

Built a web application using Node-Red

The screenshot shows the IBM Watson IoT Platform dashboard. The top navigation bar includes 'Browse', 'Action', 'Device Types', and 'Interfaces'. A search bar is present. The main content area displays a table of devices. The 'trainingid' device is selected, and its details are shown in a modal window. The details include:

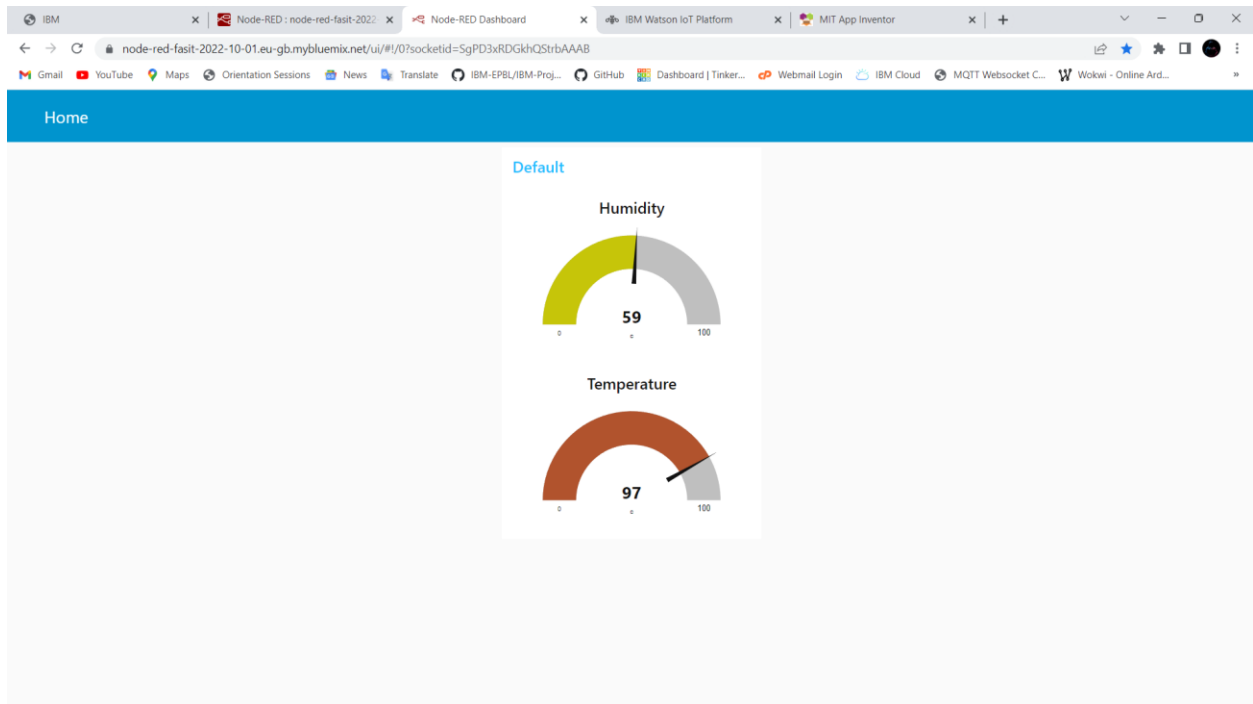
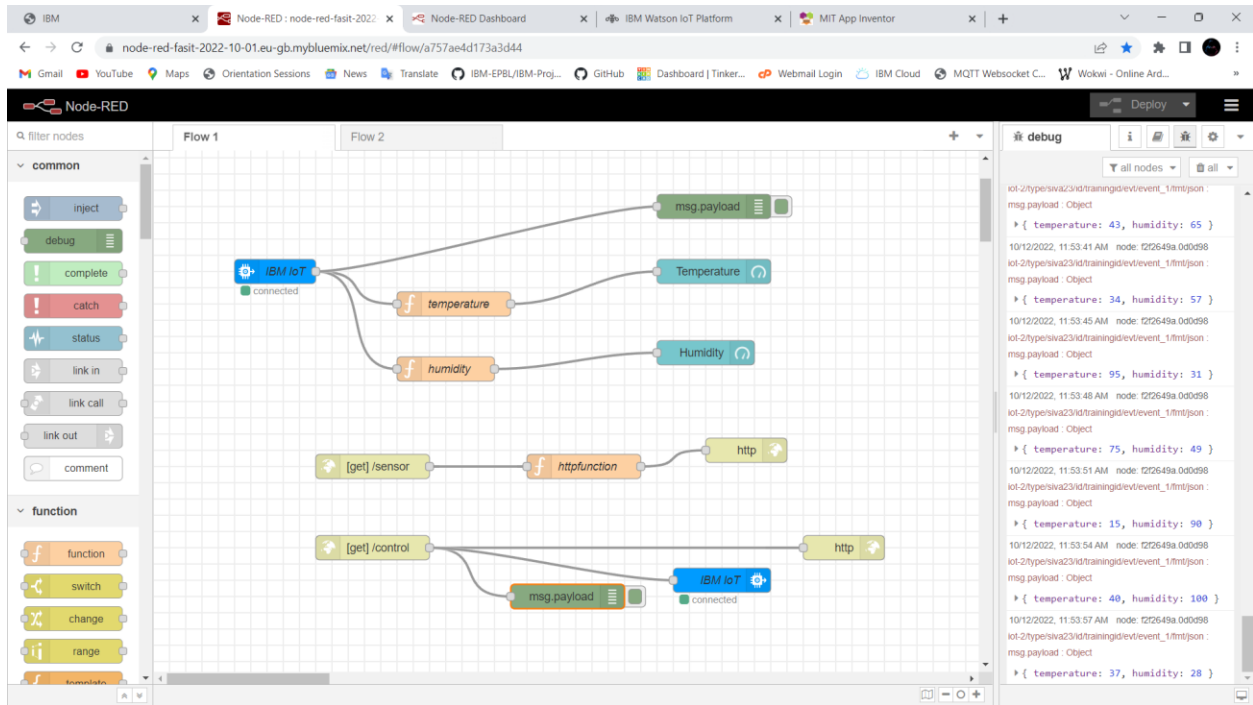
- Device ID:** trainingid
- Device Type:** siva23
- Date Added:** Oct 1, 2022 6:44 PM
- Added By:** 511919106010@smartinternz.com
- Connection Status:** Disconnected
- Last Connected:** Oct 12, 2022 11:17 AM
- Client Address:** 106.197.128.19
- SecureToken:** [redacted]
- Duration:** 2 minutes
- Data Transferred:** 22.8 KB

The bottom of the modal shows '1 Simulation running'.

The screenshot shows the IBM Watson IoT Platform dashboard. The top navigation bar includes 'Browse', 'Action', 'Device Types', and 'Interfaces'. A search bar is present. The main content area displays a table of devices. The 'trainingid' device is selected, and its details are shown in a modal window. The 'Recent Events' tab is active, showing a list of events:

Event	Value	Format	Last Received
event_1	{"temperature":54,"humidity":8}	json	a few seconds ago
event_1	{"temperature":42,"humidity":92}	json	a few seconds ago
event_1	{"temperature":35,"humidity":2}	json	a few seconds ago
event_1	{"temperature":0,"humidity":80}	json	a few seconds ago
event_1	{"temperature":90,"humidity":86}	json	a few seconds ago

The bottom of the modal shows '1 Simulation running'.



```
light on & off.py - C:\Users\K.SIVASHANKAR\OneDrive\Documents\OneDrive\Documents\light on & off.py (3.7.0)
File Edit Format Run Options Window Help

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

#Provide your IBM Watson Device Credentials
organization = "31b6or"
deviceType = "siwa23"
deviceId = "trainingid"
authMethod = "token"
authToken = "zxm-aQ89j5YD(G7OXQ"
# Initialize GPIO
temp=60
pulse=70
oxygen= 30
lat = 17
lon = 18
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="lighton":
        print("light is on")
    else:
        print("light is off")
    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

Ln: 1 Col: 0
```

```
light on & off.py - C:\Users\K.SIVASHANKAR\OneDrive\Documents\OneDrive\Documents\light on & off.py (3.7.0)
File Edit Format Run Options Window Help

#Provide your IBM Watson Device Credentials
organization = "31b6or"
deviceType = "siwa23"
deviceId = "trainingid"
authMethod = "token"
authToken = "zxm-aQ89j5YD(G7OXQ"
# Initialize GPIO
temp=60
pulse=70
oxygen= 30
lat = 17
lon = 18
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="lighton":
        print("light is on")
    else:
        print("light is off")
    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 remove it from the cloud
deviceCli.disconnect()

Ln: 54 Col: 0
```

```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help

Command received: light-on
light is on
Command received: light-off
light is off
Command received: light-on
light is on
Command received: light-on
light is on
Command received: light-off
light is off
Published Temperature = 60 C Humidity = 70 % to IBM Watson
Command received: light-off
light is off
Command received: light-on
light is on
Command received: light-off
light is off
Command received: light-on
light is on
Command received: light-off
light is off
Published Temperature = 60 C Humidity = 70 % to IBM Watson
light is off
Published Temperature = 60 C Humidity = 70 % to IBM Watson
Published Temperature = 60 C Humidity = 70 % to IBM Watson
Published Temperature = 60 C Humidity = 70 % to IBM Watson
Published Temperature = 60 C Humidity = 70 % to IBM Watson
Published Temperature = 60 C Humidity = 70 % to IBM Watson
Published Temperature = 60 C Humidity = 70 % to IBM Watson
Command received: light-on
light is on
Published Temperature = 60 C Humidity = 70 % to IBM Watson
Published Temperature = 60 C Humidity = 70 % to IBM Watson
Published Temperature = 60 C Humidity = 70 % to IBM Watson
Published Temperature = 60 C Humidity = 70 % to IBM Watson
Command received: light-off
light is off
Published Temperature = 60 C Humidity = 70 % to IBM Watson
Published Temperature = 60 C Humidity = 70 % to IBM Watson

Ln: 1 Col: 0
```