

PROJECT DEVELOPMENT PHASE

DELIVERY OF - SPRINT 1

TEAM ID	PNT2022TMID06962
PROJECT NAME	Smart Waste managementsystem for metropolitan cities

CODE

```
import random
```

```
import time
```

```
import sys
```

```
import ibmiotf.application
```

```
import ibmiotf.device
```

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

```
organization = "48az6e" # repalce it with organization ID
```

```
deviceType = "DGGI" # replace it with device type
```

```
deviceId = "1234" # repalce with device id
```

```
authMethod = "token"
```

```
authToken = "12345678" # repalce with token
```

```

def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status == 'lighton':
        print("LIGHT ON")
    elif status == 'lightoff':
        print("LIGHT 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()

deviceCli.connect()

while True:
    w= random.randint(0,100)
    l = random.randint(0,100)

```

```
# Send Temperature & Humidity to IBM Watson
```

```
data = {'weight': w,'level':1}
```

```
# print data
```

```
def myOnPublishCallback():
```

```
    print("Published data",data, "to IBM Watson")
```

```
    success = deviceCli.publishEvent("event", "json", data, 0,  
myOnPublishCallback)
```

```
if not success:
```

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

```
time.sleep(5)
```

```
deviceCli.commandCallback = myCommandCallback
```

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

```
devicecli.disconnect()
```

PROGRAM IN PYTHON IDE



```
iotpy - C:\Users\JD\AppData\Local\Programs\Python\Python37\iotpy (3.7.0)
File Edit Format Run Options Window Help

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

# Provide your IBM Watson Device Credentials

organization = "48az26e" # repalce it with organization ID
deviceType = "0061" # replace it with device type
deviceId = "1234" # repalce with device id
authMethod = "token"
authToken = "12345678" # repalce with token

def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status == 'lighton':
        print("LIGHT ON")
    elif status == 'lighttoff':
        print("LIGHT 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()

deviceCli.connect()

while True:
    w= random.randint(0,100)
```

```
iotpy - C:\Users\JD\AppData\Local\Programs\Python\Python37\iotpy (3.7.0)
File Edit Format Run Options Window Help

def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status == 'lighton':
        print("LIGHT ON")
    elif status == 'lightoff':
        print("LIGHT 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()

deviceCli.connect()

while True:
    w= random.randint(0,100)
    l = random.randint(0,100)

    # Send Temperature & Humidity to IBM Watson
    data = {'weight': w, 'level': l}

    # print data
    def myOnPublishCallback():
        print("Published data",data, "to IBM Watson")

    success = deviceCli.publishEvent("event", "json", data, 0, myOnPublishCallback)
    if not success:
        print("Not connected to IoT")
        time.sleep(5)

    deviceCli.commandCallback = myCommandCallback

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

OUTPUT

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

Published data ('weight': 90, 'level': 88) to IBM Watson
Published data ('weight': 79, 'level': 84) to IBM Watson
Published data ('weight': 8, 'level': 57) to IBM Watson
Published data ('weight': 43, 'level': 26) to IBM Watson
Published data ('weight': 67, 'level': 83) to IBM Watson
Published data ('weight': 63, 'level': 8) to IBM Watson
Published data ('weight': 24, 'level': 20) to IBM Watson
Published data ('weight': 90, 'level': 18) to IBM Watson
Published data ('weight': 0, 'level': 60) to IBM Watson
Published data ('weight': 77, 'level': 40) to IBM Watson
Published data ('weight': 13, 'level': 54) to IBM Watson
Published data ('weight': 64, 'level': 27) to IBM Watson
Published data ('weight': 94, 'level': 3) to IBM Watson
Published data ('weight': 69, 'level': 7) to IBM Watson
Published data ('weight': 13, 'level': 51) to IBM Watson
Published data ('weight': 28, 'level': 85) to IBM Watson
Published data ('weight': 97, 'level': 54) to IBM Watson
Published data ('weight': 43, 'level': 36) to IBM Watson
Published data ('weight': 77, 'level': 99) to IBM Watson
Published data ('weight': 81, 'level': 4) to IBM Watson
Published data ('weight': 6, 'level': 80) to IBM Watson
Published data ('weight': 15, 'level': 100) to IBM Watson
Published data ('weight': 31, 'level': 21) to IBM Watson
Published data ('weight': 8, 'level': 23) to IBM Watson
Published data ('weight': 66, 'level': 82) to IBM Watson
Published data ('weight': 10, 'level': 70) to IBM Watson
Published data ('weight': 2, 'level': 80) to IBM Watson
Published data ('weight': 92, 'level': 89) to IBM Watson
Published data ('weight': 37, 'level': 42) to IBM Watson
Published data ('weight': 89, 'level': 20) to IBM Watson
Published data ('weight': 0, 'level': 41) to IBM Watson
Published data ('weight': 2, 'level': 89) to IBM Watson
Published data ('weight': 1, 'level': 92) to IBM Watson
Published data ('weight': 37, 'level': 76) to IBM Watson
Published data ('weight': 9, 'level': 50) to IBM Watson
Published data ('weight': 82, 'level': 17) to IBM Watson
Published data ('weight': 100, 'level': 64) to IBM Watson
Published data ('weight': 12, 'level': 76) to IBM Watson
Published data ('weight': 21, 'level': 22) to IBM Watson
Published data ('weight': 41, 'level': 71) to IBM Watson
Published data ('weight': 66, 'level': 93) to IBM Watson
Published data ('weight': 48, 'level': 9) to IBM Watson
Published data ('weight': 19, 'level': 38) to IBM Watson
Published data ('weight': 17, 'level': 31) to IBM Watson
Published data ('weight': 42, 'level': 81) to IBM Watson
Published data ('weight': 95, 'level': 19) to IBM Watson
```