DEVELOP A PYTHON SCRIPT

TEAM ID	PNT2022TMID14066
PROJECT NAME	Signs with Smart Connectivity for Better Road Safety

STEPS TO DEVELOP A PYTHON SCRIPT:

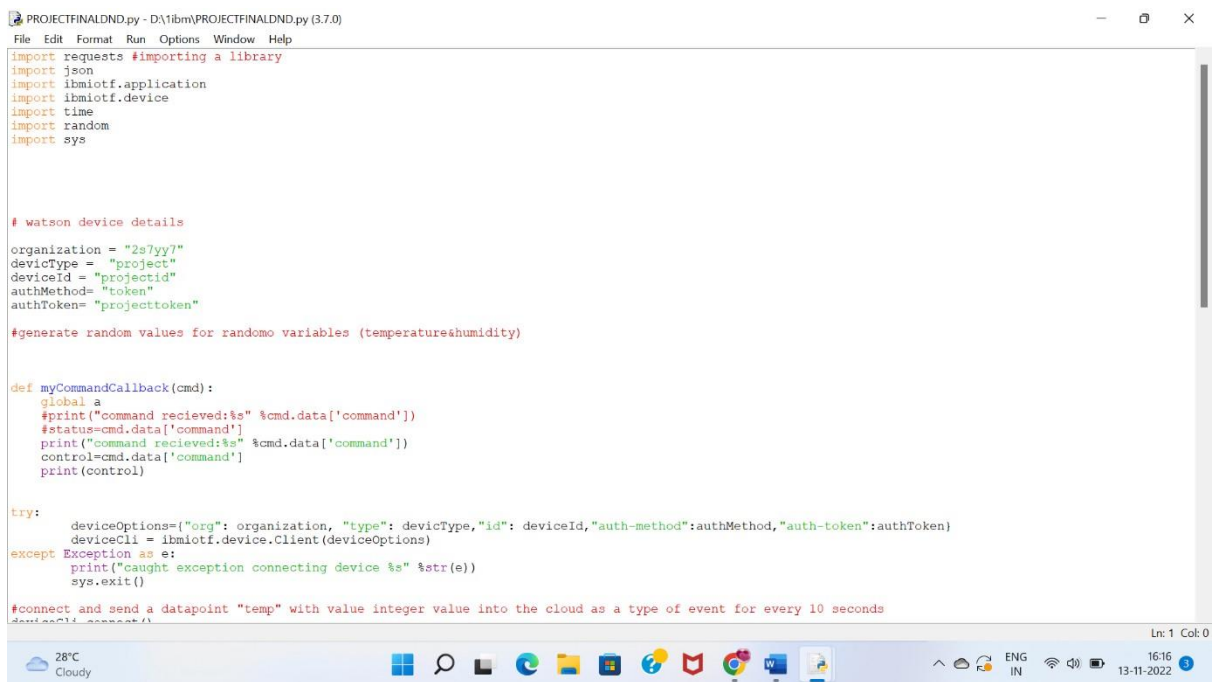**STEP-1:** Install Python IDLE 3.7 from any browser.

STEP-2: Open PYTHON IDLE. Click on File->New file. A new screen opens.

STEP-3: Develop a python program to satisfy the required needs.

STEP-4: Save the file and click Run-> Run Module.

STEP-5: The program gets executed and it reports any error if present, else produces the output.

PYTHON CODE:


```python
import requests #importing a library
import json
import ibmiotf.application
import ibmiotf.device
import time
import random
import sys


# watson device details

organization = "2s7yy7"
devicType = "project"
deviceId = "projectid"
authMethod= "token"
authToken= "projecttoken"

#generate random values for randomo variables (temperature&humidity)


def myCommandCallback(cmd):
    global a
    #print("command recieved:%s" %cmd.data['command'])
    #status=cmd.data['command']
    print("command recieved:%s" %cmd.data['command'])
    control=cmd.data['command']
    print(control)


try:
    deviceOptions={"org": organization, "type": devicType,"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 "temp" with value integer value into the cloud as a type of event for every 10 seconds
deviceCli.connect()
```

```python
#connect and send a datapoint "temp" with value integer value into the cloud as a type of event for every 10 seconds
deviceCli.connect()

while True:

#get sensor data from DHT11


    a = "https://api.openweathermap.org/data/2.5/weather?q=Chennai,%20IN&appid=e2bea247ed9ad643a04d9a8e55499d5f"
    r=requests.get(url=a)
    data=r.json()

    Temp= data['main']['temp']
    Humd= data['main']['humidity']
    data= {'temp':Temp,'humid':Humd}
    dist=random.randint(0,20)
    dis={'dista':dist}

    if(Humd<100):
        warn={'alert':'PLEASE SLOW DOWN!!!!!!'}
    if(dist<20):
        insta={'inst':'stop'}


    def myOnPublishCallback():

        print("published Temperature = %s c" %Temp,"humidity:%s %%" %Humd)
        print(warn)
        print(dis)




    success=deviceCli.publishEvent ("IoTSensor","json",insta,qos=0,on_publish= myOnPublishCallback)
    success=deviceCli.publishEvent ("IoTSensor","json",data,qos=0,on_publish= myOnPublishCallback)
```

```python
    Humd= data['main']['humidity']
    data= {'temp':Temp,'humid':Humd}
    dist=random.randint(0,20)
    dis={'dista':dist}

    if(Humd<100):
        warn={'alert':'PLEASE SLOW DOWN!!!!!!'}
    if(dist<20):
        insta={'inst':'stop'}


    def myOnPublishCallback():

        print("published Temperature = %s c" %Temp,"humidity:%s %%" %Humd)
        print(warn)
        print(dis)




    success=deviceCli.publishEvent ("IoTSensor","json",insta,qos=0,on_publish= myOnPublishCallback)
    success=deviceCli.publishEvent ("IoTSensor","json",data,qos=0,on_publish= myOnPublishCallback)
    success=deviceCli.publishEvent ("IoTSensor","json",warn,qos=0,on_publish= myOnPublishCallback)
    success=deviceCli.publishEvent ("IoTSensor","json",dis,qos=0,on_publish= myOnPublishCallback)

    if not success:
        print("not connected to ibmiot")
    time.sleep(1)



    deviceCli.commandCallback=myCommandCallback
#disconnect the device
deviceCli.disconnect()
```