Home About Us Content Contact Us

All About Pizza!

THE GREAT PIZZA ANALYTICS CHALLENGE

PREPARED BY DIVYANSHI DOSER

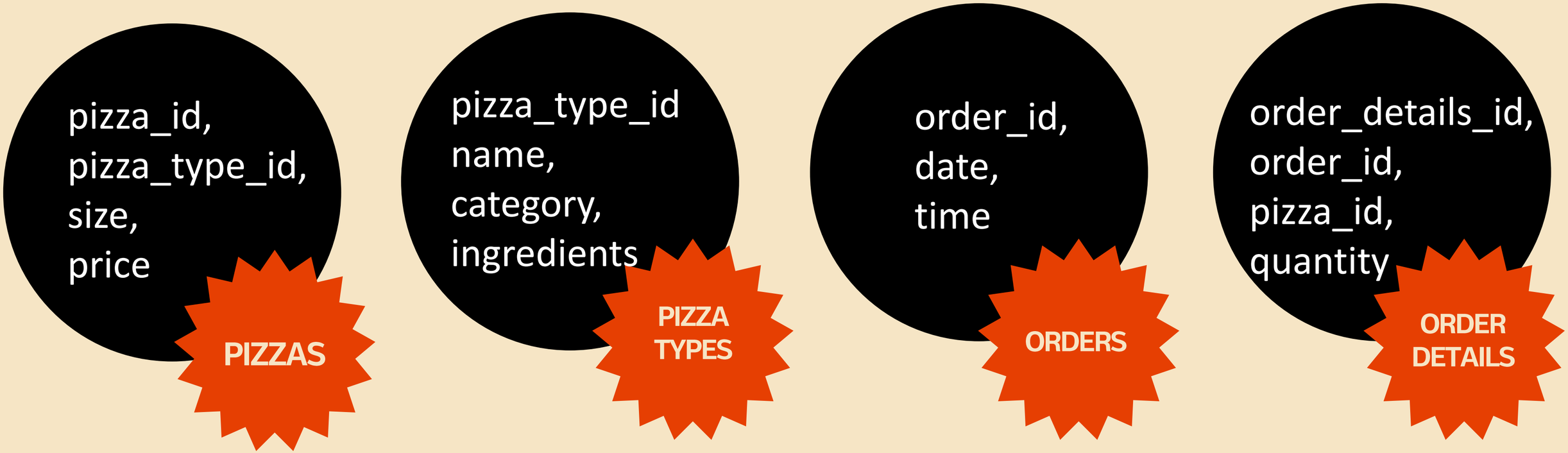
PROJECT OVERVIEW

OBJECTIVE: Transform raw pizza sales data into actionable insights using SQL

DATASET: IDC_Pizza database with tables for pizzas, pizza types, orders, and order details

SKILLS PRACTICED: SQL querying (SELECT, JOINs, GROUP BY, HAVING) - Data filtering & cleaning - Aggregations & sales analysis

DATABASE SETUP



PHASE 1

FOUNDATION & INSPECTION

1. Install IDC_Pizza.dump as IDC_Pizza server

2.List all unique pizza categories (`DISTINCT`).

- SELECT DISTINCT category FROM pizza_types;

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

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

4. Check for pizzas missing a price (`IS NULL`).

- SELECT * FROM pizzas WHERE price IS NULL;

PHASE 2

FILTERING & EXPLORATION

1. 1. Orders placed on `'2015-01-01'`.
- SELECT * FROM orders WHERE date = "2015-01-01";

2. List pizzas with `price` descending.
- SELECT * FROM pizzas ORDER BY price DESC;

3. Pizzas sold in sizes `'L'` or `'XL'.
- SELECT * FROM pizzas WHERE size IN("L","XL");

4. Pizzas priced between $15.00 and $17.00.
- SELECT * FROM pizzas WHERE price BETWEEN 15 AND 17;

5. Pizzas with `"Chicken"` in the name.
- SELECT * FROM pizza_types WHERE name LIKE "%Chicken%";

6. Orders on `'2015-02-15'` or placed after 8 PM.
- SELECT * FROM orders WHERE date = "2015-02-15" OR HOUR(time) > 20;

PHASE 3

SALES PERFORMANCE

1. Total quantity of pizzas sold.
- SELECT SUM(quantity) AS quantity_sold FROM order_details;

2. Average pizza price.
- SELECT ROUND(AVG(price),2) AS avg_pizza_price FROM pizzas;

3. Total order value per order.
- SELECT o.order_id, SUM(od.quantity*p.price) AS order_value FROM orders AS o JOIN order_details AS od ON o.order_id = od.order_id JOIN pizzas AS p ON od.pizza_id = p.pizza_id GROUP BY o.order_id;

4. Total quantity sold per pizza category.
- SELECT pt.category, SUM(od.quantity) AS total_quantity_sold FROM order_details AS od JOIN pizzas AS p ON od.pizza_id = p.pizza_id JOIN pizza_types AS pt ON p.pizza_type_id = pt.pizza_type_id GROUP BY pt.category;

5. Categories with more than 5,000 pizzas sold.
- SELECT pt.category, SUM(od.quantity) AS total_quantity_sold FROM order_details AS od JOIN pizzas AS p ON od.pizza_id = p.pizza_id JOIN pizza_types AS pt ON p.pizza_type_id = pt.pizza_type_id GROUP BY pt.category HAVING total_quantity_sold > 5000;

6. Pizzas never ordered.
- SELECT p.* FROM pizzas AS p LEFT JOIN order_details AS od ON p.pizza_id = od.pizza_id WHERE od.order_id IS NULL;

7. Price differences between different sizes of the same pizza.
- SELECT a.pizza_type_id, a.size AS size1, b.size AS size2, (a.price-b.price) AS price_difference FROM pizzas AS a JOIN pizzas AS b ON a.pizza_type_id = b.pizza_type_id AND a.size <> b.size;

THANK YOU