**Trigger**

**Ein Trigger (Auslöser) kann man auslösen: nach Dateneinfügungen, Aktualisierungen und Löschungen.**

In **Inserted** – Daten-Objekt befinden sich die zu einfügenden Daten

In **Deleted** – Daten-Objekt befinden sich die zu löschenden Daten

**Aufgabe**: Erstelle Trigger, der bei jeder Änderung an den Daten der **BestellungPositionen** den **Gesamtbetrag** neu berechnet wird.

Dieser Trigger soll **auch „tote Bestellungen“** aus der Datenbank entfernen. Das heißt: Bestellung ohne mindestens einer BestellPosition muss weg.

Also bei jedem INSERT, UPDATE und DELETE

**CREATE OR ALTER TRIGGER [dbo].[BestellPositionOnInsertUpdateDelete]**

ON **[dbo].[BestellPosition]**

**AFTER** **INSERT, UPDATE, DELETE**

AS

BEGIN

DECLARE @BestellungId INT = (SELECT BestellungId FROM Inserted) -- bei Einfügen

IF @BestellungId IS NULL

SET @BestellungId = (SELECT BestellungId FROM Deleted) -- bei Löschen

EXEC SetGesamtBetrag @BestellungId, NULL

-- Löschen von "toten Bestellungen" - ohne Bestellpositionen

IF (SELECT COUNT(\*) FROM BestellPosition WHERE BestellungId = @BestellungId) = 0

BEGIN

DELETE Bestellung WHERE BestellungId = @BestellungId

END

END

**Aufgabe: Erstelle einen Trigger, der bei jedem Update (Aktualisierung) in der Produkt-Tabelle den Gesamtbetrag neu berechnet.**

**CREATE OR ALTER TRIGGER ProduktOnUpdate**

ON **Produkt**

AFTER **UPDATE**

AS

BEGIN

IF OBJECT\_ID(N'tempdb..##LocalCustomer') IS NOT NULL

BEGIN

DROP TABLE #LocalCustomer

END

CREATE TABLE #BestellungIds (BestellungIdsId INT IDENTITY(1,1), BestellungId INT NOT NULL)

INSERT #BestellungIds

SELECT DISTINCT BestellungId

FROM BestellPosition bp INNER JOIN Inserted p

ON bp.ProduktId = p.ProduktId

SELECT \* FROM #BestellungIds

DECLARE @Zaehler INT = 1

DECLARE @Count INT = (SELECT COUNT(\*) FROM #BestellungIds)

DECLARE @BestellungId INT

WHILE(@Zaehler <= @Count)

BEGIN

SELECT @BestellungId = BestellungId FROM #BestellungIds WHERE BestellungIdsId = @Zaehler

EXEC SetGesamtBetrag @BestellungId, NULL

SET @Zaehler = @Zaehler + 1

END

DROP TABLE #BestellungIds

END

GO

*Beachten:****INSTEAD OF-Trigger****: Diese Trigger ersetzen die auslösende Aktion und führen anstelle der eigentlichen Operation eigene Aktionen durch. Diese Art von Trigger ist besonders nützlich, wenn man z. B. bei einer Sicht sicherstellen möchte, dass alle Änderungen in einer bestimmten Weise behandelt werden.*