Setting up your Development Environment

Alex Flückiger

11 März 2021

Contents

1 Introduction

This guide aims to ease a correct set up of the development environment for Windows 10 and macOS. Depending on your operating system, you have to install additional components to be able to program in Python and perform common data wrangling tasks in the command-line. Moreover, the proposed installation strikes a good balance between cross-platform usage of the tools and relative simplicity (e.g., no virtual environments in Python). The instructions cover the installation of the following components:

- Python 3.8 with Anaconda Platform (programming language)
- Tesseract (Optical Character Recognition)
- various Bash tools
- Text editor Atom

Be aware that the installation and setup of software is sometimes more difficult and poorer documented than its usage. Beyond potential problems during the installation, there are also different ways to set up the development environment.

The easiest way to install Python 3 is via the Anaconda platform, which provides common packages used in Data Science out of the box. We write Python code using the JupyterLab[1] which runs as a web-application in any browser. It is a lightweight editor commonly used in research as it allows for an immediate execution of code and showing the output next to the code that created it.

Before you proceed with the installation, back up your files[2] and make sure that your computer fulfill the following two requirements:

- run the most current version of your operating system (Win10, macOS). If not, update your system.
- has at least 15 GB free disk space.

Keep in mind that there is *no* feedback in the command-line unless there is an issue. At first, the lack of feedback after a successful action may be confusing as it counters the general experience when working with graphical interfaces. Nevertheless, the lack of feedback just means the command was executed accordingly; thus, there is no need to bother you with further messages.

Let me know if you are struggling with any of the steps.

[1] JupyterLab is the success of the well-established Jupyter Notebook providing additional features

[2] "Better save than sorry." Backing up files is a must and may save you tears as a computer is a delicate piece of hardware that does not live forever and sometimes its user are messing up files or software. In these cases, you just want go back one step.

2 Installation Guide for macOS

2.1 Text Editor

- When you are still using the standard text editor on your system (or none editor at all), install the `Atom` editor. Download and install the installation files from the official website https://atom.io/.

2.2 Install Development Environment Xcode

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 installed Xcode already, type your in the Terminal window:

   ```
   xcode-select -p
   ```

3. If you receive the following output, then Xcode is installed: `/Library/Developer/CommandLineTools`. If you see an error, then install Xcode from the App Store via your web browser and accept the default options.

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.

Source: Digital Ocean

2.3 Install Package Manager Homebrew

1. To install Homebrew, type the following command into your Terminal window.

   ```
   /bin/bash -c "$(curl -fsSL \
   https://raw.githubusercontent.com/Homebrew/install/master/install.sh)"
   ```

   When there is an issue executing this command (i.e., unprintable characters), copy the identical looking installation command from the offical website into your Terminal.

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. Although it is not needed for now, you can upgrade outdated packages altogether with:

   ```
   brew upgrade
   ```

Source: Homebrew

2.4 Install Anaconda Platform (Python 3)

Please note that the following instructions are for macOS Big Sur `11.x`. Update your system if you still run macOS Catalina `10.15`.

1. Download the 64-bit command-line installer for Python 3.8 for macOS from here.

2. Open a Terminal window (command-line). The following command requires that the installer is located in the folder `Downloads` and has its original name. If you have changed its location or name, then adapt the file path in the following command before execution:

   ```
   bash ~/Downloads/Anaconda3-2020.11-MacOSX-x86_64.sh
   ```

3. In the subsequent installation dialogue, you can follow the suggestions by pressing `Enter` multiple times.

4. The installer may prompt `Do you wish the installer to initialize Anaconda3 by running conda init?`. Accept by typing `yes`.

5. After the initialization, you can close and reopen the Terminal to activate the modification.

6. 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 are ready to run Python programs.

7. To enable community-driven packages within Anaconda distribution, you also need to add a new channel:

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

Source: Anaconda, conda-forge

2.5 Install command-line Tools

2.5.1 Tesseract

1. Install the text recognition engine Tesseract, which allows extracting text from images, with:

   ```
   brew install tesseract
   ```

2. Install the various language models for Tesseract with:

   ```
   brew install tesseract-lang
   ```

Source: Tesseract

2.5.2 wget

1. Install the tool wget that allows you to retrieve content from web servers via the command-line with:

   ```
   brew install wget
   ```

2.6 Install Kite

Install Kite from the official website. Kite assists you with smart recommendations while programming.

3 Installation Guide for Windows

3.1 Text Editor

- When you are still using the standard text editor on your system (or none editor at all), install the `Atom` editor. Download and install the installation files from the official website https://atom.io/.

3.2 Install Ubuntu in a Windows Subsystem

To use the powerful bash tools on Windows, we install Linux Ubuntu system within the Windows environment using WSL1.

1. Before installing any Linux distribution, you must ensure that the “Windows Subsystem for Linux” optional feature is enabled. Open `PowerShell` as administrator by right-clicking on the application icon in the start menu and run the following command in the shell:

```
dism.exe /online /enable-feature `
/featurename:Microsoft-Windows-Subsystem-Linux /all /norestart
```

2. Restart your computer after completing step above.

3. Install `Linux Ubuntu 20.04 LTS` via the Microsoft Store.

4. On the distribution’s page, select `Get`. You can safely decline to create an account when you are asked in the confirmation dialogue. Ubuntu will be downloaded in the background. You can check the progress after clicking the corresponding icon located at the top-right.

5. After downloading, you can launch Ubuntu by clicking the `Launch` button in the Microsoft Store app, or launching the Ubuntu from the start menu.

6. The first time you launch a newly installed Linux distribution, a console window will open and you will be asked to wait for a minute or two for files to decompress and be stored on your PC. All future launches should take less than a second.

7. Once the installation is complete, you will be prompted to create a new user account (and its password).
   IMPORTANT: Remember these credentials as they are used to switch to the administrator mode on your Linux system. You may choose the same account name and password as in your hosting Windows system.

8. Update your freshly installed Ubuntu system:

   ```
   sudo apt update && sudo apt upgrade
   ```

9. Create a symbolic link in the Bash to easily access your files on Windows. For example, you may want to link the folder `Documents` on Windows to the symbolic folder `documents` on Ubuntu. To do this, replace the `<YOUR_USERNAME>` with the actual one and run the following commands:

   ```
   cd ~
   ln -s /mnt/c/Users/<YOUR_WINDOWS_USERNAME>/Documents documents
   ```

10. Annoyingly, the copy/paste behavior is different in command lines on Windows. Any selected text is copied automatically and to paste it, you have to right-click on your mouse. You may want to reassign the shortcuts to `Ctrl+Shift+C` and `Ctrl+Shift+V` in the menu (`right-click` on the windows title bar → `Properties` → `Options`). Using `Ctrl+C` is not possible as it is used to cancel a running program.

Source: Microsoft

3.3 Install Anaconda Platform (Python 3) within Windows Subsystem

For the best user experience, we are installing Anaconda on the Ubuntu Subsystem on Windows. This comes with the advantage that you can use Bash as a single entry point for any application using Python or the command line.

1. Open the Ubuntu that you have installed in the previous step.

2. Download the Anaconda Platform installer for Linux using the command line:

```
wget https://repo.anaconda.com/archive/Anaconda3-2020.11-Linux-x86_64.sh
```

3. Run the installation script:

   ```
   bash Anaconda3-2020.11-Linux-x86_64.sh
   ```

4. Scroll down the license agreement by pressing the space bar on your keyboard and follow the prompts to accept.

5. When being asked if you'd like the installer to prepend it to the path, say yes.

6. Initialize conda when being prompted to finalize the installation

7. Close the window and reopen the Ubuntu Bash to reload .bash configs.

8. To test that Anaconda is set up correctly worked, run the command below. It should print a path that has `anaconda` in it (e.g., `/home/<YOUR_USERNAME>/anaconda3/bin/python`). If it doesn't have `anaconda` in the path, do the next step. Otherwise, the installation was successful.

   ```
   which python
   ```

9. **IMPORTANT**: Proceed only if indicated at the previous step. Add manually the Anaconda `bin` folder to your `PATH` variable. To do this, replace the `<YOUR_USERNAME>` with the actual one and run the following command:

   ```
   echo "export PATH=/home/<YOUR_USERNAME>/anaconda3/bin:$PATH" >> ~/.bashrc
   ```

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

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

Source: Anaconda, conda-forge

3.4 Install command-line tools

- Open Ubuntu and install essential tools via the command-line:

  ```
  sudo apt-get install build-essential
  ```

3.4.1 Tesseract

1. Install the text recognition engine Tesseract, which allows extracting text from images, with:

   ```
   sudo apt install tesseract-ocr
   ```

2. Install the German language model for Tesseract with:

```
    sudo apt install tesseract-ocr-deu
```

Source: Tesseract

3.5 Install Kite

Install Kite that assist you with smart recommendations when programming.

```
bash -c "$(wget -q -O - https://linux.kite.com/dls/linux/current)"
```

4 Installation Guide for Linux

You presumably have installed some of the required tools already or you have the necessary knowledge to do so. Otherwise, ask me anytime or just Google for a manual. Anyways, you are lucky as it is simpler than on the other platforms. Linux rules them all when it comes to development.

4.1 Install Kite

Even when you are a pro, Kite is worth a shot. It assists you with smart recommendations while programming. Install Kite before you proceed with the setup of JupyterLab as described in the next chapter.

```
bash -c "$(wget -q -O - https://linux.kite.com/dls/linux/current)"
```

5 Install Python packages (all Systems)

In this step, you install a few additional Python packages using the Anaconda Package Manager conda (alternative to pip). Confirm when you are asked to install the packages.

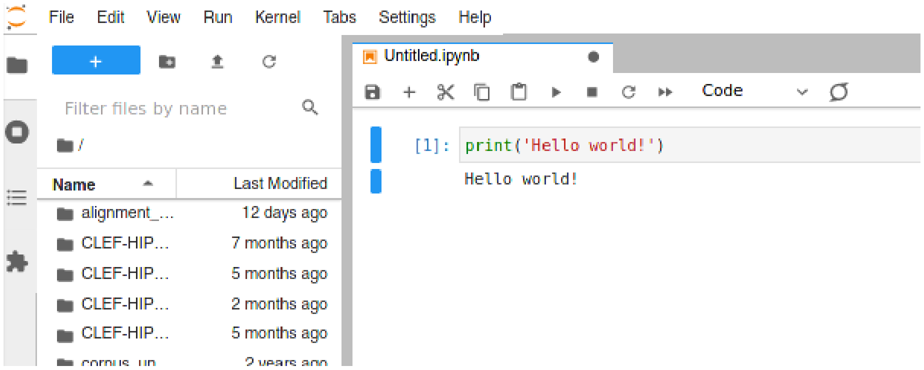
```
conda install flake8
conda install -c anaconda nodejs
conda install -c anaconda spacy
conda install -c conda-forge textacy
conda install -c conda-forge plotnine
```

Additionally, we install some extensions for the JupyterLab that make programming easier. If you run into issues, skip these steps and let me know:

```
# set up Kite for jupyterlab
pip install "jupyter-kite<2.0.0"
jupyter labextension install "@kiteco/jupyterlab-kite"

# Flake8
jupyter labextension install jupyterlab-flake8
```

6 First Steps in Python (all Systems)



Name	Last Modified
alignment_...	12 days ago
CLEF-HIP...	7 months ago
CLEF-HIP...	5 months ago
CLEF-HIP...	2 months ago
CLEF-HIP...	5 months ago
corpus_un	2 years ago

```
[1]: print('Hello world!')
     Hello world!
```

Figure 1: Write your first Python script within JupyterLab

Start the JupyterLab, an editor for Python, which you have installed together with Anaconda. For this, open your command line and run the following command:

```
jupyter lab --no-browser
```

Copy the the provided link and paste it into your browser to enter the web-application. Mac and Linux User can run the command above without the `--no-browser` flag get redirected to the appropriate browser window directly. Example:

```
http://localhost:8888/lab?token=bf39883d2d5b637352e1cf56656c4a59f
```

As a kind of initiation ritual, say hello to the world in Python by issuing the following code lines and press the `Enter` button:

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

Congrats, you wrote your first little program in Python. It may not be as impressive, but you can go along and will learn by practicing. The list of tutorials below provides a great starting point to learn the basics of Python by solving little exercises interactively.

- Python Principles
- LearnPython