



# Veri Tabanı Yönetim Sistemleri

## Bölüm - 6



# İçerik

---

- Fonksiyonlar
  - Tek Satır Fonksiyonlar
    - Karakter Fonksiyonlar
    - Sayısal Fonksiyonlar
    - Tarih ve Saat Fonksiyonları
    - Dönüştürücü Fonksiyonlar
  - Çoklu Satır Fonksiyonlar (Grup Fonksiyonlar)
    - GROUP BY ifadesi



# Fonksiyonlar

---

- Fonksiyonlar veya bir başka deyişle işlevler, SQL'e bazı güçlü özellikler katan olanaklar olarak değerlendirilir.
- SQL fonksiyonları, aynen programlama dillerinde olduğu gibi, kullanıcıya bazı özel işlemleri ve hesaplamaları otomatik olarak sağlar.
- Bir fonksiyon; çeşitli argümanları girdi olarak alıp bir sonuç üreten hazır bileşenlerdir.
- Kullanıcının, fonksiyonun adını ve gerekli argümanları tanımlaması yeterlidir. Bu bilgilere dayalı olarak fonksiyon, bir sonuç döndürür (üretir).
- Fonksiyonlar uygulamadan uygulamaya farklılık gösterebilirler. Bu ders kapsamında MySQL veritabanında kullanılan SQL fonksiyonları kullanılacaktır.



# Fonksiyon Çeşitleri

---

Fonksiyonları iki ana grup altında incelemek mümkündür:

- **Tek Satır Fonksiyonları:** Bu fonksiyonlar tablonun bir satırı için uygulanır ve her satır için bir sonuç üretilir.
- **Çoklu Satır Fonksiyonları:** Bu fonksiyonlar, bir grup satıra uygulanır ve sonuç buna göre elde edilir. Verileri gruplayarak analiz edebilmek çoklu satır fonksiyonlarının kullanımı ile mümkün olabilir.



# Tek Satır Fonksiyonları

---

Tek satır fonksiyonları (genelde sadece “Fonksiyon” olarak adlandırılır), tablonun her bir satırına uygulanan fonksiyonlardır. Tek satır fonksiyonlarının **Karakter, Tarih, Sayısal, Dönüştürme** gibi birçok türü bulunmaktadır.

Fonksiyonların kullanım şekli aşağıda belirtildiği gibidir.

---

**Fonksiyon\_adı(sütun|ifade, [argümanlar])**

**Fonksiyon:** Herhangi bir SQL fonksiyonunu simgeler. Örneğin LCASE bir fonksiyon ismidir. Bu ismi değiştiremeyiz.

**Sütun:** Herhangi bir tabloya ilişkin sütun adı.

**İfade:** Herhangi bir karakter dizisi ya da hesaplama ifadesi.

**Argümanlar:** Fonksiyonla birlikte kullanılacak argümanlar.



# Karakter Fonksiyonları

---

Karakter fonksiyonları; karakter verileri girdi olarak alan, karakter ya da sayısal değerler döndürebilen fonksiyonlardır.

Bu fonksiyonların en çok kullanılanlarını şu şekilde sıralayabiliriz;

<b>LCASE</b>	Alfabetik karakterleri küçük harflere dönüştür.
<b>UCASE</b>	Alfabetik karakterleri büyük harflere dönüştür.
<b>CONCAT</b>	Katarları (karakter dizileri) birleştirmek için kullanılır.
<b>SUBSTRING</b>	Bir katarın <b>m.</b> pozisyonundan itibaren <b>n</b> karakterinden yeni bir katar oluşturur.
<b>LENGHT</b>	Bir sütunun ya da katarın içerdiği karakter sayısını döndürür.



# Karakter Fonksiyonları: LCASE

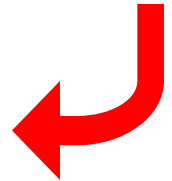
<u>adsoyad</u>	<u>notu</u>	<u>sinif</u>
Orçun Madran	70	4
Ahmet Kemali	80	3
Osman Musa	60	2
Hatice Kurt	90	1
Veli Faydalı	90	1
Ayşe Şahmeran	67	1
Mustafa Atlı	56	2
Zehra Kamuran	34	1
Oktay Bağcı	69	4
Demir Bülbül	24	4
Fatma Yerlisu	38	3
Hulusi Ay	86	3
Mahmut Şahin	100	4
Duygu Mutlu	90	4

Örnek tablomuzda (veritest) yer alan öğrencilerin ad ve soyad bilgilerini tamamı küçük harflerden oluşacak şekilde döndürmek istiyoruz.

SQL cümlecği aşağıdaki gibi olmalıdır.

```
SELECT LCASE(adsoyad) AS kucuk  
FROM veritest
```

<u>kucuk</u>
orçun madran
ahmet kemali
osman musa
hatice kurt
veli faydalı
ayşe şahmeran
mustafa atlı
zehra kamuran
oktay bağcı
demir bülbül
fatma yerlisu
hulusi ay
mahmut şahin
duygu mutlu





# Karakter Fonksiyonları: UCASE

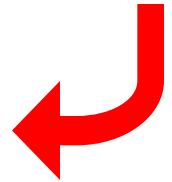
<b>adsoyad</b>	<b>notu</b>	<b>sinif</b>
Orçun Madran	70	4
Ahmet Kemali	80	3
Osman Musa	60	2
Hatice Kurt	90	1
Veli Faydalı	90	1
Ayşe Şahmeran	67	1
Mustafa Atlı	56	2
Zehra Kamuran	34	1
Oktay Bağcı	69	4
Demir Bülbül	24	4
Fatma Yerlisu	38	3
Hulusi Ay	86	3
Mahmut Şahin	100	4
Duygu Mutlu	90	4

Örnek tablomuzda (veritest) yer alan öğrencilerin ad ve soyad bilgilerini tamamı büyük harflerden oluşacak şekilde döndürmek istiyoruz.

SQL cümlecği aşağıdaki gibi olmalıdır.

```
SELECT UCASE(adsoyad) AS buyuk  
FROM veritest
```

<b>buyuk</b>
ORÇUN MADRAN
AHMET KEMALİ
OSMAN MUSA
HATİCE KURT
VELİ FAYDALI
AYŞE ŞAHMERAN
MUSTAFA ATLI
ZEHRA KAMURAN
OKTAY BAĞCI
DEMİR BÜLBÜL
FATMA YERLİSU
HULUSİ AY
MAHMUT ŞAHİN
DUYGU MUTLU







# Karakter Fonksiyonları: CONCAT

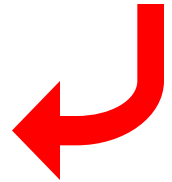
<u>adsoyad</u>	<u>notu</u>	<u>sinif</u>
Orçun Madran	70	4
Ahmet Kemali	80	3
Osman Musa	60	2
Hatice Kurt	90	1
Veli Faydalı	90	1
Ayşe Şahmeran	67	1
Mustafa Atlı	56	2
Zehra Kamuran	34	1
Oktay Bağcı	69	4
Demir Bülbül	24	4
Fatma Yerlisu	38	3
Hulusi Ay	86	3
Mahmut Şahin	100	4
Duygu Mutlu	90	4

Örnek tablomuzda (veritest) yer alan öğrencilerin ad soyad ve not bilgilerini tek bir sütunda birleştirmek istiyoruz.

SQL cümlecği aşağıdaki gibi olmalıdır.

```
SELECT CONCAT(adsoyad, ", notunuz: ", notu)
AS sonuc
FROM veritest
```

<u>sonuc</u>
Orçun Madran, notunuz: 70
Ahmet Kemali, notunuz: 80
Osman Musa, notunuz: 60
Hatice Kurt, notunuz: 90
Veli Faydalı, notunuz: 90
Ayşe Şahmeran, notunuz: 67
Mustafa Atlı, notunuz: 56
Zehra Kamuran, notunuz: 34
Oktay Bağcı, notunuz: 69
Demir Bülbül, notunuz: 24
Fatma Yerlisu, notunuz: 38
Hulusi Ay, notunuz: 86
Mahmut Şahin, notunuz: 100
Duygu Mutlu, notunuz: 90





# Karakter Fonksiyonları: SUBSTRING

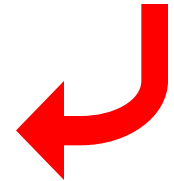
<u>adsoyad</u>	<u>notu</u>	<u>sinif</u>
Orçun Madran	70	4
Ahmet Kemali	80	3
Osman Musa	60	2
Hatice Kurt	90	1
Veli Faydalı	90	1
Ayşe Şahmeran	67	1
Mustafa Atlı	56	2
Zehra Kamuran	34	1
Oktay Bağcı	69	4
Demir Bülbül	24	4
Fatma Yerlisu	38	3
Hulusi Ay	86	3
Mahmut Şahin	100	4
Duygu Mutlu	90	4

Örnek tablomuzda (veritest) yer alan öğrencilerin adlarının ilk üç harfini döndürmek istiyoruz.

SQL cümlecği aşağıdaki gibi olmalıdır.

```
SELECT SUBSTRING(adsoyad, 1, 3) AS uc  
FROM veritest
```

<u>uc</u>
Orç
Ahm
Osm
Hat
Vel
Ayg
Mus
Zeh
Okt
Dem
Fat
Hul
Mah
Duy





# Karakter Fonksiyonları: LENGHT

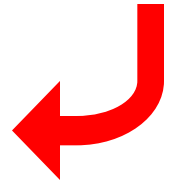
<u>adsoyad</u>	<u>notu</u>	<u>sinif</u>
Orçun Madran	70	4
Ahmet Kemali	80	3
Osman Musa	60	2
Hatice Kurt	90	1
Veli Faydalı	90	1
Ayşe Şahmeran	67	1
Mustafa Atlı	56	2
Zehra Kamuran	34	1
Oktay Bağcı	69	4
Demir Bülbül	24	4
Fatma Yerlisu	38	3
Hulusi Ay	86	3
Mahmut Şahin	100	4
Duygu Mutlu	90	4

Örnek tablomuzda (veritest) yer alan öğrencilerin ad ve soyadlarının toplam karakter sayısını döndürmek istiyoruz.

SQL cümlecği aşağıdaki gibi olmalıdır.

```
SELECT adsoyad, LENGTH(adsoyad) AS sayi  
FROM veritest
```

<u>adsoyad</u>	<u>sayi</u>
Orçun Madran	13
Ahmet Kemali	12
Osman Musa	10
Hatice Kurt	11
Veli Faydalı	13
Ayşe Şahmeran	15
Mustafa Atlı	13
Zehra Kamuran	13
Oktay Bağcı	13
Demir Bülbül	14
Fatma Yerlisu	13
Hulusi Ay	9
Mahmut Şahin	13
Duygu Mutlu	11





# Sayısal Fonksiyonlar

---

Sayısal veriler üzerinde çeşitli işlemleri gerçekleştirmek üzere özel fonksiyonlar bulunmaktadır. Sayısal fonksiyonlar, sayısal veriler alır ve yine sayısal sonuçlar üretirler.

Bu fonksiyonların en çok kullanılanlarını şu şekilde sıralayabiliriz;

<b>ROUND</b>	Sayısal değerleri yuvarlatmak amacıyla kullanılır.
<b>TRUNC</b>	Sayısal değeri belirtilen ondalığa göre budamak (kesmek) amacıyla kullanılır.
<b>MOD</b>	İki sayısal değer, $m$ ve $n$ 'nin birbirine bölümünden elde edilen kalanı döndürür.



# Sayısal Fonksiyonlar: ROUND

<u>adsoyad</u>	<u>notu</u>	<u>sinif</u>
Orçun Madran	70	4
Ahmet Kemali	80	3
Osman Musa	60	2
Hatice Kurt	90	1
Veli Faydalı	90	1
Ayşe Şahmeran	67	1
Mustafa Atlı	56	2
Zehra Kamuran	34	1
Oktay Bağcı	69	4
Demir Bülbül	24	4
Fatma Yerlisu	38	3
Hulusi Ay	86	3
Mahmut Şahin	100	4
Duygu Mutlu	90	4

Örnek tablomuzda (veritest) yer alan öğrencilerin notlarından %5 ceza puanı düşülmesini ve sonucun yuvarlanarak dönmesini istiyoruz.

SQL cümlecığı aşağıdaki gibi olmalıdır.

```
SELECT adsoyad, ROUND(notu/1.05) AS ceza  
FROM veritest
```

<u>adsoyad</u>	<u>ceza</u>
Orçun Madran	66.6667
Ahmet Kemali	76.1905
Osman Musa	57.1429
Hatice Kurt	85.7143
Veli Faydalı	85.7143
Ayşe Şahmeran	63.8095
Mustafa Atlı	53.3333
Zehra Kamuran	32.3810
Oktay Bağcı	65.7143
Demir Bülbül	22.8571
Fatma Yerlisu	36.1905
Hulusi Ay	81.9048
Mahmut Şahin	95.2381
Duygu Mutlu	85.7143

ROUND

<u>adsoyad</u>	<u>ceza</u>
Orçun Madran	67
Ahmet Kemali	76
Osman Musa	57
Hatice Kurt	86
Veli Faydalı	86
Ayşe Şahmeran	64
Mustafa Atlı	53
Zehra Kamuran	32
Oktay Bağcı	66
Demir Bülbül	23
Fatma Yerlisu	36
Hulusi Ay	82
Mahmut Şahin	95
Duygu Mutlu	86



# Sayısal Fonksiyonlar: TRUNCATE

adsoyad	notu	sinif
Orçun Madran	70	4
Ahmet Kemali	80	3
Osman Musa	60	2
Hatice Kurt	90	1
Veli Faydalı	90	1
Ayşe Şahmeran	67	1
Mustafa Atlı	56	2
Zehra Kamuran	34	1
Oktay Bağcı	69	4
Demir Bülbül	24	4
Fatma Yerlisu	38	3
Hulusi Ay	86	3
Mahmut Şahin	100	4
Duygu Mutlu	90	4

Örnek tablomuzda (veritest) yer alan öğrencilerin notlarından %5 ceza puanı düşülmesini ve sonucun tek ondalık basamak ile dönmesini istiyoruz. SQL cümlecği aşağıdaki gibi olmalıdır.

```
SELECT adsoyad, TRUNCATE(notu/1.05, 1) AS ceza FROM veritest
```

adsoyad	ceza
Orçun Madran	66.6667
Ahmet Kemali	76.1905
Osman Musa	57.1429
Hatice Kurt	85.7143
Veli Faydalı	85.7143
Ayşe Şahmeran	63.8095
Mustafa Atlı	53.3333
Zehra Kamuran	32.3810
Oktay Bağcı	65.7143
Demir Bülbül	22.8571
Fatma Yerlisu	36.1905
Hulusi Ay	81.9048
Mahmut Şahin	95.2381
Duygu Mutlu	85.7143

TRUNCATE

adsoyad	ceza
Orçun Madran	66.6
Ahmet Kemali	76.1
Osman Musa	57.1
Hatice Kurt	85.7
Veli Faydalı	85.7
Ayşe Şahmeran	63.8
Mustafa Atlı	53.3
Zehra Kamuran	32.3
Oktay Bağcı	65.7
Demir Bülbül	22.8
Fatma Yerlisu	36.1
Hulusi Ay	81.9
Mahmut Şahin	95.2
Duygu Mutlu	85.7



# Sayısal Fonksiyonlar: MOD

---

İki sayısal değerin birbirine bölünmesi sonucunda elde edilen kalanı döndürmek için; MOD() fonksiyonu kullanılır.

Bu fonksiyon şu şekilde tanımlanır;

**MOD ( $m$ ,  $n$ )**

Burada **m** ve **n** birbirine bölünecek sayısal değerleri ifade etmektedir.

**SELECT MOD (234, 10)** ir.



# Tarih ve Saat Fonksiyonları

---

Tarih ve saat işlemlerinde kullanılan birçok fonksiyon vardır. Bu fonksiyonların en çok kullanılanlarından biri **NOW** fonksiyonudur.

**NOW** fonksiyonunun herhangi bir argümanı bulunmamaktadır. O andaki tarih ve zaman bilgisini elde etmek amacıyla kullanılır.

O anki saat ve tarihi ayrı ayrı alabileceğiniz fonksiyonlar aşağıdaki tabloda yer almaktadır.

<b>NOW</b>	O anki tarih ve saati birlikte döndürür.
<b>CURDATE</b>	O anki tarihi döndürür.
<b>CURTIME</b>	O anki saati döndürür.

<b>SELECT NOW()</b>	2006-04-11 23:07:40
<b>SELECT CURDATE()</b>	2006-04-11
<b>SELECT CURTIME()</b>	23:07:40





# Dönüştürme Fonksiyonları

Karakter, sayısal ve tarih verilerinin birbirlerine dönüştürülmesi ve özellikle biçimlendirme işlemlerinde çeşitli fonksiyonlardan yararlanılır.

Bu fonksiyonlardan tarih ve saat bilgisinin formatlanmasında kullanılan DATEFORMAT fonksiyonunun kullanımı aşağıda yer almaktadır.

<code>SELECT DATE_FORMAT(now(), '%Y')</code>	2006
<code>SELECT DATE_FORMAT(now(), '%y')</code>	06
<code>SELECT DATE_FORMAT(now(), '%m.%d.%Y')</code>	04.11.2006
<code>SELECT DATE_FORMAT(now(), '%H:%i:%s')</code>	23:34:52
<code>SELECT DATE_FORMAT(now(), '%m.%d.%Y - %H:%i:%s')</code>	04.11.2006 - 23:26:13
<code>SELECT DATE_FORMAT(now(), '%M')</code>	April
<code>SELECT DATE_FORMAT(now(), '%W')</code>	Tuesday



# Çoklu Satır Fonksiyonları

---

Tek satır fonksiyonları, tablonun bir satırına uygulanabiliyor ve buna karşılık gelen bir sonuç satırı elde edilebiliyordu.

Bir grup satıra bir fonksiyonun uygulanması söz konusu ise; "**çoklu satır**" ya da bir başka deyişle "**grup fonksiyonlarından**" söz edilir.

Grup fonksiyonları tablonun tüm satırlarına uygulanabildiği gibi, **GROUP BY** sözcüğü yardımıyla da alt gruplara uygulanabilir.

Bir tablonun tüm satırları için grup fonksiyonu uygulanacak ise, **SELECT** deyimi aşağıda belirtildiği biçimde kullanılır;

```
SELECT grup fonksiyonu(sütun)  
FROM tablo  
[WHERE koşul];
```



# Grup Fonksiyonları

<b>AVG</b>	Herhangi bir sütunun içerdiği sayısal değerlerin aritmetik ortalamasını hesaplamak amacıyla; <b>AVG()</b> fonksiyonu kullanılır. Bu fonksiyonun uygulandığı sütunun veri türü sayısal olmalıdır.
<b>SUM</b>	Sütunların içerdiği sayısal değerleri toplamak amacıyla <b>SUM()</b> fonksiyonu kullanılır.
<b>STDDEV</b>	Standart sapma, sayısal verilerin aritmetik ortalamalardan farklarının kareli ortalaması olarak bilinir. Söz konusu hesaplamayı yapan SQL fonksiyonu ise; <b>STDDEV()</b> 'dir.
<b>VARIANCE</b>	Varyans, sayısal değerler arasındaki değişkenliği ölçen bir kavramdır. Standart sapmanın karesi olarak bilinmektedir. Bir dizi sayısal değer için varyansını hesaplamak için <b>VARIANCE()</b> fonksiyonu kullanılır.
<b>MAX ve MIN</b>	Tablonun içerdiği değerlerin en büyük ve en küçük olanlarını bulmak için <b>MIN()</b> ve <b>MAX()</b> fonksiyonları kullanılır. Bu fonksiyonlar herhangi bir veri türüne uygulanabilir.
<b>COUNT</b>	Bir tablodaki kayıtların sayılması amacıyla <b>COUNT()</b> fonksiyonu kullanılır. <b>COUNT(*)</b> fonksiyonu, <b>NULL</b> değerleri de içeren tüm kayıtların sayılmasına neden olur. <b>WHERE</b> ile birlikte kullanılırsa, bu koşula uygun tüm kayıtların sayısının öğrenilmesini sağlar. İkinci biçim tercih edilirse, yani <b>COUNT(sütun)</b> biçiminde kullanılırsa, söz konusu sütunda <b>NULL</b> değerler içermeyen tüm kayıtların sayılmasına neden olur.



# Grup Fonksiyonları: AVG

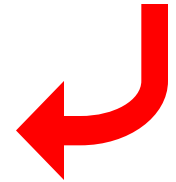
<u>adsoyad</u>	<u>notu</u>	<u>sinif</u>
Orçun Madran	70	4
Ahmet Kemali	80	3
Osman Musa	60	2
Hatice Kurt	90	1
Veli Faydalı	90	1
Ayşe Şahmeran	67	1
Mustafa Atlı	56	2
Zehra Kamuran	34	1
Oktay Bağcı	69	4
Demir Bülbül	24	4
Fatma Yerlisu	38	3
Hulusi Ay	86	3
Mahmut Şahin	100	4
Duygu Mutlu	90	4

Örnek tablomuzda (veritest) yer alan öğrencilerin not ortalamasını döndürmek istiyoruz

SQL cümlecği aşağıdaki gibi olmalıdır.

```
SELECT AVG(notu) AS ortalama  
FROM veritest
```

<b>ortalama</b>
68.1429





# Grup Fonksiyonları: SUM

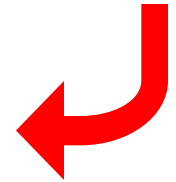
<u>adsoyad</u>	<u>notu</u>	<u>sinif</u>
Orçun Madran	70	4
Ahmet Kemali	80	3
Osman Musa	60	2
Hatice Kurt	90	1
Veli Faydalı	90	1
Ayşe Şahmeran	67	1
Mustafa Atlı	56	2
Zehra Kamuran	34	1
Oktay Bağcı	69	4
Demir Bülbül	24	4
Fatma Yerlisu	38	3
Hulusi Ay	86	3
Mahmut Şahin	100	4
Duygu Mutlu	90	4

Örnek tablomuzda (veritest) yer alan öğrencilerin not toplamını döndürmek istiyoruz

SQL cümlecği aşağıdaki gibi olmalıdır.

```
SELECT SUM(notu) AS toplam  
FROM veritest
```

<b>toplam</b>
954





# Grup Fonksiyonları: MAX

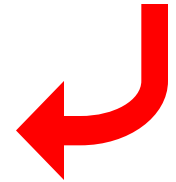
<u>adsoyad</u>	<u>notu</u>	<u>sinif</u>
Orçun Madran	70	4
Ahmet Kemali	80	3
Osman Musa	60	2
Hatice Kurt	90	1
Veli Faydalı	90	1
Ayşe Şahmeran	67	1
Mustafa Atlı	56	2
Zehra Kamuran	34	1
Oktay Bağcı	69	4
Demir Bülbül	24	4
Fatma Yerlisu	38	3
Hulusi Ay	86	3
Mahmut Şahin	100	4
Duygu Mutlu	90	4

Örnek tablomuzda (veritest) yer alan öğrencilerin aldığı en yüksek notu döndürmek istiyoruz.

SQL cümlecığı aşağıdaki gibi olmalıdır.

```
SELECT MAX(notu) AS enyukse  
FROM veritest
```

enyukse
100





# Grup Fonksiyonları: MIN

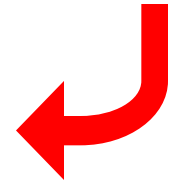
<u>adsoyad</u>	<u>notu</u>	<u>sinif</u>
Orçun Madran	70	4
Ahmet Kemali	80	3
Osman Musa	60	2
Hatice Kurt	90	1
Veli Faydalı	90	1
Ayşe Şahmeran	67	1
Mustafa Atlı	56	2
Zehra Kamuran	34	1
Oktay Bağcı	69	4
Demir Bülbül	24	4
Fatma Yerlisu	38	3
Hulusi Ay	86	3
Mahmut Şahin	100	4
Duygu Mutlu	90	4

Örnek tablomuzda (veritest) yer alan öğrencilerin aldığı en yüksek notu döndürmek istiyoruz.

SQL cümlecği aşağıdaki gibi olmalıdır.

```
SELECT MIN(notu) AS endusuk  
FROM veritest
```

endusuk
24





# GROUP BY ifadesinin kullanımı

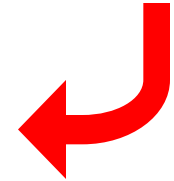
<u>adsoyad</u>	<u>notu</u>	<u>sinif</u>
Orçun Madran	70	4
Ahmet Kemali	80	3
Osman Musa	60	2
Hatice Kurt	90	1
Veli Faydalı	90	1
Ayşe Şahmeran	67	1
Mustafa Atlı	56	2
Zehra Kamuran	34	1
Oktay Bağcı	69	4
Demir Bülbül	24	4
Fatma Yerlisu	38	3
Hulusi Ay	86	3
Mahmut Şahin	100	4
Duygu Mutlu	90	4

Örnek tablomuzda (veritest) yer alan öğrencilerin not ortalamalarını sınıf bazında döndürmek istiyoruz.

SQL cümlecği aşağıdaki gibi olmalıdır.

```
SELECT sınıf, AVG(notu) AS ortalama  
FROM veritest  
GROUP BY sınıf
```

<u>sinif</u>	<u>ortalama</u>
1	70.2500
2	68.0000
3	68.0000
4	70.6000







# GROUP BY ile ORDER BY ifadesinin kullanımı

<u>adsoyad</u>	<u>notu</u>	<u>sinif</u>
Orçun Madran	70	4
Ahmet Kemali	80	3
Osman Musa	60	2
Hatice Kurt	90	1
Veli Faydalı	90	1
Ayşe Şahmeran	67	1
Mustafa Atlı	56	2
Zehra Kamuran	34	1
Oktay Bağcı	69	4
Demir Bülbül	24	4
Fatma Yerlisu	38	3
Hulusi Ay	86	3
Mahmut Şahin	100	4
Duygu Mutlu	90	4

Örnek tablomuzda (veritest) yer alan öğrencilerin not ortalamalarını sınıf bazında en yüksekten en düşüğe göre döndürmek istiyoruz.

SQL cümlecği aşağıdaki gibi olmalıdır.

```
SELECT sınıf, AVG(notu) AS ortalama
FROM veritest
GROUP BY sınıf
ORDER BY ortalama DESC
```

<u>sinif</u>	<u>ortalama</u>
4	70.6000
1	70.2500
3	68.0000
2	58.0000

