

# numpy\_mathematical\_functions\_workshop\_answers

September 21, 2020

## 1 NUMPY MATHEMATICAL FUNCTIONS WORKBOOK ANSWERS

Remember, there are different ways to write code to get the same answer, so your answer can be correct and different to the answer example!

If you feel stuck and want some in person help, then have a look at the events page to join in a workshop <https://swamphen.co.uk/events>.

```
In [ ]: # import numpy
import numpy as np
```

```
In [ ]: # set up an array of Booleans, integer, float, strings 5 long each
bool_array = np.array([True, False, False, True, True])
int_array = np.array([7,6,5,4,3])
flt_array = np.array([1.1,2.2,3.3,4.4,5.5])
str_array = np.array(['another', 'one', 'bites', 'the', 'dust'])
```

```
In [ ]: # add your Boolean and float arrays together
new_array = np.add(bool_array, flt_array)
```

```
In [ ]: # why does this work?

# True = 1, False = 0
```

```
In [ ]: # add your Boolean, integer and float arrays together
np.add(new_array, int_array)
```

```
In [ ]: # divide your float array by your integer array
np.divide(flt_array, int_array)
```

```
In [ ]: # cube your integer array
np.power(int_array, 3)
```

```
In [ ]: # what is the maximum of your integer array
np.max(int_array)
```

```
In [ ]: # calculate the standard deviation of
np.std(bool_array)
```

```
In [ ]: # is this a meaningful thing to calculate?

        # no

In [ ]: # calculate a more meaningful standard deviation
        np.std(flt_array)

In [ ]: # calculate the average of your integer array giving numbers 5 and under a weight of 2
        # all other numbers have a weight of 1
        np.average(int_array, weights = np.array([1,1,2,2,2]))

In [ ]: # calculate the ceiling of your float array
        np.ceil(flt_array)

In [ ]: # assume your float array is a list of angles in degrees
        # convert them into radians and calculate the sin of them
        rad = np.radians(flt_array)
        np.sin(rad)
```