SPRINT -1

|  |  |
| --- | --- |
| Date | November 2022 |
| Team id | PNT2022TMID11798 |
| 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, "authmethod": 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)

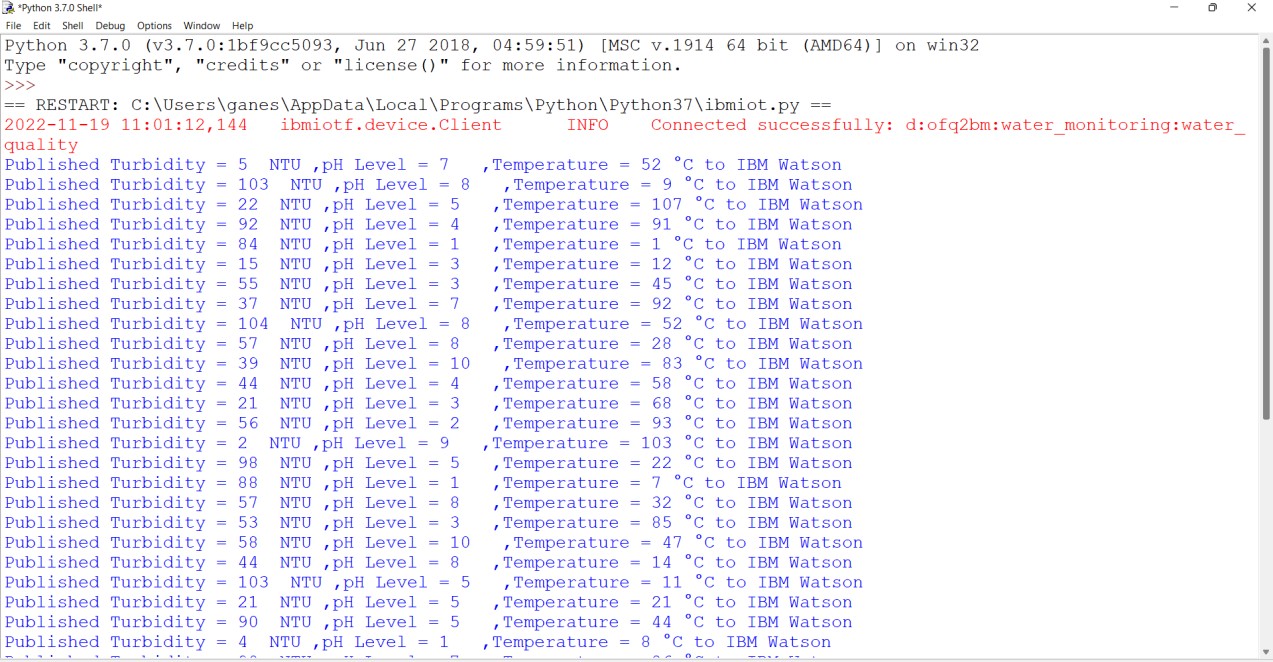
data = { 'turbidity' : turbidity, 'pHLevel': pHLevel ,'temperature':temperature } #print data def

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 IoTF") time.sleep(10) deviceCli.commandCallback = myCommandCallback

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

Iot Sensor is connected and data is published.

