

Python Programming for Chemists: Python Setup

Christoph Loschen
WS 2024/2025

What you will learn...

- Basics of programming: data structures
- Reading & plotting data with Python
- How to program simple chemical models
- Leverage great tools & libraries from smart people (open source, ChatGPT)
- How to use programming to solve chemical problems!
- Improve your understanding of chemistry by programming!
- **Fun with (Python) programming :-)**

Overview

- **Lectures**
 - Computer & programming basics
 - Python data types & data structures
 - Data analysis & visualization
 - Introduction to scientific computing (numpy&scipy)
 - Outlook: Cheminformatics
- **Assignment**
 - Present a concept for implementing a chemical model / equation
 - Write a small program implementing the model

Useful Textbooks

- Lubanovic, Bill. Introducing Python: Modern Computing in Simple Packages. " O'Reilly Media, Inc.", 2019.
- Hill, Christian. Python for Chemists. Cambridge University Press, 2023.
- Haffner, Ernst Georg. Informatik für Dummies, Das Lehrbuch. John Wiley & Sons, 2023.

Useful Links

Learning Python

<https://www.learnpython.org/>

<https://www.freecodecamp.org/learn/python-for-everybody/>

Learning Python via Gamification

<https://www.codingame.com/home>

<https://developer.apple.com/swift-playgrounds/>

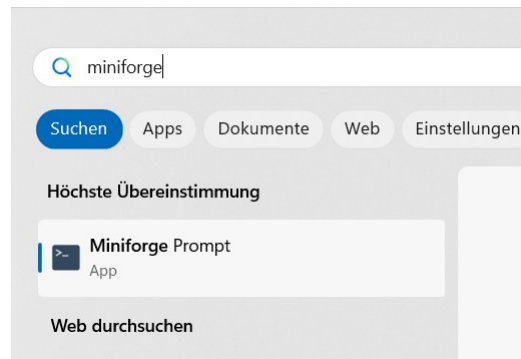
Python Overview / “Cheatsheet”

<https://www.utc.fr/~jlaforet/Suppl/python-cheatsheets.pdf>

Installation of Python on Windows

- Go to the **miniforge** releases page on GitHub: <https://github.com/conda-forge/miniforge#miniforge>
- **Download** the installer for Windows (MiniForge-<version>-Windows-x86_64.exe)
(change browser in case you get problems)
- Locate the downloaded installer and double-click to **run**.
- Follow the prompts to complete the installation. Choose **default installation** options.
- You should now have a “**Miniforge Prompt**.”
Type miniforge in windows search bar to find it:
- **Change directory** to Z: (no write permission in C:)
- Create your own **Python environment**:

```
mamba create -y -n myenv python=3.11
```



Starting Interactive Python Session

- **Activate the environment:**

```
mamba activate myenv
```

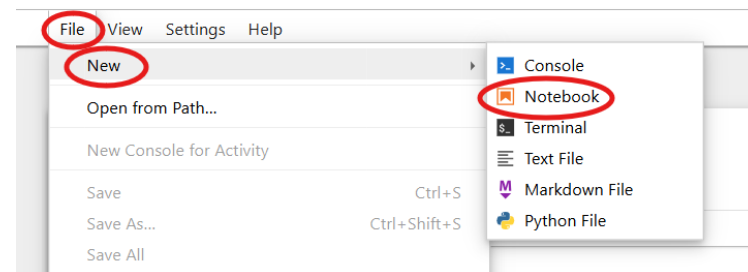
- **Install the package `jupyter`:**

```
mamba install -y jupyter notebook
```

- **Start jupyter notebook server:** `jupyter notebook`

- **Go to:** File → New → Notebook

- **Select kernel:** "Python3"



- **Within the notebook cell, type:** `print("Hello!")` **and press shift+return**

Jupyter Notebook / Lab

Why Jupyter?

- **Interactivity**
- Prototyping & fast iterations
- Visualization
- Can be used for “computer experiments”
- **Documentation** via Markdown

When not to use:

- Do not use them for large programs and when building libraries → IDE

Jupyter notebooks

- **Cells**
 - Code Cells: interactive programming – run `shift+enter` for code execution
 - Code is executed one cell after another
 - Markdown Cells: documentation & formulaes
- **Kernel**
 - Computational engine: Mostly python, but also other languages possible
 - Restart the kernel to delete all variables
- Special (magic) commands: `%time` or e.g. `!ls` for command line options
- Autocompletion: use `Tab` to get suggestions for functions!
- Widgets: Build simple graphical user interfaces

Integrated Development Environment (IDE)

- Use a IDE for more heavy-weight programming
- An IDE is much more than a text-editor:
 - autocompletion, code highlighting, debugging, searching,code navigating, renaming, refactoring, testing ...
- Important IDEs for Python:
 - VSCode: Very general & powerful – most used IDE
 - Pycharm: Customized for Python and data science apps
 - Thonny: Minimalistic – for beginners

Installation of IDE (thonny)

- Go to <https://thonny.org/>
- Download portable variant python 3.10 64bit
- Extract file in Downloads folder
- Copy directory to Z:
- Start IDE by clicking on thonny.exe

Additional Material

Technical Setup - Fallback

- Sharing notebooks via binder: <https://ovh2.mybinder.org/>

Technical links

- Setup jupyter: Working with **jupyter notebooks** for programming: <https://docs.jupyter.org/en/latest/running.html>
- setup binder: <https://jupyter.org/binder>
- setup colab: https://course19.fast.ai/start_colab.html
- Working with **github** for versioning control: <https://docs.github.com/en/get-started/start-your-journey/hello-world>
- Working with **jupyter notebooks** for programming: <https://docs.jupyter.org/en/latest/running.html>
- How to open github notebook with colab documentation: Open github notebook with colab
- Open Notebook in browser from github or google drive:
- https://colab.research.google.com/github/chrissly31415/lecture_cheminformatics
- Open master branch: https://colab.research.google.com/github/chrissly31415/lecture_cheminformatics/blob/master
-