Most modulefiles include a one-line description of what the software package does, which can be seen with the command:

```
$ module whatis module-name
```

For example:

```
$ module whatis cufflinks/intel/2.1.1

Cufflinks assembles transcripts, estimates their abundances, and tests for differential expression and regulation in RNA-Seq samples
```

A modulefile may include more detailed help for the software package, which can be seen with:

```
$ module help module-name
```

For example:

```
$ module help cufflinks/intel/2.1.1

--------- Module Specific Help for 'cufflinks/intel/2.1.1'
---------------------------
Adds cufflinks to your environment.
```

As the example illustrates, the NYU HPC modulefiles currently have limited help information. We recommend visiting the website of the software package for accurate and up-to-date documentation.

You can also see exactly what effect loading the module will have with:

```
$ module show module-name
```

For example:

```
$ module show cufflinks/intel/2.1.1

-----------------------------------------------
/share/apps/modules/modulefiles/cufflinks/intel/2.1.1:
module-whatis Cufflinks assembles transcripts, estimates their abundances, and tests for differential expression and regulation in RNA-Seq samples
conflict cufflinks
module load intel/11.1.046
module load zlib/intel/1.2.5
module load python/intel/2.7.2
module load boost/intel/serial/1.47.0
module load eigen/3.0.5
prepend-path PATH /share/apps/cufflinks/2.1.1/intel/bin
setenv CUFFLINKS_ROOT /share/apps/cufflinks/2.1.1/intel
-----------------------------------------------
```

Note that the cufflinks module, like many others on the NYU HPC clusters, loads a number of other modules. This is to ensure that when you use a module, your environment is consistent with the environment in which the package was built and tested. To see what modules are currently loaded in your environment use "module list" as described below (TODO link).