PREREQUISITES

SOFTWARE

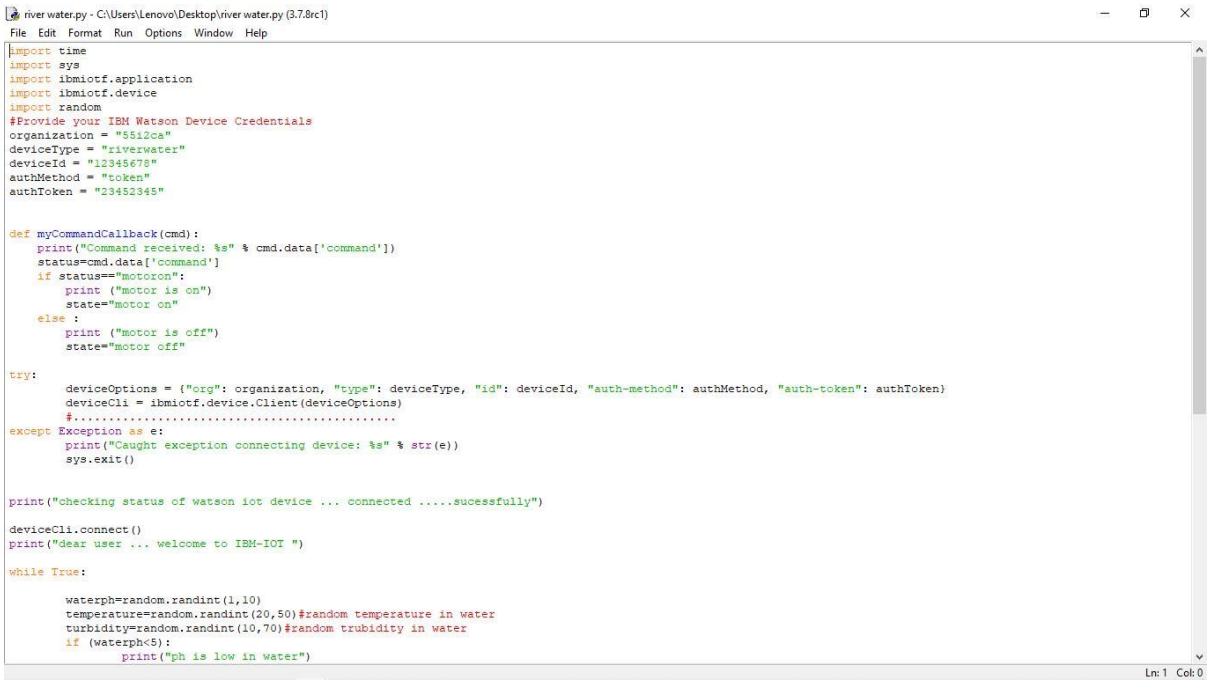
TEAM ID	PNT2022TMID41301
PROJECT NAME	Real-Time River Water Quality Monitoring and Control System

SOFTWARE

Python script:

I. We create a python code to detect temperature, turbidity and ph values of the river water.

II. Send the status of temperature,turbidity , ph values to the IBM Watson using python script



river water.py - C:\Users\Lenovo\Desktop\river water.py (3.7.8rc1)

File Edit Format Run Options Window Help

```
print("checking status of watson iot device ... connected .....sucessfully")

deviceCli.connect()
print("dear user ... welcome to IBM-IOT ")

while True:

        waterph=random.randint(1,10)
        temperature=random.randint(20,50)#random temperature in water
        turbidity=random.randint(10,70)#random trubidity in water
        if (waterph<5):
                print("ph is low in water")
                waterphstatus="low ph ,bad water"
        elif(waterph>5)and(waterph<7):
                print("normal ph in water")
                waterphstatus="good ph,good water"
        else:
                print("normal ph in water")
                waterphstatus="high ph,bad water"
        if (turbidity<30):
                print("turbidity is low in water")
                turbiditystatus="low turbidity , dust particles is low"
        elif( turbidity>30)and(waterph<7):
                print("normal turbidity in water")
                turbiditystatus="good turbidity, dust particles is medium "
        else:
                print("normal turbidity in water")
                turbiditystatus="high turbidity,dust particles is more "
        data = { 'temp' : temperature,'turb':turbidity,'ph':waterph,'waterphstatus':waterphstatus,'turbiditystatus':turbiditystatus}
        #print data
        def myOnPublishCallback():
            print ("Published Temperature = %s C" % temperature,"turbidity = %s %%" % turbidity,"waterph = %s %%" % waterph )
        success = deviceCli.publishEvent("espwatermodule", "json", data, qos=0, on_publish=myOnPublishCallback)
        if not success:
            print("Not connected to IoTF")
        time.sleep(5)
        deviceCli.commandCallback = myCommandCallback


# Disconnect the device and application from the cloud
deviceCli.disconnect()
```

Ln: 1 Col: 0

Python 3.7.8rc1 Shell

File Edit Shell Debug Options Window Help

```
Python 3.7.8rc1 (tags/v3.7.8rc1:5f3933d61d, Jun 17 2020, 16:59:29) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
================ RESTART: C:\Users\Lenovo\Desktop\river water.py ================
checking status of watson iot device ... connected .....sucessfully
dear user ... welcome to IBM-IOT
ph is low in water
turbidity is low in water
2022-11-11 18:53:59,943   ibmiotf.device.Client      INFO    Connected successfully: d:55i2ca:riverwater:12345678
Published Temperature = 48 C turbidity = 26 % waterph = 1 %
ph is low in water
turbidity is low in water
Published Temperature = 50 C turbidity = 14 % waterph = 2 %
ph is low in water
normal turbidity in water
Published Temperature = 24 C turbidity = 43 % waterph = 4 %
normal ph in water
normal turbidity in water
Published Temperature = 35 C turbidity = 44 % waterph = 9 %
normal ph in water
turbidity is low in water
Published Temperature = 25 C turbidity = 24 % waterph = 9 %
normal ph in water
turbidity is low in water
Published Temperature = 20 C turbidity = 27 % waterph = 8 %
normal ph in water
turbidity is low in water
Published Temperature = 20 C turbidity = 28 % waterph = 5 %
ph is low in water
turbidity is low in water
Published Temperature = 39 C turbidity = 27 % waterph = 3 %
normal ph in water
turbidity is low in water
Published Temperature = 45 C turbidity = 13 % waterph = 7 %
```

Ln: 5 Col: 0