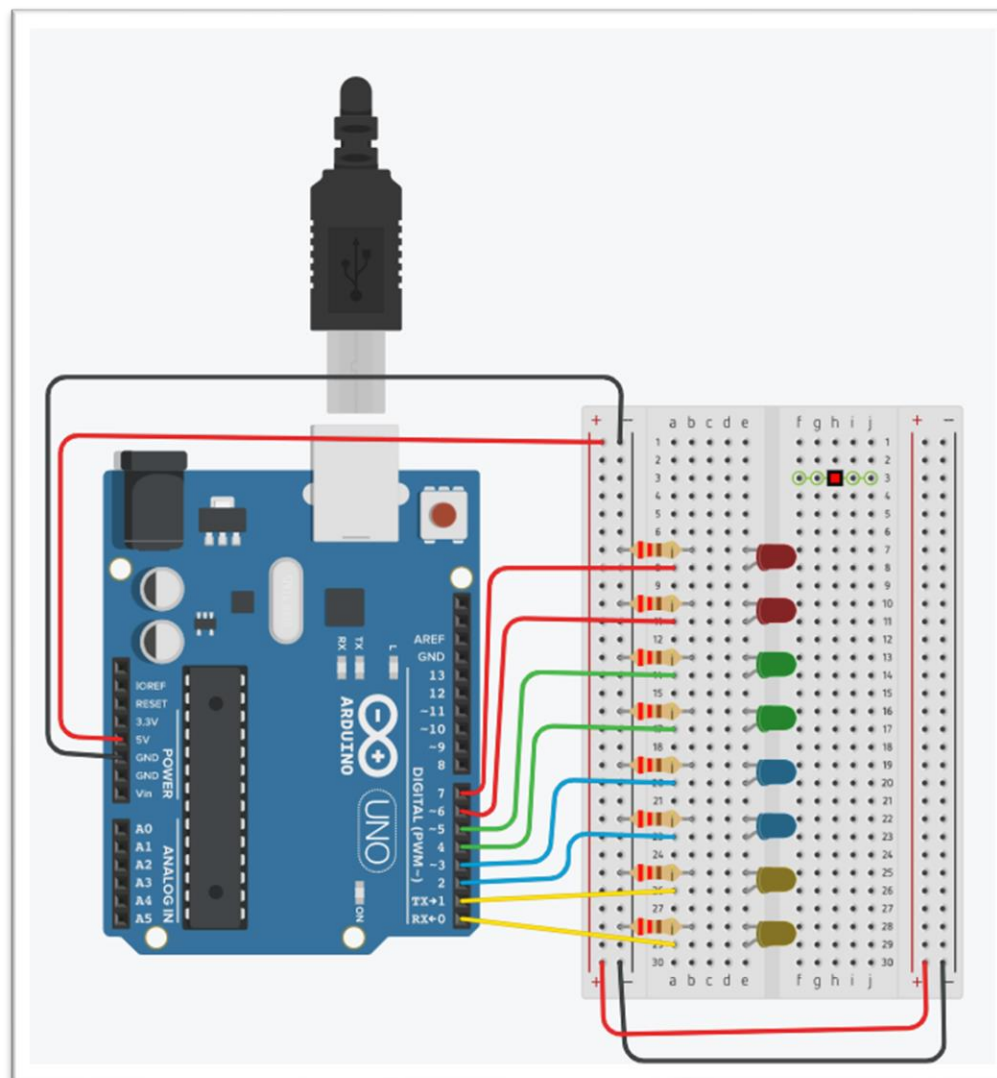


PLACA ARDUÍNO

CIRCUITOS – SHOW DE LEDS

SHOW DE LEDS

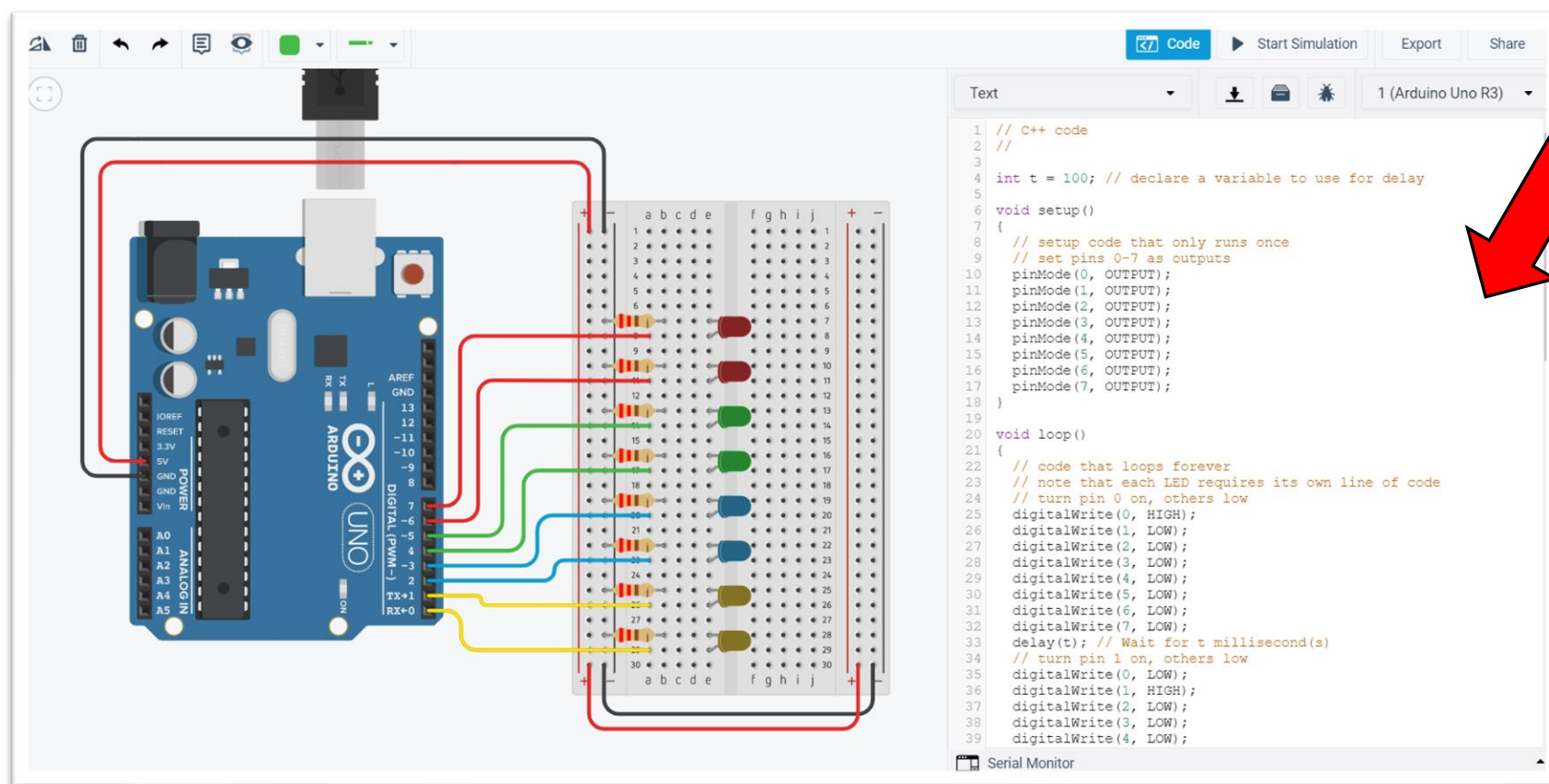
TINKERCAD



SHOW DE LEDS

TINKERCAD

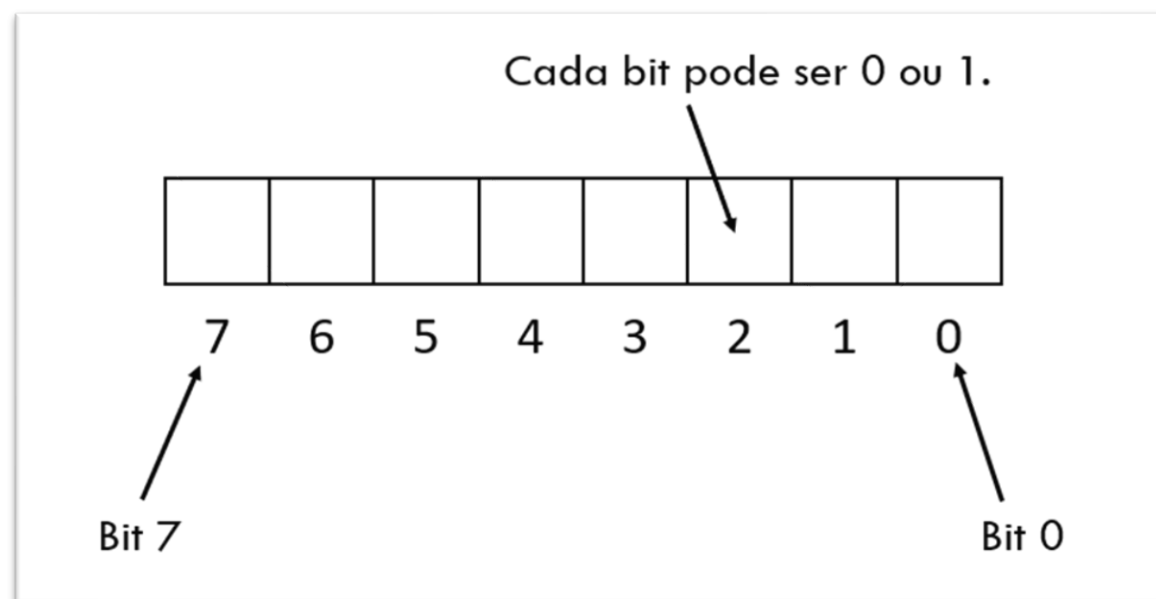
Usar `pinMode()` e `digitalWrite()` para controlar LEDs requer uma linha de código por pino para programar animações de circuitos com muitos LEDs.



SHOW DE LEDS

TINKERCAD

Cada registo tem 8 bits:



Conte os bits da direita para a esquerda, começando em 0.

SHOW DE LEDS

TINKERCAD

Os bits no registo correspondem a pinos físicos no Arduino:



Bit 7

Bit 6

Bit 5

Bit 4

Bit 3

Bit 2

Bit 1

Bit 0

O registo DDRD substitui o comando `pinMode()`.

Definir um pouco como 1 define o pino como uma saída.

Definir um pouco como 0 define o pino como uma entrada.

Por exemplo:

```
DDRD = 0b11111111;
```

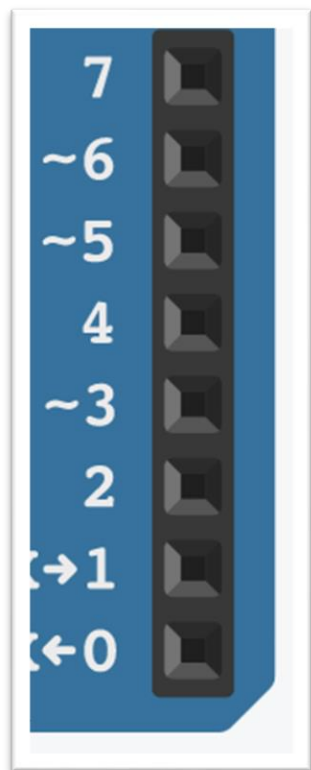
Esta linha de código define pinos de 0 a 7 como saídas.

Nota: o "0b" diz ao Arduino que este número está em binário.

SHOW DE LEDS

TINKERCAD

O registro PORTD substitui o comando digitalWrite().
Definir um pouco para 1 define o pino HIGH.
Definir um pouco como 0 define o pino BAIXO.



Por exemplo:

```
PORTD = 0b10000000;
```

Esta linha de código define o
pino 7 alto e todos os outros
pinos baixos.

SHOW DE LEDS

TINKERCAD

Agora você pode reduzir oito linhas de código em uma única linha!

```
pinMode(0, OUTPUT);  
pinMode(1, OUTPUT);  
pinMode(2, OUTPUT);  
pinMode(3, OUTPUT);  
pinMode(4, OUTPUT);  
pinMode(5, OUTPUT);  
pinMode(6, OUTPUT);  
pinMode(7, OUTPUT);
```

→ `DDRD = 0b11111111;`

```
digitalWrite(0, HIGH);  
digitalWrite(1, LOW);  
digitalWrite(2, LOW);  
digitalWrite(3, LOW);  
digitalWrite(4, LOW);  
digitalWrite(5, LOW);  
digitalWrite(6, LOW);  
digitalWrite(7, LOW);
```

→ `PORTD = 0b10000000;`

SHOW DE LEDS

TINKERCAD

Eles são controlados pelos registros DDRB e PORTB.

DDRB: definir o modo de pino (entrada ou saída)

PORTB: definir pino alto ou baixo se for uma saída



Atenção: os números de bits nos registos não correspondem aos números de pinos do Arduino! Isso pode ser confuso, por isso é mais fácil trabalhar com pinos de 0 a 7 quando você está aprendendo a usar registradores pela primeira vez.

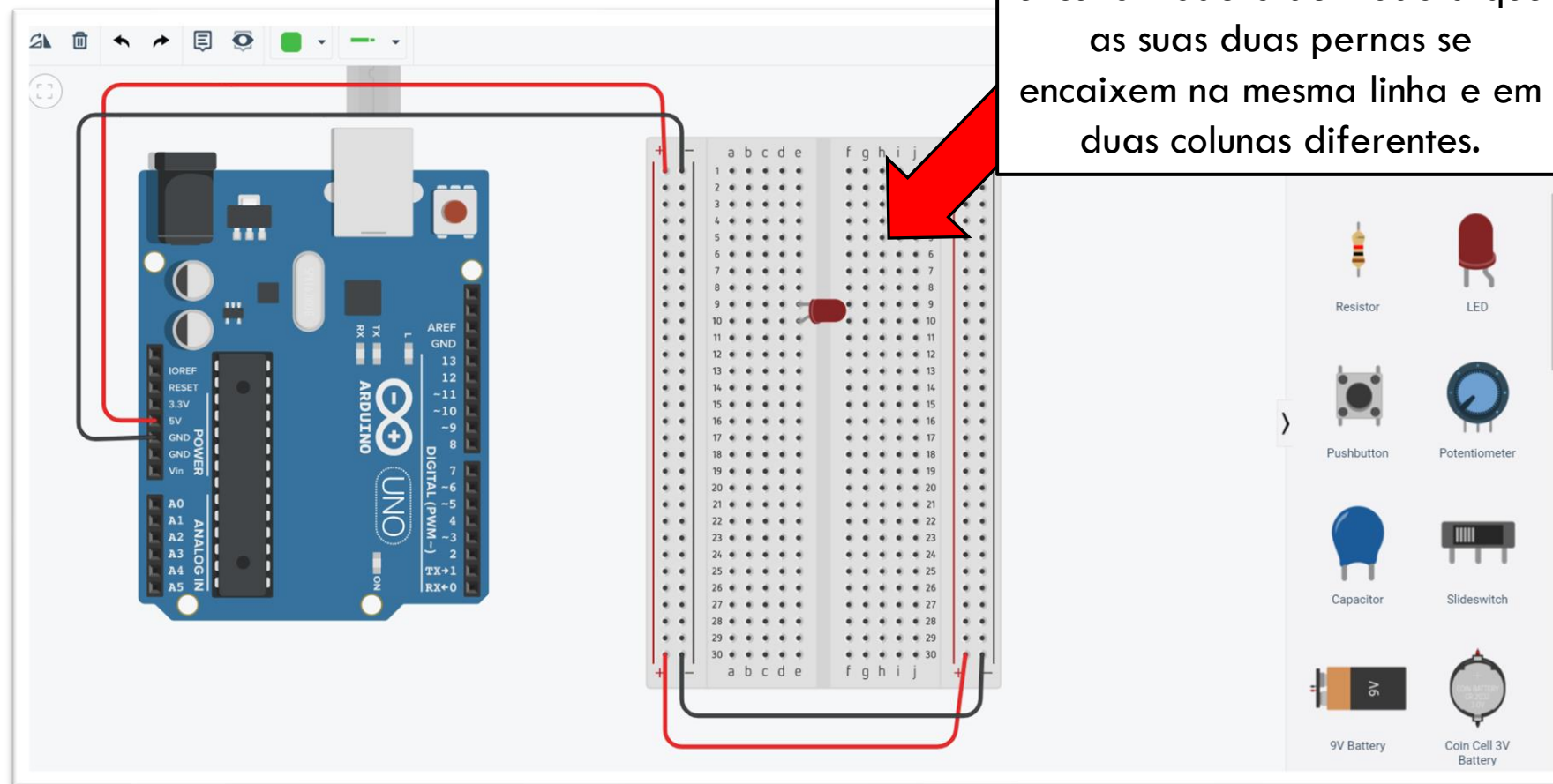
Por exemplo, essas duas linhas de código definiriam todos os pinos como saídas e, em seguida, definiriam o pino 13 (bit 5) de altura:

```
DDRB = 0b00111111;
```

```
PORTB = 0b00100000;
```

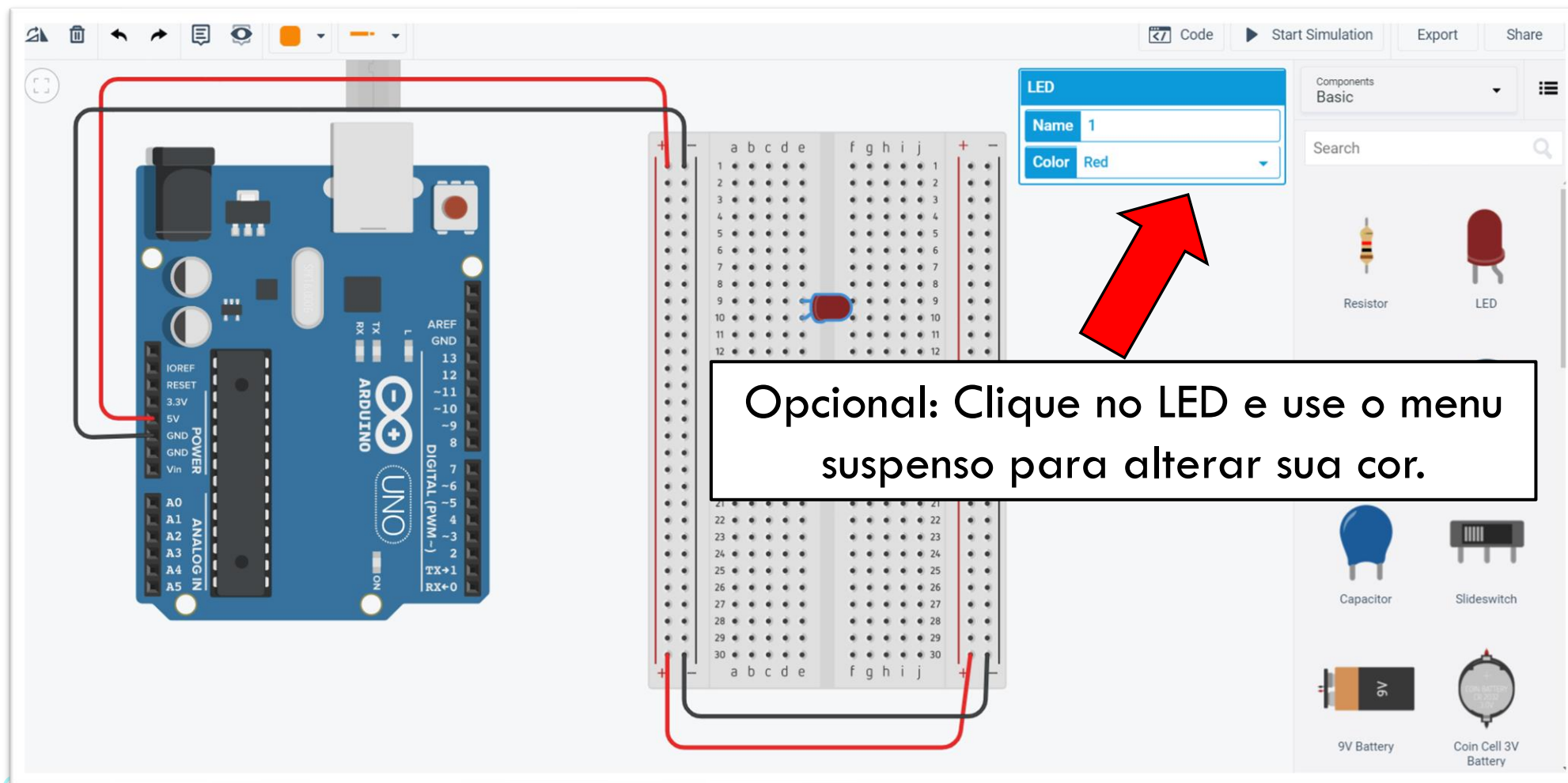
SHOW DE LEDS

TINKERCAD



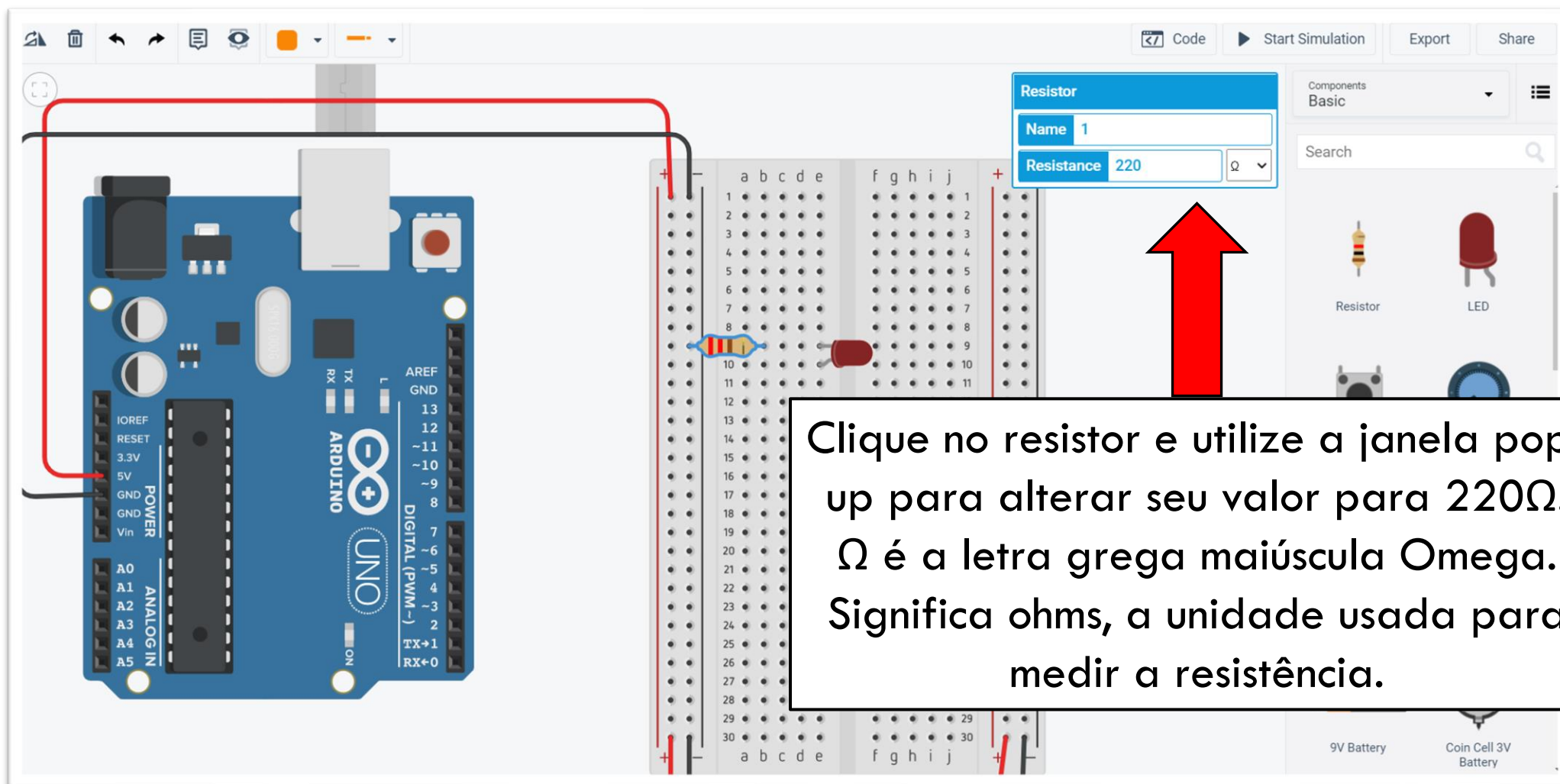
SHOW DE LEDS

TINKERCAD



SHOW DE LEDS

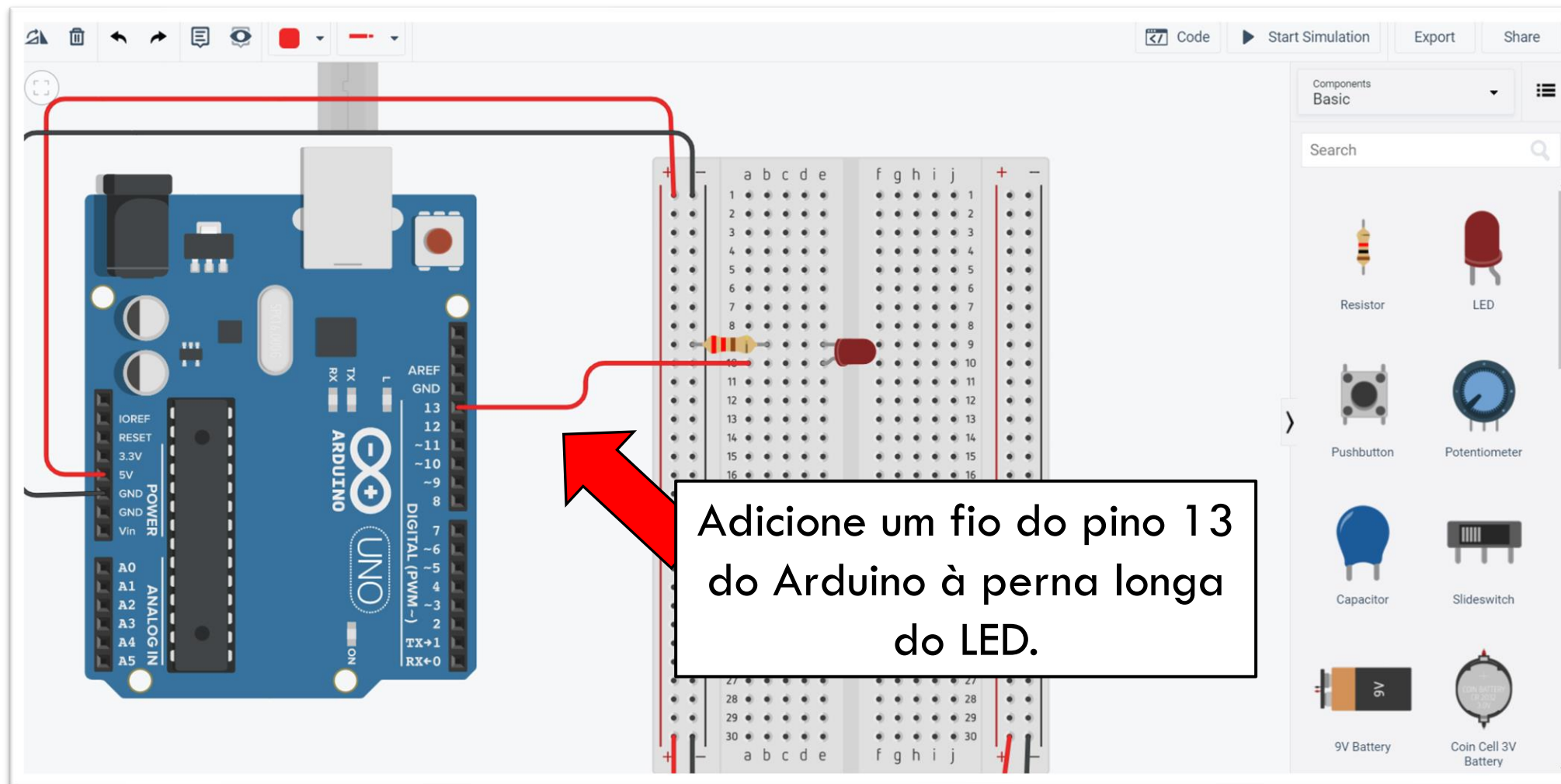
TINKERCAD



Clique no resistor e utilize a janela pop-up para alterar seu valor para 220Ω. Ω é a letra grega maiúscula Omega. Significa ohms, a unidade usada para medir a resistência.

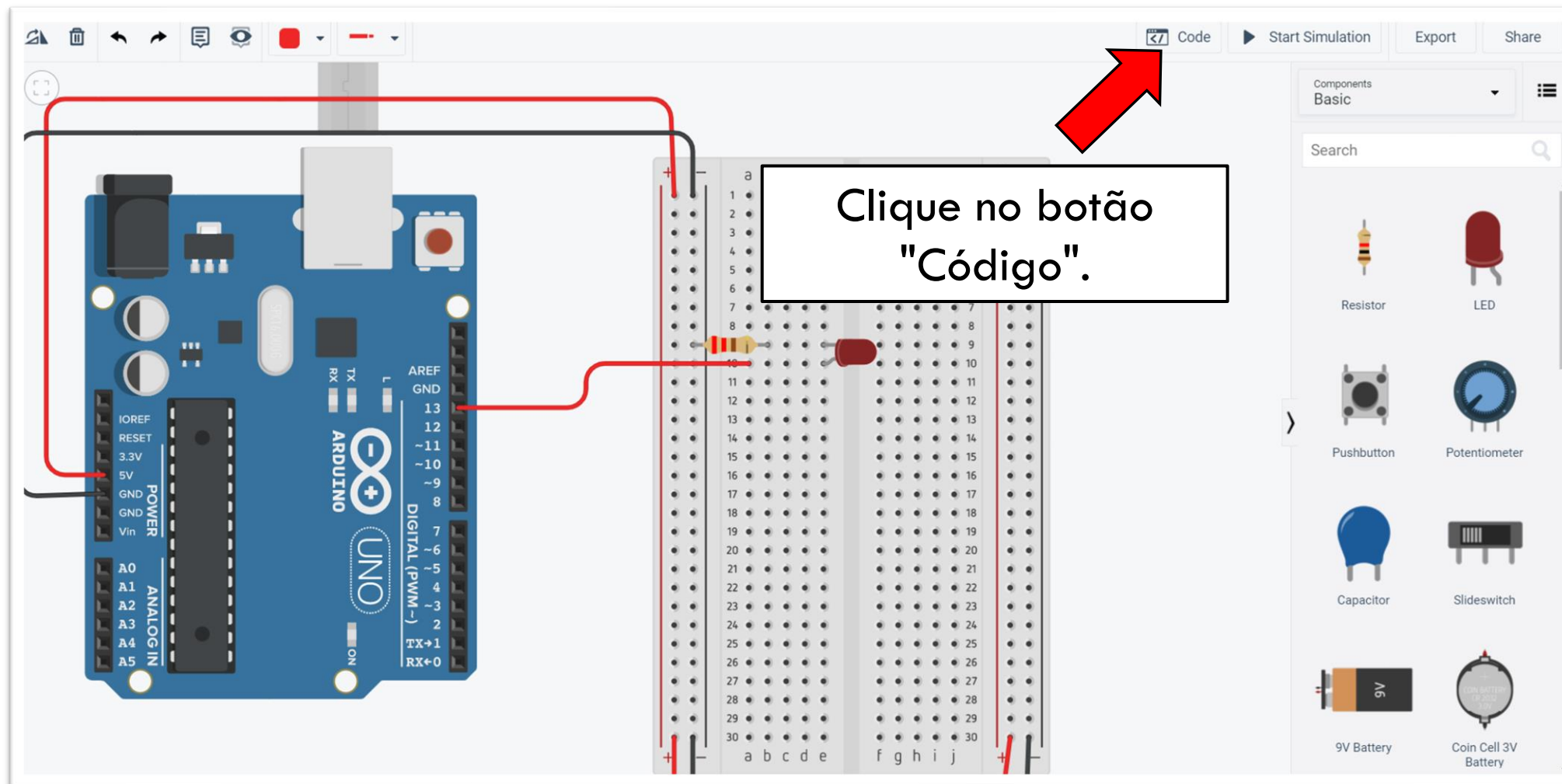
SHOW DE LEDS

TINKERCAD



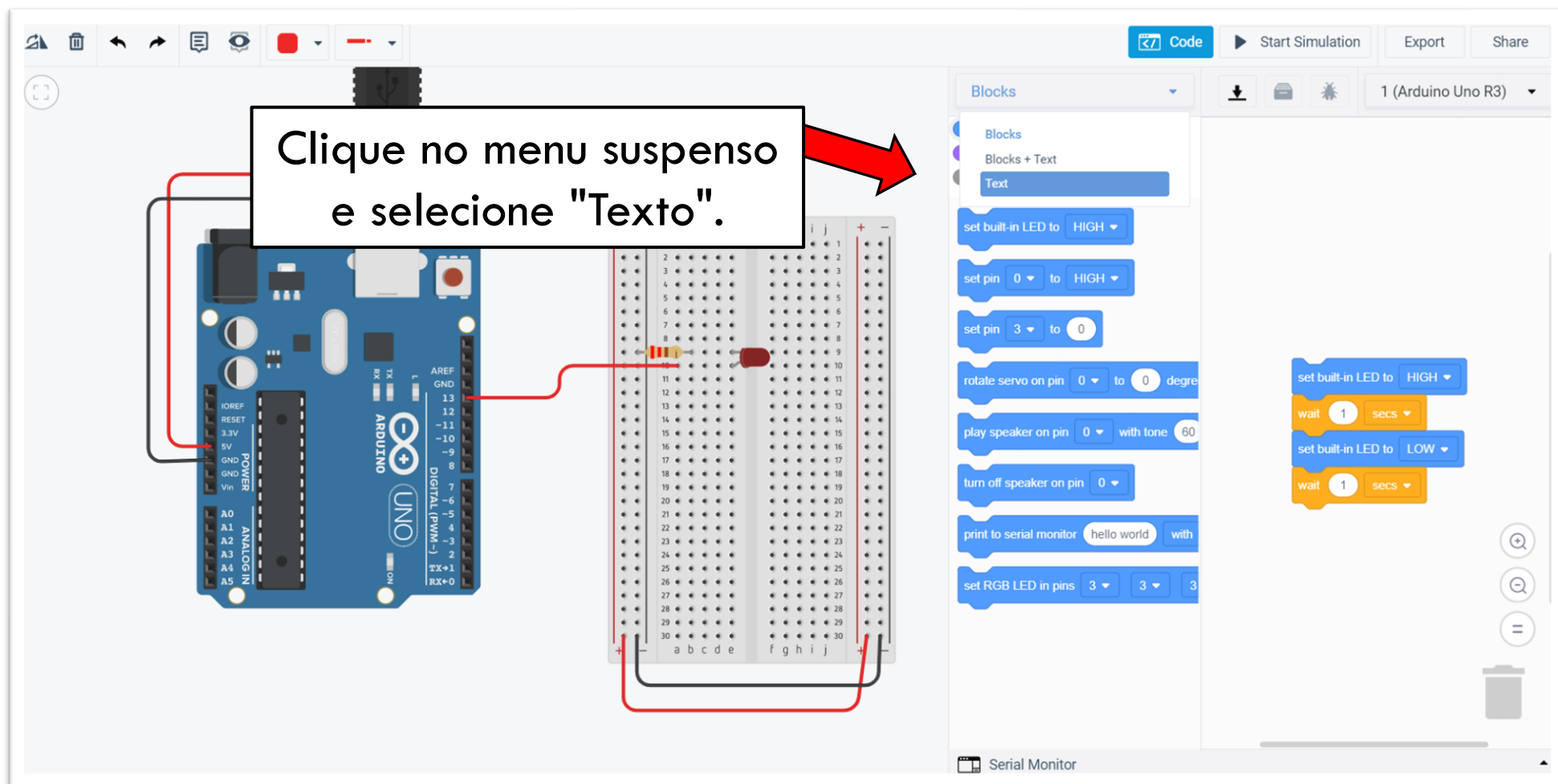
SHOW DE LEDS

TINKERCAD



SHOW DE LEDS

TINKERCAD



SHOW DE LEDS

TINKERCAD

O código na função de configuração só é executado uma vez. Ele usa o comando `pinMode` para dizer ao Arduino para usar o pino 13 como saída.

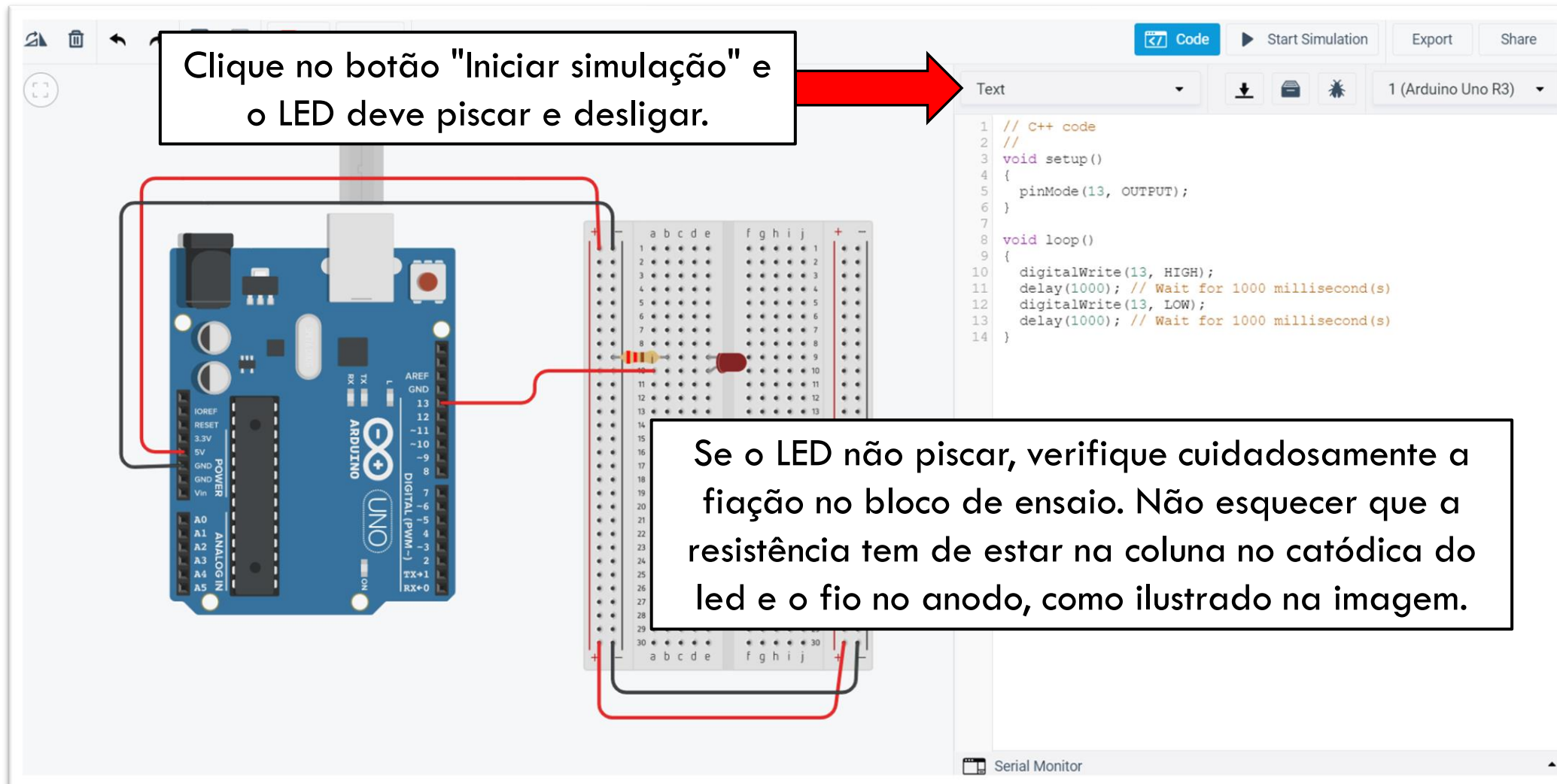
O código na função `loop()` é executado repetidamente para sempre. Ele usa o comando `digitalWrite` para ligar ou desligar o LED (alto ou baixo) e o comando `delay` para fazer o código aguardar por um determinado período de tempo. Este código diz ao Arduino para ligar o LED, esperar um segundo (1000 milissegundos), desligar o LED, esperar um segundo e repetir.

```
1 // C++ code
2 //
3 void setup()
4 {
5   pinMode(13, OUTPUT);
6 }
7
8 void loop()
9 {
10  digitalWrite(13, HIGH);
11  delay(1000); // Wait for 1000 millisecond(s)
12  digitalWrite(13, LOW);
13  delay(1000); // Wait for 1000 millisecond(s)
14 }
```

SHOW DE LEDS

TINKERCAD

Clique no botão "Iniciar simulação" e o LED deve piscar e desligar.



```
// C++ code
//
void setup()
{
  pinMode(13, OUTPUT);
}

void loop()
{
  digitalWrite(13, HIGH);
  delay(1000); // Wait for 1000 millisecond(s)
  digitalWrite(13, LOW);
  delay(1000); // Wait for 1000 millisecond(s)
}
```

Se o LED não piscar, verifique cuidadosamente a fiação no bloco de ensaio. Não esquecer que a resistência tem de estar na coluna no catódica do led e o fio no anodo, como ilustrado na imagem.

SHOW DE LEDS

TINKERCAD

Com os conhecimentos adquiridos, programe o seu próprio show de luzes LED.

Pode conectar até 14 LEDs ao seu Arduino (observe o exemplo, imagem seguinte, como os pinos começam a contar em 0).

Lembre-se:

Use o comando `pinMode` para definir pinos como saídas.

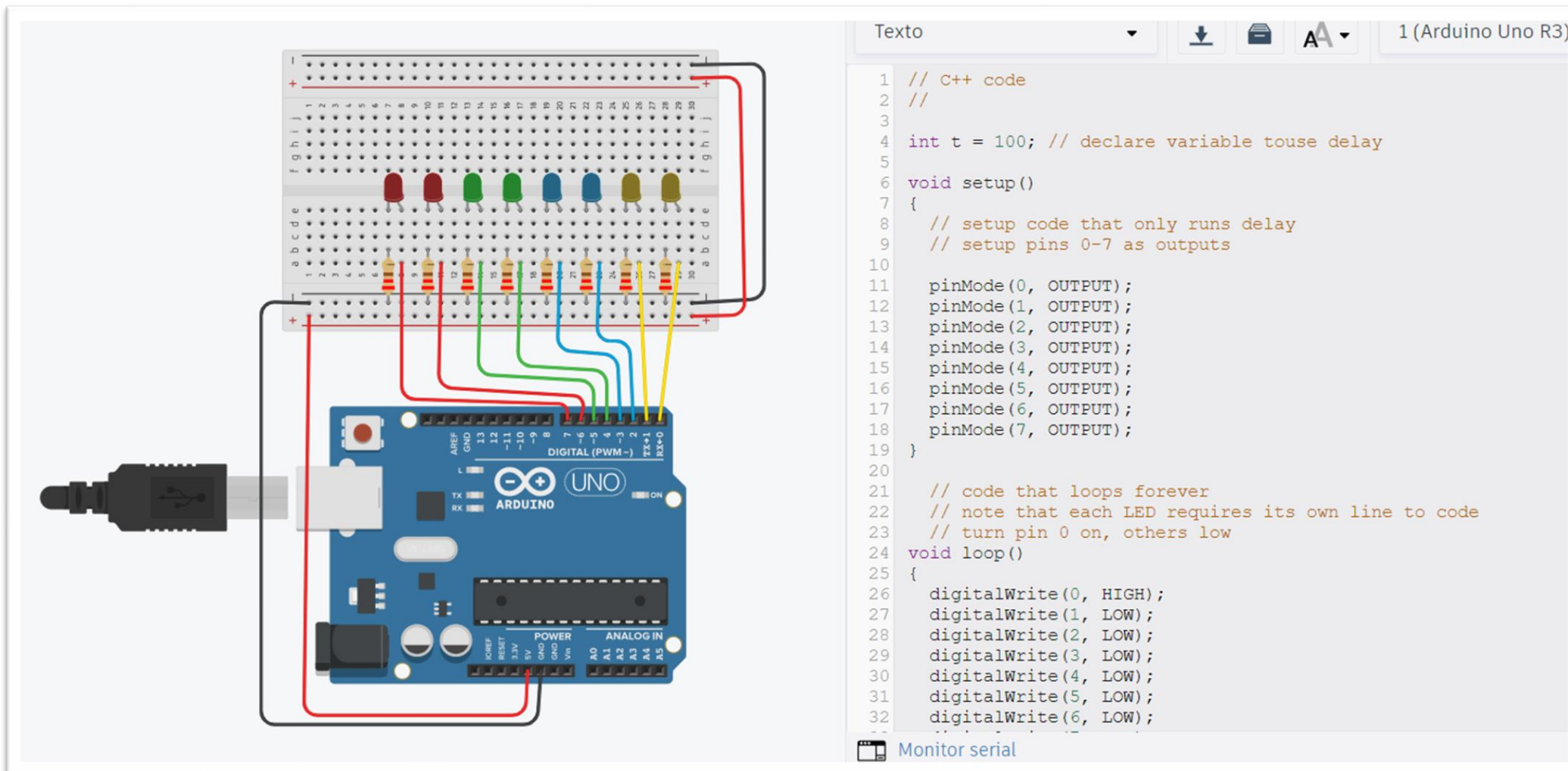
Use o comando `digitalWrite` para ativar e desativar os pinos.

Use o comando `delay` para fazer o programa esperar.

Você pode usar variáveis para facilitar o ajuste dos tempos de atraso.

SHOW DE LEDS

TINKERCAD



The screenshot displays the Tinkercad environment. On the left, an Arduino Uno R3 is connected to a breadboard. Eight LEDs are connected to the breadboard: two red, two green, two blue, and two yellow. The breadboard is wired to the Arduino's digital pins. On the right, the code editor shows the following C++ code:

```
1 // C++ code
2 //
3
4 int t = 100; // declare variable touse delay
5
6 void setup()
7 {
8   // setup code that only runs delay
9   // setup pins 0-7 as outputs
10
11   pinMode(0, OUTPUT);
12   pinMode(1, OUTPUT);
13   pinMode(2, OUTPUT);
14   pinMode(3, OUTPUT);
15   pinMode(4, OUTPUT);
16   pinMode(5, OUTPUT);
17   pinMode(6, OUTPUT);
18   pinMode(7, OUTPUT);
19 }
20
21 // code that loops forever
22 // note that each LED requires its own line to code
23 // turn pin 0 on, others low
24 void loop()
25 {
26   digitalWrite(0, HIGH);
27   digitalWrite(1, LOW);
28   digitalWrite(2, LOW);
29   digitalWrite(3, LOW);
30   digitalWrite(4, LOW);
31   digitalWrite(5, LOW);
32   digitalWrite(6, LOW);
```

Below the code editor, there is a "Monitor serial" button.

SHOW DE LEDS

TINKERCAD

```
33 digitalWrite(7, LOW);
34 delay(t); // Wait for t millisecond(s)
35
36 digitalWrite(0, LOW);
37 digitalWrite(1, HIGH);
38 digitalWrite(2, LOW);
39 digitalWrite(3, LOW);
40 digitalWrite(4, LOW);
41 digitalWrite(5, LOW);
42 digitalWrite(6, LOW);
43 digitalWrite(7, LOW);
44 delay(t); // Wait for t millisecond(s)
45
46 digitalWrite(0, LOW);
47 digitalWrite(1, LOW);
48 digitalWrite(2, HIGH);
49 digitalWrite(3, LOW);
50 digitalWrite(4, LOW);
51 digitalWrite(5, LOW);
52 digitalWrite(6, LOW);
53 digitalWrite(7, LOW);
54 delay(t); // Wait for t millisecond(s));
55
56 digitalWrite(0, LOW);
57 digitalWrite(1, LOW);
58 digitalWrite(2, LOW);
59 digitalWrite(3, HIGH);
60 digitalWrite(4, LOW);
61 digitalWrite(5, LOW);
62 digitalWrite(6, LOW);
63 digitalWrite(7, LOW);
```



E assim, sucessivamente.

SÍNTESE

- Implementar um circuito de Show de LEDs no Tinkercad.

SUGESTÕES

- Acrescente som ao um circuito de Show de LEDs.
- Implemente um novo circuito para produzir um teclado com som.
- Criar um parque de estacionamento.

Esta proposta educativa foi traduzida e/ou adaptada do projeto
[Arduíno Education](#)

Link de projetos no TinKerCAD

<https://www.tinkercad.com/projects/Building-a-Simple-Electronic-Piano-Using-Tinkercad>