

Assignment –3

NAME – SARANSH PRATAP SINGH

REGISTRATION NUMBER – 19BCY10035

Q- Develop a code to upload the water tank level and light intensity values to the IBM IoT platform and visualize them in the web application.

```
import wiotp.sdk.device
import time
import random
myConfig = {
    "identity": {
        "orgId": "2nyph2",
        "typeId": "Device",
        "deviceId": "12345"
    },
    "auth": {
        "token": "123456789"
    }
}

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:
    temp=random.randint(-20,125)
    hum=random.randint(0,100)
    myData={'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()

```

The screenshot shows the Visual Studio Code interface with a file named `WateTankLevel.py` open. The code defines a `myCommandCallback` function and a `while True` loop that generates random temperature and humidity data, publishes it as JSON events, and prints the data. The terminal output shows several successful publications with varying values for temperature and humidity.

```

1  import wiotp.sdk.device
2  import time
3  import random
4  myConfig = {
5      "identity": {
6          "orgId": "2myph2",
7          "typeId": "Device",
8          "deviceId": "12345"
9      },
10     "auth": {
11         "token": "123456789"
12     }
13 }
14
15 def myCommandCallback(cmd):
16     print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
17     #cmd.data['command']
18
19 client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
20 client.connect()
21
22 while True:
23     temp=random.randint(-20,125)
24     hum=random.randint(0,100)
25     myData={'temperature':temp, 'humidity':hum}
26     client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0, onPublish=None)
27     print("Published data Successfully: %s", myData)
28     time.sleep(2)
29 client.disconnect()

```

Terminal Output:

```

Published data Successfully: %s {'temperature': 28, 'humidity': 98}
Published data Successfully: %s {'temperature': 45, 'humidity': 23}
Published data Successfully: %s {'temperature': 3, 'humidity': 46}
Published data Successfully: %s {'temperature': -6, 'humidity': 88}
Published data Successfully: %s {'temperature': 51, 'humidity': 68}
Published data Successfully: %s {'temperature': 95, 'humidity': 7}
Published data Successfully: %s {'temperature': 5, 'humidity': 17}
^CTraceback (most recent call last):
  File "/home/saransh/IOT/Assignment-3/WateTankLevel.py", line 29, in <module>
    time.sleep(2)
KeyboardInterrupt

```

This screenshot is similar to the one above, showing the same code and terminal output. The code in the editor is identical, and the terminal shows the same sequence of published data points and a keyboard interrupt.

```

5  "identity": {
6      "orgId": "2myph2",
7      "typeId": "Device",
8      "deviceId": "12345"
9  },
10  "auth": {
11      "token": "123456789"
12  }
13 }
14
15 def myCommandCallback(cmd):
16     print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
17     #cmd.data['command']
18
19 client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
20 client.connect()
21
22 while True:
23     temp=random.randint(-20,125)
24     hum=random.randint(0,100)
25     myData={'temperature':temp, 'humidity':hum}
26     client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0, onPublish=None)
27     print("Published data Successfully: %s", myData)
28     client.commandCallback = myCommandCallback
29     time.sleep(2)
30 client.disconnect()

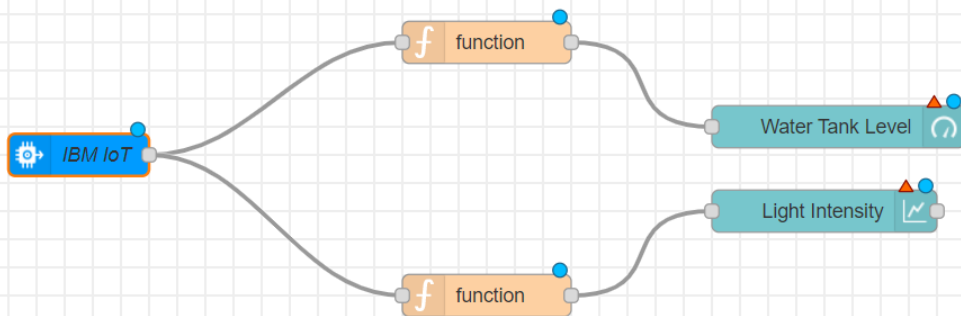
```

Terminal Output:

```

Published data Successfully: %s {'temperature': 28, 'humidity': 98}
Published data Successfully: %s {'temperature': 45, 'humidity': 23}
Published data Successfully: %s {'temperature': 3, 'humidity': 46}
Published data Successfully: %s {'temperature': -6, 'humidity': 88}
Published data Successfully: %s {'temperature': 51, 'humidity': 68}
Published data Successfully: %s {'temperature': 95, 'humidity': 7}
Published data Successfully: %s {'temperature': 5, 'humidity': 17}
^CTraceback (most recent call last):
  File "/home/saransh/IOT/Assignment-3/WateTankLevel.py", line 29, in <module>
    time.sleep(2)
KeyboardInterrupt

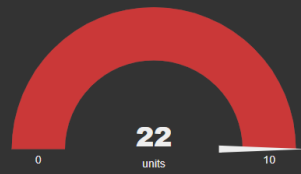
```



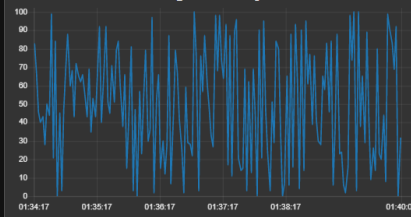
<input type="checkbox"/>	Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location	
▼	12345	Disconnected	Device	Device	12 Dec 2021 23:38		→ ...
Identity Device Information <u>Recent Events</u> State Logs ×							
The recent events listed show the live stream of data that is coming and going from this device.							
Event	Value	Format	Last Received				
status	{"temperature":12,"humidity":92}	json	a few seconds ago				
status	{"temperature":-14,"humidity":78}	json	a few seconds ago				
status	{"temperature":82,"humidity":50}	json	a few seconds ago				
status	{"temperature":40,"humidity":13}	json	a few seconds ago				
status	{"temperature":31,"humidity":61}	json	a few seconds ago				

Assignment-3

Water Tank Level

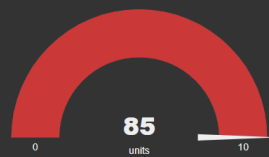


Light Intensity



Assignment-3

Water Tank Level



Light Intensity

