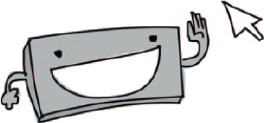
Desconocido

### Crear un botón clicable

Para nuestro primer ejemplo, utilizaremos el módulo tkinter para crear una aplicación básica con un botón. Introduce este código:

>>> from tkinter import \*  
>>> tk = Tk()  
>>> btn = Button(tk, text='click me')  
>>> btn.pack()

En la primera línea, importamos el contenido del módulo tkinter . La línea from *module-name* import \* nos permite utilizar el contenido de un módulo sin utilizar su nombre. En cambio, al utilizar import turtle en los ejemplos anteriores, necesitábamos incluir el nombre del módulo para acceder a su contenido, de esta forma:



import turtle  
t = turtle.Turtle()

Cuando utilizamos import \* , no necesitamos llamar a turtle.Turtle , como hicimos en los [Capítulos 4](ch04.xhtml#ch04) y [9 .](ch09.xhtml#ch09) Esto es muy útil cuando utilizas módulos con muchas clases y funciones, porque reduce la cantidad que tienes que escribir:

from turtle import \*  
t = Turtle()

En la siguiente línea de nuestro ejemplo del botón, creamos una variable que contiene un objeto de la clase Tk con tk = Tk() , igual que creamos un objeto Turtle para la tortuga. El objeto tk crea una ventana básica a la que luego podemos añadir otras cosas, como botones, cuadros de entrada o un lienzo en el que dibujar. Ésta es la clase principal que proporciona el módulo tkinter ; sin crear un objeto de la clase Tk , no podrás hacer gráficos ni animaciones.

En la tercera línea, creamos un botón con btn = Button , y pasamos la variable tk como primer parámetro y "haz clic en mí" como texto que mostrará el botón con text=’click me’ . Aunque hemos añadido este botón a la ventana, no se mostrará hasta que introduzcas la línea btn.pack() , que indica al botón que aparezca. También alinea todo correctamente en la pantalla si  hay otros botones u objetos que mostrar. El resultado debería ser algo parecido a [la Figura 10-1 .](ch10.xhtml#ch10fig01)

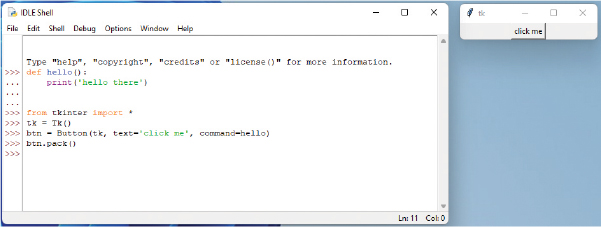


Figura 10-1*: Una aplicación tkinter con un solo* botón

Ahora mismo, el botón Haz clic en mí no hace gran cosa. Puedes pulsarlo todo el día, pero no pasará nada hasta que cambiemos un poco el código. (¡Asegúrate de cerrar la ventana que has creado antes!)

Primero, creamos una función para imprimir un texto:

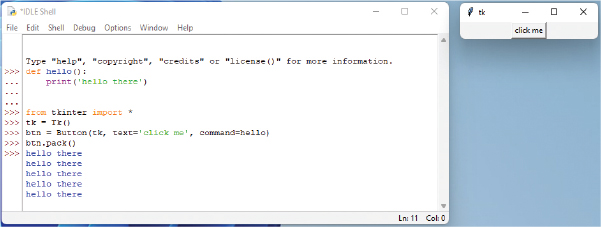
>>> def hello():  
 print('hello there')

A continuación, modificamos nuestro ejemplo para utilizar esta nueva función:

>>> from tkinter import \*  
>>> tk = Tk()  
>>> btn = Button(tk, text='click me', command=hello)  
>>> btn.pack()

Sólo hemos hecho un ligero cambio en la versión anterior de este código, añadiendo el parámetro command , que indica a Python que utilice la función hello cuando se pulse el botón.

Ahora, cuando pulses el botón, verás escrito "hola" en la Shell de Python. Esto aparecerá cada vez que se pulse el botón. En [la Figura 10-2](ch10.xhtml#ch10fig02) , he pulsado el botón cinco veces.



Figura*10-2: Pulsar el* botón

Ésta es la primera vez que utilizamos parámetros con nombre en cualquiera de nuestros ejemplos de código, así que vamos a hablar un poco de ellos antes de continuar con nuestro dibujo.

[anterior](ch10_1.html)[Subtema 2 de 17: (Ver todo)](ch10.html)[siguiente](ch10_3.html)