Here’s a simple business case scenario that illustrates the use of UPDATE and DELETE with JOIN in a PostgreSQL database:

**Business Scenario:**

A company manages its customer data in two tables:

1. **Customers** (stores customer details like name, address, etc.)
2. **Orders** (stores order details including customer ID, order status, etc.)

The company needs to perform two operations:

1. **UPDATE**: Update the customer status to "Inactive" for customers who have no active orders.
2. **DELETE**: Remove customers from the database who have been marked as "Inactive" and haven't placed any orders in the last year.

**1. UPDATE Example (Mark Customers as Inactive)**

Business requirement: If a customer has no active orders, mark them as "Inactive."

sql

Copy code

UPDATE Customers

SET status = 'Inactive'

FROM Orders

WHERE Customers.customer\_id = Orders.customer\_id

AND Orders.status = 'Completed';

In this query, the Customers table is updated with a status of 'Inactive' for customers whose orders have been completed (or no active orders exist).

**2. DELETE Example (Remove Inactive Customers with No Recent Orders)**

Business requirement: Delete customers who have been marked as "Inactive" and have no orders in the past year.

sql

Copy code

DELETE FROM Customers

USING Orders

WHERE Customers.customer\_id = Orders.customer\_id

AND Customers.status = 'Inactive'

AND Orders.order\_date < NOW() - INTERVAL '1 year';

This query deletes rows from the Customers table where the customer is inactive and hasn’t placed any orders in the last year.

These operations help maintain a clean and updated database by removing or updating customers based on their order activity.

4o