SQL PROJECT ON PIZZA SALES

HELLO!!!!

I am Aditya Reddy. I have made a project which analyzes pizza sales data to provide insights into customer preferences, revenue patterns, and the popularity of different pizza types. By using data visualization and statistical analysis, the project aims to identify trends that can guide business strategies, such as inventory management, marketing efforts, and customer engagement. With this data-driven approach, the project helps highlight areas to optimize sales and improve overall business performance in the competitive food industry.

DATABASE AND QUESTIONS

DATABASE

I have got the data for my project from the link given below:
https://github.com/Ayushi0214/pizza-sales---SQL/blob/main/Questions.txt

QUESTIONS

The link aslo included some questiions segregated in three domains:Easy,Medium and Hard.

RETRIEVE THE TOTAL NUMBER OF ORDERS PLACED.

```
#QUERY:
    select count(order_id) as total_orders from orders;
```

Result Grid

	total_orders
▶	10337

IDENTIFY THE MOST COMMON PIZZA SIZE ORDERED.

```sql
SELECT
    pizzas.size, COUNT(order_details.quantity) 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
LIMIT 1;
```

Result Grid | Filte

	size	order_count
▶	L	389

CALCULATE THE TOTAL REVENUE GENERATED FROM PIZZA SALES.

```sql
#QUERY:
SELECT
    SUM(pizzas.price * order_details.quantity) AS total_revenue_generated
FROM
    pizzas
        JOIN
    order_details ON pizzas.pizza_id = order_details.pizza_id;
```

Result Grid | Filter Rows

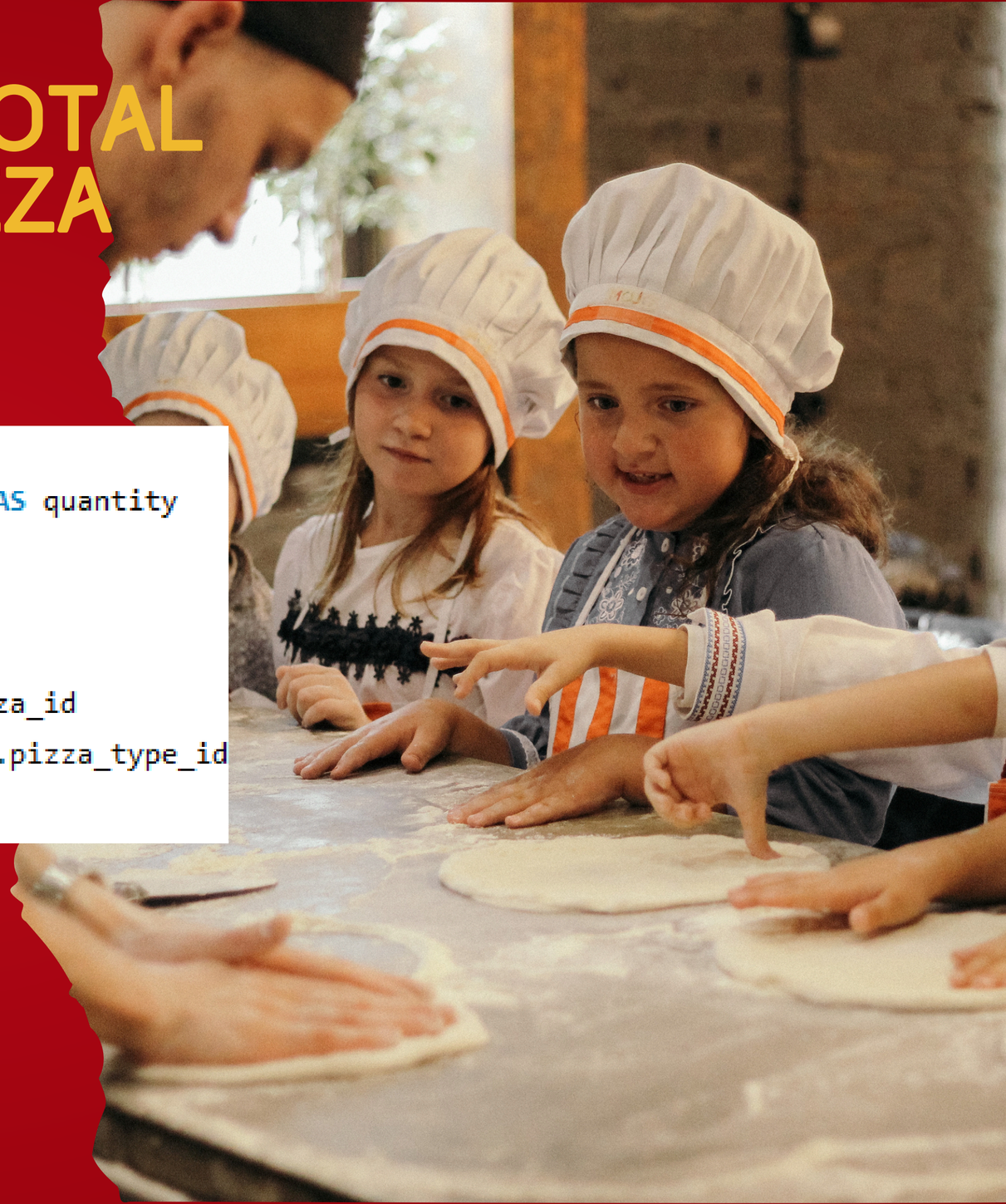
	total_revenue_generated
▶	16503.95

JOIN THE NECESSARY TABLES TO FIND THE TOTAL QUANTITY OF EACH PIZZA CATEGORY ORDERED.

```sql
SELECT
    pizza_types.category, COUNT(order_details.quantity) AS quantity
FROM
    pizzas
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
    join pizza_types on pizzas.pizza_type_id=pizza_types.pizza_type_id
GROUP BY category;
```

Result Grid | Filter

	category	quantity
▶	Classic	296
	Veggie	227
	Supreme	232
	Chicken	220



IDENTIFY THE HIGHEST-PRICED PIZZA.

```sql
#QUERY:
SELECT
    pizza_types.name, pizzas.price AS highest_priced_pizza
FROM
    pizzas
        JOIN
    pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
ORDER BY price DESC
LIMIT 1;
```

Result Grid | Filter Rows:

	name	highest_priced_pizza
▶	The Greek Pizza	35.95

LIST THE TOP 5 MOST ORDERED PIZZA TYPES ALONG WITH THEIR QUANTITIES.

```sql
SELECT
    pizza_types.name,
    COUNT(order_details.quantity) AS quantity
FROM
    pizzas
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
    join pizza_types on pizza_types.pizza_type_id=pizzas.pizza_type_id
GROUP BY name
ORDER BY quantity DESC
LIMIT 5;
```

Result Grid | Filter Rows:

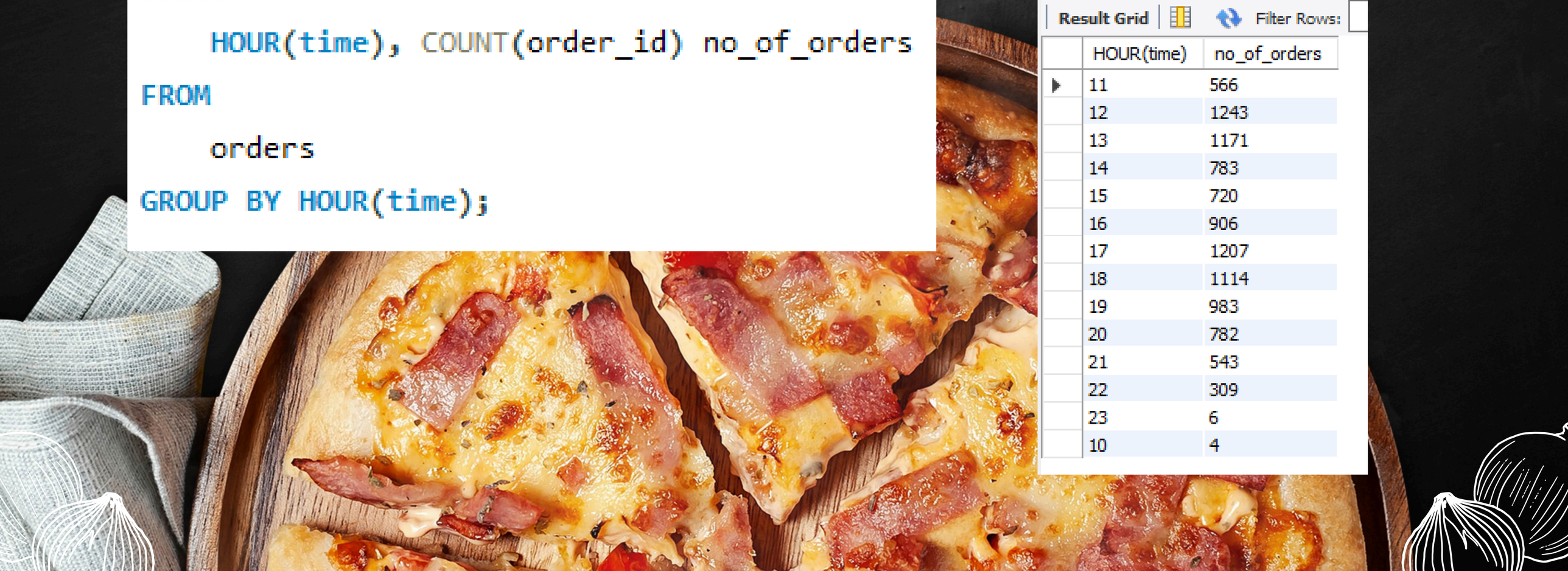
name	quantity
The Thai Chicken Pizza	54
The Pepperoni Pizza	54
The Classic Deluxe Pizza	51
The Italian Supreme Pizza	49
The Hawaiian Pizza	48

DETERMINE THE DISTRIBUTION OF ORDERS BY HOUR OF THE DAY.

```sql
SELECT
    HOUR(time), COUNT(order_id) no_of_orders
FROM
    orders
GROUP BY HOUR(time);
```

Result Grid | Filter Rows:

HOUR(time)	no_of_orders
11	566
12	1243
13	1171
14	783
15	720
16	906
17	1207
18	1114
19	983
20	782
21	543
22	309
23	6
10	4

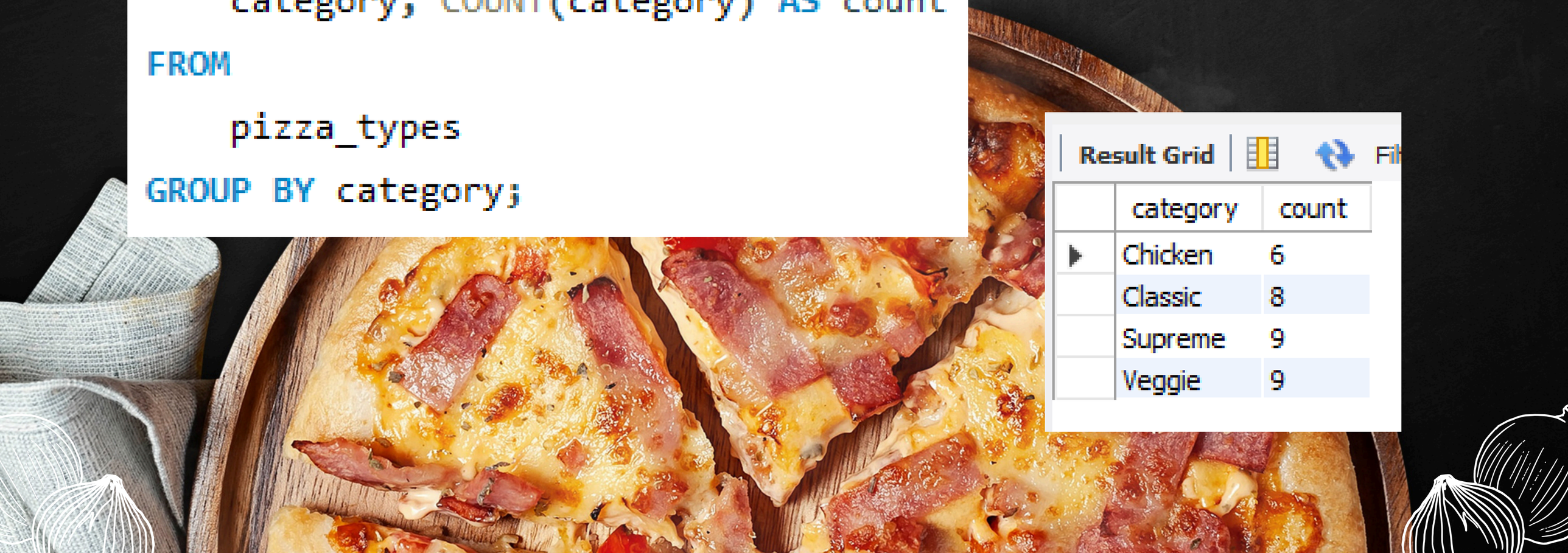


JOIN RELEVANT TABLES TO FIND THE CATEGORY-WISE DISTRIBUTION OF PIZZAS.

```sql
SELECT
    category, COUNT(category) AS count
FROM
    pizza_types
GROUP BY category;
```

Result Grid

	category	count
▶	Chicken	6
	Classic	8
	Supreme	9
	Veggie	9



GROUP THE ORDERS BY DATE AND CALCULATE THE AVERAGE NUMBER OF PIZZAS ORDERED PER DAY.

```sql
SELECT AVG(quantity)
FROM (
    SELECT DATE(date) AS order_date, COUNT(order_id) AS quantity
    FROM orders
    GROUP BY DATE(date)
) AS orders_quan;
```

Result Grid

	AVG(quantity)
▶	73.8357

DETERMINE THE TOP 3 MOST ORDERED PIZZA TYPES BASED ON REVENUE.

```sql
SELECT
    pizza_types.name AS name,
    SUM(pizzas.price * order_details.quantity) AS revenue,
    pizza_types.category AS category
FROM
    pizzas
        JOIN
    pizza_types ON pizzas.pizza_type_id = pizza_types.pizza_type_id
        JOIN
    order_details ON order_details.pizza_id = pizzas.pizza_id
GROUP BY name,category
ORDER BY revenue DESC
LIMIT 3;
```

Result Grid | Filter Rows:

name	revenue	category
The Thai Chicken Pizza	996.5	Chicken
The Barbecue Chicken Pizza	897.5	Chicken
The Italian Supreme Pizza	869	Supreme

THANK YOU!