QUESTION 1- Retrieve The total number of orders placed.

```
4 •  SELECT
5        COUNT(order_id) AS total_orders
6    FROM
7        orders
```

Result Grid | Filter Rows: | Export:

	total_orders
▶	21350

QUESTION 2- calculate the total revenue genertaed fro pizza sales.

```
3 •  SELECT
4        ROUND(SUM(order_details.quantity * pizzas.price),
5                2) AS total_sales
6    FROM
7        order_details
8            JOIN
9        pizzas ON pizzas.pizza_id = order_details.pizza_id
10
11
12
```

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

	total_sales
▶	817860.05

QUESTION 3- Identify the highest-priced pizza.

```sql
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
```

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

name	price
The Greek Pizza	35.95

QUESTION 4- Identify the most common pizza size ordered.

```sql
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
```

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

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

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

```sql
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
```

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

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

QUESTION 6 - join the necessary table to find the total quantity of each pizza category ordered.

```sql
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
```

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

category	quantity
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

QUESTION 7 - determine the distribution of orders by hour of the day.

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

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

hour	order_count
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

QUESTION 8 - join relevant tables to find the category-wise distribution of pizzas

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

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

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

QUESTION 9 - group the orders by date and calculate the average number of pizza ordered perday

```sql
select avg(quantity) 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_quantity
```

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

avg(quantity)
138.4749

QUESTION 10 - determine the top 3 most ordered pizza types based on revenue

```sql
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 | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

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

QUESTION 11 -Analyze the cumulative revenue generated over time.

```sql
select order_date,
sum(revenue) over(order by order_date) as cum_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 | Filter Rows: | Export: | Wrap Cell Content:

order_date	cum_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