

Programming basics and Python for everyone

Carlo Resta

Abstract

This course will hopefully provide you with everything you need to start writing computer programs. No previous experience is required. We'll begin with a few fundamentals, including:

- what kind of reasoning is necessary to write code;
- how to write basic programs in Python (mostly to explore and analyse data);
- what to do when the course is over and you don't know how to do something;
- how to do good and accessible graphics with Python.

We'll take some time during the first meeting to tailor the lesson plan to the specific interests of the group. To make the lessons more useful, everyone is encouraged to think of a specific application relating to their research: during the course we'll work together to turn it into a functioning program, using a flipped classroom approach. This means some homework, I'm sorry. Since it's everywhere and it really makes no sense to ignore it, we'll also talk about how to use and control AI-assisted coding tools... without being controlled by them.

Lessons (about 2 hours each)

Unfortunately a laptop will be necessary. If you're not able to find one, please let me know and we'll try to find a solution. The course will be held in person, because most of the lessons will be interactive and will involve group work. For the same reason, there is a maximum of 30 participants.

Some of the more advanced contents we'll decide together, so take this only as a tentative lesson plan:

1. **Introduction:** hopes and expectations from the course. What kind of reasoning is necessary to write programs, and some examples (you'll start right away). Participatory design of the course contents.
2. **Basic data structures and functionalities of Python:** how to write basic but useful code. We'll look at the tools with which all Python code is written.
3. **Some libraries to manipulate data** (for example `numpy`, `scipy` and `pandas`): interacting with tables, text and external files.
4. **Graphics 1:** why should we do graphics with code, and how.
5. **Graphics 2:** less about code, more about making good and most of all accessible plots.
6. **Flipped classroom 1:** we work together on your projects and share knowledge.
7. **Flipped classroom 2:** same as before.
8. **Flipped classroom 3:** final lesson. Celebration for having made it this far (possibly cake).

You don't need to take the whole seminar if you already know some stuff. Any attendance of at least 50% will be considered okay, and you'll get a certificate for the corresponding number of hours. But please come to the first lesson so that we can plan your programme together. There is no mandatory exam; only the (small) personal project, which will hopefully be useful to your research too. But if you need an exam to get your credits recognised we can do it (or something equivalent).