

Python 데이터 분석과 이미지 처리

나동빈

Numpy의 기본 사용법

Numpy란?

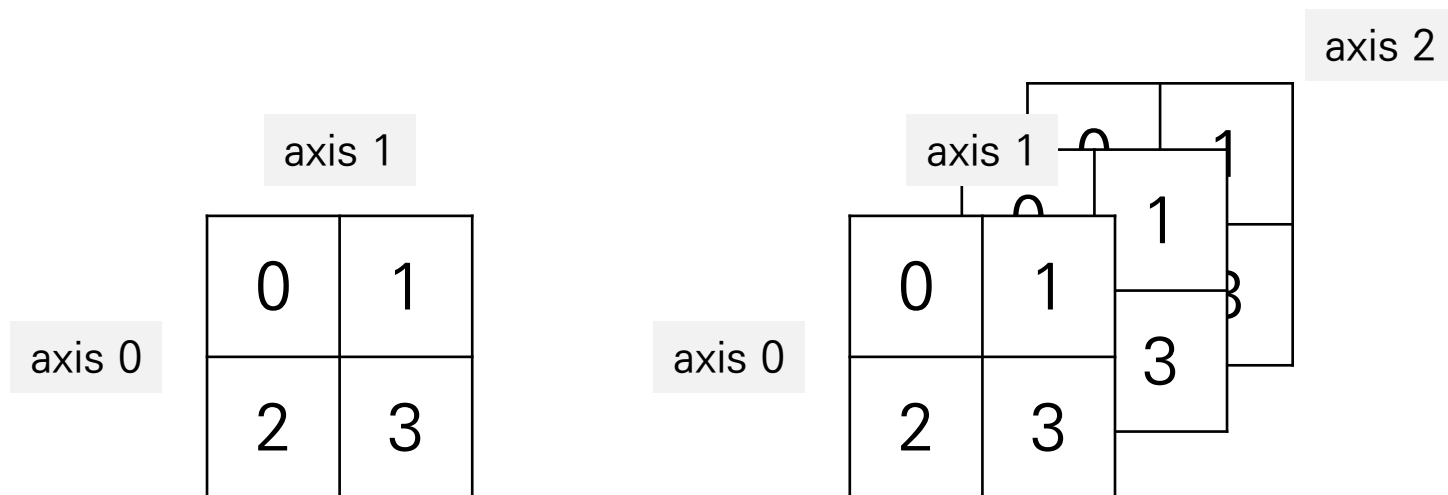
- Numpy는 다차원 배열을 효과적으로 처리할 수 있도록 도와주는 도구입니다.
- 현실 세계의 다양한 데이터는 배열 형태로 표현할 수 있습니다.
- Python의 기본 List에 비해 빠르고 강력한 기능을 제공합니다.

	홍길동	이순신	나동빈
영어	98	100	97
수학	87	97	95

Numpy의 기본 사용법

Numpy의 차원

- 1차원 축(행): axis 0 => Vector
- 2차원 축(열): axis 1 => Matrix
- 3차원 축(채널): axis 2 => Tensor(3차원 이상)



Numpy의 기본 사용법

List를 Numpy로 바꾸어 보기

```
import numpy as np

array = np.array([1, 2, 3])

print(array.size) # 배열의 크기
print(array.dtype) # 배열 원소의 타입
print(array[2]) # 인덱스 2의 원소
```

Numpy의 기본 사용법

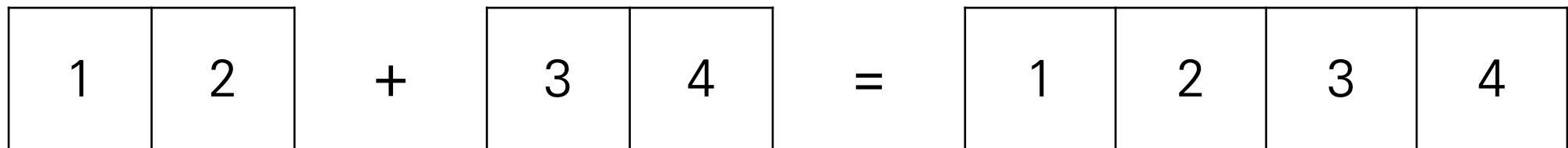
Numpy 배열 초기화하기

```
import numpy as np

# 0부터 3까지의 배열 만들기
array1 = np.arange(4)
array2 = np.zeros((4, 4), dtype=float) # 0으로 초기화
array3 = np.ones((3, 3), dtype=str) # 1로 초기화
# 0부터 9까지 랜덤하게 초기화 된 배열 만들기
array4 = np.random.randint(0, 10, (3, 3))
# 평균이 0이고 표준편차가 1인 표준 정규를 띄는 배열
array5 = np.random.normal(0, 1, (3, 3))
print(array1)
print(array2)
print(array3)
print(array4)
print(array5)
```

Numpy의 기본 사용법

Numpy 배열 합치기



Numpy의 기본 사용법

Numpy 배열 합치기

```
import numpy as np

array1 = np.array([1, 2, 3])
array2 = np.array([4, 5, 6])
array3 = np.concatenate([array1, array2])

print(array3.shape)
print(array3)
```

Numpy의 기본 사용법

Numpy 배열 형태 바꾸기

```
import numpy as np

array1 = np.array([1, 2, 3, 4])
array2 = array1.reshape((2, 2))

print(array2.shape)
```


Numpy의 기본 사용법

Numpy 배열 세로 축으로 합치기

1	2
---	---

+

3	4
---	---

=

1	2
3	4

Numpy의 기본 사용법

Numpy 배열 세로 축으로 합치기

```
import numpy as np

array1 = np.arange(4).reshape(1, 4)
array2 = np.arange(8).reshape(2, 4)

array3 = np.concatenate([array1, array2], axis=0)
print(array3.shape)
```

Numpy의 기본 사용법

Numpy 배열 나누기

```
import numpy as np

array = np.arange(8).reshape(2, 4)
left, right = np.split(array, [2], axis=1)
print(left.shape)
print(right.shape)
print(right[1][1])
```