REAL-TIME RIVER WATER QUALITY MONITORING AND CONTROL SYSTEM

SPRINT-1

PYTHON SOURCE CODE:

```
import ibmiotf.application
import ibmiotf.device
import time
import random
import sys
import requests
import json
import urllib.request
import urllib.parse
url="https://www.fast2sms.com/dev/bulkV2"
organization = "swz5ou"
deviceType = "abcd"
deviceId = "12"
authMethod = "token"
authToken = "12345678"


def sms(ph,temp,turbidity):
   message='Water quality degraded PH value:'+str(ph)+'temperature value:'+str(temp)+'tubidity value:'+str(turbidity)
   my_data = {
      'sender_id': 'TXTIND',
      'message': message,
      'language': 'english',
      'route': 'p',
      'numbers': '9150661026, 6369521344,9840981094'
   }
   headers = {
```

```
        'authorization':
'cjshq2uY05KWVOxSDndGMNyvAmR6rgzfUpI3Pe8JkE49ZXlBbwq2plfEB6IZ31CjywSchzNtRQkixoV0',
        'Content-Type': "application/x-www-form-urlencoded",
        'Cache-Control': "no-cache"
    }
    response = requests.request("POST",url,data=my_data,headers=headers)
    returned_msg = json.loads(response.text)
    print(returned_msg['message'])


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 exception connecting device: %s" % str(e))
    sys.exit()


deviceCli.connect()


while True:


    pH = random.randint(1, 14)
    turbidity = random.randint(1, 1000)
    temperature = random.randint(0, 100)


    if pH<6 or temperature >120 or turbidity > 500:
        alert = 1
    else:
        alert = 0


    data = {'pH': pH, 'turbid': turbidity, 'temp': temperature, 'alert':alert}
```

```
    def myOnPublishCallback():
        print("Published pH= %s" % pH, "Turbidity:%s" % turbidity, "Temperature:%s" % temperature)
    success = deviceCli.publishEvent("water monitoring", "json", data, qos=0, on_publish=myOnPublishCallback)
    if not success:
        print("Not Connected to ibmiot")
    time.sleep(1)

deviceCli.disconnect()
```

OUTPUT:

```
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) [MSC v.1914 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: C:\IBM\new source code.py ==========
2022-11-19 11:27:36,018  ibmiotf.device.Client  INFO  Connected successfully: d:swz5ou:abcd:12
Published pH= 8 Turbidity:270 Temperature:75
Published pH= 5 Turbidity:7 Temperature:83
Published pH= 3 Turbidity:931 Temperature:11
Published pH= 2 Turbidity:657 Temperature:7
Published pH= 7 Turbidity:410 Temperature:47
Published pH= 14 Turbidity:353 Temperature:25
Published pH= 3 Turbidity:968 Temperature:60
Published pH= 12 Turbidity:751 Temperature:86
Published pH= 3 Turbidity:80 Temperature:89
Published pH= 4 Turbidity:814 Temperature:41
Published pH= 1 Turbidity:401 Temperature:96
Published pH= 9 Turbidity:389 Temperature:53
Published pH= 4 Turbidity:294 Temperature:9
Published pH= 11 Turbidity:838 Temperature:76
Published pH= 6 Turbidity:739 Temperature:60
Published pH= 12 Turbidity:661 Temperature:44
Published pH= 13 Turbidity:744 Temperature:31
Published pH= 11 Turbidity:963 Temperature:13
Published pH= 12 Turbidity:117 Temperature:93
Published pH= 12 Turbidity:855 Temperature:39
Published pH= 4 Turbidity:242 Temperature:100
Published pH= 4 Turbidity:190 Temperature:16
Published pH= 12 Turbidity:509 Temperature:92
Published pH= 14 Turbidity:50 Temperature:93
Published pH= 8 Turbidity:688 Temperature:71
Published pH= 7 Turbidity:969 Temperature:52
Published pH= 8 Turbidity:517 Temperature:89
Published pH= 8 Turbidity:461 Temperature:83
```