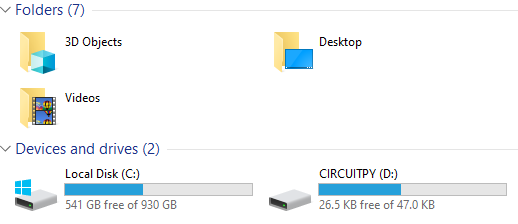
**Uso de CircuitPython con el Feather M0**

Una vez instalado CircuitPython como firmware para el Feather M0 RFM95 se puede comenzar a programar algunos scripts de Python para aprovechar las funciones que brinda este microcontrolador.

Se recomienda instalar Mu Editor, es mu fácil de utilizar y proporciona una interfaz de edición de código, consola serial, graficador serial y REPL.

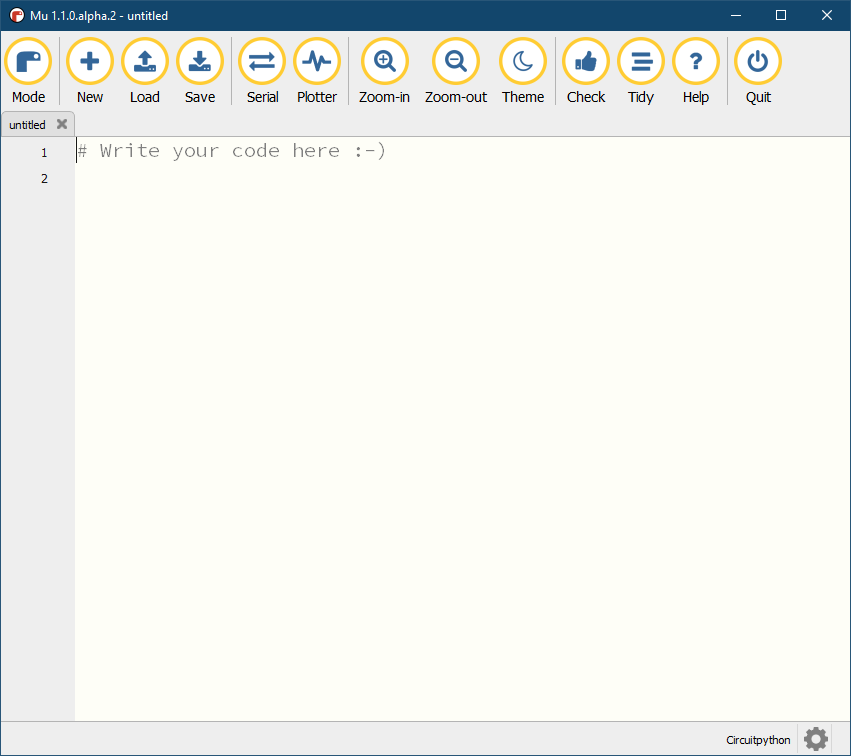
**Pasos de uso:**

1. Conectar el Feather M0 mediante el puerto serial a la computadora. Utilice cualquier cable micro USB-USB estándar (con línea de datos).



En Windows 10, se visualiza el Feather como una unidad de almacenamiento externo, llamada CIRCUITPY. El nombre se puede cambiar con solo tocar el nombre de la unidad o en propiedades. Si el Feather está en modo “boot” aparecerá con otro nombre.

1. Abrir Mu Editor, inmediatamente se muestra en la parte inferior que un nuevo dispositivo compatible con CircuitPython ha sido detectado, de lo contrario se mostrará un aviso de que no se ha detectado ningún dispositivo.

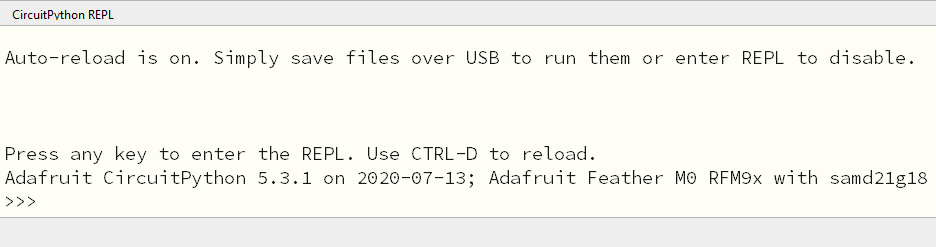


1. Al seleccionar la opción “Serial” en la parte izquierda de la barra de herramientas del editor Mu se puede acceder a Consola Serial y REPL.

REPL o Read-Evaluate-Print-Loop es la otra característica de conexión en serie. REPL permite ingresar líneas individuales de código y hacer que se ejecuten inmediatamente. Es muy útil si hay problemas con un programa en particular y no se sabe el por qué. Es interactivo, por lo que es ideal para probar con varios comandos.

1. Para utilizar REPL primero debe estar conectado a la consola serie. Una vez que se haya establecido esa conexión, se debe presionar Ctrl + C.

Si no hay un código corriendo, se entrará directo a REPL al presionar cualquier tecla. Lo primero que se verá es la información de la placa (Feather M0)



Si hay un código ejecutándose al momento de querer ingresar a REPL el proceso se interrumpirá. Se mostrará KeyboardInterrupt al presionar Ctrl + C.

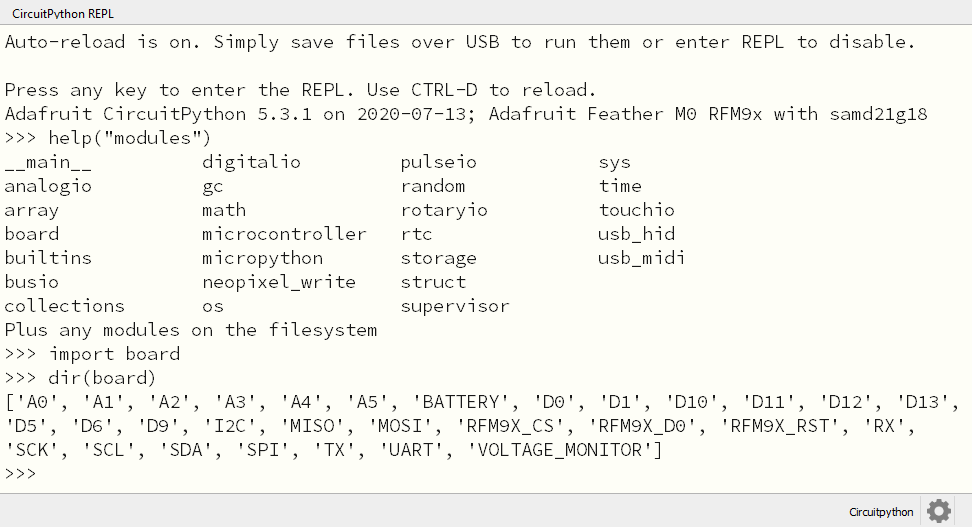


1. Para salir del REPL y regresar a la consola serial, simplemente se debe presionar Ctrl + D. Esto recargará la placa y volverá a ingresar a la consola serial. Se reiniciará el programa que estaba en ejecución antes de ingresar al REPL (si es que lo había). En la ventana de la consola, se verá cualquier resultado del programa que se estaba ejecutando.

REPL es como interactuar con Python Shell. Al escribir import board dentro de REPL y dar ENTER, la declaración de import simplemente le dice al código que espere hacer algo con ese módulo. En este caso, le está diciendo al REPL que planea hacer algo con ese módulo. Se despliega una lista de todos los pines de la placa que están disponibles para usarlos en el código.

Para revisar cuáles librerías están instaladas en el core de CircuitPython se debe utilizar el siguiente comando en REPL: help("modules")

Esto es importante para evitar algún error de importación en futuros scripts.



DEBIDO A LA POCA MEMORIA DEL FEATHER M0 se recomienda instalar solo las librerías necesarias y no todo el bundle de CircuitPython que hay disponible. Estas librerías se instalan como carpetas o archivos que tiene extensión .mpy

1. Copie el siguiente código en el editor Mu. Este código hacer parpadear el LED interno (Pin #13) del Feather, a la vez que imprime un “Hello World”.

import board

import digitalio

import time

led = digitalio.DigitalInOut(board.D13)

led.direction = digitalio.Direction.OUTPUT

while True:

    led.value = True

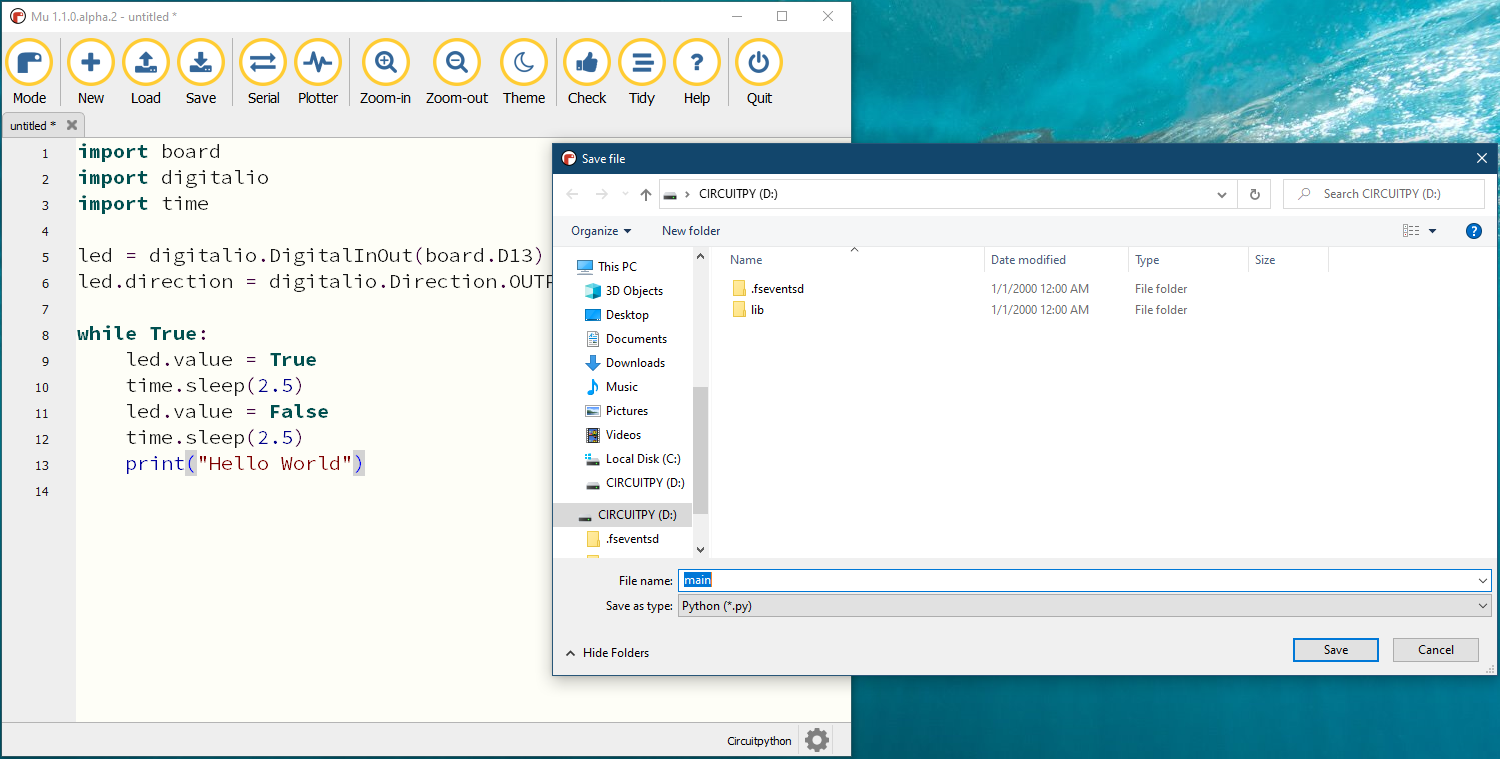
    time.sleep(2.5)

    led.value = False

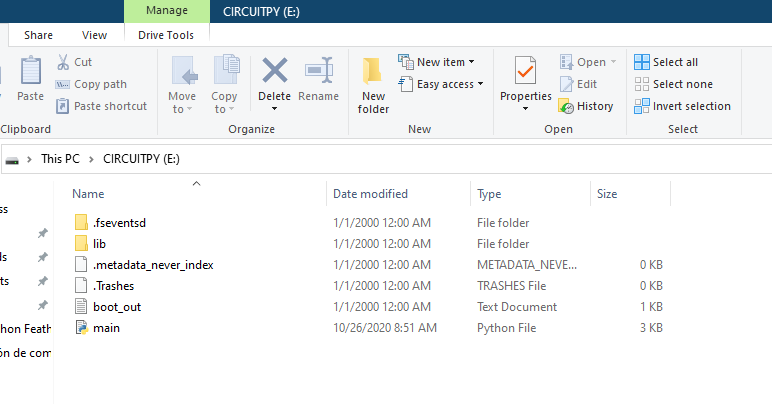
    time.sleep(2.5)

    print("Hello World")

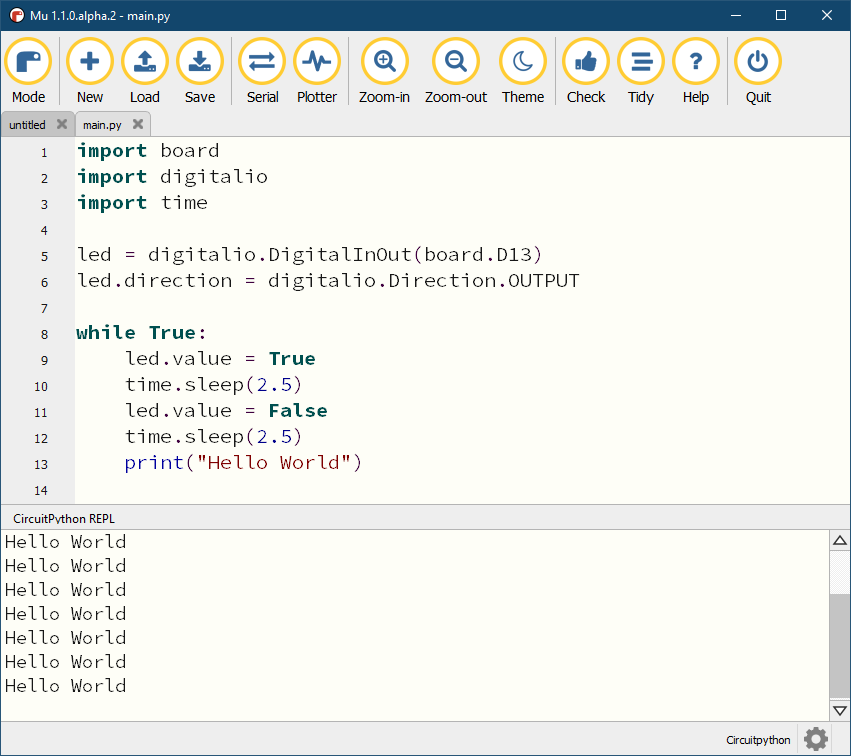
1. Guarde el código en la unidad de almacenamiento correspondiente al Feather M0 (con nombre CIRCUITPY). Para ello haga click en Save, guarde el código con el nombre **main.py** o **code.py** (es necesario!!!).



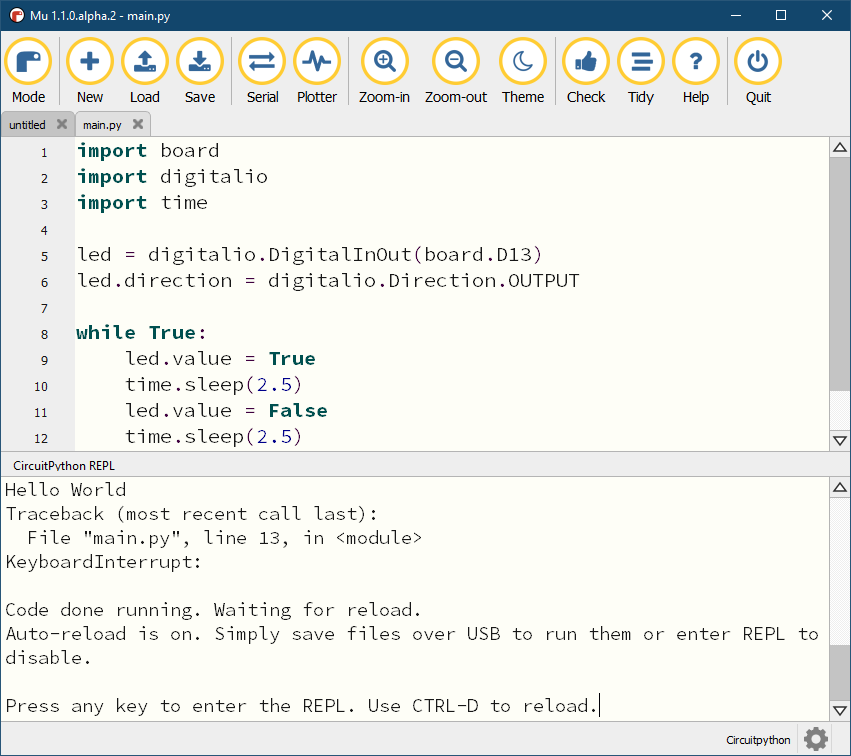
Al guardarse el código, el directorio CIRCUITPY se debería ver de la siguiente manera:



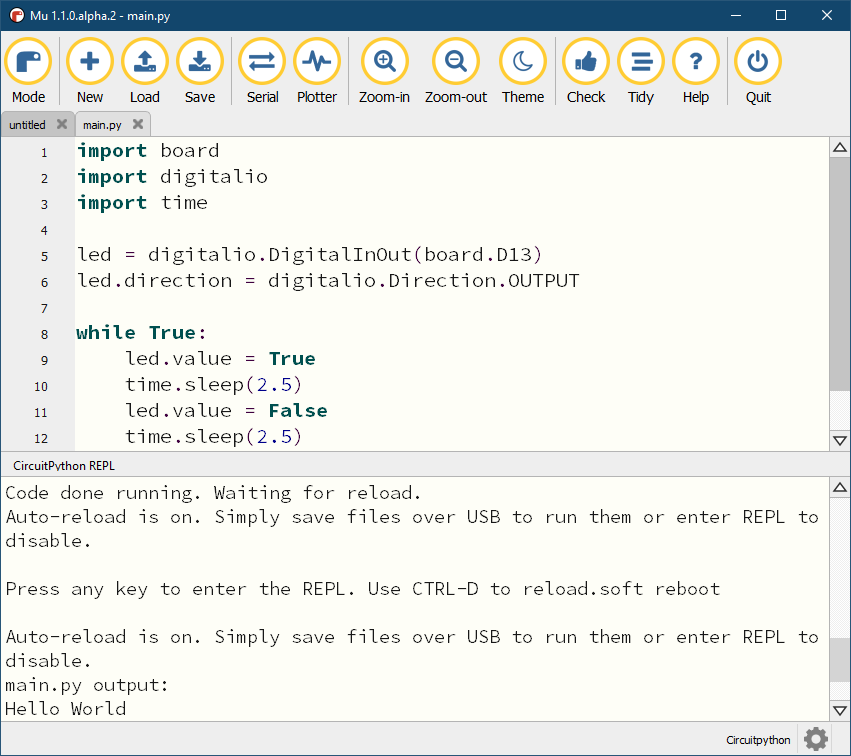
1. Al guardar el código con el nombre main.py, se debería comenzar a ejecutar automáticamente el código. Si ocurre algún error de importación de alguna librería o no está incluida dentro de CircuitPython, algún error de sintaxis, etc, entonces al abrir la consola serial se desplegarán los errores correspondientes.



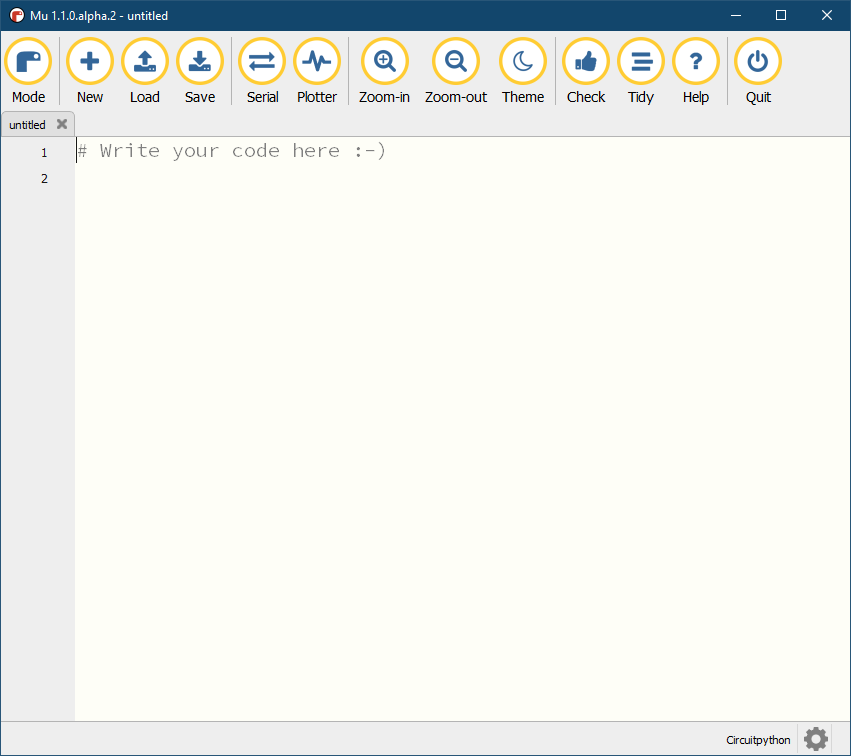
* Note que al teclear Ctrl + C en la consola serial se activa REPL, por ende se interrumpe la ejecución del código actual, tal y como se indicó en el paso **4**.

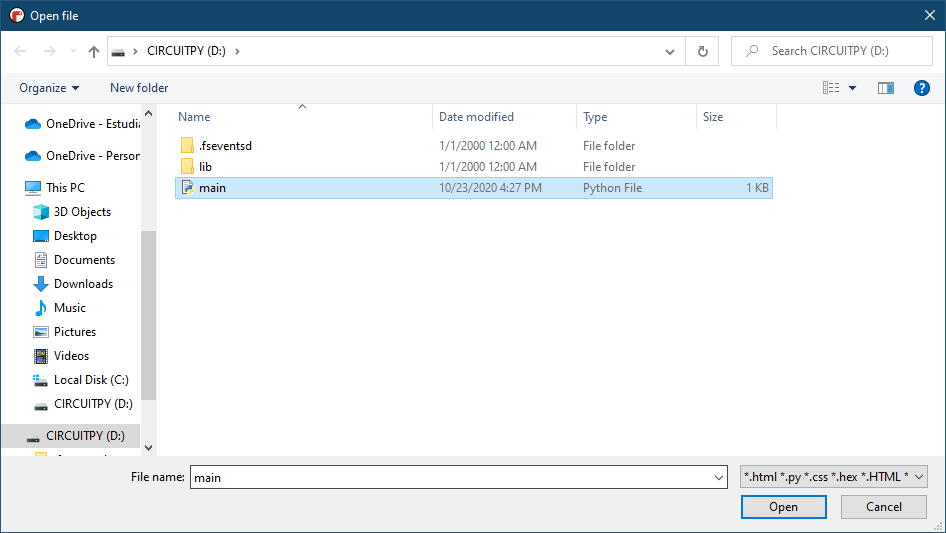


* Note que al teclear Ctrl + D se auto-ejecutará el código de nuevo, se saldrá de REPL y volverá a la consola serial, tal y como se indicó en el paso **5**.



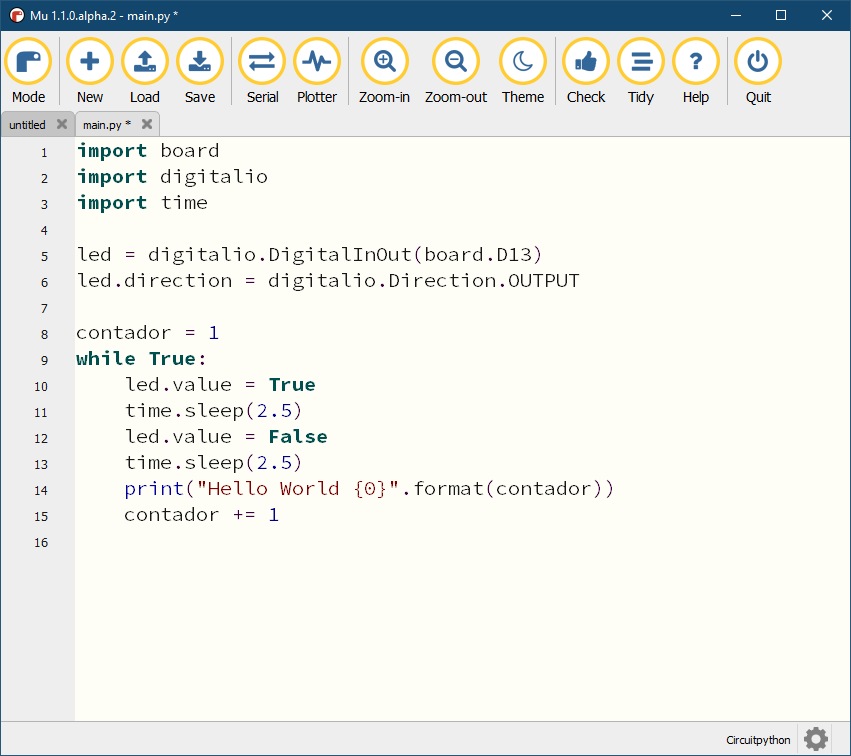
1. Suponiendo que no se conoce el contenido del código que está ejecutando el Feather actualmente, dentro del editor Mu se puede cargar el código guardado en el directorio CIRCUITPY del Feather M0. Para ello, dar click en “Load” para abrir el código de Python que se quiere consultar.



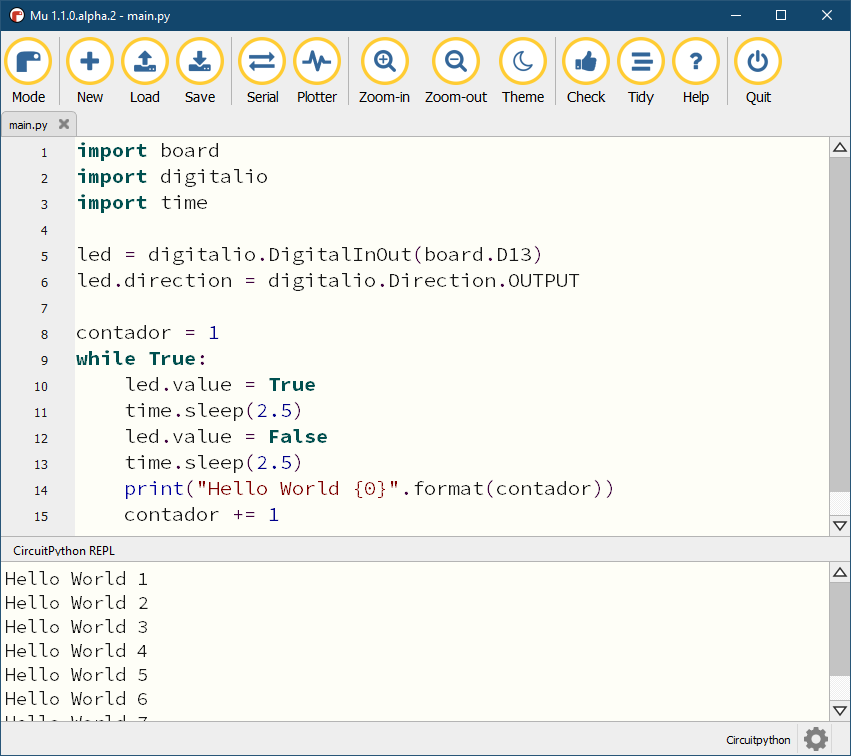


En este caso es el mismo código que se mostró al inicio.

1. Supongamos que la subrutina time.sleep(2.5) se reedefine como time.sleep(5), al hacer click en “Save” inmediatamente se comenzará a ejecutar el código en el microcontrolador del Feather M0. Si el código hace uso de la consola serial (por ejemplo mediante un print()) entonces haga click en “Serial” para ver la respuesta del código en ejecución o errores que pueda retornar el código. NOTA: a veces la consola serial tarda en responder.



1. Si al hacer click en Save o mover un script de código desde otro editor (archivo .py) al directorio CIRCUITPY (o el nombre que corresponda si se lo cambió) sucede que el Feather no responde o no ejecuta el código, luego de un tiempo simplemente desconéctelo de la computadora y vuélvalo a conectar. Tras esto debería ejecutar el código, siempre es bueno revisar la consola serial.



Los cambios en el código se ejecutan tan pronto como se termina de guardar el archivo. CircuitPython dentro del Feather M0 detecta cuando el contenido del código se modifica o escribe y por ende este se reiniciará automáticamente.

Sin embargo, debe esperar hasta que el archivo haya terminado de guardarse antes de desconectar o reiniciar la placa. En Windows con algunos editores puede tardar hasta 90 segundos, en Linux puede tardar 30 segundos en completarse porque el editor de texto no guarda el archivo por completo. Esto es muy importante de tener en cuenta. Si desconecta o reinicia la placa antes de que la computadora termine de escribir el archivo dentro de esta, puede corromper la unidad. Si esto sucede, se puede perder el código, por lo que es importante hacer una copia de seguridad del código en la computadora con regularidad. En caso de haber corrompido la unidad revisar acá: <https://learn.adafruit.com/welcome-to-circuitpython/troubleshooting>

En editores como MU y Visual Studio Code los archivos se guardan inmediatamente. En Windows la unidad (CIRCUITPY) del Feather se debe expulsar de forma segura (como una llave maya), para evitar corromperla.

**Guía basada en:**

<https://learn.adafruit.com/welcome-to-circuitpython/creating-and-editing-code>