Virtual environments

WE'LL COVER THE FOLLOWING

- Installation

Virtual environments can be really handy for testing software. That's true in programming circles too. Ian Bicking created the virtualenv project, which is a tool for creating isolated Python environments. You can use these environments to test out new versions of your software, new versions of packages you depend on or just as a sandbox for trying out some new package in general. You can also use virtualenv as a workspace when you can't copy files into site-packages because it's on a shared host. When you create a virtual environment with virtualenv, it creates a folder and copies Python into it along with a site-packages folder and a couple others. It also installs pip. Once your virtual environment is active, it's just like using your normal Python. And when you're done, you can just delete the folder to cleanup. No muss, no fuss. Alternatively, you can keep on using it for development.

In this chapter, we'll spend some time getting to know virtualenv and how to use it to make our own magic.

Installation

First of all, you probably need to install virtualenv. You can use pip or easy_install to install it or you can download the virtualenv.py file from their website and install it from source using its **setup.py** script.

If you have Python 3.4, you will find that you actually have the **venv** module, which follows an API that is very similar to the **virtualenv** package. This chapter will focus on just the virtualenv package, however.