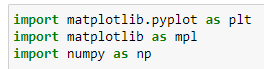
**Matplotlib tutorial**

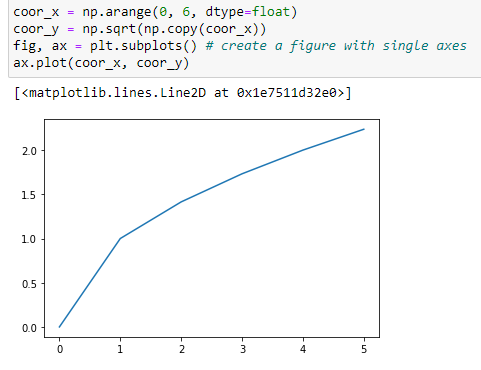
# ***Basic Usage***

This tutorial covers some basic usage patterns and best practices to help started with Matplotlib. For started:



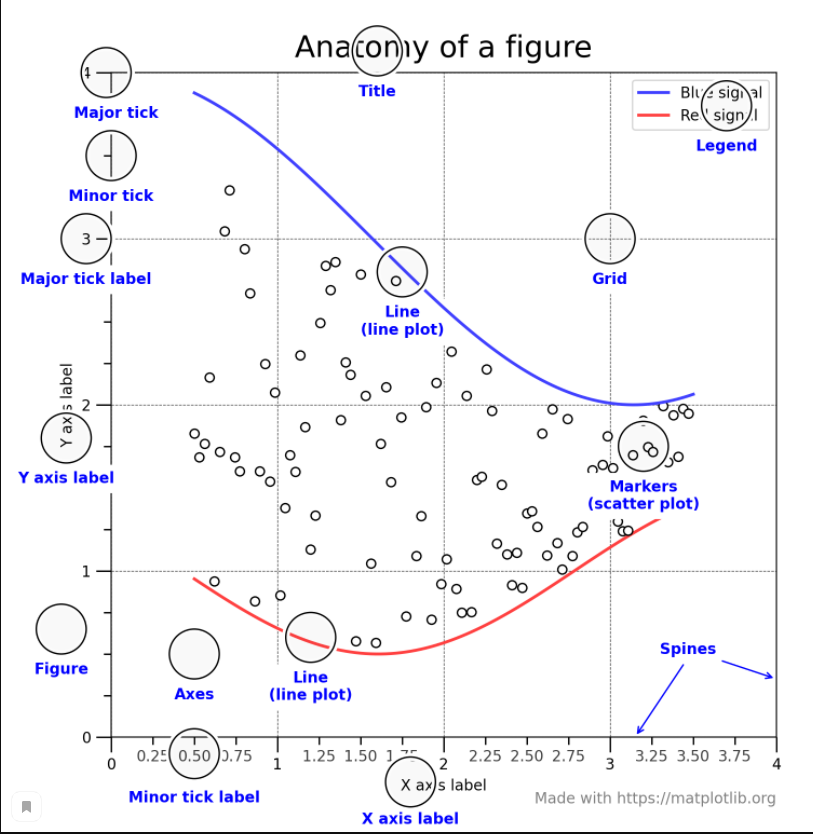
# ***A simple example***

First step is create figure and axes:



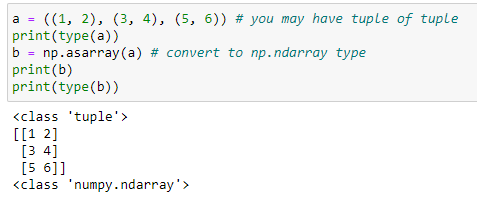
# ***Parts of a Figure***

Below you can see components of a Matplotlib Figure:

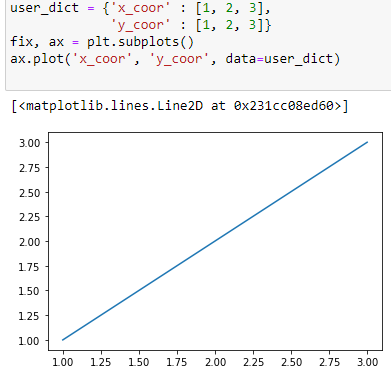


# ***Types of inputs to plotting functions***

Common convention as to convert array-like object into np.array object for plotting.

You can convert array-like object with help *np.asarray()* function. For example:  


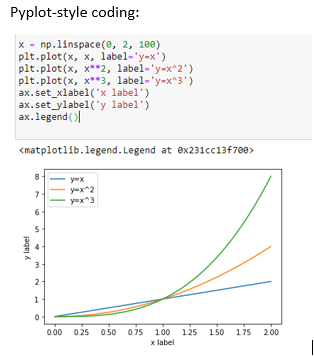
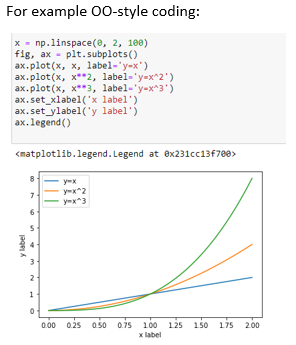
You can specify dictionary and his keywords for create graph:



# ***Coding styles***

Essentially you have two ways to use Matplotlib:

* Explicitly create Figures and Axes, and call methods on them
* Rely on pyplot to automatically create and manage the Figures and Axes, and use pyplot functions for plotting



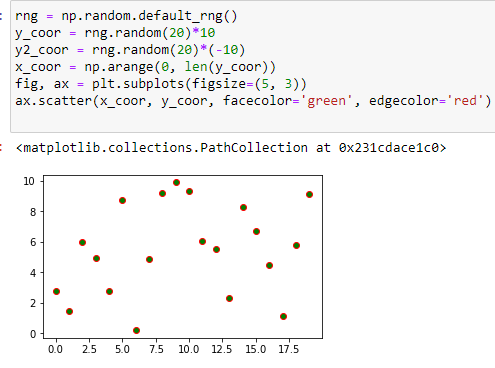
Note that, you may find older examples that use the pylab. This approach is strongly deprecated.

# ***Styling Artists***

Most plotting methods have styling options. We can specify, example: *color, linewidth, linestyle…*



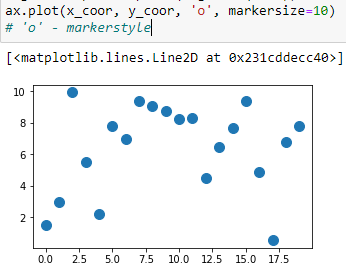
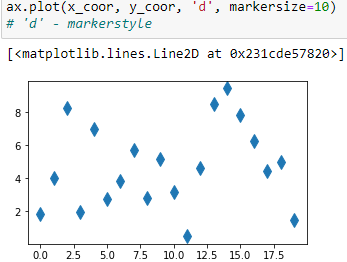
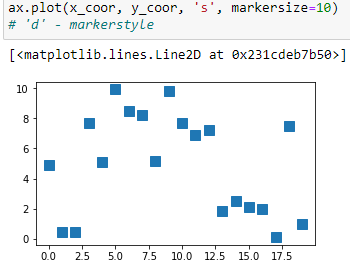
Matplotlib has a very flexible array of colors. Some Artists will take multiple color, example, for a scatter plot. The color of edge and interior can be different. See below:



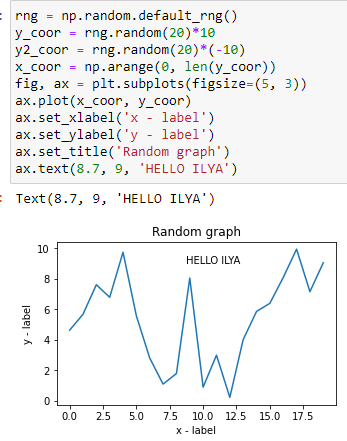
For all colors you can see: *colors tutorial.* Also, you can specify linewidths, linestyle, markersize, markerstyle.

Markerstyle:

* ‘o’ - circle
* ‘d’ - diamond
* ‘v’ - triangle
* ‘s’ - square

# ***Labelling plots***

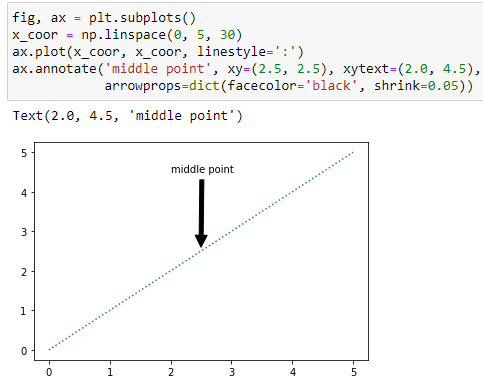
This section covers: *set\_xlabel, set\_ylabel, set\_title, text.* These functions allow add text on plot.

set\_xlabel – name of axis X

set\_ylabel – name of axis Y

set\_title – name of plot

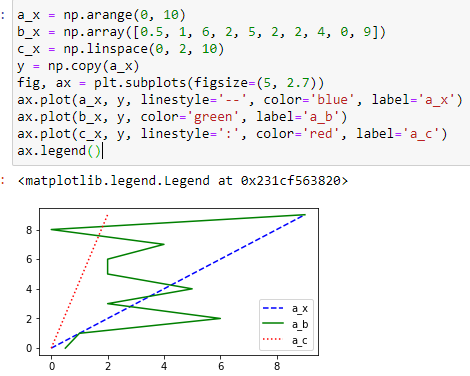
We can also annotate points on a plot, often by connecting an arrow pointing *xy*, to a piece of text at *xytext:*

*xy –* coordinate for pointing

*xytext –* coordinate for text

*arrowprops –* it`s arrow between the positions `xy` and `xytext` will be centered on the text.

Often, we want to identify lines or markers on plot. We can use *ax.legeng():*



# ***Axis scales and ticks***