SQL QUERY RESULTS ANALYSIS SNIPPETS FOR PIZZA SALES

THE QUESTIONS

Basic:

Retrieve the total number of orders placed.

Calculate the total revenue generated from pizza sales.

Identify the highest-priced pizza.

Identify the most common pizza size ordered.

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

Intermediate:

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

Determine the distribution of orders by hour of the day.

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

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

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

Advanced:

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

Analyze the cumulative revenue generated over time.

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

CODE SNIPPETS

```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 price desc limit 1;
```

	name text	price numeric
1	The Greek Pizza	35.95

```sql
--identify the most common pizza size sold
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 "char"	order_count bigint
1	L	18526

```sql
-- Determine the distribution of orders by hour of the day.
select
    EXTRACT(HOUR FROM time) as hour_of_day,
    count(order_id)
from orders
group by hour_of_day
order by hour_of_day;
```

	hour_of_day numeric	count bigint
1	9	1
2	10	8
3	11	1231
4	12	2520
5	13	2455
6	14	1472
7	15	1468
8	16	1920
9	17	2336
10	18	2399
11	19	2009
12	20	1642
13	21	1198
14	22	663

```sql
/*list the top 5 most ordered pizza
types along with their quantities*/
select
    pizza_types.name,
    sum(order_details.quantity) as quantity_ordered
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_ordered desc limit 5;
```

	name text	quantity_ordered bigint
1	The Classic Deluxe Pizza	2453
2	The Barbecue Chicken Pizza	2432
3	The Hawaiian Pizza	2422
4	The Pepperoni Pizza	2418
5	The Thai Chicken Pizza	2371

```sql
/*Group the orders by date and calculate
the average number of pizzas ordered per day.*/
select
    round(avg(quantity),0) as order_per_day
from(
    select
        orders.date as order_date,
        sum(order_details.quantity) as quantity
    from orders
    join order_details
    on orders.order_id=order_details.order_id
    group by order_date
) as quantity
```

	order_per_day numeric
1	138

```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 pizza_types.pizza_type_id=pizzas.pizza_type_id
join order_details
on pizzas.pizza_id=order_details.pizza_id
group by pizza_types.name
order by revenue desc limit 3;
```

	name text	revenue numeric
1	The Thai Chicken Pizza	43434.25
2	The Barbecue Chicken Pizza	42768.00
3	The California Chicken Pizza	41409.50

```sql
--Calculate the percentage contribution of each pizza type to total revenue
with total_sales_cte as(
select
    round(sum(order_details.quantity * pizzas.price),2) as total_sales
    from order_details
    join pizzas
    on order_details.pizza_id=pizzas.pizza_id
 )

select
    pizza_types.category,
    round(
    (sum(order_details.quantity * pizzas.price) / total_sales_cte.total_sales)*100,2)
    as revenue_percent
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,
total_sales_cte
group by pizza_types.category, total_sales_cte.total_sales
order by revenue_percent desc;
```

	category text	revenue_percent numeric
1	Classic	26.91
2	Supreme	25.46
3	Chicken	23.96
4	Veggie	23.68

```sql
-- Analyze the cumulative revenue generated over time.
select date,
sum(revenue) over(order by date ) as cum_revenue
from
(select
    orders.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 order_details.pizza_id=pizzas.pizza_id
group by orders.date
order by orders.date ) as sales;
```

date date	cum_revenue numeric
2015-01-01	2713.85
2015-01-02	5445.75
2015-01-03	8108.15
2015-01-04	9863.60
2015-01-05	11929.55
2015-01-06	14358.50
2015-01-07	16560.70
2015-01-08	19399.05
2015-01-09	21526.40
2015-01-10	23990.35
2015-01-11	25862.65
2015-01-12	27781.70
2015-01-13	29831.30
2015-01-14	32358.70
2015-01-15	34343.50

```sql
/*Determine the top 3 most ordered pizza types
based on revenue for each pizza category.*/
select category, name, revenue from
(select category,name,revenue,
rank() over(partition by category order by revenue desc) as ranks
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 pizzas.pizza_id=order_details.pizza_id
group by pizza_types.category, pizza_types.name) as a) as b
where ranks<=3
```

	category text	name text	revenue numeric
1	Chicken	The Thai Chicken Pizza	43434.25
2	Chicken	The Barbecue Chicken Pizza	42768.00
3	Chicken	The California Chicken Pizza	41409.50
4	Classic	The Classic Deluxe Pizza	38180.5
5	Classic	The Hawaiian Pizza	32273.25
6	Classic	The Pepperoni Pizza	30161.75
7	Supreme	The Spicy Italian Pizza	34831.25
8	Supreme	The Italian Supreme Pizza	33476.75
9	Supreme	The Sicilian Pizza	30940.50
10	Veggie	The Four Cheese Pizza	32265.70
11	Veggie	The Mexicana Pizza	26780.75
12	Veggie	The Five Cheese Pizza	26066.5