# Project 1

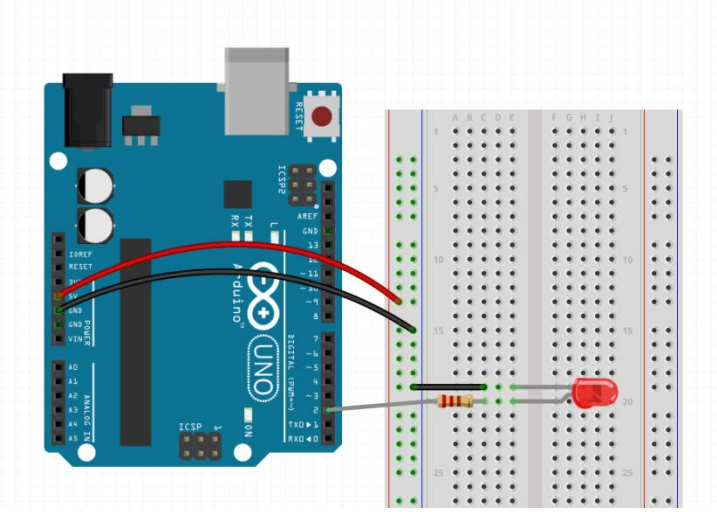
# Blink

## Objectives:

- Understand how to install and use the Arduino IDE
- Understand basic Arduino commands and code structure
- Understand concepts related to inputs and outputs using the Arduino microcontroller
- Understand simple coding logic and design

Get into groups of 3 or 4 and complete the following exercises.

# Blink

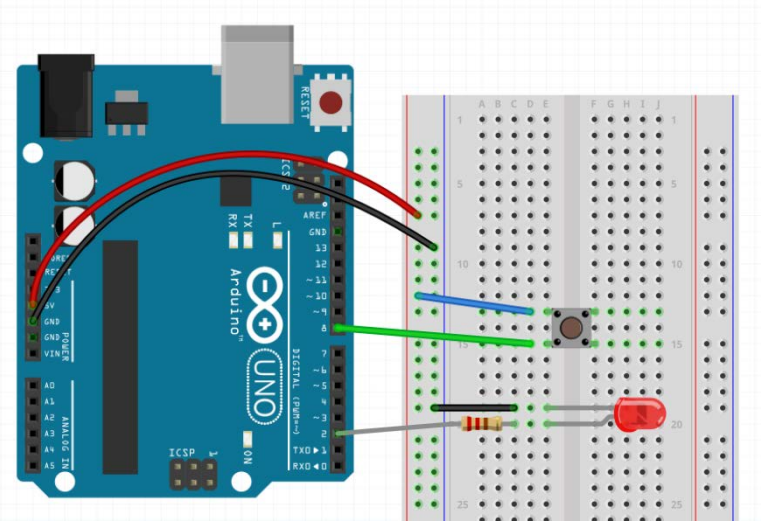


```
/*
  Blink
  Turns on an LED on for one second, then off for one second, repeatedly.

  Most Arduinos have an on-board LED you can control. On the Uno and
  Leonardo, it is attached to digital pin 13. If you're unsure what
  pin the on-board LED is connected to on your Arduino model, check
  the documentation at http://arduino.cc

  This example code is in the public domain.

  modified 8 May 2014
  by Scott Fitzgerald
 */


// the setup function runs once when you press reset or power the board
void setup()
{
  // initialize digital pin 2 as an output.
  pinMode(2, OUTPUT);
}

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

## Changing It Up

Now that you know how outputs work, with your group, make the LED blink faster. How did you make it blink faster? Explain.

# Blink Button



```
/* Basic Digital Read
* turns on and off a light emitting diode(LED) connected to digital
 * pin 13, when pressing a pushbutton attached to pin 7. It illustrates the
 * concept of Active-Low, which consists in connecting buttons using a
 * 1K to 10K pull-up resistor.
 * Created 1 December 2005
 * copyleft 2005 DojoDave <http://www.0j0.org>
 * http://arduino.berlios.de
 */

int ledPin = 13;  // choose the pin for the LED
int inPin = 7;      // choose the input pin (for a pushbutton)
int val = 0;         // variable for reading the pin status

void setup()
{
  pinMode(ledPin, OUTPUT );  // declare LED as output
  pinMode(inPin, INPUT);        // declare pushbutton as input
}

void loop()
{
  val = digitalRead(inPin);         // read input value

  if (val == HIGH)                    // check if the input is HIGH (button released)
  {
    digitalWrite(ledPin, LOW);  // turn LED OFF
  }
  else
  {
    digitalWrite(ledPin, HIGH);  // turn LED ON
  }
}
```

## Changing It Up

Now that you have an understanding of how to use an input (BUTTON) to control an output(LED), lets change things up a bit. In this exercise we want the output (LED) to turn on with one press of the button, and then turn off with another press of a button (like a light switch). How did your group accomplish this task? Explain.