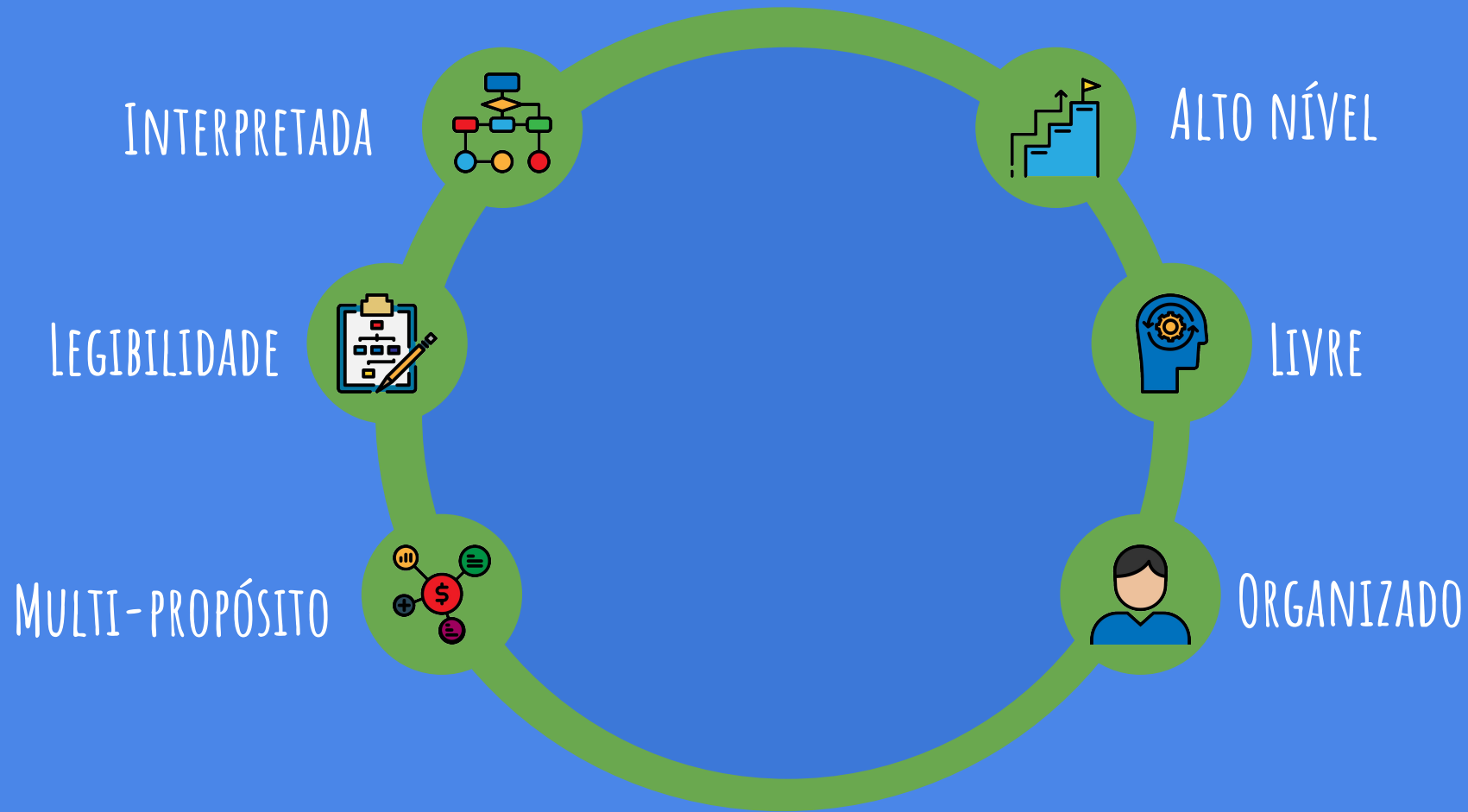


Fabrício Ribeiro

Rafael Ribeiro



2.7

Jan
2020



3.x+

▪
▪
▪

"Python é um experimento de quanta liberdade os programadores precisam. Muita liberdade e ninguém consegue ler o código alheio; pouca liberdade e a livre expressão fica comprometida."

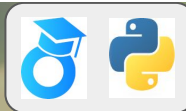
~ Guido van Rossum, (BDFL)



Python é uma linguagem muito utilizada?

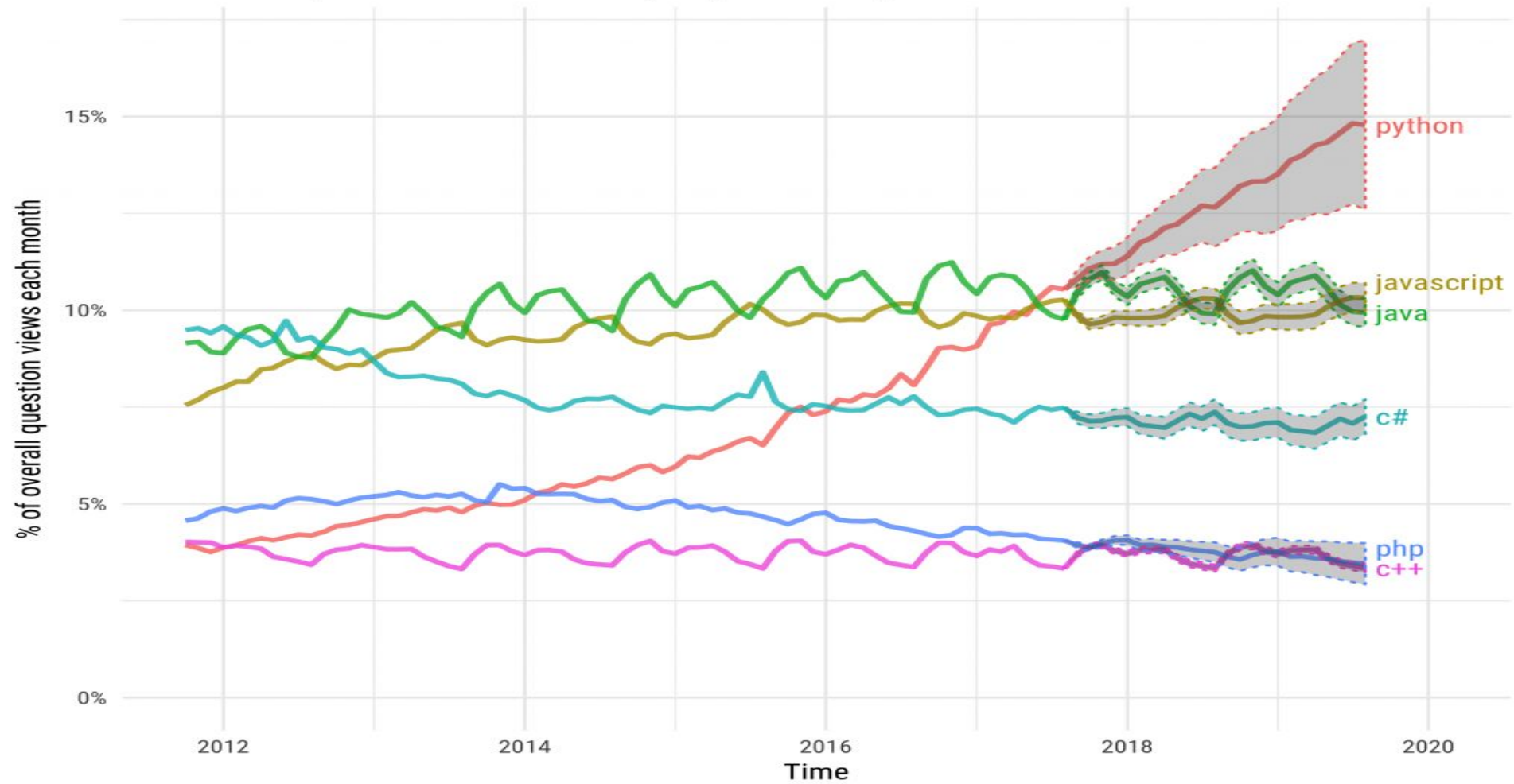
Aug 2019	Aug 2018	Change	Programming Language	Ratings	Change
1	1		Java	16.028%	-0.85%
2	2		C	15.154%	+0.19%
3	4	⬆	Python	10.020%	+3.03%
4	3	⬇	C++	6.057%	-1.41%
5	6	⬆	C#	3.842%	+0.30%
6	5	⬇	Visual Basic .NET	3.695%	-1.07%
7	8	⬆	JavaScript	2.258%	-0.15%
8	7	⬇	PHP	2.075%	-0.85%
9	14	⬆	Objective-C	1.690%	+0.33%
10	9	⬇	SQL	1.625%	-0.69%

Fonte:

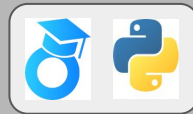
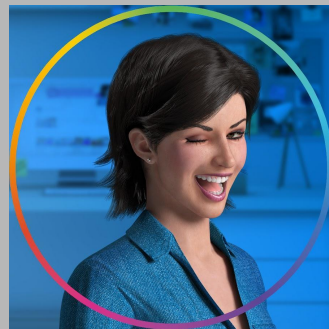
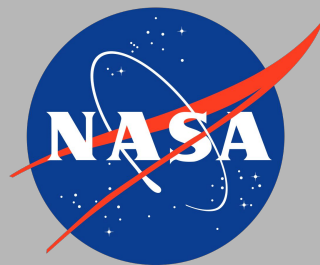
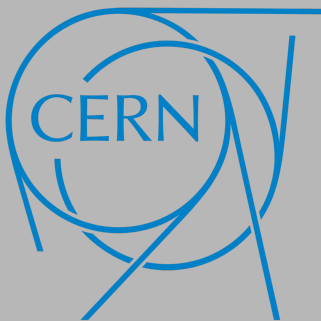


Projections of future traffic for major programming languages

Future traffic is predicted with an STL model, along with an 80% prediction interval.



Quem usa Python?

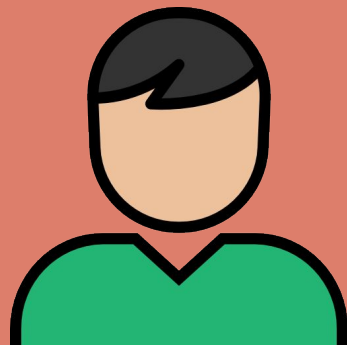


Para quais finalidades posso usar Python?

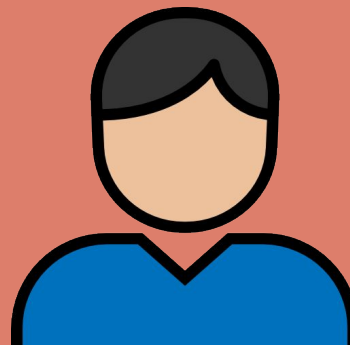
- Linguagem de script;
- Desenvolvimento web;
- Computação científica;
- Computação matemática;
- Ciência de Dados;
- Manipulação de imagens 2D e 3D;
- A.I.;
- Processamento de texto / manipulação de strings;
- Machine learning;
- Segurança da informação (ethical hacking)...



Olá, tudo bem?



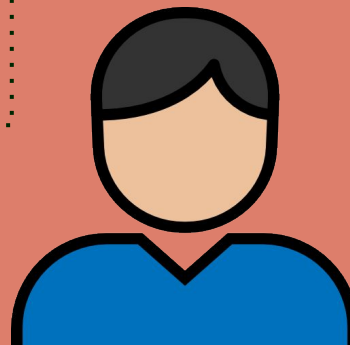
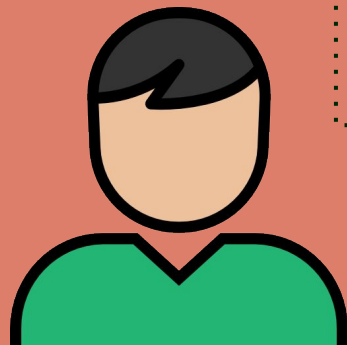
Γεια σας, πώς είσαι;

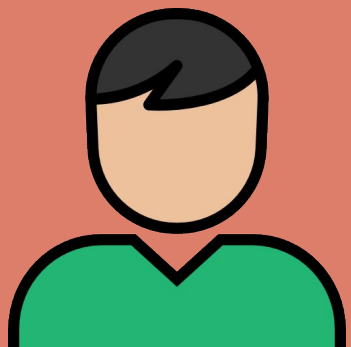


Οιά, tudo bem?

Γεια σας, πώς είσαι;

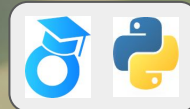
Καλά και εσύ





```
fabricio@localhost:~  
fabricio@localhost:~ 85x26  
Python 3.7.4 (default, Jul  9 2019, 16:32:37)  
[GCC 9.1.1 20190503 (Red Hat 9.1.1-1)] on linux  
Type "help", "copyright", "credits" or "license" for more information.  
>>> import this  
The Zen of Python, by Tim Peters  
  
Beautiful is better than ugly.  
Explicit is better than implicit.  
Simple is better than complex.  
Complex is better than complicated.  
Flat is better than nested.  
Sparse is better than dense.  
Readability counts.  
Special cases aren't special enough to break the rules.  
Although practicality beats purity.  
Errors should never pass silently.  
Unless explicitly silenced.  
In the face of ambiguity, refuse the temptation to guess.  
There should be one-- and preferably only one --obvious way to do it.  
Although that way may not be obvious at first unless you're Dutch.  
Now is better than never.  
Although never is often better than *right* now.  
If the implementation is hard to explain, it's a bad idea.  
If the implementation is easy to explain, it may be a good idea.  
Namespaces are one honking great idea -- let's do more of those!  
>>> █
```

Interpretador Python

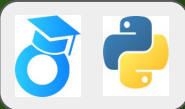




O Anaconda é uma distribuição gratuita e de código aberto das linguagens de programação Python e R para computação científica, que visa simplificar o gerenciamento e a implantação de pacotes. As versões do pacote são gerenciadas pelo sistema de gerenciamento de pacotes Conda.



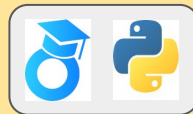
Bora criar nosso
primeiro ambiente, e
começar a codar em 🐍 ?



```
$ conda create --name labsschool python=3.7  
$ conda activate labsschool
```

Documento sobre como gerenciar environments (ambientes) com o Conda:

<https://docs.conda.io/projects/conda/en/latest/user-guide/tasks/manage-environments.html>





<https://github.com/flribeiro/lis-intro-python>

