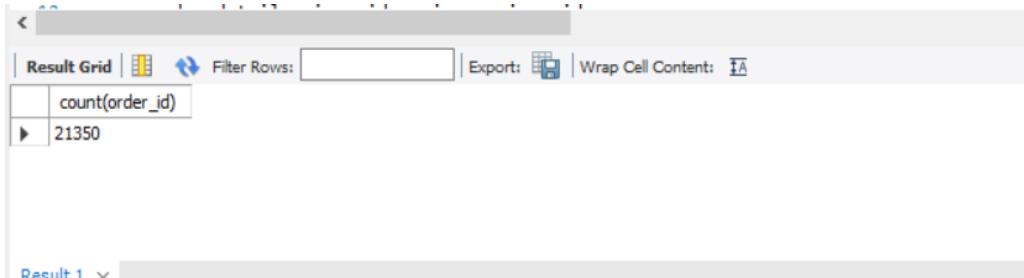


SQL Queries

1. Retrieve total no. of orders placed

```
select count(order_id) from pizzahut.orders
```

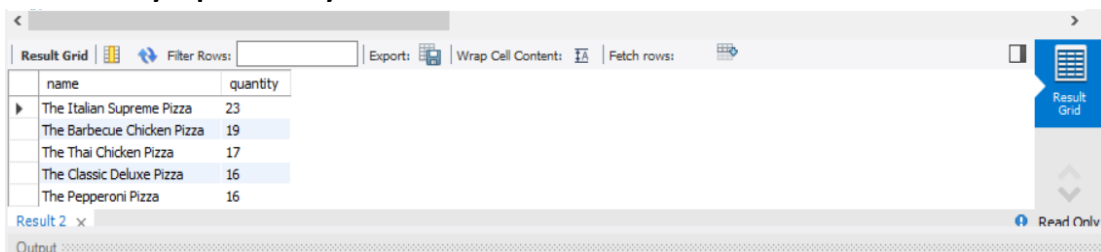


A screenshot of a database query result interface. The interface includes a toolbar with options like 'Result Grid', 'Filter Rows', 'Export', and 'Wrap Cell Content'. Below the toolbar, a table displays the result of the query 'select count(order_id) from pizzahut.orders'. The table has one column labeled 'count(order_id)' and one row with the value '21350'.

count(order_id)
21350

2. Top 5 pizza types along with their quantity

```
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;
```

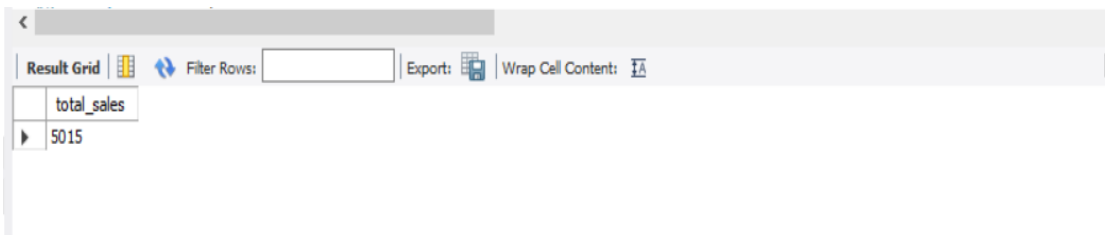


A screenshot of a database query result interface showing the top 5 pizza types by quantity. The interface includes a toolbar with options like 'Result Grid', 'Filter Rows', 'Export', 'Wrap Cell Content', 'Fetch rows', and 'Read Only'. Below the toolbar, a table displays the results of the query. The table has two columns: 'name' and 'quantity'. The results are ordered by quantity in descending order, with 'The Italian Supreme Pizza' having the highest quantity (23).

name	quantity
The Italian Supreme Pizza	23
The Barbecue Chicken Pizza	19
The Thai Chicken Pizza	17
The Classic Deluxe Pizza	16
The Pepperoni Pizza	16

3. Calculate total revenue generated from pizza sales

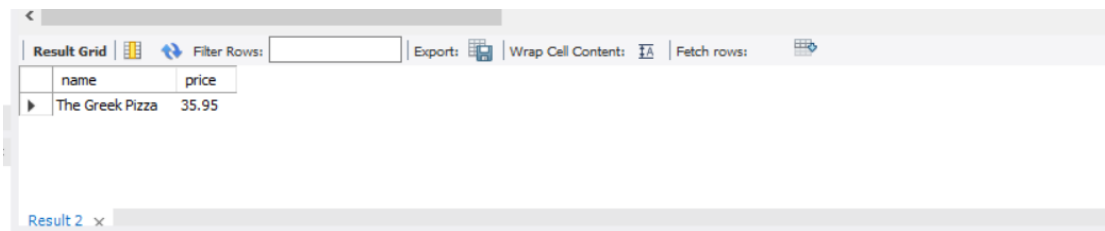
```
select  
sum(order_details.quantity * pizzas.price) as total_sales  
from order_details join pizzas  
on pizzas.pizza_id = order_details.pizza_id
```



total_sales
5015

4. Identify 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;
```



name	price
The Greek Pizza	35.95

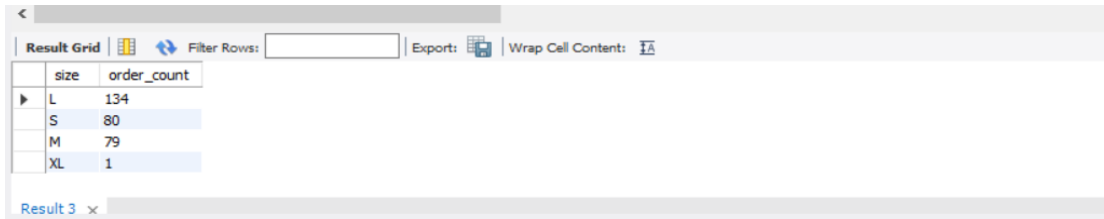
5. Identify most common pizza size ordered

```
select pizzas.size, count(order_details.order_details_id) as  
order_count
```

```

from pizzas join order_details
on pizzas.pizza_id = order_details.pizza_id
group by pizzas.size order by order_count desc ;

```



The screenshot shows a database query result grid with the following data:

size	order_count
L	134
S	80
M	79
XL	1

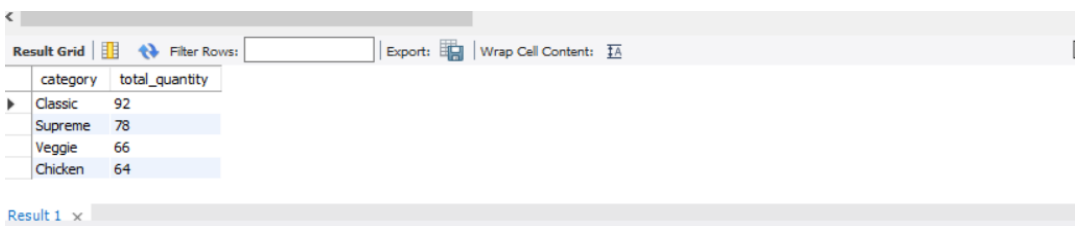
Result 3 x

6. 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;

```



The screenshot shows a database query result grid with the following data:

category	total_quantity
Classic	92
Supreme	78
Veggie	66
Chicken	64

Result 1 x

7. Distribution of orders by hour of day

```

select hour(orders.time) ,count( order_id ) from orders
group by hour(orders.time);

```

hour(orders.time)	count(order_id)
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

8. Find category wise distribution of pizza

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

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

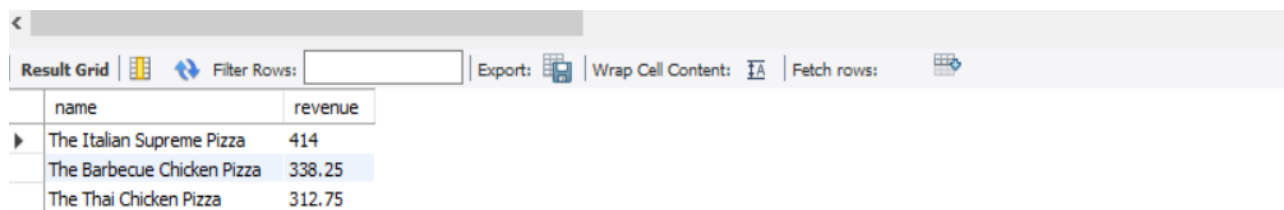
9. Group the orders by date and calculate the avg no of pizzas ordered per day

```
select avg(quantity) from
(select orders.date , sum(order_details.order_details_id)
as quantity
from orders join order_details
on orders.order_id = order_details.order_id
group by orders.date) as order_quantity;
```

avg(quantity)
21682.5000

10. Top 3 most ordered pizza based on revenue

```
select 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.name order by revenue desc limit 3;
```



The screenshot shows a database interface with a 'Result Grid' tab. The grid displays the results of the SQL query for the top 3 pizzas by revenue. The columns are 'name' and 'revenue'. The rows are: 'The Italian Supreme Pizza' with revenue 414, 'The Barbecue Chicken Pizza' with revenue 338.25, and 'The Thai Chicken Pizza' with revenue 312.75. The second row is highlighted in blue.

name	revenue
The Italian Supreme Pizza	414
The Barbecue Chicken Pizza	338.25
The Thai Chicken Pizza	312.75

11. Calculate % contribution of each pizza type to total revenue

```
select pizza_types.category,  
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 order by revenue desc limit  
3;
```

<

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

Fetch rows:

	category	revenue
▶	Supreme	1393.25
	Classic	1353.5
	Chicken	1144