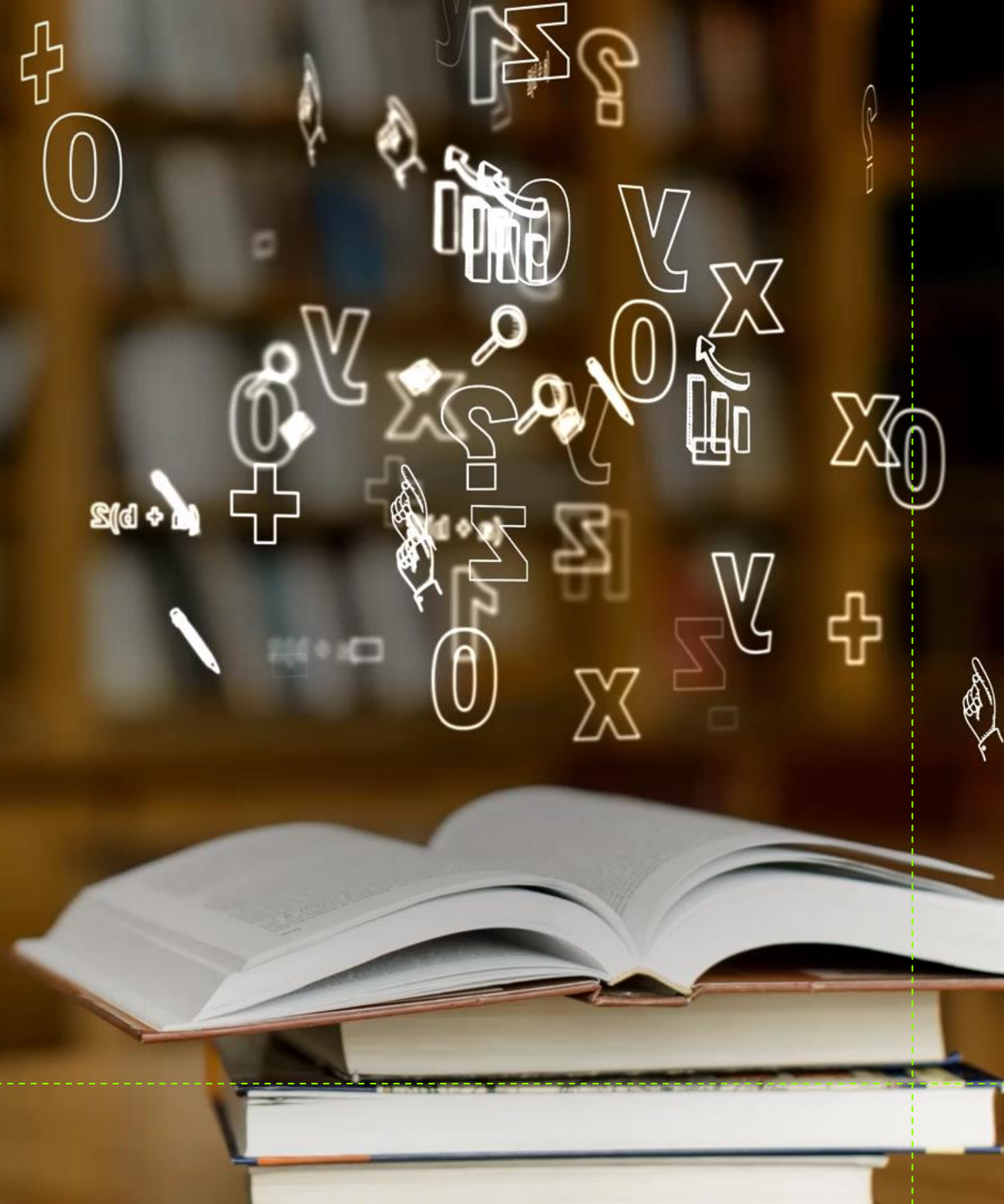


# Library Numpy

Ledy Elsera Astrianty,  
S.Kom., M.Kom



# Apa itu NumPy?

- NumPy adalah paket dasar yang diperlukan untuk komputasi ilmiah dengan Python. Itu mengandung:
  - objek array N-dimensi yang kuat
  - fungsi aljabar linear dasar
  - transformasi Fourier dasar
  - kemampuan nomor acak yang canggih
  - alat untuk mengintegrasikan kode Fortran
  - alat untuk mengintegrasikan kode C/C++

# dokumentasi NumPy

- Dokumentasi resmi
  - <http://docs.scipy.org/doc/>
- Buku NumPy
  - <http://www.tramy.us/numpybook.pdf>
- Daftar contoh
  - [http://www.scipy.org/Numpy\\_Example\\_List\\_With\\_Doc](http://www.scipy.org/Numpy_Example_List_With_Doc)

# Struktur data ndarray

- NumPy menambahkan struktur data baru ke Python – ndarray
  - Array N-dimensi adalah kumpulan "item" homogen yang diindeks menggunakan bilangan bulat N
  - Didefinisikan oleh:
    1. bentuk array, dan
    2. jenis item yang terdiri dari array

# Bentuk susunan

- ndarray berbentuk persegi panjang
- Bentuk array adalah tuple dari N bilangan bulat (satu untuk setiap dimensi)

# Jenis item array

- Setiap ndarray adalah kumpulan homogen dari tipe data yang persis sama
  - setiap item menggunakan blok memori dengan ukuran yang sama
  - setiap blok memori dalam array diinterpretasikan dengan cara yang persis sama

Type	Bit-Width	Character
<b>bool_</b>	boolXX	' ? '
byte	intXX	' b '
short		' h '
intc		' i '
<b>int_</b>		' l '
longlong		' q '
intp		' p '
ubyte	uintXX	' B '
ushort		' H '
uintc		' I '
uint		' L '
ulonglong		' Q '
uintp		' P '
single	floatXX	' f '
<b>float_</b>		' d '
longfloat		' g '
csingle	complexXX	' F '
<b>complex_</b>		' D '
clongfloat		' G '
<b>object_</b>		' O '
<b>str_</b>		' S# '
<b>unicode_</b>		' U# '
void		' V# '

Type	Bit-Width	Character
<b>bool_</b>	boolXX	'?'
byte	intXX	'b'
short		'h'
intc		'i'
<b>int_</b>		'l'
longlong		'q'
intp		'p'
ubyte	uintXX	'B'
ushort		'H'
uintc		'I'
uint		'L'
ulonglong		'Q'
uintp		'P'
single	floatXX	'f'
<b>float_</b>		'd'
longfloat		'g'
csingle	complexXX	'F'
<b>complex_</b>		'D'
clongfloat		'G'
<b>object_</b>		'O'
str_		'S#'
unicode_		'U#'
void		'V#'



# Contoh: membuat array

```
pip install numpy
```

```
import numpy
```

```
a = array([[1, 2, 3],  
          [4, 5, 6],  
          [7, 8, 9]])
```

```
shape
```

```
a.dtype
```

# Pengindeksan array

- Gunakan tuple untuk mengindeks array multi-dimensi
- Contoh:  
`a[1, 2]`

# Mengiris array

- Slicing arrays hampir sama dengan slicing list, kecuali Anda dapat menentukan beberapa dimensi

# Contoh: Mengiris array

`a[1]`

`a[1, :]`

`a[1, 1:]`

`a[:1, 1:]`

# Beberapa metode ndarray

- `ndarray.tolist()`
  - Isi diri sebagai daftar bersarang
- `ndarray.copy()`
  - Kembalikan salinan array
- `ndarray.fill(scalar)`
  - Isi array dengan nilai skalar

# Operasi Dasar pada Array NumPy

---

Penjumlahan:  $\text{arr1} + \text{arr2}$

---

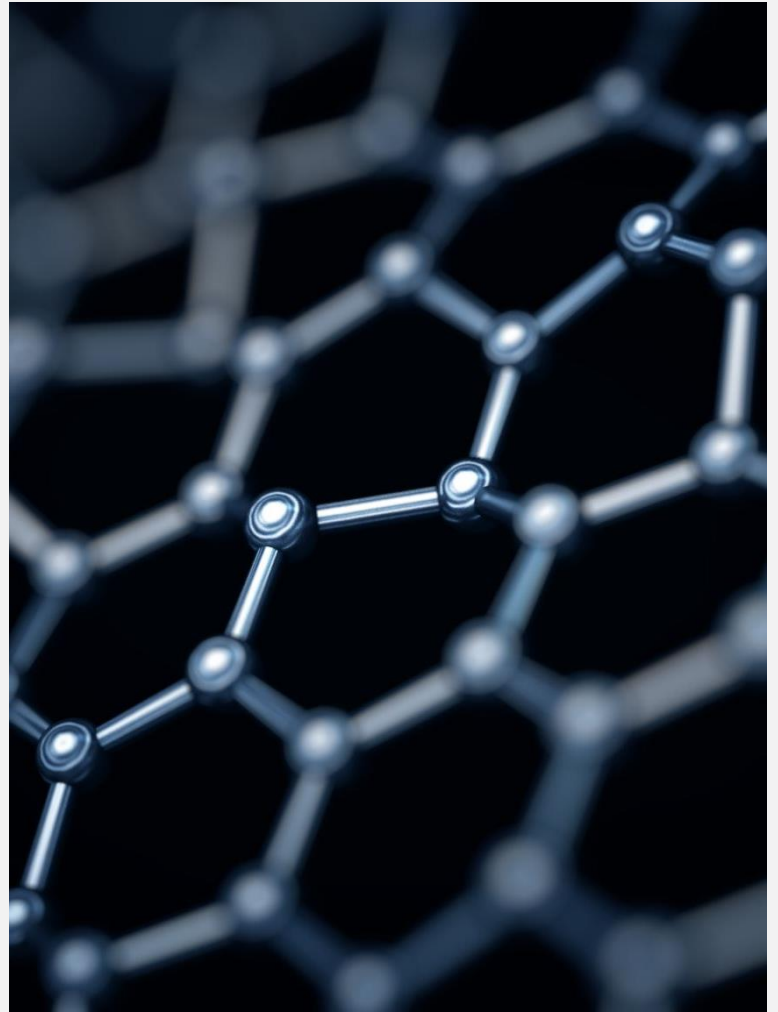
Pengurangan:  $\text{arr1} - \text{arr2}$

---

Perkalian:  $\text{arr1} * \text{arr2}$

---

Pembagian:  $\text{arr1} / \text{arr2}$





# Fungsi Statistik pada NumPy



`numpy.mean():`  
menghitung rata-rata



`numpy.median():`  
menghitung median



`numpy.std():` menghitung  
standar deviasi



`numpy.var():` menghitung  
varians

# Fungsi Aljabar Linear

`numpy.dot():`

- menghitung hasil kali matriks

`numpy.transpose():`

- menghitung transpose matriks

`numpy.linalg.inv():`

- menghitung invers matriks

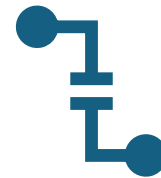
`numpy.linalg.det():`

- menghitung determinan matriks



# Fungsi Dasar

- `np.array()`: Membuat array NumPy dari data yang diberikan.
- `np.zeros()`: Membuat array NumPy dengan nilai 0.
- `np.ones()`: Membuat array NumPy dengan nilai 1
- `np.empty()`: Membuat array NumPy dengan nilai acak.
- `np.full()`: Membuat array NumPy dengan nilai yang sama.
- `np.eye()`: Membuat array NumPy dengan nilai 1 di diagonal.
- `np.identity()`: Membuat array NumPy dengan nilai 1 di diagonal.





---

## Operasi Aritmatika

- `np.add()`: Menambahkan dua array NumPy.
- `np.subtract()`: Mengurangi dua array NumPy.
- `np.multiply()`: Mengalikan dua array NumPy.
- `np.divide()`: Membagi dua array NumPy.
- `np.mod()`: Menghitung sisa dari pembagian dua array NumPy.
- `np.power()`: Menghitung pangkat dari dua array NumPy.
- `np.sqrt()`: Menghitung akar kuadrat dari array NumPy.

# Operasi Statistik

- `np.mean()`: Menghitung rata-rata dari array NumPy.
- `np.median()`: Menghitung median dari array NumPy.
- `np.std()`: Menghitung standar deviasi dari array NumPy.
- `np.var()`: Menghitung variansi dari array NumPy.
- `np.min()`: Menghitung nilai minimum dari array NumPy.
- `np.max()`: Menghitung nilai maksimum dari array NumPy.
- `np.ptp()`: Menghitung rentang nilai dari array NumPy.

# Operasi Array

- `np.reshape()`: Mengubah bentuk array NumPy.
- `np.transpose()`: Mengubah orientasi array NumPy.
- `np.concatenate()`: Menggabungkan dua atau lebih array NumPy.
- `np.split()`: Memisahkan array NumPy menjadi beberapa bagian.
- `np.array_split()`: Memisahkan array NumPy menjadi beberapa bagian dengan ukuran yang sama.
- `np.hsplit()`: Memisahkan array NumPy menjadi beberapa bagian secara horizontal.
- `np.vsplit()`: Memisahkan array NumPy menjadi beberapa bagian secara vertikal.



# Fungsi Matematika

---

`np.sin()`: Menghitung sinus dari nilai dalam array NumPy.

---

`np.cos()`: Menghitung kosinus dari nilai dalam array NumPy.

---

`np.tan()`: Menghitung tangen dari nilai dalam array NumPy.

---

`np.exp()`: Menghitung eksponen dari nilai dalam array NumPy.

---

`np.log()`: Menghitung logaritma dari nilai dalam array NumPy.

---

`np.sqrt()`: Menghitung akar kuadrat dari nilai dalam array NumPy.

# Fungsi Random

---



`np.random.rand()`: Mengembalikan array NumPy dengan nilai acak antara 0 dan 1.



`np.random.randn()`: Mengembalikan array NumPy dengan nilai acak dari distribusi normal.



`np.random.randint()`: Mengembalikan array NumPy dengan nilai acak dari rentang integer.



`np.random.choice()`: Mengembalikan array NumPy dengan nilai acak dari daftar nilai.

# Fungsi Lainnya

- 
- `np.where()`: Mengembalikan nilai berdasarkan kondisi yang diberikan.
  - `np.all()`: Mengembalikan True jika semua nilai dalam array NumPy adalah True.
  - `np.any()`: Mengembalikan True jika ada nilai dalam array NumPy yang adalah True.
  - `np.sort()`: Mengurutkan nilai dalam array NumPy.
  - `np.argsort()`: Mengembalikan indeks nilai yang diurutkan dalam array NumPy.
  - `np.searchsorted()`: Mengembalikan indeks nilai yang dicari dalam array NumPy yang diurutkan.
  - `np.unique()`: Mengembalikan nilai unik dalam array NumPy.
  - `np.copy()`: Mengembalikan

# Thank You

- ♦ Ledy Elsera Astrianty
- ♦ [ledyselera@gmail.com](mailto:ledyselera@gmail.com)

