

BASIC PYTHON

Arif Laksito, M. Kom www.ariflaksito.net





Numpy merupakan library python yang digunakan untuk bekerja dengan array Numpy hampir menyerupai List pada Python tetapi lebih powerful.

Ada beberap kelebihan NumPy dibandingkan List seperti size, performance dan functionally.

Sebelum menggunakan NumPy di Python, terlebih dahulu library ini harus diimport : import numpy as np

Jika belum ada NumPy dapat diinstall terlebih dahulu menggunakan pip : pip install numpy



MEMBUAT ARRAY

Kita bisa membuat Array menggunakan NumPy dengan membungkusnya terlebih dahulu di List dan dirubah menjadi numpy array atau langsung tanpa membuat variabel list.

```
import numpy as np

mylist = [8,9,10,11,12]
    arr1 = np.array([1, 2, 3, 4, 5])
    arr2 = np.array(mylist)

print(arr1)
    print(arr2)
```



INDEXING & SLICING

Array index dimulai dari 0 dan cara memanggilnya menggunakan notasi bracket "[]": print(arr[1])

Akses array 2D: print(arr[0, 1])

Slicing berarti mengambil elemen dari satu index ke index yang lain, dengan cara seperti berikut : [start:end]

Bisa juga dengan cara berikut : [start:end:step]



FUNGSI STATISTIK

NumPy mempunyai fungsi matematika yang dapat digunakan pada Array Beberapa contoh fungsi statistik antara lain min(), max(), mean(), sum(), std()

```
arrku = np.array([1,2,3,4,5])

print("Nilai minimal = ", arrku.min())
print("Nilai maksimal = ", arrku.max())
print("Nilai rata-rata = ", arrku.mean())
print("Total nilai = ", arrku.sum())
print("Standar Deviasi = ", arrku.std())
```





Pandas merupakan library Python yang dibangun diatas NumPy dan cocok digunakan untuk melakukan preprocessing data dan analisis

Panda mampu mengolah berbagai jenis data seperti CSV, JSON, Excel, TXT, HTML, XML, PDF, Pickle dan lain lain

Untuk menggunakan Pandas kita harus import terlebih dahulu : import pandas as pd

jika belum ada library pandas install terlebih dahulu dengan pip: pip install pandas





Series adalah struktur data dasar Pandas yang berbentuk array berlabel satu dimensi. Series mampu menampung berbagai jenis data seperti integer, string, float, dll

Kita dapat membuat Series dengan input dari List dan Array

```
data_series = pd.Series([2,4,6,8,10])
data_series = pd.Series(np.array([2,4,6,8,10]))
```

Kita juga dapat memodifikasi bentuk indeksnya seperti ini

```
data = np.array([2,4,6,8,10])
nilai_series = pd.Series(data, index=['a','b','c','d','e'])
```



DATAFRAME

Dataframe adalah struktur data 2 dimensi yang berbentuk tabular (mempunyai baris dan kolom)

Dataframe memiliki 2 index, yaitu index baris dan index columns.

Dalam satu column dataframe harus memiliki tipe data yang sama, tapi antar columnnya dataframe bisa memiliki jenis data yang berbeda.

Dataframe dapat dibuat lebih dari satu Series atau dapat kita katakan bahwa dataframe adalah kumpulan Series



DATAFRAME

DataFrame dapat di bentuk dari dictionary dengan perintah berikut:

```
df_staff = pd.DataFrame(dict_staff)
Print(df_stff.head())
```

```
# Filtering DataFrame
df_staff[df_staff['age']==31]
```

```
# Slicing DataFrame
df_staff[{'div','jk','rating'}][10:16]
```





Tutorial Numpy, https://ngodingdata.com/tutorial-dasar-numpy-python/

W3Schools - Python, https://www.w3schools.com/python/

Python Pandas: Pengenalan Series, https://ngodingdata.com/python-pandas-pengenalan-series/

Python Pandas: Pandas DataFrame, https://ngodingdata.com/python-pandas-dataframe/



TERIMAKASIH

about us:

www.tlab.co.id