Desconocido

## Introducción



¿Por qué aprender programación informática? La programación fomenta la creatividad, el razonamiento y la resolución de problemas. El programador tiene la oportunidad de crear algo de la nada, utilizar la lógica para convertir las construcciones de programación en una forma que un ordenador pueda ejecutar y, cuando las cosas no funcionan tan bien como se esperaba, utilizar la resolución de problemas para averiguar qué ha ido mal. La programación es una actividad divertida, a veces desafiante (y ocasionalmente frustrante), y las habilidades que se aprenden en ella pueden ser útiles tanto en la escuela como en el trabajo, incluso si tu carrera no tiene nada que ver  con los ordenadores. Y, por si fuera poco, programar es una forma estupenda de pasar una tarde cuando el tiempo fuera es desapacible.

### Por qué Python

Python es un lenguaje de programación fácil de aprender que tiene algunas características realmente útiles para un programador principiante. El código es bastante fácil de leer en comparación con otros lenguajes de programación, y tiene un intérprete de comandos interactivo en el que puedes introducir tus programas y ver cómo se ejecutan.

Además de su sencilla estructura de lenguaje y un intérprete de comandos interactivo con el que experimentar, Python tiene algunas funciones que aumentan enormemente el proceso de aprendizaje y te permiten montar animaciones sencillas para crear tus propios juegos. Una de ellas es el módulo turtle , inspirado en los gráficos Turtle (utilizados por el lenguaje de programación Logo en los años 60) y diseñado para uso educativo. Otro es el módulo tkinter , una interfaz para el conjunto de herramientas de interfaz gráfica de usuario (GUI) Tk, que proporciona una forma sencilla de crear programas con gráficos y animaciones algo más avanzados.

### Cómo aprender a programar

Como con cualquier cosa que intentas hacer por primera vez, siempre es mejor empezar por lo básico, así que empieza por los primeros capítulos y resiste el impulso de saltar a los capítulos posteriores. Nadie puede tocar una sinfonía orquestal la primera vez que coge un instrumento. Los estudiantes de pilotaje no empiezan a pilotar un avión antes de comprender los controles básicos. Los gimnastas no son (normalmente) capaces de hacer volteretas hacia atrás en su primer intento. Si avanzas demasiado deprisa, no sólo no se te quedarán grabadas las ideas básicas, sino que el contenido de los capítulos posteriores te parecerá más complicado de lo que realmente es.

A medida que vayas avanzando en el libro, prueba todos los ejemplos para ver cómo funcionan. La mayoría de los capítulos también incluyen rompecabezas de programación para que los pruebes, que te ayudarán a mejorar tus habilidades de programación. Recuerda que cuanto mejor entiendas lo básico, más fácil te resultará comprender ideas más complicadas más adelante. Cuando encuentres algo frustrante o demasiado desafiante, he aquí algunas cosas que a mí me resultan útiles:

1. Divide un problema en trozos más pequeños. Intenta comprender lo que hace un pequeño trozo de código o piensa sólo en una pequeña parte de una idea difícil (céntrate en un pequeño trozo de código en lugar de intentar comprenderlo todo de una vez).
2. Si aun así no te ayuda, lo mejor será que lo dejes durante un tiempo. Consúltalo con la almohada y vuelve a él otro día. Ésta es una buena forma de resolver muchos problemas, y puede ser especialmente útil para los programadores informáticos.

### Quién debería leer este libro

Este libro es para cualquier persona interesada en la programación informática, ya sea un niño o un adulto que se acerca a la programación por primera vez. Si quieres aprender a escribir tu propio software en lugar de limitarte a utilizar los programas desarrollados por otros, *Python para niños* es un buen punto de partida.

En los siguientes capítulos, encontrarás información que te ayudará a instalar Python, iniciar el Shell de Python y realizar cálculos básicos, imprimir texto en la pantalla y crear listas, y realizar operaciones sencillas de flujo de control utilizando las sentencias if y los bucles for (¡y aprender qué son las sentencias if y los bucles for !). Aprenderás a reutilizar código con funciones, los fundamentos de las clases y los objetos, y descripciones de algunas de las muchas funciones y módulos incorporados de Python.

Encontrarás capítulos sobre gráficos tortuga sencillos y avanzados, así como sobre el uso del módulo tkinter para dibujar en la pantalla del ordenador. Al final de muchos capítulos hay puzzles de programación de diversa complejidad, que te ayudarán a afianzar tus nuevos conocimientos dándote la oportunidad de escribir pequeños programas por ti mismo. Una vez que hayas consolidado tus conocimientos fundamentales de programación, aprenderás a escribir tus propios juegos. Desarrollarás dos juegos gráficos y aprenderás sobre detección básica de colisiones, eventos y diferentes técnicas de animación.

La mayoría de los ejemplos de este libro utilizan el shell IDLE (Entorno Integrado de Desarrollo) de Python. IDLE proporciona resaltado de sintaxis, funciones de copiar y pegar (similares a las que utilizarías en otras aplicaciones) y una ventana de editor en la que puedes guardar tu código para utilizarlo más adelante. Esto significa que IDLE  funciona tanto como un entorno interactivo para la experimentación como algo parecido a un editor de texto. Los ejemplos funcionarán igual de bien con la consola estándar y un editor de texto normal, pero el resaltado de sintaxis y el entorno ligeramente más fácil de usar de IDLE pueden facilitar la comprensión, por lo que el primer capítulo te muestra cómo configurarlo.

### ¿Qué contiene este libro?

He aquí un breve resumen de lo que encontrarás en cada capítulo.

[**El Capítulo 1**](ch01.xhtml#ch01) es una introducción a la programación con instrucciones para instalar Python por primera vez.

El Capítulo[**2**](ch02.xhtml#ch02) presenta cálculos básicos y variables, y [**el Capítulo 3**](ch03.xhtml#ch03) describe algunos de los tipos básicos de Python, como cadenas, listas y tuplas.

El Capítulo[**4**](ch04.xhtml#ch04) es la primera toma de contacto con el módulo turtle . Pasaremos de la programación básica a mover una tortuga (en forma de flecha) por la pantalla.

El Capítulo[**5**](ch05.xhtml#ch05) cubre las variaciones de las condiciones y las sentencias if , y [**el Capítulo**](ch06.xhtml#ch06) 6 pasa a los bucles for y los bucles while .

En el Capítulo[**7**](ch07.xhtml#ch07) empezamos a utilizar y crear funciones, y en [**el Capítulo**](ch08.xhtml#ch08) 8 tratamos las clases y los objetos. Cubrimos suficientes ideas básicas para apoyar algunas de las técnicas de programación que necesitaremos en los capítulos de desarrollo de juegos más adelante en el libro. En este punto, el material empieza a complicarse un poco más.

En el Capítulo[**9**](ch09.xhtml#ch09) volvemos al módulo turtle para experimentar con formas más complicadas. En [**el Capítulo 10**](ch10.xhtml#ch10) pasamos a utilizar el módulo tkinter para la creación de gráficos más avanzados.

En los Capítulos [**11**](ch11.xhtml#ch11) y [**12**](ch12.xhtml#ch12) , creamos nuestro primer juego, *¡Rebota!* , que se basa en los conocimientos adquiridos en los capítulos anteriores, y en [**los Capítulos 13**](ch13.xhtml#ch13) [**- 16**](ch16.xhtml#ch16) , creamos otro juego, *Mr. Stick Man Races for the Exit .* En los capítulos de desarrollo del juego es donde las cosas pueden empezar a ir muy mal. Si todo lo demás falla, descarga el código del sitio web complementario ( [*http://python-for-kids.com*](http://python-for-kids.com) ), y compara tu código con estos ejemplos que funcionan.

Por último, en el [**Epílogo**](aft.xhtml) , vemos brevemente cómo utilizar el instalador de paquetes de Python ( pip ) para instalar el módulo PyGame y un breve ejemplo PyGame , antes de ver algunos ejemplos de otros lenguajes de programación.

En el [**Apéndice A**](app01.xhtml#app01) , encontrarás una lista de las palabras clave de Python, y en el [**Apéndice B**](app02.xhtml#app02) , una lista de algunas de las útiles funciones incorporadas (aprenderás qué son las *palabras clave* y las *funciones* más adelante en el libro). En [**el Apéndice C**](app03.xhtml#app03) encontrarás información para solucionar problemas comunes.

### Sitio web de Python para niños

Si necesitas ayuda mientras lees, visita el sitio web http://python-for-kids.com [*.*](http://python-for-kids.com) Allí encontrarás descargas para todos los ejemplos de este libro y enlaces a más información, incluido dónde puedes descargar el código fuente utilizado en el libro.

### ¡Diviértete!

Recuerda que programar puede ser divertido. No pienses en ello como un trabajo. Piensa en la programación como una forma de crear juegos o aplicaciones divertidas que puedes compartir con tus amigos o con otras personas.

Aprender a programar es un ejercicio mental maravilloso, y los resultados pueden ser muy gratificantes. Pero sobre todo, hagas lo que hagas, ¡diviértete!