



SQL PROJECT ON

PIZZA HUT SALES REPORT





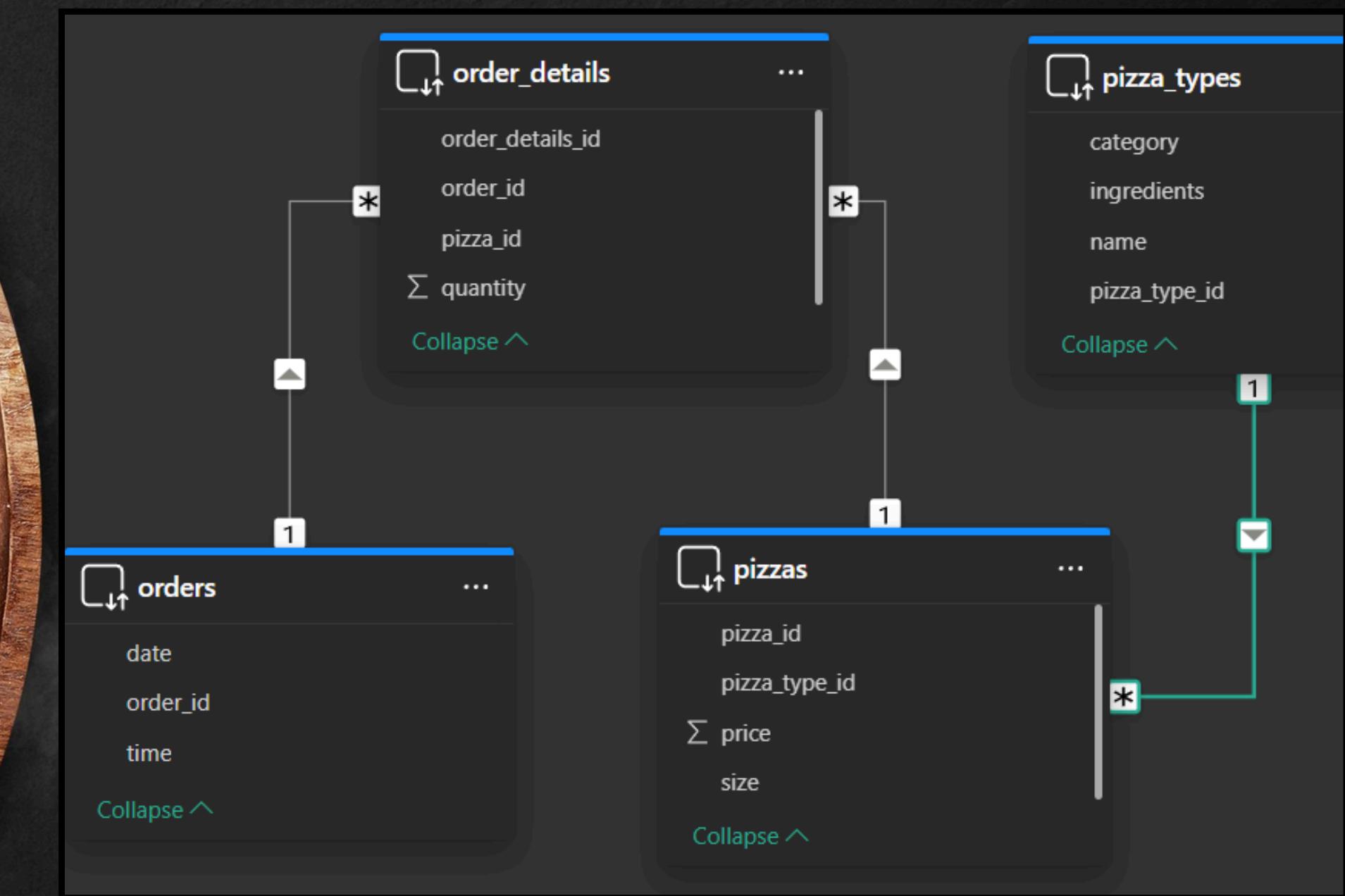
LARANA PIZZA

HELLO,

my name is Vishal, and this project showcases my work using SQL to analyze Pizza Hut sales data. The focus is on extracting insights into revenue, identifying top-selling pizza types, and understanding category performance. This analysis demonstrates the power of SQL in uncovering trends and supporting data-driven decision-making.



DATA MODEL



RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED

```
- select count(order_id) as Total_Orders  
| from order_details
```

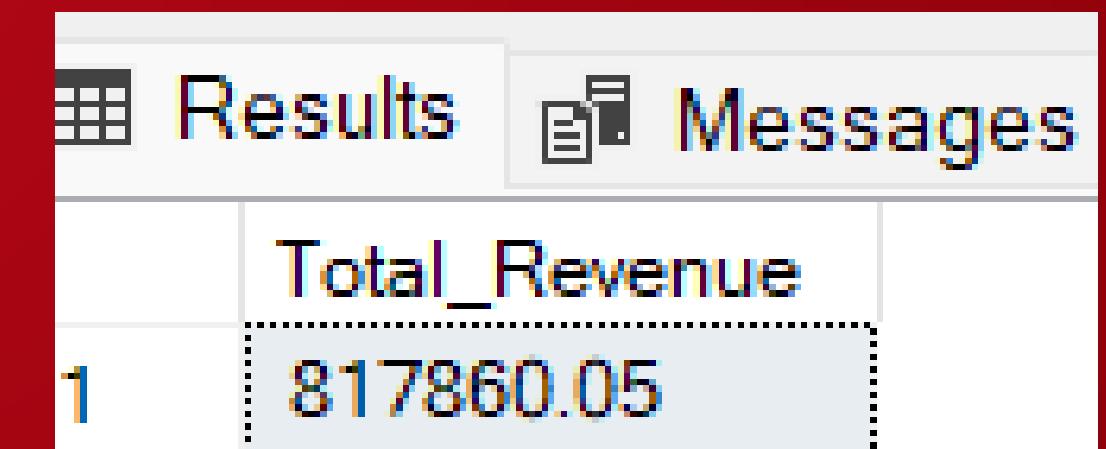


The screenshot shows a database interface with two tabs: 'Results' and 'Messages'. The 'Results' tab is selected and displays a single row of data in a table format. The table has one column labeled 'Total_Orders' with the value '48620'.

	Total_Orders
1	48620

CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES

```
- select round(sum(od.quantity * pz.price),2) as Total_Revenue  
from order_details od  
inner join pizzas pz  
on od.pizza_id = pz.pizza_id;
```



The screenshot shows a database query results window with two tabs: "Results" and "Messages". The "Results" tab is selected, displaying a single row of data. The column header is "Total_Revenue" and the value is "817860.05". The "Messages" tab is also visible.

	Total_Revenue
1	817860.05

IDENTIFY THE HIGHEST-PRICED PIZZA

```
select top 1 pt.name as highest_priced_pizza_name, round(pz.price,2) as Price  
from pizza_types pt  
inner join pizzas pz  
on pt.pizza_type_id = pz.pizza_type_id  
order by pz.price desc
```

Results Messages

	highest_priced_pizza_name	Price
1	The Greek Pizza	35.95

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED

```
select top 1 pz.size as Most_common_size_ordered , count(od.order_id) as Order_count
from order_details od
inner join pizzas pz
on od.pizza_id = pz.pizza_id
group by pz.size
order by Order_count desc
```

Results Messages

	Most_common_size_ordered	Order_count
1	L	18526

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

```
select Top 5 pt.name as Pizza_Name ,sum(od.quantity) as Quantity  
from pizza_types pt  
join pizzas pz  
on pt.pizza_type_id = pz.pizza_type_id  
join order_details od  
on pz.pizza_id =od.pizza_id  
  
group by pt.name  
order by Quantity Desc
```

	Pizza_Name	Quantity
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

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

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

	category	Total_quantity
1	Classic	14888
2	Supreme	11987
3	Veggie	11649
4	Chicken	11050

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY

```
- select DATEPART(hour, time) as order_hour, COUNT(order_id) as Orders  
  from orders  
 group by DATEPART(hour, time)  
 order by order_hour
```

	order_hour	Orders
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
15	23	28

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

```
with daily_orders_quantity as(  
    select o.date, sum(od.quantity) as No_of_orders  
    from orders o  
    join order_details od  
    on o.order_id=od.order_id  
    group by date)  
  
select cast(round(avg(No_of_orders*1.0),2) as decimal(10,2)) as Avg_Pizza_ordered_per_day  
from daily_orders_quantity
```

Results Messages

	Avg_Pizza_ordered_per_day
1	138.47

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

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

Results Messages

	category	count
1	Chicken	6
2	Classic	8
3	Supreme	9
4	Veggie	9

DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE

```
select top 3 pt.name, round(sum(pz.price*od.quantity),2) as Revenue_per_pizza  
from pizza_types pt  
join pizzas pz  
on pt.pizza_type_id = pz.pizza_type_id  
join order_details od  
on pz.pizza_id = od.pizza_id  
group by pt.name  
order by Revenue_per_pizza desc
```

Results Messages

	name	Revenue_per_pizza
1	The Thai Chicken Pizza	43434.25
2	The Barbecue Chicken Pizza	42768
3	The California Chicken Pizza	41409.5

CALCULATE THE PERCENTAGE CONTRIBUTION OF EACH PIZZA TYPE TO TOTAL REVENUE

```
-with Revenue_per_pizza as
(select pt.category , Round(sum(pz.price*od.quantity),0) as Total_revenue
from pizza_types pt
join pizzas pz
on pt.pizza_type_id = pz.pizza_type_id
join order_details od
on pz.pizza_id = od.pizza_id
group by pt.category)

select category, Round((Total_revenue * 100/sum(total_revenue) over()),2) as Percentage_contirbution
from Revenue_per_pizza
order by Percentage_contirbution desc
```

Results Messages

	category	Percentage_contirbution
1	Classic	26.91
2	Supreme	25.46
3	Chicken	23.96
4	Veggie	23.68

ANALYZE THE CUMULATIVE REVENUE GENERATED OVER TIME

```

with Revenue_per_day as (select o.date , Round(sum(od.quantity*pz.price),2) as Revenue
from orders o
join order_details od
on o.order_id = od.order_id
join pizzas pz
on od.pizza_id = pz.pizza_id
group by o.date
)
select date,Revenue,SUM(Revenue) over(order by date) as cumulative_Revenue
from
Revenue_per_day
order by date|

```

	date	Revenue	cumulative_Revenue
1	2015-01-01	2713.85	2713.85
2	2015-01-02	2731.9	5445.75
3	2015-01-03	2662.4	8108.15
4	2015-01-04	1755.45	9863.6
5	2015-01-05	2065.95	11929.55
6	2015-01-06	2428.95	14358.5
7	2015-01-07	2202.2	16560.7
8	2015-01-08	2838.35	19399.05
9	2015-01-09	2127.35	21526.4
10	2015-01-10	2463.95	23990.35
11	2015-01-11	1872.3	25862.65
12	2015-01-12	1919.05	27781.7
13	2015-01-13	2049.6	29831.3
14	2015-01-14	2527.4	32358.7
15	2015-01-15	1984.8	34343.5
16	2015-01-16	2594.15	36937.65
17	2015-01-17	2064.1	39001.75
18	2015-01-18	1976.85	40978.6
19	2015-01-19	2387.15	43365.75
20	2015-01-20	2397.9	45763.65
21	2015-01-21	2040.55	47804.2
22	2015-01-22	2496.7	50300.9
23	2015-01-23	2423.7	52724.6
24	2015-01-24	2289.25	55013.85
25	2015-01-25	1617.55	56631.4
26	2015-01-26	1884.4	58515.8
27	2015-01-27	2528.05	61043.85
28	2015-01-28	2016	63059.85
29	2015-01-29	2045.3	65105.15
30	2015-01-30	2270.3	67375.45
31	2015-01-31	2417.85	69793.3
32	2015-02-01	3189.2	72982.5

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

```

select name,category, Revenue
from
    (Select name,category,round(sum(price*quantity),2) as Revenue,
    ROW_NUMBER() OVER (PARTITION BY category ORDER BY SUM(price * quantity) DESC) AS Rank
    from pizza_types pt
    join pizzas pz
    on pt.pizza_type_id =pz.pizza_type_id
    join order_details od
    on pz.pizza_id =od.pizza_id
    group by name, category
) as Revenue_per_category
where rank <=3
order by category, Revenue desc

```

Results Messages

	name	category	Revenue
1	The Thai Chicken Pizza	Chicken	43434.25
2	The Barbecue Chicken Pizza	Chicken	42768
3	The California Chicken Pizza	Chicken	41409.5
4	The Classic Deluxe Pizza	Classic	38180.5
5	The Hawaiian Pizza	Classic	32273.25
6	The Pepperoni Pizza	Classic	30161.75
7	The Spicy Italian Pizza	Supreme	34831.25
8	The Italian Supreme Pizza	Supreme	33476.75
9	The Sicilian Pizza	Supreme	30940.5
10	The Four Cheese Pizza	Veggie	32265.7
11	The Mexicana Pizza	Veggie	26780.75
12	The Five Cheese Pizza	Veggie	26066.5



PIZZA HUT

THANK YOU!

