

# MATLAB'A GİRİŞ

## Contents

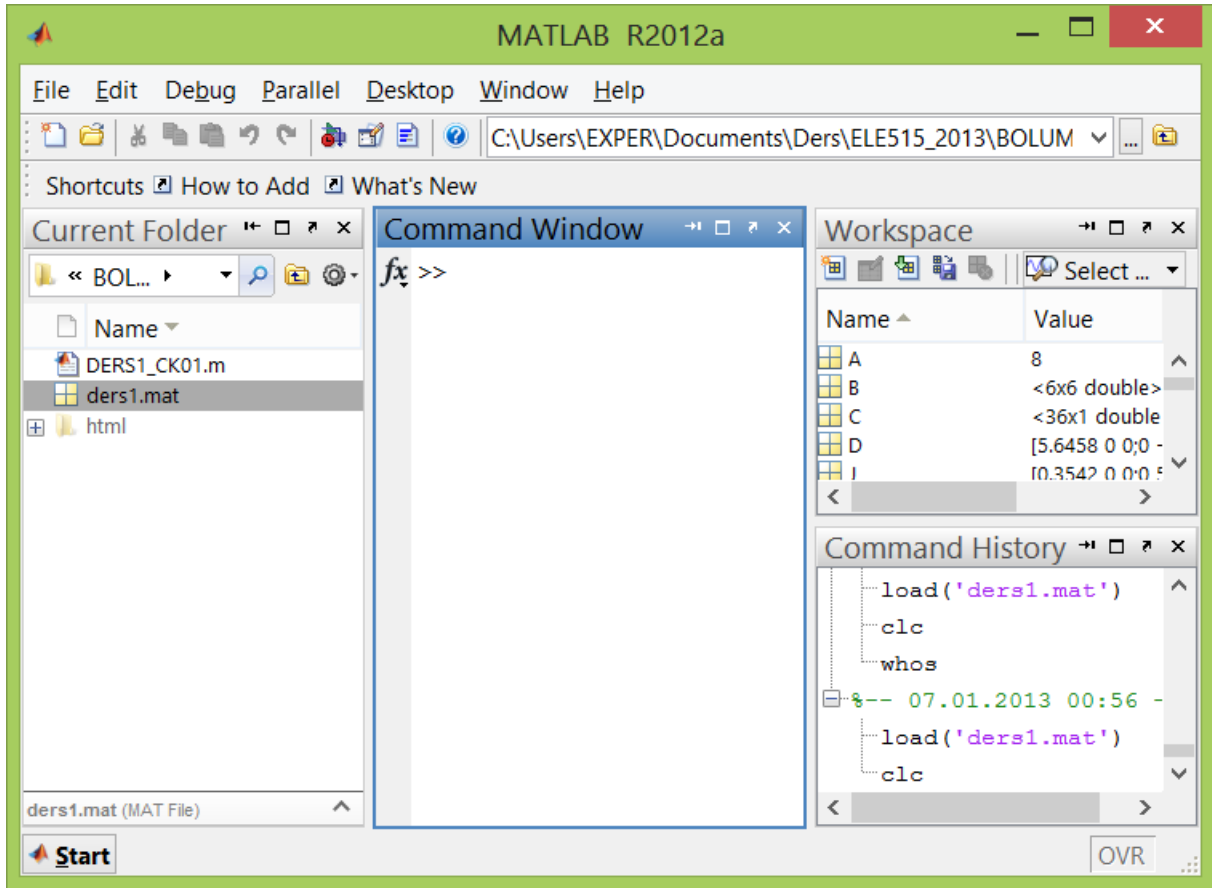
---

- [Genel bakış](#)
- [Basit hesaplamalar](#)
- [Değişken atama ve bunlarla işlemler](#)
- [Yardım komutları](#)
- [Değişkenleri görme ve silme](#)
- [Ekranı temizleme](#)
- [Matris işlemleri](#)
- [Matrisler için dört işlem, üst alma, eksponansiyel vb. işlemler](#)
- [Matrisler üzerinde eleman eleman işlem yaptırmak](#)
- [Standart fonksiyonları matrislere uygulama](#)
- [Vektörlerde minimum, maksimum vb. işlemler](#)
- [Matrislerde minimum, maksimum vb. işlemler](#)
- [Karşılaştırma ve mantık işlemleri](#)
- [Karmaşık sayılarla ilgili bazı faydalı komutlar](#)
- [Bazı faydalı özel matrisler](#)
- [Matrisler üzerinde bazı doğrusal cebir işlemleri](#)
- [Şekil çizdirmek](#)
- [Tek şekil üzerine birden fazla fonksiyon çizmek](#)
- [İki şekli aynı grafikte üst üste çizdirmek](#)
- [Şekli alt şekillere bölme \(subplot\)](#)
- [Üç boyutlu fonksiyonlar ve çizimleri](#)
- [Programlama komutları](#)
- [Oturumdaki değişkenleri kaydetme ve yeniden yükleme](#)

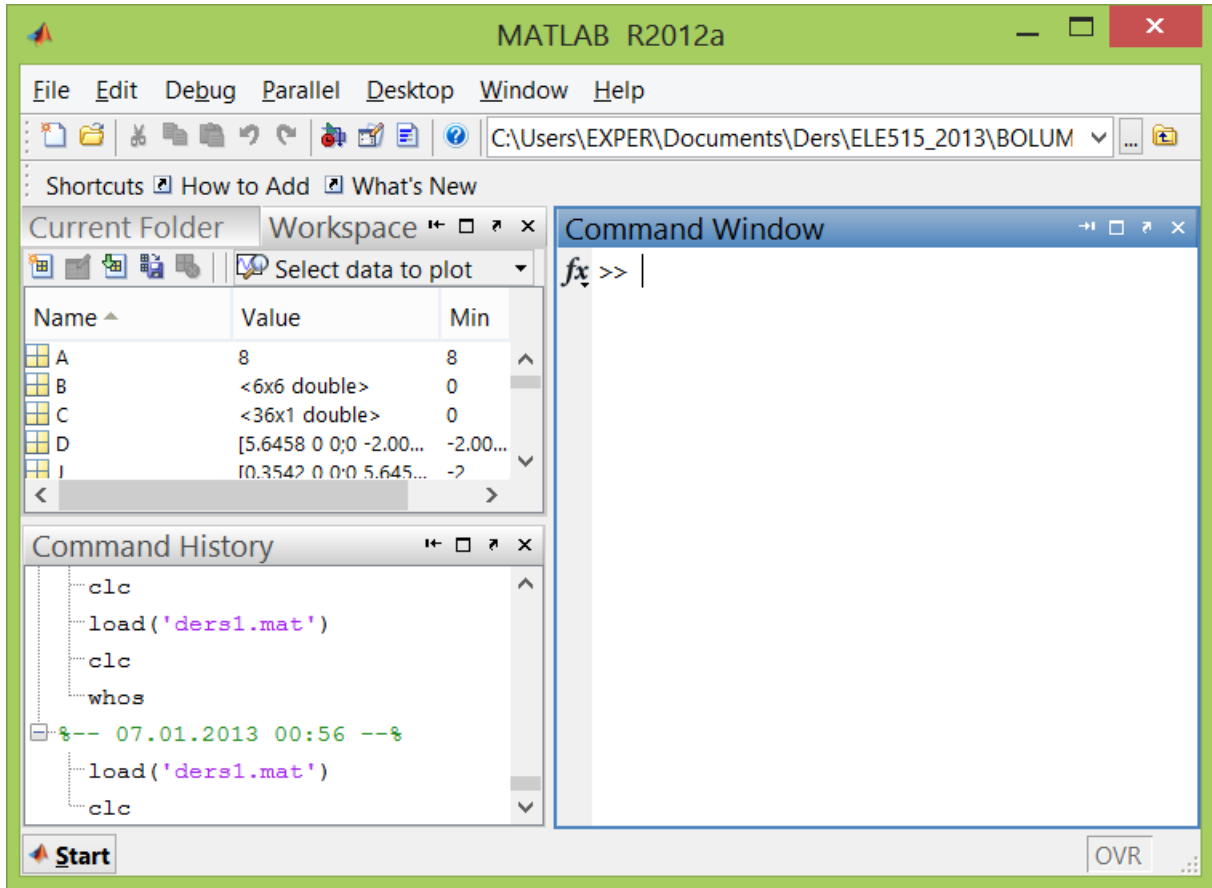
## Genel bakış

---

MATLAB ilk çalıştırıldığında gelen görüntü aşağıdaki gibidir.



Bu ekranda en önemli kısım Command Window kısmı olup, MATLAB'a komutlar buradan girilir. Current Folder yazan kısım şu andaki çalışma klasörü olup, Workspace yazan kısım şu anda tanımlı değişkenleri ve Command History kısmı son çağırılan komutları gösterir. Bu pencereleri tutup sürükleyerek ekrandaki yerleri tercihe göre değiştirilebilir, örneğin ben aşağıdaki şekilde kullanmayı tercih ediyorum.



## Basit hesaplamalar

MATLAB'da her türlü basit hesaplama kolayca yapılabilir.

```
2+3 % toplama
```

```
ans =
```

```
5
```

```
5*7 % çarpma
```

```
ans =
```

```
35
```

```
2^4 % 2 üzeri 4
```

```
ans =
```

16

```
sqrt(9) % kare kök
```

```
ans =
```

3

```
3*4/2+3 % karışık bir işlem
```

```
ans =
```

9

```
pi % pi sayısı MATLAB'da iç değişken olarak tanımlı
```

```
ans =
```

3.1416

```
sin(pi/2) % Not: Trigonometrik fonksiyonlara açılar hep RADYAN girilmeli!
```

```
ans =
```

1

```
exp(1) % e sayısı
```

```
ans =
```

2.7183

```
exp(3) % e^3
```

```
ans =
```

20.0855

**Karmaşık sayılar oluşturmak için i veya j iç değişkenlerini kullanınız.**

**5+4\*i**

ans =

5.0000 + 4.0000i

**5+4i**

ans =

5.0000 + 4.0000i

**6+7\*j**

ans =

6.0000 + 7.0000i

**6+7j**

ans =

6.0000 + 7.0000i

**sqrt(-1)**

ans =

0 + 1.0000i

```
2+sin(pi/8)*exp(3)+5.3*log(37) % karışık bir işlem
```

```
ans =
```

```
28.8243
```

## Değişken atama ve bunlarla işlemler

MATLAB'da değişkenlere değer atamak ve bunları kullanmak da kolaydır.

```
x=8 % x değişkenine sekiz değerini ver
```

```
x =
```

```
8
```

```
x=8; % bir komutun sonuna noktalı virgül konursa komut çalışır ama sonucu  
ekranda görünmez  
2*x
```

```
ans =
```

```
16
```

```
x=4
```

```
ans =
```

```
4
```

```
sqrt(x) % karekök
```

```
ans =
```

```
2.8284
```

`log(x) % e tabanında logaritma`

`ans =`

2.0794

`log10(x) % 10 tabanında logaritma`

`ans =`

0.9031

`exp(x) % e üzeri x`

`ans =`

2.9810e+03

`y=x^2 % x'in karesi`

`y =`

64

`sin(x) % sinüs x`

`ans =`

0.9894

`z=x^y-4 % x üzeri y eksi 4`

`z =`

6.2771e+57

```
w = x+y+z; % x,y ve z'nin toplamı
```

**MATLAB son işlemin sonucunu ans değişkeninde tutar**

```
ans
```

```
ans =
```

```
0.9894
```

```
ans+1
```

```
ans =
```

```
1.9894
```

```
ans+1
```

```
ans =
```

```
2.9894
```

```
ans+1
```

```
ans =
```

```
3.9894
```

**Değişken ismi seçerken dikkat edilmesi gereken bir nokta: Eğer MATLAB'ın iç değişkenlerinden birine (mesela i, j, pi gibi) atama yaparsak artık o değişkenin standart değerini yitiririz ve kullanamayız. Örneğin:**

```
pi=4; % pi değişkenine 4 verelim  
pi % artık pi 4 oldu, standart pi'yi (3.1415926...) yitirdik
```

```
pi =
```



```
sin(pi/2) % pi artık 4 olduğu için bu komut sin(2)'yi veriyor!
```

```
ans =
```

```
0.9093
```

**Not:** Eğer pi değişkenini standart değerine tekrar döndürmek istersek bunu yapmanın bir yolu var; biraz aşağıda clear komutuna değinirken bundan bahsedeceğiz.

## Yardım komutları

MATLAB'da yardım komutlarını kullanabilmek çok önemlidir. MATLAB çok büyük bir programdır ve binlerce fonksiyon barındırır. Bunların hepsini ve nasıl kullanıldıklarını akılda tutmak imkansızdır. Bu nedenle komutların nasıl kullanıldığını hatırlamak veya bir işi yaptırmak için hangi komutlar kullanılmalı gerektiğini öğrenmek için yardım dokümanlarından düzenli olarak faydalanmak gerekecektir.

```
help exp % exp komutu hakkında hızlı yardım
```

```
EXP      Exponential.
```

```
EXP(X) is the exponential of the elements of X, e to the X.
```

```
For complex Z=X+i*Y, EXP(Z) = EXP(X)*(COS(Y)+i*SIN(Y)).
```

```
See also EXPM1, LOG, LOG10, EXPM, EXPINT.
```

```
Overloaded methods:
```

```
zpk/exp
```

```
tf/exp
```

```
codistributed/exp
```

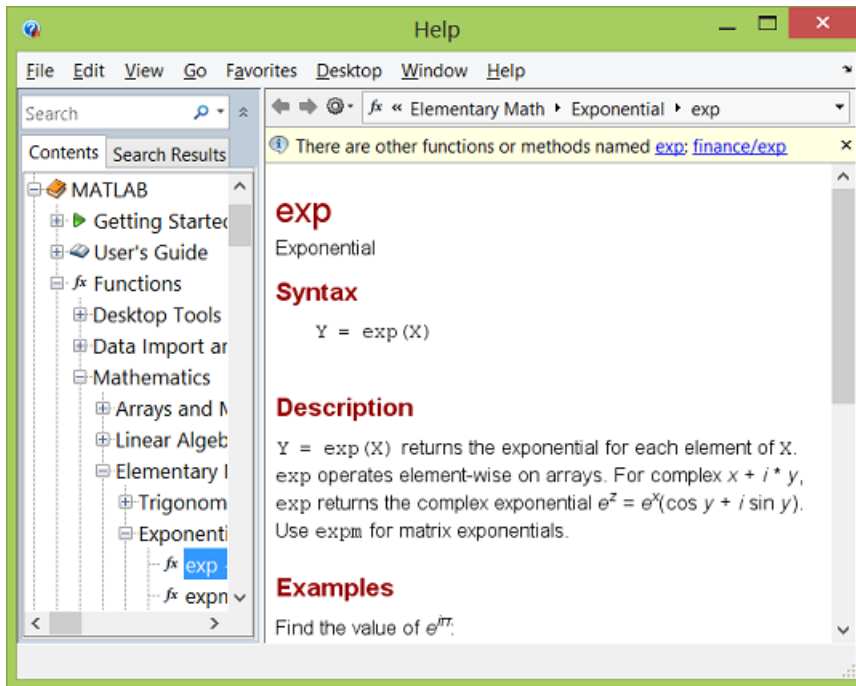
```
fints/exp
```

```
xregcovariance/exp
```

```
Reference page in Help browser
```

```
doc exp
```

```
doc exp % exp komutu hakkında Yardım Penceresinde detaylı yardım
```



```
lookfor 'inverse tangent' % bir işi yapan komut arama
```

```
atan                - Inverse tangent, result in radians.
atan2               - Four quadrant inverse tangent.
atand              - Inverse tangent, result in degrees.
cordicatan2        - CORDIC-based four quadrant inverse tangent.
```

## Değişkenleri görme ve silme

```
who % tanımlı olan değişkenlerin isimlerini gösterir
```

```
Your variables are:
```

```
ans  pi  w  x  y  z
```

```
whos % tanımlı değişkenleri detaylı bilgilerle gösterir
```

Name	Size	Bytes	Class	Attributes
ans	1x1	8	double	
pi	1x1	8	double	
w	1x1	8	double	
x	1x1	8	double	

```
y          1x1          8 double
z          1x1          8 double
```

```
clear x % x değişkenini siler, artık x'i kullanamayız
who % değişken listesinde bakarsak artık x yok çünkü silindi
```

Your variables are:

```
ans pi w y z
```

```
clear y z % y ve z değişkenlerini de silelim
who % değişken listesinde bakarsak y ve z'nin de silinmiş olduğunu görürüz
```

Your variables are:

```
ans pi w
```

Eğer bir iç değişkene bir değer atayıp o değişkeni kaybettiysek (mesela yukarıda pi'ye 4 verip, standart pi'yi kaybetmiştik) o değişkene clear uyguladığımızda bizim yaptığımız atamayı siler, böylece o değişkenin standart değerini yeniden elde etmiş oluruz:

```
pi % pi'ye dört vermiştik
```

```
pi =
```

```
4
```

```
clear pi % pi'ye bizim yaptığımız atamayı siler, standart değerine döner
pi % pi yeniden 3.1415926... oldu
```

```
ans =
```

```
3.1416
```

```
clear all % tanımlı tüm değişkenleri siler
who % değişken listesi boş, tanımlı hiç bir değişken yok
whos % değişken listesi boş, tanımlı hiç bir değişken yok
```

## Ekranı temizleme

Komut satırını temizlemek için clc komutu kullanılır.

```
clc
```

## Matris işlemleri

MATLAB aslında her değişkeni matris olarak görür. Matrisi oluşturan elemanlar [ ve ] arasında yazılır. Aynı satırdaki elemanlar , veya boşluk ile ayrılır. Bir alt sütuna geçmek için ; kullanılır.

```
vector = [1 5 -3] % bir satır vektörü
```

```
vector =
```

```
1 5 -3
```

```
vector = [1,5,-3] % aynı satır vektörü
```

```
vector =
```

```
1 5 -3
```

```
vector2 = [2; 1; 6] % bir sütun vektörü
```

```
vector2 =
```

```
2
```

```
1
```

```
6
```

```
matrix=[2 3 2^2 2*3-1;4 0 -5 1] % 2x5 bir matris
```

```
matrix =
```

```
2 3 4 5
```

```
4 0 -5 1
```

```
whos % değişkenlere bakalım
```

Name	Size	Bytes	Class	Attributes
------	------	-------	-------	------------

```
matrix      2x4      64 double
vector      1x3      24 double
vector2     3x1      24 double
```

```
clear vector % vektor deęişkenini silemim
whos % tekrar deęişkenlere bakalım
```

```
Name      Size      Bytes Class      Attributes

matrix    2x4      64 double
vector2   3x1      24 double
```

```
matrix=[matrix; 1 2 3 4] % matrisin altına satır ekle
```

```
matrix =
```

```
     2     3     4     5
     4     0    -5     1
     1     2     3     4
```

```
matrix=[matrix, [1;2;3]] % matrisin saęına satır ekle
```

```
matrix =
```

```
     2     3     4     5     1
     4     0    -5     1     2
     1     2     3     4     3
```

```
v=[1;2;3]
```

```
v =
```

```
     1
     2
```

```
matrix=[matrix,v]
```

```
matrix =
```

```
     2     3     4     5     1     1
     4     0    -5     1     2     2
     1     2     3     4     3     3
```

```
matrix(1,:) % matrisin birinci satirini seç
```

```
ans =
```

```
     2     3     4     5     1     1
```

```
matrix(:,2) % matrisin ikinci sutununu seç
```

```
ans =
```

```
     3
```

```
     0
```

```
     2
```

**matrisin birinci satirini deęiřtir**

```
matrix(1,:) = [7 7 7 7 7 7]
```

```
matrix =
```

```
     7     7     7     7     7     7
     4     0    -5     1     2     2
     1     2     3     4     3     3
```

```
matrix
```

```
matrix =
```

```
7 7 7 7 7 7
4 0 -5 1 2 2
1 2 3 4 3 3
```

```
matrix(:,2)=[-1; -1; -1] % matrisin ikinci sütununu değiştir
```

```
matrix =
```

```
7 -1 7 7 7 7
4 -1 -5 1 2 2
1 -1 3 4 3 3
```

```
matrix
```

```
matrix =
```

```
7 -1 7 7 7 7
4 -1 -5 1 2 2
1 -1 3 4 3 3
```

## Matrisler için dört işlem, üst alma, eksponansiyel vb. işlemler

```
A = [1 2 3;5 5 7] % 2x3 matris
```

```
A =
```

```
1 2 3
5 5 7
```

**B = [1 2 -1; 3 5 2; -2 -5 -1] % 3x3 matris**

B =

1	2	-1
3	5	2
-2	-5	-1

**C = [-1 0 1; -1 0 2] % 2x3 matris**

C =

-1	0	1
-1	0	2

**M = A+C % Matris toplama**

M =

0	2	4
4	5	9

**M=A\*B % Matris çarpımı**

M =

1	-3	0
6	0	-2

**M = B^-1 % Matris tersi**

M =

0.6250	0.8750	1.1250
-0.1250	-0.3750	-0.6250



-0.6250    0.1250    -0.1250

**M = B<sup>2</sup> % Matrisin karesi (ikinci kuvveti)**

M =

9    17    4  
14    21    5  
-15    -24    -7

**M = B\*B**

M =

9    17    4  
14    21    5  
-15    -24    -7

**M = B\*B\*B % Matrisin küpü (üçüncü kuvveti)**

M =

52    83    21  
67    108    23  
-73    -115    -26

**M = B<sup>3</sup>**

M =

52    83    21  
67    108    23  
-73    -115    -26

```
M = expm(B) % Matris exponansiyeli e^B (Dikkat: exp ile karıştırmayın!)
```

```
M =
```

```
62.1908 100.0514 21.5535
84.2242 136.3143 31.3354
-88.5987 -144.1913 -32.1342
```

```
M = A' % Matrisin transpozesi (devriği)
```

```
M =
```

```
1 5
2 5
3 7
```

**Not: Matris işlemi tanımlı değilse MATLAB hata verir. Örneğin:  $M = B \cdot A$  % Boyutlar uymuyor  
 $M = A + B$  % Boyutlar uymuyor**

$M = A^3$  % A kare matris değil, kuvveti alınamaz

$M = C^{-1}$  % C kare matris değil, tersi alınamaz

## Matrisler üzerinde eleman eleman işlem yaptırmak

MATLAB Aksi belirtilmedikçe her zaman matris işlemi yapacaktır. Eleman eleman işlem yapmak için işlemde önce nokta koyun.

```
M = A.^2 % A'nın her elemanının karesi
```

```
M =
```

```
1 4 9
25 25 49
```

```
M = B.^2 % B'nin her elemanının karesi
```

M =

```
1     4     1
9     25    4
4     25    1
```

```
M = C./A % C'nin her elemanının A'daki karşılık gelen elemana bölümü
```

M =

```
-1.0000     0     0.3333
-0.2000     0     0.2857
```

## Standart fonksiyonları matrislere uygulama

```
% Standart fonksiyonlar matrislere, vektörlere de uygulanabilir. Bu durumda
% tek tek her elemana o fonksiyonu uygulayacaktır.
```

```
sqrt(B) % Elemanların tek tek karekökü
```

```
sqrt([1:9]) % Elemanların tek tek karekökü
```

```
sin([pi/4 pi/2 pi]) % Elemanların tek tek sinüsü
```

ans =

```
1.0000     1.4142     0 + 1.0000i
1.7321     2.2361     1.4142
0 + 1.4142i     0 + 2.2361i     0 + 1.0000i
```

ans =

Columns 1 through 7

```
1.0000     1.4142     1.7321     2.0000     2.2361     2.4495     2.6458
```

Columns 8 through 9

2.8284 3.0000

ans =

0.7071 1.0000 0.0000

**Bazı özel durumlar: exp komutu eleman eleman eksponansiyel alır. expm % ise matris eksponansiyeli alır. expm sadece kare matrisler için tanımlıdır. Örneğin kare matris B için:**

```
M = expm(B) % Matris eksponansiyeli
```

M =

62.1908 100.0514 21.5535  
84.2242 136.3143 31.3354  
-88.5987 -144.1913 -32.1342

```
M = exp(B) % Her elemanın tek tek eksponansiyeli - üstteki sonuçtan farklı!
```

M =

2.7183 7.3891 0.3679  
20.0855 148.4132 7.3891  
0.1353 0.0067 0.3679

**Kare olmayan matris A için eleman eleman eksponansiyel alınabilir:**

```
M = exp(A)
```

M =

1.0e+03 \*

```
0.0027    0.0074    0.0201
0.1484    0.1484    1.0966
```

Fakat kare olmayan A matrisinin eksponansiyeli alınamaz, aşağıdaki komut çalıştırılırsa hata verecektir:  $M = \text{expm}(A)$

## Vektörlerde minimum, maksimum vb. işlemler

İki vektör oluşturalım

```
a=[-1 5 0 -2]
```

```
b=[0 7 -9 3]
```

a =

```
-1    5    0   -2
```

b =

```
0    7   -9    3
```

```
min(a) % Vektörün minimumu
```

```
ans =
```

```
-2
```

```
max(a) % Vektörün maksimumu
```

```
ans =
```

```
5
```

```
length(a) % Vektörünü uzunluğu
```

```
ans =
```

```
4
```

```
sort(a) % Vektörü sıralama
```

```
ans =
```

```
-2 -1 0 5
```

```
mean(a) % Vektörün ortalama değeri
```

```
ans =
```

```
0.5000
```

```
median(a) % Vektörün medyan değeri
```

```
ans =
```

```
-0.5000
```

```
min(a,b) % a ve b'yi eleman eleman karşılaştır, minimum değerleri seç
```

```
ans =
```

```
-1 5 -9 -2
```

**end komutu kullanılarak vektörün son elemanına ulaşılabilir.**

```
a(end) % a'nın son elemanını göster
```

```
ans =
```

```
-2
```

```
a(end) = 77 % a'nın son elemanını 77 yap
```

```
a =
```

```
-1    5    0    77
```

```
a(end) = [] % a'nın son elemanını sil
```

```
a =
```

```
-1    5    0
```

```
a(end+1:end+3)=[-1 -2 -3] % a'nın sonuna [-1 -2 -3] sayılarını ekle
```

```
a =
```

```
-1    5    0    -1    -2    -3
```

```
a(end+1)=100 % a'nın sonuna 100 sayısını ekle
```

```
a =
```

```
-1    5    0    -1    -2    -3    100
```

## Matrislerde minimum, maksimum vb. işlemler

İki matris oluşturalım.

```
A=[8 7 9 11; 9 -1 -6 9; 2 7 -4 0]
```

```
B=[3 6 9 12; -3 -6 -9 -12; 0 0 0 0]
```

```
A =
```

```
8     7     9    11
9    -1    -6     9
```

```
2    7    -4    0
```

B =

```
3    6    9    12
-3   -6   -9   -12
0    0    0    0
```

```
min(A) % Her sütunun minimumu
```

ans =

```
2    -1    -6    0
```

```
min(A(:)) % Tüm matrisini minimumu
```

ans =

```
-6
```

```
min(B) % Her sütunun minimumu
```

ans =

```
-3    -6    -9    -12
```

```
min(B(:)) % Tüm matrisini minimumu
```

ans =

```
-12
```



```
max(A) % Her sütunun maksimumu
```

```
ans =
```

```
9 7 9 11
```

```
max(A(:)) % Tüm matrisin maksimumu
```

```
ans =
```

```
11
```

```
size(A) % A'nın boyutları
```

```
ans =
```

```
3 4
```

```
length(A) % A'nın en uzun boyutu
```

```
ans =
```

```
4
```

```
sort(A) % Her sütunu sırala
```

```
ans =
```

```
2 -1 -6 0  
8 7 -4 9  
9 7 9 11
```

```
mean(A) % Her sütunun ortalaması
```

```
ans =
```

6.3333 4.3333 -0.3333 6.6667

```
median(A) % Her sütunun medyanı
```

```
ans =
```

8 7 -4 9

```
min(A,B) % A ve B'yi eleman eleman karşılaştır, minimum değerleri seç
```

```
ans =
```

3 6 9 11  
-3 -6 -9 -12  
0 0 -4 0

## Karşılaştırma ve mantık işlemleri

Matrislerle (ve vektörlerde) mantık işlemleri yapabiliriz. İşlemler eleman eleman yapılır. Bu tür işlemler için 0 (false) dışındaki bütün elemanlar 1 (true) kabul edilir.

İki matris oluşturalım.

```
a=[-1 5 0 7 -5 4]
```

```
b=[ 0 7 -9 2 4 6]
```

```
a =
```

-1 5 0 7 -5 4

```
b =
```

0 7 -9 2 4 6

```
c = a&b % a ve b
```

```
c =
```

```
0 1 0 1 1 1
```

```
c = a|b % a veya b
```

```
c =
```

```
1 1 1 1 1 1
```

```
c = xor(a,b) % a xor b
```

```
c =
```

```
1 0 1 0 0 0
```

**Matrislerle karşılaştırma işlemleri de yapabiliriz. İşlemler eleman eleman yapılır.**

```
ind = a>b
```

```
ind =
```

```
0 0 1 1 0 0
```

```
ind = a<=b
```

```
ind =
```

```
1 1 0 0 1 1
```

```
ind = a==b
```

```
ind =
```

```
0 0 0 0 0 0
```

Bir matrisi bir skalerle karşılaştırdığımızda tüm elemanlar o skalerle karşılaştırılır

```
ind = a > 1 % a'nın tüm elemanlarını birle kıyaslar
```

```
ind =
```

```
0 1 0 1 0 1
```

Karşılaştırma sonuçlarını indis olarak kullanabiliriz. Örneğin a'nın ikiden büyük olan elemanlarını seçmek için aşağıdaki komutları kullanabiliriz.

```
ind = a > 2
```

```
ind =
```

```
0 1 0 1 0 1
```

```
a(ind)
```

```
ans =
```

```
5 7 4
```

## Karmaşık sayılarla ilgili bazı faydalı komutlar

Karmaşık sayılardan bir vektör oluşturalım

```
karmasik = [1+j, j, 2*j]
```

```
karmasik =
```

```
1.0000 + 1.0000i
```

```
0 + 1.0000i
```

```
0 + 2.0000i
```

```
genlik = abs(karmasik) % Genlik hesaplama
```

```
genlik =
```

```
1.4142    1.0000    2.0000
```

```
faz = angle(karmasik) % Faz hesaplama. NOT: Açılar hep RADYAN çıkar!
```

```
faz =
```

```
0.7854    1.5708    1.5708
```

```
fazDerece = faz*180/pi % Radyanı dereceye çevirmek istersek böyle yaparız.
```

```
fazDerece =
```

```
45    90    90
```

## Bazı faydalı özel matrisler

MATLAB'da birim matris, sıfır matris vb. özel matrisler yaratmak için bazı özel komutlar vardır. Tek argümanla çağırılırlarsa kare matris oluştururlar. İki argümanla % çağırıldıklarında o boyutlarda matris oluştururlar.

```
A=zeros(3) % 3x3 sıfırlardan oluşan matris
```

```
A =
```

```
0    0    0
0    0    0
0    0    0
```

```
A=zeros(3,2) %3x2 sıfırlardan oluşan matris
```

```
A =
```

```
0    0
0    0
```

0 0

```
A=ones(4) % 4x4 birlerden oluşan matris
```

A =

```
1 1 1 1
1 1 1 1
1 1 1 1
1 1 1 1
```

```
A=eye(3) % 3x3 birim matris
```

A =

```
1 0 0
0 1 0
0 0 1
```

```
B= repmat(A,2,2) % B matrisini blok olarak 2'ye 2 defa tekrarla
```

B =

```
1 0 0 1 0 0
0 1 0 0 1 0
0 0 1 0 0 1
1 0 0 1 0 0
0 1 0 0 1 0
0 0 1 0 0 1
```

```
C=B(:) % Matrisin tüm elemanlarını tek bir sütuna diz
```

C =

1  
0  
0  
1  
0  
0  
0  
1  
0  
0  
0  
1  
0  
0  
0  
1  
0  
0  
1  
1  
0  
0  
1  
0  
0  
0  
1  
0  
0  
1  
0  
0  
1  
0

0  
0  
1  
0  
0  
1

```
D=rand(3) % 3x3 rastgele matris
```

D =

```
0.8147    0.9134    0.2785  
0.9058    0.6324    0.5469  
0.1270    0.0975    0.9575
```

```
D=rand(size(B)) % B'nin boyutunda, rastgele matris
```

D =

```
0.9649    0.1419    0.0357    0.3922    0.0462    0.0344  
0.1576    0.4218    0.8491    0.6555    0.0971    0.4387  
0.9706    0.9157    0.9340    0.1712    0.8235    0.3816  
0.9572    0.7922    0.6787    0.7060    0.6948    0.7655  
0.4854    0.9595    0.7577    0.0318    0.3171    0.7952  
0.8003    0.6557    0.7431    0.2769    0.9502    0.1869
```

## Matrisler üzerinde bazı doğrusal cebir işlemleri

Bir matris oluşturalım

```
A=[0 1 2; 3 4 3;2 1 0]
```

A =



```
0    1    2
3    4    3
2    1    0
```

```
diag(A) % Matrisin köşegeni
```

```
ans =
```

```
0
4
0
```

```
inv(A) % Matrisin tersi
```

```
ans =
```

```
0.7500    -0.5000    1.2500
-1.5000    1.0000   -1.5000
1.2500    -0.5000    0.7500
```

```
A^-1 % Matrisin tersi
```

```
ans =
```

```
0.7500    -0.5000    1.2500
-1.5000    1.0000   -1.5000
1.2500    -0.5000    0.7500
```

```
A' % Matrisin transpozesi (devriği)
```

```
ans =
```

```
0    3    2
1    4    1
```

2 3 0

```
[V,D]=eig(A) % Matrisin özdeğerleri ve özvektörleri
```

V =

```
-0.2557 -0.7071 0.4608
-0.9323 -0.0000 -0.7584
-0.2557 0.7071 0.4608
```

D =

```
5.6458 0 0
0 -2.0000 0
0 0 0.3542
```

```
det(A) % Matrisin determinantı
```

ans =

-4.0000

```
rank(A) % Matrisin rankı
```

ans =

3

```
rref(A) % Matrisin eşelon biçimi
```

ans =

1 0 0

```
0    1    0
0    0    1
```

```
[V,J] = jordan(A) % Matrisin Jordan biçimi
```

```
V =
```

```
1.0000    1.0000   -1.0000
-1.6458    3.6458         0
1.0000    1.0000    1.0000
```

```
J =
```

```
0.3542         0         0
         0    5.6458         0
         0         0   -2.0000
```

## Şekil çizdirmek

MATLAB'da kolaylıkla güzel grafikler oluşturulabilir.

```
t = 0:0.1:10 % 0'dan 5'e birer birer artan zaman vektörü
```

```
t =
```

```
Columns 1 through 7
```

```
0    0.1000    0.2000    0.3000    0.4000    0.5000    0.6000
```

```
Columns 8 through 14
```

```
0.7000    0.8000    0.9000    1.0000    1.1000    1.2000    1.3000
```

Columns 15 through 21

1.4000 1.5000 1.6000 1.7000 1.8000 1.9000 2.0000

Columns 22 through 28

2.1000 2.2000 2.3000 2.4000 2.5000 2.6000 2.7000

Columns 29 through 35

2.8000 2.9000 3.0000 3.1000 3.2000 3.3000 3.4000

Columns 36 through 42

3.5000 3.6000 3.7000 3.8000 3.9000 4.0000 4.1000

Columns 43 through 49

4.2000 4.3000 4.4000 4.5000 4.6000 4.7000 4.8000

Columns 50 through 56

4.9000 5.0000 5.1000 5.2000 5.3000 5.4000 5.5000

Columns 57 through 63

5.6000 5.7000 5.8000 5.9000 6.0000 6.1000 6.2000

Columns 64 through 70

6.3000 6.4000 6.5000 6.6000 6.7000 6.8000 6.9000

Columns 71 through 77

7.0000 7.1000 7.2000 7.3000 7.4000 7.5000 7.6000

Columns 78 through 84

7.7000 7.8000 7.9000 8.0000 8.1000 8.2000 8.3000

Columns 85 through 91

8.4000 8.5000 8.6000 8.7000 8.8000 8.9000 9.0000

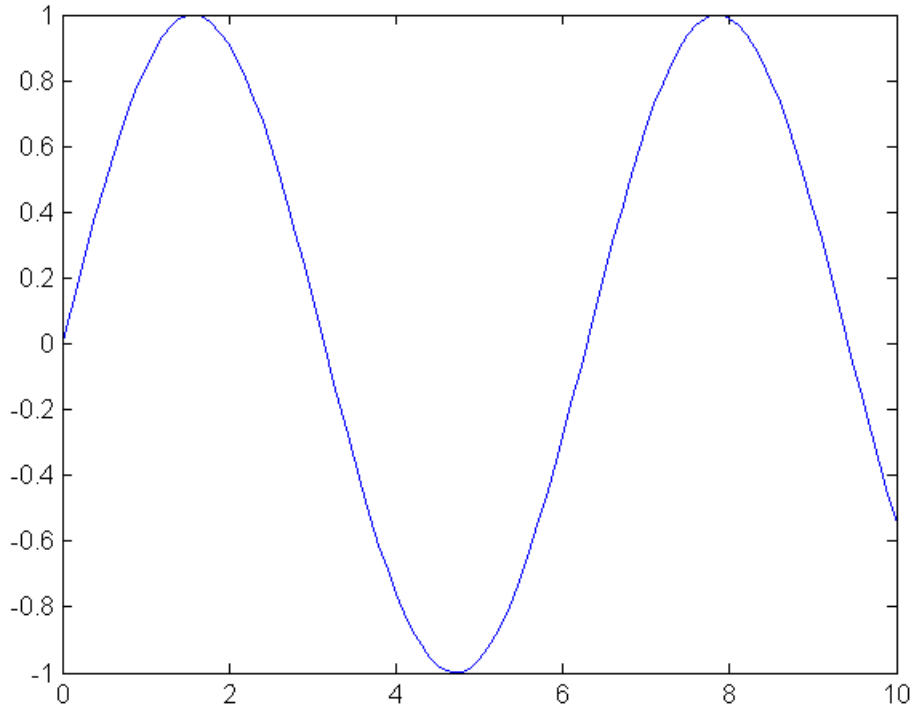
Columns 92 through 98

9.1000 9.2000 9.3000 9.4000 9.5000 9.6000 9.7000

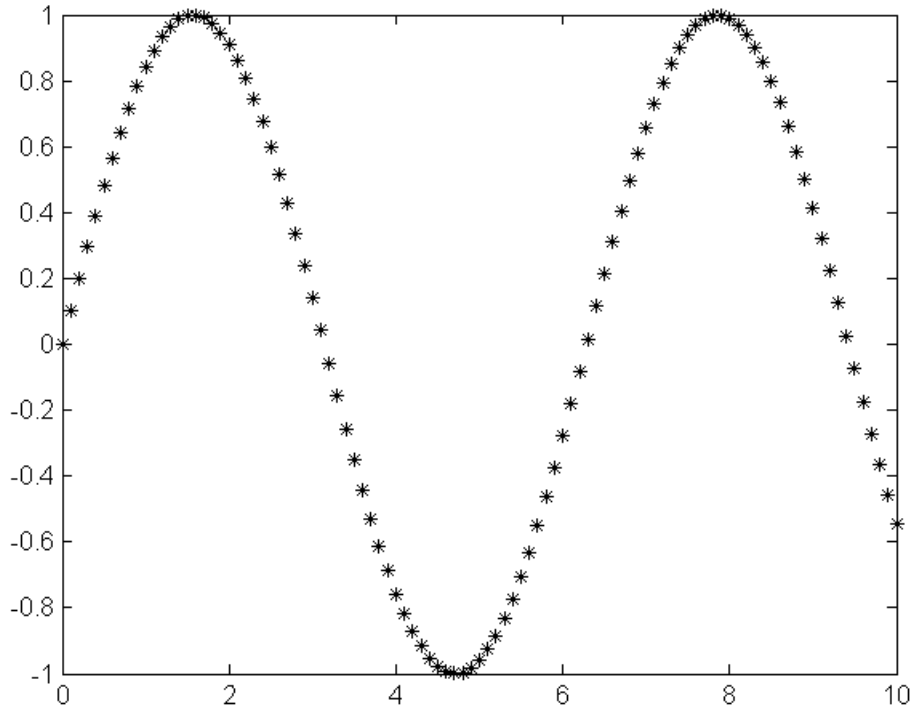
Columns 99 through 101

9.8000 9.9000 10.0000

```
s = sin(t); % t'nin sinüsünü al
plot(t,s); % s'yi t'ye karşı çizdir
```



```
plot(t,s, 'k*'); % Renk ve çizdirme biçimleri değiştirilebilir. (bkz. doc  
plot)
```



```
close; % Açık olan şekli kapat
```

**Tek şekil üzerine birden fazla fonksiyon çizmek**

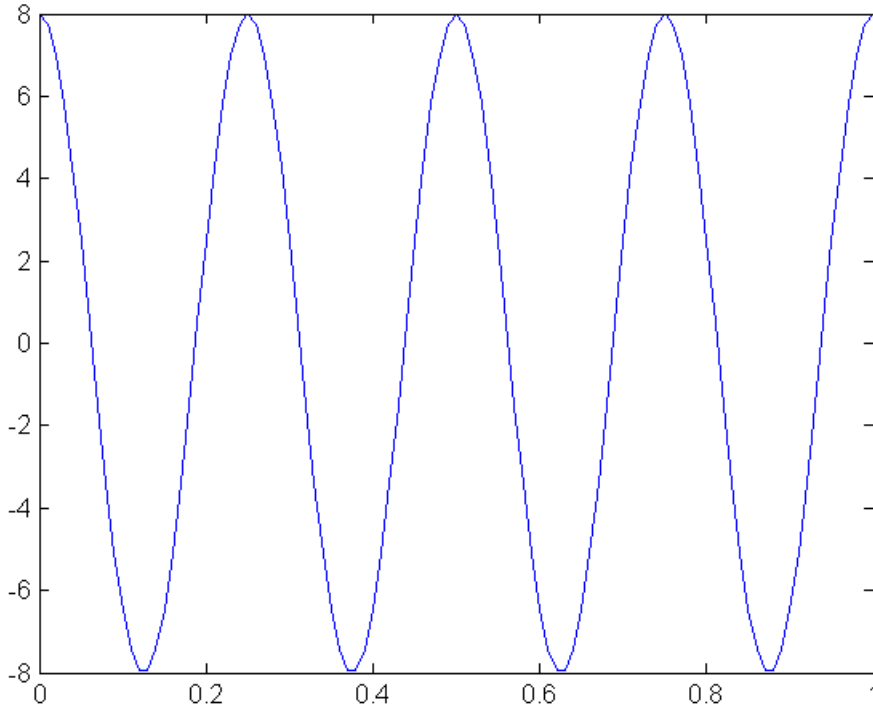
```
t = 0:0.01:1; % 0'dan 1'e kadar 0.01'lik adımlarla artan zaman vektörü
A = 8;

f_1 = 2;

s_1 = A*sin(2*pi*f_1*t); % Genliği 8, frekansı 2 Hz olan sinüs
f_2 = 4;
s_2 = A*cos(2*pi*f_2*t); % Genliği 8, frekansı 4 Hz olan kosinüs
```

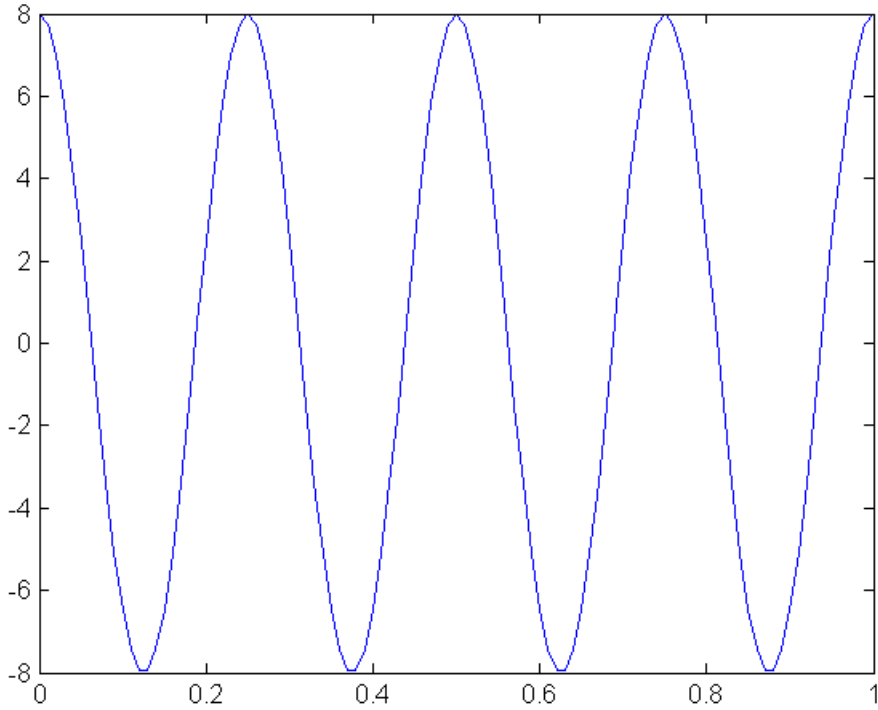
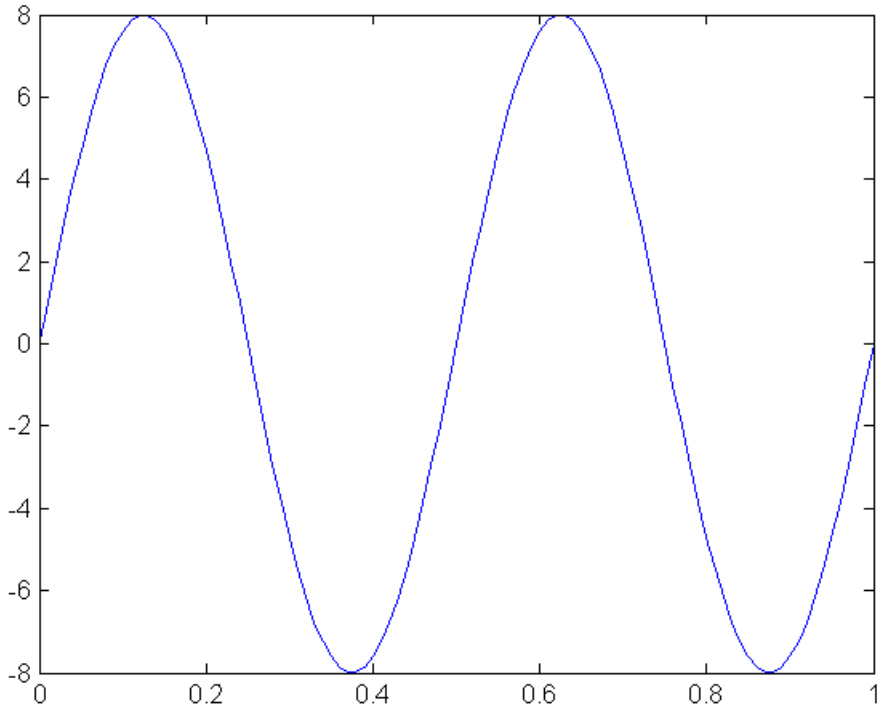
**Normalde çizim komutları açık olan şekli silip üzerine çizer mesela**

```
plot(t,s_1); % s_1'i çizer
plot(t,s_2); % Açık şekli silip üzerine s_2'yi çizer
```



**Eğer açık şeklin üzerine değil yeni bir şekle çizmek istersek**

```
plot(t,s_1); % s_1'i çizer
figure; % Yeni bir şekil açar
plot(t,s_2); % s_2'yi yeni şekle çizer
```

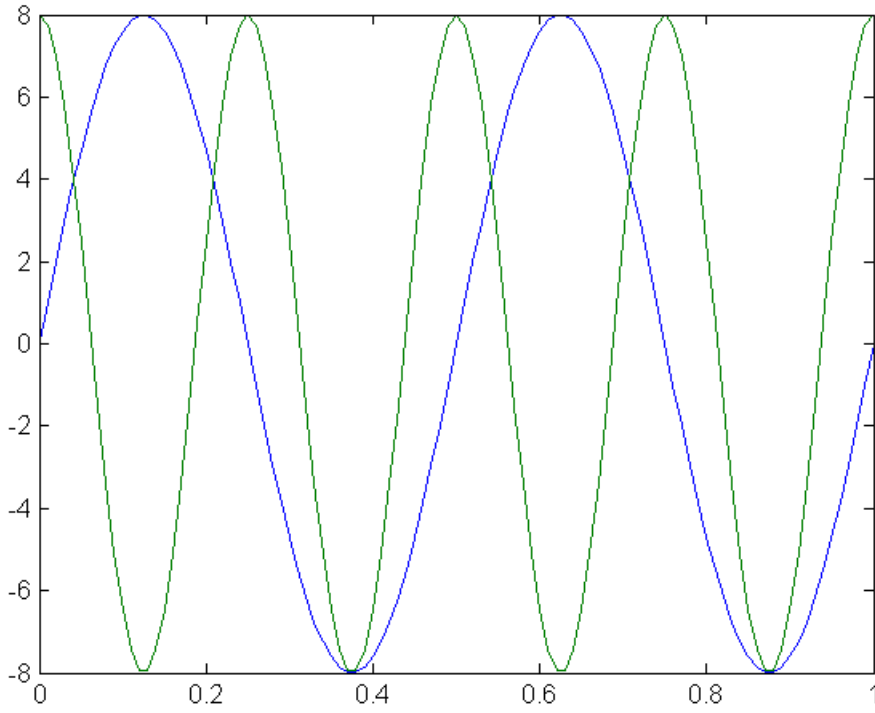


```
close all; % Açık olan tüm şekilleri kapat
```

## İki şekli aynı grafikte üst üste çizdirmek

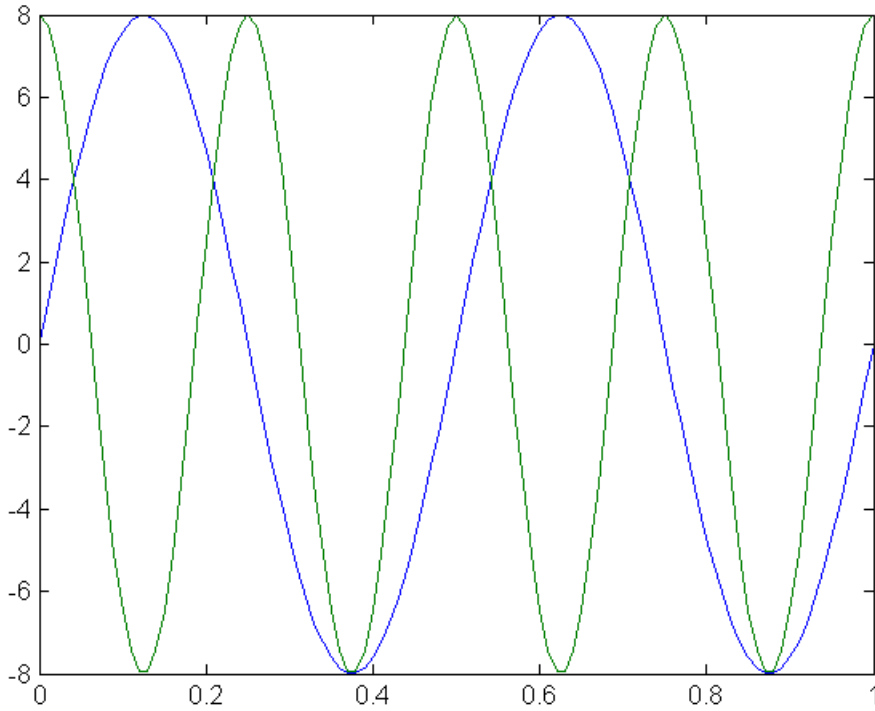
```
plot(t,s_1,t,s_2); % s_1 ve s_2'yi t_ye karşı aynı grafikte çizdir
```





### İki şekli aynı grafikte çizdirmenin alternatif yolu

```
plot(t,s_1); % Birinci şekli çizdir  
hold all; % Şekli tut  
plot(t,s_2); % İkinci şekli çizdir  
hold off; % Şekli bırak
```

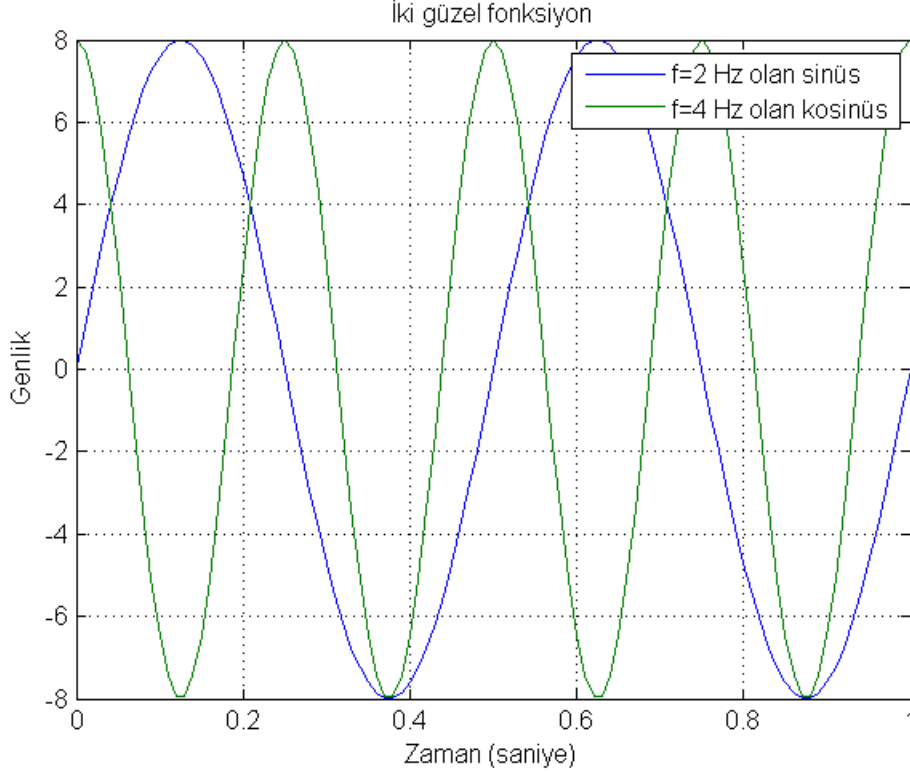


### Çizdiğimiz şekillere başlık, etiket vs. koyabiliriz

```

title('İki güzel fonksiyon'); % Şekle başlık koy
ylabel('Genlik'); % y eksenine etiket ver
xlabel('Zaman (saniye)'); % x etiket ver
legend('f=2 Hz olan sinüs','f=4 Hz olan kosinüs') % lejant koy
grid on; % Şekli karelere böl

```



Yukarıdaki ayarları ve daha fazlasını bir arayüzden de yapabiliriz. Bunun için `plottools` komutunu kullanabiliriz veya şeklin üstündeki menüden ilgili düğmeye basarız.

## Şekli alt şekillere bölme (subplot)

Aynı şekli bir kaç parçaya bölerek, her bölmeğe değişik grafikler de çizdirebiliriz.

```

subplot(2,2,1); % Şekli 2x2'lik 4 alt şekil olarak düşün, 1.yi seç
plot(t,s_1); % s_1'i çiz
title('2Hz sinüs dalgasi'); % Başlık
ylabel('Genlik')
xlabel('Zaman');

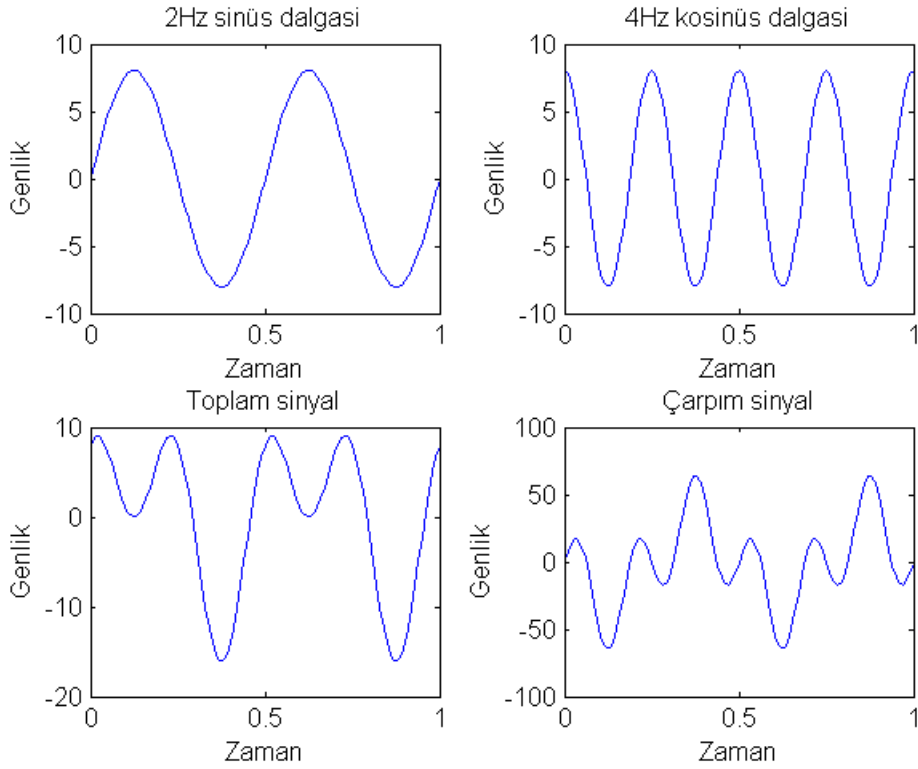
subplot(2,2,2); % Şekli 2x2'lik 4 alt şekil olarak düşün, 2.yi seç
plot(t,s_2); % s_2'yi çiz
title('4Hz kosinüs dalgasi'); % Başlık
ylabel('Genlik'); % y eksenini isimlendir
xlabel('Zaman'); % x eksenini isimlendir

subplot(2,2,3); % Şekli 2x2'lik 4 alt şekil olarak düşün, 3.yü seç
plot(t,s_1+s_2); % s_1+s_2'yi çiz
title('Toplam sinyal'); % Başlık
ylabel('Genlik'); % y eksenini isimlendir
xlabel('Zaman'); % x eksenini isimlendir

subplot(2,2,4); % Şekli 2x2'lik 4 alt şekil olarak düşün, 4.yü seç
plot(t,s_1.*s_2); % s_1*s_2'yi çiz
title('Çarpım sinyal'); % Başlık
ylabel('Genlik'); % y eksenini isimlendir

```

```
xlabel('Zaman'); % x eksenini isimlendir
```



## Üç boyutlu fonksiyonlar ve çizimleri

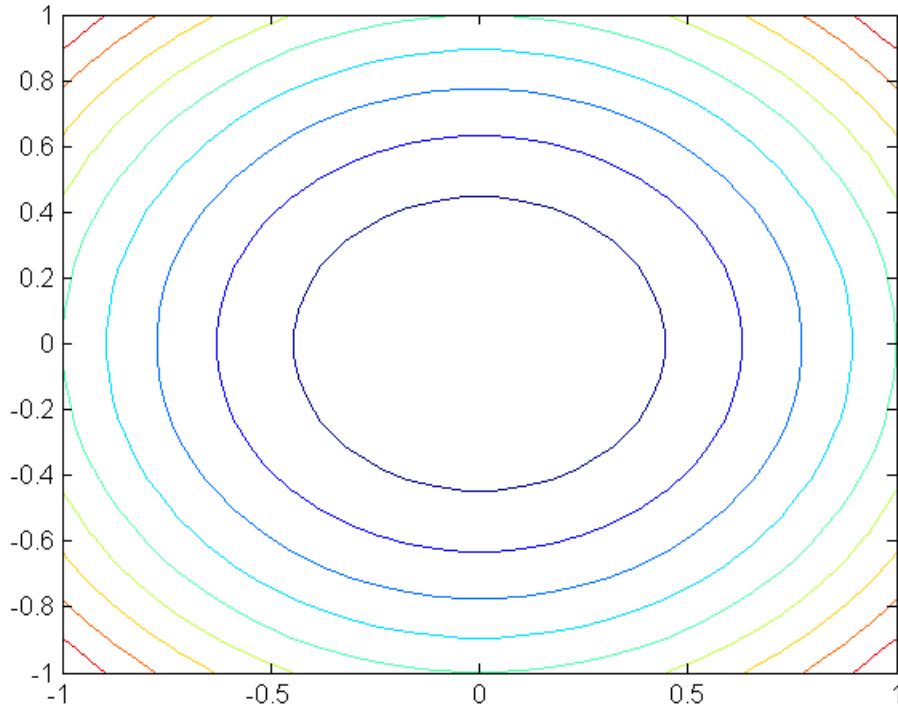
```
x = linspace(-1,1,30); %-1'den 1'e eşit aralıklı 30 elemanlı vektör oluştur  
y = linspace(-1,1,30); %-1'den 1'e eşit aralıklı 30 elemanlı vektör oluştur
```

```
[xg,yg] = meshgrid(x,y); % 3B çizim için 30x30'luk bir örgü oluştur
```

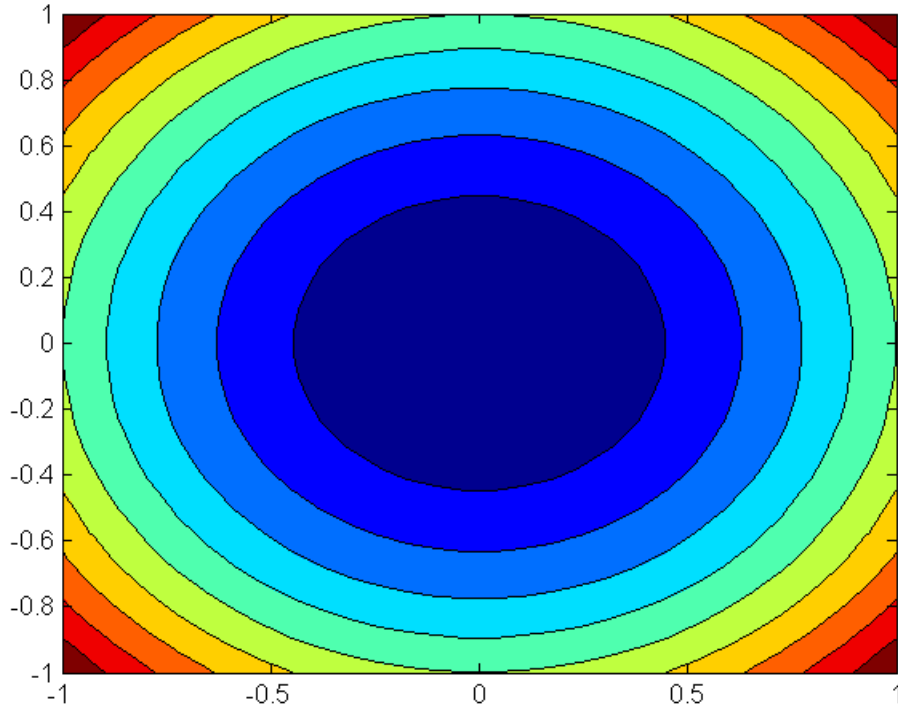
```
zg = xg.^2 + yg.^2; % Çizdirilecek fonksiyonu örgü üzerinde hesapla
```

```
figure; % Yeni bir şekil aç
```

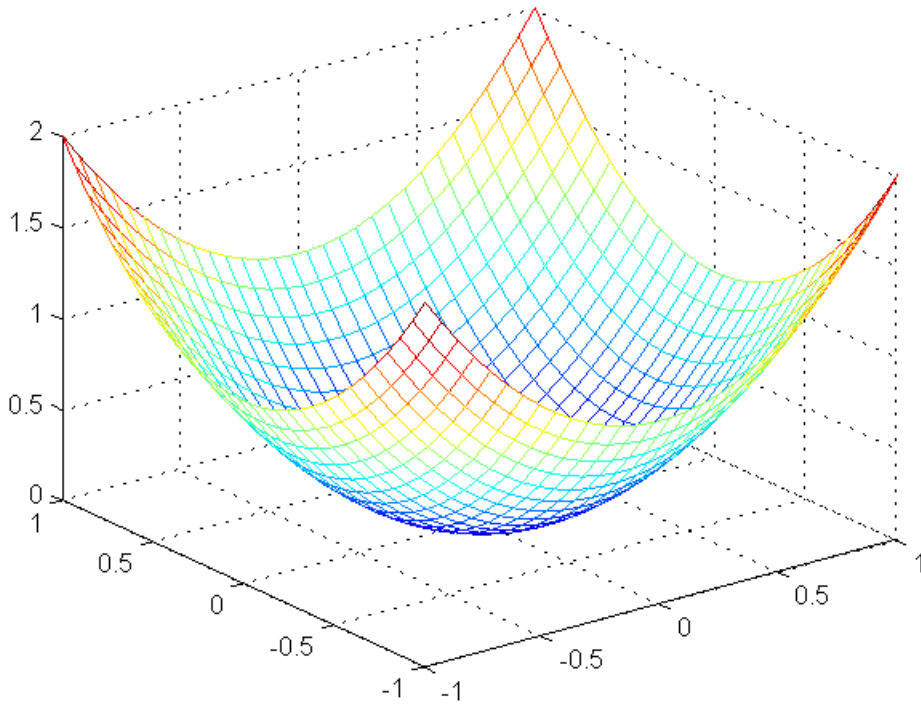
```
contour(xg,yg,zg); % Fonksiyonu eşyüselte haritası formatında çiz
```



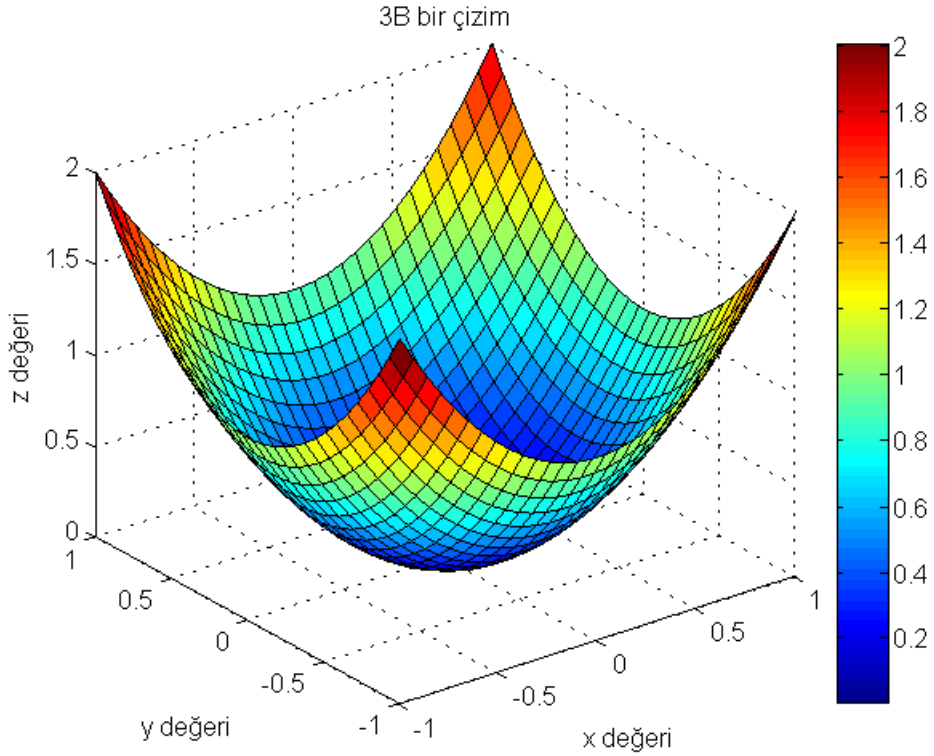
```
figure; % Yeni bir şekil aç  
contourf(xg,yg,zg); % Fonksiyonu içi dolu eşyüseliti haritası formatında  
çiz
```



```
figure; % Yeni bir şekil aç  
mesh(xg,yg,zg); % Fonksiyonu içi çiz
```



```
figure; % Yeni bir şekil aç
surf(xg,yg,zg); % Fonksiyonu içi dolu olarak çiz
colorbar; % Şekle, renklerin temsil ettiği değerleri gösteren çubuk ekle
title('3B bir çizim'); % Başlık koy
xlabel('x değeri'); % x eksenine etiket koy
ylabel('y değeri'); % y eksenine etiket koy
zlabel('z değeri'); % z eksenine etiket koy
```



# Programlama komutları

if, else, elseif ile komutları belli şartlara bağlı olarak çalıştırılması sağlanabilir.

```
x = 7;  
  
if x>0  
    disp('x pozitiftir.') % Eğer x>0 ise bunu yapar  
elseif x==0  
    disp('x sıfırıdır.') % Eğer x>0 değilse ama x==0 ise bunu yapar  
else  
    disp('x negatiftir.') % Eğer x>0 değilse ve x==0 değilse bunu yapar  
end
```

x pozitiftir.

for komutu ile belli sayıda çalışacak bir döngü yaratılabilir.

```
for i=1:10  
    i*2  
end
```

ans =

2

ans =

4

ans =

6

ans =

8

ans =

10

ans =

12

ans =

14

ans =

16

ans =

18

ans =

20

while komutu ile belli bir koşul sağlandığı sürece çalışacak bir döngü yaratılabilir

```
toplam=0;

while toplam < 1000
    toplam = toplam + 100; % toplam 1000'den küçük olduğu sürece bunu yapar
end
toplam

toplam =
```

1000

NOT: MATLAB'da döngüler yavaş çalışır. Mümkün olduğunca büyük döngüler kullanmamaya çalışın.

## Oturumdaki değişkenleri kaydetme ve yeniden yükleme

```
whos % Şu anda tanımlı değişkenler
```

Name	Size	Bytes	Class	Attributes
A	1x1	8	double	
B	6x6	288	double	
C	36x1	288	double	
D	3x3	72	double	
J	3x3	72	double	
M	2x3	48	double	
V	3x3	72	double	
a	1x6	48	double	
ans	1x1	8	double	
b	1x6	48	double	
c	1x6	6	logical	
f_1	1x1	8	double	
f_2	1x1	8	double	
faz	1x3	24	double	
fazDerece	1x3	24	double	
genlik	1x3	24	double	



i	1x1	8	double	
ind	1x6	6	logical	
karmasik	1x3	48	double	complex
matrix	3x6	144	double	
s	1x101	808	double	
s_1	1x101	808	double	
s_2	1x101	808	double	
t	1x101	808	double	
toplam	1x1	8	double	
v	3x1	24	double	
vector2	3x1	24	double	
x	1x1	8	double	
xg	30x30	7200	double	
y	1x30	240	double	
yg	30x30	7200	double	
zg	30x30	7200	double	

```

save ders1 % Bütün deęişkenleri ders1.mat dosyasına kaydedelim
clear all % Tüm deęişkenleri temizledik
whos % Bütün deęişkenlerin silindięini görelim.
load ders1 % Dosyadan deęişkenleri tekrar yükleyelim
whos % Deęişkenler tekrar yüklendięini görelim.

```

Name	Size	Bytes	Class	Attributes
A	1x1	8	double	
B	6x6	288	double	
C	36x1	288	double	
D	3x3	72	double	
J	3x3	72	double	
M	2x3	48	double	
V	3x3	72	double	
a	1x6	48	double	

ans	1x1	8	double	
b	1x6	48	double	
c	1x6	6	logical	
f_1	1x1	8	double	
f_2	1x1	8	double	
faz	1x3	24	double	
fazDerece	1x3	24	double	
genlik	1x3	24	double	
i	1x1	8	double	
ind	1x6	6	logical	
karmasik	1x3	48	double	complex
matrix	3x6	144	double	
s	1x101	808	double	
s_1	1x101	808	double	
s_2	1x101	808	double	
t	1x101	808	double	
toplam	1x1	8	double	
v	3x1	24	double	
vector2	3x1	24	double	
x	1x1	8	double	
xg	30x30	7200	double	
y	1x30	240	double	
yg	30x30	7200	double	
zg	30x30	7200	double	