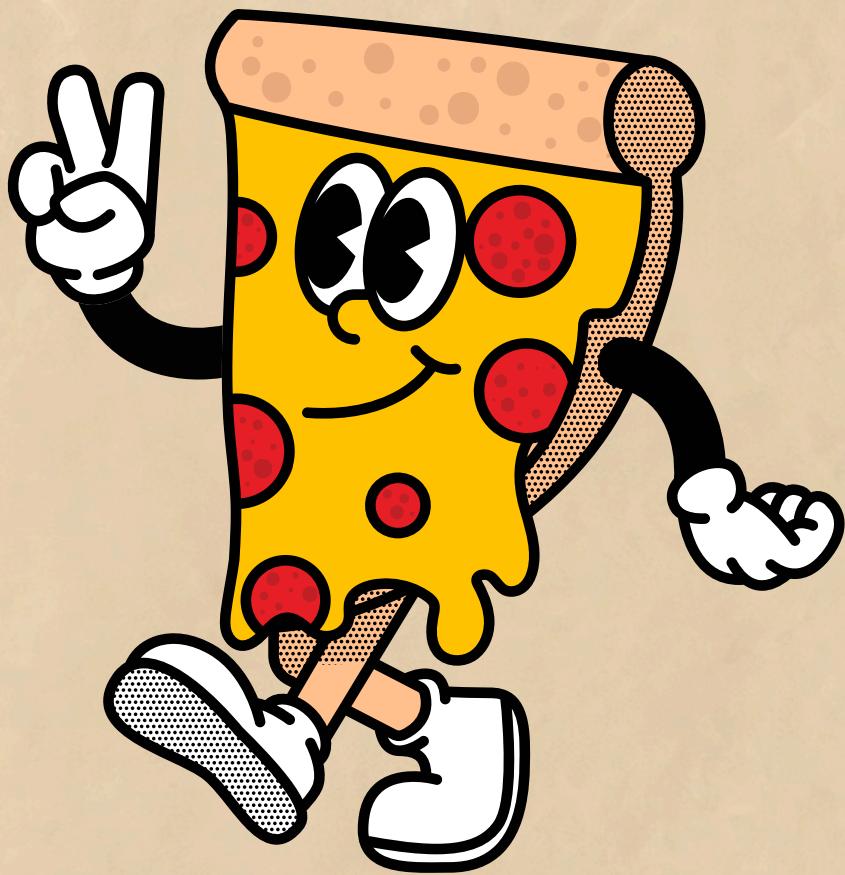


PIZZA

Pizza Hut Sales Analysis



By Sanjana Devi V V



SQL Project

This project aims is to clean the data, prep the data and create the tables and import the data from csv files to the SQL Server database and find actionable insights.



Project Goals

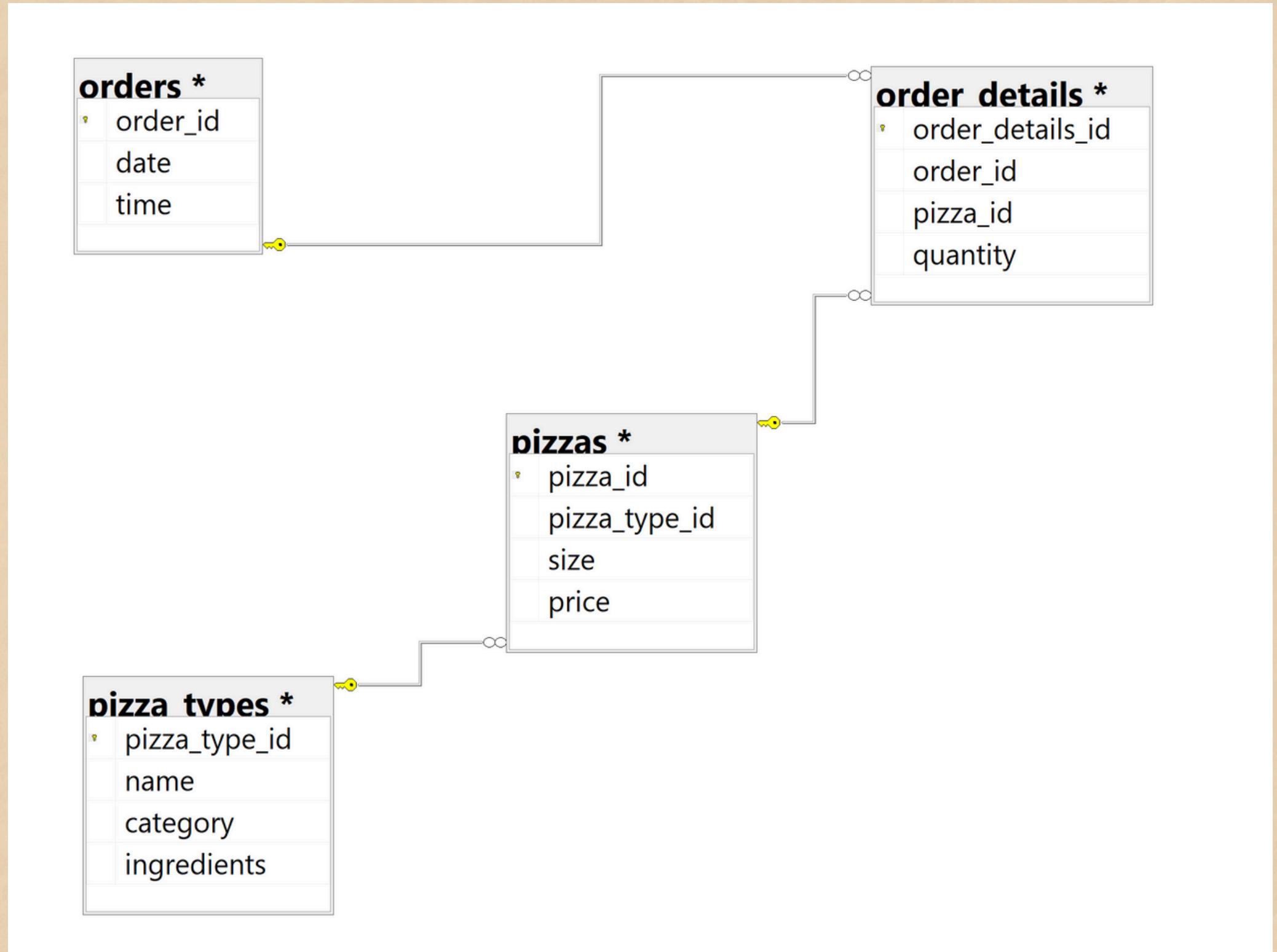
- Retrieve the total number of orders placed.
- Calculate the total revenue generated from pizza sales.
- Identify the highest-priced pizza.
- Identify the most common pizza size ordered.
- List the top 5 most ordered pizza types along with their quantities.

- 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.

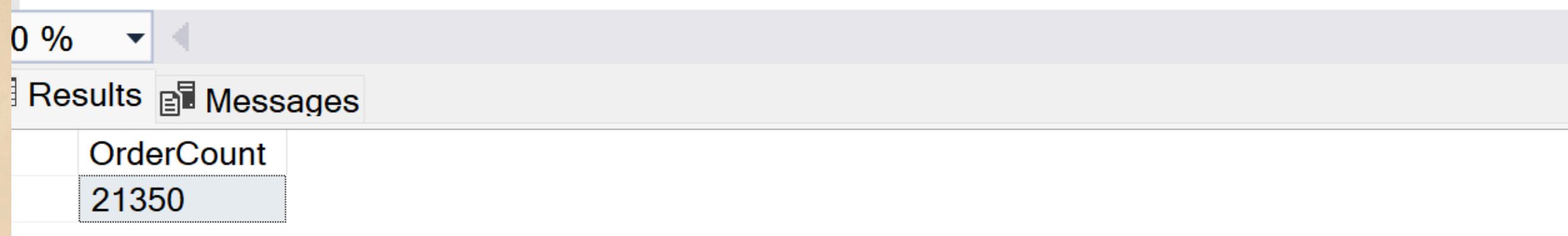


ER Diagram



Findings

```
1 --Retrieve the total number of orders placed.  
2 select count(order_id) as OrderCount from orders
```



The screenshot shows a SQL query results window. At the top, there is a progress bar indicating 0 % completion. Below the progress bar, there are 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 "OrderCount" with the value "21350".

OrderCount
21350



Findings

--Calculate the total revenue generated from pizza sales.

```
select round( sum( P.Price*OD.quantity),2) AS Total_Amount  
from Pizzas P  
join order_details OD  
on P.pizza_id=OD.Pizza_id
```

Total_Amount
817860.05



Findings

--Identify the highest-priced pizza.

```
select Top 1(P.Pizza_Id),round(P.Price,2) as TotalAmt from Pizzas P
```

```
Join Order_Details OD  
on P.Pizza_ID=OD.Pizza_ID  
order by TotalAmt desc
```

Pizza_Id	TotalAmt
the_greek_xxl	35.95



Findings

--Identify the most common pizza size ordered

```
SELECT P.size  
      ,count([Order_Details_Id]) AS OrderedTimes  
  FROM Pizzas P  
  Join  
order_details OD  
  on P.Pizza_ID=OD.Pizza_ID  
  Group by (P.Size)  
  order by OrderedTimes Desc
```

size	OrderedTimes
L	18526
M	15385
S	14137
XL	544
XXL	28



Findings

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

```
SELECT Top 5  
    PT.Name  
    ,Sum(OD.Quantity) as Quantity  
FROM Pizza_Types PT  
Join [pizzas] P  
    on PT.Pizza_Type_Id=P.Pizza_Type_Id  
Join Order_Details OD  
    on P.Pizza_Id=OD.Pizza_ID  
Group by PT.Name  
Order by quantity desc
```



Findings

Results Messages

	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



Findings

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

```
SELECT distinct([category])
    ,sum(OD.Quantity) as Quantity
FROM [pizza_types] PT
Join Pizzas P
    on PT.Pizza_Type_Id=P.Pizza_Type_Id
Join Order_Details OD
    on P.Pizza_Id=OD.Pizza_Id
Group by Category
Order by Quantity Desc
```

0 % ▾

Results Messages

category	Quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050



Findings

--Determine the distribution of orders by hour of the day.

```
SELECT DATEPART(HH, time) AS Hours  
      ,count(order_Id) OrderCount from orders  
Group by DATEPART(HH, time)  
Order by Hours
```

0 %

Results Messages

Hours	OrderCount
9	1
10	8
11	1231
12	2520
13	2455
14	1472
15	1468
16	1920
17	2336
0	18
1	19
2	20
3	21
4	22
5	23



Findings

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

```
SELECT distinct(PT.[category])
```

```
,Count(PT.Name )as CountPizza
```

```
FROM [pizza_types] PT
```

```
-Group by Category
```

100 % ▾

Results Messages

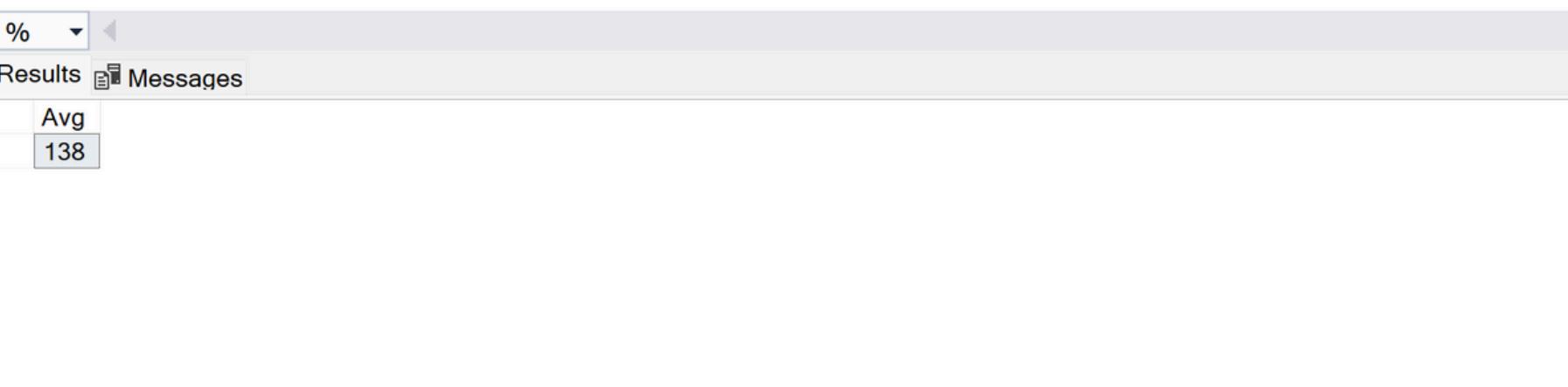
	category	CountPizza
1	Chicken	6
2	Classic	8
3	Supreme	9
4	Veggie	9



Findings

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

```
select avg(quantity) AS Avg from  
(select date ,sum(OD.quantity) as Quantity from orders O  
join Order_Details OD  
On O.Order_ID=OD.Order_ID  
group by date)  
as Order_quantity
```



The screenshot shows a SQL query results window. At the top, there are tabs for 'Results' and 'Messages'. The 'Results' tab is selected, displaying a single row with the column name 'Avg' and the value '138'.

Avg
138



Findings

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

```
select Top 3(PT.name)
```

```
,sum(OD.quantity*P.Price) As Revenue from pizza_types PT
```

```
join Pizzas P
```

```
on P.Pizza_Type_ID=PT.Pizza_Type_ID
```

```
Join Order_Details OD
```

```
on P.Pizza_Id=OD.Pizza_ID
```

```
Group By PT.Name
```

```
order by Revenue desc
```

% ▾
Results Messages

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



Findings

```
1 --Calculate the percentage contribution of each pizza type to total revenue.  
2 select (PT.Category)  
3     ,Round((sum(OD.quantity*P.Price) /(select Round(sum( OD.quantity*P.Price),2)  
4 from order_details OD  
5 join Pizzas P on P.Pizza_Id=OD.Pizza_Id))*100,2) As Revenue  
6 from pizza_types PT  
7 join Pizzas P  
8     on PT.Pizza_Type_ID=P.Pizza_Type_ID  
9 Join Order_Details OD  
10    on P.Pizza_Id=OD.Pizza_ID  
11 Group By PT.Category  
12 order by Revenue desc
```

00 % ▶

Results Messages

	Category	Revenue
1	Classic	26.91
2	Supreme	25.46
3	Chicken	23.96
4	Veggie	23.68



Findings

```
1 --Analyze the cumulative revenue generated over time.  
2 Select date  
3 ,sum(revenue) over (order by date)as Cum_revenue  
4 from  
5 (select o.date  
6 ,Round(sum( OD.quantity*P.Price),2) Revenue from orders O  
7 Join Order_details OD on  
8 O.Order_id=OD.order_id  
9 Join Pizzas P on  
10 OD.Pizza_id=P.Pizza_id  
11 group by (o.date)  
12 ) As Sales
```

% ▾

Results Messages

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
2015-01-09	21526.4
2015-01-10	23990.35
2015-01-11	25862.65
2015-01-12	27781.7
2015-01-13	29831.3
2015-01-14	32358.7
2015-01-15	34343.5



Findings

```
1 --Determine the top 3 most ordered pizza types based on revenue
2 select Top 3 (category)
3     ,PT.Pizza_Type_Id
4     ,Round(sum( OD.quantity*P.Price),2) Revenue
5 from [pizza_types] PT
6 join pizzas P
7     on P.Pizza_Type_Id=PT.Pizza_Type_Id
8 Join Order_details OD
9     on OD.Pizza_Id=P.Pizza_Id
10 Group by Category,PT.Pizza_Type_Id
11 order by Revenue desc
```

100 %

Results Messages

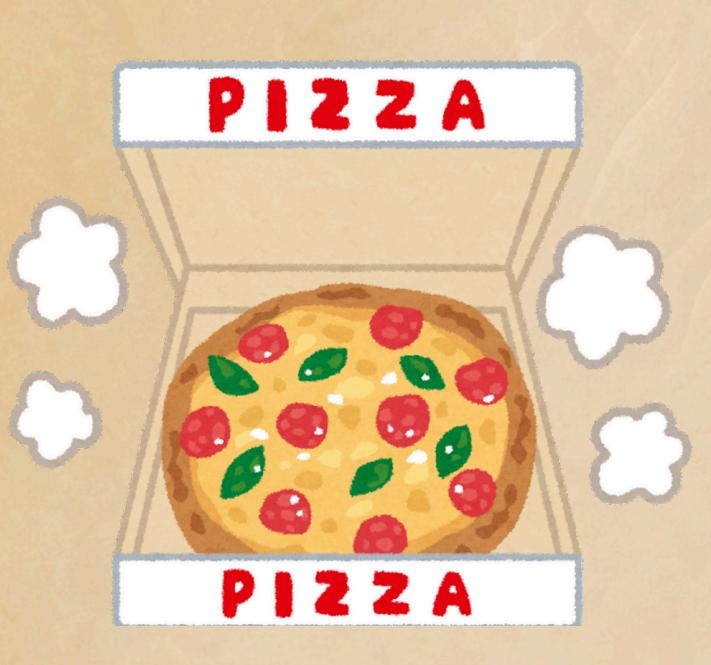
	category	Pizza_Type_Id	Revenue
1	Chicken	thai_ckn	43434.25
2	Chicken	bbq_ckn	42768
3	Chicken	cali_ckn	41109.5



Result

- Percentage Contribution of Each Pizza Type to Total Revenue:
- Analyze the data to identify which pizza types are the top contributors to total revenue. This will help in understanding customer preferences and the financial impact of each pizza type.
- Cumulative Revenue Over Time:
- Create a time series analysis to visualize how revenue has grown over time. Identify any trends, seasonal patterns, or anomalies that can provide insights into sales performance.
- Top 3 Most Ordered Pizza Types Based on Revenue for Each Pizza Category:
- Investigate the top-performing pizzas within each category. This analysis can help in inventory management, marketing strategies, and menu optimization.

PIZZA



Thank You!

