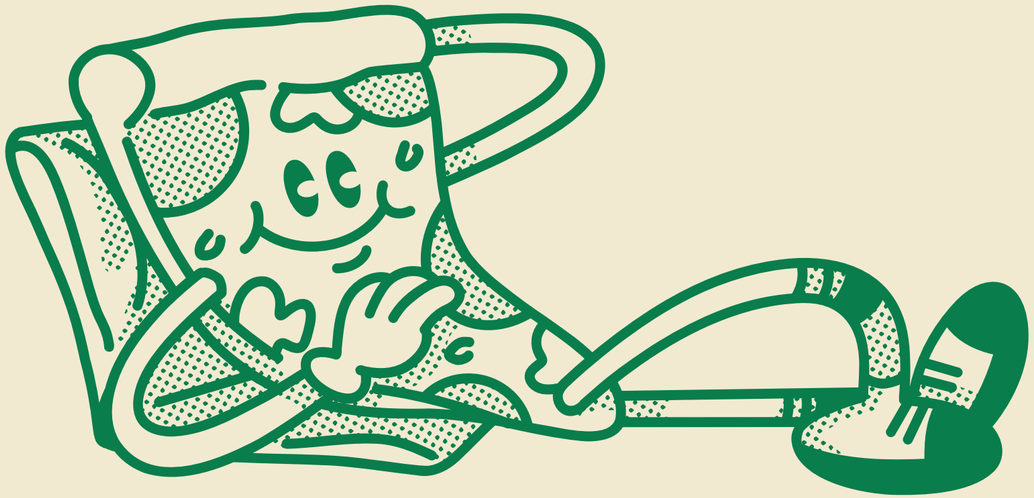
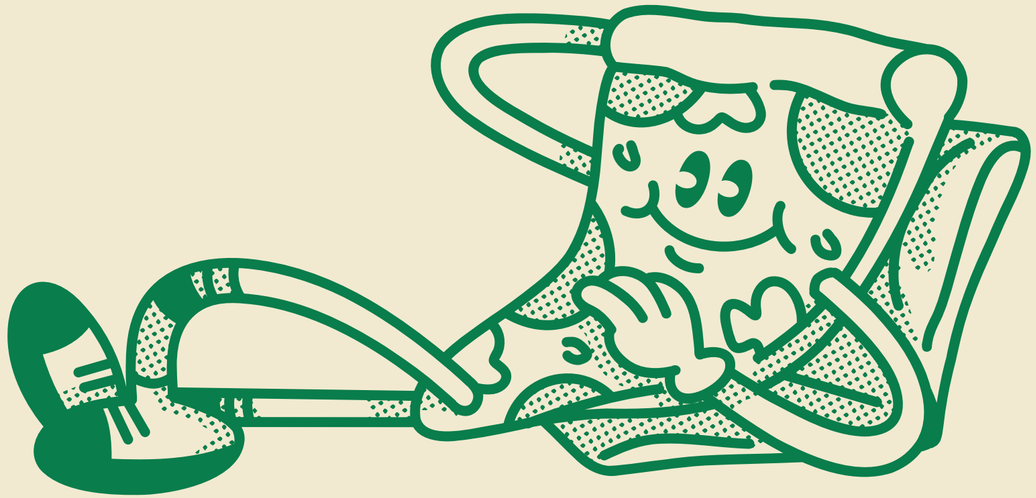
PIZZA SALES

DATASET ANALYSIS MYSQL PROJECT

- SAMRUDDHI SAVALE



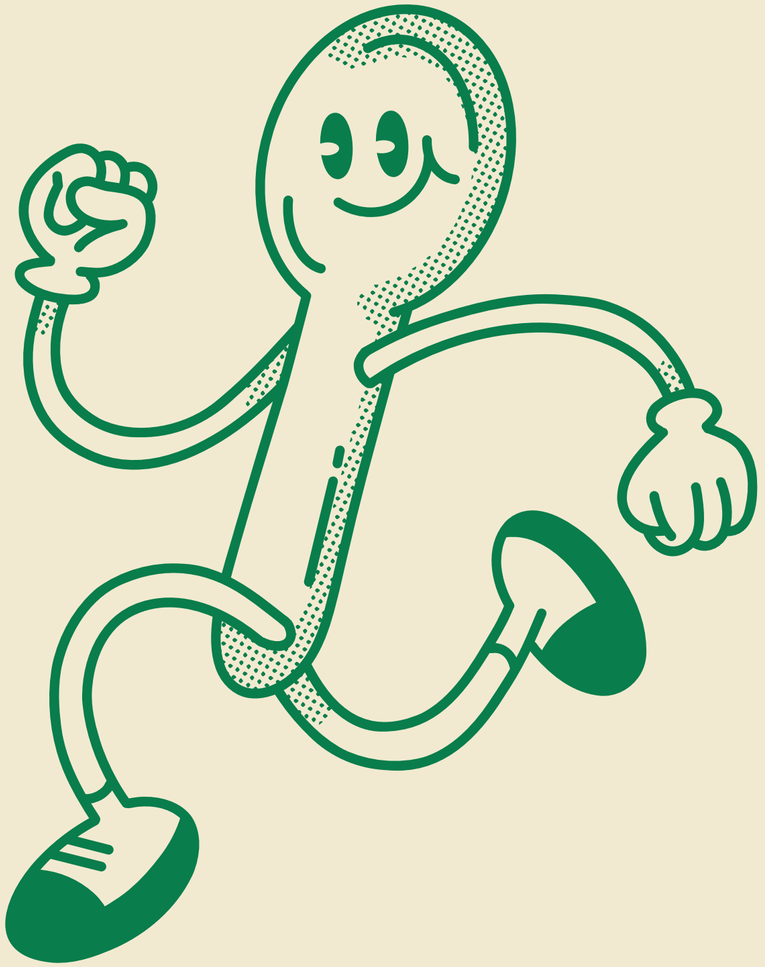


Objectives

TO TRANSFORM RAW TRANSACTION DATA INTO ACTIONABLE BUSINESS INTELLIGENCE, DRIVING STRATEGIC DECISION-MAKING FOR REVENUE GROWTH AND OPERATIONAL EFFICIENCY.

Introduction

- **REVENUE & VOLUME:** QUANTIFIED TOTAL SALES VOLUME AND REVENUE, ESTABLISHING A BASELINE FOR FINANCIAL PERFORMANCE.
- **PRODUCT PERFORMANCE:** IDENTIFIED TOP-SELLING AND HIGHEST-REVENUE PIZZA TYPES, HIGHLIGHTING MENU WINNERS AND OPPORTUNITIES FOR PRICING STRATEGY.
- **CUSTOMER PATTERNS:** MAPPED ORDER DISTRIBUTION TO HOURLY TRENDS, PINPOINTING PEAK DEMAND PERIODS FOR OPTIMIZED STAFFING AND MARKETING.
- **OPERATIONAL BENCHMARK:** CALCULATED THE AVERAGE DAILY PIZZA OUTPUT, PROVIDING A CRITICAL METRIC FOR INVENTORY FORECASTING AND SUPPLY CHAIN MANAGEMENT.



Retrieve the total number of orders placed.

```sql
SELECT COUNT(ORDER_ID) AS TOTAL_ORDER
FROM ORDERS;
```

Result Grid | Filter

	TOTAL_ORDER
▶	21350

Calculate the total revenue generated from pizza sales.

```sql
SELECT ROUND(SUM(OD.QUANTITY * P.PRICE),2) AS TOTAL_SALES
FROM ORDER_DETAILS AS OD
JOIN PIZZAS AS P
ON P.PIZZA_ID = OD.PIZZA_ID;
```

Result Grid

	TOTAL_SALES
▶	817860.05

Identify the highest-priced pizza.

```sql
SELECT PT.NAME,P.PRICE
FROM PIZZA_TYPES AS PT
JOIN PIZZAS AS P
ON PT.PIZZA_TYPE_ID = P.PIZZA_TYPE_ID
ORDER BY P.PRICE DESC
LIMIT 1;
```

Result Grid | Filter Rows:

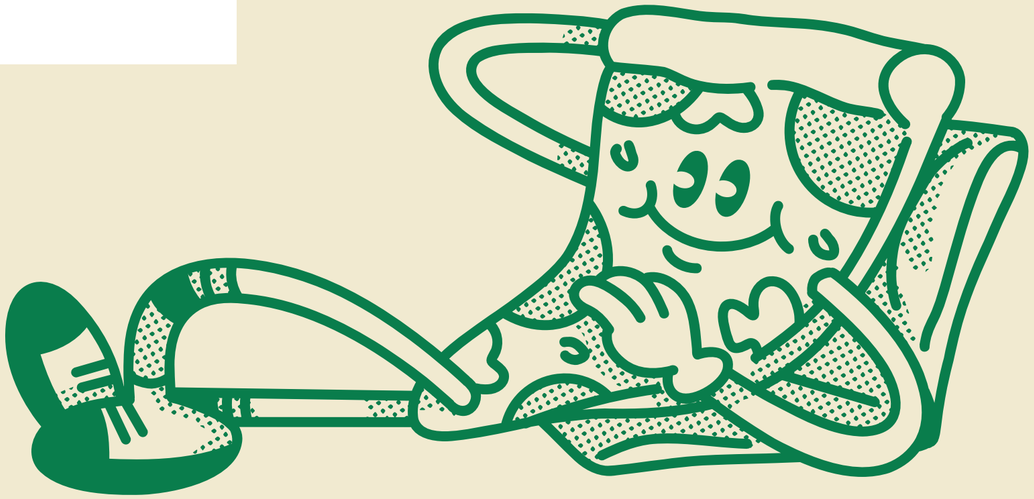
NAME	PRICE
The Greek Pizza	35.95

Identify the most common pizza size ordered.

```sql
SELECT P.SIZE, COUNT(OD.ORDER_DETAIL_ID) AS ORDER_COUNT
FROM PIZZAS AS P
JOIN ORDER_DETAILS AS OD
ON P.PIZZA_ID = OD.PIZZA_ID
GROUP BY P.SIZE
ORDER BY ORDER_COUNT DESC;
```

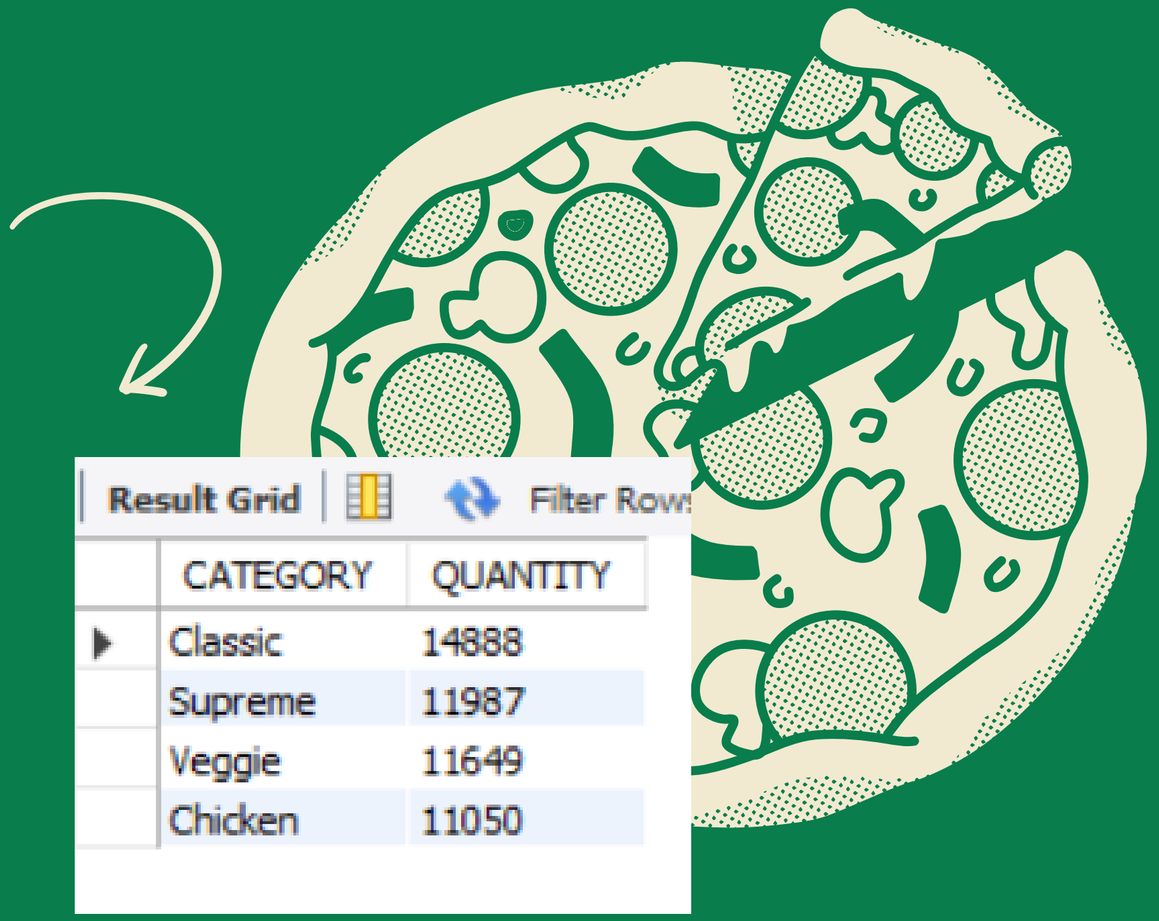
Result Grid | Filter Rows:

SIZE	ORDER_COUNT
L	18526
M	15385
S	14137
XL	544
XXL	28



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

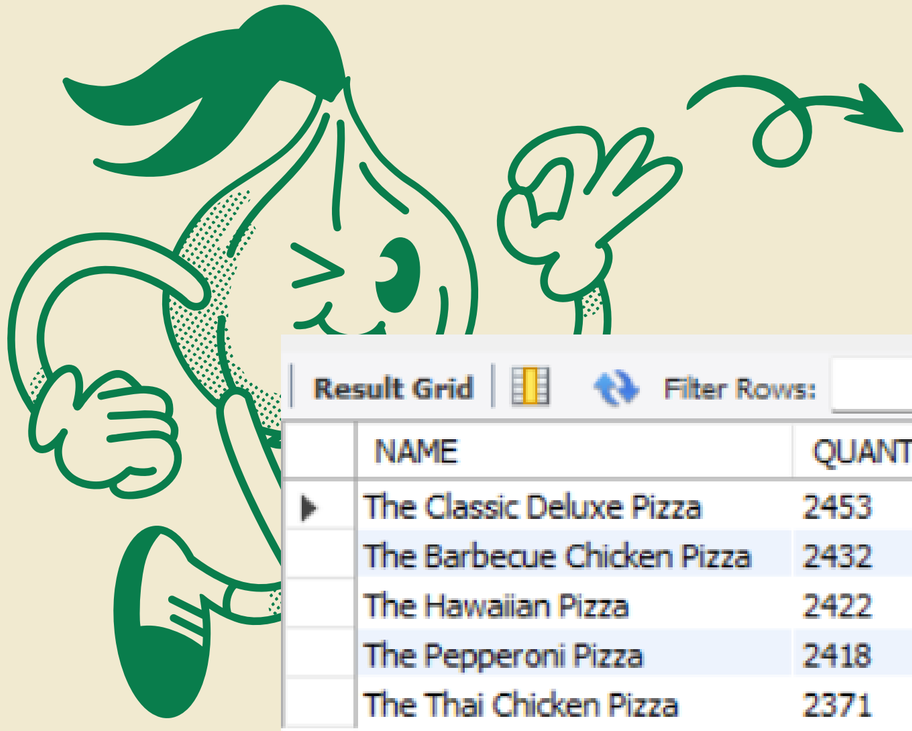
```sql
SELECT PIZZA_TYPES.CATEGORY,
SUM(ORDER_DETAILS.QUANTITY) AS QUANTITY
FROM PIZZA_TYPES
JOIN PIZZAS
ON PIZZA_TYPES.PIZZA_TYPE_ID = PIZZAS.PIZZA_TYPE_ID
JOIN ORDER_DETAILS
ON ORDER_DETAILS.PIZZA_ID = PIZZAS.PIZZA_ID
GROUP BY PIZZA_TYPES.CATEGORY
ORDER BY QUANTITY DESC;
```



Result Grid | Filter Rows

CATEGORY	QUANTITY
Classic	14888
Supreme	11987
Veggie	11649
Chicken	11050

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



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

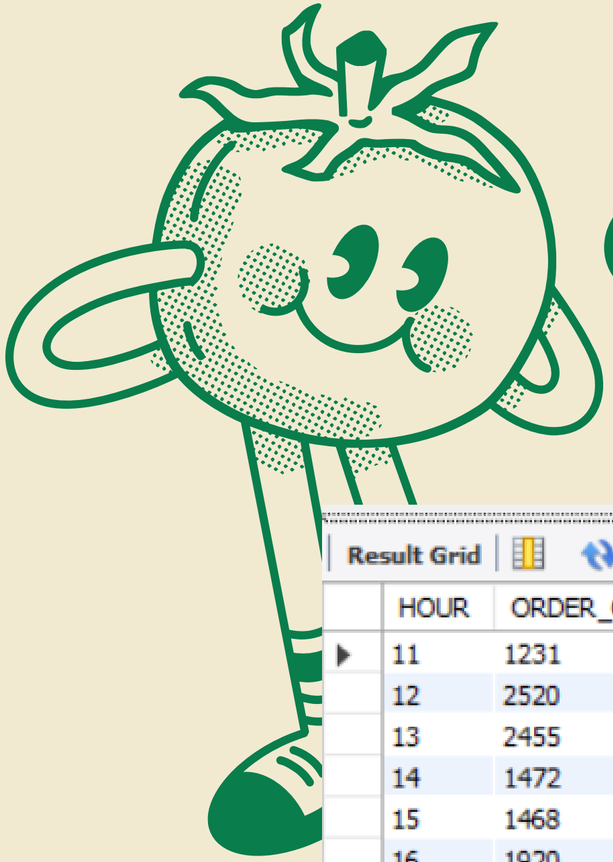
```sql
SELECT PIZZA_TYPES.NAME,
SUM(ORDER_DETAILS.QUANTITY) AS QUANTITY
FROM PIZZA_TYPES
JOIN PIZZAS
ON PIZZA_TYPES. PIZZA_TYPE_ID = PIZZAS. PIZZA_TYPE_ID
JOIN ORDER_DETAILS
ON ORDER_DETAILS.PIZZA_ID = PIZZAS.PIZZA_ID
GROUP BY PIZZA_TYPES.NAME
ORDER BY QUANTITY DESC
LIMIT 5;
```

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

```sql
SELECT ROUND(AVG(QUANTITY),0) AS AVG_PIZZA_ORDER_PER_DAY
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:

AVG_PIZZA_ORDER_PER_DAY
138



Determine the distribution of orders by hour of the day.

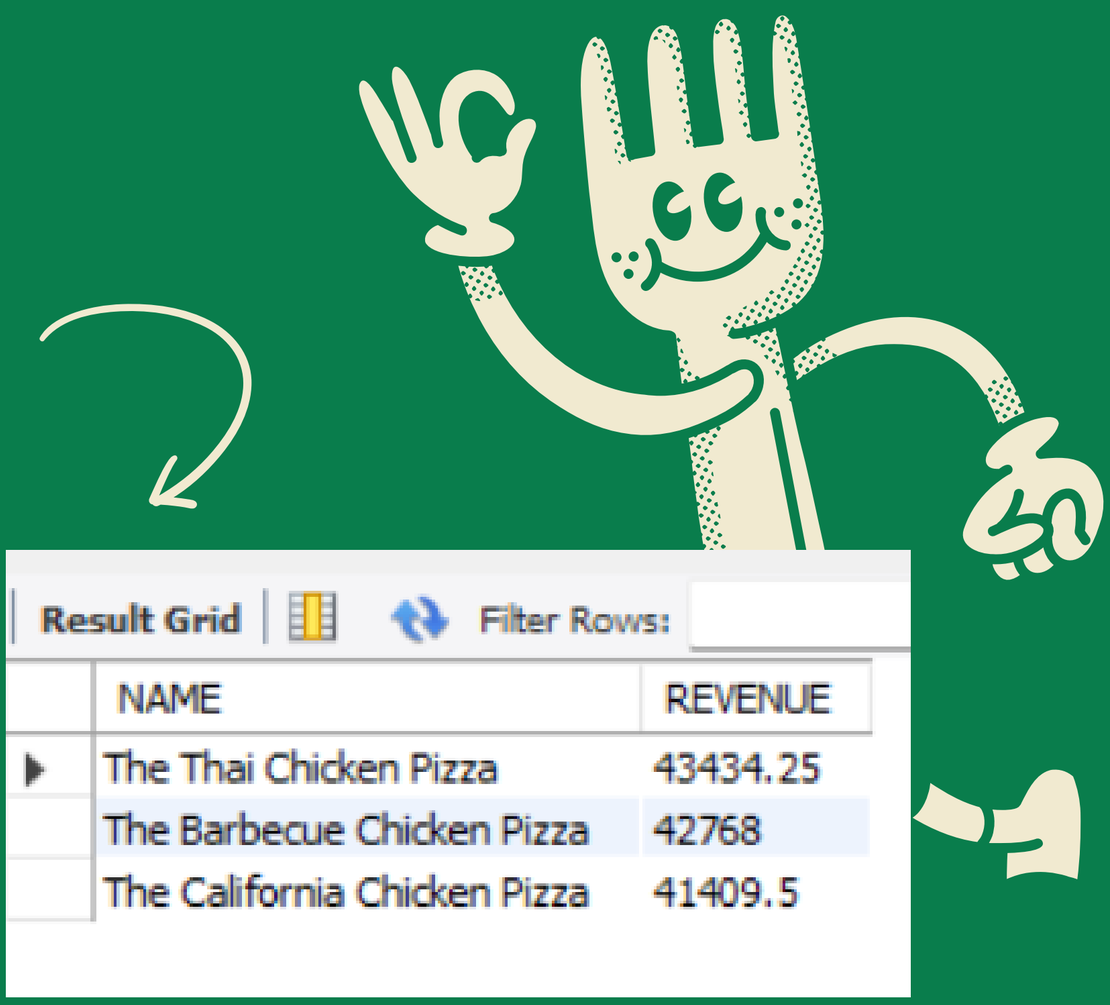
Result Grid | Filter Rows:

HOUR	ORDER_OF_COUNTS
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

```sql
SELECT HOUR(ORDER_TIME) AS HOUR, COUNT(ORDER_ID) AS ORDER_OF_COUNTS
FROM ORDERS
GROUP BY HOUR(ORDER_TIME);
```

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 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 REVENUE DESC
LIMIT 3;
```



Result Grid | Filter Rows:

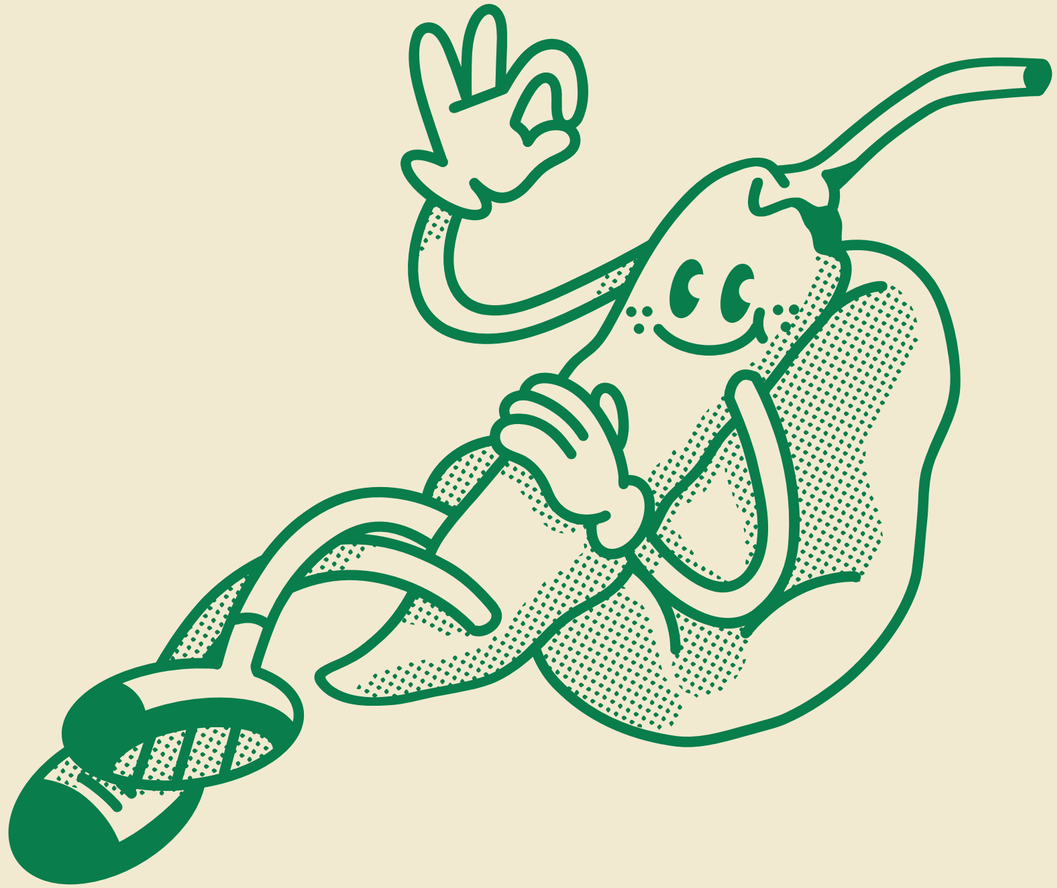
NAME	REVENUE
The Thai Chicken Pizza	43434.25
The Barbecue Chicken Pizza	42768
The California Chicken Pizza	41409.5

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



THANK YOU