order_id =order_details.order_id
join pizzas
on pizzas.pizza_id=order_details.pizza_id
group by orders.order_date)as sales
```



10.CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPES TO TOTAL REVENUE.

```
select pizza_types.category,
sum(pizzas.price*order_details.quantity)as total_revenue
from pizza_types join pizzas
on  pizza_types.pizza_type_id =pizzas.pizza_type_id
join order_details
on pizzas.pizza_id = order_details.pizza_id
group by pizza_types.category
order by total_revenue desc ;
```

```
# 11. determine the top 3 most orderd pizza types based on
revenue
select round(sum(pizzas.price*order_details.quantity),0)as
total_revenue
from pizzas join order_details
on pizzas.pizza_id=order_details.pizza_id
join pizza_types
on pizzas.pizza_type_id=pizza_types.pizza_type_id
group by pizza_types.name
order by total_revenue desc limit 3;
```



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

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

THANK YOU