파이썬을 이용한 시각화 기본

학습 목표

- matplotlib를 활용한 시각화에 대해 알아봅니다.

학습 내용

- 1. 라이브러리 불러오기
- 2. 그래프 그려보기
- 3. 여러 개의 그래프 그리기
- 4. 범례 표시 - 기본

목차

01. 라이브러리 불러오기

In [1]:

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

- 버전 확인 : **version**

In [2]:

```
print(matplotlib.__version__)
```

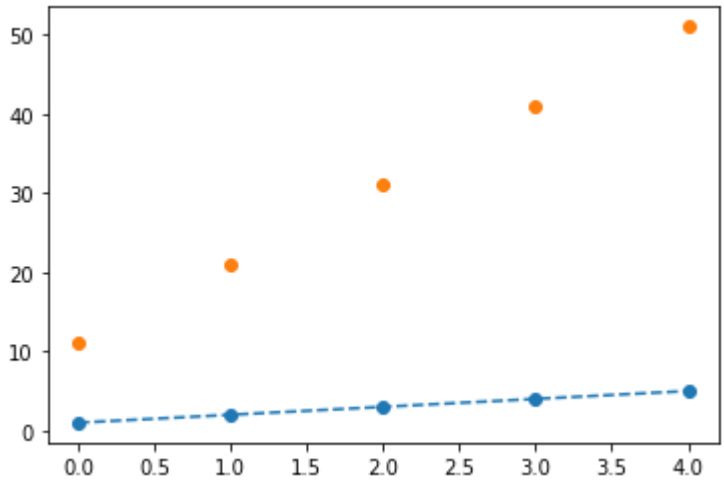
```
3.3.2
```

02. 그래프 그려보기

- 대부분의 Matplotlib 라이브러리에서 그래프를 그리는 것이 pyplot 서브 모듈 아래에 있다. 보통 약자로 plt를 이용합니다.
- 기본 plot() 함수로 그래프를 표시합니다.

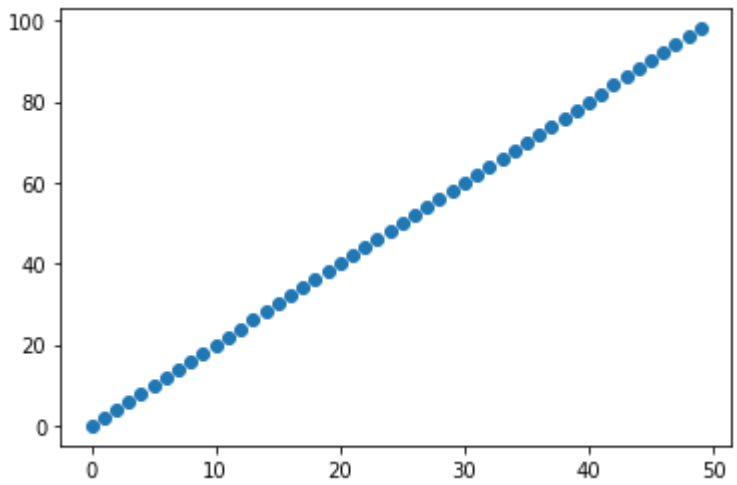
In [9]:

```
import matplotlib.pyplot as plt

# marker='o'(점), '-'(선)
# marker 생략 가능
plt.plot([1,2,3,4,5], marker='o', linestyle="--")  # '-' 선 : 기본값
plt.plot([11,21,31,41,51], "o")  # '-' 선 : 기본값
plt.show()
```



In [13]:

```
x = range(0, 50)
y = range(0, 100,2)
plt.plot(x, y, 'o')
plt.show()
```

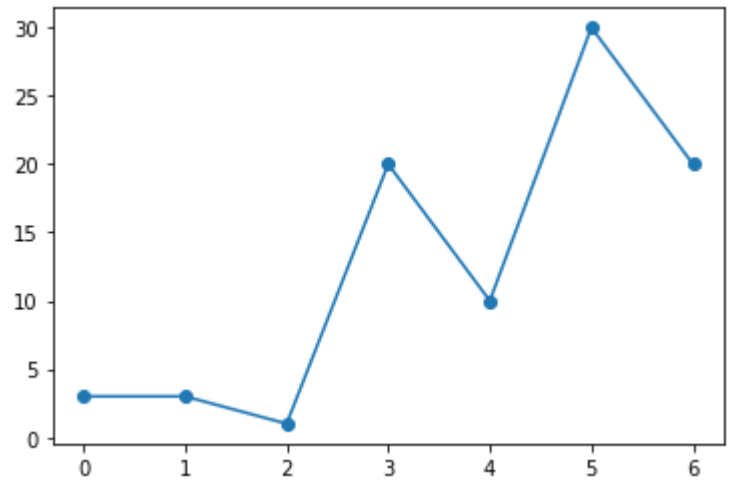


Matplotlib Markers

- marker 인자 키워드를 사용하여 특별한 점을 강조한다.

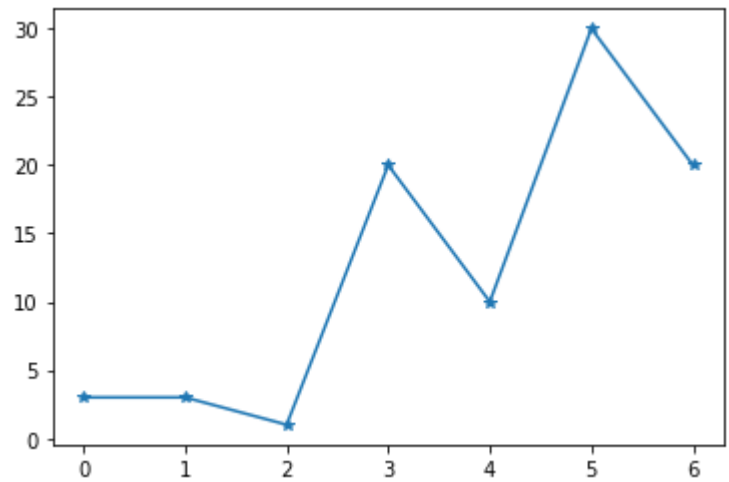
In [32]:

```
import numpy as np
ypoints = np.array([3, 3, 1, 20, 10, 30, 20])    # 7개의 y값 지정

plt.plot(ypoints, marker = 'o')
plt.show()
```



In [34]:

```
plt.plot(ypoints, marker = '*')
plt.show()
```



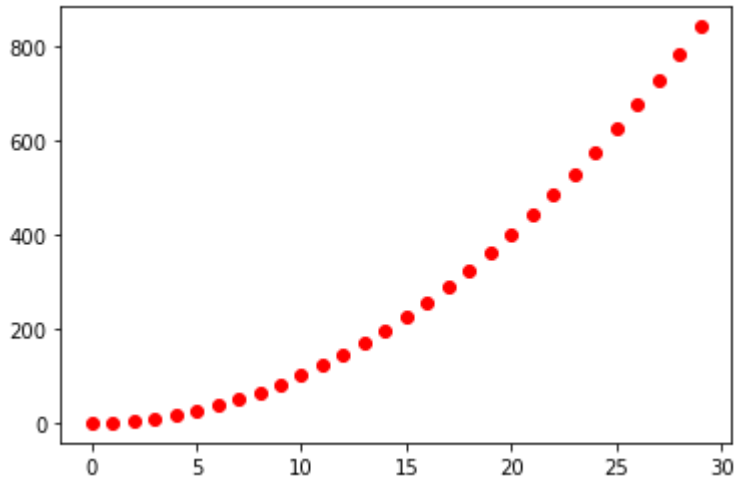
- Marker Reference (다양한 표현)
  - https://www.w3schools.com/python/matplotlib_markers.asp (https://www.w3schools.com/python/matplotlib_markers.asp)

y = x * x 의 그래프 그려보기

In [35]:

```
x = range(0, 30)
y = [ v * v for v in x ]
print(x)
print(y)

plt.plot(x, y, 'ro')  # r:빨간색, o:점
plt.show()
```

```
range(0, 30)
[0, 1, 4, 9, 16, 25, 36, 49, 64, 81, 100, 121, 144, 169, 196, 225, 25
6, 289, 324, 361, 400, 441, 484, 529, 576, 625, 676, 729, 784, 841]
```



- 'ro'에서 'r'은 red를 의미하고, 'o'는 그래프의 마커 모양을 의미한다.

matplot의 주요 색상

문자	색상
b	blue(파란색)
g	green(녹색)
r	red(빨간색)
c	cyan(청록색)
m	magenta(마젠타색)
y	yelow(노란색)
k	black(검은색)
w	white(흰색)

주요 마커

마커	의미
o	circle(원)
v	triangle_down(역 삼각형)
^	triangle_up(삼각형)
s	square(네모)
+	plus(플러스)
.	point(점)

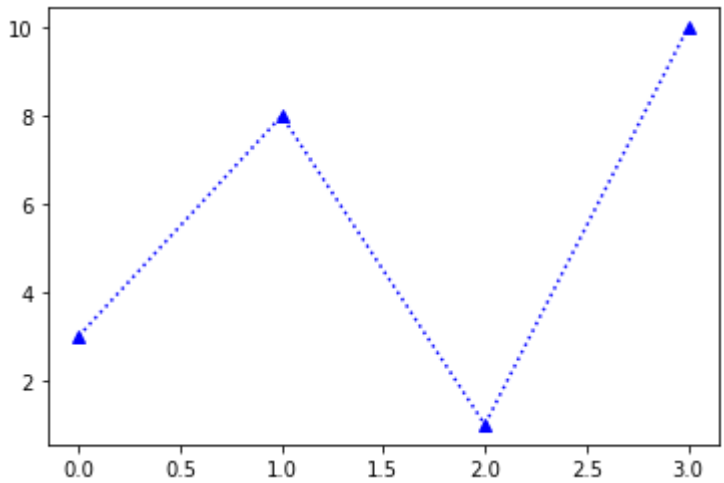
선 종류	설명
'-'	Solid line
':'	Dotted line
'--'	Dashed line
'-.'	Dashed/dotted line

fmt의 파라미터를 다음과 같이 사용

- marker | line | color

In [36]:

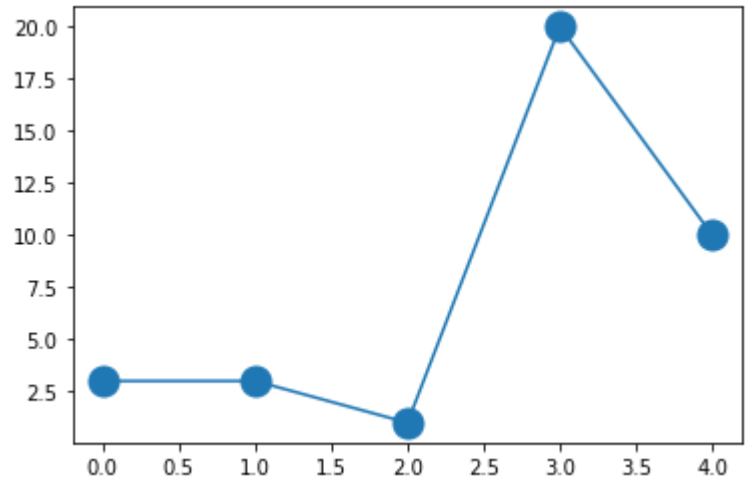
```
y = np.array([3, 8, 1, 10])

# ^:b => triangle_up(삼각형)| Dashed line | blue
plt.plot(y, '^:b')  #
plt.show()
```



마커 사이즈(ms)

In [37]:

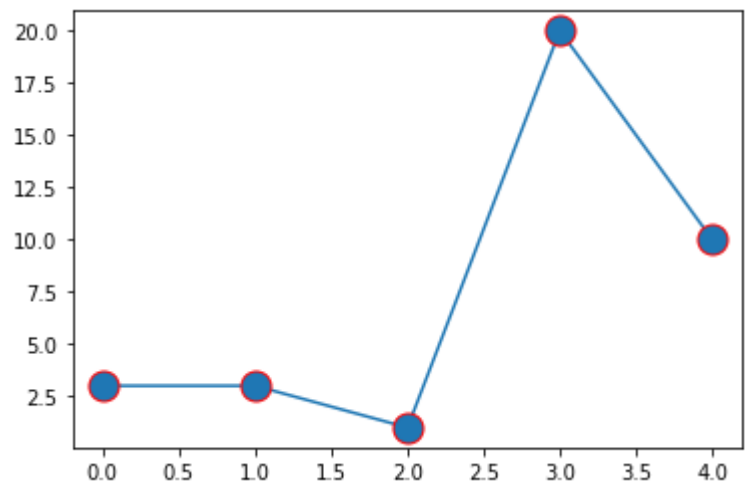
```
y = np.array([3, 3, 1, 20, 10])

plt.plot(y , 'o-', ms=15)   # o(marker), -(line)
plt.show()
```



마커 Edge 색 - marker edge color(mec)

In [39]:

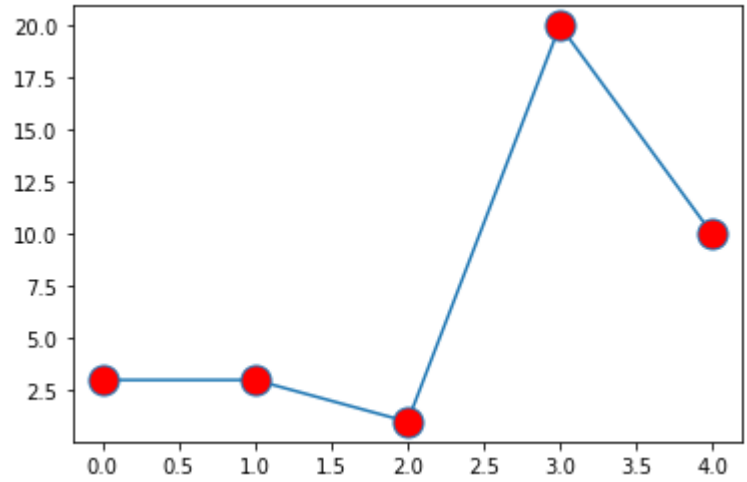
```
y = np.array([3, 3, 1, 20, 10])

plt.plot(y , marker = 'o', ms=15, mec = 'r')
plt.show()
```



markerfacecolor

In [40]:

```
plt.plot(y , marker = 'o', ms=15, mfc = 'r')
plt.show()
```

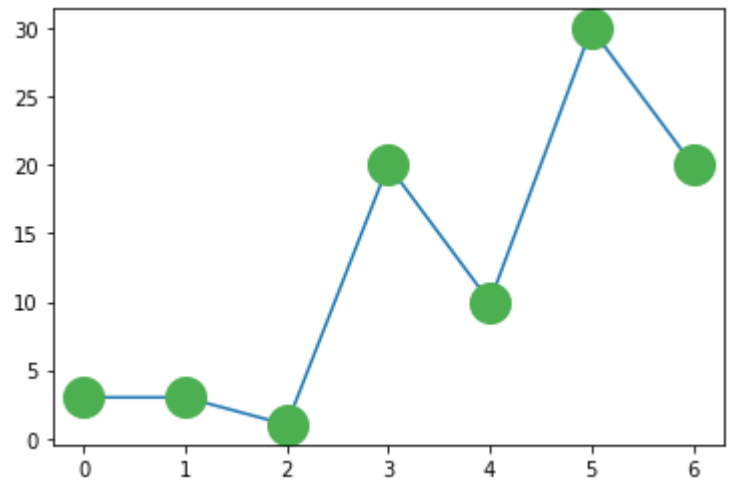


In [41]:

```
plt.plot(ypoints, marker = 'o', ms = 20, mec = '#4CAF50', mfc = '#4CAF50')
```

Out[41]:

```
[<matplotlib.lines.Line2D at 0x7fdc71d43910>]
```

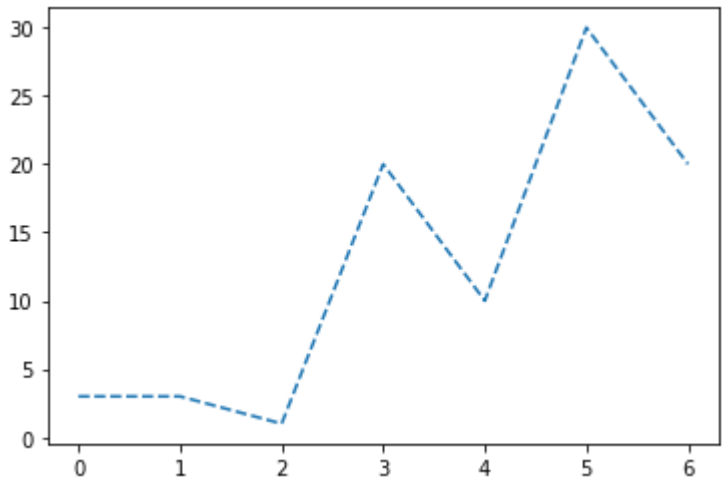


In [42]:

```
plt.plot(ypoints, linestyle = 'dashed')
```

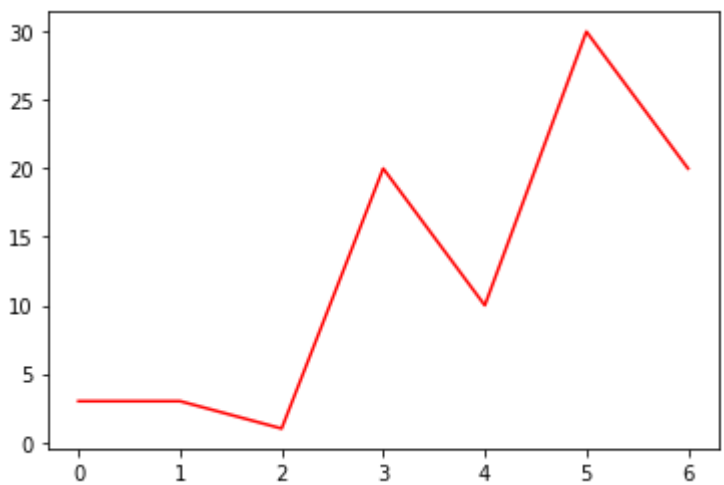
Out[42]:

```
[<matplotlib.lines.Line2D at 0x7fdc727b2970>]
```



In [43]:

```
plt.plot(ypoints, color = 'r')
plt.show()
```

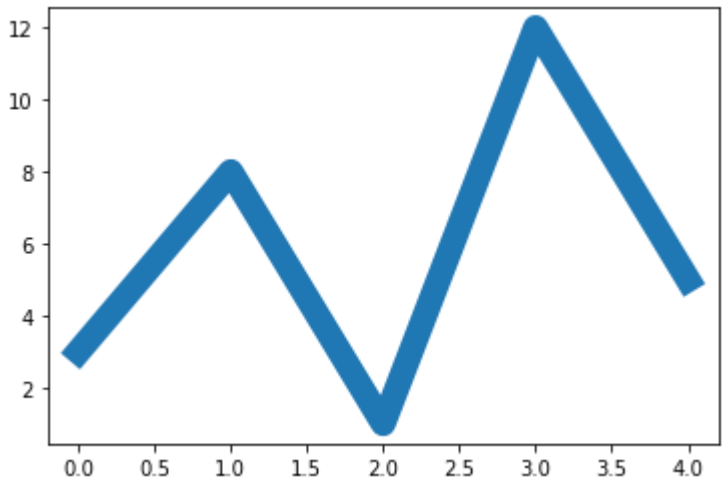


In [44]:

```
### 선 굵기
y = np.array([3, 8, 1, 12, 5])
plt.plot(y, linewidth = '12.5')
```

Out[44]:

```
[<matplotlib.lines.Line2D at 0x7fdc72607f70>]
```

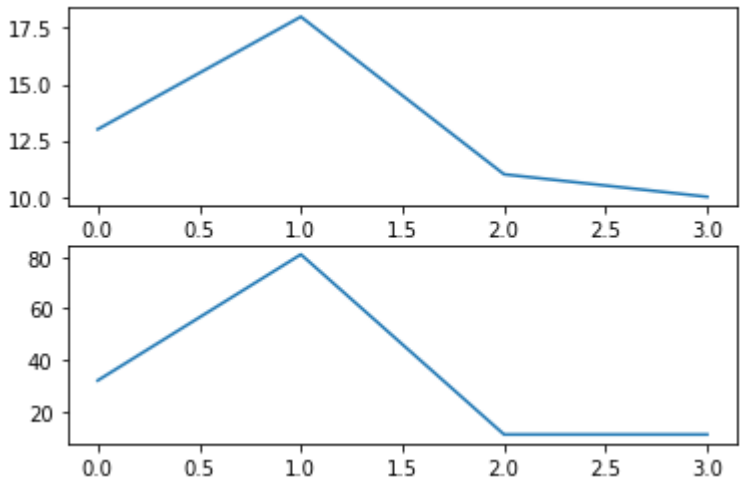


03. 여러 개의 그래프 그리기

- 한 화면에 여러개의 그래프를 그리기 위해서는
  - (1) figure 함수를 통해 Figure 객체를 만든다.
  - (2) add_subplot 메서드를 통해 그리려는 그래프 개수만큼 subplot를 만들면 된다.

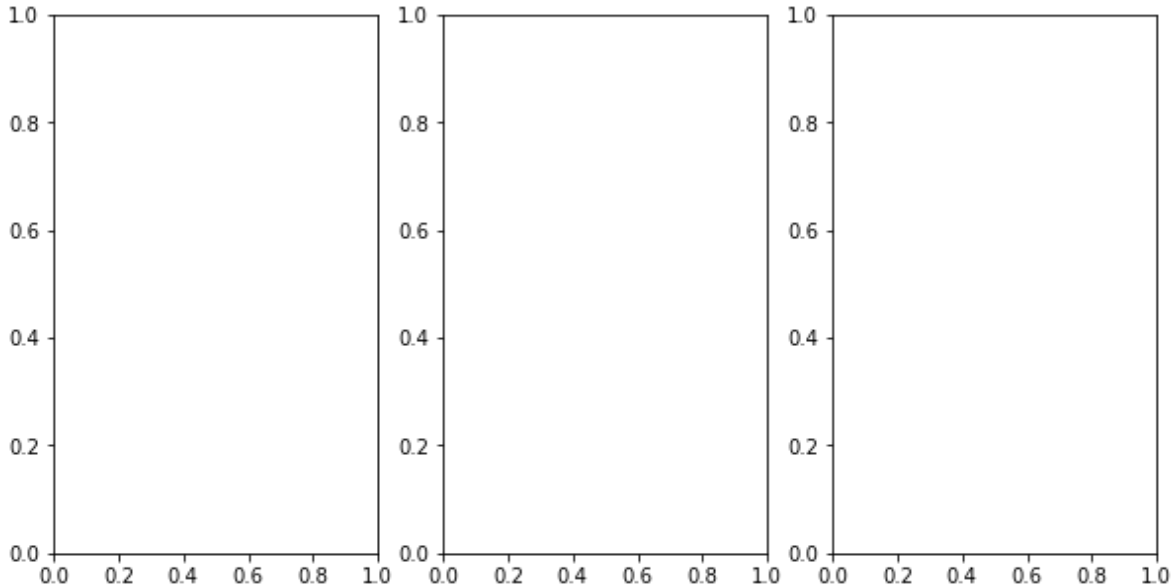
In [45]:

```
fig = plt.figure()
ax1 = fig.add_subplot(2,1,1)
ax2 = fig.add_subplot(2,1,2)

x1 = np.array([0, 1, 2, 3])
y1 = np.array([13, 18, 11, 10])

x2 = np.array([0, 1, 2, 3])
y2 = np.array([32, 81, 11, 11])

ax1.plot(x1, y1)
ax2.plot(x2, y2)
plt.show()
```



In [47]:

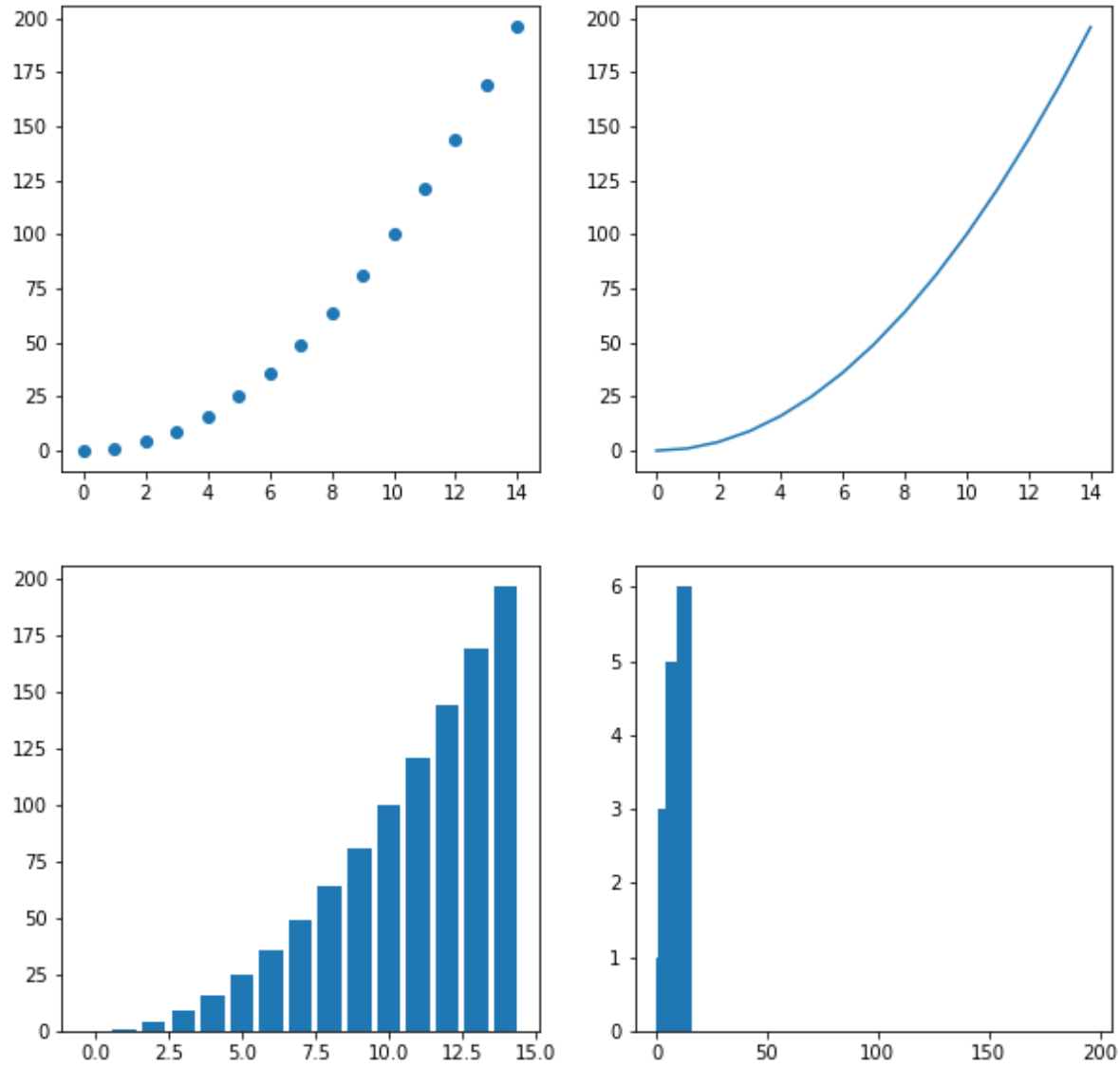
```
fig = plt.figure(figsize=(10,5))
ax1 = fig.add_subplot(1,3,1)  # 1행 3열 의 것중에 첫번째
ax2 = fig.add_subplot(1,3,2)  # 1행 3열 의 것중에 두번째
ax3 = fig.add_subplot(1,3,3)  # 1행 3열 의 것중에 두번째
plt.show()
```



plt.subplots(행, 열) 이용

In [60]:

```
### plot, bar 그래프 그리기
x = range(0,15)
y = [v*v for v in x]

fig, axs = plt.subplots(2,2, figsize=(10,10))
axs[0,0].plot(x, y, 'o')
axs[0,1].plot(x, y, '-')
axs[1,0].bar(x, y)
axs[1,1].hist(x, y)

plt.show()
```

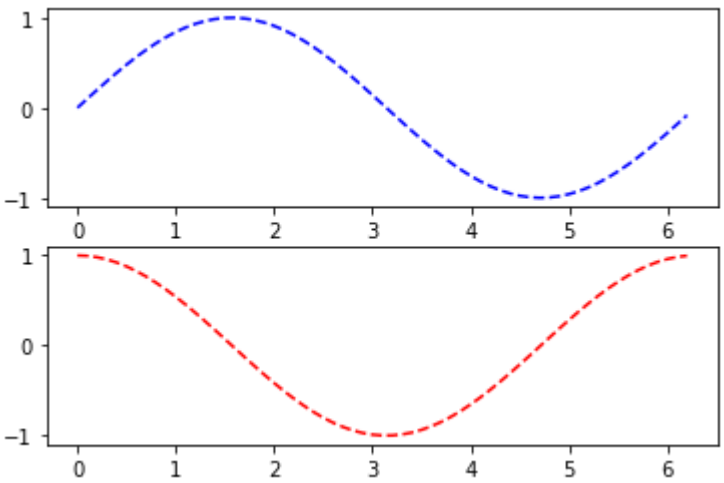


sin, cos 그래프 그려보기

In [55]:

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

x = np.arange(0.0, 2 * np.pi, 0.1)
sin_y = np.sin(x)
cos_y = np.cos(x)

plt.figure(figsize=(10,12))
fig, axs = plt.subplots(2)
axs[0].plot(x, sin_y, 'b--')
axs[1].plot(x, cos_y, 'r--')
plt.show()
```

```
<Figure size 720x864 with 0 Axes>
```



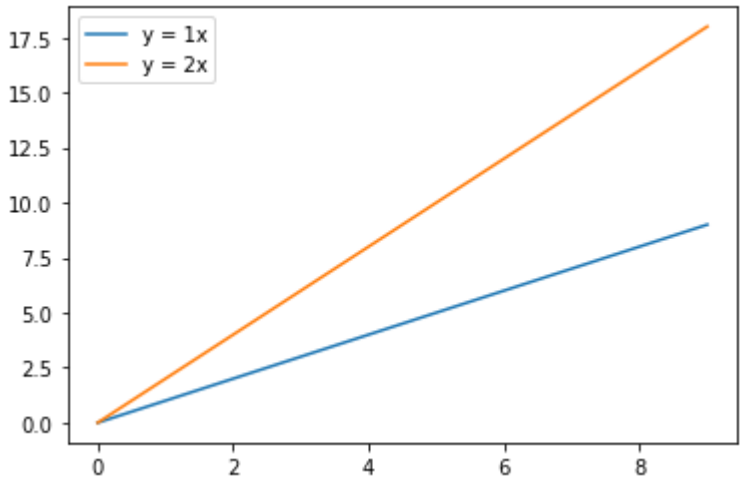
(실습2)

2행 2열의 그래프를 그려보자.
1행 1열: sin() 그래프
1행 2열: cos() 그래프 (표시 형식:빨간색 사각형)
2행 1열: tan() 그래프 (표시 형식:청록색 점)
2행 2열: arctan() 그래프 (표시 형식:노란색 원)

04. 범례 표시 - 기본

In [20]:

```
x = np.arange(10)
plt.plot(x, 1*x, label='y = %ix' % 1)
plt.plot(x, 2*x, label='y = %ix' % 2)
plt.legend()
plt.show()
```



5개의 그래프 그려보기

In [21]:

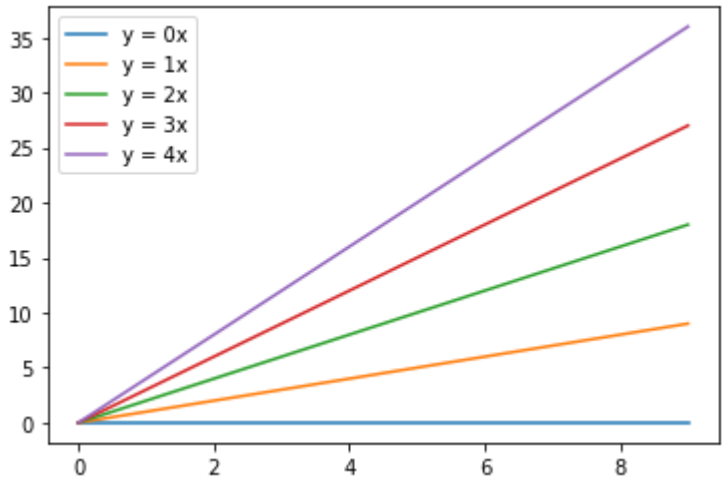
```
x = np.arange(10)
list(x)
```

Out[21]:

```
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

In [22]:

```
for i in range(5):
    plt.plot(x, i*x, label='y = %ix' % i)
plt.legend()
plt.show()
```



범례의 위치 변경과 subplots() 함수

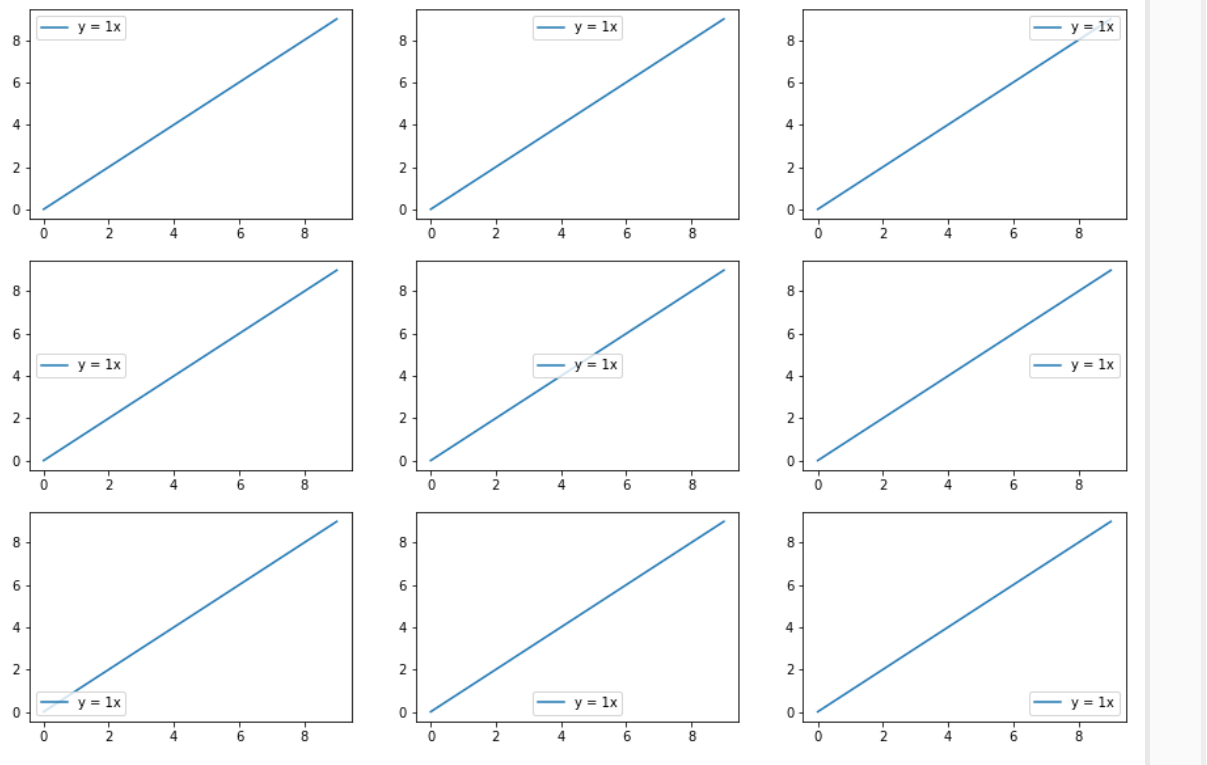
- subplots() 함수는 3개의 인수를 사용한다.

In [23]:

```
plt.figure(figsize=(15,10))

plt.subplot(3,3,1)
plt.plot(x, 1*x, label='y = %ix' % 1)
plt.legend(loc='upper left')    # 2

plt.subplot(3,3,2)
plt.plot(x, 1*x, label='y = %ix' % 1)
plt.legend(loc='upper center')  # 9

plt.subplot(3,3,3)
plt.plot(x, 1*x, label='y = %ix' % 1)
plt.legend(loc='upper right')  # 1

plt.subplot(3,3,4)
plt.plot(x, 1*x, label='y = %ix' % 1)
plt.legend(loc='center left')  # 6

plt.subplot(3,3,5)
plt.plot(x, 1*x, label='y = %ix' % 1)
plt.legend(loc='center')  # 10

plt.subplot(3,3,6)
plt.plot(x, 1*x, label='y = %ix' % 1)
plt.legend(loc='center right')  # 5

plt.subplot(3,3,7)
plt.plot(x, 1*x, label='y = %ix' % 1)
plt.legend(loc='lower left')  # 3

plt.subplot(3,3,8)
plt.plot(x, 1*x, label='y = %ix' % 1)
plt.legend(loc='lower center')  # 8

plt.subplot(3,3,9)
plt.plot(x, 1*x, label='y = %ix' % 1)
plt.legend(loc='lower right')  # 4
```

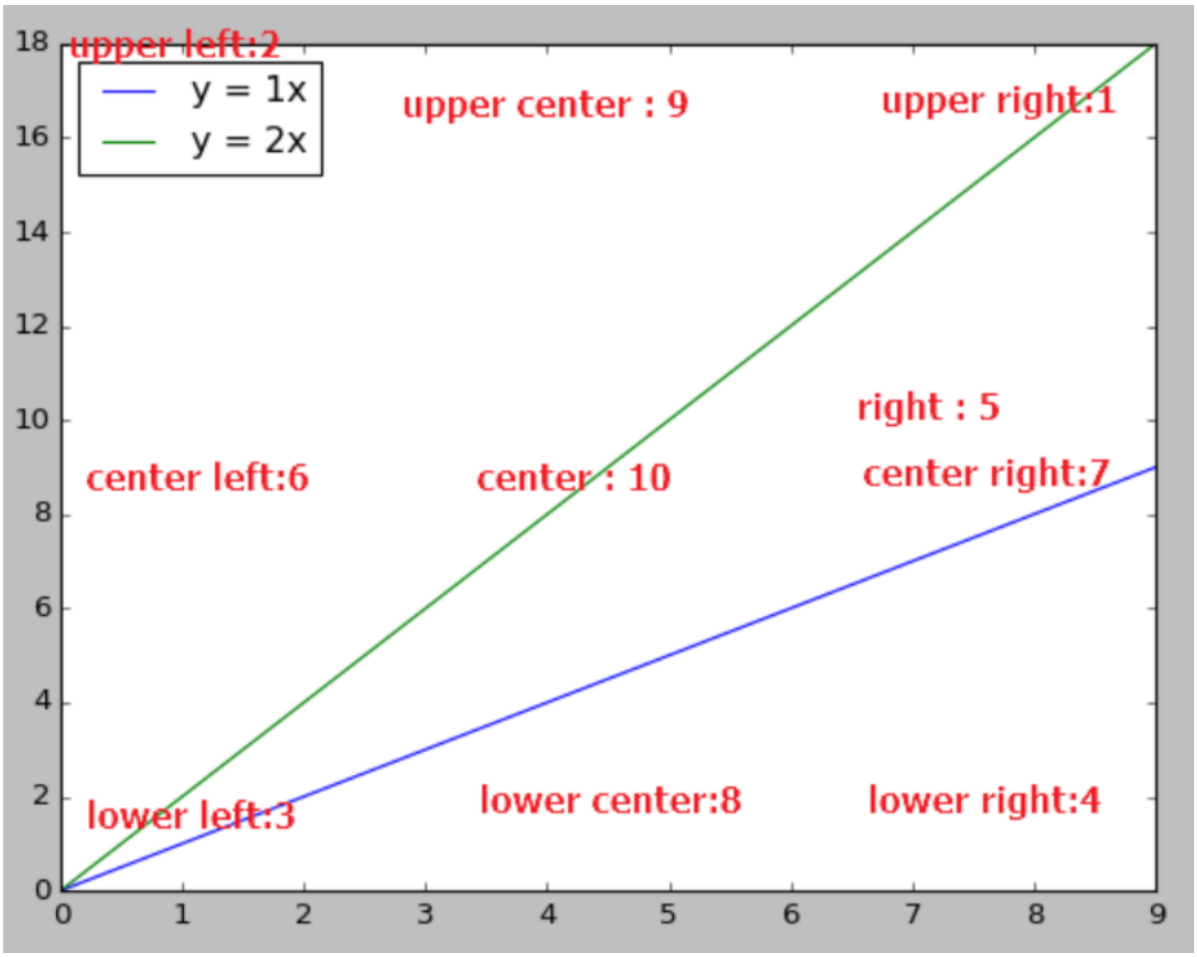
Out[23]:

```
<matplotlib.legend.Legend at 0x24f8ce67160>
```



In [24]:

```
import IPython.display as display
from PIL import Image

display.display(Image.open('./plt_legend_0622.png'))
```



Reference :

- https://www.w3schools.com/colors/colors_names.asp (https://www.w3schools.com/colors/colors_names.asp)