

PYTHON SCRIPT TO PUBLISH DATA TO IBM CLOUD

Date	09 November 2022
Team ID	PNT2022TMID42279
Project Name	Project -Smart Waste Management System in Metropolitan Cities.
Maximum Marks	8 Marks

PYTHON CODE:

```
#IBM Watson IOT Platform
#pip install wiotp-sdk
import wiotp.sdk.device
import time
import random
myConfig = {
    "identity": {
        "orgId": "udgvx5",
        "typeId": "Level_Monitoring",
        "deviceId": "Python_Script"
    },
    "auth": {
        "token": "IBM_TEAM@123"
    }
}

def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s" %
    cmd.data['command'])
    m=cmd.data['command']

client = wiotp.sdk.device.DeviceClient(config=myConfig,
logHandlers=None)
client.connect()
while True:
    latitude=random.uniform(27.2046,125.25)
    longitude=random.uniform(77.4977,100.1526)
```

```

binlevel=random.randint(10,100)
if binlevel >= 90:
    myData={'latitude':latitude,'longitude':longitude,'binlevel':binlevel}
    client.publishEvent(eventId="status",
        msgFormat="json", data=myData, qos=0,
        onPublish=None)
    ##print("Published data Successfully: %s", myData)
    print("BIN IS FULL!!!!",myData)
    client.commandCallback = myCommandCallback
    time.sleep(2)
else :
    print("BIN IS IN NORMAL LEVEL")
    time.sleep(2)
client.disconnect()

```

PYTHON CODE:

```

pythonscript.py - C:/Users/GOMAL/Documents/IBM documents/Team Deliverables/pythonscript.py (3.9.6)
File Edit Format Run Options Window Help
#IBM Watson IoT Platform
#pip install wiotp-sdk
import wiotp.sdk.device
import time
import random
myConfig = {
    "identity": {
        "orgid": "udgvx5",
        "typeId": "Level_Monitoring",
        "deviceId": "Python_Script"
    },
    "auth": {
        "token": "IBM_TEAM@123"
    }
}
def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s" %
        cmd.data['command'])
    m=cmd.data['command']

client = wiotp.sdk.device.DeviceClient(config=myConfig,
    logHandlers=None)
client.connect()
while True:
    latitude=random.uniform(27.2046,125.25)
    longitude=random.uniform(77.4977,100.1526)
    binlevel=random.randint(10,100)
    if binlevel >= 90:
        myData={'latitude':latitude,'longitude':longitude,'binlevel':binlevel}
        client.publishEvent(eventId="status",
            msgFormat="json", data=myData, qos=0,
            onPublish=None)
        ##print("Published data Successfully: %s", myData)
        print("BIN IS FULL!!!!",myData)
        client.commandCallback = myCommandCallback
        time.sleep(2)
    else :
        print("BIN IS IN NORMAL LEVEL")
        time.sleep(2)
client.disconnect()

```

[illegible]

```
"IDLE Shell 3.9.6"
File Edit Shell Debug Options Window Help

Python 3.9.6 (tags/v3.9.6:db3ff76, Jun 28 2021, 15:26:21) [MSC v.1929 64 bit (AMD64)] on win
32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/GOMAL/Documents/IBM
documents/Team Deliverables/pythonscript.py =====
=====
2022-11-09 20:03:00.814 wiotp.sdk.device.client.DeviceClient INFO Connected successful
ly: dudygwX5:Level_Monitoring:Python_Script
BIN IS IN NORMAL LEVEL
BIN IS IN NORMAL LEVEL
BIN IS IN NORMAL LEVEL
BIN IS IN NORMAL LEVEL
BIN IS IN NORMAL LEVEL
BIN IS IN NORMAL LEVEL
BIN IS FULL!!!! ('latitude': 83.39636569909015, 'longitude': 95.20230389702733, 'binlevel':
96)
BIN IS IN NORMAL LEVEL
BIN IS IN NORMAL LEVEL
BIN IS IN NORMAL LEVEL
BIN IS IN NORMAL LEVEL
BIN IS IN NORMAL LEVEL
BIN IS IN NORMAL LEVEL
BIN IS IN NORMAL LEVEL
BIN IS IN NORMAL LEVEL
BIN IS IN NORMAL LEVEL
BIN IS IN NORMAL LEVEL
BIN IS FULL!!!! ('latitude': 121.69499757852562, 'longitude': 92.62011757871596, 'binlevel':
90)
BIN IS IN NORMAL LEVEL
BIN IS IN NORMAL LEVEL
BIN IS IN NORMAL LEVEL
BIN IS IN NORMAL LEVEL
BIN IS IN NORMAL LEVEL
BIN IS IN NORMAL LEVEL
BIN IS IN NORMAL LEVEL
BIN IS IN NORMAL LEVEL
BIN IS IN NORMAL LEVEL
BIN IS IN NORMAL LEVEL
BIN IS IN NORMAL LEVEL

pythonscript.py - C:/Users/GOMAL/Documents/IBM documents/Team Deliverables/pythonscript.py (3.9.6)
File Edit Format Run Options Window Help

[IBM Watson IoT Platform]
#pip install wiotp-sdk
import wiotp.sdk.device
import time
import random
myConfig = {
    "identity": {
        "orgId": "udgywX5",
        "typeId": "Level_Monitoring",
        "deviceId": "Python_Script"
    },
    "auth": {
        "token": "IBM_TEAM@123"
    }
}
def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s" %
cmd.data['command'])
m=cmd.data['command']

client = wiotp.sdk.device.DeviceClient(config=myConfig,
logHandlers=None)
client.connect()
while True:
    latitude=random.uniform(27.2046,125.25)
    longitude=random.uniform(77.4977,100.1526)
    binlevel=random.randint(10,100)
    if binlevel >= 90:
        myData={'latitude':latitude,'longitude':longitude,'binlevel':binlevel}
        client.publishEvent(eventId="status",
msgFormat="json", data=myData, qos=0,
onPublish=None)
##print("Published data Successfully: %s", myData)
print("BIN IS FULL!!!!",myData)
client.commandCallback = myCommandCallback
time.sleep(2)
else :
    print("BIN IS IN NORMAL LEVEL")
    time.sleep(2)
client.disconnect()
```

IBM WATSON OUTPUT:

The screenshot displays the IBM Watson IoT Platform interface. The top navigation bar includes 'Browse', 'Action', 'Device Types', and 'Interfaces'. A user profile is visible in the top right corner with the email 'g.guhar2202@gmail.com' and ID 'u1gpx5'. The main content area shows a device named 'Python_Script' with a status of 'Connected'. The 'Recent Events' tab is selected, displaying a table of events. The table has four columns: 'Event', 'Value', 'Format', and 'Last Received'. The events are all 'status' events, each containing a JSON object with 'latitude' and 'longitude' coordinates. The 'Last Received' column shows the time since the event was received, ranging from 'a few seconds ago' to '2 minutes ago'. The bottom status bar indicates '0 Simulations running'.

Event	Value	Format	Last Received
status	{"latitude":40.39402098307,"longitude":94.219...	json	a few seconds ago
status	{"latitude":65.65569502609739,"longitude":96....	json	a few seconds ago
status	{"latitude":116.25290465561541,"longitude":9...	json	a few seconds ago
status	{"latitude":121.69499757852562,"longitude":9...	json	a minute ago
status	{"latitude":83.39636569909015,"longitude":95....	json	2 minutes ago