

```

1 import numpy as np
2 d = np.array([[1,2,3,4,5],[2,4,5,6,7],[5,7,8,9,10]]) #배열 만들기
3 d

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

```

```
1 d[0][-1]
```

```
5
```

```
1 d[1:,3:] #행, 열
```

```
array([[ 6,  7],
       [ 9, 10]])
```

```
1 d[0:2,2:]
```

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

- 배열의 크기 알아내기 : shape

```
1 d.shape #배열의 크기 알아내기
```

```
(3, 5)
```

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

```
2 e.dtype #배열 유형검사
```

```
dtype('int64')
```

▼ 배열 원소 유형 바꾸기:astype()

```
1 data = np.arange(1,5)
```

```
2 data.dtype
```

```
3 data
```

```
array([1, 2, 3, 4])
```

```
1 data.astype('float64')
```

```
array([1., 2., 3., 4.])
```

```
1 np.zeros((2,10)) #2행 10열
```

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

```
1 np.ones((2,10))
```

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

np.arange() 자동 생성

```
1 num = np.arange(2,10)
```

```
2 num
```

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

```
1 a = np.ones((2,3))
```

```
2 a
```

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

행과 열을 바꾸기: np.transpose()

```
1 b = np.transpose(a)
```

```
2 b
```

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

```
1 arr1 = np.array([[2,3,4],[6,7,8]])
```

```
2 arr2 = np.array([[12,23,14],[36,47,58]])
```

```
3
```

```
4 arr1+arr2
```

```
5
```

```
array([[14, 26, 18],  
       [42, 54, 66]])
```

```
1 arr1*arr2
```

```
array([[ 24,  69,  56],  
       [216, 329, 464]])
```

```
1 arr1/arr2
```

```
array([[0.16666667, 0.13043478, 0.28571429],
       [0.16666667, 0.14893617, 0.13793103]])
```

```
1 arr3 = np.array([100,200,300]) # 열의 길이가 맞아야 계산됨
2 arr1 + arr3
```

```
array([[102, 203, 304],
       [106, 207, 308]])
```

- 파이썬 리스트와 배열의 차이점

```
1 d = np.array([[1,2,3,4,5],[2,4,5,6,7],[5,7,8,9,9]])
2 d_list = [[1,2,3,4,5],[2,4,5,6,7],[5,7,8,9,9]]
3 d_list
```

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

```
1 d
```

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

```
1 type(d_list)
```

```
list
```

```
1 type (d)
```

```
numpy.ndarray
```

```
1 d_list[:2] = 0
```

```
-----
TypeError                                Traceback (most recent call last)
<ipython-input-68-7779e178cd5c> in <module>()
----> 1 d_list[:2] = 0
```

```
TypeError: can only assign an iterable
```

SEARCH STACK OVERFLOW

```
1 d[:2]=0
```

```
2 d
```

```
array([[0, 0, 0, 0, 0],
       [0, 0, 0, 0, 0],
       [5, 7, 8, 9, 9]])
```

