Air quality management

Program(python):

#define BLYNK\_TEMPLATE\_ID "TMPLwToQUqRw"

#define BLYNK\_TEMPLATE\_NAME "Air Quality Management"

#define BLYNK\_AUTH\_TOKEN "C8Y7T0Fr54QF8pdfQ5dZsdfhhSdiQBFLj8mYe"

#define BLYNK\_PRINT **Serial**

#include <WiFi.h>

#include <BlynkSimpleEsp32.h>

#include <DHT.h>

#include <LiquidCrystal\_I2C.h>

LiquidCrystal\_I2C lcd(0x27, 16, 2);

  byte degree\_symbol[8] =

              {

                0b00111,

                0b00101,

                0b00111,

                0b00000,

                0b00000,

                0b00000,

                0b00000,

                0b00000

              };

char auth[] = BLYNK\_AUTH\_TOKEN;

char ssid[] = "WiFi Username";  // type your wifi name

char pass[] = "WiFi Password";  // type your wifi password

BlynkTimer timer;

int gas = 32;

int sensorThreshold = 100;

#define DHTPIN 2 //Connect Out pin to D2 in NODE MCU

#define DHTTYPE DHT11

DHT dht(DHTPIN, DHTTYPE);

void sendSensor()

{

  float h = dht.readHumidity();

  float t = dht.readTemperature(); // or dht.readTemperature(true) for Fahrenheit

     if (isnan(h) || isnan(t)) {

**Serial**.println("Failed to read from DHT sensor!");

    return;

  }

   int analogSensor = analogRead(gas);

  Blynk.virtualWrite(V2, analogSensor);

**Serial**.print("Gas Value: ");

**Serial**.println(analogSensor);

  // You can send any value at any time.

  // Please don't send more that 10 values per second.

    Blynk.virtualWrite(V0, t);

    Blynk.virtualWrite(V1, h);

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

**Serial**.print(t);

**Serial**.print("    Humidity : ");

**Serial**.println(h);

}

void setup()

{

**Serial**.begin(115200);

 //pinMode(gas, INPUT);

  Blynk.begin(auth, ssid, pass);

  dht.begin();

  timer.setInterval(30000L, sendSensor);

 //Wire.begin();

   lcd.begin(16,2);

//  lcd.backlight();

 // lcd.clear();

  lcd.setCursor(3,0);

  lcd.print("Air Quality");

  lcd.setCursor(3,1);

  lcd.print("Management");

  delay(2000);

  lcd.clear();

  }

Output:

