SPRINT 1 (CODING)

TEAMID	PNT2022TMID01063
PROJECT NAME	SIGNS WITH SMART CONNECTIVITY FOR BETTER ROAD SAFETY

BRAIN.PY

```
#python code
import weather
from datetime import datetime as dt



# ------------------------------------------------
# UTILITY LOGIC SECTION STARTS
def processConditions(myLocation,APIKEY,localityInfo):
    weatherData = weather.get(myLocation,APIKEY)


    finalSpeed = localityInfo["usualSpeedLimit"] if "rain" not in weatherData else
localityInfo["usualSpeedLimit"]/2
    finalSpeed = finalSpeed if weatherData["visibility"]>35 else finalSpeed/2


    if(localityInfo["hospitalsNearby"]):
        # hospital zone
        doNotHonk = True
    else:
        if(localityInfo["schools"]["schoolZone"]==False):
            # neither school nor hospital zone
```

```
            doNotHonk = False
        else:
            # school zone
            now = [dt.now().hour,dt.now().minute]
            activeTime = [list(map(int,_.split(":"))) for _ in localityInfo["schools"]["activeTime"]]
            doNotHonk = activeTime[0][0]<=now[0]<=activeTime[1][0] and
activeTime[0][1]<=now[1]<=activeTime[1][1]

    return({
        "speed" : finalSpeed,
        "doNotHonk" : doNotHonk
    })

# UTILITY LOGIC SECTION ENDS
```

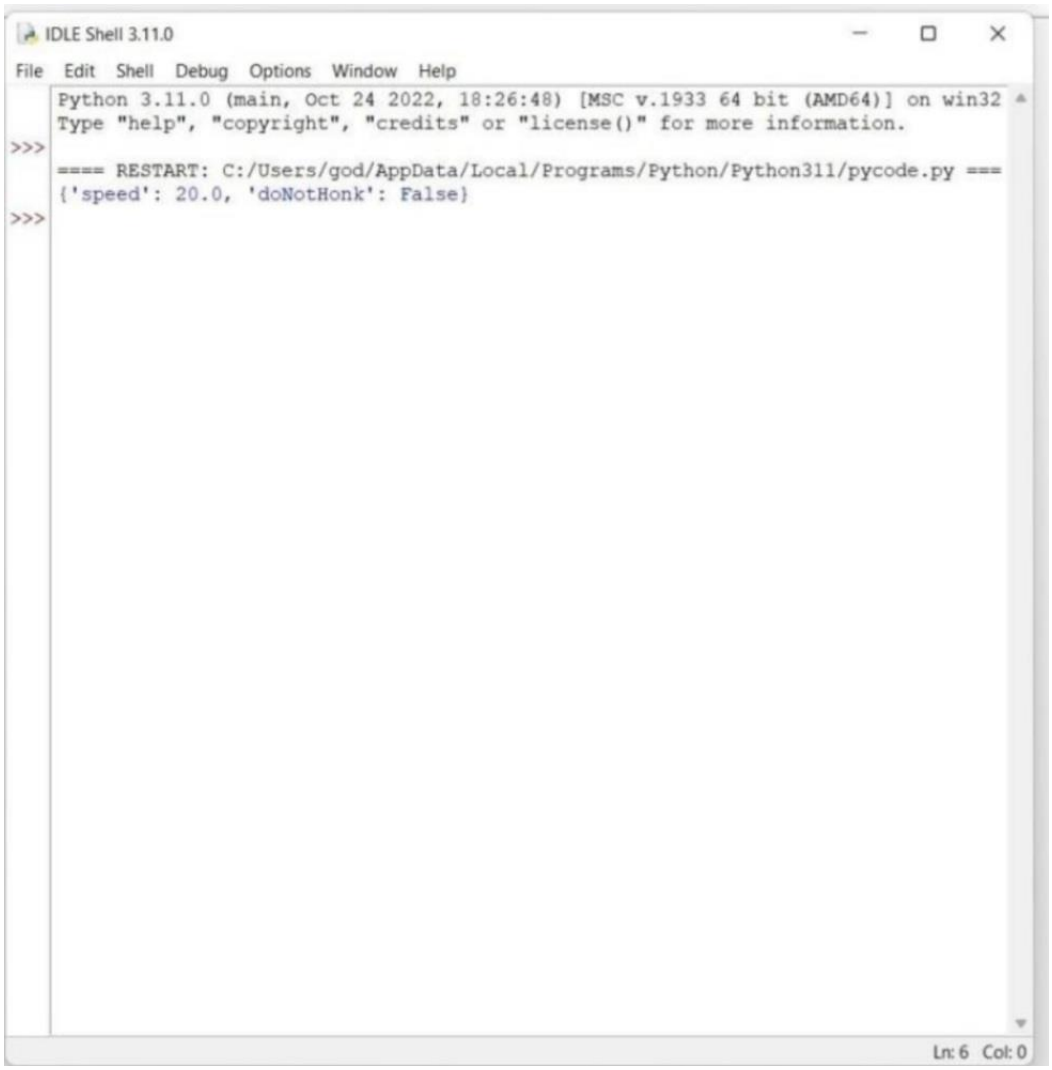
WEATHER.PY

```
# Python code

import requests as reqs

def get(myLocation,APIKEY):
    apiURL = f"https://api.openweathermap.org/data/2.5/weather?q={myLocation}&appid={APIKEY}"
    responseJSON = (reqs.get(apiURL)).json()
    returnObject = {
        "temperature" : responseJSON['main']['temp'] - 273.15,
        "weather" : [responseJSON['weather'][_]['main'].lower() for _ in
range(len(responseJSON['weather']))],
        "visibility" : responseJSON['visibility']/100, # visibility in percentage where 10km is 100% and 0km is
0%
    }
```

```
    if("rain" in responseJSON):
        returnObject["rain"] = [responseJSON["rain"][key] for key in responseJSON["rain"]]
    return(returnObject)
```

MAIN.PY

```
# IMPORT SECTION STARTS

import brain

# IMPORT SECTION ENDS
# ---------------------------------------------
# USER INPUT SECTION STARTS

myLocation = "Chennai,IN"
APIKEY = "bf4a8d480ee05c00952bf65b78ae826b"

localityInfo = {
    "schools" : {
        "schoolZone" : True,
        "activeTime" : ["7:00","17:30"] # schools active from 7 AM till 5:30 PM
        },
    "hospitalsNearby" : False,
    "usualSpeedLimit" : 40 # in km/hr
}

# USER INPUT SECTION ENDS
```



PUBLISH DATA CODE:

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



#Provide your IBM Watson Device Credentials
organization = "8dxkha"
```

```
deviceType = "madhu"

deviceId = "madhu"

authMethod = "use-token-auth"

authToken = "yah&46&uqf!k4Rq!n+"


# Initialize GPIO


temp=60

pulse=70

oxygen= 30

lat =  17

lon = 18



def myCommandCallback(cmd):

   print("Command received: %s" % cmd.data['command'])

   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 application from the cloud

deviceCli.disconnect()
```

OUTPUT SNAPSHOTS:

subs.py - C:/Users/revan/AppData/Local/Programs/Python/Python37/subs.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 = "8dxkha"
deviceType = "madhu"
deviceId = "madhu"
authMethod = "use-token-auth"
authToken = "yah&46&uqf!k4Rq!n"

# Initialize GPIO

temp=60
pulse=70
oxygen= 30
lat =  17
lon = 18


def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    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")
```

Ln: 12 Col: 31