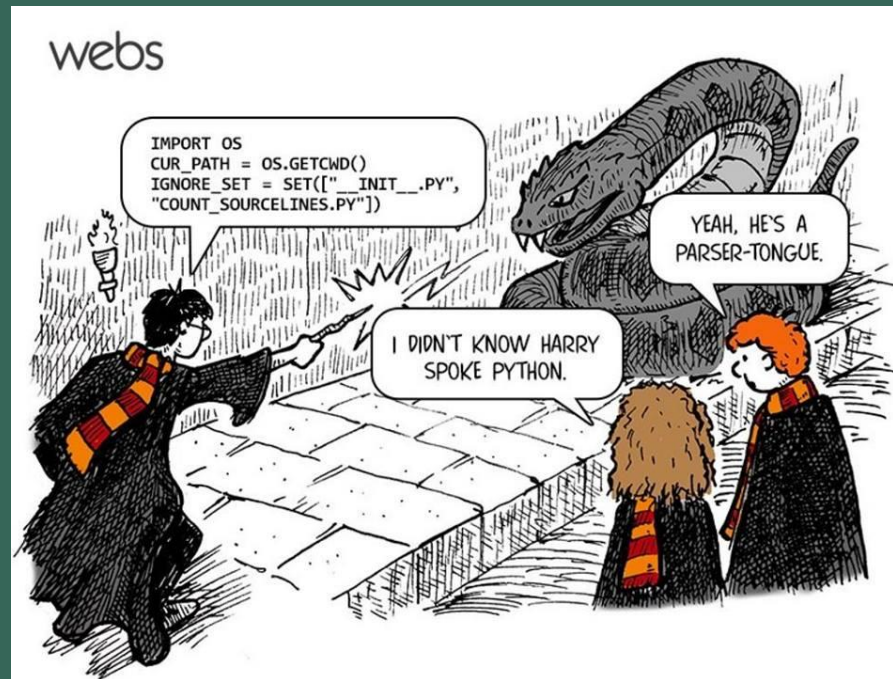


Introducción a Python



ORGANIZACIÓN

SESIÓN I

- Introducción a Python
- Tipos de Datos
- Estructuras de control
- Funciones
- Ficheros

SESIÓN II

- Módulo, paquetes y gestores de librerías
- Expresiones regulares
- Entornos virtuales
- Documentación

```
#!/usr/bin/python

import sys

for i in range(1,11):
    print i
    if i <= 5:
        print "AHHHHHHH"
    elif i >= 6 and i <= 9:
        print "WHERE ARE ALL THE BRACKETS??"
    else:
        print "HOW DO YOU PEOPLE READ THIS SYNTAX EASILY"
```

SESIÓN I: PRINCIPIOS BÁSICOS








EL ORIGEN DE PYTHON

- Python fue creado por Guido van Rossum a finales de los años ochenta.
- Se creó como un lenguaje sucesor del lenguaje de programación ABC, capaz de manejar excepciones e interactuar con el sistema operativo Amoeba.
- Recibe su nombre del popular grupo cómico británico Monty Python
- Según la leyenda Guido creó Python durante unas vacaciones de navidad en las que (al parecer) estaba algo aburrido.
- Sitio web oficial: <https://www.python.org/>

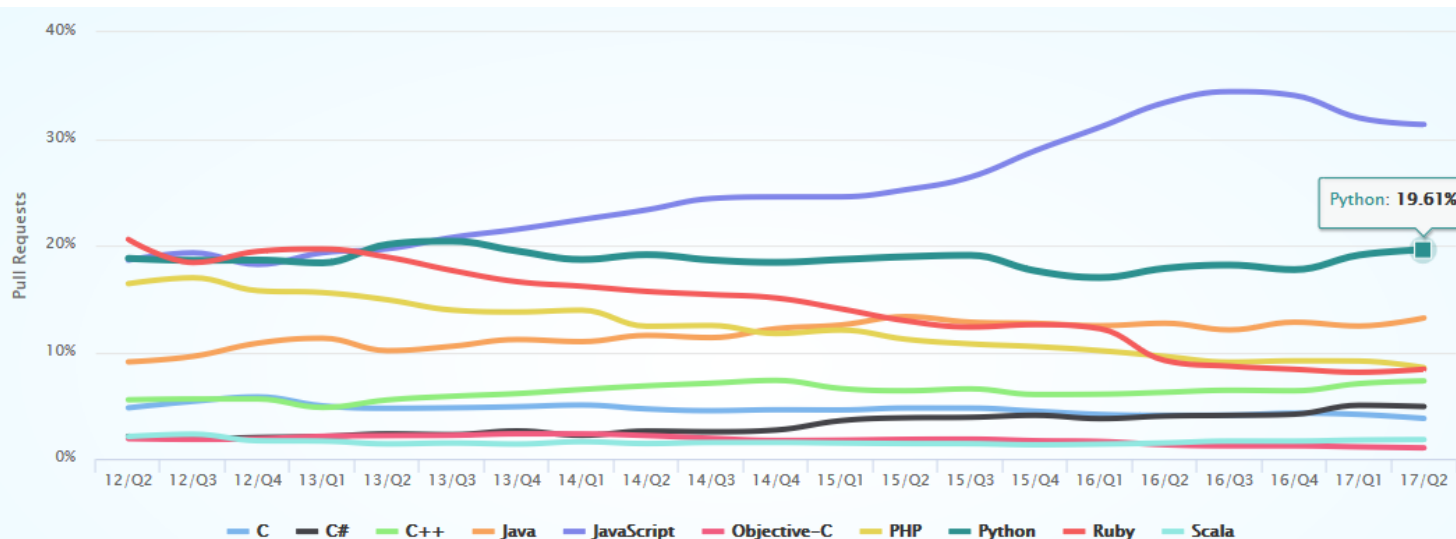


De Doc Searls -
2006oscon_203.JPG, CC BY-SA
2.0,
<https://commons.wikimedia.org/w/index.php?curid=4974869>

Language Rank	Types	Spectrum Ranking
1. Python	 	100.0
2. C	  	99.7
3. Java	  	99.5
4. C++	  	97.1
5. C#	  	87.7
6. R		87.7
7. JavaScript	 	85.6
8. PHP		81.2
9. Go	 	75.1
10. Swift	 	73.7

¿CÓMO DE POPULAR ES PYTHON?

Análisis de los lenguajes de programación dentro de la comunidad IEEE
Fuente: IEEE Spectrum. (<https://spectrum.ieee.org/computing/software/the-2017-top-programming-languages>)



Repositorios en Github
por lenguaje de
programación
Fuente:
<https://madnight.github.io/github/>

EJEMPLOS DE ÉXITO DEL USO DE PYTHON



<http://www.bittorrent.com/>



Gestión de actualizaciones de
sus dispositivos



Gestión del tráfico aéreo
www.frequentis.com



Minería de datos
<http://www.forecastwatch.com/>



Industrial Light & Magic
<http://www.ilm.com/>



[Python Streamlines Space Shuttle Mission Design](#)

Y muchos más ejemplos se pueden ver en <https://www.python.org/about/success/>

CARACTERÍSTICAS DE PYTHON

- Lenguaje de scripting de propósito general.
- Open Source y código interoperable (“write once run everywhere”)
- Muy legible y elegante → evitar código ofuscado.
- Simple y poderoso:
 - Minimalista: todo aquello innecesario no hay que escribirlo (;, {, }...)
 - Muy denso: poco código hace mucho.



Java	Python
<pre>public class HolaMundo { public static void main(String[] args) { System.out.println("Hola Mundo"); } }</pre>	<pre>#!/usr/bin/env python print "Hola Mundo"</pre>

CARACTERÍSTICAS DE PYTHON (II)

- Soporta objetos y estructuras de datos de alto nivel.
- Múltiples niveles de organizar código:
 - funciones,
 - clases,
 - módulos,
 - y paquetes
- El tipado de variables es dinámico.
- Python standard library (<http://docs.python.org/>) contiene un sinnúmero de clases de utilidad
- Si hay áreas en las que Python no puede desempeñar su función de forma eficiente se puede reemplazar este código por plugins en C o C++, siguiendo la API para extender o empotrar Python en una aplicación, o a través de herramientas como SWIG, sip o Pyrex.

VENTAJAS Y DESVENTAJAS

👍 Python es bueno para...

“Pegamento” entre componentes de diversa naturaleza.

Prototipos rápidos de sistemas informáticos.

Elaboración de aplicaciones cliente.

Desarrollo web y sistemas distribuidos.

Uso en tareas científicas.

👎 No es tan bueno para...

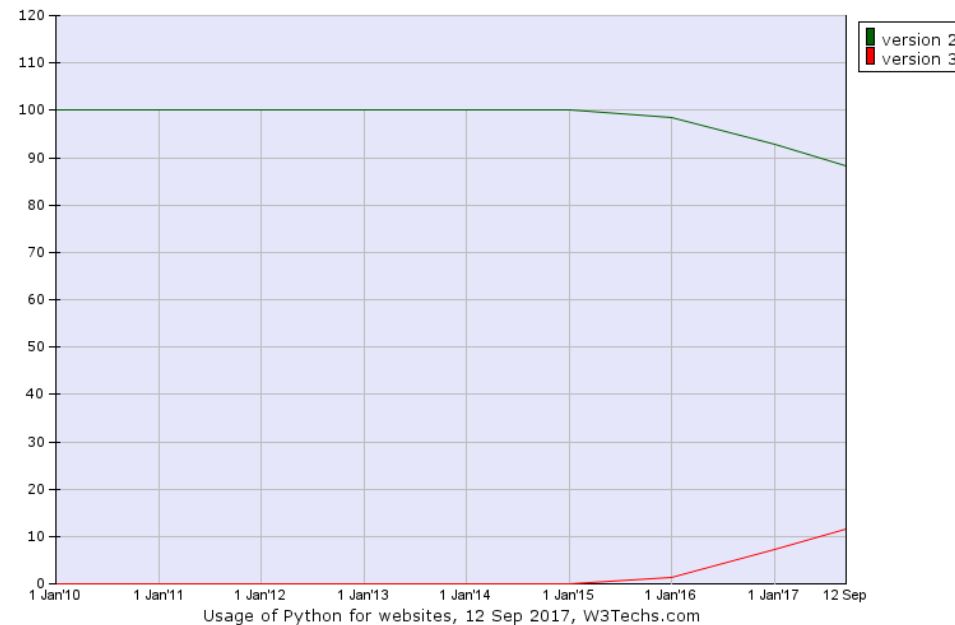
Programación a muy bajo nivel como por ejemplo drivers o firmware, que deban gestionar hardware directamente.
Se trata de un lenguaje de muy alto nivel.

Aplicaciones de muy alto nivel de computo, al ser un lenguaje interpretado el rendimiento siempre es mejor con C.

VERSIONES DE PYTHON

- Rama 2.x
 - En versiones más actualizadas de distribuciones Linux, como Ubuntu o CentOS, está disponible Python 2.7
- Rama 3.x (Python 3000, Py3k, Python 3.5,...)
 - Limpia y reestructura Python
 - Incompatible con la rama 2.x
 - Problema: muchos paquetes sólo existen para la versión 2.7
- El plan siempre fue convivir Python 2.7 con Python 3.x hasta que Python 2.x esté en desuso (cada vez más cerca).

The diagram shows only Python versions with more than 1% usage.



Fuente: https://w3techs.com/technologies/history_details/pl-python/all/y

¿QUÉ NECESITO PARA EJECUTAR PYTHON?

- Puedo bajarme el núcleo del lenguaje directamente desde www.python.org
 - En Windows hay que instalar el ejecutable.
 - En Linux suele estar instalado por defecto.
 - Podemos usar el interprete directamente.
- Crear programas → utilizar cualquier editor de texto básico con una extensión para Python o editores de código compatibles:
 - Notepad++: <https://notepad-plus-plus.org/download/v7.5.1.html>
 - SublimeText: <https://www.sublimetext.com/>
 - PyDev (Eclipse para Python): <http://www.pydev.org/>
 - PyCharm: <https://www.jetbrains.com/pycharm/?fromMenu>
 - Spyder: <https://pypi.python.org/pypi/spyder>
 - ...





INTERPRETE DE PYTHON

- No se declaran ni las variables, ni las funciones ni los métodos en código.
- Los tipos de datos se determinan en tiempo de ejecución.
- Entonces es cuando surgen los errores.
- Por ejemplo la variable `a` es “int” y “string” a la vez.

```
Símbolo del sistema - python
Microsoft Windows [Versión 10.0.14393]
(c) 2016 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Llanos>python
Python 2.7.9 (default, Dec 10 2014, 12:24:55) [MSC v.1500 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> a=17
>>> a
17
>>> a+2
19
>>> a='hola'
>>> a
'hola'
>>> len(a)
4
>>> a+len(a)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: cannot concatenate 'str' and 'int' objects
>>>
```

EJECUTAR UN PROGRAMA

Seleccionamos
el resaltado
para Python si
no lo hemos
hecho ya

Escribimos el
código y lo
guardamos con
extensión .py

Ejecutamos con
la orden Python
seguido del
nombre del
fichero

Para facilitar el proceso existen diversas distribuciones de Python que nos ayudan con estos pasos

DISTRIBUCIONES DE PYTHON



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



<https://www.enthought.com/product/canopy/>



<https://www.activestate.com/activepython>



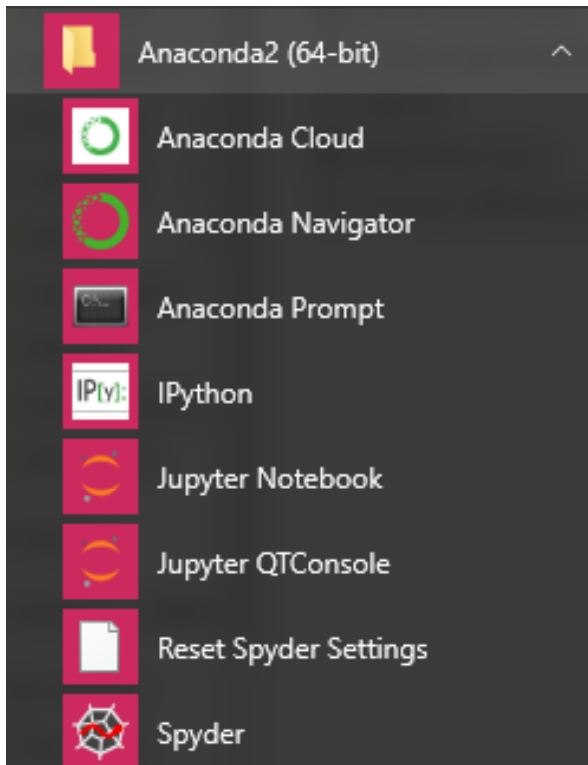
<https://python-xy.github.io/>



<http://winpython.github.io/>

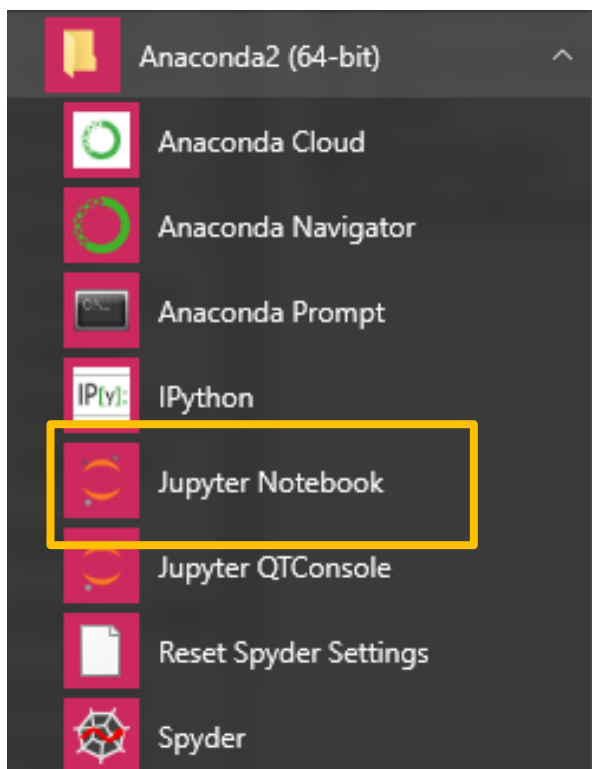
Incluyen: Python, un editor de desarrollo, soporte a los notebooks de Python, gestión de librerías, librería populares,.....

ANACONDA



- Anaconda Navigator: es una aplicación que permite gestionar la distribución, añadir nuevas herramientas y paquetes.
- Jupyter Notebook: es un editor online basado en “notebooks”. Es un entorno interactivo para probar scripts.
- Spyder: es un editor textual junto con herramientas asociadas como terminales o debugger para programas más complejos
- Anaconda Prompt: nos permite una consola con acceso directo a la distribución Python

JUPYTER NOTEBOOKS



- Cada celda puede contener:
 - Texto
 - Markdown
 - Latex
 - Código ejecutable
- Podemos exportarlo luego a PDF o a Python.

COMENTARIOS EN EL CÓDIGO

- Como siempre es imprescindible documentar nuestro código con comentarios significativos.
- Dentro de Python tenemos varias formas de añadir comentarios:
 - Si es un comentario sencillo de una línea o la final de la línea de código usaremos `#` seguido del comentario.
 - Usaremos tres comillas dobles para enmarcar los comentarios que vamos a usar como docstring. (Hablaemos de esto más adelante)

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Wed Sep 20 08:13:20 2017
4
5 @author: Llanos
6 """
7
8 #comentario sencillo
9 """
10 Comentario en varias líneas
11 Para poder documentar mejor el
12 código
13 """
```

La primera línea indica el encoding: en que codificación escribimos

RECURSOS PARA APRENDER PYTHON

- Tutorial Python por Google (inglés): <https://developers.google.com/edu/python/>
- CodeAcademy para Python: <https://www.codecademy.com/es/tracks/python-traduccion-al-espanol-america-latina-clone-1>
- Intro to Python for Data Scientists (DataCamp - Inglés):
<https://www.datacamp.com/courses/intro-to-python-for-data-science>
- Libros:
 - Python para todos: <http://mundogeek.net/tutorial-python/>
 - Oficial de Python argentina: <http://docs.python.org.ar/tutorial/pdfs/TutorialPython2.pdf>
 - Python para Informáticos: <http://do1.dr-chuck.net/py4inf/ES-es/book.pdf>
 - Curso de Python para principiantes: <http://www.iaa.es/python/curso-python-para-principiantes.pdf>