

## Introducción

### Historia de Python

Python fue creado a finales de los 80 por [Guido Van Rossum](#) (@gvanrossum en Twitter/X) y desde entonces se ha ido mejorando y desarrollando por una enorme comunidad de usuarios. Si quieres saber más sobre su historia, puedes encontrar más detalles en la [página sobre la historia de Python en la wikipedia](#).

### Algunas características de Python

- [Debe su nombre al grupo Monty Python](#), del que su creador Guido es un gran fan (no sé porqué he puesto este dato en primer lugar...).
- Python es open source, lo que quiere decir que se puede usar en todos los sistemas operativos y por supuesto totalmente gratuito. En [este repositorio de github, llamado cPython \(código C de Python\)](#) puedes encontrar su código fuente, al que han contribuido casi 1500 desarrolladores.
- En su [página web](#) podemos encontrar documentación y tutoriales de todos los niveles. También podemos descargar [desde su página](#) todas las versiones y para casi todos los sistemas operativos.
- La versión actual a día de hoy (5 de Julio de 2025) es la 3.13, estando la 3.14 en la fase beta de desarrollo. Puedes ver los cambios que traerá en la [página de la versión 3.14](#).

Sobre las ramas anteriores más recientes se arreglan errores, sobre las más antiguas sólo se arreglan temas de seguridad y la versión 2.7, presentada en 2010, se considera que ya llegó al fin de su vida útil en 2020.

- Existe la [Python Software Foundation \(PSF\)](#) que ayuda a la creación de eventos globales y locales. En 2019 entregó más de 300.000\$ en becas y ayudas en 60 países.
- ¿Para qué sirve Python? Es un lenguaje de propósito general, es decir podemos usarlo para hacer casi cualquier tipo de programa o aplicación. Por ejemplo:
  - Aplicaciones web: de hecho existen varios framework (entornos de desarrollo) como django, especializados en la creación de páginas web
  - Aplicaciones de escritorio en cualquier plataforma
  - Scripts (ficheros de lotes) para automatizar tareas repetitivas.
  - Creación de juegos, tenemos multitud de módulos que nos facilitan enormemente la creación de videojuegos.
- Es multiplataforma, es decir podemos usar nuestro código en cualquier plataforma (siempre que todos los módulos que usemos estén disponibles): Windows, MacOS, Linux, Unix,... De hecho viene instalado en los sistemas operativos Linux y MacOS, puesto que parte de los programas incluidos en éstos, lo usan.

Cuando decimos que es multiplataforma también nos referimos a que existe una versión de Python adaptada para ejecutarse en dispositivos embebidos y microcontroladores llamada [microPython](#) que funciona en equipos con reducidas prestaciones.

- Usa una sintaxis muy sencilla y fácil de comprender, lo que nos ayuda a aprender a programar leyendo otros programas. De hecho en su creación se puso un gran énfasis en conseguir un código muy legible.
- La estructura del código es más legible a simple vista.
- Es muy didáctico y uno de los lenguajes más adecuados para quien no ha programado nunca, y actualmente muchas universidades lo utilizan como lenguaje de entrada al mundo de la programación
- La curva de aprendizaje tiene una pendiente muy alta, pero un escalón inicial muy bajo, lo que nos permite empezar muy rápido a hacer cosas interesantes, casi desde el primer momento.
- Cuenta con multitud de librerías y módulos, lo que nos facilita mucho la tarea de programar, sólo tenemos que buscar si ya existe un módulo que nos sirva y adaptar algún ejemplo. Además estos módulos suelen ser también open source.

Algunos de estos módulos están programados completamente en Python y otros son recubrimientos de Python que internamente usan librerías escritas en C para optimizar el rendimiento

Podemos encontrar módulos para procesamiento de imágenes o para Data Mining (procesamiento masivo de datos) o para trabajar en IA (inteligencia artificial).

- Por todo esto se ha convertido en un estándar de hecho en el mundo de la ciencia, en todo lo relacionado con la investigación de IA, visión artificial, bioinformática, genética y en la investigación científica.
- Casi todas las grandes empresas del mundo del software utilizan Python, con Google a la cabeza, seguida por Facebook, Industrial Light & Magic (la de efectos especiales), Instagram, Spotify, Netflix, Dropbox,... hasta la mismísima NASA. En [esta página](#) puedes ver casos de éxito de desarrollos en Python.
- También son muchos los programas/aplicaciones/webs que usamos cada día y que no funcionarían sin Python, como por ejemplo Youtube, Google, Instagram, Reddit, Blender, Dropbox, BitTorrent, OpenShot,

Algunas características técnicas de Python

- Es interpretado: lo que quiere decir que nuestro código se va ejecutando línea a línea, sin que se haga una traducción global a un ejecutable. Eso nos permite que una de las formas de desarrollar nuestro código sea desde una consola interactiva, donde vamos añadiendo las órdenes y estas se ejecutan. Posteriormente podemos pasar estas órdenes a un archivo. Sin duda es una de las características más importantes de Python. También existen compiladores de Python que nos permiten convertir nuestro código a un ejecutable que el ordenador entiende directamente, pero no es la forma habitual de trabajar con Python

Al ser interpretado nos podemos encontrar con que el rendimiento de nuestros programas no es el mejor y no llegamos a exprimir la potencia del ordenador.

Con la potencia de los ordenadores de hoy en día y utilizando determinados módulos y características podemos mejorar mucho el rendimiento de nuestro código

Puedes encontrar algo más de detalle sobre este tema en el siguiente apéndice.

- Python no utiliza delimitadores para indicar los bloques de código como otros lenguajes (`{...}` en C, C++ o Java), si no que la indentación de las líneas (los espacios iniciales), marca estos bloques. Esto nos obliga a que la estructura del código sea más legible a simple vista.
- Tiene una comunidad muy activa, tanto en el desarrollo del lenguaje como en la documentación y la cantidad de módulos que se crean.