



ESP32を使用して温度をリアルタイムでブラウザに表示するためには、ESP32がセンサーデータを取得し、そのデータをWebサーバーを介してブラウザに送信する必要があります。これを実現するためには、HTML、JavaScript（特にAjax）、およびESP32のコードが必要です。以下に手順を説明します。

必要なもの

- ESP32
- 温度センサー（例えば、DHT11、DHT22、またはDS18B20）
- Arduino IDE
- HTMLとJavaScriptの基礎知識

手順

1. Arduino IDEのセットアップ

- ESP32のボードをArduino IDEに追加します。
- DHTセンサーライブラリ（DHTセンサーを使用する場合）をインストールします。

2. ESP32のコード

ESP32のコードでは、Wi-Fiに接続し、Webサーバーを起動し、センサーデータを取得してブラウザに送信します。以下にサンプルコードを示します。

cpp

📄 コードをコピーする

```
#include <WiFi.h>
#include <WebServer.h>
#include <DHT.h>

const char* ssid = "YOUR_SSID";
const char* password = "YOUR_PASSWORD";

WebServer server(80);

#define DHTPIN 4 // DHTセンサーのピン番号
#define DHTTYPE DHT11 // 使用するDHTセンサーのタイプ (DHT11, DHT22, etc.)

DHT dht(DHTPIN, DHTTYPE);

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

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

  server.on("/", handleRoot);
  server.on("/temperature", handleTemperature);

  server.begin();
  Serial.println("HTTP server started");
}
```

```

void loop() {
  server.handleClient();
}

void handleRoot() {
  server.send(200, "text/html", R"rawliteral(
    <!DOCTYPE HTML><html>
    <head>
      <title>ESP32 Temperature</title>
      <meta http-equiv="refresh" content="5">
    </head>
    <body>
      <h1>ESP32 Temperature</h1>
      <p>Current temperature: <span id="temperature">Loading...</span> &deg;C</p>
      <script>
        function loadTemperature() {
          var xhr = new XMLHttpRequest();
          xhr.open('GET', '/temperature', true);
          xhr.onreadystatechange = function() {
            if (xhr.readyState == 4 && xhr.status == 200) {
              document.getElementById('temperature').innerText = xhr.responseText;
            }
          };
          xhr.send();
        }
        setInterval(loadTemperature, 5000); // 5秒ごとにデータを取得
        loadTemperature();
      </script>
    </body>
  </html>
)rawliteral");
}

void handleTemperature() {
  float t = dht.readTemperature();
  if (isnan(t)) {
    server.send(200, "text/plain", "Error reading temperature");
  } else {
    server.send(200, "text/plain", String(t));
  }
}

```



```
server.send(200, "text/plain", String(t));  
}  
}
```

このコードは、ESP32がWi-Fiに接続し、HTTPサーバーを起動して、温度センサーからデータを取得し、ブラウザに送信する基本的な機能を提供します。

3. HTMLとJavaScript

上記のコードの中にHTMLとJavaScriptが含まれています。JavaScriptの部分はAjaxを使用して、ESP32から温度データを5秒ごとに取得して更新します。

4. ESP32にコードを書き込む

上記のコードをArduino IDEにコピーして、適切なWi-FiのSSIDとパスワードを入力します。次に、ESP32にコードを書き込みます。

5. ブラウザで確認

ESP32がWi-Fiに接続されると、シリアルモニターにIPアドレスが表示されます。このIPアドレスをブラウザに入力すると、温度データがリアルタイムで表示されるはずです。

この手順を実行することで、ESP32を使用して温度データをリアルタイムでブラウザに表示することができます。



Arduino Uno R4 WiFiを使用して温度と湿度を2秒間隔でブラウザにリアルタイム表示する方法を説明します。
DHT11またはDHT22センサーを使用する例を示します。

必要なもの

- Arduino Uno R4 WiFi
- DHT11またはDHT22センサー
- Arduino IDE
- HTMLとJavaScriptの基礎知識

手順

1. Arduino IDEのセットアップ

- Arduino IDEにUno R4 WiFiのボードを追加します。
- DHTセンサーライブラリをインストールします。

2. Arduino Uno R4 WiFiのコード

Arduinoのコードでは、Wi-Fiに接続し、Webサーバーを起動し、センサーデータを取得してブラウザに送信します。
以下にサンプルコードを示します。

cpp

コードをコピーする

```
#include <WiFiNINA.h>
#include <DHT.h>

const char* ssid = "YOUR_SSID";
const char* password = "YOUR_PASSWORD";

WiFiServer server(80);

#define DHTPIN 4 // DHTセンサーのピン番号
#define DHTTYPE DHT22 // 使用するDHTセンサーのタイプ (DHT11, DHT22, etc.)
```



ブラウザに温度と湿度を表示するだけでなく、出力ON/OFFのスイッチを2つ追加するためのコードを以下に示します。
出力ON/OFFスイッチは、例えばLEDの制御に使用できるように設計します。

Arduino Uno R4 WiFiのコード

cpp

コードをコピーする

```
#include <WiFiNINA.h>
#include <DHT.h>

const char* ssid = "YOUR_SSID";
const char* password = "YOUR_PASSWORD";

WiFiServer server(80);

#define DHTPIN 4 // DHTセンサーのピン番号
#define DHTTYPE DHT22 // 使用するDHTセンサーのタイプ (DHT11, DHT22, etc.)
#define LED_PIN1 12 // LED 1のピン
#define LED_PIN2 13 // LED 2のピン

DHT dht(DHTPIN, DHTTYPE);

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

  pinMode(LED_PIN1, OUTPUT);
  pinMode(LED_PIN2, OUTPUT);

  // WiFi接続の試行
  Serial.print("Connecting to ");
  Serial.println(ssid);
  WiFi.begin(ssid, password);
```



ブラウザに温度と湿度を表示するだけでなく、出力ON/OFFのスイッチを2つ追加するためのコードを以下に示します。
出力ON/OFFスイッチは、例えばLEDの制御に使用できるように設計します。

Arduino Uno R4 WiFiのコード

cpp

コードをコピーする

```
#include <WiFiNINA.h>
#include <DHT.h>

const char* ssid = "YOUR_SSID";
const char* password = "YOUR_PASSWORD";

WiFiServer server(80);

#define DHTPIN 4 // DHTセンサーのピン番号
#define DHTTYPE DHT22 // 使用するDHTセンサーのタイプ (DHT11, DHT22, etc.)
#define LED_PIN1 12 // LED 1のピン
#define LED_PIN2 13 // LED 2のピン

DHT dht(DHTPIN, DHTTYPE);

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

  pinMode(LED_PIN1, OUTPUT);
  pinMode(LED_PIN2, OUTPUT);

  // WiFi接続の試行
  Serial.print("Connecting to ");
  Serial.println(ssid);
  WiFi.begin(ssid, password);
```



スイッチをモーメンタリにするには、ボタンを押したときに一時的にLEDを点灯させ、一定時間後に自動的に消灯させる必要があります。以下に、モーメンタリスイッチの機能を持つコードを示します。

Arduino Uno R4 WiFiのコード

```
cpp コードをコピーする

#include <WiFiNINA.h>
#include <DHT.h>

const char* ssid = "YOUR_SSID";
const char* password = "YOUR_PASSWORD";

WiFiServer server(80);

#define DHTPIN 4 // DHTセンサーのピン番号
#define DHTTYPE DHT22 // 使用するDHTセンサーのタイプ (DHT11, DHT22, etc..)
#define LED_PIN1 12 // LED 1のピン
#define LED_PIN2 13 // LED 2のピン

DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(115200);
  dht.begin();
}
```

HTMLとJavaScript

上記のコードには、HTMLとJavaScriptも含まれています。JavaScriptはAjaxを使用して、2秒ごとにセンサーデータを取得して更新し、ボタンを押すとLEDの状態を一時的に切り替えます。

Arduinoにコードを書き込む

上記のコードをArduino IDEにコピーし、適切なWi-FiのSSIDとパスワードを入力します。次に、Arduinoにコードを書き込みます。

ブラウザで確認

ArduinoがWi-Fiに接続されると、シリアルモニターにIPアドレスが表示されます。このIPアドレスをブラウザに入力すると、温度と湿度のデータがリアルタイムで表示され、LEDのモーメンタリスイッチも表示されます。ボタンを押すと、対応するLEDが一時的に点灯し、1秒後に自動的に消灯します。

この手順を実行することで、Arduino Uno R4 WiFiを使用して温度と湿度のデータを2秒間隔でブラウザに表示し、モーメンタリの出力ON/OFFスイッチを2つ追加することができます。

