

SPRINT -1

Date	29 October 2022
Team id	PNT2022TMID01284
Project name	Real Time River Water Quality Monitoring And Control System
Maximum marks	20 marks

In Sprint – 1, we have done simulation creation. We connected the sensor Arduino with python code

PYTHON CODE :

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

#Provide your IBM Watson Device Credentials

```
organization = "ofq2bm"
deviceType = "water_monitoring"
deviceId = "water_quality"
authMethod = "token"
authToken = "YC9348Ol6xz(Pqb7pL"
```

Initialize GPIO

```
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="lighton":
        print ("led is on")
    elif status == "lightoff":
        print ("led 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 exception connecting device: %s" % str(e))
    sys.exit()

# Connect and send a datapoint "hello" with value "world" into the cloud as an event of type
"greeting" 10 times
deviceCli.connect()

while True:
    #Get Sensor Data from DHT11

    turbidity=random.randint(0,110)
    pHLevel=random.randint(0,10)
    temperature = random.randint(0,110)

    data = { 'turbidity' : turbidity, 'pHLevel': pHLevel , 'temperature':temperature }
    #print data
    def myOnPublishCallback():
        print ("Published Turbidity = %s NTU" % turbidity, ", " "pH Level = %s " % pHLevel, ", "
"Temperature = %s °C"% temperature, "to IBM Watson")

    success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0,
on_publish=myOnPublishCallback)
    if not success:
        print("Not connected to IoT")
        time.sleep(10)

    deviceCli.commandCallback = myCommandCallback

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

```

lot Sensor is connected and data is published.

The screenshot displays the IBM Watson IoT Platform interface. The top navigation bar includes 'Browse', 'Action', 'Device Types', and 'Interfaces'. A search bar is present with the text 'Search by Device ID'. The main content area shows a table of devices. The first device, 'water_quality', is in a 'Connected' state. Below the device list, a detailed view for 'water_quality' is shown, including tabs for 'Identity', 'Device Information', 'Recent Events', 'State', and 'Logs'. The 'Recent Events' tab is active, displaying a list of events from the 'IoTSensor' device. Each event contains a JSON payload with 'turbidity', 'pHLevel', and 'temperature' values. A status message at the bottom right indicates '1 Simulation running'.

Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
water_quality	Connected	water_monitoring	Device	Nov 17, 2022 9:48 AM	

Event	Value	Format	Last Received
IoTSensor	{"turbidity":103,"pHLevel":1,"temperature":38}	json	a few seconds ago
IoTSensor	{"turbidity":94,"pHLevel":8,"temperature":94}	json	a few seconds ago
IoTSensor	{"turbidity":106,"pHLevel":9,"temperature":6}	json	a few seconds ago
IoTSensor	{"turbidity":74,"pHLevel":7,"temperature":100}	json	a few seconds ago
IoTSensor	{"turbidity":37,"pHLevel":4,"temperature":68}	json	a few seconds ago

1 Simulation running