



Search



(/).  
([https://www.sitepoint.com/themes/?utm\\_source=blog&utm\\_medium=main-nav](https://www.sitepoint.com/themes/?utm_source=blog&utm_medium=main-nav))



Podcast  
(<https://www.sitepoint.com/versioning-show/>)



Forums(<https://www.sitepoint.com/community/>)

Premium(<https://www.sitepoint.com/premium/>)

• [Programming\(https://www.sitepoint.com/programming/\)](https://www.sitepoint.com/programming/) Article

# Virtual Environments in Python Made Easy

By [Oltjano Terpollari \(https://www.sitepoint.com/author/oterpollari/\)](https://www.sitepoint.com/author/oterpollari/)

April 17, 2015

Most Python newcomers don't know how to set up a development environment that follows the latest standards used by professional programmers, so this tutorial will teach you how to properly create a fully working Python development environment using industry accepted best practices.



**Ending Soon: Get Every SitePoint Ebook and Course for FREE!**

87 Ebooks, 70 Courses and 300+ Tutorials all yours for FREE from SitePoint!

## RECOMMENDED

- 1 [Picking the Brains of Your Customers \(https://www.sitepoint.com/picking-the-brains-of-your-customers/\)](https://www.sitepoint.com/picking-the-brains-of-your-customers/)  
[utm\\_source=sitepoint&utm\\_medium=referral](https://www.sitepoint.com/picking-the-brains-of-your-customers/?utm_source=sitepoint&utm_medium=referral)
- 2 [Is It Possible to Write and Run PHP Code on an iPad? \(https://www.sitepoint.com/is-it-possible-to-write-and-run-php-code-on-an-ipad/\)](https://www.sitepoint.com/is-it-possible-to-write-and-run-php-code-on-an-ipad/)  
[utm\\_source=sitepoint&utm\\_medium=referral](https://www.sitepoint.com/is-it-possible-to-write-and-run-php-code-on-an-ipad/?utm_source=sitepoint&utm_medium=referral)
- 3 [Sourcehunt September - Hacktoberfest! \(https://www.sitepoint.com/sourcehunt-september-hacktoberfest/\)](https://www.sitepoint.com/sourcehunt-september-hacktoberfest/)  
[utm\\_source=sitepoint&utm\\_medium=referral](https://www.sitepoint.com/sourcehunt-september-hacktoberfest/?utm_source=sitepoint&utm_medium=referral)
- 4 [There Are More Virtual Reality Headsets Than You Realize! \(https://www.sitepoint.com/there-are-more-virtual-reality-headsets-than-you-realize/\)](https://www.sitepoint.com/there-are-more-virtual-reality-headsets-than-you-realize/)  
[utm\\_source=sitepoint&utm\\_medium=referral](https://www.sitepoint.com/there-are-more-virtual-reality-headsets-than-you-realize/?utm_source=sitepoint&utm_medium=referral)
- 5 [Developing Add-ons for Enterprise Applications](#)



Claim This Deal!  
([https://www.sitepoint.com/Premium/L/Sitepoint-Premium-Siteground?](https://www.sitepoint.com/Premium/L/Sitepoint-Premium-Siteground?Utm_source=Sitepoint&Utm_medium=Mid-Article&Utm_campaign=Siteground)  
[Utm\\_source=Sitepoint&Utm\\_medium=Mid-](https://www.sitepoint.com/Premium/L/Sitepoint-Premium-Siteground?Utm_source=Sitepoint&Utm_medium=Mid-Article&Utm_campaign=Siteground)  
[Article&Utm\\_campaign=Siteground\)](https://www.sitepoint.com/Premium/L/Sitepoint-Premium-Siteground?Utm_source=Sitepoint&Utm_medium=Mid-Article&Utm_campaign=Siteground)

([https://www.sitepoint.com/developing-jira/?](https://www.sitepoint.com/developing-jira/?utm_source=sitepoint&utm_medium=)  
[utm\\_source=sitepoint&utm\\_medium=](https://www.sitepoint.com/developing-jira/?utm_source=sitepoint&utm_medium=)

## SPONSORS

### THEMES STORE

• [THEMES STORE](#)  
[Download our new SitePoint Base Theme. 100% FREE. GPL 2.0.](#)

([the utm prog](#)

► 00:11:17

• [WEB An Introduction to Elm Data Structures](#)

([ht int](#)

# Virtual Environments

There are so many libraries I use for my own personal projects, three web application development frameworks I use and many others I would love to explore in the near future. It is very clear that a serious project in Python depends on other packages written by other developers. If you are a Django developer, I am very sure you make use of **South** for performing automatic database migrations, **Django Debug Toolbar** for gathering various debug information about the current request/response, **Celery** for taking care of real-time operations and scheduling as well, and so on.

I, for example, use the package **requests** a lot for my projects, and a Django web application I am currently working on depends on version 2.3.0. According to the official documentation, the latest version of this package is version 2.5.1. Say I go ahead and install the latest version of the library on my Ubuntu machine because I need it for another project, and everything seems to work fine until I try to make use of my older project which worked fine with 2.3.0. Suddenly, everything is broken. What happened? Maybe the API of the latest version of requests has changed since version 2.3.0? The reason does not matter at this point, as my older project is broken and no longer works.

A conflict between two projects has been created. They make use of the same library, but they require different versions of it.

The tool [virtualenv](https://virtualenv.pypa.io/en/latest/) (<https://virtualenv.pypa.io/en/latest/>) is the solution to this problem.

## virtualenv

Virtualenv helps solve project dependency conflicts by creating isolated environments which can contain all the goodies Python programmers need to develop their projects. A virtual environment created using this tool includes a fresh copy of the Python binary itself as well as a copy of the entire Python standard library.

To create a virtual environment for your Python project, you type the command **virtualenv** followed by the name you want to give to the virtual environment like shown below.

```
virtualenv virt1
```

It is very important to know that each time you want to use a created virtual environment, you need to activate it. The following command does this:

```
source virt1/bin/activate
```

Everything installed in the **virt1** directory will not affect the global packages or the system wide installations, thus avoiding dependency conflicts.

If I now install the newest version of requests, the old one does not get uninstalled. The new one is installed in the virtual environment exclusively.

```
pip -I install requests
```

The above command produces the following output:

```
Downloading/unpacking requests
  Downloading requests-2.5.1-py2.py3-none-any.whl (464kB): 464kB downloaded
Installing collected packages: requests
Successfully installed requests
Cleaning up...
```

## pip

A word on the **pip** command above – it's a package manager that is very useful when you want to distribute your project to others, as it allows the developer to install all the required packages with a **requirements.txt** file.

For example, a fellow developer can activate a virtual environment and then run the following command to install the dependencies of the project:

```
pip install -r requirements.txt
```

To generate the dependencies file of your project, you run the following command.

```
pip freeze > requirements.txt
```

The pip tool can also be used to uninstall a package from your machine:

```
pip uninstall some-package-name
```

One can install pip system-wide with the default Python package manager:

```
sudo easy_install pip
```

And then use pip to install **virtualenv**:

```
sudo pip install virtualenv
```

## After the environment is created

Once the virtual environment gets activated, your terminal prompt changes to show the user in which directory they are working:

```
(virt1)oltjano@baby:~/Desktop/myproject$
```

The following command will let you deactivate the virtual environment:

```
deactivate
```

You can use the **which** command to check the Python binary that is used in the current virtual environment.

```
which python
```

If everything is working well, you should get something similar to the following output.

```
/home/oltjano/Desktop/myproject/virt1/bin/python
```

If you **deactivate** and **which** again, you should get a different output.

```
/usr/bin/python
```

This is because when working inside a virtual environment, the binary copy placed inside that environment is being used. The same applies to packages.

We can use the option `-p` while working with `virtualenv` in order to use a specific version of Python which is globally installed on the machine.

For example, the following command can be used to create the virtual environment **virt2** with Python3 in it, if you have Python3 installed on your machine.

```
virtualenv -p /usr/bin/python3 virt2
```

And to delete a virtual environment you use the `rm -r` command like you do with any other directory you want to delete.

```
rm -r virt2
```

## virtualenvwrapper

Virtualenvwrapper provides very useful commands that make working with virtual environments even easier. For example, the **workon** command can be used to activate and work on a virtual environment:

```
workon virt1
```

It, too, is installed easily with `pip`.

```
pip install virtualenvwrapper
```

I like to keep all my virtual environments in a single place. To accomplish this, run the following command to create a new directory for your virtual environments.

```
mkdir ~/.virtualenvs
```

Then set `WORKON_HOME` to `~/.virtualenvs` like shown below.

```
export WORKON_HOME=~/.virtualenvs
```

Once this is done, we can open the `.bashrc` file inside the home directory using a text editor and add the following line to the end of it.

```
. /usr/local/bin/virtualenvwrapper.sh
```

Then, we reload the `.bashrc` file with the following command.

```
source ~/.bashrc
```

Now, the `mkvirtualenv` command can be used to easily make new environments placed by default inside this folder.

```
mkvirtualenv sitepoint
```

We can easily activate the virtual environment with the help of the `workon` command:

```
workon sitepoint
```

The command to deactivate the virtual environment is the same as the one we used before.

```
deactivate
```

It's very easy to switch between different virtual environments. For example, to `workon` another virtual environment:

```
workon another_virtualenv
```

To delete a virtual environment, the command **rmvirtualenv** should be used.

```
rmvirtualenv sitepoint
```

## Autoenv

Another very useful tip is the one that helps automatically activate the virtual environments when the one cd's into their project.

For this, a **.env** file should be placed inside the project directory.

```
cd sitepointproject  
vim .env
```

Inside the **.env** file, we insert the following line.

```
workon name_of_virtualenv_here
```

The **autoenv** tool will help us automagically activate the environment when navigating into a directory containing a **.env** file. To install it, we clone it.

```
git clone git://github.com/kennethreitz/autoenv.git ~/.autoenv
```



Then we use **echo** to place another line inside the **.bashrc** file. You can also use a text editor to do it.

```
echo 'source ~/.autoenv/activate.sh' >> ~/.bashrc
```

Now, every time you **cd** into your project, the virtual environment will get activated automatically.

# Conclusion

In this tutorial, we learned some interesting hacks for efficiency when working with Python virtual environments. The tools are out there – embrace them and improve your workflow. Do you use any other interesting approaches in your Python development workflow? Let us know!

Was this helpful?  

More: [autoenv](https://www.sitepoint.com/tag/autoenv/) (<https://www.sitepoint.com/tag/autoenv/>), [python](https://www.sitepoint.com/tag/python-2/) (<https://www.sitepoint.com/tag/python-2/>), [virtualenv](https://www.sitepoint.com/tag/virtualenv/) (<https://www.sitepoint.com/tag/virtualenv/>)

[https://  
www.sitepoint.com](https://www.sitepoint.com)

Meet the author

**Oltjano Terpollari** (<https://www.sitepoint.com/author/oterpollari/>)  (<https://twitter.com/jebird1013>)

## Comments

**Have Your Say** (<http://community.sitepoint.com/t/virtual-environments-in-python-made-easy/145972>)

## LATEST THEMES >

([themes/](/themes/))

[Browse all 9 themes \(/themes/\)](/themes/) >



 **PREMIUM THEME**

**[SitePoint WordPress Restaurant Theme](https://www.sitepoint.com/wordpress-restaurant-theme/)**

(<https://www.sitepoint.com/wordpress-restaurant-theme/>)

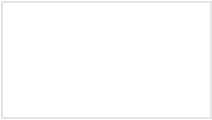


 **PREMIUM THEME**

**[SitePoint WordPress Ecommerce Theme](https://www.sitepoint.com/wordpress-ecommerce-theme/)**

(<https://www.sitepoint.com/wordpress-ecommerce-theme/>)





 **PREMIUM THEME**

**SitePoint WordPress Portfolio Theme**

[\(https://www.sitepoint.com/wordpress-portfolio-theme/\)](https://www.sitepoint.com/wordpress-portfolio-theme/)

**LATEST COURSES >**

[\(/premium/courses/\)](/premium/courses/)

[Browse all 71 courses \(/premium/courses/\)](/premium/courses/) >

⌚ 2h 2m

 **PREMIUM COURSE**

**Responsive Web Development**

[\(https://www.sitepoint.com/premium/courses/responsive-web-development-2929\)](https://www.sitepoint.com/premium/courses/responsive-web-development-2929)

⌚ 1h 46m

 **PREMIUM COURSE**

**Elm: A Beginners' Guide to Elm and Data**

[\(https://www.sitepoint.com/premium/courses/elm-a-beginners-guide-to-elm-and-data-2940\)](https://www.sitepoint.com/premium/courses/elm-a-beginners-guide-to-elm-and-data-2940)

⌚ 44m

 **PREMIUM COURSE**

**Introducing TypeScript**

[\(https://www.sitepoint.com/premium/courses/introducing-typescript-2933\)](https://www.sitepoint.com/premium/courses/introducing-typescript-2933)

**LATEST BOOKS >**

[\(/premium/books/\)](/premium/books/)

[Browse all 87 books \(/premium/books/\)](/premium/books/) >

 **PREMIUM BOOK**

**Designing UX: Prototyping**

[\(https://www.sitepoint.com/premium/books/designing-ux-prototyping\)](https://www.sitepoint.com/premium/books/designing-ux-prototyping)

 **PREMIUM BOOK**

**Researching UX: Analytics**

[\(https://www.sitepoint.com/premium/books/researching-ux-analytics\)](https://www.sitepoint.com/premium/books/researching-ux-analytics)

 **PREMIUM BOOK**

**Rails: Novice to Ninja**

[\(https://www.sitepoint.com/premium/books/rails-novice-to-ninja\)](https://www.sitepoint.com/premium/books/rails-novice-to-ninja)

Get the latest in Front-end, once a week, for free.

Enter your email

Subscribe

#### About

[Our Story \(/about-us/\)](#)  
[Advertise \(/advertise/\)](#)  
[Press Room \(/press/\)](#)  
[Reference \(http://reference.sitepoint.com/css/\)](http://reference.sitepoint.com/css/)  
[Terms of Use \(/legals/\)](#)  
[Privacy Policy \(/legals/#privacy\)](#)  
[FAQ \(https://sitepoint.zendesk.com/hc/en-us\)](https://sitepoint.zendesk.com/hc/en-us)  
[Contact Us \(mailto:feedback@sitepoint.com\)](mailto:feedback@sitepoint.com)  
[Contribute \(/write-for-us/\)](#)

#### Visit

[SitePoint Home \(/\)](#)  
[Themes \(/themes/?utm\\_source=blog&utm\\_medium=footer\)](#)  
[Podcast \(/versioning-show/\)](#)  
[Forums \(https://www.sitepoint.com/community/\)](https://www.sitepoint.com/community/)  
[Newsletters \(/newsletter/\)](#)  
[Premium \(/premium/\)](#)  
[References \(/sass-reference/\)](#)  
[Versioning \(https://www.sitepoint.com/versioning/\)](https://www.sitepoint.com/versioning/)

#### Connect

[\(https://www.sitepoint.com/feed/\)](https://www.sitepoint.com/feed/)   
[\(/newsletter/\)](#)   
[k.com/sitepoint](https://www.facebook.com/sitepoint)   
[. \(http://twitter.com/sitepointdotcom\)](http://twitter.com/sitepointdotcom)   
[om/+sitepoint](https://plus.google.com/+sitepoint)

© 2000 – 2017 SitePoint Pty. Ltd.

