

Computergestütztes Programmieren mit PsychoPy: Installation guide

Lucca Zachmann | lucca.zachmann@unifr.ch

March 2020

Introduction

Depending on your operating system, you need to install various components to be able to program in Python and build your own experiment with PsychoPy.

Please keep in mind that the installation and setup of software are sometimes harder and much more poorly documented than mere usage. Moreover, there are many different ways to achieve the installation. Thus, this guide aims at easing the *proper* set up of the development environment for **macOS Catalina** and **Windows 10**. Specifically, the proposed installation reconciles a cross-platform usage of the tools with relative simplicity (e.g., no virtual environments). Make sure that you come to class with one of the above mentioned operating systems set up on your machine. In case your machine does not meet said requirements, borrow a computer from friends or family.

The easiest way to install Python 3 is via the Anaconda platform, which already includes standard packages and also comes with handy applications such as Jupyter Notebook and Spyder. Spyder is a cross-platform integrated development environment (IDE) for the Python language similar to what RStudio is for the programming language R.

This guide provides instructions to install the following components:

- Python 3 with Anaconda Platform
- PsychoPy package (and potentially others...)

Installation Guide for macOS Catalina

The macOS installation is a bit more involved, as Apple provides Python 2 by default on its machines. However, we want to make use of Python 3, a more stable and up-to-date version of the programming language. We need to inform our system about this preference.

Install Python 3 with the Anaconda Platform

1. Download the 64-bit command-line installer for Python 3.7 for macOS from here:

https://repo.anaconda.com/archive/Anaconda3-2020.02-MacOSX-x86_64.sh.

2. Open a Terminal to get a command-line interface. In case you cannot find the application in your system tray, press the **command** and **spacebar** keys and type **Terminal** to search for it. Type the following command on a new line and hit **enter**:

```
bash ~/Downloads/Anaconda3-2020.02-MacOSX-x86_64.sh
```

NOTE: The computer will not reply to a successfully executed command in the terminal. Instead, it will open a new line with **(base) <username>@<computername> ~ %** indicating that the system is ready to take the next command. If errors occur, the system will most definitely let you know.

3. In the subsequent installation dialogue, you follow the suggestions by pressing **enter** multiple times.

IMPORTANT: On the question **Do you wish the installer to initialize Anaconda3 by running conda init?**, you need to type in: **no**

4. To activate the environment, you enter in the Terminal window:

```
/Users/<Your username>/anaconda3/bin/activate
```

NOTE: You can ask for your username typing **whoami** in the Terminal.

5. Then initialize the change with:

```
conda init zsh
```

6. After the initialization, you close and reopen the terminal to activate the modification.

7. Finally, you can check the installation by entering **conda list** in the terminal. If you see a list of modules, you have successfully installed the Anaconda environment. You can start to program in Python.

8. *Additional remark. To enable community-driven packages within the Anaconda distribution, you need to add a new channel (This takes some time):

```
conda config --append channels conda-forge
```

Sources: Towards Data Science, conda-forge

Install PsychoPy

PsychoPy requires dependencies that require the macOS command line tools. Consequently, we install these tools from the terminal:

```
xcode-select --install
```

Use the Python Package Index pip to install PsychoPy or any other package (recommended).

```
/Users/<Your usermae>/anaconda3/bin/python -m pip install <package name>
```

Alternatively, install PsychoPy or any other package that you might want to use with the Anaconda Package Manager conda (requires 8. from above).

```
conda install -c conda-forge psychopy
```

Installation Guide for Windows 10

Install Python 3 with Anaconda Platform

1. We download the 64-bit graphical installer for Python 3.7 for Windows from here:

```
https://repo.anaconda.com/archive/Anaconda3-2020.02-Windows-x86\_64.exe.
```

2. After the download is complete, you can execute the downloaded file to start the installation process.

IMPORTANT: Under Advanced Options you should tick the checkbox ‘Add Anaconda to my PATH environment variable’. Otherwise you can accept the recommended options.

3. *Additional remark. To enable community-driven packages within Anaconda distribution, you also need to add a new channel. Open an Anaconda Prompt from your application list and execute:

```
conda config --append channels conda-forge
```

This can take a few minutes.

Source: Anaconda, conda-forge

Install Python packages

Using the same Anaconda Prompt as before, you can pip install packages into Anaconda (recommended):

```
pip install psychopy
```

Alternatively, using the Anaconda Prompt, you can install the packages with the Anaconda Package Manager conda (requires 3. from above).

```
conda install -c conda-forge psychopy
```

First Steps in Python

Open Spyder, an editor for Python, that you have installed together with Anaconda. As a kind of initiation ritual, say hello to the world in Python by issuing the following code lines:

```
print("Hello, World!")
```

Congrats, you wrote your first little program in Python. It may not be as impressive, but hang on, it soon will be! The list of tutorials below provides a great starting point to learn the basics of Python by solving little exercises interactively.

- [LearnPython](#)
- [Python Principles](#)

*Extension for macOS: Homebrew via Xcode

This is not required for this course! Still, Homebrew is a handy package manager that you might want to use down the road in your Python journey. Xcode is an integrated development environment (IDE) comprising various software development tools for macOS. Among other things, the installation of Xcode is a requirement for the subsequent installation of the package manager Homebrew. To check if you have Xcode already installed, you need the command-line.

1. Open a Terminal to get a command-line interface. When you cannot find the application in your system tray, press the **command** and **spacebar** keys to search and type **Terminal** to search for it.

2. To check if you have already installed Xcode, type your in the Terminal window:

```
xcode-select -p
```

3. If you receive the following output, then Xcode is installed:
`/Applications/Xcode.app/Contents/Developer`. If you see an error, then install Xcode from the App Store via your web browser and accept the default options. The application is large (~10GB).

4. Once Xcode is installed, return to your Terminal window. Next, you'll need to install Xcode's separate Command Line Tools app, which you can do by typing:

```
xcode-select --install
```

5. At this point, Xcode and its Command Line Tools app are fully installed, and we are ready to install the package manager Homebrew.

Install Package Manager Homebrew

1. To install Homebrew, type this into your Terminal window:

```
/usr/bin/ruby -e "$(curl -fsSL \  
https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

2. You can make sure that Homebrew was successfully installed by typing:

```
brew doctor
```

3. To ensure that your installation of Homebrew is up to date, run:

```
brew update
```

4. It is not needed for now, yet you can upgrade outdated packages altogether:

```
brew upgrade
```

Source: Homebrew, Digital Ocean