

SESIÓN 3



MAKERS

TEMAS DE AVANCE

- Programación Básica
- Led
- Pines Digitales (Salida)
- Pines Digitales (Entrada)

TIPOS DE VARIABLES

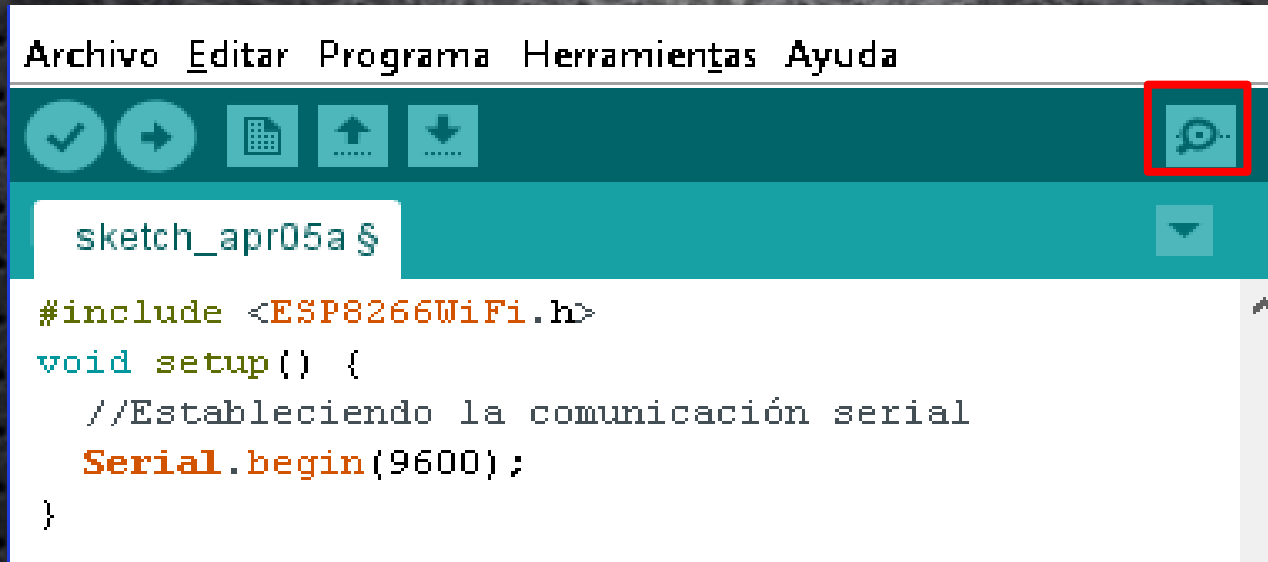
```
1 const int num = 2;  
2 int num2 = 3;  
3 String texto = "Hola Mundo";  
4 char character= 'a';  
5 bool booleano= true;  
6 long numero = 12345678901234567890;  
7 float flotante = 1.2;  
8
```

LIBRERÍAS

```
1 #include <DFPlayer_Mini_Mp3.h>
2 #include <TinyGPS++.h>
3 #include <LiquidCrystal_I2C.h>
4 #include <SimpleTimer.h>
5 #include <DHT.h>
6 #include <DHT_U.h>
7 #include <Servo.h>
```

```
8
```

MONITOR SERIAL



El 9600 indica la cantidad de baudios que manejará el puerto serie en esta ocasión.

Se define baudio como una unidad de medida que representa los números de símbolos por segundo en un medio de transmisión.

COMUNICACIÓN SERIAL

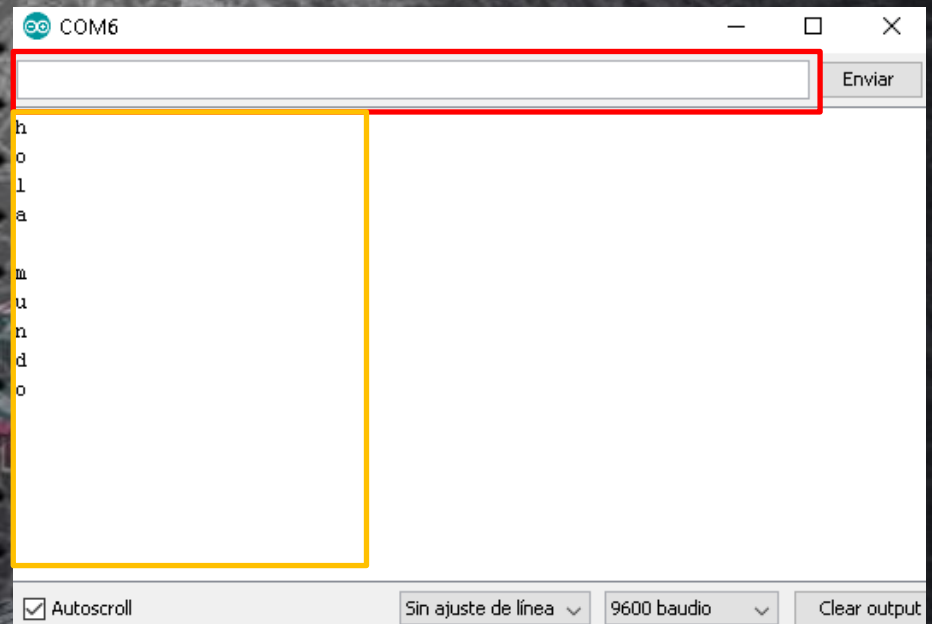
(Entrada de datos)



```
#include <ESP8266WiFi.h>

void setup() {
  //Estableciendo la comunicación serial
  Serial.begin(9600);
}

void loop() {
  //Esperando la lectura de
  if(Serial.available()>0){
    char dato=Serial.read();
    Serial.println(dato);
  }
}
```



En el cuadro de color rojo se deben ingresar los datos, y en el cuadro de color naranja se muestran los datos ingresados.

COMUNICACIÓN SERIAL

(Salida de datos)

```
Archivo  Editar  Programa  Herramientas  Ayuda

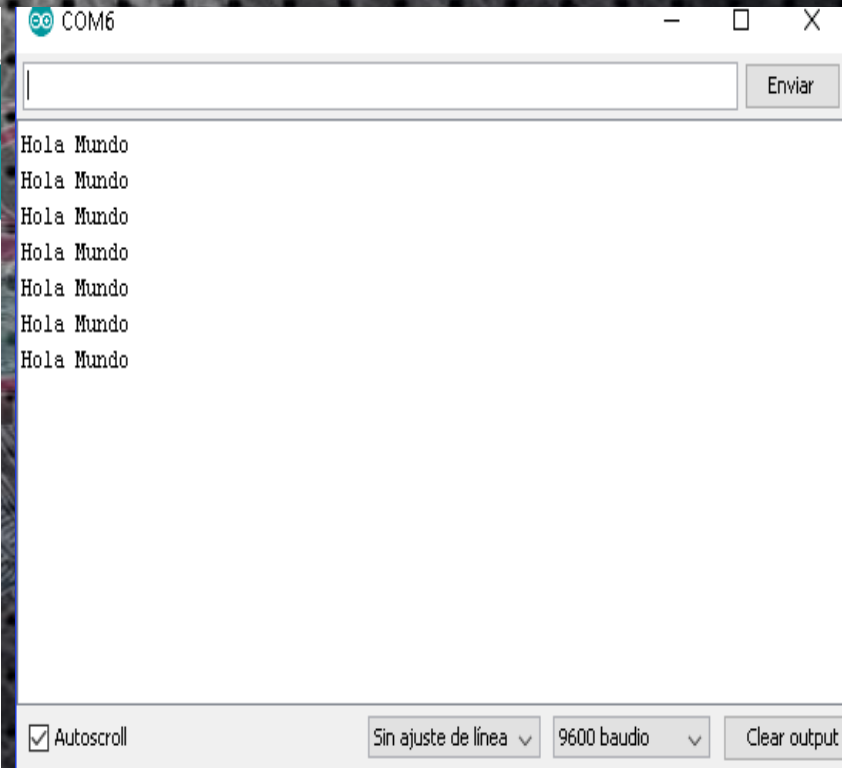
✓ → 📄 ⬆ ⬇

sketch_apr05a $

#include <ESP8266WiFi.h>

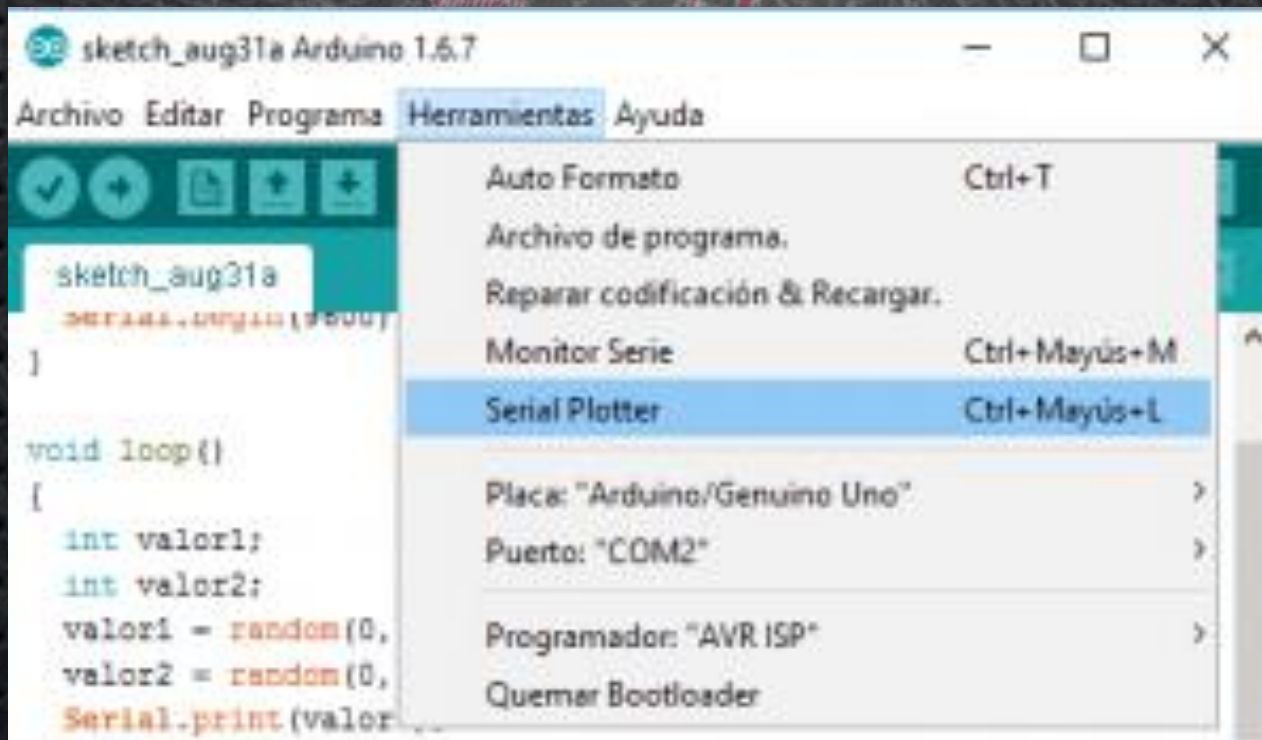
void setup() {
  //Estableciendo la comunicación serial
  Serial.begin(9600);
}

void loop() {
  //Mostrara el mensaje HOLA MUNDO en el monitor serial
  Serial.println("Hola Mundo");
  //El mensaje Hola Mundo se mostrar cada 2segundos
  delay(2000);
}
```



SERIAL PLOTTER

Permite crear gráficas a partir de las variables que le indiquemos.



SERIAL PLOTTER

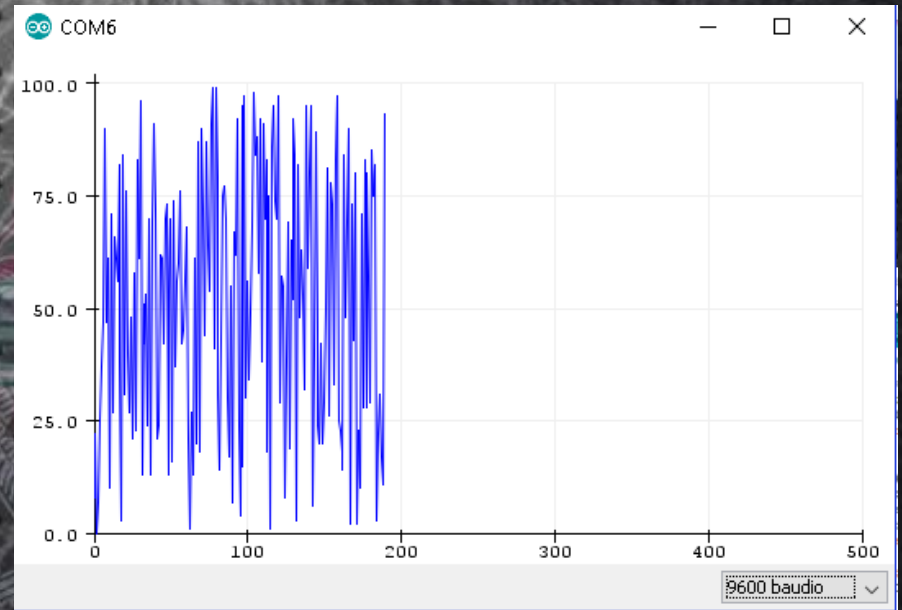


The screenshot shows the Arduino IDE interface. The title bar reads "sketch_apr05a Arduino 1.8.5". The menu bar includes "Archivo", "Editar", "Programa", "Herramientas", and "Ayuda". The toolbar contains icons for checking, running, serial monitor, and uploading. The sketch name "sketch_apr05a" is in the top bar. The code in the editor is as follows:

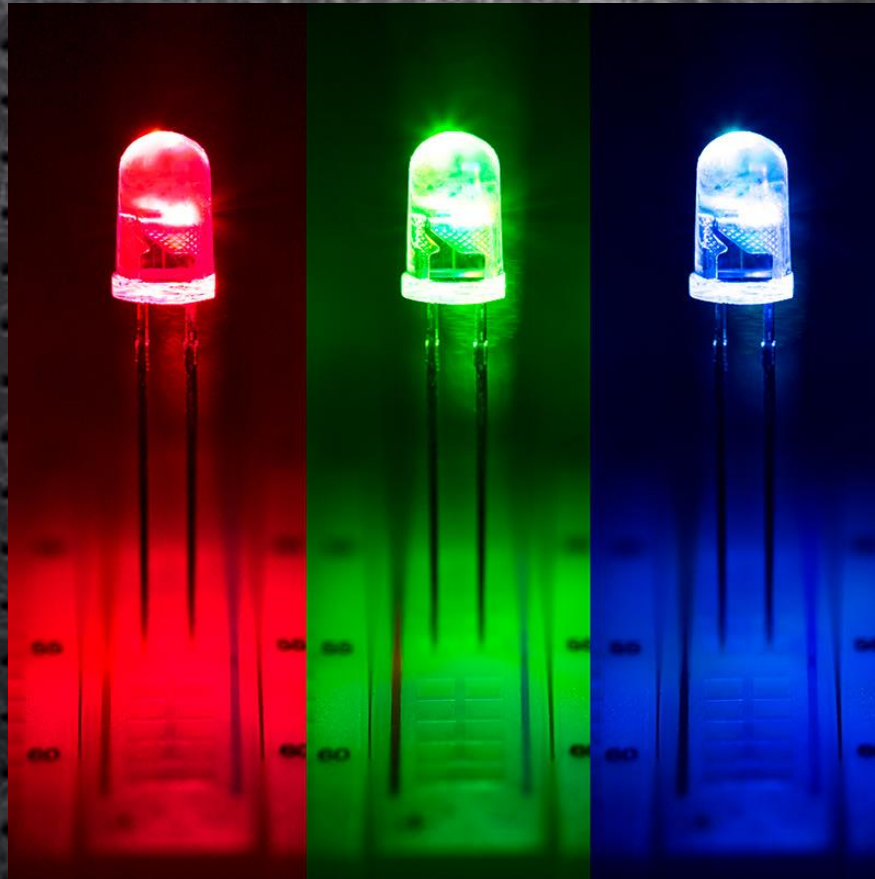
```
#include <ESP8266WiFi.h>

void setup() {
  //Estableciendo la comunicación serial
  Serial.begin(9600);
}

void loop() {
  int valor;
  //Los valores que se almacenaran en la
  //variable valor estaran en el rango de 0 a 100
  valor=random(0,100);
  //Estos valores se podrian ver desde el Plooter
  Serial.println(valor);
  delay(500);
}
```

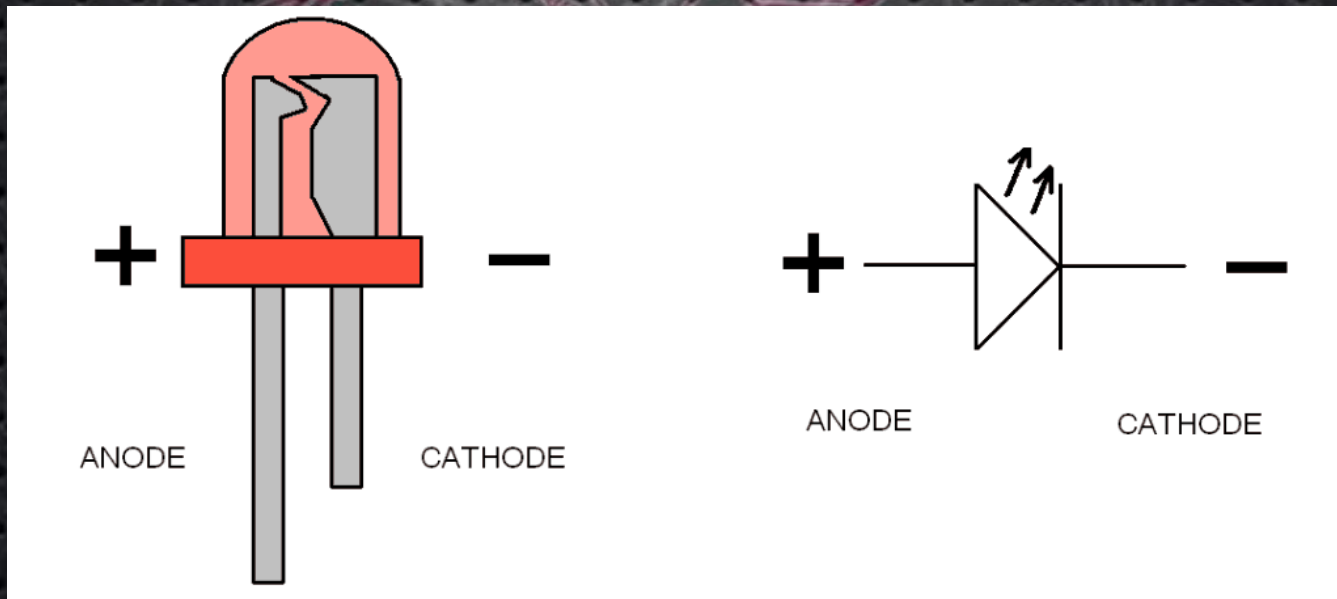


LED



LED

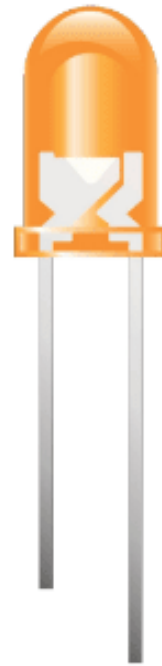
Un led es un componente optoelectrónico pasivo, concretamente un diodo que emite luz.



ÁNODO Y CÁTODO

- **ÁNODO.-** es el electrodo que pierde electrones en una reacción oxidación.
- Normalmente se vincula al polo **positivo** del transito de la corriente eléctrica.
- **CÁTODO.-** es el electrodo con carga negativa, que en la reacción química sufre una reacción de reducción.
- Normalmente se vincula al polo **negativo** del transito de la corriente eléctrica.

PINES DIGITALES (SALIDA)



SINTAXIS

pinMode(pin, tipo);

***Donde pin:** es el número de pin digital (D0, D1, D2, D3, D4, D5, D6, D7, D8).*

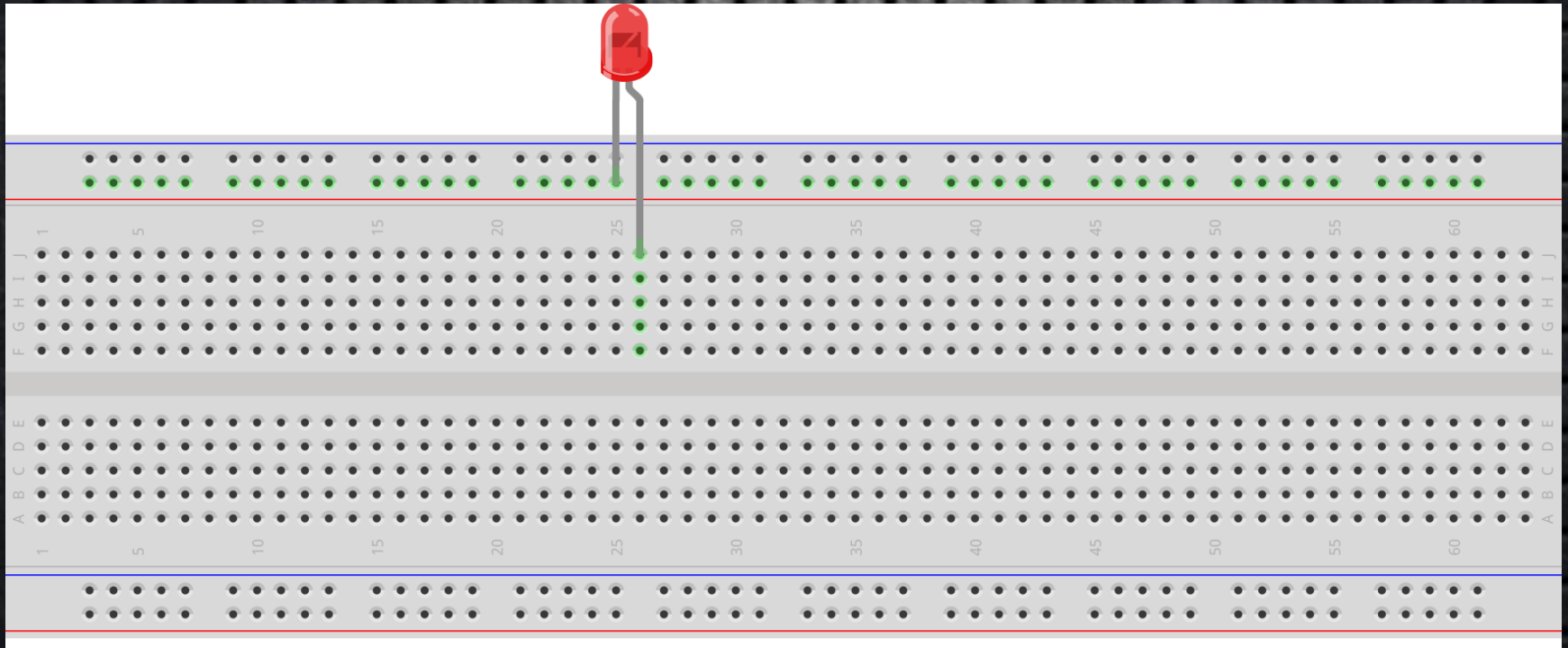
***Tipo:** INPUT o OUTPUT. EN ESTE CASO OUTPUT*

digitalWrite(pin, valor);

***Donde pin:** es el número de pin digital (D0, D1, D2, D3, D4, D5, D6, D7, D8).*

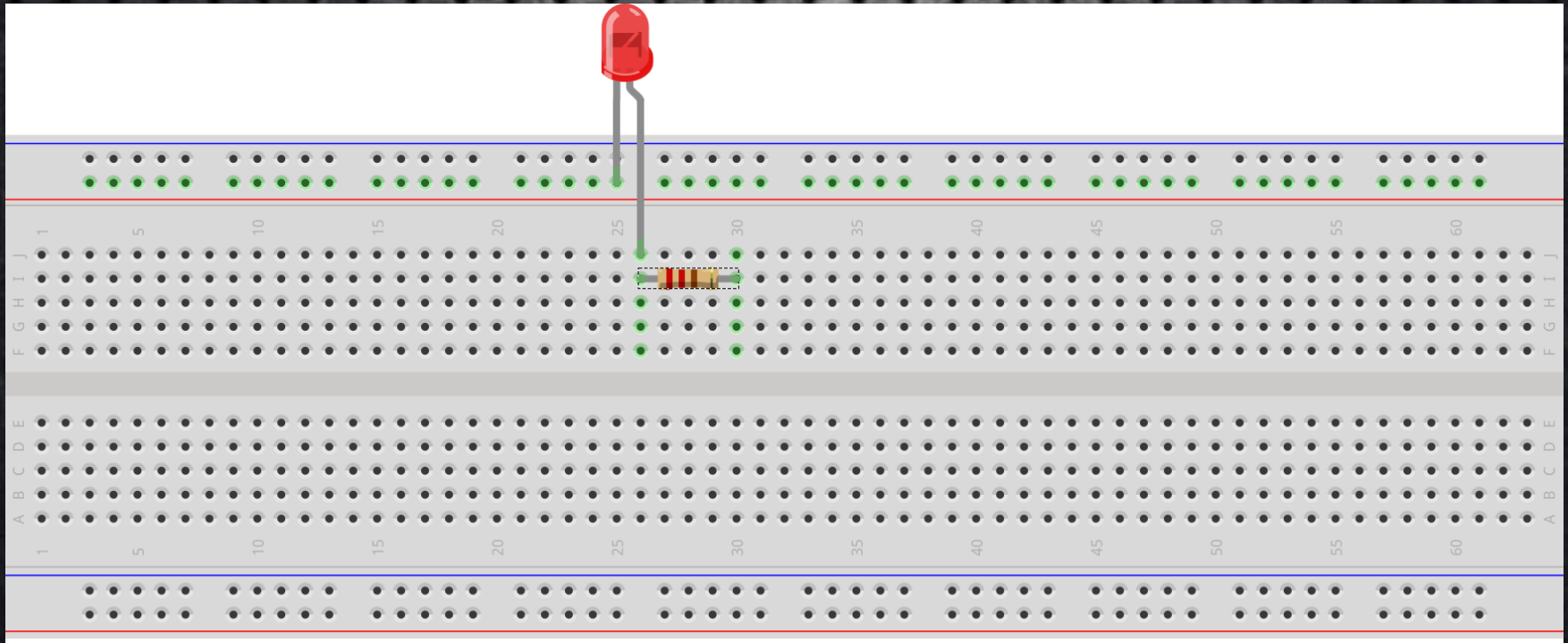
***Valor:** es HIGH (1) o LOW (0).*

PROTOTIPO



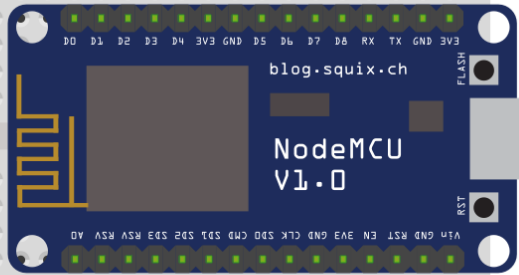
La pata mas larga del LED será quien recibirá la señal del nodemcu , la pata corta estará conectado a tierra (GND).

PROTOTIPO



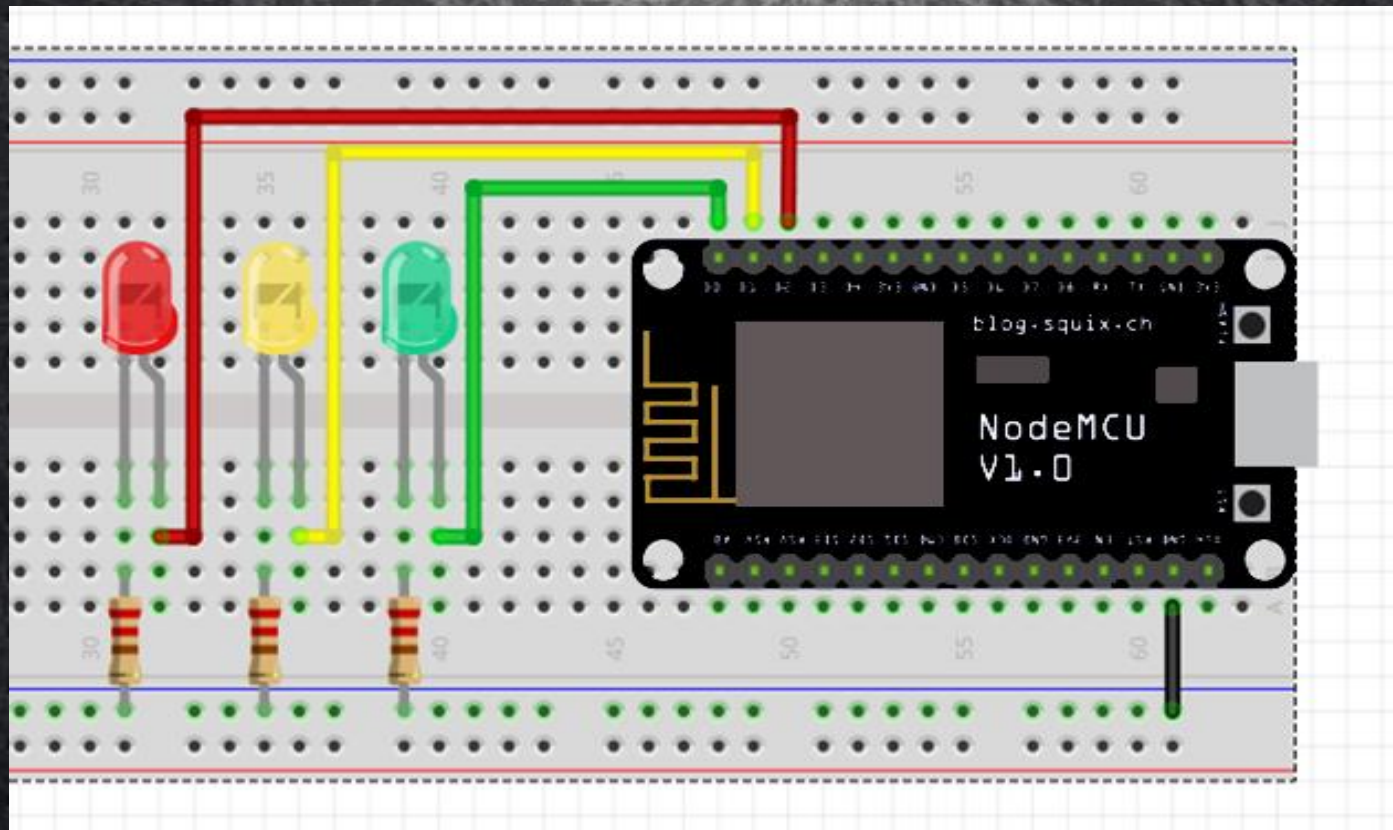
La resistencia a usar es de $220\Omega \pm 5$ (rojo, rojo, café, dorado) , un extremo de la resistencia estará conectado al led y el otro recibirá la señal del nodemcu..

• • • • •



The **MAKERS**
INFORMATION

EJEMPLO SEMÁFORO



CÓDIGO

```
1 int rojo=D2;
2 int amarillo=D1;
3 int verde=D0;
4 void setup() {
5     pinMode(rojo, OUTPUT);
6     pinMode(amarillo, OUTPUT);
7     pinMode(verde, OUTPUT);
8 }
9
10 void loop() {
11     digitalWrite(rojo, HIGH);
12     delay(3000);
13     digitalWrite(rojo, LOW);
14     digitalWrite(verde, HIGH);
15     delay(3000);
16     digitalWrite(verde, LOW);
17     digitalWrite(amarillo, HIGH);
18     delay(1000);
19     digitalWrite(amarillo, LOW);
20 }
```

PINES DIGITALES (ENTRADA)



SINTAXIS

pinMode(pin, tipo);

Donde pin: es el número de pin digital (D0, D1, D2, D3, D4, D5, D6, D7, D8).

Tipo: INPUT o OUTPUT. EN ESTE CASO INPUT

digitalRead(pin);

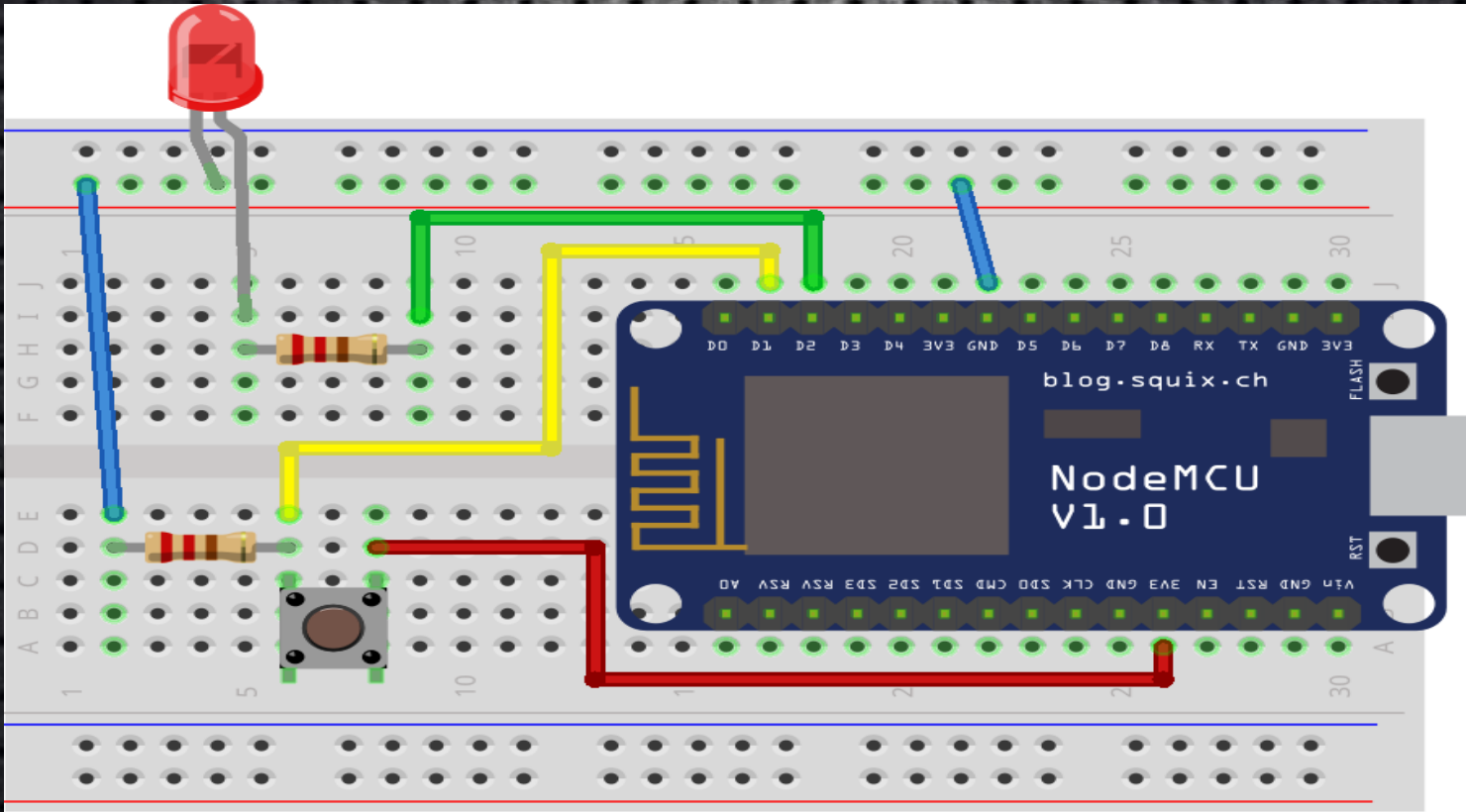
Donde pin: es el número de pin digital (D0, D1, D2, D3, D4, D5, D6, D7, D8).

PULSADOR

- Un interruptor es un dispositivo simple con dos posiciones, EN y AP (ENcendido y APagado).
- Nuestro interruptor (en nuestro caso de dos patas) viene en dos formas:
- Normalmente abierto.- el circuito no hace contacto.
- Normalmente cerrado.- el circuito esta en contacto



ENCENDER UN LED CON UN PULSADOR



CÓDIGO

entradaDigital

```
#include <ESP8266WiFi.h>

void setup() {
    // put your setup code here, to run once:
    pinMode(D2,OUTPUT);
    pinMode(D1,INPUT);
}

void loop() {
    // put your main code here, to run repeatedly:
    if(digitalRead(D1)==HIGH){
        digitalWrite(D2,HIGH);
    }
    else{
        digitalWrite(D2,LOW);
    }
}
```


ACLARACIÓN

Se pudo notar que con el anterior código, el led solo se enciende cuando se mantiene presionado el pulsador. Entonces cuando el pulsador se encuentra presionado manda el estado **HIGH** y cuando se suelta el pulsador manda el estado **LOW**

CÓDIGO

Solucionando el código anterior

```
#include <ESP8266WiFi.h>
int estadoActual=0;
int estadoAuxiliar=0;
int estadoAnterior=0;

void setup() {
  pinMode(D2,OUTPUT);
  pinMode(D1,INPUT);
  Serial.begin(9600);
}

void loop() {
  estadoActual=digitalRead(D1);
  Serial.println(" Estado : ");
  Serial.print(estadoActual);
  Serial.println();
  Serial.println(" Estado Anterior : ");
  Serial.print(estadoAnterior);
  Serial.println();
```

```
if((estadoActual) && (estadoAnterior==LOW)){
  estadoAuxiliar=1-estadoAuxiliar;
  delay(100);
} //Cierre del primer IF
estadoAnterior=estadoActual;
if(estadoAuxiliar==1){
  digitalWrite(D2,HIGH);
} //Cierre del segundo IF
else{
  digitalWrite(D2,LOW);
} //Cierre del ELSE
} //Cierre del LOOP
```




GRACIAS