Robogals Science Challenge

Minor Challenge Set #3
STEM Field: Electrical Engineering
Level: Senior
Challenge Name: Build an RGB LED Circuit
Project Cost: 0 USD
Materials Required:

- Laptop/ Computer with Internet access
- An account on TinkerCAD (a free software)
    - Registration for an account is necessary to save progress

Duration:

- The challenge takes approximately one to two hours to finish, however, the time guideline is an estimation only, and students and mentors can complete the tasks around their schedules

Introduction:

Arduino is a platform used to create interactive electronic projects. We can use Arduino boards to read an input, for example, light on a sensor, and turn it into an output, such as turning on an LED. If you want to tinker with a project that involves both software (coding) and hardware (building electrical circuits), Arduino is a great way to start.

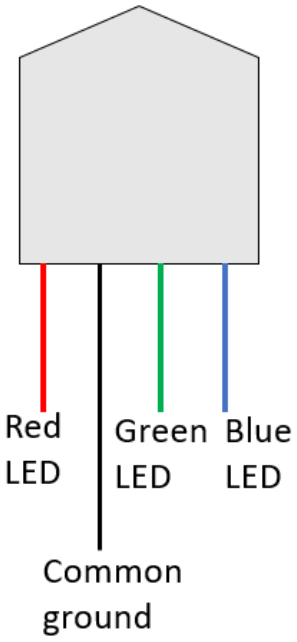
To work on an Arduino project, you will need an Arduino board, wires, batteries, resistors, LEDs and more components, depending on the project you are working on. It may be difficult to gather all the required

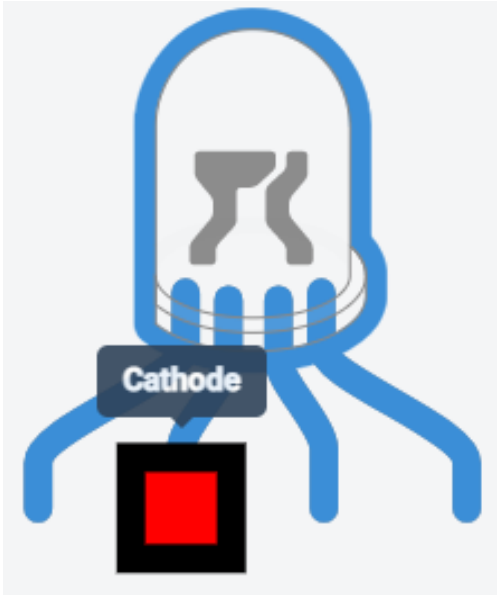
components, therefore, in this activity, we will use TinkerCAD - a free software with tools to build electronic circuits and simulate them.

In this project, we will build an electronic circuit with RGB LED using Arduino and breadboard. Upon completion of this activity, you may like to build more difficult circuits and submit them as a Major Challenge.

Why do we use RGB LEDs? An RGB LED bulb is a combination of three LEDs with three main colours: red, yellow, and green. These colours can be mixed and matched to produce a variety of different colours! An RGB LED bulb has four legs, one for each colour, and one works as the common ground for all internal LEDs. The common ground can also be called a “common cathode”.

To differentiate each leg, you can hover your mouse over the RGB LED on TinkerCAD. You will see a label for “Red”, “Green”, or “Blue” LED, and a label called “Cathode”.





Instructions:

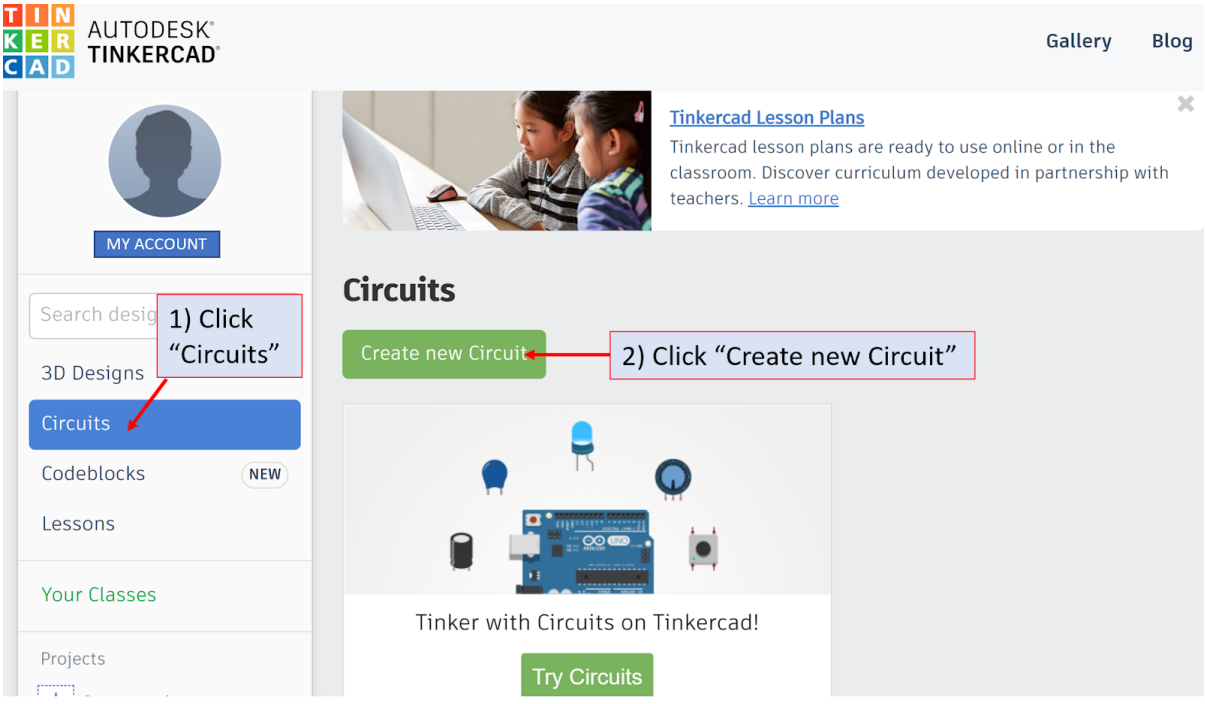
Part 1: Setting Up

1. Navigate to the website: https://www.tinkercad.com. It is recommended that you open this website on a browser such as Chrome. This software is free-to-use, and creating an account is necessary to save the progress of your project.

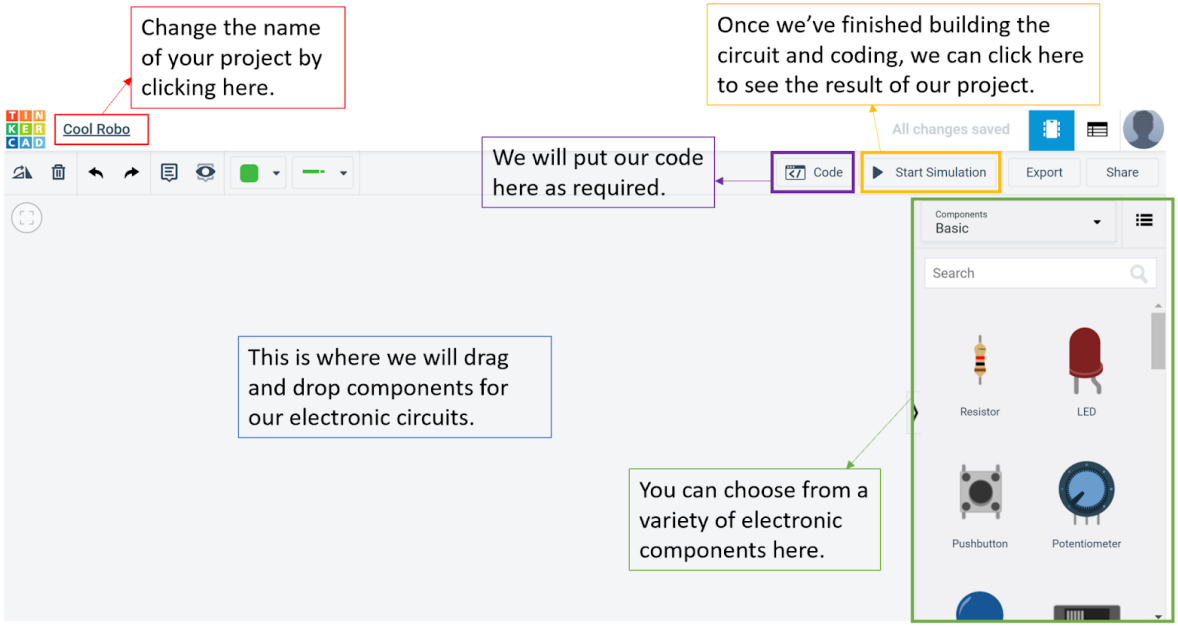
   If you have not registered, click “JOIN NOW”, then select “Create a personal account”.

   Alternatively, click “Sign In” to log into your TinkerCAD account.

2. After you have logged in, you will see a dashboard with projects you are working on. TinkerCAD can be used to create 3D models or other software projects. In this activity, we will build our electronic circuit, so select “Circuits”, then “Create new Circuit” (See the figure below).



3. This is what your project dashboard will look like. The figure below shows the main functions you should be aware of before building your project.



4. In this project, you will need 1x Arduino Uno board, 1x small breadboard, 1x RGB LED and 3x resistors. Drag and drop the components to the space provided.

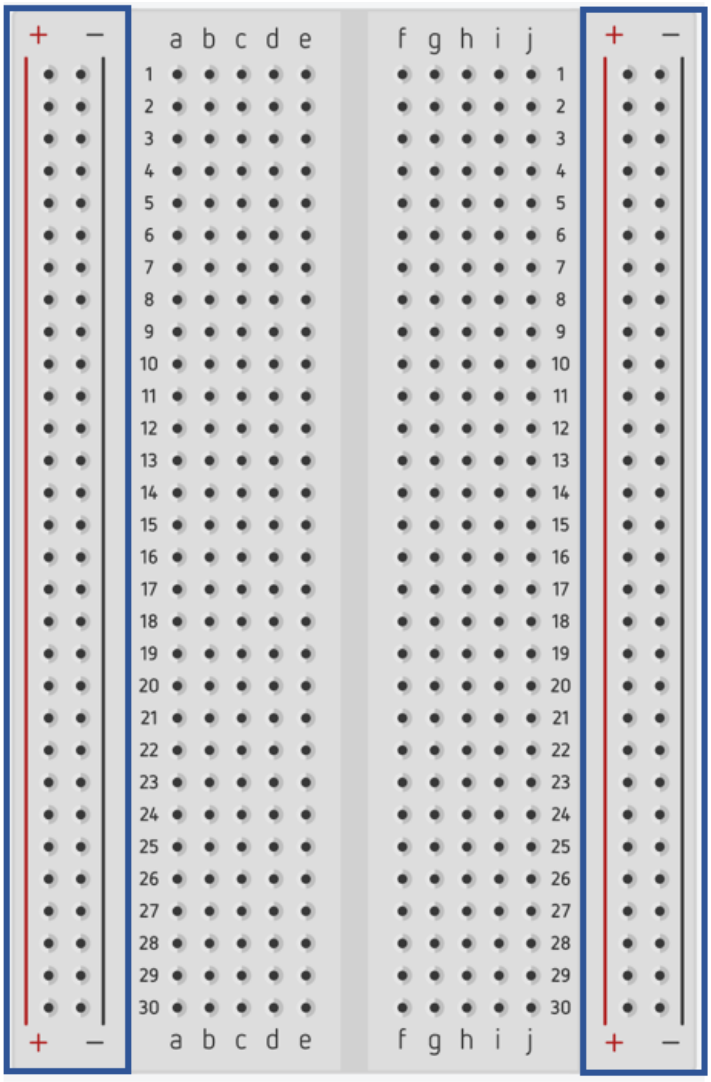
5. Click on each resistor and change the resistance to 220 Ω. The symbol "Ω" is read as "ohms", which is a unit used to measure the resistance of an electrical device. Resistance is a property that reduces the current flow in a circuit.

   Explanation: We need to use a resistor for our LED circuit. If there is too much current flowing through the LED, it may burn the LED out. A resistor is a component used to limit the current flow, and so helps prevent the LED from burning.

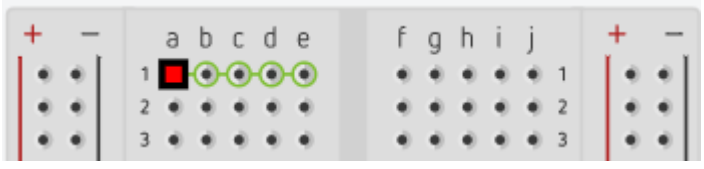
Part 2: Using the Breadboard

A breadboard is a useful tool to use when you want to build circuits with multiple electronic components.

On the left and right hand side of the breadboard are power rails (highlighted in the blue box). When you connect a power source (for example, a battery, an Arduino) to these power rails, they provide power to other electronic components as you connect them to the breadboard.

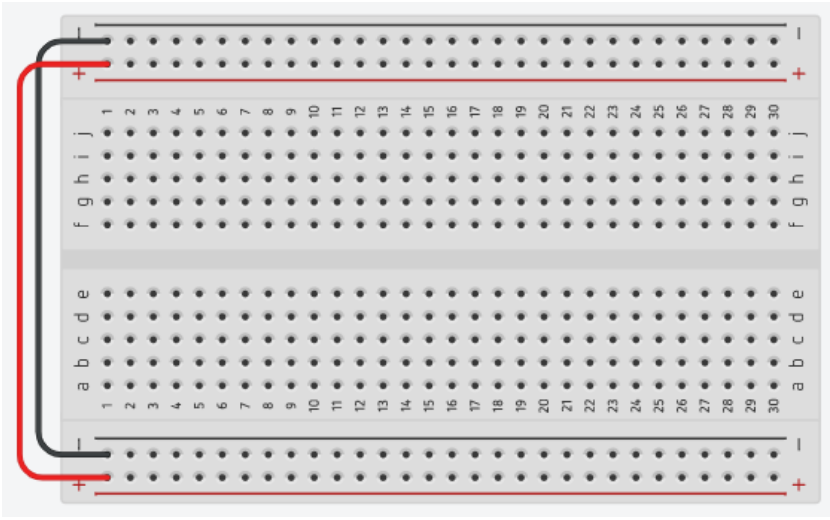


Hover your mouse over a row in the middle of the breadboard, you will see the entire row is highlighted in green. This means that they are electrically connected to one another.



As both sides of the breadboard are not electrically connected, it is best practice to connect the **+** and **-** rails on both sides together using wires.

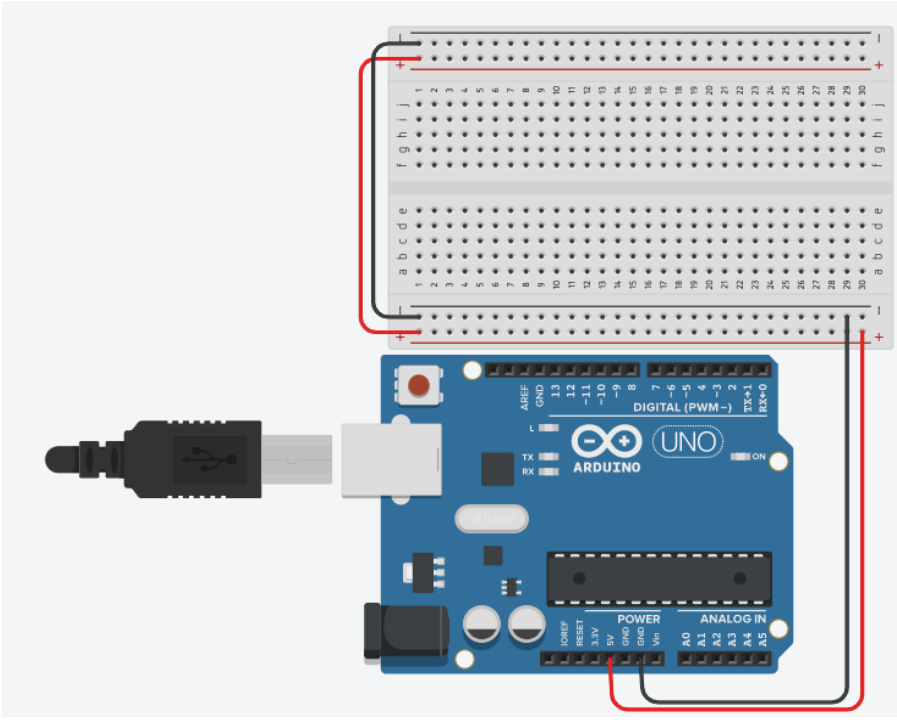
6. To add wires, click on two ends of the components you want to connect. Add wires to the breadboard as shown below. Click on each wire and change the colour to match the figure.



Part 3: Connect the Power Source

7. Add wire to connect the “5V” pin on the Arduino to one of the **+** power rails on the breadboard. Change the colour of the wire to red.

8. Similarly, add a black wire to connect the “GND” (Ground) pin on the Arduino to one of the **-** power rails on the breadboard.



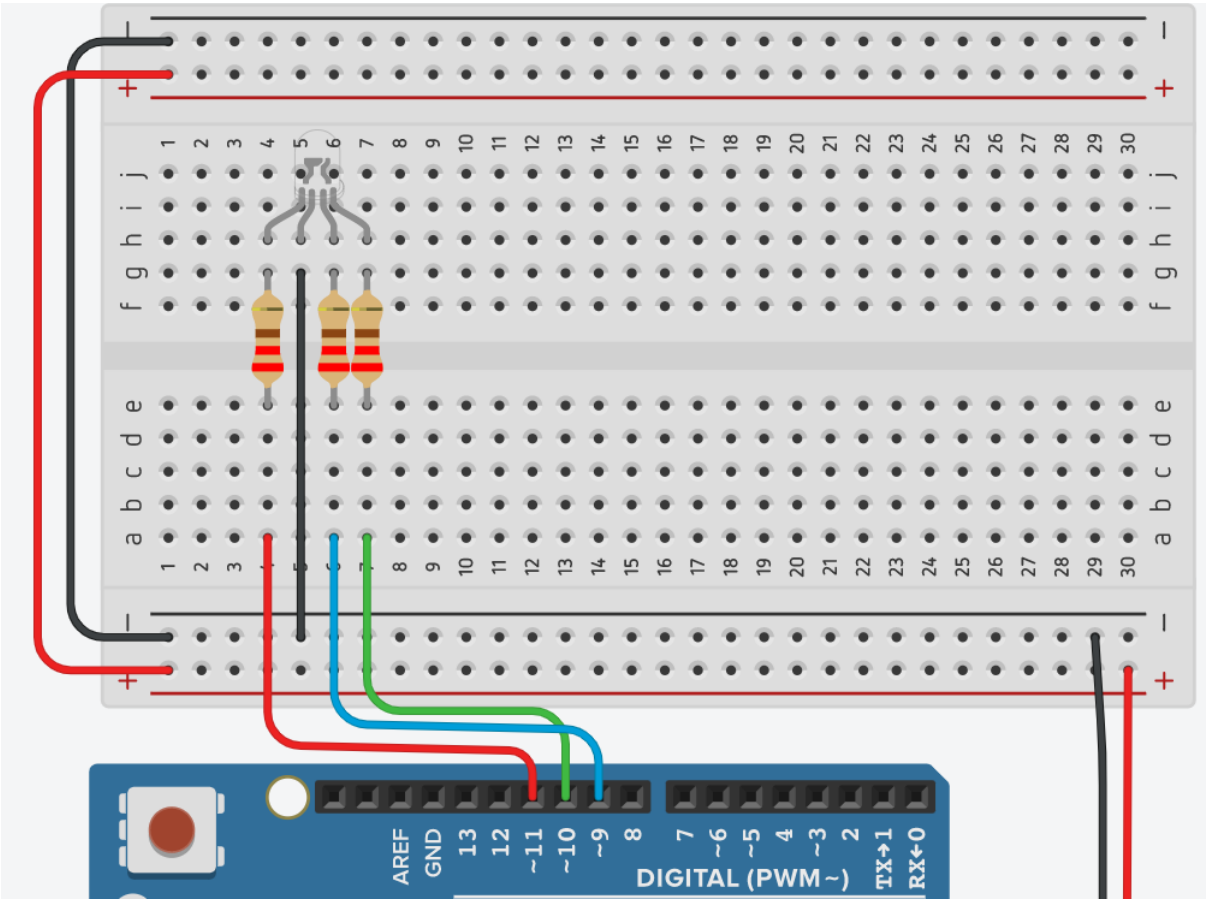
Part 4: Build the Circuit

9. Connecting the RGB LED is different to how we connect a single LED to the breadboard. With a normal LED, we would connect the positive (or longer, bent) leg to the output pin on the Arduino, and the negative (or shorter, straight) leg to the GND (ground) pin.

   Here, we connect each of the “Red”, “Blue”, and “Green” legs of the RGB LED to a resistor, then connect the same resistor and leg to the output pin, as shown in the figure below.

   Note: We will only connect the RGB LED to the output pins numbered **11, 10, 9, 6, 5, and 3** on the Arduino. Read the Extension section for more explanation.

   We will also connect the “Cathode” leg of the RGB LED to the **-** power rail.



Part 5: Adding Code

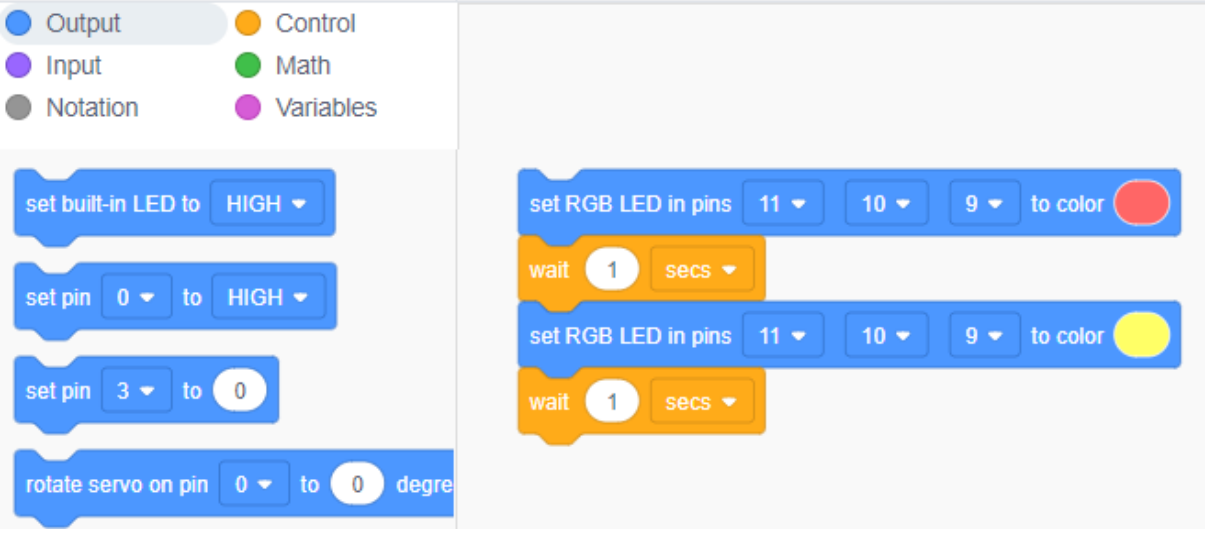
10. We will now add code to our electrical circuit by clicking on the 

symbol.

11. We want to change the RGB LED to two different colours, with each blinking for one second. In this case, our RGB LED is connected to pins numbered 11, 10, and 9.

    To do this, you can replicate the code below. You can also modify the blinking colours to colours of your choice, and adjust the blinking time.

    The blue code blocks can be found in the “Output” section, and the yellow code blocks in the “Control” section.



12. You can now click on “Start Simulation” and see how your circuit behaves.

Extension

Explanation - Why can we only connect the RGB LED to the output pins numbered 11, 10, 9, 6, 5, and 3 on the Arduino?

You may have learned at school that red, blue, and green are primary colours that are added together in various ways to produce other colours. An RGB LED is made up of three LEDs with these three primary colours. Hence, these colours need to be mixed and matched in different ways to produce other colours for our light display.

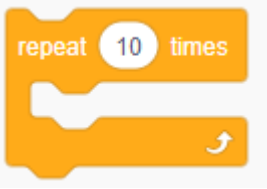
To mix and match these colours inside the RGB LED, we need to change the intensity of each of the “Red”, “Green”, and “Blue” LEDs. The intensity can be controlled by inputting less, or more power to these LEDs.

The output pins with number 11, 10, 9, 6, 5, and 3 are called Pulse-Width Modulation (PWM) pins on the Arduino. PWM is a technique of controlling the power. And so, by connecting the RGB LED to these pins, we can control the intensity of the LED, to output different colours!

Activity - Build your own LED circuit!

From what you have learned above, try to build your own circuit using a combination of single LEDs, and RGB LEDs! Modify the blinking colours, blinking patterns, blinking time to have a colourful light display. Be creative!

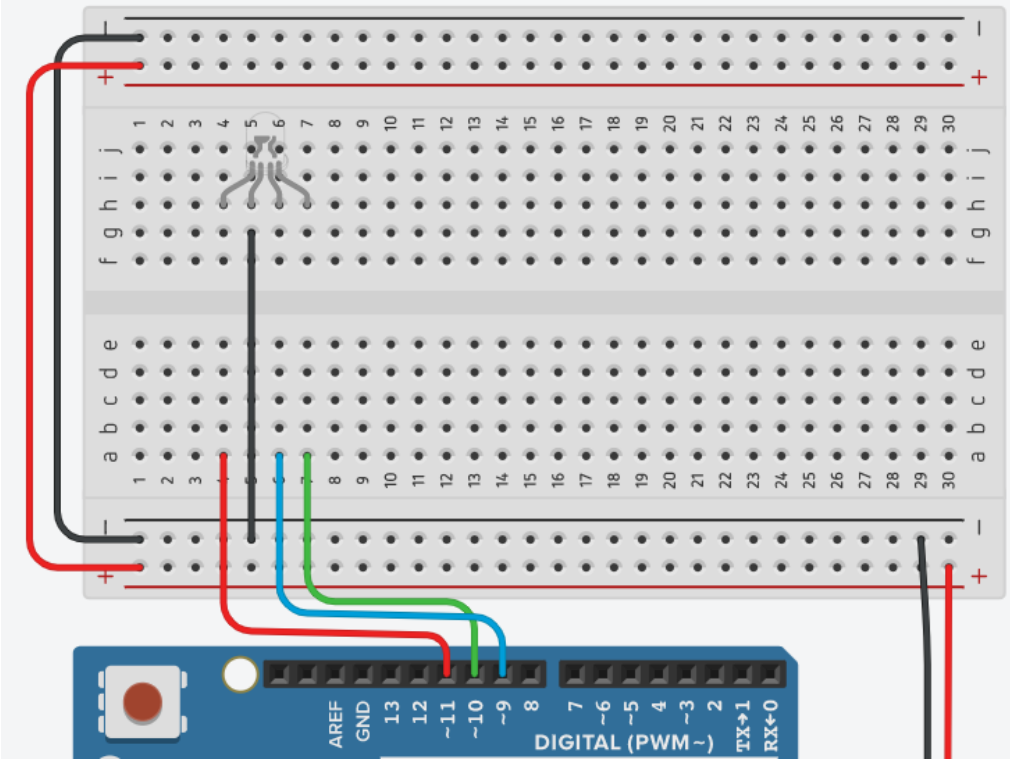
> **Tip:**
>
> Use the “Repeat” code block if you want to repeat a blinking pattern, instead of repeating the codes. Change the number of times you want to repeat the pattern.



Reflection Questions:

- Are there any improvements you would make to this challenge?
- What are the key science and engineering concepts that relate to this challenge?

- Can you think of real life examples where RGB LEDs may be used?
- What other circuits would you like to build and why?
- In the figure below, if you remove all three resistors from the circuit, then click “Start Simulation”, you may find that the LED does not blink at all. Can you explain why this is the case?



- If you remove any resistor, then click “Start Simulation”, you may notice some very interesting patterns. For example, if you remove the resistor that is connected to the “Red” LED, you probably cannot see an output colour of red, or any colours that have the primary colour as red. Can you explain why this is the case?

Submission Guidelines:

- Submit a photo of the code and your circuit, as well as any code or circuit you built for the Extension task. Include a short summary that addresses the reflection questions.

> Note: Remember, if you want to upload pictures of your Minor Challenge that also include you, please check if it is OK with your parent or guardian first.

- The submission form is on the Minor Challenges page: https://sciencechallenge.org.au/index.php/minor-challenges/ Fill out the details and make sure you upload your submission.

Learn More! Resources:

- This website has lots of projects and lessons for beginners - https://www.tinkercad.com/learn/circuits/learning
- If you want to see what other people are building or tinkering with, have a look at the Gallery section of TinkerCAD - https://www.tinkercad.com/things