LED FIREWALKER CODE

1. These are separate files of code that someone else has already written and that we're going to use, so we have to tell our program that we want to include them

   #include <Firewalker.h>
   #include <Adafruit_NeoPixel.h>

2. Next, we need to define some values that will stay constant throughout the entire program, but that we still want to give a name

   #define N_LEDS 8    // Number of LEDs in the strip
   #define SENSOR_PIN A9   // The pin that the sensor is connected to
   #define LED_PIN 6     // The pin that the LED strip is connected to

   #define STEP_DOWN_THRESHOLD 400    // The highest value of the sensor when you’re definitely stepping down on it
   #define STEP_UP_THRESHOLD 550      // The lowest value of the sensor
   // when you’re definitely not stepping down on it

3. Then we declare our "variables"-- values that might change while the program is running

   boolean stepping = false;     // This "variable" keeps track of whether or not
                                   // you’re stepping down on the sensor
4. Finally, we declare some "objects", which are sort of like variables, but a little more complicated. We can use these objects to access the extra code that we included in step 1.

Firewalker firewalker(SENSOR_PIN, N_LEDS, STEP_UP_THRESHOLD);
// This "object" keeps track of the colors of the LED based on sensor values

Adafruit_NeoPixel strip = // This "object" keeps track of the LED strip
    Adafruit_NeoPixel(N_LEDS, LED_PIN, NEO_GRB + NEO_KHZ800);

5. The setup() "function" is run once, at the beginning of the program

void setup() {
    // Get everything set up
    firewalker.begin();
    strip.begin();
}
Make Your Own Wearables Workshop: Speaker Notes

LED FIREFALKER CODE

6. The loop() "function" is run over and over, as fast as the microcontroller can go, forever, or until the microcontroller is reset or loses power

```cpp
void loop() {
    // Read the sensor value from the analog pin
    firewalker.updateSensorValue();

    // If the sensor value is higher than the step up threshold
    // we are not stepping on it
    if (firewalker.getSensorValue() > STEP_UP_THRESHOLD) {
        stepping = false;
    }

    // If the sensor value is lower than the step down threshold
    // we are stepping on it
    if (firewalker.getSensorValue() < STEP_DOWN_THRESHOLD) {
        stepping = true;
    }
}
```
7. Update the status of the LED colors

    firewalker.updateStep(stepping);
    // Set each LED in the strip to the correct color
    // using a "for loop"!

    for (int i = 0; i < N_LEDS; ++i) {
        strip.setPixelColor(i, firewalker.getLEDColor(i));
    }

    // Show the colors on the LED strip
    strip.show();

    // Slow down the microcontroller a little so you can
    // see the color pattern!
    delayMicroseconds(1500);