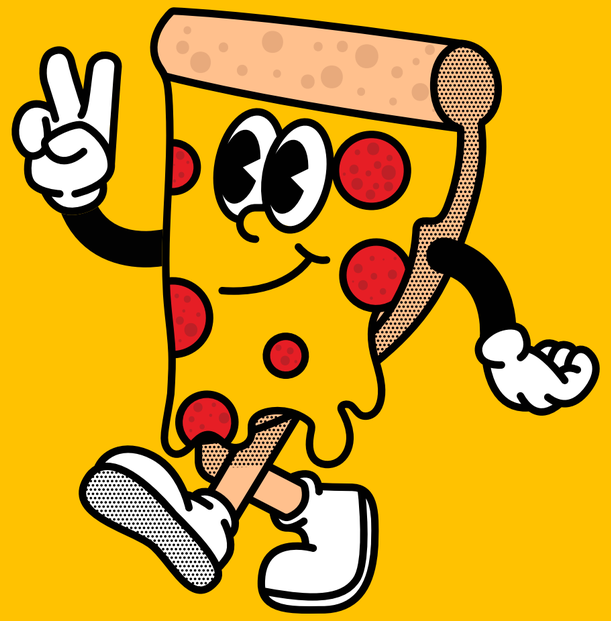
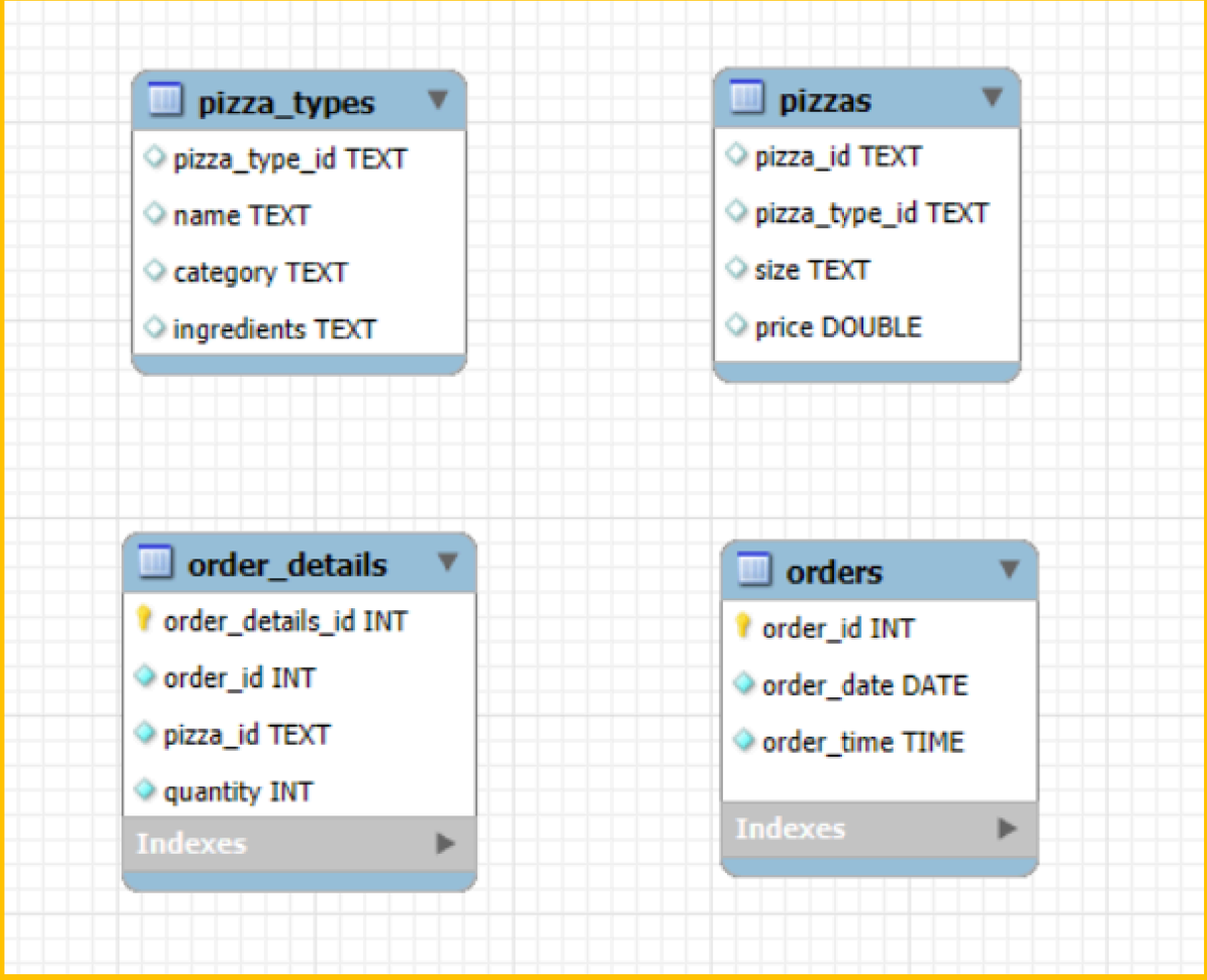


PIZZAHUT SALES ANALYSIS

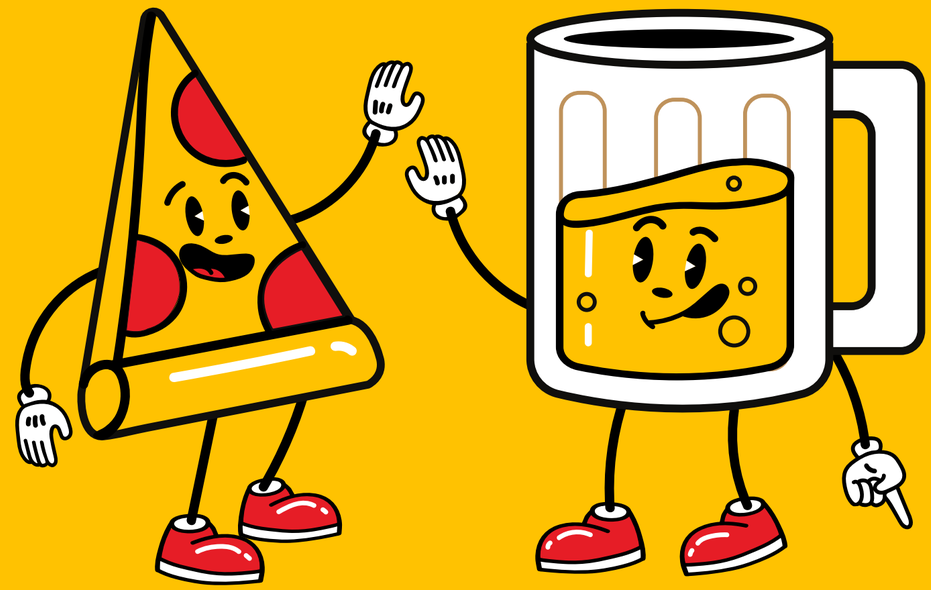
Hello!! My Name is Siddharth !

And In this Project I have utilize the SQL queries to solves a questions that are related to Pizza sales.

DATABASE

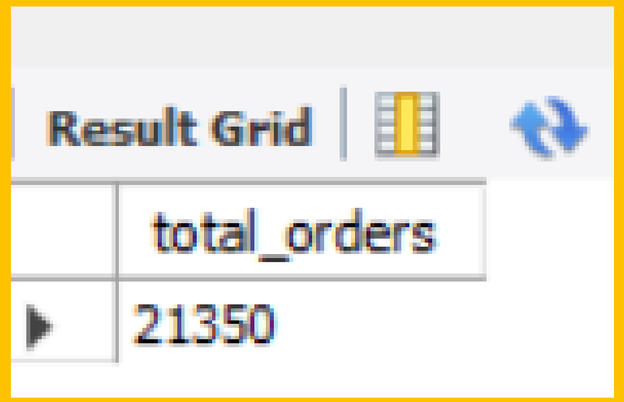


OVER 5 BILLION PIZZAS ARE SOLD GLOBALLY EACH YEAR!



Retrieve the total number of orders placed ?

```
SELECT COUNT(order_id) AS total_orders
FROM orders;
```

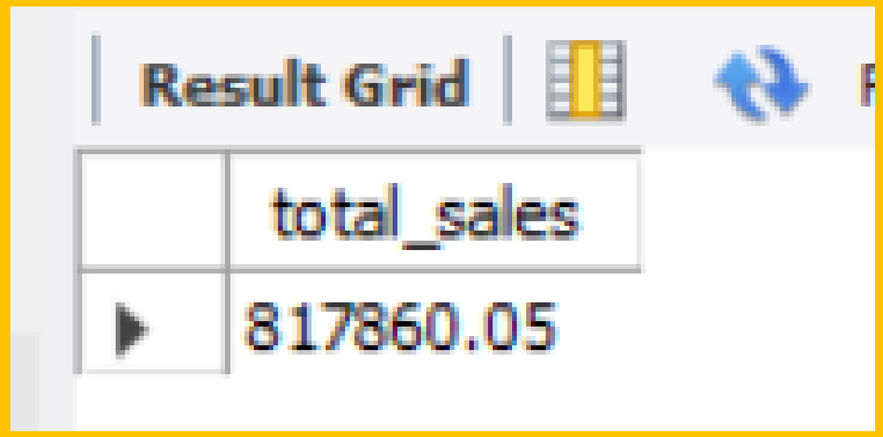


Business Implication:

"This shows the total customer transactions, helping assess overall business activity.

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;
```

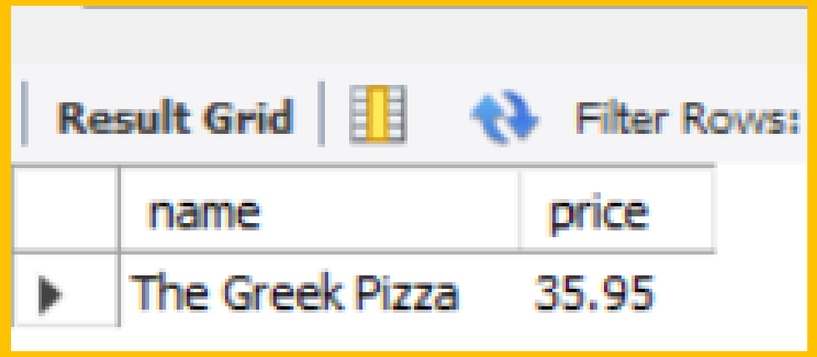


Business Implication:

This KPI is fundamental for financial performance assessment

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;
```



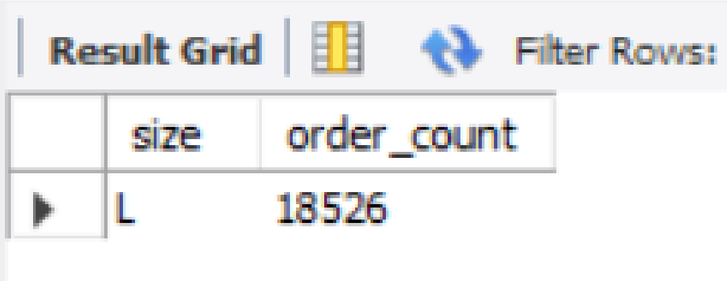
	name	price
▶	The Greek Pizza	35.95

Business Implication:

Identifying premium products helps with pricing strategy and marketing focus.

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 | Filter Rows:

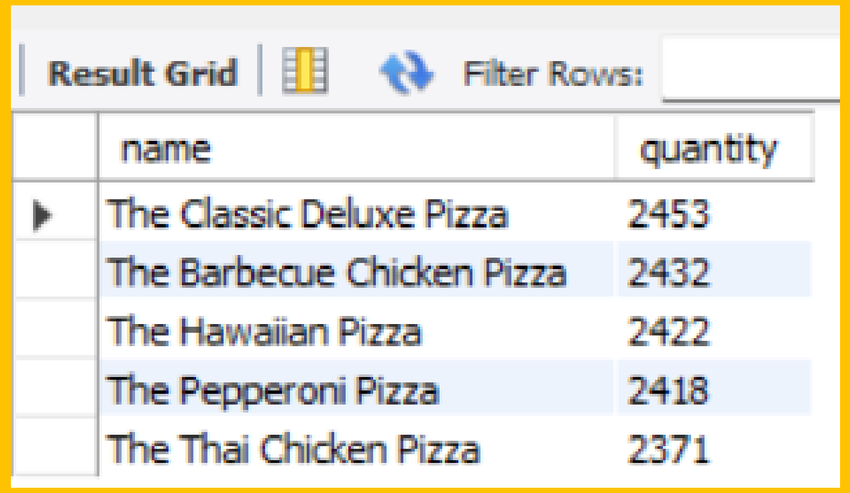
	size	order_count
▶	L	18526

Business Implication:

Understanding size popularity helps with inventory planning and production focus.

List the top 5 most ordered pizza types 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:

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

Business Implication:

Shows customer favorites.

Join the necessary tables 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;
```

Result Grid | Filter Rows:

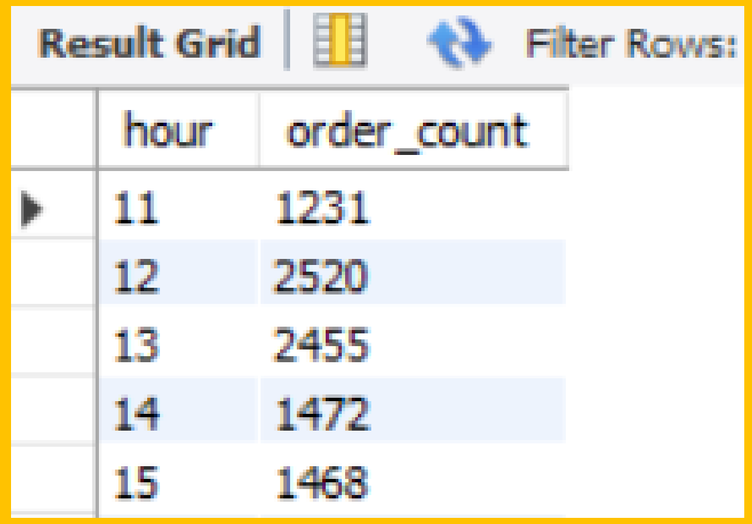
category	quantity
Chicken	11050
Veggie	11649
Supreme	11987
Classic	14888

Business Implication:

Helps understand customer preferences by category for menu planning.

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);
```



Result Grid | Filter Rows:

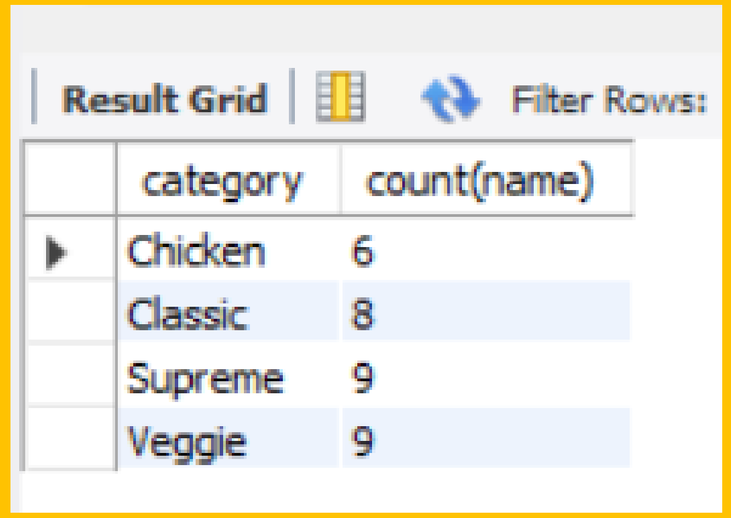
hour	order_count
11	1231
12	2520
13	2455
14	1472
15	1468

Business Implication:

Identifies peak hours for staffing and potential promotion opportunities.

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

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

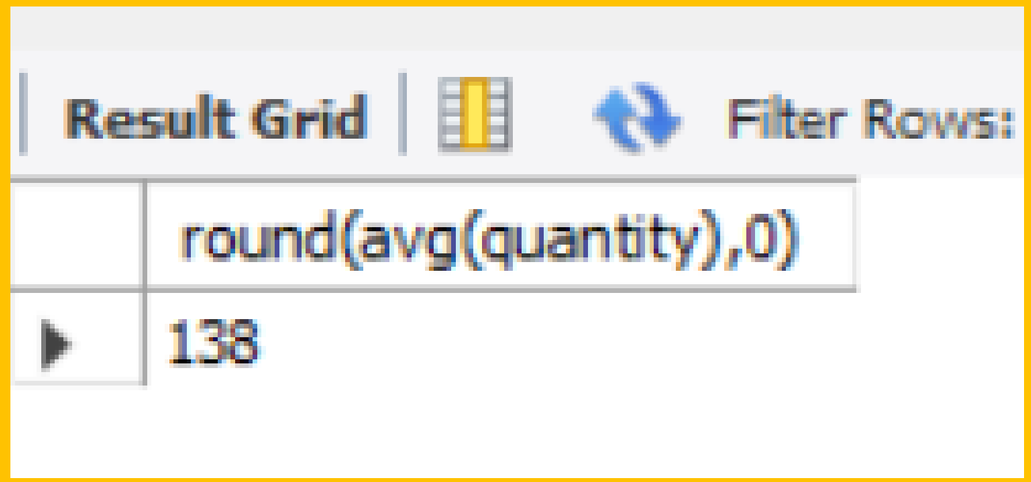


Result Grid | Filter Rows:

	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(quantity),0) 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 ;
```



THANK YOU