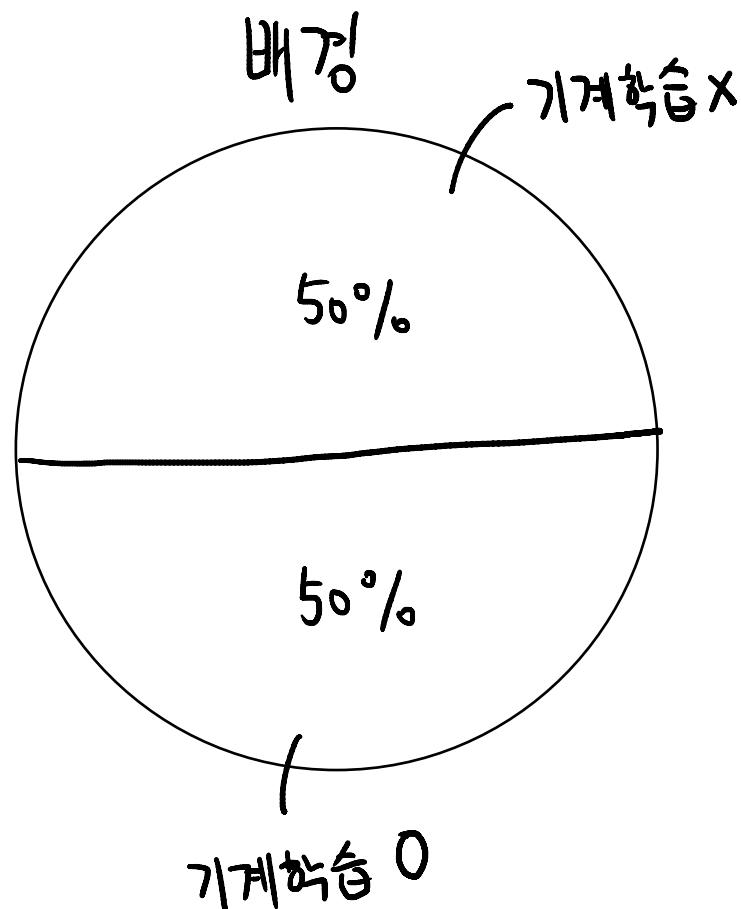
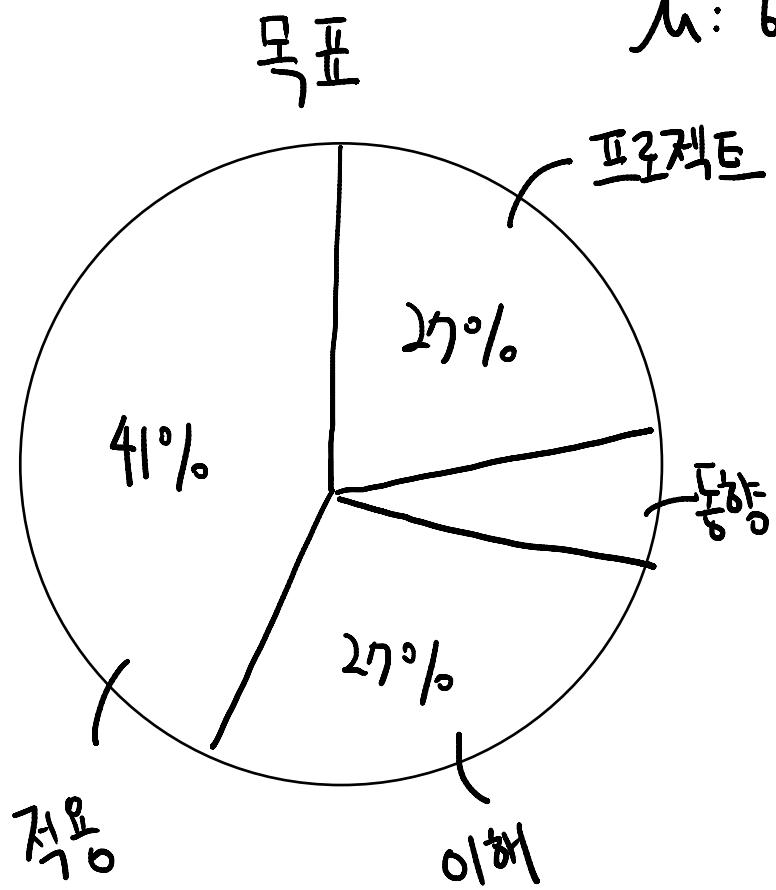


파이썬 딥러닝

이성주

seongjoo@codebasic.io

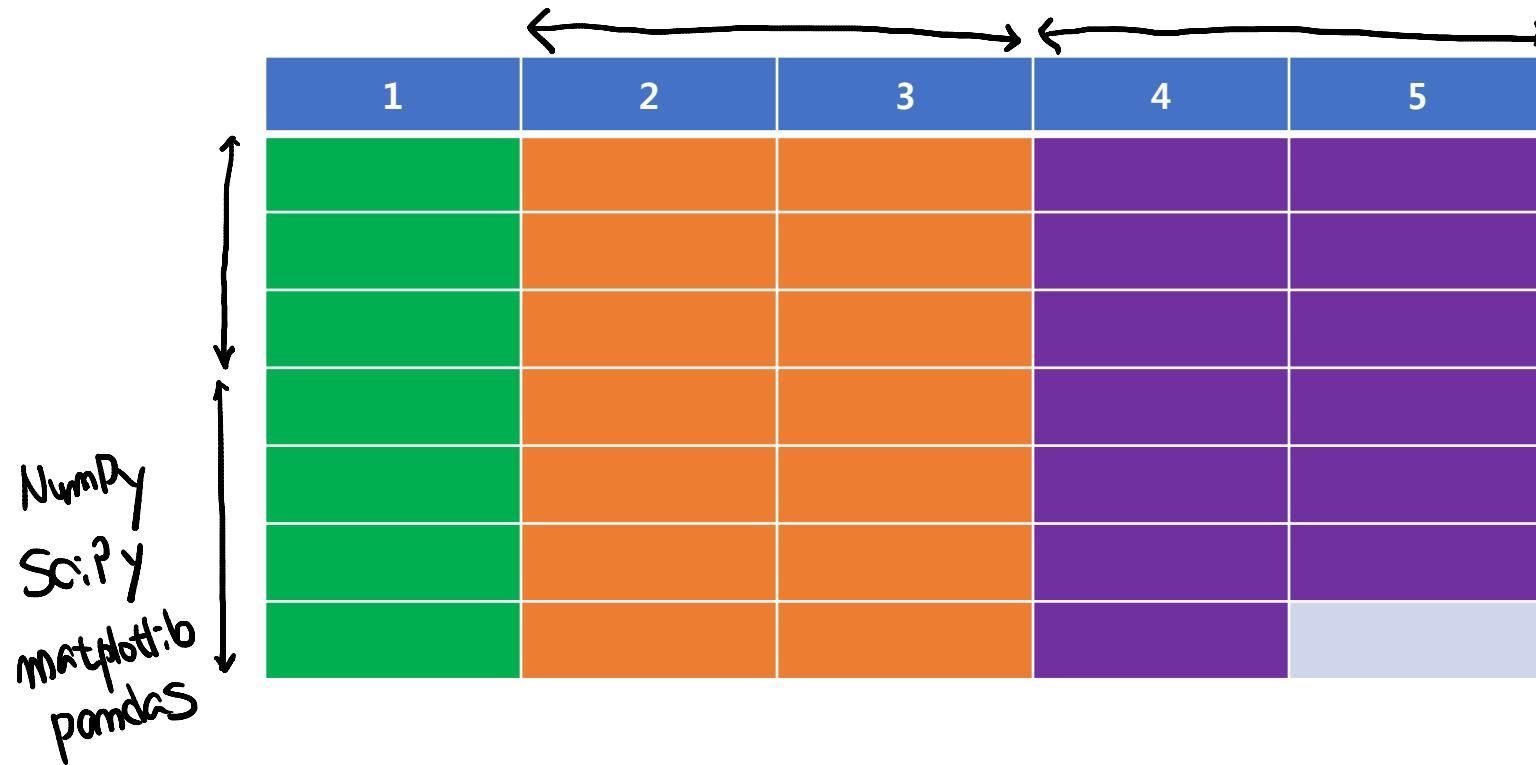
파이션 문법
 $\mu: 63$



기계학습

Deep Learning

Scikit-Learn





localhost: "내 컴퓨터"

Logout

HOME



Name ↑

Last Modified ↑



□ [Contacts](#) 4 months ago

□ [Desktop](#) 2 hours ago

□ [Documents](#) an hour ago

□ [Downloads](#) 4 months ago

□ [Favorites](#) 4 months ago

□ [Links](#) 39 minutes ago

□ [Music](#) 4 months ago

□ [Pictures](#) 4 months ago

□ [Saved Games](#) 4 months ago

□ [\[empty folder\]](#) 1 hrs ago





jupyter Day 1



Logout

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



CELL ↴

In [1]:

```
인사 = '파이썬 딥러닝 알아봅시다!'
print(인사)
print('셀 실행은 shift+Enter')
```

실행결과 ↴

파이썬 딥러닝 알아봅시다!
셀 실행은 shift+Enter

In []:

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help

In [2]: `print(x)`

```
-----
-----
1) NameError Trace
2) back (most recent call last)
   <ipython-input-2-81745ac23551> in <module>()
-----> 1) print(x)

3) NameError: name 'x' is not defined
```

(Note: The number 1 is circled in red.)

In []:

Notebook saved

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



In [1]: 인사 = '파이썬 딥러닝 알아봅시다!'
print(인사)
print('셀 실행은 shift+Enter')

[1] → [3] → [4]

인사 X

파이썬 딥러닝 알아봅시다!
셀 실행은 shift+Enter

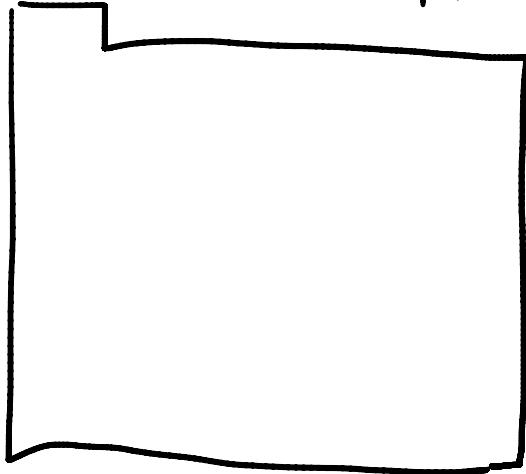
In [4]: print(x)

1

In [3]: x = 1

In []:

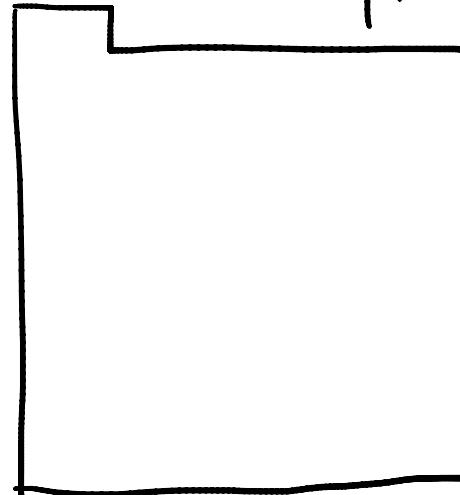
Notebook1.ipynb



KERNEL #1

$x=1$

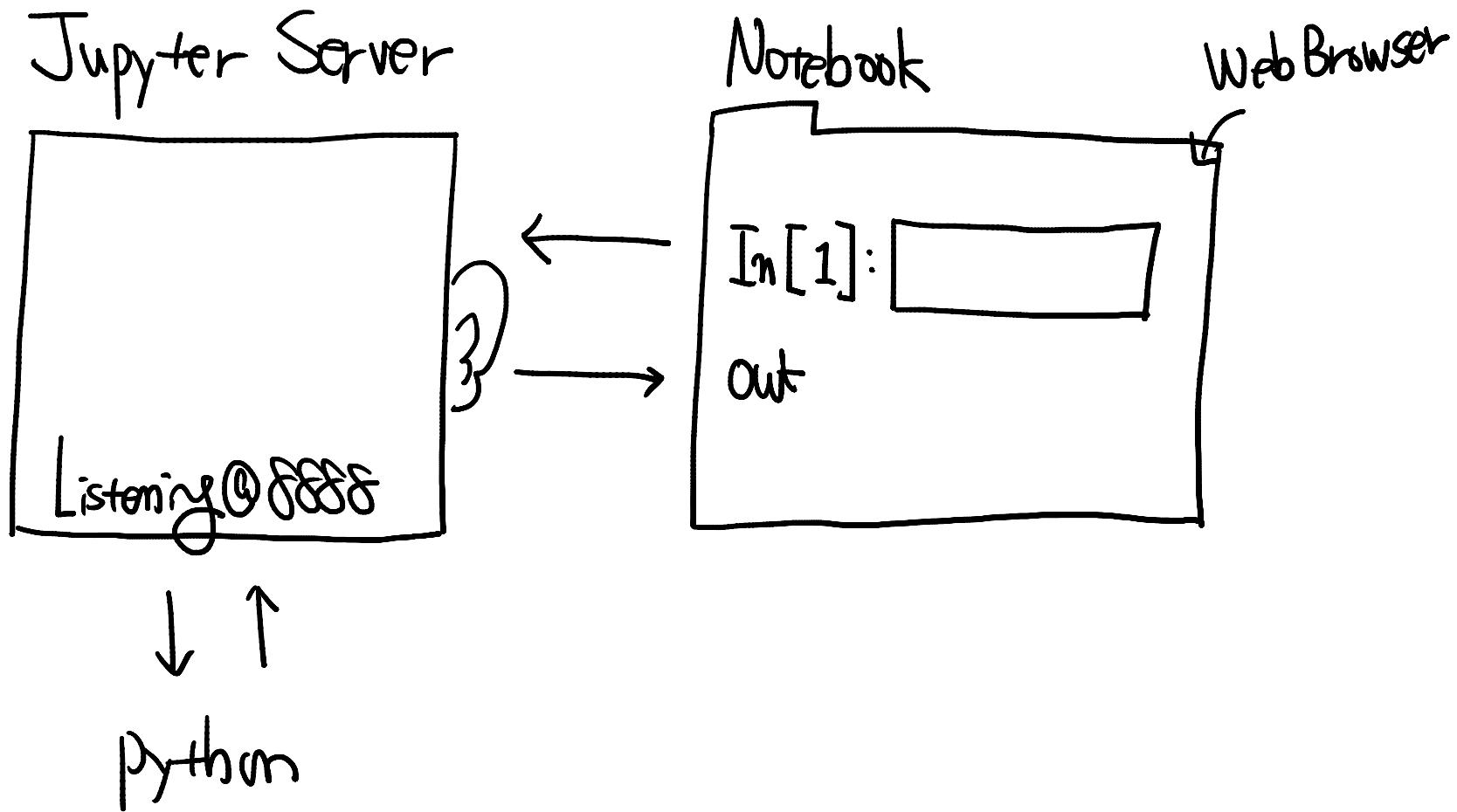
Notebook2.ipynb



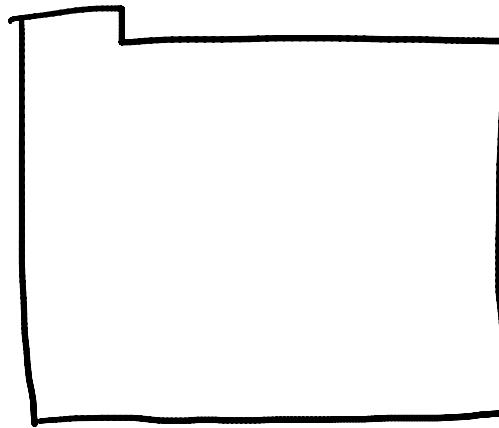
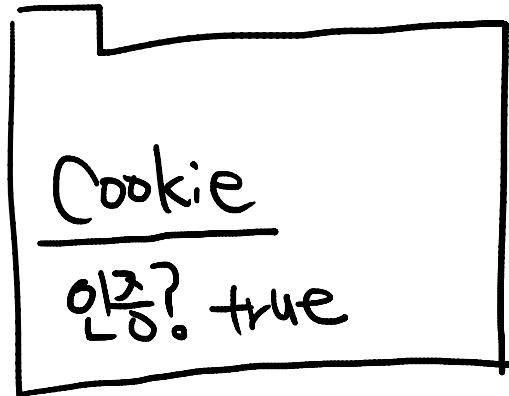
KERNEL #2

$x=2$

IPython Notebook → Jupyter Notebook



http://localhost:8888?token=_____



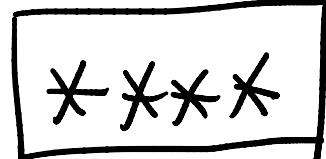
Anaconda Prompt

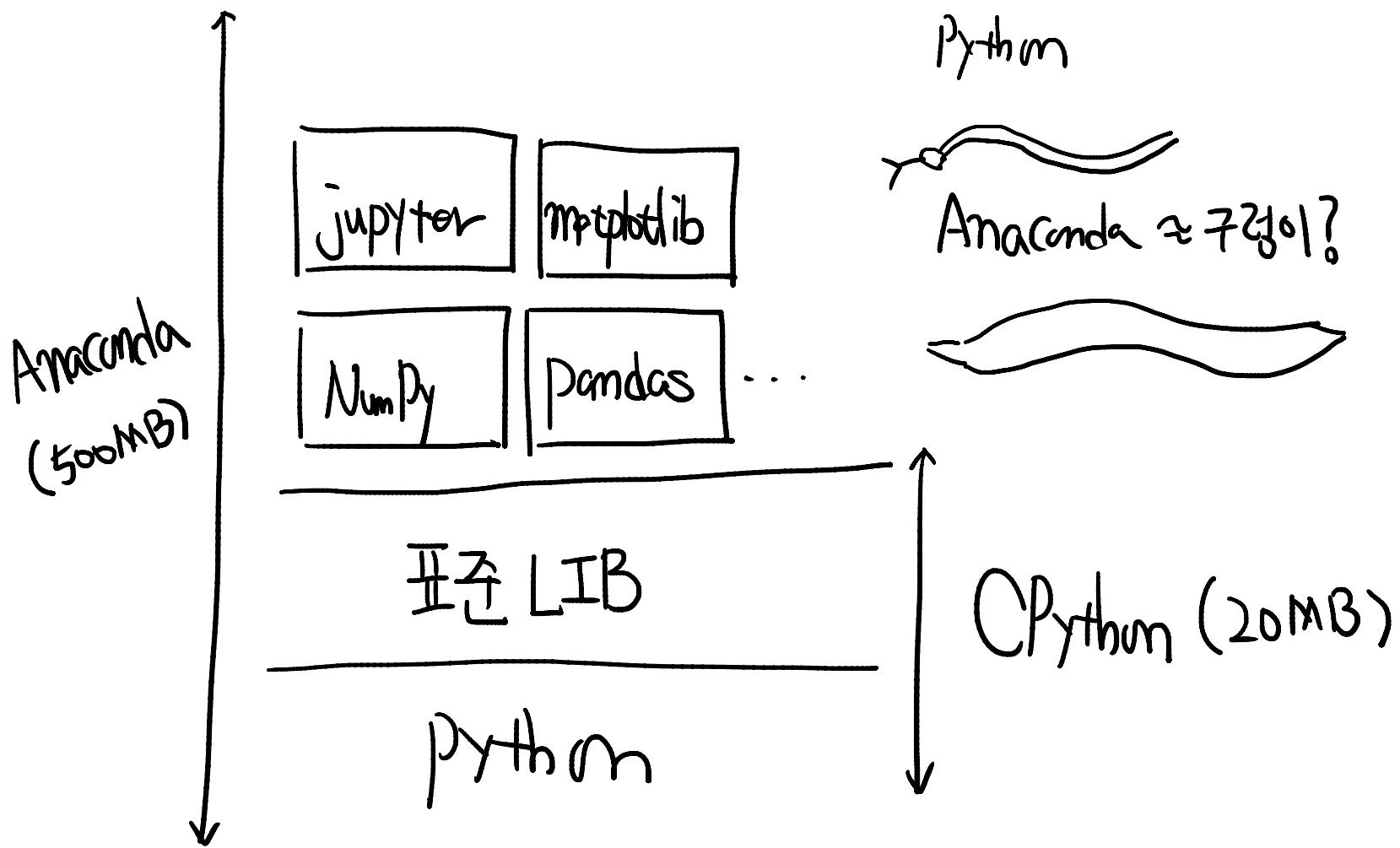
\$ jupyter notebook password

Enter password:

* Restart Server *

localhost:8888

password : 



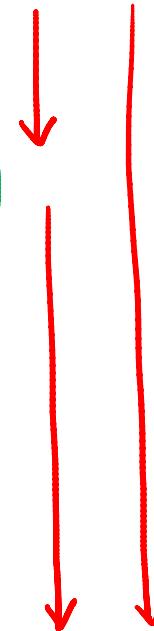
In[1] :

In[2] :

In[3] :

In[4] :

Run Above



Run ALL

Run Below

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



셀 실행은 shift+Enter

In []:

↑ [A]bove

In [2]: x = 1

↓ [B]elow

In []:

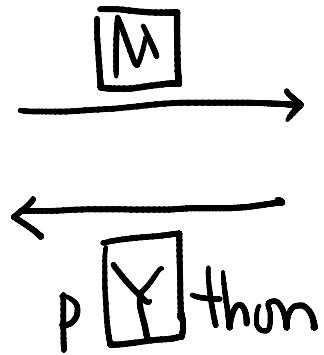
In [3]: print(x)

1

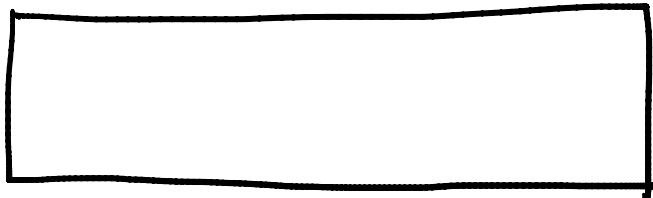
In []:

CODE

In[] :



MarkDown



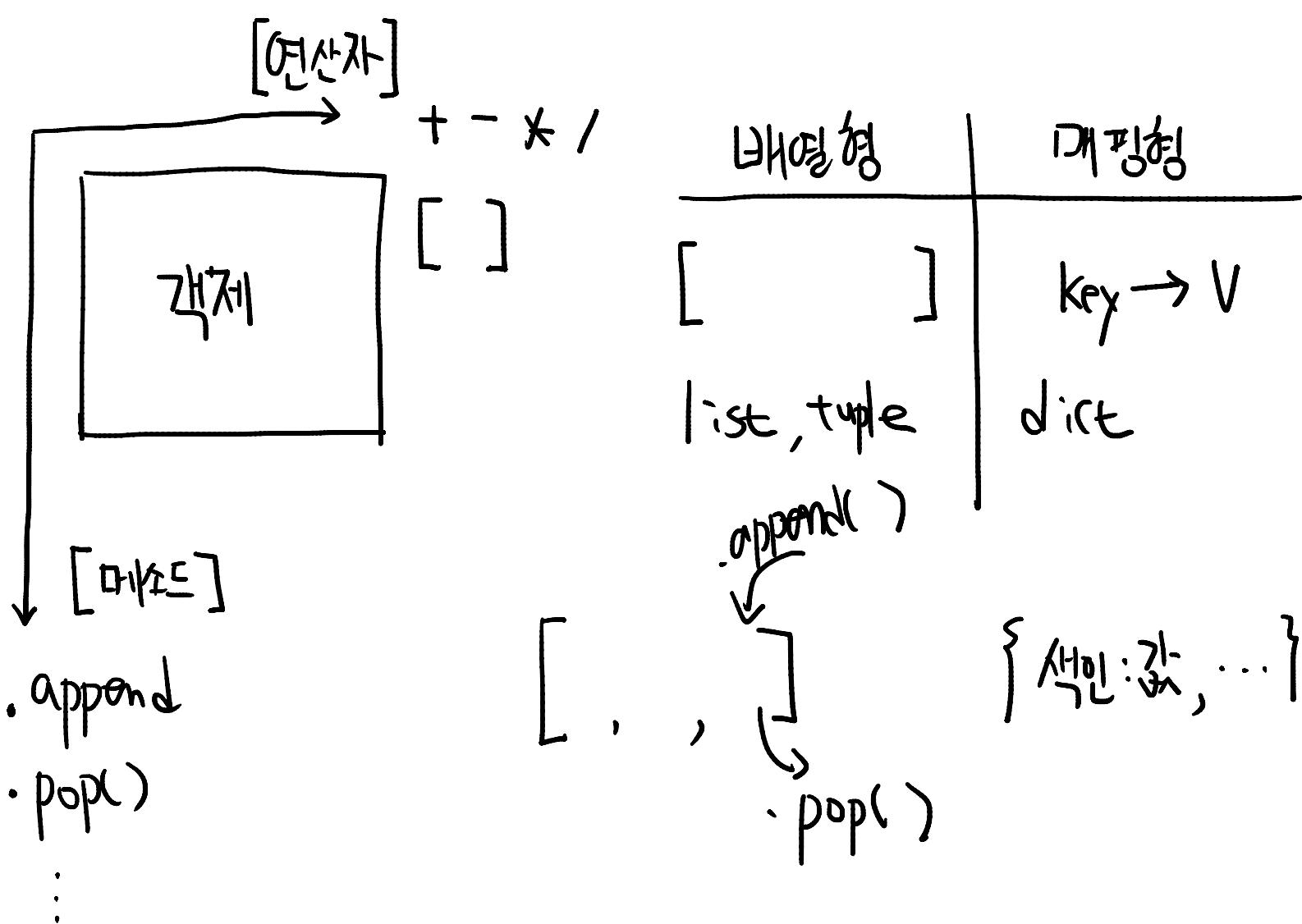
{ 색인 : 형식 }

선택적

0 1
↓ ↓

'{0}' '{1}'.format('A', 'B')

{ 0: .2f }
0.00 float



Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



In [32]: 좌표[:2]

Out[32]: (1.2, 2.3)

In [33]: 좌표[-2:]

Out[33]: (2.3, 3.4)

IPython Magic

"매직"
~ 설정

In [34]: %pwd

Out[34]: 'C:\Users\student\Documents\pydeep'

In []:

Notebook saved

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



Out[34]: 'C:\Users\student\Documents\pydeep'

```
In [36]: data = []
for n in range(10):
    data.append(n**2)
data
```

Out[36]: [0, 1, 4, 9, 16, 25, 36, 49, 64, 81]

$$\begin{bmatrix} x_1 & x_2 & x_3 & \cdots & x_n \end{bmatrix}$$
$$\downarrow \quad \downarrow \quad \downarrow \quad \downarrow$$
$$\begin{bmatrix} f(x_1) & f(x_2) & f(x_3) & \cdots & f(x_n) \end{bmatrix}$$

```
In [37]: data = [n**2 for n in range(10)]
data
```

Out[37]: [0, 1, 4, 9, 16, 25, 36, 49, 64, 81]

In []:

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



In [38]: money = 7

"else:f" →

```
if money > 9:  
    menu = 'A'  
elif money > 7:  
    menu = 'B'  
elif money > 5:  
    menu = 'C'  
else:  
    menu = 'D'  
  
print(money, menu)
```

7 C

In []:

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



In [46]: `def 주문(항목='A'):
 메뉴 = {'A': 10, 'B': 20}
 if not 항목 in 메뉴:
 return
 가격 = 메뉴[항목]
 return(항목, 가격)`

기본값

In [47]: 주문()

Out [47]: ('A', 10)

In [48]: 항목, 가격 = 주문('B')

'B' 20

In [49]: print('{}은 {}원입니다.'.format(항목, 가격))

B은 20원입니다.

In []:

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



In [50]: `def 집계(방법, *data):
 return 방법(data)`

In [53]: `집계(sum)`

Out[53]: 0

가변인자 (0개 이상)

f

집계(Sum, 1, 2, 3)

In [51]: `집계(sum, 1)`

Out[51]: 1

집계(Sum, [1, 2, 3])

2

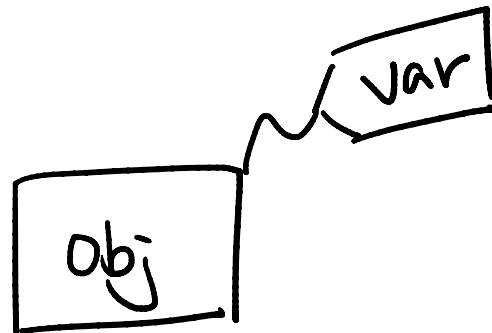
In [52]: `집계(sum, 1, 2, 3)`

Out[52]: 6

In []:

함수도 객체

1. 변수 ← 객체
2. 변수 ← 함수



Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



Out[52]: 6

In [54]: agg = 집계

In [55]: agg(sum, 1, 2, 3)

Out[55]: 6

In [60]: open('data.txt')

Out[60]: <_io.TextIOWrapper name='data.txt' mode='r' encoding='cp949'>



In []:

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



```
[1, 2, 3, 'A']
[4, 5, 6, 'B']
[7, 8, 9, 'C']
```

In [64]: data

Out[64]: [7, 8, 9, 'C']

In [65]: *수치, 라벨 = data

In [67]: 수치, 라벨

Out[67]: ([7, 8, 9], 'C')

In []:

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



Out[60]: <_io.TextIOWrapper name='data.txt' mode='r' encoding='cp949'>

In [68]:

```
data = []
for line in open('data.txt'):
    샘플 = [item.strip() for item in line.strip().split(',')]  
    *수치, 라벨 = 샘플  
    수치 = [int(n) for n in 수치]  
    data.append((라벨, 수치))
```

unpacking →

file

Str.method()

[for]

tuple

list

In [69]: data

Out[69]: [('A', [1, 2, 3]), ('B', [4, 5, 6]), ('C', [7, 8, 9])]

In [65]: *수치, 라벨 = data

In [67]: 수치, 라벨

Out[67]: ([7, 8, 9], 'C')

In []:

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



Code

수치 = [1, 2, 3] for i in 수치:
data.append((라벨, 수치))

'{}:{}\n'

In [69]: data

e.g. 'A: 1,2,3\n'

Out[69]: [('A', [1, 2, 3]), ('B', [4, 5, 6]), ('C', [7, 8, 9])]

In [71]: with open('data2.txt', 'w', encoding='utf-8') as file:
for sample in data:
 라벨, 수치 = sample
 file.write('{}:{}\n'.format(
 라벨, ','.join(str(n) for n in 수치)))

file.write(str)

AUTO
file.close()

In []:

[1, 2, 3]

↓ ↓ ↓ .join()

'1' '2' '3' → '1, 2, 3'

[] x { }

ndarray

스칼라

벡터

행렬

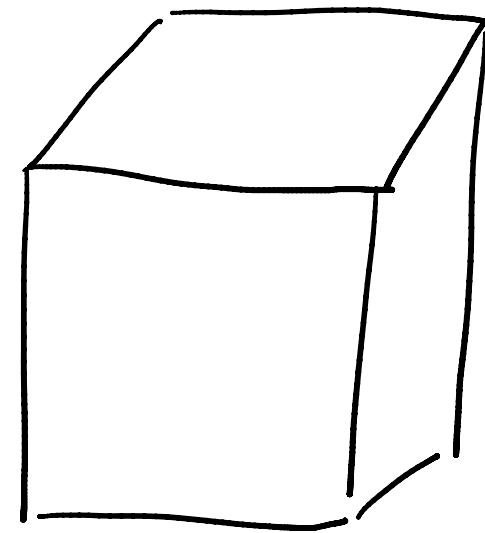
텐서 ($n > 2$)

X

$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

$$\begin{bmatrix} & & \\ & & \end{bmatrix}$$

$$\begin{bmatrix} & & \\ & & \end{bmatrix}$$



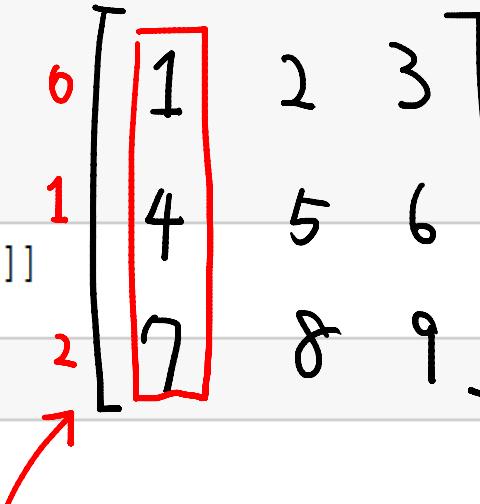
Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help

In [82]: `data = [[1,2,3], [4, 5, 6], [7, 8, 9]]`In [83]: `data2 = []
for row in data:
 new_row = [n*2 for n in row]
 data2.append(new_row)
data2`Out[83]: `[[2, 4, 6], [8, 10, 12], [14, 16, 18]]`In [84]: `data2[0]`Out[84]: `[2, 4, 6]`In [85]: `data[0][0], data[1][0], data[2][0]`Out[85]: `(1, 4, 7)`

In []:


$$\begin{matrix} & \begin{matrix} 1 & 2 & 3 \end{matrix} \\ \begin{matrix} 0 \\ 1 \\ 2 \end{matrix} & \left[\begin{matrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{matrix} \right] \end{matrix}$$

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help

In [86]: `import numpy as np`In [87]: `arr2d = np.array(data)`

ndarray

In [88]: `arr2d`Out[88]: `array([[1, 2, 3],
[4, 5, 6],
[7, 8, 9]])`

n-d "다차원"

In [89]: `type(arr2d)` "배열"Out[89]: `numpy.ndarray`

In []:

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



In [90]: [1, '2', 3.14]

Out[90]: [1, '2', 3.14]

In [91]: arr2d.dtype

Out[91]: dtype('int32')

In [92]: np.array([1, '2', 3.14])

Out[92]: array(['1', '2', '3.14'],
dtype='|<U11')

In []:

.astype()

PythonintNumpy

int8

16

32

64

UNICODE

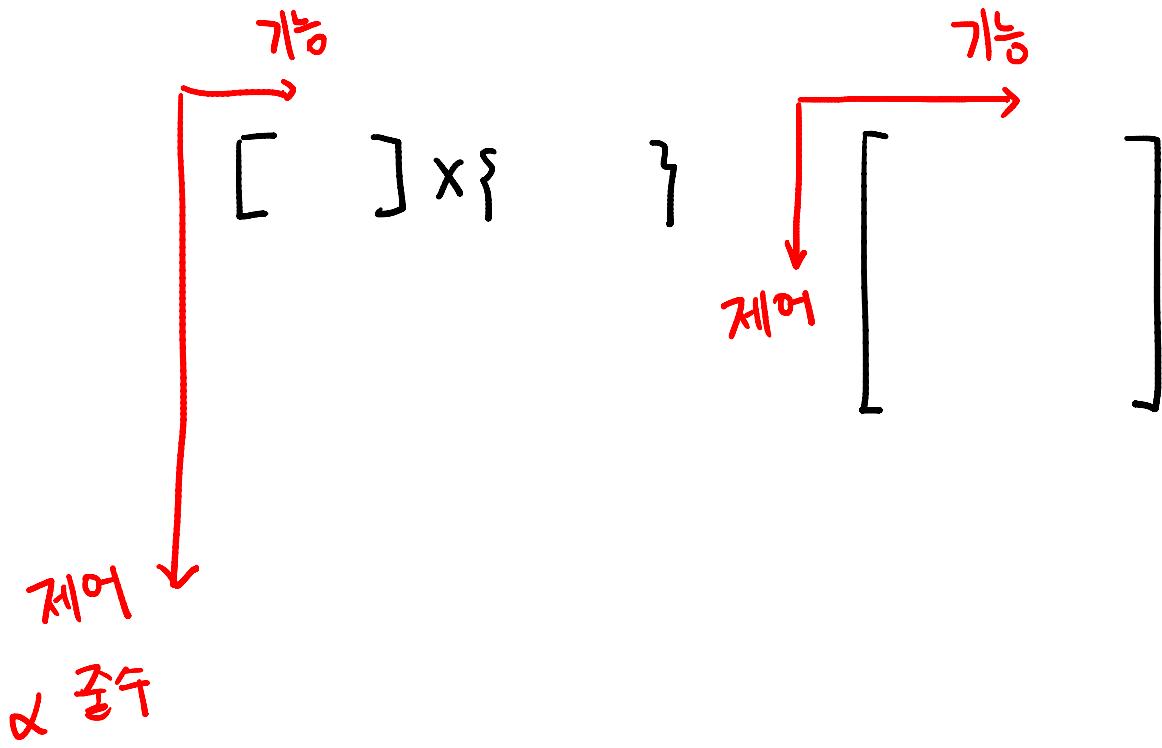
float

float32

64

ndarray

```
[ [ 1   2   3  
  4   5   6  
  .  
  7   8   9 ] ]
```



Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



[14, 10, 10]]

In [98]: `arr = np.arange(10**7)`
`nums = arr.tolist()`

ndarray
[0, 1, ..., 9,999,999]
list



In [99]: `%timeit [n*1.1 for n in nums]`

1 loop, best of 3: 866 ms per loop

In [100]: `%timeit arr * 1.1` x30

10 loops, best of 3: 28.9 ms per loop

In []:

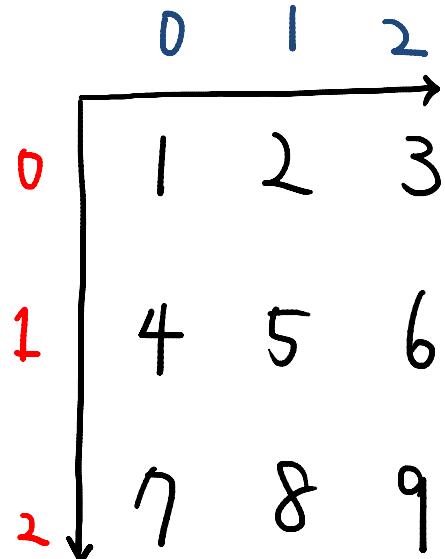
n-d Slice

$[:2]$

1 2 3
4 5 6

$[:, :2]$

0 1 2
4 5
7 8



$[1, :2]$

4 5

$[:2]$
 ~~$[0, 1:]$~~
 $[:2]$
 ~~$[0, -2:]$~~
2 3

5 6

팬시 셙인

list

$\left[\begin{bmatrix} 6, 2 \end{bmatrix} \right]$

1 2 3

7 8 9

$\left[\begin{bmatrix} 2, 0 \end{bmatrix} \right]$

7 8 9

1 2 3

0 1 2

0 1 2 3

1 4 5 6

2 7 8 9

$\left[: , \begin{bmatrix} 0, 2 \end{bmatrix} \right]$

1 3
4 6
7 9

$\left[: , \begin{bmatrix} 2, 0 \end{bmatrix} \right]$

3 1
6 4
9 7

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



In [113]: arr2d[:, [2, 0]] 0 1 2

Out[113]: array([[3, 1],
[6, 4],
[9, 7]]) 0 1 2 3
1 4 5 6

In [117]: arr2d[0, [0, 2]] 2 7 8 9

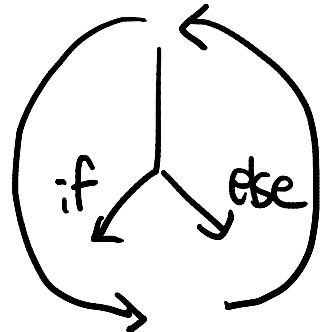
Out[117]: array([1, 3])

In [118]: arr2d[[1, 0], [0, 2]]

Out[118]: array([4, 3]) 4 5 6

In []: 1 2 3

7-11 & 1



Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



Trusted

Python 3

In [125]: arr = np.array([0, 1, 2, 3, -4, -5])

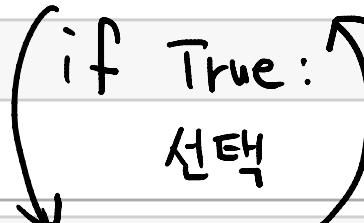


In [126]: arr

Out[126]: array([0, 1, 2, 3, -4, -5])

In [129]: arr[[True, True, True, False, False, False]]

Out[129]: array([0, 1, 2])



In [127]: arr > 0

Out[127]: array([False, True, True, True, False, False], dtype=bool)

In [128]: arr[arr > 0]

Out[128]: array([1, 2, 3])

In []:

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



Out[120]: array([1, 2, 3])

In [130]: arr2d

Out[130]: array([[1, 2, 3],
[4, 5, 6],
[7, 8, 9]])

3x3 → 9

In [131]: arr2d > 5

Out[131]: array([[False, False, False],
[False, False, True],
[True, True, True]], dtype=bool)

In [132]: arr2d[arr2d > 5]

Out[132]: array([6, 7, 8, 9])

In []:

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



```
[raise, raise, True],  
[ True, True, True]], dtype=bool)
```

In [132]: arr2d[arr2d > 5]

not True → False

Out[132]: array([6, 7, 8, 9])

~[True, False] → [False, True]

In [137]: arr > 0

Out[137]: array([False, True, True, True, False, False], dtype=bool)

In [136]: ~(arr > 0)

Out[136]: array([True, False, False, False, True, True], dtype=bool)

In []:

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help

**NOT**In [136]: `~(arr > 0)`Out[136]: `array([True, False, False, False, True, True], dtype=bool)`**AND**In [138]: `(arr > 0) & (arr < 3)`Out[138]: `array([False, True, True, False, False, False], dtype=bool)`**OR**In [139]: `(arr < 0) | (arr > 1)`Out[139]: `array([False, False, True, True, True, True], dtype=bool)`

In []:

배열 전치

TRANSPOSE

arr

0 3

1 4

2 5

3×2

arr. T

0 1 2

3 4 5

2×3

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



In [140]: arr2d

Out[140]: array([[1, 2, 3],
[4, 5, 6],
[7, 8, 9]])

TRANSPOSE

In [142]: arr2d.T

Out[142]: array([[1, 4, 7],
[2, 5, 8],
[3, 6, 9]])

In []:

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help

In [143]: `nums = list(range(1, 11))`In [144]: `nums`Out[144]: `[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]`In [145]: `from math import sqrt` \sqrt{n} In [146]: `sqrt(nums)`

TypeError

|| last)

<ipython-input-146-7015091d3fbe> in <module>()

----> 1 sqrt(nums)

Traceback (most recent ca

TypeError: must be real number, not list

In []:

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



```
In [148]: arr = np.array(nums)  
arr
```

```
Out[148]: array([ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10])
```

```
In [149]: np.sqrt(arr)
```

```
Out[149]: array([ 1.          ,  1.41421356,  1.73205081,  2.          ,  
    2.23606798,  
    2.44948974,  2.64575131,  2.82842712,  3.          ,  
    3.16227766])
```

```
In [ ]:
```

"벡터의 각 원소에 대해"

x	y	ufun(x)	ufunc(x,y)
x_1	y_1	$f(x_1)$	$f(x_1, y_1)$
x_2	y_2	$f(x_2)$	$f(x_2, y_2)$
x_3	y_3	$f(x_3)$	$f(x_3, y_3)$

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help

In [150]: `x = np.array([0, 1])`In [152]: `x`

X	Y	np.maximum(x,y)
0	1	1

Out[152]: `array([0, 1])`In [151]: `x[::-1]`

1	0	1
---	---	---

Out[151]: `array([1, 0])`In [153]: `np.maximum(x, x[::-1])`Out[153]: `array([1, 1])`

In []:

0 0 0

STEP (STRIDE)

[0, 1, 2, 3, 4, ..., 9]

[시작 : 끝 : 스텝]

[:] → [0, 1, 2, ..., 9] "복사"

e.g. [0:2] → $0 \leq i < 2$

STEP
[::2] → [0, 2, 4, 6, 8]

[0:] → $0 \leq i \leq \text{END}$

"BACK STEP"
[::-1] → [9, 8, 7, ..., 1, 0]

[:] → $0 \leq i \leq \text{END}$

[0:4:2] → $0 \leq i < 4$

[::2] → $0 \leq i \leq \text{END}$

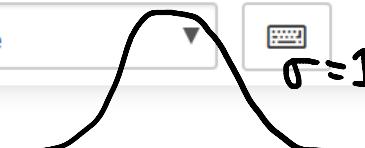
Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



Code



Out[153]: array([1, 1])

In [154]: arr = np.random.randn(5)

In [155]: arr

Out[155]: array([-0.57926298, -1.0692947 , 0.37433456, -0.03383657, -0.65861861])

In [156]: np.mean(arr)

Out[156]: -0.3933356598988681

In [157]: arr.mean()

Out[157]: -0.3933356598988681

In []:

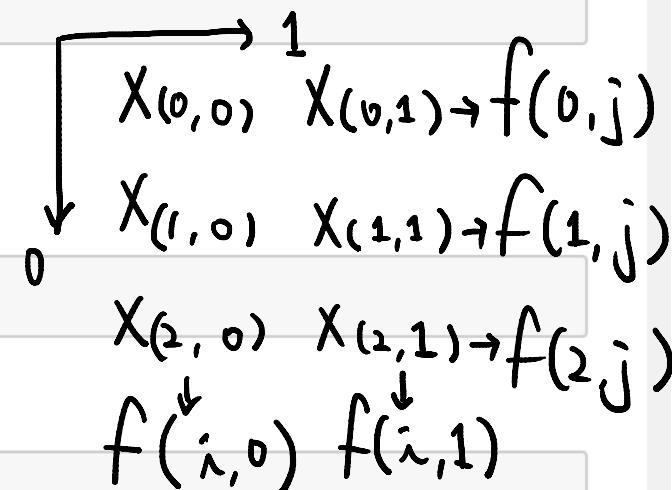
Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help

In [158]: `arr2 = np.random.randn(3, 2)`In [159]: `arr2`Out[159]: `array([[-0.25167777, -0.43402344],
[0.27307806, 1.7720567],
[0.046342 , 0.03345522]])`In [160]: `arr2.mean()`Out[160]: `0.23987179388638266`In [161]: `arr2.mean(axis=0)`Out[161]: `array([0.02258076, 0.45716283])`In [162]: `arr2.mean(axis=1)`Out[162]: `array([-0.34285061, 1.02256738, 0.03989861])`

In []:



Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



Code



== 1 .in1d()

In [164]: arr

0 False False

Out[164]: array([0, 1, 2, 3, 0, 1, 2, 3])

1 True True

In [165]: np.unique(arr)

2 False True

Out[165]: array([0, 1, 2, 3])

3 False False

In [166]: np.in1d(arr, [1, 2])

Out[166]: array([False, True, True, False, False, True, True, False], dtype=bool)

In []:

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



In [167]: `x = np.array([0, 1, 2, 3])
y = np.array([1, 2, 3, 4])`

$x \quad 0 \quad 1 \quad 2 \quad 3$

In [168]: `np.intersect1d(x, y)`

$y \quad 1 \quad 2 \quad 3 \quad 4$

Out[168]: `array([1, 2, 3])`

$\begin{array}{r} y \\ \hline 1 & 2 & 3 \end{array}$

In [169]: `np.union1d(x, y)`

$0 \quad 1 \quad 2 \quad 3 \quad 4$

Out[169]: `array([0, 1, 2, 3, 4])`

0

In [170]: `np.setdiff1d(x, y)`

— 0

Out[170]: `array([0])`

XOR 0

In [171]: `np.setxor1d(x, y)`

4

Out[171]: `array([0, 4])`

In []:

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



Code

Out[176]: array([[1, -1],
[-1, 1]])In [177]: $x * y$ Out[177]: array([[1, -1],
[-2, 2]])

$$\begin{array}{c} X \\ \left[\begin{array}{cc} x_0 & x_1 \\ x_2 & x_3 \end{array} \right] \end{array} \quad \begin{array}{c} y \\ \left[\begin{array}{c} y_0 \\ y_1 \end{array} \right] \end{array} \quad \begin{array}{c} x \cdot y \\ \left[\begin{array}{c} x_0y_0 + x_1y_1 \\ x_2y_0 + x_3y_1 \end{array} \right] \end{array}$$

Handwritten annotations: A red arrow points from the top of the column vector y to the second row of the matrix X. Another red arrow points from the bottom of the column vector y to the second element of the first row of the matrix X.

In [178]: $\text{np.dot}(x, y)$ $2 \times 2 \quad \cdot \quad 2 \times 1 \quad \rightarrow \quad 2 \times 1$ Out[178]: array([[0, 0],
[0, 0]])

In []:

$$3x + 6y - 5z = 12$$

$$AX = B$$

$$x - 3y + 2z = -2$$

$$X = A^{-1}B$$

$$5x - y + 4z = 10$$

$$\begin{bmatrix} 3 & 6 & -5 \\ 1 & -3 & 2 \\ 5 & -1 & 4 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 12 \\ -2 \\ 10 \end{bmatrix}$$

A X B

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



In [179]: `A = np.array([[3, 6, -5], [1, -3, 2], [5, -1, 4]])`
`B = np.array([12, -2, 10])`

"Linear
Algebra"

In [180]: `np.linalg.inv(A)` A^{-1}

Out [180]: `array([[0.15625 , 0.296875, 0.046875],`
`[-0.09375 , -0.578125, 0.171875],`
`[-0.21875 , -0.515625, 0.234375]])`

In [182]: `np.dot(np.linalg.inv(A), B)` $A^{-1} \cdot B$

Out [182]: `array([1.75, 1.75, 0.75])`

In []:

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help

In [188]: `np.random.randint(0, 11, size=(2, 5))`Out[188]: `array([[3, 6, 3, 4, 1],
 [10, 4, 3, 6, 3]])`In [189]: `%timeit [random.randint(0, 10) for _ in range(10**6)]`1 loop, best of 3: 1.73 s per loop **2s**In [190]: `%timeit np.random.randint(0, 11, size=10**6)`

100 loops, best of 3: 10.4 ms per loop

In []:

10ms × 200

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



In [192]: x

Out[192]: array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])

In [193]: np.random.shuffle(x) "원본 변경"

In [194]: x

Out[194]: array([0, 4, 7, 2, 8, 5, 3, 9, 1, 6])

In [195]: np.random.permutation(x) "새 배열 반환"

Out[195]: array([8, 5, 3, 6, 4, 0, 7, 2, 9, 1])

In [196]: x

Out[196]: array([0, 4, 7, 2, 8, 5, 3, 9, 1, 6])

In []:

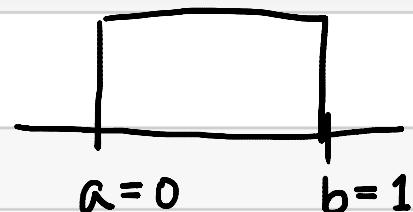
Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



Out[196]: array([0, 4, 7, 2, 8, 5, 3, 9, 1, 6])



In [197]: np.random.rand(5)

Out[197]: array([0.78472701, 0.22427951, 0.22028279, 0.91896181, 0.48810274])

a
v b

In [198]: np.random.uniform(0, 1, size=5)

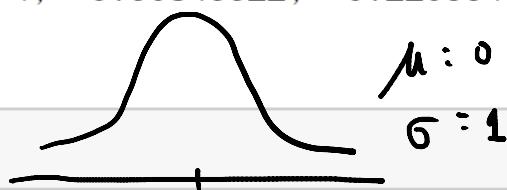
Out[198]: array([0.093135 , 0.55527574, 0.90843322, 0.22063438, 0.23387209])

In []:

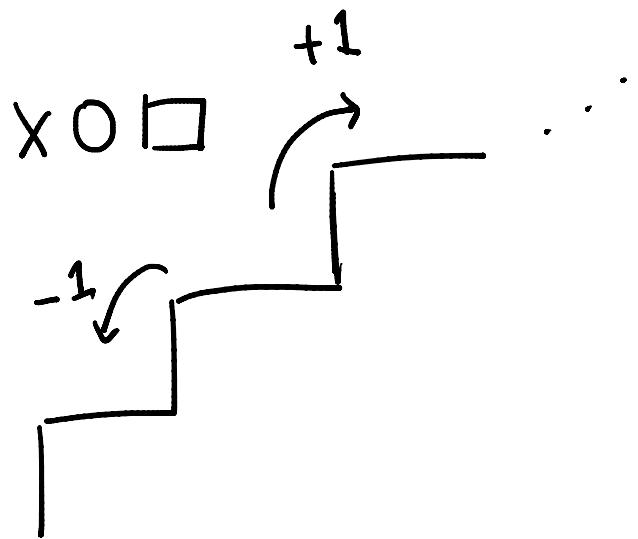
Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help

In [198]: `np.random.uniform(0, 1, size=5)`Out [198]: `array([0.093135 , 0.55527574, 0.90843322, 0.22063438, 0.23387209])`*"normal"*In [199]: `np.random.randn(5)`Out [199]: `array([-1.14781149, -0.33829873, 0.97664139, -0.8668183 , -0.01187981])`In [201]: `np.random.normal(0, 1, size=5)` μ , σ Out [201]: `array([1.60480895, 0.04597212, -0.01975557, -0.10895433, 0.58685902])`

In []:





1000

$$\text{승폐} \quad [0, 1, 1, 0, \dots]$$

$$\text{검은} \quad [-1, 1, 1, -1, \dots]$$

$$\text{Cumsum} \quad [-1, 0, 1, 0, \dots]$$

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



In [206]: 위치변화[:10]

Out[206]: [0, -1, -2, -3, -4, -5, -6, -7, -6, -7]

참 거짓

In [207]: 승패 = np.random.randint(0, 2, size=1000)

↓
np.where(Bool, x, y)

In [208]: 걸음 = np.where(승패 > 0, 1, -1)

In [209]: 위치변화 = 걸음.cumsum()

True x₁ y₁ → x₁

In [210]: 위치변화[:10]

False x₂ y₁ → y₁

Out[210]: array([-1, 0, 1, 0, -1, -2, -3, -4, -3, -2], dtype=int32)

In []:

승패

1000

]

:

5k

[

]

[

→

]

]

]

]

]

]

0 경음

1

.CumSum(1)

위치변화

np.where(>0, 1, -1)

]

→

File Edit View Insert Cell Kernel Widgets Help



위치 += 걸음
위치변화.append(위치)

1 loop, best of 3: 9.95 s per loop

10,600 ms

In [206]: 위치변화[:10]

65 ms

Out [206]: [0, -1, -2, -3, -4, -5, -6, -7, -6, -7]

In [215]: %%timeit

승패 = np.random.randint(0, 2, size=(5000, 1000))

걸음 = np.where(승패 > 0, 1, -1)

위치변화 = 걸음.cumsum(1)

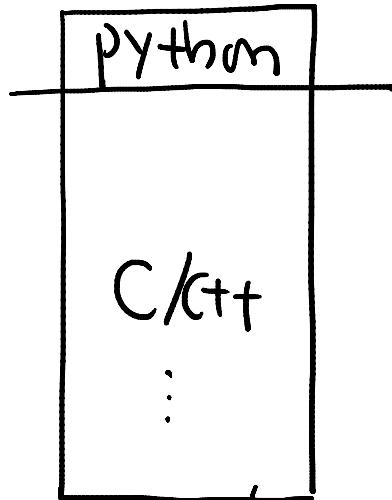
10 loops, best of 3: 64.7 ms per loop

In [210]: 위치변화[:10]

Out [210]: array([-1, 0, 1, 0, -1, -2, -3, -4, -3, -2], dtype=int32)

In []:

Interface



CPython



Tensorflow
Theano
⋮

GPU

SciPy

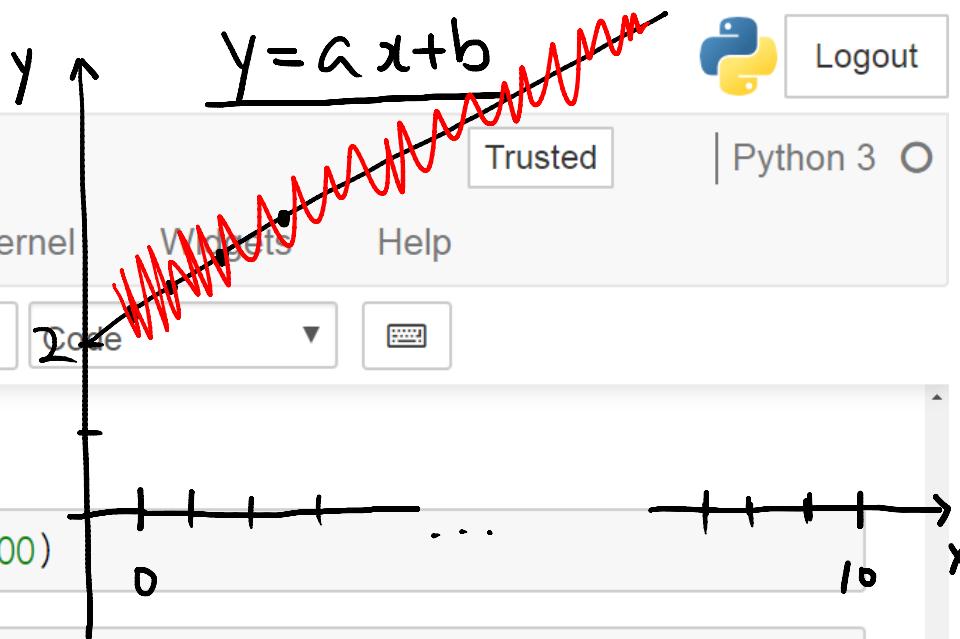
numpy.ndarray

[
 data
]



scipy.package.func(arr)

"Solution"



```
In [216]: x = np.linspace(0, 10, 100)
```

```
In [219]: x[:5]
```

```
Out[219]: array([ 0.          ,  0.1010101,  0.2020202,  0.3030303,  0.4040404])
```

```
In [220]: def model_func(x, a, b):
           return a*x+b
```

```
In [221]: y = model_func(x, 1, 2)
```

노이즈

```
In [222]: yn = y + 0.9 * np.random.randn(len(x))
```

```
In [ ]:
```

Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



In [220]: `def model_func(x, a, b):
 return a*x+b`

$a \leftarrow a + \Delta a$
 $b \leftarrow b + \Delta b$ ERROR

In [221]: `y = model_func(x, 1, 2)`

In [222]: `yn = y + 0.9 * np.random.randn(len(x))`

In [223]: `from scipy.optimize import curve_fit`

In [224]: `popt, pcov = curve_fit(model_func, x, yn)`

In [225]: `a, b = popt`

In [226]: `a, b`

Out[226]: `(0.96685101688127939, 2.0722312028871519)`

In []:

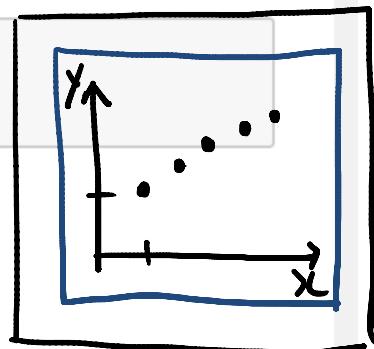
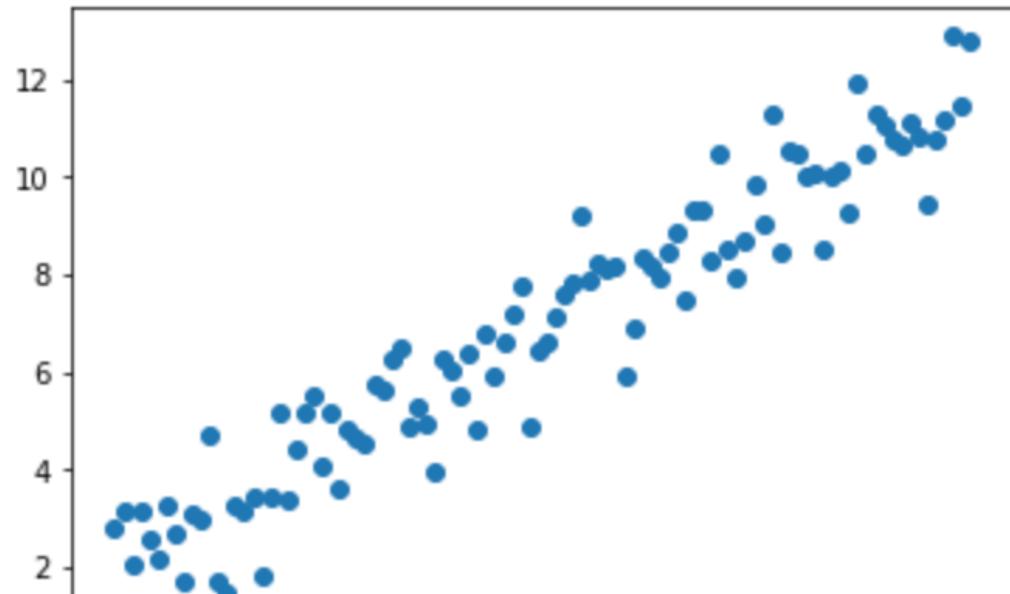
Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help

In [227]: `import matplotlib.pyplot as plt`In [228]: `%matplotlib inline` 노트북 그래프 출력 설정`plt.figure()`In [230]: `plt.figure()
plt.scatter(x, yn)`

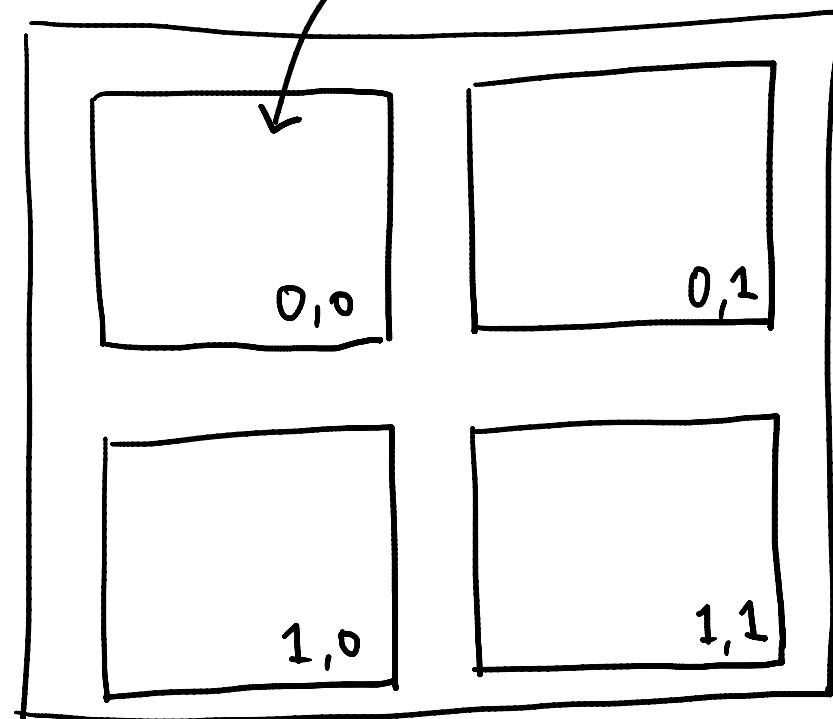
Out [230]: <matplotlib.collections.PathCollection at 0xb9cc898>



Matplotlib

subplot

Figure
"그리프 툴"



Trusted

Python 3

File Edit View Insert Cell Kernel Widgets Help



In [232]:

```
plt.figure()
plt.scatter(x, yn)
plt.plot(x, y, color='black')
plt.plot(x, model_func(x, a, b), color='red')
```

plt. 그림 추가

plot ↗

Out[232]: [`<matplotlib.lines.Line2D at 0xc19a940>`]

Scatter ⋮

