# DEVELOPING A PYTHON SCRIPT

| | |
|---|---|
| Date | 20 OCTOBER 2022 |
| Team ID | PNT2022TMID52857 |
| Project Name | IoT-Based Safety Gadget for Child Safety Monitoring and Notification |

## Code for python script

```
import time
import sys
import ibmiotf.application
import ibmiotf.device import
random
#Provide your IBM Watson Device Credentialsorganization =
"awb990"
deviceType = "NodeMCU"
deviceId = "12345"
authMethod = "token"
authToken = "12345678"

# Initialize GPIO

def myCommandCallback(cmd):

    print("Command received: %s" % cmd.data['command'])status=cmd.data['command']
if  status=="lighton":  print
("led is on")
    elif  status==  "lightoff": print
      ("led is off")
    else :
```

```
    print ("please send proper 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 anevent of
type "greeting" 10 times
deviceCli.connect()
while True:
    #Get Sensor Data from DHT11
    temp=random.randint(90,100)
    Humid=random.randint(60,100)

    data = { 'temp' : temp, 'Humid': Humid }#print data
    def myOnPublishCallback():
        print ("Published 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
```

# Disconnect the device and application from the cloud

deviceCli.disconnect()

**OUTPUT:**

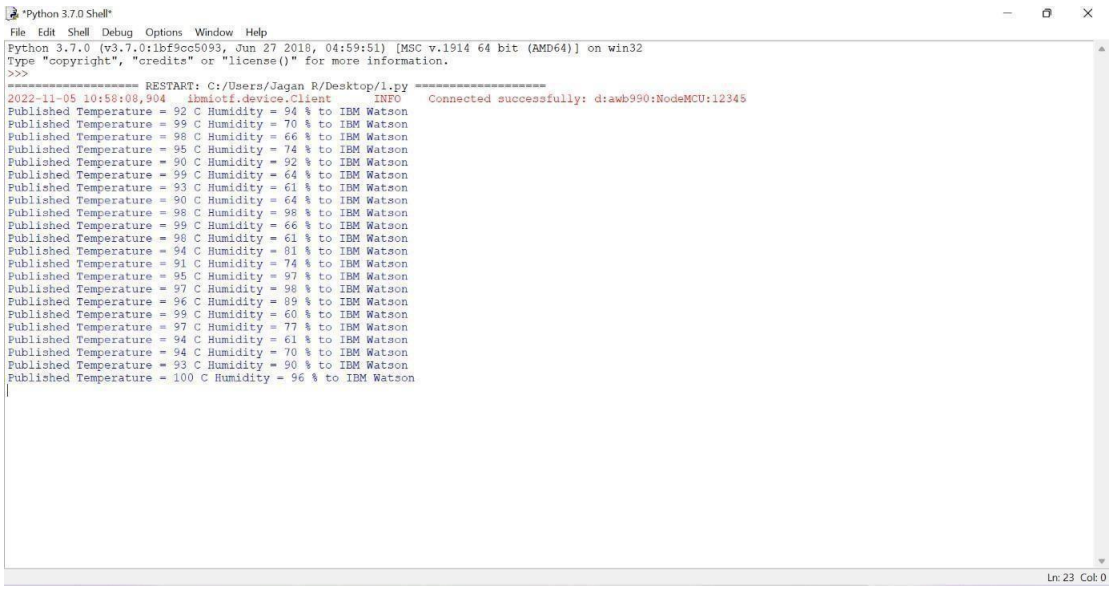


Figure 7.1 Output window

Figure 7.1 shows the output of the above code which executed in the Python software

## 7.1 Data publish to IBM cloud

WOKWI Simulation using ESP32 and Ultrasonic Sensor:

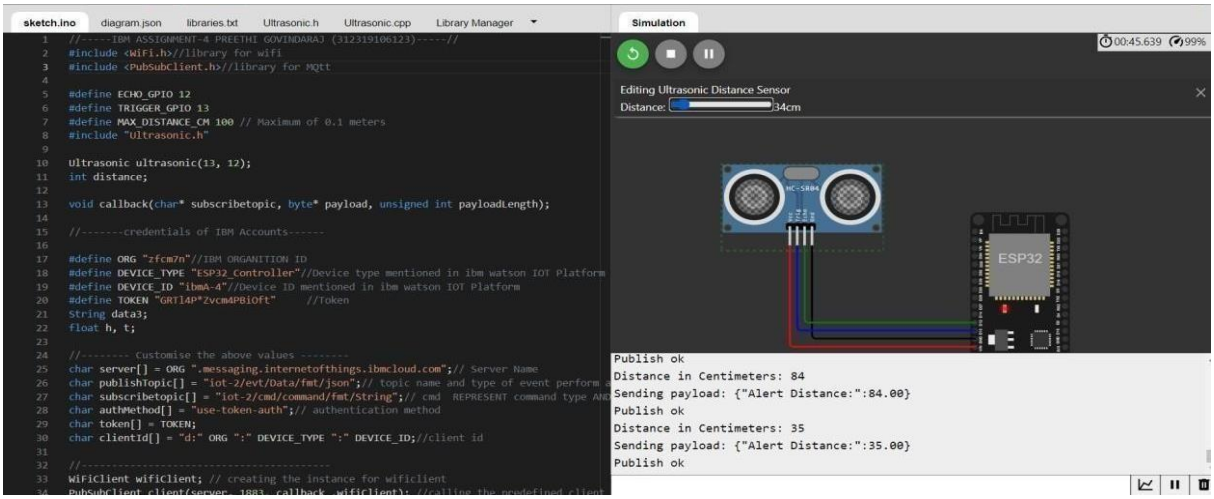


Figure 7.2 Wokwi simulation

Figure 7.2 shows the wokwi simulation of the ultrasonic sensor and wifi module

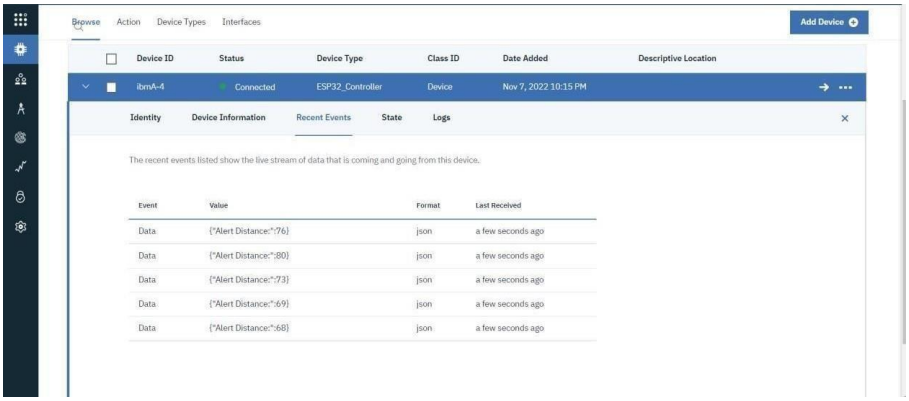


Figure 7.3 IBM Watson

Figure 7.3 shows the IBM Watson platform device-event log which shows different setof data.

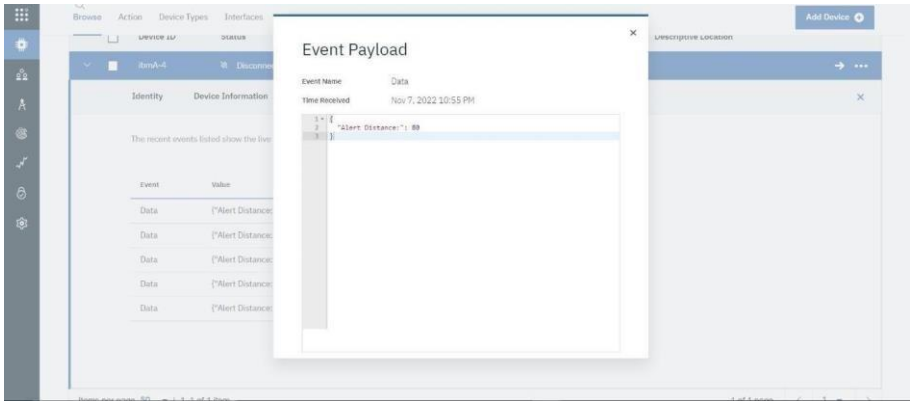


Figure 7.4 Event payload

Figure 7.4 shows the event payload that shows the alert system.