
Nombre: **Roel De la Rosa**

Módulo 4 - Evidencias

Clase: **Inteligencia artificial avanzada para la ciencia de datos**

Profesor: **Antonio Carlos Bento**

Fecha: **12 de Septiembre de 2022**

1 Actividad Semana 1

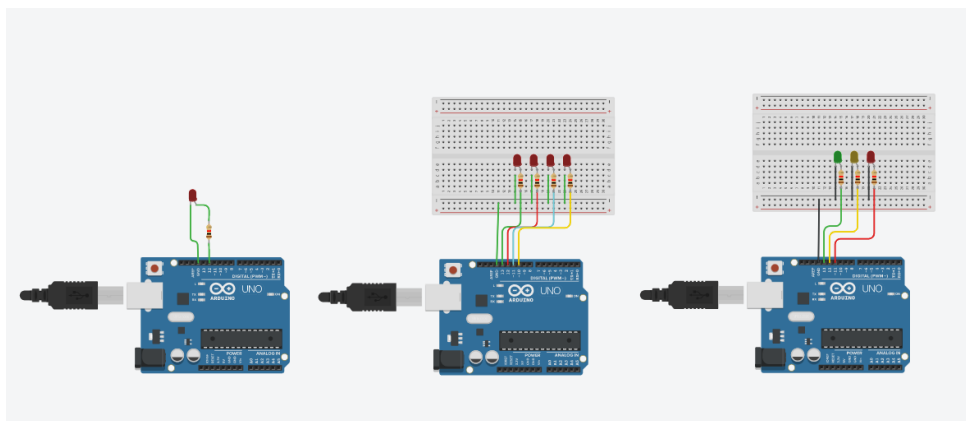


Figure 1: Tres arduinos de la actividad 1

1.1 Código del primer arduino

```
int led = 12;
```

```
// the setup routine runs once when you press reset: void setup() {
```

```
void setup()  
{
```

```
pinMode(led ,OUTPUT);
Serial.begin(9600);

}

// the loop routine runs over and over again forever: void loop() {
void loop()
{
  delay(500);

  digitalWrite(led , HIGH);

  // wait for 500 ms
}
}
```

1.2 Código del segundo arduino

```
int led1 = 13;
int led2 = 12;
int led3 = 11;
int led4 = 10;

// the setup routine runs once when you press reset: void setup() {

void setup()
{

  pinMode(led1 ,OUTPUT);
```

```
pinMode(led2 ,OUTPUT);
pinMode(led3 ,OUTPUT);
pinMode(led4 ,OUTPUT);
Serial.begin(9600);

}

// the loop routine runs over and over again forever: void loop() {
void loop()
{
  delay(500);
  digitalWrite(led1 , HIGH);

  digitalWrite(led2 , HIGH);

  digitalWrite(led3 , HIGH);

  digitalWrite(led4 , HIGH);
```

1.3 Código del tercer Arduino

```
int led1 = 13;
int led2 = 12;
int led3 = 11;
// the setup routine runs once when you press reset: void setup() {

void setup()
{

  pinMode(led1 ,OUTPUT);
  pinMode(led2 ,OUTPUT);
  pinMode(led3 ,OUTPUT);
```

```
Serial.begin(9600);

}

// the loop routine runs over and over again forever: void loop() {
void loop()
{

digitalWrite(led1 , HIGH);
digitalWrite(led2 , LOW);
digitalWrite(led3 , LOW);
delay(8000);

digitalWrite(led1 , LOW);
digitalWrite(led2 , HIGH);
digitalWrite(led3 , LOW);
delay(3000);

digitalWrite(led1 , LOW);
digitalWrite(led2 , LOW);
digitalWrite(led3 , HIGH);
delay(10000);
}
```

1.4 Tres arduinos a media ejecución

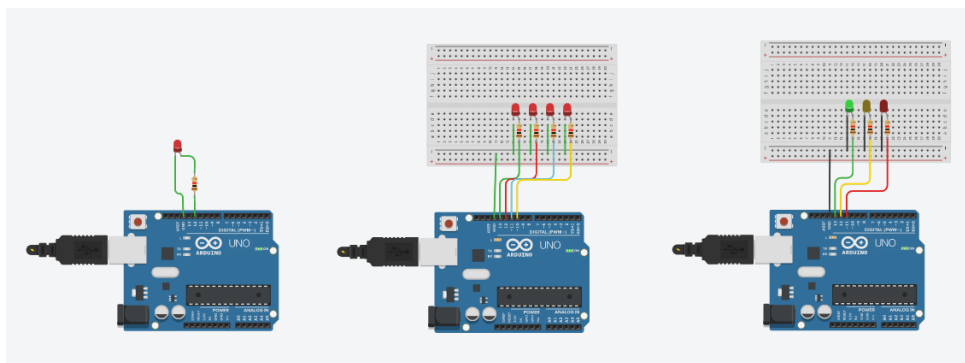


Figure 2: Tres arduinos de la actividad a media ejecución

2 Actividad Semana 2

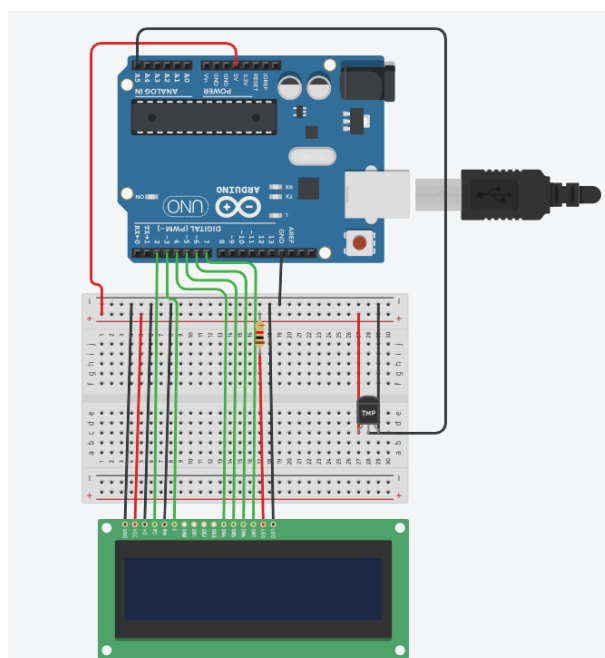


Figure 3: Arduino de la actividad 2

2.1 Código de la actividad 2

```
#include <LiquidCrystal.h>

// C++ code
//
LiquidCrystal lcd(2,3,4,5,6,7);
void setup()
```

```
{
  Serial.begin(9600);
}

void loop()
{
  double temperatura=((analogRead(A5)*(5.0/1024))-0.5)/0.01;
  lcd.clear();
  lcd.print(String(temperatura) + " C ");

  delay(3000);

  if (temperatura < 18){
    lcd.clear();
    lcd.print("Hace_frio");
  } else {
    if(temperatura >=18 && temperatura < 32){
      lcd.clear();
      lcd.print("Esta_perfecto");
    } else {
      lcd.clear();
      lcd.print("Hace_Calor");
    }
  }
  delay(3000);
}
```

2.2 Ejecución del arduino de la actividad 2

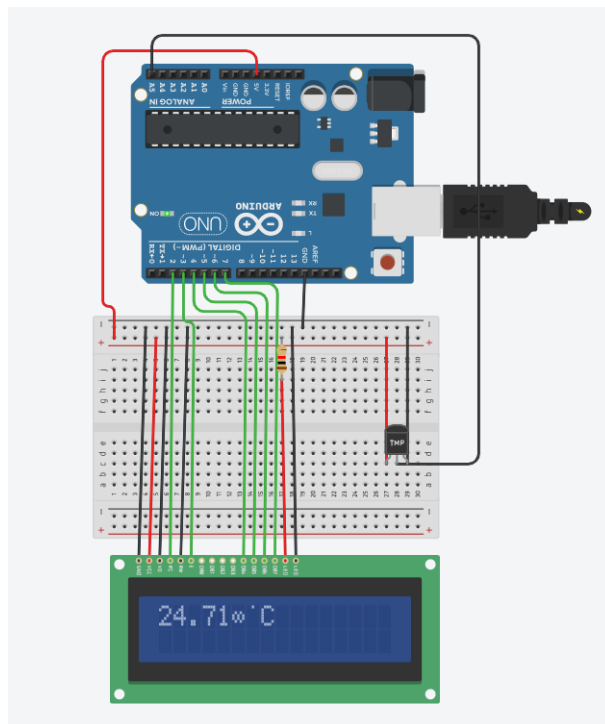


Figure 4: Arduino en ejecución de la actividad 2

3 Actividad Semana 3

3.1 Arduino de la actividad 3

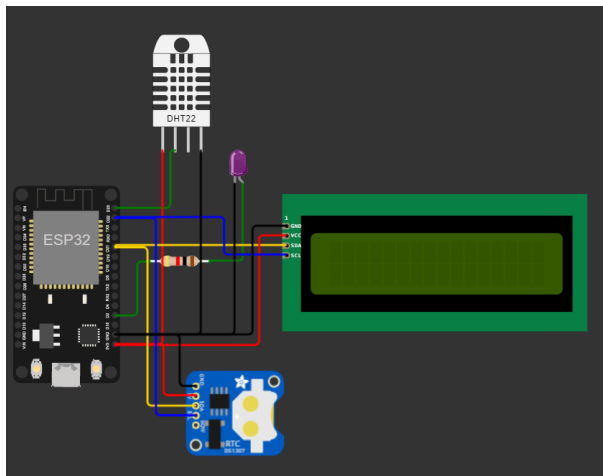


Figure 5: Arduino de la actividad 3

3.2 Código de la actividad 3

```
#include <LiquidCrystal_I2C.h>
#include <DHTesp.h>
#include <HTTPClient.h>
#include <Arduino.h>
#include <ArduinoJson.h>
#include <WiFi.h>
#include <RTCLib.h>

//setup
#define DHT_PIN 23
#define LED 2
LiquidCrystal_I2C LCD = LiquidCrystal_I2C(0x27, 16, 2);
DHTesp dhtSensor;
HTTPClient client;
RTC_DS1307 rtc;
```



```
//variables
float temperatura;
float humedad;
String temp_Post;
String hum_Post;
const char* ssid = "Wokwi-GUEST";
const char* password = "";

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

    // hardware
    LCD.init();
    LCD.backlight();
    LCD.setCursor(0, 1);
    dhtSensor.setup(DHT_PIN, DHTesp::DHT22);
    pinMode(LED, OUTPUT);

    // firebase
    WiFi.mode(WIFI_STA);
    WiFi.begin(ssid, password);

    // esperar conexion
    while (WiFi.status() != WL_CONNECTED)
    {
        Serial.println("Attempting connection...");
        delay(500);
    }

    client.begin("https://exemplo-clase-default-rtdb.firebaseio.com/.json");
    client.addHeader("Content-Type", "application/json");
```

```
// fecha
if (! rtc.begin()) {
    Serial.println("Couldn't find RTC");
    Serial.flush();
    abort();
}

if (! rtc.isrunning()) {
    Serial.println("RTC is NOT running, let's set the time!");
    rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
}
}

void loop() {
    TempAndHumidity sensorData = dhtSensor.getTempAndHumidity();
    DateTime time = rtc.now();
    temperatura = sensorData.temperature;
    humedad = sensorData.humidity;

    if (temperatura < 18)
    {
        temp_Post = "Temp: Frio      ";
    }
    else if (temperatura >= 18 && temperatura < 32)
    {
        temp_Post = "Temp: Perfecta";
    }
    else
    {
        temp_Post = "Temp: Caliente   ";
    }
}
```

```
LCD.setCursor(0,0);
LCD.println(temp_Post);
LCD.setCursor(0,1);
LCD.println("HUMEDAD: " + String(humedad, 1) + "%");

String json;
StaticJsonDocument<200> doc;
doc["Humedad"] = String(humedad, 1) + "%";
doc["Temperatura"] = String(temperatura, 2) + " C ";
doc["Sensor"] = "DHT22";

if (temperatura < 18)
{
    doc["Status"] = "Frio";
}
else if (temperatura >= 18 && temperatura < 32)
{
    doc["Status"] = "Perfecta";
}
else
{
    doc["Status"] = "Caliente";
}

doc["Fecha"] = time.timestamp(DateTime::TIMESTAMP_DATE);
serializeJson(doc, json);
client.PATCH(json);
String payload = client.getString();
Serial.println(payload);

for (int i = 0; i < 5; i++)
```

```
{  
    digitalWrite(LED, HIGH);  
    delay(500);  
    digitalWrite(LED, LOW);  
    delay(500);  
}  
  
delay(30000);  
}
```