

## TRIGGER

Trigger'lar, tablo üzerinde tanımlanabilen ve bu tablo üzerinde bir işlem gerçekleştiğinde tetiklenen programlama öğeleridir. Trigger tanım olarak da tetikleyici anlamındadır.

İki tip trigger vardır.

1. DML Trigger
  - 1.1. AFTER Trigger (FOR Trigger)
  - 1.2. INSTEAD OF TRIGGER
2. DDL Trigger

### DML Trigger

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

### DDL Trigger

Veritabanı üzerinde tablo veya diğer nesneler üzerinde create, alter, drop ifadelerinden sonra tetiklenen triggerlardır.

```
CREATE TRIGGER triger_ad
ON tanımlanacak_tablo_adı
[AFTER/INSTEAD OF] [INSERT|UPDATE|DELETE]
AS
    yapılacak_işlemler
```

Örnek:

Satış tablosuna ürün satıldıktan sonra veri eklenince urun tablosundan stok adedini düşüren sorgu

```
CREATE TRIGGER TRG_StokDusur
ON tbl_Satis
AFTER INSERT
AS
    DECLARE @urunId INT
    DECLARE @satisAdedi INT

    SELECT @urunId=urun_id,@satisAdedi=adet FROM inserted

    UPDATE tbl_urun
    SET stokAdet=stokAdet-@satisAdedi
    WHERE urun_id=@urunId
```

Satış tablosuna veri eklemekten önceki ve sonraki halini görüntüleyen sorgu.

```
SELECT * FROM dbo.tbl_urun u
INSERT INTO tbl_satis(mno, urun_id, adet) VALUES (1,1,2)
SELECT * FROM dbo.tbl_urun u
```

Update örneği:

```
CREATE TRIGGER TRG_StokArttir
ON tbl_Siparis
AFTER UPDATE,DELETE
```

AS

```
DECLARE @urunId INT
DECLARE @satisAdedi INT

SELECT @urunId=urun_id,@satisAdedi=satisAdet FROM deleted

UPDATE tbl_urun
SET stokAdet=stokAdet+@satisAdedi
WHERE urun_id=@urunId
```

```
UPDATE tbl_siparis
SET siparisDurum_id = 3
WHERE siparis_id=8
```

```
DELETE FROM tbl_siparis WHERE siparis_id = 3
```

Son silinen, değişen veya eklenen bilgiye erişmek için trigger iki tane sanal tablo kullanır.

- inserted tablosu
- deleted tablosu

Bu tablonun içinde neler bulunur?

- Bir tabloya insert deyimi ile veri girildiğinde, girilen veri inserted tablosunda bulunur.
- Bir tabloda delete deyimi ile veri silindiğinde silinen kayıt deleted tablosunda bulunur.
- Bir tabloda update deyimi ile veri değiştiğinde, verinin değişmeden önceki hali deleted tablosunda, değiştikten sonraki hali inserted tablosunda bulunur.

Örnek: Satis tablosu üzerinde insert, update ve delete durumunda çalışan ve inserted ve deleted tablosu içeriğini veren bir trigger yazınız.

```
CREATE TRIGGER trgOgrenci
ON tbl_ogrenci
AFTER insert, update, delete
AS
BEGIN
    SELECT * FROM inserted
    SELECT * FROM deleted
END
```

Aşağıdaki sorguları çalıştırarak triggerın çalışmasını sağlayalım

```
INSERT INTO tbl_ogrenci (ogrNo, tcNo, ad, soyad, dogumYeri, bolum_id,
dogumTarihi,medeniDurum)
VALUES (13759013,1, 'zeki', 'müren', 'denizli',1, '01.01.1980',1)

DELETE FROM tbl_ogrenci WHERE ogrNo=13759013

UPDATE tbl_ogrenci
SET ad = 'abdurahman'
WHERE ogrNo=1012487
```

Örnek:Instead of triggerına bir örnek verecek olursak. Öğrenci tablomuzdan herhangi bir öğrencinin hiçbir şekilde silinmemesini sağlayalım.

Öğrenci silinmek istendiğinde öğrencinin durumunu pasif yapsın.

```

CREATE TRIGGER TRG_PasifYap
ON tbl_ogrenci

INSTEAD OF DELETE  --INSERT , UPDATE , DELETE
AS
    DECLARE @ogr_id INT

    SELECT @ogr_id=ogr_id FROM DELETED

    UPDATE tbl_ogrenci
    SET durum=0
    WHERE ogr_id=@ogr_id

    /*  UPDATE ogr SET durum=0
        FROM tbl_ogrenci ogr
        JOIN deleted d ON ogr.ogr_id = d.ogr_id */

CREATE TRIGGER TRG_urunEklemeKontrol
ON tbl_urun
INSTEAD OF INSERT
AS
    INSERT INTO tbl_urun(urunAd,fiyat,stokAdet)
    SELECT urunAd,fiyat,0 FROM inserted

INSERT INTO tbl_urun (urunAd, fiyat, stokAdet)
VALUES('Eldiven',5,500)
SELECT *FROM tbl_urun u

```

Triggerimizi tetikleyecek sorguyu çalıştıralım.

```
DELETE FROM tbl_ogrenci WHERE ogr_id=42
```

Oluşturulan triggerların bir süre durdurulması (pasif edilmesi) istenirse

```
DISABLE TRIGGER trg_adi --on database
```

Tekrar aktif etmek için ise

```
ENABLE TRIGGER trg_adi --on database
```

Buraya kadar olan örnekler DML triggerlar için verilen örneklerdi.  
Şimdi DDL Triggerlar için bir örnek yazarsak:

```

CREATE TRIGGER tr_TabloSilinmesin ON database
FOR DROP_TABLE
AS
    Print 'Veritabanından tablo silinemez'
    ROLLBACK

```

Triggerimizi tetikleyen sorgumuzu çalıştıralım.  
DROP TABLE adresler

Oluşturduğumuz triggeri veritabanından silmek istersek  
`DROP TRIGGER tr_TabloSilinmesin ON DATABASE`