#include <ESP8266WiFi.h>

#define rojo D6

#define verde D7

#define azul D5

#define max 255

int i=0;

String ssid = "ITZ2022";

String password = "ABCD2022";

WiFiServer server(80); //objeto de la clase WiFiServer

int estado = 0;

void setup() {

 // Definimos los pins

 pinMode(rojo, OUTPUT); // Red: D6

 pinMode(verde, OUTPUT); // Green: D7

 pinMode(azul, OUTPUT); // Blue: D5

 // Inicia Serial

 Serial.begin(115200);

 Serial.println("\n");

 color(0,0,0);

 // Conexión WIFI

 WiFi.begin(ssid, password);

 while (WiFi.status() != WL\_CONNECTED ) {

 color(max,0,0);

 delay(250);

 color(0,0,0);

 delay(250);

 Serial.print(".");

 }

 color(0,max,0);

 delay(3000);

 color(0,0,0);

 Serial.println("");

 Serial.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

 Serial.print("Conectado a la red WiFi: ");

 Serial.println(WiFi.SSID());

 Serial.print("IP: ");

 Serial.println(WiFi.localIP());

 Serial.print("macAdress: ");

 Serial.println(WiFi.macAddress());

 Serial.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

 server.begin(); //begin() levantamos el servidor

}

void color (int R, int G, int B) {

 analogWrite(rojo, max-R);

 analogWrite(verde, max-G);

 analogWrite(azul, max-B);

}

void loop() {

 WiFiClient client = server.available(); //objeto de la clase WiFiClient

 // avalaible() detecta un cliente nuevo del objeto de la clase WifiServer

 if(!client){

 return;

 }

 Serial.println("Nuevo cliente...");

 while(!client.available()){ //espera a un cliente diponible

 delay(1);

 }

 String peticion = client.readStringUntil('\r'); //lee la peticion del cliente

 Serial.println(peticion);

 client.flush(); //limpia la peticion del cliente

 if(peticion.indexOf("LED=AZUL") != -1)

 {

 color(0,0,max);

 }

 if(peticion.indexOf("LED=ROJO") != -1)

 {

 color(max,0,0);

 }

 if(peticion.indexOf("LED=VERDE") != -1)

 {

 color(0,max,0);

 }

 if(peticion.indexOf("LED=OFF") != -1)

 {

 color(0,0,0);

 }

 if(peticion.indexOf("LED=EFECTO1") != -1)

 {

 for(i=1;i<=5;i++){

 delay(150);

 color(0,max,0);

 delay(150);

 color(0,0,0);

 }

 for(i=1;i<=5;i++){

 delay(150);

 color(max,0,0);

 delay(150);

 color(0,0,0);

 }

 for(i=1;i<=5;i++){

 delay(150);

 color(0,0,max);

 delay(150);

 color(0,0,0);

 }

 }

 if(peticion.indexOf("LED=EFECTO2") != -1)

 {

 for(i=1;i<=5;i++){

 color(0,max,0);

 delay(350);

 color(0,0,0);

 delay(150);

 color(max,0,0);

 delay(350);

 color(0,0,0);

 delay(150);

 color(0,0,max);

 delay(350);

 color(0,0,0);

 delay(150);

 }

 }

 client.println("HTTP/1.1 200 OK");

 client.println("");

 client.println("");

 client.println("");

 client.println("");

 delay(10);

 Serial.println("Peticion finalizada");

 Serial.println("");

}

#include <ESP8266WiFi.h>

String ssid = "ITZACATEPEC";

String password = "";

byte cont = 0;

byte max\_intentos = 50;

void setup() {

 // Inicia Serial

 Serial.begin(115200);

 Serial.println("\n");

 // Conexión WIFI

 WiFi.begin(ssid, password);

 while (WiFi.status() != WL\_CONNECTED and cont < max\_intentos) { //Cuenta hasta 50 si no se puede conectar lo cancela

 cont++;

 delay(500);

 Serial.print(".");

 }

Serial.println("");

 if (cont < max\_intentos) { //Si se conectó

 Serial.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

 Serial.print("Conectado a la red WiFi: ");

 Serial.println(WiFi.SSID());

 Serial.print("IP: ");

 Serial.println(WiFi.localIP());

 Serial.print("macAdress: ");

 Serial.println(WiFi.macAddress());

 Serial.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

 }

 else { //No se conectó

 Serial.println("------------------------------------");

 Serial.println("Error de conexion");

 Serial.println("------------------------------------");

 }

}

void loop() {

}

void setup() {

// inicializar GPIO 2 como salida

pinMode(miLed, OUTPUT);

}

// la función loop corre continuamente

void loop() {

// enciende el LED (HIGH es ALTO y es el nivel de voltaje)

digitalWrite(miLed, HIGH);

delay(1000); // espera 1000 milisegundos = un segundo

digitalWrite(miLed, LOW);

// apaga el LED (LOW es BAJO y es el nivel de voltaje)

delay(1000); // espera 1000 milisegundos = un segundo

}