



BEM VINDOS!
VENTURUS⁴TECH



Módulo IV Internet das Coisas

VENTURUS4TECH

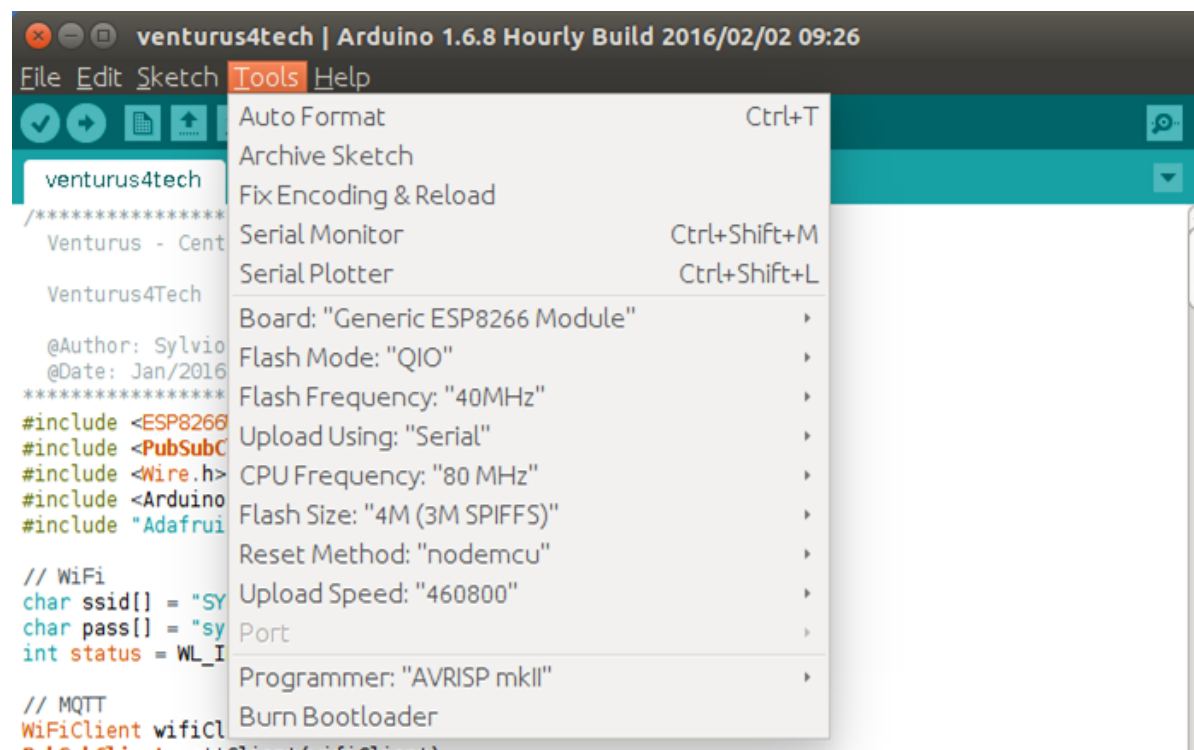
Pedro Dionísio
pedro.pereira@venturus.org.br

Preparando o ambiente (MAC)

- Instalar FTDI Driver:
 - <https://goo.gl/lXSTRO>
- Adicionar ESP8266 board package:
 - http://arduino.esp8266.com/stable/package_esp8266com_index.json
- Instalar board: Menu Tools/Board/Boards Manager
 - Instalar esp8266
- Adicionar Libraries: Menu Sketch/Include Library/Manage Libraries
 - PubSubClient
 - ArduinoJson
 - Adafruit MCP9808

Configurando a placa ESP8266

- Port:
 - usbserial OU
 - usbmodem



Na Aula 1: (link: <https://https://goo.gl/FJvxzJ>)

// Funções obrigatórias

```
void setup();
```

```
void loop();
```

// Outras funções

```
pinMode(porta, OUTPUT); // define se uma porta é entrada ou saída
digitalWrite(porta, LOW); // coloca a porta em LOW ou HIGH (liga ou desliga)
digitalRead(porta); // le estado de uma porta. Retorna LOW ou HIGH
delay(1000); // causa delay de 1s bloqueando thread
Serial.begin(115200); // inicia porta serial em 115200 baud
Serial.println("texto"); // envia o texto para a serial
WiFi.begin(ssid, pass); // inicia WiFi com ssid e senha
```

Aula 2

```
Serial.available()      // verifica se há dados na serial  
Serial.read();          // le um caractere da serial  
Serial.readStringUntil(); // le uma string até encontrar caractere
```

```
ArduinoJson           // tratar strings json  
Adafruit_MCP9808();   // sensor de temperatura  
PubSubClient          // Cliente MQTT
```

Exemplo 6 - Ler e imprimir Serial

```
Serial.available()    // verifica se há dados na serial  
Serial.read();        // le um caractere da serial
```

Exemplo 6 - Ler e imprimir Serial

```
void setup() {  
  pinMode(0, OUTPUT);    // led vermelho  
  Serial.begin(115200);  
  delay(2000);  
  
  Serial.println("Ligando Arduino...");  
}
```

```
void loop() {  
  // Se existe conteúdo na porta serial  
  if (Serial.available()) {  
  
    // leia o caractere  
    char c = Serial.read();  
  
    // imprimir o caractere  
    Serial.print(c);  
  }  
}
```



Exemplo 7 - Acionar LED pela Serial

```
void ligaLed(int porta);    // liga o led na porta
void desligaLed(int porta); // desliga o led na porta
Serial.available();
Serial.readStringUntil('\n')
String texto;

if (texto == "led=1") {
    ligaLed(0);
} else if (texto == "led=0") {
    desligaLed(0);
}
```

Exemplo 7 - Acionar LED pela Serial

```
void ligaLed(int porta);    // liga led porta
void desligaLed(int porta); // desliga led porta

void setup() {
    pinMode(0, OUTPUT);    // led vermelho
    desligaLed(0);

    Serial.begin(115200);
    delay(2000);

    Serial.println("Ligando Arduino...");
    Serial.println("Aguardando entrada...");
}
```

```
void loop() {
    // Se existe conteúdo na porta serial
    if (Serial.available()) {
        // le porta serial ate final de linha
        String texto = Serial.readStringUntil('\n');

        if (texto == "led=1") {
            ligaLed(0);
        } else if (texto == "led=0") {
            desligaLed(0);
        }

        Serial.print(texto);
        Serial.println();
    }
}
```

Exemplo 8 - Criando ArduinoJson

```
#include <ArduinoJson.h>

void cria_json();

void setup() {
  pinMode(0, OUTPUT); // led vermelho

  Serial.begin(115200);
  delay(2000);

  Serial.println("Ligando Arduino...");
  Serial.println();

  cria_json();
}
```

```
void cria_json() {

  StaticJsonBuffer<200> jsonBuffer;

  JsonObject& buffer = jsonBuffer.createObject();
  buffer["lampada"] = true;
  buffer["temperatura"] = 25.4;

  buffer.printTo(Serial);
  Serial.println();
  buffer.prettyPrintTo(Serial);
}

void loop() {
}
```

Exemplo 9 - Extraindo ArduinoJson

```
#include <ArduinoJson.h>

void extrai_json(String json);

void setup() {
    pinMode(0, OUTPUT);    // led vermelho

    Serial.begin(115200);
    delay(2000);

    Serial.println("Ligando Arduino...");
    Serial.println();
}

void loop() {
    (...)
}
```

```
// JSON: {"lampada":true,"temperatura":25.4}
void extrai_json(String json) {
    StaticJsonBuffer<200> jsonBuffer;
    JsonObject& buffer = jsonBuffer.parseObject(json);

    if (!buffer.success()) {
        Serial.println("Json apresenta erros..");
        return;
    }

    if (buffer.containsKey("lampada")) {
        boolean lampada = buffer["lampada"];
        Serial.print("Lâmpada: ");
        Serial.println(lampada == true ? "Ligada" : "Desligada");
    }

    if (buffer.containsKey("temperatura")) {
        double temperatura = buffer["temperatura"];
        Serial.print("Temperatura: ");
        Serial.println(temperatura);
    }
}
```

Exemplo 10 - Sensor MCP9808

- Como funciona?
 - I²C - SCL/SDA
- Como Ligar?
 - Arduino < - > MCP9808
 - SCL < ----- > SCL
 - SDA < ----- > SDA
 - 3V < ----- > VDD
 - GND < ----- > GND
- NÃO PODE LIGAR ERRADO! QUEIMA!
 - ~~VCC <-----> GND~~

Exemplo 10 - Sensor MCP9808

```
#include <Wire.h>
#include "Adafruit_MCP9808.h"

Adafruit_MCP9808 sensor = Adafruit_MCP9808();

void mostra_temperatura();

void setup() {
  Serial.begin(115200);
  delay(2000);

  Serial.println("Ligando Arduino...");
  Serial.println();

  // inicializa comunicação com sensor de temperatura
  if (!sensor.begin()) {
    Serial.println("Sensor não encontrado!");
    while (1);
  }

  // inicializa sensor
  sensor.shutdown_wake(0);
}
```

```
void mostra_temperatura() {
  float c = sensor.readTempC();
  delay(250);

  Serial.print("Temperatura: ");
  Serial.print(c);
  Serial.println(" C");
}

void loop() {
  mostra_temperatura();
  delay(1000);
}
```

Exemplo 11 - Juntando tudo!

```
// recebe um Json da serial e extrai as informações
// {"status":0} -> Deve retornar um json completo com temperatura e estado do LED
// {"seta_led":ESTADO} -> liga ou desliga o LED. Valores: ligado, desligado, piscando
// {"pisca_led":MILLIS} -> led fica piscando. Valores: delay em millis (1000 = 1 segundo)
void analisaJsonSerial(String json);

// envia um json completo
// {"led":ESTADO,"temperatura":VALOR} -> ESTADO "ligado", "desligado", "piscando"
void enviaJsonSerial();

void ligaLed(int porta);
void desligaLed(int porta);

// liga, desliga ou faz led piscar
void atualiza_led();
```