# 1 Cómo empezar



En este capítulo, ejecutarás tu primer programa Python, *hello\_world.py*. En primer lugar, tendrás que comprobar si tienes instalada en tu ordenador una versión reciente de Python; si no es así, la instalarás. También instalarás un editor de texto para trabajar con tus programas Python. Los editores de texto reconocen el código Python y resaltan las secciones a medida que escribes, facilitando la comprensión de la estructura de tu código.

## Configurar tu entorno de programación

Python difiere ligeramente en los distintos sistemas operativos, por lo que tendrás que tener en cuenta algunas consideraciones. En las siguientes secciones, nos aseguraremos de que Python está configurado correctamente en tu sistema.

### Versiones de Python

Todo lenguaje de programación evoluciona a medida que surgen nuevas ideas y tecnologías, y los desarrolladores de Python han hecho que el lenguaje sea cada vez más versátil y potente. En el momento de escribir estas líneas, la última versión es Python 3.11, pero todo lo que aparece en este libro debería funcionar con Python 3.9 o posterior. En esta sección, averiguaremos si Python ya está instalado en tu sistema y si necesitas instalar una versión más reciente. El Apéndice A también contiene detalles adicionales sobre la instalación de la última versión de Python en cada uno de los principales sistemas operativos.

### Ejecutar fragmentos de código Python

Puedes ejecutar el intérprete de Python en una ventana de terminal, lo que te permite probar fragmentos de código Python sin tener que guardar y ejecutar un programa entero.

A lo largo de este libro, verás fragmentos de código con este aspecto:

>>> print("Hello Python interpreter!")  
Hello Python interpreter!

La indicación de tres corchetes angulares (>>>), a la que nos referiremos como *Python prompt*, indica que debes utilizar la ventana del terminal. El texto en negrita es el código que debes escribir y luego ejecutar pulsando ENTER. La mayoría de los ejemplos de este libro son pequeños programas autónomos que ejecutarás desde tu editor de textos en lugar de desde el terminal, porque escribirás la mayor parte del código en el editor de textos. Pero a veces, los conceptos básicos se mostrarán en una serie de fragmentos ejecutados a través de una sesión de terminal de Python para demostrar conceptos concretos de forma más eficaz. Cuando veas tres corchetes angulares en un listado de código, estarás viendo código y salida de una sesión de terminal. Dentro de un momento intentaremos codificar en el intérprete de tu sistema.

También utilizaremos un editor de texto para crear un sencillo programa llamado *Hello World!* que se ha convertido en un elemento básico para aprender a programar. Existe una larga tradición en el mundo de la programación según la cual imprimir el mensaje Hello world! en la pantalla como primer programa en un nuevo lenguaje te traerá buena suerte. Un programa tan sencillo tiene un propósito muy real. Si se ejecuta correctamente en tu sistema, cualquier programa Python que escribas también debería funcionar.

### Acerca del editor de código VS

*VS Code* es un potente editor de texto de calidad profesional, gratuito y apto para principiantes. VS Code es ideal tanto para proyectos sencillos como complejos, de modo que si te sientes cómodo utilizándolo mientras aprendes Python, podrás seguir usándolo a medida que avances hacia proyectos más grandes y complicados. VS Code puede instalarse en todos los sistemas operativos modernos, y es compatible con la mayoría de los lenguajes de programación, incluido Python.

Enel Apéndice B encontrarás información sobre otros editores de texto. Si sientes curiosidad por las otras opciones, tal vez quieras hojear ese apéndice en este punto de . Si quieres empezar a programar rápidamente, puedes utilizar VS Code para empezar. Luego puedes considerar otros editores, una vez que hayas adquirido cierta experiencia como programador. En este capítulo, te guiaré a través de la instalación de VS Code en tu sistema operativo.

## Nota

Si ya tienes instalado un editor de texto y sabes cómo configurarlo para ejecutar programas Python, puedes utilizar ese editor en su lugar.

## Python en diferentes sistemas operativos

Python es un lenguaje de programación multiplataforma, lo que significa que funciona en los principales sistemas operativos. Cualquier programa Python que escribas debería funcionar en cualquier ordenador moderno que tenga Python instalado. Sin embargo, los métodos para configurar Python en diferentes sistemas operativos varían ligeramente.

En esta sección aprenderás a configurar Python en tu sistema. Primero comprobarás si hay una versión reciente de Python instalada en tu sistema, y la instalarás si no es así. Luego instalarás VS Code. Estos son los dos únicos pasos que son diferentes para cada sistema operativo.

En las secciones siguientes, ejecutarás *hello\_world.py* y solucionarás cualquier problema que no funcione. Te guiaré a través de este proceso para cada sistema operativo, para que tengas un entorno de programación Python en el que puedas confiar.

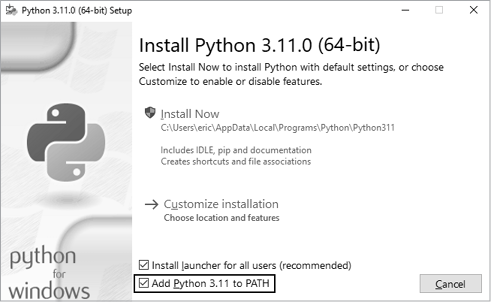
### Python en Windows

Windows no suele venir con Python, así que probablemente tendrás que instalarlo y luego instalar VS Code.

#### Instalar Python

En primer lugar, comprueba si Python está instalado en tu sistema. Abre una ventana de comandos introduciendo command en el menú Inicio y haciendo clic en la aplicación **Símbolo del sistema**. En la ventana de terminal, introduce python en minúsculas. Si recibes como respuesta un prompt de Python (>>>), Python está instalado en tu sistema. Si aparece un mensaje de error indicándote que python no es un comando reconocido, o si se abre la tienda de Microsoft, Python no está instalado. Cierra la tienda de Microsoft si se abre; es mejor descargar un instalador oficial que utilizar la versión de Microsoft.

Si Python no está instalado en tu sistema, o si ves una versión anterior a Python 3.9, necesitas descargar un instalador de Python para Windows. Ve a <https://python.org> y pasa el ratón por encima del enlace **Descargas** . Deberías ver un botón para descargar la última versión de Python. Haz clic en el botón, que debería iniciar automáticamente la descarga del instalador correcto para tu sistema. Cuando hayas descargado el archivo, ejecuta el instalador. Asegúrate de seleccionar la opción **Añadir Python al PATH**, que facilitará la correcta configuración de tu sistema. La [Figura 1-1](#figure1-1) muestra esta opción seleccionada.



Figura

1-1



: Asegúrate de seleccionar la casilla etiquetada *Add Python to PATH*.

#### Ejecutar Python en una sesión de terminal

Abre una nueva ventana de comandos e introduce python en minúsculas. Deberías ver un prompt de Python (>>>), lo que significa que Windows ha encontrado la versión de Python que acabas de instalar.

C:\> python  
Python 3.x.x (main, Jun . . . , 13:29:14) [MSC v.1932 64 bit (AMD64)] on win32  
Type "help", "copyright", "credits" or "license" for more information.  
>>>

## Nota

Si no ves esta salida o algo similar, consulta las instrucciones de instalación más detalladas del Apéndice A.

Introduce la siguiente línea en tu sesión de Python:

>>> print("Hello Python interpreter!")  
Hello Python interpreter!  
>>>

Deberías ver la salida Hello Python interpreter! Cada vez que quieras ejecutar un fragmento de código Python, abre una ventana de comandos e inicia una sesión de terminal Python. Para cerrar la sesión de terminal, pulsa CTRL-Z y después ENTER, o introduce el comando exit().

#### Instalar código VS

Puedes descargar un instalador para VS Code en <https://code.visualstudio.com>. Haz clic en el botón **Descargar para Windows** y ejecuta el instalador. Omite las secciones siguientes sobre macOS y Linux, y sigue los pasos de "Ejecutar un programa Hola Mundo" en la página 9.

### Python en macOS

Python no está instalado por defecto en las últimas versiones de macOS, así que tendrás que instalarlo si aún no lo has hecho. En esta sección, instalarás la última versión de Python y, a continuación, instalarás VS Code y te asegurarás de que está configurado correctamente.

## Nota

Python 2 se incluía en versiones anteriores de macOS, pero es una versión obsoleta que no deberías utilizar.

#### Comprobar si Python 3 está instalado

Abre una ventana de terminal yendo a **Aplicaciones▶Utilidades▶Terminal**. También puedes pulsar la barra espaciadora ⌘, escribir terminal y pulsar INTRO. Para ver si tienes instalada una versión suficientemente reciente de Python, escribe python3. Lo más probable es que veas un mensaje sobre la instalación de *command line developer tools*. Es mejor instalar estas herramientas después de instalar Python, así que si aparece este mensaje, cancela la ventana emergente.

Si la salida muestra que tienes instalado Python 3.9 o una versión posterior, puedes saltarte la siguiente sección y pasar a "Ejecutar Python en una sesión de terminal". Si ves alguna versión anterior a Python 3.9, sigue las instrucciones de la siguiente sección para instalar la última versión.

Ten en cuenta que en macOS, siempre que veas el comando python en este libro, deberás utilizar en su lugar el comando python3 para asegurarte de que estás utilizando Python 3. En la mayoría de los sistemas macOS, el comando python o bien apunta a una versión obsoleta de Python que sólo deberían utilizar las herramientas internas del sistema, o bien no apunta a nada y genera un mensaje de error.

#### Instalar la última versión de Python

Puedes encontrar un instalador de Python para tu sistema en [https://python.org.](https://python.org) Pasa el ratón por encima del enlace **Descargar**, y deberías ver un botón para descargar la última versión de Python. Haz clic en el botón, que debería iniciar automáticamente la descarga del instalador correcto para tu sistema. Una vez descargado el archivo, ejecuta el instalador.

Una vez ejecutado el instalador, debería aparecer una ventana del Finder. Haz doble clic en el archivo *Install Certificates.command*. Ejecutar este archivo te permitirá instalar más fácilmente bibliotecas adicionales que necesitarás para proyectos del mundo real, incluidos los proyectos de la segunda mitad de este libro.

#### Ejecutar Python en una sesión de terminal

Ahora puedes probar a ejecutar fragmentos de código Python abriendo una nueva ventana de terminal y escribiendo python3:

$ python3  
Python 3.x.x (v3.11.0:eb0004c271, Jun . . . , 10:03:01)  
[Clang 13.0.0 (clang-1300.0.29.30)] on darwin  
Type "help", "copyright", "credits" or "license" for more information.  
>>>

Este comando inicia una sesión de terminal de Python. Deberías ver un prompt de Python (>>>), lo que significa que macOS ha encontrado la versión de Python que acabas de instalar.

Introduce la siguiente línea en la sesión de terminal:

>>> print("Hello Python interpreter!")  
Hello Python interpreter!  
>>>

Deberías ver el mensaje Hello Python interpreter!, que debería imprimirse directamente en la ventana de terminal actual. Puedes cerrar el intérprete de Python pulsando CTRL-D o introduciendo el comando exit().

## Nota

En los sistemas macOS más recientes, verás un signo de porcentaje (%) como indicador de terminal en lugar de un signo de dólar ($).

#### Instalar VS Code

Para instalar el editor VS Code, necesitas descargar el instalador en [https://code.visualstudio.com.](https://code.visualstudio.com) Haz clic en el botón **Descargar** y, a continuación, abre una ventana **del Finder** y ve a la carpeta **Descargas**. Arrastra el instalador de **Visual Studio Code** a tu carpeta Aplicaciones y, a continuación, haz doble clic en el instalador para ejecutarlo.

Omite la siguiente sección sobre Python en Linux, y sigue los pasos de "Ejecutar un programa Hola Mundo" en la página 9.

### Python en Linux

Los sistemas Linux están diseñados para programar, por lo que Python ya está instalado en la mayoría de los ordenadores Linux. Las personas que escriben y mantienen Linux esperan que hagas tu propia programación en algún momento, y te animan a hacerlo. Por esta razón, hay muy poco que instalar y sólo unos pocos ajustes que cambiar para empezar a programar.

#### Comprobar tu versión de Python

Abre una ventana de terminal ejecutando la aplicación Terminal en tu sistema (en Ubuntu, puedes pulsar CTRL-ALT-T). Para saber qué versión de Python está instalada, escribe python3 con minúscula *p*. Cuando Python está instalado, este comando inicia el intérprete de Python. Deberías ver una salida indicando qué versión de Python está instalada. También deberías ver un prompt de Python (>>>) donde puedes empezar a introducir comandos de Python:

$ python3  
Python 3.10.4 (main, Apr . . . , 09:04:19) [GCC 11.2.0] on linux  
Type "help", "copyright", "credits" or "license" for more information.  
>>>

Esta salida indica que Python 3.10.4 es actualmente la versión por defecto de Python instalada en este ordenador. Cuando hayas visto esta salida, pulsa CTRL-D o escribe exit() para salir del indicador de Python y volver al indicador de terminal . Siempre que veas el comando python en este libro, introduce python3 en su lugar.

Necesitarás Python 3.9 o posterior para ejecutar el código de este libro. Si la versión de Python instalada en tu sistema es anterior a Python 3.9, o si quieres actualizarte a la última versión disponible actualmente, consulta las instrucciones del Apéndice A.

#### Ejecutar Python en una sesión de terminal

Puedes probar a ejecutar fragmentos de código Python abriendo un terminal e introduciendo python3, como hiciste al comprobar tu versión. Hazlo de nuevo, y cuando tengas Python en ejecución, introduce la siguiente línea en la sesión de terminal:

>>> print("Hello Python interpreter!")  
Hello Python interpreter!  
>>>

El mensaje debería imprimirse directamente en la ventana actual del terminal. Recuerda que puedes cerrar el intérprete de Python pulsando CTRL-D o introduciendo el comando exit().

#### Instalación de VS Code

En Ubuntu Linux, puedes instalar VS Code desde el Centro de Software de Ubuntu. Haz clic en el icono de Software de Ubuntu en tu menú y busca *vscode*. Haz clic en la aplicación llamada **Visual Studio Code** (a veces denominada *code*) y, a continuación, haz clic en **Instalar**. Una vez instalada, busca *VS Code* en tu sistema e inicia la aplicación.

## Ejecutar un programa Hola Mundo

Con una versión reciente de Python y VS Code instalados, estás casi listo para ejecutar tu primer programa Python escrito en un editor de texto. Pero antes de hacerlo, necesitas instalar la extensión de Python para VS Code.

### Instalar la extensión de Python para VS Code

VS Code funciona con muchos lenguajes de programación diferentes; para sacarle el máximo partido como programador de Python, necesitarás instalar la extensión Python. Esta extensión añade soporte para escribir, editar y ejecutar programas Python.

Para instalar la extensión Python, haz clic en el icono Administrar, que parece un engranaje en la esquina inferior izquierda de la aplicación VS Code. En el menú que aparece, haz clic en **Extensiones**. Introduce python en el cuadro de búsqueda y haz clic en la extensión de **Python** . (Si ves más de una extensión llamada *Python*, elige la suministrada por Microsoft). Haz clic en **Instalar** e instala las herramientas adicionales que tu sistema necesite para completar la instalación. Si ves un mensaje indicando que necesitas instalar Python, y ya lo has hecho, puedes ignorar este mensaje.

## Nota

Si estás utilizando macOS y una ventana emergente te pide que instales *command line developer tools*, haz clic en **Instalar**. Es posible que aparezca un mensaje indicando que la instalación tardará demasiado tiempo, pero sólo debería tardar unos 10 ó 20 minutos con una conexión a Internet razonable.

### Ejecutar hola\_mundo.py

Antes de escribir tu primer programa, crea una carpeta llamada *python\_work* en tu escritorio para tus proyectos. Es mejor utilizar letras minúsculas y guiones bajos en lugar de espacios en los nombres de archivos y carpetas, porque Python utiliza estas convenciones de nomenclatura. Puedes hacer esta carpeta en otro lugar que no sea el escritorio, pero será más fácil seguir algunos pasos posteriores si guardas la carpeta *python\_work* directamente en tu escritorio.

Abre VS Code, y cierra la pestaña **Empezar** si aún está abierta. Crea un nuevo archivo haciendo clic en Archivo▶Nuevo**Archivo** o pulsando CTRL-N (⌘-N en macOS). Guarda el archivo como *hello\_world.py* en tu carpeta *python\_work*. La extensión *.py* indica a VS Code que tu archivo está escrito en Python, y le indica cómo ejecutar el programa y resaltar el texto de forma útil.

Después de guardar tu archivo, introduce la siguiente línea en el editor:

**hello\_world.py**

print("Hello Python world!")

Para ejecutar tu programa, selecciona Ejecutar▶Ejecutar**sin depurar** o pulsa CTRL-F5. Debería aparecer una pantalla de terminal en la parte inferior de la ventana de VS Code, mostrando la salida de tu programa:

Hello Python world!

Es probable que veas alguna salida adicional que muestre el intérprete de Python que se utilizó para ejecutar tu programa. Si quieres simplificar la información que se muestra para que sólo veas la salida de tu programa, consulta el Apéndice B. También puedes encontrar sugerencias útiles sobre cómo utilizar VS Code de forma más eficiente en el Apéndice B.

Si no ves esta salida, es posible que algo haya ido mal en el programa. Comprueba cada carácter de la línea que has introducido. ¿Has puesto accidentalmente print en mayúsculas? ¿Olvidaste una o ambas comillas o paréntesis? Los lenguajes de programación esperan una sintaxis muy específica, y si no la proporcionas, obtendrás errores. Si no consigues que el programa se ejecute, consulta las sugerencias de la siguiente sección.

## Solución de problemas

Si no consigues que *hello\_world.py* se ejecute, aquí tienes algunos remedios que puedes probar y que también son buenas soluciones generales para cualquier problema de programación:

* Cuando un programa contiene un error significativo, Python muestra un *traceback*, que es un informe de errores. Python examina el archivo e intenta identificar el problema. Comprueba el rastreo; puede darte una pista sobre qué problema impide que el programa se ejecute.
* Aléjate del ordenador, tómate un breve descanso y vuelve a intentarlo. Recuerda que la sintaxis es muy importante en programación, por lo que algo tan simple como unas comillas o unos paréntesis mal colocados puede impedir que un programa se ejecute correctamente. Vuelve a leer las partes pertinentes de este capítulo, revisa tu código e intenta encontrar el error.
* Vuelve a empezar. Probablemente no necesites desinstalar ningún programa, pero podría tener sentido borrar tu archivo *hello\_world.py* y volver a crearlo desde cero.
* Pide a otra persona que siga los pasos de este capítulo, en tu ordenador o en otro distinto, y observa atentamente lo que hace. Puede que te hayas saltado un pequeño paso que otra persona haya captado por casualidad.
* Consulta las instrucciones adicionales de instalación en el Apéndice A; algunos de los detalles incluidos en el Apéndice pueden ayudarte a resolver tu problema.
* Busca a alguien que sepa Python y pídele que te ayude a instalarlo. Si preguntas por ahí, puede que te encuentres con que conoces inesperadamente a alguien que utiliza Python.
* Las instrucciones de configuración de este capítulo también están disponibles a través del sitio web que acompaña a este libro en [https://ehmatthes.github.io/pcc\_3e.](https://ehmatthes.github.io/pcc_3e) La versión online de estas instrucciones puede funcionar mejor porque puedes simplemente cortar y pegar código y hacer clic en los enlaces a los recursos que necesites.
* Pide ayuda en Internet. El Apéndice C proporciona una serie de recursos, como foros y sitios de chat en directo, donde puedes pedir soluciones a personas que ya han resuelto el problema al que te enfrentas actualmente.

No te preocupes por molestar a programadores experimentados. Todos los programadores se han atascado alguna vez, y la mayoría de ellos estarán encantados de ayudarte a configurar correctamente tu sistema. Siempre que puedas exponer claramente lo que intentas hacer, lo que ya has probado y los resultados que obtienes, es muy probable que alguien pueda ayudarte. Como se ha mencionado en la introducción, la comunidad Python es muy amistosa y acogedora con los principiantes.

Python debería funcionar bien en cualquier ordenador moderno. Los primeros problemas de configuración pueden ser frustrantes, pero merece la pena resolverlos. Una vez que tengas *hello\_world.py* funcionando, podrás empezar a aprender Python, y tu trabajo de programación será más interesante y satisfactorio.

## Ejecutar programas Python desde un terminal

Ejecutarás la mayoría de tus programas directamente en tu editor de texto. Sin embargo, a veces es útil ejecutar programas desde un terminal. Por ejemplo, puedes querer ejecutar un programa existente sin abrirlo para editarlo.

Puedes hacerlo en cualquier sistema con Python instalado si sabes cómo acceder al directorio donde está almacenado el archivo del programa. Para intentarlo, asegúrate de que has guardado el archivo *hello\_world.py* en la carpeta *python\_work* de tu escritorio.

### En Windows

Puedes utilizar el comando de terminal cd, para *change directory*, para navegar por tu sistema de archivos en una ventana de comandos. El comando dir, para *directory*, te muestra todos los archivos que existen en el directorio actual.

Abre una nueva ventana de terminal e introduce los siguientes comandos para ejecutar *hello\_world.py*:

C:\> cd Desktop\python\_work  
C:\Desktop\python\_work> dir  
hello\_world.py  
C:\Desktop\python\_work> python hello\_world.py  
Hello Python world!

En primer lugar, utiliza el comando cd para navegar hasta la carpeta *python\_work*, que se encuentra en la carpeta *Desktop*. A continuación, utiliza el comando dir para asegurarte de que *hello\_world.py* se encuentra en esta carpeta. A continuación, ejecuta el archivo utilizando el comando python hello\_world.py.

La mayoría de tus programas funcionarán bien directamente desde el editor. Sin embargo, a medida que tu trabajo se vuelva más complejo, querrás ejecutar algunos de tus programas desde un terminal.

### En macOS y Linux

Ejecutar un programa Python desde una sesión de terminal es lo mismo en Linux que en macOS. Puedes utilizar el comando de terminal cd, para *change directory*, para navegar por tu sistema de archivos en una sesión de terminal. El comando ls, para *list*, te muestra todos los archivos no ocultos que existen en el directorio actual.

Abre una nueva ventana de terminal e introduce los siguientes comandos para ejecutar *hello\_world.py*:

~$ cd Desktop/python\_work/  
~/Desktop/python\_work$ ls  
hello\_world.py  
~/Desktop/python\_work$ python3 hello\_world.py  
Hello Python world!

En primer lugar, utiliza el comando cd para navegar hasta la carpeta *python\_work*, que se encuentra en la carpeta *Desktop*. A continuación, utiliza el comando ls para asegurarte de que *hello\_world.py* se encuentra en esta carpeta. A continuación, ejecuta el archivo utilizando el comando python3 hello\_world.py.

La mayoría de tus programas funcionarán bien directamente desde tu editor. Pero a medida que tu trabajo se vuelva más complejo, querrás ejecutar algunos de tus programas desde un terminal.

## Pruébalo tú mismo

Los ejercicios de este capítulo son de naturaleza exploratoria. A partir del Capítulo 2, los retos que resolverás se basarán en lo que has aprendido.

1-1. python.org: Explora la página principal de Python[(https://python.org)](https://python.org) para encontrar temas que te interesen. A medida que te familiarices con Python, las distintas partes del sitio te serán más útiles.

1-2. Hola Mundo Typos: Abre el archivo *hello\_world.py* que acabas de crear. Escribe una errata en alguna parte de la línea y vuelve a ejecutar el programa. ¿Puedes hacer una errata que genere un error? ¿Puedes entender el mensaje de error? ¿Puedes hacer una errata que no genere un error? ¿Por qué crees que no generó un error?

1-3. Habilidades infinitas: Si tuvieras infinitas habilidades de programación, ¿qué construirías? Estás a punto de aprender a programar. Si tienes un objetivo final en mente, tendrás un uso inmediato para tus nuevas habilidades; ahora es un buen momento para escribir breves descripciones de lo que quieres crear. Es una buena costumbre llevar un cuaderno de "ideas" al que puedas acudir siempre que quieras empezar un nuevo proyecto. Dedica ahora unos minutos a describir tres programas que quieras crear.

## Resumen

En este capítulo, has aprendido un poco sobre Python en general, y has instalado Python en tu sistema si aún no estaba allí. También has instalado un editor de texto para facilitar la escritura de código Python. Has ejecutado fragmentos de código Python en una sesión de terminal y has ejecutado tu primer programa, *hello\_world.py*. Probablemente también aprendiste un poco sobre resolución de problemas.

En el próximo capítulo, aprenderás sobre los distintos tipos de datos con los que puedes trabajar en tus programas Python, y también empezarás a utilizar variables.