

The background features a central, irregular watercolor shape in shades of blue and teal. Surrounding this are other watercolor shapes in light blue, green, and yellow. Scattered throughout are small, dark blue dots of varying sizes. At the bottom center, there are faint, stylized line drawings of what appear to be fingers or a hand.

Python: Package Management



Edwin Kofler

[Website](#)

[GitHub](#)

[Twitter](#)

[LinkedIn](#)

Contents

01

Why?

Why use a package manager

02

What?

What is a package manager exactly?

03

Poetry

Poetry: A Python Package Manager

04

Installing Python

How to install Python

05

Installing Poetry

How to install Poetry

06

Using datascience

How to use the datascience package

The background features a large, irregular watercolor shape in shades of blue and green. The top portion is a darker blue, while the bottom portion is a lighter green. Scattered around this central shape are numerous small, dark blue dots. A thin, dark blue line curves across the right side of the image, starting from the top and ending near the bottom right. In the bottom right corner, there is a small, stylized drawing of a hand holding a pen, with the pen tip pointing towards the central text area.

01 & 02

Why/What?

Why Package Managers?

Package managers make it easier to use other people's software

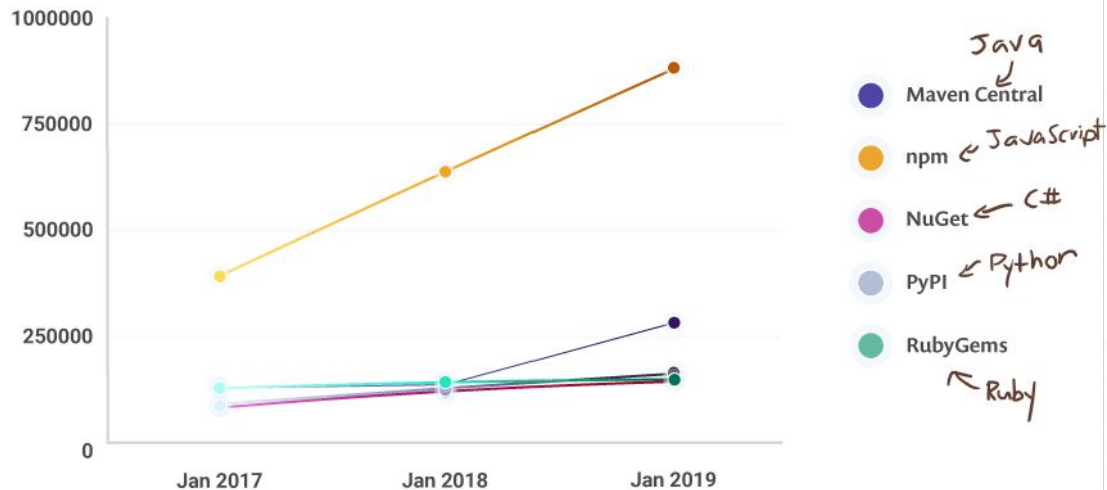
- a. Instead of going to a website, downloading an installer, running the installer / installation wizard, you can do everything in one step
1. Terminology
 - a. *package*
 - i. A reusable piece of software (think of a .zip file that stores a bunch of Python files)
 - b. *package repository / package registry / package index*
 - i. A location or website that stores packages
 - c. *package manager*
 - i. A program that downloads packages from a *package repository*

Analogy:

- Costco is like a *package repository*. The *package manager* is like the shopper looking for and buying various products. And *packages* are like the various products

Big Graph

Total packages indexed per ecosystem



The background features a large, irregular watercolor wash in shades of light blue and green. The blue wash is at the top, and the green wash is at the bottom. Scattered throughout the background are numerous small, dark blue dots of varying sizes. A thin, dark blue line curves across the right side of the image, starting from the top and ending near the bottom right. In the bottom right corner, there is a small, stylized drawing of a hand holding a pen, with the pen tip pointing towards the green wash.

03

Poetry



Poetry

- [Official website](#)
- [GitHub](#)
- First released in late 2018
- Very popular and stable

Package manager for Python

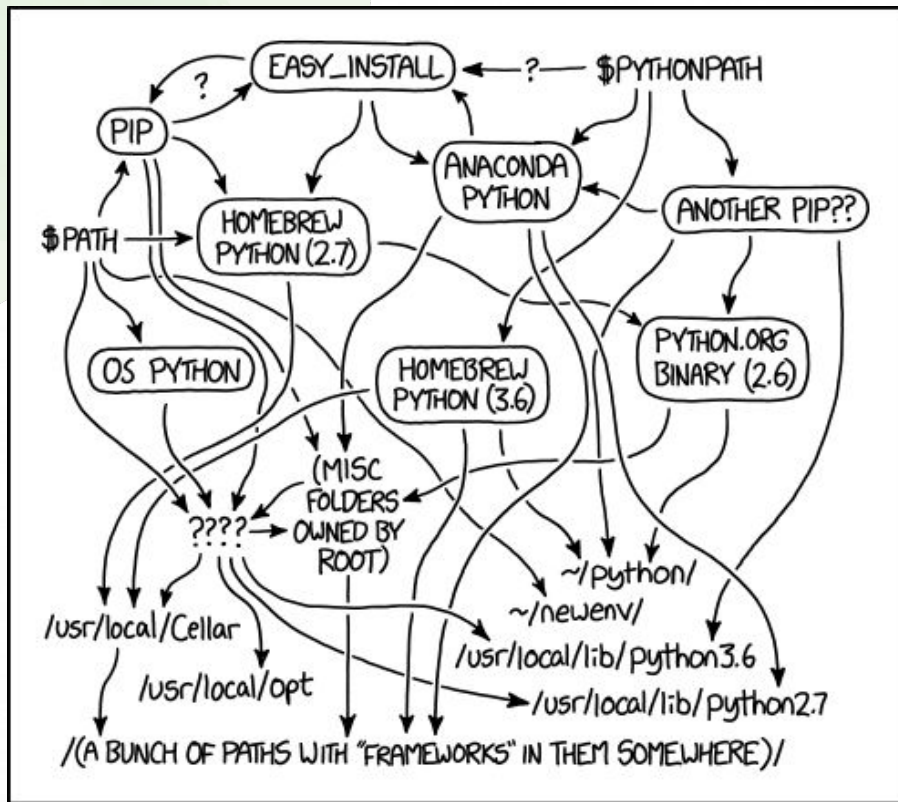
04

Installing Python





- [Official website](#)
- [GitHub](#)

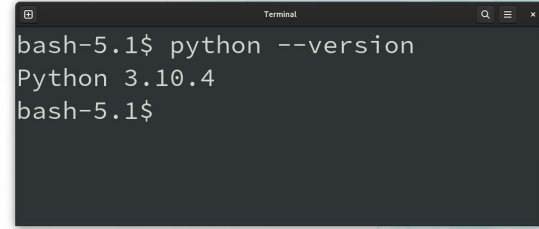


Python Environment

Source: [xkcd/1987](#)



Linux

A terminal window titled "Terminal" with a dark background. It shows the command `python --version` being executed, resulting in the output `Python 3.10.4`. The prompt `bash-5.1$` is visible at the top and bottom of the window.

```
bash-5.1$ python --version
Python 3.10.4
bash-5.1$
```

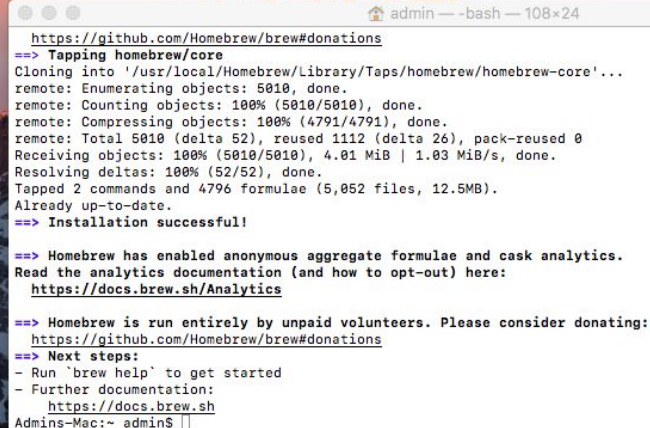
1. Usually, Python is already installed, so we don't have to do anything extra
 - a. To check, search for and open "Terminal" program
 - b. Type `python --version`
2. Dependencies required for the datascience package
 - a. Debian/Ubuntu/Mint/PopOS: `sudo apt install libopenblas-dev liblapack-dev gfortran python3-dev`
 - b. Fedora: `sudo dnf install openblas-devel lapack-devel gfortran python3-devel`
 - c. Arch Linux/Manjaro/Endeavour: `sudo pacman -S openblas-lapack gfortran python3`



macOS

Installation: macOS

1. Install Brew (package manager for macOS)
 - a. [Instructions](#)
2. Install Python with Brew
 - a. Search for and open "Terminal" program
 - b. `brew install python3`
3. Dependencies required for datascience package
 - a. macOS: `brew install lapack openblas`



A screenshot of a macOS Terminal window with a dark background and a window title bar showing 'admin' and '-bash' with a zoom level of '108x24'. The terminal output shows the process of installing Homebrew. It starts with the URL 'https://github.com/Homebrew/brew#donations', followed by 'Tapping homebrew/core'. The output details the cloning process into '/usr/local/Homebrew/Library/Taps/homebrew/homebrew-core', showing progress for enumerating, counting, and compressing objects. It then shows the total size (5010 delta, 1112 reused), the time taken (4.01 MiB at 1.03 MiB/s), and the resolution of deltas. The final output states 'Installation successful!' and provides links for analytics and donations. It also lists next steps: running 'brew help' and visiting the documentation at 'https://docs.brew.sh'.

```
admin — -bash — 108x24
https://github.com/Homebrew/brew#donations
==> Tapping homebrew/core
Cloning into '/usr/local/Homebrew/Library/Taps/homebrew/homebrew-core'...
remote: Enumerating objects: 5010, done.
remote: Counting objects: 100% (5010/5010), done.
remote: Compressing objects: 100% (4791/4791), done.
remote: Total 5010 (delta 52), reused 1112 (delta 26), pack-reused 0
Receiving objects: 100% (5010/5010), 4.01 MiB | 1.03 MiB/s, done.
Resolving deltas: 100% (52/52), done.
Tapped 2 commands and 4796 formulae (5,052 files, 12.5MB).
Already up-to-date.
==> Installation successful!

==> Homebrew has enabled anonymous aggregate formulae and cask analytics.
Read the analytics documentation (and how to opt-out) here:
  https://docs.brew.sh/Analytics

==> Homebrew is run entirely by unpaid volunteers. Please consider donating:
  https://github.com/Homebrew/brew#donations
==> Next steps:
- Run `brew help` to get started
- Further documentation:
  https://docs.brew.sh
Admins-Mac:~ admin$
```




Windows

Installation: Windows (Python)

Three primary ways to install on Windows

1. Install [from the Microsoft Store](#)
 - a. Note: Neither Poetry nor datascience will work properly
2. Installing using [python.org](#)'s Official Installer
 - a. Note: datascience will not work properly
 - b. For installation steps see [video](#) (up until 3:07)
3. Installing using [MSYS2](#)
 - a. Go to [website](#) and download and run the latest installer
 - b. Search and open "MSYS2 MSYS" program
 - c. In open terminal, run `pacman -S python`
 - d. Dependencies required for datascience package:
 - i. `pacman -S zlib zlib-devel mingw-w64-clang-x86_64 {lcms2,libwebp,libjpeg-turbo,zlib,libimagequant,libraqm}`

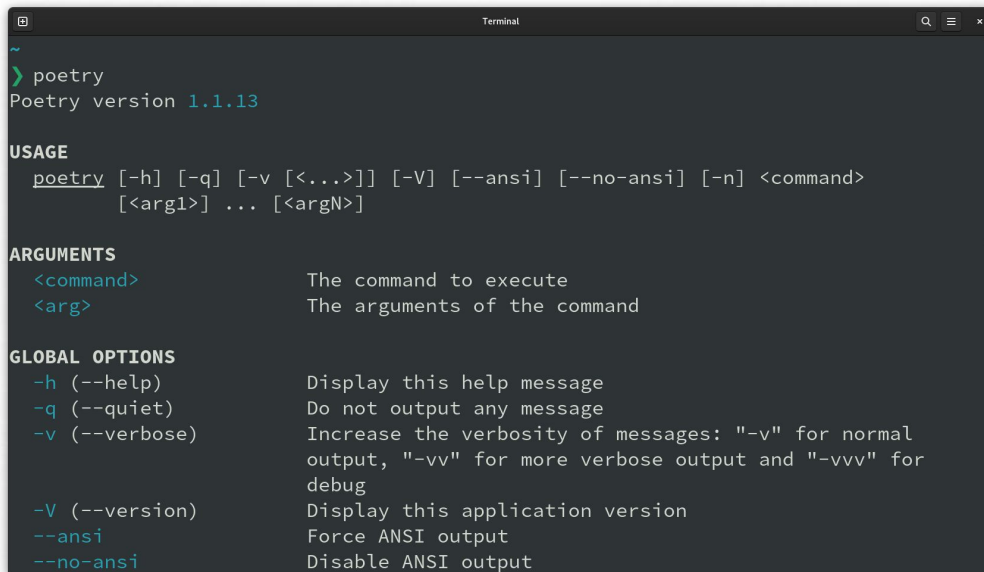
The background features a light cream color with abstract watercolor splashes in shades of light blue and light green. Scattered throughout are small, dark blue dots of varying sizes. A thin, dark teal line curves across the right side of the image, ending in a series of three loops.

05

Installing Poetry

Installing (Poetry)

1. Must follow instructions from [this permalink](#)
2. To test if Poetry installed successfully:
 - a. Open "Terminal"
 - b. Type "poetry"
3. If Poetry "command cannot be found", follow [these instructions](#)



```
~  
> poetry  
Poetry version 1.1.13  
  
USAGE  
  poetry [-h] [-q] [-v [<...>]] [-V] [--ansi] [--no-ansi] [-n] <command>  
        [<arg1>] ... [<argN>]  
  
ARGUMENTS  
  <command>    The command to execute  
  <arg>        The arguments of the command  
  
GLOBAL OPTIONS  
  -h (--help)    Display this help message  
  -q (--quiet)   Do not output any message  
  -v (--verbose) Increase the verbosity of messages: "-v" for normal  
                  output, "-vv" for more verbose output and "-vvv" for  
                  debug  
  -V (--version) Display this application version  
  --ansi         Force ANSI output  
  --no-ansi      Disable ANSI output
```

The background features a light cream color with abstract watercolor washes in shades of light blue and light green. Scattered throughout are numerous small, dark blue dots of varying sizes. A thin, dark teal line curves across the right side of the image, ending in a series of three loops.

06

Using datascience

datascience

A Python package for numerical computing. It includes other packages like [numpy](#).

1. `poetry init`
2. `poetry add requests datascience`
 - a. `datascience` [Example](#) on replit.com

- Quick Links

- [requests Website](#)
- [datascience Documentation](#)



Thanksies ^w^

Attributions

- Template by Slidesgo
- Icons by Flaticon