



Introduction To Data Analysis

# 데이터 분석 입문

## Lecture 10. matplotlib 라이브러리로 그래프 그리기

인공지능소프트웨어학과 강환수 교수

## 학습개요

- ✓ Matplotlib, numpy 설치 확인
- ✓ 그래프의 제목, 레이블, 범례, x축, y축 최대 최소 지정
- ✓ 선의 스타일, 색상, 굵기, 마커의 스타일, 색상



## 학습목표

- ✓ Matplotlib에서 그래프의 제목, 레이블, 범례, x축, y축 최대 최소를 지정해 그래프를 그릴 수 있다.
- ✓ 선의 스타일, 색상, 굵기, 마커의 스타일, 색상 등을 수정해 그래프를 그릴 수 있다.

LESSON 01

# plot 함수로 선 그래프 그리기



## ◦ Matplotlib 설치 확인

```
pip show matplotlib
```

```
Name: matplotlib  
Note: you may need to restart the kernel to use updated packages.
```

```
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:\ProgramData\anaconda3-23-09\Lib\site-packages
```

```
Requires: contourpy, cycler, fonttools, kiwisolver, numpy, packaging, pillow, pyparsing, python-dateutil
```

```
Required-by: seaborn
```

## → Numpy 설치 확인

```
pip show numpy
```

Name: numpy

Version: 1.24.3

Summary: Fundamental package for array computing in Python

Home-page: <https://www.numpy.org>

Author: Travis E. Oliphant et al.

Author-email:

License: BSD-3-Clause

Location: C:\ProgramData\anaconda3-23-09\Lib\site-packages

Requires:

Required-by: astropy, bokeh, Bottleneck, contourpy, daal4py, datasets, datashader, datashape, gensim, h5py, holoviews, hvplot, imagecodecs, imageio, imbalanced-learn, matplotlib, mkl-fft, mkl-random, numba, numexpr, pandas, patsy, pyarrow, pyerfa, PyWavelets, scikit-image, scikit-learn, scipy, seaborn, statsmodels, tables, tifffile, transformers, xarray

Note: you may need to restart the kernel to use updated packages.

## — 버전 확인 및 matplotlib 설정

```
import matplotlib as mpl  
mpl.__version__
```

'3.7.2'

```
import numpy as np  
np.__version__
```

'1.24.3'

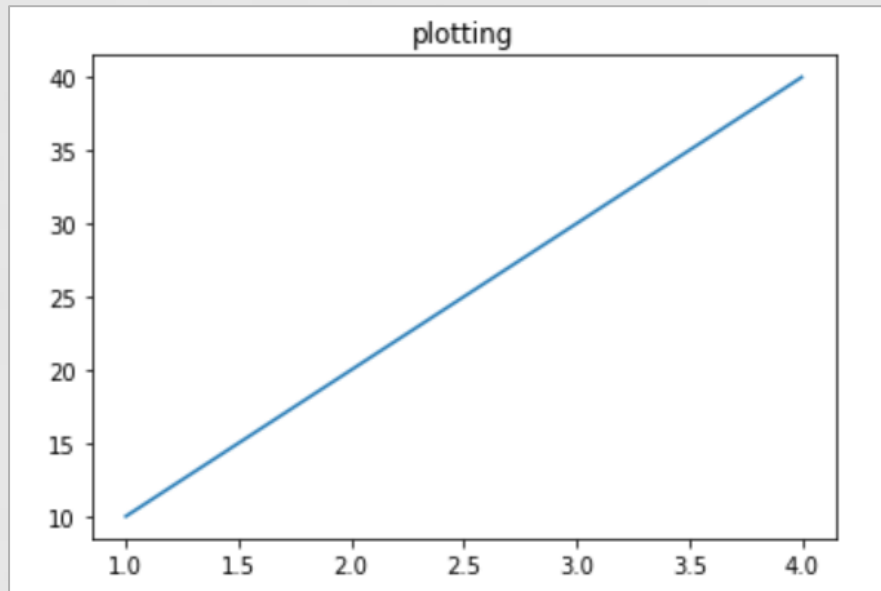
**matplotlib 선명하게**

```
%config InlineBackend.figure_format = 'retina' I
```

## — 그래프에 제목(Title) 추가하기

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

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



## → 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)
```

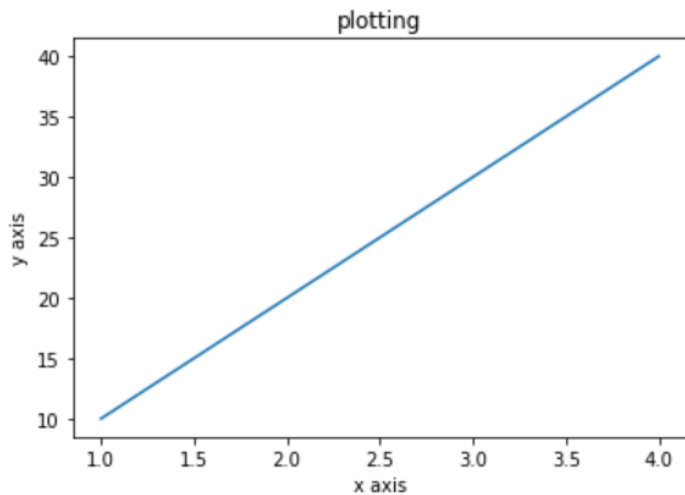
x축 데이터와 y축 데이터의  
개수가 같지 않으면 오류가 발생합니다.



## — 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.

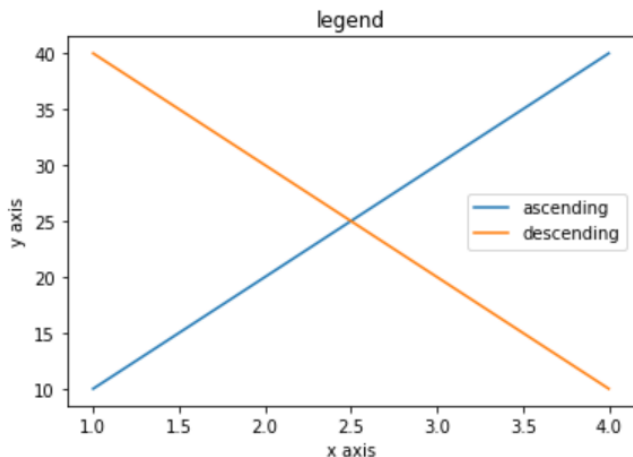
## — 그래프에 범례(legend) 추가하기

## + Legend : (명사) 전설, (명사) 범례

```
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()
```

속성 label='범례에\_기술되는\_이름'



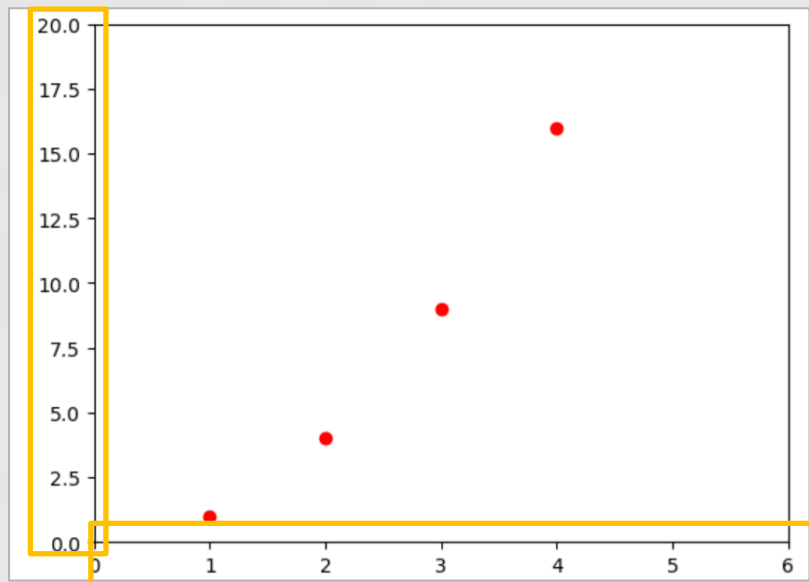
경우에 따라서 그래프를 출력할 때  
흑백으로 출력하는 경우가 있습니다.  
이 경우 색(Color)으로 범례를 구분하는  
것이 어려울 수 있습니다.

## — 색상과 스타일, 축의 최대 최소 지정

```
import matplotlib.pyplot as plt

plt.plot([1, 2, 3, 4], [1, 4, 9, 16], 'ro')
plt.axis((0, 6, 0, 20))
plt.show()
```

`plt.axis = [xmin, xmax, ymin, ymax]`



LESSON 02

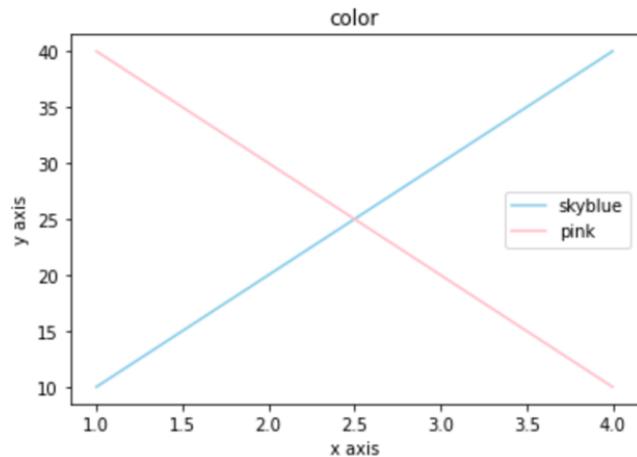
# 그래프의 다양한 속성 지정



## — 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

## → 그래프 선 모양 바꾸기 (1)

### 선 스타일 linestyle 또는 ls

- ✓ 실선(solid), 대시선(dashed)
- ✓ 점선(dotted), 대시-점선(dash-dot)

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

### 색상 color

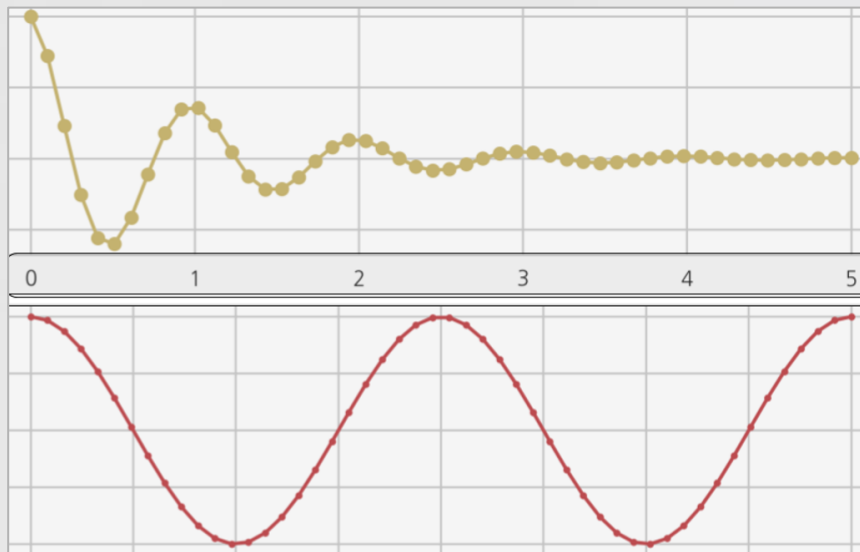
문자열	약자
blue	b
green	g
red	r
cyan	c
magenta	m
yellow	y
black	k
white	w

## → 그래프 선 모양 바꾸기 (2)

## 마커 marker

✓ 데이터 위치를 나타내는 기호

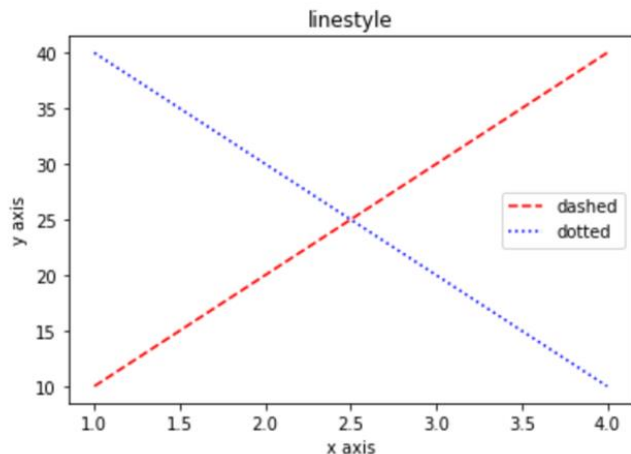
+ 실제 데이터를 돋보이게 그리기



## — 그래프 선 모양 바꾸기 (3)

```
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)

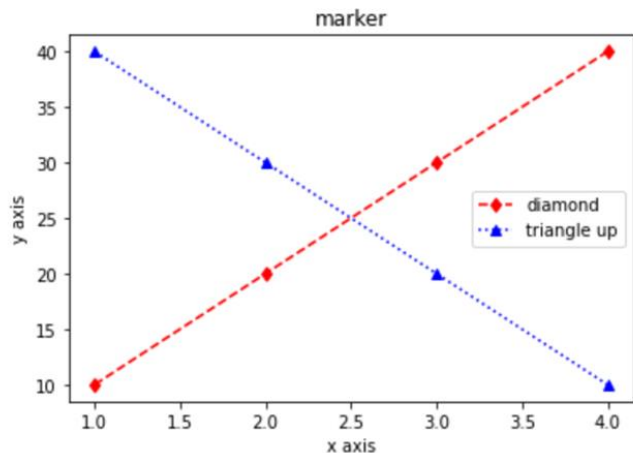
선 스타일 문자열	의미
-	solid line style
--	dashed line style
-.	dash-dot line style
:	dotted line style



## 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()
```



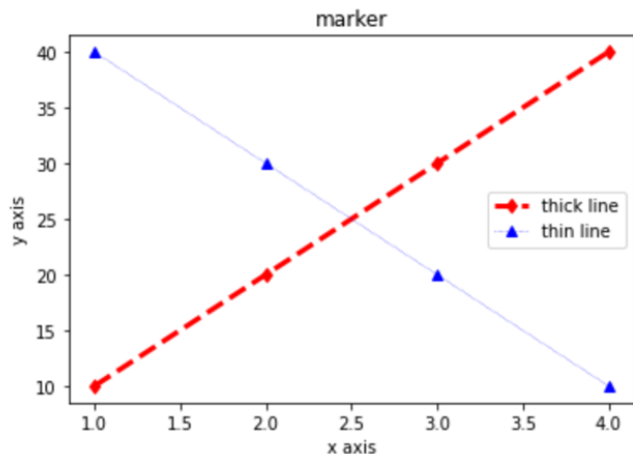
마커 문자열 의미

.	point marker	1	tri_down marker	h	hexagon1 marker
,	pixel marker	2	tri_up marker	H	hexagon2 marker
o	circle marker	3	tri_left marker	+	plus marker
v	triangle_down marker	4	tri_right marker	x	x marker
^	triangle_up marker	s	square marker	D	diamond marker
<	triangle_left marker	P	pentagon marker	d	thin_diamond marker
>	triangle_right marker	*	star marker		

## — 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()
```



## 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:'`

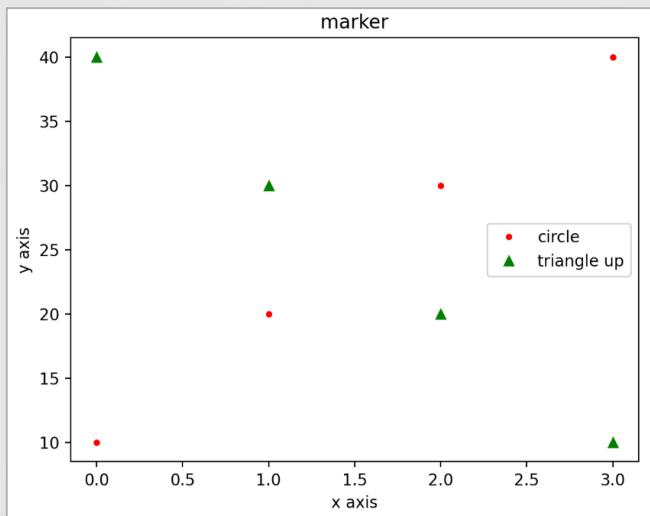
✓ # black triangle\_up markers connected by a dotted line

## — 라인스타일이 없으면 마커만 그림 (1)

```
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()
```

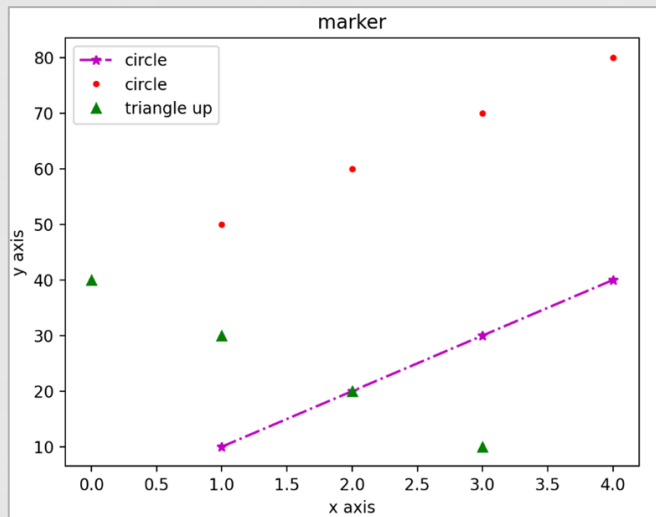


## — 라인스타일이 없으면 마커만 그림 (2)

```
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

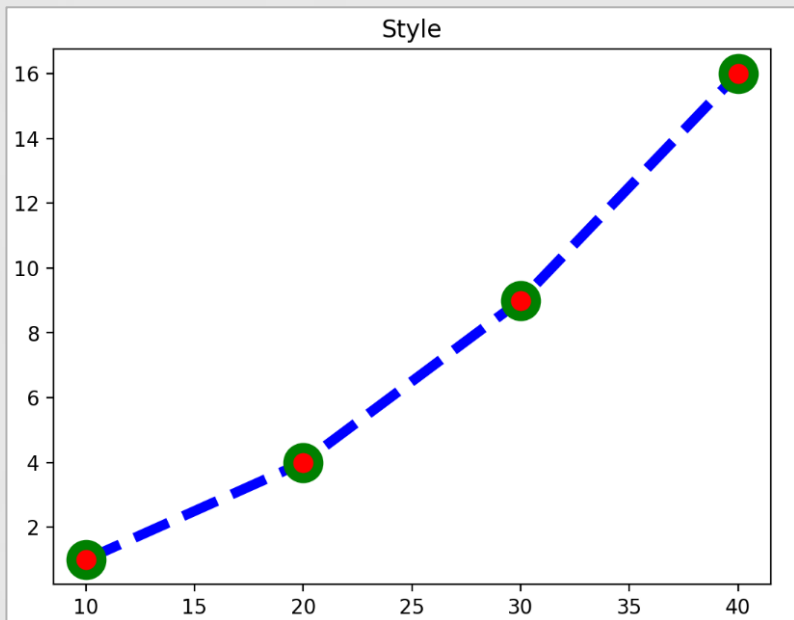
plt.title('marker') # 제목 설정
plt.xlabel('x axis')
plt.ylabel('y axis')
plt.legend() # 범례 표시
plt.show()
```



## 기타 속성 (1)

## 다양한 그림 속성 (1)

```
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()
```

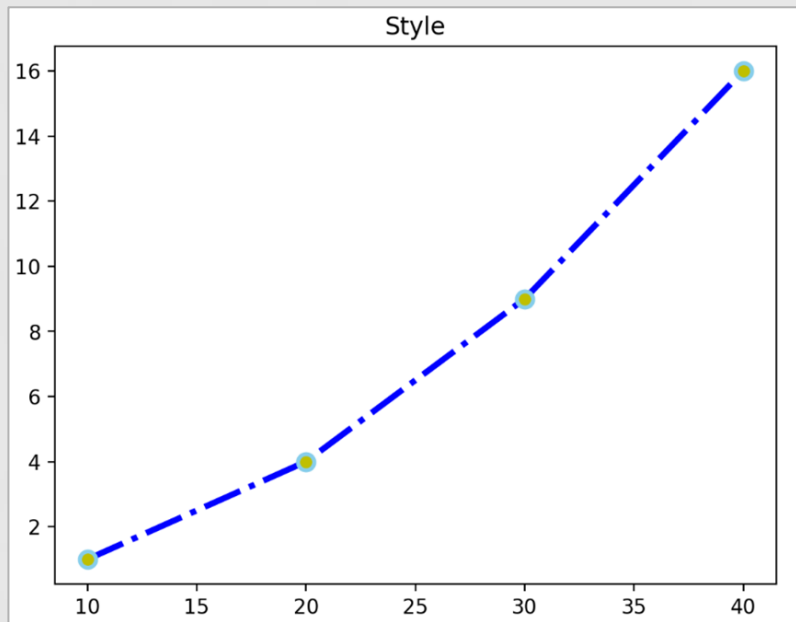


스타일 문자열	약자	의미
color	c	선 색깔
linewidth	lw	선 굵기
linestyle	ls	선 스타일
marker		마커 종류
markersize	ms	마커 크기
markeredgecolor	mec	마커 선 색깔
markeredgewidth	mew	마커 선 굵기
markerfacecolor	mfc	마커 내부 색깔

## 기타 속성 (2)

## 다양한 그림 속성 (2)

```
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

SUMMARY

# 학습정리





## ⚙ 제목, 축 제목 지정

➤ title() xlabel() ylabel()





## ⚙ 축의 최대 최소 지정, 범례 지정

➤ axis() legend()

## ⚙ 색상과 선 스타일

➤ k, r, g, b, m, y, w, c

➤ - solid, -- dashed, -. dash-dot, : dotted

	dashed
	dotted
	dashed dot
	solid line



## ⚙️ 다양한 속성

- c, color, linewidth, lw, linestyle, ls
- marker, ms, mec, mew, mfc

## ⚙️ 문자열 '[color][marker][line]'

- 선 색상, 마커 모양, 선 모양을 지정
- 선 모양이 없으면 마커만 그림

