SciPro - Setup Instructions

Table of Contents

Installing Python3 via Pyenv

Pyenv lets you easily switch between multiple versions of Python. It's simple, unobtrusive, and follows the UNIX tradition of single-purpose tools that do one thing well. [Source](#)

MacOS / Linux

Step 1:

Getting pyenv via the automatic installer:

```
$ curl https://pyenv.run | bash
```

Step 2:

Now set up your shell environment for Pyenv by copy and pasting the following commands:

For Bash:

```
echo 'export PYENV_ROOT="$HOME/.pyenv"' >> ~/.bashrc
echo 'command -v pyenv >/dev/null || export PATH="$PYENV_ROOT/bin:$PATH"' >> ~/.bashrc
echo 'eval "$(pyenv init -)"' >> ~/.bashrc
```

For Zsh:

```
echo 'export PYENV_ROOT="$HOME/.pyenv"' >> ~/.zshrc
echo 'command -v pyenv >/dev/null || export PATH="$PYENV_ROOT/bin:$PATH"' >> ~/.zshrc
echo 'eval "$(pyenv init -)"' >> ~/.zshrc
```

Step 3:

Restart your shell by using the appropriate command for your type of shell:

```
$ source ~/.bashrc   # BASH
$ source ~/.zshrc    # ZSH
```

Step 4:

Install Python build dependencies

On MacOS:

```
brew install openssl readline sqlite3 xz zlib tcl-tk
```

If the brew command is not found, make sure to install [Homebrew](#) first. It's as simple as copy and pasting a single command.

On Linux:

```
sudo apt-get update -y && sudo apt-get install -y build-essential libssl-dev zlib1g-dev \
libbz2-dev libreadline-dev libsqlite3-dev curl \
libncursesw5-dev xz-utils tk-dev libxml2-dev libxmlsec1-dev libffi-dev liblzma-dev
```

Step 5:

Listing available Python versions (MacOS / Linux)

```
$ pyenv install -l
```

Step 6:

Installing Python 3.11.5 (MacOS / Linux)

```
$ pyenv install 3.11.5
```

Step 7:

After Python 3.11.5 is installed you have to set it as the default python version for your system by running the following command:

```
$ pyenv global 3.11.5
```

Step 8:

You can check the installed versions by running `pyenv versions`:

```
$ pyenv versions
  system
* 3.11.5 (set by [...]/.pyenv/versions)
```

Thats it! 🎉

Bonus (Best practice)

It is recommended to create a virtual environment for each new Python project. This way you can manage your projects dependencies more easily.

For this class I recommend creating a "SciPro" environment that you will youse throughout the course.

Creating a virtual environment

```
$ pyenv virtualenv 3.11.5 SciPro
```

This creates a new virtual environment called "**SciPro**" running Python **3.11.5**

I also recommend you creating a dedicated directory for this course where you keep all of your future assignment organised in subdirectories.

To automatically assign the **SciPro** environment to the dedicated "SciPro" directory, you have to navigate to that directory using your Terminal and run:

```
$ pyenv local SciPro
```

This assigns the previously created virtual environment called "**SciPro**" to the current working directory.

Now every time you navigate inside that directory, Pyenv automatically switches to the assigned environment.

Thats pretty cool if you ask me!

Installing Python3 from Python.org

Step 1: Getting Python (All Operating Systems)

Visit python.org/downloads using a web browser of your choice.

Go ahead and download the latest version of Python3 for your target operating system.
At the time of writing, the latest version is ***Python 3.11.5***.

Step 2: Installing Python (Windows)

1. Double-click the installer you just downloaded
2. Make sure to check "***Add python.exe to PATH***"
3. Click "***Install Now***" and confirm the installation prompt.
4. Once you see the "***Setup was successful***" message, make sure to click "***Disable path length limit***" and confirm any prompts.

Background Info:

> "Disabling the path limit length is recommended after Python setup is successful, because if python was installed in a directory with a path length greater than 260 characters, adding it to the path could fail. So don't worry about that action and proceed to it."
> Source: https://answers.microsoft.com/en-us/windows/forum/all/path-length-limit/412d48cd-7ddf-435a-9935-cf51af16f8a4
> (Last visited: 9/30/2023)

6. You are all set and can close the installer. 🎉

To check if python was installed properly, open the "***Command Prompt***" (cmd.exe) and type in `python --version` and hit enter. If the installed version of python is shown like below, everything went fine.

```
C:\Users\username>python --version
Python 3.11.5

C:\Users\username>
```

Step 2: Installing Python (MacOS)

1. Double-click the installer you just downloaded.
2. Click "***Continue***" (Introduction)
3. Click "***Continue***" (Read Me)
4. Click "***Continue***" (License)
5. Agree to the software license agreement
6. Now click "***Install***"
7. You might get prompted to enter an administrator username and password (do so and click "***Install Software***")

8. The directory with all the bundled python tools will open so you can see what was just installed.
9. You are all set and can close the installer. 🎉

To check if python was installed properly, open up a **Terminal** (Terminal.app) and type in `python3 --version` and hit enter. If the installed version of python is shown like below, everything went fine.

```
user@host ~ % python3 --version
Python 3.11.5
user@host ~ %
```

Step 2: Installing Python (Linux)

1. Install development packages required to build Python using the following command.

```
$ sudo apt-get update -y && sudo apt-get install -y build-essential zlib1g-dev \ libncurses5-dev
libgdbm-dev libnss3-dev \ libssl-dev libreadline-dev libffi-dev curl
```

2. Extract the previously downloaded tarball using:

```
$ tar -xf Python-3.11.5.tar.xz
```

3. Configure the script. (This may take some time.)

```
$ cd Python-3.* && ./configure
```

4. Start the build process. (This will take some time.)

```
$ sudo make install
```

5. Verify the installation

```
$ python3 --version
```

or

```
$ python --version
```

Thats it! 🎉

Installing Jupyter Notebook

This section requires Python already being installed.

Windows / MacOS / Linux

Python comes with "**pip**" (Pythons package installer). You can use **pip** to install packages from the Python Package Index and other indexes.

To install Jupyter Notebook, simply run the following command in your Terminal / Command Prompt

```
$ pip install notebook
```

Thats it! 🎉

Running Jupyter Notebook

On all systems: navigate to your dedicated SciPro directory using a Terminal / Command Prompt and run the following command inside that directory:

```
$ jupyter-notebook
```

This will automatically startup a local Jupyter Notebook Server. You can access it using one of the links printed to your Terminal / Command Prompt.

Thats it! 🎉

Created by Nicolai Kürtös