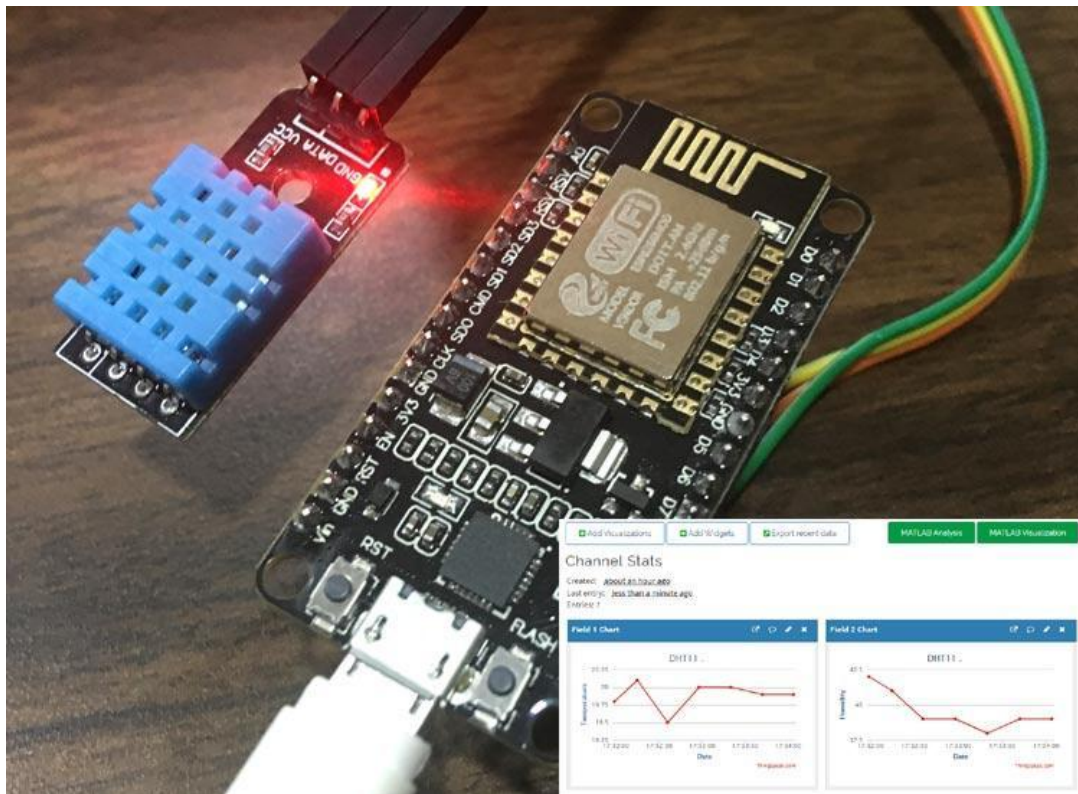


# IoT Based Temperature and Humidity Monitoring using ThingSpeak and ESP8266



**ESP8266 Wi-Fi transceiver** is one of the most popular WiFi module for IoT based Applications. In this project we will use it to connect with ThingSpeak IoT cloud Platform.

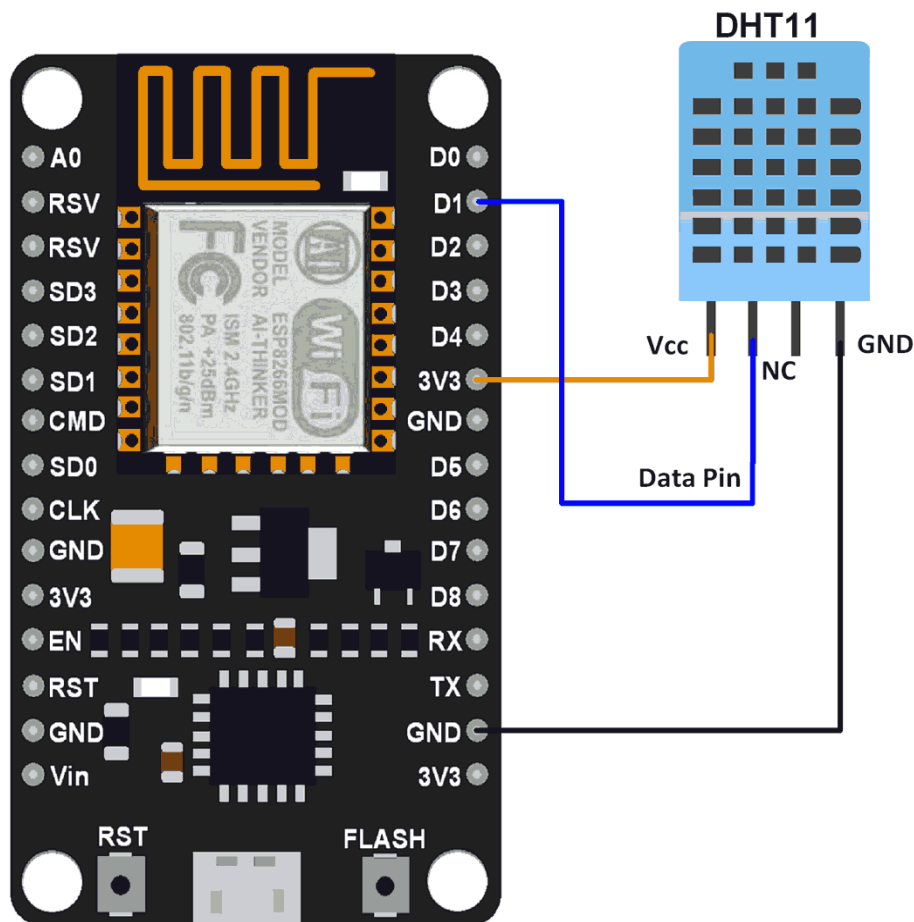
In this project, we are going to send **Temperature and Humidity sensor data** to Thingspeak using **DHT11**. By this method we can monitor our DHT11 sensor's temperature and humidity data over internet using ThingSpeak IOT server, and we can view the logged data and graph over time on the ThingSpeak dashboard. NodeMCU reads the current temperature and humidity from DHT11 and sends it to ThingSpeak server for live monitoring from anywhere in the world.

**ThingSpeak** is a data platform for monitoring your data online, targeted to be used for IoT applications. In Thingspeak channel you can set the data as private or public according to your choice. ThingSpeak takes minimum of 15 seconds to update your readings. Its a great and very easy to use platform for building IOT projects.

## Components Required

- NodeMCU
- DHT11 Temperature and Humidity Sensor
- Jumper Wires

## Circuit Diagram



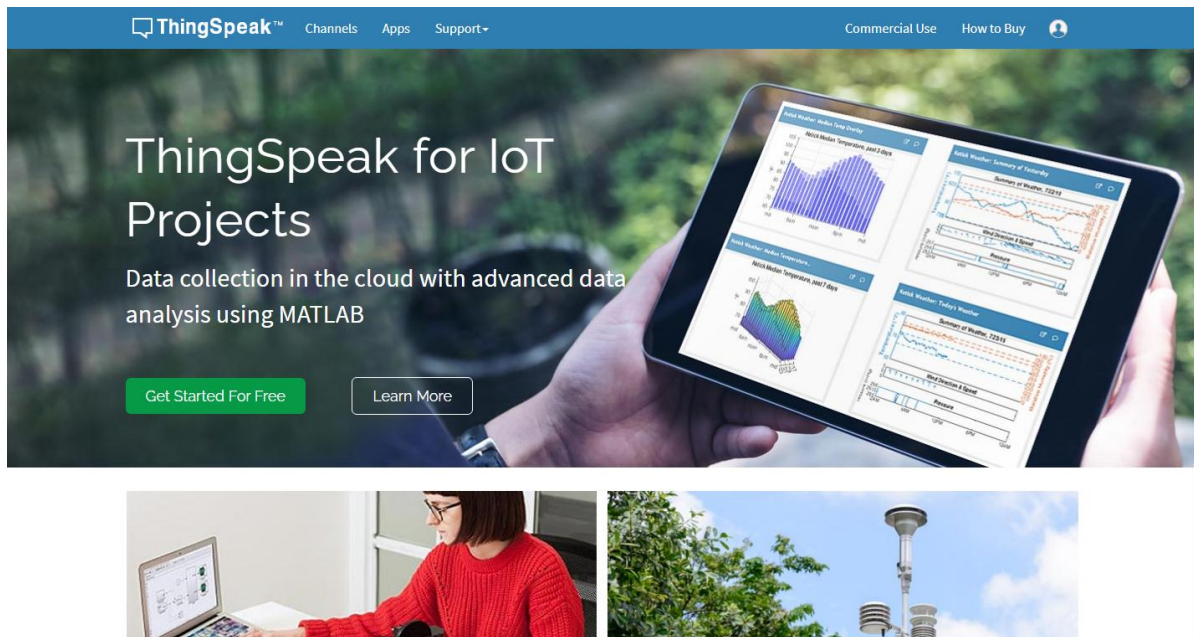
The circuit diagrams above shows the ESP8266 and DHT11 connections. Follow the connections suggested in the below table:

S.NO.	NodeMCU	DHT11
1	Vin	VCC
2	Data	D1
3	GND	GND

### Step 1: ThingSpeak Setup for Temperature and Humidity Monitoring

For creating your channel on ThingSpeak you first need to sign up on ThingSpeak. In case if you already have account on ThingSpeak just sign in using your id and password.

For creating your account go to [www.thingspeak.com](http://www.thingspeak.com)



Click on signup if you don't have account and if you already have account click on sign in.  
After clicking on signup fill your details.

Create MathWorks Account

Email Address

Missing required information

**i** To access your organization's MATLAB license, use your school or work email.

Location

First Name

Last Name

Continue

After this verify your E-mail id and click on continue.

## Step 2: Create a Channel for Your Data

Once you Sign in after your account verification, Create a new channel by clicking “New Channel” button

Name	DHT11	
Description	To Send sensor Data	
Field 1	Temperature	<input checked="" type="checkbox"/>
Field 2	Humidity	<input checked="" type="checkbox"/>
Field 3		<input type="checkbox"/>
Field 4		<input type="checkbox"/>
Field 5		<input type="checkbox"/>

- **Channel Name:** Enter a unique name for the ThingSpeak channel.
- **Description:** Enter a description of the ThingSpeak channel.
- **Field1:** Check the box to enable the field, and enter a field name. Each ThingSpeak channel can have up to 8 fields.
- **Metadata:** Enter information about channel data, including JSON, XML, or CSV data.
- **Tags:** Enter keywords that identify the channel. Separate tags with commas.
- **Link to External Site:** If you have a website that contains information about your ThingSpeak channel, specify the URL.
- **Show Channel Location:**
  - **Latitude:** Specify the latitude position in decimal degrees. For example, the latitude of the city of London is 51.5072.

Enter the name of your data 'Temperature' in Field1 and 'Humidity' in Field2. If you want to use more than one Field you can check the box next to Field option and enter the name and description of your data.

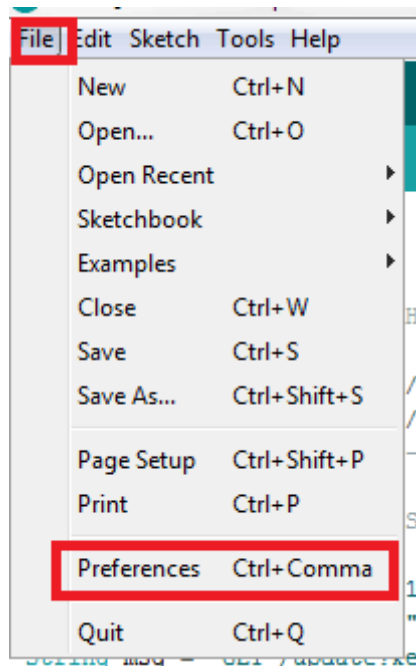
### Step 3: API Key

Click on “API Keys” button to get your unique API key for uploading DHT11 sensor data.

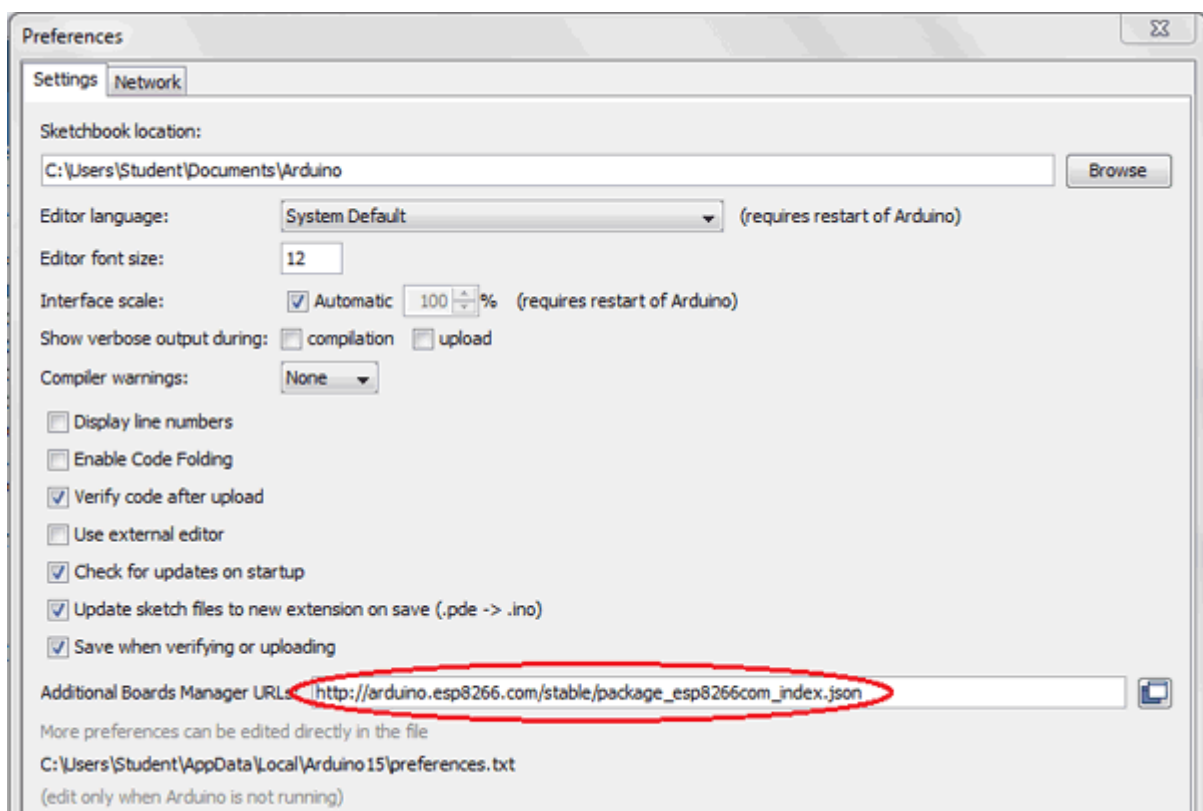
Now copy your “Write API Key”. We will use this API key in our code.

## Programming NodeMCU with Arduino IDE

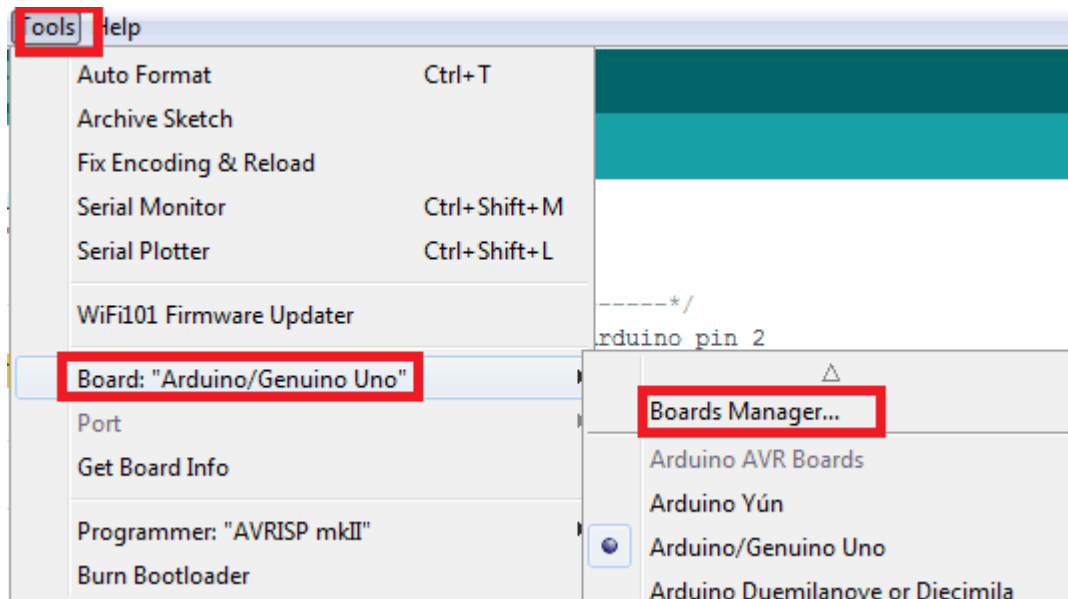
To program NodeMCU with Arduino IDE go to File→Preferences→Settings.



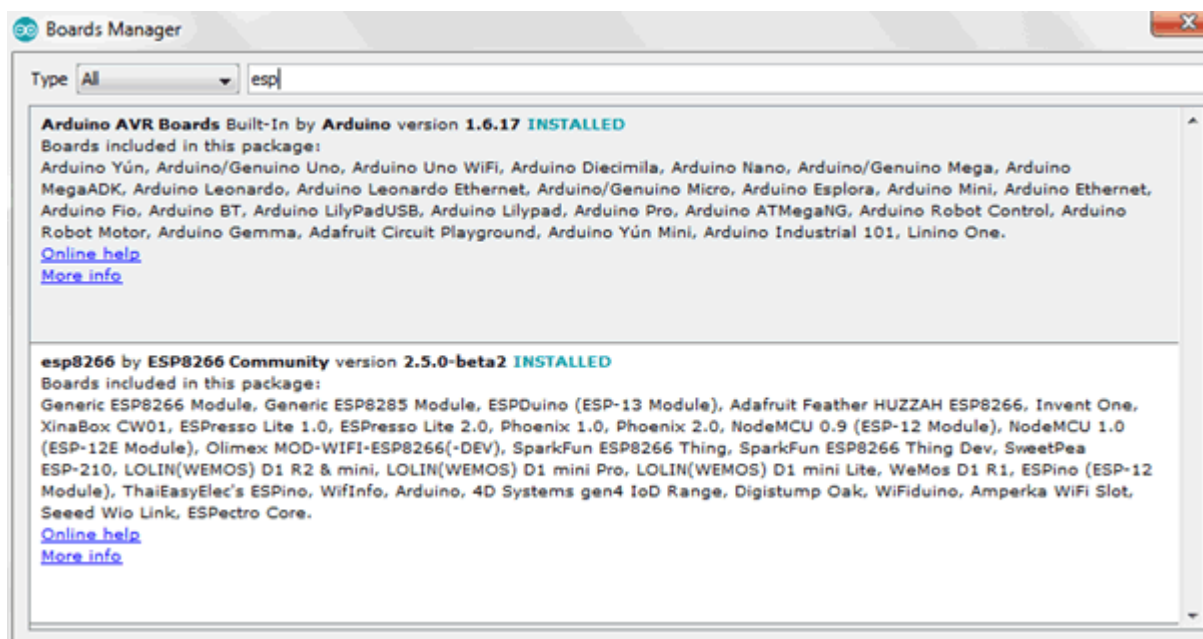
Enter [http://arduino.esp8266.com/stable/package\\_esp8266com\\_index.json](http://arduino.esp8266.com/stable/package_esp8266com_index.json) into 'Additional Board Manager URL' field and click 'Ok'.



Now go to Tools > Board > Boards Manager.



In Boards Manager window, Type esp in the search box, esp8266 will be listed there below. Now select latest version of board and click on install.



After installation is complete, go to Tools >Board >and select NodeMCU 1.0(ESP-12E Module). Now you can program NodeMCU with Arduino IDE.

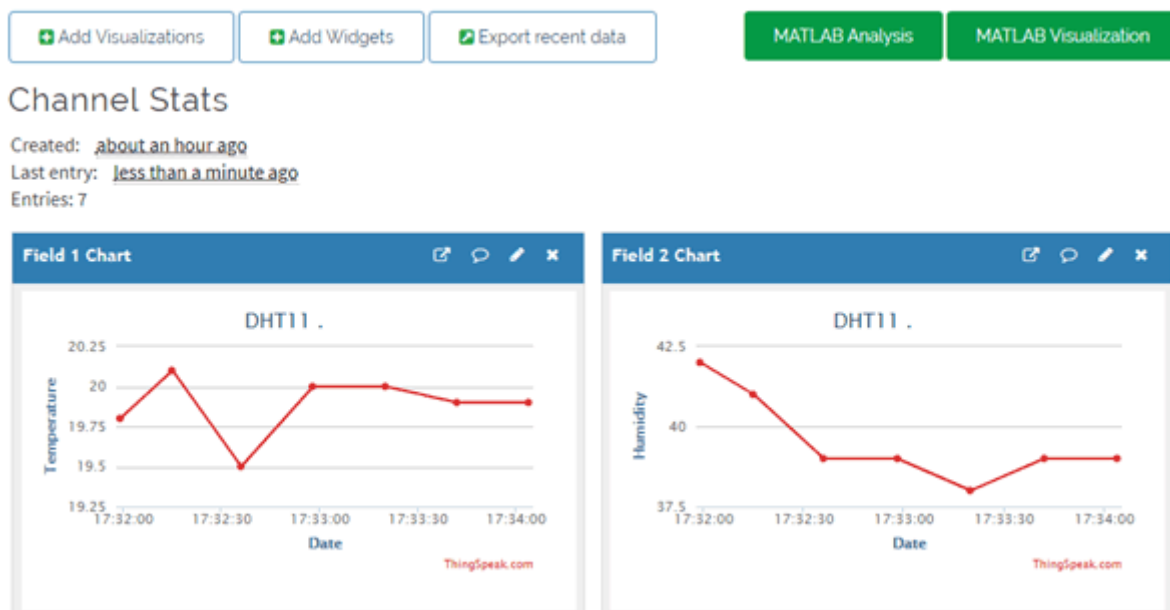
Programming ESP8266 for uploading data to ThingSpeak

Complete code is given at the end of this practical. Upload it in ESP8266 NodeMCU. If you successfully upload your program, Serial monitor will look like this:



```
.
WiFi connected
Temperature: 20.30 degrees Celcius, Humidity: 41.00%. Send to Thingspeak.
Waiting...
Temperature: 20.10 degrees Celcius, Humidity: 41.00%. Send to Thingspeak.
Waiting...
Temperature: 20.10 degrees Celcius, Humidity: 40.00%. Send to Thingspeak.
Waiting...
Temperature: 19.50 degrees Celcius, Humidity: 39.00%. Send to Thingspeak.
Waiting...
Temperature: 20.20 degrees Celcius, Humidity: 40.00%. Send to Thingspeak.
Waiting...
Temperature: 20.00 degrees Celcius, Humidity: 39.00%. Send to Thingspeak.
Waiting...
Temperature: 20.10 degrees Celcius, Humidity: 39.00%. Send to Thingspeak.
Waiting...
Temperature: 20.00 degrees Celcius, Humidity: 38.00%. Send to Thingspeak.
Waiting...
Temperature: 20.00 degrees Celcius, Humidity: 38.00%. Send to Thingspeak.
Waiting...
Temperature: 19.90 degrees Celcius, Humidity: 39.00%. Send to Thingspeak.
Waiting...
```

After this, navigate to your Thingspeak page and open your channel in ThingSpeak and output will be shown as below.



Code

```
#include <DHT.h>
#include <ESP8266WiFi.h>

String apiKey = "Your API KEY"; // Enter your Write API key here
const char *ssid = "WiFi Name"; // Enter your WiFi Name
const char *pass = "WiFi Password"; // Enter your WiFi Password
```

```

const char* server = "api.thingspeak.com";

#define DHTPIN 4      // GPIO Pin where the dht11 is connected

DHT dht(DHTPIN, DHT11);

WiFiClient client;

void setup()
{
    Serial.begin(115200);
    delay(10);
    dht.begin();
    Serial.println("Connecting to ");
    Serial.println(ssid);
    WiFi.begin(ssid, pass);
    while (WiFi.status() != WL_CONNECTED)
    {
        delay(550);
        Serial.print(".");
    }
    Serial.println("");
    Serial.println("WiFi connected");
}

void loop()
{
    float h = dht.readHumidity();
    float t = dht.readTemperature();

    if (isnan(h) || isnan(t))
    {
        Serial.println("Failed to read from DHT sensor!");
        return;
    }

    if (client.connect(server,80))
    {

```



```

String postStr = apiKey;
postStr += "&field1=";
postStr += String(t);
postStr += "&field2=";
postStr += String(h);
postStr += "\r\n\r\n";

client.print("POST /update HTTP/1.1\n");
client.print("Host: api.thingspeak.com\n");
client.print("Connection: close\n");
client.print("X-THINGSPEAKAPIKEY: "+apiKey+"\n");
client.print("Content-Type: application/x-www-form-urlencoded\n");
client.print("Content-Length: ");
client.print(postStr.length());
client.print("\n\n");
client.print(postStr);
Serial.print("Temperature: ");
Serial.print(t);
Serial.print(" degrees Celcius, Humidity: ");
Serial.print(h);
Serial.println("%. Send to Thingspeak.");
}

client.stop();
Serial.println("Waiting...");
delay(10000);
}

```