FINAL CODE

Date	**19 November 2022**
Team ID	**PNT2022TMID23164**
Project	**Signs with Smart Connectivity for Better Road Safety**

PROGRAM CODE:

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

myConfig = { #Configuration
    "identity": {
"orgId": “fvh76j”,
"typeId": "SMARTBOARD",
"deviceId":"SMARTCONNECTIVITY"},
#API Key
"auth": {
"token": "12345678"
}
}

#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.commandCallback= myCommandCallback
client.connect()

#OpenWeatherMap Credentials
BASE_URL ="https://api.openweathermap.org/data/2.5/weather?"
CITY = "Chennai"
URL = BASE_URL + "q=" + CITY + "&units=metric"+"&appid=" +
" aacfd527963a5d91a8b5db80c6fe67b4"

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="SLOW DOWN, SCHOOL IS NEAR"
elif msg==2:
        message="NEED HELP, POLICE STATION AHED"
elif msg==3:
        message="EMERGENCY, HOSPITAL NEARBY"
elif msg==4:
        message="DINE IN, RESTAURENT AVAILABLE"
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==3:
        signMsg="Left Diversion"
elif sign==5:
        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: %s", myData)
  client.commandCallback= myCommandCallback
time.sleep(5)
```

client.disconnect()

OUTPUT: