



DATABASE

Öğr. Gör. Evgin GÖÇERİ

8 Aralık 2014

Tetikleyici (Trigger)

- Bir tablo üzerinde belirli bir olaya bağlı olarak tetiklenip çalışan SQL kodlarına **tetikleyici** (trigger) denir
- Tablo üzerindeki triggerları tetikleyen olaylar *insert, update, delete* olaylarıdır

Trgigger ne zaman kullanılmalıdır ?

- **Değişiklikleri takip etmek için** (Örneğin, kritik kayıtlar üzerinde değişiklik yapan kişiler ve değişiklik miktarını ya da kayıtların takibi tutmak için)
- **Mail atmak gibi işlevleri otomatikleştirmek için** (Örneğin, bir sipariş verildiği anda müşteriye siparişin alındığına dair mesaj göndermek için)
- **Karmaşık iş kurallarını gerçeklemek için** (Örneğin, bazen uygulama veritabanının dışında bir veritabanına karşı yabancı anahtar geçerliliği denetlemek için)
- **Yeni nesne oluşturma işlemlerini, sunucu seviyeli veritabanı tutarlılıklarını sağlamak için**
- **Nesne değişikliklerini takip etmek veya engellemek için**

Trgigger Özellikleri

- Bir trigger birden fazla olayı gerçekleştirebilir. (Örneğin, bir tabloya kayıt eklendiğinde devreye giren bir trigger, bu kayıtları başka yerlere de ekledikten sonra, başka bir tablo üstünde de güncelleme yapabilir)
- Bir trigger birden fazla olay tarafından tetiklenebilir
- SQL Server, trigger'lar için satır satır erişimi desteklemez . Değişen kayıtların neler olduğunu anlamak için SQL Server tarafından iki *sözde tablo* olan *inserted* ve *deleted* tabloları kullanılır.
- SQL Server, Oracle VTYS'de yer alan BEFORE trigger'ına destek vermez. Fakat bu, SQL Server'da triggerler her zaman olaylardan sonra devreye girer demek değildir. SQL Server 2000 ile gelen INSTEAD OF trigger'lar gerçek tablolar değişiklikten etkilenmeden önce devreye girerler.
- SQL Server'da iki çeşit trigger vardır : AFTER ve INSTEAD OF.
AFTER trigger'ı sadece tablolar üstünde tanımlanabilir.
INSTEAD OF trigger'ı hem tablolar hem de view'lar üstünde tanımlanabilir.

Trgigger Özellikleri (devam)

- **AFTER** trigger'ı ile **INSTEAD OF** trigger'ı arasındaki en önemli fark: **AFTER** triggerı tetiklendiğinde ilgili olayın olmuş olduğu kabul edilir ve akış bu trigger'a bırakılır. Fakat **INSTEAD OF** triggerı istenilen işlemi gerçekleştirmez. Sadece sözde tablolara kaydetmekle yetinir.
- **FOR** trigger'ı **AFTER** trigger ile aynı anlamda kullanılmakta olup, **SQL Server**'ın eski sürümleri ile uyumluluğu devam ettirmek için desteklenmeye devam edilmektedir.

Trigger Nasıl Çalışır ?

- Triggerı tetikleyen bir işlem ile karşılaşıldığında, işlem ile trigger ikisi birlikte bir transaction bloğu olarak ele alınır (Yani trigger kendisini çağıran olayı onaylamak manasında hiçbir şey yapmayabilir, ya da ROLLBACK ile geçersiz sayılabilir)
- Triggerlar çalıştığı zaman inserted ve deleted tablolarını kullanılır. Bu tablolar ana tablo ile (yani triggerın tetiklendiği tablo ile) eşdeğer sahalara sahiptir. Bu tablolar mantıksal tablo şeklinde RAM'de bulunur.
- Gerçek tabloya bir kayıt eklendiği zaman bu kayıt aynı zamanda inserted tablosuna da eklenir. İhtiyaç olduğunda, yeni eklenen kayıtlara bu tablo üzerinden erişilerek analiz yapılır
- Tablodan bir kayıt silindiğinde ise, silinen kayıt deleted tablosuna aktarılır.
- Bir kayıt güncellendiğinde ise, orijinal kayıt deleted tablosuna aktarılır. Değişen kayıtda inserted tablosuna aktarılır (ve gerçek tabloda da saklanır)

Trigger Oluşturmak

Genel olarak trigger oluşturma ifadesi;

CREATE TRIGGER trigger_adi

ON tablo_adi

[WITH seçenekler]

{FOR | AFTER | INSTEAD OF } {INSERT | UPDATE | DELETE}

AS

Otomatik olarak çalışacak SQL ifadeleri

Trgigger Oluşturmak (devam)

- Trigger içinde şu ifadelere yer verilmez:

CREATE DATABASE, ALTER DATABASE, DROP DATABASE, DISK
INIT, DISK RESIZE, RESTORE DATABASE, RESTORE LOG

Örnek :

Yeni bir sipariş verildiği zaman ali@hotmail.com adresine mail gönderecek bir trigger :

```
CREATE TRIGGER siparisGeldi
ON tblSiparis
FOR INSERT
AS
DECLARE @faturaKod BIGINT, @msg VARCHAR(255)
SELECT @faturaKod = inserted.faturaKod
FROM inserted

SET @msg = 'Yeni siparis alındı. Fatura Kodu : ' + CAST(FaturaKod AS
    VARCHAR(16) )
EXEC master..xp_sendmail 'ali@hotmail.com', @msg
```

Uygulama Dersinde Yapılacak Örnekler

INSERT Trigger'ı,
UPDATE Trigger'ı
DELETE Trigger'ı,
INSTEAD OF Trigger'ı