Aprende a programar con Minecraft

### **Un bucle while sencillo**

Los bucles while se utilizan para repetir bloques de código. Similar a las sentencias if, un bucle while ejecutará el código que contiene mientras una condición sea Verdadera. Es decir, debe cumplirse una condición para que se ejecute el cuerpo de la sentencia.

La diferencia entre un bucle while y una sentencia if es que el código de la sentencia if sólo se ejecuta una vez como máximo, mientras que el código del bucle while puede repetirse muchas veces. Los programadores llaman *iteración* a la repetición del código *.* Cuando un bucle se repite, se dice que *itera*.

Por ejemplo, este código utiliza un bucle while para imprimir los números del 1 al 5:

count = 1  
while count <= 5:  
print(count)  
count += 1  
print("Loop finished")

La variable count registra el número de veces que se ha repetido el bucle. Comienza con el valor 1. La condición del bucle while comprueba si la cuenta es menor o igual que 5.

**NOTA**

*En* el [*Capítulo 3*](ch03.xhtml#ch03) *aprendiste que* += *es un operador abreviado. Podrías utilizar el operador estándar de suma* cuenta = cuenta + 1 *para hacer* lo mismo.

La primera vez que se ejecuta el bucle, el valor de cuenta es 1, que es menor que 5. La condición del bucle es Verdadero, y el cuerpo del bucle se ejecuta. A continuación, el programa imprime el valor de la cuenta en el intérprete de comandos de Python, y luego añade 1 al valor de la cuenta. El bucle while vuelve a empezar y comprueba de nuevo la condición, pasando por cada paso hasta que la variable count sea mayor que 5.

Fuera del bucle hay una última línea, que imprime "Bucle finalizado".

Guarda este programa y ejecútalo; deberías ver la siguiente salida:

1  
2  
3  
4  
5  
Loop finished

Prueba a experimentar un poco con el código. Cambia las condiciones para que aparezcan más de 5 números o cambia la cantidad en que aumenta la variable de recuento. Aquí tienes un repaso de cómo funciona el código. La sentencia while sigue estos pasos

1. Comprueba si la condición es Verdadera.
2. Si la condición es Verdadera:
   1. Ejecuta el cuerpo del código.
   2. Repite el paso 1.
3. Si la condición es Falsa:
   1. Ignora el cuerpo de código.
4. Continúa en la línea siguiente al bloque del bucle while.

¡Intentemos utilizar un bucle while en Minecraft para teletransportarnos a un montón de sitios nuevos!

#### **Misión nº 33: Un viaje de teletransporte aleatorio**

En la Misión nº 3[(página 44](ch02.xhtml#page_44)), teletransportaste al jugador a distintas posiciones del juego. Vamos a reescribir ese programa utilizando un bucle while para que puedas repetir el teletransporte una y otra vez.

Mediante un bucle de código que teletransportará al jugador a un lugar aleatorio, puedes hacer que el programa sea más potente *y* mucho más fácil de leer. Mola, ¿eh?

El siguiente código teletransportará al jugador a un lugar aleatorio una vez, eligiendo valores aleatorios en el mundo del juego para las variables x, y y z. Luego fijará la posición del jugador utilizando esas variables.

import random  
from mcpi.minecraft import Minecraft  
mc = Minecraft.create()  
  
➊ # Add the count variable here  
➋ # Start the while loop here  
➌ x = random.randint(-127, 127) # Indent the code from this line  
y = random.randint(0, 64)  
z = random.randint(-127, 127)  
  
mc.player.setTilePos(x, y, z)  
➍ # Add 1 to the value of the count variable here

Ahora mismo, sin embargo, el código sólo teletransportará al jugador una vez. Aunque eso está muy bien, puedes hacerlo totalmente genial. Escribamos un bucle para que el código se repita cinco veces, convirtiéndolo en todo un viaje relámpago.

Para cambiar el código y utilizar un bucle, sigue estos cuatro pasos:

1. Crea una variable de recuento para controlar el bucle ➊.
2. Añade un bucle while con una condición basada en el recuento ➋.
3. Sangrar el cuerpo de la sentencia while ➌.
4. Incrementa el valor de count con cada bucle ➍.

La finalidad de la variable count y del incremento de count es llevar la cuenta del número de veces que se ha repetido el bucle. Hablaré más sobre ellos en la siguiente sección. Por ahora, todo lo que necesitas saber es que count nos permite controlar cuántas veces se repite este código.

[El listado 7-1](ch07.xhtml#ch7ex1) muestra el código con los cambios añadidos.

*randomTeleport.py*

import random  
from mcpi.minecraft import Minecraft  
mc = Minecraft.create()  
  
count = 0  
while count < 5:  
x = random.randint(-127, 127)  
y = random.randint(0, 64)  
z = random.randint(-127, 127)  
mc.player.setTilePos(x, y, z)  
count += 1

*Listado 7-1: Código para teletransportar aleatoriamente al jugador por el mundo del juego*

Copia [el Listado 7-1](ch07.xhtml#ch7ex1) en un nuevo archivo, guárdalo como *randomTeleport.py* en una nueva carpeta llamada *whileLoops*, y ejecuta el código. Deberías ver cómo el jugador se desplaza por el mundo de Minecraft. ¡Pero el código se ejecuta demasiado rápido! Todo el viaje termina en menos de un segundo. Vamos a arreglarlo juntos.

Utilizarás el módulo de tiempo para ralentizar el código. Sigue estos pasos:

1. En la primera línea del programa, añade la sentencia import time. Esto importa el módulo tiempo de Python, que contiene un conjunto de prácticas funciones relacionadas con el tiempo y otras cosas.
2. Añade la línea time.sleep(10) al final del cuerpo de tu bucle while para añadir un retardo de 10 segundos a tu programa. ¡Asegúrate de sangrar esta nueva línea final de tu programa para que quede dentro del bucle while!

Guarda el programa y ejecútalo. Ahora el jugador debería teletransportarse a un nuevo lugar aleatorio cada 10 segundos. [La Figura 7-1](ch07.xhtml#ch7fig1) muestra mi programa en ejecución.



*Figura 7-1: Cada 10 segundos, el programa me teletransporta a un nuevo lugar.*

**OBJETIVO EXTRA: DUERME TRANQUILO**

De momento, el programa esperará 10 segundos al final de cada bucle. ¿Qué ocurre si mueves la sentencia time.sleep(10) al inicio del bucle?

[anterior](ch07_1.html)[Subtema 2 de 6: (Ver todo)](ch07.html)[siguiente](ch07_3.html)