HTTP REST Server - Node MCU

Objectives : -

- Connecting to an existing Wi-Fi network
- Setting up Node MCU as an Wi-Fi Access Point
- Create a HTTP Server hosted by Node MCU
- DNS masking of IP assigned by the AP
- Controlling GPIO Through REST APIs
- Reading ADC Value through REST APIs

Connecting to an existing Wi-Fi network

#include <ESP8266WiFi.h>

- Header File

WiFi.begin(<ssid>,<password>);

- Setting SSID and Password of the AP

while (WiFi.status() != WL_CONNECTED)

- Check Wi-Fi status

WiFi.localIP()

- Read IP Address Assigned

Setting up Node MCU as an Wi-Fi Access Point

include<ESP8266WebServer.h>

- Header File

ESP8266WebServer server;

- Web server variable

```
server.on("/", []() {
server.send(200,"text/plain", "Hello World.. ! ");
});
```

- Home Page of the server

server.on("/<path>", <callback Function>);

- Setting up a function that will be executed if the <path> is called

server.handleClient();

- Default method to handle the clients

Create a HTTP Server hosted by Node MCU

#include <ESP8266WiFi.h>

- Header File

IPAddress ip(192, 168, 11, 4);

- Setting up IP for setting up as an AP

IPAddress gateway(192, 168, 11, 1);

- Setting up Gateway for setting up as an AP

IPAddress Subnet(255,255,255,0)

- Setting up Subnet for setting up as an AP

WiFi.softAPConfig(ip, gateway, subnet);

- Applying options for AP

WiFi.softAP(<SSID>,<Password>)

- Starting the Wifi AP

DNS masking of IP assigned by the AP

```
#include <ESP8266mDNS.h>
```

- Header File

```
MDNSResponder mdns;
```

- Dns Responder variable

```
MDNS.addService("http", "tcp", 80);
```

- Adding the server

Controlling GPIO Through REST APIs

```
server.on("/toggle", toggleLED);
```

- Go to <ip>/toggle and it call toggleLED Function

```
void toggleLED()
{
flag = 0;
digitalWrite(pin_led, !digitalRead(pin_led));
server.send(200, "text/plain", "LED Toggled!");
}
```

- Callback function of /toggle
- Toggling the LED
- Displaying LED Toggled on the Web Page

Reading ADC Through REST APIs

```
server.on("/adc", adcRead);
```

- Go to <ip>/adc and it call adcRead Function

```
void adcRead()
{
int sens = analogRead(A0);
String one1 = String(sens);
String one =String("ADC: "+one1);
server.send(200,"text/plain", one);
}
```

- Reading analog data
- Creating String to be displayed as a web page

Thank You

Follow the link for sample code : -

https://github.com/ersanyamarya/Node-Mcu-HTTP-REST-Server