

# Tópicos B

---

**PROF. DR. GUILHERME A. MADALOZZO**

# Introdução

---

O que vamos ver na disciplina?

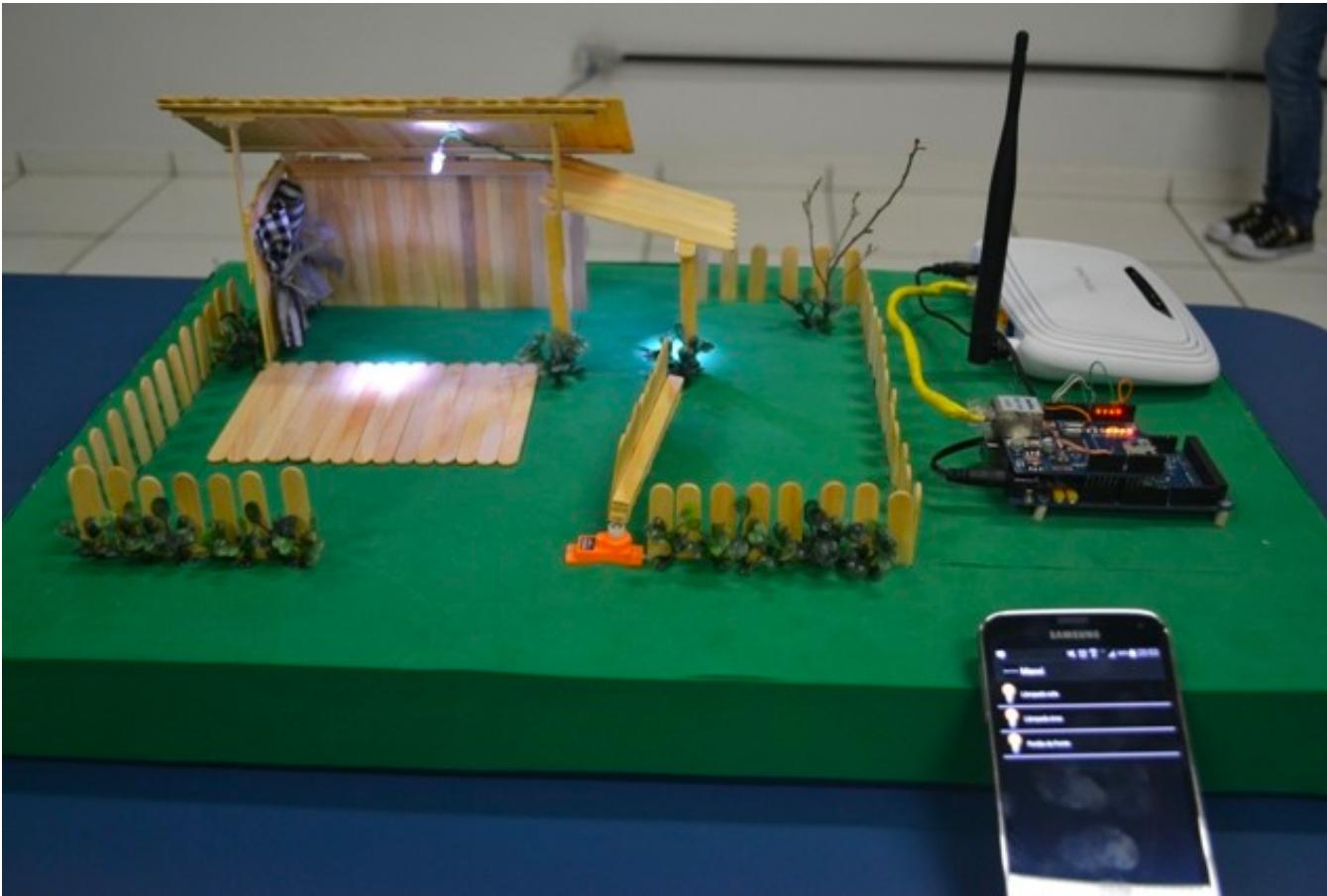
- Se tudo der certo...
  
- Programar comunicação de um aplicativo mobile com um microcontrolador



# Conteúdo programático

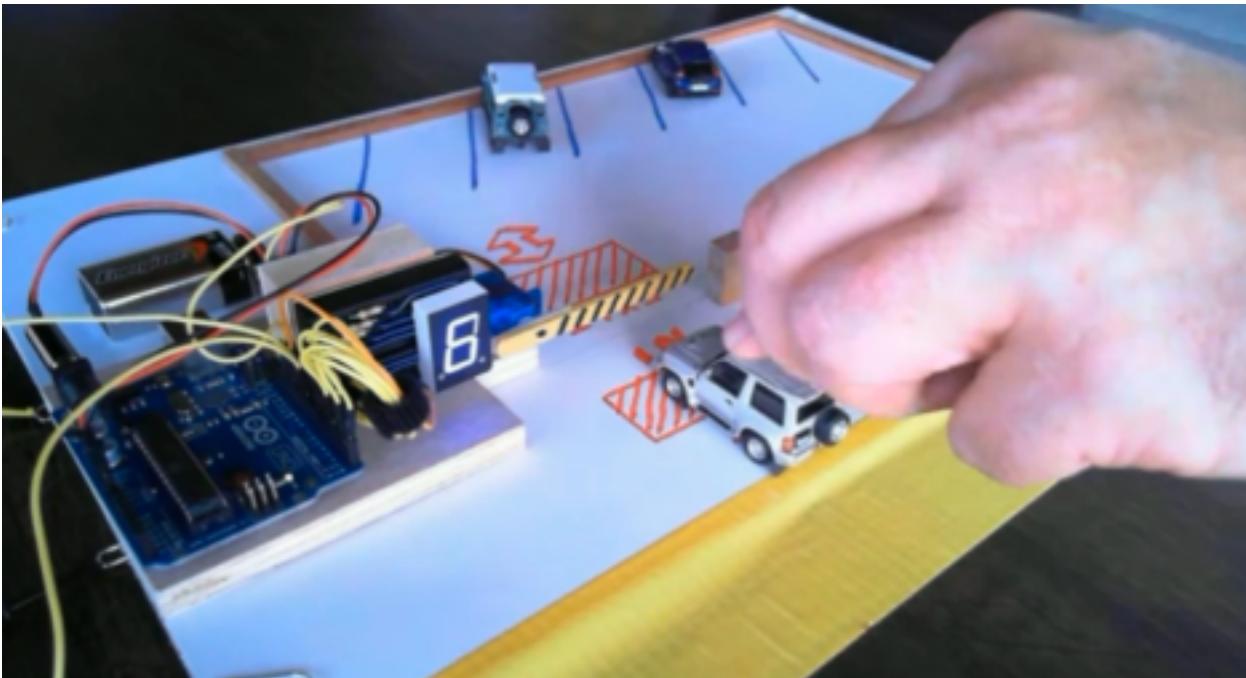
Datas	Conteúdo
26/02/2019	Apresentação do Plano de Ensino. Definição das 3 grandezas eletrônicas
12/03/2019	História componentes eletrônicos. Evolução de microcontroladores. Introdução ao Desenvolvimento Móvel com Robótica
19/03/2019	Apresentação do TinkerCad. Exemplos de modelagem de circuitos. Task point (exercícios para modelagem de circuitos).
26/03/2019	Arduino: componentes e elementos. Task point (Pesquisa de projeto para primeira atividade da disciplina)
02/04/2019	Codificação com arduino
09/04/2019	Modelagem e produção do primeiro trabalho.
16/04/2019	Apresentação de trabalhos (duplas).
23/04/2019	Introdução a comunicação embarcada.
30/04/2019	Tecnologias móveis e conceitos de comunicação embarcadas. Task point (Pesquisa sobre o tema)
07/05/2019	Introdução React-Native
14/05/2019	Desenvolvimento Front-End do App. Task point (melhoramento)
21/05/2019	Desenvolvimento Back-End do App. Task point (nova funcionalidade)
28/05/2019	Comunicando React-Native com Arduino
04/06/2019	Tempo para o desenvolvimento do trabalho da disciplina.
11/06/2019	Tempo para o desenvolvimento do trabalho da disciplina.
18/06/2019	Tempo para o desenvolvimento do trabalho da disciplina.
25/06/2019	Apresentação de trabalho prático (duplas).
02/07/2019	Avaliação de Recuperação.

# O que desejo para o final



# O que desejo para o final

---



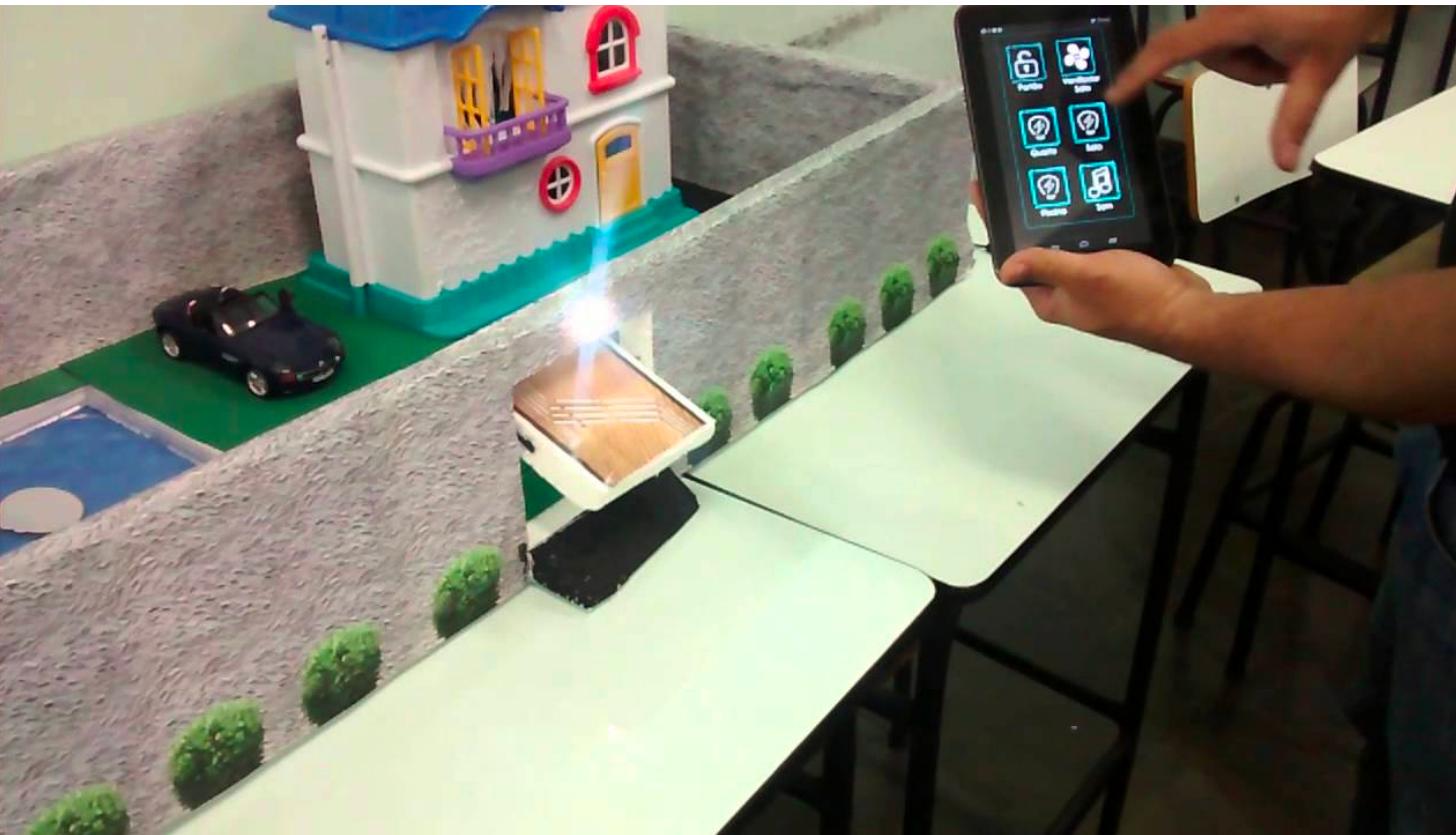
# O que desejo para o final

---



# O que desejo para o final

---



# Conceitos

---

Antes de introduzirmos novos conceitos sobre Arduino, precisamos aprender/relembrar conceitos básicos da eletrônica

- Tensão
- Corrente
- Resistência

# Conceitos

---

## Tensão

- É a quantidade de energia transportada por uma corrente, em VOLTS (V)
  
- Pense na tensão como a força e o susto de um tapa quando você se encosta em um fio da rede elétrica

# Conceitos

---

## Corrente

- É a quantidade de elétrons circulando por um condutor, em AMPERES (A)
- Pensa na corrente como a dor ou a queimadura gerada por um choque

# Conceitos

---

## Resistência

- É a intensidade que um material condutor se impõe ao fluxo dos elétrons, em OHMS ( $\Omega$ )
- Pensa na resistência sendo o quanto você suporte levar o choque

# Conceitos

---

## Relação das 3 grandezas da eletrônica

- Para uma mesma resistência, quanto maior for a tensão, maior será a corrente

# Conceitos

---

## Relação das 3 grandezas da eletrônica

- Para uma mesma tensão, quanto maior for a resistência, menor será a corrente

# Conceitos

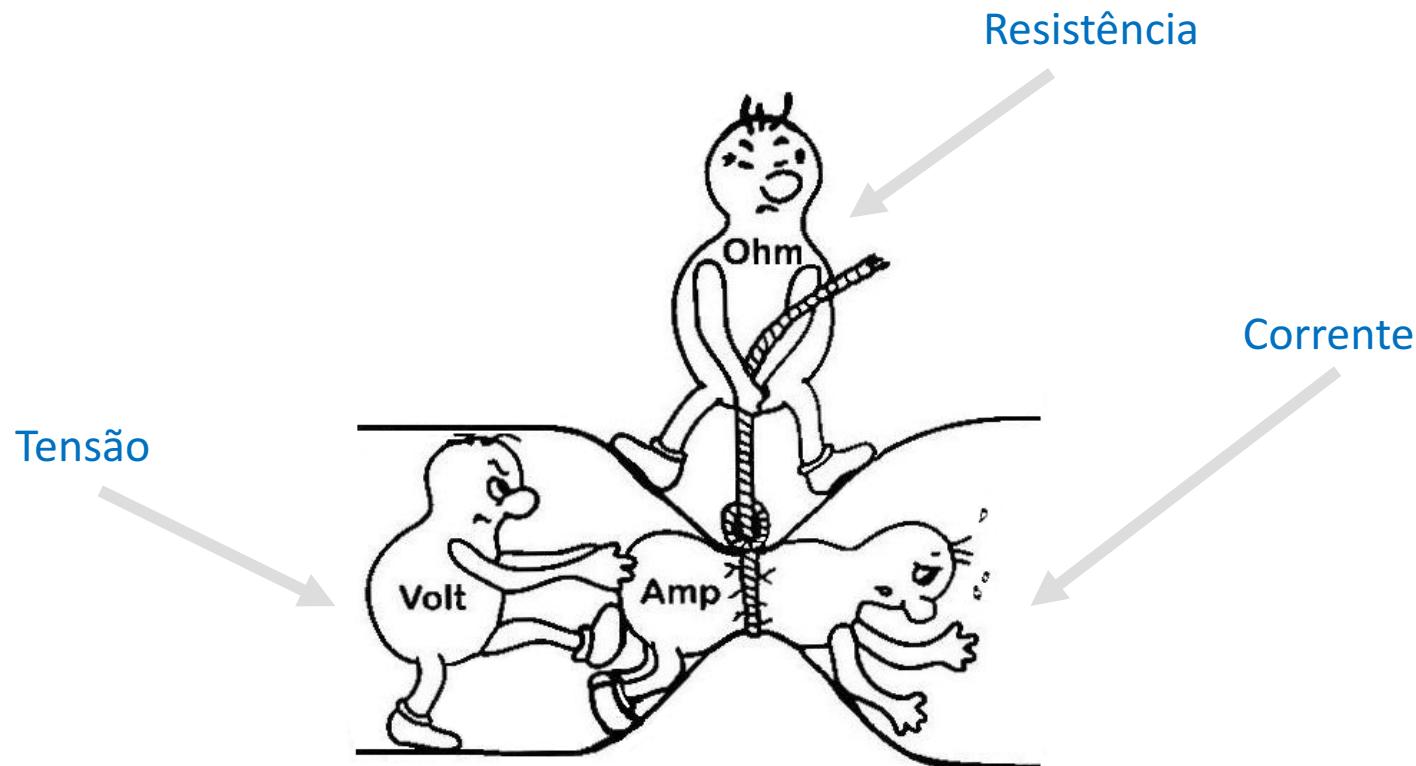
---

## Relação das 3 grandezas da eletrônica

- Para uma mesma corrente, quanto maior for a resistência, maior será a tensão

# Conceitos

Relação das 3 grandezas da eletrônica



# Círculo Elétrico

A partir das 3 grandezas, podemos analisar um círculo elétrico

- É composto por diversos tipos de componentes
  - Fonte de energia e demais componentes
- Todos os componentes transformam a energia elétrica em outro tipo de energia
  - Térmica
  - Eletromagnética
  - Mecânica
  - Etc...

# Círculo Elétrico

A partir das 3 grandezas, podemos analisar um círculo elétrico

- É composto por diversos tipos de componentes
  - Fonte de energia e demais componentes
- Todos os componentes transformam a energia elétrica em outro tipo de energia
  - Térmica (resistências de chuveiro)
  - Eletromagnética (lâmpadas)
  - Mecânica (motores)
  - Etc...

# Círculo Elétrico

## Analisando um círculo elétrico

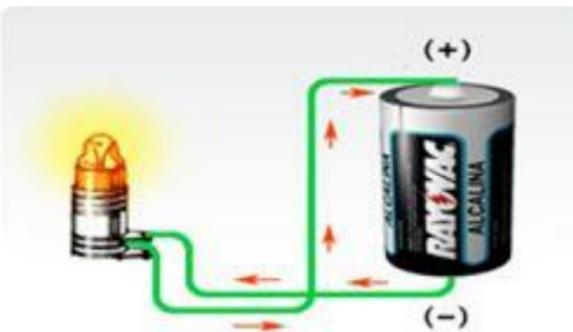
---

- Temos dois tipos de representação de círculos
  - Círculo real
  - Círculo esquemático

# Círculo Elétrico

Analisando um círculo elétrico

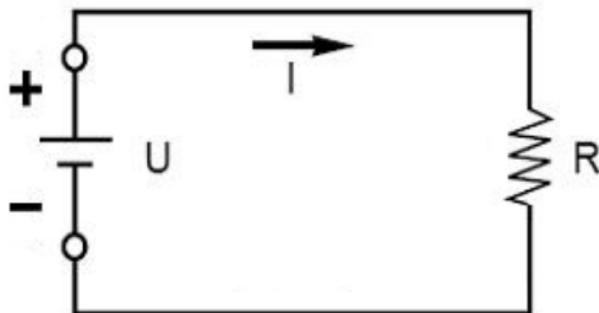
- Temos dois tipos de representação de circuitos
  - Círculo real



# Círculo Elétrico

## Analisando um círculo elétrico

- Temos dois tipos de representação de circuitos
  - Círculo esquemático



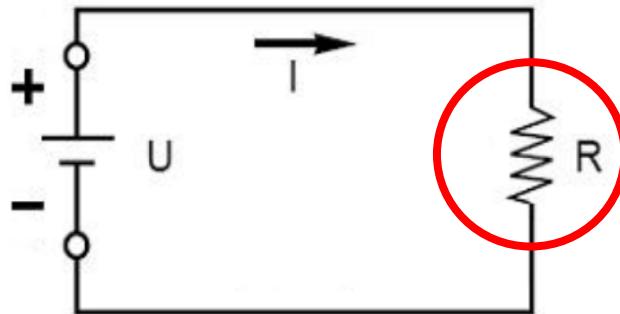
Onde:

U ou V → Tensão  
I → Corrente  
R → Resistência

# Círculo Elétrico

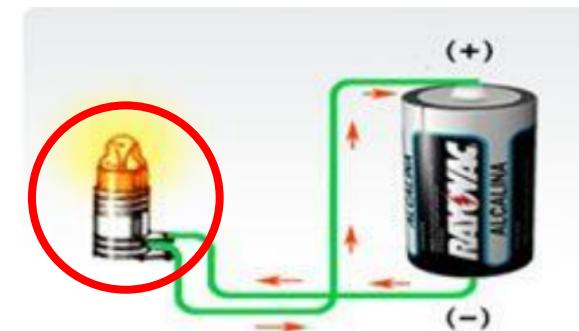
## Analisando um círculo elétrico

- O LED (lâmpada) será representado como uma resistência **R**



Onde:

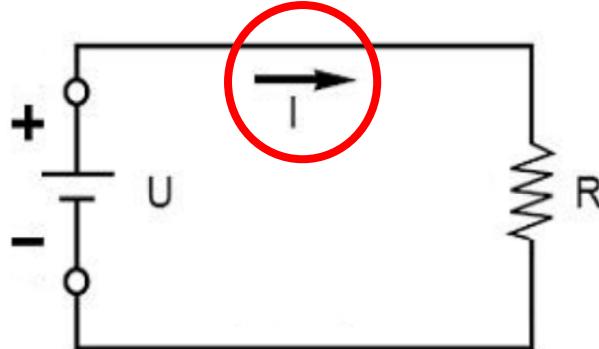
U ou V → Tensão  
I → Corrente  
R → Resistência



# Círculo Elétrico

## Analisando um círculo elétrico

- O LED (lâmpada) será representado como uma resistência **R**
- Convencionalmente, o sentido da corrente é do positivo (+) para o negativo (-)



Onde:

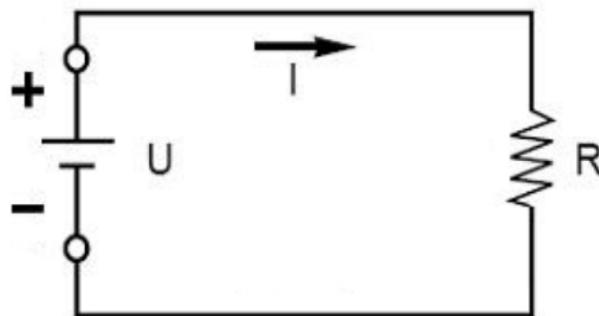
U ou V → Tensão  
I → Corrente  
R → Resistência



# Círculo Elétrico

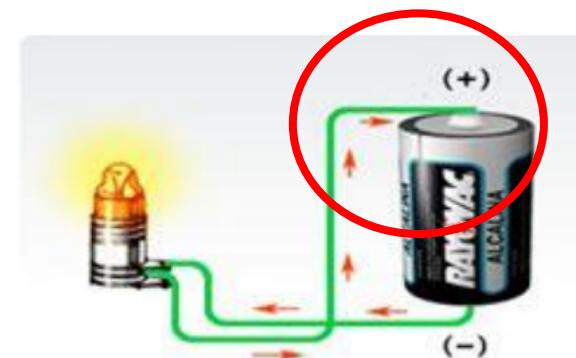
## Analisando um círculo elétrico

- O LED (lâmpada) será representado como uma resistência **R**
- Convencionalmente, o sentido da corrente é do positivo (+) para o negativo (-)
- Entretanto, na física real do circuito, os elétrons se propagam do negativo (-) para o positivo (+)



Onde:

U ou V → Tensão  
I → Corrente  
R → Resistência



# Círculo Elétrico

## Resumindo...

- Tensão é definida pela fonte de alimentação
  - Rede elétrica 127v ou 220v
  - Fonte de computador com 3.3v, 5v ou 12v
- Cada equipamento possui sua faixa de operação
  - Equipamentos 220v ou 127v
  - Arduino 5v
  - ARM 3.3v
  - Processadores 1.8v
- Deve-se utilizar a fonte adequada para cada equipamento

# Círculo Elétrico

---

## Cuidados...

- Sempre verificar se a fonte utilizada é adequada para o componente
- A tensão não é responsável por danificar ou queimar um componente
- O que queima o componente é a corrente
  - O mesmo que nos da choque
- Uma tensão acima do suportado irá induzir uma corrente excessiva, assim queimando o componente

# Duplas

---

## Pesquisar projetos Arduino

- Projeto IoT utilizando Arduino e montando maquete
  - Procure por projetos no Google, também pode-se procurar no GitHub
  - Também, pode-se montar o projeto conforme gosto, misturando tantos outros
  - Entregar uma descrição de o que será feito e o que será controlado
    - Gostaria que a maquete apresente no mínimo 3 controladores

# Duplas

## Pesquisar projetos Arduino

- Projeto IoT utilizando Arduino e montando maquete
  - Procure por projetos no Google, também pode-se procurar no GitHub
  - Também, pode-se montar o projeto conforme gosto, misturando tantos outros
  - Entregar uma descrição de o que será feito e o que será controlado
    - Gostaria que a maquete apresente no mínimo 3 controladores

Inicialmente os 3 (ou mais) controles serão desenvolvidos e reproduzidos via botões conectados diretamente ao Arduino

# Duplas

## Pesquisar projetos Arduino

- Projeto IoT utilizando Arduino e montando maquete
  - Procure por projetos no Google, também pode-se procurar no GitHub
  - Também, pode-se montar o projeto conforme gosto, misturando tantos outros
  - Entregar uma descrição de o que será feito e o que será controlado
    - Gostaria que a maquete apresente no mínimo 3 controladores

Inicialmente os 3 (ou mais) controles serão desenvolvidos e reproduzidos via botões conectados diretamente ao Arduino

Objetivo final é esses controles serem controlados via App de celular

# Arduino

# Arduino

---

É uma plataforma de aprendizado altamente versátil

No âmbito da eletrônica, já inclui o básico para o uso do microcontrolador

No âmbito da computação, possui uma interface simplificada para operação do componente, deixando para o usuário apenas a necessidade de implementar o necessário

# Arduino

Para projetar circuitos com Arduino, vamos utilizar a ferramenta tinkerCAD



Tinkercad | Create 3D digital +

VPN | www.tinkercad.com

UPF Programming Online TV Movies Neuroinformatics PodCasts Geral Startup Adroid Clinica Laravel AWS Ionic SO GIT Bibliografias

**TINKERCAD FOR...** RECURSOS GALERIA COMMUNITY APRENDA ENSINAR ENTRAR INSCREVER-SE

O Tinkercad é um aplicativo simples e on-line de projeto e impressão 3D para todos os usuários.

O Tinkercad é usado por projetistas, entusiastas, professores e crianças para criar brinquedos, protótipos, decoração da casa, modelos do Minecraft, joias... a lista é realmente infinita.

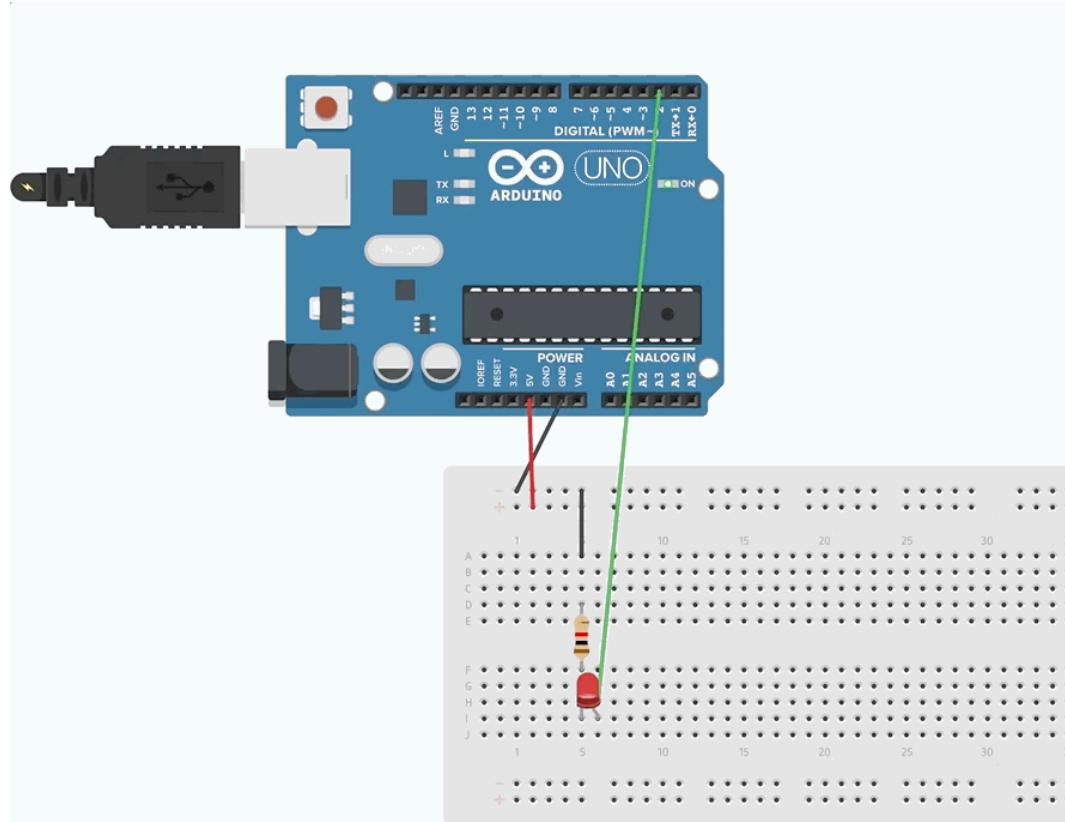
Comece a usar o Tinkercad agora



# Projeto Pisca-LED

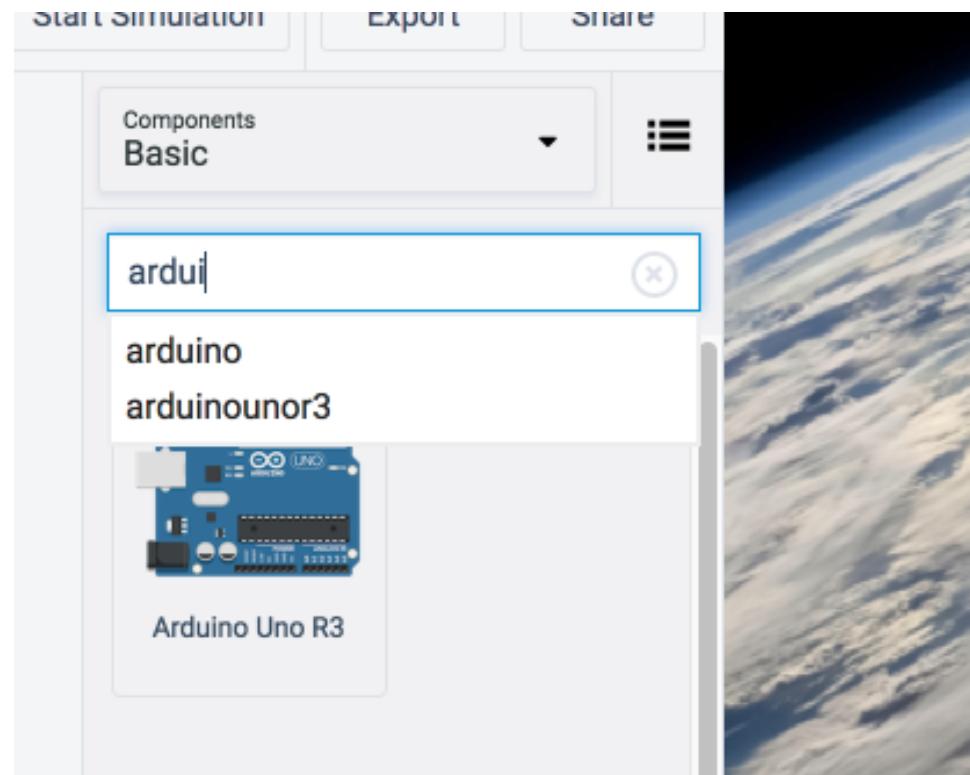
# Arduino

Vamos projetar nosso primeiro projeto com Arduino... **pisca-LED**



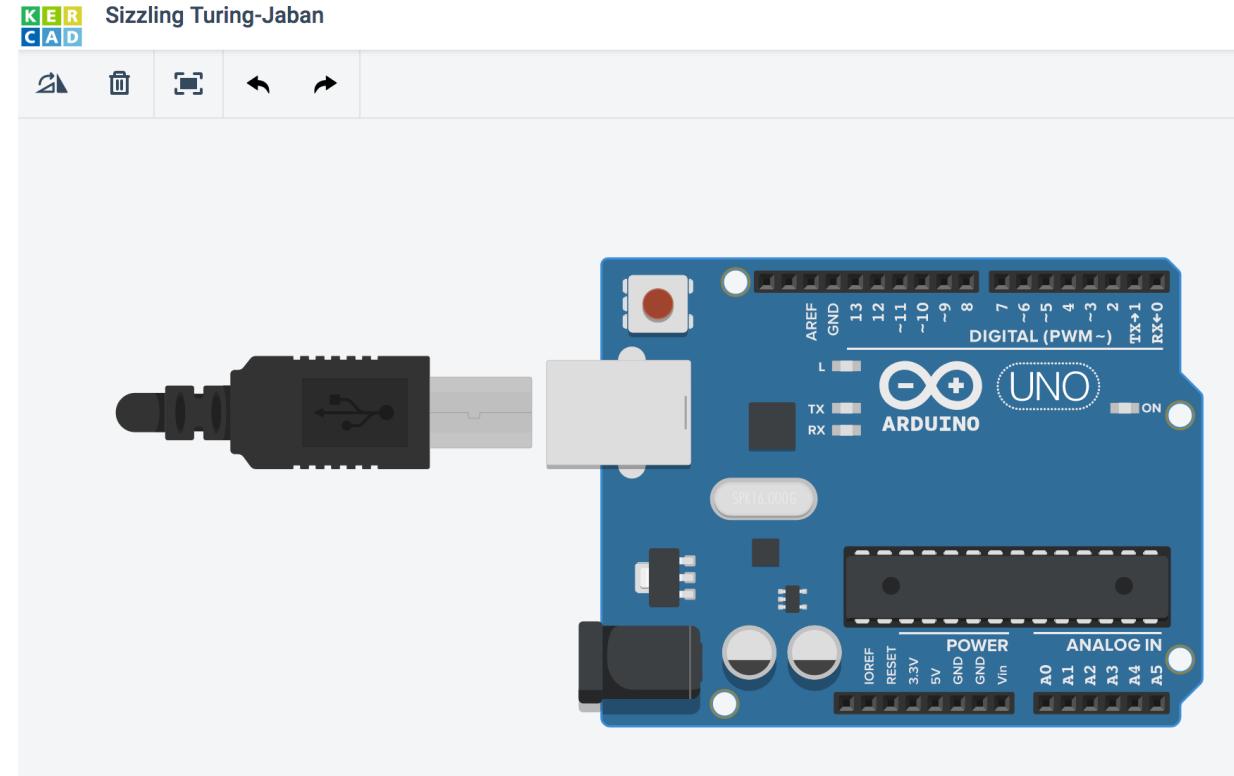
# Arduino (**PISCA-LED**)

Primeiro, vamos adicionar a placa do Arduino a partir da barra de componentes



# Arduino (PISCA-LED)

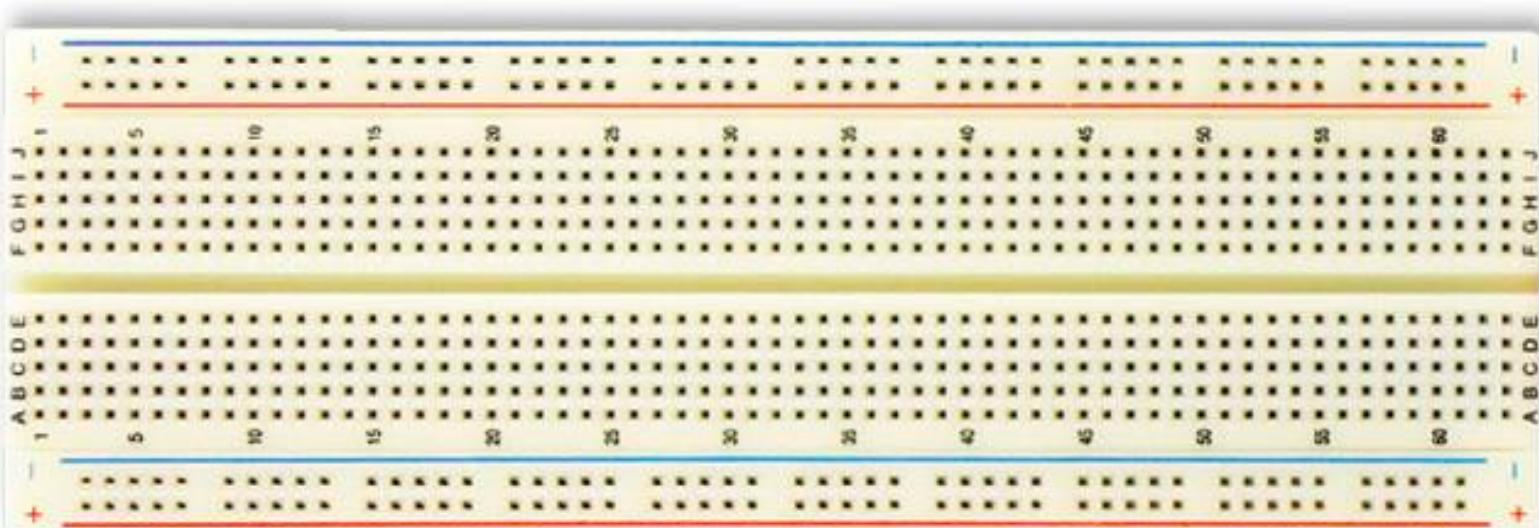
Primeiro, vamos adicionar a placa do Arduino a partir da barra de componentes



# Arduino (**PISCA-LED**)

Agora, vamos adicionar uma protoboard

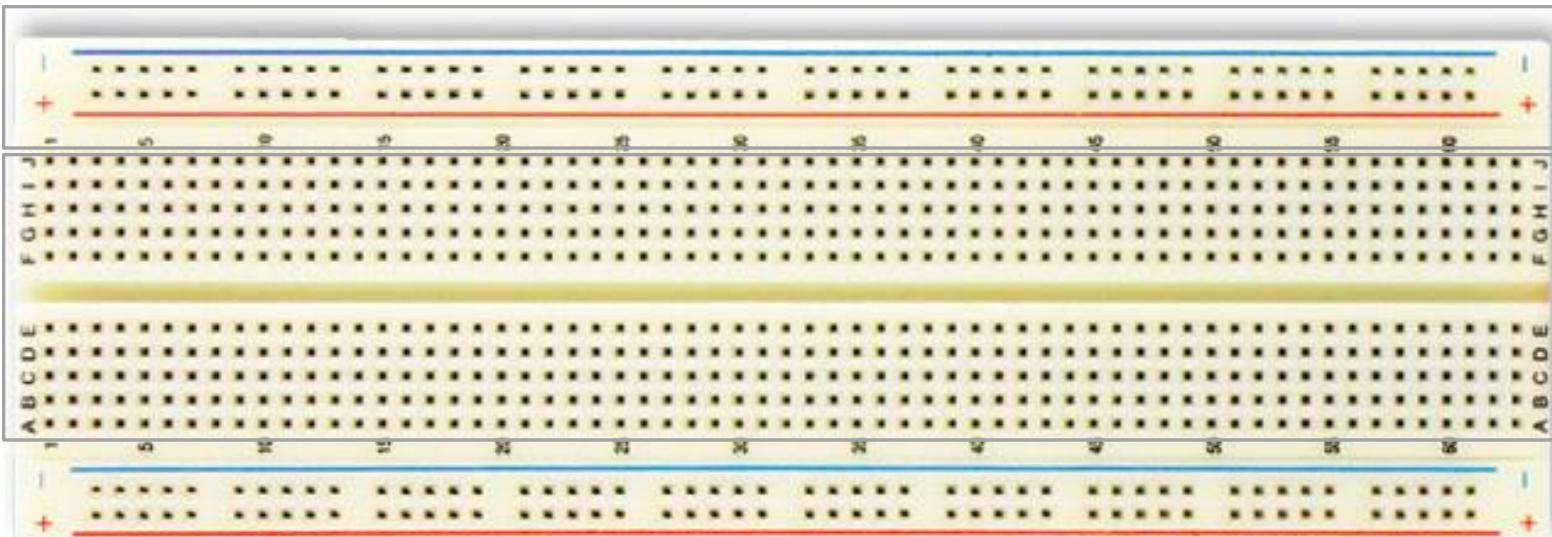
É utilizada para fazer montagens provisórias e teste de projetos



# Arduino (**PISCA-LED**)

Agora, vamos adicionar uma protoboard

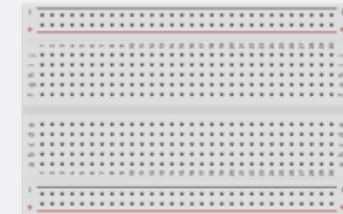
Orifícios dispostos em colunas e linhas. As linhas encontram-se nas extremidades da protoboard e as colunas ao centro



# Arduino (PISCA-LED)

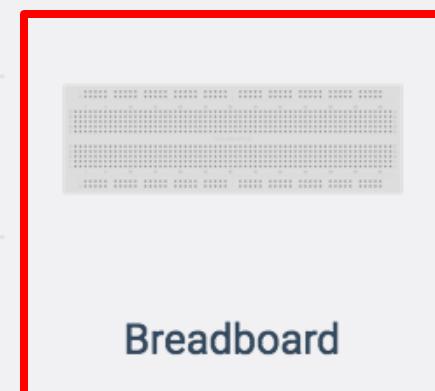
Deve-se adicionar a Breadboard

board

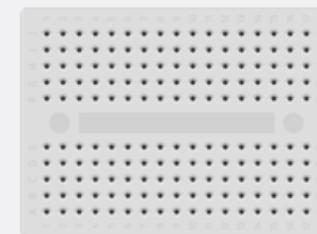


Breadboard Small

## Other Components



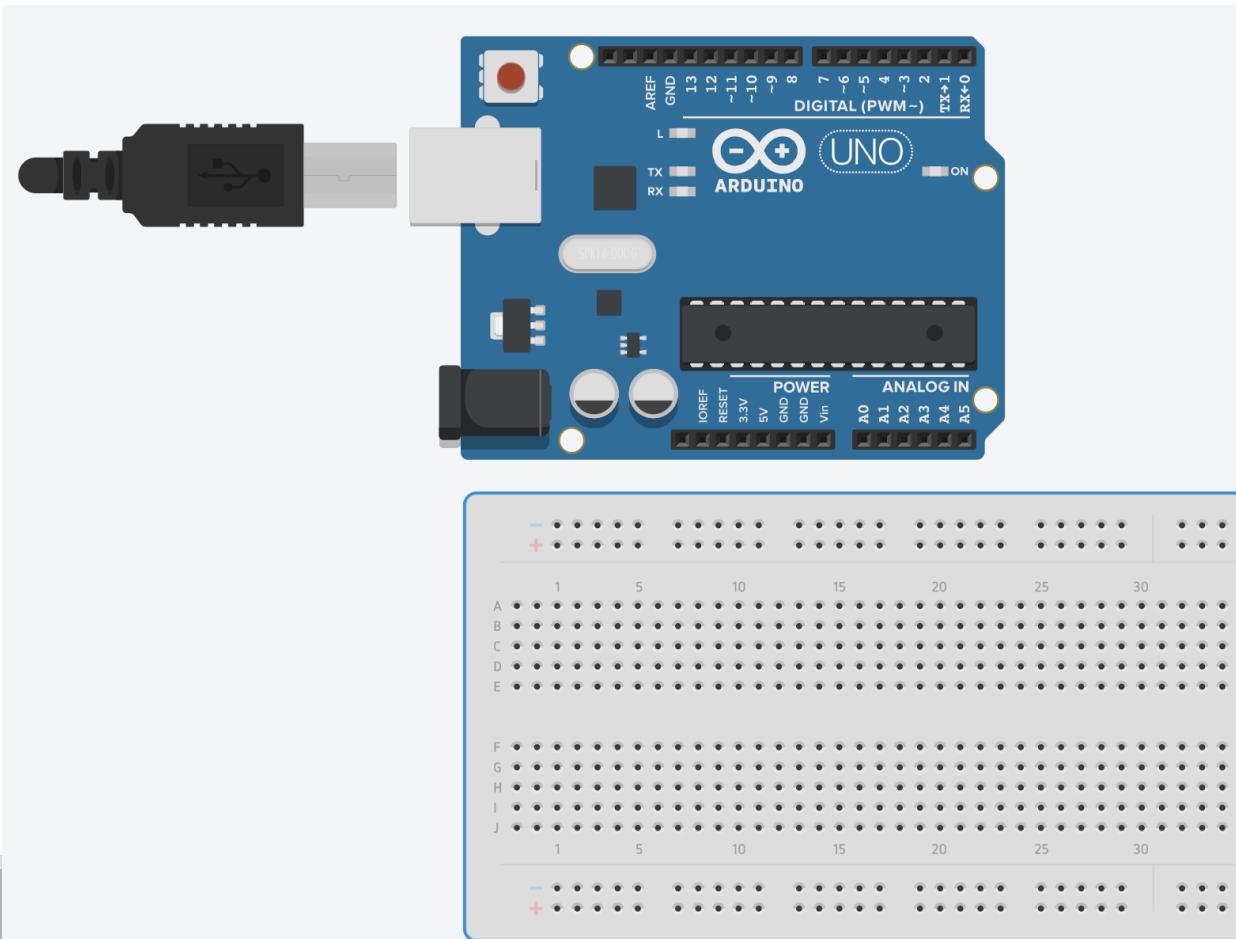
Breadboard



Breadboard Mini

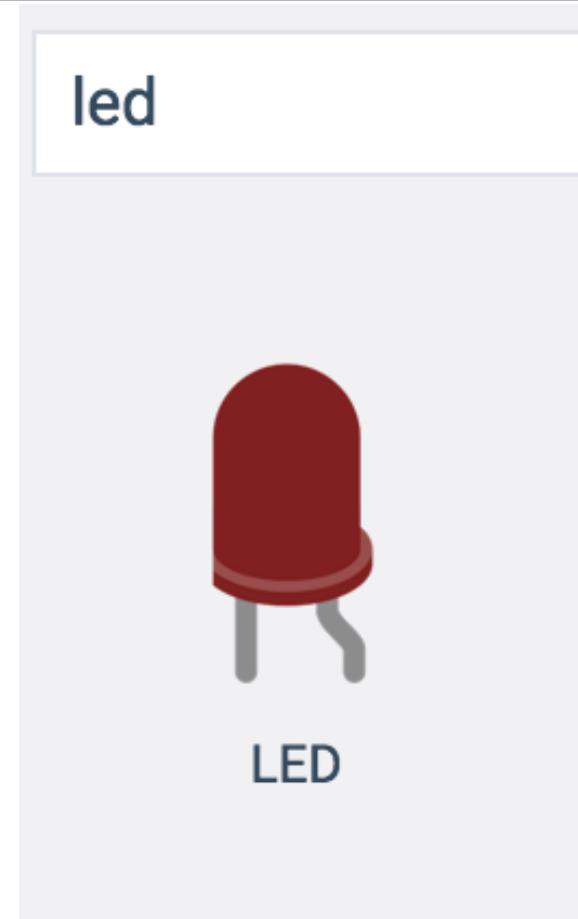
# Arduino (PISCA-LED)

Deve-se adicionar a Breadboard



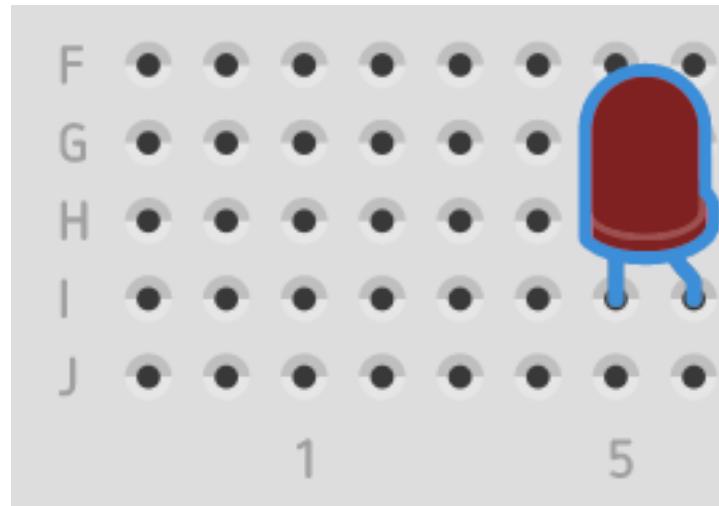
# Arduino (**PISCA-LED**)

Vamos adicionar um LED



# Arduino (**PISCA-LED**)

Vamos adicionar um LED no pino i5 e i6 da protoboard



# Arduino (**PISCA-LED**)

Vamos adicionar um resistor

Pois a voltagem do arduino (5) é maior que a do LED (2)

resist

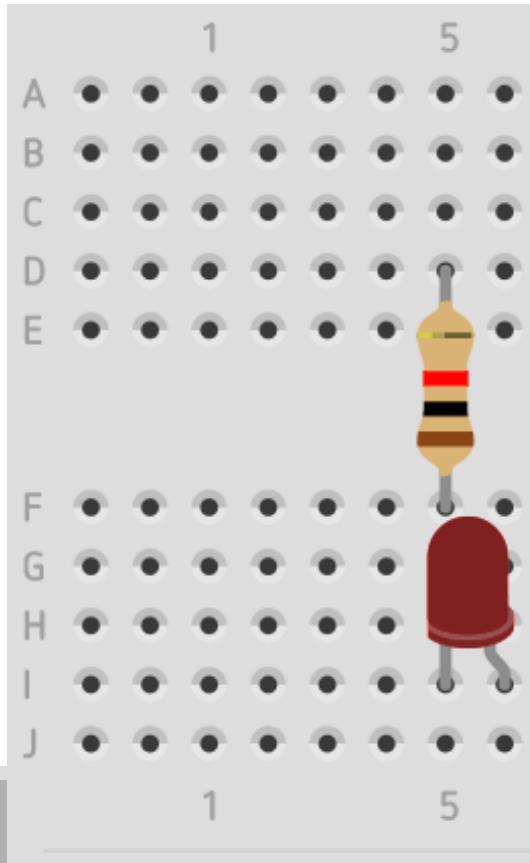


Resistor

# Arduino (**PISCA-LED**)

Vamos adicionar um resistor nos pinos f5 e d5

A faixa marrom deve ficar próxima ao f5, a vermelho próxima ao d5



# Arduino (**PISCA-LED**)

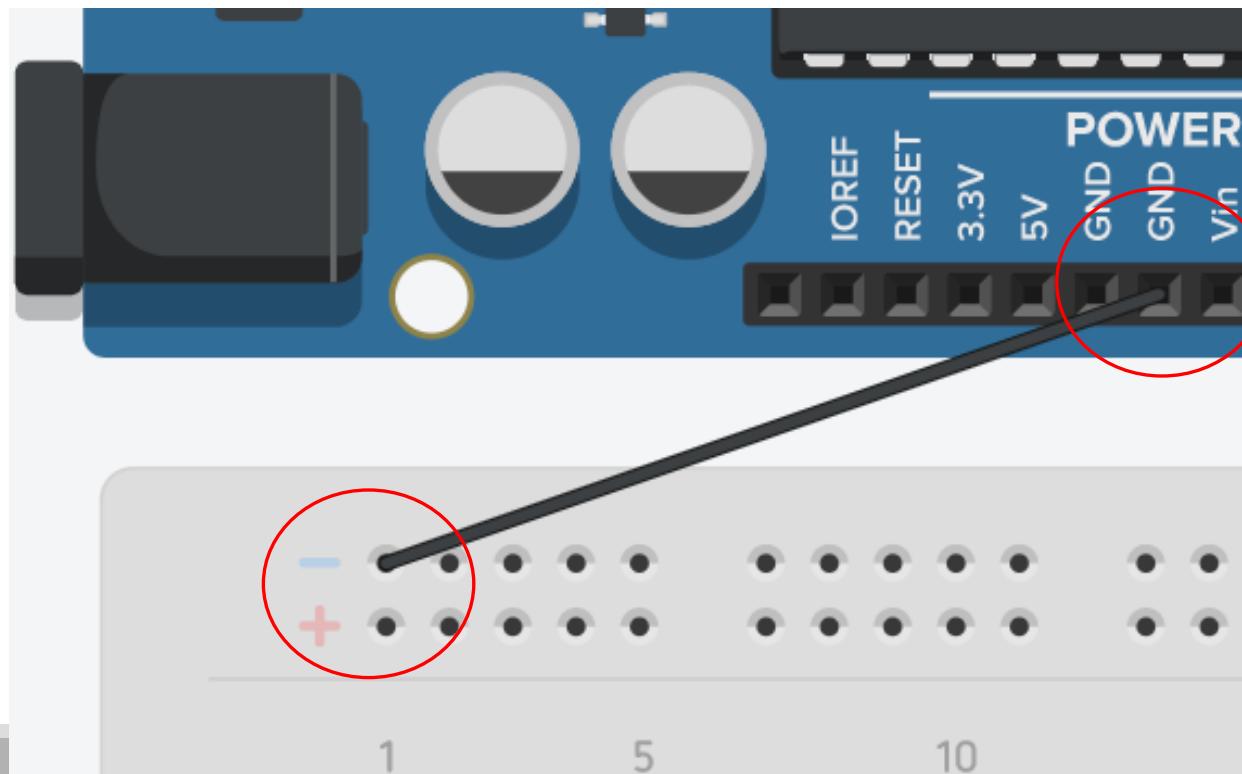
---

Agora temos que fazer a ligação dos componentes

# Arduino (**PISCA-LED**)

Agora temos que fazer a ligação dos componentes

Vamos ligar a porta power GND (terra – filtro graduado de densidade neutra)

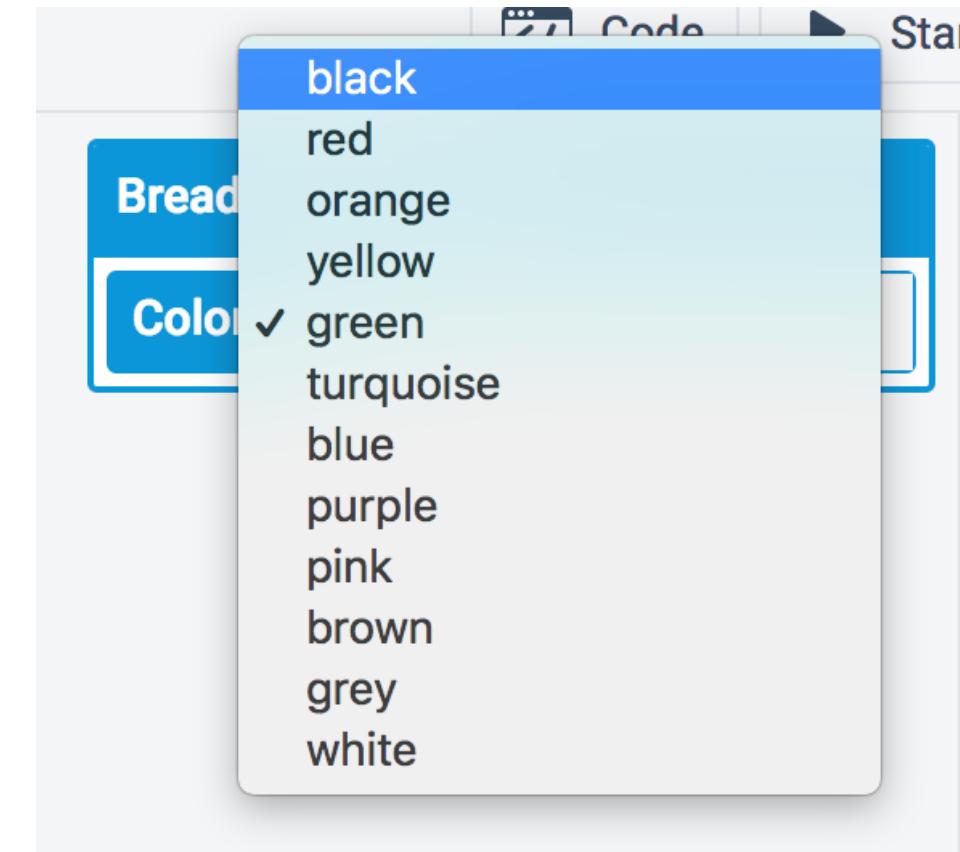


# Arduino (**PISCA-LED**)

Agora temos que fazer a ligação dos componentes

Vamos ligar a porta power GND (terra)

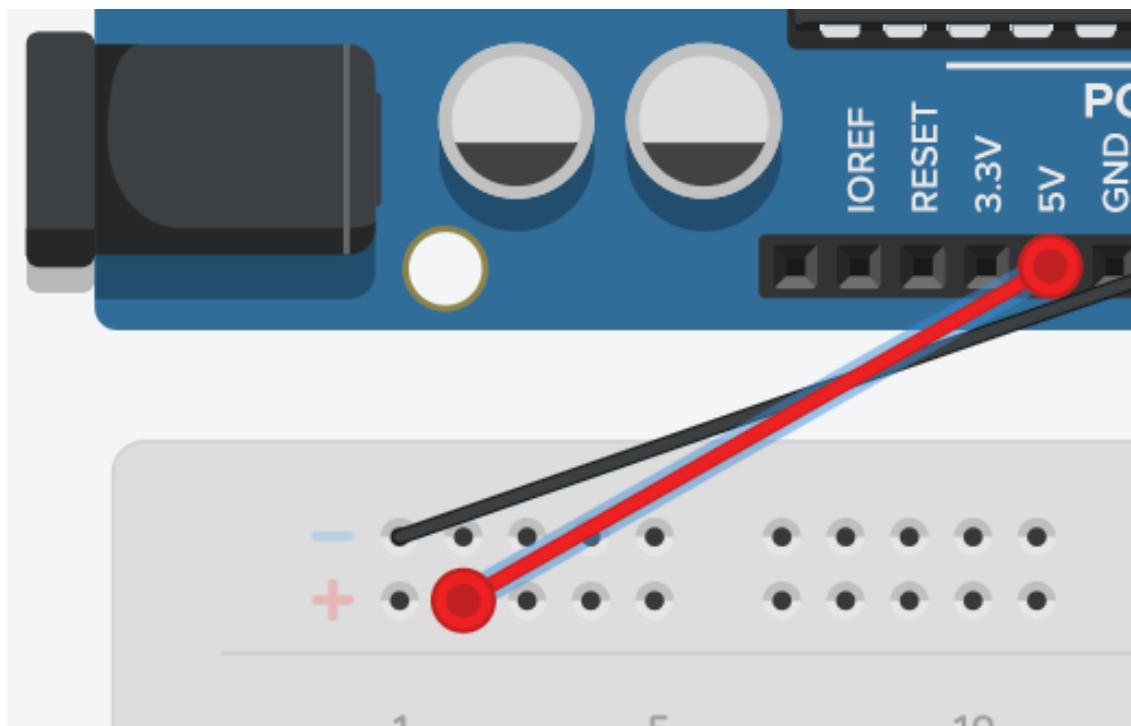
NA COR PRETA



# Arduino (**PISCA-LED**)

Agora temos que fazer a ligação dos componentes

Vamos ligar a porta fonte 5v no canal positivo

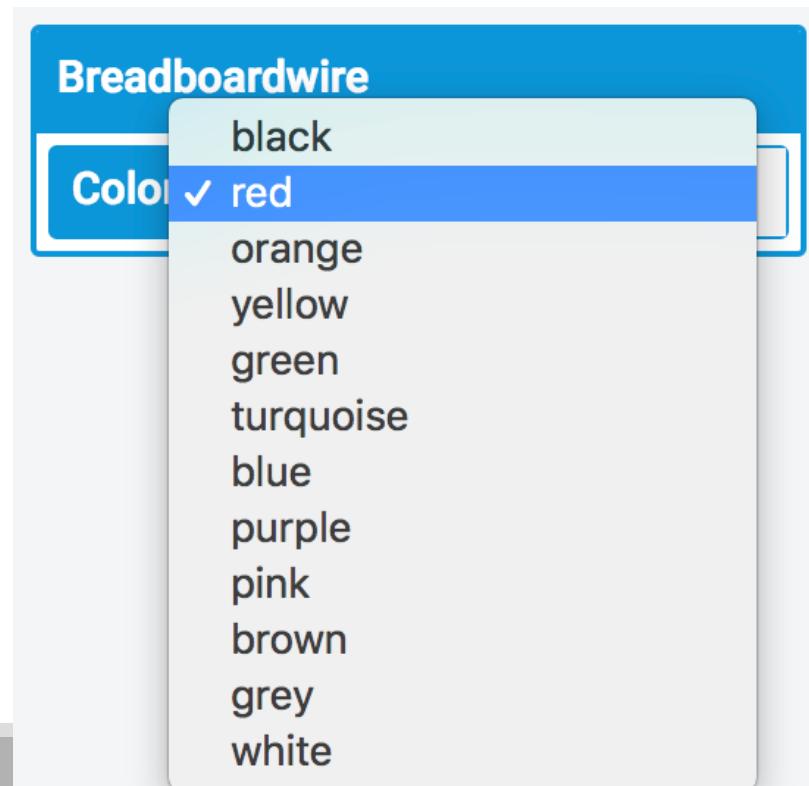


# Arduino (**PISCA-LED**)

Agora temos que fazer a ligação dos componentes

Vamos ligar a porta fonte 5v no canal positivo

NA COR VERMELHA



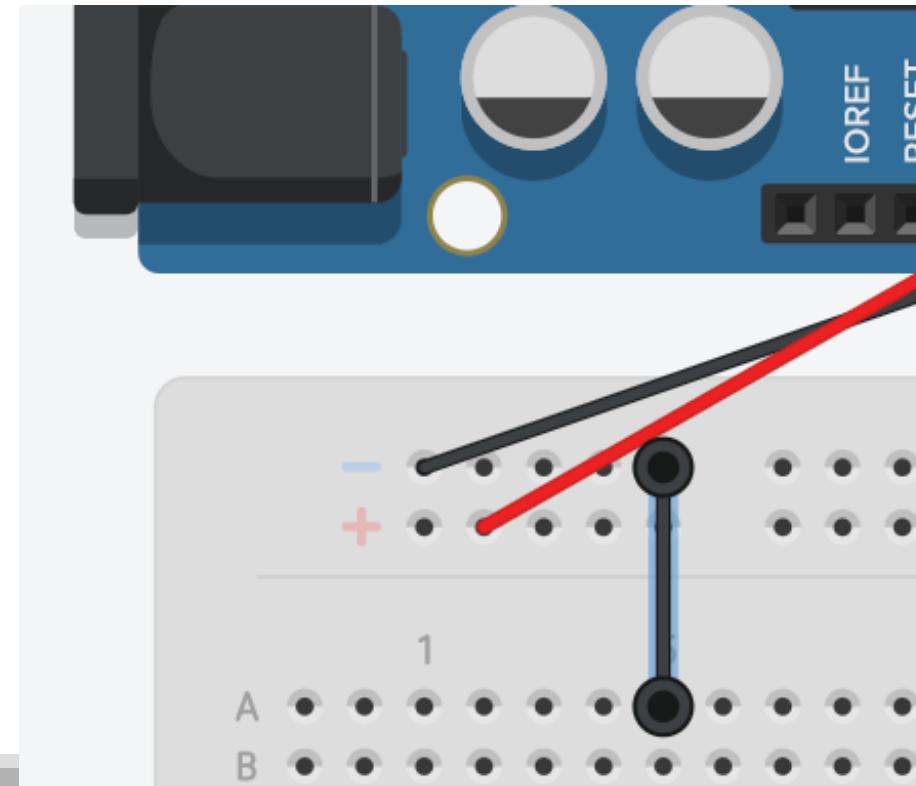
# Arduino (PISCA-LED)

Agora temos que fazer a ligação dos componentes

Vamos ligar o canal GND para o restante dos componentes da protoboard

A5 conectado em negativo 5

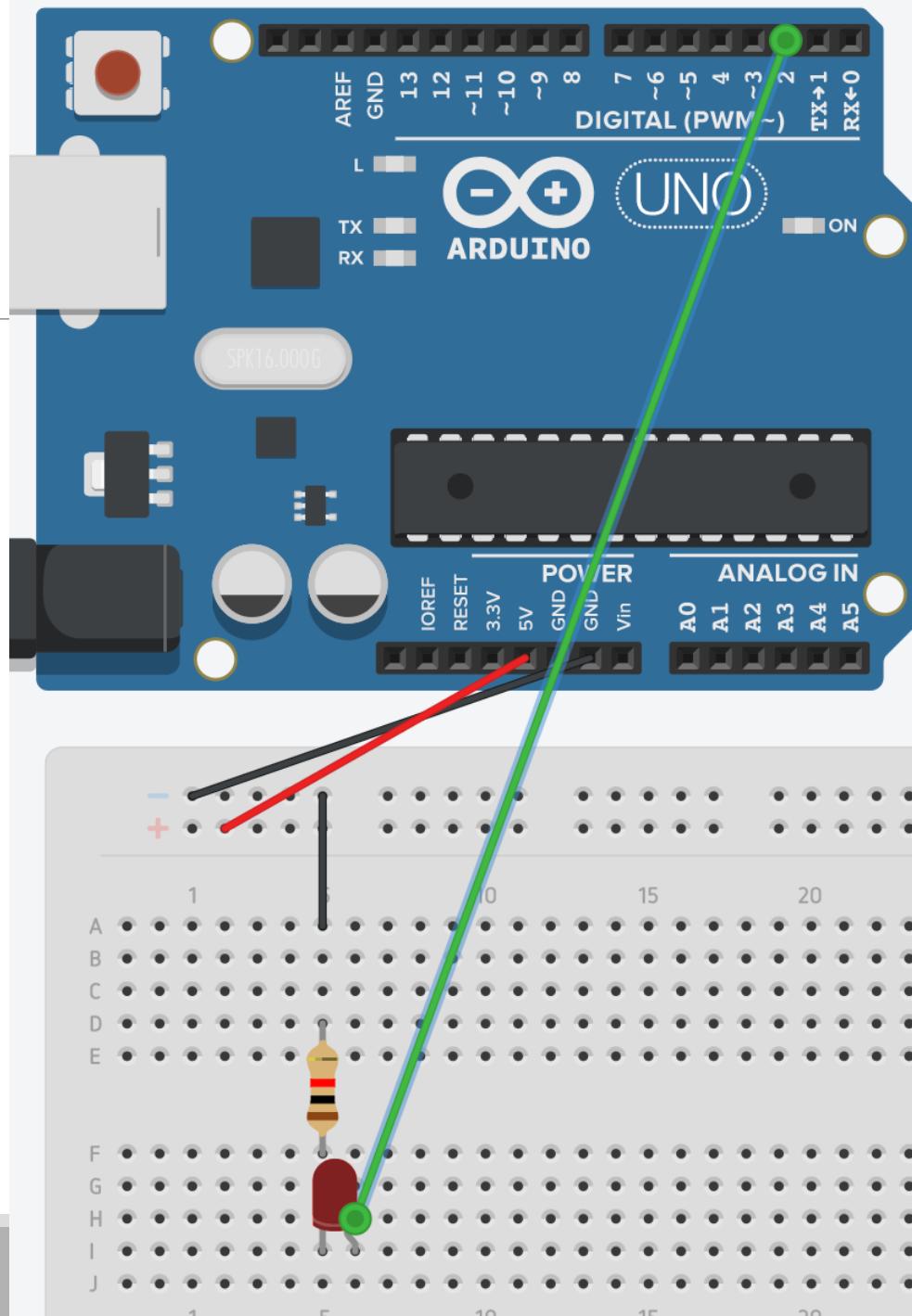
Cor preta



# Arduino (PISCA-LED)

Agora temos que fazer a ligação dos componentes

Vamos ligar o h6 com o pino 2 do arduino



# Programação

# Programação

---

O Arduino é uma placa programável

Desenvolvemos a programação no computador e enviamos o código para processamento na placa

# Programação

---

A estrutura de programação

```
1 void setup()
2 {
3
4 }
5
6 void loop()
7 {
8
9 }
```

# Programação

## A estrutura de programação

```
1 void setup()
2 {
3
4 }
```

Sempre executaremos uma função  
chamada setup

Executamos apenas 1 vez, ao ligar  
o arduino

```
5
6 void loop()
7 {
8
9 }
```

# Programação

## A estrutura de programação

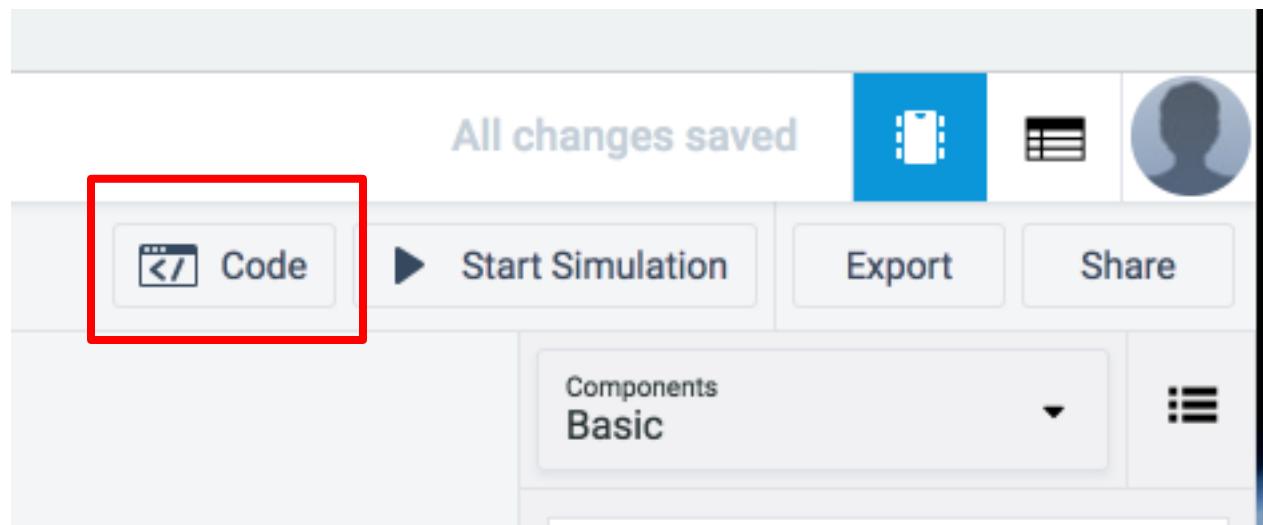
```
1 void setup()
2 {
3
4 }
5
6 void loop()
7 {
8
9 }
```

Depois, executamos o Loop

Essa função irá executar para  
sempre até que o arduino seja  
desligado

# Arduino (**PISCA-LED**)

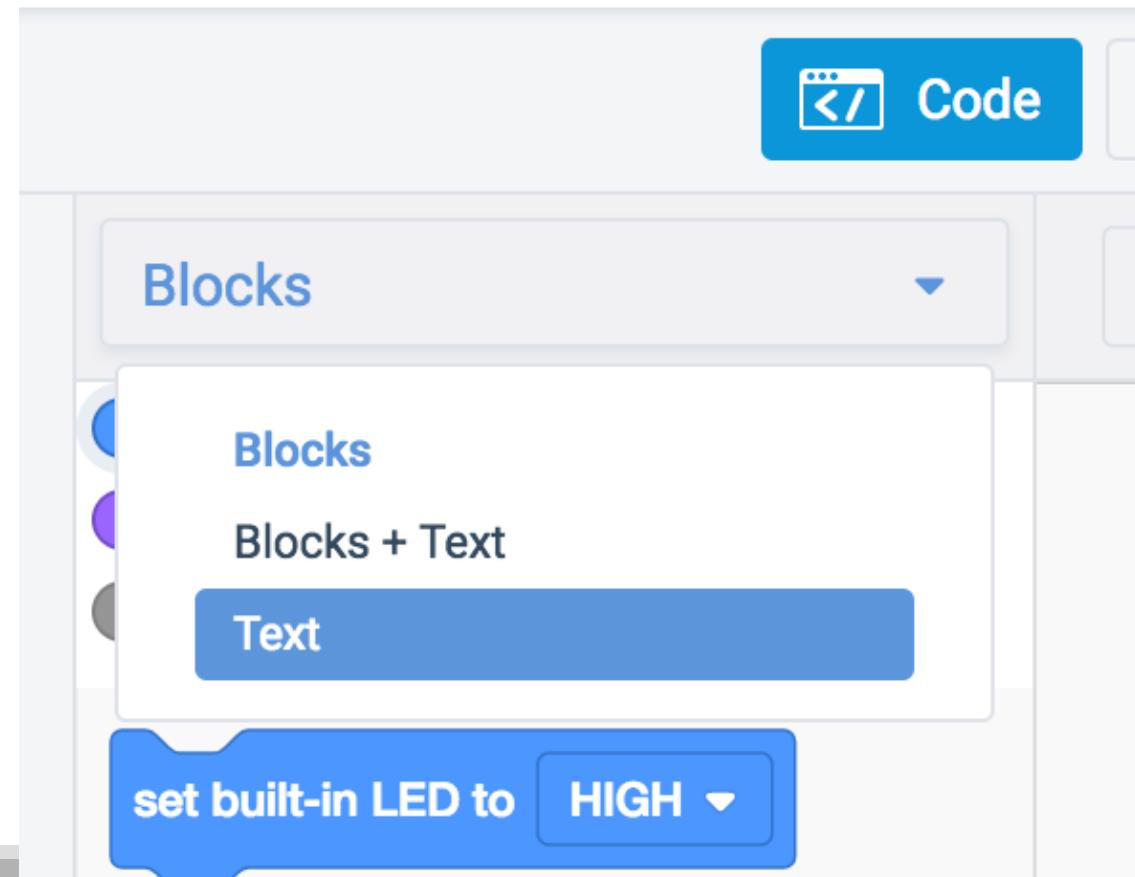
Por fim, vamos programar o comportamento do circuito, acessando o código



# Arduino (**PISCA-LED**)

Por fim, vamos programar o comportamento do circuito, acessando o código

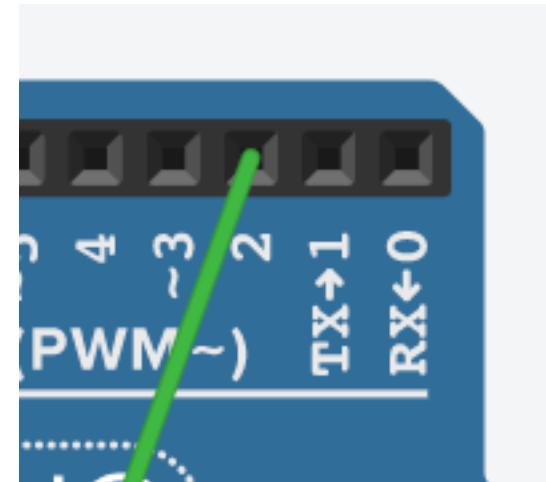
Acesso o código em **TEXTO**



# Arduino (**PISCA-LED**)

Por fim, vamos programar o comportamento do circuito, acessando o código  
O código padrão gerencia o pino 13 do arduino

Mas, nós conectamos o pino 2



# Arduino (**PISCA-LED**)

Por fim, vamos programar o comportamento do circuito, acessando o código  
Então, no código, substituímos o pino 13 pelo pino 2

```
1 void setup()
2 {
3     pinMode(2, OUTPUT);
4 }
5
6 void loop()
7 {
8     digitalWrite(2, HIGH);
9     delay(1000); // Wait for 1000 millisecond(s)
10    digitalWrite(2, LOW);
11    delay(1000); // Wait for 1000 millisecond(s)
12 }
```

# Estudo

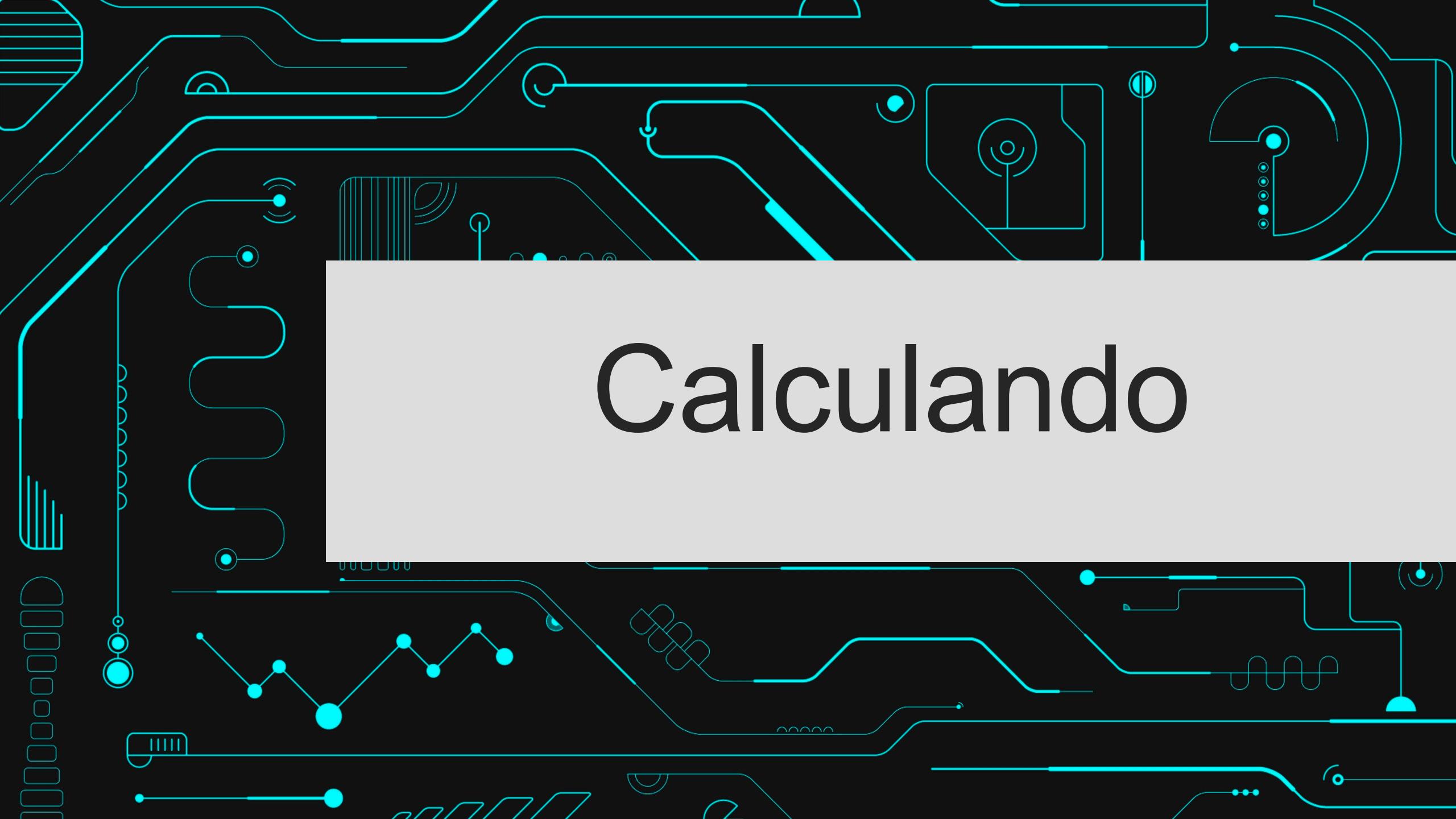
# Estudo

---

No projeto criado, altere

- Primeiro altere o delay para 200 e veja a diferença
- Depois, acrescente um novo led e pisque um de cada vez

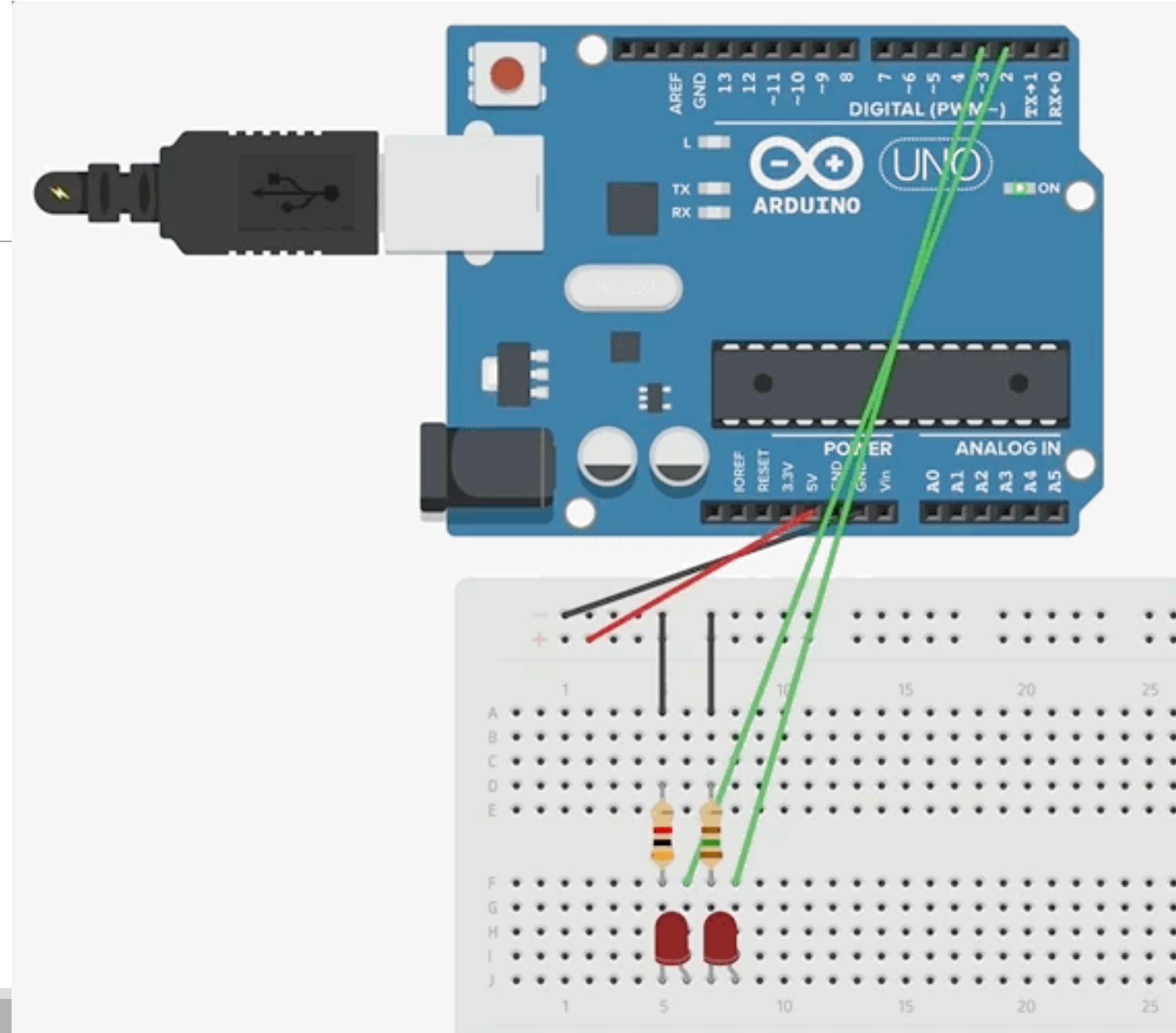




# Calculando

# Resistor

Note a diferença



# Resistor

---

Note a diferença

- Um LED está piscando com mais intensidade que o outro
- PORQUE ISSO?????

# Resistor

---

Isso ocorre pois a resistência do resistor não está calculada corretamente para o LED de menor intensidade

Então, vamos dimensionar os componentes

# Resistor

---

Isso ocorre pois a resistência do resistor não está calculada corretamente para o LED de menor intensidade

Então, vamos dimensionar os componentes

- Temos um cabo ligado no PINO 2 do arduino em um LED
- Temos que conectar um resistor ao LED para que a corrente de 5V do arduino não ligue diretamente no LED (2V)
  - O LED queimaria
- Com isso, 2V ficarão para o LED e 3V para o resistor

# Resistor

---

Então, vamos aplicar a Lei de Ohm ( $\Omega$ )

- Tensão (V) do Pino2 do Arduino = 5V
- Tensão do LED = 2V
- Corrente (I) do LED = 20mA
  
- Esta é a corrente e tensão padrão para um LED simples

# Resistor

---

Então, vamos aplicar a Lei de Ohm ( $\Omega$ )

- Tensão (V) do Pino2 do Arduino = 5V
- Tensão do LED = 2V
- Corrente (I) do LED = 20mA
  
- Tensão do resistor = ????

# Resistor

---

Então, vamos aplicar a Lei de Ohm ( $\Omega$ )

- Tensão (V) do Pino2 do Arduino = 5V
- Tensão do LED = 2V
- Corrente (I) do LED = 20mA
  
- Tensão do resistor =  $V = VR1 + V_{led}$

$$5V = VR1 + 2V$$

$$VR1 = 5V - 2V$$

$$VR1 = 3V$$

# Resistor

---

Então, vamos aplicar a Lei de Ohm ( $\Omega$ )

- Tensão (V) do Pino2 do Arduino = 5V
- Tensão do LED = 2V
- Corrente (I) do LED = 20mA
  
- Tensão do resistor = 3V

# Resistor

---

Então, vamos aplicar a Lei de Ohm ( $\Omega$ )

- Tensão (V) do Pino2 do Arduino = 5V
- Tensão do LED = 2V
- Corrente (I) do LED = 20mA
  
- Tensão do resistor = 3V
  
- Corrente do resistor = ????

# Resistor

---

Então, vamos aplicar a Lei de Ohm ( $\Omega$ )

- Tensão (V) do Pino2 do Arduino = 5V
- Tensão do LED = 2V
- Corrente (I) do LED = 20mA
  
- Tensão do resistor = 3V
  
- Corrente do resistor =  $I_{led}$

# Resistor

---

Então, vamos aplicar a Lei de Ohm ( $\Omega$ )

- Tensão (V) do Pino2 do Arduino = 5V
- Tensão do LED = 2V
- Corrente (I) do LED = 20mA
  
- Tensão do resistor = 3V
  
- Corrente do resistor = O circuito possui a mesma corrente (20mA)

# Resistor

---

Então, vamos aplicar a Lei de Ohm ( $\Omega$ )

- Tensão (V) do Pino2 do Arduino = 5V
- Tensão do LED = 2V
- Corrente (I) do LED = 20mA
  
- Tensão do resistor = 3V
  
- Corrente do resistor = O circuito possui a mesma corrente (20mA)
- Resistência = ?????

# Resistor

---

Então, vamos aplicar a Lei de Ohm ( $\Omega$ )

- Pela Lei de Ohm a resistência é calculada pela seguinte fórmula

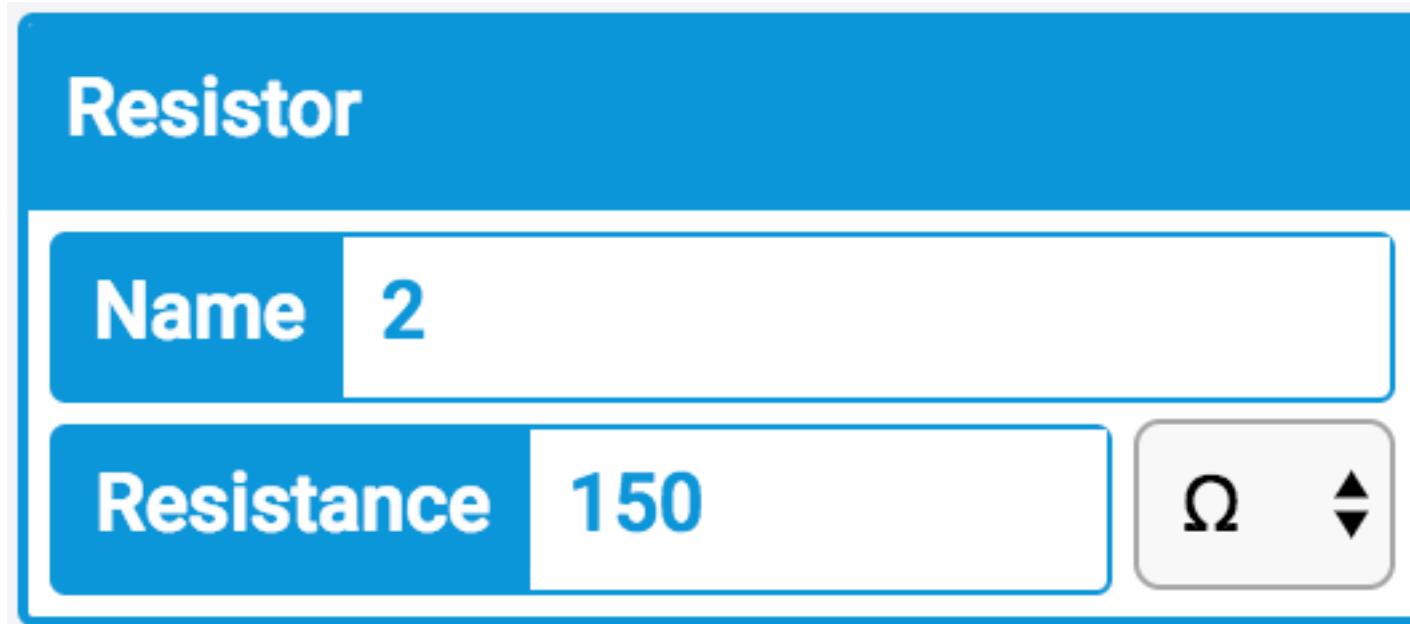
$$R1 = \frac{Vr1}{Lr1}$$

$$R1 = \frac{3V}{20mA}$$

$$R1 = 150\Omega$$

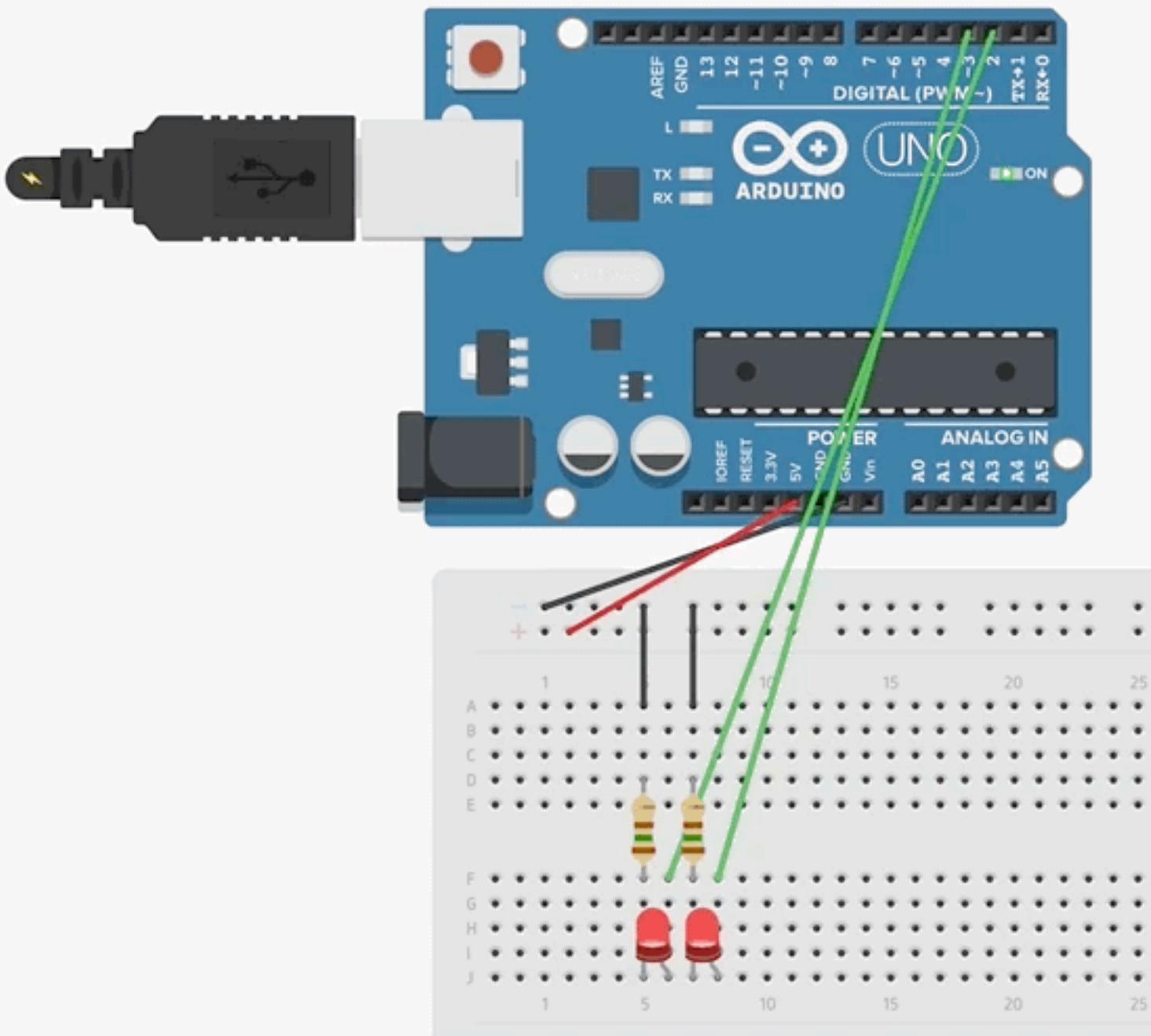
# Resistor

Aplicamos, agora, esta resistência ao resistor do circuito



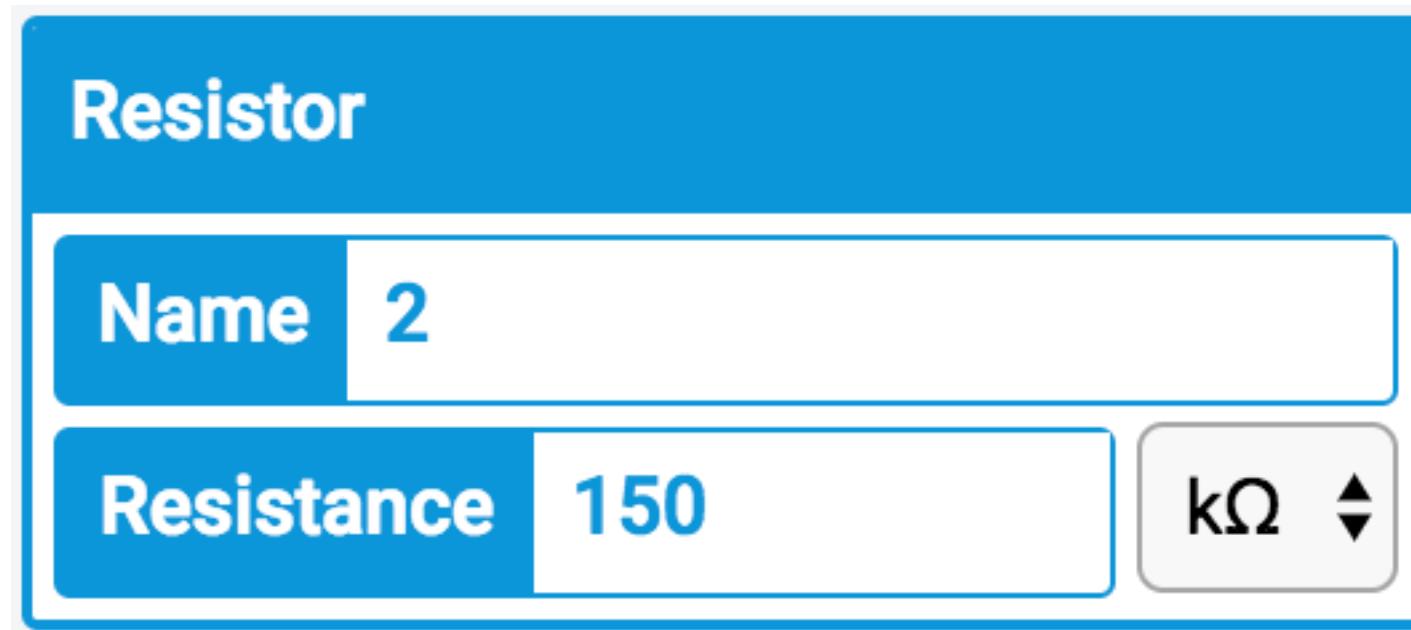
# Resist

Aplicamos,



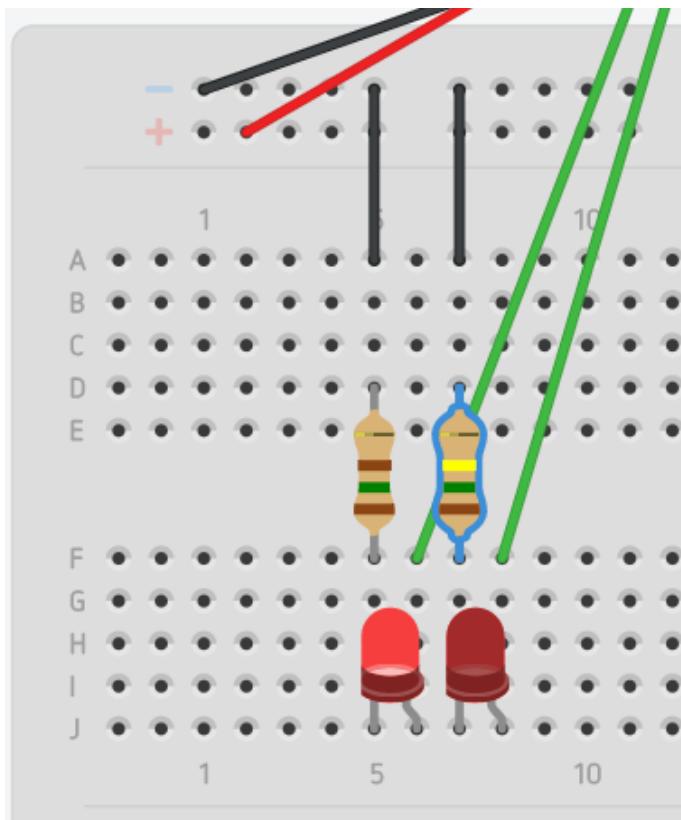
# Resistor

Se aplicarmos uma resistência superior a calculada??



# Resistor

Se aplicarmos uma resistência superior a calculada??

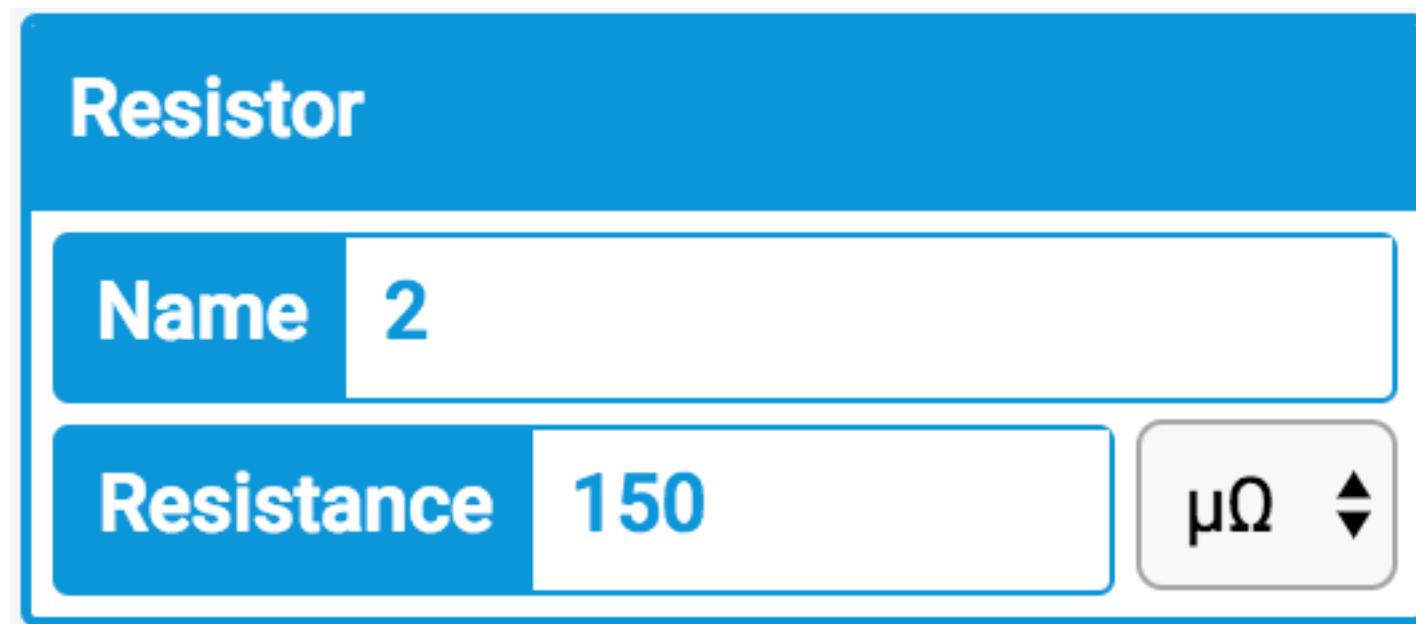


O segundo LED (resistência superior) terá menor luminosidade

Isso ocorre pois o resistor acaba tendo a maior voltagem

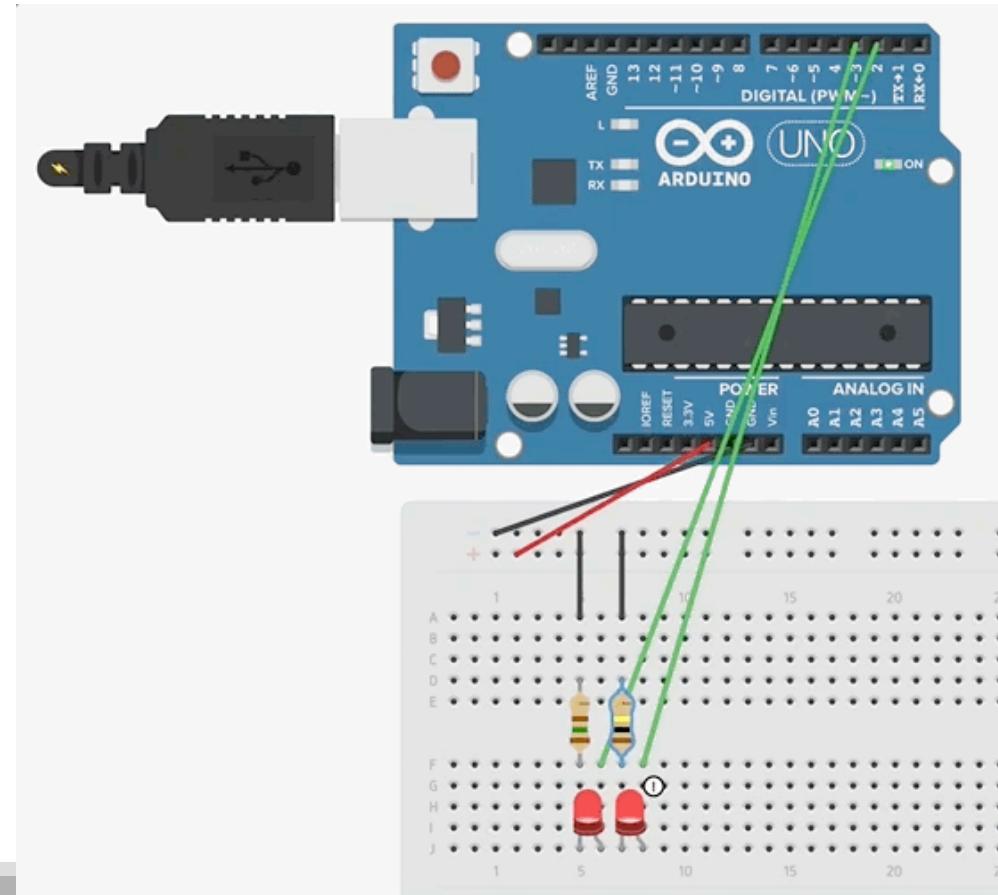
# Resistor

Se aplicarmos uma resistência menor a calculada??



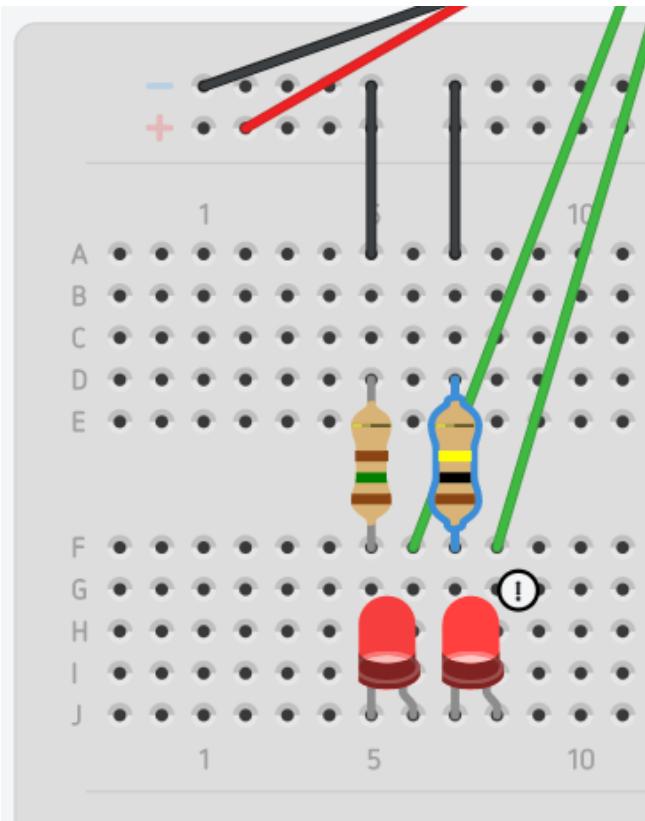
# Resistor

Se aplicarmos uma resistência menor a calculada??



# Resistor

Se aplicarmos uma resistência menor a calculada??



O segundo LED (resistência menor) irá queimar

Isso ocorre pois o resistor acaba ficando com uma menor tensão, permitindo uma voltagem maior ao LED

# Estudo

# Estudo

## Pesquisa sobre resistores

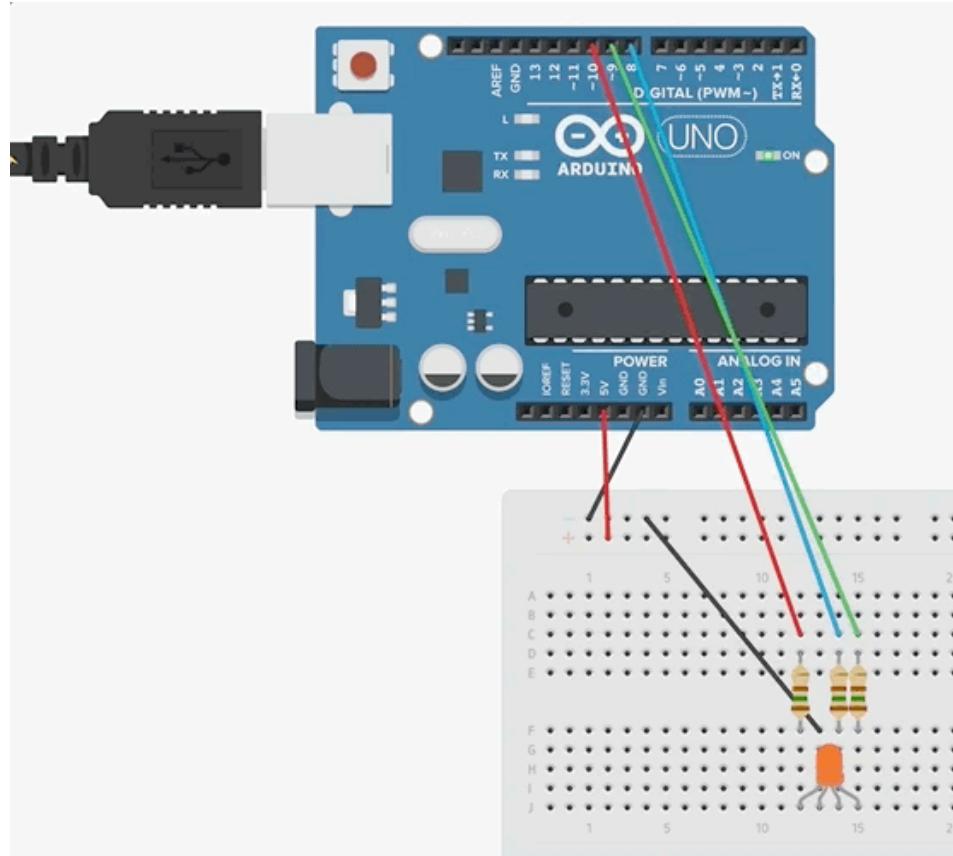
---

- Entender o motivo de os resistores apresentarem cores diferentes
- Apresentar e explicar as combinações de cores em resistores

# Projeto LED RGB

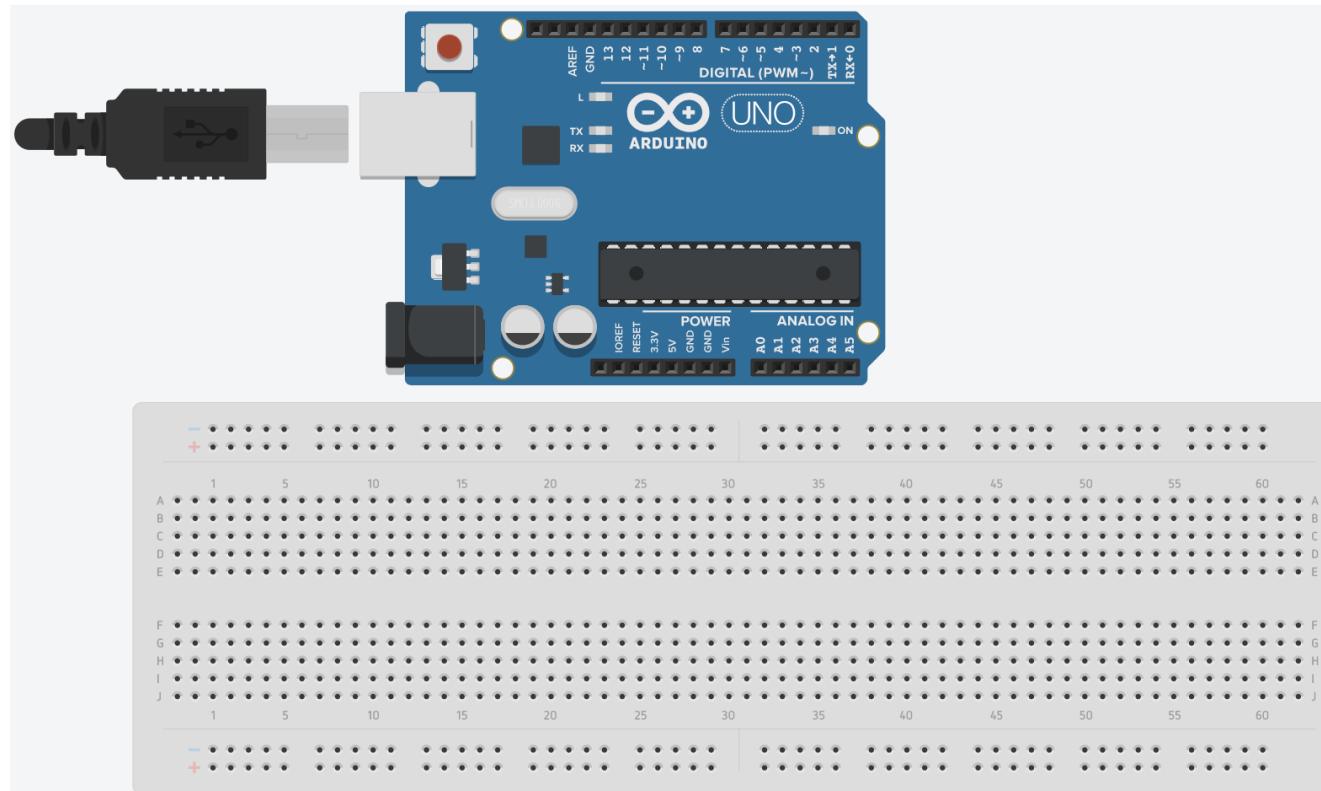
# Arduino

Vamos projetar circuito com **LED RGB**



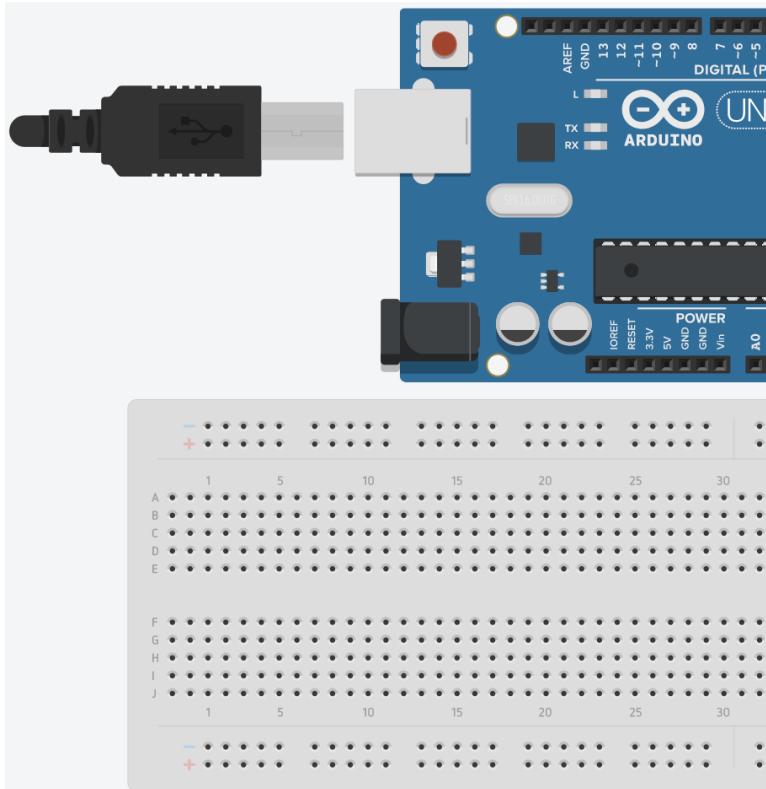
# Arduino

# Vamos projetar circuito com **LED RGB**

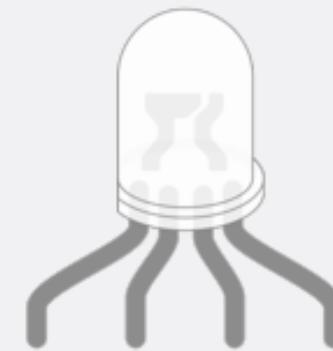


# Arduino

Vamos projetar circuito com **LED RGB**



rgb

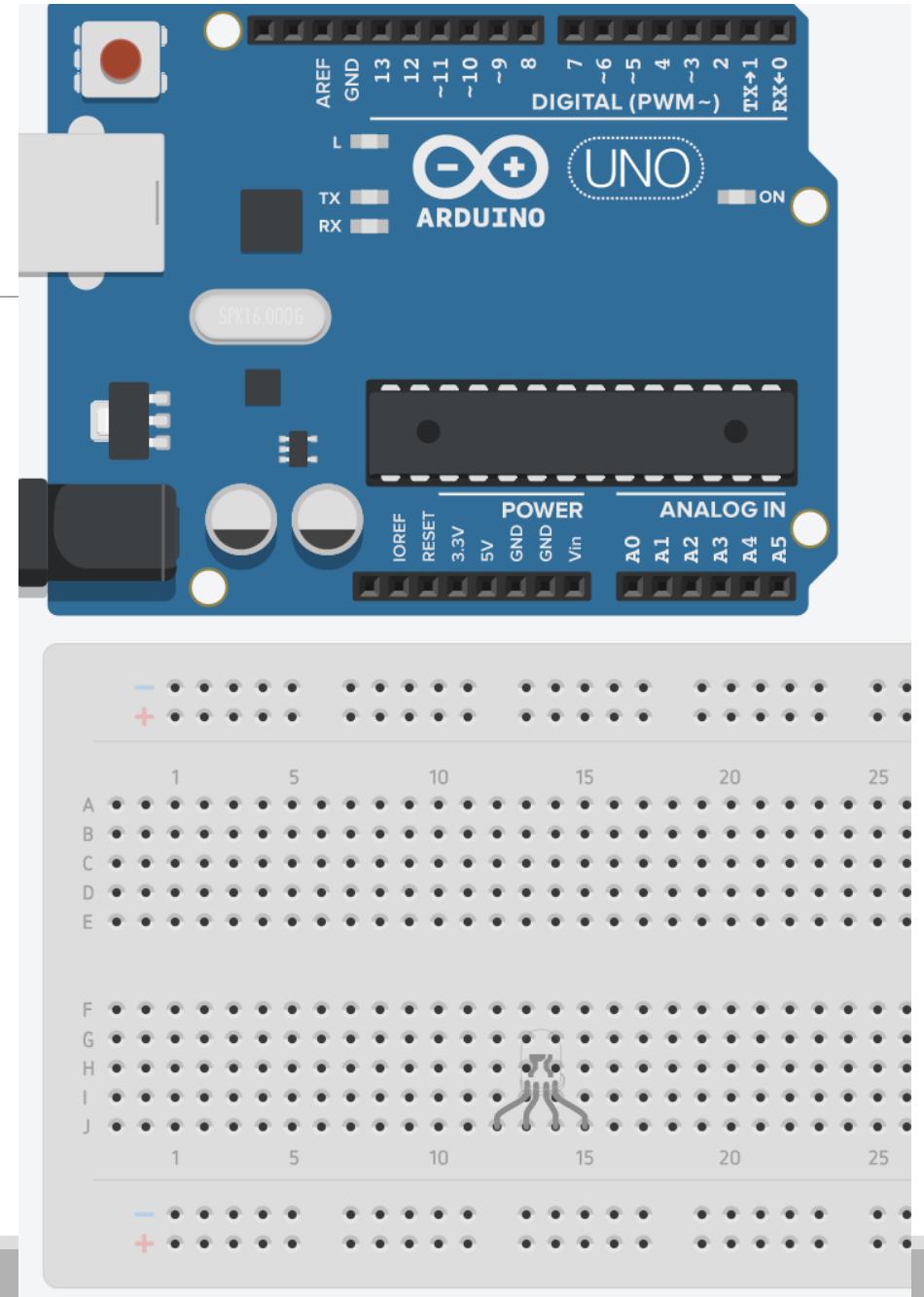


LED RGB

# Arduino

Vamos projetar circuito com **LED RGB**

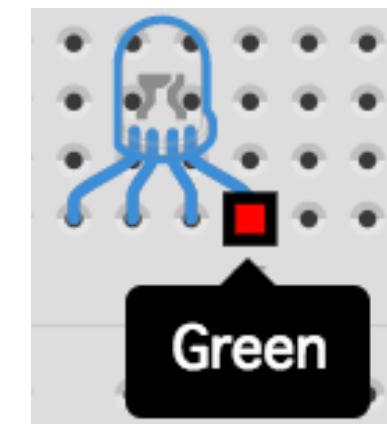
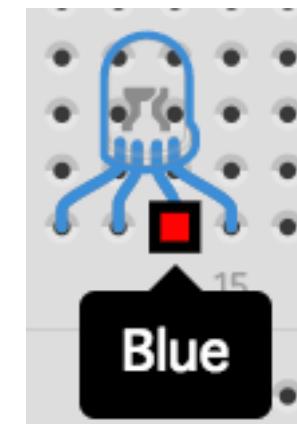
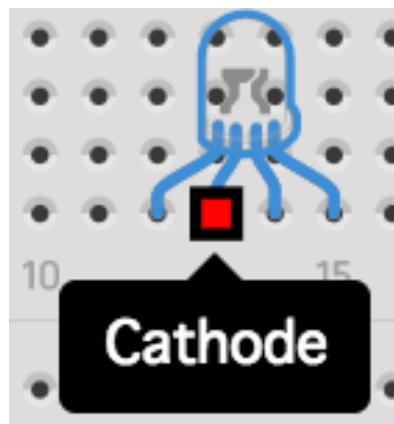
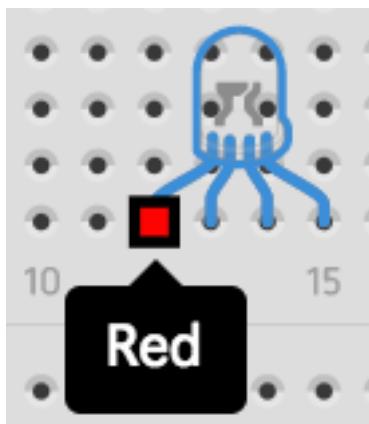
Adicionaremos o LED RGB nos pinos 12 13 14 15



# Arduino

Vamos projetar circuito com **LED RGB**

Adicionaremos o LED RGB nos pinos 12 13 14 15



# Estudo

---

Montar o circuito com LED RGB alterando cores

- Dicas
  - É apenas 1 LED
  - Porém, são 3 entradas de cores Red Blue Green
  - Então.... 3 pinos.....
  - A combinação das 3 cores geram outras tantas cores!!

# Estudo

---

Montar o circuito com LED RGB com LED normal

- Agora, deve-se criar um circuito com 1 LED RGB e 5 LEDs normais (azul, verde, vermelho, laranja e amarelo)
- Piscar um LED normal por vez (3x por segundo) enquanto o RGB fica na cor do LED normal

# Estudo

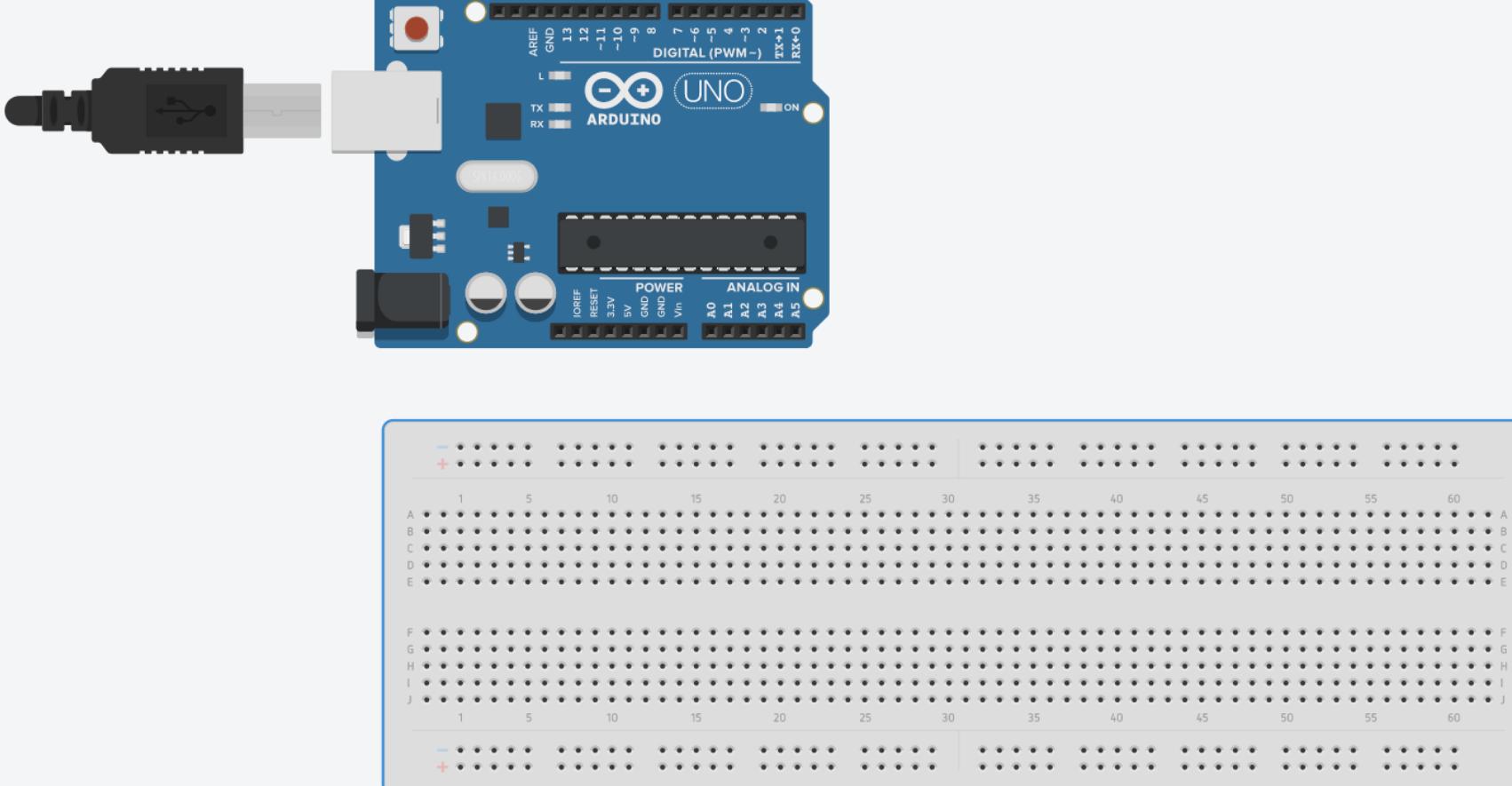
---

## Arduino IDE

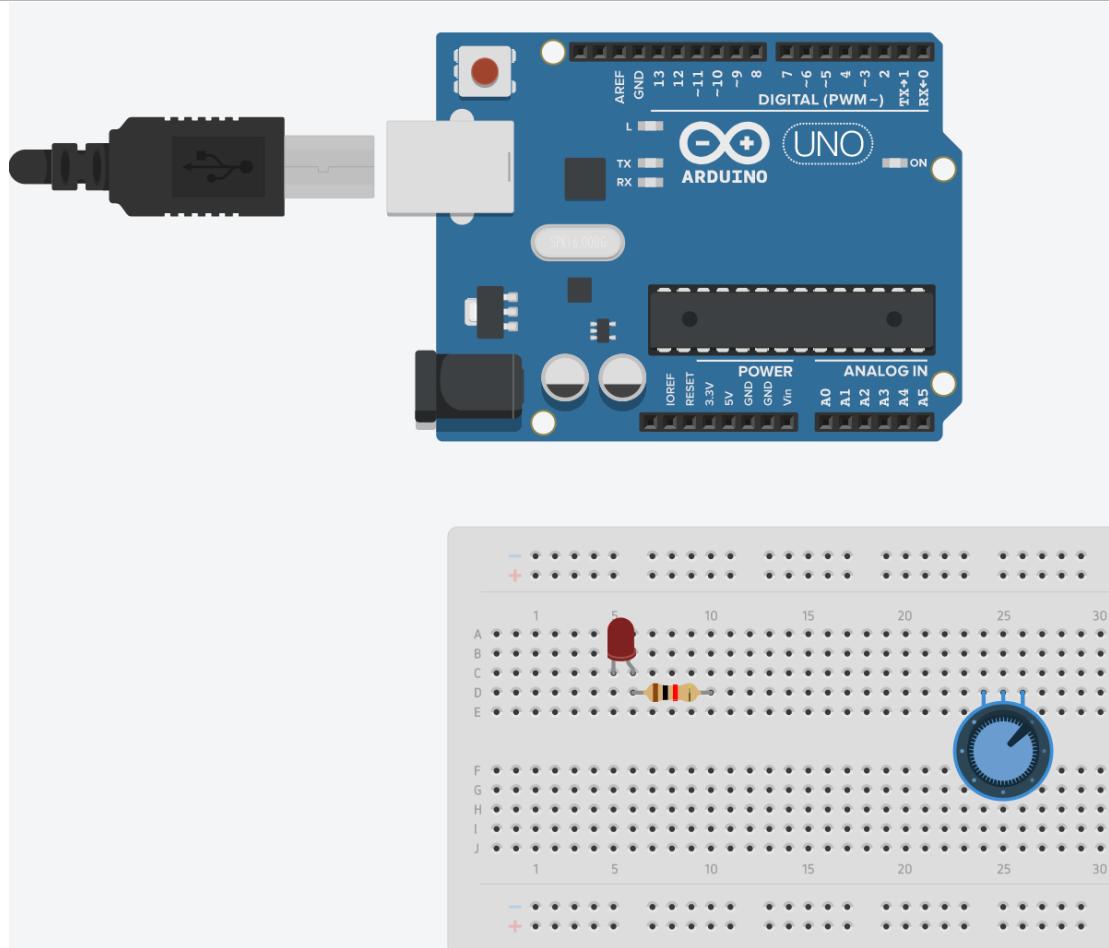
- Baixar (caso não tenha ainda) o Arduino IDE

# Projeto LED com Potenciômetro

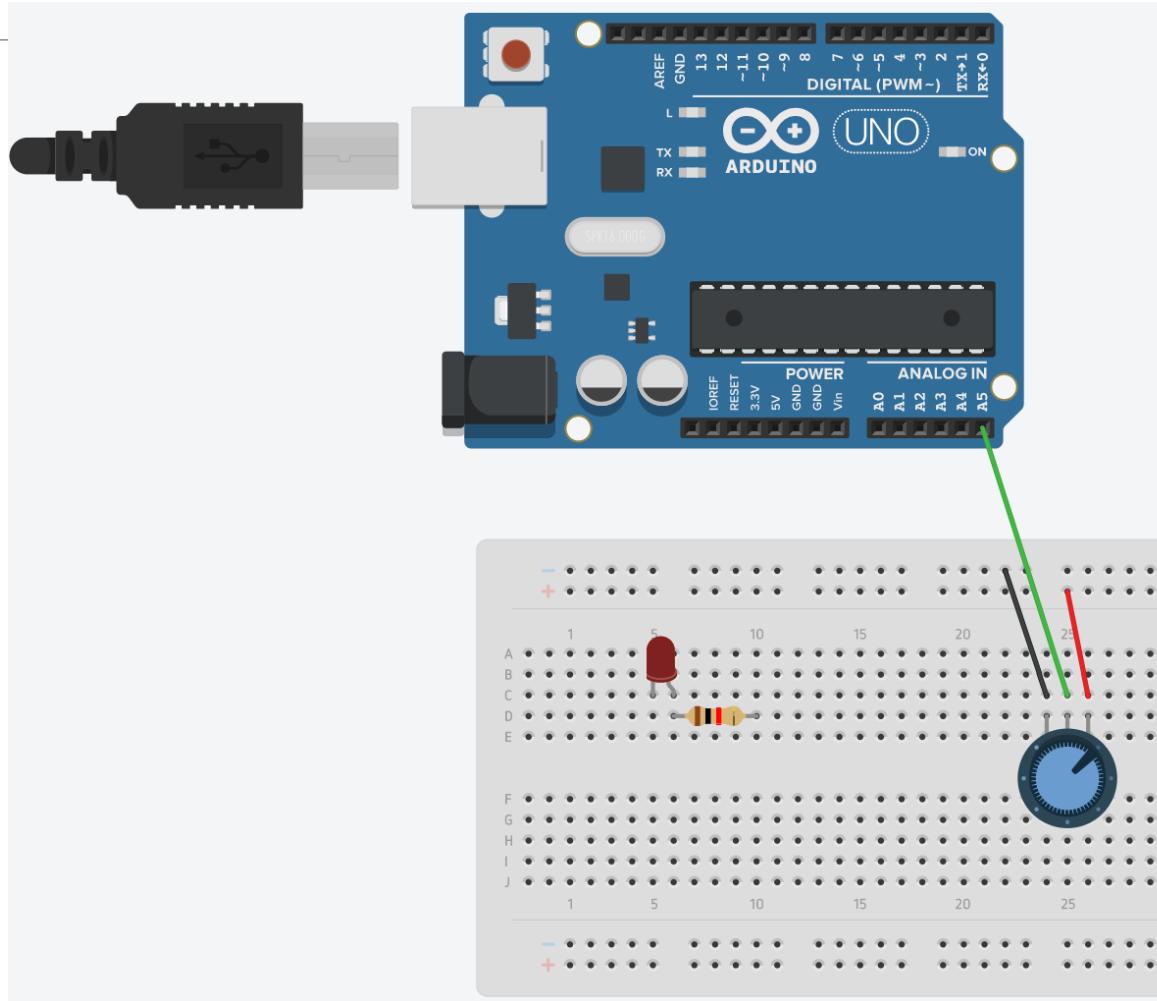
# LED com Potenciômetro



# LED com Potenciômetro

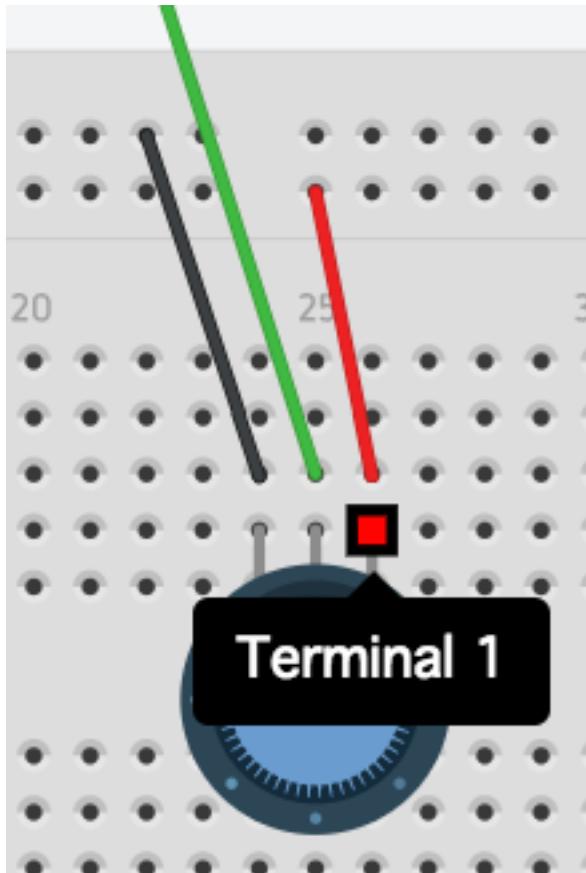


# LED com Potenciômetro

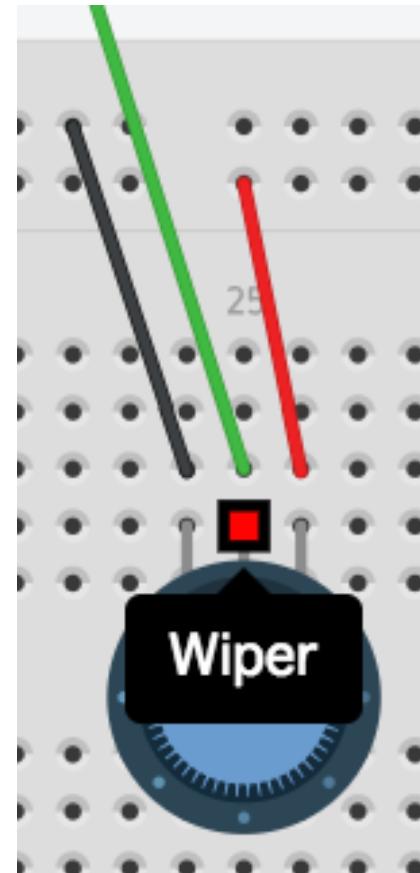


# LED com Potenciômetro

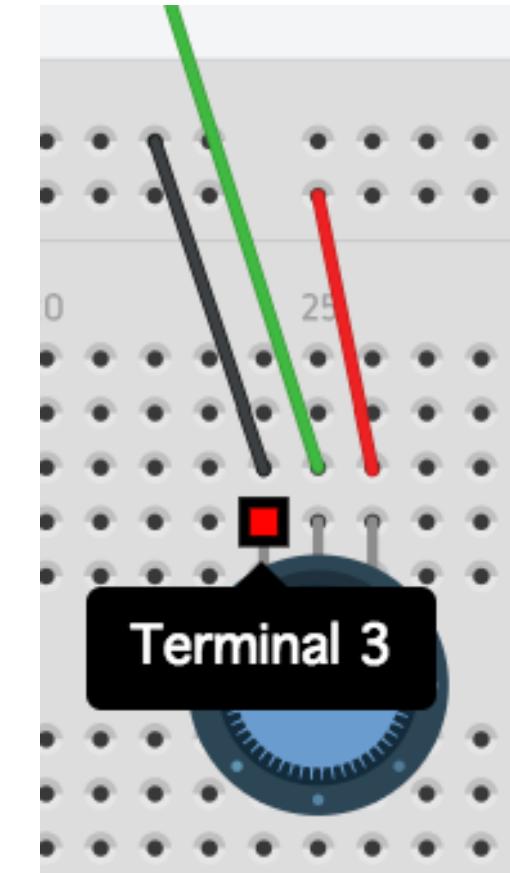
Terminal conectado na fonte



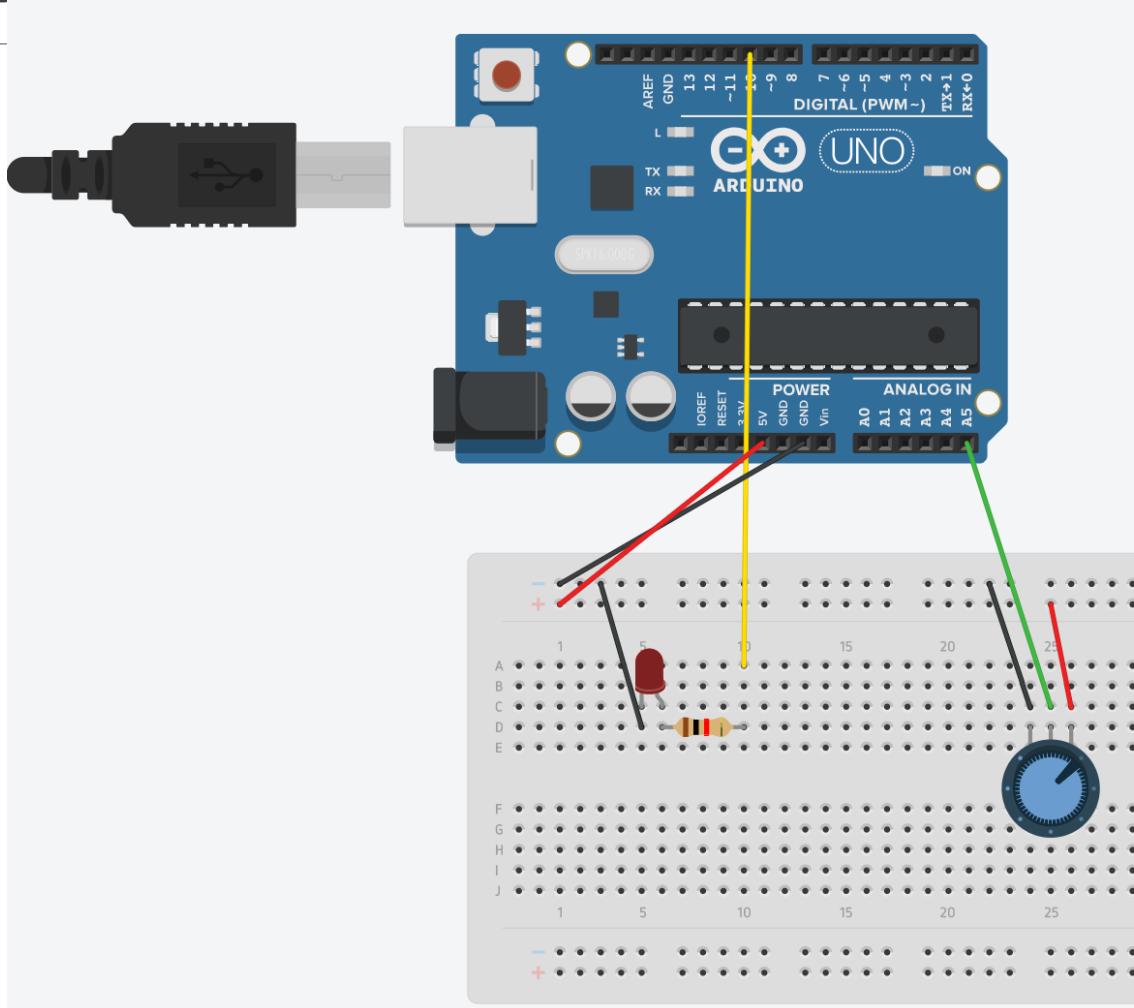
Terminal de saída



Terminal conectado no terra



# LED com Potenciômetro



# LED com Potenciômetro

# LED com Potenciômetro

```
1 const int pinoLED = 10; /* PINO DIGITAL  
2 UTILIZADO PELO LED */  
3 const int pinoPOT = A5; /* PINO ANALÓGICO UTILIZADO  
4 PELO POTENCIÔMETRO */  
5 int leituraA5 = 0; /* VARIÁVEL QUE ARMAZENA O  
6 VALOR LIDO NA PORTA ANALÓGICA */  
7  
8 float luminosidadeLED = 0; /* VARIÁVEL QUE ARMAZENA O VALOR DA  
9 LUMINOSIDADE QUE SERÁ APLICADA AO LED */  
10  
11 void setup(){  
12     pinMode(pinoLED, OUTPUT); // DEFINE A PORTA COMO SAÍDA  
13     pinMode(pinoPOT, INPUT); // DEFINE A PORTA COMO ENTRADA  
14 }  
15  
16 void loop(){  
17     leituraA5 = analogRead(pinoPOT); /* LÊ O VALOR NA PORTA ANALÓGICA  
18         (VALOR LIDO EM BITS QUE VAI DE  
19             0 A 1023 BITS) */  
20     luminosidadeLED = map(leituraA5, 0, 1023, 0, 255); /* EXECUTA A FUNÇÃO "map"  
21         DE ACORDO COM OS PARÂMETROS  
22             PASSADOS */  
23     analogWrite(pinoLED, luminosidadeLED); /* APLICA AO LED O VALOR DE LUMINOSIDADE  
24             GERADO PELA FUNÇÃO "map" */  
25 }
```

**Função MAP mapeia um  
intervalo atual para um novo  
intervalo**