**BÁO CÁO TUẦN BÀI TẬP 5**

1. **HTTP Get**
   1. **Khai báo các thư viện và các giá trị khởi tạo**

#include <LiquidCrystal\_I2C.h>

#include "DHTesp.h"

#include <WiFi.h>

#include <HTTPClient.h>

#include <ArduinoJson.h>

LiquidCrystal\_I2C lcd(0x27,20,4);

#define DHT\_PIN 15

#define LED\_PIN 26

#define PIR\_PIN 13

DHTesp dhtSensor;

const String serverName = "HTTP://postman-echo.com/get";

void setup() {

  Serial.begin(115200);

  Serial.println("Connecting to WiFi");

  WiFi.begin("Wokwi-GUEST", "",6);

  // put your setup code here, to run once:

  dhtSensor.setup(DHT\_PIN, DHTesp::DHT22);

  pinMode(LED\_PIN, OUTPUT);

  pinMode(PIR\_PIN, INPUT);

}

* 1. **Thực hiện vòng lặp và thực hiện lấy nhiệt độ và độ ẩm từ cảm biến bằng các phương thức đã được cài đặt được trình bày cụ thể trong bài tập tuần 4 sau đó sẽ tạo một request get lên sever bằng cách kết hợp các giá trị nhiệt độ và độ ẩm từ sensor.**

int value = 1 ;

void loop() {

  HTTPClient HTTP;

  TempAndHumidity data = dhtSensor.getTempAndHumidity();

  int temp = data.temperature;

  int humid = data.humidity;

  if (WiFi.status() ==  WL\_CONNECTED){

    String params = "?temp=" + String(temp) +"&humid=" + String(humid);

    String serverPath = serverName + params;

    HTTP.begin(serverPath.c\_str());

    int HTTPResponseCode = HTTP.GET();

    if (HTTPResponseCode == 200){

      String payload  = HTTP.getString();

      Serial.println(payload);

    } else{

      Serial.println("error");

      Serial.println(HTTPResponseCode);

    }

    delay(3000);

  }

1. **Kết quả thực hiện chương trình với HTTP get.**

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Phần mềm đa phương tiện

Mô tả được tạo tự động

1. **Link bài mô phỏng**

[HTTPGet Lê Anh Vũ - Wokwi ESP32, STM32, Arduino Simulator](https://wokwi.com/projects/380339619700505601)

1. **HTTP Post**
   1. **Khai báo các thư viện**

Phần này hoàn toàn giống với trong phần 1 phương thức HTTP get ( Nên không trình bày lại)

**2.2 Thực hiện vòng lặp và lấy các giá trị nhiệt độ và độ ẩm như trong phần trước phần này chỉ khác ở cách thực hiện HTTP post.**

void loop() {

  HTTPClient HTTP;

  TempAndHumidity data = dhtSensor.getTempAndHumidity();

  int temp = data.temperature;

  int humid = data.humidity;

  if (WiFi.status() ==  WL\_CONNECTED){

    HTTP.begin(serverName);

    HTTP.addHeader("Content-Type", "application/x-www-form-urlencoded");

    String HTTPRequestData = "&temp=" + String(temp) + "&humid=" + String(humid);

**Serial**.println(HTTPRequestData);

    int HTTPResponseCode = HTTP.POST(HTTPRequestData);

    if (HTTPResponseCode == 200){

      String payload  = HTTP.getString();

**Serial**.println(payload);

    } else{

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

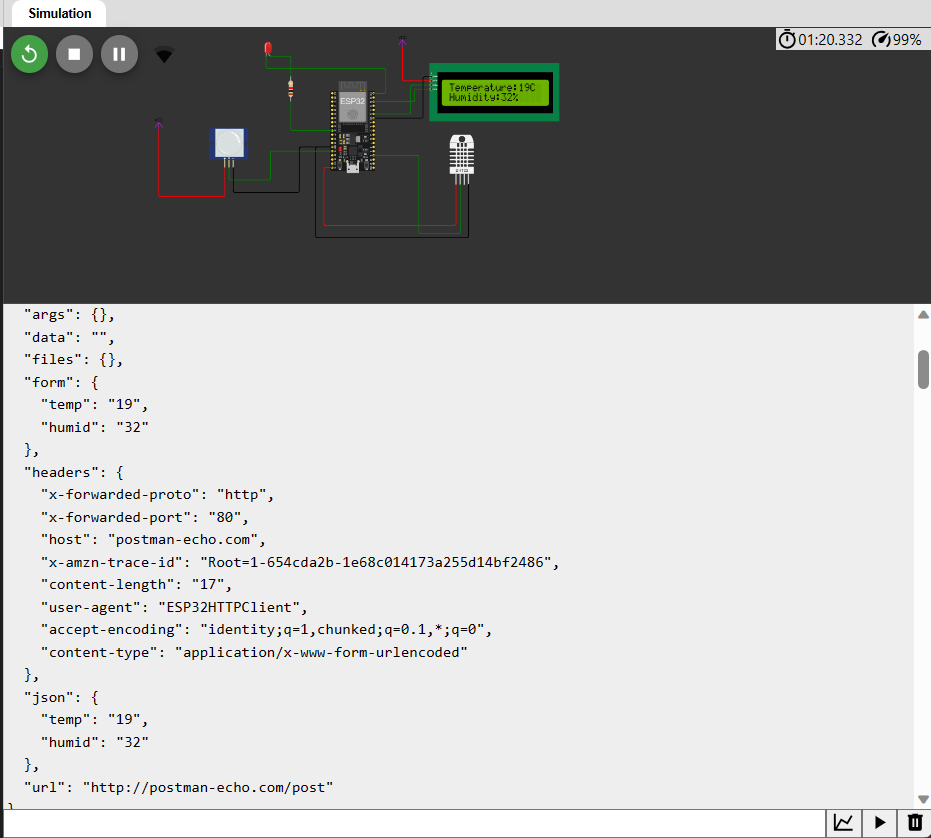
**Serial**.println(HTTPResponseCode);

    }

    delay(3000);

  }

* 1. **Kết quả mô phỏng trên wokwi**



* 1. **Link mô phỏng HTTP post**

[HTTPPost Lê Anh Vũ - Wokwi ESP32, STM32, Arduino Simulator](https://wokwi.com/projects/380922161513781249)

1. **HTTP post Json**
   1. **Khai báo thư viện và các giá trị khởi tạo tương tự như hai phần trước chỉ cần thêm một thư viện khác để làm việc với file json.**
   2. **Thực hiện vòng lặp để lấy nhiệt độ và độ ẩm từ cảm biến sau đó đóng gói trong file json bằng thư viện và thực hiện gửi đi.**
2. void loop() {
3. HTTPClient http;
4. TempAndHumidity data = dhtSensor.getTempAndHumidity();
5. int temp = data.temperature;
6. int humid = data.humidity;
7. if (WiFi.status() ==  WL\_CONNECTED){
8. http.begin(serverName);
9. http.addHeader("Content-Type", "application/json");
10. DynamicJsonDocument doc(1024);
11. String jsonstr;
12. JsonObject root = doc.to<JsonObject>();
13. root["temperature"] = temp;
14. root["humidity"] = humid;
15. serializeJson(doc, jsonstr);
16. String httpRequestData = jsonstr;
17. Serial.println(httpRequestData);
18. int httpResponseCode = http.POST(httpRequestData);
19. if (httpResponseCode == 200){
20. String payload  = http.getString();
21. Serial.println(payload);
22. } else{
23. Serial.println("error");
24. Serial.println(httpResponseCode);
25. }
26. delay(3000);
27. }

**3.3 Kết quả mô phỏng trên wokwi**

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Biểu tượng máy tính

Mô tả được tạo tự động

* 1. **Link bài mô phỏng với HTTP post json.**

[HTTPPostJson Lê Anh Vũ - Wokwi ESP32, STM32, Arduino Simulator](https://wokwi.com/projects/380921788926369793)