

## My Channels

New Channel

Name	Created
<div>Temp and Hum</div> <div> <a href="#">Private</a> <a href="#">Public</a> <a href="#">Settings</a> <a href="#">API Keys</a> <a href="#">Data Import / Export</a> </div>	2017-02-06

## Help

Collect data in a ThingSpeak channel from a device, from another channel, or from the web. Click **New Channel** to create a new ThingSpeak channel.

Learn to [create channels](#), explore and transform data.


Learn more about [ThingSpeak Channels](#).

## Examples

- [Arduino](#)
- [Arduino MKR1000](#)
- [ESP8266](#)
- [Raspberry Pi](#)
- [Netduino Plus](#)

Channels ของเราจะ

จากนั้น เมื่อเราคลิกเลือกปุ่ม New Channel (ปุ่มเขียว) เราก็จะเจอหน้าต่างให้กรอกรายละเอียดต่างๆ ดังนี้ค่ะทุกคน



[Channels](#)
[Apps](#)
[Community](#)
[Support](#)

[How to Buy](#)
[Account](#)
[Sign Out](#)

## New Channel

Name

Temperature and Humidity Monitoring

Description

Field 1

Temperature

☒

Field 2

Humidity

☒

Field 3

☐

Field 4

☐

Field 5

☐

Field 6

☐

Field 7

☐

Field 8

☐

Metadata

Tags

(Tags are comma separated)

Make Public

☐

URL

Elevation

Show Location

☐

Latitude

0.0

Longitude

0.0

## Help

Channels store all the data that a ThingSpeak application collects. Each channel includes eight fields that can hold any type of data, plus three fields for location data and one for status data. Once you collect data in a channel, you can use ThingSpeak apps to analyze and visualize it.

### Channel Settings

- **Channel Name:** Enter a unique name for the ThingSpeak channel.
- **Description:** Enter a description of the ThingSpeak channel.
- **Fields:** 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.
- **Latitude:** Specify the position of the sensor or thing that collects data in decimal degrees. For example, the latitude of the city of London is 51.5072.
- **Longitude:** Specify the position of the sensor or thing that collects data in decimal degrees. For example, the longitude of the city of London is -0.1275.
- **Elevation:** Specify the position of the sensor or thing that collects data in meters. For example, the elevation of the city of London is 35.052.
- **Make Public:** If you want to make the channel publicly available, check this box.
- **URL:** If you have a website that contains information about your ThingSpeak channel, specify the URL.
- **Video ID:** If you have a YouTube™ or Vimeo® video that displays your channel information, specify the full path of the video URL.

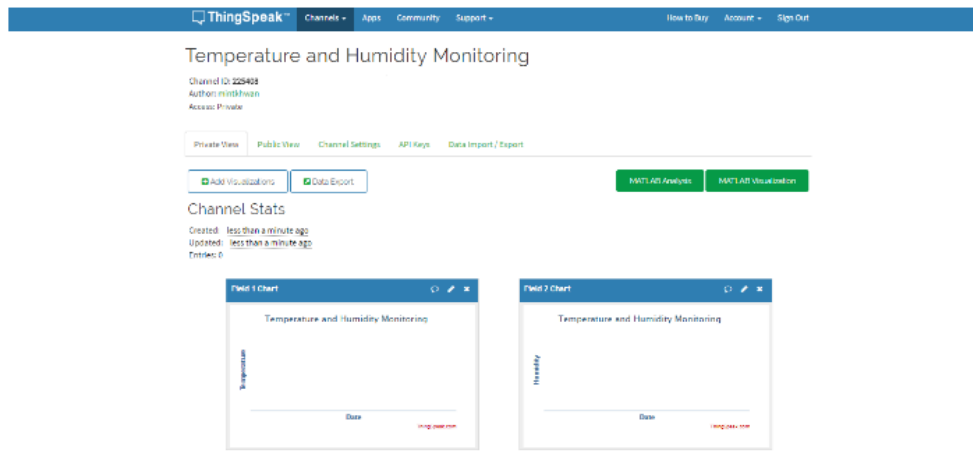
### Using the Channel

You can get data into a channel from a device, website, or another ThingSpeak channel. You can then visualize data and transform it using [ThingSpeak Apps](#).

See [Tutorial: ThingSpeak and MATLAB](#) for an example of measuring dew point from a weather station that acquires data from an Arduino® device.

[Learn More](#)

- **Name** คือ ชื่อ Channel ของเรานะคะ ซึ่งจะบอกก่อนเลขนะค่ะว่า ชื่อนี้จะติดอยู่บนด้านบนของกราฟข้อมูลที่เรากำลังต้องการแสดงด้วยนะค่ะ เพราะฉะนั้นควรตั้งให้ครอบคลุมกับแต่ละกราฟข้อมูลด้วยนะค่ะ มันจะได้เป็น **Temperature and Humidity Monitoring**
  - **Field ต่างๆ** คือ ชื่อของข้อมูลที่เราต้องการจะแสดงนะค่ะ แต่ละ Field ก็จะได้แยกกราฟกันค่ะ มันจะแยกเป็นสองกราฟนะค่ะ **Field1** คือ **Temperature** และ **Field2** คือ **Humidity** ค่ะ
- เบื้องต้นเราก็จะกรอกรายละเอียดไปก่อนก่อนค่ะ จากนั้นกด **Save Channel** แล้วลองมา Test ดูซิ ว่ากราฟของเราขึ้นโชว์หรือเปล่า



แท่นแท่น กราฟของเรามาแล้วค่ะ มาแต่โครงแต่ข้อมูลยังไม่มา Entries: 0 เลย ฮ่าๆ

ต่อมาให้เราไปลอง Check ที่ เมนู API Keys ค่ะ ซึ่งในหน้านี้น่าจะสำคัญมากเลยนะค่ะ มีทั้ง Key ที่ใช้สำหรับส่งข้อมูล และ Key ที่เอาไว้ใช้สำหรับรับข้อมูลค่ะ แต่ในกรณีนี้เราใช้

ต่อมาเรามาประกอบวงจรให้ Sensor DHT22 ทำการรับค่า Temperature และ Humidity กันค่ะ ซึ่งมันจะต่อตามภาพด้านล่างนี้นะค่ะ

```

#include
<ESP8266WiFi.h>

const char* ssid = "....."; //ssid ของWi-Fi ที่เราต้องการเชื่อมต่อค่ะ
const char* pass = "....."; //password ของWi-Fi ที่เราต้องการเชื่อมต่อค่ะ


#include <DHT.h>
#define DHTPIN D7
#define DHTTYPE DHT22


char thingSpeakAddress[] = "api.thingspeak.com";
String writeAPIKey = "....."; //เอาWrite API Key ของเรามาใส่ค่ะ
WiFiClient client; //เรียกใช้การเชื่อมต่อกับ Wi-Fi


DHT dht (DHTPIN, DHTTYPE);


void setup() {
  Serial.begin(115200);
  WiFi.begin(ssid, pass);
  dht.begin();

  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(500);
  }

  Serial.println("");
  Serial.println("Wi-Fi connected");
  Serial.print("IP : ");
  Serial.println(WiFi.localIP());
}


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

  Serial.print("temp : ");
  Serial.println(t);

```

```
Serial.print("hum : ");
Serial.println(h);
delay(1000);
```

```
String temp = (String)t; //เนื่องจาก temp เป็น float ซึ่งจะส่งค่าไม่ได้เราจึงต้องแปลงเป็น String ก่อนค่ะ
String hum = (String)h; //เนื่องจาก hum เป็น float ซึ่งจะส่งค่าไม่ได้เราจึงต้องแปลงเป็น String ก่อนค่ะ
String data = "field1=" + temp + "&field2=" + hum; //ข้อมูล String ที่เราจะส่งค่าไปยัง
ThingSpeak ค่ะ
```

```
if (client.connect(thingSpeakAddress, 80)) {
    Serial.println(data);
    client.print("POST /update HTTP/1.1\n"); //การส่งค่าข้อมูลผ่าน HTTP ค่ะ
    client.print("Host: api.thingspeak.com\n");
    client.print("Connection: close\n");
    client.print("X-THINGSPEAKAPIKEY: " + writeAPIKey + "\n");
    client.print("Content-Type: application/x-www-form-urlencoded\n");
    client.print("Content-Length: ");
    client.print(data.length());
    client.print("\n\n");
    client.print(data); //path ข้อมูลที่เราจะส่งค่าไปยัง ThingSpeak ค่ะ
}

}
```

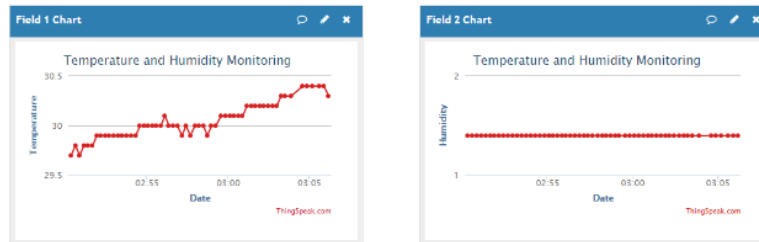
เช็คค่าที่ Temperature และ Humidity จาก Serial Monitor ค่ะ



ขั้นสุดท้าย เราลองมาเช็คที่หน้าเว็บดูนะค่ะ ให้เราเลือกไปที่เมนู private view ใน Channel ที่เราสร้างไว้ค่ะ ซึ่งกราฟที่เราสร้างไว้ตอนแรกก็จะมีค่า Temperature และ Humidity ที่เรารับจากตัว Sensor DHT22 ส่งข้อมูลมายัง ThingSpeak ค่ะ

#### Channel Stats

Created: 37 minutes ago  
Updated: less than a minute ago  
Last entry: less than a minute ago  
Entries: 58



เห็นมั๊ยคะ ข้อมูล Temperature และ Humidity ของเรา ที่เราลงมือทำเอง ก็ไปแสดงบนกราฟ บน Cloud Service ได้เรียบร้อยแล้ว หวังว่าทุกคนจะชอบบทความนี้นะคะ วันนี้มันขอตัวไปนอนก่อนนะค่ะ ฮ่าๆ ผินดีค่ะทุกคน ZZZZzzz



Thingspeak เป็นบริการ Platform as a Services ที่ให้บริการ เก็บข้อมูลแบบ Real-time, แสดงข้อมูลกราฟ จากที่ใดก็ได้ในโลก และสามารถเปิดดูจากที่ไหนก็ได้ อีกเช่นกัน ซึ่งก็คล้ายกับ data.sparkfun.com แต่สิ่งที่ต่างกัน คือ Thingspeak จะแสดงผลข้อมูลผ่านกราฟได้ และ คึงข้อมูล (PUT DATA) ไปใช้แสดงผลที่เว็บเราได้ผ่าน <iframe> ซึ่งมีทั้งบริการทั้งแบบฟรี และ เสียเงิน แต่ในเวอร์ชันฟรี จะยอมให้เราบันทึกข้อมูล ทุกๆ 15 วินาที นะครับ

สำหรับ Thinkspeak จะเปิดให้เรา Sending Data ได้ 2 รูปแบบ

## 1. HTTP Request

การส่งข้อมูลจะใช้ GET Method รูปแบบของ HTTP Request แบบที่ data.sparkfun.com ใช้งาน สำหรับ Thinkspeak จะใช้รูปแบบดังรูป

**Sending Data** ([more help](#))

Add data by sending a POST or GET to:

`https://api.thingspeak.com/update`

Please include your write API key and some data, for example:

`https://api.thingspeak.com/update?key=RxEC70XG8MALL272&field1=0`

## 2. REST API (Representational State Transfer)

เป็น Web Service แบบเรียบง่าย โดยเรียกใช้ผ่านทาง HTTP Method GET/POST/PUT/DELETE ซึ่งมันออกแบบไว้เป็น เชื่อมต่อกับจัดการฐานข้อมูล แสดงผล, เพิ่มรายการ, แก้ไขรายการ และ ลบรายการ ซึ่งมันทำได้มากกว่า HTTP Request

Method	การทำงาน	รูปแบบ URI
GET	เรียกข้อมูลมาแสดงหลายรายการ	http://example.com/tests/
	เรียกข้อมูลมาแสดงทีละรายการ	http://example.com/tests/123
POST	เพิ่มข้อมูล	http://example.com/tests/
PUT	แก้ไขข้อมูล	http://example.com/tests/123
DELETE	ลบข้อมูล	http://example.com/tests/123

ซึ่งใน API ของ Thinkspeak ใช้แค่ GET และ POST ครั้น ซึ่งเราจะใช้แบบ HTTP Request เหมือนเดิมก็ได้เช่นกัน แต่ในบทความนี้ ทางผมจะเลือกใช้ REST API บ้าง เพื่อให้พวกเราเห็นวิธีการส่งข้อมูล หลากหลายรูปแบบครับ ซึ่งใครอยากลอง HTTP Request ลองเอง ก็ได้ครับ ไม่ยาก แค่จัด format ที่จะ Request ใหม่เท่านั้น