동양미래대학교

데이터분석입문

Lecture 05. matplotlib 라이브러리로 그래프 그리기

동양미래대학교
인공지능소프트웨어학과
강 환수

목차

Copyright 2023. Dongyang Mirae University all rights reserved.

01. matplotlib 라이브러리 소개

02. plot 함수로 선 그래프 그리기
03. hist 함수로 히스토그램 그리기
04. boxplot 함수로 상자 그림 그리기

Copyright 2023. Dongyang Mirae University all rights reserved.

❖ matplotlib 라이브러리란? (1/7)

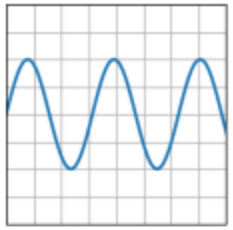
- 파이썬으로 데이터를 시각화할 때 가장 많이 사용하는 라이브러리 중 하나
- 2D 형태의 그래프, 이미지 등을 그릴 때 사용
- 실제 과학 컴퓨팅 연구 분야나 인공지능 연구 분야에서도 많이 활용

Copyright 2023. Dongyang Mirae University all rights reserved.

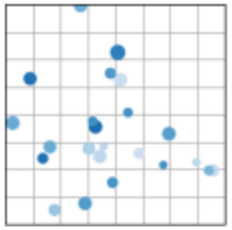
❖ matplotlib 라이브러리란? (2/7)

- 어떤 그래프들을 그릴 수 있나요?

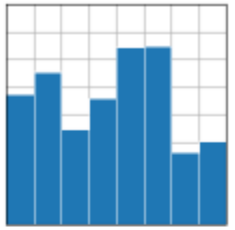
Basic Plot Types



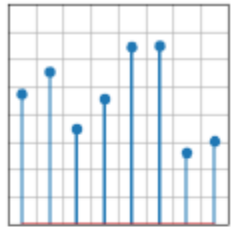
plot(x, y)



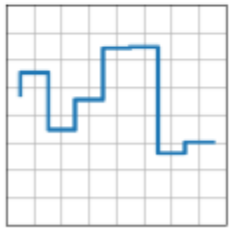
scatter(x, y)



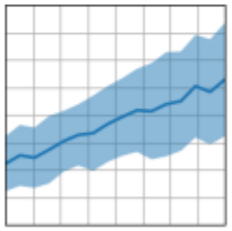
bar(x, height) / barh(y, width)



stem(x, y)



step(x, y)



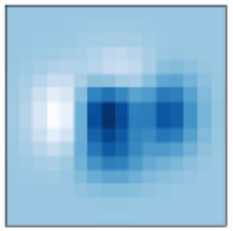
fill_between(x, y1, y2)

Copyright 2023. Dongyang Mirae University all rights reserved.

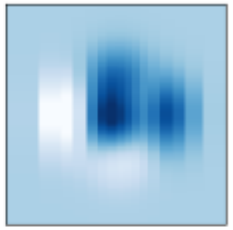
❖ matplotlib 라이브러리란? (3/7)

- 어떤 그래프들을 그릴 수 있나요?

Plots of Arrays and Fields



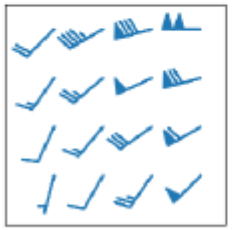
imshow(Z)



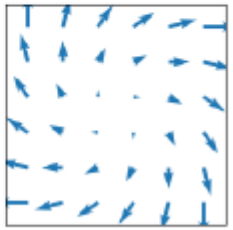
pcolormesh(X, Y, Z)

contour(X, Y, Z)

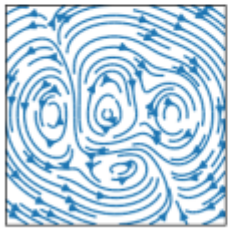
contourf(X, Y, Z)



barbs(X, Y, U, V)



quiver(X, Y, U, V)



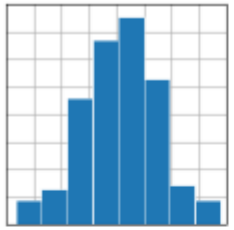
streamplot(X, Y, U, V)

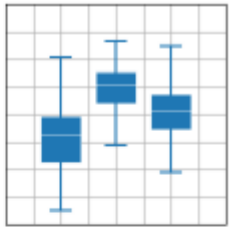
Copyright 2023. Dongyang Mirae University all rights reserved.

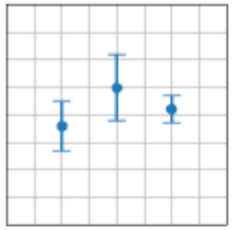
❖ matplotlib 라이브러리란? (4/7)

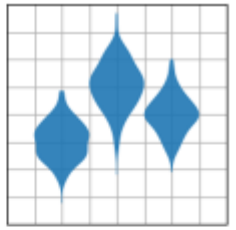
- 어떤 그래프들을 그릴 수 있나요?

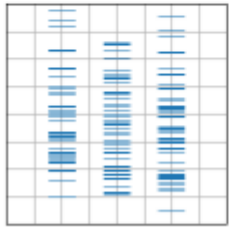
Statistics Plots

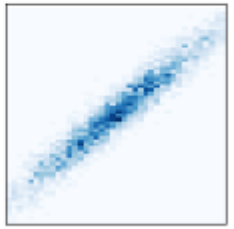

hist(x)

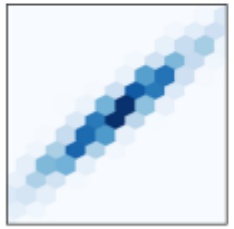

boxplot(X)


errorbar(x, y, yerr, xerr)


violinplot(D)


eventplot(D)


hist2d(x, y)


hexbin(x, y, C)

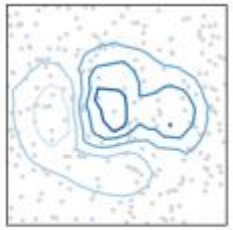

pie(x)

Copyright 2023. Dongyang Mirae University all rights reserved.

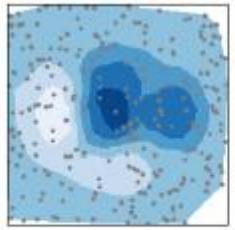
❖ matplotlib 라이브러리란? (5/7)

- 어떤 그래프들을 그릴 수 있나요?

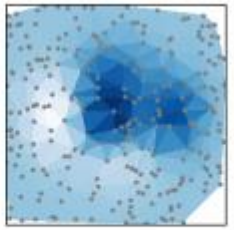
Unstructured Coordinates



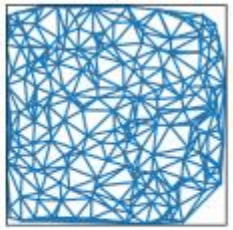
tricontour(x, y, z)



tricontourf(x, y, z)



tripcolor(x, y, z)

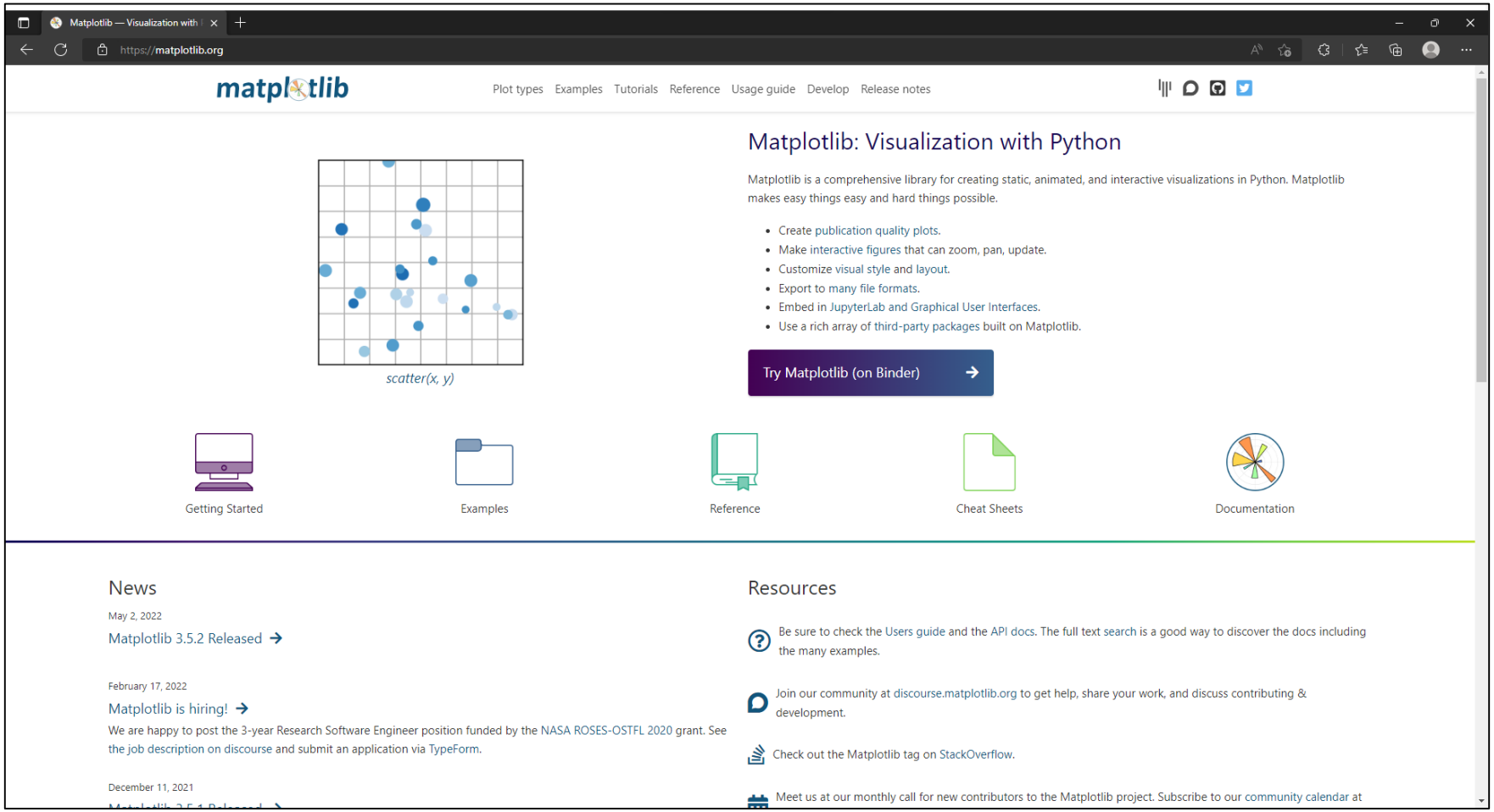


triplot(x, y)

Copyright 2023. Dongyang Mirae University all rights reserved.

❖ matplotlib 라이브러리란? (6/7)

- matplotlib 홈페이지: https://matplotlib.org/



Copyright 2023. Dongyang Mirae University all rights reserved.

❖ matplotlib 라이브러리란? (6/7)

- Matplotlib 갤러리 웹사이트
- https://matplotlib.org/2.0.2/gallery.html

Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 2D 그래프를 위한 데스크탑 패키지

- 파이썬에서 자료를 차트(chart)나 플롯(plot)으로 시각화(visualization)하는 패키지
- 정형화된 차트나 플롯 이외에도 저수준 API를 사용한 다양한 시각화 기능을 제공
  - ◆ 라인 플롯(line plot)
  - ◆ 스캐터 플롯(scatter plot)
  - ◆ 컨투어 플롯(contour plot)
  - ◆ 서피스 플롯(surface plot)
  - ◆ 바 차트(bar chart)
  - ◆ 히스토그램(histogram)
  - ◆ 박스 플롯(box plot)

❖ 2002년 존 헌터가 시작

Copyright 2023. Dongyang Mirae University all rights reserved.

❖ matplotlib 라이브러리란? (7/7)

- matplotlib의 pyplot 모듈 불어오기

```
import matplotlib.pyplot
```

```
from matplotlib import pyplot
```

- matplotlib 라이브러리의 pyplot 모듈을 'plt'라는 별명으로 부르기 (alias)

```
import matplotlib.pyplot as plt
```

```
from matplotlib import pyplot as plt
```


Copyright 2023. Dongyang Mirae University all rights reserved.

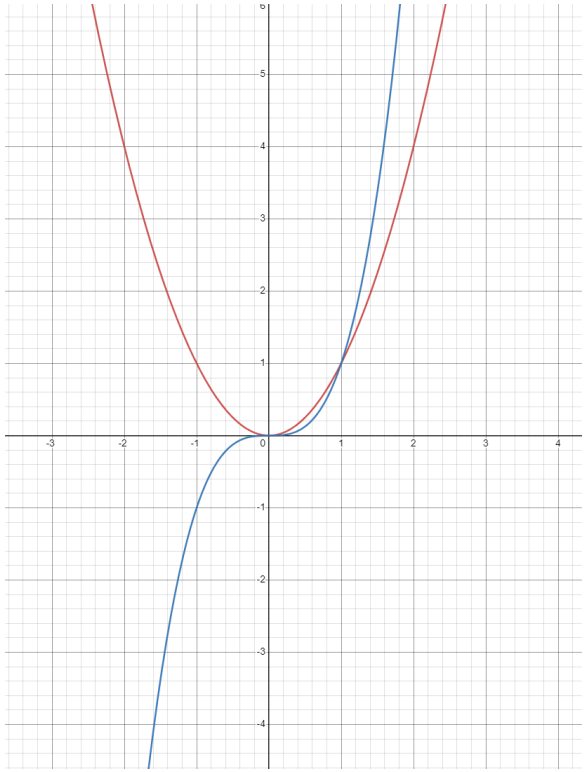
02. plot 함수로 선 그래프 그리기

01. matplotlib 라이브러리 소개
03. hist 함수로 히스토그램 그리기
04. boxplot 함수로 상자 그림 그리기

Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 선 그래프 그리기: plt.plot()

- 가장 간단한 플롯은 선을 그리는 라인 플롯(line plot)
- 라인 플롯은 데이터가 시간, 순서 등에 따라 어떻게 변화하는지 보여주기 위해 사용
- http://matplotlib.org/api/pyplot_api.html#Matplotlib.pyplot.plot

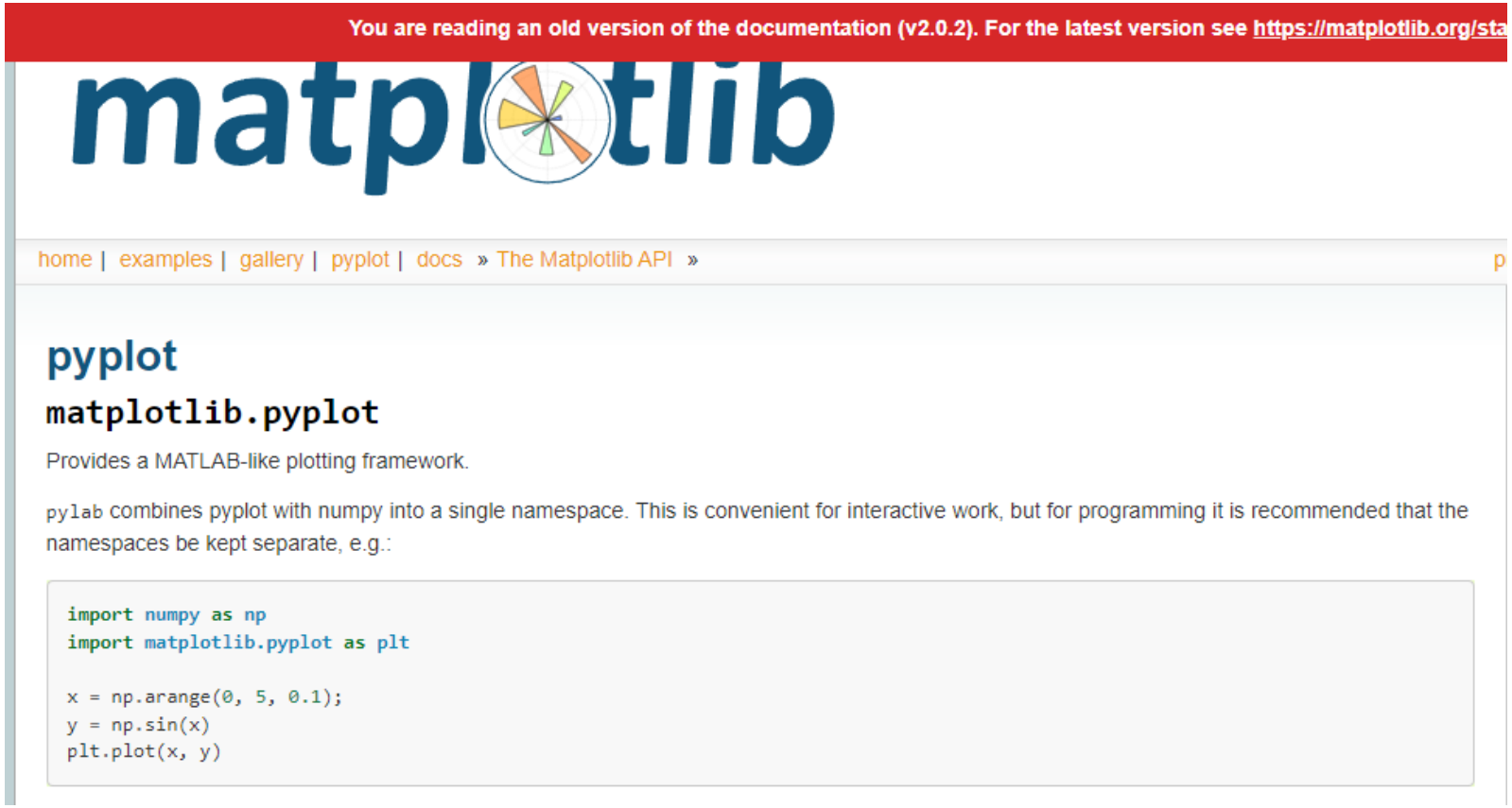


Copyright 2023. Dongyang Mirae University all rights reserved.

02. plot 함수로 선 그래프 그리기

❖ 선 그래프 그리기: plt.plot()

- http://matplotlib.org/api/pyplot_api.html#Matplotlib.pyplot.plot



Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 설치 점검

설치 점검 및 모듈 불러오기

In [2]:
```
import site
site.getsitepackages()
```

Out[2]:
```
['C:₩₩Users₩₩PC₩₩AppData₩₩Local₩₩Programs₩₩Python₩₩Python311',
 'C:₩₩Users₩₩PC₩₩AppData₩₩Local₩₩Programs₩₩Python₩₩Python311₩₩Lib₩₩site-packages']
```

In [4]:
```
pip show matplotlib
```

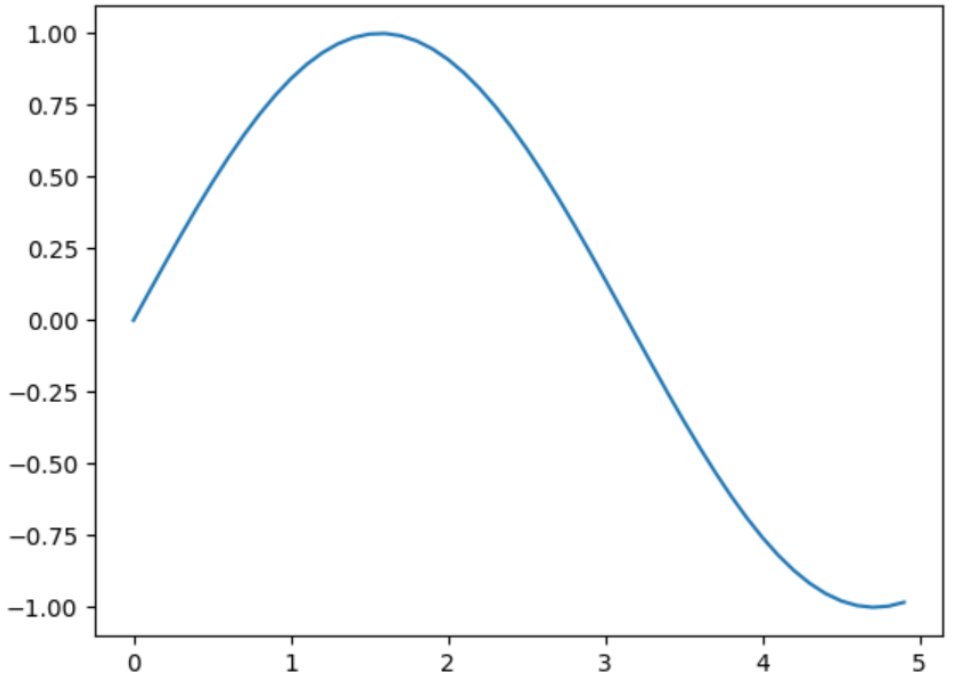
```
Name: matplotlib
Version: 3.7.2
Summary: Python plotting package
Home-page: https://matplotlib.org
Author: John D. Hunter, Michael Droettboom
Author-email: matplotlib-users@python.org
License: PSF
Location: C:₩Users₩PC₩AppData₩Local₩Programs₩Python₩Python311₩Lib₩site-packages
Requires: contourpy, cycler, fonttools, kiwisolver, numpy, packaging, pillow, pypa
rsing, python-dateutil
Required-by:
Note: you may need to restart the kernel to use updated packages.
```


Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 선 그래프 그리기: plt.plot()

- http://matplotlib.org/api/pyplot_api.html#Matplotlib.pyplot.plot

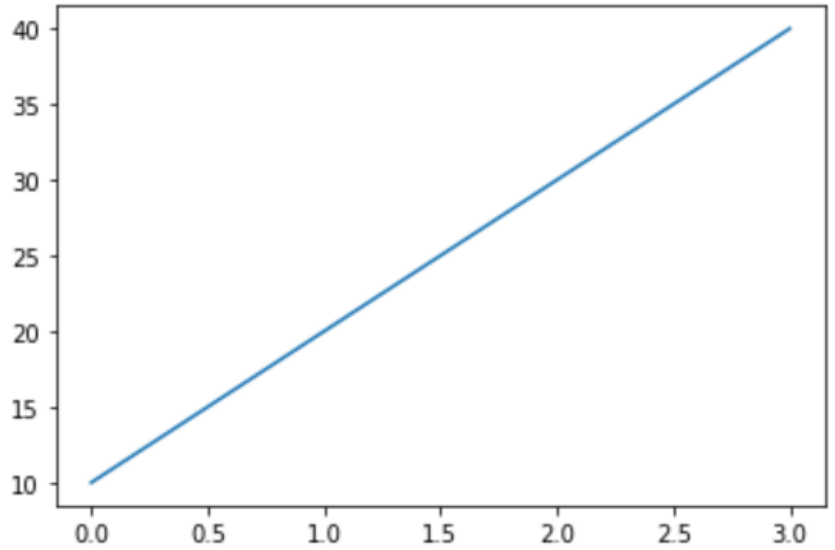
In [6]:
```
import numpy as np
import matplotlib.pyplot as plt

x = np.arange(0, 5, 0.1);
y = np.sin(x)
plt.plot(x, y)
plt.show()
```



Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 선 그래프 그리기: ① plt.plot([y축 데이터])

```
import matplotlib.pyplot as plt

plt.plot([10, 20, 30, 40])
plt.show()
```



현재 plt.plot( )에 입력된 리스트 값은 y축 값입니다.

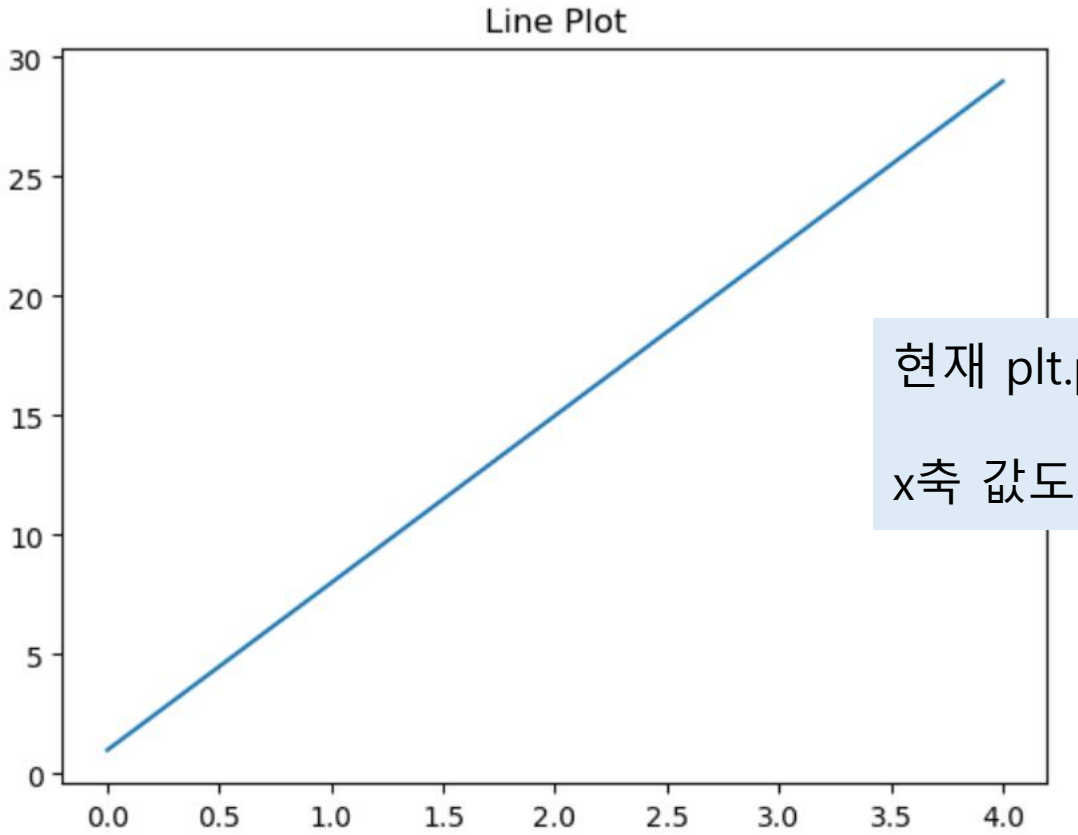
x축 값도 입력하고 싶다면 어떻게 해야 할까요?

Copyright 2023. Dongyang Mirae University all rights reserved.

02. plot 함수로 선 그래프 그리기

❖ 선 그래프 그리기: ① plt.plot([y축 데이터])

```
In [7]: import matplotlib.pyplot as plt

plt.title('Line Plot')
plt.plot(range(1, 30, 7))
plt.show()
```



현재 plt.plot( )에 입력된 리스트 값은 y축 값입니다.

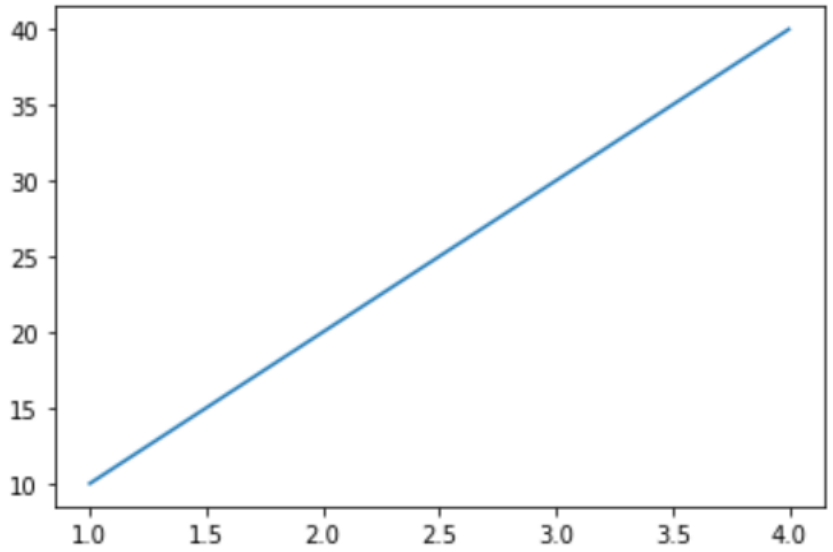
x축 값도 입력하고 싶다면 어떻게 해야 할까요?

Copyright 2023. Dongyang Mirae University all rights reserved.

02. plot 함수로 선 그래프 그리기

❖ 선 그래프 그리기: ② plt.plot([x축 데이터], [y축 데이터])

```
import matplotlib.pyplot as plt

plt.plot([1, 2, 3, 4], [10, 20, 30, 40])
plt.show()
```



첫 번째 리스트 값은 x축

두 번째 리스트 값은 y축을 나타냅니다.

Copyright 2023. Dongyang Mirae University all rights reserved.

02. plot 함수로 선 그래프 그리기

❖ 선 그래프 그리기: ② plt.plot([x축 데이터], [y축 데이터])

```
import matplotlib.pyplot as plt

plt.plot([1, 2, 3, 4], [10, 20, 30, 40, 50])
plt.show()
```

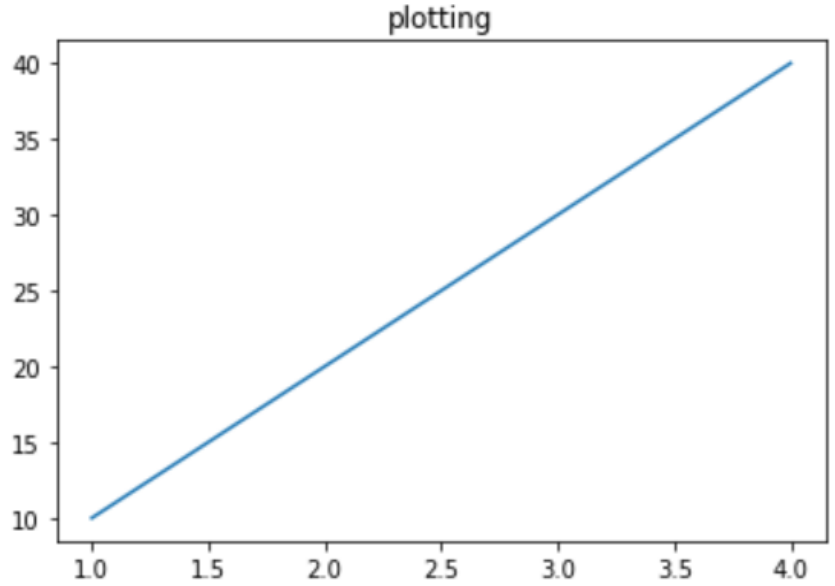
```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
~₩AppData₩Local₩Temp/ipykernel_6736/1955072484.py in <module>
      1 import matplotlib.pyplot as plt
      2
----> 3 plt.plot([1, 2, 3, 4], [10, 20, 30, 40, 50])
      4 plt.show()

~₩anaconda3₩lib₩site-packages₩matplotlib₩pyplot.py in plot(scalex, scaley, data, *args, **kwargs)
   3017 @_copy_docstring_and_deprecators(Axes.plot)
   3018 def plot(*args, scalex=True, scaley=True, data=None, **kwargs):
-> 3019     return gca().plot(
   3020         *args, scalex=scalex, scaley=scaley,
   3021         **({"data": data} if data is not None else {}), **kwargs)
```

x축 데이터와 y축 데이터의 개수가 같지 않으면 오류가 발생합니다.

Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 그래프에 제목(Title) 추가하기

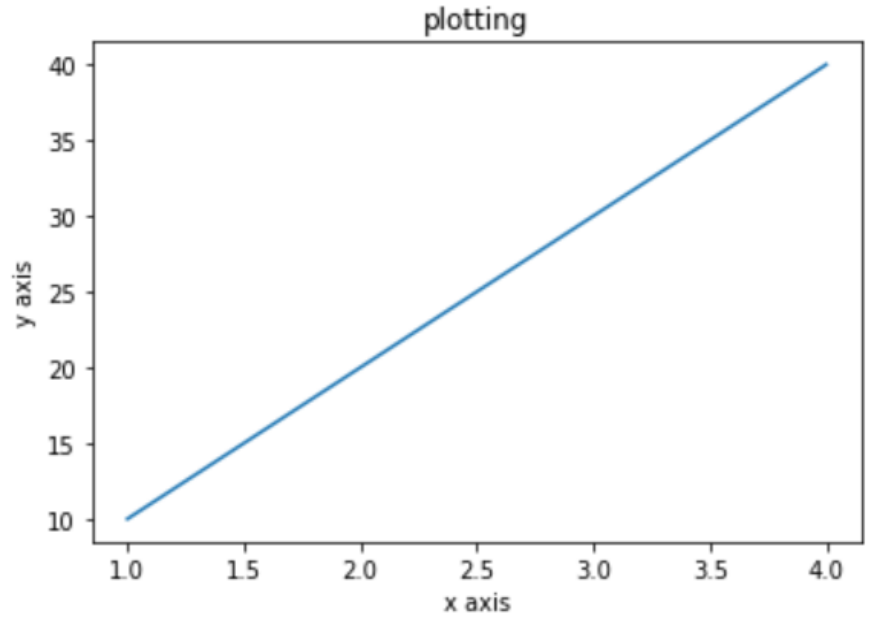
```
import matplotlib.pyplot as plt

plt.plot([1, 2, 3, 4], [10, 20, 30, 40])
plt.title('plotting')
plt.show()
```



Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 그래프에 x축 및 y축 레이블(Label) 정보 추가하기

```
import matplotlib.pyplot as plt

plt.plot([1, 2, 3, 4], [10, 20, 30, 40])
plt.title('plotting')
plt.xlabel('x axis')
plt.ylabel('y axis')
plt.show()
```

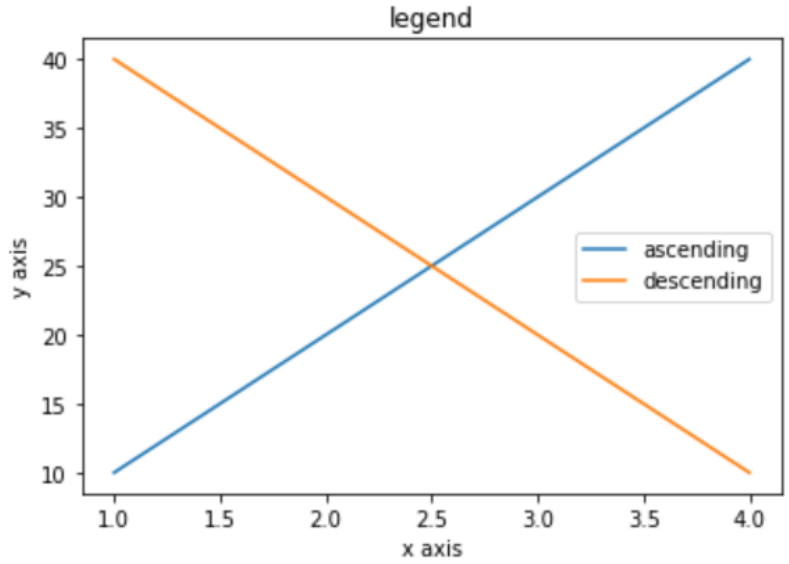


다른 사람이 내가 만든 그래프를 보았을 때,
한 번에 이해할 수 있게
x, y축 레이블 정보를 추가하도록 합시다

Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 그래프에 범례(Legend) 추가하기

```
import matplotlib.pyplot as plt

plt.plot([1, 2, 3, 4], [10, 20, 30, 40], label='ascending')
plt.plot([1, 2, 3, 4], [40, 30, 20, 10], label='descending')
plt.title('legend')
plt.xlabel('x axis')
plt.ylabel('y axis')
plt.legend()
plt.show()
```



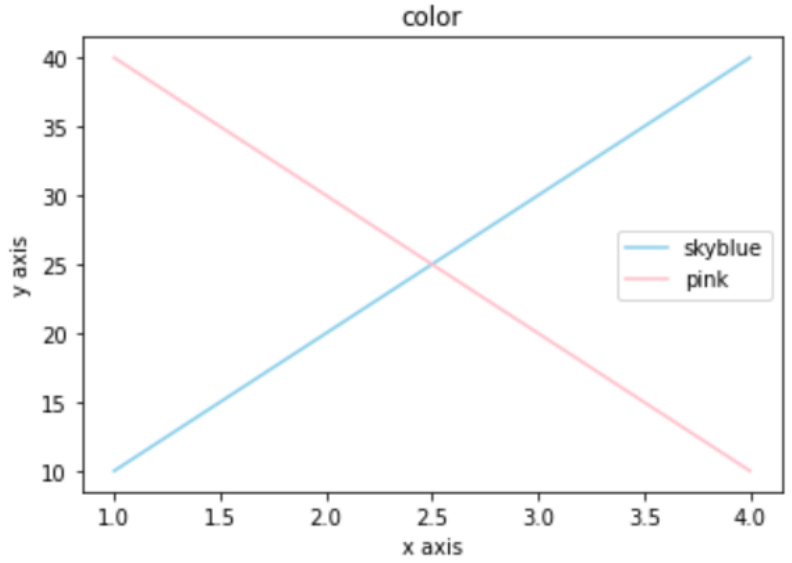
경우에 따라서 그래프를 출력할 때
흑백으로 출력하는 경우가 있습니다.
이 경우 색(Color)으로 범례를 구분하는
것이 어려울 수 있습니다.

- legend
  (명사) 전설
  (명사) 범례

Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 그래프 색상 바꾸기

```
import matplotlib.pyplot as plt

plt.plot([1, 2, 3, 4], [10, 20, 30, 40], color='skyblue', label='skyblue')
plt.plot([1, 2, 3, 4], [40, 30, 20, 10], 'pink', label='pink')
plt.title('color')
plt.xlabel('x axis')
plt.ylabel('y axis')
plt.legend()
plt.show()
```



아래 색상에 대해서는 약자로 표기할 수 있습니다.

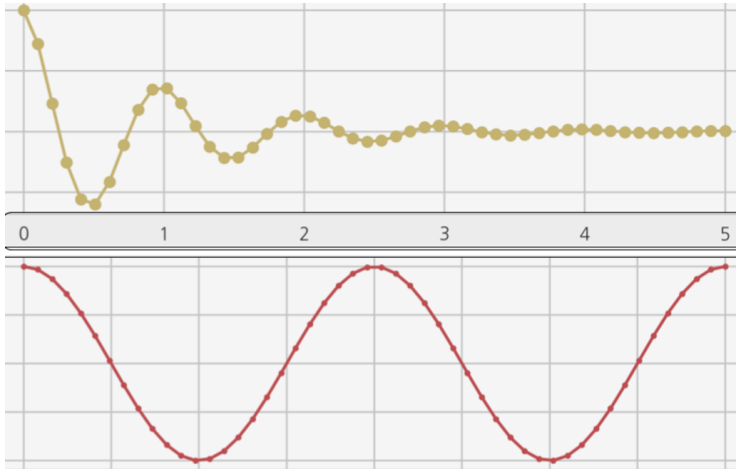
red → r
green → g
blue → b
black → k
yellow → y
white → w
magenta → m
cyan → c

•http://matplotlib.org/examples/color/named_colors.html

Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 그래프 선 모양 바꾸기

- 선 스타일 linestyle 또는 ls
  - ◆ 실선(solid), 대시선(dashed)
  - ◆ 점선(dotted), 대시-점선(dash-dot)
- 색상 color
- 마커(marker) marker
  - ◆ 데이터 위치를 나타내는 기호
    - ➢ 실제 데이터를 돋보이게 그리기



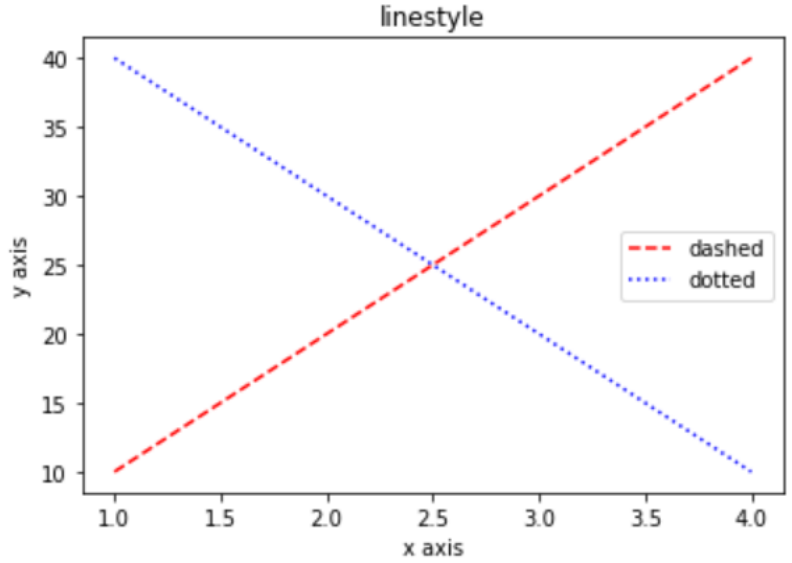
선 스타일 문자열	의미
`-`	solid line style
`--`	dashed line style
`-.`	dash-dot line style
`:`	dotted line style

문자열	약자
`blue`	`b`
`green`	`g`
`red`	`r`
`cyan`	`c`
`magenta`	`m`
`yellow`	`y`
`black`	`k`
`white`	`w`

Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 그래프 선 모양 바꾸기

```
import matplotlib.pyplot as plt

plt.plot([1, 2, 3, 4], [10, 20, 30, 40], color='r', linestyle='--', label='dashed')
plt.plot([1, 2, 3, 4], [40, 30, 20, 10], 'b', ls=':', label='dotted')
plt.title('linestyle')
plt.xlabel('x axis')
plt.ylabel('y axis')
plt.legend()
plt.show()
```



선 스타일

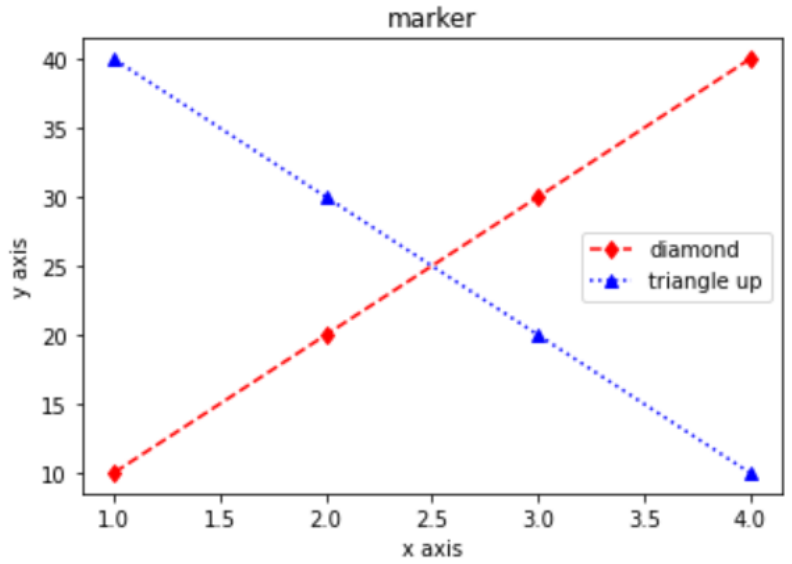
선 스타일에는 실선(solid), 대시선(dashed), 점선(dotted), 대시-점선(dash-dit)

선 스타일 문자열	의미
-	solid line style
--	dashed line style
-.	dash-dot line style
:	dotted line style

Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 그래프 마커(Marker) 모양 바꾸기

```
import matplotlib.pyplot as plt

plt.plot([1, 2, 3, 4], [10, 20, 30, 40], color='r', linestyle='--', marker='d', label='diamond')
plt.plot([1, 2, 3, 4], [40, 30, 20, 10], 'b', ls=':', marker='^', label='triangle up')
plt.title('marker')
plt.xlabel('x axis')
plt.ylabel('y axis')
plt.legend()
plt.show()
```



마커 문자열	의미
.	point marker
,	pixel marker
o	circle marker
v	triangle_down marker
^	triangle_up marker
<	triangle_left marker
>	triangle_right marker

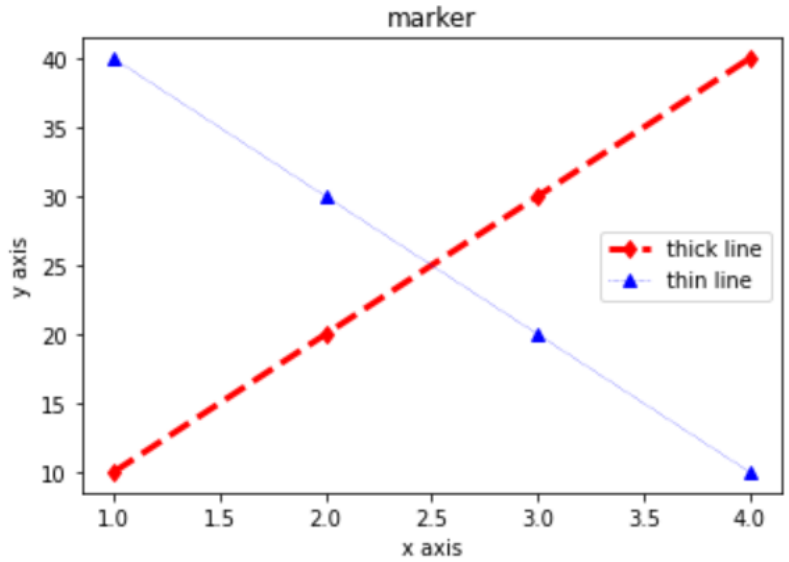
1	tri_down marker
2	tri_up marker
3	tri_left marker
4	tri_right marker
s	square marker
p	pentagon marker
*	star marker

h	hexagon1 marker
H	hexagon2 marker
+	plus marker
x	x marker
D	diamond marker
d	thin_diamond marker

Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 그래프 선 굵기 바꾸기

```
import matplotlib.pyplot as plt

plt.plot([1, 2, 3, 4], [10, 20, 30, 40], color='r', linestyle='--', marker='d', linewidth=3, label='thick line')
plt.plot([1, 2, 3, 4], [40, 30, 20, 10], 'b', ls=':', marker='^', linewidth=0.5, label='thin line')
plt.title('marker')
plt.xlabel('x axis')
plt.ylabel('y axis')
plt.legend()
plt.show()
```



Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 그래프 선명하게

그림을 선명하게

In [31]:
```
%config InlineBackend.figure_format = 'retina'
```


Copyright 2023. Dongyang Mirae University all rights reserved.

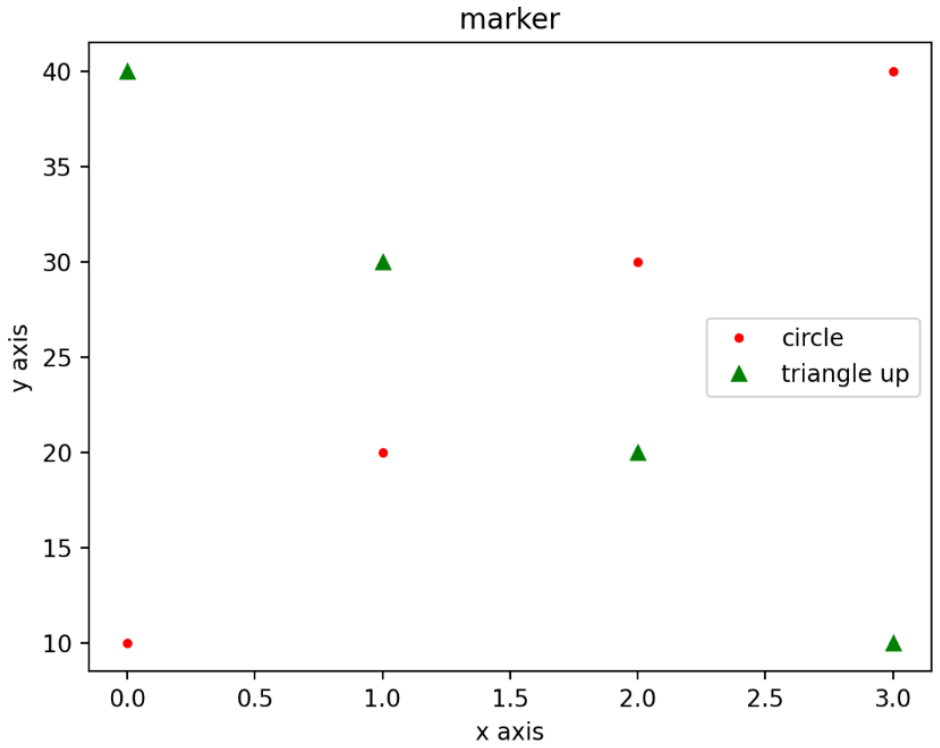
❖ 문자열 형태로 라인 모양

- fmt = '[color][marker][line]' 또는 '[marker][line][color]'
  - ◆파악만 되면 순서는 상관 없음
- 'b'
  - ◆# blue markers with default shape
- 'or'
  - ◆# red circles
- '-g'
  - ◆# green solid line
- '--'
  - ◆# dashed line with default color
- '^k:'
  - ◆# black triangle_up markers connected by a dotted line

Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 라인스타일이 없으면 마커만 그림

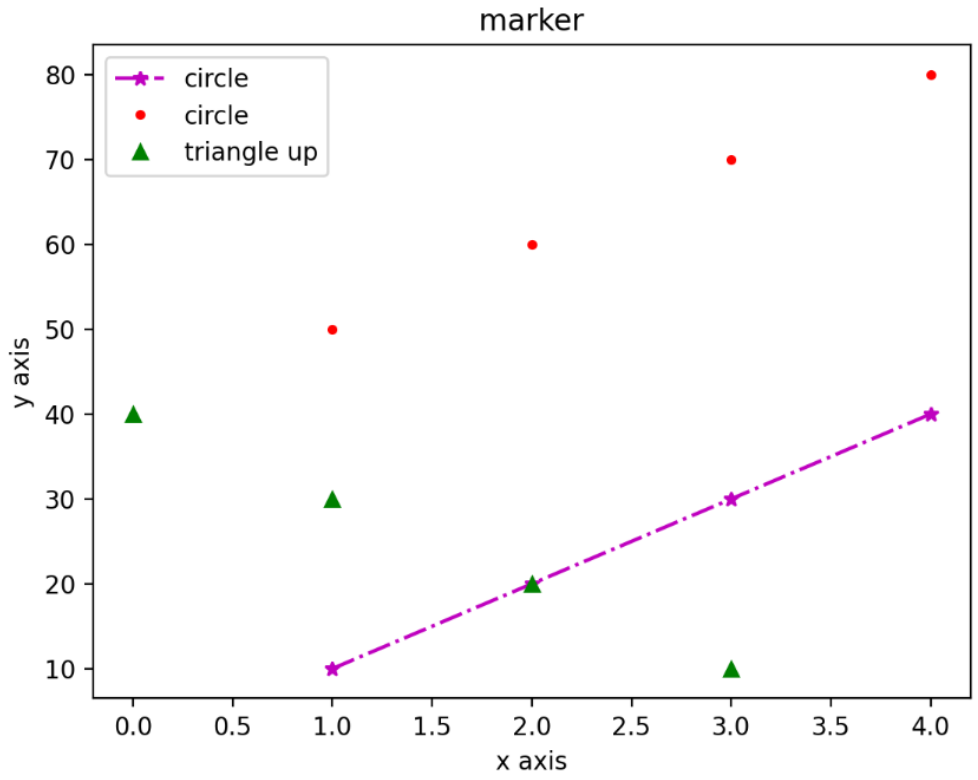
In [26]:
```
import matplotlib.pyplot as plt

# 라인스타일이 없으면 마커만 찍힘
plt.plot([10, 20, 30, 40], 'r.', label ='circle') # 빨간색 원형 마커 그래프
plt.plot([40, 30, 20, 10], 'g^', label ='triangle up') # 초록색 삼각형 마커 그래프

plt.title('marker') # 제목 설정
plt.xlabel('x axis')
plt.ylabel('y axis')
plt.legend() # 범례 표시
plt.show()
```



Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 라인스타일이 없으면 마커만 그림

In [33]:
```
import matplotlib.pyplot as plt

plt.plot([1, 2, 3, 4], [10, 20, 30, 40], '-.m*', [1, 2, 3, 4], [50, 60, 70, 80],  'r.', label ='circle')
plt.plot([40, 30, 20, 10], 'g^', label ='triangle up') # 초록색 삼각형 마커 그래프

plt.title('marker') # 제목 설정
plt.xlabel('x axis')
plt.ylabel('y axis')
plt.legend() # 범례 표시
plt.show()
```

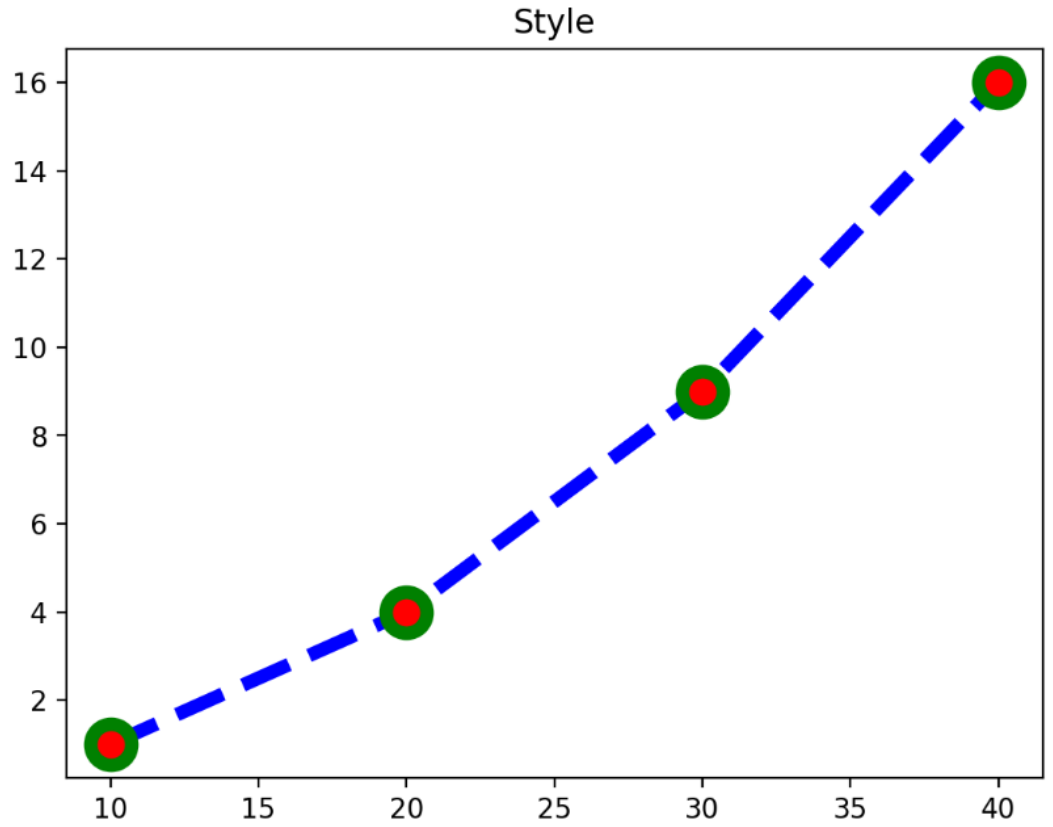


Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 기타 속성

다양한 그림 속성

In [38]:
```
plt.plot([10, 20, 30, 40], [1, 4, 9, 16], c="b",
         lw=5, ls="--", marker="o", ms=15, mec="g", mew=5, mfc="r")
plt.title("Style")
plt.show()
```

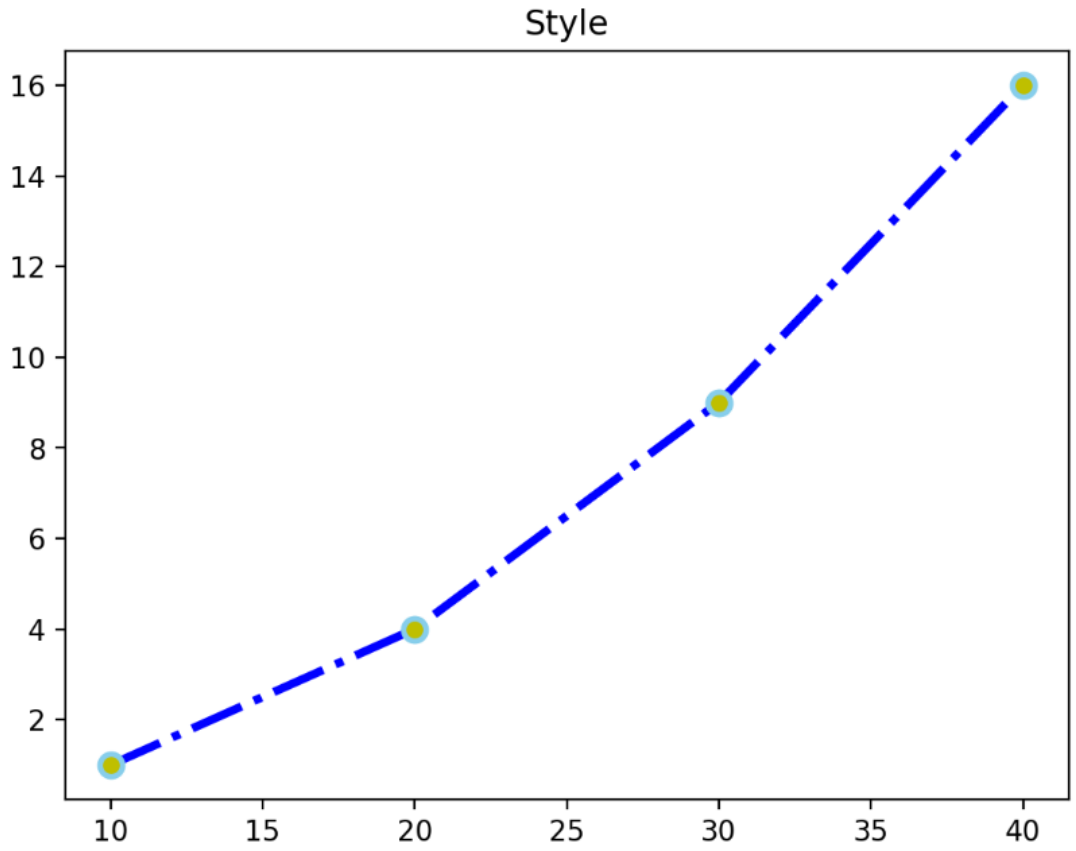


스타일 문자열	약자	의미
color	c	선 색깔
linewidth	lw	선 굵기
linestyle	ls	선 스타일
marker		마커 종류
markersize	ms	마커 크기
markeredgecolor	mec	마커 선 색깔
markeredgewidth	mew	마커 선 굵기
markerfacecolor	mfc	마커 내부 색깔

Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 기타 속성

```
In [40]: plt.plot([10, 20, 30, 40], [1, 4, 9, 16], "-.bo", lw=3, ms=8, mec="skyblue", mew=2, mfc="y")
         plt.title("Style")
         plt.show()
```



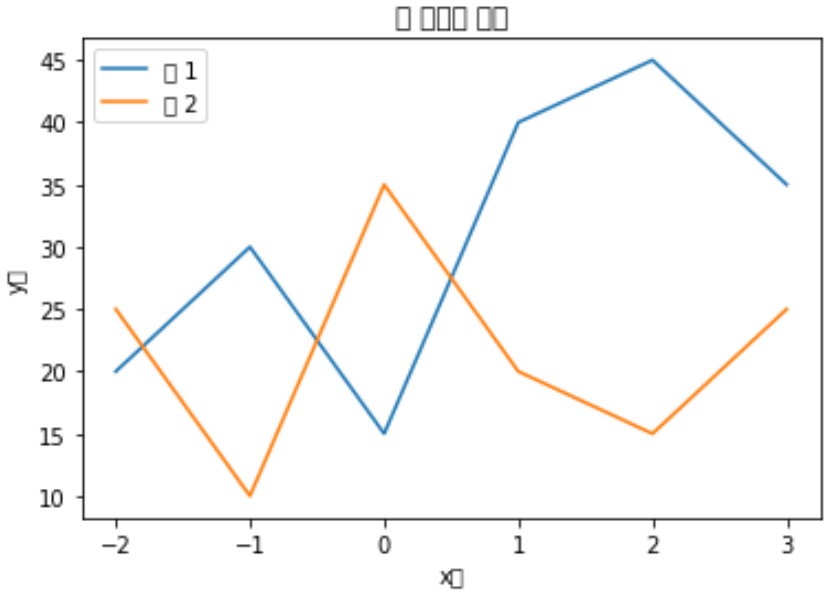
스타일 문자열	약자	의미
color	c	선 색깔
linewidth	lw	선 굵기
linestyle	ls	선 스타일
marker		마커 종류
markersize	ms	마커 크기
markeredgecolor	mec	마커 선 색깔
markeredgewidth	mew	마커 선 굵기
markerfacecolor	mfc	마커 내부 색깔

Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 한글 폰트 사용하기

- 그래프의 제목, 범례, x축 및 y축 레이블에 한글로 내용을 작성하면 어떻게 될까요?

```
import matplotlib.pyplot as plt

plt.plot([-2, -1, 0, 1, 2, 3], [20, 30, 15, 40, 45, 35], label='선 1')
plt.plot([-2, -1, 0, 1, 2, 3], [25, 10, 35, 20, 15, 25], label='선 2')
plt.title('선 그래프 예제')
plt.xlabel('x축')
plt.ylabel('y축')
plt.legend()
plt.show()
```



Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 경고 무시하기

```
import warnings

# 경고 메시지를 무시하고 숨기거나
warnings.filterwarnings(action='ignore')

# 숨기기했던 경고 메시지를 다시 보이게
warnings.filterwarnings(action='default')
```


Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 사용 폰트 목록 보이기

폰트 목록 보이기

In [46]:
```
import matplotlib as mpl
from matplotlib import font_manager

font_list = sorted([f.name for f in font_manager.fontManager.ttflist])
font_list
```

```
'Lucida Sans Typewriter',
'Lucida Sans Typewriter',
'Lucida Sans Typewriter',
'Lucida Sans Typewriter',
'Lucida Sans Unicode',
'MS Gothic',
'MS Outlook',
'MS Reference Sans Serif',
'MS Reference Specialty',
'MT Extra',
'MV Boli',
'Magic R',
'Magneto',
'Maiandra GD',
'Malgun Gothic',
'Malgun Gothic',
'Malgun Gothic',
'Malgun Gothic',
'Marlett',
'Matura MT Script Capitals',
'MesloLGS NF',
```

In [45]:
```
'Malgun Gothic' in font_list
```

Out[45]: True

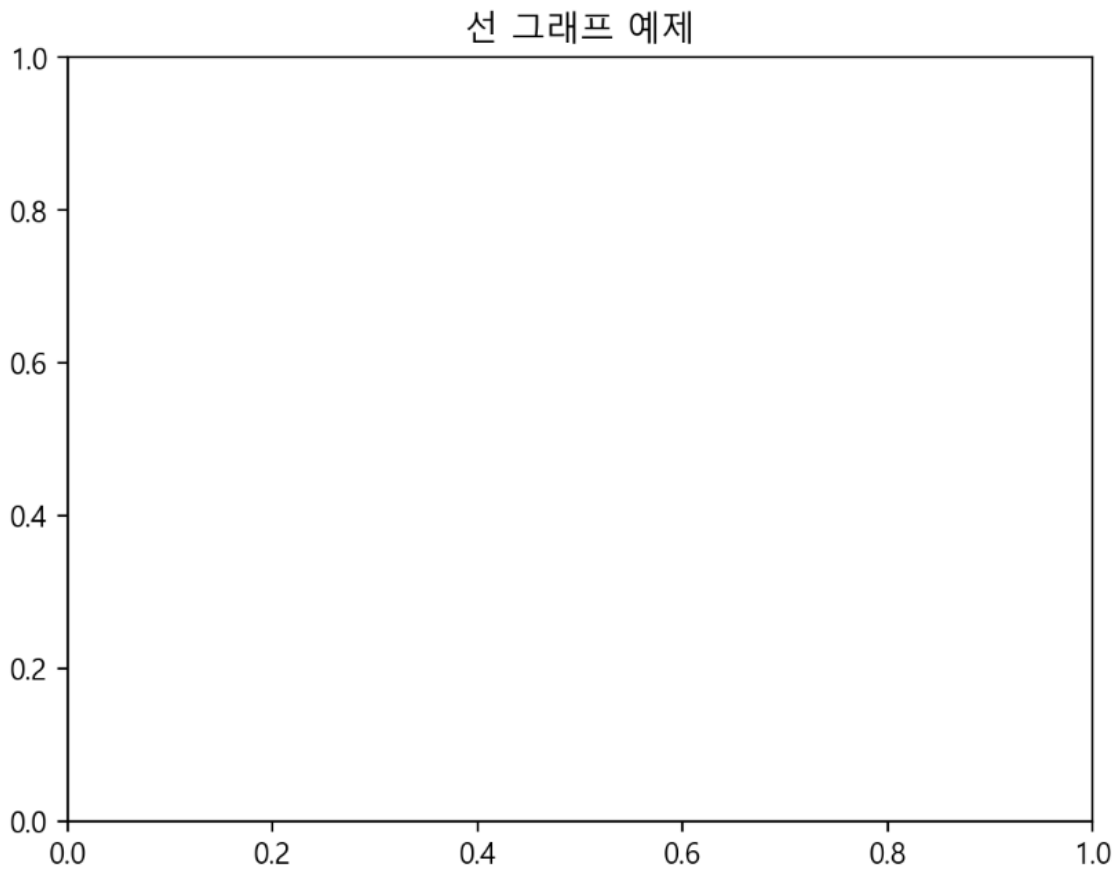
```
import matplotlib as mpl
from matplotlib import font_manager

font_list = sorted([f.name for f in font_manager.fontManager.ttflist])
font_list

'Malgun Gothic' in font_list
```


Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 한글 폰트 사용하기

In [62]:
```
plt.rc('font', family='Malgun Gothic')
plt.title('선 그래프 예제')
```

Out[62]: Text(0.5, 1.0, '선 그래프 예제')



Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 한글 폰트 사용하기

- Malgun Gothic은 "맑은 고딕" 확인

만약 macOS 운영체제를 사용하고 있다면 "AppleGothic"이라고 쓰세요.

지원 폰트 확인

In [4]:
```
import matplotlib as mpl
from matplotlib import font_manager
```

In [6]:
```
font_list = sorted([f.name for f in font_manager.fontManager.ttflist])
'Malgun Gothic' in font_list
```

Out[6]: True

Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 한글 폰트 사용하기

- Malgun Gothic은 "맑은 고딕"입니다.

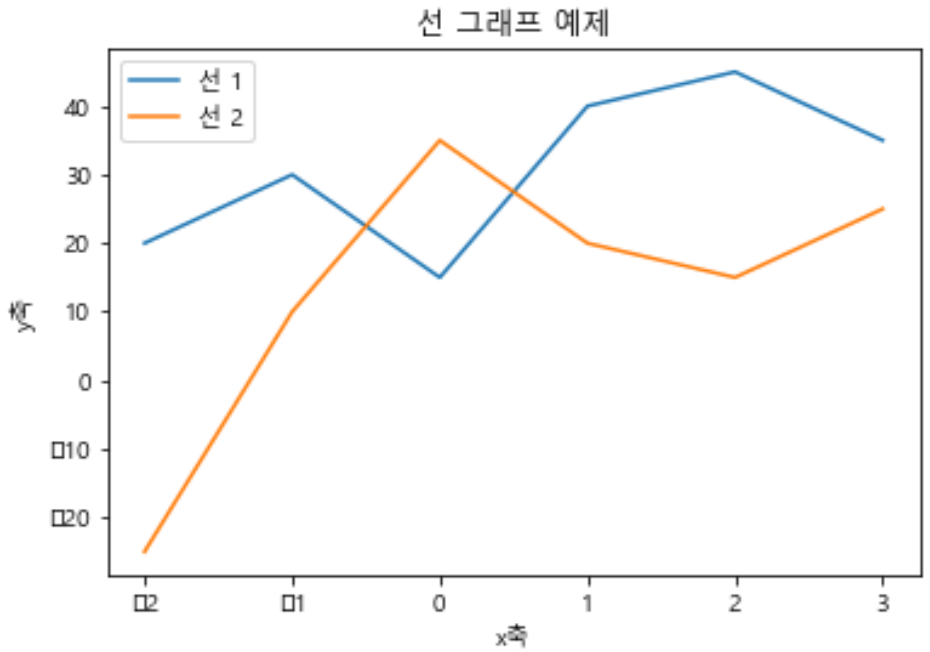
```
plt.rc('font', family='Malgun Gothic')
plt.title('선 그래프 예제')
```

만약 macOS 운영체제를 사용하고 있다면 "AppleGothic"이라고 쓰세요.

Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 한글 폰트 사용하기

```
import matplotlib.pyplot as plt

plt.rc('font', family='Malgun Gothic')
plt.plot([-2, -1, 0, 1, 2, 3], [20, 30, 15, 40, 45, 35], label='선 1')
plt.plot([-2, -1, 0, 1, 2, 3], [-25, 10, 35, 20, 15, 25], label='선 2')
plt.title('선 그래프 예제')
plt.xlabel('x축')
plt.ylabel('y축')
plt.legend()
plt.show()
```



한글로 작성했던 내용들은 잘 표시되지만,
마이너스 부호가 제대로 표시되지 않습니다.

Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 한글 폰트 사용하기

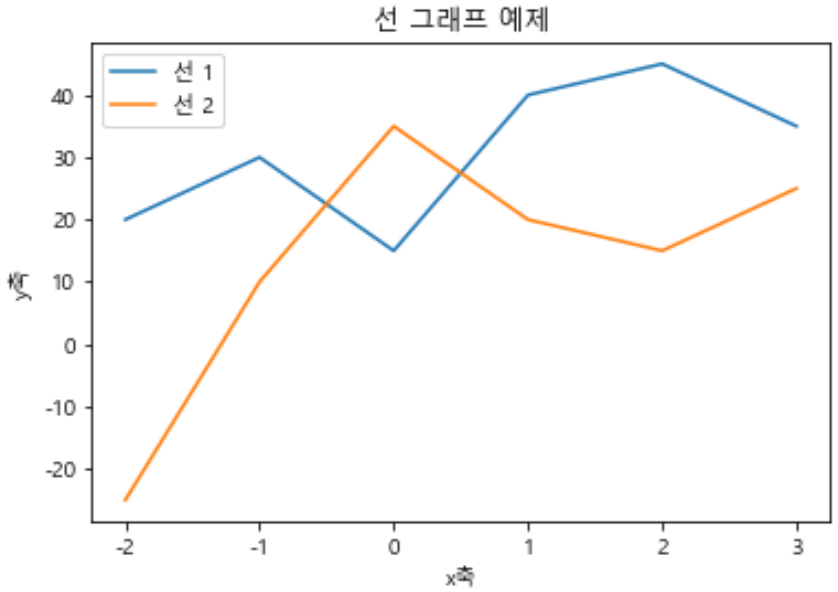
- 한글 폰트 사용시 마이너스 부호 표현하기

```
plt.rcParams['axes.unicode_minus'] = False
```


Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 한글 폰트 사용하기

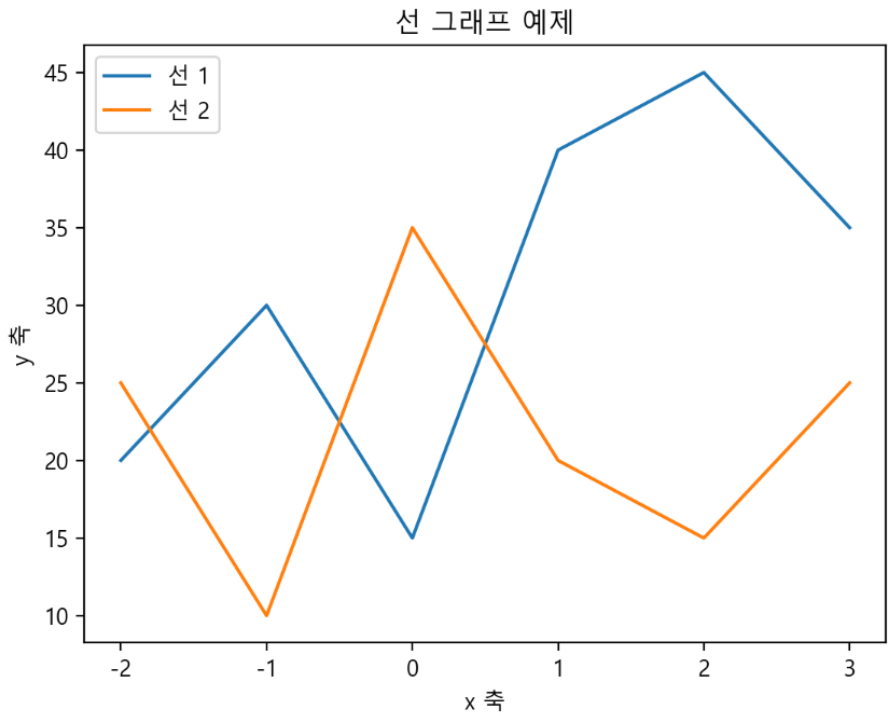
```
import matplotlib.pyplot as plt

plt.rc('font', family='Malgun Gothic')
plt.rcParams['axes.unicode_minus'] = False
plt.plot([-2, -1, 0, 1, 2, 3], [20, 30, 15, 40, 45, 35], label='선 1')
plt.plot([-2, -1, 0, 1, 2, 3], [-25, 10, 35, 20, 15, 25], label='선 2')
plt.title('선 그래프 예제')
plt.xlabel('x축')
plt.ylabel('y축')
plt.legend()
plt.show()
```



Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 한글 폰트 사용하기

In [24]:
```
import matplotlib.pyplot as plt

plt.plot([-2, -1, 0, 1, 2, 3], [20, 30, 15, 40, 45, 35], label= '선 1')
plt.plot([-2, -1, 0, 1, 2, 3], [25, 10, 35, 20, 15, 25], label= '선 2')

plt.title('선 그래프 예제')
plt.xlabel('x 축')
plt.ylabel('y 축')
plt.legend() # 범례 표시

plt.rc('font', family='Malgun Gothic')
plt.rcParams['axes.unicode_minus'] = False
plt.show()
```

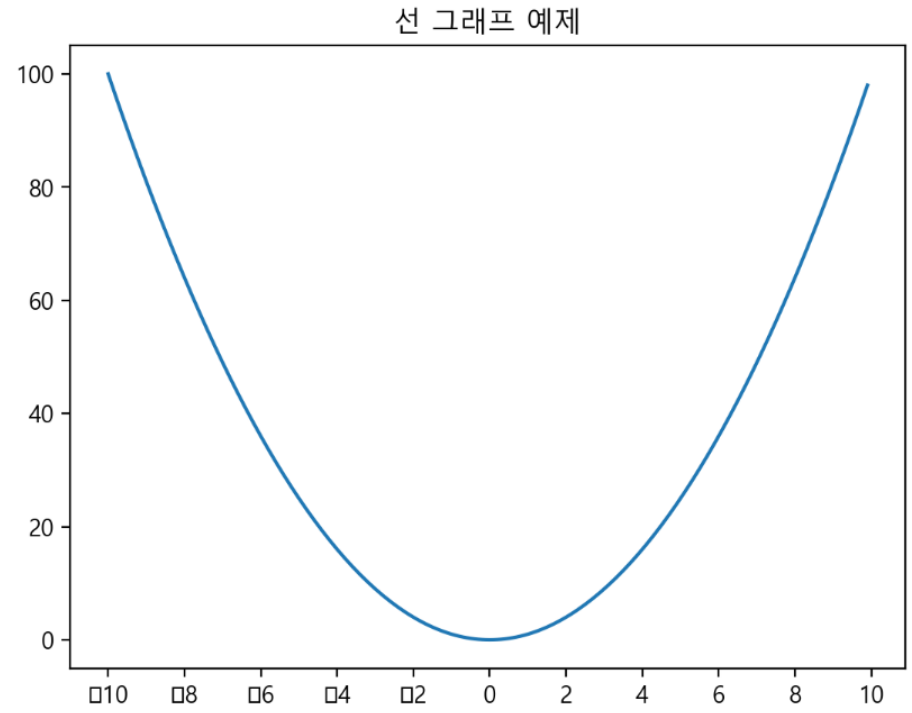


Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 한글 폰트 사용하기, x, y축 눈금과 내부 그리드 선

In [54]:
```
import numpy as np

x = np.arange(-10, 10, 0.1)
plt.plot(x, x**2)

plt.xticks(np.arange(-10, 11, 2)) # X 눈금
plt.yticks(np.arange(0, 101, 20)) # Y 눈금
# plt.grid(True)
plt.rc('font', family='Malgun Gothic')
plt.title('선 그래프 예제')
```

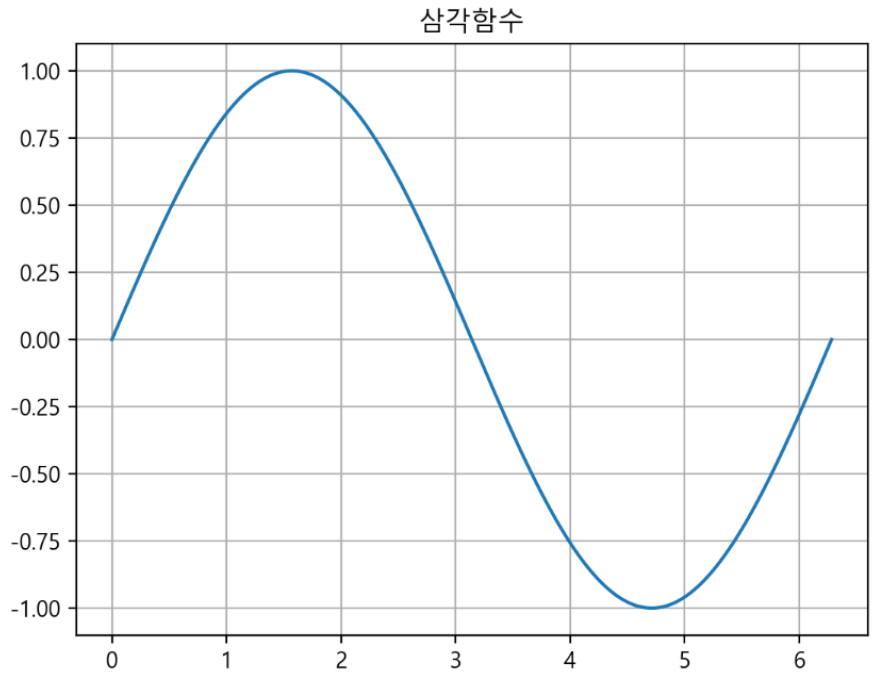
Out[54]: Text(0.5, 1.0, '선 그래프 예제')



Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 한글 폰트 사용하기, x, y축 눈금과 내부 그리드 선

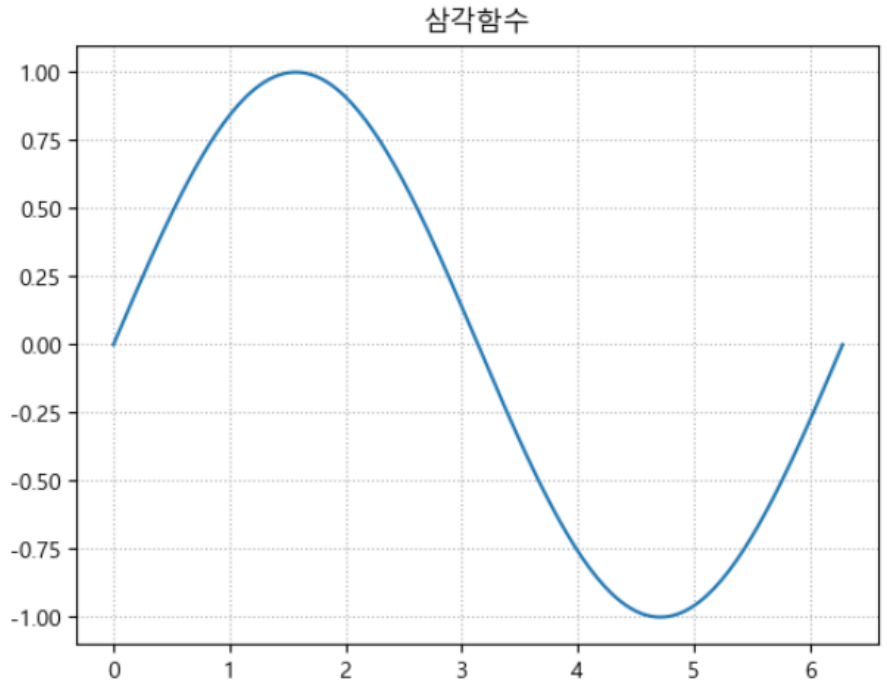
In [26]:
```
import matplotlib.pyplot as plt
import numpy as np

x = np.linspace(0, 2 * np.pi, 200) # 0에서 2*pi까지 경계선을 포함해 200등분
y = np.sin(x)
plt.plot(x, y)

plt.grid(True)
plt.rc('font', family='Malgun Gothic')
plt.rcParams['axes.unicode_minus'] = False
plt.title('삼각함수')
plt.show()
```



Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 한글 폰트 사용하기, x, y축 눈금과 내부 그리드 선

In [8]:
```
import matplotlib.pyplot as plt
import numpy as np

plt.rc('font', family='Malgun Gothic')
plt.rcParams['axes.unicode_minus'] = False

x = np.linspace(0, 2 * np.pi, 200) # 0에서 2*pi까지 경계선을 포함해 199등분
y = np.sin(x)
plt.plot(x, y)

plt.grid(True, ls=':')
plt.title('삼각함수')
plt.show()
```



Copyright 2023. Dongyang Mirae University all rights reserved.

❖ Numpy.linspace()

```
In [28]: help(np.linspace)
```

```
Help on function linspace in module numpy:

linspace(start, stop, num=50, endpoint=True, retstep=False, dtype=None, axis=0)
    Return evenly spaced numbers over a specified interval.

    Returns `num` evenly spaced samples, calculated over the
    interval [`start`, `stop`].

    The endpoint of the interval can optionally be excluded.

    .. versionchanged:: 1.16.0
        Non-scalar `start` and `stop` are now supported.

    .. versionchanged:: 1.20.0
        Values are rounded towards ``-inf`` instead of ``0`` when an
        integer ``dtype`` is specified. The old behavior can
        still be obtained with ``np.linspace(start, stop, num).astype(int)``

    Parameters
    ----------
    start : array_like
        The starting value of the sequence.
    stop : array_like
        The end value of the sequence, unless `endpoint` is set to False.
        In that case, the sequence consists of all but the last of ``num + 1``
        evenly spaced samples, so that `stop` is excluded.  Note that the step
        size changes when `endpoint` is False.
    num : int, optional
        Number of samples to generate. Default is 50. Must be non-negative.
    endpoint : bool, optional
        If True, `stop` is the last sample. Otherwise, it is not included.
        Default is True.
    retstep : bool, optional
        If True, return (`samples`, `step`), where `step` is the spacing
        between samples.
```

```
print(np.linspace(0, 2, 3))
# 0에서 2까지, 일정한 간격이 되도록 숫자 3개를 반환
# 결국 (3-1) 등분
```


Copyright 2023. Dongyang Mirae University all rights reserved.

❖ Numpy.linspace()

```
import numpy as np

print(np.linspace(0, 2, 3)) # 0에서 2까지, 일정한 간격이 되도록 숫자 3개를 반환, 결국 (3-1) 등분
```

```
[0. 1. 2.]
```


Copyright 2023. Dongyang Mirae University all rights reserved.

❖ Numpy.linspace()

```
print(np.linspace(2.0, 3.0, num=5)) # 5-1 = 4분등

print(np.linspace(2.0, 3.0, num=5, retstep=True)) # 5-1 = 4등분, 수와 등분 간격을 반환
```

```
[2.   2.25 2.5  2.75 3.  ]
(array([2.  , 2.25, 2.5 , 2.75, 3.  ]), 0.25)
```


Copyright 2023. Dongyang Mirae University all rights reserved.

❖ Numpy.linspace()

```
np.linspace(2.0, 3.0, num=6, retstep=True) # 5등분
```

```
(array([2. , 2.2, 2.4, 2.6, 2.8, 3. ]), 0.2)
```

```
np.linspace(2.0, 3.0, num=6, endpoint=False, retstep=True) # 마지막 점을 빼고 5등분
```

```
(array([2.        , 2.16666667, 2.33333333, 2.5       , 2.66666667,
        2.83333333]),
 0.16666666666666666)
```

```
1/6
```

```
0.16666666666666666
```


Copyright 2023. Dongyang Mirae University all rights reserved.

❖ Numpy.zeros(n)

- N개의 원소에 0을 저장한 배열 생성

```
import numpy as np

y = np.zeros(3)
y
```

```
array([0., 0., 0.])
```

```
np.zeros(6)
```

```
array([0., 0., 0., 0., 0., 0.])
```

```
3 + np.zeros(8)
```

```
array([3., 3., 3., 3., 3., 3., 3., 3.])
```

```
np.ones(5)
```

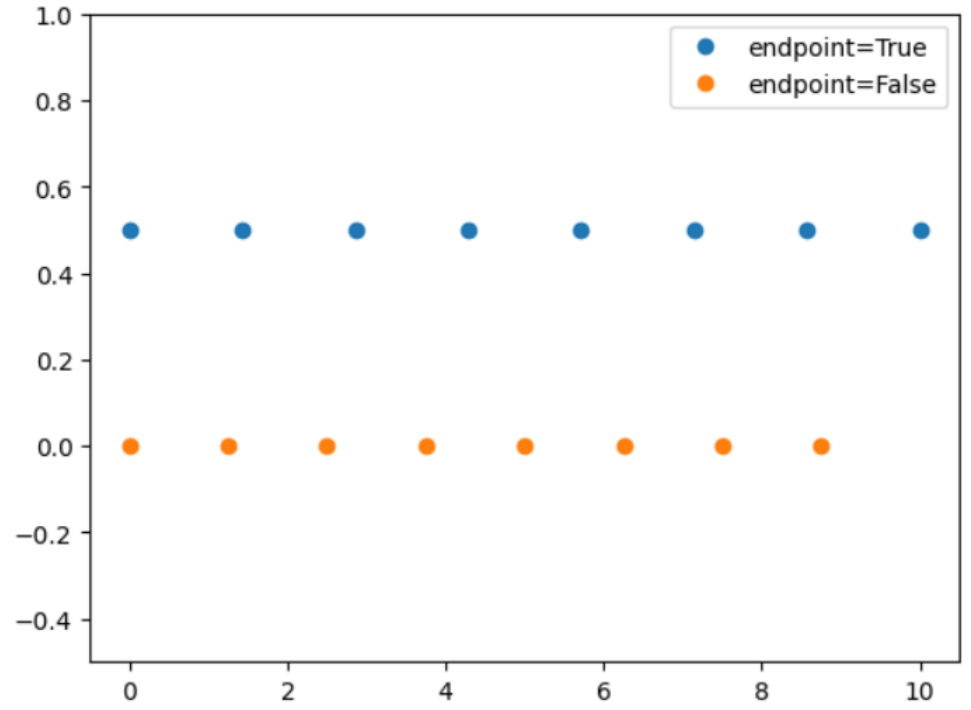
```
array([1., 1., 1., 1., 1.])
```


Copyright 2023. Dongyang Mirae University all rights reserved.

02. plot 함수로 선 그래프 그리기

❖ Numpy.linspace()

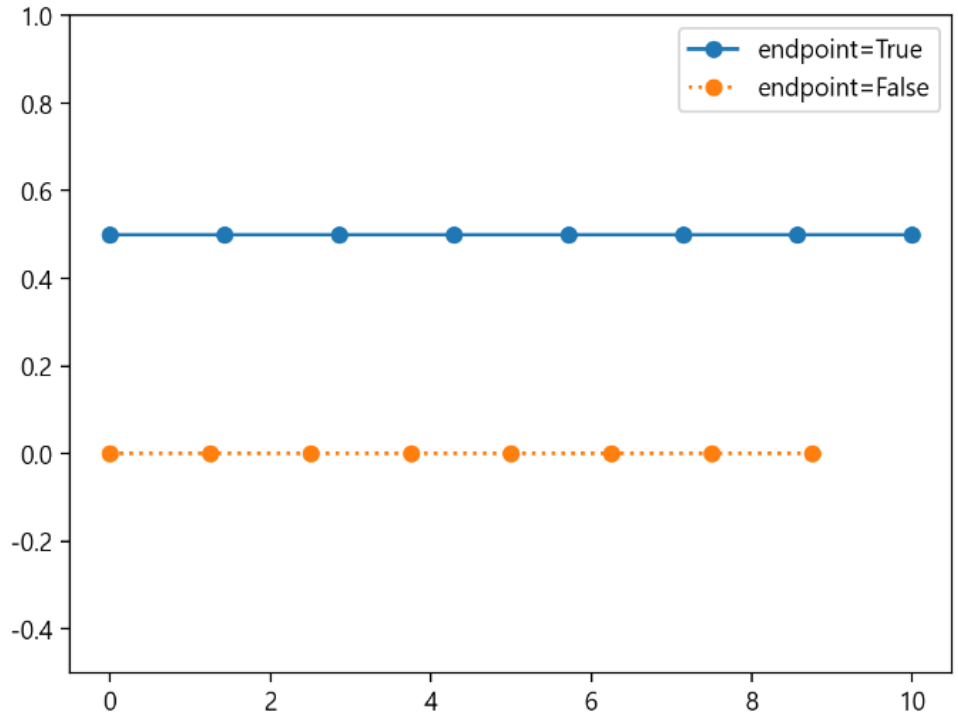
```
import matplotlib.pyplot as plt
import numpy as np

N = 8
y = np.zeros(N)

x1 = np.linspace(0, 10, N, endpoint=True) # 포함해서 7등분
x2 = np.linspace(0, 10, N, endpoint=False) # 빼고 7등분
plt.plot(x1, y + 0.5, 'o', label='endpoint=True')
plt.plot(x2, y, 'o', label='endpoint=False')

plt.legend()
plt.ylim([-0.5, 1]) # y 축 범위 지정
plt.show()
```



Copyright 2023. Dongyang Mirae University all rights reserved.

❖ Numpy.linspace()

```
import matplotlib.pyplot as plt
import numpy as np

N = 8
y = np.zeros(N)

x1 = np.linspace(0, 10, N, endpoint=True) # 포함해서 7등분
x2 = np.linspace(0, 10, N, endpoint=False) # 빼고 7등분
plt.plot(x1, y + 0.5, 'o-', label='endpoint=True')
plt.plot(x2, y, 'o:', label='endpoint=False')

plt.legend()
plt.ylim([-0.5, 1])
plt.show()
```

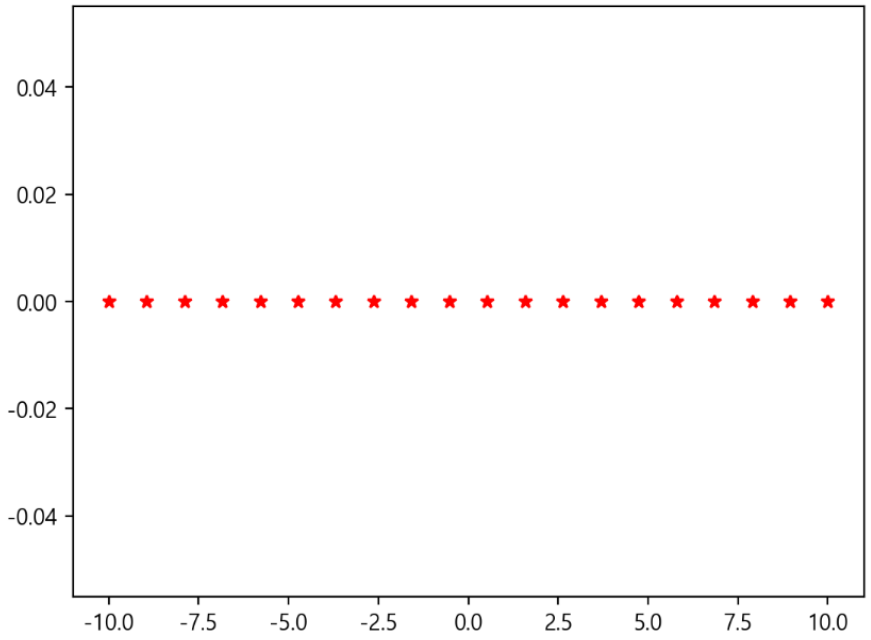


Copyright 2023. Dongyang Mirae University all rights reserved.

❖ Numpy.linspace()

In [37]:
```
import matplotlib.pyplot as plt
import numpy as np

x = np.linspace(-10, 10, 20) # -10에서 10까지 (20-1)등분
# x = np.linspace(-10, 10, 21) # -10에서 10까지 (20-1)등분
y = [0]*20
# y = np.zeros(20)

plt.plot(x, y, 'r*')
print(x)
```

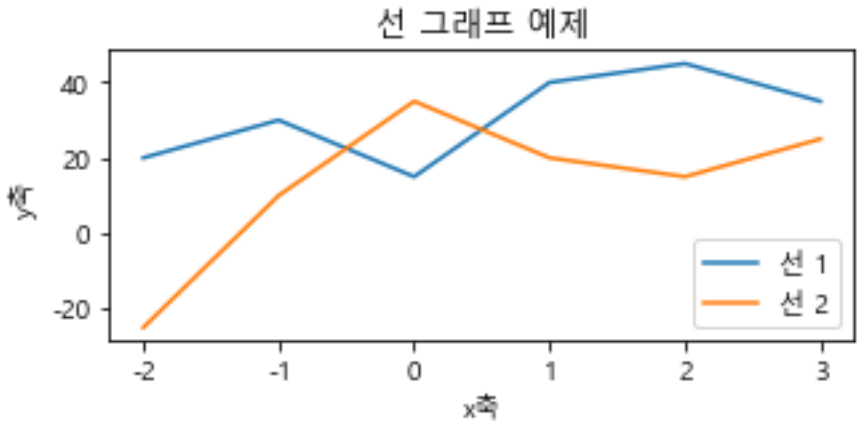
```
[-10.          -8.94736842  -7.89473684  -6.84210526  -5.78947368
  -4.73684211  -3.68421053  -2.63157895  -1.57894737  -0.52631579
   0.52631579   1.57894737   2.63157895   3.68421053   4.73684211
   5.78947368   6.84210526   7.89473684   8.94736842  10.        ]
```



Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 그래프 크기 설정하기

```
import matplotlib.pyplot as plt

plt.figure(figsize=(5, 2))
plt.rc('font', family='Malgun Gothic')
plt.rcParams['axes.unicode_minus'] = False
plt.plot([-2, -1, 0, 1, 2, 3], [20, 30, 15, 40, 45, 35], label='선 1')
plt.plot([-2, -1, 0, 1, 2, 3], [-25, 10, 35, 20, 15, 25], label='선 2')
plt.title('선 그래프 예제')
plt.xlabel('x축')
plt.ylabel('y축')
plt.legend()
plt.show()
```



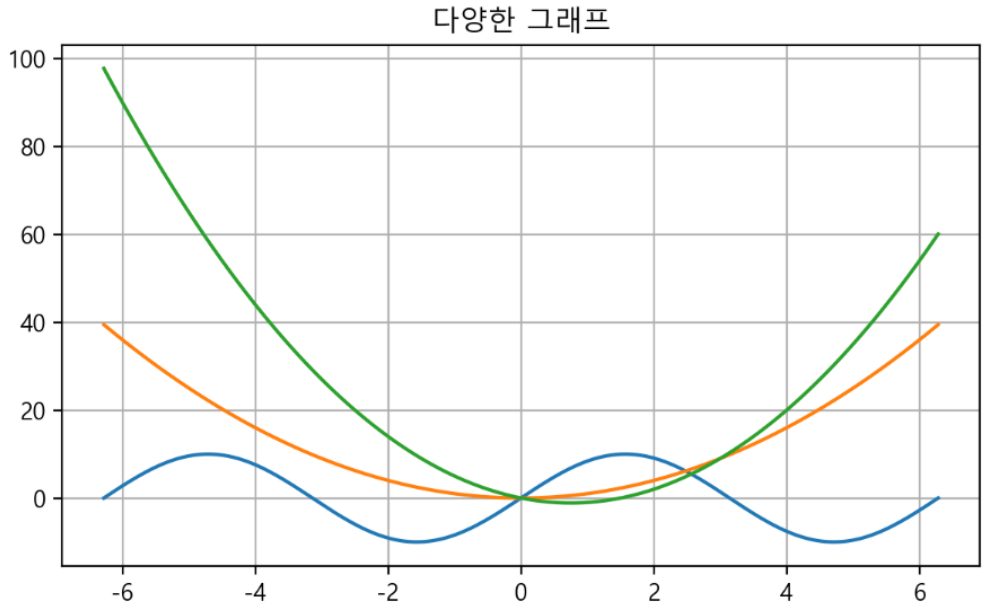
figsize=(가로 길이, 세로 길이)

길이의 단위 인치(Inch)
1 inch = 2.54 cm

Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 그래프 크기 설정하기

In [46]:
```
import matplotlib.pyplot as plt
import numpy as np

plt.figure(figsize=(7, 4))
plt.rc('font', family='Malgun Gothic')
plt.rcParams['axes.unicode_minus'] = False

x = np.linspace(-2 * np.pi, 2 * np.pi, 200) # 0에서 2*pi까지 경계선을 포함해 199등분
plt.plot(x, np.sin(x)*10)
plt.plot(x, x**2)
plt.plot(x, 2 * x**2 - 3*x)

plt.grid(True)
plt.title('다양한 그래프')
plt.show()
```



figsize=(가로 길이, 세로 길이)

길이의 단위 인치(Inch)
1 inch = 2.54 cm

Copyright 2023. Dongyang Mirae University all rights reserved.

03. hist 함수로 히스토그램 그리기

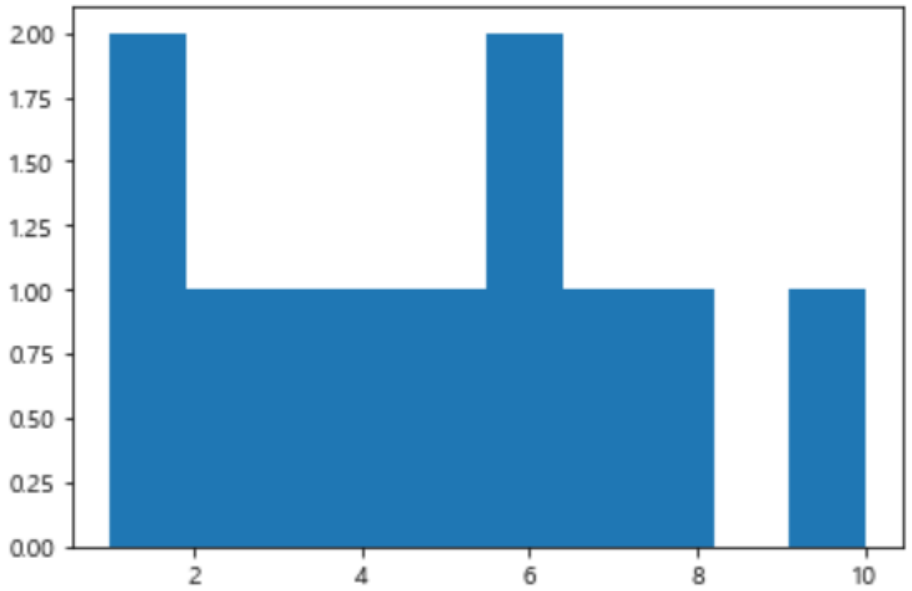
01. matplotlib 라이브러리 소개
02. plot 함수로 선 그래프 그리기
04. boxplot 함수로 상자 그림 그리기

Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 히스토그램 (1/6)

- 히스토그램(Histogram)은 데이터의 분포 상태를 직사각형 모양의 막대 그래프로 나타냅니다.
- 데이터의 빈도에 따라 직사각형의 높이가 결정됩니다.
- hist( ) 함수

```
import matplotlib.pyplot as plt

plt.hist([1, 1, 2, 3, 4, 5, 6, 6, 7, 8, 10])
plt.show()
```

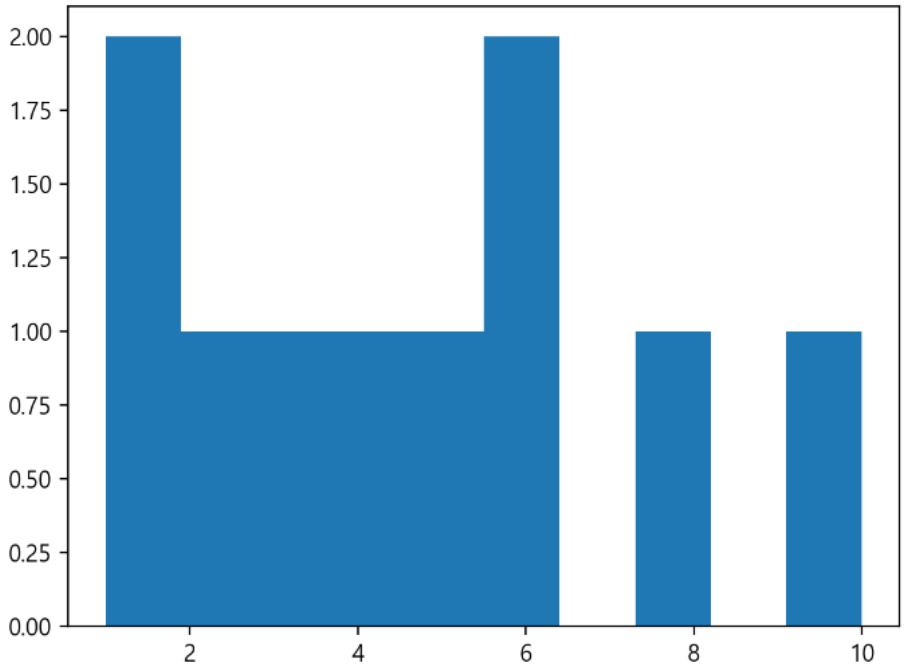


Copyright 2023. Dongyang Mirae University all rights reserved.

03. hist 함수로 히스토그램 그리기

❖ 히스토그램 반환 값

In [66]:
```
import matplotlib.pyplot as plt

arrays, bins, patches = plt.hist([1, 1, 2, 3, 4, 5, 6, 6, 8, 10])
plt.show()
```



In [67]:
```
arrays
```

Out[67]:
```
array([2., 1., 1., 1., 1., 2., 0., 1., 0., 1.])
```

In [68]:
```
bins
```

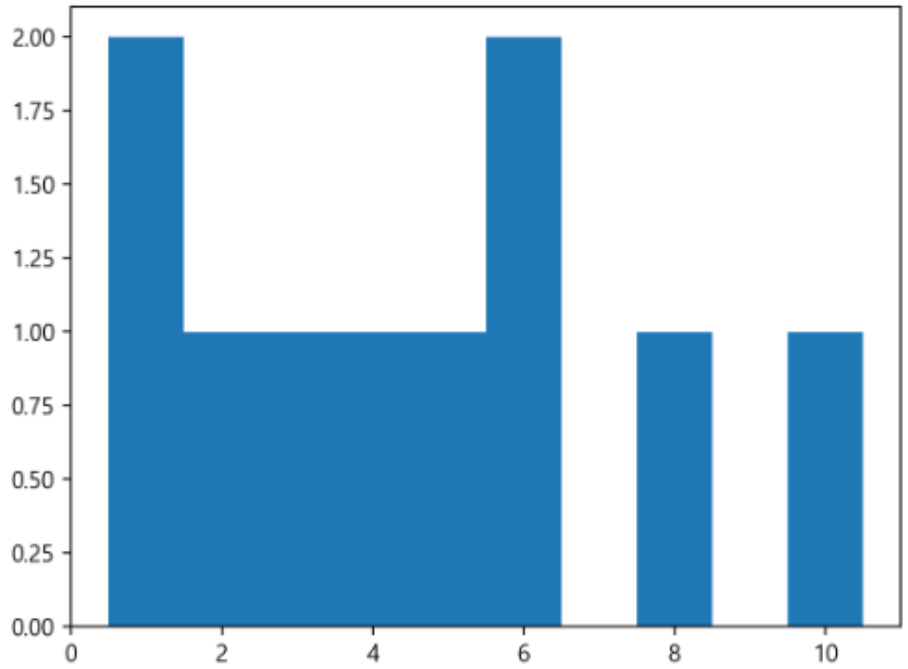
Out[68]:
```
array([ 1. ,  1.9,  2.8,  3.7,  4.6,  5.5,  6.4,  7.3,  8.2,  9.1, 10. ])
```


Copyright 2023. Dongyang Mirae University all rights reserved.

03. hist 함수로 히스토그램 그리기

❖ 히스토그램 반환 값

In [113]:
```
import matplotlib.pyplot as plt
import numpy as np

arrays, bins, patches = plt.hist([1, 1, 2, 3, 4, 5, 6, 6, 8, 10], bins=np.arange(0.5, 11.5))
plt.show()
```



In [114]:
```
arrays
```

Out[114]:
```
array([2., 1., 1., 1., 1., 2., 0., 1., 0., 1.])
```

In [115]:
```
bins
```

Out[115]:
```
array([ 0.5,  1.5,  2.5,  3.5,  4.5,  5.5,  6.5,  7.5,  8.5,  9.5, 10.5])
```


Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 히스토그램 (2/6)

- 주사위 시뮬레이션
  - ◆ Step 1) 주사위를 굴린다. — 1과 6사이의 랜덤 숫자를 만든다.
  - ◆ Step 2) 나온 결과를 기록한다. — 리스트에 저장한다.
  - ◆ Step 3) Steps 1-2의 과정을 n번 반복한다. — for 반복문
  - ◆ Step 4) 주사위의 눈이 나온 횟수를 히스토그램으로 그린다.

Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 히스토그램 (3/6)

- 1과 6사이의 랜덤 숫자 만들기

```
import random

print(random.randint(1, 6))
```


Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 히스토그램 (4/6)

- 주사위 시뮬레이션 5회 수행하기

```
import random

dice = []

for i in range(5):
    dice.append(random.randint(1, 6))

print(dice)
```

```
[3, 6, 4, 1, 5]
```

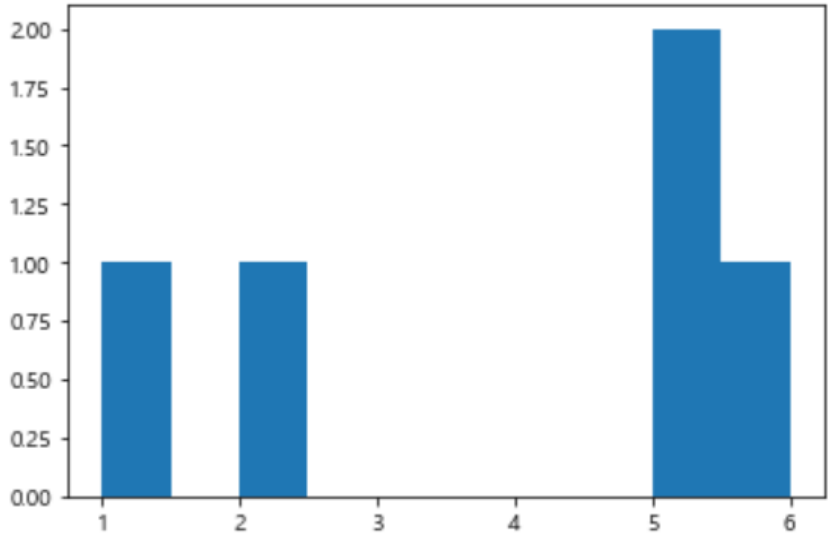
매 수행마다 값이 달라집니다.

Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 히스토그램 (5/6)

- 주사위 시뮬레이션 5회 수행 결과를 히스토그램으로 시각화하기

```
import random
import matplotlib.pyplot as plt

dice = []

for i in range(5):
    dice.append(random.randint(1, 6))

plt.hist(dice, bins=6)
plt.show()
```

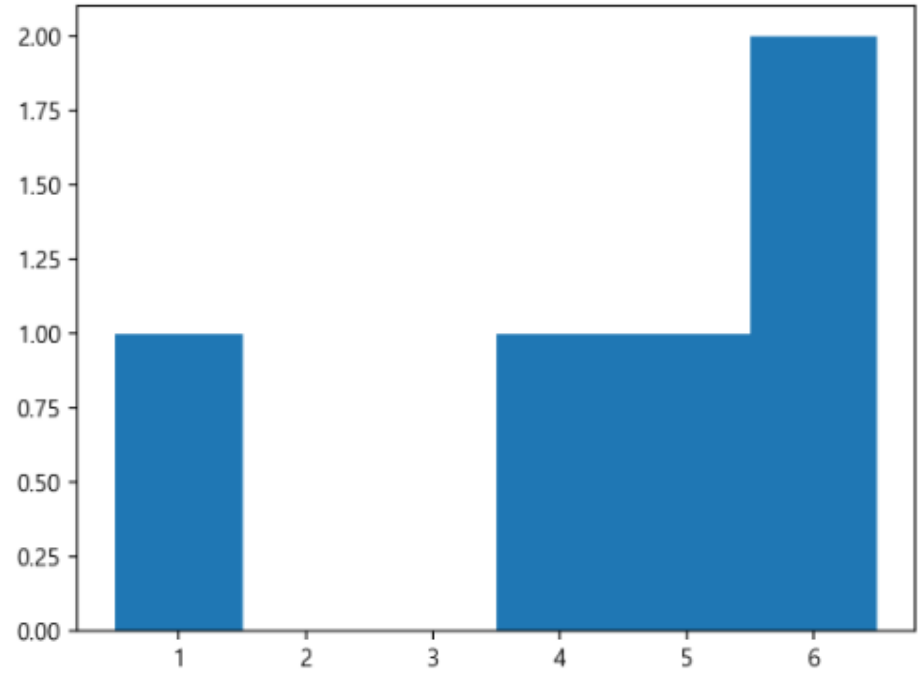


Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 히스토그램 (5/6)

- 주사위 시뮬레이션 5회 수행 결과를 히스토그램으로 시각화하기

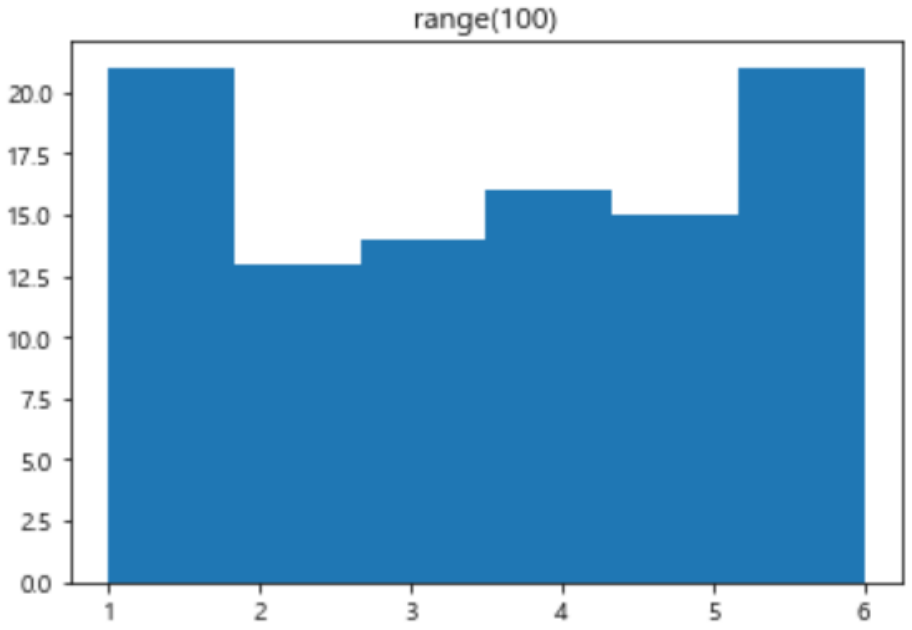
In [116]:
```
import random
import matplotlib.pyplot as plt

dice = []

for i in range(5):
    dice.append(random.randint(1, 6))

plt.hist(dice, bins=np.arange(0.5, 7.5))
plt.show()
```

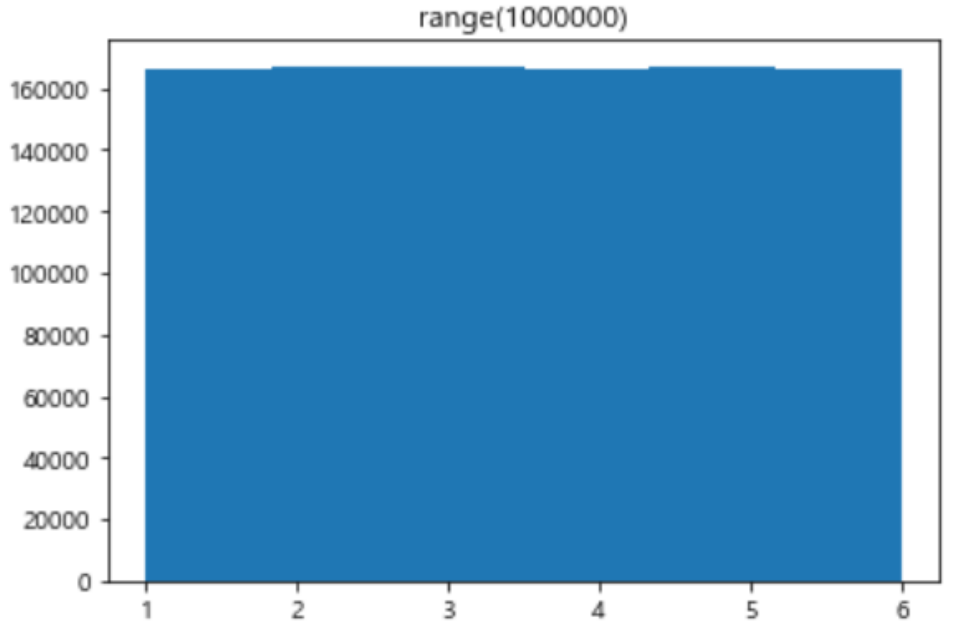


Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 히스토그램 (6/6)

- range(5)를 range(100), range(1000000)으로 수정하여 히스토그램 결과를 확인합니다.





주사위를 던지는 횟수가 늘어날 수록 주사위의 특정 숫자가 나오는 횟수가 전체의 1/6에 가까워 짐을 알 수 있습니다.

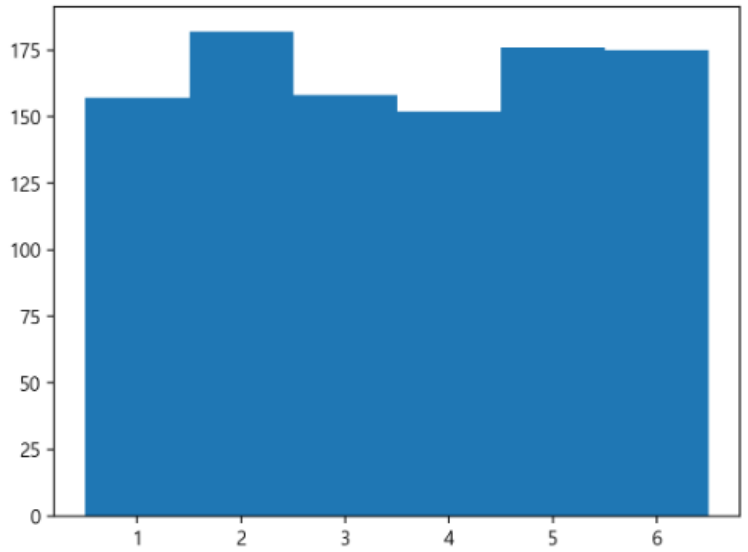
큰 수의 법칙

Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 히스토그램 (6/6)

- range(5)를 range(100), range(1000000)으로 수정하여 히스토그램 결과를 확인합니다.

In [117]:
```
import random
import matplotlib.pyplot as plt

dice = []

for i in range(1000):
    dice.append(random.randint(1, 6))

plt.hist(dice, bins=np.arange(0.5, 7.5))
plt.show()
```



주사위를 던지는 횟수가 늘어날 수록 주사위의 특정 숫자가 나오는 횟수가 전체의 1/6에 가까워 짐을 알 수 있습니다.

큰 수의 법칙

Copyright 2023. Dongyang Mirae University all rights reserved.

04. boxplot 함수로 상자 그림 그리기

01. matplotlib 라이브러리 소개
02. plot 함수로 선 그래프 그리기
03. hist 함수로 히스토그램 그리기

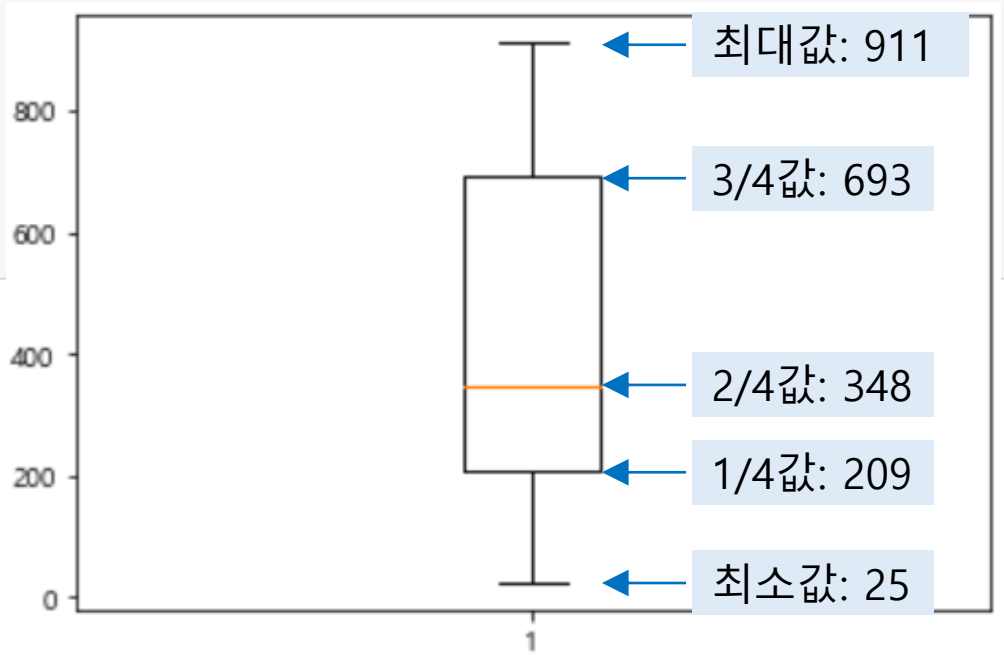
Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 상자 그림 (1/3)

- 상자 그림(Boxplot)은 데이터에서 얻어낸 최대값, 최소값, 상위 1/4, 2/4 (중앙), 3/4에 위치한 값을 보여주는 그래프입니다.

```
import matplotlib.pyplot as plt
import random

result = []
for i in range(13):
    result.append(random.randint(1, 1000))

print(sorted(result))

plt.boxplot(result)
plt.show()
```

```
[25, 56, 138, 209, 286, 323, 348, 386, 468, 693, 756, 843, 911]
```



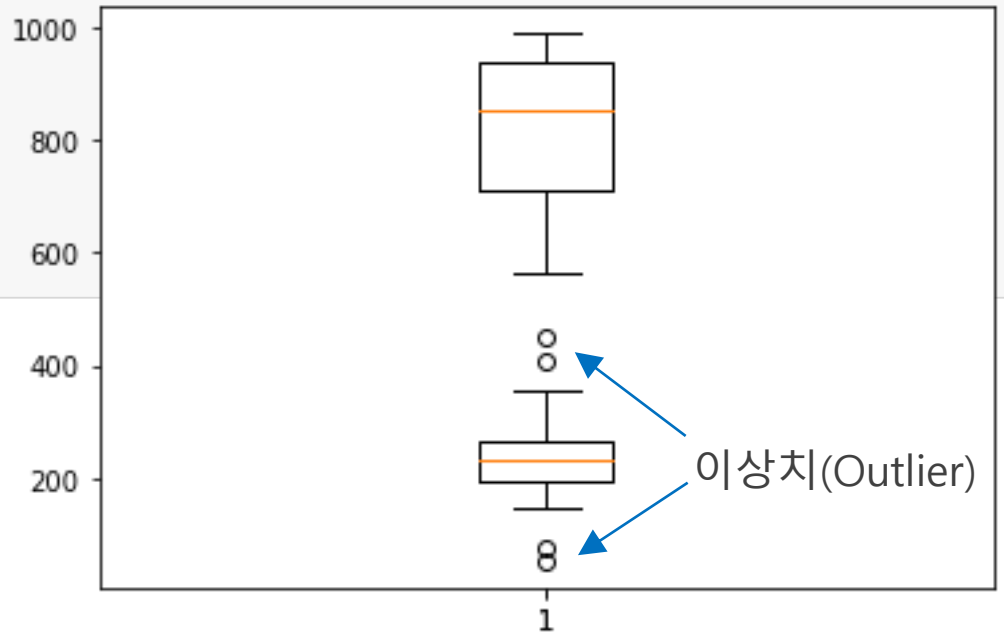
Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 상자 그림 (2/3)

- 하나의 그래프에서 두 데이터의 상자 그림 그리기

```
import matplotlib.pyplot as plt
import random

result1 = []
result2 = []
for i in range(13):
    result1.append(random.randint(1, 500))
    result2.append(random.randint(501, 1000))

plt.boxplot(result1)
plt.boxplot(result2)
plt.show()
```

두 데이터를
분리하여 표현할 순 없을까요?

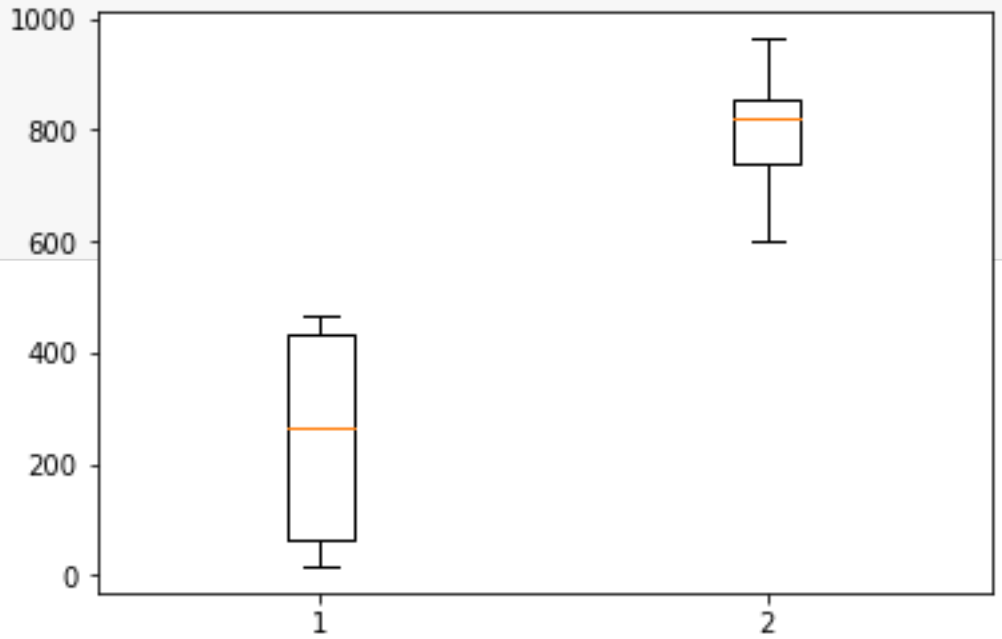


Copyright 2023. Dongyang Mirae University all rights reserved.

❖ 상자 그림 (3/3)

- 하나의 그래프에서 두 데이터의 상자 그림 그리기

```
import matplotlib.pyplot as plt
import random

result1 = []
result2 = []
for i in range(13):
    result1.append(random.randint(1, 500))
    result2.append(random.randint(501, 1000))

plt.boxplot([result1, result2])
plt.show()
```



Copyright 2023. Dongyang Mirae University all rights reserved.

Copyright 2023. Dongyang Mirae University all rights reserved.

Colab matplotlib 한글 지원

❖ 모듈 설치

```
---------------------------------------------------------------------------
# colab 한글지원 #1, #2, 이후에 바로 가능

# 1. 모듈 설치 koreanize-matplotlib
# 참조 사이트 https://github.com/ychoi-kr/koreanize-matplotlib
!pip install koreanize-matplotlib

# 2. 한글을 선명하게
%config InlineBackend.figure_format = 'retina'
```

```
[1]  # 1. 모듈 설치 koreanize-matplotlib
     # 참조 사이트 https://github.com/ychoi-kr/koreanize-matplotlib
     !pip install koreanize-matplotlib
```

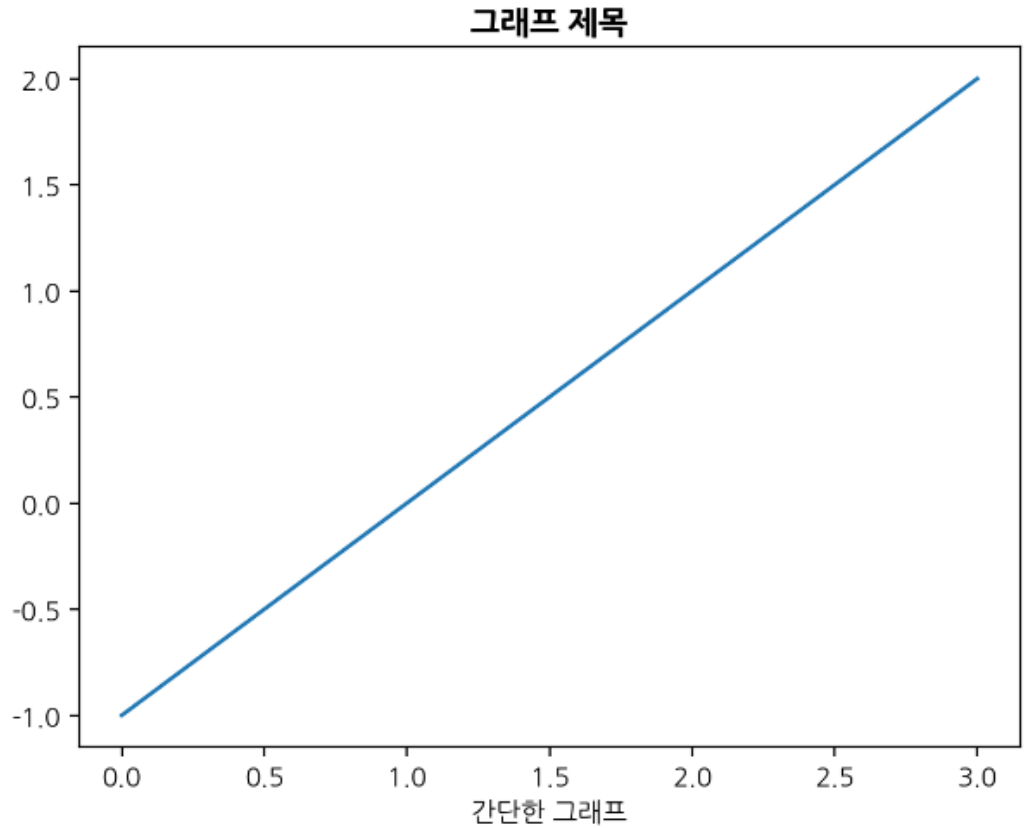
```
Requirement already satisfied: koreanize-matplotlib in /usr/local/l
Requirement already satisfied: matplotlib in /usr/local/lib/python3
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/p
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/pytho
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/
Requirement already satisfied: numpy>=1.20 in /usr/local/lib/python
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/py
Requirement already satisfied: pillow>=6.2.0 in /usr/local/lib/pyth
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/p
Requirement already satisfied: python-dateutil>=2.7 in /usr/local/l
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.1
```

```
[2]  %config InlineBackend.figure_format = 'retina'
```


Copyright 2023. Dongyang Mirae University all rights reserved.

```
# 3. 바로 실행 테스트
import koreanize_matplotlib
import matplotlib.pyplot as plt

plt.plot([-1, 0, 1, 2])
plt.title('그래프 제목', fontweight="bold")
plt.xlabel('간단한 그래프')
plt.show()
```

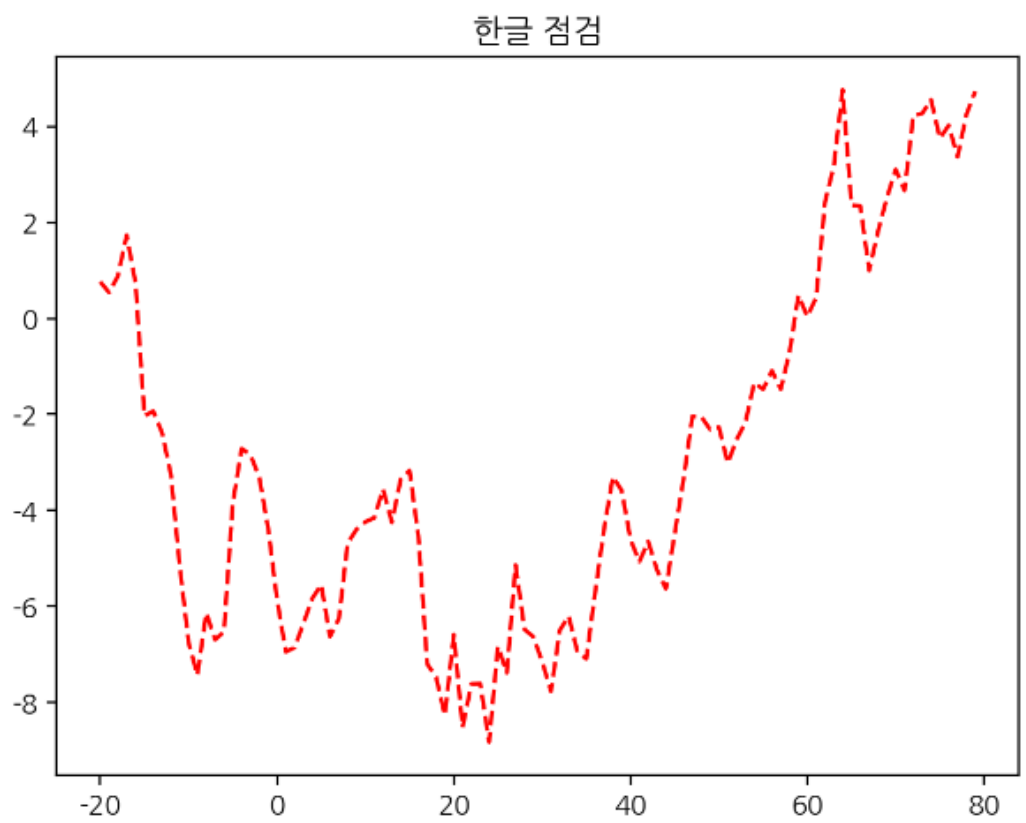
```
import matplotlib.pyplot as plt
import koreanize_matplotlib

plt.plot([-1, 0, 1, 2])
plt.title('그래프 제목', fontweight="bold")
plt.xlabel('간단한 그래프')
plt.show()
```



Copyright 2023. Dongyang Mirae University all rights reserved.

Colab matplotlib 한글 지원 테스트2

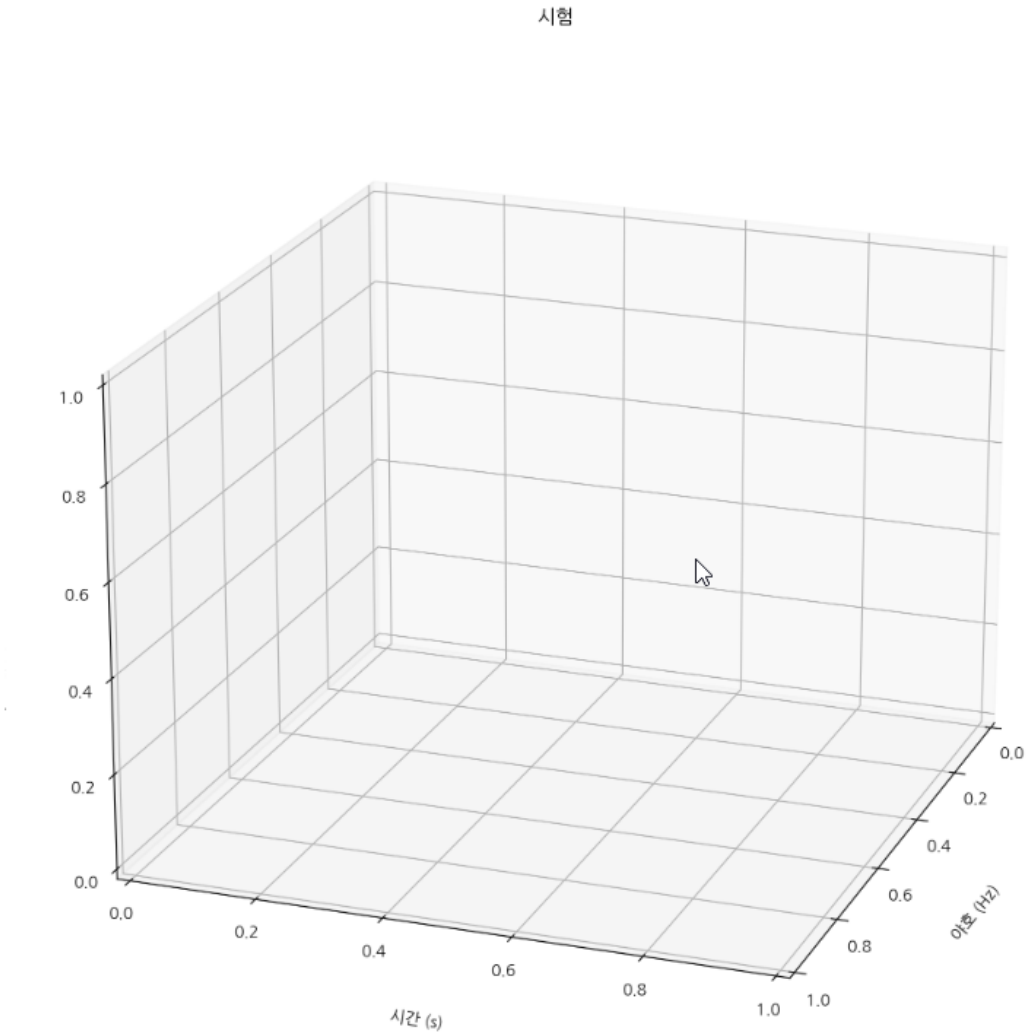

```
import numpy as np
import matplotlib.pyplot as plt

plt.plot(range(-20, 80), np.random.randn(100).cumsum(), 'r--')
plt.title('한글 점검');
```



Copyright 2023. Dongyang Mirae University all rights reserved.

Colab matplotlib 한글 지원 테스트3

```
fig = plt.figure(figsize=(15, 12))
fig, ax = plt.subplots(subplot_kw=dict(projection='3d'),
figsize=(15, 12))

graph_title = '시험'
ax.set_title(f'{graph_title}')
ax.set_xlabel('야호 (Hz)', labelpad=20)
ax.set_ylabel('시간 (s)', labelpad=20)
ax.set_zlabel('이건 뭐지 (dB/Hz)', labelpad=20)

ax.view_init(20, 20)
plt.show()
```



Copyright 2023. Dongyang Mirae University all rights reserved.

끝맺음

- 01. matplotlib 라이브러리 소개
- 02. plot 함수로 선 그래프 그리기
- 03. hist 함수로 히스토그램 그리기
- 04. boxplot 함수로 상자 그림 그리기

Copyright 2023. Dongyang Mirae University all rights reserved.

THANK YOU!
Q & A

- Name: 강환수
- Office: 동양미래대학교 2호관 706호 (02-2610-1941)
- E-mail: hsknag@dongyang.ac.kr
- Homepage: https://github.com/ai7dnn/2023-DA

Copyright 2023. Dongyang Mirae University all rights reserved.