



Père Castor 🐻, raconte-nous une histoire (de Python) 🐍

David Aparicio

Meetup Python
Jeudi 13 Février 2024, 19h00



Devoxx France 2022
10^{ème} édition - 20 au 22 avril 2022



Firefox File Edit View History Bookmarks Tools Window Help

DevoxxFR22] - Père Castor - C x + https://docs.google.com/presentation/d/1-6sN2r5dnvwXuGsoCc_CMlGXcrKD12Rj68NorBvN4ZA/edit#slide=id.g124cb3a35ce_0_216

Père Castor 🐻, raconte nous une histoire (d'OPS)

David Aparicio @dadideo



@dadideo

David Aparicio

15/ DD INSA de Lyon / UNICAMP (Brésil)

Facebook Open Academy / MIT AppInventor

17/ Dev(Sec)Ops @ AMADEUS (Nice, 2 ans)

19/ Data(Sec)Ops @ OVHcloud (Lyon, 5 ans)

24/ Senior Dev(Sec)Ops @ Sopht (Lyon, 3 mois)





Une solution pensée pour un pilotage centralisé de votre écosystème IT

Collecte automatisée

Sur toute la chaîne de valeur d'un écosystème IT

Observabilité à 360°

sur les scopes 1, 2 et 3 avec des données granulaires et dynamiques

Actionnabilité guidée

Au moyen de simulations et recommandations contextualisées

Suivi d'impact

Avec l'incidence de vos actions sur votre trajectoire en surveillant l'impact CO2/€

The screenshot displays two main dashboards from the sopht platform:

- Devices Monitoring:** Shows YTD Emissions (102 kgCO2eq), Range Emissions (53 kgCO2eq), and YTD Energy Cons (41 kgCO2eq) with year-over-year comparisons. It also shows YTD Emissions (232 kgCO2eq) for Manufacturing Emissions.
- Devices Usage:** A scatter plot showing device usage data across various categories like RU, BR, and IT. A modal window prompts the user to "Select a device in the list below".

The sidebar navigation includes sections like Overview, Trajectories, Plans, Modules (Cloud & Infra, Workplace), and Asset Optimization.



Observabilité

- Mesure complète de l'écosystème IT
- Couverture des scopes 1, 2 et 3
- Facteurs d'émission multicritères
- Exploration dynamique des données
- Projection et prévision d'impact
- Filtrage des données par entité





Nos principales références



BNP PARIBAS
La banque d'un monde qui change



AIRBUS



Adecco

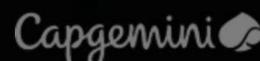
SEQENS

CED

soitec



Nos partenaires



accenture

Intégrations aux plateformes ESG



SWEET



“

Évaluer et comprendre nos principaux facteurs d'émission est évidemment un enjeu en soi, mais les piloter de manière dynamique avec des plans d'actions concrets, c'est ce que nous permet la plateforme Sopht.



Marie Ait-Daoud
Green IT manager
Vinci Groupe



Agenda

Introduction

Histoires

Conseils

Conclusion



Introduction





Comment ai-je découvert Python ? en 2011

The screenshot shows the PyPI project page for pwntools 4.14.0. The title is "pwntools 4.14.0". Below it is a button to "pip install pwntools". To the right, there's a "Latest version" button and a note that it was released on Jan 15, 2025. A description below the title reads: "Pwntools CTF framework and exploit development library." The left sidebar has a "Navigation" section with "Project description" (which is currently selected), "Release history", and "Download files". The main content area is titled "Project description" and contains the heading "pwntools - CTF toolkit". It features a large red "PWNTOOLS" logo with a flag icon. Below the logo, there are several status badges: "pypi v4.14.0", "docs passing", "build passing", "Coversalls", "license MIT", "repositories 11", "Discord 122 online", and a "Follow @Pwntools" link. A text block states: "Pwntools is a CTF framework and exploit development library. Written in Python, it is designed for rapid prototyping and development, and intended to make exploit writing as simple as possible." A code snippet is shown in a box:

```
from pwn import *
context(arch = 'i386', os = 'linux')

r = remote('exploitme.example.com', 31337)
# EXPLOIT CODE GOES HERE
r.send(asn(shellcraft.sh()))
r.interactive()
```

Navigation

Project description

Release history

Download files

Verified details

These details have been [verified by PyPI](#)

Project links

download

GitHub Statistics

Repository

Stars: 12345

Forks: 1736

Open issues: 111

Project description

pwntools - CTF toolkit



PWNTOOLS

pypi v4.14.0 docs passing build passing Coversalls license MIT repositories 11 Discord 122 online Follow @Pwntools

Pwntools is a CTF framework and exploit development library. Written in Python, it is designed for rapid prototyping and development, and intended to make exploit writing as simple as possible.

```
from pwn import *
context(arch = 'i386', os = 'linux')

r = remote('exploitme.example.com', 31337)
# EXPLOIT CODE GOES HERE
r.send(asn(shellcraft.sh()))
r.interactive()
```



[Pip / Pwntools](#)



Comment ai-je découvert Python ?

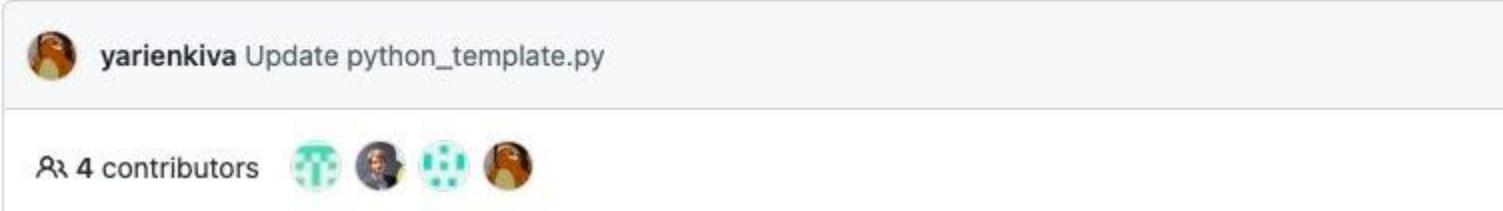
h25io / competitive-programming-templates Public

<> Code Issues Pull requests Actions Projects Wiki Security Insights

master ▾ competitive-programming-templates / python_template.py / <> Jump to ▾

yarienkiva Update python_template.py

4 contributors





Comment ai-je découvert Python ?

The screenshot shows a video call interface with a participant named "MathisHammel". The participant's profile picture is visible, and they are currently live, as indicated by the "LIVE" button in the top right corner of the video frame. The video frame displays a GitHub repository page for "h25io / competitive-programming-templates". The repository is public and contains 9 commits. The most recent commit is a merge pull request from "yarienkiva/master" to "master". The repository includes files like README.md, python_template.py, and python_template_tests.py. The README.md file describes the repository as a collaborative set of templates for coding competitions. The repository has no releases or packages published. Contributors listed include MathisHammel, cup2of2tea, magemax, and yarienkiva. A sidebar on the right shows a video feed of the participant, with a "GUARDIA CYBERSECURITY SCHOOL" logo overlaid. The bottom of the screen shows a navigation bar with icons for back, forward, and search.

MathisHammel

Analyse de données Twitter

Playing Software and Game Development for 38 viewers

Press esc to exit full screen

LIVE

Search or jump to... Pull requests Issues Marketplace Explore

h25io / competitive-programming-templates Public

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

master branch 0 tags

MathisHammel Merge pull request #3 from yarienkiva/master ... 5c347de on 23 Dec 2020 9 commits

README.md Added first version of Python template 2 years ago

python_template.py Update python_template.py 12 months ago

python_template_tests.py Ajout des tests pour primalité 2 years ago

Go to file Add file Code

About No description, website, or topics provided.

Readme

Releases No releases published Create a new release

Packages No packages published Publish your first package

Contributors

MathisHammel Mathis HAMMEL

cup2of2tea

magemax

yarienkiva Aloï

GUARDIA CYBERSECURITY SCHOOL

Advent of Code : 224 ★/300

https://github.com/h25io/competitive-programming-templates/blob/master/python_template.py

feh \$STREAM/layouts/3.png



Imports utiles pour les h4ck3rz

Pour se préparer pour l'iCTF, il faudra connaître quelques libs utiles:

- `ictf` : L'iCTF est centré sur cette bibliothèque pour soumettre les flags et lancer des exploits.
- `requests` : pour faire des requêtes HTTP simplement
- `socket` : Utilitaires pour sockets
- `pwn` : Pour faire des exploits distants
- `angr` : Pour faciliter le reverse de binaires en local

Il y a une lib pour tout en python, il suffit de bien chercher ! <https://xkcd.com/353/>



Tips supplémentaires

Les fonctions `chr()` et `ord()` servent à convertir des lettres en nombre et vice-versa (code ASCII)

On peut utiliser `0x` et `0b` avant les nombres pour éviter d'avoir à les convertir depuis le binaire ou l'hexa. Par exemple `0xa2` vaudra `162`.

Les structures de données de la STL C++ sont présentes en Python. Voir `set`, `queue`, `deque`... Certains imports peuvent être nécessaires comme `collections`

Les inputs se font avec `input()` (python 3) ou `raw_input()` (python 2)



Python vs Java



Performance

Readability

Collaboration

Hiring

Learning Difficulty

Dynamic vs Static

Performance

Readability

Collaboration

Hiring

Learning Difficulty

Dynamic vs Static



Mais

Python 2.7

str

bytearray

unicode

Python 3.x

bytes

bytearray

str

ASCII string, array of bytes

Mutable list of bytes

Unicode string (UTF-8)





Mais

Python2

```
<pre>  
print 'Hello, world'  
</pre>
```

```
<pre>  
int a = 5 / 2  
print a 2  
</pre>
```

Python3

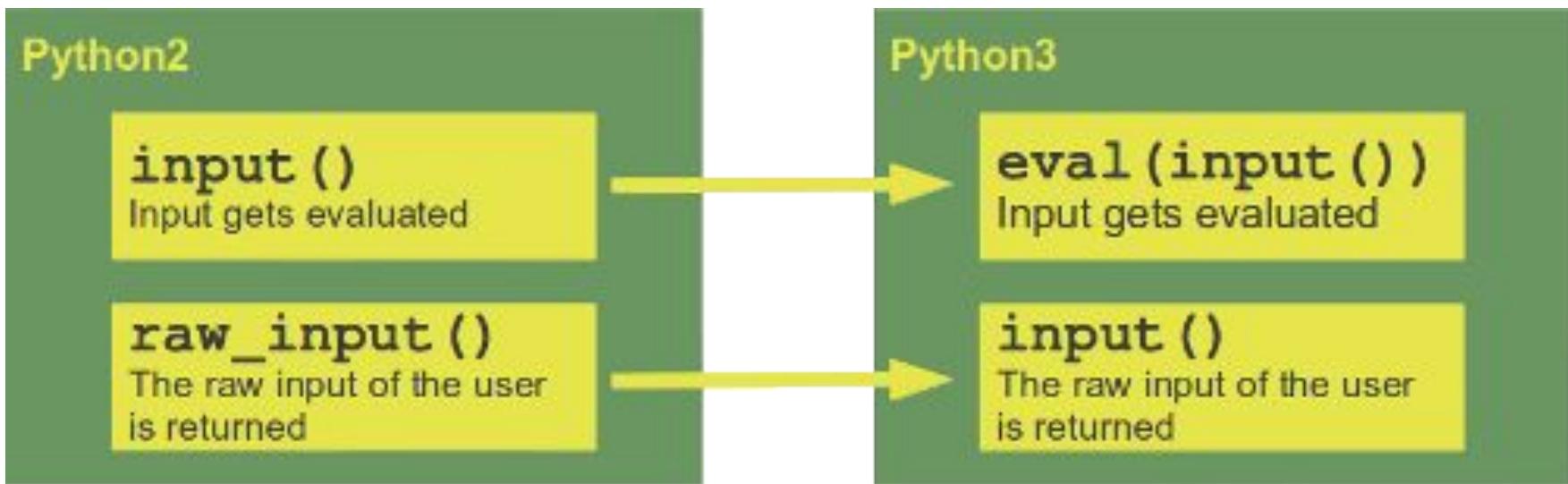
```
<pre>  
print(Hello, world)  
</pre>
```

```
<pre>  
int a = 5 / 2  
print a 2.5  
</pre>
```





Mais





Mais ?

`python2 -m SimpleHTTPServer 8000`

`python3 -m http.server 8000`



Mais

PYTHON 2.X



PYTHON 3.X

← LEGACY

FUTURE →

It is still entrenched in the software at certain companies



LIBRARY

Many older libraries built for Python 2 are not forwards compatible

0100 ASCII
0001

Strings are stored as ASCII by default



$$7/2=3$$

It rounds your calculation down to the nearest whole number



print "WELCOME TO GEEKSFORGEEKS"

It rounds your calculation down to the nearest whole number

It will take over Python 2 by the end of 2019



Many of today's developers are creating libraries strictly for use with Python 3

0000
0000
0100
0001

UNICODE

Text Strings are Unicode by default

7/2=3.5



This expression will result in the expected result

print("WELCOME TO GEEKSFORGEEKS")



This expression will result in the expected result





Mais

©2012-2015 - Laurent Postal - Monicas v2.0.4
License Creative Commons Attribution 4

Base Types	Container Types
<pre>integer, float, boolean, string, bytes int 783 0 -192 0b010 0o642 0x#3 float 9.23 0.0 -1.7e-6 bool True False str "One\nTwo" escaped new line escaped tab escaped \n escaped \t escaped \r bytes b'toto\xfe\x7f\x75' hexadecimal octal 2 immutables</pre>	<pre>* ordered sequences, fast index access, repeatable values list [1,5,9] ["x","11,8.9"] {"not":1} tuple (1,5,9) (11,"y",7.4) {"not":1} Non-mutable values (immutables) 2 expression with just commas --tuple str bytes (ordered sequences of chars / bytes)</pre>
	<pre>* key containers, no a priori order, fast key access, each key is unique dictionary dict {"key1": "value"} dict(a=3,b=4,k="v") [] (key-value associations) {1:"one",3:"three",2:"two",3.14:"pi"} set set {"key1", "key2"} (1,9,3.0) set() 2 key-hashed values (hash types, immutables...) frozenset(immutable set)</pre>
	<pre>Conversions type(expression) specify integer number base in 2nd parameter int("15") → 15 int("3E",16) → 63 int(15.56) → 15 truncate decimal part float("-11.24e8") → -1124000000.0 round(15.56,1) → 15.6 rounding to 1 decimal (0 decimal → integer number) bool(x) False for null x, empty container x, None or False x:True for other x str(x) → "..." representation string of x for display (cf. formating on the back) chr(64) → '@' ord('@')→64 code → char repr(x) → "... " literal representation string of x</pre>
	<pre>Convertions bytes([72,9,64]) → b'Hello' list('abc') → ['a','b','c'] dict([(3,'three'),(1,'one')]) → {1:'one',3:'three'} set(['one','two']) → {'one', 'two'} separator str and sequence of str → assembled str :'.join(['toto','12','pawd']) → "toto:12:pawd" str splitted on whitespaces → list of str "words with spaces".split() → ['words','with','spaces'] str splitted on separator str → list of str "1,4,8,2".split(",") → [1,4,8,2] sequence of one type → list of another type (via comprehension list) [int(x) for x in ('1','29','-3')] → [1,29,-3]</pre>
	<pre>Sequence Containers Indexing lists, tuples, strings, bytes... Items count Individual access to items via lst[index] len(lst) → 10 → first one lst[1]→20 lst[0] → 10 → last one lst[-1]→40 lst[-1] → 50 → last one lst[-2]→40 # index from 0 to 4 (here from 0 to 4) On mutable sequences (list), remove with del lst[3] and modify with assignment lst[4]=25</pre>
	<pre>Access to sub-sequences via slice [start slice : end slice : step] lst[:-1]→[10,20,30,40] lst[::-1]→[50,40,30,20,10] lst[1:-3]→[20,30] lst[:3]→[10,20,30] lst[1:-1]→[10,20,30,40] lst[:::-1]→[50,30,10] lst[::2]→[50,30,10] lst[-3:-1]→[30,40] lst[3:]→[40,50] lst[::2]→[10,20,30,50] lst[::]→[10,20,30,40,50] shallow copy of sequence</pre>
	<pre>Missing slice indication → from start / up to end. On mutable sequences (list) remove with del lst[3:5] and modify with assignment lst[1:4]=[15,25]</pre>
<pre>Boolean Logic Comparators: < <= > >= != ! (boolean results) ≤ ≥ = = and and and or or or a or b logical or one or other a and b logical and and and and i.e. all and or return value of a or b (under short circuit evaluation). to ensure that a and b are booleans not a logical not True False</pre>	<pre>Statements Blocks parent statement: statement block 1... parent statement: statement block2... next statement after block 1</pre>
<pre>Maths from math import sin,pi sin(pi/4) → 0.707 cos(2*pi/3) → -0.4999... sqrt(81) → 9.0 log(e**2) → 2.0 ceil(1.5) → 2 floor(1.5) → 1 modf(1.5) → (1.0, 0.5) decimal, fractions, numpy, re, etc doc</pre>	<pre>Modules/Names Imports from monmod import nom1,nom2 as fct import monmod → direct access to names, renaming with as modules and packages searched in path (cf sys.path) statement block executed only if a condition is true if logical condition: statements block Can go with several elif, else, and only one final else. Only the block of first time condition is executed. with a var x: if bool(x)==True: → if x: statements block else: statements block Signaling an error raise Exception(...) Errors processing: try: ... except: ... finally Block for final processing in all cases.</pre>
	<pre>Exceptions on Errors with a var x: if bool(x)==True: → if x: error processing else: error processing except Exception as e: error processing block</pre>





#MeetupPython
#PereCastor #OVHcloud

@dadideo



22



Python ? en 2022



Ham Gretsky
@ham_gretsky

Stitch

github.com/nathanlopez/St...

This is a cross platform python framework which allows you to build custom payloads for Windows, Mac OSX and Linux as well. You are able to select whether the payload binds to a specific IP and port, listens for a connection on a

nathanlopez/**Stitch**

Python Remote Administration Tool (RAT)



3 Contributors 37 Issues 2k Stars 552 Forks

github.com

GitHub - nathanlopez/Stitch: Python Remote Administration Tool (RAT)
Python Remote Administration Tool (RAT). Contribute to nathanlopez/Stitch development by creating an account on GitHub.

10:54 AM · Feb 25, 2022 · Media-Poster_V_2



Le Rouen Tech @lerouentech · 6h

Une lib/un tuto pour faire des API (trèèèès simple) en Python ?

Je suis rouillé alors vraiment un truc très simple svp ^^

2



Tip



Le Rouen Tech @lerouentech · 6h

Peut être flask ?

Le Deep Learning

La stack

DL4J

JS

Sonnet

python™

C++

TensorFlow

Java

Caffe2

ONNX

Keras

PYTORCH

GLUON

mxnet

Chainer

molt
academy



Python ? en 2022

```
"msg": "Reading package lists... \nBuilding dependency tree... \nReading state information... \nReading extended state information... \nInitializing package states... \nBuilding tag database... \nThe following packages will be upgraded:\n    apache2 apache2-bin apache2-data apache2-utils python-lxml\nThe following packages are RECOMMENDED but will NOT be installed:\n    python-bs4 python-html5lib\n5 packages upgraded, 0 newly installed, 0 to remove and 0 not upgraded.\nNeed to get 3108 kB of archives. After unpacking 4096 B will be used.\nWould download/install/remove packages.\n"}\n\nTASK [Update all packages to their latest version] ****\nThursday 27 January 2022  08:07:33 +0100 (0:00:00.076)      0:00:06.767 ****\nchanged: [vps-bea09116.vps.ovh.net]\n\nTASK [Upgrade the OS (apt-get dist-upgrade)] ****\nThursday 27 January 2022  08:07:40 +0100 (0:00:06.680)      0:00:13.448 ****\nok: [vps-bea09116.vps.ovh.net]\n\nTASK [Remove useless packages from the cache] ****\nThursday 27 January 2022  08:07:41 +0100 (0:00:01.356)      0:00:14.805 ****\nchanged: [vps-bea09116.vps.ovh.net]\n\nTASK [Remove dependencies that are no longer required] ****\nThursday 27 January 2022  08:07:42 +0100 (0:00:01.180)      0:00:15.986 ****\nok: [vps-bea09116.vps.ovh.net]\n\nTASK [Check if reboot is required] ****\nThursday 27 January 2022  08:07:44 +0100 (0:00:01.453)      0:00:17.440 ****\nok: [vps-bea09116.vps.ovh.net]\n\nTASK [display result] ****\nThursday 27 January 2022  08:07:45 +0100 (0:00:00.895)      0:00:18.335 ****\nok: [vps-bea09116.vps.ovh.net] => {\n    \"reboot_required_file.rc\": \"VARIABLE IS NOT DEFINED!\"\n}\n\nPLAY RECAP ****\nvps-bea09116.vps.ovh.net : ok=9    changed=4    unreachable=0    failed=0    skipped=0\n    rescued=0   ignored=0\n\nThursday 27 January 2022  08:07:45 +0100 (0:00:00.068)      0:00:18.403 ****\n=====\nUpdate all packages to their latest version ----- 6.68s\nOnly run \"update_cache=yes\" if the last one is more than 3600 seconds ago --- 4.81s\nCheck for package upgradeable ----- 1.84s\nRemove dependencies that are no longer required ----- 1.45s\nUpgrade the OS (apt-get dist-upgrade) ----- 1.36s\nRemove useless packages from the cache ----- 1.18s\nCheck if reboot is required ----- 0.90s\nShow me ----- 0.08s
```



Python ? en 2022

```
[i] Pi-hole blocking will be enabled
[i] Enabling blocking
[✓] Pi-hole Enabled
Pi-hole version is v5.2.4 (Latest: v5.2.4)
AdminLTE version is v5.3.2 (Latest: v5.3.2)
FTL version is v5.6 (Latest: v5.6)
[cont-init.d] 20-start.sh: exited 0.
^C
dparicio@raspberrypi:~ $ logout
debug1: client_input_channel_req: channel 0 rtype exit-status reply 0
debug1: client_input_channel_req: channel 0 rtype eow@openssh.com reply 0
debug1: channel 0: free: client-session, nchannels 1
Connection to raspberrypi closed.
Transferred: sent 5736, received 8756 bytes, in 8.0 seconds
Bytes per second: sent 715.5, received 1092.2
debug1: Exit status 130
▶(myenv)
vps-bea09116 on ▶ main [!?] via @myenv took 9s
→ cd ~/code/vps-bea09116/; ansible-playbook -i inventory.yml full_upgrade.yml -K --limit rpi
BECOME password:

PLAY [all] ****
TASK [Only run "update_cache=yes" if the last one is more than 3600 seconds ago] ***
Wednesday 01 December 2021  16:02:32 +0100 (0:00:00.031)      0:00:00.031 ****
fatal: [raspberrypi]: FAILED! => {"ansible_facts": {"discovered_interpreter_python": "/usr/bin/python"}, "changed": false, "msg": "Failed to update apt cache: W:This must be accepted explicitly before updates for this repository can be applied. See apt-secure(8) manpage for details., E:Repository 'http://archive.raspberrypi.org/debian buster InRelease' changed its 'Suite' value from 'testing' to 'oldstable', W:This must be accepted explicitly before updates for this repository can be applied. See apt-secure(8) manpage for details., E:Repository 'http://raspbian.raspberrypi.org/raspbian buster InRelease' changed its 'Suite' value from 'stable' to 'oldstable'"}

PLAY RECAP ****
raspberrypi      : ok=0      changed=0      unreachable=0      failed=1      skipped=0      rescued=0      ignored=0

Wednesday 01 December 2021  16:03:18 +0100 (0:00:45.499)      0:00:45.531 ****
=====
Only run "update_cache=yes" if the last one is more than 3600 seconds ago -- 45.50s
▶(myenv)
```



Python ? en 2022

I can see some mitogen entries are you using mitogen as a strategy_plugin in your defaults?
Suggest to try commenting that(strategy_plugin, strategy) in your ansible.cfg file.

0

By the looks of the logs you seem to be running Ansible version 2.9.9. Mitogen version 0.2.9 seems to work for Ansible 2.3-2.8 along with Python 2.6, 2.7, 3.6 and 3.7

Verify Mitogen version and try commenting it in ansible.cfg file

Reference link: https://mitogen.networkgenomics.com/ansible_detailed.html#noteworthy-differences



Python ? en 2022

```
(integration2) ~/c/s/D/silver-deployer >>> pip --version
pip 21.1.2 from /Users/daparici/lib/miniconda3/envs/integration2/lib/python3.6/site-packages/pip (python 3.6)
(integration2) ~/c/s/D/silver-deployer >>> pip install -f requirements.txt
ERROR: You must give at least one requirement to install (maybe you meant "pip install requirements.txt"?)  

(integration2) ~/c/s/D/silver-deployer >>> pip install requirements.txt
ERROR: Could not find a version that satisfies the requirement requirements.txt (from versions: none)
ERROR: No matching distribution found for requirements.txt
(integration2) ~/c/s/D/silver-deployer >>> █
```



Sunset

www.python.org/doc/sunset-python-2/

The screenshot shows the Python.org homepage with a banner at the top for "Sunset". Below the banner, there's a navigation bar with links for About, Downloads, Documentation, Community, Success Stories, News, and Events. On the left, there's a sidebar with a section titled "The PSF" which says: "The Python Software Foundation is the organization behind Python. Become a member of the PSF and help advance the software and our mission." The main content area has a large heading "Sunsetting Python 2" followed by several paragraphs of text explaining the decision to sunset Python 2.

The PSF

The Python Software Foundation is the organization behind Python. Become a member of the PSF and help advance the software and our mission.

Sunsetting Python 2

We are volunteers who make and take care of the Python programming language. We have decided that January 1, 2020, was the day that we sunset Python 2. That means that we will not improve it anymore after that day, even if someone finds a security problem in it. You should upgrade to Python 3 as soon as you can.

Why are you doing this?

We needed to sunset Python 2 so we can help Python users by improving Python faster.

We released Python 2.0 in 2000. We realized a few years later that we needed to make big changes to improve Python. So in 2006, we started Python 3.0. Many people did not upgrade, and we did not want to hurt them. So, for many years, we have kept improving and publishing both Python 2 and Python 3.

But this makes it hard to improve Python. There are improvements Python 2 can't handle. And we have less time to work on making Python 3 better and faster.

And if many people keep using Python 2, then that makes it hard for the volunteers who use Python to make software. They can't use the good new things in Python 3 to improve the tools they make.

We did not want to hurt the people using Python 2. So, in 2008, we announced that we would sunset Python 2 in 2015, and asked people to upgrade before then. Some did, but many did not. So, in 2014, we extended that sunset till 2020.

How long is it till the sunset date?



[sunset-python2](#)

The sunset date has now passed; it was January 1st, 2020.



Sunset

Python 2.x EOL Information

Name	AWS Lambda Runtime Identifier	PSF EOL	Last Update Available	AWS Lambda Python 2.7 EOL	Heroku Python 2.7 EOL	GCP App Engine Python 2.7 EOL
Python 2.7	python2.7	January 1, 2020	April 20, 2020	December 31, 2020	February 1, 2021	Unspecified Date





Java 8

Java SE 8 (1.8)	LTS	52	18th March 2014	April 2019 for Oracle November 2026 for Eclipse Temurin ^[15] November 2026 for Red Hat ^[10] November 2026 for Azul ^[9] December 2030 for Amazon Corretto ^[16] April 2025 for IBM Java SDK ^[13] November 2026 for IBM Semeru Runtimes ^[17]	December 2030 for Oracle ^[4] December 2030 for Azul ^[9] March 2031 for BellSoft Liberica ^[12]
--------------------	-----	----	-----------------	--	---

OpenJDK Lifecycle Dates and RHEL versions

	RHEL 5 Support Added	RHEL 6 Support Added	RHEL 7 Support Added	RHEL 8 Support Added	RHEL 9 Support Added	End of Full Support for OpenJDK version	End of ELS-1 Support Phase
OpenJDK 6 (1.6)	5.3	6.0	7.0	N/A	N/A	December 31, 2016	N/A
OpenJDK 7 (1.7)	5.9	6.3	7.0	N/A	N/A	June 30, 2020	N/A
OpenJDK 8 (1.8)	N/A	6.6*	7.1	8.0	see note below^	November 30, 2026*	N/A
OpenJDK 11	N/A	N/A	7.6	8.0	9.0	October 31, 2024	October 31, 2027
OpenJDK 17	N/A	N/A	N/A	8.4	9.0	October 31, 2027	N/A
OpenJDK 21	N/A	N/A	N/A	8.9	9.3	December 31, 2029	N/A





#MeetupPython
#PereCastor #OVHcloud

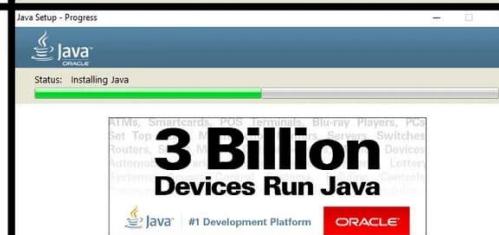
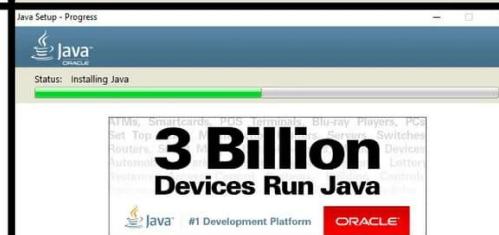
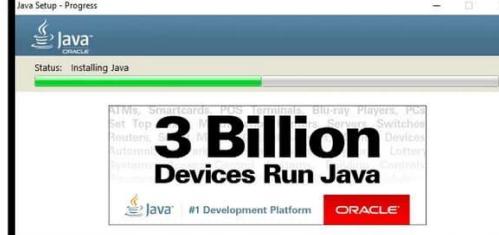
Java 8

1997

2002

2012

2020



@dadideo

The Arrival of Java 18 22nd March 2022

Used by over 10 million developers and running on **56 billion devices globally**, the Java Platform truly moves the world forward, and now even faster with Java 18.

[Download Java 18](#)

[Learn Java](#)

Currently Java 18 • [Previous releases](#)



[Reddit](#)



#MeetupPython
#PereCastor #OVHcloud

@dadideo



31



10:32



Sign in

Combien de milliar...



Combien de milliards d'appareils
ont Python dessus ? Vu que
Java c'est 56



Searching the web...

Python est un langage de programm

+ Ask me anything!





#MeetupPython
#PereCastor #OVHcloud

10:32 ☀️ 📺 🔍 3G

Sign in

Combien de milliar...

@dadideo



32

Combien de milliards d'appareils
ont Python dessus ? Vu que
Java c'est 56



Thought for 1.5s

Python est un langage de programmation très populaire, notamment dans le domaine de l'Internet des objets (IoT). Avec l'essor de l'IoT, de nombreux petits appareils de faible puissance connectés à Internet sont capables d'exécuter du code Python. Ces appareils incluent des dispositifs variés comme des smartphones, des consoles de jeu, des fours à micro-ondes, des smartwatches, et bien d'autres. Python s'est imposé comme un choix privilégié pour ces appareils en raison de sa simplicité et de sa productivité. De nombreux appareils supportent soit une version complète de Python, soit MicroPython, une version allégée de Python adaptée aux appareils plus simples ¹. Cependant, il n'existe pas de chiffre précis indiquant combien de milliards d'appareils exactement utilisent Python. On sait juste que Java est présent sur 56 milliards d'appareils ².

Sources

10:32 AM

+ Ask me anything!



10:33

Sign in How many devices ... < +

How many devices run Python ?

M

As of my last update in October 2023, Python is one of the most widely used programming languages in the world, running on a vast array of devices. Here are some categories of devices that commonly run Python:

- Personal Computers:** Millions of desktops and laptops run Python for various applications, including software development, data analysis, and automation.
- Servers:** Many servers use Python for web applications, data processing, and backend services.
- Mobile Devices:** Python can run on mobile devices through frameworks like Kivy and BeeWare, although it's not as common as native languages like Swift or Java.
- Embedded Systems:** Python is used in embedded systems and IoT devices through MicroPython and CircuitPython.
- Scientific and Research Equipment:** Many scientific instruments and research tools use Python for data analysis and control.
- Cloud Services:** Cloud platforms like AWS, Google Cloud, and Azure support Python for various services and applications.
- Educational Tools:** Python is widely used in educational settings, from schools to universities, for teaching programming and computational thinking.

While it's challenging to provide an exact number of devices running Python due to its widespread use and the diversity of applications, it's safe to say that the number is in the hundreds of millions, if not billions, when considering all types of devices and platforms.

10:32 AM

+ Ask me anything! →



While it's challenging to provide an exact number of devices running Python due to its widespread use and the diversity of applications, it's safe to say that the number is in the hundreds of millions, if not billions, when considering all types of devices and platforms.

10:32 AM



#MeetupPython
#PereCastor #OVHcloud



34



Léalinux @lea_linux · 1h

Portage de Python mais sans OS derrière :
youtu.be/bYQ_lq5dcvM

@dadideo

PyCon 2015

Montréal • April 8-16

Porting Python to run without an OS

Josh Triplett



1



1



8



Tip



Léalinux @lea_linux · 11m

Le module python bits: github.com/biosbits/bits
import bits
c = bits.cpuid(0, 0)
print(c.eax, c.ebx, c.ecx)

...

biosbits/bits

BIOS Implementation Test Suite





Home



NFT LATAM #uruguay #...

Best



Lionel Laské @lionellaske · 43m

L'index de popularité des langages recherchés dans Google: 1) Python, 2) Java, 3) JavaScript. L'index TIOBE (qui utilise un autre calcul) donne: 1) Python, 2) C, 3) Java [#classement](#) [#langage](#) [#python](#) [#java](#) [#javascript](#) [#c](#) bit.ly/3I1SWNL



2

3





#MeetupPython
#PereCastor #OVHcloud

13:17



@dadideo



36



← Tweet

dlnraja Retweeted



Lionel Laské
@lionelllaske

L'index de popularité des langages recherchés dans Google: 1) Python, 2) Java, 3) JavaScript. L'index TIOBE (qui utilise un autre calcul) donne: 1) Python, 2) C, 3) Java
#classement #langage #python #java #javascript #c bit.ly/3I1SWNL

Translate Tweet

12:33 · 30 Nov 21 · TWIST@C2S

1 Retweet 2 Likes



Tweet your reply





Michael Cade @MichaelCade1 · 1h

...

This is some great research. I wonder the path these people take as well. I have been wondering how we get to cloud positions. I am sure it's "everyone" but is it a majority sys admin, virtualisation, storage or dev moving into this role



Andrew Brown @andrewbrown · 18h

.@loujaybee analysed 100+ Cloud Jobs Descriptions and this is what was found:

70% AWS
47% Terraform
43% Azure
40% Kubernetes
36% Python
29% Linux

youtube.com/watch?v=IjYo-L...



1



1



...



Tip



Michael Cade @MichaelCade1 · 55m

My belief is that the virtualisation admin will evolve into cloud and DevOps roles. Everyone will need to know whichever platform, OS, IaC and wider automation techniques



#MeetupPython
#PereCastor #OVHcloud

22:15



@dadideo



38

← Thread

♥ GRZ and 13 others liked



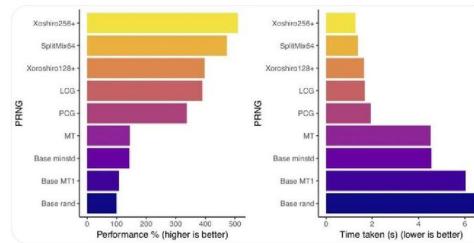
Mathis Hammel

@MathisHammel

THREAD : J'ai trouvé un bug qui affecte toutes les versions récentes de Python, et il est interdit de le réparer !

Je vous explique pourquoi 🔍

[Translate Tweet](#)



14:29 · 16 Nov 21 · Twitter Web App

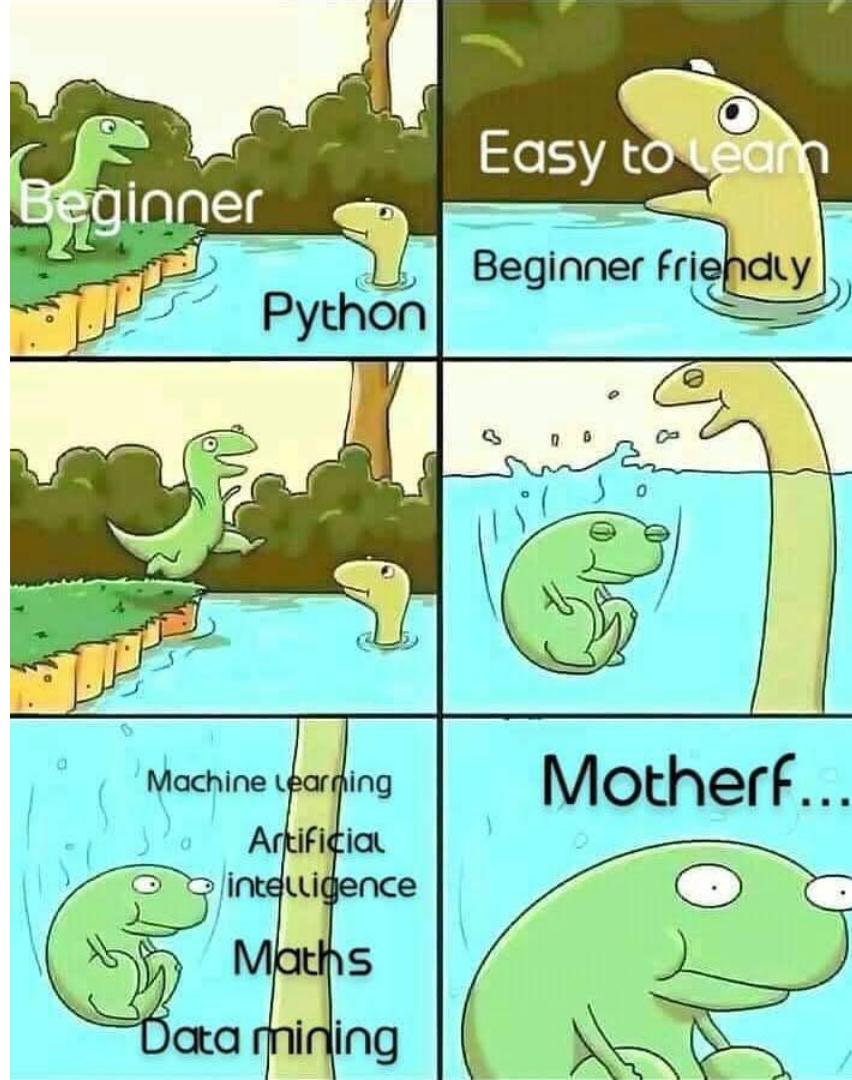
87 Retweets 8 Quote Tweets 373 Likes

Tweet your reply



[Twitter / Mathis Hammel](#)







#MeetupPython
#PereCastor #OVHcloud

@dadideo



41



MACHINE LEARNING AND GO

Francesc Campoy Flores
dotGo 2017

More talks on
dotconferences.com/talks



dotGo 2017 - Francesc Campoy Flores - Machine Learning and Go

14K views • 7 years ago

dot dotconferences

All the cool kids are doing Machine Learning, but they are all doing it in Python! Should we change that? Can we solve ML ...



16 chapters [Intro](#) | [History of Machine Learning](#) | [AlphaGo](#) | [AlphaGo Master](#) | [GPU CPU](#) | [Machine Artificial...](#)



2017 - E_TOO_EXCITE

<i>name</i>	<i>elo rating</i>	<i>matches</i>	<i>training hardware</i>
Fan Hui	3,144	5:0 against Fan Hui	176 GPU
Lee Sedol	3,739	4:1 against Lee Sedol	48 TPU
Master	4,858	60:0 against professional players	4 TPU
Zero	5,185	100:0 against AlphaGo Lee	4 TPU
		89:11 against AlphaGo Master	



High performance inference. Any model. Any hardware. No compromise.

ZML simplifies model serving,
ensuring peak performance and maintainability in production.



Star on GitHub

★ 2.0K



Read the Docs



AI Framework, batteries included

ZML primitives are reusable, self-contained, and modular. When combined, they compound to help you run faster.



Python-free

Production demands precision, not hacks. ZML is written without any Python code in the stack, it's clean, robust, and built for production.



Very, very high-performance

Every line of code delivers—no overhead, just raw speed.



Highly expressive

Code that's as easy to read as it is to trust.



Truly hardware agnostic with no compromise

Switch hardware with a single command, with no compromises.



Build once, run everywhere ... delivered

Delightful experience, it just works!



Open-source

We are building the future of AI inference with you.



#MeetupPython
#PereCastor #OVHcloud

@dadideo



45



A video thumbnail for a talk at dotAI 2024. On the left, there's a yellow circular logo with the word "dot" and a white "AI" next to it. To its right, the title "STEEVE MORIN ZML: ML FRAMEWORK FOR ZIG" is displayed in white capital letters. The main image shows a man with a beard, wearing a black t-shirt with "AUT ZML" and "AUT NIHL" printed on it, standing on stage and gesturing with his hands. The background is dark with colorful, abstract circular icons related to technology and machine learning.

dotAI 2024 - Steeve Morin - ZML: ML framework for Zig

634 views • 3 months ago

dot dotconferences

In this talk we will introduce ZML, a compiling machine learning framework for Zig that's based on MLIR, XLA and Bazel. ZML is ...

13:01



Python Golfer

[Github](#)[Q&A](#)

Enter your ungolfed Python code

```
import asyncio
import sys
async def run_command(*args):
    process = await asyncio.create_subprocess_exec(
        *args,
        stdout=asyncio.subprocess.PIPE,
        stderr=asyncio.subprocess.PIPE)
    stdout, stderr = await process.communicate()
    return stdout.decode().strip(), stderr.decode().strip(),
process.returncode
async def main():
    loop = asyncio.get_event_loop()
try:
    loop.run_until_complete(main())
finally:
    loop.close()
```

448 chars

 Use dflook's[python minifier](#)before golfing ? (might not reduce size)[Golf it!](#)

Golfed code

```
exec(bytes('浩潰瑣搘搔滅杖灭杖*梶@獨渥敷奮影潤渥涓..慣呆..」†袖倒激?睡榆*獨
渥楣*初憶整猶於袖倒搘†礎捲さ††††慣呆w†††獨搗峻減獨渥楣*疊灑消散獨僕偉m†††獨
搗彷灑獨渥楣*疊灑消散獨僕偉..†獨搗峻v獨搗彷?睡榆*袖倒激*潤渥湊搗瑚..」††敵曉潤獨
搗峻*數潤敷*獻抵轟*達數性標搘摯..」達械..@€灑消散獨胞渥牽据摯*獨渥敷愁漚..」灑灯涵溫
板据灑尤疣敷敵環江渦..」抵搔..†氣渦*窟彭湊彬潤敬整洨榆..」吳渙污徑..」灑灯拂渙數
```



```
 1 from fastapi import FastAPI, status
 2 from pydantic import BaseModel
 3
 4 from fastapi.staticfiles import StaticFiles
 5 from prometheus_fastapi_instrumentator import Instrumentator
 6
 7
 8 app = FastAPI()
 9
10
11 class HealthCheck(BaseModel):
12     """Response model to validate and return when performing a health check."""
13
14     status: str = "OK"
15
16
17 @app.get(
18     "/health",
19     tags=["healthcheck"],
20     summary="Perform a Health Check",
21     response_description="Return HTTP Status Code 200 (OK)",
22     status_code=status.HTTP_200_OK,
23     response_model=HealthCheck,
24 )
25 def get_health() -> HealthCheck:
26     """
27     ## Perform a Health Check
28     Endpoint to perform a healthcheck on. This endpoint can primarily be used Docker
29     to ensure a robust container orchestration and management is in place. Other
30     services which rely on proper functioning of the API service will not deploy if this
31     endpoint returns any other HTTP status code except 200 (OK).
32     Returns:
33         HealthCheck: Returns a JSON response with the health status
34     """
35     return HealthCheck(status="OK")
36
37
38 Instrumentator().instrument(app).expose(app)
39 app.mount("/", StaticFiles(directory="static", html=True), name="static")
```



Pause pour les questions





Axes d'amélioration





Python / Go

- Formatage
- Black formatter / Prettier / PyLint / Ruff
- Tests
- Pytest vs unittest (py2.1) / Pytests-mock
- Coverage (test -coverage)
- Benchmarks (test -bench=.)
- Data Race (test -race)
- Profiling (test -cpuprofile=a.prof)
- Pprof vs FlameGraph
- Trace (test -trace=a.bin)
- Fuzzing (test -fuzz=FuzzFuncA)
- Go Run
- Go Build



Python3 / Une STD plus complète

- Requests
- Gestion de Windows ?
- Gestion des paquets moins confuse
- Typage
- mypy vs pyright



Bensuperpc liked



#BlackLivesMatter @CodeDoesMeme · 33m

...

Windows troubleshooter source code leaked. (source: redd.it/mfjv60)

The screenshot shows a Sublime Text window with a dark theme. The title bar reads "C:\Users\Knz\AppData\Local\Programs\Python\Python37\new.py - Sublime Text". The menu bar includes File, Edit, Selection, Find, View, Goto, Tools, Project, Preferences, and Help. Below the menu is a tab bar with "Python notes.py", "knz.js", "new.py", and "cse.html". The main editor area contains the following Python code:

```
1 import os
2 def Windows_Diagnostics():
3     print('Looking for problems')
4     waitTimer(20000ms)
5     print('Unable to fix the issue.')
6
```



Forcer le passage à Python3 sans casser l'OS ?

```
grep -rl python . | xargs sed -i '/s/python/python3/g'
```



Forcer le passage à Python3 sans casser l'OS ?

```
[root@ovh-ipa ~]# Updating : rpcbind-0.2.0-49.el7.x86_64 239/781
Error: Cannot open file /var/cache/yum/x86_64/7/base/packages/polkit-0.112-26.el7.x86_64.rpm: [Errno 2] No such file or directory: '/var/cache/yum/x86_64/7/base/packages/polkit-0.112-26.el7.x86_64.rpm'.
TypeError: an integer is required
FATAL ERROR: python callback <bound method RPMTransaction.callback of <yum.rpmtrans.RPMTransaction instance at 0x7facb272ee60>> failed, aborting!
[daparici@ovh-ipa ~]$ sudo yum update
sudo: unknown uid 322800048: who are you?
[daparici@ovh-ipa ~]$
```



Forcer le passage à Python3 sans casser l'OS ?

```
→ python --version
Python 2.7.16
▶
~
→ source ~/.enable_conda.sh && conda activate myenv
(myenv)
~ via Ⓜmyenv
→ python --version
Python 3.9.1
```



Pause pour les questions





Histoires





WTF !?





Où est l'erreur ?

```
(qrcode) ~/Desktop >>> python qrcode.py
Traceback (most recent call last):
  File "qrcode.py", line 1, in <module>
    import qrcode
  File "/Users/daparicio/Desktop/qrcode.py", line 18, in <module>
    qr = qrcode.QRCode(border=0)
AttributeError: module 'qrcode' has no attribute 'QRCode'
(qrcode) ~/Desktop >>> python qrcode.py
Traceback (most recent call last):
  File "qrcode.py", line 1, in <module>
    import qrcode
  File "/Users/daparicio/Desktop/qrcode.py", line 18, in <module>
    qr = qrcode.QRCode(border=0)
AttributeError: module 'qrcode' has no attribute 'QRCode'
```



AsyncIO





4. Running the Async Program: Use `asyncio.run()` to run the top-level entry point function.

Here's a simple example:

python

Copy

```
import asyncio

# Define an asynchronous function
async def say_hello(delay):
    await asyncio.sleep(delay)
    print(f"Hello after {delay} seconds!")

# Define the main entry point
async def main():
    # Schedule say_hello functions to run asynchronously
    await asyncio.gather(
        say_hello(1),
        say_hello(2),
        say_hello(3)
    )

# Run the main function
asyncio.run(main())
```



For Python versions prior to 3.7, like Python 3.6, the approach to implementing asynchronous code using `asyncio` is slightly different, primarily in how you start and run the event loop. In Python 3.6, `asyncio.run()` is not available, so you need to use the `get_event_loop` method to get the event loop and then run your async functions. Here's the modified example for Python 3.6:

```
python                                ⚡ Copy

import asyncio

# Define an asynchronous function
async def say_hello(delay):
    await asyncio.sleep(delay)
    print(f"Hello after {delay} seconds!")

# Define the main entry point
async def main():
    # Schedule say_hello functions to run asynchronously
    await asyncio.gather(
        say_hello(1),
        say_hello(2),
        say_hello(3)
    )

    # Get the event loop
    loop = asyncio.get_event_loop()

    # Run the main function using the event loop
    try:
        loop.run_until_complete(main())
    finally:
        loop.close()
```



For Python versions prior to 3.7, like Python 3.6, the approach to implementing asynchronous code using `asyncio` is slightly different, primarily in how you start and run the event loop. In Python 3.6, `asyncio.run()` is not available, so you need to use the `get_event_loop` method to get the event loop and then run your `async` functions. Here's the modified example for Python 3.6:

```
python

import asyncio

# Define an asynchronous function
async def say_hello(delay):
    await asyncio.sleep(delay)
    print(f"Hello after {delay} seconds!")

# Define the main entry point
async def main():
    # Schedule say_hello functions to run asynchronously
    await asyncio.gather(
        say_hello(1),
        say_hello(2),
        say_hello(3)
    )

    # Get the event loop
loop = asyncio.get_event_loop()

# Run the main function using the event loop
try:
    loop.run_until_complete(main())
finally:
    loop.close()
```



```
async def run_command(*args):
    # Create subprocess
    process = await asyncio.create_subprocess_exec(
        *args,
        stdout=asyncio.subprocess.PIPE,
        stderr=asyncio.subprocess.PIPE)

    # Wait for the process to finish and capture output
    stdout, stderr = await process.communicate()

    # Return the output, error, and exit status
    return stdout.decode().strip(), stderr.decode().strip(), process.returncode

async def main():
    # Run two commands concurrently and get their results
    results = await asyncio.gather(
        run_command(sys.executable, '-c', 'print("Hello from Python 1")'),
        run_command(sys.executable, '-c', 'print("Hello from Python 2")'),
        return_exceptions=False
    )

    # Check if both subprocesses were successful
    if all(result[-1] == 0 for result in results):
        print("Both subprocesses completed successfully. Running the third subprocess")
        stdout, stderr, returncode = await run_command(sys.executable, '-c', 'print("')

        if returncode == 0:
            print(f"Third subprocess output: {stdout}")
        else:
            print(f"Error in third subprocess: {stderr}")
    else:
        print("Error in one or both of the initial subprocesses. Terminating script.")
        for stdout, stderr, returncode in results:
            if returncode != 0:
                print(f"Subprocess failed with exit code {returncode}: {stderr}")
```

Copy

@dadideo



64





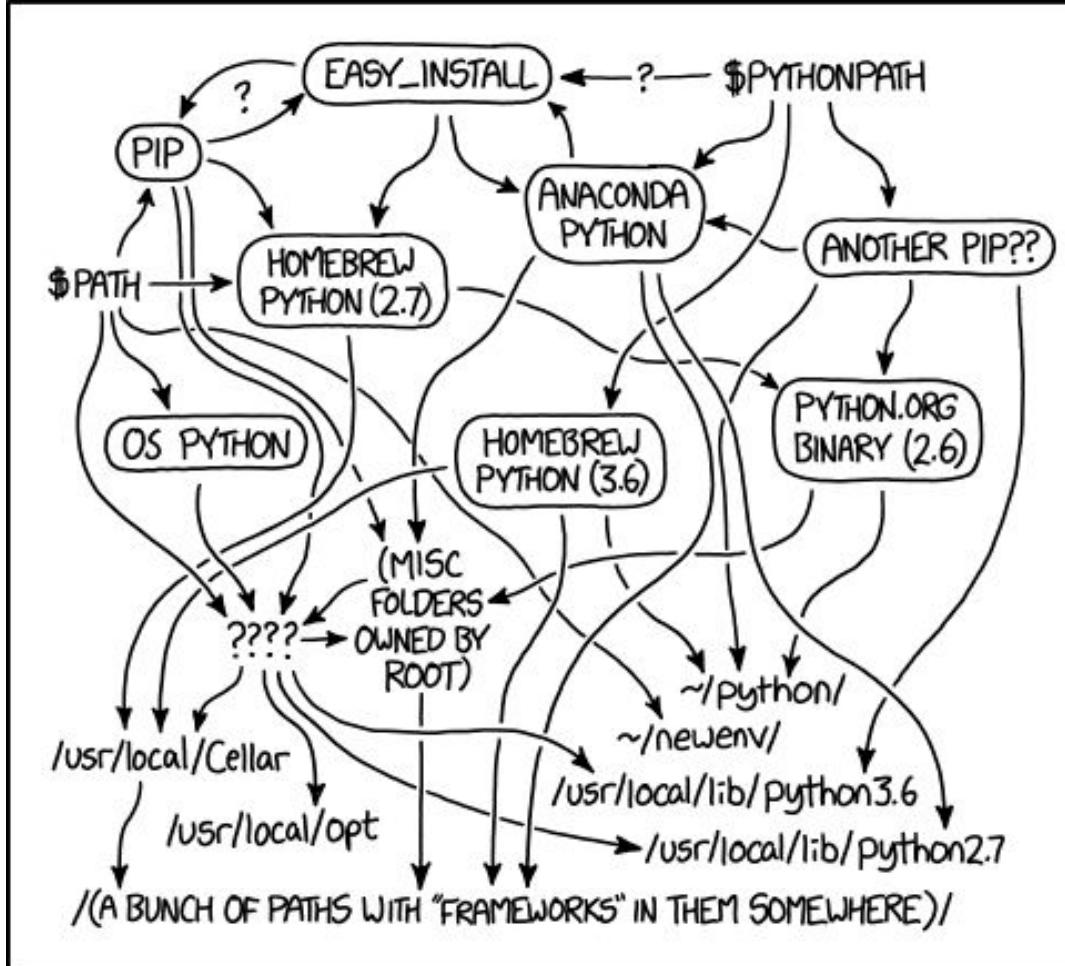
Pause pour les questions





Packaging





MY PYTHON ENVIRONMENT HAS BECOME SO DEGRADED
THAT MY LAPTOP HAS BEEN DECLARED A SUPERFUND SITE.





```
(fullhunt) ~/c/g/f/log4j-scan >>> conda install --file requirements.txt
Collecting package metadata (current_repodata.json): done
Solving environment: failed with initial frozen solve. Retrying with flexible solve.
Solving environment: failed with repodata from current_repodata.json, will retry with next repodata source.
Collecting package metadata (repodata.json): done
Solving environment: failed with initial frozen solve. Retrying with flexible solve.
Solving environment: \
Found conflicts! Looking for incompatible packages.
This can take several minutes. Press CTRL-C to abort.
failed

UnsatisfiableError: The following specifications were found
to be incompatible with the existing python installation in your environment:

Specifications:

- pycryptodome → python[version='≥2.7,<2.8.0a0|≥3.6,<3.7.0a0|≥3.7,<3.8.0a0|≥3.8,<3.9.0a0|≥3.9,<3.10.0a0']
- termcolor → python[version='≥2.7,<2.8.0a0|≥3.6,<3.7.0a0|≥3.7,<3.8.0a0|≥3.8,<3.9.0a0|≥3.9,<3.10.0a0|≥3.5,<3.6.0a0']

Your python: python=3.10

If python is on the left-most side of the chain, that's the version you've asked for.
When python appears to the right, that indicates that the thing on the left is somehow
not available for the python version you are constrained to. Note that conda will not
change your python version to a different minor version unless you explicitly specify
that.

The following specifications were found to be incompatible with each other:

Output in format: Requested package → Available versions

Package bzip2 conflicts for:
python=3.10 → bzip2[version='≥1.0.8,<2.0a0']
requests → python[version='≥3.6'] → bzip2[version='≥1.0.8,<2.0a0']

(fullhunt) ~/c/g/f/log4j-scan >>>
```



webex_sendmessage.py — python_webex

Select Interpreter



Selected Interpreter: ~/lib/homebrew/bin/python3

+ Enter interpreter path...

★ Python 3.9.13 64-bit ~/lib/homebrew/bin/python3

Recommended

Python 3.10.4 ('ansible') ~/lib/miniconda3/envs/ansible/bin/python

Conda

Python 3.10.4 ('qrcode') ~/lib/miniconda3/envs/qrcode/bin/python

Python 3.10.4 ('webex') ~/lib/miniconda3/envs/webex/bin/python

Python 3.9.7 ('fullhunt') ~/lib/miniconda3/envs/fullhunt/bin/python

Python 3.7.7 ('base') ~/lib/miniconda3/bin/python

Python 3.6.13 ('integration2') ~/lib/miniconda3/envs/integration2/bin/python

Python 3.6.10 ('silver') ~/lib/miniconda3/envs/silver/bin/python

Python 3.9.13 64-bit ~/lib/homebrew/bin/python3

Global

Python 3.8.2 64-bit /usr/bin/python3

Python 2.7.16 64-bit /usr/bin/python



```
> cat pyproject.toml
[project]
name = "diagrams-test"
version = "0.1.0"
description =
authors = [
    {name = "David Aparicio",email = "david.aparicio@ovhcloud.com"}
]
readme = "README.md"
requires-python = ">=3.12,<4.0"
dependencies = [
    "diagrams (>=0.24.1,<0.25.0)"
]

[build-system]
requires = ["poetry-core>=2.0.0,<3.0.0"]
build-backend = "poetry.core.masonry.api"
[tool.poetry]
package-mode = false%
```



```
> diagrams version

code/local/diagrams_test is 🎨 v0.1.0 via 🐍 v3.9.6
> poetry env use python3.12 && \
poetry install && \
poetry run python diagram.py
Creating virtualenv diagrams-test-0EbYmfIG-py3.12 in /Users/daparici/Library/Caches/pypoetry/virtualenvs
Using virtualenv: /Users/daparici/Library/Caches/pypoetry/virtualenvs/diagrams-test-0EbYmfIG-py3.12
Updating dependencies
Resolving dependencies... (0.4s)

The current project's supported Python range (>=3.12) is not compatible with some of the required packages Python requirement:
- diagrams requires Python <4.0,>=3.8, so it will not be satisfied for Python >=4.0

Because no versions of diagrams match >0.24.1,<0.25.0
and diagrams (0.24.1) requires Python <4.0,>=3.8, diagrams is forbidden.
So, because diagrams-test depends on diagrams (>=0.24.1,<0.25.0), version solving failed.

* Check your dependencies Python requirement: The Python requirement can be specified via the `python` or `markers` properties

For diagrams, a possible solution would be to set the `python` property to ">=3.12,<4.0"

https://python-poetry.org/docs/dependency-specification/#python-restricted-dependencies,
https://python-poetry.org/docs/dependency-specification/#using-environment-markers
```

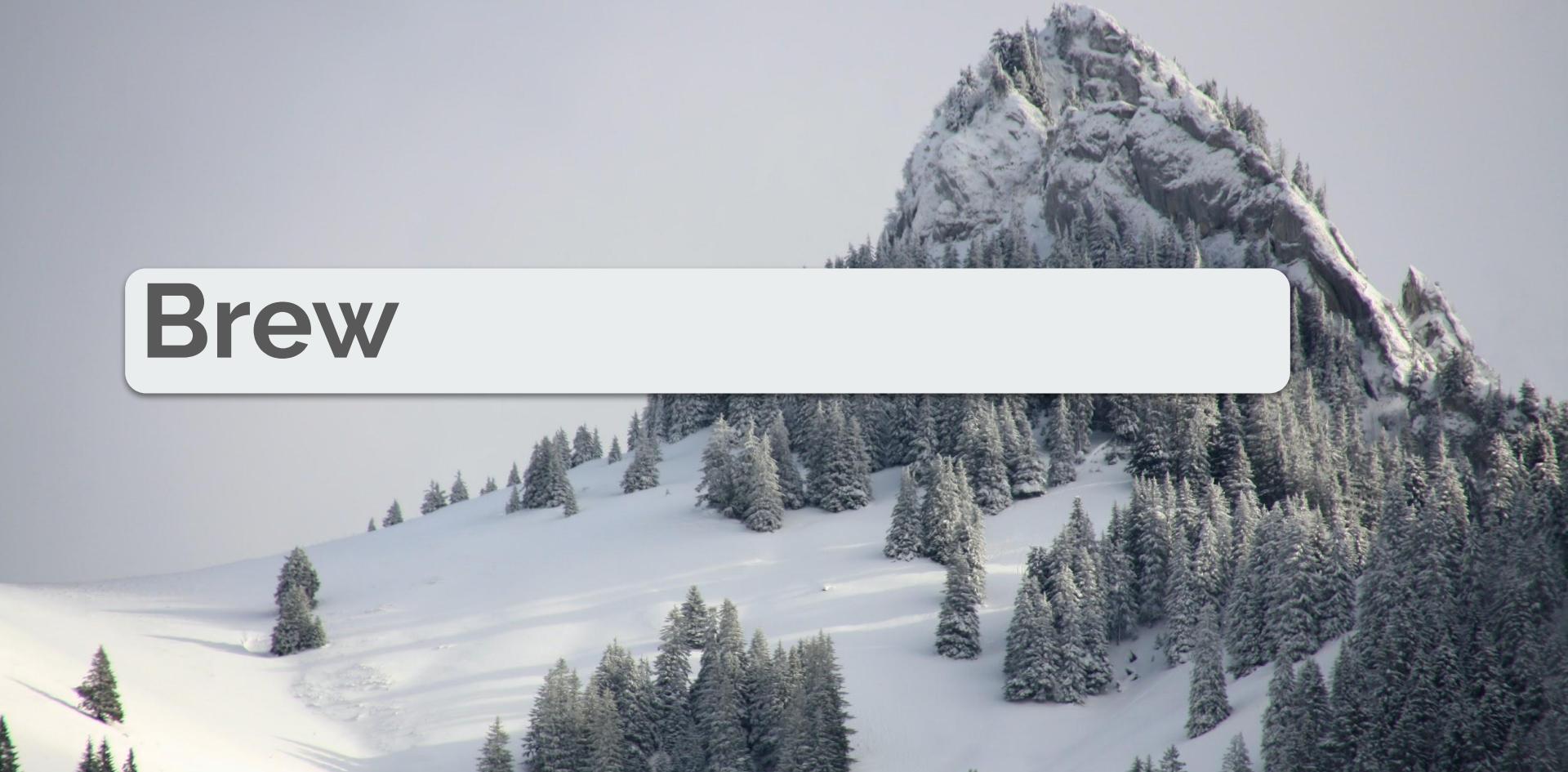


Pause pour les questions





Brew





```
Removing: /opt/homebrew/Cellar/python@3.12/3.12.4... (2,762 files, 48.2MB)
Warning: Permission denied @ apply2files - /opt/homebrew/Cellar/python@3.12/3.12.4/Frameworks/
Python.framework/Versions/3.12/lib/python3.12/curses/_pycache_/_init__.cpython-312.pyc
==> This operation has freed approximately 48.2MB of disk space.
Error: Could not cleanup old kegs! Fix your permissions on:
/opt/homebrew/Cellar/python@3.12/3.12.4

~ took 2m38s
> ls -larth $(brew --prefix)/Cellar/python@3.12
total 0
drwxrwxr-x  42 daparicio  admin   1.3K Jul 30 10:16 ..
drwxr-xr-x@  4 daparicio  admin  128B Aug 13 22:47 .
drwxr-xr-x@ 14 daparicio  admin  448B Aug 13 22:47 3.12.5
drwxr-xr-x@  5 daparicio  admin  160B Aug 13 22:47 3.12.4

~
> ls -larth $(brew --prefix)/Cellar/asitop
total 0
drwxr-xr-x@  3 daparicio  admin   96B May 29 20:49 .
drwxr-xr-x@  9 daparicio  admin  288B May 29 20:49 0.0.24
drwxrwxr-x  42 daparicio  admin   1.3K Jul 30 10:16 ..

~
> find /opt/homebrew -exec stat -f "%Su %N" {} + | grep ^root
root /opt/homebrew/Cellar/python@3.12/3.12.4/Frameworks/Python.framework/Versions/3.12/lib/python3.12/
curses/_pycache_-
root /opt/homebrew/Cellar/python@3.12.4/Frameworks/Python.framework/Versions/3.12/lib/python3.12/
curses/_pycache_/_init__.cpython-312.pyc
root /opt/homebrew/Cellar/python@3.12.4/Frameworks/Python.framework/Versions/3.12/lib/python3.12/
curses/_pycache_/_has_key.cpython-312.pyc
root /opt/homebrew/Cellar/python@3.12.4/Frameworks/Python.framework/Versions/3.12/lib/python3.12/
__pycache__/_tty.cpython-312.pyc

~
> sudo chown -R $(whoami) $(brew --prefix)/Cellar/python@3.12
sudo chown -R $(whoami) $(brew --prefix)/Cellar/asitop
Password:
Password:

~ took 5s
> sysup
==> Updating Homebrew...
Already up-to-date.
Removing: /opt/homebrew/Cellar/python@3.12/3.12.4... (2,762 files, 48.2MB)
==> This operation has freed approximately 48.2MB of disk space.
```



<https://nono.ma/brew-error-could-not-clean-up-old-kegs-fix-your-permissions>



#MeetupPython
#PereCastor #OVHcloud

@dadideo



75



<https://github.com/Homebrew/homebrew-core/issues/174620>

Running formulae as `sudo` breaks `python@3.12` permissions #174620

Closed

#174728



z4pf1sh opened on Jun 15, 2024

...

`brew gist-logs <formula> link OR brew config AND brew doctor output`

```
HOMEBREW_VERSION: 4.3.5
ORIGIN: https://github.com/Homebrew/brew.git
HEAD: 3f08c75e7b950d4340dab462f3e7f77e8093fa2b
Last commit: 4 days ago
Core tap JSON: 15 Jun 08:56 UTC
Core cask tap JSON: 15 Jun 08:56 UTC
HOMEBREW_PREFIX: /opt/homebrew
HOMEBREW_CASK_OPTS: []
HOMEBREW_MAKE_JOBS: 8
HOMEBREW_NO_AUTO_UPDATE: set
Homebrew Ruby: 3.3.2 => /opt/homebrew/Library/Homebrew/vendor/portable-ruby/3.3.2/bin/ruby
CPU: octa-core 64-bit arm_ibiza
Clang: 15.0.0 build 1500
Git: 2.39.3 => /Library/Developer/CommandLineTools/usr/bin/git
Curl: 8.6.0 => /usr/bin/curl
macOS: 14.5-arm64
CLT: 15.3.0.0.1.1708646388
Xcode: 15.4
Rosetta 2: false

Your system is ready to brew.
```



Assigned

No one a

Labels

bug

Type

No type

Projects

No projec

Milestone

No miles

Relations

None yet

Developers

asitc
Home

Notifications

You're not in this thread.

Participants

Verification

- My brew doctor output says Your system is ready to brew. and am still able to reproduce my issue.
- I ran brew update and am still able to reproduce my issue.
- I have resolved all warnings from brew doctor and that did not fix my problem.
- I searched for recent similar issues at <https://github.com/Homebrew/homebrew-core/issues?q=is%3Aissue> and found no duplicates.

What were you trying to do (and why)?

In my recent brew update (with `brew update && brew upgrade`), `python@3.12` was among those updated.



GitHub #174620



Permissions





Same player, Shoot again !





Reproducible ?





Demo

```
..p-python-lyon (-zsh)
```

```
Last login: Wed Feb 12 21:25:02 on ttys000
```

```
▶
```

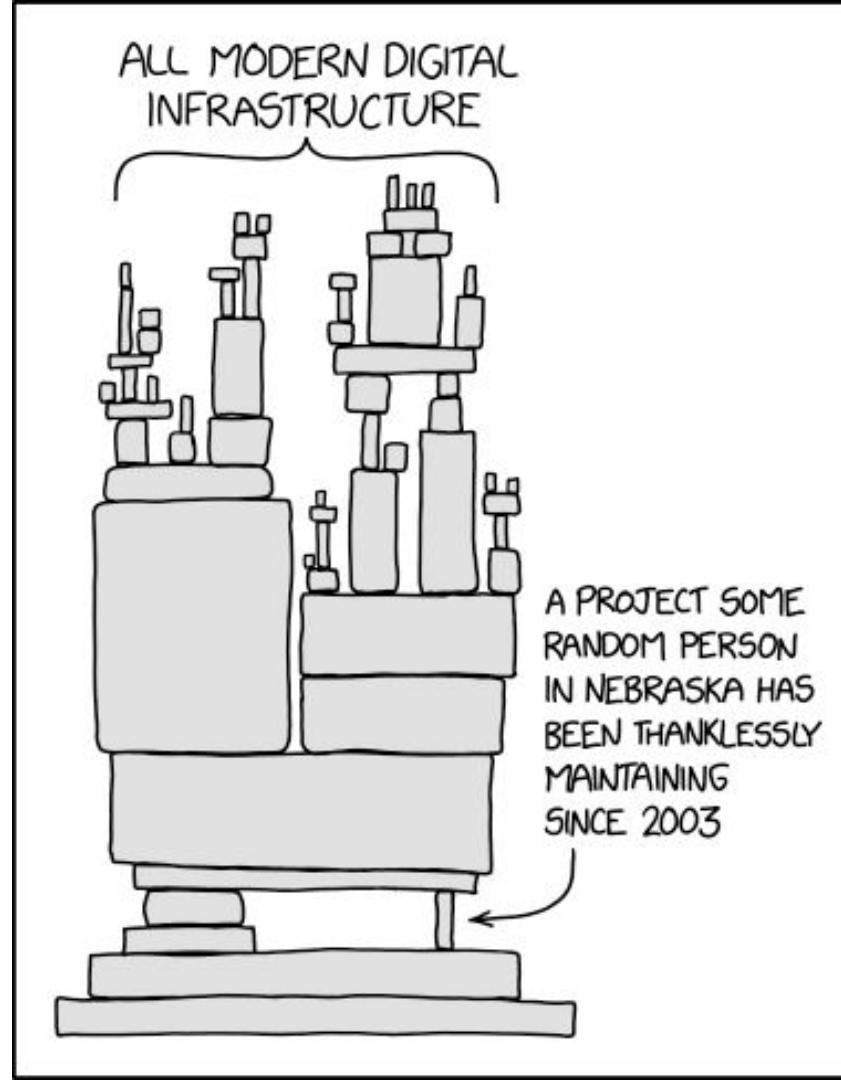
```
~
```

```
› nix-shell --packages python312Packages.diagrams --pure  
bck-i-search: nix_
```



Versioning







<https://github.com/netplex/json-smart-v2/issues/240#issuecomment-2653054828>

Closed

2.5.2 Release Breaking Upstream Dependencies #240

ConsenSys/line... #1801



hezhangjian 6 hours ago · edited by hezhangjian

Edits

Collaborator

...

`maven-metadata.xml` might be updated by central support. They reply an email to me and [@UrielCh](#)
I think we can close this issue, feel free to comment if still have problems.

```
from: Support <central-support@sonatype.com>
reply-to: Support <central-support@sonatype.com>
to: Zhangjian He <hezhangjian97@gmail.com>
cc: Uriel Chemouni <uriel.chemouni@gmail.com>
date: Feb 12, 2025, 11:36 PM
subject: [Central Support by Sonatype] Re: Inquiry About JsonSmart Maven Metadata Versioning
mailed-by: sonatype.com
```



We believe that we have successfully addressed all of the issues with the `net.minidev` namespace's `maven-metadata`.

Thank you,
The Central Team



20

2

1



Spring Issue



Python Support

Pillow supports these Python versions.

Python	3.10	3.9	3.8	3.7	3.6	3.5	3.4	2.7
Pillow >= 9.0	Yes	Yes	Yes	Yes				
Pillow 8.3.2 - 8.4	Yes	Yes	Yes	Yes	Yes			
Pillow 8.0 - 8.3.1		Yes	Yes	Yes	Yes			
Pillow 7.0 - 7.2			Yes	Yes	Yes	Yes		
Pillow 6.2.1 - 6.2.2			Yes	Yes	Yes	Yes		Yes
Pillow 6.0 - 6.2.0				Yes	Yes	Yes		Yes
Pillow 5.2 - 5.4				Yes	Yes	Yes	Yes	Yes

Python	3.6	3.5	3.4	3.3	3.2	2.7	2.6	2.5	2.4
Pillow 5.0 - 5.1	Yes	Yes	Yes			Yes			
Pillow 4	Yes	Yes	Yes	Yes		Yes			
Pillow 2 - 3		Yes	Yes	Yes	Yes	Yes	Yes		
Pillow < 2						Yes	Yes	Yes	Yes

Python Support

Pillow supports these Python versions.

Python	3.13	3.12	3.11	3.10	3.9	3.8	3.7	3.6	3.5
Pillow >= 11	Yes	Yes	Yes	Yes	Yes				
Pillow 10.1 - 10.4		Yes	Yes	Yes	Yes	Yes			
Pillow 10.0			Yes	Yes	Yes	Yes			
Pillow 9.3 - 9.5			Yes	Yes	Yes	Yes	Yes		
Pillow 9.0 - 9.2				Yes	Yes	Yes	Yes		
Pillow 8.3.2 - 8.4					Yes	Yes	Yes	Yes	
Pillow 8.0 - 8.3.1						Yes	Yes	Yes	
Pillow 7.0 - 7.2							Yes	Yes	Yes

Python	3.8	3.7	3.6	3.5	3.4	3.3	3.2	2.7	2.6	2.5	2.4
Pillow 6.2.1 - 6.2.2	Yes	Yes	Yes	Yes						Yes	
Pillow 6.0 - 6.2.0		Yes	Yes	Yes						Yes	
Pillow 5.2 - 5.4		Yes	Yes	Yes	Yes					Yes	
Pillow 5.0 - 5.1			Yes	Yes	Yes					Yes	
Pillow 4				Yes	Yes	Yes	Yes			Yes	
Pillow 2 - 3					Yes	Yes	Yes	Yes	Yes	Yes	
Pillow < 2								Yes	Yes	Yes	Yes



[Pillow Python Support](#)



GPU (Linux)

VERSION	PYTHON VERSION	COMPILER	BUILD TOOLS	CUDNN	CUDA
tensorflow-2.9.0	3.7-3.10	GCC 9.3.1	Bazel 5.0.0	8.1	11.2
tensorflow-2.8.0	3.7-3.10	GCC 7.3.1	Bazel 4.2.1	8.1	11.2
tensorflow-2.7.0	3.7-3.9	GCC 7.3.1	Bazel 3.7.2	8.1	11.2
tensorflow-2.6.0	3.6-3.9	GCC 7.3.1	Bazel 3.7.2	8.1	11.2
tensorflow-2.5.0	3.6-3.9	GCC 7.3.1	Bazel 3.7.2	8.1	11.2
tensorflow-2.4.0	3.6-3.8	GCC 7.3.1	Bazel 3.1.0	8.0	11.0
tensorflow-2.3.0	3.5-3.8	GCC 7.3.1	Bazel 3.1.0	7.6	10.1
tensorflow-2.2.0	3.5-3.8	GCC 7.3.1	Bazel 2.0.0	7.6	10.1
tensorflow-2.1.0	2.7, 3.5-3.7	GCC 7.3.1	Bazel 0.27.1	7.6	10.1
tensorflow-2.0.0	2.7, 3.3-3.7	GCC 7.3.1	Bazel 0.26.1	7.4	10.0
tensorflow_gpu-1.15.0	2.7, 3.3-3.7	GCC 7.3.1	Bazel 0.26.1	7.4	10.0
tensorflow_gpu-1.14.0	2.7, 3.3-3.7	GCC 4.8	Bazel 0.24.1	7.4	10.0
tensorflow_gpu-1.13.1	2.7, 3.3-3.7	GCC 4.8	Bazel 0.19.2	7.4	10.0
tensorflow_gpu-1.12.0	2.7, 3.3-3.6	GCC 4.8	Bazel 0.15.0	7	9





Data tooling





topology_script.py python3

All Videos Images News Maps More

About 812 results (0.40 seconds)

<https://community.cloudera.com> › td-p ...

Use of Python version 3 scripts for pyspark with HDP 2.4

Aug 30, 2019 — ... and we face an issue when running a Python 3 script that use - 141125. ...

ScriptBasedMapping: Exception running /etc/hadoop/conf/topology_script.py ...

[Python3.x Compatibility in HDP 2.6.x and 3.x](#) Jan 15, 2019

[topology.py not Python 3 compatible](#) - Cloudera Community Feb 14, 2017

[File "/etc/hadoop/conf/topology_script.py"](#), line 63 print rack Jan 16, 2018

[Getting error in Python3 kernel for pyspark with HDP 2.6.2](#) Oct 4, 2018

[More results from community.cloudera.com](#)

<https://stackoverflow.com> › questions › how-to-enable-... ...

How to Enable Python3 Support on HDP 2.6 - Stack Overflow

Oct 18, 2018 · 1 answer

The restriction are in few files. /usr/bin/hdp-select; /etc/hadoop/conf/topology_script.py. The 2to3 application can be used to convert ...

[How to ignore spark-submit warnings for pyspark](#) - Stack ... Oct 19, 2018

[PySpark: Permission denied although permission is given](#) Feb 14, 2018

[More results from stackoverflow.com](#)



Quick modeling of Survived on TrainSnow_prepared



Predict Survived (Binary classification)

DESIGN

RESULT

BASIC

Target

Train / Test Set

Metrics

Debugging

FEATURES

Features handling

Feature generation

Feature reduction

MODELING

Algorithms

Hyperparameters

ADVANCED

Runtime environment

Weighting strategy

Probability calibration

Target

Prediction type

Two-class classification

RE-DETECT SETTINGS

Target

Survived

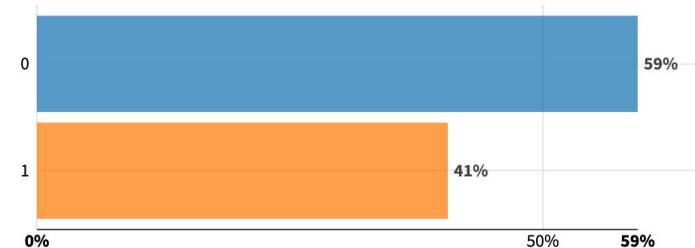
Partitioned models

Partitioning

 Not available: input dataset is not partitioned.

Target classes

Proportions of classes in the guess sample

If some classes are missing here (because they were not in the guess sample), you can [manually edit the mapping](#) (advanced)



Quick modeling of Survived on TrainSnow_prepared / Models / Random forest

Report

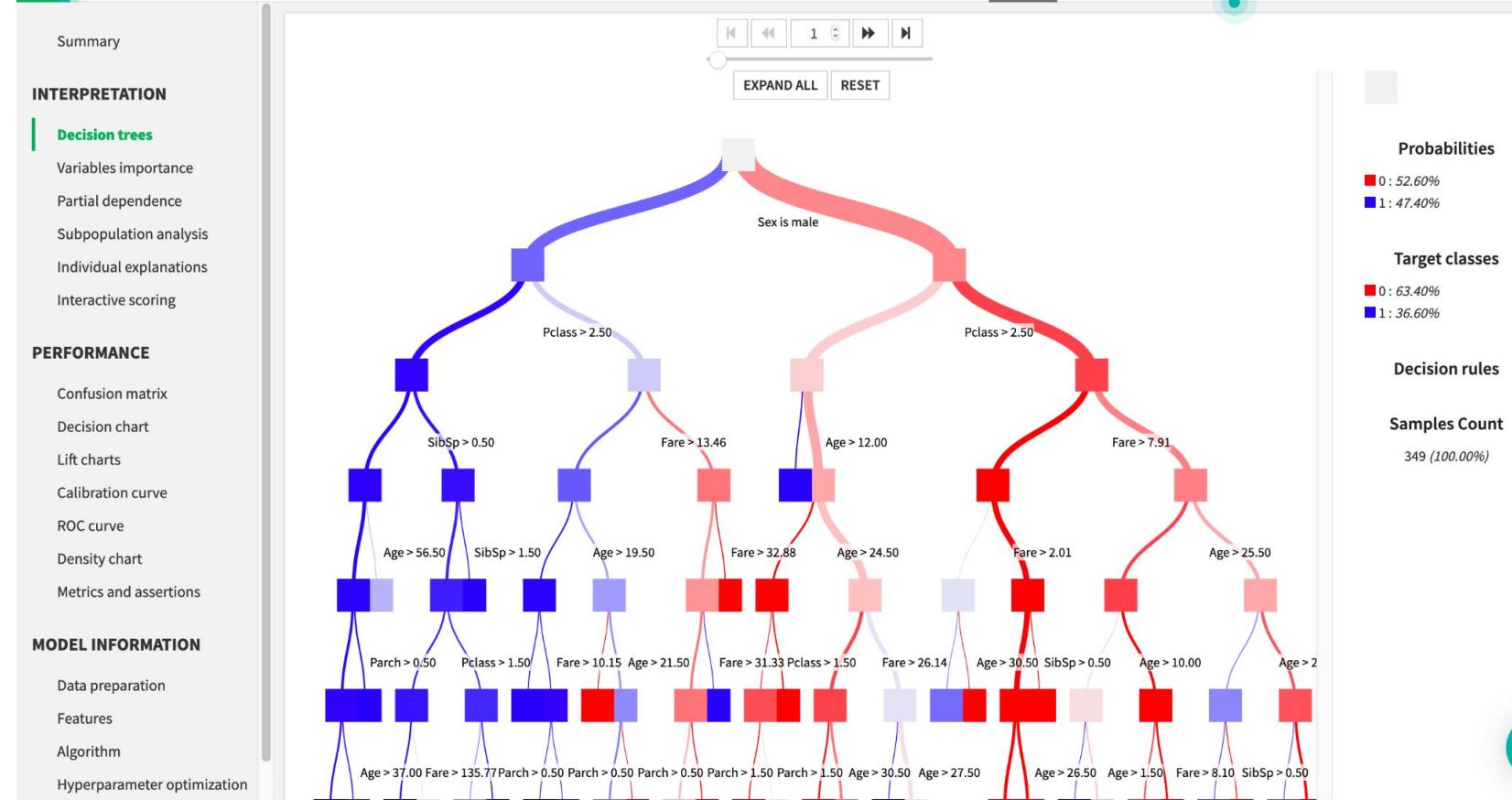
Predicted data

Charts

PUBLISH

DEPLOY

A





#MeetupPython
#PereCastor #OVHcloud

@dadideo



89



<input type="checkbox"/>	SESSION 1	:
<input type="checkbox"/>	Random forest	🏆 0.878 ⭐
<input type="checkbox"/>	Logistic Regression	0.871 ⭐
<input type="checkbox"/>	Decision Tree	0.820 ⭐
<input type="checkbox"/>	Artificial Neural Network	0.862 ⭐
<input type="checkbox"/>	LASSO-LARS	0.871 ⭐



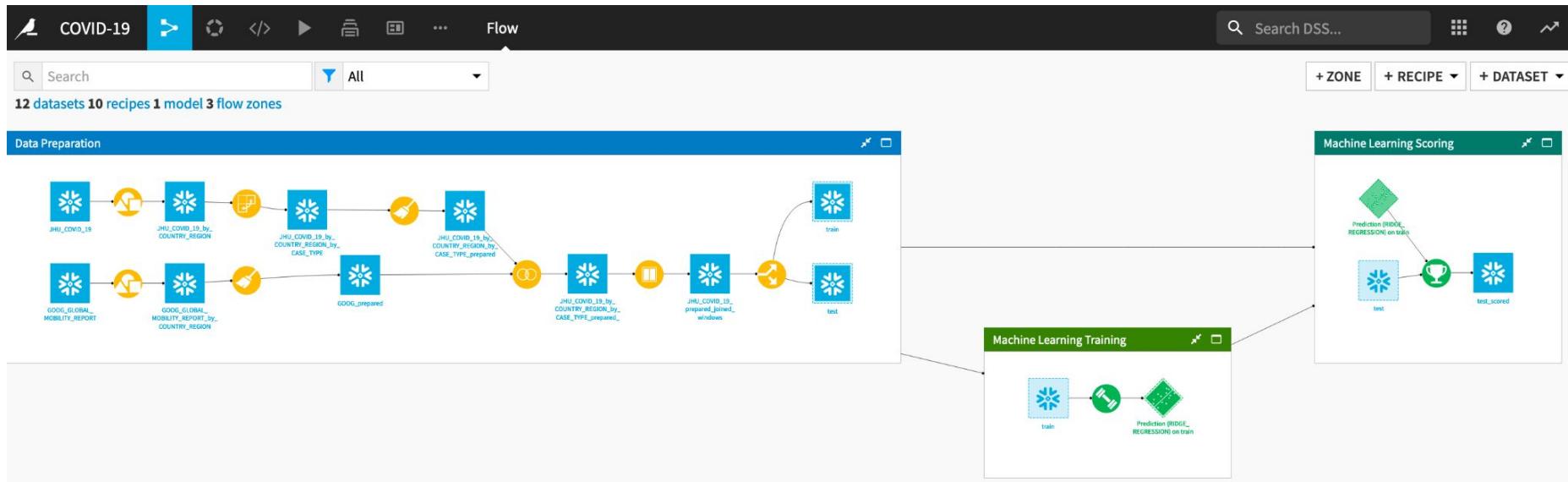


#MeetupPython
#PereCastor #OVHcloud

@dadideo



90





Script output

714 rows, 7 cols (714 - 177)

Survived

	bigint	Int
0	0	
0	0	
1	1	
1	0	
0	0	
0	0	
0	0	
0	0	
0	0	
1	1	
1	1	
0	0	
0	0	
1	1	

Recipe engine

Local stream

In-database (SQL) Run in database

+ 4 non-selectable engines Hide

Hadoop MapReduce The input dataset TITANICML.TrainSnow is not stored on HDFS

Spark (Optimized) Run on Spark
Forbidden by configuration

Hive Run on Hive
Forbidden by configuration

Impala Run on Impala
Forbidden by configuration

RESET ENGINE TO DEFAULT CLOSE

AutoML Let Dataiku create your models.

Quick Prototypes

Get some models, generic and quick.

In-memory ▾

In-memory

In-memory training using Python libraries (scikit-learn)

MLlib (Unavailable)

Distributed ML using Spark MLlib
Spark is not enabled

H2O (Unavailable)

Distributed ML using H2O Sparkling Water
Spark is not enabled.



There is no specific way to call multiprocessing in a Python recipe within DSS. Please note that python code being ran in DSS is the same as python being ran outside of DSS, we don't do anything special to it.

Are you referring to "python recipes" or "pyspark recipes"?

If you're referring to "python recipes", then I can confirm that it is not possible to offload the computation via YARN; instead, DSS offers containerized execution capabilities via Kubernetes. To start, my suggestion would be to review the "Elastic AI Computation" section of our documentation, in particular the "Concepts" page; afterwards, please let me know if you have any questions.



```
# Run on a YARN cluster in cluster deploy mode
export HADOOP_CONF_DIR=XXX
./bin/spark-submit \
--class org.apache.spark.examples.SparkPi \
--master yarn \
--deploy-mode cluster \
--executor-memory 20G \
--num-executors 50 \
/path/to/examples.jar \
1000

# Run a Python application on a Spark standalone cluster
./bin/spark-submit \
--master spark://207.184.161.138:7077 \
examples/src/main/python/pi.py \
1000

# Run on a Mesos cluster in cluster deploy mode with supervise
./bin/spark-submit \
--class org.apache.spark.examples.SparkPi \
--master mesos://207.184.161.138:7077 \
--deploy-mode cluster \
--supervise \
--executor-memory 20G \
--total-executor-cores 100 \
http://path/to/examples.jar \
1000

# Run on a Kubernetes cluster in cluster deploy mode
./bin/spark-submit \
--class org.apache.spark.examples.SparkPi \
--master k8s://xx.yy.zz.wv:443 \
--deploy-mode cluster \
--executor-memory 20G \
--num-executors 50 \
http://path/to/examples.jar \
1000
```



Juste pour info j'ai cette erreur dans l'execution d'un notebook d'un user dataiku :

```
Exception: Python in worker has different version 2.7 than that in driver 3.6, PySpark cannot run with
different minor versions. Please check environment variables PYSPARK PYTHON and PYSPARK DRIVER PYTHON are
correctly set.
```

J'ai testé plusieurs versions de python sur le Kernel Jupyter et j'ai toujours cette erreur

Thank you for contacting Datiaku Support. The error suggests that the worker nodes are using Python version 2.7 while DSS is using Python 3.6. This setup is not compatible.

You will need to ensure::

1. That Python 3.6 is installed across all of your worker nodes. If it's not, you will need to install it on all of them.
2. Specify the absolute path to the Python 3.6 binary on your worker nodes in DSS under Administration > Code Envs > {Your Code Env} > Spark > Yarn Python Executable.

En gros pour chaque env on peut lui spécifier le paht



Version	Released	Security Support	Release
3.10	2 months and 2 weeks ago (04 Oct 2021)	Ends in 4 years and 9 months (04 Oct 2026)	3.10.1
3.9	1 year and 2 months ago (05 Oct 2020)	Ends in 3 years and 9 months (05 Oct 2025)	3.9.9
3.8	2 years ago (14 Oct 2019)	Ends in 2 years and 9 months (14 Oct 2024)	3.8.12
3.7	3 years and 5 months ago (27 Jun 2018)	Ends in 1 year and 6 months (27 Jun 2023)	3.7.12
3.6	4 years and 12 months ago (23 Dec 2016)	Ends in 3 days (23 Dec 2021)	3.6.15
3.5	6 years ago (30 Sep 2015)	Ended 1 year and 3 months ago (13 Sep 2020)	3.5.10
3.4	7 years ago (16 Mar 2014)	Ended 2 years and 9 months ago (18 Mar 2019)	3.4.10
3.3	9 years ago (29 Sep 2012)	Ended 4 years ago (29 Sep 2017)	3.3.7
2.7	11 years ago (03 Jul 2010)	Ended 1 year and 11 months ago (01 Jan 2020)	2.7.18



Python is an interpreted, high-level, general-purpose programming language.



Hello,

Pour info nous avons contacté le support Dataiku sur la question de la fin de vie de Python 3.6 (<https://endoflife.date/python>).

Leur réponse : "DSS8 does not support Py37. It may work but we cannot guarantee there will be no issues.

You need to upgrade to DSS9/10 to have the full support of this Python interpreter version.".

Nous ne pouvons pas upgrader DSS car non compatible avec HDP2.6.



network tf

ⓘ Installed terraform-ls 0.19.0.



Source: HashiCorp Terraform (Extension)

[View Changelog](#)

✖ Python version is too old to run Snooty language server. Must use ...

> NPM SCRIPTS

⚠ Go Update Available



✖ Python version is too old to run Snooty language server. Must use 3.7 and above.



Source: reStructuredText (Extension)

> NPM SCRIPTS



Dataiku: GLIBCXX_3.4.30 not found

GCC not found

make not found

Rust not found

Because python / cryptography are too old



```
date && sudo chattr -i /usr/bin/python3.6 && ls -larth /usr/bin/python3 /usr/bin/python3.6 && sudo chown root:root /usr/bin/python3.6 && sudo chown -h root:root /usr/bin/python3 && sudo chmod 755 /usr/bin/python3.6 && sudo chmod 755 /usr/bin/python3 && ls -larth /usr/bin/python3 /usr/bin/python3.6
```



#MeetupPython
#PereCastor #OVHcloud

@dadideo



100



```
sudo chattr +i /usr/bin/python3.6
```





```
(fullhunt) ~/c/g/f/log4j-scan >>> conda install --file requirements.txt
Collecting package metadata (current_repodata.json): done
Solving environment: failed with initial frozen solve. Retrying with flexible solve.
Solving environment: failed with repodata from current_repodata.json, will retry with next repodata source.
Collecting package metadata (repodata.json): done
Solving environment: failed with initial frozen solve. Retrying with flexible solve.
Solving environment: \
Found conflicts! Looking for incompatible packages.
This can take several minutes. Press CTRL-C to abort.
failed

UnsatisfiableError: The following specifications were found
to be incompatible with the existing python installation in your environment:

Specifications:

- pycryptodome → python[version='≥2.7,<2.8.0a0|≥3.6,<3.7.0a0|≥3.7,<3.8.0a0|≥3.8,<3.9.0a0|≥3.9,<3.10.0a0']
- termcolor → python[version='≥2.7,<2.8.0a0|≥3.6,<3.7.0a0|≥3.7,<3.8.0a0|≥3.8,<3.9.0a0|≥3.9,<3.10.0a0|≥3.5,<3.6.0a0']

Your python: python=3.10

If python is on the left-most side of the chain, that's the version you've asked for.
When python appears to the right, that indicates that the thing on the left is somehow
not available for the python version you are constrained to. Note that conda will not
change your python version to a different minor version unless you explicitly specify
that.

The following specifications were found to be incompatible with each other:

Output in format: Requested package → Available versions

Package bzip2 conflicts for:
python=3.10 → bzip2[version='≥1.0.8,<2.0a0']
requests → python[version='≥3.6'] → bzip2[version='≥1.0.8,<2.0a0']

(fullhunt) ~/c/g/f/log4j-scan >>>
```



#MeetupPython
#PereCastor #OVHcloud

@dadideo



103



github.com/psf/requests/issues/5901

Closed

Cannot import requests on python 3.10 rc1 #5901

rkhwaja opened this issue on 10 Aug · 1 comment



nateprewitt commented on 10 Aug

Member



Hi @rkhwaja, the failure you're receiving is for a code path that hasn't existed in urllib3 for over 4 years, so the solution would be to update to a newer version (>1.23 at the minimum).

This isn't an issue with Requests and we can't raise our pin version range without breaking a large portion of the Python ecosystem. There shouldn't be anything inhibiting you from installing the correct version of urllib3 though.



1



Versions & YAML





⤵ Pierre Zemb Retweeted



Chris Holdgraf @choldgraf · Oct 9

Just encountered my first Python 3.10 gotcha!

In YAML configuration for GitHub actions, you might have your Python matrix like this:

3.7, 3.8, 3.9

Guess what? 3.10 resolves to 3.1

you need to use 3.7, 3.8, 3.9, "3.10"

oh, YAML



Futur talk ?





#MeetupPython
#PereCastor #OVHcloud

♥ Idriss Neumann ☠ liked



Olivier Poncet
@ponceto91

@dadideo



107



Bonjour les piou-pious, pour les amateurs de Python, Microsoft Pyjion arrive en version 1.0. C'est un package Python s'appuyant sur .NET 6 et sa JIT analysant le code Python et l'optimisant à la volée permettant d'obtenir des performances plus élevées

[Translate Tweet](#)



Tweet your reply



Developpez.com

twitter.com/ponceto91



#MeetupPython
#PereCastor #OVHcloud

@dadideo



108



← → ⌛ https://github.com/microsoft/Pyzion

Code Issues 38 Pull requests 2 Actions Projects Security Insights

This repository has been archived by the owner on Oct 12, 2022. It is now read-only.

Pyzion

Public archive

Watch 106 Fork 130 Star 1.6k

master 3 Branches 0 Tags Go to file Code

brettcannon Make the move more visible · 47f20e7 · 5 years ago 582 Commits

.github/workflows rename github actions (#239) · 5 years ago

CoreCLR @ c6de64f Use release/1.0.0-rc2 branch of CoreCLR, and fix resultin... · 9 years ago

Docs Fix simple typo: targetting -> targeting (#236) · 6 years ago

Patches/CoreCLR Remove Python patches · 8 years ago

Perf Update benchmark info · 9 years ago

Pyzion Revert "Linux" · 7 years ago

Python @ 2773add Move tracking branch to 3.6 · 8 years ago

Test Get rid of TraceInfo, move state into PyzionJittedCode · 8 years ago

Tests Fixup formatting · 8 years ago

Tools Fix generated output to handle when a type doesn't lead ... · 10 years ago

.gitattributes Update CoreCLR to a newer commit so that it will build u... · 10 years ago

.gitignore Don't include Sphinx build output in wc · 9 years ago

.gitmodules Use the release branch of CoreCLR · 9 years ago

BuildDebugPython.bat Make building from the command-line much easier · 9 years ago

BuildDeps.cmd Use release/1.0.0-rc2 branch of CoreCLR, and fix resultin... · 9 years ago

CODE_OF_CONDUCT.md Add code of conduct referenced in CONTRIBUTING.md · 8 years ago

CONTRIBUTING.md Make building from the command-line much easier · 9 years ago

CopyFiles.bat Fixup EXTENDED_ARG handling for branches · 8 years ago

DebugBuild.bat Make building from the command-line much easier · 9 years ago

About

Pyzion - A JIT for Python based upon CoreCLR

- Readme
- MIT license
- Code of conduct
- Security policy
- Activity
- Custom properties
- 1.6k stars
- 106 watching
- 130 forks

Report repository

Releases

No releases published

Packages

No packages published

Contributors 16

+ 2 contributors

Languages

C++ 97.6% Other 2.4%

 GitHub / Pyzion



#MeetupPython
#PereCastor #OVHcloud



Samuel Path | Angular Infinida...
@smlpth

@dadideo



109



J'entends surtout des devs C#/Java parler de DDD, Event Sourcing, CQRS, etc., et plus rarement des devs Python/Ruby/Go/JS, etc. Je me demande s'il s'agit de raisons essentielles aux langages et écosystèmes, ou plutôt accidentnelles. Si vous avez des idées...

[Translate Tweet](#)

16:45 · 09 Nov 21 · [Twitter Web App](#)



twitter.com/smlpth



J'entends surtout des devs C#/Java parler de DDD, Event Sourcing, CQRS, etc., et plus rarement des devs Python/Ruby/Go/JS, etc. Je me demande s'il s'agit de raisons essentielles aux langages et écosystèmes, ou plutôt accidentnelles. Si vous avez des idées...

[Translate post](#)

4:45 PM · Nov 9, 2021



2



1



Post your reply

Reply



Samuel Path @smlpth · Nov 9, 2021



...

Intéressant, mais je vois tout de même pas mal de gros projets avec beaucoup de complexité métier partir sur du Ruby/Python par exemple. Et je suis curieux de voir leur manière à eux d'implémenter les concepts équivalents au DDD/CQRS/ES, etc.



Daniel Gonçalves, dealer de produits du t... @dag... · Nov 9, 2021



...

Peut-être est-ce lié à la typologie des projets? Je ne sais plus où j'ai lu ça, mais Microsoft réserve le DDD pour les applications les plus complexes: en dessous d'un certain seuil ce qu'on perd en temps de conception n'est pas pas rattrapé.



twitter.com/smlpth



#MeetupPython
#PereCastor #OVHcloud

@dadideo



111



Samuel Path | Angular Infinidash Certified 🚀 @smlpth · 14m

...

Dur de retourner à des langages à typage dynamique quand on s'est habitué au confort du typage statique... Je comprends Gary Bernhardt, méga expert Ruby et Python, qui a passé tout ses back-ends sur TypeScript (pour aussi avoir le même language back et front, un atout majeur).



2



3



Tip



Samuel Path | Angular Infinidash Certified 🚀 @smlpth · 36s

...

*tous ses back-ends
*le même langage

Tweet fast, make typos



twitter.com/smlpth



#MeetupPython
#PereCastor #OVHcloud

@dadideo



112



Teddy FERDINAND @TeddyFERDINAND1 · 3h

...

Python is the new PHP.

J'adore voir la rage envers ces langages qui pourtant portent énormément d'applications et de site web :)

Les deux souffrent du même souci : Ils sont très simples à prendre en main, et il est donc facile de faire de la merde avec...



2



1



9



[Show more replies](#)



Teddy FERDINAND @TeddyFERDINAND1 · 5m

...

Replying to @idriss_neumann

Je dois avouer que je ne connais pas assez node pour avoir un avis objectif :o



1



Idriss Neumann ✨🐧🌐💡 @idriss_neumann · 26s

...

Bah je pense qu'aujourd'hui c'est le plus mainstream et surtout le plus mal utilisé (python et php sont assez proche, on risque moins de bêtise en utilisant python comme php, en node ne pas changer de paradigme avec des apis synchrones et pull de threads s'est se tirer une balle)



#MeetupPython
#PereCastor #OVHcloud

10:41



@dadideo



113



← Tweet



Pierre-Antoine Grégoire ...
@zepag

Unpopular dev opinion (?): I like both Java and JavaScript. I like both Rust and Python. Usually what bugs me is my own bad knowledge of the underlying tooling/platform/ecosystem, or trying to map the experience of a platform to another.

8:25 · 17 Oct 21 · Twitter for Android

3 Retweets 15 Likes



Side Projects' Gardener ... · 1h

Replying to @zepag

The same



Tweet your reply





PEP 416 – Add a frozendict builtin type

Author: Victor Stinner <vstinner at python.org>

Status: Rejected

Type: Standards Track

Created: 29-Feb-2012

Python-Version: 3.3

► [Table of Contents](#)

Rejection Notice

I'm rejecting this PEP. A number of reasons (not exhaustive):

- According to Raymond Hettinger, use of frozendict is low. Those that do use it tend to use it as a hint only, such as declaring global or class-level "constants": they aren't really immutable, since anyone can still assign to the name.
- There are existing idioms for avoiding mutable default values.
- The potential of optimizing code using frozendict in PyPy is unsure; a lot of other things would have to change first. The same holds for compile-time lookups in general.
- Multiple threads can agree by convention not to mutate a shared dict, there's no great need for enforcement. Multiple processes can't share dicts.
- Adding a security sandbox written in Python, even with a limited scope, is frowned upon by many, due to the inherent difficulty with ever proving that the sandbox is actually secure. Because of this we won't be adding one to the stdlib any time soon, so this use case falls outside the scope of a PEP.

On the other hand, exposing the existing read-only dict proxy as a built-in type sounds good to me. (It would need to be changed to allow calling the constructor.) GvR.

Update (2012-04-15): A new `MappingProxyType` type was added to the `types` module of Python 3.3.



Containers





#MeetupPython
#PereCastor #OVHcloud

Jean-Louis Queguiner @JiliJeanlouis · 1h
Dockerhub build limitations :

- 2hours build
- 30Gb of disk space
- 2Gb of RAM
- 1vcpu

- paying won't raise limitation (to my knowledge)

- AI images are usually > 30Gb

- I don't want to reinvent the wheel

Any idea ?

hours to complete:

- 2 GB RAM
- 1CPU
- 30 GB Disk Space

So, for larger builds you have to either break them into several Automated Builds connected by FROM statements and Repository Links, or build them locally on your machine and push them to the repository.

Reference

- Build canceled without any explanation #1259
- Docker Hub timeout in automated build

3 1 1 1

Piou! 🍏 @Kedare · 13m
More than 30GB in your images ? What are you putting inside ?

1 1 1

Jean-Louis Queguiner @JiliJeanlouis · 7m
nvidia cuda, cudnn, all cuda libs, opencl, jax, python, TF, Torch, etc.. with a ubuntu:20.04 base

@dadideo



116





Python application images:

@dadideo



117



- from ubuntu:14.04 - 438MB => 16.8MB (minified by **25.99X**)
- from python:2.7-alpine - 84.3MB => 23.1MB (minified by **3.65X**)
- from python:2.7.15 - 916MB => 27.5MB (minified by **33.29X**)
- from centos:7 - 647MB => 23MB (minified by **28.57X**)
- from centos/python-27-centos7 - 700MB => 24MB (minified by **29.01X**)
- from python2.7:distroless - 60.7MB => 18.3MB (minified by **3.32X**)

Ruby application images:

- from ubuntu:14.04 - 433MB => 13.8MB (minified by **31.31X**)
- from ruby:2.2-alpine - 319MB => 27MB (minified by **11.88X**)
- from ruby:2.5.3 - 978MB => 30MB (minified by **32.74X**)

Golang application images:

- from golang:latest - 700MB => 1.56MB (minified by **448.76X**)
- from ubuntu:14.04 - 531MB => 1.87MB (minified by **284.10X**)
- from golang:alpine - 258MB => 1.56MB (minified by **165.61X**)
- from centos:7 - 615MB => 1.87MB (minified by **329.14X**)

Rust application images:

- from rust:1.31 - 2GB => 14MB (minified by **147.16X**)

JAVA application images:

- from ubuntu:14.04 - 743.6 MB => 100.3 MB



#MeetupPython
#PereCastor #OVHcloud

@dadideo



118



chainguard

Search Chainguard Images

Directory

Sign In

python

Last changed 1 day ago

FIPS available

Try the Latest Version for Free

```
docker pull cgr.dev/chainguard/python
```

Contact us.

Create your Free Account

Be the first to hear about exciting product updates, critical vulnerability alerts, compare alternative images, and more.

Sign Up

< Versions Overview Provenance Specifications SBOM Vulnerabilities >

Version: latest

Search...

CVE ID	Severity ↓	Package	Version	Last detected

This image is free of CVEs as of 3 hours ago!

Image updated 2025-02-11 11:09:17 PM GMT+1
Last scanned 2025-02-12 09:09:31 PM GMT+1

Chainguard Images are rebuilt daily from source and rapidly patched against new vulnerabilities.

Looking for a status of a CVE? Visit [Security advisories](#)



#MeetupPython
#PereCastor #OVHcloud

@dadideo



119

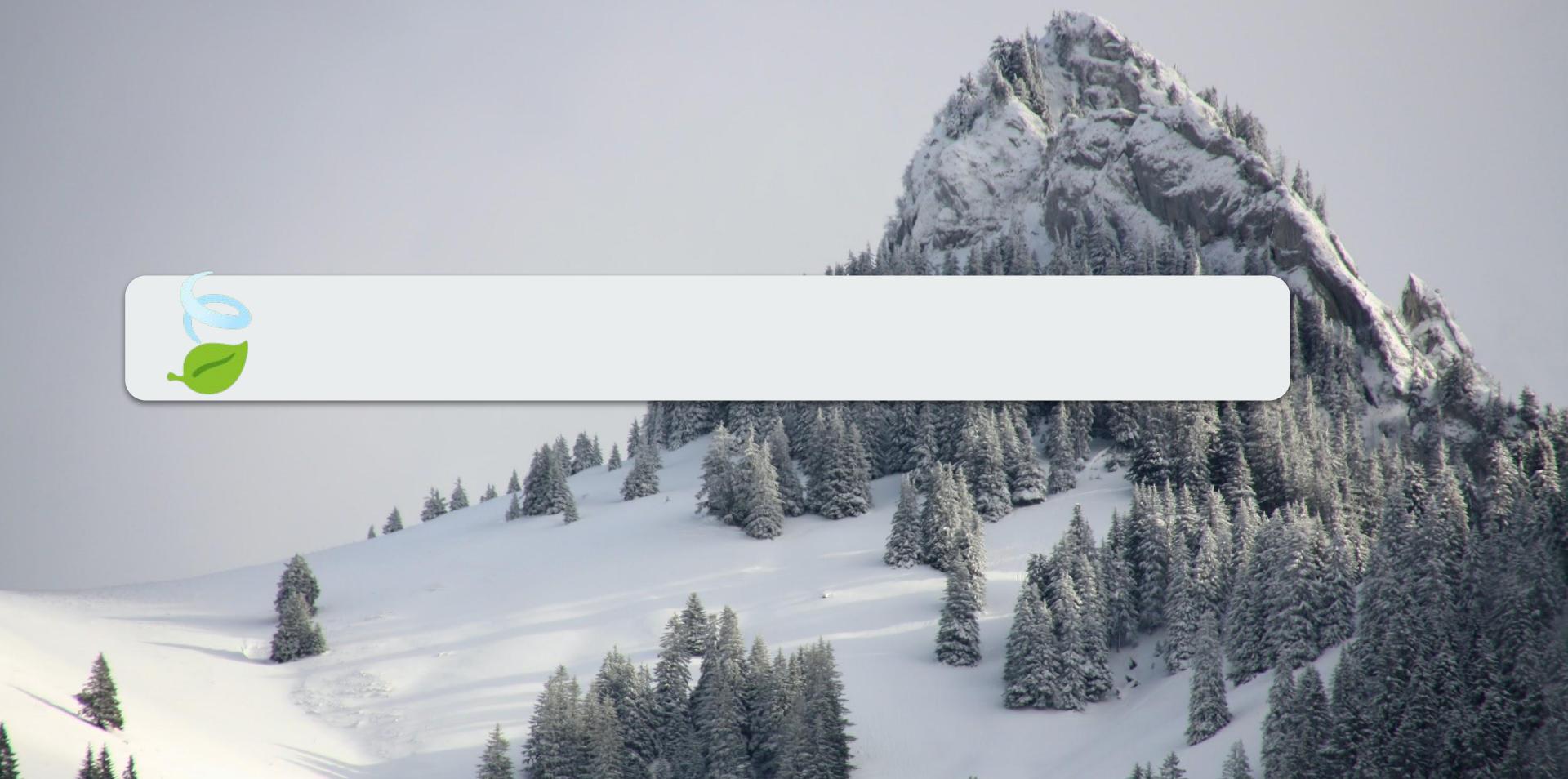




Table 4. Normalized global results for Energy, Time, and Memory



SLE 2017 Energy efficiency

	Total		
	Energy	Time	Mb
(c) C	1.00	(c) C	1.00
(c) Rust	1.03	(c) Rust	1.04
(c) C++	1.34	(c) C++	1.56
(c) Ada	1.70	(c) Ada	1.85
(v) Java	1.98	(v) Java	1.89
(c) Pascal	2.14	(c) Chapel	2.14
(c) Chapel	2.18	(c) Go	2.83
(v) Lisp	2.27	(c) Pascal	3.02
(c) Ocaml	2.40	(c) Ocaml	3.09
(c) Fortran	2.52	(v) C#	3.14
(c) Swift	2.79	(v) Lisp	3.40
(c) Haskell	3.10	(c) Haskell	3.55
(v) C#	3.14	(c) Swift	4.20
(c) Go	3.23	(c) Fortran	4.20
(i) Dart	3.83	(v) F#	6.30
(v) F#	4.13	(i) JavaScript	6.52
(i) JavaScript	4.45	(i) Dart	6.67
(v) Racket	7.91	(v) Racket	11.27
(i) TypeScript	21.50	(i) Hack	26.99
(i) Hack	24.02	(i) PHP	27.64
(i) PHP	29.30	(v) Erlang	36.71
(v) Erlang	42.23	(i) Jruby	43.44
(i) Lua	45.98	(i) TypeScript	46.20
(i) Jruby	46.54	(i) Ruby	59.34
(i) Ruby	69.91	(i) Perl	65.79
(i) Python	75.88	(i) Python	71.90
(i) Perl	79.58	(i) Lua	82.91

Table 1. CLBG corpus of programs.

Benchmark	Description	Input
n-body	Double precision N-body simulation	50M
fannkuch-redux	Indexed access to tiny integer sequence	12
spectral-norm	Eigenvalue using the power method	5,500
mandelbrot	Generate Mandelbrot set portable bitmap file	16,000
pidigits	Streaming arbitrary precision arithmetic	10,000
regex-redux	Match DNA 8mers and substitute magic patterns	fasta output
fasta	Generate and write random DNA sequences	25M
k-nucleotide	Hashtable update and k-nucleotide strings	fasta output
reverse-complement	Read DNA sequences, write their reverse-complement	fasta output
binary-trees	Allocate, traverse and deallocate many binary trees	21
chameneos-redux	Symmetrical thread rendezvous requests	6M
meteor-contest	Search for solutions to shape packing puzzle	2,098
thread-ring	Switch from thread to thread passing one token	50M



#MeetupPython
#PereCastor #OVHcloud

@dadideo



121



The screenshot shows a video player interface. At the top left is the BreizhCamp logo with the tagline "La conférence à l'Ouest". At the top right is the text "12^e EDITION". The main video frame features a woman speaking, with a white callout bubble containing the title "Optimisation de performance bénéfice ou sacrifice ?" and the speaker's name "Emmanuel-Lin TOULEMONDE". Below the video frame, a small R2-D2 sticker says "BREIZHCAMP, YOU'RE MY ONLY HOPE". The video player has a progress bar at 0:01 / 42:18 and various control icons like play, volume, and settings. The bottom right corner of the video frame shows the Octo Technology logo.

Optimisation de performance bénéfice ou sacrifice ? (Emmanuel-Lin Toulemonde)



BreizhCamp
5.7K subscribers

Subscribe

Like 8 | Share Save

457 views 5 months ago

Plus le code est rapide, plus le code est bon. Au-delà de cette phrase tarte à la crème, ce talk propose de voir quand il faut optimiser du code, comment s'y prendre, et à quel moment s'arrêter. Les premières optimisations relèvent des bonnes pratiques que tout le monde devrait savoir faire, les suivantes sont sacrificielles : elles dégradent la lisibilité, la maintenabilité du code. Pensé comme une déclinaison concrète de la loi de Eroom proposé par Tristan Nitot, ce talk, commencera avec un exemple de code bien mal codé comme nous l'avons tous déjà fait, puis au fil des ...more



O En résumé, les gains de performance

Version	Pic mémoire	Plus gros objet	Temps de calcul	Quand ?	Années Eroom
Dev	48 Mb	104 b	82s	Jamais ?	
Naïve	3 100 Mb	3 700 Mb	43s	Prototype / minidonnées	Référence
Frugale	-13%	-11%	-12%	Toujours	0,35
Parquet	-55%	-93%	-63%	Toujours	2
Chunk	-95%	-99,9%	-53%	Données > Mémoire	8
Parallélisé	-98%	-99,4%	-78%	Calculs lourds & CPU disponible	12
Pyspark	?	?	-18%	>> 10Gb / jours	?
Polars	?	-90%	-98%	Si besoin vraiment spécifique	11



#MeetupPython
#PereCastor #OVHcloud

@dadideo



123



Conclusion





Analogie

« Nul n'est censé ignorer la loi »





Ma devise

« Nul développeur n'est censé ignorer la sécurité »



De la lecture générale



#MeetupPython
#PereCastor #OVHcloud

@dadideo



126



[La Sécurité dès la conception \(Secure by design\)](#), Programmez! (Hors-série 8 Septembre/Octobre 2022)



#MeetupPython
#PereCastor #OVHcloud

@dadideo



127



Pour aller plus loin

The screenshot shows the Bright Data web interface. On the left, under 'Processes & Scraping', there's a 'Web Scrapers' section. In the center, a 'residential' proxy profile is selected. It shows 'Access Details' like 'Mode: Immediate access', 'Host: brd-scraper-01:3333', and 'Username: [redacted]'. Below that is the 'Usage overview' section, which includes a chart of bandwidth usage from December 2024 to January 2025. The chart shows a significant spike in bandwidth usage starting around January 1st. At the bottom, there's a table of metrics for November 2024, December 2024, January 2025, and January 19, 2025. To the right, a code editor window shows a Python script named 'main.py' with environment variables defined in '.env':

```
1 SE_BRIGHT_DATA=true
2 RIGHT_DATA_CUSTOMER=brd-customer-hl_60fc588c
3 RIGHT_DATA_PASSWORD=[REDACTED]
4 RIGHT_DATA_ZONE=
5 RIGHT_DATA_PORT=
6
```

Coder avec l'IA : scraping de cinéma en Python



Docstring
108K subscribers

Subscribed

173



Share



Thanks



Clip



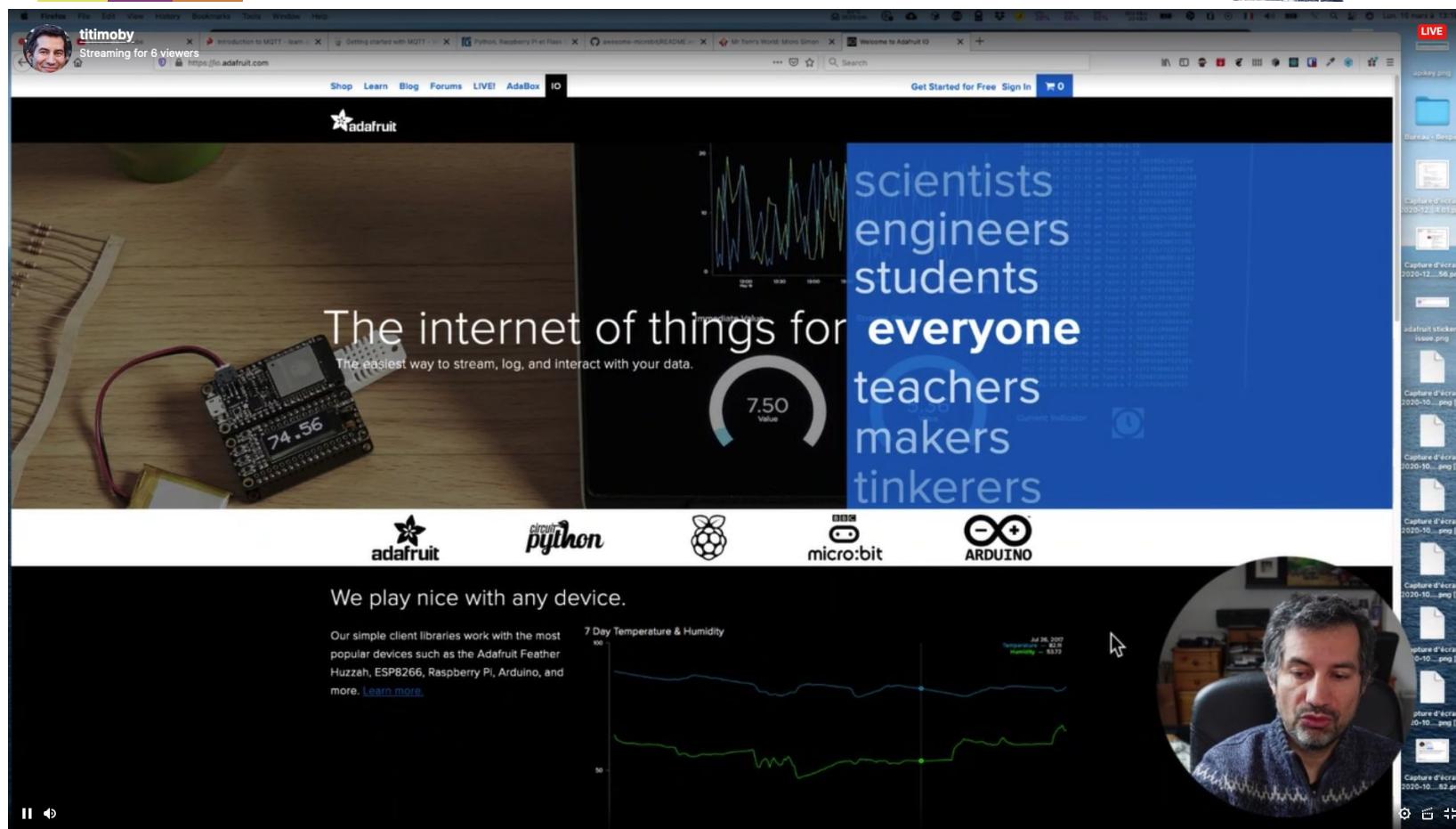
Save

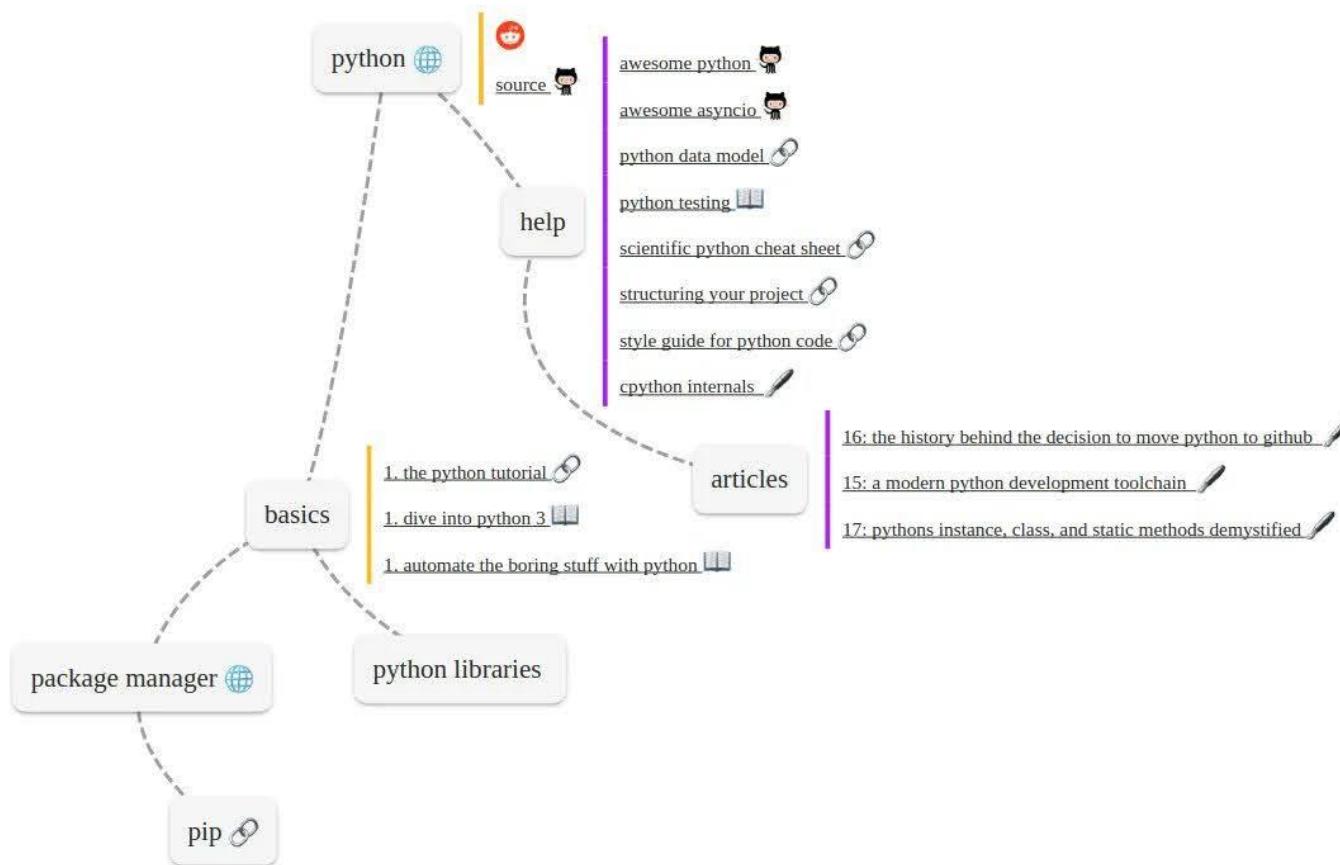


#MeetupPython
#PereCastor #OVHcloud



128

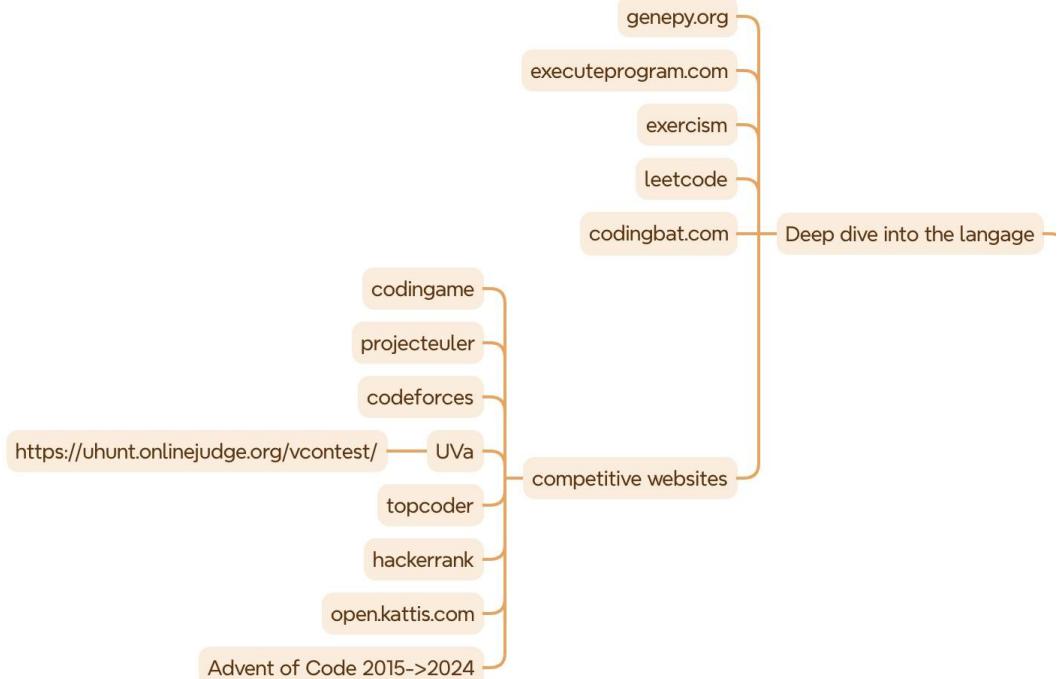








🎯 Pour aller plus loin





Passons à la pratique !

Plusieurs ressources pour apprendre le python en codant :

<https://open.kattis.com/> (aidez à remonter l'INSA dans le classement !)

<https://www.codingame.com/>

<https://projecteuler.net/>

Quand vous pensez maîtriser un peu plus le langage, essayez de faire des requêtes en utilisant `requests` (`challs` de programmation root-me par exemple), ou utiliser `socket` ou `pwn` pour faire des challenges de pwn !



Merci beaucoup aux
organisateurs & à OVHcloud
pour nous accueillir ce soir





Merci pour votre attention !

 N'oubliez pas de me donner votre avis sur cette session: <https://tinyurl.com/pythononlyon25>

 Lien des slides dans les commentaires 



Faites-vous confiance aux QR Codes ?