The Great Pizza Analytics Challenge

Transforming IDC Pizza's raw sales data into actionable business insights using SQL

- Pratik Mendon

Project Overview

Explore Database

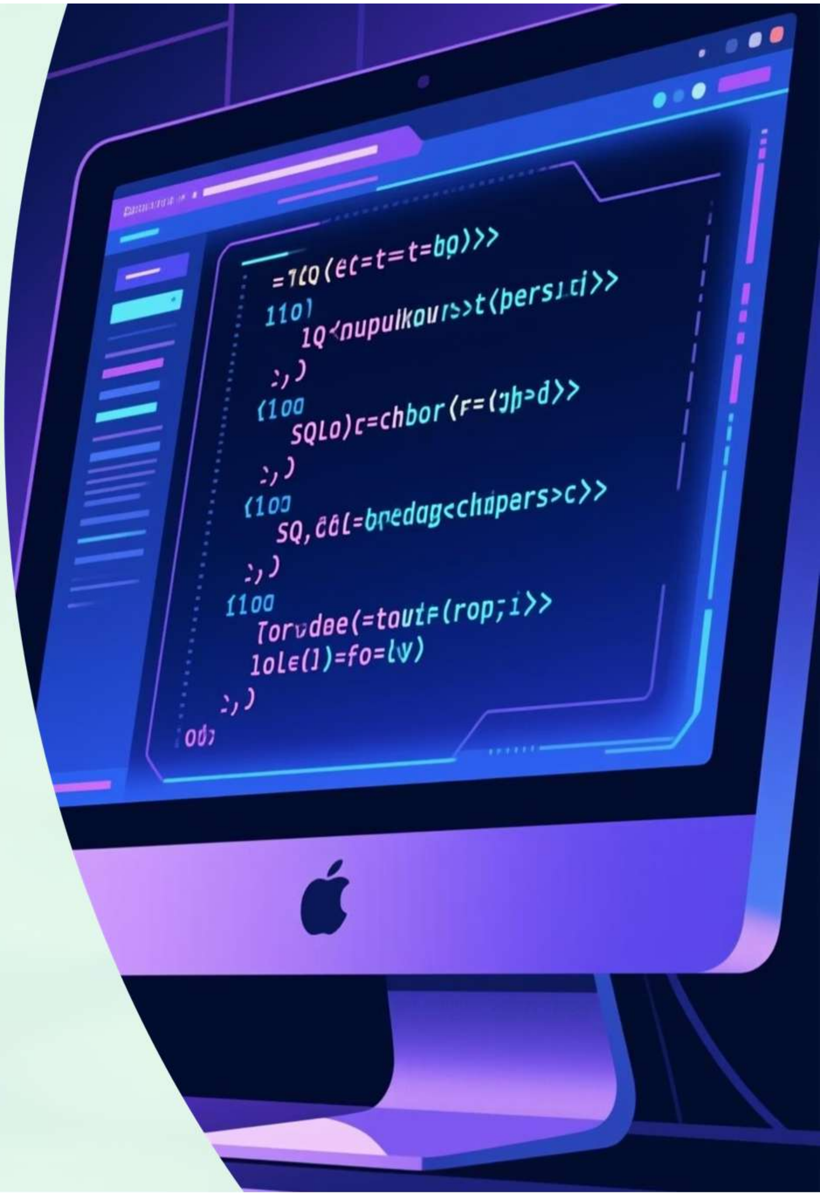
Navigate IDC Pizza's relational database structure and understand table relationships

Apply SQL Techniques

Use filtering, aggregation, joins, and advanced queries to extract insights

Derive Insights

Transform raw sales data into business intelligence for strategic decision-making



Dataset Overview

IDC Pizza's relational database consists of four interconnected tables that capture the complete order and product ecosystem:

1

orders

order_id, date, time

2

order_details

order_id, pizza_id, quantity

3

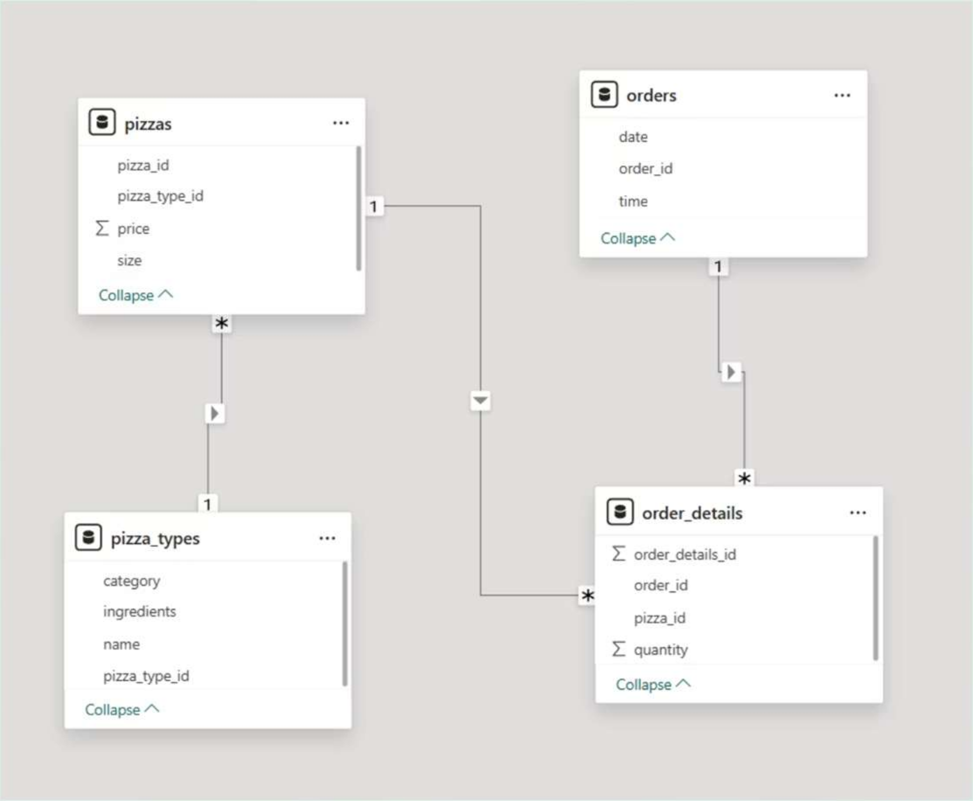
pizzas

pizza_id, pizza_type_id, size, price

4

pizza_types

pizza_type_id, name, category, ingredients



Skills Demonstrated

Throughout this 15-day SQL challenge, we applied a comprehensive range of database querying techniques:

Core Queries

SELECT, WHERE, DISTINCT, LIMIT, OFFSET for data retrieval and filtering

Aggregations

SUM, AVG, COUNT, MIN, MAX with GROUP BY and HAVING clauses

Table Joins

INNER, LEFT, RIGHT, FULL joins to combine related datasets

Data Manipulation

String functions, date operations, NULL handling with COALESCE

Conditional Logic

CASE WHEN statements for dynamic categorization and calculations

Pattern Matching

Advanced filtering techniques and text pattern recognition

Phase 1: Foundation & Inspection

01

Database Installation

Set up IDC_Pizza database and verify table structures

02

Categorical Analysis

List unique pizza categories and understand product segmentation

03

Data Quality Checks

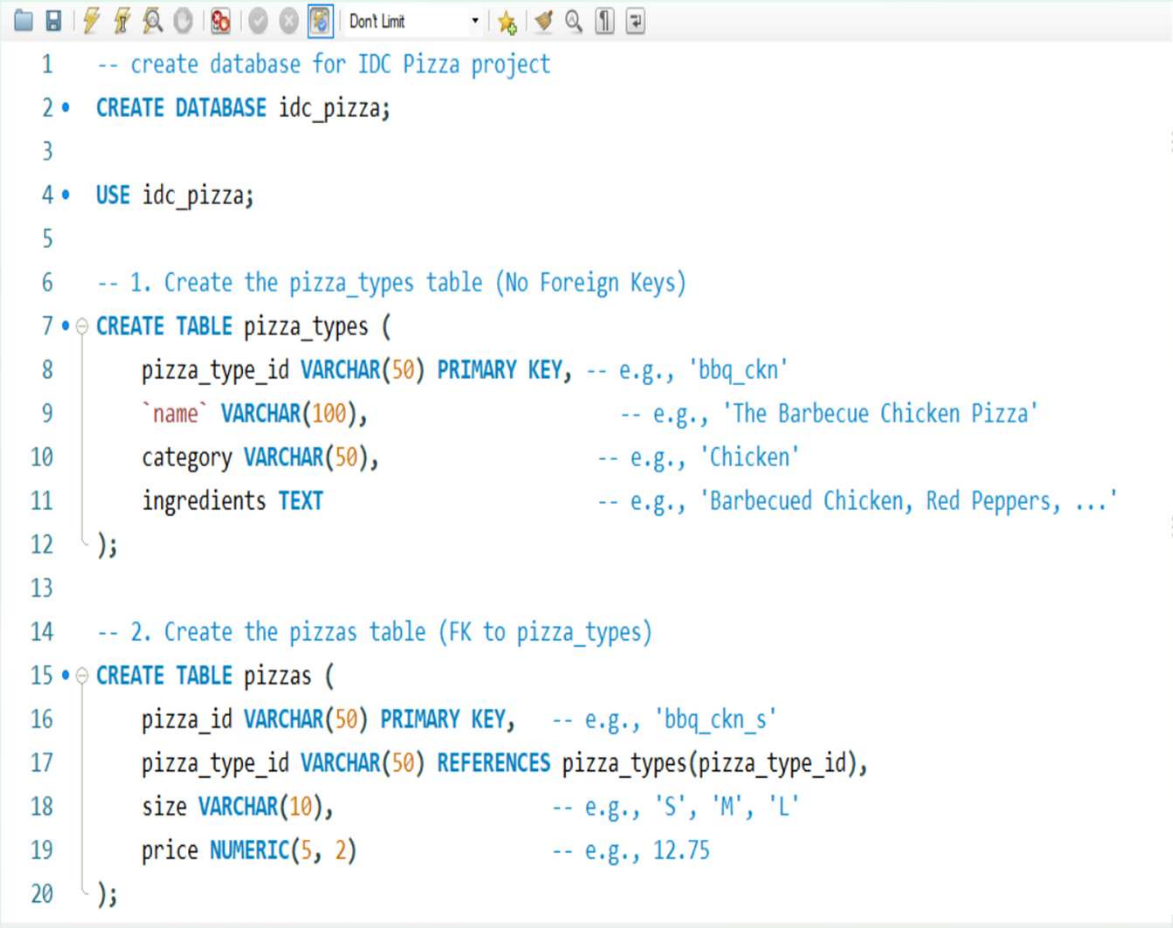
Identify NULL ingredients and pizzas missing price information

04

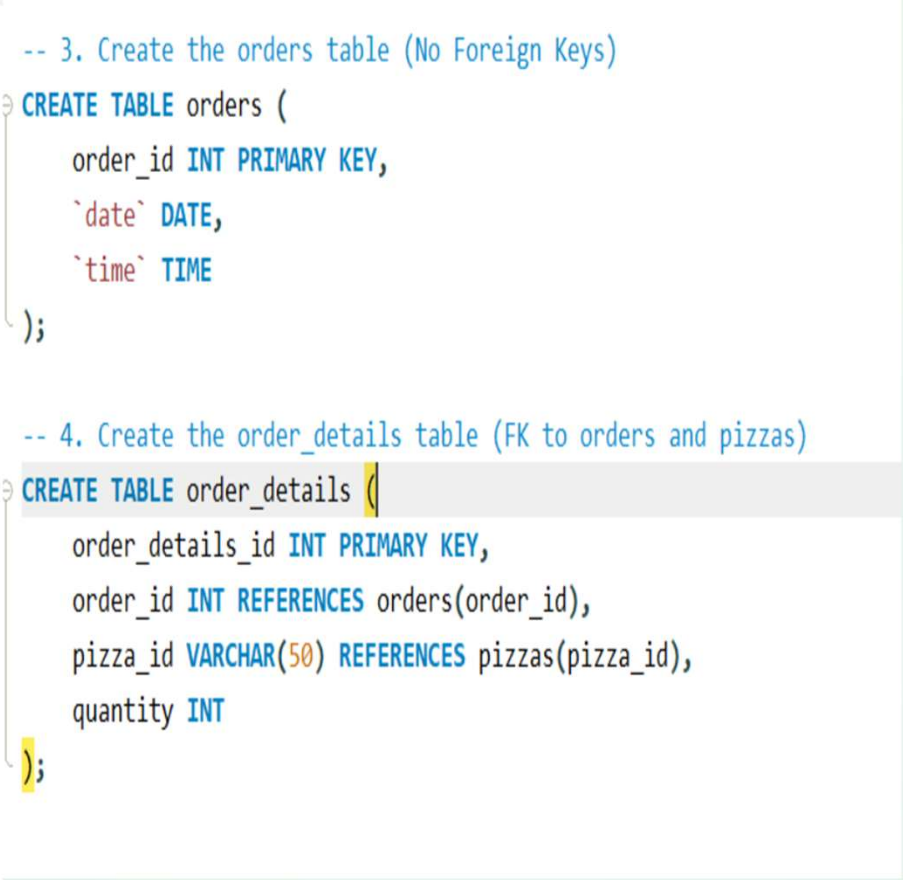
Initial Data Review

Display and examine first rows with cleaned, standardized data

1. Database Installation

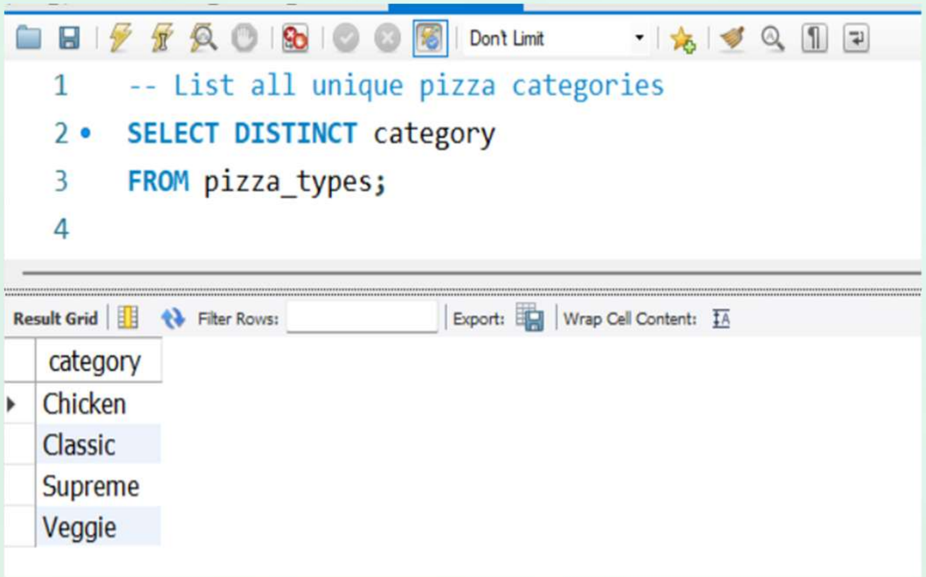


```sql
-- create database for IDC Pizza project
CREATE DATABASE idc_pizza;

USE idc_pizza;

-- 1. Create the pizza_types table (No Foreign Keys)
CREATE TABLE pizza_types (
    pizza_type_id VARCHAR(50) PRIMARY KEY, -- e.g., 'bbq_ckn'
    `name` VARCHAR(100),                   -- e.g., 'The Barbecue Chicken Pizza'
    category VARCHAR(50),                 -- e.g., 'Chicken'
    ingredients TEXT                      -- e.g., 'Barbecued Chicken, Red Peppers, ...'
);

-- 2. Create the pizzas table (FK to pizza_types)
CREATE TABLE pizzas (
    pizza_id VARCHAR(50) PRIMARY KEY,    -- e.g., 'bbq_ckn_s'
    pizza_type_id VARCHAR(50) REFERENCES pizza_types(pizza_type_id),
    size VARCHAR(10),                    -- e.g., 'S', 'M', 'L'
    price NUMERIC(5, 2)                  -- e.g., 12.75
);
```



```sql
-- 3. Create the orders table (No Foreign Keys)
CREATE TABLE orders (
    order_id INT PRIMARY KEY,
    `date` DATE,
    `time` TIME
);

-- 4. Create the order_details table (FK to orders and pizzas)
CREATE TABLE order_details (
    order_details_id INT PRIMARY KEY,
    order_id INT REFERENCES orders(order_id),
    pizza_id VARCHAR(50) REFERENCES pizzas(pizza_id),
    quantity INT
);
```

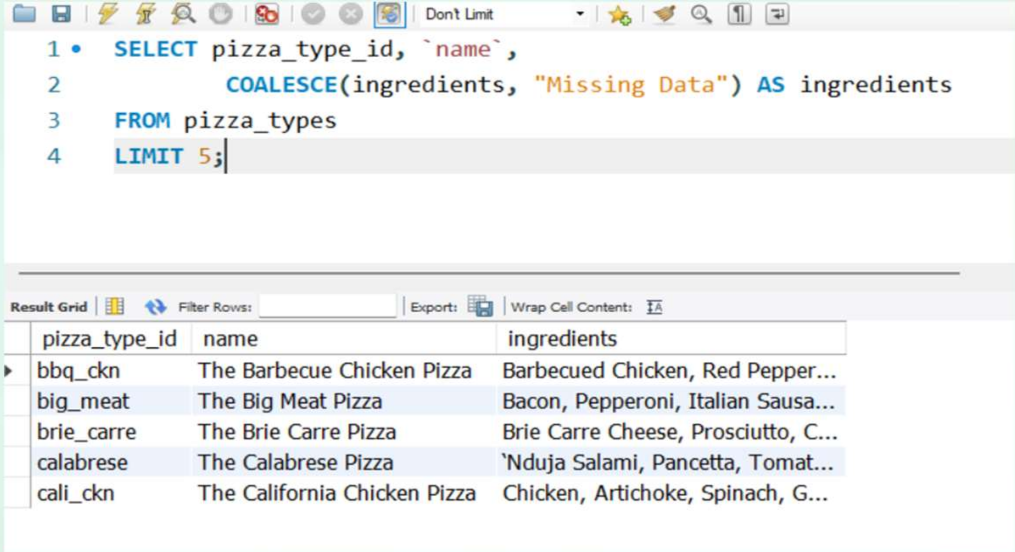
2. List all unique pizza categories



```sql
-- List all unique pizza categories
SELECT DISTINCT category
FROM pizza_types;
```

category
Chicken
Classic
Supreme
Veggie

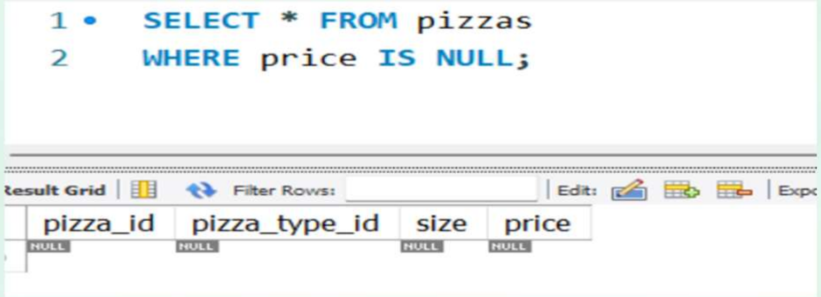
3. Display pizza_type_id, name, and ingredients, replacing NULL ingredients with "Missing Data". Show first 5 rows.



```sql
SELECT pizza_type_id, `name`,
       COALESCE(ingredients, "Missing Data") AS ingredients
FROM pizza_types
LIMIT 5;
```

pizza_type_id	name	ingredients
bbq_ckn	The Barbecue Chicken Pizza	Barbecued Chicken, Red Pepper...
big_meat	The Big Meat Pizza	Bacon, Pepperoni, Italian Sausa...
brie_carre	The Brie Carre Pizza	Brie Carre Cheese, Prosciutto, C...
calabrese	The Calabrese Pizza	'Nduja Salami, Pancetta, Tomat...
cali_ckn	The California Chicken Pizza	Chicken, Artichoke, Spinach, G...

4. Check for pizzas missing a price >



```sql
SELECT * FROM pizzas
WHERE price IS NULL;
```

pizza_id	pizza_type_id	size	price
NULL	NULL	NULL	NULL

Phase 2: Filtering & Exploration

Applied advanced filtering techniques to slice and dice the pizza sales data:

→ **Date-Based Filtering**

Extract orders from specific dates to analyze daily patterns

→ **Size & Price Ranges**

Filter pizzas by size (L, XL) and price brackets for targeted analysis

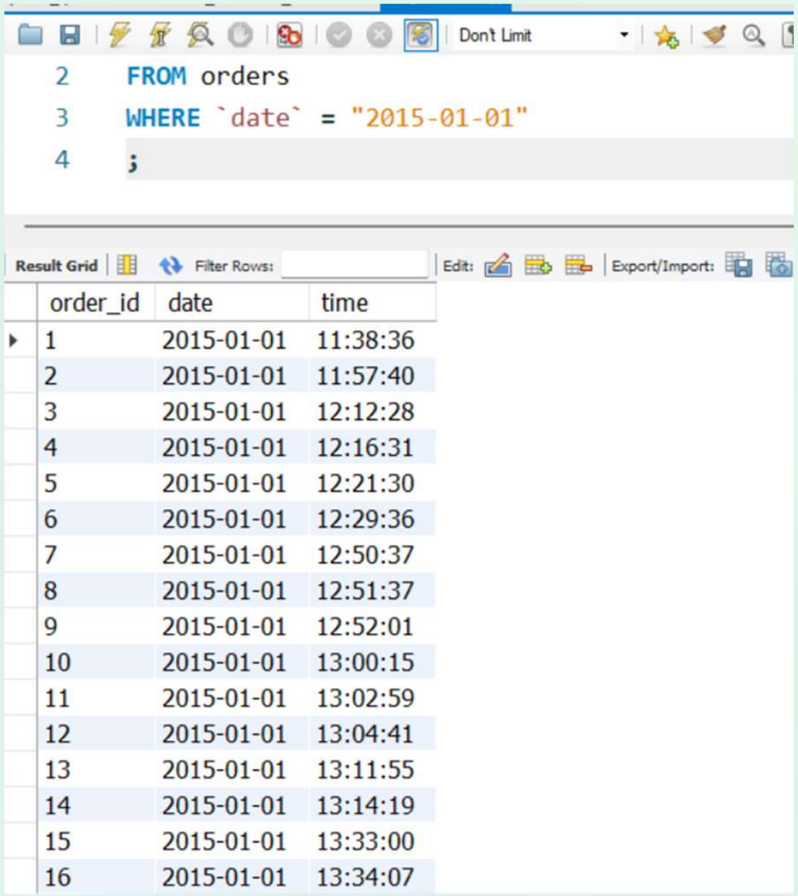
→ **Keyword Searches**

Identify pizzas containing specific ingredients like "Chicken"

→ **Custom Sorting**

Order results by price, quantity, or other metrics for insight generation

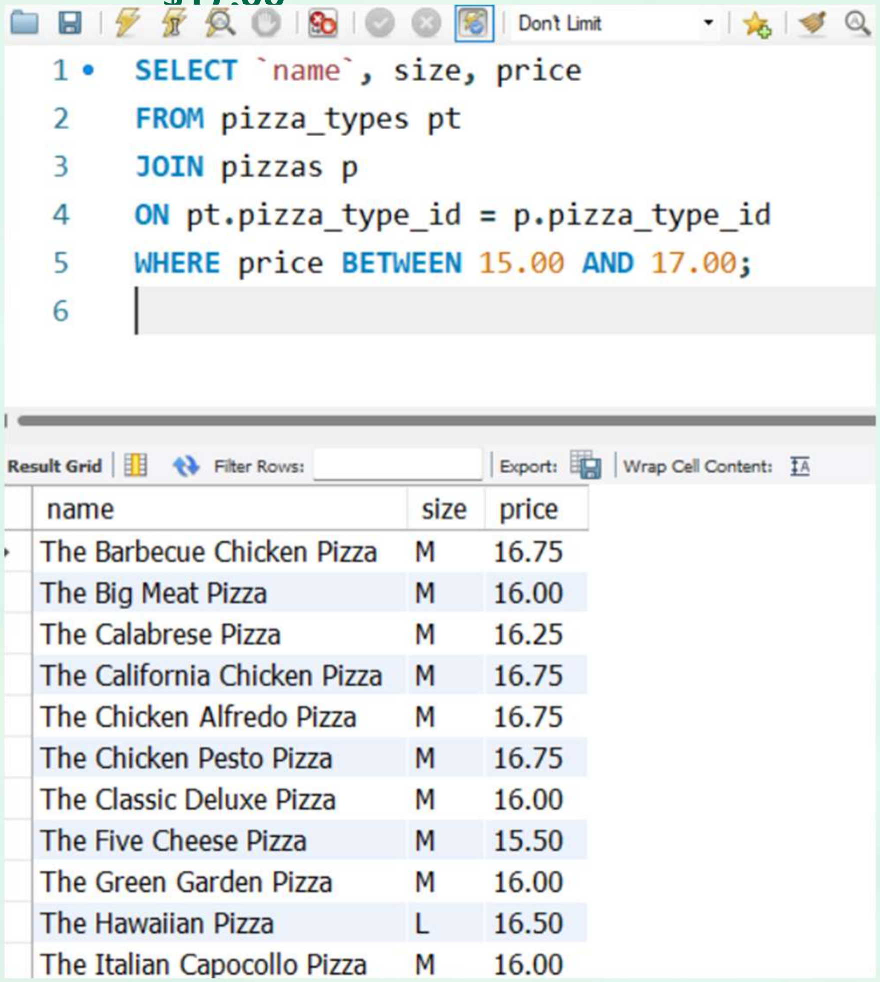
5. Orders placed on '2015-01-01'



```
FROM orders
WHERE `date` = "2015-01-01"
;
```

order_id	date	time
1	2015-01-01	11:38:36
2	2015-01-01	11:57:40
3	2015-01-01	12:12:28
4	2015-01-01	12:16:31
5	2015-01-01	12:21:30
6	2015-01-01	12:29:36
7	2015-01-01	12:50:37
8	2015-01-01	12:51:37
9	2015-01-01	12:52:01
10	2015-01-01	13:00:15
11	2015-01-01	13:02:59
12	2015-01-01	13:04:41
13	2015-01-01	13:11:55
14	2015-01-01	13:14:19
15	2015-01-01	13:33:00
16	2015-01-01	13:34:07

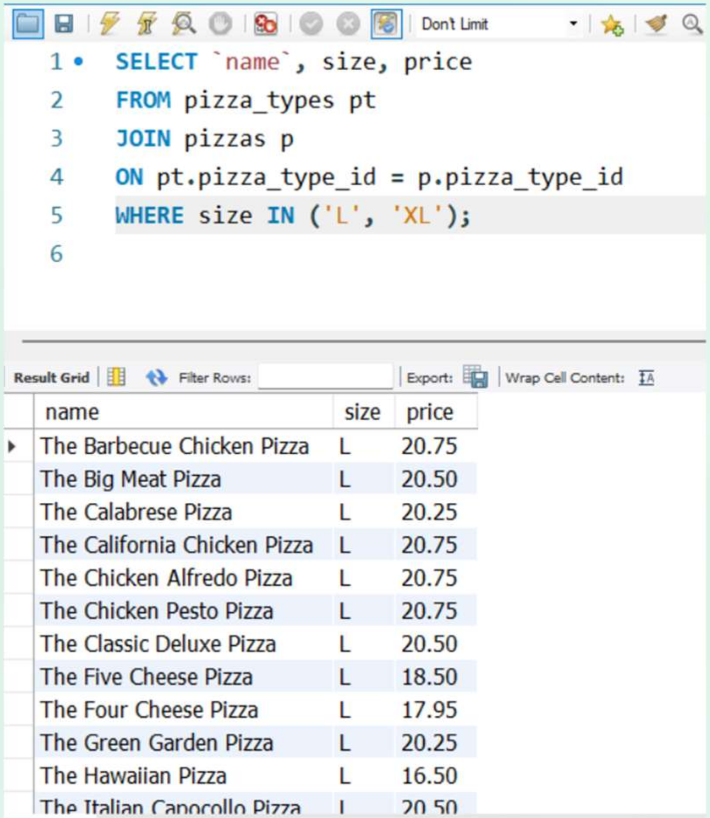
6. Pizzas priced between $15.00 and $17.00



```
SELECT `name`, size, price
FROM pizza_types pt
JOIN pizzas p
ON pt.pizza_type_id = p.pizza_type_id
WHERE price BETWEEN 15.00 AND 17.00;
```

name	size	price
The Barbecue Chicken Pizza	M	16.75
The Big Meat Pizza	M	16.00
The Calabrese Pizza	M	16.25
The California Chicken Pizza	M	16.75
The Chicken Alfredo Pizza	M	16.75
The Chicken Pesto Pizza	M	16.75
The Classic Deluxe Pizza	M	16.00
The Five Cheese Pizza	M	15.50
The Green Garden Pizza	M	16.00
The Hawaiian Pizza	L	16.50
The Italian Capocollo Pizza	M	16.00

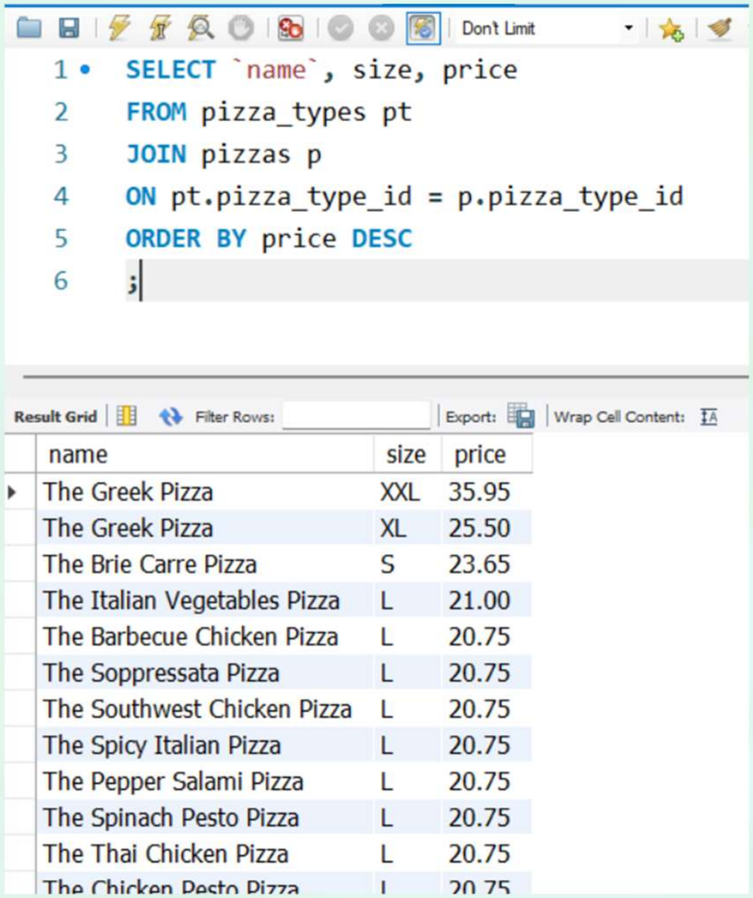
7. Pizzas sold in L' or XL



```sql
SELECT `name`, size, price
FROM pizza_types pt
JOIN pizzas p
ON pt.pizza_type_id = p.pizza_type_id
WHERE size IN ('L', 'XL');

```

name	size	price
The Barbecue Chicken Pizza	L	20.75
The Big Meat Pizza	L	20.50
The Calabrese Pizza	L	20.25
The California Chicken Pizza	L	20.75
The Chicken Alfredo Pizza	L	20.75
The Chicken Pesto Pizza	L	20.75
The Classic Deluxe Pizza	L	20.50
The Five Cheese Pizza	L	18.50
The Four Cheese Pizza	L	17.95
The Green Garden Pizza	L	20.25
The Hawaiian Pizza	L	16.50
The Italian Capocollo Pizza	L	20.50

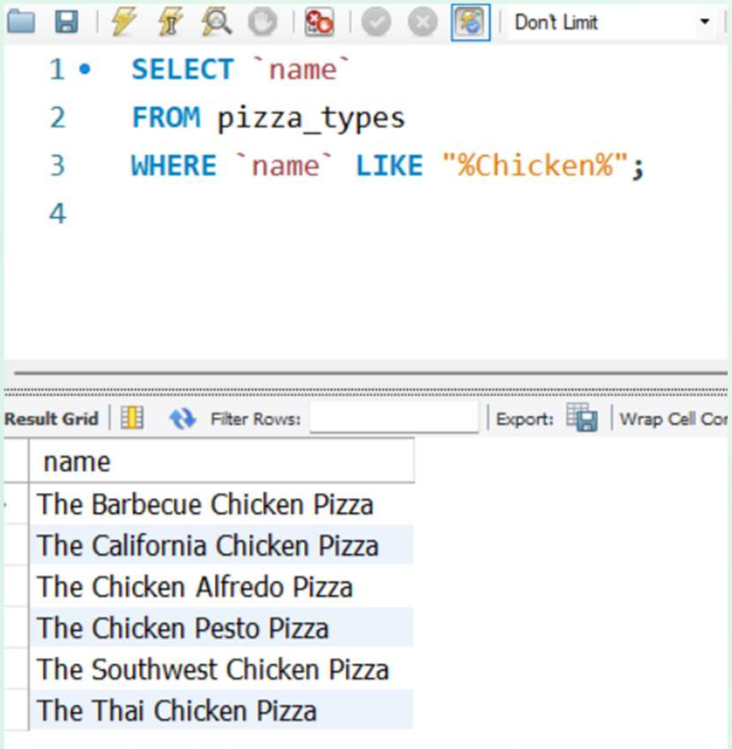
8. List pizzas with price descending



```sql
SELECT `name`, size, price
FROM pizza_types pt
JOIN pizzas p
ON pt.pizza_type_id = p.pizza_type_id
ORDER BY price DESC
;
```

name	size	price
The Greek Pizza	XXL	35.95
The Greek Pizza	XL	25.50
The Brie Carre Pizza	S	23.65
The Italian Vegetables Pizza	L	21.00
The Barbecue Chicken Pizza	L	20.75
The Soppressata Pizza	L	20.75
The Southwest Chicken Pizza	L	20.75
The Spicy Italian Pizza	L	20.75
The Pepper Salami Pizza	L	20.75
The Spinach Pesto Pizza	L	20.75
The Thai Chicken Pizza	L	20.75
The Chicken Pesto Pizza	L	20.75

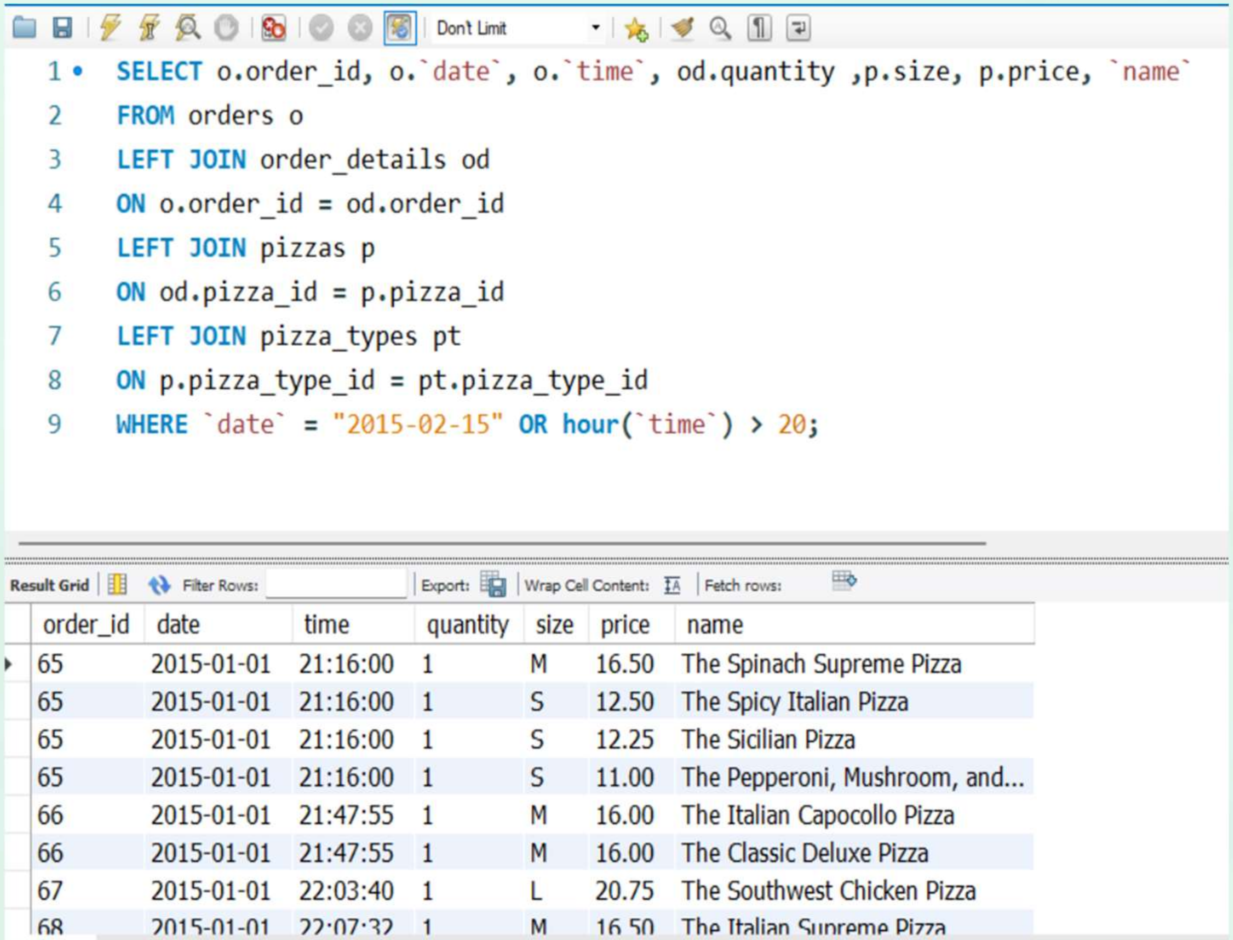
9. Pizzas with Chicken” in name



```sql
SELECT `name`
FROM pizza_types
WHERE `name` LIKE "%Chicken%";
```

name
The Barbecue Chicken Pizza
The California Chicken Pizza
The Chicken Alfredo Pizza
The Chicken Pesto Pizza
The Southwest Chicken Pizza
The Thai Chicken Pizza

10. Orders on '2015-02-15'or placed after 8 PM.



```sql
SELECT o.order_id, o.`date`, o.`time`, od.quantity ,p.size, p.price, `name`
FROM orders o
LEFT JOIN order_details od
ON o.order_id = od.order_id
LEFT JOIN pizzas p
ON od.pizza_id = p.pizza_id
LEFT JOIN pizza_types pt
ON p.pizza_type_id = pt.pizza_type_id
WHERE `date` = "2015-02-15" OR hour(`time`) > 20;
```

order_id	date	time	quantity	size	price	name
65	2015-01-01	21:16:00	1	M	16.50	The Spinach Supreme Pizza
65	2015-01-01	21:16:00	1	S	12.50	The Spicy Italian Pizza
65	2015-01-01	21:16:00	1	S	12.25	The Sicilian Pizza
65	2015-01-01	21:16:00	1	S	11.00	The Pepperoni, Mushroom, and...
66	2015-01-01	21:47:55	1	M	16.00	The Italian Capocollo Pizza
66	2015-01-01	21:47:55	1	M	16.00	The Classic Deluxe Pizza
67	2015-01-01	22:03:40	1	L	20.75	The Southwest Chicken Pizza
68	2015-01-01	22:07:32	1	M	16.50	The Italian Supreme Pizza

Phase 3: Sales Performance Analysis

Volume Metrics

Total pizzas sold and average price calculations across all orders

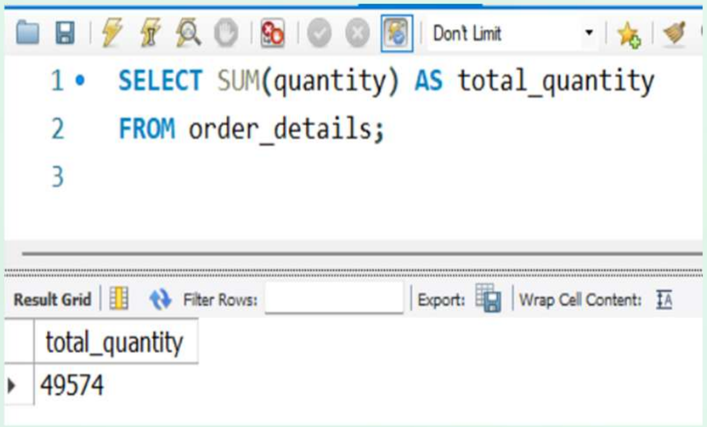
Revenue Analysis

Order-level value computation and category performance tracking

Product Performance

Identify top performers, zero-sale items, and pricing differences by size

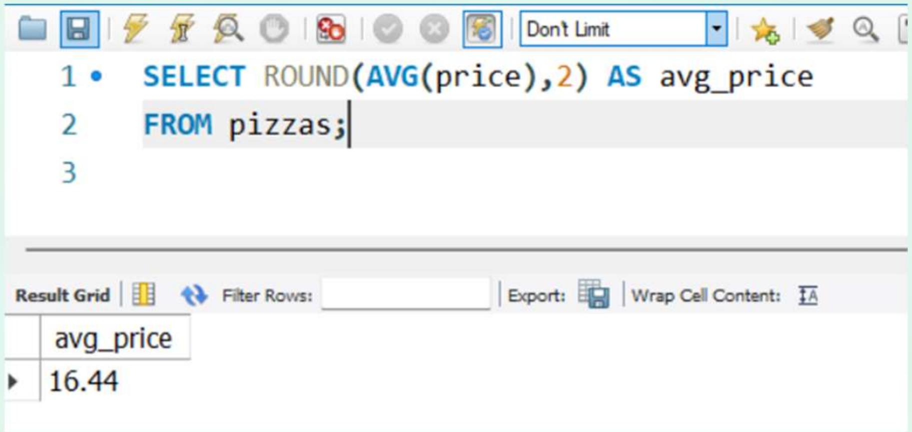
11. Total Quantity of Pizzas sold



```sql
SELECT SUM(quantity) AS total_quantity
FROM order_details;
```

```
total_quantity
49574
```

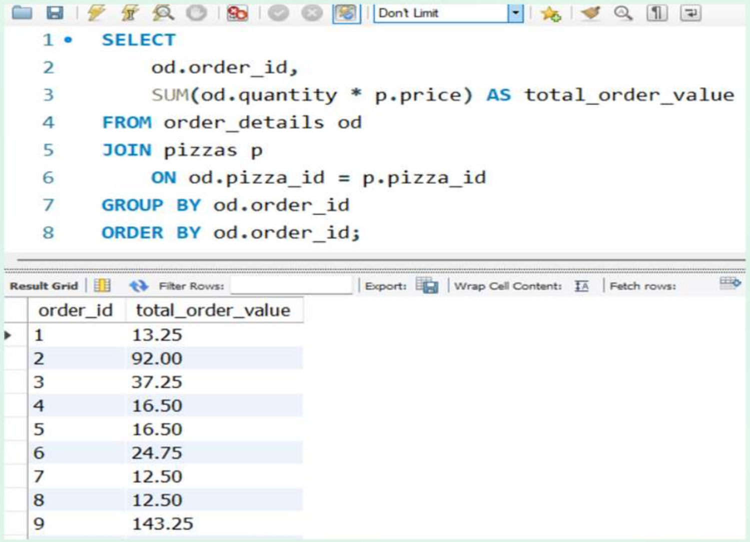
12. Average Pizza Price



```sql
SELECT ROUND(AVG(price),2) AS avg_price
FROM pizzas;
```

```
avg_price
16.44
```

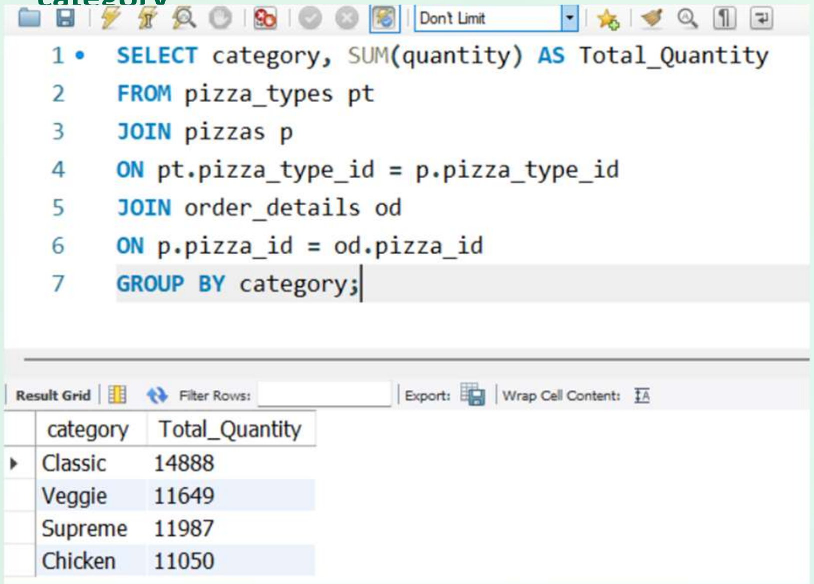
13. Total Order value per order >



```sql
SELECT
    od.order_id,
    SUM(od.quantity * p.price) AS total_order_value
FROM order_details od
JOIN pizzas p
    ON od.pizza_id = p.pizza_id
GROUP BY od.order_id
ORDER BY od.order_id;
```

```
order_id  total_order_value
1         13.25
2         92.00
3         37.25
4         16.50
5         16.50
6         24.75
7         12.50
8         12.50
9         143.25
```

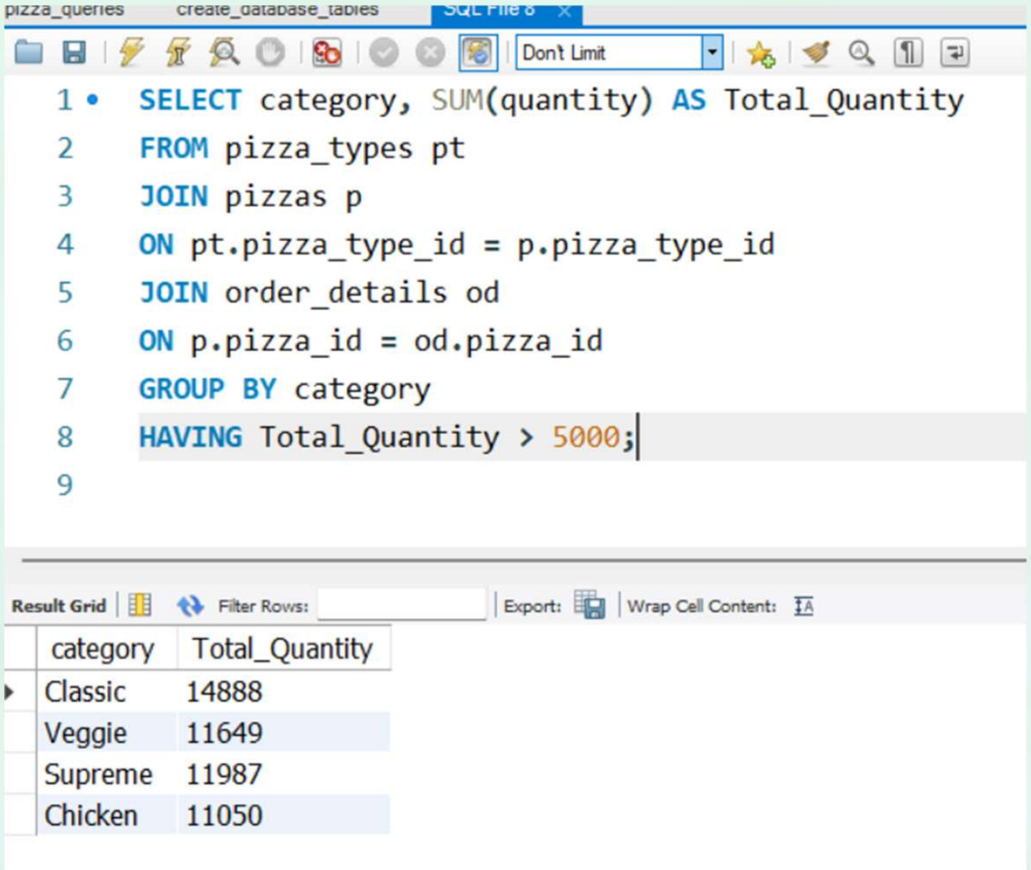
14. Total Quantity sold per category category



```sql
SELECT category, SUM(quantity) AS Total_Quantity
FROM pizza_types pt
JOIN pizzas p
ON pt.pizza_type_id = p.pizza_type_id
JOIN order_details od
ON p.pizza_id = od.pizza_id
GROUP BY category;
```

category	Total_Quantity
Classic	14888
Veggie	11649
Supreme	11987
Chicken	11050

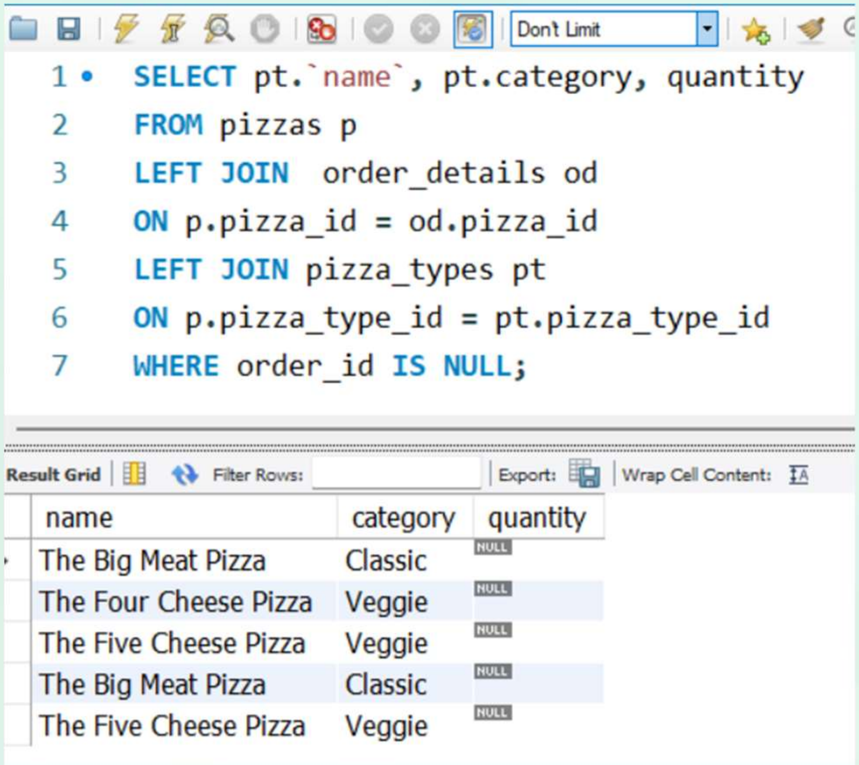
15. Categories with more than 5000 pizza sold



```sql
SELECT category, SUM(quantity) AS Total_Quantity
FROM pizza_types pt
JOIN pizzas p
ON pt.pizza_type_id = p.pizza_type_id
JOIN order_details od
ON p.pizza_id = od.pizza_id
GROUP BY category
HAVING Total_Quantity > 5000;
```

category	Total_Quantity
Classic	14888
Veggie	11649
Supreme	11987
Chicken	11050

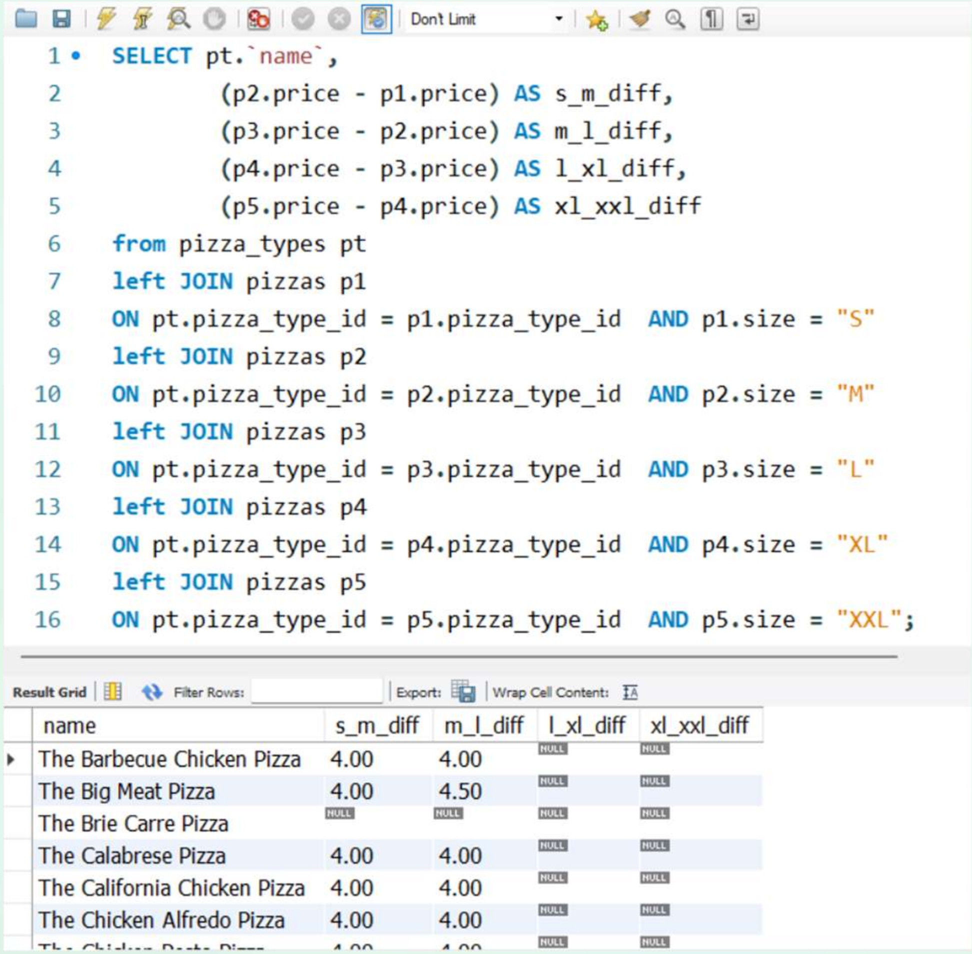
16. Pizzas never ordered



```sql
SELECT pt.`name`, pt.category, quantity
FROM pizzas p
LEFT JOIN  order_details od
ON p.pizza_id = od.pizza_id
LEFT JOIN pizza_types pt
ON p.pizza_type_id = pt.pizza_type_id
WHERE order_id IS NULL;
```

name	category	quantity
The Big Meat Pizza	Classic	NULL
The Four Cheese Pizza	Veggie	NULL
The Five Cheese Pizza	Veggie	NULL
The Big Meat Pizza	Classic	NULL
The Five Cheese Pizza	Veggie	NULL

17. Price differences between different sizes of the same pizza



```sql
SELECT pt.`name`,
        (p2.price - p1.price) AS s_m_diff,
        (p3.price - p2.price) AS m_l_diff,
        (p4.price - p3.price) AS l_xl_diff,
        (p5.price - p4.price) AS xl_xxl_diff
from pizza_types pt
left JOIN pizzas p1
ON pt.pizza_type_id = p1.pizza_type_id  AND p1.size = "S"
left JOIN pizzas p2
ON pt.pizza_type_id = p2.pizza_type_id  AND p2.size = "M"
left JOIN pizzas p3
ON pt.pizza_type_id = p3.pizza_type_id  AND p3.size = "L"
left JOIN pizzas p4
ON pt.pizza_type_id = p4.pizza_type_id  AND p4.size = "XL"
left JOIN pizzas p5
ON pt.pizza_type_id = p5.pizza_type_id  AND p5.size = "XXL";
```

name	s_m_diff	m_l_diff	l_xl_diff	xl_xxl_diff
The Barbecue Chicken Pizza	4.00	4.00	NULL	NULL
The Big Meat Pizza	4.00	4.50	NULL	NULL
The Brie Carre Pizza	NULL	NULL	NULL	NULL
The Calabrese Pizza	4.00	4.00	NULL	NULL
The California Chicken Pizza	4.00	4.00	NULL	NULL
The Chicken Alfredo Pizza	4.00	4.00	NULL	NULL
The Chicken Pesto Pizza	4.00	4.00	NULL	NULL

Key SQL Queries Solved

1

Category Sales Calculation

Aggregate total quantity sold and revenue by pizza category using GROUP BY

2

Top-Performing Pizzas

Rank pizzas by revenue and quantity using ORDER BY and aggregation functions

3

Zero-Sale Products

Identify pizzas never ordered using LEFT JOIN and NULL filtering techniques

4

Order-Level Revenue

Calculate total value per order by joining tables and multiplying price × quantity

5

Size Price Comparison

Use self-joins and CASE statements to compare pricing across pizza sizes

Insights & Findings

49,574

Total Pizzas Sold

Across all orders and categories during analysis period

5

Zero-Sale Products

Pizzas never ordered—candidates for promotion or removal

$35.95

Highest Price

Greek Pizza (Large) commanded premium pricing

Peak Hours

Evening sales surge beyond 8 PM requires optimized staffing

Strong Performers

Classic pizzas consistently drive high sales volume

Category Success

All categories exceeded 5,000 units sold threshold

Business Impact & Next Steps

Key Takeaways for IDC Pizza

Menu Optimization

Remove or rebrand the 5 zero-sale pizzas to streamline offerings

Pricing Strategy

Leverage size comparison insights to maximize profit margins

Operational Efficiency

Align kitchen staffing with evening rush patterns

Inventory Management

Ensure stock levels match category sales performance data

Project Success: Demonstrated end-to-end SQL mastery while delivering actionable insights that drive real business value for IDC Pizza.