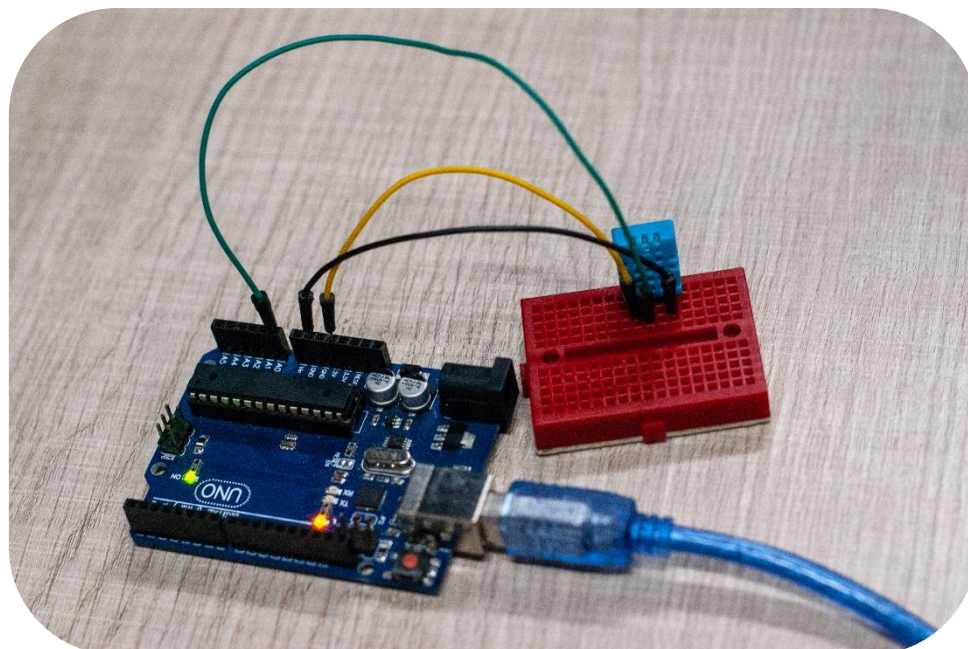
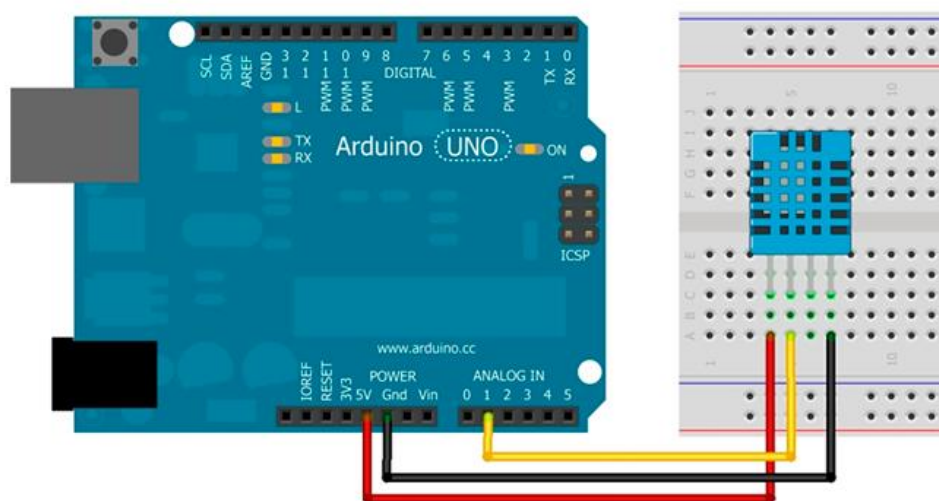




## Este é o Manual de Instalação do sensor DHT11 no Arduino

Neste manual são utilizados os materiais: 1 Arduino, 1 Protoboard, 1 cabo de dados, 1 sensor DHT11 e 3 cabos de energia.

Para fazer a instalação do DHT11, é necessário primeiramente entender seu funcionamento. O sensor DHT11 tem um funcionamento muito simples, basicamente ele é composto por um “corpo” e quatro “pernas”, este corpo é basicamente o sensor, e as pernas são as conexões de dados e energia do sensor. A perna esquerda de frente para o sensor (Parte com furos), é a entrada de energia, onde se utilizam 5 volts (5V), já a última perna à direita, é a saída de energia, que fica conectado na entrada terra (GND) do Arduino e a segunda perna à esquerda, é a saída de dados do sensor para o Arduino (A0 – A5). A terceira perna da esquerda para a direita não é utilizada.



## Como visualizar a temperatura com o sensor DHT11

Para visualizar o funcionamento do sensor e do Arduino após a montagem, é necessário você utilizar o Arduino IDE, IDE oficial do Arduino, que utiliza C++. Nesta IDE você precisará escrever um código que irá dizer ao Arduino o que fazer. Para este sensor, temos como exemplo o código abaixo, que fará a leitura dos dados de temperatura e umidade do sensor e os mostrará na tela convertidos em graus e em porcentagem relativa.

```
// Aqui nós importamos o módulo do DHT
#include "DHT.h"
#define dht_type DHT11
// Aqui nós definimos para o Arduino o sensor DHT, na porta 0
int dht_pin = A0;
DHT dht_1 = DHT(dht_pin, dht_type);

// Declaração da velocidade de transferência de dados da porta USB
void setup() {
  Serial.begin(9600);
  dht_1.begin();
}

// Algoritmo que fará a leitura de dados do sensor e mostrará na tela
void loop() {
  float umidade = dht_1.readHumidity();
  float temperatura = dht_1.readTemperature();
  if (isnan(temperatura) or isnan(umidade)) {
    Serial.println("Erro ao ler o DHT");
  } else {
    Serial.print(umidade);
    Serial.print(";");
    Serial.println(temperatura);
  }
  delay(2000);
}
```

