

How To: Terminal, Conda Environments and Running Python Files

1 Terminal and Anaconda Prompt

For those using **Mac**, we will be using **terminal**. You can open terminal by pressing *command + space bar* and searching for “terminal”, then pressing enter. Alternatively, click on the little magnifying glass in the top right of your monitor and open the search bar that way. For those using **Windows**, we will be using **Anaconda Prompt**. This should be automatically installed when you download Anaconda. You can press *windows + s* to search, or click on the little magnifying glass. Search for “Anaconda Prompt” and press enter. **Note:** Everything below is written with Mac users in mind. Windows users should assume identical commands as used in terminal, unless otherwise specified.

2 Navigating in terminal

Files are stored on your computer at particular locations. For example, if you have a python file called “main.py” saved in a desktop folder called “CNRI”, the location of that file is likely “/Users/yourusername/Desktop/CNRI/main.py”. Terminal can only access one location at a time, and it will clearly tell you where you are currently. You can view the files in the current location terminal is accessing by typing ‘ls’ and pressing enter. Try it now. Note: windows uses ‘dir’ instead of ‘ls’.

```
(base) [yourusername] ~ $ ls
```

You can navigate in terminal by using the ‘cd’ command. E.g., to access the files saved on your desktop, simply cd into Desktop.

```
(base) [yourusername] ~ $ cd Desktop/
```

You will immediately see terminal update its current location.

```
(base) [yourusername] ~/Desktop $
```

and using ‘ls’ here would show you all the files on your desktop instead. To cd into our hypothetical CNRI folder and run our python file there, we just do this

```
(base) [yourusername] ~/Desktop $ cd CNRI/
(base) [yourusername] ~/Desktop/CNRI $ python main.py
```

To go backwards one directory, you cd into ‘../’.

```
(base) [yourusername] ~/Desktop/CNRI $ cd ../
(base) [yourusername] ~/Desktop $
```

Lastly, terminal supports auto-completion of folders/files, as long as there aren’t multiple files with the letters already typed. So, you can type ‘cd Desk’ and press **tab**, and terminal should autocomplete to ‘cd Desktop/’, unless there is a different folder called ‘DeskSupplies’, since then terminal wouldn’t know which to autocomplete to. Always use ‘ls’ to see which folders/files are available at your currently location!

3 Conda Environments

First, check that conda is working, by checking its version. If not, you’ll need to go install anaconda online.

```
(base) [yourusername] ~ $ conda --version
conda 4.12.0
```

The package psychopy we use for this course requires python 3.8. However, it is almost guaranteed that your default installed python is different than version 3.8. To check the version of python, run the same command.

```
(base) [yourusername] ~ $ python --version
Python 3.11.4
```

How would we get psychopy working? One option is to completely delete your current version of python and download 3.8. But what if next week you need to use a different package, that only supports python 3.7? And the next week you want to use the most updated version of python because of a new feature? You can clearly see that this is not sustainable, because you would be breaking all your old code every time you changed python versions.

The solution is to create a “virtual environment” that contains a certain version of python. You can activate that environment when you need python 3.8, and then deactivate it to go back to your regular (default or base) environment. For this we can use something called a package manager, of which Anaconda is a famous one, and we’ll use a conda environment. Create a conda environment called “psychopy” with the following code, and notice how we request that python’s version be 3.8.

```
(base) [yourusername] ~ $ conda create —name psychopy python=3.8
```

Having created the environment, we now activate it. Notice how “(base)” changes to “(psychopy)”. When we check the version of python now, we should see that it is equal to 3.8.

```
(base) [yourusername] ~ $ python —version
Python 3.11.4
(base) [yourusername] ~ $ conda activate psychopy
(psychopy) [yourusername] ~ $ python —version
Python 3.8.16
```

4 Using psychopy

Unlike some common packages (numpy, random, etc), some packages need to first be installed into our conda environment before they can be used. The most common installer is called “pip”, which stands for (humorously given the recursion) “pip installs packages”. With our psychopy environment active, we can now install psychopy.

```
(psychopy) [yourusername] ~ $ pip install psychopy
```

Hopefully this works without any errors. Once finished installing, you can check which packages are installed in the current active environment by running

```
(psychopy) [yourusername] ~ $ conda list
```

Now, any python file that is run (using ‘python filename.py’) while the psychopy environment is active will be able to import and use the psychopy package.