1. Matplotlib Styles: There are various built-in styles in style package, and we can also write customized style files and, then, to use those styles all you need to import them and apply on the graphs and plots. In this way, we need not write various lines of code for each plot individually again and again i.e. the code is reusable whenever required.

Importing styles:

from matplotlib import style

Example:

A)**import** numpy as np

**import** matplotlib.pyplot as plt

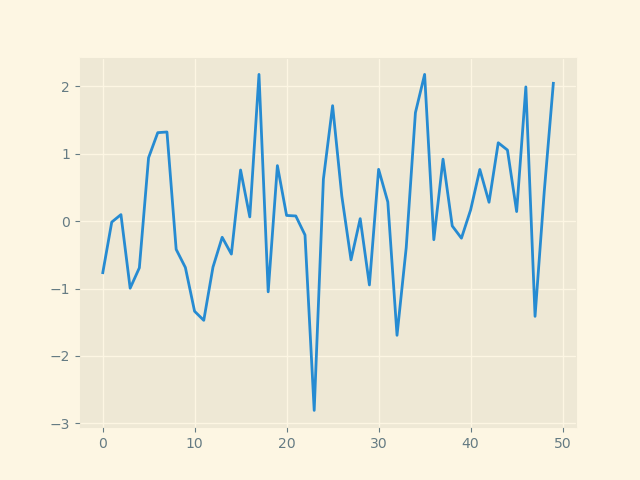
**from** matplotlib **import** style

data **=** np.random.randn(50)

plt.style.use('Solarize\_Light2')

plt.plot(data)

plt.show()



B)**import** numpy as np

**import** matplotlib.pyplot as plt

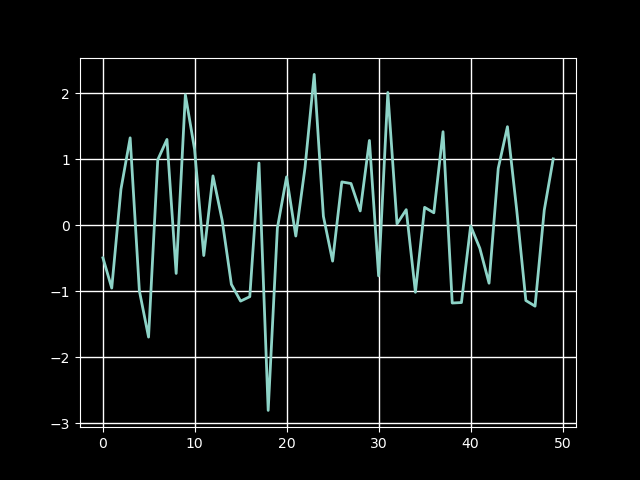
**from** matplotlib **import** style

data **=** np.random.randn(50)

plt.style.use('dark\_background')

plt.plot(data)

plt.show()



C)**import** numpy as np

**import** matplotlib.pyplot as plt

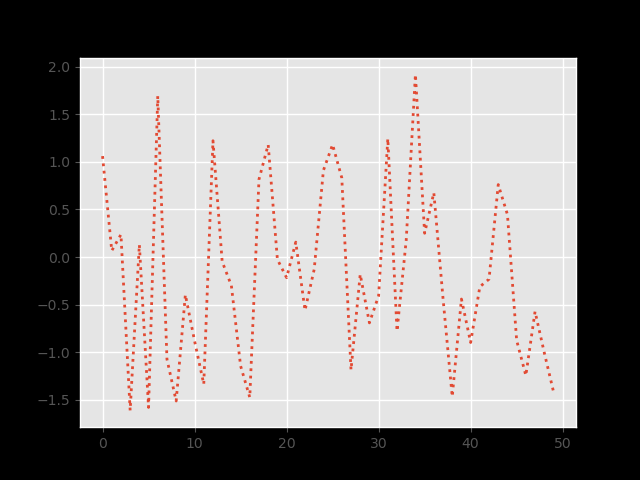
**from** matplotlib **import** style

data **=** np.random.randn(50)

plt.style.use('ggplot')

plt.plot(data, linestyle**=**":", linewidth**=**2)

plt.show()

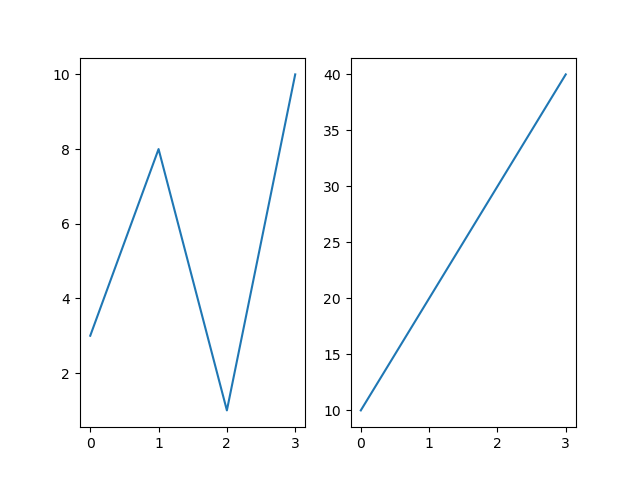


1. Subplots:With the subplots() function you can draw multiple plots in one figure.

Example:

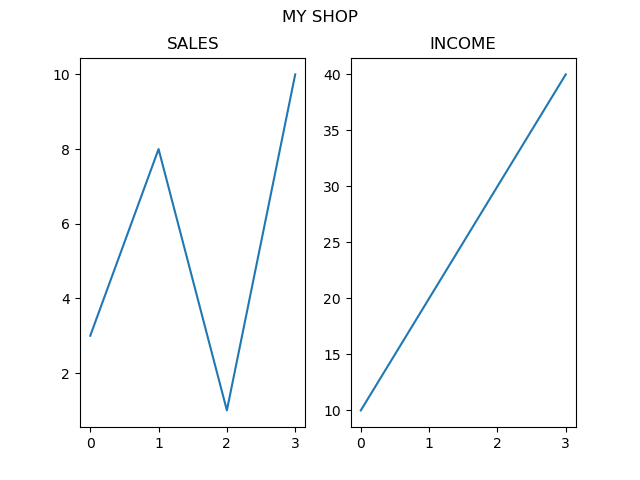
A)

import matplotlib.pyplot as plt  
import numpy as np  
  
#plot 1:  
x = np.array([0, 1, 2, 3])  
y = np.array([3, 8, 1, 10])  
  
plt.subplot(1, 2, 1)  
plt.plot(x,y)  
  
#plot 2:  
x = np.array([0, 1, 2, 3])  
y = np.array([10, 20, 30, 40])  
  
plt.subplot(1, 2, 2)  
plt.plot(x,y)  
  
plt.show()



B)

import matplotlib.pyplot as plt  
import numpy as np  
  
#plot 1:  
x = np.array([0, 1, 2, 3])  
y = np.array([3, 8, 1, 10])  
  
plt.subplot(1, 2, 1)  
plt.plot(x,y)  
plt.title("SALES")  
  
#plot 2:  
x = np.array([0, 1, 2, 3])  
y = np.array([10, 20, 30, 40])  
  
plt.subplot(1, 2, 2)  
plt.plot(x,y)  
plt.title("INCOME")  
  
plt.suptitle("MY SHOP")  
plt.show()



1. Multiple plots, charts, figures: The two integer arguments to this function specify the number of rows and columns of the subplot grid. The function returns a figure object and a tuple containing axes objects equal to nrows\*ncols. Each axes object is accessible by its index.

Using of subplots :

Plt.subplots(nrows, ncols)

Example:

import matplotlib.pyplot as plt

fig,a = plt.subplots(2,2)

import numpy as np

x = np.arange(1,5)

a[0][0].plot(x,x\*x)

a[0][0].set\_title('square')

a[0][1].plot(x,np.sqrt(x))

a[0][1].set\_title('square root')

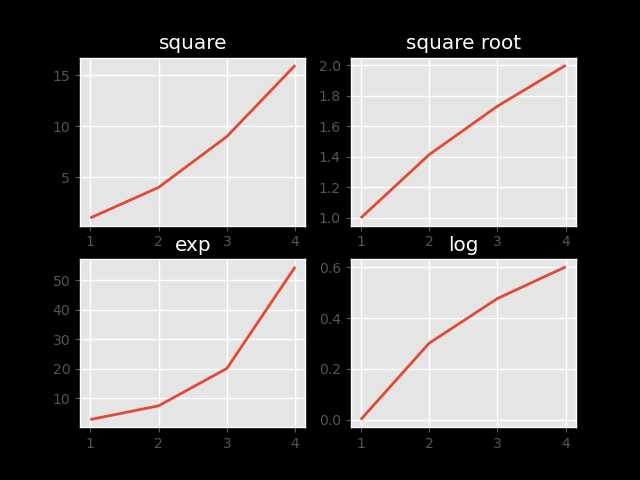
a[1][0].plot(x,np.exp(x))

a[1][0].set\_title('exp')

a[1][1].plot(x,np.log10(x))

a[1][1].set\_title('log')

plt.show()



1. Using subplot to grid:

import matplotlib.pyplot as plt

a1 = plt.subplot2grid((3,3),(0,0),colspan = 2)

a2 = plt.subplot2grid((3,3),(0,2), rowspan = 3)

a3 = plt.subplot2grid((3,3),(1,0),rowspan = 2, colspan = 2)

import numpy as np

x = np.arange(1,10)

a2.plot(x, x\*x)

a2.set\_title('square')

a1.plot(x, np.exp(x))

a1.set\_title('exp')

a3.plot(x, np.log(x))

a3.set\_title('log')

plt.tight\_layout()

plt.show()

