

# MATLAB 실습 1

## -기본 구성 및 그래프 그리기-

박사과정 서종상

azuresky@snu.ac.kr

Tel:02-880-1942

301-113

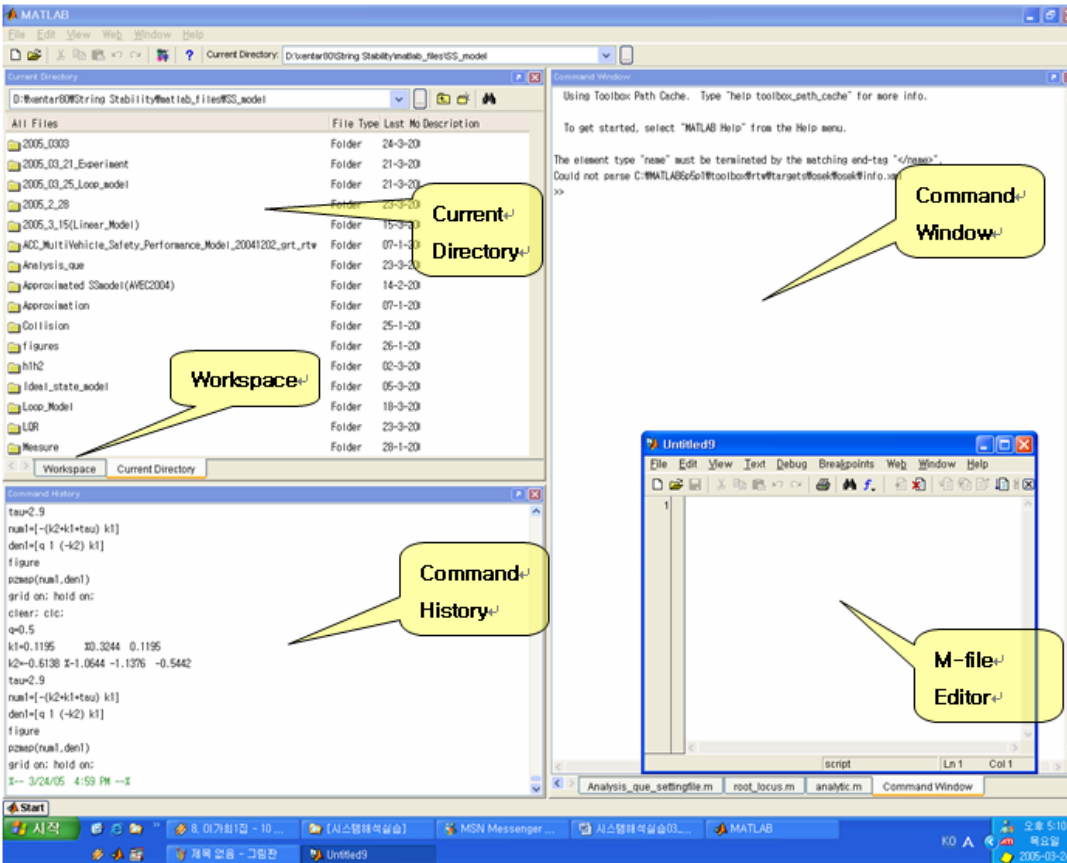
Vehicle Dynamics & Control  
Laboratory

# MATLAB 이란?

---

1. Mathworks사에서 개발한 수치 해석 및 프로그래밍 환경을 제공하는 공학용 소프트웨어
2. 함수와 데이터의 그래프 표현, 알고리즘 적용, 사용자 인터페이스 생성, 다른 프로그래밍 언어와의 연결 가능
3. 응용 개발 도구로 사용할 수 있도록 GUI(Graphical User Interface)도구 제공
4. 수치해석 뿐만 아니라 하드웨어, 소프트웨어 개발까지 여러 방면에서 이용 가능한 프로그램

# MATLAB 구성



1. Command Window : 명령어를 직접 입력하고, 결과를 보여주는 창
2. M-file Editor : 명령어들을 이용하여 사용자 프로그램 작성하는 창
3. Current Directory : 현재 작업이 이루어지는 창, 작업 파일을 저장하거나 로딩을 할 때 쓰는 디렉토리(폴더)를 보여주는 창
4. Workspace : 연산을 할 때 쓰이는 변수들이 메모리에 어떻게 저장되는지 보여주는 창
5. Command History : Command Window에 입력했던 명령어들을 보여주는 창

# Workspace와 기본 계산 [1/6]

```
a = 5;  
b = 3;  
A = 7;  
B = 9;
```

The screenshot shows the MATLAB 7.4.0 (R2007a) Workspace window. The menu bar includes File, Edit, View, Graphics, Debug, Desktop, and Window. The toolbar contains icons for file operations and help. The Workspace window title is "Workspace" and "Current Directory". The Stack is set to "Base". The workspace contains the following variables:

Name	Value
A	7
B	9
a	5
b	3

```
A = [5 15; 20 25];  
B = [7 22; 40 11];
```

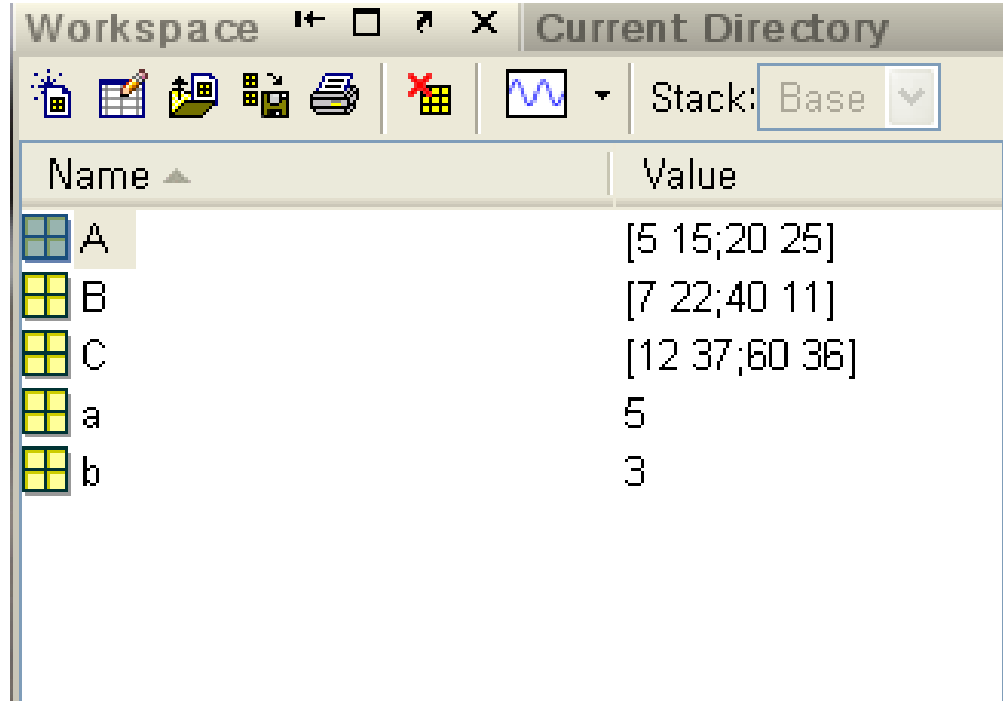
The screenshot shows the MATLAB 7.4.0 (R2007a) Workspace window. The menu bar includes File, Edit, Text, Go, Cell, Tools, Debug, and Desktop. The toolbar contains icons for file operations and help. The Workspace window title is "Workspace" and "Current Directory". The Stack is set to "Base". The workspace contains the following variables:

Name	Value
A	[5 15; 20 25]
B	[7 22; 40 11]
a	5
b	3

# Workspace와 기본 계산 [2/6]

C = A + B;

각각의  
성분끼리  
더해짐



The screenshot shows a workspace window titled "Workspace" with a "Current Directory" tab. The window contains a table with two columns: "Name" and "Value". The table lists five variables: A, B, C, a, and b. Each variable is represented by a small grid icon. The values for A, B, and C are 2x2 matrices, while a and b are scalars. The stack is currently empty, showing "Base" as the current directory.

Name	Value
A	[5 15; 20 25]
B	[7 22; 40 11]
C	[12 37; 60 36]
a	5
b	3











# 반복문 for

```
for k = k의 초기값 : k의 증분값 : k의 최종값
```

```
( 한 loop동안 실행할 명령문)
```

```
end
```

**for문은 k의 초기 값에서 시작해서 1루프 당 증분 값만큼 증가시켜 k가 최종 값에 도달할 때까지 루프를 반복시킨다. 만약 증분값이 생략되면 기본적으로 1의 증분값을 가진다.**

# 반복문 for 예시 1

## 1) m 파일 editor window (명령어 입력)

```
clc; clear;      % 'clc'는 command 창을 초기화, 'clear'는 workspace를 초기화
for i=1:2:9      % 1부터 9 까지 2씩 증가하여 9가 될 경우 까지 loop를 반복. 이 경우 값이 변수 'i' 에 저장
    i            % i 에 semi-colon(:)이 없으므로 command 창에 i의 값이 출력
end              % for 명령문의 끝 i=9 인 경우 다음 줄 명령 실행하며 아닌 경우 다시 위의 for문으로 돌아가 다음 i 값 저장, loop반복
```

## 2) Command window (결과)

```
i =
    1
i =
    3
i =
    5
i =
    7
i =
    9
>>
```

# 반복문 for 예시 2 [1/2]

## 1) m 파일 editor window (명령어 입력)

```
clc; clear;
a=[3 8 4];           % a라는 변수에 1X3 vector 값을 저장 [3 8 4]
b=[5 2 7; 3 6 8; 1 4 9]; % b라는 변수에 3X3 행렬값을 저장
for i=1:3            % 1부터 3 까지 1씩 증가하여 3이 될 경우 까지 loop를 반복. 이 경우 값이 변수 'i' 에 저장
    a(i)             % a라는 vector의 i번째 항을 출력 즉, i=1인 경우 a(1)=3이므로 3이 출력.
    b(:,i)           % b라는 행렬의 i번째 열(column)을 출력
    b(i,:)           % b라는 행렬의 i번째 행(row)를 출력
end
```

# 반복문 for 예시 2 [2/2]

## 2) Command window (결과)

```
ans =  
    3  
  
ans =  
    5  
    3  
    1  
  
ans =  
    5    2    7  
  
ans =  
    8  
  
ans =  
    2  
    6  
    4  
  
ans =  
    3    6    8
```

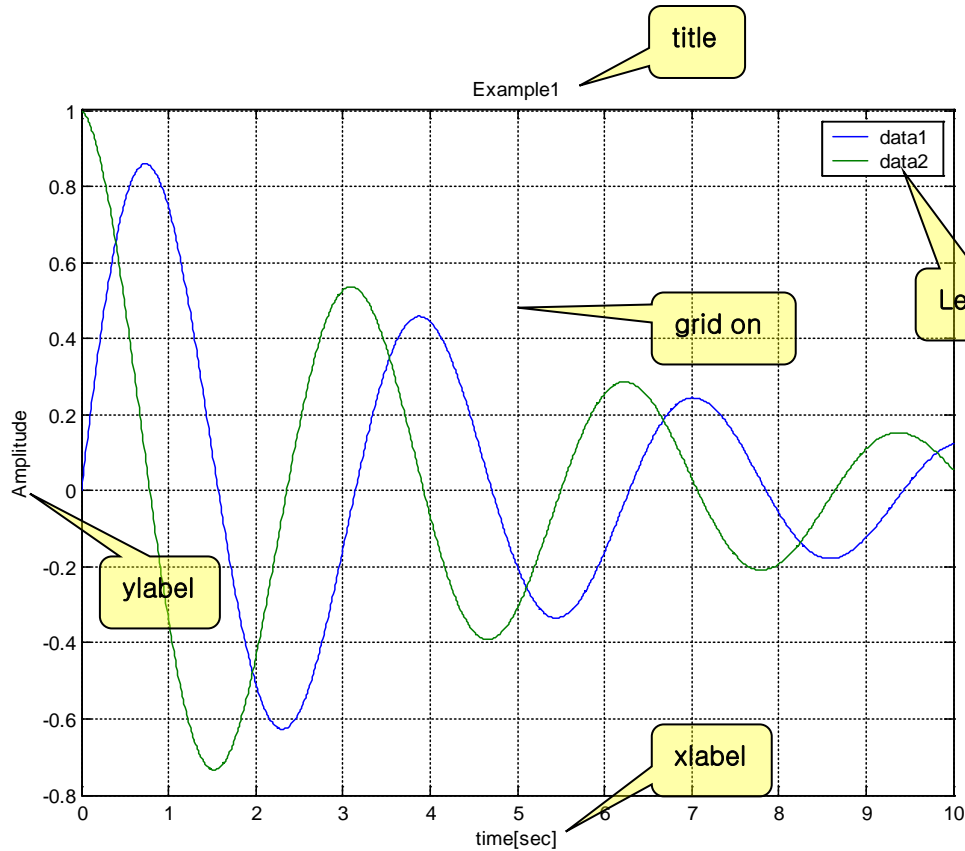
i=1인 경우  
a(1)  
b의 1열  
b의 1행의 값

i=2인 경우  
a(2)  
b의 2열  
b의 2행의 값

```
ans =  
    4  
  
ans =  
    7  
    8  
    9  
  
ans =  
    1    4    9  
  
>>
```

i=3인 경우  
a(3)  
b의 3열  
b의 3행의 값

# MATLAB에서 그래프 그리기 [1/3]



## - m-file 구성

```
figure(1)           % (1)번 그래프 창을 생성
t=0:0.01:10;       % 시작점:간격:끝점
y1=exp(-0.2*t).*sin(2*t); % t에 대한 결과값을 y에 저장
plot(t,y1)         % 그래프를 그림
hold on;           % 이미 그려진 그래프를 지우지 않고 유지
y2=exp(-0.2*t).*cos(2*t); % t에 대한 결과값을 y에 저장
plot(t,y2);        % 그래프를 그림
grid on;           % 격자선의 출력
legend('data1','data2'); % 범례의 생성
title('Example1'); % 그래프 제목생성
xlabel('time[sec]'); % x축 라벨의 생성
ylabel('Amplitude'); % y축 라벨의 생성
```

# MATLAB에서 그래프 그리기 [2/3]

## 1) 가능한 선의 color

Matlab 에서의 기호	Color
c	Cyan
m	Magenta
y	Yellow
r	Red
g	Green
b	Blue
w	White
k	black

## 2) 가능한 선의 style

Matlab 에서의 기호	Style
-	Solid line
--	Dashed line
:	Dotted line
-.	Dash-dot line

# MATLAB에서 그래프 그리기 [3/3]

## 3) 가능한 선의 marker

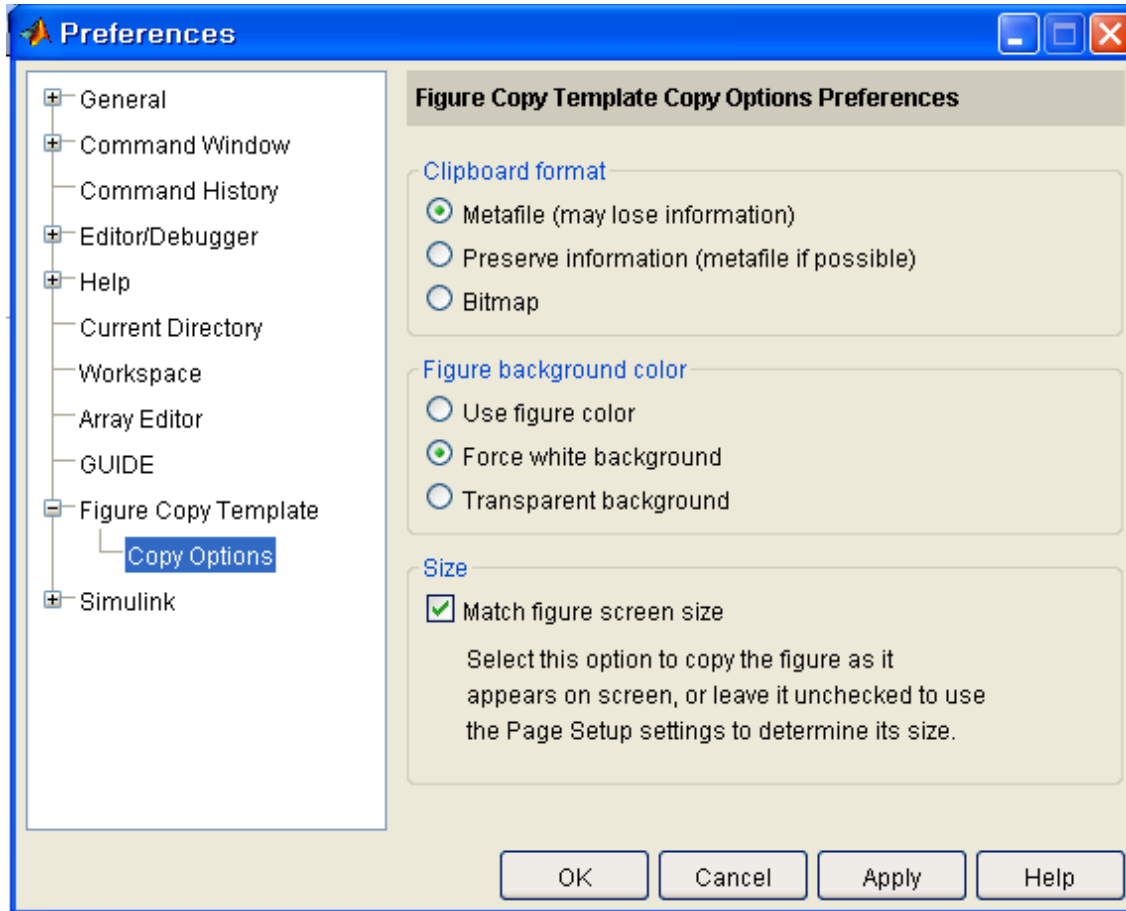
Matlab 에서의 기호	Marker style
+	+
O	O
*	*
.	•
x	X
square	□
diamond	◇

Matlab 에서의 기호	Marker style
^	△
v	▽
<	◁
>	▷
pentagram	☆
hexagram	*
none	default

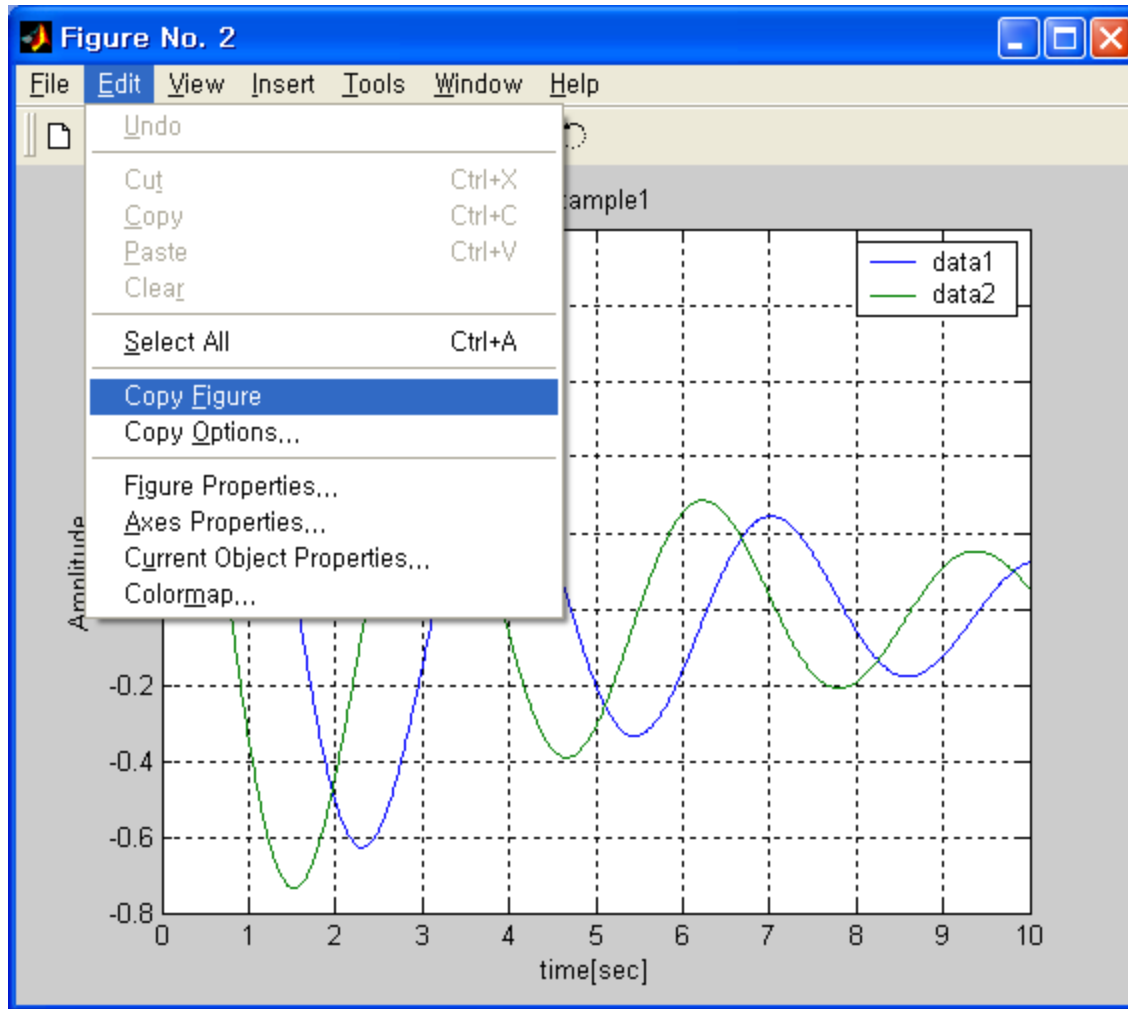


# 문서에 그래프 옮기기 Tip [1/2]

- File – Preference – Figure Copy Template – Copy Options에서 아래와 같이 설정



# 문서에 그래프 옮기기 Tip [2/2]



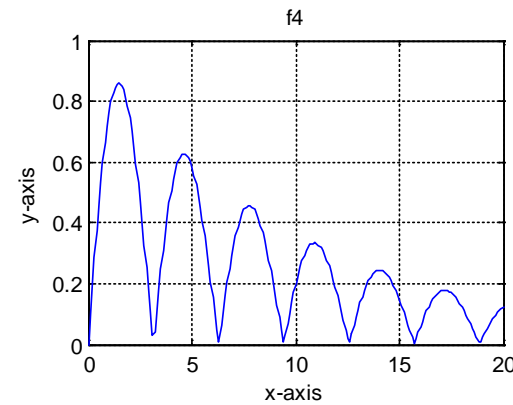
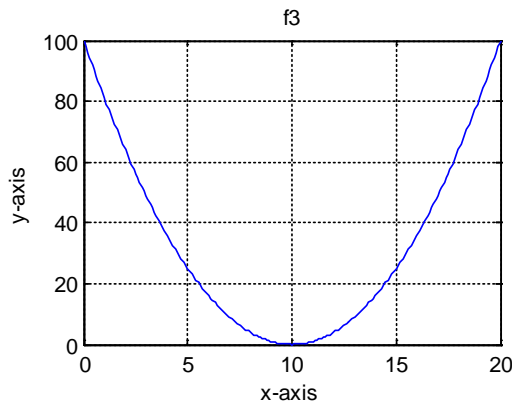
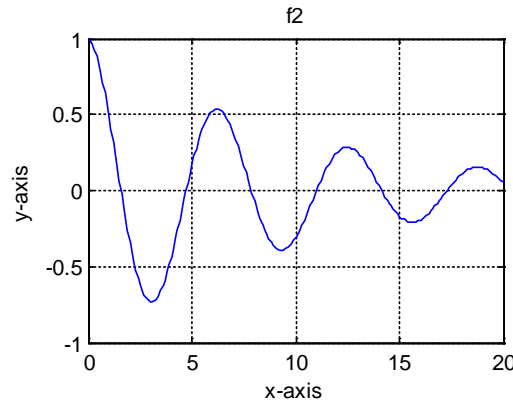
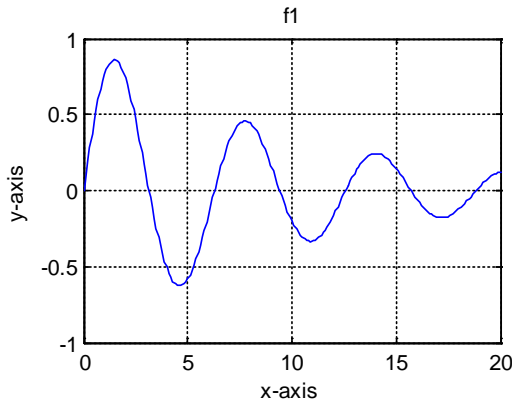
▪ Figure창에서 Edit – Copy Figure로 그래프를 복사

-왼쪽의 과정을 수행한 후 문서에  
그림을 붙여 넣기 한다.

- 이때 문서에 붙여 넣을 정도의 크기로 미리 창의  
크기를 조절하여야 그래프의 문자들이 깨지지 않는다.

# 그래프 그리기 Tip – subplot [1/2]

`subplot(row,column,index)`    % (행의갯수, 열의갯수, 그래프번호)



## 함수의 정의

```
t=0:0.1:20;  
f1=exp(-0.1*t).*sin(t);  
f2=exp(-0.1*t).*cos(t);  
f3=(t-10).^2;  
f4=exp(-0.1*t).*abs(sin(t));
```

## subplot

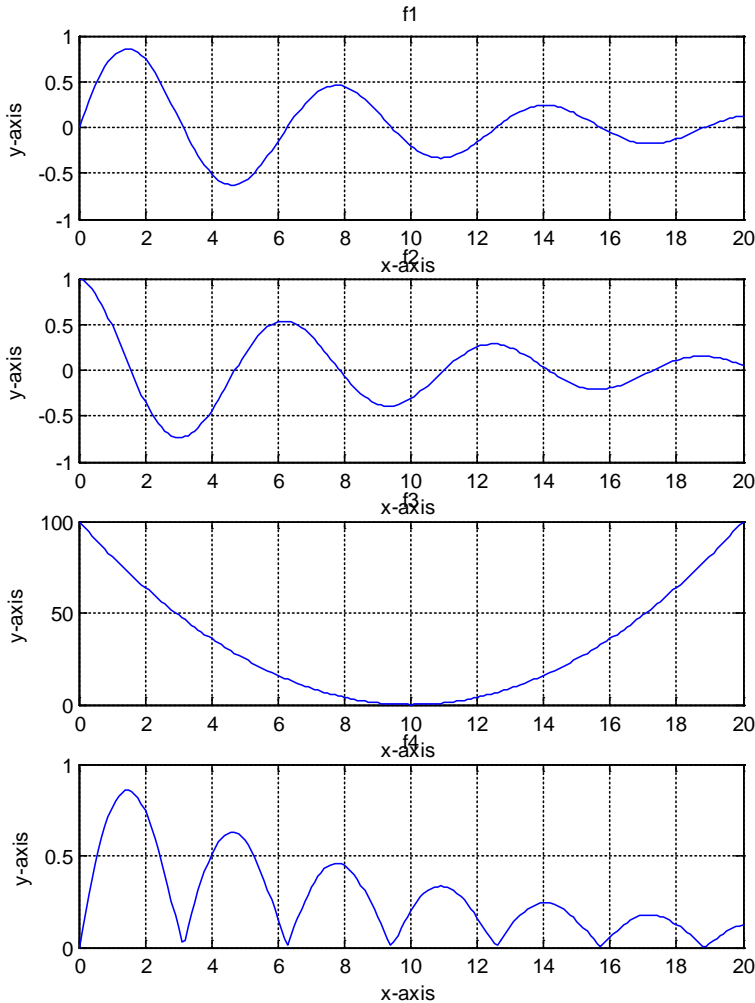
```
subplot(221)  
plot(t,f1);  
title('f1'); xlabel('x-axis'); ylabel('y-axis');  
grid on;
```

```
subplot(222)  
plot(t,f2);  
title('f2'); xlabel('x-axis'); ylabel('y-axis');  
grid on;
```

```
subplot(223)  
plot(t,f3);  
title('f3'); xlabel('x-axis'); ylabel('y-axis');  
grid on;
```

```
subplot(224)  
plot(t,f4);  
title('f4'); xlabel('x-axis'); ylabel('y-axis');  
grid on;
```

# 그래프 그리기 Tip – subplot [2/2]



## 함수의 정의

```
t=0:0.1:20;  
f1=exp(-0.1*t).*sin(t);  
f2=exp(-0.1*t).*cos(t);  
f3=(t-10).^2;  
f4=exp(-0.1*t).*abs(sin(t));
```

## subplot

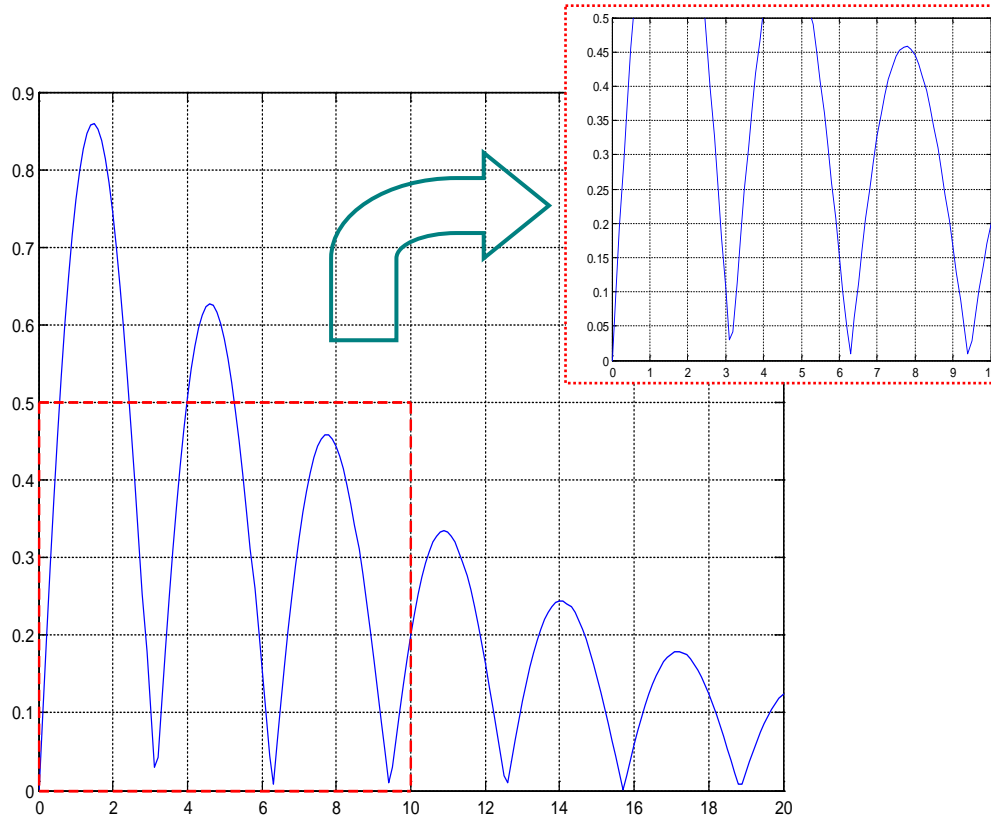
```
subplot(411)  
plot(t,f1);  
title('f1'); xlabel('x-axis'); ylabel('y-axis');  
grid on;
```

```
subplot(412)  
plot(t,f2);  
title('f2'); xlabel('x-axis'); ylabel('y-axis');  
grid on;
```

```
subplot(413)  
plot(t,f3);  
title('f3'); xlabel('x-axis'); ylabel('y-axis');  
grid on;
```

```
subplot(414)  
plot(t,f4);  
title('f4'); xlabel('x-axis'); ylabel('y-axis');  
grid on;
```

# 그래프 그리기 Tip - axis

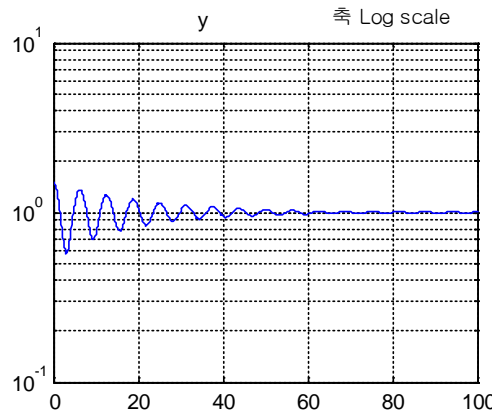
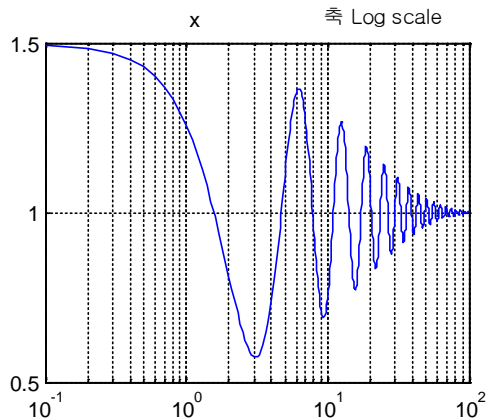
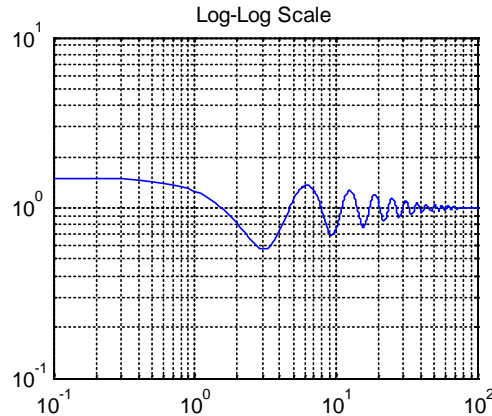
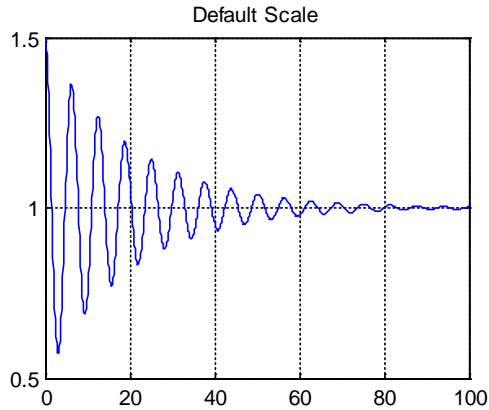


## ■ 축의 한계값의 설정

```
plot(t,f4); % 그래프를 먼저 그린다.  
grid on;  
% 데이터에 맞게 자동으로 축이 조정  
% x축은 0에서 10까지 y축은 0에서 0.5까지  
v=[0,10,0,0.5];  
axis(v) % 축의 설정
```

# 그래프 그리기 Tip – scale plot

- 주로 Frequency Response를 해석할 때 쓰이는 Bode plot을 할 때 쓰인다.



```
t=0:0.1:100;
f=1+0.5*exp(-0.05*t).*sin(t+ pi/2);
```

```
subplot(221)
plot(t,f);
v=[0,100,0.5,1.5];
axis(v)
title('Default Scale');
grid on;
```

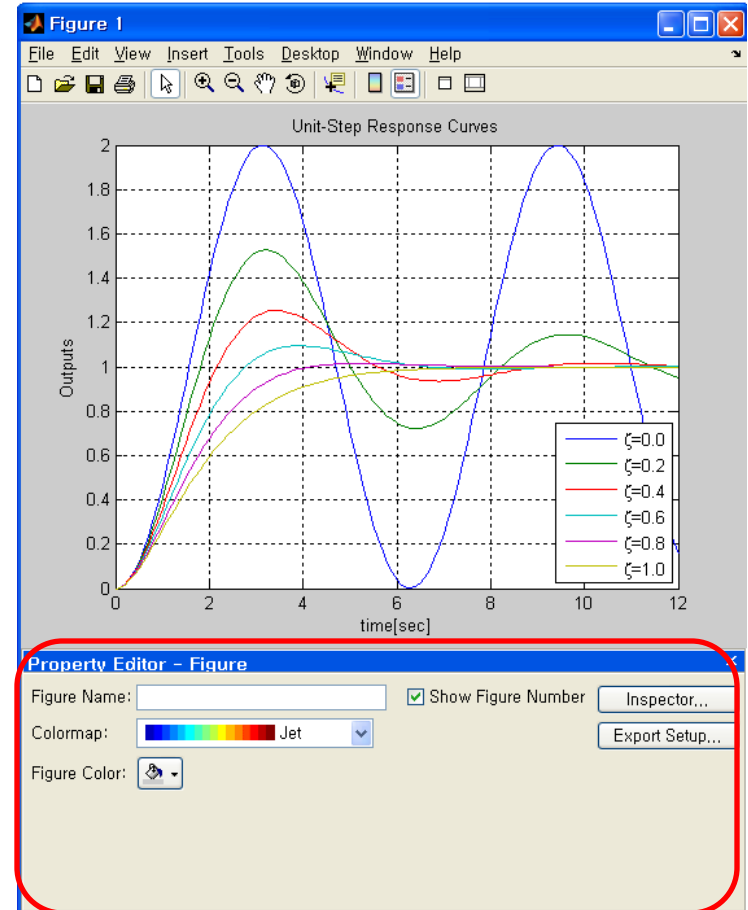
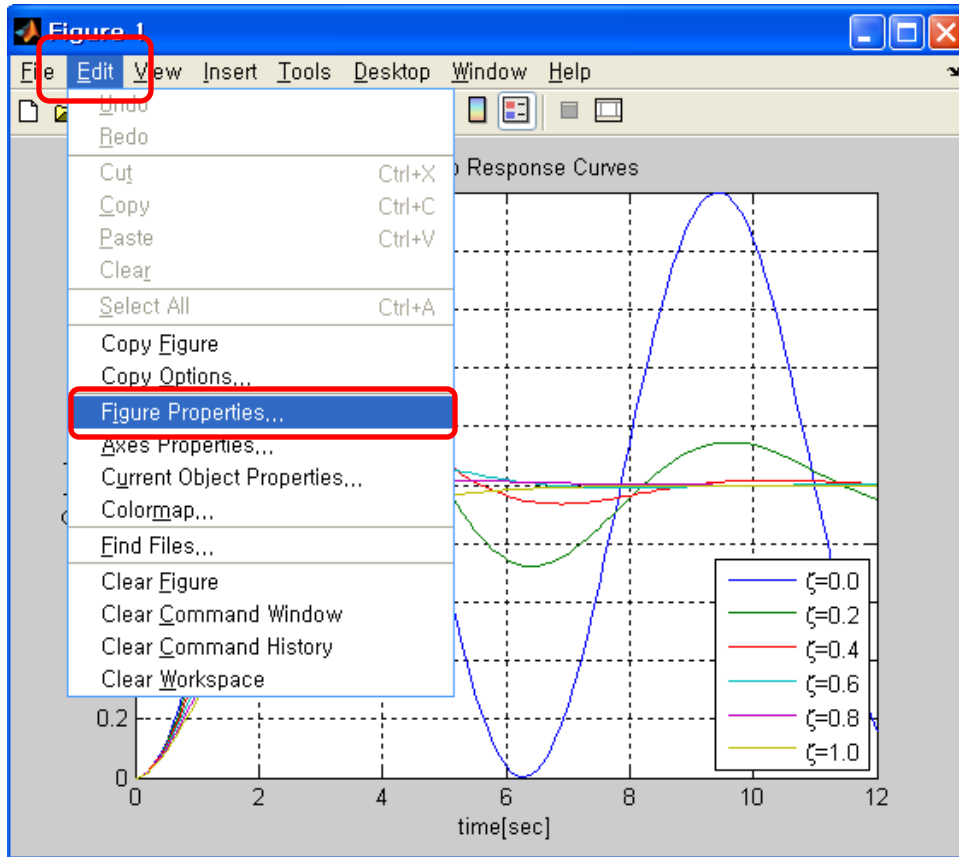
```
subplot(222)
loglog(t,f); % x축과 y축 모두 Log Scale
v=[0,100,10^(-1),10^1];
axis(v)
title('Log-Log Scale');
grid on;
```

```
subplot(223)
semilogx(t,f); % x축만 Log Scale
v=[0,100,0.5,1.5];
axis(v)
title('x축 Log scale');
grid on;
```

```
subplot(224)
semilogy(t,f); % y축만 Log Scale
v=[0,100,10^(-1),10^1];
axis(v)
title('y축 Log scale');
grid on;
```

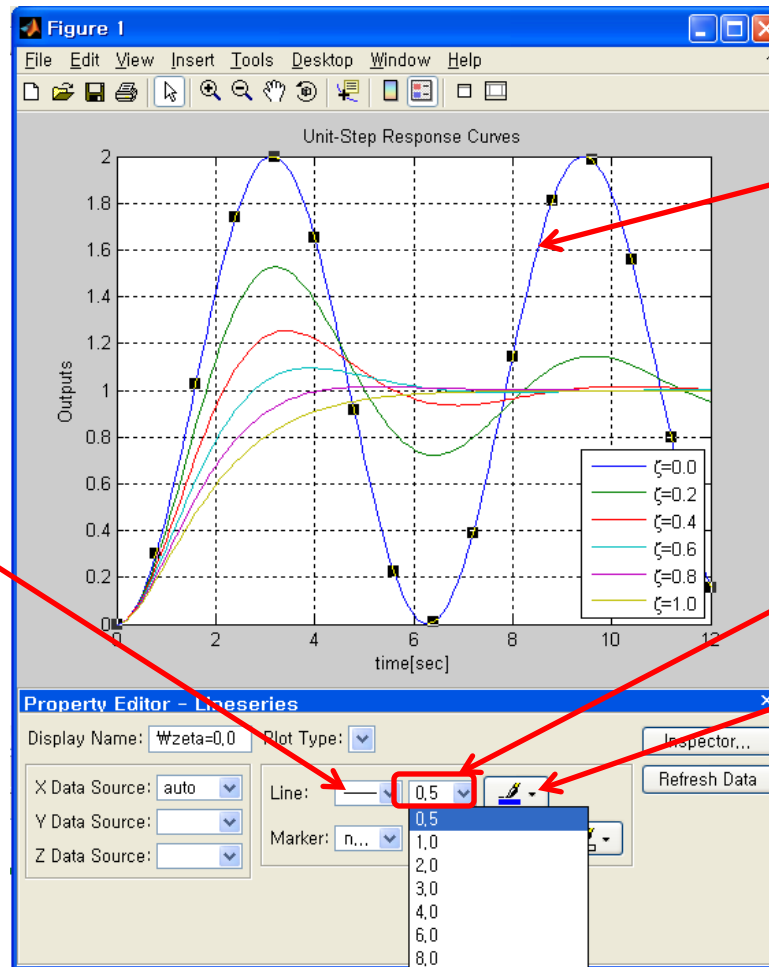
# 그래프 그리기 Tip – figure properties [1/4]

1) edit 탭의 Figure Properties를 클릭하면 아래에 창이 하나 생긴다.



# 그래프 그리기 Tip – figure properties [2/4]

## 2) 선의 굵기 및 종류, 색 수정 (프린트 할 경우 필수)



1) 선을 하나 클릭하여 활성화

2) 굵기는 2정도로 한다

이 버튼으로 선 색 수정 가능

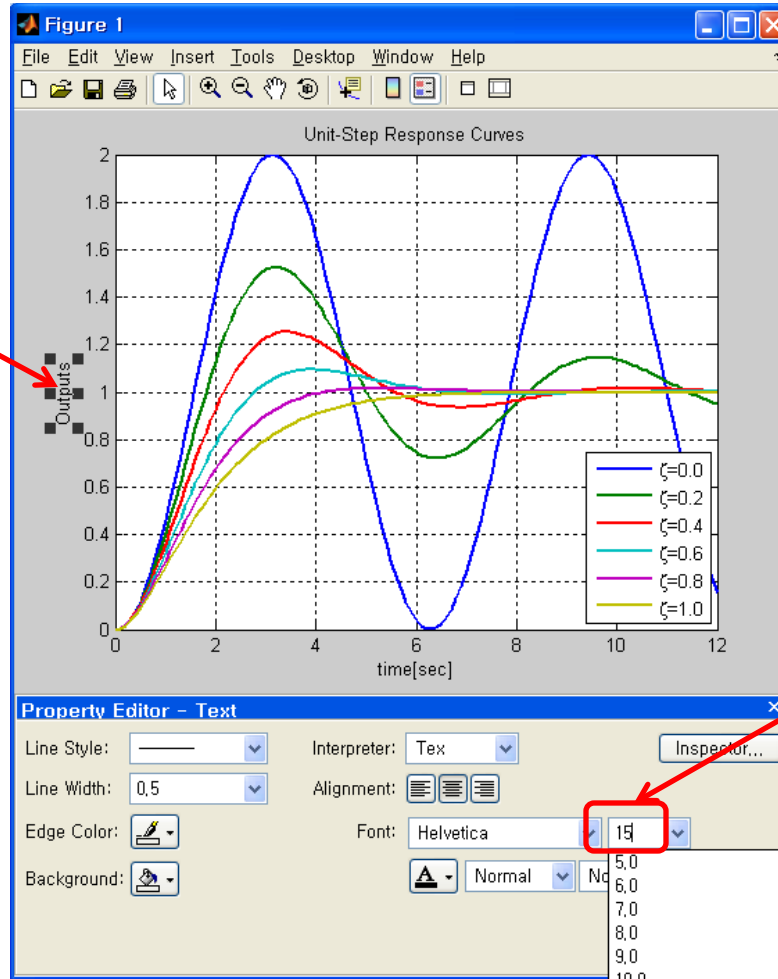
이 버튼으로 선 종류 수정 가능  
(흑백 인쇄 시 반드시 필요  
즉, 6가지 경우에 대하여 선 종류 및 굵기를 다르게 하여 각각 무엇을 나타내는 것인지를 명확히 하여야 한다)



# 그래프 그리기 Tip – figure properties [3/4]

## 3) 폰트 크기 증가 (프린트 할 경우 필수)

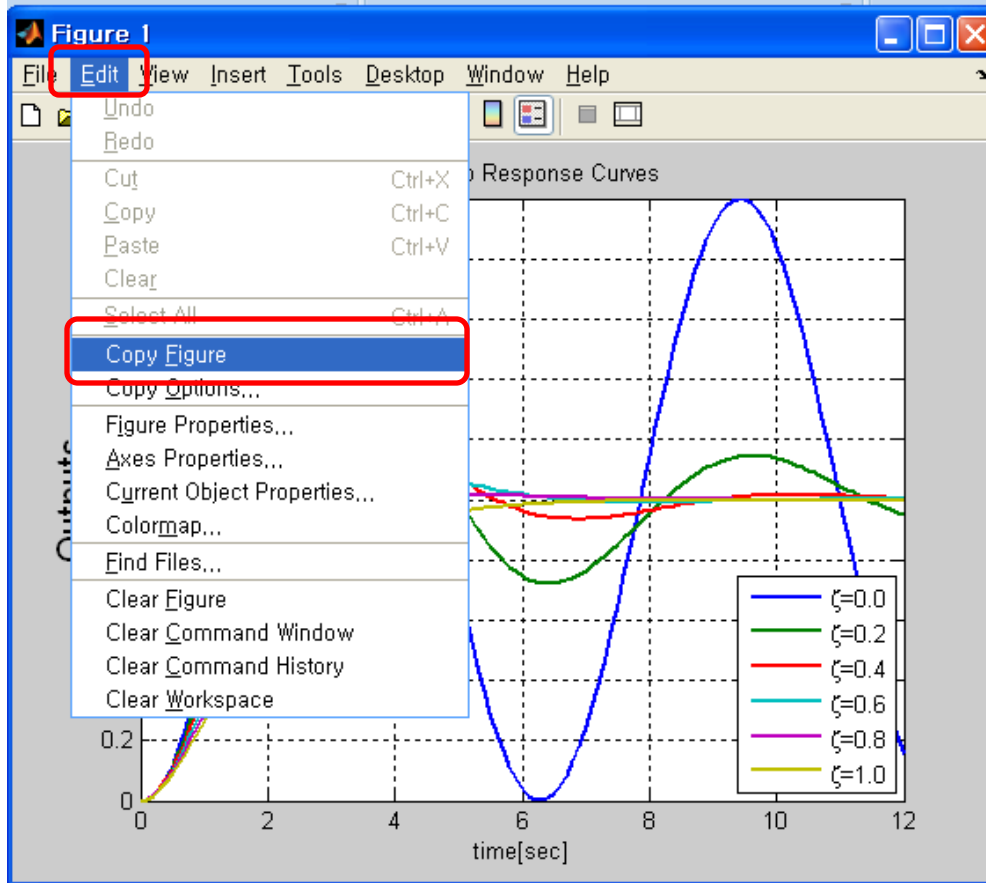
1) 크게 만들고자 하는 텍스트를 클릭하여 활성화



2) 굵기는 15 정도로 한다

# 그래프 그리기 Tip – figure properties [4/4]

## 4) Figure 복사



Edit 탭의 copy figure 옵션을 클릭하면 Figure가 메모리에 저장된다.

이제 그냥 워드나 한글에 붙여 넣기 하면 된다.

이 때 figure의 크기를 가능하면 복사하기 전에 수정하는 것이 좋다.

워드나 한글에 붙여 넣기 한 다음 수정할 경우 글자크기가 좌우로 벌어지거나 폭이 좁아지는 경우가 발생할 수도 있다.

# MATLAB Hotkey Tip

---

1) 주석 처리 단축명령어 : **ctrl+r** (명령문을 주석처리 하면 실행되지 않는다)

2) 주석 해제 단축명령어 : **ctrl+t**

3) M파일에서 **Drag**한 부분만 실행하고자 할 경우 : **F9** 버튼

혹은 마우스 오른쪽버튼 누르고

**Evaluate selection** 탭 클릭