

데이터분석입문

Lecture 05. matplotlib 라이브러리로 그래프 그리기

동양미래대학교
인공지능소프트웨어학과
강 환수

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

01. matplotlib 라이브러리 소개

02. plot 함수로 선 그래프 그리기

03. hist 함수로 히스토그램 그리기

04. boxplot 함수로 상자 그림 그리기

❖ matplotlib 라이브러리란? (1/7)

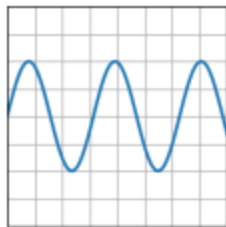
- 파이썬으로 데이터를 시각화할 때 가장 많이 사용하는 라이브러리 중 하나
- 2D 형태의 그래프, 이미지 등을 그릴 때 사용
- 실제 과학 컴퓨팅 연구 분야나 인공지능 연구 분야에서도 많이 활용

matplotlib

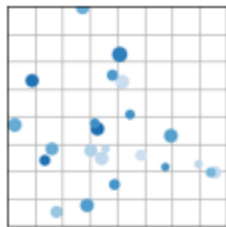
❖ matplotlib 라이브러리란? (2/7)

- 어떤 그래프들을 그릴 수 있나요?

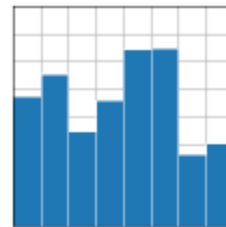
Basic Plot Types



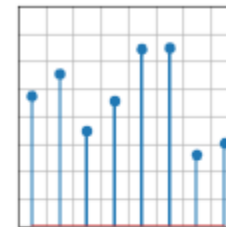
`plot(x, y)`



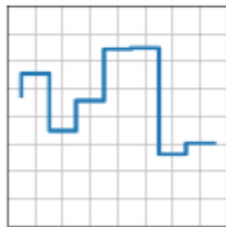
`scatter(x, y)`



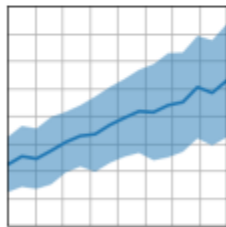
`bar(x, height) / barh(y, width)`



`stem(x, y)`



`step(x, y)`

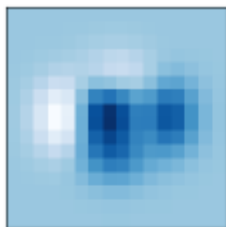


`fill_between(x, y1, y2)`

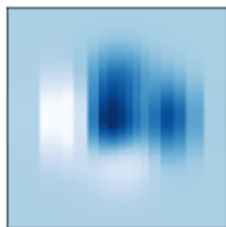
❖ matplotlib 라이브러리란? (3/7)

- 어떤 그래프들을 그릴 수 있나요?

Plots of Arrays and Fields



`imshow(Z)`



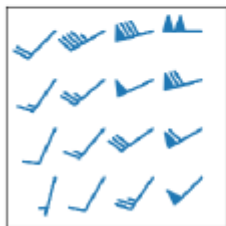
`pcolormesh(X, Y, Z)`



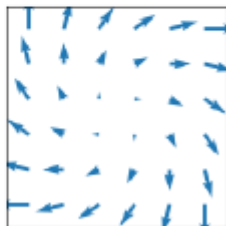
`contour(X, Y, Z)`



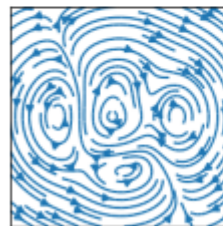
`contourf(X, Y, Z)`



`barbs(X, Y, U, V)`



`quiver(X, Y, U, V)`

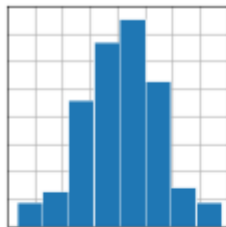


`streamplot(X, Y, U, V)`

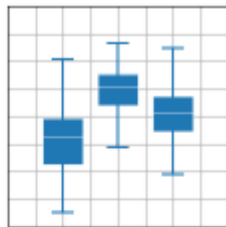
❖ matplotlib 라이브러리란? (4/7)

- 어떤 그래프들을 그릴 수 있나요?

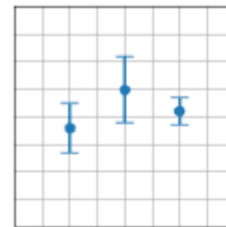
Statistics Plots



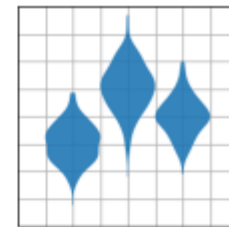
hist(x)



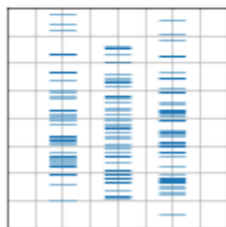
boxplot(X)



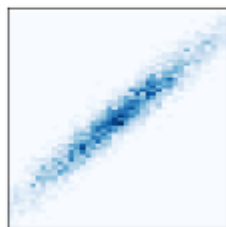
errorbar(x, y, yerr, xerr)



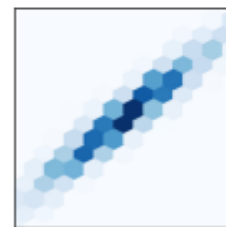
violinplot(D)



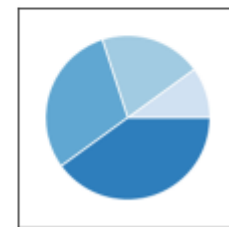
eventplot(D)



hist2d(x, y)



hexbin(x, y, C)

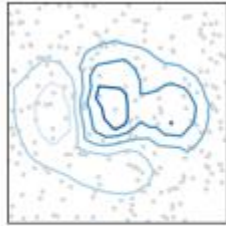


pie(x)

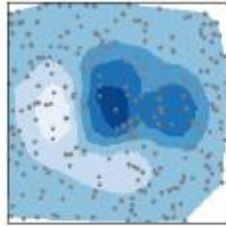
❖ matplotlib 라이브러리란? (5/7)

- 어떤 그래프들을 그릴 수 있나요?

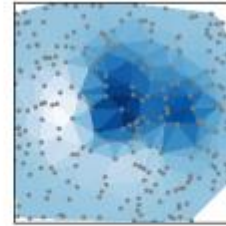
Unstructured Coordinates



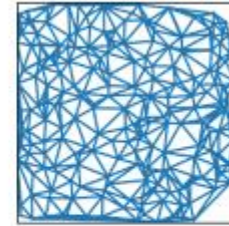
`tricontour(x, y, z)`



`tricontourf(x, y, z)`



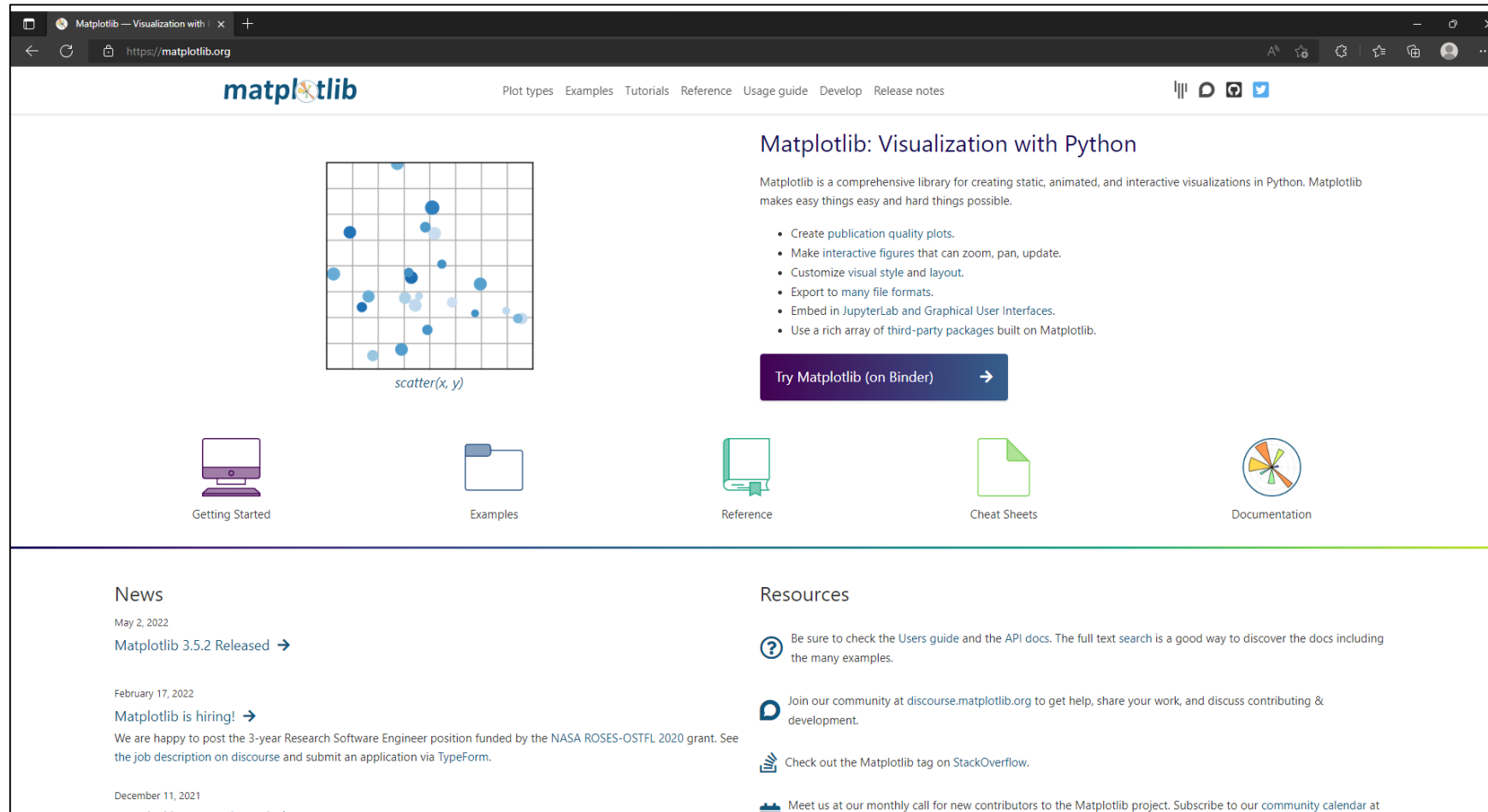
`tripcolor(x, y, z)`



`triplot(x, y)`

❖ matplotlib 라이브러리란? (6/7)

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



❖ matplotlib 라이브러리란? (7/7)

- matplotlib의 pyplot 모듈 불러오기

```
import matplotlib.pyplot
```

```
from matplotlib import pyplot
```

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

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

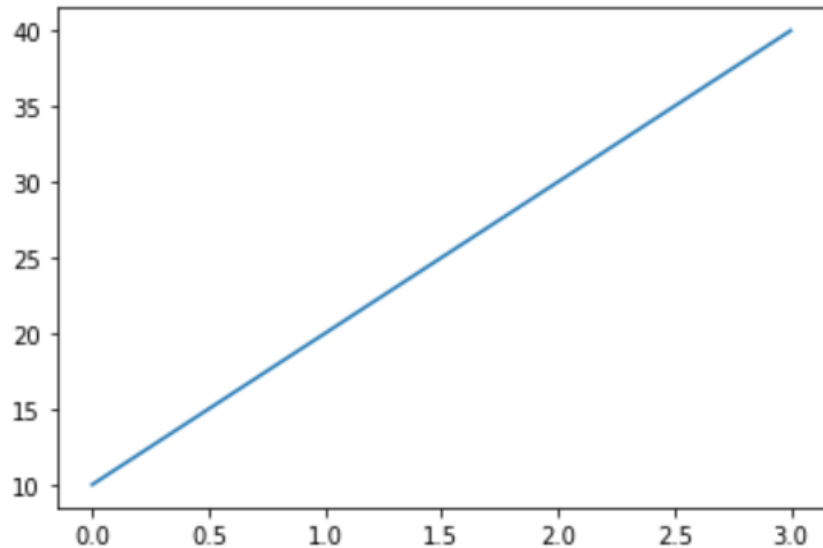
```
from matplotlib import pyplot as plt
```

02. plot 함수로 선 그래프 그리기

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

❖ 선 그래프 그리기: ① plt.plot([y축 데이터])

```
import matplotlib.pyplot as plt  
  
plt.plot([10, 20, 30, 40])  
plt.show()
```

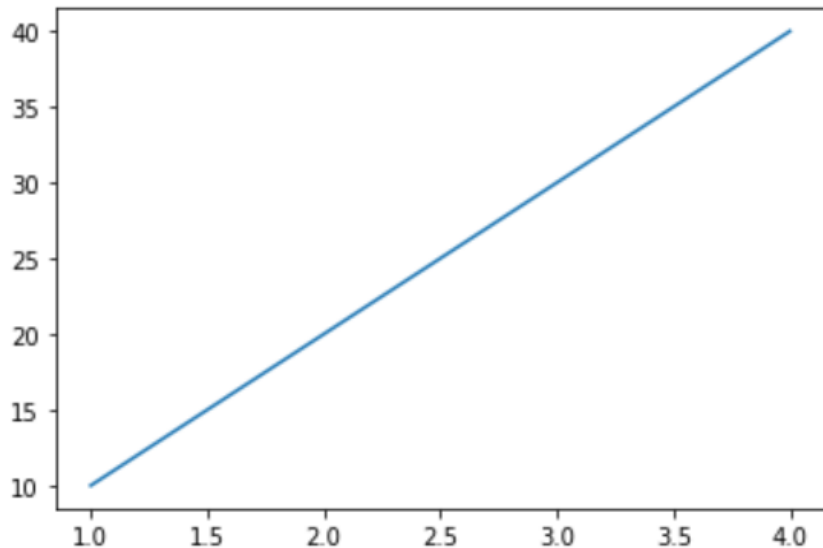


현재 plt.plot()에 입력된 리스트 값은 y축 값입니다.
x축 값도 입력하고 싶다면 어떻게 해야 할까요?

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.

❖ 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()
```

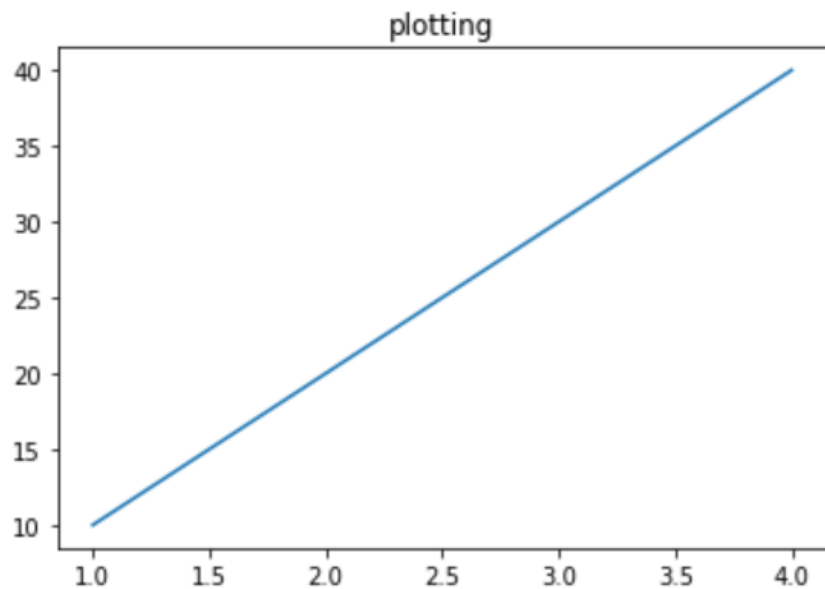
x축 데이터와 y축 데이터의 개수가 같지 않으면
오류가 발생합니다.

~#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)
```

❖ 그래프에 제목(Title) 추가하기

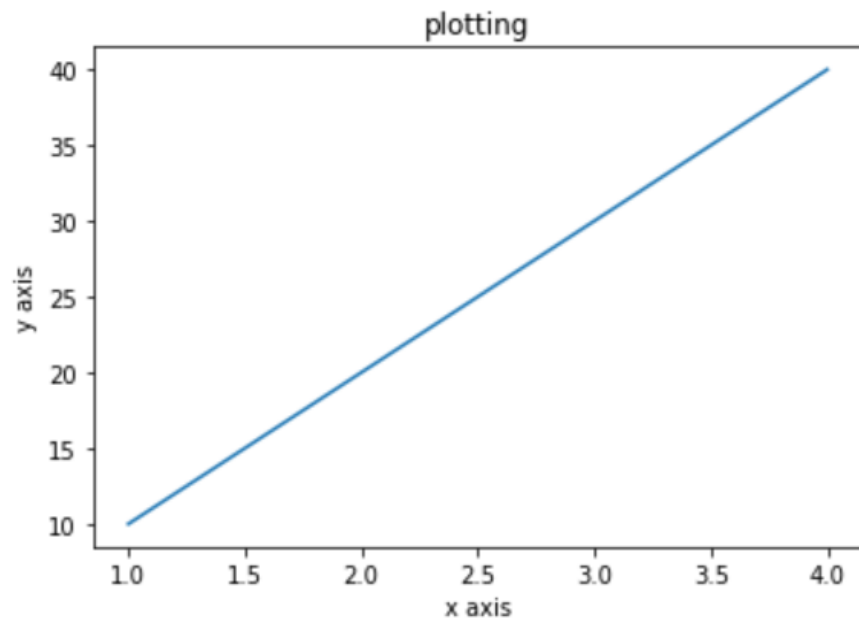
```
import matplotlib.pyplot as plt  
  
plt.plot([1, 2, 3, 4], [10, 20, 30, 40])  
plt.title('plotting')  
plt.show()
```



❖ 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()
```

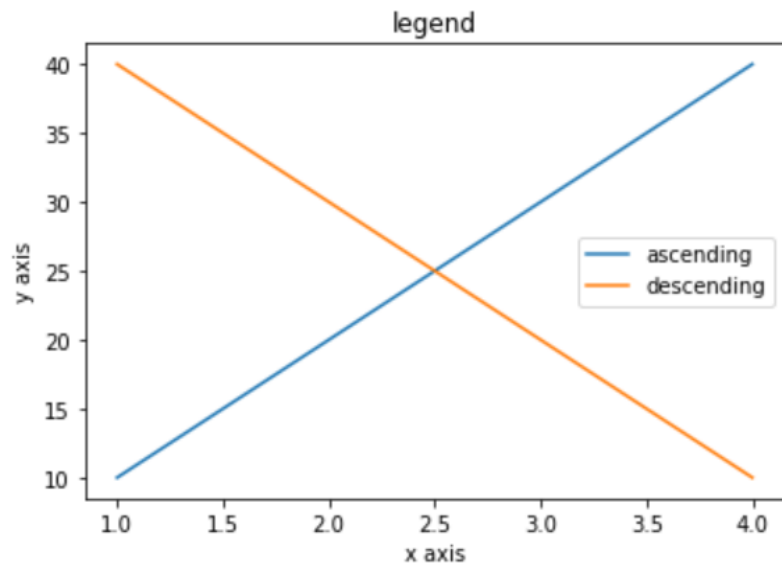


다른 사람이 내가 만든 그래프를 보았을 때,
한 번에 이해할 수 있게
x, y축 레이블 정보를 추가하도록 합시다

❖ 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()
```



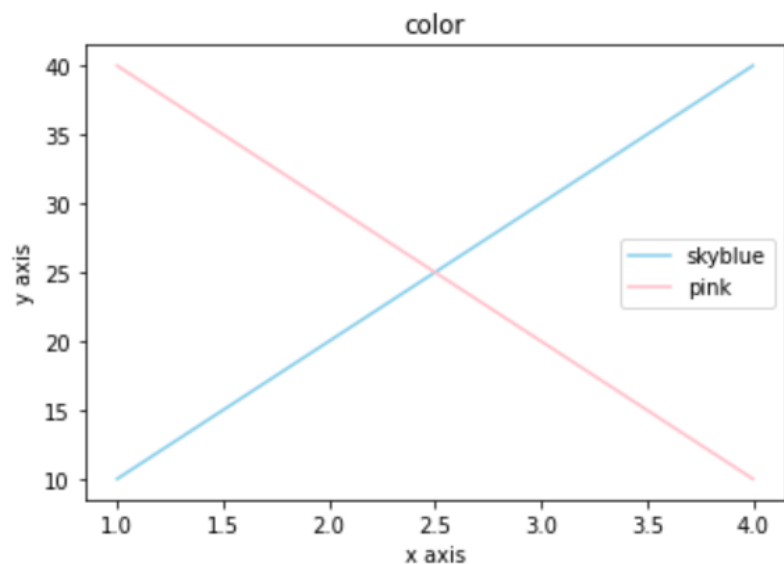
- legend
(명사) 전설
(명사) 범례

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

❖ 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()
```



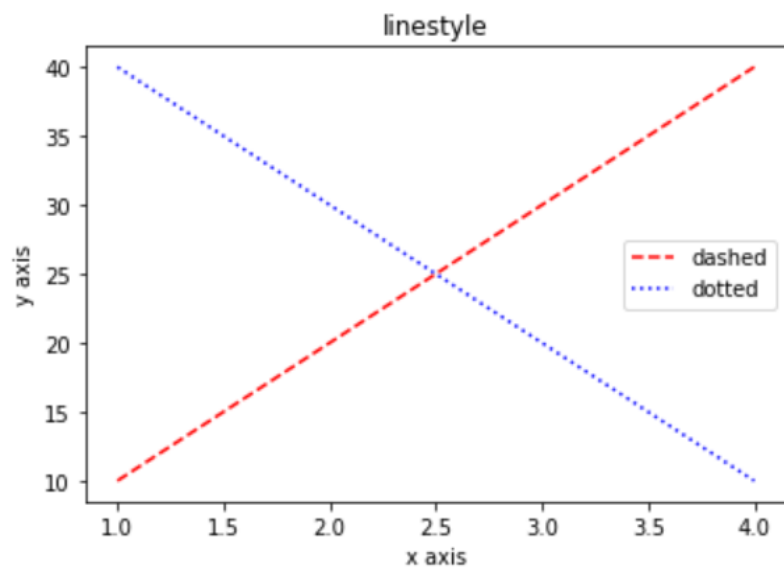
아래 색상에 대해서는 약자로 표기할 수 있습니다.

- red → r
- green → g
- blue → b
- black → k
- yellow → y

❖ 그래프 선 모양 바꾸기

```
import matplotlib.pyplot as plt

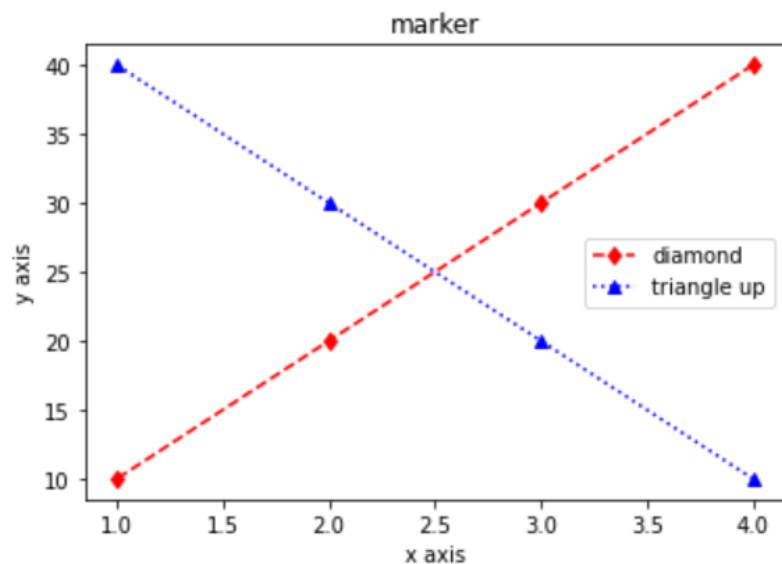
plt.plot([1, 2, 3, 4], [10, 20, 30, 40], color='r', linestyle='--', label='dashed')
plt.plot([1, 2, 3, 4], [40, 30, 20, 10], 'b', ls=':', label='dotted')
plt.title('linestyle')
plt.xlabel('x axis')
plt.ylabel('y axis')
plt.legend()
plt.show()
```



❖ 그래프 마커(Marker) 모양 바꾸기

```
import matplotlib.pyplot as plt

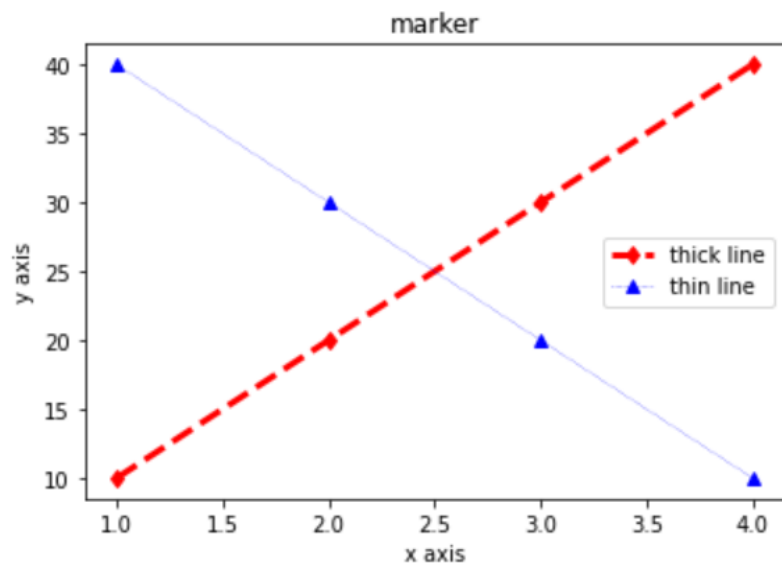
plt.plot([1, 2, 3, 4], [10, 20, 30, 40], color='r', linestyle='--', marker='d', label='diamond')
plt.plot([1, 2, 3, 4], [40, 30, 20, 10], 'b', ls=':', marker='^', label='triangle up')
plt.title('marker')
plt.xlabel('x axis')
plt.ylabel('y axis')
plt.legend()
plt.show()
```



❖ 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()
```

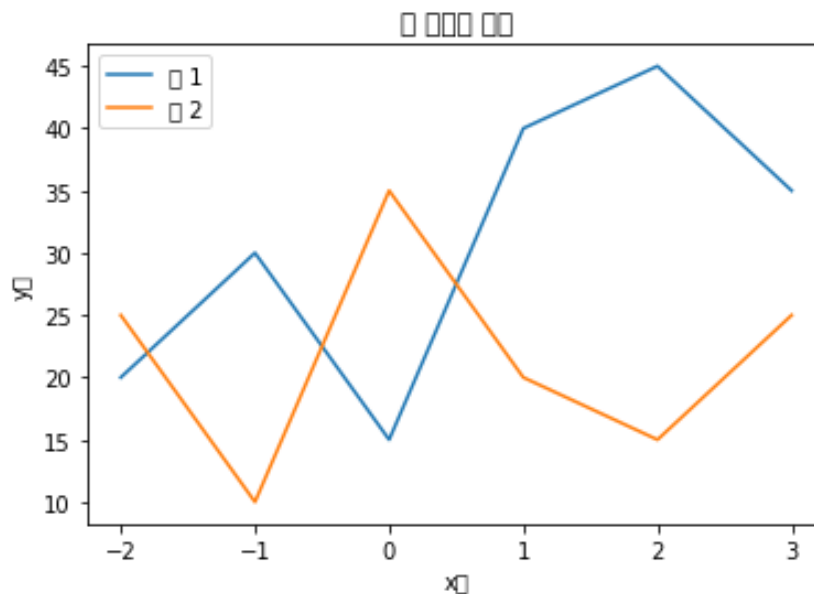


❖ 한글 폰트 사용하기 (1/5)

- 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()
```



❖ 한글 폰트 사용하기 (2/5)

- Malgun Gothic은 "맑은 고딕"입니다.

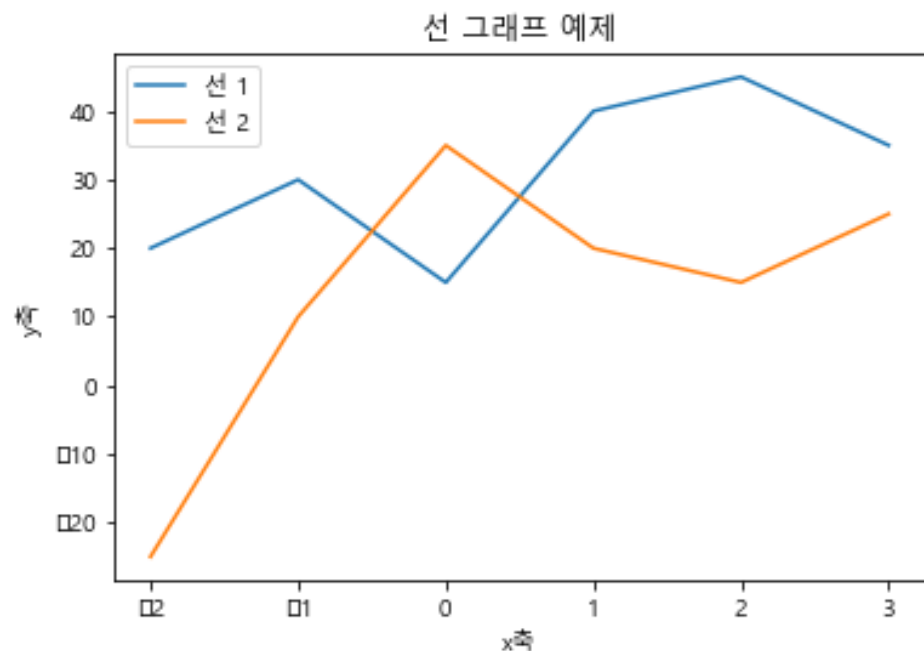
```
plt.rc('font', family='Malgun Gothic')  
plt.title('선 그래프 예제')
```

만약 macOS 운영체제를 사용하고 있다면
"AppleGothic"이라고 쓰세요.

❖ 한글 폰트 사용하기 (3/5)

```
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.

❖ 한글 폰트 사용하기 (4/5)

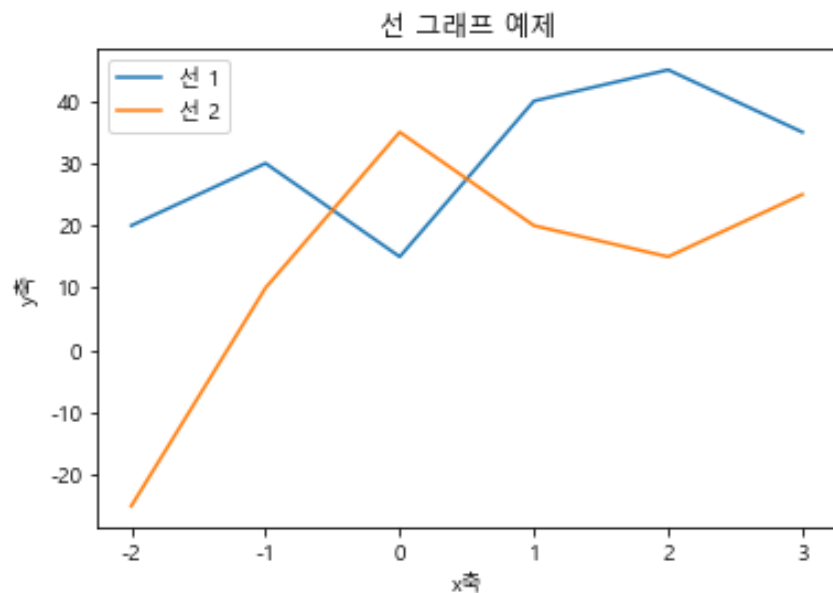
- 한글 폰트 사용시 마이너스 부호 표현하기

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

❖ 한글 폰트 사용하기 (5/5)

```
import matplotlib.pyplot as plt

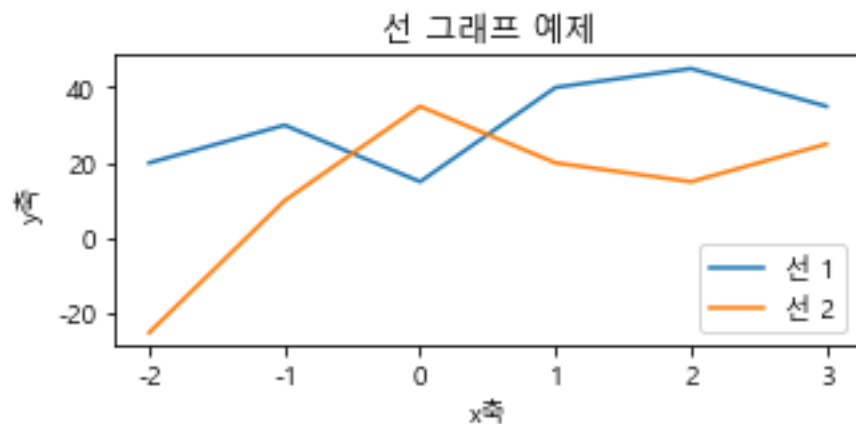
plt.rc('font', family='Malgun Gothic')
plt.rcParams['axes.unicode_minus'] = False
plt.plot([-2, -1, 0, 1, 2, 3], [20, 30, 15, 40, 45, 35], label='선 1')
plt.plot([-2, -1, 0, 1, 2, 3], [-25, 10, 35, 20, 15, 25], label='선 2')
plt.title('선 그래프 예제')
plt.xlabel('x축')
plt.ylabel('y축')
plt.legend()
plt.show()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



❖ 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

❖ 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'

```
✓ 6초 [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
```

```
✓ 0초 [2] %config InlineBackend.figure_format = 'retina'
```

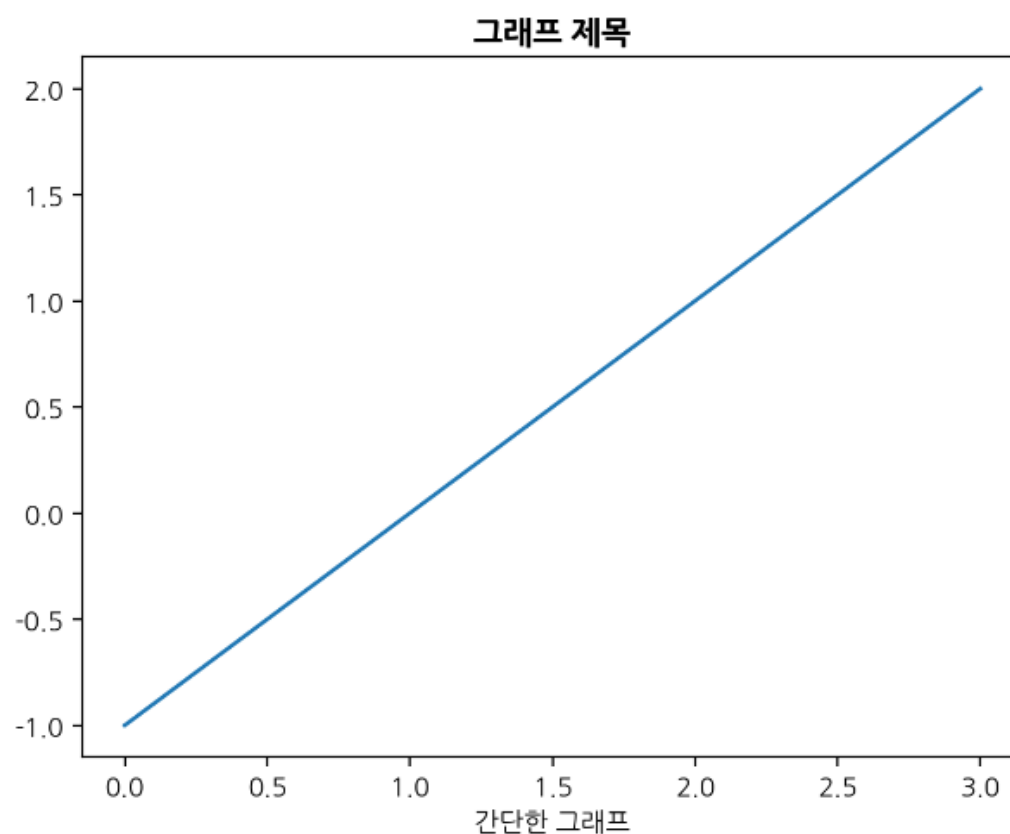
3. 바로 실행 테스트

```
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()
```



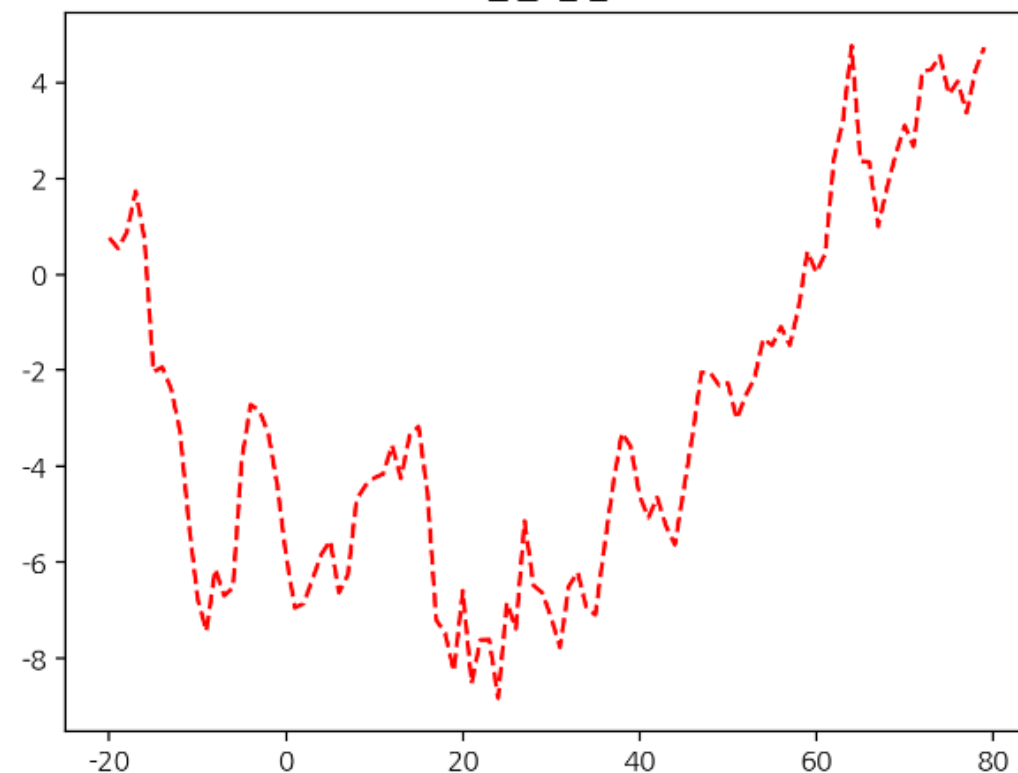
```
import numpy as np  
import matplotlib.pyplot as plt
```

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

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



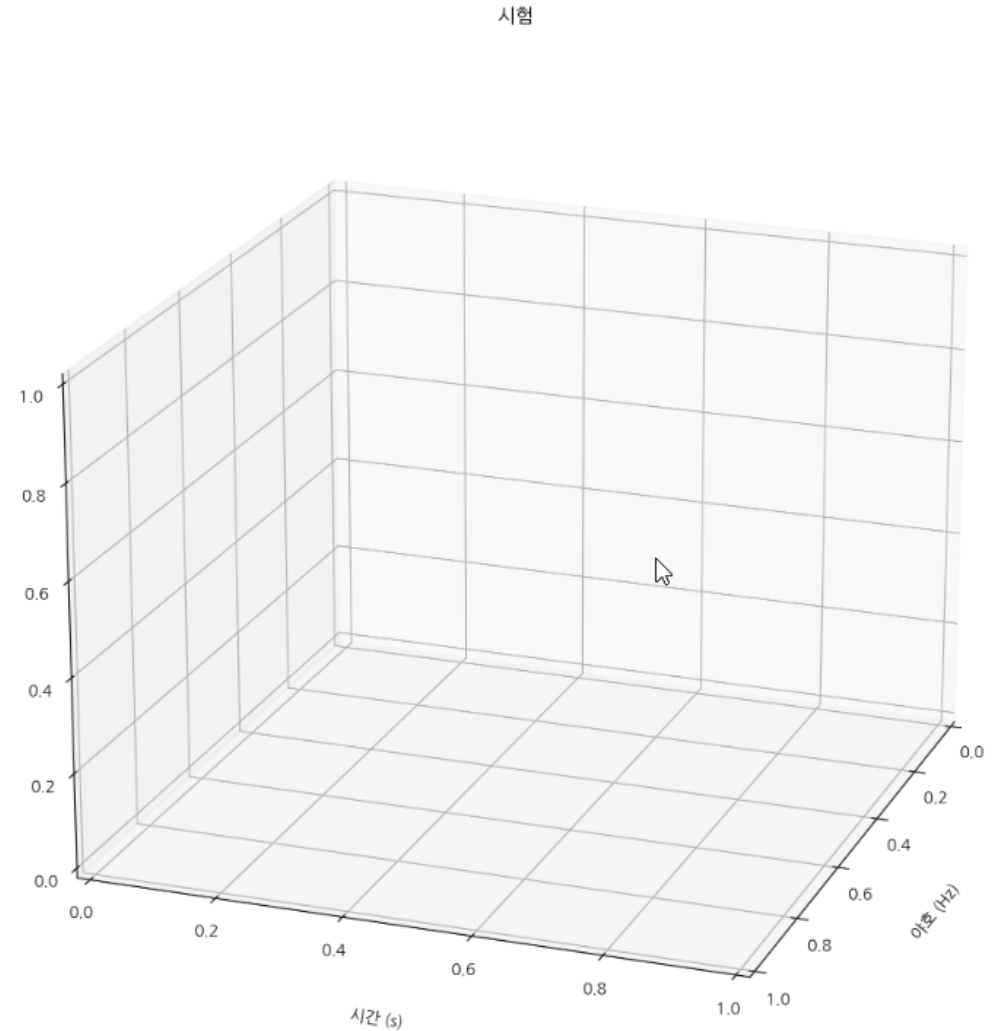
한글 점검



```
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()
```



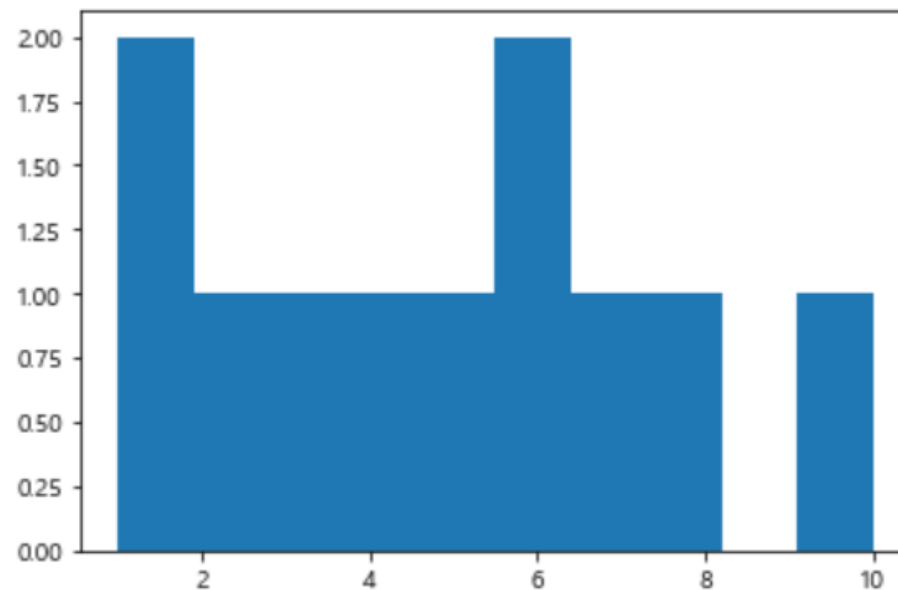
03. hist 함수로 히스토그램 그리기

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

❖ 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()
```



❖ 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.

❖ 히스토그램 (3/6)

- 1과 6사이의 랜덤 숫자 만들기

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

❖ 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.

❖ 히스토그램 (5/6)

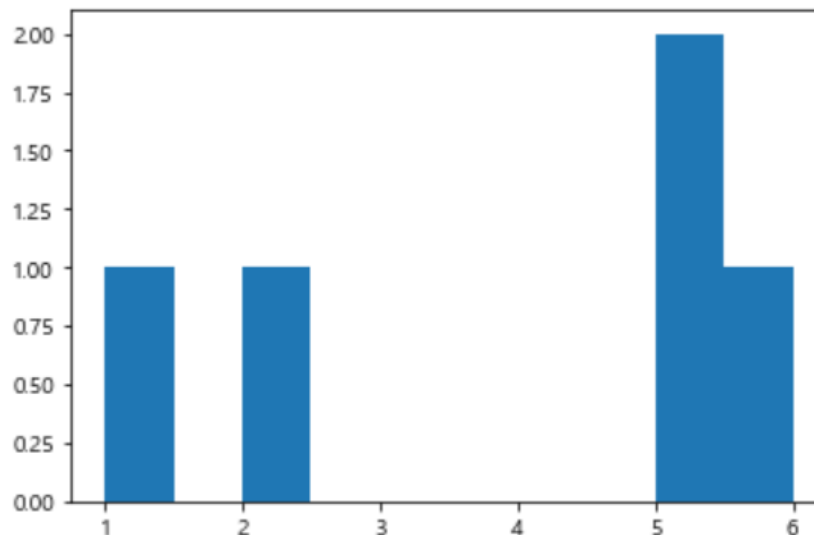
- 주사위 시뮬레이션 5회 수행 결과를 히스토그램으로 시각화하기

```
import random
import matplotlib.pyplot as plt

dice = []

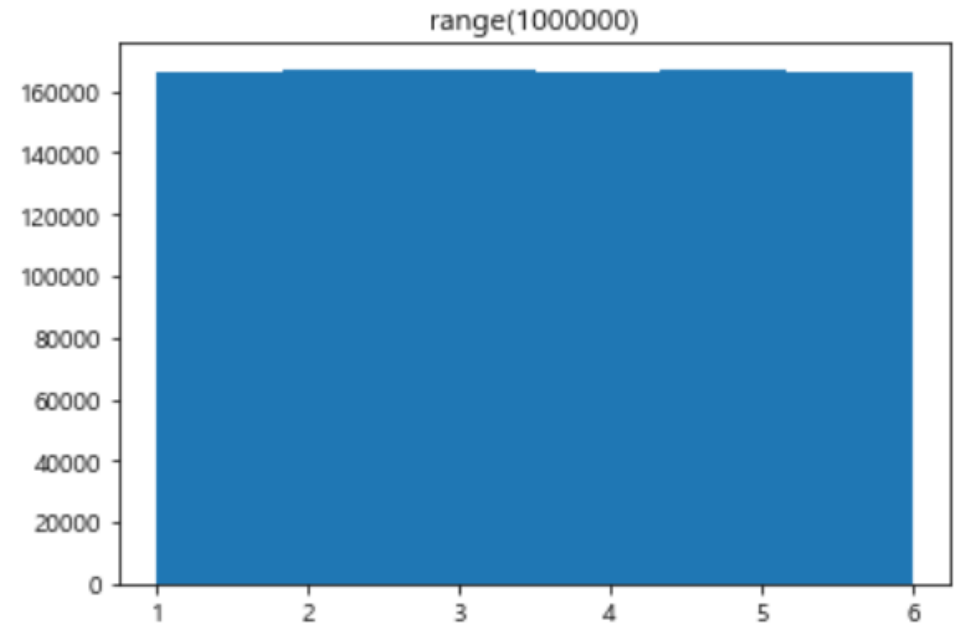
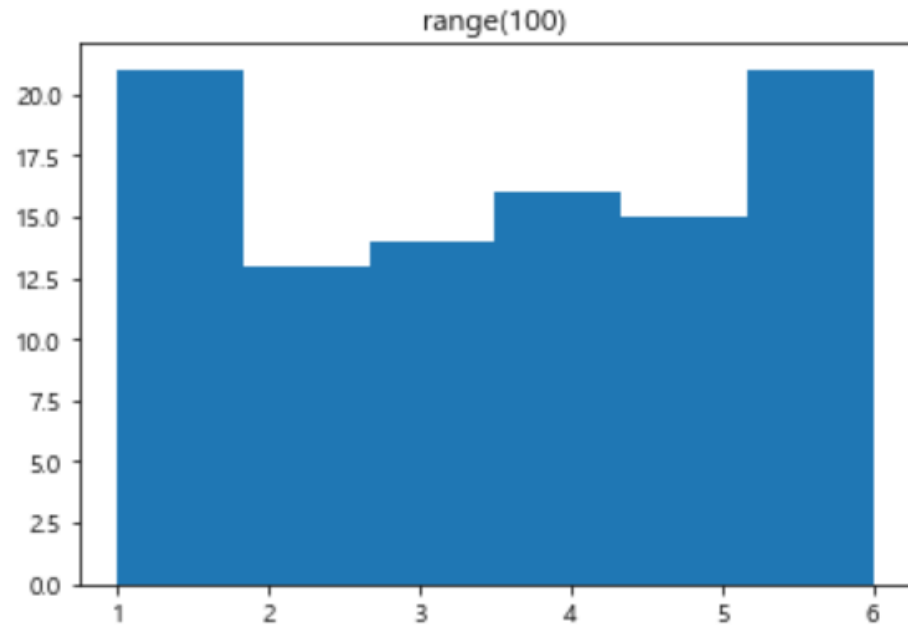
for i in range(5):
    dice.append(random.randint(1, 6))

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



❖ 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

04. boxplot 함수로 상자 그림 그리기

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

❖ 상자 그림 (1/3)

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

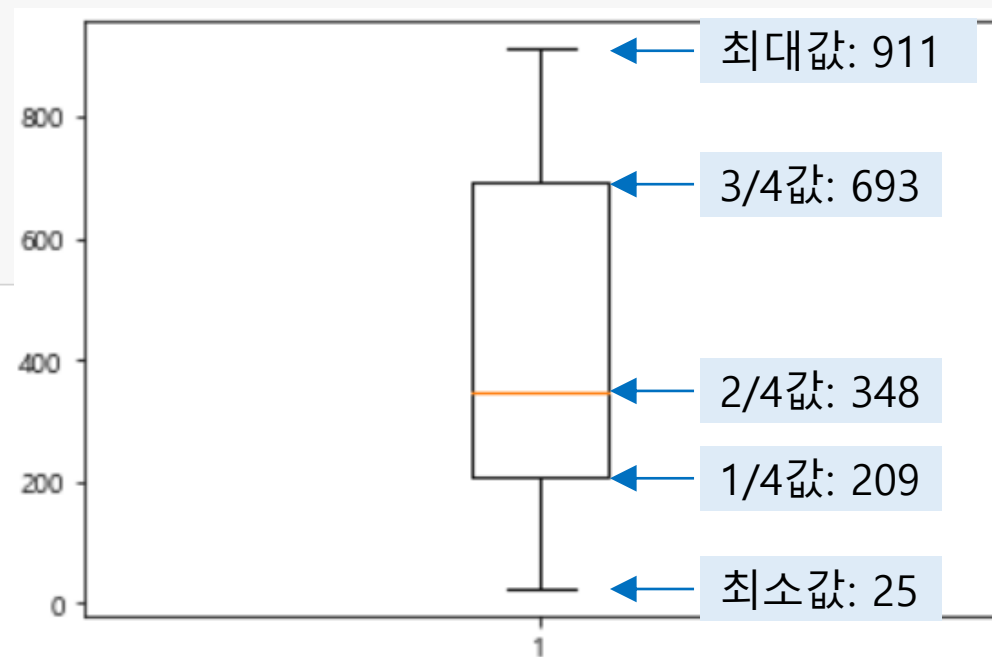
```
import matplotlib.pyplot as plt
import random

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]



❖ 상자 그림 (2/3)

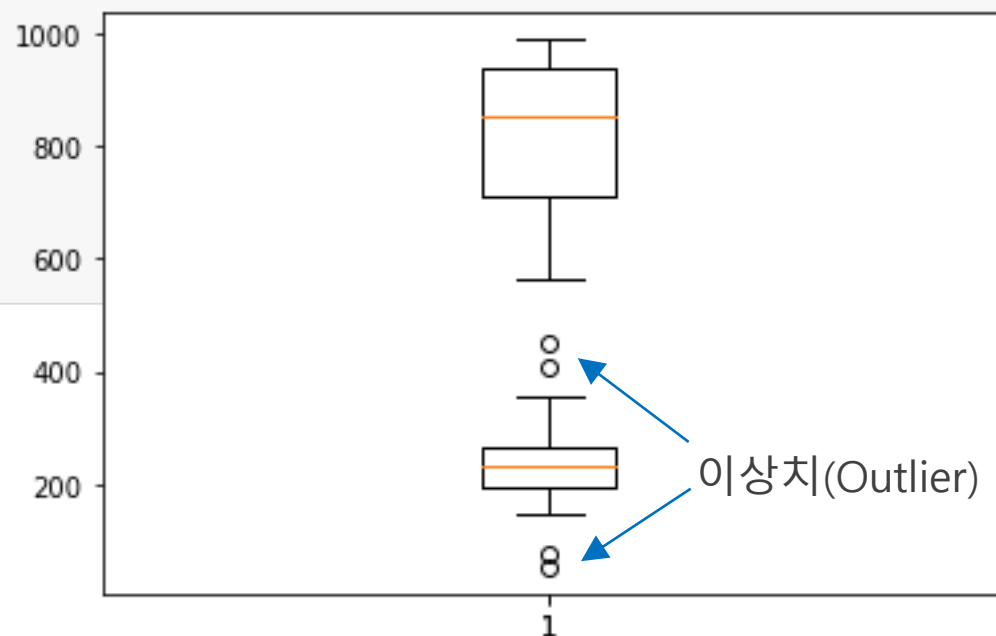
- 하나의 그래프에서 두 데이터의 상자 그림 그리기

```
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?



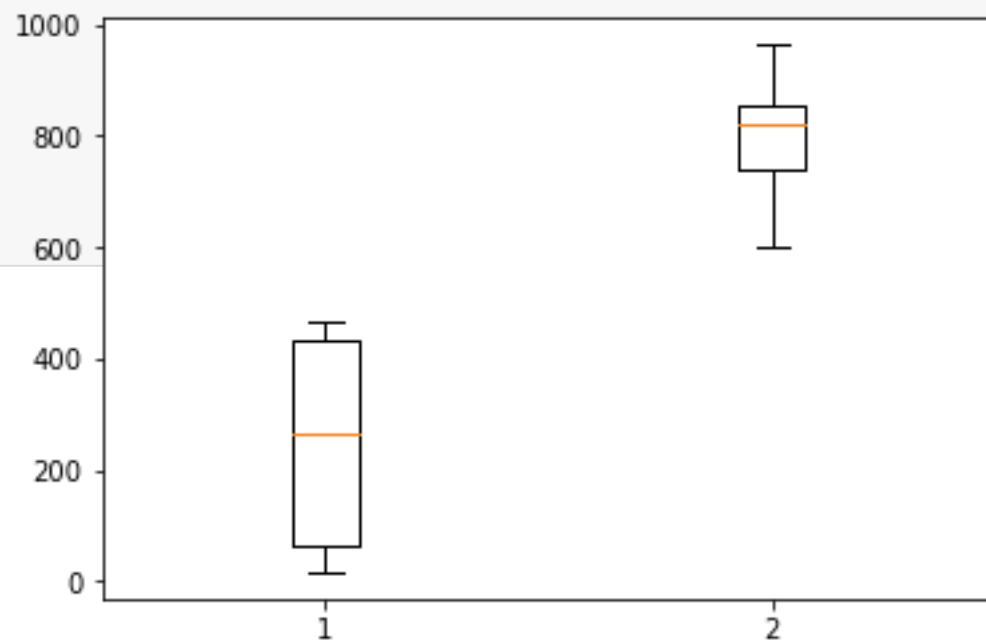
❖ 상자 그림 (3/3)

- 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()
```



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

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>