

MATLAB Live Editor ile Grafik İşlemleri Dökümantasyonu Hazırlama

İçindekiler

MATLAB Grafik İşlemleri.....	2
Birinci Yöntem.....	3
İkinci Yöntem.....	4
Üçüncü Yöntem.....	5
Dördüncü Yöntem.....	9
Bütün Kodlar.....	10

MATLAB Grafik İşlemleri

Bu bölümde 2 boyutlu grafik çizdirmek için gereken değişkenleri belirleyelim. Birbirinden farklı dört grafik üzerinden örnekler verelim. Grafik çizdirmek için **"plot(?)"** komutunu kullanalım. Burada en çok karşılan hata eksenlerdeki eleman sayılarının birbirine eşit olmamasıdır. Her X eksen elemanlarına karşılık aynı sayıda Y eksen elemanları olmalıdır. Ayrıca MATLAB fonksiyonlarının nasıl kullanıldığını öğrenmek için **"help *fonksiyonun adı"** komutunu kullanabilirsiniz.

```
%Yazar:Ozgur KURT
help plot
```

plot Linear plot.

plot(X,Y) plots vector Y versus vector X. If X or Y is a matrix, then the vector is plotted versus the rows or columns of the matrix, whichever line up. If X is a scalar and Y is a vector, disconnected line objects are created and plotted as discrete points vertically at X.

plot(Y) plots the columns of Y versus their index.

If Y is complex, **plot(Y)** is equivalent to **plot(real(Y),imag(Y))**. In all other uses of **plot**, the imaginary part is ignored.

Various line types, plot symbols and colors may be obtained with **plot(X,Y,S)** where S is a character string made from one element from any or all the following 3 columns:

b	blue	.	point	-	solid
g	green	o	circle	:	dotted
r	red	x	x-mark	-.	dashdot
c	cyan	+	plus	--	dashed
m	magenta	*	star	(none)	no line
y	yellow	s	square		
k	black	d	diamond		
w	white	v	triangle (down)		
		^	triangle (up)		
		<	triangle (left)		
		>	triangle (right)		
		p	pentagram		
		h	hexagram		

For example, **plot(X,Y,'c+:')** plots a cyan dotted line with a plus at each data point; **plot(X,Y,'bd')** plots blue diamond at each data point but does not draw any line.

plot(X1,Y1,S1,X2,Y2,S2,X3,Y3,S3,...) combines the plots defined by the (X,Y,S) triples, where the X's and Y's are vectors or matrices and the S's are strings.

For example, **plot(X,Y,'y-',X,Y,'go')** plots the data twice, with a solid yellow line interpolating green circles at the data points.

The **plot** command, if no color is specified, makes automatic use of the colors specified by the axes ColorOrder property. By default, **plot** cycles through the colors in the ColorOrder property. For monochrome systems, **plot** cycles over the axes LineStyleOrder property.

Note that RGB colors in the ColorOrder property may differ from similarly-named colors in the (X,Y,S) triples. For example, the second axes ColorOrder property is medium green with RGB [0 .5 0],

while **plot**(X,Y,'g') plots a green line with RGB [0 1 0].

If you do not specify a marker type, **plot** uses no marker.
If you do not specify a line style, **plot** uses a solid line.

plot(AX,...) plots into the axes with handle AX.

plot returns a column vector of handles to lineseries objects, one handle per plotted line.

The X,Y pairs, or X,Y,S triples, can be followed by parameter/value pairs to specify additional properties of the lines. For example, **plot**(X,Y,'LineWidth',2,'Color',[.6 0 0]) will create a plot with a dark red line width of 2 points.

Example

```
x = -pi:pi/10:pi;  
y = tan(sin(x)) - sin(tan(x));  
plot(x,y,'--rs','LineWidth',2,...  
      'MarkerEdgeColor','k',...  
      'MarkerFaceColor','g',...  
      'MarkerSize',10)
```

See also **plottools**, **semilogx**, **semilogy**, **loglog**, **plotyy**, **plot3**, **grid**, **title**, **xlabel**, **ylabel**, **axis**, **axes**, **hold**, **legend**, **subplot**, **scatter**.

Reference page for **plot**

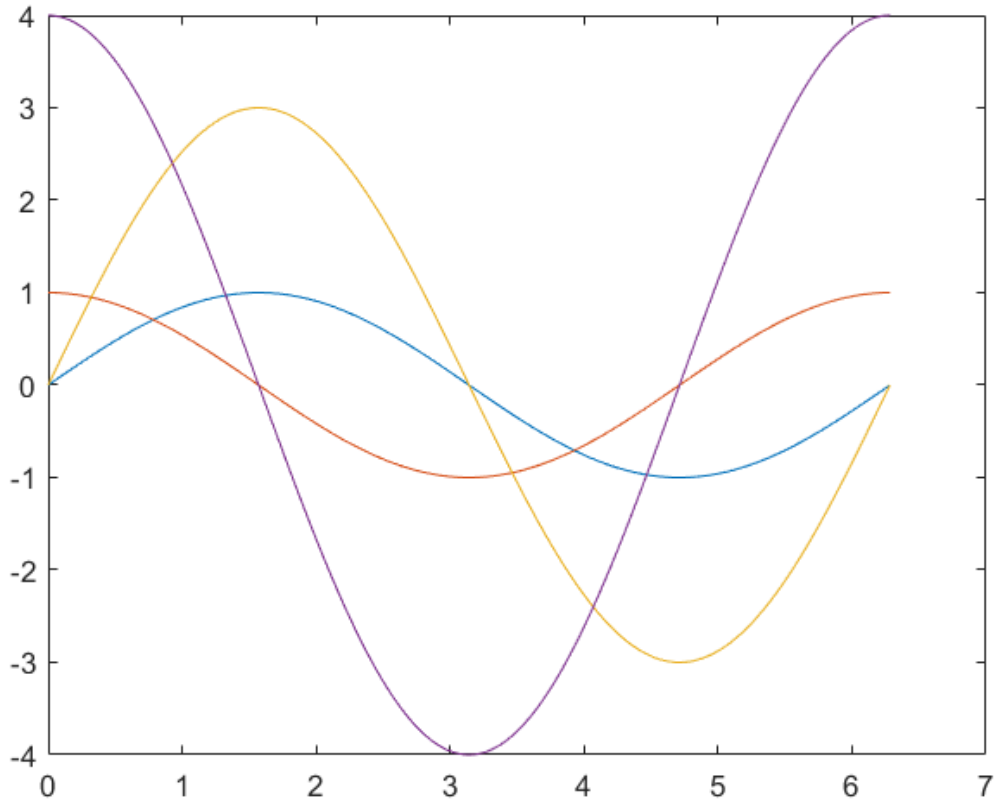
Other functions named **plot**

```
%X Ekseni  
x=0:pi/360:2*pi;  
%Y Ekseni  
y1=sin(x);  
y2=cos(x);  
y3=3*sin(x);  
y4=4*cos(x);
```

Birinci Yöntem

İlk yöntemde tek bir pencerede tek bir fonksiyon kullanarak birden fazla grafik çizdirmeyi gösterelim.

```
%Birinci Yontem  
plot(x,y1,x,y2,x,y3,x,y4)
```

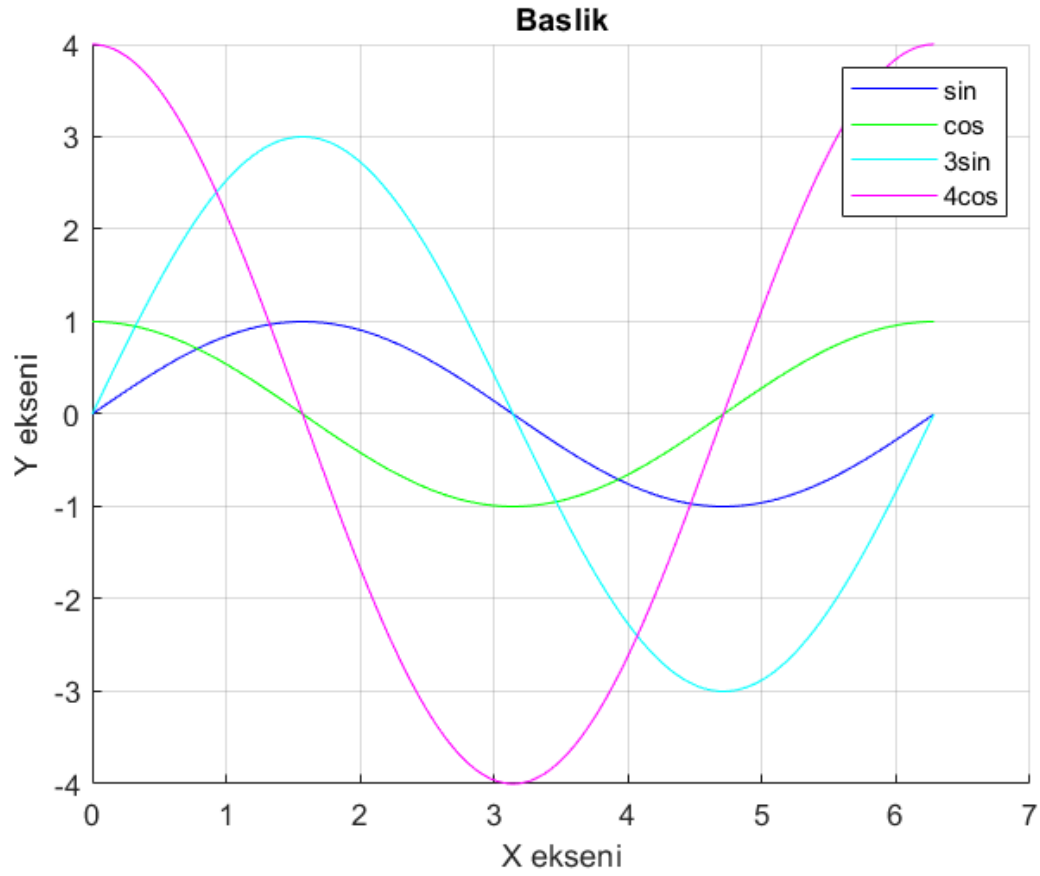


İkinci Yöntem

İkinci yöntemde yine tek bir pencerede birden fazla fonksiyon kullanarak birden fazla grafik çizdirmeyi gösterelim. Burada **"hold on"** komutu kullanarak grafiklerin aynı pencerede birlikte gösterilmesini sağlayalım. Ek olarak label,title,legend fonksiyonlarını da kullanarak özellikler ekleyelim.

```
%İkinci Yontem
figure
hold on
plot(x,y1,'b')
pause(1)
plot(x,y2,'g')
pause(1)
plot(x,y3,'c')
pause(1)
plot(x,y4,'m')
grid
pause(1)
xlabel('X eksenı')
pause(1)
ylabel('Y eksenı')
pause(1)
title('Baslık')
pause(1)
```

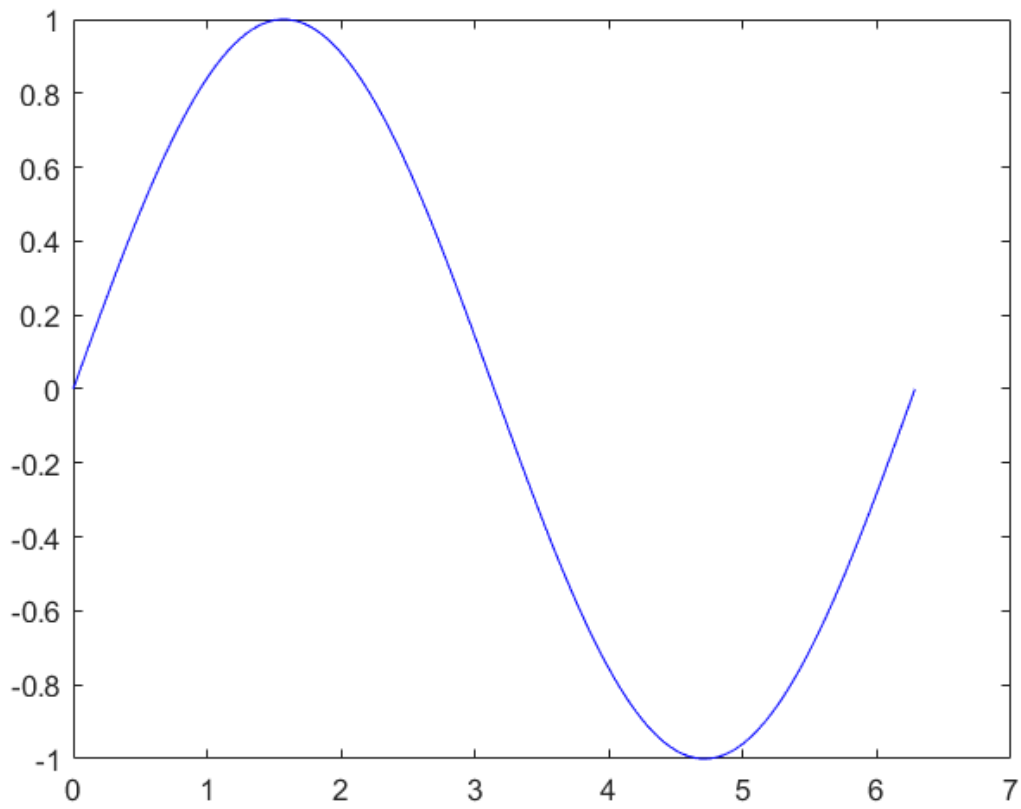
```
legend('sin','cos','3sin','4cos')
```



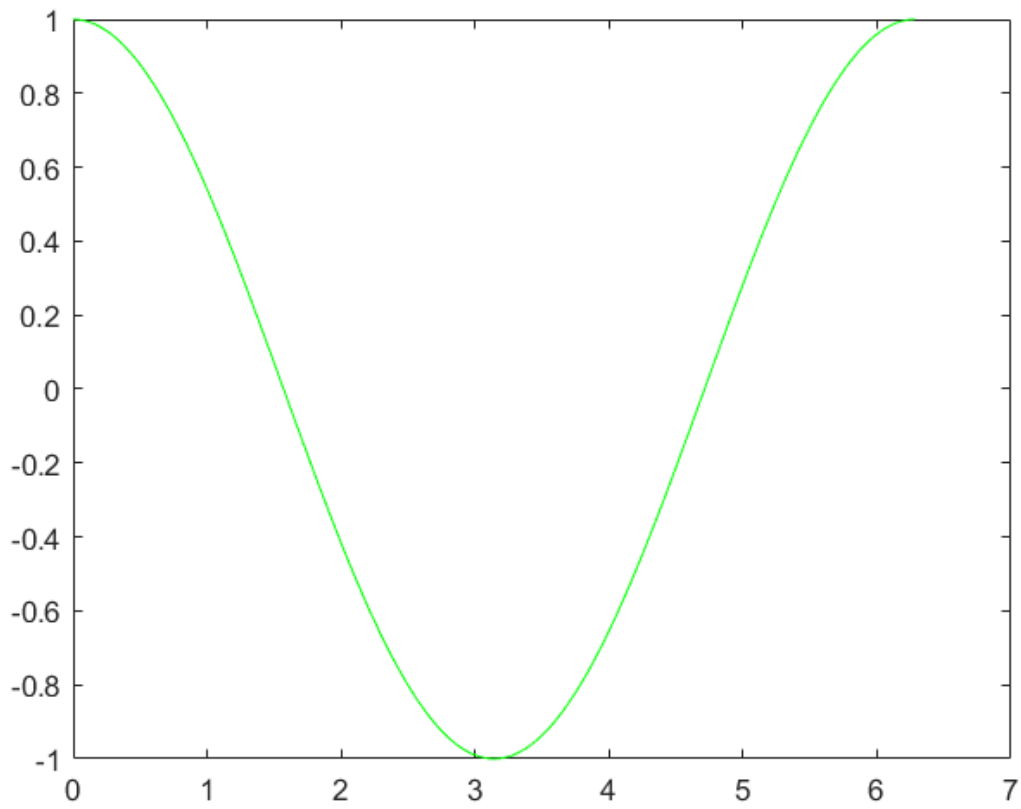
Üçüncü Yöntem

Üçüncü yöntemde birden fazla pencerede **"plot(?)"** fonksiyonunu kullanarak birden fazla grafik çizdirmeyi gösterelim. Burada her **"plot(?)"** fonksiyonunu kullanmadan önce **"figure"** kullanmaya dikkat edelim, **"figure"** komutunu kullanarak yeni pencere de açılmasını sağlayalım.

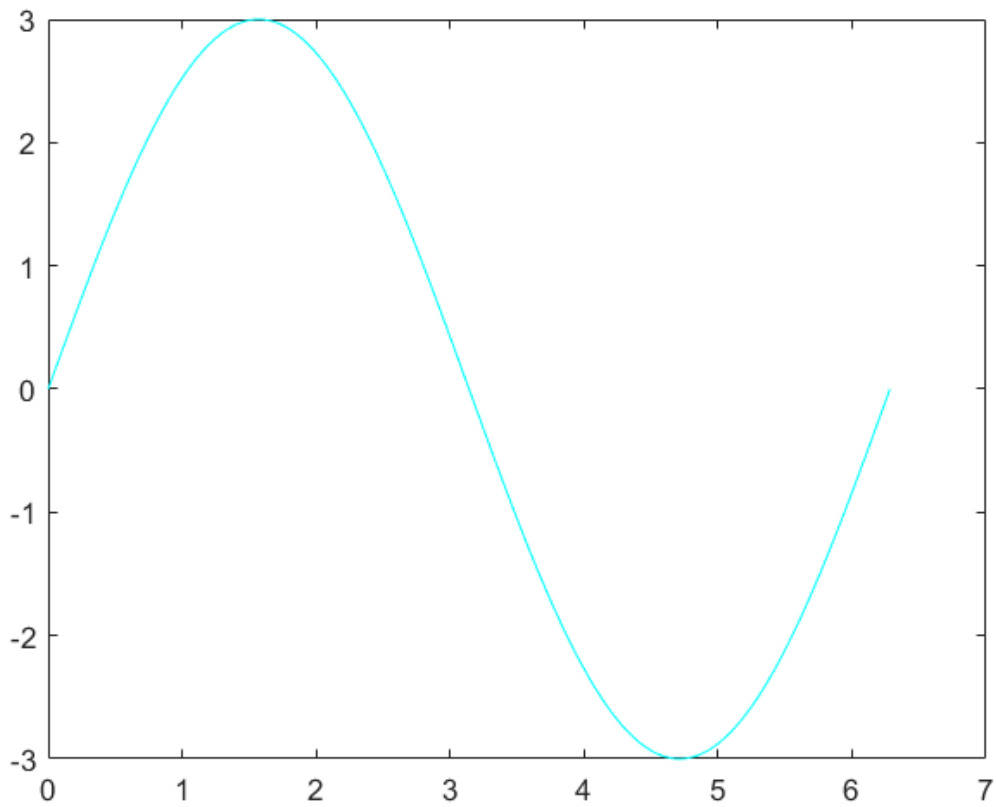
```
%Ucuncu Yontem  
figure  
plot(x,y1,'b')
```



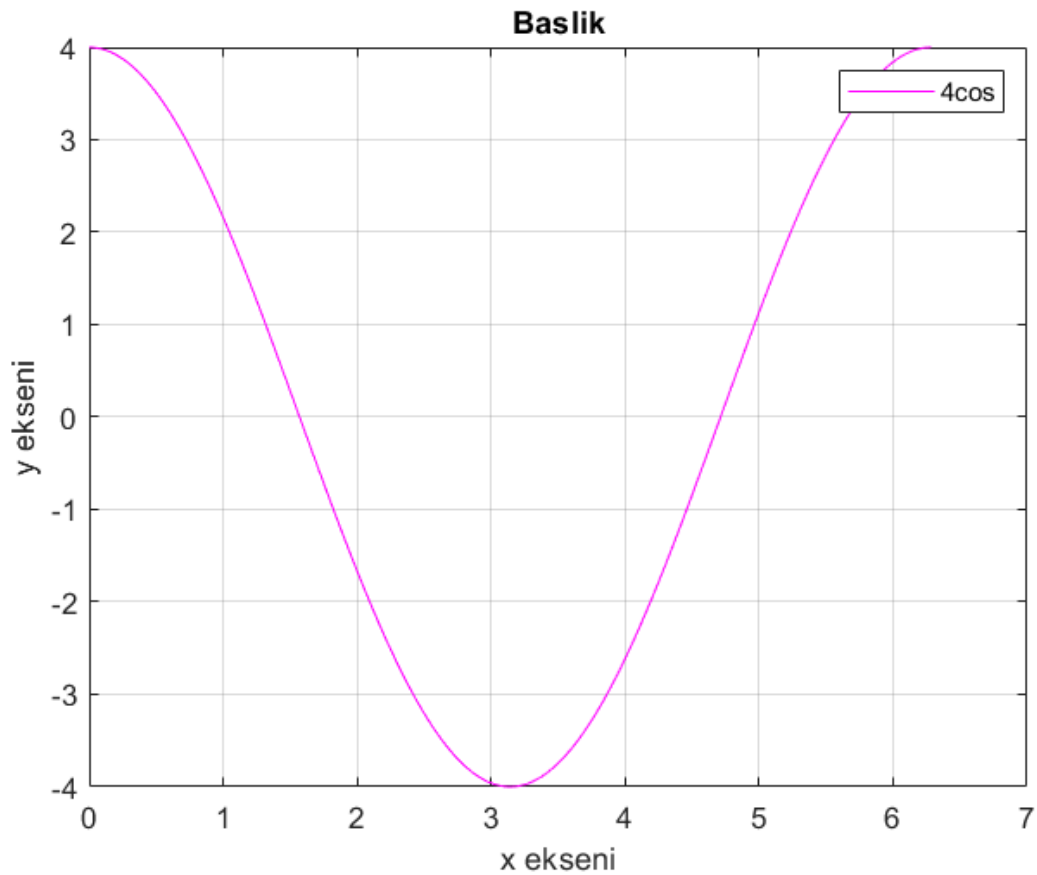
```
pause(1)  
figure  
plot(x,y2,'g')
```



```
pause(1)  
figure  
plot(x,y3,'c')
```



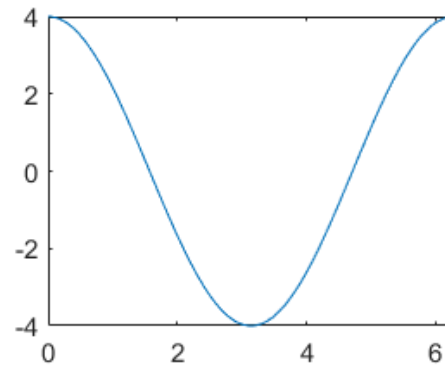
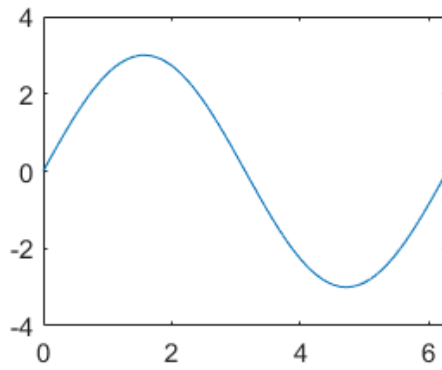
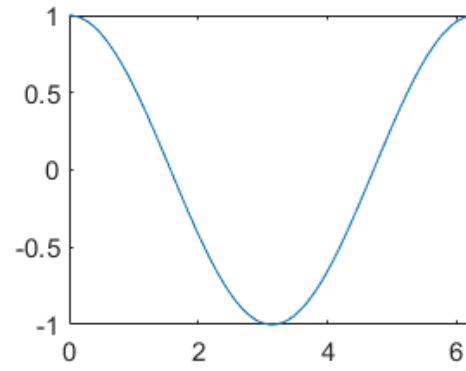
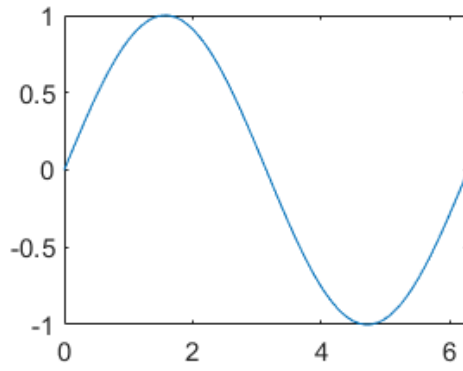
```
pause(1)
figure
plot(x,y4,'m')
grid
pause(1)
xlabel('x eksenı')
pause(1)
ylabel('y eksenı')
pause(1)
title('Baslık')
pause(1)
legend('4cos')
```

Dördüncü Yöntem

Dördüncü yöntemde ise tek bir pencerede alt grafikler çizdirelim. Bunun için "**subplot(?)**" fonksiyonunu kullanalım. Dört grafik için dört alt alan (2x2) oluşturmak için "**subplot(2,2,?)**" komutunu kullanalım. Eğer altı grafik olsaydı altı alt alan (3x2),(2x3) oluşturmak için "**subplot(3,2,?)**" veya "**subplot(2,3,?)**" komutlarını kullanabilirdik.

```
%Dorduncu yontem
subplot(2,2,1);
plot(x,y1)
pause(1)
subplot(2,2,2);
plot(x,y2)
pause(1)
subplot(2,2,3);
plot(x,y3)
pause(1)
subplot(2,2,4);
plot(x,y4)
```



Bütün Kodlar

```
% %Yazar:Ozgur KURT
% help plot
% %X Ekseni
% x=0:pi/360:2*pi;
% %Y Ekseni
% y1=sin(x);
% y2=cos(x);
% y3=3*sin(x);
% y4=4*cos(x);
% %-----
%
% %Birinci Yontem
% plot(x,y1,x,y2,x,y3,x,y4)
% %-----
%
% %Ikinci Yontem
% figure
% hold on
% plot(x,y1,'b')
% pause(1)
% plot(x,y2,'g')
% pause(1)
```

```

% plot(x,y3,'c')
% pause(1)
% plot(x,y4,'m')
% grid
% pause(1)
% xlabel('X eksenini')
% pause(1)
% ylabel('Y eksenini')
% pause(1)
% title('Baslik')
% pause(1)
% legend('sin','cos','3sin','4cos')
% %-----
%
% %Ucuncu Yontem
% figure
% plot(x,y1,'b')
% pause(1)
% figure
% plot(x,y2,'g')
% pause(1)
% figure
% plot(x,y3,'c')
% pause(1)
% figure
% plot(x,y4,'m')
% grid
% pause(1)
% xlabel('x eksenini')
% pause(1)
% ylabel('y eksenini')
% pause(1)
% title('Baslik')
% pause(1)
% legend('4cos')
% %-----
%
% %Dorduncu yontem
% subplot(2,2,1);
% plot(x,y1)
% pause(1)
% subplot(2,2,2);
% plot(x,y2)
% pause(1)
% subplot(2,2,3);
% plot(x,y3)
% pause(1)
% subplot(2,2,4);
% plot(x,y4)

```

NOT: *"pause"* komutu kodların daha iyi anlaşılabilmesi adına işlemleri yavaşlatmak için kullanılmıştır.
Kodların içerisinden silebilirsiniz.