

Python / Guía 1

Introducción

a Python

## 

## 

## 

# 

# 

# ¿Por qué Python?

Python es un lenguaje de programación interpretado de código abierto, multiparadigma, principalmente orientado a objetos de alto nivel. Su sintaxis hace énfasis en la legibilidad del código, lo que facilita su depuración y, en efecto, favorece la productividad. Ofrece la potencia y la flexibilidad de los lenguajes compilados, con una curva de aprendizaje suave.

## 1. Python es un lenguaje interpretado

Python fue creado por Guido Van Rossum en 1991 como lenguaje de programación interpretado de uso general.

### ¿Qué significa que un lenguaje de programación sea interpretado?

Los lenguajes de bajo nivel, como lenguaje máquina o ensamblador, pueden ejecutarse directamente en una computadora. Los de alto nivel, como Java, C, C++ o el propio Python, por su parte, tienen que ser reinterpretados (compilados) como lenguajes de bajo nivel antes de poder ser ejecutados. Esto hace que los tiempos de ejecución normalmente sean más lentos. Sin embargo, hoy día, esto no es un problema, ya que los avances en computación en la nube (cloud computing), permiten disponer de capacidades de cálculo a medida a costos muy asequibles. También influye cuán optimizado está el código.

**Con Python, programar es sencillo**

Programar en código máquina es costoso y difícil. Python ofrece una sintaxis mucho más sencilla y cercana a la lógica de los humanos. Un código más legible resulta más sencillo de generar, depurar, y mantener. Por ello, la curva de aprendizaje de los lenguajes interpretados es mucho más suave, y la efectividad de su mantenimiento es mayor.

## 

## 

## 2. Python es potente, flexible y versátil

Veamos el porqué de esta potencia y versatilidad. Argumentos, no nos faltan.

### Es un lenguaje de propósito general

Al ser un lenguaje de propósito general, y no creado específicamente para desarrollo web, Python permite crear todo tipo de programas y herramientas.

### Es compatible con otros lenguajes de programación

Su interoperabilidad con otros lenguajes de programación como C, Java, R, etc., es otro de los factores que han ayudado a su gran difusión en distintos ámbitos.

### Permite trabajar con distintos modelos de programación

En Python, todo es un objeto. No obstante, aunque sea un lenguaje principalmente orientado a objetos, combina propiedades de distintos modelos o paradigmas de programación (imperativa, funcional, procedural o reflexiva).

### Ofrece librerías y entornos especializados en temas muy diversos

Por otra parte, Python ofrece librerías y entornos de desarrollo muy potentes para Machine Learning, Ciencias, visualización de datos, etc. Por ejemplo:

* Matemáticos y científicos utilizan SciPy y NumPy en sus investigaciones
* Los lingüistas analizan textos con NLTK (the Natural Language Tool Kit)
* Los estadísticos usan Pandas para analizar datos
* Los equipos de IT configuran y gestionan recursos en la nube con OpenStack
* Los desarrolladores usan Django, Flask o FastAPI para crear aplicaciones web

### 

### Es el lenguaje de referencia en Data Science y Machine Learning

De hecho, se ha convertido en el lenguaje de referencia en Data Science, siendo el preferido por un 57% de los científicos de datos y desarrolladores. Si tenemos en cuenta la evolución de los dos últimos años de entornos Python para Deep Learning, incluyendo la creación de Tensorflow y otras librerías especializadas, no nos extrañará que haya dejado atrás otros lenguajes como R, Julia, Scala, Ruby, Octave, MATLAB y SAS.

### Es lenguaje de referencia en educación

El hecho de que sea un lenguaje tan sencillo como para que lo puedan usar desde principiantes a programadores profesionales también lo ha convertido en el lenguaje de programación por excelencia en entornos educativos. Y no solo por su sencillez, sino también por el hecho de que se pueda ejecutar en distintos sistemas operativos (Microsoft Windows, Mac OS X, Linux, o usando el intérprete correspondiente). También es accesible a través de servicios web como Python Anywhere.

Esto es especialmente importante para el sector educativo porque se puede usar desde los equipos del aula de informática de las escuelas, o incluso desde casa, sin necesidad de instalar software adicional.

Gracias a esto, Python ha estado en el centro de distintos proyectos educativos muy interesantes, como los que veremos a continuación.

En 2015, la BBC lanzó el proyecto MicroBit project. Se trata de un pequeño dispositivo programable, micro:bit, cuyo objetivo es inspirar a una nueva generación de creadores, makers y coders, dirigido a niños a partir de 11 años de edad.

Otros proyectos, como MicroPython project, permiten trabajar con otros pequeños dispositivos, como Raspberry Pi, y usarse como base de muchos interesantes y entretenidos proyectos de electrónica para controlar pantallas, altavoces, micrófonos, motores, etc. Incluso, crear robots sencillos.

En resumen, Python sirve para crear todo tipo de herramientas, se puede ejecutar sobre distintos sistemas operativos, es compatible con otros lenguajes de programación y ofrece librerías y frameworks especializados en distintas áreas de conocimiento.

## 3. Python es un proyecto de software libre (free software)

*Python Project* nació como un proyecto de código libre (*free software*). Hasta hace muy poco, todavía era dirigido por su creador, Guido van Rossum, quien, nuevo guiño a Monty Python, ejerció durante casi tres décadas el rol de «benevolente dictador vitalicio» (*Benevolent dictator for life*) de la PSF.

### ¿Qué caracteriza al software libre?

El software libre, *“free software”* no es necesariamente siempre gratuito (aunque Python sí lo sea), sino que se caracteriza por el escrupuloso respeto a las conocidas como “4 libertades”:

* La libertad de usar el programa, con cualquier propósito. (libertad 0)
* La libertad de estudiar cómo funciona el programa, y adaptarlo a tus necesidades. (libertad 1)
* La libertad de distribuir copias, con lo que puedes ayudar a tu vecino. (libertad 2)
* La libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie. (libertad 3)
* Para que las libertades 1 y 3 sean posibles es necesario que los usuarios puedan tener acceso al código fuente de los programas.

Software libre son aquellos programas que una vez obtenidos pueden ser usados, copiados, estudiados, modificados y redistribuidos libremente. Por tanto, la “libertad” del software está relacionada con los permisos que su autor ofrece y no con su precio.

Python se publica bajo la licencia *Python Software Foundation License*. La PSF es una organización sin ánimo de lucro, que fue creada en 2001 con el objetivo de gestionar el proyecto (desarrollo, administración de derechos, recaudación de fondos, etc). Y es compatible con la GPL de GNU (Licencia Pública General de GNU, a partir de la versión 2.1.1).

## 4. Python es un lenguaje de código abierto (open source)

Además de libre, Python es un lenguaje de código abierto, que es parecido, pero no es lo mismo. Según Richard Stallman, tanto el software libre como el código abierto persiguen un objetivo común: dar mayor libertad y transparencia al mundo del software. Sin embargo, difieren en la manera de llevarlo a cabo.

El software libre se define por su ética. Se considera software libre no solo los programas cuyo código esté abierto, sino todos aquellos programas que respetan las cuatro libertades esenciales del usuario definidas por la *Free Software Foundation* (1985).

El concepto de software abierto surgió en 1998, cuando se creó la OSI (*Open source initiative*) como una escisión del software libre. En este caso, en lugar de las 4 libertades del software libre, se definieron 10 requisitos que debe cumplir un software para ser considerado abierto.

La diferencia principal entre uno y otro tipo de software es sutil. El software libre prioriza los aspectos éticos, mientras que el software de código abierto prioriza los de índole técnica. Por ello, es normalmente menos estricto, de modo que todo el software libre es también de código abierto, pero no necesariamente a la inversa.

En cualquier caso, Python es libre y abierto. Y en esto radica una de las claves del éxito de Python: la comunidad Python. Se trata de una comunidad amplia y muy activa, que contribuye al desarrollo y mejora del código fuente, según las necesidades y demandas de los usuarios. Porque, aunque muchas empresas y organizaciones, como Google, Microsoft o Red Hat, utilizan mucho este lenguaje y tienen influencia en su evolución, ninguna ejerce un control sobre el mismo.

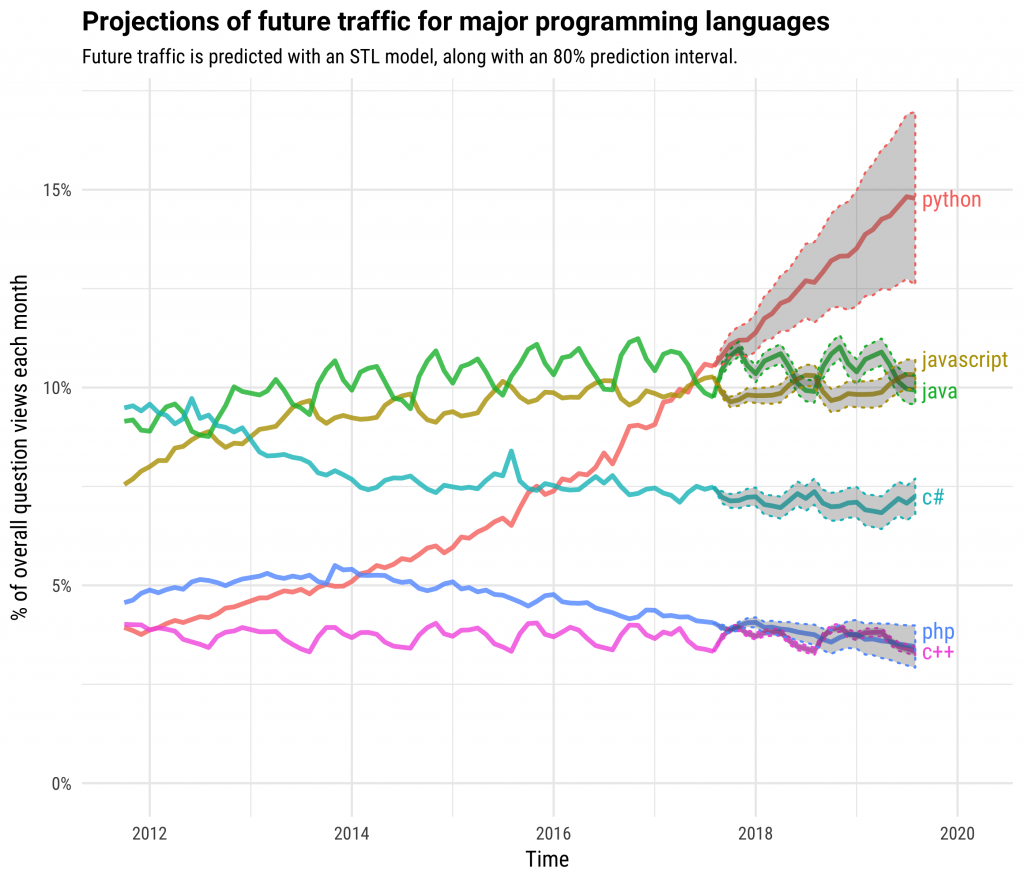
Este carácter libre y abierto, sin duda, también ha facilitado la versatilidad, flexibilidad y potencia mencionadas en el punto anterior, ya que además de todo lo mencionado anteriormente, Python es un lenguaje multiplataforma. Es decir, podemos ejecutarlo en diferentes sistemas operativos como Windows o Linux simplemente usando el intérprete correspondiente.

## 5. Y además… es gratis

Como hemos apuntado antes, a pesar de la confusión que puede generar el hecho de que, en inglés, *“free”*, además de libre, significa gratis, el software libre no tiene por qué ser necesariamente gratis. Sin embargo, sí se puede afirmar, que para programar en Python, no es necesario pagar ningún tipo de licencia. Sin embargo, no debemos olvidar nunca que cualquier código no desarrollado por nosotros mismos puede estar sometido a algún tipo de licenciamiento.

## Conclusión

La sencillez, versatilidad, y potencia de Python lo han convertido en ese lenguaje de programación todoterreno que puede ayudar a impulsar la alfabetización digital, y puede hacer accesible la programación a personas y profesionales de todo tipo.



<https://stackoverflow.blog/2017/09/06/incredible-growth-python/>

# 

# Instalación

Tienes a disposición en el Aula Virtual un video donde se explica cómo instalar Python en Windows.

Si eres usuario Linux o Mac, es probable que ya tengas instalado el intérprete de Python. Aquí tienes la página donde lo puedes descargar para estos sistemas operativos: <https://www.python.org/downloads/>

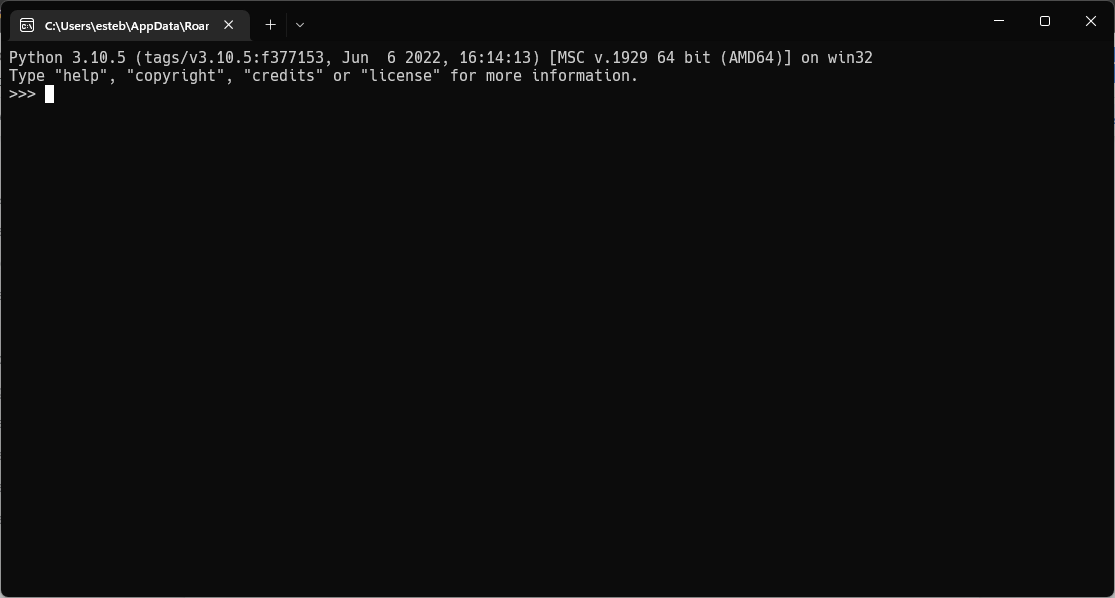
Una vez instalado, abre la terminal (también llamada consola / símbolo de sistema / Powershell / bash) y ejecuta:

**python**

En un sistema operativo que no sea Windows, quizás debas escribir python3. En Linux puedes usar el siguiente comando en la consola para usar en adelante el comando python y no python3:

**sudo apt install python-is-python3**

Se abrirá el "**Intérprete de Python**":



Si no se llegara abrir, mira de vuelta el video, (recuerda, en Windows, activar agregar path de Python a las variables de entorno) o sigue los pasos que te indica la página oficial de Python.