

## ▼ Hello NumPy I

더블클릭

드라이브에서 찾기

더블클릭

새 Python 3 노트

새 Python 2 노트

### ▼ OBJECT

• T

노트 열기...

Ctrl+O

• 본

노트 업로드...

자

이름 바꾸기...

휴지통으로 이동

import r

드라이브에 사본 저장...

a = np.a

GitHub Gist로 사본 저장...

print(ty

GitHub에 사본 저장...

print(a,

print(a,

a[0] = 5

저장

Ctrl+S

print(a)

버전 저장 및 고정

Ctrl+M S

☐→

&lt;t

업데이트 기록

(3

(1

[5

.ipynb 다운로드

.py 다운로드

b = np.a

print(b)

드라이브 미리보기 업데이트

in a rank of 2 (rank가 2인 배열 생성)

☐→

인쇄

Ctrl+P

print(b.shape)

# "(2, 3)"

☐→

print(b[0, 0])

☐→

print(b[1, 1])

☐→

print(b[1, 2])

☐→

## ▼ Axis / axes

- the nth coordinate to index an array in Numpy.
- multidimensional arrays can have one index per axis.

```
import numpy as np

a = np.array([[1, 2], [3, 4]])
print a
```



- If not specified, the overall mean will be obtained (지정하지 않으면 전체 평균을 구하게 됨)

```
print(np.mean(a))
```



Axis 0 (↓)

```
print(np.mean(a, axis=0)) # [ 2.  3.]
```



Axis 1 (→)

```
print(np.mean(a, axis=1)) # [ 1.5  3.5]
```



## ▼ Broadcast

- Calculate arrays with different shapes 형상이 다른 배열을 계산하기 위해서 지원하는 기능

```
A = np.array([[1, 2], [3, 4]])
B = np.array([10, 20])
print(A)
print('-----')
print(B)
```



Please observe how it is multiplied. (어떻게 곱해지는지 잘 관찰바랍니다.)

```
print(A*B)
```



## ▼ Stack

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

- stack vertically (세로로 쌓기)

```
c = np.vstack((a,b))  
print(c)
```



```
print(c.shape)
```



- stack horizontally (가로로 쌓기)

```
d = np.hstack((a,b))  
print(d)
```



```
print(d.shape)
```



