| **Job Sheet** |
| --- |

| **Module** | **:** | IoT Fundamentals | **Module Code** | **:** |  | |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  | |
| **Job No** | **:** | 9 | **Duration** | **:** | F/T: | 3 hours |
|  |  |  |  |  | P/T: |  |

| **Job Title** | **:** | Setup communication from Device to Platform |
| --- | --- | --- |
|  |  |  |
| **Objectives** | **:** | 1. Prepare RPi and Upswift 2. Connect DHT11 to RPi Pins 3. Send Temperature and Humidity data to ThingSpeak via HTTP requests 4. MQTT Subscribe using another RPi |

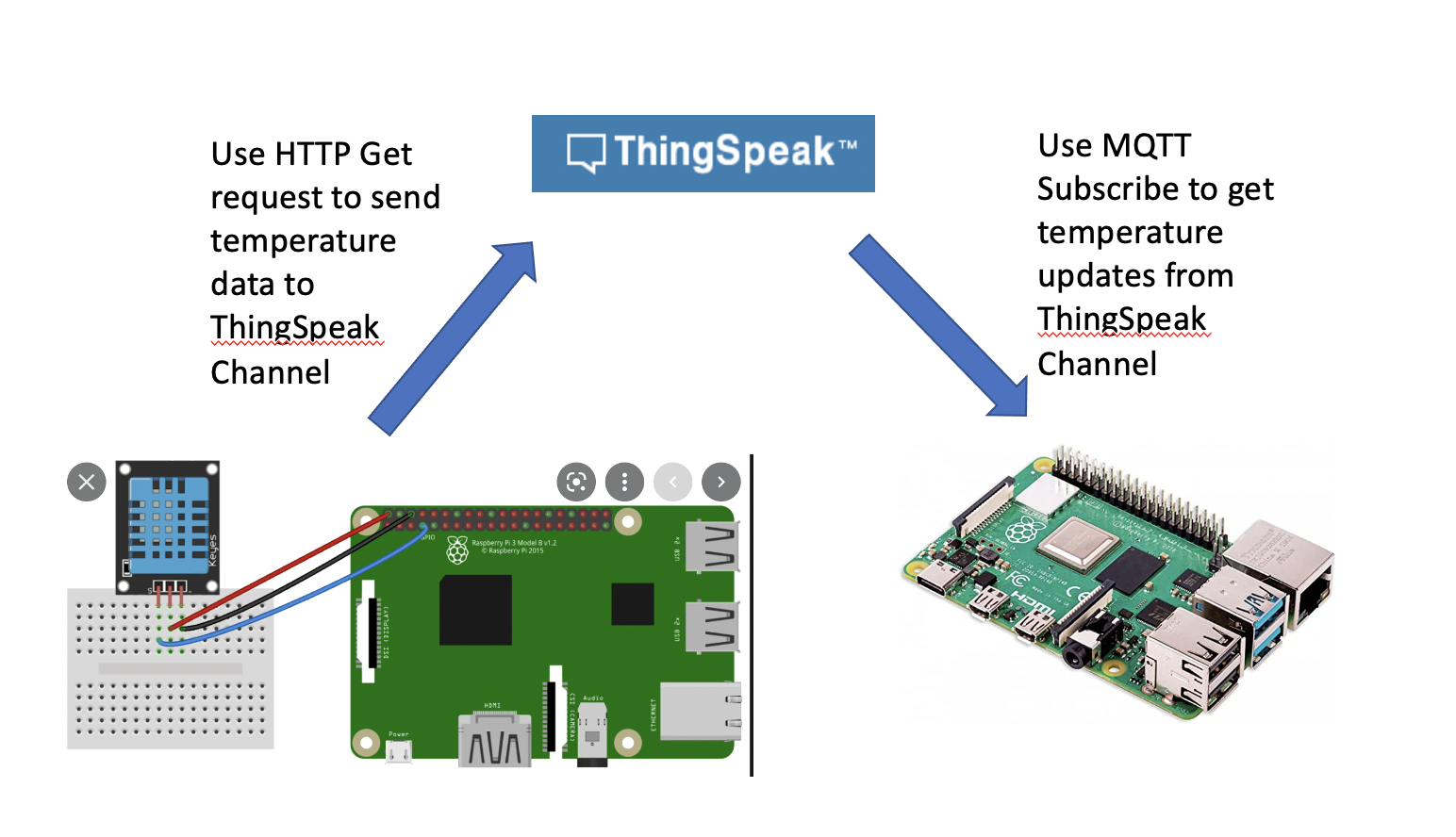
**Tools, Equipment and Materials**

| PC |
| --- |
| RPi with microSD x 2 |
| RPi Power Adaptor  DHT11 x 1  Breadboard  4G WiFi Router |
|  |

**Number of Tasks to Complete**: 6

This jobsheet will require you to work in pairs. One person will setup a Raspberry Pi with a DHT 11 sensor and send data to a ThingSpeak Channel. The other person will setup a Raspberry Pi to subscribe to updates from the ThingSpeak Channel. You can work in pairs to complete all the tasks together. Concept shown in diagram below.

Two different protocols are used to send and receive data; HTTP and MQTT.

****

**Part A: Prepare RPi and Upswift (0.5 hour)**

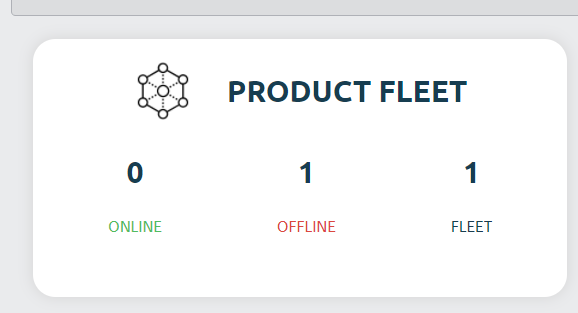
Before performing any hardware configurations, we need to ensure that the RPi is accessible remotely.

YOU MUST HAVE COMPLETED JOB 1 BEFORE DOING THIS

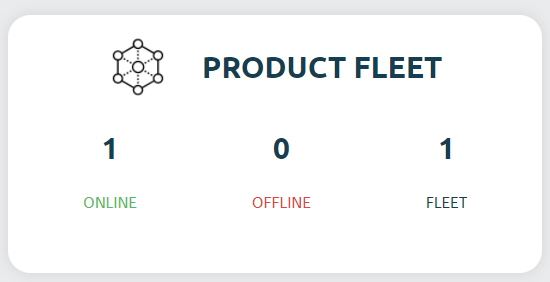
**Task 1 – Prepare RPi and UpSwift**

Perform the following steps to setup RPi:

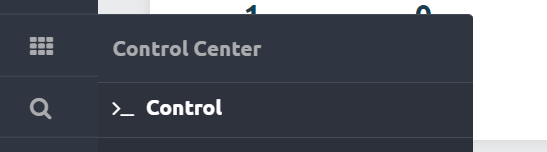
1. Connect only the power adaptor to RPi.
2. Turn on the power
3. Go to <https://dashboard.upswift.io/dashboard/> and login. The device would be **offline**.



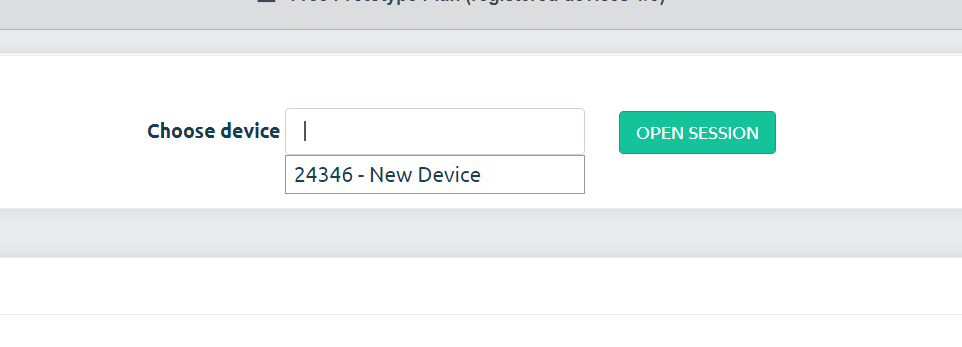
1. Wait for a while and refresh the web page by clicking . Device should appear online after 1 - 2 mins.



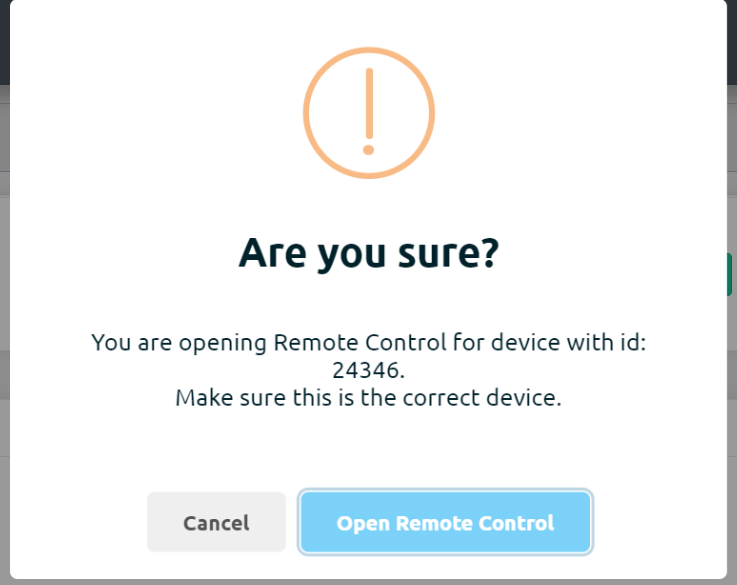
1. On the left side of your menu, click on the **Control** option under Control center.



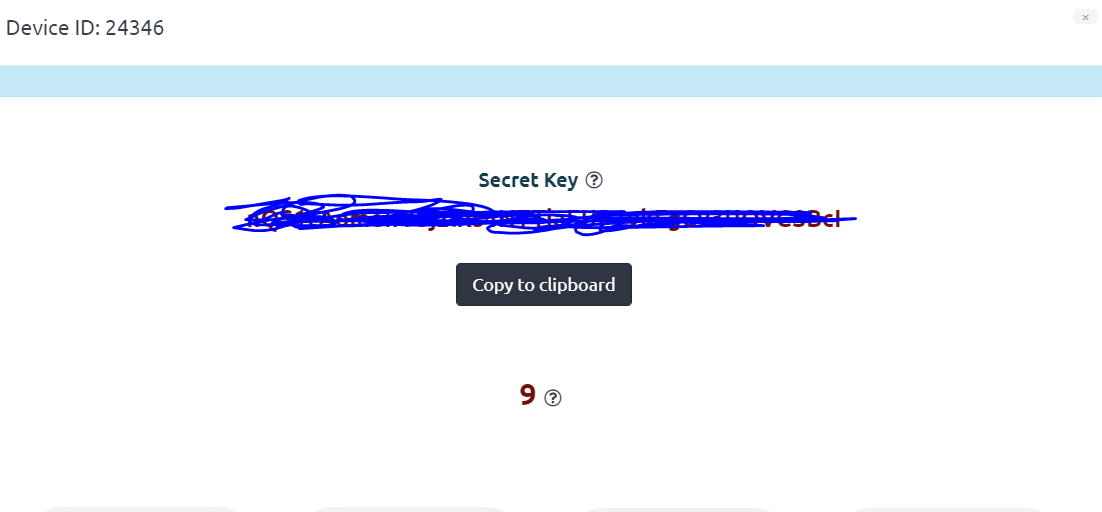
1. On the page center, select your device and click open session.



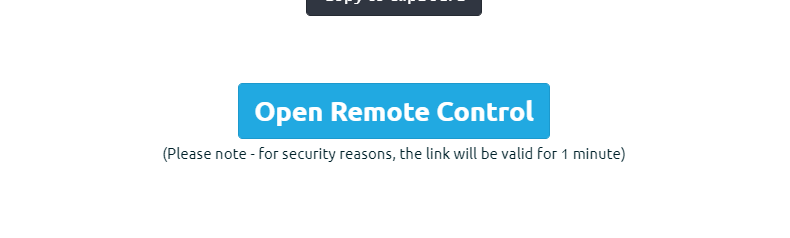
1. For the prompt, click **Open Remote Control.**



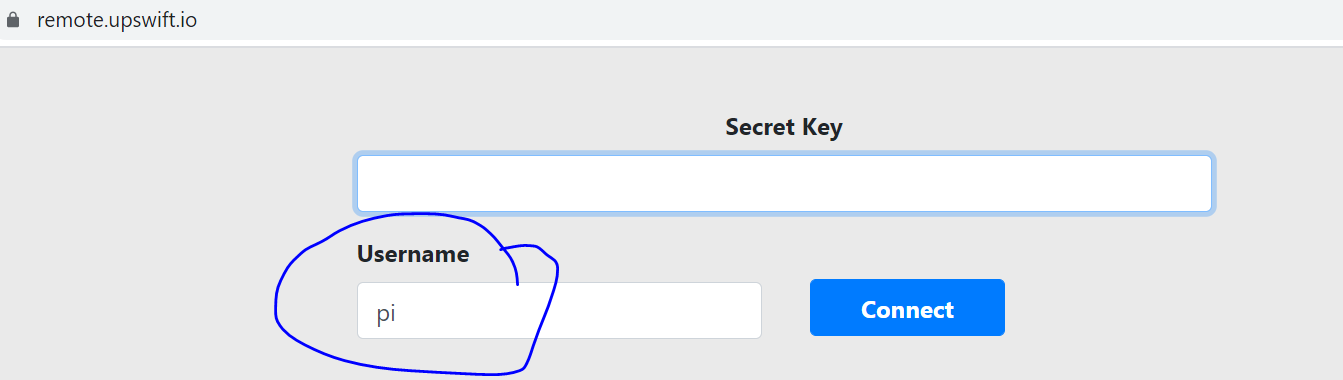
1. You will be shown a secret key. Click on **Copy to clipboard** button

****

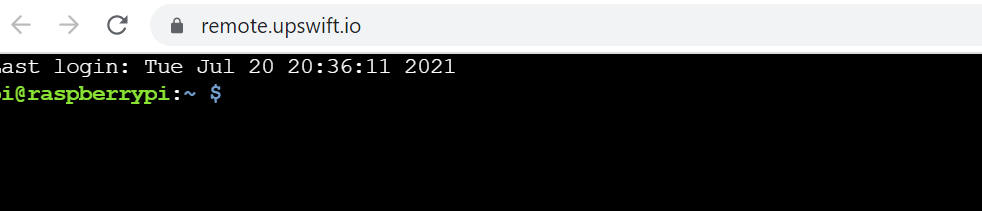
1. Once the timer is up, click on the **Open Remote Control** button.

****

1. **Paste** the secret key in the input text box and change username to pi. Click on **Connect**.

****

1. You will be presented with a browser based RPi terminal.

****

**Part B: Connect DHT11 to RPi Pins**

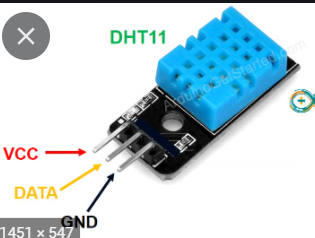
DHT11 **Temperature** & **Humidity** Sensor features a temperature & humidity sensor

complex with a **calibrated digital signal output.** By using the exclusive digital-signal-acquisition technique and temperature & humidity sensing technology, it ensures high reliability and excellent long-term stability. This sensor includes a **resistive-type humidity measurement** component and an **NTC temperature measurement component.**

The datasheet can be found here:

<https://www.mouser.com/datasheet/2/758/DHT11-Technical-Data-Sheet-Translated-Version-1143054.pdf>

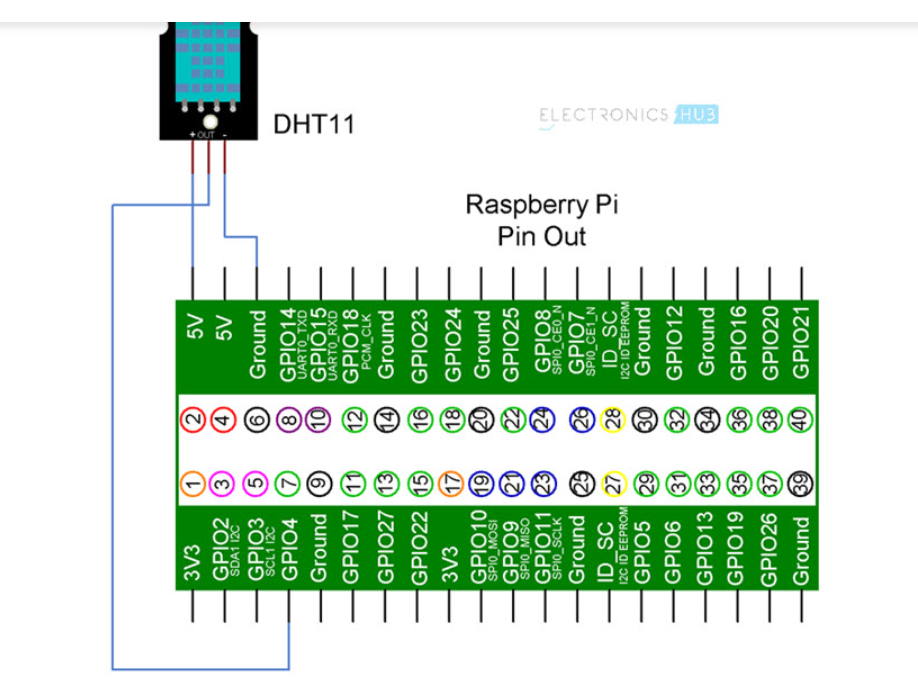
The DHT11 has 3 pins, Vcc, Data and GND.



**Task 2 – Connect DHT11 to RPi**

Perform the following steps

1. Connect GPIO extension board from RPi to breadboard.
2. Connect DHT11 to the RPI using the breadboard as shown



**Part C: Execute program send Temperature and Humidity to ThingSpeak**

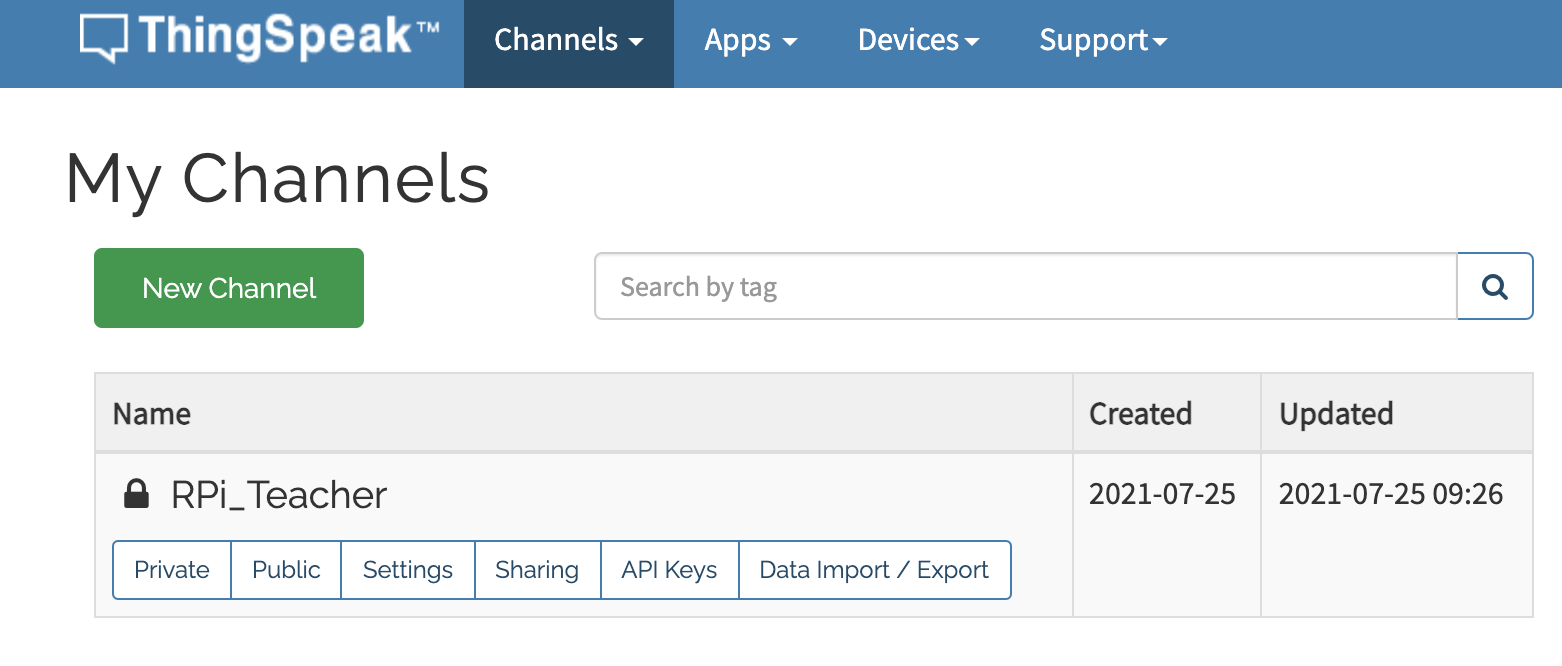
We will be using a HTTP GET request to send temperature and humidity data to the ThingSpeak Channel.

In order to do that, first we need to get a write key from ThingSpeak.

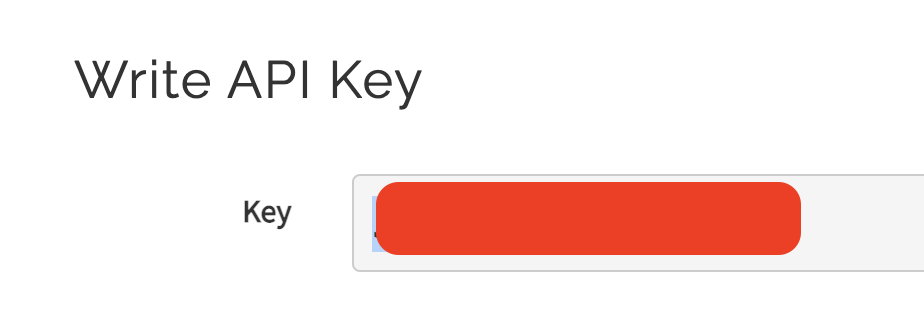
**Task 3 – Get Write Key from ThingSpeak**

Perform the following steps.

1. Login to <https://thingspeak.com/>
2. Look for your Channel created in Jobsheet 8.



1. Click on **API Keys.**
2. Copy the **Write API Key.** We need to use it later in the python code on the RPi.

****

**Task 4 – Install Requests Library**

In order to send data by making a HTTP request, we need to use a python library to help us. The requests library makes sending HTTP requests simple, with one line of code.

<https://pypi.org/project/requests/>

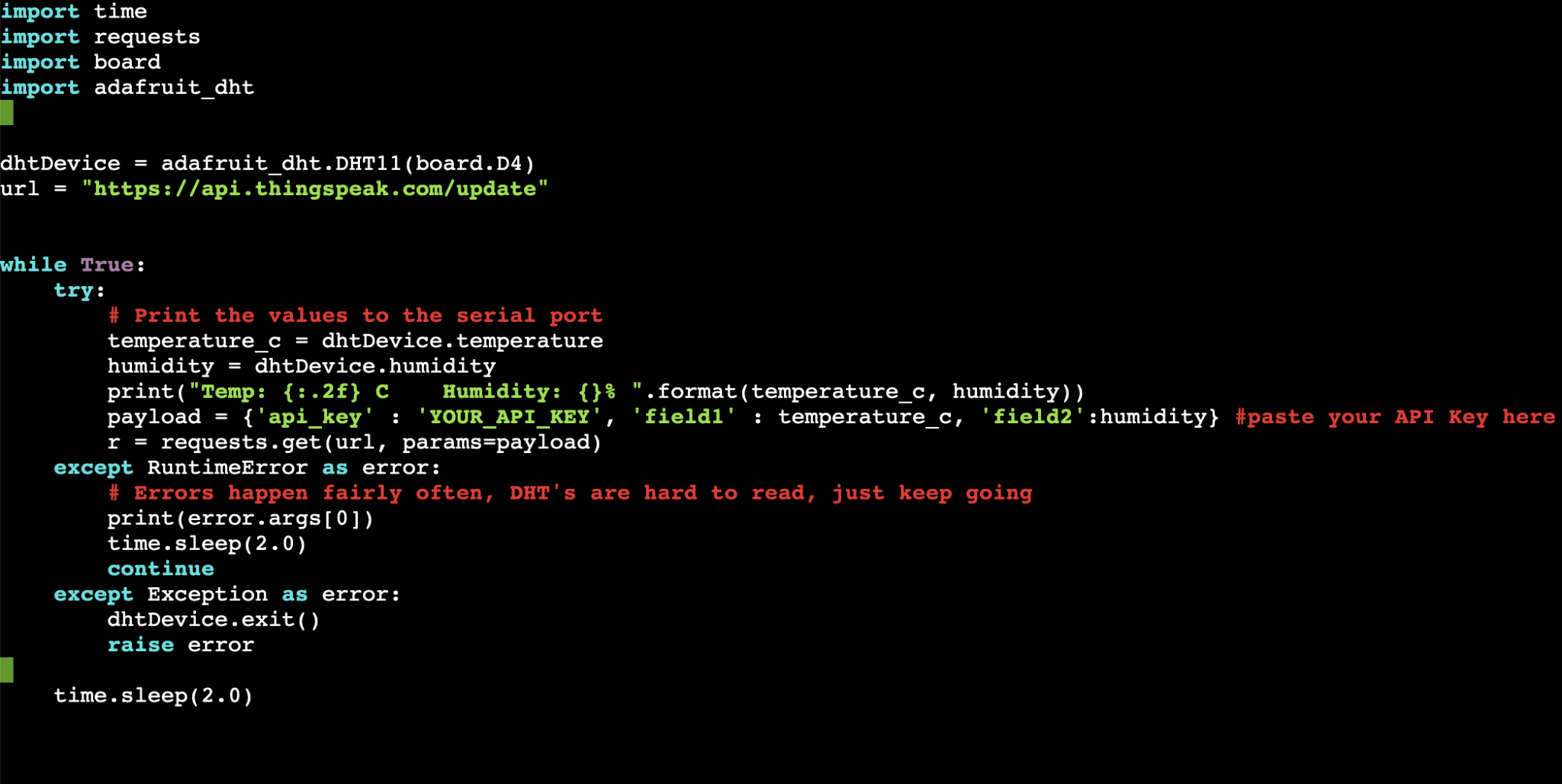
Perform the following steps.

1. In the RPi terminal, run **pip3 install requests**

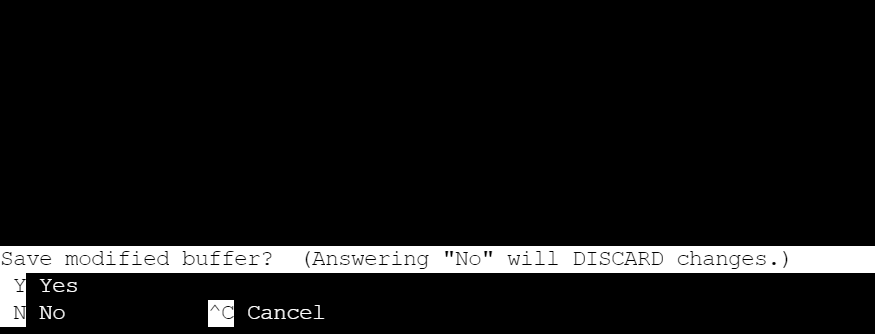
**Task 5 – Send data to ThingSpeak via HTTP**

Perform the following steps.

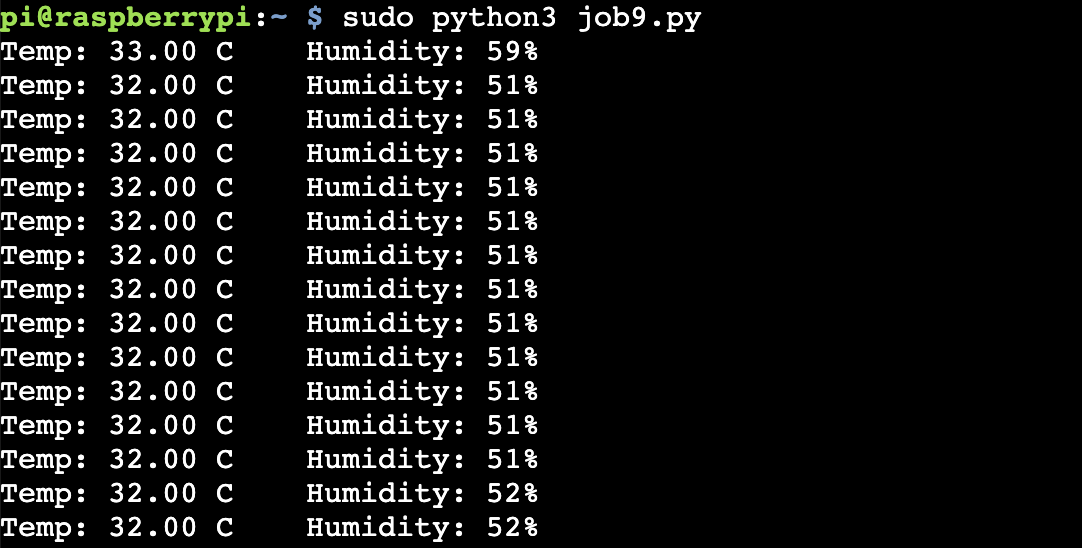
1. Run **sudo nano job9.py.**
2. Copy the code found at <https://github.com/Prakashash18/iot-fundamentals/blob/main/job9.py>
3. Paste it into the nano editor by right clicking and pasting. Replace the YOUR\_API\_KEY with the **Write API Key** copied earlier.



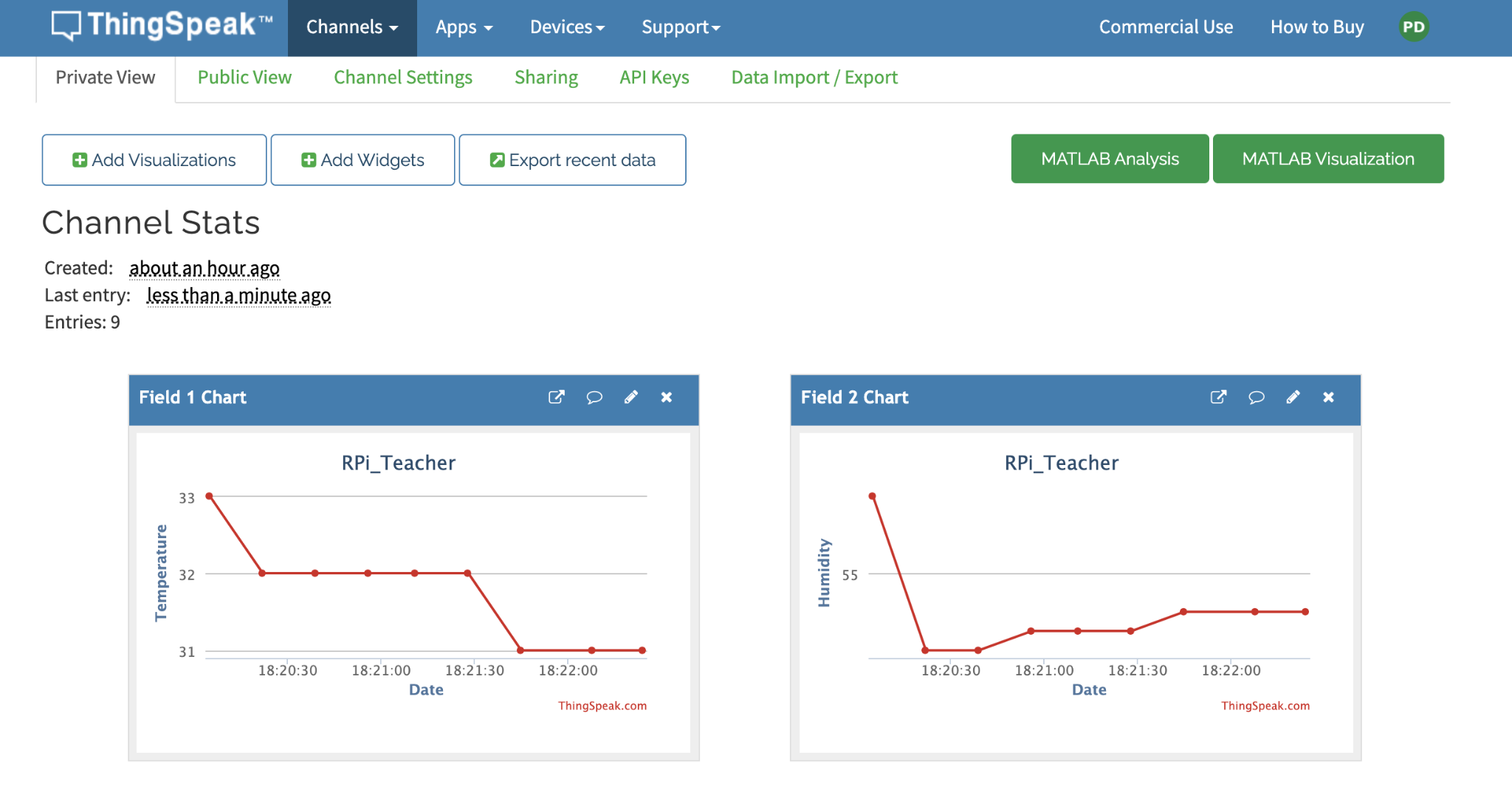
1. [Keyboard] Press CTRL + X to save your work. You will be prompted with the following message.



1. [Keyboard] Press **y** for yes and hit **Enter**
2. On the terminal, run **sudo python3 job9.py** to execute the code. Note that we are using python version 3.
3. You should see the output on the terminal



1. Back on ThingSpeak dashboard under **Private View**, every 15 seconds, you’d see data being shown



**Task 6 – MQTT Subscribe using another RPi**

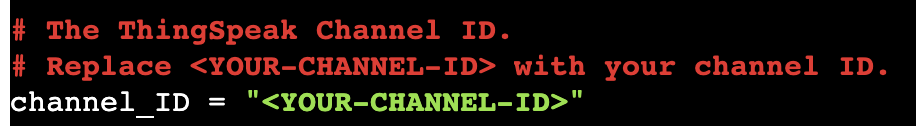
1. Run the following command to install a MQTT library.

****

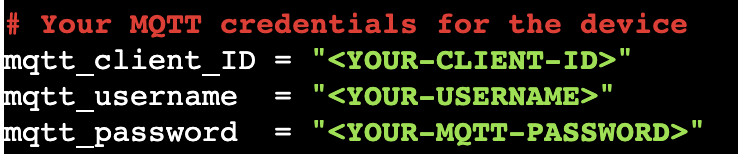
1. In the terminal, run **sudo nano job9\_subscribe.py.**

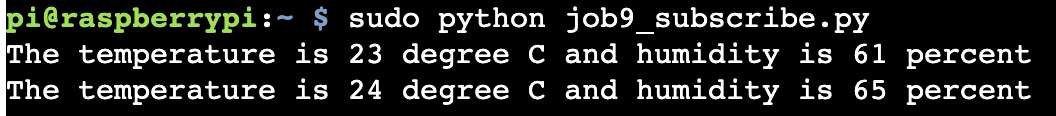
****

1. Copy the code at <https://github.com/Prakashash18/iot-fundamentals/blob/main/job9_subscribe.py>
2. Replace <YOUR-CHANNEL-ID> with the channel ID from thingspeak (Refer to job 8, task 2, step 5).



1. Replace <YOUR-CLIENT-ID>, <YOUR-USERNAME>, <YOUR-PASSWORD> with the credentials downloaded from thingspeak (Refer to job 8, task 3, step 6)



1. Run your code as shown, you should get outputs printed every 15 seconds as shown.
2. 

**Questions**

1. What Application Layer protocol is used to send data from the RPi to ThingSpeak in the experiment above?



1. What HTTP method is used in the experiment above?



1. Which library in python is used to make HTTP requests in the experiment above?

