THE FOUNDER’S GUIDE:

**How to Install and Manage Multiple Python Versions in WSL2**

The expanded tutorial with concise explanations and screenshots



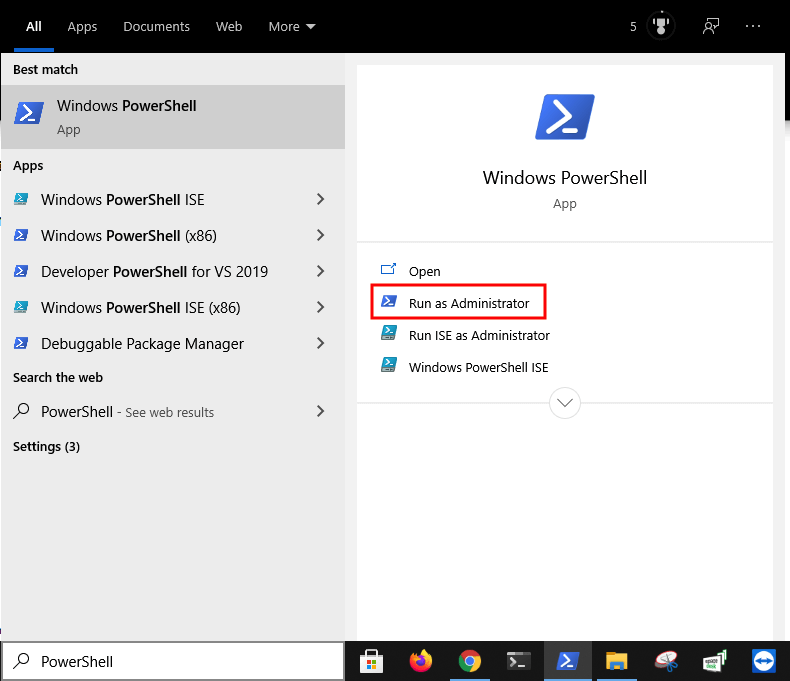
Image by [Isabella and Louisa Fischer](https://unsplash.com/photos/9_5z24t0DFA)

“The [condensed version](https://medium.com/p/35d8eab2e221) of this article uses copy and paste code to help you get the outcome ASAP ⚡”

**Open PowerShell:**

*PowerShell* is a command-line [shell](https://codeburst.io/how-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58#994c) and object-oriented scripting language that’s used to automate administrative tasks and configure system settings. It can be used to automate practically anything in the operating system. It also replaced Command Prompt as the default system shell for Windows 10.

1. Press “⊞ Windows”
2. Enter “PowerShell” into the search bar
3. Click “Run as Administrator”

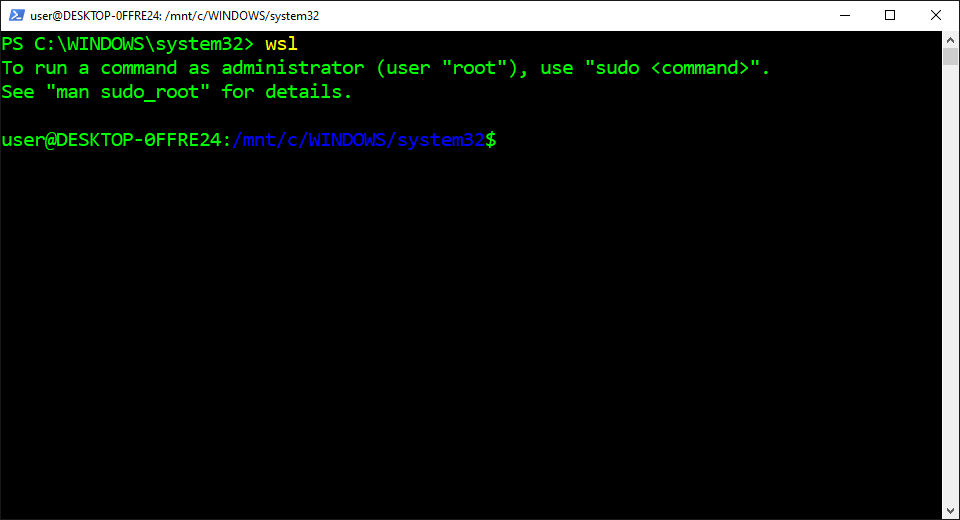


**Open WSL2:**

*Windows Subsystem for Linux 2 (WSL2)* is a Windows 10 feature that allows users run Linux on Windows without using dual-boot or a virtual machine. It has full access to both filesystems, GPU support, and network application support. It also provides access to thousands of Linux command-line tools.

1. Copy the command from below these instructions
2. Paste the command into PowerShell
3. Press “Enter”

wsl

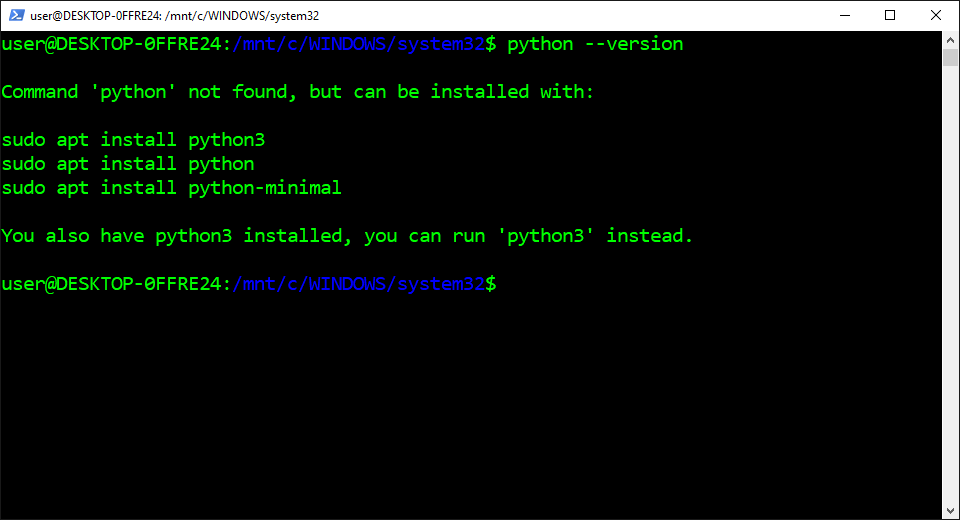


**Check the Default Version:**

The *Version (V)* option is used to check which version of Python is currently selected as the default version. It consists of three numbers separated by periods that represent the [major](https://codeburst.io/how-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58#7870), [minor](https://codeburst.io/how-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58#679b), and [micro](https://codeburst.io/how-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58#ff0b) version number. It also displays an error message if Python isn’t already installed on the computer.

1. Copy the command from below these instructions
2. Paste the command into PowerShell
3. Press “Enter”

python --version

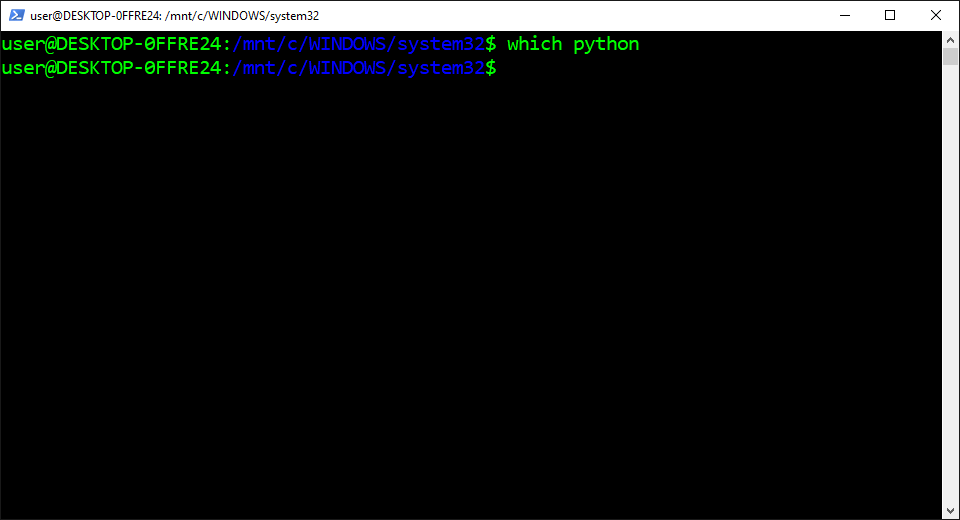


**Check the Executable File:**

The *Which* command is used to identify the location of the executable file for the specified command name. It searches for the executable file in the list of directories from the [PATH](https://codeburst.io/how-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58#b3fe) environment variable. It also displays the absolute path of the executable file if it exists in the one of the directories.

1. Copy the command from below these instructions
2. Paste the command into PowerShell
3. Press “Enter”

which python

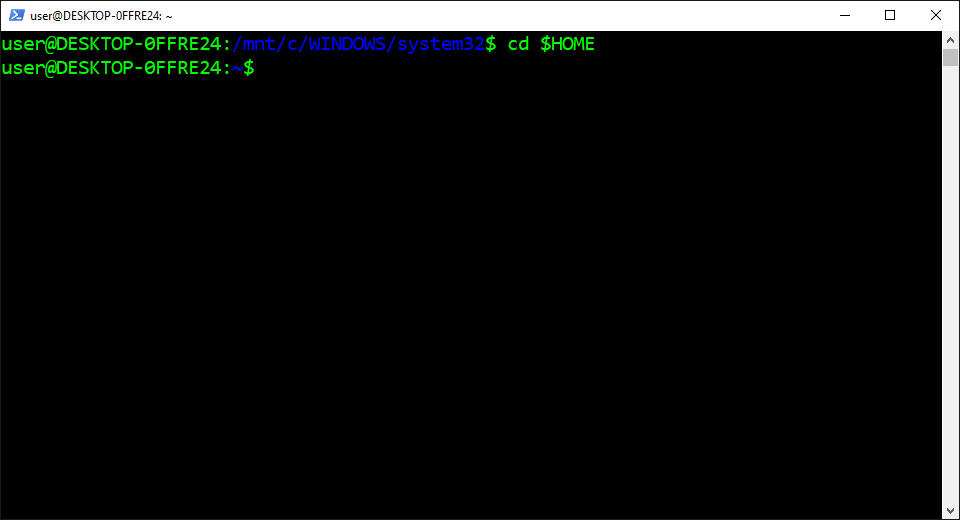


**Open the Home Directory:**

The *Change Directory (cd)* command is used to change the current working directory to the specified directory. It can navigate to absolute and relative paths that start from the root and current working directory, respectively. It can also navigate to paths stored in [variables](https://codeburst.io/how-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58#2b28) and [environment variables](https://codeburst.io/how-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58#9d2c).

1. Copy the command from below these instructions
2. Paste the command into PowerShell
3. Press “Enter”

cd $HOME

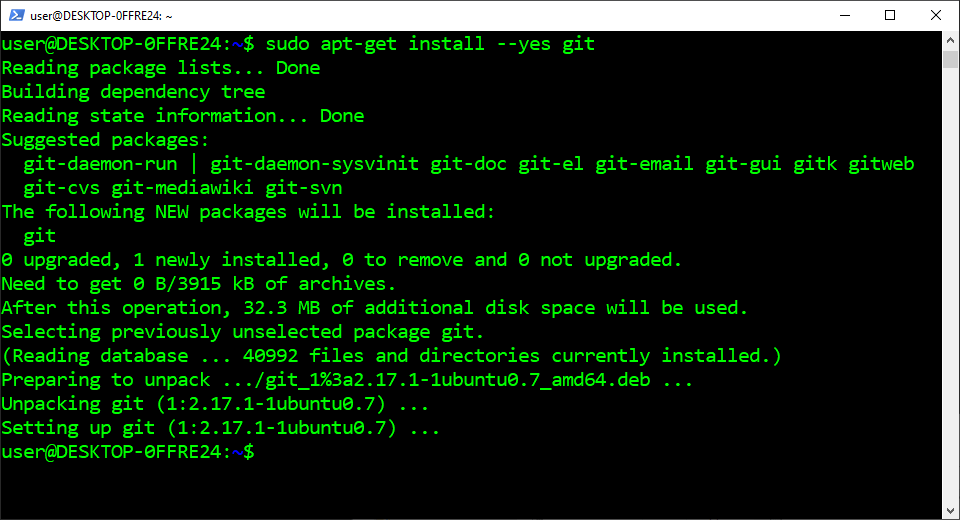


**Install Git:**

*Git* is a program that’s used to track changes that are made to the source code over time. It can handle projects of all sizes and allows multiple teams and people to make changes to the same repository. It can also restore the source code to a previous version from the entire history of the repository.

1. Copy the command from below these instructions
2. Paste the command into PowerShell
3. Press “Enter”

sudo apt-get install --yes git

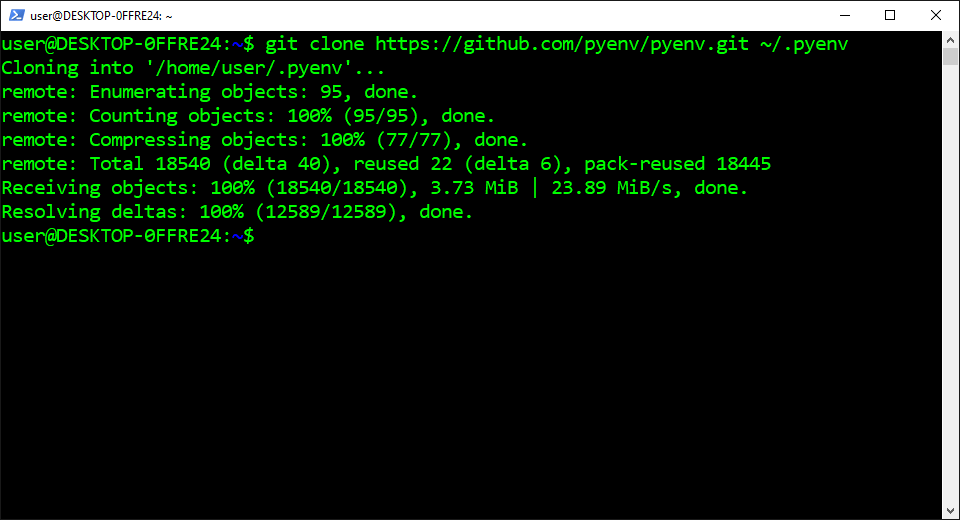


**Clone the Pyenv Repository:**

*Pyenv* is a program that’s used for Python version management on macOS and Linux. It can install multiple Python versions, specify the version that’s used system-wide, and specify the version that’s used in specific directories. It can also create and manage virtual environments using specific versions.

1. Copy the command from below these instructions
2. Paste the command into PowerShell
3. Press “Enter”

git [clone](https://codeburst.io/how-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58#1b3c) https://github.com/pyenv/pyenv.git ~/.pyenv

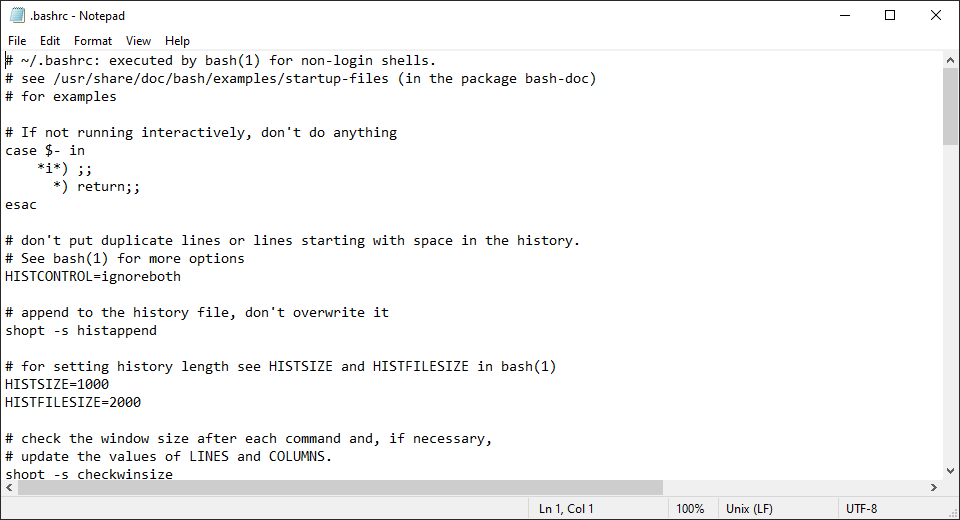


**Open the Bash Configuration File:**

The *Shell Configuration File* is a script that’s automatically executed when a shell is opened by the user. It contains code that’s used to change the look of the shell, run scripts and commands, create aliases, and load environment variables. It also creates a separate configuration file for the different shells.

1. Copy the command from below these instructions
2. Paste the command into PowerShell
3. Press “Enter”

notepad ~/.bashrc

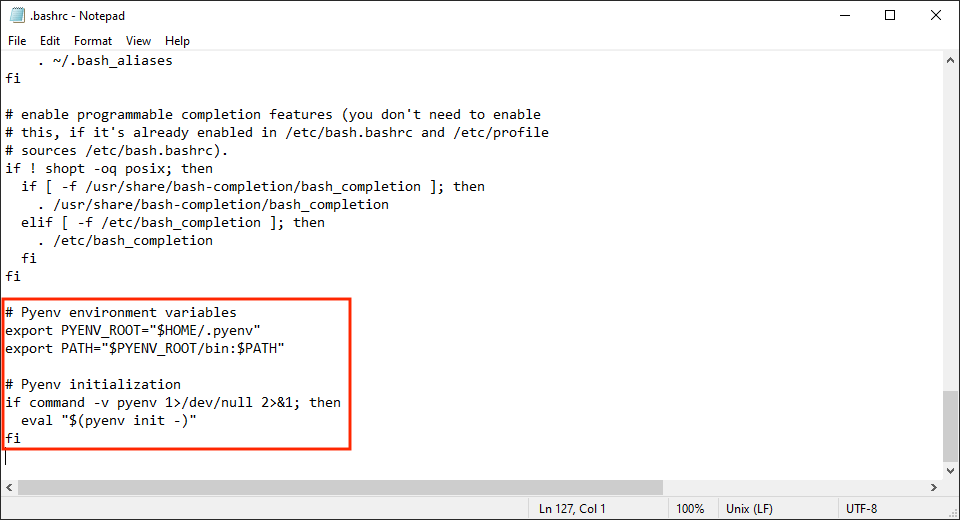


**Edit the Bash Configuration File:**

*Bourne Again Shell (Bash)* is a command-line shell and scripting language that’s used to automate administrative tasks and configure system settings. It can be used to automate practically anything in the operating system. It has also been the default shell for most Linux-based operating systems.

1. Copy the code from below these instructions
2. Paste the code into Notepad
3. Click the “File” menu
4. Click “Save”

# Pyenv environment variables  
export PYENV\_ROOT="$HOME/.pyenv"  
export PATH="$PYENV\_ROOT/bin:$PATH"# Pyenv initialization  
if command -v pyenv 1>/dev/null 2>&1; then  
 eval "$(pyenv init --path)"  
fi

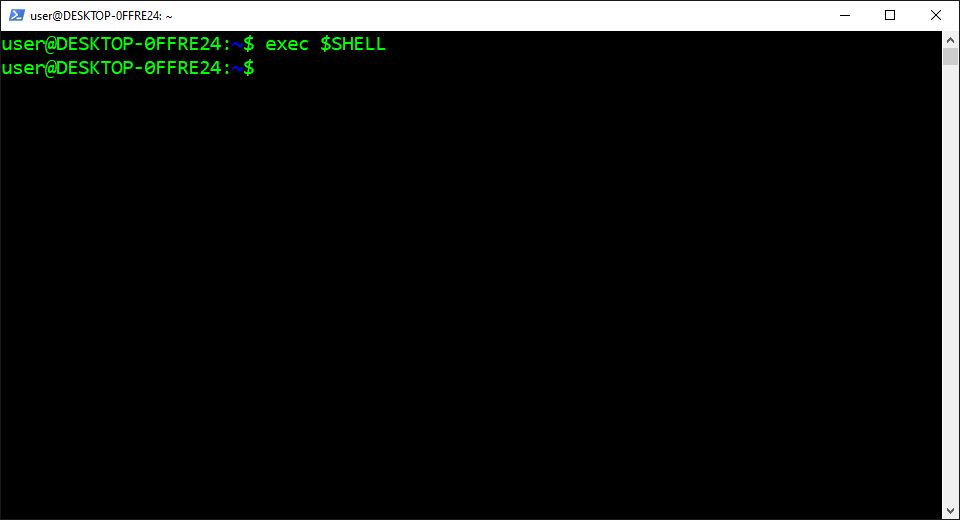


**Restart WSL2:**

The *Exec (e)* command is used to execute the specified command that’s provided as an argument. It destroys the current process and replaces it with the specified command without creating a new process. It can also restart the shell to reload the configuration file into the environment.

1. Copy the command from below these instructions
2. Paste the command into PowerShell
3. Press “Enter”

exec $SHELL

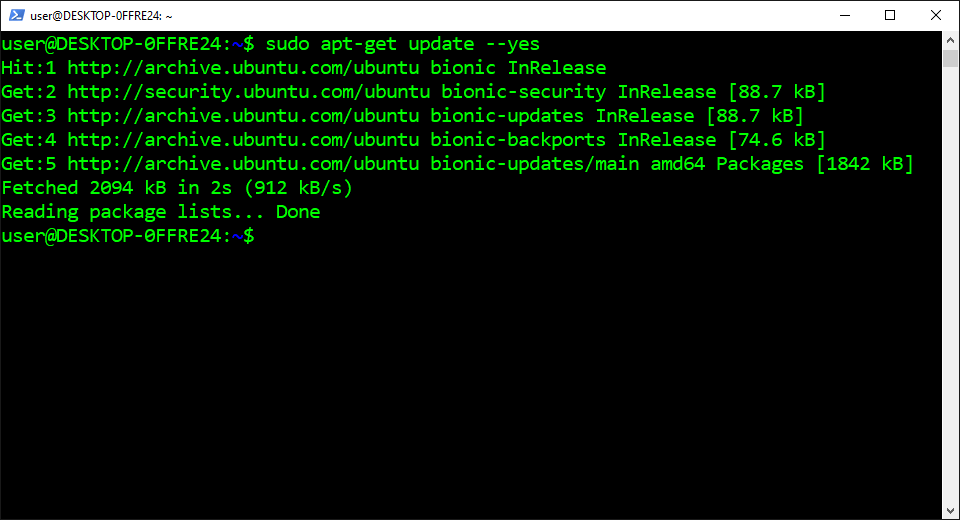


**Update the Source List and Source List Directory:**

The *Update* command is used to ensure the list of available packages is up to date. It downloads a package list from the repositories on the system which contains information about new and upgradable packages. It only updates information about the packages and doesn’t actually upgrade the packages.

1. Copy the command from below these instructions
2. Paste the command into PowerShell
3. Press “Enter”

sudo apt-get update --yes

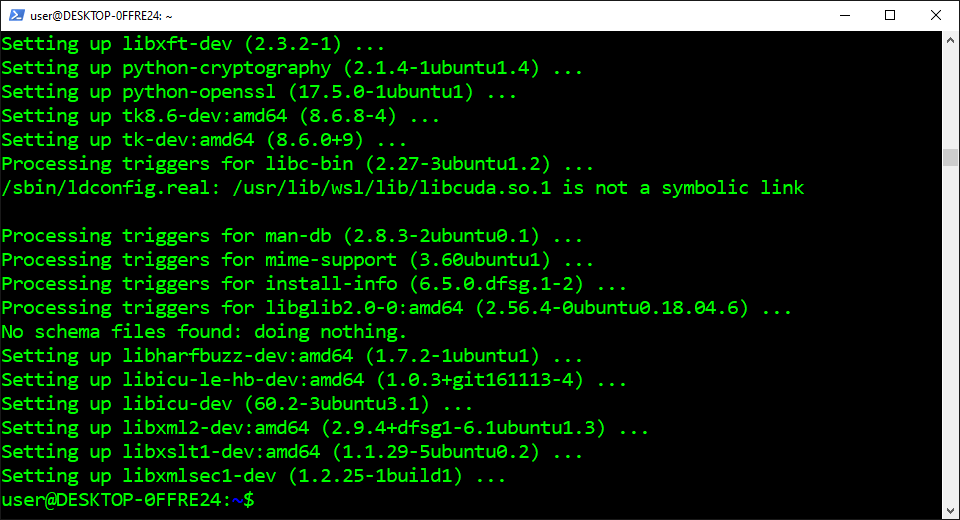


**Install the Pyenv Dependencies:**

The Dependency is an additional binary package that a particular binary package needs to work properly. It can require multiple dependencies to build almost any program that’s distributed by package managers. It also gets downloaded and installed automatically by some package managers.

1. Copy the command from below these instructions
2. Paste the command into PowerShell
3. Press “Enter”

sudo apt-get install --yes libssl-dev zlib1g-dev libbz2-dev libreadline-dev libsqlite3-dev llvm libncurses5-dev libncursesw5-dev xz-utils tk-dev libgdbm-dev lzma lzma-dev tcl-dev libxml2-dev libxmlsec1-dev libffi-dev liblzma-dev wget curl make build-essential python-openssl



**View the Python Versions:**

The *List (l)* flag is used to display the Python versions that are available in Pyenv. It includes final versions that are released from Python, Anaconda, PyPy, Jython, and Stackless. This includes all the major, minor, and micro versions but it doesn’t include alpha, beta, or release candidate versions.

1. Copy the command from below these instructions
2. Paste the command into PowerShell
3. Press “Enter”

pyenv install --list

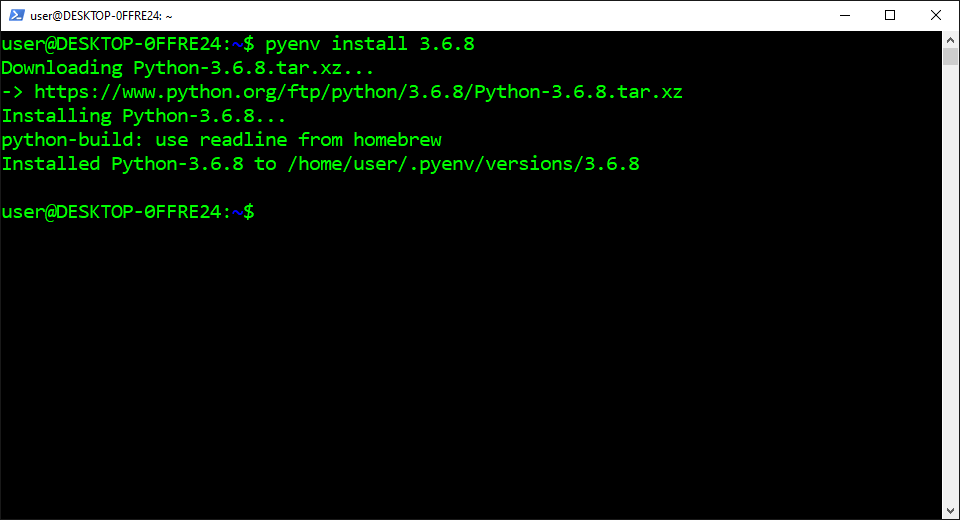


**Install Python:**

*Python* is an object-oriented language that’s known for its simple syntax, code readability, flexibility, and scalability. It mostly gets used to develop web and software applications. It also has become one of the most popular languages for artificial intelligence, machine learning, and data science.

1. Find the version from below these instructions
2. Copy the provided command
3. Paste the command into PowerShell
4. Press “Enter”
5. Repeat

**Python 3.5:**  
pyenv install 3.5.4**Python 3.6:**  
pyenv install 3.6.8**Python 3.7:**  
pyenv install 3.7.9**Python 3.8:**  
pyenv install 3.8.6**Python 3.9:**  
pyenv install 3.9.0

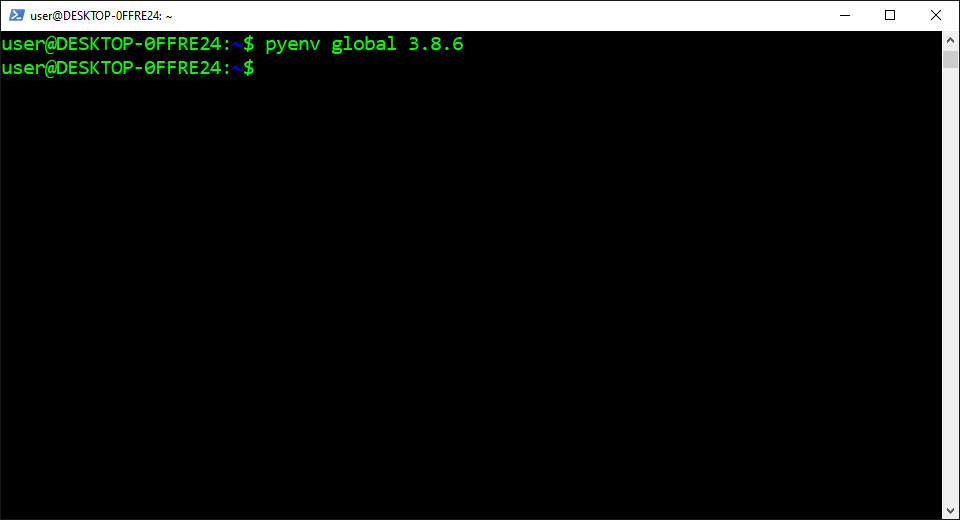


**Set the Default Version for the Computer:**

The *Global* command is used in Pyenv to specify the default Python version for the entire system. It creates a text file in the Pyenv directory that stores the specified version. This is used by Pyenv to activate the default version but it gets overwritten by the local Pyenv text file and environment variable.

1. Copy the command from below these instructions
2. Paste the command into PowerShell
3. Press “Enter”

pyenv global 3.8.6

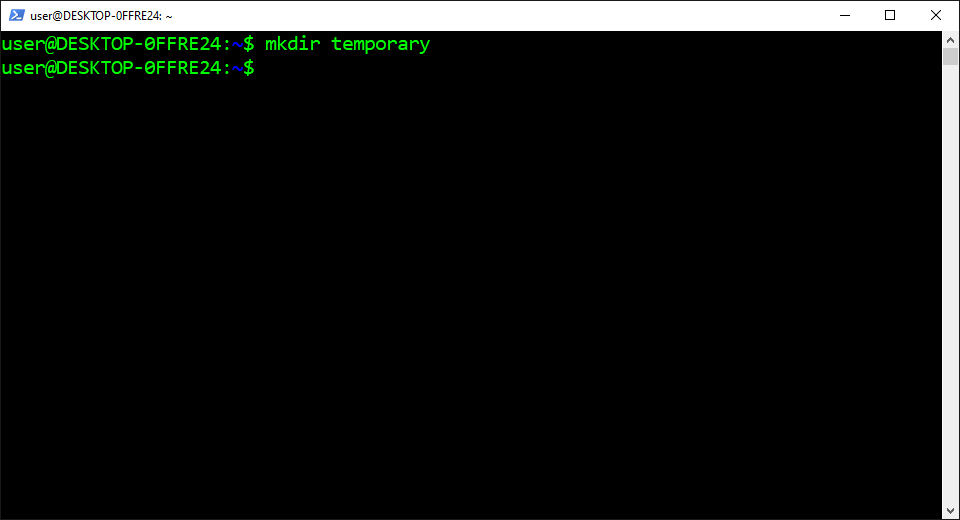


**Create the Temporary Directory:**

The *Make Directory (mkdir)*command is used to create new directories. It can specify one or more relative or absolute paths with the name of the new directories to be created. It can also be used with the “Parents” flag to create parent directories as needed without overwriting a path that already exists.

1. Copy the command from below these instructions
2. Paste the command into PowerShell
3. Press “Enter”

mkdir temporary

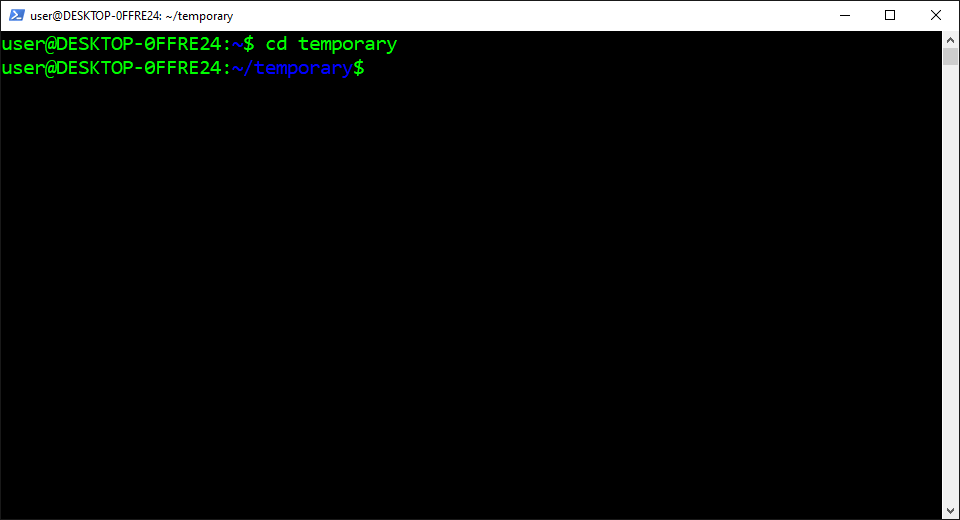


**Open the Temporary Directory:**

The *Change Directory (cd)* command is used to change the current working directory to the specified directory. It can navigate to absolute and relative paths that start from the root and current working directory, respectively. It can also navigate to paths stored in variables and environment variables.

1. Copy the command from below these instructions
2. Paste the command into PowerShell
3. Press “Enter”

cd temporary

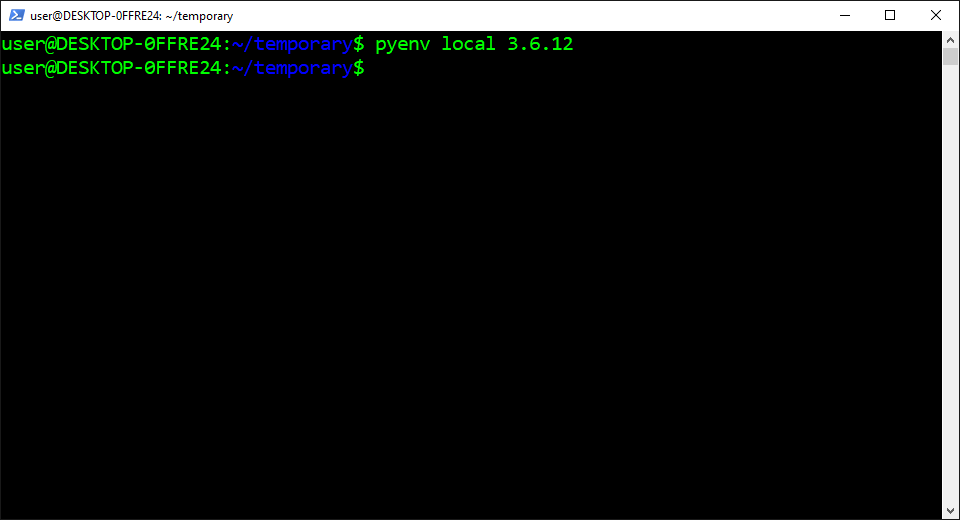


**Set the Default Version for the Directory:**

The *Local* command is used in Pyenv to specify the default Python version for the current directory. It creates a text file in the current directory that stores the specified version. This is automatically detected by Pyenv which activates the Python version in the current directory and subdirectories.

1. Copy the command from below these instructions
2. Paste the command into PowerShell
3. Press “Enter”

pyenv local 3.6.8

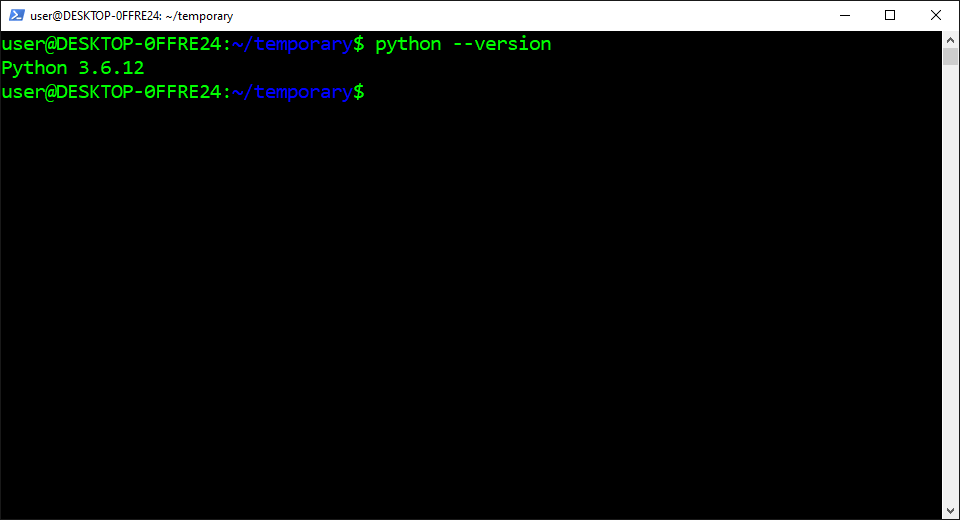


**Check the Default Version:**

The *Version (V)* option is used to check which version of Python is currently selected as the default version. It consists of three numbers separated by periods that represent the major, minor, and micro version number. It also displays an error message if Python isn’t already installed on the computer.

1. Copy the command from below these instructions
2. Paste the command into PowerShell
3. Press “Enter”

python --version

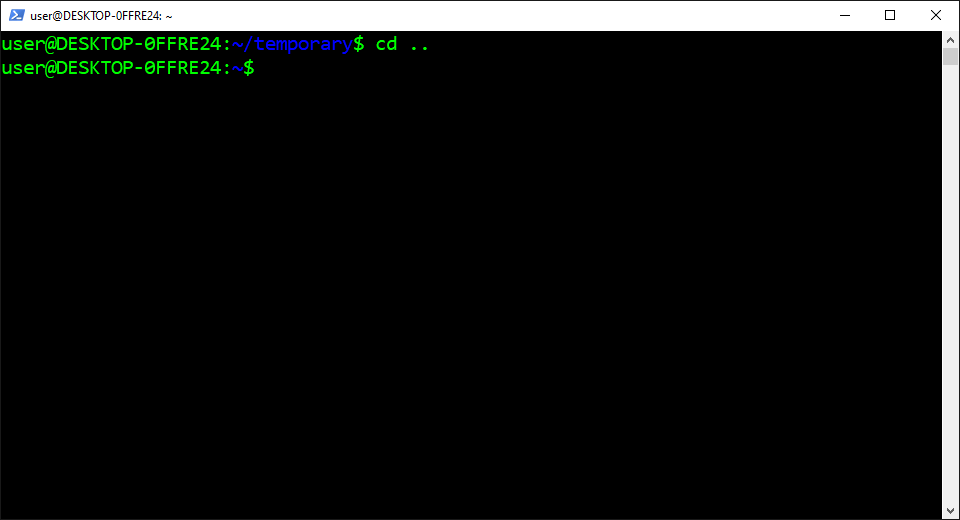


**Open the Parent Directory:**

The *Change Directory (cd)* command is used to change the current working directory to the specified directory. It can navigate to absolute and relative paths that start from the root and current working directory, respectively. It can also navigate to paths stored in variables and environment variables.

1. Copy the command from below these instructions
2. Paste the command into PowerShell
3. Press “Enter”

cd ..

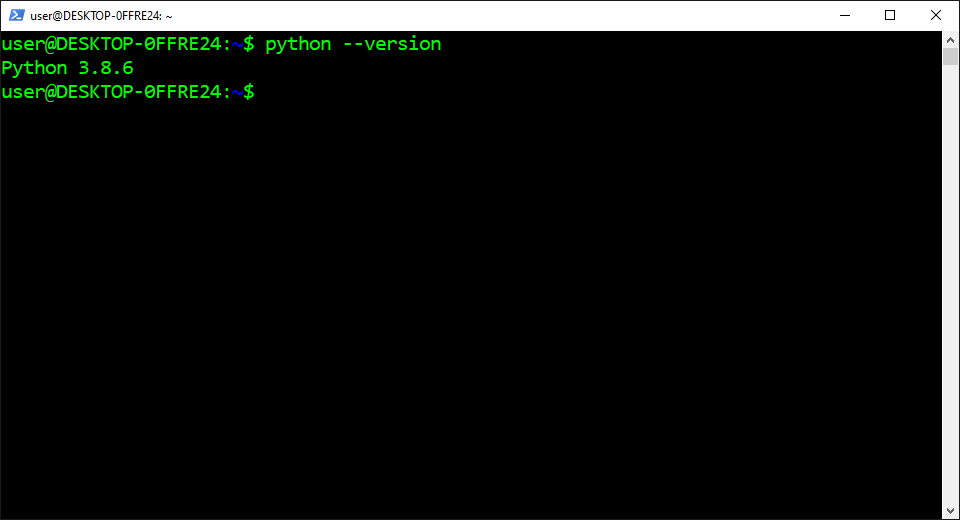


**Check the Default Version:**

The *Version (V)* option is used to check which version of Python is currently selected as the default version. It consists of three numbers separated by periods that represent the major, minor, and micro version number. It also displays an error message if Python isn’t already installed on the computer.

1. Copy the command from below these instructions
2. Paste the command into PowerShell
3. Press “Enter”

python --version

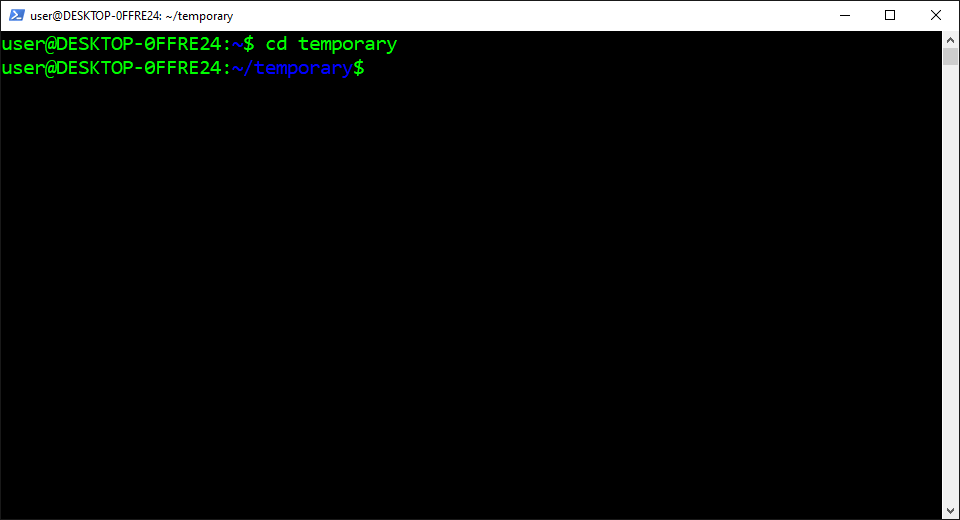


**Open the Temporary Directory:**

The *Change Directory (cd)* command is used to change the current working directory to the specified directory. It can navigate to absolute and relative paths that start from the root and current working directory, respectively. It can also navigate to paths stored in variables and environment variables.

1. Copy the command from below these instructions
2. Paste the command into PowerShell
3. Press “Enter”

cd temporary

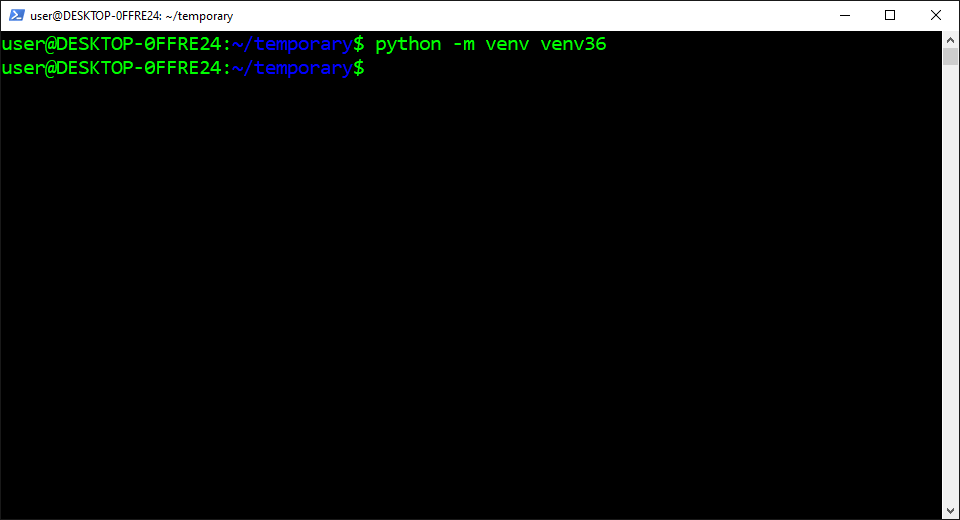


**Create the Virtual Environment:**

The *Virtual Environment* is an isolated Python installation directory that has its own interpreter, site-packages, and scripts. It mostly gets used to prevent version conflicts between dependencies from different projects. It also gets used to meet dependency requirements of different programs from GitHub.

1. Copy the command from below these instructions
2. Paste the command into PowerShell
3. Press “Enter”

python -m venv venv36

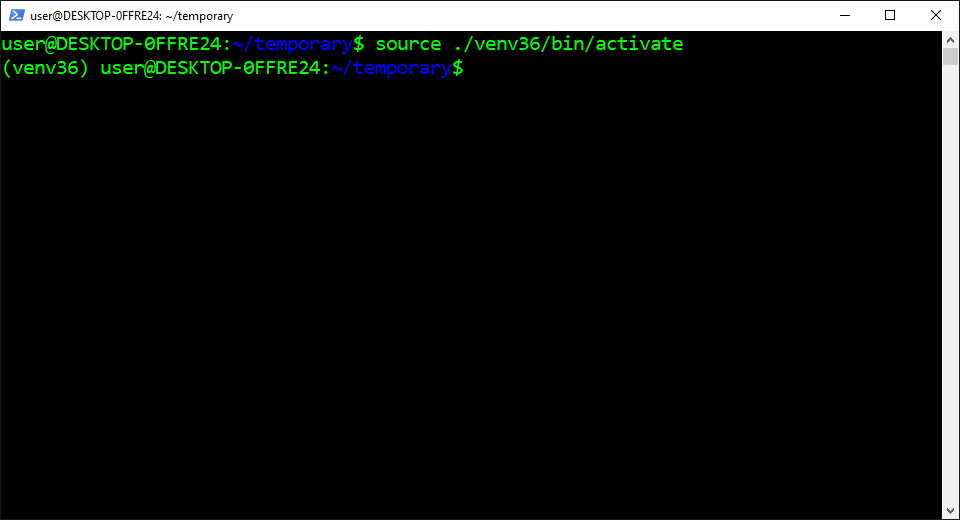


**Activate the Virtual Environment:**

The *Activate* script is used to start the virtual environment. It prepends the virtual environment path to the PATH environment variable which sets the new Python interpreter and package manager as the default version. It also sets packages to install in the virtual environment installation directory.

1. Copy the command from below these instructions
2. Paste the command into PowerShell
3. Press “Enter”

source ./venv36/bin/activate

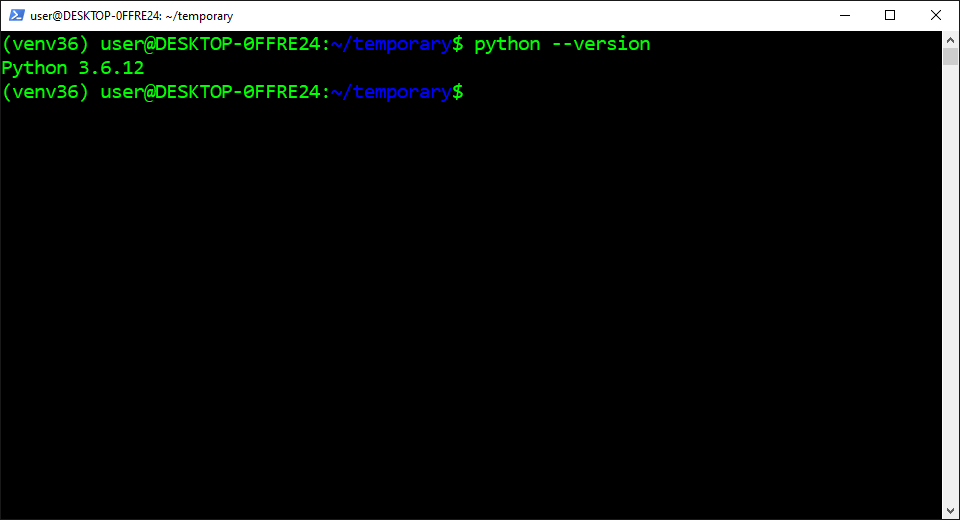


**Check the Default Version:**

The *Version (V)* option is used to check which version of Python is currently selected as the default version. It consists of three numbers separated by periods that represent the major, minor, and micro version number. It also displays an error message if Python isn’t already installed on the computer.

1. Copy the command from below these instructions
2. Paste the command into PowerShell
3. Press “Enter”

python --version

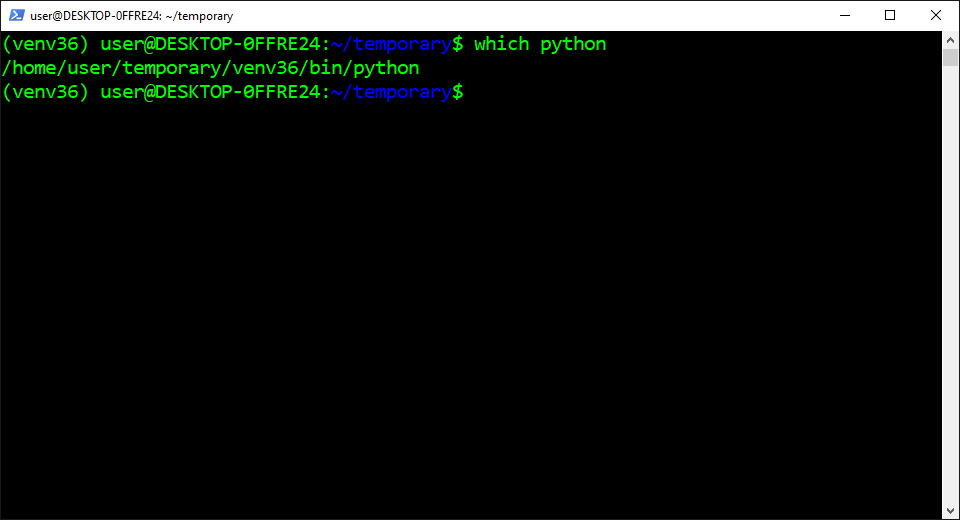


**Check the Executable File:**

The *Which* command is used to identify the location of the executable file for the specified command name. It searches for the executable file in the list of directories from the PATH environment variable. It also displays the absolute path of the executable file if it exists in the one of the directories.

1. Copy the command from below these instructions
2. Paste the command into PowerShell
3. Press “Enter”

which python

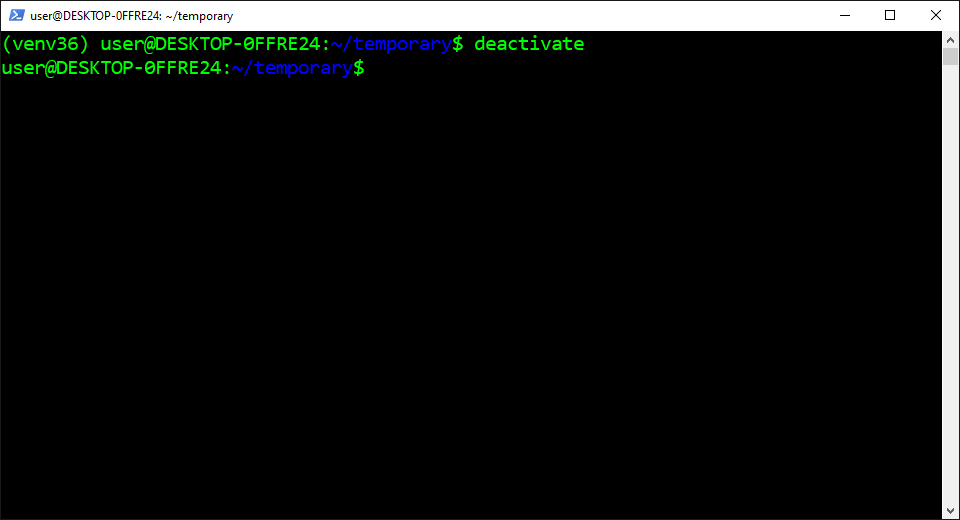


**Deactivate the Virtual Environment:**

The *Deactivate* command is used to stop the virtual environment. It removes the virtual environment path from the PATH environment variable which sets the last Python interpreter and package manager as the default version. It also sets packages to install in the system Python installation directory.

1. Copy the command from below these instructions
2. Paste the command into PowerShell
3. Press “Enter”

deactivate



“Hopefully, this article helped you get the 👯‍♀️🏆👯‍♀️, remember to subscribe to get more content 🏅”

**Next Steps:**

This article is part of a mini-series that helps readers set up everything they need to start learning about artificial intelligence, machine learning, deep learning, and or data science. It includes articles that contain instructions with copy and paste code and screenshots to help readers get the outcome as soon as possible. It also includes articles that contain instructions with explanations and screenshots to help readers learn about what’s happening.

**Linux:**  
01. [Install and Manage Multiple Python Versions](https://medium.com/p/916990dabe4b)  
02. [Install the NVIDIA CUDA Driver, Toolkit, cuDNN, and TensorRT](https://medium.com/p/cd5b3a4f824)  
03. [Install the Jupyter Notebook Server](https://medium.com/p/b2c14c47b446)  
04. [Install Virtual Environments in Jupyter Notebook](https://medium.com/p/1556c8655506)  
05. [Install the Python Environment for AI and Machine Learning](https://medium.com/p/765678fcb4fb)**WSL2:**  
01. [Install Windows Subsystem for Linux 2](https://medium.com/p/cbdd835612fb)  
02. [Install and Manage Multiple Python Versions](https://medium.com/p/1131c4e50a58)  
03. [Install the NVIDIA CUDA Driver, Toolkit, cuDNN, and TensorRT](https://medium.com/p/9800abd74409)   
04. [Install the Jupyter Notebook Server](https://medium.com/p/7c96b3705df1)  
05. [Install Virtual Environments in Jupyter Notebook](https://medium.com/p/3e6bf456041b)  
06. [Install the Python Environment for AI and Machine Learning](https://medium.com/p/612240cb8c0c)  
07. [Install Ubuntu Desktop With a Graphical User Interface](https://medium.com/p/95911ee2997f) (Bonus)**Windows 10:**  
01. [Install and Manage Multiple Python Versions](https://medium.com/p/c90098d7ba5a)  
02. [Install the NVIDIA CUDA Driver, Toolkit, cuDNN, and TensorRT](https://medium.com/p/55febc19b58)  
03. [Install the Jupyter Notebook Server](https://medium.com/p/e8f3e9436044)  
04. [Install Virtual Environments in Jupyter Notebook](https://medium.com/p/5c189856479)  
05. [Install the Python Environment for AI and Machine Learning](https://medium.com/p/23c34b2baf12)**Mac:**01. [Install and Manage Multiple Python Versions](https://medium.com/p/ca01a5e398d4)  
02. [Install the Jupyter Notebook Server](https://medium.com/p/2a276f679e0)  
03. [Install Virtual Environments in Jupyter Notebook](https://medium.com/p/e3de97491b3a)  
04. [Install the Python Environment for AI and Machine Learning](https://medium.com/p/2b2353d7bcc3)

**Glossary:**

The *Shell* is an [interpreter](https://codeburst.io/how-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58#2ba5) that presents the [command-line interface](https://codeburst.io/how-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58#bf45) to users and allows them to interact with the [kernel](https://codeburst.io/how-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58#ed6d). It lets them control the system using commands entered from a keyboard. It also translates the commands from the programming language into the machine language for the kernel.  
[[Return](https://codeburst.io/how-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58#e938)]

The *Interpreter* is a program that reads through instructions that are written in human readable programming languages and executes the instructions from top to bottom. It translates each instruction to a machine language the hardware can understand, executes it, and proceeds to the next instruction.  
[[Return](https://codeburst.io/how-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58#994c)]

The *Command-Line Interface (CLI)*is a program that accepts text input from the user to run commands on the operating system. It lets them configure the system, install software, and access features that aren’t available in the graphical user interface. It also gets referred to as the terminal or console.  
[[Return](https://codeburst.io/how-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58#994c)]

The *Kernel* is the program at the heart of the operating system that controls everything in the computer. It facilitates the memory management, process management, disk management, and task management. It also facilitates communication between the programs and hardware in machine language.  
[[Return](https://codeburst.io/how-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58#994c)]

The *Major* version is a number that represents an update that introduces significant changes to a program. It can add different architecture, features, and functionality that’s incompatible with previous versions. This usually causes programs that were built with a previous version to stop working.  
[[Return](https://codeburst.io/how-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58#a4cd)]

The *Minor* version is a number that represents an update that introduces moderate changes to a program. It can add small features and functionality that’s compatible with the latest major update. This usually doesn’t cause problems with programs that were built with the previous minor updates.  
[[Return](https://codeburst.io/how-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58#a4cd)]

The *Micro* version is a number that represents an update that introduces very modest changes to a program. It can add bug fixes and performance enhancements that are compatible with the previous updates. This doesn’t cause problems with programs that were built with the previous updates.  
[[Return](https://codeburst.io/how-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58#a4cd)]

The *PATH* is an environment variable that contains the list of directories the computer uses to find executable files. It looks for the requested executable file in each directory on the list from top to bottom. It also stops searching once it finds a matching executable file and runs the program or command.  
[[Return](https://codeburst.io/how-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58#5574)]

The *Variable* is the container that’s used to store different types of values. It can assign or update a value by placing an equals sign between the specified variable name and value without a space around it. It can also reference the stored value by placing a dollar sign in front of the existing variable name.  
[[Return](https://codeburst.io/how-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58#9dcf)]

The*Environment Variable* is a variable that’s automatically created and maintained by the computer. It helps the system know where to install files, find programs, and check for user and system settings. It can also be used by graphical and command-line programs from anywhere on the computer.  
[[Return](https://codeburst.io/how-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58#9dcf)]

The *Clone* command is used in Git to download the specified repository from GitHub. It copies all the files, subdirectories, branches, and commits for the entire history of the repository. It can also be combined with the recursive flag to copy the [submodules](https://codeburst.io/how-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58#bf25) the repository uses as dependencies.  
[[Return](https://codeburst.io/how-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58#435c)]

The *Submodule* is a repository that’s nested inside of a different repository as its subdirectory. It contains a copy of all the files for a specific version of the repository at a particular point in time. It also contains a copy of all the submodules within the repository that are considered its dependencies.  
[[Return](https://codeburst.io/how-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58#1b3c)]

155

[More from codeburst](https://codeburst.io/?source=post_page-----1131c4e50a58-----------------------------------)

[Follow](https://medium.com/m/signin?actionUrl=https%3A%2F%2Fmedium.com%2F_%2Fsubscribe%2Fcollection%2Fcodeburst%2F1131c4e50a58&operation=register&redirect=https%3A%2F%2Fcodeburst.io%2Fhow-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58&collection=codeburst&collectionId=61061eb0c96b&source=post_page-----1131c4e50a58---------------------follow_footer--------------)

Bursts of code to power through your day. Web Development articles, tutorials, and news.

The Educative Team

[The Educative Team](https://educative-inc.medium.com/?source=post_page-----1131c4e50a58----0-------------------------------)

[·Jan 26, 2021](https://codeburst.io/learn-javascript-oop-a-primer-for-web-development-e8a8b5710664?source=post_page-----1131c4e50a58----0-------------------------------)

**[Learn JavaScript OOP: A primer for web development](https://codeburst.io/learn-javascript-oop-a-primer-for-web-development-e8a8b5710664?source=post_page-----1131c4e50a58----0-------------------------------)**

[JavaScript may not be what comes to mind when someone mentions an OOP language, but the fact is it has great support for OOP — it just has its intricacies that need to be understood first. If you’re coding in JavaScript, getting familiar with OOP principles can make your life…](https://codeburst.io/learn-javascript-oop-a-primer-for-web-development-e8a8b5710664?source=post_page-----1131c4e50a58----0-------------------------------)

[Programming](https://medium.com/tag/programming?source=post_page-----1131c4e50a58---------------programming--------------------)

[10 min read](https://codeburst.io/learn-javascript-oop-a-primer-for-web-development-e8a8b5710664?source=post_page-----1131c4e50a58----0-------------------------------)

[[](https://codeburst.io/learn-javascript-oop-a-primer-for-web-development-e8a8b5710664?source=post_page-----1131c4e50a58----0-------------------------------)](https://codeburst.io/learn-javascript-oop-a-primer-for-web-development-e8a8b5710664?source=post_page-----1131c4e50a58----0-------------------------------)

Share your ideas with millions of readers.

[Write on Medium](https://medium.com/new-story?source=post_page_footer_cta_write----------------------------------------)

Clyde D'Souza

[Clyde D'Souza](https://clydedz.medium.com/?source=post_page-----1131c4e50a58----1-------------------------------)

[·Jan 26, 2021](https://codeburst.io/what-really-gets-packaged-f2b49397c2a5?source=post_page-----1131c4e50a58----1-------------------------------)

**[What Really Gets Packaged?](https://codeburst.io/what-really-gets-packaged-f2b49397c2a5?source=post_page-----1131c4e50a58----1-------------------------------)**

[Including and excluding files in your npm package Introduction After publishing an article on developing and publishing an npm package using TypeScript, I received a suggestion on Twitter to “create .npmignore in addition to .gitignore or else your dist won’t be included (if dist is in .gitignore).” While this is true…](https://codeburst.io/what-really-gets-packaged-f2b49397c2a5?source=post_page-----1131c4e50a58----1-------------------------------)

[Technology](https://medium.com/tag/technology?source=post_page-----1131c4e50a58---------------technology--------------------)

[4 min read](https://codeburst.io/what-really-gets-packaged-f2b49397c2a5?source=post_page-----1131c4e50a58----1-------------------------------)

[[](https://codeburst.io/what-really-gets-packaged-f2b49397c2a5?source=post_page-----1131c4e50a58----1-------------------------------)](https://codeburst.io/what-really-gets-packaged-f2b49397c2a5?source=post_page-----1131c4e50a58----1-------------------------------)

Kajal Yadav

[Kajal Yadav](https://techykajal.medium.com/?source=post_page-----1131c4e50a58----2-------------------------------)

[·Jan 26, 2021](https://codeburst.io/youtube-to-gif-maker-using-command-line-96c6f9edca0a?source=post_page-----1131c4e50a58----2-------------------------------)

**[Youtube to GIF Maker Using Command Line](https://codeburst.io/youtube-to-gif-maker-using-command-line-96c6f9edca0a?source=post_page-----1131c4e50a58----2-------------------------------)**

[Turn your YouTube video into a GIF using the command line — Introduction A Gif (or Graphical Interchange Format) is a series of images or soundless videos that continuously loop without pressing the play button; such as short scenes from popular TV shows and movies. GIFs are a popular and fun way to communicate on the internet. GIFs can be better than a…](https://codeburst.io/youtube-to-gif-maker-using-command-line-96c6f9edca0a?source=post_page-----1131c4e50a58----2-------------------------------)

[GIF](https://medium.com/tag/gif?source=post_page-----1131c4e50a58---------------gif--------------------)

[5 min read](https://codeburst.io/youtube-to-gif-maker-using-command-line-96c6f9edca0a?source=post_page-----1131c4e50a58----2-------------------------------)

[[](https://codeburst.io/youtube-to-gif-maker-using-command-line-96c6f9edca0a?source=post_page-----1131c4e50a58----2-------------------------------)](https://codeburst.io/youtube-to-gif-maker-using-command-line-96c6f9edca0a?source=post_page-----1131c4e50a58----2-------------------------------)

David Littlefield

[David Littlefield](https://david-littlefield.medium.com/?source=post_page-----1131c4e50a58----3-------------------------------)

[·Jan 26, 2021](https://codeburst.io/how-to-install-virtual-environments-in-jupyter-notebook-in-wsl2-3e6bf456041b?source=post_page-----1131c4e50a58----3-------------------------------)

**[How to Install Virtual Environments in Jupyter Notebook in WSL2](https://codeburst.io/how-to-install-virtual-environments-in-jupyter-notebook-in-wsl2-3e6bf456041b?source=post_page-----1131c4e50a58----3-------------------------------)**

[The expanded tutorial with concise explanations and screenshots — “The condensed version of this article uses copy and paste code to help you get the outcome ASAP ⚡” Open Terminal: The Terminal is an emulator that’s used in Jupyter Notebook to access the file system of the server from the remote computer. It runs on the same computer that’s hosting the…](https://codeburst.io/how-to-install-virtual-environments-in-jupyter-notebook-in-wsl2-3e6bf456041b?source=post_page-----1131c4e50a58----3-------------------------------)

[Data Science](https://medium.com/tag/data-science?source=post_page-----1131c4e50a58---------------data_science--------------------)

[13 min read](https://codeburst.io/how-to-install-virtual-environments-in-jupyter-notebook-in-wsl2-3e6bf456041b?source=post_page-----1131c4e50a58----3-------------------------------)

[[](https://codeburst.io/how-to-install-virtual-environments-in-jupyter-notebook-in-wsl2-3e6bf456041b?source=post_page-----1131c4e50a58----3-------------------------------)](https://codeburst.io/how-to-install-virtual-environments-in-jupyter-notebook-in-wsl2-3e6bf456041b?source=post_page-----1131c4e50a58----3-------------------------------)

David Littlefield

[David Littlefield](https://david-littlefield.medium.com/?source=post_page-----1131c4e50a58----4-------------------------------)

[·Jan 26, 2021](https://codeburst.io/how-to-install-the-nvidia-cuda-driver-toolkit-cudnn-and-tensorrt-in-wsl2-9800abd74409?source=post_page-----1131c4e50a58----4-------------------------------)

**[How to Install the NVIDIA CUDA Driver, Toolkit, cuDNN, and TensorRT in WSL2](https://codeburst.io/how-to-install-the-nvidia-cuda-driver-toolkit-cudnn-and-tensorrt-in-wsl2-9800abd74409?source=post_page-----1131c4e50a58----4-------------------------------)**

[The expanded tutorial with explanations and screenshots — “The condensed version of this article uses copy and paste code to help you get the outcome ASAP ⚡” Join the NVIDIA Developer Program: The NVIDIA Developer Program is a free program that gives members access to the NVIDIA software development kits, tools, resources, and trainings. It lets members submit issues and feature requests to…](https://codeburst.io/how-to-install-the-nvidia-cuda-driver-toolkit-cudnn-and-tensorrt-in-wsl2-9800abd74409?source=post_page-----1131c4e50a58----4-------------------------------)

[Data Science](https://medium.com/tag/data-science?source=post_page-----1131c4e50a58---------------data_science--------------------)

[10 min read](https://codeburst.io/how-to-install-the-nvidia-cuda-driver-toolkit-cudnn-and-tensorrt-in-wsl2-9800abd74409?source=post_page-----1131c4e50a58----4-------------------------------)

[[](https://codeburst.io/how-to-install-the-nvidia-cuda-driver-toolkit-cudnn-and-tensorrt-in-wsl2-9800abd74409?source=post_page-----1131c4e50a58----4-------------------------------)](https://codeburst.io/how-to-install-the-nvidia-cuda-driver-toolkit-cudnn-and-tensorrt-in-wsl2-9800abd74409?source=post_page-----1131c4e50a58----4-------------------------------)

[Read more from codeburst](https://codeburst.io/?source=post_page-----1131c4e50a58-----------------------------------)

More from Medium

[[](https://prateek-ashtikar512.medium.com/spring-batch-call-existing-service-a48ab49b7f3f?source=post_internal_links---------0-------------------------------)](https://prateek-ashtikar512.medium.com/spring-batch-call-existing-service-a48ab49b7f3f?source=post_internal_links---------0-------------------------------)

[Spring Batch — call existing Service](https://prateek-ashtikar512.medium.com/spring-batch-call-existing-service-a48ab49b7f3f?source=post_internal_links---------0-------------------------------)

[On this example, we’ll see how to call the existing service and get its data in the ItemReader and process it.](https://prateek-ashtikar512.medium.com/spring-batch-call-existing-service-a48ab49b7f3f?source=post_internal_links---------0-------------------------------)

[[](https://anuja-shukla.medium.com/build-your-own-alexa-using-python-febc71eda4c7?source=post_internal_links---------1-------------------------------)](https://anuja-shukla.medium.com/build-your-own-alexa-using-python-febc71eda4c7?source=post_internal_links---------1-------------------------------)

[Build Your Own Alexa using Python](https://anuja-shukla.medium.com/build-your-own-alexa-using-python-febc71eda4c7?source=post_internal_links---------1-------------------------------)

[Have you ever thought of having your personal assistant just like Alexa and YOU can build it on your own!!!](https://anuja-shukla.medium.com/build-your-own-alexa-using-python-febc71eda4c7?source=post_internal_links---------1-------------------------------)

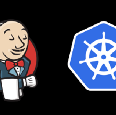
[[](https://jayden-chua.medium.com/which-python-string-formatting-to-use-bbf097aff671?source=post_internal_links---------2-------------------------------)](https://jayden-chua.medium.com/which-python-string-formatting-to-use-bbf097aff671?source=post_internal_links---------2-------------------------------)

[Which Python String Formatting to use?](https://jayden-chua.medium.com/which-python-string-formatting-to-use-bbf097aff671?source=post_internal_links---------2-------------------------------)

[Here are my go-to steps to decide which python string formatting to use.](https://jayden-chua.medium.com/which-python-string-formatting-to-use-bbf097aff671?source=post_internal_links---------2-------------------------------)

[[](https://dataninja.medium.com/build-scalable-data-pipelines-with-apache-spark-b4e7f71fd10e?source=post_internal_links---------3-------------------------------)](https://dataninja.medium.com/build-scalable-data-pipelines-with-apache-spark-b4e7f71fd10e?source=post_internal_links---------3-------------------------------)

[Build Scalable Data Pipelines with Apache Spark™](https://dataninja.medium.com/build-scalable-data-pipelines-with-apache-spark-b4e7f71fd10e?source=post_internal_links---------3-------------------------------)

[[](https://faun.pub/jenkins-inside-kubernetes-bdee0340e0ca?source=post_internal_links---------4-------------------------------)](https://faun.pub/jenkins-inside-kubernetes-bdee0340e0ca?source=post_internal_links---------4-------------------------------)

[Jenkins inside Kubernetes](https://faun.pub/jenkins-inside-kubernetes-bdee0340e0ca?source=post_internal_links---------4-------------------------------)

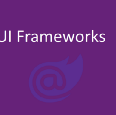
[Running Jenkins with Persistent volumes inside Kubernetes.](https://faun.pub/jenkins-inside-kubernetes-bdee0340e0ca?source=post_internal_links---------4-------------------------------)

[Gravium Masternode on a Raspberry PI3](https://medium.com/@sidewinder/gravium-masternode-on-a-raspberry-pi3-fd0648e760a4?source=post_internal_links---------5-------------------------------)

[This guide will show you how to run a gravium masternode on a raspberry pi3, it is not the most efficient hosting compared to a cheap VPS…](https://medium.com/@sidewinder/gravium-masternode-on-a-raspberry-pi3-fd0648e760a4?source=post_internal_links---------5-------------------------------)

[[](https://expertise.jetruby.com/project-management-made-simple-agile-vs-waterfall-d7a21eab12a6?source=post_internal_links---------6-------------------------------)](https://expertise.jetruby.com/project-management-made-simple-agile-vs-waterfall-d7a21eab12a6?source=post_internal_links---------6-------------------------------)

[Project Management Made Simple: Agile vs Waterfall](https://expertise.jetruby.com/project-management-made-simple-agile-vs-waterfall-d7a21eab12a6?source=post_internal_links---------6-------------------------------)

[[](https://medium.com/@alexandre.malavasi/top-10-nice-free-blazor-components-b42875e56b28?source=post_internal_links---------7-------------------------------)](https://medium.com/@alexandre.malavasi/top-10-nice-free-blazor-components-b42875e56b28?source=post_internal_links---------7-------------------------------)

[Top 10 nice free Blazor components](https://medium.com/@alexandre.malavasi/top-10-nice-free-blazor-components-b42875e56b28?source=post_internal_links---------7-------------------------------)

[I’ve been watching, testing, using and verifying open-source projects related to Blazor over the last three years and I compiled a list in…](https://medium.com/@alexandre.malavasi/top-10-nice-free-blazor-components-b42875e56b28?source=post_internal_links---------7-------------------------------)

[Get started](https://medium.com/m/signin?operation=register&redirect=https%3A%2F%2Fcodeburst.io%2Fhow-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58&source=--------------------------nav_reg--------------)

[Sign In](https://medium.com/m/signin?operation=login&redirect=https%3A%2F%2Fcodeburst.io%2Fhow-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58&source=--------------------------nav_reg--------------)



[[](https://david-littlefield.medium.com/)](https://david-littlefield.medium.com/)

[David Littlefield](https://david-littlefield.medium.com/)

326 Followers

From: Non-Technical | To: Technical | Writes: To Make It Easier For Everyone | Topics: #Setup #Basics #AI #MachineLearning #DeepLearning #Repositories

[Follow](https://medium.com/m/signin?actionUrl=https%3A%2F%2Fmedium.com%2F_%2Fsubscribe%2Fuser%2Fb07fc33d437e&operation=register&redirect=https%3A%2F%2Fcodeburst.io%2Fhow-to-install-and-manage-multiple-python-versions-in-wsl2-1131c4e50a58&user=David+Littlefield&userId=b07fc33d437e&source=publisher_header_actions-b07fc33d437e-------------------------follow_profile--------------)

Related

[[](https://python.plainenglish.io/disaster-tweets-identification-model-with-tensorflow-ff92f4347831?source=read_next_recirc---------0---------------------9bd23654_f1c5_41ee_9f0c_3b8e747f7250----------)](https://python.plainenglish.io/disaster-tweets-identification-model-with-tensorflow-ff92f4347831?source=read_next_recirc---------0---------------------9bd23654_f1c5_41ee_9f0c_3b8e747f7250----------)

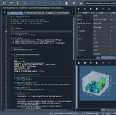
[Disaster Tweets Identification Model with TensorFlow](https://python.plainenglish.io/disaster-tweets-identification-model-with-tensorflow-ff92f4347831?source=read_next_recirc---------0---------------------9bd23654_f1c5_41ee_9f0c_3b8e747f7250----------)

[[](https://levelup.gitconnected.com/explaining-error-detection-and-correction-codes-with-python-be517596d42f?source=read_next_recirc---------1---------------------9bd23654_f1c5_41ee_9f0c_3b8e747f7250----------)](https://levelup.gitconnected.com/explaining-error-detection-and-correction-codes-with-python-be517596d42f?source=read_next_recirc---------1---------------------9bd23654_f1c5_41ee_9f0c_3b8e747f7250----------)

[Explaining error detection and correction codes with Python](https://levelup.gitconnected.com/explaining-error-detection-and-correction-codes-with-python-be517596d42f?source=read_next_recirc---------1---------------------9bd23654_f1c5_41ee_9f0c_3b8e747f7250----------)

[[](https://medium.com/geekculture/how-to-install-and-manage-multiple-python-versions-in-wsl2-6d6ce1b6f55b?source=read_next_recirc---------2---------------------9bd23654_f1c5_41ee_9f0c_3b8e747f7250----------)](https://medium.com/geekculture/how-to-install-and-manage-multiple-python-versions-in-wsl2-6d6ce1b6f55b?source=read_next_recirc---------2---------------------9bd23654_f1c5_41ee_9f0c_3b8e747f7250----------)

[How to Install and Manage Multiple Python Versions in WSL2](https://medium.com/geekculture/how-to-install-and-manage-multiple-python-versions-in-wsl2-6d6ce1b6f55b?source=read_next_recirc---------2---------------------9bd23654_f1c5_41ee_9f0c_3b8e747f7250----------)

[[](https://pallawi-ds.medium.com/how-to-run-spyder-with-anaconda-on-macos-7744ee60d3b9?source=read_next_recirc---------3---------------------9bd23654_f1c5_41ee_9f0c_3b8e747f7250----------)](https://pallawi-ds.medium.com/how-to-run-spyder-with-anaconda-on-macos-7744ee60d3b9?source=read_next_recirc---------3---------------------9bd23654_f1c5_41ee_9f0c_3b8e747f7250----------)

[How to run Spyder with anaconda on macOS](https://pallawi-ds.medium.com/how-to-run-spyder-with-anaconda-on-macos-7744ee60d3b9?source=read_next_recirc---------3---------------------9bd23654_f1c5_41ee_9f0c_3b8e747f7250----------)

[Introduction](https://pallawi-ds.medium.com/how-to-run-spyder-with-anaconda-on-macos-7744ee60d3b9?source=read_next_recirc---------3---------------------9bd23654_f1c5_41ee_9f0c_3b8e747f7250----------)

[Help](https://help.medium.com/hc/en-us)

[Status](https://medium.statuspage.io/)

[Writers](https://about.medium.com/creators/)

[Blog](https://blog.medium.com/)

[Careers](https://medium.com/jobs-at-medium/work-at-medium-959d1a85284e)

[Privacy](https://policy.medium.com/medium-privacy-policy-f03bf92035c9)

[Terms](https://policy.medium.com/medium-terms-of-service-9db0094a1e0f)

[About](https://medium.com/about?autoplay=1)

[Knowable](https://knowable.fyi/)