





16

Introduction to Python | HT17

Secure | <https://nbisweden.github.io/PythonCourse/ht17/>

NBS

TopicsProjectPreliminariesHelp

Introduction to Python - HT17

This course provides a practical introduction to the writing of [Python](#) programs for the complete novice. Participants are lead through the core aspects of Python illustrated by a series of example programs. Upon completion of the course, attentive participants will be able to write simple Python programs from scratch and to customize more complex code to fit their needs.

The craft of programming is about taking design decisions to avoid overwhelming complexity and permit easy maintenance over time, insuring reliability (which goes beyond debugging) and utilizing computer resources efficiently.

- The course is suitable for complete beginners and assumes no prior programming experience (beyond the ability to use a text editor).
- A very basic knowledge of UNIX would be an advantage, such as navigating through folders and issuing commands at a shell prompt. We will not teach Unix in detail: Other course are available at SciLifeLab for it.

Before the first lecture, we require you to follow these [preliminary steps](#).

Schedule

From Monday October 9th, to Friday October 13th 2017 (week 41).

- 09:00 - 12:00: Lectures + Hands-on (including a Fika break)
- 12:00 - 13:00: Lunch
- 13:00 - 15:00: Lectures + Hands-on
- 15:00 - 17:00: Practice time (with assistants)

Course Content

During this course, you will learn about:

- Core concepts about Python syntax: Data types, blocks and indentation, variable scoping, iteration, functions, methods and arguments
- Different ways to control program flow using loops and conditional tests
- Regular expressions and pattern matching
- Writing functions and best-practice ways of making them usable
- Reading from and writing to files
- Code packaging and Python libraries
- How to work with biological data using external libraries (if time allows).

Learning Outcomes

After this course you should be able to:

- Edit and run Python code
- Write file-processing python programs that produce output to the terminal and/or external files.
- Create stand-alone python programs to process biological data
- Know how to develop your skills in Python after the course (including debugging)

Learning objectives (ie goals for the teachers)

- Increase the student's toolbox for better quality and performance at work
- Make students understand that there is more to programming than only knowing the syntax of a language. This expertise is precisely what [NBIS](#) provides.

Format

15

Introduction to Python - HT17

This course provides a practical introduction to the writing of [Python](#) programs for the complete novice. Participants are lead through the core aspects of Python illustrated by a series of example programs. Upon completion of the course, attentive participants will be able to write simple Python programs from scratch and to customize more complex code to fit their needs.

The craft of programming is about taking design decisions to avoid overwhelming complexity and permit easy maintenance over time, insuring reliability (which goes beyond debugging) and utilizing computer resources efficiently.

- The course is suitable for complete beginners and assumes no prior programming experience (beyond the ability to use a text editor).
- A very basic knowledge of UNIX would be an advantage, such as navigating through folders and issuing commands at a shell prompt. We will not teach Unix in detail: Other course are available at SciLifeLab for it.

Before the first lecture, we require you to follow these [preliminary steps](#).

» Schedule

From Monday October 9th, to Friday October 13th 2017 (week 41).

- **09:00 - 12:00:** Lectures + Hands-on (including a Fika break)
- **12:00 - 13:00:** Lunch
- **13:00 - 15:00:** Lectures + Hands-on
- **15:00 - 17:00:** Practice time (with assistants)

» Course Content

During this course, you will learn about:

- Core concepts about Python syntax: Data types, blocks and indentation, variable scoping, iteration, functions, methods and arguments
- Different ways to control program flow using loops and conditional tests
- Regular expressions and pattern matching
- Writing functions and best-practice ways of making them usable
- Reading from and writing to files
- Code packaging and Python libraries
- How to work with biological data using external libraries (if time allows).

» Learning Outcomes

After this course you should be able to:

- Edit and run Python code
- Write file-processing python programs that produce output to the terminal and/or external files.
- Create stand-alone python programs to process biological data
- Know how to develop your skills in Python after the course (including debugging)

Learning objectives (ie goals for the teachers)

- Increase the student's toolbelt for better quality and performance at work
- Make students understand that there is more to programming than only *knowing* the syntax of a language. This expertise is precisely what [NBIS](#) provides.

» Format