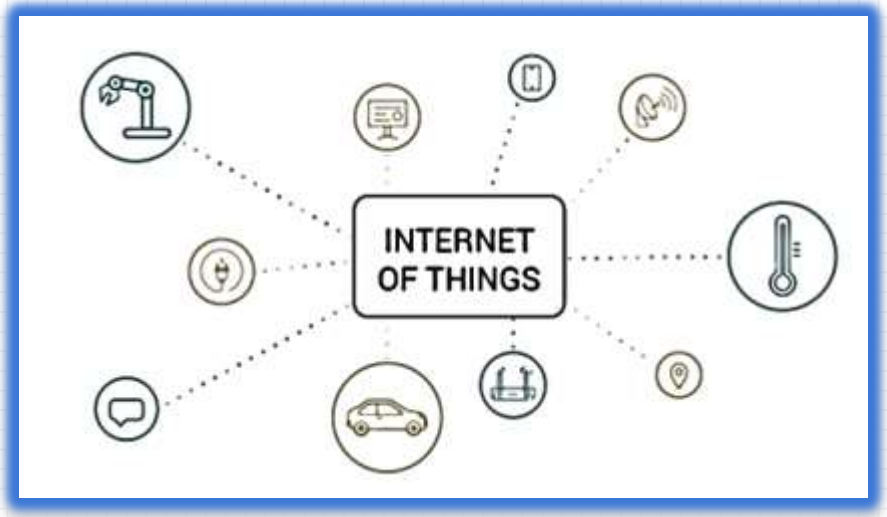
IOT or Internet of THings

What is IoT ?

The Internet of Things (IoT) describes the network of physical objects–"things"–that are embedded with sensors, software, and other technologies for the purpose of connecting and exchanging data with other devices and systems over the internet.

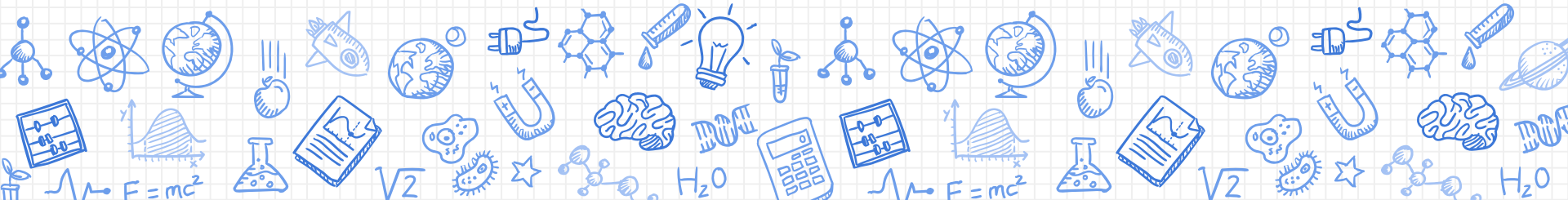


HELLO!

We are Shivam & Gaurav.

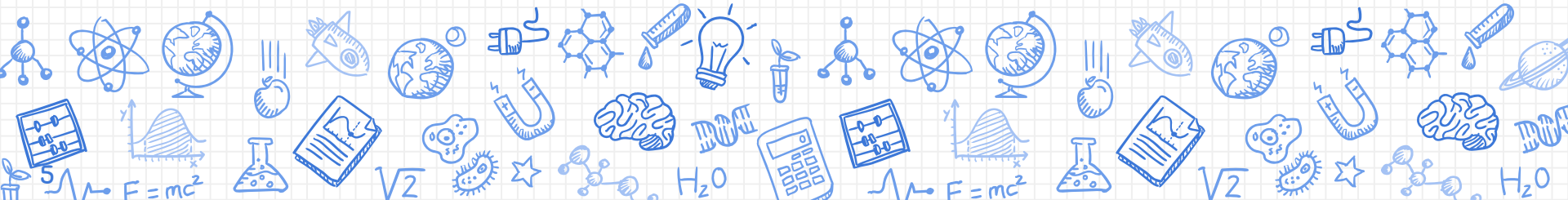
Why IoT ?

The internet of things helps people live and work smarter, as well as gain complete control over their lives. In addition to offering smart devices to automate homes, IoT is essential to business.

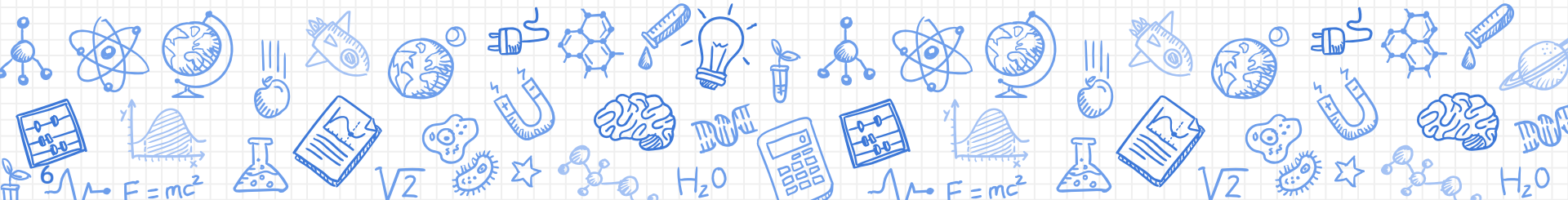


System Requirement

1. Microcontrollers
2. Sensors
3. Web & Mobile Applications



Microcontrollers



Types of Microcontrollers for IoT.

Node MCU

ESP-32

Node MCU

Node MCU

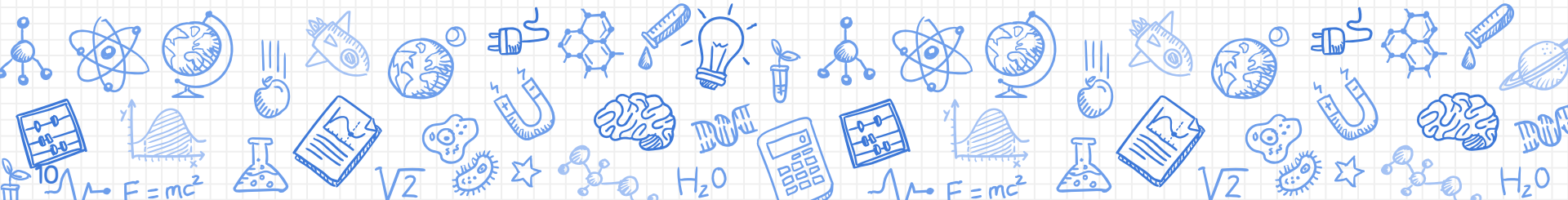
- Microcontroller: Tensilica 32-bit RISC CPU Xtensa LX106
- Operating Voltage: 3.3V
- Input Voltage: 7-12V
- Digital I/O Pins (DIO): 16
- Analog Input Pins (ADC): 1
- UARTs: 1
- SPIs: 1
- I2Cs: 1
- Flash Memory: 4 MB
- SRAM: 64 KB
- Clock Speed: 80 MHz
- USB-TTL based on CP2102 is included onboard, Enabling Plug n Play
- PCB Antenna
- Small Sized module to fit smartly inside your IoT projects

Types of Microcontrollers for IoT.

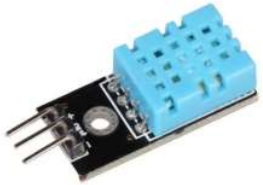
ESP-32

- Microcontroller: Tensilica Xtensa Dual-Core 32-bit LX6 with 600 DMIPS
- Operating Voltage: 3.3V
- Input Voltage: 7-12V
- Digital I/O Pins (DIO): 39 (of which 34 is normal GPIO pin)
- Analog Input Pins (ADC): 12-bit, 18 Channel
- UARTs: 3
- SPIs: 3
- I2Cs: 2
- Flash Memory: 4 MB
- SRAM: 520 KB
- Clock Speed: 160 MHz
- USB-TTL based on CP2102 is included onboard, Enabling Plug n Play
- PCB Antenna
- Small Sized module to fit smartly inside your IoT projects

Sensors



Types of Sensors for IoT.



DHT-11

BMP-280

ECG Sensor

Temperature Sensor

Types of Sensors for IoT.

DHT-11

1. Operating Voltage: 3.5V to 5.5V
2. Operating current: 0.3mA (measuring) 60uA (standby)
3. Output: Serial data
4. Temperature Range: 0°C to 50°C
5. Humidity Range: 20% to 90%
6. Resolution: Temperature and Humidity both are 16-bit
7. Accuracy: ±1°C and ±1%

Types of Sensors for IoT.

BMP-280

- Model: GY-BMP280-3.3
- Chip: BMP280
- Power supply: 3V/3.3V DC
- Peak current: 1.12mA
- Air pressure range : 300-1100hPa (equi. to +9000...-500m above sea level)
- Temperature range: -40 ... +85 °C
- Digital interfaces: I²C (up to 3.4 MHz) and SPI (3 and 4 wire, up to 10 MHz)
- Current consumption of sensor BMP280: 2.7µA @ 1 Hz sampling rate

Types of Sensors for IoT.

ECG Sensor

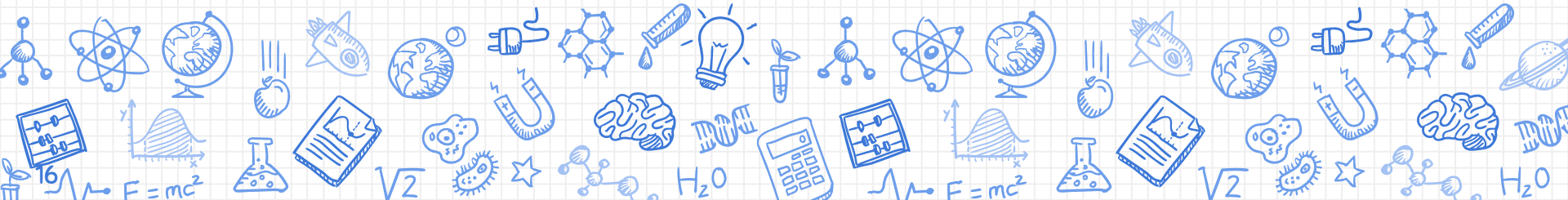
- Fully integrated single-lead ECG front end
- Common-mode rejection ratio: 80 dB (dc to 60 Hz)
- Two or three-electrode configurations
- Qualified for automotive application
- Single-supply operation: 2.0 V to 3.5
- Fast restore feature improves filter settling
- Size: 3.5cm x 3cm

Types of Sensors for IoT.

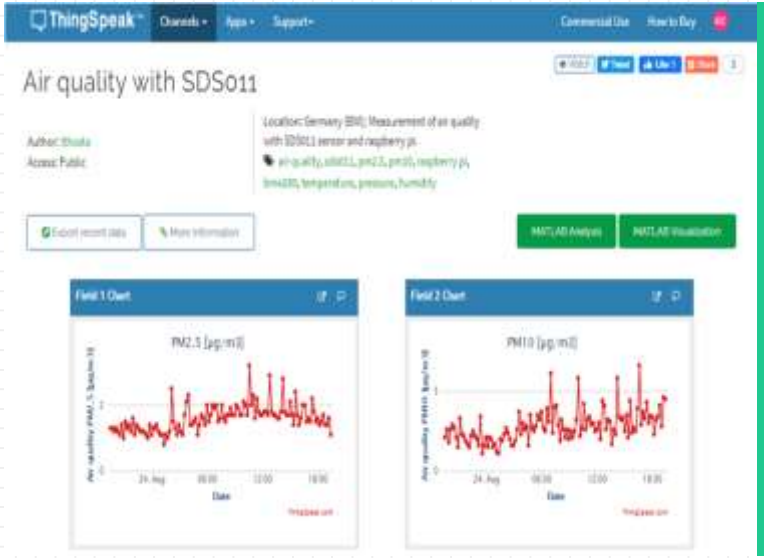
Temperature Sensor

- Programmable Digital Temperature Sensor
- Communicates using 1-Wire method
- Operating voltage: 3V to 5V
- Temperature Range: -55°C to +125°C
- Accuracy: ±0.5°C
- Output Resolution: 9-bit to 12-bit (programmable)
- Unique 64-bit address enables multiplexing
- Conversion time: 750ms at 12-bit
- Programmable alarm options
- Available as To-92, SOP and even as a waterproof sensor

Different Web and Mobile application.

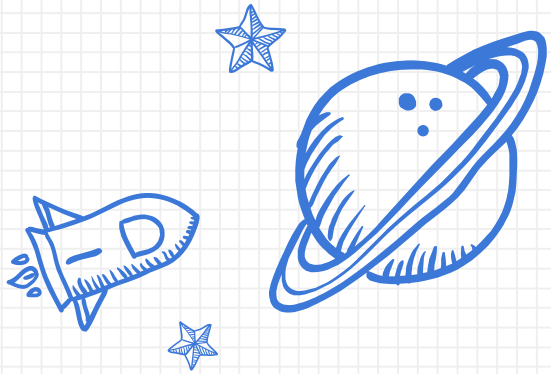


Different Web and Mobile application.



ThingSpeak

Blynk



So let's Start then

First Install the Arduino IDE.

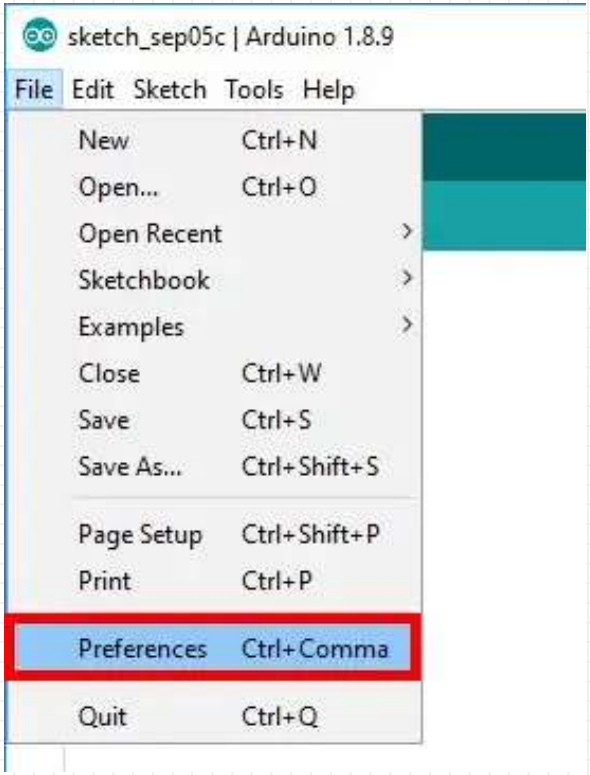
Step 1 – Open this website www.arduino.cc.

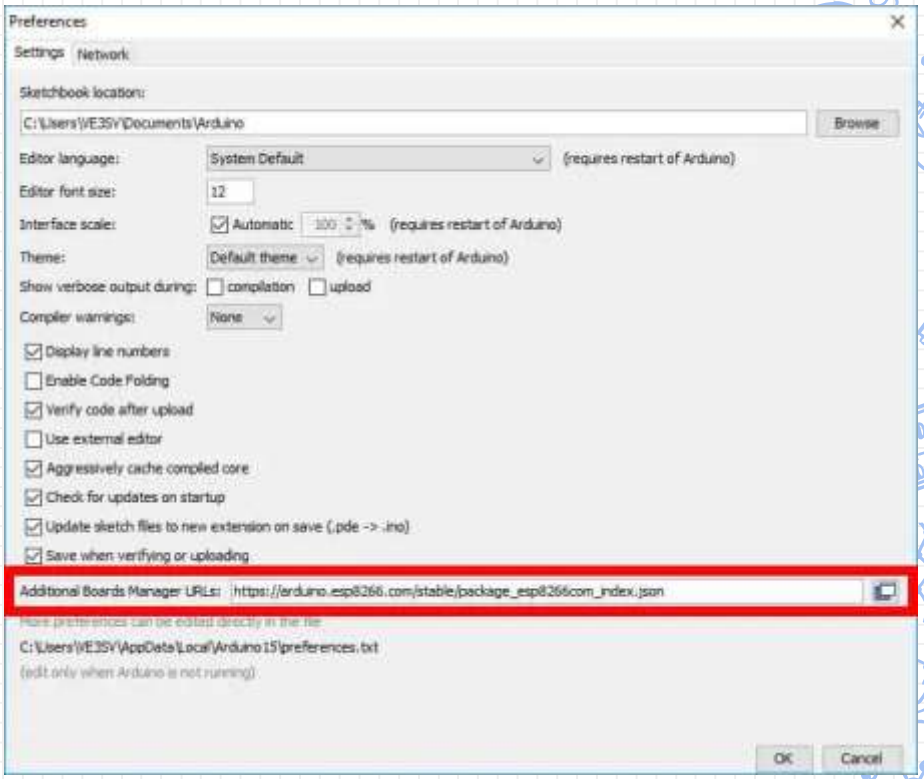
Step 2– Go to software → Downloads → Arduino IDE 1.8.19
→ Windows win 7 and newer.

Setup a programming environment for Node MCU in Arduino IDE.

1. Select the ***File →Preferences***
2. This will open up a new window, where we set the IDE to point to the configuration file.
3. In the field "**Additional Boards Manager URLs:**" enter
4. **http://arduino.esp8266.com/stable/package_esp8266com_index.json**
5. Select the **OK** button at the bottom of the window when the link address is entered.

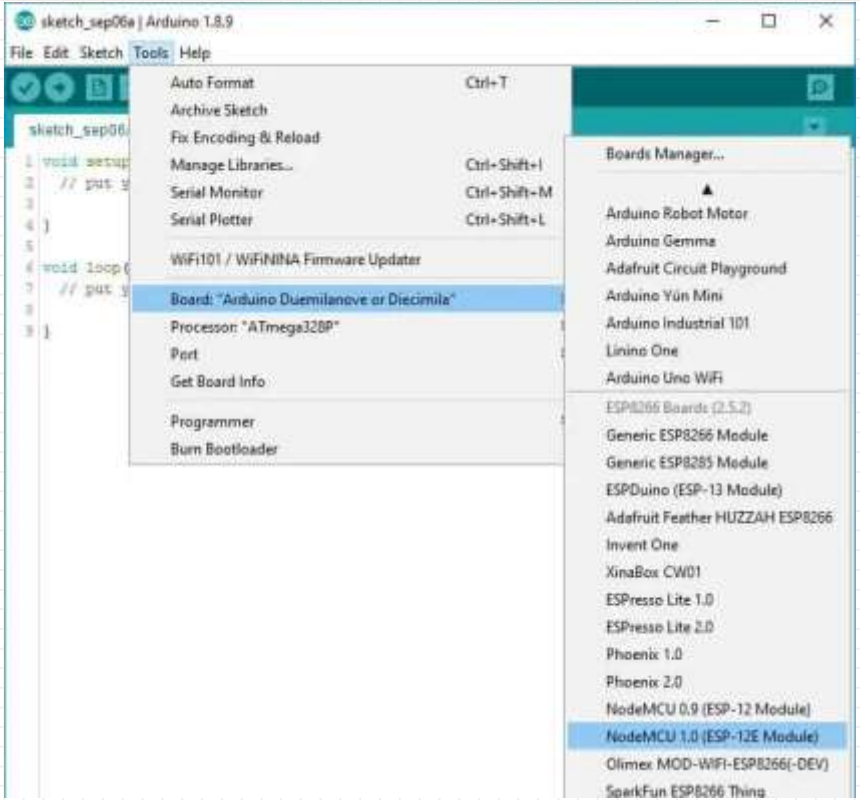
How to setup a programming environment for Node MCU in Arduino IDE.



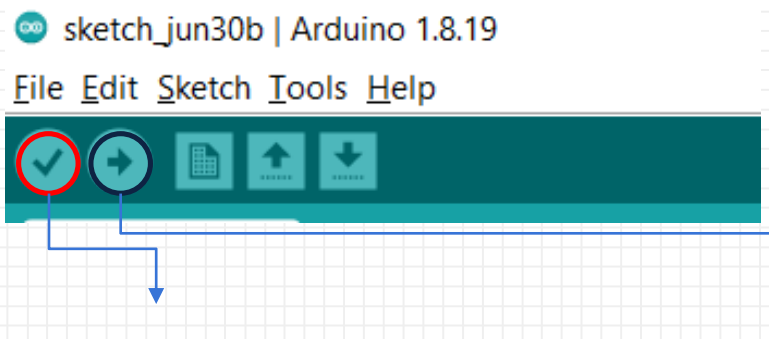


How to setup a programming environment for Node MCU in Arduino IDE.

How to setup a programming environment for Node MCU in Arduino IDE.



Uploading and Verifying of the sketch.

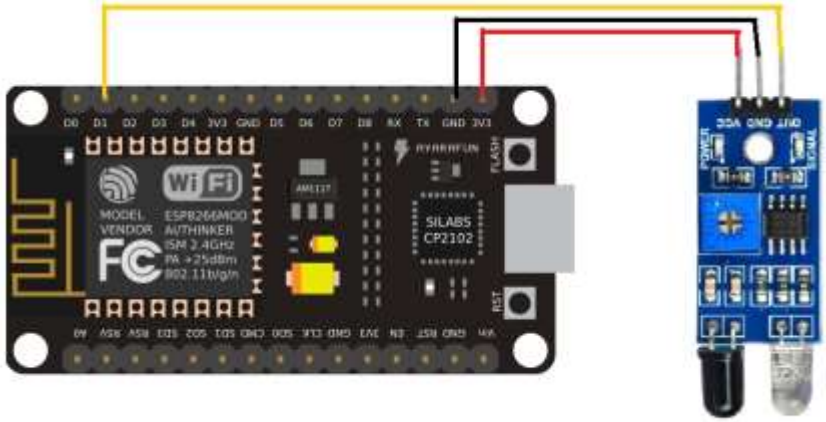


Click this for Verification of the sketch.

Click this for Uploading the sketch to the board.

Interface sensors with Node MCU.

- ✗ Connect Vcc pin of sensor to 3V3 pin of Node MCU.
- ✗ Connect GND pin of sensor to GND pin of Node MCU.
- ✗ Connect signal pin of sensor to D1 pin of Node MCU.



Interface sensors with Node MCU.

Program for IR Sensor :–

```
int IR = D1; //Define the pin Name.
void setup() {
 Serial.begin(9600); //Begin the serial communcation between Node MCU and Serial monitor.
 pinMode(IR, INPUT); // Declare the sensor as Input.
}

void loop() {
 int s = digitalRead(IR); //This will read the sensor data and store it in a variable.
 Serial.println(s); //This will print the sensor data to serial monitor.
}
```


Interface sensors with Node MCU.

IR_with_Node_MCU | Arduino 1.8.19

File Edit Sketch Tools Help

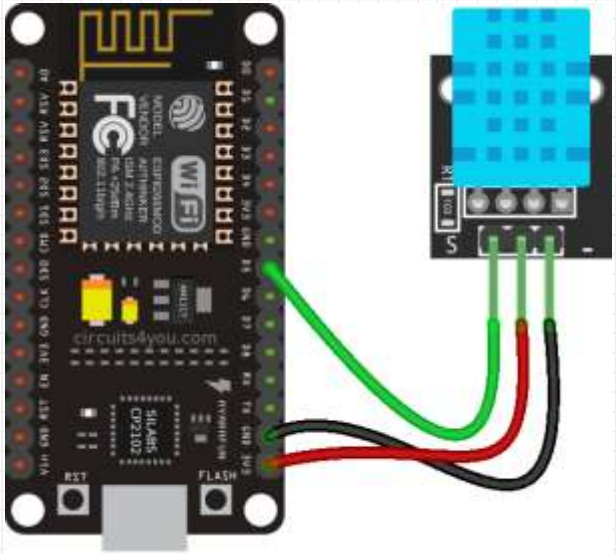
IR_with_Node_MCU §

```
int IR = D1; //Define the pin Name.
void setup() {
  Serial.begin(9600); //Begin the serial communcation between Node MCU and Serial monitor.
  pinMode(IR, INPUT); // Declare the sensor as Input.
}

void loop() {
  int s = digitalRead(IR); //This will read the sensor data and store it in a variable.
  Serial.println(s); //This will print the sensor data to serial monitor.
}
```

Interface sensors with Node MCU.

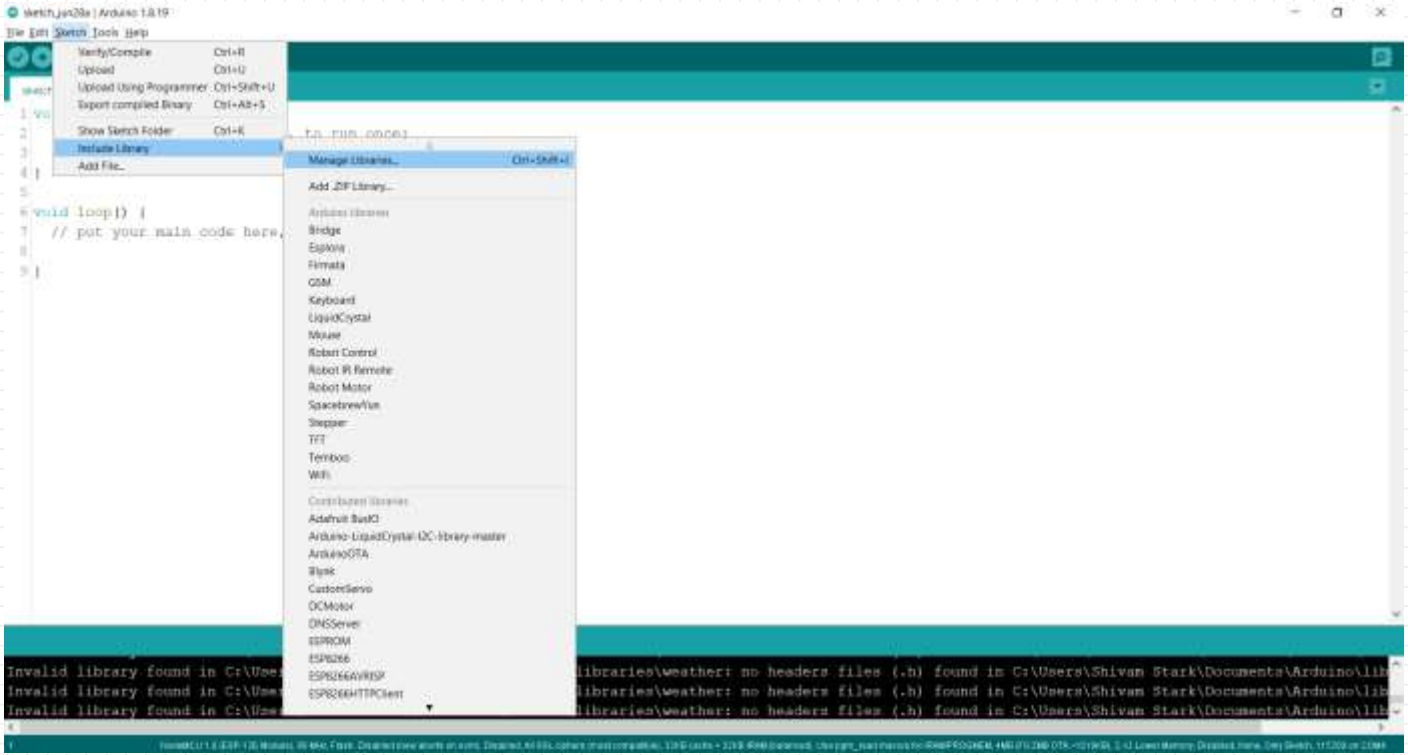
- ✗ Connect Vcc pin of sensor to 3V3 pin of Node MCU.
- ✗ Connect GND pin of sensor to GND pin of Node MCU.
- ✗ Connect signal pin of sensor to D5 pin of Node MCU.



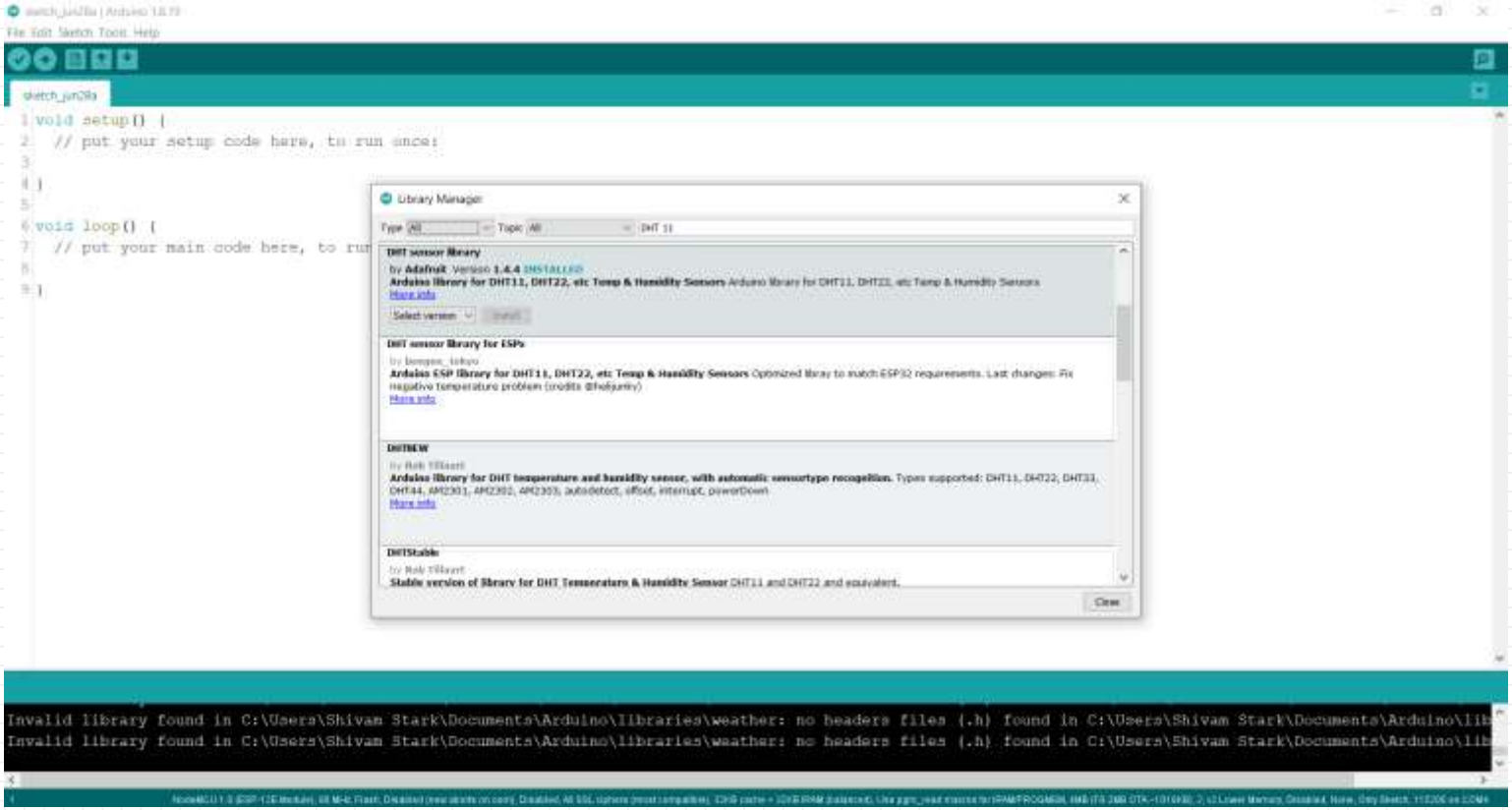
Install DHT Library.

Go to menu **Sketch → Include Library → Manage Library**

Interface sensors with Node MCU.



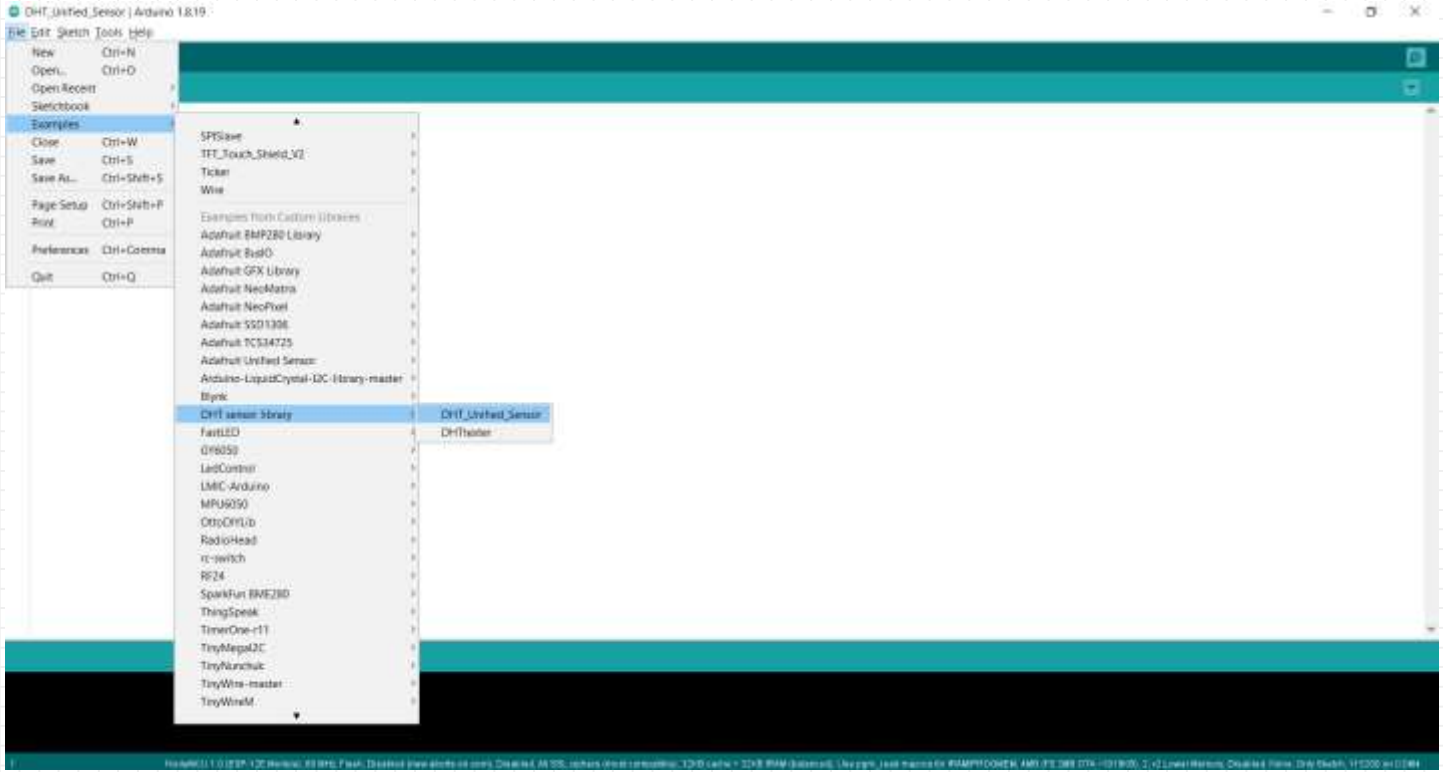
Interface sensors with Node MCU.



Program

Go to menu **File >> Example >> DHT sensor library >> DHT_Unified_Sensor**

Interface sensors with Node MCU.



Interface sensors with Node MCU.

1. Comment the 20th line in the code.
2. Uncomment the 19th line in the code.
3. Change the pin number from 2 to 5 in 14th line of the code.

Interface sensors with Node MCU.

Connect Node MCU with Internet.

Program

```
#include<ESP8266WiFi.h>
const char* ssid = "xxxxxxxxxx"; // Name of the Network.
const char* password = "xxxxxxxxxx"; // Password of the Network.
void setup() {
 Serial.begin(9600);
 WiFi.begin(ssid, password); // This will connect the Node MCU with WiFi.
 while(WiFi.status() != WL_CONNECTED){
  delay(500);
  Serial.print(".");
 }
 Serial.println("");
 Serial.println("WiFi connected");
 Serial.println(WiFi.localIP()); //This will print the Local IP address of the Network.
} void loop() {
 // put your main code here, to run repeatedly:
}
```


Connect Node MCU with Internet.

Node_MCU_WiFi_Code | Arduino 1.8.19

File Edit Sketch Tools Help

Node_MCU_WiFi_Code

```
#include <ESP8266WiFi.h>
const char* ssid = "xxxxxxxxxx"; // Name of the Network.
const char* password = "xxxxxxxxxx"; // Password of the Network.
void setup() {
  Serial.begin(9600);
  WiFi.begin(ssid, password); // This will connect the Node MCU with WiFi.
  while(WiFi.status() != WL_CONNECTED){
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println(WiFi.localIP()); //This will print the Local IP address of the Network.
}

void loop() {
  // put your main code here, to run repeatedly:

}
```

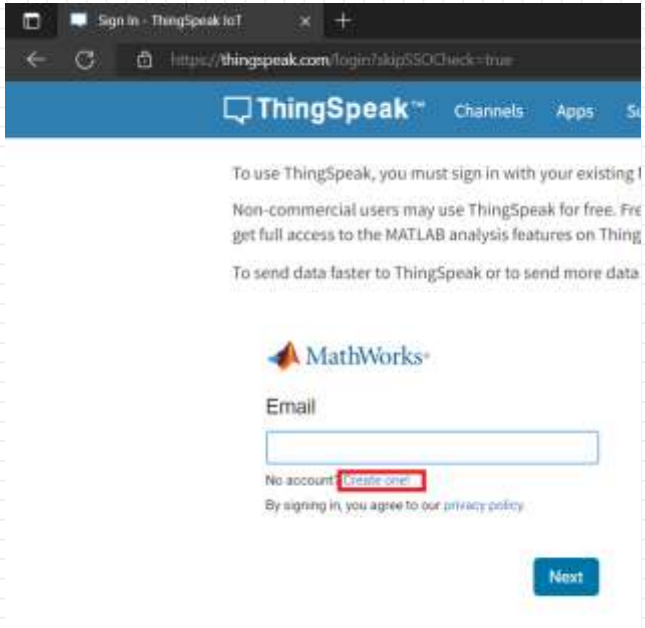
Send DHT-11 sensor Data to ThingSpeak.

Step 1 – Open this website www.thingspeak.com

Step 2– Click on **Get Stared For Free.**

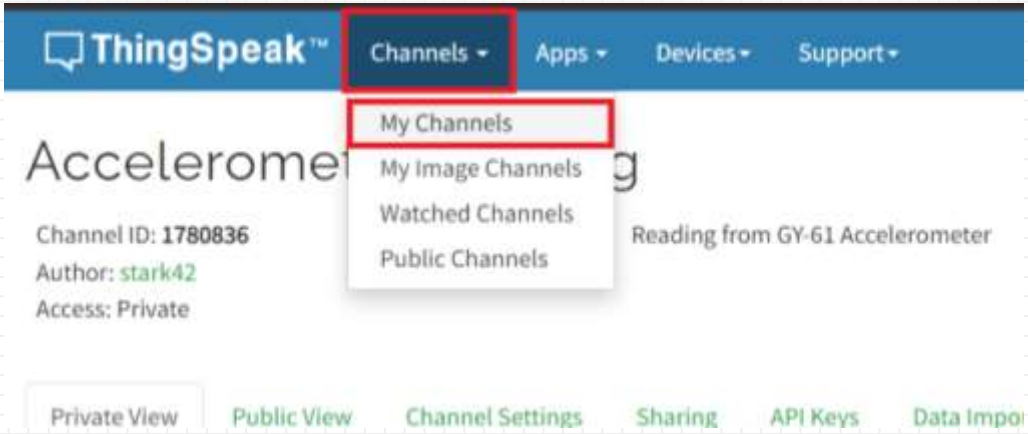
Send DHT-11 sensor Data to ThingSpeak.

Step 3 - Click on **Create one!** To create a login account.



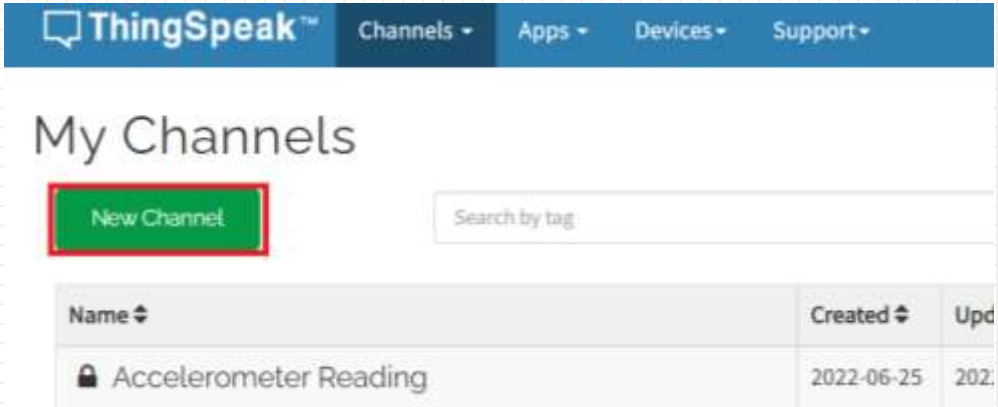
Send DHT-11 sensor Data to ThingSpeak.

Step 4 – Click on **Channels** and then click on **My Channels**



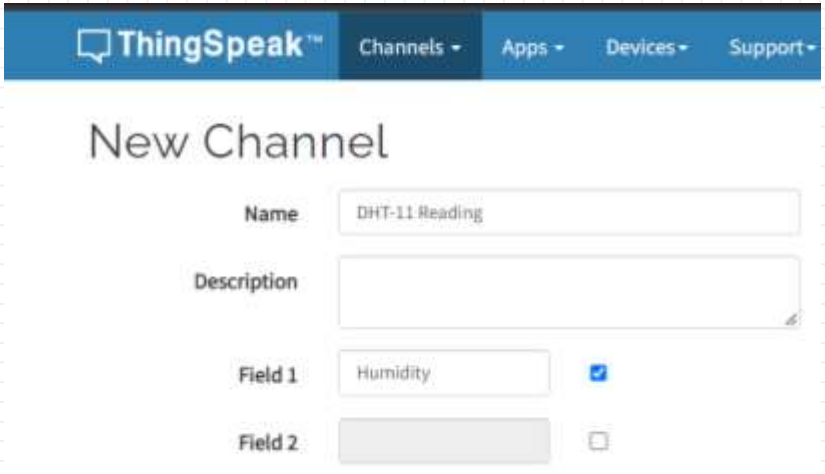
Send DHT-11 sensor Data to ThingSpeak.

Step 5 – Click on **New Channel.**



Send DHT–11 sensor Data to ThingSpeak.

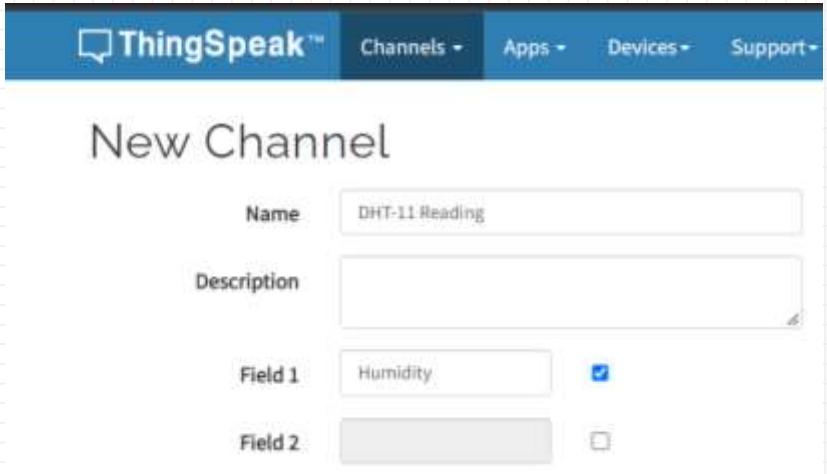
Step 6 – Write the Name of the channel and field .



Send DHT–11 sensor Data to ThingSpeak.

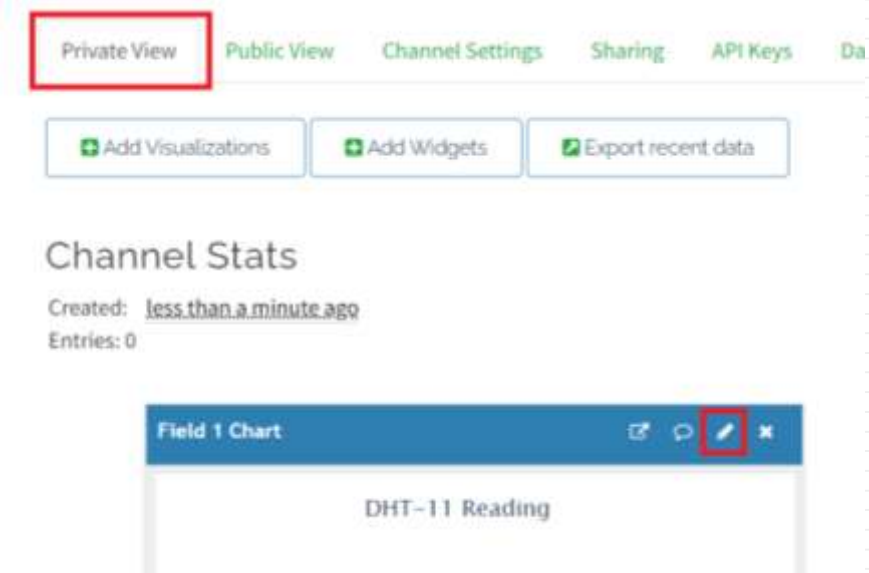
Step 7 – Write the Name of the channel and field .

Step 8 – Click the Save Channel button to create and save your channel.



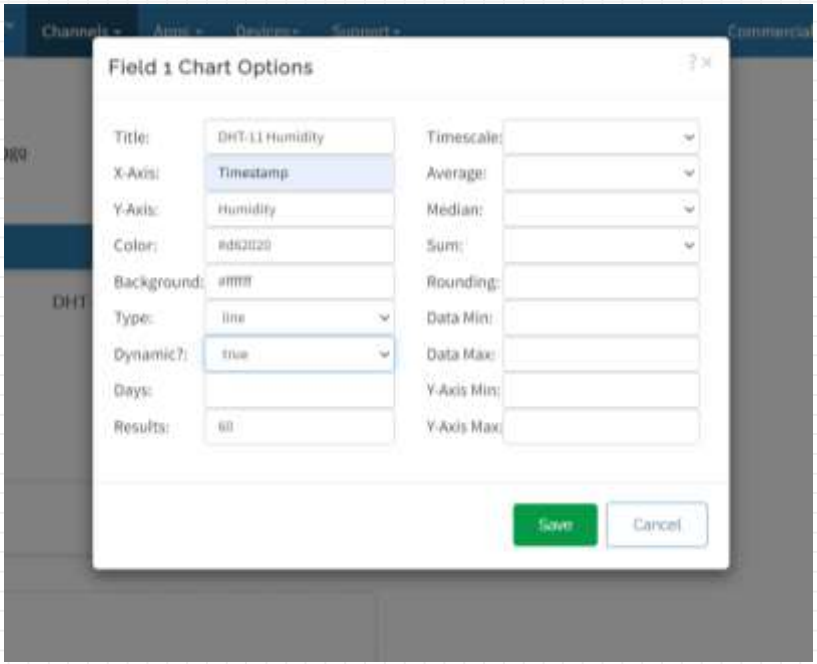
Send DHT-11 sensor Data to ThingSpeak.

Step 9 – Go to your Private View tab and click on the edit icon.



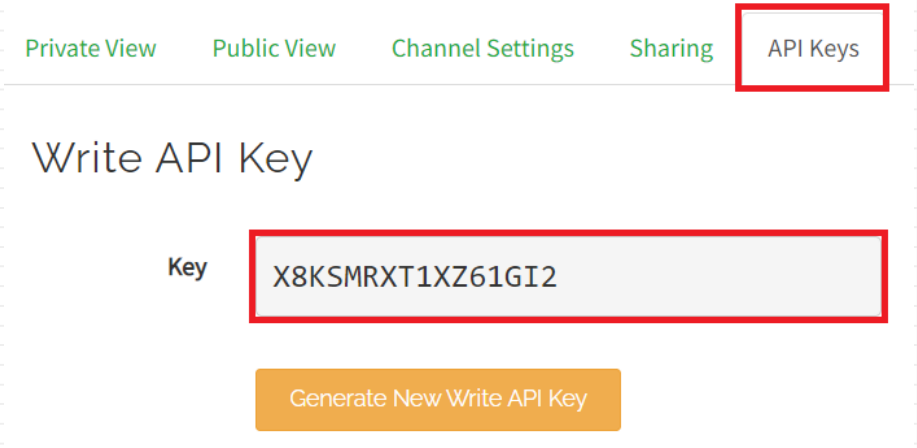
Send DHT–11 sensor Data to ThingSpeak.

Step 10 – You can give a title to your chart, customize the background color, x and y axis, and much more.



Send DHT–11 sensor Data to ThingSpeak.

Step 11 – To send values from the Node MCU to ThingSpeak, you need the Write API Key. Open the "API Keys" tab and copy the Write API Key.



Send DHT–11 sensor Data to ThingSpeak.

Program

```
#include <ESP8266WiFi.h>
#include "ThingSpeak.h"
#include <Adafruit_BMP280.h>
#include <DHT.h>

const char* ssid = "xxxxxxxxxx"; // Name of the Network.
const char* password = "xxxxxxxxxx"; // Password of the Network.

WiFiClient client; // Create a Wi-Fi client to connect to ThingSpeak.

unsigned long myChannelNumber = 1; // Channel No
const char * myWriteAPIKey = "xxxxxxxxxx"; // API write key.
```

Send DHT–11 sensor Data to ThingSpeak.

```
#define DHTPIN D5
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);

void setup() {
 Serial.begin(115200);
 WiFi.begin(ssid, password); // This will connect the Node MCU with WiFi.
 WiFi.mode(WIFI_STA); // Set the Node MCU as a Wi-Fi station.
 ThingSpeak.begin(client); // Initialize ThingSpeak.

 dht.begin();
}

void loop() {
 while (WiFi.status() != WL_CONNECTED) {
  Serial.print(".");
```

Send DHT–11 sensor Data to ThingSpeak.

```
delay(500);
 }
 float h = dht.readHumidity();
 Serial.println("\nConnected. ");
 ThingSpeak.writeField(myChannelNumber, 1, h, myWriteAPIKey);
 //the channel number.
 //the field number (in our case, we just have one field).
 //the value you want to publish (x).
 //your write API key.
}
```

Send DHT-11 sensor Data to ThingSpeak.

Node_MCU_Thingspeak_code | Arduino 1.8.19

File Edit Sketch Tools Help

Node_MCU_Thingspeak_code

```
#include <ESP8266WiFi.h>
#include "ThingSpeak.h"
#include <Adafruit_BMP280.h>
#include <DHT.h>

const char* ssid = "xxxxxxxxxx"; // Name of the Network.
const char* password = "xxxxxxxxxx"; // Password of the Network.

WiFiClient client; // Create a Wi-Fi client to connect to ThingSpeak.

unsigned long myChannelNumber = 1; // Channel No
const char * myWriteAPIKey = "xxxxxxxxxx"; // API write key.


#define DHTPIN D5
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(115200);
  WiFi.begin(ssid, password); // This will connect the Node MCU with WiFi.
  WiFi.mode(WIFI_STA); // Set the Node MCU as a Wi-Fi station.
  ThingSpeak.begin(client); // Initialize ThingSpeak.

  dht.begin();
}

void loop() {
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(500);
  }
  float h = dht.readHumidity();
  Serial.println("\nConnected. ");
  ThingSpeak.writeField(myChannelNumber, 1, h, myWriteAPIKey);
  //the channel number.
  //the field number (in our case, we just have one field).
  //the value you want to publish (x).
  //your write API key.
}
```

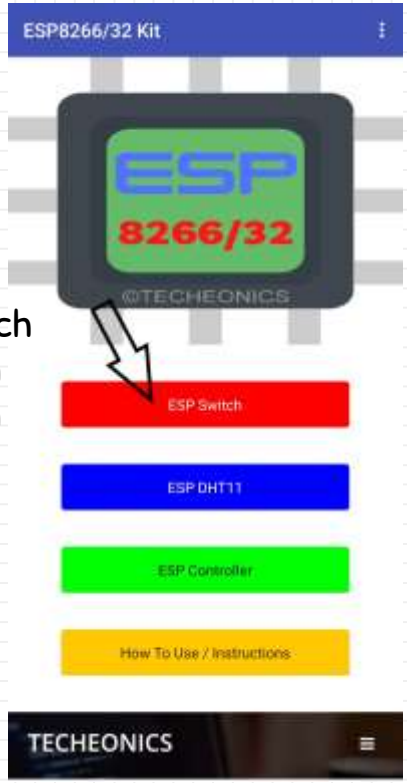
Done Saving

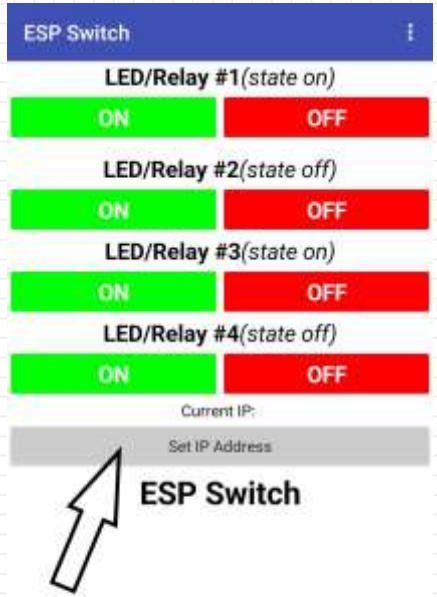
Download mobile application for ESP 8266/32.

Download ESP 8266/32 KIT APP –
GitHub – gauravk5/ESP-8266-32-KIT-APP: ESP 8266/32 KIT APP

HOW TO USE ESP SWITCH FEATURE

To use the switch feature click on the ESP Switch button

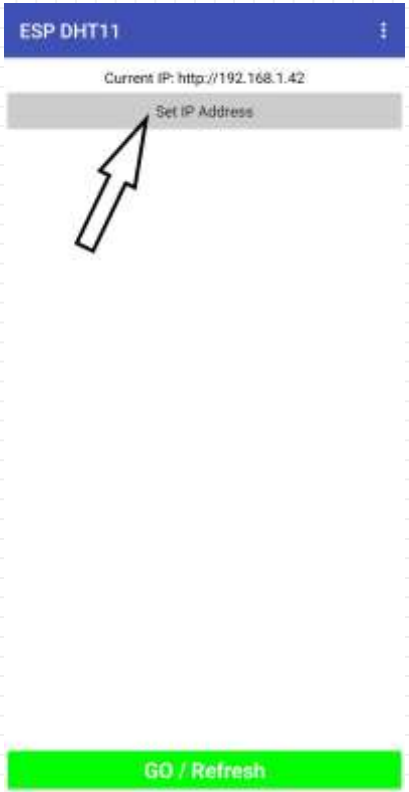




Now put the IP address of you Node MCU and Save it.

HOW TO USE ESP DHT11 FEATURE

To use the Sensor Dashboard feature click on the ESP DHT11 button



Now Click on the Set IP Address Button

Now put the IP address of your Node MCU and Save it.

Click on the GO button

Final Result
Dashboard
Sensor Data
Output

Any Questions ??

Reference - https://techeonics.com/

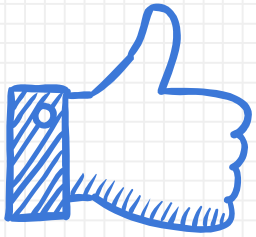
Projects - https://techeonics.com/?page_id=9

Youtube - https://www.youtube.com/c/THETECHEONICS

Contect us-

Gaurav Kumar - https://linktr.ee/Gauravk5

Shivam - https://www.linkedin.com/in/shivam-4267a31a8/



THANKS!