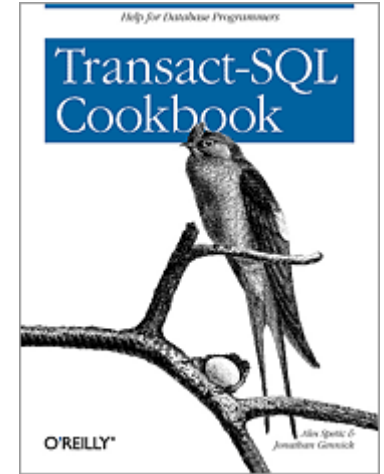




08221 Veri Tabanı II

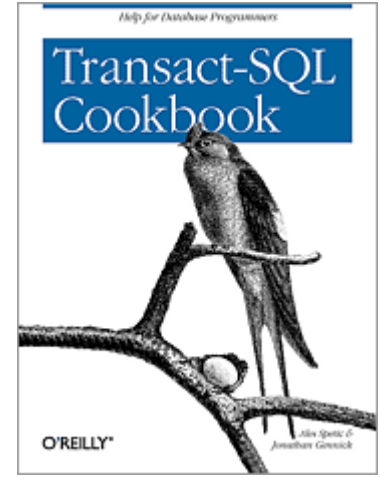
Elbistan Meslek Yüksek Okulu
2015 – 2016 GÜZ Yarıyılı



08221 Veri Tabanı II

T-SQL KOMUTLARI

- Veritabanları ANSI SQL dediğimiz standart dil kullanılmaktadır.
- Bu dil karmaşık işlemler gerçekleştirmeyi, örneğin bir if...else yapısı, while döngüsü gibi fonksiyonları ve değişken tanımlamayı desteklememektedir.





08221 Veri Tabanı II

T-SQL KOMUTLARI

- Bu durumda yapılması gereken farklı bir programlama dili öğrenmek ve bu vasıta ile bu işlevleri yerine getiren ara programlar yazmaktır.
- T-SQL dil yapısı tüm bu fonksiyon ve karmaşık işlemleri SQL Serverda yapabilmek adına geliştirilmiş standart dışı bir dildir.



08221 Veri Tabanı II

T-SQL-DEĞİŞKENLER

- T-SQL kullanmanın en büyük sebeplerinden biri değişken kullanımına olanak tanımasıdır.

DEĞİŞKEN NEDİR ?

08221 Veri Tabanı II

T-SQL-DEĞİŞKENLER

DEĞİŞKEN NEDİR ?

- Değişken diğer tüm programlama dillerinde yer alan bir veri tipi ile sınırlandırılmış, yaratılmasının ardından hafızada belli bir yer kaplayan, üzerine veri ataması yapılabilen ve daha sonra ismi kullanılarak program içerisinde çağrılıp kullanılacak yapıdır.



08221 Veri Tabanı II

T-SQL-DEĞİŞKENLER

- Yazım şekli :

```
declare @degisken_adi <veri_tipi> <(boyut)>
```

şeklindedir.



08221 Veri Tabanı II

T-SQL-DEĞİŞKENLER

- **Yazım Şekli Örnek :**

```
declare @ogr_no Char(8)  
declare @ogr_adi VarChar(30)  
declare @ogr_soyadi VarChar(30)
```



08221 Veri Tabanı II

T-SQL-DEĞİŞKENLER

- Değişkene bir gerçek karşılık atanması durumunda değişkene değer atamak denir.
- **SET** komutu değişkene değer atamakta kullanılır.
- Değişkene atanan değer programın ilerleyen aşamalarında farklı bir değer ile değiştirilebilir.
- Yeni değer atanan eski değer yerine geçer.

08221 Veri Tabanı II

T-SQL-DEĞİŞKENLER

- **Örnek :**
declare @ogr_no Char(8)
Set @ogr_no = 12BP2020
Set @ogr_no = 12BP2030
- //Öğrenci No artık "12BP2030" değerinde dir.

08221 Veri Tabanı II

T-SQL-SİSTEM FONKSİYONLARI

- Veritabanlarında sistem oturum bazında çalışır ve birden fazla oturumu destekleyebilirler.
- Değişkenler oturum bazında değer alırlar.
- Bazen tek bir oturum yerine tüm oturumlarda değer alabilecek ve değer döndürebilecek yapılara ihtiyaç duyulur.
- Bu nedenle SQL Server sistem fonksiyonları vardır.



08221 Veri Tabanı II

T-SQL-SİSTEM FONKSİYONLARI

- Yazımı;

```
Select @@<fonksiyon_adi>
```



08221 Veri Tabanı II

T-SQL-SİSTEM FONKSİYONLARI

- MS SQL Server sistemi hakkında bize bilgi veren hazır sistem fonksiyonlarını öncelikle öğrenelim.
- *Tüm örneklerde "Select" ifadesi kullanılmıştır.*
- *Bu ifade fonksiyon çıktısını ekranda görüntülemek için konulmuştur. T-SQL Clause kullanımlarda "Select" ifadesine yer vermeniz gerekmez.*

08221 Veri Tabanı II

T-SQL-SİSTEM FONKSİYONLARI

Örnek :

Sql serverda açık connection sayısını almak için sistemde daha önce tanımlanmış olan fonk. çalıştırmak.

```
Select @@connections;
```

Bize döndüreceği sonuç;
>>1 veya kaç connection varsa ...

şeklindedir.



08221 Veri Tabanı II

T-SQL-SİSTEM FONKSİYONLARI

Örnek : **CURRENT_USER** : O anda çalışılan kullanıcı bilgisini geri döndürür.

```
Select Current_User
```



08221 Veri Tabanı II

T-SQL-SİSTEM FONKSİYONLARI

Örnek : **@@LANGUAGE** : Sistem dil bilgisini geri döndürür.

```
Select @@Language
```



08221 Veri Tabanı II

T-SQL-SİSTEM FONKSİYONLARI

Örnek : **@@MAX_CONNECTIONS** : Sistemde bağlantı yapabilecek maksimum kullanıcı sayısını verir.

```
Select @@Max_Connections
```




08221 Veri Tabanı II

T-SQL-SİSTEM FONKSİYONLARI

Örnek : **@@SERVERNAME** : Çalışılan sistemdeki SQL adını (instance name) geri döndürür.

```
Select @@ServerName
```



08221 Veri Tabanı II

T-SQL-SİSTEM FONKSİYONLARI

Örnek : **@@SERVICENAME:** Çalışılan servis hakkında bilgi geri döndürür.

```
Select @@ServiceName
```



08221 Veri Tabanı II

T-SQL-SİSTEM FONKSİYONLARI

Örnek : **@@SUSER_NAME():** Sistemde bağlı çalışan kullanıcı adını (adlarını) geri döndürür.

```
Select @@Suser_Name()
```

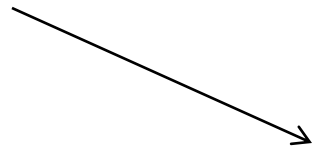
Parametre (?)

08221 Veri Tabanı II

T-SQL-SİSTEM FONKSİYONLARI – DİĞER FONKSİYONLAR

Örnek : **ASCII():** Tek bir parametre alır, parametre olarak girdiğiniz karakterin ascii karşılığını verir.

Select ASCII('A')



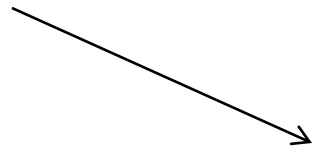
069

08221 Veri Tabanı II

T-SQL-SİSTEM FONKSİYONLARI – DİĞER FONKSİYONLAR

Örnek : **CHAR():** Tek bir parametre alır, parametre olarak girdiğiniz ascii karşılığın karakterini verir.

Select CHAR(65)



A

08221 Veri Tabanı II

T-SQL-SİSTEM FONKSİYONLARI – DİĞER FONKSİYONLAR

Örnek : **AVG():** Tek bir parametre alır, verilen kolon (sütun) bilgisinin ortalamasını verir.

Select AVG(StandartCost) from Production Product

- Toplam Kolon sayısını hesaplar
- Kolonu toplar ve ortalama için kolon sayısına böler.

08221 Veri Tabanı II

T-SQL-SİSTEM FONKSİYONLARI – DİĞER FONKSİYONLAR

Örnek : **CREATE PROCEDURE Skor**

 @score1 smallint
 @score2 smallint,
 @score3 smallint,
 @myAvg smallint OUTPUT
AS SELECT @myAVG= (skor1+skor2+skor3)/3

// Skor değişkenini deklare edelim
DECLARE @Avgskor smallint

EXEC skor 10, 8, 8 @Avgskor OUTPUT
SELECT 'ORTALAMA : ' Avgskor

GO



08221 Veri Tabanı II

T-SQL-SİSTEM FONKSİYONLARI – DİĞER FONKSİYONLAR

Örnek : **SUM():** Tek bir parametre alır, verilen kolon (sütun) bilgisinin toplamını verir.

```
Select AVG(StandartCost) from Production Product
```




08221 Veri Tabanı II

T-SQL-SİSTEM FONKSİYONLARI – DİĞER FONKSİYONLAR

Örnek : **MAX/MIN():** Tek bir parametre alır, verilen kolon Sayı ise en büyük-en küçük, tarih ise en yakın en uzak tarih bilgisini verir.

```
Select MAX(StandartCost) from Production Product
```

```
Select MIN(StandartCost) from Production Product
```



08221 Veri Tabanı II

T-SQL-SİSTEM FONKSİYONLARI – DİĞER FONKSİYONLAR

Örnek : **GETDATE():** O andaki tarih ve saat bilgisini verir.

```
Select GetDate()
```

08221 Veri Tabanı II

T-SQL-SİSTEM FONKSİYONLARI – DİĞER FONKSİYONLAR

Örnek : **DATEADD():** Üç Parametre alır, birinci parametre eklemek istenen zaman birimi, ikinci parametre ne kadar zaman ekleneceği, üçüncü parametre eklenecek zaman için referans zaman değeridir.

```
Select DateAdd(dd,117,'01.11.2012)
```

Gün : d, dd, Ay: m, mm, Yıl: yy, yyyy, Hafta: wk, ww, saat: hh Dakika min, n Saniye s, ss Milisaniye: ms

08221 Veri Tabanı II

T-SQL-SİSTEM FONKSİYONLARI – DİĞER FONKSİYONLAR

Örnek : **DATEDIFF():** Üç Parametre alır, birinci parametre datepart iki tarih arasındaki fark, ikinci parametre başlangıç zamanı, üçüncü parametre ise bitiş zamanıdır.

```
Select DateDiff('10.10.2012','12.12.2012')
```



08221 Veri Tabanı II

T-SQL-SİSTEM FONKSİYONLARI – DİĞER FONKSİYONLAR

Örnek : **DATENAME():** iki Parametre alır, birinci parametre datepart ikincisi ismini gireceğiniz tarih bilgisidir.

```
Select DateName(m, '2012.10.10')
```



08221 Veri Tabanı II

T-SQL-SİSTEM FONKSİYONLARI – DİĞER FONKSİYONLAR

Örnek : **SUBSTRING():** Üç Parametre alır, birinci parametre metnin kendisi, ikinci parametre kaçınıcı karakterden başlayacağı, üçüncü parametre ise kaç karakter alınacağıdır.

```
Declare @ogr_adi Varchar(20)  
Set @ogr_adi='Ahmet Yılmaz'  
Select SUBSTRING(@ogr_adi, 1, 5)
```

Sonuç : Ahmet



08221 Veri Tabanı II

T-SQL-SİSTEM FONKSİYONLARI – DİĞER FONKSİYONLAR

Örnek : **UPPER/LOWER():** Bir parametre alır. Parametre olarak gönderilen text i UPPER Büyük, LOWER Küçük harf yapar.

```
Select UPPER('Muratok')
```

Sonuç : MURATOK



08221 Veri Tabanı II

T-SQL-SİSTEM FONKSİYONLARI – DİĞER FONKSİYONLAR

Örnek : **SQUARE:** Bir parametre alır. Parametre olarak gönderilen sayının karesini geri döndürür.

```
Select SQUARE(5)
```

Sonuç : 25

08221 Veri Tabanı II

T-SQL-SİSTEM FONKSİYONLARI – DİĞER FONKSİYONLAR

Örnek : **DAY/MOUNT/YEAR:** Bir parametre alır.

Parametre olarak gönderilen tarih tipindeki alanın gün ay yıl bilgisini döndürür.

Select DAY(2012.10.10) - Salı

Select MOUNT(2012.10.10) – Ekim

Select YEAR(2012.10.10) – 2012

08221 Veri Tabanı II

T-SQL-SİSTEM FONKSİYONLARI – DİĞER FONKSİYONLAR

Örnek : **COS/SIN/TAN/COT:** Bir parametre alır.

Parametre olarak gönderilen sayının Cos, Sin, Tan, Cot değerini döndürür.

Select COS(25) - ?

Select SIN(25) - ?

Select TAN(25) - ?

Select COT(25) - ?



08221 Veri Tabanı II

T-SQL-PRINT KOMUTU

Değişken durumları, prosedur sonuçları, işlem sonuçları gibi verileri ekrana yazdırmak istediğimizde bu komutu kullanırız.



08221 Veri Tabanı II

T-SQL-PRINT KOMUTU

Örnek :

```
Declare @Toplam int
```

```
Select @Toplam = Count(*) from TBL_NOTLAR
```

```
Print @Toplam
```

```
GO
```

Bu işlemin bize döndüreceği sonuç TBL_NOTLARdaki kayıt sayısıdır ve ekrana aynen yazılır.



08221 Veri Tabanı II

T-SQL-AKIŞ KONTROLLERİ

ANSI SQLde bulunmayan bu yapı bir programlama dili kadar geniş ve etkin kullanıma yetmese de ANSI SQL'e nazaran daha geniş sorgular ve sonuçlar çıkarmanızı sağlayacaktır.



08221 Veri Tabanı II

T-SQL-AKIŞ KONTROLLERİ

T-SQL akış kontrolleri BEGIN ile başlayıp END ile bitmektedir.

Genel Kullanımı;

BEGIN

//Program Kodları

END



08221 Veri Tabanı II

T-SQL-AÇIKLAMA SATIRLARI

T-SQLde bir satırın dikkate alınmamasını istiyorsanız "--" kullanabilir yada /**/ kullanabilirsiniz.

Query Analizer de çalışırken "/*" yapısından sonra gelen alan da dikkate alınmayacaktır.



08221 Veri Tabanı II

T-SQL-ŞARTLI İFADELER

IF.....ELSE

Klasik şartlı ifade kontrolü olan If....Else bloğu T-SQL tarafından da desteklenmektedir.



08221 Veri Tabanı II

T-SQL-ŞARTLI İFADELER

IF(//Şartlar)

BEGIN

//Program Kodları

END

ELSE IF(//Şartlar)

BEGIN

//Program Kodları

END



08221 Veri Tabanı II

T-SQL-ŞARTLI İFADELER

```
Declare @fld_Arac_Model int
Select @fld_Arac_Model = Max(fld_Arac_Model) From
TBL_ARACLAR
If (@fld_Arac_Model = 123)
    BEGIN
        Print "Bu Aracın Markası XXXXX"
    END
Else If(@fld_Arac_Model = 124)
    BEGIN
        Print "Bu Aracın Markası YYYYYYY"
    END
GO
```



08221 Veri Tabanı II

T-SQL-GO (Git) komutu

Sql sunucu birden fazla işlemi aynı talep içinde cevaplayabilir.

GO komutu Sql sunucu gönderilen komut dizisinin sonunu belirtmek için kullanılır.

Aslında T-SQLin bir parçası olmayan kod Sql Server Query Analizer tarafından da algılanır.



08221 Veri Tabanı II

T-SQL-GO Kullanımında Dikkat Edilmesi Gerekenler

- GO komutu diğer T-SQL komutları ile aynı satırda kullanılamaz
- GO komutundan önce tanımlanan bir değişken GO komutundan sonra kullanılamaz. Zira daha önceki kod yığınının sona erdiğini belirtmektedir.
- Kodun sonunda GO kullanılmadığı sürece komutlar sunucuya gönderilmez.



08221 Veri Tabanı II

KOMPLEX SORGULAR

- Sorgulama yaparken hedef sadece istenen verinin elde edilmesidir.
- SQL-de bu amaca hizmet eden ifadeler vardır.
- İlk ifademiz olan GROUP BY ile başlayalım.

08221 Veri Tabanı II

KOMPLEX SORGULAR

Sınavlar tablosundan bir derse ait vize, final ortalamaları alınmak istenince aşağıdaki gibi bir kod yazılabilir.

```
SELECT AVG(VİZE) AS [VİZE ORTALAMA], AVG(FİNAL) AS [FİNAL ORTALAMA]  
FROM SINAVLAR  
WHERE [DERS KODU] = 'BPR255'
```

	VİZE ORTALAMA	FİNAL ORTALAMA
1	57	53

Ancak, bu sorgu ile sadece bir derse ait ortalamayı almış oluruz.

08221 Veri Tabanı II

KOMPLEX SORGULAR

```
SELECT [DERS KODU], AVG(VİZE) AS [VİZE ORTALAMA],  
       AVG(FİNAL) AS [FİNAL ORTALAMA]  
FROM   SINAVLAR  
GROUP BY [DERS KODU]
```

	DERS KODU	VİZE ORTALAMA	FİNAL ORTALAMA
1	BPR106	56	NULL
2	BPR151	70	NULL
3	BPR152	57	NULL
4	BPR251	66	53
5	BPR252	66	53
6	BPR255	57	53

Kayıtları gruplamak (DERSKODU) için GROUPBY ifadesi kullanılır.

AVG, SUM, MAX, MIN, COUNT ifadelerinden biride kullanılmalıdır.

08221 Veri Tabanı II

KOMPLEX SORGULAR

Gruplama GROUP BY ifadesinde koşul kullanmak için HAVING ifadesini kullanırız.

```
SELECT [DERS KODU], AVG(VİZE) AS [VİZE ORTALAMASI]
FROM SINAVLAR
GROUP BY [DERS KODU]
HAVING AVG(VİZE) >= 50
```

	DERS KODU	VİZE ORTALAMASI
1	BPR106	56
2	BPR151	70
3	BPR152	57
4	BPR251	66
5	BPR252	66
6	BPR255	57

Vize notu 50 ye eşit ve büyük olması şartı sonucu.

08221 Veri Tabanı II

KOMPLEX SORGULAR

Sıralama yaparken ORDER BY

```
SELECT  [ÖĞRENCİ NO], VİZE  
FROM    SINAVLAR  
WHERE   [DERS KODU] = 'BPR251'  
ORDER BY VİZE
```

Ders kodu ‘.’ olanların VİZE ye göre küçükten büyüğe sırala

08221 Veri Tabanı II

KOMPLEX SORGULAR

Sıralama yaparken ORDER BY ve DESC - ASC

```
SELECT  [ÖĞRENCİ NO], VİZE  
FROM    SINAVLAR  
WHERE   [DERS KODU] = 'BPR201'  
ORDER BY VİZE DESC
```

```
SELECT  NUMARA, ADI, SOYADI  
FROM    ÖĞRENCİLER  
WHERE   BÖLÜM = 'BPR1'  
ORDER BY SOYADI ASC
```

VİZE- SOYADI ya göre küçükten büyüğe, Büyükten Küçüğe sırala

08221 Veri Tabanı II

KOMPLEX SORGULAR

Sıralama yaparken AND ifadesi çift şart veya NULL

```
SELECT  [ÖĞRENCİ NO], (VİZE*0.4+FİNAL*0.6) AS ORTALAMA
FROM    SINAVLAR
WHERE   [DERS KODU] = 'BPR25'AND FİNAL IS NOT NULL
```

Sorgu yaparken Güncelleme

```
UPDATE  SINAVLAR
SET     ORTALAMA = VİZE*0.4+FİNAL*0.6
WHERE   VİZE IS NOT NULL AND FİNAL IS NOT NULL
```

08221 Veri Tabanı II

KOMPLEX SORGULAR

Birden fazla tablodan sorgulama yapma

```
SELECT * FROM ÖĞRENCİLER, BÖLÜMLER
```

	NUMARA	ADI	SOYADI	BÖLÜM
1	1001001	Özlem	KURT	BPR1
2	1001002	Ersin	ÇOLAK	BPR1

	BÖLÜM KODU	BÖLÜM ADI
1	BPR1	Bilgisayar Programcılığı
2	BPR2	Bilgisayar Programcılığı(i.Ö.)
3	GMI	Gemi İnşaatı

08221 Veri Tabanı II

KOMPLEX SORGULAR

Birden fazla tablodan, sütundan sorgulama yapma

```
SELECT * FROM ÖĞRENCİLER, BÖLÜMLER  
WHERE ÖĞRENCİLER.BÖLÜM = BÖLÜMLER.[BÖLÜM KODU]
```

	NUMARA	ADI	SOYADI	BÖLÜM	BÖLÜM KODU	BÖLÜM ADI
37	1001039	Vedat	ERDURAN	BPR1	BPR1	Bilgisayar Programcılığı
38	1001040	Yakup	YELKENCİ	BPR1	BPR1	Bilgisayar Programcılığı
39	1001041	Ozan	ÖZAYDIN	BPR1	BPR1	Bilgisayar Programcılığı

08221 Veri Tabanı II

KOMPLEX SORGULAR

Sorgulamada View (Görüntü Tablo) Yaratma

```
CREATE VIEW [SINAV SONUCLARI]
AS
SELECT O.NUMARA, O.ADI, O.SOYADI, D.[DERS KODU],
       D.[DERS ADI], S.VİZE, S.FİNAL, S.ORTALAMA
FROM   ÖĞRENCİLER AS O, DERSLER AS D, SINAVLAR AS S
WHERE  O.NUMARA = S.[ÖĞRENCİ NO] AND D.[DERS KODU] = S.[DERS KODU]
```

```
SELECT * FROM [SINAV SONUCLARI] WHERE [DERS KODU] = 'BPR251'
```

	NUMARA	ADI	SOYADI	DERS KODU	DERS ADI	VİZE	FİNAL	ORTALAMA
1	1001001	Özlem	KURT	BPR251	GÖRSEL PROGRAMLAMA I	100	34	60
2	1001002	Ersin	ÇOLAK	BPR251	GÖRSEL PROGRAMLAMA I	42	53	48
3	1001003	Gamze	ALAGÖZ	BPR251	GÖRSEL PROGRAMLAMA I	66	28	43

08221 Veri Tabanı II

KOMPLEX SORGULAR

```
SELECT * FROM [SINAV SONUCLARI]  
WHERE NUMARA = 1001001
```

	NUMARA	ADI	SOYADI	DERS KODU	DERS ADI	VİZE	FINAL	ORTALAMA
1	1001001	Özlem	KURT	BPR106	VERİ TABANI YÖNETİM SİSTEMLERİ I	68	NULL	NULL
2	1001001	Özlem	KURT	BPR151	ALGORITMA VE PROGRAMLAMAYA GİRİŞ I	40	NULL	NULL
3	1001001	Özlem	KURT	BPR152	ALGORITMA VE PROGRAMLAMAYA GİRİŞ II	52	NULL	NULL

Sorgulamada View (Görüntü Tablo) Silme

```
DROP VIEW [SINAV SONUCLARI]
```

Asıl tablo yerinde durur, sadece view silinir.

08221 Veri Tabanı II



DEVAMI HAFTAYA