DATE	17 NOV 2022
TEAM ID	PNT2022TMID48690
PROJECT NAME	HAZARDOUS AREA MONITORING FOR INDUSTRIAL PLANT

Python code to check temperature and humidity:

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

#Provide your IBM Watson Device Credentials
organization = "7xapma"
deviceType = "demo-1"
deviceId = "12345"
authMethod = "token"
authToken = "sq7FNdgf5rnYgpUs_E"

# Initialize GPIO
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="Alert":
        print ("Temprature and Humidity is High")
    else :
        print ("please send proper command")

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
    temp=random.randint(90,110)
    Humid=random.randint(60,100)
    data = { 'temp' : temp, 'Humid': Humid }
    #print data
    def myOnPublishCallback():
      print ("Published data of 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(10)
    deviceCli.commandCallback = myCommandCallback
# Disconnect the device and application from the cloud
```

```
deviceCli.disconnect()
```