SLab Windows Installation

This document describes how to install the SLab system in a Windows PC. The installation have been tested on Windows 10 but it should be similar in any other Windows version.

V1.0(18/5/2019) License information is at the end of the document

In order to have use the **SLab system**, you need a **Python 3.x** environment that includes the required modules:

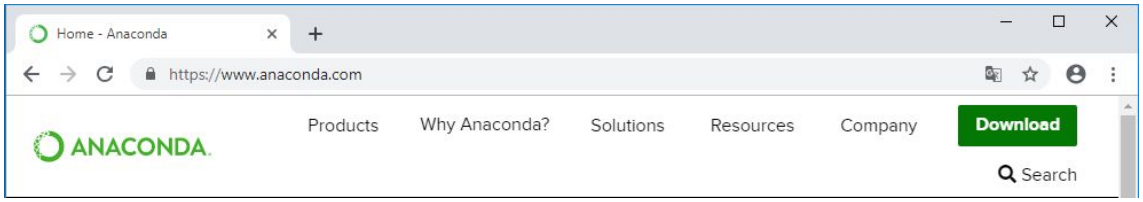
- Numpy
- Matplotlib
- PySerial
- Jupyter

Installing Anaconda

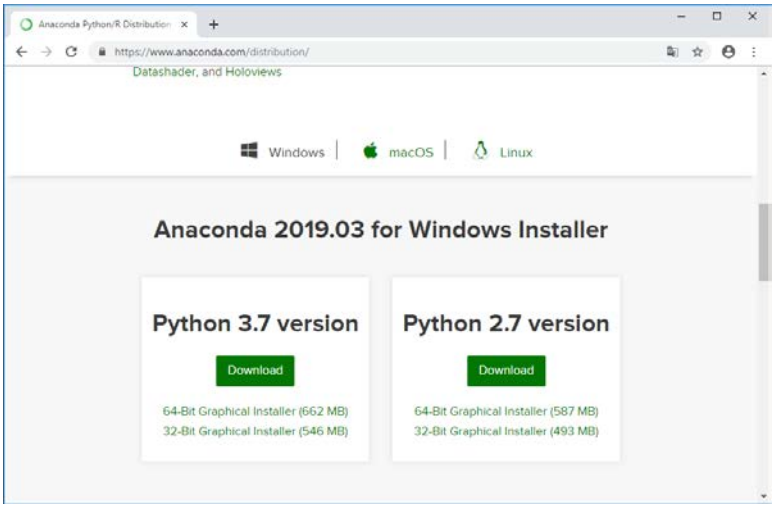
Although you can manually install Python and add the needed packages afterwards, we will only cover in this tutorial the simpler approach of using the **Anaconda** Python platform. You can get information about it on:

https://www.anaconda.com/

First we will go to this web page:

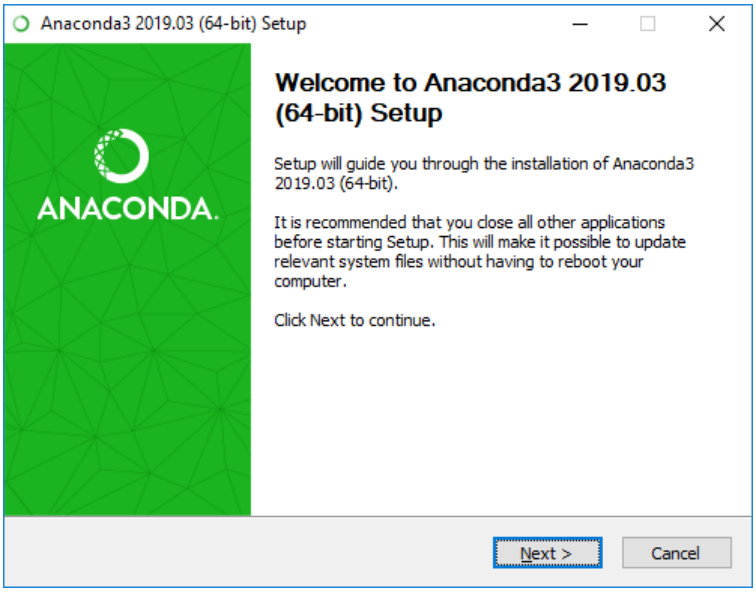


Then go to the download tab:

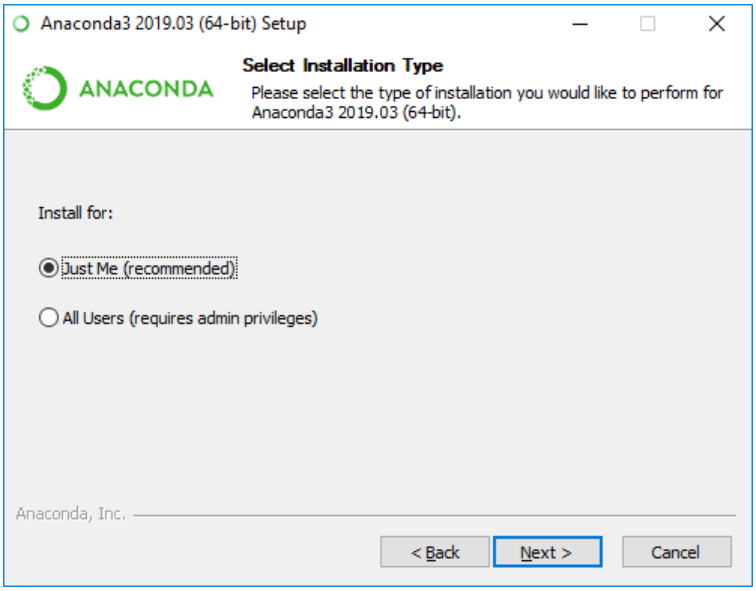


As this manual only covers Windows installations, select either the 64bit or 32bit of the Python 3.x installer depending on the version of your operation system. Nowadays, most probably you should select the 64bit version.

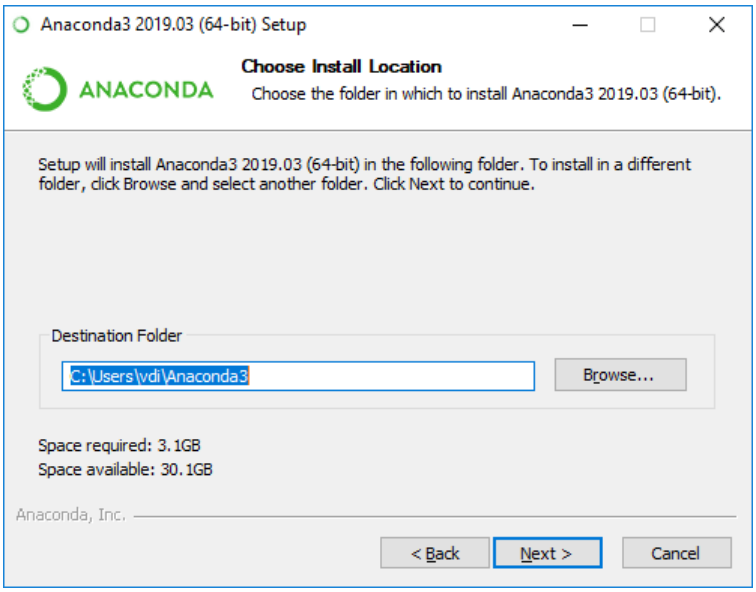
After downloading the installer, run it:



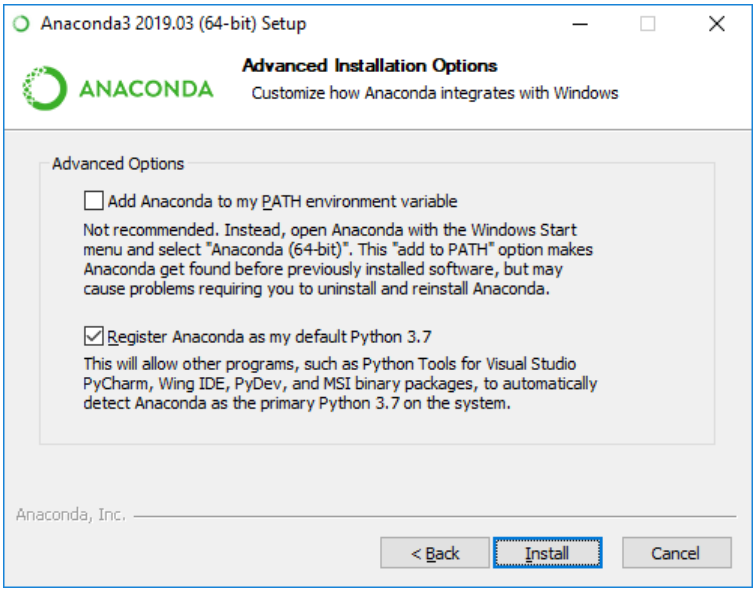
You can choose if you install just for you or all users in the system. If only you will use the SLab system, it is best to use the recommended *Just Me* setting.



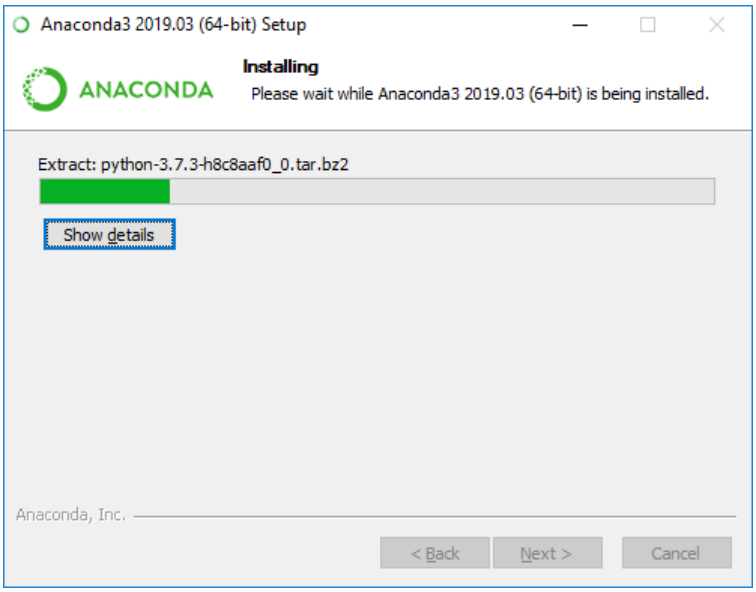
The installer lets you select the installation location. You can leave the default recommendation, but take note of where Anaconda is installed because you will need that information later.



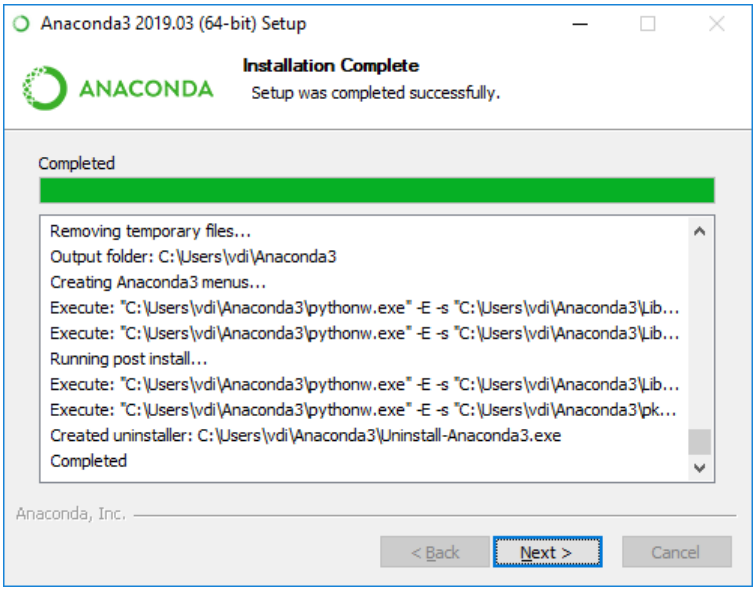
You can register Anaconda adding information to the *PATH* environment variable. It could ease some later procedures, but we will use the other recommended option as see on the following image.



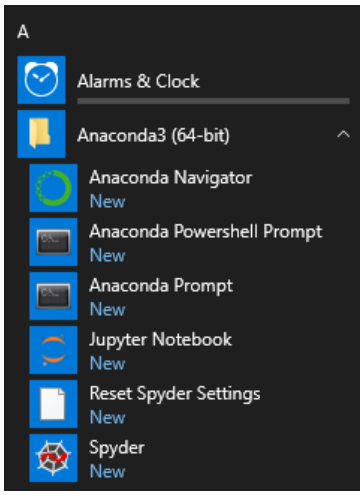
Now, Anaconda will start the installation. It would take some time so be patient.



After the copy of files is completed, go to the rest of screens until you end the installation.



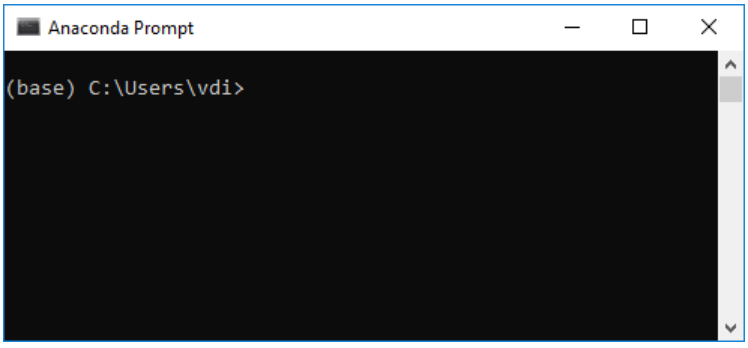
Now Anaconda should be available on the Windows Start Menu.



Adding PySerial

Anaconda comes preinstalled with most of the modules we need, but it usually lacks the **PySerial** module we need to communicate with the **Hardware Board**. So, the next step is to install this module.

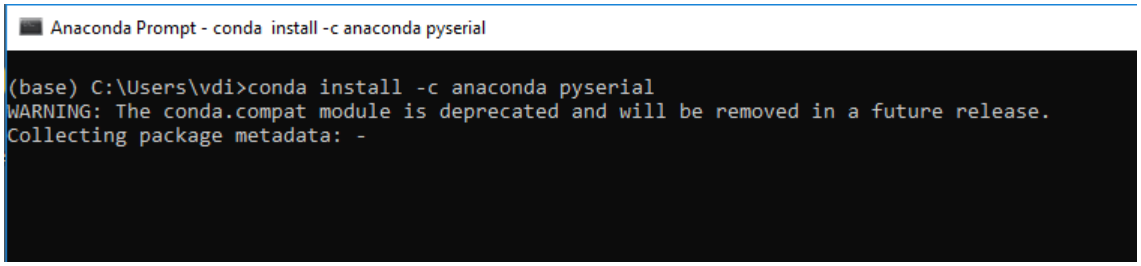
Locate the **Anaconda Prompt** application and run it.



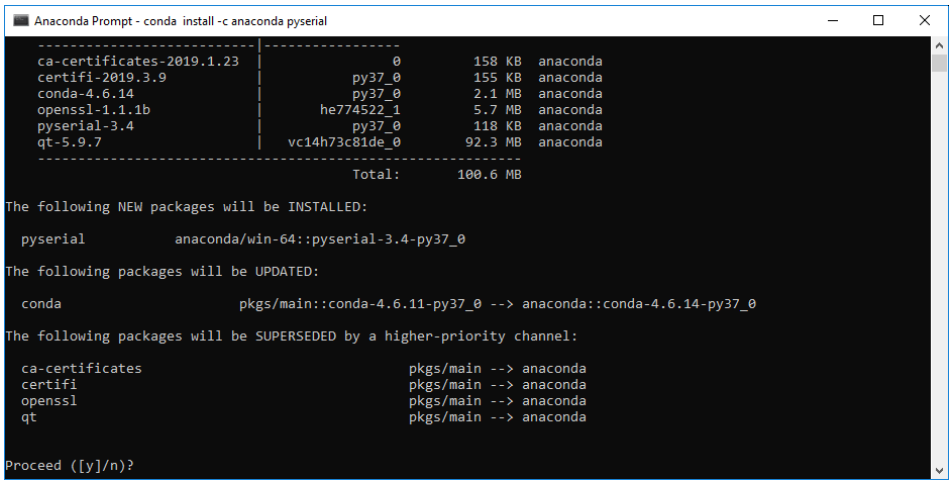
Then execute the following line:

```
conda install -c anaconda pyserial
```

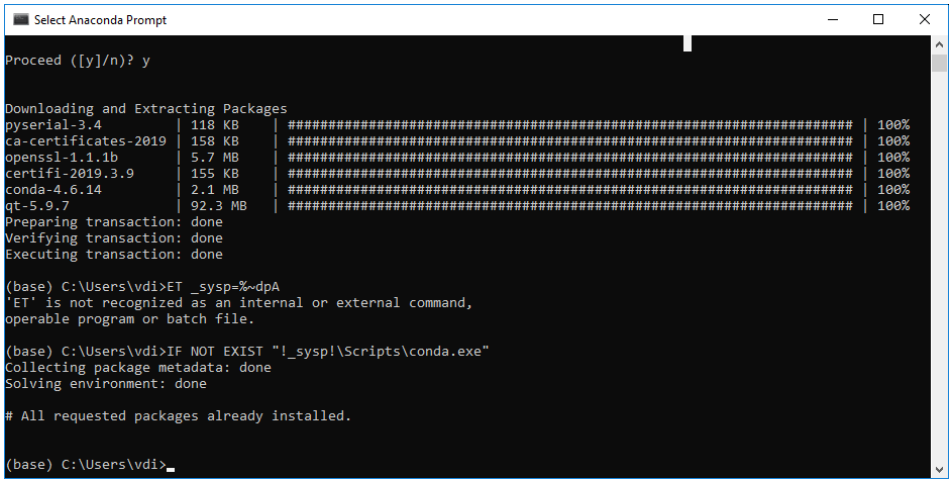
Just like it is shown in the following image.



After some checking, the window will ask if you want to proceed, respond Yes (“Y”).



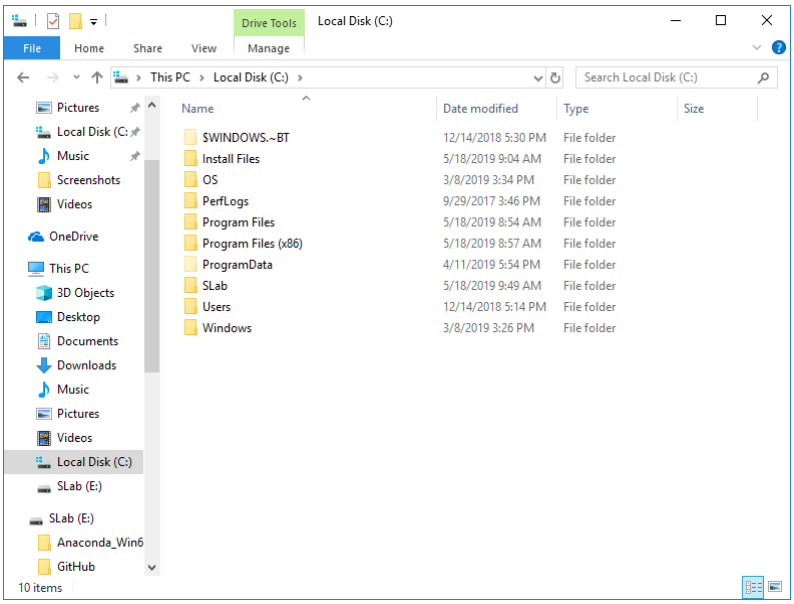
If all goes ok, the installer should end after the needed modules are updated.



At this point you have a fully fledged Anaconda Python 3.x system. Now it is time to add the **SLab** files.

Adding the SLab files

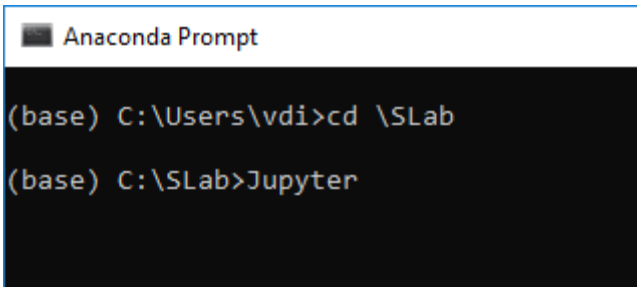
Locate the SLab installation zip file and uncompress its main **SLab** folder on the root of the C: drive so that it shows as in the figure below.
You can copy this file to other location but, in that case, you will need to change the configuration of the *PYTHONPATH* environment variable as will be explained at the end of this document.



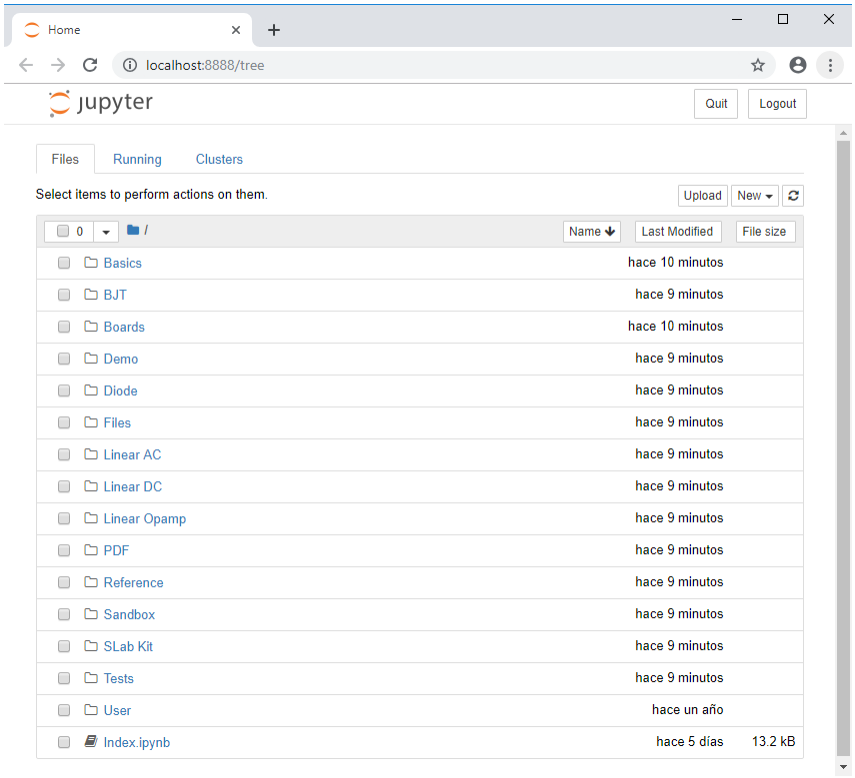
To start the **SLab system**, just open again the **Anaconda Prompt** from the Start Menu and execute the following two lines of code:

```
cd \SLab
Jupyter
```

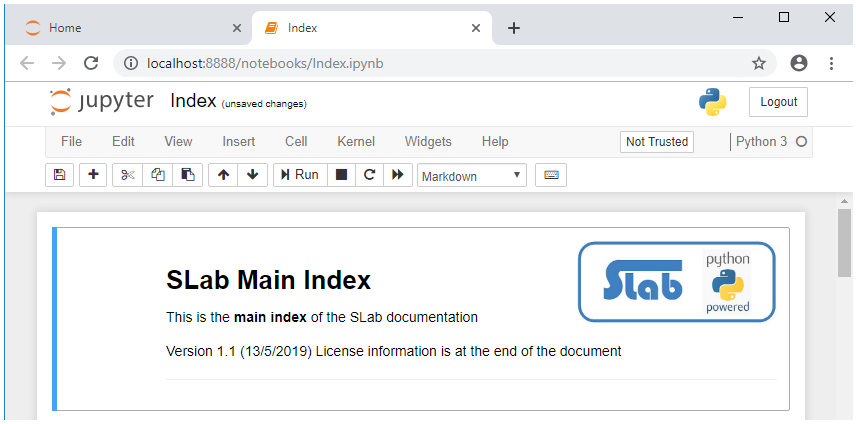
Just like is shown in the figure below.



Jupyter is a programming environment that uses web pages for the user interface. After a successful start, your default navigator should open and show a navigator for the files on the *\SLab\Jupyter* folder.



Locate the **Index.ipynb** file and click on it to open the notebook.



That's all. You now have a full SLab system. From now, do as indicated on the Index file.

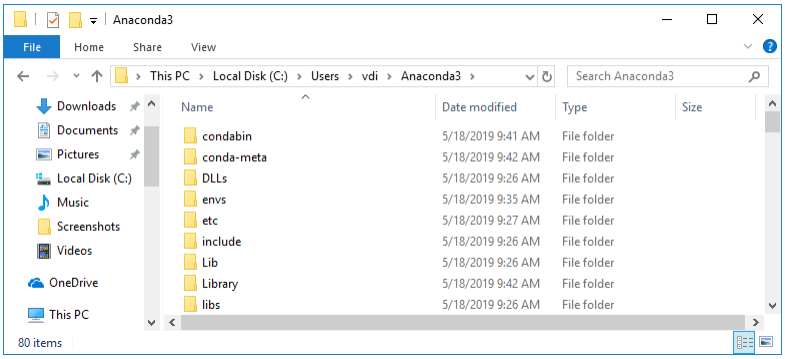
If you don't want to use the Anaconda prompt to start Jupyter, you can define a batch file to start the SLab system. Keep reading if you want to do that.

Create a custom launch file

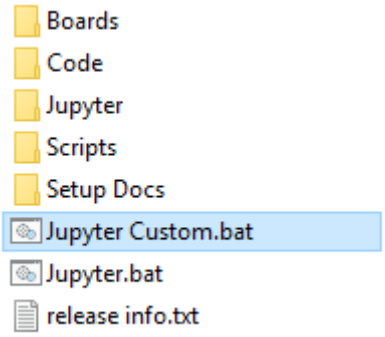
First, locate where Anaconda is installed, in this particular case it is on:

```
C:\Users\vdi\Anaconda3
```

In your case it could be different.



Now, locate the *Jupyter Custom.bat* file in the *\SLab* folder



Open this file for editing; you should get contents similar to the ones below:

```
SET PATH=%PATH%;\Anaconda_Win64
SET PATH=%PATH%;\Anaconda_Win64\Scripts
SET PATH=%PATH%;\Anaconda_Win64\Library\bin;
set PYTHONPATH=\SLab\Code
cd \SLab\Jupyter
jupyter notebook
```

Lines 1 to 3 set the three paths needed for Python to operate on the SLab system.

Line 4 set the *PYTHONPATH* environment variable to the location of the \SLab\Code folder than contains the SLab specific Python code. If you don't have copied the SLab zip contents to the C: root, you will need to change this variable to suit the position of the ***Code*** folder.

Line 5 enters the *\SLab\Jupyter* folder. Again, if you have changed the SLab folder location from the default one, you will need to change this line.

Finally, line 6 starts the **Jupyter** application.

Modify the file so that it matches your environment. In our particular case that has the SLab folder in the default location and Anaconda installed on:

```
C:\Users\vdi\Anaconda3
```

The proper file is:

```
SET PATH=%PATH%; C:\Users\vdi\Anaconda3
SET PATH=%PATH%; C:\Users\vdi\Anaconda3\Scripts
SET PATH=%PATH%; C:\Users\vdi\Anaconda3\Library\bin;
set PYTHONPATH=\SLab\Code
cd \SLab\Jupyter
jupyter notebook
```

Save the file and use it to start the SLab system.
Note that you can place this file whenever you want as long as it is in the C: drive.

Portable Installation

It is interesting to note that Anaconda does not put any file outside its installation folder. You can copy the Anaconda folder to a pendrive to have a **portable Python system**. Adding also the **SLab folder** to the root of the pendrive makes a fully portable SLab system.
In order for SLab to operate, you just need to modify the batch file to suit the location of the Anaconda paths. For instance, if you have Anaconda in an **`Anaconda_Win64`** folder at the root pendrive, you can modify *Jupyter Custom.bat* to be like:

```
SET PATH=%PATH%;\Anaconda_Win64
SET PATH=%PATH%;\Anaconda_Win64\Scripts
SET PATH=%PATH%;\Anaconda_Win64\Library\bin;
set PYTHONPATH=\SLab\Code
cd \SLab\Jupyter
jupyter notebook
```

It is possible that you also need to modify the **kernel.json** file that Jupyter uses to locate the Python kernel. This file is located in the folder:

```
share\jupyter\kernels\python3
```

Observe that this file should contain a path to the **python.exe** file. If you have Anaconda in an **`Anaconda_Win64`** folder at the root pendrive, the following contents can be used. Note the double \\ to generate the \ char after escape substitutions.

```
{
 "argv": [
  "\\Anaconda_Win64\\python.exe",
  "-m",
  "ipykernel_launcher",
  "-f",
  "{connection_file}"
 ],
 "display_name": "Python 3",
 "language": "python"
}
```