

## ◇ Scenario 1: Schreibfehler in Produktbeschreibung

### Technique Used: SCD Type 1

#### Motivation:

Der Schreibfehler ist geringfügig ("garms" statt "grams"). Die Beschreibung soll direkt überschrieben werden, ohne Historie zu führen.

#### SQL (Fehler absichtlich einfügen + korrigieren):

```
-- Fehler simulieren
UPDATE products
SET quantity_per_unit = REPLACE(quantity_per_unit, 'grams', 'garms')
WHERE quantity_per_unit LIKE '%grams%';

-- Fehler korrigieren
UPDATE products
SET quantity_per_unit = REPLACE(quantity_per_unit, 'garms', 'grams')
WHERE quantity_per_unit LIKE '%garms%';
```

## ◇ Scenario 2: Rückwirkende Änderung der Beschreibung

### Technique Used: SCD Type 2

#### Motivation:

Da bereits fehlerhafte Verkaufsdaten vorhanden sind, soll die Änderung mit Historie gespeichert werden.

#### SQL:

```
-- Historientabelle erstellen
CREATE TABLE IF NOT EXISTS products_history (
    products_history_id SERIAL PRIMARY KEY,
    product_id SMALLINT NOT NULL,
    product_name VARCHAR(40) NOT NULL,
    quantity_per_unit VARCHAR(20),
    supplier_id SMALLINT,
    category_id SMALLINT,
    unit_price REAL,
```

```

        units_in_stock SMALLINT,
        units_on_order SMALLINT,
        reorder_level SMALLINT,
        discontinued INTEGER NOT NULL,
        valid_from DATE,
        valid_to DATE
    );

-- Bestehenden Stand übernehmen (einmalig)
INSERT INTO products_history (
    product_id, product_name, quantity_per_unit, supplier_id,
    category_id,
    unit_price, units_in_stock, units_on_order, reorder_level,
    discontinued,
    valid_from, valid_to
)
SELECT
    product_id, product_name, quantity_per_unit, supplier_id,
    category_id,
    unit_price, units_in_stock, units_on_order, reorder_level,
    discontinued,
    DATE '2020-01-01', NULL
FROM products
WHERE product_id = 1;

-- Alten Stand beenden
UPDATE products_history
SET valid_to = DATE '2023-12-03'
WHERE product_id = 1 AND valid_to IS NULL;

-- Neuen Stand ab 04.12.2023 einfügen
INSERT INTO products_history (
    product_id, product_name, quantity_per_unit, supplier_id,
    category_id,
    unit_price, units_in_stock, units_on_order, reorder_level,
    discontinued,
    valid_from, valid_to
)
SELECT
    product_id, product_name, '24 cans x 330ml', supplier_id,
    category_id,
    unit_price, units_in_stock, units_on_order, reorder_level,

```

```
discontinued,  
    DATE '2023-12-04', NULL  
FROM products_history  
WHERE product_id = 1 AND valid_to = DATE '2023-12-03';
```

### ◇ Scenario 3: Store-Umbenennung – Alt/Neu-Vergleich

**Technique Used:** SCD Type 3

**Motivation:**

Vergleich zwischen altem und neuem Store-Namen in Berichten.

**SQL:**

```
-- Alte Namensspalte ergänzen  
ALTER TABLE resellers  
ADD COLUMN previous_store_name VARCHAR(100);  
  
-- Namensänderung mit Historie durchführen  
UPDATE resellers  
SET previous_store_name = store_name,  
    store_name = 'Nelson Mandela Bay Store'  
WHERE reseller_id = 42;
```

### ◇ Scenario 4: Kundenbindungsprogramm mit Adressverfolgung

**Technique Used:** SCD Type 2

**Motivation:**

Kundendaten ändern sich über Zeit (z. B. Adresse). Historie muss für Marketing nachvollziehbar sein.

**SQL:**

```
-- Tabelle für Kundenhistorie erstellen  
CREATE TABLE IF NOT EXISTS customer_loyalty_history (  
    customer_loyalty_id SERIAL PRIMARY KEY,
```

```

    customer_id INT NOT NULL,
    title VARCHAR(10),
    name VARCHAR(50),
    surname VARCHAR(50),
    gender VARCHAR(10),
    marital_status VARCHAR(20),
    date_of_birth DATE,
    age INT,
    address TEXT,
    postal_code VARCHAR(10),
    valid_from DATE,
    valid_to DATE
);

-- Erste Anmeldung zum Programm ab 01.01.2025
INSERT INTO customer_loyalty_history (
    customer_id, title, name, surname, gender, marital_status,
    date_of_birth, age, address, postal_code,
    valid_from, valid_to
)
VALUES (
    5001, 'Herr', 'Max', 'Muster', 'Männlich', 'Verheiratet',
    DATE '1985-04-12', 39, 'Beispielweg 7, Berlin', '10115',
    DATE '2025-01-01', NULL
);

-- Änderung z. B. bei Umzug
UPDATE customer_loyalty_history
SET valid_to = DATE '2025-06-30'
WHERE customer_id = 5001 AND valid_to IS NULL;

INSERT INTO customer_loyalty_history (
    customer_id, title, name, surname, gender, marital_status,
    date_of_birth, age, address, postal_code,
    valid_from, valid_to
)
VALUES (
    5001, 'Herr', 'Max', 'Muster', 'Männlich', 'Verheiratet',
    DATE '1985-04-12', 39, 'Musterstraße 88, Hamburg', '20095',
    DATE '2025-07-01', NULL
);

```