Task 0.- Tools installation

1.- Installation of the working environment

To be able to run the notebooks of this course on your personal computers, there are three options. You could use "Google colab" (Google server), "JupyterHub" (course server) or "**JupyterLab**" (personal server installed on your personal computers).

This guide indicates the steps to follow to install Miniconda, Python, "JupiterLab" and various Python modules on your personal computers. The installation time is approximately 60 minutes.

These steps have been successfully tested with the following operating systems:

- Windows 10 Home, Windows 10 pro, Linux kubuntu 18.04, …

1.1.- Installation of Miniconda together with Python 3

Python 3: Python is an interpreted programming language whose philosophy emphasizes the readability of its code. It is a multiplatform and multiparadigm programming language, since it supports object orientation, imperative programming and functional programming.

Miniconda: installs Python and allows you to easily install/uninstall most of the modules available in Python.

See the link below, to see the available Miniconda and Python versions:

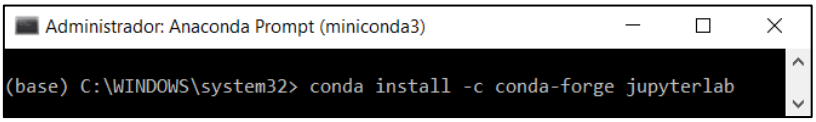
https://docs.conda.io/en/latest/miniconda.html

Download the appropriate Miniconda version according to your operating system, with Python version 3.8 (or later), and then run the downloaded file with administrator permissions accepting all default options.

1.2.- Installation of "JupyterLab"

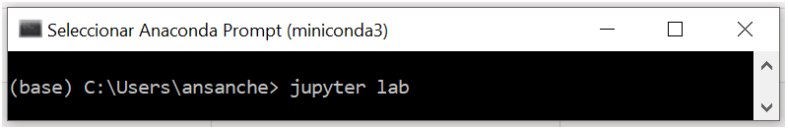
"JupyterLab" is installed as a module with Miniconda. To do this, run the "Anaconda Prompt (miniconda 3)" terminal as an administrator. This terminal can be found in the home search engine. Then, run the following command (continue the execution by typing the keys `y` and `enter`, when you are asked):

```
(base) C:\WINDOWS\system32> conda install -c conda-forge jupyterlab
```

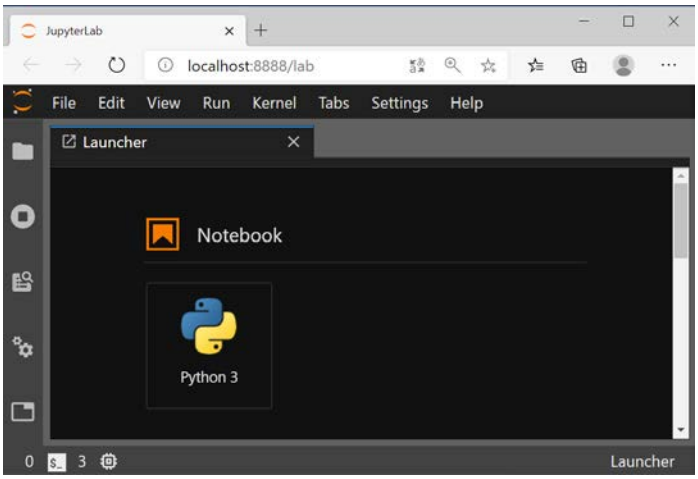


When the installation is complete, a local server can be launched by running the following command over an "Anaconda Prompt (miniconda3)" terminal:

```
(base) C:\Users\username> jupyter lab
```



The server will ask you which browser to use. Select a browser, and a tab will automatically open in the browser with the work environment ready to start programming a notebook in Python. In case the browser does not open automatically, you can open one manually and navigate to the url http://localhost:8888/lab.



Note that to close the "JupyterLab" server, you will need to type `control+C` over the terminal from which it was launched.

Finally, it is only necessary to explain how to install the external modules that require the notebooks of the course and how to change the default work folder of "JupyterLab".

Using the command `jupyter lab --version` you can view the installed version.

1.3.- Installation of some Python modules

The process of installing modules with Miniconda is always the same. An "Anaconda Prompt (miniconda 3)" terminal must be opened and after that the appropriate `conda` command executed.

Install the following basic modules by running these commands:

(base) C:\Users\system32> conda install -c conda-forge matplotlib
(base) C:\Users\system32> conda install numpy
(base) C:\Users\system32> conda install scipy
(base) C:\Users\system32> conda install scikit-learn
(base) C:\Users\system32> conda install pandas
(base) C:\Users\system32> conda install -c anaconda pillow
(base) C:\Users\system32> conda install scikit-image

Install OpenCV, running the command:

(base) C:\Users\ system32> conda install -c conda-forge opencv

Install Pytorch (https://pytorch.org/), running:

➔ in case of Windows o Linux:

(base) C:\Users\system32> conda install pytorch torchvision cpuonly -c pytorch

➔ In case of Mac:

(base) C:\Users\system32> conda install pytorch torchvision -c pytorch

Install ipywidgets, running the command:

➔ in case of JupyterLab 3:

(base) C:\Users\system32> conda install -c conda-forge ipywidgets

Install open3d, running the command:

(base) C:\Users\system32> conda install -c open3d-admin -c conda-forge open3d

Install TKinter, running the command:

(base) C:\Users\system32> conda install -c conda-forge tk

The command conda list displays the installed packages.

1.4.- Change the JupyterLab working folder

"JupyterLab" defines a default working folder, where all notebooks are expected to be saved. In my case this folder is "C:\Users\ansanche", which matches the directory from where it was installed.

If you want to change this working directory by default, you can follow the following procedure:

1.- Create the configuration file, executing the following command:

(base) C:\Users\username> jupyter notebook --generate-config

This command generates the following file:

`C:\Users\username\.jupyter\jupyter_notebook_config.py`

2.- Open this file with a text editor and search the definition of the following variable:

`# c.NotebookApp.notebook_dir = ''`

3.- Define the new working directory (remember to delete the initial character `#`) :

`c.NotebookApp.notebook_dir = "C:\\Visión"`

In Windows, to define a correct "path", it is recommended to use the double quotation marks `""` (this will accept the spaces) and the separator between double backslash directories `\\` (this avoids possible problems with control characters).

4.- In the JupyterLab working folder, create the folders "imgs" to store all the images, "videos" to store all the videos and "datasets" for all the datasets.

5.- Copy the original notebooks (*.ipynb) and possible Python files (*.py) from the PoliformaT folder "Resources/...", to the "JupyterLab" working folder of your computer (with all the images, videos and datasets in their respective folders).

6.- Run the local server "JupyterLab" and... That's it...

Note: If during the execution of a notebook, a module cannot be imported, this module should preferably be installed with a `conda` command (or as an alternative with a `pip` command).

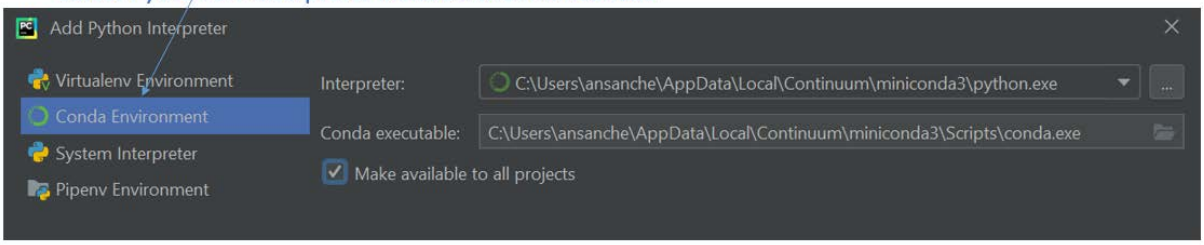
2.- Installation of the PyCharm integrated development environment

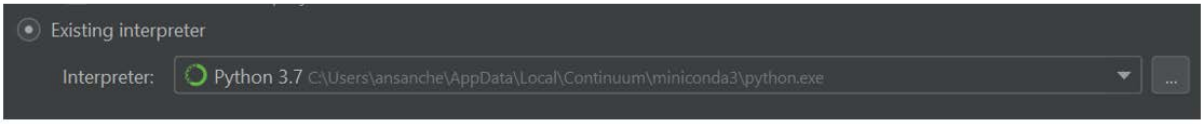
Download and install the "Community" version of PyCharm from the link:

https://www.jetbrains.com/es-es/pycharm/download/#section=windows

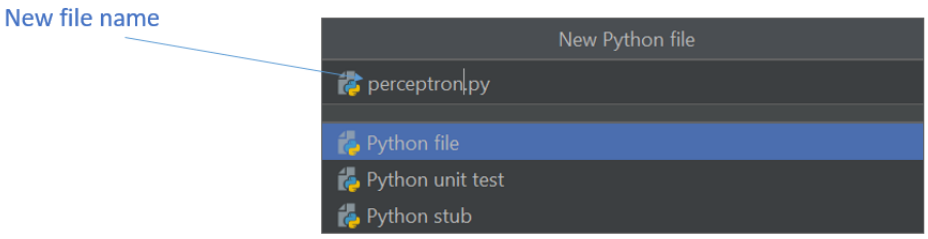
2.1.- Create a new project

Select Python 3.7 interpreter installed with miniconda3





- File -> New -> Python File



- Docs:
  - Mastering PyCharm : use PyCharm with fluid efficiency, Islam, Quazi Nafiul, author. Birmingham : Packt Publishing, 2015.

3.- Installation of plotly and Node.js

Run the "anaconda prompt (miniconda 3)" terminal as an administrator.

Install plotly, running the command:

(base) C:\Users\system32> conda install -c plotly plotly

Install Node.js:

Download and install the Node.js version from the link: https://nodejs.org/en/

Run the "anaconda prompt (miniconda 3)" terminal as an administrator.

(base) C:\WINDOWS\system32> jupyter labextension install jupyterlab-plotly@4.14.3
(base) C:\WINDOWS\system32> jupyter labextension install @jupyter-widgets/jupyterlab-manager plotlywidget@4.14.3