

Mu Editor Instruction Manual

Created October 1st, 2019

Artel[®]

Revision History

Date Revised	Revised Contents
2019/10/1	First release
2019/11/20	Plotter information added
2020/04/23	Administration screen information added

Index

1. Getting Started.....	1
2. GUI	1
3. The Administration Screen	9

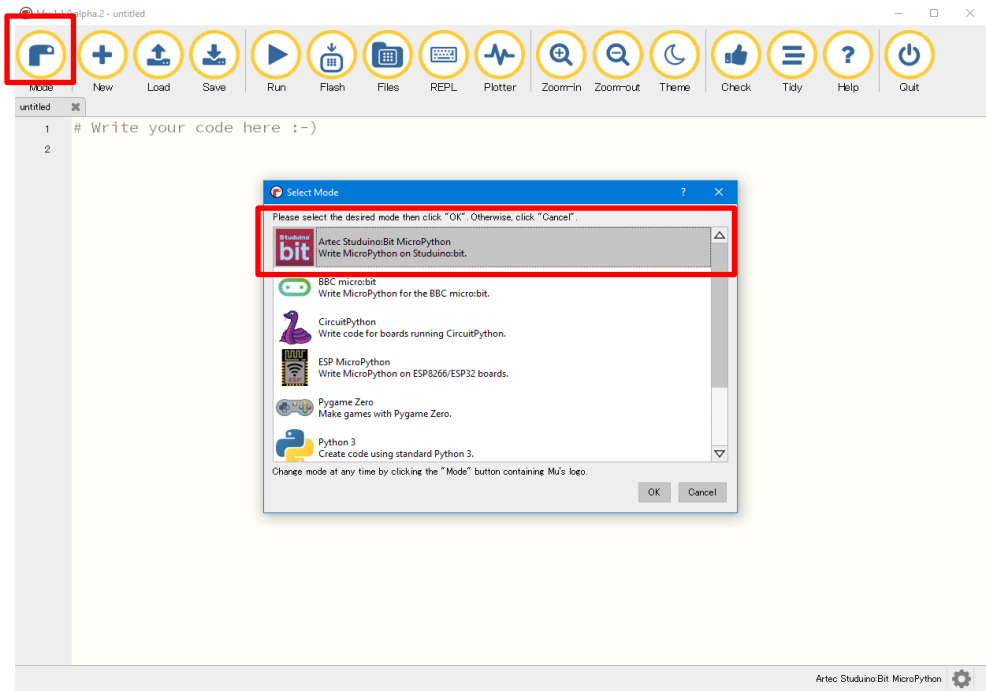
1. Getting Started

You can use Mu Editor to write Python scripts (programs) you can transfer to your ArtecRobo Core Unit. This manual explains how to use Mu Editor.

2. GUI

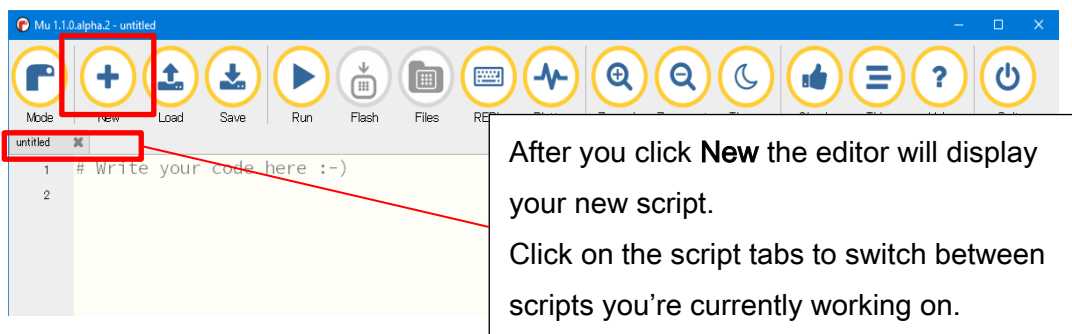
- Mode

Use the **Mode** button to open the **Select Mode** window. Select **Studuino:bit**.



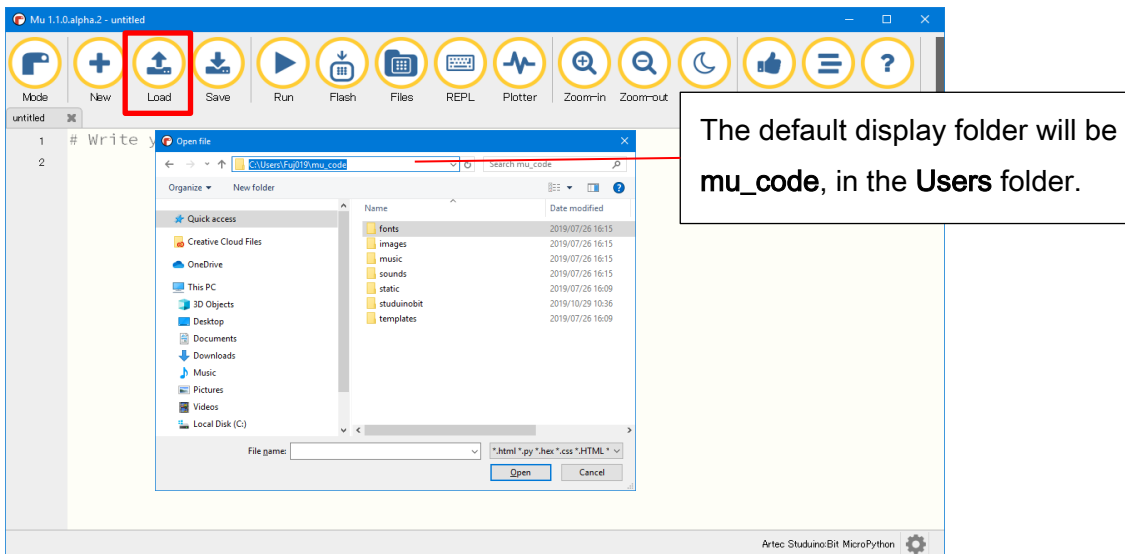
- New

Use the **New** button to create a new script in the editor.



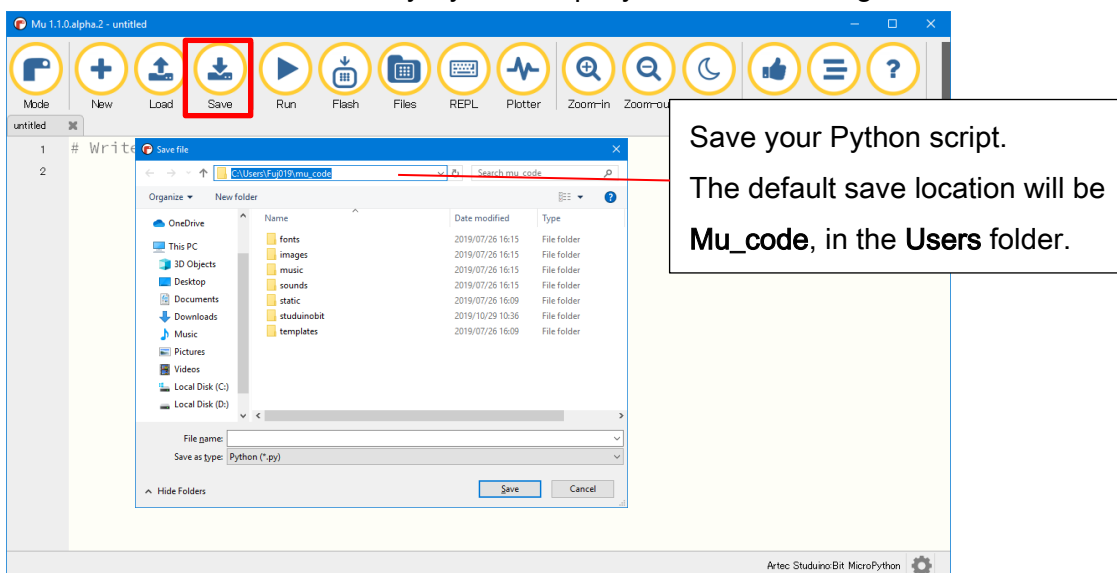
- Load

Use the **Load** button to open any Python scripts you've saved.



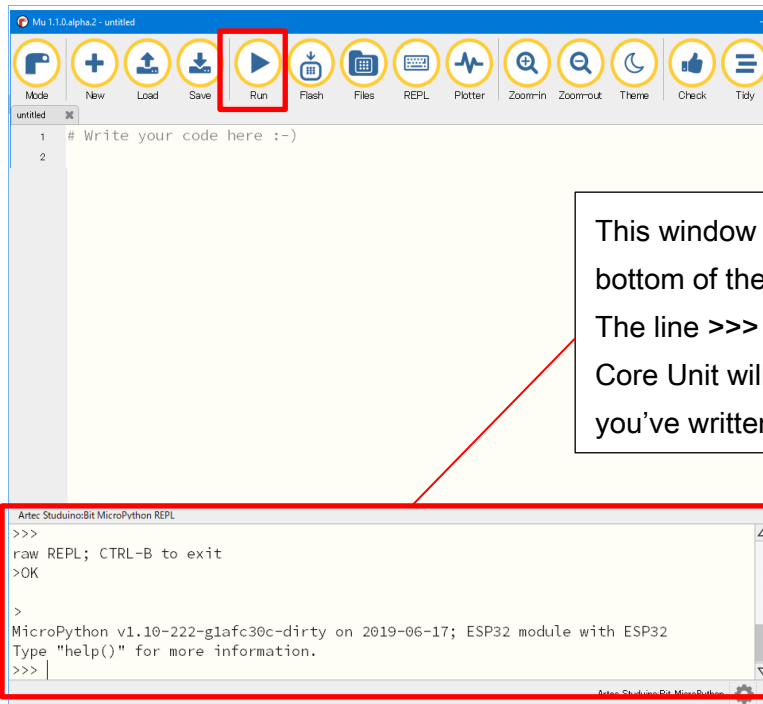
- Save

Use the **Save** button to save any Python scripts you've edited using Mu Editor.



- Run

Use the **Run** button to run the Python **scripts** you've edited using Mu Editor on your ArtecRobo Core Unit.



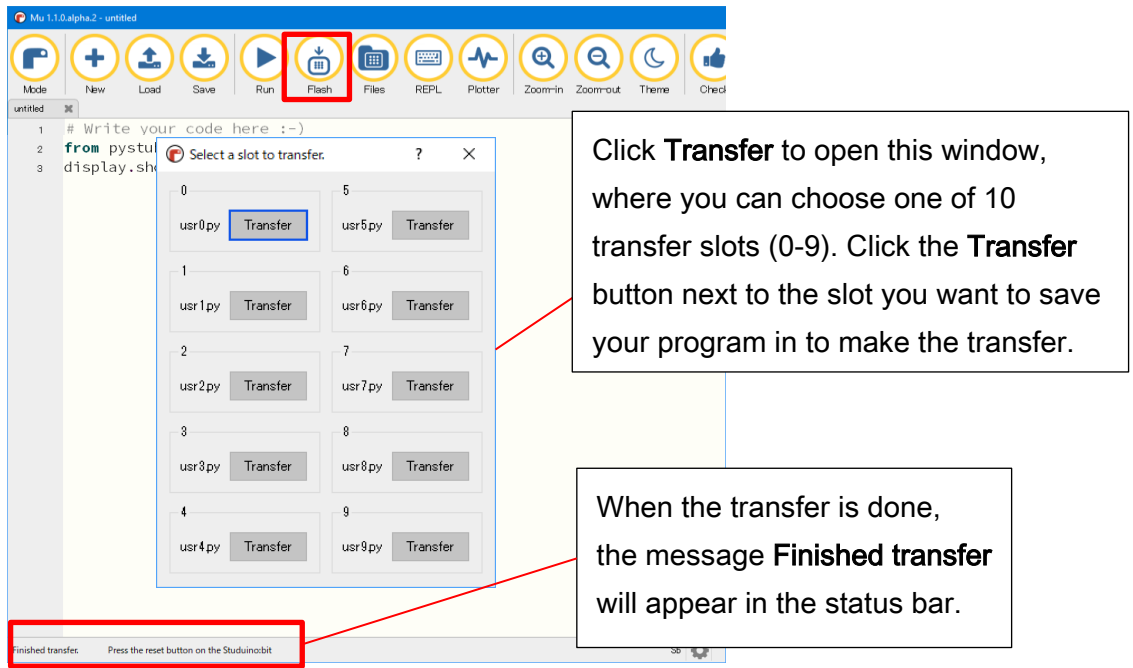
When you click **Run**, a window will open at the bottom of the screen that lets you use REPL. As long as REPL is open, the **Transfer** and **Files** buttons will be grayed out and unavailable. To close the REPL window and make **Transfer** and **Files** usable, click the **REPL** button.

- Transfer

The Core Unit comes with ten sample programs installed on it. If you start up the Core Unit while holding down the A Button, you'll enter Program Select Mode, where you can pick one of the sample programs (numbered 0-9) to run. For more information on the sample programs and how to run them, see sections 3. **Starting the Core Unit** and 4. **Running the Sample Programs** in the manual linked below.

https://www.artec-kk.co.jp/artecrobo2/pdf/en/82552_man_K0419_E.pdf

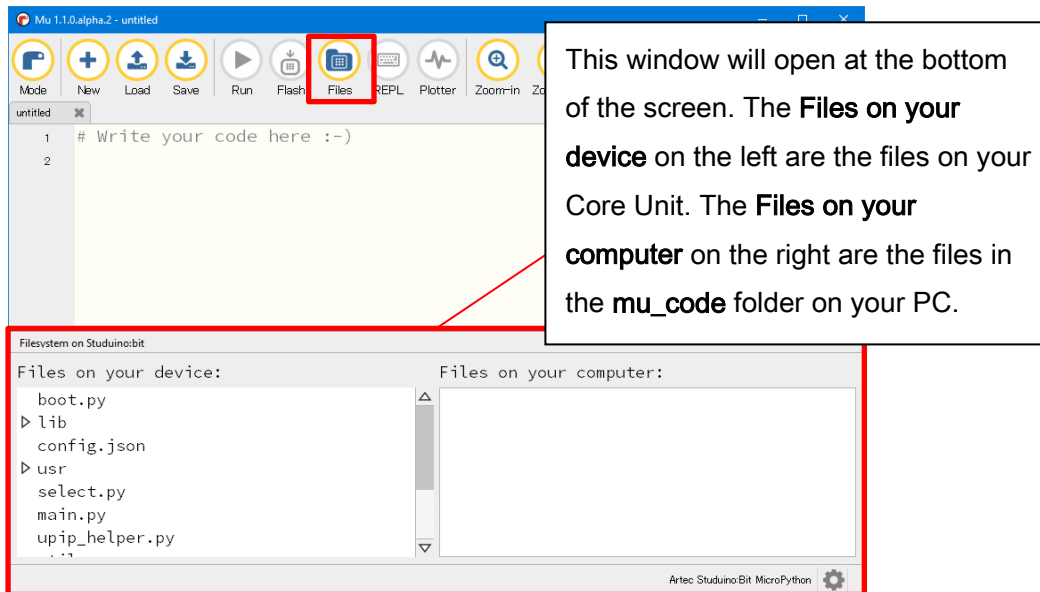
You can transfer Python scripts you write in Mu Editor to the same slots on the Core Unit where the sample programs are saved. Click the **Transfer** button, and the transfer slot selection window will appear.



You can run your transferred program using the Core Unit's Program Select Mode.

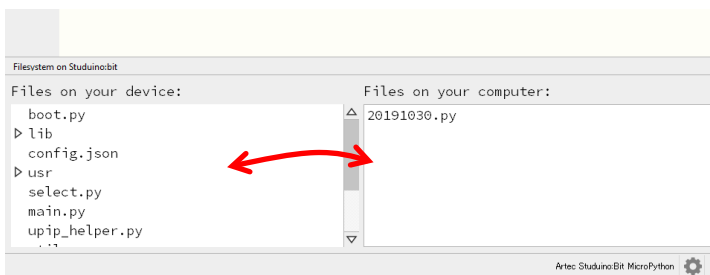
- Files

Use the **Files** button to view the files on your Core Unit and your PC while you're editing.

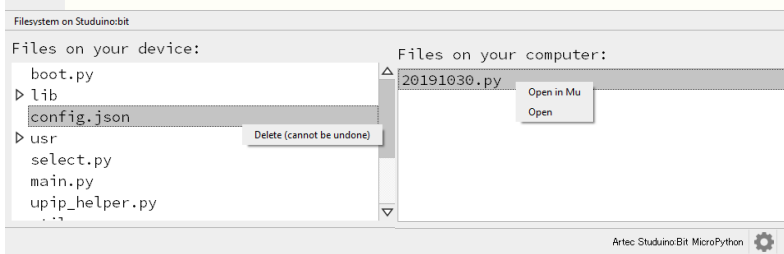


When you click **Files**, a window will open at the bottom of the screen. As long as this window is open, the **Run**, **Transfer**, **REPL** and **Plotter** buttons will be grayed out and unavailable. To close this window and make these other buttons usable, click the **Files** button.

Drag and drop files between the **Files on your device** and **Files on your computer** windows to move files between your Core Unit and PC.

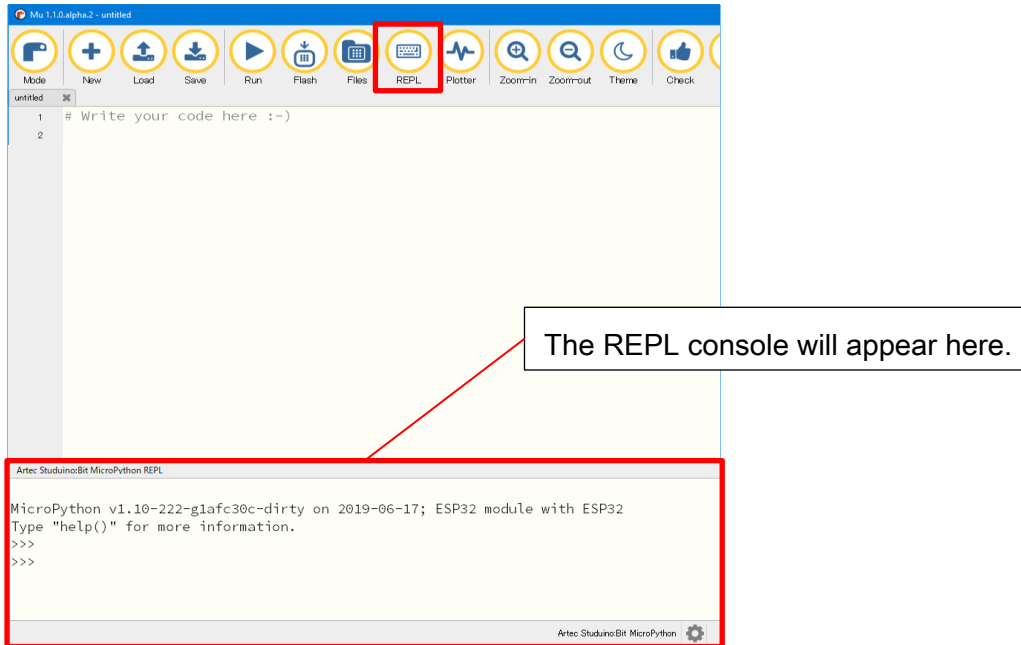


Right clicking a file from the **Files on your device** window lets you delete it. Right clicking a file from the **Files on your computer** window lets you open it in Mu Editor. To delete files from the **Files on your computer** window, access them directly using your PC's file explorer instead.



● **REPL**

Use the **REPL** button to open a REPL window.



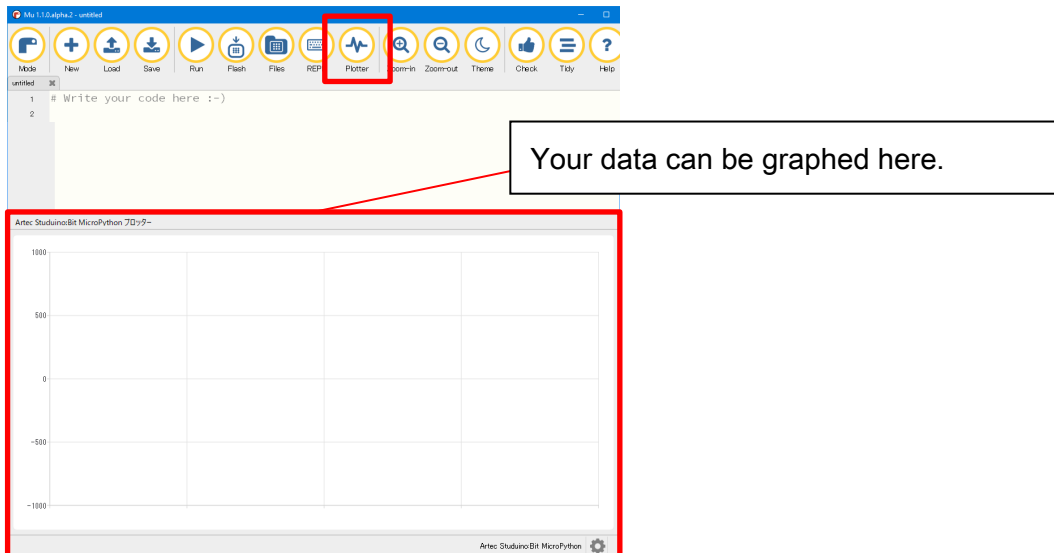
When you click **REPL**, a window will open at the bottom of the screen that lets you use REPL. As long as REPL is open, the **Transfer** and **Files** buttons will be grayed out and unavailable. To close the REPL window and make **Transfer** and **Files** usable, click the **REPL** button.

You can use the REPL console to run each line of your script one-by-one. Type in the program from the example on page 1 after the >>> as shown below, and you'll get an identical result (the word "Hello" will appear on the LED display).



- **Plotter**

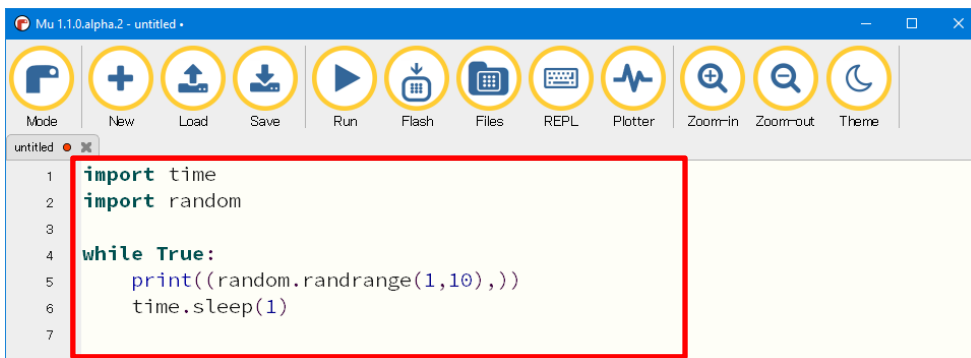
This button opens a graphing screen.



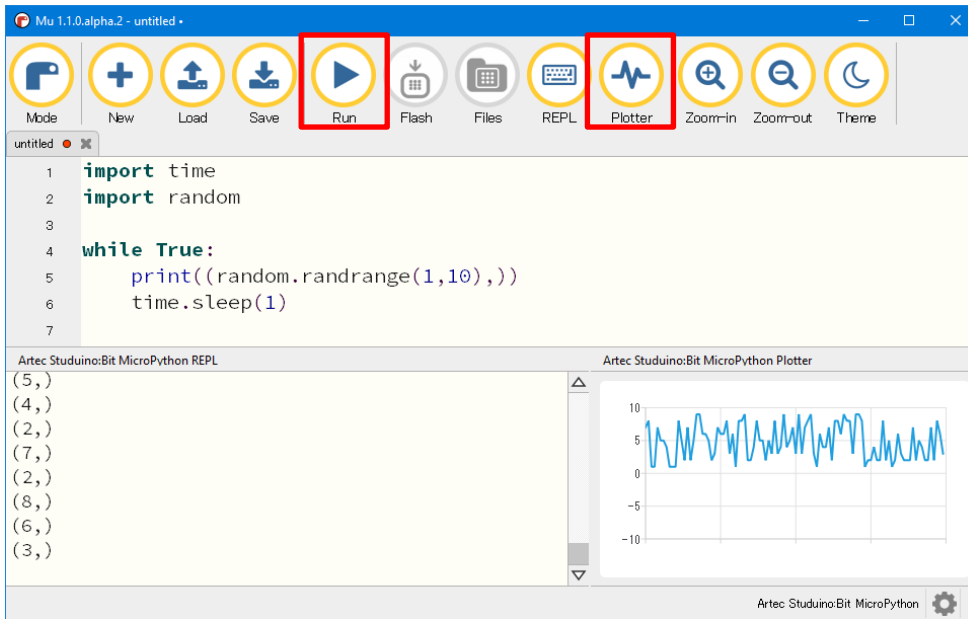
When the graphing screen is open, the **Transfer** and **Files** buttons will be grayed out and unavailable. Clicking **Plotter** again to close the graphing screen window will make **Transfer** and **Files** usable.

The plotter can create graphs of data it receives in a tuple format.

For example, the program shown below picks a random number from 1 to 10 once every second and gives its results as a tuple.

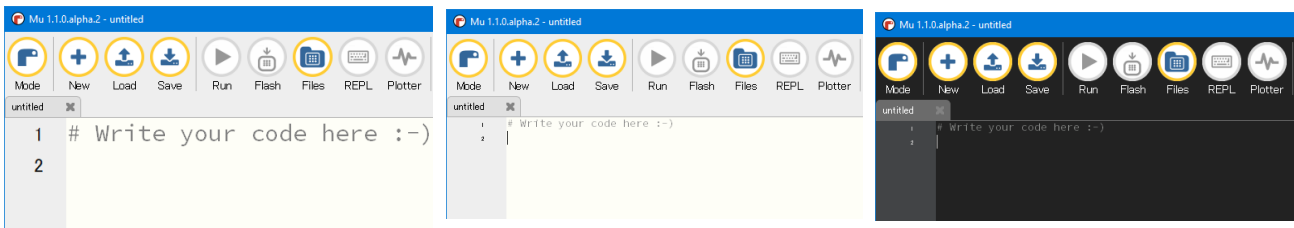


Write this program in the editor, then connect your PC to the Core Unit via USB and press the **Run** button. Once the program is running, press **Plotter** to see the random numbers the program generates each second graphed on the screen.



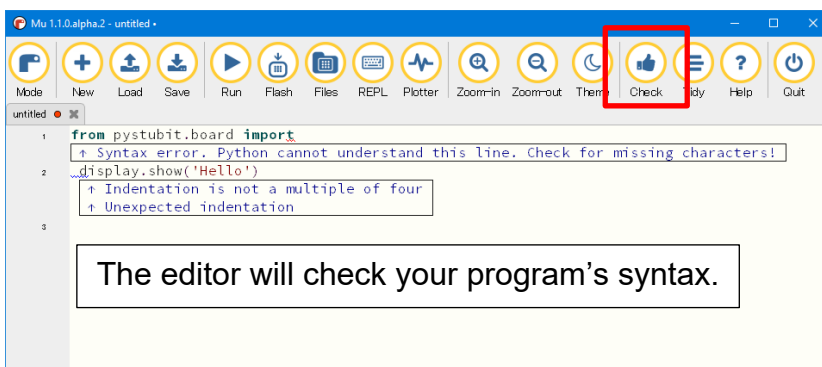
- Zoom-in, Zoom-out, Theme

Use these buttons to change the size of the text or the appearance of the editor.



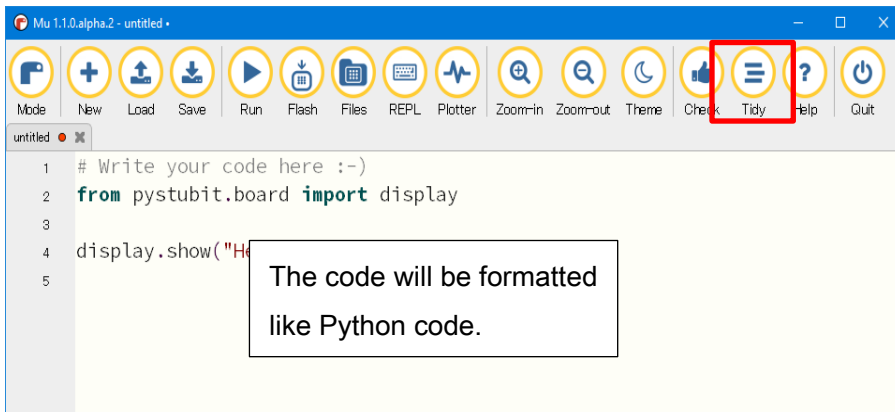
- Check

Use the **Check** button to run a syntax check.



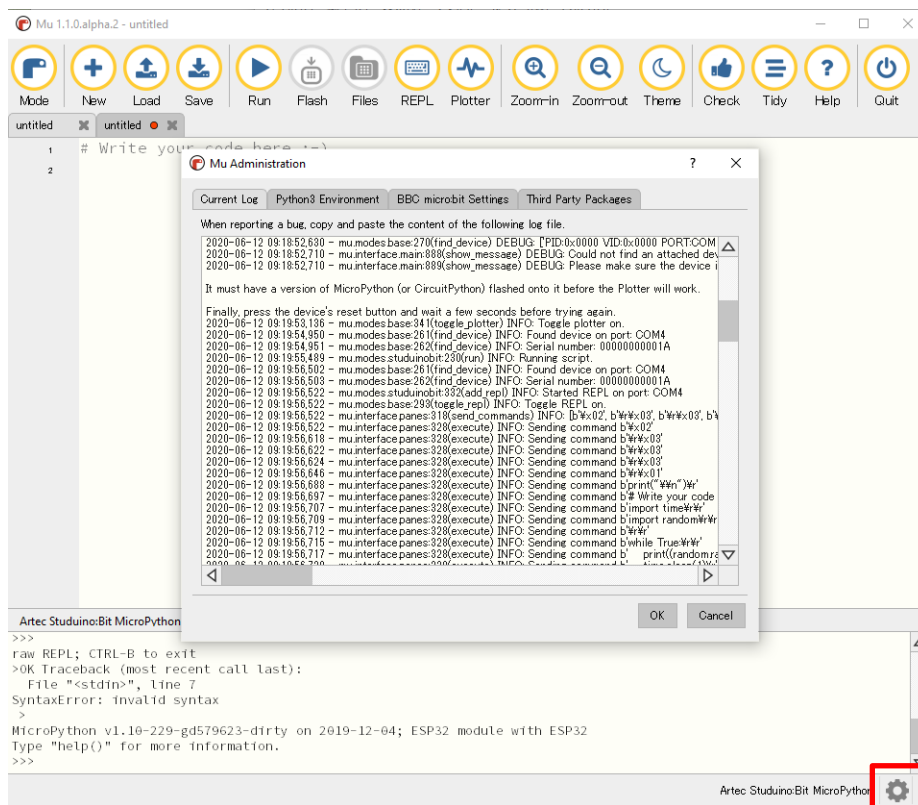
- Tidy

Use the **Tidy** button to change your program code into Python syntax.



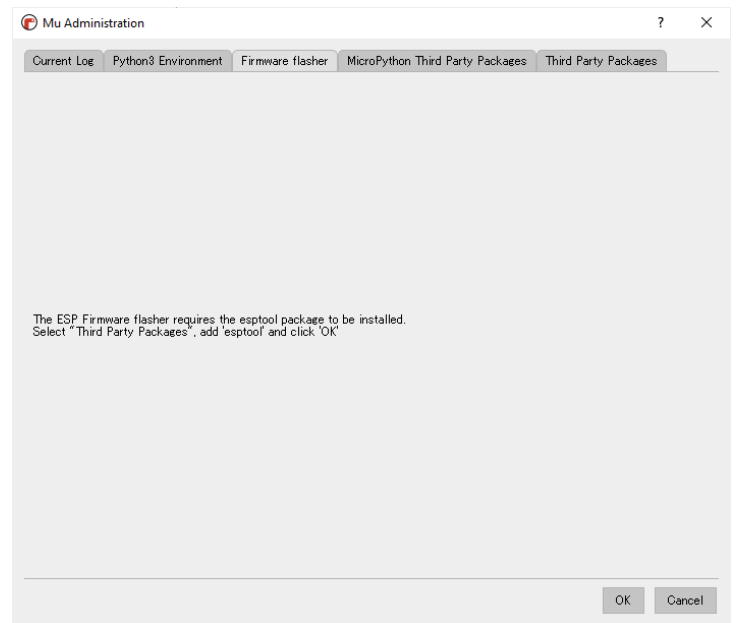
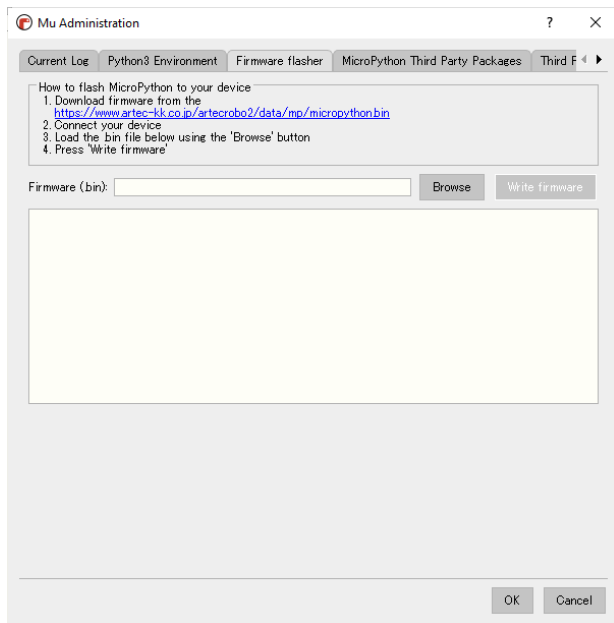
3. The Administration Screen

You can use the administration screen to change the settings for the Mu editor and your Core Unit. Click the Mu Administration button at the bottom right of the window to open the Mu Administration screen. We'll explain the Studuino:bit-specific features (**Firmware flasher** and **MicroPython Third Party Packages**) in this manual. Check out https://codewith.mu/en/howto/1.0/read_logs to learn about the **Current Log** tab, and https://codewith.mu/en/howto/1.0/python3_envars to learn about the **Python3 Environment** tab.



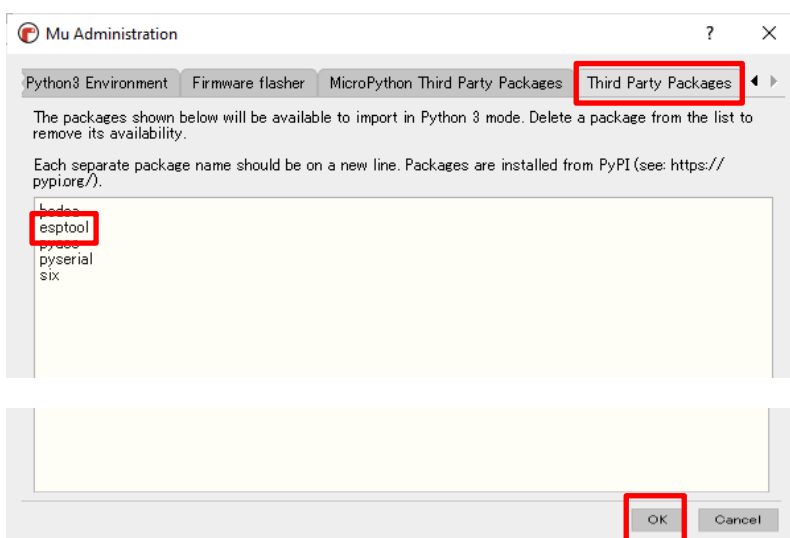
● Firmware Flasher

This feature allows you to update MicroPython. The screen this tab displays will be different depending on your Mu editor's settings. If the screen shown below on the left appears, follow the **How to flash MicroPython to your device** instructions it shows to update MicroPython. If the screen show below on the right appears, follow the instructions below to change the Mu editor's settings.



The **Firmware flasher** requires the **esptool** module. Follow these steps to add the esptool module to the Mu editor.

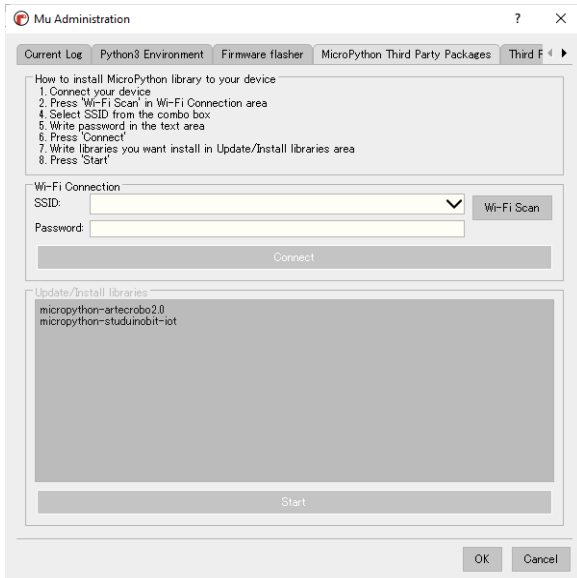
- ① Go to the **Third Party Packages** tab.
- ② Add **esptool**.
- ③ Click **OK**.



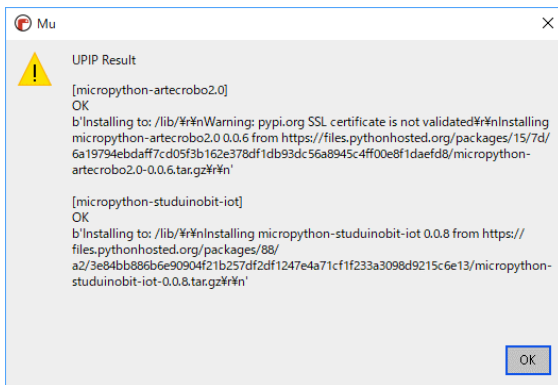
The window will close automatically after you change this setting.

- MicroPython Third Party Packages

This feature lets you install the MicroPython package provided by PyPI* on your Core Unit. Consult the on-screen **How to install MicroPython library to your device** instructions.



This installation uses a network connection, so it may take some time. After installation, the following dialog window will appear on screen. If the installation was successful, the message will include an OK.



*PyPI (the Python Package Index) is a repository used to store packages made in Python. Visit the URL below to view the packages registered there.

<https://pypi.org/>

Using this feature in the editor lets you update the libraries that control ArtecRobo2.0 and IoT functions, which can be found on PyPI.

<https://pypi.org/project/micropython-artecrobo2.0/>

<https://pypi.org/project/micropython-studuino-bit-iot/>

- Third Party Packages

This feature lets you install or uninstall the MicroPython package provided by PyPI in the Mu editor.

Write the name of the package you want install in the editable box (outlined in red below). To uninstall a package, delete its name from the editable area.

When you click the OK button, the package will be installed or uninstalled.

