Code in Python IDLE:

deviceCli.commandCallback = myCommandCallback

PROGRAM:

```
smartfarmingiot.py - C:\Users\kavipriya devi\AppData\Local\Programs\Python\Python37\smartfarmingiot.py (3.7.0)
  File Edit Format Run Options Window Help import time
 #Provide your IBM Watson Device Credentials organization = "b84wgg" deviceType = "abi" deviceIge = "2345678" authMethod = "token" authToken = "87654321"
 # Initialize GPIO
  print ("Motor is OFF")
     #print(cmd)
           deviceOptions = {"org": organization, "type": de
deviceCli = ibmiotf.device.Client(deviceOptions)
#....
                                                                ": deviceType, "id": deviceId, "auth-method": authMethod, "auth-token": authToken)
          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)
Humid=random.randint(0,100)
                                                                                                                                                                                            Ln: 17 Col: 0
                                                         ■ Q ■ D ■ © ■ ● ♥ © 0 □ A △ ■ ENG © Ф 18.48 © 06-11-2022 ©
smartfarmingiot.py - C:\Users\kavipriya devi\AppData\Local\Programs\Python\Python37\smartfarmingiot.py (3.7.0)
 File Edit Format Run Options Window Help
# Initialize GPIO
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status="motoron":
        print ("Motor is ON")
    print ("Motor is ON")
else:
print ("Motor is OFF")
try:
          Exception as e:
print("Gaught 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
         data = { 'Temp' : Temp, 'Humid': Humid }
#print data
def myOnPublishCallback():
    print ("Published Temperature = %s C" % Temp, "Humidity = %s %%" % Humid, "to IBM Watson")
          success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0, on_publish=myOnPublishCallback)
if not success:
          print("Not connected to ToTF")
time.sleep(1)
```

18.48 to 19.10 to 19.

Program used in the code:

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
#Provide your IBM Watson Device Credentials
organization = "b84wgs"
deviceType = "abi"
deviceId = "12345678"
authMethod = "token"
authToken = "87654321"
# Initialize GPIO
def myCommandCallback(cmd):
  print("Command received: %s" % cmd.data['command'])
  status=cmd.data['command']
  if status=="lighton":
     print ("led is on")
  else:
     print ("led 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)
     Humid=random.randint(0,100)
     data = { 'temp' : temp, 'Humid': Humid }
     #print data
     def myOnPublishCallback():
        print ("Published Temperature = %s C" % temp,
"Humidity = %s %%" % Humid, "to IBM Watson")
     success = deviceCli.publishEvent("IoTSensor", "json", data,
gos=0, on publish=myOnPublishCallback)
     if not success:
        print("Not connected to IoTF")
     time.sleep
     deviceCli.commandCallback = myCommandCallback
```