

Develop the python code and subscribe to IBM IoT Platform

Date	25 October 2022
Team ID	PNT2022TMID54368
Project Name	Smart Farmer - IoT Enabled Smart Farming Application

Step 1: Import wiotp in python shell

```
The MIT Shell Does Not Have Admin Privs  
$ ./src 3.3.0 test/01test02.c, for 27 28 3, (4,2,2) 200 x,100.0 at 1.1 (MIPS) result 0  
Type 'help' or '?' for help.  
src> help, sleep  
src>
```

Step 2 : write the code to connect with IBM Watson platform

```
#!/usr/bin/env python
# coding: utf-8

import sys
import os
import random
import time
import json
import paho.mqtt.client as mqtt

# IBM Watson IoT Platform
api_key = 'api_key'
apikeyurl = 'apikeyurl'

myDevice = {
    "id": "12345",
    "type": "DeviceType",
    "deviceID": "12345"
}

def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
    cmd.data['command']
    if cmd.data['command'] == "on":
        print("Motor is switched on")
        # Call API to switch on motor
    elif cmd.data['command'] == "off":
        print("Motor is switched off")
        # Call API to switch off motor

client = mqtt.Client('DeviceType-12345-12345-12345-12345')
client.username = 'apikeyurl'
client.password = 'api_key'
client.connect('mqtt://iotplatform.com')

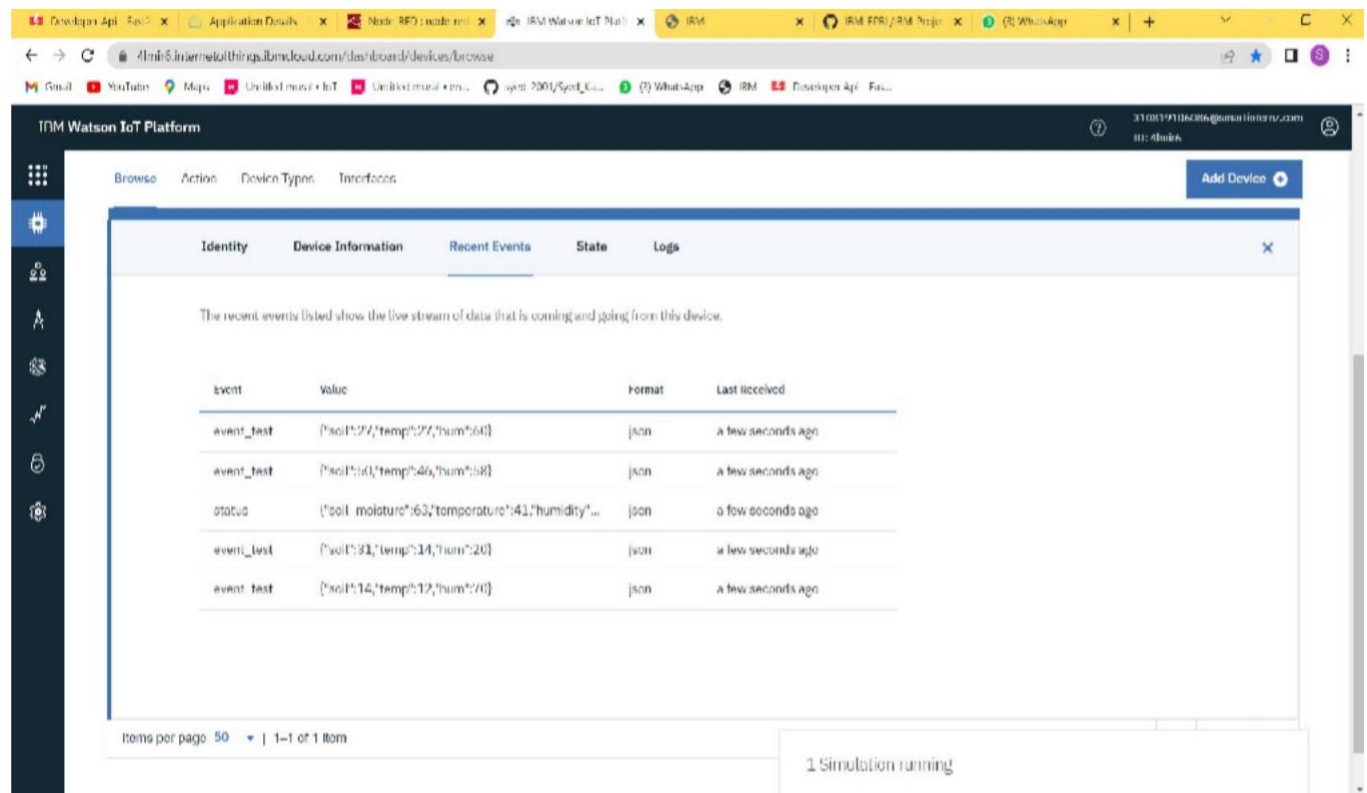
while True:
    # Send data to IBM IoT Platform
    data = {'command': 'on', 'deviceID': '12345'}
    client.publish('events/status', json.dumps(data))
    print("Published data successfully: %s" % data)
    time.sleep(10)
client.disconnect()
```

Step 3 : Python is connected with IBM watson and the result is shown in the console

```
Python 3.6.5 Shell
File Edit Shell Debug Console Window Help

Published data Successfully: 82 [{"soil_moisture": 58, "temperature": 52, "humidity": 17}]
Published data Successfully: 83 [{"soil_moisture": 6, "temperature": 107, "humidity": 54}]
Published data Successfully: 84 [{"soil_moisture": 73, "temperature": 24, "humidity": 7}]
Published data Successfully: 85 [{"soil_moisture": 93, "temperature": 88, "humidity": 18}]
Published data Successfully: 86 [{"soil_moisture": 57, "temperature": 1, "humidity": 52}]
Published data Successfully: 87 [{"soil_moisture": 44, "temperature": -14, "humidity": 73}]
Published data Successfully: 88 [{"soil_moisture": 90, "temperature": 149, "humidity": 59}]
Published data Successfully: 89 [{"soil_moisture": 54, "temperature": 76, "humidity": 73}]
Published data Successfully: 90 [{"soil_moisture": 10, "temperature": 17, "humidity": 19}]
Published data Successfully: 91 [{"soil_moisture": 35, "temperature": 43, "humidity": 96}]
Published data Successfully: 92 [{"soil_moisture": 29, "temperature": 76, "humidity": 21}]
Published data Successfully: 93 [{"soil_moisture": 21, "temperature": 27, "humidity": 18}]
Published data Successfully: 94 [{"soil_moisture": 10, "temperature": 73, "humidity": 38}]
Published data Successfully: 95 [{"soil_moisture": 4, "temperature": 72, "humidity": 51}]
Published data Successfully: 96 [{"soil_moisture": 10, "temperature": 139, "humidity": 56}]
Published data Successfully: 97 [{"soil_moisture": 65, "temperature": 2, "humidity": 42}]
Published data Successfully: 98 [{"soil_moisture": 19, "temperature": 73, "humidity": 88}]
Published data Successfully: 99 [{"soil_moisture": 54, "temperature": 21, "humidity": 53}]
Published data Successfully: 100 [{"soil_moisture": 47, "temperature": 33, "humidity": 63}]
Published data Successfully: 101 [{"soil_moisture": 92, "temperature": 1, "humidity": 33}]
Published data Successfully: 102 [{"soil_moisture": 73, "temperature": 96, "humidity": 18}]
Published data Successfully: 103 [{"soil_moisture": 0, "temperature": 21, "humidity": 92}]
Published data Successfully: 104 [{"soil_moisture": 58, "temperature": 98, "humidity": 87}]
Published data Successfully: 105 [{"soil_moisture": 70, "temperature": 26, "humidity": 77}]
Published data Successfully: 106 [{"soil_moisture": 34, "temperature": 2, "humidity": 35}]
Published data Successfully: 107 [{"soil_moisture": 63, "temperature": 11, "humidity": 8}]
Published data Successfully: 108 [{"soil_moisture": 71, "temperature": 92, "humidity": 90}]
Published data Successfully: 109 [{"soil_moisture": 109, "temperature": 114, "humidity": 16}]
Published data Successfully: 110 [{"soil_moisture": 27, "temperature": 76, "humidity": 55}]
Published data Successfully: 111 [{"soil_moisture": 12, "temperature": 76, "humidity": 51}]
Published data Successfully: 112 [{"soil_moisture": 87, "temperature": 119, "humidity": 74}]
Published data Successfully: 113 [{"soil_moisture": 7, "temperature": -13, "humidity": 58}]
Published data Successfully: 114 [{"soil_moisture": 53, "temperature": 1, "humidity": 85}]
Published data Successfully: 115 [{"soil_moisture": 40, "temperature": 124, "humidity": 21}]
Published data Successfully: 116 [{"soil_moisture": 29, "temperature": 94, "humidity": 41}]
Published data Successfully: 117 [{"soil_moisture": 58, "temperature": 8, "humidity": 88}]
Published data Successfully: 118 [{"soil_moisture": 90, "temperature": 21, "humidity": 26}]
Published data Successfully: 119 [{"soil_moisture": 25, "temperature": 96, "humidity": 99}]
Published data Successfully: 120 [{"soil_moisture": 8, "temperature": 21, "humidity": 89}]
Published data Successfully: 121 [{"soil_moisture": 66, "temperature": 2, "humidity": 90}]
Published data Successfully: 122 [{"soil_moisture": 90, "temperature": 80, "humidity": 5}]
Published data Successfully: 123 [{"soil_moisture": 47, "temperature": 36, "humidity": 11}]
Published data Successfully: 124 [{"soil_moisture": 30, "temperature": 37, "humidity": 85}]
Published data Successfully: 125 [{"soil_moisture": 71, "temperature": 36, "humidity": 95}]
Published data Successfully: 126 [{"soil_moisture": 14, "temperature": 36, "humidity": 51}]
Published data Successfully: 127 [{"soil_moisture": 28, "temperature": 54, "humidity": 25}]
```

Step 4 : As the python code is connected to IBM IoT platform ,then run the program.



The screenshot displays the IBM Watson IoT Platform interface. The top navigation bar includes tabs for 'Browse', 'Action', 'Device Types', and 'Interfaces'. A sidebar on the left contains icons for various platform features. The main content area shows a list of recent events for a device, with columns for 'event', 'Value', 'format', and 'Last received'. The events are listed in a table, showing data such as 'event_test' with values like '["soil":27,"temp":27,"hum":60]' and 'status' with values like '["soil_molature":63,"temperature":41,"humidity":...'. The table is paginated, showing 'Items per page: 50' and '1-1 of 1 item'. A status bar at the bottom indicates '1 Simulation running'.

event	Value	format	Last received
event_test	["soil":27,"temp":27,"hum":60]	json	a few seconds ago
event_test	["soil":30,"temp":40,"hum":38]	json	a few seconds ago
status	["soil_molature":63,"temperature":41,"humidity":...]	json	a few seconds ago
event_test	["soil":31,"temp":14,"hum":20]	json	a few seconds ago
event_test	["soil":14,"temp":12,"hum":70]	json	a few seconds ago

Result :

The Python Code is developed and Subscribed to IBM IoT Platform successfully.