

SPRINT -3

Date	12 NOVEMBER 2022
Team ID	PNT2022TMID22101
Project Name	Project – SIGNS WITH SMART CONNECTIVITY FORBETTER ROAD SAFETY
Maximum Marks	4 Marks

```

import wiotp.sdk.device
import time
import random
import ibmiotf.application
import ibmiotf.device
import requests, json

myConfig = {
    "identity": {
        "orgId": "4gh14s",
        "typeId": "ESP32",
        "deviceId": "1234"
    },
    "auth": {
        "token": "5xp6Zc74hThvC!qyOY"
    }
}

def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
    m=cmd.data['command']
    if(m=="alarm ON"):
        print("Alarm is turned ON")
    elif(m=="alarm OFF"):
        print("Alarm is turned OFF")
    print(" ")
client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()
BASE_URL = "https://api.openweathermap.org/data/2.5/weather?"
CITY = "Jaipur, IN"
URL = BASE_URL + "q=" + CITY + "&units=metric"+"&appid=" + "be42a38741dd6a72d994a4bc7d9a5025"

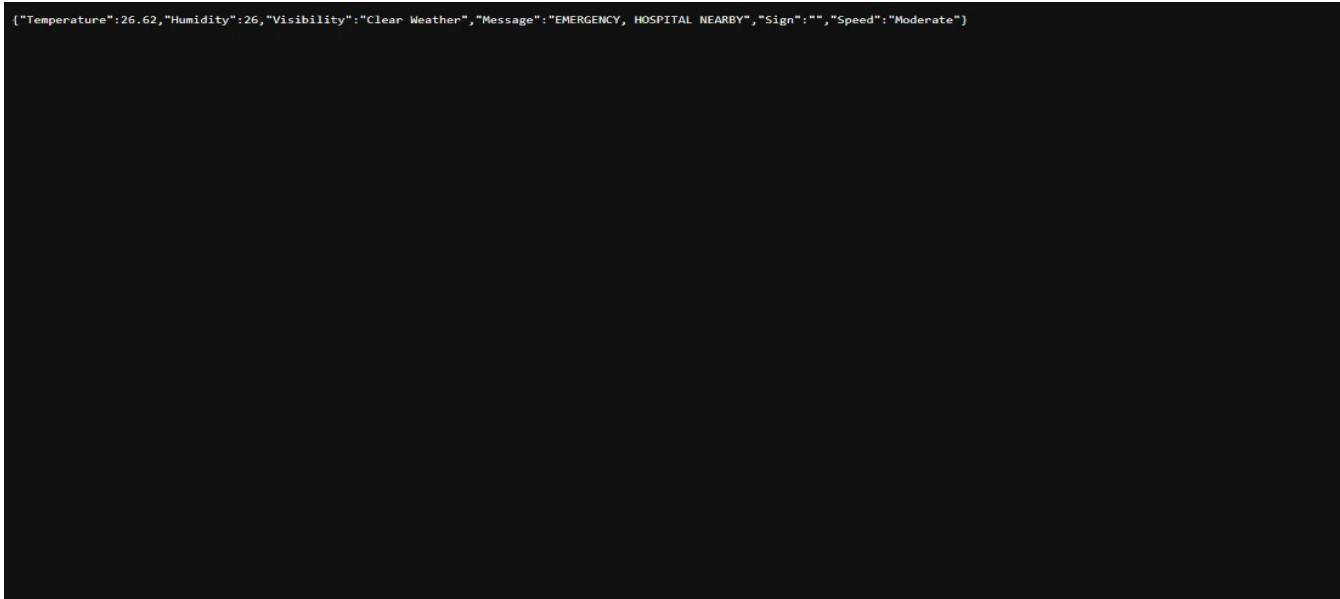
```

```

while True:
    response = requests.get(URL)
    if response.status_code == 200:
        data = response.json()
    main = data['main']
    temperature = main['temp']
    humidity = main['humidity']
    pressure = main['pressure']
    report = data['visibility']
    msg=random.randint(0,5)
    if msg==1:
        message="GO SLOW, SCHOOL ZONE AHEAD"
    elif msg==2:
        message="NEED HELP, POLICE STATION AHEAD"
    elif msg==3:
        message="EMERGENCY, HOSPITAL NEARBY"
    elif msg==4:
        message="DINE IN, RESTAURENT AVAILABLE"
    elif msg==5:
        message="PETROL BUNK NEARBY"
    else:
        message=""
    speed=random.randint(0,150)
    if speed>=100:
        speedMsg=" Limit Exceeded"
    elif speed>=60 and speed<100:
        speedMsg="Moderate"
    else:
        speedMsg="Slow"
    sign=random.randint(0,5)
    if sign==1:
        signMsg="Right Diversion"
    elif sign==2:
        signMsg="Speed Breaker"
    elif sign==3:
        signMsg="Left Diversion"
    speed=random.randint(0,150)
    if speed>=100:
        speedMsg=" Limit Exceeded"
    elif speed>=60 and speed<100:
        speedMsg="Moderate"
    else:
        speedMsg="Slow"
    sign=random.randint(0,5)
    if sign==1:
        signMsg="Right Diversion"
    elif sign==2:
        signMsg="Speed Breaker"
    elif sign==3:
        signMsg="Left Diversion"
    elif sign==4:
        signMsg="U Turn"
    else:
        signMsg=""
    if temperature < 24:
        visibility="Fog Ahead, Drive Slow"
    elif temperature < 20:
        visibility="Bad Weather"
    else:
        visibility="Clear Weather"

myData={'Temperature':temperature, 'Humidity':humidity,'Visibility':visibility,'Message':message, 'Sign':signMsg, 'Speed':speedMsg}
client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0, onPublish=None)
print("Published data Successfully: ", myData)
print("-----")
client.commandCallback = myCommandCallback
time.sleep(2)
client.disconnect()

```



RoadSafety

screen1 | Add Screen... | Remove Screen | Publish to Gallery | Designer | block

Blocks

- Built-in
 - Control
 - Logic
 - Math
 - Text
 - Lists
 - Dictionaries
 - Colors
 - Variables
 - Procedures
- Screen1
 - VerticalArrangement1
 - HorizontalArrangement1
 - VerticalArrangement1
 - Label1
 - item_data
 - VerticalArrangement1
 - VerticalArrangement1

Viewer

```
[{"Temperature":26.62,"Humidity":26,"Visibility":"Clear Weather","Message":"EMERGENCY, HOSPITAL NEARBY","Sign":"","Speed":"Moderate"}]
```

when Clock1 = Timer
do set [Web1 -> Uri] to "https://node-red-pwubyd-2022-11-15.au-syd.mybluemix.net/api/v1/road-safety"
call [Web1 -> Get]

when [Web1 -> Content]
url responseCode responseType responseContent
do set [item data -> Text] to look up in pairs key "Temperature"
pairs call [Web1 -> JsonTextDecode] jsonText get responseContent
notFound
set [Label4 -> Text] to look up in pairs key "Humidity"
pairs call [Web1 -> JsonTextDecode] jsonText get responseContent
notFound
set [Label12 -> Text] to look up in pairs key "Speed"
pairs call [Web1 -> JsonTextDecode] jsonText get responseContent
notFound
set [item data2 -> Text] to look up in pairs key "Visibility"
pairs call [Web1 -> JsonTextDecode] jsonText get responseContent
notFound
set [Label1 -> Text] to look up in pairs key "Message"
pairs call [Web1 -> JsonTextDecode] jsonText get responseContent
notFound

Show Warnings

Media

Upload File ...

The screenshot displays a mobile application development environment with four main panels:

- Palette**: A sidebar on the left containing a search bar and a list of UI components: Button, CheckBox, DatePicker, Image, Label, ListPicker, ListView, Notifier, PasswordTextBox, Slider, Spinner, Switch, TextBox, TimePicker, and WebViewer. The "Layout" item at the bottom is currently selected.
- Viewer**: The central area showing a simulated smartphone screen with a "Home" tab bar. The main content area is divided into four sections:
 - Top-left: "Temperature" and "Text for Label2".
 - Top-right: "Humidity" and "Text for Label4".
 - Middle: "Speed" and "Text for Label12".
 - Bottom: "Visibility" and "Message".
- Components**: A tree view on the right showing the hierarchical structure of the application's components. It includes a "Screen1" node with nested "VerticalArrangement1" and "HorizontalArrangement1" nodes, which further contain "Label1", "Label3", "Label4", "Label5", "Label12", and "Label2".
- Properties**: A panel on the far right listing various properties for the selected component:
 - Screen1
 - AboutScreen
 - AccentColor (Default)
 - AlignHorizontal (Left: 1)
 - AlignVertical (Top: 1)
 - AppName (RoadSafety)
 - BackgroundColor (Default)
 - BackgroundImage (None)
 - BigDefaultText (None)
 - BlocksToolkit (All +)
 - CloseScreenAnimation (Default)
 - DefaultFileScope (App)
 - HighContrast (None)