



ING. ANDREAS HAIDER

KMU-Experte mit MEHRwert



SQL

Automatisierung: **Trigger**

Eine besondere Art von Prozeduren sind **Trigger**, die **automatisch** ausgeführt werden, wenn unterschiedliche Ereignisse in der Datenbank ausgelöst werden. Die Art der Ereignisse bestimmen dabei die Art, die Funktionsweise und den Namen des Triggers. Je nach Triggertyp können sie an eine **Tabelle** oder **Sicht** oder an eine **Datenbank** bzw. an den **Server** andockt werden. Im Gegensatz zu einer Prozedur oder Funktion kann ein Trigger ausschließlich automatisch durch das Eintreten eines Ereignisses überhaupt „abgefeuert“ werden.

Als Ereignisse gelten hierbei zusammenfassend die folgenden:

Data Manipulation Language (DML) - Befehle wie **INSERT-**, **UPDATE-** oder **DELETE-**Anweisungen

Data Definition Language (DDL) - Befehle wie **CREATE-**, **ALTER-** und **DROP-**Anweisungen

LOGON-Trigger – Einrichtung einer Benutzersitzung, ... (LOGOFF, ERRORS, STARTUP, SHUTDOWN, ...)



SQL

Sinn und Zweck: **Trigger**

Sinn und Zweck von Triggern ist es, **komplexe Integritätsregeln** und **automatische Funktionsabläufe** zur **Datensicherheit/-integrität und –qualität** einzurichten, die sich **mit gewöhnlichen SQL-Anweisungen nicht umsetzen lassen**.

Mit Triggern hat man nun die Möglichkeiten, ein ganzes „Programm“ zu schreiben, das mit Fallunterscheidungen, Variablen sowie allen möglichen T-SQL/.NET-Möglichkeiten ausgestattet ist. Um Anforderungen im Bereich der Integritätssicherung zu erfüllen.

Trigger für Datenmanipulation (DML)-Trigger dienen dazu, anspruchsvolle und über gewöhnliche SQL-Angaben hinausgehende Geschäftsregeln und Datenintegritäten aufzubauen.

SQL

Syntax: **DML-Trigger**

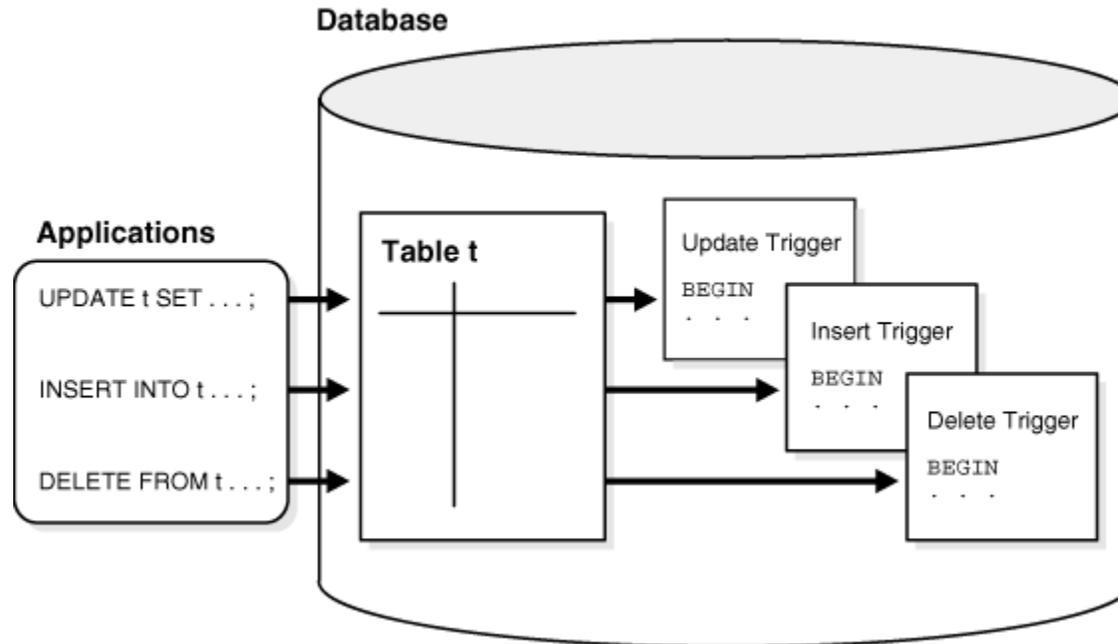
```
CREATE [ OR ALTER ] TRIGGER [ schema_name . ]trigger_name  
ON { table | view }  
[ WITH <dml_trigger_option> [ ,...n ] ]  
{ FOR | AFTER | INSTEAD OF }  
{ [ INSERT ] [ , ] [ UPDATE ] [ , ] [ DELETE ] } [ WITH APPEND ] [ NOT FOR REPLICATION ]  
AS { sql_anweisung [ ; ] [ ,...n ] | EXTERNAL NAME <method specifier [ ; ] > }
```

```
<dml_trigger_option> ::= [ ENCRYPTION ]  
[ EXECUTE AS Clause ]
```

```
<method_specifier> ::= assembly_name.class_name.method_name
```

SQL

Ablaufschema: **DML-Trigger**



SQL

FOR | AFTER | INSTEAD OF: **DML-Trigger**

FOR

Der **T-SQL FOR-Trigger** wird immer dann ausgeführt, **bevor die eigentliche Aufgabe des Triggers ausgeführt** wird, zum Beispiel das Hinzufügen eines neuen Datensatzes.

AFTER

AFTER gibt an, dass der DML-Trigger nur dann ausgelöst wird, **nachdem alle Vorgänge, die in der den Trigger auslösenden SQL-Anweisung festgelegt sind, erfolgreich ausgeführt wurden**. Alle referenziellen CASCADE-Aktionen und Einschränkungsüberprüfungen müssen ebenfalls erfolgreich ausgeführt worden sein, bevor dieser Trigger ausgelöst wird. AFTER ist die Standardeinstellung, wenn FOR das einzige angegebene Schlüsselwort ist. AFTER-Trigger können für Sichten nicht definiert werden.

INSTEAD OF

Gibt an, dass der **DML-Trigger *anstelle* der auslösenden SQL-Anweisung ausgeführt wird**, wodurch die Aktionen der auslösenden Anweisungen überschrieben werden

SQL

Aufgabenstellung: **DML-Trigger**

Bei einem Hinzufügen und bei einer Änderung eines Datensatzes in der Tabelle **customers (Datenbank Nordwind / Northwind)** soll im entsprechenden Datensatz

das **Erstellt- / Änderungsdatum**, die **Erstellt- / Änderungszeit** und der **User** der den Datensatz erstellt bzw. geändert hat

eingetragen werden

ID	FirmenName	Klassifikation	Erstellt (Datum/Zeit)	Erstellt (User)	Geändert (Datum/Zeit)	Geändert (User)
1	ALFKI Alfreds Futterkiste	NULL	NULL	NULL	NULL	NULL
2	ANATR Ana Trujillo Emparedados y helados	NULL	NULL	NULL	NULL	NULL
3	ANDY Haider	B	2018-08-26 20:14:52.973	AzureAD\AndreasHaider	2018-08-26 20:15:07.393	AzureAD\AndreasHaider
4	ANTON Antonio Moreno Taquería	NULL	NULL	NULL	NULL	NULL
5	AROUT Around the Horn	NULL	NULL	NULL	NULL	NULL
6	BERGS Berglunds snabbköp	NULL	NULL	NULL	NULL	NULL
7	BLAUS Blauer See Delikatessen	A	NULL	NULL	2018-08-26 20:15:58.243	AzureAD\AndreasHaider
8	BLONP Blondesdsi père et fils	NULL	NULL	NULL	NULL	NULL
9	BOLID Bólido Comidas preparadas	NULL	NULL	NULL	NULL	NULL
10	BONAP Bon app'	NULL	NULL	NULL	NULL	NULL
11	BOTTM Bottom-Dollar Markets	NULL	NULL	NULL	NULL	NULL
12	BSBEV B's Beverages	NULL	NULL	NULL	NULL	NULL
13	CACTU Cactus Comidas para llevar	NULL	NULL	NULL	NULL	NULL
14	CENTC Centro comercial Moctezuma	NULL	NULL	NULL	NULL	NULL

Die Abfrage wurde erfolgreich ausgeführt. DESKTOP-HJ82DAL\SQLEXPRESS ... AzureAD\AndreasHaider ... Northwind 00:00:00 92 Zeilen

SQL

Tabelle – customers – Felder hinzufügen **DML-Trigger**

TABELLE dbo.customers

FELD Classification_01 mit 1 alphanumerischen Zeichen,
FELD Create_DateTime als Datum und Zeit-Feld,
FELD Create_User mit 50 alphanumerischen Zeichen,
FELD Modify_DateTime als Datum und Zeit-Feld,
FELD Modify_User mit 50 alphanumerischen Zeichen

SQL

Tabelle – customers – Felder hinzufügen **DML-Trigger**

```
ALTER TABLE dbo.customers
ADD Classification_01 nvarchar(1) DEFAULT NULL,
    Create_DateTime datetime DEFAULT NULL,
    Create_User nvarchar(50) DEFAULT NULL,
    Modify_DateTime datetime DEFAULT NULL,
    Modify_User nvarchar(50) DEFAULT NULL;
```



Erstellen des Triggers „insert & update_customers“ **DML-Trigger**

Abfrage ob der Trigger bereits vorhanden ist, wenn JA -> Löschen

```
CREATE TRIGGER Insert_Customers und Update_Customers  
AFTER INSERT / UPDATE  
SYSDATETIME  
SYSTEM_USER
```

```
--Trigger 'Insert_Customers'  
  
IF OBJECT_ID('Insert_Customers') IS NOT NULL DROP TRIGGER Insert_Customers;  
GO  
  
CREATE TRIGGER Insert_Customers  
ON Customers  
AFTER INSERT  
AS  
BEGIN  
SET NOCOUNT ON;  
UPDATE Customers  
SET Create_DateTime = SYSDATETIME(),  
    Create_User = system_user  
FROM Customers  
JOIN inserted  
ON Customers.CustomerID = inserted.CustomerID  
END
```

```
--Trigger 'Update_Customers'  
  
IF OBJECT_ID('Update_Customers') IS NOT NULL DROP TRIGGER Update_Customers;  
GO  
  
CREATE TRIGGER Update_Customers  
ON Customers  
AFTER UPDATE  
AS  
BEGIN  
SET NOCOUNT ON;  
UPDATE Customers  
SET Modify_DateTime = SYSDATETIME(),  
    Modify_User = system_user  
FROM Customers  
JOIN inserted  
ON Customers.CustomerID = inserted.CustomerID  
END
```

```
-- INSERT Datensatz customers
```

```
INSERT INTO Customers (CustomerID, CompanyName, Classification_01)  
VALUES ('ANDY', 'Haider', 'A');
```

```
-- UPDATE Datensatz customers
```

```
UPDATE Customers  
SET Classification_01 = 'A'  
WHERE CustomerID = 'BLAUS,
```

```
-- UPDATE Datensatz customers
```

```
UPDATE Customers  
SET Classification_01 = 'B'  
WHERE CustomerID = 'ANDY'
```

02_dbo.customers_U...ndreasHaider (52) 02_dbo.customers_U...ndreasHaider (54) 02_dbo.customers_S...ndreasHaider (55) X

```

SELECT
    ID = CustomerID,
    FirmenName = CompanyName,
    Klassifikation = Classification_01,
    [Erstellt (Datum/Zeit)] = Create_DateTime,
    [Erstellt (User)] = Create_User,
    [Geändert (Datum/Zeit)] = Modify_DateTime,
    [Geändert (User)] = Modify_User
FROM
    dbo.Customers
  
```

100 %

Ergebnisse Meldungen

ID	FirmenName	Klassifikation	Erstellt (Datum/Zeit)	Erstellt (User)	Geändert (Datum/Zeit)	Geändert (User)
1	ALFKI	Alfreds Futterkiste	NULL	NULL	NULL	NULL
2	ANATR	Ana Trujillo Emparedados y helados	NULL	NULL	NULL	NULL
3	ANDY	Haider	2018-08-26 20:14:52.973	AzureAD\AndreasHaider	2018-08-26 20:15:07.393	AzureAD\AndreasHaider
4	ANTON	Antonio Moreno Taquería	NULL	NULL	NULL	NULL
5	AROUT	Around the Horn	NULL	NULL	NULL	NULL
6	BERGS	Berglunds snabbköp	NULL	NULL	NULL	NULL
7	BLAUS	Blauer See Delikatessen	NULL	NULL	2018-08-26 20:15:58.243	AzureAD\AndreasHaider
8	BLONP	Blondesddsl père et fils	NULL	NULL	NULL	NULL
9	BOLID	Bólido Comidas preparadas	NULL	NULL	NULL	NULL
10	BONAP	Bon app'	NULL	NULL	NULL	NULL
11	BOTTM	Bottom-Dollar Markets	NULL	NULL	NULL	NULL
12	BSBEV	B's Beverages	NULL	NULL	NULL	NULL
13	CACTU	Cactus Comidas para llevar	NULL	NULL	NULL	NULL
14	CENTC	Centro comercial Moctezuma	NULL	NULL	NULL	NULL
15	CHOPS	Chop-suey Chinese	NULL	NULL	NULL	NULL
16	COMMI	Comércio Mineiro	NULL	NULL	NULL	NULL
17	CONSH	Consolidated Holdings	NULL	NULL	NULL	NULL
18	DRACD	Drachenblut Delikatessen	NULL	NULL	NULL	NULL
19	DUM	Du monde entier	NULL	NULL	NULL	NULL
20	EASTC	Eastern Connection	NULL	NULL	NULL	NULL

Die Abfrage wurde erfolgreich ausgeführt. DESKTOP-HJ82DAL\SQLEXPRESS ... | AzureAD\AndreasHaider ... | Northwind | 00:00:00 | 92 Zeilen

SQL

Verwendete Literatur / Auszüge daraus **DML-Trigger**

- **SQL Server 2012 / T-SQL – Abfragen und Analysen / Skulschus | Widerstein / Comelio GmbH**
- **SQL Server 2014 / Implementierung, Administration und Programmierung mit T-SQL / Skulschus | Widerstein / Comelio GmbH**
- **Abfragen von Microsoft SQL Server 2012 – Original Microsoft Training für Examen 70-461 / Published by Microsoft Press Deutschland**