

Computing Workshop: Intro Python – Syllabus

Jacob Errington & Eric Mayhew*

Summer 2022 – 15 to 19 August

Website <https://computing-workshop.com/>

Location Burnside Hall, 5th floor, Geographic Information Center (GIC)

Time 15 - 19 August, 10 am - 3 pm with lunchbreak noon - 1 pm

Audience Incoming graduate students without background in computer science, interested in using Python for their research

Learning Goals

A participant will be able to:

- Use control flow mechanisms, e.g. `if-` and `for-`statements.
- Organize a program into functions; follow the Single Responsibility Principle.
- Use common I/O mechanisms: standard input/output, files, HTTP.
- Run Python scripts on their computer.
- Justify/critique the use of software in solving real-world problems.

Lesson sequence

15 Aug. Welcome! Land acknowledgement. Statements, expressions, types, and values. Program execution. Compilation vs interpretation.

16 Aug. Control flow: conditions, loops, functions.

17 Aug. Data structures: lists, dictionaries, trees, graphs.

18 Aug. The environment: filesystem, network, standard I/O. Batch vs interactive.

19 Aug. Applications: basic data science with NumPy, SciPy.

Description

Software is created by writing a program in a *programming language*. This course provides a guided first step towards software development by equipping you with fundamental skills in a very popular and beginner-friendly programming language, *Python*. These fundamentals are easily transferrable to other programming languages, too!

This is a five-session, one-week intensive workshop, consisting of twenty total hours of classtime. Each day will consist of a 2-hour morning session and a 2-hour afternoon session. The morning session will lean more on the theoretical side, with lectures punctuated by small group discussions and worksheets. The afternoon session will lean more on the practical side, where you will work on coding problems in pairs to cement your understanding of the morning material.

This course aims to give a foundation for further study of popular kinds of Python software development, in particular machine learning and data science. At the same time, we aim to impart to you a critical understanding of software development, and especially of machine learning: you must think not only of whether software *can* solve a problem, but especially of whether it *should*. Expect to make ethical considerations before and alongside the software development you will undertake in solving problems in this course!

*Funding and venue is kindly provided by McGill's Computational and Data Systems Initiative (CDSI), <https://mcgill.ca/cdsi>