

MODUL VISUALISASI DATA

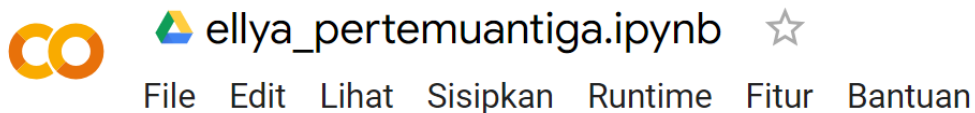
Package / Library untuk membuat grafik pada python adalah matplotlib, sedangkan modul yang digunakan untuk membuat plot/grafik adalah pyplot.

Untuk praktikum visualisasi data, kita menggunakan google colab agar lebih mudah untuk proses pemanggilan package.

Jika menghendaki menggunakan vscode, mahasiswa harus menginstal python terlebih dahulu, kemudian mengaktifkan virtual environment pada vscode dan baru menginstal matplotlib.

Langkan praktik :

1. Buka google colab dan ganti nama file dengan “visualisasidata.ipynb”



2. Kita bisa melakukan pemeriksaan apakah matplotlib sudah terinstal dalam g.colab dengan memberikan perintah:

```
try :  
    import matplotlib  
    print("Modul Matplotlib Terinstall")  
except ModuleNotFoundError:  
    print("Modul Matplotlib belum terinstall")
```

Untuk menjalankan kode program dengan menekan tombol:



Capture hasilnya

3. Untuk mengetahui versi matplotlib dengan perintah:

```
import matplotlib  
print(matplotlib.__version__)
```

Capture hasilnya

4. Membuat grafik sederhana dengan variable berupa integer biasa

```
import matplotlib.pyplot as plt # memanggil modul pyplot yang terdapat didalam matplotlib kemudian disingkat menjadi plt  
x = [2,4,6,8,10] #list nilai integer variabel x  
y = [3,5,1,2,9] # list nilai integer dari variabel y  
plt.plot(x,y) #memasukkan nilai dari list x dan y pada sumbunya  
plt.show() # menampilkan hasil grafik
```

Capture hasilnya

5. Selanjutnya adalah menampilkan list yang kita inputkan pada variable

```
x = [2,4,5]  
y = [5,1,9]  
print(x)  
print(y)
```

Capture hasilnya

6. Untuk membuat grafik titik dengan perintah:

```
plt.plot(x, y, 'o')
plt.show()
```

Capture hasilnya

7. Menggunakan data komputasional untuk menampilkan grafik dengan menggunakan package numpy atau sering diberi alias nama dengan np

```
import numpy as np #package dasar untuk komputasi ilmiah

xpoints = np.array([0, 3, 6])
ypoints = np.array([0, 100, 250])

print(xpoints)
print(ypoints)
```

Capture hasilnya

8. Kemudian menampilkan hasil input data komputasional dalam grafik dengan perintah:

```
plt.plot(xpoints, ypoints)
plt.show()
```

Capture hasilnya

9. Menampilkan grafik dengan hanya menggunakan salah satu sumbu, serta memberi perintah menampilkan lebih dari 1 gambar.

```
ypoints = np.array([1, 3, 6, 5]) #sumbu x nya bagaimana?
plt.plot(ypoints, linestyle = 'dotted')
plt.plot(ypoints, linestyle = 'dashed')
plt.plot(ypoints, linestyle = 'dashdot')# ada berapa tampilan gambar?
plt.show()
```

Capture hasilnya dan jawab pertanyaan yang terdapat dalam komentar pengkodean

10. Cara menampilkan banyak grafik dalam 1 gambar

```
ypoints = np.array([1, 3, 6, 5])
#plot 1
plt.subplot(1, 2, 1) #buat space grafik 1 baris 2 kolom kemudian tempatkan di nomor 1
plt.plot(ypoints, linestyle = 'dotted')
#plot 2
plt.subplot(1, 2, 2) #buat space grafik 1 baris 2 kolom kemudian tempatkan di nomor 2
plt.plot(ypoints)
```

Capture hasilnya

11. Buatlah 2 grafik dalam 1 tampilan dengan cara ditampilkan dalam kolom yang sama dan 2 baris yang berbeda
12. Buatlah 4 grafik dalam 1 tampilan dengan cara ditampilkan pada 2 baris dan 2 kolom.
13. Sedangkan membuat multiple data dalam 1 tampilan grafik

```
import numpy as np
X = np.linspace(0,30,100)
y1 = np.cos(X)
y2 = np.sin(X)
y3 = np.cos(X*2)
fig,ax = plt.subplots(figsize=(10,6)) # fig = membuat gambar baru, subplot = menggambar banyak gambar dalam 1 gambar
ax.plot(X, y1, color='red') #plot beberapa data
ax.plot(X, y2, color='blue')
ax.plot(X, y3, color='yellow')
plt.show()
```

Capture hasilnya

14. Membuat banyak grafik baris pada 1 plot grafik

```
import matplotlib.pyplot as plt
import numpy as np
x = np.array([1,3,5,7,9])
y1 = np.array([2,1,5,0,7])
y2 = np.array([1,2,3,4,5])
y3 = np.array([2,6,10,14,18])
fig,ax= plt.subplots(figsize=(10,20))
ax.plot(x,y1,color = 'red', marker = 'o')
ax.plot(x,y2,color = 'blue', marker = '*')
ax.plot(x,y3,color = 'green', marker = '+')
plt.show()
```

Capture hasilnya

15. Memberikan style pada latar grafik. Sebelum itu cek style dengan perintah

```
print(plt.style.available) # melihat style yang ada untuk membuat tampilan grafik
```

16. Perintah untuk memberikan style.

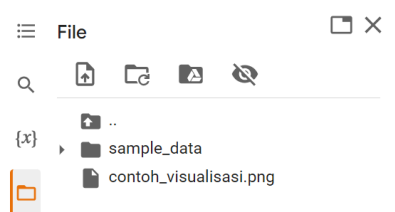
```
plt.style.use('tableau-colorblind10')
X = np.linspace(0, 30, 100) #membuat 1 set angka dengan spasi merata dalam interval yang telah ditentukan
y1 = np.cos(X*2)
y2 = np.cos(X)
y3 = np.sin(X/3)
y4 = np.sin(X)
fig, ax = plt.subplots(figsize=(10,6))
ax.plot(X, y1)
ax.plot(X, y2)
ax.plot(X, y3)
ax.plot(X, y4)
plt.show()
```

Capture hasilnya

17. Menyimpan gambar grafik

```
plt.style.use('ggplot')
X = np.linspace(0, 30, 100)
y1 = np.cos(X*2)
y2 = np.cos(X)
y3 = np.sin(X/3)
y4 = np.sin(X)
fig, ax = plt.subplots(figsize=(10,6))
ax.plot(X, y1)
ax.plot(X, y2)
ax.plot(X, y3)
ax.plot(X, y4)
plt.show()
fig.savefig('contoh_visualisasi.png')
```

Hasil gambar yang tersimpan, akan berada pada folder di frame sebelah kiri



18. Memanggil gambar grafik

```
from IPython.display import Image
Image('contoh_visualisasi.png')
```

Capture hasilnya

19. Memberi label pada sumbu x dan sumbu y

```
import matplotlib.pyplot as plt
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.figure(figsize=(12,6)) # merubah ukuran grafik
plt.plot(x, y)
plt.xlabel("Sumbu X") #memberikan nama pada label x
plt.ylabel("Sumbu Y") # memberikan nama pada label y
plt.show()
```

Capture hasilnya

20. Membuat grafik batang

```
x = np.array(["2005", "2010", "2015", "2020"])
y = np.array([3, 8, 1, 10])
plt.bar(x,y)
plt.show()
```

capture hasilnya

21. Membuat grafik scattered

```
x = np.array([5,7,8,7,2,17,2,9,4,11,12,9,6])
y = np.array([99,86,87,88,111,86,103,87,94,78,77,85,86])
plt.scatter(x, y)
plt.show()
```

capture hasilnya

22. Membuat grafik lingkaran

```
y = np.array([25, 30, 35, 10])

plt.pie(y)
plt.show()
```

Capture hasilnya

23. Membuat grafik lingkaran dengan labal

```
y = np.array([25, 30, 35, 10])
mylabels = ["Apples", "Bananas", "Cherries", "Dates"]

plt.pie(y, labels = mylabels)
plt.show()
```

Capture hasilnya

24. Membuat grafik lingkaran dengan sudut tidak 0 derajat

```
y = np.array([25, 30, 35, 10])
mylabels = ["Apples", "Bananas", "Cherries", "Dates"]

plt.pie(y, labels = mylabels, startangle = 45)
plt.show()
```

Capture hasilnya

25. Membuat grafik lingkaran 3d

```
y = np.array([25, 30, 35, 10])
nama = ["Apel", "Jeruk", "Mangga", "Melon"]
pisah = [0.1,0,0,0]
plt.pie(y, labels = mylabels, explode = pisah, shadow = True)
plt.legend(title = 'nama buah')
plt.show()
```

Capture hasilnya