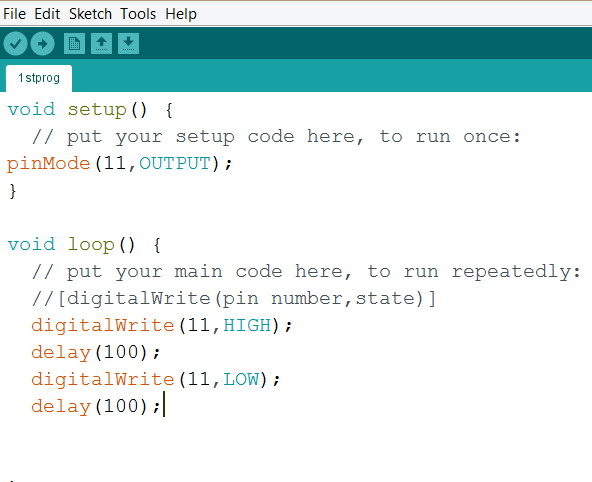
**ARDUINO PROGRAMMING TUTORIAL.**

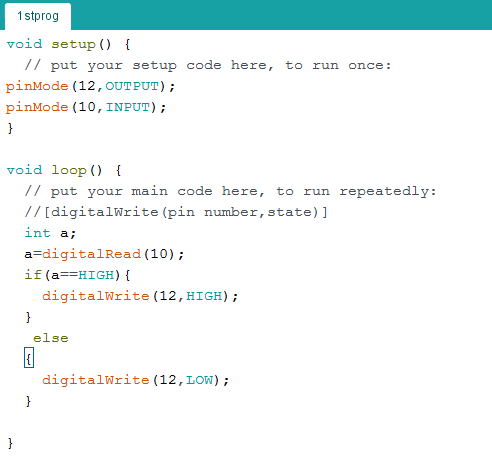
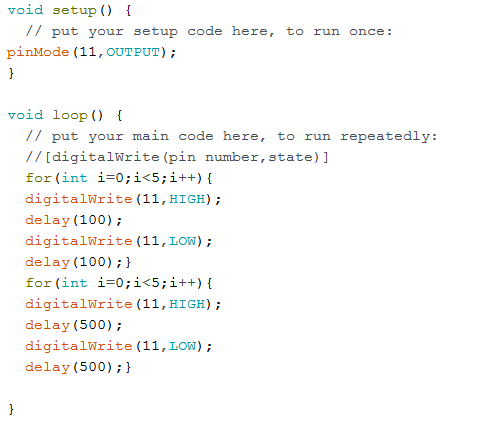
**DAY 1**

**[analog and digital]**

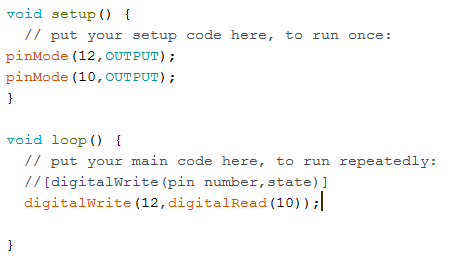
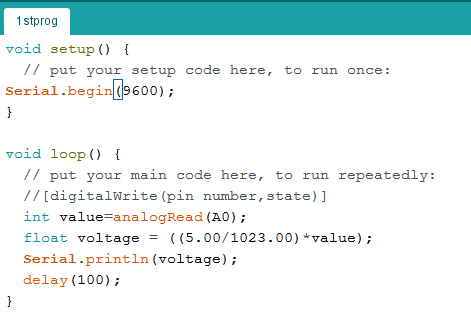
**Q1.first code......led light high and low. [here 11 is the +ve o/p and GND –ve ]**



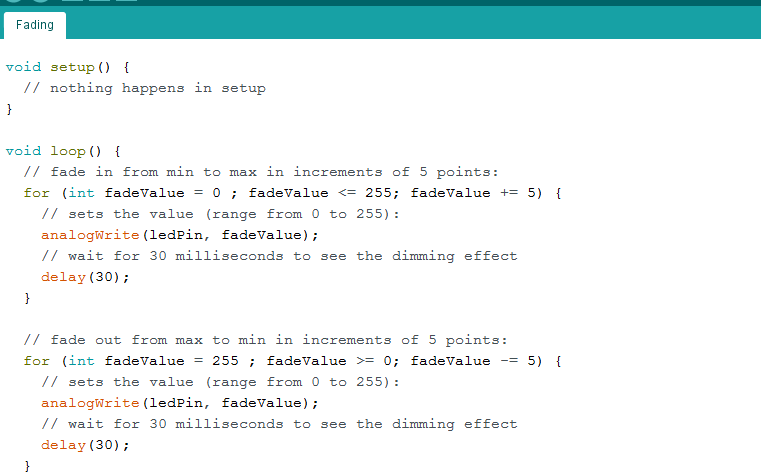
**Link for downloading node mcu driver...**



**Using for loop(digitalWrite(12,HIGH)) using if statement(digitalRead(A0))[here 12 is the +ve o/p and GND –ve i/p and 10 is i/p]**

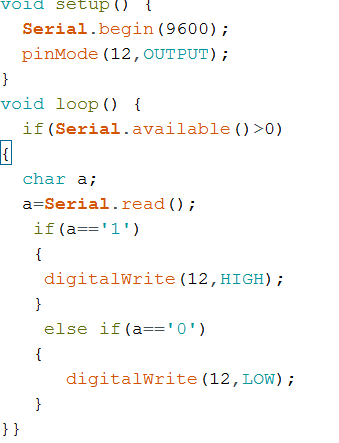
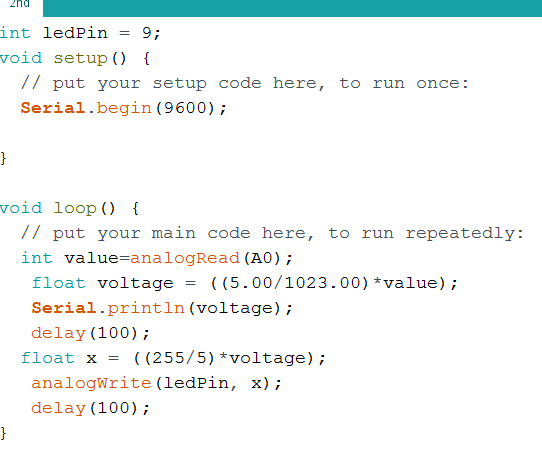
Analogtodigital( ananlogRead(A0))



Analogtodigital( ananlogWrite(,)).fadding

**DAY 2**

**[Serial communication]**



**Q1.The led o/p will be given accordingly the voltage has been given as a i/p through analogRead(A0).**

**Q2(bt).using bt app the input is given 1 or 0.if i/p will 1 then led on else i/p 0 then led off. Here the o/p pin is 12 in arduino.**

**#include <SoftwareSerial.h>**

**SoftwareSerial swSer(14, 12, 128); // SoftwareSerial sayan(10,11);// Rx , Tx**

**void setup() {**

**Serial.begin(115200); swSer.begin(115200);**

**Serial.println("\nSoftware serial test started");**

**for (char ch = ' '; ch <= 'z'; ch++) {**

**swSer.write(ch); }**

**swSer.println("");}**

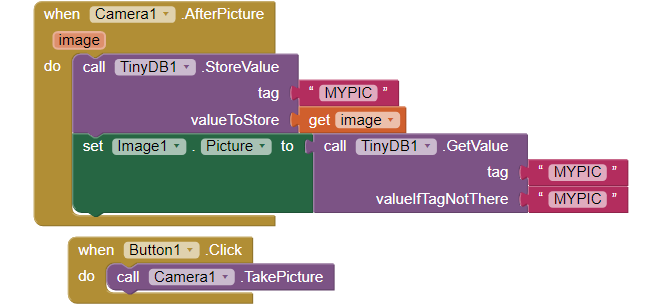
**void loop() {**

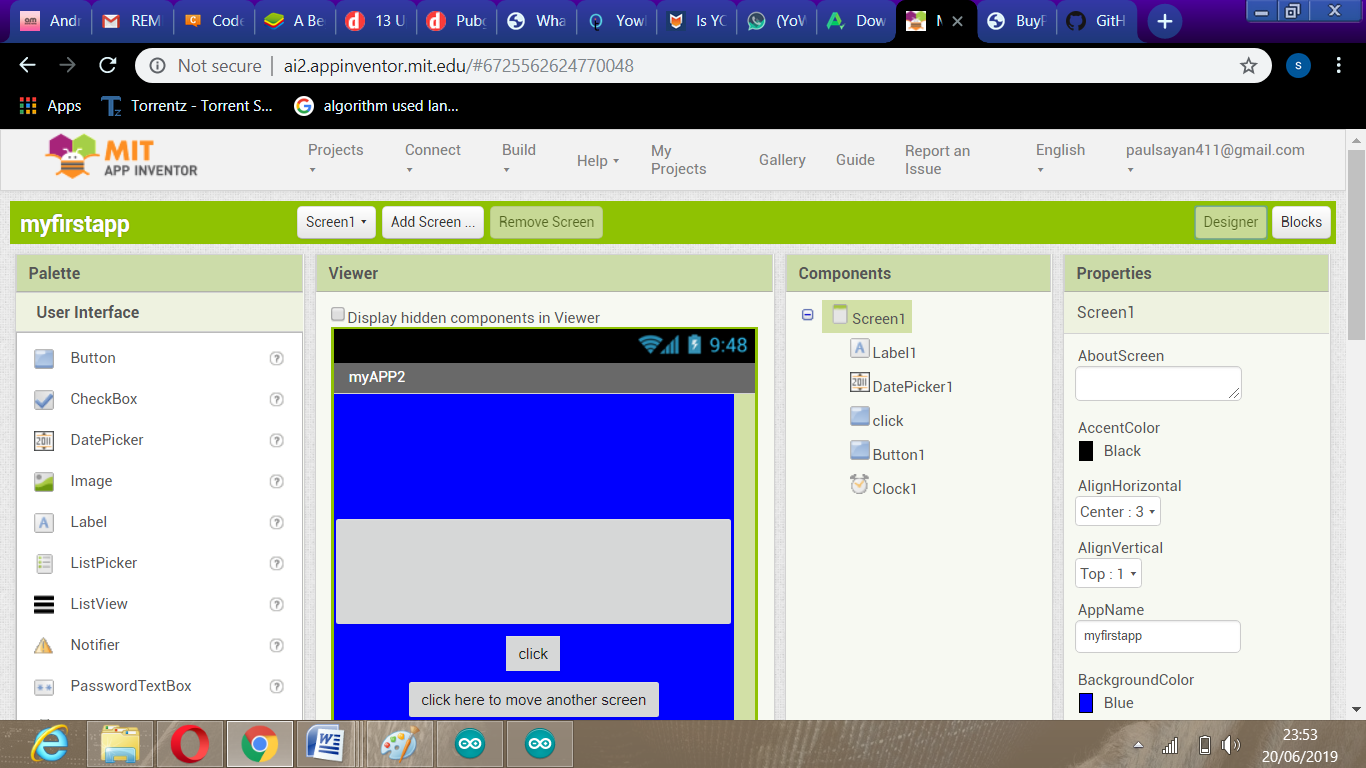
**if (swSer.available()) { Serial.write(swSer.read()); }**

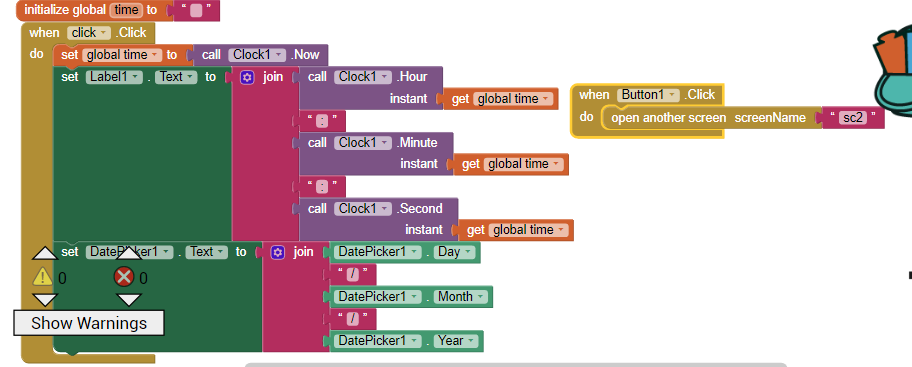
**if (Serial.available()) { swSer.write(Serial.read());**

**DAY 3**

**1.camera application and 2.date and time application**

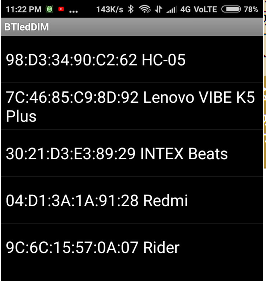
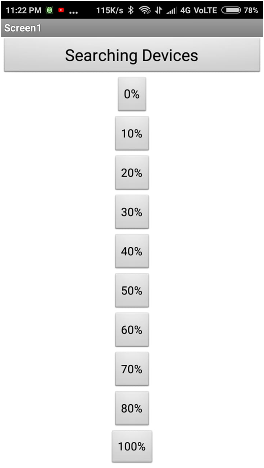
**} }**

****

****

**DAY 4**

**[Bluetooth communication with apk(led blynk)]**

**int ledPin = 12, x;**

**void setup() {**

**Serial.begin(9600); pinMode(12,OUTPUT);}**

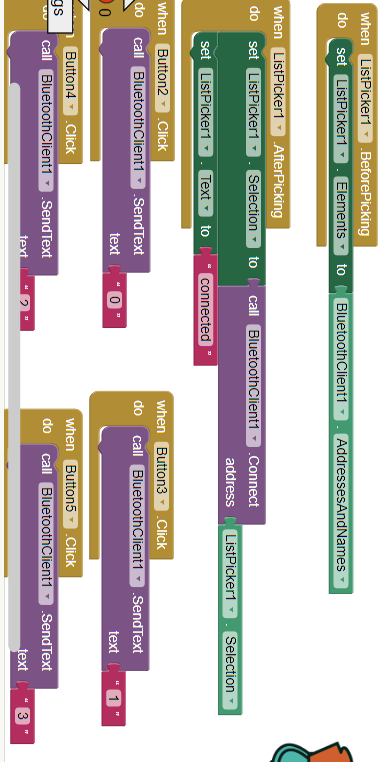
**void loop() {**

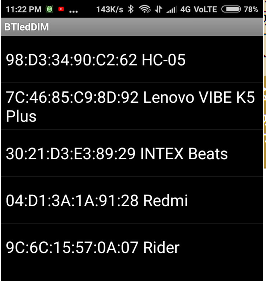
**if(Serial.available()>0)**

**{char a; a=Serial.read();**

**if(a=='0')**

**{ x = ((255/100)\*0);analogWrite(ledPin, x);}**

** else if(a=='1')**

** { x = ((255/100)\*10); analogWrite(ledPin, x); }**

**else if(a=='2')**

**{ x = ((255/100)\*20); analogWrite(ledPin, x); }**

**else if(a=='3')**

**{ x = ((255/100)\*30); analogWrite(ledPin, x); }**

**else if(a=='4')**

**{ x = ((255/100)\*40); analogWrite(ledPin, x); }**

**else if(a=='5')**

**{ x = ((255/100)\*50); analogWrite(ledPin, x); }**

**else if(a=='6')**

**{ x = ((255/100)\*60); analogWrite(ledPin, x); }**

**else if(a=='7')**

**{ x = ((255/100)\*70); analogWrite(ledPin, x); }**

**else if(a=='8')**

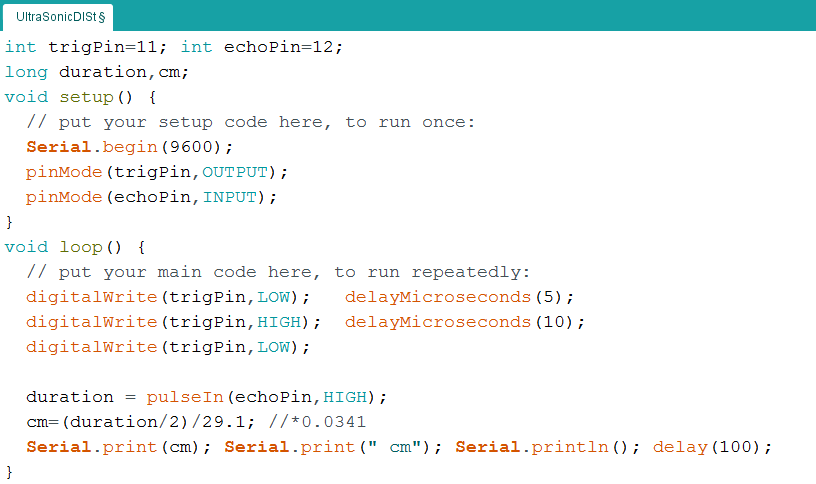
**{ x = ((255/100)\*80); analogWrite(ledPin, x); }**

**else if(a=='9') { x = ((255/100)\*100); analogWrite(ledPin, x); } } }**

**DAY-5**

**[ULTRA SONIC SENSOR(HC-SR04)]**

**Q.using ultra-sonic sensor(HC-SR04) measuring the distance of any obstacles in cm.here trigPin=11 and echoPin=12.in this sensor contain 4 pin (gnd,vcc,trig,echo).**



**Q.using ultra-sonic sensor(HC-SR04) change the intensity of led using map function. measuring the distance of any obstacles in cm.here trigPin=11 and echoPin=12.**

**int trigPin=9; int echoPin=10;**

**int ledPin = 11; long duration,cm;**

**void setup() {**

**// put your setup code here, to run once:**

**Serial.begin(9600);**

**pinMode(trigPin,OUTPUT); pinMode(echoPin,INPUT); pinMode(ledPin,OUTPUT); }**

**void loop() { // put your main code here, to run repeatedly:**

**digitalWrite(trigPin,LOW); delayMicroseconds(5);**

**digitalWrite(trigPin,HIGH); delayMicroseconds(10);**

**digitalWrite(trigPin,LOW);**

**duration = pulseIn(echoPin,HIGH);**

**cm=(duration/2)/29.1; //\*0.0341**

**/\*Serial.print(cm); Serial.print(" cm"); Serial.println(); delay(100);\*/**

**cm=map(cm,0,100,255,0);//(value,init,dest,high,low)**

**analogWrite(ledPin, cm); Serial.println(cm); }**

**[GYRO SENSOR(MPU-6050)]**

**Q.using Gyro-scope sensor(MPU-6050) blynk the led light according to the respective axix (x,y,z). measuring the distance of any obstacles in cm. in this sensor contain 4 pin (gnd,vcc,scl,sda).[Remember that gyro-scope have only 3.3v as vcc].**

**#include <Wire.h> #include <MPU6050.h>**

**MPU6050 mpu;**

**int ledPinx=11; int ledPiny=12; int ledPinz=13;**

**void loop()**

**{**

**Vector rawGyro = mpu.readRawGyro(); Vector normGyro = mpu.readNormalizeGyro();**

**Serial.print(" Xnorm = "); Serial.print(normGyro.XAxis);**

**if(normGyro.XAxis>200 || normGyro.XAxis<-123)**

**digitalWrite(ledPinx,HIGH);**

**else**

**digitalWrite(ledPinx,LOW);**

**Serial.print(" Ynorm = "); Serial.print(normGyro.YAxis);**

**if(normGyro.YAxis>200 || normGyro.YAxis<-123)**

**digitalWrite(ledPiny,HIGH);**

**else**

**digitalWrite(ledPiny,LOW);**

**Serial.print(" Znorm = "); Serial.println(normGyro.ZAxis);**

**if(normGyro.ZAxis>200 || normGyro.ZAxis<-123)**

**digitalWrite(ledPinz,HIGH);**

**else**

**digitalWrite(ledPinz,LOW);**

**delay(50);**

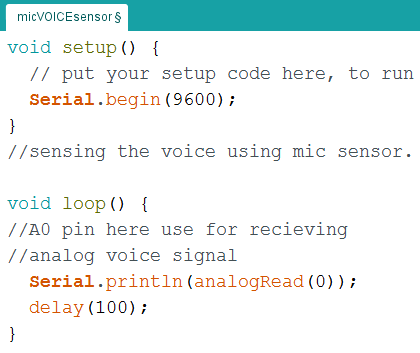
**}**

**[here only the loop functions means only the basic logic is given that will give our bynking output. And if we need the total code then we need to open that code via example sctions. The process is given**

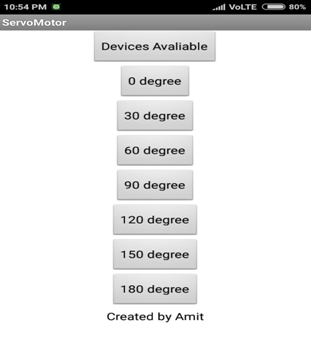
**File->Examples ->arduino-mpu6050-master->mpu6050\_gyro\_simple]**

**DAY-6**

**[SERVO MOTOR & MIC SENSOR]**

****

**Q.servo motor given angle wise rotation using Bluetooth android app.**

**#include<Servo.h>//header file**

**Servo ogma ;//object**

**void setup() { ogma.attach(9);**

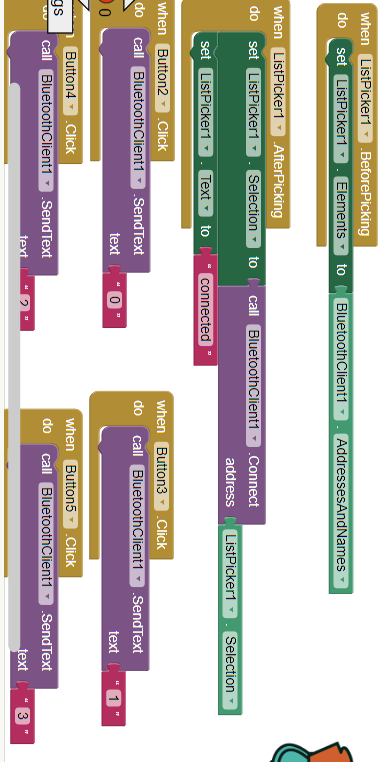
**Serial.begin(9600);**

**}**

**void loop() {**

**if(Serial.available()>0)**

**{char a;**

** a=Serial.read();**

**if(a=='0')**

**{ ogma.write(0);//angle given}**

**else if(a=='1')**

**{ ogma.write(30);//angle given}**

**else if(a=='2')**

**{ ogma.write(60);//angle given }**

**else if(a=='3')**

**{ ogma.write(90);//angle given }**

**else if(a=='4')**

**{ ogma.write(120);//angle given}**

**else if(a=='5')**

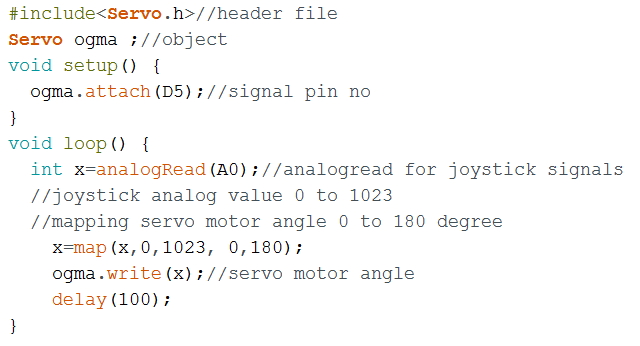
**{ ogma.write(150);//angle given}**

**else if(a=='6')**

**{ ogma.write(180);//angle given}**

**} }**

**Q.servo motor angle wise rotation by using joystick.**

****

**\*Google voice enable (using wifi/hotspot) led on and of using arduino is stored in arduino folder.**

**Q.Arduino led on and off using html web pages through local area network/LAN(mobile hotspot).[Remember you need to connect your laptop (web client) need to be connected to same as arduino connected mobile hotspot].**

**#include <ESP8266WiFi.h>**

const char\* ssid = "sayan"; //mobile hotspot name(LAN network)

const char\* password = "sp123456"; mobile hotspot password(LAN network)

**int ledPin = 13; // GPIO13---D7 of NodeMCU**

**WiFiServer server(80);**

void setup() {

Serial.begin(115200); delay(10);

pinMode(ledPin, OUTPUT); digitalWrite(ledPin, LOW);

// Connect to WiFi network

Serial.println(); Serial.println();

Serial.print("Connecting to "); Serial.println(ssid);

WiFi.begin(ssid, password);

while (WiFi.status() != WL\_CONNECTED) { delay(500); Serial.print("."); }

Serial.println(""); Serial.println("WiFi connected");

// Start the server

server.begin(); Serial.println("Server started");

// Print the IP address

Serial.print("Use this URL to connect: ");

Serial.print("http://");

Serial.print(WiFi.localIP());

Serial.println("/"); }

void loop() {

// Check if a client has connected

WiFiClient client = server.available();

if (!client) { return; }

// Wait until the client sends some data

Serial.println("new client");

while(!client.available()){ delay(1); }

// Read the first line of the request

String request = client.readStringUntil('\r');

Serial.println(request);

client.flush();

**// Match the request**

int value = LOW;

if (request.indexOf("/LED=ON") != -1) {

digitalWrite(ledPin, HIGH); value = HIGH; }

if (request.indexOf("/LED=OFF") != -1) {

digitalWrite(ledPin, LOW);

value = LOW; }

**// Set ledPin according to the request**

**//digitalWrite(ledPin, value);**

// Return the response

client.println("HTTP/1.1 200 OK");

client.println("Content-Type: text/html");

client.println(""); // do not forget this one

client.println("<!DOCTYPE HTML>");

client.println("<html>");

client.print("Led is now: ");

if(value == HIGH) {

client.print("On");

} else { client.print("Off"); }

client.println("<br><br>");

client.println("<a href=\"/LED=ON\"\"><button>On </button></a>");

client.println("<a href=\"/LED=OFF\"\"><button>Off </button></a><br />");

client.println("</html>");

**delay(1);**

**Serial.println("Client disonnected"); Serial.println(""); }**

**Q.** **solar intensity(VOLTAGE) display in LCD.others demo programs are available in example sections(LiquidCrystal) .**

****

@For downloading the herculis serial monitor(hercules\_3-2-8.exe) click this link .it is used for local network communication in nodemcu arduino.

<https://www.hw-group.com/software/hercules-setup-utility>

#include <ESP8266WiFi.h> // Include the Wi-Fi library

//using local

const char\* ssid = "sayan"; // The SSID (name) of the Wi-Fi network you want to connect to

const char\* password = "sp123456"; // The password of the Wi-Fi network

WiFiServer server(80);

void setup() {

Serial.begin(115200); // Start the Serial communication to send messages to the computer

WiFi.mode(WIFI\_STA);

delay(10);

Serial.println('\n');

WiFi.begin(ssid, password); // Connect to the network

Serial.print("Connecting to ");

Serial.print(ssid); Serial.println(" ...");

int i = 0;

while (WiFi.status() != WL\_CONNECTED) { // Wait for the Wi-Fi to connect

delay(1000);

Serial.print(++i); Serial.print(' ');

}

Serial.println('\n');

Serial.println("Connection established!");

Serial.print("IP address:\t");

Serial.println(WiFi.localIP()); // Send the IP address of the ESP8266 to the computer

server.begin();

}

void loop() {

WiFiClient client = server.available(); // Listen for incoming clients

for(int i=100;i<200;i++){

client.println(i);

delay(100);

}

}

Upload this code into nodemcu and open the serial monitor in arduino ide .There you will get your local IP which is given by your LAN/ROUTER(mobile hotspot). Put this local IP into your herculis serial monitor (click tcp client->update mobile ip->connect) and you will get the calculation which you given above in your code in arduino ide and upload in arduino microcontroller.

**Q.Reading Temperature and Humidity using DHT11 Temperature and Humidity sensor and ploting it on GOOGLE SHEET through GOOGLE DRIVE.remember in google sheet you have to provide appropriate column name exactly same as JS code and arduino code.[temperature,humidity]**

**Arduino code:**

**/ //https://script.google.com/macros/s/AKfycbzee0Wk4vY8meQFqMv639-ZYMK8G-w\_Ox4EWOcivOslznsvFgw/exec //js code**

#include <ESP8266WiFi.h>

#include <WiFiClientSecure.h>

#include<DHT.h>

#define DHTPIN D2

#define DHTTYPE DHT11

void sendData(int, int);

DHT dht(DHTPIN, DHTTYPE);

const char\* ssid = "sayan"; // name of your wifi network!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

const char\* password = "sp123456"; // wifi pasword !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

const char\* host = "script.google.com";

const int httpsPort = 443;

// Use WiFiClientSecure class to create TLS connection

WiFiClientSecure client;

// SHA1 fingerprint of the certificate, don't care with your GAS service

const char\* fingerprint = "46 B2 C3 44 9C 59 09 8B 01 B6 F8 BD 4C FB 00 74 91 2F EF F6";

String GAS\_ID = "AKfycbzee0Wk4vY8meQFqMv639-ZYMK8G-w\_Ox4EWOcivOslznsvFgw"; // Replace by your GAS service id !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!//

int it;

int ih;

void setup()

{

dht.begin(); // sensor

Serial.begin(115200); //Serial

Serial.println();

//connecting to internet

Serial.print("connecting to ");

Serial.println(ssid);

WiFi.mode(WIFI\_STA);

WiFi.begin(ssid, password);

while (WiFi.status() != WL\_CONNECTED) {

delay(500);

Serial.print(".");

}

Serial.println("");

Serial.println("WiFi connected");

Serial.println("IP address: ");

Serial.println(WiFi.localIP());

}

void loop()

{

float h = dht.readHumidity();

float t = dht.readTemperature();

Serial.print("Temp = ");

Serial.print(t);

Serial.print(" HUM= ");

Serial.println(h);

it = (int) t;

ih = (int) h;

sendData(it, ih);

delay(2000);

}

// Function for Send data into Google Spreadsheet

void sendData(int tem, int hum)

{

Serial.print("connecting to ");

Serial.println(host);

if (!client.connect(host, httpsPort)) {

Serial.println("connection failed");

return;

}

if (client.verify(fingerprint, host)) {

Serial.println("certificate matches");

} else {

Serial.println("certificate doesn't match");

}

String string\_temperature = String(tem, DEC);

String string\_humidity = String(hum, DEC);

String url = "/macros/s/" + GAS\_ID + "/exec?temperature=" + string\_temperature + "&humidity=" + string\_humidity;

Serial.print("requesting URL: ");

Serial.println(url);

client.print(String("GET ") + url + " HTTP/1.1\r\n" +

"Host: " + host + "\r\n" +

"User-Agent: BuildFailureDetectorESP8266\r\n" +

"Connection: close\r\n\r\n");

Serial.println("request sent");

while (client.connected()) {

String line = client.readStringUntil('\n');

if (line == "\r") {

Serial.println("headers received");

break;

}

}

String line = client.readStringUntil('\n');

if (line.startsWith("{\"state\":\"success\"")) {

Serial.println("esp8266/Arduino CI successfull!");

} else {

Serial.println("esp8266/Arduino CI has failed");

}

Serial.println("reply was:");

Serial.println("==========");

Serial.println(line);

Serial.println("==========");

Serial.println("closing connection");

} /https://

Java script code:

function doGet(e) {

Logger.log( JSON.stringify(e) ); // view parameters

var result = 'Ok'; // assume success

if (e.parameter == 'undefined') {

result = 'No Parameters';

}

else {

var sheet\_id = '1mWFDN8zH\_OYvnw0or1vnDgqTJ4-ldye2OY3R5WAHVGU'; // Spreadsheet ID

var sheet = SpreadsheetApp.openById(sheet\_id).getActiveSheet(); // get Active sheet

var newRow = sheet.getLastRow() + 1;

var rowData = [];

rowData[0] = new Date(); // Timestamp in column A

for (var param in e.parameter) {

Logger.log('In for loop, param=' + param);

var value = stripQuotes(e.parameter[param]);

Logger.log(param + ':' + e.parameter[param]);

switch (param) {

case 'temperature': //Parameter

rowData[1] = value; //Value in column B

result = 'Written on column B';

break;

case 'humidity': //Parameter

rowData[2] = value; //Value in column C

result += ' ,Written on column C';

break;

default:

result = "unsupported parameter";

}

}

Logger.log(JSON.stringify(rowData));

// Write new row below

var newRange = sheet.getRange(newRow, 1, 1, rowData.length);

newRange.setValues([rowData]);

}

// Return result of operation

return ContentService.createTextOutput(result);

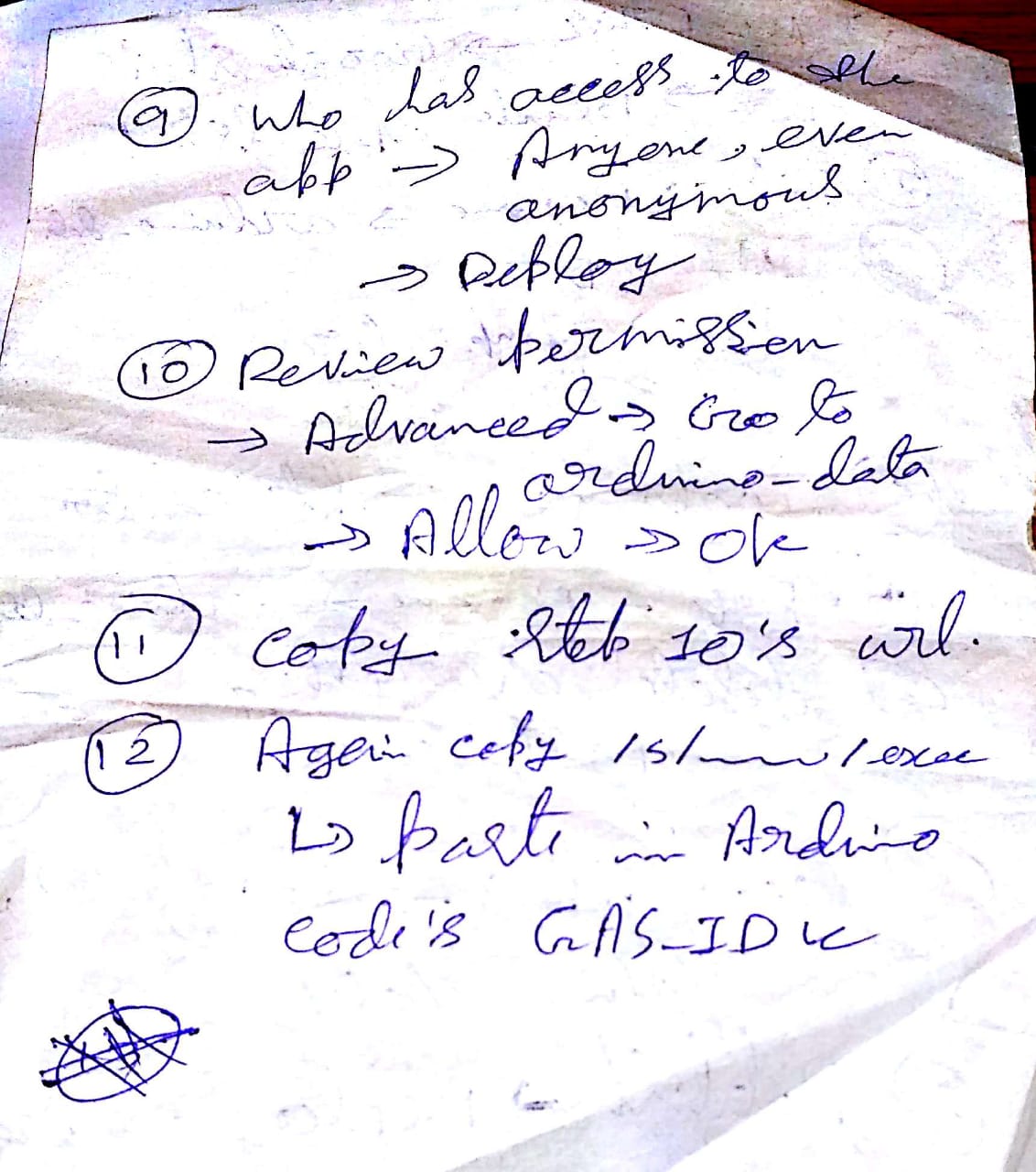
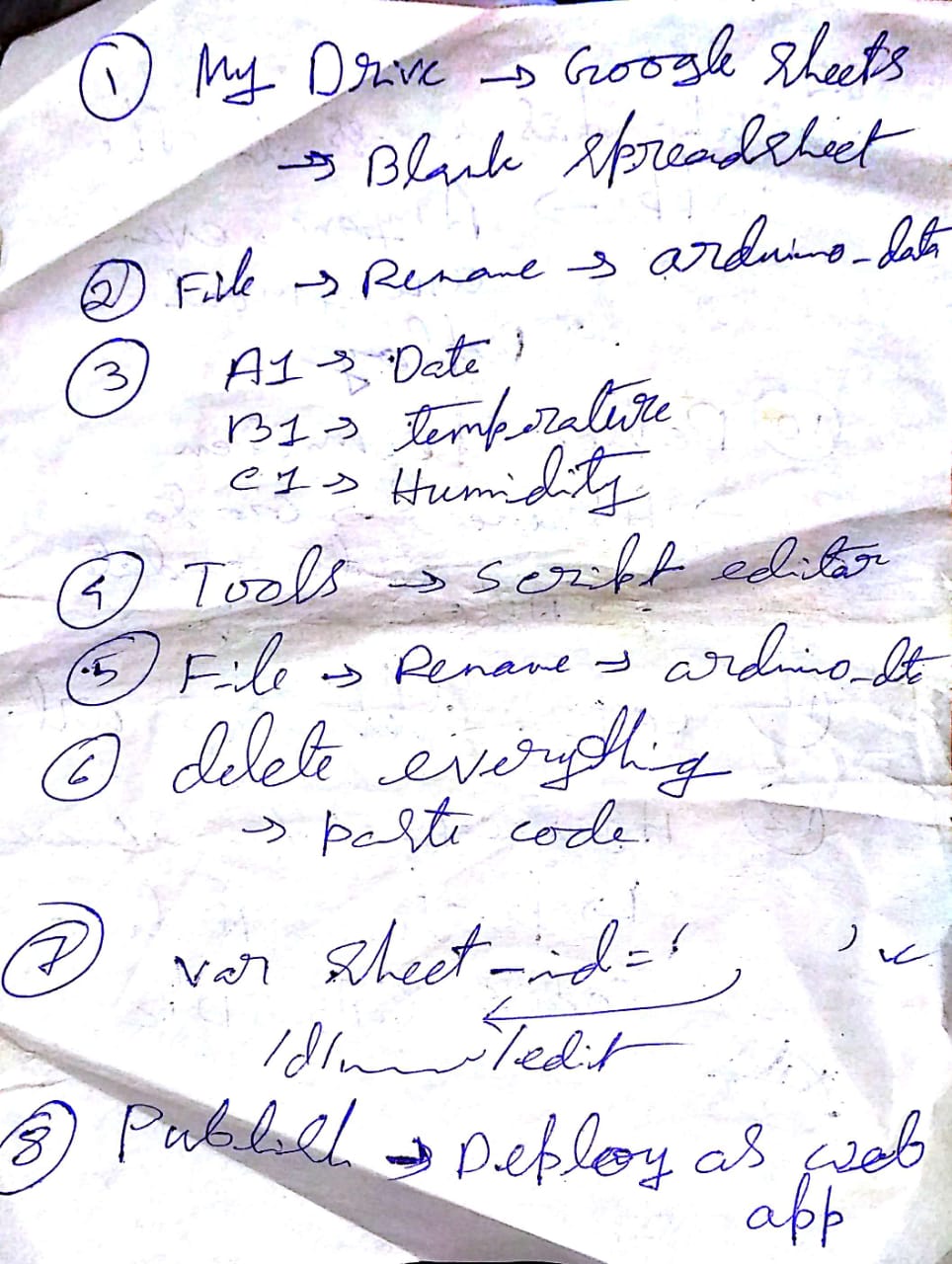
}

function stripQuotes( value ) {

return value.replace(/^["']|['"]$/g, "");

}

Fofollowing steps are given how you can collect your temperature and humidity into your GOOGLE SHEET .also the given arduino and Js code are given into the arduino folder->Google sheet folder in D drive.



script.google.com/macros/s/AKfycbzee0Wk4vY8meQFqMv639-

