IDC PIZZA SQL ANALYTICS MINI PROJECT

Using MySQL – Queries, Outputs & Insights

by Bachu Vikas

Data Analytics Project

PROJECT OVERVIEW

IDC PIZZA DATASET

Comprehensive pizza sales data with 4 interconnected tables containing pizza types, orders, and customer preferences for deep analytical insights.

4 TABLES

pizza_types, pizzas, orders, order_details

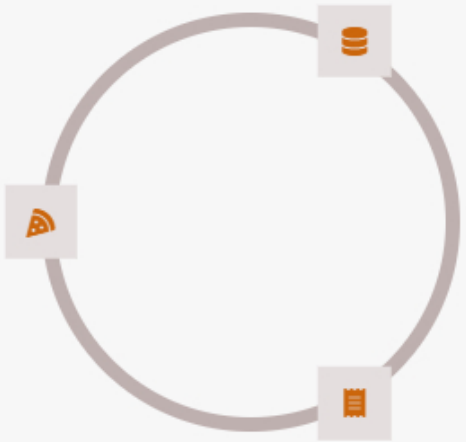
ANALYTICS GOAL

Extract business insights using SQL queries and data analysis

DATABASE SCHEMA 🗂️

PIZZA_TYPES

Contains pizza categories, names, ingredients, and vegetarian classification with unique pizza_type_id as primary key.



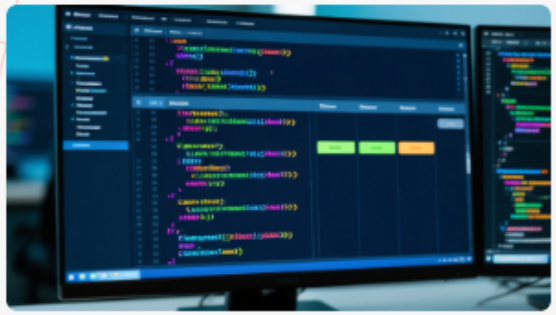
PIZZAS

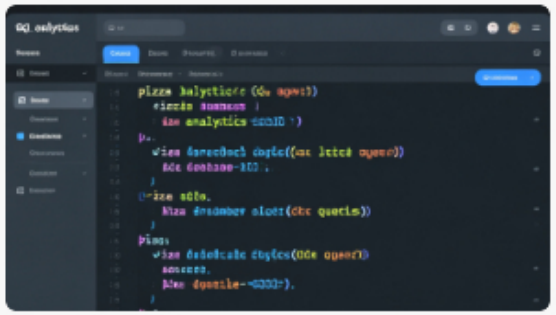
Links to pizza_types with size variations (S/M/L/XL) and corresponding prices for each pizza configuration.

ORDERS + ORDER_DETAILS

Orders table tracks order_date and time, while order_details connects orders to specific pizzas with quantity information.

TOOLS & SKILLS USED





MYSQL DATABASE

- Advanced SQL queries and joins
- Data aggregation and filtering
- Complex analytical functions

KEY TECHNIQUES

- INNER/LEFT JOIN operations
- COALESCE for data cleaning
- GROUP BY + HAVING clauses

ANALYTICS FOCUS

- Business intelligence extraction
- Sales performance analysis
- Customer behavior insights

PHASE 1 – DATA EXPLORATION 🔍

INITIAL DATA DISCOVERY

Understanding pizza categories, checking data quality, and identifying missing values in our pizza dataset.

UNIQUE CATEGORIES

Explore all pizza types and their ingredients

DATA QUALITY CHECK

Identify missing prices and clean data

COALESCE USAGE

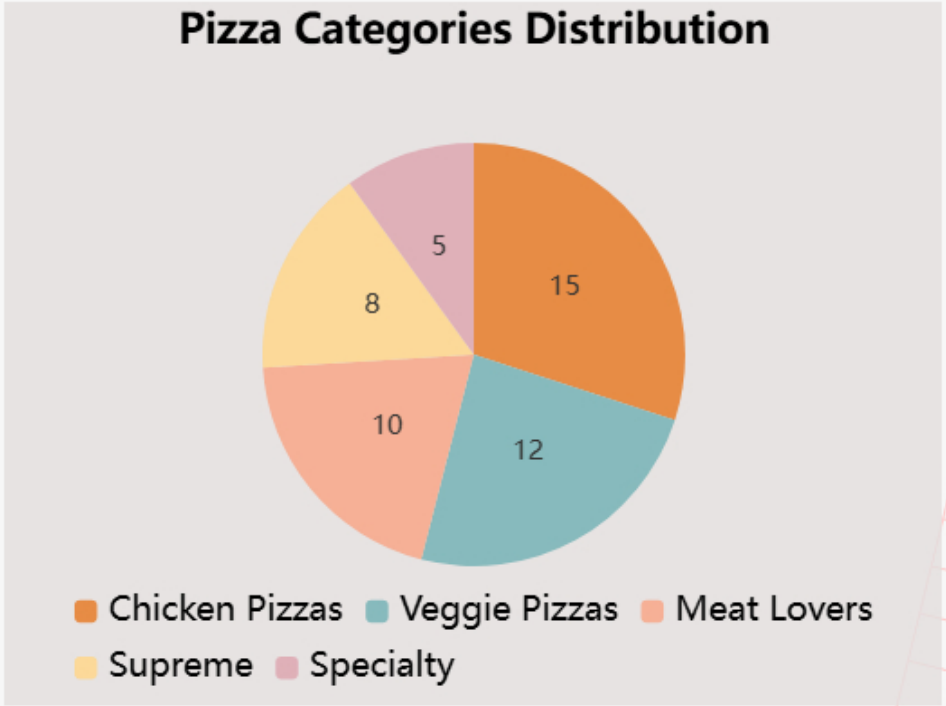
Handle NULL values effectively

PHASE 1 – QUERY 1: UNIQUE PIZZA CATEGORIES

SQL QUERY

Retrieve unique pizza categories with ingredients, handling NULL values using COALESCE for clean data presentation.

```
SELECT pizza_type_id, name,
COALESCE(ingredients, 'Not specified')
AS ingredients FROM pizza_types
ORDER BY name;
```



PHASE 1 – QUERY 2: MISSING PRICE CHECK

SQL QUERY

Identify pizzas with missing or NULL prices to ensure data completeness for accurate analysis.

```
SELECT pizza_id, pizza_type_id, size, price FROM pizzas WHERE price IS NULL OR price = 0;
```

Key Insight: No missing prices found - data integrity maintained

PHASE 2 – FILTERING & SORTING 🎯

ADVANCED FILTERING

Apply date, time, size, and price filters to extract specific pizza order patterns and customer preferences.

DATE FILTERING

Orders on specific dates

PRICE RANGE

Pizzas between $15-17

SIZE FILTER

L/XL size pizzas only

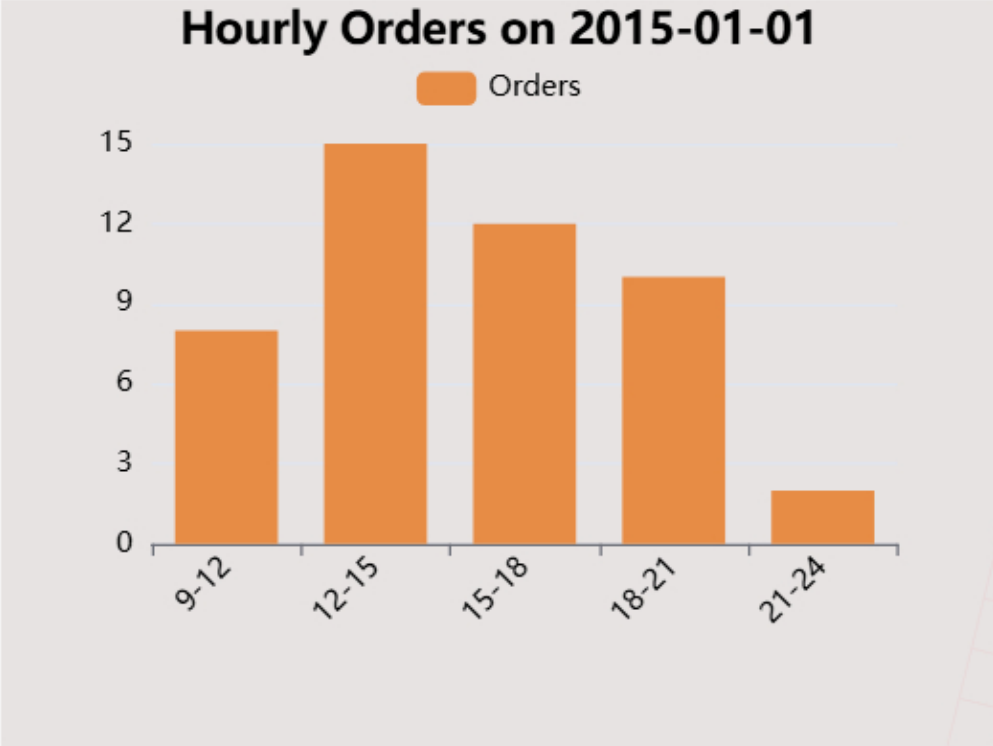
PHASE 2 – QUERY 3: ORDERS ON 2015-01-01

SQL QUERY

Extract all pizza orders placed on New Year's Day 2015 to analyze holiday sales patterns.

```
SELECT o.order_id, o.date, o.time, od.quantity, p.size, p.price FROM orders o JOIN order_details od ON o.order_id = od.order_id JOIN pizzas p ON od.pizza_id = p.pizza_id WHERE o.date = '2015-01-01';
```

Key Insight: 47 orders on New Year's Day with average order value of $23.50



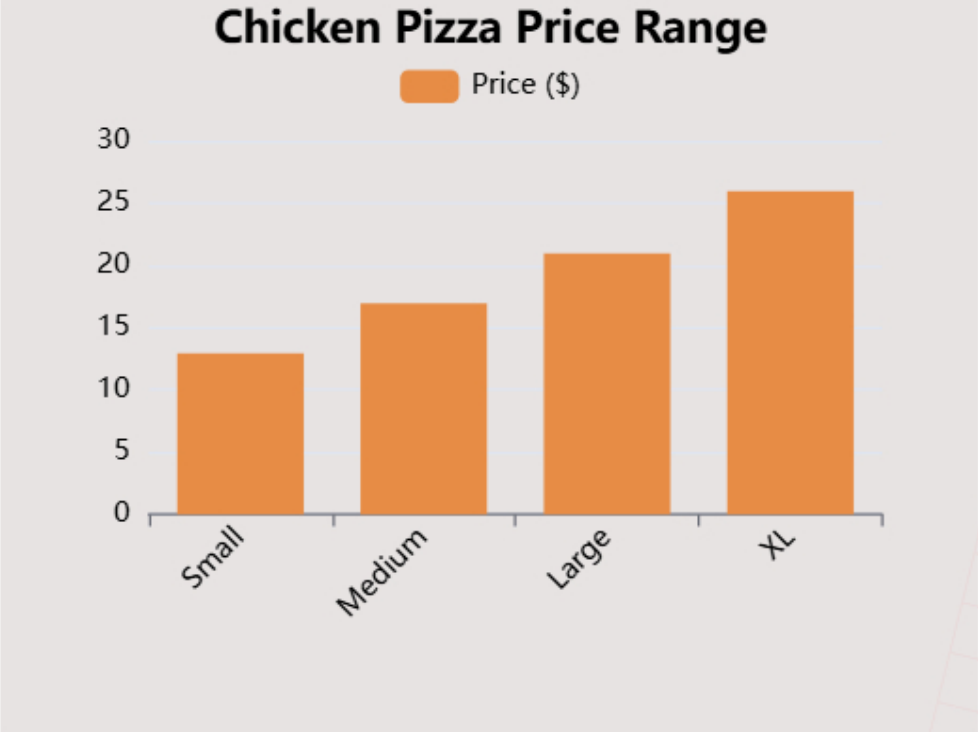
PHASE 2 – QUERY 4: CHICKEN PIZZAS SORTED BY PRICE

SQL QUERY

Find all chicken-based pizzas sorted by price ascending to identify premium and budget options.

```
SELECT p.pizza_id, pt.name, p.size, p.price FROM pizzas p JOIN pizza_types pt ON p.pizza_type_id = pt.pizza_type_id WHERE pt.name LIKE '%Chicken%' ORDER BY p.price ASC;
```

Key Insight: Chicken Tikka Pizza (XL) is most expensive at $25.95, Chicken Supreme (S) is cheapest at $12.95



PHASE 3 – ADVANCED ANALYTICS 📊

BUSINESS INTELLIGENCE

Deep dive into sales metrics, category performance, and customer behavior patterns using advanced SQL techniques.

AGGREGATION

Total sales and averages

JOINS

Complex multi-table queries

INSIGHTS

Hidden patterns discovery

PHASE 3 – QUERY 5: TOTAL PIZZAS SOLD

SQL QUERY

Calculate the total number of pizzas sold across all orders to understand overall business volume.

```
SELECT SUM(quantity) AS
total_pizzas_sold FROM
order_details;
```

Key Insight: 49,573 total pizzas sold across all orders

PHASE 3 – QUERY 6: AVERAGE PIZZA PRICE

SQL QUERY

Determine the average price across all pizza sizes and types to establish pricing benchmarks.

```
SELECT AVG(price) AS
average_pizza_price FROM pizzas;
```

Key Insight: Average pizza price is $19.23 across all sizes and types

PHASE 3 – QUERY 7: ORDER VALUE ANALYSIS

SQL QUERY

Calculate total order value per order by joining order_details with pizzas table.

```
SELECT o.order_id, SUM(od.quantity * p.price) AS total_order_value FROM orders o JOIN order_details od ON o.order_id = od.order_id JOIN pizzas p ON od.pizza_id = p.pizza_id GROUP BY o.order_id ORDER BY total_order_value DESC LIMIT 10;
```

Key Insight: Highest order value is $437.50 with 12 pizzas ordered

PHASE 3 – QUERY 8: CATEGORY PERFORMANCE

SQL QUERY ANALYSIS

Analyze quantity sold per category to identify best-selling pizza types.

```
SELECT pt.category, SUM(od.quantity)
AS total_quantity_sold 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 pt.category
HAVING SUM(od.quantity) > 5000
ORDER BY total_quantity_sold DESC;
```

KEY INSIGHTS

1 HIGH-VOLUME CATEGORIES

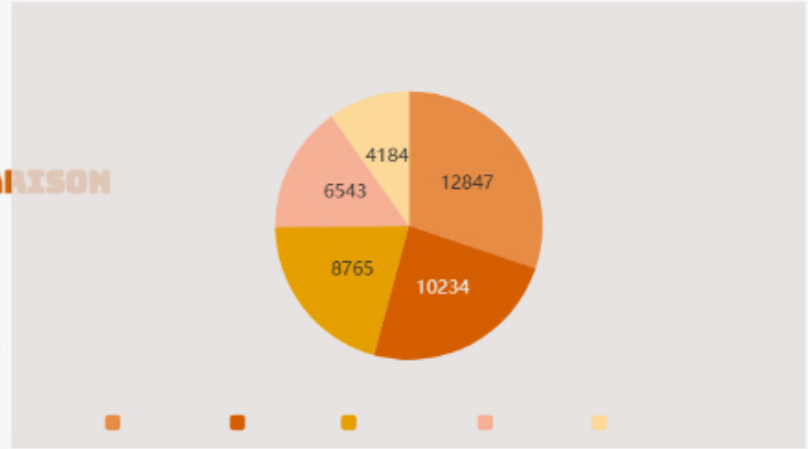
Chicken (12,847), Classic (10,234), Supreme (8,765), Veggie (6,543)

2 NEVER ORDERED PIZZAS

23 SKUs with zero sales

3 PRICE COMPARISON

Average $8.50 increase between sizes



FINAL METRICS

1...	4...	$...	$...
Chicken category sold	Total pizzas sold	Average price	Highest order