

MY PROJECT ON SQL DATA QUERY

Insights from PizzaHut's Data!

I recently completed a project analyzing 40,000+ orders in a multi-table PizzaHut dataset to sharpen my SQL skills.
Here's a snapshot of what I achieved:



RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.

```
SELECT  
    COUNT(order_id) AS total_orders  
FROM  
    order_details;
```

	total_orders
▶	48620

CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

```
SELECT  
    SUM(order_details.quantity * pizzas.price) AS total_sales  
FROM  
    order_details  
    JOIN  
    pizzas ON pizzas.pizza_id = order_details.pizza_id;
```

	total_sales
▶	817860.0499999993

IDENTIFY THE HIGHEST-PRICED PIZZA

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

	name	price
▶	The Greek Pizza	35.95

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED

```
SELECT
    COUNT(order_details.order_id) AS most_ordered_pizzas,
    pizzas.size
FROM
    order_details
    JOIN
    pizzas ON order_details.pizza_id = pizzas.pizza_id
GROUP BY size
ORDER BY most_ordered_pizzas DESC
LIMIT 1;
```

	most_ordered_pizzas	size
▶	18526	L

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

```
SELECT
    pizza_types.name, SUM(order_details.quantity) AS 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.name
ORDER BY Quantity DESC
LIMIT 5;
```

name	Quantity
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

```
SELECT
    pizza_types.category, COUNT(order_details.order_id) AS count_
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 count_ DESC
```

category	count_
Classic	14579
Supreme	11777
Veggie	11449
Chicken	10815

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY

```
SELECT
    HOUR(orders.order_time) as `Time`,
    COUNT(orders.order_id) as Count
FROM
    orders group by `Time` order by `Time`;
```

Time	Count
9	1
10	8
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

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

```
SELECT
    pizza_types.category AS count,
    COUNT(pizza_types.name) total_pizzas_of_the_category
FROM
    pizza_types
GROUP BY count;
```

count	total_pizzas_of_the_category
Chicken	6
Classic	8
Supreme	9
Veggie	9

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

```
SELECT
    orders.order_date,
    SUM(order_details.quantity) AS average_quantity
FROM
    order_details
    JOIN
    orders ON order_details.order_id = orders.order_id
GROUP BY orders.order_date;
```

order_date	average_quantity
2015-01-01	162
2015-01-02	165
2015-01-03	158
2015-01-04	106
2015-01-05	125
2015-01-06	147
2015-01-07	138
2015-01-08	173
2015-01-09	127
2015-01-10	146
2015-01-11	116
2015-01-12	119
2015-01-13	120
2015-01-14	150
2015-01-15	123
2015-01-16	158
2015-01-17	125
2015-01-18	122

DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE

```
SELECT
    pizza_types.name,
    round(SUM(pizzas.price * order_details.quantity),2) as 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 Revenue desc limit 3;
```

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
    pizza_types.category,
    ROUND((SUM(pizzas.price * order_details.quantity) / (SELECT
        SUM(pizzas.price * order_details.quantity)
        FROM
            order_details
            JOIN
                pizzas ON order_details.pizza_id = pizzas.pizza_id)),
        4) * 100 AS revenue_percent
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 pizza_types.category;
```

category	revenue_percent
Chicken	23.96
Classic	26.91
Supreme	25.46
Veggie	23.68

ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME

```
select order_date, rev_per_day, round( sum(rev_per_day)
over(order by order_date),4) as cumulative_revenue from
(SELECT
    orders.order_date,
    ROUND(SUM(pizzas.price * order_details.quantity),
        4) AS rev_per_day
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;
```

order_date	rev_per_day	cumulative_revenue
2015-01-01	2713.85	2713.85
2015-01-02	2731.9	5445.75
2015-01-03	2662.4	8108.15
2015-01-04	1755.45	9863.6
2015-01-05	2065.95	11929.55
2015-01-06	2428.95	14358.5
2015-01-07	2202.2	16560.7
2015-01-08	2838.35	19399.05
2015-01-09	2127.35	21526.4
2015-01-10	2463.95	23990.35
2015-01-11	1872.3	25862.65
2015-01-12	1919.05	27781.7
2015-01-13	2049.6	29831.3
2015-01-14	2527.4	32358.7
2015-01-15	1984.8	34343.5
2015-01-16	2594.15	36937.65
2015-01-17	2064.1	39001.75
2015-01-18	1976.85	40978.6
2015-01-19	2387.15	43365.75
2015-01-20	2397.9	45763.65
2015-01-21	2040.55	47804.2
2015-01-22	2496.7	50300.9
2015-01-23	2423.7	52724.6
2015-01-24	2289.25	55013.85

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

```
select name,category, Revenue,Rank_ from (
select name,category,Revenue, rank()
over(partition by category order by Revenue desc) as Rank_ from
(SELECT
    pizza_types.name, pizza_types.category,
    round(SUM(pizzas.price * order_details.quantity),2) as 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,pizza_types.category order by Revenue )as rev)
as b where Rank_ <=3;
```

name	category	Revenue	Rank_
The Thai Chicken Pizza	Chicken	43434.25	1
The Barbecue Chicken Pizza	Chicken	42768	2
The California Chicken Pizza	Chicken	41409.5	3
The Classic Deluxe Pizza	Classic	38180.5	1
The Hawaiian Pizza	Classic	32273.25	2
The Pepperoni Pizza	Classic	30161.75	3
The Spicy Italian Pizza	Supreme	34831.25	1
The Italian Supreme Pizza	Supreme	33476.75	2
The Sicilian Pizza	Supreme	30940.5	3
The Four Cheese Pizza	Veggie	32265.7	1
The Mexicana Pizza	Veggie	26780.75	2
The Five Cheese Pizza	Veggie	26066.5	3

THANK YOU!

