

Week 7

```
#include <DHT.h>
#include <DHT_U.h>
#include <ESP8266WiFi.h>
String apiKey = "ZYUP9R7N150EBYRO"; //
const char *ssid = "surekha";
const char *pass = "sakhison";
const char* server = "api.thingspeak.com";
#define DHTPIN D3
DHT dht(DHTPIN, DHT11);
WiFiClient client;

void setup()
{
  Serial.begin(115200);
  delay(1000);
  dht.begin();
  Serial.println("Connecting to ");
  Serial.println(ssid);
  WiFi.begin(ssid, pass);
  while (WiFi.status() != WL_CONNECTED)
  {
    delay(2000);
    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)) // "184.106.153.149" or api.thingspeak.com
  {
    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(1000);
}

```

Output

```

. Variables and constants in RAM (global, static), used 28672 / 80192 bytes (35%)
| SEGMENT BYTES DESCRIPTION
|-----|-----|-----|
| DATA   1568   initialized variables
| RODATA   1292   constants
| BSS     25872   zeroed variables
. Instruction RAM (IRAM_ATTR, ICACHE_RAM_ATTR), used 60331 / 65536 bytes (92%)
| SEGMENT BYTES DESCRIPTION
|-----|-----|-----|
| ICACHE   32768   reserved space for flash instruction cache
| IROM     27563   code in IROM
. Code in flash (default, ICACHE_FLASH_ATTR), used 248692 / 1048576 bytes (23%)
| SEGMENT BYTES DESCRIPTION
|-----|-----|-----|
| IROM     248692   code in flash

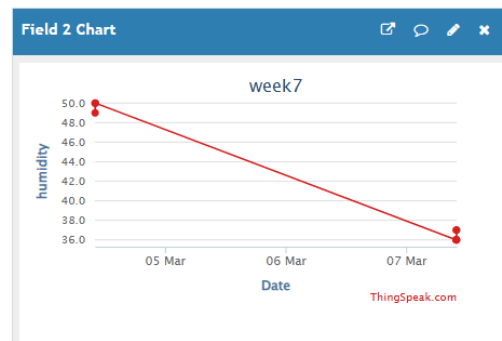
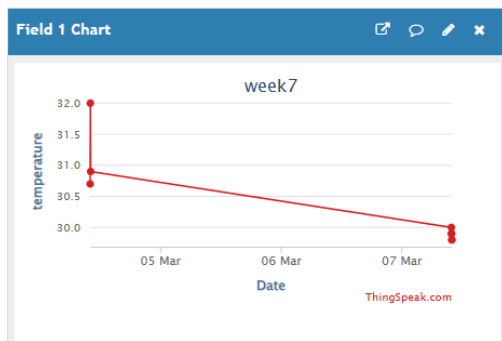
esptool.py v3.0
Serial port COM12
Connecting...
Chip is ESP8266EX
Features: WiFi
Crystal is 26MHz
MAC: 8c:aa:b5:6e:cd:40
Uploading stub...
Running stub...
Stub running...
Configuring flash size...
Auto-detected Flash size: 4MB
Compressed 283200 bytes to 207813...
Writing at 0x00000000... (7 %)
Writing at 0x00004000... (15 %)
Writing at 0x00008000... (23 %)
Writing at 0x0000c000... (30 %)
Writing at 0x00010000... (38 %)
Writing at 0x00014000... (46 %)
Writing at 0x00018000... (53 %)
Writing at 0x0001c000... (61 %)
Writing at 0x00020000... (69 %)
Writing at 0x00024000... (76 %)
Writing at 0x00028000... (84 %)
Writing at 0x0002c000... (92 %)
Writing at 0x00030000... (100 %)
Wrote 283200 bytes (207813 compressed) at 0x00000000 in 18.5 seconds (effective 122.7 kbit/s)...
Hash of data verified.

Leaving...
Hard resetting via RTS pin...

```

```
ailed to read from DHT sensor!  
ailed to read from DHT sensor!  
ailed to read from DHT sensor!  
ailed to read from DHT sensor!  
ailed to read from DHT sensor!  
ailed to read from DHT sensor!  
ailed to read from DHT sensor!  
ailed to read from DHT sensor!  
ailed to read from DHT sensor!  
ailed to read from DHT sensor!  
ailed to read from DHT sensor!  
ailed to read from DHT sensor!  
ailed to read from DHT sensor!  
ailed to read from DHT sensor!  
ailed to read from DHT sensor!  
ailed to read from DHT sensor!  
ailed to read from DHT sensor!  
ailed to read from DHT sensor!  
ailed to read from DHT sensor!  
emperature: 32.90 degrees Celcius, Humidity: 43.00%. Send to Thingspeak.  
aiting...  
emperature: 32.90 degrees Celcius, Humidity: 41.00%. Send to Thingspeak.  
aiting...  
emperature: 32.90 degrees Celcius, Humidity: 41.00%. Send to Thingspeak.  
aiting...
```

Entries: 9



Week 8

```
#include "ThingSpeak.h"
#include <ESP8266WiFi.h>
#include<DHT.h>
const char ssid[] = "surekha"; // your network SSID (name)
const char pass[] = "sakhison"; // your network password
int statusCode = 0;
WiFiClient client;

//-----Channel Details-----//
unsigned long counterChannelNumber = 2846266; // Channel ID
const char * myCounterReadAPIKey = "SXJMCVLJWIZMZLLF"; // Read API Key
const int FieldNumber1 = 1; // The field you wish to read
const int FieldNumber2 = 2; // The field you wish to read
//-----//

void setup()
{
  Serial.begin(115200);
  WiFi.mode(WIFI_STA);
  ThingSpeak.begin(client);
}

void loop()
{
  //----- Network -----//
  if (WiFi.status() != WL_CONNECTED)
  {
    Serial.print("Connecting to ");
    Serial.print(ssid);
    Serial.println(" ....");
    while (WiFi.status() != WL_CONNECTED)
    {
      WiFi.begin(ssid, pass);
      delay(5000);
    }
    Serial.println("Connected to Wi-Fi Succesfully.");
  }
  //----- End of Network connection-----//

  //----- Channel 1 -----//
  long temp = ThingSpeak.readLongField(counterChannelNumber, FieldNumber1,
myCounterReadAPIKey);
  statusCode = ThingSpeak.getLastReadStatus();
  if (statusCode == 200)
  {
    Serial.print("Temperature: ");
    Serial.println(temp);
  }
  else
  {
    Serial.println("Unable to read channel / No internet connection");
  }
}
```

```

delay(100);
//----- End of Channel 1 -----//

//----- Channel 2 -----//
long humidity = ThingSpeak.readLongField(counterChannelNumber, FieldNumber2,
myCounterReadAPIKey);
statusCode = ThingSpeak.getLastReadStatus();
if (statusCode == 200)
{
    Serial.print("Humidity: ");
    Serial.println(humidity);
}
else
{
    Serial.println("Unable to read channel / No internet connection");
}
delay(100);
//----- End of Channel 2 -----//
}

```

```

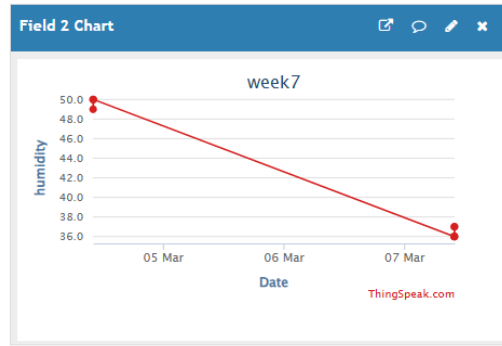
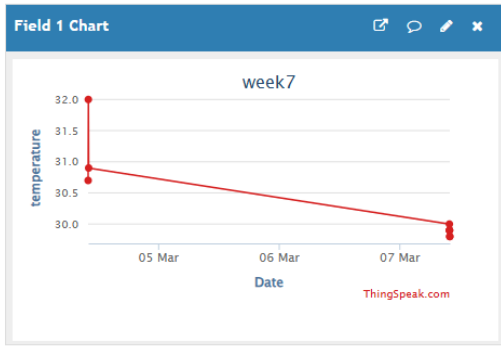
. Variables and constants in RAM (global, static), used 28864 / 80192 bytes (35%)
| SEGMENT BYTES DESCRIPTION
|-----|-----|
| DATA 1584 initialized variables
| RODATA 1264 constants
| BSS 26896 zeroed variables
| Instruction RAM (IRAM_ATTR, ICACHE_RAM_ATTR), used 59747 / 65536 bytes (91%)
| SEGMENT BYTES DESCRIPTION
|-----|-----|
| ICACHE 32768 reserved space for flash instruction cache
| IROM 26979 code in IROM
| Code in Flash (default, ICACHE_FLASH_ATTR), used 249012 / 1048576 bytes (23%)
| SEGMENT BYTES DESCRIPTION
|-----|-----|
| IROM 249012 code in flash

esptool.py v3.0
Serial port COM12
Connecting....
Chip is ESP8266EX
Features: WiFi
Crystal is 26MHz
MAC: 8c:aa:b5:6e:cd:40
Uploading stub...
Running stub...
Stub running...
Configuring flash size...
Auto-detected Flash size: 4MB
Compressed 282912 bytes to 207372...
Writing at 0x00000000... (7 %)
Writing at 0x00004000... (15 %)
Writing at 0x00008000... (23 %)
Writing at 0x0000c000... (30 %)
Writing at 0x00010000... (38 %)
Writing at 0x00014000... (46 %)
Writing at 0x00018000... (53 %)
Writing at 0x0001c000... (61 %)
Writing at 0x00020000... (69 %)
Writing at 0x00024000... (76 %)
Writing at 0x00028000... (84 %)
Writing at 0x0002c000... (92 %)
Writing at 0x00030000... (100 %)
Wrote 282912 bytes (207372 compressed) at 0x00000000 in 18.4 seconds (effective 122.8 kbit/s)...
Hash of data verified.

Leaving...
Hard resetting via RTS pin...

```

Entries: 9

[illegible]

Week 6

```
#include <ESP8266WiFi.h>

String apiKey = "5NAVIX2DRXW0AV41";
const char *ssid = "surekha";
const char *pass = "sakhison";
const char* server = "api.thingspeak.com";

#define IRpin D3

WiFiClient client;

int value;

void setup()
{
  Serial.begin(115200);
  pinMode(IRpin, INPUT);
  delay(1000);
}
```

```

Serial.println("Connecting to ");
Serial.println(ssid);
WiFi.begin(ssid, pass);
while (WiFi.status() != WL_CONNECTED)
{
    delay(1000);
    Serial.print(".");
}
Serial.println(" ");
Serial.println("WiFi connected");
}

void loop(){
    value = digitalRead(IRpin);
    Serial.println(value);
    if(value==0)
    {
        Serial.println("object detected");
    }
    else
    {
        Serial.println("no object detected");
    }
    if (client.connect(server,80))
    {
        String postStr = apiKey;
        postStr += "&field1=";
        postStr += String(value);
        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);
        client.stop();

        Serial.println("Waiting...");

        delay(1000);
    }
}

```