

numpy 모듈

ndarray I

홍정하

Jupyter Notebook 실행

▶ 실행

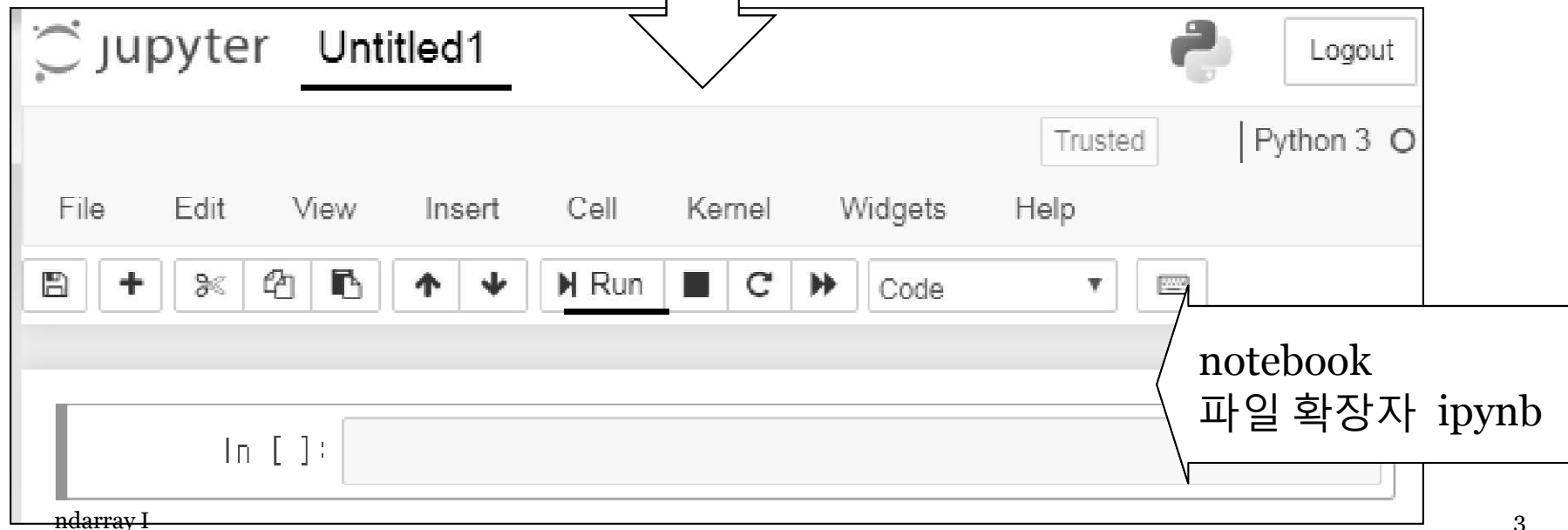
- 윈도우 명령 프롬프트/파워셸에 다음 코드 입력
jupyter notebook
- 작업폴더
 - 기본적으로 C:\Users\<사용자계정> 폴더
 - 파일을 불러오고 저장하는 기본 경로 위치

▶ 작업폴더 변경

- 윈도우 명령 프롬프트/파워셸에 다음 코드 입력
jupyter notebook --notebook-dir C:\project

▶ 작업폴더를 매번 변경하기 귀찮다면

jupyter_notebook_config.py 파일 활용 ⇒ 구글링 통해 구체적인 방법 확인 바람



import 모듈명

```
import numpy
```

```
a = [6, 7.5, 8, 0, 1]
```

```
numpy.array(a)
```

```
array([6. , 7.5, 8. , 0. , 1. ])
```

```
array(a)
```

```
-----  
NameError
```

```
<ipython-input-
```

```
----> 1 array(a)
```

```
NameError: name
```

```
del numpy
```

```
numpy.array(a)
```

```
-----  
NameError
```

```
<ipython-input-6-b8509d644d80> in <mo
```

```
----> 1 numpy.array(a)
```

```
NameError: name 'numpy' is not defined
```

모듈 제거

import 모듈명 as 약자

```
import numpy as np
```

```
a = [6, 7.5, 8, 0, 1]
```

```
np.array(a)
```

```
array([6. , 7.5, 8. , 0. , 1. ])
```

```
numpy.array(a)
```

```
-----  
NameError                                Trac  
<ipython-input-12-b8509d644d80> in <module>  
----> 1 numpy.array(a)
```

```
NameError: name 'numpy' is not defined
```

리스트 vs. ndarray

```
import numpy as np
```

```
a = [6, 7.5, 8, 0, 1]
```

```
b = np.array(a)
```

```
a
```

```
[6, 7.5, 8, 0, 1]
```

```
b
```

```
array([6. , 7.5, 8. , 0. , 1. ])
```

```
type(a)
```

```
type(b)
```

```
numpy.ndarray
```

multi-dimensional
array

```
(b)
```

```
[6.0, 7.5, 8.0, 0.0, 1.0]
```

```
a + 2
```

```
-----  
TypeError
```

```
<ipython-input-20-ead3c2111b4d> in <module>
```

```
----> 1 a + 2
```

```
TypeError: can only concatenate list (not "int") to list
```

```
b + 2
```

```
array([ 8. ,  9.5, 10. ,  2. ,  3. ])
```

```
a * 2
```

```
[6, 7.5, 8, 0, 1, 6, 7.5, 8, 0, 1]
```

```
b * 2
```

```
array([12., 15., 16.,  0.,  2.]
```

어떤 특성이
있는가?

```
b + b
```

```
array([12., 15., 16.,  0.,  2.])
```

```
b - b
```

```
array([0., 0., 0., 0., 0.])
```

```
b * b
```

```
array([36.  , 56.25, 64.  ,  0.  ,  1.  ])
```

```
b / b
```

```
c:\users\kleist\appdata\local\programs\python\python27\python.exe  
Entered in true_divide
```

```
"""Entry point for launching an IPython kernel"""
```

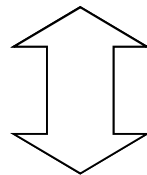
```
array([ 1.,  1.,  1., nan,  1.])
```

어떤 특성이
있는가?

ndarray 입력

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

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



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

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

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

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

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

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

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

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

```
np.array(((1,2,3,4), (5,6,7,8)))
```

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

어떤 특성이
있는가?

ndarray와 데이터구조, 차원

```
a = 'abcd'; b = list(a); c = tuple(a)
d = dict([('a', 1), ('b', 2)])
e = set(a)
```

```
np.array(a)
```

```
array('abcd', dtype='<U4')
```

```
np.array(b)
```

```
array(['a', 'b', 'c', 'd'], dtype='<U1')
```

```
np.array(c)
```

```
array(['a', 'b', 'c', 'd'], dtype='<U1')
```

```
np.array(d)
```

```
array({'a': 1, 'b': 2}, dtype=object)
```

```
np.array(e)
```

```
array({'a', 'd', 'b', 'c'}, dtype=object)
```

size, shape

```
np.array(a).size
```

```
1
```

```
np.array(b).size
```

```
4
```

```
np.array(c).size
```

```
4
```

```
np.array(d).size
```

```
1
```

```
np.array(e).size
```

```
1
```

```
np.array(a).shape
```

```
()
```

```
np.array(b).shape
```

```
(4,)
```

```
np.array(c).shape
```

```
(4,)
```

```
np.array(d).shape
```

```
()
```

```
np.array(e).shape
```

```
()
```

```
np.array([d, e])
```

```
array([{'a': 1, 'b': 2}, {'a', 'd', 'b', 'c'}], dtype=object)
```

```
np.array([d, e]).size
```

```
np.array([d, e]).shape
```

```
f = np.array(b)
f
```

```
array(['a', 'b', 'c', 'd'], dtype='<U1')
```

```
f.shape = (2, 2)
f
```

```
array([[ 'a', 'b'],
       [ 'c', 'd']], dtype='<U1')
```

```
f.size
```

```
4
```

```
f.shape = (1, 4)
f
```

```
array([[ 'a', 'b', 'c', 'd']], dtype='<U1')
```

```
f.size
```

```
4
```

```
f.shape = (4, 1)
f
```

```
array([[ 'a'],
       [ 'b'],
       [ 'c'],
       [ 'd']], dtype='<U1')
```

```
f.size
```

```
4
```

ndarray 연산

```
a = np.array([6, 7.5, 8, 1])  
a
```

```
array([6. , 7.5, 8. , 1. ])
```

```
a + 2
```

```
array([ 8. ,  9.5, 10. ,  3. ])
```

```
a + a
```

```
array([12., 15., 16.,  2.])
```

```
a * list(a)
```

```
array([36. , 56.25, 64. ,  1. ])
```

```
a.shape = (2,2)
```

```
a
```

```
array([[6. , 7.5],  
       [8. , 1. ]])
```

```
a + a
```

```
array([[12., 15.],  
       [16.,  2.]])
```

ndarray 입력과 차원

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

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

```
b.size
```

```
b.shape
```

```
b.shape = (2, 2, 2)  
b
```

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

```
b.shape = (2, 2, 3)  
b
```

```
-----  
ValueError:   
<ipython>
```

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

```
array([[1, 2, 3, 4],  
       [4, 5, 6, 7]])
```


dtype

```
a = np.array(['a', 'b', 'c'])  
b = np.array([1, 2, 3])  
c = np.array([1, 2., 3])
```

```
a.dtype
```

```
dtype('<U1')
```

```
b.dtype
```

```
dtype('int32')
```

```
c.dtype
```

```
dtype('float64')
```

```
np.array([b, b]).dtype
```

```
dtype('int32')
```

```
np.array([b, c]).dtype
```

```
dtype('float64')
```

```
np.array([a, b]).dtype
```

```
dtype('<U11')
```

```
np.array([1, 2, 3], dtype='float').dtype
```

```
dtype('float64')
```

```
np.array([1, 2., 3], dtype='int').dtype
```

```
dtype('int32')
```

```
np.array([1, 2., 3], dtype='str').dtype
```

```
dtype('<U3')
```

```
b.astype('str')
```

```
array(['1', '2', '3'], dtype='<U11')
```

```
b.astype('float')
```

```
array([1., 2., 3.])
```

연습문제 1: 네모 칸 코드?

```
a = [1, 2, 3, 4, 5, 6, 7, 8]
b = 
b
[10, 20, 30, 40, 50, 60, 70, 80]
```

변수 a 이용

```
c = 
c
array([10., 20., 30., 40., 50., 60., 70., 80.])
```

변수 b 이용

```

c
array([[10., 20., 30., 40.],
       [50., 60., 70., 80.]])
```

변수 c 이용

연습문제 2: 네모 칸 출력결과?

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

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

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