# A Instalación y resolución de problemas



Hay muchas versiones de Python disponibles y numerosas formas de configurarlo en cada sistema operativo. Si el método del Capítulo 1 no ha funcionado, o si quieres instalar una versión de Python distinta de la que tienes instalada, las instrucciones de este apéndice pueden ayudarte.

## Python en Windows

Las instrucciones del Capítulo 1 te muestran cómo instalar Python utilizando el instalador oficial en [https://python.org.](https://python.org) Si no has conseguido que Python se ejecute después de utilizar el instalador, las instrucciones de resolución de problemas de esta sección deberían ayudarte a poner Python en marcha.

### Utilizar py en lugar de python

Si ejecutas un instalador reciente de Python y luego emites el comando python en un terminal, deberías ver el prompt de Python para una sesión de terminal (>>>). Cuando Windows no reconozca el comando python, abrirá la Microsoft Store porque cree que Python no está instalado, o recibirás un mensaje como "No se encontró Python". Si se abre el Microsoft Store, ciérralo; es mejor utilizar el instalador oficial de Python de <https://python.org> que el que mantiene Microsoft.

La solución más sencilla, sin hacer ningún cambio en tu sistema, es probar el comando py. Se trata de una utilidad de Windows que encuentra la última versión de Python instalada en tu sistema y ejecuta ese intérprete. Si este comando funciona y quieres utilizarlo, simplemente utiliza py en cualquier lugar donde veas el comando python o python3 en este libro.

### Volver a ejecutar el instalador

La razón más común por la que python no funciona es que la gente olvida seleccionar la opción Añadir Python al PATH al ejecutar el instalador; es un error fácil de cometer. La variable PATH es una configuración del sistema que indica a Python dónde buscar los programas más utilizados. En este caso, Windows no sabe cómo encontrar el intérprete de Python.

La solución más sencilla en esta situación es volver a ejecutar el instalador. Si hay un nuevo instalador disponible en [https://python.org,](https://python.org) descárgalo y ejecútalo, asegurándote de marcar la casilla **Añadir Python al PATH**.

Si ya tienes el último instalador, ejecútalo de nuevo y selecciona la opción **Modificar**. Verás una lista de funciones opcionales; mantén seleccionadas las opciones por defecto en esta pantalla. A continuación, haz clic en **Siguiente** y marca la casilla **Añadir Python a las variables de entorno**. Por último, haz clic en **Instalar**. El instalador reconocerá que Python ya está instalado, y añadirá la ubicación del intérprete de Python a la variable PATH. Asegúrate de cerrar los terminales que tengas abiertos, porque seguirán utilizando la antigua variable PATH. Abre una nueva ventana de terminal y emite de nuevo el comando python; deberías ver un prompt de Python (>>>).

## Python en macOS

Las instrucciones de instalación del Capítulo 1 utilizan el instalador oficial de Python en [https://python.org.](https://python.org) El instalador oficial lleva años funcionando bien, pero hay algunas cosas que pueden despistarte. Esta sección te ayudará si algo no funciona de forma sencilla.

### Instalar accidentalmente la versión de Python de Apple

Si ejecutas el comando python3 y Python aún no está instalado en tu sistema, lo más probable es que veas un mensaje indicando que es necesario instalar *command line developer tools*. Lo mejor en este momento es cerrar la ventana emergente que muestra este mensaje, descargar el instalador de Python desde [https://python.org,](https://python.org) y ejecutar el instalador.

Si decides instalar las herramientas de desarrollo de la línea de comandos en este punto, macOS instalará la versión de Apple de Python junto con las herramientas de desarrollo. El único problema es que la versión de Python de Apple suele ir algo por detrás de la última versión oficial de Python. Sin embargo, puedes descargar y ejecutar el instalador oficial desde <https://python.org>, y python3  apuntará entonces a la versión más reciente. No te preocupes por tener instaladas las herramientas para desarrolladores; hay algunas herramientas útiles, como el sistema de control de versiones Git, del que se habla en el Apéndice D.

### Python 2 en versiones antiguas de macOS

En versiones antiguas de macOS, anteriores a Monterey (macOS 12), se instalaba por defecto una versión obsoleta de Python 2. En estos sistemas, el comando python apunta al intérprete anticuado del sistema. Si utilizas una versión de macOS con Python 2 instalado, asegúrate de utilizar el comando python3, y siempre utilizarás la versión de Python que hayas instalado.

## Python en Linux

Python está incluido por defecto en casi todos los sistemas Linux. Sin embargo, si la versión por defecto de tu sistema es anterior a Python 3.9, debes instalar la última versión. También puedes instalar la última versión si quieres las características más recientes, como los mensajes de error mejorados de Python. Las siguientes instrucciones deberían funcionar para la mayoría de los sistemas basados en apt.

### Utilizar la instalación por defecto de Python

Si quieres utilizar la versión de Python a la que apunta python3, asegúrate de tener instalados estos tres paquetes adicionales:

$ sudo apt install python3-dev python3-pip python3-venv

Estos paquetes incluyen herramientas útiles para desarrolladores y herramientas que te permiten instalar paquetes de terceros, como los utilizados en la sección de proyectos de este libro.

### Instalar la última versión de Python

Utilizaremos un paquete llamado deadsnakes, que facilita la instalación de varias versiones de Python. Introduce los siguientes comandos:

$ sudo add-apt-repository ppa:deadsnakes/ppa  
$ sudo apt update  
$ sudo apt install python3.11

Estos comandos instalarán Python 3.11 en tu sistema.

Introduce el siguiente comando para iniciar una sesión de terminal que ejecute Python 3.11:

$ python3.11  
>>>

En cualquier lugar donde veas el comando python en este libro, utiliza en su lugar python3.11. También querrás utilizar este comando cuando ejecutes programas desde el terminal.

Necesitarás instalar dos paquetes más para sacar el máximo partido a tu instalación de Python:

$ sudo apt install python3.11-dev python3.11-venv

Estos paquetes incluyen módulos que necesitarás al instalar y ejecutar paquetes de terceros, como los utilizados en los proyectos de la segunda mitad del libro.

## Nota

El paquete deadsnakes se ha mantenido activamente durante mucho tiempo. Cuando salgan nuevas versiones de Python, puedes utilizar estos mismos comandos, sustituyendo python3.11 por la última versión disponible en ese momento.

## Comprobar qué versión de Python estás utilizando

Si tienes algún problema al ejecutar Python o al instalar paquetes adicionales, puede ser útil saber exactamente qué versión de Python estás utilizando. Puedes tener varias versiones de Python instaladas y no tener claro qué versión se está utilizando actualmente.

Ejecuta el siguiente comando en un terminal:

$ python --version  
Python 3.11.0

Esto te dice exactamente a qué versión está apuntando actualmente el comando python. El comando más corto python -V dará el mismo resultado.

## Palabras clave y funciones incorporadas de Python

Python viene con su propio conjunto de palabras clave y funciones incorporadas. Es importante tenerlas en cuenta cuando nombres cosas en Python: tus nombres no pueden coincidir con estas palabras clave y no deben coincidir con los nombres de las funciones, o sobrescribirás las funciones.

En esta sección, enumeraremos las palabras clave de Python y los nombres de las funciones incorporadas, para que sepas qué nombres debes evitar.

### Palabras clave de Python

Cada una de las siguientes palabras clave tiene un significado específico, y verás un error si intentas utilizar cualquiera de ellas como nombre de variable.

False await else import pass  
None break except in raise  
True class finally is return  
and continue for lambda try  
as def from nonlocal while  
assert del global not with  
async elif if or yield

### Funciones incorporadas de Python

No obtendrás un error si utilizas una de las siguientes funciones incorporadas fácilmente disponibles como nombre de variable, pero anularás el comportamiento de esa función:

abs() hash() slice()  
aiter() help() sorted()  
all() hex() staticmethod()  
any() id() str()  
anext() input() sum()  
ascii() int() super()  
bin() isinstance() tuple()  
bool() issubclass() type()  
breakpoint() iter() vars()  
bytearray() len() zip()  
bytes() list() \_\_import\_\_()  
callable() locals()  
chr() map()  
classmethod() max()  
compile() memoryview()  
complex() min()  
delattr() next()  
dict() object()  
dir() oct()  
divmod() open()  
enumerate() ord()  
eval() pow()  
exec() print()  
filter() property()  
float() range()  
format() repr()  
frozenset() reversed()  
getattr() round()  
globals() set()  
hasattr() setattr()