**WEEK 3 ASSIGNMENT**

|  |  |  |
| --- | --- | --- |
| NAME:JAYANTH R | REG NO: 20BEC1176 | INTERN DOMAIN: IOT |

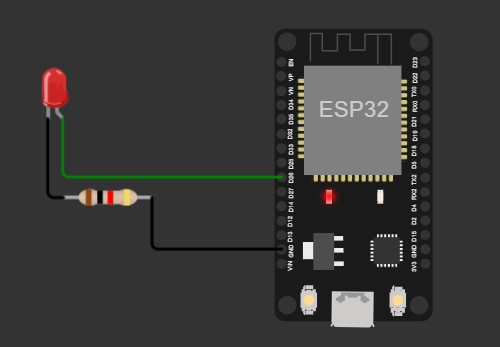
**TASK:**

To make led glow using Node red and IBM cloud

**WOKWI LINK:**

**https://wokwi.com/projects/366873520832747521**

**Circuit Diagram:**



**Code:**

#include <WiFi.h>//library for wifi

#include <PubSubClient.h>//library for MQtt

#define LED 26

void callback(char\* subscribetopic, byte\* payload, unsigned int payloadLength);

//-------credentials of IBM Accounts------

#define ORG "bn7nog"//IBM ORGANITION ID

#define DEVICE\_TYPE "abcd"//Device type mentioned in ibm watson IOT Platform

#define DEVICE\_ID "1234"//Device ID mentioned in ibm watson IOT Platform

#define TOKEN "12345678"     //Token

String data3;

//-------- Customise the above values --------

char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server Name

char publishTopic[] = "iot-2/evt/Data/fmt/json";// topic name and type of event perform and format in which data to be send

char subscribetopic[] = "iot-2/cmd/command/fmt/String";// cmd  REPRESENT command type AND COMMAND IS TEST OF FORMAT STRING

char authMethod[] = "use-token-auth";// authentication method

char token[] = TOKEN;

char clientId[] = "d:" ORG ":" DEVICE\_TYPE ":" DEVICE\_ID;//client id

//-----------------------------------------

WiFiClient wifiClient; // creating the instance for wificlient

PubSubClient client(server, 1883, callback ,wifiClient); //calling the predefined client id by passing parameter like server id,portand wificredential

void setup() {

**Serial**.begin(115200);

  pinMode(LED,OUTPUT);

  delay(10);

**Serial**.println();

  wificonnect();

  mqttconnect();

}

void loop() {

  delay(1000);

  if (!client.loop()) {

    mqttconnect();

  }

}

void mqttconnect() {

  if (!client.connected()) {

**Serial**.print("Reconnecting client to ");

**Serial**.println(server);

    while (!!!client.connect(clientId, authMethod, token)) {

**Serial**.print(".");

      delay(500);

    }

    initManagedDevice();

**Serial**.println();

  }

}

void wificonnect() {//function defination for wificonnect

**Serial**.println();

**Serial**.print("Connecting to ");

  WiFi.begin("Wokwi-GUEST", "", 6);//passing the wifi credentials to establish the connection

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

    delay(500);

**Serial**.print(".");

  }

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

**Serial**.println("WiFi connected");

**Serial**.println("IP address: ");

**Serial**.println(WiFi.localIP());

}

void initManagedDevice() {

  if (client.subscribe(subscribetopic)) {

**Serial**.println((subscribetopic));

**Serial**.println("subscribe to cmd OK");

  }

  else {

**Serial**.println("subscribe to cmd FAILED");

  }

}

void callback(char\* subscribetopic, byte\* payload, unsigned int payloadLength) {

**Serial**.print("callback invoked for topic: ");

**Serial**.println(subscribetopic);

  for (int i = 0; i < payloadLength; i++) {

    data3 += (char)payload[i];

  }

**Serial**.println("data: "+ data3);

  if(data3=="lighton") {

**Serial**.println(data3);

    digitalWrite(LED,HIGH);

  }

  else {

**Serial**.println(data3);

    digitalWrite(LED,LOW);

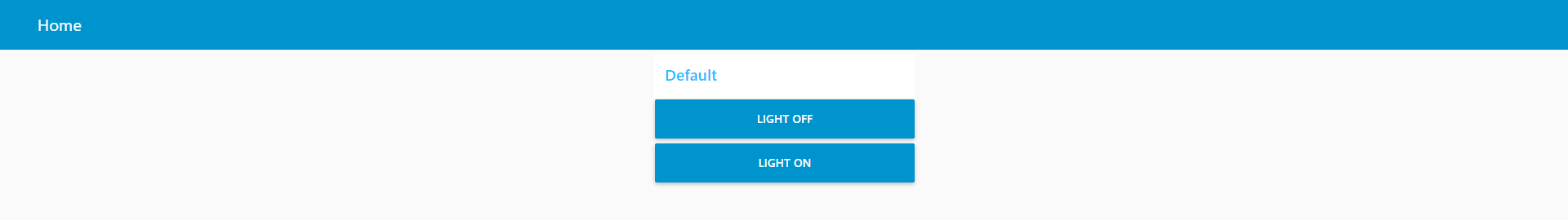
  }

  data3="";

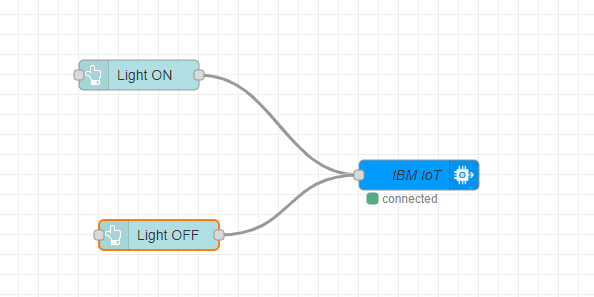
}

**DASHBOARD:**

**(NODE RED)**

****

**(NODE RED)**

****

**RESULT:** Given task was carried out successfully.