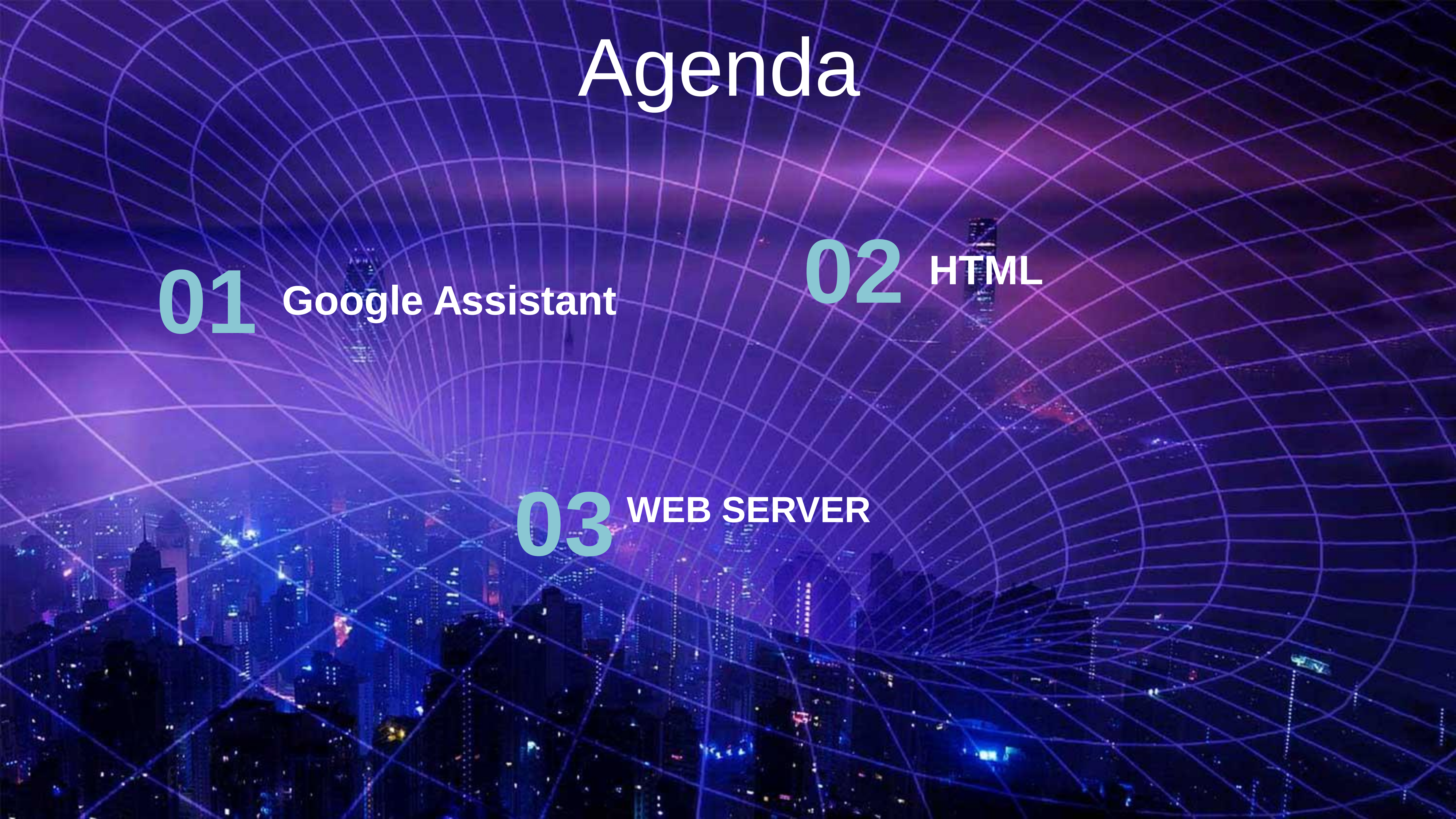
Session 7



Agenda

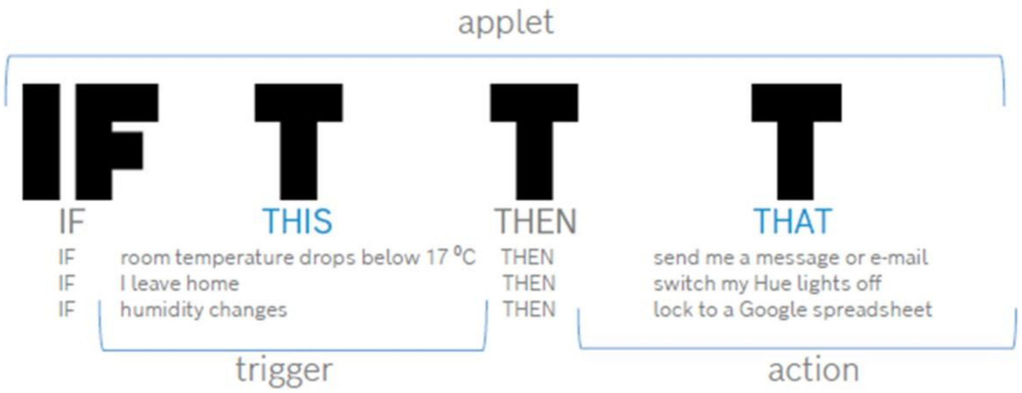
01 Google Assistant

02 HTML

03 WEB SERVER

IFTTT

- ❖ Ifttt runs services that allow a user to program a response to events in the world.
- ❖ IFTTT has partnerships with different service providers that supply event notifications to IFTTT and execute commands that implement the responses



applet

IF	THIS	THEN	THAT
IF	room temperature drops below 17 ⁰C	THEN	send me a message or e-mail
IF	I leave home	THEN	switch my Hue lights off
IF	humidity changes	THEN	lock to a Google spreadsheet

trigger

action

NodeMCU & Google Assistant

Iftttt make event connected in Way :
IF Say “SomeThing” in Google Assistant Then “Do Something”.

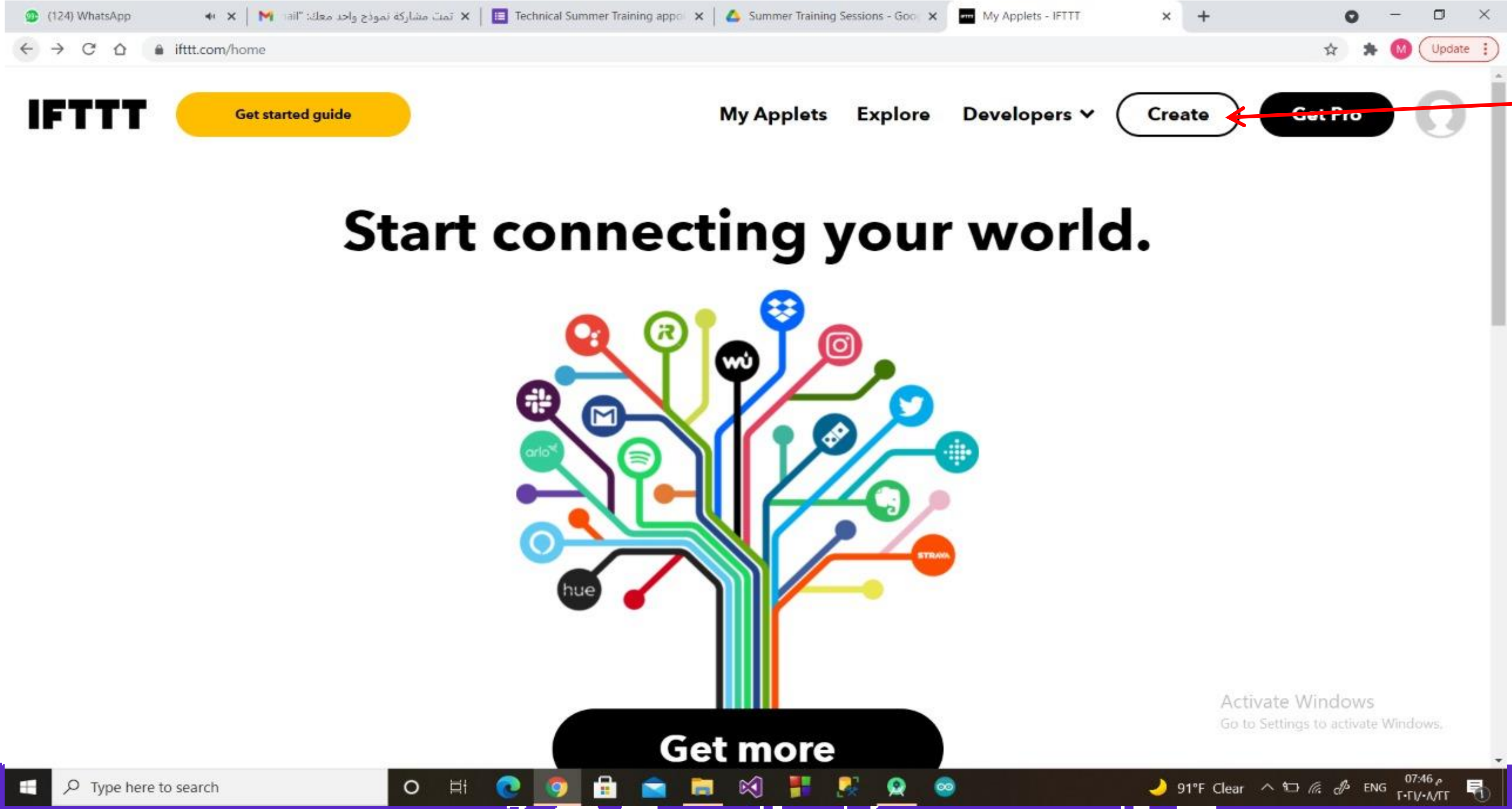
To Get Action With NodeMCU we Use ThingSpeak

Make Google Assistant communicate with ThingSpeak
then ThingSpeak communicate with NodeMCU.

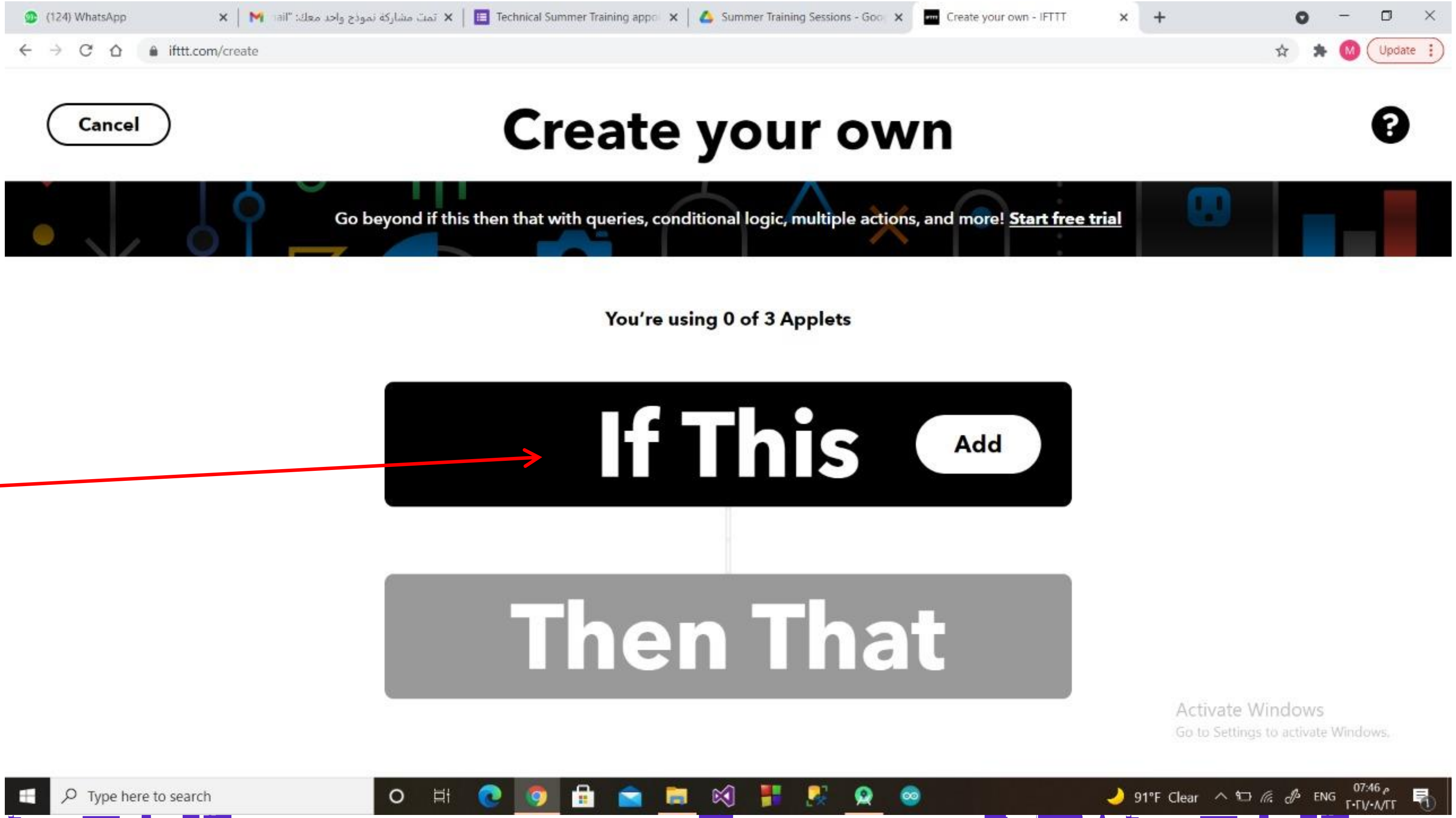
Ready To Use

We Need To make Account on ifttt.com

Then Click “Create” in Main page



Ready To Use



(124) WhatsApp
تمت مشاركة نموذج واحد معك: "ail
Technical Summer Training appo
Summer Training Sessions - Goo
Create your own - IFTTT
ifttt.com/create
Update
Cancel

Create your own

Go beyond if this then that with queries, conditional logic, multiple actions, and more! Start free trial

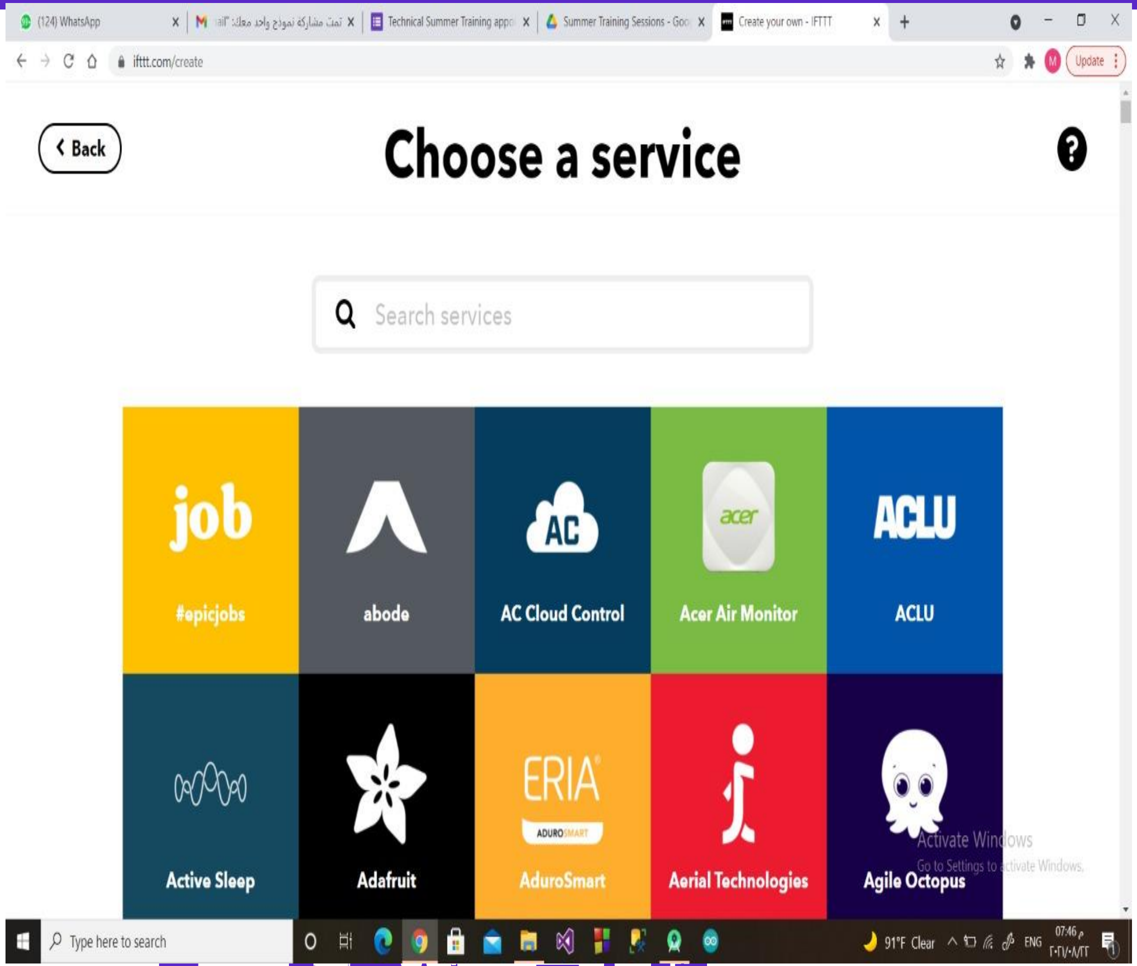
You're using 0 of 3 Applets

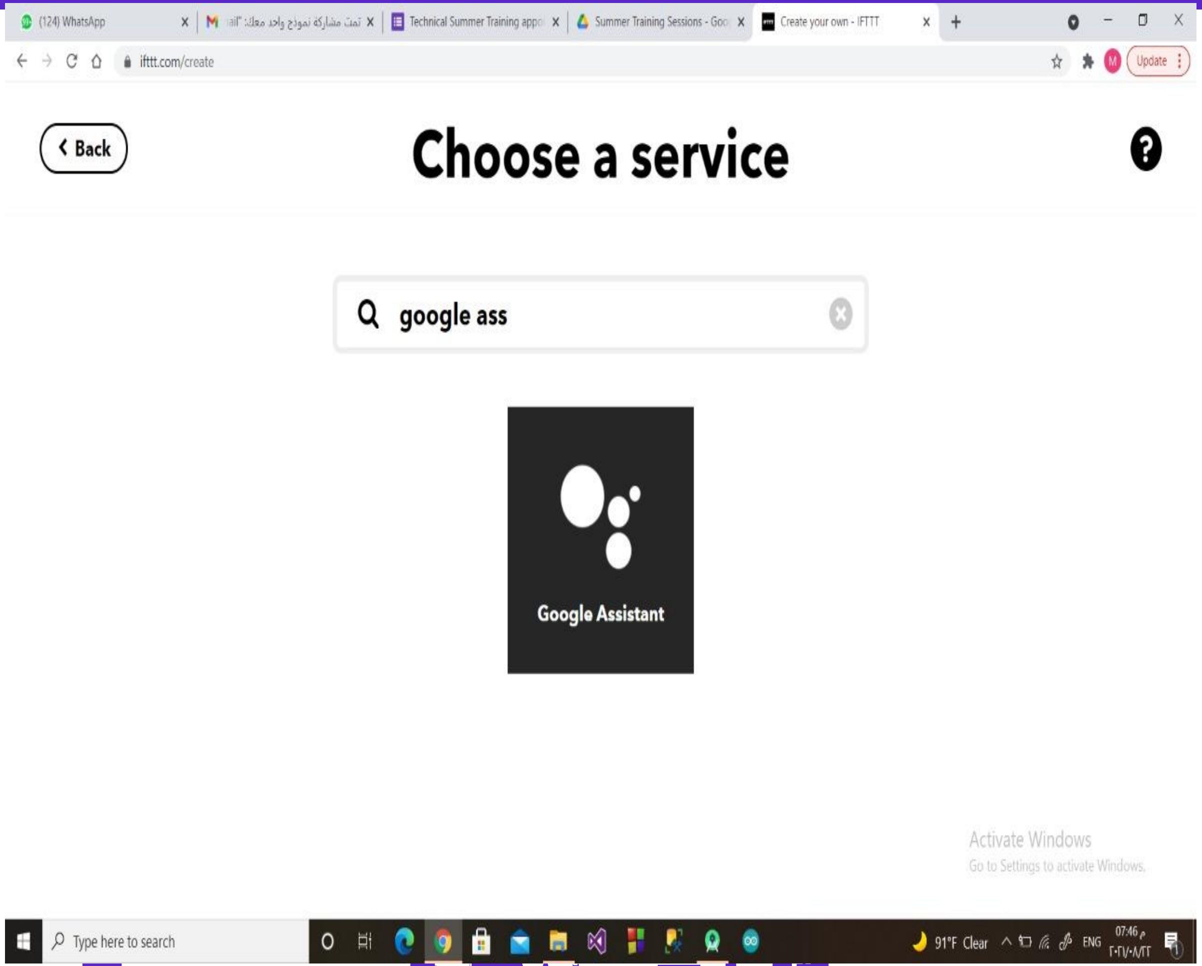
If This Add

Then That

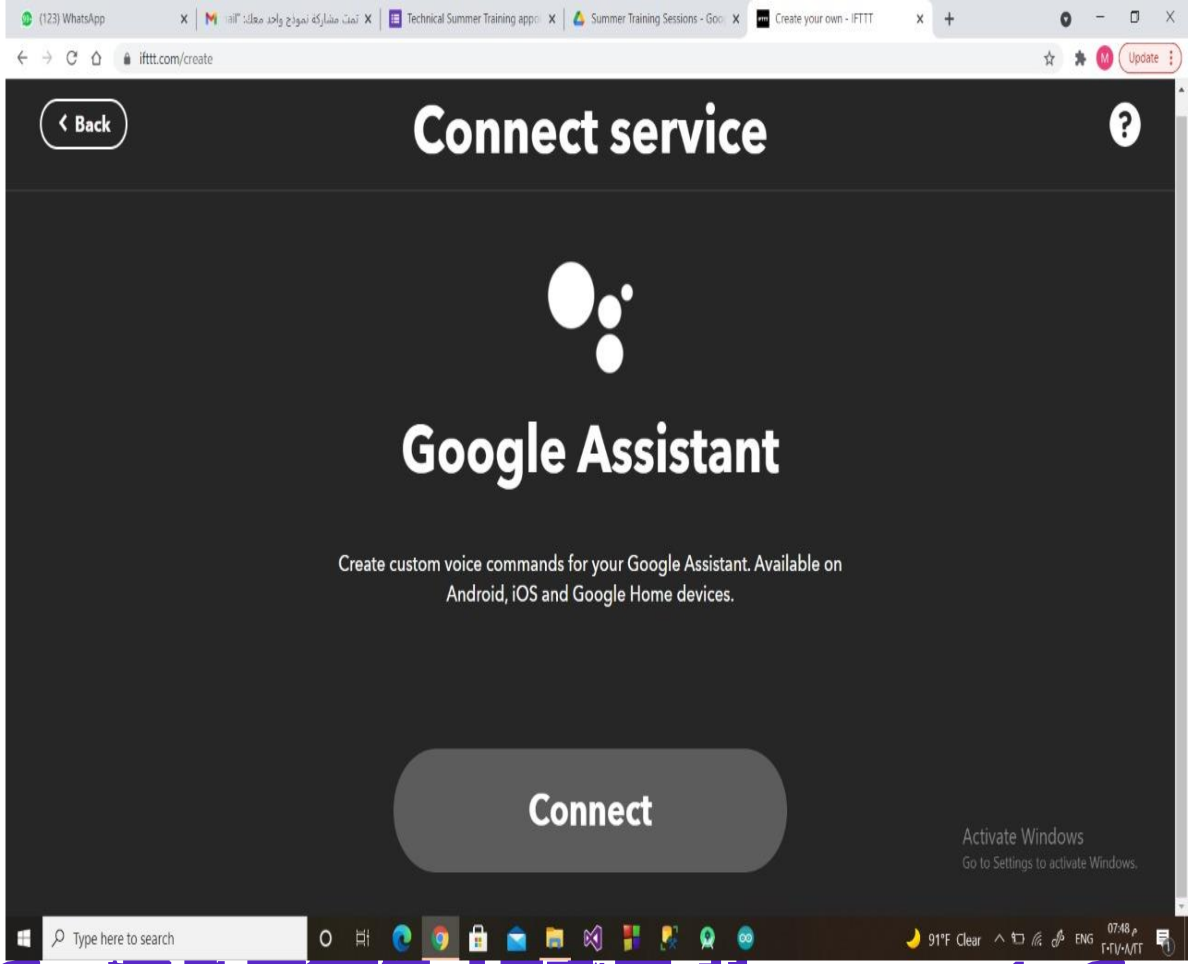
IF : what event Needed

Ready To Use


ifttt.com/create
Back
Choose a service
Search services
#epicjobs
abode
AC Cloud Control
Acer Air Monitor
ACLU
Active Sleep
Adafruit
AduroSmart
Aerial Technologies
Agile Octopus


ifttt.com/create
Back
Choose a service
google ass
Google Assistant

Ready To Use



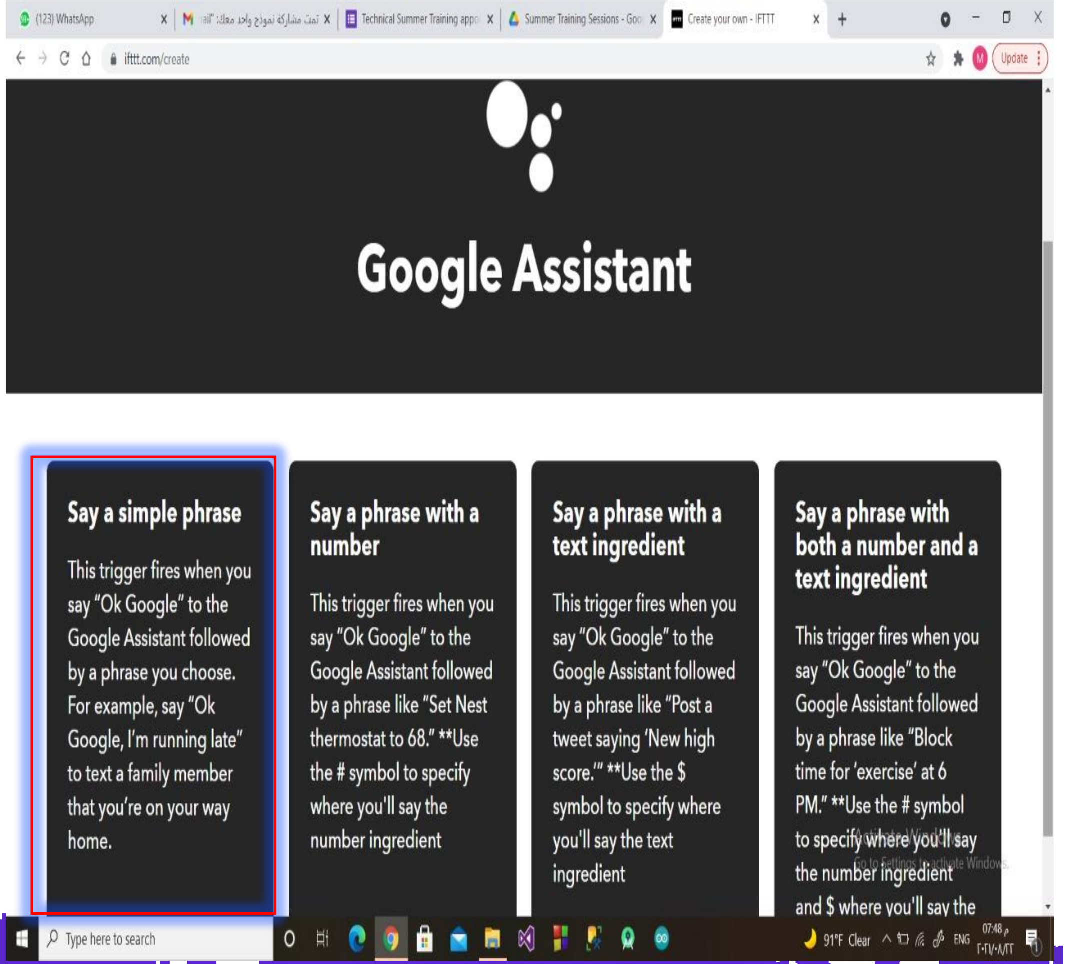
Connect service

Google Assistant

Create custom voice commands for your Google Assistant. Available on Android, iOS and Google Home devices.

Connect

Activate Windows
Go to Settings to activate Windows.



Google Assistant

Say a simple phrase

This trigger fires when you say "Ok Google" to the Google Assistant followed by a phrase you choose. For example, say "Ok Google, I'm running late" to text a family member that you're on your way home.

Say a phrase with a number

This trigger fires when you say "Ok Google" to the Google Assistant followed by a phrase like "Set Nest thermostat to 68." **Use the # symbol to specify where you'll say the number ingredient

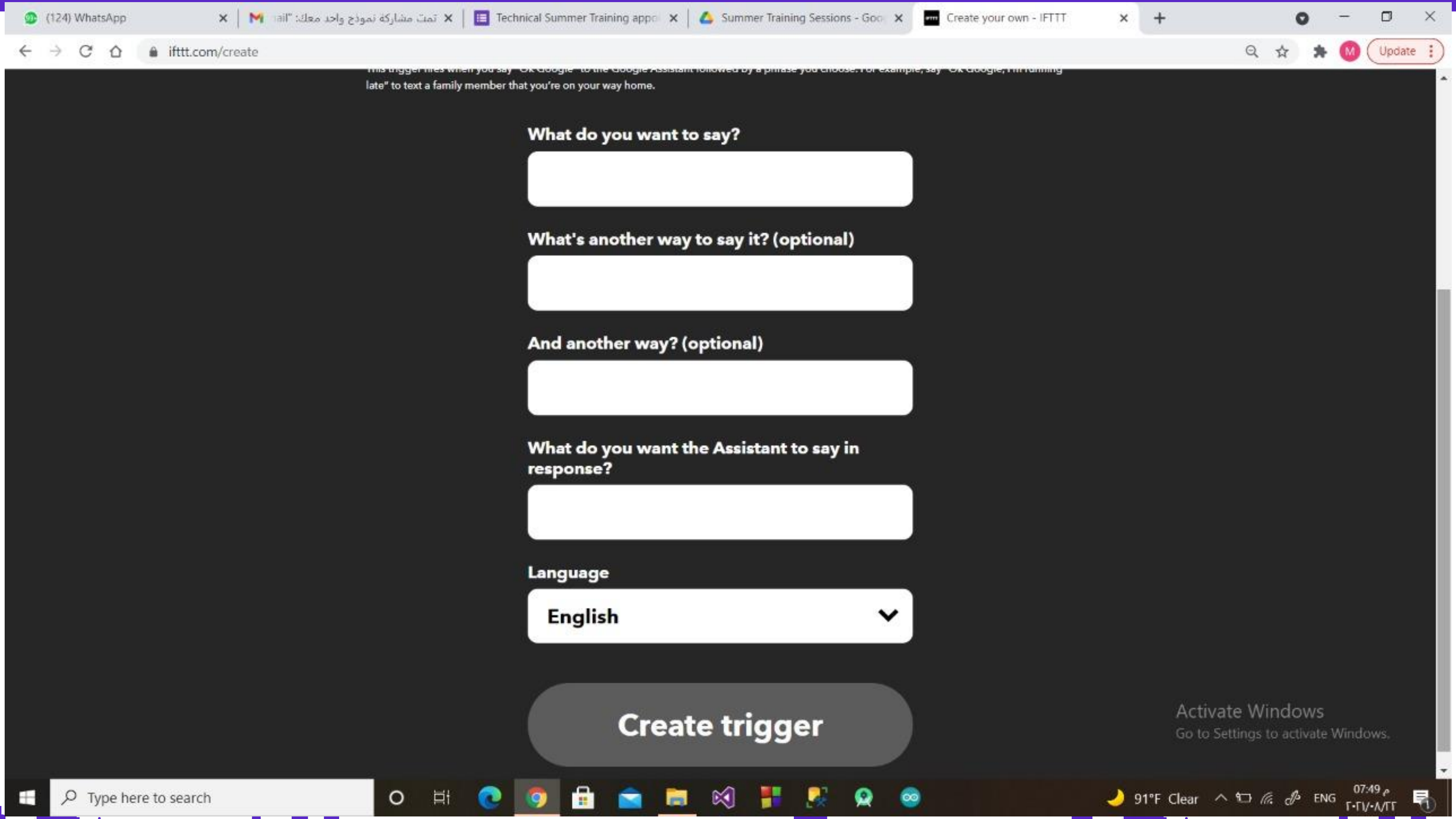
Say a phrase with a text ingredient

This trigger fires when you say "Ok Google" to the Google Assistant followed by a phrase like "Post a tweet saying 'New high score.'" **Use the $ symbol to specify where you'll say the text ingredient

Say a phrase with both a number and a text ingredient

This trigger fires when you say "Ok Google" to the Google Assistant followed by a phrase like "Block time for 'exercise' at 6 PM." **Use the # symbol to specify where you'll say the number ingredient and $ where you'll say the

Ready To Use



This trigger fires when you say "Ok Google" to the Google Assistant followed by a phrase you choose. For example, say "Ok Google, I'm running late" to text a family member that you're on your way home.

What do you want to say?

What's another way to say it? (optional)

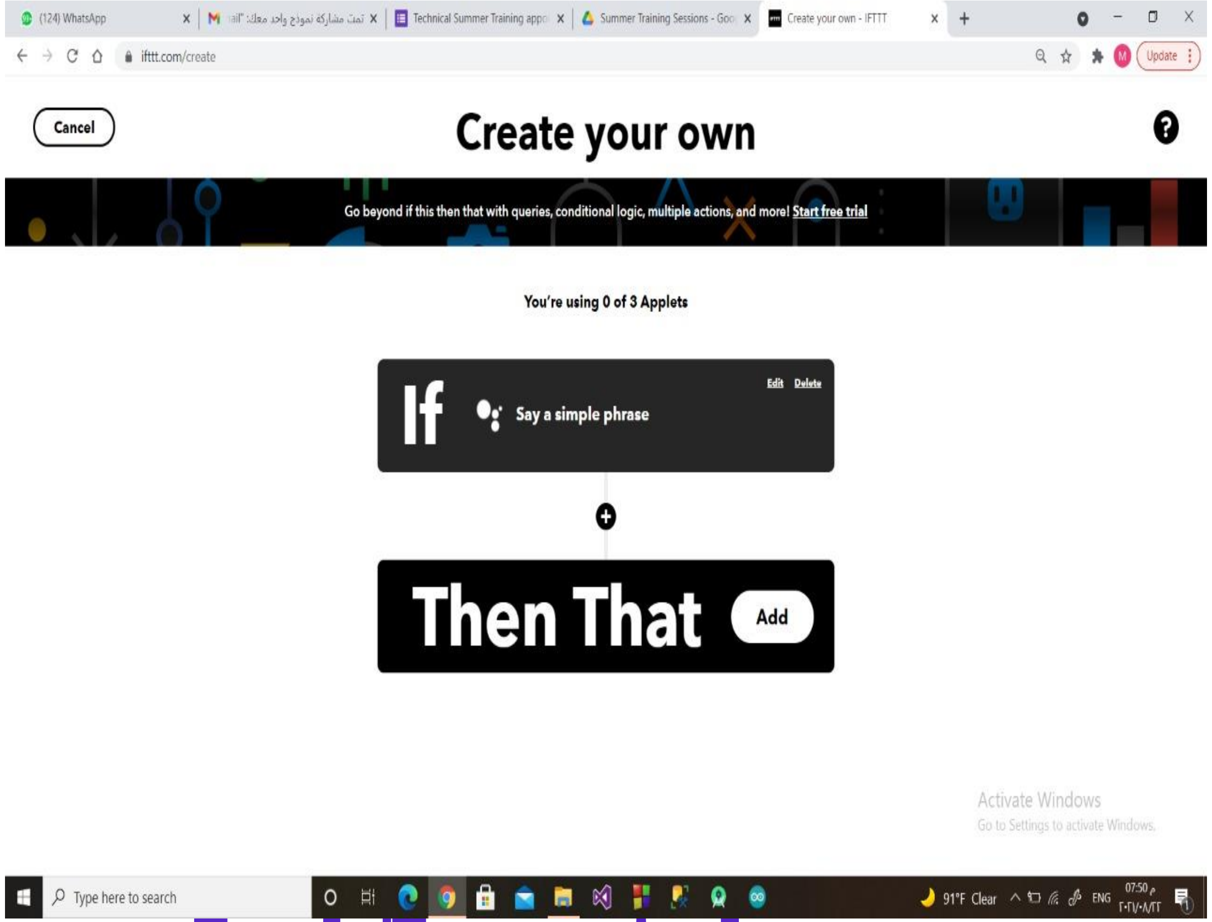
And another way? (optional)

What do you want the Assistant to say in response?

Language

English

Ready To Use



ifttt.com/create

Create your own

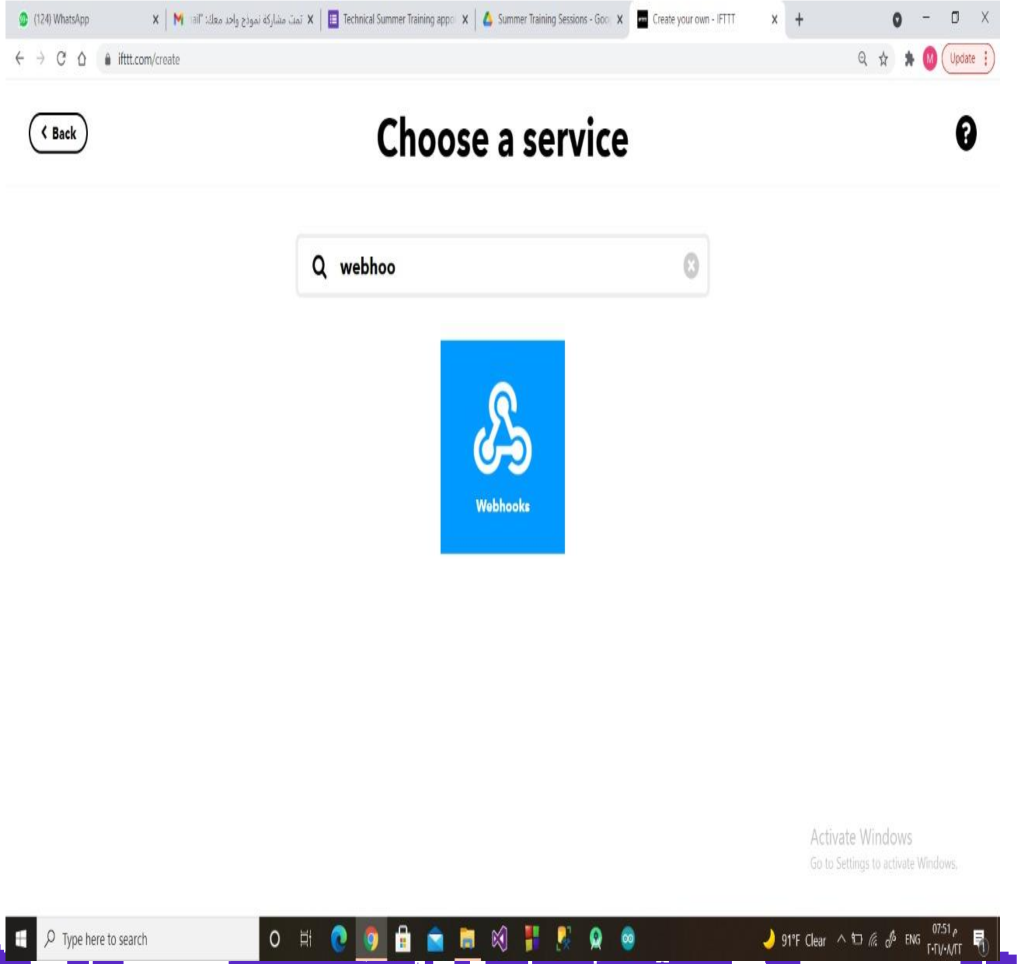
Go beyond if this then that with queries, conditional logic, multiple actions, and more! Start free trial

You're using 0 of 3 Applets

If Say a simple phrase

Then That

Activate Windows
Go to Settings to activate Windows.



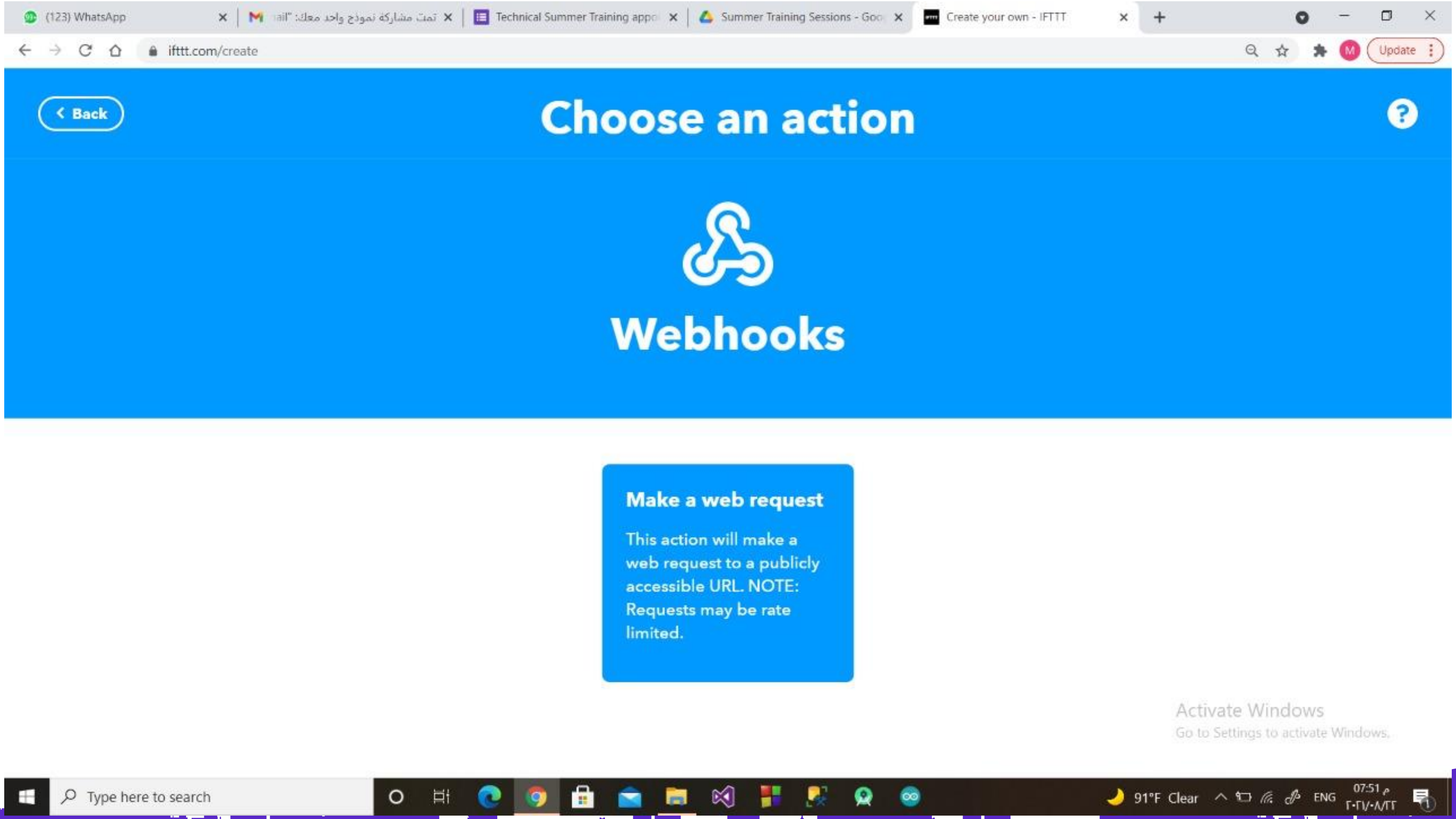
ifttt.com/create

Choose a service

webhoo

Activate Windows
Go to Settings to activate Windows.

Ready To Use



Choose an action

Webhooks

Make a web request

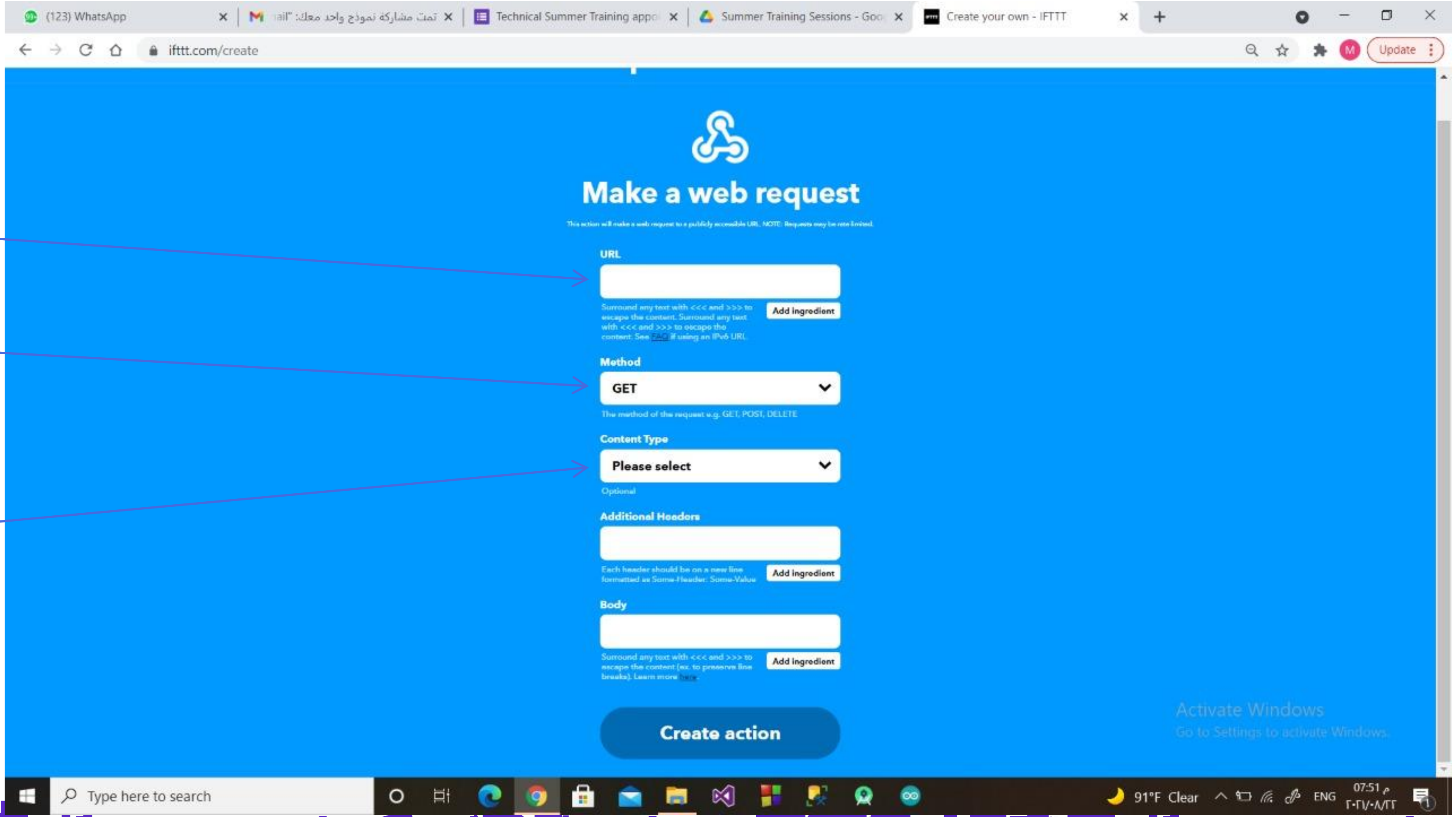
This action will make a web request to a publicly accessible URL. NOTE: Requests may be rate limited.

Ready To Use

Write API from ThingSpeak

The method we will use is GET there is others like POST

Content Type with response we will use application/json



(123) WhatsApp
تمت مشاركة نموذج واحد معك: "ail
Technical Summer Training appo
Summer Training Sessions - Goo
Create your own - IFTTT
ifttt.com/create
Update

Make a web request

This action will make a web request to a publicly accessible URL. NOTE: Requests may be rate limited.

URL

Surround any text with <<< and >>> to escape the content. Surround any text with <<< and >>> to escape the content. See FAQ if using an IPv6 URL.

Add ingredient

Method

GET

The method of the request e.g. GET, POST, DELETE

Content Type

Please select

Optional

Additional Headers

Each header should be on a new line formatted as Some-Header: Some-Value

Add ingredient

Body

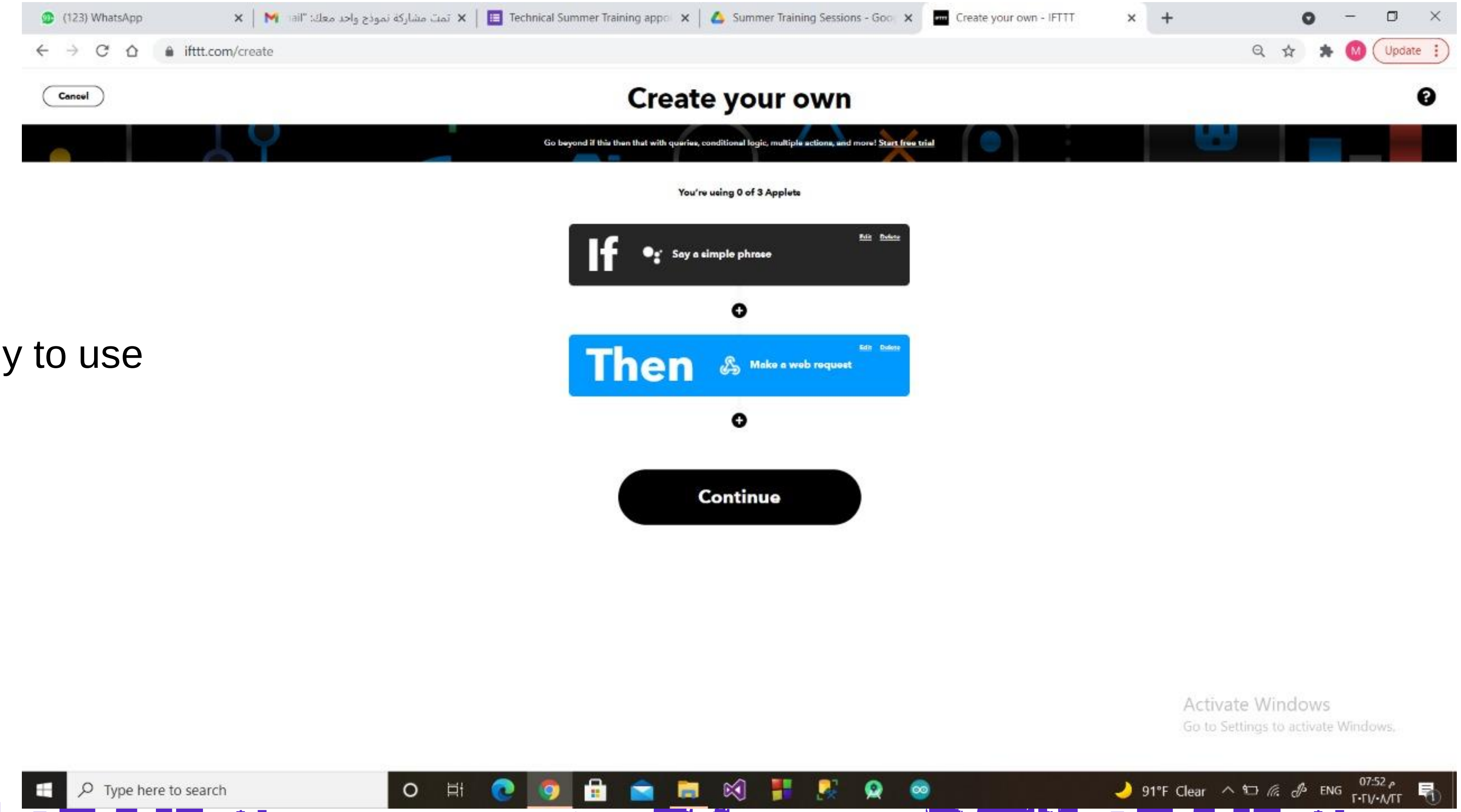
Surround any text with <<< and >>> to escape the content (ex. to preserve line breaks). Learn more here.

Add ingredient

Create action

Ready To Use

Finish
Google Assistant ready to use



Create your own

Go beyond if this then that with queries, conditional logic, multiple actions, and more! Start free trial

You're using 0 of 3 Applets

If Say a simple phrase

Then Make a web request

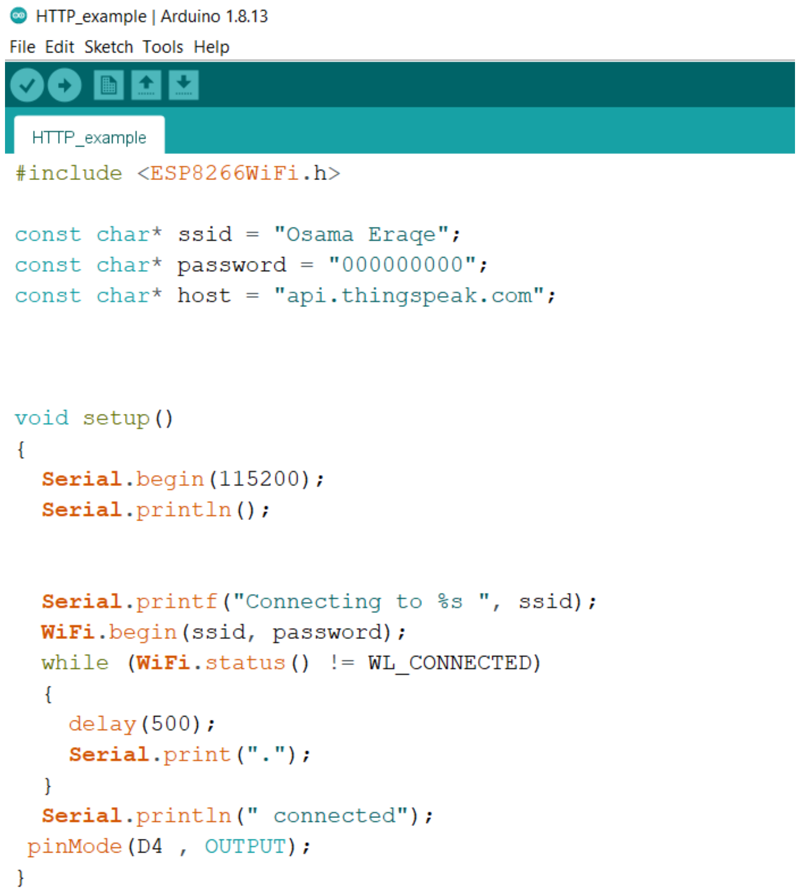
Continue

Practice

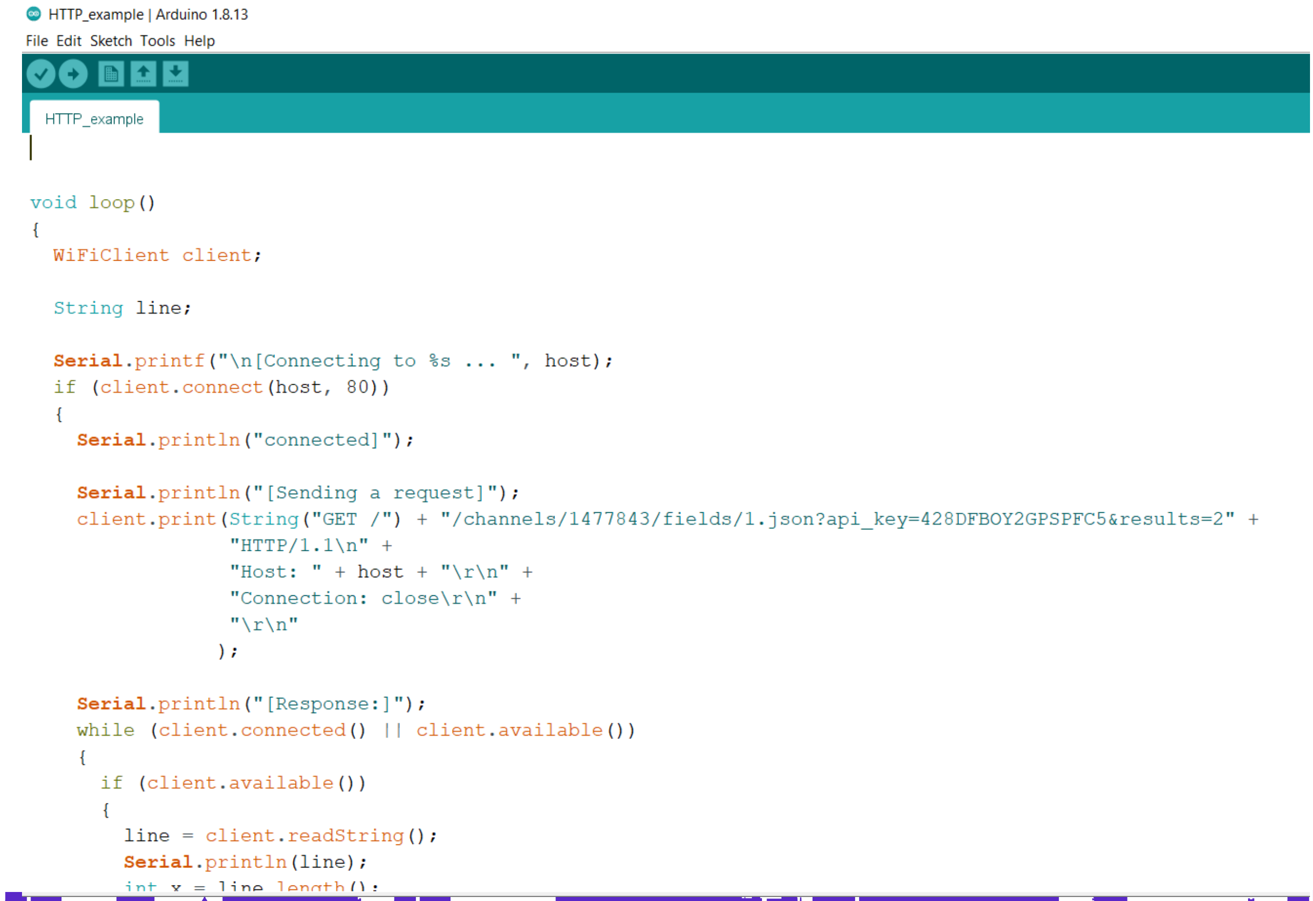
Control Led Using Google Assistant

Con. Assistant with NodeMCU

Using Same Code of Thingspeak

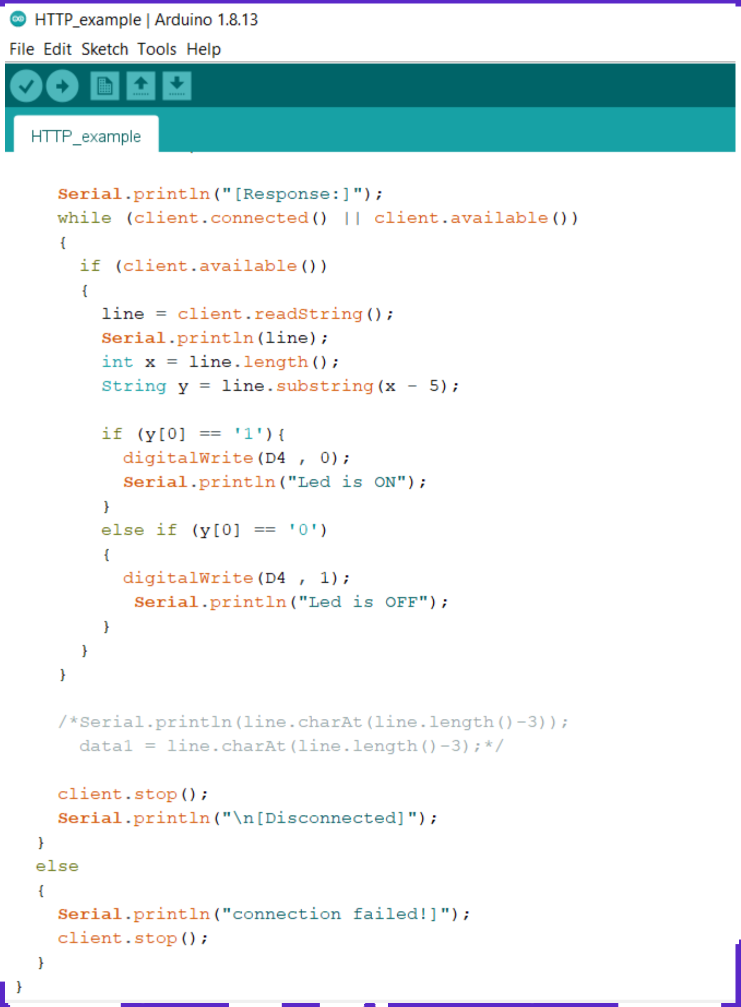


```
#include <ESP8266WiFi.h>

const char* ssid = "Osama Eraqe";
const char* password = "000000000";
const char* host = "api.thingspeak.com";



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


  Serial.printf("Connecting to %s ", ssid);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED)
  {
    delay(500);
    Serial.print(".");
  }
  Serial.println(" connected");
 pinMode(D4 , OUTPUT);
}
```



```

void loop()
{
  WiFiClient client;

  String line;

  Serial.printf("\n[Connecting to %s ... ", host);
  if (client.connect(host, 80))
  {
    Serial.println("connected]");

    Serial.println("[Sending a request]");
    client.print(String("GET /") + "/channels/1477843/fields/1.json?api_key=428DFBOY2GPSPFC5&results=2" +
                 "HTTP/1.1\n" +
                 "Host: " + host + "\r\n" +
                 "Connection: close\r\n" +
                 "\r\n"
                );

    Serial.println("[Response:]");
    while (client.connected() || client.available())
    {
      if (client.available())
      {
        line = client.readString();
        Serial.println(line);
        int x = line.length();
```

Con. Assistant with NodeMCU

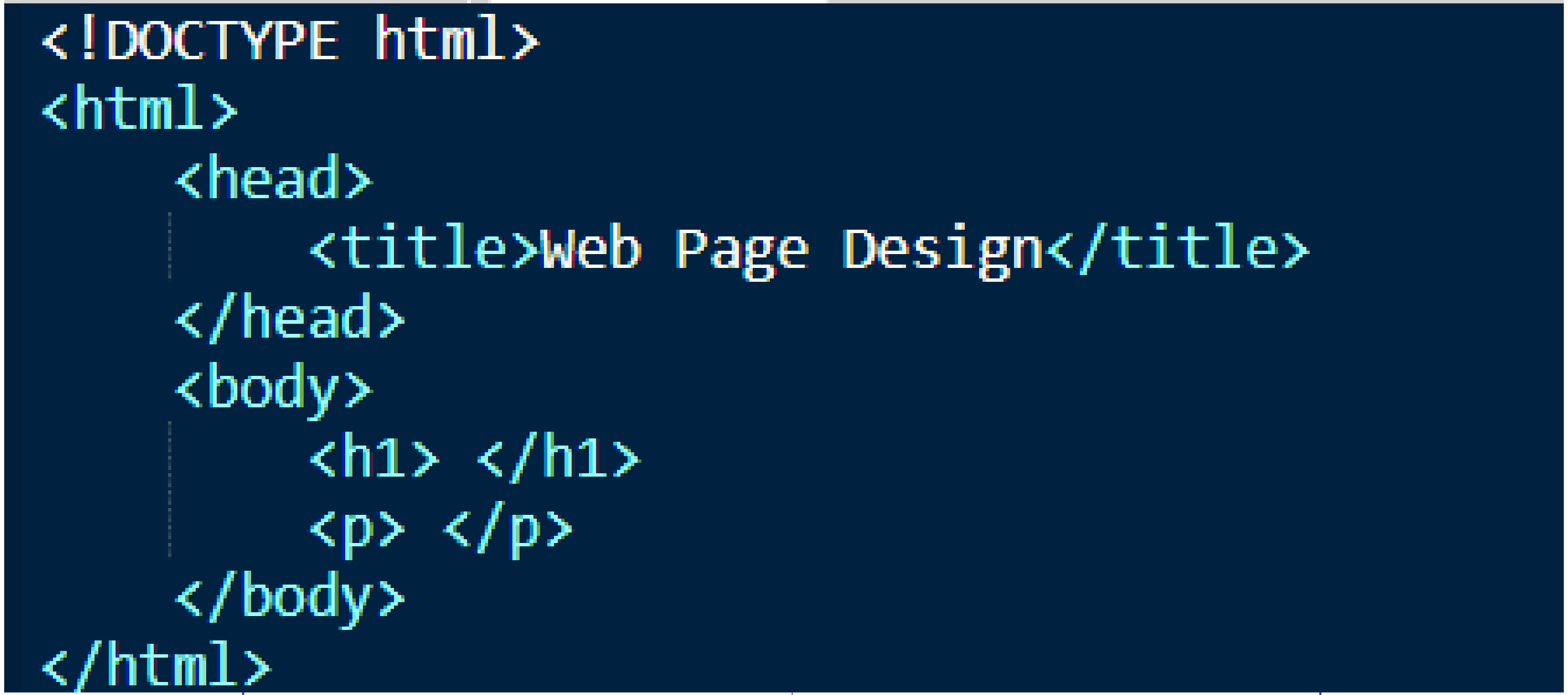


```
    Serial.println("[Response:]");
    while (client.connected() || client.available())
    {
      if (client.available())
      {
        line = client.readString();
        Serial.println(line);
        int x = line.length();
        String y = line.substring(x - 5);

        if (y[0] == '1'){
          digitalWrite(D4 , 0);
          Serial.println("Led is ON");
        }
        else if (y[0] == '0')
        {
          digitalWrite(D4 , 1);
           Serial.println("Led is OFF");
        }
      }
    }

    /*Serial.println(line.charAt(line.length()-3));
      data1 = line.charAt(line.length()-3);*/

    client.stop();
    Serial.println("\n[Disconnected]");
  }
  else
  {
    Serial.println("connection failed!]");
    client.stop();
  }
}
```

HTML

- **Hyper Text Markup Language**, or **HTML** stands for Hyper Text Markup Language
- HTML is the standard markup language for creating Web pages
- HTML describes the structure of a Web page
- HTML consists of a series of elements and the elements tell the browser how to display the content
- HTML elements label pieces of content such as "this is a heading", "this is a paragraph", "this is a link", etc.

Con. HTML



```
<!DOCTYPE html>
<html>
    <head>
        <title>Web Page Design</title>
    </head>
    <body>
        <h1> </h1>
        <p> </p>
    </body>
</html>
```

WWW Overview

- **WWW** stands for **World Wide Web.** A technical definition of the World Web is : all the resources and users on the Internet that are using the Hypertext Transfer Protocol (HTTP).

- A broader definition comes from the organization that Web inventor **Tim Berners-Lee** helped found, the **World Wide Web Consortium (W3C).**

- The World Wide Web is a way of exchanging information between computers on the Internet, tying them together into a vast collection of interactive multimedia resources.

TIP

Internet and Web is not the same thing: Web uses internet to pass over the information.

Web Page Overview

- **web page** is a document available on world wide web. Web Pages on web server and can be viewed using a web browser.

- A web page can cotain huge information including text, graphics, audio, and hyper links. These hyper links are the link to other web pages.

- **Static Web Page** contain only static information. User can only read the information but can't do any modification or interact with the information.
- **Dynamic Web Page** is possible to change a portation of a web page without loading the entire web page.

Collection of linked web pages on a web server is known as **website.** There is unique **Uniform Resource Locator (URL)** is associated with each web page

Server

- **Server :**

A server is **a computer or system that provides resources, data, services, or programs to other computers**

- The other devices are called hosts or clients
- The server can provide different services to the clients and It can also share resources between multiple clients or perform computations for a client. Moreover, the server is capable of providing services to multiple clients simultaneously
- client can obtain services from multiple servers. It is possible for a server and client to be in the same device or in separate devices

Con.Server

- **Types of Servers :**
    - **Print Server** – Connect printers to the clients in the network
    - **Database Server** – Contain a database application to provide database functionalities to client applications
    - **Web servers–** Provide the required web pages to the clients
    - **File Servers** – Provide the required files to the clients
    - **Game Servers** – Contain authoritative source of events in a multiplayer video game
    - **Mail servers** – Allows sending and receiving emails.

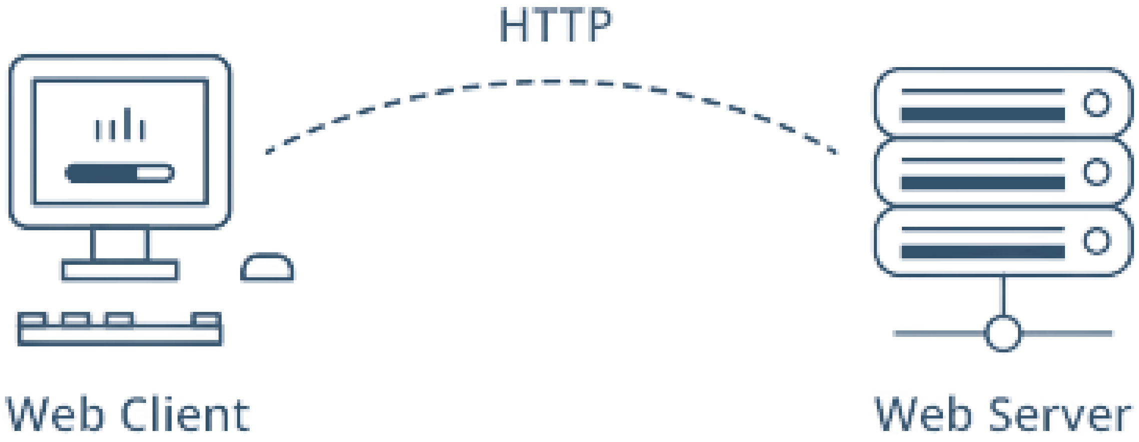
WebServer

- **web server** is a software or hardware that provides web content via the World Wide Web (WWW)

- It stores, process and provides web pages to the clients. Moreover, the client and the web server communicate via Hypertext Transfer Protocol (HTTP).

Con. WebServer

- when client sends request for a web page, the web server search for the requested page if requested page is found then it will send it to client with an HTTP response.
- If the requested web page is not found, web server will the send an **HTTP response:Error 404 Not found.**
- If client has requested for some other resources then the web server will contact to the application server and data store to construct the HTTP response



Con. WebServer

- Web Server Examples

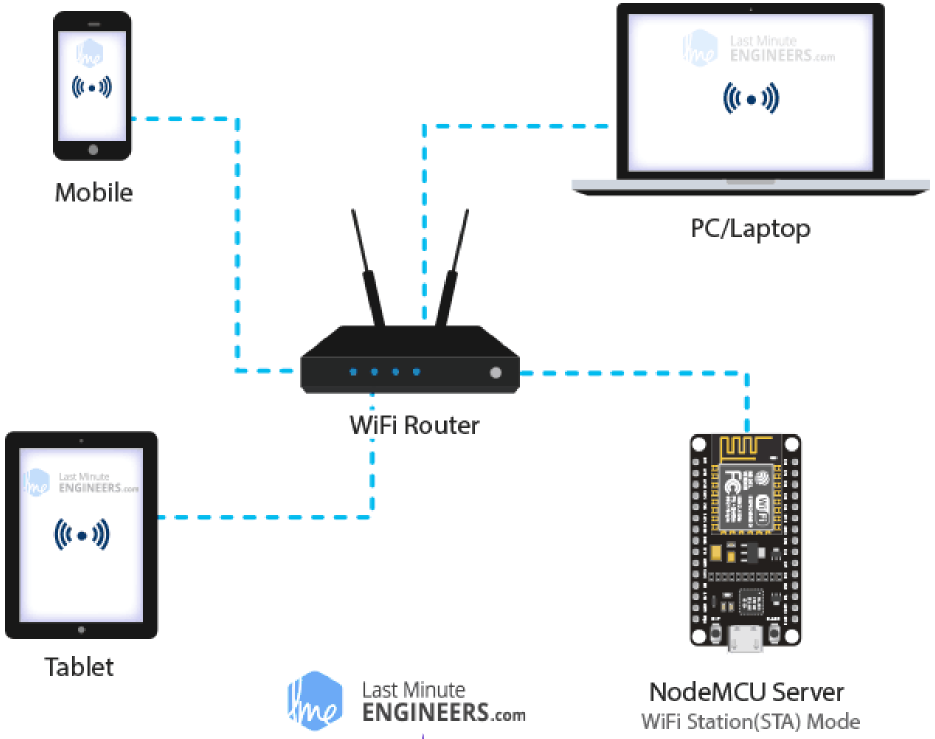
- **Apache HTTP Server**
  This is the most popular web server in the world developed by the Apache Software Foundation. Apache web server is an open source software and can be installed on almost all operating systems including Linux, UNIX, Windows, FreeBSD, Mac OS X and more. About 60% of the web server machines run the Apache Web Server.
- **Internet Information Services (IIS)**
  The Internet Information Server (IIS) is a high performance Web Server from Microsoft. This web server runs on Windows NT/2000 and 2003 platforms.

WebServer With NodeMCU

- One of the greatest features ESP8266 provides is that it cannot only connect to an existing WiFi network and act as a Web Server
- it can also set up a network of its own, allowing other devices to connect directly to it and access web pages
- This is possible because ESP8266 can operate in three different modes: Station mode, Soft Access Point mode, and both at the same time.

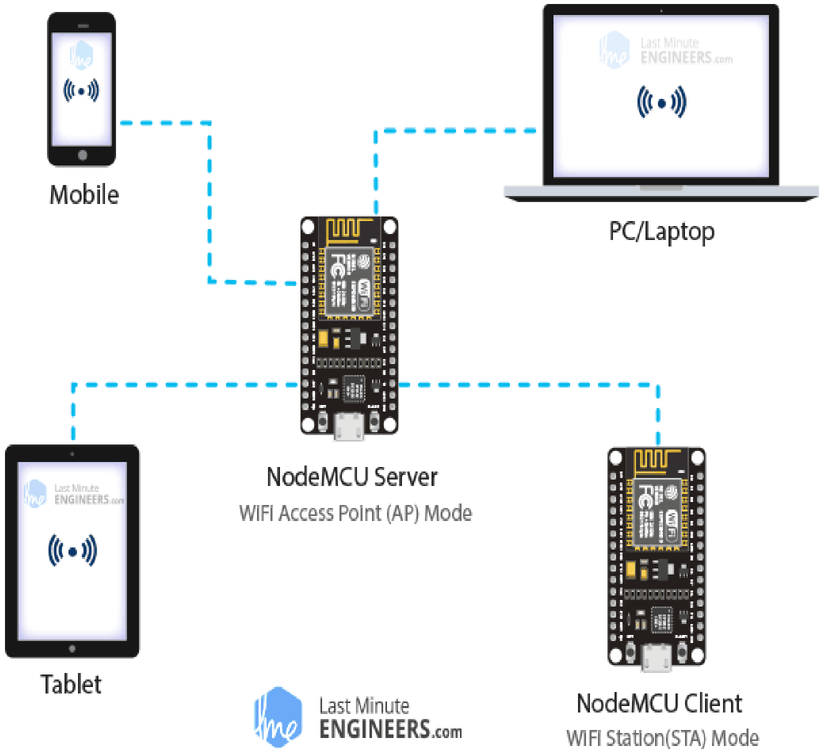
ESP8266 Operating Modes

- Station (STA) Mode
  - The ESP8266 that connects to an existing WiFi network (one created by your wireless router) is called Station (STA)
  - In STA mode ESP8266 gets IP from wireless router to which it is connected. With this IP address, it can set up a web server and deliver web pages to all connected devices under existing WiFi network.



Con. ESP8266 Operating Modes

- Soft Access Point (AP) Mode
  - The ESP8266 that creates its own WiFi network and acts as a hub (Just like WiFi router) for one or more stations is called Access Point (AP).
  - In AP mode ESP8266 creates a new WiFi network and sets SSID (Name of the network) and IP address to it. With this IP address, it can deliver web pages to all connected devices under its own network.



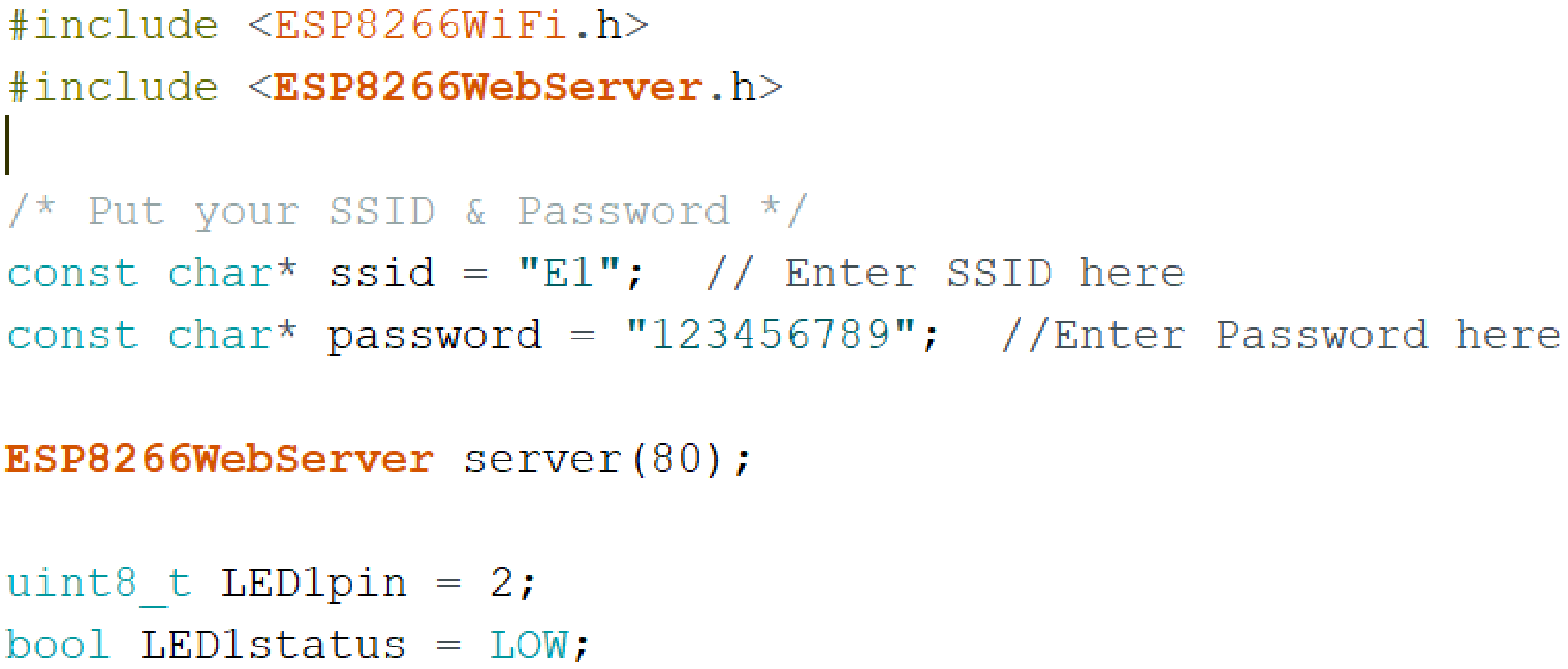
Turn NodeMCU to Acess point

Make HTML Page

Display the state of led

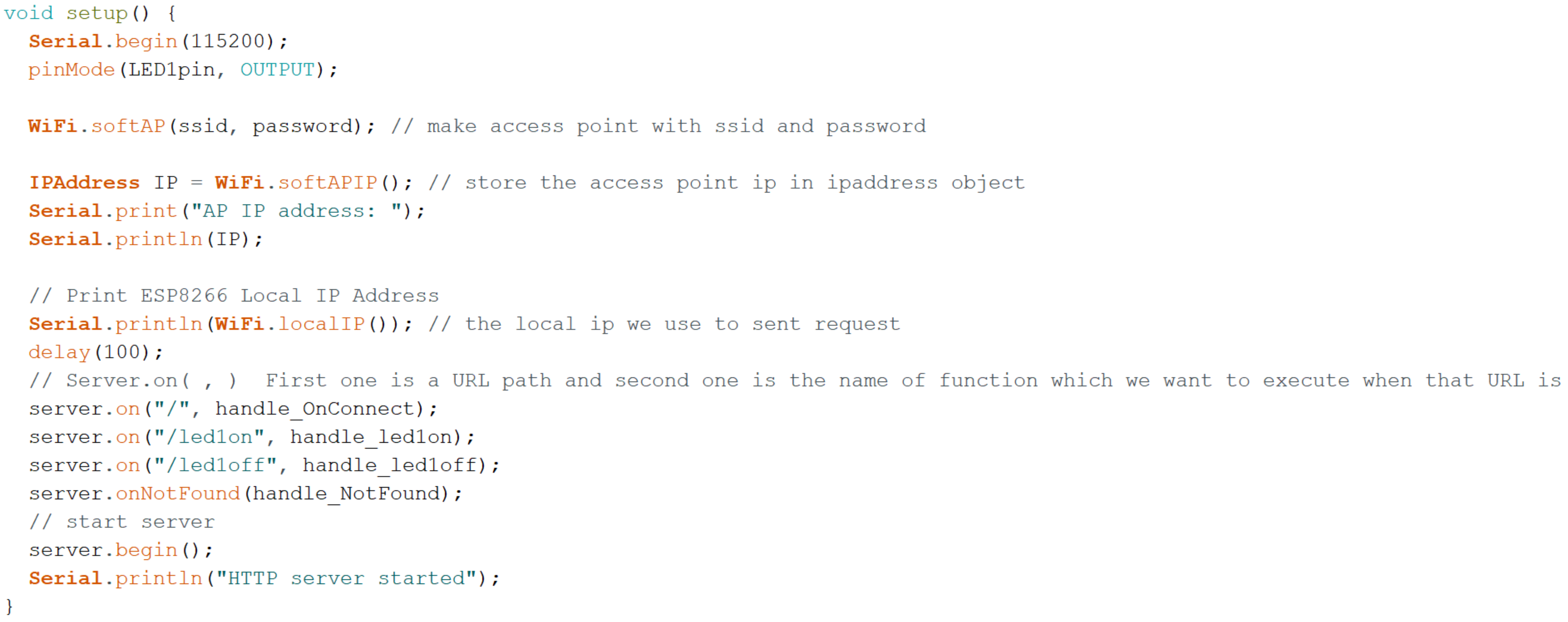
control led using button in this page

Code



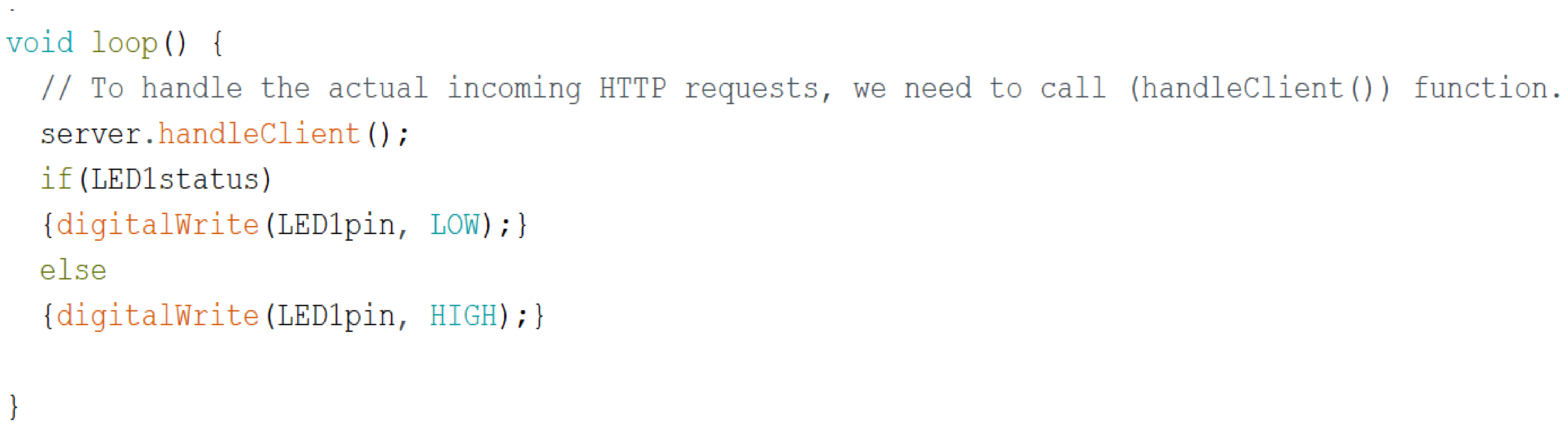
```
#include <ESP8266WiFi.h>
#include <ESP8266WebServer.h>

/* Put your SSID & Password */
const char* ssid = "E1";  // Enter SSID here
const char* password = "123456789";  //Enter Password here

ESP8266WebServer server(80);

uint8_t LED1pin = 2;
bool LED1status = LOW;
```

Code



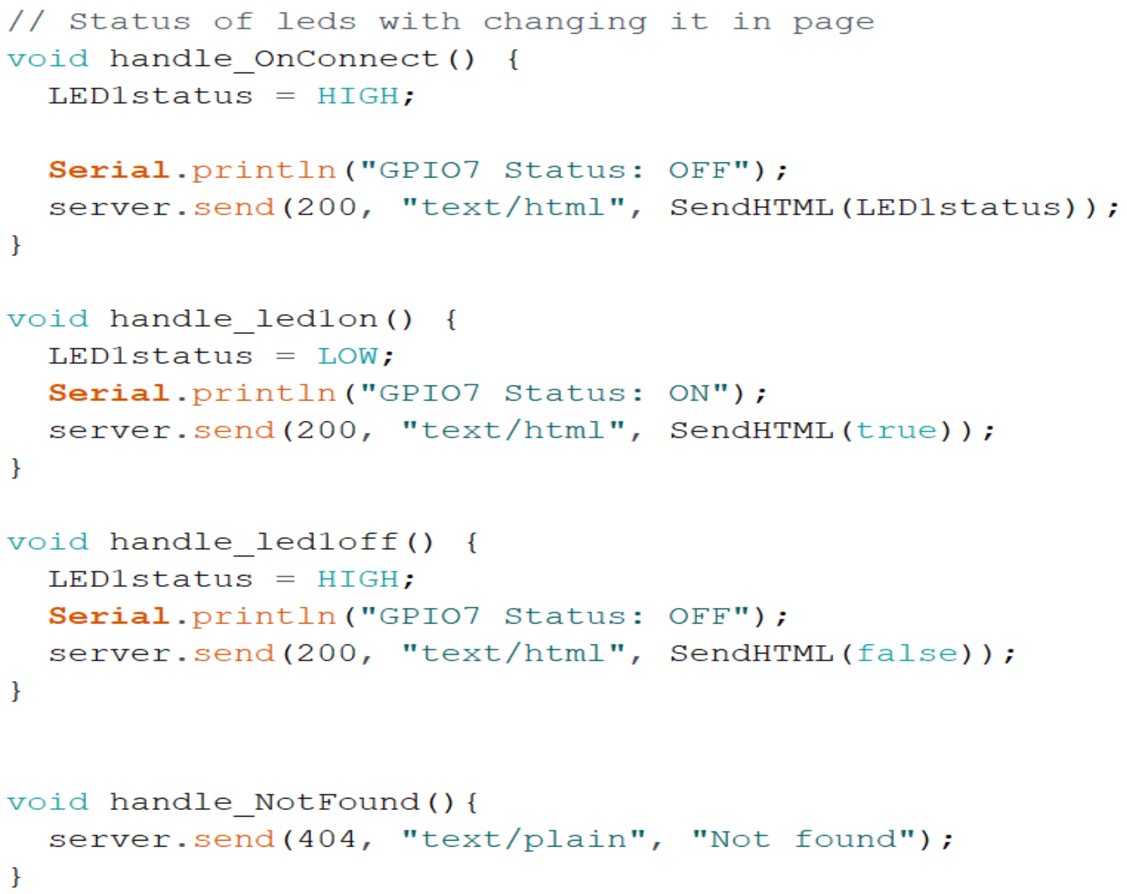
```
void setup() {
  Serial.begin(115200);
  pinMode(LED1pin, OUTPUT);

  WiFi.softAP(ssid, password); // make access point with ssid and password

  IPAddress IP = WiFi.softAPIP(); // store the access point ip in ipaddress object
  Serial.print("AP IP address: ");
  Serial.println(IP);

  // Print ESP8266 Local IP Address
  Serial.println(WiFi.localIP()); // the local ip we use to sent request
  delay(100);
  // Server.on( , )  First one is a URL path and second one is the name of function which we want to execute when that URL is
  server.on("/", handle_OnConnect);
  server.on("/led1on", handle_led1on);
  server.on("/led1off", handle_led1off);
  server.onNotFound(handle_NotFound);
  // start server
  server.begin();
  Serial.println("HTTP server started");
}
```

Code



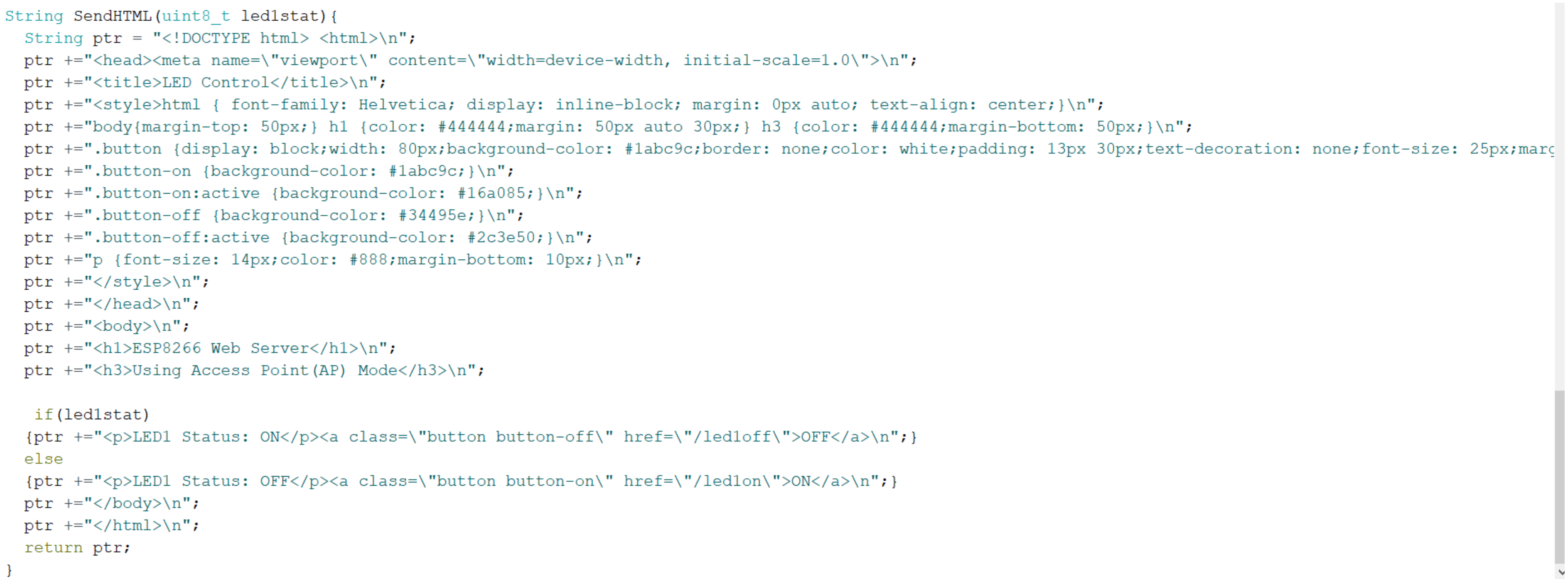
```
void loop() {
  // To handle the actual incoming HTTP requests, we need to call (handleClient()) function.
  server.handleClient();
  if(LED1status)
  {digitalWrite(LED1pin, LOW);}
  else
  {digitalWrite(LED1pin, HIGH);}

}
```

Code



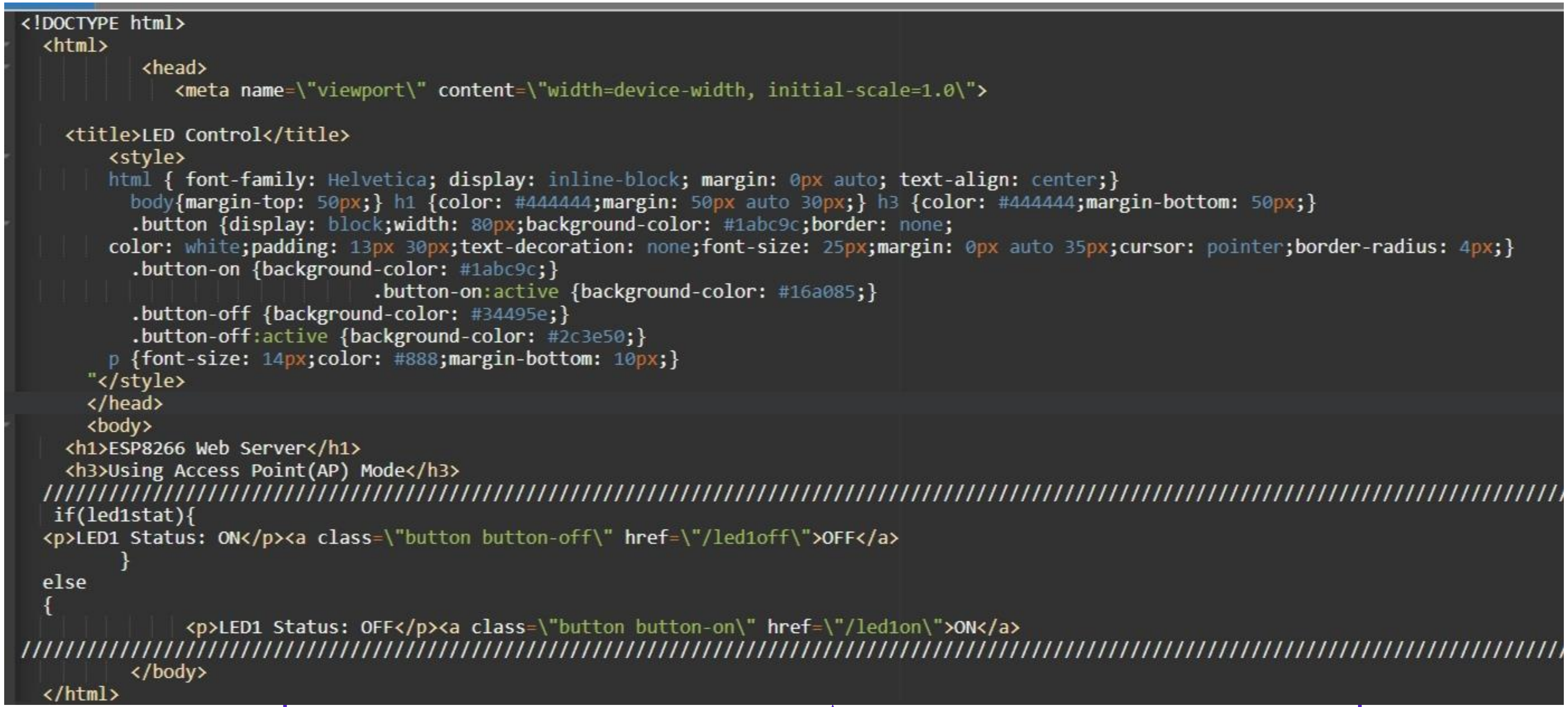
```
// Status of leds with changing it in page
void handle_OnConnect() {
  LED1status = HIGH;

  Serial.println("GPIO7 Status: OFF");
  server.send(200, "text/html", SendHTML(LED1status));
}

void handle_led1on() {
  LED1status = LOW;
  Serial.println("GPIO7 Status: ON");
  server.send(200, "text/html", SendHTML(true));
}

void handle_led1off() {
  LED1status = HIGH;
  Serial.println("GPIO7 Status: OFF");
  server.send(200, "text/html", SendHTML(false));
}


void handle_NotFound(){
  server.send(404, "text/plain", "Not found");
}
```

Code

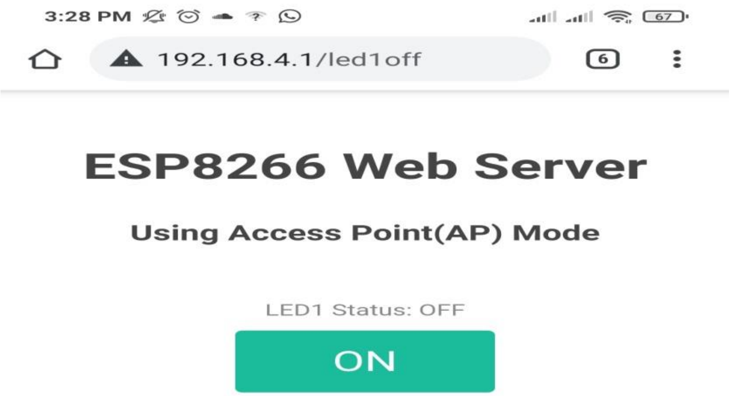


```
String SendHTML(uint8_t led1stat){
  String ptr = "<!DOCTYPE html> <html>\n";
  ptr +="<head><meta name=\"viewport\" content=\"width=device-width, initial-scale=1.0\">\n";
  ptr +="<title>LED Control</title>\n";
  ptr +="<style>html { font-family: Helvetica; display: inline-block; margin: 0px auto; text-align: center;}\n";
  ptr +="body{margin-top: 50px;} h1 {color: #444444;margin: 50px auto 30px;} h3 {color: #444444;margin-bottom: 50px;}\n";
  ptr +=".button {display: block;width: 80px;background-color: #1abc9c;border: none;color: white;padding: 13px 30px;text-decoration: none;font-size: 25px;marg
  ptr +=".button-on {background-color: #1abc9c;}\n";
  ptr +=".button-on:active {background-color: #16a085;}\n";
  ptr +=".button-off {background-color: #34495e;}\n";
  ptr +=".button-off:active {background-color: #2c3e50;}\n";
  ptr +="p {font-size: 14px;color: #888;margin-bottom: 10px;}\n";
  ptr +="</style>\n";
  ptr +="</head>\n";
  ptr +="<body>\n";
  ptr +="<h1>ESP8266 Web Server</h1>\n";
  ptr +="<h3>Using Access Point(AP) Mode</h3>\n";

   if(led1stat)
  {ptr +="<p>LED1 Status: ON</p><a class=\"button button-off\" href=\"/led1off\">OFF</a>\n";}
  else
  {ptr +="<p>LED1 Status: OFF</p><a class=\"button button-on\" href=\"/led1on\">ON</a>\n";}
  ptr +="</body>\n";
  ptr +="</html>\n";
  return ptr;
}
```

Code



```
<!DOCTYPE html>
  <html>
           <head>
              <meta name=\"viewport\" content=\"width=device-width, initial-scale=1.0\">

    <title>LED Control</title>
        <style>
        html { font-family: Helvetica; display: inline-block; margin: 0px auto; text-align: center;}
          body{margin-top: 50px;} h1 {color: #444444;margin: 50px auto 30px;} h3 {color: #444444;margin-bottom: 50px;}
          .button {display: block;width: 80px;background-color: #1abc9c;border: none;
        color: white;padding: 13px 30px;text-decoration: none;font-size: 25px;margin: 0px auto 35px;cursor: pointer;border-radius: 4px;}
          .button-on {background-color: #1abc9c;}
                                .button-on:active {background-color: #16a085;}
          .button-off {background-color: #34495e;}
          .button-off:active {background-color: #2c3e50;}
        p {font-size: 14px;color: #888;margin-bottom: 10px;}
      "</style>
      </head>
      <body>
    <h1>ESP8266 Web Server</h1>
    <h3>Using Access Point(AP) Mode</h3>
  ///////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
   if(led1stat){
  <p>LED1 Status: ON</p><a class=\"button button-off\" href=\"/led1off\">OFF</a>
         }
  else
  {
               <p>LED1 Status: OFF</p><a class=\"button button-on\" href=\"/led1on\">ON</a>
/////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
          </body>
  </html>
```



ESP8266 Web Server

Using Access Point(AP) Mode

LED1 Status: OFF

The End

THANK YOU