PYTHON SCRIPT:

```
#IBM Watson IOT Platform
#pip install wiotp-sdk
import wiotp.sdk.device
import time
import random
myConfig = {
    "identity": {
        "orgId": "afblzo",
        "typeId": "raspberrypi",
        "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()


while True:
    temperature=random.randint(-20,125)
    vehiclescount=random.randint(0,100)
    workingarea_distance=random.randint(0,60)
    accidentalarea_distance=random.randint(1,25)
    p="Your Prefered Speed"
    q="Speed Limit is 30 km\hr"
    r="Take another route"
    s="As Your Wish"
    t="Go Slow"
    u="Moderate speed"
    v="it's accidental area, Be Carefull"
```

```
    w="Beyond the accidental area! Have a safe journey"

    a={'Condition_for_Speed':p}
    b={'Condition_for_Speed':q}
    c={'Condition_for_Direction':r}
    d={'Condition_for_Direction':s}
    e={'Cond_for_Speed':t}
    f={'Cond_for_Speed':u}
    g={'Condition_for_Drive':v}
    h={'Condition_for_Drive':w}



    myData1={'Temperature':temperature}
    myData2={'Vehiclescount':vehiclescount}

myData3={'WorkingArea_Distance':workingarea_dista
nce}

myData4={'AccidentalArea_Distance':accidentalarea_di
stance}
```

```
client.publishEvent(eventId="status",msgFormat="json
",data=myData1,qos=0,onPublish=None)
    print("Published:%s",myData1)
    if temperature>=21:


        client.publishEvent(eventId="status",
msgFormat="json",data=a,qos=0,onPublish=None)
        print(a)
        print("\n")
    else :
        client.publishEvent(eventId="status",
msgFormat="json",data=b,qos=0,onPublish=None)
        print(b)
        print("\n")



client.publishEvent(eventId="status",msgFormat="json
",data=myData2,qos=0,onPublish=None)
    print("Published:%s",myData2)
    if vehiclescount>=53:
```

```
client.publishEvent(eventId="status",msgFormat="json
",data=c,qos=0,onPublish=None)
        print(c)
        print("\n")
    else:

client.publishEvent(eventId="status",msgFormat="json
",data=d,qos=0,onPublish=None)
        print(d)
        print("\n")



client.publishEvent(eventId="status",msgFormat="json
",data=myData3,qos=0,onPublish=None)
    print("Published:%s",myData3)
    if workingarea_distance>=4:
        client.publishEvent(eventId="status",
msgFormat="json",data=f,qos=0,onPublish=None)
        print(f)
        print("\n")
    else :
```

```
        client.publishEvent(eventId="status",
msgFormat="json",data=e,qos=0,onPublish=None)
        print(e)
        print("\n")


client.publishEvent(eventId="status",msgFormat="json
",data=myData4,qos=0,onPublish=None)
    print("Published:%s",myData4)
    if accidentalarea_distance>=3:
        client.publishEvent(eventId="status",
msgFormat="json",data=h,qos=0,onPublish=None)
        print(h)
        print("\n")
    else :
        client.publishEvent(eventId="status",
msgFormat="json",data=g,qos=0,onPublish=None)
        print(g)
        print("\n")
```

```
    client.commandCallback=myCommandCallback
    time.sleep(10)
client.disconnect()
```

OUTPUT: