Sprint 3

Date	16 November 2022
Team ID	PNT2022TMID45187
Project Name	Smart Farmer-IoT Enabled smart Farming Application
Maximum Marks	4 Marks

INTRODUCTION:

The main aim of this project is to help farmers automate their farms by providing them with a Web App through which they can monitor the parameters of the field like Temperature, soil moisture, humidity and etc and control the equipment like water motor and other devices remotely via internet without their actual presencein the field.

Sprint-3

MIT App Inventor, Dashboard (Application for your project using MIT App, Design the model and test the App)

Steps to configure:

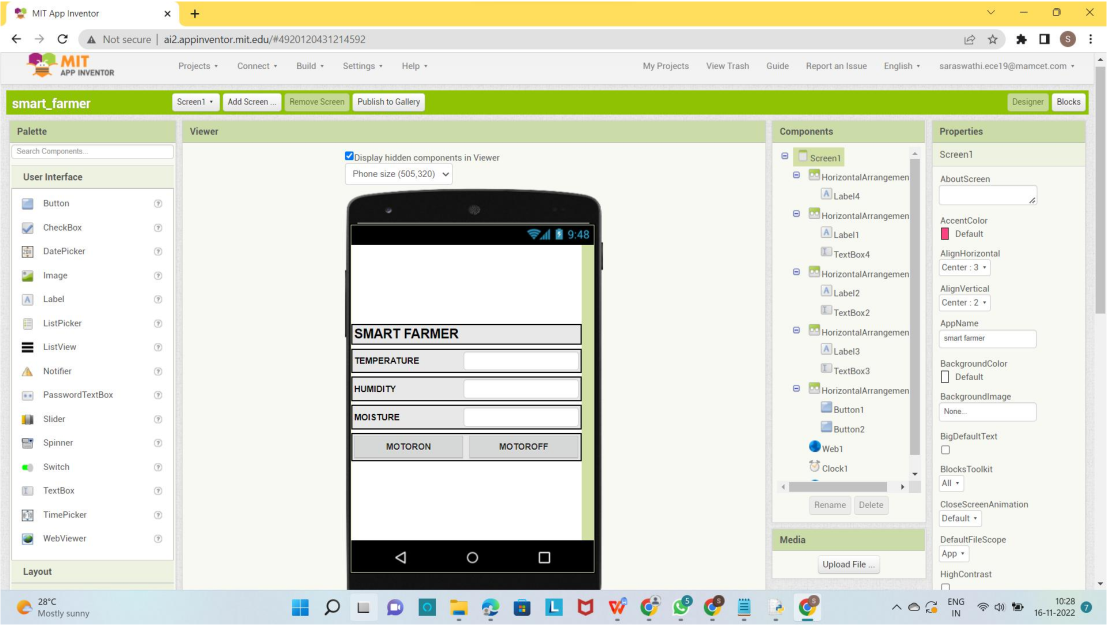
1) Create a account in the MIT App Inventor.
2) Then choose create apps and create a new project and name it.
3) Design the Designer and Blocks for your Requirement.

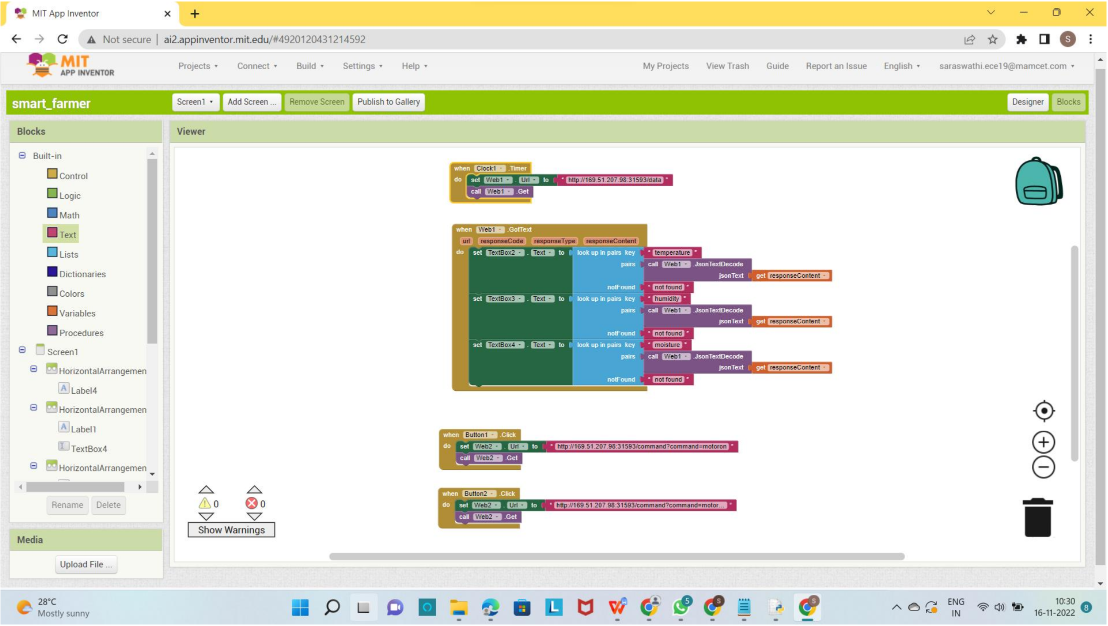
4) And connect with your MIT APP Companion in your phone

(Install the MIT Companion using Playstore)

5) Finally run the program it shows the data to your mobile.

THE PROCESS:



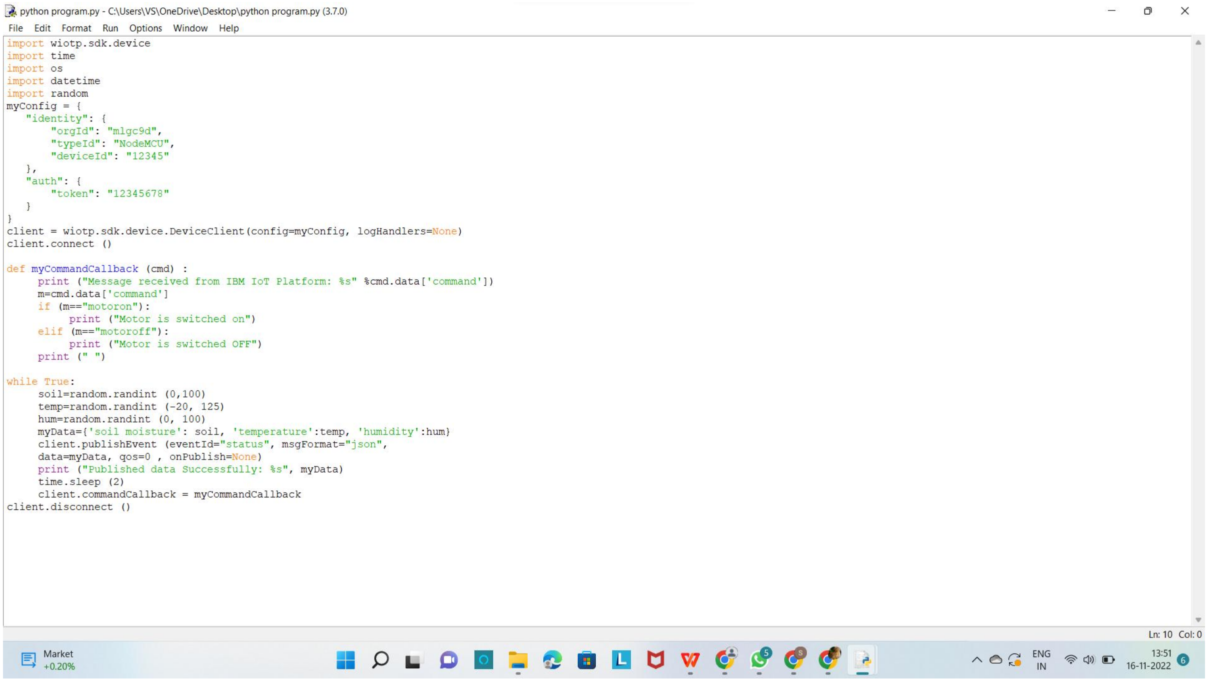


```
when Clock1 .Timer
do  set Web1 . Url to "http://169.51.207.98:31593/data"
    call Web1 .Get

when Web1 .GotText
 url  responseCode  responseType  responseContent
do  set TextBox2 . Text to  look up in pairs  key "temperature"
                                              pairs  call Web1 .JsonTextDecode
                                                               jsonText  get responseContent
                                              notFound "not found"
    set TextBox3 . Text to  look up in pairs  key "humidity"
                                              pairs  call Web1 .JsonTextDecode
                                                               jsonText  get responseContent
                                              notFound "not found"
    set TextBox4 . Text to  look up in pairs  key "moisture"
                                              pairs  call Web1 .JsonTextDecode
                                                               jsonText  get responseContent
                                              notFound "not found"

when Button1 .Click
do  set Web2 . Url to "http://169.51.207.98:31593/command?command=motoron"
    call Web2 .Get

when Button2 .Click
do  set Web2 . Url to "http://169.51.207.98:31593/command?command=motor..."
    call Web2 .Get
```

Python program

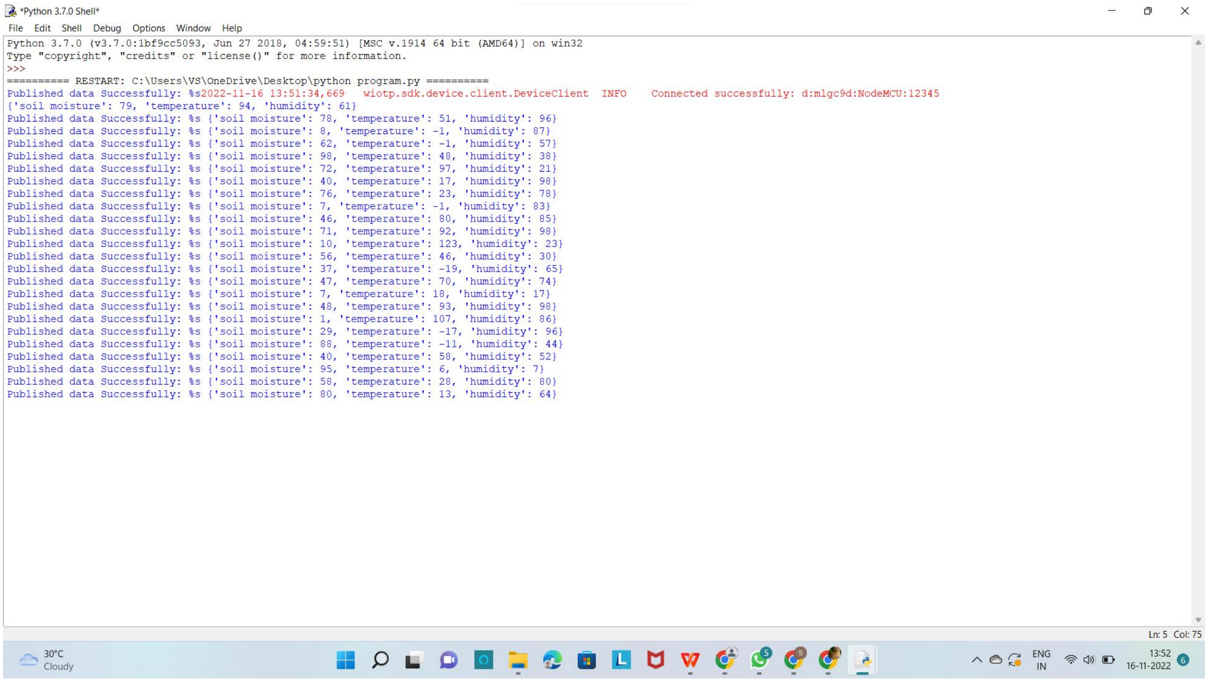
```
import wiotp.sdk.device

import time

import os

import datetime

import random

myConfig = {

    "identity": {

        "orgId": "mlgc9d",

        "typeId": "NodeMCU",

        "deviceId": "12345"

    },

    "auth": {
      "token": "12345678"
   }
}
client = wiotp.sdk.device.DeviceClient(config=myConfig,
logHandlers=None)
client.connect ()

def myCommandCallback (cmd) :
    print ("Message received from IBM IoT
Platform: %s" %cmd.data['command'])
    m=cmd.data['command']
    if (m=="motoron"):
        print ("Motor is switched on")
    elif (m=="motoroff"):
        print ("Motor is switched OFF")
    print (" ")

while True:
    soil=random.randint (0,100)
    temp=random.randint (-20, 125)
    hum=random.randint (0, 100)
```

```
        myData={'soil moisture': soil, 'temperature':temp,
'humidity':hum}
        client.publishEvent (eventId="status", msgFormat="json",
        data=myData, qos=0 , onPublish=None)
        print ("Published data Successfully: %s", myData)
        time.sleep (2)
        client.commandCallback = myCommandCallback
client.disconnect ()
```


python program.py - C:\Users\VS\OneDrive\Desktop\python program.py (3.7.0)
File Edit Format Run Options Window Help
Ln: 10 Col: 0
Market
+0.20%
ENG
IN
13:51
16-11-2022

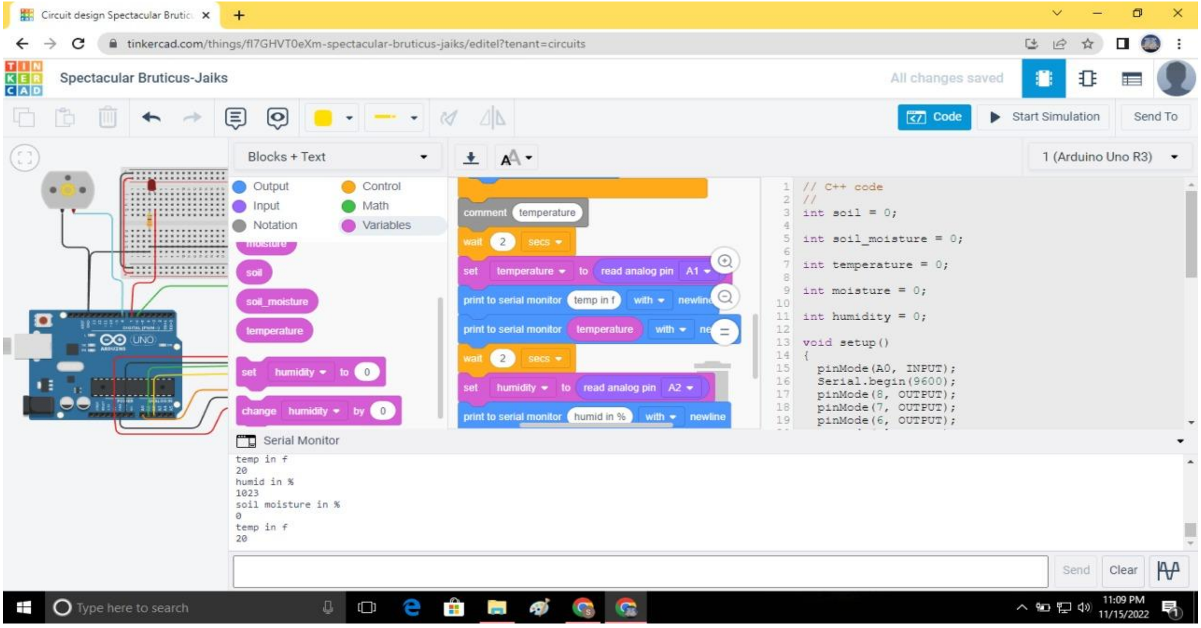
```
import wiotp.sdk.device
import time
import os
import datetime
import random
myConfig = {
   "identity": {
       "orgId": "mlgc9d",
       "typeId": "NodeMCU",
       "deviceId": "12345"
   },
   "auth": {
       "token": "12345678"
   }
}
client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect ()

def myCommandCallback (cmd) :
     print ("Message received from IBM IoT Platform: %s" %cmd.data['command'])
     m=cmd.data['command']
     if (m=="motoron"):
          print ("Motor is switched on")
     elif (m=="motoroff"):
          print ("Motor is switched OFF")
     print (" ")

while True:
     soil=random.randint (0,100)
     temp=random.randint (-20, 125)
     hum=random.randint (0, 100)
     myData={'soil moisture': soil, 'temperature':temp, 'humidity':hum}
     client.publishEvent (eventId="status", msgFormat="json",
     data=myData, qos=0 , onPublish=None)
     print ("Published data Successfully: %s", myData)
     time.sleep (2)
     client.commandCallback = myCommandCallback
client.disconnect ()
```

Program output



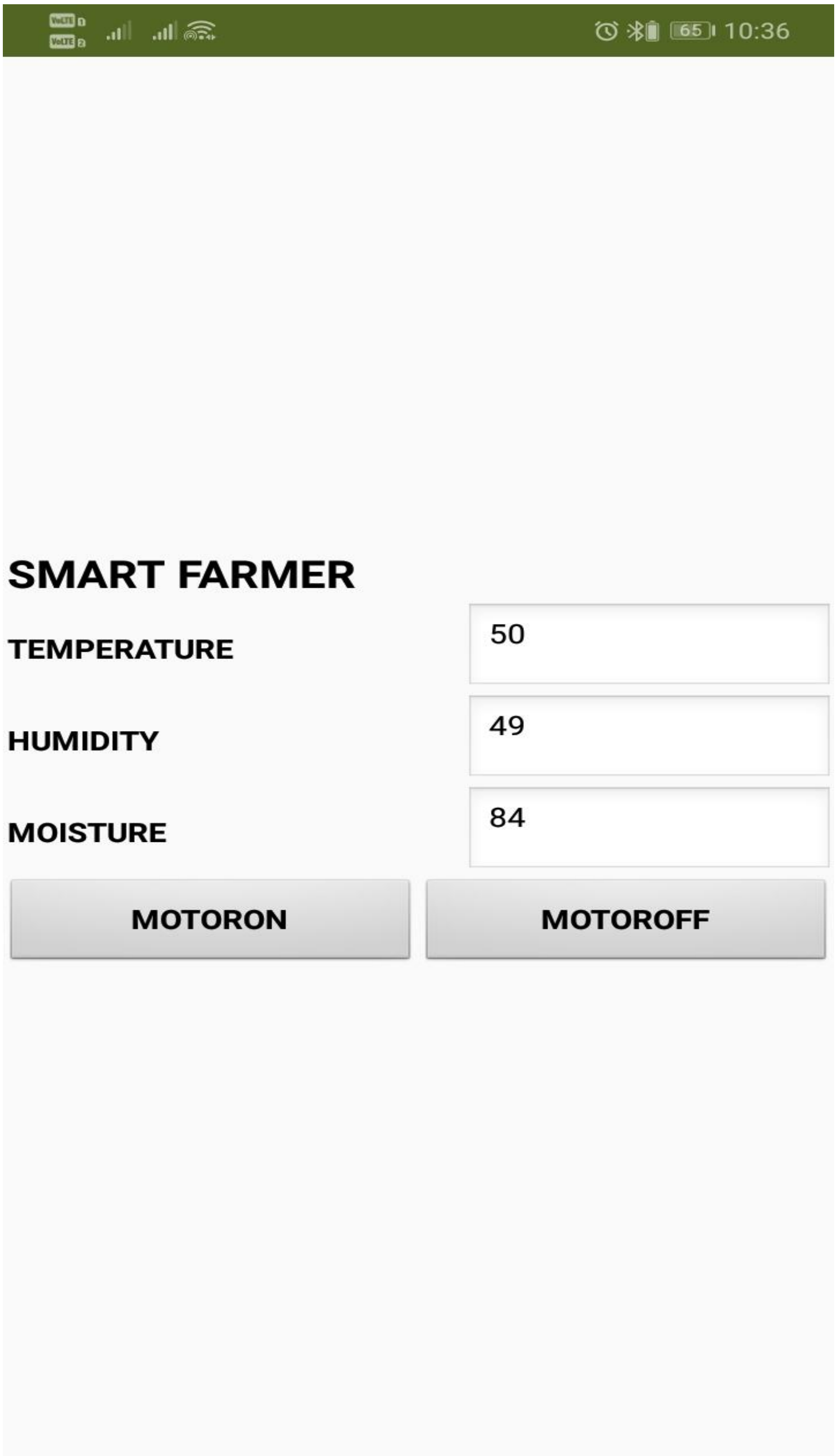
```
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) [MSC v.1914 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> 
========== RESTART: C:\Users\VS\OneDrive\Desktop\python program.py ==========
Published data Successfully: %s2022-11-16 13:51:34,669   wiotp.sdk.device.client.DeviceClient  INFO    Connected successfully: d:mlgc9d:NodeMCU:12345
{'soil moisture': 79, 'temperature': 94, 'humidity': 61}
Published data Successfully: %s {'soil moisture': 78, 'temperature': 51, 'humidity': 96}
Published data Successfully: %s {'soil moisture': 8, 'temperature': -1, 'humidity': 87}
Published data Successfully: %s {'soil moisture': 62, 'temperature': -1, 'humidity': 57}
Published data Successfully: %s {'soil moisture': 98, 'temperature': 48, 'humidity': 38}
Published data Successfully: %s {'soil moisture': 72, 'temperature': 97, 'humidity': 21}
Published data Successfully: %s {'soil moisture': 40, 'temperature': 17, 'humidity': 98}
Published data Successfully: %s {'soil moisture': 76, 'temperature': 23, 'humidity': 78}
Published data Successfully: %s {'soil moisture': 7, 'temperature': -1, 'humidity': 83}
Published data Successfully: %s {'soil moisture': 46, 'temperature': 80, 'humidity': 85}
Published data Successfully: %s {'soil moisture': 71, 'temperature': 92, 'humidity': 98}
Published data Successfully: %s {'soil moisture': 10, 'temperature': 123, 'humidity': 23}
Published data Successfully: %s {'soil moisture': 56, 'temperature': 46, 'humidity': 30}
Published data Successfully: %s {'soil moisture': 37, 'temperature': -19, 'humidity': 65}
Published data Successfully: %s {'soil moisture': 47, 'temperature': 70, 'humidity': 74}
Published data Successfully: %s {'soil moisture': 7, 'temperature': 18, 'humidity': 17}
Published data Successfully: %s {'soil moisture': 48, 'temperature': 93, 'humidity': 98}
Published data Successfully: %s {'soil moisture': 1, 'temperature': 107, 'humidity': 86}
Published data Successfully: %s {'soil moisture': 29, 'temperature': -17, 'humidity': 96}
Published data Successfully: %s {'soil moisture': 88, 'temperature': -11, 'humidity': 44}
Published data Successfully: %s {'soil moisture': 40, 'temperature': 58, 'humidity': 52}
Published data Successfully: %s {'soil moisture': 95, 'temperature': 6, 'humidity': 7}
Published data Successfully: %s {'soil moisture': 58, 'temperature': 28, 'humidity': 80}
Published data Successfully: %s {'soil moisture': 80, 'temperature': 13, 'humidity': 64}
```

Mobile Application output using MIT inventor


Circuit design Spectacular Brutic
tinkercad.com/things/fl7GHVT0eXm-spectacular-bruticus-jaiks/editel?tenant=circuits
Spectacular Bruticus-Jaiks
All changes saved
Code
Start Simulation
Send To
Blocks + Text
1 (Arduino Uno R3)
Output
Control
Input
Math
Notation
Variables
soil
soil_moisture
temperature
set humidity to 0
change humidity by 0
comment temperature
wait 2 secs
set temperature to read analog pin A1
print to serial monitor temp in f with newline
print to serial monitor temperature with newline
wait 2 secs
set humidity to read analog pin A2
print to serial monitor humid in % with newline
Serial Monitor
Send
Clear
Type here to search
11:09 PM
11/15/2022

```
// C++ code
//
int soil = 0;

int soil_moisture = 0;

int temperature = 0;

int moisture = 0;

int humidity = 0;

void setup()
{
  pinMode(A0, INPUT);
  Serial.begin(9600);
  pinMode(8, OUTPUT);
  pinMode(7, OUTPUT);
  pinMode(6, OUTPUT);
```

```
temp in f
20
humid in %
1023
soil moisture in %
0
temp in f
20
```



SMART FARMER

TEMPERATURE	50
HUMIDITY	49
MOISTURE	84