

6

Aritmetiksel Operatörler ve Hazır Fonksiyonlar

Veritabanı 1

Aritmetiksel Operatörler

- **SELECT** adi,soyadi, maas + maas*10/100 zamliMaas
FROM tbl_personel
- **select** 3*5
- **select** 5+3
- **select** 3*5 ,3+5, 3/5
- **select** 3*5 carpım ,3+5 toplam

Fonksiyonlar

- Kullanıcıya hesaplama ve dönüşüm gibi işlemlerde yardımcı olması için yazılmış hazır fonksiyonlar sql de mevcuttur.
 - `getdate()` → bir fonksiyondur.
- Karakter Kümesi Fonksiyonları
- Sayısal Değerler İçin Fonksiyonlar
- Tarih ve Saat Fonksiyonları
- Dönüşüm Fonksiyonları

Karakter Kümesi Fonksiyonları

- **+** (Birleştirme Operatörü)
- **SUBSTRING()**
- **LEFT()** ve **RIGHT()**
- **LOWER()** ve **UPPER()**
- **LTRIM()** ve **RTRIM()**
- **LEN()**
- **REPLACE()**
- **CHARINDEX()**
- **REVERSE()**

Karakter Kümesi Fonksiyonları -1

- İki Karakter Kümesini Birleştirmek
 - `SELECT dersKod + ' ' + dersAd + ' ' + dersVeren dersBilgileri`
`FROM tbl_ders`
 - **SUBSTRING() Verininin İstenilen Bölümünü Almak**
 - Karakter kümesi içerisinde verinin istenilen bölümünün alınması için kullanılır.
 - Kullanımı `Substring(veri_kümesi, başlangıç, karakter_sayısı)`
 - `SELECT Substring('Ali Coşkun bilgisayar',5,6) → Coşkun`
 - `SELECT ogrNo, ad, soyad , SUBSTRING(ad,1,1)+soyad eposta`
`FROM tbl_ogrenci`

Karakter Kümesi Fonksiyonları - 2

- **Verinin İstenilen Bölümünü Almak – LEFT() ve RIGHT()**
 - LEFT() başlangıcından itibaren kaç karakter alınacağını belirler.
 - RIGHT() verinin sondan kaç karakter alınacağını belirler.
 - Kullanımı LEFT(bilginin_alınacağı_yer,karakter_sayısı)
RIGHT(bilginin_alınacağı_yer,karakter_sayısı)
- **SELECT** ogrNo, ad, soyad , LEFT(ad,1)+RIGHT(ad,1) isminBasHarfiSonHarfi
FROM tbl_ogrenci

Karakter Kümesi Fonksiyonları - 3

- **Büyük-Küçük Harf Dönüşümü - LOWER() ve UPPER()**
 - LOWER() küçük harfe dönüştürür, UPPER() büyük harfe dönüştürür.
 - Kullanımı LOWER(dönüştürülecek_bilgi)
UPPER(dönüştürülecek_bilgi)
 - **SELECT** ad, soyad , **LOWER**(ad) adKucuk, **UPPER**(soyad) soyadBuyuk
FROM tbl_ogrenci

Karakter Kümesi Fonksiyonları -4

➤ Boşlukları Kaldırmak – LTRIM() ve RTRIM()

➤ LTRIM() verinin başındaki boşlukları kaldırır.

➤ RTRIM() verinin sonundaki boşlukları kaldırır.

➤ Kullanımı LTRIM(bilgi)

RTRIM (bilgi)

➤ `SELECT LTRIM(' Ali Coşkun '), RTRIM(' Ali Coşkun ')`

➤ `SELECT LTRIM(RTRIM(' Ali Coşkun '))` herikiYondekiBosluk

Karakter Kümesi Fonksiyonları -5

➤ Karakter Sayısını Bulmak – LEN()

- LEN() veride bulunan karakter sayısını döndürür.
- Kullanımı LEN(bilgi)

➤ `SELECT LEN('Ali Coşkun') →10`

➤ `SELECT ogrNo, ad, soyad , LEN(ad) adUz, LEN(soyad) soyadUz
FROM tbl_ogrenci`

Karakter Kümesi Fonksiyonları -6

➤ İstenilen Bilgiyi Değiştirmek – REPLACE()

- REPLACE() verideki değiştirilecek değerın yerine yeni değeri yazar.
- Kullanımı REPLACE(sutun_adı, değışecek_bilgi, yeni_bilgi)

➤ `SELECT REPLACE('Ali Coşkun','Ali', 'Veli') → Veli Coşkun`

➤ `SELECT adi, soyadi ,
REPLACE(gorevi,'Tekniker', 'Teknisyen') unvan
FROM tbl_personel`

Karakter Kümesi Fonksiyonları -7

➤ Karakter Aramak – CHARINDEX()

- CHARINDEX() text verinin içerisinde başka bir karakter kümesinin yerinin bulunması için kullanılır. Geriye başlangıç noktasını döndürür.
- Kullanımı CHARINDEX(aranacak_bilgi, arama_yeri, başangıç_nokt)

➤ `SELECT CHARINDEX('Coş','Ali Coşkun Coşkun',0) →5`

➤ Tersten Yazdırma – REVERSE()

- REVERSE() aldığı veriyi tersten yazdırır.
- Kullanımı REVERSE(tersten_yazılacak_veri)

➤ `SELECT REVERSE('Ali Coşkun') → nukşoC ilA`

Sayısal Değerler İçin Fonksiyonlar

- **ABS()**
- **POWER()**
- **SQRT()**
- Bölümden Kalanı Bulma (%)
- **ROUND()**
- **FLOOR()** ve **CEILING()**
- **SUM()**
- **AVG()**
- **MAX()** ve **MIN()**
- **COUNT()**

Sayısal Değerler İçin Fonksiyonlar -1

➤ Bir Sayının Mutlak Değerini Alma– ABS()

➤ `SELECT ABS(-1.25) →1.25`

➤ Bir Sayının Kuvvetini Alma– POWER()

➤ `POWER(x,y) --> Xy`

➤ `SELECT POWER(2,3) →23 →8`

➤ Bir Sayının Karekökünü Alma– SQRT()

➤ `SELECT SQRT(4) →2`

➤ Bir Sayının Başka Bir Sayıya Bölümünden Kalanı Bulma– %

➤ Kullanımı `X % Y`

➤ `SELECT 4 % 2 →0`

➤ `SELECT 23 % 10 →3`

Sayısal Değerler İçin Fonksiyonlar -2

➤ En Yakın Tamsayıya Yuvarlama– ROUND()

- ROUND() aldığı ondalık değeri aldığı basamak sayısına göre aşağı veya yukarı yuvarlar.
- Kullanımı ROUND(ondalık_değer, basamakSayısı)

- `SELECT ROUND(2.5,0) →3`
- `SELECT ROUND (2.4,0) →2`
- `SELECT ROUND (2.47,1) →2.5`
- `SELECT ROUND (2.44,1) →2.4`
- `SELECT ROUND (2.018,2) →2.02`
- `SELECT ROUND (2.011,2) →2.01`

Sayısal Değerler İçin Fonksiyonlar -3

- **Aşağı ve Yukarı Yuvarlama– FLOOR() ve CEILING()**
 - FLOOR() aldığı ondalık değeri her koşulda aşağı yuvarlar.
 - CEILING() aldığı ondalık değeri her koşulda yukarı yuvarlar

- **SELECT FLOOR(2.5) →2**
- **SELECT FLOOR(2.3) →2**
- **SELECT FLOOR(2.99) →2**

- **SELECT CEILING(2.5) →3**
- **SELECT CEILING (2.01) →3**
- **SELECT CEILING (2.9) →3**

Sayısal Değerler İçin Fonksiyonlar -4

➤ Toplam Bulma – SUM()

- SUM() fonksiyonu bir sütun için toplam almakta kullanılır.
- Yanına bir sütun daha yazılabilmesi için ya SUM() benzeri bir fonksiyonla kullanılmalıdır yada GROUP BY deyimi ile gruplandırılmalıdır.
- `SELECT SUM(maas) FROM tbl_personel`
- `SELECT SUM(maas) FROM tbl_personel WHERE gorevi= 'Mühendis'`
- `SELECT adi,soyadi,maas, SUM(maas)`
`FROM tbl_personel` → şeklindeki kullanım hata verecektir.
- `SELECT SUM(borc) toplamBorc, SUM(alacak) toplamAlacak`
`FROM tbl_musteri`
- `SELECT SUM(borc) - SUM(alacak) toplamNet, SUM(borc-alacak) net`
`FROM tbl_musteri`

Sayısal Değerler İçin Fonksiyonlar -5

➤ Ortalama Değer Hesaplama – AVG()

- AVG() fonksiyonu bir sütun için ortalama değeri almak için kullanılır.
- Kullanım biçimi sum ile aynıdır.

➤ `SELECT AVG(maas) FROM tbl_personel`

➤ `SELECT AVG(maas) FROM tbl_personel WHERE gorevi= 'Mühendis'`

➤ `SELECT SUM(maas) toplamOdenen , AVG(maas) ortalamaMaas
FROM tbl_personel`

Sayısal Değerler İçin Fonksiyonlar -6

- **En Büyük ve En Küçük Değer Bulma – MAX() ve MIN()**
 - Bir sütunun içerdiği;
en büyük değeri bulmak için MAX() fonksiyonu ,
en küçük değeri bulmak için MIN() fonksiyonu kullanılır.
 - Sayısal veri tiplerinde en büyük yada en küçük değeri bulurlar. Text veri tipinde ascii koduna göre en son yada en ilk gelene göre hesaplar.
 - Kullanım biçimi sum ve avg ile aynıdır.
- `SELECT MAX(maas) FROM tbl_personel`
- `SELECT MIN(maas) FROM tbl_personel`
- `SELECT SUM(maas) toplamOdenen , AVG(maas) ortalamaMaas
, MIN(maas) enDusukMaas, MAX(maas) enYuksekMaas
FROM tbl_personel`

Sayısal Değerler İçin Fonksiyonlar -7

➤ Satır Sayısını Bulma– COUNT()

- COUNT() fonksiyonu bir sütun içerisindeki satır sayısını bulmak için kullanılır.
- Distinct ile kullanılırsa benzersiz kayıtların sayısını verir.
- `SELECT COUNT(per_id) FROM tbl_personel`
- `SELECT COUNT(*) FROM tbl_personel`
- `SELECT COUNT(per_id) FROM tbl_personel WHERE gorevi= 'Mühendis'`
- `SELECT COUNT(DISTINCT soyadi) FROM tbl_personel`
- `SELECT SUM(maas) / COUNT(per_id) ortMaas, AVG(maas) ortMaasAvg FROM tbl_personel`

Tarih ve Saat Fonksiyonları

- **GETDATE()**
- **DATEPART()**
- **DATENAME()**
- **DATEDIFF()**
- **DATEADD()**

Tarih ve Saat Fonksiyonları -1

- **Güncel Tarih ve Saat Bilgisini Alma– GETDATE()**
 - GETDATE() fonksiyonu veritabanının üzerinde bulunduğu serverin tarih ve saatini alır.
 - **SELECT GETDATE()**
- **Peki ya sadece saati yada tarihi almak istiyorsam.**

Tarih ve Saat Fonksiyonları -2

➤ Verilen Tarihten Tipine Göre Veri Döndürme– DATEPART()

- DATEPART() fonksiyonu verilen tarihten almak istediğimiz kısmı döndürür.
- Kullanımı;

DATEPART(alınacak_bilgi,tarih_saat_bilgisi)

Kısım Adı	Açıklama	Örnek SELECT GETDATE()	Dönen Değer 02.04.2012 13:45:20
YEAR	Yıl bilgisini döndürür	SELECT DATEPART(YEAR, GETDATE())	2012
MONTH	Ay bilgisini döndürür	SELECT DATEPART(MONTH, GETDATE())	04
DAY	Gün bilgisini döndürür	SELECT DATEPART(DAY, GETDATE())	02
HOUR	Saati döndürür.	SELECT DATEPART(HOUR, GETDATE())	13
MINUTE	Dakikayı döndürür.	SELECT DATEPART(MINUTE, GETDATE())	45
SECOND	Saniyeyi döndürür.	SELECT DATEPART(SECOND,GETDATE())	20
WEEK	Yılın kaçınıcı haftası olduğunu döndürür.	SELECT DATEPART(WEEK, GETDATE())	14
QUARTER	Yılın kaçınıcı çeyreğinde olduğunu döndürür	SELECT DATEPART(QUARTER,GETDATE())	2

Tarih ve Saat Fonksiyonları -3

➤ Verilen Tarihten Tarih Tipi Adını Döndürme– DATENAME()

- DATENAME() fonksiyonu verilen tarihten almak istediğimiz kısmın ismini döndürür.
- Kullanımı; DATENAME(alınacak_bilgi,tarih_saat_bilgisi)

Kısım Adı	Açıklama	Örnek <code>SELECT GETDATE()</code>	Dönen Değer <code>02.04.2012 13:45:20</code>
MONTH	Ay ismini döndürür	<code>SELECT DATENAME(MONTH, GETDATE())</code>	Nisan
WEEKDAY	Gün bilgisini döndürür	<code>SELECT DATENAME(WEEKDAY, GETDATE())</code>	Pazartesi

Tarih ve Saat Fonksiyonları -4

- İki tarih arasındaki farkı karşılaştırmak– DATEDIFF()
 - DATEDIFF() fonksiyonu verilen iki tarih arasındaki almak istediğimiz bilginin farkını hesaplar.
 - Kullanımı; DATEDIFF(alınacak_bilgi,ilk_tarih,son_tarih)
 - `SELECT DATEDIFF(YEAR,'01.02.2000', '01.01.2012')` →12
 - `SELECT DATEDIFF(MONTH,'01.01.2012', '02.04.2012')` →3
 - `SELECT DATEDIFF(DAY,'01.01.2012', '02.04.2012')` →92

Tarih ve Saat Fonksiyonları -5

➤ Tarihe Belli Değerler Ekleme– DATEADD()

- DATEADD() fonksiyonu belirtilen tarihe belirli gün, ay, yıl vb bilgi eklendiğinde oluşan tarihi hesaplamak için kullanılır.
- Kullanımı; DATEADD(eklenecek_bilgi,eklenme_miktari,tarih)
- `SELECT DATEADD(YEAR,2, '01.01.2012')` →01.01.2014
- `SELECT DATEADD(MONTH,3, '02.04.2012')` →02.07.2012
- `SELECT DATEADD(DAY,29, '02.04.2012')` →01.05.2012

Dönüşüm Fonksiyonları

➤ CAST()

Kullanımı; **CAST** (değer AS veri_tipi)

```
SELECT CAST('Ali Coşkun' AS NVARCHAR(5)) → Ali C
```

➤ CONVERT()

Kullanımı; **CONVERT**(veri_tipi, değer)

```
SELECT CONVERT( INT ,5.45) → 5
```

```
SELECT CONVERT( NVARCHAR(5) ,2010) → 2010
```