Team ID	PNT2022TMID32813
Project Name	Project - INDUSTRY-SPECIFIC INTELLIGENCE FIRE MANAGEMENT SYSTEM

DEVELOPING A PYTHON CODE

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
organization="1s2adz"
deviceType="ardiuno"
deviceId="0910"
authMethod="token"
authToken="12345678"
def myCommandCallback(cmd):
    print("command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="sprinkleron":
        print("Sprinkler is on")
    else:
        print("Sprinkler is off")
try:
    deviceOptions ={"org": organization, "type": deviceType, "id": deviceId, "auth-method": authMethod, "auth-token": authToken}
    deviceCli = ibmiotf.device.Client(deviceOption
except Exception as e:
    print("caught exception connecting device: %s" % str(e))
    sys.exit()
deviceCli.connect()
while True:
```

```
    temp=random.randint(90,110)
    Humid=random.randint(60,100)
    data={'temp':temp,'Humid':Humid}
    def myOnPublishCallback():
        print("Publish Temperature = %s c" % temp,"Humidity = %s %%" % Humid,"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
deviceCli.disconnect()
```