micro:bit

Artronix Jams

ARTRONIX

micro:bit Movie Makers

How to code and attach the micro:bit to an LED

What you’ll need:

A micro:bit, battery pack and micro USB cable, plus a suitable computer for programming.
1 small servo motor (eg a SG90 3,3v servo)
3 crocodile clip leads.
3 breadboard jumper wires.

Step One: The Basic Code

You’re going to use the Python editor on the micro:bit website for this jam.
Open the editor here: ***http://python.microbit.org/editor.html***

Copy the Python script below to the editor:

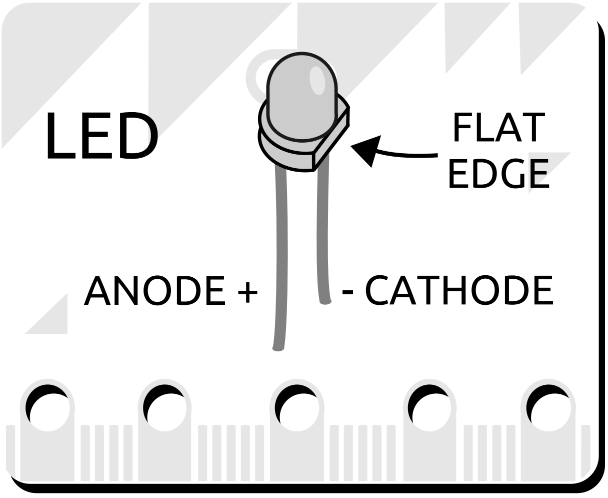
```
# Artronix Jam - Jeepers Bleepers Script 1
# This script blinks 3 LEDs on Pins 0, 1 and 2
# Copy and paste this into the compiler at http://python.microbit.org/editor.html

from microbit import *

# Loop 3 blinking LEDs forever.

while True:

# turn LEDs off.
   pin0.write_digital(0)
   sleep(500)
   pin1.write_digital(0)
   sleep(500)
   pin2.write_digital(0)

   sleep(500)

# turn LEDs on.
   pin0.write_digital(1)
   sleep(500)
   pin1.write_digital(1)
   sleep(500)
   pin2.write_digital(1)

   sleep(500)
```

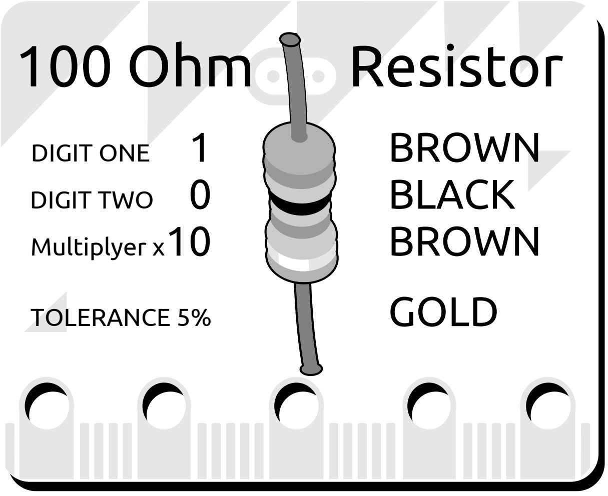
When you're done, download the .hex file and flash it to your micro:bit. Remember to give the script a memorable name!

Step Two: The Components

Lets take a look at the LED.
Like many electronic component, it has to be connected the right way round.

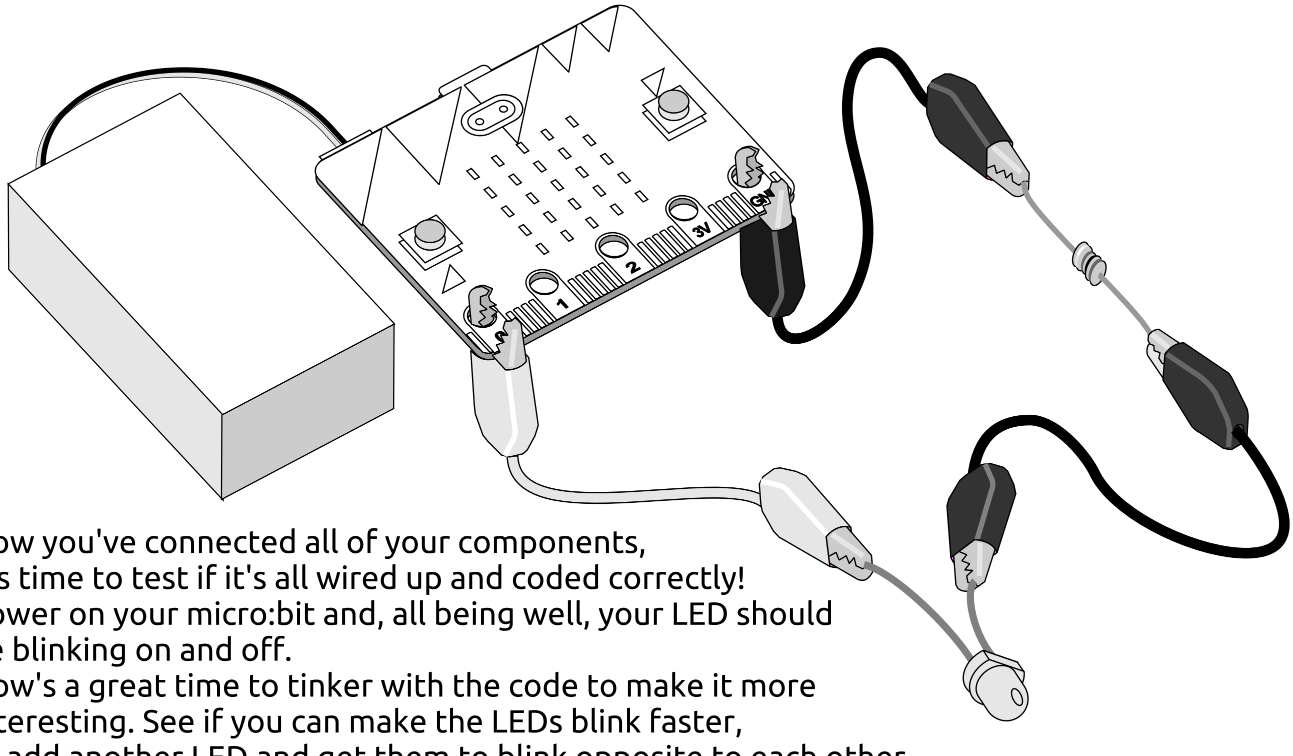
Note that there's a long leg and a wee short leg. The long leg (anode) connects to the positive side. The short leg (cathode) connects to the negative, or Ground side.
Oh, and you'll notice that there's a small flat edge at the base of the LED. This also helps indicate the cathode. Connecting the LED to the power backwards is very likely to cause a puff of smoke to escape and your LED to mysteriously stop working...





Step Three: Wiring Up the Project

First things first make sure your micro:bit is powered down.
Use the crocodile clips to connect the short (negative) leg of the LED to one end of a 100 ohm resistor (it doesn't matter what end). Connect the other end of the resistor to GND on the micro:bit.
Lastly, connect the long leg of the LED to Pin 0 on the micro:bit.
With the circuit complete, current will flow from Pin 0 to GND through the LED you've connected.



Now you've connected all of your components, it's time to test if it's all wired up and coded correctly!
Power on your micro:bit and, all being well, your LED should be blinking on and off.
Now's a great time to tinker with the code to make it more interesting. See if you can make the LEDs blink faster, or add another LED and get them to blink opposite to each other.

Step Four: A More Advanced Code

On, off. On, off... all a bit digital, isn't it? How about a nice pulsing and fading LED?
We can generate a more analogue effect using “pulse width modulation” It's still just turning the LED on and off but it's so fast it looks like the brightness is changing.

If you want to create a pulsing LED effect, disconnect the LEDs and connect the micro:bit to your computer again.

```
# Artronix Jam - Jeepers Bleepers Script 2
# This script makes a single LED pulse on and off using 'pulse width modulation'.
# Copy and paste this into the compiler at http://python.microbit.org/editor.html

from microbit import *

while True:

   for i in range(0,128,1):
      pin0.write_analog(i)
      sleep(10)

   for i in range(128,0,-1):
      pin0.write_analog(i)
      sleep(10)
```

Open the Python editor and replace the script you've previously written with this one, then download the .hex file and update your micro:bit

Step Five: It's Just Step Three Again

Reconnect the LED to Pin 0 as before and power on your micro:bit.
Hopefully you now have a gently pulsing LED.
When you're confident how this new script works, tinker with the code some more to really make it your own.

And don’t forget to share anything you create with us at #artronixjam