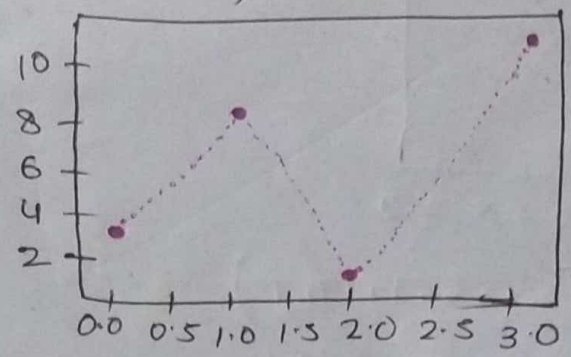
Format Strings fmt

format strings contain "replacement fields" surrounded by curly braces {}.

Marker|line|color

```
ypoints = np.array([3,8,1,10])
plt.plot(ypoints, 'o:r')
plt.show().
```



Line Syntax

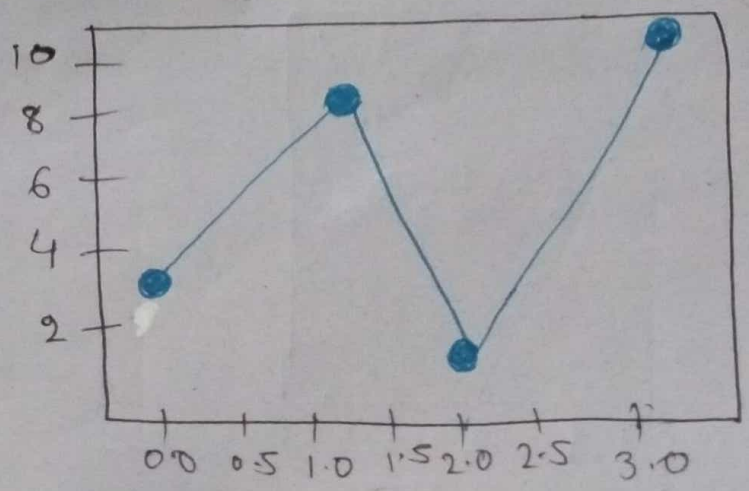
'-'	Solid line	'--'	Dashed line
':'	Dotted line	'-.'	Dashed / dotted line

Color Syntax

'r'	red	'k'	Black
'g'	green	'w'	white
'b'	blue		
'c'	Cyan		
'm'	Magenta		
'y'	Yellow		

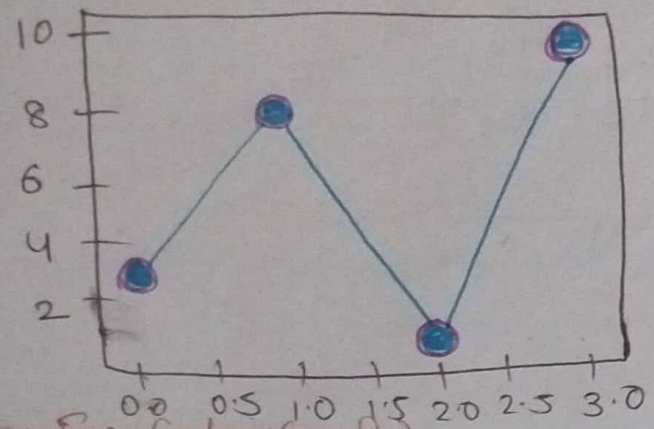
Marker Size

```
ypoints = np.array([3,8,1,10])
plt.plot(ypoints, marker='o', ms=20)
plt.show()
```



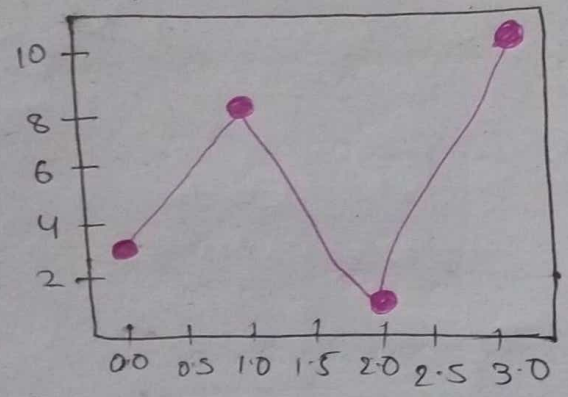
Marker Color

```
ypoints = np.array([3, 8, 1, 10])
plt.plot(ypoints, marker='o', ms=20, mec='r')
plt.show()
```



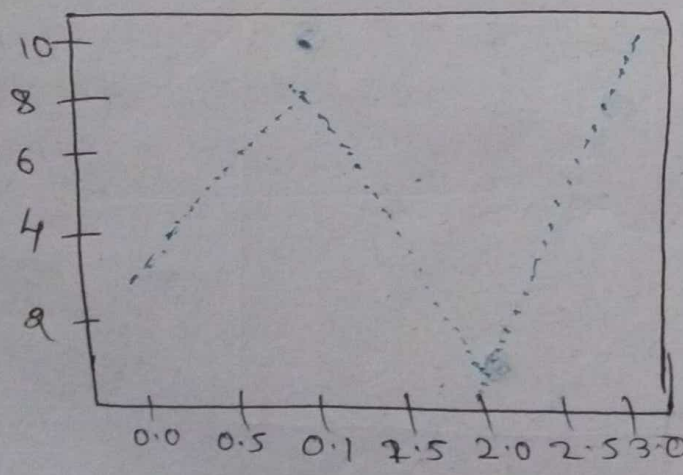
Marker Face Color (mfc)

```
ypoints = np.array([3, 8, 1, 10])
plt.plot(ypoints, marker='o', ms=20, mec='r',
         mfc='r')
plt.show()
```



Matplotlib Line (ls) Argument linestyle ('ls')

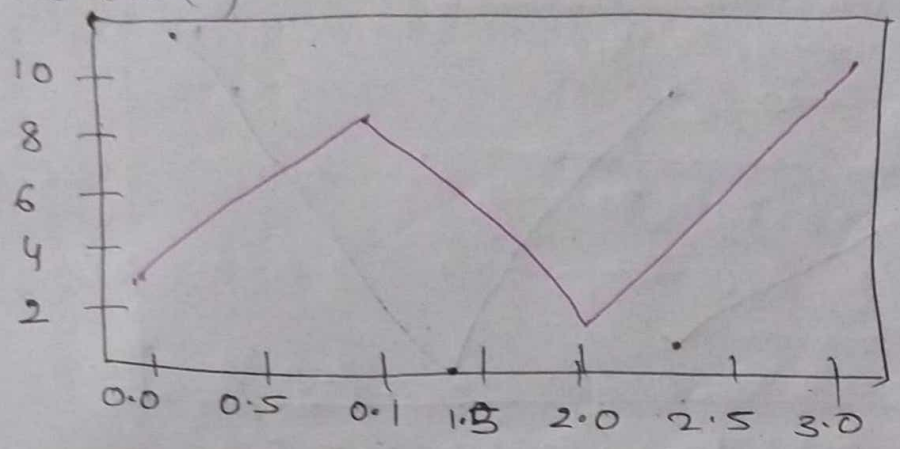
```
ypoints = np.array([3, 8, 1, 10])
plt.plot(ypoints, ls = 'dotted')
plt.show()
```



Line Color

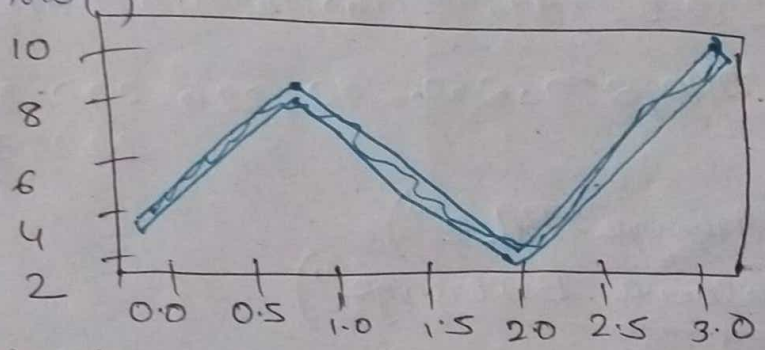
We can use the keyword argument color or the shorter 'c' to set the color of the line

```
ypoints = np.array([3,5,1,10])
plt.plot(ypoints, c = 'r')
plt.show()
```



Linewidth (lw)

```
ypoints = np.array([3,8,1,10])
plt.plot(ypoints, linewidth = '20.5')
plt.show()
```



Multiple Lines

```
y1 = np.array([3,8,1,10])
y2 = np.array([6,2,7,1])
plt.plot(y1)
plt.plot(y2)
plt.show()
```

```
x1 = np.array([0,1,2,3])
y1 = np.array([3,8,1,10])
x2 = np.array([0,1,2,3])
y2 = np.array([6,2,7,11])
plt.plot(x1,y1,x2,y2)
plt.show()
```