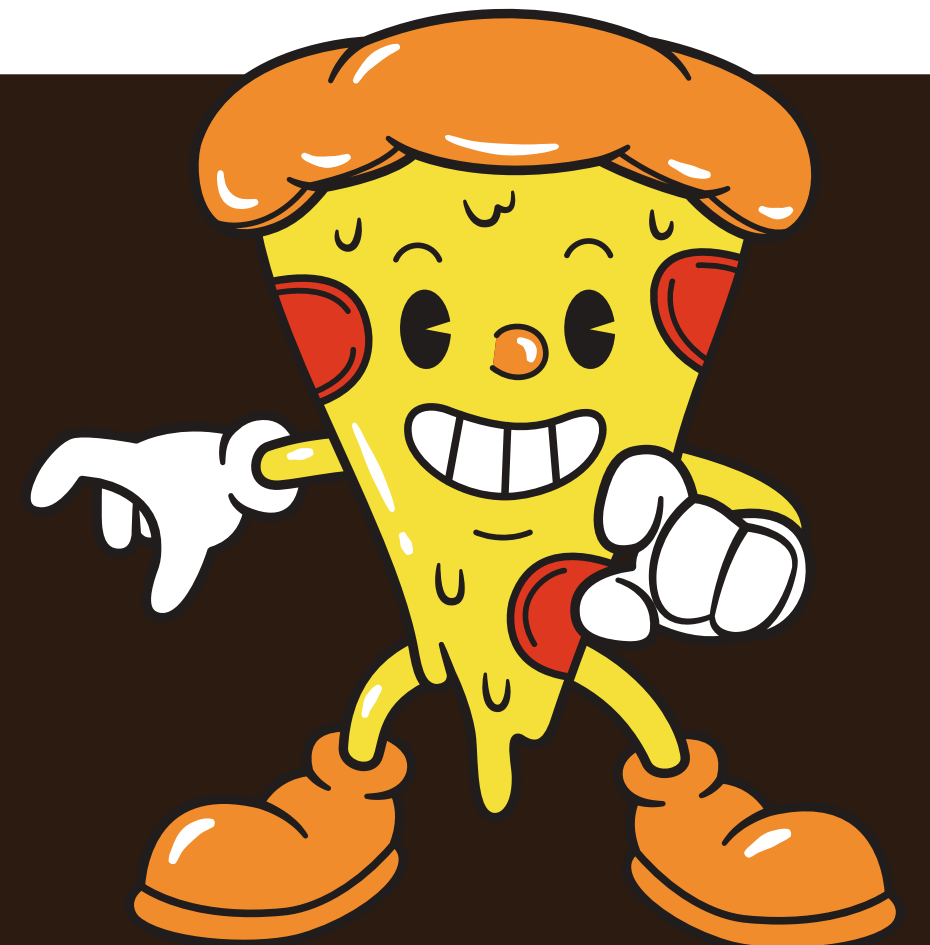


# PIZZA SALES ANALYSIS



### 🍕 Basic Analysis Question :

- Total number of orders placed 📊
- Total revenue generated from pizza sales 💰
- Highest-priced pizza identified 💵
- Most common pizza size ordered 📏
- Top 5 most ordered pizza types with quantities 📋


### 🍕 Intermediate Analysis Question:

- Joining tables to find total quantity of each pizza category ordered 🔍
- Distribution of orders by hour of the day 🕒
- Category-wise distribution of pizzas 🍕 📊
- Average number of pizzas ordered per day 📅 17
- Top 3 most ordered pizza types based on revenue 💵

### 🍕 Advanced Analysis Question:


- Percentage contribution of each pizza type to total revenue 📈
- Cumulative revenue generated over time ⌚
- Top 3 most ordered pizza types based on revenue for each pizza category 🏆 🍕

# 1: Total number of orders placed



Limit to 1000

```
20  -- Question 1
21
22  • SELECT
23      COUNT(order_id)
24  FROM
25      orders;
26
27
```



Result Grid

	COUNT(order_id)
▶	21350

## 2 : Total revenue generated from pizza sales 💰

```
29  -- Question 2
30
31 • SELECT
32     ROUND(SUM(orders_details.quantity * pizzas.price),
33           2) AS total_sales
34 FROM
35     orders_details
36     JOIN
37     pizzas ON pizzas.pizza_id = orders_details.pizza_id;
```

Result Grid	
	total_sales
▶	817860.05

# 3 : Highest-priced pizza identified \$

```
39  -- Question 3
40
41 • SELECT
42     pizza_types.name, pizzas.price
43 FROM
44     pizza_types
45     JOIN
46     pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
47 ORDER BY pizzas.price DESC
48 LIMIT 1;
49
```

Result Grid			Filter Rows
	name	price	
▶	The Greek Pizza	35.95	

# 4 : Most common pizza size ordered 🍷

```
Pizza x
-- Question 4
50
51 • SELECT
52     pizzas.size,
53     COUNT(orders_details.order_details_id) AS order_count
54 FROM
55     pizzas
56     JOIN
57     orders_details ON pizzas.pizza_id = orders_details.pizza_id
58 GROUP BY pizzas.size
59 ORDER BY order_count DESC;
60
```

Result Grid			Filter
	size	order_count	
▶	L	18526	
	M	15385	
	S	14137	
	XL	544	
	XXL	28	



# 5 : Top 5 most ordered pizza types with quantities

```
61 -- Question 5
62 • SELECT
63     pizza_types.name, SUM(orders_details.quantity) AS quantity
64 FROM
65     pizza_types
66     JOIN
67     pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
68     JOIN
69     orders_details ON orders_details.pizza_id = pizzas.pizza_id
70 GROUP BY pizza_types.name
71 ORDER BY quantity DESC
72 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	

## 6 : Joining tables to find total quantity of each pizza category ordered 🔍

```
74 -- Question 6
75 • SELECT
76     pizza_types.category,
77     SUM(orders_details.quantity) AS quantity
78 FROM
79     pizza_types
80     JOIN
81     pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
82     JOIN
83     orders_details ON orders_details.pizza_id = pizzas.pizza_id
84 GROUP BY pizza_types.category
85 ORDER BY quantity DESC;
```

Result Grid     Filter		
	category	quantity
▶	Classic	14888
	Supreme	11987
	Veggie	11649
	Chicken	11050



# 7 : Distribution of orders by hour of the day 🕒

```
86  -- Question 7
87
88  • SELECT
89      HOUR(order_time) AS hour, COUNT(order_id) AS order_count
90  FROM
91      orders
92  GROUP BY HOUR(order_time);
93
```

	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

## 8 : Category-wise distribution of pizzas 🍕 📊

```
93  -- Question 8
94
95  •  SELECT
96      category, COUNT(name)
97  FROM
98      pizza_types
99  GROUP BY category;
100
```

Result Grid   📊 ↺ Filter Rows: [		
	category	COUNT(name)
▶	Chicken	6
	Classic	8
	Supreme	9
	Veggie	9

# 9 : Average number of pizzas ordered per day

```
101  -- Question 9
102  • SELECT
103      ROUND(AVG(quantity), 0) AS Avg_pizza_ordered_per_day
104  FROM
105      (SELECT
106          orders.order_date, SUM(orders_details.quantity) AS quantity
107      FROM
108          orders
109      JOIN orders_details ON orders.order_id = orders_details.order_id
110      GROUP BY orders.order_date) AS order_quantity;
```

	Avg_pizza_ordered_per_day
▶	138

# 10 : Top 3 most ordered pizza types based on revenue

```
112 -- Question 10
113 • SELECT
114     pizza_types.name,
115     SUM(orders_details.quantity * pizzas.price) AS revenue
116 FROM
117     pizza_types
118     JOIN
119     pizzas ON pizzas.pizza_type_id = pizza_types.pizza_type_id
120     JOIN
121     orders_details ON orders_details.pizza_id = pizzas.pizza_id
122 GROUP BY pizza_types.name
123 ORDER BY revenue DESC
124 LIMIT 3;
```

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

# 11 : Percentage contribution of each pizza type to total revenue

```
126 -- Question 11
127 • SELECT
128     pizza_types.category,
129     ROUND(SUM(orders_details.quantity * pizzas.price) / (SELECT
130         ROUND(SUM(orders_details.quantity * pizzas.price), 2) AS total_sales
131     FROM
132         orders_details
133         JOIN
134         pizzas ON pizzas.pizza_id = orders_details.pizza_id) * 100, 2) AS revenue
135 FROM
136     pizza_types
137     JOIN
138     pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
139     JOIN
140     orders_details ON orders_details.pizza_id = pizzas.pizza_id
141 GROUP BY pizza_types.category
142 ORDER BY revenue DESC;
```

category	revenue
Classic	26.91
Supreme	25.46
Chicken	23.96
Veggie	23.68

# 12 : Cumulative revenue generated over time

```
144  -- Question 12
145  •  SELECT
146      order_date,
147      SUM(revenue) OVER (ORDER BY order_date) AS cum_revenue
148  FROM
149  (SELECT
150      orders.order_date,
151      SUM(orders_details.quantity * pizzas.price) AS revenue
152  FROM
153      orders_details
154  JOIN
155      pizzas ON orders_details.pizza_id = pizzas.pizza_id
156  JOIN
157      orders ON orders.order_id = orders_details.order_id
158  GROUP BY
159      orders.order_date) AS sales;
```

order_date	cum_revenue
2015-01-01 00:00:00	2713.85000000000004
2015-01-02 00:00:00	5445.75
2015-01-03 00:00:00	8108.15
2015-01-04 00:00:00	9863.6
2015-01-05 00:00:00	11929.55
2015-01-06 00:00:00	14358.5

# 13 : Top most ordered pizza types based on revenue for each pizza category 🏆🍕

```
161 -- Question 13
162 • SELECT name, revenue
163 FROM
164     (SELECT category, name, revenue,
165          RANK() OVER (PARTITION BY category ORDER BY revenue DESC) AS rn
166     FROM
167         (SELECT
168             pizza_types.category,
169             pizza_types.name,
170             SUM((orders_details.quantity) * pizzas.price) AS revenue
171         FROM pizza_types
172         JOIN pizzas ON pizza_types.pizza_type_id = pizzas.pizza_type_id
173         JOIN orders_details ON orders_details.pizza_id = pizzas.pizza_id
174         GROUP BY
175             pizza_types.category, pizza_types.name) AS a) AS b
176 WHERE rn <= 3;
177
```

name	revenue
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5
The Classic Deluxe Pizza	38180.5
The Hawaiian Pizza	32273.25
The Pepperoni Pizza	30161.75