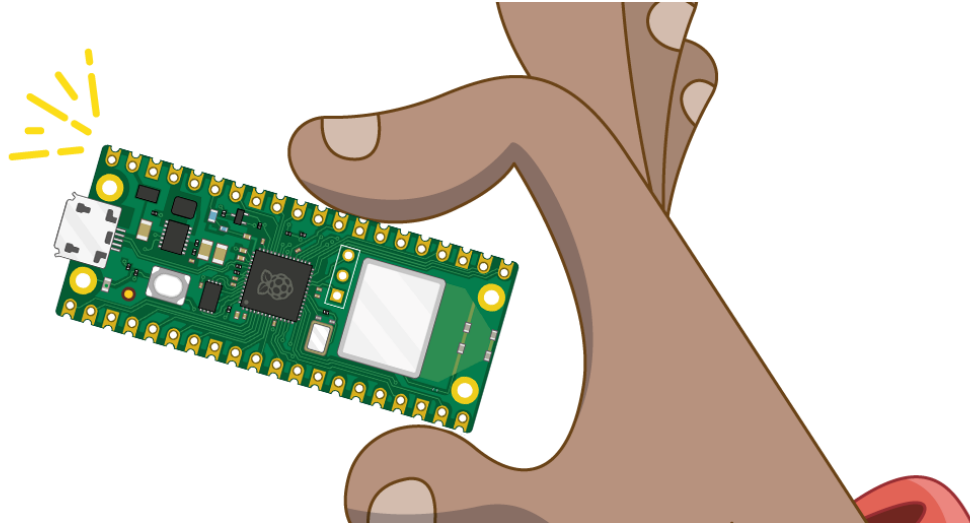


Primeros pasos con su Raspberry Pi Pico W

Frambuesa Pi Pico Pitón



Contents

¡Guarda tu progreso!

Si desea volver a este proyecto más tarde, puede crear una cuenta de Raspberry Pi para guardar su progreso hasta el momento. En tu cuenta también verás todos los proyectos que completes.

Iniciar sesión o registrarse

abrir un zócalo

En este paso, utilizará la conexión a su WLAN para abrir un socket.

```
>>> %Run -c $EDITOR_CONTENT
Waiting for connection...
Waiting for connection...
Waiting for connection...
Connected on 192.168.1.143
<socket state=1 timeout=-1 incoming=0 off=0>
```

Un **socket** es la forma en que un **servidor** puede escuchar a un **cliente** que quiere conectarse a él. La página web que está viendo actualmente está alojada en los servidores de la Fundación Raspberry Pi. Estos servidores tienen un socket abierto que espera a que su navegador web establezca una conexión, momento en el que el contenido de la página web

se envía a su computadora. En este caso, su servidor será su Raspberry Pi Pico W y el cliente será un navegador web en otra computadora.

Para abrir un socket, debe proporcionar la dirección IP y un número de puerto. Las computadoras utilizan los números de puerto para identificar dónde deben enviarse las solicitudes. Por ejemplo, el puerto 80 se usa normalmente para el tráfico web; Stardew Valley usa el puerto 24642 cuando juegas un juego multijugador. Como está configurando un servidor web, utilizará el puerto 80.



Cree una nueva función a la que se pueda llamar para abrir un socket. Debe estar por encima de su `try/ except`. Comience dando al socket una dirección IP y un número de puerto.

servidor_web.py

```
def open_socket(ip):  
    # Open a socket  
    address = (ip, 80)  
  
    try:  
        connect()  
    except KeyboardInterrupt:  
        machine.reset()
```



Ahora cree su socket y luego haga que escuche las solicitudes en el puerto 80. No olvide llamar a su función en la parte inferior de su código.

servidor_web.py

```
def open_socket(ip):  
    # Open a socket  
    address = (ip, 80)  
    connection = socket.socket()  
    connection.bind(address)  
    connection.listen(1)  
    print(connection)  
  
try:  
    ip = connect()  
    open_socket(ip)  
except KeyboardInterrupt:  
    machine.reset()
```



Prueba: ejecute su código y debería ver un resultado similar a este.

```
>>> %Run -c $EDITOR_CONTENT
Waiting for connection...
Waiting for connection...
Waiting for connection...
Waiting for connection...
Waiting for connection...
Connected on 192.168.1.143
<socket state=1 timeout=-1 incoming=0 off=0>
socket state=1le dice que su socket está funcionando.
□
```

Por último, reemplace your `print` con a `return` luego almacene la conexión de socket devuelta como una variable.

servidor_web.py

```
def open_socket(ip):
    # Open a socket
    address = (ip, 80)
    connection = socket.socket()
    connection.bind(address)
    connection.listen(1)
    return connection

try:
    ip = connect()
    connection = open_socket(ip)
except KeyboardInterrupt:
    machine.reset()
```

Ahora tiene su Raspberry Pi Pico W escuchando conexiones a su dirección IP en el puerto **80**. Esto significa que está listo para comenzar a servir código HTML, de modo que un navegador web conectado pueda ver una página web.

Guarda tu proyecto



Crear una página web

¿Detectó un error? ¿Disfrutando del proyecto? ¿Alguna opinión sobre la web? ¡Háznos saber!

Enviar comentarios

Publicado por [Raspberry Pi Foundation](#) bajo una [licencia Creative Commons](#) . [Ver proyecto y licencia en GitHub](#)

[Accesibilidad](#)

[Política de cookies](#)

[Política de privacidad](#)