

MODUL 5

PYTHON MODULES

5.1 Tujuan Praktikum

1. Praktikan diharapkan dapat memahami konsep Modules dalam Python.
2. Praktikan diharapkan dapat menjelaskan perbedaan antara Python Modules dan Python Matplotlib.
3. Praktikan diharapkan dapat melakukan instalasi dan penggunaan Python Modules secara praktis.

5.2 Alat dan Bahan

1. Laptop
2. Mouse
3. Perangkat Lunak Visual Studio Code
4. Python (3.12 ke atas)

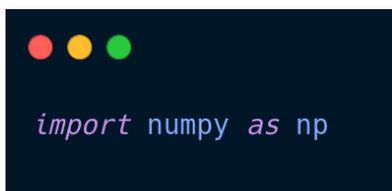
5.3 Dasar Teori

5.3.1 Python Modules

Modul dalam Python merupakan sebuah berkas dengan ekstensi **.py**, di mana nama modul tersebut sesuai dengan nama berkasnya. Modul Python dapat berisi sekumpulan *function*, *class*, atau *variable* yang telah didefinisikan dan diimplementasikan.

A. NumPy Tutorial

NumPy (Numerical Python) adalah sebuah *library* dalam Python yang dirancang khusus untuk komputasi ilmiah. Meskipun memiliki kesamaan dengan List di Python, NumPy menawarkan keunggulan dalam beberapa aspek, termasuk ukuran, kinerja, dan fungsionalitas. Sebelum memanfaatkan NumPy dalam program Python, *library* ini perlu di-*import* terlebih dahulu.

A terminal window with a dark blue background and three colored window control buttons (red, yellow, green) in the top left corner. The text `import numpy as np` is displayed in a light blue monospace font.

Jika Numpy belum terpasang, ketikkan perintah berikut di **Terminal** (untuk pengguna **Linux**), **CMD** atau **PowerShell** (untuk pengguna **Windows**):

```
pip install numpy
```

- Membuat Array NumPy

Membuat array menggunakan NumPy dengan terlebih dahulu membungkusnya dalam sebuah list, kemudian mengonversinya menjadi array NumPy.

```
import numpy as np  
arr= np.array([1,2,3,4,5])  
print(arr)  
print(type(arr))
```

Output akan menunjukkan:

```
[1 2 3 4 5]
```

```
<class 'numpy.ndarray'>
```

Terdapat beberapa fungsi yang dapat digunakan untuk membuat array, antara lain:

```
zeros()      -> Membuat array dengan nilai 0  
ones()       -> Membuat array dengan nilai 1  
arange()     -> Membuat array dengan nilai dalam range  
linspace()   -> Membuat array dengan nilai dalam interval  
logspace()   -> Membuat array dengan nilai log base10 dalam interval
```

B. Pandas Tutorial

Pandas adalah sebuah *library* Python *open source* yang dilisensikan di bawah **lisensi BSD**, yang dirancang untuk menyediakan struktur dan analisis data yang intuitif dan efisien. Pustaka ini sering digunakan untuk membangun tabel, mengubah dimensi data, serta melakukan pemeriksaan data, dan berbagai fungsi lainnya.

Struktur data utama dalam Pandas dikenal sebagai **DataFrame**, yang memungkinkan pengguna untuk dengan mudah membaca berkas dalam berbagai format, seperti **.txt**, **.csv**, dan **.tsv**. Dengan fitur ini, data dapat diolah menjadi format tabel dan dimanipulasi melalui berbagai operasi, termasuk **join**, **distinct**, **group by**, serta teknik-teknik lain yang umum digunakan dalam SQL

- Memasang Pandas:

```
pip install pandas
```

- Pengoperasian Pandas:

```
import pandas as pd

df =
pd.read_csv('data.csv')
print(df.to_string())
```

Hasilnya akan menampilkan *output* berikut yang menunjukkan data dari file **data.csv**:

```
(venv) ciper@system:~/Documents/telyu/telr
_48/modul5/modul5.py
      Nama  Usia  Negara Pekerjaan
0  Risnanda   19  Indonesia  Asprak
1   Audrey   19  Indonesia  Asprak
2  Rebecca   19  Indonesia  Asprak
```

C. SciPy Tutorial

SciPy (Scientific Python) adalah sebuah pustaka *open-source* yang dirancang untuk perhitungan ilmiah tingkat lanjut. Pustaka ini dibangun di atas ekstensi NumPy dan berfungsi secara sinergis untuk menangani komputasi yang kompleks. NumPy bertanggung jawab atas pengurutan dan pengindeksan data array, sementara SciPy menyediakan kode untuk pengolahan data numerik.

```
from scipy import constants
print(constants.liter)
```

Hasilnya akan menunjukkan: **0.001**

D. Django

Django merupakan kerangka kerja (*framework*) *full-stack* yang dirancang untuk pengembangan aplikasi web menggunakan bahasa pemrograman Python. *Framework* ini memfasilitasi proses pembuatan situs web, sehingga memungkinkan pengembang untuk bekerja dengan lebih efisien dibandingkan dengan menulis kode dari awal.

5.3.2 Python Matplotlib

Matplotlib adalah pustaka visualisasi grafis tingkat rendah yang dirancang untuk digunakan dengan bahasa pemrograman Python, yang dikembangkan oleh **John D. Hunter**. Sebagai perangkat lunak *open source*, Matplotlib dapat diakses secara gratis. Pustaka ini sebagian besar ditulis dalam bahasa Python, meskipun terdapat beberapa komponen yang ditulis dalam C, Objective-C, dan JavaScript untuk memastikan kompatibilitas lintas platform.

A. Matplotlib Pyplot

- *Import* modul **pyplot** dengan menggunakan perintah berikut:

```
import matplotlib.pyplot as plt
```

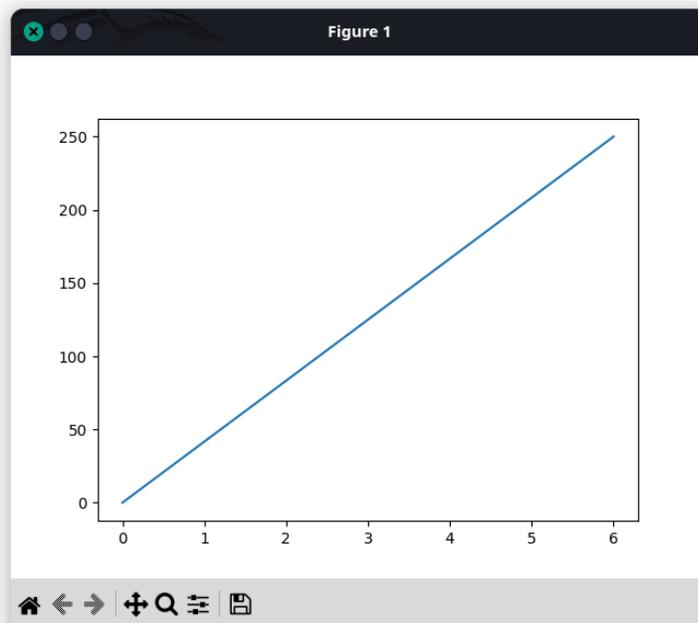
- Menggambar garis pada diagram dari titik (0,0) ke titik (6,250):

```
import matplotlib.pyplot as plt
import numpy as np

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

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

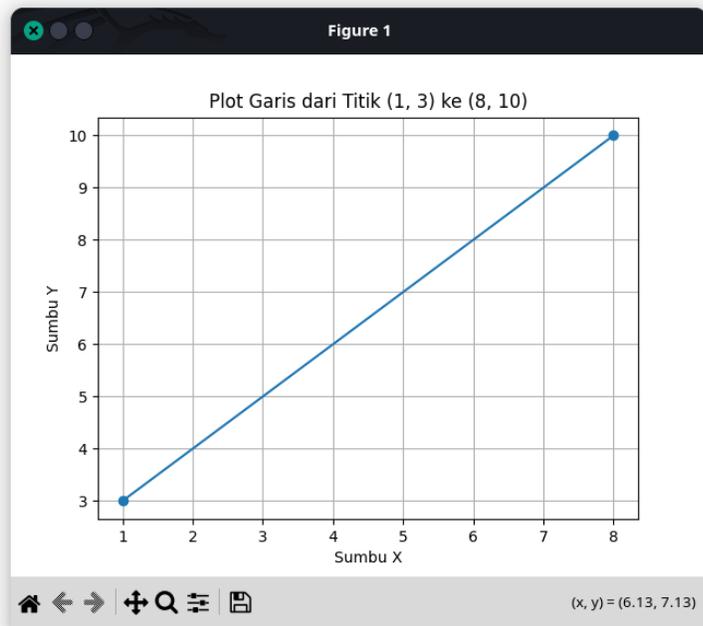
Keluaran:



B. Matplotlib Plotting

1. Plotting x dan y points

Fungsi **plot()** digunakan untuk menggambarkan titik-titik (penanda) dalam sebuah diagram. Secara *default*, fungsi ini akan menghubungkan titik-titik tersebut dengan garis. Fungsi ini menerima parameter yang digunakan untuk menentukan posisi titik dalam diagram. Parameter pertama adalah array yang berisi nilai-nilai pada sumbu x, sedangkan parameter kedua adalah array yang berisi nilai-nilai pada sumbu y. Sebagai contoh, untuk memplot garis dari titik (1, 3) ke titik (8, 10), kita perlu mengirimkan dua array, yaitu [1, 8] dan [3, 10], ke dalam fungsi **plot()**.



2. Plotting tanpa garis

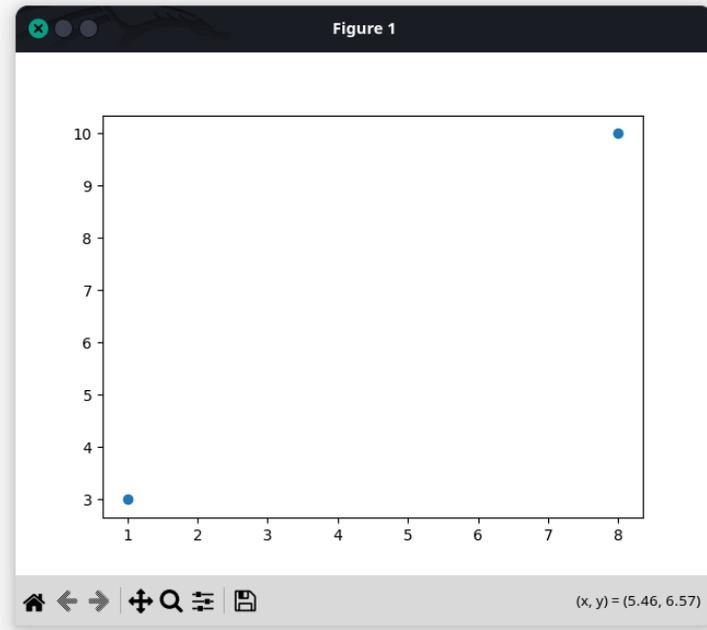
Untuk memplot penanda, kita dapat menggunakan parameter notasi string pintasan `'o'`. Sebagai contoh:

```
import matplotlib.pyplot as plt
import numpy as np

xpoints = np.array([1, 8])
ypoints = np.array([3, 10])

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

Keluaran:



3. Multiple Points

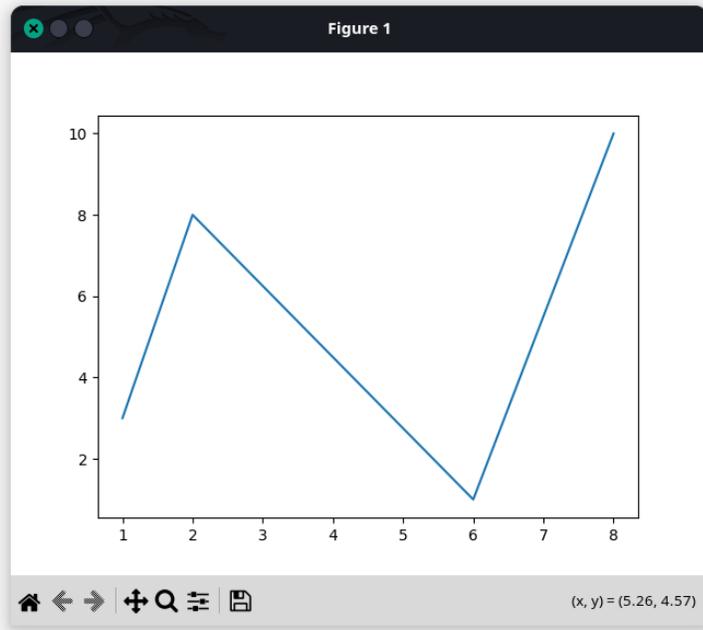
Kita dapat membuat plot titik sesuai dengan keinginan atau sesuai dengan permintaan dalam soal. Namun, perlu diperhatikan bahwa jumlah titik pada kedua sumbu haruslah sama. Sebagai contoh:

```
import matplotlib.pyplot as plt
import numpy as np

xpoints = np.array([1, 2, 6, 8])
ypoints = np.array([3, 8, 1, 10])

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

Keluaran:



4. Default X-Points

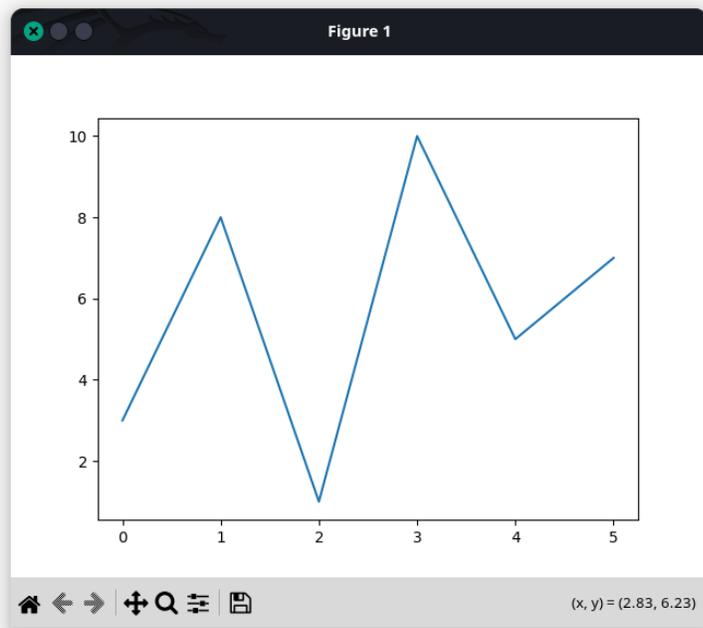
Apabila kita tidak menetapkan titik pada **sumbu x**, secara otomatis akan diberikan nilai default 0, 1, 2, 3, dan seterusnya, sesuai dengan jumlah titik pada **sumbu y**. Oleh karena itu, jika kita menggunakan contoh yang sama seperti sebelumnya dan tidak menyertakan titik pada **sumbu x**, diagram yang dihasilkan akan tampak seperti berikut:

```
import matplotlib.pyplot as plt
import numpy as np

ypoints = np.array([3, 8, 1, 10, 5, 7])

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

Keluaran:



#TelnetLaboratory

#TroubleshootingMakeUsStronger