

PROJECT DEVELOPMENT PHASE
SPRINT – 1(USN-2)

TEAM ID	PNT2022TMID48694
PROJECT NAME	SIGNS WITH SMART CONNECTIVITY FOR BETTER ROAD SAFETY

PYTHON CODE :

#OPENWEATHER MAP(SPRINT 1)

import wiotp.sdk.device #importing library files for connecting with
CLOUD,sdk=software developement kit

import requests #for API request

import json #converting it to json(key:values)

myConfig = {

 "identity": {

 "orgId": "7f5hee",

 "typeId": "testdevicetype", #configuration wit CLOUD,finding identity

 "deviceId":"12345"

 },

 "auth": {

 "token": "AQCLi6rYJrcoiDpW6?" #authenticating with cloud device

 }

}

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

#initialising device client with above myconfig detail

client.connect()

while True:

 print("=====")

```
weatherData =  
requests.get('https://api.openweathermap.org/data/2.5/weather?q=Chennai,%20IN&appid=b966927276060e981c650a5ca4409f8b&units=metric')
```

```
a=weatherData.text
```

```
b=json.loads(a)
```

```
temp = b["main"]["temp"]
```

```
humi = b["main"]["humidity"]
```

```
main = b["weather"][0]["main"]    #0th index is taken from the object
```

```
description = b["weather"][0]["description"]
```

```
visibility = b["visibility"]
```

```
TemperatureRecommendation = ""
```

```
SpeedRecommendation = ""
```

```
RecommendationForVisibilty= ""
```

```
#print("Temperature(celcius) :",b["main"]["temp"])
```

```
if (temp>33):
```

```
    TemperatureRecommendation="Temperature is higher than ideal value"
```

```
    #print("Temperature is higher than ideal value")
```

```
elif (temp<19):
```

```
    TemperatureRecommendation="Temperature is lower than ideal value"
```

```
    #print("Temperature is lower than ideal value")
```

```
else:
```

```
    TemperatureRecommendation="Temperature is ideal"
```

```
    #print("Temperature is ideal ")
```

```

#print("Humidity :",b["main"]["humidity"])
#print("WeatherCondition",(b["weather"][0]["main"]))
if (main == "Rain"):
    rain = b["rain"]["1h"]
    SpeedRecommendation = "30KM/HR ,ROAD WILL BE SLIPPERY"
    #print("Rain:",b["rain"]["1h"])
    #print("SPEED RECOMMENDATION : 30KM/HR ,ROAD WILL BE SLIPPERY")
elif (main == "Drizzle"):
    SpeedRecommendation = "30KM/HR"
    #print("SPEED RECOMMENDATION : 30KM/HR")
elif (main == "Mist"):
    SpeedRecommendation = "30KM/HR and switch on the headlight"
    #print("SPEED RECOMMENDATION : 30KM/HR and switch on the Headlight")

#print("Description of weather :", (b["weather"][0]["description"]))
#print("visibility", (b["visibility"]))
if (visibility<1000):
    RecommendationForVisibilty = "SPEED RECOMMENDATION : 30KM/HR and SWITCH ON THE HEAD LIGHT"
else:
    RecommendationForVisibilty = "visibility range is ideal for vechicles"

#print("SPEED RECOMMENDATION : 30KM/HR and SWITCH ON THE HEAD LIGHT")

mydata={"temperature":temp,
"TemperatureRecommendation":TemperatureRecommendation,"humidity":humidity,"WeatherCondition":main,"SpeedRecommendation":SpeedRecommendation

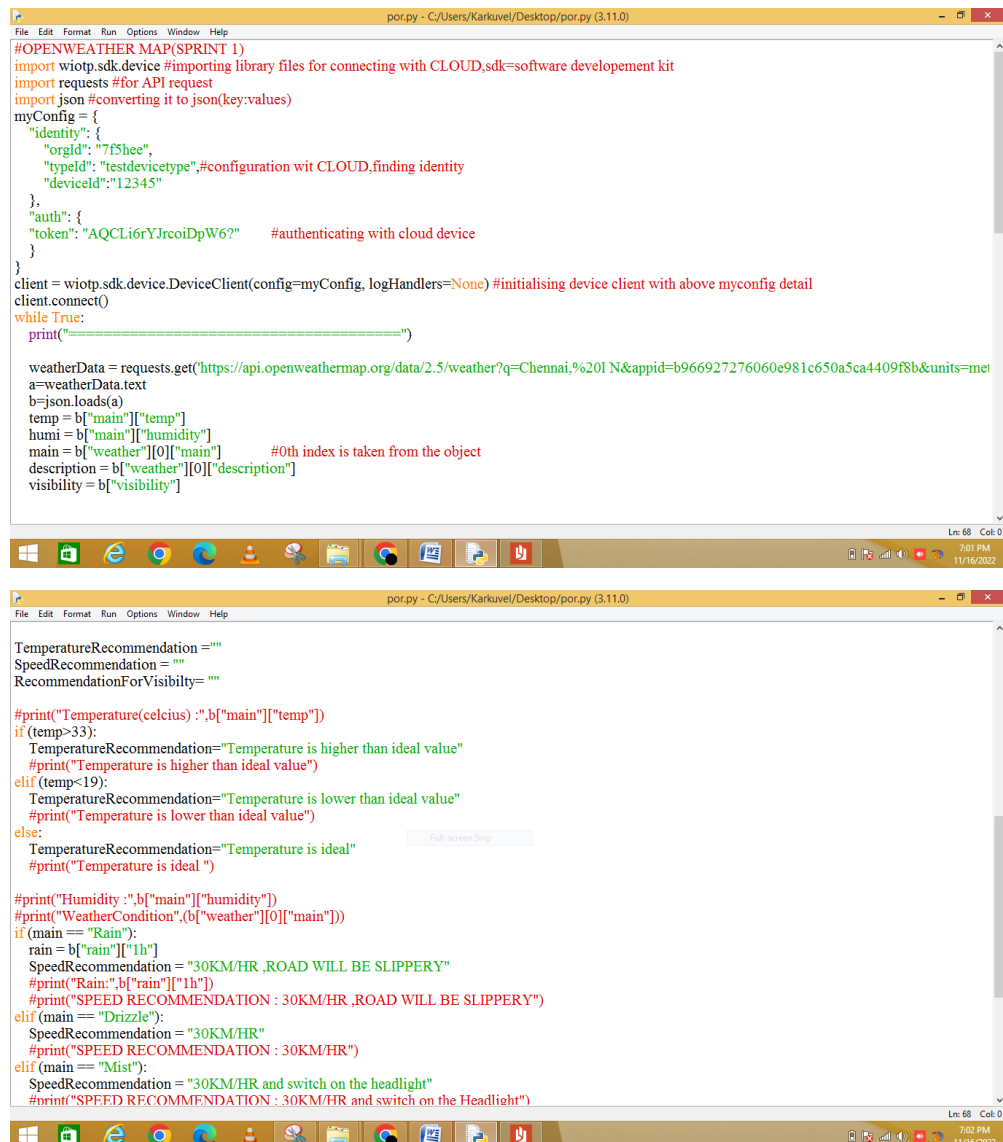
```

```
, "DescriptionOfWeather": description, "visibility": visibility, "RecommendationFor  
rVisibilty": RecommendationForVisibilty}
```

```
print(mydata)
```

```
client.publishEvent("12345", "json", mydata)
```

CODE IN PYTHON IDLE



The image displays two screenshots of a Python IDE window titled 'por.py - C:/Users/Karkuvel/Desktop/por.py (3.11.0)'. The top screenshot shows the initial code setup for connecting to a cloud device and fetching weather data from an API. The bottom screenshot shows the logic for processing the fetched data and providing recommendations based on temperature, humidity, and weather conditions.

```
#OPENWEATHER MAP(SPRINT 1)
import wiotp.sdk.device #importing library files for connecting with CLOUD,sdk=software development kit
import requests #for API request
import json #converting it to json(key:values)
myConfig = {
    "identity": {
        "orgId": "7f5hee",
        "typeId": "testdevicetype", #configuration wit CLOUD,finding identity
        "deviceId": "12345"
    },
    "auth": {
        "token": "AQCLi6rYJrcoiDpW6?" #authenticating with cloud device
    }
}
client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None) #initialising device client with above myconfig detail
client.connect()
while True:
    print("=====")

    weatherData = requests.get('https://api.openweathermap.org/data/2.5/weather?q=Chennai,%20IN&appid=b966927276060e981c650a5ca4409f8b&units=metric')
    a=weatherData.text
    b=json.loads(a)
    temp = b["main"]["temp"]
    humi = b["main"]["humidity"]
    main = b["weather"][0]["main"] #0th index is taken from the object
    description = b["weather"][0]["description"]
    visibility = b["visibility"]

    TemperatureRecommendation = ""
    SpeedRecommendation = ""
    RecommendationForVisibilty= ""

    #print("Temperature(cecius) :",b["main"]["temp"])
    if (temp>33):
        TemperatureRecommendation="Temperature is higher than ideal value"
        #print("Temperature is higher than ideal value")
    elif (temp<19):
        TemperatureRecommendation="Temperature is lower than ideal value"
        #print("Temperature is lower than ideal value")
    else:
        TemperatureRecommendation="Temperature is ideal"
        #print("Temperature is ideal ")

    #print("Humidity :",b["main"]["humidity"])
    #print("WeatherCondition",(b["weather"][0]["main"]))
    if (main == "Rain"):
        rain = b["rain"]["1h"]
        SpeedRecommendation = "30KM/HR ,ROAD WILL BE SLIPPERY"
        #print("Rain:",b["rain"]["1h"])
        #print("SPEED RECOMMENDATION : 30KM/HR ,ROAD WILL BE SLIPPERY")
    elif (main == "Drizzle"):
        SpeedRecommendation = "30KM/HR"
        #print("SPEED RECOMMENDATION : 30KM/HR")
    elif (main == "Mist"):
        SpeedRecommendation = "30KM/HR and switch on the headlight"
        #print("SPEED RECOMMENDATION : 30KM/HR and switch on the Headlight")
```

