Matlab Basics

Innova Lee(이상훈)
gcccompil3r@gmail.com

명령 창

```
>> why
The bald and not excessively bald and not excessively smart hamster obeyed a terrified and not excessively terrified hamster.
>> why
To fool the tall good and smart system manager.
>> why
The rich rich and tall and good system manager suggested it.
>> why
He wanted it that way.
>> why
The programmer suggested it.
>> why
Mara suggested it.
>> why
To please a very terrified and smart and tall engineer.
>> why
The tall system manager obeyed some engineer.
>> why
To satisfy some programmer.
>> why
Mary Ann wanted it that way.
>> why
Can you rephrase that?
>> why
Because Mary Ann wanted it that way.
>> why
How should I know?
>>
```

명령 창

```
>> x = 2;
>> y = 3 * (1 - x)^2

y =

     3

>> x = 1 + 2 * j;
>> y = 3 * (1 - x)^2

y =

   -12

>>
```

명령 창

```
>> x = [1 2 3];
>> y = 3 * (1 - x).^2

y =

     0     3    12

>> y = 3 * (1 - x)^2
다음 사용 중 오류가 발생함: ^
입력값은 스칼라 및 정사각 행렬이어야 합니다.
POWER를 요소별로 계산하려면 POWER (.^)을 사용하십시오.

>>
```

명령 창

```
>> a = 1 + j;
>> angle(a) * 180 / pi

ans =

    45

>> b = 1 - j;
>> angle(b) * 180 / pi

ans =

   -45
```

```
>> c = -1 + j;
>> angle(c) * 180 / pi

ans =

   135

>> d = -1 - j;
>> angle(d) * 180 / pi

ans =

  -135

fx >>
```

작업 공간

이름	값
a	1.0000 + 1.0000i
ans	-135
b	1.0000 - 1.0000i
c	-1.0000 + 1.0000i
d	-1.0000 - 1.0000i
x	[1,2,3]
y	[0,3,12]

명령 창

```
>> atan2(imag(c), real(c)) * 180 / pi

ans =

   135

>> atan2(imag(d), real(d)) * 180 / pi

ans =

  -135

>>
```

명령 창

```
>> atan(imag(a) / real(a)) * 180 / pi

ans =

    45

>> atan(imag(b) / real(b)) * 180 / pi

ans =

   -45

>> atan(imag(c) / real(c)) * 180 / pi

ans =

   -45

>> atan(imag(d) / real(d)) * 180 / pi

ans =

    45

>>
```

명령 창

```
>> x = 1 + 2i;
>> y = 3 + 4i;
>> abs(x + y)

ans =

    7.2111

>> sqrt(real(x + y)^2 + imag(x + y)^2)

ans =

    7.2111

fx >>
```

명령 창

```
>> t = [0:0.1:1]

t =

         0    0.1000    0.2000    0.3000    0.4000    0.5000    0.6000    0.7000    0.8000    0.9000    1.0000

>> t = linspace(0, 1, 11)

t =

         0    0.1000    0.2000    0.3000    0.4000    0.5000    0.6000    0.7000    0.8000    0.9000    1.0000
```

명령 창

```
>> ones(1, 3)

ans =

     1     1     1

>> zeros(3, 1)

ans =

     0
     0
     0

>>
```

명령 창

```
>> a = [1 1 1];
>> b = [1 1];
>> c = [1 0 0];
>> abc = conv(a, conv(b, c))

abc =

     1     2     2     1     0     0

>> roots(abc)

ans =

   0.0000 + 0.0000i
   0.0000 + 0.0000i
  -1.0000 + 0.0000i
  -0.5000 + 0.8660i
  -0.5000 - 0.8660i

>>
```

명령 창

```
>> a = 2;
>> b = [1 2 3];
>> a * b

ans =

     2     4     6

>> a ./ b

ans =

    2.0000    1.0000    0.6667

>> a / b
다음 사용 중 오류가 발생함: /
행렬의 차원이 일치해야 합니다.

fx >>
```

명령 창

```
>> c = [1 + j 2 + 2j 3 + 3j];
>> c'

ans =

   1.0000 - 1.0000i
   2.0000 - 2.0000i
   3.0000 - 3.0000i

>> conj(c')

ans =

   1.0000 + 1.0000i
   2.0000 + 2.0000i
   3.0000 + 3.0000i

>> c

c =

   1.0000 + 1.0000i   2.0000 + 2.0000i   3.0000 + 3.0000i

fx >>
```

명령 창

```
>> eye(2)

ans =

     1     0
     0     1

>> ones(2)

ans =

     1     1
     1     1

>> eye(3)

ans =

     1     0     0
     0     1     0
     0     0     1

fx >>
```

명령 창

```
>> A = [1 2 3; 4 5 6; 7 8 9];
>> tril(A)

ans =

     1     0     0
     4     5     0
     7     8     9

>> triu(A)

ans =

     1     2     3
     0     5     6
     0     0     9

fx >>
```

명령 창

```
>> A = [1 2 3; 4 5 6; 7 8 9];
>> A(2, 2)

ans =

     5

>> A(1, 1:2)

ans =

     1     2

>> A(1:2, 2)

ans =

     2
     5

>>
```

명령 창

```
>> A(1, :)

ans =

     1     2     3

>> A(:, 1)

ans =

     1
     4
     7

>> A(1:2, 2:3)

ans =

     2     3
     5     6

>>
```

명령 창

```
>> A = [5 3 -3; 3 2 -1; 2 -1 2];
>> B = [-1 -10 -8]'

B =

    -1
   -10
    -8

>> A

A =

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

>>
```

명령 창

```
>> x = A # B

x =

    0.2500
  -10.0000
   -9.2500

>> x = inv(A) * b
'b'은(는) 정의되지 않은 함수 또는 변수입니다.

정정 제안:
>> x = inv(A) * B

x =

    0.2500
  -10.0000
   -9.2500

>>
```

명령 창

```
>> A

A =

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

>> det(A)

ans =

    12

>> inv(A)

ans =

    0.2500   -0.2500    0.2500
   -0.6667    1.3333   -0.3333
   -0.5833    0.9167    0.0833

>> rank(A)

ans =

     3

fx >>
```

명령 창

```
>> B = [-1 2; -2 -1];
>> [V, D] = eig(B)

V =

   0.0000 - 0.7071i   0.0000 + 0.7071i
   0.7071 + 0.0000i   0.7071 + 0.0000i


D =

  -1.0000 + 2.0000i   0.0000 + 0.0000i
   0.0000 + 0.0000i  -1.0000 - 2.0000i

fx >>
```

명령 창

```
>> t1 = clock;
>> for i = 1:10000, a(i) = log(i);
end;
>> t = etime(clock, t1)

t =

   65.3540

>>
```

명령 창

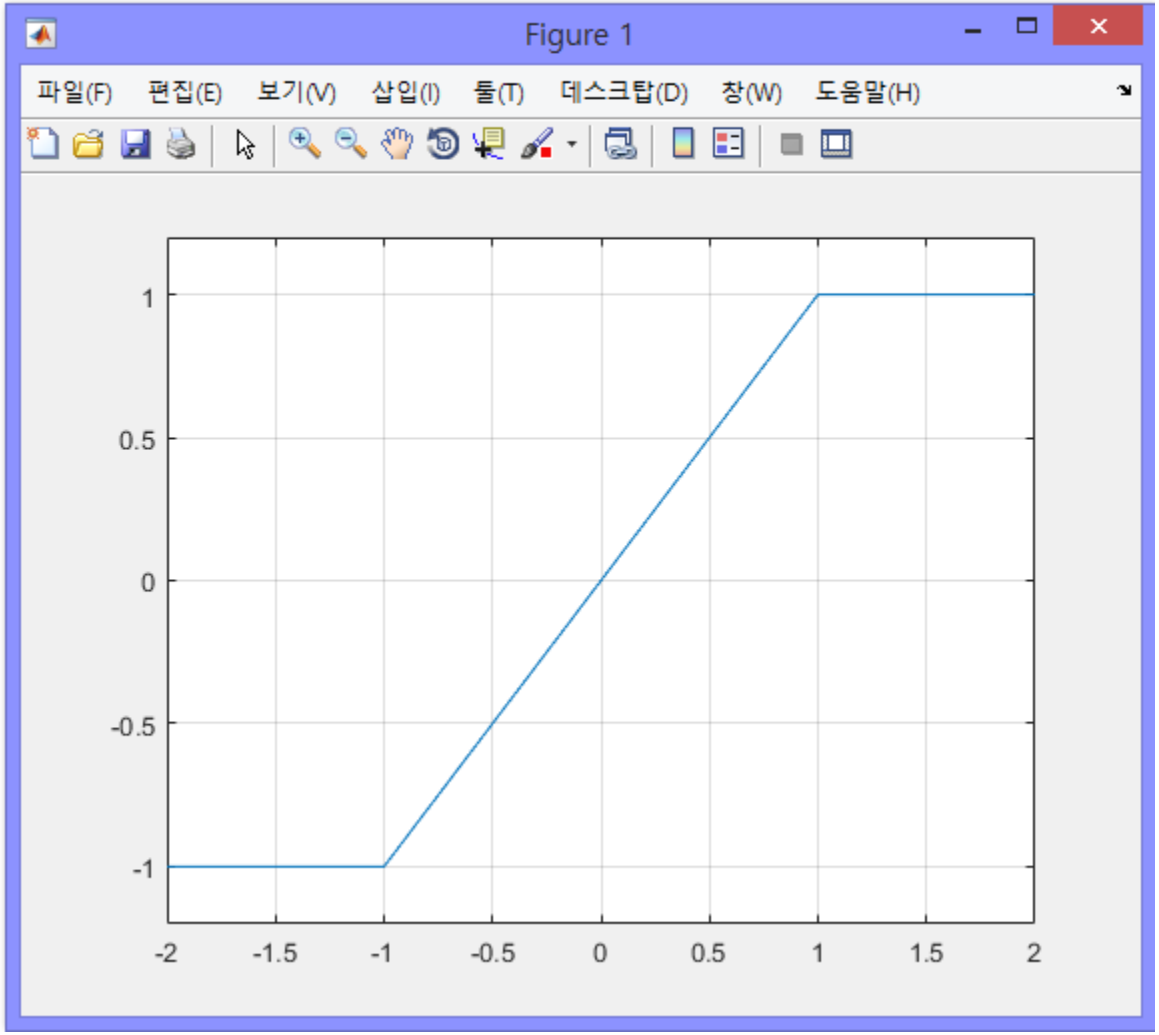
```
>> clear
>> x = [-2:0.01:2];
>> for i = 1:length(x),
if x(i) <= -1, y(i) = -1;
elseif x(i) <= 1, y(i) = x(i)
else y(i) = 1;
end
end
```

명령 창

```
열 211 ~ 225

  0.1000    0.1100    0.1200    0.1300    0.1400    0.1500    0.1600    0.1700    0.1800    0.1900

열 226 ~ 240

  0.2500    0.2600    0.2700    0.2800    0.2900    0.3000    0.3100    0.3200    0.3300    0.3400

열 241 ~ 255

  0.4000    0.4100    0.4200    0.4300    0.4400    0.4500    0.4600    0.4700    0.4800    0.4900

열 256 ~ 270

  0.5500    0.5600    0.5700    0.5800    0.5900    0.6000    0.6100    0.6200    0.6300    0.6400
```

명령 창

```
>> plot(x, y), axis([-2 2 -1.2 1.2]), grid
>>
```



명령 창

```
>> clear
>> t = [0:0.01:2];
>> nt = 300;
>> dt = [1 20 300];
>> syst = tf(nt, dt);
>> y = step(syst, t);
>> finalt = length(t);
>> yss = y(finalt);
>> for i = 1:finalt,
if y(i) > 1.02 * yss, ts = t(i);
elseif y(i) < 0.98 * yss, ts = t(i);
end
end
>> yss, ts, yup = 1.02 * ones(finalt, 1) * yss;

yss =

    1.0000


ts =

    0.3400

>> ydown = 0.98 * ones(finalt, 1) * yss;
>> plot(t, y, t, ydown, '--', t, yup, '--')
>> xlabel('time[sec]'), ylabel('y(t)'), title('Step Response')
>>
```

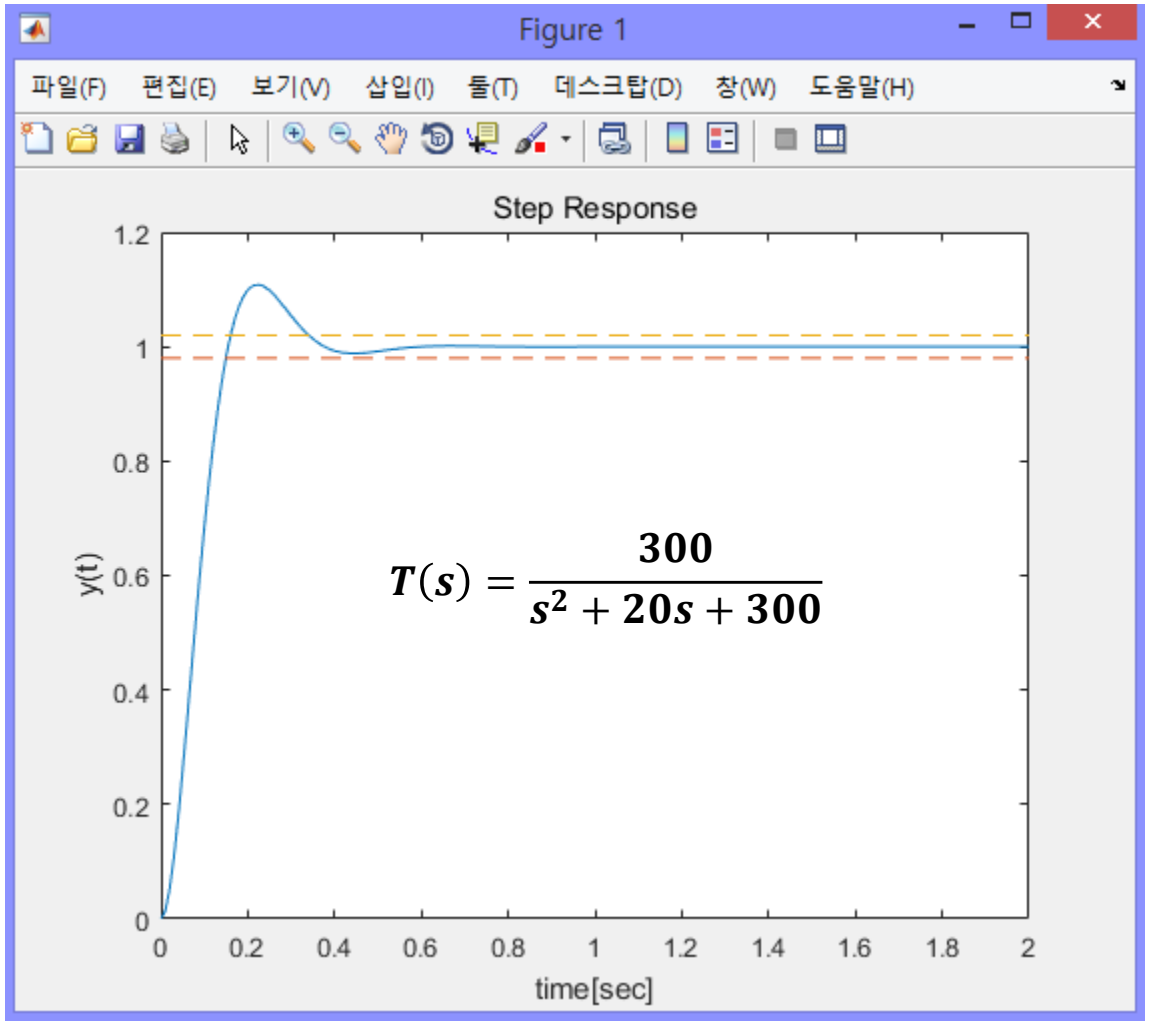


Figure 1
파일(F) 편집(E) 보기(V) 삽입(I) 툴(T) 데스크탑(D) 창(W) 도움말(H)
Step Response
1.2
1
0.8
0.6
0.4
0.2
0
y(t)
0 0.2 0.4 0.6 0.8 1 1.2 1.4 1.6 1.8 2
time[sec]
$T(s) = \frac{300}{s^2 + 20s + 300}$

inverse laplace transform (300 / (s^2 + 20s + 300)) * (1/s)

Assuming "s" is a variable | Use "s^2" as a unit instead

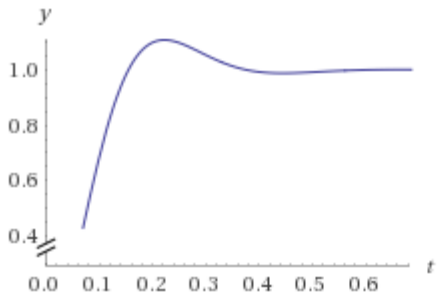
Input:

$$\mathcal{L}_s^{-1}\left[\frac{300}{s^2+20\,s+300}\times\frac{1}{s}\right](t)$$

Result:

$$300\left(\frac{1}{300}-\frac{e^{-10\,t}\left(\sin\left(10\sqrt{2}\;t\right)+\sqrt{2}\;\cos\left(10\sqrt{2}\;t\right)\right)}{300\sqrt{2}}\right)$$

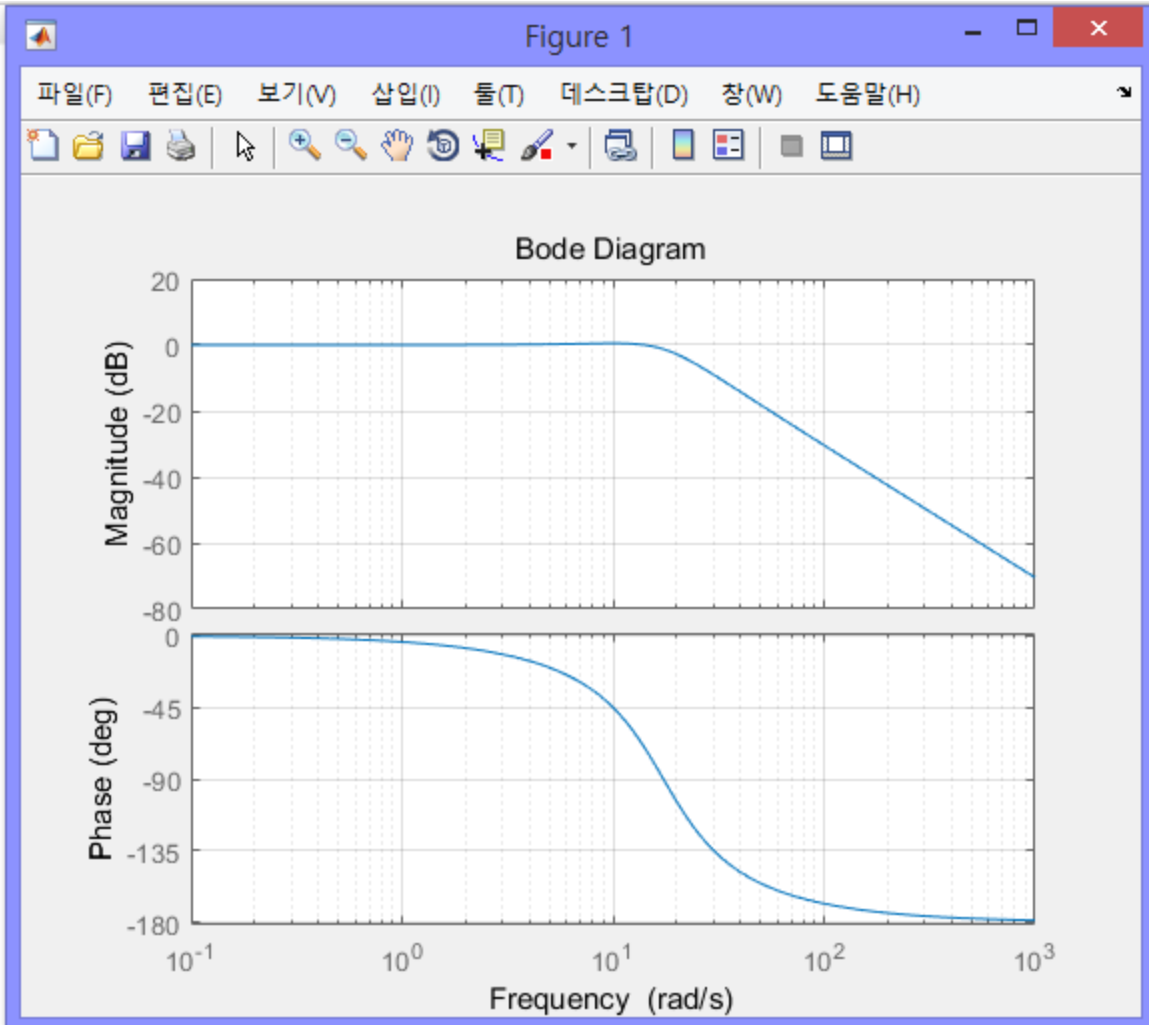
Plots:



(t from 0 to 0.7)

명령 창

```
>> clear
>> w = logspace(-1, 3, 400);
>> nt = 300;
>> dt = [1 20 300];
>> syst = tf(nt, dt);
>> [mag, phase] = bode(syst, w);
>> magdB = 20 * log10(mag);
>> index = 1;
>> for i = 1:length(w) - 1
if magdB(i) >= -3,
index = index + 1;
else,
index = index;
end
end
>> BW = w(index), bode(syst, w), grid

BW =

   20.6914

>>
```



명령 창

```
>> 3 < 5

ans =

     1

>> 3 > 5

ans =

     0

>> 3 == 5

ans =

     0

>> 3 == 3

ans =

     1

>>
```

명령 창

```
>> a = rand(2)

a =

    0.6324    0.2785
    0.0975    0.5469

>> b = triu(a)

b =

    0.6324    0.2785
         0    0.5469

>> a == b

ans =

     1     1
     0     1

>>
```

명령 창

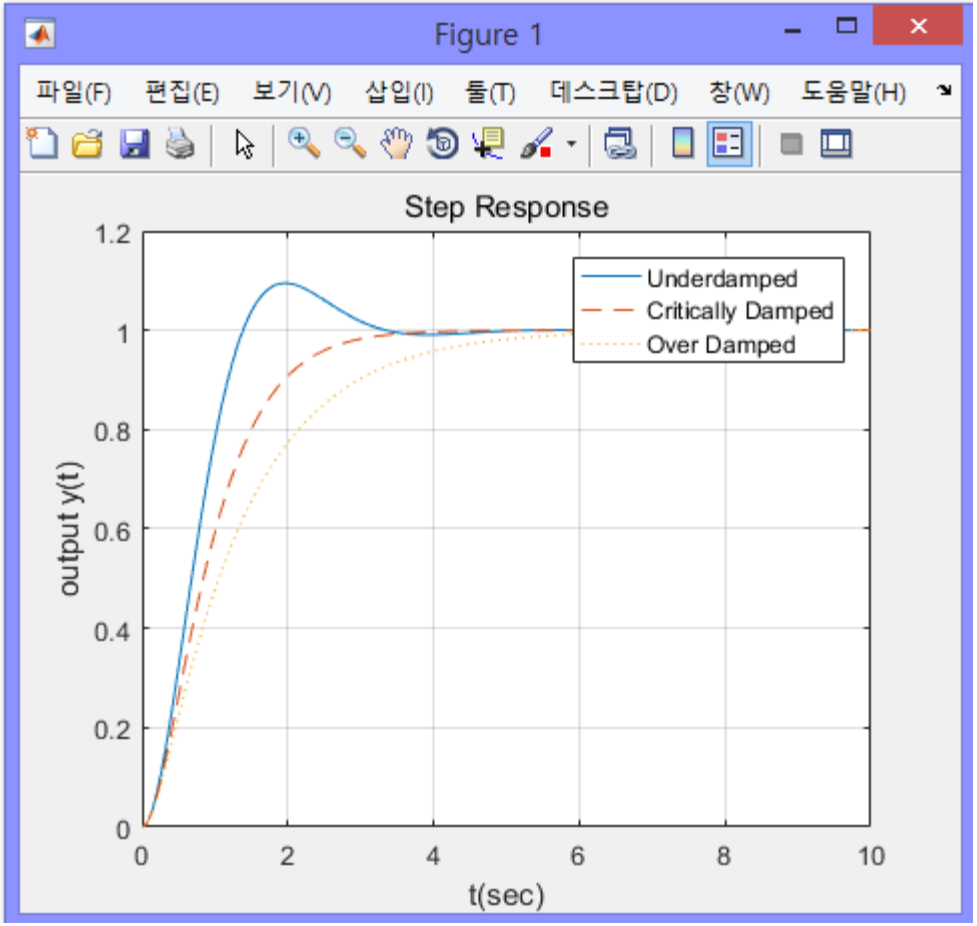
```
>> a = rand(3)

a =

    0.9575    0.9706    0.8003
    0.9649    0.9572    0.1419
    0.1576    0.4854    0.4218

>> b = triu(a)

b =

    0.9575    0.9706    0.8003
         0    0.9572    0.1419
         0         0    0.4218

>> c = tril(a)

c =

    0.9575         0         0
    0.9649    0.9572         0
    0.1576    0.4854    0.4218

>> a == c

ans =

     1     0     0
     1     1     0
     1     1     1

>>
```

명령 창

```
>> clear
>> x = 3;
>> if x ~= abs(x), error('x is not real')
elseif x ~= fix(x), error('x is not integer')
elseif x < 0, error('x is not positive')
end
>> y = rem(x, 2)

y =

     1

>> switch y
case 0, disp('x is even!')
otherwise, disp('x is odd!')
end
x is odd!
>>
```

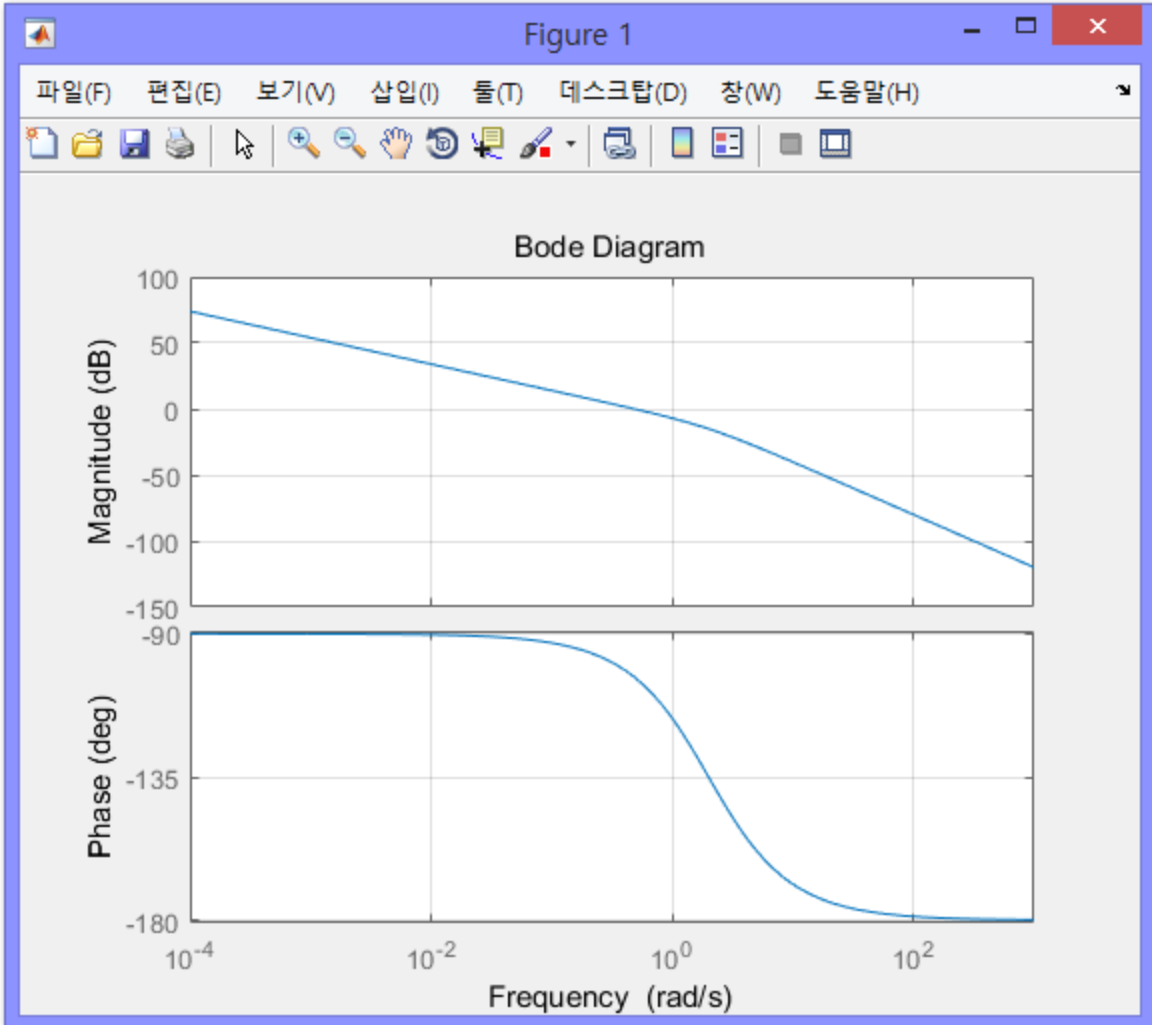
명령 창

```
>> wn = 2;
>> z1 = 0.6;
>> z2 = 1;
>> z3 = 1.4;
>> num = wn^2;
>> d1 = [1 2 * z1 * wn wn^2];
>> d2 = [1 2 * z2 * wn wn^2];
>> d3 = [1 2 * z3 * wn wn^2];
>> t = [0:0.01:10];
>> y1 = step(num, d1, t);
>> y2 = step(num, d2, t);
>> y3 = step(num, d3, t);
>> plot(t, y1, t, y2, '--', t, y3, ':'), grid
>> xlabel('t(sec)')
>> ylabel('output y(t)')
>> title('Step Response')
>> legend('Underdamped', 'Critically Damped', 'Over Damped')
>>
```



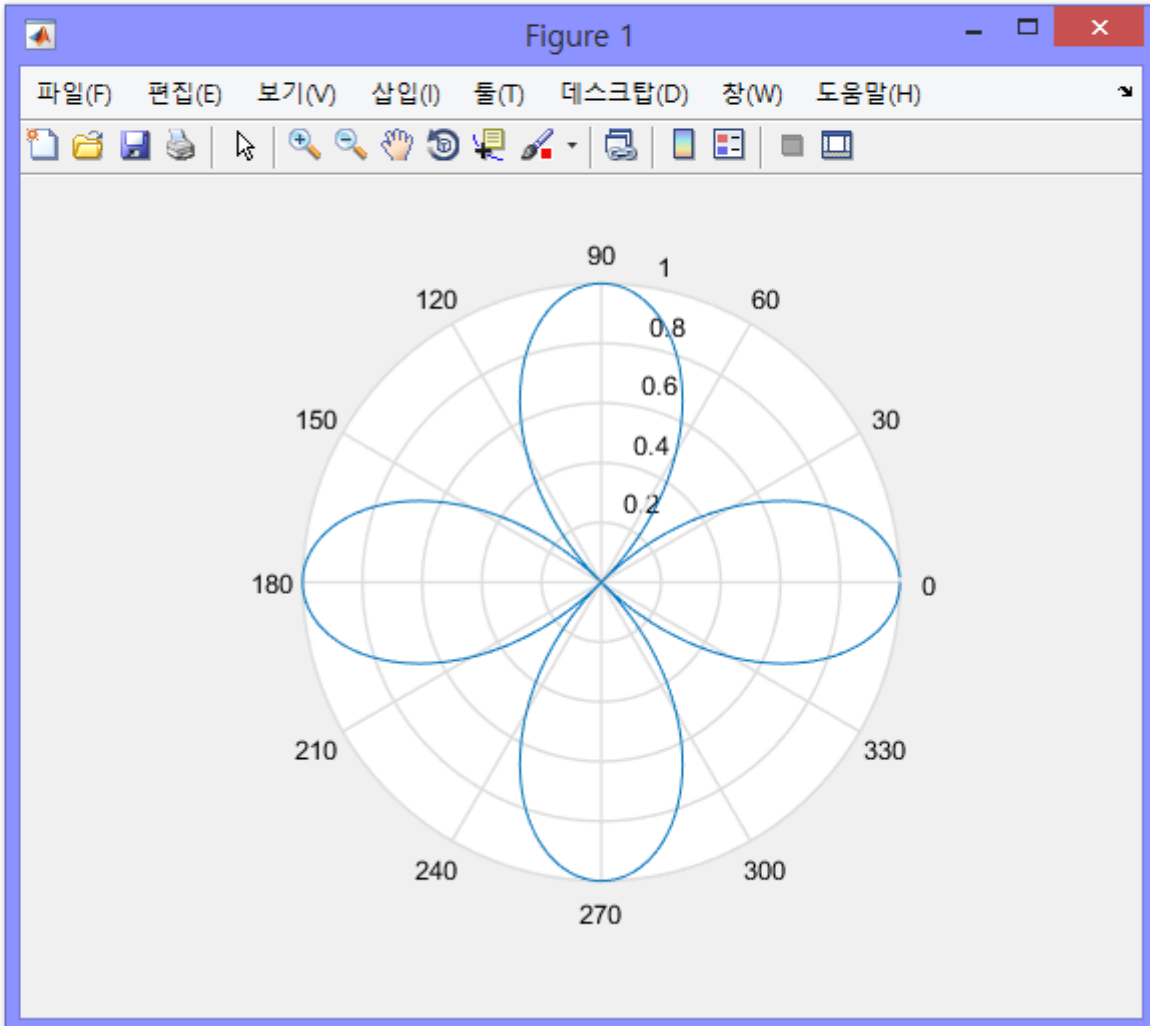
명령 창

```
>> w = logspace(-4, 3, 700);
>> num = [1];
>> den = [1 2 0];
>> sysp = tf(num, den);
>> bode(sysp, w), grid
>>
```



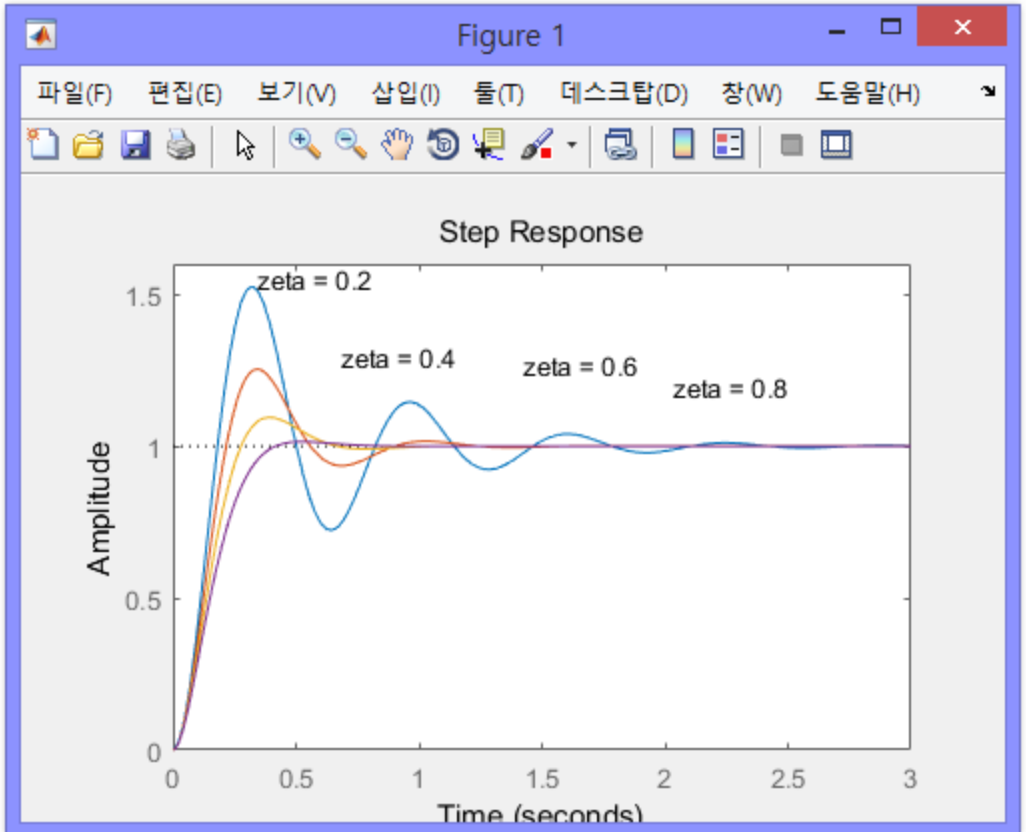
명령 창

```
>> theta = [pi / 200: pi/200: 2 * pi];
>> r = cos(2 * theta);
>> polar(theta, r), grid
>>
```



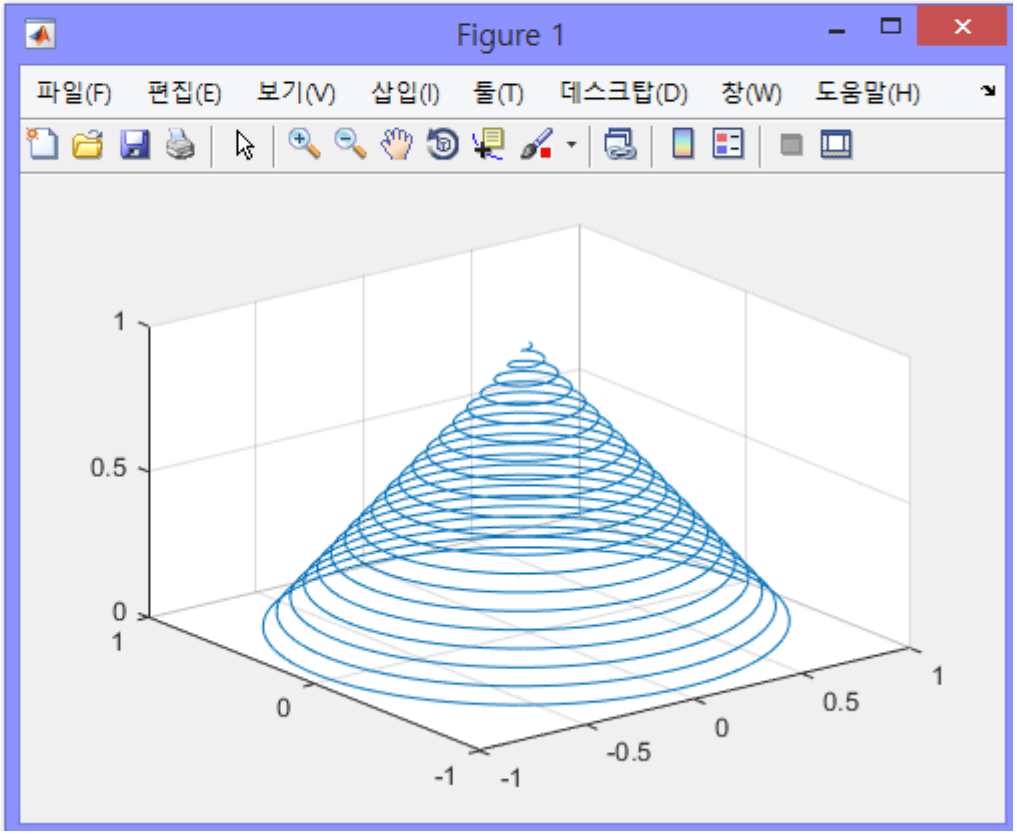
명령 창

```
>> zeta = [0.2:0.2:0.8];
>> for i = 1:length(zeta), num = [100];
den = [1 20 * zeta(i) 100];
step(num, den);
hold on
end
>> hold off, gtext('zeta = 0.2'), gtext('zeta = 0.4'), gtext('zeta = 0.6'), gtext('zeta = 0.8')
>>
```



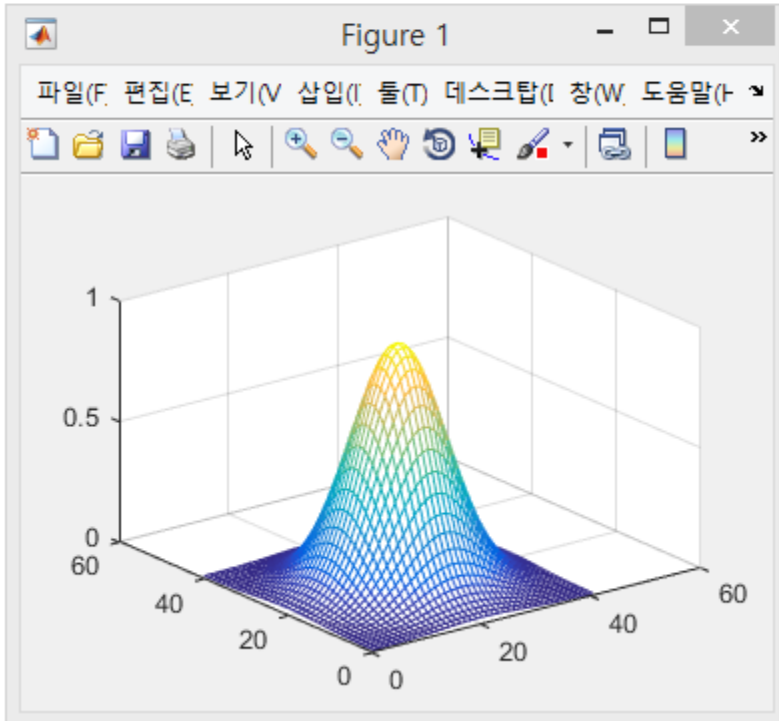
명령 창

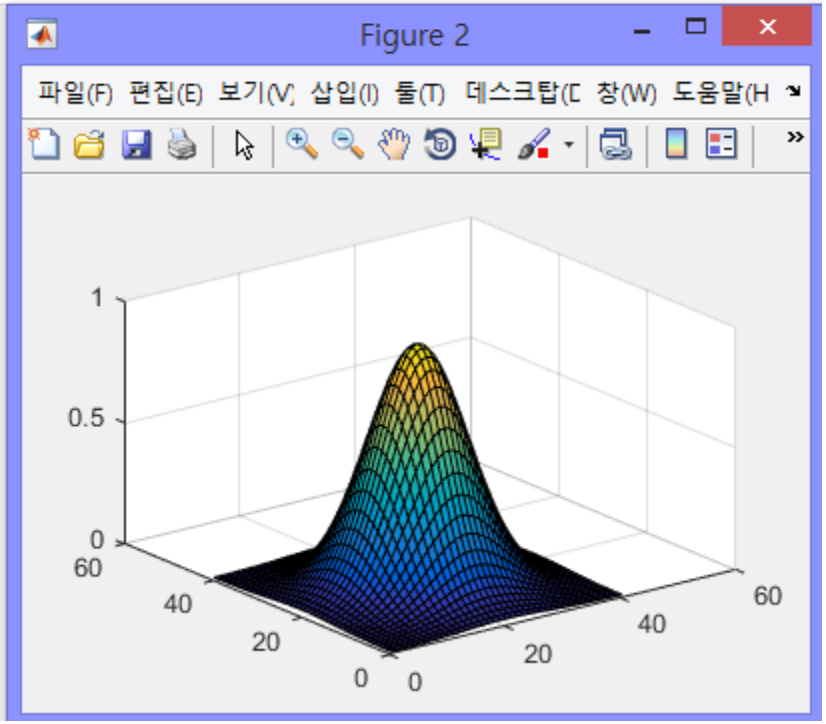
```
>> f = 20 * 2 * pi;
>> theta = linspace(0, f, 4000);
>> x = cos(theta) .* (f - theta) ./ f;
>> y = sin(theta) .* (f - theta) ./ f;
>> z = theta ./ f;
>> plot3(x, y, z), grid
>>
```



명령 창

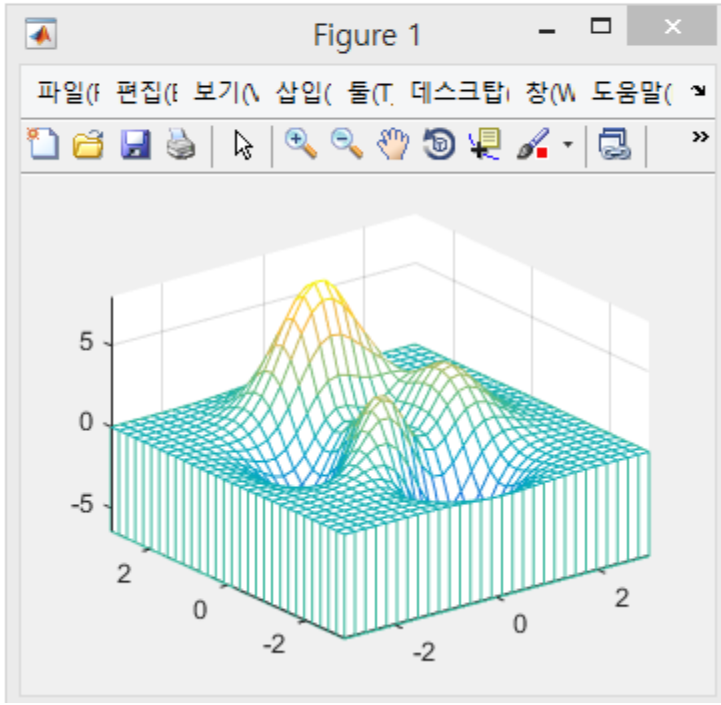
```
>> xx = [-2:0.1:2];
>> yy = xx;
>> [x, y] = meshgrid(xx, yy);
>> z = exp(-x.^2 - y.^2);
>> figure(1), mesh(z);
>> figure(2), surf(z);
>>
```

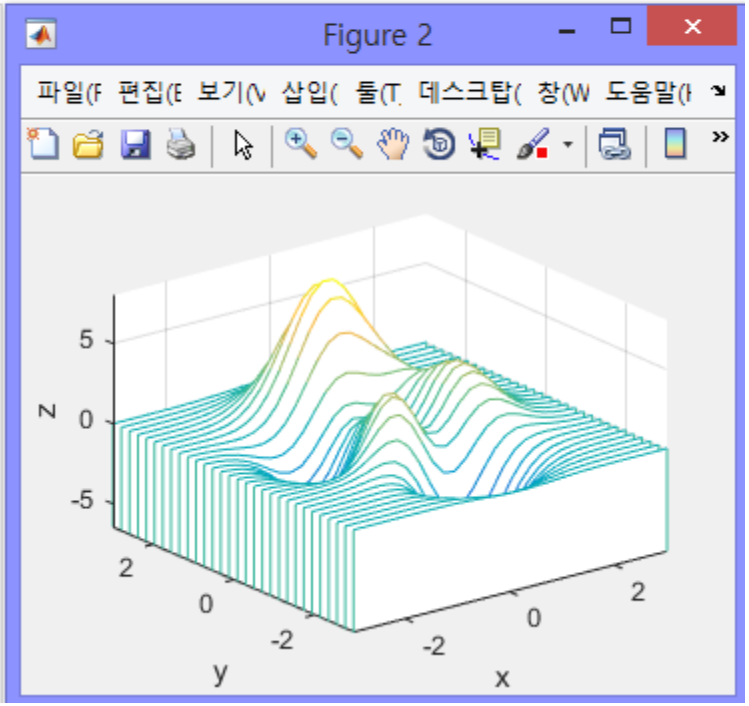




명령 창

```
>> [x, y, z] = peaks(30);
>> figure(1), meshz(x, y, z), axis('tight')
>> figure(2), waterfall(x, y, z), axis tight, xlabel('x'), ylabel('y'), zlabel('z')
>>
```





명령 창

```
>> [x, y, z] = peaks(30);
>> figure(1), contour(x, y, z, 10), colorbar
>> figure(2), contourf(x, y, z, 10), colorbar
>>
```