HOT & SPICY

PIZZA SALES ANALYSIS USING SQL

@FORAM PANDYA

HELLO !

MY NAME IS FORAM, AND I'M UTILIZING SQL TO SOLVE SQL QUERIES FOR PIZZA SALES ANALYSIS IN THIS PROJECT.

PROJECT’S AIM

THE GOAL OF THIS SQL PROJECT ON PIZZA SALES ANALYSIS IS TO APPLY DATA ANALYSIS TECHNIQUES TO EXTRACT VALUABLE INSIGHTS FROM THE DATABASE. THIS ENABLES STAKEHOLDERS TO MAKE INFORMED BUSINESS DECISIONS AIMED AT FOSTERING THE COMPANY'S GROWTH.

RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED

```sql
SELECT
    COUNT(quantity) AS total_orders
FROM
    order_details
```

	total_orders
▶	48620

CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES

```sql
SELECT
    ROUND(SUM(od.quantity * p.price), 2) AS total_sales
FROM
    order_details AS od
        JOIN
    pizzas AS p ON od.pizza_id = p.pizza_id
```

	total_sales
▶	817860.05

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED

```sql
SELECT
    p.size, COUNT(od.quantity) AS total_qty_sold
FROM
    order_details AS od
        Join pizzas AS p USING (pizza_id)
GROUP BY p.size
ORDER BY total_qty_sold DESC LIMIT 1
```

size	total_qty_sold
L	18526

LIST THE TOP 5 MOST ORDERED PIZZA TYPES ALONG WITH THEIR QUANTITIES

```sql
select
    pt.name,
    sum(od.quantity) as total_qty_sold
from order_details as od
join pizzas as p using(pizza_id)
join pizza_types as pt on p.pizza_type_id=pt.pizza_type_id
group by pt.name
order by total_qty_sold desc limit 5
```

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

JOIN THE NECESSARY TABLES TO FIND THE TOTAL QUANTITY OF EACH PIZZA CATEGORY ORDERED

```sql
select
    pt.category,
    sum(od.quantity) as quantity
from pizza_types as pt
join pizzas as p using (pizza_type_id)
join order_details as od on p.pizza_id=od.pizza_id
group by pt.category order by quantity desc
```

category	quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY

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

hour	order_count
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642
21	1198
22	663
23	28
10	8
9	1

JOIN RELEVANT TABLES TO FIND THE CATEGORY-WISE DISTRIBUTION OF PIZZAS

```sql
select
    category,
    count(pizza_type_id) as pizza
from pizza_types
group by category
```

category	pizza
Chicken	6
Classic	8
Supreme	9
Veggie	9

DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE

```
select
    pt.name,
    round(sum(od.quantity*p.price),0)as revenue
from order_details as od
join pizzas as p using(pizza_id)
join pizza_types as pt on p.pizza_type_id=pt.pizza_type_id
group by pt.name
order by revenue desc
limit 3
```

name	revenue
The Thai Chicken Pizza	43434
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41410

CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPE TO TOTAL REVENUE

```
select
    pt.category,
   round(sum(od.quantity*p.price)/
        (select round(sum(quantity*price),2) as total_sales
        from order_details
        join pizzas using (pizza_id))*100,2)as revenue
from pizza_types as pt
join pizzas as p using(pizza_type_id)
join order_details as od on od.pizza_id=p.pizza_id
group by pt.category
order by revenue desc
```

category	revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME

```sql
select order_date,
    sum(revenue) over(order by order_date) as cum_revenue
from
(select
    o.order_date,
    round(sum(od.quantity*p.price),2) as revenue
from order_details as od
join pizzas as p using (pizza_id)
join orders as o using(order_id)
group by o.order_Date) as sales;
```

order_date	cum_revenue
2015-01-01	2713.85
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
2015-01-08	19399.05

DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE FOR EACH PIZZA CATEGORY.

```sql
select
    pt.name,
    pt.category,
     sum(od.quantity*p.price) as revenue
from order_details as od
join pizzas as p using(pizza_id)
join pizza_types as pt on p.pizza_type_id =pt.pizza_type_id
group by pt.category,pt.name
order by revenue desc limit 3
```

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

PIZZA SALES REPORT

THANK YOU !

PANDYA.FORAM05@GMAIL.COM