

EXISTS VE NOT EXISTS fonksiyonları

Kullanımı

EXISTS (sorgu) : eğer sorgudan bir yada daha fazla kayıt dönerse değeri true (doğru), hiç kayıt dönmez ise değeri false (yanlış) tır.

NOT EXISTS : exists fonksiyonunun olumsuzudur.

```
IF EXISTS (select * from Satis where  
Sipno=1)  
    PRINT 'Kayıt var'  
ELSE  
    PRINT 'Kayıt yok'
```

TRIGGER

Bir çeşit prosedürdür. Ancak çalışması için, bir tabloya veri eklenmesi veya veri değişmesi veri silinmesi yeterlidir. Bu üç durumdan biri gerçekleştiğinde otomatik olarak çalışır.

Herhangi bir tablo üzerinde yapılan, insert, delete, update işlemi sonrasında otomatik olarak aynı tablo veya başka bir tabloda veri ekleme, veri değiştirme veya veri silme işlemi yapılmak istendiğinde trigger kullanılır.

2 çeşit trigger var

* After trigger: insert, update ve delete işlemi sonrasında çalışan trigger

* instead of trigger : insert , update, delete işlemi öncesinde çalışır.

Örnek :

Satış tablosuna yeni veri girildiğinde çalışacak ve “veri eklendi” mesajı verecek bir trigger yazınız.

```
CREATE TRIGGER trgSatisGir
ON Satis
AFTER insert
AS
    PRINT ‘Veri eklendi’
```

Çalıştırmak için aşağıdaki kodu yazalım
INSERT Satis (SipNo, UrunNo, miktar,
GirisCikis) VALUES (100,'G',15,0)

Son Silinen, değişen veya eklenen bilgiye erişmek

Bu verilere erişmek için iki tane sanal tablo kullanılırç

- inserted tablosu
- deleted tablosu

Bu tablolar içinde neler bulunur?

- Bir tabloya insert deyimi ile veri girildiğinde, girilen veri inserted tablosunda bulunur.
- Bir tabloda delete deyimi ile veri silindiğinde, silinen bilgiler deleted tablosunda bulunur
- Bir tabloda update deyimi ile veri değiştiğinde, verinin ilk hali deleted tablosunda, son hali inserted tablosunda bulunur.

Örnek:

Satis tablosu üzerinde çalışan bir trigger yazınız. Bu trigger insert, update ve delete durumunda çalışsın ve inserted , deleted tablosu içeriğini listelesin.

```
CREATE TRIGGER trgSatis
ON Satis
AFTER insert, update, delete
AS
BEGIN
    SELECT * FROM inserted
```

```
SELECT * FROM deleted  
END
```

Aşağıdaki kodları yazarak triggerin çalışmasını sağlayalım

```
INSERT Satis(SipNo,UrunNo, miktar)  
values (2500,'WWW',1750)
```

```
UPDATE Satis SET  
UrunNo='tttt',miktar=1750 where  
SipNo=2500
```

```
DELETE FROM Satis WHERE  
SipNo=2500
```

Örnek:

Satış tablosuna veri girildiğinde çalışan bir trigger yazınız. Bu trigger girilen ürün, ürün tablosunda yok ise bu kaydı geri alsın.

Trigger içinde çalışan kodu yazalım.

```
-- UrunNo değişkenini tanımlayalım  
DECLARE @UrunNo varchar(10)  
-- son girilen kayıttak, ürünno bilgisini  
-- @UrunNo değişkenine atayalım
```

```
SELECT @UrunNo=UrunNo from  
inserted
```

```
-- @UrunNo ürün tablosunda yok mu?
```

```
IF NOT EXISTS (select * from urun  
whereUrunNo=@UrunNo)
```

```
BEGIN
```

```
-- Geri al ve mesaj ver
```

```
Print @UrunNo + ' Urun tablosunda  
yok'
```

```
ROLLBACK TRAN
```

```
END
```

Tümünü tek kod halinde yazalım.

```
CREATE TRIGGER trgSatisGir
ON Satis
AFTER insert
AS
BEGIN
    -- Değişken tanımı
    DECLARE @UrunNo varchar(10)
    -- Değişkene son girilen ürünü ata
    SELECT @UrunNo=UrunNo from
inserted

    IF not exists(select * from Urun where
UrunNo=@UrunNo)
    Begin
        Print @UrunNo+ ' Urun tablosunda
yok'
        ROLLBACK TRAN
    end
END
```

Yukarıdaki trigger'ın çalışması için aşağıdaki kod çalıştıralım

```
INSERT Satis(SipNo,UrunNo, miktar)  
values (2900,'B',1750)
```

<http://uygulama.kilicaslan.nom.tr>