


```
$module show modulename      -- show module information
```

```
$module whatis modulename    -- show modulefile information
```

```
$module update modulename    -- reload module
```

```
$module help                  -- show module help
```

Examples

```
$ which gcc          -> /usr/bin/gcc
$ module load gcc/10.2.0
$ which gcc          -> /opt/envhpc/utils/rhel6/gcc/10.2.0/bin/gcc
```

You can see how the application executable is being used before and after loading the module.

```
$ module av 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
```

```
$ which python      -> /usr/bin/python
$ which python3     -> /usr/bin/which: no python3 in (/usr/local/bin:...)
```

```
$ module load python/3.7.9/gcc -> Requisito cargado openssl/1.1.1k/gcc
```

```
python/3.7.9/gcc(10):ERROR:151: Module 'python/3.7.9/gcc' depends on one of the
module(s) 'openssl/1.1.1k/gcc'.
python/3.7.9/gcc(10):ERROR:102: Tcl command execution failed: prereq openssl/1.1.1k/gcc.
```

```
$ module load openssl/1.1.1k/gcc python/3.7.9/gcc
$ which python3     -> /opt/envhpc/utils/rhel6/python/3.7.9/gcc-10.2.0/bin/python3
```

As first step, look for some version of python available on the system.

Next, you will notice that *python3* command is not available.

When loading the python module, say *openssl / 1.1.1k / gcc* is a requirement to load *python/3.7.9/gcc*.

After these two modules are loaded, you can see how python version 3.7.9 is available.

¿How to use modules modules with a script?