

Develop the python code and subscribe to IBM IoT Platform

Date	03 October 2022
Team ID	PNT2022TMID14790
Project Name	SMART FARMER – IOT ENABLED SMART FARMING APPLICATIONS

Step 1: Import wiotp in python shell and write the code to connect with IBM Watson platform

```
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": "41mir6",
        "typeid": "TestDeviceType",
        "deviceId": "12345"
    },
    "auth": {
        "token": "dxV#N9UcZhSp4lc6+u"
    }
}

def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
    cmd = cmd.data['command']
    if (cmd == "motoron"):
        print("Motor is switched on")
    elif (cmd == "motorooff"):
        print("Motor is switched OFF")
    print(" ")

client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()

while True:
    soil=random.randint(0,100)
    temp=random.randint(-20,125)
    hum=random.randint(0,100)
    myData = {'soil_moisture':soil,'temperature':temp, 'humidity':hum}
    client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0, onPublish=None)
    print("Published data Successfully: %s" % myData)
    client.commandCallback = myCommandCallback
    time.sleep(2)
client.disconnect()
```

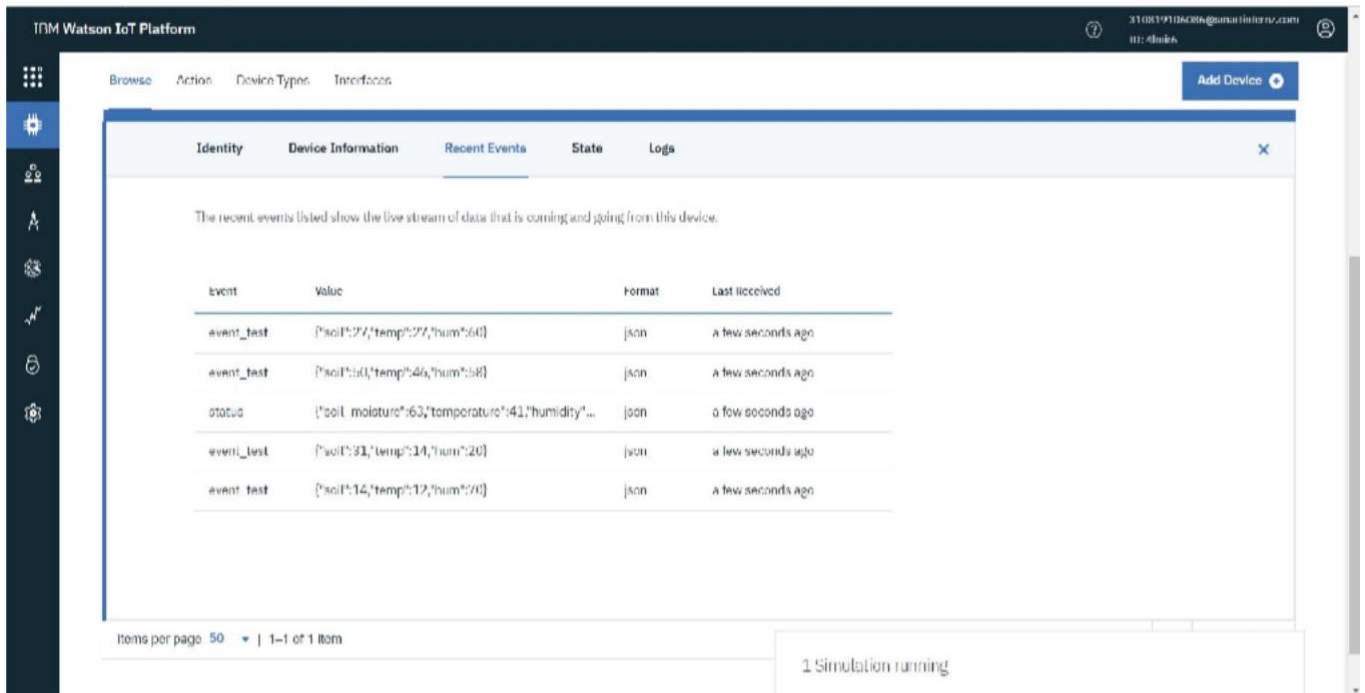
Step 2: Python is connected with IBM watson and the result is shown in the console

```
Python 2.7.10 Shell
File Edit Shell Debug Console Window Help
Published data Successfully: 82 {"soil_moisture": 58, "temperature": 52, "humidity": 13}
Published data Successfully: 83 {"soil_moisture": 6, "temperature": 101, "humidity": 54}
Published data Successfully: 84 {"soil_moisture": 73, "temperature": 24, "humidity": 71}
Published data Successfully: 85 {"soil_moisture": 83, "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": 78}
Published data Successfully: 88 {"soil_moisture": 90, "temperature": 149, "humidity": 49}
Published data Successfully: 89 {"soil_moisture": 84, "temperature": 76, "humidity": 33}
Published data Successfully: 90 {"soil_moisture": 14, "temperature": 17, "humidity": 10}
Published data Successfully: 91 {"soil_moisture": 35, "temperature": 40, "humidity": 96}
Published data Successfully: 92 {"soil_moisture": 29, "temperature": 70, "humidity": 21}
Published data Successfully: 93 {"soil_moisture": 21, "temperature": 27, "humidity": 88}
Published data Successfully: 94 {"soil_moisture": 10, "temperature": 70, "humidity": 18}
Published data Successfully: 95 {"soil_moisture": 4, "temperature": 72, "humidity": 51}
Published data Successfully: 96 {"soil_moisture": 10, "temperature": 15, "humidity": 50}
Published data Successfully: 97 {"soil_moisture": 65, "temperature": 2, "humidity": 42}
Published data Successfully: 98 {"soil_moisture": 17, "temperature": 77, "humidity": 88}
Published data Successfully: 99 {"soil_moisture": 54, "temperature": 21, "humidity": 58}
Published data Successfully: 100 {"soil_moisture": 47, "temperature": 33, "humidity": 83}
Published data Successfully: 101 {"soil_moisture": 92, "temperature": 1, "humidity": 38}
Published data Successfully: 102 {"soil_moisture": 73, "temperature": 96, "humidity": 18}
Published data Successfully: 103 {"soil_moisture": 0, "temperature": 31, "humidity": 88}
Published data Successfully: 104 {"soil_moisture": 58, "temperature": 88, "humidity": 83}
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": 61, "temperature": 11, "humidity": 81}
Published data Successfully: 108 {"soil_moisture": 71, "temperature": 51, "humidity": 90}
Published data Successfully: 109 {"soil_moisture": 10, "temperature": 114, "humidity": 14}
Published data Successfully: 110 {"soil_moisture": 27, "temperature": 76, "humidity": 50}
Published data Successfully: 111 {"soil_moisture": 12, "temperature": 19, "humidity": 91}
Published data Successfully: 112 {"soil_moisture": 67, "temperature": 119, "humidity": 74}
Published data Successfully: 113 {"soil_moisture": 7, "temperature": -13, "humidity": 59}
Published data Successfully: 114 {"soil_moisture": 59, "temperature": 1, "humidity": 95}
Published data Successfully: 115 {"soil_moisture": 46, "temperature": 124, "humidity": 31}
Published data Successfully: 116 {"soil_moisture": 29, "temperature": 14, "humidity": 84}
Published data Successfully: 117 {"soil_moisture": 58, "temperature": 8, "humidity": 88}
Published data Successfully: 118 {"soil_moisture": 70, "temperature": 21, "humidity": 30}
Published data Successfully: 119 {"soil_moisture": 25, "temperature": 96, "humidity": 99}
Published data Successfully: 120 {"soil_moisture": 6, "temperature": 11, "humidity": 69}
Published data Successfully: 121 {"soil_moisture": 60, "temperature": 2, "humidity": 94}
Published data Successfully: 122 {"soil_moisture": 70, "temperature": 80, "humidity": 51}
Published data Successfully: 123 {"soil_moisture": 42, "temperature": 36, "humidity": 11}
Published data Successfully: 124 {"soil_moisture": 38, "temperature": 17, "humidity": 88}
Published data Successfully: 125 {"soil_moisture": 71, "temperature": 34, "humidity": 95}
Published data Successfully: 126 {"soil_moisture": 14, "temperature": 38, "humidity": 51}
Published data Successfully: 127 {"soil_moisture": 45, "temperature": 16, "humidity": 25}
```

Step 3: As the python code is connected to IBM IoT platform ,then run theprogram.

Result:

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



The screenshot displays the IBM Watson IoT Platform interface. The top navigation bar includes 'Browse', 'Action', 'Device Types', and 'Interfaces'. The main content area is titled 'Recent Events' and shows a table of events. The table has four columns: 'Event', 'Value', 'Format', and 'Last received'. The events listed are 'event_test' with JSON values for soil, temperature, and humidity, and 'status' with a truncated JSON value. The 'Last received' column indicates that all events were received 'a few seconds ago'. A status bar at the bottom right shows '1 Simulation running'.

Event	Value	Format	Last received
event_test	{\"soil\":27,\"temp\":27,\"hum\":64}	json	a few seconds ago
event_test	{\"soil\":31,\"temp\":46,\"hum\":58}	json	a few seconds ago
status	{\"soil_moisture\":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\":90}	json	a few seconds ago