

UPDATE () Fonksiyonu
insert yada update deyimi ile bir kolondaki
veri deđiřtiđinde deđeri true(dođru) ,
deđiřmediđinde ise deđeri false(yanlıř) dır.

Örnek miktar kolonu deđiřtiđinde ki durum
için trigger içinde kod yazalım.

```
IF UPDATE(miktar)
BEGIN
    ..... SQL KODLARI
END
```

Örnek

Satıř tablosunda veri deđiřtiđinde çalıřacak
bir trigger yazınız. Miktar kolonunda veri
deđiřtiđinde deđiřen ürünü, ürün tablosuna
ekleyen bir trigger yazınız.

Çözüm:

İhtiyaçlar

Miktar, ürünno

Değişken tanımlama

```
DECLARE @miktar float, @UrunNo  
varchar(10)
```

-- Değişen kayıtlardaki miktar ve ürünno bilgisini tanımladığımız değişkenlere aktaralım.

```
Select @miktar=miktar,  
@UrunNo=UrunNo from inserted
```

-- miktar kolonu değişti mi?

```
IF UPDATE(miktar)
```

```
BEGIN
```

```
    Print 'miktar kolonundaki veri değişti'
```

```
    INSERT Urun (UrunNo, StokMiktar)
```

```
values (@UrunNo, @miktar)
```

```
END
```

Tümünü yazalım

```
CREATE TRIGGER trgSatisDegistir
```

```
ON Satis
AFTER update
As
BEGIN
    -- değişken tanımla
    DECLARE @miktar float, @UrunNo
varchar(10)
    -- değişen bilgileri değişkene aktar
    SELECT @miktar=miktar,
@UrunNo=UrunNo from inserted

    -- miktar kolonu değişti mi?
    IF UPDATE(miktar)
    BEGIN
        PRINT 'Miktar değeri değişti'
        INSERT Urun(UrunNo,
StokMiktar) values (@UrunNo, @miktar)
    END
END
```

Aşağıdaki kodlarla triggerın çalışmasını sağlayınız

```
UPDATE Satis SET SipNo=13,  
Miktar=100, GirişÇıkış=1 where  
UrunNo='EH'
```

```
UPDATE Satis SET Miktar=98 where  
UrunNo='EH'
```

2 çeşit trigger vardı

- AFTER trigger idi, uygulamaları bunun üzerinden yaptık
- INSTEAD OF trigger

INSTEAD OF TRIGGER

Bu trigger'ı şöyle özetleyebiliriz.
“Kullanıcı şunu yapmak istiyor ama sen onu yapma bunu yap”

```
CREATE TRIGGER trgSatisGir  
ON Satis  
INSTEAD OF insert
```

AS
PRINT 'Trigger çalıştı'

go

Sorular:

Satis Tablosunda olup, ürün tablosunda bulunmayan ürünleri veren sorguyu yazınız.

Çözüm:

Çözüm kursorlerle olacak.

Satis tablounun her satırında yazılı olan ürün numarasını okuyacağız.

-- kursordeki her satırda bulunan ürün numarasını aktarmak için bir değişken tanımlayacağız.

```
DECLARE @UrunNo varchar(10)
```

```
-- kursorü tanımlayalım
```

```
DECLARE SatisKursor CURSOR FOR  
select UrunNo from Satis
```

```
Open SatisKursor
```

```
-- ilk kayda konumlan
```

```
FETCH NEXT FROM SatisKursor INTO  
@UrunNo
```

```
WHILE @@fetch_status=0  
BEGIN
```

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

```
        PRINT @UrunNo + ' urun  
tablosunda yok'
```

```
        FETCH NEXT FROM SatisKursor  
INTO @UrunNo  
END
```

CLOSE SatisKursor DEALLOCATE SatisKursor

Soru :

Fatura tablosunda veri değiştiren bir stored procedure yazınız. UrunNo ve mno su bilinen kayıtlarda, miktar ve fiyatını değiştiren ve kaç kayıt değişti bilgisini parametreden döndüren bir prosedure yazınız.

Çözüm:

Parametreler:

Urunno ve mno

Miktar ve fiyat

KayitSayısı

Parametreleri gerekli

CREATE PROCEDURE spDegistir

@UrunNo varchar(10),

@mno int,

@miktar float,

```
@fiyat Money,  
@KayitSayisi int output  
As  
Update Fatura SET miktar=@miktar,  
fiyat=@fiyat WHERE mno=@mno and  
UrunNo=@UrunNo  
SELECT @KayitSayisi=@@rowcount  
Go
```