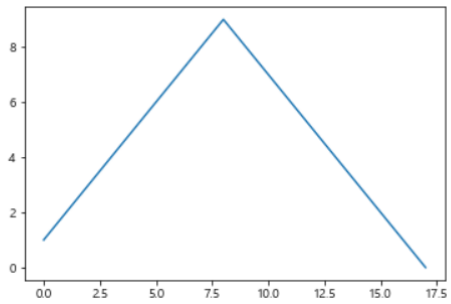
[그래프 그리기 기초 – matplotlib]

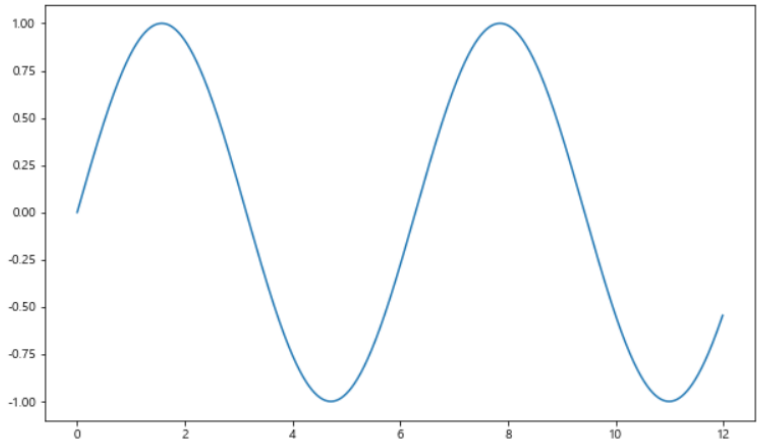
In [358]:
```
import matplotlib.pyplot as plt
%matplotlib inline
```

In [359]:
```
plt.figure()
plt.plot([1,2,3,4,5,6,7,8,9,8,7,6,5,4,3,2,1,0])  #그래프를 그릴 데이터 준비
plt.show()  #그래프 그림
```

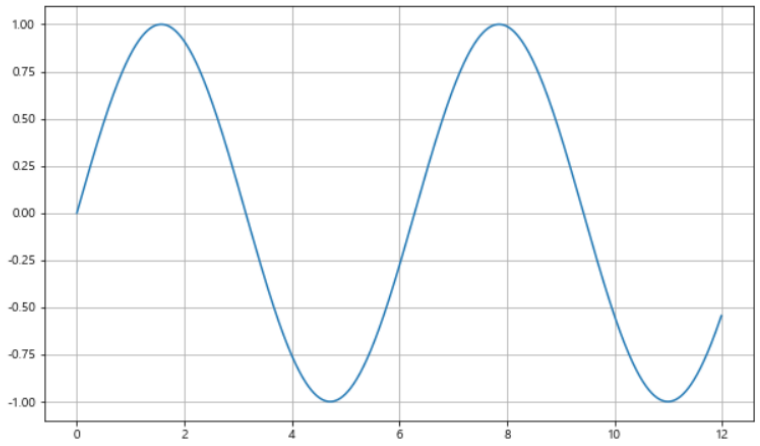


In [360]:
```
import numpy as np

t = np.arange(0,12,0.01) #0부터 12까지 0.01. 간격으로 데이터 생성. (1200개 정도의 일종의 배열)

y = np.sin(t)    # sin 값 반환
```

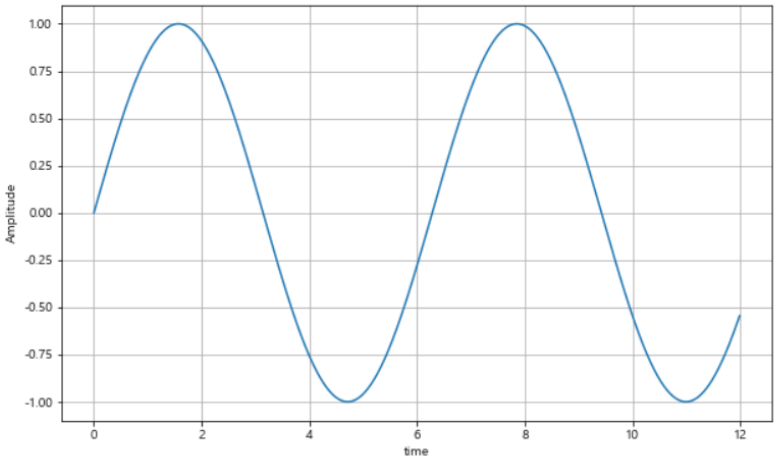
In [361]:
```
plt.figure(figsize=(10,6))   #그래프 크기
plt.plot(t, y) #그래프를 그릴 x축 데이터는 t.  y축데이터 y.
plt.show()
```



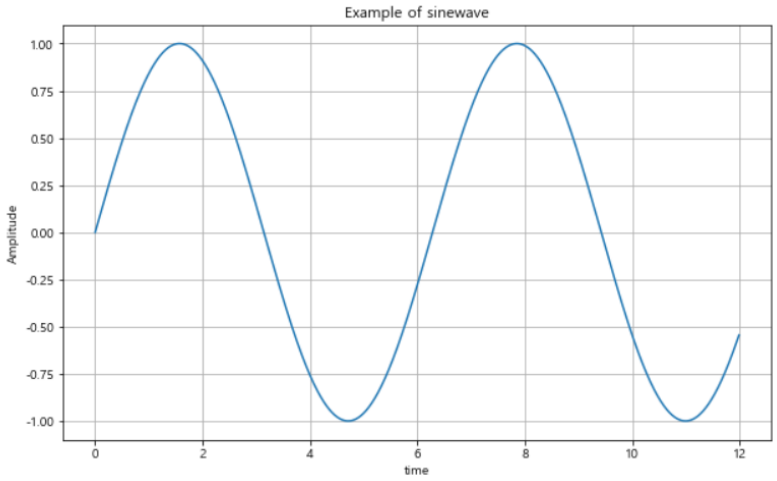
In [362]:
```
plt.figure(figsize=(10,6))
plt.plot(t, y)
plt.grid() # 그리드 적용하기
plt.show()
```



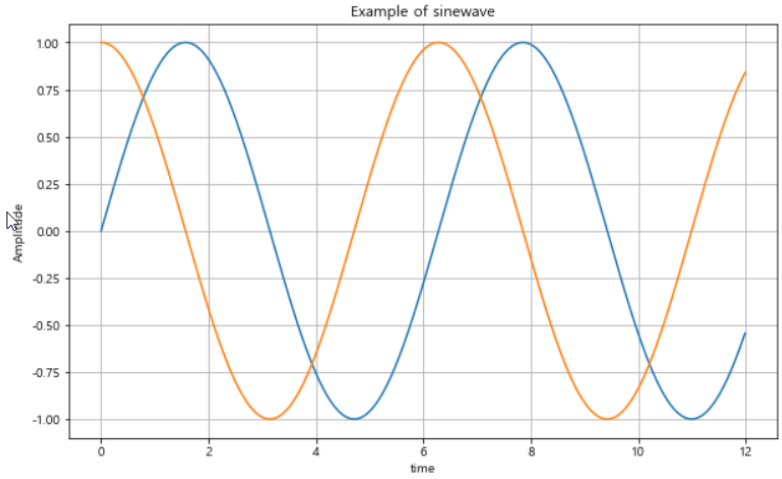
In [363]:
```
plt.figure(figsize=(10,6))
plt.plot(t, y)
plt.grid()
plt.xlabel('time')      # x축 라벨 적용하기
plt.ylabel('Amplitude') # y축 라벨 적용하기
plt.show()
```



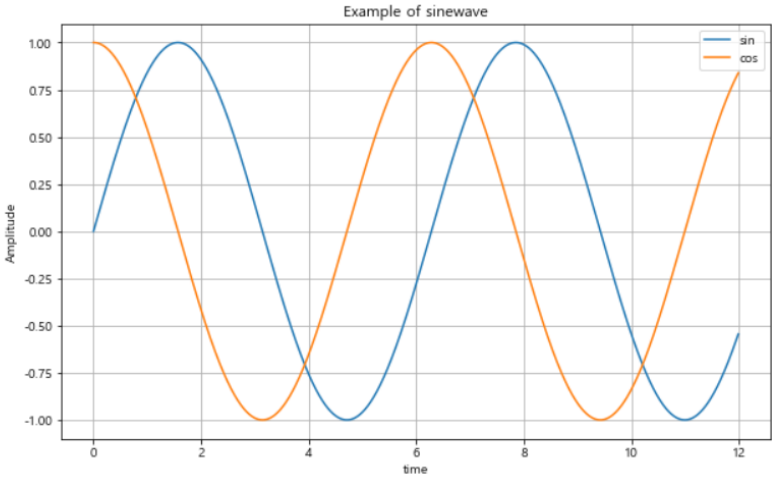
In [364]:
```
plt.figure(figsize=(10,6))
plt.plot(t, y)
plt.grid()
plt.xlabel('time')
plt.ylabel('Amplitude')
plt.title('Example of sinewave')  # 그래프 제목
plt.show()
```



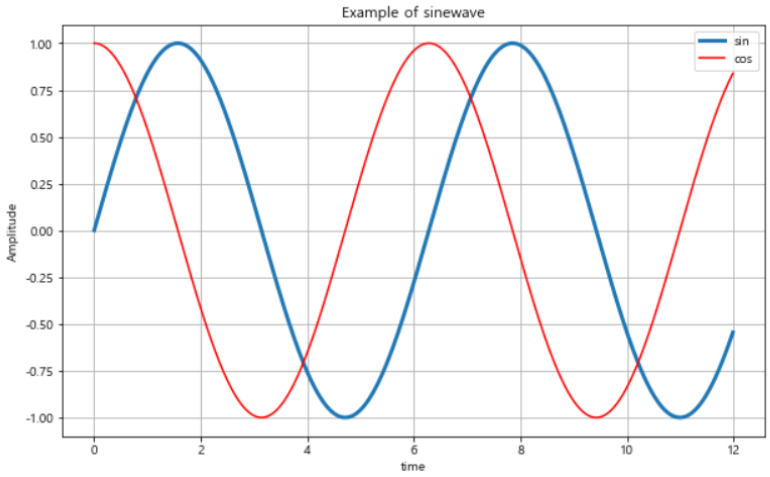
In [365]:
```
plt.figure(figsize=(10,6))
plt.plot(t, np.sin(t)) #그래프 첫번째 데이터  sin     #선이 두개 그려짐
plt.plot(t, np.cos(t)) #그래프 두번째 데이터  cos
plt.grid()
plt.xlabel('time')
plt.ylabel('Amplitude')
plt.title('Example of sinewave')
plt.show()
```



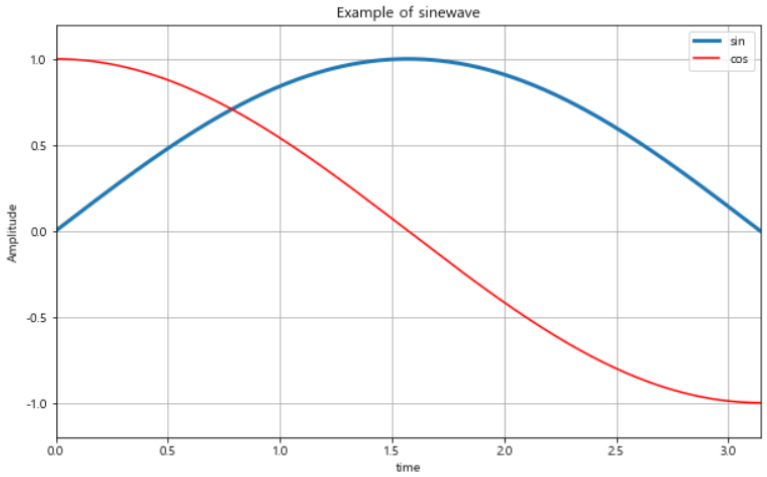
In [366]:
```
plt.figure(figsize=(10,6))
plt.plot(t, np.sin(t), label='sin')  #레전드 라벨   label='sin '
plt.plot(t, np.cos(t), label='cos')  #레전드 라벨   label='cos '
plt.grid()
plt.legend()  # 레전드
plt.xlabel('time')
plt.ylabel('Amplitude')
plt.title('Example of sinewave')
plt.show()
```



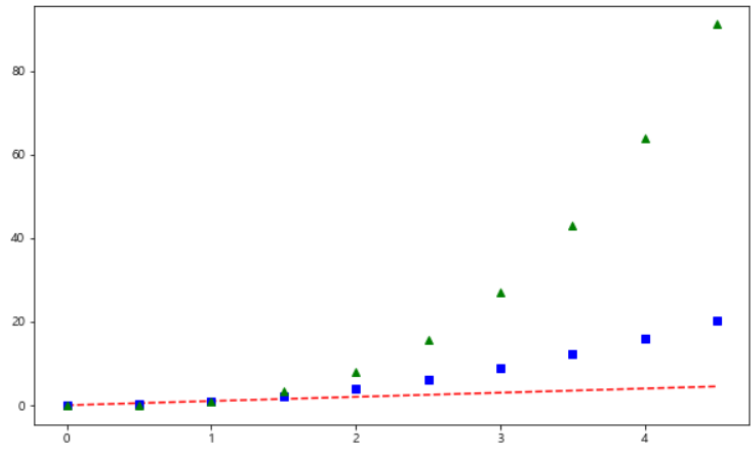
In [367]:
```
plt.figure(figsize=(10,6))
plt.plot(t, np.sin(t), lw=3, label='sin')  # lw = 3  선 두께
plt.plot(t, np.cos(t), 'r', label='cos')   # 'r'  선색
plt.grid()
plt.legend()
plt.xlabel('time')
plt.ylabel('Amplitude')
plt.title('Example of sinewave')
plt.show()
```



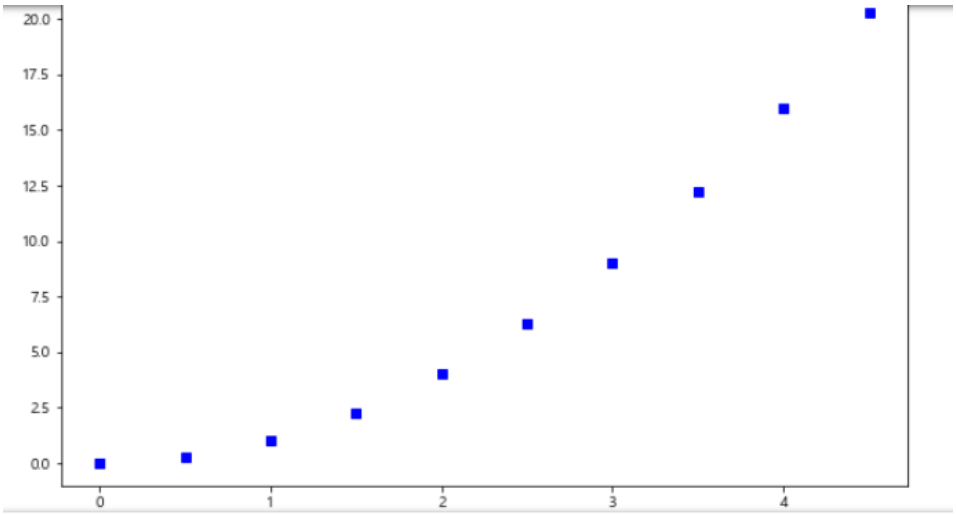
In [368]:
```
plt.figure(figsize=(10,6))
plt.plot(t, np.sin(t), lw=3, label='sin')
plt.plot(t, np.cos(t), 'r', label='cos')
plt.grid()
plt.legend()
plt.xlabel('time')
plt.ylabel('Amplitude')
plt.title('Example of sinewave')
plt.ylim(-1.2, 1.2)   #x 축 범위 제한
plt.xlim(0, np.pi)    #y 축 범위 제한
plt.show()
```

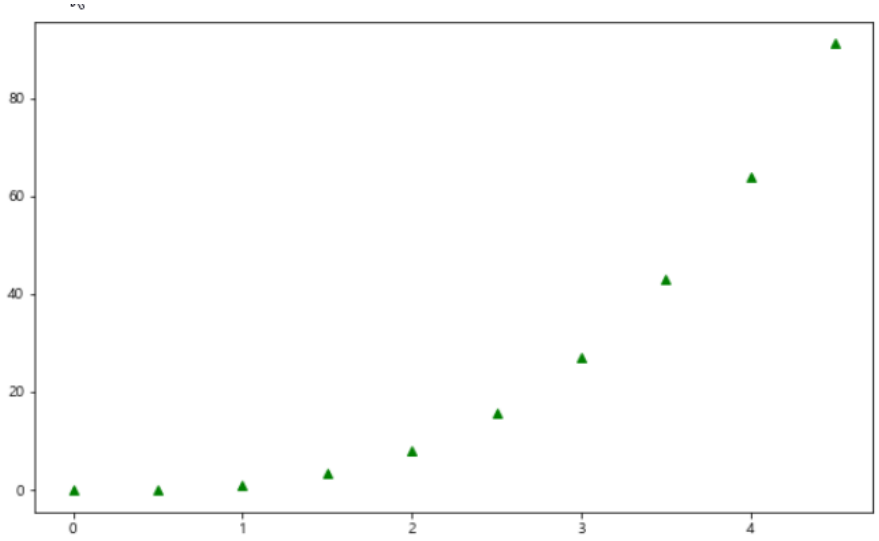


In [369]:
```
t = np.arange(0, 5, 0.5)  # 난수 발생.  (0부터 5까지 0.5간격)

plt.figure(figsize=(10,6))
plt.plot(t, t, 'r--')    #x축 데이터:t   ,  y축 데이터: t          red --
plt.plot(t, t**2, 'bs')  #x축 데이터:t   ,  y축 데이터: t**2  |    blue square
plt.plot(t, t**3, 'g^')  #x축 데이터:t   ,  y축 데이터: t**3       green ^
plt.show()
```

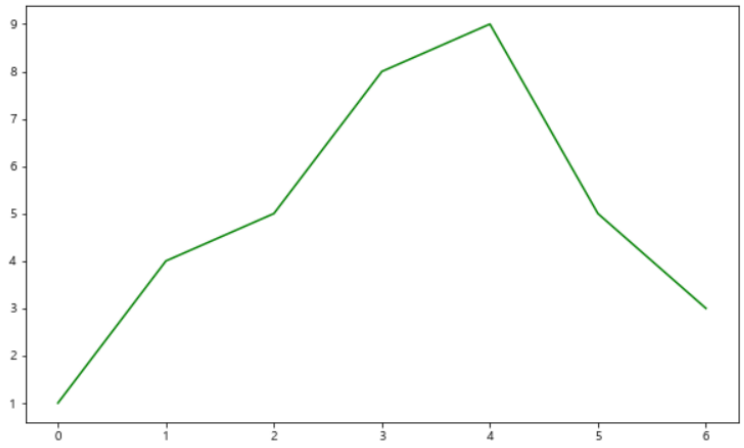


In [370]:
```
t = np.arange(0, 5, 0.5)  # 난수 발생.  (0부터 5까지 0.5간격)

plt.figure(figsize=(10,6))
pl1 = plt.plot(t, t**2, 'bs')  #x축 데이터:t   ,  y축 데이터: t**2       blue square

plt.figure(figsize=(10,6))
pl2 = plt.plot(t, t**3, 'g^')  #x축 데이터:t   ,  y축 데이터: t**3       green ^

plt.show()
```

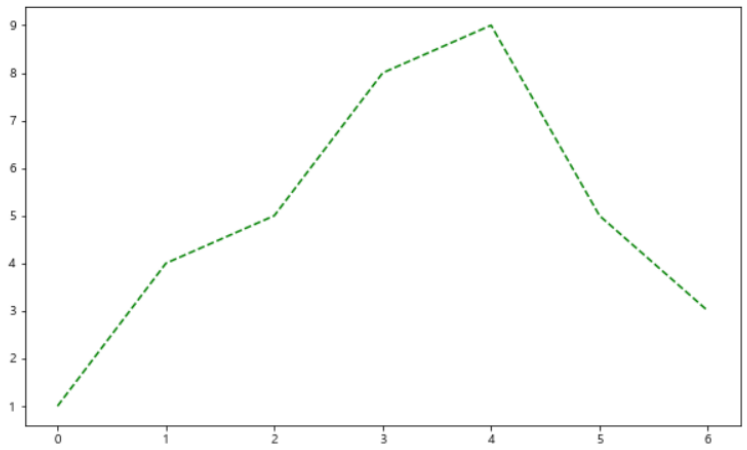




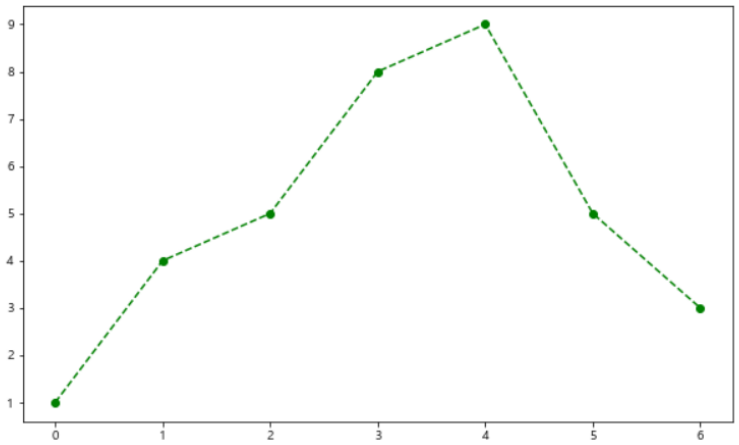
In [371]:
```
t = [0, 1, 2, 3, 4, 5, 6]
y = [1, 4, 5, 8, 9, 5, 3]

plt.figure(figsize=(10,6))
plt.plot(t, y, color='green')
plt.show()
```



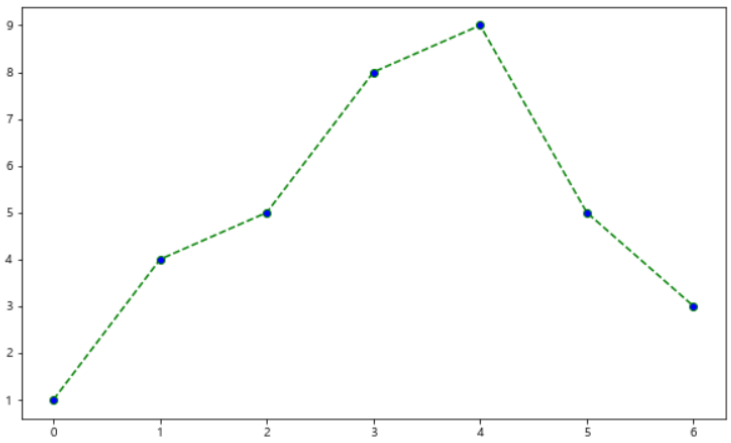
In [372]:
```
plt.figure(figsize=(10,6))
plt.plot(t, y, color='green', linestyle='dashed')
plt.show()
```



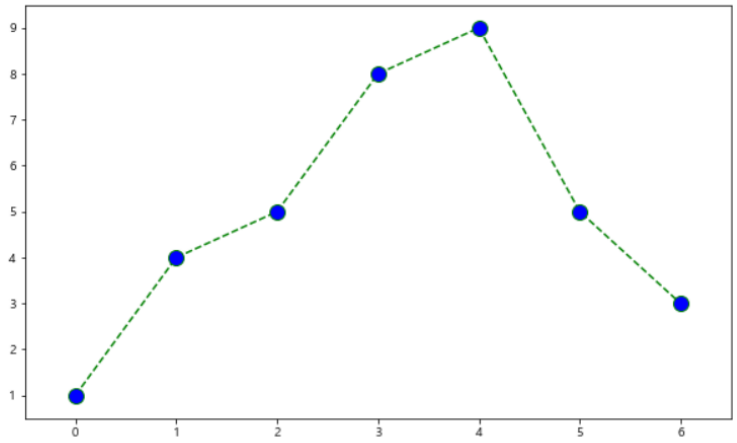
In [373]:
```
plt.figure(figsize=(10,6))
plt.plot(t, y, color='green', linestyle='dashed', marker='o')
plt.show()
```



In [374]:
```
plt.figure(figsize=(10,6))
plt.plot(t, y, color='green', linestyle='dashed', marker='o',
         markerfacecolor = 'blue')
plt.show()
```

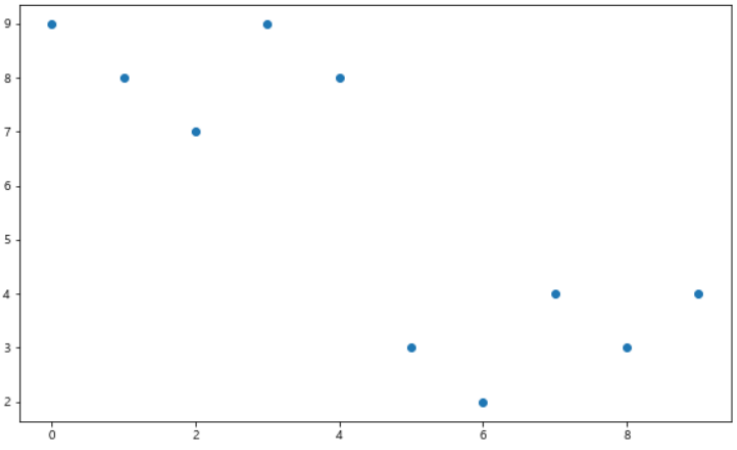


In [375]:
```
plt.figure(figsize=(10,6))
plt.plot(t, y, color='green', linestyle='dashed', marker='o',
        markerfacecolor = 'blue', markersize=12)

plt.xlim([-0.5, 6.5])
plt.ylim([0.5, 9.5])
plt.show()
```

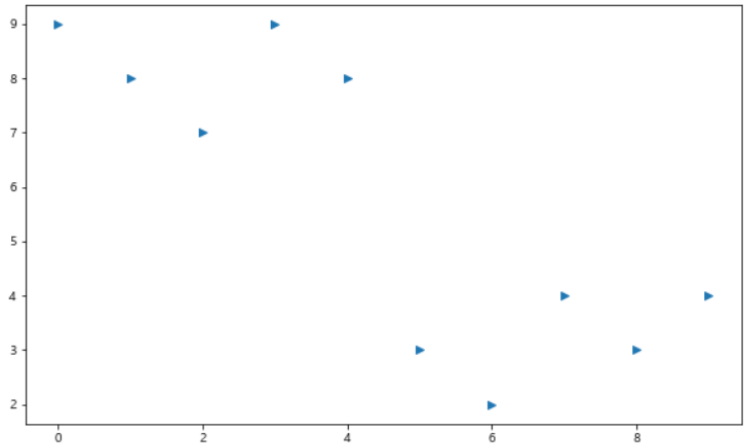


In [376]:
```
t = np.array([0,1,2,3,4,5,6,7,8,9])
y = np.array([9,8,7,9,8,3,2,4,3,4])
```

In [377]:
```
plt.figure(figsize=(10,6))
plt.scatter(t,y)
plt.show()
```



In [378]:
```
plt.figure(figsize=(10,6))
plt.scatter(t,y, marker='>')
plt.show()
```



Matplotlib은 한글 지원하지 않아, 한글 폰트 등록해야 함.

In [387]:
```
import platform

from matplotlib import font_manager, rc
plt.rcParams['axes.unicode_minus'] = False

if platform.system() == 'Darwin':
    rc('font', family='AppleGothic')
elif platform.system() == 'Windows':
    path = "c:/Windows/Fonts/malgun.ttf"
    font_name = font_manager.FontProperties(fname=path).get_name()
    rc('font', family=font_name)
else:
    print('Unknown system... sorry~~~~')
```