

BASES DE DATOS – EXAMEN FINAL - 10/02/2025

Ejercicio 2:

Codificar una función en PostgreSQL denominada *f_productos_afines()* para obtener **grupos de tres artículos** que aparecen juntos en la misma factura más veces considerando un umbral mínimo a tener en cuenta a partir del cual se contabilizan. Por ejemplo:

Producto 1 Id	Cantidad producto 1	Producto 2 Id	Cantidad producto 1 y 2	Producto 3 Id	Cantidad producto 1, 2 y 3
230	68	231	37	233	37
46	76	230	7	233	6
46	76	231	6	233	5
230	68	233	49	303	5
231	69	233	44	303	5

En esta salida se utiliza como umbral 5 ocurrencias; de haber menos no se tiene en cuenta esa combinación. Puede verse que hay 37 facturas que contienen simultáneamente los productos **230, 231 y 233**. La salida de la función debe retornar registros de acuerdo con lo mostrado en la tabla anterior. La función debe tener al umbral como parámetro.

SOLUCIÓN ESPERADA:

```

CREATE OR REPLACE FUNCTION f_productos_afines(umbral INTEGER)
RETURNS TABLE (producto1_id bigint, producto1_cantidad bigint,
                producto2_id bigint, producto1y2_cantidad bigint,
                producto3_id bigint, producto1y2y3_cantidad bigint) AS $$

BEGIN
    -- Paso 1: Crear una tabla temporal para producto1
    CREATE TEMP TABLE temp_producto1 AS
    SELECT
        producto_id AS producto1_id,
        COUNT(*) AS cantidad1
    FROM factura_detalle
    GROUP BY producto_id;

    -- Paso 2: Crear una tabla temporal para producto1 y producto2 juntos
    CREATE TEMP TABLE temp_producto12 AS
    SELECT
        t1.producto_id AS producto1_id,
        t2.producto_id AS producto2_id,
        COUNT(*) AS cantidad12
    FROM factura_detalle t1
    JOIN factura_detalle t2
        ON t1.pedido_id = t2.pedido_id
        AND t1.producto_id < t2.producto_id
    GROUP BY t1.producto_id, t2.producto_id;

    -- Paso 3: Crear una tabla temporal para producto1, producto2 y producto3 juntos
    CREATE TEMP TABLE temp_producto123 AS
    SELECT
        t1.producto_id AS producto1_id,
        t2.producto_id AS producto2_id,
        t3.producto_id AS producto3_id,
        COUNT(*) AS cantidad123
    FROM factura_detalle t1
    JOIN factura_detalle t2
        ON t1.pedido_id = t2.pedido_id
        AND t1.producto_id < t2.producto_id
    JOIN factura_detalle t3
        ON t1.pedido_id = t3.pedido_id
        AND t2.producto_id < t3.producto_id
    GROUP BY t1.producto_id, t2.producto_id, t3.producto_id;

    -- Paso 4: Retornar resultados
    RETURN QUERY
    SELECT
        cast(p123.producto1_id as bigint) AS producto1_id,
        p1.cantidad1 AS producto1_cantidad,

```

```

cast(p123.producto2_id as bigint) AS producto2_id,
p12.cantidad12 AS producto1y2_cantidad,
cast(p123.producto3_id as bigint) AS producto3_id,
p123.cantidad123 AS producto12y3_cantidad
FROM temp_producto123 p123
JOIN temp_producto12 p12
  ON p123.producto1_id = p12.producto1_id
  AND p123.producto2_id = p12.producto2_id
JOIN temp_producto1 p1
  ON p123.producto1_id = p1.producto1_id
WHERE p123.cantidad123 >= umbral;

-- Paso 6: Limpiar las tablas temporales
DROP TABLE temp_producto1;
DROP TABLE temp_producto12;
DROP TABLE temp_producto123;

END;
$$ LANGUAGE plpgsql;

select f_productos_afines(5)

ALTERNATIVA:
-- Select de solución alternativa que:
-- No devuelve la cantidad correcta de producto1 y producto 2.
-- Pero la "idea", estaría.
SELECT
  p1.producto_id AS "producto1_id",
  COUNT(f1.pedido_id) AS cantidad1,
  -- Nivel 2
  p2.producto_id AS "producto2_id",
  COUNT(f2.pedido_id) AS "cantidad_prod1y2",
  -- Nivel 3
  p3.producto_id AS "producto3_id",
  COUNT(f3.pedido_id) AS "cantidad_prod1y2y3"
FROM factura_detalle f1
JOIN producto p1 ON p1.producto_id = f1.producto_id
JOIN factura_detalle f2 ON f1.pedido_id = f2.pedido_id AND f1.producto_id < f2.producto_id
JOIN producto p2 ON p2.producto_id = f2.producto_id
join factura_detalle f3 ON f3.pedido_id = f2.pedido_id AND f2.producto_id < f3.producto_id
join producto p3 on p3.producto_id = f3.producto_id
GROUP BY p1.producto_id, p2.producto_id, p3.producto_id
having COUNT(f1.pedido_id) > 5 and COUNT(f2.pedido_id) > 5 and COUNT(f3.pedido_id)

```