# How to use pip and pypi

Pip is a package manager for Python. You can use it to install modules. Sometimes systems have two versions of pip in the store, you need version 3 (the newest).

A module is code: in the form of functions or objects. You can include this in your program and build upon it. You could see this as premade parts that you use build your project with.

PyPI is the Python package index, where all the official modules for Python are stored.

## Pip

### Install pip

Installation of pip is easy. You can install it with the system package manager. If you use Linux it’s often already installed.

On Linux you can install it with:

|  |
| --- |
| *# debian and ubuntu linux* sudo apt-get install python3-pip  *# fedora linux* sudo yum install python3-pip |

On Mac OS X, install it with easy\_install.

|  |
| --- |
| sudo easy install pip |

### Install module

Is pip installed? It can install packages from the PyPi repository. It’s the official repository for python modules.

Software you install with pip is downloaded from the PyPi repo and installed.

To install a module, simply type:

|  |
| --- |
| pip install modulename |

This could be any module from the PiPy index, lets take playsound:

|  |
| --- |
| pip install playsound |

You’d always want to install inside a virtual environment and not globally on your system.

Virtualenv creates an isolated environment, so packages you install won’t affect other python projects. You can do that in this way:

|  |
| --- |
| virtualenv name cd name source bin/activate |

Then your environment is active and you can install modules with pip,

|  |
| --- |
| pip install package |

If you want to end working with the project type deactivate.

### Search

You can find packages in the PyPi index either online or in the command line.  
To search in the command line type the command below, where topic is the word you want to search for.

|  |
| --- |
| pip search topic |
| This will show a list of available software modules. |