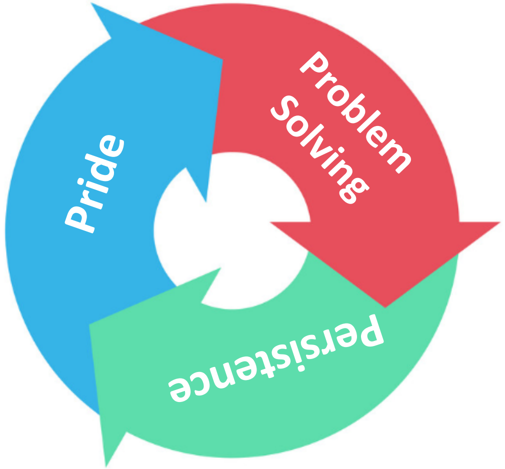
Become an Arduino Wizard!

Kim Brand, 1st Maker Space

1ST MAKER SPACE
Design.
Build.
Sustain.

Why Arduino? Why Code?

- ***The Arduino is***
  - Open-source: Hundreds of manufacturers, millions of users, widely supported
  - Economical: Get started for under $10
  - Powerful enough
  - Evolving: Embedded, Wearable, IoT, Industrial PLC, etc.
  - Safe!

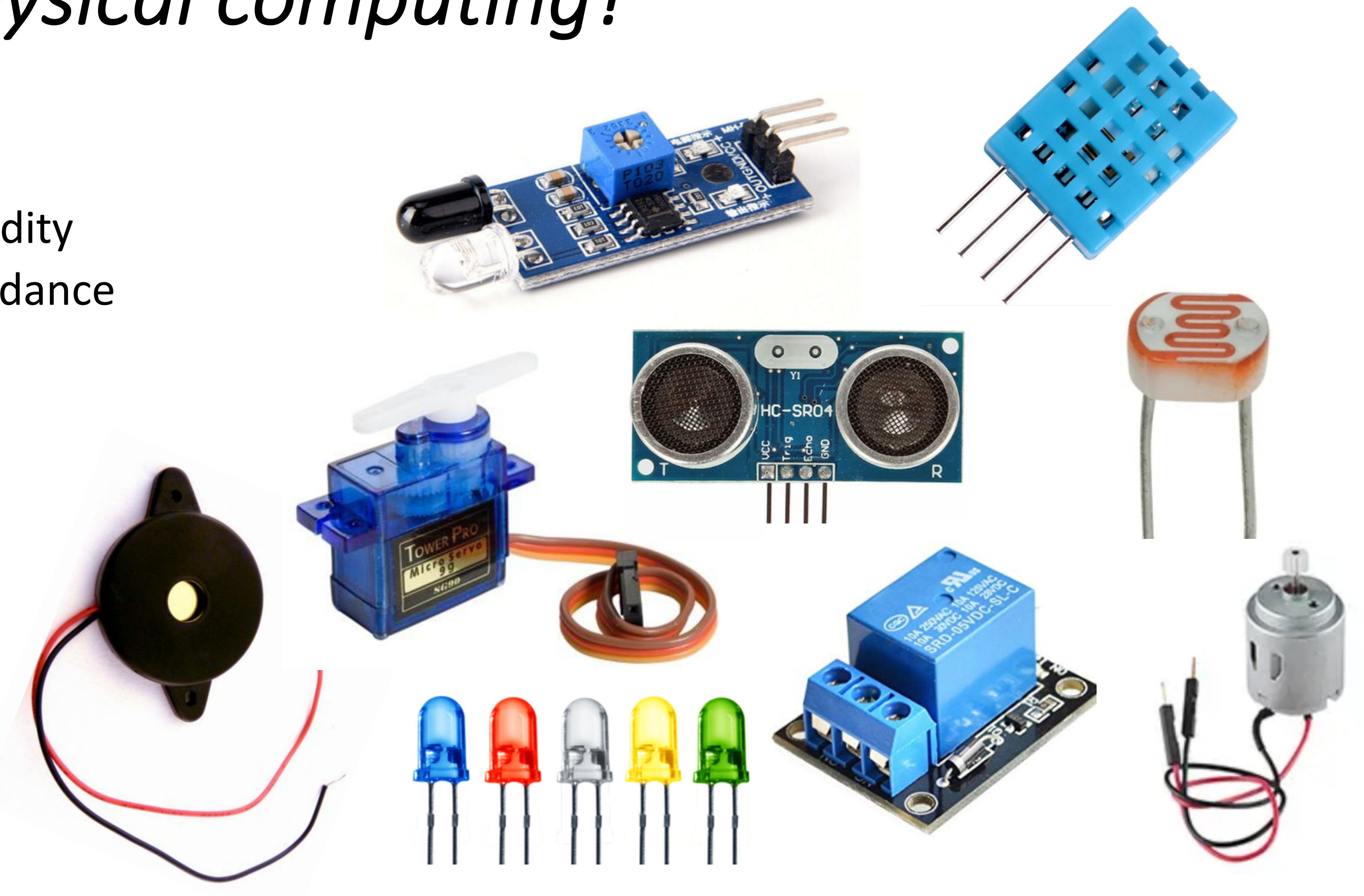


- ***Coding is***
  - Control
  - Confidence
  - Nearly perfect ‘Zone of Proximal Development’ challenges
  - Workforce-ready (New Arduino Certification)

1ST MAKER SPACE
Design.
Build.
Sustain.

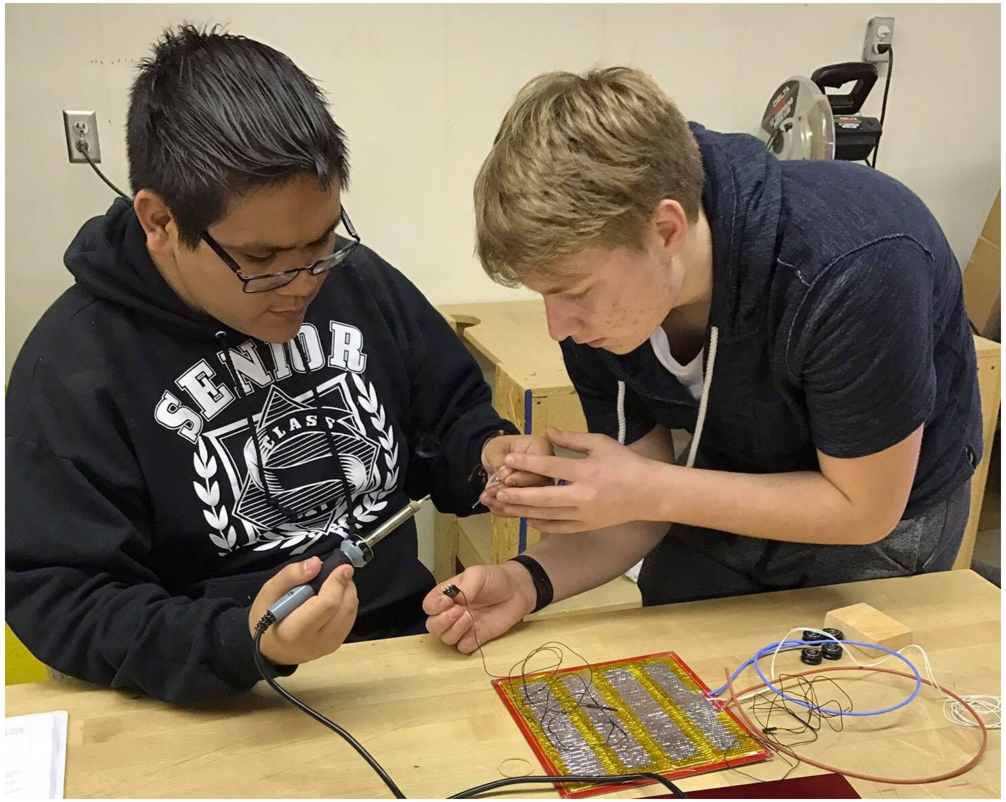
What is physical computing?

- Sensors
  - Temp & Humidity
  - Obstacle Avoidance
  - Sonar
  - Light
- Actuators
  - Servos
  - Relays
  - Motors
- Light & Sound



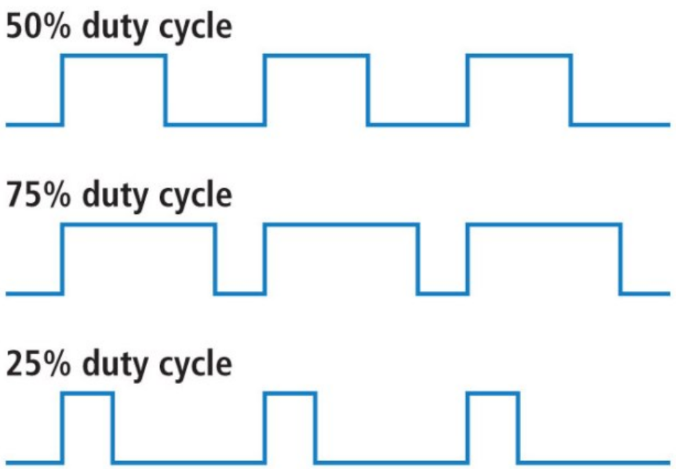
Why is physical computing so important?

- Problems that inspire not discourage!
- Surprises
- Collaboration
- Competition
- Engagement
- Relevance
- Exhibit potential

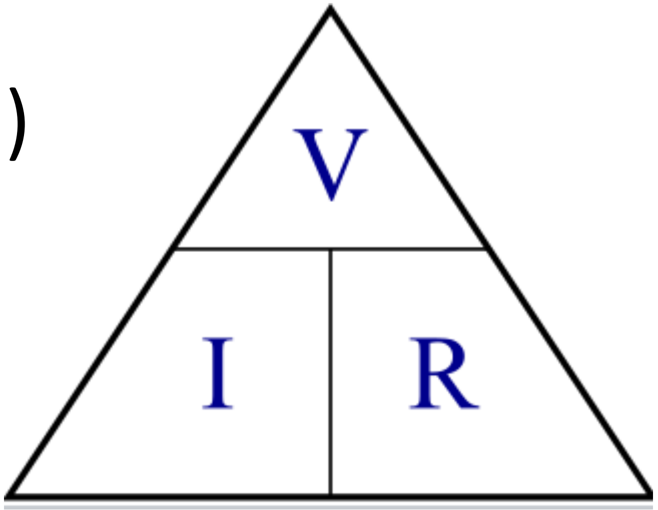


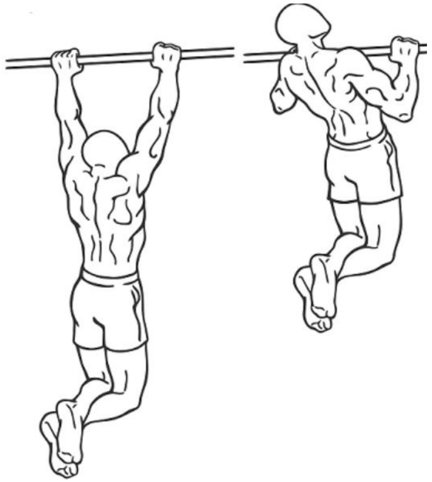
Arduino Video: Massimo Banzi, co-founder

Essential Arduino Vocabulary



- GPIO & PWM
- Github.com
- Ohms Law
- JSON Files (Arduino IDE Config)
- IoT – Internet of Things (ESP8266, ESP32)
- ADC – DAC (Analog – Digital Conversion)
- Pull-Up Resistors
- IDE – Integrated
  Development
  Environment





1ST MAKER SPACE
Design.
Build.
Sustain.

Microcontroller / Electronics Concepts

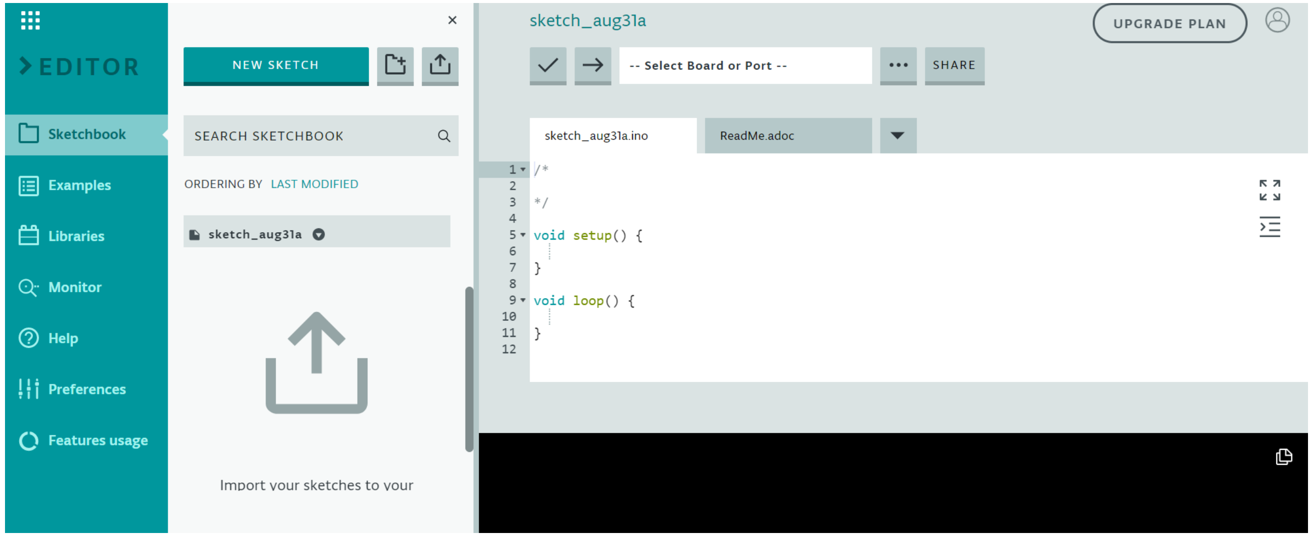
- On – Off, Yes – No, High – Low, True – False, One and zero . . . **Logic**!
- **Voltage** –
  - 5v is On/Yes/High/True/One –
  - 0 v is Off/No/Low/False/Zero
- The playground of a computer is measured in **bits**
- **Circuits** – like plumbing with bits (I call it bit plumbing)
- **Input and output** – working in either the real or virtual worlds
- **Programming** is power to tell the computer what to do
- **Data** can be numbers, colors, sound, words, graphics
- **Memory** stores programs & data – even when the power if off!

1ST MAKER SPACE
Design.
Build.
Sustain.

Arduino Create for Chromebooks

- Cloud based IDE
- Web editor & USB device connector
- Project Hub
- Subscription:

  $0.30 - $0.99/student/month (based on # subscriptions)
- Drama



```
/*

*/

void setup() {

}

void loop() {

}

```

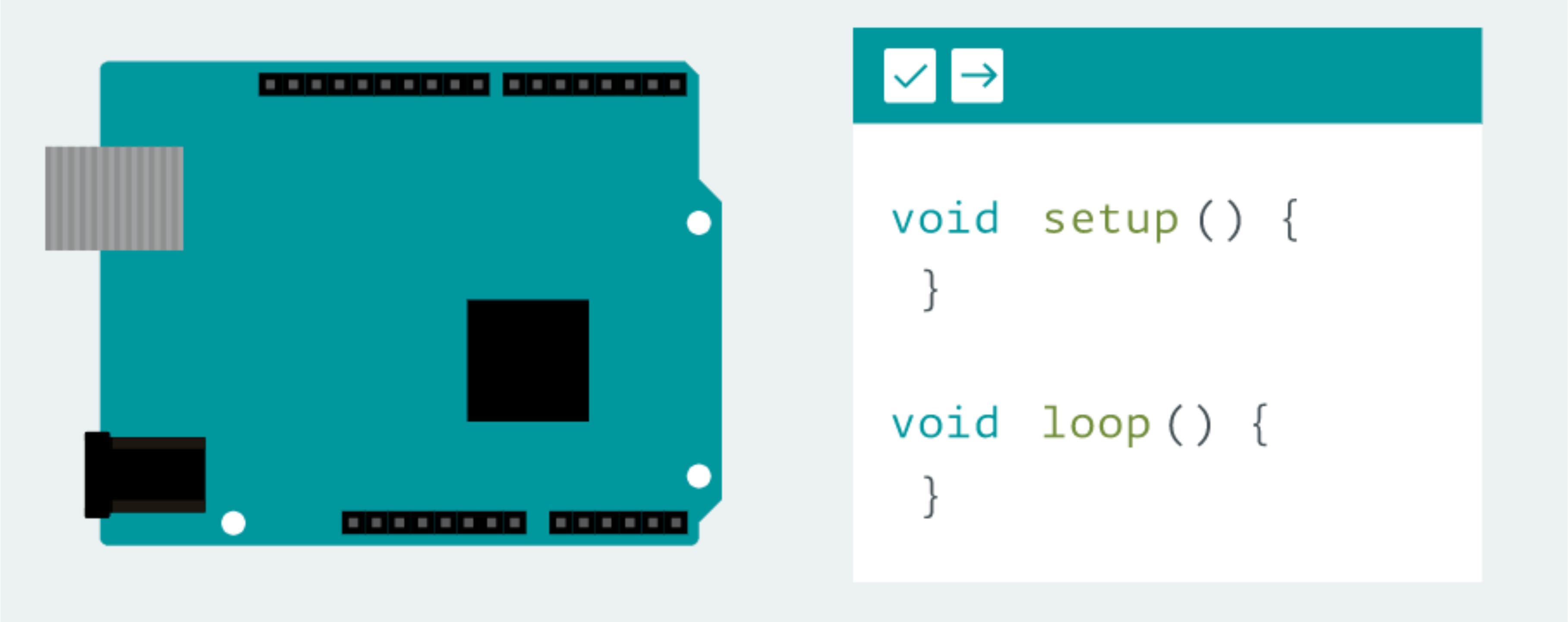
Import your sketches to your

Top 6 Free Arduino Wizard Guides

- Instructables: A Beginner's Guide to Arduino
- Arduino.cc: Foundations & Language Guide
- Adafruit.com: Ladyada's Learn Arduino
- By students: Programming in Arduino
- Brian Evans: Arduino Programming Notebook
- Makerspaces.com: Simple Arduino Projects

1ST MAKER SPACE
Design.
Build.
Sustain.

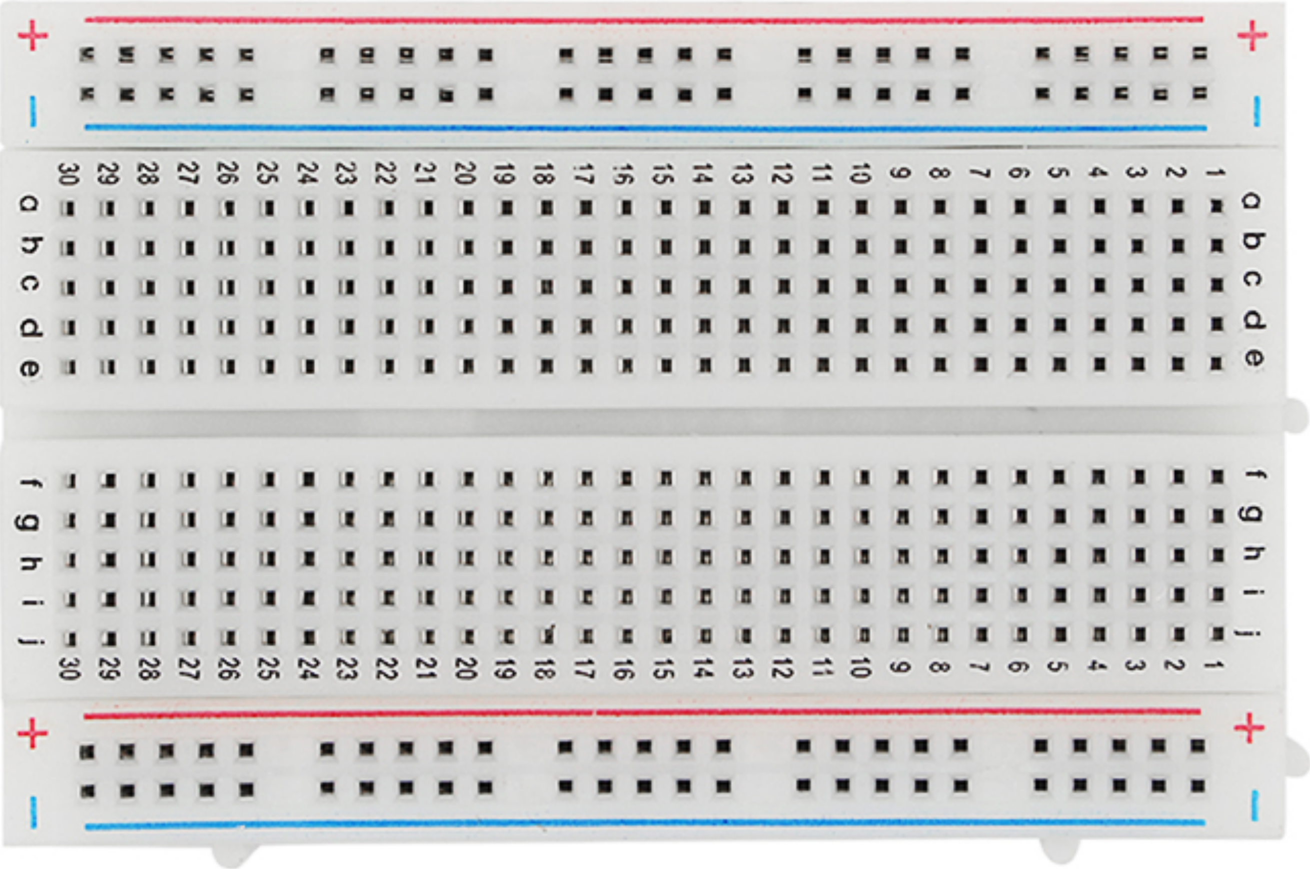
Arduino Physical vs Virtual



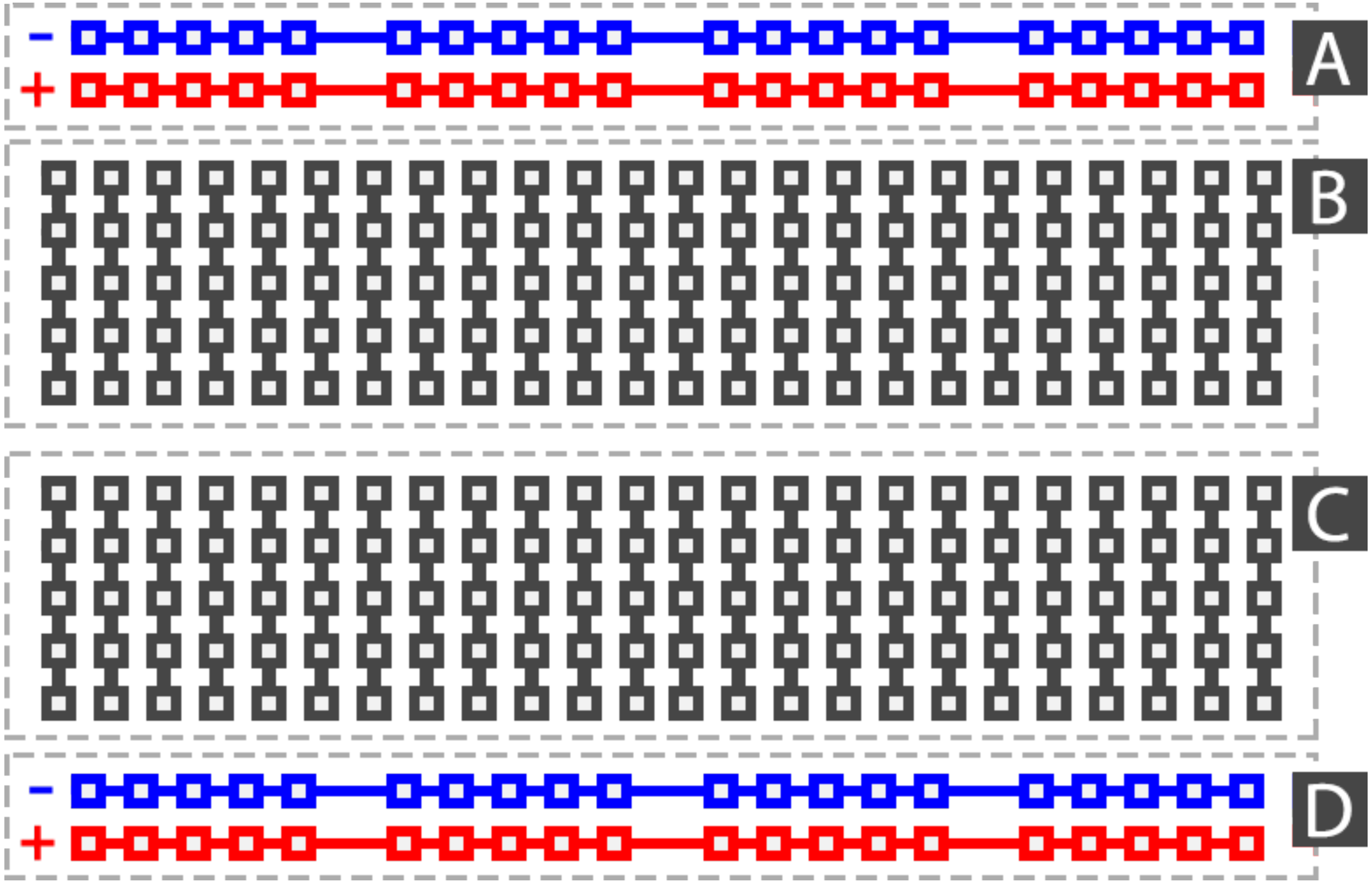
```
void  setup() {
 }

void  loop() {
 }
```

Assembling your first Arduino circuit!

- Solderless breadboards make it easy!



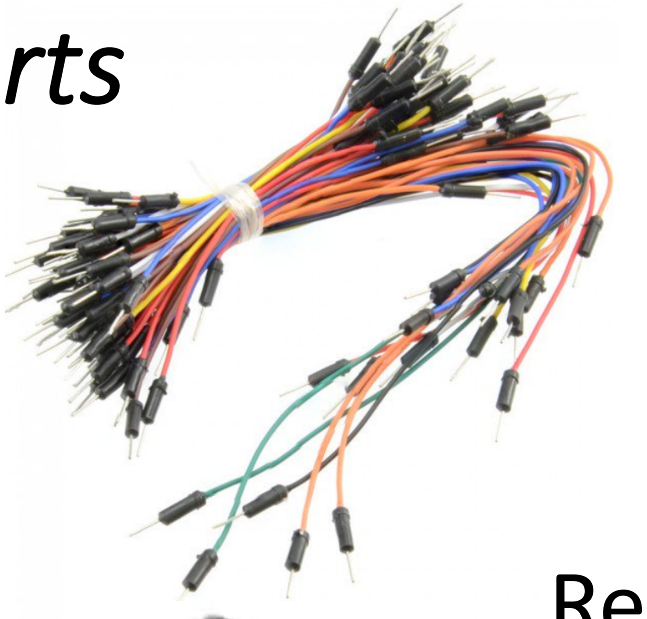
How the holes are connected



1ST MAKER SPACE
Design.
Build.
Sustain.

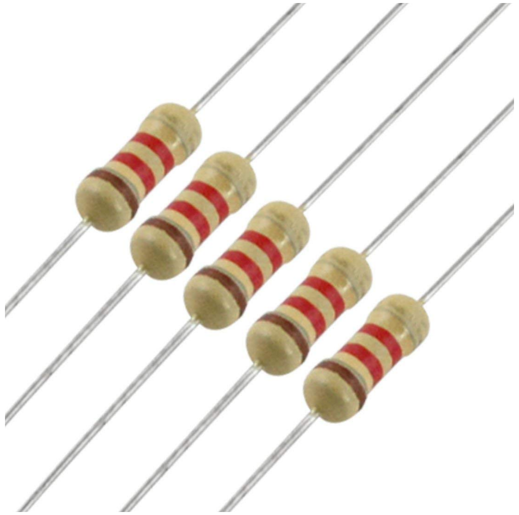
Electronic parts

Connector Wires



Switches

Resistors

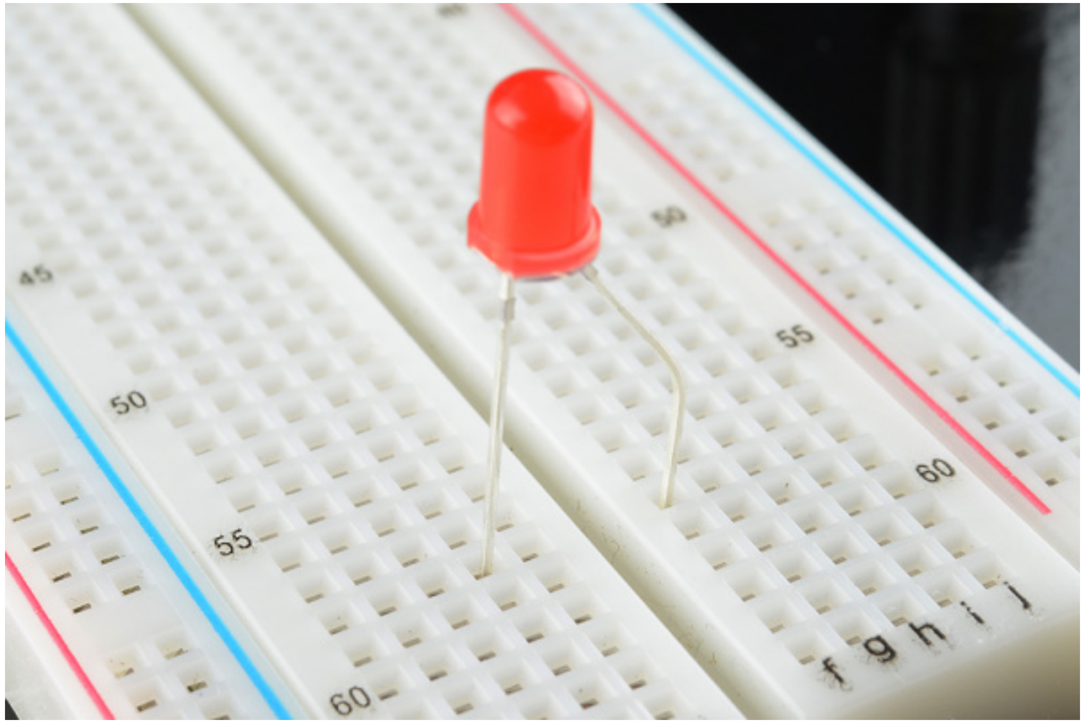


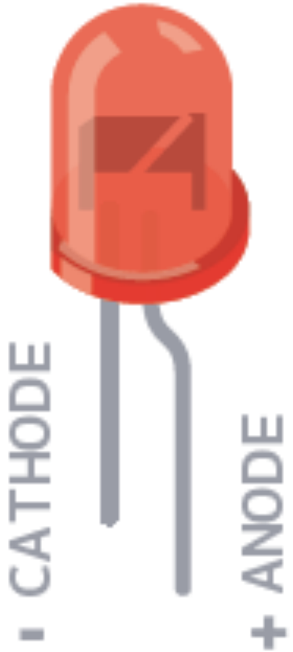
LEDs

How parts are inserted

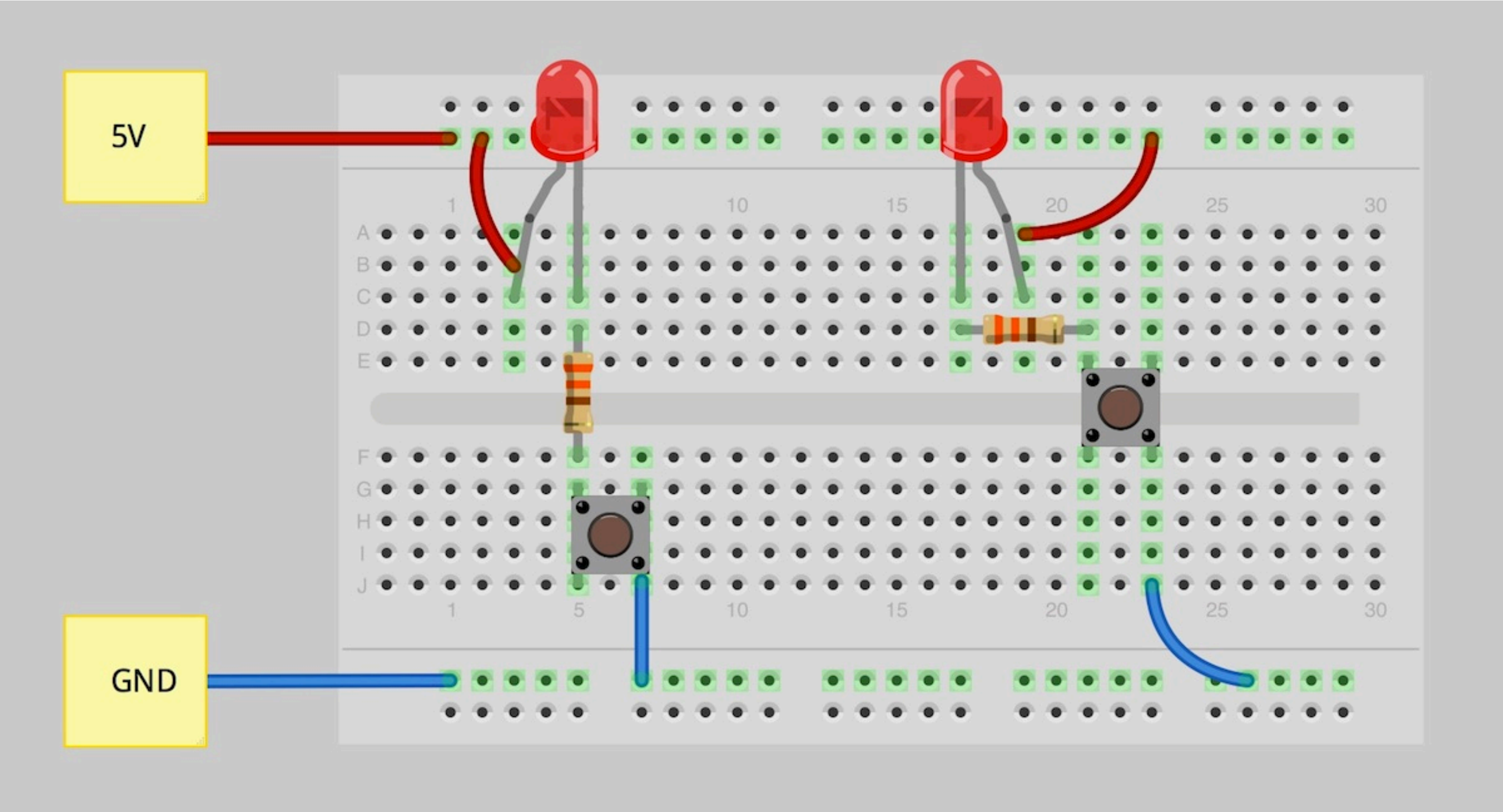
Note: LEDs have a positive and negative (+ / -) Polarity

The longer leg is positive: '+'



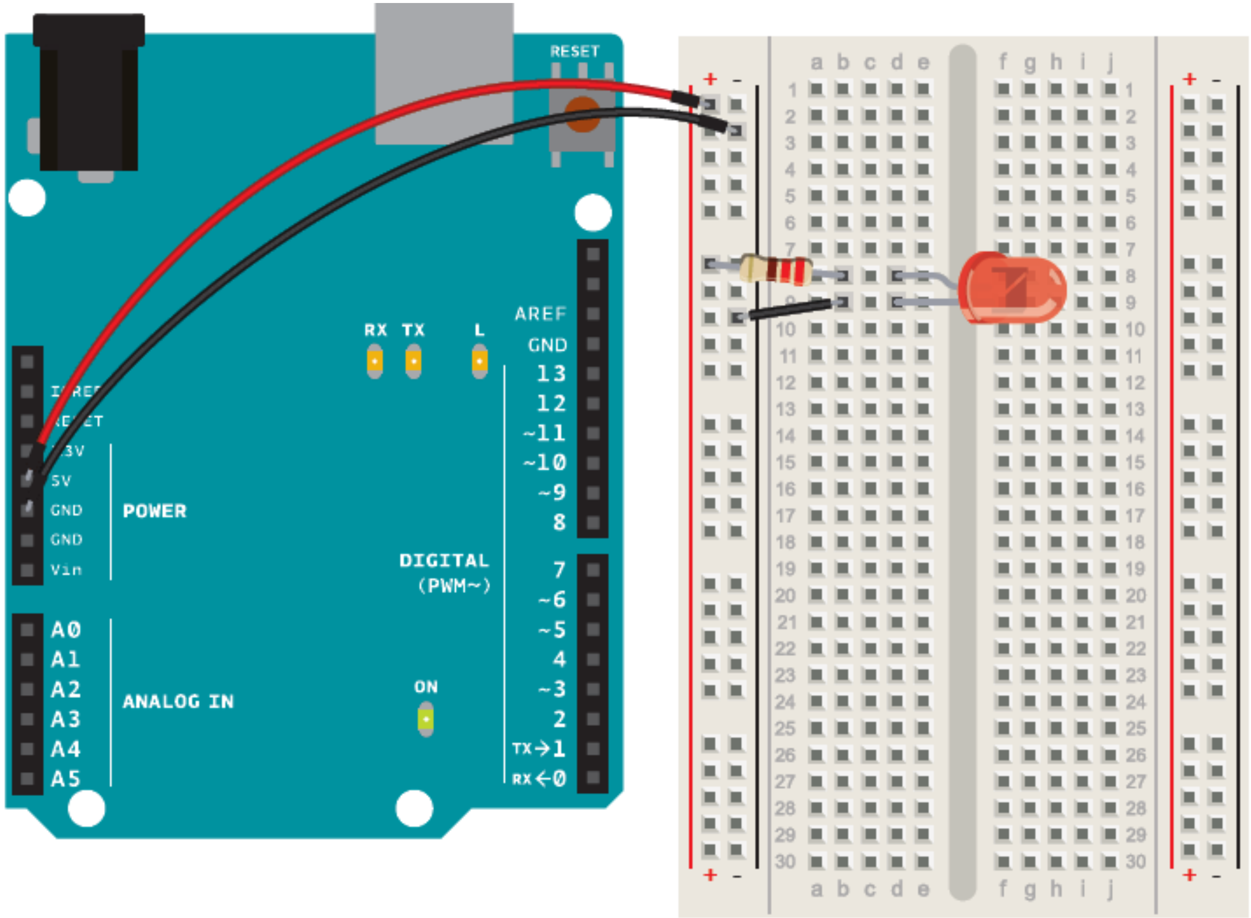


How parts are connected:



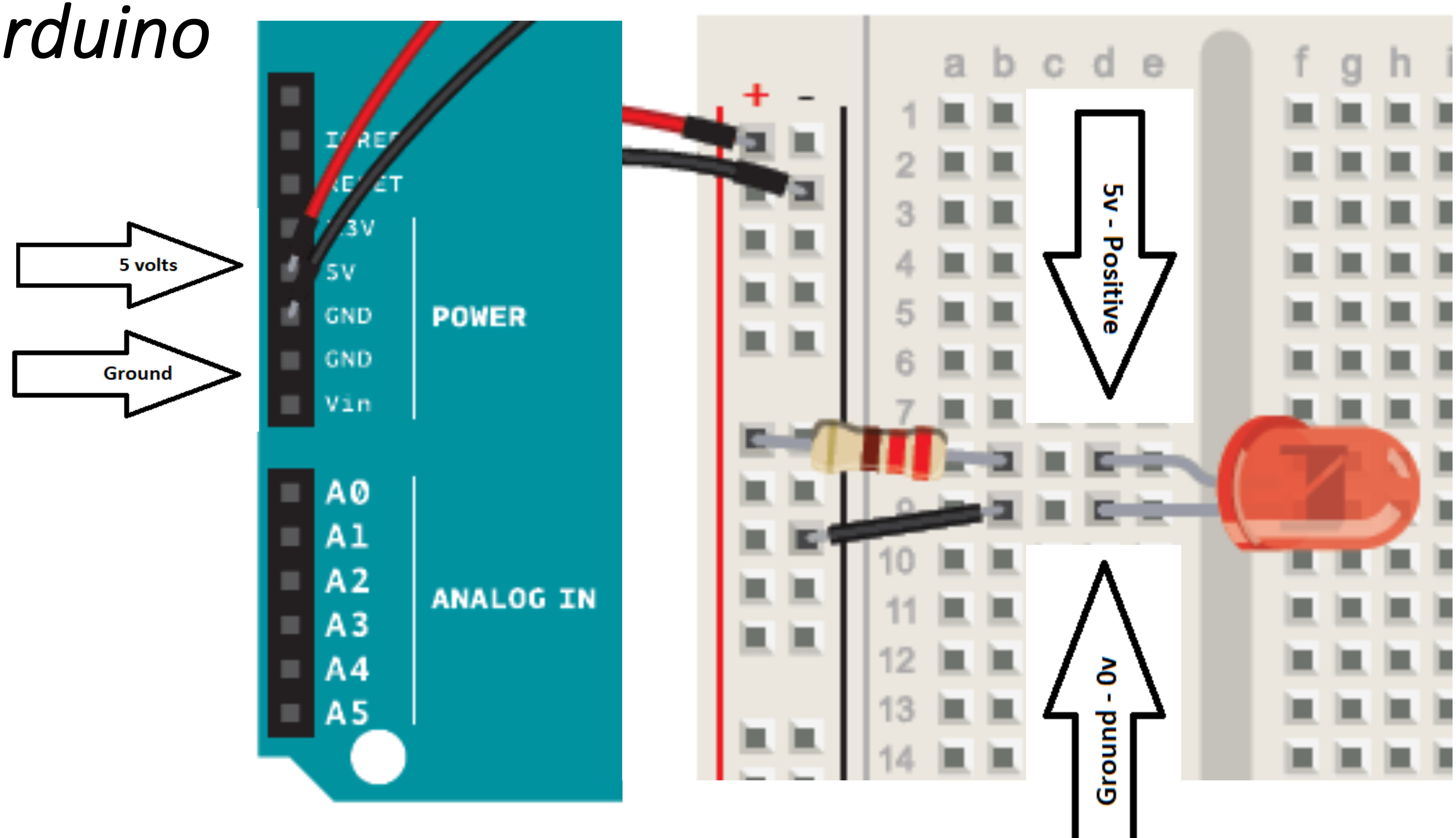
Sometimes Ground (GND) is labeled negative '– ' or 0 volts

Wiring power to the breadboard from the Arduino

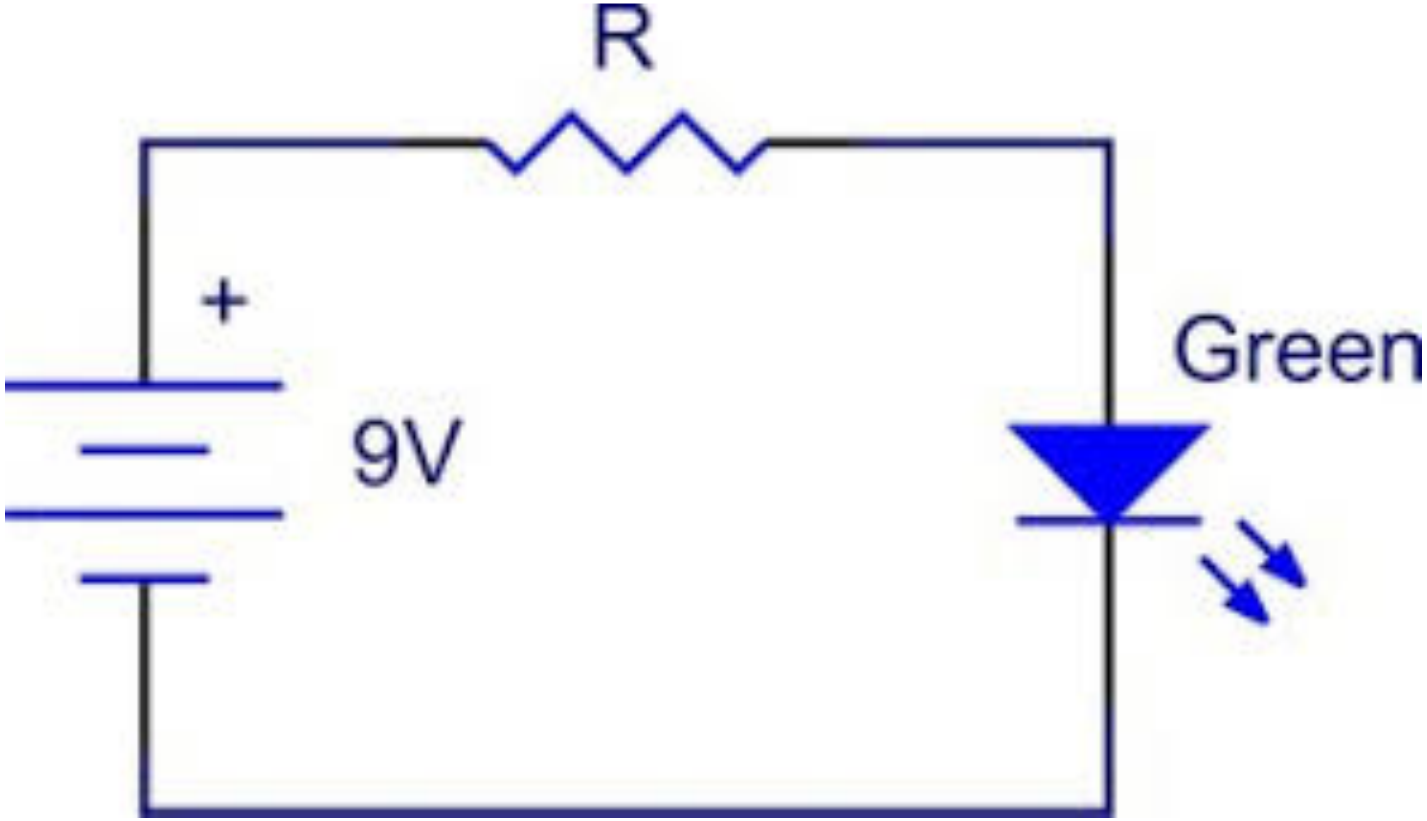


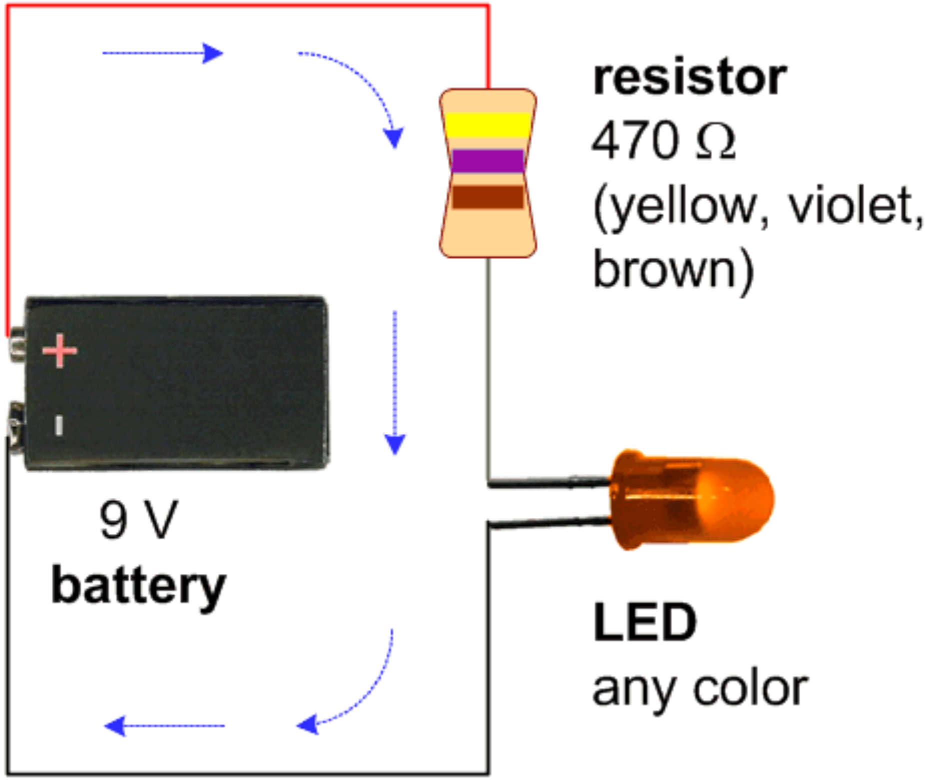
1ST MAKER SPACE
Design.
Build.
Sustain.

Wiring power to the breadboard from the Arduino

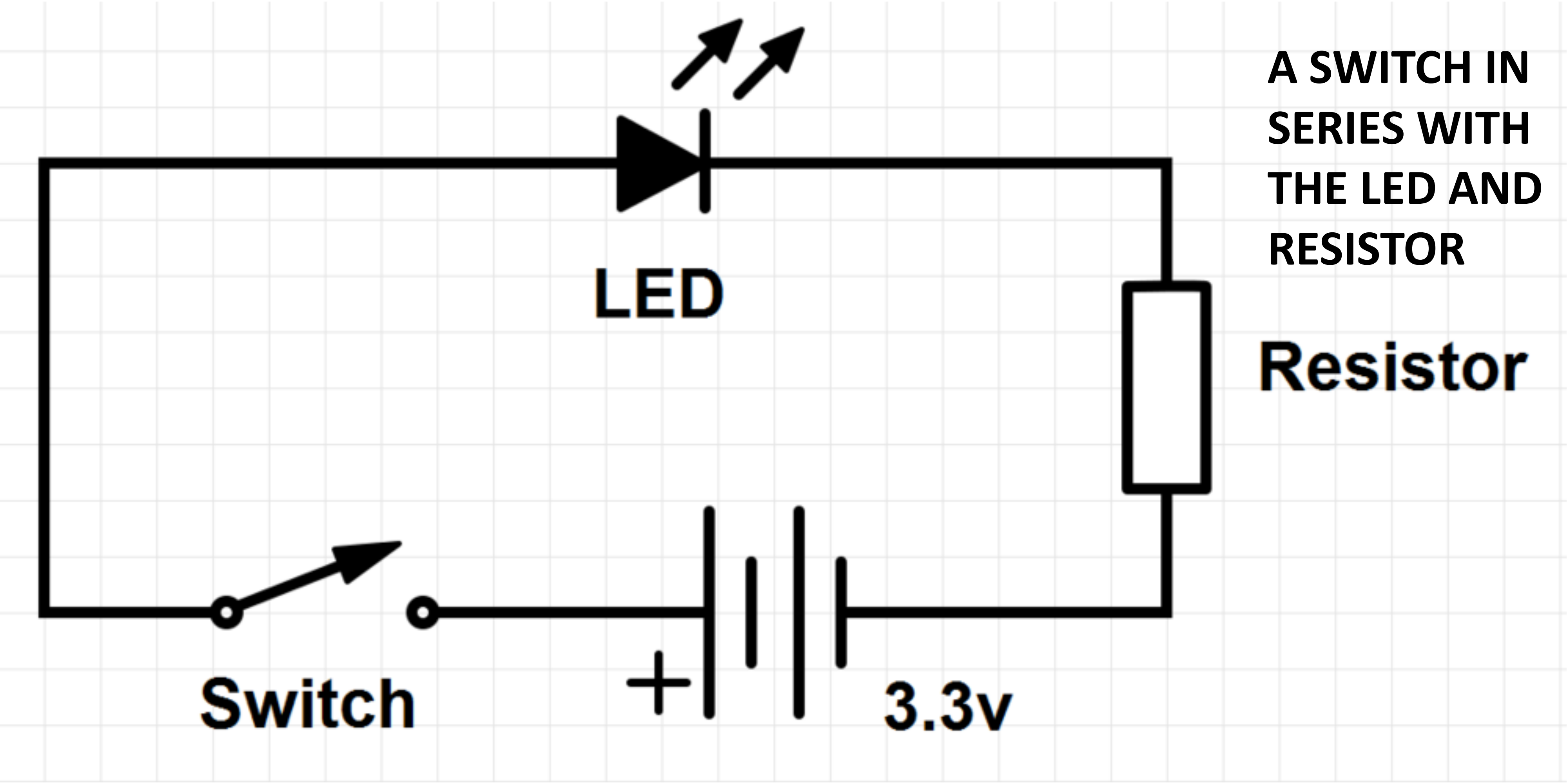


Circuits need to connect to both sides of the power source and with the right polarity: + and –



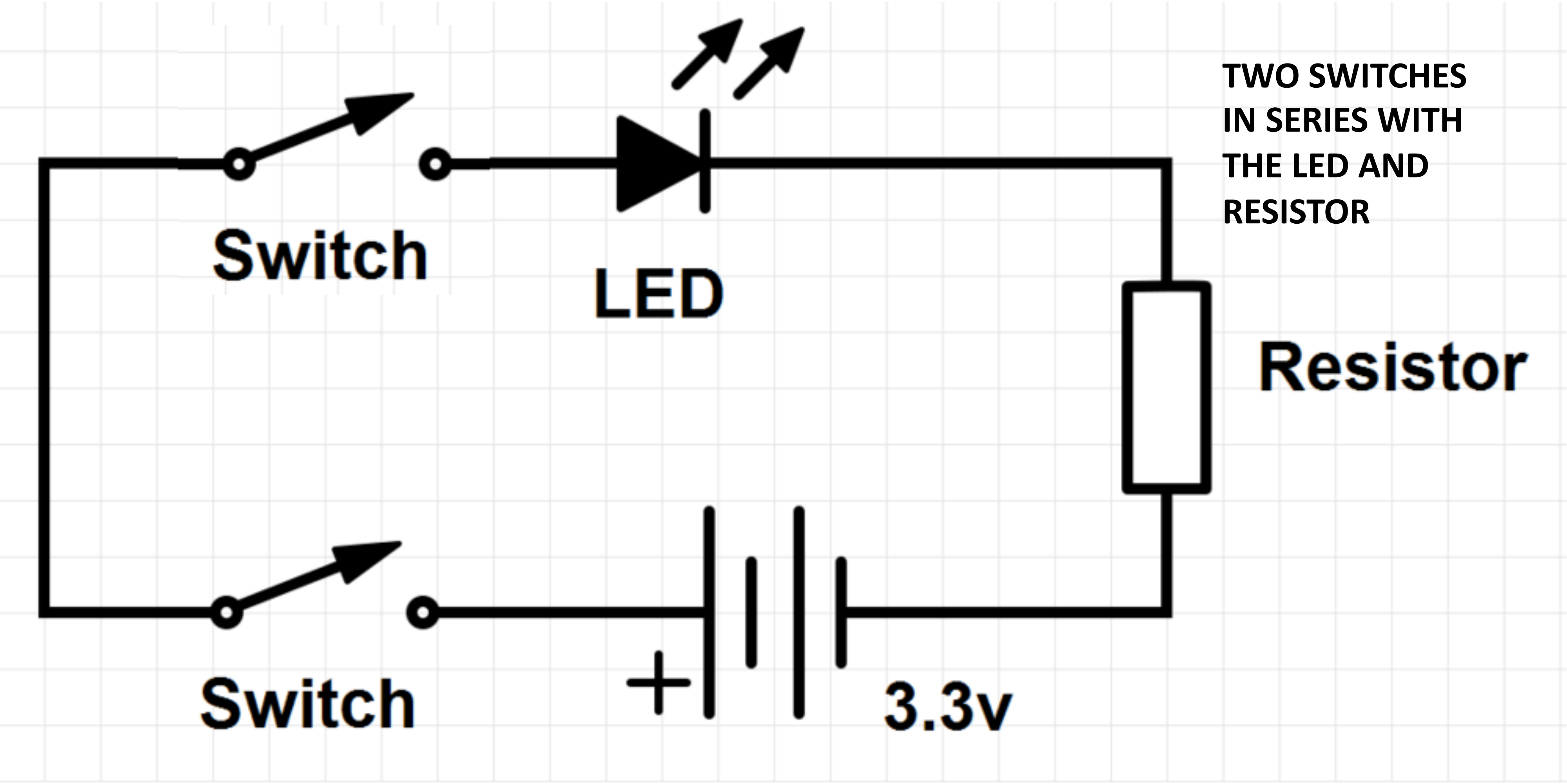


Circuits need to connect to both sides of the battery and with the right polarity: + and –



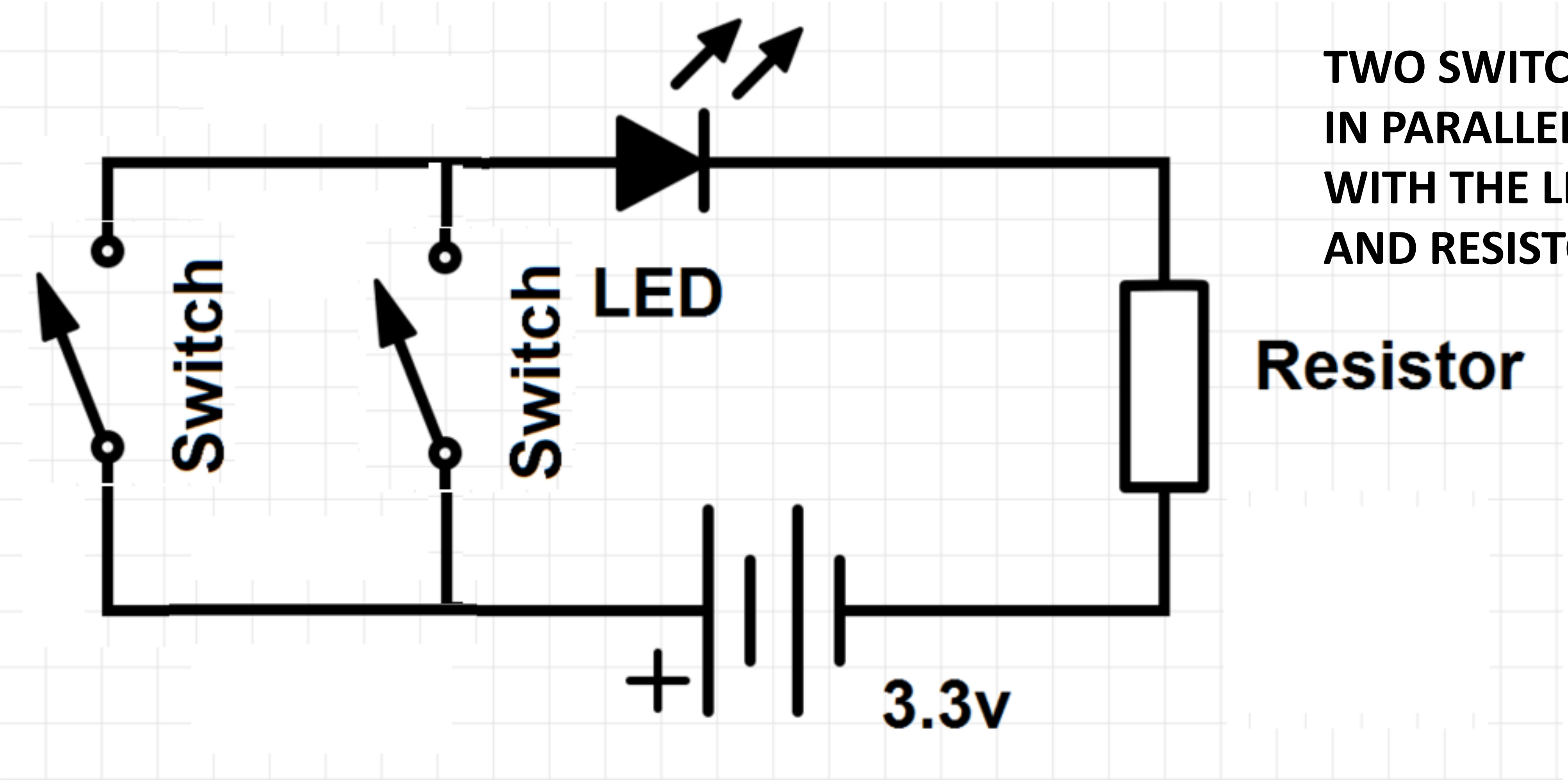
A SWITCH IN SERIES WITH THE LED AND RESISTOR

Circuits need to connect to both sides of the battery and with the right polarity: + and –



TWO SWITCHES IN SERIES WITH THE LED AND RESISTOR

Circuits need to connect to both sides of the battery and with the right polarity: + and –



TWO SWITCHES
IN PARALLEL
WITH THE LED
AND RESISTOR

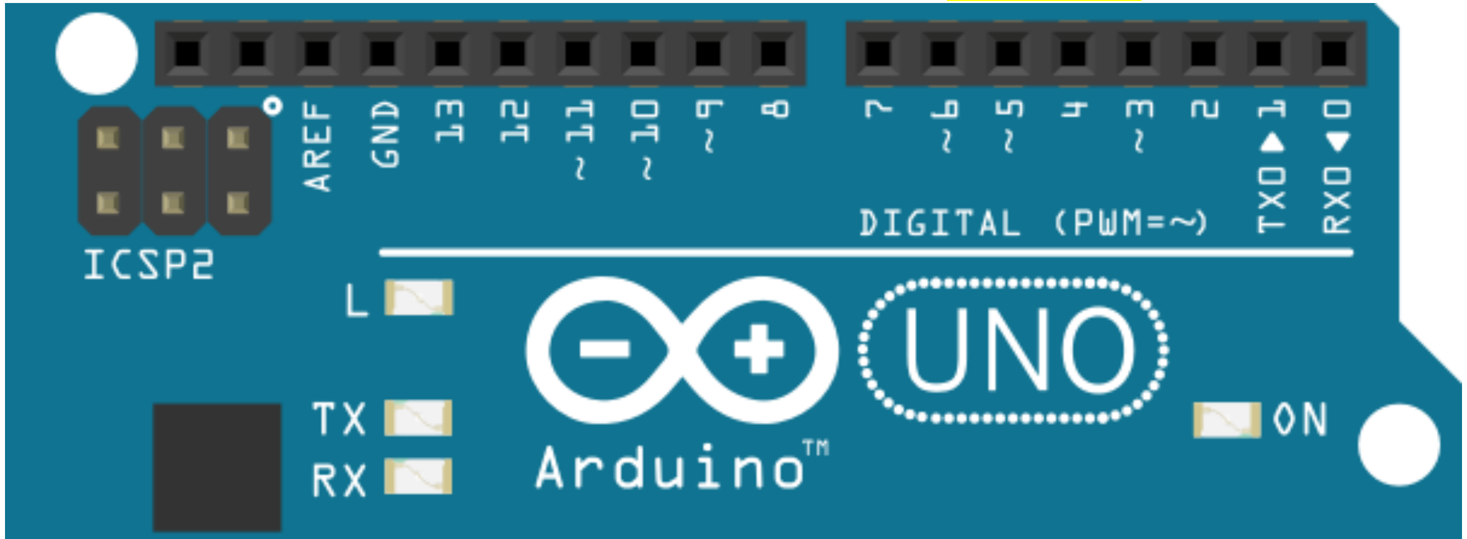
Programming the Arduino – commands

setup()	Initialize the Arduino
loop()	Do over and over – the ‘program’
pinMode(pin,mode)	pin = PIN #, mode = INPUT or OUTPUT
digitalWrite(pin,value)	pin = PIN #, value = HIGH or LOW

pin may be LED_BUILTIN

or a number: 2 through 7

← PIN # →



Programming the Arduino – flashing the LED

```
// LED_BUILTIN = PORT #13

void setup() {                                  // the setup function only runs once
        pinMode(LED_BUILTIN, OUTPUT);           // initialize digital pin LED_BUILTIN as an
output.
}

void loop() {                                   // the loop function runs over and over again forever
        digitalWrite(LED_BUILTIN, HIGH);        // turn the LED on (HIGH is the voltage level)
        delay(1000);                            // wait for 1000 milliseconds
        digitalWrite(LED_BUILTIN, LOW);         // turn the LED off by making the voltage
LOW
        delay(1000);                            // wait for 1000 milliseconds
}
```

Programming the Arduino – Flashing Port #2

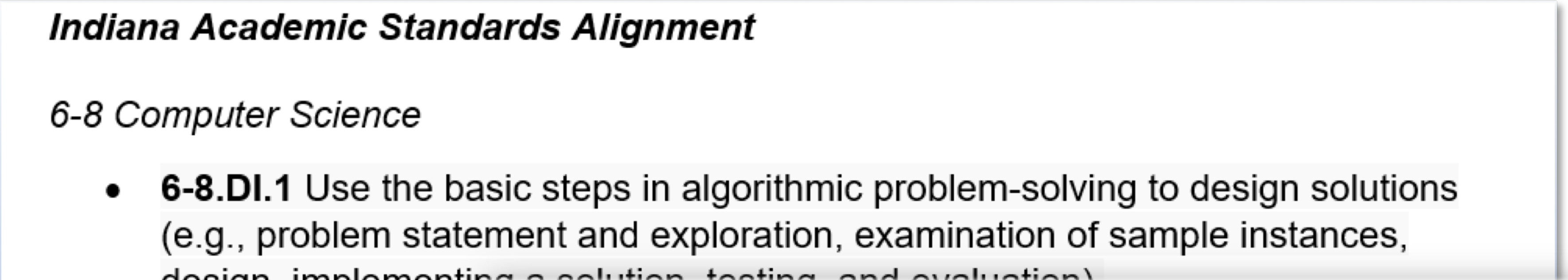
```
// Change LED_BUILTIN to 2 to flash the LED connected to PIN #2

void setup() {                          // the setup function only runs once
        pinMode(2, OUTPUT);             // initialize digital pin 2 as an output.
}

void loop() {                           // the loop function runs over and over again
        digitalWrite(2, HIGH);          // turn the LED on (HIGH is the voltage level)
        delay(1000);                    // wait for 1000 milliseconds
        digitalWrite(2, LOW);                   // turn the LED off by making the
voltage LOW
        delay(1000);                    // wait for 1000 milliseconds
}
```

1ST MAKER SPACE
Design.
Build.
Sustain.

Programming the Arduino – Flashing Port #2 & 3

```
void setup() {                                      // the setup function only runs once
        pinMode(2, OUTPUT);                         // initialize digital pin 2 as an output.
        pinMode(3, OUTPUT);                         // and pin 3
}

void loop() {                              // 'toggle' the LEDs on and off – like a railroad crossing
        digitalWrite(2, HIGH);             // turn on the LED on PIN #2 (HIGH = 5v)
        digitalWrite(3, LOW);              // turn off the LED on PIN #3 (LOW = 0v)
        delay(1000);                       // wait for 1000 milliseconds
        digitalWrite(2, LOW);              // turn off the LED on PIN #2 (LOW = 0v)
        digitalWrite(3, HIGH);             // turn on the LED on PIN #3 (HIGH = 5v)
        delay(1000);                       // wait for 1000 millisecond
}
```

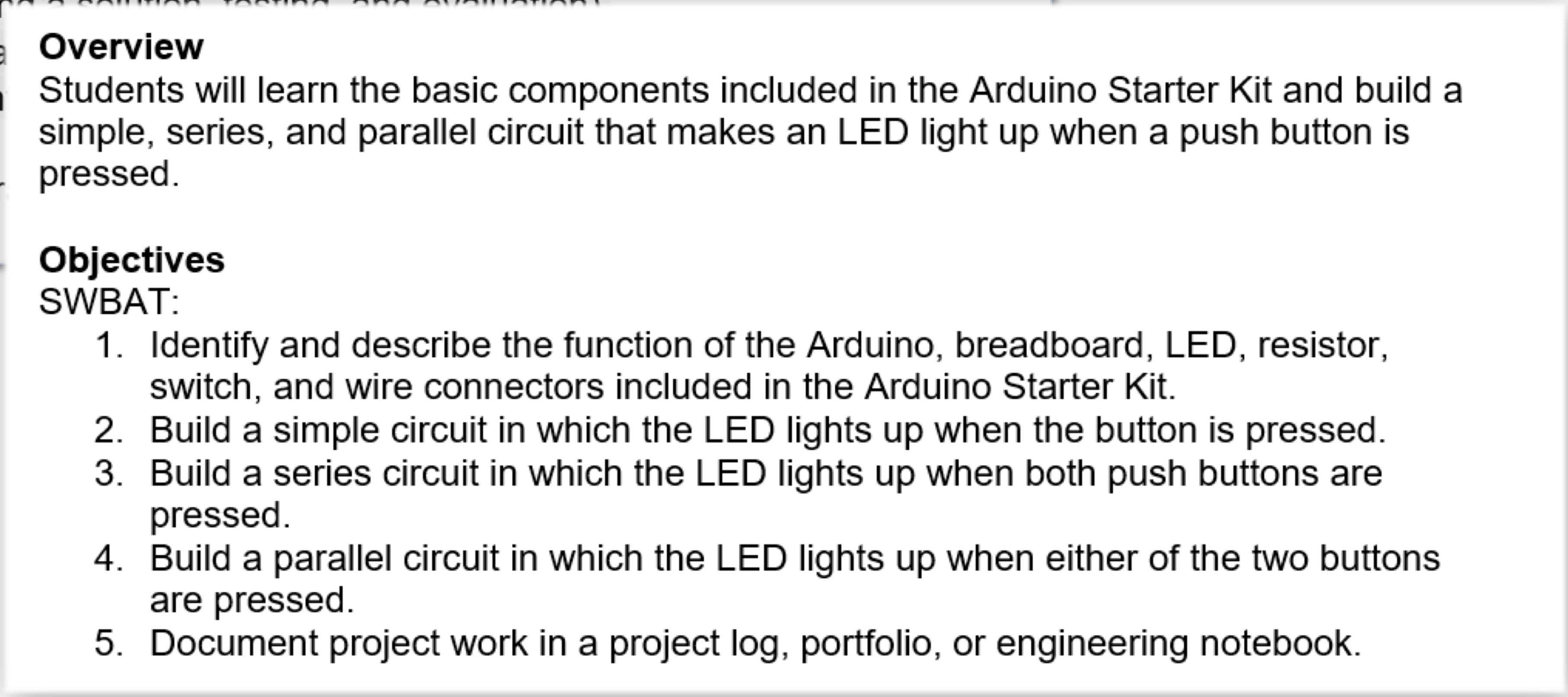
@MakerspaceMary's Arduino Lesson Plans



Indiana Academic Standards Alignment

6-8 Computer Science

- **6-8.DI.1** Use the basic steps in algorithmic problem-solving to design solutions (e.g., problem statement and exploration, examination of sample instances, design, implementing a solution, testing, and evaluation).
- **6-8.DI.5** Demonstra
interact with conten
research.
- **6-8.CD.1** Demonstr
and software.



Overview
Students will learn the basic components included in the Arduino Starter Kit and build a simple, series, and parallel circuit that makes an LED light up when a push button is pressed.

Objectives
SWBAT:

1. Identify and describe the function of the Arduino, breadboard, LED, resistor, switch, and wire connectors included in the Arduino Starter Kit.
2. Build a simple circuit in which the LED lights up when the button is pressed.
3. Build a series circuit in which the LED lights up when both push buttons are pressed.
4. Build a parallel circuit in which the LED lights up when either of the two buttons are pressed.
5. Document project work in a project log, portfolio, or engineering notebook.

1ST MAKER SPACE

Design. Build. Sustain.

Kim Brand
President/CEO
kim@1stmakerspace.com

Mary Rinehart
Education Director
mary@1stmakerspace.com

Share your experience on Social Media!

@1stMakerSpace

#MadeinaMakerspace

www.1stmakerspace.com