



# Manual de Iniciação



Região Autónoma  
da Madeira  
Governo Regional

Secretaria Regional  
**de Educação**  
Direção Regional de Educação



Ambientes Inovadores  
de Aprendizagem



# O que é o Arduino?

O Arduino é uma plataforma de programação de código aberto projetada para tornar a experiência com componentes eletrônicos divertida e intuitiva. O Arduino tem uma linguagem de programação própria e simplificada e uma vasta rede de suporte e milhares de potenciais usos, tornando-se a plataforma perfeita para entusiastas iniciantes e avançados.

## É um computador para todos

Podemos pensar no Arduino como o filho dos computadores tradicionais. O Arduino é essencialmente um pequeno computador. É capaz de receber entradas (como o apertar de um botão ou a leitura de um sensor de luz) e interpretar essas informações para controlar várias saídas (colocar uma luz LED a piscar ou ligar um motor elétrico).

O Arduino Uno é uma placa eletrônica baseada no ATmega328. É uma placa muito utilizada por causa da sua extensa rede de suporte e versatilidade. Possui 14 pinos de entrada / saída digital (6 dos quais podem ser saídas PWM), 6 entradas analógicas, uma frequência de funcionamento de 16 MHz, uma ligação USB, um conector de energia, e um botão de reset.

# Guia Rápido de Utilização

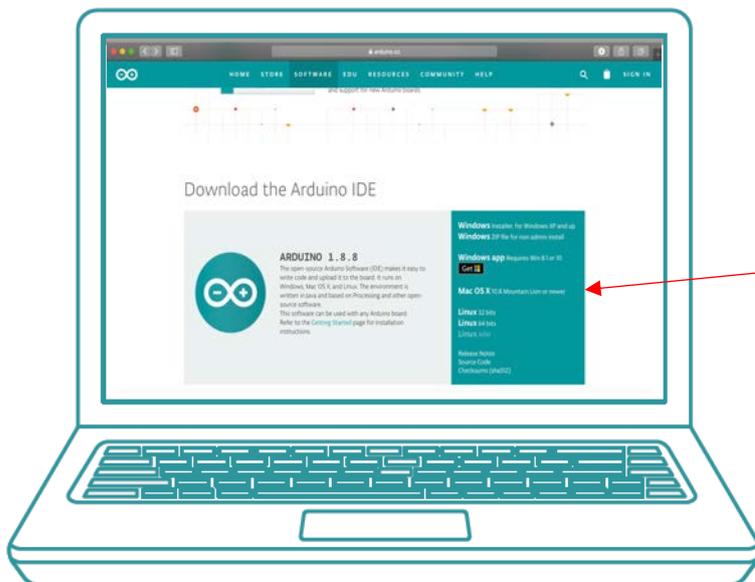
Para colocar a funcionar o seu Arduino, será necessário descarregar alguns programas de [www.arduino.cc](http://www.arduino.cc) (é grátis!). Este software, conhecido como IDE (*Integrated Development Environment*) do Arduino, permite que possa programar o Arduino para fazer o que quiser. É como um processador de texto para escrever programas. Com um computador compatível ligado à Internet, abra o seu navegador favorito e digite o seguinte URL na barra de endereço:

<https://www.arduino.cc/en/Main/Software>



1

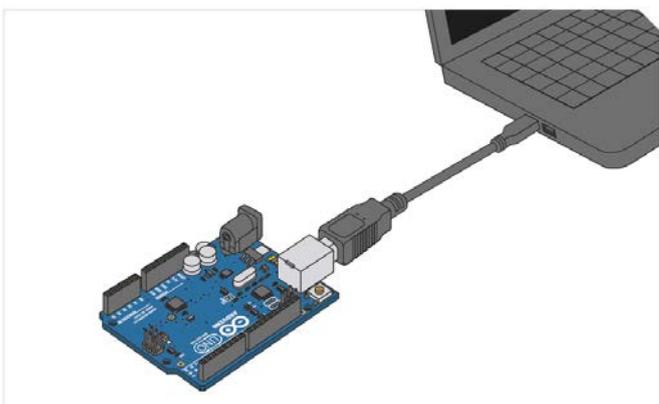
## Descarregar o Arduino IDE



Escolha o pacote de instalação do sistema operativo do seu computador, e faça a instalação do software no seu computador.

2

## Ligue o seu Arduino ao computador



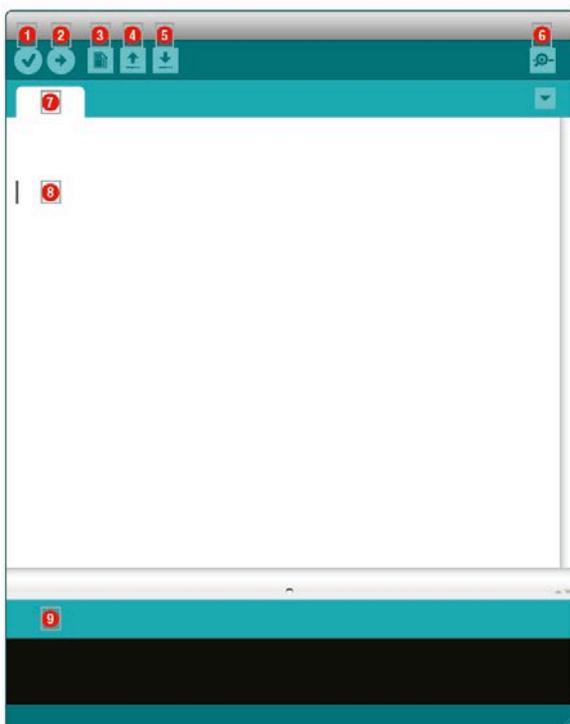
Use o cabo USB fornecido no kit para ligar o Arduino a uma das entradas USB do seu computador.



# 3

## Executar o software Arduino IDE

Abra o software Arduino IDE no seu computador. Dê uma olhadela e conheça a interface. Ainda não vamos programar, isto é apenas a introdução. Esta etapa é para configurar o seu IDE para identificar o seu Arduino Uno.

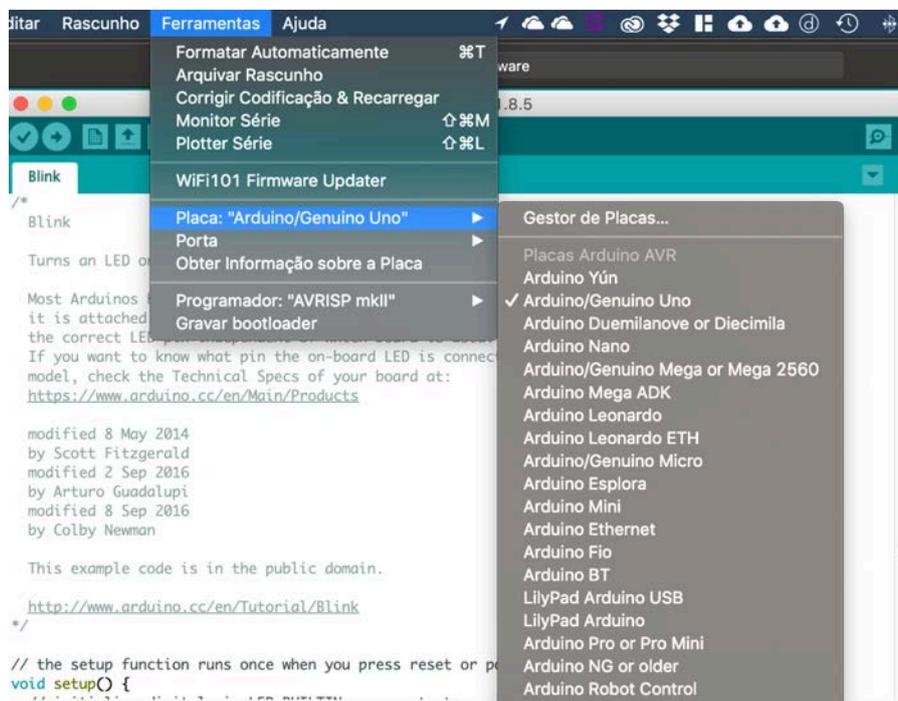


- 1 Verificar: compila e verifica se o seu código contém erros;
- 2 Envio: envia o seu código para a placa Arduino. Se tal ocorrer sem problemas, as luzes da placa piscarão rapidamente.
- 3 Novo: Abre uma nova janela para escrever código.
- 4 Abrir: Abre uma janela de código guardada no seu computador.
- 5 Guardar: guarda o seu código no seu computador.
- 6 Monitor Série: abre uma janela que mostra alguma informação que o Arduino está a transmitir.
- 7 Nome do programa: mostra o nome do código.
- 8 Área do código: é aqui que se escreve o código.
- 9 Área de mensagem: é aqui que o IDE lhe mostra os erros que o seu código possa ter.



## 4

## Selecionar a placa: Arduino Uno e a Porta COM3



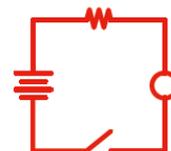
Selecione a porta de entrada da placa Arduino no menu Ferramentas | Porta. É provável que seja a porta **COM 3** ou superior (COM1 e COM2 são geralmente reservados para portas do sistema).



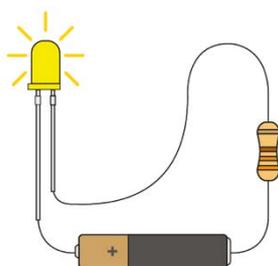
# Entender os circuitos elétricos

## O que é um circuito elétrico?

Um circuito elétrico é basicamente um circuito eletrônico com um ponto de partida e um ponto final - com um número variado de componentes entre eles. Os circuitos podem incluir resistências, díodos, transistores, sensores de todos os tamanhos e formas, motores bem como centenas de milhares de outros componentes.



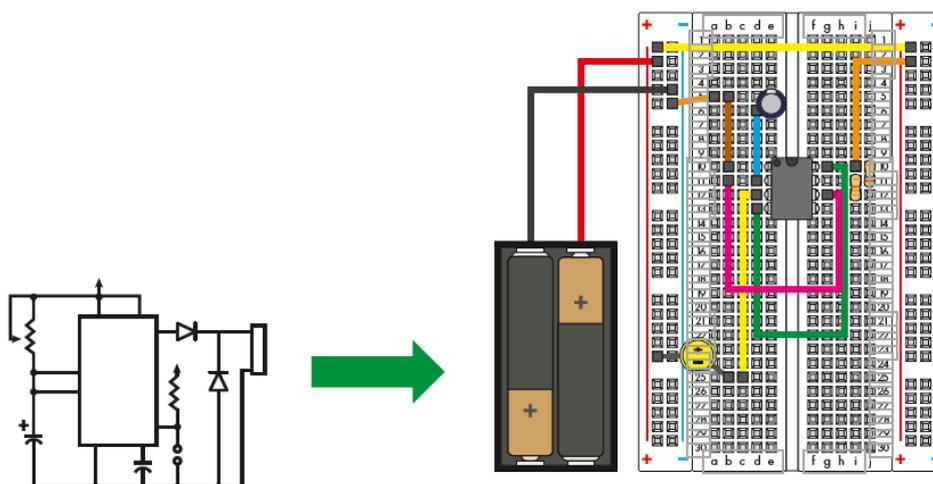
## O mundo está cheio de circuitos



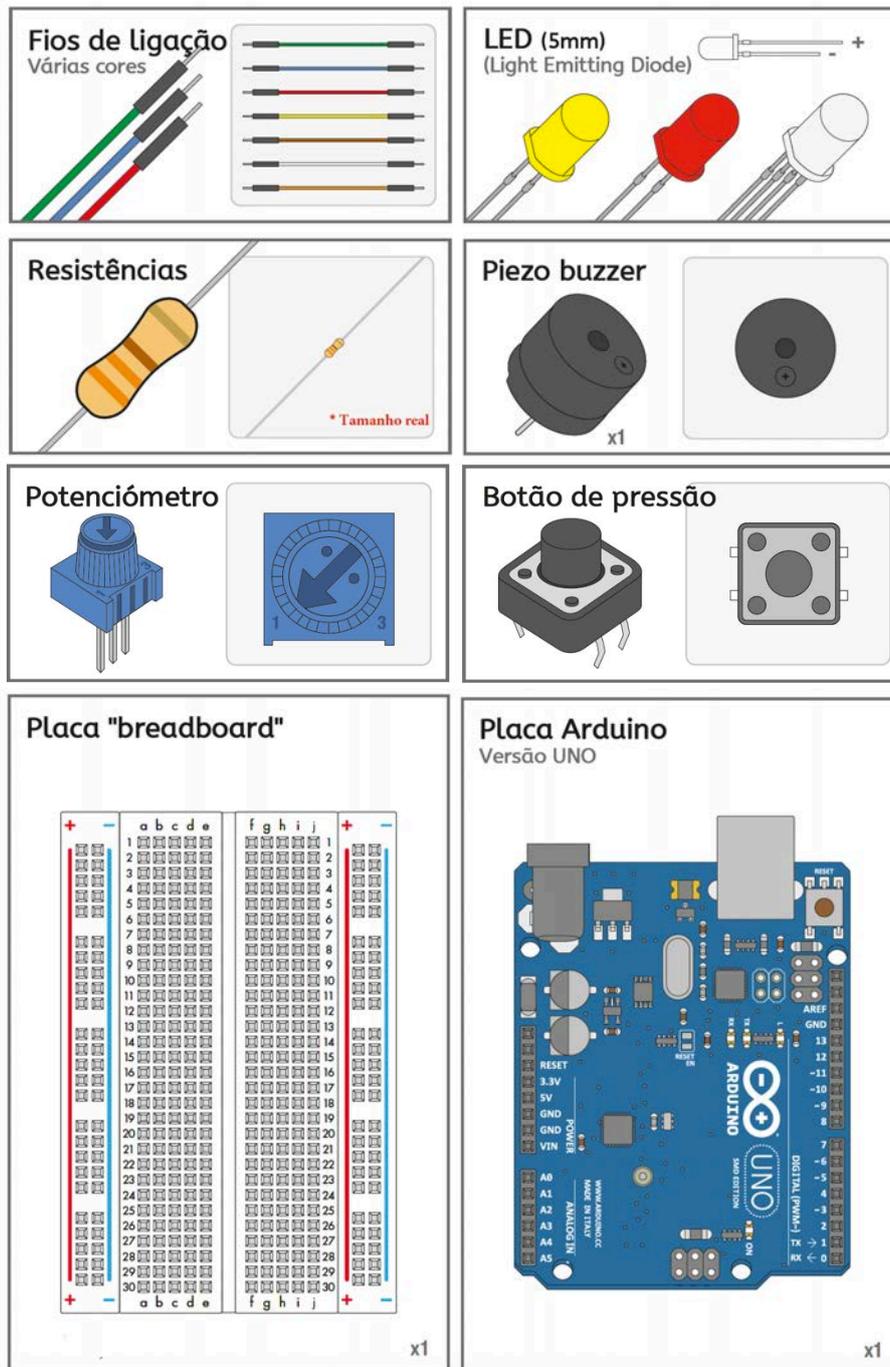
Para onde quer que olhe, encontrará circuitos. O seu telemóvel, o computador que controla o sistema de emissões do seu carro, a consola de videojogos - todos estão cheios de circuitos. Neste guia, poderá montar alguns circuitos simples e aprenderá a essência do mundo da eletrónica.

## Circuitos simples e complexos

Neste guia, poderá montar circuitos simples, mas incríveis!...

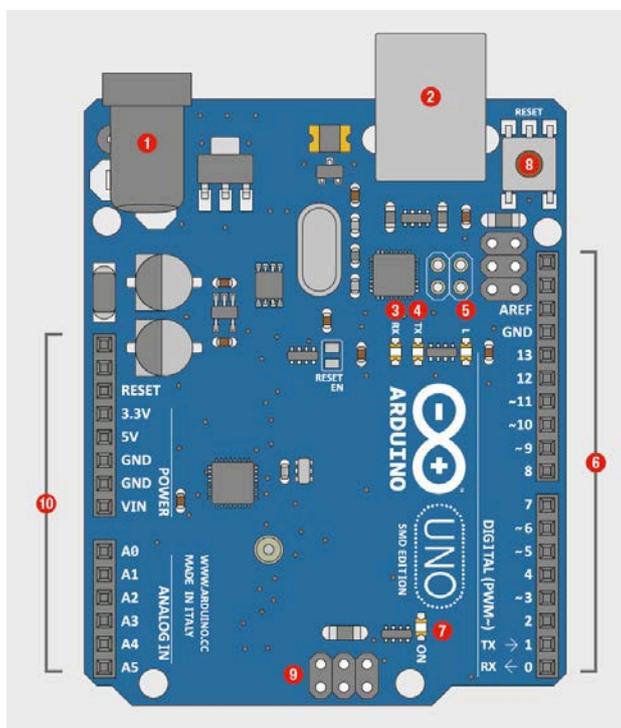


# Os principais componentes eletrônicos





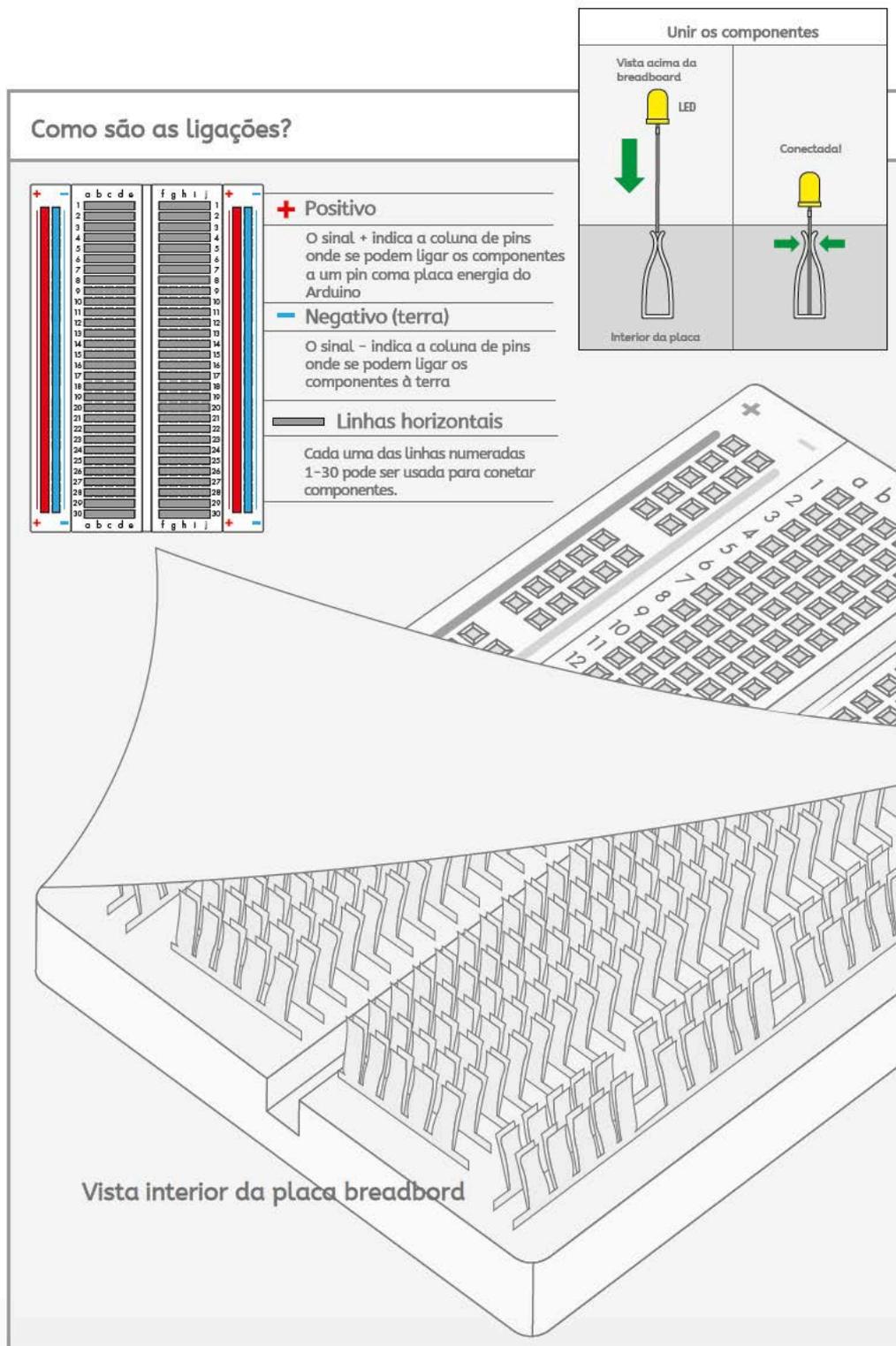
## A placa arduino



- 1 **Entrada de corrente elétrica** – Pode ser usada uma pilha de 9V
- 2 **Entrada USB** – Assegura a energia e quando está ligado ao computador via USB permite troca de dados com o PC.
- 3 **LED (RX: a receber)** – Quando está aceso indica que a placa está a receber dados do computador.
- 4 **LED (TX: a enviar)** – Quando está aceso indica que a placa Arduino está a enviar dados para o PC.
- 5 **LED (Pin 13: verificação de erros)** – Este LED indica que a placa está a executar o programa normalmente
- 6 **Pins (ARef, Ground, Digital, Rx, Tx)** – Estes pins podem ser usados como entradas, saídas, energia e terra.
- 7 **LED (Indica que o Arduino está ligado)**
- 8 **Botão RESET** – Carregar neste botão permite reiniciar manualmente o programa carregado na placa.
- 9 **Pins (Entrada analógica, Energia, Terra, Reset)** – Estes pins podem ser usados como entradas, saídas, energia e terra.



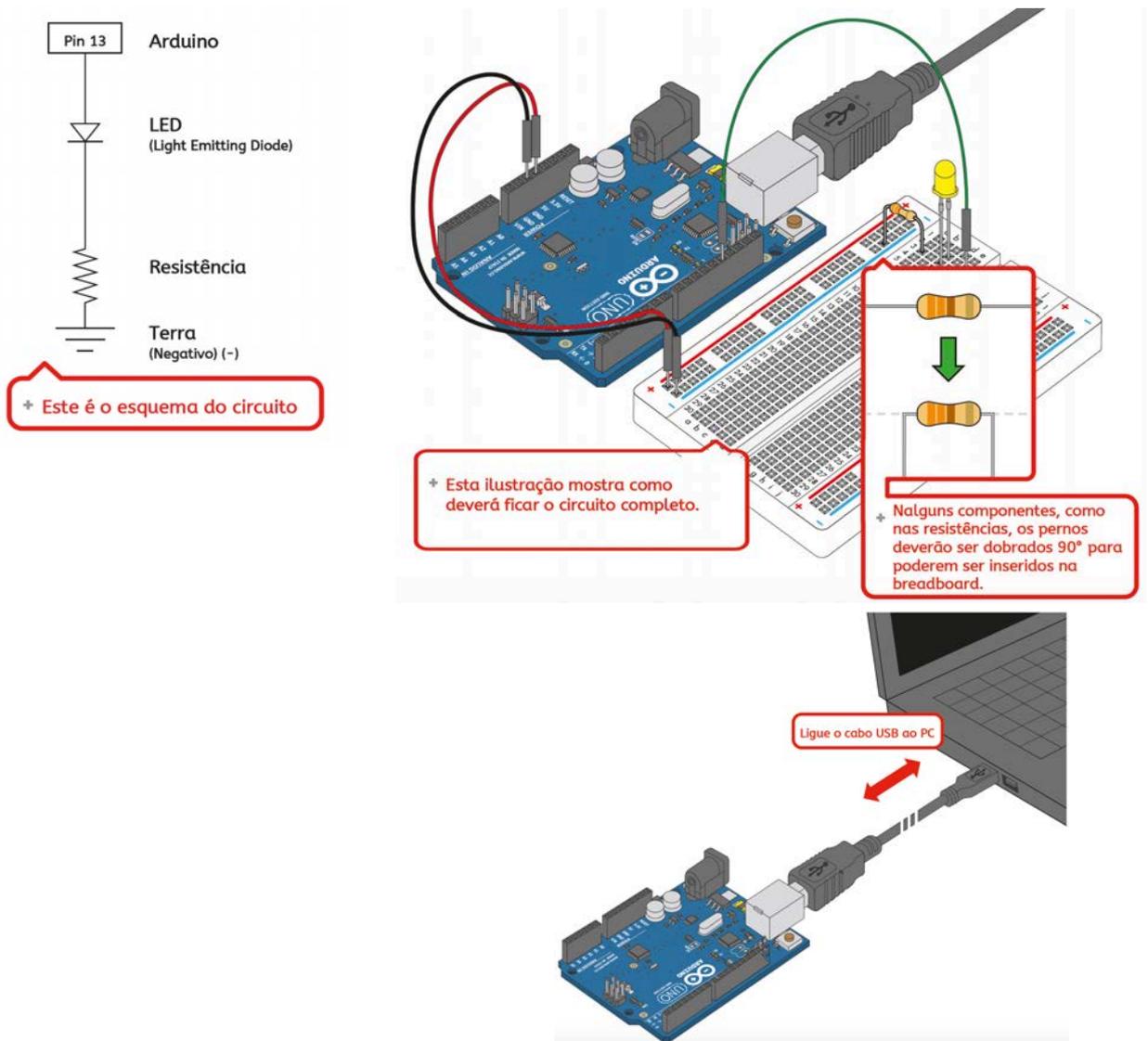
## A placa “breadboard”



# Um projeto simples:

## Piscar um LED

LEDs (díodos emissores de luz) são luzes pequenas e potentes que são utilizadas em muitas aplicações diferentes. Neste primeiro projeto, iremos colocar um LED a piscar. É quase tão simples quanto ligar e desligar uma lâmpada da sala usando um interruptor.



Pin 13    Arduino

LED  
(Light Emitting Diode)

Resistência

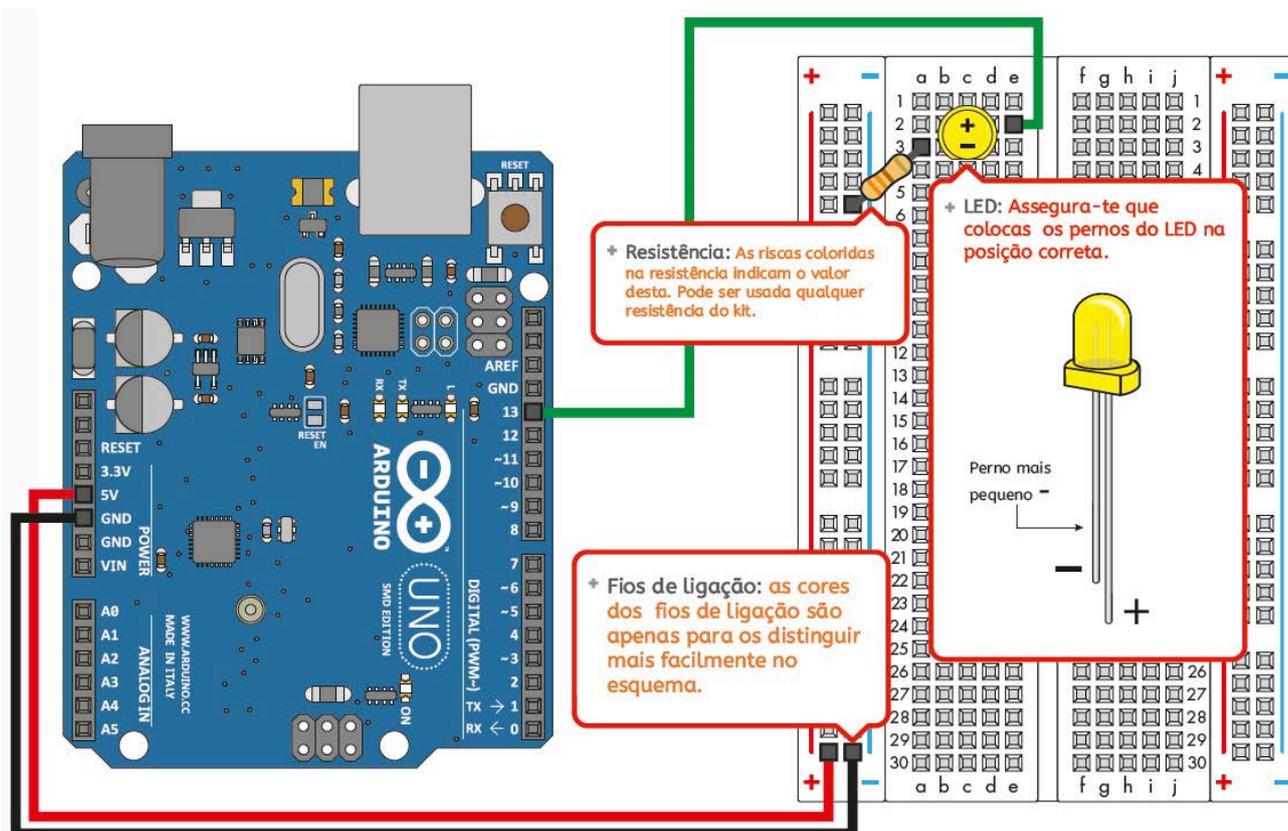
Terra  
(Negativo) (-)

+ Este é o esquema do circuito

+ Esta ilustração mostra como deverá ficar o circuito completo.

+ Nalguns componentes, como nas resistências, os pernos deverão ser dobrados 90° para poderem ser inseridos na breadboard.

Ligue o cabo USB ao PC



## Programar a placa arduino

Na programação de placas arduino existem sempre duas funções: `setup()` e `loop()`.

- `setup()` : Esta parte do código “corre” apenas uma vez no início. É aqui que são identificados os pins da placa que serão usados.
- `loop()` : Nesta parte o código “correrá” vezes sem conta enquanto houver energia elétrica ligada à placa. Quando chegar à última linha do `loop()` volta à primeira e assim sucessivamente.



### Clique em abrir

Inicie o software Arduino IDE no computador. Abra o código 01.Basics/Blink.



```

Blink | Arduino 1.8.5

Blink

modified 8 May 2014
by Scott Fitzgerald
modified 2 Sep 2016
by Arturo Guadalupi
modified 8 Sep 2016
by Colby Newman

This example code is in the public domain.

http://www.arduino.cc/en/Tutorial/Blink
*/

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

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

```

### Entender o código:

Para que o LED pisque é necessário alternar o estado dele entre ligado e desligado.

**pinMode(numero do pin, INPUT ou OUTPUT ou INPUT\_PULLUP):** indica qual é o pin em uso. Neste programa é usado o pin 13 (é o pin que tem o LED inserido na placa)

**digitalWrite(numero do pin; HIGH ou LOW):** indica à placa se deve colocar o pin com 5V (HIGH), que faz acender o LED ou a 0V (LOW) que o desliga.

**delay(tempo) :** para o programa durante o tempo indicado em milissegundos. Neste caso o programa para 1s quando o LED se acende e para 1s quando o LED se desliga.



**Verificar**

**Clique** para verificar se existe algum erro no código e para o computador compilar o código. O software irá transformar o texto em instruções que a placa Arduino possa entender.

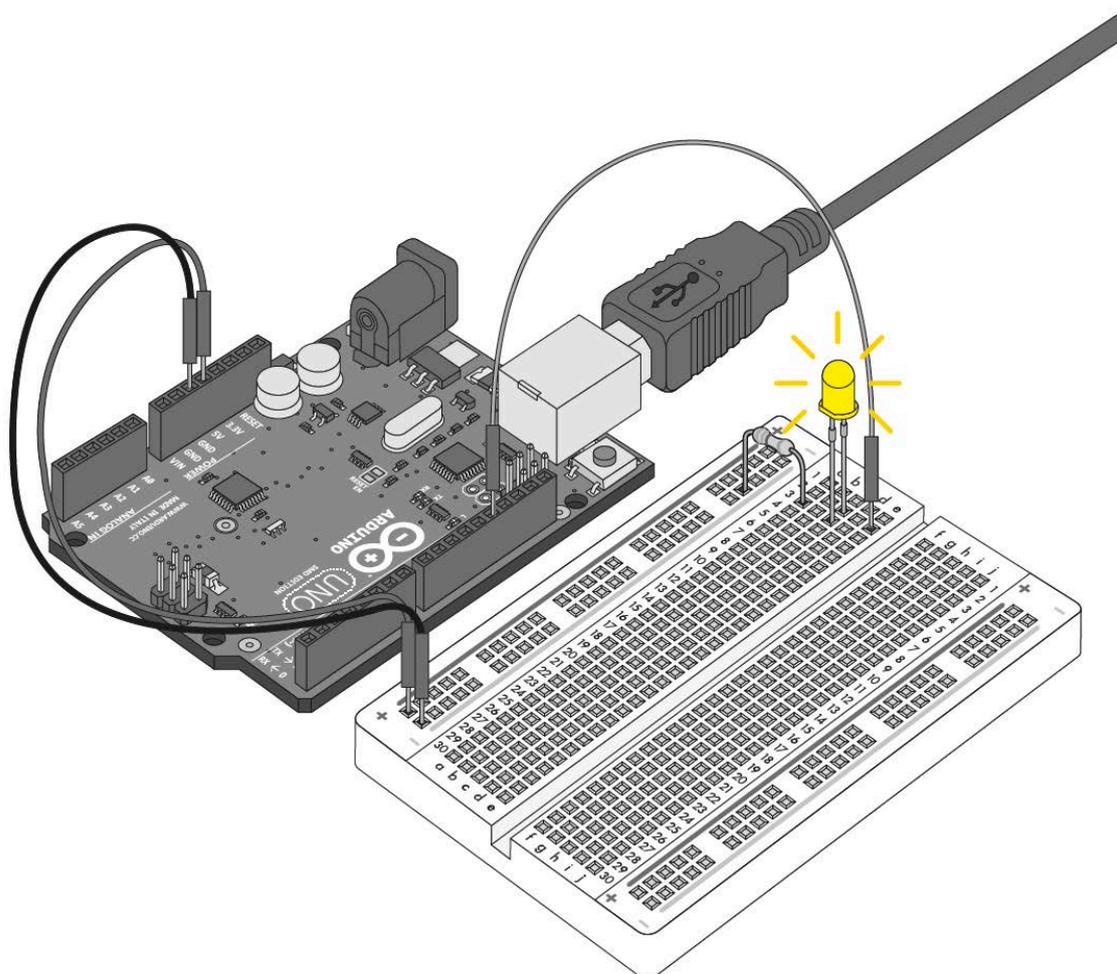


**Enviar**

**Clique** para enviar as instruções para a placa Arduino via cabo USB. O arduino irá começar a "correr" o código automaticamente.



Este é o resultado final do projeto.



## Fontes:

<https://www.arduino.cc> – página oficial Arduino onde se pode encontrar um vasto leque de recursos e uma comunidade de entusiastas pronta a ajudar.

<https://learn.sparkfun.com>