SUN ROBOTICS

IDEASPARK
WEATHER STATION KIT

www.sunrobotics.co.in

SunRobotics Technologies
A/302, Swaminarayan Avenue, Vasna – A'bad – Gujarat – India
Ph no: 079 26608128 : support@sunrobotics.co.in

Preface

Sun Robotics is a technical service team of open source software and hardware. Dedicated to applying the Internet and the latest industrial technology in open source area, we strive to provide best hardware support and software service for general makers and electronic enthusiasts around the world. We aim to create infinite possibilities with sharing. No matter what field you are in, we can lead you into the electronic world and bring your ideas into reality.

This is ideaspeak weather station kit for Arduino. Some common electronic components and sensors are included. Through the learning, you will get a better understanding of Arduino, and be able to make fascinating works based on Arduino.

Contents

Getting Started with Arduino

What is an Arduino?

Arduino is an open-source physical computing platform designed to make experimenting with electronics more fun and intuitive. Arduino has its own unique, simplified programming language, a vast support network, and thousands of potential uses, making it the perfect platform for both beginner and advanced DIY enthusiasts.

www.arduino.cc

A Computer for the Physical World:

The friendly blue board in your hand (or on your desk) is the Arduino. In some ways you could think of Arduino as the child of traditional desktop and laptop computers. At its roots, the Arduino is essentially a small portable computer. It is capable of taking inputs (such as the push of a button or a reading from a light sensor) and interpreting that information to control various outputs (like a blinking LED light or an electric motor). That's where the term "physical computing" is born - an Arduino is capable of taking the world of electronics and relating it to the physical world in a real and tangible way. Trust us - this will all make more sense soon.

Arduino UNO SMD R3

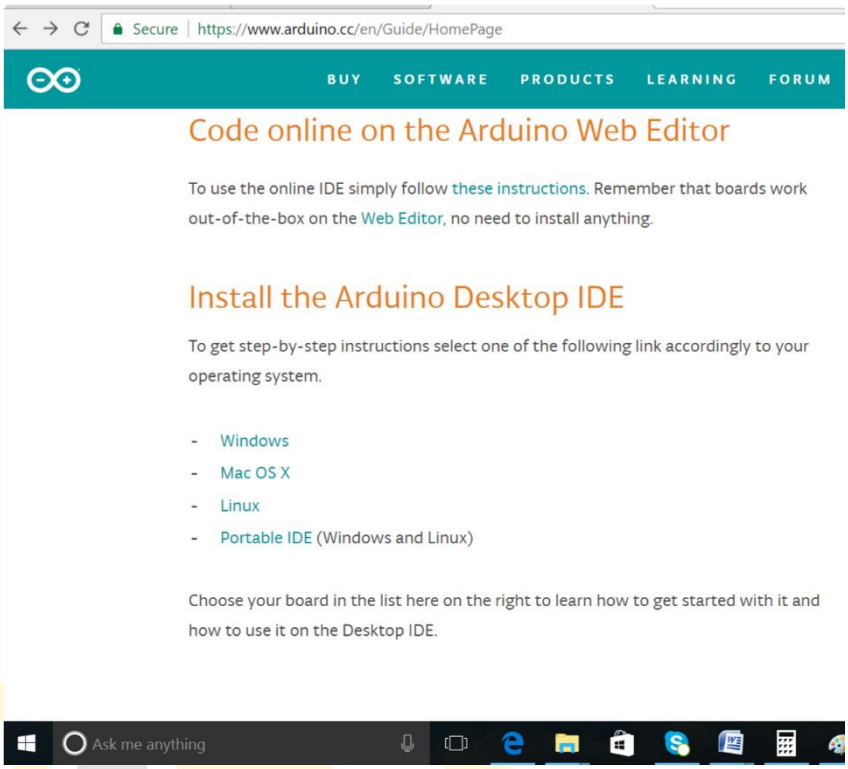
The Arduino Uno is one of several development boards based on the ATmega328. We like it mainly because of its extensive support network and its versatility. It has 14 digital input/output pins (6 of which can be PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. Don't worry, you'll learn about all these later.

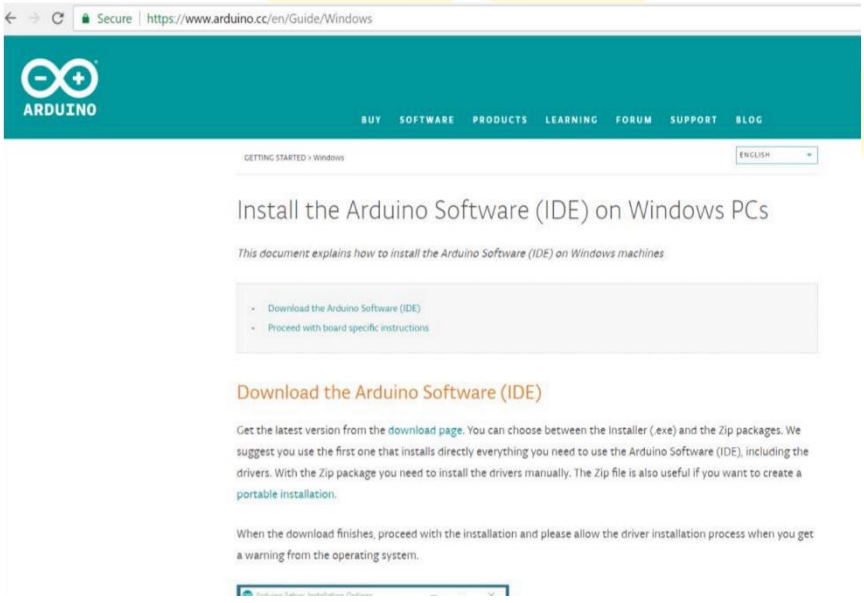
Installing Arduino IDE and Using Uno R3 board

STEP-1: Download the Arduino IDE (Integrated Development Environment)

Access the Internet: In order to get your Arduino up and running, you'll need to download some software first from www.arduino.cc (it's free!). This software, known as the Arduino IDE, will allow you to program the Arduino to do exactly what you want. It's like a word processor for writing programs. With an internet-capable computer, open up your favorite browser and type in the following URL into the address bar:

www.arduino.cc/en/Main/Software


Secure | https://www.arduino.cc/en/Guide/HomePage
BUY SOFTWARE PRODUCTS LEARNING FORUM
Code online on the Arduino Web Editor
To use the online IDE simply follow these instructions. Remember that boards work out-of-the-box on the Web Editor, no need to install anything.
Install the Arduino Desktop IDE
To get step-by-step instructions select one of the following link accordingly to your operating system.
- Windows
- Mac OS X
- Linux
- Portable IDE (Windows and Linux)
Choose your board in the list here on the right to learn how to get started with it and how to use it on the Desktop IDE.
Ask me anything


Secure | https://www.arduino.cc/en/Guide/Windows
ARDUINO
BUY SOFTWARE PRODUCTS LEARNING FORUM SUPPORT BLOG
GETTING STARTED > Windows
ENGLISH
Install the Arduino Software (IDE) on Windows PCs
This document explains how to install the Arduino Software (IDE) on Windows machines
Download the Arduino Software (IDE)
Proceed with board specific instructions
Download the Arduino Software (IDE)
Get the latest version from the download page. You can choose between the Installer (.exe) and the Zip packages. We suggest you use the first one that installs directly everything you need to use the Arduino Software (IDE), including the drivers. With the Zip package you need to install the drivers manually. The Zip file is also useful if you want to create a portable installation.
When the download finishes, proceed with the installation and please allow the driver installation process when you get a warning from the operating system.

For different operating system platforms, the way of using Arduino IDE is different. Please refer to the following links: Windows User ： http://www.arduino.cc/en/Guide/Windows Mac OS
User:http://www.arduino.cc/en/Guide/MacOSXLinuxUserhttp://playground.arduino.cc/Learning/Linux For more detailed information about Arduino IDE, please refer to the following link: **http://www.arduino.cc/en/Guide/HomePage**

STEP-2: Connect your Arduino Uno to your Computer:

Use the USB cable provided in the kit to connect the Arduino to one of your computer's USB inputs.

STEP-3: Install Drivers

Depending on your computer's operating system, you will need to follow specific instructions. Please go to the URLs below for specific instructions on how to install the drivers onto your Arduino Uno.

Windows Installation Process: Go to the web address below to access the instructions for installations on a Windows-based computer.

http://arduino.cc/en/Guide/Windows

Macintosh OS X Installation Process: Macs do not require you to install drivers. Enter the following URL if you have questions. Otherwise proceed to next page.

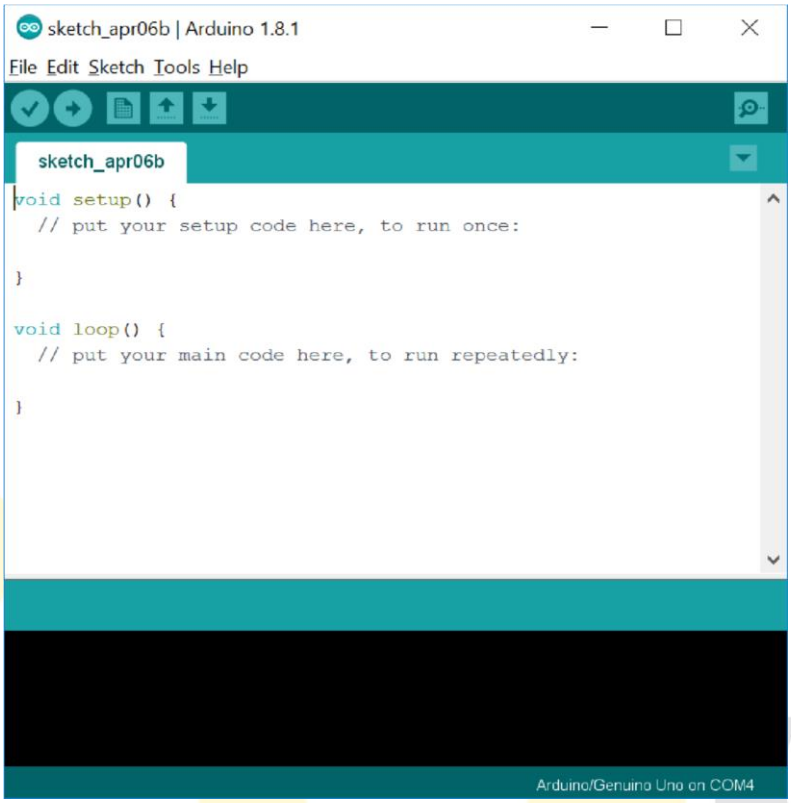
http://arduino.cc/en/Guide/MacOSX

Linux: 32 bit / 64 bit, Installation Process Go to the web address below to access the instructions for installations on a Linux-based computer.

http://www.arduino.cc/playground/Learning/Linux

STEP-4: Open the Arduino IDE

Open the Arduino IDE software on your computer. Poke around and get to know the interface. We aren't going to code right away, this is just an introduction. The step is to set your IDE to identify your Arduino Uno.



GUI (Graphical User Interface)

Verify

Checks your code for errors compiling it.

Upload

Compiles your code and uploads it to the configured board. See uploading below for Details .Note: If you are using an external programmer with your board, you can hold Down the "shift" key on your computer when using this icon. The text will change to "Upload using Programmer" New Creates a new sketch.

Open

Presents a menu of all the sketches in your sketchbook. Clicking one will open it within The current window overwriting its content. Note: due to a bug in Java, this menu Doesn't scroll; if you need to open a sketch late in the list, use the File | Sketch book Menu instead.

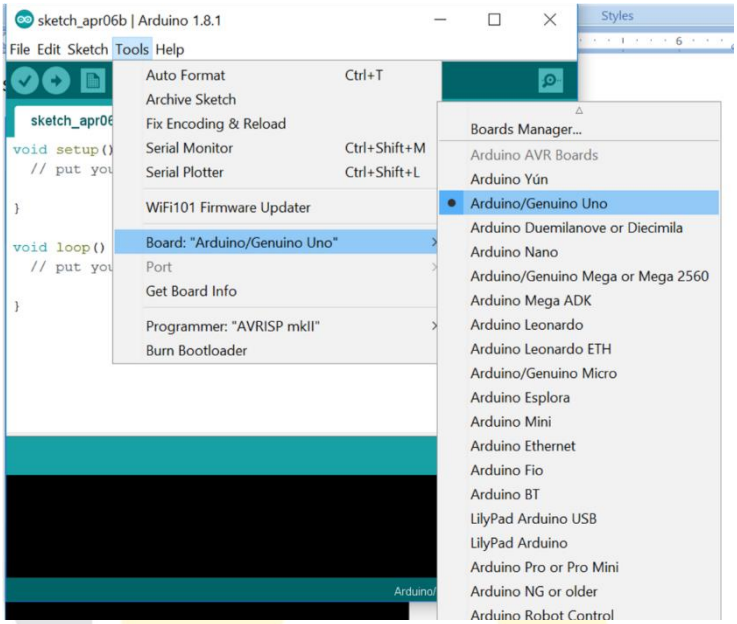
Save

Saves your sketch.

Serial Monitor

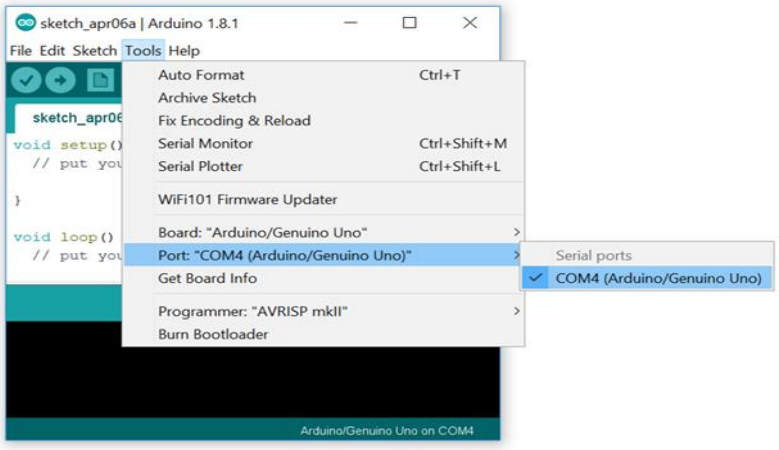
Opens the serial monitor.

STEP-5: Select your board: Arduino Uno



STEP-6: Select your Serial Device

Windows: Select the serial device of the Arduino board from the Tools | Serial Port menu. This is likely to be com3 or higher (COM1 and COM2 are usually reserved for hardware serial ports). To find out, you can disconnect your Arduino board and re-open the menu; the entry that disappears should be the Arduino board. Reconnect the board and select that serial port.



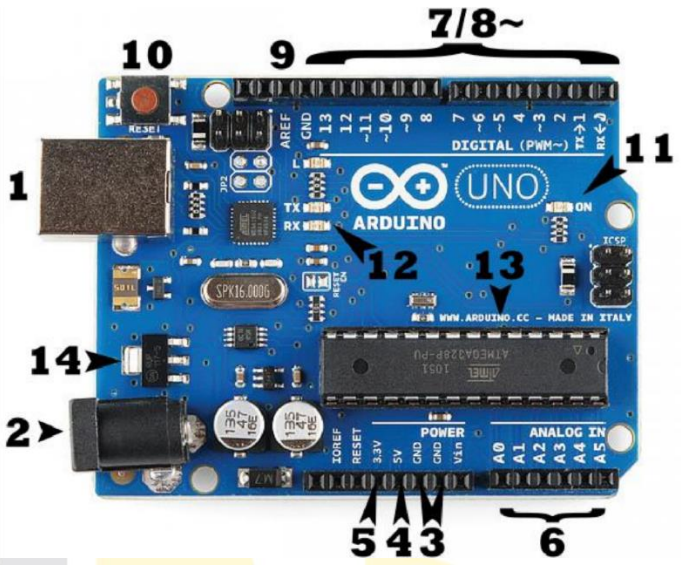
Mac OS: Select the serial device of the Arduino board from the Tools > Serial Port menu. On the Mac, this should be something with /dev/tty.usbmodem (for the Uno or Mega 2560) or /dev/tty.usbserial (for older boards) in it.

Linux: http://playground.arduino.cc/Learning/Linux

About Arduino Uno R3 board

What's on the board?

There are many varieties of Arduino boards that can be used for different purposes. Some boards look a bit different from the one below, but most Arduino have the majority of these components in common:



Power (USB / Barrel Jack)

Every Arduino board needs a way to be connected to a power source. The Arduino UNO can be powered from a USB cable coming from your computer or a wall power supply that is terminated in a barrel jack. In the picture above the USB connection is labeled **(1)** and the barrel jack is labeled **(2).** .

NOTE: Do NOT use a power supply greater than 20 Volts as you will overpower (and thereby destroy) your Arduino. The recommended voltage for most Arduino models is between 6 and 12 Volts. Pins (5V, 3.3V, GND, Analog, Digital, PWM, AREF)

The pins on your Arduino are the places where you connect wires to construct a circuit (probably in conjunction with a breadboard and some wire. They usually have black plastic 'headers' that allow you to just plug a wire right into the board. The Arduino has several different kinds of pins, each of which is labeled on the board and used for different functions.

- **GND (3):** Short for 'Ground'. There are several GND pins on the Arduino, any of which can be used to ground your circuit.
- **5V (4) & 3.3V (5):** As you might guess, the 5V pin supplies 5 volts of power, and the 3.3V pin supplies 3.3 volts of power. Most of the simple components used with the Arduino run happily off of 5 or 3.3 volts.
- **Analog (6):** The area of pins under the 'Analog In' label (A0 through A5 on the UNO) are Analog In pins. These pins can read the signal from an analog sensor (like a temperature sensor) and convert it into a digital value that we can read.
- **Digital (7)**: Across from the analog pins are the digital pins (0 through 13 on the UNO). These pins can be used for both digital input (like telling if a button is pushed) and digital output (like powering an LED).

- **PWM (8)**: You may have noticed the tilde (~) next to some of the digital pins (3, 5, 6, 9, 10, and 11 on the UNO). These pins act as normal digital pins, but can also be used for something called Pulse-Width Modulation (PWM). We have a tutorial on PWM, but for now, think of these pins as being able to simulate analog output (like fading an LED in and out).
- **AREF (9)**: Stands for Analog Reference. Most of the time you can leave this pin alone. It is sometimes used to set an external reference voltage (between 0 and 5 Volts) as the upper limit for the analog input pins.

Reset Button

Just like the original Nintendo, the Arduino has a reset button **(10)**. Pushing it will temporarily connect the reset pin to ground and restart any code that is loaded on the Arduino. This can be very useful if your code doesn't repeat, but you want to test it multiple times. Unlike the original Nintendo however, blowing on the Arduino doesn't usually fix any problems.

Power LED Indicator

Just beneath and to the right of the word "UNO" on your circuit board, there's a tiny LED next to the word 'ON' **(11)**. This LED should light up whenever you plug your Arduino into a power source. If this light doesn't turn on, there's a good chance something is wrong. Time to re-check your circuit!

TX RX LEDs

TX is short for transmit, RX is short for receive. These markings appear quite a bit in electronics to indicate the pins responsible for serial communication. In our case, there are two places on the Arduino UNO where TX and RX appear – once by digital pins 0 and 1, and a second time next to the TX and RX indicator LEDs **(12)**. These LEDs will give us some nice visual indications whenever our Arduino is receiving or transmitting data (like when we're loading a new program onto the board).

Main IC

The black thing with all the metal legs is an IC, or Integrated Circuit **(13)**. Think of it as the brains of our Arduino. The main IC on the Arduino is slightly different from board type to board type, but is usually from the AT mega line of IC's from the ATMEL company. This can be important, as you may need to know the IC type (along with your board type) before loading up a new program from the Arduino software. This information can usually be found in writing on the top side of the IC. If you want to know more about the difference between various IC's, reading the datasheets is often a good idea.

Voltage Regulator

The voltage regulator **(14)** is not actually something you can (or should) interact with on the Arduino. But it is potentially useful to know that it is there and what it's for. The voltage regulator does exactly what it says – it controls the amount of voltage that is let into the Arduino board. Think of it as a kind of gatekeeper; it will turn away an extra voltage that might harm the circuit. Of course, it has its limits, so don't hook up your Arduino to anything greater than 20 volts.

WEATHER STATION

Component List:

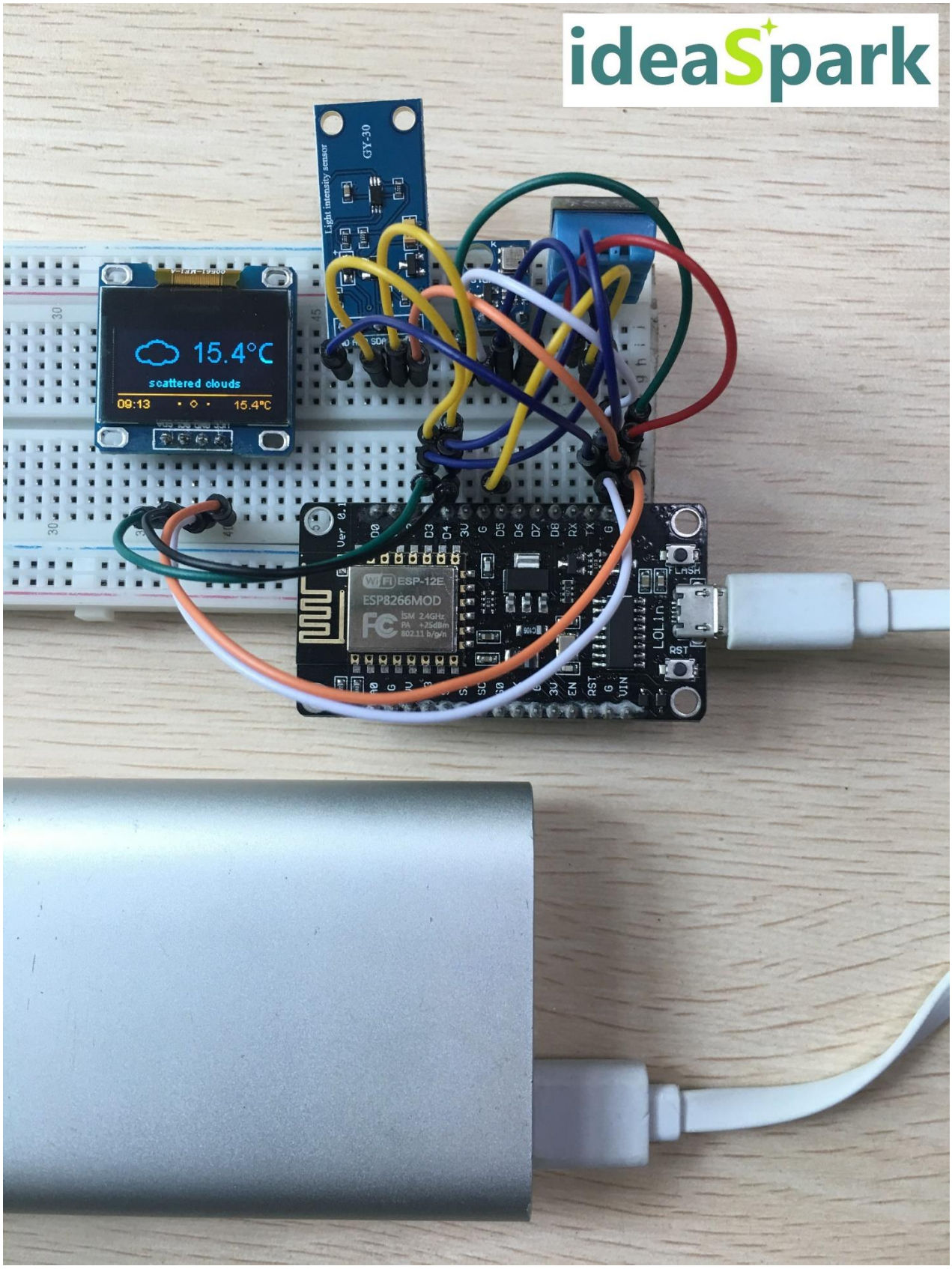
- 1 x ESP8266 -12E
- 1 x DHT11
- 1 x BMP180
- 1 x BH1750FVI
- 1 x OLED Display
- 1 x USB Cable
- 2 x Mini Breadboard
- 20 x Jumper wire

Weather station by Arduino IDE:

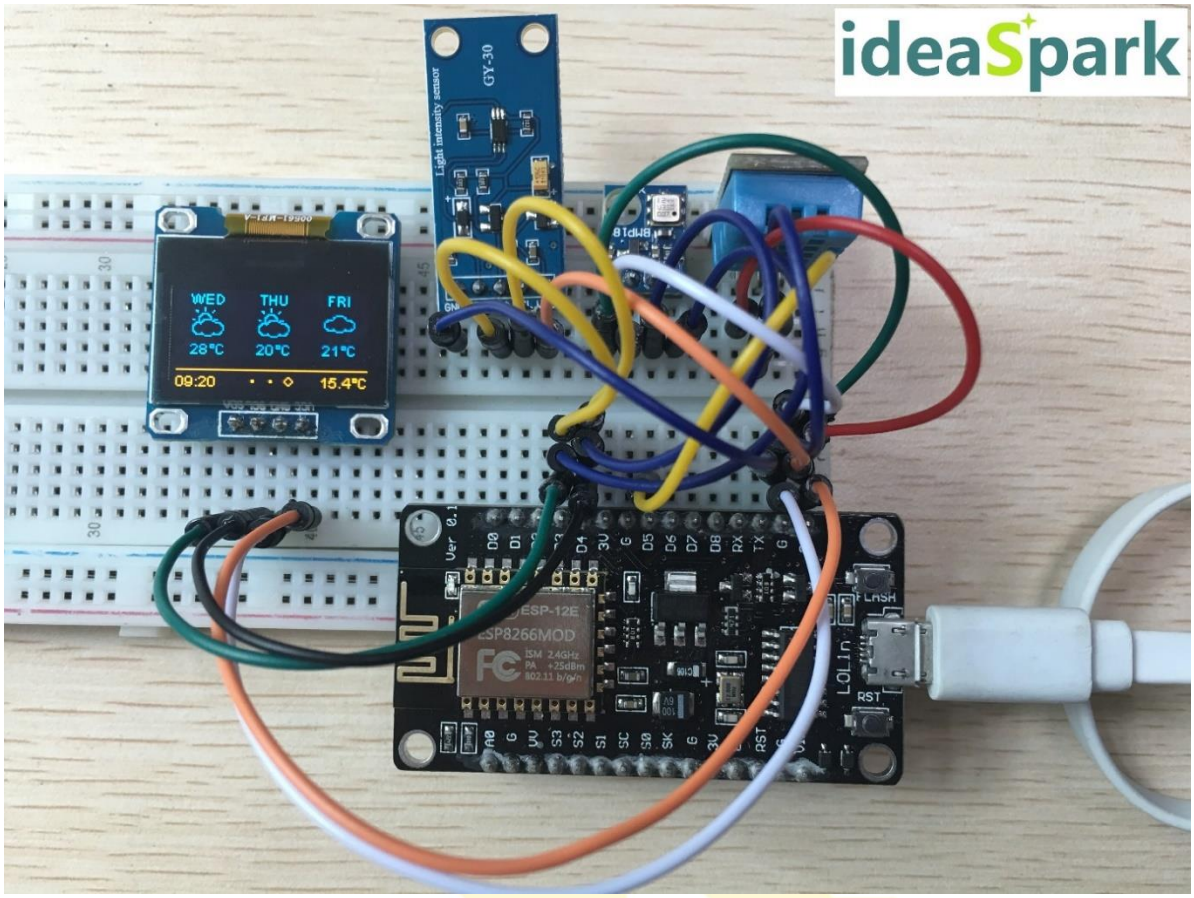
Features:

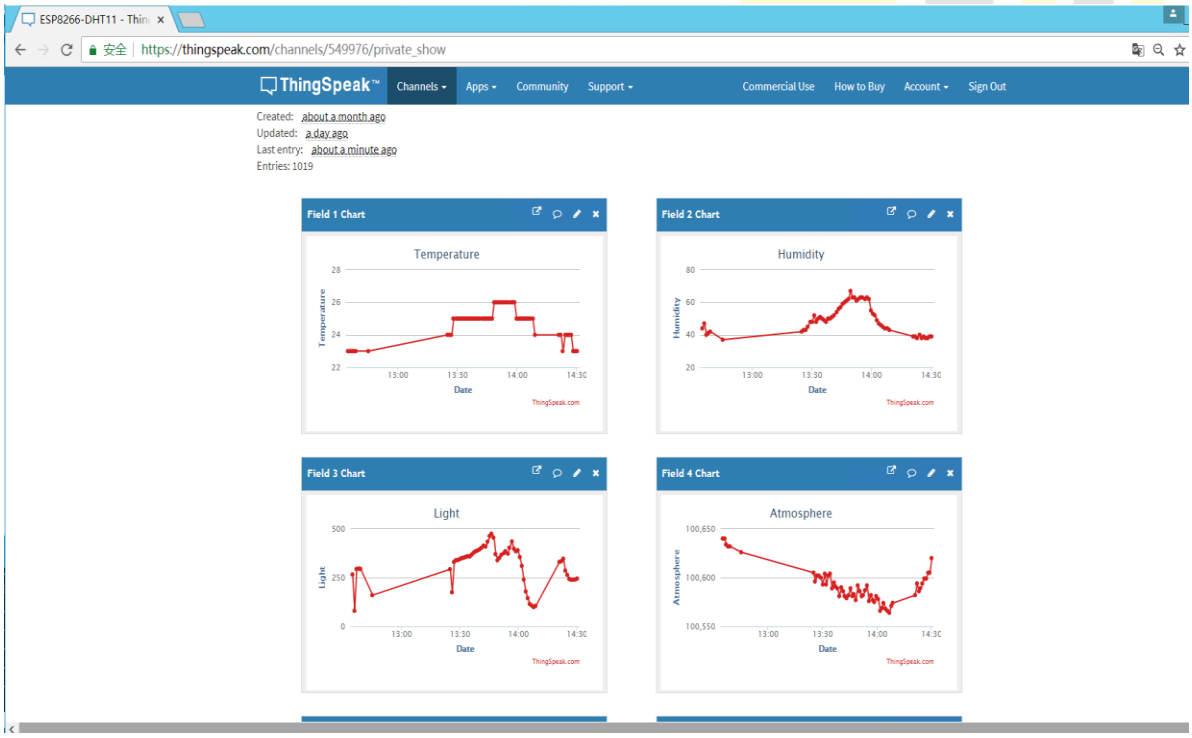
1) Get weather data from OpenWeathermap, showing the weather of today and the weather forecast for the next 3 days in any city in the world.
2) Read the current temperature, humidity.
3) Read atmospheric pressure and light intensity.
4) Upload temperature, humidity, atmospheric pressure and light intensity data to thingspeak.com at regular intervals.
5) View weather forecast on Display and view environmental monitoring chart on thingspeak.com

Result:



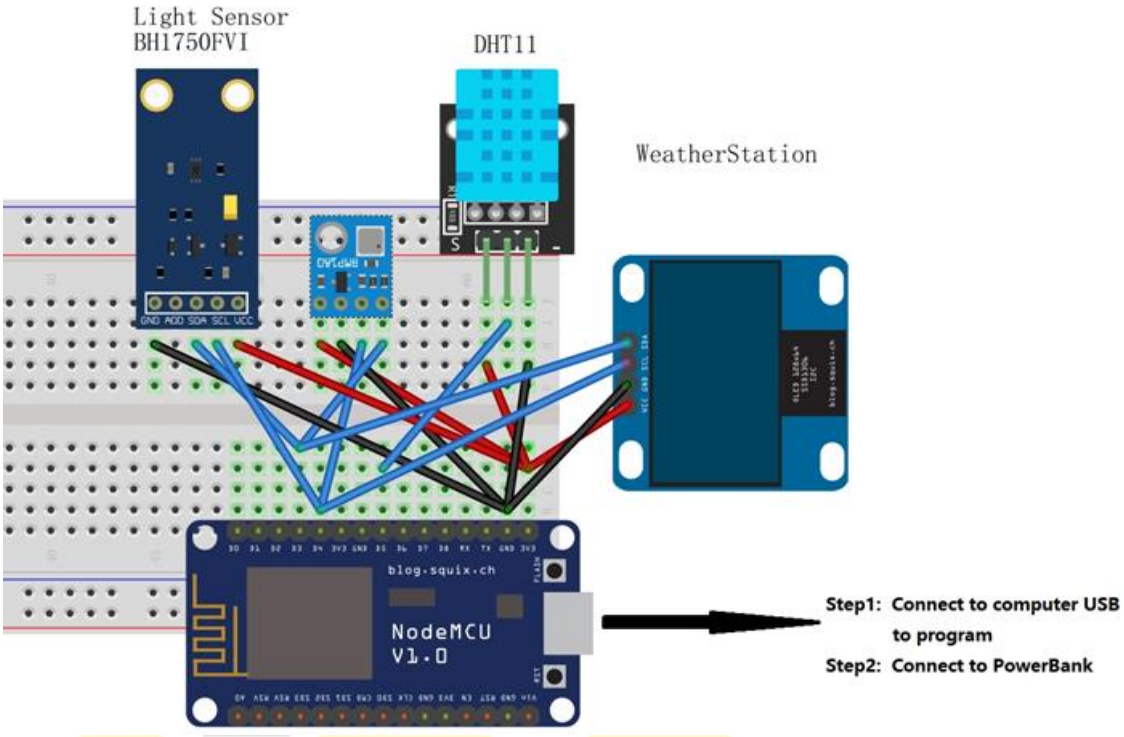
Web: www.sunrobotics.co.in
Email: support@sunrobotics.co.in





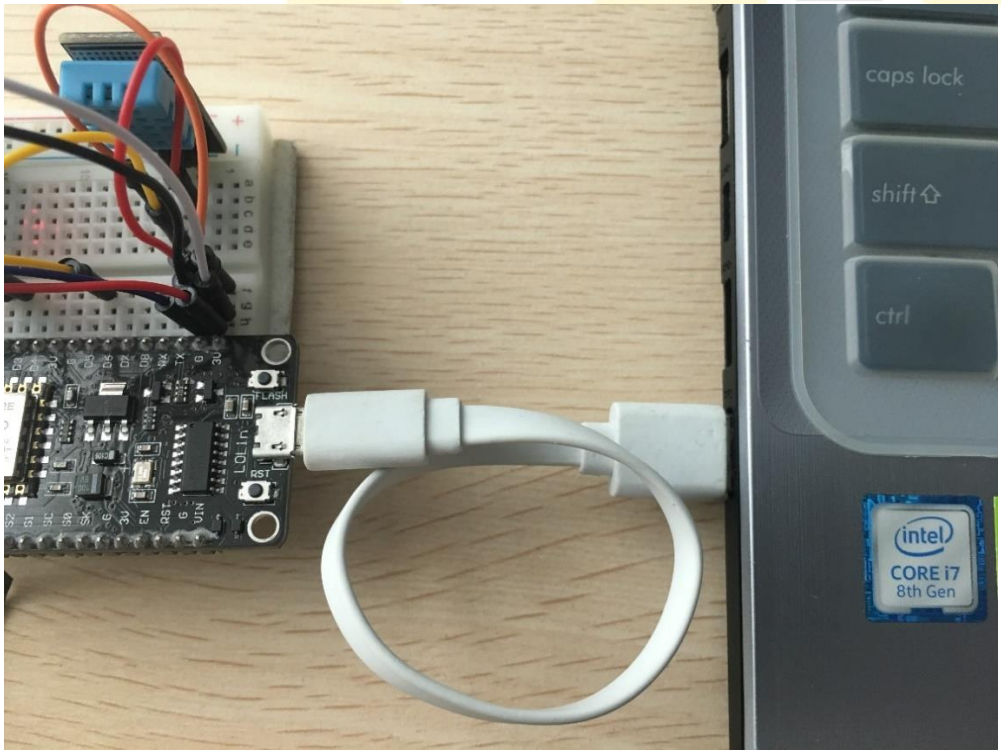
Web: www.sunrobotics.co.in
Email: support@sunrobotics.co.in

Breadboard Connection:

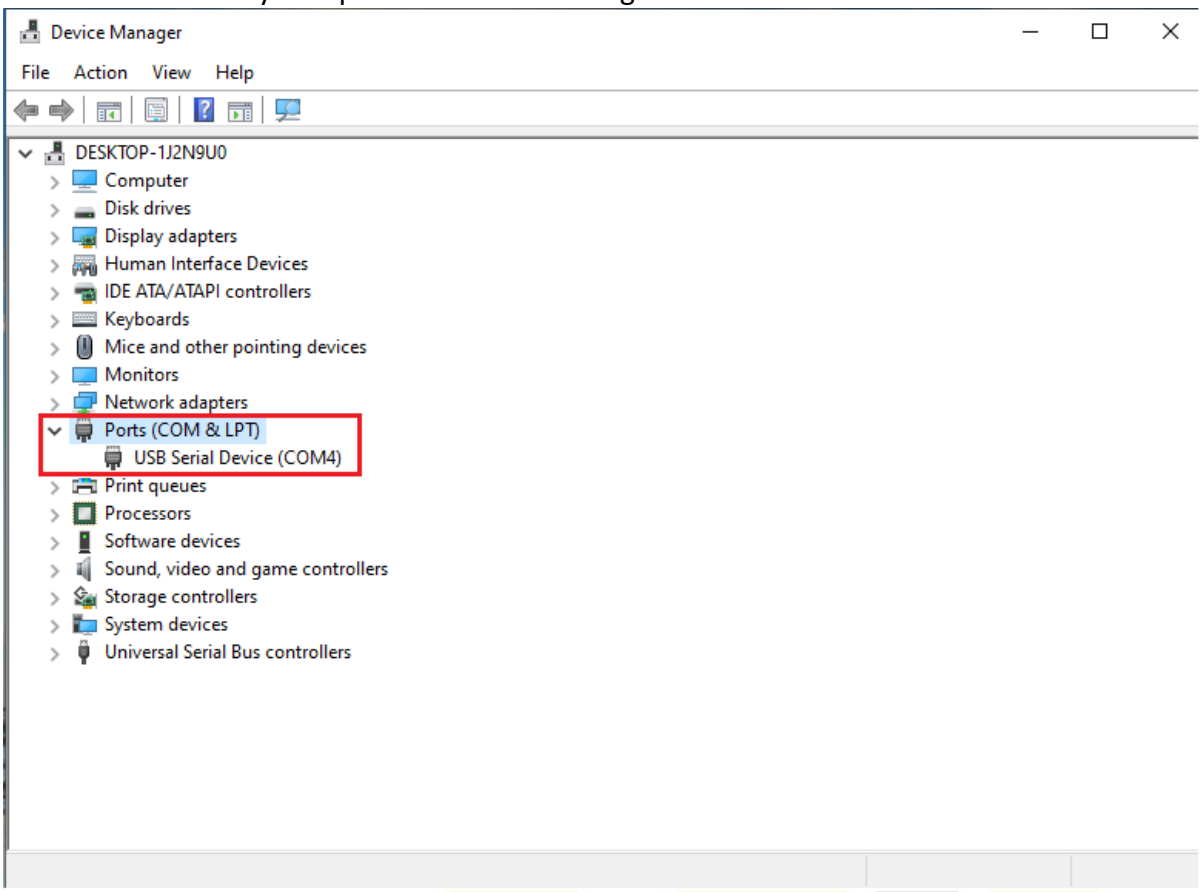


Step 1: Install USB-To-Serial.

1) We need computer to recognize the Serial Port, so install USB-to-Serial
   usb-to-serial-win10.rar
2) After the installation is successful, connect as shown below. Your computer will recognize ESP8266:



3) You can see it on My Computer -> Device Manager：



Step 2: Install Arduino IDE for ESP8266

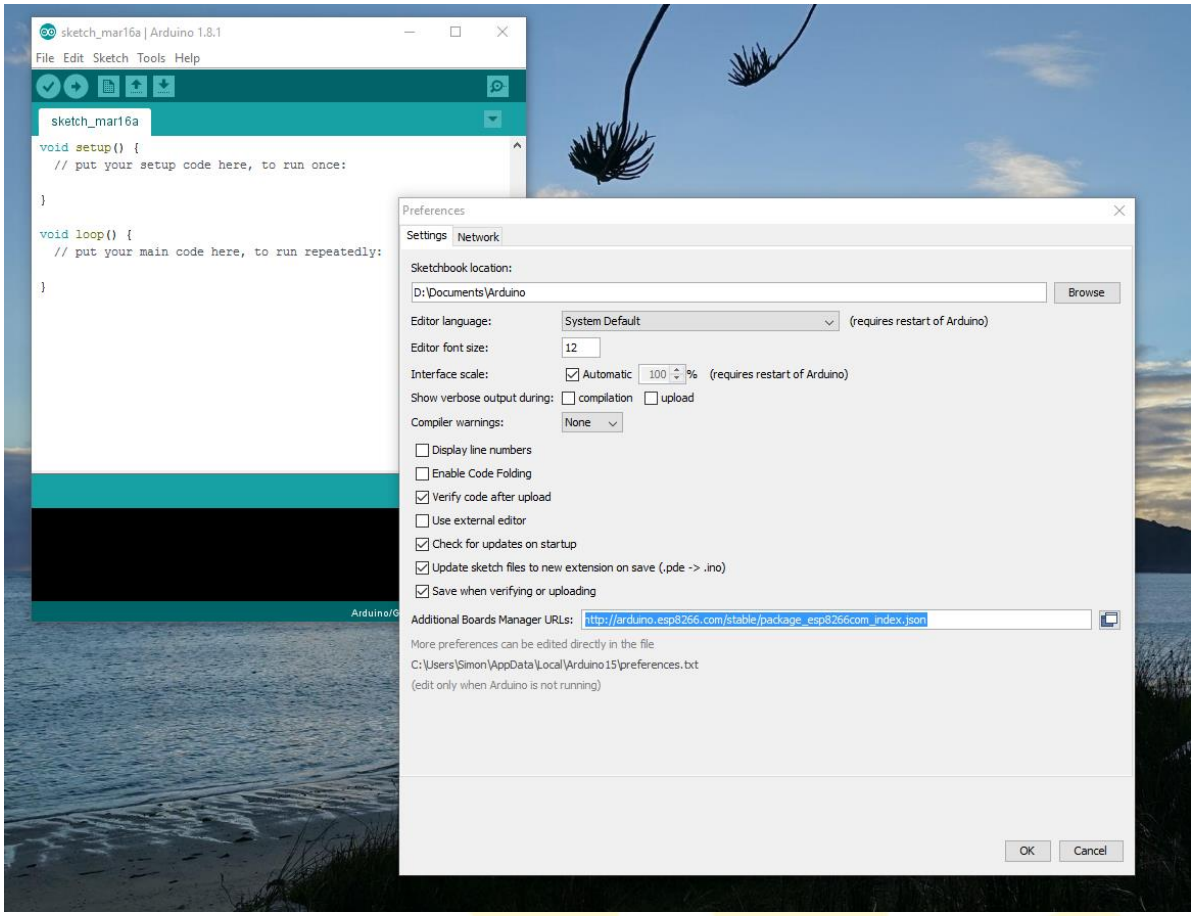
Only Arduino IDE version 1.8.4 is recommended, other version may can't work with the weather station kits source code.

Arduino IDE 1.8.4 download URL:

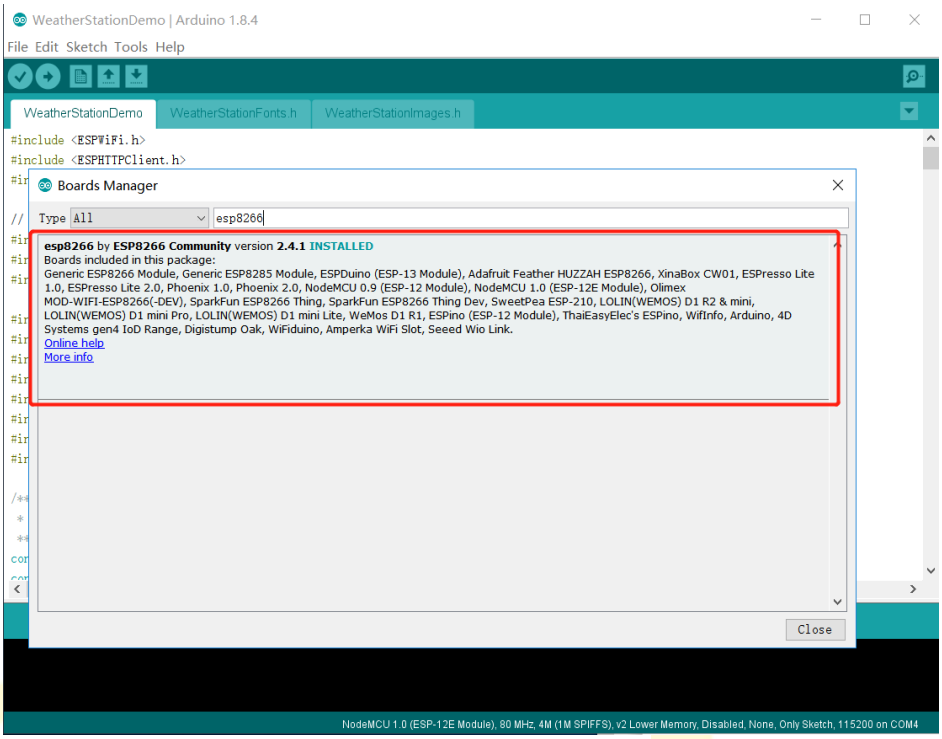
https://www.amazon.com/clouddrive/share/5hYUsaMRYFatPBizWinTapxlSzqdW8LQtVyiihAYIS5

Open Arduino IDE
File > Preferences
Insert the following link in the "Additional Boards Manager URLs:" text box：
http://arduino.esp8266.com/stable/package_esp8266com_index.json

Web: www.sunrobotics.co.in
Email: support@sunrobotics.co.in



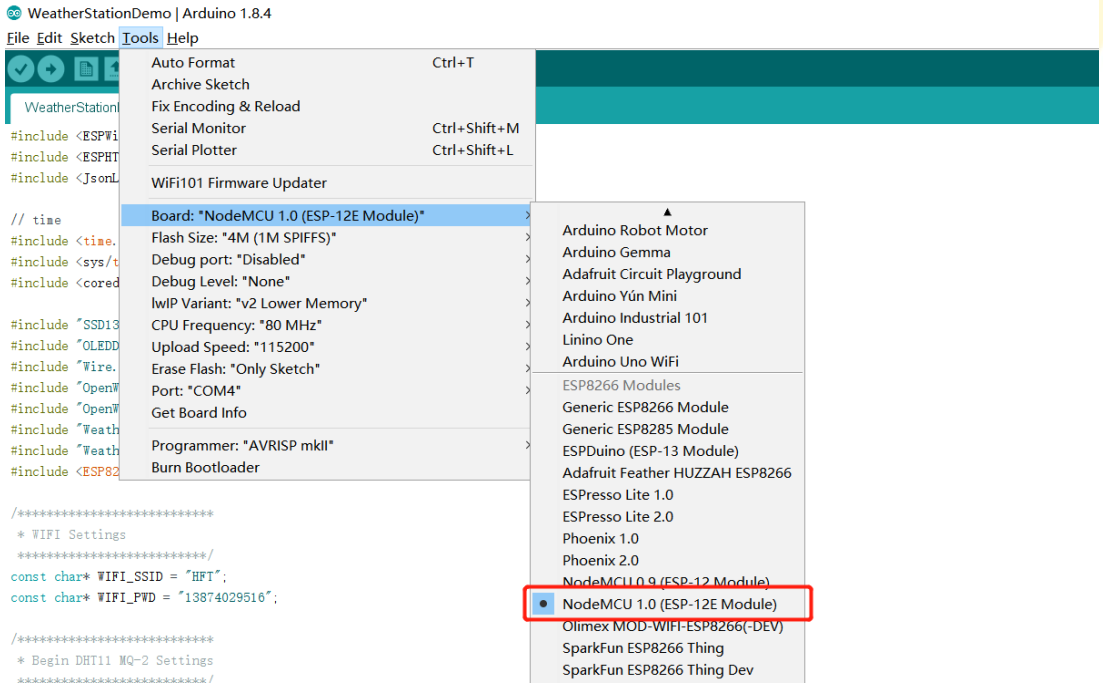
4) Click "OK" to close the dialog.
5) Go to Tools > Board > Board Manager and click this option
6) Type "ESP8266" in the text box, search, and display an option "ESP8266 by ESP8266 Community"



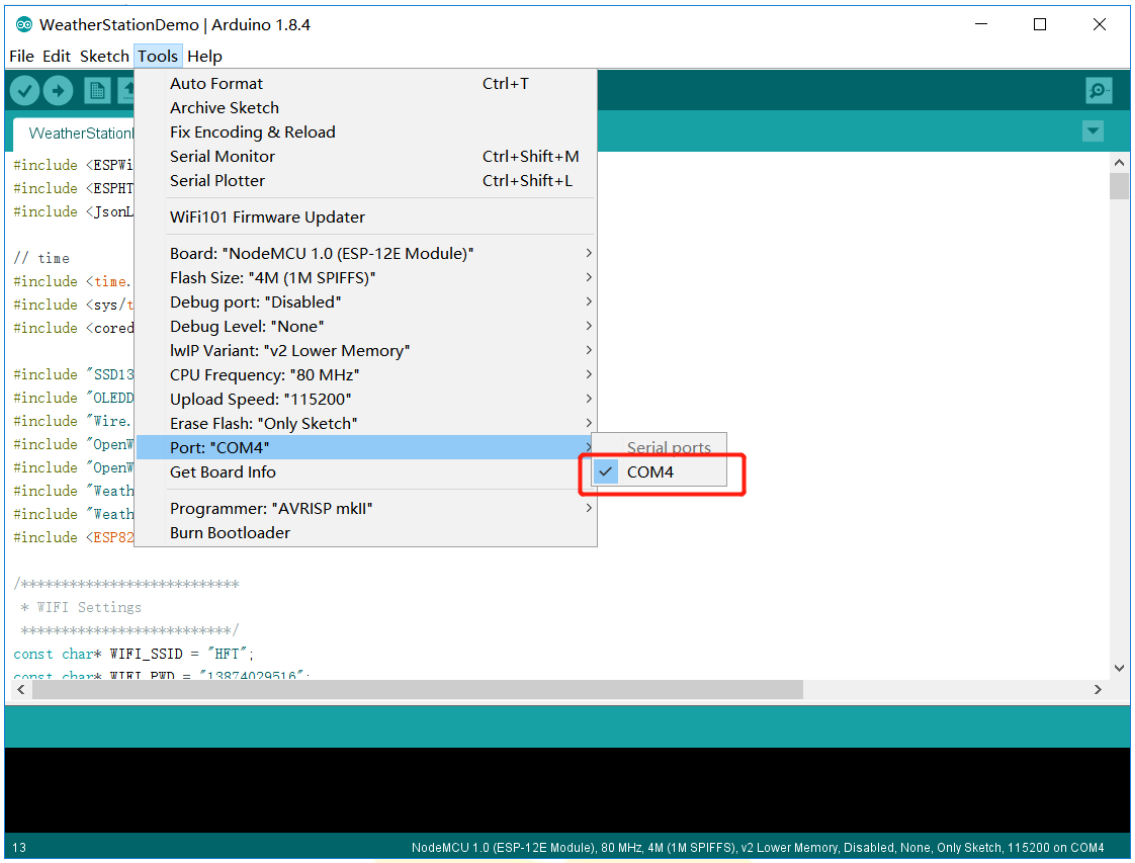
7) Click on "more info" and then click on the "Install" button

8) After the installation process is complete (may take 1 minute or 2 minutes), you can close the dialog by pressing the "Close" button

9) Go back to Tools > Boards and you should list some "new" boards at the bottom of the list. Select "NodeMCU 1.0 (ESP-12E Module)"



10) Choose the correct port:

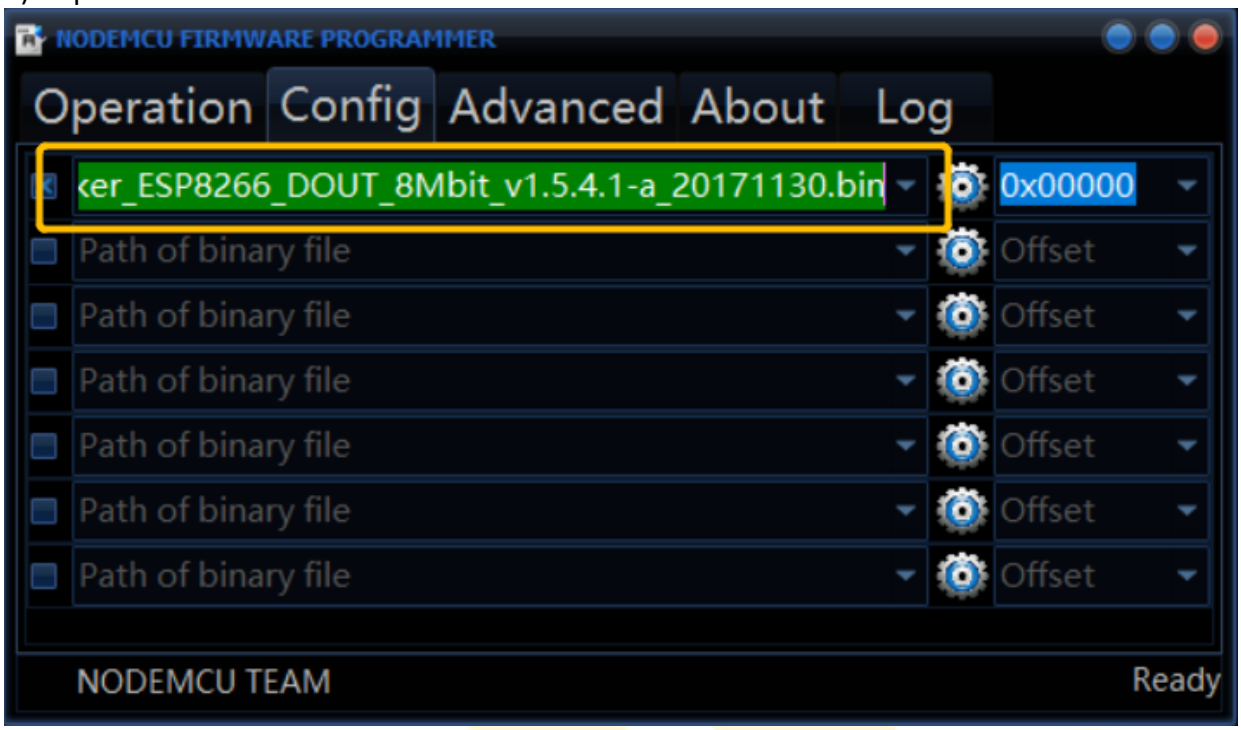


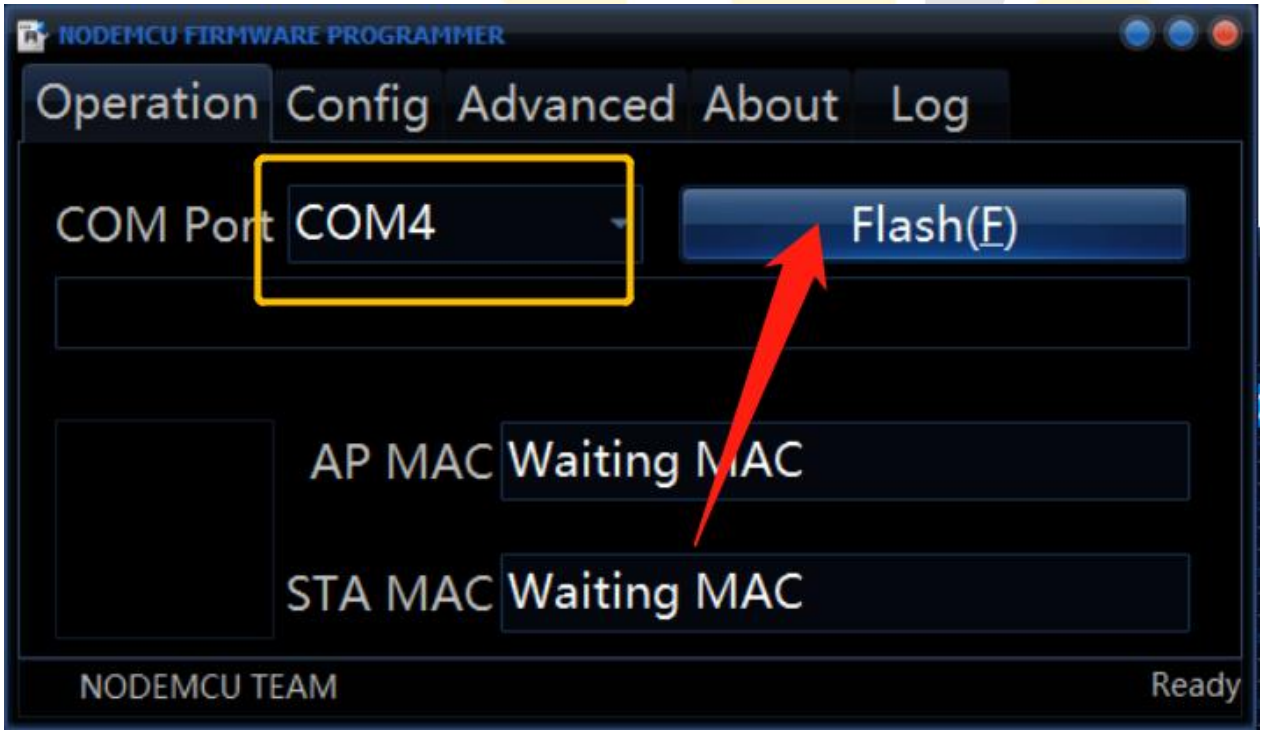
Step 3: Burning Firmware to ESP8266

1) The computer is connected to the ESP8266 via the data cable:

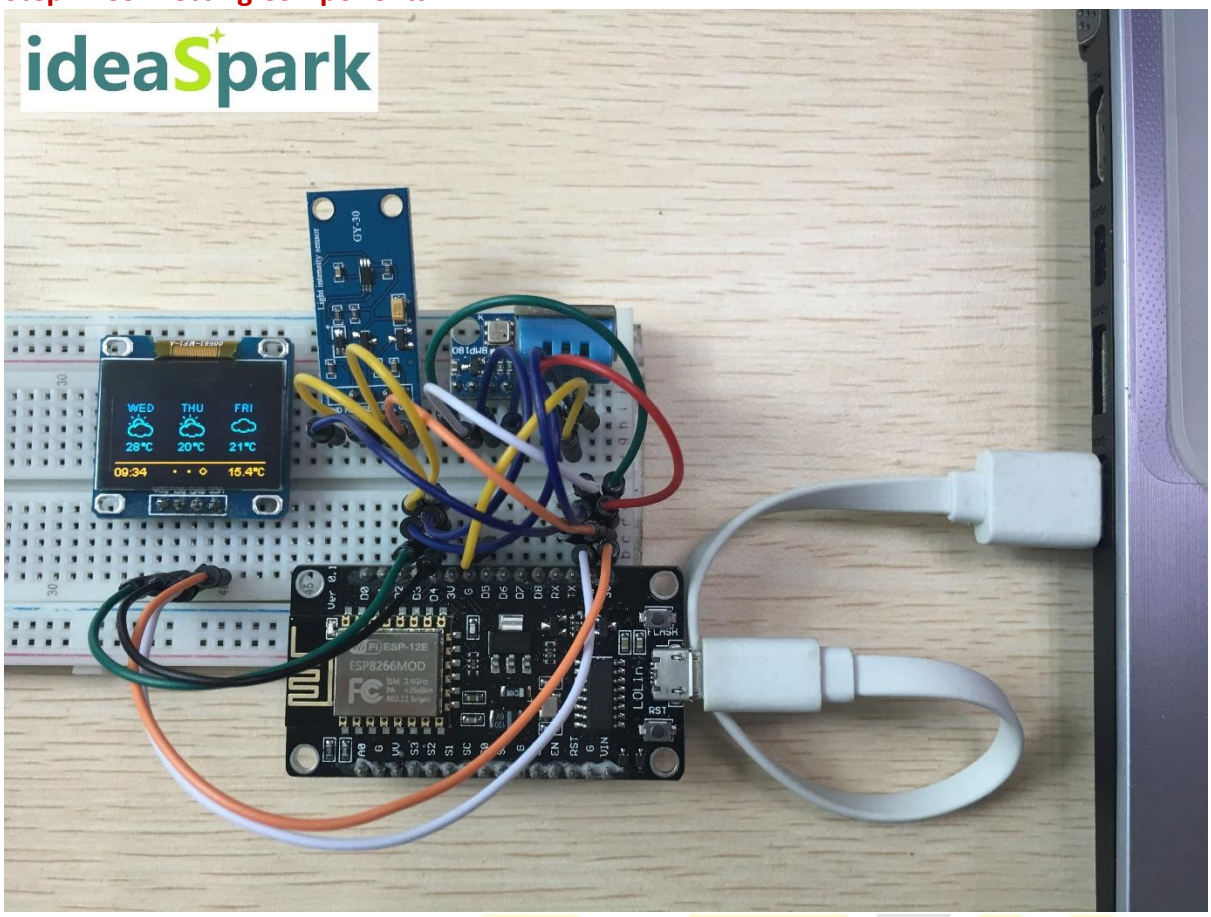
2) Use ESP8266Flasher.exe Burning Ai-Thinker_ESP8266_DOUT_8Mbit_v1.5.4.1-a_20171130.bin to ESP8266 ESP8266Flasher.exe Ai-Thinker_ESP8266_DOUT_8Mbit_v1.5.4.1-a_20171130.bin

3) Operation as shown：





Step 4: Connecting Components



1) Pins Map

ESP8266-12E	OLED
3.3V	VCC
GND	GND
D3	SDA
D4	SCL

ESP8266-12E	DHT11
3.3V	VCC
GND	GND
D5	DATA

ESP8266-12E	BH1750FVI
3.3V	VCC

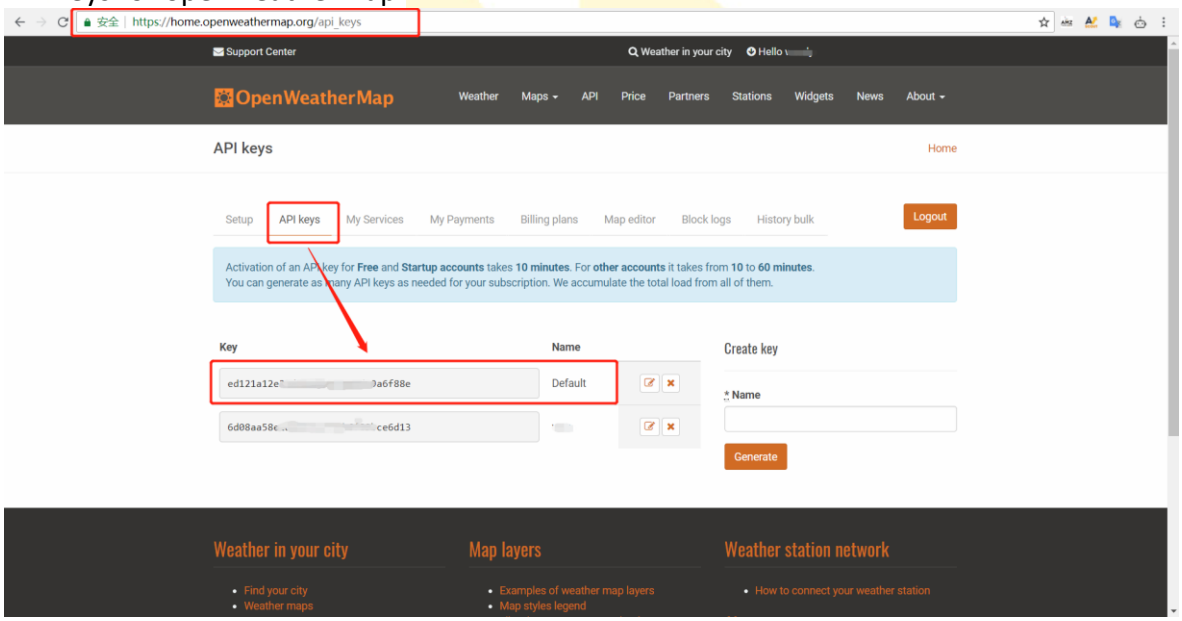
GND	GND
D3	SDA
D4	SCL

ESP8266-12E	BMP180
3.3V	VCC
GND	GND
D3	SDA
D4	SCL

ESP8266-12E	Computer
Micro-USB	USB

Step 5: Register OpenWeathermap, thingspeak new account

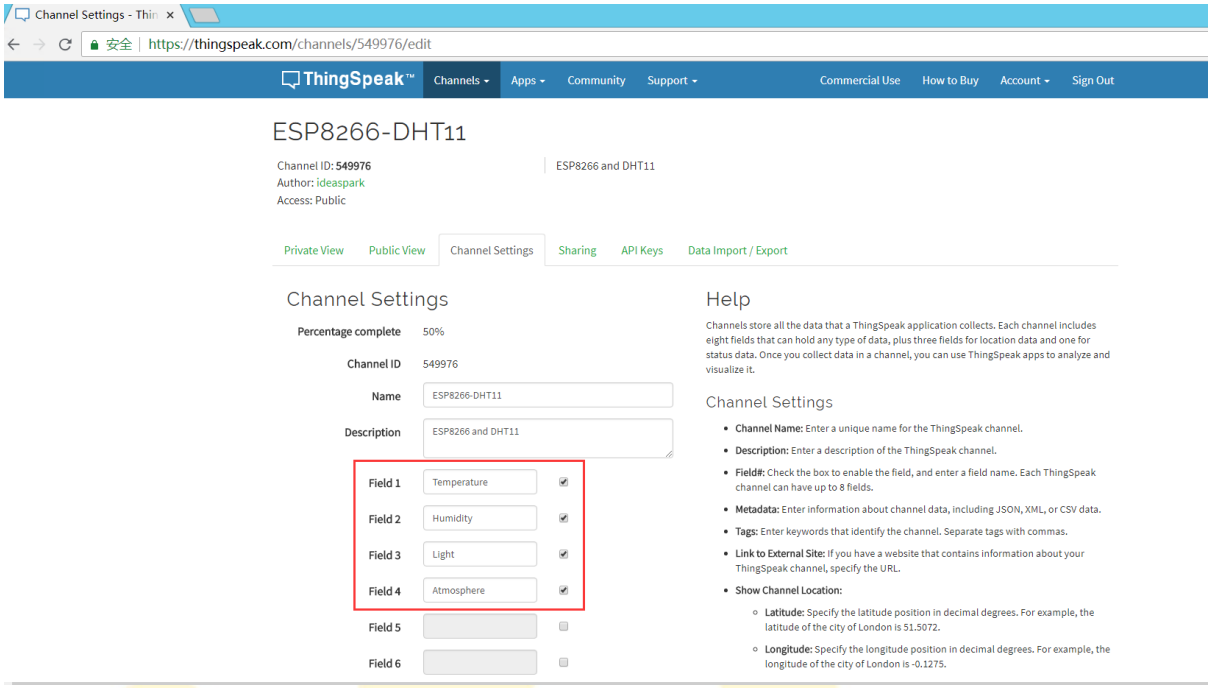
1) Sign up new account at https://home.openweathermap.org/api_keys to get the API Keys for openweathermap:



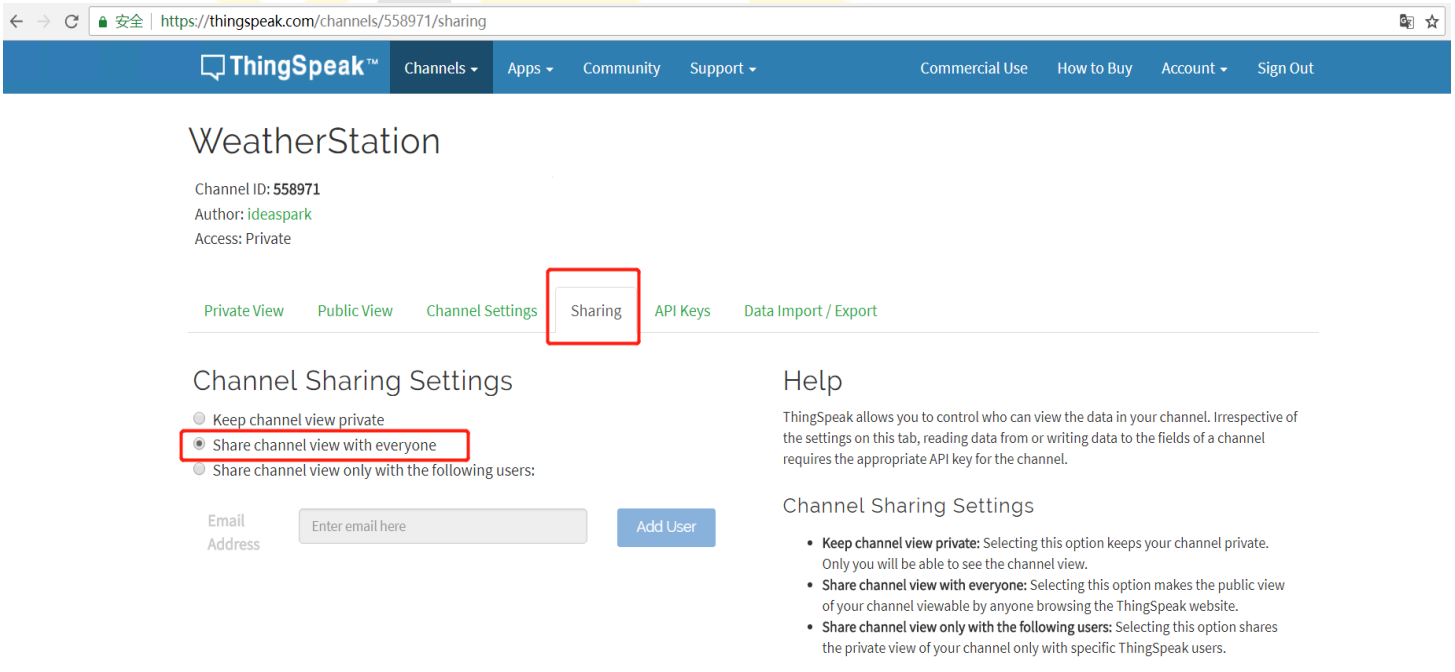
1) Sign up for a new account at https://thingspeak.com

A) Create a new Channel and receive the temperature, humidity, and light atmosphere data from ESP8266-12E (WeatherStation)

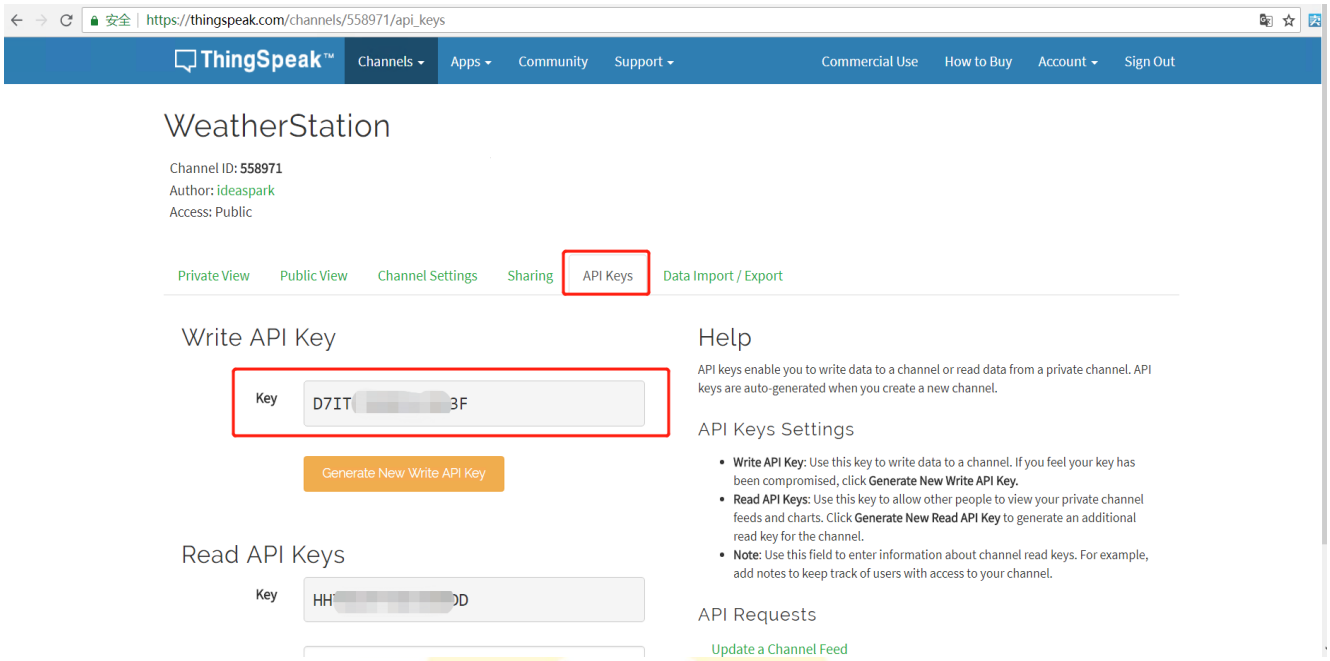
Web: www.sunrobotics.co.in
Email: support@sunrobotics.co.in



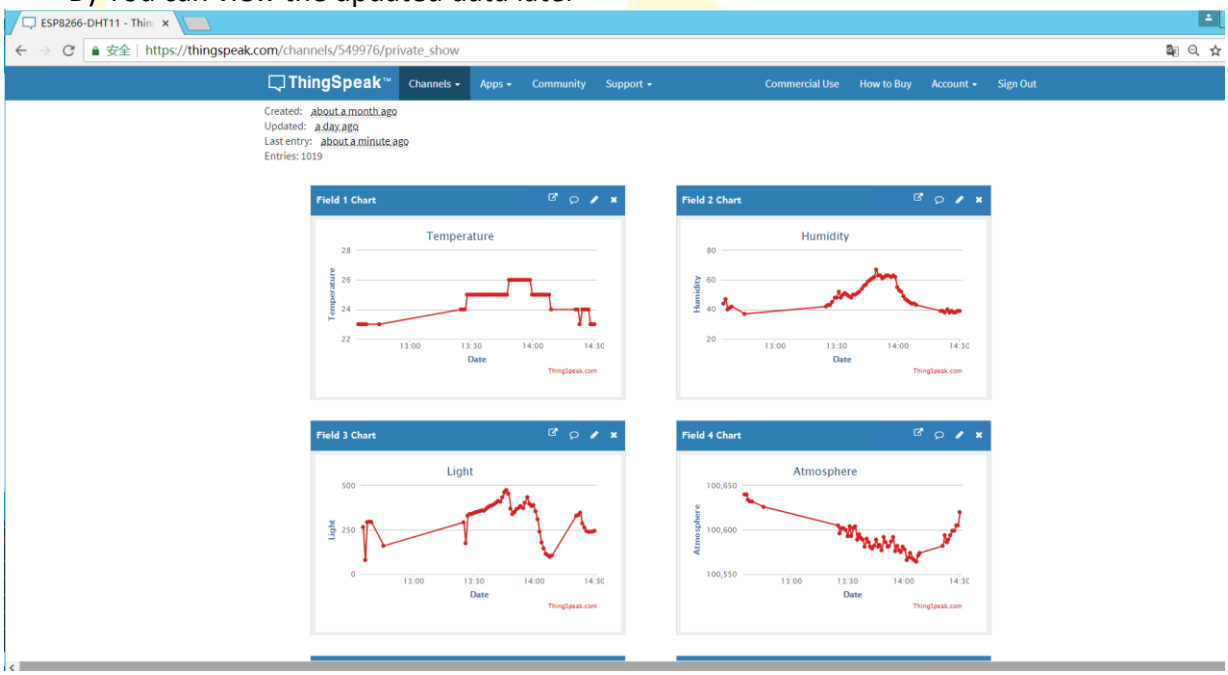
B) Set Channel to Public



C) Get the API_keys of the channel



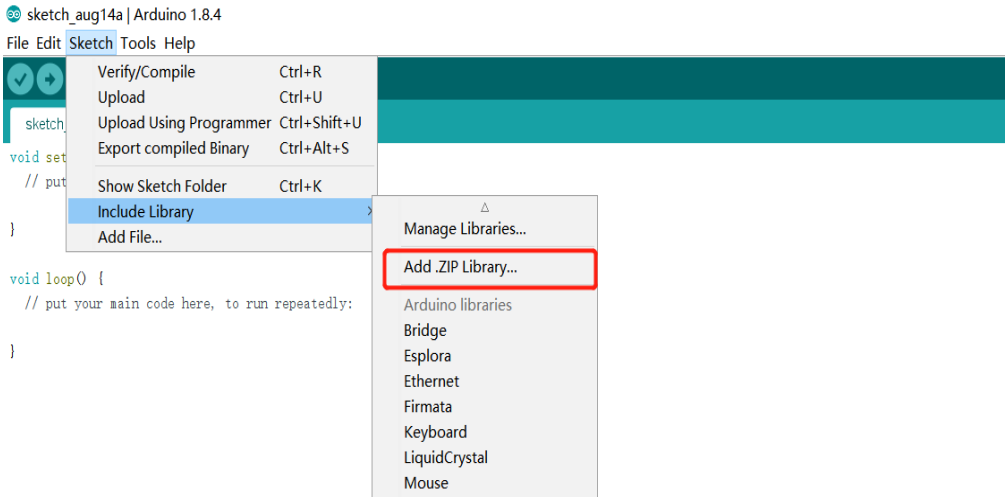
D) You can view the updated data later

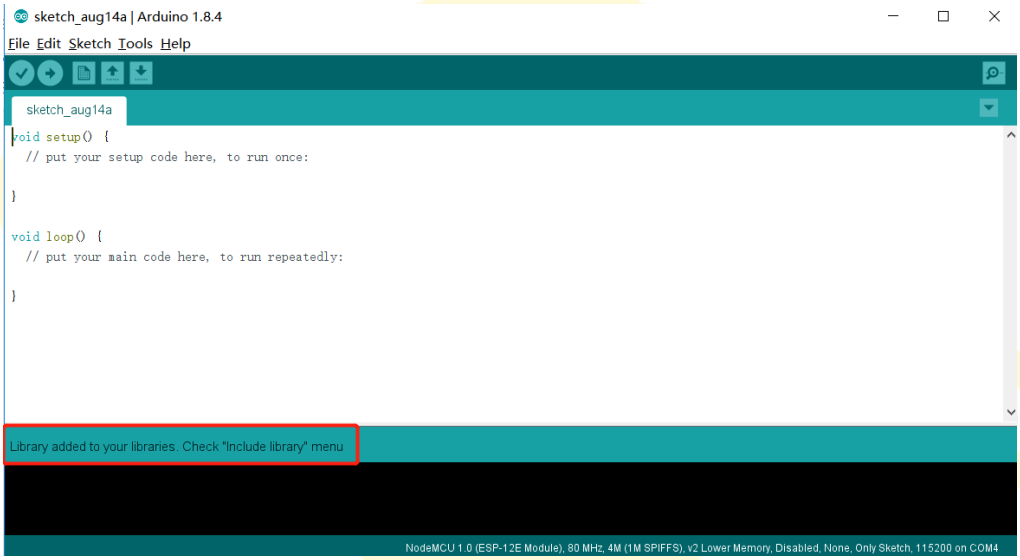


Step 6: Import WeatherStation code in the Arduino IDE

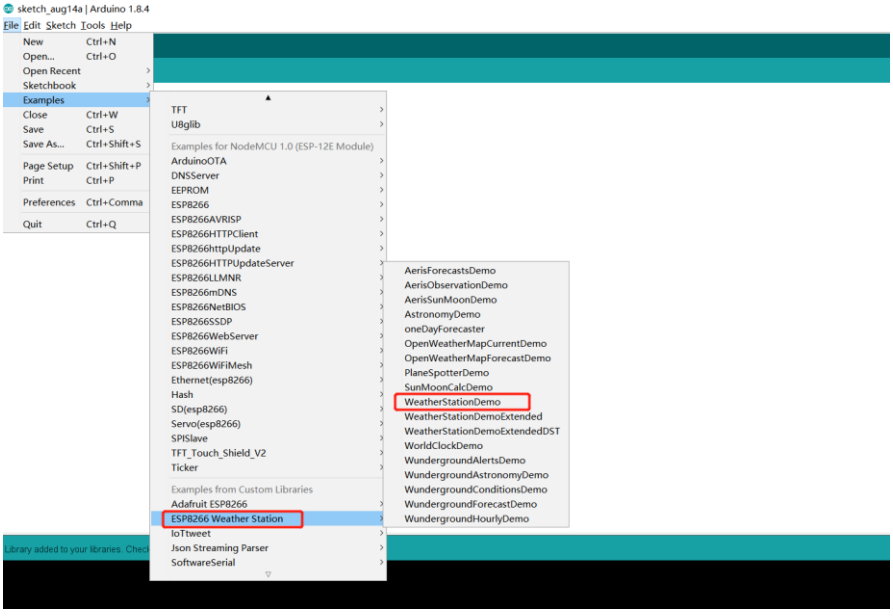
1) Import \Weather Station\Arduino IDE for ESP8266 Code\esp8266-weather-station-master.zip

esp8266-weath
er-station-maste

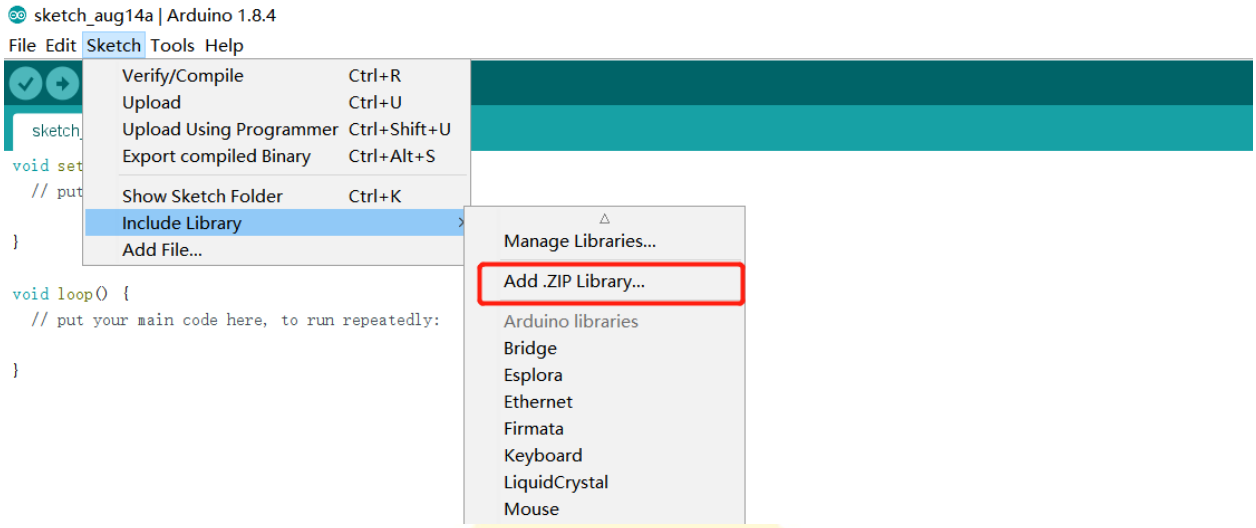




1） View Demo code：



Step 7: Add Library



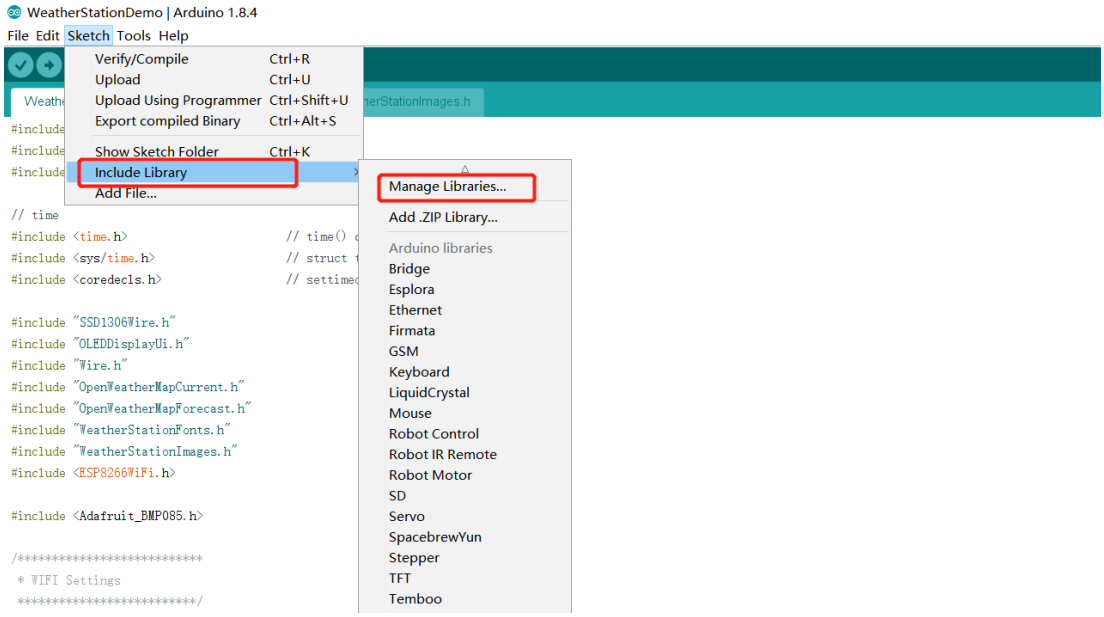
1) Add \Weather Station\Arduino IDE for ESP8266 Library\json-streaming-parser-master.zip

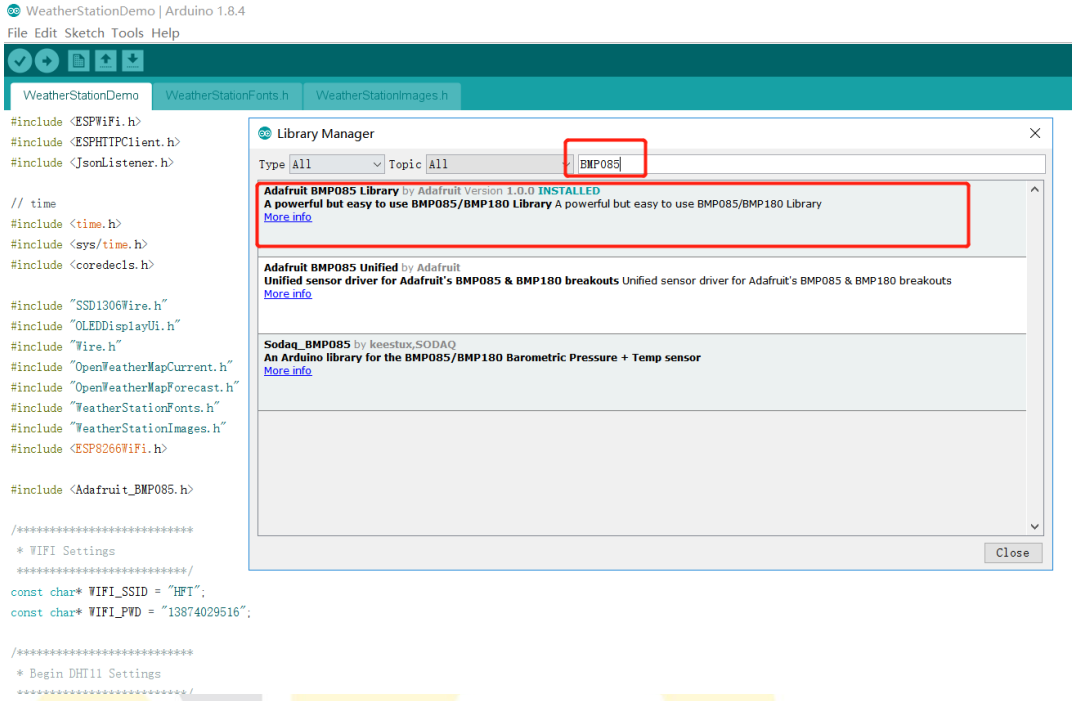
json-streaming-parser-master.zip

2) Add \Weather Station\Arduino IDE for ESP8266 Library\esp8266-oled-ssd1306-master.zip

esp8266-oled-ssd1306-master.z

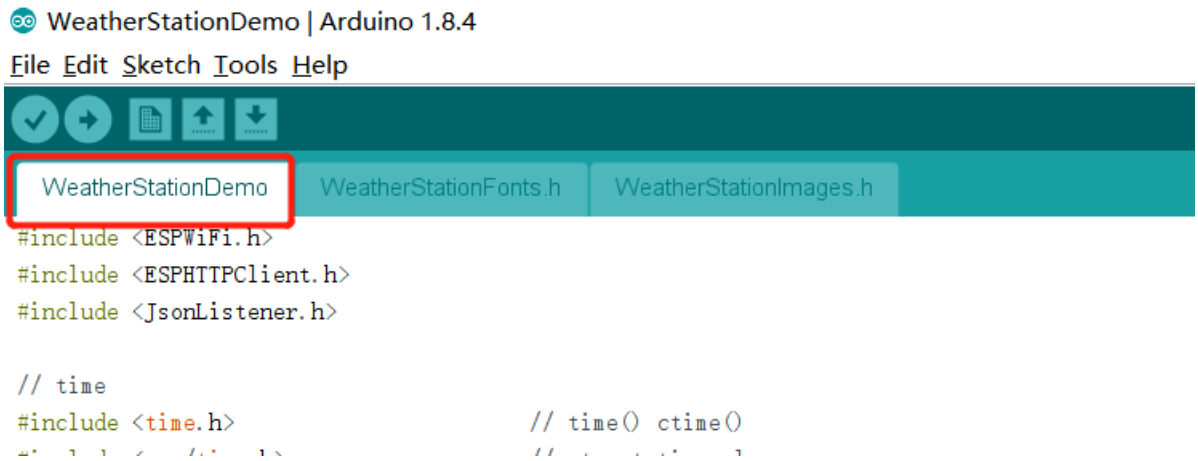
3) Add Library Adafruit_BMP085



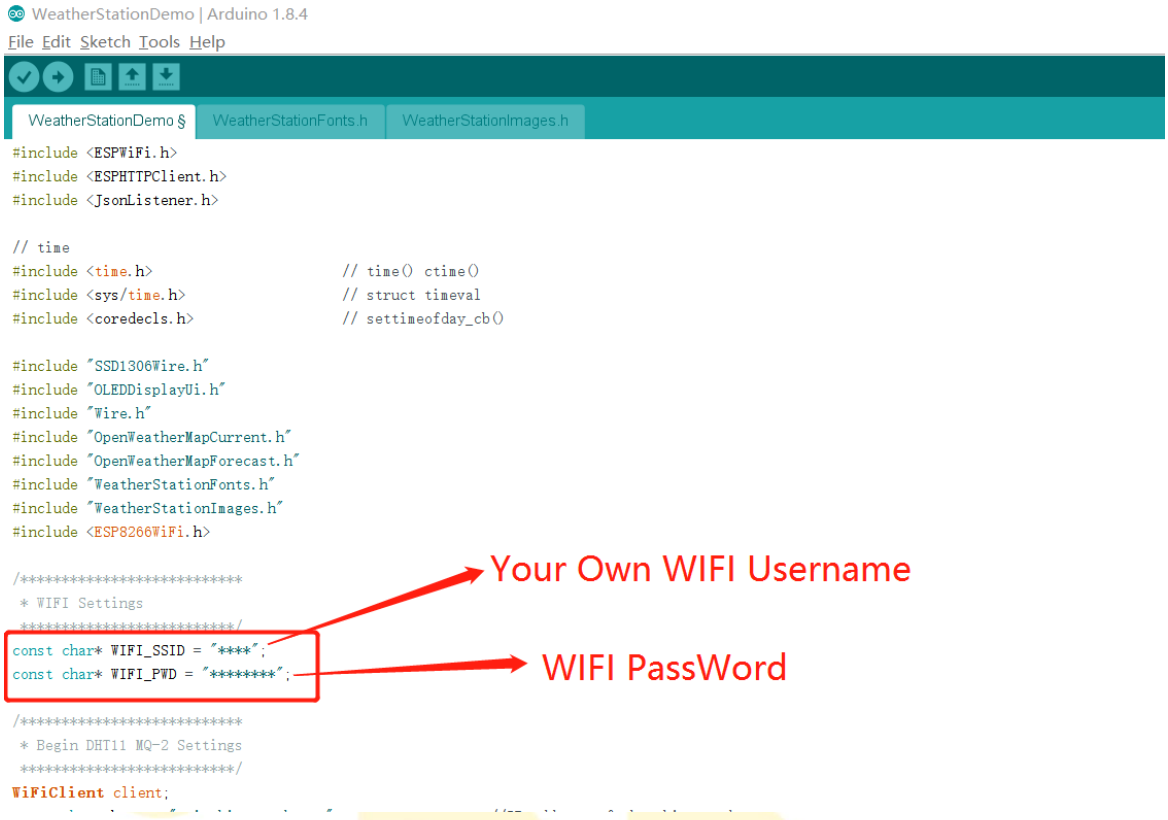


Step 8: Modify File WeatherStation.ino

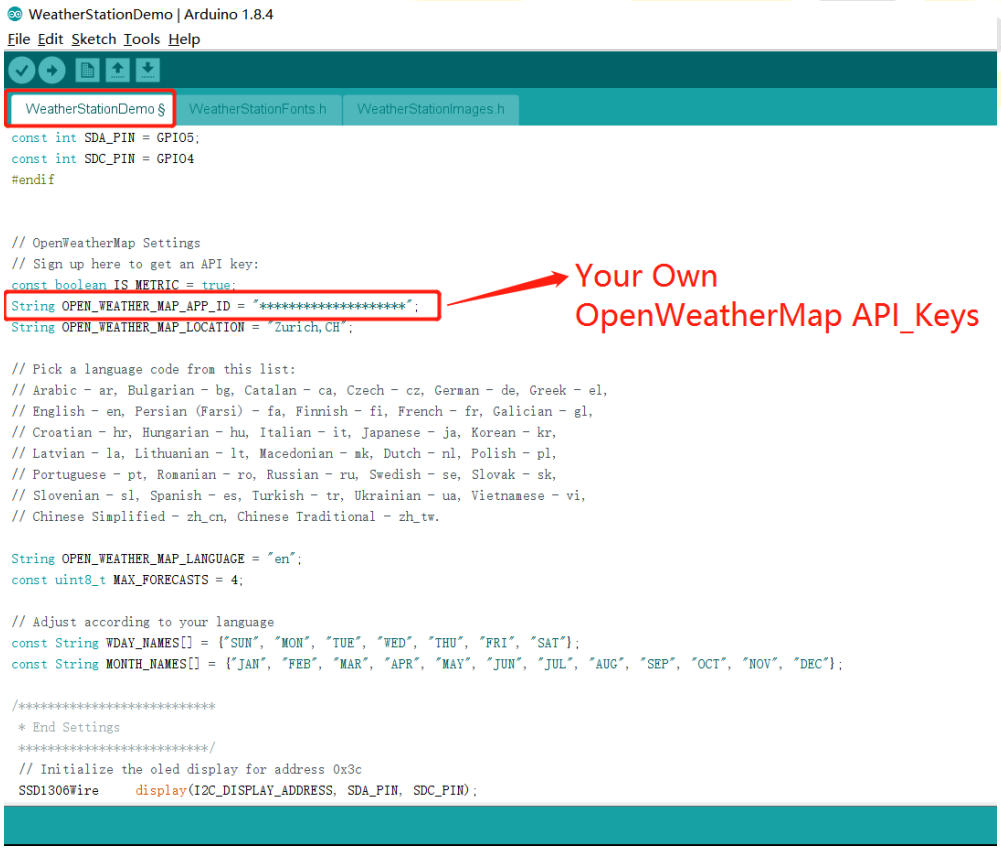
1）Open WeatherStation.ino



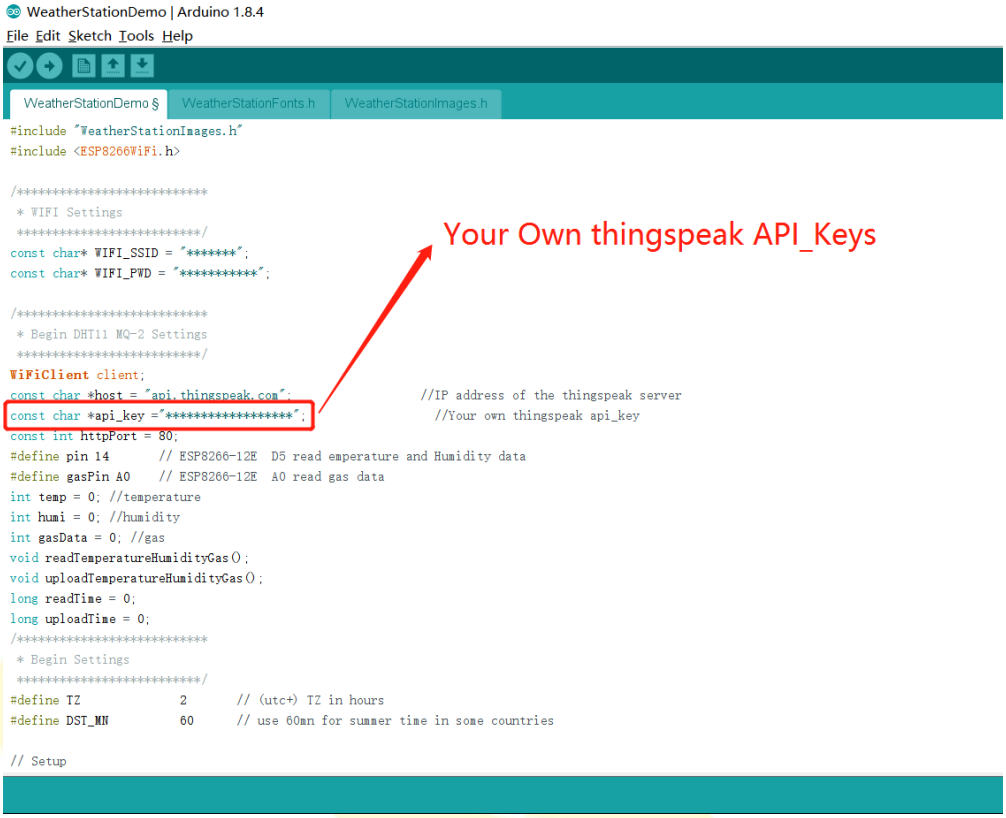
2）Update your WIFI username and password:



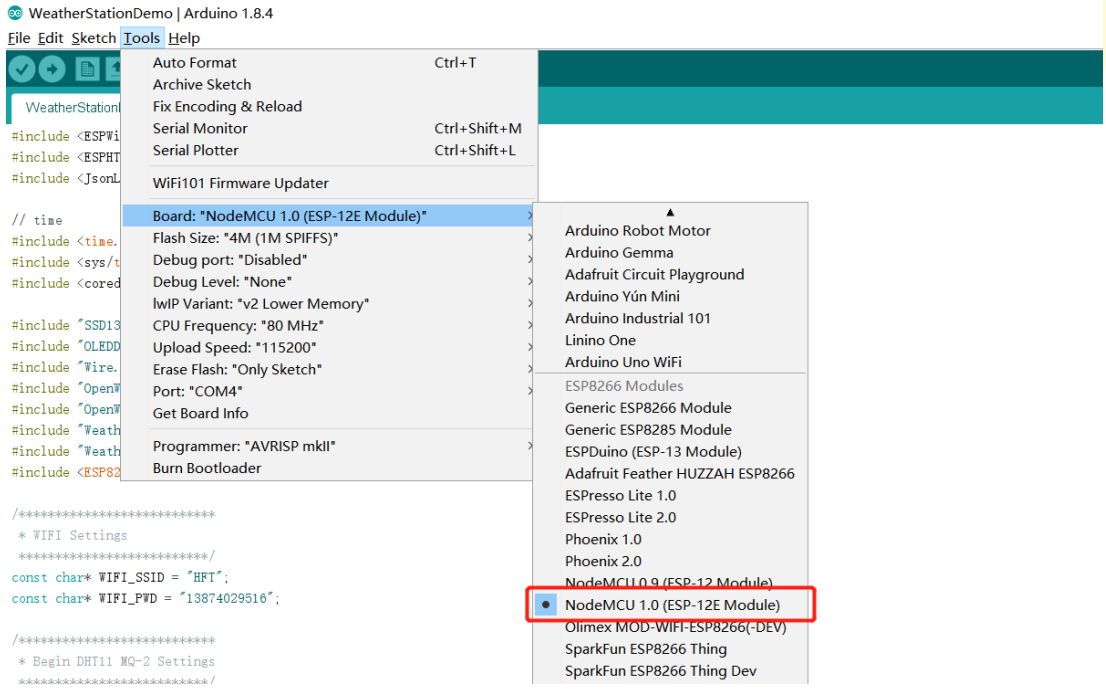
3）Update API_Keys of OpenWeatherMap.org

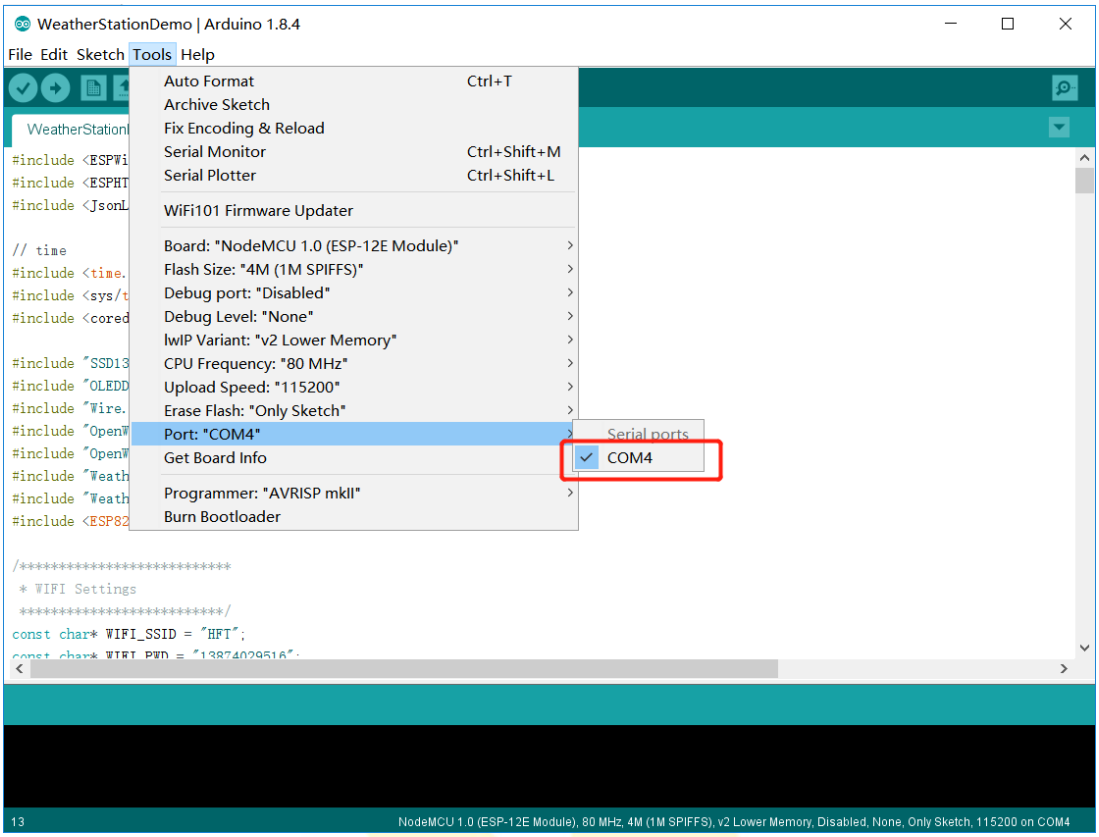


4）Update the API_Keys of thingspeak.com



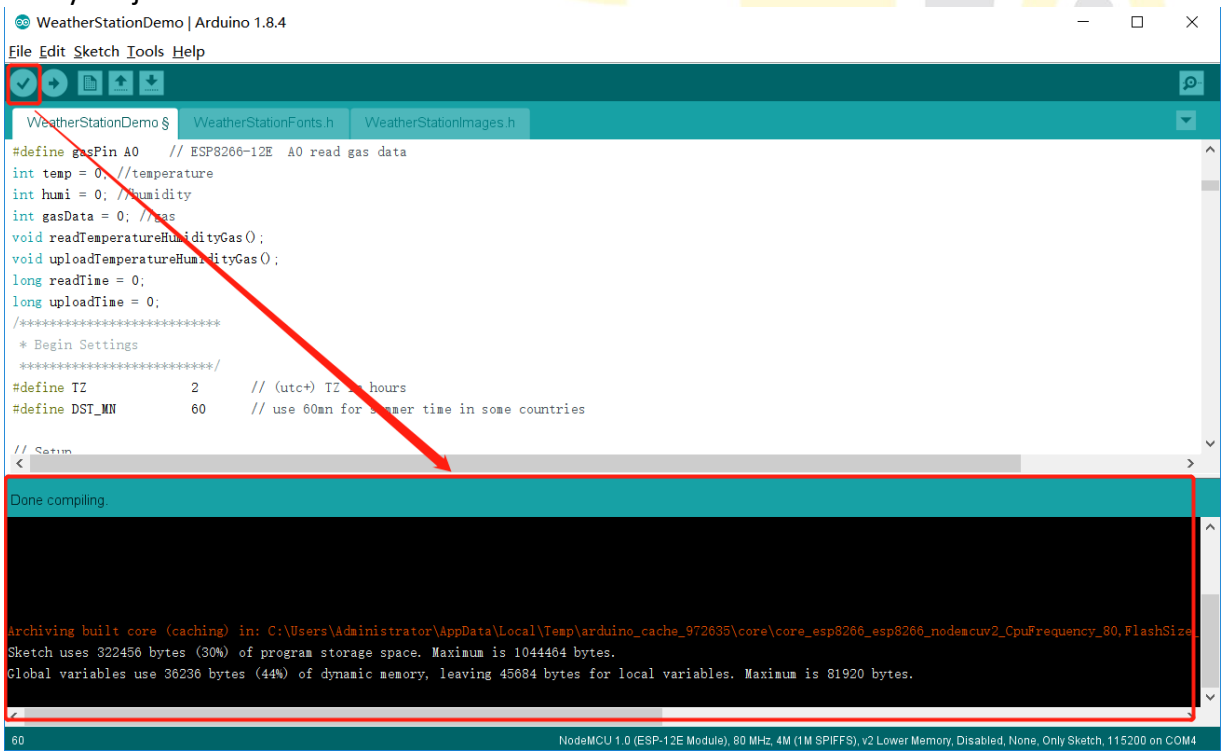
Step 9: Set the Board and Port again



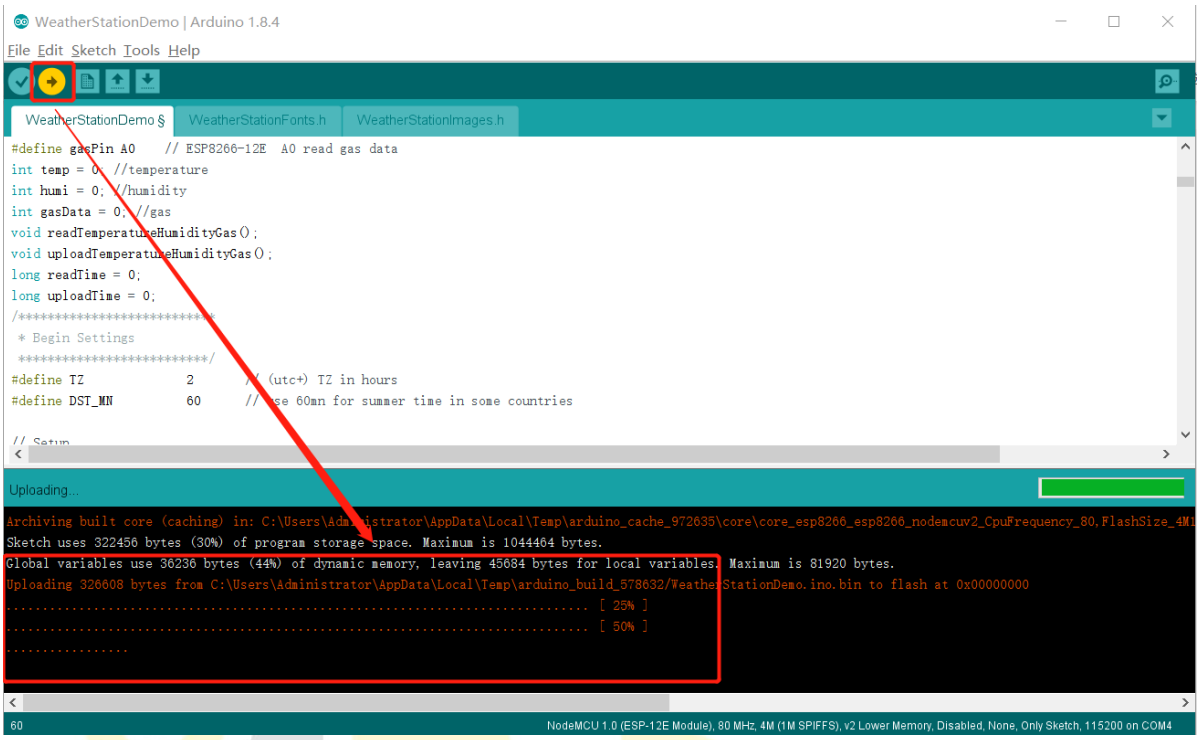


Step 10: Burn code to ESP8266

Verify Project

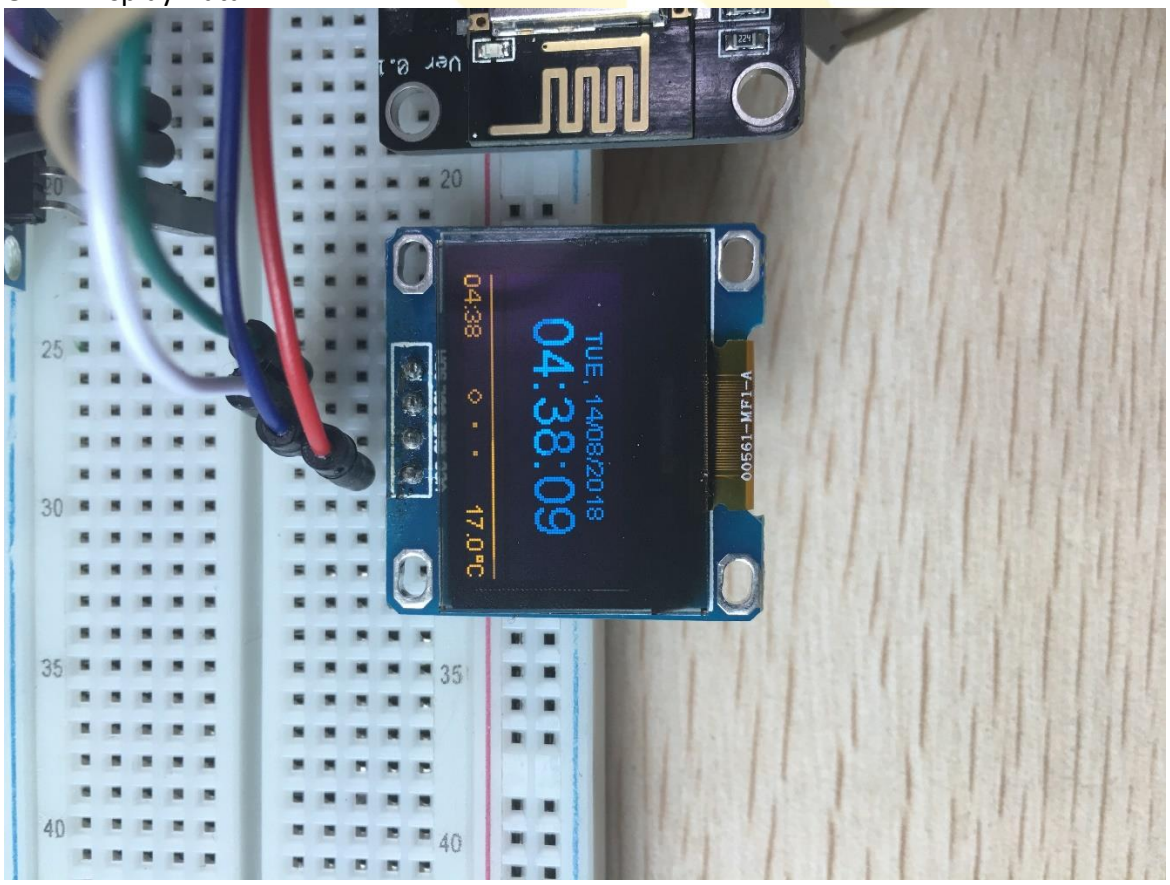


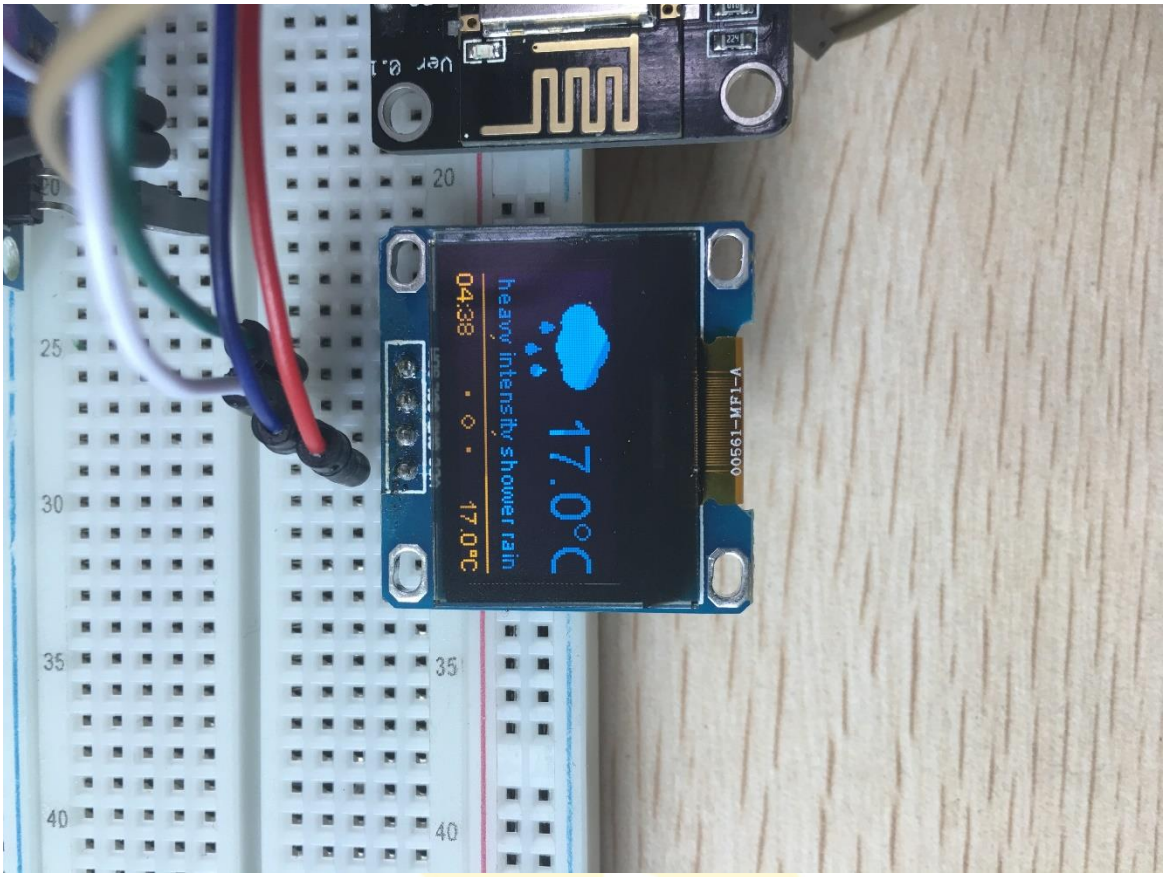
1) Upload Project

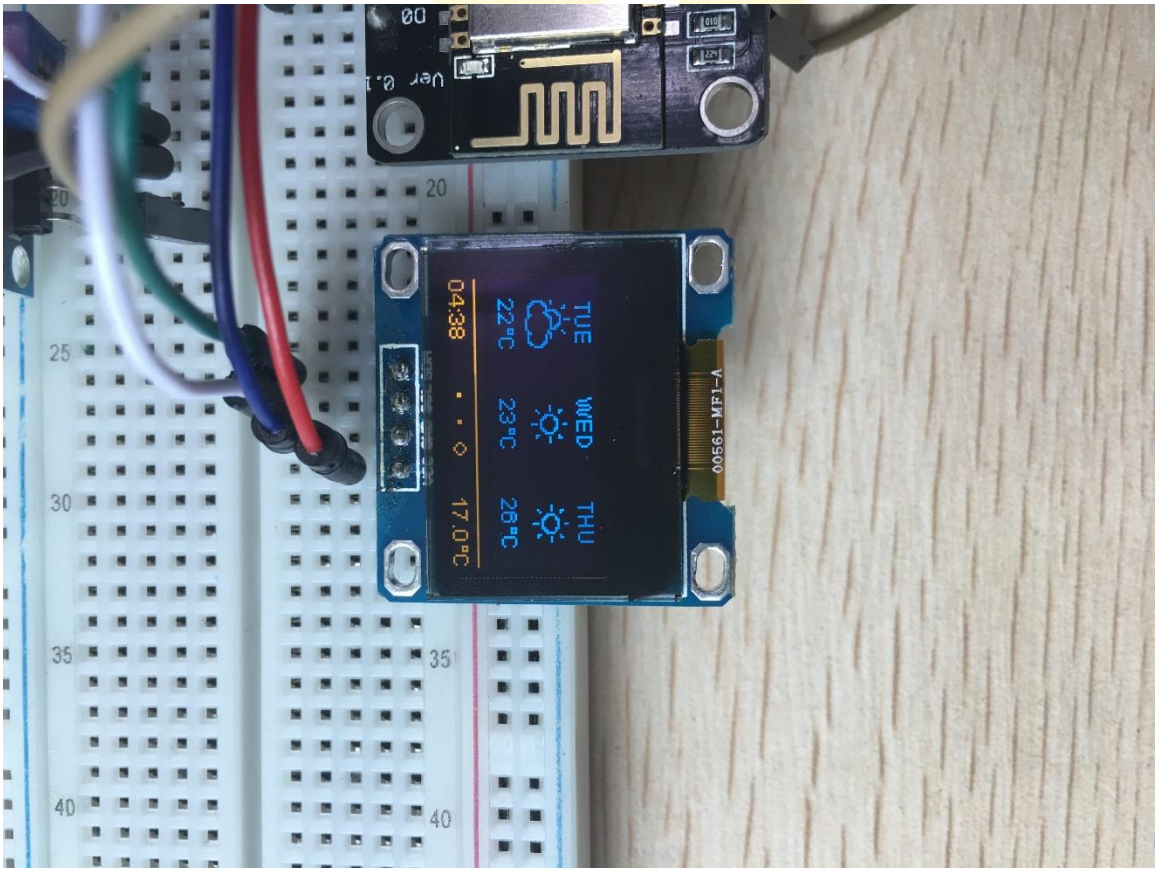


Step 11: Result

OLED Display Data

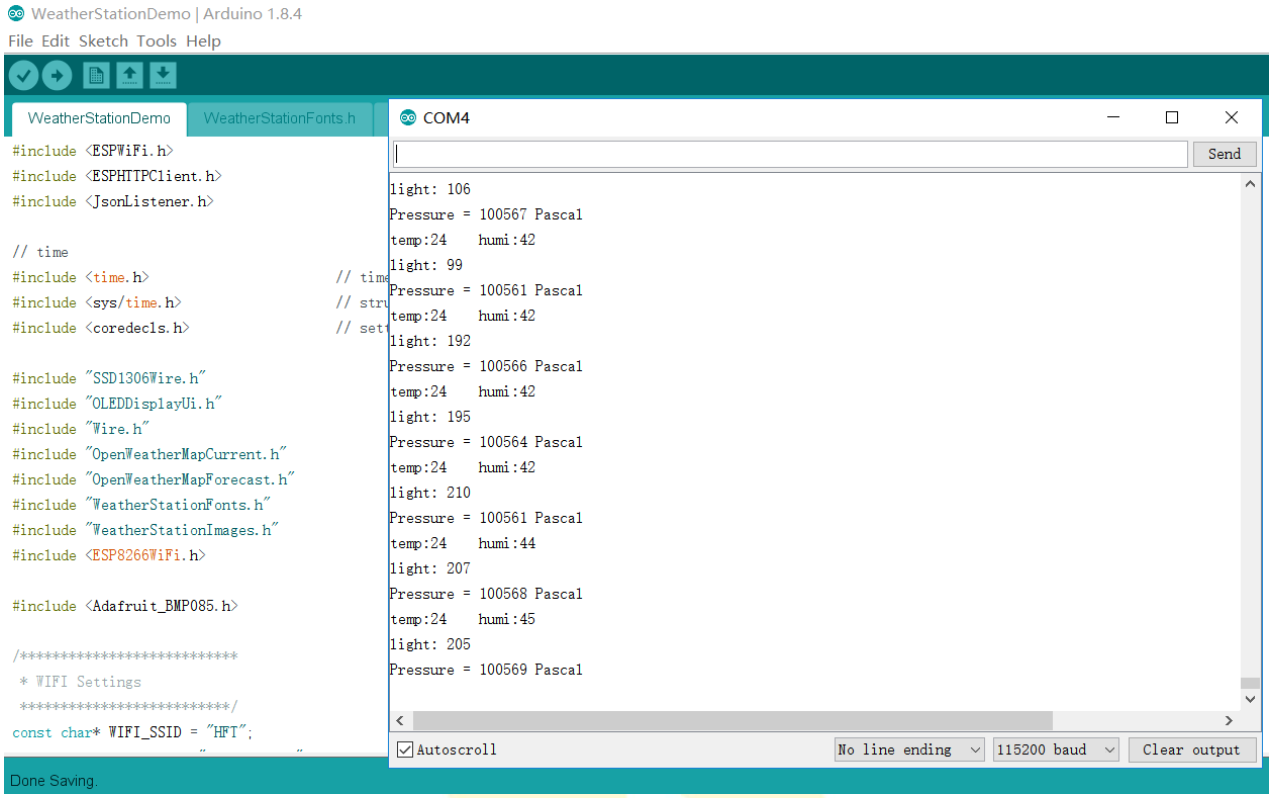




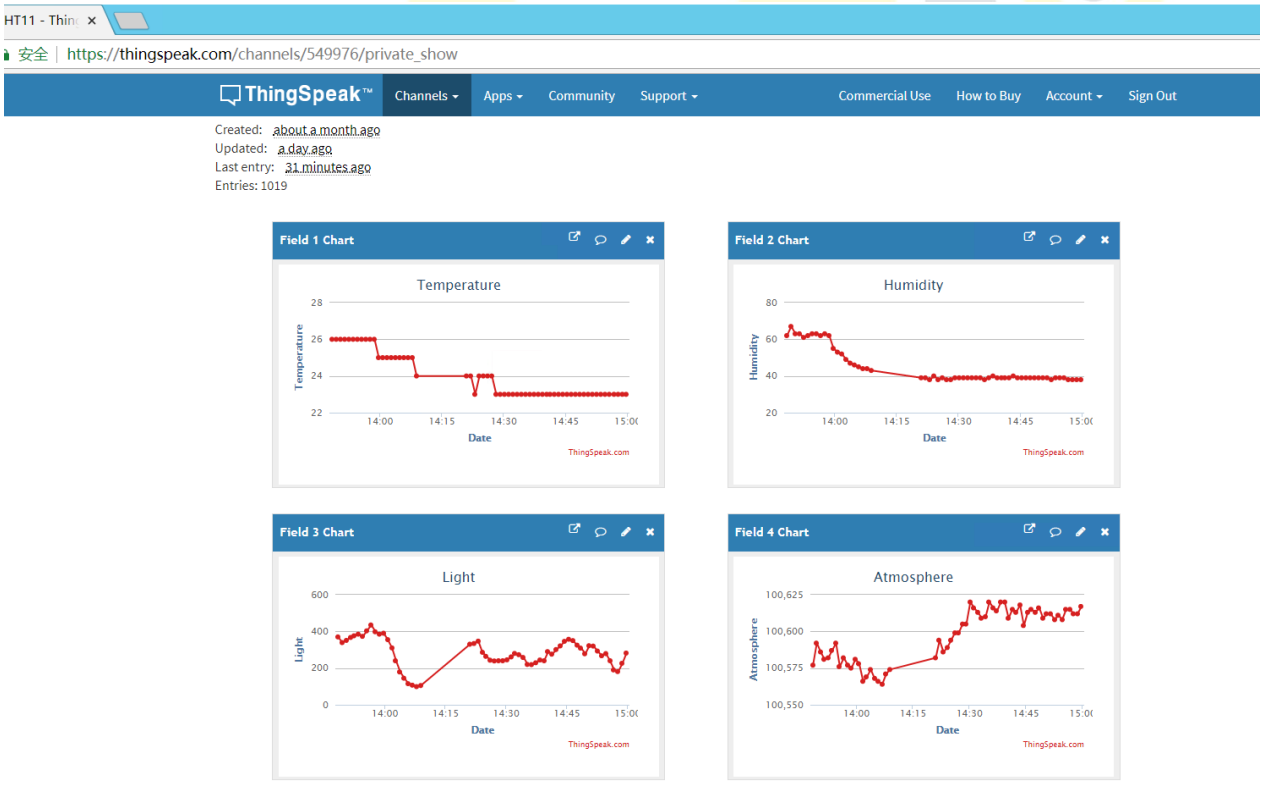


Web: www.sunrobotics.co.in
Email: support@sunrobotics.co.in

1) Serial Monitor Data



2) thinkspeak.com Data



Web: www.sunrobotics.co.in
Email: support@sunrobotics.co.in

Thank You

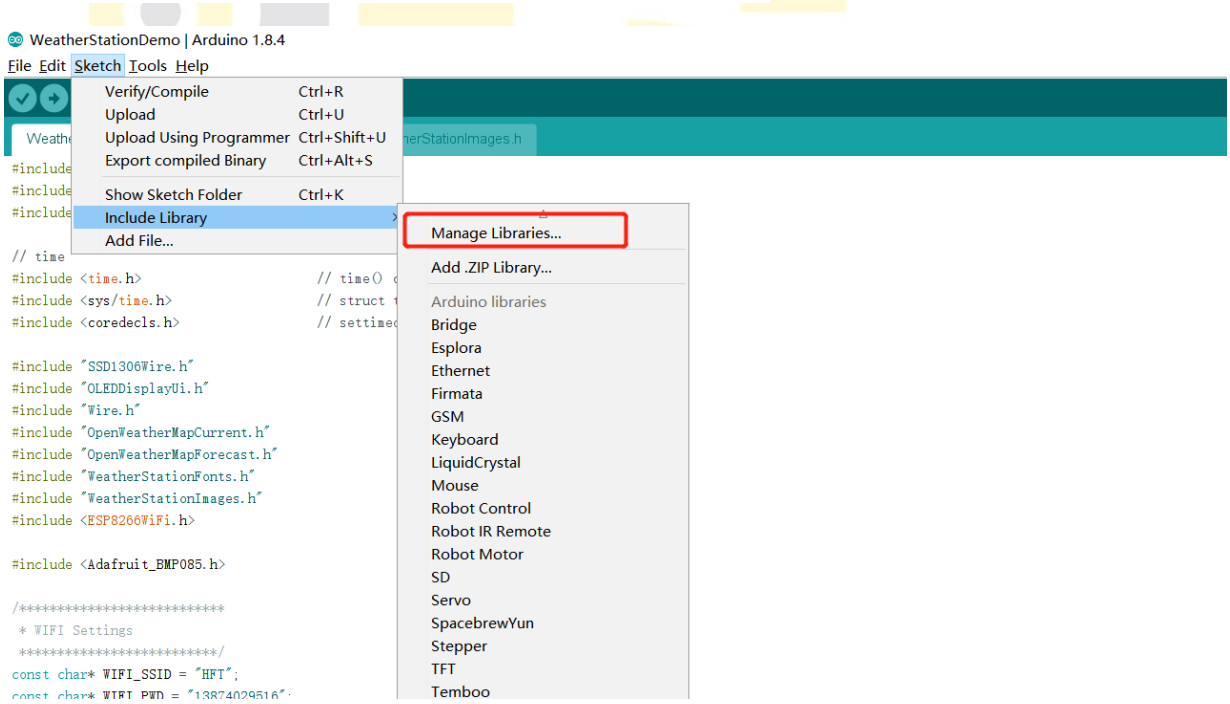
If the following error occurs in the Arduino IDE:

```
C:\Users\jb\Documents\Arduino\libraries\esp8266-weather-station-master\src\TimeClient.cpp: In member function 'long int TimeClient::getCurrentEpochWithUtcOffset()':
C:\Users\jb\Documents\Arduino\libraries\esp8266-weather-station-master\src\TimeClient.cpp:124:67: error: invalid operands of types 'double' and 'long int' to binary 'operator%'
  return round(getCurrentEpoch() + 3600 * myUtcOffset + 86400L) % 86400L;
.....
Using library Adafruit_BMP085_Library at version 1.0.0 in folder: C:\Users\jb\Documents\Arduino\libraries\Adafruit_BMP085_Library
exit status 1
Error compiling for board NodeMCU 1.0 (ESP-12E Module).
```

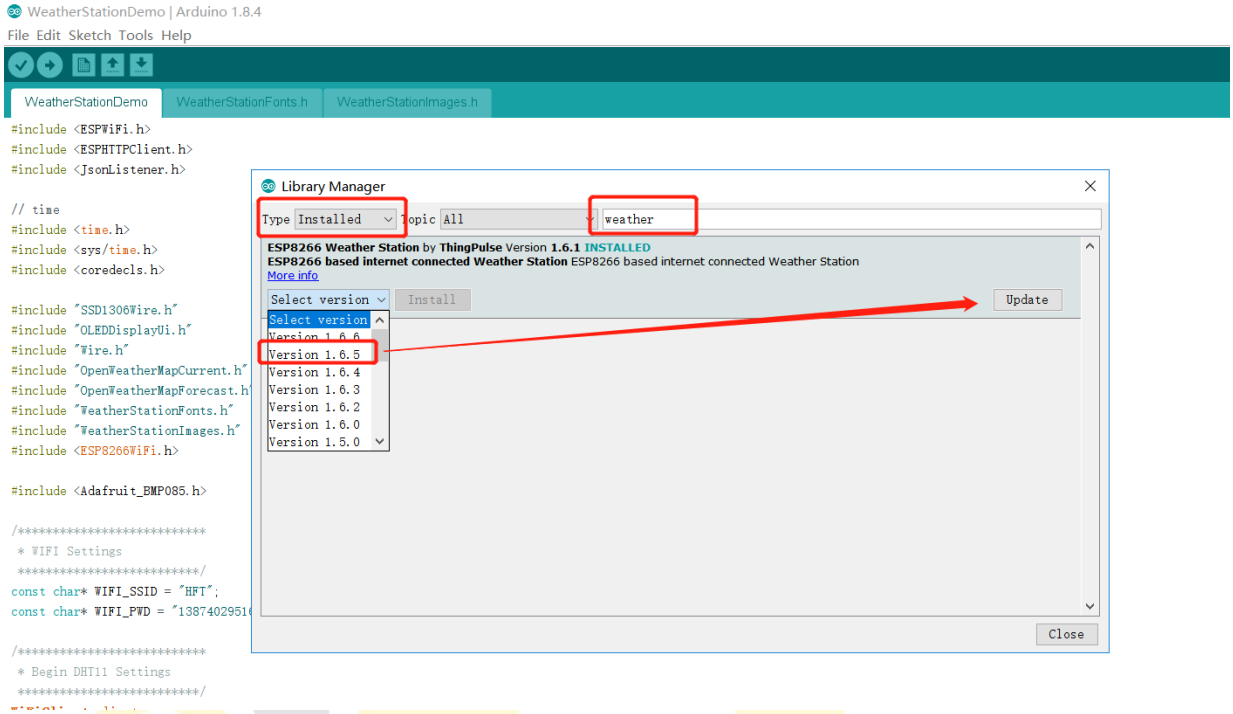
Step 1: Setup Recommended Arduino IDE version 1.8.4 for Windows:

https://www.amazon.com/clouddrive/share/5hYUsaMRYFatPBizWinTapxlSzqdW8LQtVyiihAYIS5

Step 2: Follow the instructions in the attached picture to update the "esp8266 weather station library" version to 1.6.5 or 1.6.6, this will resolve this error:



Web: www.sunrobotics.co.in
Email: support@sunrobotics.co.in



If you still have questions, please feel free to contact us

Web: www.sunrobotics.co.in
Email: support@sunrobotics.co.in

THANK YOU

SUN ROBOTICS

www.sunrobotics.co.in