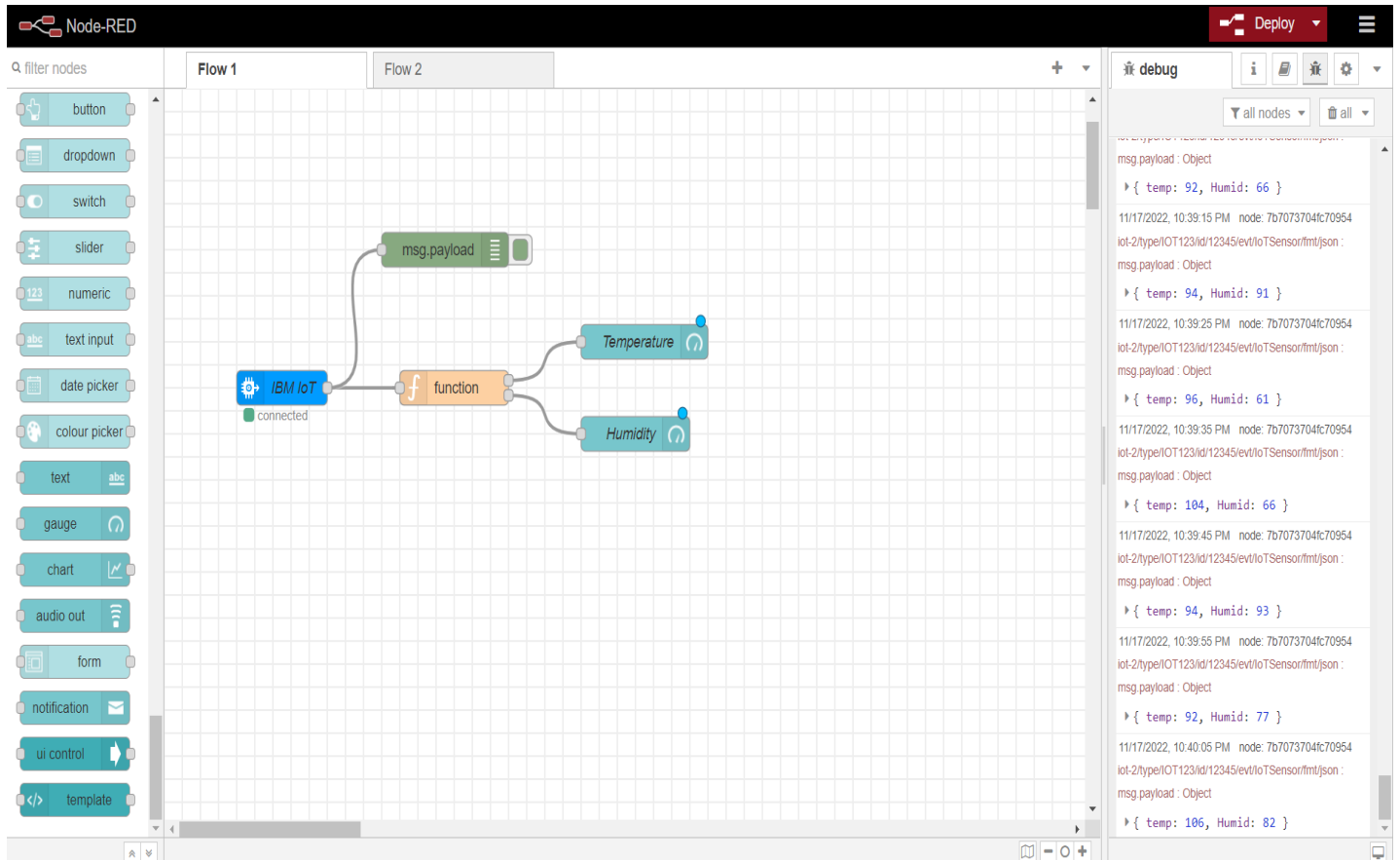


DEVELOP A WEB APPLICATION USING NODE RED SERVICE (DEVELOP THE WEB APPLICATION USING NODE RED SERVICE)

Team ID	PNT2022TMID34718
Project Name	Industry-Specific Intelligent Fire Management System



```

ibmpublish.py - C:\Python\Python37\ibmpublish.py (3.7.4)
File Edit Format Run Options Window Help

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

#Provide your IBM Watson Device Credentials
organization = "zoieul"
deviceType = "IOT123"
deviceId = "12345"
authMethod = "token"
authToken = "123456789"

# 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-tok
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

    temp=random.randint(90,110)
    Humid=random.randint(60,100)

    data = { 'temp' : temp, 'Humid': Humid }
    #print data
    def myOnPublishCallback():
        print ("Published Temperature = %s C" % temp, "Humidity = %s %" % Humid, "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()

```

IBM Watson IoT Platform

?

dahlia.201924@sxce.edu.in

ID: zoleul

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.

Q Search by Device ID

Device Simulator

	Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
>	<div></div> 12345	<div></div> Connected	IOT123	Device	Nov 17, 2022 8:23 PM	
>	<div></div> 12345	<div></div> Disconnected	NodeMCU	Device	Nov 15, 2022 2:10 PM	

Items per page 50 | 1-2 of 2 items

1 of 1 page

<

1

>