

# 조교 실습 1

250911

# MATLAB Introduction

- **MATLAB: MATrix LABortory**
- MathWorks사에서 출시된 공학, 시뮬레이션 전문 개발 소프트웨어
- 행렬을 기초로 하여 수학, 공학 계산을 수행하는 일종의 언어
- 코드 문법은 C/C++와 유사
- UNIX, MAC 운영체제에서도 사용가능

## MATLAB 응용 분야:



### 제어 시스템

제어 시스템의 설계, 테스트, 구현



### 딥러닝

심층 신경망에 사용할 수 있는 데이터 준비, 설계, 시뮬레이션 및 배포



### 영상 처리 및 컴퓨터 비전

알고리즘 개발과 시스템 설계를 위한 영상 및 비디오의 수집, 처리 및 분석



### 머신러닝

모델을 학습시키고 파라미터를 조정하며 생산 시스템 또는 에지 기기에 배포



### 예측 정비

상태 감시 및 예측 정비 소프트웨어 개발 및 배포



### 로봇공학

로봇공학 관련 아이디어 및 개념을 실제 환경에서 원활하게 작동하는 자율



### 신호 처리

신호 및 시계열 데이터 분석. 신호 처리 시스템 모델링, 설계 및 시뮬레이션



### 테스트 및 측정

데이터 수집, 분석, 탐색 및 테스트 자동화



### 무선 통신

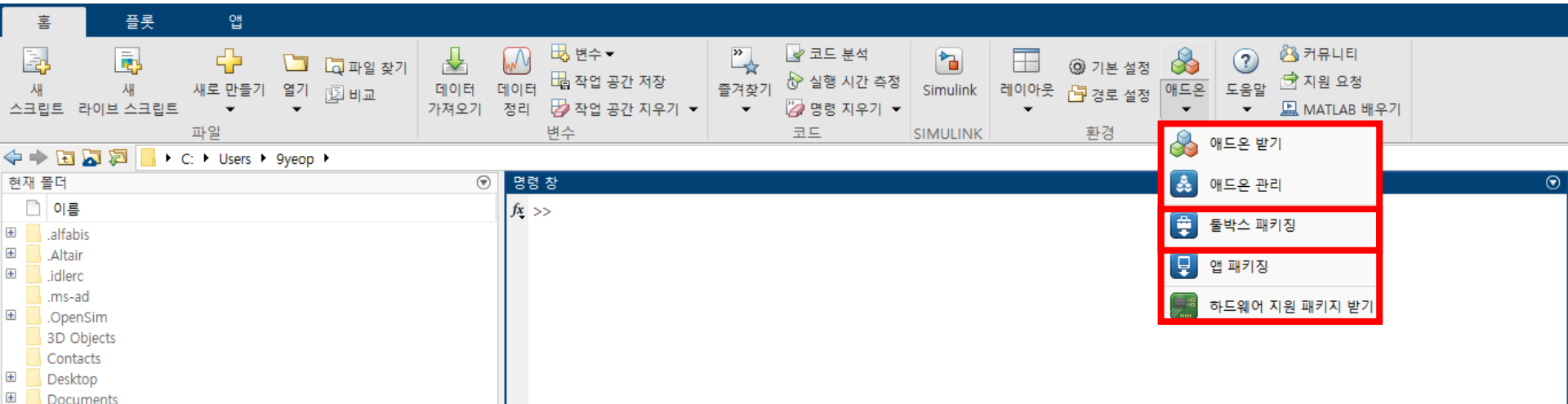
무선 통신 시스템 제작, 설계, 테스트 및 검증

# MATLAB Introduction

## • Toolbox

- 신호처리, 통계학, 제어 등 다양한 전공 분야의 내용을 심도 있게 지원하는 함수 라이브러리
- 모든 toolbox가 필수는 아니고 필요한 toolbox를 그때그때 다운로드

MATLAB R2024b - academic use



# MATLAB Introduction

## Toolbox

- 신호처리, 통계학, 제어 등 다양한 전공 분야의 내용을 심도 있게 지원하는 함수 라이브러리
- 모든 toolbox가 필수는 아니고 필요한 toolbox를 그때그때 다운로드

애드온 탐색기

가이드 | 애드온 관리

애드온 검색

개인 맞춤

내 제품 113  
추천 제품 60

출처별 필터링

MathWorks 428  
커뮤니티 50,352

카테고리별 필터링

Using MATLAB

MATLAB 11,688

Using Simulink

Simulink 841  
Physical Modeling 2,215  
Event-Based Modeling 68  
Real-Time Simulation and Testing 35

Workflows

Parallel Computing 189  
Reporting and Database Access 205  
Systems Engineering 27  
Code Generation 327  
Application Deployment 123  
Verification, Validation, and Test 170  
Cloud Capabilities 68  
Teaching and Learning 3

Applications

AI and Statistics 3,093  
Mathematics and Statistics 2,478

소프트웨어 살펴보기

사용할 수 있는 모든 MathWorks 제품을 확인하십시오.

내 제품 보기

개인 맞춤 추천

60개 모두 표시

MATLAB Schemer

Apply and save color schemes in MATLAB with ease.

라이선스에 포함

Audio Toolbox

Design and analyze speech, acoustic, and audio processing systems

ECG simulation using MATLAB

This code generates all possible forms of ECG signals with the...

라이선스에 포함

Global Optimization Toolbox

Solve multiple maxima, multiple minima, and nonsmooth optimization...

MATLAB Basemap Data - colorterrain

Color terrain basemap data for geographic visualization

GPU Code Interface for Deep Learning

Use GPU Code to generate optimized CUDA code for deep learning...

MathWorks 툴박스와 제품

123개 모두 표시

5G Toolbox

라이선스에 포함

DSP HDL Toolbox

라이선스에 포함

Simulink Control Design

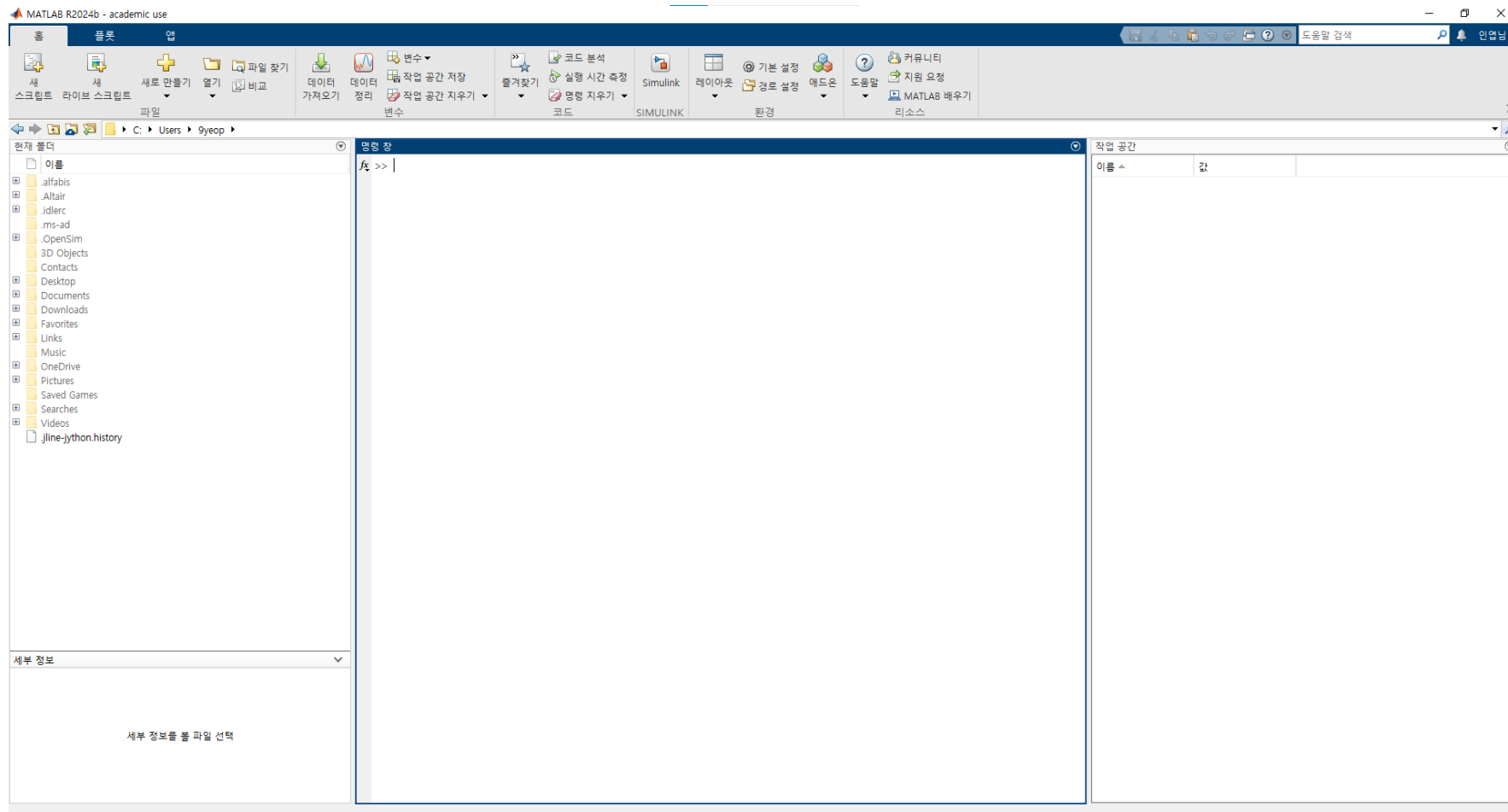
라이선스에 포함

System Composer

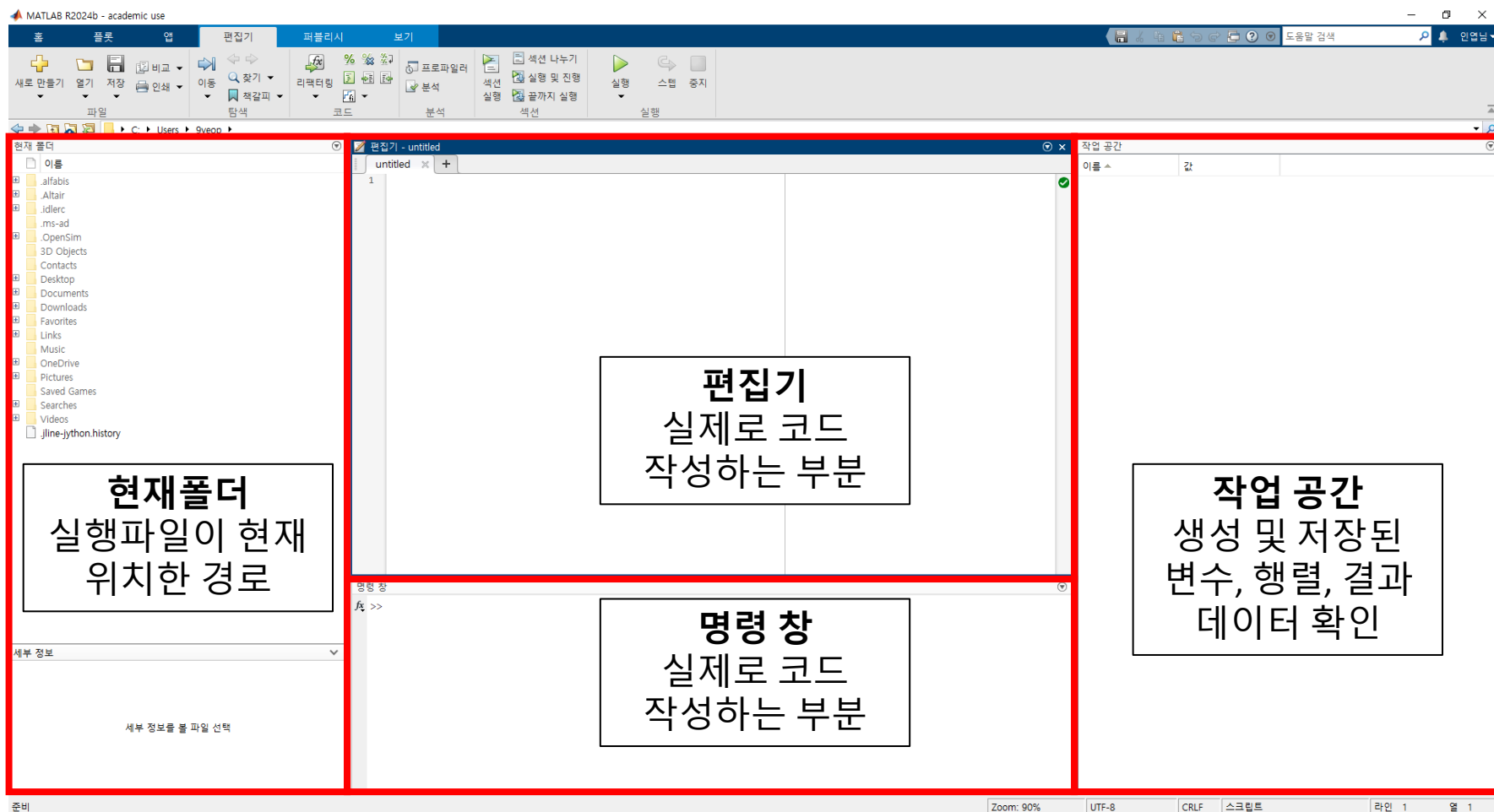
라이선스에 포함

# MATLAB UI

## MATLAB 초기 실행 화면



# MATLAB UI



# MATLAB command

- MATLAB에서 명령어는 왼쪽부터 순차적으로 진행

- Ex1) 명령 창에 'a=1, b=2, c=3' 입력 결과 확인

- 명령어 중간에 ' ,(кома)'를 사용하면 여러가지 명령어 집합을 한 줄에 표현 가능

- 명령어 끝에 '...'을 사용하면 너무 길어서 한 줄에 쓸 수 없는 명령어도 입력 가능

- Ex1) a=1+2+3+...

4+5+6+...

7+8+9+10

- 명령어 끝에 ' ;(세미콜론)'를 사용하면 출력 미표시

- Ex1) a=1;

b=2;

c=a+b;

c 입력 결과 확인

# MATLAB command

•범위지정자 명령어 '**:(콜론)**'를 사용하면 생성 변수의 범위를 지정하거나 행렬의 선택 범위 지정 가능

- $X=n:m$              $n$ 부터  $m$ 까지 단위 간격의 벡터 생성
- $X=n:m:l$          $n$ 부터  $l$ 까지 균일한  $m$  간격의 벡터 생성
- $A(:,m)$            행렬  $A$ 의  $m$ 번째 열 전체
- $A(n,:)$           행렬  $A$ 의  $n$ 번째 행 전체
- $A(:)$               $A$ 의 모든 요소를 단일 열 벡터로 형태 변경

•명령어 줄 제일 앞에 '**%**'를 사용하면 해당 줄은 주석문으로 처리

- Ex1) %학번\_이름        **입력결과확인**
- 주석문은 명령어의 수행에 전혀 영향X
- Ctrl + R: 주석처리, Ctrl + T: 주석처리 제거



# MATLAB command

## •명령어 **clear, clc, close all**

- **clear** : 현재 작업공간에서 모든 변수를 제거
- **clc** : 명령창 표시의 모든 입력값과 출력값을 제거
- **close all** : 핸들이 숨겨지지 않은 모든 Figure를 제거

## •명령어 **tic, toc**

- **tic**과 **toc** 사이에 입력된 코드의 실행시간을 측정

# MATLAB command

## •while

- 반복 조건이 참(true)이면 while 영역에 있는 명령문을 계속 반복 실행시키는 함수

## •for

- 지정된 횟수만큼 for 영역에 있는 명령문을 실행시키는 함수

## •if, elseif, else

- if : 조건이 참(true)인 경우 명령문을 실행시키는 함수
- elseif : if 외의 지정된 조건이 참(true)인 경우 명령문을 실행시키는 함수
- else : if와 elseif의 조건을 만족하지 않는 경우 명령문을 실행시키는 함수

# MATLAB command

## • 연산자 명령어

연산	기호	예시
덧셈	+	6+2
뺄셈	-	6-2
곱셈	*	6*2
오른쪽 나눗셈	/	6/2
왼쪽 나눗셈	\( <del>\w</del> )	2\6(=6/2)
지수연산	^	6^2

우선순위	수학연산
1	괄호 (괄호가 중첩된 경우, 가장 안쪽의 괄호부터 수행)
2	거듭제곱
3	곱하기, 나누기 (우선순위가 동등함, 왼쪽에서 오른쪽 방향으로 수행)
4	더하기, 빼기 (우선순위가 동등함, 왼쪽에서 오른쪽 방향으로 수행)

# MATLAB command

- 수학 내장함수

함수	설명
<code>sqrt(x)</code>	제곱근
<code>exp(x)</code>	지수함수
<code>abs(x)</code>	절대값
<code>log(x)</code>	자연로그 {밑이 e인 로그(ln)}
<code>log10(x)</code>	밑이 10인 로그
<code>factorial(x)</code>	계승함수 $x!$ ( $x$ 는 양의 정수이어야 함)

# MATLAB command

- 삼각함수:  $x$ 의 범위 먼저 지정

함수	설명
sin(x) sind(x)	각도 $x$ 의 sin ( $x$ =radian) 각도 $x$ 의 sin ( $x$ =degree)
cos(x) cosd(x)	각도 $x$ 의 cos ( $x$ =radian) 각도 $x$ 의 cos ( $x$ =degree)
tan(x) tand(x)	각도 $x$ 의 tan ( $x$ =radian) 각도 $x$ 의 tan ( $x$ =degree)
cot(x) cotd(x)	각도 $x$ 의 cot ( $x$ =radian) 각도 $x$ 의 cot ( $x$ =degree)
sec(x) secd(x)	각도 $x$ 의 sec ( $x$ =radian) 각도 $x$ 의 sec ( $x$ =degree)
csc(x) cscd(x)	각도 $x$ 의 csc ( $x$ =radian) 각도 $x$ 의 csc ( $x$ =degree)

# MATLAB Practice1

## • if, elseif, else example

```
nrows = 4;      %nrows를 값 4로 선언
ncols = 6;      %ncols를 값 6으로 선언
A = ones(nrows,ncols); %A를 4 x 6 인 행렬로 선언

for c = 1:ncols    %c가 1부터 ncols까지 1씩 증가하는 동안
    for r = 1:nrows %r가 1부터 nrows까지 1씩 증가하는 동안
        if r == c    %r과 c가 같을 때
            A(r,c) = 2; %r행 c열의 값을 2로 저장
        elseif abs(r-c) == 1 %r과 c를 뺀 절대값이 1일때
            A(r,c) = -1; %r행 c열의 값을 -1로 저장
        else          %if와 else if를 만족하지 않을 때
            A(r,c) = 0; %r행 c열의 값을 0으로 저장
        end          %if문 종료
    end              %for문 종료
end                  %for문 종료
A                    %A의 결과값을 출력
```

```
>> nrows = 4;      %nrows를 값 4로 선언
ncols = 6;      %ncols를 값 6으로 선언
A = ones(nrows,ncols); %A를 4 x 6 인 행렬로 선언

for c = 1:ncols    %c가 1부터 ncols까지 1씩 증가하는 동안
    for r = 1:nrows %r가 1부터 nrows까지 1씩 증가하는 동안
        if r == c    %r과 c가 같을 때
            A(r,c) = 2; %r행 c열의 값을 2로 저장
        elseif abs(r-c) == 1 %r과 c를 뺀 절대값이 1일때
            A(r,c) = -1; %r행 c열의 값을 -1로 저장
        else          %if와 else if를 만족하지 않을 때
            A(r,c) = 0; %r행 c열의 값을 0으로 저장
        end          %if문 종료
    end              %for문 종료
end                  %for문 종료
A                    %A의 결과값을 출력

A =

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

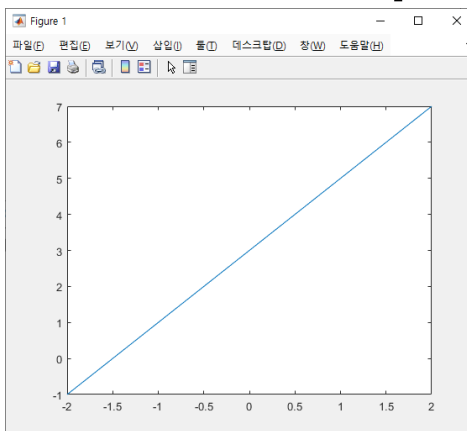
# MATLAB Practice2

## •Plot function

- Figure(): 그래프 그리는 창을 여는 함수
- Plot(x,y): 2차원 그래프 그리는 함수

## •Figure창을 열고 $Y=2*X+3$ 그래프를 plot

```
figure(1)  
X=-2:0.1:2;  
Y=2*X+3;  
plot(X,Y);
```



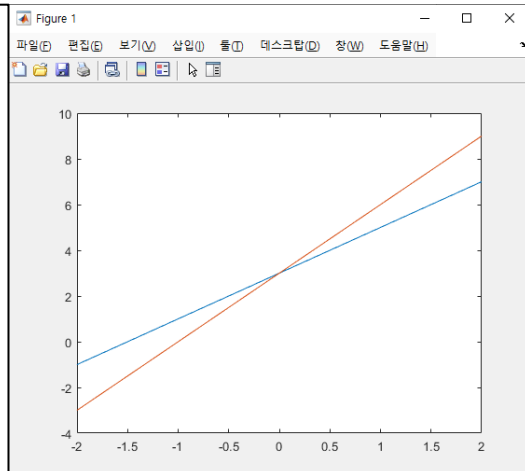
- 변수선언: X의 눈금 간격을 지정하고 싶다면
  - $X = n:m:l$  사용 (n부터 l까지 균일한 m간격의 벡터 생성)  
cf.  $x = [-2 \ -1.9 \ -1.8 \ \dots \ 1.9 \ 2]$  데이터를 벡터형식으로 선언
  - plot(X,Y) X값에 대한 Y데이터의 2차원 선형 플롯을 생성함

# MATLAB Practice3

- 한 figure 창에 여러 개의 그래프를 표시

```
figure(1)
X=-2:0.1:2;
Y=2*X+3;
plot(X,Y);
```

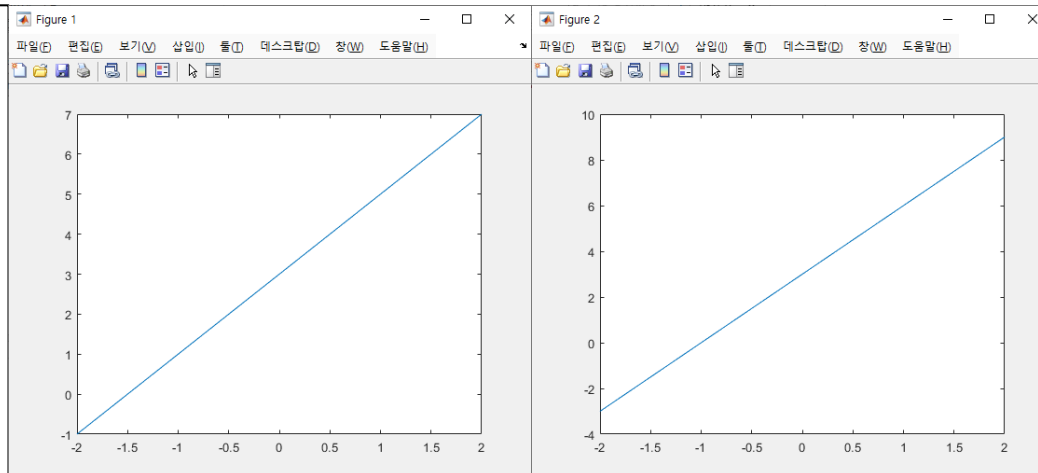
```
hold on
X1 = -2:0.1:2;
Y1 = 3*X1+3;
plot(X1,Y1);
```



- 여러 figure 창을 동시에 표시

```
figure(1)
X=-2:0.1:2;
Y=2*X+3;
plot(X,Y);
```

```
figure(2)
X1 = -2:0.1:2;
Y1 = 3*X1+3;
plot(X1,Y1);
```

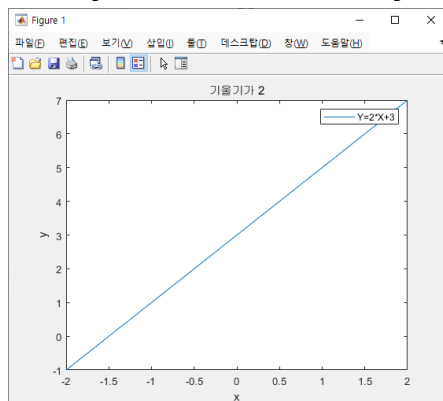




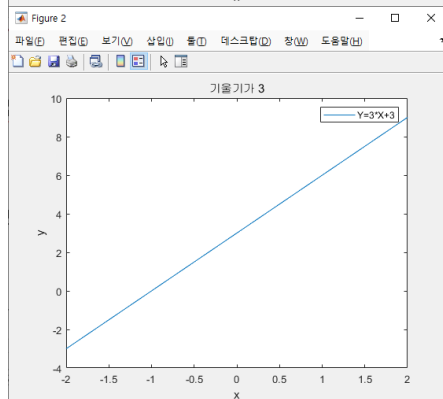
# MATLAB Practice4

- plot의 제목(title), 범례(legend), 축제목(x, y label)
  - Title(' '): 그래프의 제목 설정
  - Legend(' '): 범례 정보 설정
  - xlabel(' '), ylabel(' '): x, y축 정보를 설정

```
figure(1)
X=-2:0.1:2;
Y=2*X+3;
plot(X,Y);
title('기울기가 2')
legend('Y=2*X+3')
xlabel('x')
ylabel('y')
```



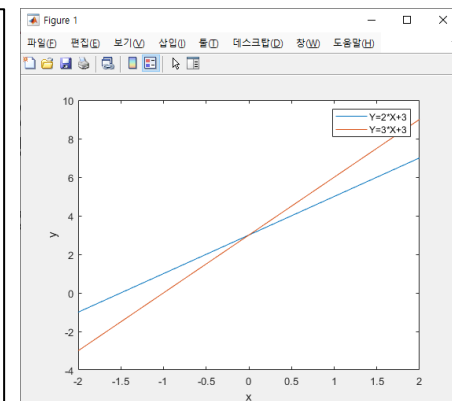
```
figure(2)
X1 = -2:0.1:2;
Y1 = 3*X1+3;
plot(X1,Y1);
title('기울기가 3')
legend('Y=3*X+3')
xlabel('x')
ylabel('y')
```



```
figure(1)
X=-2:0.1:2;
Y=2*X+3;
plot(X,Y);

xlabel('x')
ylabel('y')

hold on
X1 = -2:0.1:2;
Y1 = 3*X1+3;
plot(X1,Y1);
legend('Y=2*X+3','Y=3*X+3')
```



# MATLAB Practice5

- plot 되는 데이터 범위 조정
  - xlim([ ]), ylim([ ]): 축 표시값 제한 설정 가능
  - grid on: 그래프의 격자 표시

```
figure('Name', 'Measured Data')
```

```
X=-2:0.1:2;
```

```
Y=2*X+3;
```

```
plot(X,Y);
```

```
xlabel('x')
```

```
ylabel('y')
```

```
hold on
```

```
X1 = -2:0.1:2;
```

```
Y1 = 3*X1+3;
```

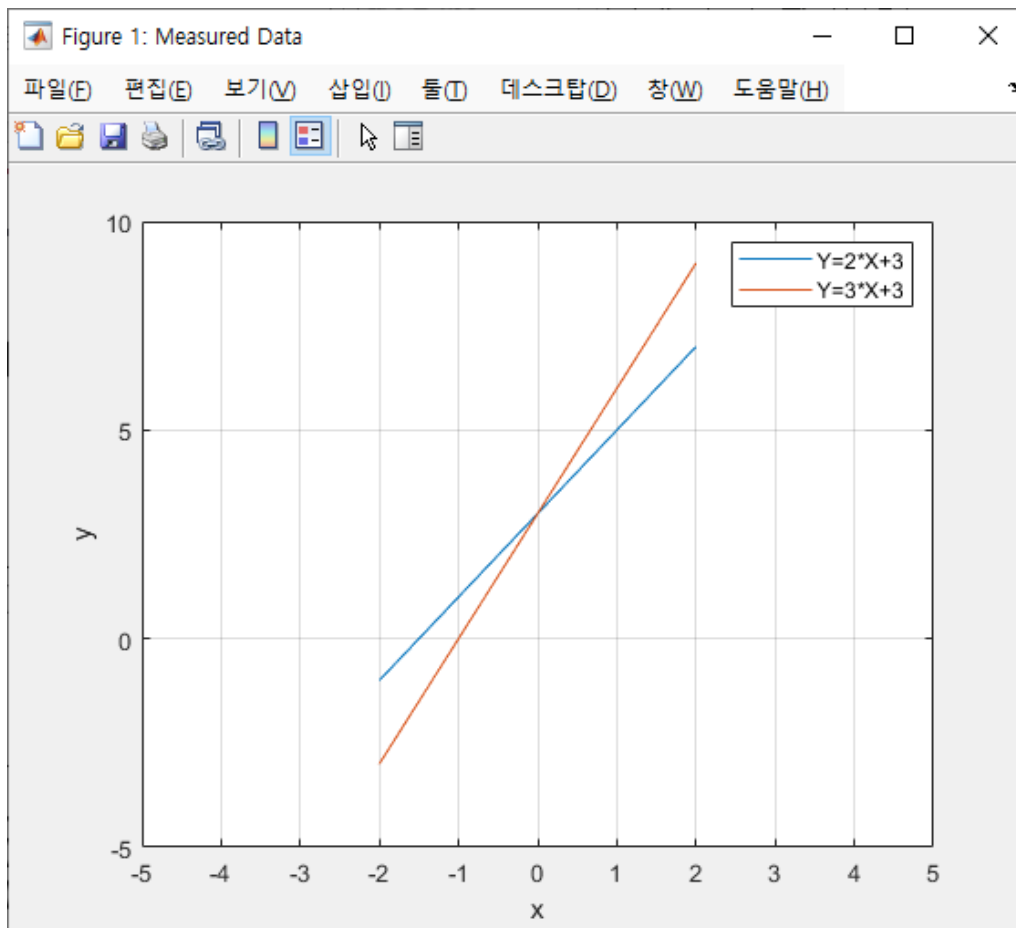
```
plot(X1,Y1);
```

```
legend('Y=2*X+3','Y=3*X+3')
```

```
xlim([-5 5]);
```

```
ylim([-5 10]);
```

```
grid on
```



# MATLAB practice

## How to import data

### 1) Excel -> txt file -> MATLAB

- 그래프를 그리고자 하는 변수  $x$ ,  $y$ 의 데이터 개수를 동일하게 설정
- txt 파일에는 숫자만 포함
- 다른 방법으로는 데이터를 드래그하여 직접 메모장에 복사 후 저장

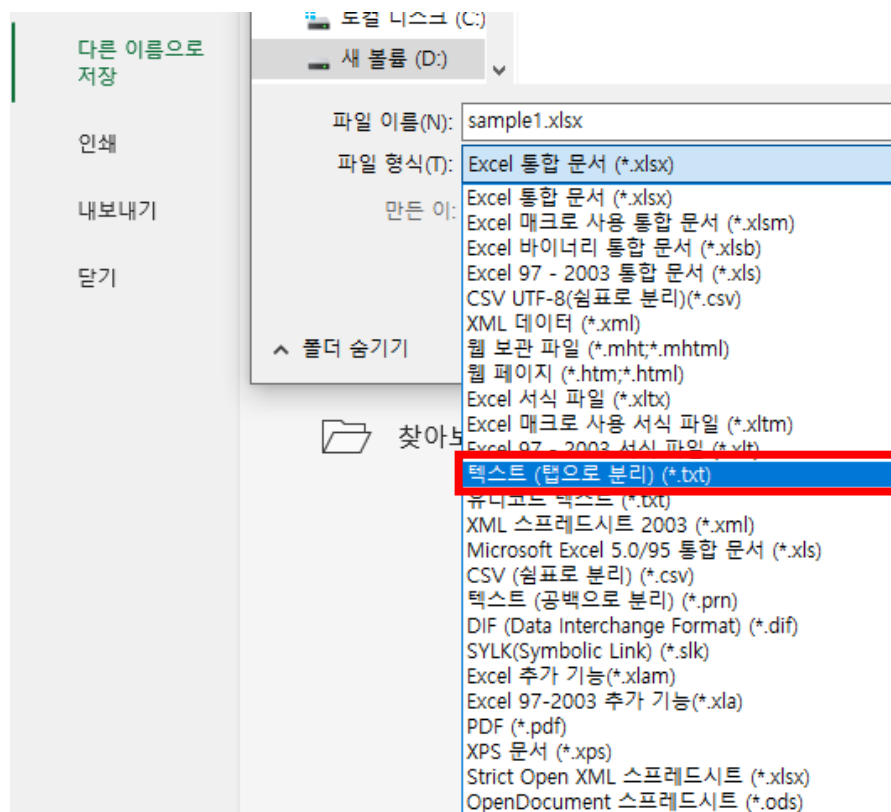
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	
1	Frame	a	b	c	d	e	f	z	h	i	z	i	l	m	n	o	
2		0	0.001317	0.10595	0.873146	-0.00882	0.105522	0.96967	-0.00196	0.106646	1.035623	0.000886	0.106939	1.096295	0.006382	0.107365	1.156715
3		1	0.001336	0.105931	0.873293	-0.00877	0.105506	0.969821	-0.00188	0.106627	1.03577	0.00099	0.106917	1.096441	0.006512	0.107339	1.156859
4		2	0.001448	0.105909	0.873308	-0.00864	0.105482	0.969838	-0.00174	0.1066	1.035786	0.001158	0.106886	1.096456	0.006704	0.107304	1.156871
5		3	0.001535	0.105899	0.873318	-0.00854	0.105474	0.969849	-0.00162	0.106591	1.035796	0.001284	0.106876	1.096465	0.006844	0.107293	1.15688
6		4	0.001669	0.105923	0.873329	-0.00839	0.105495	0.969861	-0.00147	0.106613	1.035807	0.001446	0.106899	1.096476	0.007012	0.107317	1.15689
7		5	0.001798	0.105984	0.873337	-0.00825	0.10556	0.969872	-0.00131	0.106681	1.035816	0.001615	0.106968	1.096485	0.007191	0.107388	1.156898
8		6	0.001926	0.106044	0.873345	-0.0081	0.105626	0.969881	-0.00115	0.106748	1.035824	0.001782	0.107034	1.096492	0.007364	0.10745	1.156905
9		7	0.002126	0.106083	0.873358	-0.00786	0.105693	0.969899	-0.00091	0.106798	1.035842	0.002003	0.107076	1.096511	0.007559	0.107476	1.156926
10		8	0.002303	0.106114	0.873373	-0.00767	0.105731	0.969915	-0.00073	0.106834	1.03586	0.002169	0.107108	1.09653	0.007698	0.107503	1.156947
11		9	0.002458	0.106159	0.873394	-0.0075	0.105783	0.969937	-0.00059	0.106887	1.035885	0.002271	0.107161	1.096556	0.00775	0.107555	1.156978
12		10	0.002589	0.10622	0.873413	-0.00733	0.10585	0.969961	-0.00044	0.106955	1.03591	0.002378	0.107226	1.096583	0.0078	0.107615	1.157011

# MATLAB practice

## How to import data

### 1) Excel -> text file -> MATLAB

- Excel 데이터를 다른 이름으로 저장할 때 [텍스트 (탭으로 분리)] 파일 형식으로 저장

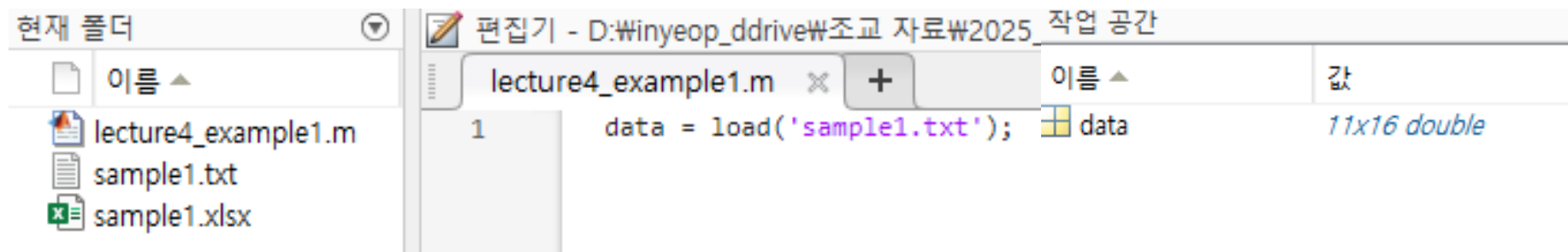


# MATLAB practice

## How to import data

### 1) Excel -> text file -> MATLAB

- 불러올 파일이 있는 경로 지정 (경로 복사 또는 현재 경로 상태에서 현재 폴더에 파일을 드래그앤드롭)
- 반드시 불러올 파일이 현재 폴더에 포함되어 있어야 import 가능
- import 함수: 행렬명 = load('파일명.txt');
- 편집기 스크립트 파일 저장 시 반드시 문자로 시작해야 함



# MATLAB practice

## How to import data

### 1) Excel -> text file -> MATLAB

- 작업공간에서 생성된 행렬 더블클릭 시 import된 데이터 확인 가능
- Data indexing: 변수 이름=행렬 이름(행 데이터, 열 데이터);
- 괄호 안 ':'은 전체 데이터를 의미

The top screenshot shows the MATLAB Editor with a script named 'lecture4\_example1.m'. The script contains the following code:

```
1 data = load('sample1.txt');
2
3 x = data(:,1);
4 y1 = data(:,2);
5 y2 = data(:,3);
6
```

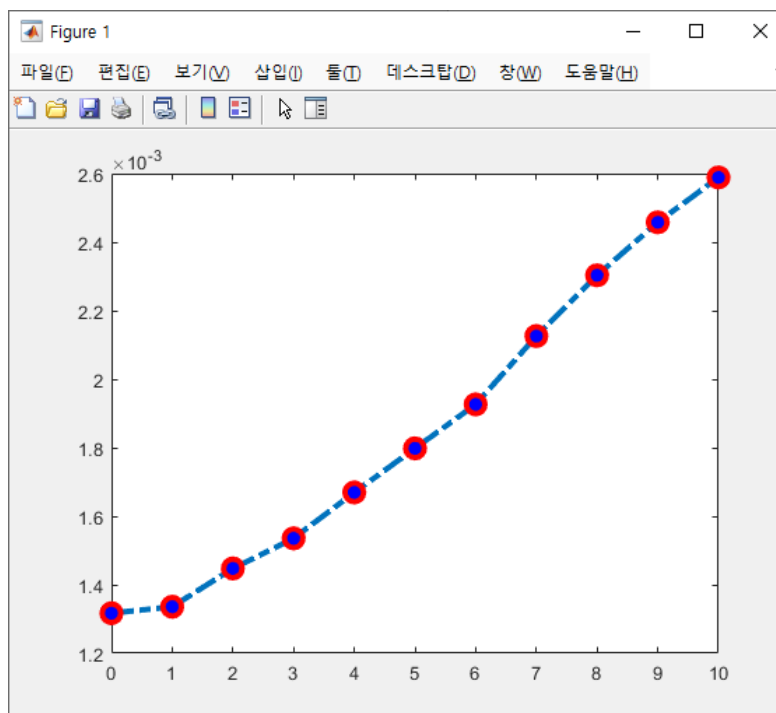
The bottom screenshot shows the MATLAB Workspace. It lists four variables: 'data' (11x16 double), 'x' (11x1 double), 'y1' (11x1 double), and 'y2' (11x1 double). Below the workspace, the 'data' variable is expanded, showing a table with 11 rows and 16 columns. The first column contains values from 0 to 10, and the subsequent columns contain various numerical values.

	1	2	3
1	0		
2	1		
3	2		
4	3		
5	4		
6	5		
7	6		
8	7		
9	8		
10	9		
11	10		

# MATLAB practice

## Plot setting

```
1 data = load('sample1.txt');  
2  
3 x = data(:,1);  
4 y1 = data(:,2);  
5 y2 = data(:,3);  
6  
7 plot(x,y1, '-.o' , 'LineWidth' , 3 , 'MarkerSize' , 10 , 'MarkerFaceColor' , 'b' , 'MarkerEdgeColor' , 'r')
```



# MATLAB practice

## Plot setting






선 스타일	설명	결과 선
"_"	실선	_____
"_ _"	파선	- - - - -
": "	점선	.....
"- ."	일점 쇄선	- . - . - .

마커	설명	결과로 생성되는 마커
"o"	원	○
"+"	플러스 기호	+
"*"	별표	✱
","	점	•
"x"	십자	×
"_"	가로선	—
" "	세로선	
"square"	정사각형	□
"diamond"	다이아몬드	◇
"^"	위쪽 방향 삼각형	△
"v"	아래쪽 방향 삼각형	▽
">"	오른쪽 방향 삼각형	▷
"<"	왼쪽 방향 삼각형	◁
"pentagram"	펜타그램	☆
"hexagram"	헥사그램	☆



# MATLAB practice

## Plot setting

색 이름	짧은 이름	RGB 3색	모양
"red"	"r"	[1 0 0]	
"green"	"g"	[0 1 0]	
"blue"	"b"	[0 0 1]	
"cyan"	"c"	[0 1 1]	
"magenta"	"m"	[1 0 1]	
"yellow"	"y"	[1 1 0]	
"black"	"k"	[0 0 0]	
"white"	"w"	[1 1 1]	

# MATLAB practice

## Help

### - 함수 설명 및 파라미터 사용법에 도움

```
명령 창
>> help plot
plot - 2차원 선 플롯
    x 값에 대한 y 데이터의 2차원 선 플롯을 생성합니다.

벡터 및 행렬 데이터
    plot(X,Y)
    plot(X,Y,LineSpec)
    plot(X1,Y1,...,Xn,Yn)
    plot(X1,Y1,LineSpec1,...,Xn,Yn,LineSpecn)
    plot(Y)
    plot(Y,LineSpec)

테이블 데이터
    plot(tbl,xvar,yvar)
    plot(tbl,yvar)

추가 옵션
    plot(ax,___)
    plot(___,Name,Value)
    p = plot(___)

입력 인수
    X - x 좌표
        스칼라 | 벡터 | 행렬
    Y - y 좌표
        스칼라 | 벡터 | 행렬
    LineSpec - 선 스타일, 마커, 색
        string형 스칼라 | 문자형 벡터
    tbl - 소스 테이블
        테이블 | 타임테이블
```

# MATLAB practice

## In class quiz #1

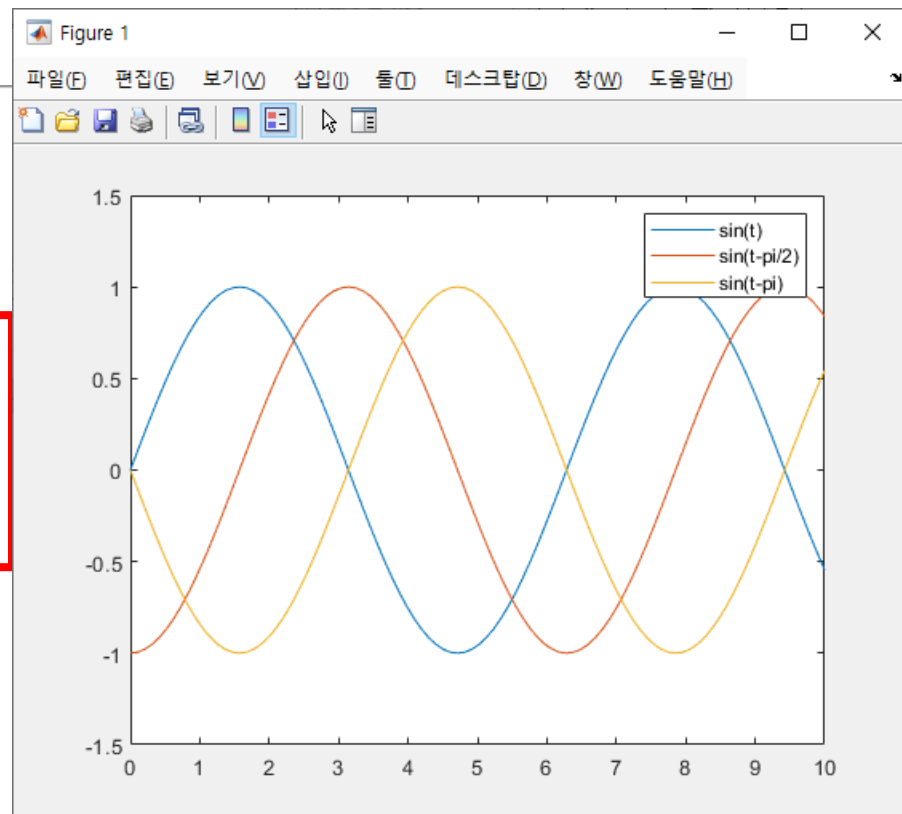
```

1  clear
2  clc
3  close all
4
5  %%%% plot practice %%%%
6  t = 0 : pi / 30 : 4 * pi;
7
8
9
10
11
12
13
14  legend('sin(t)' , 'sin(t-pi/2)' , 'sin(t-pi)')

```

?

(Hint: plot , xlim , ylim)



# Q&A

# Thank you