# PROJECT SCOPE

**BY-NIDHI SINGH** **DATE-17-06-2020**

**PROJECT SUMMARY**

The development of UI based system is to be bulid using IBM cloud watson, Node-red application.The purpose of the project is to unable customers to see the relevant content and get use of the information in app.This project uses this in the agriculture sector and helps farmers to control the irrigation from anywhere.

**PROJECT REQUIREMENTS**

-Github
-Slack Channel
-Zoho Writer
-IBM Cloud
-IBM Watson
-Node-Red

**PROJECT TEAM**

Individual Project.

**FUNCTIONAL REQUIREMENTS**

1. Development of sensor data receivers
2. Development of Open Weather API data receivers
3. Development of actuator controls

**TECHNICAL REQUIREMENTS**

1. Soil humidity and temperature sensors

2. Weather forecast API integration
3. Watering actuators

**SOFTWARE REQUIREMENTS**

1. Open Weather API
2. Watson IoT
3. Python IDE

**PROJECT DELIVERABLES**

An integrated IoT app that enables the farmer to monitor the temperature,humidity and soil moisture parameters along with weather forecasting details.

**PROJECT SCHEDULE**

| **WEEK 1** | **WEEK 2** | **WEEK 3** | **WEEK 4** |
|---|---|---|---|
| planned my project | created device in IBM cloud | started working with node-red | started to prepare reports on zoho writer |
| set up the development environment | installed node red locally | configured IBM iot sensor and node-red UI | pushed my files to github repository |
| created accounts in IBM cloud | installed the required nodes | downloaded python idle and ran the code | recorded the feedback video |

# REPORT

**SUBMITTED BY - NIDHI SINGH**
**EMAIL - komalsingh9961@gmail.com**

# PROJECT NAME : SMARTAGRICULTURE SYSTEM BASED ON IOT

**DATE :**16-06-2020

# PROJECT OVERVIEW

SMART AGRICULTURE SYSTEM IS COMMERCIALLY SCALABLE METHOD WHICH IS BEING BUILT TO REDUCE THE EFFORTS OF FARMERES.THIS PROJECT WILL SOLVE THE MOST IMPORTANT PROBLEM THE FARMERS ARE FACING THAT IS WATERING THE CROPS AT RIGHT TIME AND ACCORDING TO THE REAL TIME FIELD CONDITIONS.

THERE ARE MANY THINGS TO TAKE CARE OF, WHILE WATERING OF CROPS.IF WE WATER THE CROPS TO MUCH, THE CROPS CAN GET DAMAGED DUE TOP WATER LOGGING.IF WE WATER THE CROPS ON A RAINY DAY ,THE EXTRA RAINWATER MAY TAMPER THE GROWTH OF CROPS.IF IN ANY CASE THE FARMER FORGET TO WETER THE CROPS FOR ONE DAY OR TWO BECAUSE OF ANY REASON,IT CAN AGAIN MAKE THE SOIL TWO DRY FOR THE PLANTS TO GET THEIR DAILY NUTRITION LEVEL.

CONSIDERING MANY FACTS ,THE PROJECT IS MAINLY FOCUSED ON AUTOMATING THE PROCESS OF WATER PUMP AND PROVIDING THE FARMERS WITH AN

APPLICATION WHERE THEY CAN SEE THE STATUS OF THE PUMP,MOISTURE OF THE SOIL,HUMIDITY ,TEMPERATURE AS WELL AS WEATHER FORECASTING SO THE FARMER CAN MANY ARRANGEMENTS ACCORDINGLY.

THE ULTIMATE GOAL OF THE PROJECT IS TO INCREASE THE QUALITY OF YIELD BY TAKING CARE OF IRRIGATION PROCESS AND STARING THE REVOLUTIONARY ERA OF FARMERS BY BRINGING TECHNOLGY INTO THE DOMAIN.

## PROJECT SCOPE

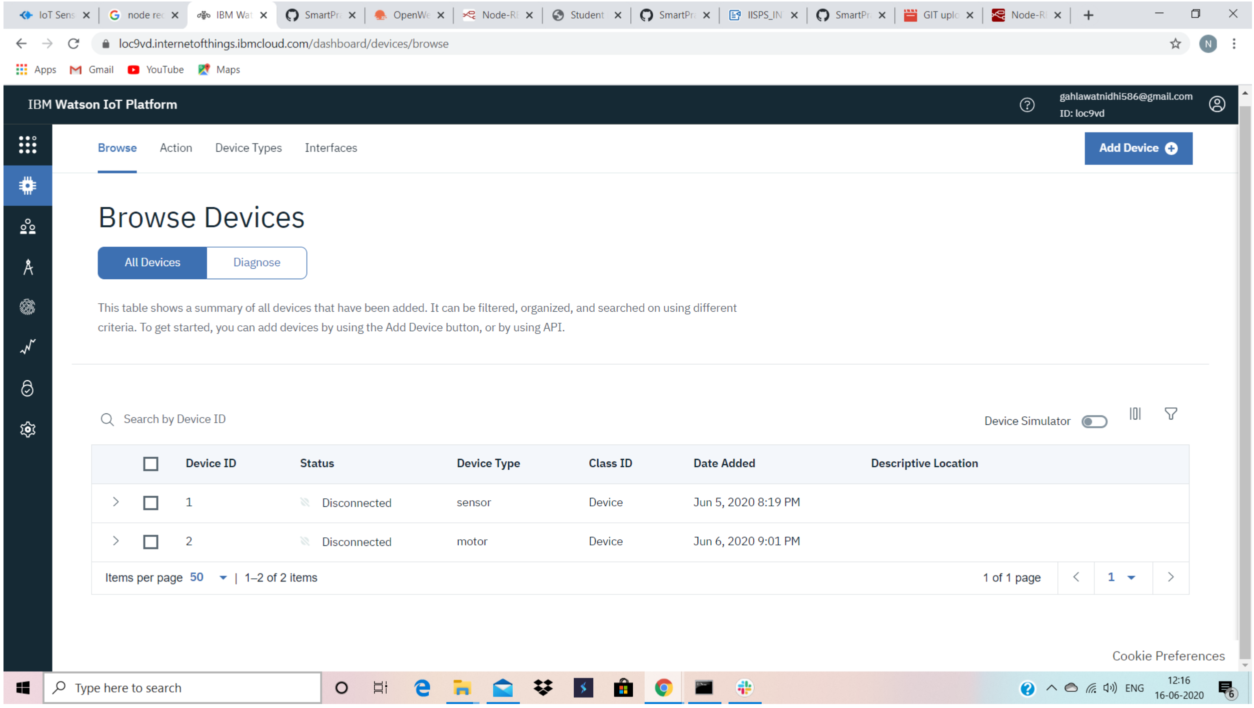
-WE ARE DEVELOPING AN UI FOR THE FARMERS TO HELP THEM CONTROL THE REAL FIELD PARAMETERS SUCH AS TEMPERATURE,HUMIDITY AND SOIL MOISTURE IN ORDER TO IRRIGATE THE FIELD ACCORDINGLY.

-THE UI IS BUILD USING NODE RED.

-THE IBM CLOUD IS USED TO STORE DATA FROM IBM WATSON IOT SIMULATOR WHICH IS KIND OF VIRTUAL SENSOR.

-THIS UI WILL HELP THE FARMERS CONTROL THE IRRIGATION ACCORDING TO THE FIELD PARAMETERS AND THEY CAN DO SO FROM ANYWHERE.

## USE OF IBM WATSON

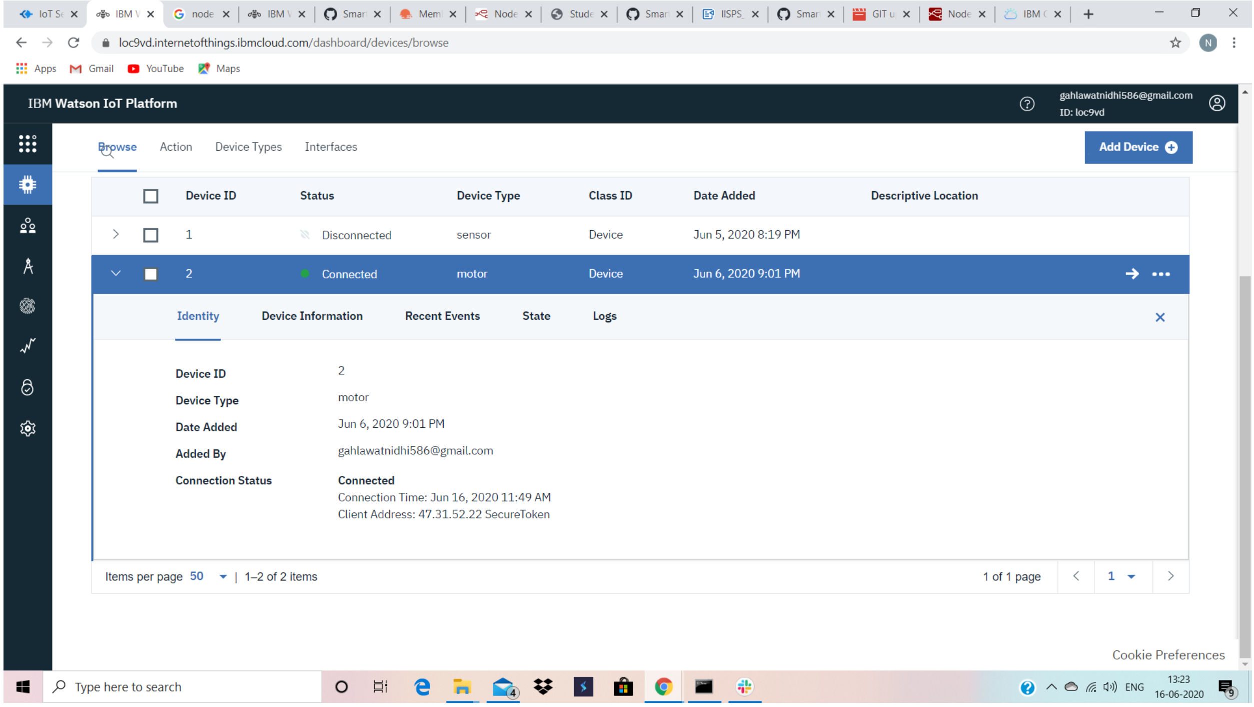


Browse Devices

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.

| | Device ID | Status | Device Type | Class ID | Date Added | Descriptive Location |
|---|---|---|---|---|---|---|
| □ | 1 | Disconnected | sensor | Device | Jun 5, 2020 8:19 PM | |
| □ | 2 | Disconnected | motor | Device | Jun 6, 2020 9:01 PM | |

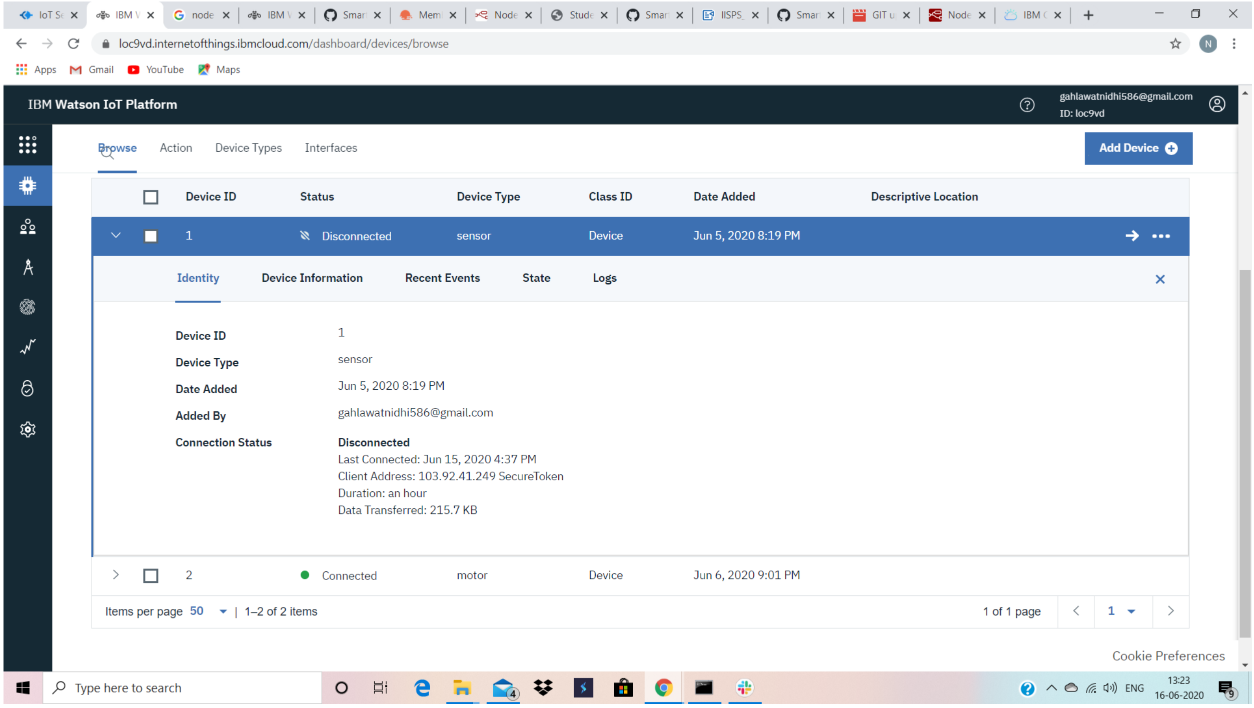
Items per page 50 | 1–2 of 2 items 1 of 1 page 1



| | Device ID | Status | Device Type | Class ID | Date Added | Descriptive Location |
|---|---|---|---|---|---|---|
| □ | 1 | Disconnected | sensor | Device | Jun 5, 2020 8:19 PM | |
| □ | 2 | Connected | motor | Device | Jun 6, 2020 9:01 PM | |

Device ID 2
Device Type motor
Date Added Jun 6, 2020 9:01 PM
Added By gahlawatnidhi586@gmail.com
Connection Status Connected
Connection Time: Jun 16, 2020 11:49 AM
Client Address: 47.31.52.22 SecureToken

Items per page 50 | 1–2 of 2 items 1 of 1 page 1



| Device ID | Status | Device Type | Class ID | Date Added | Descriptive Location |
|---|---|---|---|---|---|
| 1 | Disconnected | sensor | Device | Jun 5, 2020 8:19 PM | |
| 2 | Connected | motor | Device | Jun 6, 2020 9:01 PM | |

Device ID 1

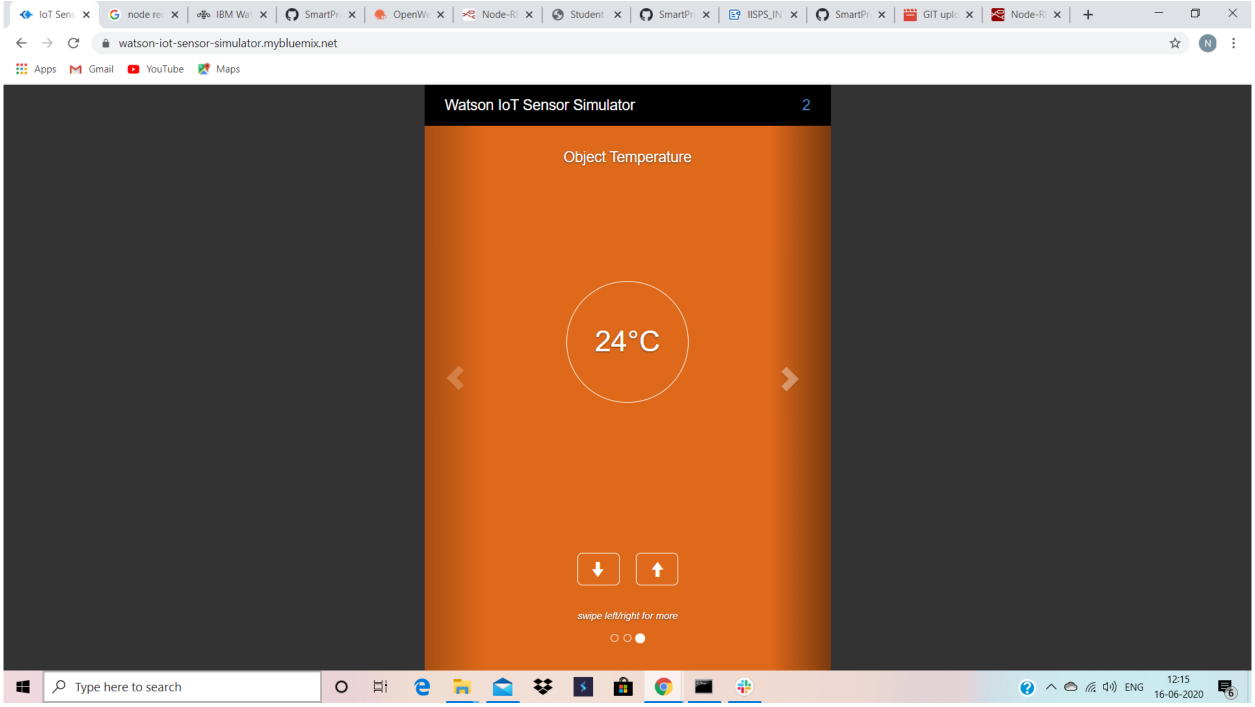
Device Type sensor

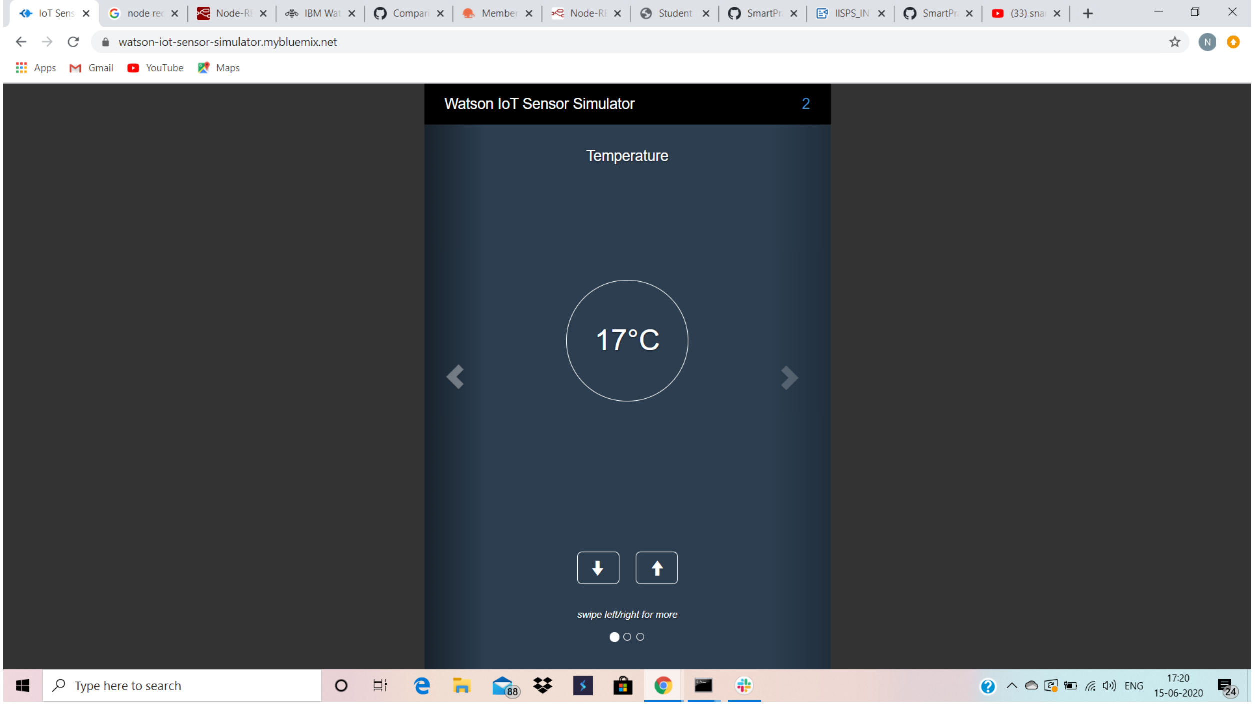
Date Added Jun 5, 2020 8:19 PM

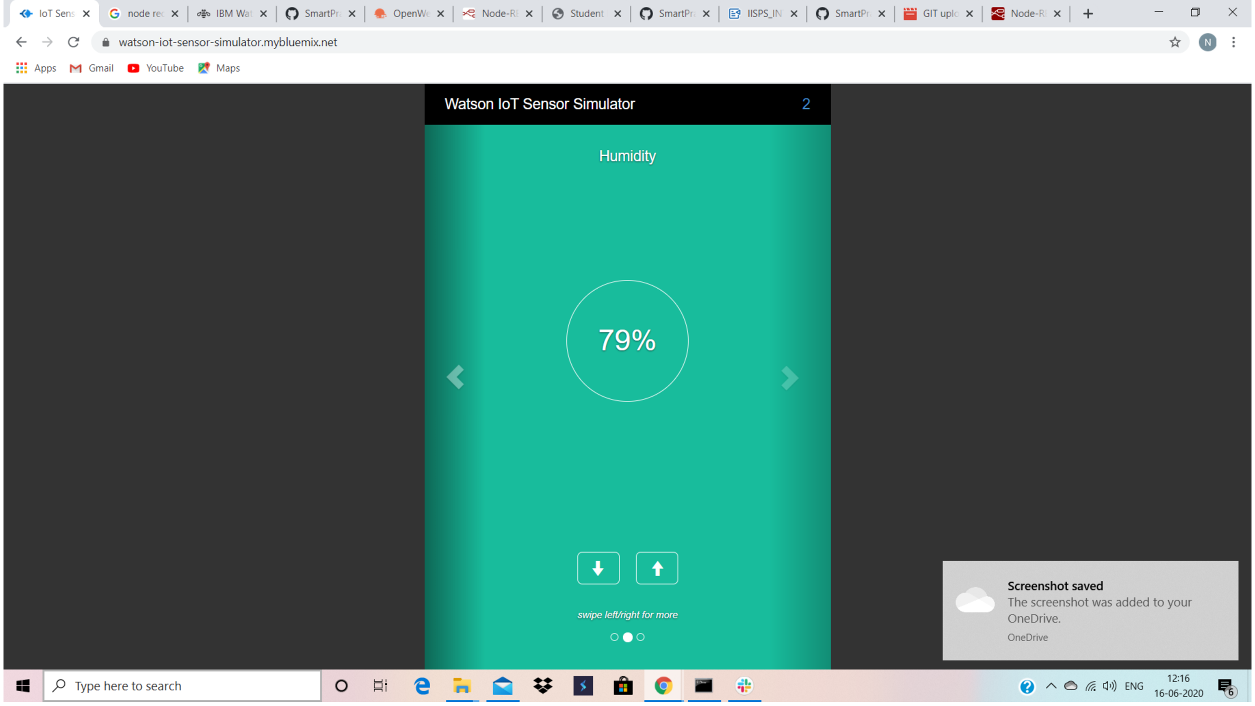
Added By gahlawatnidhi586@gmail.com

Connection Status **Disconnected**
Last Connected: Jun 15, 2020 4:37 PM
Client Address: 103.92.41.249 SecureToken
Duration: an hour
Data Transferred: 215.7 KB

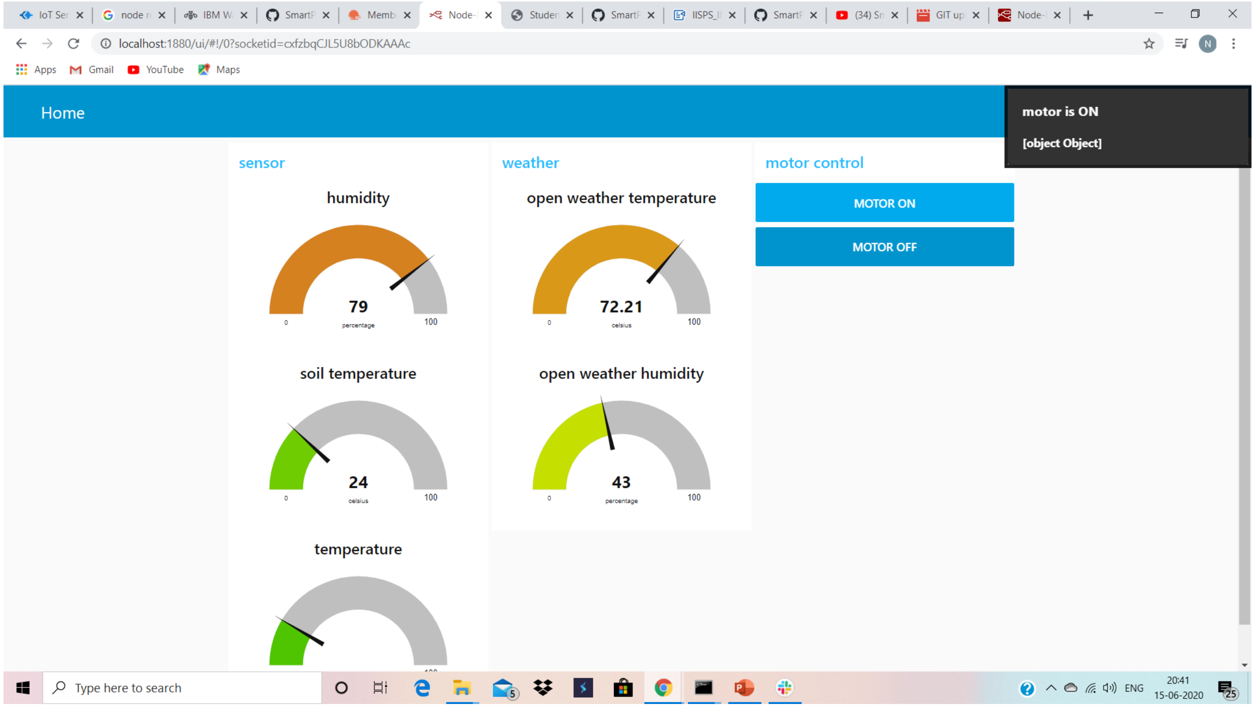
Items per page 50 | 1–2 of 2 items 1 of 1 page 1


watson-iot-sensor-simulator.mybluemix.net
Apps
Gmail
YouTube
Maps
Watson IoT Sensor Simulator
2
Object Temperature
24°C
swipe left/right for more
Type here to search
ENG
12:15
16-06-2020


watson-iot-sensor-simulator.mybluemix.net
Apps
Gmail
YouTube
Maps
Watson IoT Sensor Simulator
2
Temperature
17°C
swipe left/right for more
Type here to search
ENG
17:20
15-06-2020


watson-iot-sensor-simulator.mybluemix.net
Apps
Gmail
YouTube
Maps
Watson IoT Sensor Simulator
2
Humidity
79%
swipe left/right for more
Type here to search
ENG
12:16
16-06-2020

Screenshot saved
The screenshot was added to your OneDrive.
OneDrive



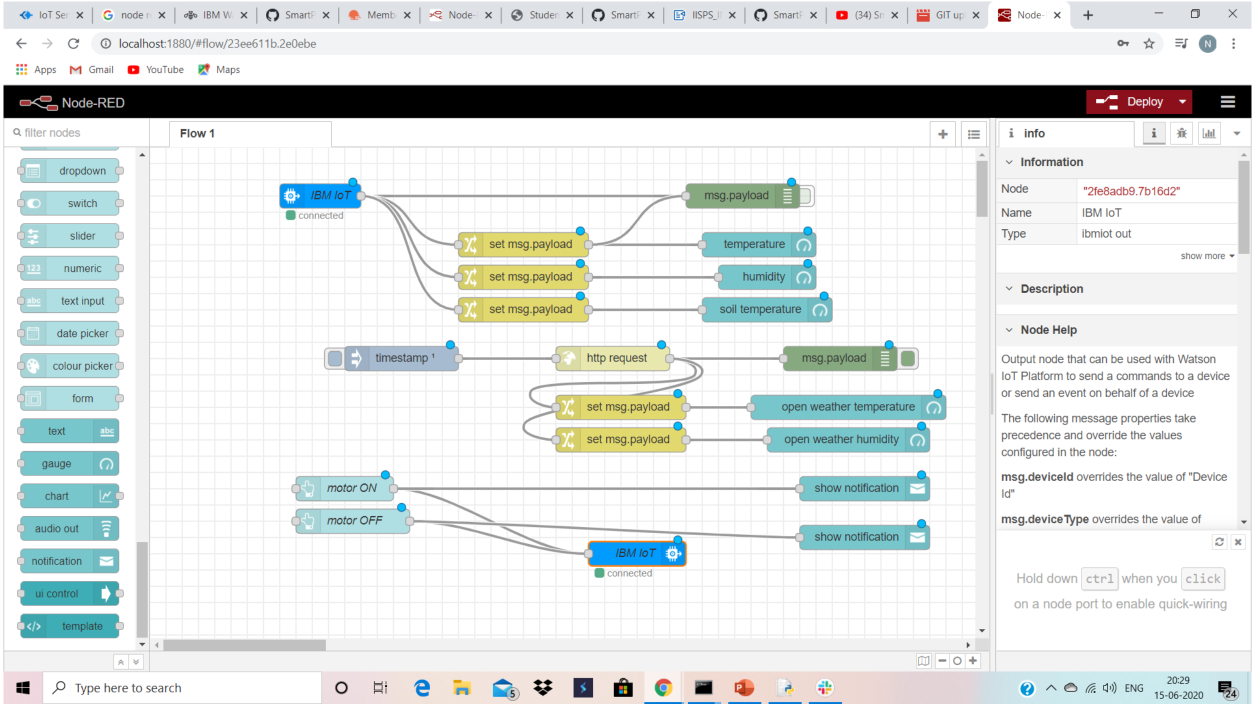
motor is ON

[object Object]

HERE WE HAVE ALL WORKING COMPONENTS SUCH AS THE IOT SENSOR.THE DEVICE WHICH CONNECTS TO THE SENSOR AND UPLOADS DATA INTO THE IBM CLOUD.THE MOTOR WHICH TAKES INPUT FROM THE WEB APP AND THEN UPLOADS INPUT VIA PYTHON CODE ON CLOUD.

THE BOTTOM CORNER OF THE IMAGE SHOWS SOME CARDS WHICH DISPLAYS THE DATA IN A VISUALLY APPEALING WAY THAN JUST NUMBERS SUCH AS LINE GRAPHS OR GAUGE WHICH MAKES IT EASY TO UNDERSTAND.

## FUNCTIONING OF THE APP



Information

| | |
|---|---|
| Node | "2fe8adb9.7b16d2" |
| Name | IBM IoT |
| Type | ibmiot out |

show more

Description

Node Help

Output node that can be used with Watson IoT Platform to send a commands to a device or send an event on behalf of a device

The following message properties take precedence and override the values configured in the node:

**msg.deviceId** overrides the value of "Device Id"

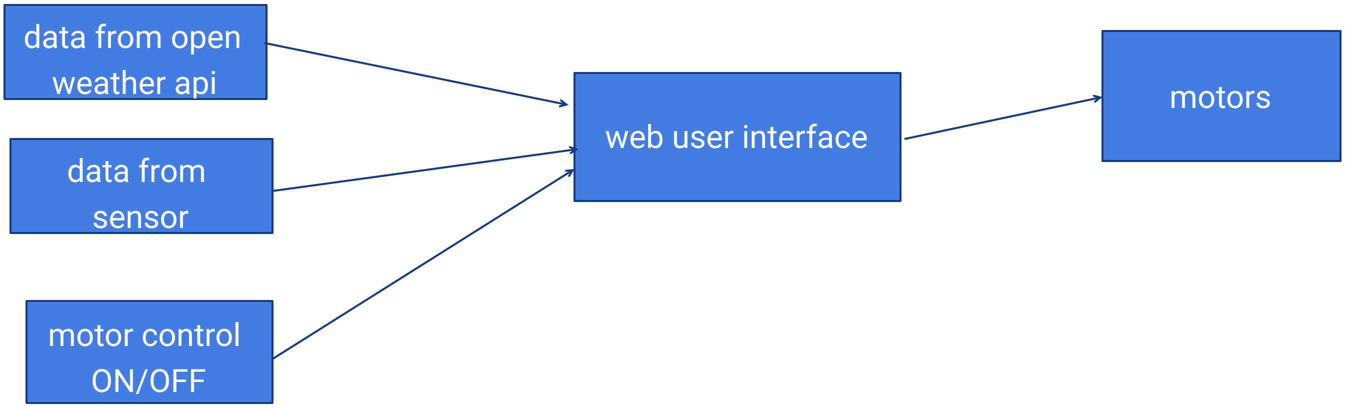
**msg.deviceType** overrides the value of

Hold down ctrl when you click on a node port to enable quick-wiring

IN THIS FLOW WE HAVE IBM NODES TO GET THE IOT SENSOR DATA FROM IBM CLOUD AND PUSH THE MOTOR ON/OFF COMMANDS FROM THE WEB DASHBOARD FROM THE WEB DASHBOARD BACK TO THE IBM CLOUD.

HERE WE ARE USING HTTP REQUEST WHICH I AM USING TO GET CURRENT WEATHER DATA AND DISPLAY ON THE DASHBOARD.

## THEORY BLOCK DIAGRAM

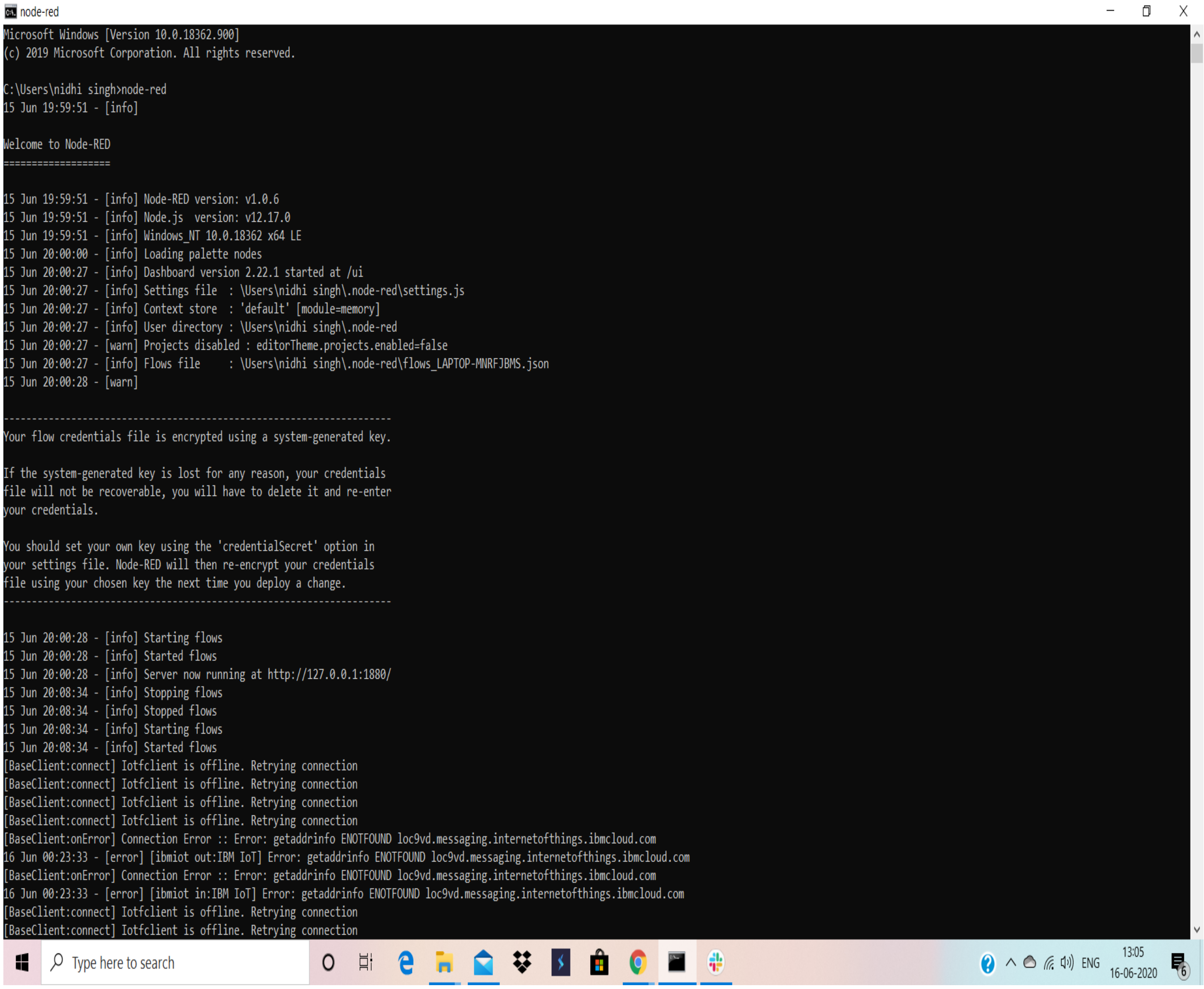


THIS BLOCK DIAGRAM IS SHOWING THEORITICAL COMPUTATION OF WHAT ARE WE BASICALLY DOING IN THIS PROJECT.

SO, HERE WE ARE BASICALLY MONITORING THE WEATHER CONDITIONS AND COLLECTING DATA FROM OPEN WEATHER API.
THE REAL TIME FIELD CONDITIONS ARE MONITORED VIA SENSOR AND THE DATA IS COLLECTED VIA SENSOR.
AND ACCORDINGLY THE MOTOR IS CONTROLLED BY ON/OFF BUTTONS.

ALL THESE DATAS ARE SEND TO THE WEB USER INTERFACE AND THE MOTOR IS CONTROLLED ACCORDINGLY.

## NODE-RED ON COMMAND PROMPT



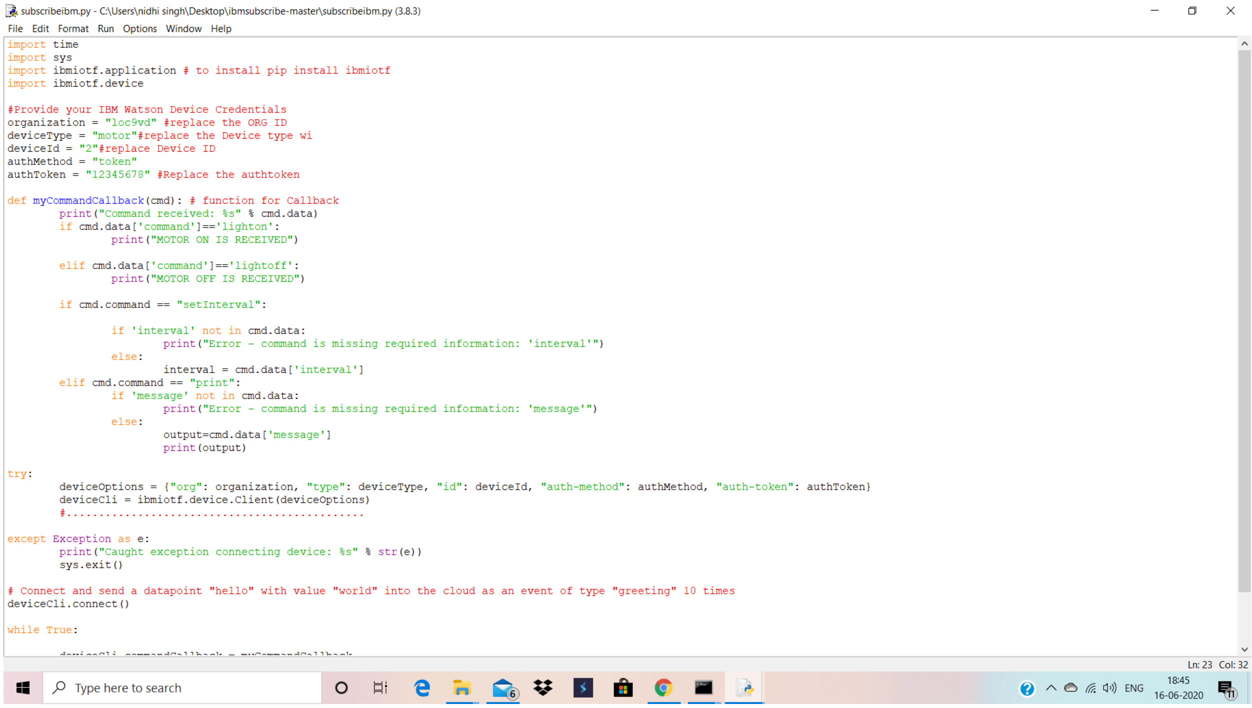
```
Microsoft Windows [Version 10.0.18362.900]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\nidhi singh>node-red
15 Jun 19:59:51 - [info]

Welcome to Node-RED
===================

15 Jun 19:59:51 - [info] Node-RED version: v1.0.6
15 Jun 19:59:51 - [info] Node.js  version: v12.17.0
15 Jun 19:59:51 - [info] Windows_NT 10.0.18362 x64 LE
15 Jun 20:00:00 - [info] Loading palette nodes
15 Jun 20:00:27 - [info] Dashboard version 2.22.1 started at /ui
15 Jun 20:00:27 - [info] Settings file  : \Users\nidhi singh\.node-red\settings.js
15 Jun 20:00:27 - [info] Context store  : 'default' [module=memory]
15 Jun 20:00:27 - [info] User directory : \Users\nidhi singh\.node-red
15 Jun 20:00:27 - [warn] Projects disabled : editorTheme.projects.enabled=false
15 Jun 20:00:27 - [info] Flows file     : \Users\nidhi singh\.node-red\flows_LAPTOP-MNRFJBMS.json
15 Jun 20:00:28 - [warn]

---------------------------------------------------------------------
Your flow credentials file is encrypted using a system-generated key.

If the system-generated key is lost for any reason, your credentials
file will not be recoverable, you will have to delete it and re-enter
your credentials.

You should set your own key using the 'credentialSecret' option in
your settings file. Node-RED will then re-encrypt your credentials
file using your chosen key the next time you deploy a change.
---------------------------------------------------------------------

15 Jun 20:00:28 - [info] Starting flows
15 Jun 20:00:28 - [info] Started flows
15 Jun 20:00:28 - [info] Server now running at http://127.0.0.1:1880/
15 Jun 20:08:34 - [info] Stopping flows
15 Jun 20:08:34 - [info] Stopped flows
15 Jun 20:08:34 - [info] Starting flows
15 Jun 20:08:34 - [info] Started flows
[BaseClient:connect] Iotfclient is offline. Retrying connection
[BaseClient:connect] Iotfclient is offline. Retrying connection
[BaseClient:connect] Iotfclient is offline. Retrying connection
[BaseClient:connect] Iotfclient is offline. Retrying connection
[BaseClient:onError] Connection Error :: Error: getaddrinfo ENOTFOUND loc9vd.messaging.internetofthings.ibmcloud.com
16 Jun 00:23:33 - [error] [ibmiot out:IBM IoT] Error: getaddrinfo ENOTFOUND loc9vd.messaging.internetofthings.ibmcloud.com
[BaseClient:onError] Connection Error :: Error: getaddrinfo ENOTFOUND loc9vd.messaging.internetofthings.ibmcloud.com
16 Jun 00:23:33 - [error] [ibmiot in:IBM IoT] Error: getaddrinfo ENOTFOUND loc9vd.messaging.internetofthings.ibmcloud.com
[BaseClient:connect] Iotfclient is offline. Retrying connection
[BaseClient:connect] Iotfclient is offline. Retrying connection
```

# PROJECT SCHEDULE

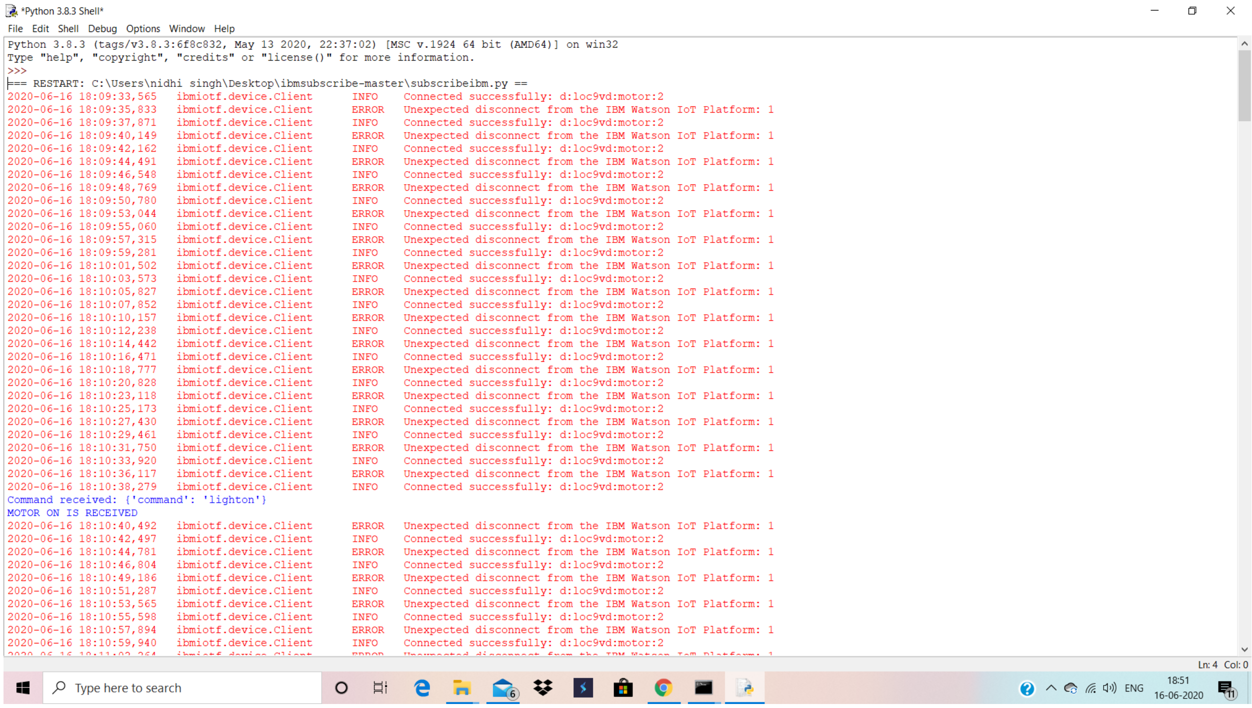
| **WEEK 1** | **WEEK 2** | **WEEK 3** | **WEEK 4** |
|---|---|---|---|
| planned my project | created device in IBM cloud | started working with node-red | started to prepare reports on zoho writer |

| | | | |
|---|---|---|---|
| set up the development environment | installed node red locally | configured IBM iot sensor and node-red UI | pushed my files to github repository |
| created accounts in IBM cloud | installed the required nodes | downloaded python idle and ran the code | recorded the feedback video |

# PYTHON CODE


subscribeibm.py - C:\Users\nidhi singh\Desktop\ibmsubscribe-master\subscribeibm.py (3.8.3)
File Edit Format Run Options Window Help
Ln: 23 Col: 32
Type here to search
ENG
18:45
16-06-2020

```
import time
import sys
import ibmiotf.application # to install pip install ibmiotf
import ibmiotf.device

#Provide your IBM Watson Device Credentials
organization = "loc9vd" #replace the ORG ID
deviceType = "motor"#replace the Device type wi
deviceId = "2"#replace Device ID
authMethod = "token"
authToken = "12345678" #Replace the authtoken

def myCommandCallback(cmd): # function for Callback
        print("Command received: %s" % cmd.data)
        if cmd.data['command']=='lighton':
                print("MOTOR ON IS RECEIVED")

        elif cmd.data['command']=='lightoff':
                print("MOTOR OFF IS RECEIVED")

        if cmd.command == "setInterval":

                if 'interval' not in cmd.data:
                        print("Error - command is missing required information: 'interval'")
                else:
                        interval = cmd.data['interval']
        elif cmd.command == "print":
                if 'message' not in cmd.data:
                        print("Error - command is missing required information: 'message'")
                else:
                        output=cmd.data['message']
                        print(output)

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:

        deviceCli.commandCallback = myCommandCallback
```



```
Python 3.8.3 (tags/v3.8.3:6f8c832, May 13 2020, 22:37:02) [MSC v.1924 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
=== RESTART: C:\Users\nidhi singh\Desktop\ibmsubscribe-master\subscribeibm.py ==
2020-06-16 18:09:33,565   ibmiotf.device.Client      INFO    Connected successfully: d:loc9vd:motor:2
2020-06-16 18:09:35,833   ibmiotf.device.Client      ERROR   Unexpected disconnect from the IBM Watson IoT Platform: 1
2020-06-16 18:09:37,871   ibmiotf.device.Client      INFO    Connected successfully: d:loc9vd:motor:2
2020-06-16 18:09:40,149   ibmiotf.device.Client      ERROR   Unexpected disconnect from the IBM Watson IoT Platform: 1
2020-06-16 18:09:42,162   ibmiotf.device.Client      INFO    Connected successfully: d:loc9vd:motor:2
2020-06-16 18:09:44,491   ibmiotf.device.Client      ERROR   Unexpected disconnect from the IBM Watson IoT Platform: 1
2020-06-16 18:09:46,548   ibmiotf.device.Client      INFO    Connected successfully: d:loc9vd:motor:2
2020-06-16 18:09:48,769   ibmiotf.device.Client      ERROR   Unexpected disconnect from the IBM Watson IoT Platform: 1
2020-06-16 18:09:50,780   ibmiotf.device.Client      INFO    Connected successfully: d:loc9vd:motor:2
2020-06-16 18:09:53,044   ibmiotf.device.Client      ERROR   Unexpected disconnect from the IBM Watson IoT Platform: 1
2020-06-16 18:09:55,060   ibmiotf.device.Client      INFO    Connected successfully: d:loc9vd:motor:2
2020-06-16 18:09:57,315   ibmiotf.device.Client      ERROR   Unexpected disconnect from the IBM Watson IoT Platform: 1
2020-06-16 18:09:59,281   ibmiotf.device.Client      INFO    Connected successfully: d:loc9vd:motor:2
2020-06-16 18:10:01,502   ibmiotf.device.Client      ERROR   Unexpected disconnect from the IBM Watson IoT Platform: 1
2020-06-16 18:10:03,573   ibmiotf.device.Client      INFO    Connected successfully: d:loc9vd:motor:2
2020-06-16 18:10:05,827   ibmiotf.device.Client      ERROR   Unexpected disconnect from the IBM Watson IoT Platform: 1
2020-06-16 18:10:07,852   ibmiotf.device.Client      INFO    Connected successfully: d:loc9vd:motor:2
2020-06-16 18:10:10,157   ibmiotf.device.Client      ERROR   Unexpected disconnect from the IBM Watson IoT Platform: 1
2020-06-16 18:10:12,238   ibmiotf.device.Client      INFO    Connected successfully: d:loc9vd:motor:2
2020-06-16 18:10:14,442   ibmiotf.device.Client      ERROR   Unexpected disconnect from the IBM Watson IoT Platform: 1
2020-06-16 18:10:16,471   ibmiotf.device.Client      INFO    Connected successfully: d:loc9vd:motor:2
2020-06-16 18:10:18,777   ibmiotf.device.Client      ERROR   Unexpected disconnect from the IBM Watson IoT Platform: 1
2020-06-16 18:10:20,828   ibmiotf.device.Client      INFO    Connected successfully: d:loc9vd:motor:2
2020-06-16 18:10:23,118   ibmiotf.device.Client      ERROR   Unexpected disconnect from the IBM Watson IoT Platform: 1
2020-06-16 18:10:25,173   ibmiotf.device.Client      INFO    Connected successfully: d:loc9vd:motor:2
2020-06-16 18:10:27,430   ibmiotf.device.Client      ERROR   Unexpected disconnect from the IBM Watson IoT Platform: 1
2020-06-16 18:10:29,461   ibmiotf.device.Client      INFO    Connected successfully: d:loc9vd:motor:2
2020-06-16 18:10:31,750   ibmiotf.device.Client      ERROR   Unexpected disconnect from the IBM Watson IoT Platform: 1
2020-06-16 18:10:33,920   ibmiotf.device.Client      INFO    Connected successfully: d:loc9vd:motor:2
2020-06-16 18:10:36,117   ibmiotf.device.Client      ERROR   Unexpected disconnect from the IBM Watson IoT Platform: 1
2020-06-16 18:10:38,279   ibmiotf.device.Client      INFO    Connected successfully: d:loc9vd:motor:2
Command received: {'command': 'lighton'}
MOTOR ON IS RECEIVED
2020-06-16 18:10:40,492   ibmiotf.device.Client      ERROR   Unexpected disconnect from the IBM Watson IoT Platform: 1
2020-06-16 18:10:42,497   ibmiotf.device.Client      INFO    Connected successfully: d:loc9vd:motor:2
2020-06-16 18:10:44,781   ibmiotf.device.Client      ERROR   Unexpected disconnect from the IBM Watson IoT Platform: 1
2020-06-16 18:10:46,804   ibmiotf.device.Client      INFO    Connected successfully: d:loc9vd:motor:2
2020-06-16 18:10:49,186   ibmiotf.device.Client      ERROR   Unexpected disconnect from the IBM Watson IoT Platform: 1
2020-06-16 18:10:51,287   ibmiotf.device.Client      INFO    Connected successfully: d:loc9vd:motor:2
2020-06-16 18:10:53,565   ibmiotf.device.Client      ERROR   Unexpected disconnect from the IBM Watson IoT Platform: 1
2020-06-16 18:10:55,598   ibmiotf.device.Client      INFO    Connected successfully: d:loc9vd:motor:2
2020-06-16 18:10:57,894   ibmiotf.device.Client      ERROR   Unexpected disconnect from the IBM Watson IoT Platform: 1
2020-06-16 18:10:59,940   ibmiotf.device.Client      INFO    Connected successfully: d:loc9vd:motor:2
2020-06-16 18:11:02,264   ibmiotf.device.Client      ERROR   Unexpected disconnect from the IBM Watson IoT Platform: 1
```

WHEN THE MOTOR COMMANDS ARE GIVEN ON THE NODE-RED DASHBOARD ,WE GET SIGNAL IN THE PYTHON PROGRAM.

# BIBLIOGRAPHY

1.https://cloud.ibm.com/
2.https://watson-iot-sensor-simulator.mybluemix.net/
3.https://loc9vd.internetofthings.ibmcloud.com/dashboard/devices/browse
4.http://localhost:1880/ui/#!/0?socketid=cxfzbqCJL5U8bODKAAAc
5.http://localhost:1880/#flow/23ee611b.2e0ebe
6.
https://github.com/SmartPracticeschool/llSPS-INT-2244-Smart-Agriculture-system-based-on-IoT