

# IOT HTTP Server

此章節解說如何架設一個 IOT HTTP Server 結合溫溼度感測器。  
如果對於建置環境不了解，先參考「NodeMCU\_HelloWorld」章節。  
如果對於溫溼度感測器不了解，先參考「DHT11」章節。

架構圖：



架構介紹：

1. 電腦A連線上 WIFI-AP
2. ESP8266連上 WIFI-AP(僅支援 2.4G)
3. 電腦A打開網頁並輸入ESP8266的IP
4. ESP8266 取得溫度及溼度並回傳網頁給電腦A

## 1. 編寫草稿碼 -> 上傳至 NodeMCU 開發板

```
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
#include <ESP8266WebServer.h>

#define STASSID "cswork_2.4G"
#define STAPSK "888888888"

const char* ssid = STASSID;
const char* password = STAPSK;
ESP8266WebServer server(80);

//Sensor-VCC -> DUT-3V pin
#include <SimpleDHT.h>

int pinDHT11 = D4; //D4 Pin腳 讀取DATA
SimpleDHT11 dht11; //dht11 class

byte gTemperature = 0;
byte gHumidity = 0;

void handleRoot() {
    char sendStr[1024];

    memset(sendStr, 0, sizeof(sendStr));

    //取得 溫度 & 濕度
    int err = SimpleDHTErrSuccess;
    if ((err = dht11.read(pinDHT11, &gTemperature, &gHumidity, NULL)) != SimpleDHTErrSuccess) {
        //讀取失敗的話，清為0
        gTemperature = 0;
        gHumidity = 0;
    }

    sprintf(sendStr, "目前溫度 %d C, 濕度 %d %%", gTemperature, gHumidity);
    server.send(200, "text/plain", sendStr);
}

void handleNotFound() {

    String message = "File Not Found\n\n";
    message += "URI: ";
    message += server.uri();
    message += "\nMethod: ";
    message += (server.method() == HTTP_GET) ? "GET" : "POST";
```

```

message += "\nArguments: ";
message += server.args();
message += "\n";
for (uint8_t i = 0; i < server.args(); i++) {
    message += " " + server.argName(i) + ": " + server.arg(i) + "\n";
}
server.send(404, "text/plain", message);
}

void setup(void)
{
    //console baud rate
    Serial.begin(115200);
    WiFi.mode(WIFI_STA);
    WiFi.begin(ssid, password);
    printf("\n");

    // Wait for WiFi connection
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }
    printf("\n");
    printf("Connected to %s, IP:%s\n", ssid, WiFi.localIP().toString().c_str());

    //定義 HTTP Server 路由
    //首頁
    server.on("/", handleRoot);

    //當 HTTP Server 路由找不到時，進入 Not Found
    server.onNotFound(handleNotFound);
    //HTTP Server 啟動
    server.begin();
    printf("HTTP server started\n");
}

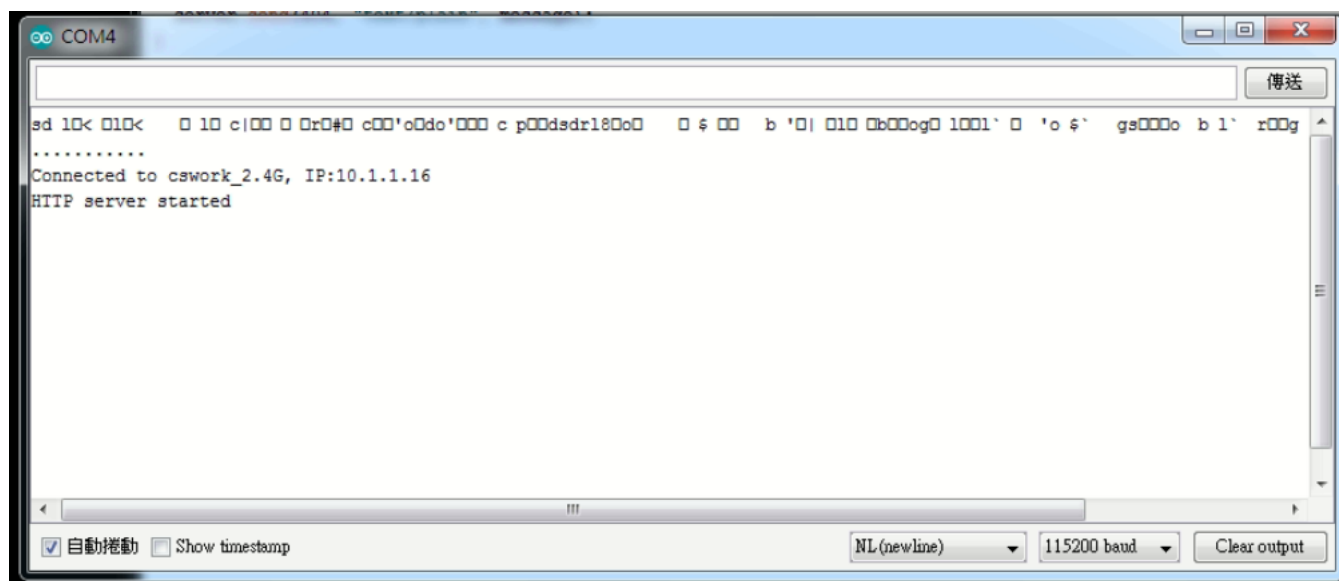
void loop(void) {
    //檢查是否有連線進入
    server.handleClient();
}

```

從程式碼可看出：

- 1.ESP8266連上WIFI得到IP
- 2.HTTP Server除了首頁外，還定義了 HTTP 路由找不到時，回傳 Not Found的頁面。
- 3.訪問首頁時，ESP8266會先取得DHT11的溫度與溼度，並將字串重組完整後，再傳送網頁。

## 2. 觀看結果

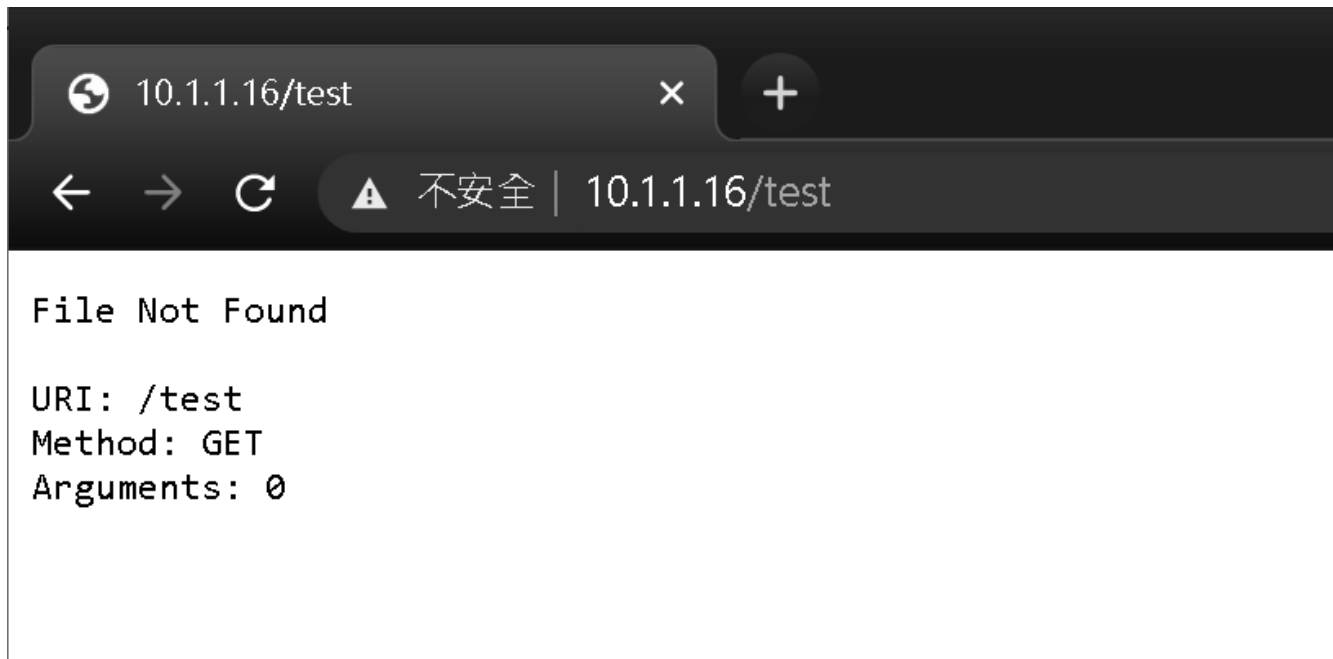


ESP8266 連上 WiFi-AP 取得 IP(10.1.1.16)後，啟動HTTP Server



首頁顯示目前溫度與溼度

**note:**如果出現亂碼的話，很可能是中文編碼在網頁顯示的問題，建議將中文換英文即可。



連上HTTP 路由找不到的頁面，得到 Not Found