



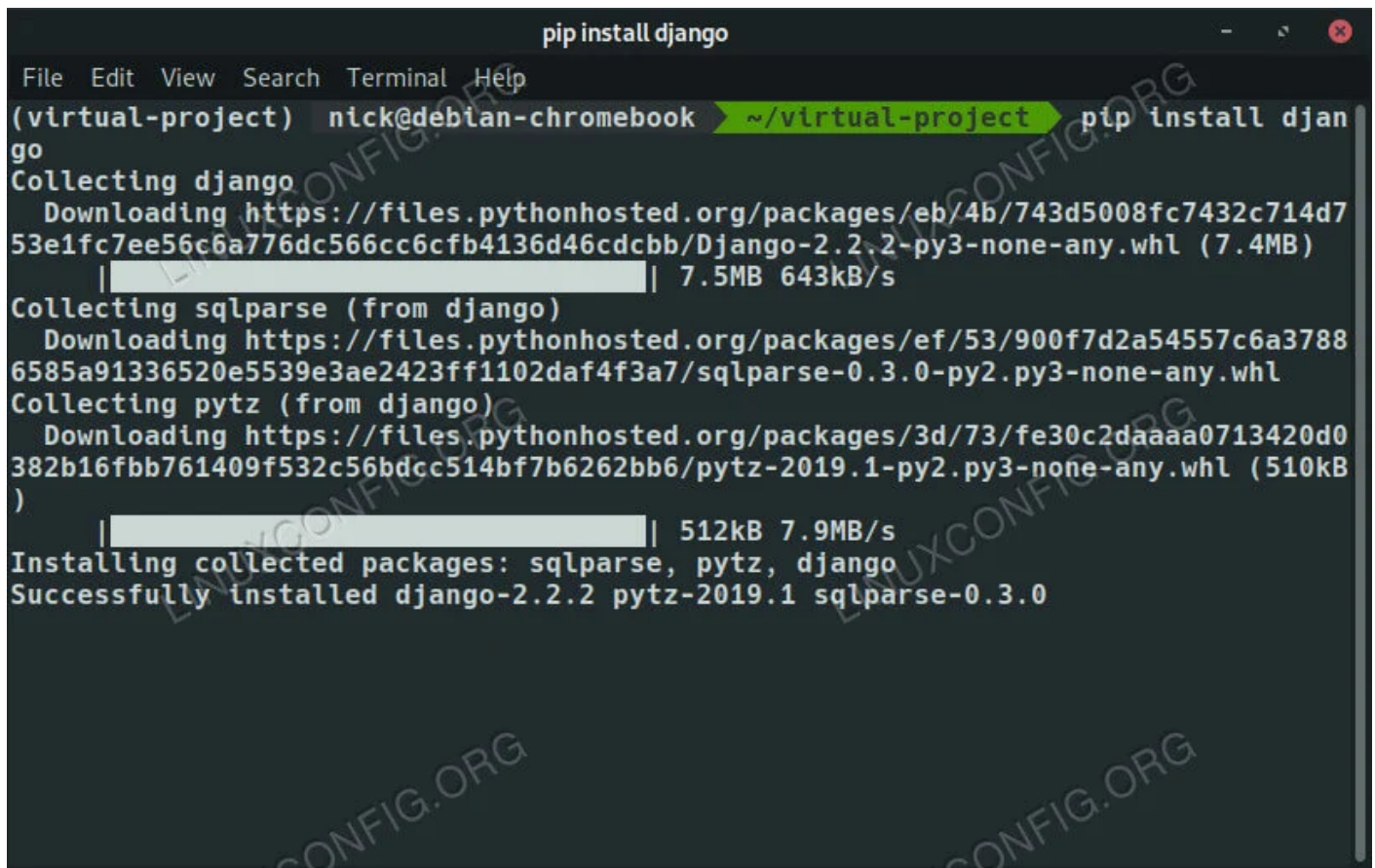
# How to Set Up a Python Virtual Environment on Debian 10 Buster

8 June 2019 by Nick Congleton

There are two very simple ways to create a Python virtual environment on Debian 10. They're very similar and offer nearly the same benefits. As an added bonus, you won't need to install anything outside of the default Debian repositories to use them.

## In this tutorial you will learn:

- How to Install the Dependencies
- How to Use Python 3's Venv
- How to Use Virtualenv



```

pip install django

(virtual-project) nick@debian-chromebook ~/virtual-project$ pip install djan
go
Collecting django
  Downloading https://files.pythonhosted.org/packages/eb/4b/743d5008fc7432c714d7
53e1fc7ee56c6a776dc566cc6cfb4136d46cdcbb/Django-2.2.2-py3-none-any.whl (7.4MB)
  |████████████████████| 7.5MB 643kB/s
Collecting sqlparse (from django)
  Downloading https://files.pythonhosted.org/packages/ef/53/900f7d2a54557c6a3788
6585a91336520e5539e3ae2423ff1102daf4f3a7/sqlparse-0.3.0-py2.py3-none-any.whl
Collecting pytz (from django)
  Downloading https://files.pythonhosted.org/packages/3d/73/fe30c2daaaa0713420d0
382b16fbb761409f532c56bdcc514bf7b6262bb6/pytz-2019.1-py2.py3-none-any.whl (510kB)
  |████████████████████| 512kB 7.9MB/s
Installing collected packages: sqlparse, pytz, django
Successfully installed django-2.2.2 pytz-2019.1 sqlparse-0.3.0

```

*Python Virtual Environment on Debian 10.*

## Software Requirements and Conventions Used

*Software Requirements and Linux Command Line Conventions*

Category	Requirements, Conventions or Software Version Used
System	Debian 10 Buster
Software	Python 3
Other	Privileged access to your Linux system as root or via the <b>sudo</b> command.
Conventions	<p># – requires given <b>linux commands</b> to be executed with root privileges either directly as a root user or by use of <b>sudo</b> command</p> <p>\$ – requires given <b>linux commands</b> to be executed as a regular non-</p>

Category	privileged user Requirements, Conventions or Software Version Used

## Install the Dependencies

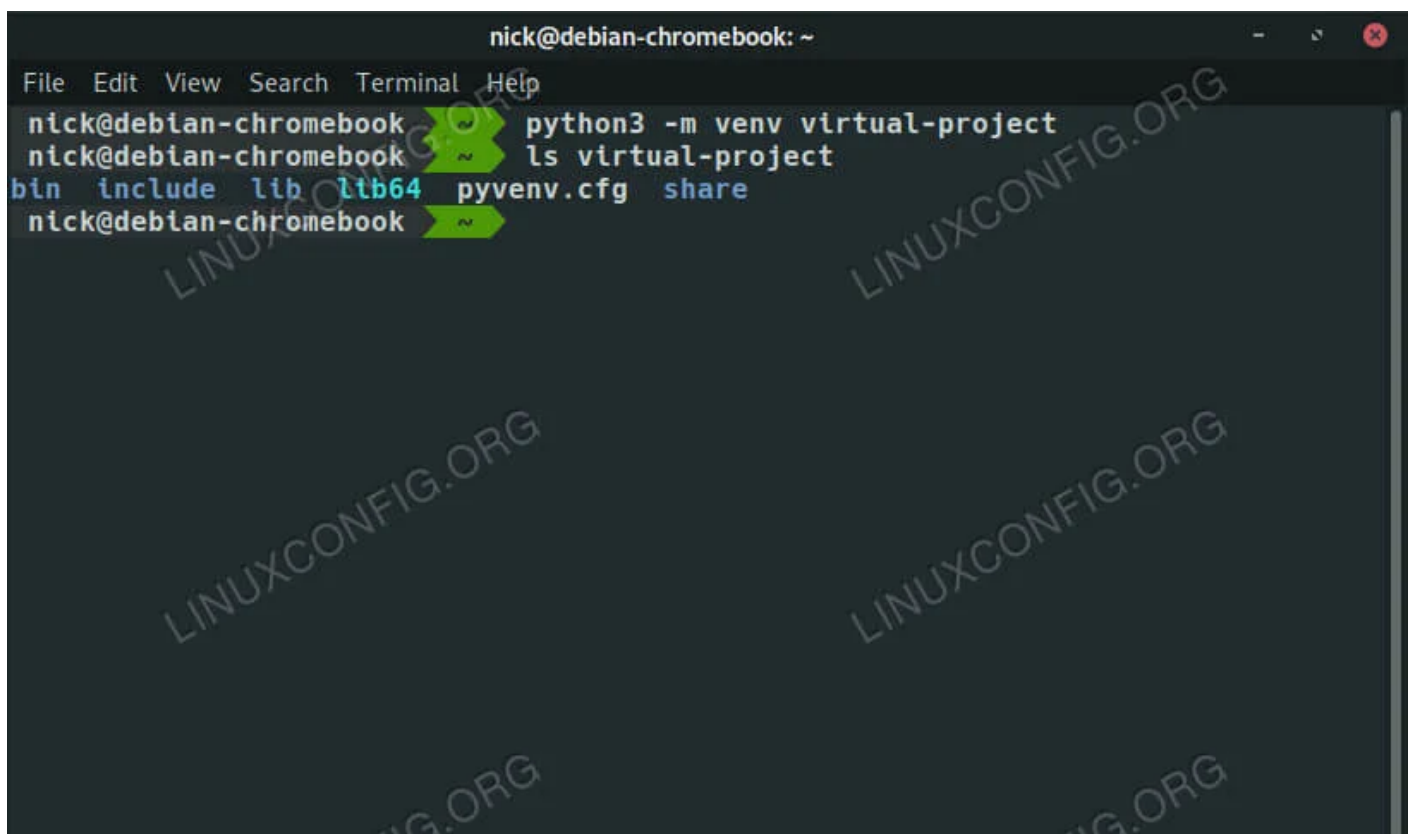
Before you get started, make sure that you have Python 3.

```
$ sudo apt install python3 python3-venv
```

Then, if you plan to use Virtualenv, install that too.

```
$ sudo apt install virtualenv python3-virtualenv
```

## Use Python 3's Venv

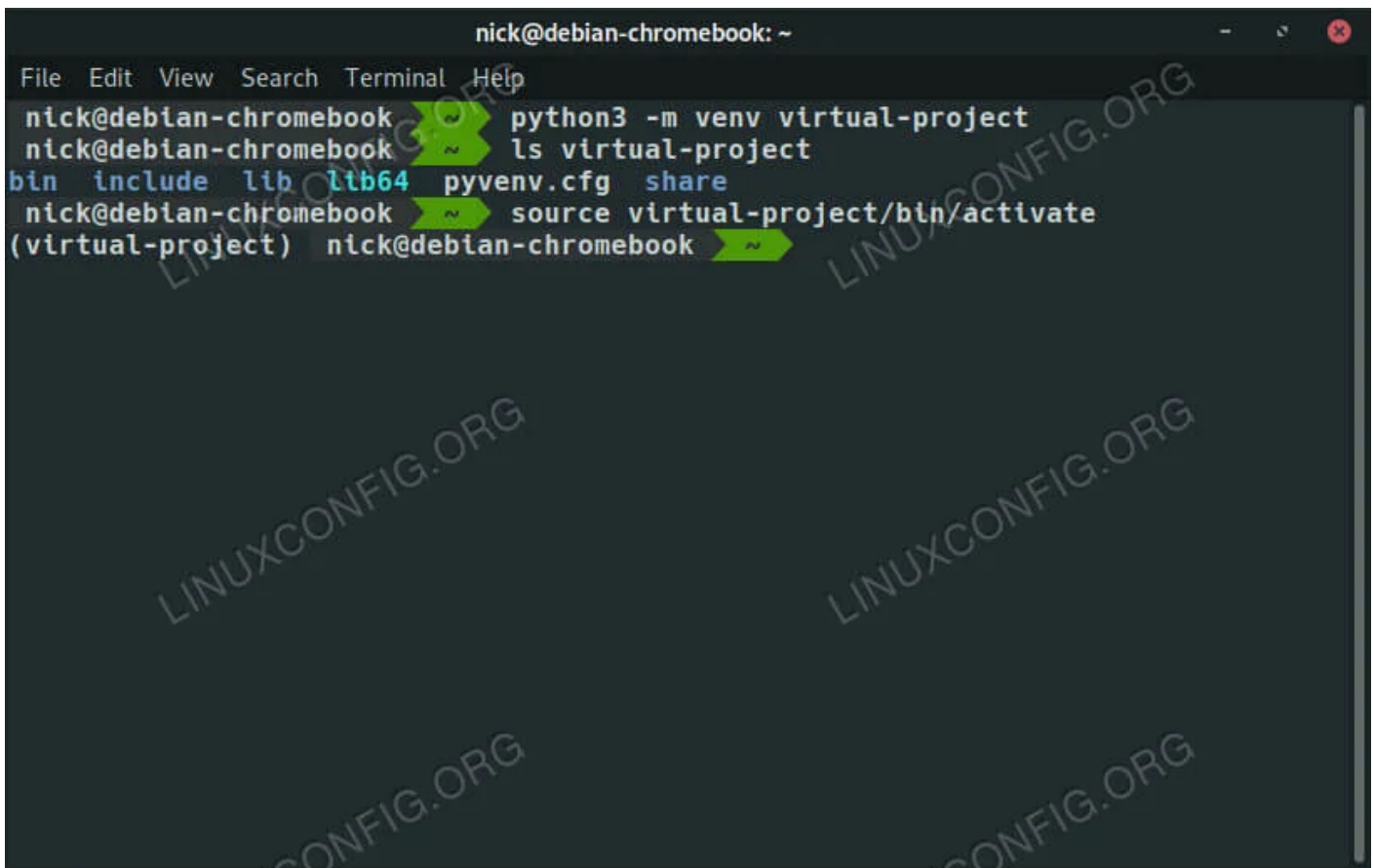


```
nick@debian-chromebook: ~  
File Edit View Search Terminal Help  
nick@debian-chromebook ~$ python3 -m venv virtual-project  
nick@debian-chromebook ~$ ls virtual-project  
bin  include  lib  lib64  pyenv.cfg  share  
nick@debian-chromebook ~$
```

## *Set Up Python Venv on Debian 10.*

Python 3's **venv** functionality is built-in, and you can use it to get set up without anything else.

```
$ python3 -m venv /path/to/virtual/environment
```



A terminal window titled 'nick@debian-chromebook: ~' showing the following commands and output:

```
nick@debian-chromebook ~$ python3 -m venv virtual-project
nick@debian-chromebook ~$ ls virtual-project
bin  include  lib  lib64  pyvenv.cfg  share
nick@debian-chromebook ~$ source virtual-project/bin/activate
(virtual-project) nick@debian-chromebook ~$
```

The terminal output shows the directory structure of the created virtual environment: `bin`, `include`, `lib`, `lib64`, `pyvenv.cfg`, and `share`. The prompt changes from `nick@debian-chromebook` to `(virtual-project)` after activation.

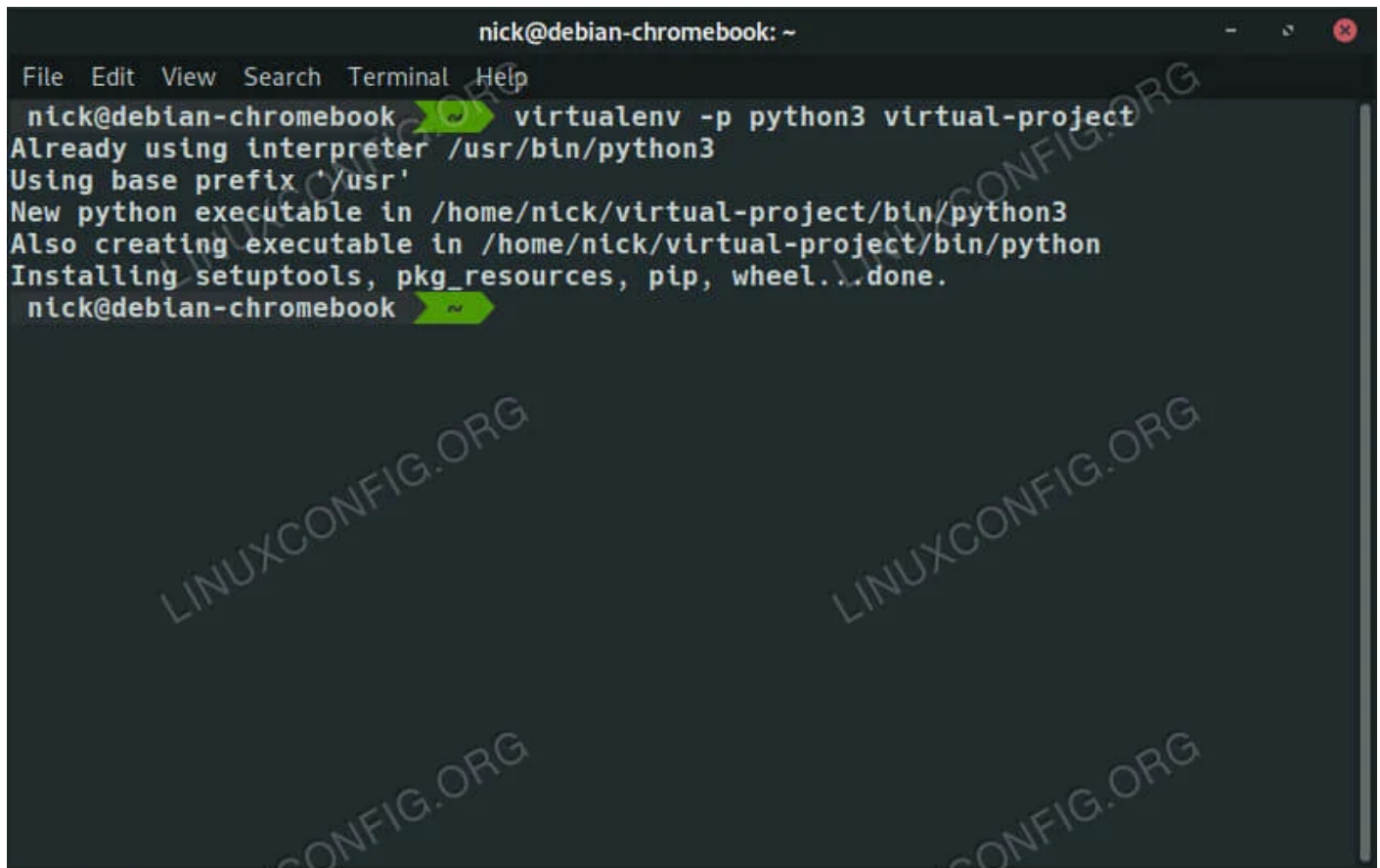
## *Activate Python Venv on Debian 10.*

It will only take a few seconds to get set up. Once it's done, you can activate the virtual environment with:

```
$ source your-project/bin/activate
```

Now, you're working with the Python install from your virtual environment, instead of the system wide one. Anything you do now, should reside in your project folder. When you're done, just run **deactivate** to exit the virtual Python.

## Use Virtualenv

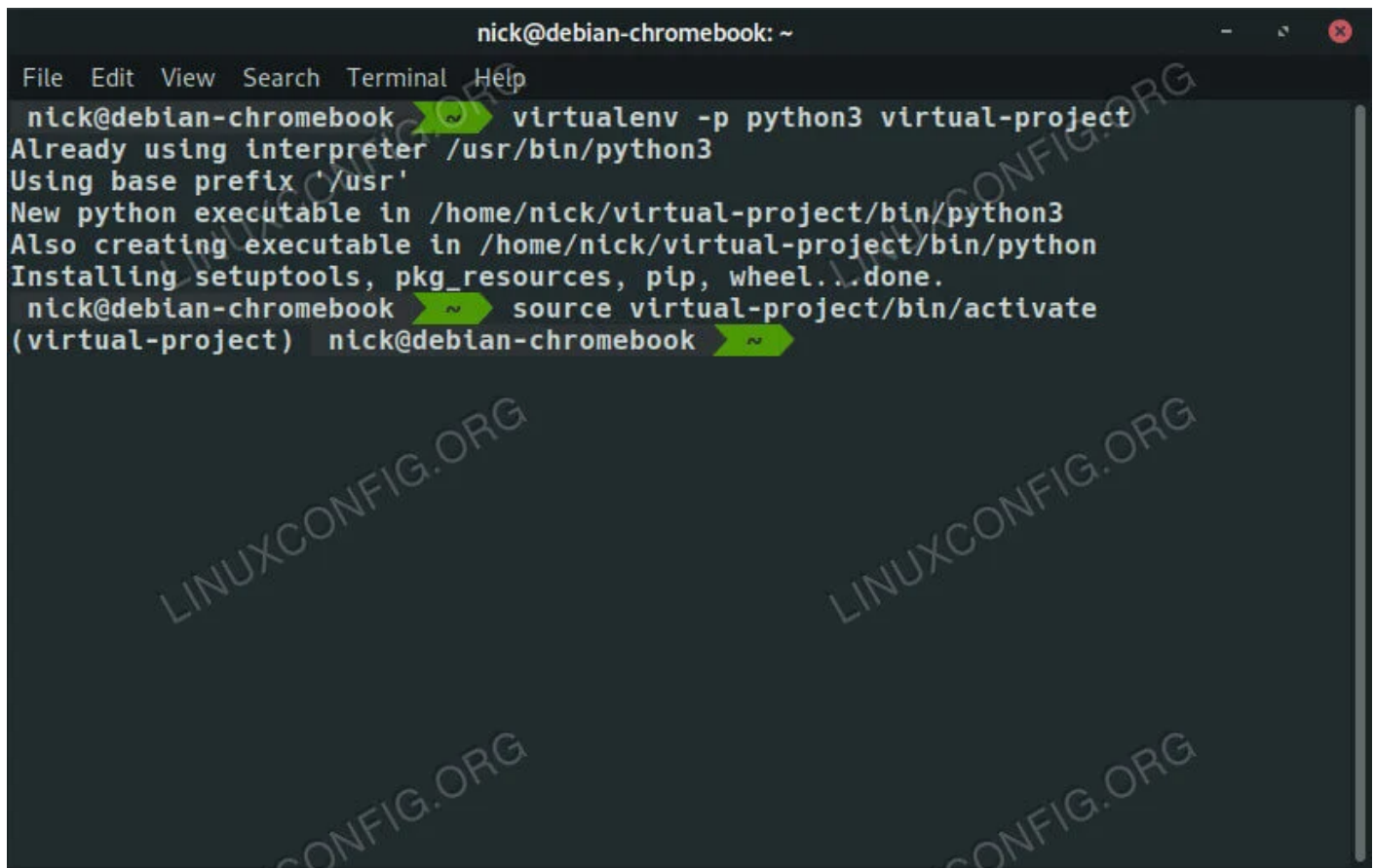
A terminal window titled 'nick@debian-chromebook: ~' showing the execution of the 'virtualenv' command. The command 'virtualenv -p python3 virtual-project' is entered, and the output shows that the environment is already using the system interpreter, but new executables are being created in the project's bin directory. The installation of setuptools, pip, and wheel is also shown as complete.

```
nick@debian-chromebook: ~  
File Edit View Search Terminal Help  
nick@debian-chromebook ➤ virtualenv -p python3 virtual-project  
Already using interpreter /usr/bin/python3  
Using base prefix '/usr'  
New python executable in /home/nick/virtual-project/bin/python3  
Also creating executable in /home/nick/virtual-project/bin/python  
Installing setuptools, pkg_resources, pip, wheel...done.  
nick@debian-chromebook ➤
```

*Create Python Virtualenv on Debian 10.*

To start, create your environment with the **virtualenv** command. You'll also need to tell it to use Python 3 with the **-p** flag.

```
$ virtualenv -p python3 /path/to/virtual/environment
```

A terminal window titled 'nick@debian-chromebook: ~' with a menu bar (File, Edit, View, Search, Terminal, Help). The terminal shows the command 'virtualenv -p python3 virtual-project' being executed. The output indicates that the environment is already using the interpreter '/usr/bin/python3' and uses the base prefix '/usr'. It then creates a new python executable in '/home/nick/virtual-project/bin/python3' and also creates an executable in '/home/nick/virtual-project/bin/python'. It also installs 'setuptools, pkg\_resources, pip, wheel...done.'. The next command is 'source virtual-project/bin/activate', which activates the environment, changing the prompt to '(virtual-project) nick@debian-chromebook'.

```
nick@debian-chromebook: ~
File Edit View Search Terminal Help
nick@debian-chromebook ~$ virtualenv -p python3 virtual-project
Already using interpreter /usr/bin/python3
Using base prefix '/usr'
New python executable in /home/nick/virtual-project/bin/python3
Also creating executable in /home/nick/virtual-project/bin/python
Installing setuptools, pkg_resources, pip, wheel...done.
nick@debian-chromebook ~$ source virtual-project/bin/activate
(virtual-project) nick@debian-chromebook ~$
```

*Activate Python Virtualenv on Debian 10.*

This will take a few seconds to get itself setup with Pip and the other Python packages it includes. When it's finished, activate the environment.

```
$ source your-project/bin/activate
```

Do your work inside the project directories. When you're done, use **deactivate** to exit the virtual environment.

## Conclusion

It's super easy to get set up with Python virtual environments, and the benefits are pretty clear. You'll be able to compartmentalize your projects, and keep things from



conflicting. It's also easier to manage Python package versions as you work.

## Related Linux Tutorials:

- [How to upgrade Debian 9 Stretch to Debian 10 Buster](#)
- [How to run the Raspberry Pi Os in a virtual machine...](#)
- [How to work with the Woocommerce REST API with Python](#)
- [Raspbian GNU/Linux upgrade from Stretch to Raspbian...](#)
- [How to use bridged networking with libvirt and KVM](#)
- [How to Install Nextcloud Client on Debian 10 Buster](#)
- [How to install software in RHEL 8](#)
- [How to install the NVIDIA RTX 3080 driver on Debian...](#)
- [Things to install on Ubuntu 20.04](#)
- [How to install Nvidia driver on Debian 10 Buster Linux](#)

### Debian

- < [How to Set Up a LAMP Server on Debian 10 Buster](#)
- > [How to install VMware Tools on RHEL 8 / CentOS 8](#)

## NEWSLETTER

Subscribe to Linux Career Newsletter to receive latest news, jobs, career advice and featured configuration tutorials.

SUBSCRIBE

## WRITE FOR US

LinuxConfig is looking for a technical writer(s) geared towards GNU/Linux and FLOSS technologies. Your articles will feature various GNU/Linux configuration tutorials and FLOSS technologies used in combination with GNU/Linux operating system.

When writing your articles you will be expected to be able to keep up with a technological advancement regarding the above mentioned technical area of expertise. You will work independently and be able to produce at minimum 2 technical articles a month.

APPLY NOW

## CONTACT US

web ( at ) linuxconfig ( dot ) org

## TAGS

18.04 administration apache applications backup bash beginner browser  
centos centos8 commands database debian desktop development docker fedora  
filesystem firewall gaming gnome Hardware installation java kali manjaro multimedia  
networking nvidia programming python redhat rhel8 scripting security server ssh  
storage terminal ubuntu ubuntu 20.04 video virtualization webapp webserver

## FEATURED TUTORIALS

How to install the NVIDIA drivers on Ubuntu 20.04 Focal Fossa Linux

Bash Scripting Tutorial for Beginners



[How to check CentOS version](#)

[How to find my IP address on Ubuntu 20.04 Focal Fossa Linux](#)

[Ubuntu 20.04 Remote Desktop Access from Windows 10](#)

[Howto mount USB drive in Linux](#)

[How to install missing ifconfig command on Debian Linux](#)

[AMD Radeon Ubuntu 20.04 Driver Installation](#)

[Ubuntu Static IP configuration](#)

[How to use bash array in a shell script](#)

[Linux IP forwarding – How to Disable/Enable](#)

[How to install Tweak Tool on Ubuntu 20.04 LTS Focal Fossa Linux](#)

[How to enable/disable firewall on Ubuntu 18.04 Bionic Beaver Linux](#)

[Netplan static IP on Ubuntu configuration](#)

[How to change from default to alternative Python version on Debian Linux](#)

[Set Kali root password and enable root login](#)

[How to Install Adobe Acrobat Reader on Ubuntu 20.04 Focal Fossa Linux](#)

[How to install the NVIDIA drivers on Ubuntu 18.04 Bionic Beaver Linux](#)

[How to check NVIDIA driver version on your Linux system](#)

[Nvidia RTX 3080 Ethereum Hashrate and Mining Overclock settings on HiveOS Linux](#)

## LATEST TUTORIALS

[How to manage Bash history](#)

[How to install the NVIDIA drivers on Ubuntu 22.04](#)

[How to create a flatpak package](#)

[How to set, change and delete music tags with Mutagen](#)

[Introduction to MySQL storage engines](#)

[Introduction to crypttab with examples](#)

[How to create temporary files using mktemp on Linux](#)

[How to crash Linux](#)

[How to build an initramfs using Dracut on Linux](#)

How to uncompress and list an initramfs content on Linux

How to install PipeWire on Ubuntu Linux

Ubuntu 22.04 kernel version

Ubuntu 22.04 network configuration

Ubuntu 22.04 minimum requirements

Ubuntu 22.04 change hostname

How to install Docker on Ubuntu 22.04

Introduction to Ansible prompts and runtime variables

How to create and extract cpio archives on Linux Examples

How to hash passwords on Linux

How to perform administration operations with Ansible modules

© 2022 TOSID Group Pty Ltd - LinuxConfig.org