SPRINT-2

|  |  |
| --- | --- |
| Date | 31 October 2022 |
| Team ID | PNT2022TMID31899 |
| Project Name | IOT BASED CROP PROTECTION SYSTEM FOR  AGRICULTURE |

**Description :**

To generate the random values for temperature , humidity and soil moisture of the field . Below the python code is deployed and tested .

**Python Code :**

import time

import sys

import ibmiotf.application

import ibmiotf.device

import random

# Provide your IBM Watson Device Credentials

organization = "ebf2oy" # replace the ORG ID

deviceType = "Humidity" # replace the Device type

deviceId = "123456" # replace Device ID

authMethod = "token"

authToken = "C4b(zFIpnKm\_OT\_C+c" # Replace the authtoken

def myCommandCallback(cmd):

print("Command received: %s" % cmd.data['command'])

print(cmd)

try:

deviceOptions = {"org": organization, "type": deviceType, "id": deviceId,"auth-method": authMethod, "auth-token": authToken}

deviceCli = ibmiotf.device.Client(deviceOptions)

#..............................................

except Exception as e:

print("Caught exception connecting device: %s" % str(e))

sys.exit()

# Connect and send a datapoint "hello" with value "world" into the cloud as an event of type "greeting" 10 times

deviceCli.connect()

while True:

temp=random.randint(0,100)

pulse=random.randint(0,100)

soil=random.randint(0,100)

data = { 'temp' : temp, 'pulse': pulse ,'soil':soil}

#print data

def myOnPublishCallback():

print ("Published Temperature = %s C" % temp, "Humidity = %s %%"% pulse,"Soil Moisture = %s %%" % soil,"to IBM Watson")

success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0, on\_publish=myOnPublishCallback)

if not success:

print("Not connected to IoTF")

time.sleep(1)

deviceCli.commandCallback = myCommandCallback

# Disconnect the device and application from the cloud

deviceCli.disconnect()

**Output :**

