Python

Page maintainer: Bouwe Andela @bouweandela

Python is the "dynamic language of choice" of the Netherlands eScience Center. We use it for data analysis and data science projects using the SciPy stack and Jupyter notebooks, and for many other types of projects: workflow management, visualization, NLP, web-based tools and much more. It is a good default choice for many kinds of projects due to its generic nature, its large and broad ecosystem of third-party modules and its compact syntax which allows for rapid prototyping. It is not the language of maximum performance, although in many cases performance critical components can be easily replaced by modules written in faster, compiled languages like C(++) or Cython.

The philosophy of Python is summarized in the Zen of Python. In Python, this text can be retrieved with the `import this` command.

Project setup

When starting a new Python project, consider using our Python template. This template provides a basic project structure, so you can spend less time setting up and configuring your new Python packages, and comply with the software guide right from the start.

Use Python 3, avoid 2

Python 2 and Python 3 have co-existed for a long time, but starting from 2020, development of Python 2 is officially abandoned, meaning Python 2 will no longer be improved, even in case of security issues. If you are creating a new package, use Python 3. It is possible to write Python that is both Python 2 and Python 3 compatible (e.g. using Six), but only do this when you are 100% sure that your package won't be used otherwise. If you need Python 2 because of old, incompatible Python 2 libraries, strongly consider upgrading those libraries to Python 3 or replacing them altogether. Building and/or using Python 2 is probably discouraged even more than, say, using Fortran 77, since at least Fortran 77 compilers are still being maintained.

- Things you're probably not using in Python 3 – but should
- Six: Python 2 and 3 Compatibility Library
- 2to3: Automated Python 2 to 3 code translation
- python-modernize: wrapper around 2to3