**Pizza Basket Analysis**

-- 1. Identify Most Frequently Ordered Pizzas.

SELECT pt.name AS pizza\_name, COUNT(od.order\_id) AS total\_orders

FROM pizza\_types pt

JOIN pizzas p ON pt.pizza\_type\_id = p.pizza\_type\_id

join order\_details od on od.pizza\_id = p.pizza\_id

GROUP BY pt.name

ORDER BY total\_orders DESC

LIMIT 10;

A screenshot of a menu

AI-generated content may be incorrect.

-- 2. Find Pizza Combinations Ordered Together

SELECT pt1.name AS pizza\_A, pt2.name AS pizza\_B, COUNT(\*) AS frequency

FROM order\_details od1

JOIN order\_details od2

ON od1.order\_id = od2.order\_id AND od1.pizza\_id <> od2.pizza\_id

JOIN pizzas p1 ON od1.pizza\_id = p1.pizza\_id

JOIN pizzas p2 ON od2.pizza\_id = p2.pizza\_id

JOIN pizza\_types pt1 ON p1.pizza\_type\_id = pt1.pizza\_type\_id

JOIN pizza\_types pt2 ON p2.pizza\_type\_id = pt2.pizza\_type\_id

WHERE pt1.name < pt2.name

GROUP BY pizza\_A, pizza\_B

ORDER BY frequency DESC

LIMIT 10;

A screenshot of a menu

AI-generated content may be incorrect.

**Calculate Support, Confidence, and Lift**

-- 3. Support (How frequently an item appears in transactions).

SELECT pt.name AS pizza\_name,

COUNT(od.pizza\_id) / (SELECT COUNT(DISTINCT order\_id) FROM order\_details) AS support

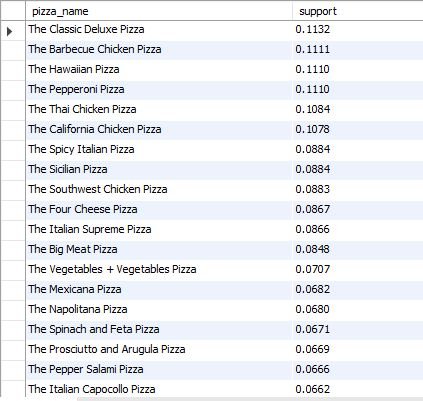
FROM order\_details od

JOIN pizzas p ON od.pizza\_id = p.pizza\_id

join pizza\_types pt on p.pizza\_type\_id = pt.pizza\_type\_id

GROUP BY pt.name

ORDER BY support DESC;



-- 4. Confidence (How often B is bought when A is bought).

WITH pizza\_order\_count AS (

SELECT pizza\_id, COUNT(\*) AS total\_orders

FROM order\_details

GROUP BY pizza\_id

)

SELECT pt1.name AS pizza\_A, pt2.name AS pizza\_B,

COUNT(\*) / poc.total\_orders AS confidence

FROM order\_details od1

JOIN order\_details od2

ON od1.order\_id = od2.order\_id AND od1.pizza\_id <> od2.pizza\_id

JOIN pizzas p1 ON od1.pizza\_id = p1.pizza\_id

JOIN pizzas p2 ON od2.pizza\_id = p2.pizza\_id

JOIN pizza\_types pt1 ON p1.pizza\_type\_id = pt1.pizza\_type\_id

JOIN pizza\_types pt2 ON p2.pizza\_type\_id = pt2.pizza\_type\_id

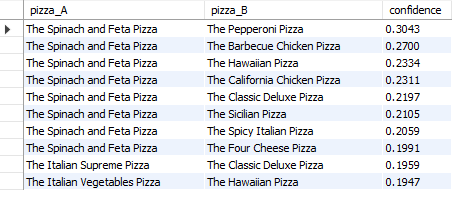
JOIN pizza\_order\_count poc ON p1.pizza\_id = poc.pizza\_id

WHERE pt1.name > pt2.name

GROUP BY pt1.name, pt2.name, poc.total\_orders

ORDER BY confidence DESC

LIMIT 10;



-- 5. Lift (Strength of the relationship).

WITH total\_orders\_cte AS (

SELECT COUNT(DISTINCT order\_id) AS total\_orders

FROM order\_details

),

pizza\_counts AS (

SELECT pizza\_id, COUNT(DISTINCT order\_id) AS orders\_count

FROM order\_details

GROUP BY pizza\_id

),

pair\_counts AS (

SELECT

od1.pizza\_id AS pizza\_A\_id,

od2.pizza\_id AS pizza\_B\_id,

COUNT(DISTINCT od1.order\_id) AS pair\_count

FROM order\_details od1

JOIN order\_details od2

ON od1.order\_id = od2.order\_id

AND od1.pizza\_id < od2.pizza\_id

GROUP BY od1.pizza\_id, od2.pizza\_id

)

SELECT

pt1.name AS pizza\_A,

pt2.name AS pizza\_B,

(pc\_pair.pair\_count \* tot.total\_orders) / (pc1.orders\_count \* pc2.orders\_count) AS lift

FROM pair\_counts pc\_pair

JOIN pizza\_counts pc1 ON pc\_pair.pizza\_A\_id = pc1.pizza\_id

JOIN pizza\_counts pc2 ON pc\_pair.pizza\_B\_id = pc2.pizza\_id

CROSS JOIN total\_orders\_cte tot

JOIN pizzas p1 ON pc\_pair.pizza\_A\_id = p1.pizza\_id

JOIN pizzas p2 ON pc\_pair.pizza\_B\_id = p2.pizza\_id

JOIN pizza\_types pt1 ON p1.pizza\_type\_id = pt1.pizza\_type\_id

JOIN pizza\_types pt2 ON p2.pizza\_type\_id = pt2.pizza\_type\_id

ORDER BY lift DESC

LIMIT 10;

