DEVELOP A PYTHON SCRIPT TO PUBLISH AND SUBSCRIBE TO IBM IOT PLATFORM

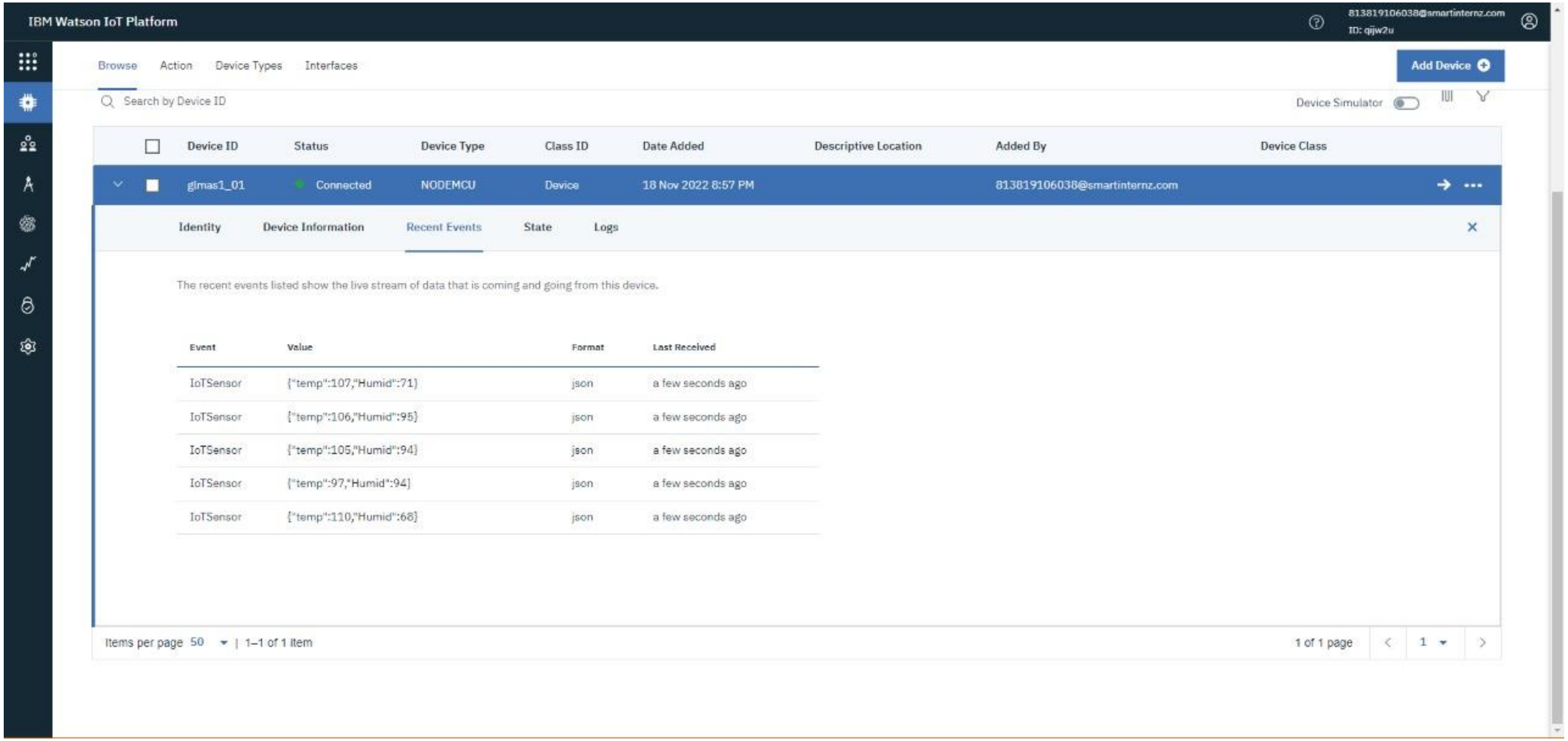
Date	15 November 2022
Team ID	PNT2022TMID32740
Project Name	Gas leakage monitoring and alerting system

PYTHON CODE:

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

#Provide your IBM Watson Device Credentials
organization = "qijw2u"
deviceType = "NODEMCU"
deviceId = "glmas1_01"
authMethod = "token"
authToken = "123456789"

# Initialize GPIO
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="lighton":
        print ("led is on")
    elif status == "lightoff":
        print ("led is off")
    else :
        print ("please send proper command")

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(90,110)
        Humid=random.randint(60,100)

        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, qos=0, on_publish=myOnPublishCallback)
        if not success:
            print("Not connected to IoTF")
        time.sleep(10)

        deviceCli.commandCallback = myCommandCallback

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

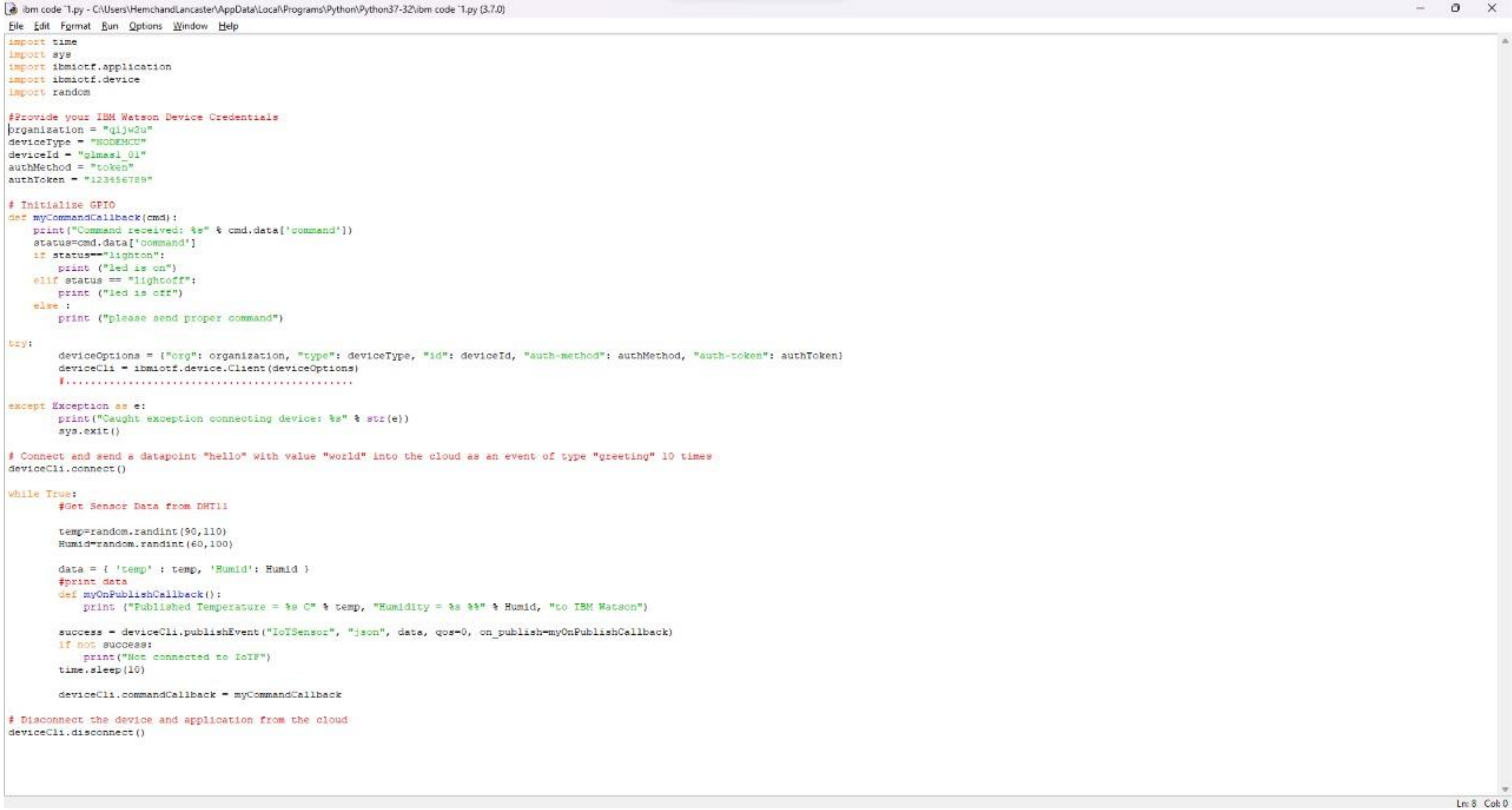
PUBLISH THE DATA TO IBM CLOUD:



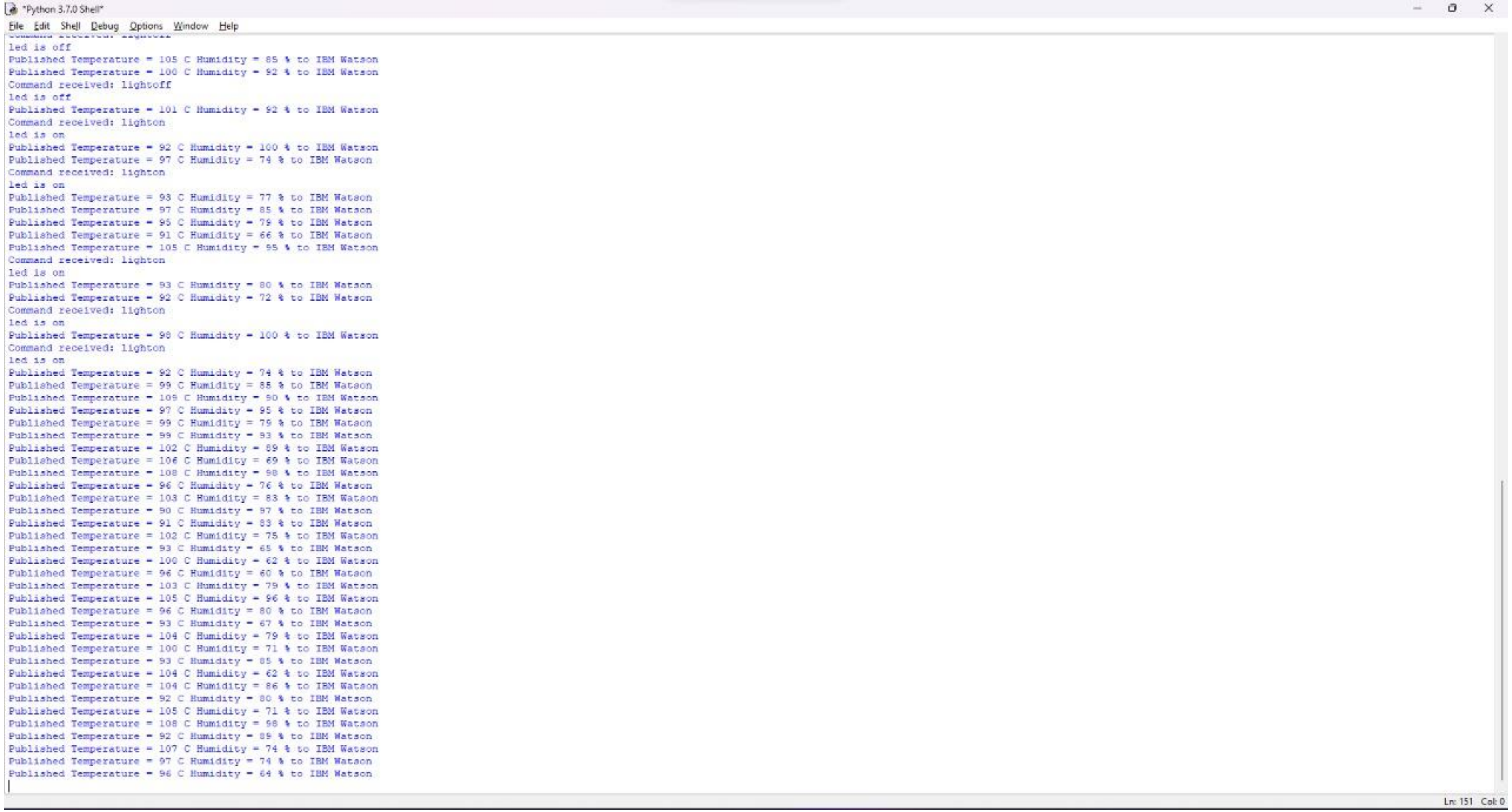
Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location	Added By	Device Class
glmas1_01	Connected	NODEMCU	Device	18 Nov 2022 8:57 PM		813819106038@smartinternz.com	

The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
IoTSensor	{"temp":107,"Humid":71}	json	a few seconds ago
IoTSensor	{"temp":106,"Humid":95}	json	a few seconds ago
IoTSensor	{"temp":105,"Humid":94}	json	a few seconds ago
IoTSensor	{"temp":97,"Humid":94]	json	a few seconds ago
IoTSensor	{"temp":110,"Humid":68}	json	a few seconds ago


ibm code `1.py - C:\Users\HemchandLancaster\AppData\Local\Programs\Python\Python37-32\ibm code `1.py (3.7.0)
File Edit Format Run Options Window Help
Ln: 8 Col: 0

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

#Provide your IBM Watson Device Credentials
organization = "qijw2u"
deviceType = "NODEMCU"
deviceId = "glmasl_01"
authMethod = "token"
authToken = "123456789"

# Initialise GPIO
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="lighton":
        print ("led is on")
    elif status == "lightoff":
        print ("led is off")
    else :
        print ("please send proper command")

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(90,110)
        Humid=random.randint(60,100)

        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, qos=0, on_publish=myOnPublishCallback)
        if not success:
            print("Not connected to IoTF")
        time.sleep(10)

        deviceCli.commandCallback = myCommandCallback

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



```
led is off
Published Temperature = 105 C Humidity = 85 % to IBM Watson
Published Temperature = 100 C Humidity = 92 % to IBM Watson
Command received: lightoff
led is off
Published Temperature = 101 C Humidity = 92 % to IBM Watson
Command received: lighton
led is on
Published Temperature = 92 C Humidity = 100 % to IBM Watson
Published Temperature = 97 C Humidity = 74 % to IBM Watson
Command received: lighton
led is on
Published Temperature = 93 C Humidity = 77 % to IBM Watson
Published Temperature = 97 C Humidity = 85 % to IBM Watson
Published Temperature = 95 C Humidity = 79 % to IBM Watson
Published Temperature = 91 C Humidity = 66 % to IBM Watson
Published Temperature = 105 C Humidity = 95 % to IBM Watson
Command received: lighton
led is on
Published Temperature = 93 C Humidity = 80 % to IBM Watson
Published Temperature = 92 C Humidity = 72 % to IBM Watson
Command received: lighton
led is on
Published Temperature = 98 C Humidity = 100 % to IBM Watson
Command received: lighton
led is on
Published Temperature = 92 C Humidity = 74 % to IBM Watson
Published Temperature = 99 C Humidity = 85 % to IBM Watson
Published Temperature = 109 C Humidity = 90 % to IBM Watson
Published Temperature = 97 C Humidity = 95 % to IBM Watson
Published Temperature = 99 C Humidity = 79 % to IBM Watson
Published Temperature = 99 C Humidity = 93 % to IBM Watson
Published Temperature = 102 C Humidity = 89 % to IBM Watson
Published Temperature = 106 C Humidity = 69 % to IBM Watson
Published Temperature = 108 C Humidity = 98 % to IBM Watson
Published Temperature = 96 C Humidity = 76 % to IBM Watson
Published Temperature = 103 C Humidity = 83 % to IBM Watson
Published Temperature = 90 C Humidity = 97 % to IBM Watson
Published Temperature = 91 C Humidity = 83 % to IBM Watson
Published Temperature = 102 C Humidity = 75 % to IBM Watson
Published Temperature = 93 C Humidity = 65 % to IBM Watson
Published Temperature = 100 C Humidity = 62 % to IBM Watson
Published Temperature = 96 C Humidity = 60 % to IBM Watson
Published Temperature = 103 C Humidity = 79 % to IBM Watson
Published Temperature = 105 C Humidity = 96 % to IBM Watson
Published Temperature = 96 C Humidity = 80 % to IBM Watson
Published Temperature = 93 C Humidity = 67 % to IBM Watson
Published Temperature = 104 C Humidity = 79 % to IBM Watson
Published Temperature = 100 C Humidity = 71 % to IBM Watson
Published Temperature = 93 C Humidity = 85 % to IBM Watson
Published Temperature = 104 C Humidity = 62 % to IBM Watson
Published Temperature = 104 C Humidity = 86 % to IBM Watson
Published Temperature = 92 C Humidity = 80 % to IBM Watson
Published Temperature = 105 C Humidity = 71 % to IBM Watson
Published Temperature = 108 C Humidity = 98 % to IBM Watson
Published Temperature = 92 C Humidity = 89 % to IBM Watson
Published Temperature = 107 C Humidity = 74 % to IBM Watson
Published Temperature = 97 C Humidity = 74 % to IBM Watson
Published Temperature = 96 C Humidity = 64 % to IBM Watson
```