PROJECT DEVELOPMENT PHASE

Sprint - 2

TEAM ID : PNT2022TMID08447

PROJECT NAME : Smart Farmer - IoT Enambled Smart Farming

Install python, if install means check in cmd..

This code is used for connect the IBM Watson Iot platform.

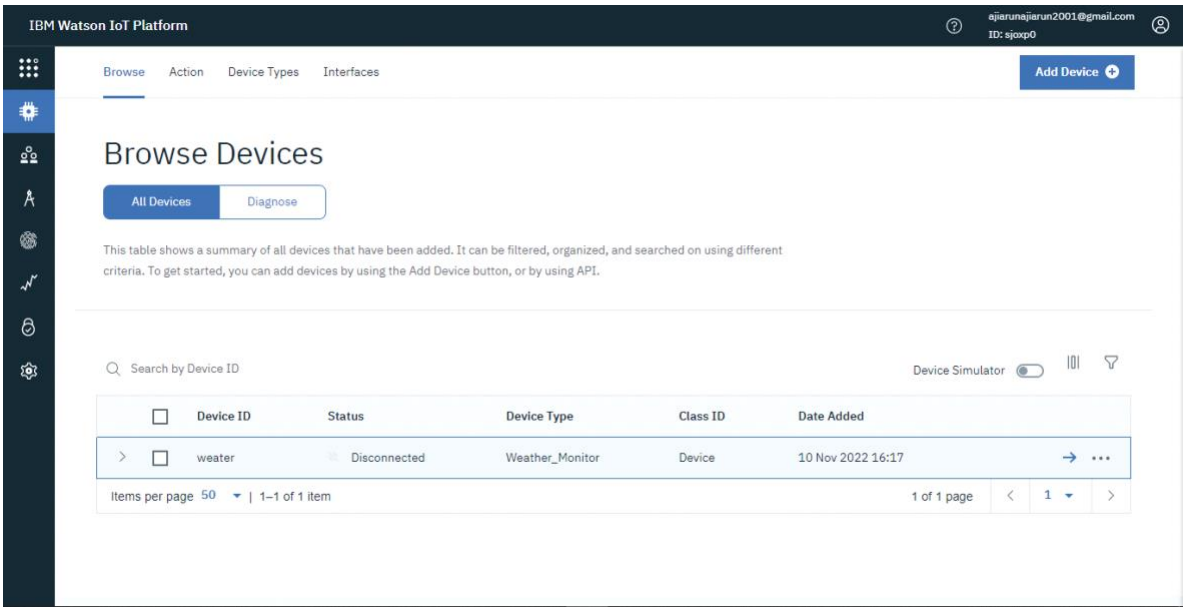
Coding:

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



#Provide your IBM Watson Device Credentials
organization = "sjoxp0"
deviceType = "Weather_Monitor"
deviceId = "weater"
authMethod = "token"
authToken = "ajithbalajimageshsridhar"



# Initialize GPIO


temp=random.randint(0,100)
pulse=random.randint(0,100)
oxygen= random.randint(0,100)
lat =  17
lon = 18
```

```
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['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 DHT11
```

```
        data = {"d":{ 'temp' : temp, 'pulse': pulse ,'oxygen': oxygen,"lat":lat,"lon":lon}}

        #print data

        def myOnPublishCallback():

            print ("Published Temperature = %s C" % temp, "Humidity = %s %%" %
pulse, "to IBM Watson")


        success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0,
on_publish=myOnPublishCallback)

        if not success:

            print("Not connected to IoTF")

        time.sleep(1)


        deviceCli.commandCallback = myCommandCallback


# Disconnect the device and application from the cloud

deviceCli.disconnect()
```

IBM Watson IoT Platform..


IBM Watson IoT Platform
ajarunajiarun2001@gmail.com
ID: sjoxp0
Browse
Action
Device Types
Interfaces
Add Device
Browse Devices
All Devices
Diagnose
This table shows a summary of all devices that have been added. It can be filtered, organized, and searched on using different criteria. To get started, you can add devices by using the Add Device button, or by using API.
Search by Device ID
Device Simulator
Device ID
Status
Device Type
Class ID
Date Added
weater
Disconnected
Weather_Monitor
Device
10 Nov 2022 16:17
Items per page 50 | 1–1 of 1 item
1 of 1 page
1