

MATPLOTLIB PART 1

MEMBUAT PLOT DAN PENYESUAIANNYA

Untuk memvisualisasikan data kita di python, kita bisa menggunakan library matplotlib

```
from matplotlib import pyplot as plt
```

Gambar 1. Import library matplotlib

Misalkan kita ingin membuat grafik yang menampilkan perbandingan umur dengan gaji sebagai developer. Gaji developer akan menjadi sumbu Y sedangkan umur akan menjadi sumbu X

```
age = [25,26, 27, 28, 29, 30, 31, 32, 33, 34, 35] #list of age 25 - 35 sebagai sumbu x  
dev_y = [38496, 42000, 46752, 49320, 53200, 56000, 62316, 64928, 67317, 68748, 73752] #salary sebagai sumbu y
```

Gambar 2. Data umur dan gaji developer

Untuk meng-inisialisasi plot, gunakan fungsi *plot()*. Untuk menampilkan plot atau grafik gunakan fungsi *show()*.

```
plt.plot(age,dev_y) #inisialisasi untuk membuat plot (standarnya line)  
plt.title('Median Salary (USD) by age') #memberi judul grafik  
plt.xlabel('Age') #label utk sumbu x  
plt.ylabel('Median Salary (USD)') #label utk sumbu y  
plt.show() #tampilkan plotnya
```

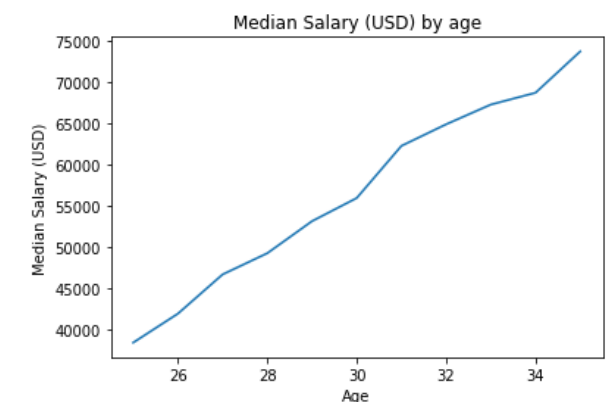
Gambar 3. Fungsi untuk menampilkan plot

Fungsi title berfungsi untuk menambahkan judul grafik

Fungsi xlabel berfungsi untuk menambahkan label di sumbu X

Fungsi ylabel berfungsi untuk menambahkan label di sumbu Y

Label-label ini berfungsi agar kita dapat membaca grafik dengan lebih mudah. Jika kita tidak memberi label tidak masalah hanya saja pembaca yang lain akan bingung apa maksud dari sumbu X dan Y tersebut, ataupun grafik ini menunjukkan tentang apa.



Gambar 4. Grafik perbandingan umur dan gaji developer

Dengan matplotlib, kita juga bisa menampilkan grafik lebih dari satu **jika nilai cumbu X nya sama**.

```
#dalam satu plot bisa memiliki 2 line grafik
age = [25,26, 27, 28, 29, 30, 31, 32, 33, 34, 35] #list of age 25 - 35 sebagai sumbu x
dev_y = [38496, 42000, 46752, 49320, 53200, 56000, 62316, 64928, 67317, 68748, 73752] #salary sebagai sumbu y

plt.plot(age,dev_y, label='All Devs') #inisialisasi untuk membuat plot (standarnya line)

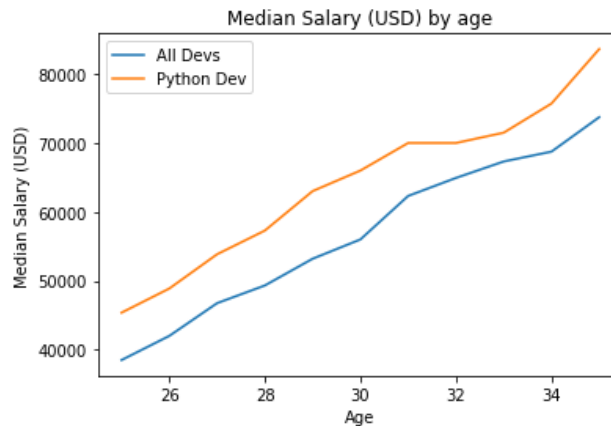
py_y = [45372, 48876, 53850, 57287, 63016, 65998, 70003, 70000, 71496, 75730, 83640] #range salary for python developer

plt.plot(x,py_y, label='Python Dev') #sb x menggunakan var yang sama tapi var y nya berbeda

plt.title('Median Salary (USD) by age') #memberi judul grafik
plt.xlabel('Age') #label utk sumbu x
plt.ylabel('Median Salary (USD)') #label utk sumbu y
plt.legend() #utk menampilkan label garis
plt.show() #tampilkan plotnya
```

Gambar 5. Snip code menampilkan 2 grafik

Berdasarkan potongan kode diatas, kita dapat melihat ada dua fungsi plot dan memiliki parameter sumbu Y yang berbeda namun memiliki sumbu X yang sama. Fungsi *label* di parameter plot berfungsi untuk memberi label pada masing-masing grafik agar user tidak bingung membaca grafiknya.



Gambar 6. Tampilan grafik dengan 2 line yang berbeda

Setelah kita mengerti dasar membuat grafik, selanjutnya kita bisa memodifikasi grafik dengan style yang telah disediakan matplotlib seperti color, linestyle, marker dan sebagainya.

```
#memodifikasi grafik
#jenis warna dan jenis garis dapat di dokumentasi matplotlib
#utk warna bisa menggunakan hex value

age = [25,26, 27, 28, 29, 30, 31, 32, 33, 34, 35] #list of age 25 - 35 sebagai sumbu x

y = [38496, 42000, 46752, 49320, 53200, 56000, 62316, 64928, 67317, 68748, 73752] #salary sebagai sumbu y
plt.plot(age,y, color='k', linestyle='--', marker='.',label='All Devs') #inisialisasi untuk membuat plot (standarnya line)

py_y = [45372, 48876, 53850, 57287, 63016, 65998, 70003, 70000, 71496, 75730, 83640] #range salary for python developer
plt.plot(age,py_y,color='b', marker='o', label='Python Dev') #sb x menggunakan var yang sama tapi var y nya berbeda

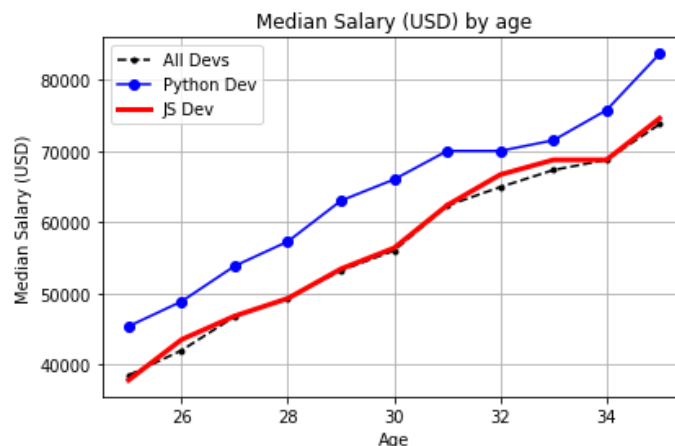
js_y = [37810, 43515, 46823, 49293, 53437, 56373, 62375, 66674, 68745, 68746, 74583]
plt.plot(age,js_y, color='r', linewidth='3',label='JS Dev')

plt.title('Median Salary (USD) by age') #memberi judul grafik
plt.xlabel('Age') #label utk sumbu x
plt.ylabel('Median Salary (USD)') #label utk sumbu y
plt.legend() #utk menampilkan label garis
plt.grid(True) #utk menambahkan grid di grafik
plt.tight_layout() #utk menambahkan padding
plt.show() #tampilkan plotnya
```

Gambar 7. Snip code modifikasi grafik

Fungsi untuk memodifikasi seperti mengubah warna line dan marker diketik di dalam parameter *plot*.

1. Color -> untuk mengubah warna, kode warna dapat dilihat di dokumentasi matplotlib namun bisa juga menggunakan **hex value**
2. Line style -> untuk mengubah jenis line yang di pakai dalam grafik seperti garis putus-putus, kode untuk linestyle dapat dilihat di dokumentasi matplotlib
3. Line width -> untuk mengubah tebal line
4. Marker -> untuk memberi penanda seperti tanda titik untuk setiap nilainya. Marker="." Artinya penanda yang digunakan berukuran kecil sedangkan marker = "o" artinya penanda yang digunakan berukuran besar
5. Plt.grid(True) -> grafik akan menambahkan grid agar user gampang membaca grafiknya
6. Plt.tight_layout() -> padding grafik akan ditambahkan sehingga kesannya lebih padat



Gambar 8. Grafik yang telah di modifikasi

Matplotlib telah menyediakan beberapa style, kita dapat melihat style apa saja yang disediakan oleh matplotlib dengan fungsi *style.available*

```
print(plt.style.available)

['Solarize_Light2', '_classic_test_patch', 'bmh', 'classic', 'dark_background', 'fast',
```

Gambar 9. Jenis-jenis style yang disediakan matplotlib

Jika kita menggunakan style yang disediakan matplotlib, kita tidak perlu menambahkan style secara manual lagi.

```
#cara menggunakan style yang disediakan matplotlib
plt.style.use('ggplot')

age = [25,26, 27, 28, 29, 30, 31, 32, 33, 34, 35] #list of age 25 - 35 sebagai sumbu x

y = [38496, 42000, 46752, 49320, 53200, 56000, 62316, 64928, 67317, 68748, 73752] #salary sebagai sumbu y
plt.plot(age,y, label='All Devs') #inisialisasi untuk membuat plot (standarnya line)

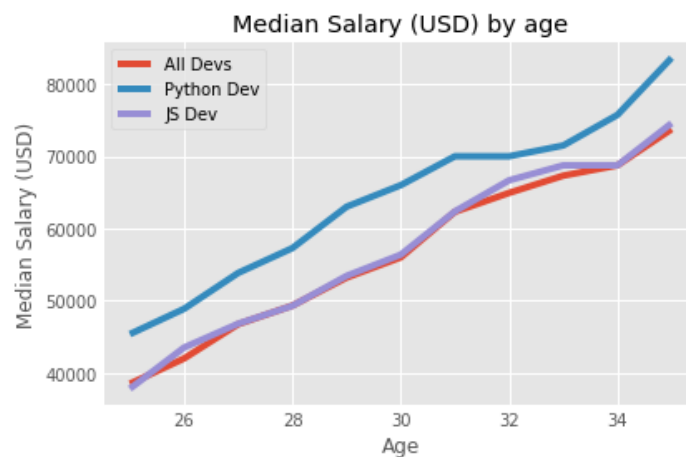
py_y = [45372, 48876, 53850, 57287, 63016, 65998, 70003, 70000, 71496, 75730, 83640] #range salary for python developer
plt.plot(age,py_y, label='Python Dev') #sb x menggunakan var yang sama tapi var y nya berbeda

js_y = [37810, 43515, 46823, 49293, 53437, 56373, 62375, 66674, 68745, 68746, 74583]
plt.plot(age,js_y,label='JS Dev')

plt.title('Median Salary (USD) by age') #memberi judul grafik
plt.xlabel('Age') #label utk sumbu x
plt.ylabel('Median Salary (USD)') #label utk sumbu y
plt.legend() #utk menampilkan label garis
plt.grid(True) #utk menambahkan grid di grafik
plt.tight_layout() #utk menambahkan padding
plt.savefig('plot.png') #save gambar
plt.show() #tampilkan plotnya
```

Gambar 10. Snip code menggunakan style bawaan matplotlib

Plt.savefig(“plot.png”) artinya gambar grafik akan disimpan dalam format png dan tersimpan dalam satu file dengan file program.



Gambar 11. Grafik menggunakan style “ggplot”