

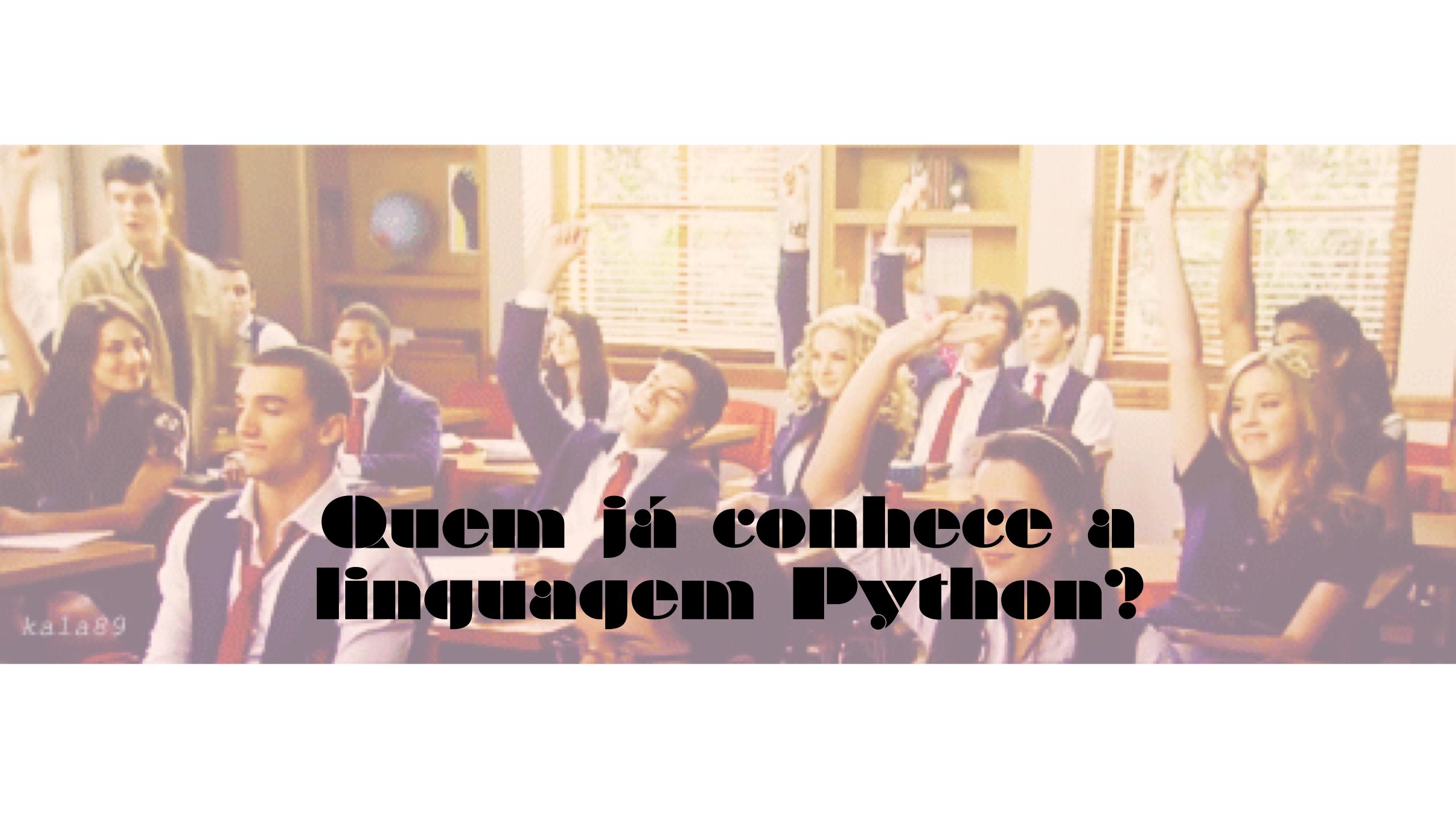
Python e suas baterias inclusas

Givanaldo Rocha de Souza

givanaldo.rocha@ifrn.edu.br

Professor de Sistemas de Informação
IFRN – Campus Parnamirim



A classroom scene with many students in school uniforms (dark blazers, white shirts, red ties) sitting at desks. They are all looking upwards and to the right, with their hands raised, as if participating in a class discussion or activity.

**Quem já conhece a
linguagem Python?**



O QUE É PYTHON?



O que é Python?

BY LUISGUS

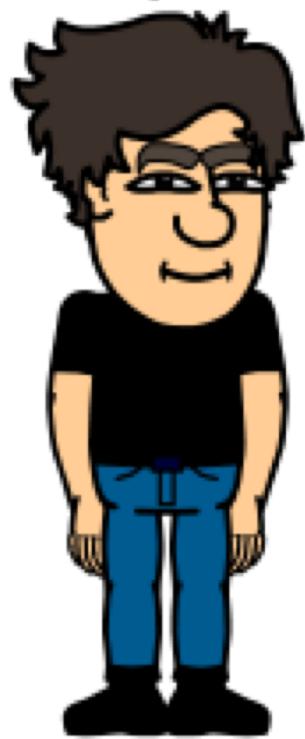
Oi Crax.
Bem vindo à
Pyctorial!

Oi. Ah, você não
devia começar
com Hello.

Não? Mas todos
fazem isso. E em Python
é verdade. Nada de
main,begin, etc..

Ok. Mas primeiro
devemos explicar o
que é python.

Python é uma linguagem de programação
dinâmica, fortemente tipada e orientada
a objetos. Contém recursos avançados como
compreensão de listas e geradores. Possui
gerenciamento automático de memória
e, sempre bom lembrar, é software livre.





Guido van Rossum
<https://gvanrossum.github.io/>

Criou a linguagem Python em 1991.

Trabalho no Google de 2005 a 2012.

Atualmente trabalha no Dropbox.

Há mais de seis anos, em dezembro de 1989, eu estava procurando por um projeto de programação como "hobby" que me mantivesse ocupado durante a semana próxima ao Natal. Meu escritório... estaria fechado, mas eu tinha um computador em casa e não muito mais do que isso em mãos. Eu decidi escrever um interpretador para a nova linguagem de scripting sobre a qual eu vinha pensando ultimamente: uma descendente da ABC que agradaria a hackers de Unix/C.



vamos falar de Python agora...





Versões atuais: **2.7.15** e **3.7.0**

<http://www.python.org/download>

** Python puro (CPython)

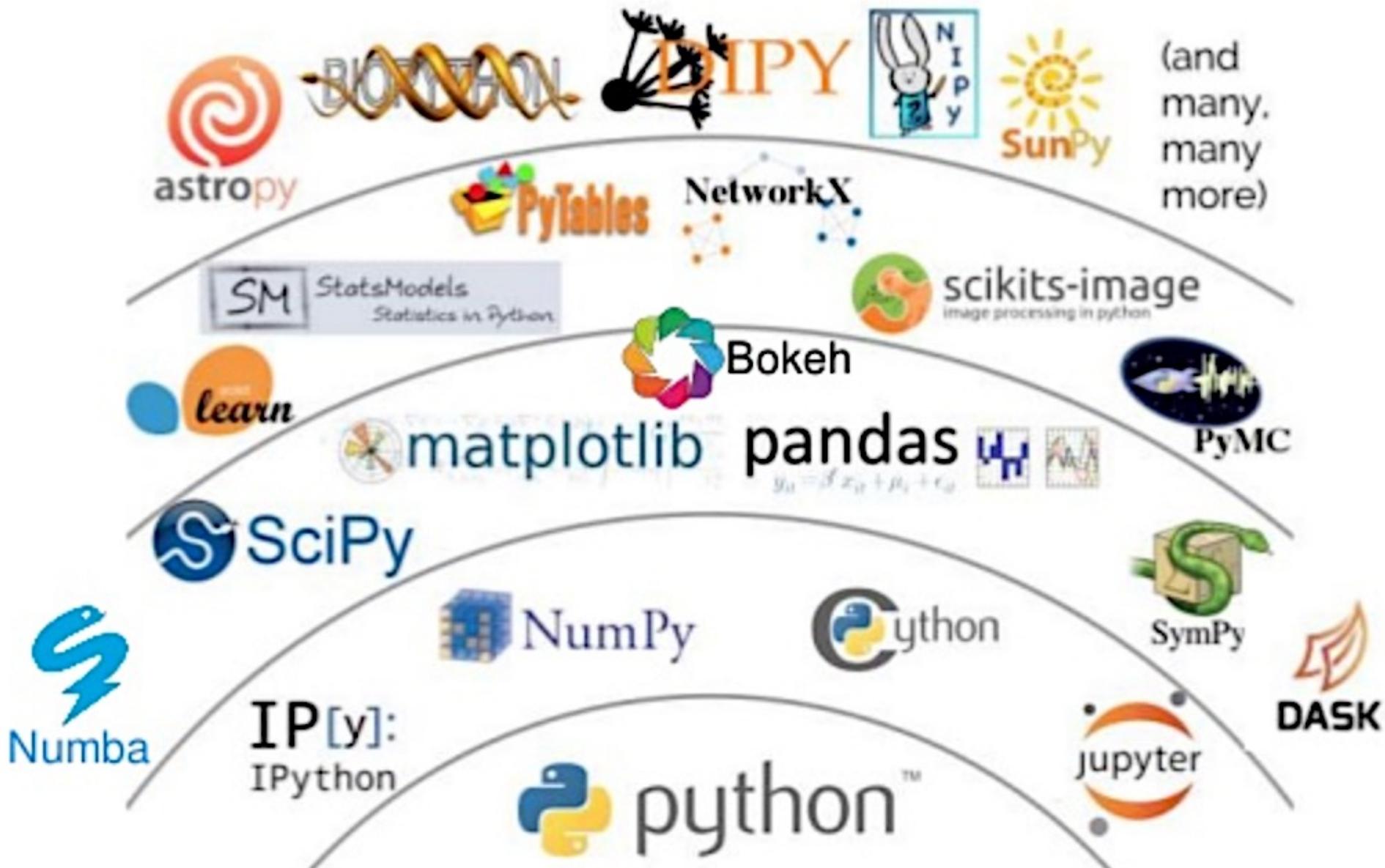
Windows: necessário instalar.

Linux e Mac: já instalado, verificar se é a versão mais recente.

Instalação alternativa



<http://anaconda.org>



Popularidade (TIOBE Index)

Aug 2018	Aug 2017	Change	Programming Language
1	1		Java
2	2		C
3	3		C++
4	5	▲	Python
5	6	▲	Visual Basic .NET
6	4	▼	C#
7	7		PHP
8	8		JavaScript
9	-	▲	SQL
10	14	▲	Assembly language

IEEE Ranking

Language Rank	Types	Spectrum Ranking
1. Python	  	100.0
2. C++	  	99.7
3. Java	  	97.5
4. C	  	96.7
5. C#	  	89.4
6. PHP		84.9
7. R		82.9
8. JavaScript	 	82.6
9. Go	 	76.4
10. Assembly		74.1

Python 2 x Python 3

Suporte
acaba em
2020.

ASCII

Unicode

```
# declaração print  
print "Hello World!"
```

```
# função print  
print("Hello World!")
```

```
raw_input()
```

```
input()
```

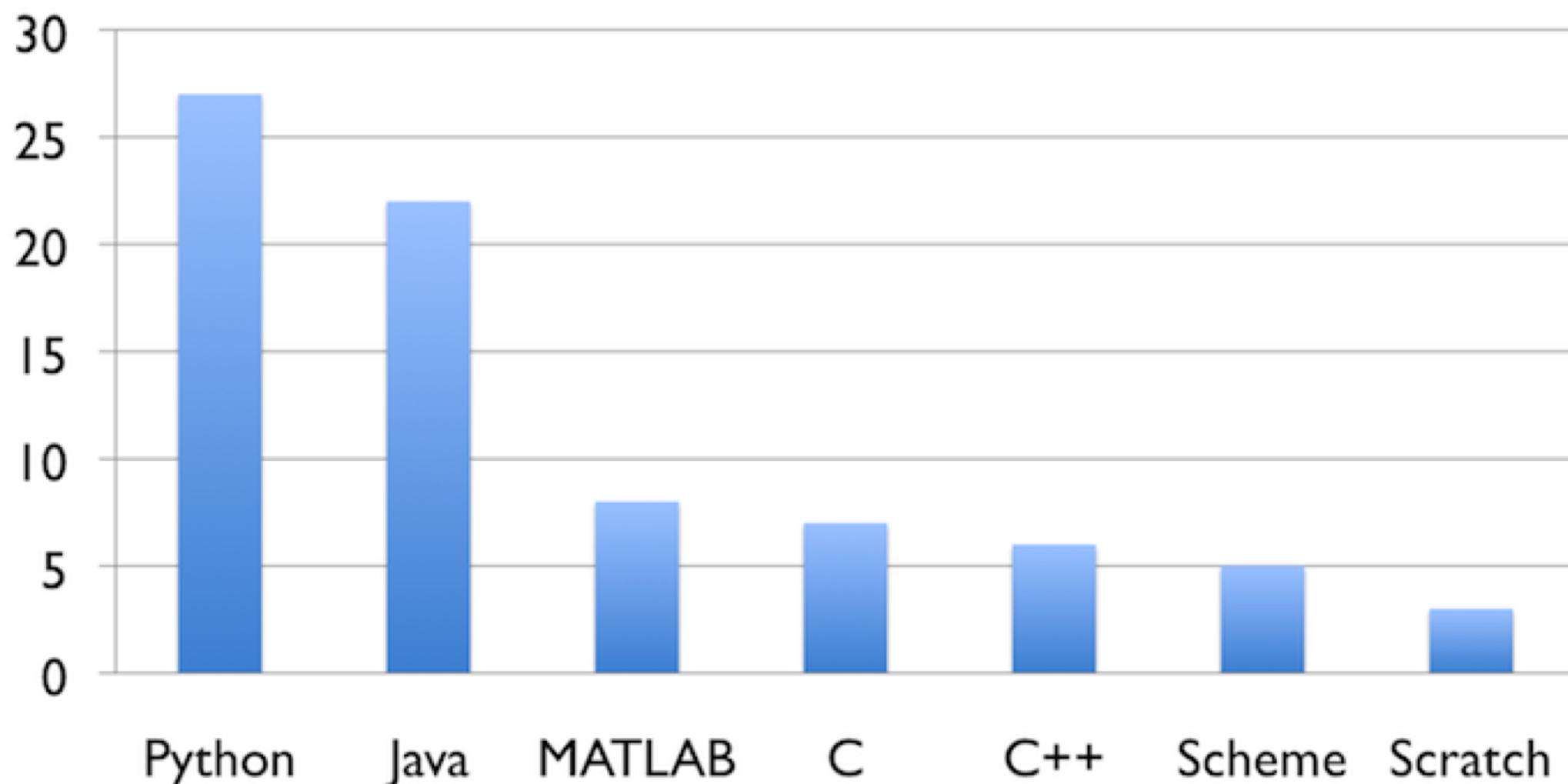
```
# Divisão de inteiro  
>>> 5 / 2  
2
```

```
# Divisão de inteiro  
>>> 5 / 2  
2.5
```

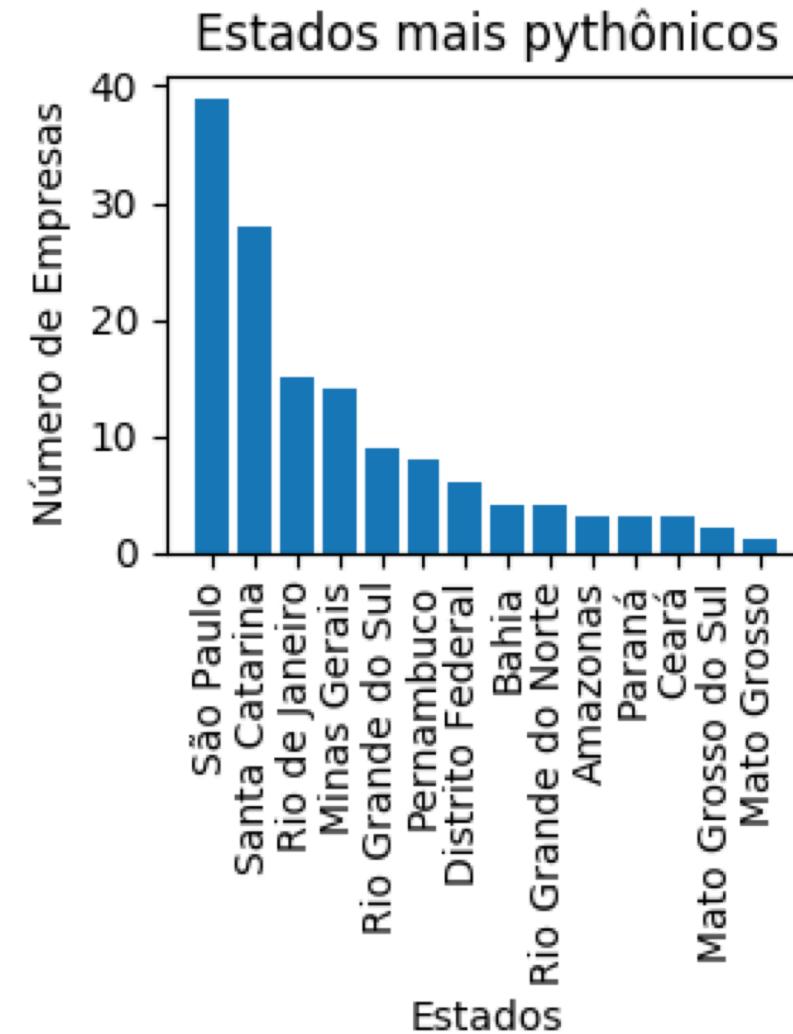
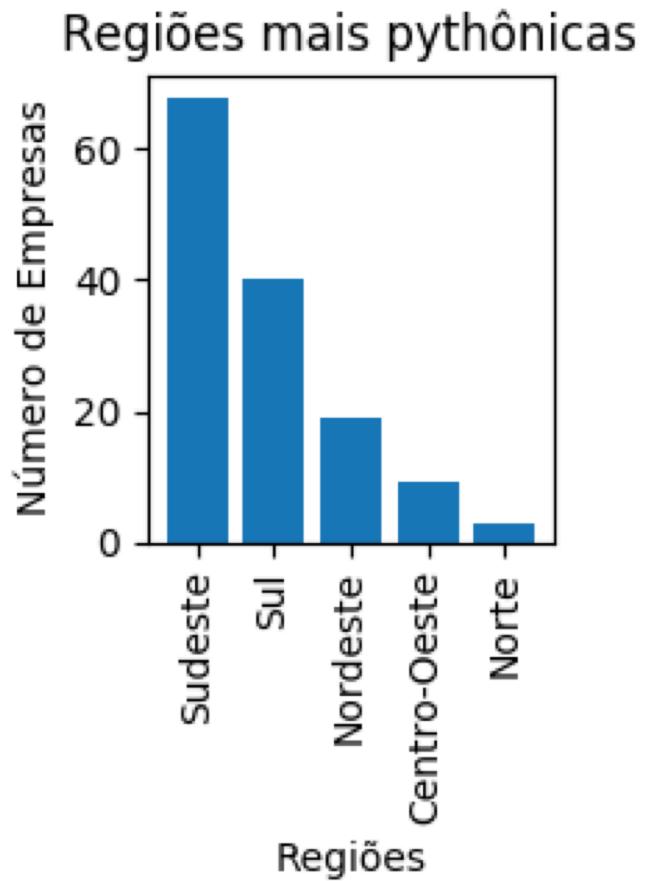
Quem usa?



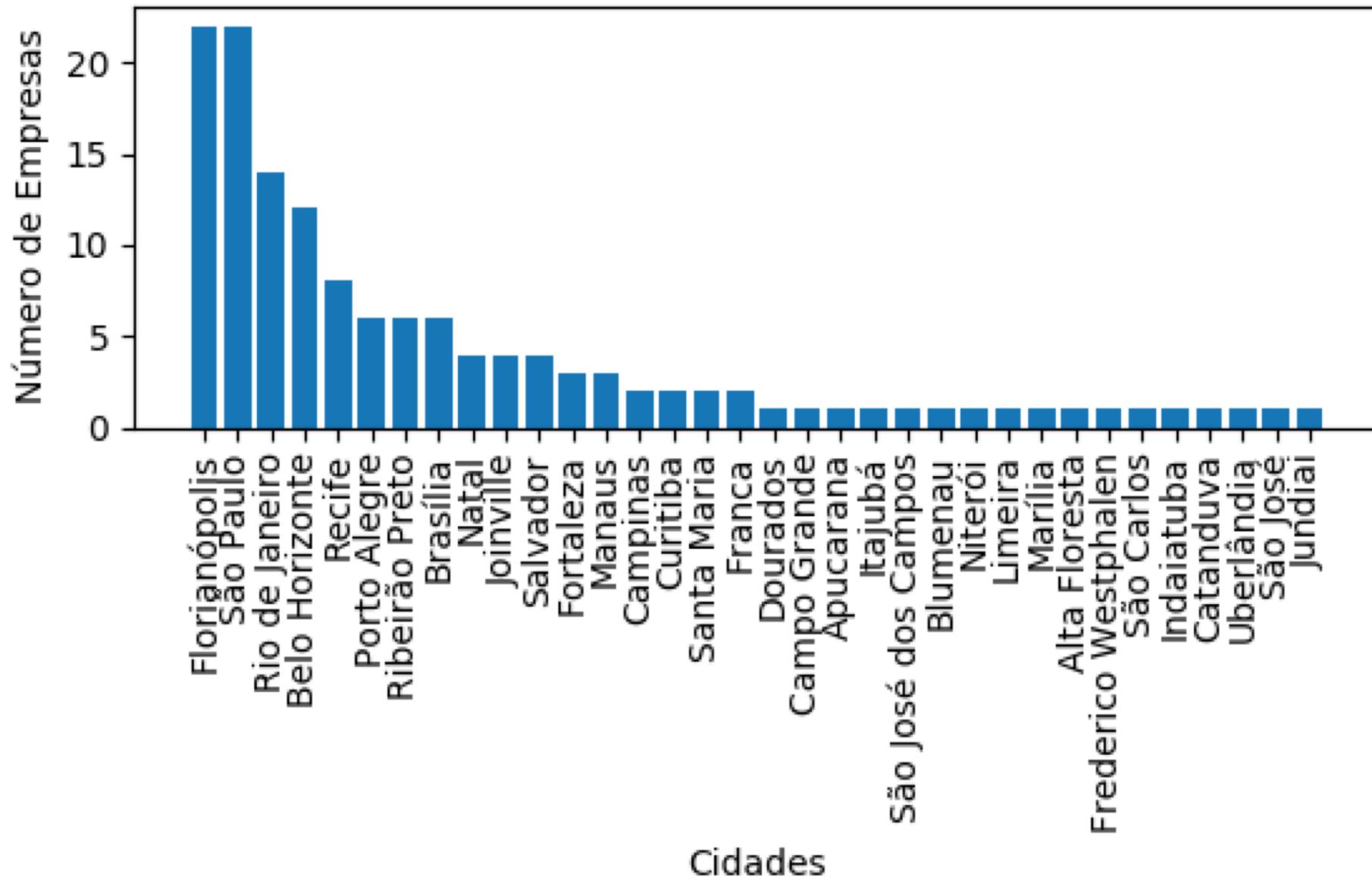
Number of top 39 U.S. computer science departments
that use each language to teach introductory courses



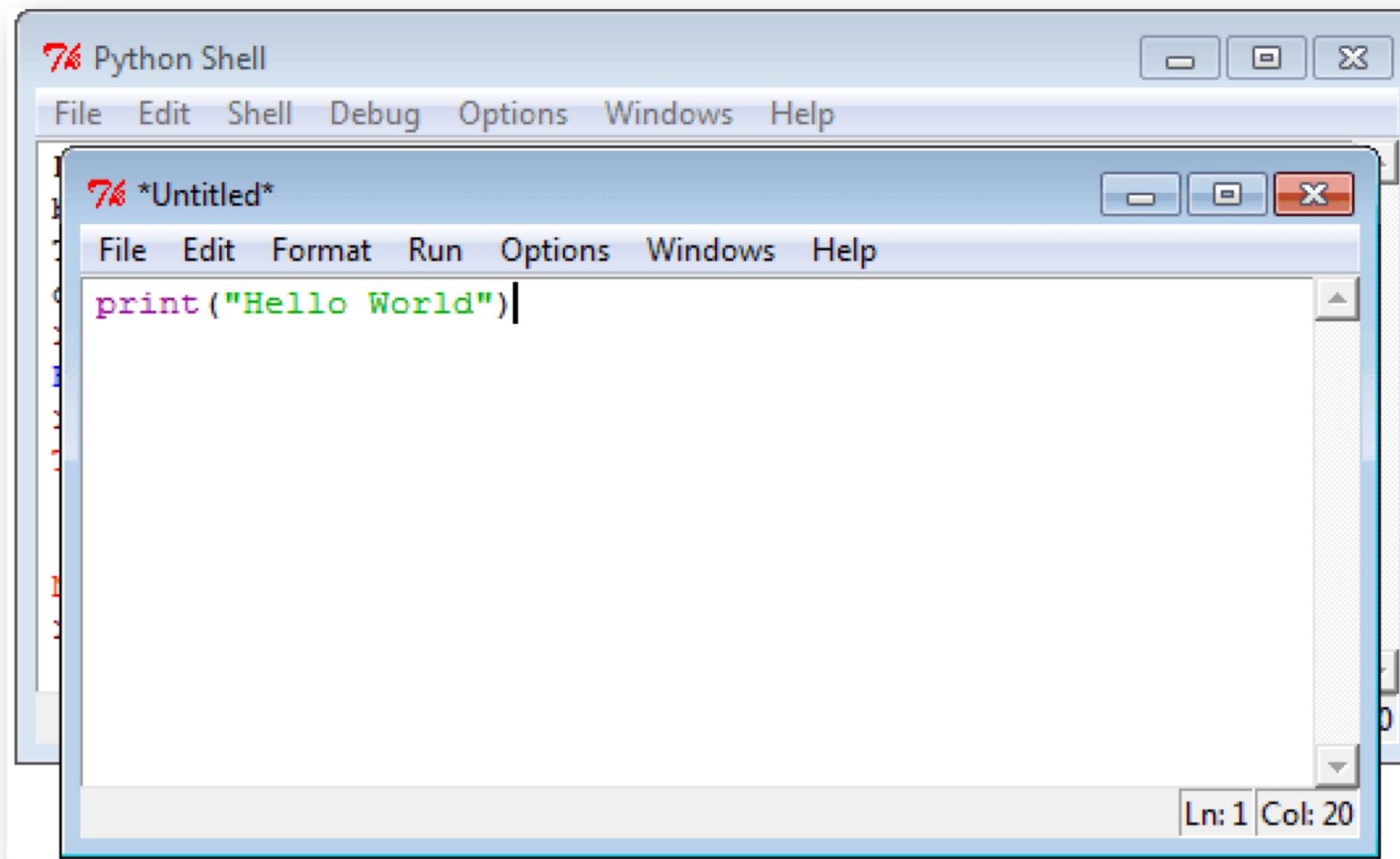
Analysis done by Philip Guo (www.pgbvine.net) in July 2014, last updated 2014-07-29



Cidades mais pythônicas



Ambientes de desenvolvimento (IDLE)



Ambientes de desenvolvimento (vs code)

The screenshot shows the Visual Studio Code interface. The title bar reads "retrieve.py - twitter-python-json - Visual Studio Code". The menu bar includes File, Edit, View, Go, and Help. On the left, there are icons for file operations, a search bar, a terminal with a '7' notification, and a problems icon. A dropdown menu titled "DEB..." is open, showing options: Python, PySpark, Python Module, Integrated Terminal/Console, External Terminal/Console, Django, Flask, Flask (old), Watson, and Attach (Remote Debug). The "Python" option is highlighted. The main editor area displays the following Python script:

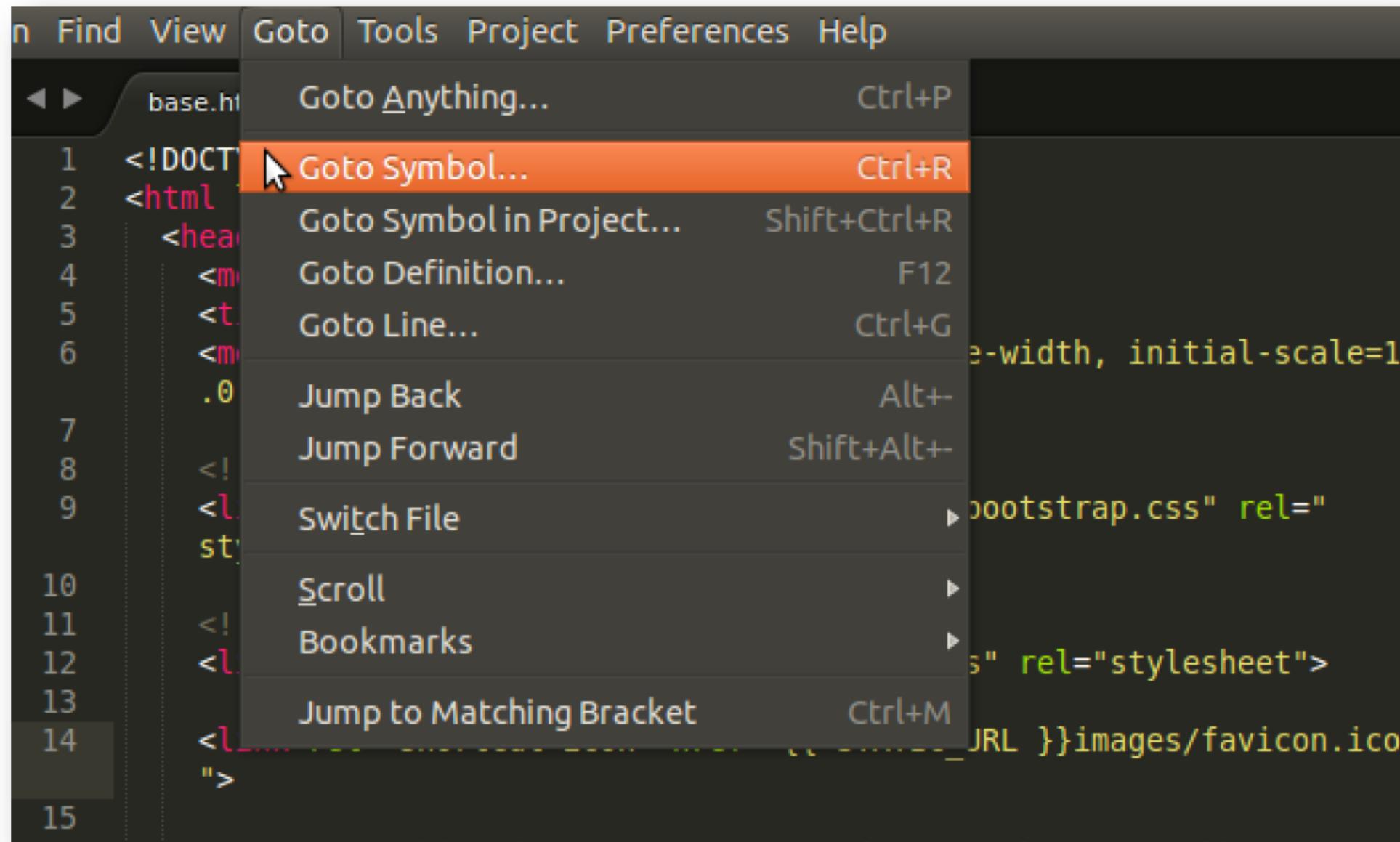
```
#!/usr/bin/env python
"""This script connects and extract from Twitter more messages (tweets) as json file
"""

import json
import configparser
from TwitterAPI import TwitterAPI
from TwitterAPI import TwitterRestPager

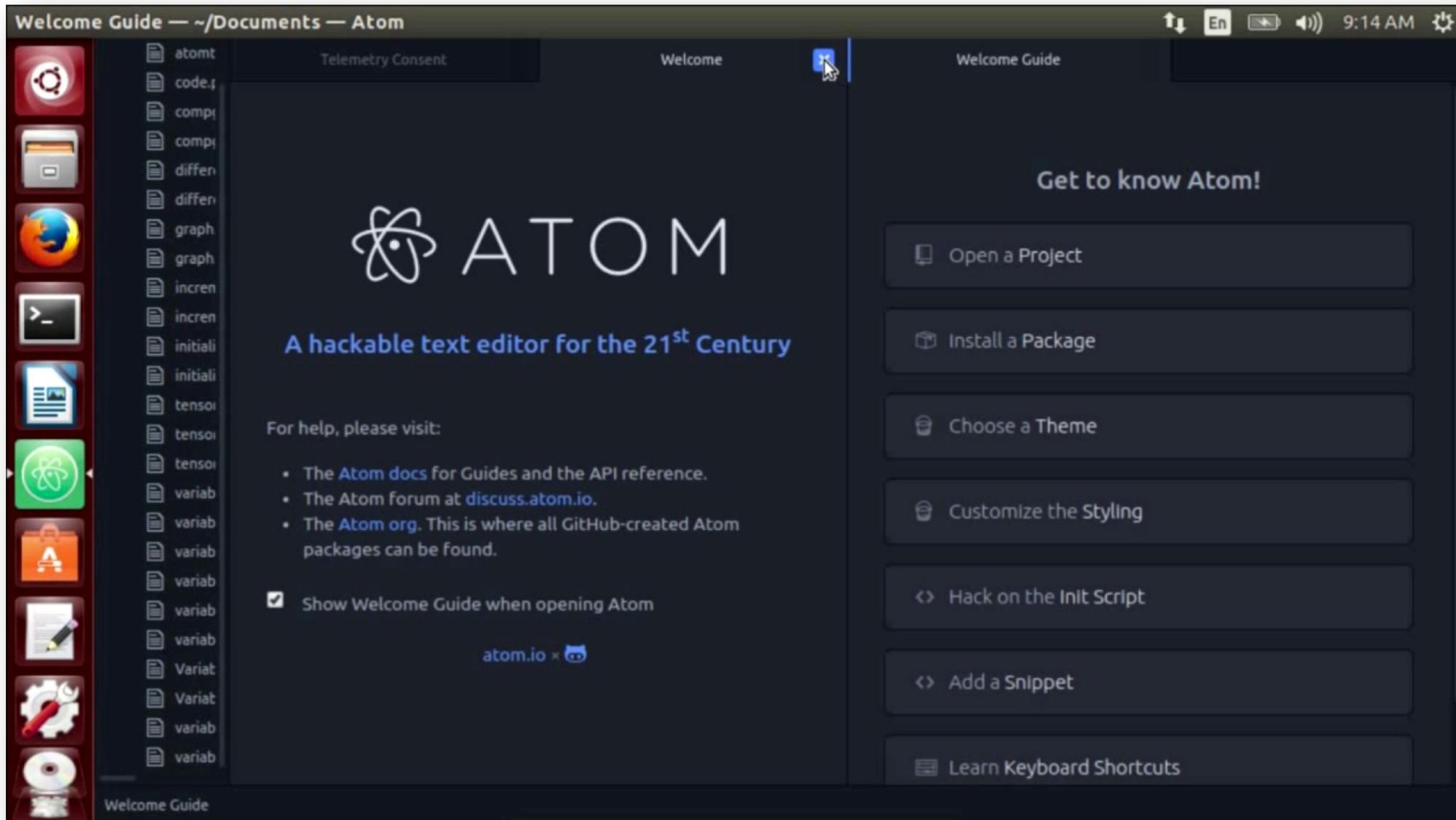
configValues = configparser.RawConfigParser()
configValues.read(r'.\config.ini')

TWITTER_API = TwitterAPI(configValues.get('TwitterSettings', 'consumer_key'),
                         configValues.get('TwitterSettings', 'consumer_secret'),
                         configValues.get('TwitterSettings', 'access_token_key'),
                         configValues.get('TwitterSettings', 'access_token_secret'))
```

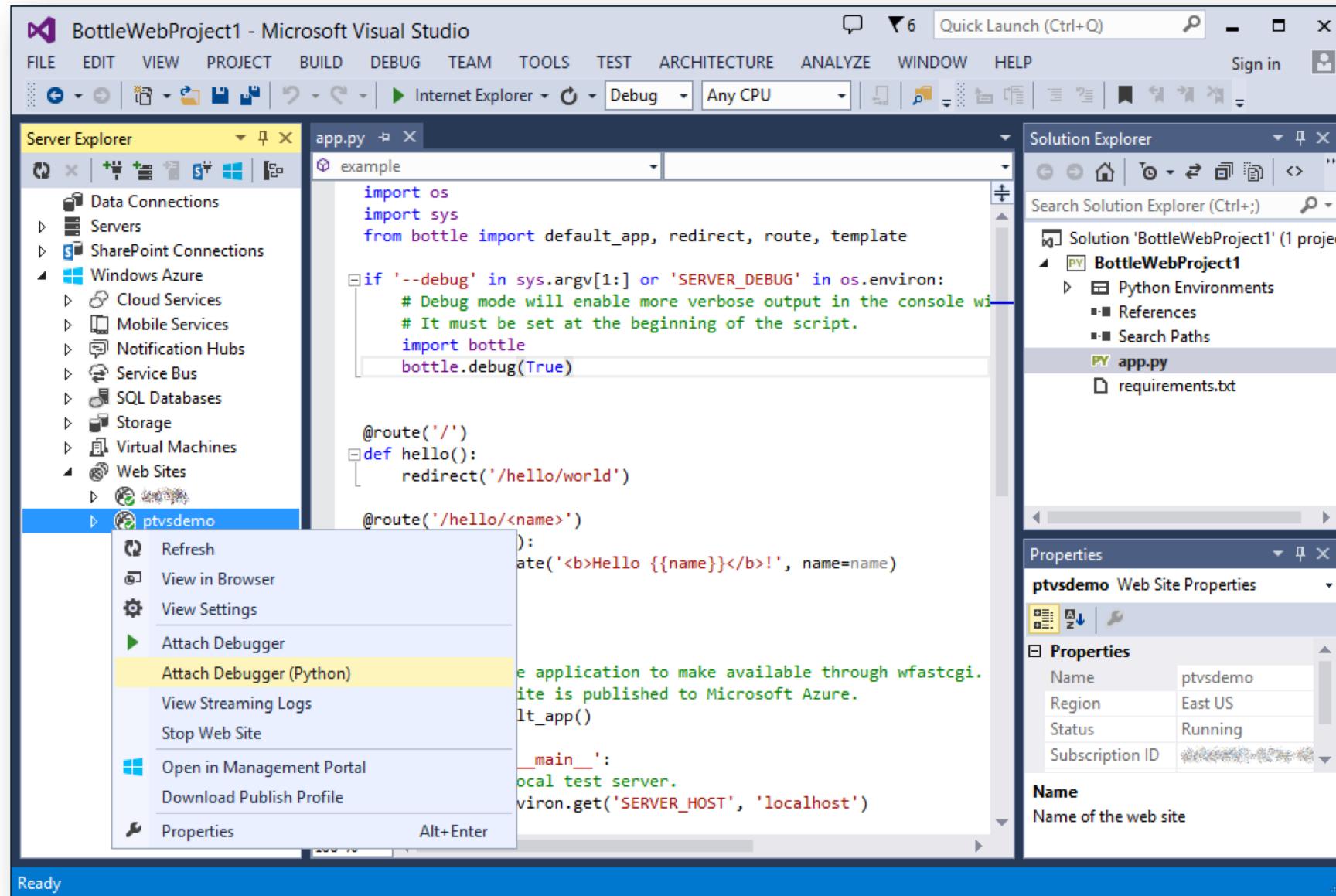
Ambientes de desenvolvimento (Sublime Text)



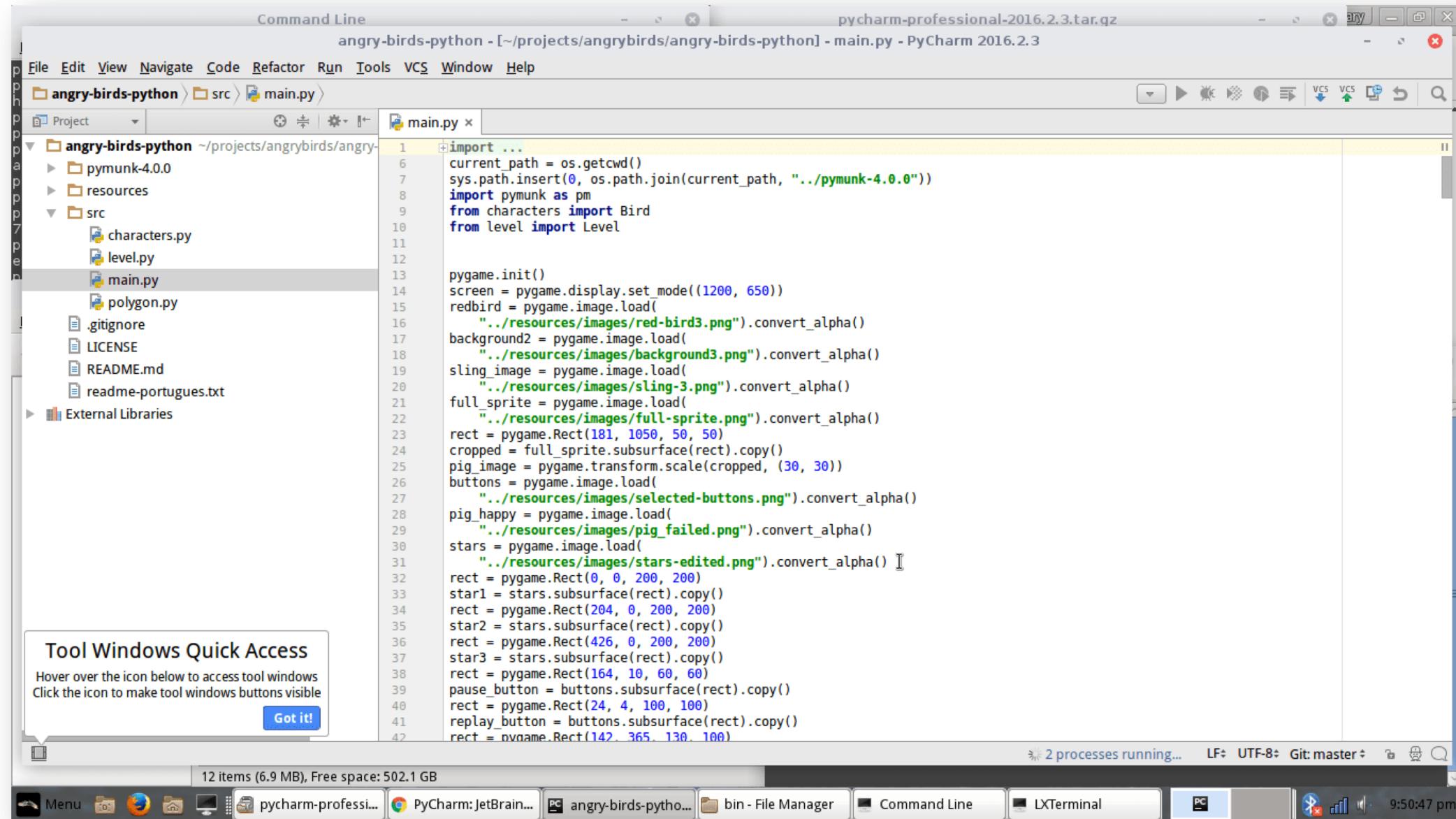
Ambientes de desenvolvimento (Atom)



Ambientes de desenvolvimento (Visual Studio)



Ambientes de desenvolvimento (PyCharm)



Ambientes de desenvolvimento (Online)

The screenshot shows a repl.it interface. At the top, there's a navigation bar with a user icon, the handle '@anonymous/TemptingHarmfulAnglerfish', a 'No description' button, and links for 'my repls', 'community BETA', and 'givanaldo...'. Below the bar, there are buttons for 'share' and 'run'. A message box says 'Not sure what to do? Run some examples (dismiss)'. The main area has a dark background with a terminal window showing Python 3.6.1 running on Linux. The code input field contains:

```
Python 3.6.1 (default, Dec 2015, 13:05:11)
[GCC 4.8.2] on linux
> print("SELINFO")
SELINFO
> num1 = 2
> num2 = 9
> print("SOMA = %d" % (num1+num2))
SOMA = 11
```

Below the code, there are buttons for 'Get live help! (NEW!)', 'Start private chat session', and 'How do I use this?'. The code editor shows:

```
Python 3.6
1 print("SELINFO")
2 num1 = 2
3 num2 = 9
4 print("SOMA = %d" % (num1+num2))
```

Execution status: 'Edit code | Live programming'. It indicates the last line was executed and the next line is to be executed. A note says 'Click a line of code to set a breakpoint; use the Back and Forward buttons to jump there.' At the bottom, there are buttons for '<< First', '< Back', 'Program terminated', 'Forward >', and 'Last >'. A footer note says 'Visualized using Python Tutor by Philip Guo (@pgbovine)'. A feedback section asks 'Help us improve this tool by clicking below whenever you learn something:' with buttons for 'I just cleared up a misunderstanding!' and 'I just fixed a bug in my code!'.

On the right side, there's a sidebar titled 'Print output (drag lower right corner to resize)' showing the output from the print statements. It also has tabs for 'Frames' and 'Objects', and a 'Global frame' section showing variable assignments.

pythontutor.com

Ambientes de desenvolvimento (Jupyter)

jupyter spectrogram (autosaved) Python 3

File Edit View Insert Cell Kernel Help

CellToolbar

Simple spectral analysis

An illustration of the [Discrete Fourier Transform](#)

$$X_k = \sum_{n=0}^{N-1} x_n \exp^{-j \frac{2\pi}{N} kn} \quad k = 0, \dots, N-1$$

In [2]: `from scipy.io import wavfile
rate, x = wavfile.read('test_mono.wav')`

And we can easily view it's spectral structure using matplotlib's builtin specgram routine:

In [5]: `fig, (ax1, ax2) = plt.subplots(1,2,figsize(16,5))
ax1.plot(x); ax1.set_title('Raw audio signal')
ax2.specgram(x); ax2.set_title('Spectrogram');`

The figure contains two subplots. The left subplot, titled 'Raw audio signal', displays a blue line graph of a raw audio signal over time, with the y-axis ranging from -30000 to 40000. The right subplot, titled 'Spectrogram', displays a heatmap representing the power spectrum of the signal over time, with the y-axis ranging from 0.0 to 1.0.

**Talk is cheap.
Show me the code.**

Linus Torvalds