

TO START

```
import numpy as np
# optional: to shorten the writing
# avoid to type np.random
from numpy.random import randint
```

ARRAYS

```
np.array(my_list)          create array from list
np.array(my_matrix)        create array from matrix
```

METHODS (to create arrays)

```
np.arange(n1, n2)          create array
np.arange(n1, n2, n3)      create array with step
np.zeros() *               zeroes array
np.ones() *                ones array
np.linspace(n1,n2,n3) *    evenly spaced numbers in an interval
np.eye()                   identity matrix
np.random.rand() *         random from uniform distribution
np.random.randn()          random from normal distribution
np.random.randint()        random integers
                             from low (inclusive)
                             to high (exclusive)
```

np.zeros/ones can take two numbers (rows and columns)
linspace(): n3 numbers from n1 to n2.
np.random.rand() can take 2 numbers (rows and columns).

ARRAY ATTRIBUTES AND METHODS

```
arr.reshape(r,c)*         change shape of array
arr.max()                  find max value
arr.argmax()               find index of maxim value
arr.min()                  find min value
arr.argmin()               find index of minim value
arr.shape                  return array shape
arr.reshape().shape        reshape array and
                             return array shape
arr.dtype                  return array type
```

(r,c) mean "row, column"

ARRAY INDEXING AND SELECTION

```
arr[]                      select item in array
arr[x:x]                   select items in range
arr[:x]                    select items up to x
arr[x:]                    select items from x and beyond
arr[0:5]=100 *            broadcast value in range
arr.copy()                 copy the array
arr_2d[r,c]*              select in 2D arrays
arr_2d[r][c]              select in 2D arrays (opt.2)
arr_2d[:2,1:]*            slicing (2 is not included)
arr_2d[2]                  show third row
arr_2d[2, :]              show third row (opt.2)
arr_2d[[2,4,6,8]]         show row 2,4,6,8
arr[arr>x]                 conditional selection
```

note: changes will happen also into the original array
[r,c] mean "row, column"

When slicing in this example, 2, is not included. Generally, the term after the column (:) is not included when slicing.

OPERATIONS

```
arr + arr                  sum two arrays
arr * arr                  multiply two arrays
arr - arr                  subtract two arrays
arr / arr                  divide two arrays
arr ** x                   array to exponential
np.sqrt(arr)               square root of array
np.exp(arr)                exponential of array (e^x)
np.max(arr)                find max value
np.sin(arr)                sin of array
np.log(arr)                log of an array
```



By **DarioPittera** (aggialavura)

Not published yet.

Last updated 24th June, 2019.

Page 1 of 1.

Sponsored by **Readable.com**

Measure your website readability!

<https://readable.com>