

## PROJECT DEVELOPEMENT PHASE

### DELIVERY OF SPRINT – 3

Date	12 November 2022
Team ID	PNT2022TMID46424
Project Name	Project - Smart Waste Management System For Metropolitan Cities

## PYTHON CODE : [ To connect IBM WATSON ]

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

```
#Provide your IBM Watson Device Credentials
```

```
organization = "cbseji"
```

```
deviceType = "abcd"
```

```
deviceId = "1234"
```

```
authMethod = "token"
```

```
authToken = "12345678"
```

```
# 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
```

```
    level=random.randint(0,100)
```

```
    weight=random.randint(0,100)
```

```
    data = { 'level' : level, 'weight': weight }
```

```
    #print data
```

```
    def myOnPublishCallback():
```

```
        print ("Published level = %s C" % level, "weight = %s %% "  
% weight, "to IBM Watson")
```

```
    success = deviceCli.publishEvent("IoTSensor", "json", data,
```

```
qos=0, on_publish=myOnPublishCallback)
```

```
if not success:
```

```
    print("Not connected to IoTTF")
```

```
    time.sleep(1)
```

```
deviceCli.commandCallback = myCommandCallback
```

```
if (level>=75):
```

```
    print("Full LED ON")
```

```
# Disconnect the device and application from the cloud
```

```
deviceCli.disconnect()
```

OUTPUT :

The screenshot displays the IBM Watson IoT Platform interface. The main dashboard shows a table of recent events for a device named 'eventbatch11'. The events are listed with their values, formats, and last update times. A modal window is open for adding a new event type, showing a schedule of 'Every Minute' and a payload editor with a JSON schema for random number generation.

Event	Value	Format	Last
eventbatch11	{\"randomNumber\":0,\"level\":47,\"weight\":947}	json	a fe
eventbatch11	{\"randomNumber\":37,\"level\":6,\"weight\":273}	json	a m
eventbatch11	{\"randomNumber\":80,\"level\":15,\"weight\":605}	json	2 m
eventbatch11	{\"randomNumber\":54,\"level\":34,\"weight\":224}	json	3 m
eventbatch11	{\"randomNumber\":45,\"level\":54,\"weight\":407}	json	4 m

Items per page 50 | 1-1 of 1 item

Modal: Device Type: abcd

Events 1

New event type +

Event type name: eventbatch11

Schedule: 1 Every Minute

Payload: Specify the event payload in the editor window or by uploading a CSV file.

```
{
  1: \"randomNumber\": random(0,100)
  2: \"level\": random(0,100)
  3: \"weight\": random(0,1000)
}
```

Upload a CSV file

Cancel Save

```
ibmiotpublishsubscribe (1).py - C:\Users\navee\Dropbox\PC\Downloads\ibmiotpublishsubscribe (1).py (3.7.0)
File Edit Format Run Options Window Help

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

#Provide your IBM Watson Device Credentials
organization = "cbseji"
deviceType = "abcd"
deviceId = "1234"
authMethod = "token"
authToken = "12345678"

# 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": de
```

```
*Python 3.7.0 Shell*
File Edit Shell Debug Options Window Help

Python 3.7.0 (v3.7.0:1b9cc5093, 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\navee\Dropbox\PC\Downloads\ibmiotpublishsubscribe (1).py =
2022-11-13 11:52:44,654 ibmiotf.device.Client INFO Connected successfully: d:cbseji:abc
d:1234
Published level = 82 C weight = 64 % to IBM Watson
Full LED ON
Published level = 5 C weight = 2 % to IBM Watson
Published level = 22 C weight = 57 % to IBM Watson
Published level = 83 C weight = 60 % to IBM Watson
Full LED ON
Published level = 16 C weight = 12 % to IBM Watson
Published level = 19 C weight = 91 % to IBM Watson
Published level = 35 C weight = 77 % to IBM Watson
Published level = 22 C weight = 46 % to IBM Watson
Published level = 85 C weight = 68 % to IBM Watson
Full LED ON
Published level = 36 C weight = 88 % to IBM Watson
Published level = 69 C weight = 72 % to IBM Watson
Published level = 14 C weight = 3 % to IBM Watson
Published level = 99 C weight = 0 % to IBM Watson
```



