

# Numpy Cheat Sheet

Numpy Provides tools for working with arrays.  
All of the following code examples to the arrays below.

## Numpy Arrays

1D Array

1	2	3
---	---	---

2D Array

1.5	2	3
4	5	6

axis1  
axis0

Import numpy:

```
import numpy as np
```

create arrays:

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

```
b = np.array([(1.5, 2, 3), (4, 5, 6)], dtype=Float)
```

```
c = np.array([(1.5, 2, 3), (4, 5, 6)],  
              [(3, 2, 1), (4, 5, 6)]],  
              dtype = Float)
```

Initial placeholders:

```
np.zeros((3, 4)) # create an array of zeros
```

```
np.ones((2, 3, 4), dtype=np.int16)
```

```
d = np.arange(10, 25, 5)
```

```
np.linspace(0, 2, 9)
```

```
e = np.full((2, 2), 7)
```

```
f = np.eye(2)
```

```
np.random.random((2, 2))
```

```
np.empty((3, 2))
```

Saving & Loading On Disk:

```
np.save('my_array', a)
```

```
np.savez('array.npz', a, b)
```



```
np.load('my_array.npy')
```

Saving & Loading Text Files:

```
np.loadtxt('my_file.txt')
```

```
np.genfromtxt('my_file.csv',  
              delimiter = ',', )
```

```
np.savetxt('myarray.txt', a,  
           delimiter = ' . ')
```

Inspecting Your Array:

```
a.shape
```

```
len(a)
```

```
b.ndim
```

```
e.size
```

```
b.dtype # data type
```

```
b.dtype.name
```

```
b.astype(int) # change data type
```

Data Types:

```
np.int64
```

```
np.float32
```

```
np.complex
```

```
np.bool
```

```
np.object
```

```
np.string_
```

```
np.unicode_
```

Array Mathematics

Arithmetic Operations:

```
>>> g = a - b
```

```
array([[ -0.5,  0. ,  0. ],  
       [ -3. ,  3. ,  3. ]])
```

```
>>> np.subtract(a, b)
```

```
>>> b + a
```

```
array([[ 2.5,  4. ,  6. ],  
       [ 5. ,  7. ,  9. ]])
```

```
>>> np.add(b, a)
```

```
>>> a/b  
array([[0.66666667, 1., 1.],  
       [0.25, 0.4, 0.5]])  
>>> np.divide(a,b)
```

```
>>> a*b  
array([[1.5, 4., 9.],  
       [4., 10., 18.]])  
>>> np.multiply(a,b)
```

```
>>> np.exp(b)  
>>> np.sqrt(b)  
>>> np.sin(a)  
>>> np.log(a)  
>>> e.dot(f)
```

Aggregate functions:

```
a.sum()  
a.min()  
b.max(axis=0)  
b.cumsum(axis=1) # cumulative sum  
a.mean()  
b.median()  
a.corrcoef() # correlation coefficient  
np.std(b) # standard deviation
```

Copying array:

```
h = a.view() # create a view  
np.copy(a)  
h = a.copy() # create a deep copy
```

Sorting arrays:

```
a.sort() # sort an array  
c.sort(axis=0)
```



## Array Manipulation

Transposing Array:

`i = np.transpose(b)`

`i.T`

Changing Array Shape:

`b.ravel()`

`g.reshape(3, -2)`

Adding / removing elements:

`h.resize((2, 6))`

`np.append(h, g)`

`np.insert(a, 1, 5)`

`np.delete(a, [1])`

Combining arrays:

`np.concatenate((a, d), axis = 0)`

`np.vstack((a, b))` # stack vertically

`np.hstack((e, f))` # stack horizontally

Splitting arrays:

`np.hsplit(a, 3)` # split horizontally

`np.vsplit(c, 2)` # split vertically

Subsetting:

`b[1, 2]`

1.5	2	3
4	5	6

Slicing:

`a[0:2]`

1	2	3
---	---	---

Boolean Indexing:

`a[a < 2]`

1	2	3
---	---	---