

```

-- Trigger 1:
-- Prozedur für Produkt-ID
CREATE OR REPLACE PROCEDURE gen_product_id
(new_num OUT VARCHAR2)
AS
    max_num NUMBER;
BEGIN
    -- höchste Produkt-id ermitteln und erhöhen
    SELECT NVL(MAX(TO_NUMBER(SUBSTR(ProduktNummer, 2))), 0) + 1
    INTO max_num
    FROM Produkt;

    -- error wenn maximale Grenze erreicht wurde
    IF max_num > 999 THEN
        RAISE_APPLICATION_ERROR(-20202, 'Maximale Anzahl an Produkten erreicht. Bitte
das Lager aufräumen!');
    END IF;

    -- neue Produkt-id im Format 'PXXX' generieren
    new_num := 'P' || LPAD(max_num, 3, '0');
END;

-- Trigger Produkt-ID
CREATE OR REPLACE TRIGGER trg_product_id_gen
BEFORE INSERT ON Produkt
FOR EACH ROW
BEGIN
    -- Produkt-id wird sie automatisch hinzugefügt
    IF :NEW.ProduktNummer IS NULL THEN
        gen_product_id(:NEW.ProduktNummer);
    END IF;
END;

-- Trigger 2:
-- Kunden-ID
CREATE OR REPLACE TRIGGER trg_customer_no_dbl_alias
BEFORE INSERT ON Kartell_Kunde
FOR EACH ROW
DECLARE
    v_count NUMBER;
BEGIN
    -- Prüfen, ob der Alias (unabhängig von Groß-/Kleinschreibung) bereits existiert
    SELECT COUNT(*)
    INTO v_count
    FROM Kartell_Kunde
    WHERE UPPER(Alias) = UPPER(:NEW.Alias);

    -- Wenn der Alias existiert, eine benutzerdefinierte Fehlermeldung auslösen
    IF v_count > 0 THEN
        RAISE_APPLICATION_ERROR(-20201, 'Der Alias ist schon vergeben, du Erbse!');
    END IF;
END;

```