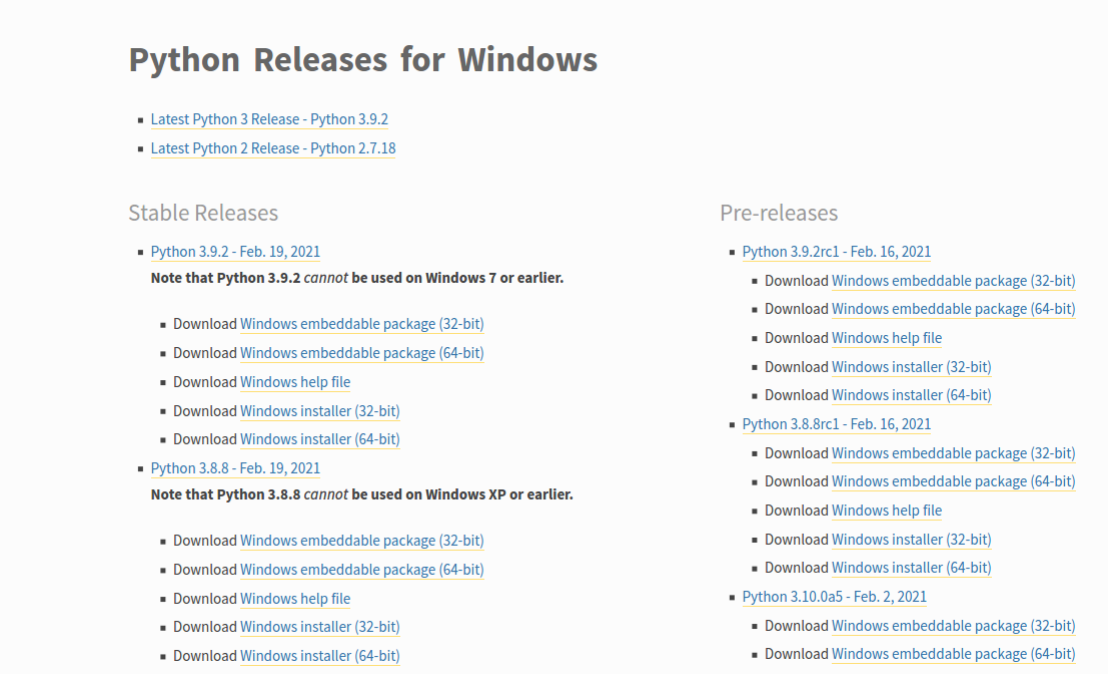
Installing Python and Jupyter on your computer

This is a short document outlining the steps needed to install Python and Jupyter on your own computer, as well as how to download the material for the labs. This is aimed at Windows users, however, the steps involved for Mac are very similar. If you do encounter problems, you can get in touch with us for help. Our contact details are below.

Name	Email
Ed	edward.antonian@thedatakirk.org.uk
Kamelia	kamelia.zhelyazkova@thedatakirk.org.uk
Fash	fash.fasoro@thedatakirk.org.uk
Chuks	chuks.ododo@thedatakirk.org.uk

1. Installing Python

The first step is to download and install Python on your computer. Head to www.python.org, click on Downloads and click Windows. You should see a page like this:



We recommend that you install Python 3.8.8. This is not in fact the most recent release, however, Python 3.9.0 is very new and may not be the most stable. Click the button that says **Windows installer (32-bit)**, unless you are confident that your computer is 64-bit, in which case you can select the 64-bit installer. (You can find out by going to My Computer, right clicking, and selecting properties). 32-bit is the safest option as it should work on all windows machines.

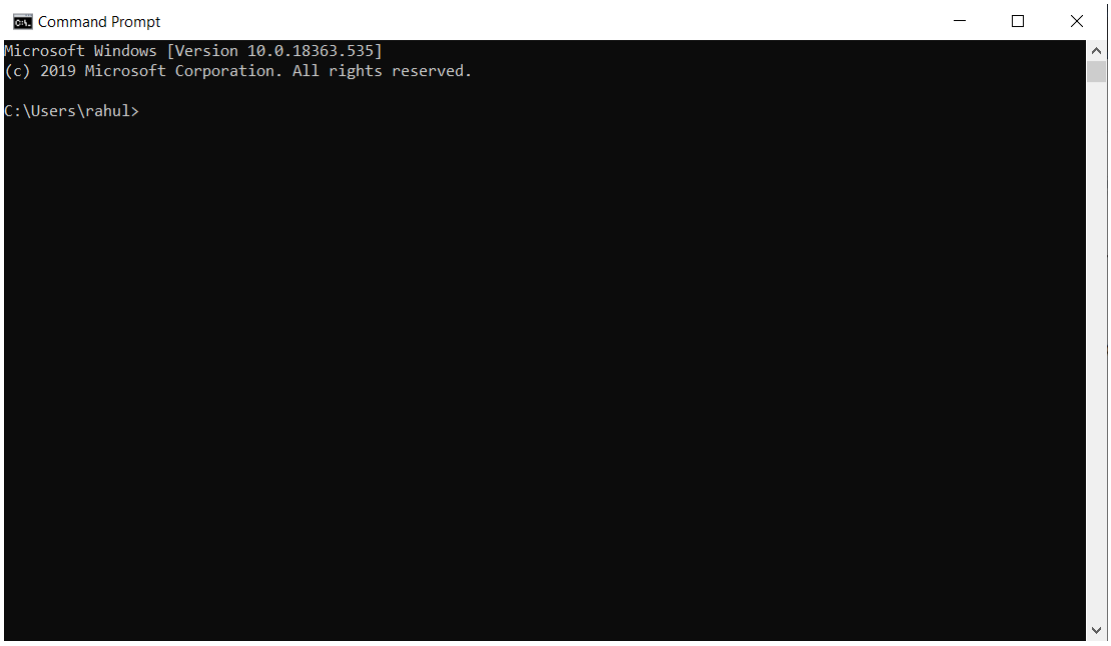
Once the download has finished, run the executable by double clicking it (or simply select run when asked what to do with the file on download). You should be presented with a box that looks something like this

IMPORTANT: You *must* click the button that says "add Python 3.8 to PATH".

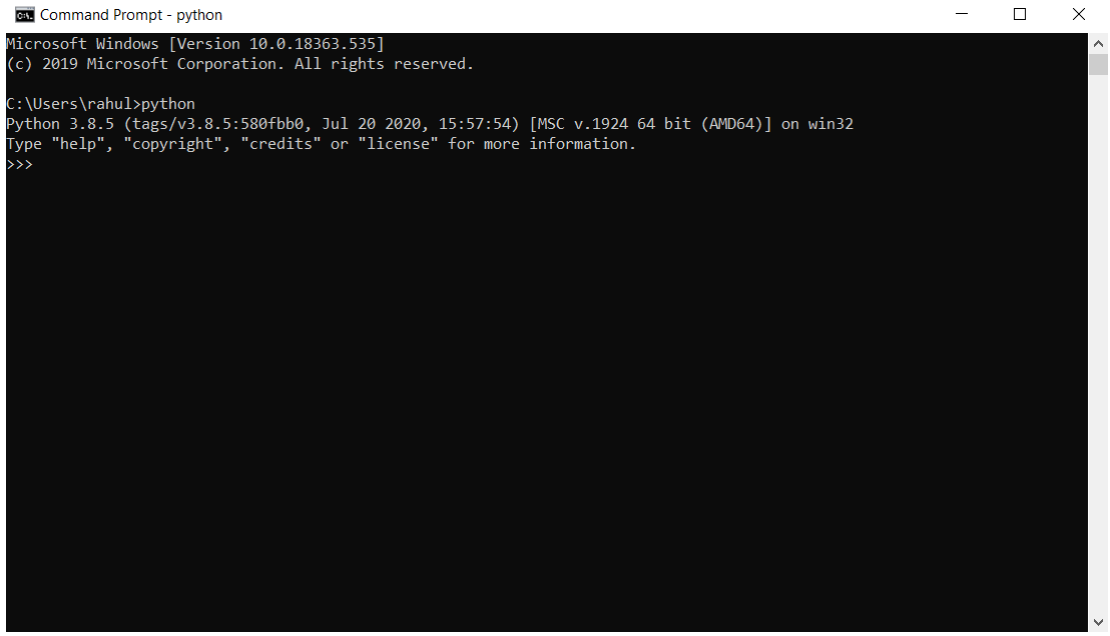
Once you have clicked add to path, you can click install now. You will be prompted to give administrator access - select yes. Once the setup is complete Python should be installed on your computer.

2. Checking the install success

In order to check that Python has successfully been installed, navigate to the command prompt. (If you are using Mac, this will be called the Terminal). On Windows, you can find this by pressing the Windows key, and typing `cmd`. This should bring up the command prompt application. Click it to start the command prompt. It should look something like this.



To verify that Python has installed correctly, type `python` in and press enter. If everything is OK, you should see something like this:



If you get a message saying something like *Python is not recognised as an internal or external command error* then you probably did not click add to PATH when installing. The easiest option is to hit the Windows key, type "add or remove programmes", scroll down to Python and uninstall. Once Python has uninstalled, begin again from the previous section.

3. Installing and running Jupyter

To install and run Jupyter, you will have to use the command prompt. If you are still in the Python interactive session (i.e. your command prompt still looks similar to the previous picture) type `exit()` and hit enter. This should take you back to the regular command prompt interface (like the picture before last). Here, type

```
pip3 install jupyter
```

and press enter. This should install Jupyter. Once this is complete (may take up to 2 minutes), test the installation success by typing into your command prompt

```
jupyter notebook
```

and press enter. This should automatically load up the Jupyter interface that you have seen in the lab sessions in your default web-browser. You are now ready to use Jupyter! You can test it by clicking 'new' and then 'python 3'. This will open a new blank Jupyter notebook where you should be able to write and run code.

You need to leave the command prompt open while you are running Jupyter, as they are connected. Once you are done with Jupyter, you can close the session by going back to the command prompt and pressing `ctrl+c`, and then `Y` to confirm the exit.

4. Installing some other packages

While you're here, you might as well install some other Python packages that we will be using later in the course. Open up a new command prompt and type

```
pip3 install numpy pandas matplotlib scikit-learn
```

and hit enter. This will also install the packages `numpy`, `pandas`, `matplotlib` and `scikit-learn` which we will be using later in the course.