Qualcomm
thinkabit lab

Multiple LEDs

What Will You Learn?

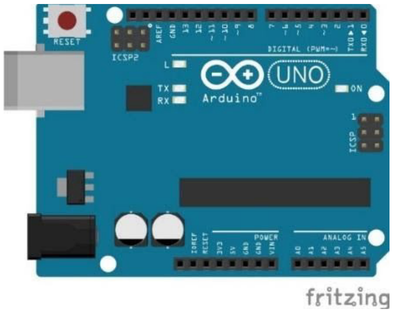
Learn to connect and write code to control multiple LEDs.

Why Should You Learn This?

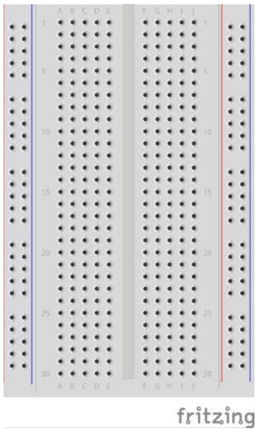
If you thought playing with 1 LED was fun, just wait until you get to play with multiple LEDs! This lesson will walk you through how to wire 4 LEDs and write code to make them behave independently.

Here's What You'll Need:

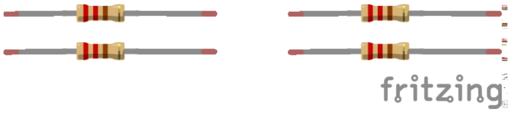
Arduino Microcontroller



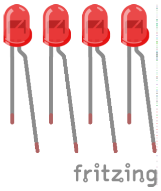
Breadboard



4 resistors (100 – 300 ohms)



4 LEDs (they can be different colors)



1 Black Wire

4 Red Wires

Safety First!

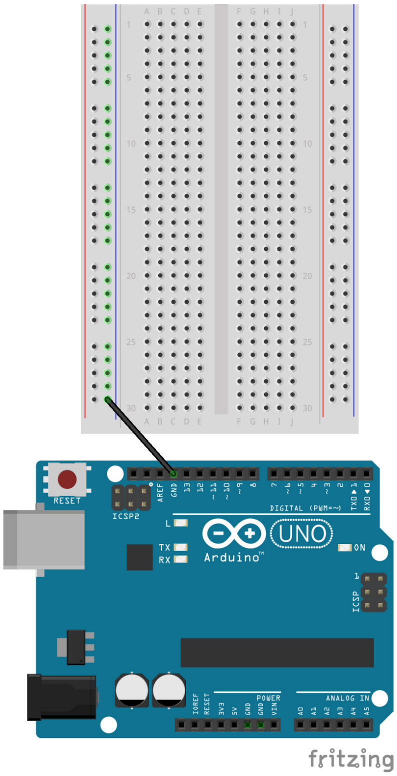
Never allow the red and black wires to touch together while they are connected to a powered Arduino board, as that creates a short circuit. A short circuit can potentially cause the wires and/or the Arduino board to get hot enough to burn the skin. In addition, in the event of a short circuit the Arduino board can potentially catch on fire.

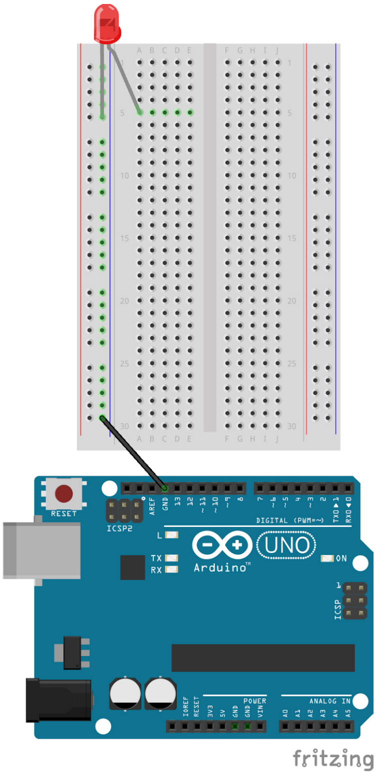
Let’s Get Connected!

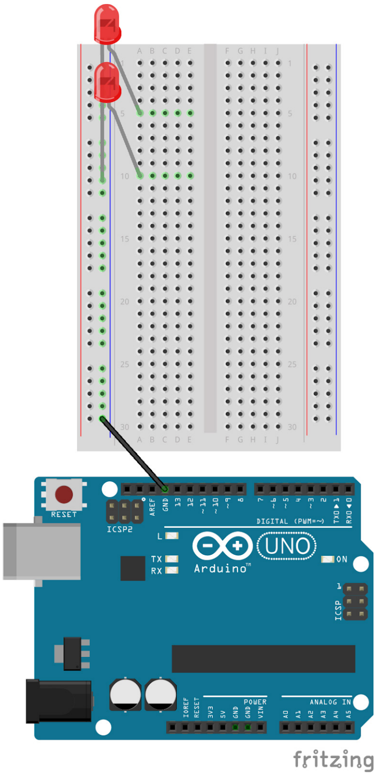
1. Start by connecting the black wire from GND to the negative rail, the blue column, of the breadboard. This makes the entire column negative.

2. Next, start placing LEDs. Start with one LED and place the short leg in the negative rail and the long leg in 5a.

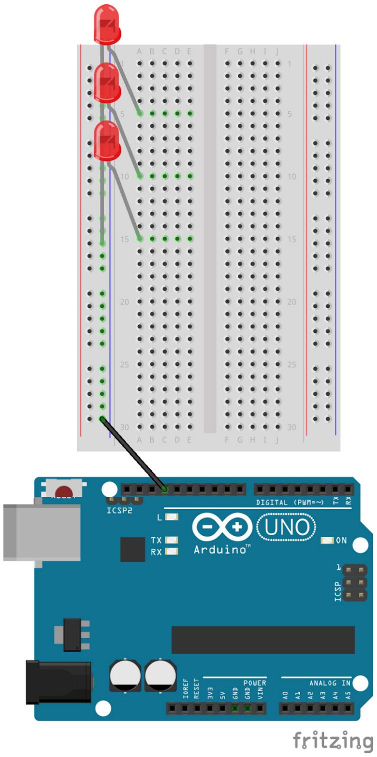
3. With the next LED, place the short leg in the negative rail and the long leg in 10a.



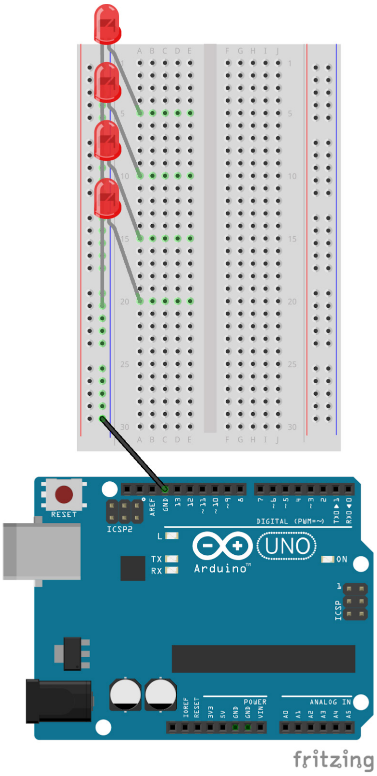




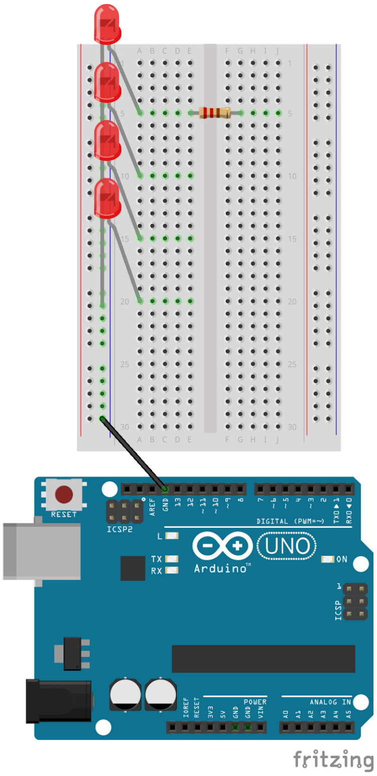
4. Next, place another LED with the short leg in the negative rail and the long in 15a



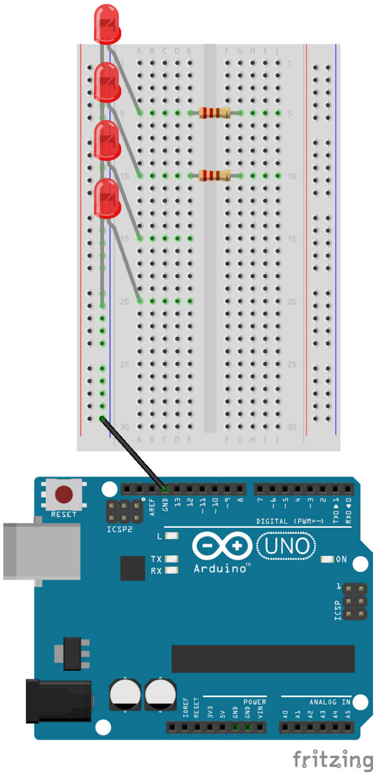
5. Last LED, place the short leg in the negative rail and the long leg in 20a.



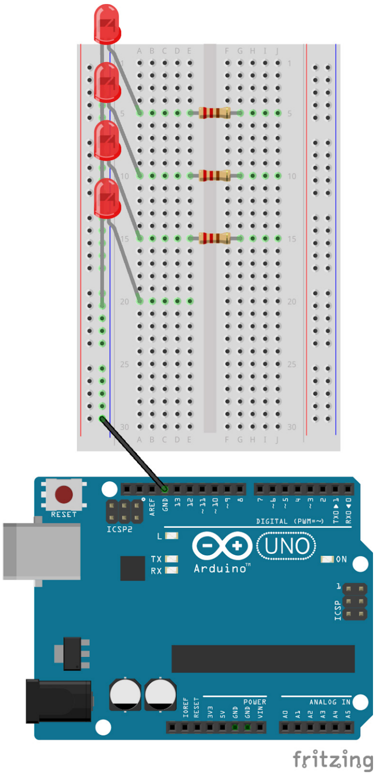
6. Now place the resistors. Start by connecting one resistor from row 5(a-e) across the center aisle to the other row 5 (f-j).



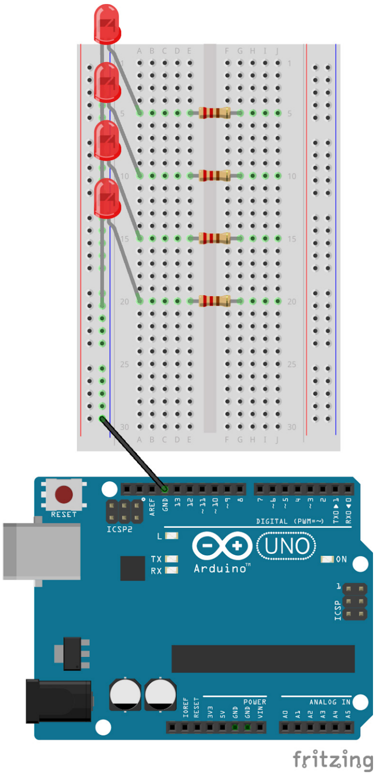
7. Connect the next resistor in row 10 (a-e) across the center aisle to the other row 10 (f-j).



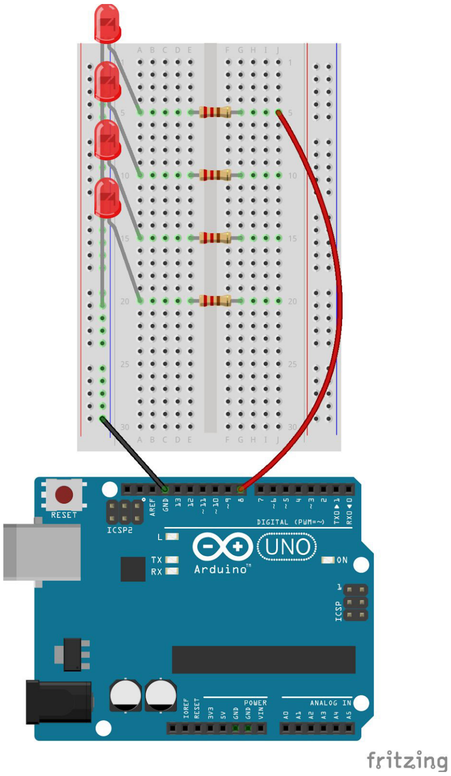
8. Connect the next resistor the same way in row 15.



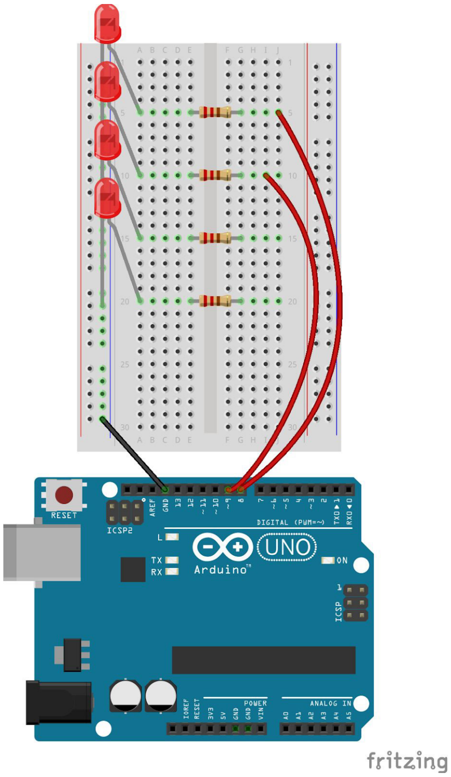
9. Connect the last resistor the same way in row 20.



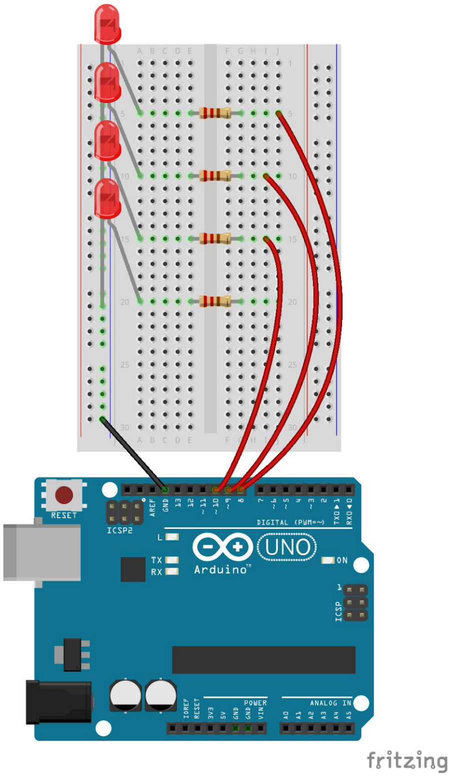
10. Now place the red wires. These will all connect to different pins on the Arduino board so that we can control each LED individually. Start by connecting a red wire from pin 8 on the Arduino to row 5 where only the resistor is (not where the LED and resistor are).



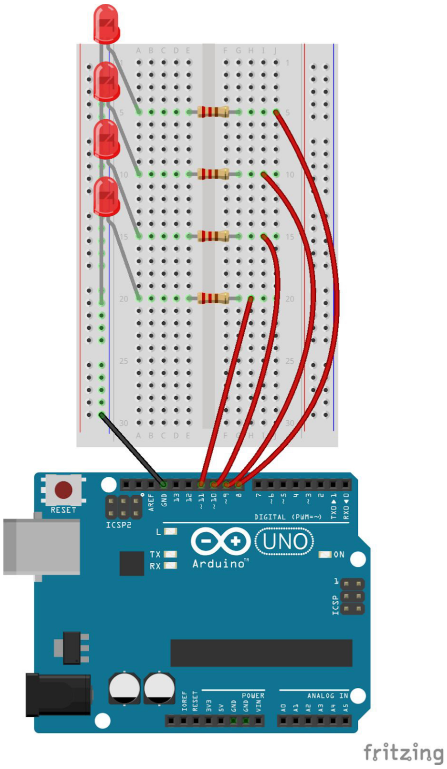
11. Connect a red wire from pin 9 to row 10 where just the resistor is.



12. Connect a red wire from pin 10 to row 15 where just the resistor is.

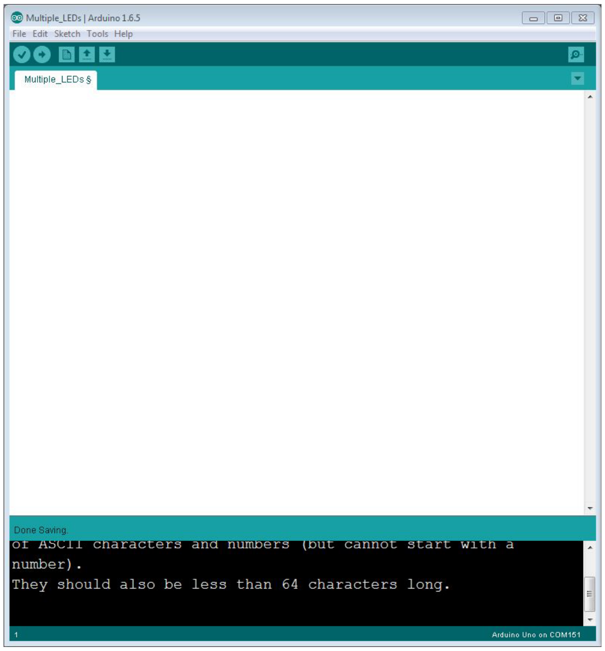


13. Lastly, connect your final red wire from pin 11 to row 20 where just the resistor is.

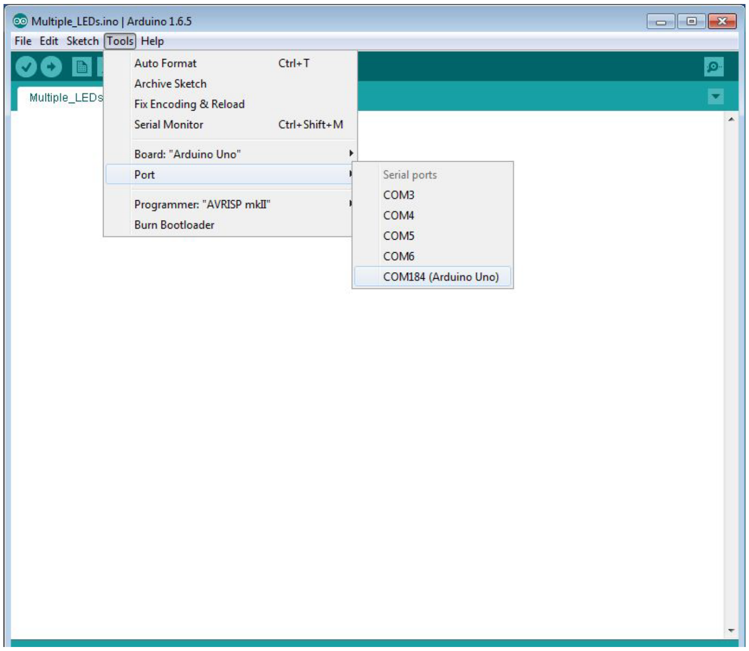


Time to Write Some Code!

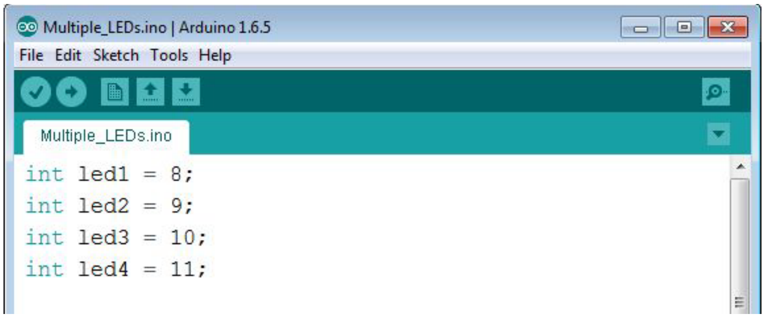
1. Start by connecting the microcontroller to the computer via the USB cable. Open a new sketch and delete all of the text. Also, make sure that you are connected to the correct port.



```
of ASCII characters and numbers (but cannot start with a
number).
They should also be less than 64 characters long.
```

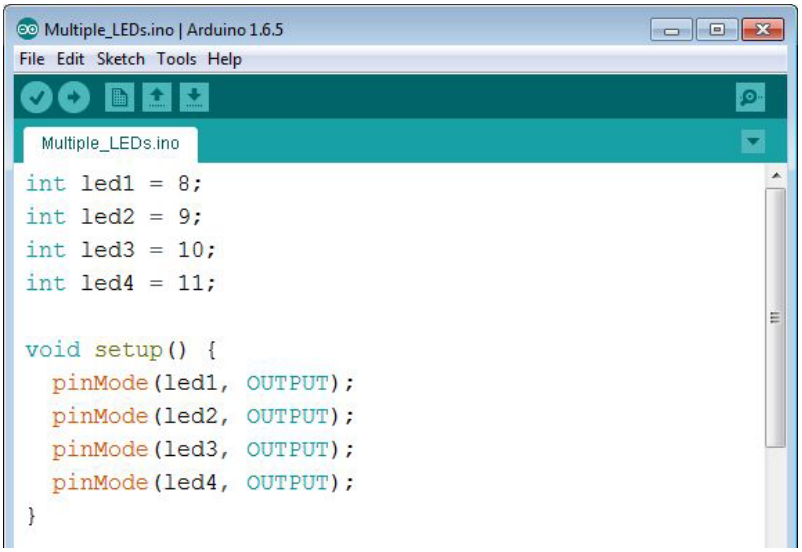


2. The first part of the program is where we declare our variables. In this activity, you will have 4 LEDs (we'll call them **"led1"**, **"led2"**, **"led3"**, and **"led4"**, but if you are using different colors you could call them by their color name) and they are connected to pins 8, 9, 10, and 11, so the first part of your code should look like this:


Multiple_LEDs.ino | Arduino 1.6.5
File Edit Sketch Tools Help
Multiple_LEDs.ino

```
int led1 = 8;
int led2 = 9;
int led3 = 10;
int led4 = 11;
```

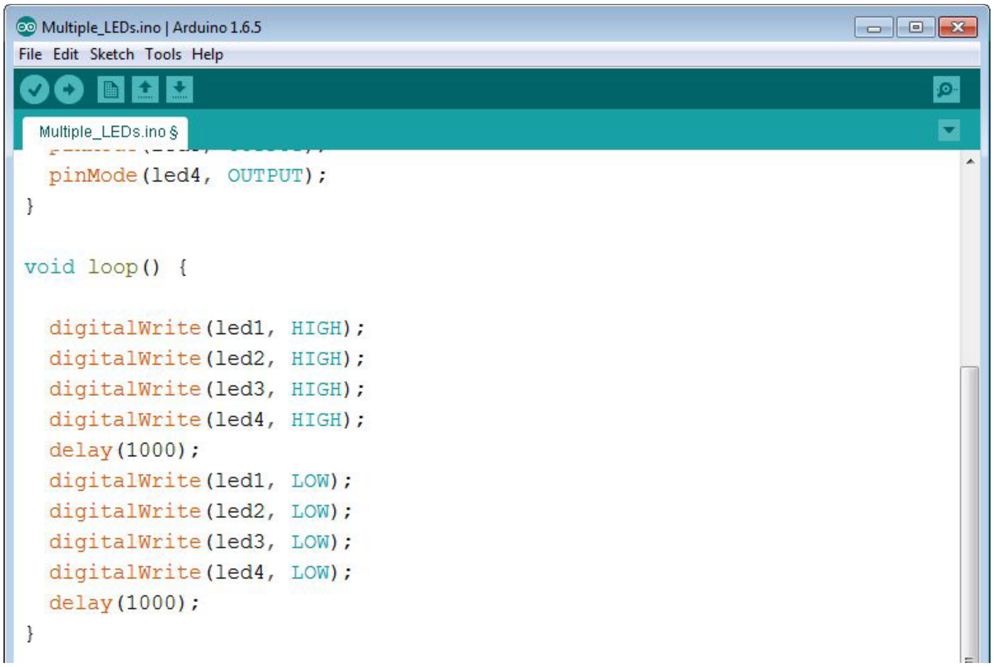
3. Now, you can write the setup function, where you will set up all 4 of these pins as an “OUTPUT” to control your LEDs.


Multiple_LEDs.ino | Arduino 1.6.5
File Edit Sketch Tools Help
Multiple_LEDs.ino

```
int led1 = 8;
int led2 = 9;
int led3 = 10;
int led4 = 11;

void setup() {
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
  pinMode(led3, OUTPUT);
  pinMode(led4, OUTPUT);
}
```

4. Next, write the loop function. The following code will cause all 4 LEDs to turn on at the same time, stay on for one second, and then turn off at the same time and stay off for one second.

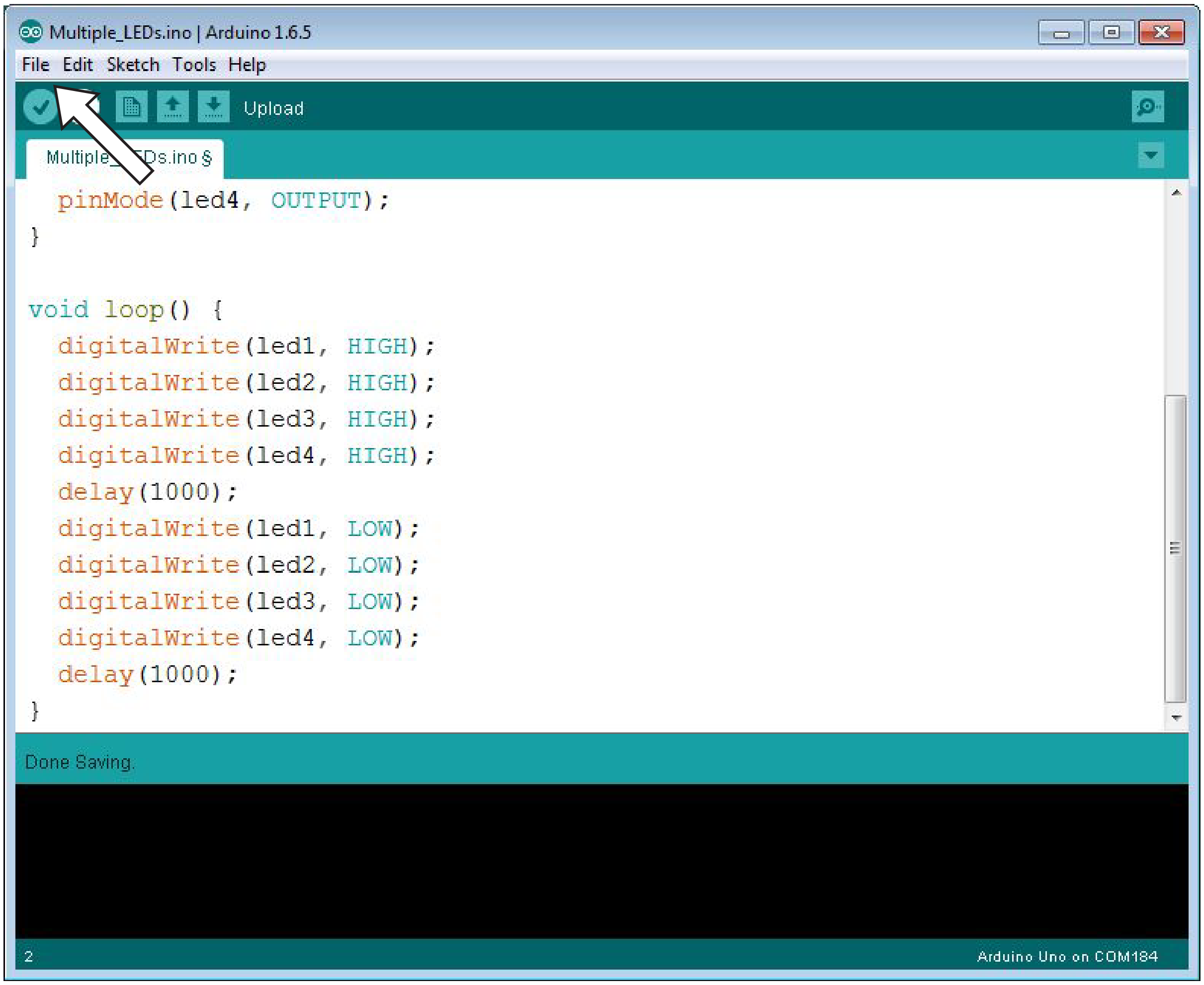

Multiple_LEDs.ino | Arduino 1.6.5
File Edit Sketch Tools Help
Multiple_LEDs.ino §

```
  pinMode(led4, OUTPUT);
}

void loop() {

  digitalWrite(led1, HIGH);
  digitalWrite(led2, HIGH);
  digitalWrite(led3, HIGH);
  digitalWrite(led4, HIGH);
  delay(1000);
  digitalWrite(led1, LOW);
  digitalWrite(led2, LOW);
  digitalWrite(led3, LOW);
  digitalWrite(led4, LOW);
  delay(1000);
}
```

5. Upload your code and your LEDs should all blink together.

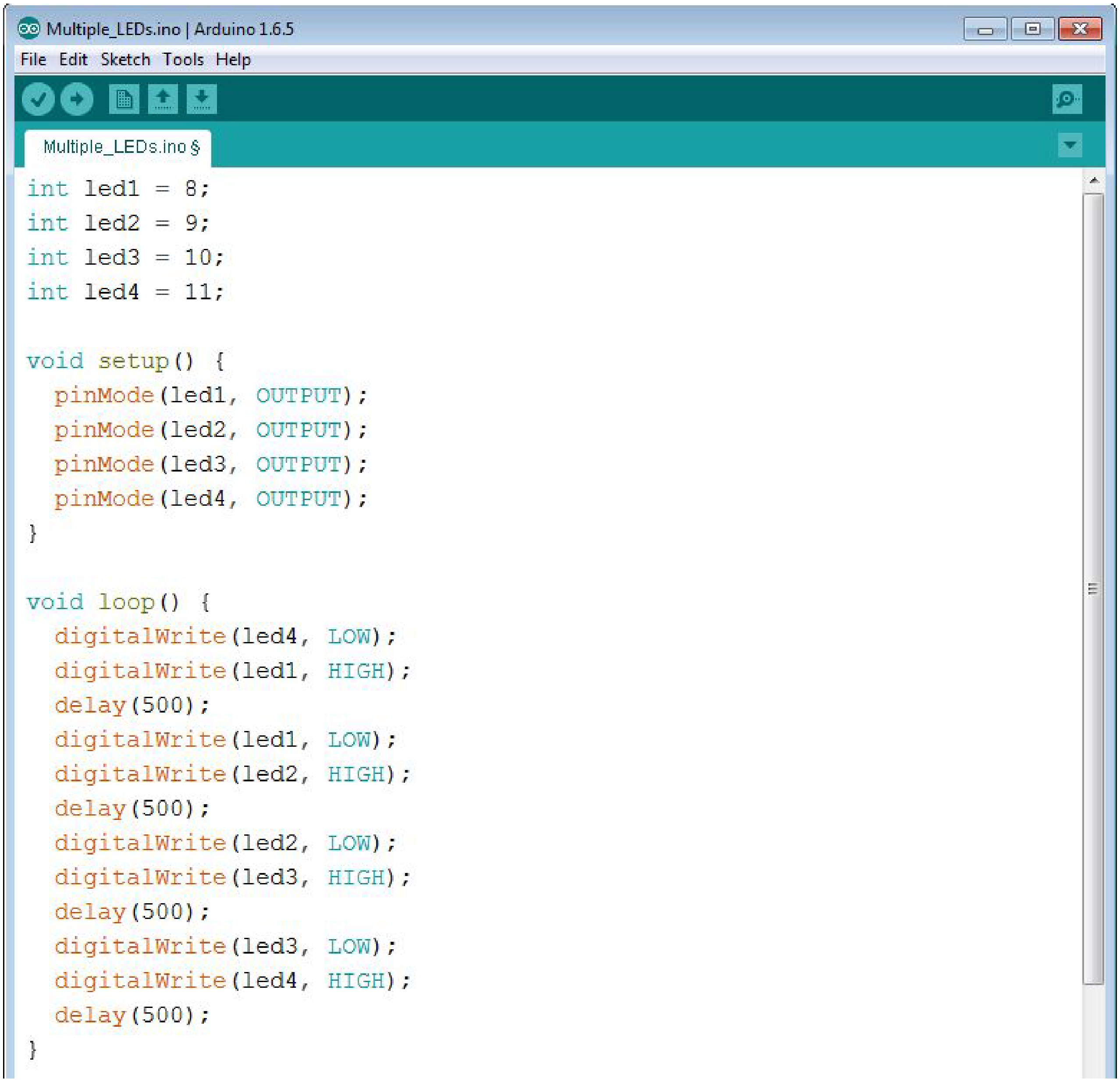

Multiple_LEDs.ino | Arduino 1.6.5
File Edit Sketch Tools Help
Upload
Multiple_LEDs.ino §

```
  pinMode(led4, OUTPUT);
}

void loop() {
  digitalWrite(led1, HIGH);
  digitalWrite(led2, HIGH);
  digitalWrite(led3, HIGH);
  digitalWrite(led4, HIGH);
  delay(1000);
  digitalWrite(led1, LOW);
  digitalWrite(led2, LOW);
  digitalWrite(led3, LOW);
  digitalWrite(led4, LOW);
  delay(1000);
}
```

Done Saving.

6. Now, change your code to make the LEDs blink in a row. This means that LED 1 will turn on and then when it turns off, LED 2 will turn on and so forth. The code shown below uses a delay of 500, or half a second, but you could use a value that is larger or smaller if you prefer. Don't forget to upload your code!


Multiple_LEDs.ino | Arduino 1.6.5
File Edit Sketch Tools Help
Multiple_LEDs.ino §

```
int led1 = 8;
int led2 = 9;
int led3 = 10;
int led4 = 11;

void setup() {
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
  pinMode(led3, OUTPUT);
  pinMode(led4, OUTPUT);
}

void loop() {
  digitalWrite(led4, LOW);
  digitalWrite(led1, HIGH);
  delay(500);
  digitalWrite(led1, LOW);
  digitalWrite(led2, HIGH);
  delay(500);
  digitalWrite(led2, LOW);
  digitalWrite(led3, HIGH);
  delay(500);
  digitalWrite(led3, LOW);
  digitalWrite(led4, HIGH);
  delay(500);
}
```

Note: The first line of the loop programs the Arduino to turn LED 4 off. While this doesn't make much sense at first (since it wasn't on to begin with) if you look at the end of the loop, you will notice that LED 4 is HIGH, or on. If you didn't include the first line of the loop that turns off LED 4, it would stay on after it ran through the loop once.

Tips for Success!

- Common errors that cause upload errors include forgetting the semicolon at the end of each line or spelling something wrong.

- Check that you are connected to the right port by going to Tools, and then Port. This was done in Step 1 of writing the code, but if you saved your code and tried to upload another day, it may not be connected.

- If your code is uploaded to your microcontroller, but your LEDs are not behaving, check the wiring. It may be easiest to pull out everything attached to your breadboard and start over. LEDs can burn out so you may need to test your LEDs.

Are You Ready for Some Challenges?

1. Try to make the LEDs blink in sequence like the code from step 6 (the next LED turns on when the current LED turns off), but this time blink down (1,2,3,4) and then back (4,3,2,1) and repeat.

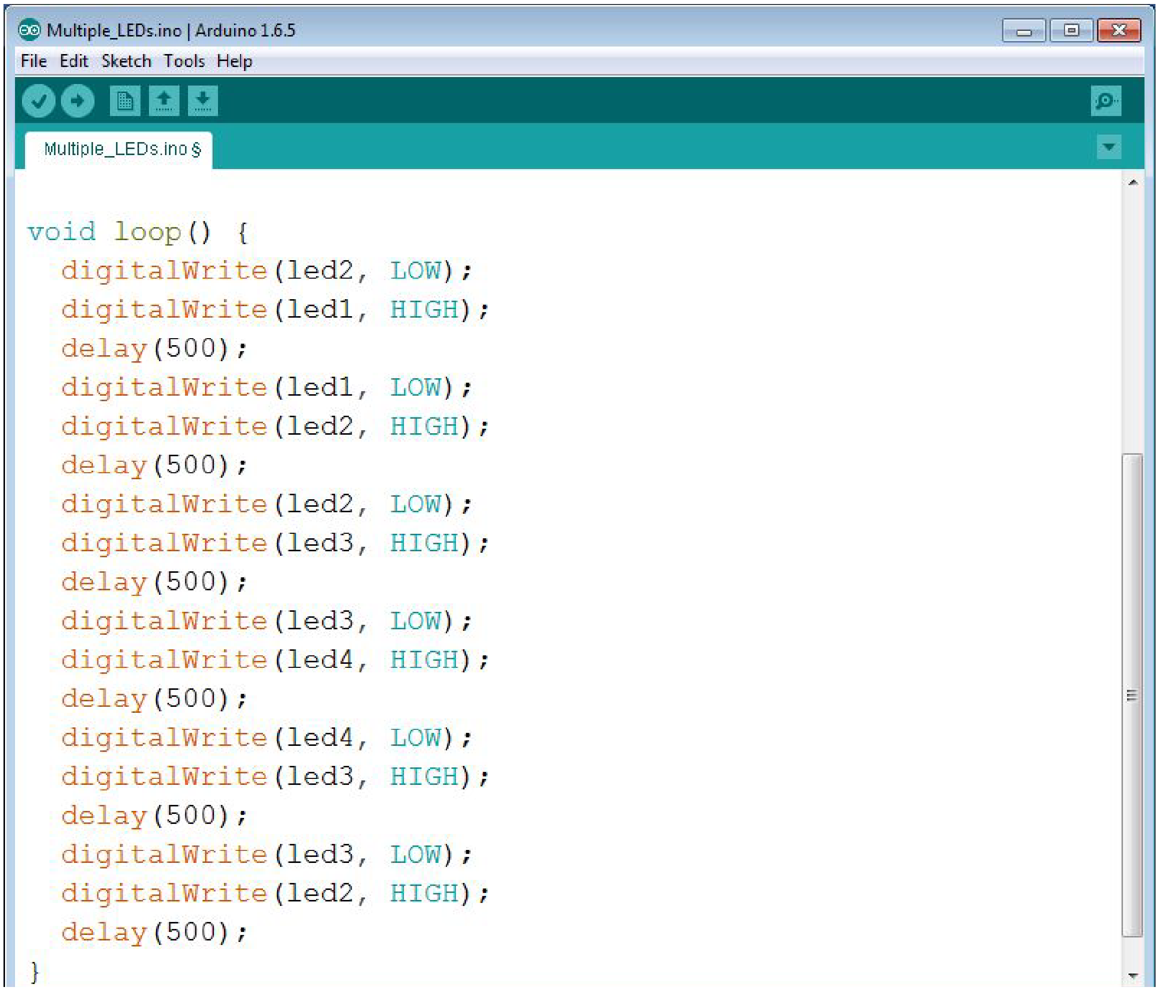
2. Try to make the LEDs turn on, and stay on, as each additional LED is turned on and once all four LEDs are on, all the LEDs turn off and the pattern starts over (1 LED on, 2 LEDs on, 3 LEDs on, 4 LEDs on, 0 LEDs on).

3. Try to program your LEDs to blink in time with a song. You may want to start with something simple like "Twinkle, Twinkle Little Star."

4. Using 3 LEDs (one green, one yellow, one red), try to make a stop light such that the green LED stays on for a long amount of time, then the yellow LED turns on for a very short amount of time, and then the red LED turns on for a long amount of time. Try to make it realistic like a traffic light.

Challenge Solutions:

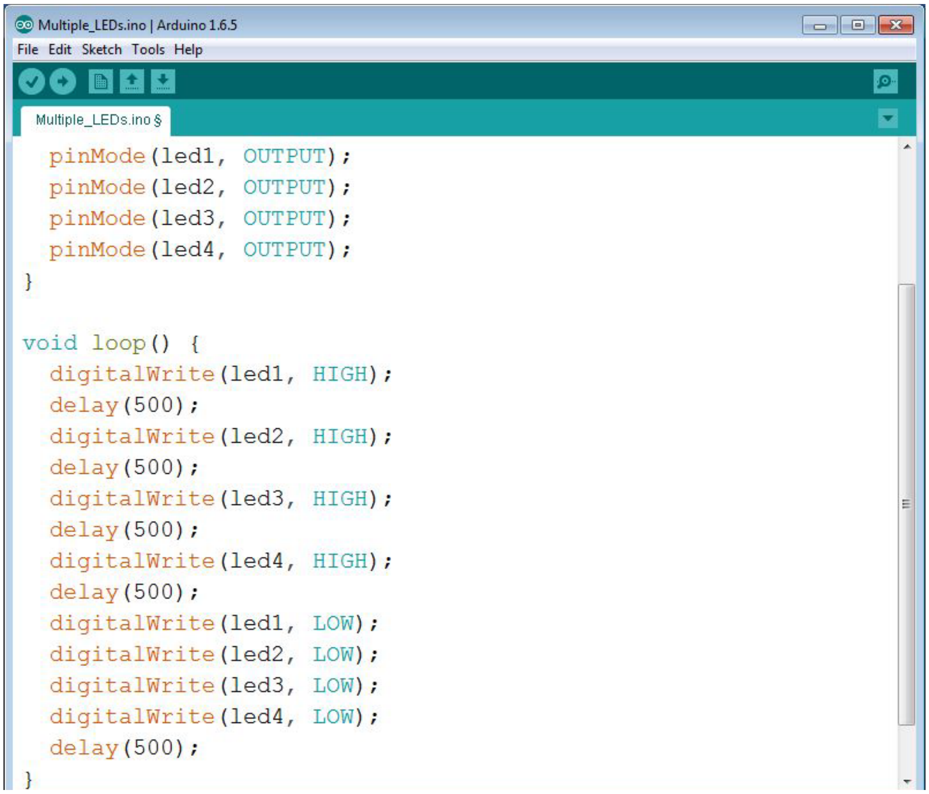
Challenge 1 Solution:



Multiple_LEDs.ino | Arduino 1.6.5
File Edit Sketch Tools Help
Multiple_LEDs.ino §

```
void loop() {
  digitalWrite(led2, LOW);
  digitalWrite(led1, HIGH);
  delay(500);
  digitalWrite(led1, LOW);
  digitalWrite(led2, HIGH);
  delay(500);
  digitalWrite(led2, LOW);
  digitalWrite(led3, HIGH);
  delay(500);
  digitalWrite(led3, LOW);
  digitalWrite(led4, HIGH);
  delay(500);
  digitalWrite(led4, LOW);
  digitalWrite(led3, HIGH);
  delay(500);
  digitalWrite(led3, LOW);
  digitalWrite(led2, HIGH);
  delay(500);
}
```

Challenge 2 Solution:


Multiple_LEDs.ino | Arduino 1.6.5
File Edit Sketch Tools Help
Multiple_LEDs.ino §

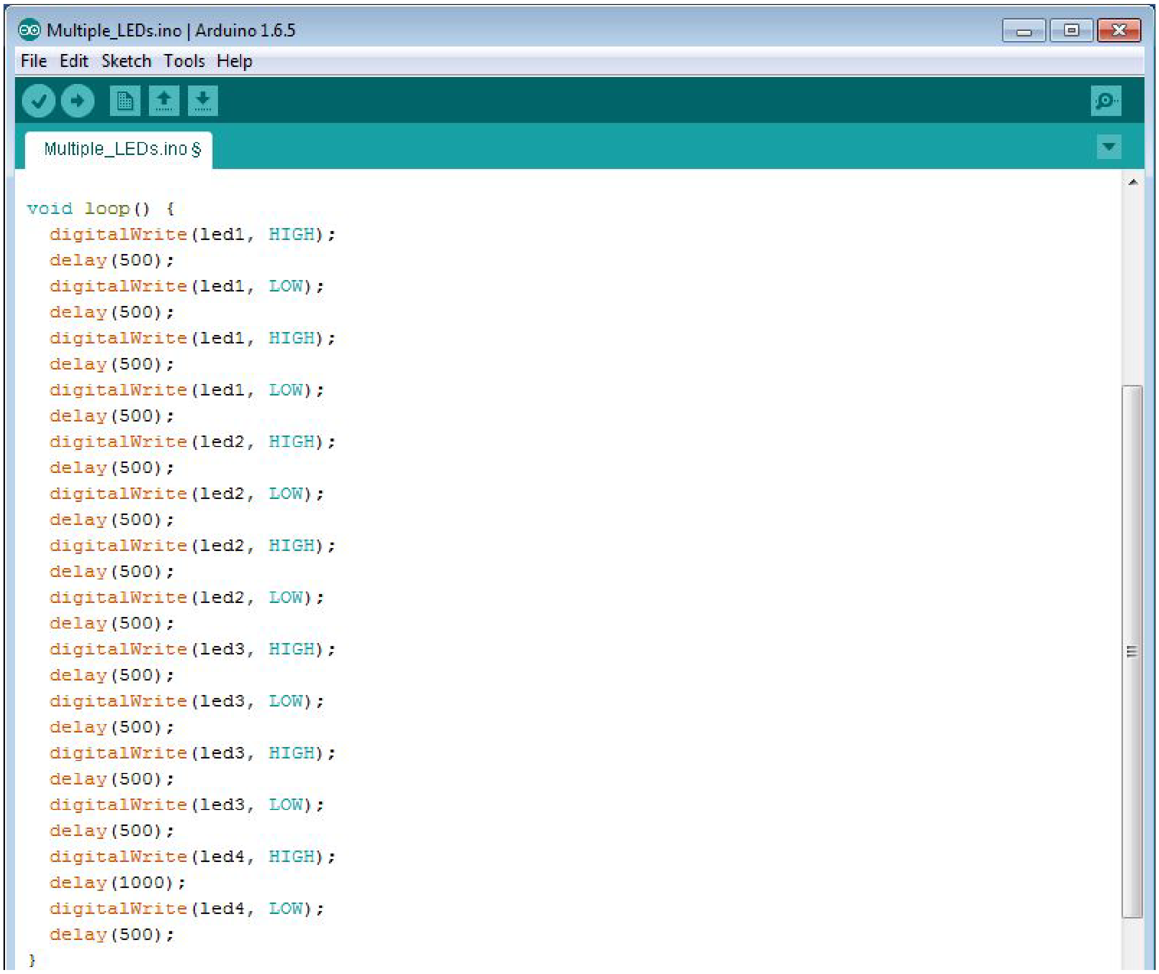
```
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
  pinMode(led3, OUTPUT);
  pinMode(led4, OUTPUT);
}

void loop() {
  digitalWrite(led1, HIGH);
  delay(500);
  digitalWrite(led2, HIGH);
  delay(500);
  digitalWrite(led3, HIGH);
  delay(500);
  digitalWrite(led4, HIGH);
  delay(500);
  digitalWrite(led1, LOW);
  digitalWrite(led2, LOW);
  digitalWrite(led3, LOW);
  digitalWrite(led4, LOW);
  delay(500);
}
```

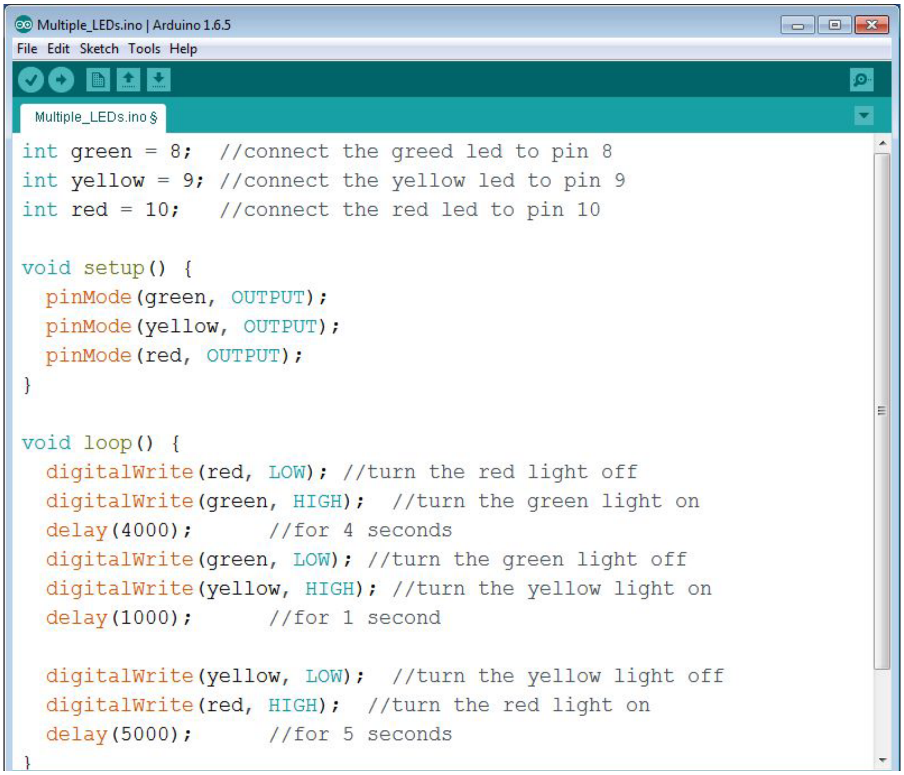
Challenge 3 Solution:

This code will make 4 LEDs blink to the tune of "Twinkle, Twinkle Little Star."

Challenge 4 Solution:


Multiple_LEDs.ino | Arduino 1.6.5
File Edit Sketch Tools Help
Multiple_LEDs.ino §

```
void loop() {
  digitalWrite(led1, HIGH);
  delay(500);
  digitalWrite(led1, LOW);
  delay(500);
  digitalWrite(led1, HIGH);
  delay(500);
  digitalWrite(led1, LOW);
  delay(500);
  digitalWrite(led2, HIGH);
  delay(500);
  digitalWrite(led2, LOW);
  delay(500);
  digitalWrite(led2, HIGH);
  delay(500);
  digitalWrite(led2, LOW);
  delay(500);
  digitalWrite(led3, HIGH);
  delay(500);
  digitalWrite(led3, LOW);
  delay(500);
  digitalWrite(led3, HIGH);
  delay(500);
  digitalWrite(led3, LOW);
  delay(500);
  digitalWrite(led4, HIGH);
  delay(1000);
  digitalWrite(led4, LOW);
  delay(500);
}
```


Multiple_LEDs.ino | Arduino 1.6.5
File Edit Sketch Tools Help
Multiple_LEDs.ino §

```
int green = 8;  //connect the greed led to pin 8
int yellow = 9; //connect the yellow led to pin 9
int red = 10;   //connect the red led to pin 10

void setup() {
  pinMode(green, OUTPUT);
  pinMode(yellow, OUTPUT);
  pinMode(red, OUTPUT);
}

void loop() {
  digitalWrite(red, LOW); //turn the red light off
  digitalWrite(green, HIGH);  //turn the green light on
  delay(4000);      //for 4 seconds
  digitalWrite(green, LOW); //turn the green light off
  digitalWrite(yellow, HIGH); //turn the yellow light on
  delay(1000);      //for 1 second

  digitalWrite(yellow, LOW);  //turn the yellow light off
  digitalWrite(red, HIGH);  //turn the red light on
  delay(5000);      //for 5 seconds
}
```