

Android
@devfest

```
book-nav-toggle"  
id="" fixed="" aria-label="Hide  
Hide side navigation"
```

LED EXERCISES

GOOGLE DEVFEST 2022

Mr. Ron McKenzie
Teacher, GECD SB

LED EXERCISES

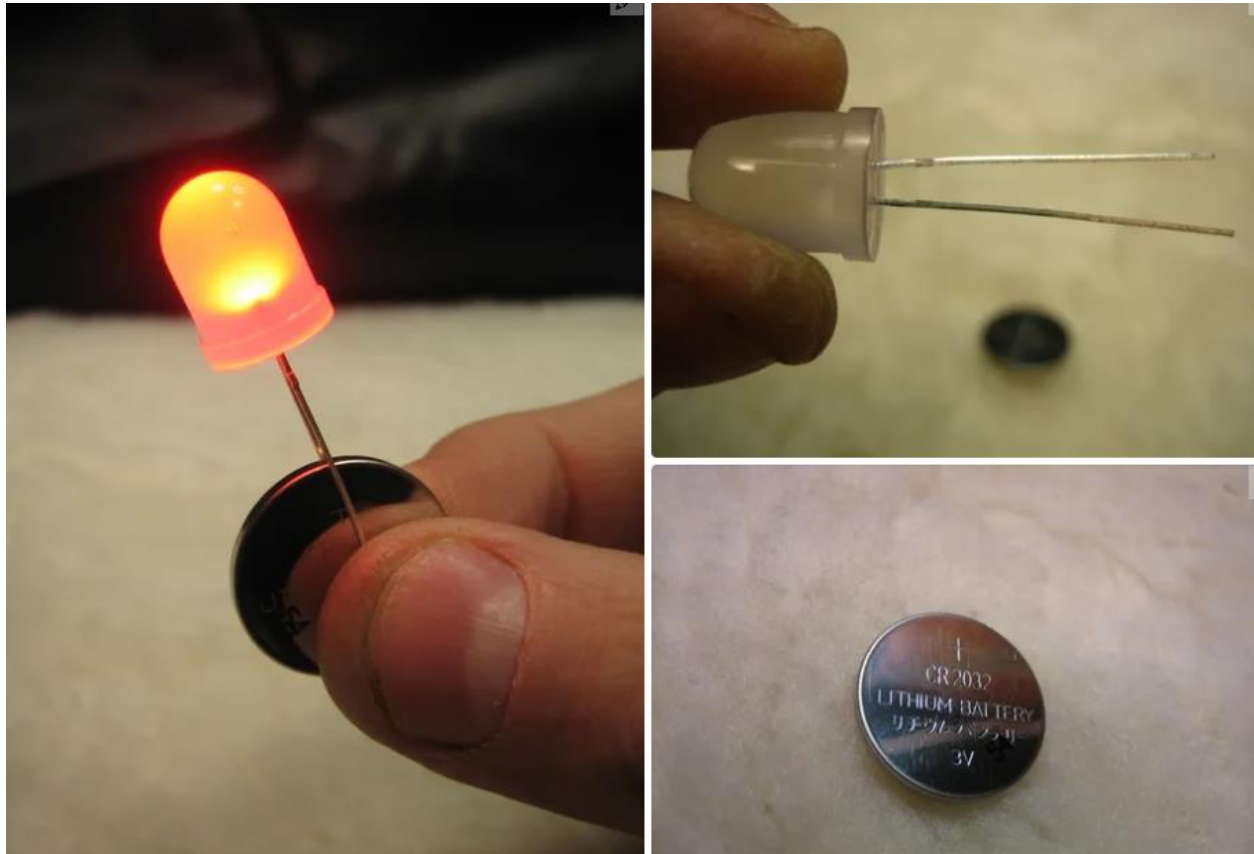
LESSON 1 – THE BREADBOARD

BASIC ELECTRONICS

The magic of electronics is all about what happens when electrons travel from the Cathode (-) to the Anode (+). In electronics we talk about current flowing from + to -. It is important that every circuit you make is complete. You cannot attach a wire to either a positive or negative source then to anything electronic and expect it to work. You must always allow electricity to flow through your device by correctly attaching both + and -.

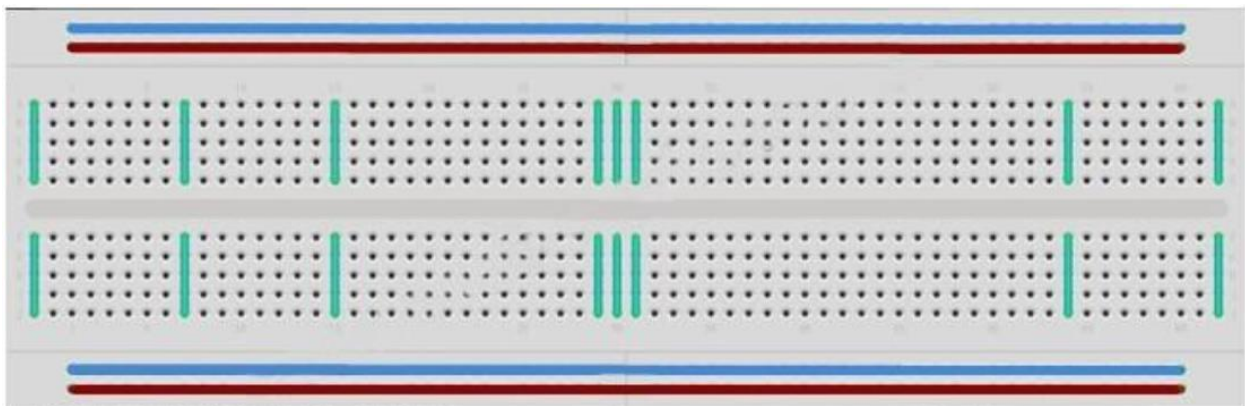
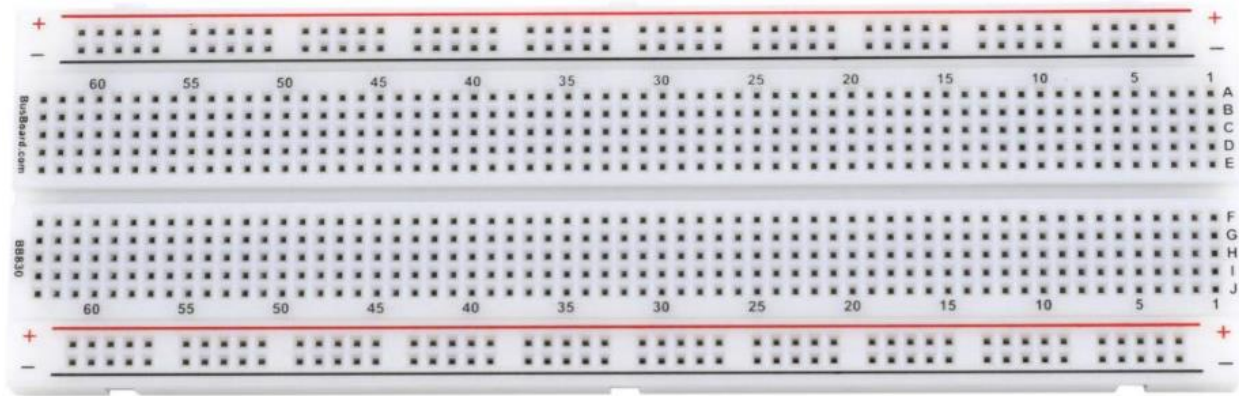
e.g.

The simplest setup: Allowing the current from a 2032 battery to pass through an LED.



THE BREADBOARD

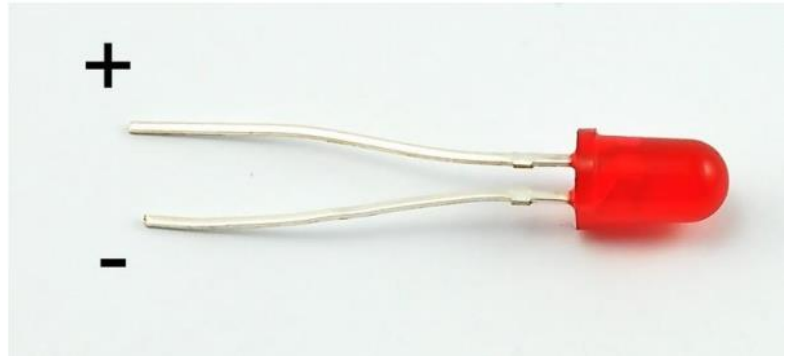
There are many ways to connect circuits. A breadboard can help keep circuits organized. The first picture shows what a standard breadboard looks like, and the second shows how the holes are connected. These holes allow you to insert pins from chips and sensors, leads of LEDs and wires used to connect everything. The top (and/or bottom) lines are connected to + and - by convention, and everything that need + and - get it from there.



LESSON 2 – BASIC LED SETUP

UNDERSTANDING AN LED

LEDs have 2 leads, + and -. If you connect it the wrong way, it will not light up. If you pass too much current through it, it will burn up. These simple ones can only handle about 2V. When using LEDs with basic circuits we will ALWAYS use a resistor to protect the LED.



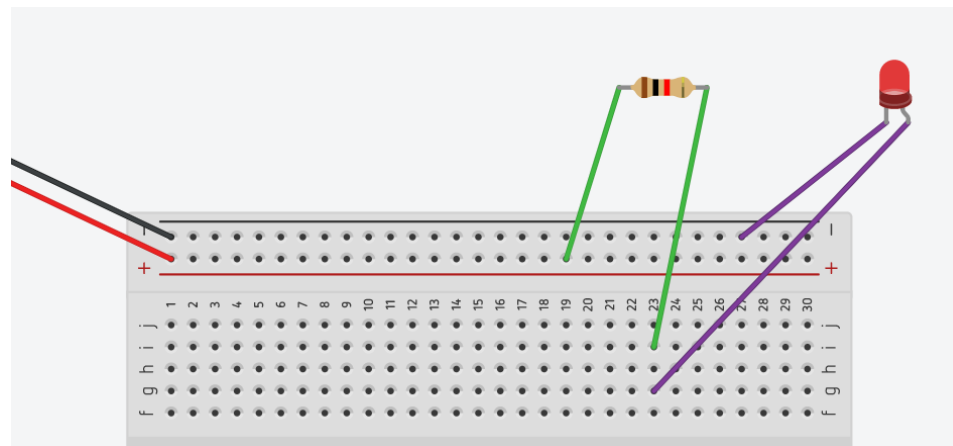
RESISTORS

Resistors, resist the flow of electricity. The higher the number, the more resistance, and thus the dimmer the light. For the sake of simplicity, we will use 220 ohms for bright, and 1k ohms for dim LEDs. The colours on the resistor are a coded way to read their resistance values.



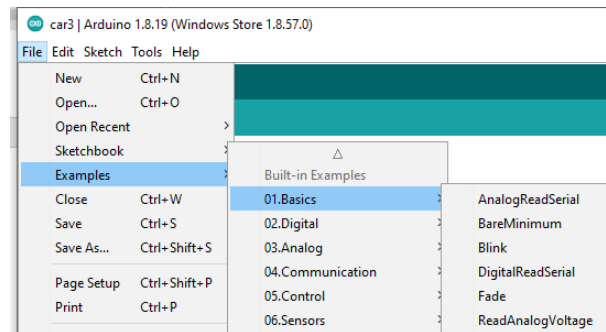
SIMPLE SETUP

When using an Arduino, we commonly use 5V for our power. If we plug in an LED directly into + and - it will burn out. Here our current flows through the resistor first to slow it down.



LESSON 3 – BLINK

- Add Nano to a breadboard with power. Load “Blink” from examples.
- Modify the program to set the LED on the board to blink.



LESSON 4 – ONE AT A TIME

- Add 4 LEDs to the board.
 - o Exercise 1 - Have the LEDs light up, one at a time from left to right
 - o Exercise 2 - Have the light go back and forth
 - o Exercise 3 - Have them light-up in a random order

LESSON 5 – UNDERSTANDING PULSE-WIDTH MODULATION

Digital output pins 3, 5, 6, 9, 10, 11 allow PWM. When setting the values of these pins, use `analogWrite()` with a value from 0-255. E.g. `analogWrite(pin, 128)`

Exercise 1

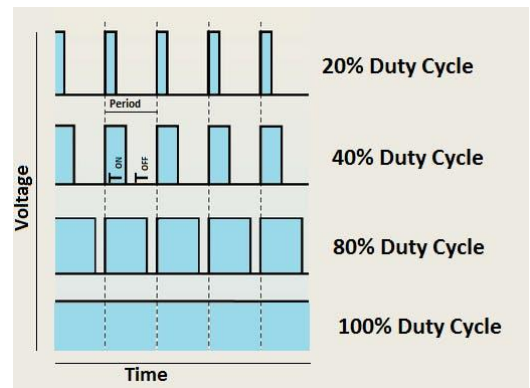
- Change the output pins from lesson 4 to use PWM pins.
- Change the “back and forth” program so that current LED is at half intensity.

Exercise 2

- Change the “back and forth” program so that the LEDs on either side of the current LED are at half intensity, and current one is at full.

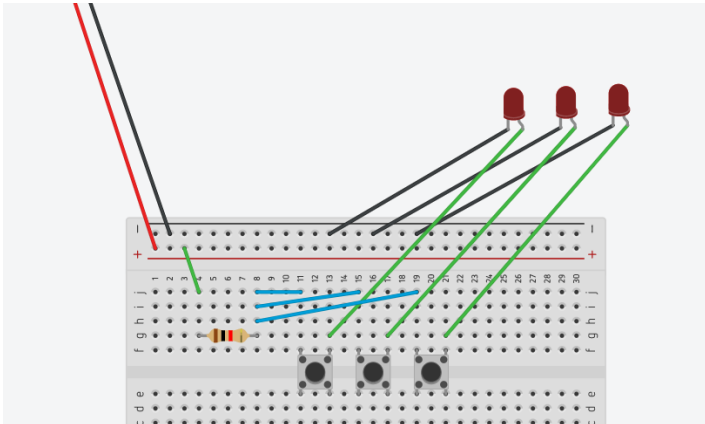
Exercise 3

- Have first LED slowly go from 0 to 255, then turn off and have the next one go from 0 to 255. Continue until the last one follows the pattern.

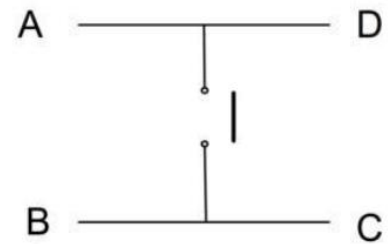


LESSON 6 – USING PUSH BUTTONS

- Start with a clean breadboard
- Add a 3 LEDs and 3 push buttons as shown. Set them up to control the LEDs with the push buttons



- The push button can be a little confusing.



- A and D are already connected at the start, and B and C are connected at the start. When the button is pressed, all 4 become connected. It is common to use these with only 2 pins, A & B or C & D. With 2 pins, it is easier to think about. If you are just using A & B, then when the button is up they are not connected, when it is pressed, they are connected.

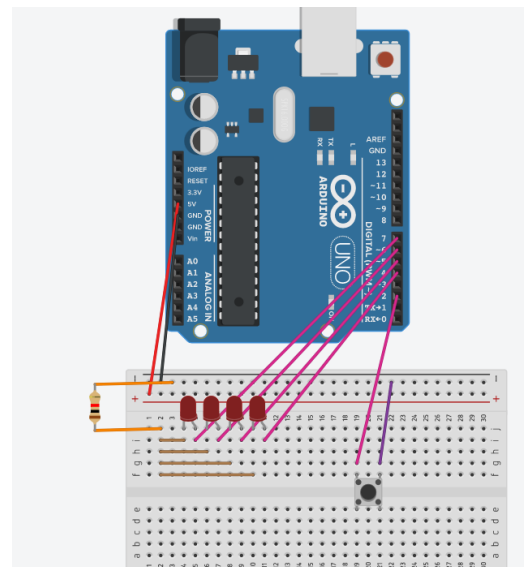
LESSON 7 – ONE PRESS AT A TIME

Exercise 1

- Set up your breadboard as follows:
- Download “One_Press.ino” from https://github.com/mss-mckenzie/DevFest2022/blob/main/LED%20Code/One_Press.ino
- Upload to your nano and experiment

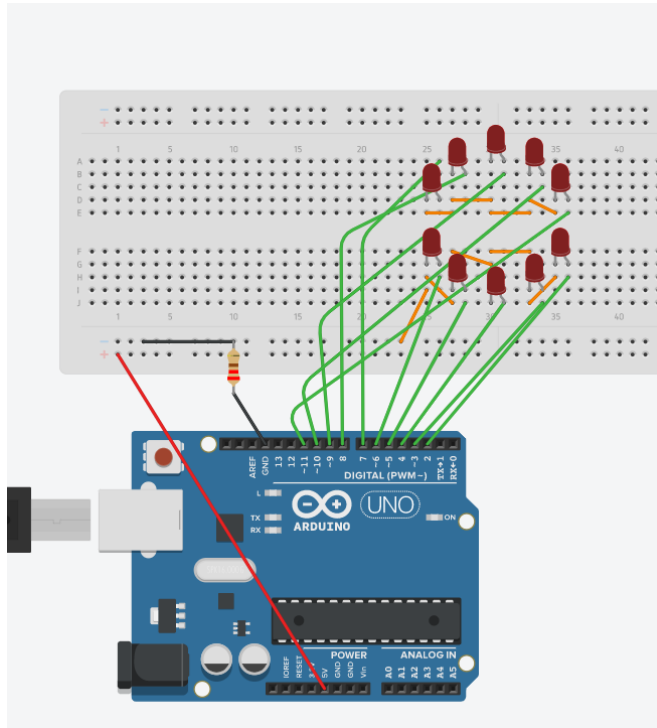
Exercise 2

- Modify the code so that it allows you to switch between the animation patterns from lesson 4.



Exercise 3

- Set up your breadboard with a ring of LEDs:



- Create 3 new patterns and add a push button to control which pattern is showing.

LESSON 8 – COUNTING IN BINARY

- Get a your program to count in binary, from 0-15 with a 500 ms delay between showing each number.

Binary	Decimal
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	10
1011	11
1100	12
1101	13
1110	14
1111	15

LESSON 9 – TANENBAUM

- Cut out a simple tree using foam-core
- Decorate with LEDs
- Create a pattern with your program