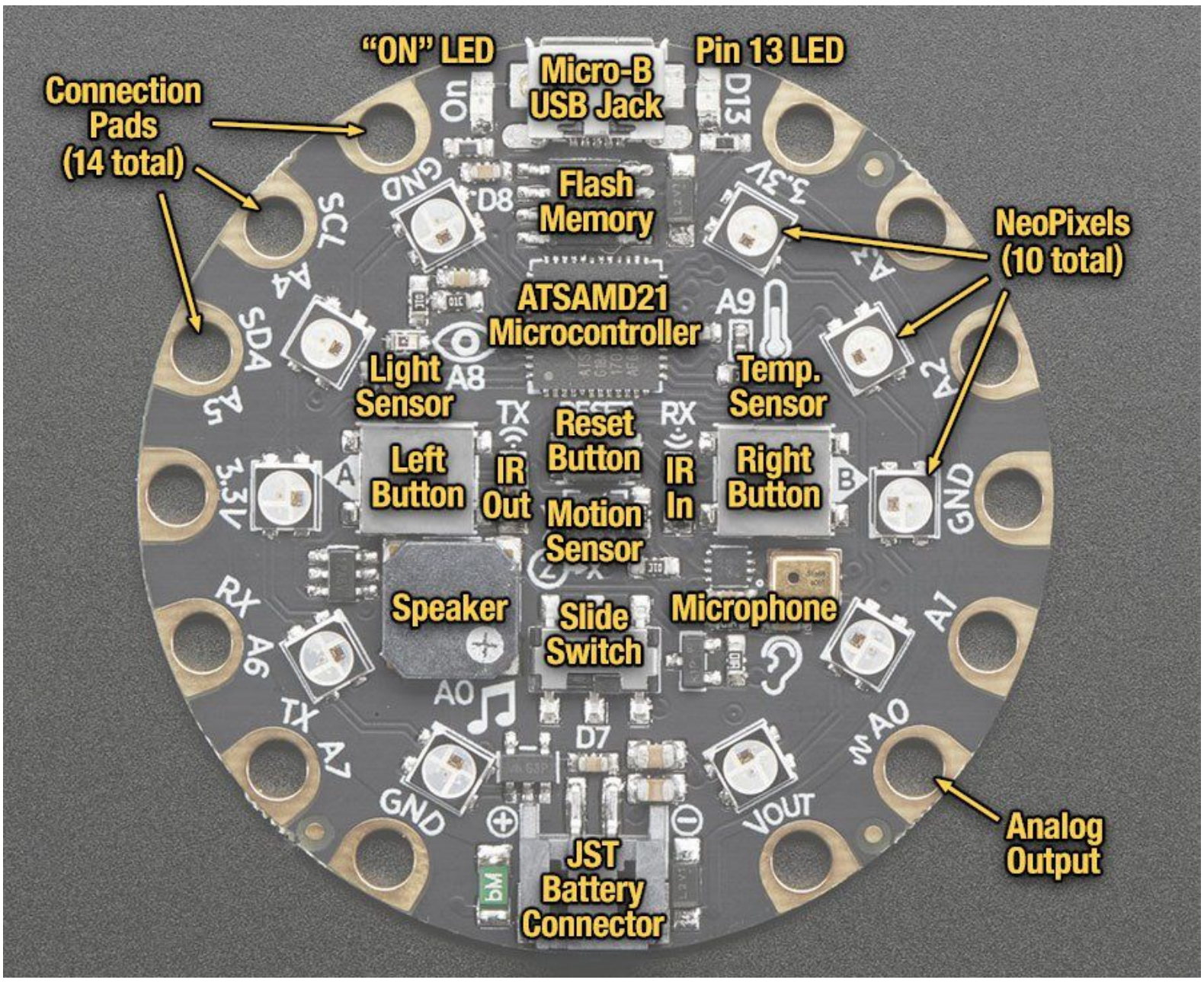
This is an example of the Circuit Playground Express board that we will be using today. The board has many sensor, lights, a switch, and buttons. All of these can be used in the programs we will be creating.



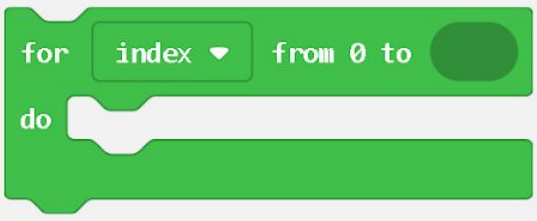
Schedule

8:30 Introductions
8:35 What we will be learning today
- Basic coding terminology
- Loops
- Conditionals/Boolean/Comparison
- Variables
- Math
- Functions

8:50 Connecting to the Guest network
9:00 What is Microsoft Makecode and how does it work
- Opening Microsoft Makecode for Circuit Playground
- Creating a new project
- How to use the project interface
    - Shapes in the project interface and how they fit together
    - Deleting shapes
    - Duplicating shapes
- Loading an existing program into Makecode
- Saving your program to a file
- Loading your program onto your Circuit Playground

9:30 Our first program - Lets make a light blink.
9:40 Demo - Lets make a light blink with a different language using the same logic.
9:45 Break
10:00 Loops - Lets build our first while loop
10:10 Loops with conditionals - If something is true do something while another thing is true.
10:20 Loops with logicals - A different way to do the same thing
10:30 Lets talk about what we just learned
10:35 What can you create using what we have already learned?
10:45 More conditionals - If/then/else logic
11:00 Break
11:10 Hot Potato - Variables, Random Number, Loops, and Luck
11:45 Free time to create with what we have learned so far
12:00 Lunch
1:00 Tilt Sensor - If/Then/Else with hardware input
1:15 Tilt Trumpet - Nested If/then/else conditionals
1:45 Break
2:00 Fireflies - updating a variable based on other inputs
2:30 Free time to create with what we have learned
2:45 Functions
3:00 Zombie Game
3:45 Break
4:00 Show and Tell

Loops

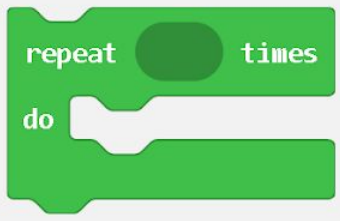


for

Repeat code for a given number of times using an index.

while

Repeat code while a condition is true.



repeat

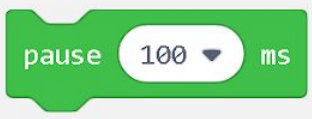
Repeat code for a given number of times.

for of

Repeat code for each item in a list.

forever

Repeats the code forever in the background.



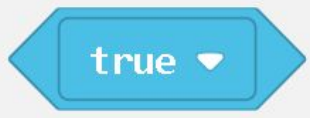
pause

Pause for the specified time in milliseconds.

Logic

if

Conditional statement.

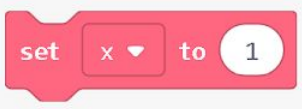


Boolean

True or false values.

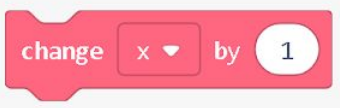
Variables

Assign (set) a variable's value



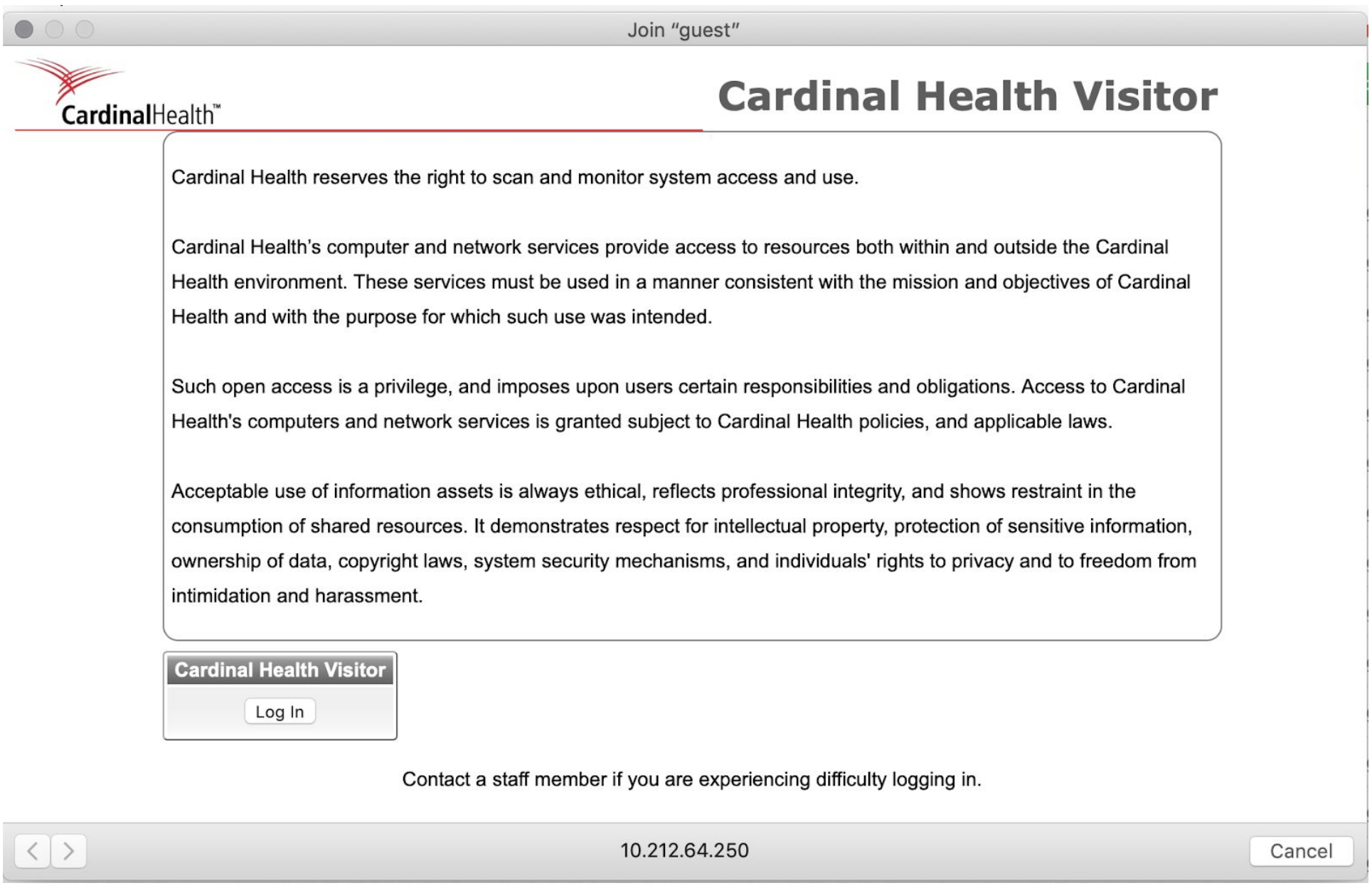
Get a variable's value

Change a variable's value



Connecting to the guest network

- On your computer click the proper icon for the wireless network
- Choose the guest network
- When it prompts you to accept the terms click the Log In button

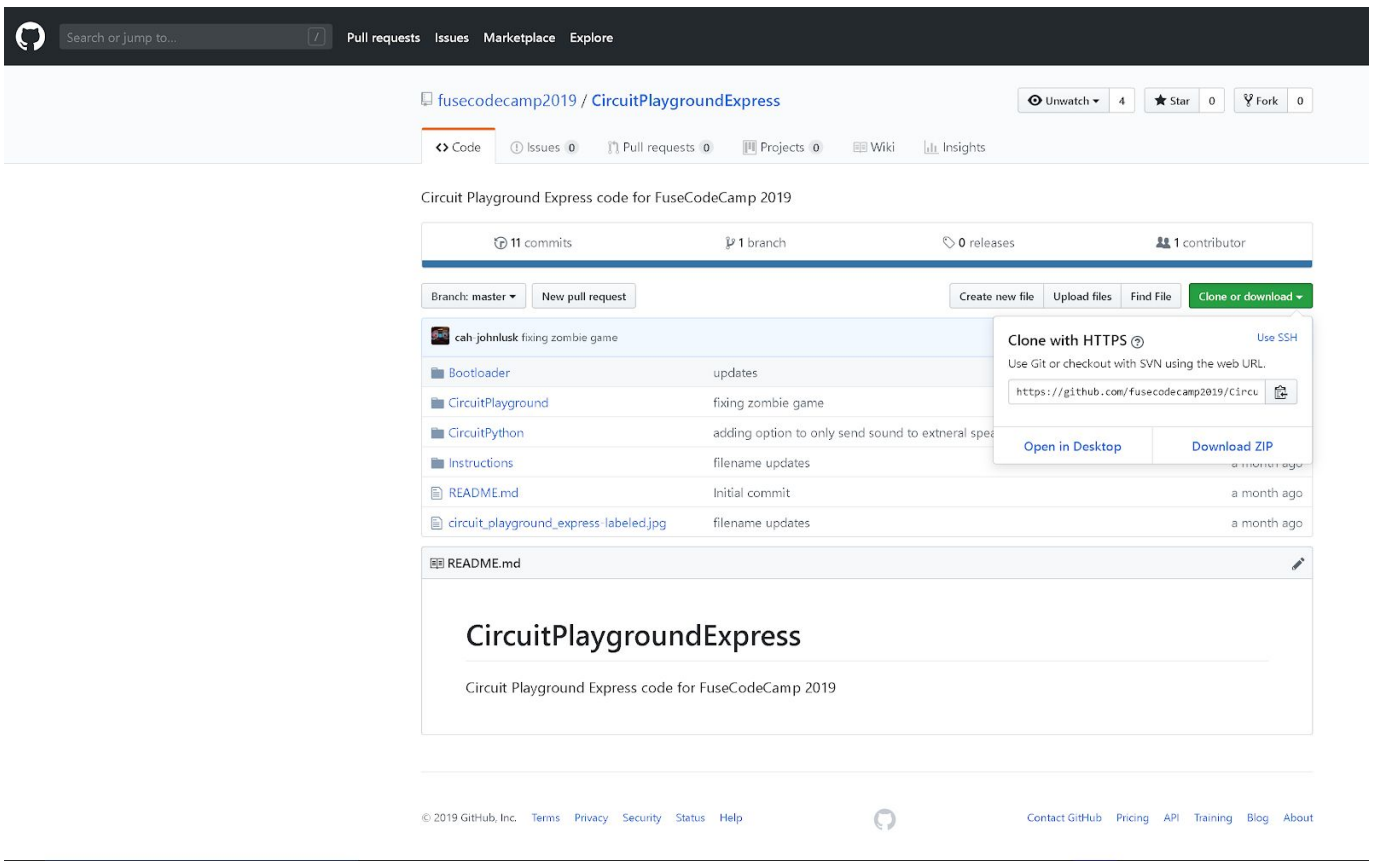


What is Microsoft Makecode

- Opening Microsoft Makecode for Circuit Playground
  - https://makecode.adafruit.com
- Creating a new project
- How to use the project interface
  - Shapes in the project interface and how they fit together
  - Deleting shapes
  - Duplicating shapes
- Loading an existing program into Makecode
- Saving your program to a file
- Loading your program onto your Circuit Playground

Github - Source Code for CircuitPlayground Express

https://github.com/fusecodecamp2019/CircuitPlaygroundExpress



Open the URL for Github above and Click on the Green Clone or Download button. Click on the Download Zip option and save the zip file somewhere you can remember such as your Documents folder or your desktop. Go to your File explorer/Finder and browse to where you saved the file. You can right click on the file and extract the folder or for OSX you can double click on the file. This contains the majority of the code that we will be working with today.

Connecting your CircuitPlayground to your Computer

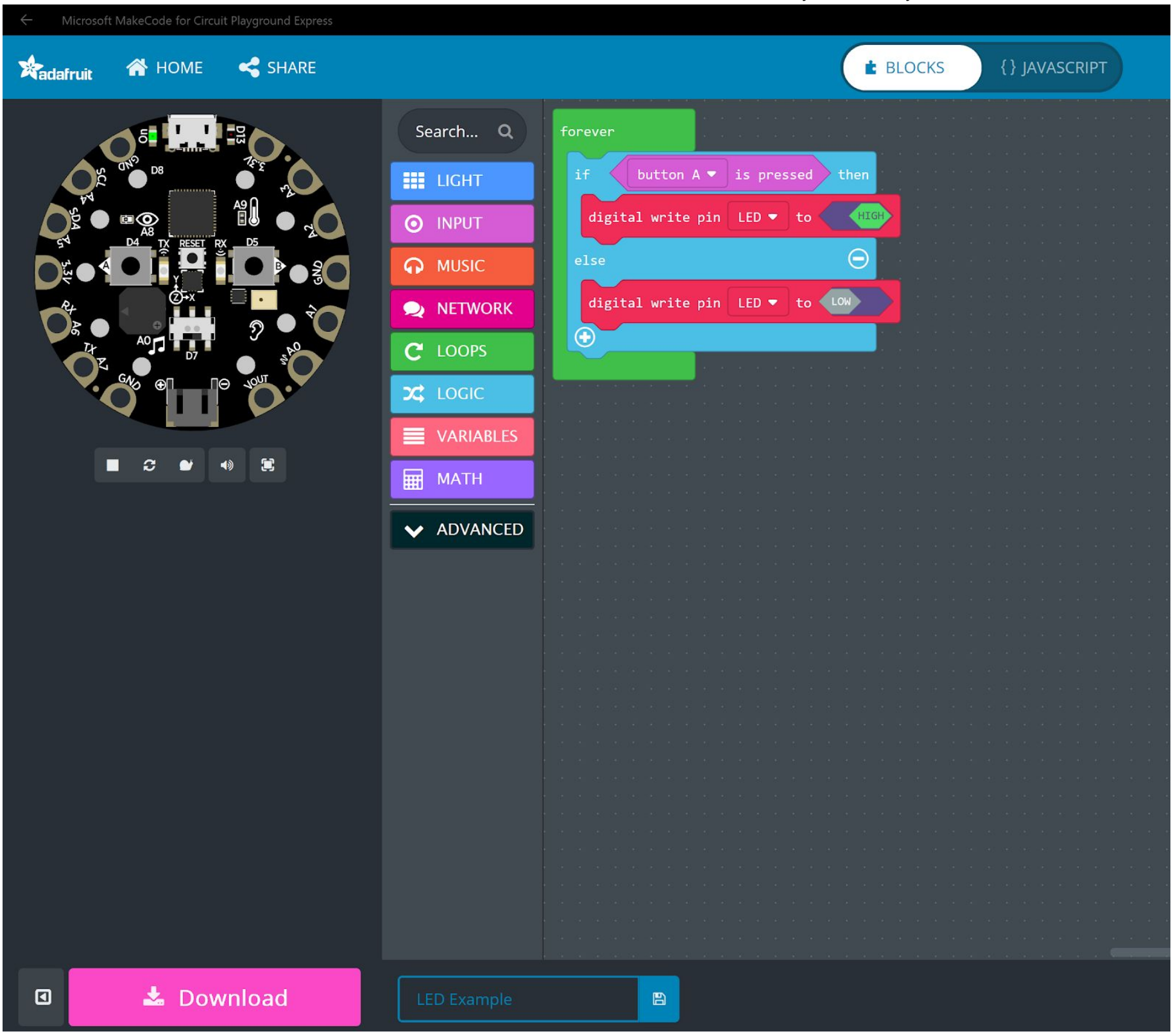
1. Open the box that contains your CircuitPlaygroundExpress Board, MicroUSB Cable, and Battery Pack
2. Plug the microUSB end of the cable into the port on the CircuitPlaygroundExpress -- **Be Careful and don't force it into the port**
3. Plug the other end of the cable into the USB port on your computer

The CircuitPlaygroundExpress should light up and all of the lights on the Neopixels should light up green. If the lights do not light up green press the small button in the center of the board.

Open your File Manager/Finder and you should see a drive named CPLAYGROUNDBOOT.

Our first program - Lets make a light blink

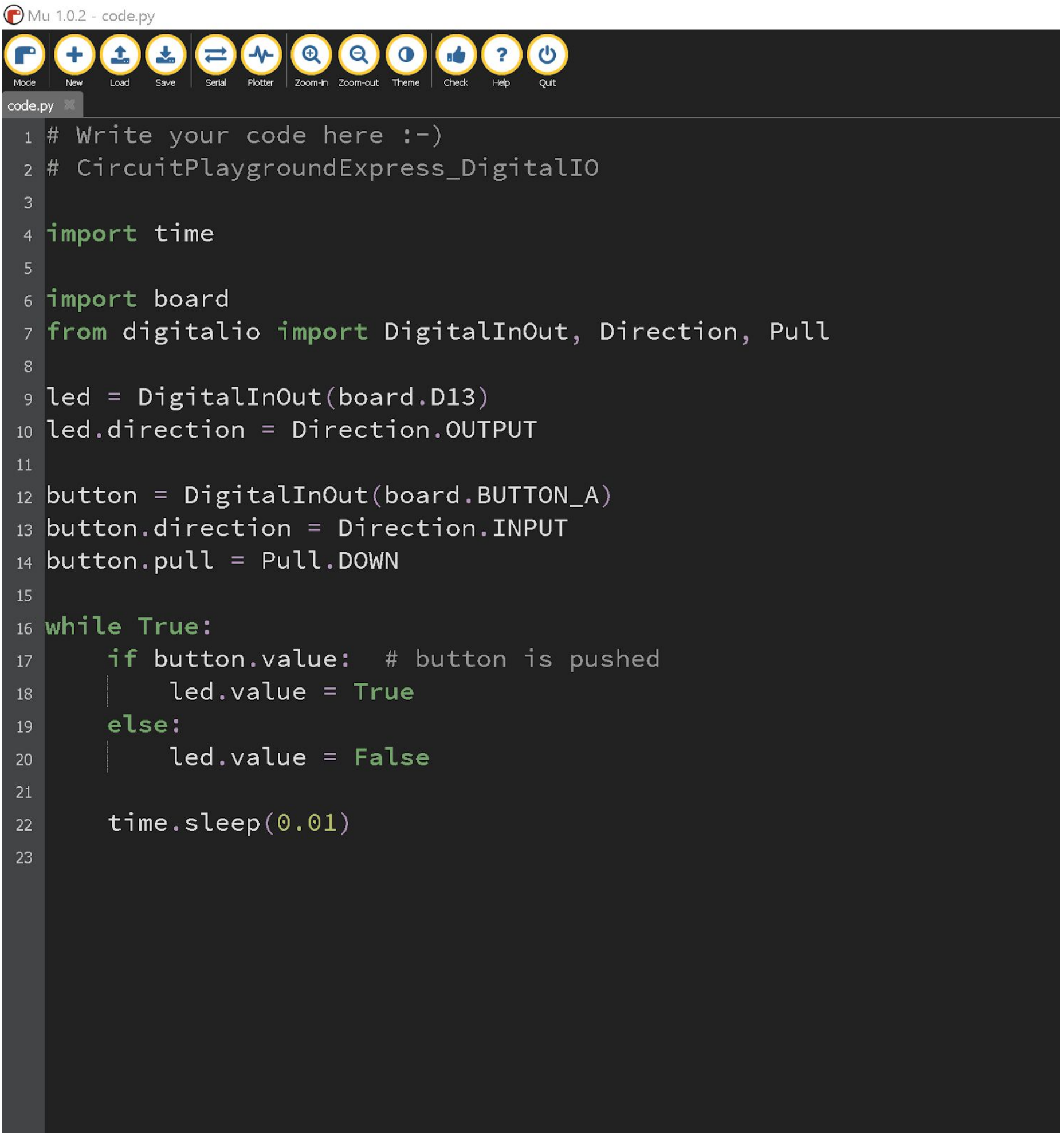
MakeCode - This will make an LED on the board blink when you press the A button. This is an example of conditionals. Conditionals use logic like If/Then/Else and If/Then/Else if/Else. We will cover different combinations of conditionals in many forms today.



This is the Javascript code which is generated by the MakeCode blocks

Demo - Lets make a light blink with a different language using the same logic

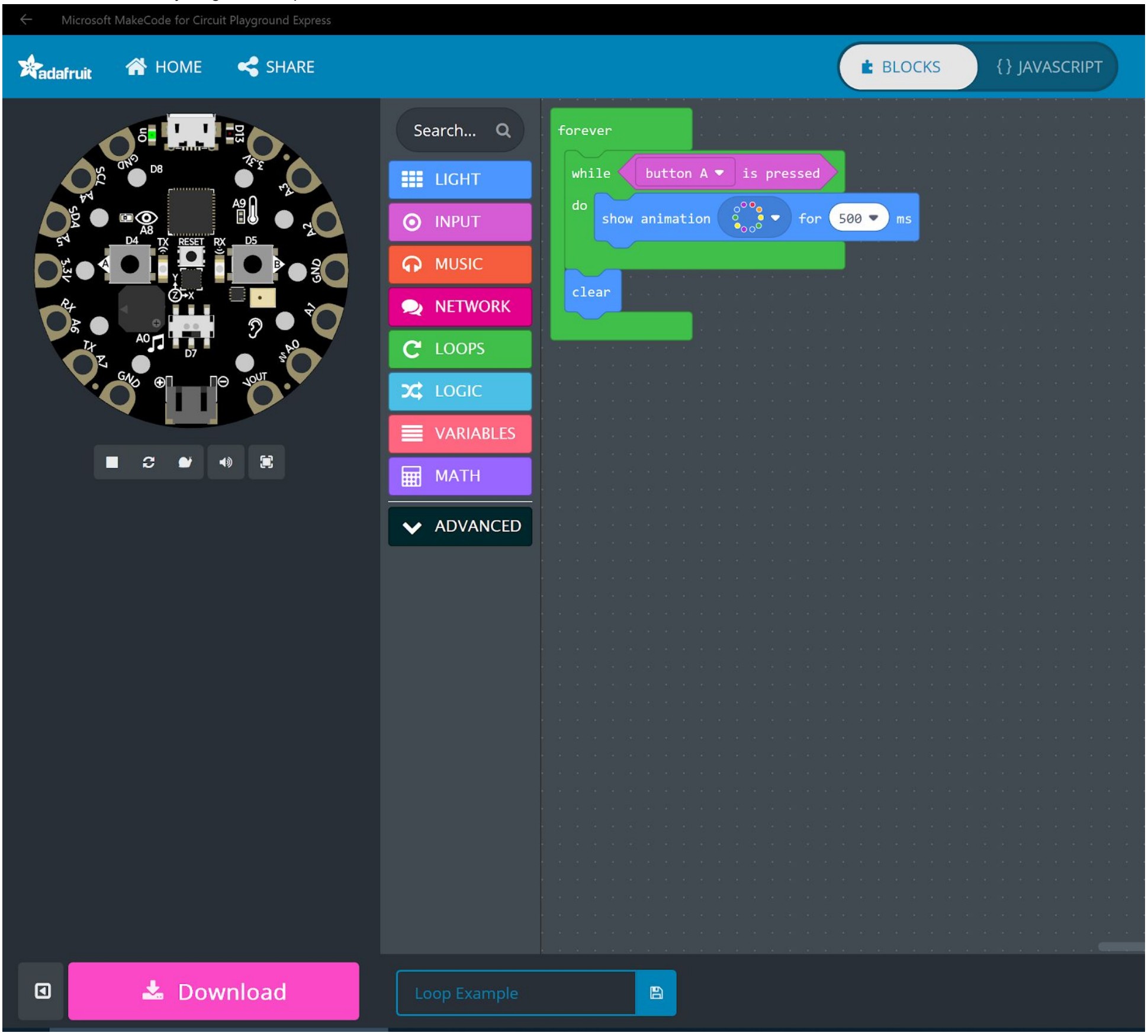
This is CircuitPython code. It is python code which will perform the same function and uses the same logic.



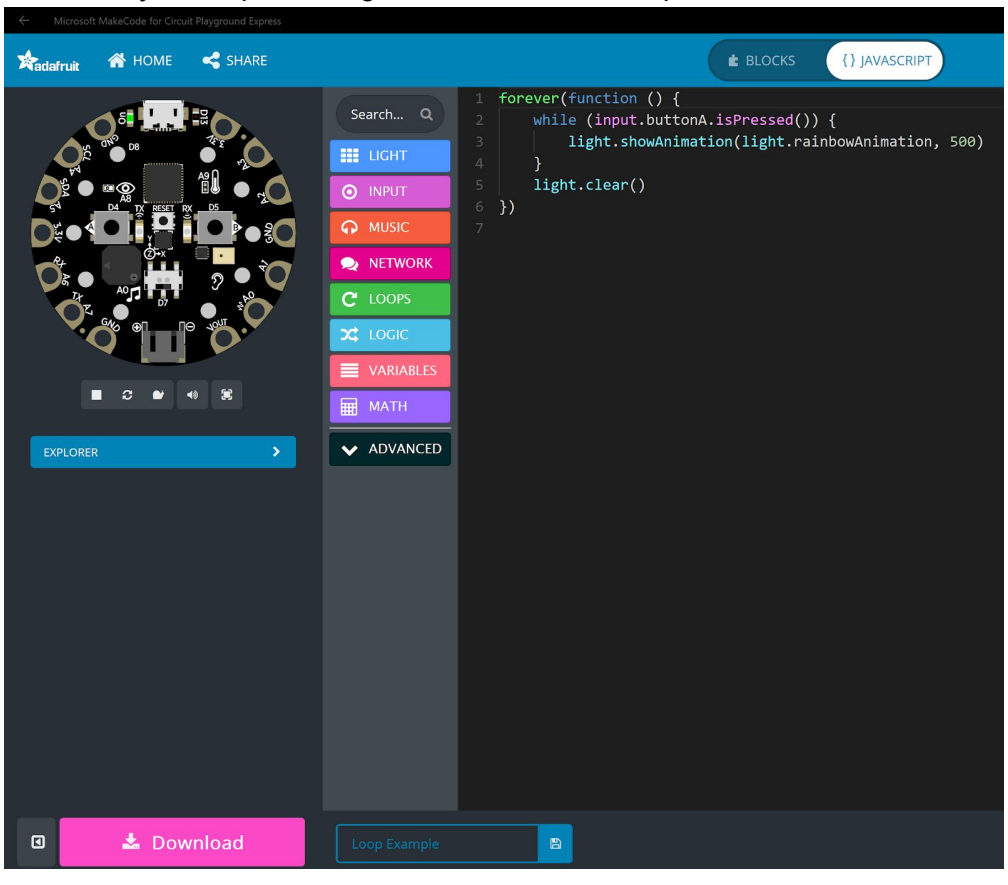
Break Time - 10 Minutes

Loops - Lets build our first while loop

This is an example of a loop. It is a simple while loop. The code inside the while loop will continue to execute **WHILE** the condition is true. When the condition is not true any longer the loop will exit.

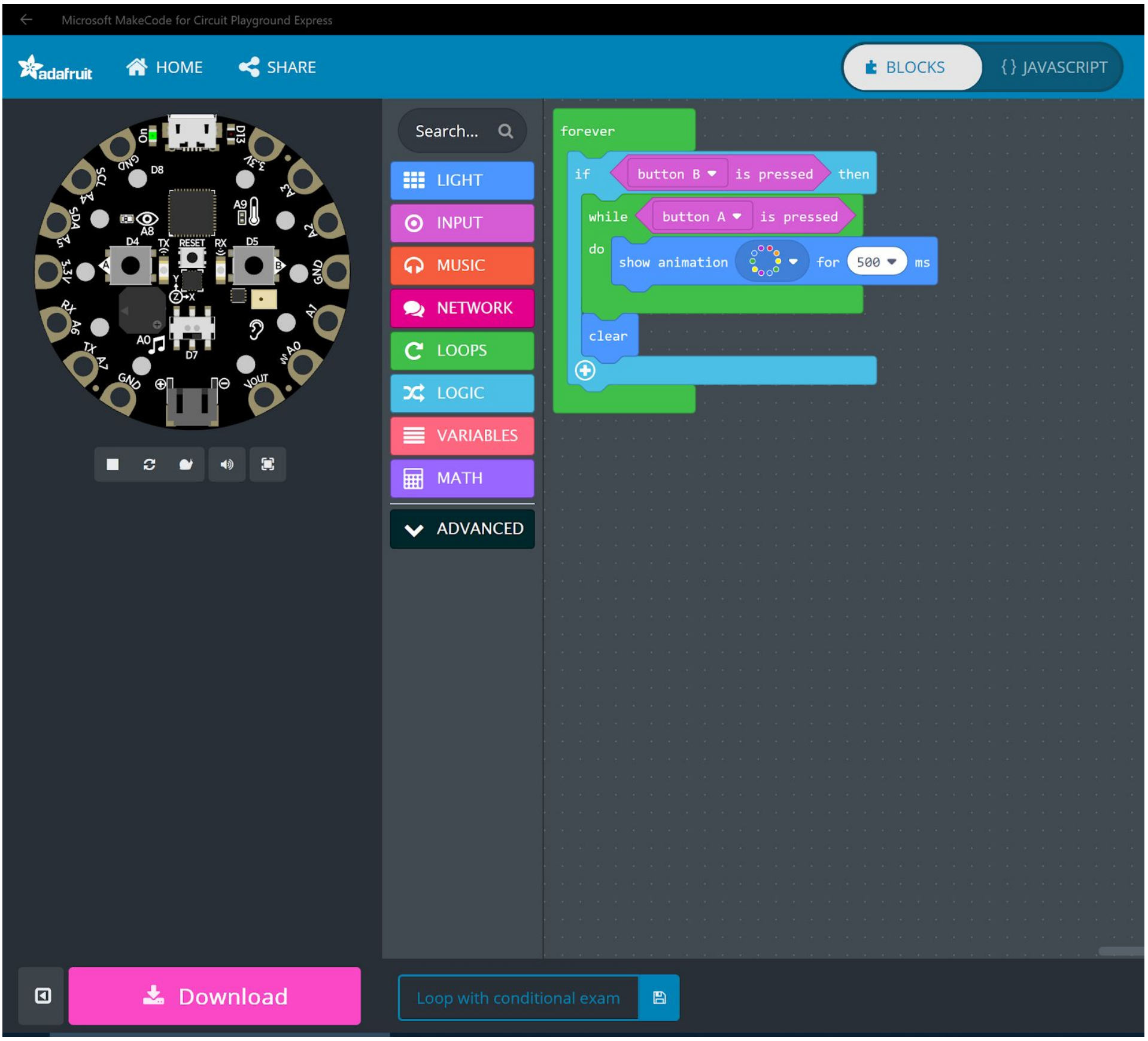


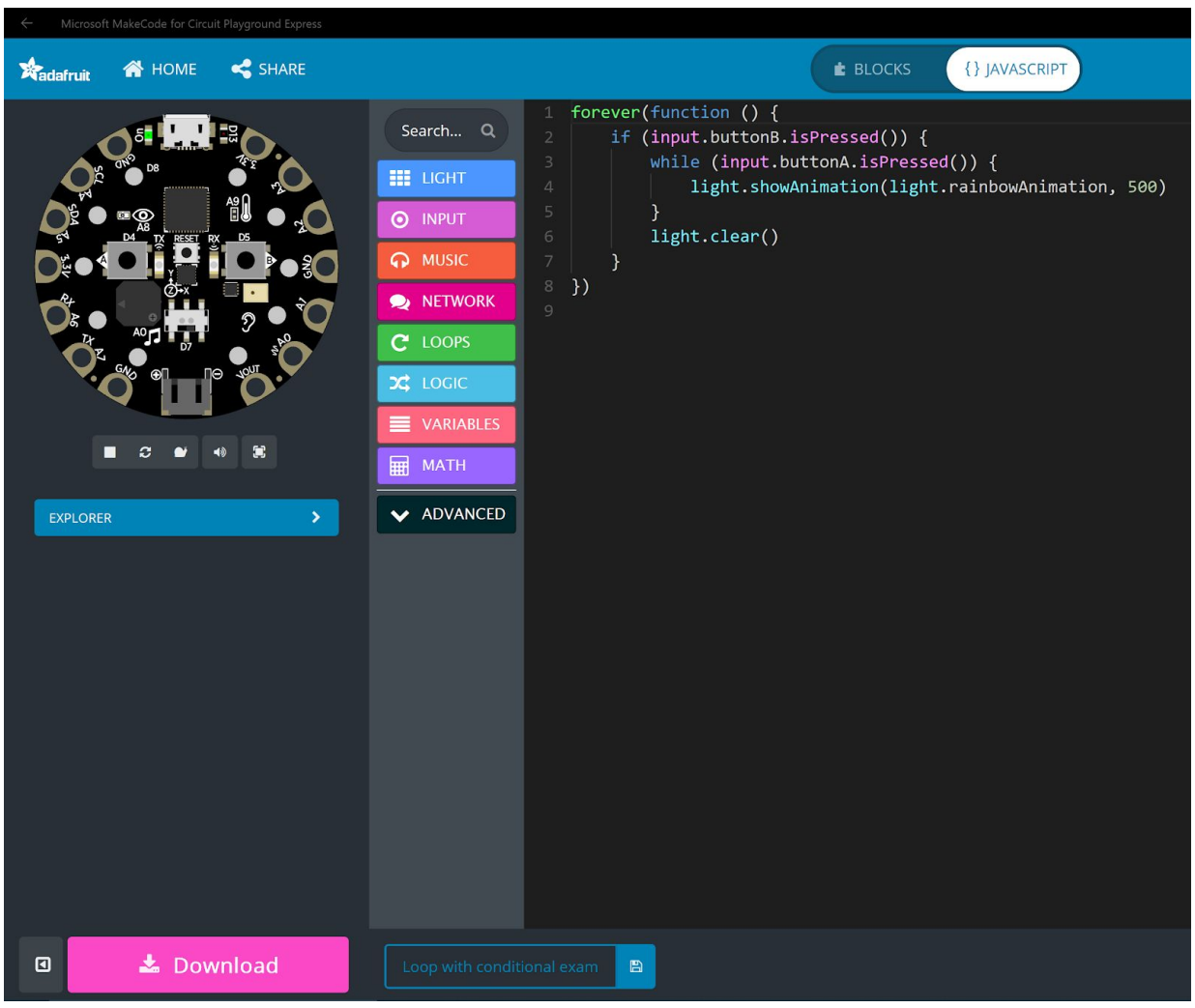
This is the javascript that is generated for the example above.



Loops with conditionals

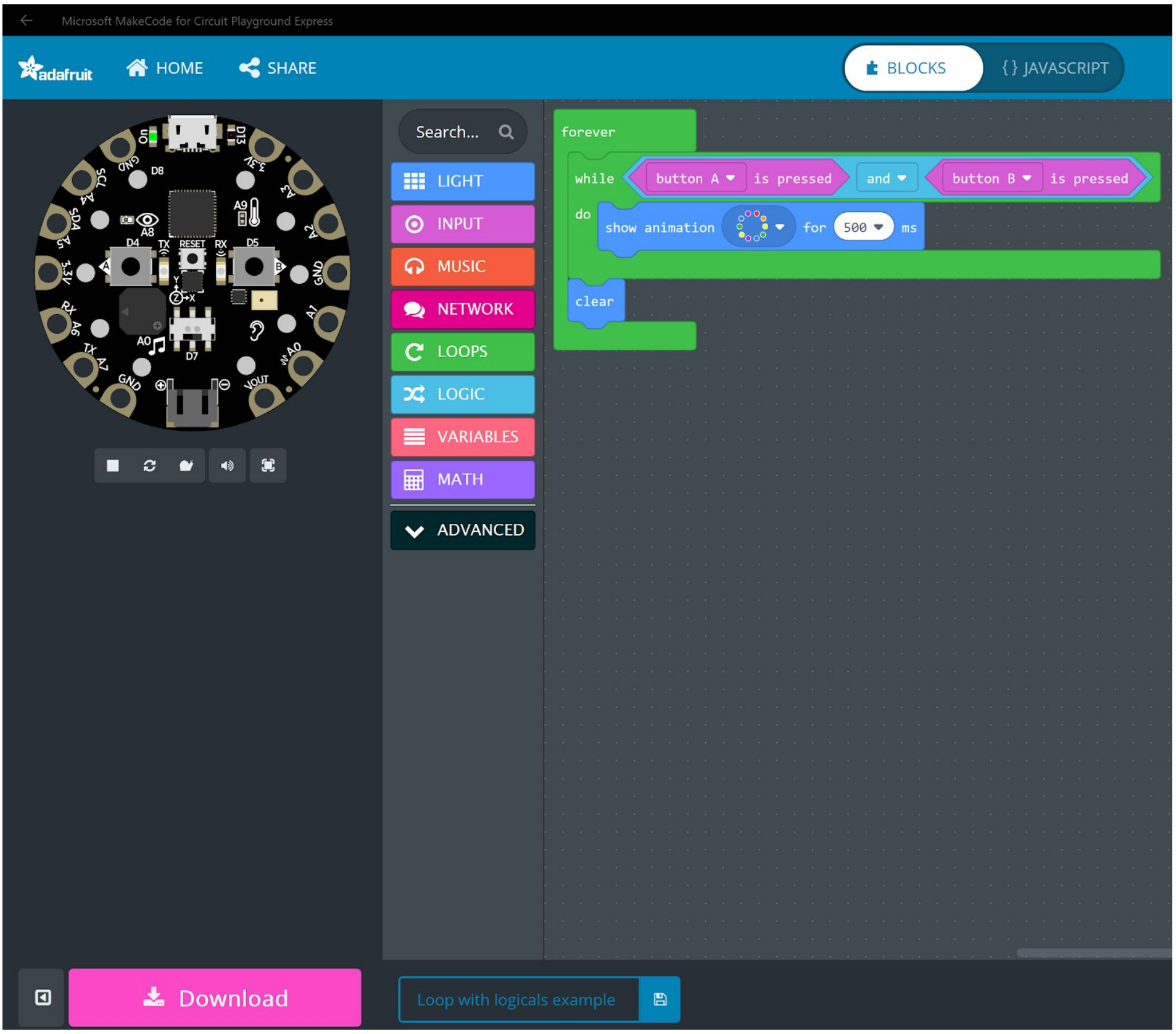
This is another example of the same while loop but it is now nested inside a conditional block. This conditional will execute the code that is inside the block if the condition is true.



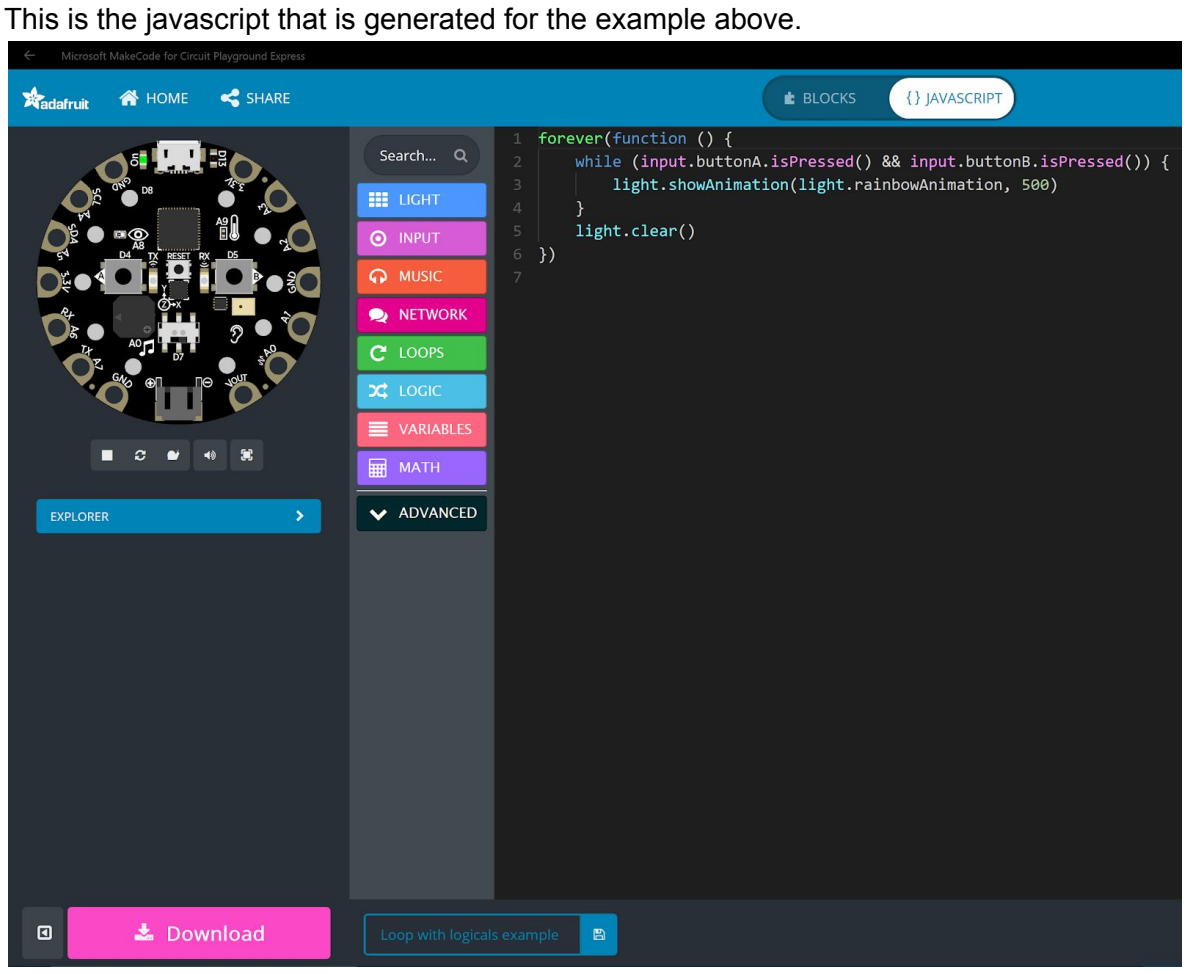


Loops with logicals

This is an example of a while loop using logicals. Logicals or logical operators are comparisons such as AND/OR/NOT. In this case two conditions must be met in order for the while to be true and to execute the code.

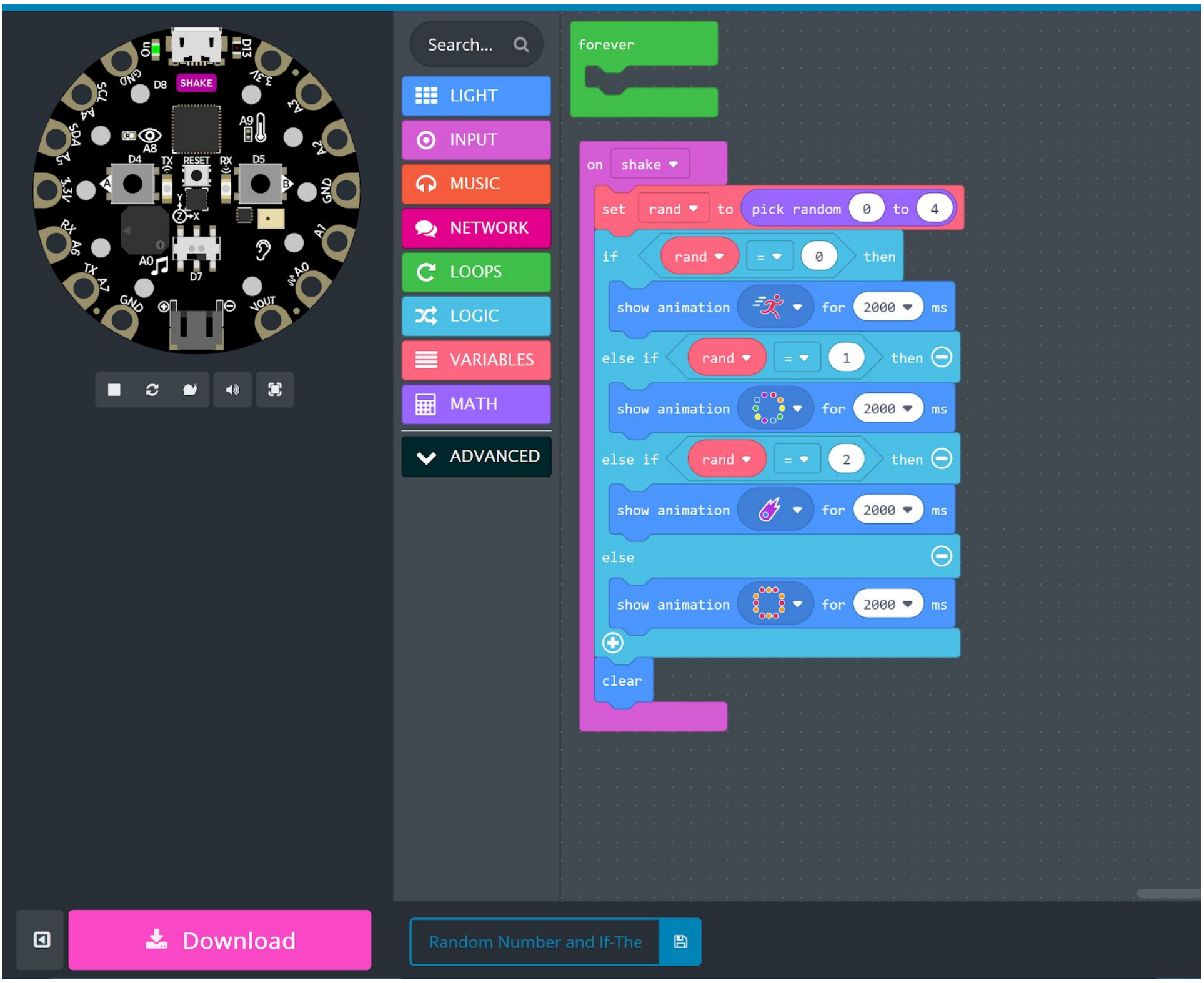


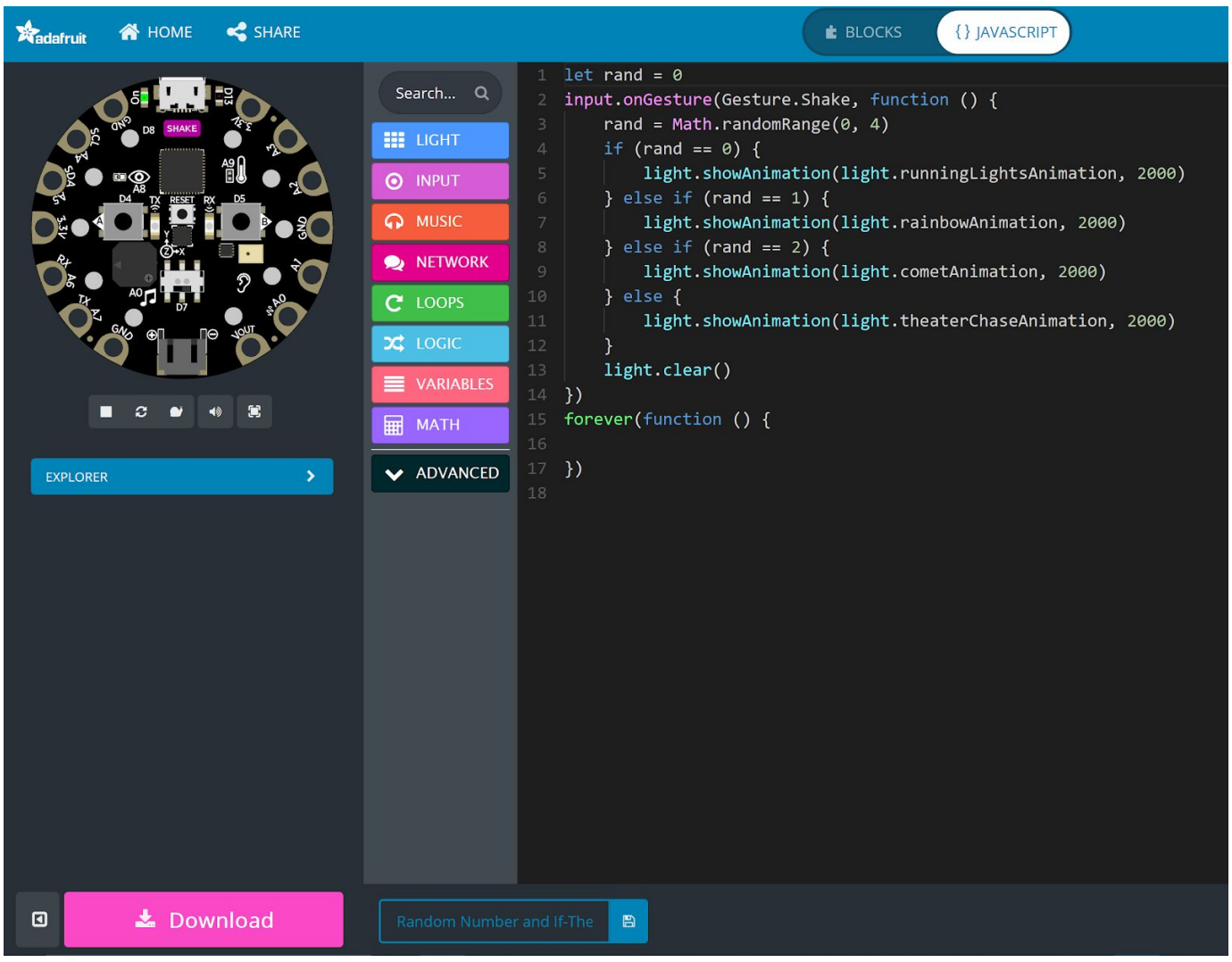
This is the javascript that is generated for the example above.



More conditionals - If/then/else logic

This example shows the expanded use of conditionals and more specifically the if/then/else condition. We also use the random number function to pick a random number and use that number to determine which of our conditions to run. Let's build this program and walk through the logic together.

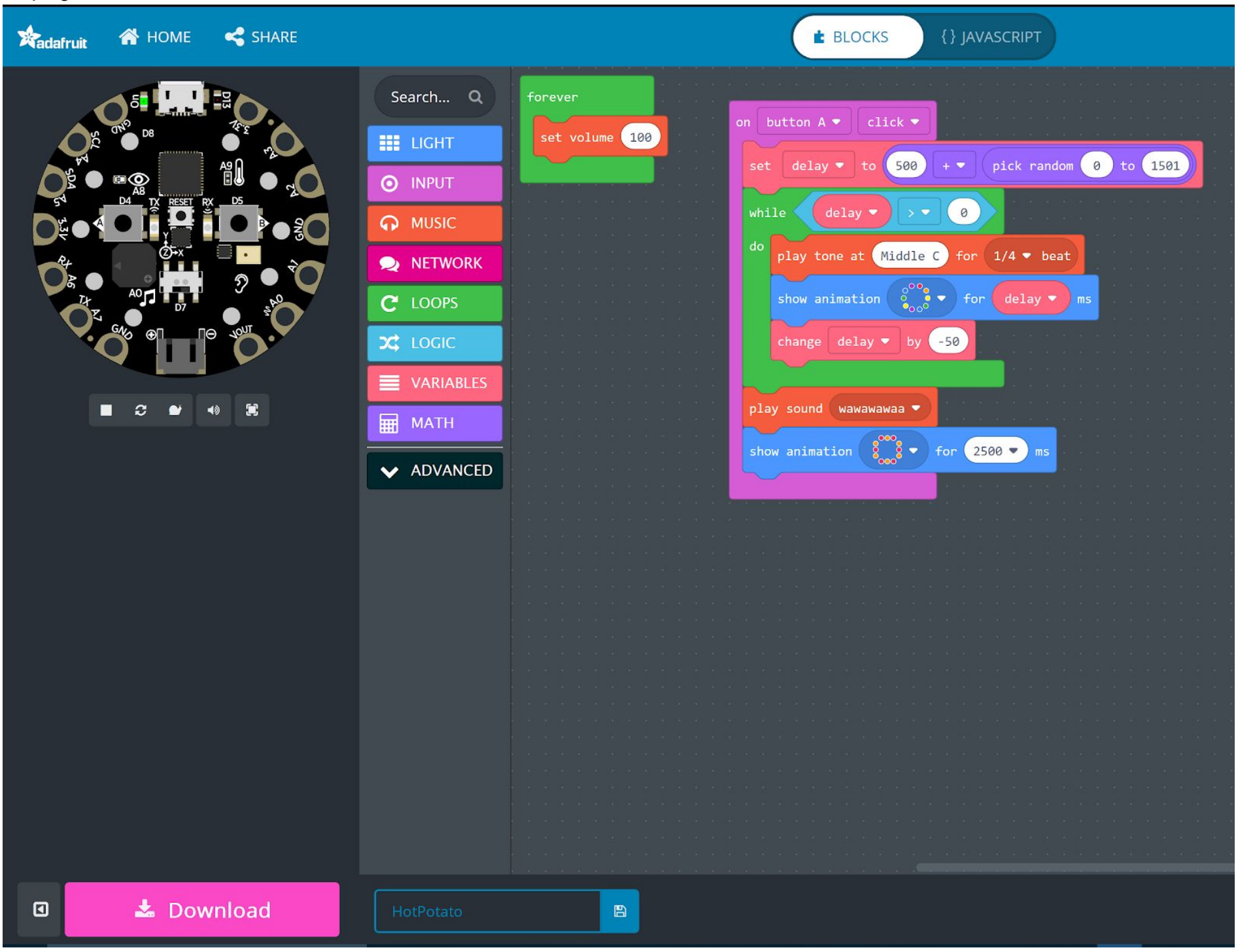


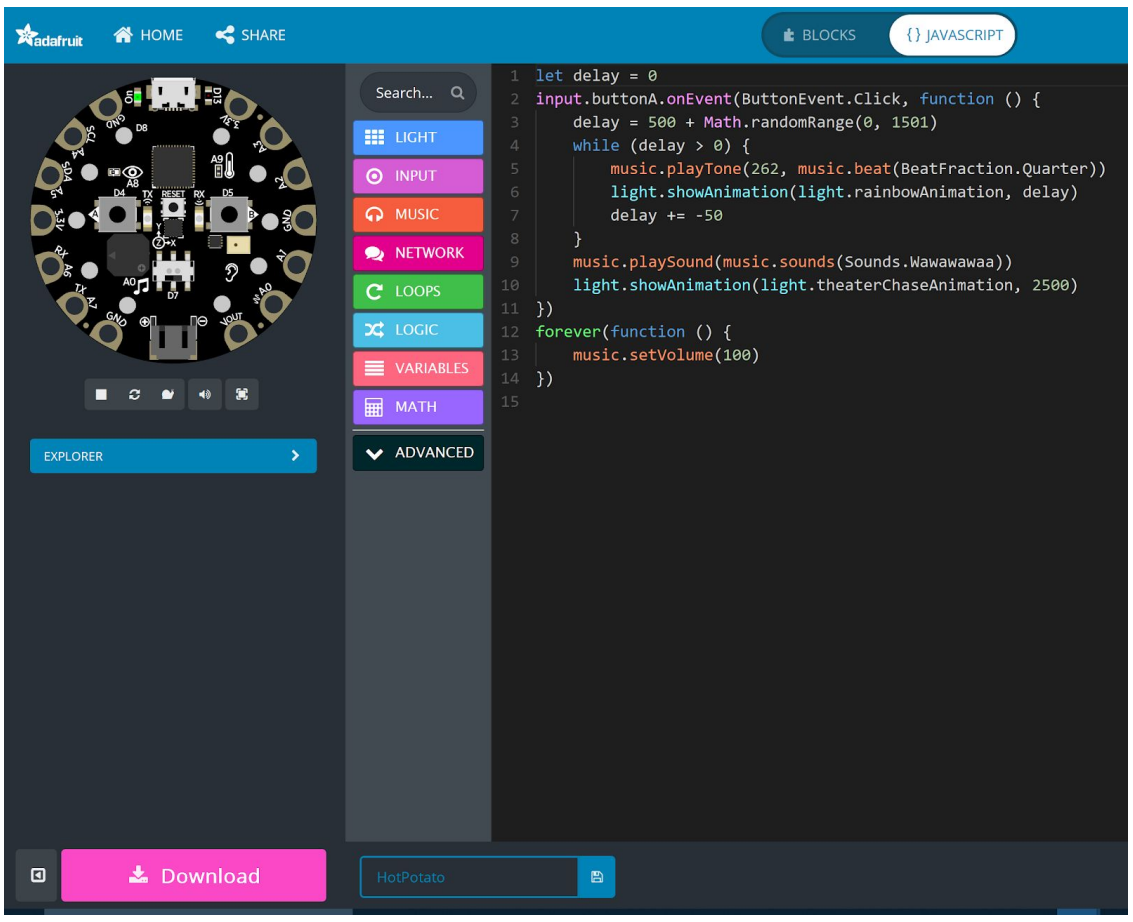


Break Time - 10 Minutes

Hot Potato - Variables, Random Number, Loops, and Luck

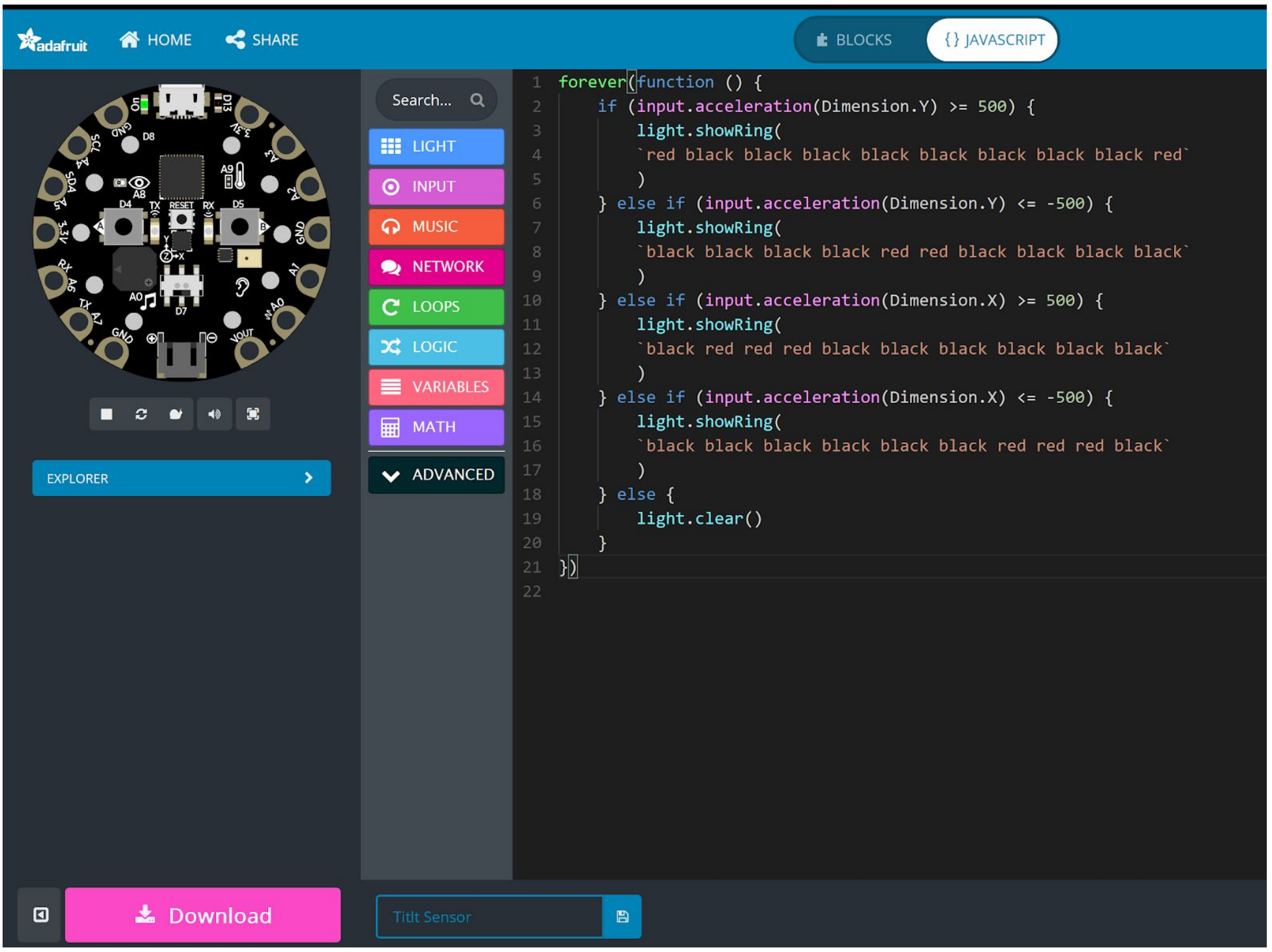
This is a game of hot potato. After we create this program and follow through the logic together we will play a few rounds of the game. This program uses inputs, variables, the random number function, and loops. As we play the game we will modify some of the parameters of the program to see how it affects the program.





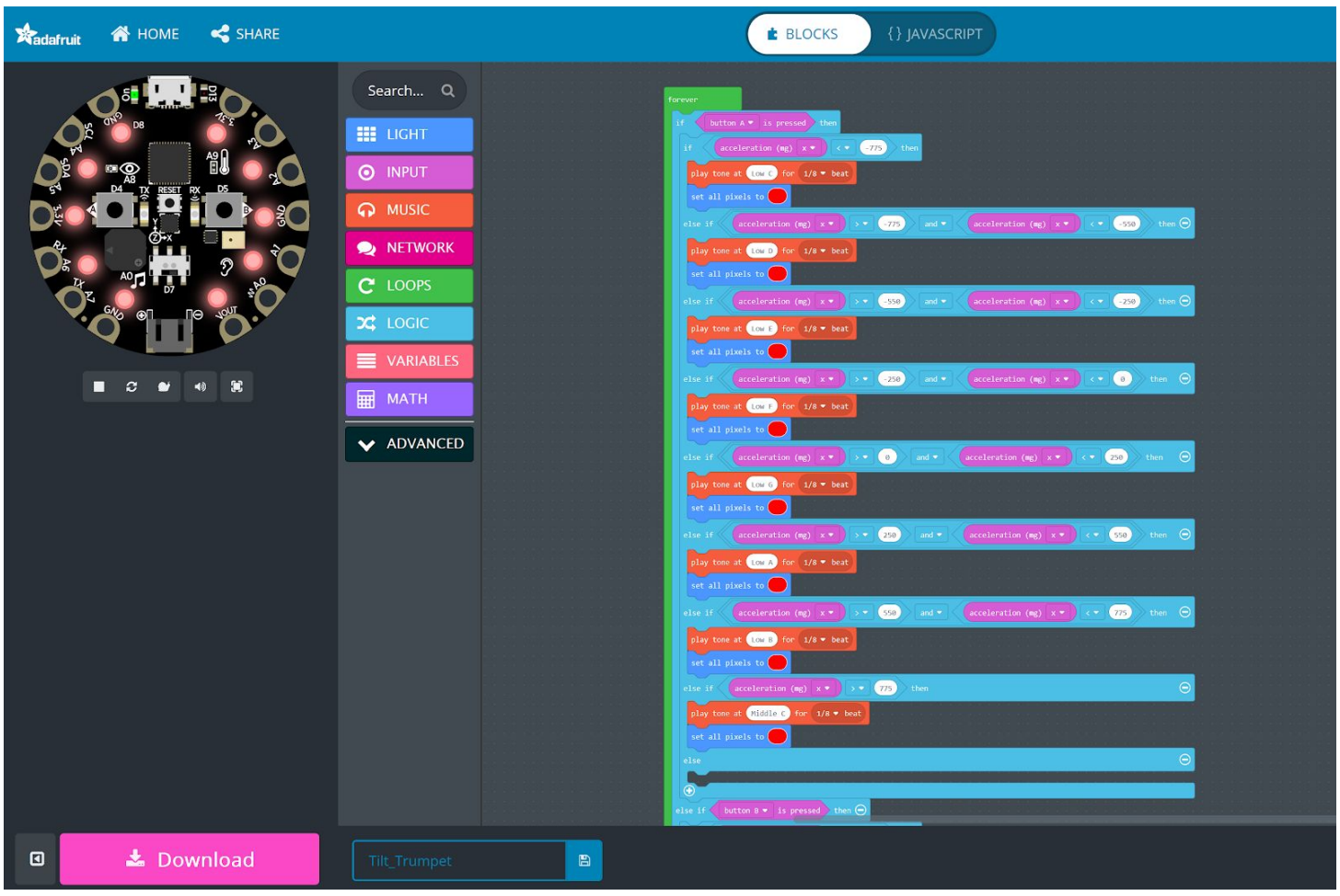
Tilt Sensor - If/Then/Else with hardware input

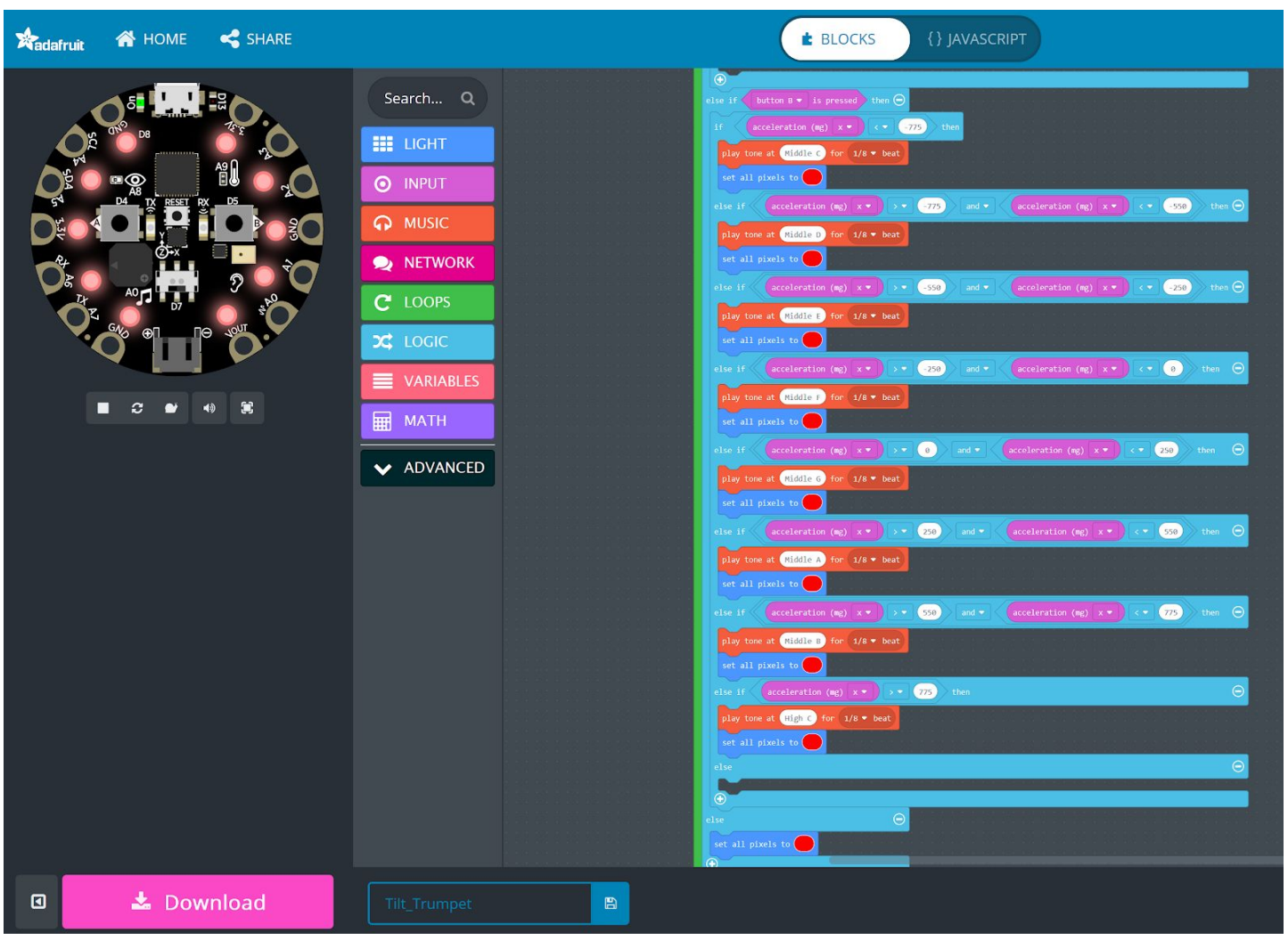
For this exercise we will see if you can follow along with the logic of a program using some of the concepts that we have learned but this time I am not going to give you the block code first. Does anyone think they can tell me what this code will do if we load it on the Circuit Playground?

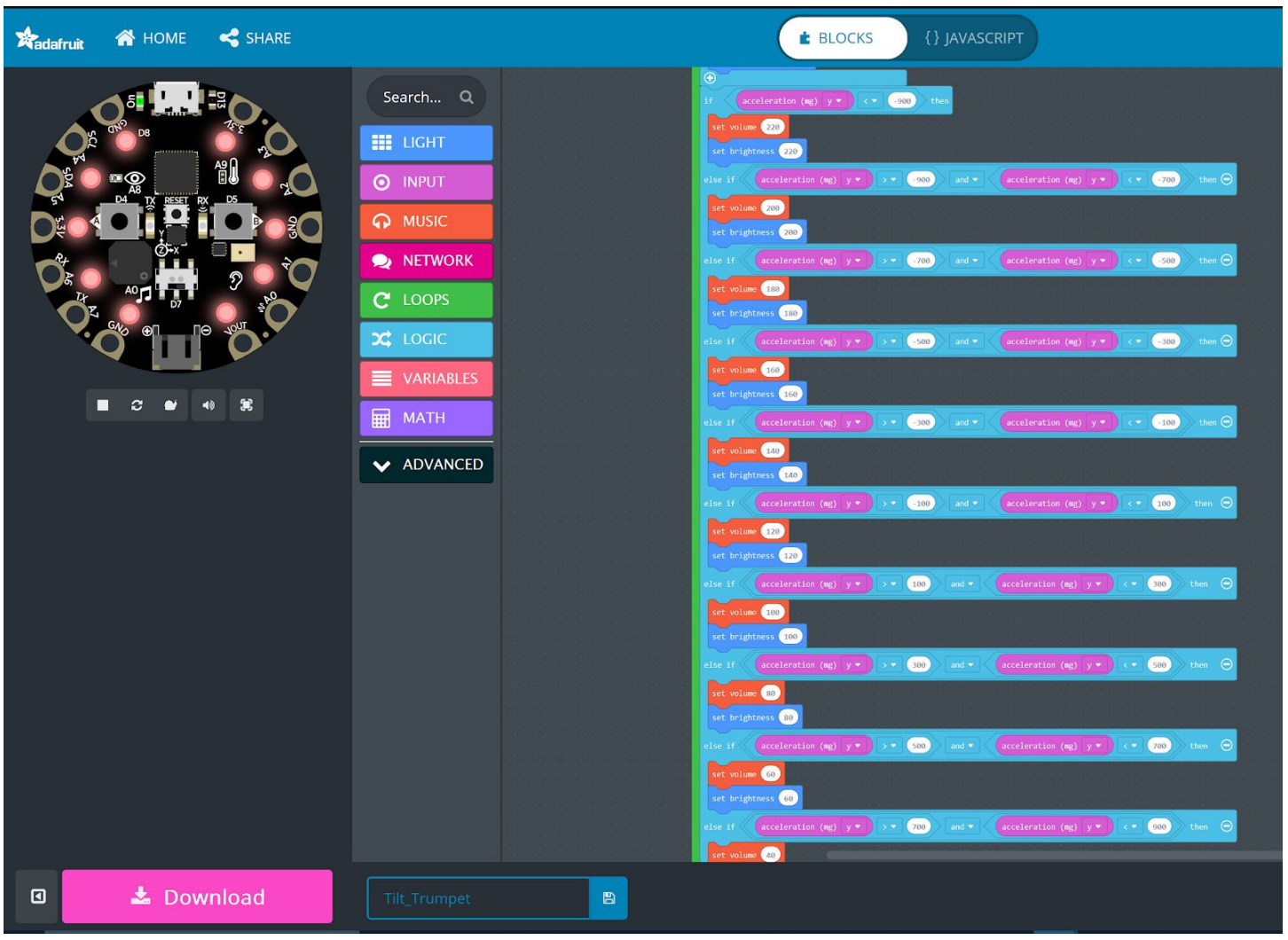


Tilt Trumpet - Nested If/then/else conditionals

This program is one of the examples that Adafruit makes available on their website. This is an example of what is commonly called a nested if. That means that we have two logical if blocks inside of each other. If the first condition is true then the second if block is evaluated and all of the else/if conditions are evaluated as part of that if block. Don't worry we are not going to make you create this by hand. We will load this program from a file and follow along with the logic. Once we are done then we can load the program onto the circuit playground and make some music.



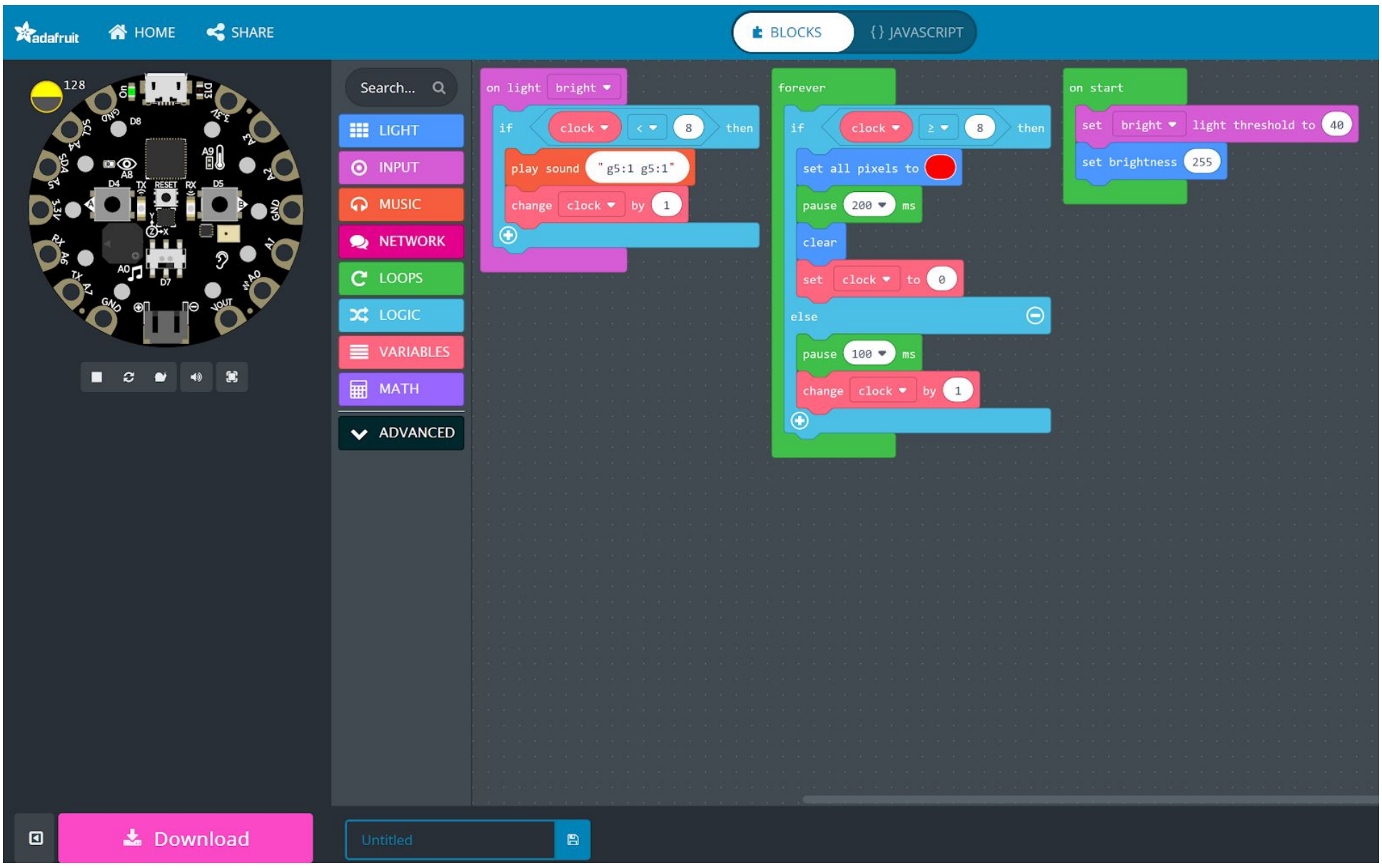


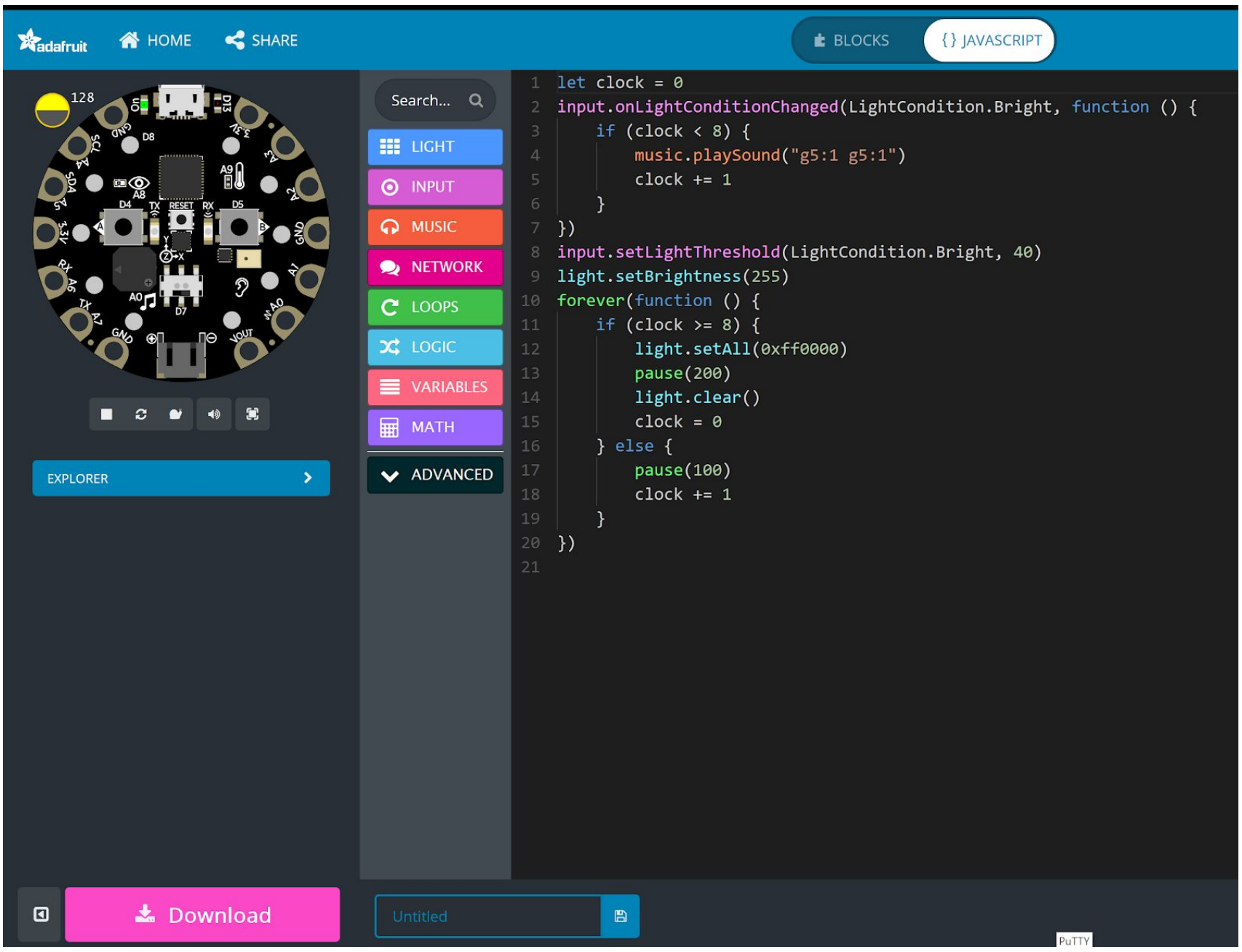

adafruit
HOME
SHARE
BLOCKS
{} JAVASCRIPT
Search...
LIGHT
INPUT
MUSIC
NETWORK
LOOPS
LOGIC
VARIABLES
MATH
ADVANCED
Download
Tilt_Trumpet

Break Time - 10 Minutes

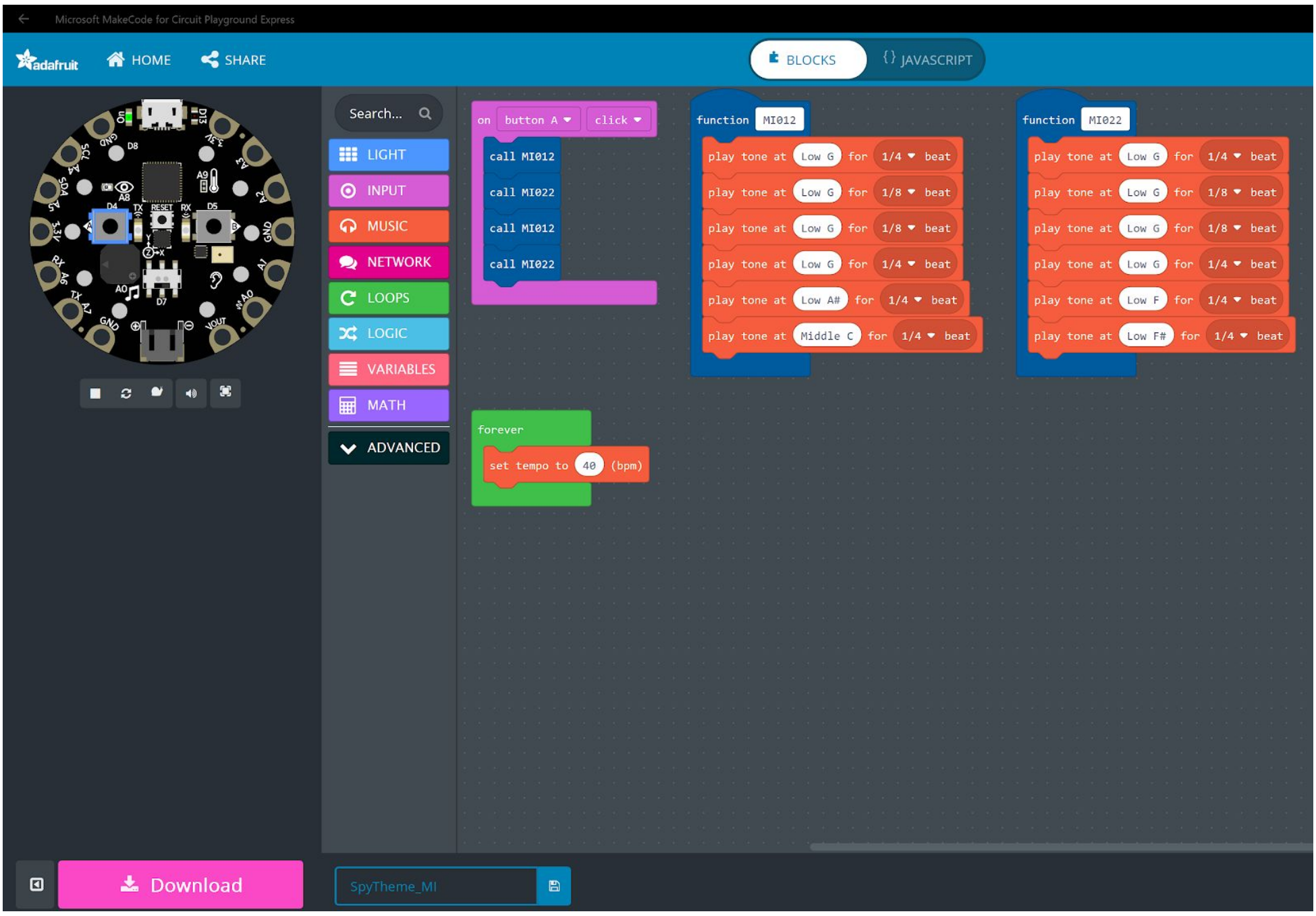
Fireflies - updating a variable based on other inputs

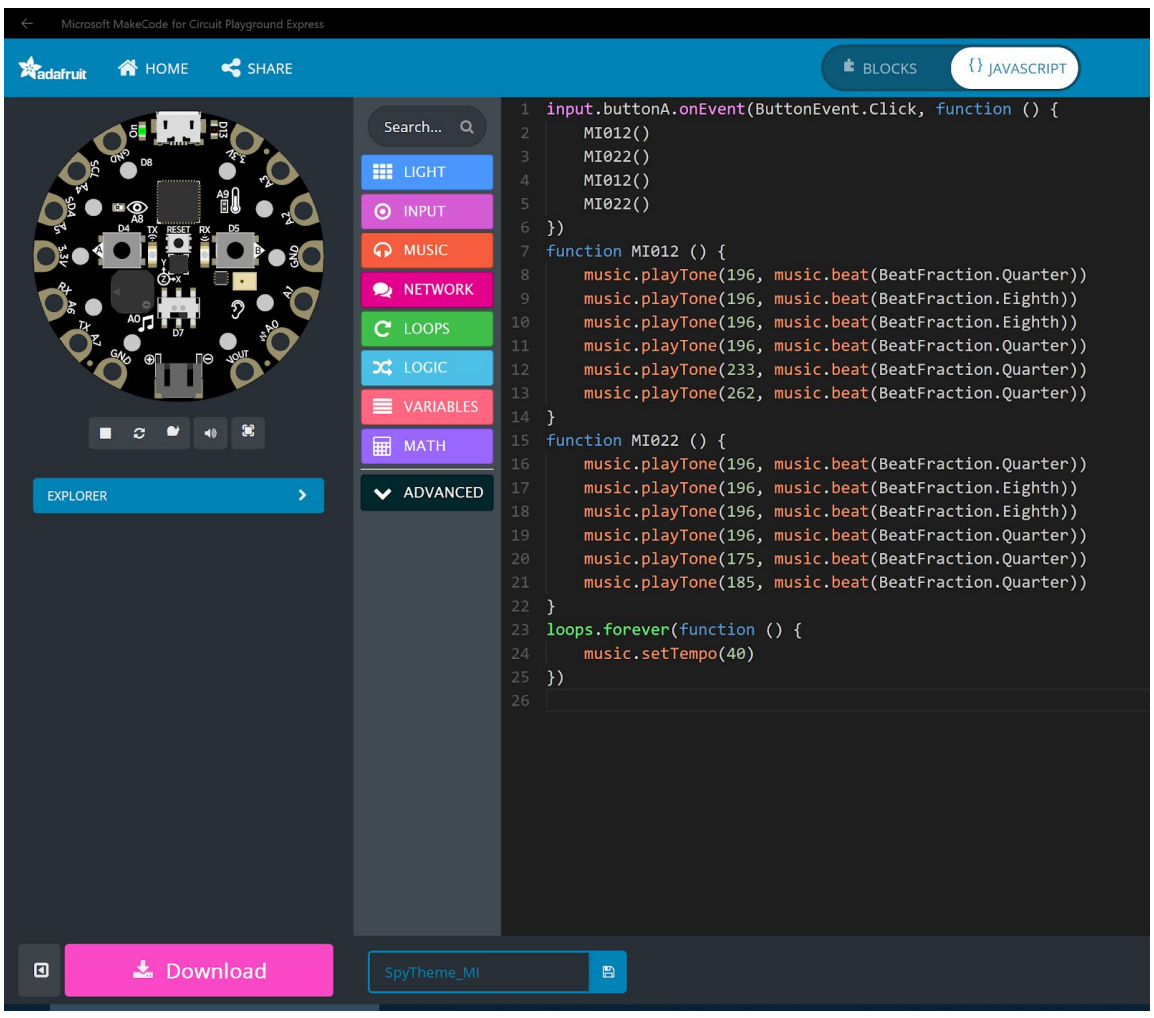
This example is called fireflies and will be interactive and will involve the entire group. In nature if you watch fireflies they will blink and after a while you will notice that they are all blinking at the same rate at the same time. This example will use the lights and the light sensor to slowly adjust a clock of when the “firefly” will blink. After enough time all of your “fireflies” should be blinking at the same time.





Functions - I need to call this more than once



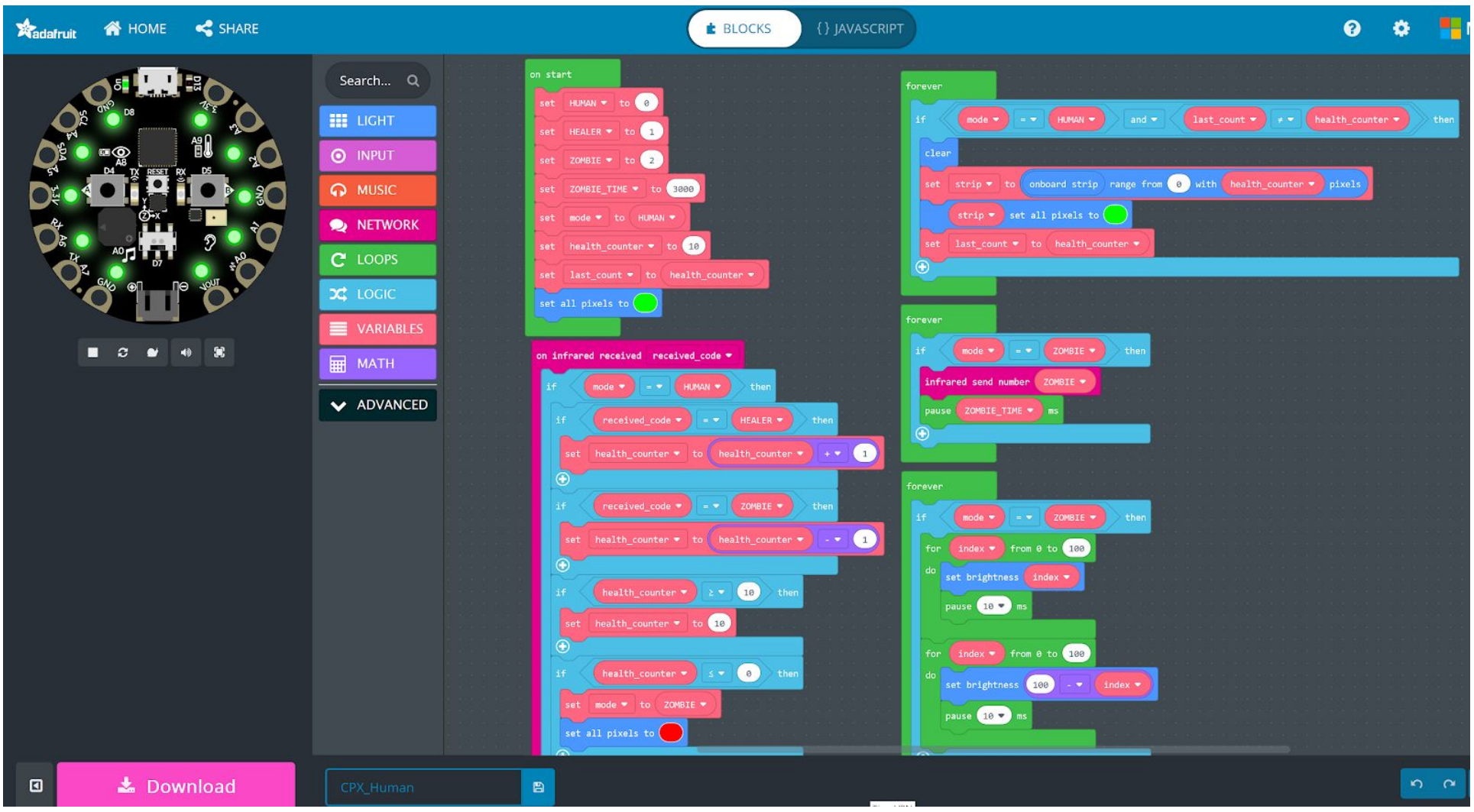


Zombie Game

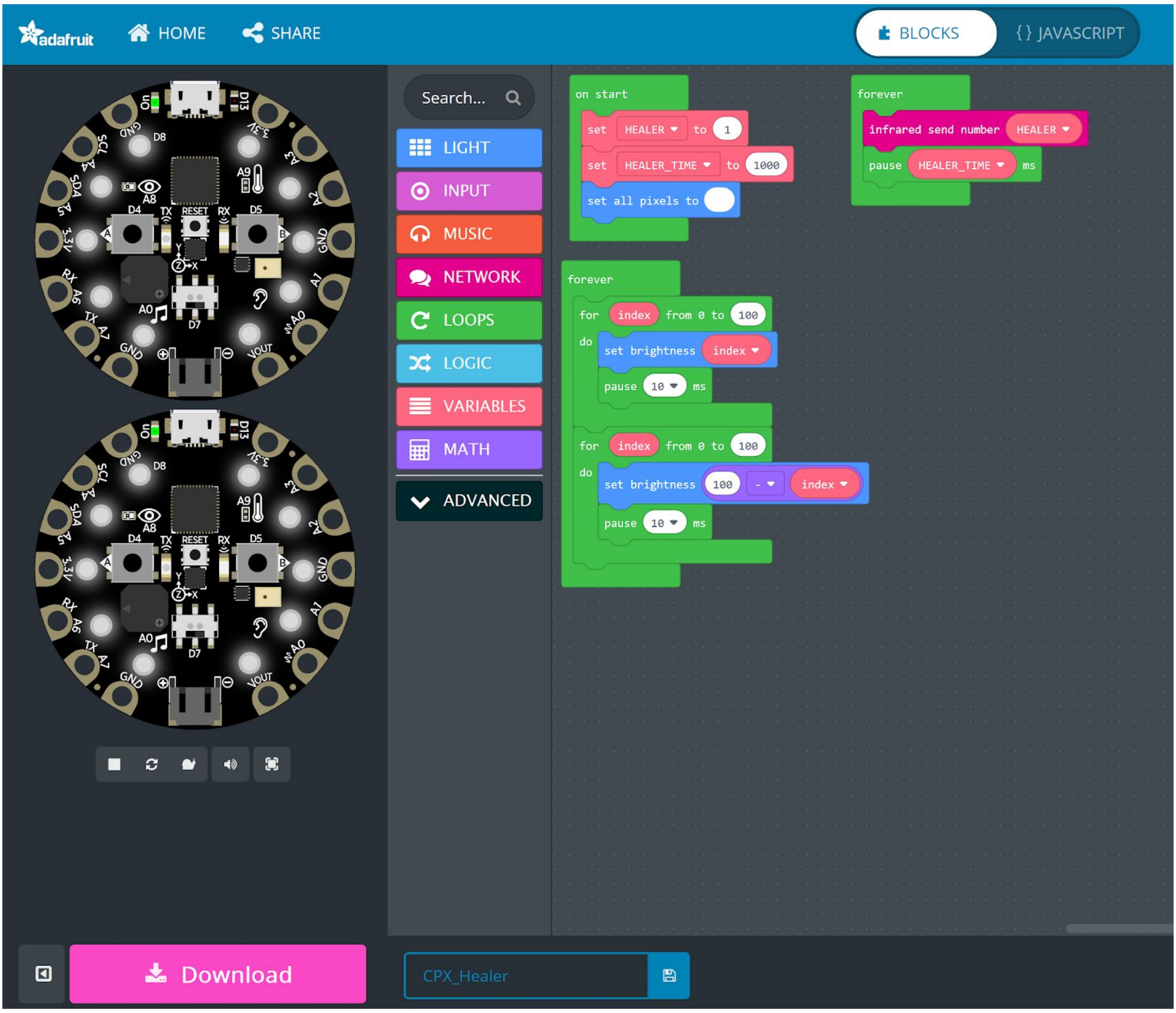
This is a zombie game which is actually three separate programs. The most complicated program is the human. The human has a health which starts at 10. The Human program is always listening for IR signals. The zombie and healer programs are always sending IR signals. The zombie will take away health and the healer will add health. If a human loses all of their health they will become a zombie and will start to send IR signals which will take health away from other humans.

How could we make these programs better?
What could we do to make this one program but still have humans, zombies, and healers?
What concepts that we learned today could be used to combine all three programs into one?

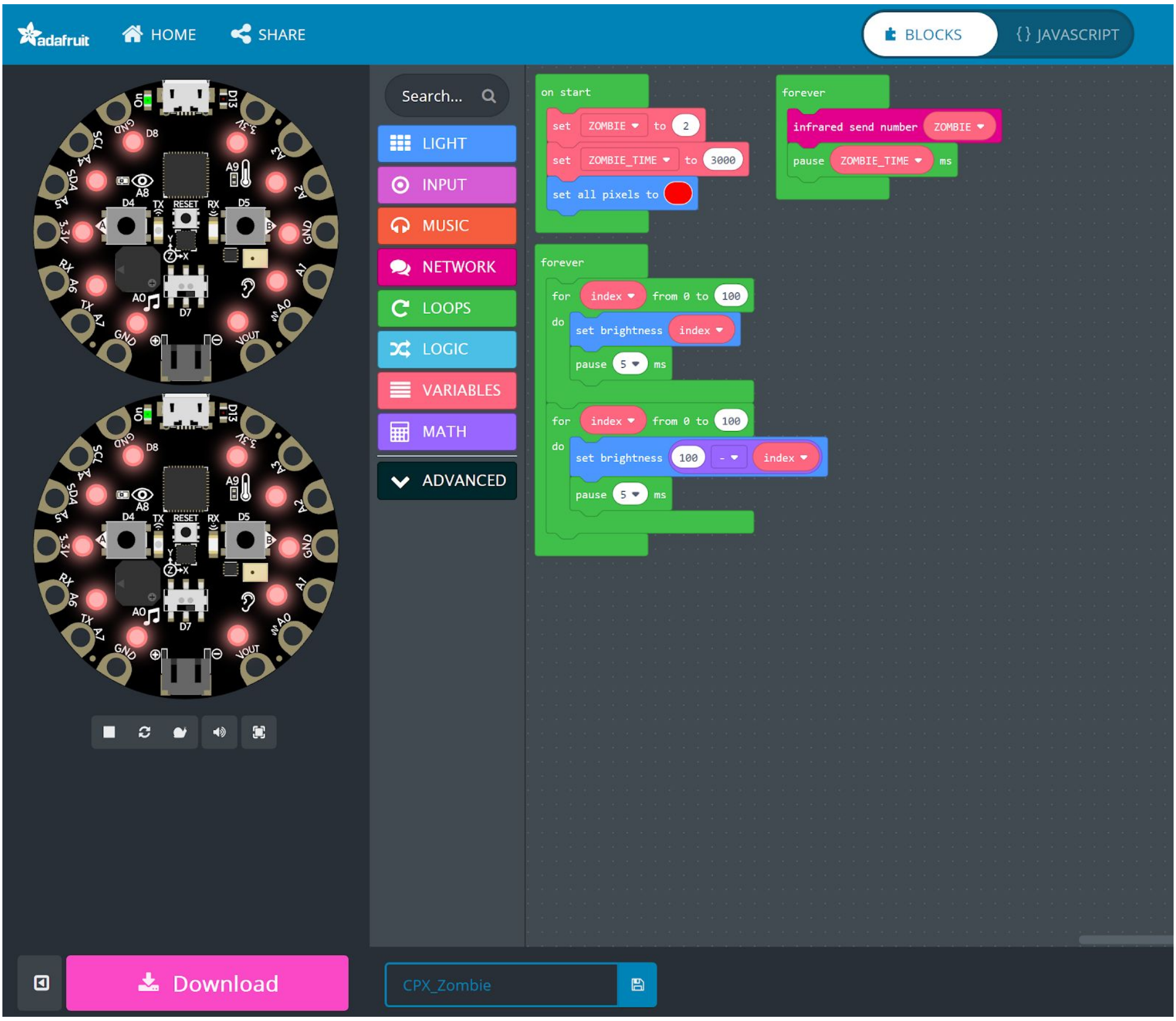
Human



Healer


adafruit
HOME
SHARE
BLOCKS
{ } JAVASCRIPT
Search...
LIGHT
INPUT
MUSIC
NETWORK
LOOPS
LOGIC
VARIABLES
MATH
ADVANCED
on start
set HEALER to 1
set HEALER_TIME to 1000
set all pixels to
forever
infrared send number HEALER
pause HEALER_TIME ms
forever
for index from 0 to 100
do
set brightness index
pause 10 ms
for index from 0 to 100
do
set brightness 100 - index
pause 10 ms
Download
CPX_Healer

Zombie


adafruit
HOME
SHARE
BLOCKS
{} JAVASCRIPT
Search...
LIGHT
INPUT
MUSIC
NETWORK
LOOPS
LOGIC
VARIABLES
MATH
ADVANCED
on start
set ZOMBIE to 2
set ZOMBIE_TIME to 3000
set all pixels to
forever
infrared send number ZOMBIE
pause ZOMBIE_TIME ms
forever
for index from 0 to 100
do
set brightness index
pause 5 ms
for index from 0 to 100
do
set brightness 100 - index
pause 5 ms
Download
CPX_Zombie

Circuit Playground Links

MakeCode Site
https://makecode.adafruit.com/

MakeCode USB connection Site
https://makecode.adafruit.com/?webusb=1#editor

MakeCode Reference for the Circuit Playground
https://makecode.adafruit.com/reference