**Project Development Phase**

**Project Development Delivery of Sprint 3**

| | |
|---|---|
| **Team ID** | PNT2022TMID41963 |
| **Project Name** | Project - Signs with smart connectivity for Better road safety |
| **Marks** | 8 Marks |

**Signs with smart connectivity for Better road safety**

## Objective :

>> Write a python code for print the random temperature, Road signs, Speed limit, Message

>> Simulate and Generate the data

>> Display the published data in IBM Watson IOT Platform

## Code for print the random temperature, Road signs, Speed limit, Message :

### ( RandomValues.py )

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

myConfig = {
        #Configuration
        "identity": {
                "orgId": "7znh86",
                "typeId": "NODE",
                "deviceId":"1234"
        },
        #API Key
        "auth": {
                "token": "123456789"
        }
}

#Receiving callbacks from IBM IOT platform def myCommandCallback(cmd): print("Message received from
        IBM IoT Platform: %s" % cmd.data['command'])
        m=cmd.data['command'] client = wiotp.sdk.device.DeviceClient(config=myConfig,
logHandlers=None) client.connect()

#OpenWeatherMap Credentials
BASE_URL = "https://api.openweathermap.org/data/2.5/weather?"
CITY = "Salem, IN"
URL = BASE_URL + "q=" + CITY + "&units=metric"+"&appid=" + "f58e4720c739a54c439aba9b05176839" 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']

    #messge part msg=random.randint(0,5) if msg==1: message="GO SLOW,
    SCHOOL / COLLEGE 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            Limit     part
    speed=random.randint(0,150)        if speed>=100:
        speedMsg=" Limit Exceeded"
    elif speed>=60 and speed<100:
        speedMsg="Moderate"else: speedMsg="Slow"

      #Diversion       part 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=""

       #Visibility if temperature < 24: visibility="Fog
      Ahead, Drive Slow"
      elif temperature < 20:
          visibility="Bad Weather"
      else: visibility="Clear Weather"
  else:
      print("Error in the HTTP request") myData={'Temperature':temperature, 'Message':message,
          'Sign':signMsg, 'Speed':speedMsg,
'Visibility':visibility} client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0, onPublish=None)
#PUBLISHING          TO  IOT  WATSON  print("Published  data
    Successfully: ", myData) print("
                                                
```

*PYTHON CODE.py - C:/Users/pc/AppData/Local/Programs/Python/Python36-32/PYTHON CODE.py (3.6.5)*

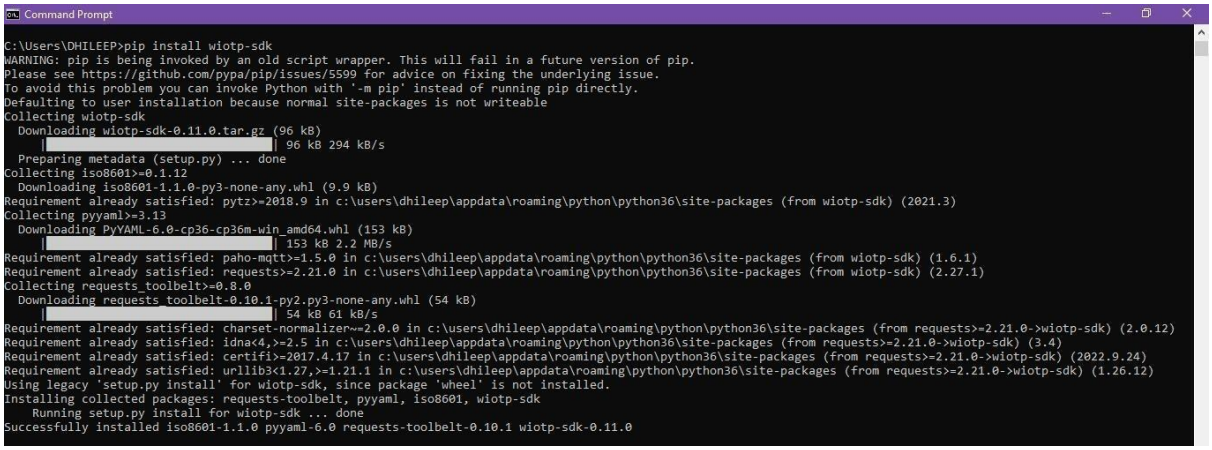
File Edit Format Run Options Window Help

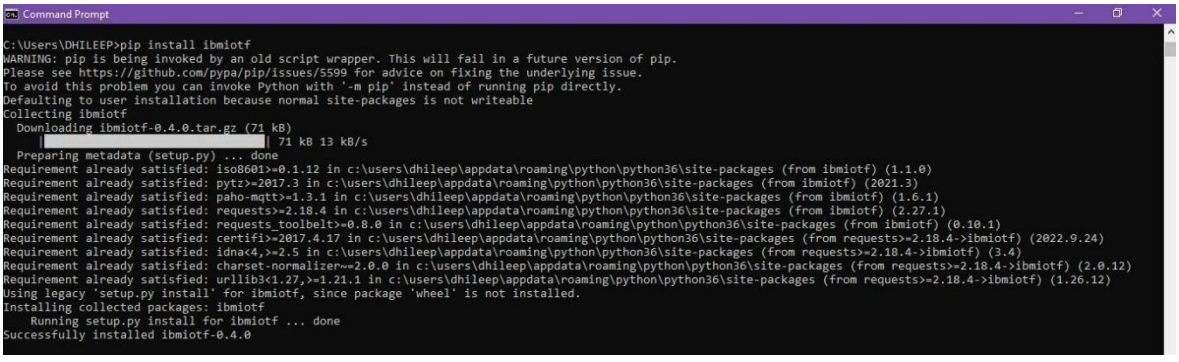
```
#IBM Watson IOT Platform
#pip install wiotp-sdk
import wiotp.sdk.device
import time
import random
myConfig = {
    "identity": {
        "orgId": "7znh86",
        "typeId": "NODE",
        "deviceId":"1234"
    },
    "auth": {
        "token": "123456789"
    }
}

def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
    m=cmd.data['command']

client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()
```

## Import wiotp-sdk & ibmiotf :




Python IDLE Output :

*Python 3.6.5 Shell*
File Edit Shell Debug Options Window Help
Published data Successfully: {'Temperature': 26.03, 'Message': 'EMERGENCY, HOSPITAL NEARBY', 'Sign': 'Speed Breaker', 'Speed': ' Limit Exceeded', 'Visibility': 'Clear Weather'}
Published data Successfully: {'Temperature': 26.03, 'Message': 'GO SLOW, SCHOOL / COLLEGE ZONE AHEAD', 'Sign': 'Right Diversion', 'Speed': 'Moderate', 'Visibility': 'Clear Weather'}
Published data Successfully: {'Temperature': 26.03, 'Message': 'PETROL BUNK NEARBY', 'Sign': 'Speed Breaker', 'Speed': ' Limit Exceeded', 'Visibility': 'Clear Weather'}
Published data Successfully: {'Temperature': 26.03, 'Message': 'EMERGENCY, HOSPITAL NEARBY', 'Sign': 'Speed Breaker', 'Speed': 'Slow', 'Visibility': 'Clear Weather'}
Published data Successfully: {'Temperature': 26.03, 'Message': '', 'Sign': '', 'Speed': ' Limit Exceeded', 'Visibility': 'Clear Weather'}
Published data Successfully: {'Temperature': 26.03, 'Message': 'EMERGENCY, HOSPITAL NEARBY', 'Sign': '', 'Speed': 'Moderate', 'Visibility': 'Clear Weather'}
Published data Successfully: {'Temperature': 26.03, 'Message': 'EMERGENCY, HOSPITAL NEARBY', 'Sign': '', 'Speed': 'Slow', 'Visibility': 'Clear Weather'}
Published data Successfully: {'Temperature': 26.03, 'Message': 'NEED HELP, POLICE STATION AHEAD', 'Sign': 'Left Diversion', 'Speed': 'Moderate', 'Visibility': 'Clear Weather'}