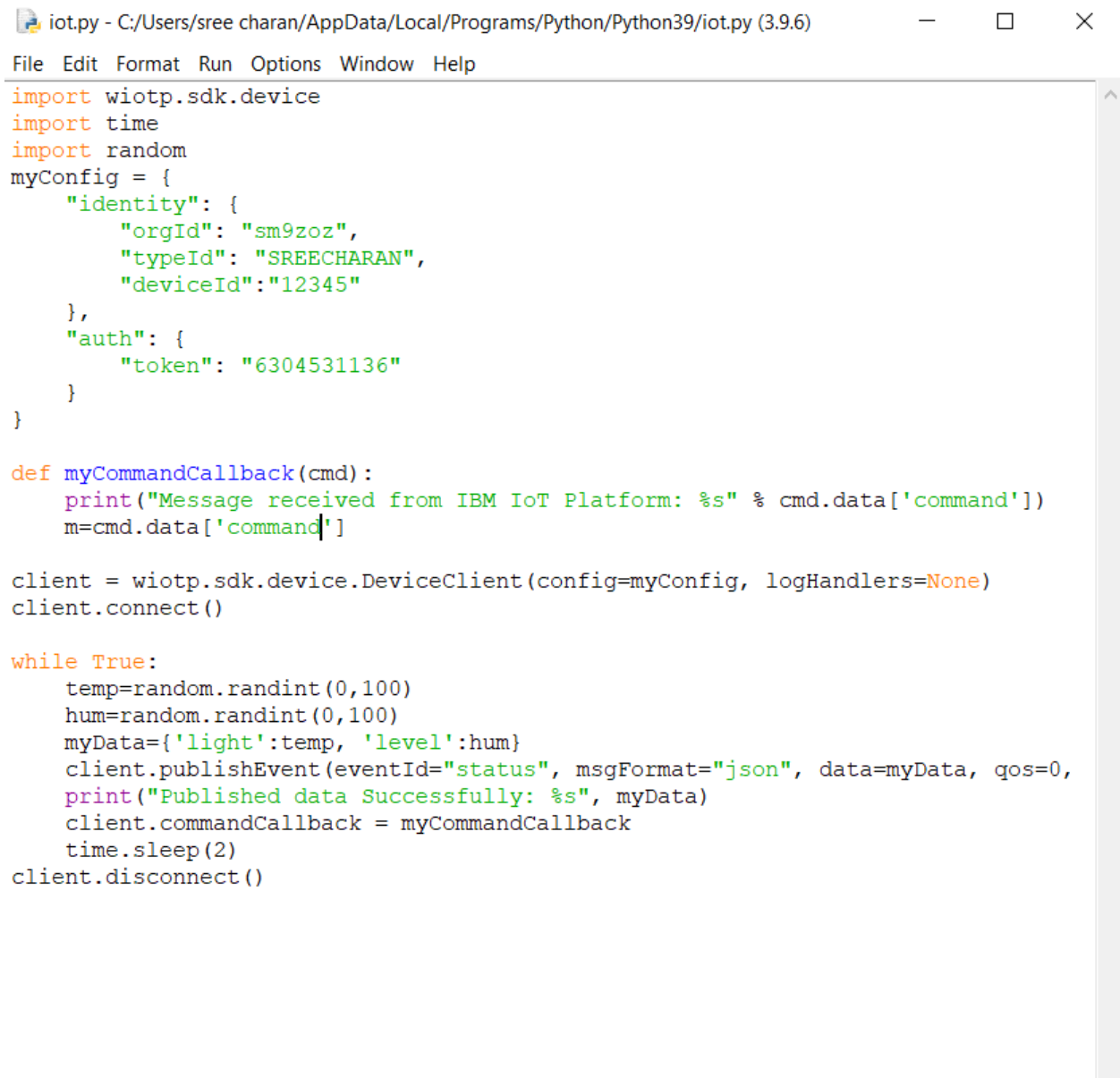


## Assignment 3

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.

A screenshot of a Python IDE window titled 'iot.py - C:/Users/sree charan/AppData/Local/Programs/Python/Python39/iot.py (3.9.6)'. The window has a menu bar with 'File', 'Edit', 'Format', 'Run', 'Options', 'Window', and 'Help'. The code is written in Python and uses the 'wiotpsdk' library. It defines a configuration dictionary 'myConfig' with fields for 'identity' (orgId, typeId, deviceId) and 'auth' (token). A callback function 'myCommandCallback' is defined to print received messages. The main logic creates a 'DeviceClient' and enters a loop where it generates random temperature and humidity values, publishes them as JSON events with eventId 'status', and sleeps for 2 seconds before disconnecting and reconnecting.

```
import wiotpsdk.device
import time
import random
myConfig = {
    "identity": {
        "orgId": "sm9zoz",
        "typeId": "SREECHARAN",
        "deviceId": "12345"
    },
    "auth": {
        "token": "6304531136"
    }
}

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

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

while True:
    temp=random.randint(0,100)
    hum=random.randint(0,100)
    myData={'light':temp, 'level':hum}
    client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0,
    print("Published data Successfully: %s", myData)
    client.commandCallback = myCommandCallback
    time.sleep(2)
client.disconnect()
```

CODE:

```
import wiotpsdk.device
import time
```

```

import random
myConfig = {
    "identity": {
        "orgId": "sm9zoz",
        "typeId": "SREECHARAN",
        "deviceId": "12345"
    },
    "auth": {
        "token": "6304531136"
    }
}

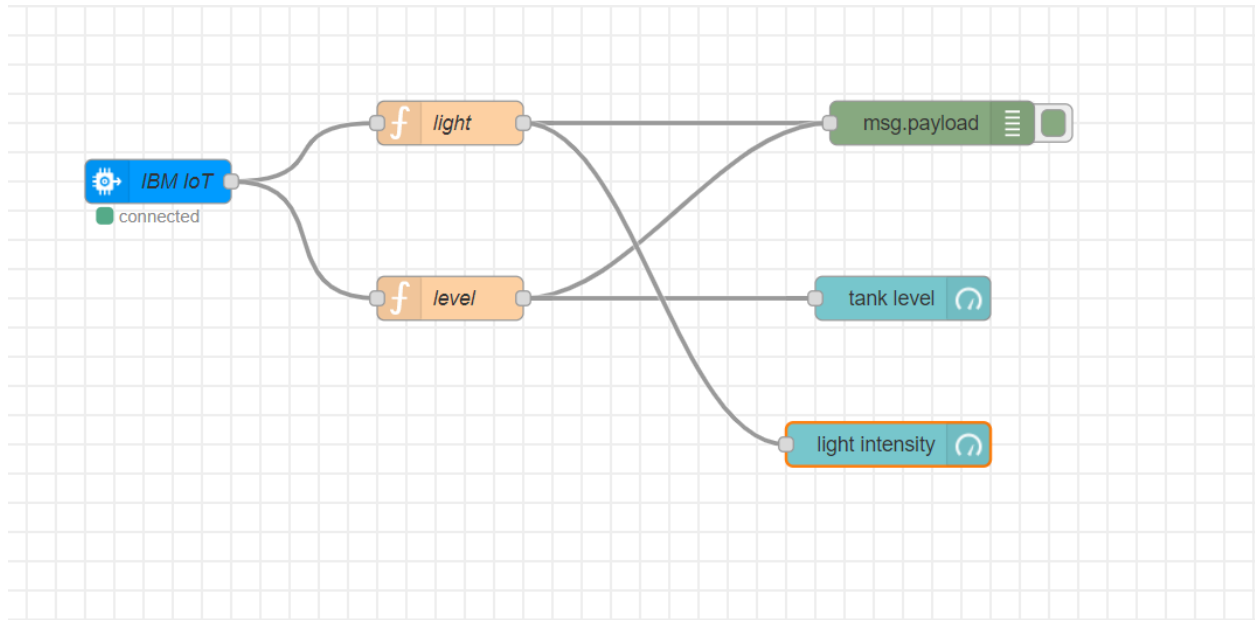
def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s" %
cmd.data['command'])
    m=cmd.data['command']

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

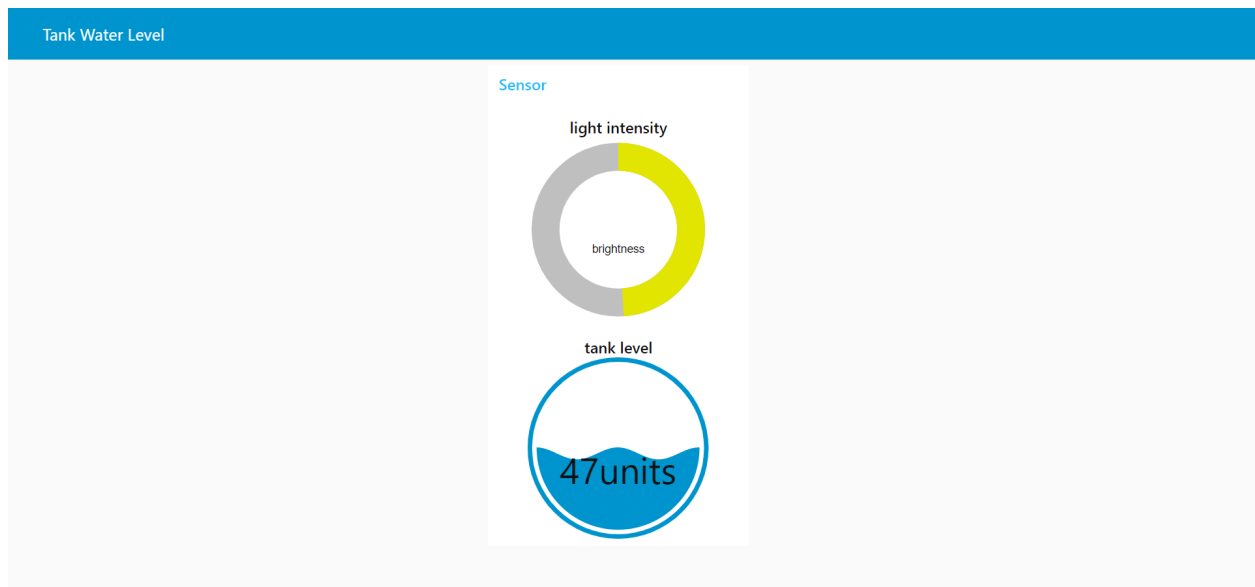
while True:
    temp=random.randint(0,100)
    hum=random.randint(0,100)
    myData={'light':temp, 'level':hum}
    client.publishEvent(eventId="status", msgFormat="json",
data=myData, qos=0, onPublish=None)
    print("Published data Successfully: %s", myData)
    client.commandCallback = myCommandCallback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

```

Node flow:



Output received in web page:



Tank Water Level

Sensor

light intensity

brightness

tank level

19units

Published data Successfully: %s ('light': 4, 'level': 51)

Published data Successfully: %s ('light': 46, 'level': 70)

Published data Successfully: %s ('light': 90, 'level': 50)

Published data Successfully: %s ('light': 17, 'level': 74)

Published data Successfully: %s ('light': 40, 'level': 42)

Published data Successfully: %s ('light': 26, 'level': 65)

Published data Successfully: %s ('light': 85, 'level': 27)

Published data Successfully: %s ('light': 57, 'level': 61)

Published data Successfully: %s ('light': 20, 'level': 23)

Published data Successfully: %s ('light': 75, 'level': 71)

Published data Successfully: %s ('light': 42, 'level': 66)

Published data Successfully: %s ('light': 58, 'level': 35)

Published data Successfully: %s ('light': 34, 'level': 48)

Published data Successfully: %s ('light': 5, 'level': 12)

Published data Successfully: %s ('light': 12, 'level': 35)

Published data Successfully: %s ('light': 57, 'level': 52)

Published data Successfully: %s ('light': 81, 'level': 66)

Published data Successfully: %s ('light': 69, 'level': 1)

Published data Successfully: %s ('light': 0, 'level': 32)

Published data Successfully: %s ('light': 60, 'level': 46)

Published data Successfully: %s ('light': 97, 'level': 95)

Published data Successfully: %s ('light': 31, 'level': 99)

Published data Successfully: %s ('light': 48, 'level': 89)

Published data Successfully: %s ('light': 3, 'level': 20)

Published data Successfully: %s ('light': 46, 'level': 9)

Published data Successfully: %s ('light': 43, 'level': 32)

Published data Successfully: %s ('light': 45, 'level': 96)

Published data Successfully: %s ('light': 77, 'level': 5)

Published data Successfully: %s ('light': 21, 'level': 86)

Published data Successfully: %s ('light': 53, 'level': 3)

Published data Successfully: %s ('light': 60, 'level': 10)

Published data Successfully: %s ('light': 16, 'level': 14)

Published data Successfully: %s ('light': 76, 'level': 99)

Published data Successfully: %s ('light': 79, 'level': 74)

Published data Successfully: %s ('light': 16, 'level': 70)

Published data Successfully: %s ('light': 4, 'level': 34)

Published data Successfully: %s ('light': 20, 'level': 41)

Published data Successfully: %s ('light': 96, 'level': 82)

Published data Successfully: %s ('light': 52, 'level': 19)