8

SPRINT-I

|  |  |
| --- | --- |
| Date | 18 November 2022 |
| Team ID | PNT2022TMID20521 |
| Project Name | Real-Time Water Quality Monitoring And Control System |

import time

import sys

import ibmiotf.application

import ibmiotf.device

import random

#Provide your IBM Watson Device Credentials

organization = "fwe3x0"

deviceType = "IOT\_device"

deviceId = "1911010"

authMethod = "token"

authToken = "1911010abcdefgh"

# Initialize GPIO

def myCommandCallback(cmd):

print("Command received: %s" % cmd.data['command'])

status=cmd.data['command']

if status == "motoron":

print ("motor is on")

elif status == "motor":

print ("motor is off")

else:

print("Please send proper command")

#print(cmd)

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 random function

pH=random.randint(0,14)

turbidity=random.randint(60,100)

data={'pH':pH,'turbidity':turbidity}

#print data

def myOnPublishCallback():

print (" Published pH = %s " % pH, "turbidity = %s %" % turbidity, "to IBM Watson")

success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0, on\_publish=myOnPublishCallback)

if not success:

print("\n Not connected to IoTF")