

## SOURCE CODE

TEAM ID	PNT2022TMID34905
PROJECT TITLE	IOT BASED SMART CROP PROTECTION SYSTEM FOR AGRICULTURE

CODE:

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random

#Provide your IBM Watson Device Credentials
organization = "k94ajl"
deviceType = "iotpjtddevicetype"
deviceId = "11121315"
authMethod = "token"
authToken = ")_QozclqyZG9QSg9Pe"

# Initialize GPIO

def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="motoron":
        print ("motor is on")
    else :
        print ("motor is off")

    #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:
```

```

#Get Sensor Data from DHT11

temp=random.randint(0,100)
moisture=random.randint(0,100)
animal_entry=random.randint(0,100)

data = { 'Temperature' : temp,'Moiture': moisture,"Animal Entry": animal_entry}
#print data
def myOnPublishCallback():
    print ("Published Temperature = %s C" % temp, "Moisture = %s %" % moisture,"Animal
attack",animal_entry,"to IBM Watson")

    success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0,
on_publish=myOnPublishCallback)
    if not success:
        print("Not connected to IoT")
        time.sleep(1)

    deviceCli.commandCallback = myCommandCallback

# Disconnect the device and application from the cloud
deviceCli.disconnect()

```