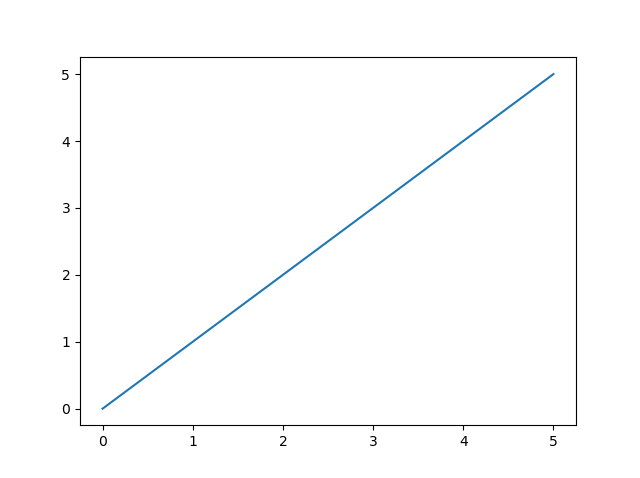
Matplotlib



## **What is Matplotlib?**

Matplotlib is a low level graph plotting library in python that serves as a visualization utility.

Matplotlib was created by John D. Hunter.

Matplotlib is open source and we can use it freely.

Matplotlib is mostly written in python, a few segments are written in C, Objective-C and Javascript for Platform compatibility.

Matplotlib Pyplot

## **Pyplot**

Most of the Matplotlib utilities lies under the pyplot submodule, and are usually imported under the plt alias:

import matplotlib.pyplot as plt

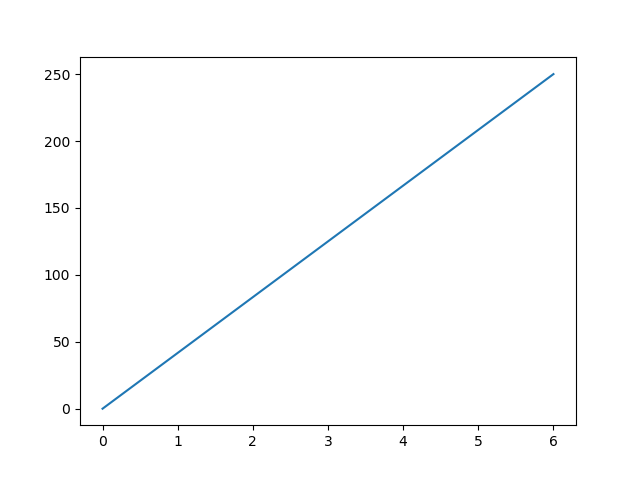
Now the Pyplot package can be referred to as plt.

### **Example**[**Get your own Python Server**](https://www.w3schools.com/python/python_server.asp)

Draw a line in a diagram from position (0,0) to position (6,250):

import matplotlib.pyplot as plt  
import numpy as np  
  
xpoints = np.array([0, 6])  
ypoints = np.array([0, 250])  
  
plt.plot(xpoints, ypoints)  
plt.show()

### **Result:**



You will learn more about drawing (plotting) in the next chapters.

# Matplotlib Plotting

## **Plotting x and y points**

The plot() function is used to draw points (markers) in a diagram.

By default, the plot() function draws a line from point to point.

The function takes parameters for specifying points in the diagram.

Parameter 1 is an array containing the points on the **x-axis**.

Parameter 2 is an array containing the points on the **y-axis**.

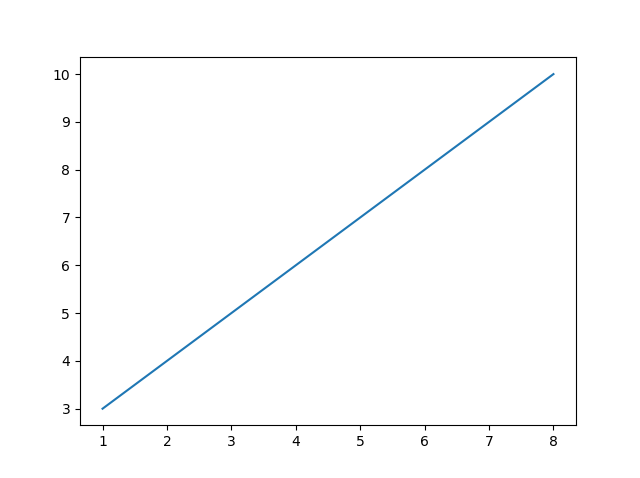
If we need to plot a line from (1, 3) to (8, 10), we have to pass two arrays [1, 8] and [3, 10] to the plot function.

### **Example**[**Get your own Python Server**](https://www.w3schools.com/python/python_server.asp)

Draw a line in a diagram from position (1, 3) to position (8, 10):

import matplotlib.pyplot as plt  
import numpy as np  
  
xpoints = np.array([1, 8])  
ypoints = np.array([3, 10])  
  
plt.plot(xpoints, ypoints)  
plt.show()

### **Result:**



[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_plotting1)

The **x-axis** is the horizontal axis.

The **y-axis** is the vertical axis.

## **Plotting Without Line**

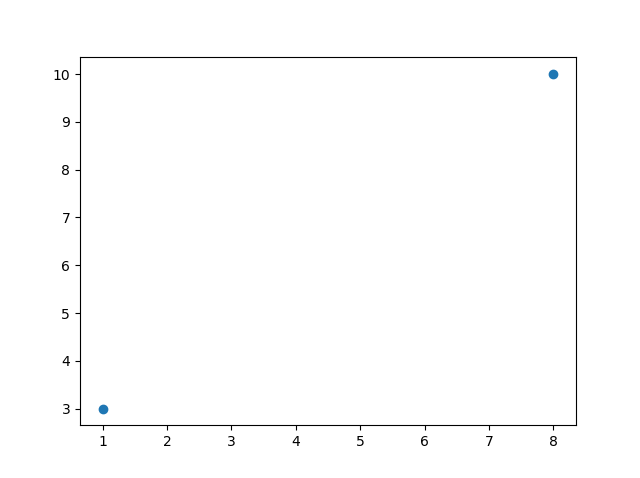
To plot only the markers, you can use shortcut string notation parameter 'o', which means 'rings'.

### **Example**

Draw two points in the diagram, one at position (1, 3) and one in position (8, 10):

import matplotlib.pyplot as plt  
import numpy as np  
  
xpoints = np.array([1, 8])  
ypoints = np.array([3, 10])  
  
plt.plot(xpoints, ypoints, 'o')  
plt.show()

### **Result:**



You will learn more about markers in the next chapter.

## **Multiple Points**

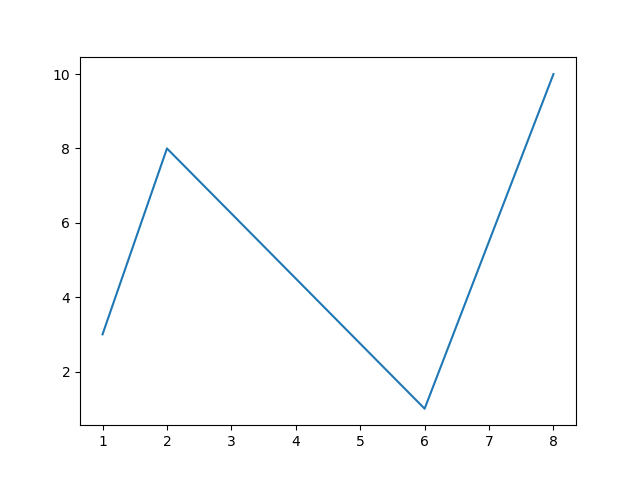
You can plot as many points as you like, just make sure you have the same number of points in both axis.

### **Example**

Draw a line in a diagram from position (1, 3) to (2, 8) then to (6, 1) and finally to position (8, 10):

import matplotlib.pyplot as plt  
import numpy as np  
  
xpoints = np.array([1, 2, 6, 8])  
ypoints = np.array([3, 8, 1, 10])  
  
plt.plot(xpoints, ypoints)  
plt.show()

### **Result:**



## **Default X-Points**

If we do not specify the points on the x-axis, they will get the default values 0, 1, 2, 3 etc., depending on the length of the y-points.

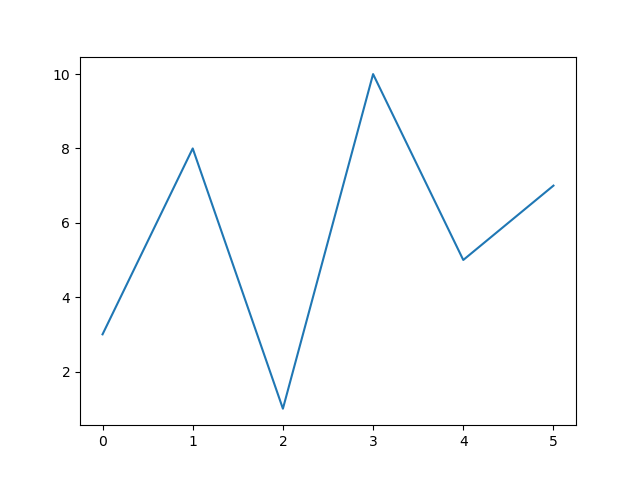
So, if we take the same example as above, and leave out the x-points, the diagram will look like this:

### **Example**

Plotting without x-points:

import matplotlib.pyplot as plt  
import numpy as np  
  
ypoints = np.array([3, 8, 1, 10, 5, 7])  
  
plt.plot(ypoints)  
plt.show()

### **Result:**



The **x-points** in the example above are [0, 1, 2, 3, 4, 5].

## **Markers**

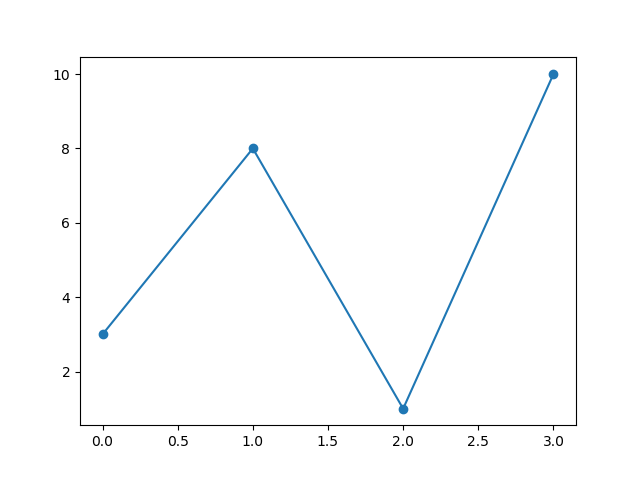
You can use the keyword argument marker to emphasize each point with a specified marker:

### **Example**[**Get your own Python Server**](https://www.w3schools.com/python/python_server.asp)

Mark each point with a circle:

import matplotlib.pyplot as plt  
import numpy as np  
  
ypoints = np.array([3, 8, 1, 10])  
  
plt.plot(ypoints, marker = 'o')  
plt.show()

### **Result:**



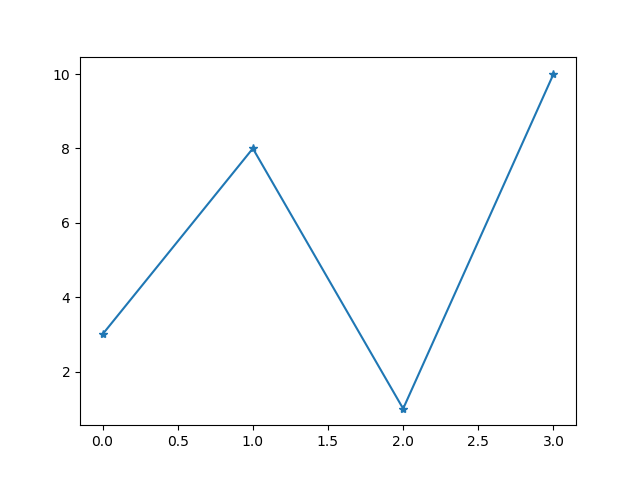
[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker1)

### **Example**

Mark each point with a star:

...  
plt.plot(ypoints, marker = '\*')  
...

### **Result:**



[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker2)

## **Marker Reference**

You can choose any of these markers:

|  |  |  |
| --- | --- | --- |
| **Marker** | **Description** | |
| 'o' | Circle | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_o) |
| '\*' | Star | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_star) |
| '.' | Point | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_point) |
| ',' | Pixel | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_pixel) |
| 'x' | X | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_x) |
| 'X' | X (filled) | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_x_filled) |
| '+' | Plus | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_plus) |
| 'P' | Plus (filled) | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_plus_filled) |
| 's' | Square | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_s) |
| 'D' | Diamond | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_D) |
| 'd' | Diamond (thin) | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_thin_d) |
| 'p' | Pentagon | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_p) |
| 'H' | Hexagon | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_H) |
| 'h' | Hexagon | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_hexagon) |
| 'v' | Triangle Down | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_v) |
| '^' | Triangle Up | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_triangle_up) |
| '<' | Triangle Left | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_triangle_left) |
| '>' | Triangle Right | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_triangle_right) |
| '1' | Tri Down | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_1) |
| '2' | Tri Up | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_2) |
| '3' | Tri Left | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_3) |
| '4' | Tri Right | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_4) |
| '|' | Vline | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_vline) |
| '\_' | Hline | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_hline) |

## **Format Strings fmt**

You can also use the shortcut string notation parameter to specify the marker.

This parameter is also called fmt, and is written with this syntax:

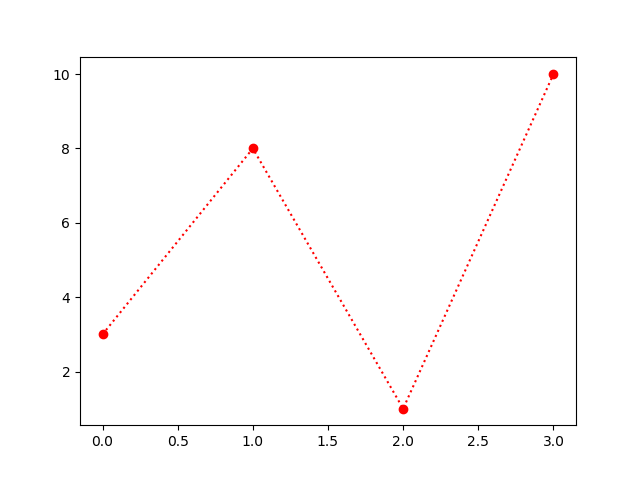
marker|line|color

### **Example**

Mark each point with a circle:

import matplotlib.pyplot as plt  
import numpy as np  
  
ypoints = np.array([3, 8, 1, 10])  
  
plt.plot(ypoints, 'o:r')  
plt.show()

### **Result:**



[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker3)

The marker value can be anything from the Marker Reference above.

The line value can be one of the following:

## **Line Reference**

|  |  |  |
| --- | --- | --- |
| **Line Syntax** | **Description** | |
| '-' | Solid line | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_fmt_line_solid) |
| ':' | Dotted line | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_fmt_line_dot) |
| '--' | Dashed line | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_fmt_line_dash) |
| '-.' | Dashed/dotted line | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_fmt_line_dashdot) |

**Note:** If you leave out the line value in the fmt parameter, no line will be plotted.

The short color value can be one of the following:

## **Color Reference**

|  |  |  |
| --- | --- | --- |
| **Color Syntax** | **Description** | |
| 'r' | Red | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_fmt_r) |
| 'g' | Green | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_fmt_g) |
| 'b' | Blue | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_fmt_b) |
| 'c' | Cyan | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_fmt_c) |
| 'm' | Magenta | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_fmt_m) |
| 'y' | Yellow | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_fmt_y) |
| 'k' | Black | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_fmt_k) |
| 'w' | White | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_fmt_w) |

## **Marker Size**

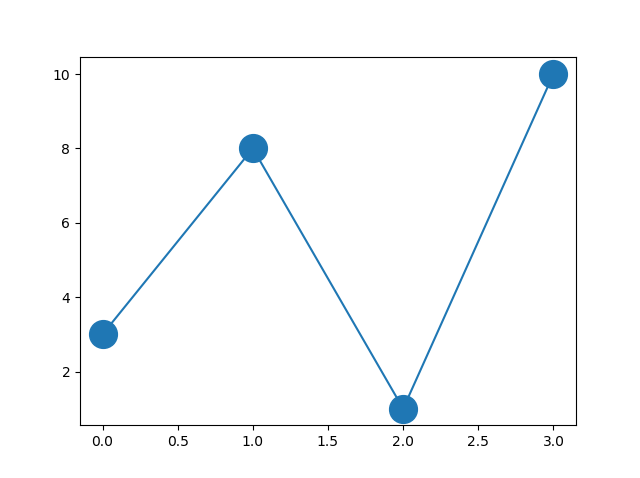
You can use the keyword argument markersize or the shorter version, ms to set the size of the markers:

### **Example**

Set the size of the markers to 20:

import matplotlib.pyplot as plt  
import numpy as np  
  
ypoints = np.array([3, 8, 1, 10])  
  
plt.plot(ypoints, marker = 'o', ms = 20)  
plt.show()

### **Result:**



[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_size)

## **Marker Color**

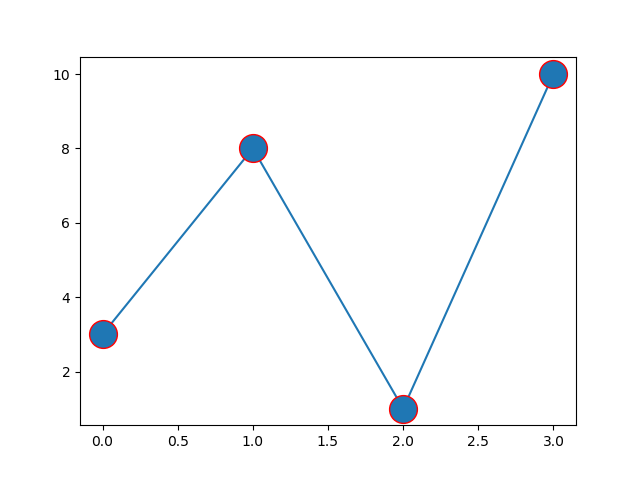
You can use the keyword argument markeredgecolor or the shorter mec to set the color of the edge of the markers:

### **Example**

Set the EDGE color to red:

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

### **Result:**



[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_color1)

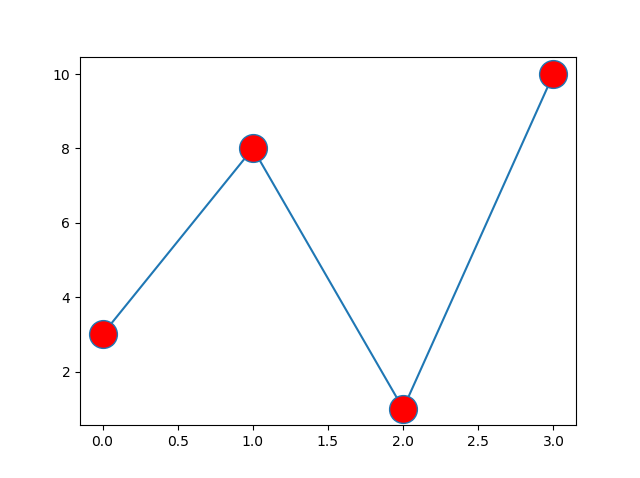
You can use the keyword argument markerfacecolor or the shorter mfc to set the color inside the edge of the markers:

### **Example**

Set the FACE color to red:

import matplotlib.pyplot as plt  
import numpy as np  
  
ypoints = np.array([3, 8, 1, 10])  
  
plt.plot(ypoints, marker = 'o', ms = 20, mfc = 'r')  
plt.show()

### **Result:**



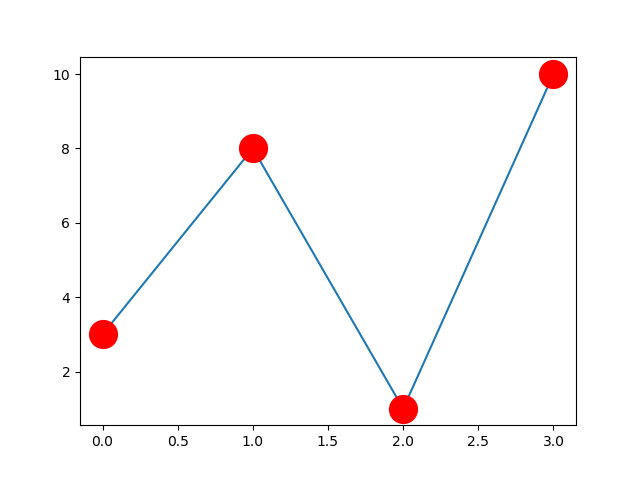
Use both the mec and mfc arguments to color the entire marker:

### **Example**

Set the color of both the edge and the face to red:

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

### **Result:**



[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_color3)

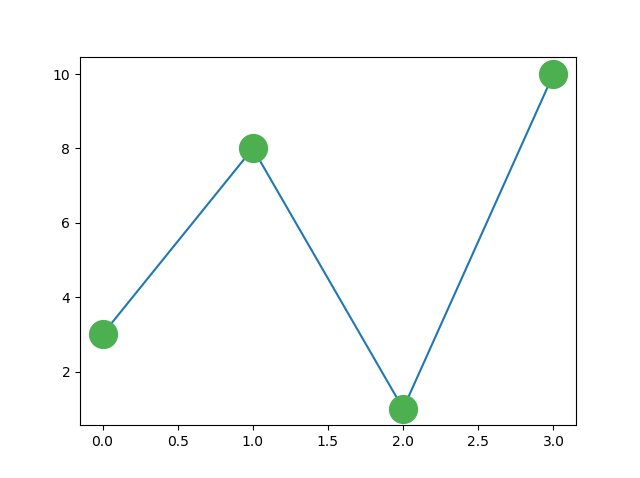
You can also use [Hexadecimal color values](https://www.w3schools.com/colors/colors_hexadecimal.asp):

### **Example**

Mark each point with a beautiful green color:

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

### **Result:**



[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_color_green)

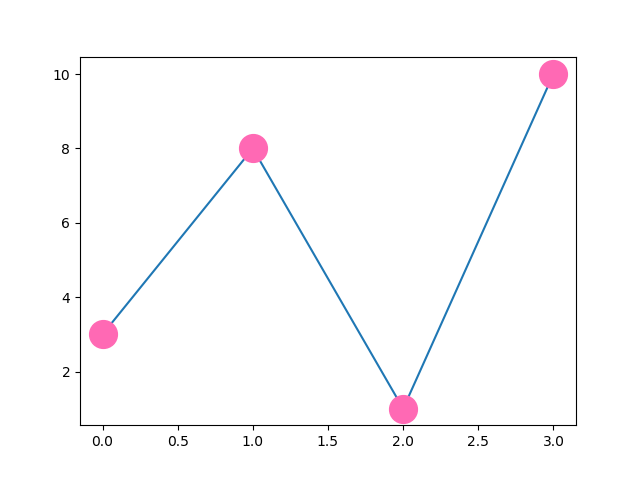
Or any of the [140 supported color names](https://www.w3schools.com/colors/colors_names.asp).

### **Example**

Mark each point with the color named "hotpink":

...  
plt.plot(ypoints, marker = 'o', ms = 20, mec = 'hotpink', mfc = 'hotpink')  
...

### **Result:**



[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_matplotlib_marker_color_hotpink)