**Python**

**Data types:**

Dictionaries have keys and values {‘key1’: ‘value1’, ‘key2’: ‘value2’ }

Dictname.items gives both the key and the value -> 2 variables

Named tuples (use package collections) are defined as such Person = collections.namedtuple("Person", ["name", "lastname", "birthday"])

Can have an array of dictionaries [ {}, {}, {}]

Can enclose an for+if statement in square brackets to return a smaller array of dictionaries [person for person in peopledict if person["birthday"] == "July 15"]

Classes of objects -> attributes and methods

Sets defined as words = set() can do ste\_name.add(“something”)

**Text files**

.strip(), .lower(), sorted(…)

**Getting changing working directories:**

import os

from pathlib import Path

os.getcwd()

p = Path.cwd()

print(p.parent)

os.chdir(p.parent)

[Get and Change the Current Working Directory in Python | note.nkmk.me](https://note.nkmk.me/en/python-os-getcwd-chdir/)

**Numpy:**

Np.loadtxt()

Np.load()

Np.zeros(number, dtype)

Dtypes include ‘d’ or ‘np.float64’ and more

Give shape as number, or in (brackets, like this)

np.zeros\_like()

np.empty() # memory is allocated but not initialized so get some random values

np.linspace(0, 1, 16) # extrema are included

np.arange(0, 1.5, 0.1) # extension of range so end number not included

random\_2d = np.random.random((8,8)) # generate array of random numbers distributed uniformly between 0 and 1 when given a shape

np.random.randint(0, 10, (8,8)) # get integers in a specified range

# there more from the random submodule

Can save numpy array as .npy files or .txt

#see pdf cheatsheet

Array slicing

Numpy slicing creates new object pointing to same space in memory