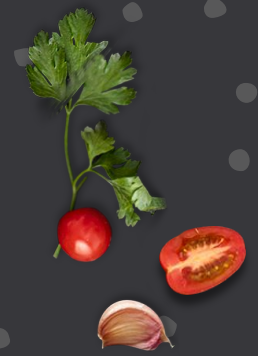
Pizza Sales Analysis
Using SQL

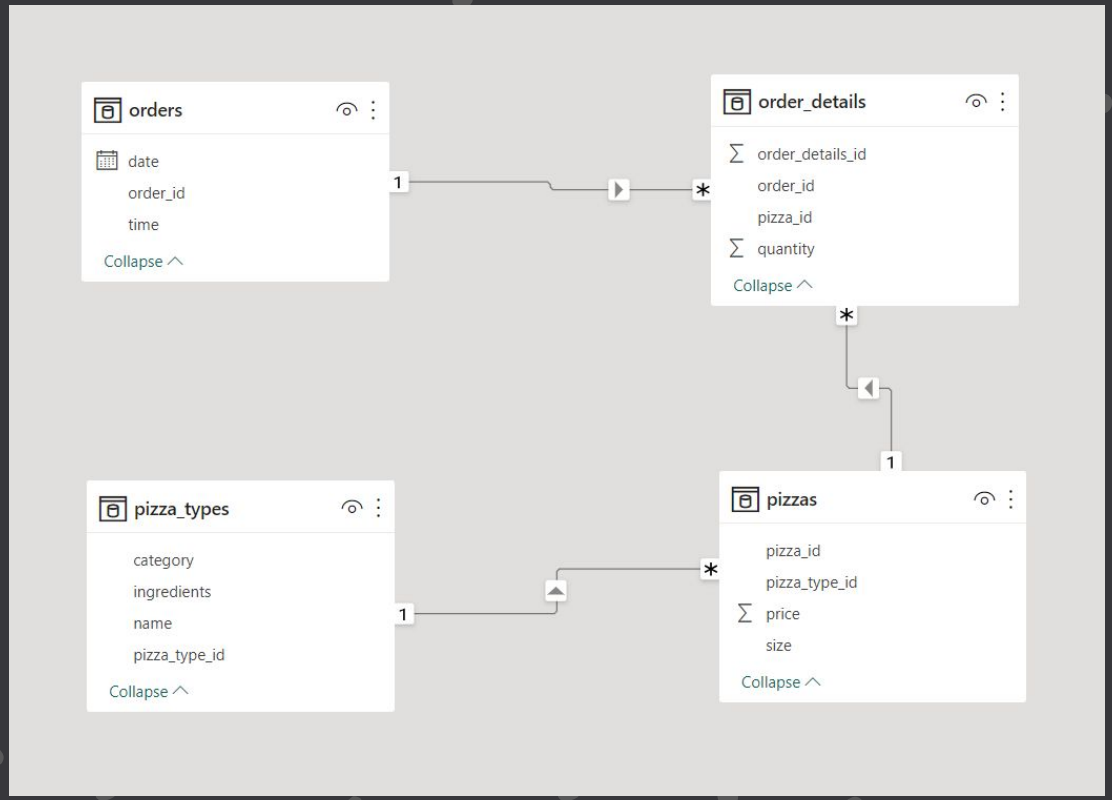
- By Khushi Tank

Objective

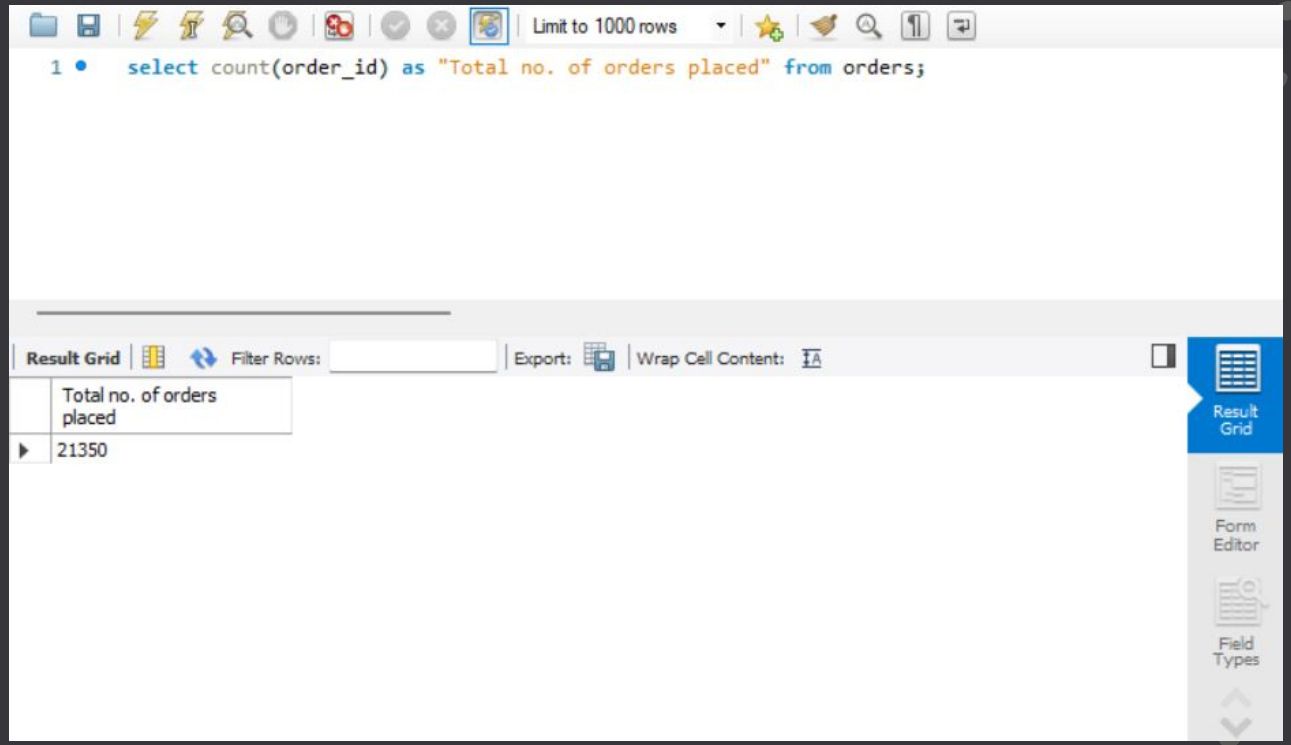
The aim of this project is to conduct a comprehensive analysis of pizza sales data to evaluate the overall performance of the restaurant. By examining sales trends and patterns, this analysis seeks to provide actionable insights that will inform strategic decision-making and enhance future planning. The ultimate goal is to optimize operational efficiency, improve customer satisfaction, and drive sustained business growth.

Model View




orders
date
order_id
time
Collapse
1
*
order_details
order_details_id
order_id
pizza_id
quantity
Collapse
*
1
pizza_types
category
ingredients
name
pizza_type_id
Collapse
1
*
pizzas
pizza_id
pizza_type_id
price
size
Collapse

Q1. Retrieve the total no. of orders placed.



Q2. Calculate total revenue generated from pizza sales.

```
1  select round(sum(order_details.quantity*pizzas.price)) as "Total Revenue"
2  from order_details
3  join pizzas
4  on order_details.pizza_id=pizzas.pizza_id;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Total Revenue
817860

Result Grid | Form Editor | Field Types

Q3. Identify highest prized pizza.

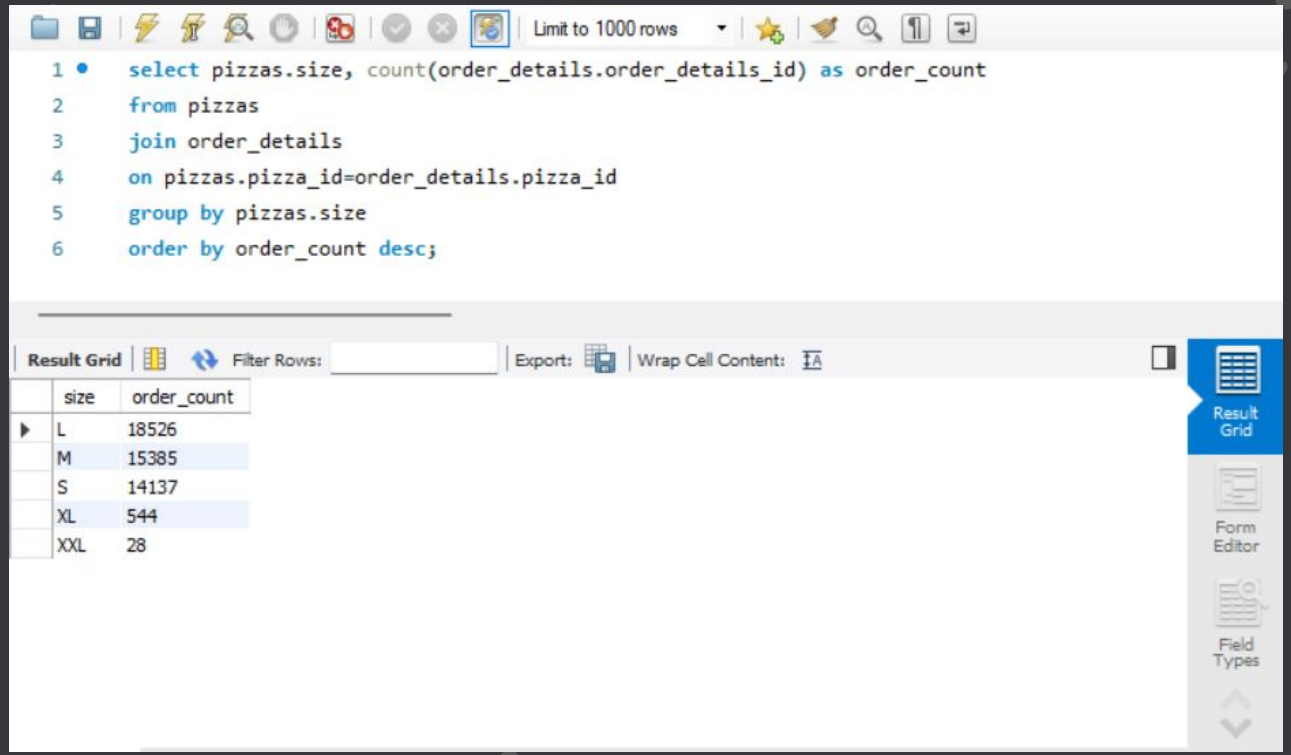
```
1  select pizza_types.name as Name, pizzas.price as "Maximum Price"
2  from pizzas
3  join pizza_types
4  on pizza_types.pizza_type_id=pizzas.pizza_type_id
5  order by pizzas.price desc
6  limit 1;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

Name	Maximum Price
The Greek Pizza	35.95

Result Grid
Form Editor
Field Types

Q4. Identify most common pizza size ordered.



```
1  select pizzas.size, count(order_details.order_details_id) as order_count
2  from pizzas
3  join order_details
4  on pizzas.pizza_id=order_details.pizza_id
5  group by pizzas.size
6  order by order_count desc;
```

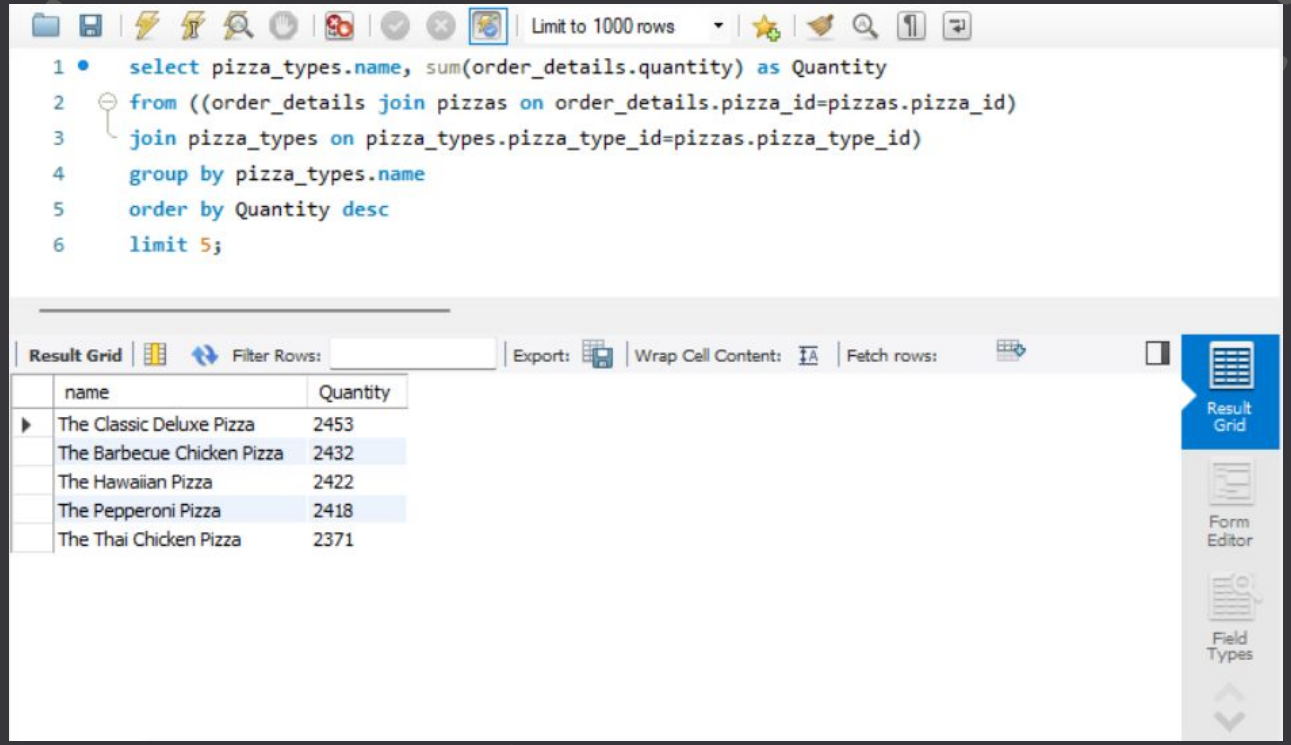
Limit to 1000 rows

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

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

Result Grid
Form Editor
Field Types

Q5. List top 5 most ordered pizza types along their quantity.



```
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 pizza_types.pizza_type_id=pizzas.pizza_type_id)
group by pizza_types.name
order by Quantity desc
limit 5;
```

Result Grid

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

Q6. Join necessary tables to find total quantity of each pizza category ordered.

```
1  select pizza_types.category, sum(order_details.quantity) as Quantity
2  from ((order_details join pizzas on order_details.pizza_id=pizzas.pizza_id)
3  join pizza_types on pizza_types.pizza_type_id=pizzas.pizza_type_id)
4  group by category
5  order by Quantity desc;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

category	Quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

Result Grid | Form Editor | Field Types

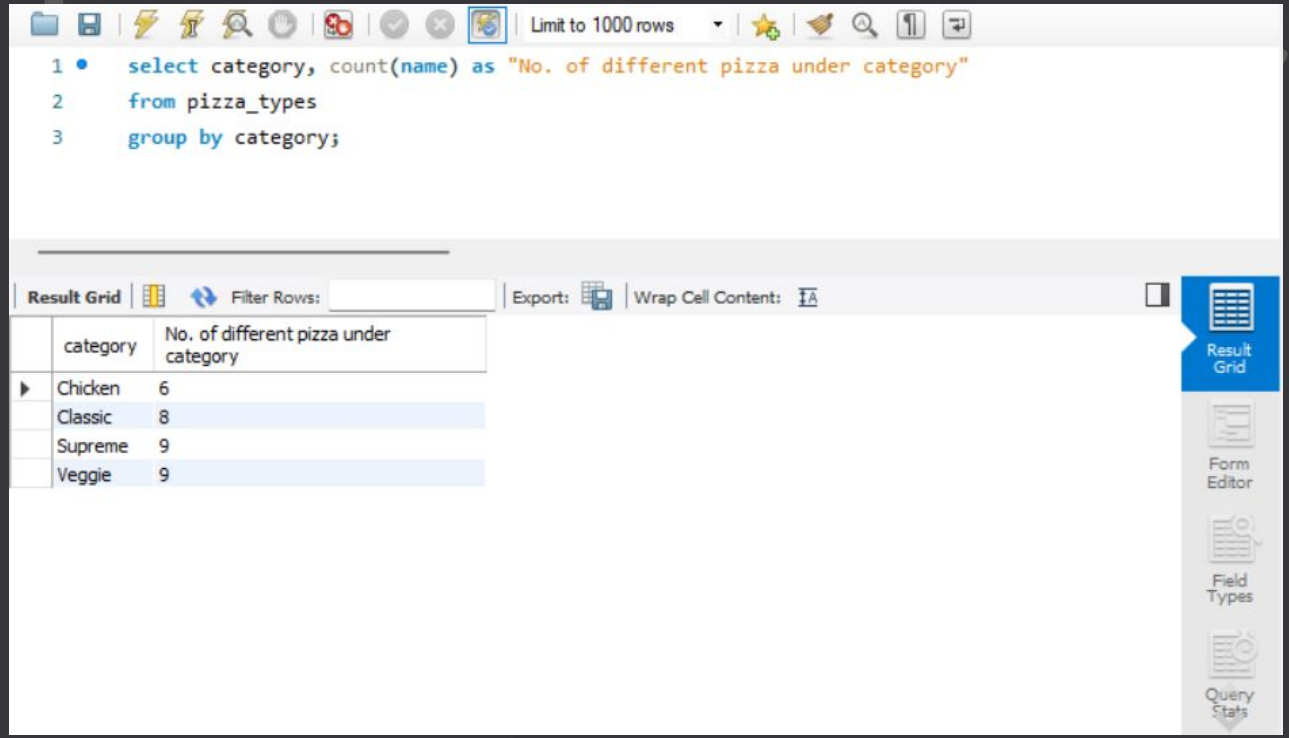
Q7. Determine distribution of orders by hour of the day.

```
1  select hour (time) as Hours, count(order_id) as "order count"
2  from orders
3  group by hour(orders.time)
4  order by hour(orders.time) asc;
```

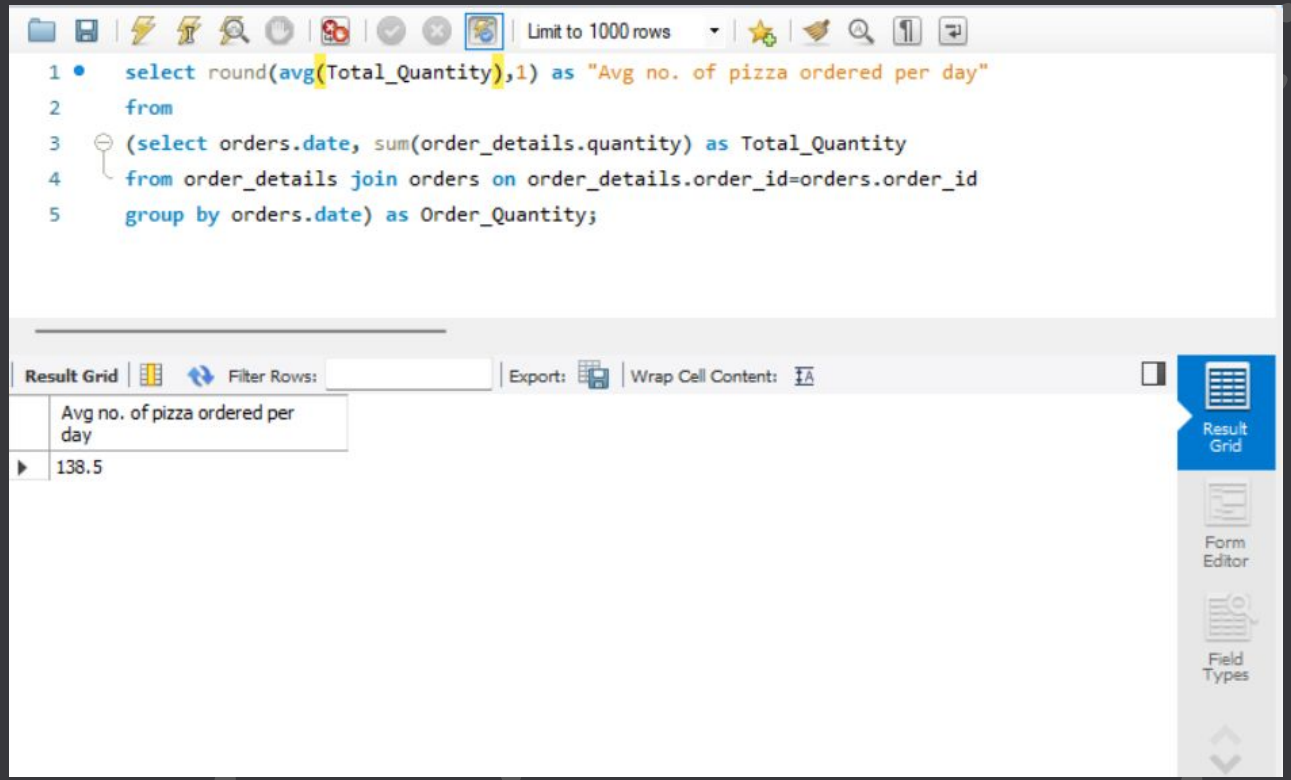
Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Hours	order 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

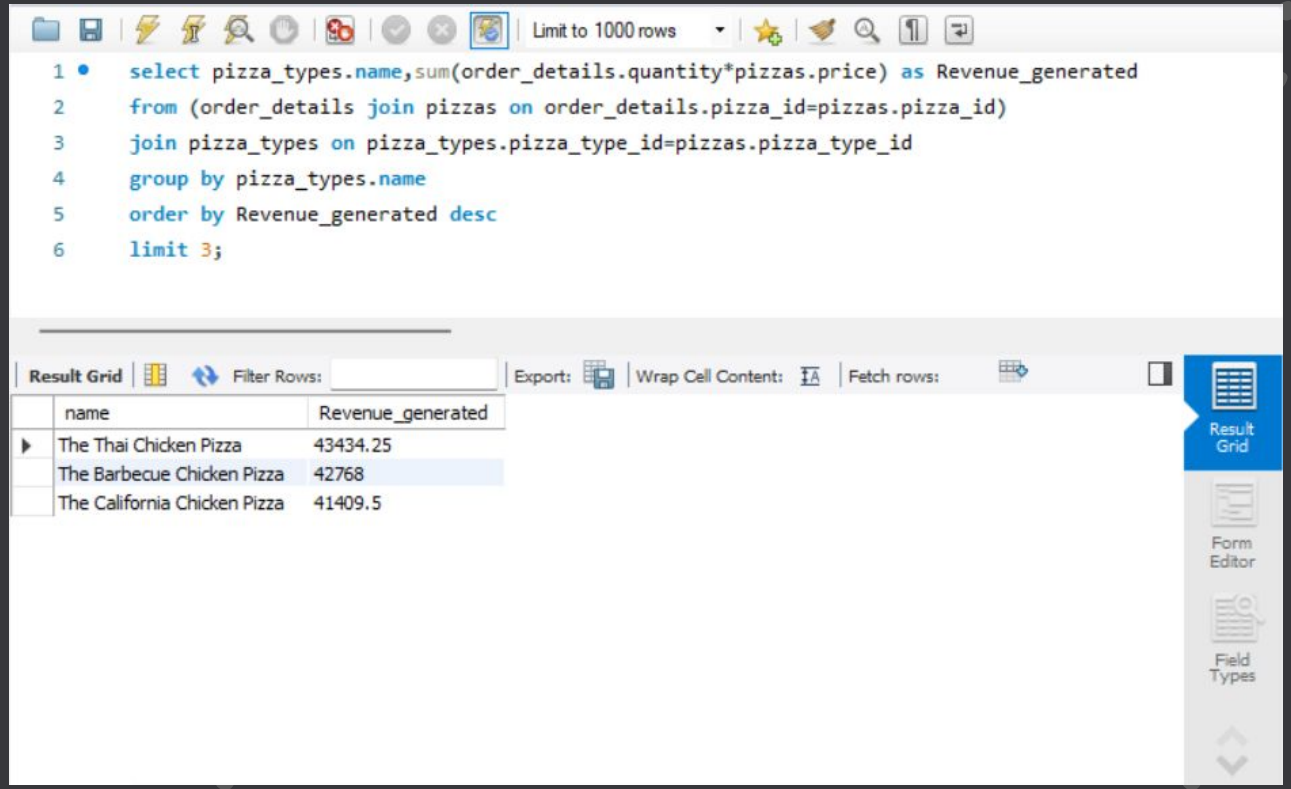
Q8. Join relevant tables to find category wise distribution of pizza



Q9. Group orders by date & calculate avg no. of pizza ordered per day.



Q10. Determine top 3 most ordered pizza types based on revenue.



```
select pizza_types.name,sum(order_details.quantity*pizzas.price) as Revenue_generated
from (order_details join pizzas on order_details.pizza_id=pizzas.pizza_id)
join pizza_types on pizza_types.pizza_type_id=pizzas.pizza_type_id
group by pizza_types.name
order by Revenue_generated desc
limit 3;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

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

Q11. Calculate percentage contribution of each pizza type to total revenue.

```
1  select pizza_types.name,sum(order_details.quantity*pizzas.price) as Revenue_generated
2  from (order_details join pizzas on order_details.pizza_id=pizzas.pizza_id)
3  join pizza_types on pizza_types.pizza_type_id=pizzas.pizza_type_id
4  group by pizza_types.name
5  order by Revenue_generated desc
6  limit 3;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

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

Result Grid | Form Editor | Field Types

Q12. Analyze the cumulative revenue generated over time.

```sql
select date,sum(Revenue) over (order by date) as Cumulative_Revenue
from (select date,sum(order_details.quantity*pizzas.price) as Revenue
      from order_details join pizzas on order_details.pizza_id=pizzas.pizza_id
join orders on orders.order_id=order_details.order_id
group by date) as Sales;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

date	Cumulative_Revenue
2015-01-22	50300.90000000001
2015-01-23	52724.600000000006
2015-01-24	55013.850000000006
2015-01-25	56631.40000000001
2015-01-26	58515.80000000001
2015-01-27	61043.85000000001
2015-01-28	63059.85000000001
2015-01-29	65105.150000000016
2015-01-30	67375.45000000001
2015-01-31	69793.30000000002
2015-02-01	72982.50000000001
2015-02-02	75311.10000000002
2015-02-03	77925.90000000002
2015-02-04	80159.80000000002
2015-02-05	82375.60000000002

Result Grid | Form Editor | Field Types | Query Stats

Q13. Determine the top 3 most ordered pizza types based on revenue for each pizza category.

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

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

name	Revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5
The Classic Deluxe Pizza	38180.5
The Hawaiian Pizza	32273.25
The Pepperoni Pizza	30161.75
The Spicy Italian Pizza	34831.25
The Italian Supreme Pizza	33476.75
The Sicilian Pizza	30940.5
The Four Cheese Pizza	32265.70000000065

Thanks!