

Creating Python Code:

```
import time

import sys

import ibmiotf.application

import ibmiotf.device

import random

#Provide your IBM Watson Device
Credentials organization = "zwx6lb"
deviceType = "ABCD" deviceId = "13"
authMethod = "token" authToken =
"12345678"

#api key {a-illza1-mbdxqo6z0s}
#api token {zSYzISuAWF&F_x7GkT}

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
print("power      on      ")
```

```

print("checking connection to
waston  iot...")  time.sleep(2)
deviceCli.connect()
print("dear user ... welcome to IBM-IOT ")
print("i can provide your children live location and temperature ")
print()
name=str(input("enter your child name:"))
while True:

    temperature=random.randint(20,50)#random temperature for your child
    latitude=random.uniform(10.781377,10.78643)#random latitude for your child
    longitude=random.uniform(79.129113,79.134014)#random longitude for your child
    a="Child inside the geofence"      b=" Child outside the geofence"      c="High
temperature"      d="Low temperature"      x={'your_child_Zone':a}
y={'your_child_Zone':b}      z={'temp_condition':c}      w={'temp_condition':d}

    data = { 'temp' : temperature, 'lat': latitude, 'lon':longitude, 'name':name }

    #print data      def
myOnPublishCallback():
print ("Published Temperature
=  %s  C"  %  temperature,
"latitude = %s  %" % latitude,

```

```
"longitude = %s %%" % longitude, "to IBM Watson")
```

```
print("\n")
```

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

```
    if latitude>=10.78200 and latitude<=10.786000 and longitude >=79.130000 and longitude  
<=79.133000:
```

```
deviceCli.publishEvent("IoTSensorgpsdata","json",data=x,qos=0,on_publish=myOnPublishCallb  
ack)
```

```
    print(x)
```

```
print("\n")    else:
```

```
deviceCli.publishEvent("IoTSensorgpsdata","json",data=y,qos=0,on_publish=myOnPublishCallb  
ack)
```

```
    print(y)
```

```
print("\n")
```

```
    if (temperature>35):
```

```
deviceCli.publishEvent("IoTSensorgpsdata","json",data=z,qos=0,on_publish=myOnPublishCallb  
ack)
```

```
    print(c)
```

```
print("\n")    else:
```

```
deviceCli.publishEvent("IoTSensorgpsdata","json",data=w,qos=0,on_publish=myOnPublishCall  
back)
```

```
    print(d)
```

```
    print("\n")
```

```

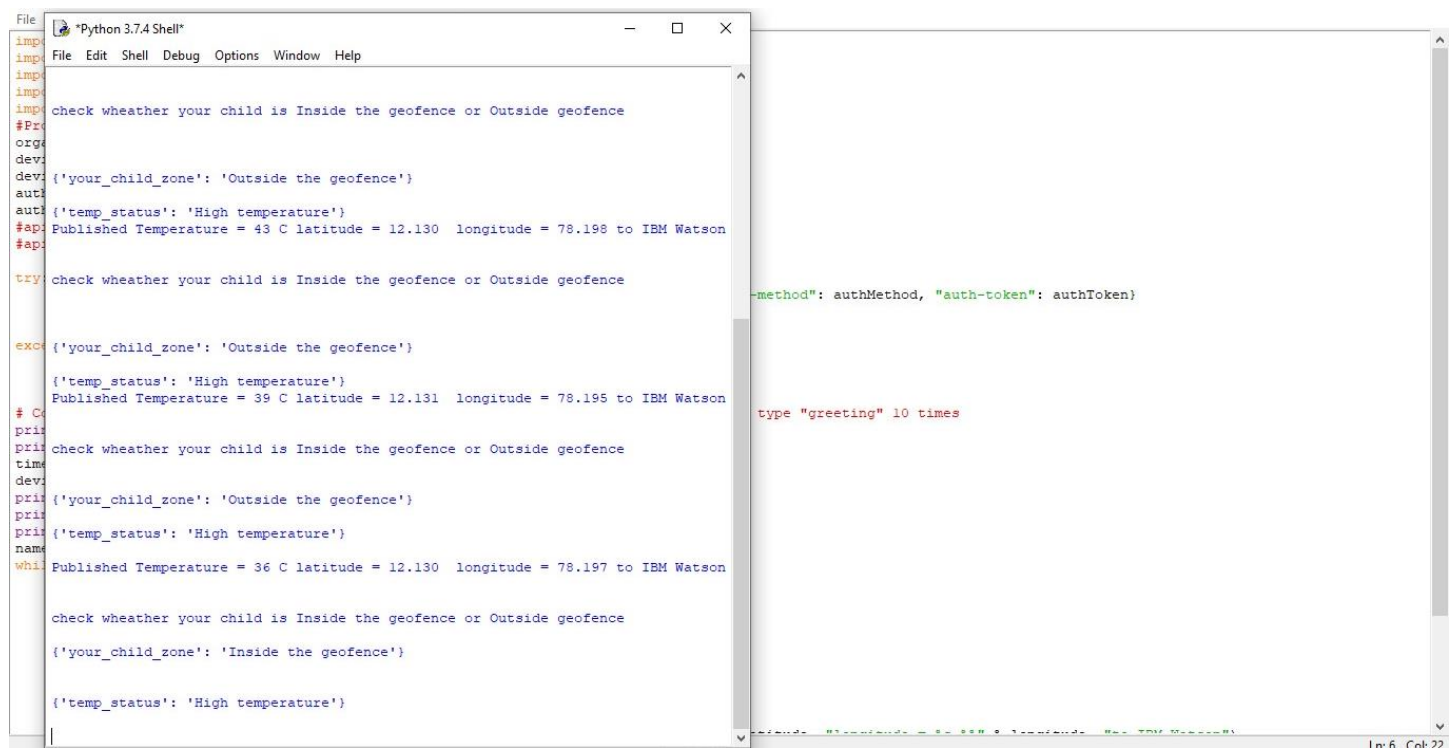
        if not success:
            print("Not
connected to IoT")

print("\n")
time.sleep(3)

# Disconnect the device and application from the cloud
deviceCli.disconnect()

```

Connecting IBM Watson and python Code:



The screenshot shows a Python 3.7.4 Shell window with the following code:

```

File Edit Shell Debug Options Window Help
import sys
import json
import requests
import time
import random

def check_wheather your child is Inside the geofence or Outside geofence
#Print the status of the device
orgId = "your_org_id"
devId = "your_dev_id"
authMethod = "your_auth_method"
authToken = "your_auth_token"

devInfo = {'your_child_zone': 'Outside the geofence'}
authInfo = {'temp_status': 'High temperature'}
#API URL
#API Key
#API Secret

try:
    check_wheather your child is Inside the geofence or Outside geofence
except:
    {'your_child_zone': 'Outside the geofence'}
    {'temp_status': 'High temperature'}
    Published Temperature = 39 C latitude = 12.131 longitude = 78.195 to IBM Watson

# Check the status of the device
print("check wheather your child is Inside the geofence or Outside geofence")
time.sleep(3)
devInfo = {'your_child_zone': 'Outside the geofence'}
print("your_child_zone: 'Outside the geofence'")
print("temp_status: 'High temperature'")
name = "your_device_name"
while True:
    Published Temperature = 36 C latitude = 12.130 longitude = 78.197 to IBM Watson

    check_wheather your child is Inside the geofence or Outside geofence
    {'your_child_zone': 'Inside the geofence'}

    {'temp_status': 'High temperature'}

```

The code defines a function to check if a child is inside or outside a geofence. It uses a dictionary to store device information and a loop to continuously check the status. The code also includes a print statement to display the status and a sleep function to delay the execution.



13	Connected	ABCD	Device	Nov 2, 2022 10:55 PM	→ ...
----	-----------	------	--------	----------------------	-------

Identity Device Information Recent Events State Logs

The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
IoTSensorgp...	{"temp_status":"High temperature"}	json	a few seconds ago
IoTSensorgp...	{"your_child_zone":"Outside the geofence"}	json	a few seconds ago
IoTSensorgp...	{"temp":50,"lat":12.132819998043411,"lon":78...	json	a few seconds ago
IoTSensorgp...	{"temp_status":"Low temperature"}	1 Simulation running	
IoTSensorgp...	{"your child zone":"Outside the geofence"}		