

# SPRINT 1

## TEAM ID: PNT2022TMID23156

### REAL TIME RIVER-WATER QUALITY MONITORING AND CONTROL SYSTEM

#### PYTHON CODE:

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random

#Provide your IBM Watson Device Credentials
organization = "no6686"
deviceType = "nodeMCU"
deviceId = "123"
authMethod = "use-token-auth"
authToken = "12345678"

def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="motoron":
        print("motor is on")
    elif status=="motoroff":
        print("motor is off")
    else:
        print("please send proper command")

try:
    deviceOptions = {"org": organization, "type": deviceType, "id": deviceId,
"auth-method":authMethod, "auth-token":authToken}
    deviceCli= ibmiotf.device.Client(deviceOptions)

#.. except Exception
as e:
    print("Caught evention connecting device: %s" % str(e))
    sys.exit()
```

```

deviceCli.connect() while True:
temp=random.randint (90,110)
Humid=random.randint (60,100)
    Ph=random.randint (0,14)
    Water_turbidity=random.randint (15,60) data = {'temp' : temp,
        'Humid': Humid, 'Ph' : Ph, 'Water_turbidity':
Water_turbidity} def
    myonPublishCallback():
        print ("Published Temperature = %s C" % temp, "Humidity = %s %" %
Humid,"Ph = %s" % Ph,"Water Turbidity = %s NTU" % Water_turbidity, "to
IBM Watson") success = deviceCli.publishEvent("IoTSensor", "json",
        data, qos=0,
on_publish = myonPublishCallback)
    if not success: print("Not
connected to IOTF")
        time.sleep (10)
        deviceCli.commandCallback = myCommandCallback
deviceCli.disconnect()

```

\*Python 3.7.0 Shell\*

File Edit Shell Debug Options Window Help

```
Watson
Published Temperature = 98 C Humidity=94% Ph = 1 Water Turbidity = 23 NTU toIBM
Watson
Published Temperature = 105 C Humidity=96% Ph = 11 Water Turbidity = 15 NTU toIBM
M Watson
Published Temperature = 96 C Humidity=79% Ph = 2 Water Turbidity = 60 NTU toIBM
Watson
Published Temperature = 107 C Humidity=97% Ph = 9 Water Turbidity = 31 NTU toIBM
Watson
Published Temperature = 105 C Humidity=74% Ph = 3 Water Turbidity = 28 NTU toIBM
Watson
Published Temperature = 101 C Humidity=80% Ph = 10 Water Turbidity = 59 NTU toIBM
M Watson
Published Temperature = 103 C Humidity=94% Ph = 6 Water Turbidity = 23 NTU toIBM
Watson
Published Temperature = 90 C Humidity=62% Ph = 8 Water Turbidity = 56 NTU toIBM
Watson
Published Temperature = 100 C Humidity=60% Ph = 11 Water Turbidity = 47 NTU toIBM
M Watson
Published Temperature = 94 C Humidity=91% Ph = 0 Water Turbidity = 59 NTU toIBM
Watson
Published Temperature = 90 C Humidity=98% Ph = 13 Water Turbidity = 57 NTU toIBM
Watson
Published Temperature = 108 C Humidity=94% Ph = 4 Water Turbidity = 16 NTU toIBM
Watson
Published Temperature = 100 C Humidity=60% Ph = 14 Water Turbidity = 32 NTU toIBM
M Watson
Published Temperature = 101 C Humidity=63% Ph = 2 Water Turbidity = 20 NTU toIBM
Watson
Published Temperature = 107 C Humidity=85% Ph = 8 Water Turbidity = 23 NTU toIBM
Watson
Published Temperature = 92 C Humidity=69% Ph = 11 Water Turbidity = 60 NTU toIBM
Watson
Published Temperature = 105 C Humidity=91% Ph = 9 Water Turbidity = 19 NTU toIBM
Watson
Published Temperature = 106 C Humidity=65% Ph = 9 Water Turbidity = 25 NTU toIBM
Watson
Published Temperature = 104 C Humidity=96% Ph = 12 Water Turbidity = 54 NTU toIBM
```

Device Simulator ☒  

Device ID

Status

Device Type

Class ID

Date Added

Descriptive Location

▼

123

Connected

NodeMCU

Device

Nov 16, 2022 12:01 AM

→

...

Identity

Device Information

Recent Events

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

IoTSensor

{"temp":110,"Humid":60,"Ph":8,"Water\_turbidity..."}

json

a few seconds ago

IoTSensor

{"temp":108,"Humid":81,"Ph":1,"Water\_turbidity..."}

json

a few seconds ago

IoTSensor

{"temp":106,"Humid":75,"Ph":10,"Water\_turbidit..."}

json

a few seconds ago

IoTSensor

{"temp":101,"Humid":87,"Ph":1,"Water\_turbidity..."}

json

a few seconds ago

IoTSensor

{"temp":103,"Humid":60,"Ph":12,"Water\_turbidit..."}

json

a few seconds ago



CODD2.py - C:/Users/91701/AppData/Local/Programs/Python/Python37/CODD2.py (3.7.0)

File Edit Format Run Options Window Help

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random

#Provide your IBM Watson Device Credentials
organization = "no6686"
deviceType = "NodeMCU"
deviceId = "123"
authMethod = "token"
authToken = "12345678"

def myCommandCallback (cmd):
    print ("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status== "motoron":
        print ("motor is on")
    elif status == "motoroff":
        print ("motor is off")
    else:
        print ("please send proper command")

try:
    deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "auth-method": authMethod, "auth-token": authToken}
    deviceCli= ibmiotf.device.Client (deviceOptions)
    #..
except Exception as e:
    print ("Caught evention connecting device: %s" %str(e))
    sys.exit()
deviceCli.connect()
while True:
    temp=random.randint (90,110)
    Humid=random.randint (60,100)
    Ph=random.randint (0,14)
    Water_turbidity=random.randint (15,60)
    data = {'temp' : temp, 'Humid': Humid, 'Ph' : Ph, 'Water_turbidity':Water_turbidity}
    def myonPublishCallback():
        print ("Published Temperature = %s C" %temp, "Humidity=%s%%"%Humid,"Ph = %s" % Ph,"Water Turbidity = %s NTU" %Water_turbidity,"toIBM Watson")
    success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0,on_publish = myonPublishCallback)
    if not success:
        print("Not connected to IOTF")
        time.sleep (10)
        deviceCli.commandCallback = myCommandCallback
deviceCli.disconnect()
```