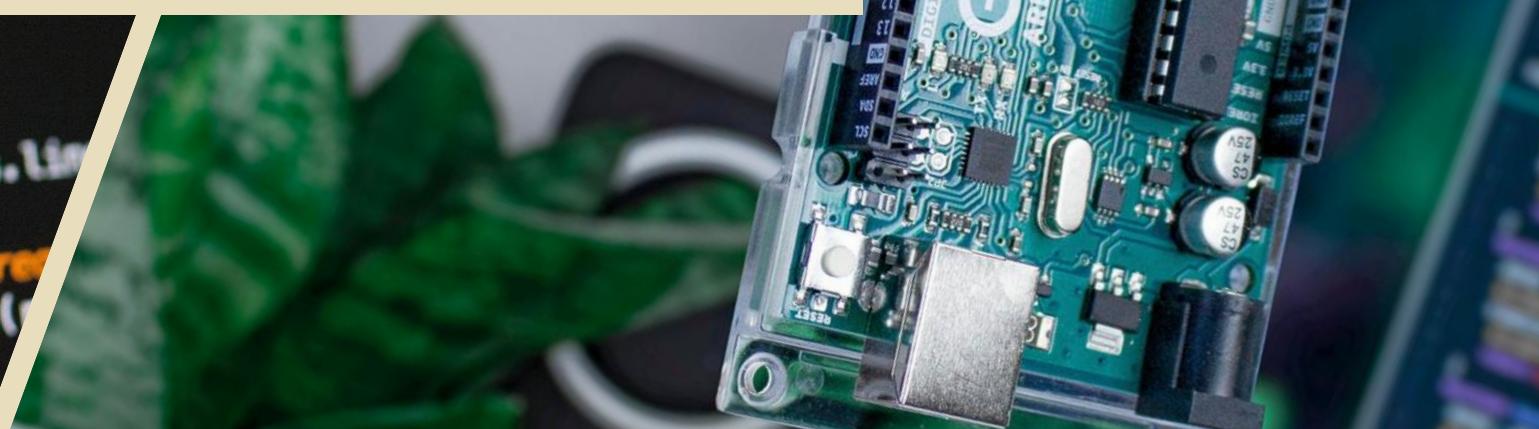


```
31 dev_ self.file = None
32 self.fingerprints = set()
33 self.logduplicates = True
34 self.debug = debug
35 self.logger = logging.getLogger(__name__)
36 if path:
37     self.file = open(os.path.join(path, 'fingerprints.txt'), 'a')
38     self.file.seek(0)
39     self.fingerprints.update(self._load_fingerprints())
40
41 @classmethod
42 def from_settings(cls, settings):
43     debug = settings.getboolean('debug', False)
44     return cls(debug=debug)
45
46 def request_setting(self, setting):
47     fp = self.fingerprints.get(setting)
48     if fp is not None:
49         return True
50     self.fingerprints.add(fp)
51     if self.file:
52         self.file.write(fp + os.linesep)
53
54     def request_fingerprint(self, setting):
55         return request_fingerprint(self, setting)
```

# PYTHON PARA ARDUINO



# WHOAMI



## MATEUS ANTONIO DA SILVA

ESTUDANTE DE ENGENHARIA DA COMPUTAÇÃO - UFPB

BOLSISTA NO LABORATÓRIO DE ENGENHARIA DE SISTEMAS E  
ROBÓTICA (LASER)

QUER APRENDER SOBRE ARDUINO, PROGRAMAÇÃO E ROBÓTICA  
NO GERAL? FALA COMIGO!

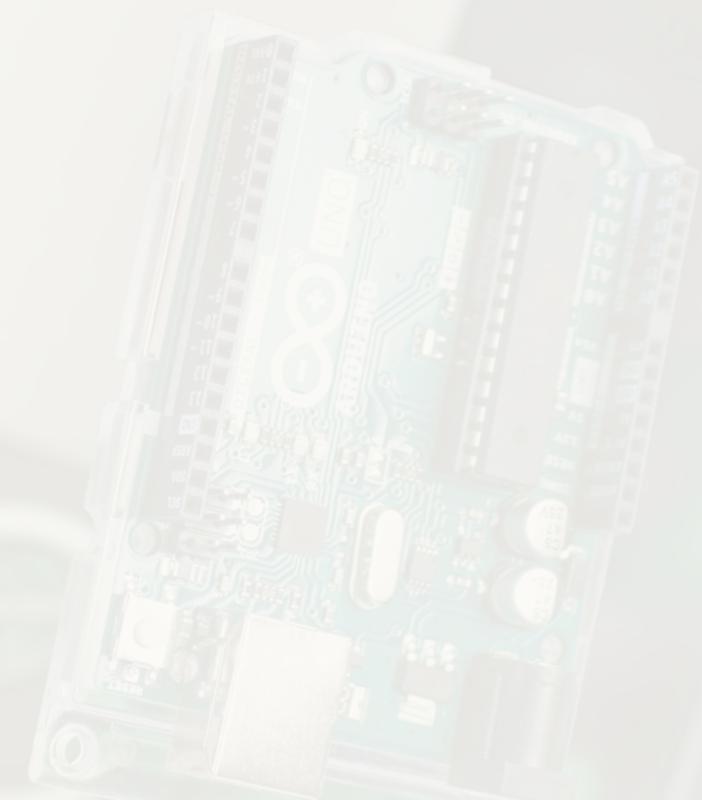
ENTUSIASSTA DA CULTURA MAKER

INTERESSADO EM ROBÓTICA, CULTURA OPEN-SOURCE,  
AUTOMAÇÃO, IOT E EM DISSEMINAR CONHECIMENTO!



# SUMÁRIO

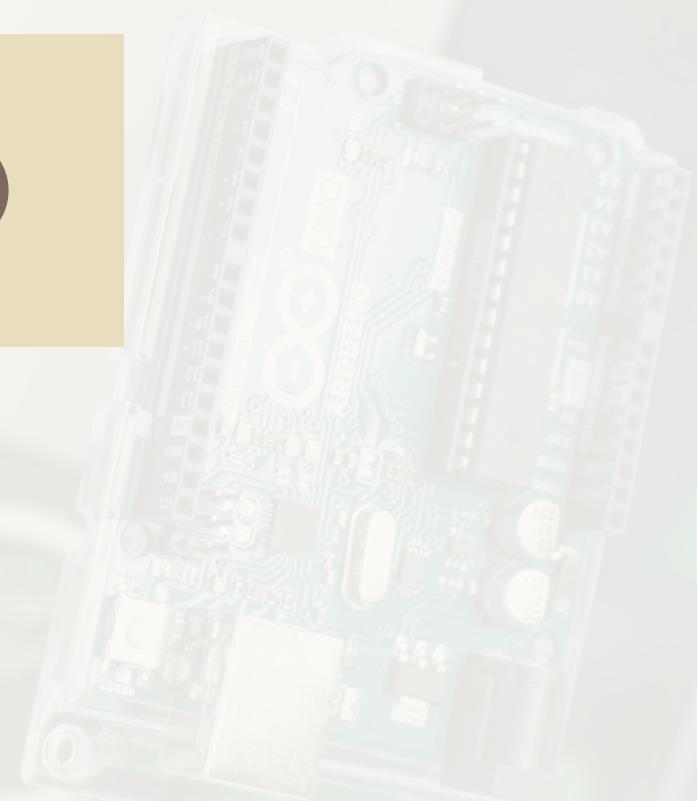
- INTRODUÇÃO
- OBJETIVO
- SOBRE ARDUINO
- SOBRE PYTHON
- APLICAÇÕES



```
31     self.file = None
32     self.fingerprints = set()
33     self.logdupes = True
34     self.debug = debug
35     self.logger = logger
36
37     if path:
38         self.file = open(path, "w")
39         self.file.seek(0)
40         self.fingerprints = set(self.file.read().split())
41
42     @classmethod
43     def from_settings(debug):
44         return cls(debug=debug)
45
46     def request_fp(self, fp):
47         fp = self._normalize_fp(fp)
48         if fp in self.fingerprints:
49             return True
50         self.fingerprints.add(fp)
51         if self.file:
52             self.file.write(fp + os.linesep)
53
54     def request_fingerprint(self, fp):
55         return request_fp(self, fp)
```



# INTRODUÇÃO





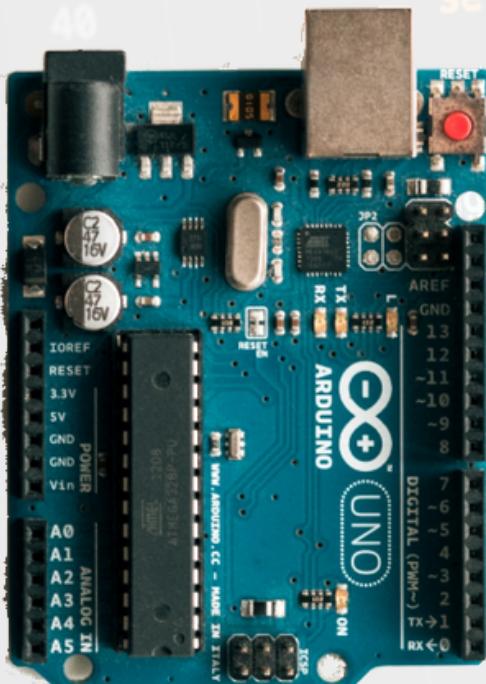
@MATTEUS\_ANTONIO



GITHUB.COM/MATEUSTOIN



MATEUSASILVA3@GMAIL.COM



```
def __init__(self, path):
    self.file = None
    self.fingerprints = set()
    self.logdups = True
    self.debug = debug
    self.logger = logging.getLogger(__name__)
    if path:
        self.file = open(os.path.join(path, 'fingerprint.log'), 'a')
        self.file.seek(0)
        self.fingerprints.update(fp + os.linesep)
        for fp in self.fingerprints:
            self.file.write(fp + os.linesep)

@classmethod
def from_settings(cls, settings):
    debug = settings.getbool('SUPERVISOR_DEBUG')
    return cls(job_dir(settings), debug)

def request_seen(self, request):
    fp = self.request_fingerprint(request)
    if fp in self.fingerprints:
        return True
    self.fingerprints.add(fp)
    if self.file:
        self.file.write(fp + os.linesep)

def request_fingerprint(self, request):
    return request_fingerprint(request)
```

# EXPECTATIVA





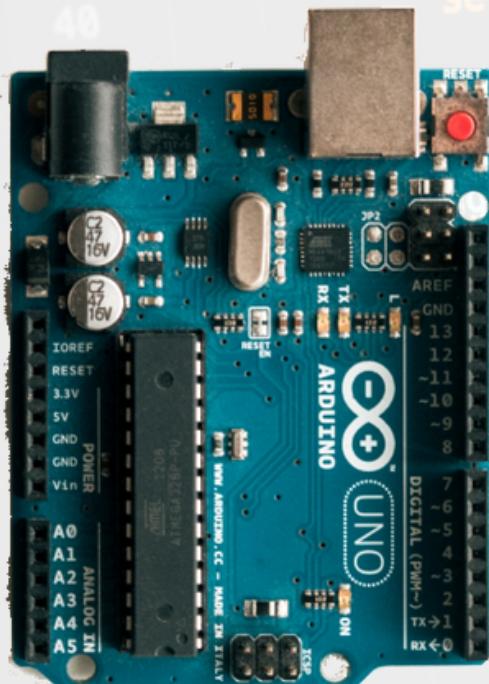
@MATTEUS\_ANTONIO



GITHUB.COM/MATEUSTOIN



MATEUSASILVA3@GMAIL.COM



```
def __init__(self, path):
    self.file = None
    self.fingerprints = set()
    self.logdupes = False
    self.debug = debug
    self.logger = logging.getLogger(__name__)
    if path:
        self.file = open(os.path.join(path, 'fingerprints.csv'), 'a')
        self.file.seek(0)
        self.fingerprints.update(line.strip() for line in self.file)

    @classmethod
    def from_settings(cls, settings):
        debug = settings.getbool("SUPERVISOR_DEBUG")
        return cls(job_dir(settings), debug)

    def request_seen(self, request):
        fp = self.request_fingerprint(request)
        if fp in self.fingerprints:
            return True
        self.fingerprints.add(fp)
        if self.file:
            self.file.write(fp + os.linesep)

    def request_fingerprint(self, request):
        return request_fingerprint(request)
```

# REALIDADE





@MATTEUS\_ANTONIO



GITHUB.COM/MATEUSTOIN



MATEUSASILVA3@GMAIL.COM



Curso Python #01 - Seja um Programador

Curso em Vídeo

YouTube - 3 de abr. de 2017

**Python para Iniciantes**  
Para que não sabe Python, é aqui que deve começar!  
Tiago Miguel  
4,1 ★★★★☆ (11.623)  
1 total hora • 12 aulas • Iniciante

**Introdução à linguagem Python**  
Aprenda a desenvolver programas de computador usando a linguagem mais popular do mercado de trabalho  
Diego Mariano, Ph.D.  
4,4 ★★★★☆ (18.675)  
2 total horas • 38 aulas • Iniciante

**Python 3 na Prática**  
Aprenda a programar em Python  
João Batista  
4,2 ★★★★☆ (515)  
2,5 total horas • 26 aulas • Iniciante

**Introdução a linguagem de programação python**  
Python para iniciantes  
Abraão Passos de Oliveira  
4,3 ★★★★☆ (320)  
7 total horas • 25 aulas • Iniciante

coursera

Udemy

YouTube



@MATTEUS\_ANTONIO



GITHUB.COM/MATEUSTOIN



MATEUSASILVA3@GMAIL.COM



```
31  
32  
33  
34  
35  
36     self.log.setLevel(logging.INFO)  
37     self.debug = debug  
38     self.logger = logger  
39  
40     if path:  
41         self.file = open(path, "w")  
42         self.file.write("")  
43         self.fingerprints = {}  
44  
45
```

# FACILIDADE DO PYTHON

```
from djitellopy import Tello # Importa biblioteca  
  
tello = Tello()           # Instancia do objeto  
  
tello.connect()          # Conecta ao Tello  
tello.takeoff()          # Faz o Tello decolar  
  
tello.move_left(100)      # Move o Tello 100 cm para a esquerda  
tello.rotate_counter_clockwise(45) # Gira o Tello 45º sentido anti horário  
  
tello.land()              # Pousa o Tello  
tello.end()                # Encerra conexão
```



```
31 dev = True
32 self.file = None
33 self.fingerprints = set()
34 self.logdupes = False
35 self.debug = debug
36 self.logger = logger
37 if path:
38     self.file = open(path, "w")
39     self.file.seek(0)
40     self.fingerprints = set(self.file.read().split())
41
42 @classmethod
43 def from_settings(debug):
44     return cls(debug)
45
46 def request_fp(self, fp):
47     fp = self._normalize_fp(fp)
48     if fp in self.fingerprints:
49         return True
50     self.fingerprints.add(fp)
51     if self.file:
52         self.file.write(fp + os.linesep)
53
54 def request_fingerprint(self, fp):
55     return request_fp(self, fp)
```



# OBJETIVO



@MATTEUS\_ANTONIO



GITHUB.COM/MATEUSTOIN



MATEUSASILVA3@GMAIL.COM



# UTILIDADES



HACKATHONS



CONTESTS



PROJETOS DE  
DISCIPLINA

PROJETOS DE  
PESQUISA



PROJETOS  
PESSOAIS



ENSINO





@MATTEUS\_ANTONIO



GITHUB.COM/MATEUSTOIN



MATEUSASILVA3@GMAIL.COM

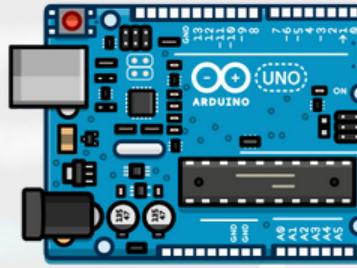


# ALÔ, BATMAN?!

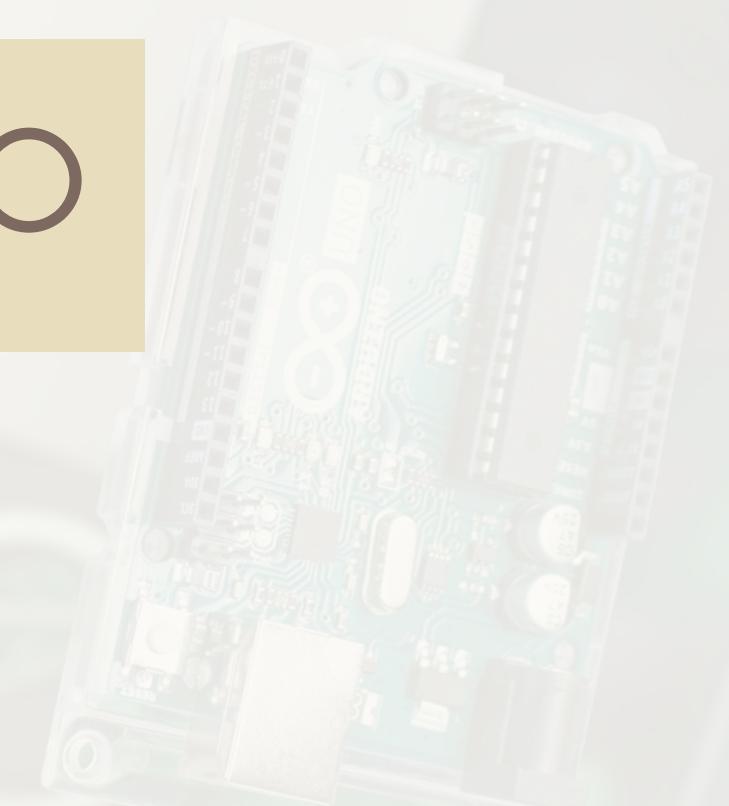


CINTO DE UTILIDADES

```
31 dev = None
32 self.file = None
33 self.fingerprints = set()
34 self.logdupes = True
35 self.debug = debug
36 self.logger = logger
37 if path:
38     self.file = open(path)
39     self.file.seek(0)
40     self.fingerprints = set(self.file.read().split())
41
42 @classmethod
43 def from_settings(debug):
44     return cls(debug)
45
46 def request_fp(self):
47     fp = self.request_fingerprint()
48     if fp in self.fingerprints:
49         return True
50     self.fingerprints.add(fp)
51     if self.file:
52         self.file.write(fp + os.linesep)
53
54 def request_fingerprint(self, request):
55     return request_fingerprint(self, request)
```



# SOBRE ARDUINO





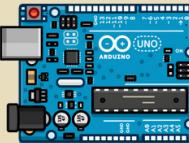
@MATTEUS\_ANTONIO



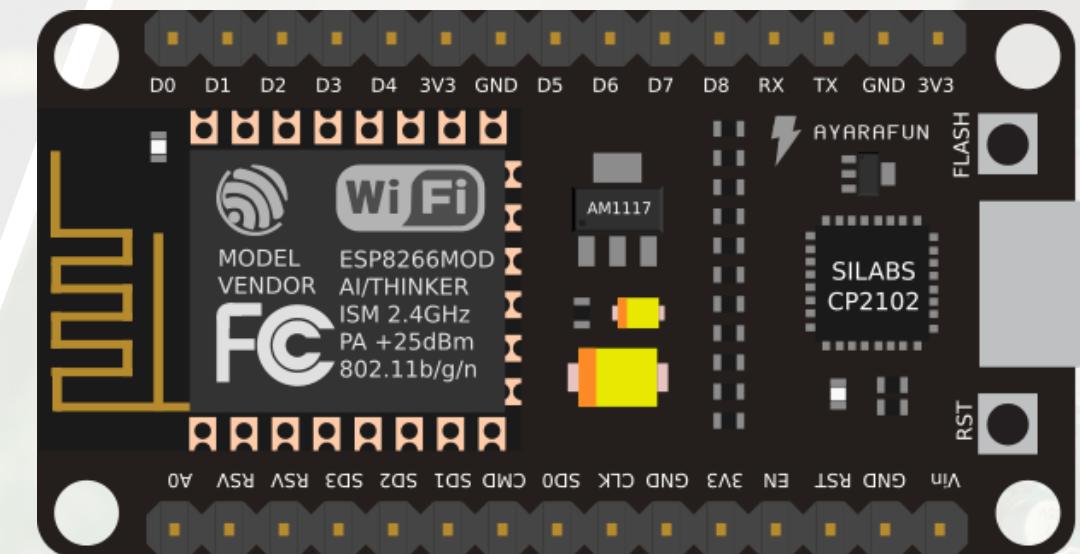
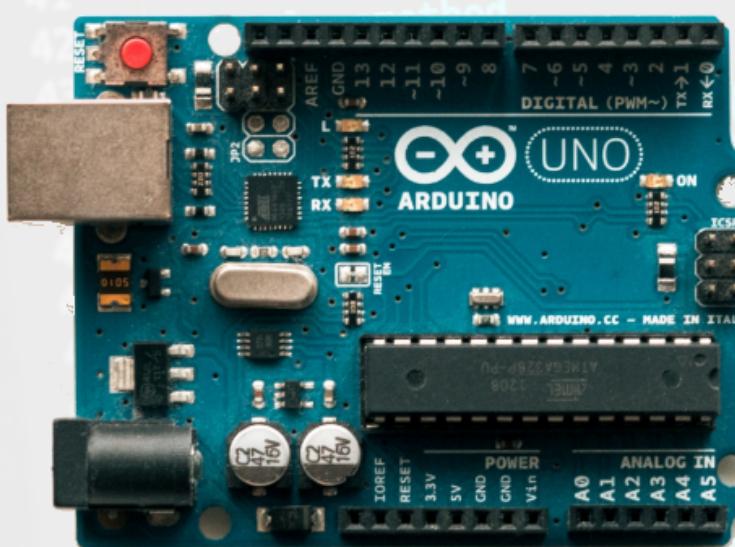
GITHUB.COM/MATEUSTOIN



MATEUSASILVA3@GMAIL.COM



# NÃO APENAS ARDUINO...





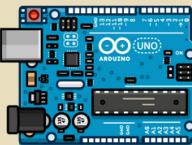
@MATTEUS\_ANTONIO



GITHUB.COM/MATEUSTOIN



MATEUSASILVA3@GMAIL.COM



# VÁRIOS TIPOS DE ARDUINO



Arduino Uno



Arduino Leonardo



Arduino Due



Arduino Yún



Arduino Tre



Arduino Micro



Arduino Robot



Arduino Esplora



Arduino Mega ADK



Arduino Ethernet



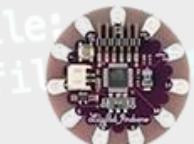
Arduino Mega 2560



Arduino Mini



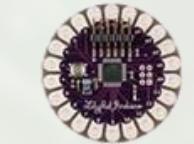
LilyPad Arduino USB



LilyPad Arduino Simple



LilyPad Arduino SimpleSnap



LilyPad Arduino



Arduino Nano



Arduino Pro Mini



@MATTEUS\_ANTONIO



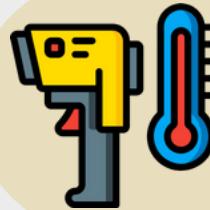
GITHUB.COM/MATEUSTOIN



MATEUSASILVA3@GMAIL.COM



# O QUE DÁ PRA FAZER?



SENSORES



ATUADORES



INTEGRAÇÃO  
COM WEB

INTEGRAÇÃO  
COM BANCO  
DE DADOS



INTEGRAÇÃO  
COM NUVEM



ETC





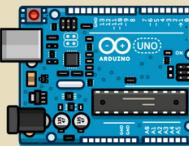
@MATTEUS\_ANTONIO



GITHUB.COM/MATEUSTOIN

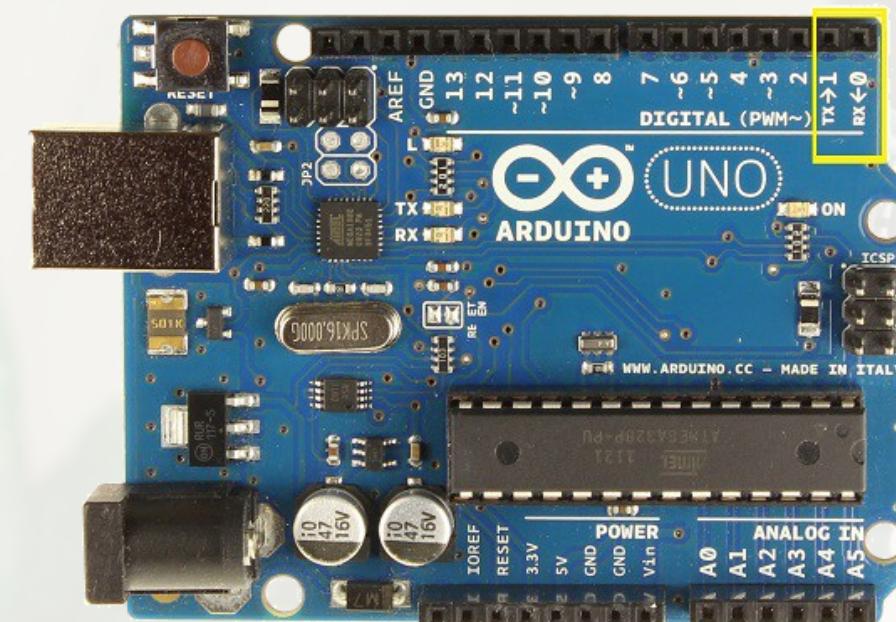


MATEUSASILVA3@GMAIL.COM



# COMUNICAÇÃO SERIAL

- O QUE É O RX E TX?
- VENDOR ID
- PRODUCT ID
- PORTAS
  - COM1, COM2, COM3...
  - /DEV/PTYACM, /DEV/PTYUSB...
- BAUDRATE
  - 300, 600, 1200, 2400, 4800, **9600**, 14400, 19200, 28800, 38400, 57600 OU 115200





# COMUNICAÇÃO SERIAL

```
Arquivo Editar Sketch Ferramentas Ajuda  
serial_read_write §  
1 #define led 13 // Porta onde o led está conectado  
2  
3 void setup(){  
4   Serial.begin(9600); // Velocidade padrão para comunicação  
5   pinMode(led, OUTPUT); // Porta onde o led será acionado, configurado como saída  
6 }  
7  
8 void loop(){  
9   if (Serial.available() > 0){  
10     char leitura = Serial.read(); // Variavel que receberá os valores enviados pelo programa em python  
11  
12     if(leitura == 'a'){  
13       digitalWrite(led, HIGH); // Liga a porta 13 se o valor recebido for 1  
14       Serial.println("Acionou led!"); // Envia mensagem para a porta Serial  
15     }  
16  
17     if(leitura == 'b'){  
18       digitalWrite(led, LOW); // Desliga a porta 13 se o valor recebido for 2  
19       Serial.println("Desligou led!"); // Envia mensagem para a porta Serial  
20     }  
21   }  
22 }
```

/dev/ttyACM0

9600 velocidade

a

Acionou led!

b

Desligou led!

DOCUMENTAÇÃO  
SERIAL ARDUINO



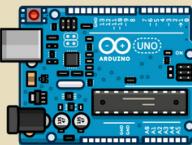
@MATTEUS\_ANTONIO



GITHUB.COM/MATEUSTOIN



MATEUS.ANTONIO@ENG.CI.UFPB.BR



self.  
self.logout  
self.debug  
self.logger  
if path:  
self.

# IMAGINE AS POSSIBILIDADES

**Arduino Altair 8800 Simulator**  
David Hansel  
Ever wanted to play with the computer that sparked the personal computer age but don't have the cash to buy an Altair? Build your own!  
83,153 Views 160 Respects

**Arduino Lie Detector**  
Dante Roumela  
Someone has been eating my cookies! Thankfully **Arduino** is here to help find out who it is.  
77,821 Views 152 Respects

**Arduino Wireless Weather Station**  
Nick Koumaris  
In this tutorial I am going to show you how to build a Wireless Weather Station with a big 3.2" Color TFT display using **Arduino**.  
77,223 Views 138 Respects

**Arduino Kitchen Timer**  
Angelo Florillo  
An **Arduino** UNO-based classic kitchen timer with LCD display and buzzer  
74,827 Views 101 Respects

**Arduino Game By LCD**  
Mohammed Magdy  
How to create a simple LCD video game with **Arduino**.  
73,324 Views 230 Respects

**Arduino UNO + 2.4 TFT LCD Display Shield Touch Panel ILI9341**  
calogerus  
Basic code to make **Arduino** communicate with ILI9341.  
72,063 Views 50 Respects

**Arduino UNO Guitar Pedal**  
electromash

**Arduino Bluetooth Basic Tutorial**  
by Mayoogh Girish  
723,873 VIEWS 73 COMMENTS 429 RESPECTS

**DIY CNC**  
**Arduino Based Mini CNC 2D Plotter**  
Project showcase by Mrinnovative  
115,526 VIEWS 19 COMMENTS 266 RESPECTS

**Arduino Battery Shield** by MarPok in Arduino  
1 ★ 1 501 62K

**Arduino Parking Assistant** by addictedToArduino in Arduino  
1 ★ Second Prize 3 405 68 73K

**Arduino Powered Headress** by calmac\_projects in Arduino  
1 ★ Second Prize 13 375 69 35K

**Getting Started With Arduino** by bekafishwa in Arduino  
1 ★ 2 327 68 29K

**Arduino Ble Rover** by Arbot in Arduino  
1 ★ 241 68 15K

**Custom Arduino guitar** by MrAtkinson in Arduino  
1 ★ 278 46 38K

**Arduino Altair 8800 Simulator**  
Project tutorial by David Hansel

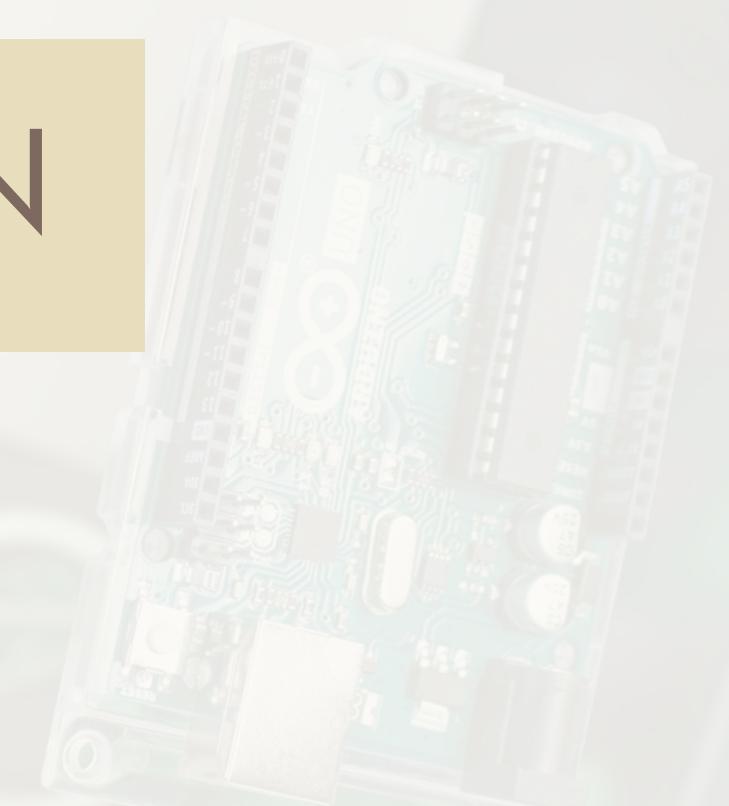
**ARDUINO FINGERPRINT SENSOR**  
Arduino Fingerprint Sensor Tutorial

**instructables**

```
31
32     self.file = None
33     self.fingerprints = set()
34     self.logdupes = True
35     self.debug = debug
36     self.logger = logger
37     if path:
38         self.file = open(path, "w")
39         self.file.seek(0)
40         self.fingerprints.add(self.file.read())
41
42     @classmethod
43     def from_settings(settings):
44         debug = settings.get("debug", False)
45         return cls(debug=debug)
46
47     def request_file(self, fp):
48         fp = self._normalize_fp(fp)
49         if fp in self.fingerprints:
50             return True
51         self.fingerprints.add(fp)
52         if self.file:
53             self.file.write(fp + os.linesep)
54
55     def request_fingerprint(self, fp):
56         return request_fingerprint(self, fp)
```



# SOBRE PYTHON





@MATTEUS\_ANTONIO



GITHUB.COM/MATEUSTOIN



MATEUSASILVA3@GMAIL.COM



```
31  
32  
33 self.  
34 self.log.setLevel(logging.INFO)  
35 self.debug = True  
36 self.logger = logging.getLogger()  
37 if path:  
38     self.file = open(path, 'w')  
39     self.file.write("")  
40
```

# COMUNICAÇÃO SERIAL

```
1 # -*- coding: iso-8859-1 -*-  
2 import serial  
3  
4 # Abre porta Serial com seus devidos parâmetros  
5 ser = serial.Serial(port='/dev/ttyACM0', baudrate=9600, timeout=1)  
6  
7 # Lê repetidamente e imprime qualquer mensagem que vem do arduino  
8 while True:  
9     msg = ser.readline().decode('ascii')  
10    print(msg)
```



CÓDIGO DE  
LEITURA

```
1 # -*- coding: iso-8859-1 -*-  
2 import serial  
3  
4 # Abre porta Serial com seus devidos parâmetros  
5 arduino = serial.Serial(port='/dev/ttyACM0', baudrate=9600, timeout=1)  
6  
7 # escreve uma string na porta serial  
8 arduino.write(b'a')  
9 #arduino.write(bytes('a', encoding='utf-8'))  
10 #  
11 arduino.close() # Fecha porta de comunicação
```



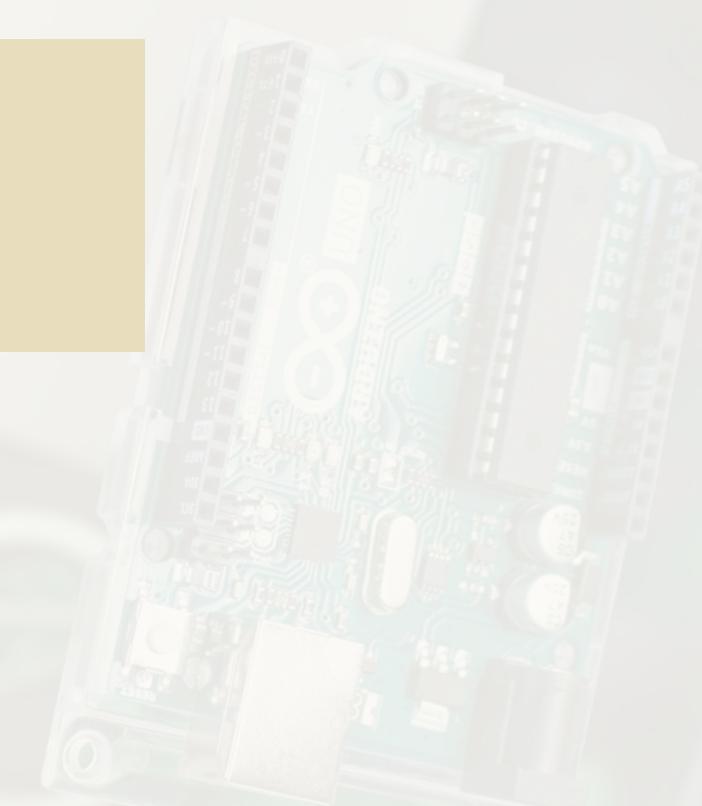
CÓDIGO DE  
ESCRITA

DOCUMENTAÇÃO  
SERIAL PYTHON

```
31 dev = True
32 self.file = None
33 self.fingerprints = set()
34 self.logdupes = True
35 self.debug = debug
36 self.logger = logger
37 if path:
38     self.file = open(path, "w")
39     self.file.seek(0)
40     self.fingerprints = set(self.file.read().split())
41
42 @classmethod
43 def from_settings(debug):
44     return cls(debug)
45
46 def request_fp():
47     fp = self.request_fingerprint()
48     if fp in self.fingerprints:
49         return True
50     self.fingerprints.add(fp)
51     if self.file:
52         self.file.write(fp + os.linesep)
53
54 def request_fingerprint(self, req_id):
55     return request_fingerprint(req_id)
```



# APLICAÇÕES





@MATTEUS\_ANTONIO



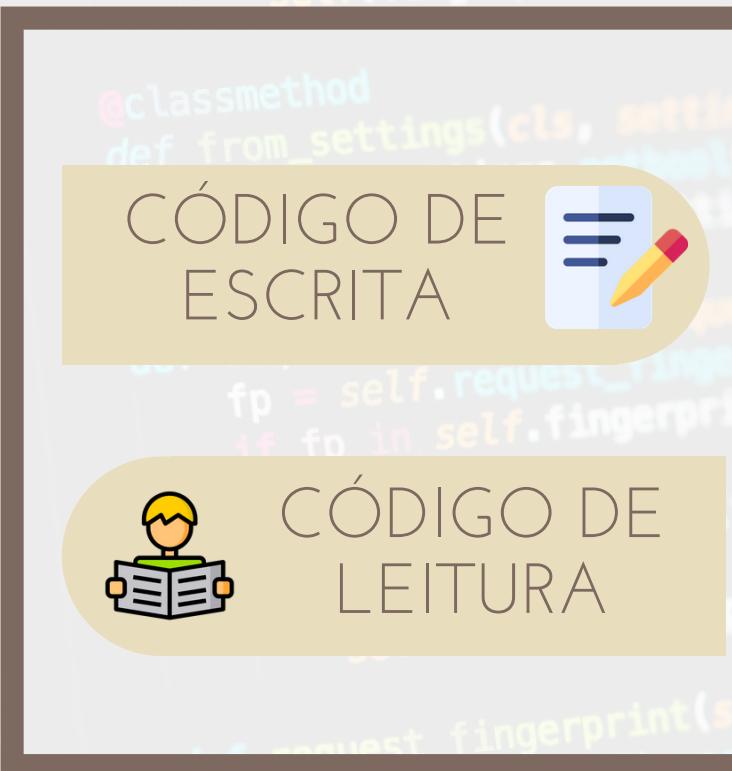
GITHUB.COM/MATEUSTOIN



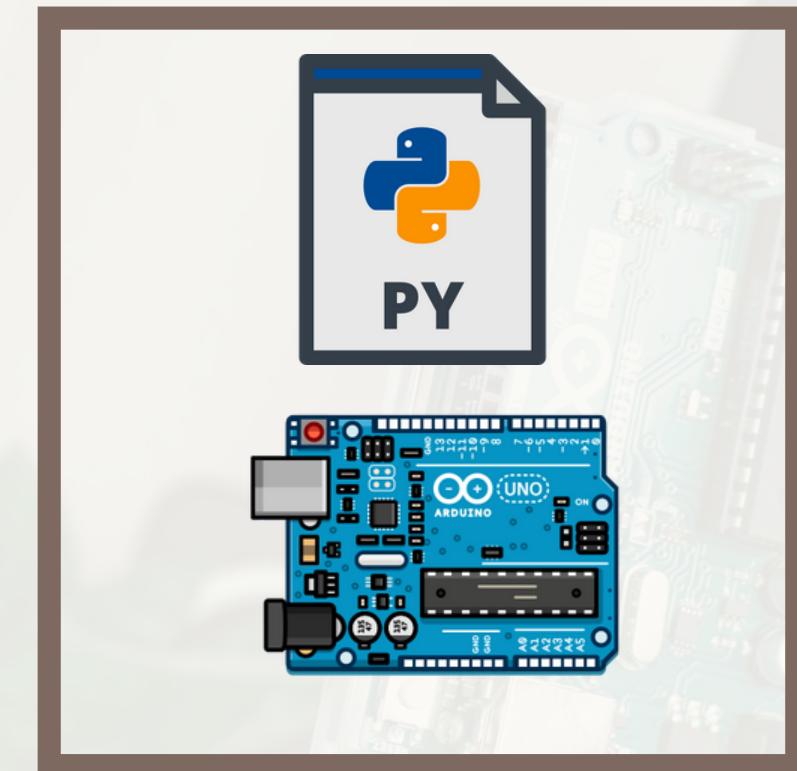
MATEUSASILVA3@GMAIL.COM



# INTEGRAÇÃO ARDUINO + PYTHON



CLIQUE NA  
IMAGEM





@MATTEUS\_ANTONIO



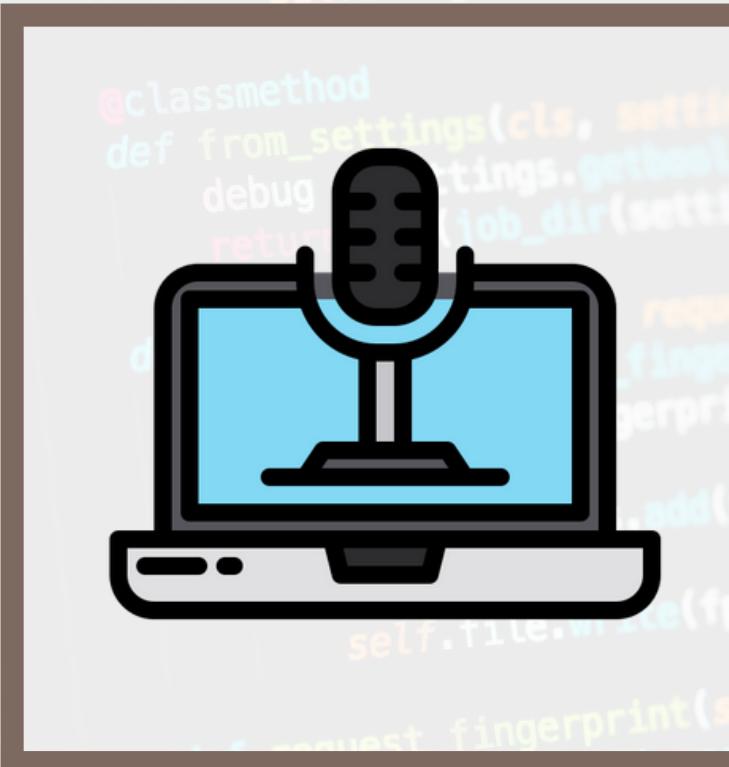
GITHUB.COM/MATEUSTOIN



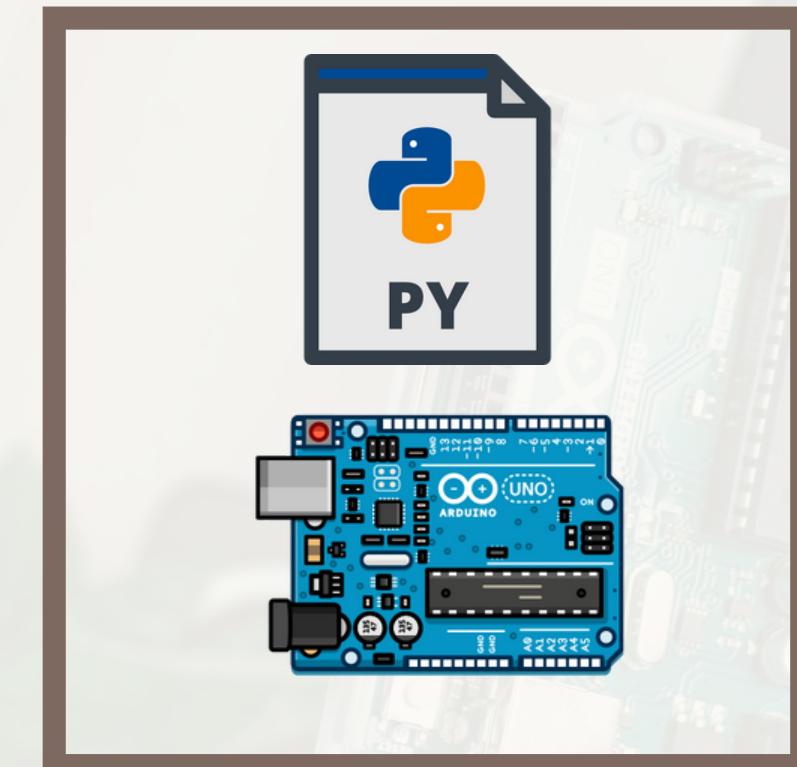
MATEUSASILVA3@GMAIL.COM



# CONTROLANDO ARDUINO POR VOZ



CLIQUE NA  
IMAGEM





@MATTEUS\_ANTONIO



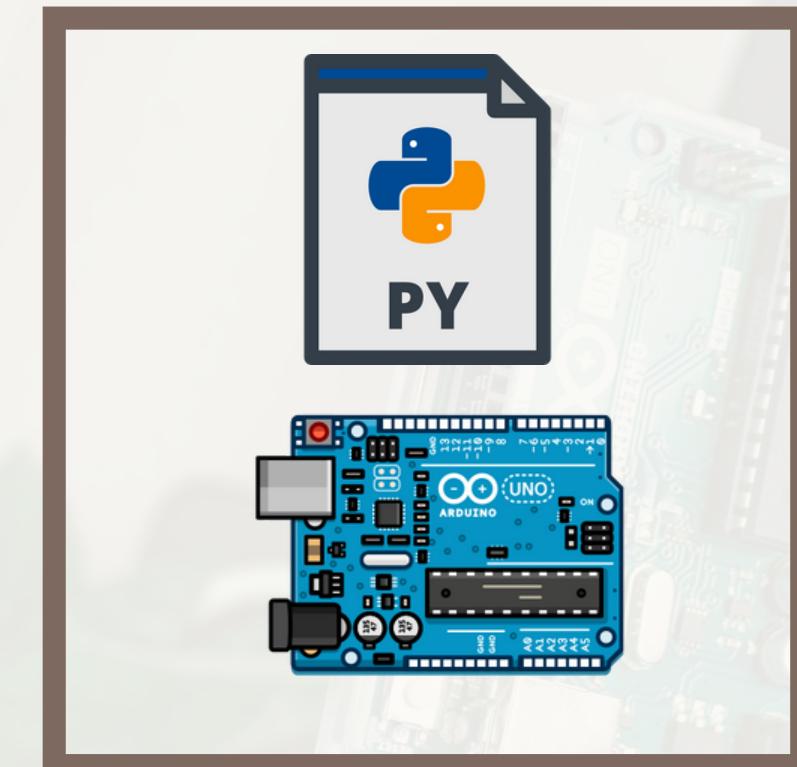
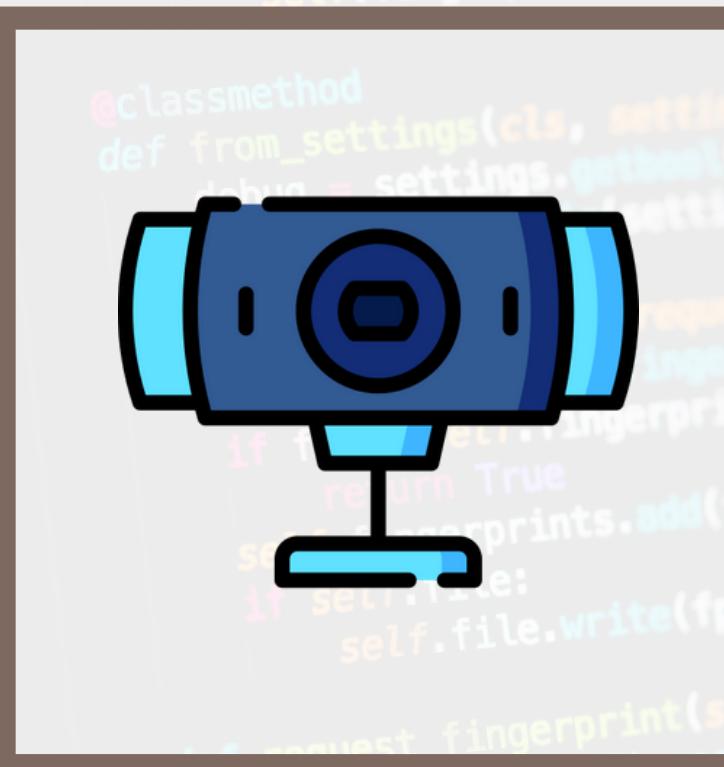
GITHUB.COM/MATEUSTOIN



MATEUSASILVA3@GMAIL.COM



# CONTROLANDO CÂMERA COM O ARDUINO



CLIQUE NA  
IMAGEM



@MATTEUS\_ANTONIO



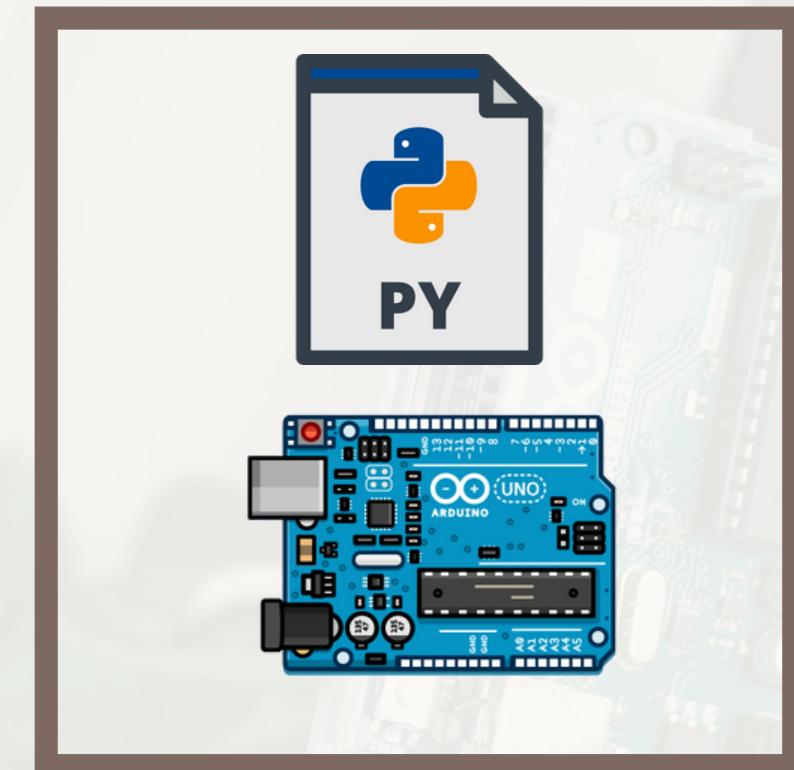
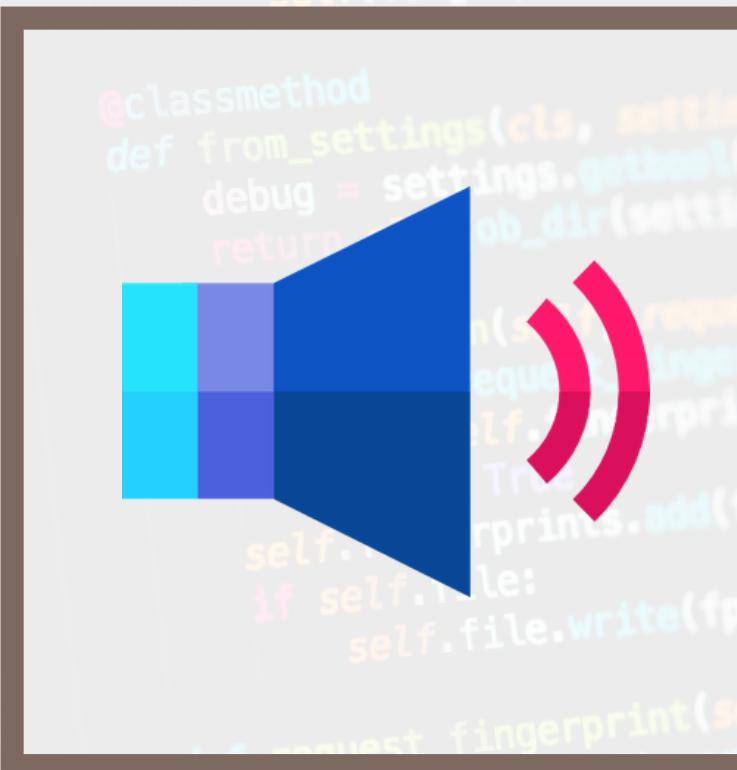
GITHUB.COM/MATEUSTOIN



MATEUSASILVA3@GMAIL.COM



# ACIONANDO SONS COM O ARDUINO



CLIQUE NA  
IMAGEM

# CONTATO



@MATTEUS\_ANTONIO



GITHUB.COM/MATEUSTOIN



MATEUSASILVA3@GMAIL.COM