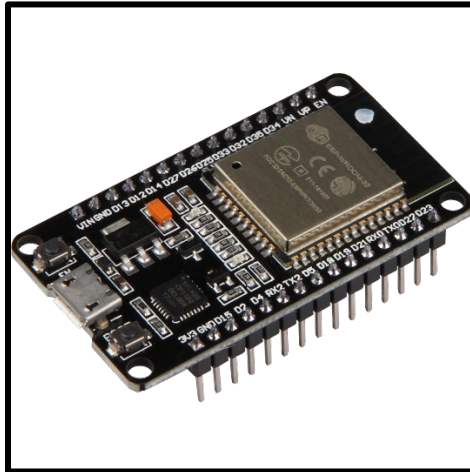# DESCRIPTION

This project to show how to monitor the data using NodeMCU with Favoriot platform. This project used ESP32 NodeMCU as a wifi module and data read from DS18B20 as a temperature sensor. FAVORIOT as a middleware platform that can show the realtime data on dashboard.
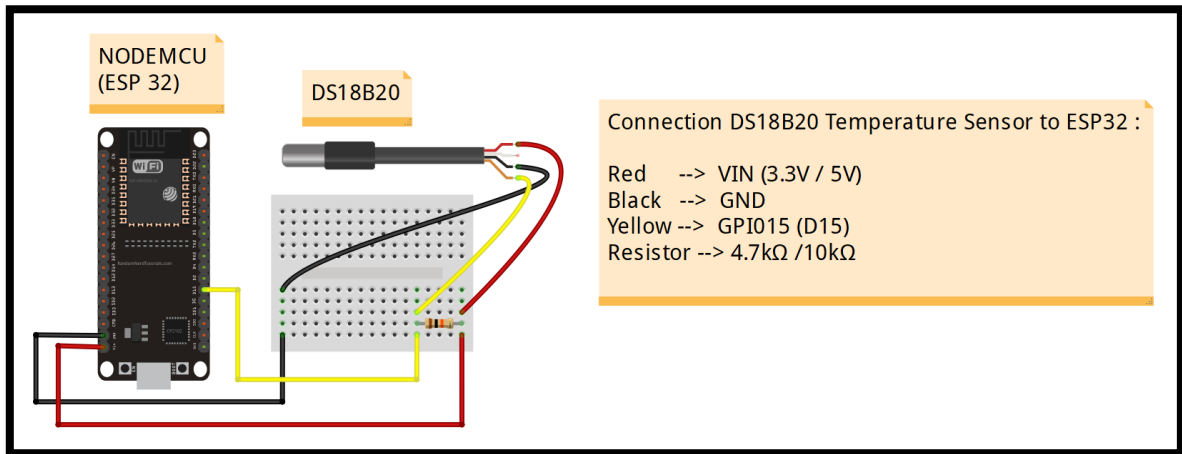
Node MCU ESP32

DS18B20 WATERPOOF TEMPERATURE SENSOR

1. Construct the circuit as shown in diagram below:

NODEMCU (ESP 32)

DS18B20

Connection DS18B20 Temperature Sensor to ESP32 :

Red   --> VIN (3.3V / 5V)
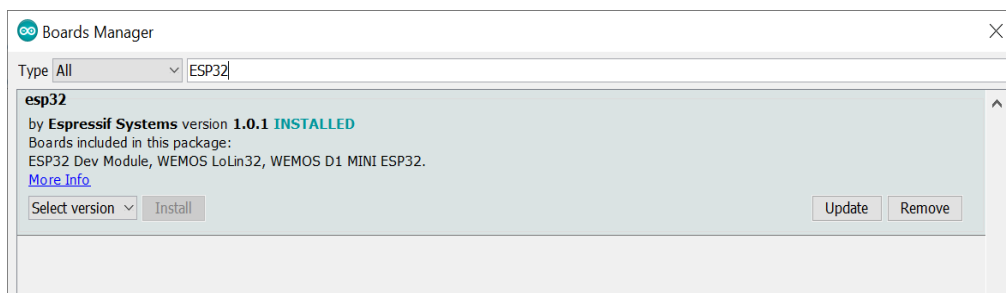Black   --> GND
Yellow --> GPIO15 (D15)
Resistor --> 4.7kΩ /10kΩ

1. Connect ESP32 board using USB MicroB cable to your USB port on your computer.
2. Open the Arduino IDE software > File > Preferences > Additional Board Manager URLs > https://dl.espressif.com/dl/package_esp32_index.json > Click OK
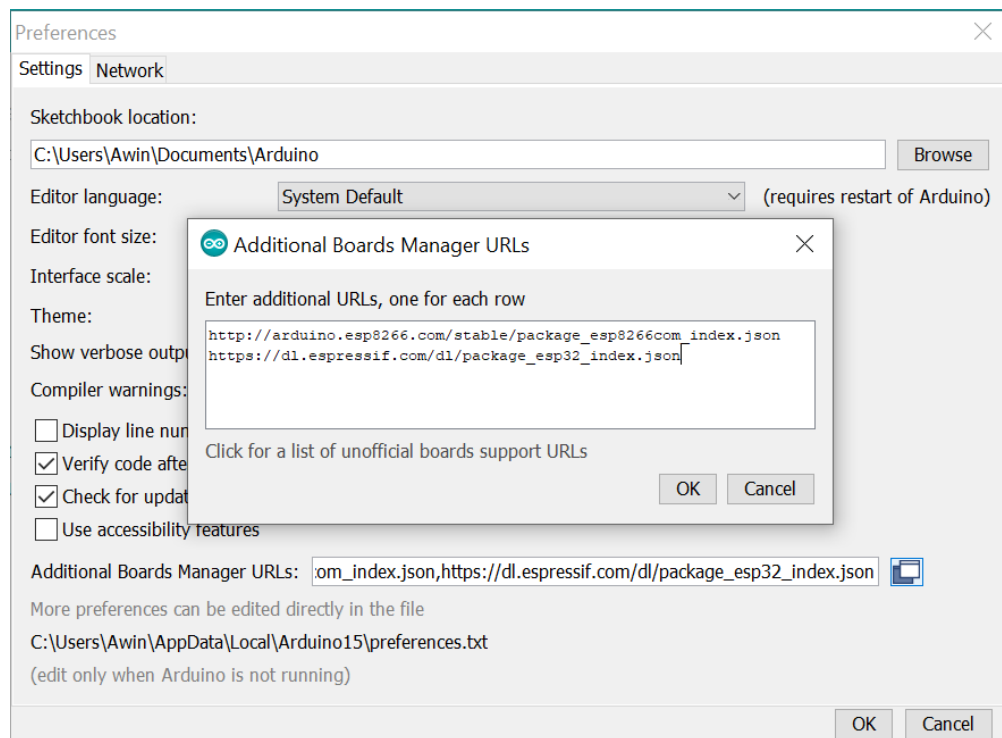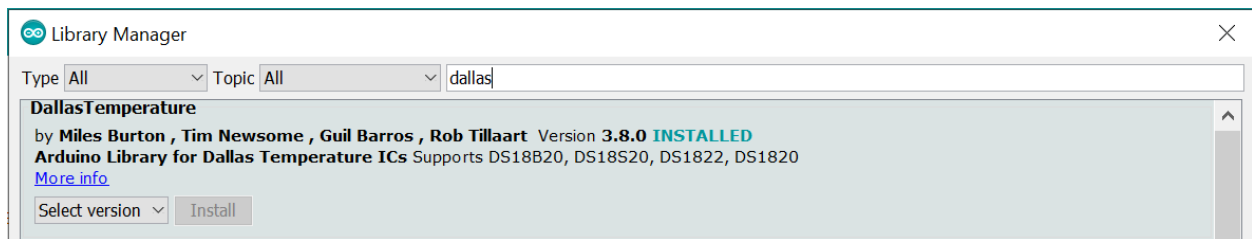

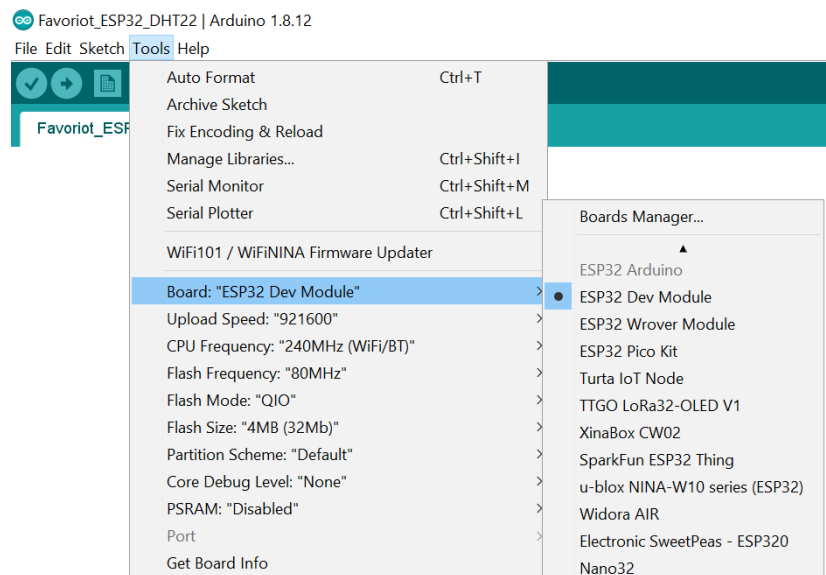3. At Arduino IDE software > Tools > Board > Boards Manager > Search ESP32, then installed

4. At Arduino IDE software > Sketch > Include Library > Manage Libraries > Search OneWire, then installed
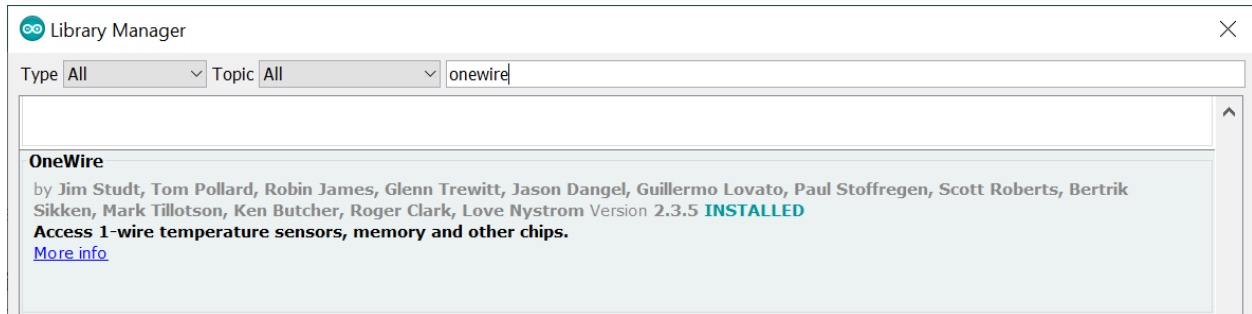
5. At Arduino IDE software > Sketch > Include Library > Manage Libraries > Search Dallas Temperature, then installed

6. At Arduino IDE software > Tools > Board > ESP32 Dev Module and Port

7. Then, write the completed code.

This is a combine code from Arduino program and FAVORIOT platform. The data will be retrieved from the sensor and will be displayed on the FAVORIOT platform in realtime. Here, the completed code for Arduino code.

```
#include <WiFi.h>
#include <HTTPClient.h>
#include <ArduinoJson.h>
#include <OneWire.h>
#include <DallasTemperature.h>

#define APIKEY  " Your API key "                      //replace API key
#define DEVICE_DEV_ID " Your device developer ID "    //replace device developer ID
#define ONE_WIRE_BUS 15      //DS18B20 data wire is connected to pin 15 on the NodeMCU

OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);

const char ssid[] = " Your wifi SSID ";             //replace wifi SSID
const char password[] = " Your Wifi Password";      //replace wifi password

unsigned long int timer;
long previousMillis = 0;
int interval = 10000; // 10 seconds

void setup()
{
  Serial.begin(115200);

  WiFi.mode(WIFI_STA);
  Serial.print("Connecting to ");
  Serial.print(ssid);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(5000);
    Serial.print(".");
  }

  Serial.println("Connected!");
  Serial.print("IP Address: ");
  Serial.println(WiFi.localIP());
}
```

```cpp
void loop()
{
  if (millis() - previousMillis > interval)
    {
    sensors.requestTemperatures();
    }

    Serial.println("Requesting temperatures: ");
    Serial.println("Celcius Temperature="+String(sensors.getTempCByIndex(0))+"degC - Fahrenheit
    Temperature="+String(sensors.getTempFByIndex(0))+"F");

    StaticJsonDocument<200> doc;

    JsonObject root = doc.to<JsonObject>(); // Json Object refer to { }
    root["device_developer_id"] = DEVICE_DEV_ID;

    JsonObject data = root.createNestedObject("data");
    data["Celcius Temperature"] = sensors.getTempCByIndex(0);
    data["Fahrenheit Temperature"] = sensors.getTempFByIndex(0);

    String body;
    serializeJson(root, body);
    Serial.println(body);

    HTTPClient http;

    http.begin("http://apiv2.favoriot.com/v2/streams");
    http.addHeader("Content-Type", "application/json");
    http.addHeader("Apikey", APIKEY);

    int httpCode = http.POST(body);
    if (httpCode > 0) {
      Serial.printf("[HTTP] POST... code: %d\n", httpCode);
      if (httpCode == HTTP_CODE_OK) {
        String payload = http.getString();
        Serial.println(payload);
      }
    }
    else {
      Serial.printf("[HTTP] POST... failed, error: %s\n", http.errorToString(httpCode).c_str());
    }
    http.end();

    previousMillis = millis();
    delay(10000);
  }
```
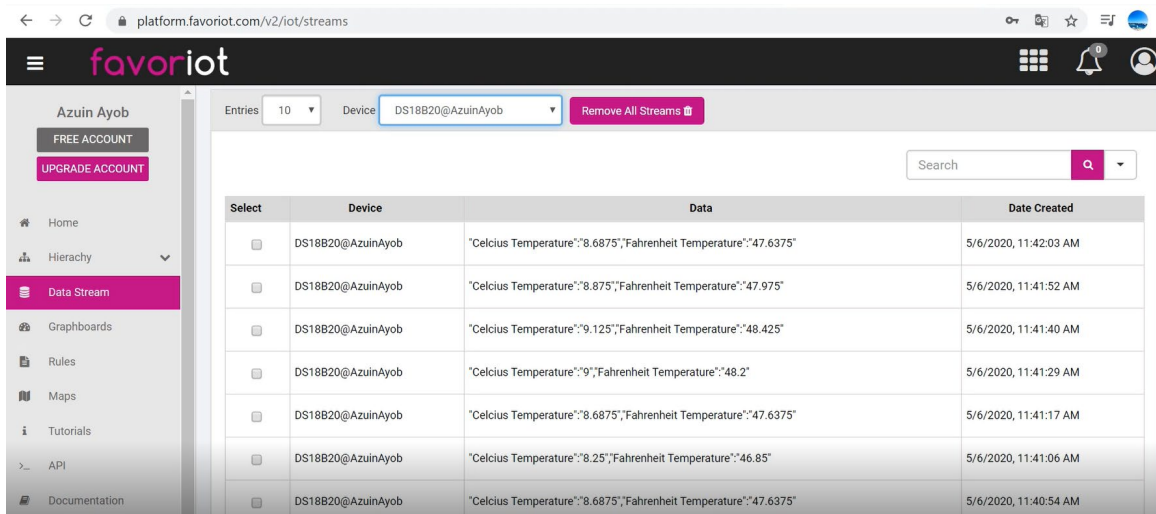
Once finished, click on the Verify button and Upload code to board. The output will be display on Serial Monitor.

Open FAVORIOT platform at browser, click at Data Stream and select a device. The output will be displayed as shown below.
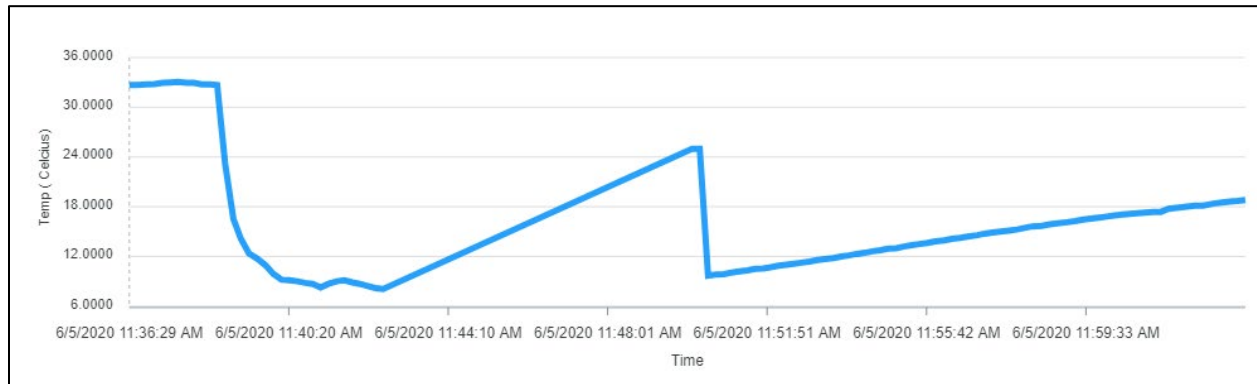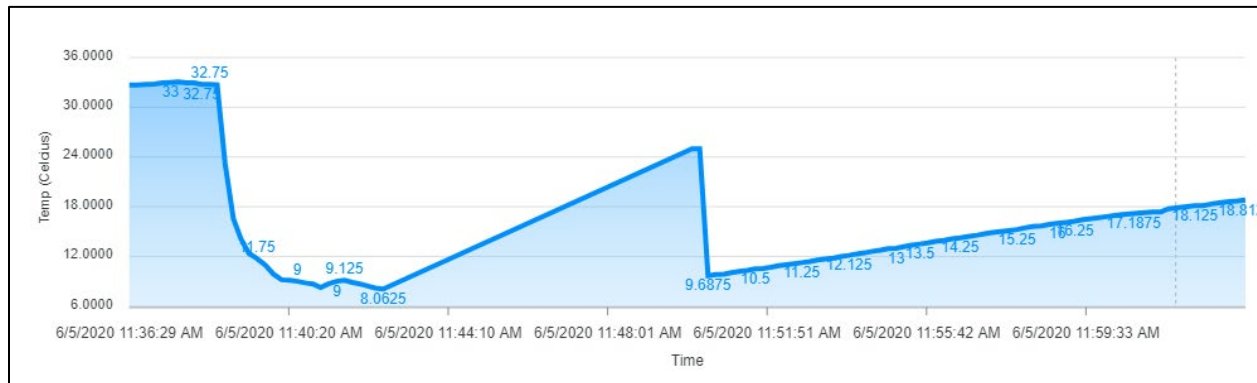
At graphboard can create various graph such as line, area, bar etc as shown below.

Line Graph

Area Graph

Bar Graph