Pizza Sales Analytics Problem Statement

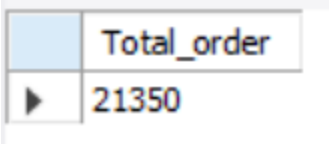
You are tasked with analyzing a pizza sales dataset to uncover insights into customer orders, revenue generation, and product performance. The dataset contains information about orders, pizzas, categories, sizes, and prices. Your goal is to analysis three levels of complexity:

Basic Analysis

1. **Retrieve the total number of orders placed.**

```
-- Retrieve the total number of orders placed.
select count(order_id) Total_order from Orders;
```

Output:



	Total_order
▶	21350

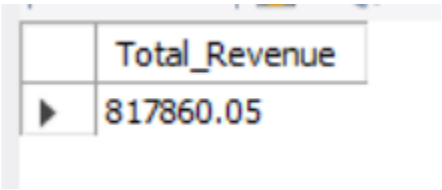
Definition: The count of all unique orders recorded in the dataset.

Business Insight: It reflects overall demand and customer activity. Managers use this to gauge sales volume, track growth, and forecast inventory needs.

2. **Calculate the total revenue generated from pizza sales.**

```
-- Calculate the total revenue generated from pizza sales.
SELECT ROUND(SUM(p.price * od.quantity), 2) AS Total_Revenue
FROM pizzas p JOIN order_details od ON p.pizza_id = od.pizza_id;
```

Output:



	Total_Revenue
▶	817860.05

Definition: The sum of all sales amounts from pizzas sold.

Business Insight: It shows how much money the restaurant earned from pizza sales during the dataset period.

Use Case: Helps managers evaluate profitability, set sales targets, and identify growth opportunities.

3. **Identify the highest-priced pizza**

```
-- Identify the highest-priced pizza.
select t.name , t.category , p.price from pizza_types t
join pizzas p  on p.pizza_type_id = t.pizza_type_id
order by p.price  desc limit 1;
```

Output:

	name	category	price
▶	The Greek Pizza	Classic	35.95

Definition: The pizza with the maximum unit price in the menu.

Business Insight: Identifies premium offerings that may appeal to niche customers or drive high-margin sales.

Use Case: Helps managers understand pricing strategy, evaluate whether premium pizzas are selling, and decide if luxury items should stay on the menu.

4. **Identify the most common pizza size ordered**

```
-- Identify the most common pizza size ordered.
select p.size,count(od.order_details_id) quantity from pizzas p join order_details od
on p.pizza_id = od.pizza_id
group by p.size
order by quantity desc  limit 1;
```

Output:

	size	quantity
▶	L	18526

Definition: The pizza size (e.g., Small, Medium, Large, Extra Large) that appears most frequently in orders.

Business Insight: Reveals customer demand patterns and helps optimize inventory, pricing, and marketing strategies.

Use Case: If medium pizzas dominate, managers can focus promotions on that size or adjust stock levels accordingly.

5. **List the top 5 most ordered pizza types along with their quantities**

```
-- List the top 5 most ordered pizza types along with their quantities.
select t.name ,sum(od.quantity) quantity from pizzas p
join pizza_types t on p.pizza_type_id = t.pizza_type_id
join  order_details od on p.pizza_id = od.pizza_id
group by t.name
order by quantity desc limit 5;
```

Output:

	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

Definition: The five pizza types (e.g., Margherita, Pepperoni, BBQ Chicken) that were ordered the most, along with the total number of units sold.

Business Insight: Identifies customer favorites and helps managers focus on inventory, marketing, and promotions.

Use Case: If “The Classic Deluxe pizza” consistently ranks #1, the restaurant can ensure steady supply, run combo deals, or highlight it in advertising.

Intermediate Analysis

1. **Join the necessary tables to find the total quantity of each pizza category ordered.**

```
-- Join the necessary tables to find the total quantity of each pizza category ordered.
select t.category ,sum(od.quantity) totalquantity from pizzas p
join pizza_types t on p.pizza_type_id = t.pizza_type_id
join  order_details od on p.pizza_id = od.pizza_id
group by t.category;
```

Output:

	category	totalquantity
▶	Classic	14888
	Veggie	11649
	Supreme	11987
	Chicken	11050

Definition: The sum of all pizzas sold, grouped by their category.

Business Insight: Shows which categories dominate sales volume.

Use Case: Helps managers understand customer preferences at a broader level (e.g., do people prefer meat-heavy pizzas or vegetarian options).

2. **Determine the distribution of orders by hour of the day**

```
-- Determine the distribution of orders by hour of the day.
select hour(order_time) By_hour ,count(order_id) from orders
group by by_hour ;
```

Output:

By_hour	count(order_id)
15	1468
16	1920
17	2336
18	2399
19	2009
20	1642
21	1198
22	663
23	28

Definition: The count of orders grouped by the hour they were placed.

Business Insight: Reveals peak hours when customers are most likely to order pizzas.

Use Case: Helps managers plan staffing, kitchen prep, and marketing campaigns (e.g., lunch vs. dinner promotions).

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

```
-- Join relevant tables to find the category-wise distribution of pizzas.
select t.category , count(name) AS total_pizzas from pizza_types t
group by t.category;
```

Output:

category	total_pizzas
Chicken	6
Classic	8
Supreme	9
Veggie	9

Definition: The breakdown of pizzas sold across categories such as *Classic, Supreme, Veggie, Chicken*.

Business Insight: Shows which categories dominate sales volume and which one's lag**.**

Use Case: Helps managers decide whether to expand certain categories, adjust pricing, or run promotions**.**

4. **Group the orders by date and calculate the average number of pizzas ordered per day.**

```
-- Group the orders by date and calculate the average number of pizzas ordered per day.
select round(avg(quantity),0) AS avg_pizzas_per_day from (select sum(od.quantity) quantity,o.order_date from
order_details od
join orders o on o.order_id =od.order_id
group by o.order_date) order_quantity;
```

Output:

	avg_pizzas_per_day
▶	138

Definition: The mean number of pizzas sold per day across the dataset period**.**

Business Insight: Provides a baseline for daily sales expectations.

Use Case: Helps managers forecast inventory, plan staffing, and set realistic daily sales targets.

5. **Determine the top 3 most ordered pizza types based on revenue.**

```
-- Determine the top 3 most ordered pizza types based on revenue.
select t.name,sum(p.price*od.quantity) Total_Revenue from pizza_types t
join pizzas p on t.pizza_type_id=p.pizza_type_id
join order_details od on p.pizza_id=od.pizza_id
group by t.name
order by Total_Revenue desc limit 3 ;
```

Output:

	name	Total_Revenue
▶	The Thai Chicken Pizza	43434.25
	The Barbecue Chicken Pizza	42768
	The California Chicken Pizza	41409.5

Definition: The three pizza types that generated the highest total revenue (quantity × price).

Business Insight: Identifies not just popular pizzas, but the ones that contribute most to the bottom line.

Use Case: Helps managers prioritize marketing, promotions, and inventory for high-earning pizzas.

Advanced Analysis

1. **Calculate the percentage contribution of each pizza type to total revenue.**

```sql
-- Calculate the percentage contribution of each pizza type to total revenue.
SELECT t.category,ROUND(SUM(p.price * od.quantity) /
(SELECT SUM(p.price * od.quantity) total_sale
FROM order_details od JOIN pizzas p ON od.pizza_id = p.pizza_id) * 100,2) AS revenue
FROM pizza_types t JOIN pizzas p ON t.pizza_type_id = p.pizza_type_id
JOIN order_details od ON p.pizza_id = od.pizza_id
GROUP BY t.category
ORDER BY revenue DESC
LIMIT 3;
```

Output:

	category	revenue
▶	Classic	26.91
	Supreme	25.46
	Chicken	23.96

Definition: The share of total revenue generated by each pizza type, expressed as a percentage.

Business Insight: Highlights which pizzas are the biggest revenue drivers and which ones contribute marginally.

Use Case: Helps managers identify star products, evaluate menu performance, and decide where to focus promotions or discounts.

2. **Analyze the cumulative revenue generated over time.**

```
-- Analyze the cumulative revenue generated over time.
select order_Date , sum(revenue) over(order by order_Date) as cum_relative from
(select o.order_Date , sum(od.quantity*p.price) as revenue from
pizzas p join order_details od on p.pizza_id = od.pizza_id
join orders o on o.order_id = od.order_id
group by o.order_Date) as sales ;
```

Output:

order_Date	cum_relative
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
2015-01-07	16560.7
2015-01-08	19399.05
2015-01-09	21526.4
2015-01-10	23990.350000000002
2015-01-11	25862.65
2015-01-12	27781.7
2015-01-13	29831.300000000003
2015-01-14	32358.700000000004
2015-01-15	34343.50000000001
2015-01-16	36937.65000000001

Definition: The running total of revenue, calculated day by day (or month by month), showing how sales accumulate over time.

Business Insight: Reveals growth trajectory, seasonality, and long-term performance.

Use Case: Helps managers track progress toward revenue targets, identify peak sales periods, and forecast future performance**.**

3. **Determine the top 3 most ordered pizza types based on revenue for each pizza category.**

```
-- Determine the top 3 most ordered pizza types based on revenue for each pizza category.
select category , name , Total_revenue, rank() over(partition by category order by Total_revenue desc) as rn from
(select t.category,t.name, round(sum(od.quantity*p.price),0) Total_revenue from pizzas p
join order_details od on od.pizza_id = p.pizza_id
join pizza_types t on t.pizza_type_id = p.pizza_type_id
group by t.category,t.name) as a;
```

Output:

category	name	Total_revenue	rn
Chicken	The Thai Chicken Pizza	43434	1
Chicken	The Barbecue Chicken Pizza	42768	2
Chicken	The California Chicken Pizza	41410	3
Chicken	The Southwest Chicken Pizza	34706	4
Chicken	The Chicken Alfredo Pizza	16900	5
Chicken	The Chicken Pesto Pizza	16702	6
Classic	The Classic Deluxe Pizza	38180	1
Classic	The Hawaiian Pizza	32273	2
Classic	The Pepperoni Pizza	30162	3
Classic	The Greek Pizza	28454	4
Classic	The Italian Capocollo Pizza	25094	5
Classic	The Napolitana Pizza	24087	6
Classic	The Big Meat Pizza	22968	7
Classic	The Pepperoni, Mushroom, ...	18834	8
Supreme	The Spicy Italian Pizza	34831	1
Supreme	The Italian Supreme Pizza	33477	2
Supreme	The Sicilian Pizza	30940	3

	Supreme	The Pepper Salami Pizza	25529	4
	Supreme	The Prosciutto and Arugula ...	24193	5
	Supreme	The Soppressata Pizza	16426	6
	Supreme	The Calabrese Pizza	15934	7
	Supreme	The Spinach Supreme Pizza	15278	8
	Supreme	The Brie Carre Pizza	11588	9
	Veggie	The Four Cheese Pizza	32266	1
	Veggie	The Mexicana Pizza	26781	2
	Veggie	The Five Cheese Pizza	26066	3
	Veggie	The Vegetables + Vegetable...	24375	4
	Veggie	The Spinach and Feta Pizza	23271	5
	Veggie	The Italian Vegetables Pizza	16019	6
	Veggie	The Spinach Pesto Pizza	15596	7
	Veggie	The Mediterranean Pizza	15360	8
	Veggie	The Green Garden Pizza	13956	9

Definition: For each pizza category (e.g., Classic, Supreme, Veggie, Chicken), identify the three pizza types that generated the highest revenue.

Business Insight: Reveals the “star performers” within each category, helping managers understand which pizzas drive revenue in their respective segments.

Use Case: Guides menu optimization, targeted promotions, and inventory planning at the category level.