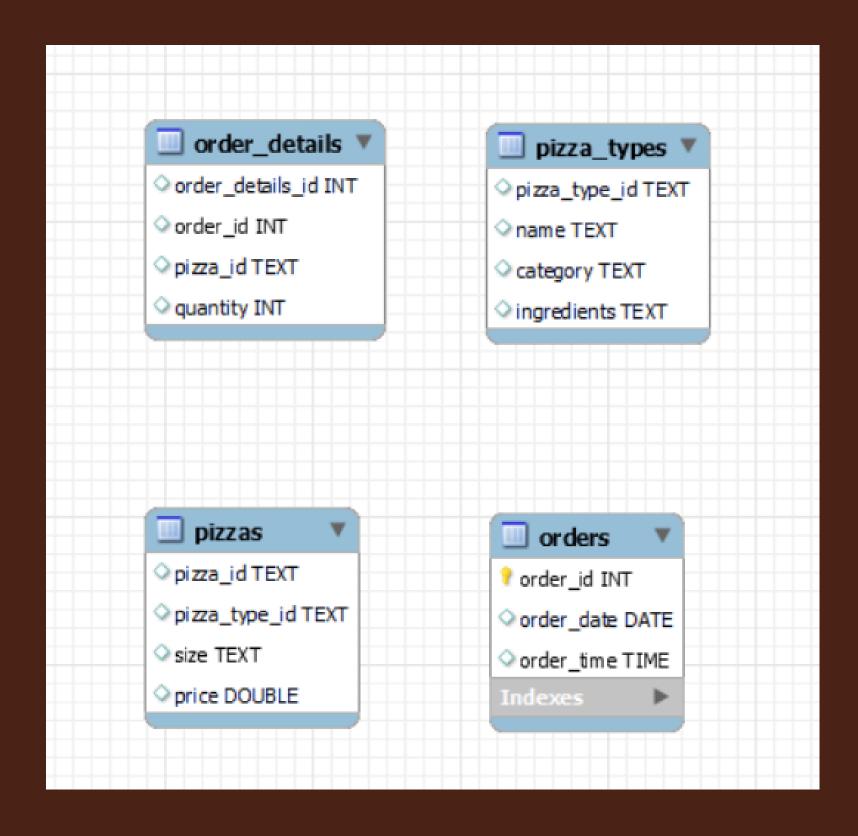
Delicious PIZZA

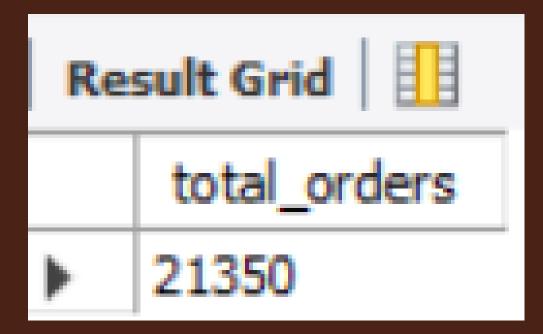
@reallygreatsite

SCHEMA



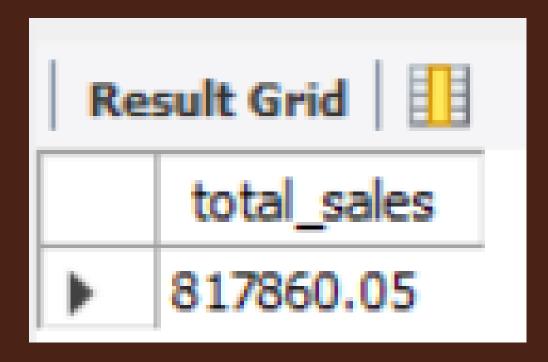
Retrieve the total number of orders placed.

select count(order_id) as total_orders from orders;



Calculate the total revenue generated from pizza sales.

```
select
round(sum(order_details.quantity * pizzas.price),2) as total_sales
from order_details join pizzas
on (pizzas.pizza_id = order_details.pizza_id);
```



Identify the highest priced pizza.

```
select pizza_types.name, pizzas.price
from pizza_types join pizzas
on (pizza_types.pizza_type_id = pizzas.pizza_type_id)
order by pizzas.price desc limit 1;
```



Identify the most common pizza size ordered.

```
select pizzas.size , count(order_details.order_details_id) as order_count
from pizzas join order_details
on (order_details.pizza_id = pizzas.pizza_id)
group by pizzas.size order by order_count desc;
```

Result Grid		Filter Rows:
	size	order_count
>	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, sum(order_details.quantity) as quantity
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.name order by quantity desc limit 5;
```

name quantity	
rianic quartery	
▶ The Classic Deluxe Pizza 2453	
The Barbecue Chicken Pizza 2432	
The Hawaiian Pizza 2422	
The Pepperoni Pizza 2418	
The Thai Chicken Pizza 2371	

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

```
select pizza_types.category , sum(order_details.quantity) as total_quantity
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 order by total_quantity desc limit 5;
```

Result Grid				
	category	total_quantity		
•	Classic	14888		
	Supreme	11987		
	Veggie	11649		
	Chicken	11050		

Determine the distribution of orders by hour of the day.

select hour(order_time), count(order_id)as order_count from orders
group by hour(order_time);

Result Grid				
	hour(order_time)	order_count		
•	11	1231		
	12	2520		
	13	2455		
	14	1472		
	15	1468		
	16	1920		

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

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

Result Grid				
	category	count(name)		
•	Chicken	6		
	Classic	8		
	Supreme	9		
	Veggie	9		

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

```
select round(avg(total_orders),0) from
  (select orders.order_date, sum(order_details.quantity) as total_orders
from orders join order_details
on orders.order_id = order_details.order_id
group by orders.order_date)as data;
```

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

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

Result Grid			
	name revenue		
•	The Thai Chicken Pizza	43434.25	
	The Barbecue Chicken Pizza	42768	
	The California Chicken Pizza 41409		

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

```
select pizza_types.category,
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 pizzas.pizza_id = order_details.pizza_id) * 100,2) 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 order by revenue desc;
```

Result Grid				
	category	revenue		
*	Classic	26.91		
	Supreme	25.46		
	Chicken	23.96		
	Veggie	23.68		

Analyze the cumulative revenue generated over time.

```
select order_date, sum(revenue) over (order by order_date) as cumulative_revenue
from
(select orders.order_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 orders.order_date) as sales;
```

Result Grid		♦ Filte	er Rows:
	order_date		cumulative_revenue
•	2015-01-01		2713.8500000000004
	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

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

```
select category , 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;</pre>
```

Result Grid				
	category	name	revenue	
•	Chicken	The Thai Chicken Pizza	43434.25	
	Chicken	The Barbecue Chicken Pizza	42768	
	Chicken	The California Chicken Pizza	41409.5	
	Classic	The Classic Deluxe Pizza	38180.5	
	Classic	The Hawaiian Pizza	32273.25	
	Classic	The Pepperoni Pizza	30161.75	

Re	Result Grid				
	category	name	revenue		
	Supreme	The Spicy Italian Pizza	34831.25		
	Supreme	The Italian Supreme Pizza	33476.75		
	Supreme	The Sicilian Pizza	30940.5		
	Veggie	The Four Cheese Pizza	32265.70000000065		
	Veggie	The Mexicana Pizza	26780.75		
	Veggie	The Five Cheese Pizza	26066.5		



THANK

