IOT

ASSIGNMENT 2
SENSOR DATA TO CLOUD AND MOBILE OPERATION

INTRODUCTION

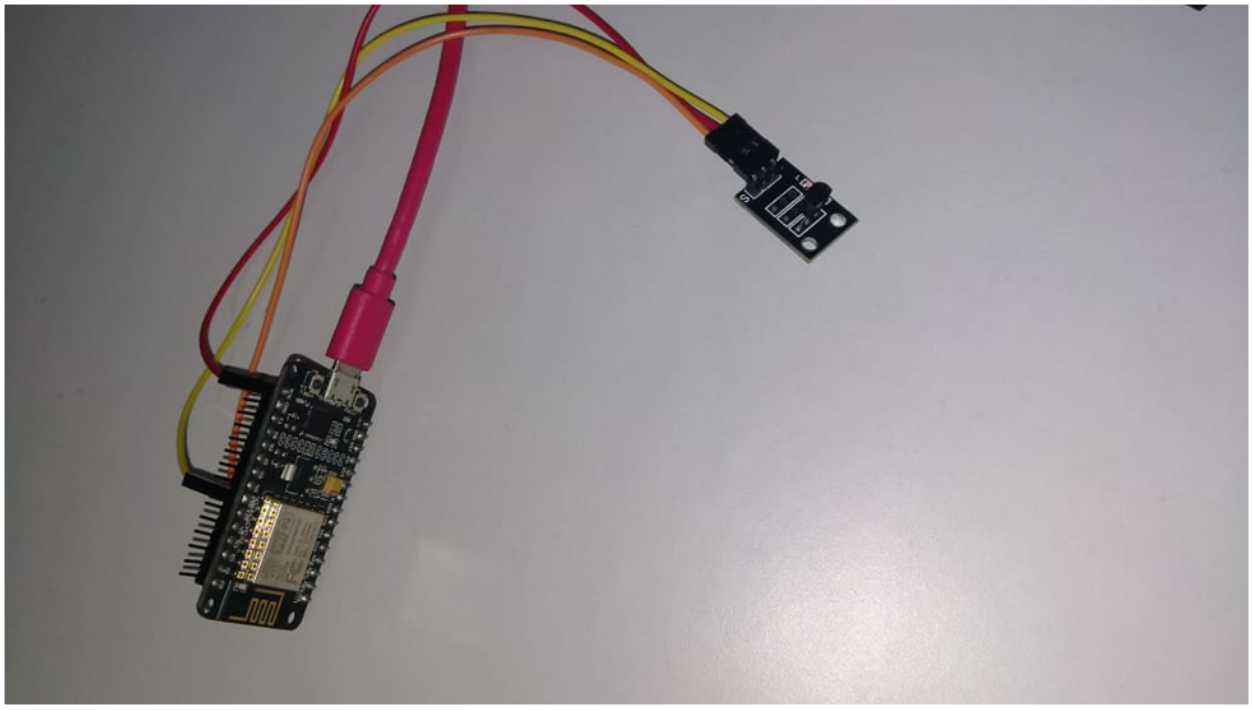
ESP8266EX is capable of functioning consistently in industrial environments, due to its wide operating temperature range. With highly-integrated on-chip features and minimal external discrete component count, the chip offers reliability, compactness and robustness.

A ***temperature sensor (TMP36)*** is an electronic device that measures the temperature of its environment and converts the input data into electronic data to record, monitor, or signal temperature changes. There are many different types of temperature sensors. Some temperature sensors require direct contact with the physical object that is being monitored (contact temperature sensors), while others indirectly measure the temperature of an object (non-contact temperature sensors).

ThingSpeak is an IoT analytics platform service that allows you to aggregate, visualize, and analyze live data streams in the cloud. You can send data to ThingSpeak from your devices, create instant visualization of live data, and send alerts.

CONNECTION

The temperature sensor LM35 has 3 legs, the first leg is VCC, you can connect this to the 5V (ESP8266 board's output is 3.3V). The middle leg is Vout (where the temperature is read from, you can connect this to the analog input of the ESP8266 pin AD0, this is located at the top right hand side of the board as shown in picture. And the right leg should be connected to the ground.

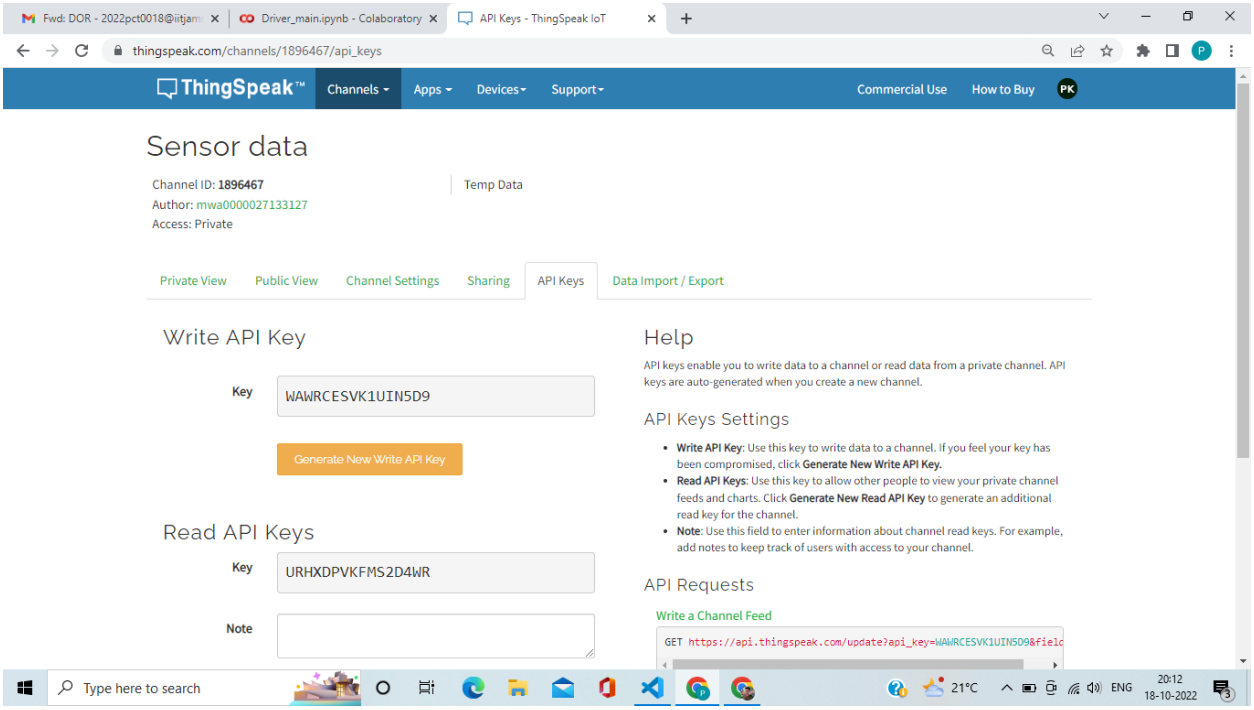


API KEY: WAWRCESVK1UIN5D9

THINGS SPEAK CHANNEL ID: 1896467

A **channel** is **where we send our data to store**. Each channel includes 8 fields for any type of data, 3 location fields, and 1 status field. Once we have a ThingSpeak Channel we publish data to the channel, have ThingSpeak process the data, then have our application retrieve the data.

The 16-digit **API key** allows you to read from a private channel and write to a channel



PROTOCOLS

GET:

GET is used to request data from a specified resour

- GET requests can be cached
- GET requests remain in the browser history
- GET requests can be bookmarked
- GET requests should never be used when dealing with sensitive data
- GET requests have length restrictions
- GET requests are only used to request data (not modify)

POST:

POST is used to send data to a server to create/update a resource.

- POST requests are never cached

- POST requests do not remain in the browser history
- POST requests cannot be bookmarked
- POST requests have no restrictions on data length

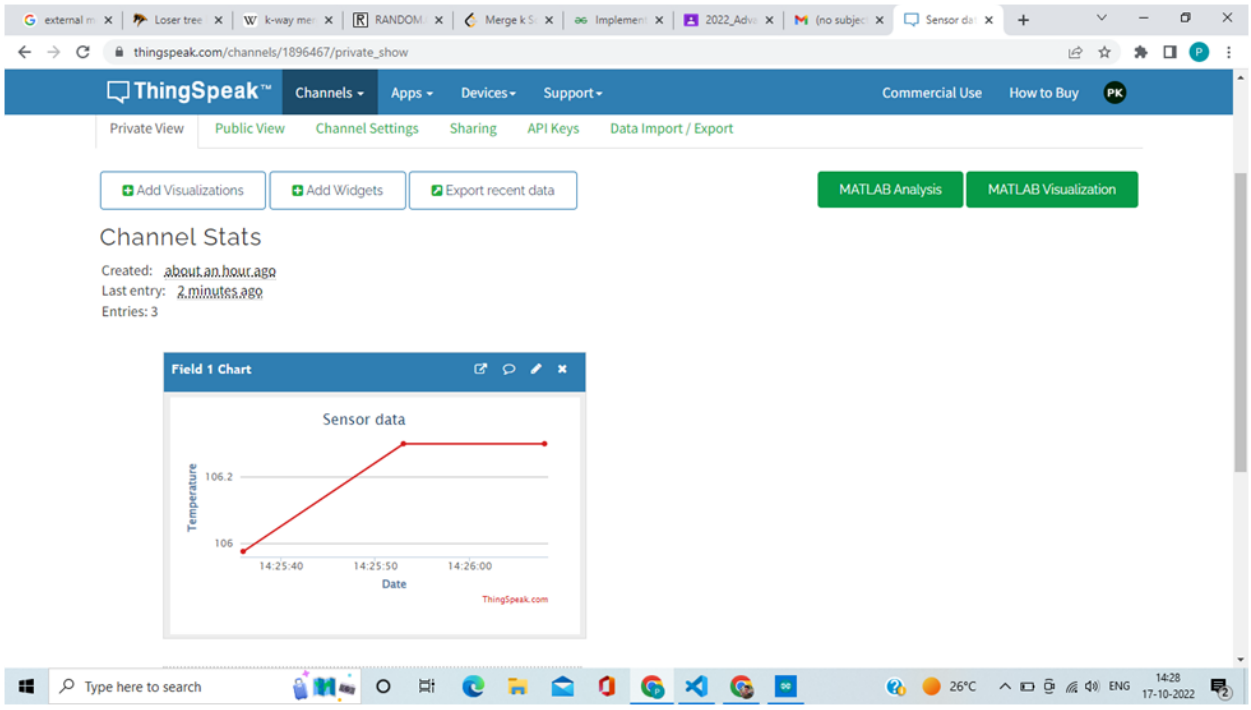
CODE

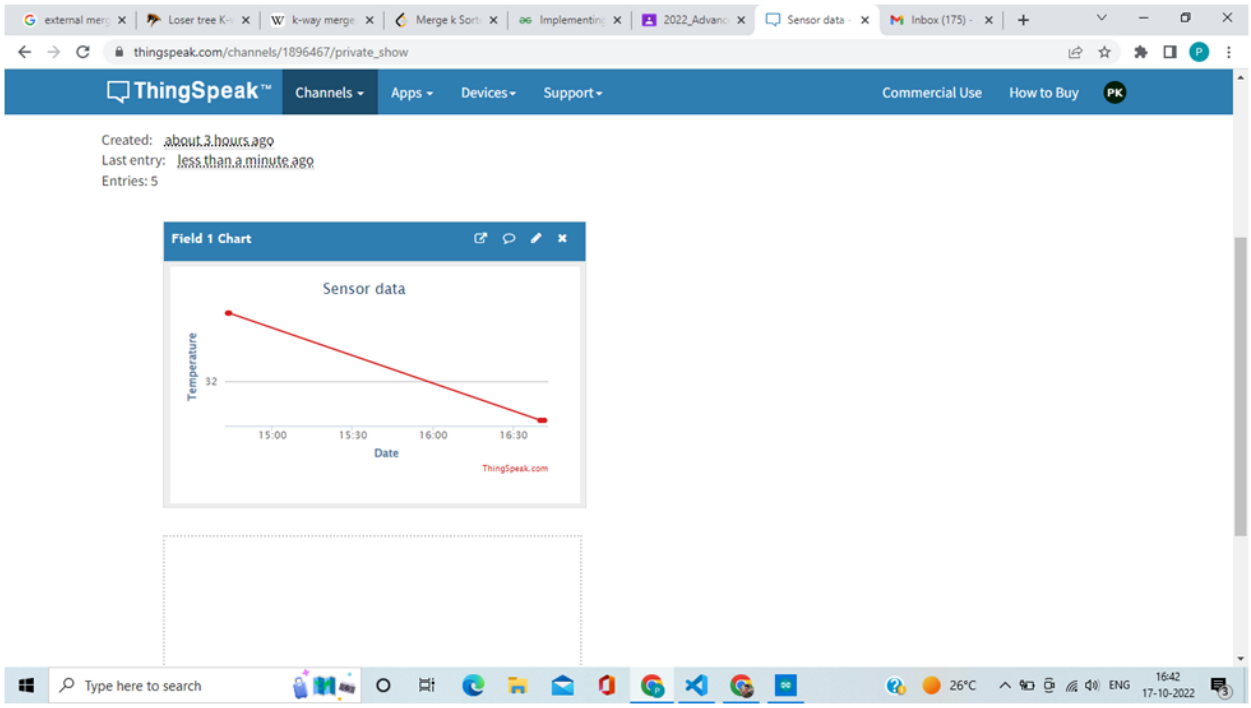
```
#include <ESP8266WiFi.h>

#include "ThingSpeak.h"

#define TMPPin A0

const char* ssid = "redmi2";   // your network SSID (name)

const char* password = "pp7654321";   // your network password

WiFiClient  client;

unsigned long myChannelNumber = 1896467;

const char * myWriteAPIKey = "WAWRCESVK1UIN5D9";

unsigned long lastTime = 0;

unsigned long timerDelay = 30000;

void setup() {

  Serial.begin(115200);  //Initialize serial

  WiFi.mode(WIFI_STA);

  ThingSpeak.begin(client);  // Initialize ThingSpeak

}

void loop() {

  if ((millis() - lastTime) > timerDelay) {



          // Connect or reconnect to WiFi
```

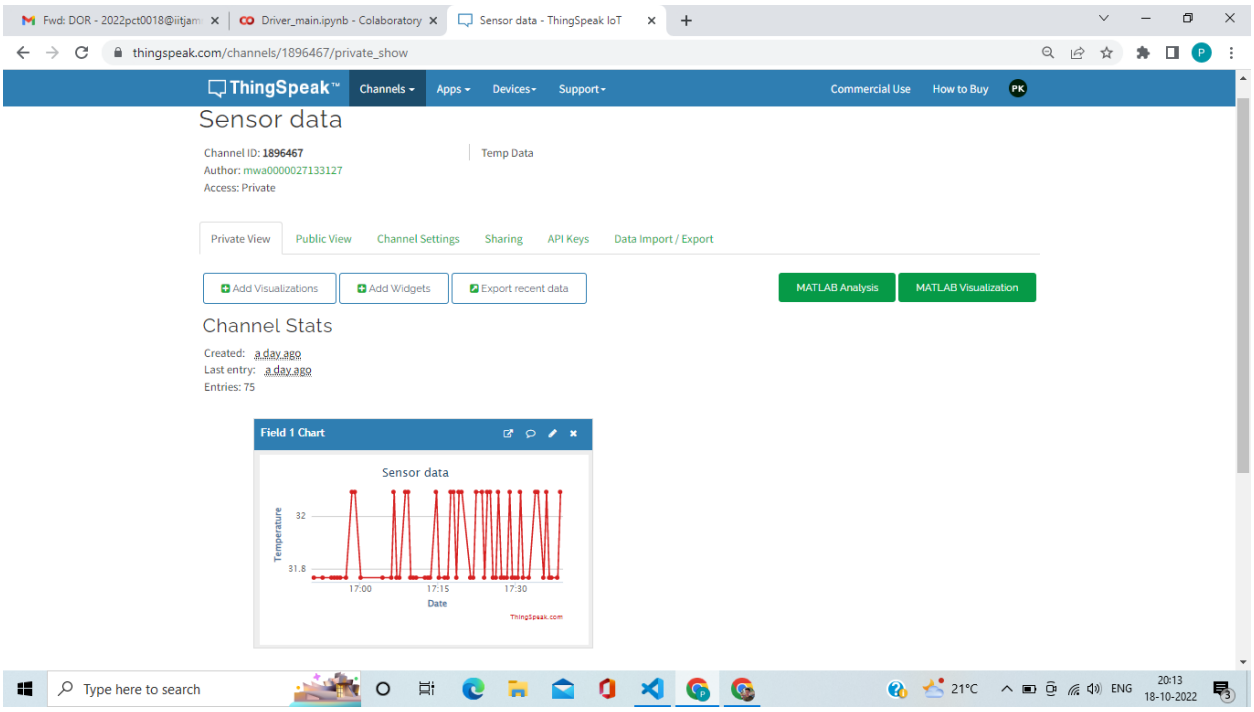
```
if(WiFi.status() != WL_CONNECTED){
  Serial.print("Attempting to connect");
  while(WiFi.status() != WL_CONNECTED){
    WiFi.begin(ssid, password);
    delay(5000);
      }
  Serial.println("\nConnected.");
      }
      // Get a new temperature reading
      int tmpValue = analogRead(TMPPin);
      float voltage = tmpValue * 3.3;// converting that reading to voltage
      voltage /= 1024.0;
      float temperatureC = (voltage - 0.5) * 100-90 ;
      float temperatureF = (temperatureC * 9.0 / 5.0) + 32.0;

Serial.print("Temperature (ºC): ");
Serial.println(temperatureC);

      //uncomment if you want to get temperature in Fahrenheit
/*temperatureF = 1.8 * bme.readTemperature() + 32;
Serial.print("Temperature (ºC): ");
Serial.println(temperatureF);*/
```

```
int x = ThingSpeak.writeField(myChannelNumber, 1, temperatureC, myWriteAPIKey);


if(x == 200){
  Serial.println("Channel update successful.");
}
else {
Serial.println("Problem updating channel. HTTP error code " + String(x));
}
lastTime = millis()}}
```

OUTPUT




thingspeak.com/channels/1896467/private_show
ThingSpeak™
Channels
Apps
Devices
Support
Commercial Use
How to Buy
Created: about 3 hours ago
Last entry: less than a minute ago
Entries: 5
Field 1 Chart
Sensor data
Temperature
32
15:00
15:30
16:00
16:30
Date
ThingSpeak.com
Type here to search
26°C
ENG
16:42
17-10-2022


thingspeak.com/channels/1896467/private_show
ThingSpeak™
Channels
Apps
Devices
Support
Commercial Use
How to Buy
Sensor data
Channel ID: 1896467
Author: mwa0000027133127
Access: Private
Temp Data
Private View
Public View
Channel Settings
Sharing
API Keys
Data Import / Export
Add Visualizations
Add Widgets
Export recent data
MATLAB Analysis
MATLAB Visualization
Channel Stats
Created: a day ago
Last entry: a day ago
Entries: 75
Field 1 Chart
Sensor data
Temperature
32
31.8
17:00
17:15
17:30
Date
ThingSpeak.com
Type here to search
21°C
ENG
20:13
18-10-2022