

SQL DATA ANALYSIS ON PIZZA SALES



INTRODUCTION OF DATA SET

The dataset is a CSV file detailing pizza sales, including information on pizza types, quantities sold, prices, and sale dates. It provides a comprehensive view of sales trends, aiding in the analysis of customer preferences and revenue generation.



RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.

```
-- Retrieve the total number of orders placed.  
select count(order_id) as Total_Number_of_Order from orders;
```

RESULT

Result Grid		Filter Row
	Total_Number_of_Order	
▶	21350	

CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

```
-- Calculate the total revenue generated from pizza sales.  
SELECT  
    round(SUM(order_details.quantity * pizzas.price)) AS Total_Revenue  
FROM  
    order_details  
    JOIN  
    pizzas ON order_details.pizza_id = pizzas.pizza_id;
```



RESULT

Result Grid	
	Total_Revenue
▶	817860

IDENTIFY THE HIGHEST-PRICED PIZZA.

```
SELECT
    pizza_types.name, price
FROM
    pizzas
    JOIN
    pizza_types ON pizza_types.pizza_type_id = pizzas.pizza_type_id
ORDER BY price DESC
LIMIT 1;
```

RESULT

Result Grid   Filter Rows		
	name	price
▶	The Greek Pizza	35.95

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.

```
SELECT
    size, COUNT(order_details.order_details_id) AS total_orders
FROM
    order_details
    JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id
GROUP BY pizzas.size
ORDER BY total_orders DESC;
```

RESULT

Result Grid			Filter
	size	total_orders	
	L	18526	
	M	15385	
	S	14137	
	XL	544	
	XXL	28	

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

```
SELECT
    pizza_types.name,
    COUNT(order_details_id) AS total_quantities
FROM
    order_details
    JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id
    JOIN
    pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
GROUP BY pizza_types.name
ORDER BY total_quantities DESC
LIMIT 5;
```

RESULT

Result Grid			Filter Rows:
	name	total_quantities	
	The Classic Deluxe Pizza	2416	
	The Barbecue Chicken Pizza	2372	
	The Hawaiian Pizza	2370	
	The Pepperoni Pizza	2369	
	The Thai Chicken Pizza	2315	

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

```
SELECT
    pizza_types.category,
    COUNT(order_details.quantity) AS Total_Quantity
FROM
    order_details
    JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id
    JOIN
    pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
GROUP BY pizza_types.category;
```

RESULT

category	Total_Quantity
Classic	14579
Veggie	11449
Supreme	11777
Chicken	10815

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.

```
SELECT
    COUNT(order_details.quantity) AS total_Quantity,
    HOUR(order_time)
FROM
    orders
    JOIN
    order_details ON orders.order_id = order_details.order_id
GROUP BY HOUR(order_time)
ORDER BY total_Quantity DESC;
```

RESULT

total_Quantity	HOUR(order_time)
6543	12
6203	13
5359	18
5143	17
4350	19
4185	16
3521	14
3487	20
3170	15
2672	11

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

```
SELECT
    COUNT(order_details.order_id) AS Total_Orders,
    pizza_types.category
FROM
    order_details
    JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id
    JOIN
    pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
GROUP BY pizza_types.category
ORDER BY Total_Orders DESC;
```

RESULT

Total_Orders	category
14579	Classic
11777	Supreme
11449	Veggie
10815	Chicken

GROUP THE ORDERS BY DATE AND CALCULATE THE AVERAGE NUMBER OF PIZZAS ORDERED PER DAY.

```
SELECT
    ROUND(AVG(quan), 0) AS Avg_per_day
FROM
    (SELECT
        SUM(quantity) AS quan, order_date AS per_day
    FROM
        order_details
    JOIN orders ON order_details.order_id = orders.order_id
    GROUP BY order_date) AS order_quan;
```

RESULT

Avg_per_day
138

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

```
SELECT
    pizza_types.name as Name,
    SUM(pizzas.price * order_details.quantity) AS Revenue
FROM
    order_details
    JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id
    JOIN
    pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
GROUP BY pizza_types.name
ORDER BY Revenue DESC
LIMIT 3;
```

RESULT

Name	Revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5

CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPE TO TOTAL REVENUE.

```
SELECT
    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 order_details.pizza_id = pizzas.pizza_id)) * 100,
        2) AS Total_Revenue,
    pizza_types.category as Category
FROM
    order_details
    JOIN
        pizzas ON order_details.pizza_id = pizzas.pizza_id
    JOIN
        pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
GROUP BY pizza_types.category;
```

RESULT

Total_Revenue	Category
26.91	Classic
23.68	Veggie
25.46	Supreme
23.96	Chicken

ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME.

```
select Order_Date,sum(Total_revenue)over(order by Order_Date) as Cumulative_revenue
from(select round(sum(order_details.quantity*pizzas.price)) as Total_revenue,orders.order_date as Order_Date
from order_details
join pizzas on order_details.pizza_id=pizzas.pizza_id
join orders on order_details.order_id=orders.order_id group by Order_Date) as sales;
```

RESULT

Order_Date	Cumulative_revenue
2015-01-01	2714
2015-01-02	5446
2015-01-03	8108
2015-01-04	9863
2015-01-05	11929
2015-01-06	14358
2015-01-07	16560
2015-01-08	19398
2015-01-09	21525
2015-01-10	23989
2015-01-11	25861
2015-01-12	27780
2015-01-13	29830
2015-01-14	32357
2015-01-15	34342

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

```
select revenue,name,category from
(select category,revenue,name,
rank() over(partition by category order by revenue desc ) as ran from
(select sum(order_details.quantity*pizzas.price) as revenue, pizza_types.category,pizza_types.name from order_details
join pizzas on order_details.pizza_id=pizzas.pizza_id
join pizza_types on pizzas.pizza_type_id=pizza_types.pizza_type_id group by category,name) as a ) as b where ran<=3;
```

RESULT

revenue	name	category
43434.25	The Thai Chicken Pizza	Chicken
42768	The Barbecue Chicken Pizza	Chicken
41409.5	The California Chicken Pizza	Chicken
38180.5	The Classic Deluxe Pizza	Classic
32273.25	The Hawaiian Pizza	Classic
30161.75	The Pepperoni Pizza	Classic
34831.25	The Spicy Italian Pizza	Supreme
33476.75	The Italian Supreme Pizza	Supreme
30940.5	The Sicilian Pizza	Supreme
32265.700000000065	The Four Cheese Pizza	Veggie
26780.75	The Mexicana Pizza	Veggie
26066.5	The Five Cheese Pizza	Veggie

THANK YOU!

