Aprende a programar con Minecraft

### **Utilizar un módulo de pip: Flask**

Flask es un módulo de Python que puedes utilizar para desarrollar sitios web. En esta sección, aprenderás a configurar un sitio web básico de Flask y luego a integrar el sitio web con Minecraft, ¡para que puedas mostrar la posición de tu jugador en un sitio web!

Con Flask, sólo necesitas unas pocas líneas de código para crear y gestionar un sitio web. Sólo tienes que escribir el código Python como lo harías normalmente y añadir a alguna información extra relacionada con Flask. A continuación, ejecutas tu código, y éste crea un sitio web al que tu ordenador puede acceder. Entonces podrás ver el sitio web en tu navegador.

[El listado 11-5](ch11.xhtml#ch11ex5) crea un sitio web Flask básico que incluye la información más importante sobre mí: mi nombre.

*namePage.py*

from flask import Flask  
➊ app = Flask(\_\_name\_\_)  
  
➋ @app.route("/")  
def showName():  
➌ return "Craig Richardson"  
  
➍ app.run()

*Listado 11-5: Un programa Python que utiliza* Flask *para crear un* sitio web

Para utilizar Flask, primero tienes que crear Flask utilizando la función Flask() ➊. El argumento \_\_name\_\_ indica a Flask que tu proyecto Flask está contenido en este archivo, y no necesita buscar en ningún otro sitio para encontrar otras partes del programa. Observa los dos guiones bajos, no uno, al principio y al final del argumento \_\_name\_\_.

La etiqueta @app.route( ) utiliza un decorador. *Los decor* adores proporcionan información adicional a Python sobre tus funciones. Por ejemplo, en este programa el decorador @app .route () indica a Flask en qué parte del sitio web se utilizará la función. En este caso, "/" le dice a Flask que la función showName () se utilizará en la página de inicio ➋. La declaración de retorno de la función indica a Flask lo que se mostrará en la página. En este ejemplo, devuelve mi nombre, por lo que mi nombre se mostrará en la página ➌. La última línea del programa indica a Python que inicie Flask cuando se ejecute este archivo ➍.

Guarda este archivo como *nombrePágina.py* en la carpeta de *archivos*. Añade tu propio texto para crear tu propio sitio web.

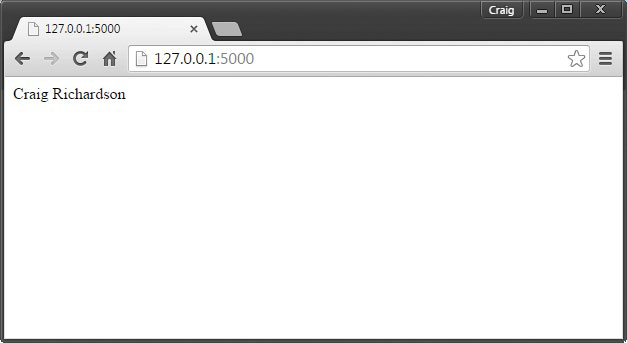
Para ejecutar el sitio web, haz clic en **Ejecutar** image **Ejecutar módulo** en IDLE. El programa se ejecuta y genera un archivo de sitio web que puedes abrir en un navegador web. Para encontrar la ubicación del sitio web, tienes que leer la línea de código que emite el programa cuando empiezas a ejecutarlo. Cuando ejecuto el programa, la salida tiene este aspecto:

\* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)

A partir de esta línea, puedo saber que si introduzco *http://127.0.0.1:5000/* en un navegador web me llevará al sitio web de Flask que acabo de empezar a ejecutar. Ahora, cuando abro el navegador y entro en este sitio, puedo ver mi nombre en la página[(Figura 11-13](ch11.xhtml#ch11fig13)).

**NOTA**

*El sitio web que creas en este programa sólo está disponible en tu ordenador en este momento. Sólo tú puedes acceder a este sitio web, nadie más en Internet puede verlo.*



*Figura 11-13: ¡Mira mi sitio web! Puedes cambiar lo que se muestra aquí por lo que quieras. ¿Qué escribirás en tu sitio web?*

Para detener el programa, entra en IDLE y pulsa CTRL-C o haz clic en **Shell** image **Reiniciar Shell**.

**NOTA**

*Este capítulo sólo cubre una introducción muy básica a* Flask. Es *muy útil aprender Flask porque te permite construir sitios web interactivos rápidamente con Python. Si quieres saber más sobre* Flask*, consulta este tutorial en el* *sitio web* de Flask *en* [http://flask.pocoo.org/docs/0.10/tutorial/.](http://flask.pocoo.org/docs/0.10/tutorial/)

#### **Misión nº 67: Posicionar sitio web**

Una de las mejores cualidades de Python es que es fácil integrar las características de diferentes módulos en un único programa. Has estado utilizando el módulo API Python de Minecraft durante todo el libro y acabas de conocer el módulo Flask. Con unos pocos pasos puedes integrar ambos.

En esta misión, combinarás la API Python de Minecraft con Flask para mostrar la posición de un jugador en una página web.

Crea un nuevo archivo en IDLE y guárdalo como *positionPage.py* en la carpeta de *archivos*. Tienes que obtener la posición del jugador desde Minecraft y mostrarla en la página web utilizando una función con la etiqueta @app.route("/") de Flask. Puedes basar el código en el ejemplo del [Listado 11-5](ch11.xhtml#ch11ex5). Haz que el código muestre la posición en el formato "x 10, y 110, z 12".

Ejecuta el programa y comprueba tu página web. Muy chulo, ¿eh? Con Flask puedes crear páginas web con todo tipo de información. ¡Incluso puedes subir estas páginas a Internet para compartirlas con tus amigos!

[anterior](ch11_5.html)[Subtema 6 de 7: (Ver todo)](ch11.html)[siguiente](ch11_7.html)