

## Lab 1:

Buat sebuah koneksi dengan HotSPot.

Solusi:

SSID = "\_\_\_\_\_"

PASSWORD = "\_\_\_\_\_"

```
#include <WiFi.h>
const char *SSID = "inixindo";
const char *PASSWORD = "270755";

void setup() {
  Serial.begin(115200);
  delay(1000);
  WiFi.disconnect(true);
  connectToHotSpot();
}
void connectToHotSpot() {

  WiFi.begin( SSID, PASSWORD );
  Serial.print("Trying to establish connection to WiFi Router");

  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.print(".");
  }
  Serial.println();
  Serial.println("Connected! IP: ");
  Serial.println(WiFi.localIP());
}
void loop() {
  // nothing here
}
```

Challenge:

Tambahkan informasi tentang MacAddress

Connected! IP: xxx.xxx.xxx.xxx dan Mac Address: xxxx.xxxx.xxxx");

## Lab 2:

Buat sebuah Web Server , dimana Client dapat mengakses nya dengan menggunakan LINK :

<p><a href="blink">LED Blinking</a></p>

<p><a href="on">LED ON</a></p>

<p><a href="off">LED OFF</a></p>

Rekomendasi: Jalankan dulu program contoh berikut kemudian dikembangkan sesuai dengan permintaan diatas.

```
#include <WiFi.h>
#define PORT 8080
#define LED_BUILTIN 2
const char *SSID = "_____";
const char *PASSWORD = "xxxxxxxxxx";
String header= "HTTP/1.1 200 OK\nContent-Type: text/
html\n\n<html><body>";
String tailer= "</body></html>";
WiFiServer server(8080);
WiFiClient client;

bool blinking=false;
bool ledOn= false;

String s;
String cmd;
void setup() {
  Serial.begin(115200);
  delay(1000);
  pinMode(LED_BUILTIN, OUTPUT);
  WiFi.begin(SSID, PASSWORD);
  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.print(".");
  }
  Serial.println("Web Server active: ");
  Serial.println( WiFi.localIP() );
  server.begin();
}

void loop() {
  // put your main code here, to run repeatedly:
  client= server.available(); //tunggu client
  //Serial.println("waiting for client ");
  if (client) { // dapat client baru
    // Serial.println("got a new client");
    while (client.connected() ) {
      if ( client.available() ) {
        //Serial.println("Data can be read");
```

```

        s = client.readString();
        cmd= getCommand(s);
        response(cmd);
        if (cmd.equals("")) {
            Serial.println("HOMEPAGE");
        } else if(cmd.equals("blink")) {
            blinking= true;
            ledOn= false;

            } else if(cmd.equals("on") ){
                blinking= false;
                ledOn= true;
            } else if(cmd.equals("off")) {
                blinking= false;
                ledOn= false;
            } else { // unknown command
                response(s);
                client.stop();
            }
            //client.stop();
            //Serial.println(cmd);
            //delay(100);
        }
        else {
            client.stop();
        }
    }
    //client.stop();
}
else {
    //Serial.println("No client yet ");
}
if (blinking || ledOn) {
    digitalWrite(LED_BUILTIN, HIGH);
    delay(1000);
}
if (blinking || ! ledOn) {
    digitalWrite(LED_BUILTIN, LOW);
    delay(1000);
}
}

void response(String konten) {
    //Serial.println(header);
    //Serial.println(konten);
    //Serial.println(tailer);
    client.println(header);
    client.println(konten);
    client.println(tailer);
}

String getCommand(String s) {
    return (s.substring(5, s.indexOf(' ',5)));
}

```

## Lab 3:



Program berikut membuat Input di Browser dengan menampilkan informasi data sensor. Bila tombol Refresh diipilih, maka Browser meminta data baru dari Microcontroller yang terkoneksi dengan sensor.

Sebelum terkoneksi dengan Sensor, eksekusi program berikut yang menggunakan random generator sebagai input.

### Challenge:

Ganti random generator dengan sensor yang diberikan oleh Fasilitator.

```
#include <WiFi.h>
#define PORT 8080
#define LED_BUILTIN 2
const char *SSID = "_____";
const char *PASSWORD = "xxxxxx";
String header= "HTTP/1.1 200 OK\nContent-Type: text/html\n\n";
String tailer= "</body></html>";
WiFiServer server(PORT);
WiFiClient client;

long randomnumber;

String s;
String cmd;
void setup() {
    Serial.begin(115200);
    delay(1000);
    pinMode(LED_BUILTIN, OUTPUT);
    WiFi.begin(SSID, PASSWORD);
    while (WiFi.status() != WL_CONNECTED) {
```

```

        delay(1000);
        Serial.print(".");
    }
    Serial.print("Web Server active: ");
    Serial.println( WiFi.localIP() );
    server.begin();
    randomSeed(42);
}

```

```

void loop() {

```

```

    client= server.available(); //tunggu client
    //Serial.println("waiting for client ");
    if (client) { // dapat client baru
        Serial.println("got a new client");
        while (client.connected() ) {
            if ( client.available() ) {
                //Serial.println("Data can be read");
                s = client.readString();
                cmd= getCommand(s);
                if (cmd.equals("")) {
                    refresh(0);
                } else if(cmd.equals("refresh?")) {
                    refresh(random(0,10000)); // min max
                } else {
                    refresh(0);
                    client.stop();
                }
                Serial.println(cmd);
                delay(100);
            }
            else {
                client.stop();
            }
        }
    }

```

```

        }
        //client.stop();
    }

```

```

    else {
        //Serial.println("No client yet ");
    }

```

```

    delay(1000);
}

```

```

void refresh (int nilai) {
    client.println(header);
    client.print ("<html><head><title>Sensor</title>");
    client.print ("<meta name='viewport' content='width=device-width,
initial-scale=1.0'>");
    client.print ("</head>");
    client.print ("<body><h1>Data Sensor</h1>");
    client.printf("<p>Nilai Data Dari Sensor adalah : %d </p>", nilai);
    client.print ("<p><form action='/refresh'><input type='submit'
value='Refresh'></p>");
}

```

```
        client.println(tailer);
    }

    String getCommand(String s) {
        int i;
        i= s.indexOf(' ',5);
        return (s.substring(5, i));
    }
}
```

Setelah program tersebut berjalan dengan baik, ganti random input dengan Sensor.

