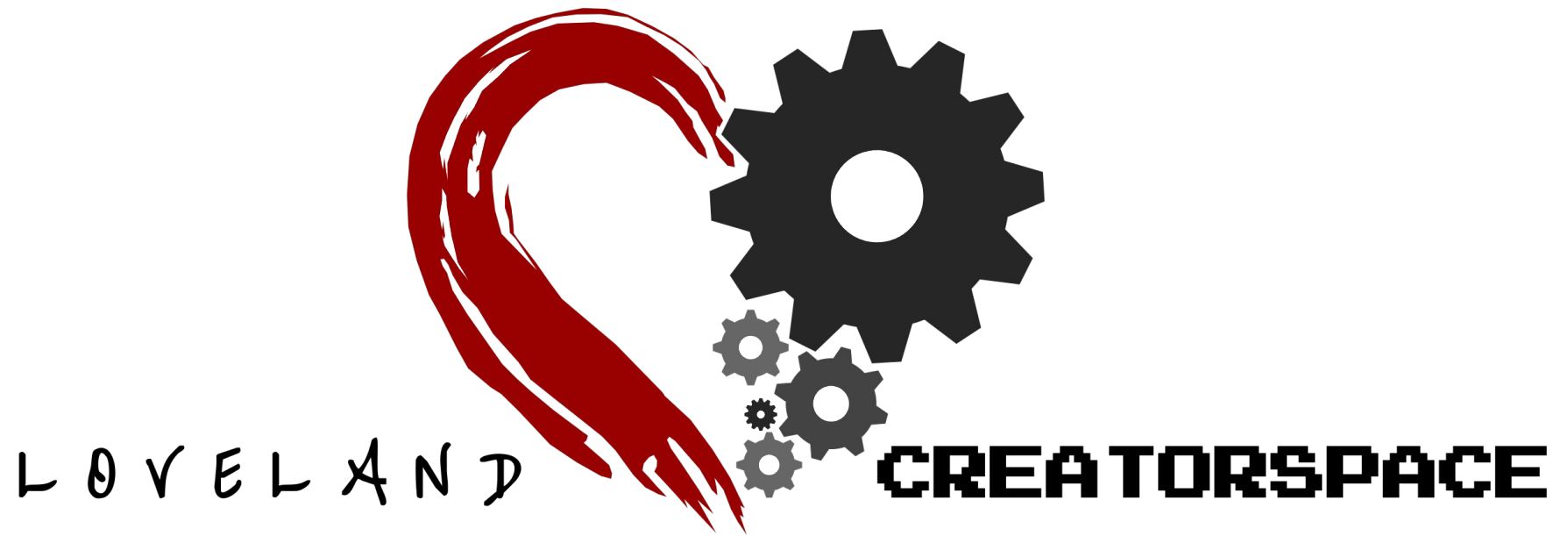


# Arduino 101

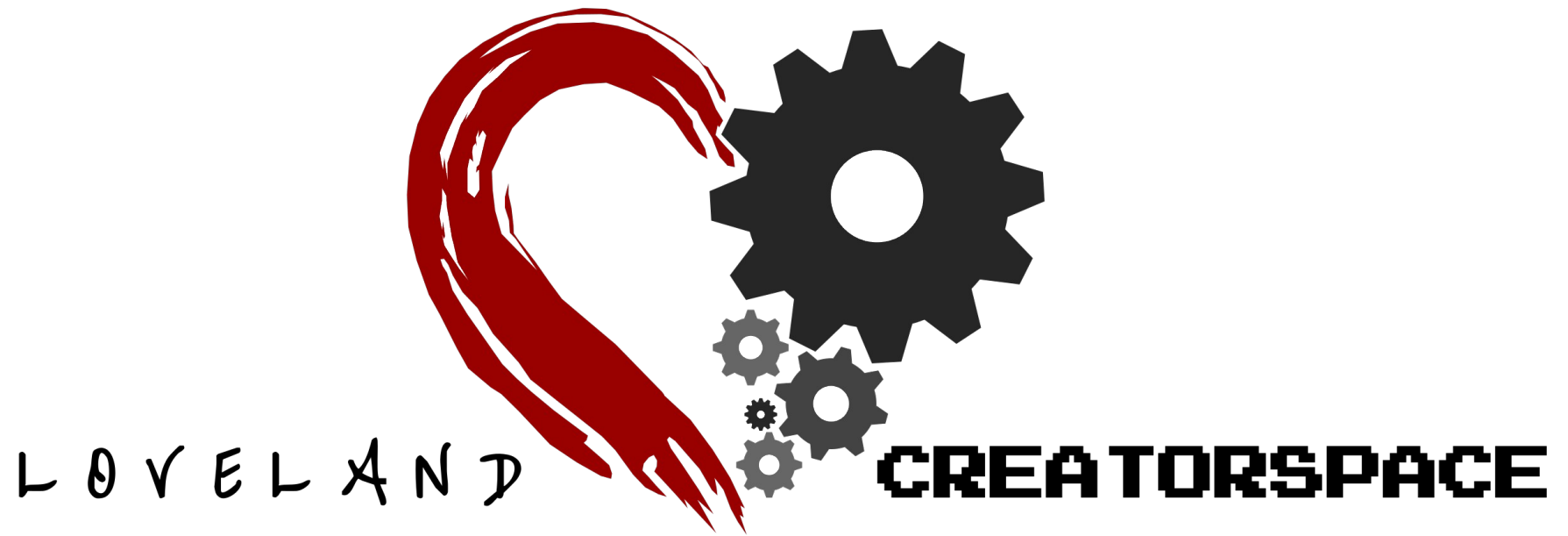
Creative Commons



*Presented by Daniel Packard - 2013/12/14*  
*With Maurice Woods III, Casey Kuhns, and Stephen Warren*



*A Maker Space for Loveland.*



*A Maker Space for Loveland.*

Special thanks to SparkFun, Ferguson Highschool, and E3 Learning



**Arduino**

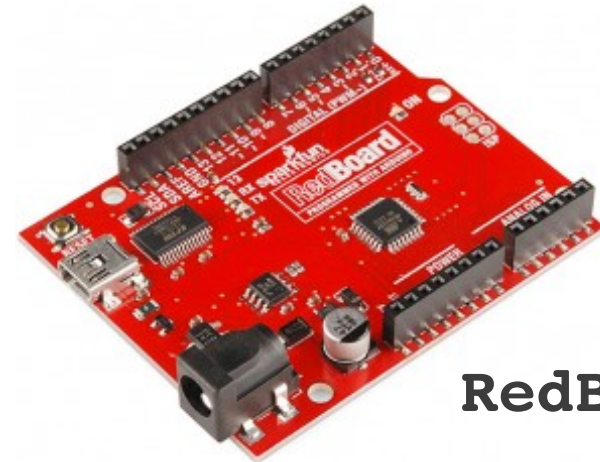
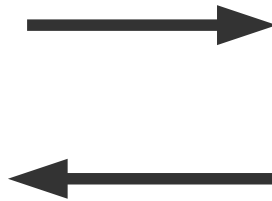


Arduino

*You've been **lied** to!*



**Arduino**



**RedBoard**

*You've been **lied** to!*

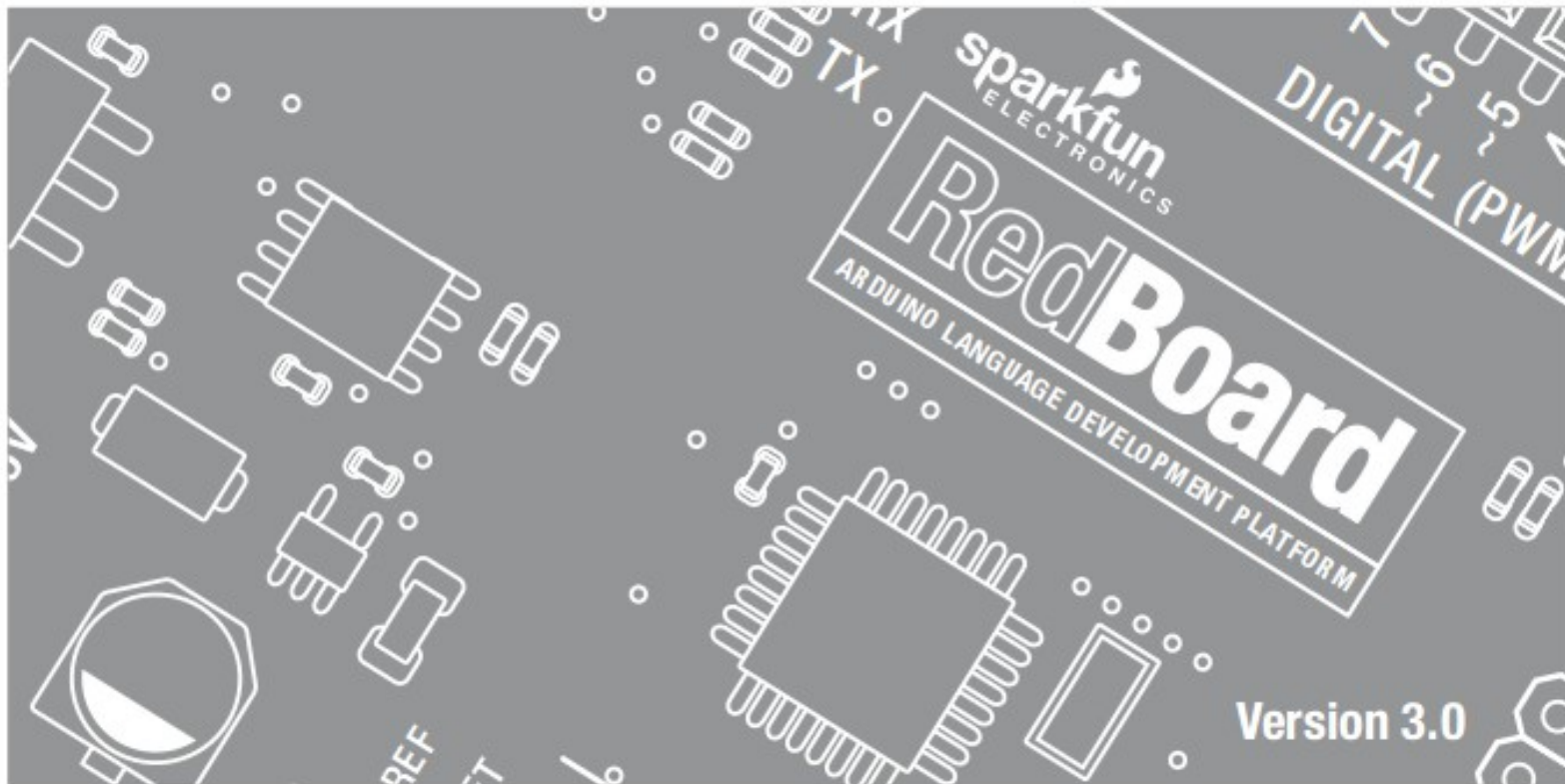
This class will use the **RedBoard** platform from



# SIK GUIDE

---

Your guide to the SparkFun Inventor's Kit for the SparkFun RedBoard



<http://dlnmh9ip6v2uc.cloudfront.net/datasheets/Kits/SFE-SIK-RedBoard-Guide-Version3.0-Online.pdf>

**Arduino (Redboard) is...**





# Arduino (Redboard) is...



A programmable micro-controller.

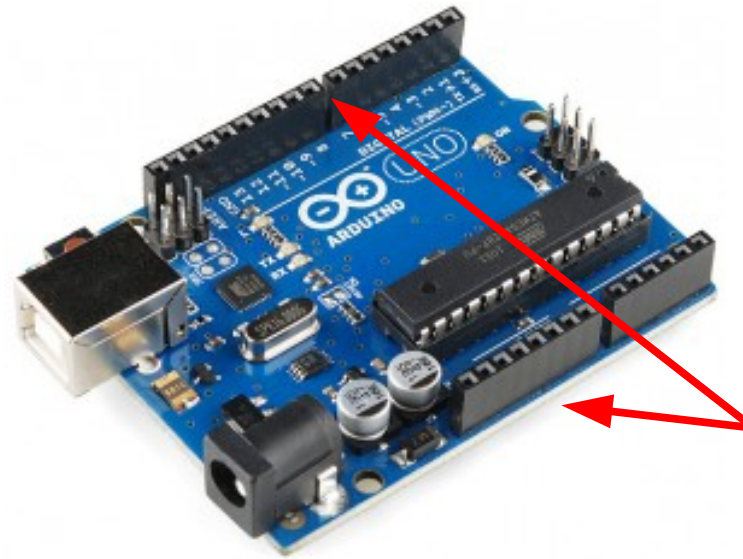
# Arduino (Redboard) is...



Atmega Processor

A programmable micro-controller.

# Arduino (Redboard) is...

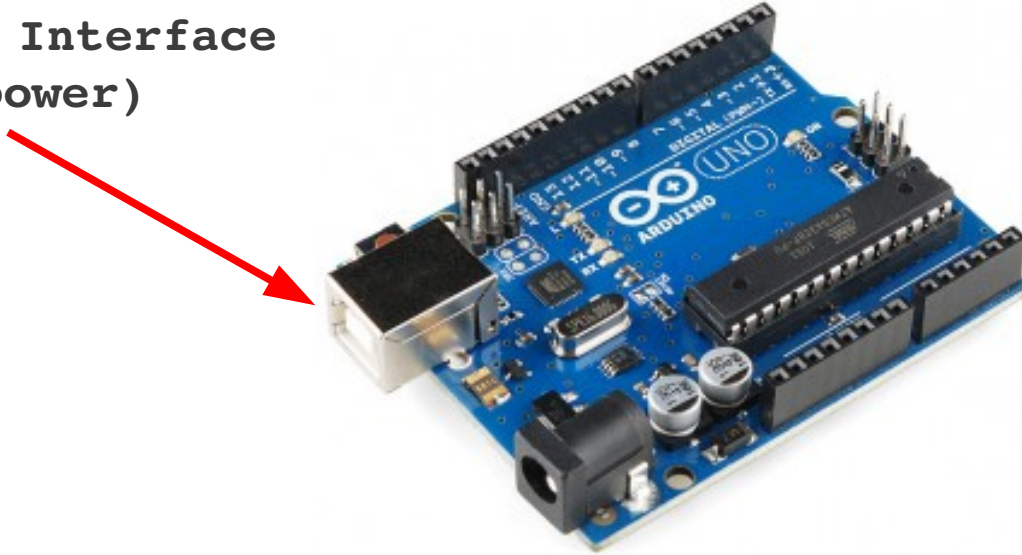


Input/Output pins  
(power)

A programmable micro-controller.

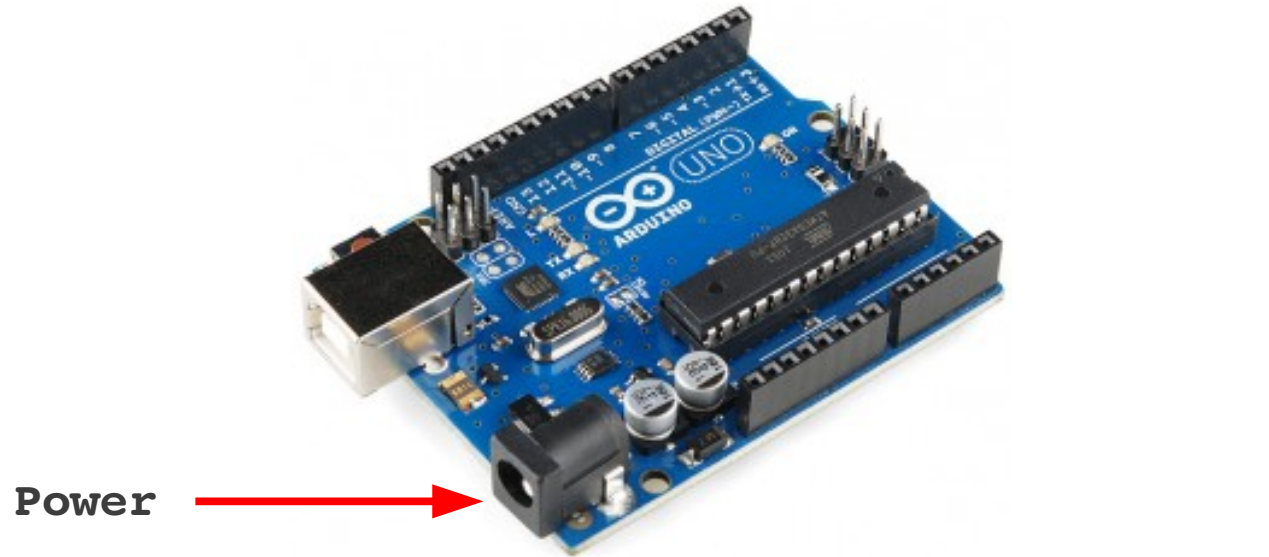
# Arduino (Redboard) is...

USB Serial Interface  
(and power)



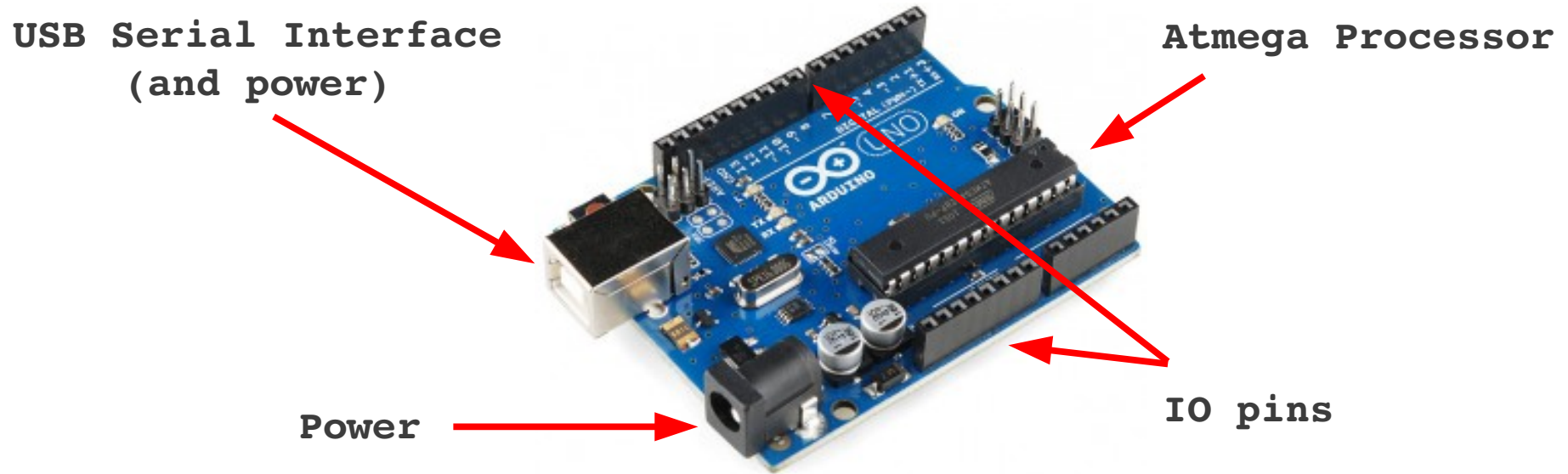
A programmable micro-controller.

# Arduino (Redboard) is...



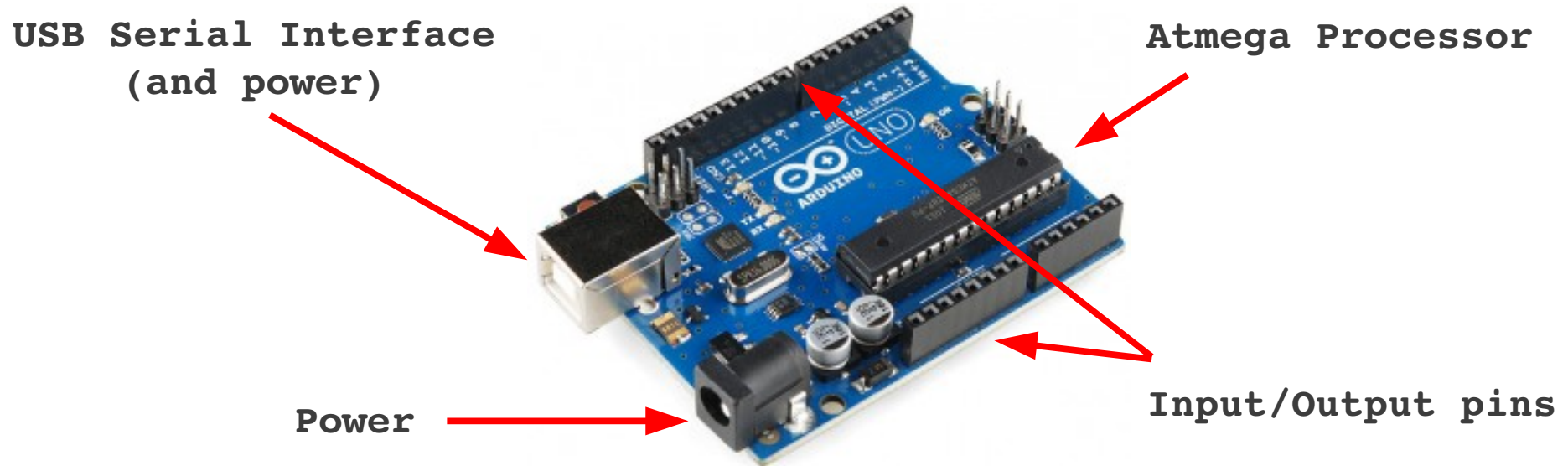
A programmable micro-controller.

# Arduino (Redboard) is...



A programmable micro-controller.

# Arduino (Redboard) is...



A programmable micro-controller.

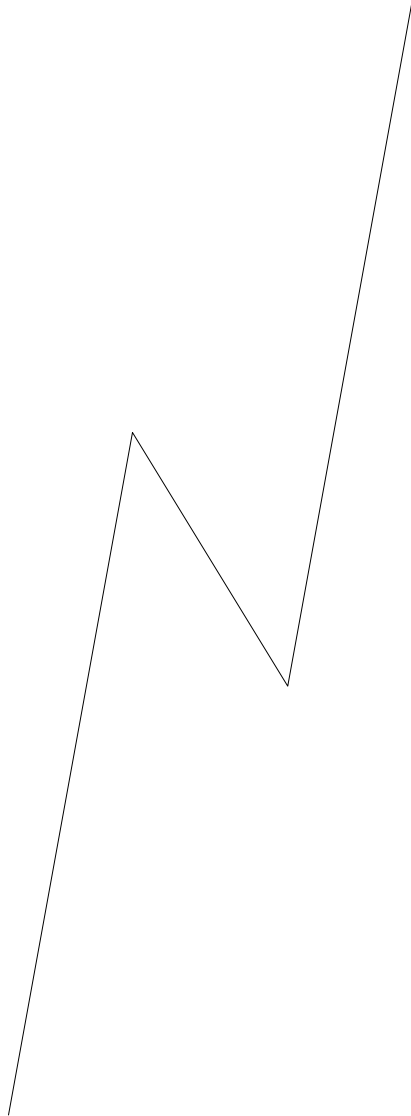
But the best way to explain the Arduino is with some examples:

<https://www.youtube.com/watch?v=6mXM-oGggrM>

<http://www.youtube.com/watch?v=yuKcLG1tqks>

<http://www.instructables.com/id/20-Unbelievable-Arduino-Projects/>

# Programmable Circuits





# Programmable Circuits

Build a simple circuit  
(electronics)

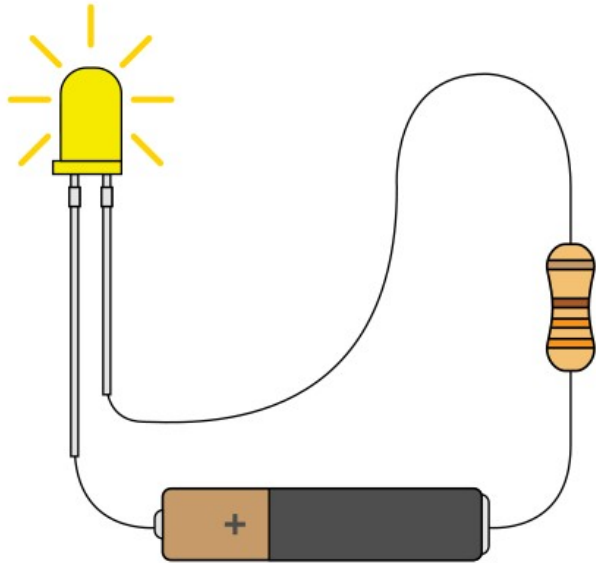


Image attributed to SparkFun Electronics

# Programmable Circuits

Build a simple circuit  
(electronics)

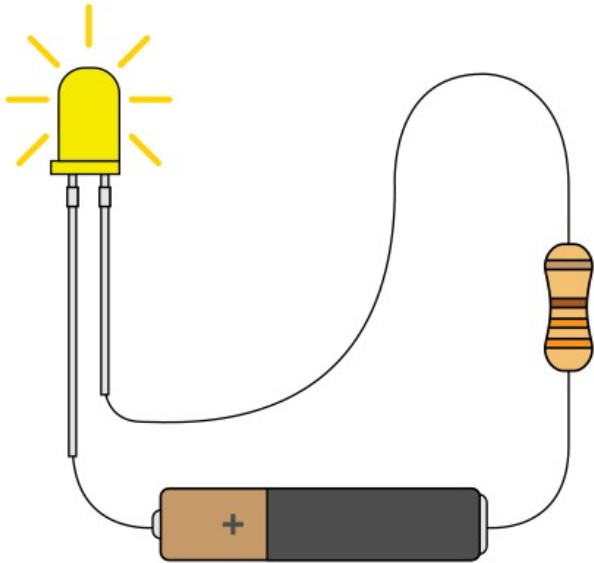


Image attributed to SparkFun Electronics

A screenshot of the Arduino IDE interface. The title bar reads "Blink | Arduino 1.0". The menu bar includes "File", "Edit", "Sketch", "Tools", and "Help". Below the menu bar is a toolbar with icons for opening, saving, and running. The main text area shows the "Blink" sketch code. The code includes a comment block describing the sketch, a setup function to initialize pin 13 as an output, and a loop function that writes HIGH to the pin, delays for 1000ms, writes LOW, and delays for 1000ms.

```
Blink | Arduino 1.0
File Edit Sketch Tools Help

Blink
/*
 * Blink
 * Turns on an LED on for one second, then off for one second, repeats.
 *
 * This example code is in the public domain.
 */

void setup() {
  // initialize the digital pin as an output.
  // Pin 13 has an LED connected on most Arduino boards
  pinMode(13, OUTPUT);
}

void loop() {
  digitalWrite(13, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);             // wait for a second
  digitalWrite(13, LOW);  // turn the LED off by making the pin LOW
  delay(1000);             // wait for a second
}
```

Write a control program  
(software)

# Arduino 101

## Installation:

### **Windows and Mac users:**

*Download the installer from:*

- <http://arduino.cc>

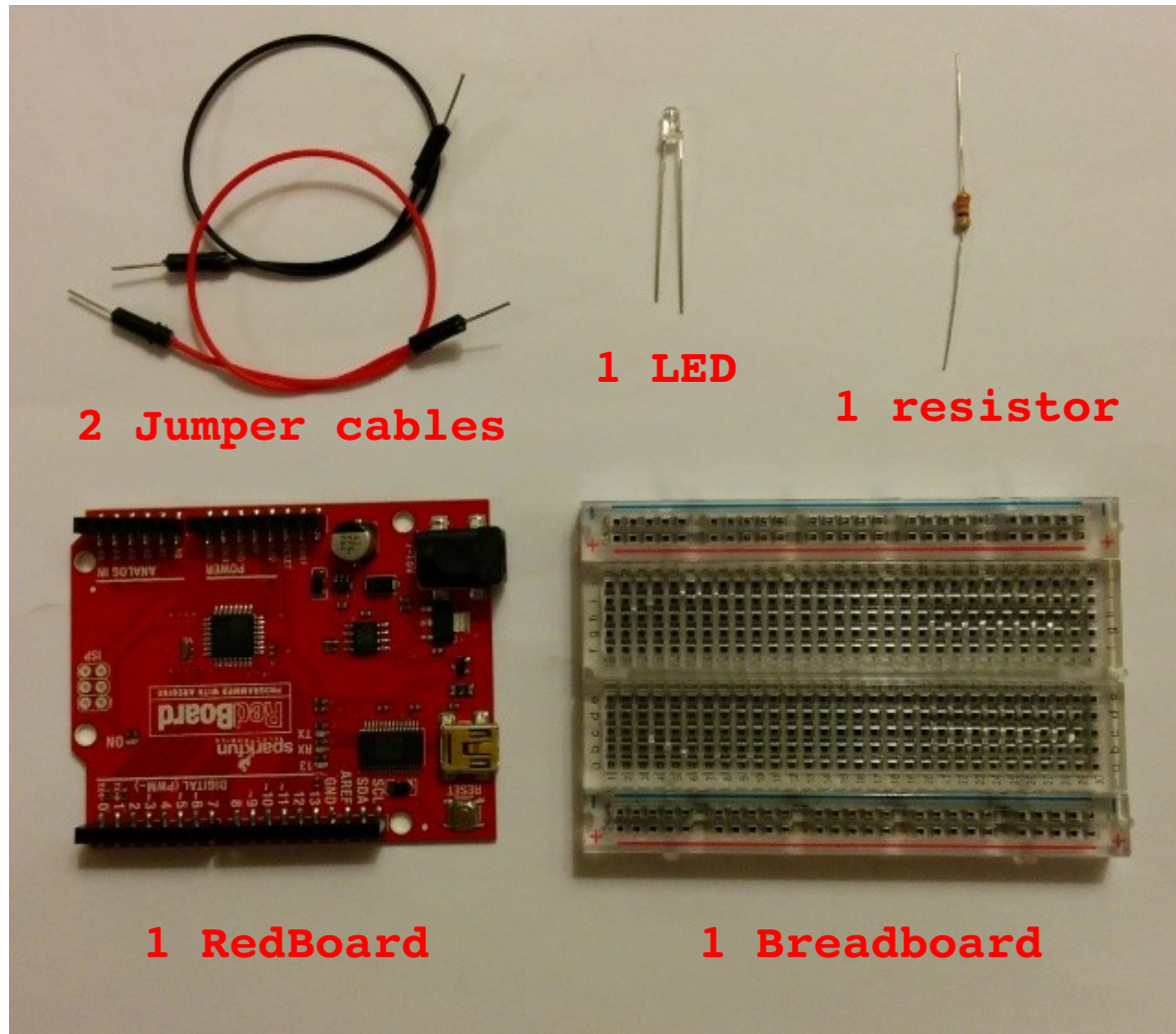
### **Linux users (specifically Debian) :**

*In a terminal:*

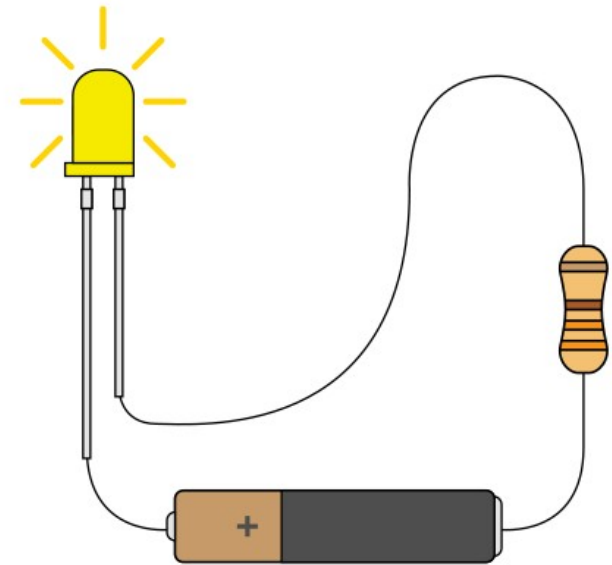
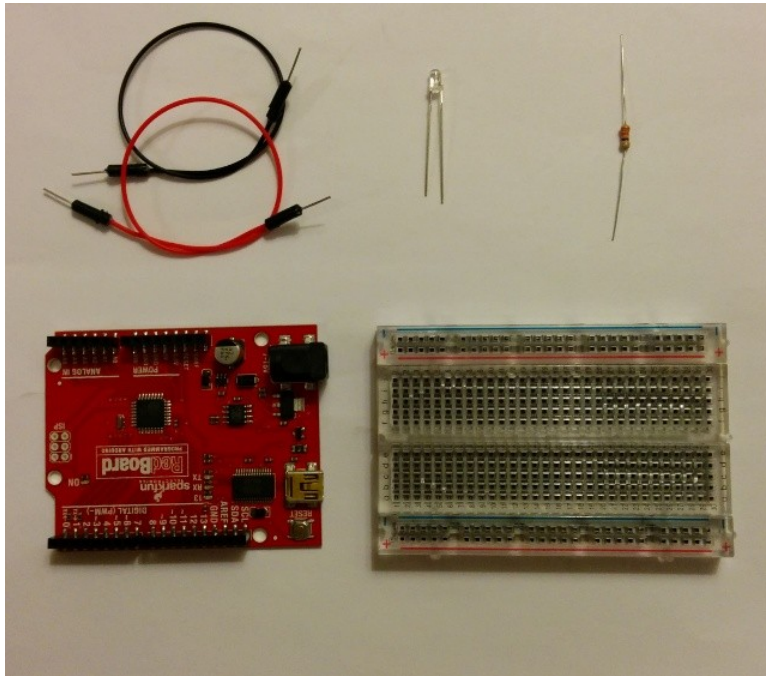
- `sudo apt-get install arduino`
- `sudo usermod -aG dialout <user>`
- Logout, and log back in for changes to take effect.

# Building the Circuit

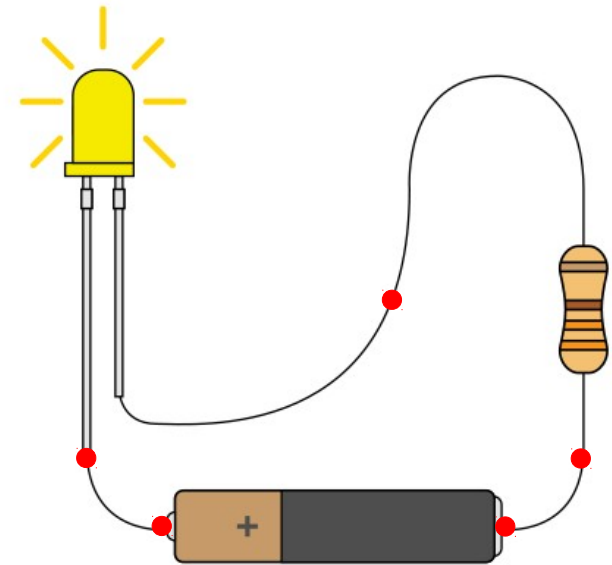
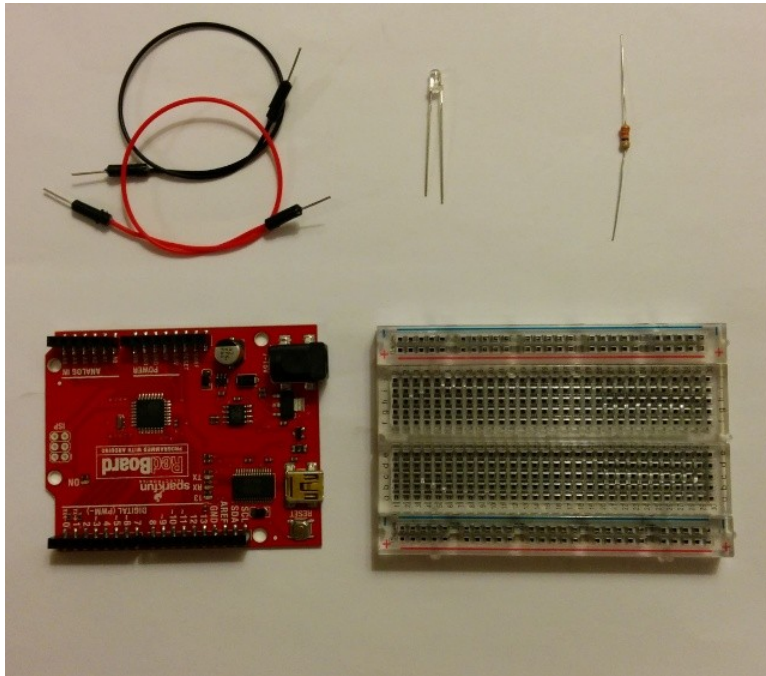
Inside your kit, you'll find:



# Building the Circuit

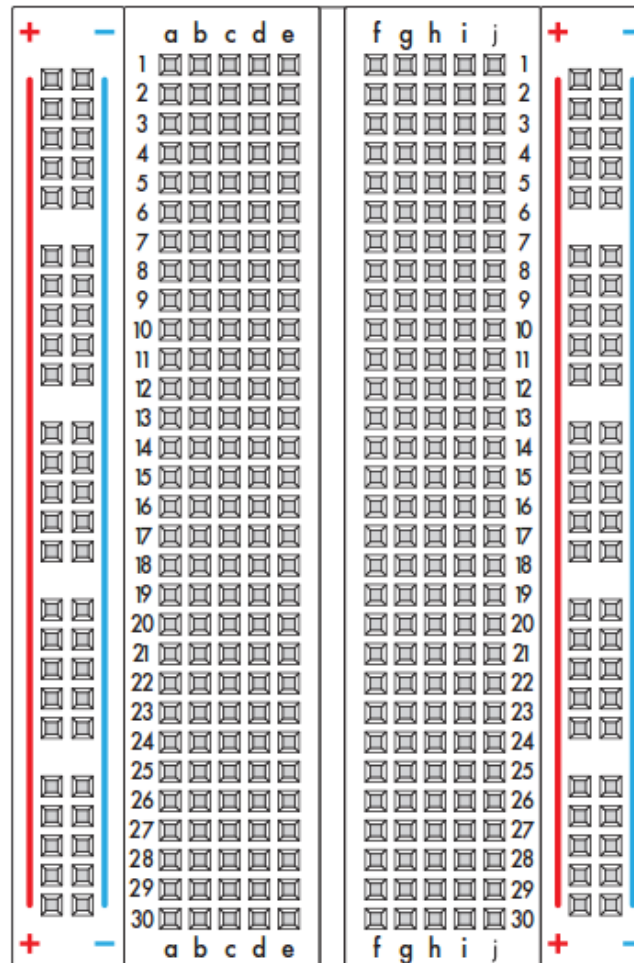


# Building the Circuit



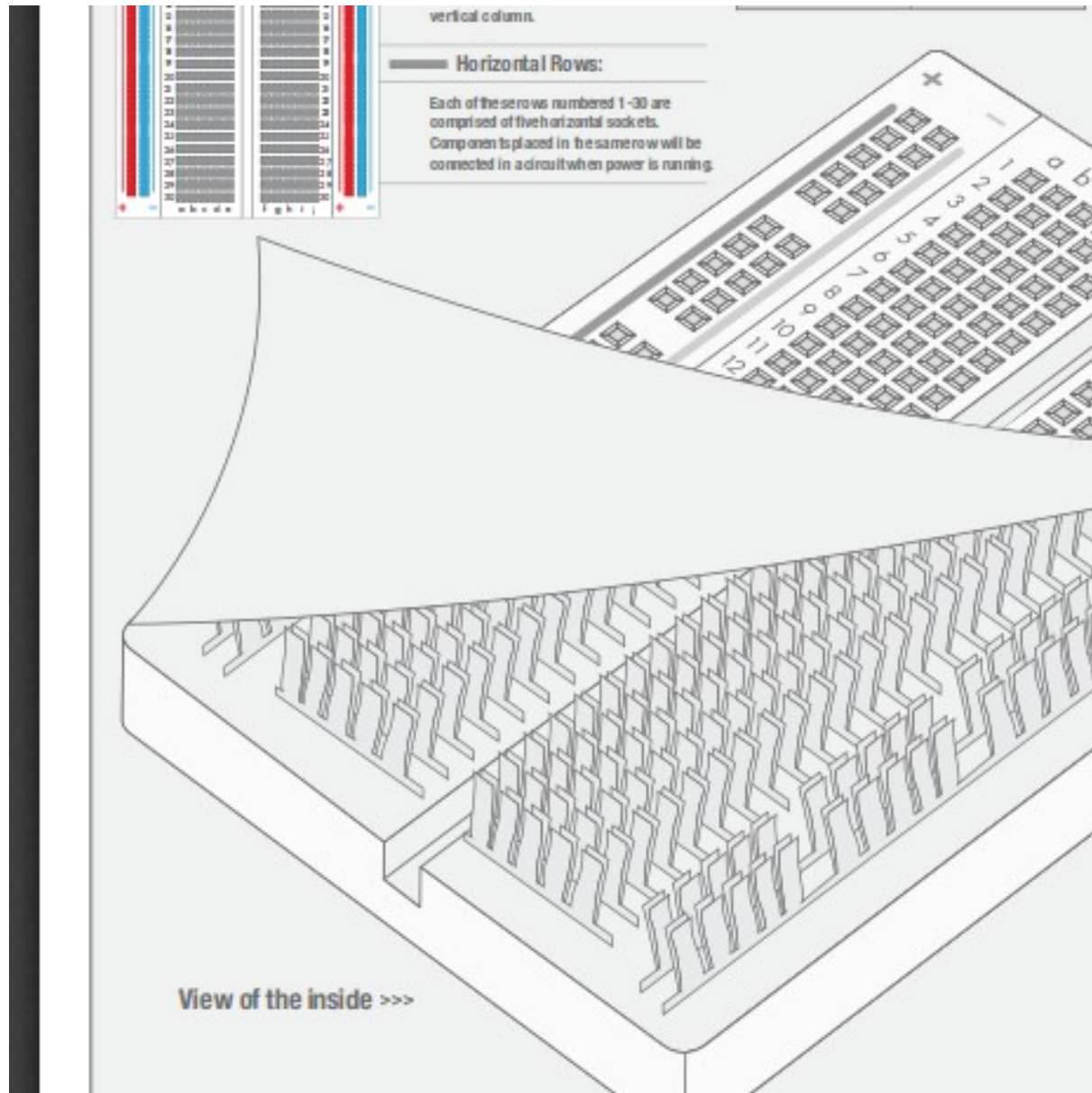
# Building the Circuit

Bread board to the rescue!





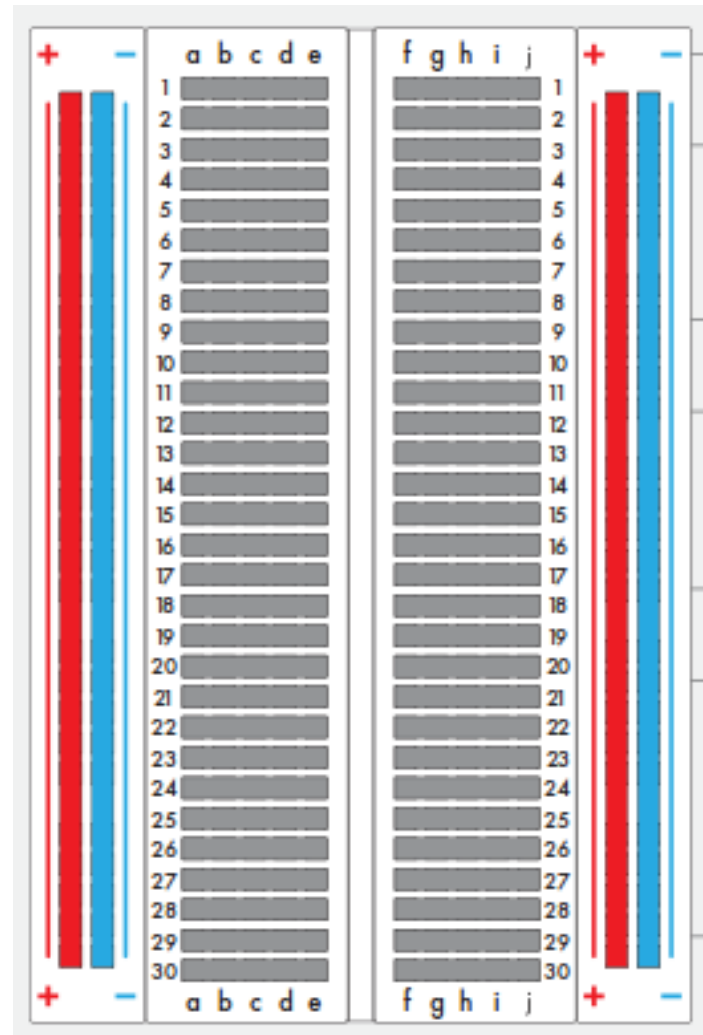
## Building the Circuit





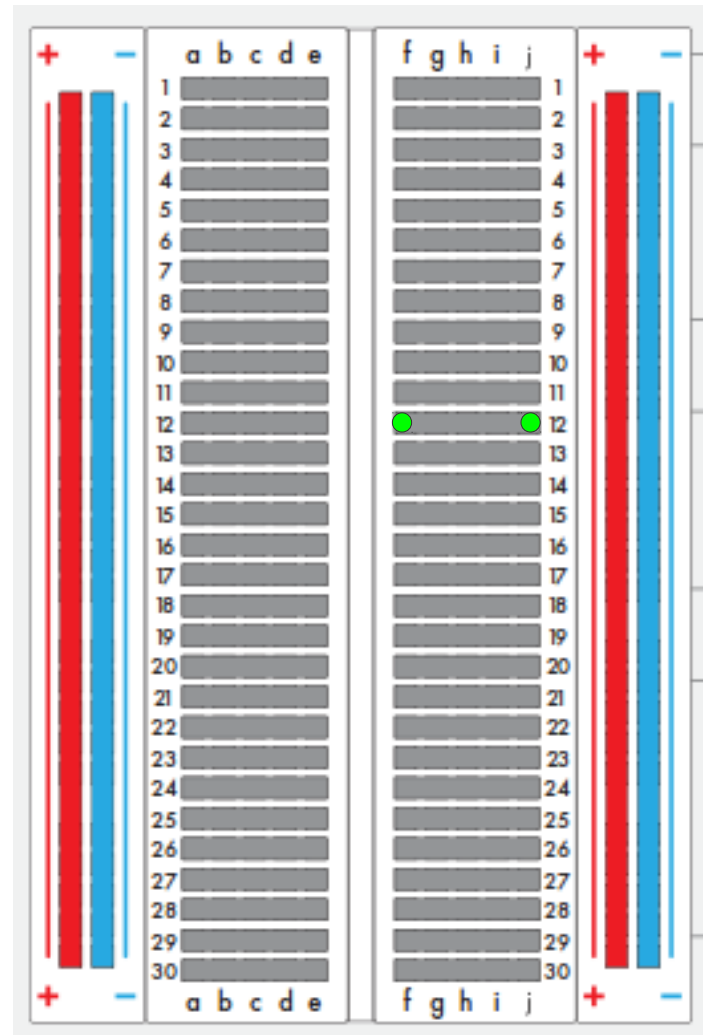
# Building the Circuit

Bread board to the rescue!



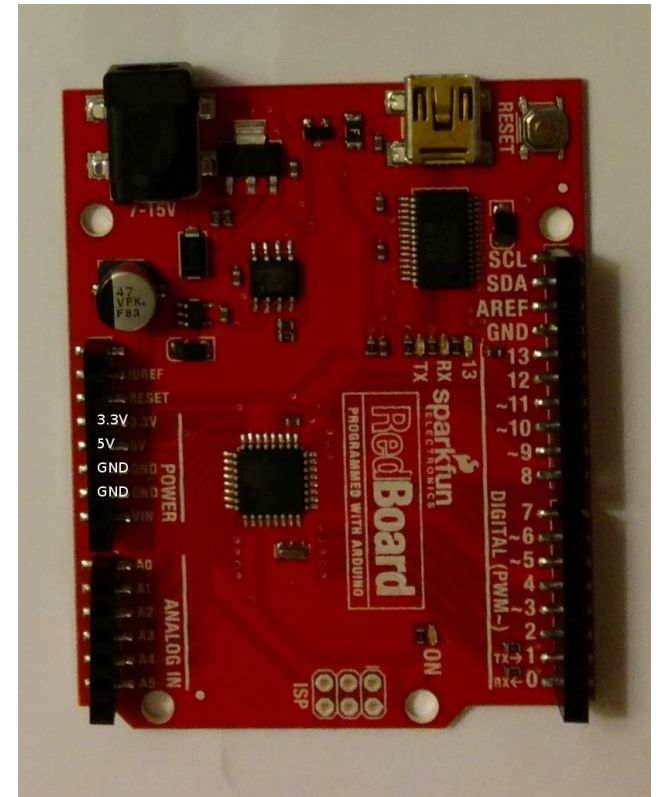
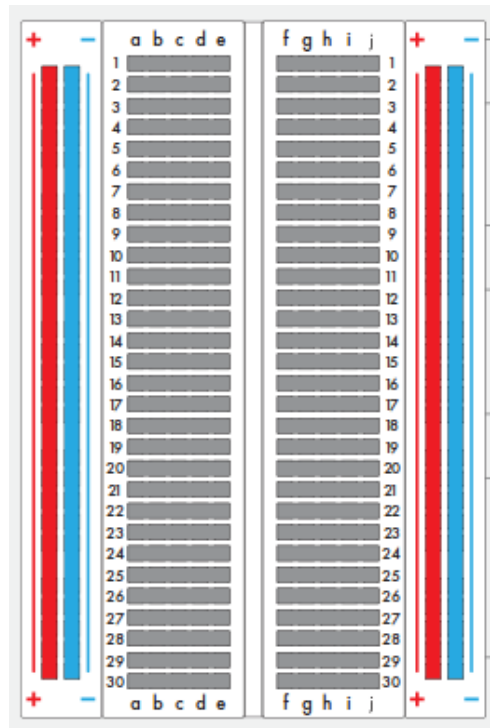
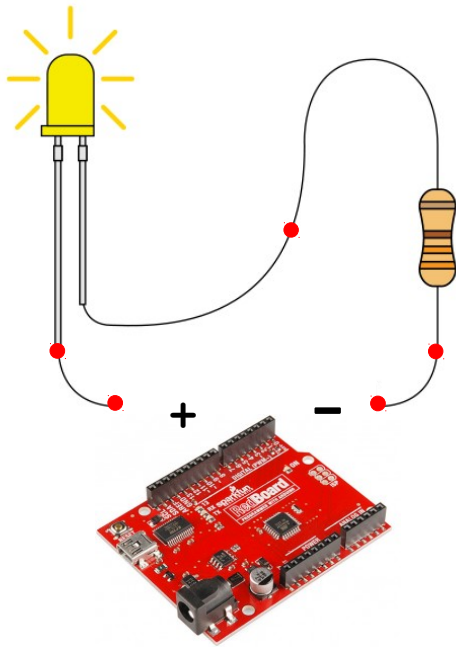
# Building the Circuit

Bread board to the rescue!



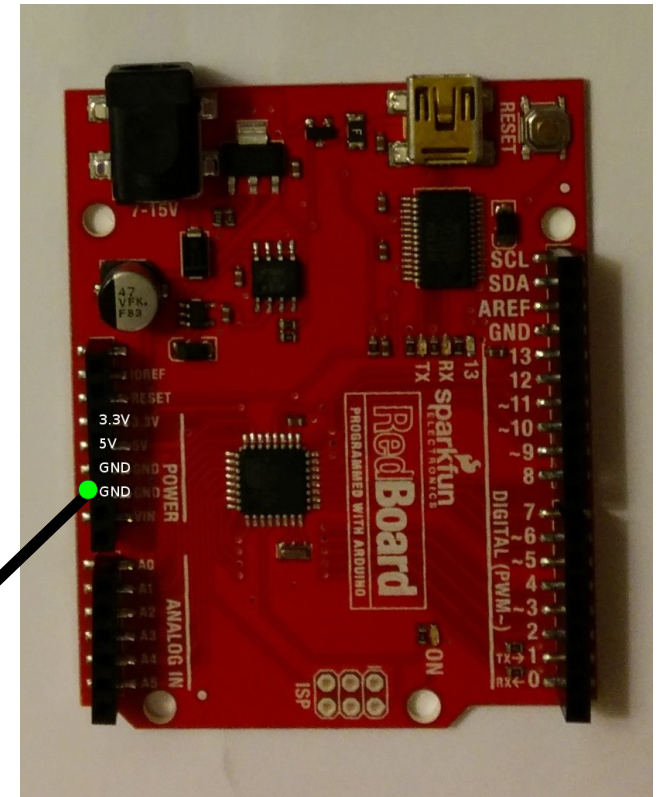
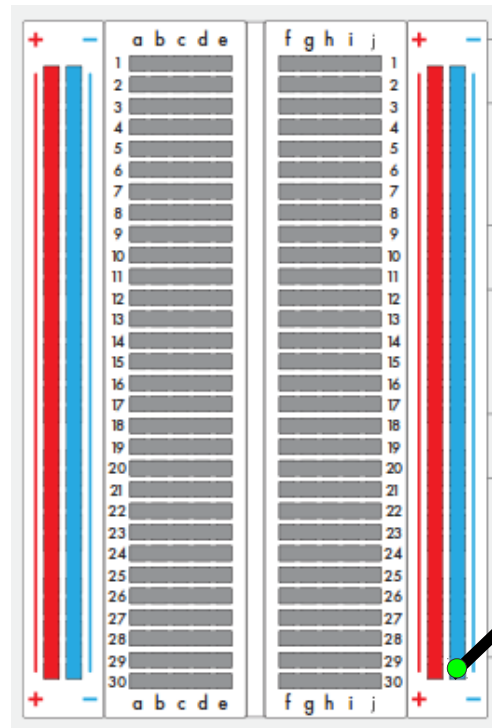
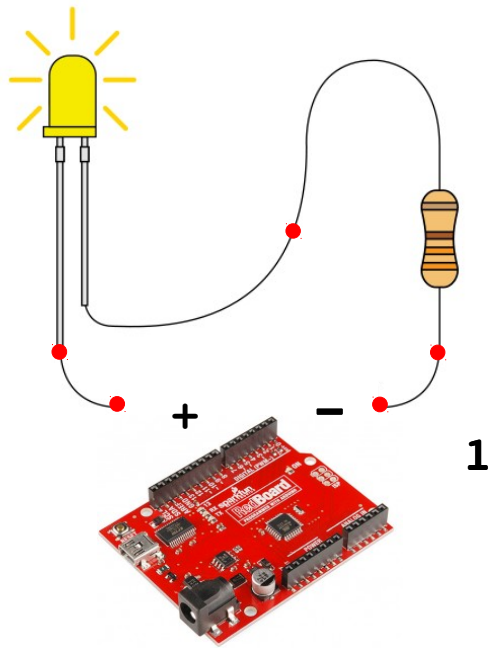
# Building the Circuit

Bread board to the rescue!



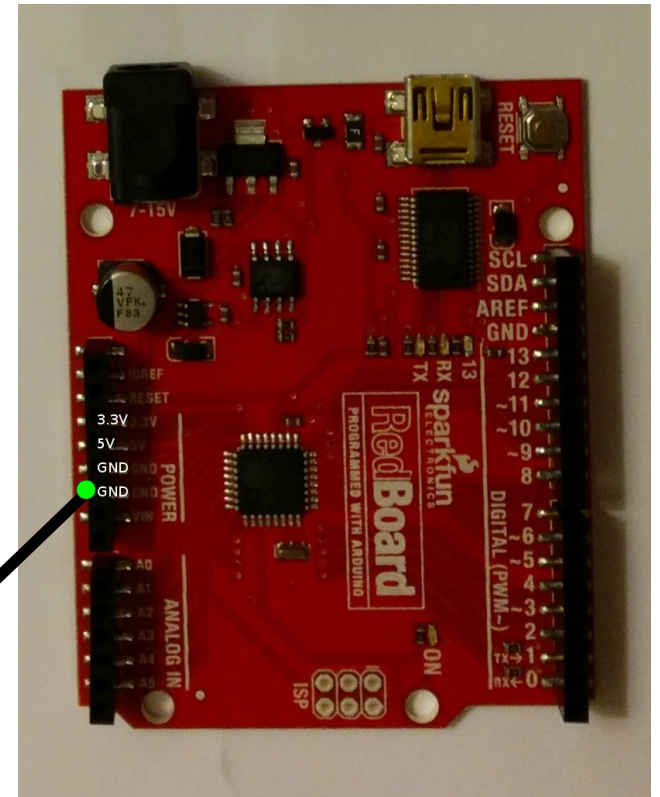
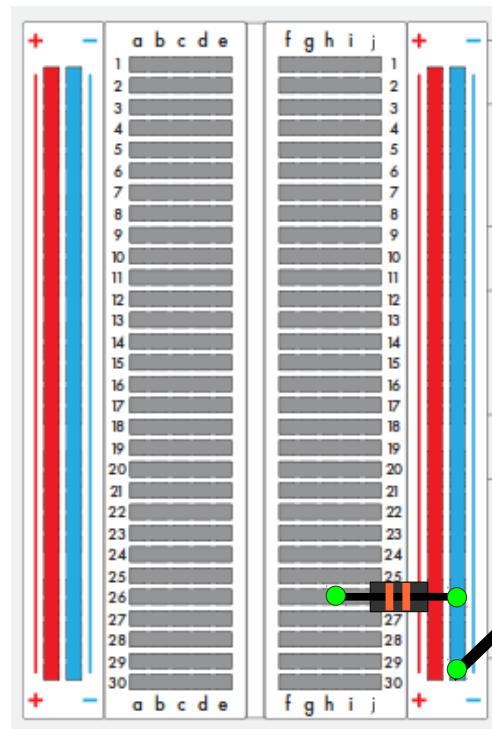
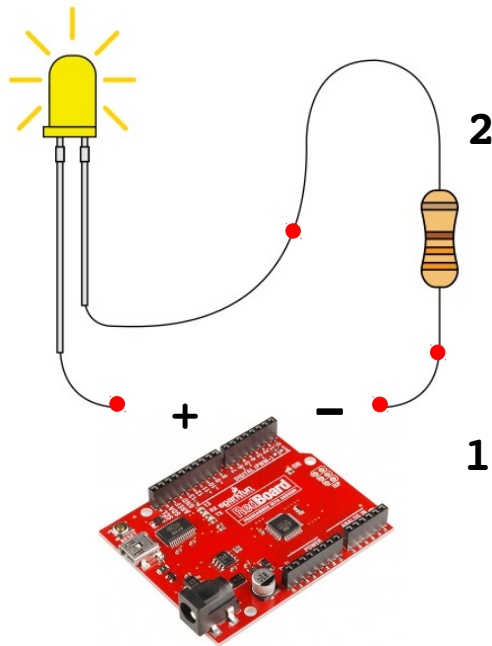
# Building the Circuit

Bread board to the rescue!



# Building the Circuit

Bread board to the rescue!

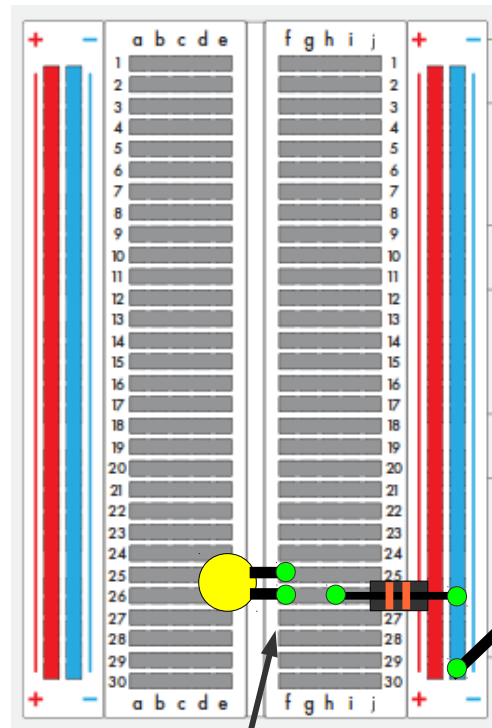
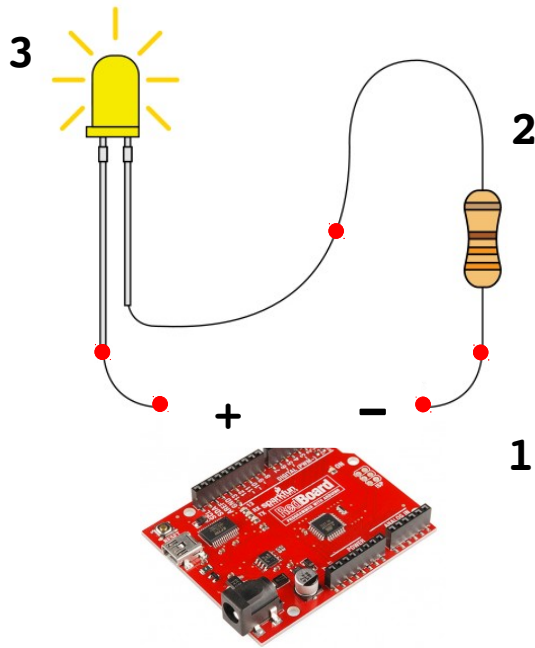


Short Leg

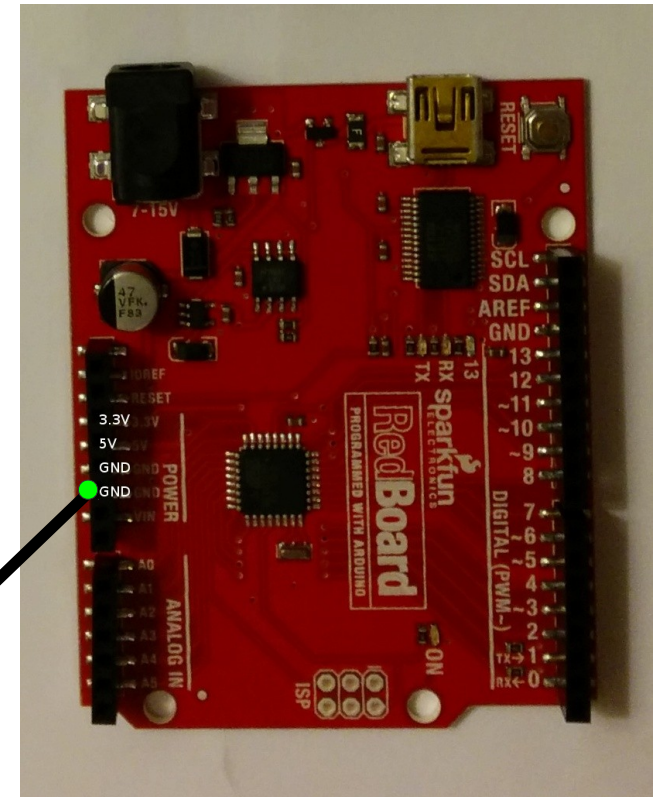


# Building the Circuit

Bread board to the rescue!

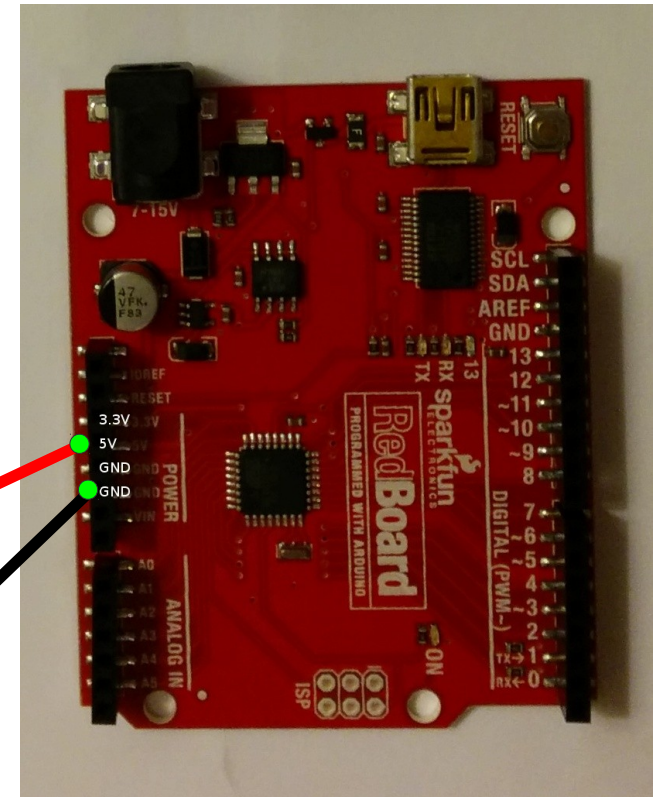
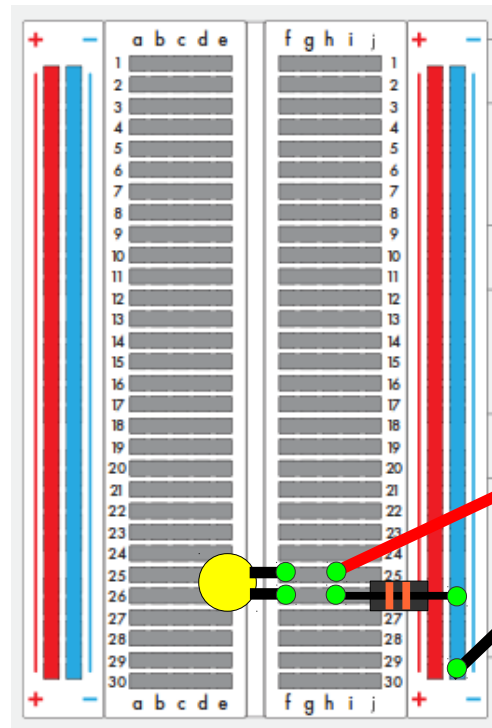
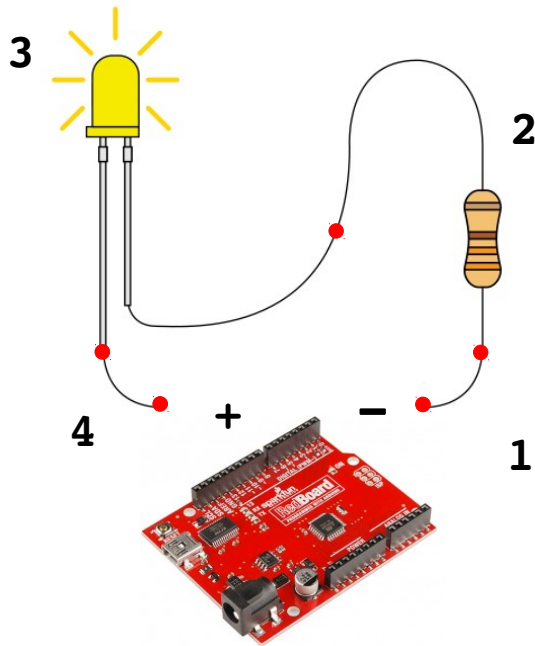


Short Leg

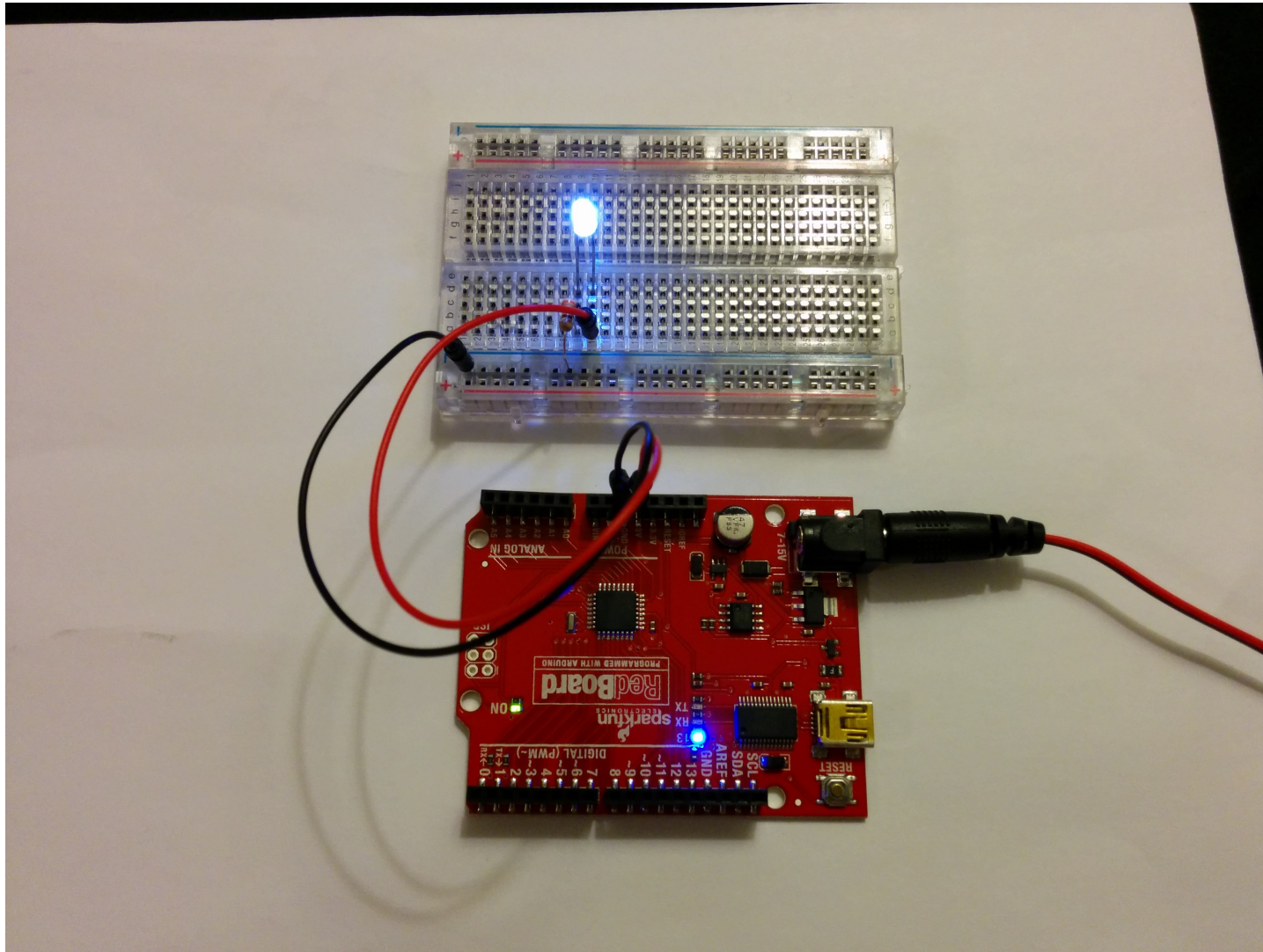


# Building the Circuit

Bread board to the rescue!



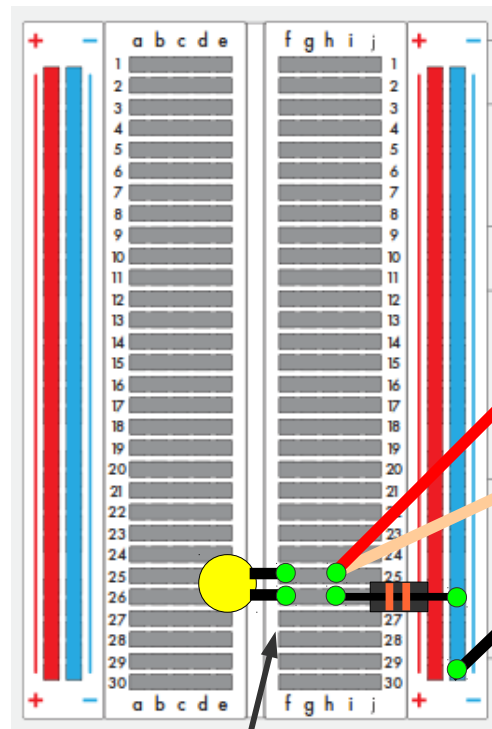
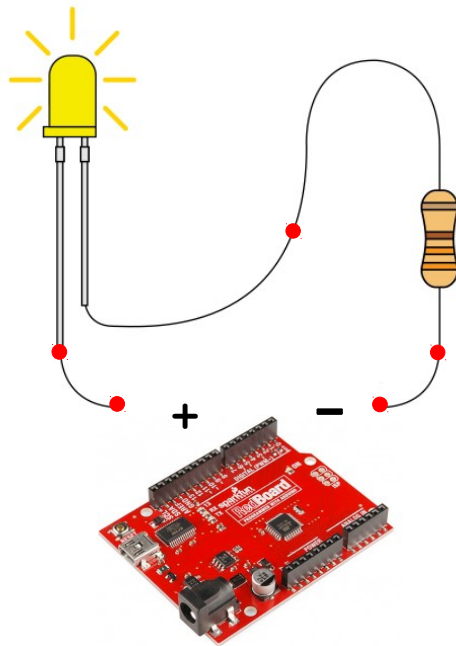
# Testing the Circuit



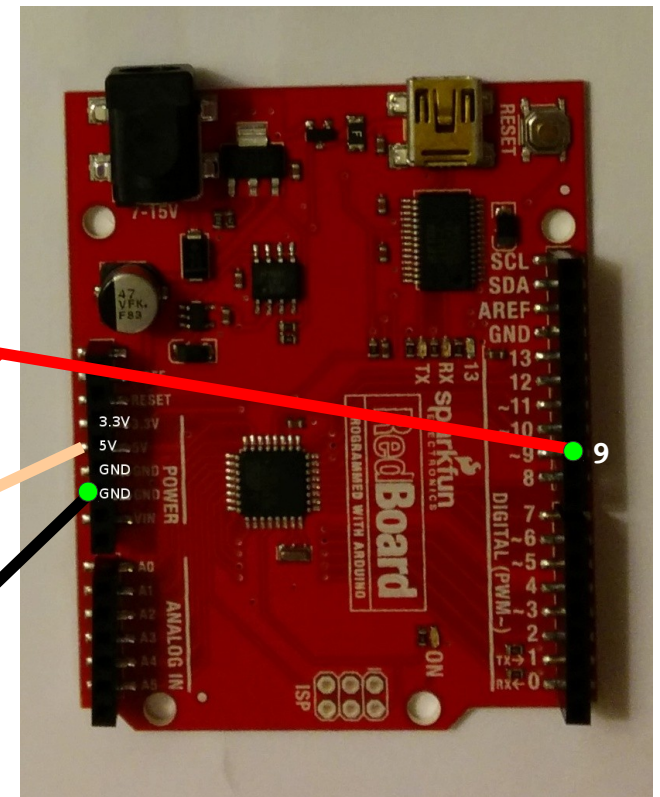


# Getting ready to program the Circuit

First, we need to connect to a pin we can control  
(instead of the 5V power pin)



Short Leg



# Programming Primer

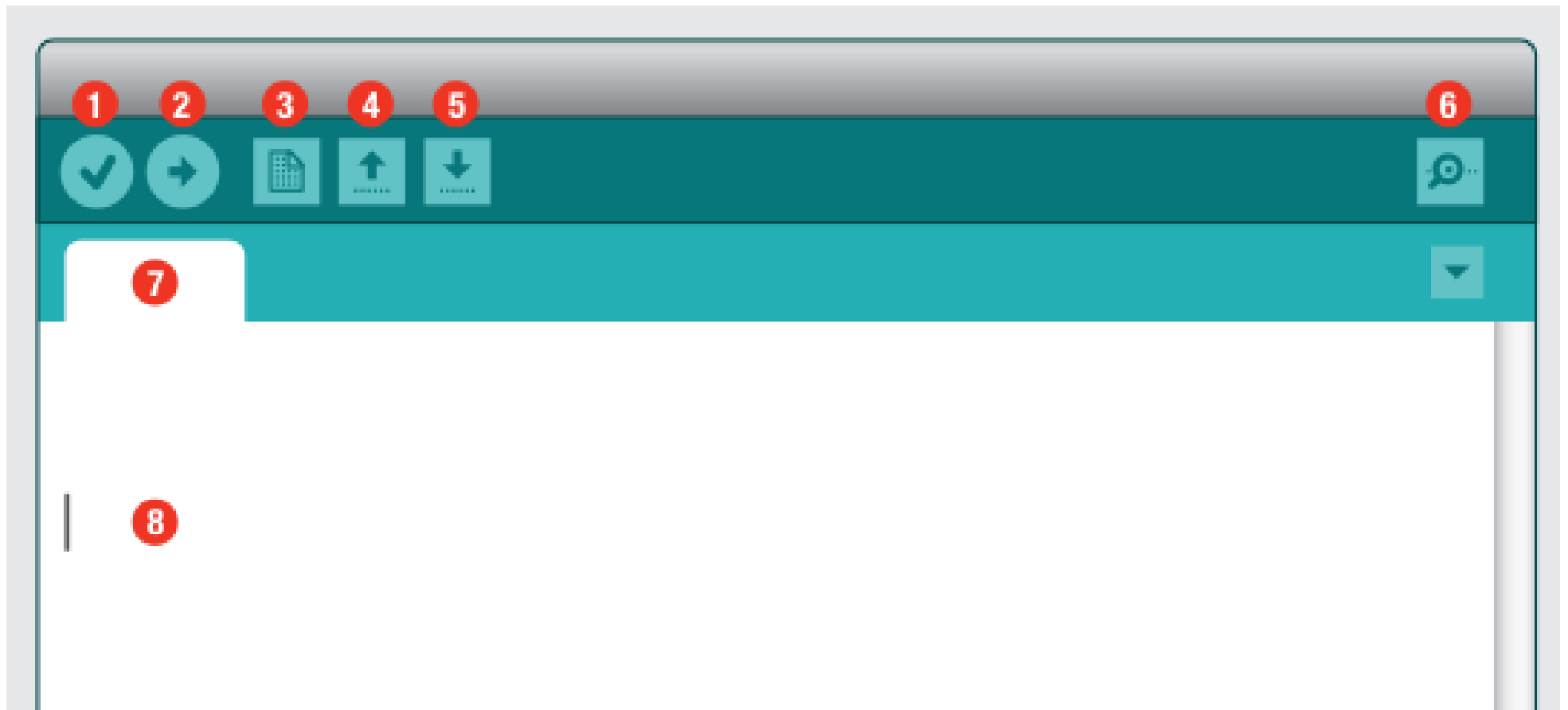
```
// single-line comments look like this.
```

```
/* Multi-line comments  
   Look like this */
```

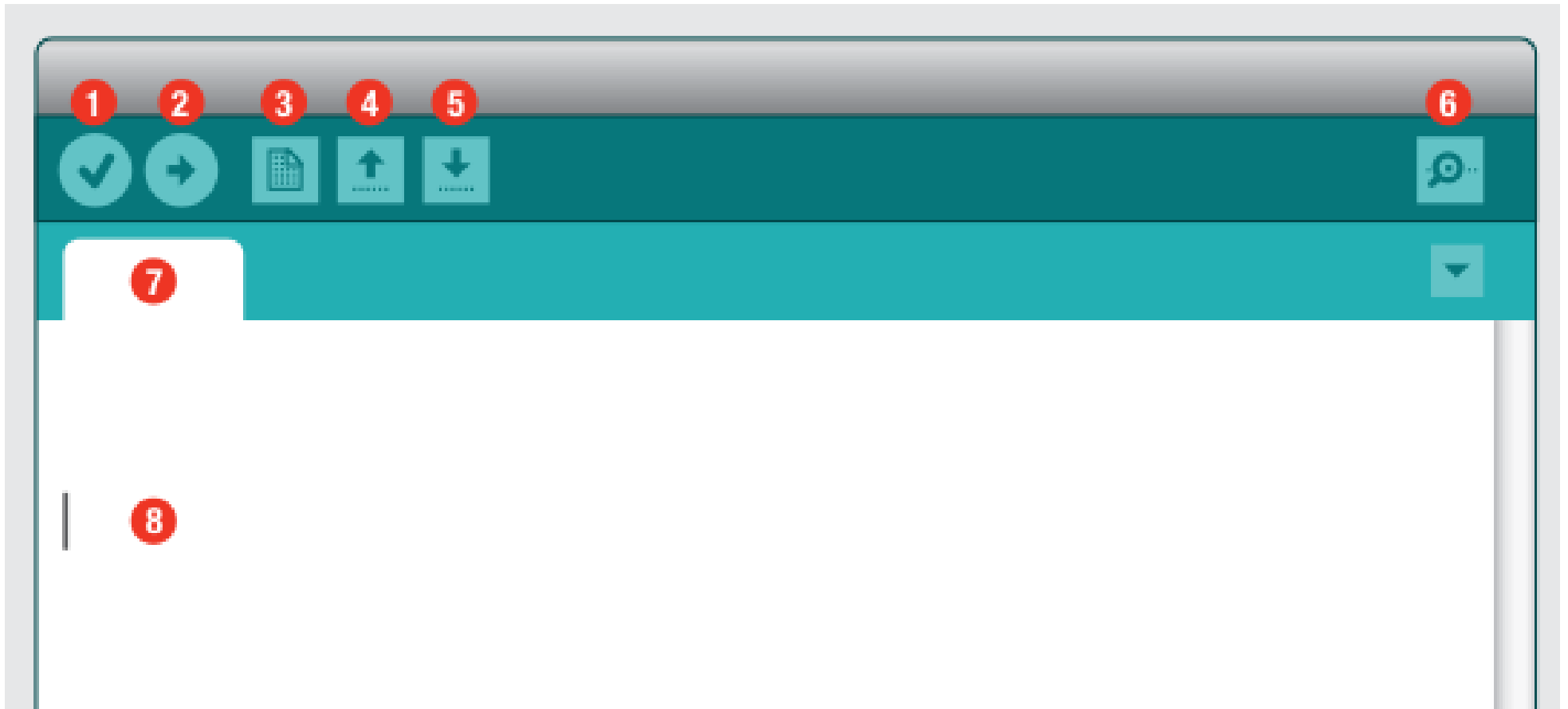
```
// variable declarations look like this  
int variable_name = 42;
```

```
// function declarations look like this  
void func_name(args...)  
{  
    // function body  
}
```

# The Arduino IDE (Integrated Development Environment)

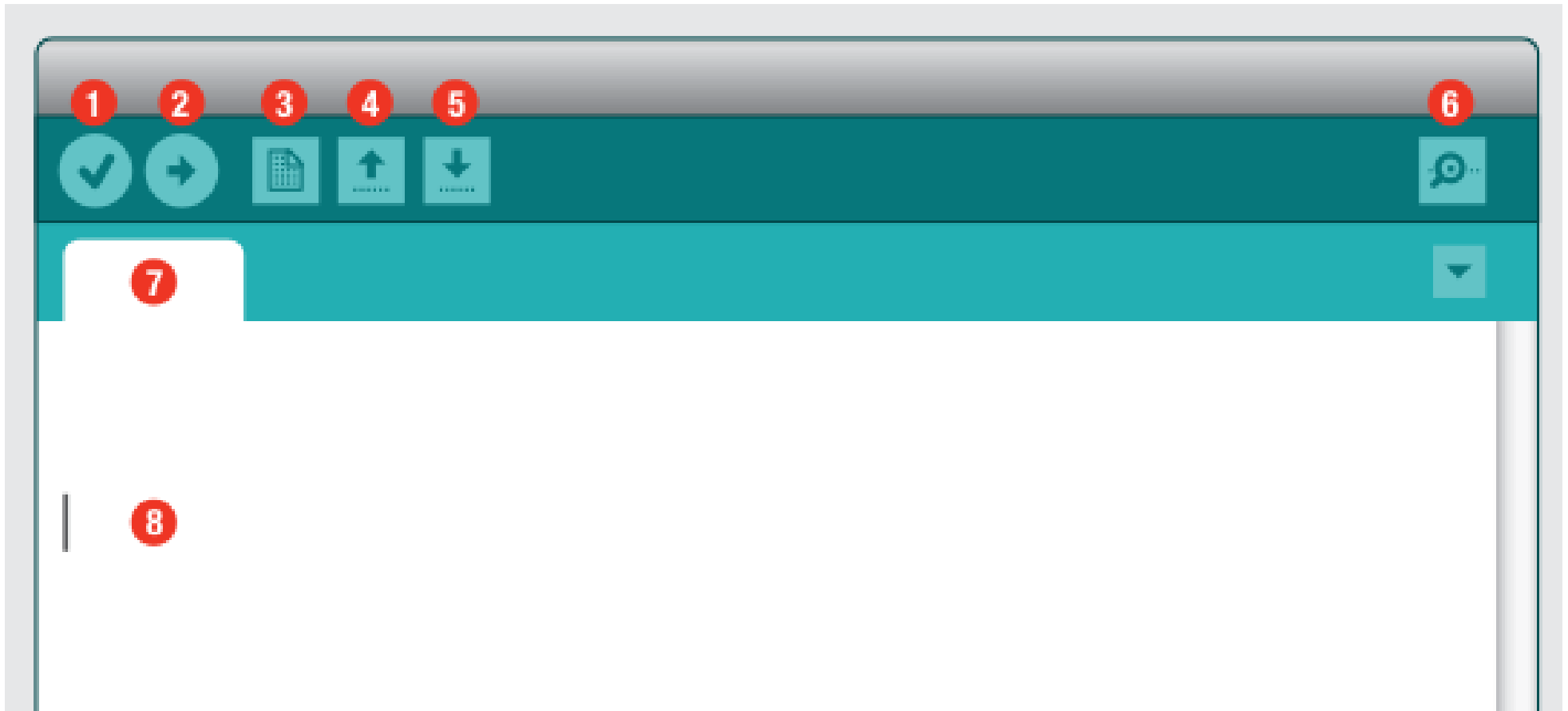


# The Arduino IDE (Integrated Development Environment)



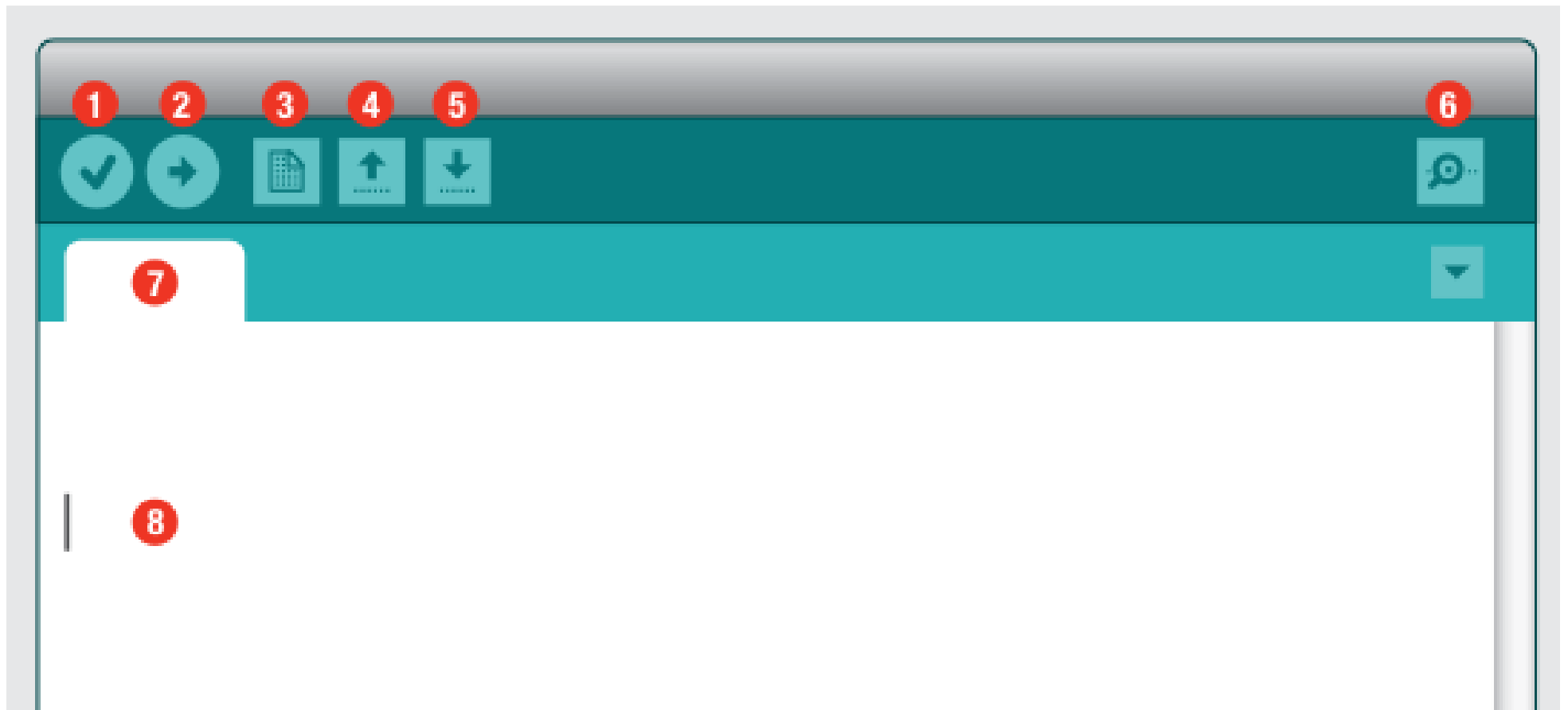
1. "verify" - check that your code is valid.

# The Arduino IDE (Integrated Development Environment)



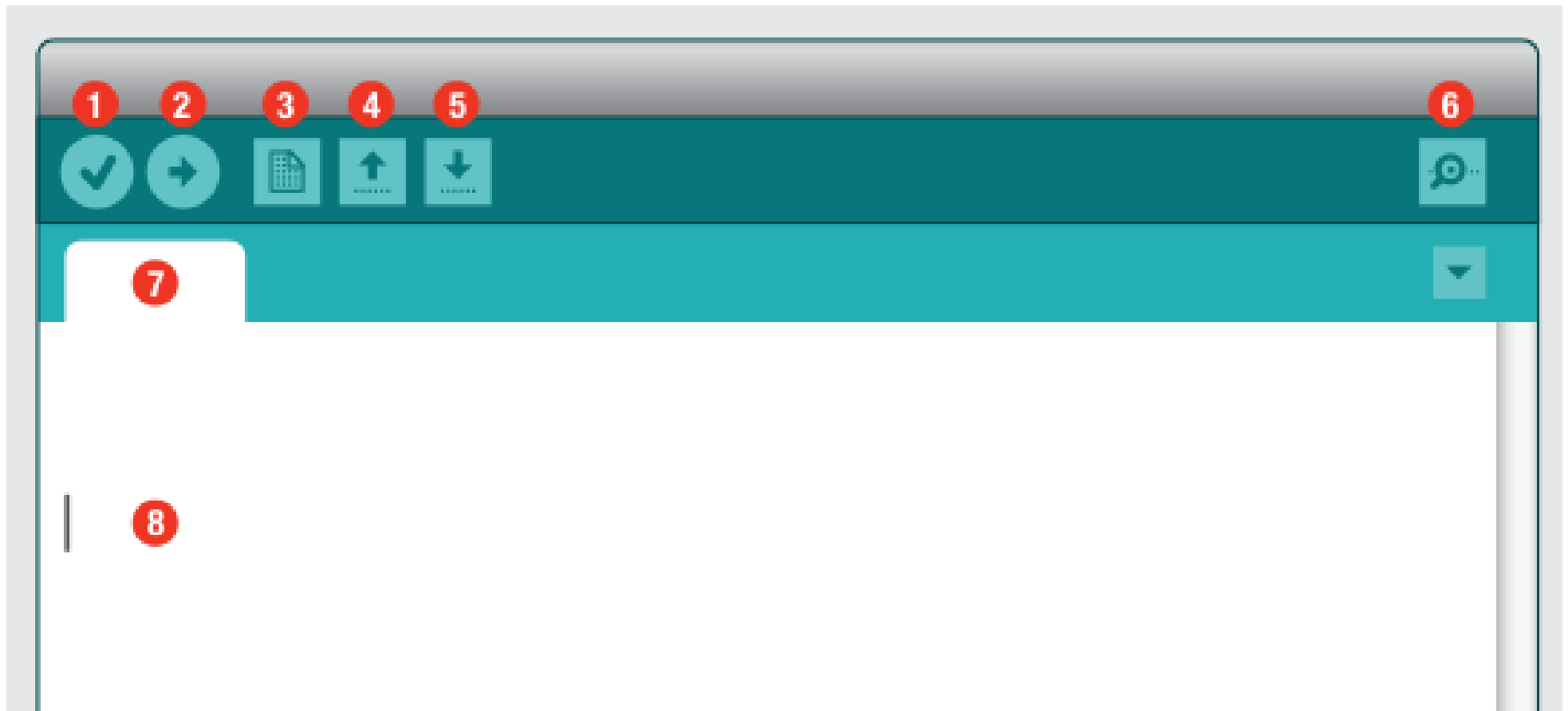
2. "upload" - loads program onto the Arduino

# The Arduino IDE (Integrated Development Environment)



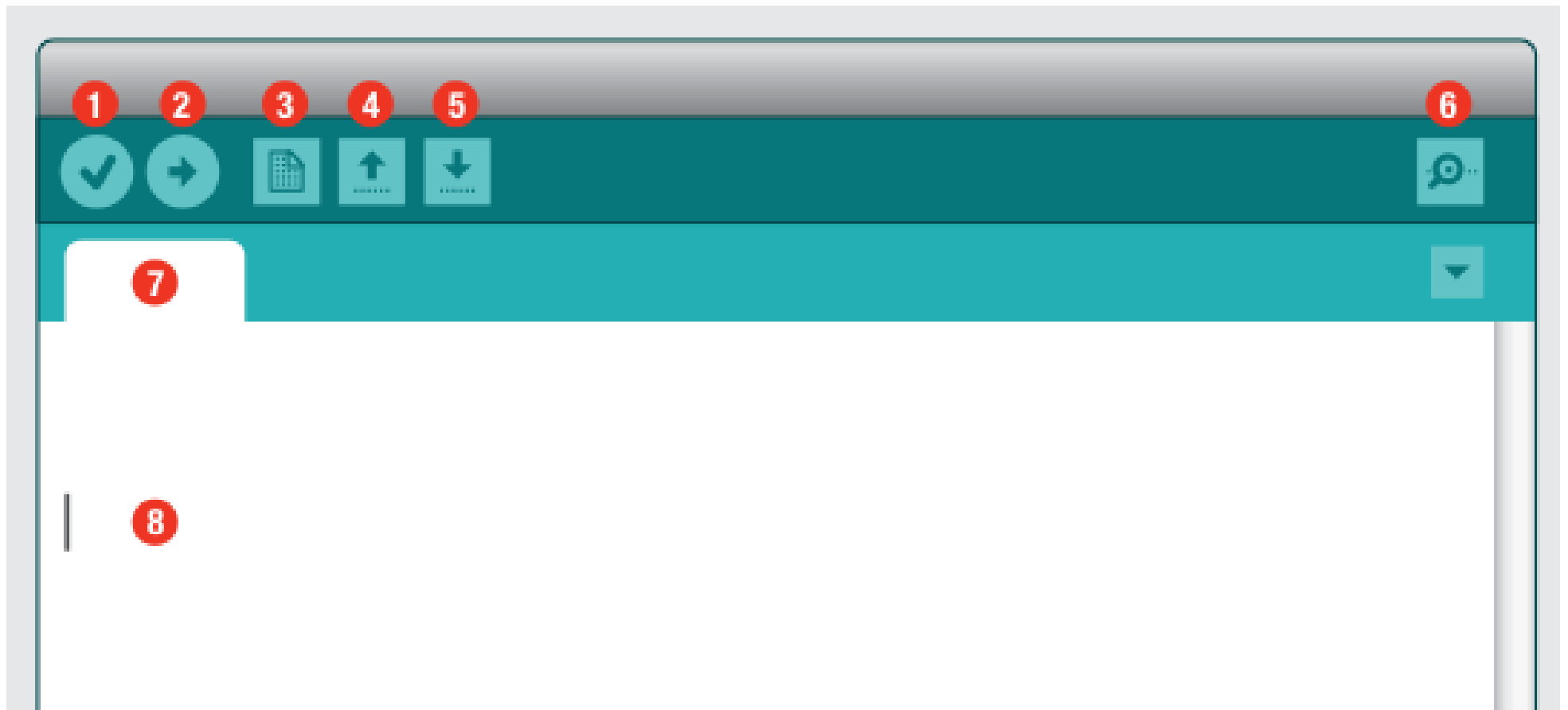
3. "New" - creates a new sketch

# The Arduino IDE (Integrated Development Environment)



5. "Save" - save the current sketch

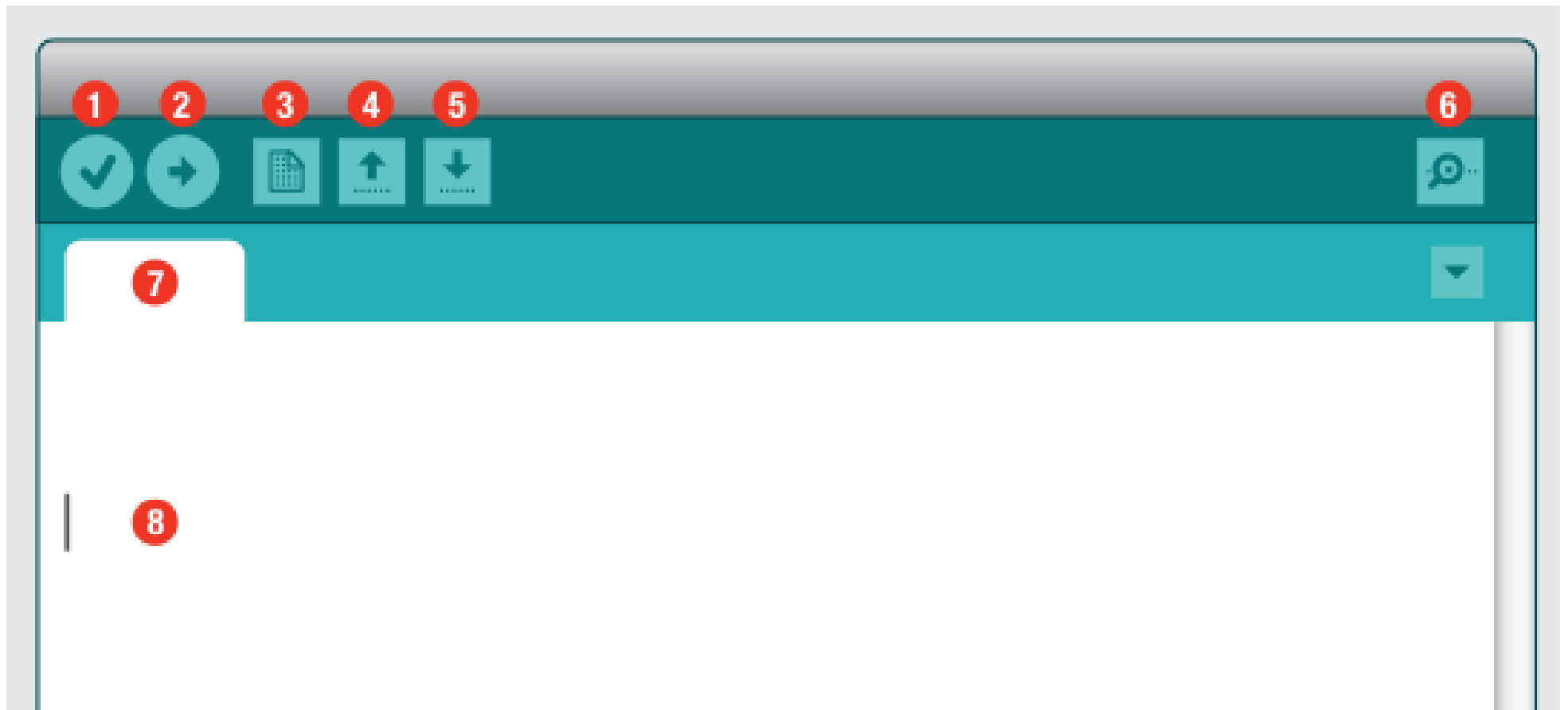
# The Arduino IDE (Integrated Development Environment)



6. "Serial Monitor" - communicate with the Arduino



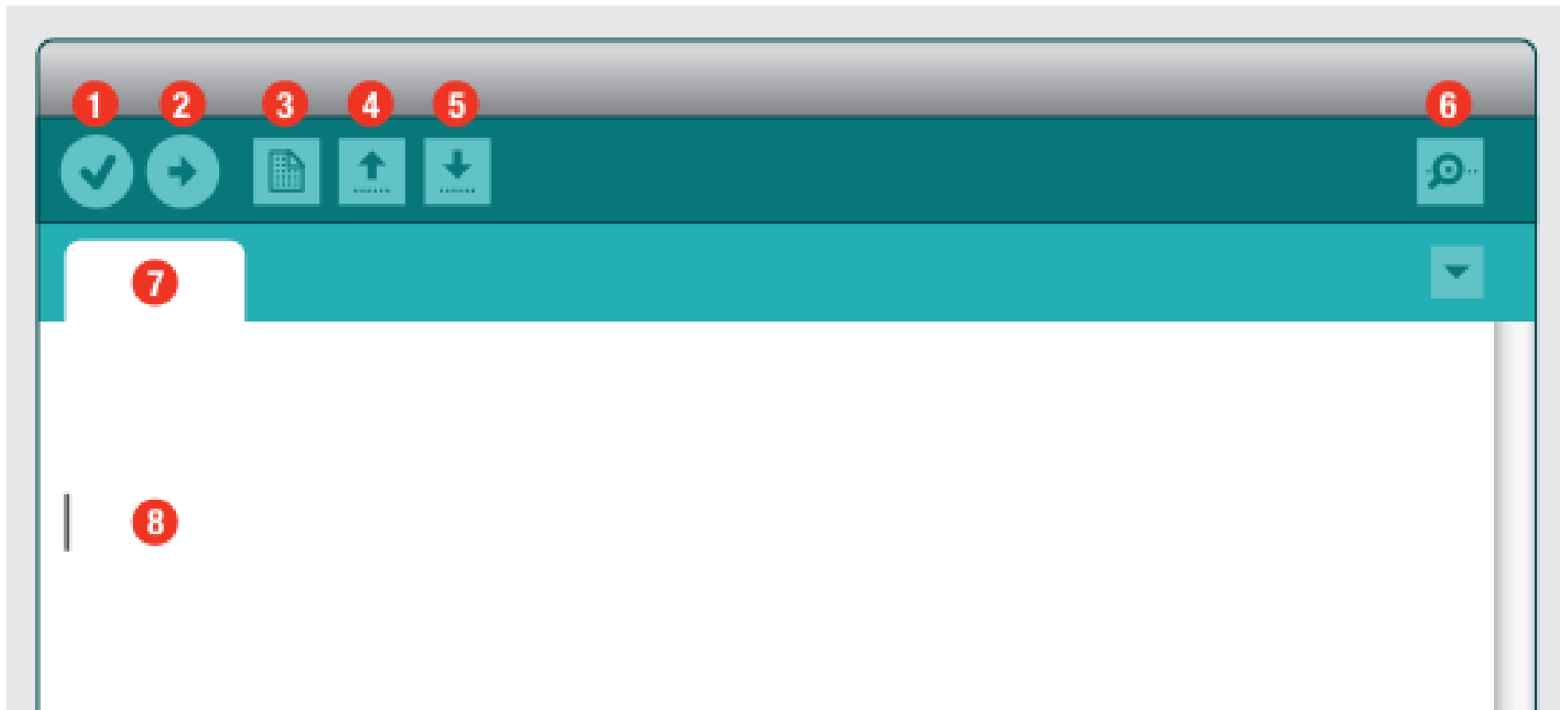
# The Arduino IDE (Integrated Development Environment)



7- "Sketch Name" – Name of the current sketch

# The Arduino IDE

## (Integrated Development Environment)



8. "Editing Window" - edit your program here.

# Anatomy of an Arduino Sketch

```
// Global constants and state data
```

```
const int LED = 9;
```

```
// setup function that is called once at power on
```

```
void setup()
```

```
{
```

```
    pinMode(LED, OUTPUT);
```

```
}
```

```
// loop function gets called in a loop
```

```
void loop()
```

```
{
```

```
    // do exciting stuff!
```

```
    digitalWrite(LED, HIGH);
```

```
    delay(500);
```

```
    digitalWrite(LED, LOW);
```

```
    delay(500);
```

```
}
```

# Anatomy of an Arduino Sketch

Global  
Data

```
// Global constants and state data  
Const int LED = 9;
```

One-time  
Setup

```
// setup function that is called once at power on  
void setup()  
{  
    pinMode(LED, OUTPUT);  
}
```

Loop

```
// loop function gets called in a loop  
void loop()  
{  
    // do exciting stuff!  
    DigitalWrite(LED, HIGH);  
    Delay(500);  
    DigitalWrite(LED, LOW);  
    Delay(500);  
}
```