**Membuat Line Plot dengan Matplotlib Python**

*Line plot* adalah salah satu jenis visualisasi data yang banyak digunakan dan merupakan jenis plot dasar dalam visualisasi data. Jenis plot ini menampilkan informasi berupa rangkaian titik data yang terhubung dengan segmen garis lurus.

Line plot dapat digunakan pada dataset yang memiliki nilai kontinu untuk melihat pergerakan data dari waktu ke waktu.

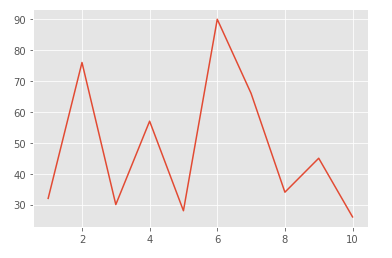
Disini kita akan membahas cara membuat plot sederhana dengan [line plot](https://matplotlib.org/3.1.0/api/_as_gen/matplotlib.pyplot.plot.html).

**Single Line**

Misalnya kita memiliki dua kumpulan data numerik yang disimpan dalam variabel x dan y. Kemudian kita ingin memvisualisasikannya menggunakan *line plot*. Perhatikan kode berikut.

import matplotlib.pyplot as plt  
%matplotlib inline  
  
plt.style.use('ggplot')  
  
x = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]  
y = [32, 76, 30, 57, 28, 90, 66, 34, 45, 26]  
  
plt.plot(x, y)  
plt.show()

**Output:**



Untuk membuat *line plot*, kita dapat menggunakan *method* .plot() dan memberikan argumen berupa data yang akan digunakan sebagai sumbu x dan y. Dalam hal ini, kita menggunakan data x untuk sumbu x dan data y untuk sumbu y, sehingga kodenya dapat kita tulis plt.plot(x,y).

plt.style.use('ggplot') adalah kode untuk menentukan *style* yang ingin digunakan untuk visualisasi data, dalah hal ini kita menggunakan ‘ggplot’ sebagai *style*-nya.

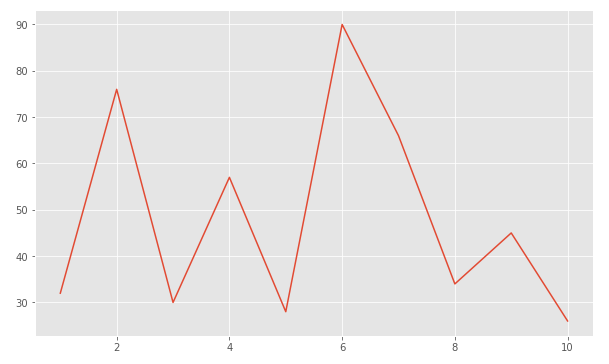
Ada berbagai macam *style* yang dapat digunakan. Anda dapat melihat *style* apa saja yang tersedia dengan menuliskan kode plt.style.available di Jupyter notebook.

Kode di atas adalah cara paling sederhana untuk membuat visualisasi dengan Matplotlib.

Selanjutnya saya akan membuat visualisasi yang sama namun dengan kode yang sedikit berbeda. Perhatikan kode berikut.

import matplotlib.pyplot as plt  
%matplotlib inline  
  
plt.style.use('ggplot')  
  
fig = plt.figure(figsize=(10,6))  
ax = fig.add\_subplot()  
  
ax.plot(x, y)  
plt.show()

**Output:**



Jika menggunakan kode seperti di atas, kita harus membuat *figure* terlebih dahulu dengan *method* .figure(). Kita dapat mengatur ukuran *figure* dengan parameter figsize, kemudian menambahkan *axes* ke dalam *figure* dengan fig.add\_subplot(). Setelah itu barulah kita membuat plot di dalam *axes* dengan ax.plot(x,y).

Cara ini terlihat lebih rumit dibanding sebelumnya, bukan? Ya, ini adalah cara *explicit* dalam membuat *line plot* ataupun jenis visualisasi data dengan Matplotlib lainnya, sedangkan yang sebelumnya adalah cara *implicit*. Cara *explicit*  sangat direkomendasikan jika ingin membuat visualisasi yang lebih kompleks.

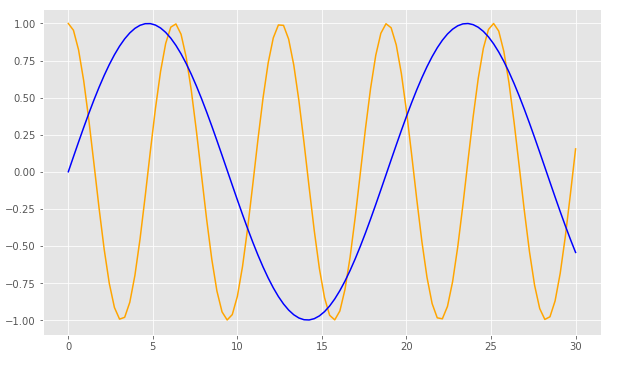
**Multiple Line**

Jika sebelumnya kita hanya menampilkan *single line* pada satu *axes*, kali ini kita akan menampilkan beberapa *line* dalam satu *axes*.

Misalnya kita ingin menampilkan visualisasi data dari fungsi matematika **cos** dan **sin** seperti di bawah ini.

import numpy as np  
import matplotlib.pyplot as plt  
%matplotlib inline  
  
plt.style.use('ggplot')  
  
X = np.linspace(0, 30, 100)  
y1 = np.cos(X)  
y2 = np.sin(X/3)  
  
fig, ax = plt.subplots(figsize=(10,6))  
  
ax.plot(X, y1, color='orange')  
ax.plot(X, y2, color='blue')  
  
plt.show()

**Output:**



Dari kode di atas, kita menggunakan *method* linspace() dari numpy untuk menghasilkan nilai dari 0 sampai 30 sebanyak 100 titik yang disimpan dalam variabel X untuk ditempatkan pada sumbu x. Selanjutnya untuk sumbu y, kita membuat dua buah data, yaitu nilai cos X yang disimpan dalam variabel y1 dan sin X/3 yang disimpan dalam variabel y2.

Disini kita menggunakan cara *explicit* untuk visualisasi data tersebut. Kita membuat dua buah plot di dalam *axes* yang sama yang dibedakan dengan warnanya.

ax.plot(X, y1, color=’orange’) menghasilkan garis cos(X) yang berwarna oranye, sedangkan ax.plot(X, y2, color=’blue’) menghasilkan garis sin(X/3) yang berwarna biru.