Python

Use Python in TeideHPC

Once connected to the login nodes, we have a default version of Python, which is version 2.6.6. To see the available Python versions, we use the *modules* tool:

```
module ava python

---------------------------------------------- /opt/envhpc/modulefiles/.rhel6 ----------------------------------------------
python/2.7.18/gcc python/3.5.4/gcc  python/3.7.9/gcc  python/3.8.11/gcc
```

To load any of the versions, we may need to load a module beforehand:

```
module load python/3.8.11/gcc
python/3.8.11/gcc(10):ERROR:151: Module 'python/3.8.11/gcc' depends on one of the module(s)
'openssl/1.1.1k/gcc'
python/3.8.11/gcc(10):ERROR:102: Tcl command execution failed: prereq openssl/1.1.1k/gcc
```

Therefore, we load the necessary modules, in the corresponding order:

```
module load openssl/1.1.1k/gcc python/3.8.11/gcc

module list
Currently Loaded Modulefiles:
 1) openssl/1.1.1k/gcc   2) python/3.8.11/gcc
```

We will now be able to use a specific version of Python and use environments, where we can install Python packages in isolation.

Install Python packages in /data

By default, Python will install packages in the user's `/home` partition. This storage is limited in size, so we should configure the environments and install the software in the `/data` partition. To do this, we need to do the following:

```
pip3 install --target=$HOME/data/tutu --install-option="--install-scripts=$HOME/data/foo" Package
```

The `--target=DIR` option tells pip where to install the requested package and all its dependencies. On the other hand, the `--install-option="--install-scripts=DIR` option indicates where the binaries will be installed, in case you use the python package from the command line directly.

Then, we have to add to our PATH the new path to use the binaries directly:

```
export PYTHONPATH="$HOME/data/tutu"
```

To make it permanent, we can add that line to the end of our `~/.bashrc`.