

CURSO DE ARDUINO

DIRIGIDO POR: MIGUEL ANGEL CALIFA URQUIZA

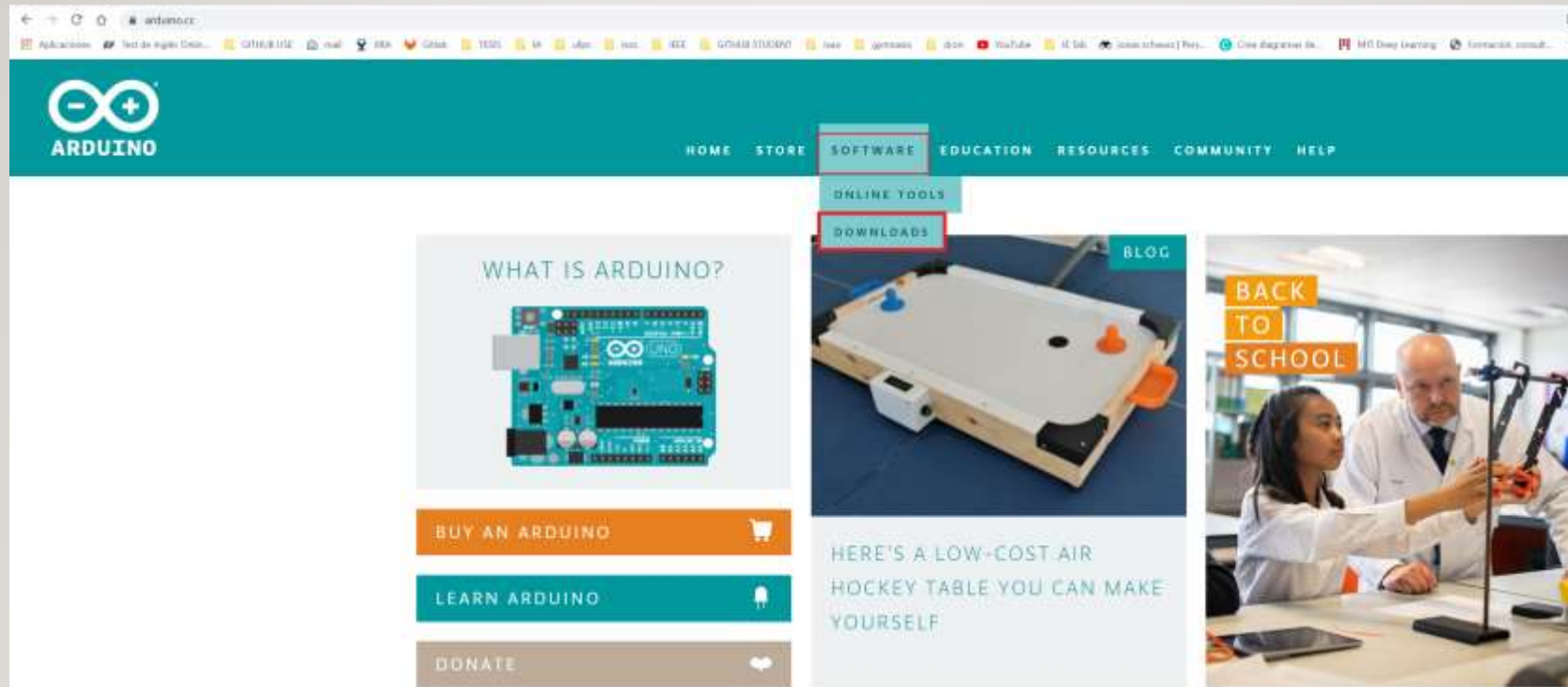


AGENDA

- Como instalar el IDE de Arduino.
- Menú de herramientas.
- Estructura básica de un programa.
- Ejemplos basicos.

COMO INSTALAR EL IDE DE ARDUINO

- URL: <https://www.arduino.cc>



COMO INSTALAR EL IDE DE ARDUINO

Download the Arduino IDE



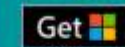
ARDUINO 1.8.9

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software.

This software can be used with any Arduino board. Refer to the [Getting Started](#) page for installation instructions.

Windows Installer, for Windows XP and up
Windows ZIP file for non admin install

Windows app Requires Win 8.1 or 10



Mac OS X 10.8 Mountain Lion or newer

Linux 32 bits

Linux 64 bits

Linux ARM 32 bits

Linux ARM 64 bits

[Release Notes](#)

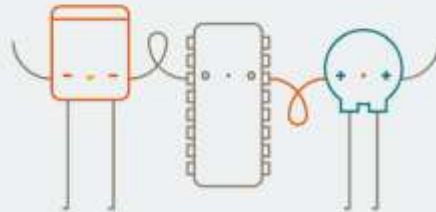
[Source Code](#)

[Checksums \(sha512\)](#)

COMO INSTALAR EL IDE DE ARDUINO

Contribute to the Arduino Software

Consider supporting the Arduino Software by contributing to its development. (US tax payers, please note this contribution is not tax deductible). [Learn more on how your contribution will be used.](#)



SINCE MARCH 2015, THE ARDUINO IDE HAS BEEN DOWNLOADED **34,809,090** TIMES. (IMPRESSIVE!) NO LONGER JUST FOR ARDUINO AND GENUINO BOARDS, HUNDREDS OF COMPANIES AROUND THE WORLD ARE USING THE IDE TO PROGRAM THEIR DEVICES, INCLUDING COMPATIBLES, CLONES, AND EVEN COUNTERFEITS. HELP ACCELERATE ITS DEVELOPMENT WITH A SMALL CONTRIBUTION! REMEMBER: OPEN SOURCE IS LOVE!

\$3

\$5

\$10

\$25

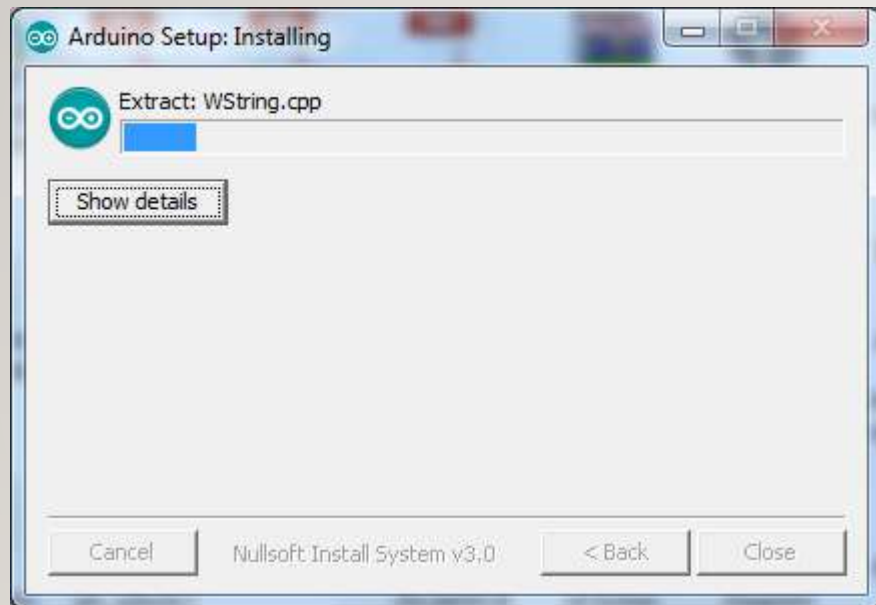
\$50

OTHER

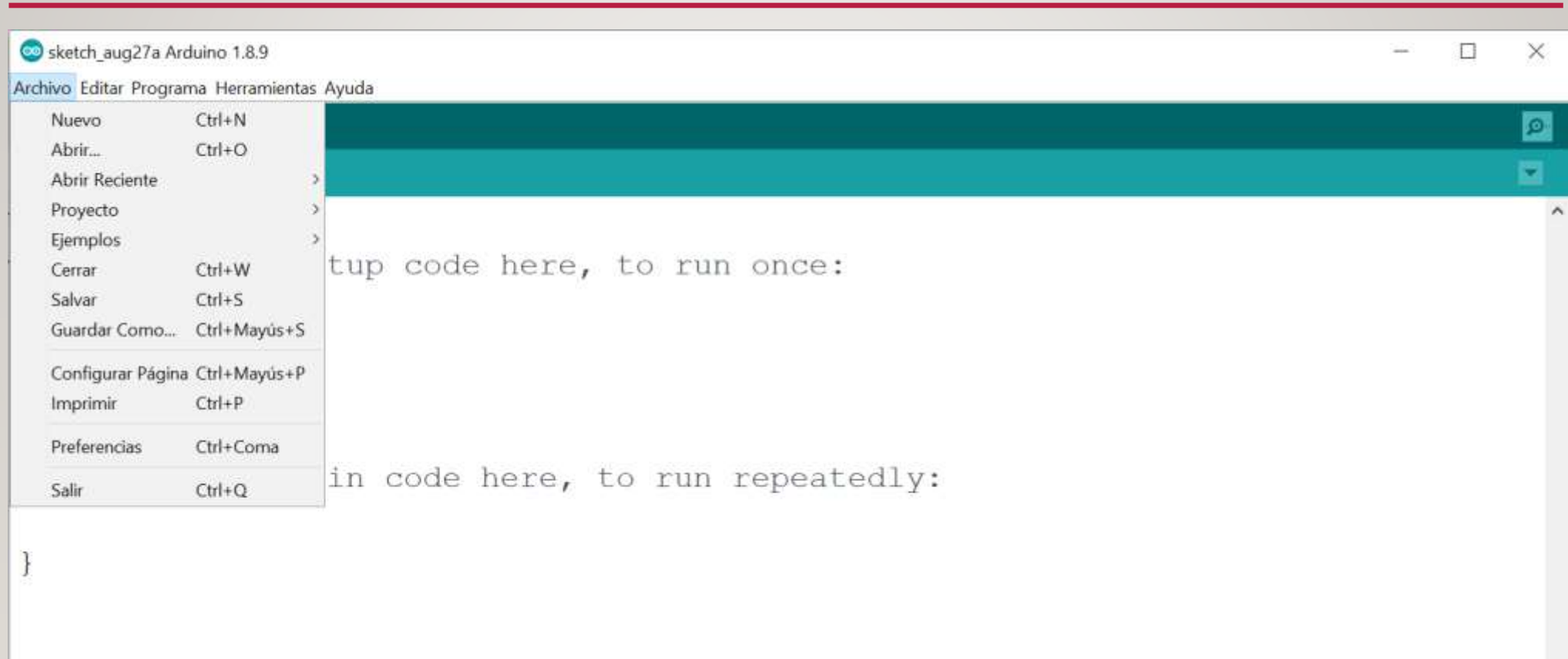
JUST DOWNLOAD

CONTRIBUTE & DOWNLOAD

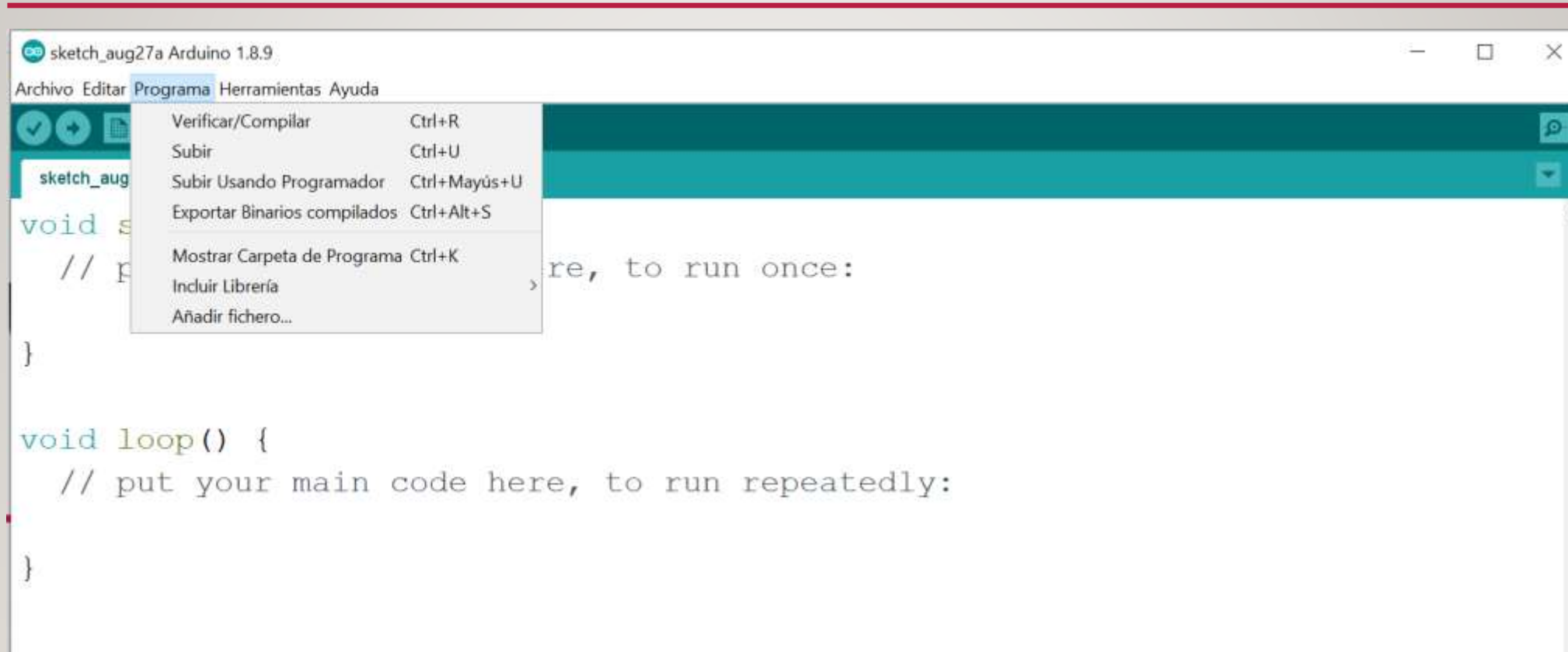
COMO INSTALAR EL IDE DE ARDUINO



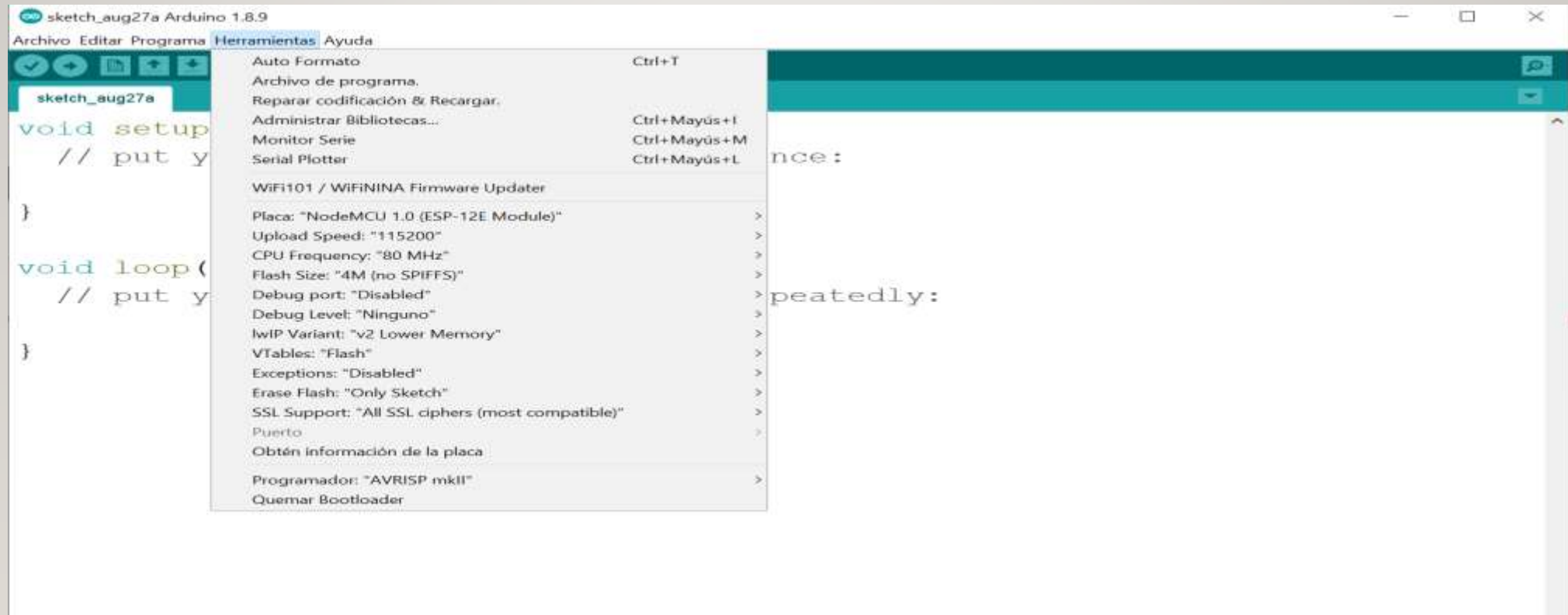
MENU DE ARDUINO



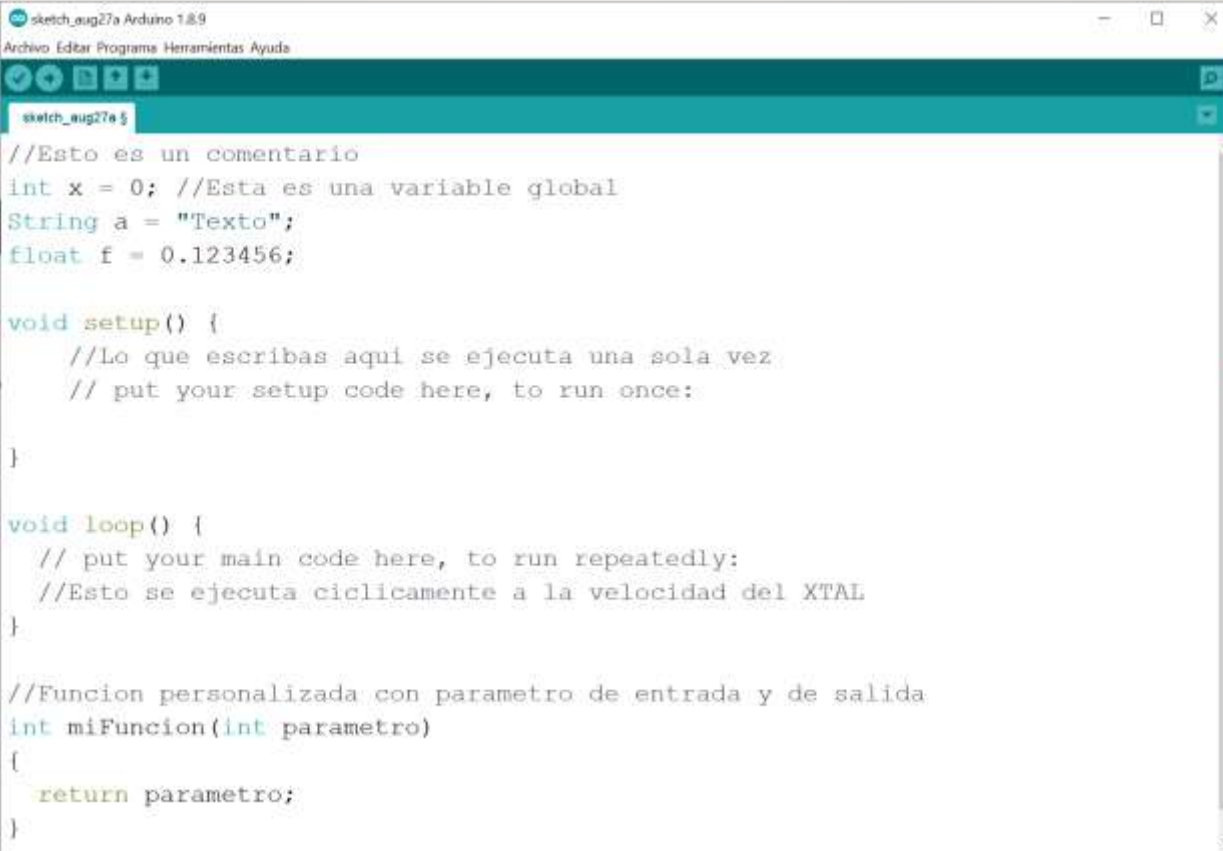
MENU DE ARDUINO



MENU DE ARDUINO



ESTRUCTURA DE UN PROGRAMA EN ARDUINO



```
sketch_aug27a Arduino 1.8.9
Archivo Editar Programa Herramientas Ayuda
sketch_aug27a §
//Esto es un comentario
int x = 0; //Esta es una variable global
String a = "Texto";
float f = 0.123456;

void setup() {
    //Lo que escribas aqui se ejecuta una sola vez
    // put your setup code here, to run once:
}

void loop() {
    // put your main code here, to run repeatedly:
    //Esto se ejecuta ciclicamente a la velocidad del XTAL
}

//Funcion personalizada con parametro de entrada y de salida
int miFuncion(int parametro)
{
    return parametro;
}
```

EJEMPLO BÁSICO (BLINK)

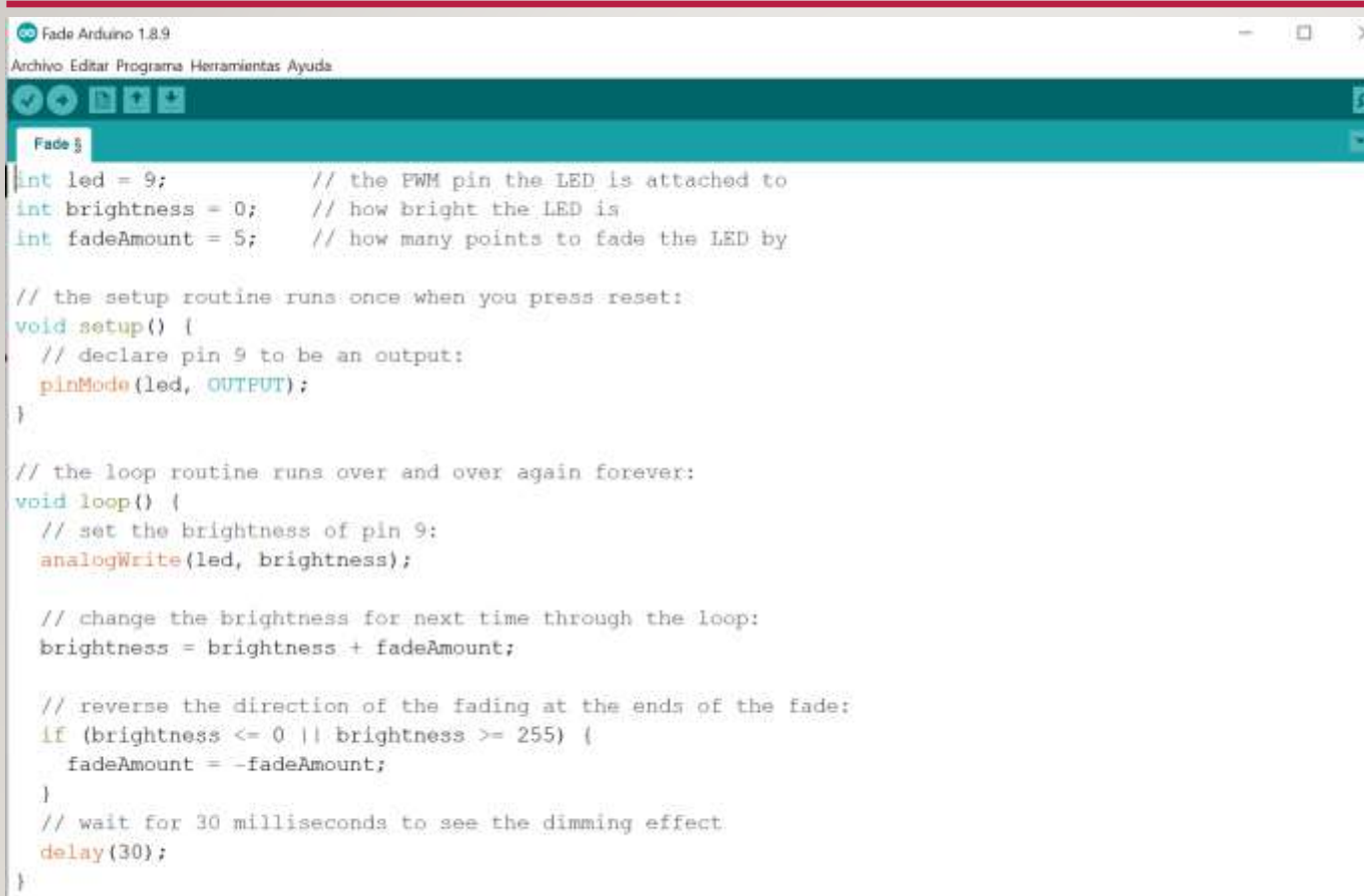
A screenshot of the Arduino IDE interface. The title bar reads "Blink Arduino 1.8.9". The menu bar includes "Archivo", "Editar", "Programa", "Herramientas", and "Ayuda". The toolbar contains icons for opening files, saving, and running. The code editor shows the following code:

```
int led = 13;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

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

EJEMPLO BÁSICO (FADE)

A screenshot of the Arduino IDE interface. The title bar reads "Fade Arduino 1.8.9". Below it is a menu bar with "Archivo", "Editar", "Programa", "Herramientas", and "Ayuda". A toolbar with various icons is visible. The main text area contains the following C++ code:

```
Fade §  
  
int led = 9;          // the PWM pin the LED is attached to  
int brightness = 0;   // how bright the LED is  
int fadeAmount = 5;   // how many points to fade the LED by  
  
// the setup routine runs once when you press reset:  
void setup() {  
  // declare pin 9 to be an output:  
  pinMode(led, OUTPUT);  
}  
  
// the loop routine runs over and over again forever:  
void loop() {  
  // set the brightness of pin 9:  
  analogWrite(led, brightness);  
  
  // change the brightness for next time through the loop:  
  brightness = brightness + fadeAmount;  
  
  // reverse the direction of the fading at the ends of the fade:  
  if (brightness <= 0 || brightness >= 255) {  
    fadeAmount = -fadeAmount;  
  }  
  // wait for 30 milliseconds to see the dimming effect  
  delay(30);  
}
```


EJEMPLO BÁSICO (ANALOGREADSERIAL)

A screenshot of the Arduino IDE interface. The title bar reads "AnalogReadSerial Arduino 1.8.9". The menu bar includes "Archivo", "Editar", "Programa", "Herramientas", and "Ayuda". The toolbar contains icons for opening, saving, and running. The code editor shows the following C++ code:

```
// the setup routine runs once when you press reset:
void setup() {
  // initialize serial communication at 9600 bits per second:
  Serial.begin(9600);
}

// the loop routine runs over and over again forever:
void loop() {
  // read the input on analog pin 0:
  int sensorValue = analogRead(A0);
  // print out the value you read:
  Serial.println(sensorValue);
  delay(1);        // delay in between reads for stability
}
```

PREGUNTAS



TRABAJO INVESTIGATIVO

- Realizar tres programas básicos diferentes a los mencionados en este video.
- Elabora nuevamente el programa “FADE” empleando ciclos For y While.
- Analiza cual es la diferencia entre el uso de delay(mS) y millis().