Cross-Platform CircuitPython coding using DojoGaiden Blocks

Create a friendly Blocks based CircuitPython NeoPixel program once and run it on a Raspberry Pi or an Adafruit Feather M0 Express

Overview

Show how to use the DojoGaiden Blocks IDE to build, deploy and run a NeoPixel program on a Raspberry Pi or Adafruit Feather M0 Express board.

This project uses:

Any Raspberry Pi

An Adafruit Feather M0 Express or other Express board

Any NeoPixel device

A Breadboard

Some jumper wires

A USB data-transfer enabled cable

Requirements needed on the Pi and Feather boards

Adafruit's Blinka (and supporting) Libraries need to be added to the Pi.

The Express board needs to be setup for CircuitPython (I used v. 4.1.0)

The Adafruit NeoPixel library (required on both the Pi and Express board)

A Samba share on a Pi directory mount (Pi Only)

Useful Resources for this Project

Guide for installing CircuitPython on a Raspberry Pi
https://learn.adafruit.com/circuitpython-on-raspberrypi-linux/overview

Setting up a Samba share on the rPi
https://www.raspberrypi.org/forums/viewtopic.php?t=205379

RP NJ CoderDojo DojoGaiden Blocks
http://dojogaiden.rpnjcoderdojo.com/

Github for our Blockly IDE (older version)
https://github.com/RP-NJcoderdojo/rpnj_cd_blockly

Adafruit Feather M0 Express Board
https://learn.adafruit.com/adafruit-feather-m0-express-designed-for-circuit-python-circuitpython/circuitpython-neopixel

Connecting the NeoPixels to the feather

Connect the Data wire to pin 12

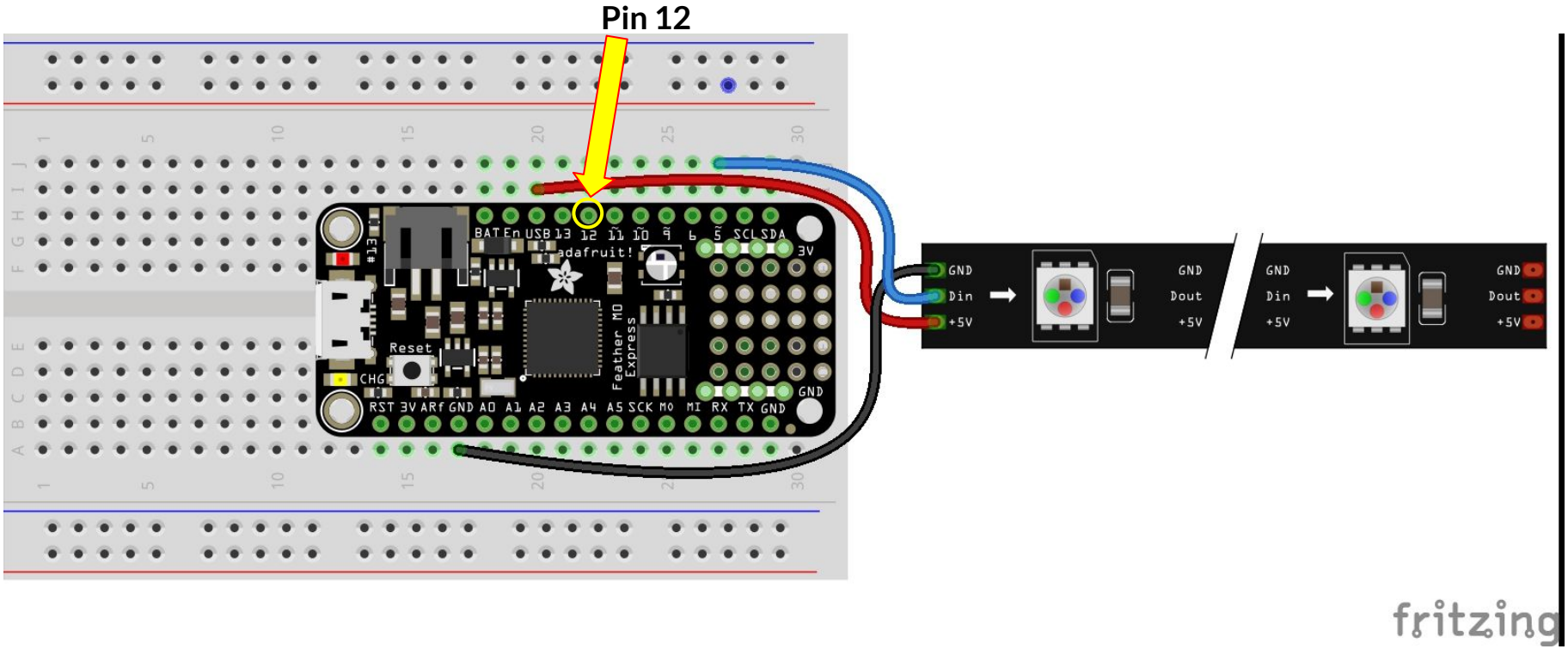


Photo Credit: Kattni Rembor

Connecting the NeoPixels to the Pi

Connect the Data wire to pin 12

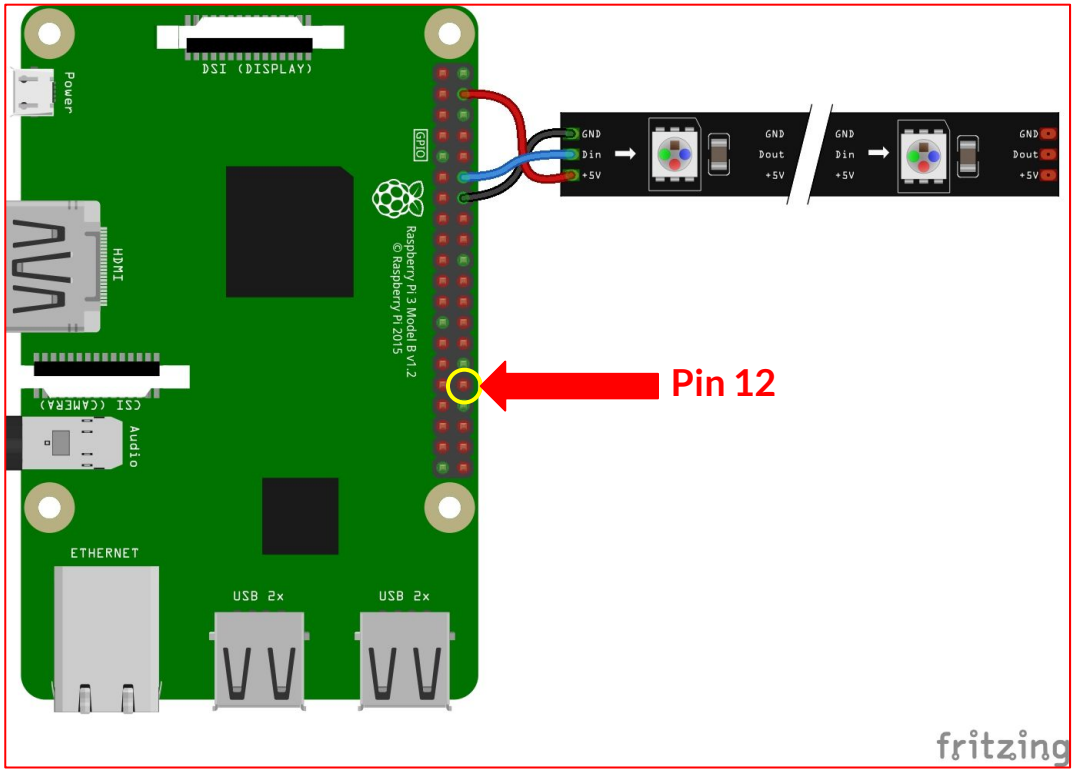
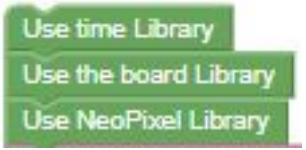


Photo Credit: Kattni Rembor

Include the necessary Libraries

The program will need the Time, Board and NeoPixel libraries. Add the **Use time Library** block by opening the menu on the left labeled **Misc** and dragging it onto the workspace.

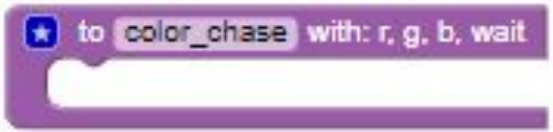


Add the **Use board Library** from the **CircuitPython** menu. Then add the **Use NeoPixel Library** from the **CircuitPython>NeoPixel** menu.

Create a re-useable function

For different color effects

To save some coding time and create efficient programs, we create functions to use code over and over again. Often when using functions you change things a little bit to acheive slightly different results. In this function we create a chasing pattern that can take Red, Green, Blue and Wait values to change the color of the chase.



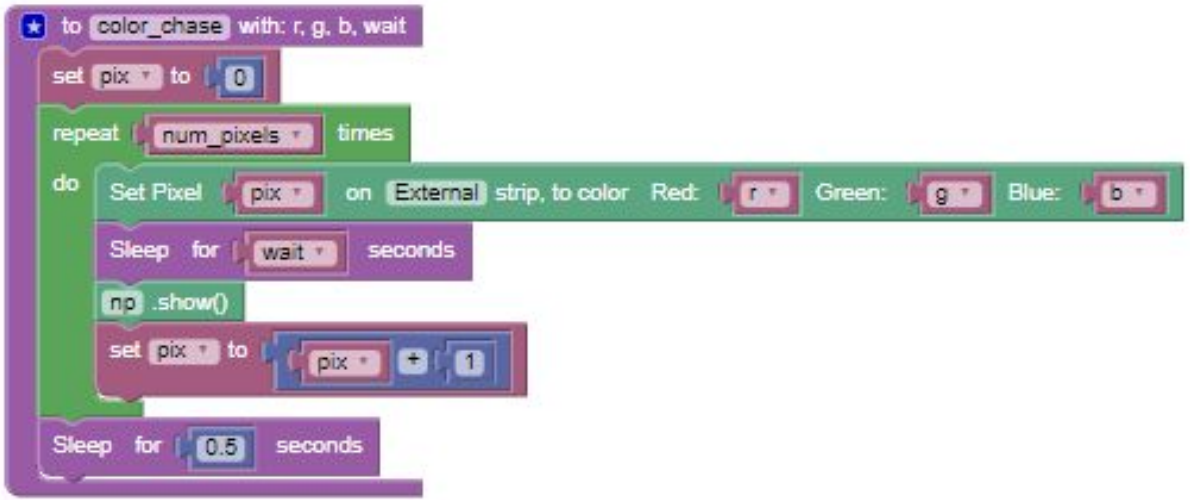
For the chase we will use a repeat block to loop through the number of available NeoPixels and change the color based on the incoming Red, Green and Blue values

Original Code credit: Kattni Rembor

Create a re-useable function

For different color effects (cont.)

For the chase effect, we will use the wait parameter to specify a delay between lighting each NeoPixel



Main Program for NeoPixels

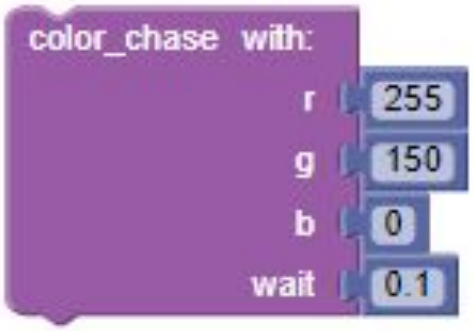
For NeoPixels, we set up a few things: the number of pixels in the string, setup the string by specifying the communication pin and brightness of the pixels

On the Feather board, we generally use a continual loop for programs. We can use the same loop on the Raspberry Pi

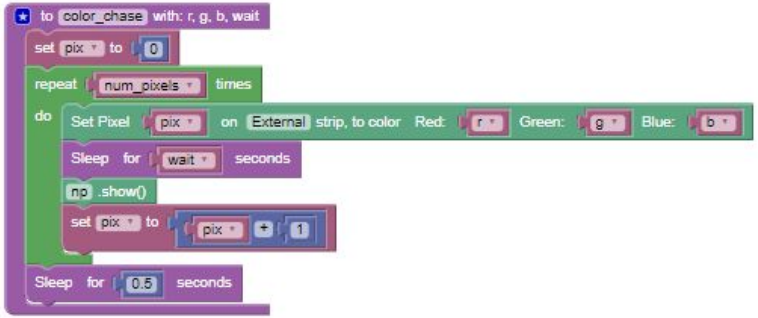
Within the loop we will do a basic single color fills of all pixels as well as call our special color_chase function.

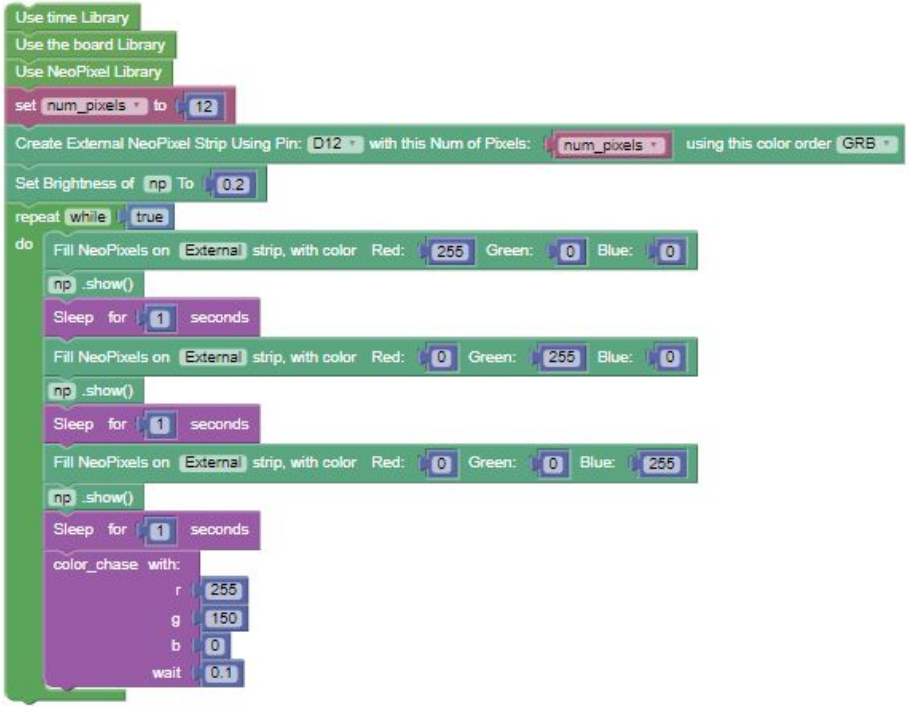
Main Program for NeoPixels (cont)

The last part of the program is to make the call to the color_chase function with the Red, Green, Blue and Wait values



This is the final program all put together





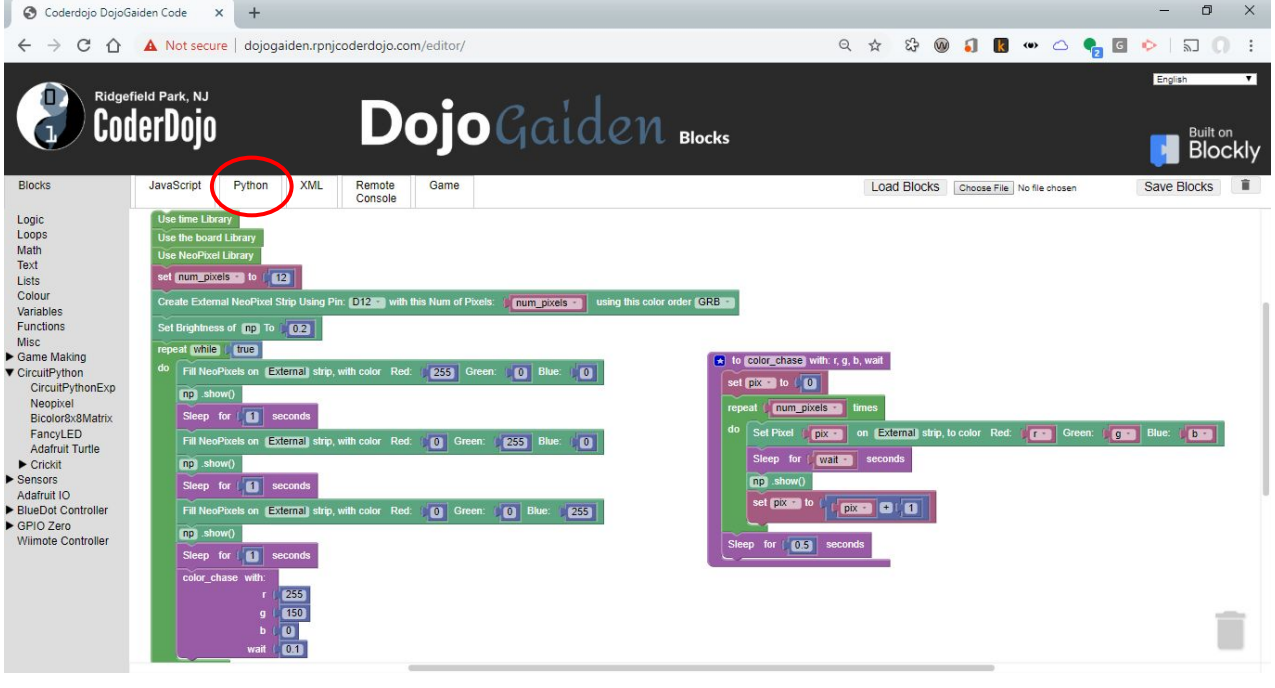
Deploying to the Feather

We need to generate Python code to copy onto the Feather board, so we click the Python button to get the generated code to save directly onto the CIRCUITPY drive for the Feather.

Copy code to the Feather

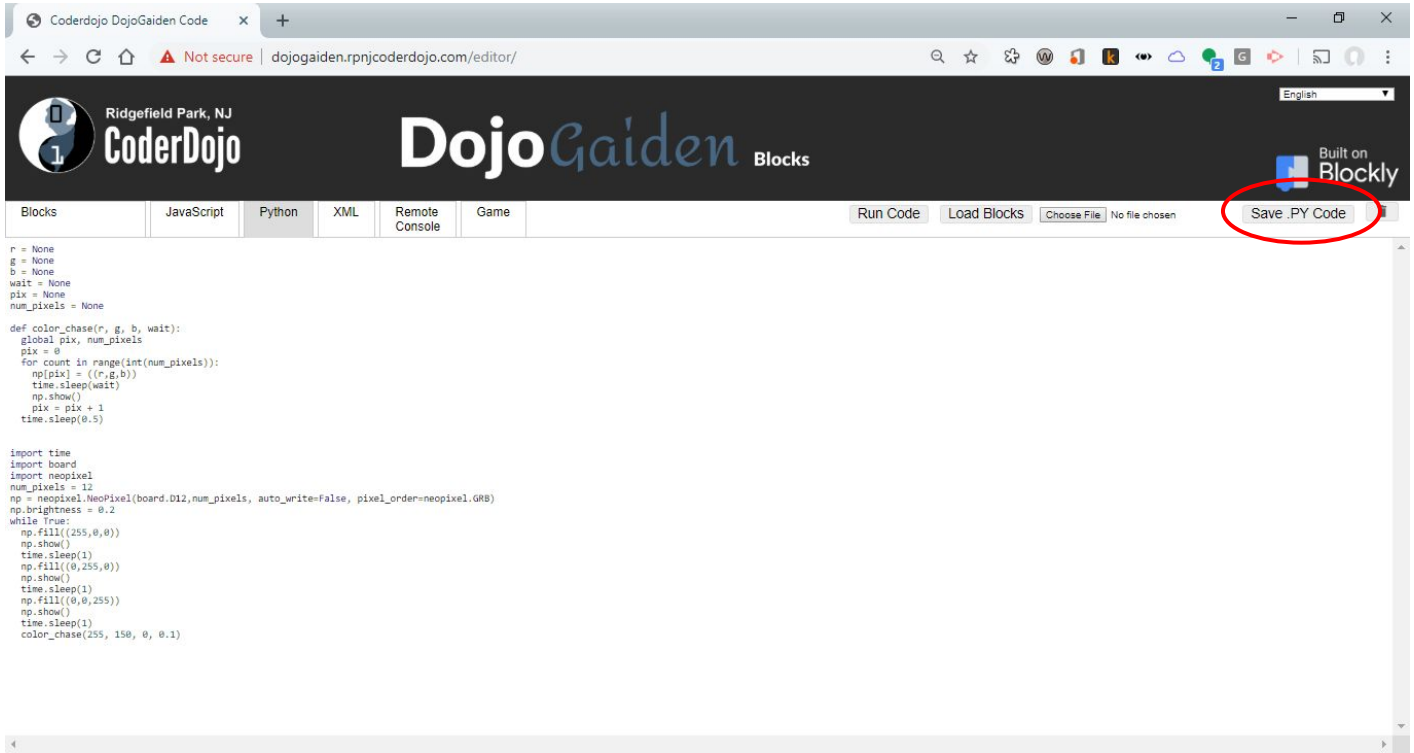
Once you connect the Feather board to the USB port, you will see a USB drive appear on your computer. This will allow us to copy our code directly onto the Feather as you would copy any file between locations on your computer. Even drag and drop works as well.

To copy our block program to the Feather USB Drive we will need to view our blocks as python code. In DojoGaiden Blocks, we do that simply by clicking the Python button.



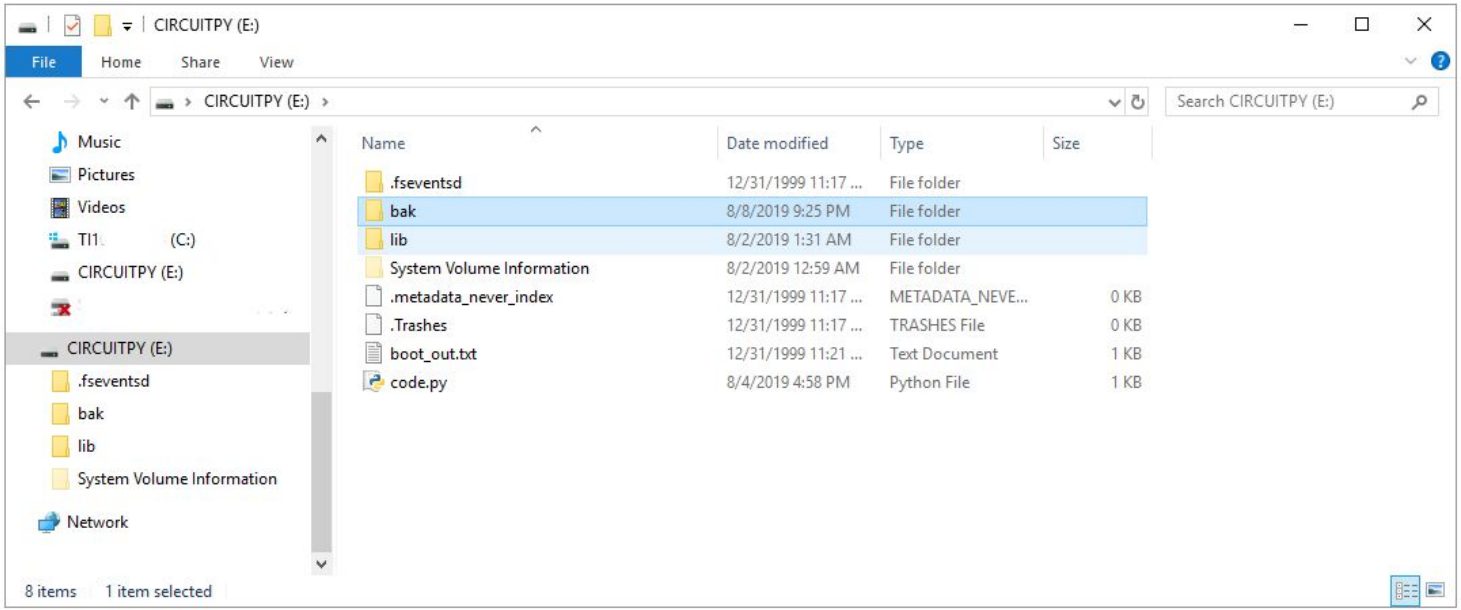
Copy code to the Feather (cont.)

When you press the Python button, the generated python code is shown in workspace window. This code will be copied to the Feather for execution. To do that, we will use the Save .PY Code button. You can save the file to a local directory, or even better directly save it to your CIRCUITPY drive.



Copy code to the Feather board

Now that we have completed our block program, we want to run the code on the Feather board. This is done by copying a file called code.py onto the CIRCUITPY drive for the Feather.



When the file is done copying onto the board, it should reboot and your code should start running. If the board does not reset automatically, press the reset button and the board should restart.

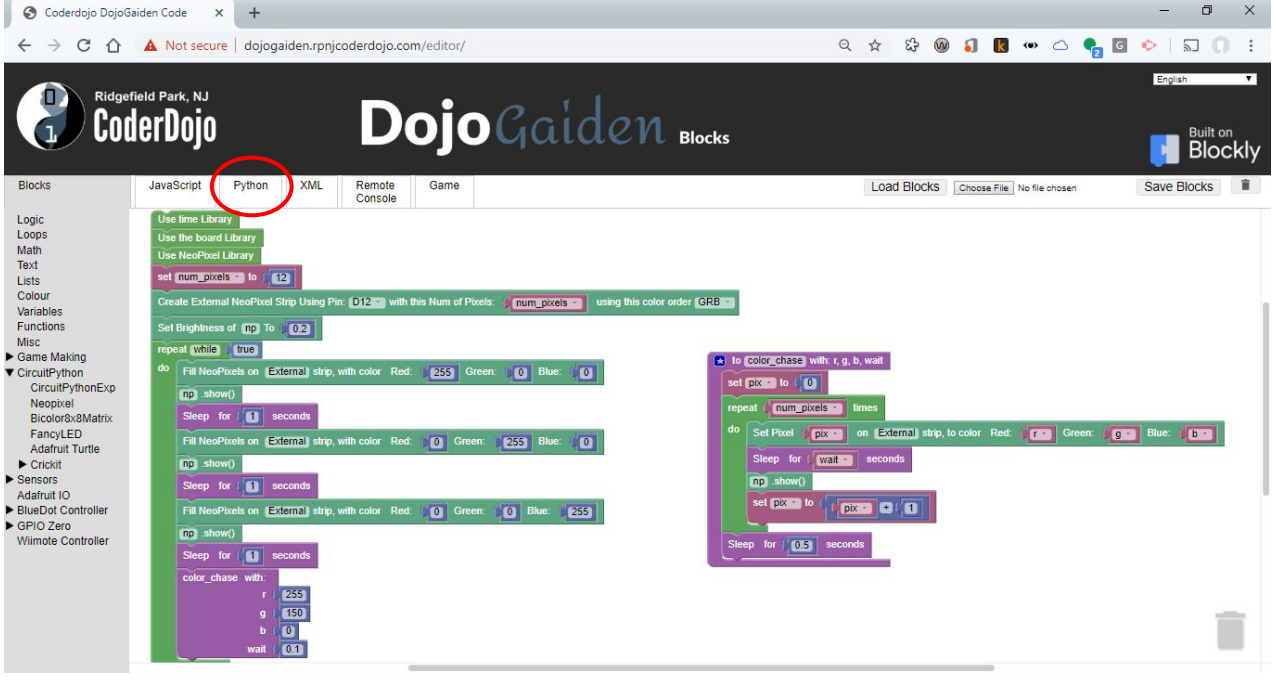
Deploying to the Pi

We need to generate Python code to copy onto the Pi, so we click the Python button to get the generated code to save directly onto the share directory on the PI.

Copy code to Raspberry Pi

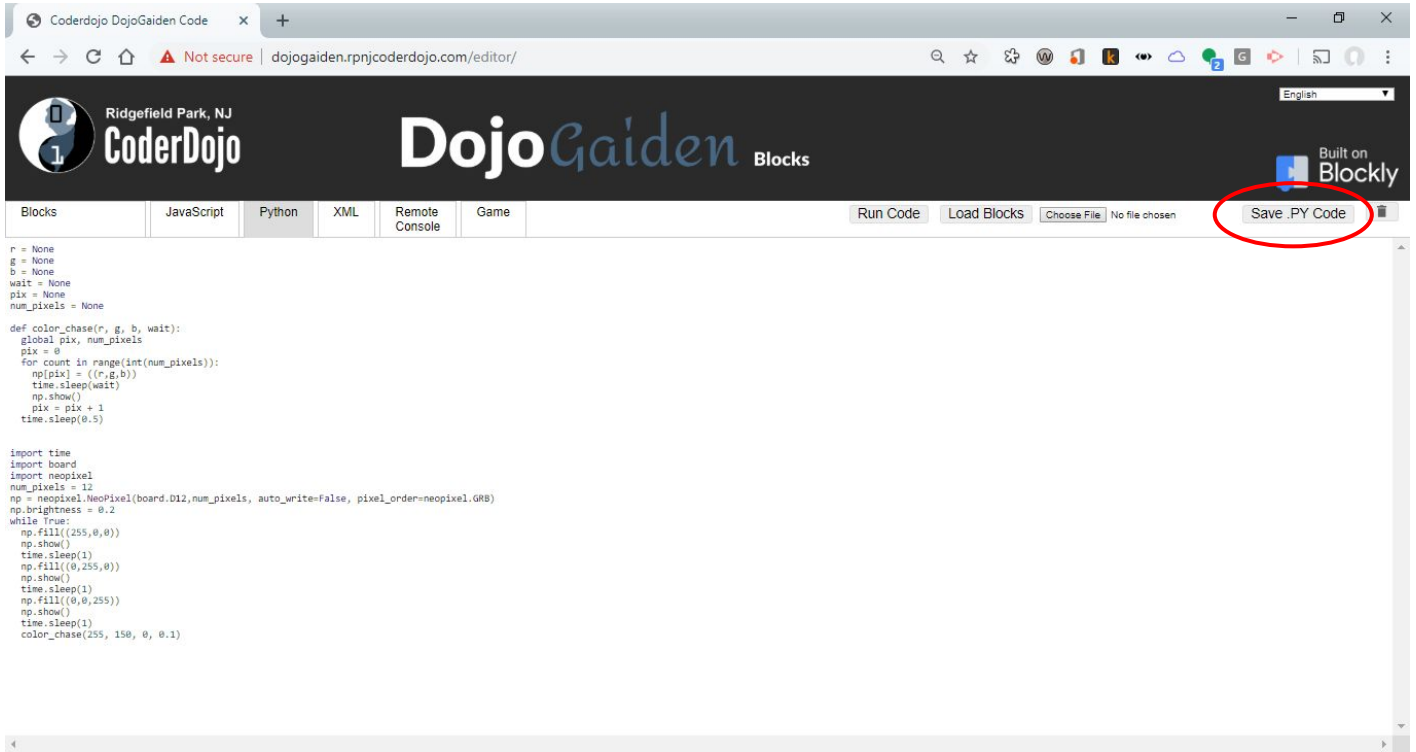
Once you are able to connect to the rPi Samba share, you can simply use that share as you would any drive on your computer. This will allow us to copy the code onto the Pi as you copy any file between locations on your computer. Even drag and drop works as well.

In order for the Pi to run our block program we need to view our blocks as python code. In DojoGaiden Blocks, we do that simply by clicking the Python button.



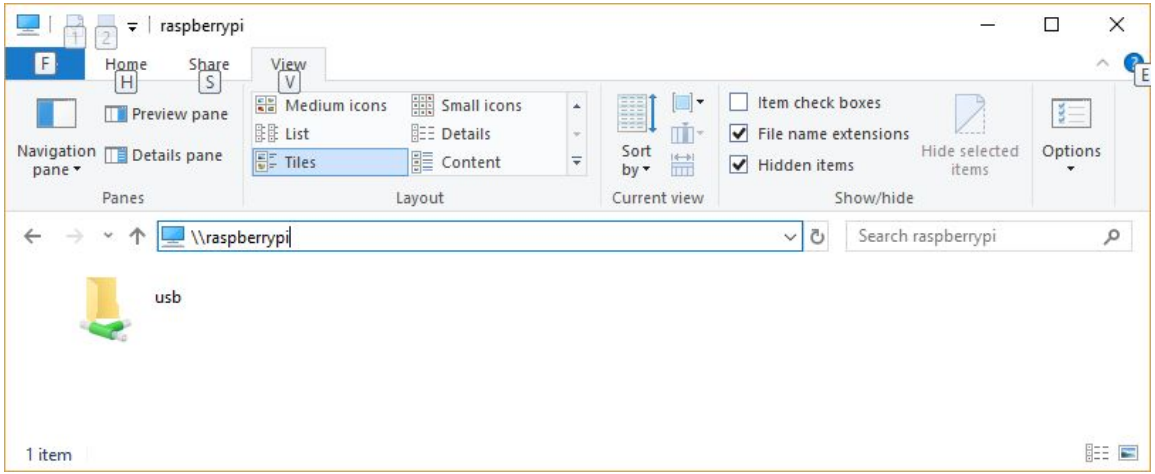
Copy code to Raspberry Pi (cont.)

When you press the Python button, the generated python code is shown in workspace window. This code will be copied to the Raspberry Pi for execution. To do that, we will use the Save .PY code button. You can save the file to a local directory, or even better directly save it to your \\raspberrypi share.



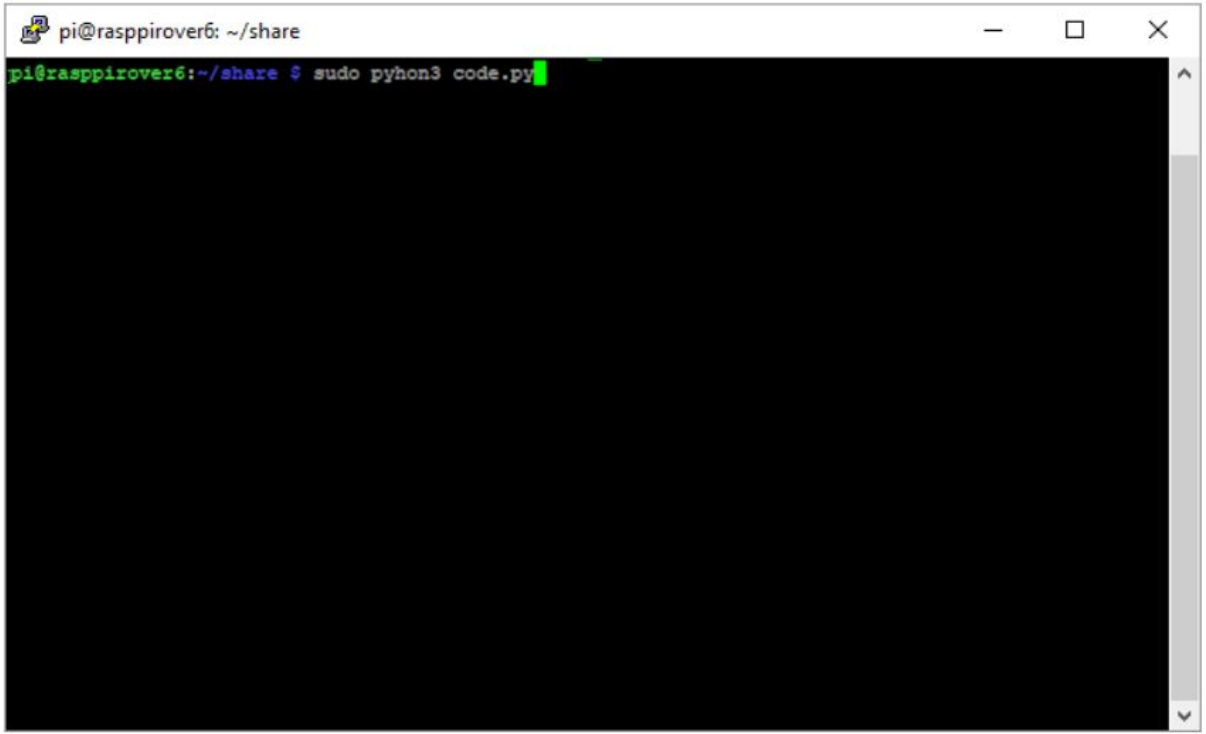
Copy code to Raspberry Pi

Now that we have completed our block program, we want to run the code on the Raspberry Pi. For this step you will have already created a Samba share on the rPi. The share that I created is called “usb”. But if you forgot the share you created, with Samba, you can actually browse to it from your File Explorer application. To do that you simply need the name of the rPi. (usually raspberrypi) To browse, use the UNC notation format \\raspberrypi in your File explorer location field.



Execute code on the Raspberry Pi

The final step in the process is to run the code on the Raspberry Pi. Since we have copied the code onto a mounted directory we can directly run the code from that directory. To do that, we have to open a terminal to the Pi through SSH. Since I named my files **code.py** I execute them using sudo python3.



DojoGaiden Blocks is an Educational Tool for Coding

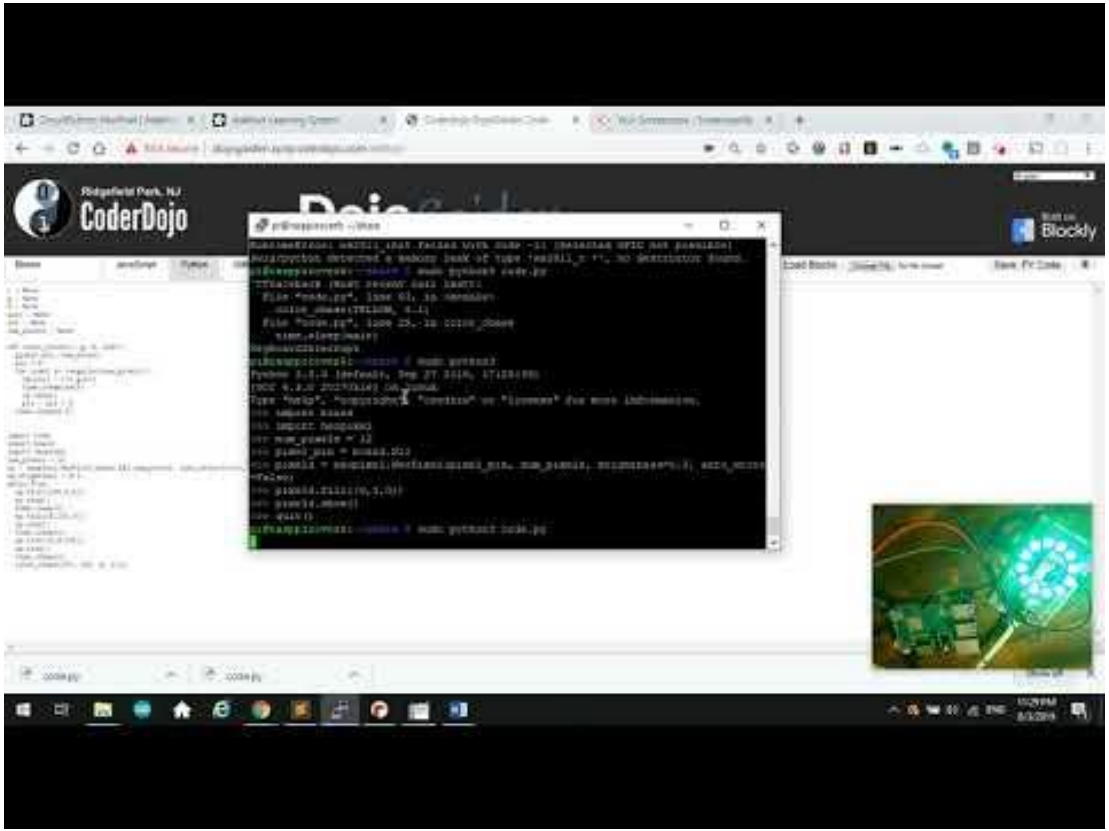
DojoGaiden Blocks is a great transitional tool which assists school age children whom have worked with Scratch and want to up their coding skills to the next level.

Screencast of building and deploying the program to the Feather and the Pi

https://youtu.be/FUWppSs1G60

Screencast of running the same program on both the Feather and the Pi

https://youtu.be/d7PV5kt3jbw



Thanks for watching

In future video, we will show different sensors and devices such as LED Matrices, Gyroscopes and more.

Stay Tuned!