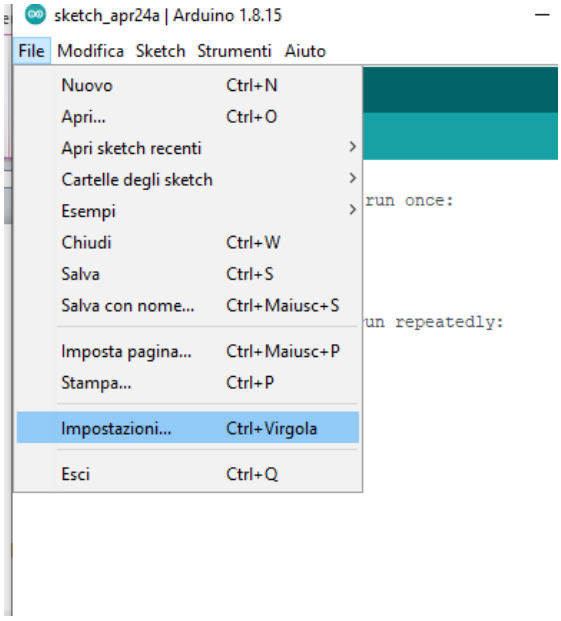
Dati sul web con 8266

ThingSpeak

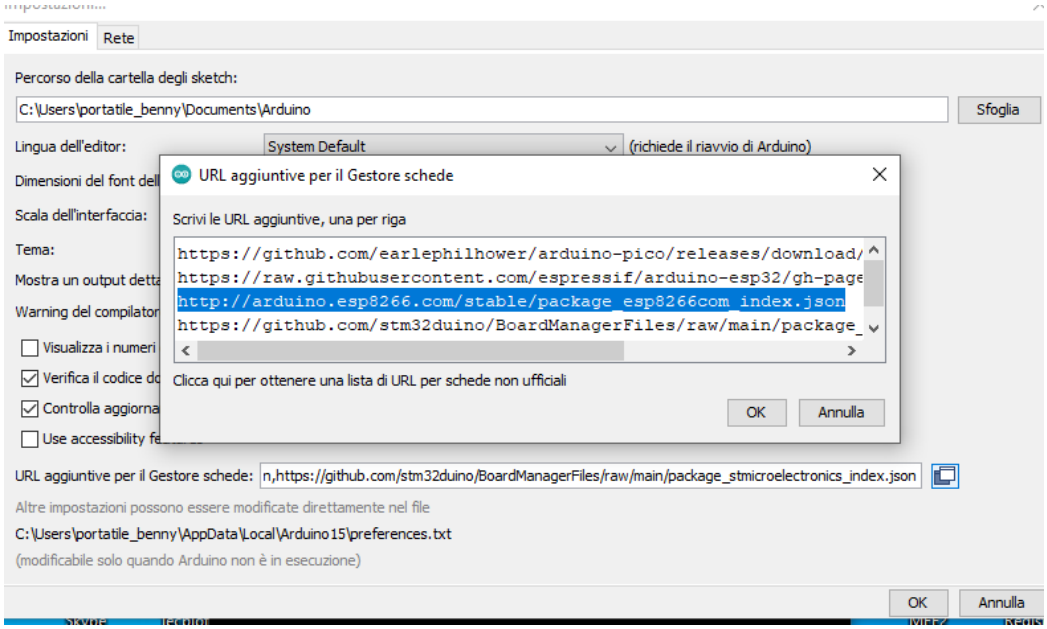
ThingSpeak™ è un servizio di analisi IoT che ti permette di aggregare, visualizzare e analizzare flussi di dati dal vivo nel cloud. ThingSpeak fornisce visualizzazioni istantanee dei dati inviati dai tuoi dispositivi a ThingSpeak. Con la possibilità di eseguire codice MATLAB® in ThingSpeak, è possibile eseguire analisi online ed elaborare i dati man mano che arrivano. ThingSpeak è spesso utilizzato per la creazione di prototipi e sistemi IoT proof-of-concept che richiedono analisi È possibile inviare dati da qualsiasi dispositivo connesso a Internet direttamente a ThingSpeak utilizzando un'API Rest o MQTT. Inoltre, le integrazioni cloud-to-cloud con The Things Network, Senet, il gateway Libelium Meshlium e Particle.io consentono ai dati dei sensori di raggiungere ThingSpeak su LoRaWAN® e connessioni cellulari 4G/3G. Con ThingSpeak, è possibile memorizzare e analizzare i dati nel cloud senza configurare i server web, e si possono creare sofisticati avvisi e-mail basati su eventi che si attivano in base ai dati in arrivo dai dispositivi collegati.

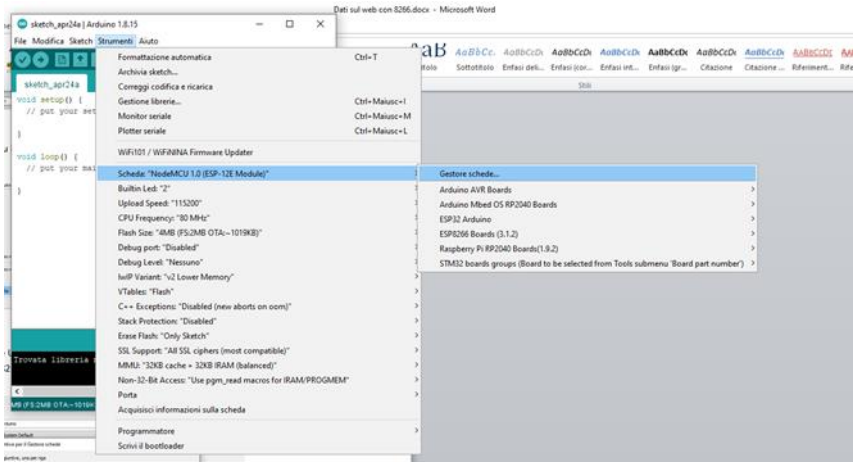
https://www.directindustry.it/prod/the-mathworks/product-12865-1892216.html

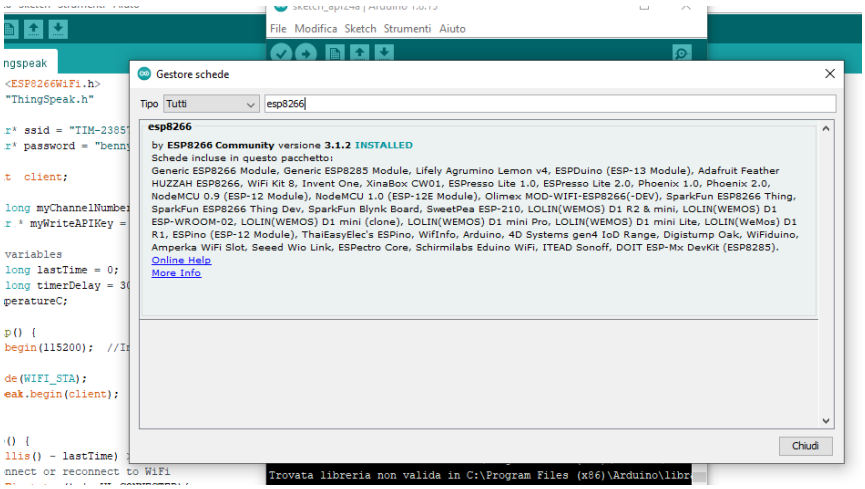
- Istallare esp8266 su IDE Arduino
  In impostazioni



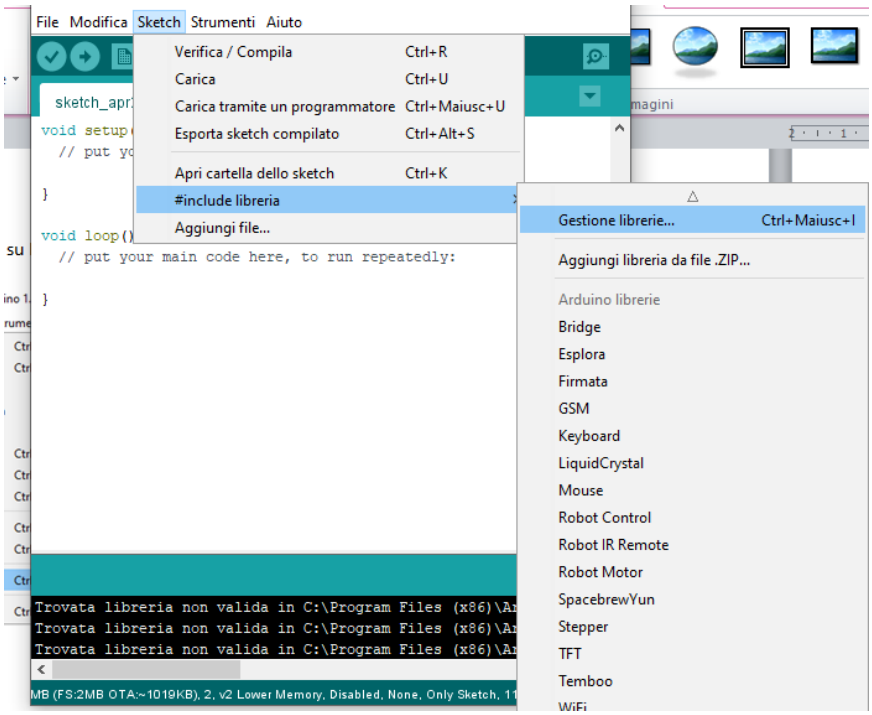
  Copiare il seguente URL
  http://arduino.esp8266.com/stable/package_esp8266com_index.json

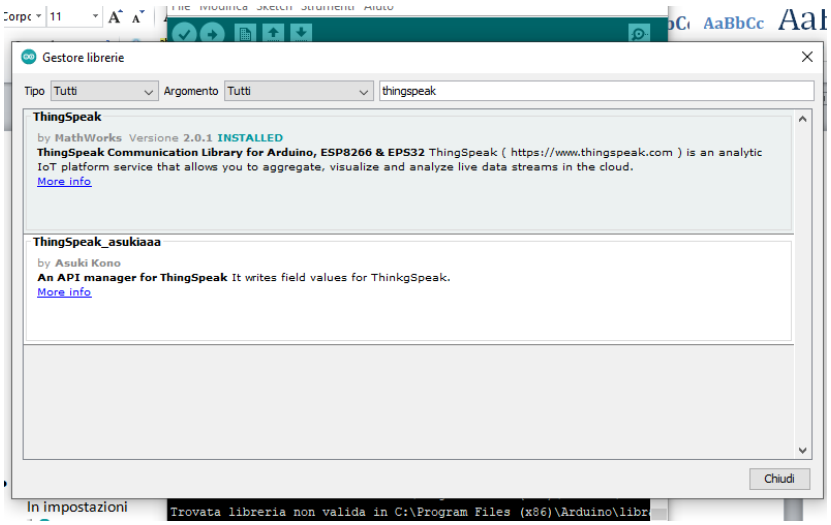




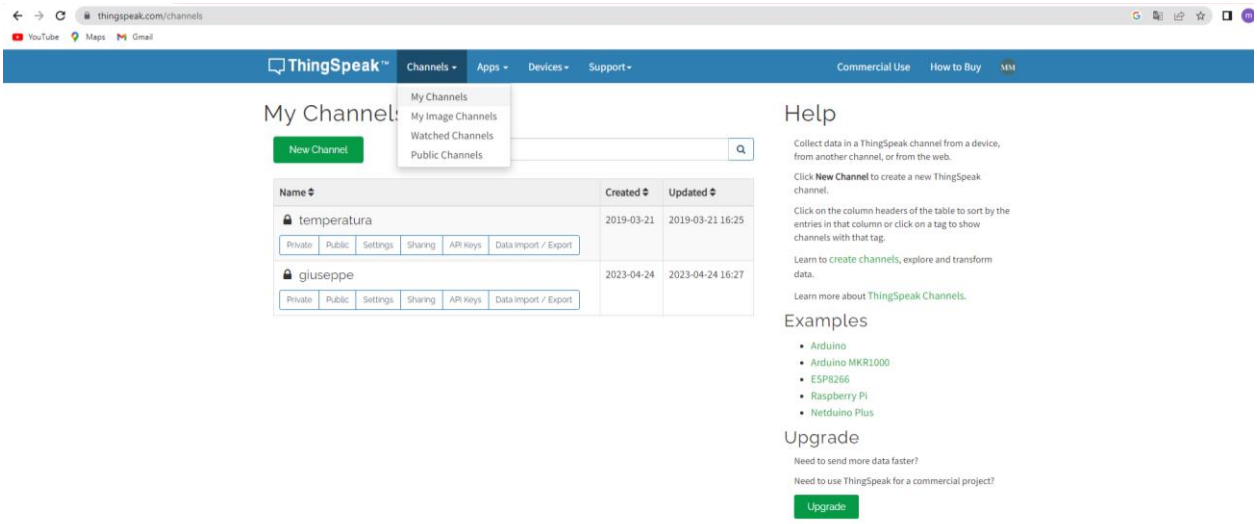


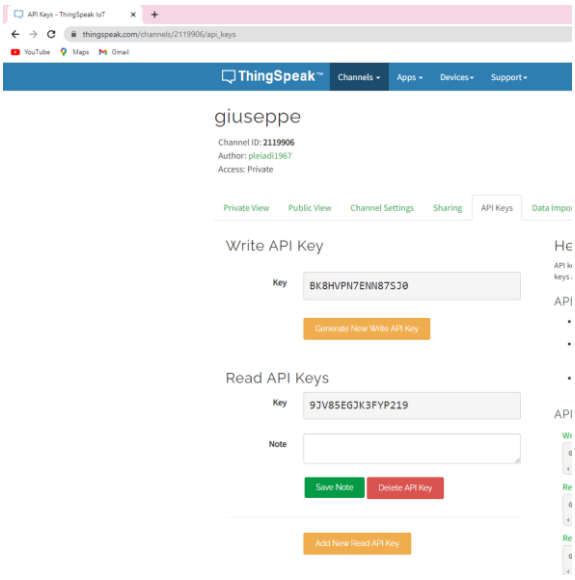
- Installare thingSpeak





- Registrazione sul sito thingspeak





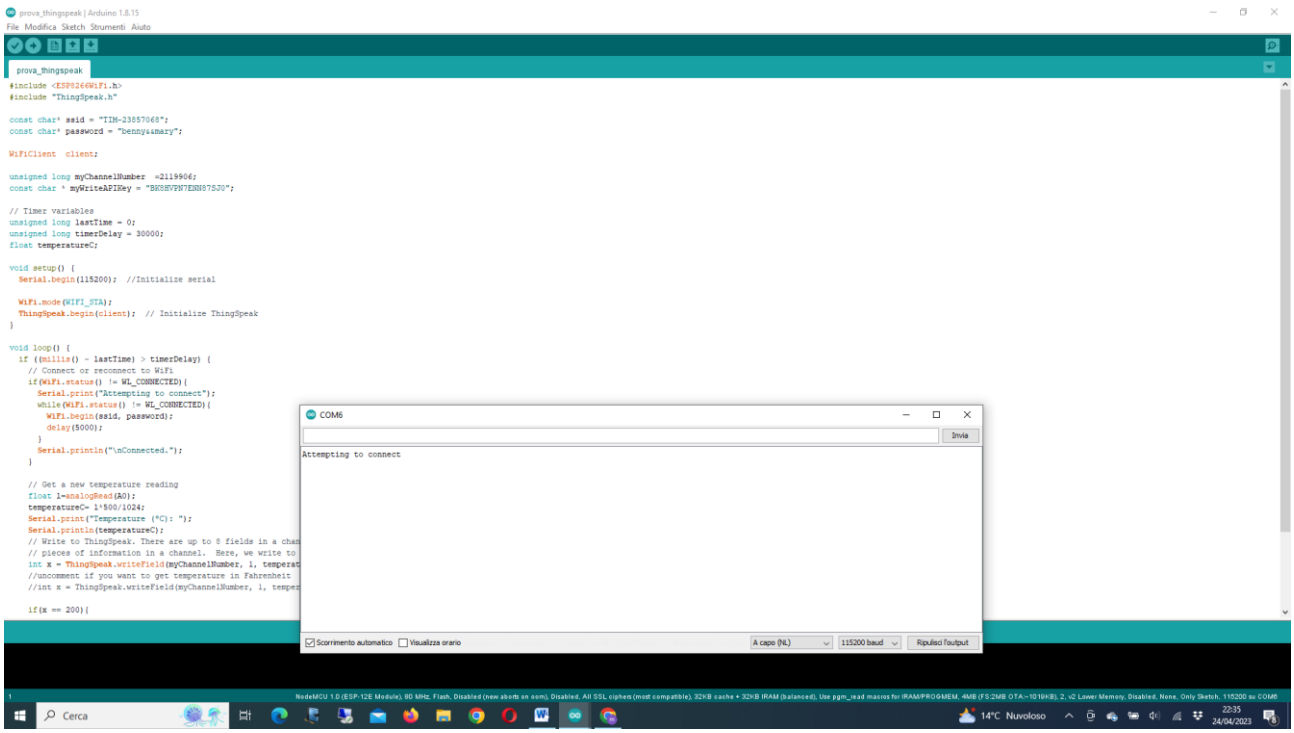
- **Programma**

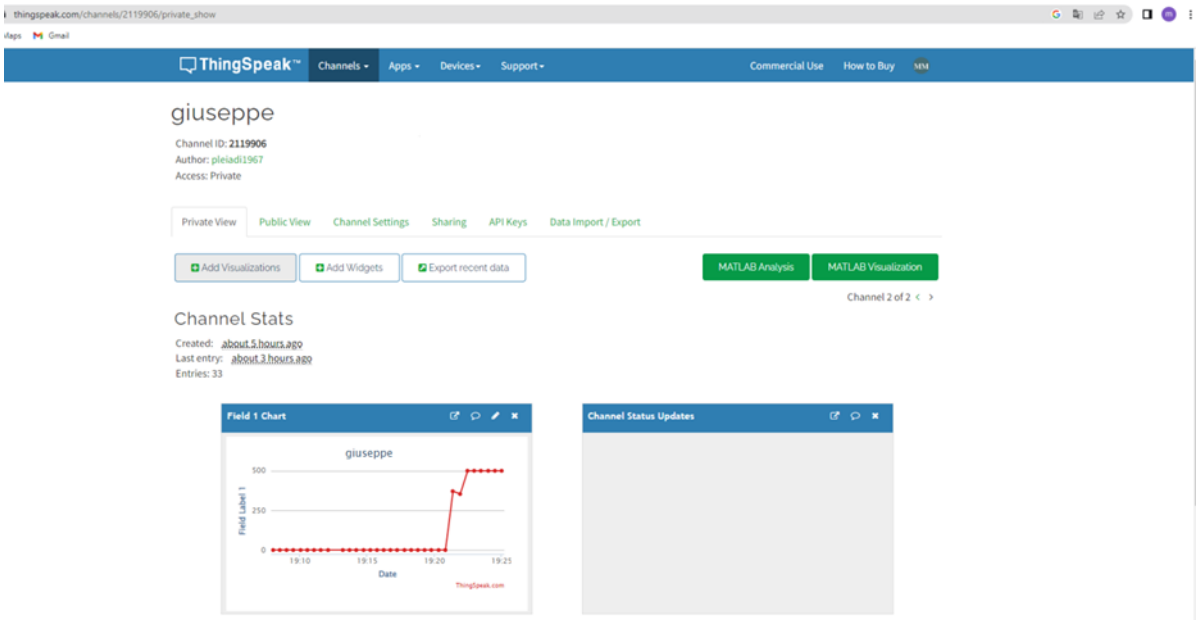
```
#include <ESP8266WiFi.h>

#include "ThingSpeak.h"

const char* ssid = "aaaa";

const char* password = "kkkk";

WiFiClient  client;

unsigned long myChannelNumber  =y;

const char * myWriteAPIKey = "xxxxxxxxx";
```

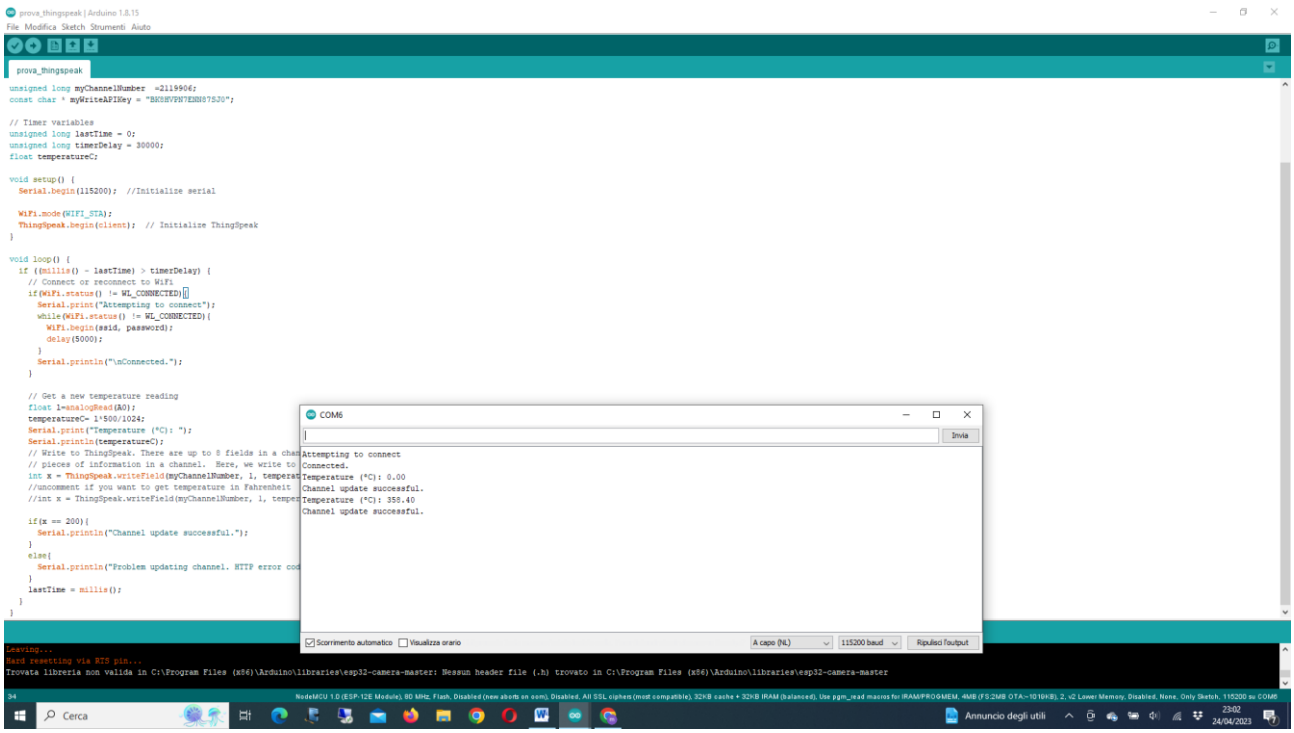
```
// Timer variables
unsigned long lastTime = 0;
unsigned long timerDelay = 30000;
float temperatureC;
void setup() {
  Serial.begin(115200);
  WiFi.mode(WIFI_STA);
  ThingSpeak.begin(client);  // Initialize ThingSpeak
}
void loop() {
  if ((millis() - lastTime) > timerDelay) {
    // Connect or reconnect 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 variable reading
    float l=analogRead(A0);
    temperatureC= l*500/1024;
    Serial.print("Temperature (ºC): ");
    Serial.println(temperatureC);
    // Write to ThingSpeak. There are up to 8 fields in a channel, allowing you to store up to 8 different
    // pieces of information in a channel.  Here, we write to field 1.
    int x = ThingSpeak.writeField(myChannelNumber, 1, temperatureC, myWriteAPIKey);
```

```
  //uncomment if you want to get temperature in Fahrenheit
  //int x = ThingSpeak.writeField(myChannelNumber, 1, temperatureF, myWriteAPIKey);
  if(x == 200){
    Serial.println("Channel update successful.");
  }
  else{
    Serial.println("Problem updating channel. HTTP error code " + String(x));
  }
  lastTime = millis();
 }
}
```

- **Programma in esecuzione**


prova_thingspeak | Arduino 1.8.15
File Modifica Sketch Strumenti Aiuto
prova_thingspeak
#include <ESP8266WiFi.h>
#include "ThingSpeak.h"
const char* ssid = "TIM-23857068";
const char* password = "benny&mary";
WiFiClient client;
unsigned long myChannelNumber =2119906;
const char * myWriteAPIKey = "BK8HVPN7EHN87SJ0";
// Timer variables
unsigned long lastTime = 0;
unsigned long timerDelay = 30000;
float temperatureC;
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
float l=analogRead(A0);
temperatureC= l*500/1024;
Serial.print("Temperature (°C): ");
Serial.println(temperatureC);
// Write to ThingSpeak. There are up to 8 fields in a chan
// pieces of information in a channel. Here, we write to
int x = ThingSpeak.writeField(myChannelNumber, 1, temperat
//uncomment if you want to get temperature in Fahrenheit
//int x = ThingSpeak.writeField(myChannelNumber, 1, tempe
if(x == 200){
COM6
Invia
Attempting to connect
Scorrimento automatico
Visualizza orario
A capo (NL)
115200 baud
Ripulisci l'output
NodeMCU 1.0 (ESP-12E Module), 80 MHz, Flash, Disabled (new aborts on oom), Disabled, All SSL ciphers (most compatible), 32KB cache + 32KB IRAM (balanced), Use pgm_read macros for IRAM/PROGMEM, 4MB (FS:2MB OTA:~1019KB), 2, v2 Lower Memory, Disabled, None, Only Sketch, 115200 su COM6
Cerca
14°C Nuvoloso
22:35
24/04/2023


thingspeak.com/channels/2119906/private_show
ThingSpeak™
Channels
Apps
Devices
Support
Commercial Use
How to Buy
giuseppe
Channel ID: 2119906
Author: pleiadi1967
Access: Private
Private View
Public View
Channel Settings
Sharing
API Keys
Data Import / Export
Add Visualizations
Add Widgets
Export recent data
MATLAB Analysis
MATLAB Visualization
Channel 2 of 2
Channel Stats
Created: about 5 hours ago
Last entry: about 3 hours ago
Entries: 33
Field 1 Chart
giuseppe
500
250
0
Field Label 1
19:10
19:15
19:20
19:25
Date
ThingSpeak.com
Channel Status Updates


prova_thingspeak | Arduino 1.8.15
File Modifica Sketch Strumenti Aiuto
prova_thingspeak
unsigned long myChannelNumber =2119906;
const char * myWriteAPIKey = "BK8HVPN7EHN87S3G";
// Timer variables
unsigned long lastTime = 0;
unsigned long timerDelay = 30000;
float temperatureC;
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
float l=analogRead(A0);
temperatureC= l*500/1024;
Serial.print("Temperature (°C): ");
Serial.println(temperatureC);
// Write to ThingSpeak. There are up to 8 fields in a chan
// pieces of information in a channel. Here, we write to
int x = ThingSpeak.writeField(myChannelNumber, 1, temperat
//uncomment if you want to get temperature in Fahrenheit
//int x = ThingSpeak.writeField(myChannelNumber, 1, temper
if(x == 200){
Serial.println("Channel update successful.");
}
else{
Serial.println("Problem updating channel. HTTP error cod
}
lastTime = millis();
}
}
COM6
Invia
Attempting to connect
Connected.
Temperature (°C): 0.00
Channel update successful.
Temperature (°C): 359.40
Channel update successful.
Scorrimento automatico
Visualizza orario
A capo (NL)
115200 baud
Ripulisci l'output
Leaving...
Hard resetting via RTS pin...
Trovata libreria non valida in C:\Program Files (x86)\Arduino\libraries\esp32-camera-master: Nessun header file (.h) trovato in C:\Program Files (x86)\Arduino\libraries\esp32-camera-master
34
NodeMCU 1.0 (ESP-12E Module), 80 MHz, Flash, Disabled (new aborts on oom), Disabled, All SSL ciphers (most compatible), 32KB cache + 32KB IRAM (balanced), Use pgm_read macros for IRAM/PROGMEM, 4MB (FS:2MB OTA:~1019KB), 2, v2 Lower Memory, Disabled, None, Only Sketch, 115200 su COM6
Cerca
Annuncio degli utili
23:02
24/04/2023