

SQL PROGRAMLAMA

BATCH

Bir batch , bir arada bulunan bir dizi SQL deyimidir. Batch ayırıcı GO deyimidir.

```
SELECT ..... }
.....          }
.....          } BATCH
UPDATE ..... }
.....          }
GO            }
.....          }
DELETE ..... }
.....          }
GO            }
.....          }
.....          }
.....          }
```

Özellikleri

- 1- Bir batch içinde bir deyimde yazım hatası olduğunda tüm batch çalışmaz hale gelir.
- 2- Aynı batch içinde bir tablonun yapısı değiştirilip kullanılamaz.

Örnek:

```
CREATE DATABASE db_deneme
-----
USE db_deneme

CREATE TABLE tbl_deneme3 (asd NVARCHAR(50), def INT
                           IDENTITY(1,1) PRIMARY KEY, ghi NVARCHAR(50))
```

Örnek:

```
CREATE TABLE urun .....
INSERT urun .....
GO
```

Yukarıdaki kod çalışmaz. Çünkü aynı batch içinde tablo oluşturuluyor ve bu tabloya ekleme yapılmak isteniyor.

```
CREATE TABLE urun .....
GO
INSERT urun .....
GO
```

Yukarıdaki kod ise hata vermeden çalışır.

SQL Server'da deęişken tanımı

Sql server'da deęişken isimlerinin başında @ sembolü bulunur.
@ad , @soyad, @tarih deęişken isimlerine birer örnektir.

Deęişken tanımı için DECLARE deyimi kullanılır.

Genel ifade

DECLARE @isim **veritipi**
Şeklinde tanımlanır.

DECLARE @ad **nvarchar**(25), @soyad **nvarchar**(25), @tarih **datetime**
DECLARE @numara **int**, @borc **Money**

@numara =10 şeklinde atama yapılamaz, hata verir.
Atama aşağıdaki gibi yapılır.

```
SELECT @numara=10  
Set @numara=10
```

```
SELECT @ad='ali', @soyad='çoşkun', @borc=50  
Yukarıdaki kodda @numara deęişkenine 10 rakamı, @ad deęişkenine 'ali', @soyad deęişkenine 'çoşkun', @borc deęişkenine 50 deęer atandı.
```

Örnek:

```
DECLARE @UrunAdi varchar(30)  
SELECT @UrunAdi='Defter'  
SELECT * FROM Urun WHERE UrunAdi=@UrunAdi
```

Örnek:

```
DECLARE @PersonelAdi varchar(20)  
SELECT @PersonelAdi = 'Ahmet'  
SELECT * FROM Personel WHERE ad=@PersonelAdi
```

SQL server'da önceden tanımlı pek çok deęişken vardır. Bunlardan iki tanesinden bahsedelim.

@@error : sql deyimi hata vermeden çalışırsa deęeri sıfır, hata verirse sıfırdan farklı bir deęer alır.

@@rowcount : Bir update, delete, insert deyimi çalıştıęında, kaç tane kayıdın etkilendięini bize bildirir.

Yazı Yazdırma:

Yazı yazdırma PRINT deyimi kullanılır.

Örnek:

```
PRINT 'Ali Coşkun'  
PRINT 5  
PRINT 'HATA'  
PRINT @numara  
Print @@error
```

Akış kontrol deyimleri

SQL serverdaki akış kontrol deyimleri aşağıdadır.

- CASE
- IF ELSE
- WHILE
- BREAK
- CONTINUE

CASE deyimi

Case deyimi SELECT, UPDATE, DELETE, SET, IN, WHERE , ORDER BY gibi deyimlerin içinde kullanılır.

Sql içinde belirli durumlara göre farklı değerler üretmek istiyorsak CASE deyimi kullanırız. İki farklı yazım şekli vardır.

Basit CASE yapısı

```
CASE GirişDenklemi
WHEN değer1 THEN sonuç1
WHEN değer2 THEN sonuç2
...
...
ELSE sonuçn
END
```

Örnek 1:

```
Select CASE medeniDurum
        WHEN 1 THEN 'Bekar'
        WHEN 2 THEN 'Evli'
        ELSE 'Bilinmiyor'
        END
```

From tbl_personel

Örnek 2:

```
DECLARE @sayi INT=5

SELECT
CASE @sayi
WHEN 1 THEN 'Zayıf'
WHEN 2 THEN 'Geçer'
WHEN 3 THEN 'Orta'
WHEN 4 THEN 'İyi'
WHEN 5 THEN 'Pekiyi'
ELSE 'Aralık Dışı Değer'
END
```

Yukarıdaki ifade tek başına yazılmaz, bir SELECT deyimi içine yazılır. medeniDurum değeri 1 ise Bekar, 2 ise Evli , hiç biri değilse else tarafında bulunan bilinmiyor değeri gelir.

İkinci CASE yazımı

```
CASE
    WHEN koşul1 THEN sonuç1
    WHEN koşul2 THEN sonuç2
    .....
    ELSE sonuçn
END
```

Örnek 1:

```
SELECT p.adi, p.soyadi,
CASE
    WHEN medeniDurum=1 THEN 'bekar'
    WHEN medeniDurum=2 THEN 'evli'
    WHEN medeniDurum=3 THEN 'dul'
    ELSE 'Bilinmiyor'
END 'MedeniDurum'
FROM tbl_personel p
```

Örnek 2:

```
DECLARE @notu INT=500
SELECT
CASE
    WHEN @notu>=0 AND @notu<50 THEN 'Kaldı'
    WHEN @notu>=50 AND @notu<=100 THEN 'Geçti'
    ELSE 'Aralık Dışı Not değeri'
END
```

Tablo : Personel

SicilNo	Departman	Ad	Soyad	Maas
2053	10	İbrahim	Öz	700
68054	10	Hakan	Bülbül	850
0415	12	Ahmet	Uzun	1200
0524	10	Halil	Doğru	1500
0211	12	Özkan	Zorlu	900
3159	12	Ali	Şimşek	2500
2543	5	Neşe	Şen	1300
2546	5	Alper	Şener	1200

Örnek:

```
SELECT *,
CASE
    WHEN maas<1000 THEN 'maaş Az'
    WHEN maas>1200 then 'maaş çok'
    ELSE 'Maaş Normal'
END AS MaasDurumu,
CASE Departman
    WHEN 10 THEN 'PAZARLAMA'
    WHEN 12 THEN 'ÜRETİM'
    WHEN 5 THEN 'İHRACAT'
END as DepartmanAdi,
CASE Departman
    WHEN 10 THEN maas*1.3
    WHEN 12 THEN maas*1.1
    else maas
end as artmismaas
FROM Personel
```

Ödev: ÖğrenciNot tablosundaki her bir öğrencinin öğrenci Numarasını, adını, soyadını ,ders adını ve 100 lük notuna göre harf notunu döndüren sorguyu yazınız. (100 lük notu harf notuna dönüştürürken case yapısı kullanınız.)

Ödevi alicoskunodev@gmail.com adresine yazdığınız sorgu ogrNo.txt dosyasında olacak ve sorgunun çalıştıktan sonraki ekran görüntüsü ogrNo.jpg dosyasında olacak şekilde maile eklenerek gönderilecektir. 30.10.2013 Çarşamba Saat 23:59 ödevin son gönderme saatidir.

Yüzlük not- Harf Not Dönüşüm Tablosu

90-100	A1	55-59	D1
80-89	A2	50-54	D2
75-79	B1	40-49	E
70-74	B2	0-39	F1
65-69	C1		
60-65	C2		

IF... ELSE... deyimi

Batch içinde koşula bağlı olarak dallanma yapar.

IF koşul
Sql deyimi

Koşul doğru ise devamındaki sql deyimi çalışır. Yanlış ise çalışmaz.

IF koşul
Sql deyimi (1)
ELSE
Sql deyimi (2)

Koşul doğru ise 1. sql deyimi, yanlış ise 2. sql deyimi çalışır.
Eğer sql deyimi bir den fazla satır dan oluşursa BEGIN END arasında yazılır.

Örnek :

Personel tablosunda Ortalama maaş 1000 çok ise “maaş ortalaması yüksek”, az ise “maaş ortalaması düşük yazsın”

```
IF (select avg(maas) from personel) > 1000  
    PRINT 'Maaş ortalaması yüksek'  
ELSE  
    PRINT 'Maaş ortalaması düşük'
```

Örnek :

2053 sicil nolu personelin maaşı ortalamadan yüksek ise “çok maaş”,
Düşük ise “az maaş” yazsın

```
IF (select maas from personel where sicilno=2053) > (select avg(maas) from  
personel)  
    PRINT 'çok maaş'
```

```
ELSE
    PRINT 'az maaş'
```

Örnek:

```
DECLARE @isim varchar(20)
SELECT @isim='Ali'

IF @isim = 'Ali'
    PRINT @isim + ' Coşkun'
DECLARE @ort INT

SET @ort = (SELECT AVG(maas) FROM tbl_personel )
PRINT @ort

IF @ort>2000
BEGIN
    PRINT 'Maas ortalamasi yüksek'
    SELECT @ort
END
ELSE
BEGIN
    PRINT 'düşük'
END
```

Örnek :

```
IF((SELECT COUNT(1)FROM tbl_ogrenciNot on1 WHERE on1.notu IS NULL)>0)

    DELETE FROM tbl_ogrenciNot WHERE notu IS NULL

ELSE
    SELECT * FROM tbl_ogrenciNot on1
```

WHILE döngüsü

While yapısı bir sql deyiminin tekrarlanmasını sağlar. Koşul yanlış oluncaya kadar döngü çalışır.

BREAK : döngüden koşul sağlanmadan yani erken çıkılmasını sağlar.

CONTINUE : devamındaki satırlar işlenmeden döngünün ilk satırına dönülmesini sağlar.

Genel ifade

```
WHILE koşul
    Sql deyimi
```

Veya

```
WHILE koşul
BEGIN
    Sql deyimleri
END
```

Veya

```
WHILE koşul
BEGIN
    Sql deyimi
    Koşula bağlı BREAK deyimi
    Sql deyimi
```

Koşula bağlı `CONTINUE` deyimi
Sql deyimi

END

Örnek: 1 den 12 ye kadar sayıları yazdıralım.

```
DECLARE @sayi int
SELECT @sayi=1

WHILE @sayi<=12
BEGIN
    PRINT @sayi
    SELECT @sayi = @sayi + 1
END
```

Örnek : Önceki örnek gibi, 1'den 30 kadar döngüyü çalıştıralım. 1'den 24'e kadar yazdıralım. Sonrasında döngüden çıkalım.

```
DECLARE @sayi int
SELECT @sayi =1
WHILE @sayi <=30
BEGIN
    PRINT @sayi
    IF @sayi = 24
        BREAK
    SELECT @sayi=@sayi+1
END
```

Örnek : 1 den 30 a kadar yazdıralım. Ancak 15-20 arası yazılmasın.

```
DECLARE @sayi int
SELECT @sayi=0

WHILE @sayi < 30
BEGIN
    SELECT @sayi=@sayi+1
    IF @sayi between 15 and 20
        continue
    PRINT @sayi
END
```