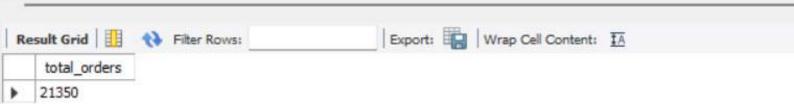
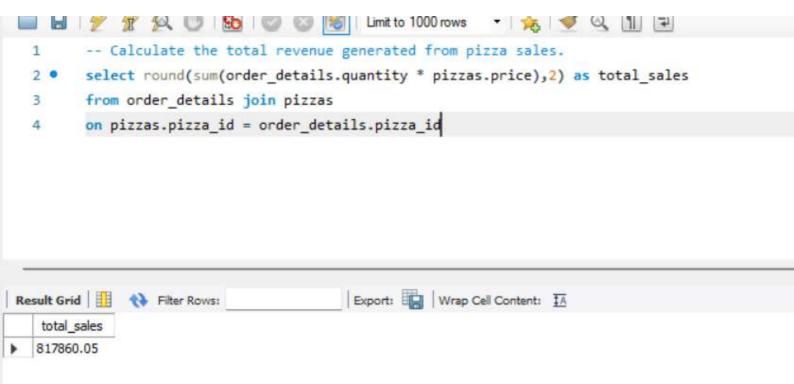
- 1 -- Retrieve the total number of orders placed.
- 2 select count(order_id) as total_orders from orders;



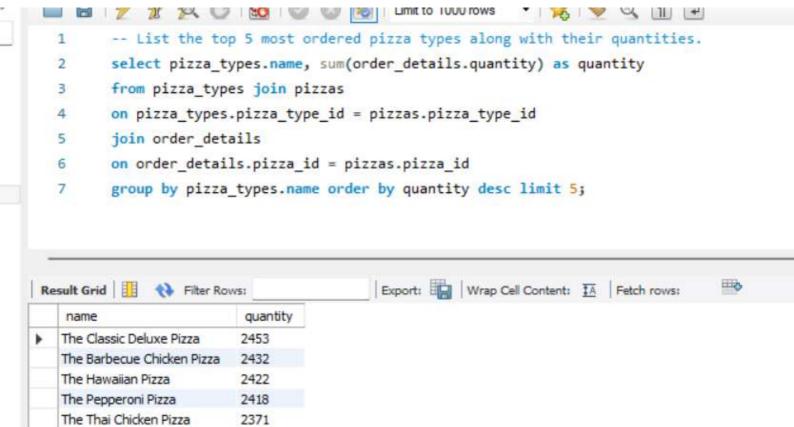


```
1   -- Identify the highest-priced pizza.
2   select pizza_types.name , pizzas.price
3   from pizza_types join pizzas
4   on pizza_types.pizza_type_id = pizzas.pizza_type_id
5   order by pizzas.price desc limit 1;
```



```
1 -- Identify the most common pizza size ordered.
2 select pizzas.size, count(order_details.order_details_id) as order_count
3 from pizzas join order_details
4 on pizzas.pizza_id = order_details.pizza_id
5 group by pizzas.size order by order_count desc;
```

R	esult Gr	id 🔠 🙌 Fi	lter Rows:	Export:	Wrap Cell Content:	ĪĀ
	size	order_count				
١	L	18526				
	M	15385				
	S	14137				
	XL	544				
	XXL	28				



```
-- find the total quantity of each pizza category ordered.

select pizza_types.category, 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.category order by quantity desc;
```

-						
Re	esult Grid	(A) Fit	ter Rows:	Export:	Wrap Cell Content:	<u>∓</u> A
	category	quantity				
•	Classic	14888				
	Supreme	11987				
	Veggie	11649				
	Chickon	11050				

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

select hour(order_time) AS Hour , count(order_id) AS order_count

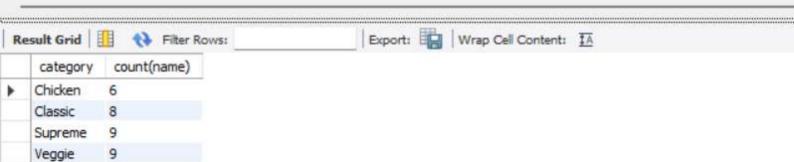
from orders

group by hour(order_time);

Export: Wrap Cell Content: IA

	Hour	order_count
٠	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
	10	8
	9	1

- 1 -- Join relevant tables to find the category-wise distribution of pizzas.
- 2 select category, count(name) from pizza_types
- 3 group by category;



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

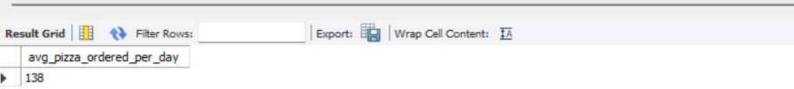
select round(avg(quantity), 0) as avg_pizza_ordered_per_day from

(select orders.order_date, sum(order_details.quantity) as quantity

from orders join order_details

on orders.order_id = order_details.order_id

group by orders.order_date) as order_quamtity;
```



```
-- 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;
```

Re	esult Grid 🚹 💎 Filter Ro	WS1	Export: Wrap Cell Content: IA Fetch rows:	-
	name	revenue		
•	The Thai Chicken Pizza	43434.25		
	The Barbecue Chicken Pizza	42768		
	The California Chicken Pizza	41409.5		

```
1
       -- Calculate the percentage contribution of each pizza type to total revenue.
 2
       select pizza types.category,
    pround(sum(order_details.quantity * pizzas.price) / (select
3
       round(sum(order_details.quantity * pizzas.price),2) as total_sales
4
      from order details join pizzas
5
      on pizzas.pizza_id = order_details.pizza_id) *100, 2) as revenue
6
7
       from pizza types join pizzas
       on pizzas.pizza_type_id = pizza_types.pizza_type_id
8
9
       join order details
       on order_details.pizza_id = pizzas.pizza_id
10
       group by pizza_types.category order by revenue desc;
11
```

R	esult Grid	∏ 🙌 Filb	er Rows:	Export:	Wrap Cell Content:	ĪĀ
	category	revenue				
	Classic	26.91				
	Supreme	25.46				
	Chicken	23.96				
	Veggie	23.68				