# Program for Aurdino Uno

```
int level = A0;

int rain =A1;

#include <SoftwareSerial.h>//Library used to create software serial port

               // library used to enable i2c port

SoftwareSerial serial2(10, 11);

#include <DHT.h>

#define DHTPIN 7    // Pin connected to the DHT11 sensor

#define DHTTYPE DHT11 // Type of DHT sensor

DHT dht(DHTPIN, DHTTYPE);


int buz= 13;

String str ="";

void setup() {

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

Serial.begin(9600);

serial2.begin(9600);

pinMode(level,INPUT);

pinMode(rain,INPUT);

pinMode(buz,OUTPUT);

   digitalWrite(buz,LOW);

 dht.begin();


}


void loop() {

  // put your main code here, to run repeatedly:

int levelVal= analogRead(level);

Serial.print(levelVal);
Serial.print(",");

int rainVal = analogRead(rain);
Serial.print(rainVal);
Serial.print(",");

int humidity = dht.readHumidity();
int temperature = dht.readTemperature();

   // Check if any errors occurred while reading the sensor
   if (isnan(humidity) || isnan(temperature)) {
    Serial.println("Failed to read data from DHT11 sensor");
    return;
   }

    // Print the temperature and humidity to the serial monitor
   // Serial.print("Temperature: ");
   Serial.print(temperature);
   Serial.print(",");

   // Serial.print(" °C, Humidity: ");
   Serial.print(humidity);
   Serial.print(",");

 if(levelVal >500 ){
  Serial.print("1");
Serial.println(",");
  sendLevel();

 }
 else if(rainVal<=900){
    Serial.print("1");
Serial.println(",");
    sendRain();


  }
  else{
    digitalWrite(buz,LOW);
    Serial.print("0");
Serial.println(",");
  }
delay(2000);


}




void sendLevel(){
        digitalWrite(buz,HIGH);
str = String ("water level is to high");





  serial2.println("AT+CMGF=1");    //Sets the GSM Module in Text Mode
  delay(1000);  // Delay of 1000 milli seconds or 1 second
  serial2.println("AT+CMGS=\"+919380757402\"\r"); // Replace x with mobile number
  delay(1000);
serial2.println(str);// The SMS text you want t,m nb  o send
  delay(100);
   serial2.println((char)26);// ASCII code of CTRL+Z
  delay(8000);
  serial2.println("ATD+ +919380757402;");
delay(20000);
```

```
serial2.println("ATH");
      digitalWrite(buz,LOW);


}


void sendRain(){
        digitalWrite(buz,HIGH);
str = String ("rain density level is to high");





   serial2.println("AT+CMGF=1");    //Sets the GSM Module in Text Mode
  delay(1000);  // Delay of 1000 milli seconds or 1 second
  serial2.println("AT+CMGS=\"+919380757402\"\r"); // Replace x with mobile number
  delay(1000);
serial2.println(str);// The SMS text you want t,m nb  o send
  delay(100);
   serial2.println((char)26);// ASCII code of CTRL+Z
  delay(8000);
  serial2.println("ATD+ +919380757402;");
delay(20000);
serial2.println("ATH");
      digitalWrite(buz,LOW);
} #include <ESP8266WiFi.h>
#include <WiFiClient.h>
#include <ThingSpeak.h>


const char* ssid = "Shri";   // Your Network SSID
const char* password = "Shri@1012";       // Your Network Password


uint32_t tsLastReport = 0;
```

```
#define REPORTING_PERIOD_MS     10000




WiFiClient client;


unsigned long myChannelNumber = 2889480; //Your Channel Number (Without Brackets)
const char * myWriteAPIKey = "1TCZCTX41TUMUEG8"; //Your Write API Key


void setup()
{
  Serial.begin(9600);


      WiFi.begin(ssid, password);
  ThingSpeak.begin(client);
  delay(100);



}


void loop()
{
if (Serial.available()) {
    String data = Serial.readStringUntil('\n');   // x data (2,0.5,37)
    int firstSeparatorIndex = data.indexOf(",");
    int secondSeparatorIndex = data.indexOf(",", firstSeparatorIndex + 1);
    int thirdSeparatorIndex = data.indexOf(",", secondSeparatorIndex + 1);
    int fourthSeparatorIndex = data.indexOf(",", thirdSeparatorIndex + 1);
        int fivthSeparatorIndex = data.indexOf(",", fourthSeparatorIndex + 1);
```

```
   int sensorValue1 = data.substring(0, firstSeparatorIndex).toInt();

   int sensorValue2 = data.substring(firstSeparatorIndex + 1, secondSeparatorIndex).toInt();

   int sensorValue3 = data.substring(secondSeparatorIndex + 1, thirdSeparatorIndex).toInt();

   int sensorValue4 = data.substring(thirdSeparatorIndex + 1, fourthSeparatorIndex).toInt();

      int sensorValue5 = data.substring(fourthSeparatorIndex + 1, fivthSeparatorIndex).toInt();



   Serial.println(sensorValue1);

    Serial.println(sensorValue2);

        Serial.println(sensorValue3);

                Serial.println(sensorValue4);

                                Serial.println(sensorValue5);




   if (millis() - tsLastReport > REPORTING_PERIOD_MS) {


ThingSpeak.setField( 1,sensorValue1); //Update in ThingSpeak

  ThingSpeak.setField( 2,sensorValue2); //Update in ThingSpeak

   ThingSpeak.setField( 3,sensorValue3); //Update in ThingSpeak

    ThingSpeak.setField( 4,sensorValue4); //Update in ThingSpeak

       ThingSpeak.setField( 5,sensorValue5); //Update in ThingSpeak



        ThingSpeak.writeFields(myChannelNumber, myWriteAPIKey);   // write all fields to the channel
and reset stored


   tsLastReport = millis();
```

```
    }

}


}
```