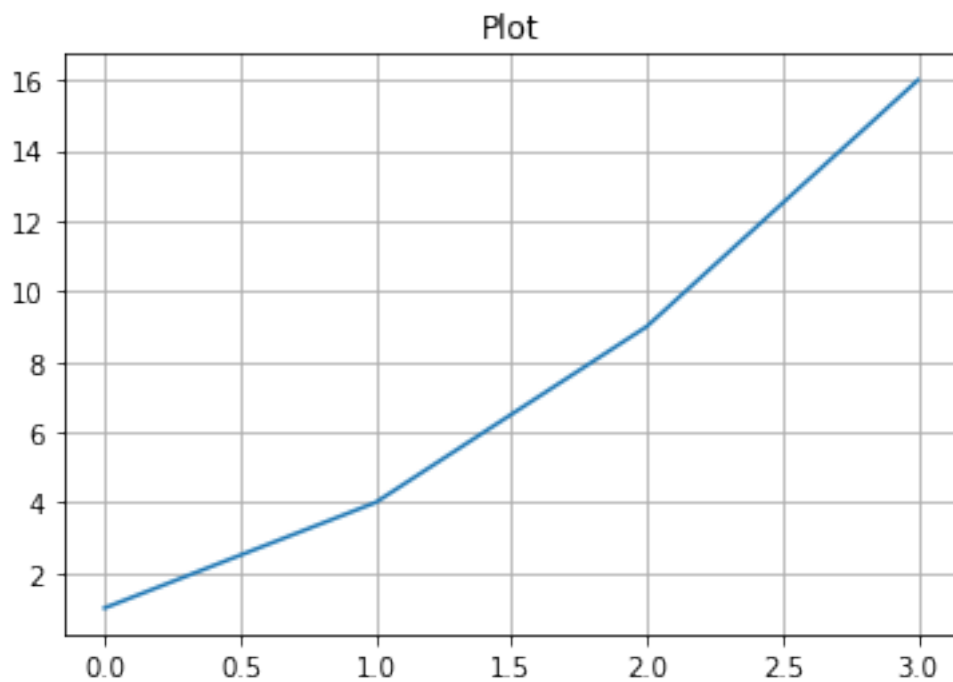


Assignment_#2

문성찬 (20171620)

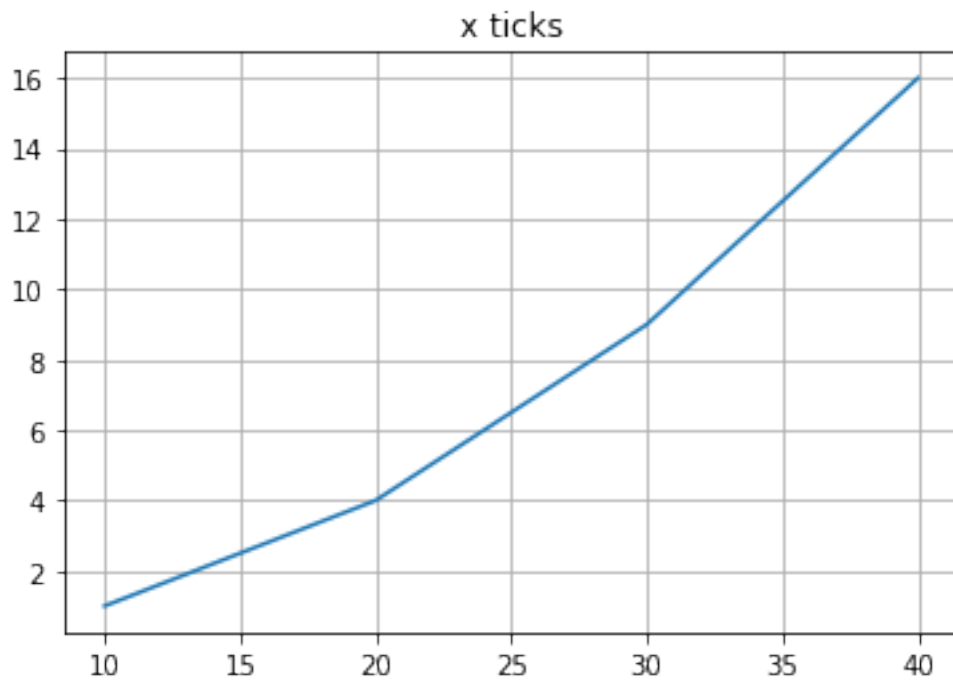
```
In [1]: import matplotlib as mpl
import matplotlib.pyplot as plt
import numpy as np
%matplotlib inline
plt.rcParams['axes.grid'] = True
```

```
In [2]: plt.title("Plot")
plt.plot([1,4,9,16])
plt.show()
```



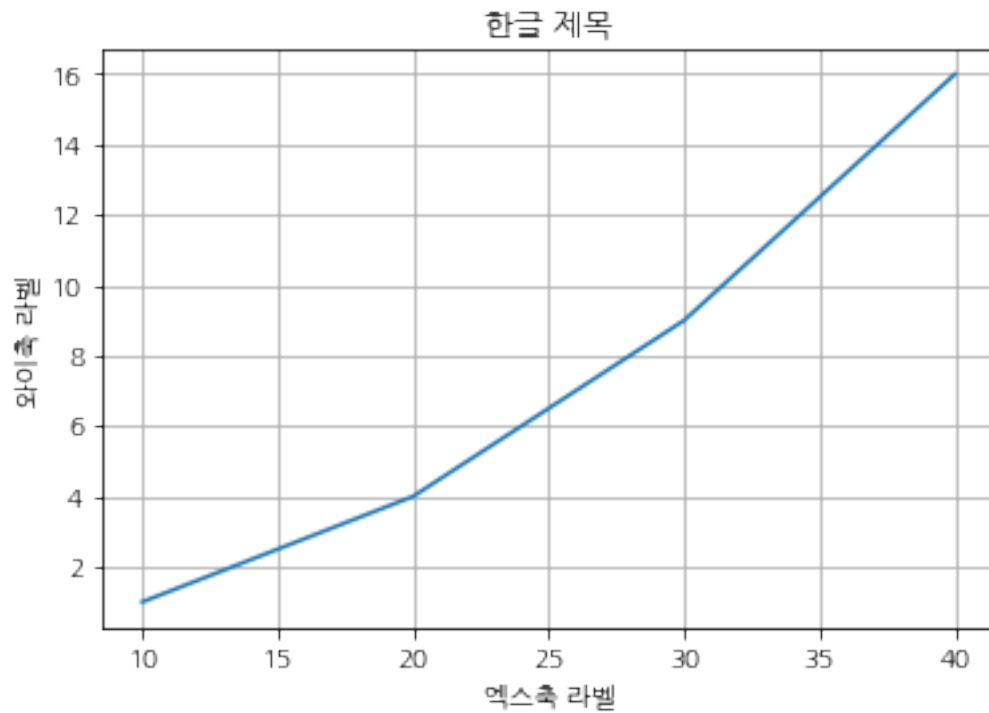
※jupyter-notebook에서 pdf파일 변환 과정에서 코드내에 한글 부분은 받아지지 않습니다.

```
In [3]: plt.title("x ticks")
plt.plot([10, 20, 30, 40], [1, 4, 9, 16])
plt.show()
```



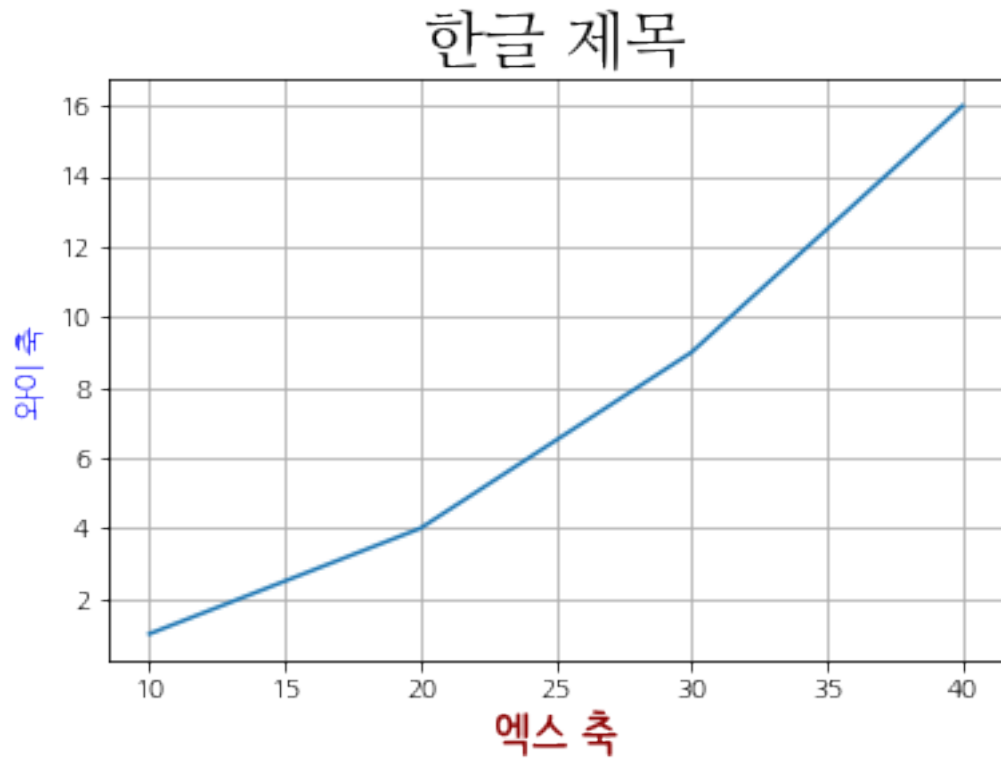
```
In [4]: #
mpl.rc('font', family='NanumGothic')
#
mpl.rc('axes', unicode_minus=False)
```

```
In [5]: plt.title(' ')
plt.plot([10, 20, 30, 40], [1, 4, 9, 16])
plt.xlabel(" ")
plt.ylabel(" ")
plt.show()
```

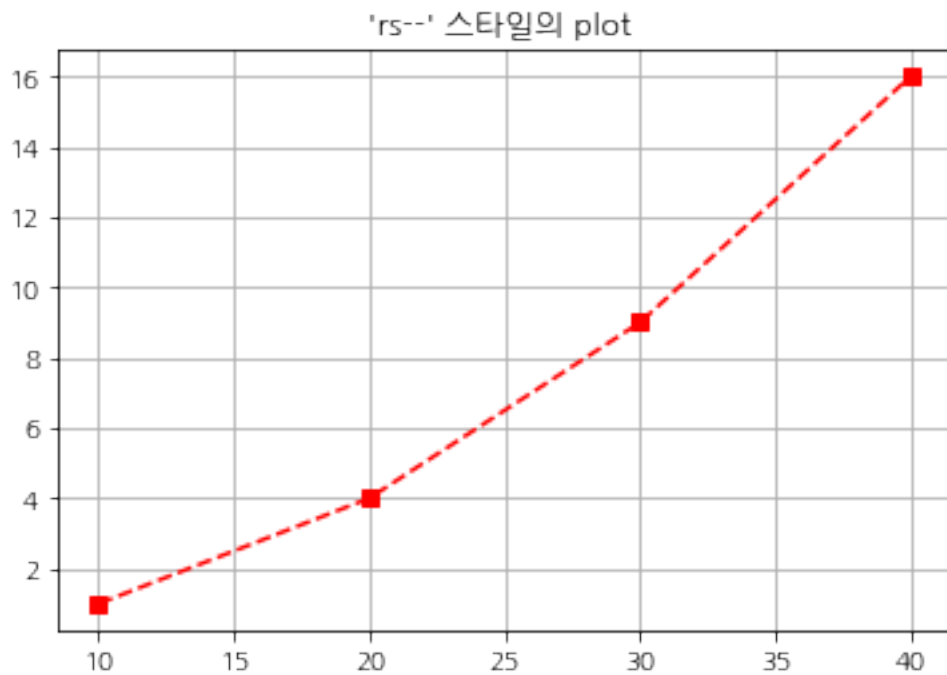


```
In [6]: font1 = {'family': 'NanumMyeongjo', 'size': 24,
                'color': 'black'}
font2 = {'family': 'NanumBarunpen', 'size': 18, 'weight': 'bold',
        'color': 'darkred'}
font3 = {'family': 'NanumBarunGothic', 'size': 12, 'weight': 'light',
        'color': 'blue'}

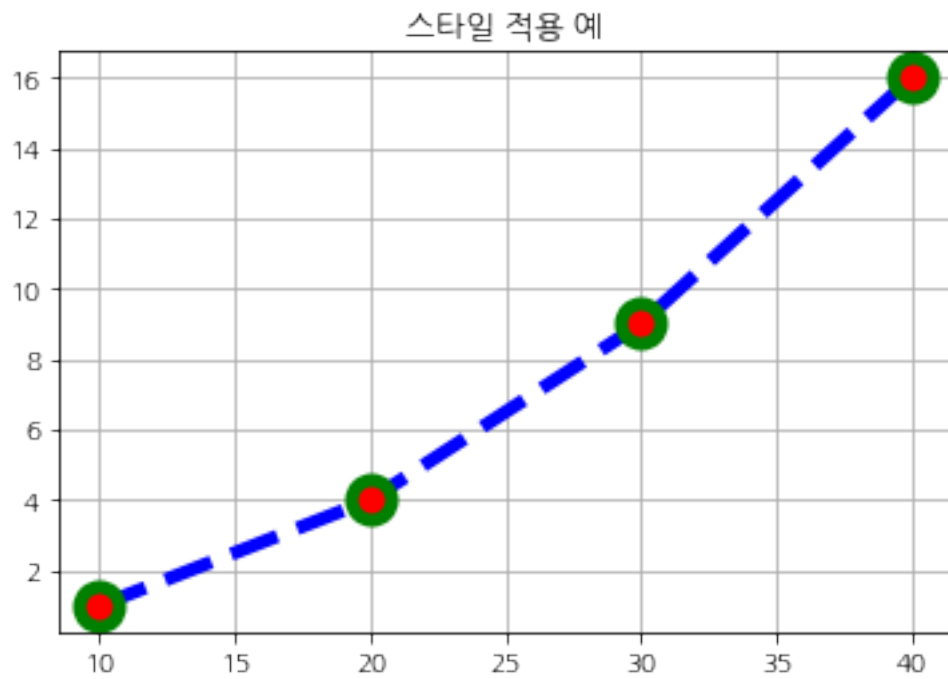
plt.plot([10, 20, 30, 40], [1, 4, 9, 16])
plt.title(' ', fontdict=font1)
plt.xlabel(' ', fontdict=font2)
plt.ylabel(' ', fontdict=font3)
plt.show()
```



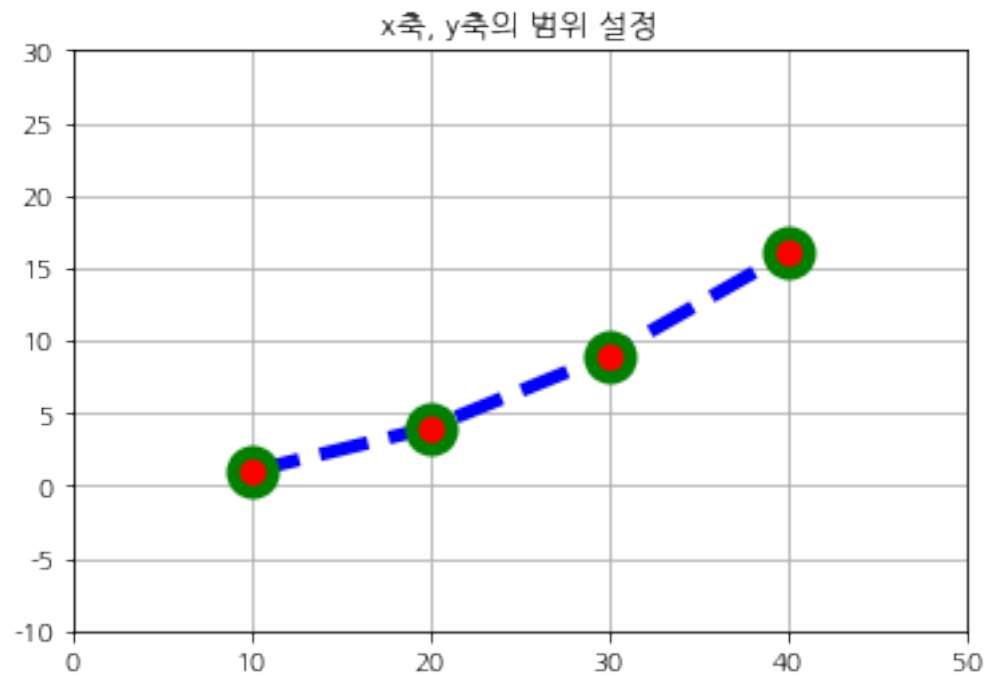
```
In [7]: plt.title("'rs--' plot ")
plt.plot([10, 20, 30, 40], [1, 4, 9, 16], 'rs--')
plt.show()
```



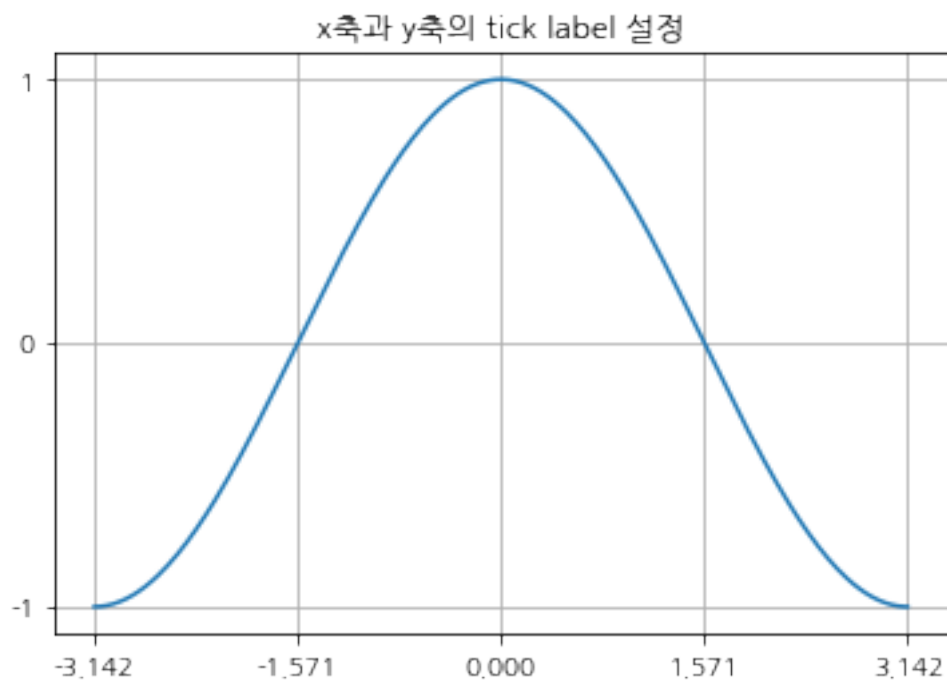
```
In [8]: plt.plot([10, 20, 30, 40], [1, 4, 9, 16], c="b",  
                lw=5, ls="--", marker="o", ms=15, mec="g", mew=5, mfc="r")  
plt.title(" ")  
plt.show()
```



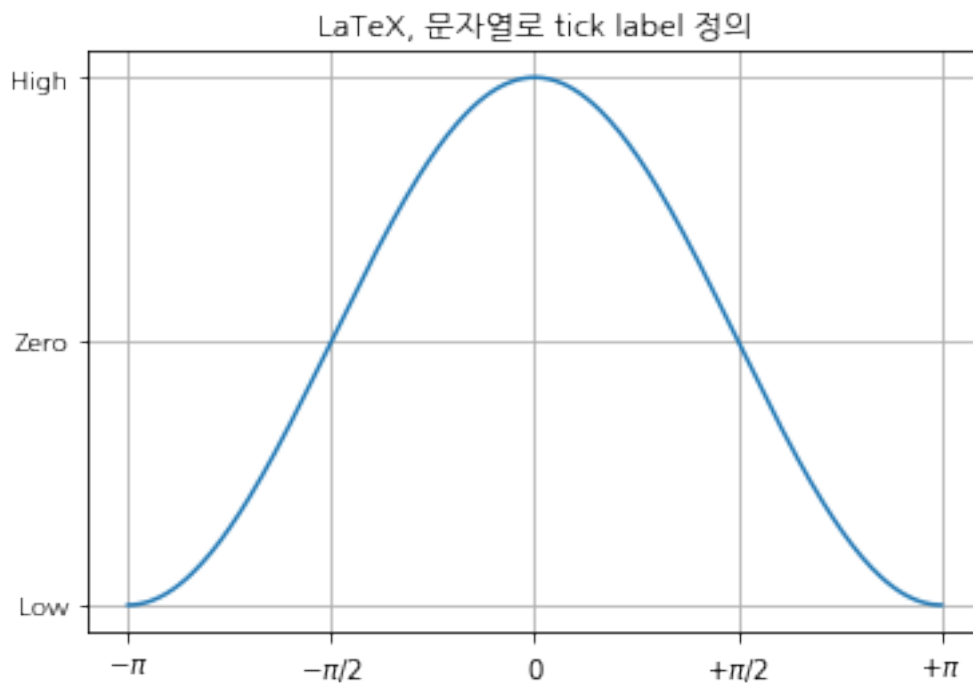
```
In [9]: plt.title("x, y ")
plt.plot([10, 20, 30, 40], [1, 4, 9, 16],
         c="b", lw=5, ls="--", marker="o", ms=15, mec="g", mew=5, mfc="r")
plt.xlim(0, 50)
plt.ylim(-10, 30)
plt.show()
```



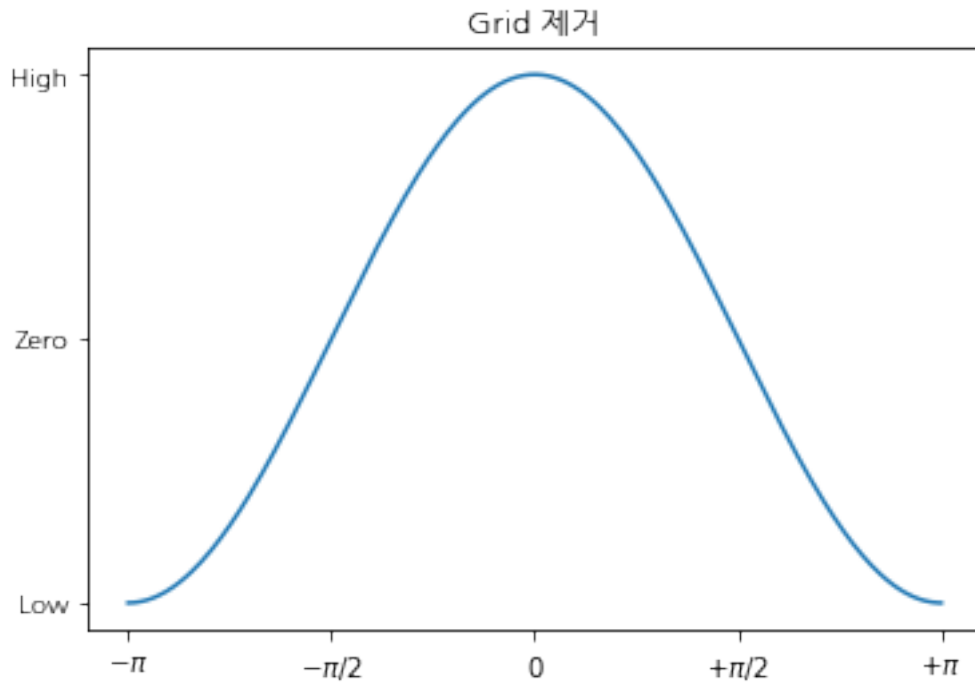
```
In [10]: X = np.linspace(-np.pi, np.pi, 256)
C = np.cos(X)
plt.title("x y tick label ")
plt.plot(X, C)
plt.xticks([-np.pi, -np.pi / 2, 0, np.pi / 2, np.pi])
plt.yticks([-1, 0, +1])
plt.show()
```



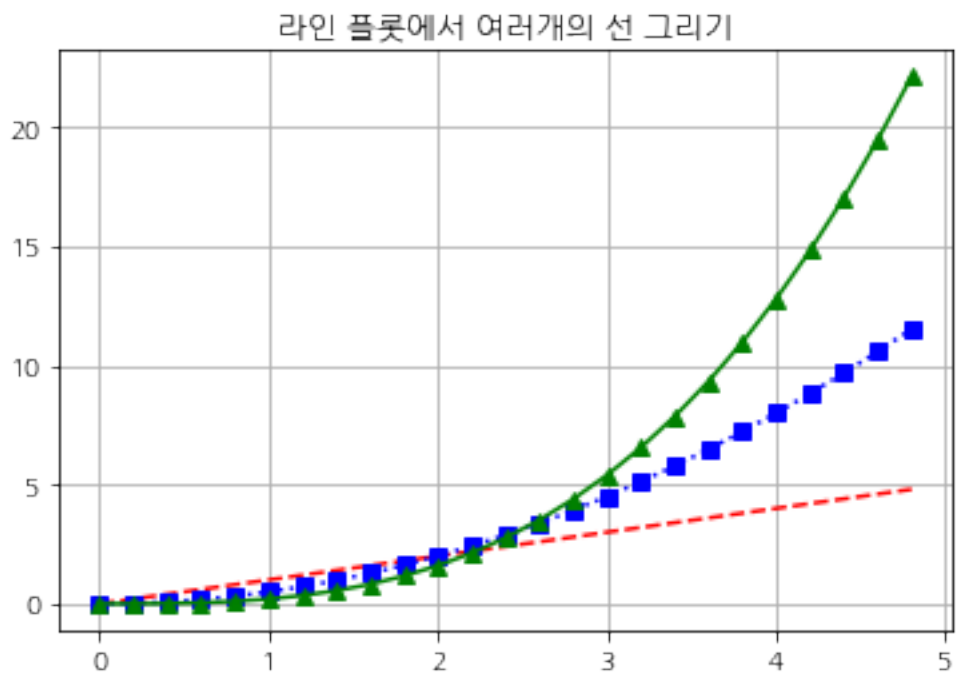
```
In [11]: X = np.linspace(-np.pi, np.pi, 256)
C = np.cos(X)
plt.title("LaTeX, tick label ")
plt.plot(X, C)
plt.xticks([-np.pi, -np.pi / 2, 0, np.pi / 2, np.pi],
           [r'$-\pi$', r'$-\pi/2$', r'$0$', r'$+\pi/2$', r'$+\pi$'])
plt.yticks([-1, 0, 1], ["Low", "Zero", "High"])
plt.show()
```



```
In [12]: X = np.linspace(-np.pi, np.pi, 256)
C = np.cos(X)
plt.title("Grid ")
plt.plot(X, C)
plt.xticks([-np.pi, -np.pi / 2, 0, np.pi / 2, np.pi],
           [r'$-\pi$', r'$-\pi/2$', r'$0$', r'$+\pi/2$', r'$+\pi$'])
plt.yticks([-1, 0, 1], ["Low", "Zero", "High"])
plt.grid(False)
plt.show()
```

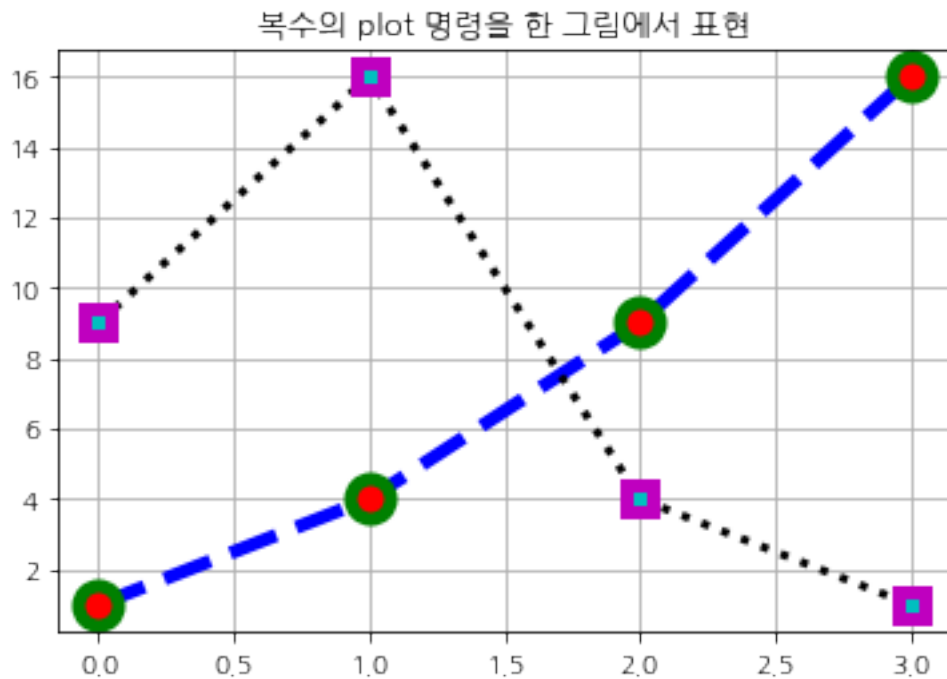
```
In [13]: t = np.arange(0., 5., 0.2)
plt.title(" ")
plt.plot(t, t, 'r--', t, 0.5 * t**2, 'bs:', t, 0.2 * t**3, 'g^-')
plt.show()
```



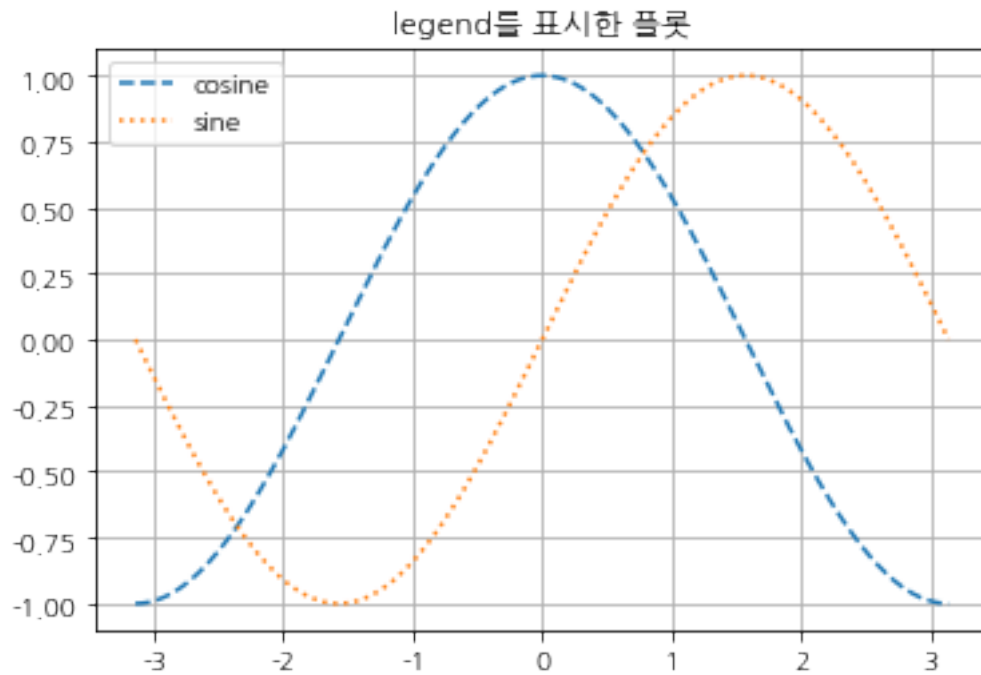
```

In [14]: plt.title(" plot ")
plt.plot([1, 4, 9, 16],
         c="b", lw=5, ls="--", marker="o", ms=15, mec="g", mew=5, mfc="r")
# plt.hold(True)
plt.plot([9, 16, 4, 1],
         c="k", lw=3, ls=":", marker="s", ms=10, mec="m", mew=5, mfc="c")
# plt.hold(False)
plt.show()

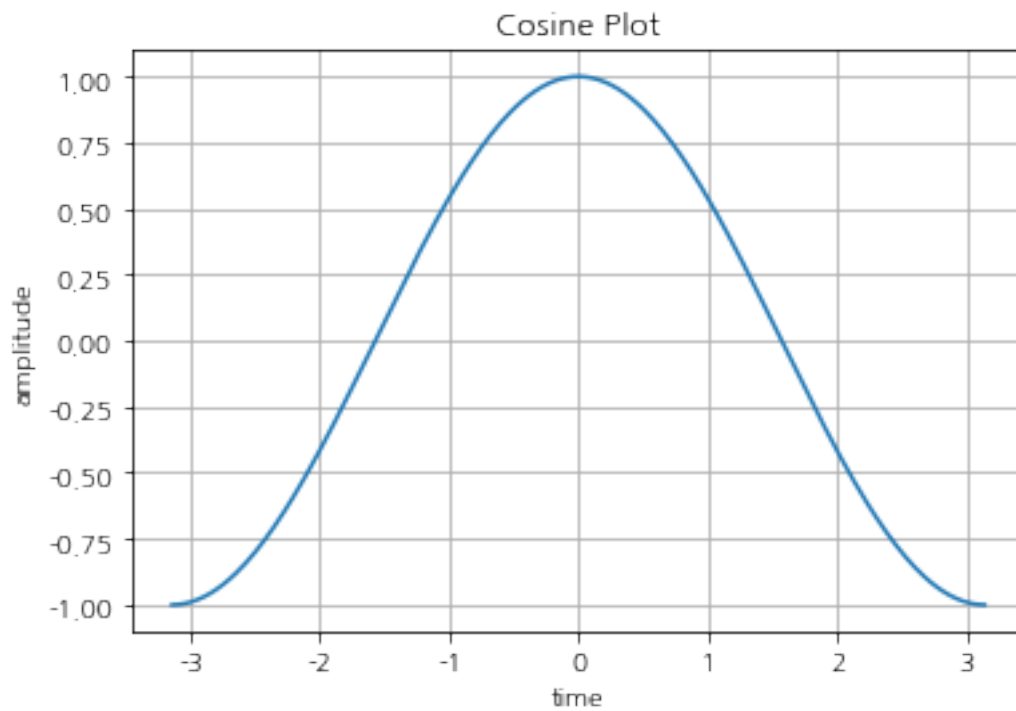
```



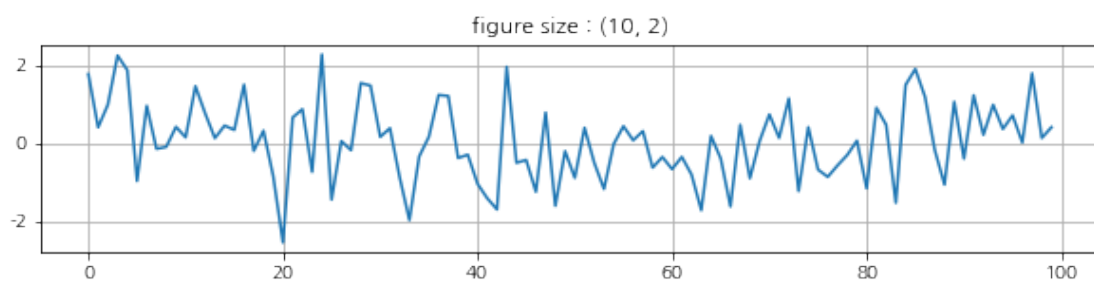
```
In [15]: X = np.linspace(-np.pi, np.pi, 256)
C, S = np.cos(X), np.sin(X)
plt.title("legend ")
plt.plot(X, C, ls="--", label="cosine")
plt.plot(X, S, ls=":", label="sine")
plt.legend(loc=2)
plt.show()
```



```
In [16]: X = np.linspace(-np.pi, np.pi, 256)
C, S = np.cos(X), np.sin(X)
plt.plot(X, C, label="cosine")
plt.xlabel("time")
plt.ylabel("amplitude")
plt.title("Cosine Plot")
plt.show()
```



```
In [17]: np.random.seed(0)
         f1 = plt.figure(figsize=(10, 2))
         plt.title("figure size : (10, 2)")
         plt.plot(np.random.randn(100))
         plt.show()
```



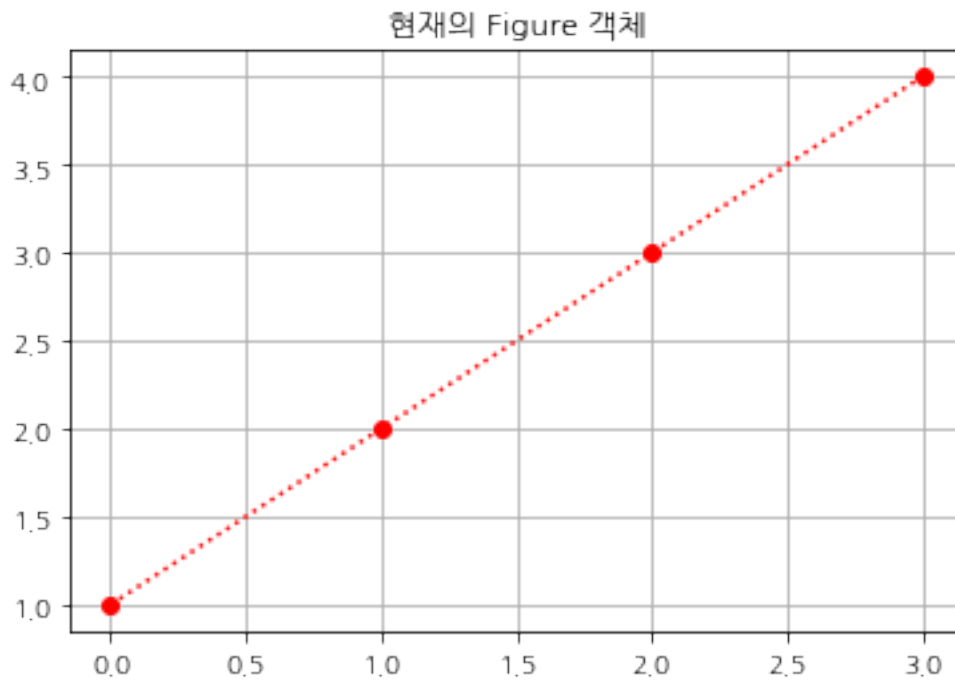
```
In [18]: f1 = plt.figure(1)
plt.title(" Figure ")
plt.plot([1, 2, 3, 4], 'ro:')

f2 = plt.gcf()
print(f1, id(f1))

print(f2, id(f2))
plt.show()
```

Figure(432x288) 140361667026448

Figure(432x288) 140361667026448



```

In [19]: x1 = np.linspace(0.0, 5.0)
         x2 = np.linspace(0.0, 2.0)
         y1 = np.cos(2 * np.pi * x1) * np.exp(-x1)
         y2 = np.cos(2 * np.pi * x2)

         ax1 = plt.subplot(2, 1, 1)
         plt.plot(x1, y1, 'yo-')
         plt.title('A tale of 2 subplots')
         plt.ylabel('Damped oscillation')
         print(ax1)

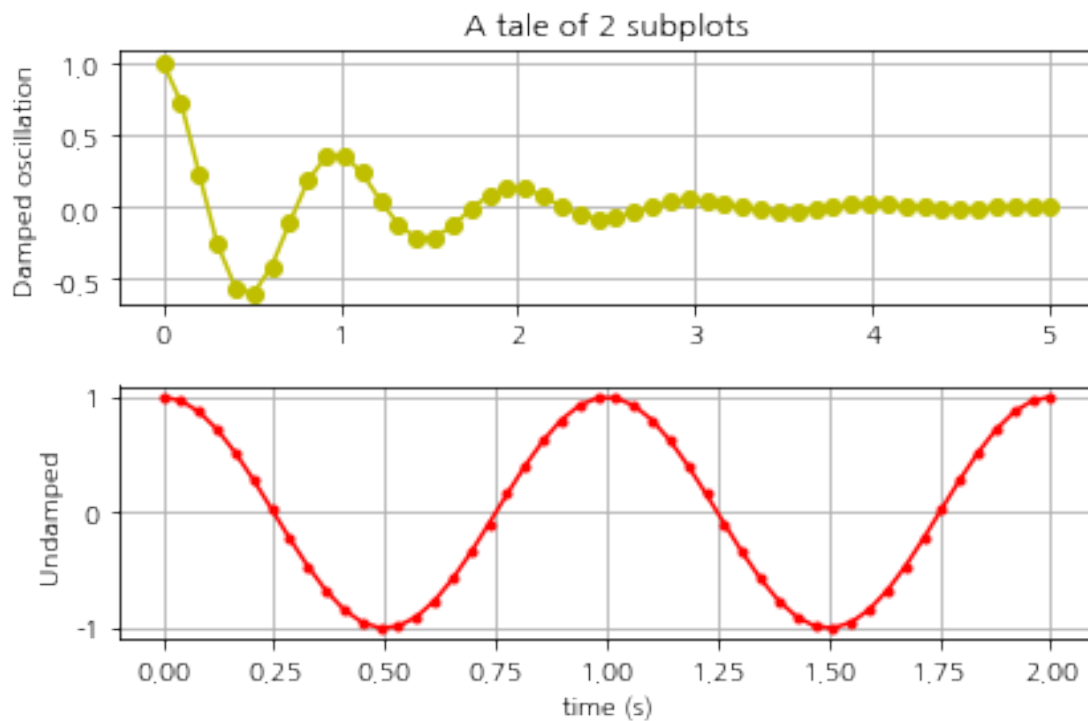
         ax2 = plt.subplot(2, 1, 2)
         plt.plot(x2, y2, 'r.-')
         plt.xlabel('time (s)')
         plt.ylabel('Undamped')
         print(ax2)

         plt.tight_layout()
         plt.show()

```

Axes(0.125,0.536818;0.775x0.343182)

Axes(0.125,0.125;0.775x0.343182)



```
In [20]: np.random.seed(0)
```

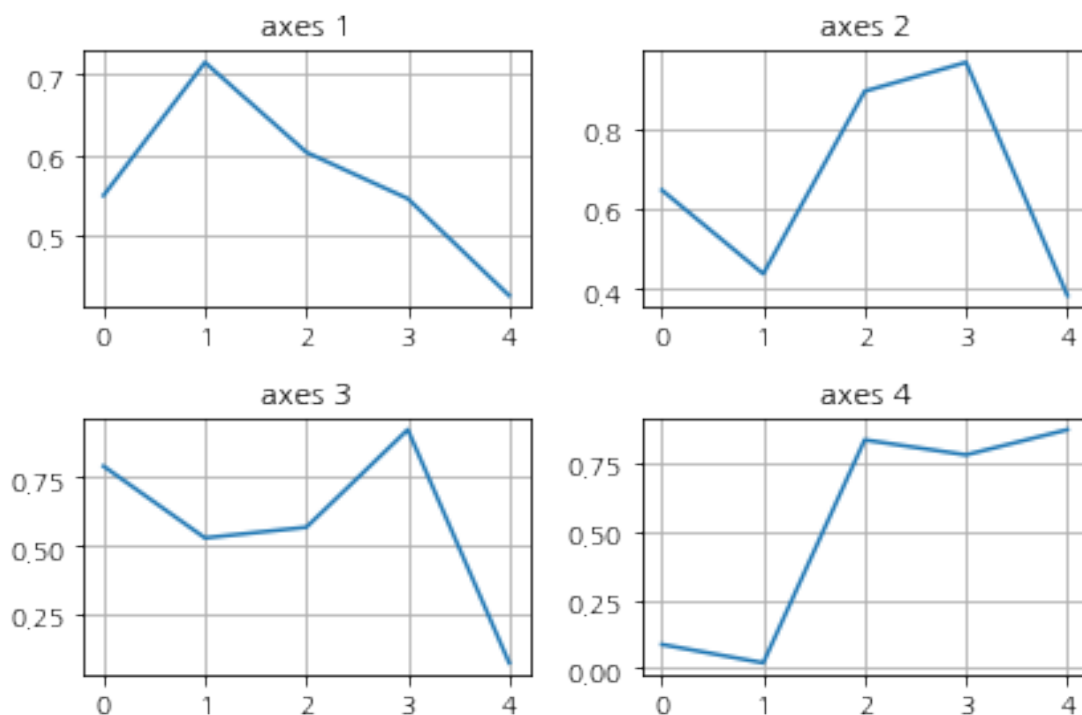
```
plt.subplot(221)  
plt.plot(np.random.rand(5))  
plt.title("axes 1")
```

```
plt.subplot(222)  
plt.plot(np.random.rand(5))  
plt.title("axes 2")
```

```
plt.subplot(223)  
plt.plot(np.random.rand(5))  
plt.title("axes 3")
```

```
plt.subplot(224)  
plt.plot(np.random.rand(5))  
plt.title("axes 4")
```

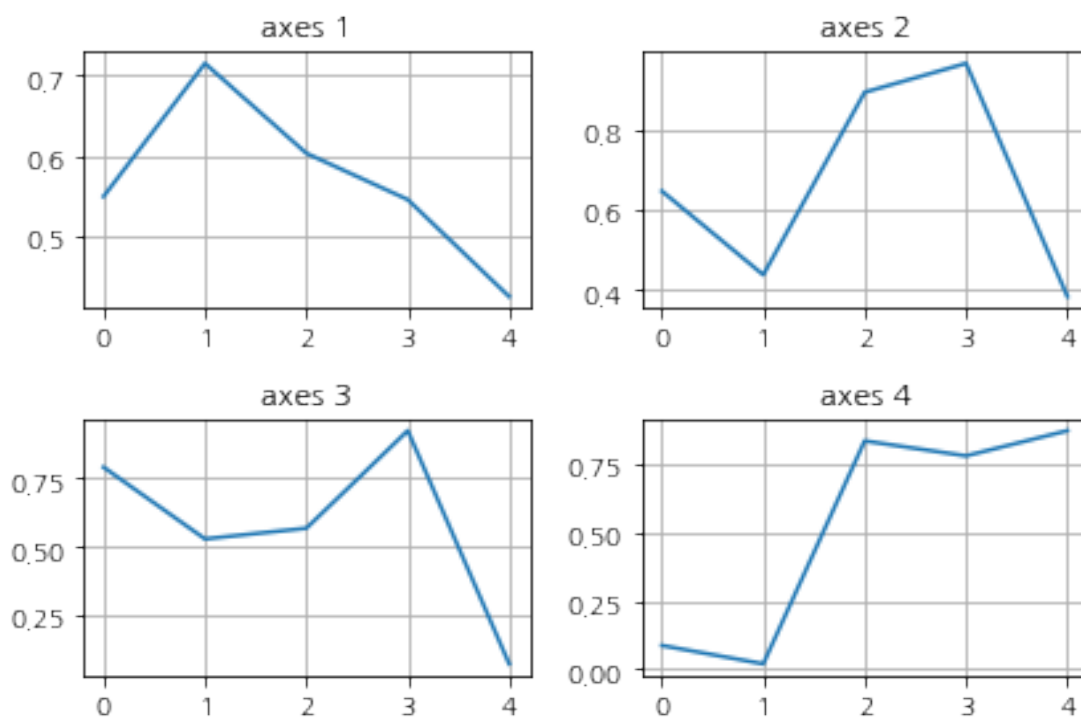
```
plt.tight_layout()  
plt.show()
```



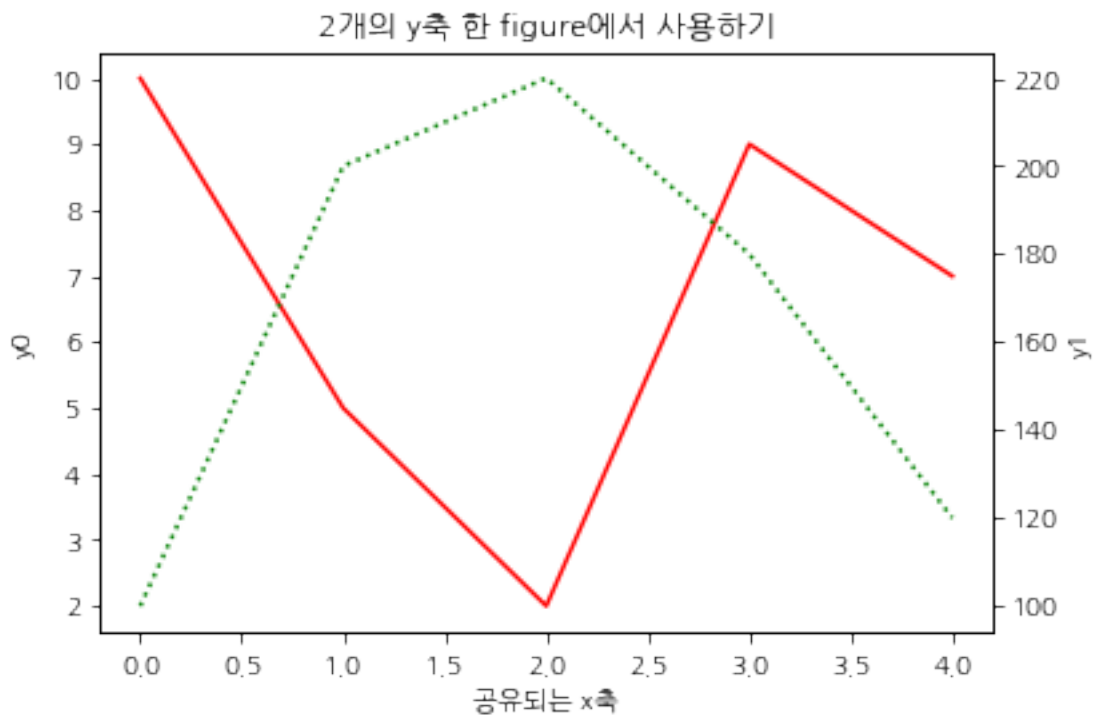
```
In [21]: fig, axes = plt.subplots(2, 2)
```

```
np.random.seed(0)
axes[0, 0].plot(np.random.rand(5))
axes[0, 0].set_title("axes 1")
axes[0, 1].plot(np.random.rand(5))
axes[0, 1].set_title("axes 2")
axes[1, 0].plot(np.random.rand(5))
axes[1, 0].set_title("axes 3")
axes[1, 1].plot(np.random.rand(5))
axes[1, 1].set_title("axes 4")
```

```
plt.tight_layout()
plt.show()
```




```
In [22]: fig, ax0 = plt.subplots()
         ax1 = ax0.twinx()
         ax0.set_title("2 y figure ")
         ax0.plot([10, 5, 2, 9, 7], 'r-', label="y0")
         ax0.set_ylabel("y0")
         ax0.grid(False)
         ax1.plot([100, 200, 220, 180, 120], 'g:', label="y1")
         ax1.set_ylabel("y1")
         ax1.grid(False)
         ax0.set_xlabel(" x")
         plt.show()
```



소감 /
 다양한 형태의 matplotlib을 실습해보면서 데이터들을 어떻게 그래프에
 표현할 수 있는지, 데이터 시각화에서 신경써야할 부분들이 무엇인지
 배울 수 있었습니다.