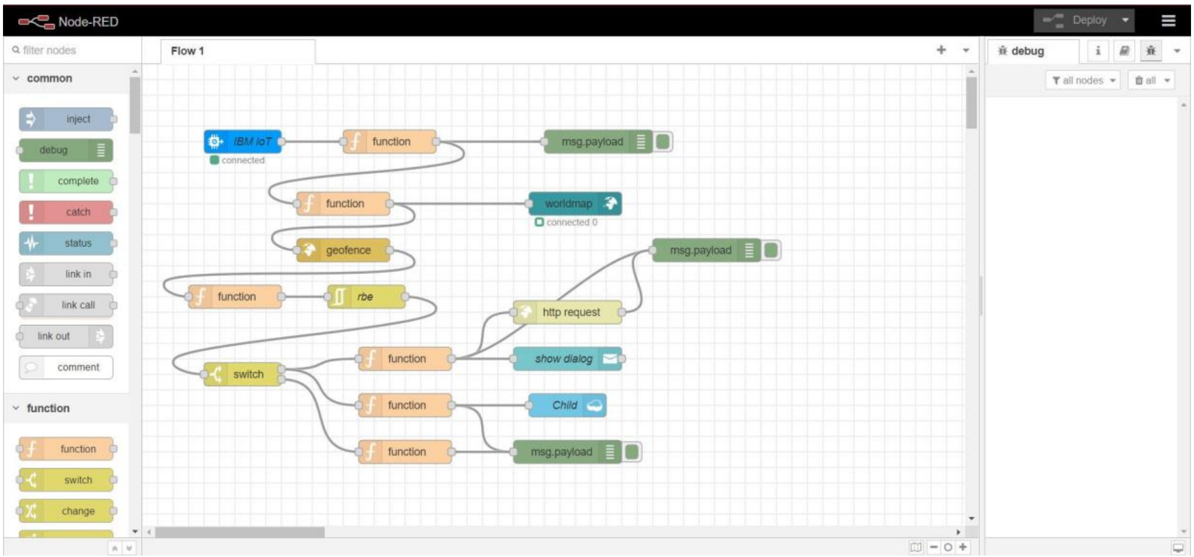
Project Development – Delivery plan sprint-3

IoT Based Safety Gadget for Child Safety Monitoring & Notification

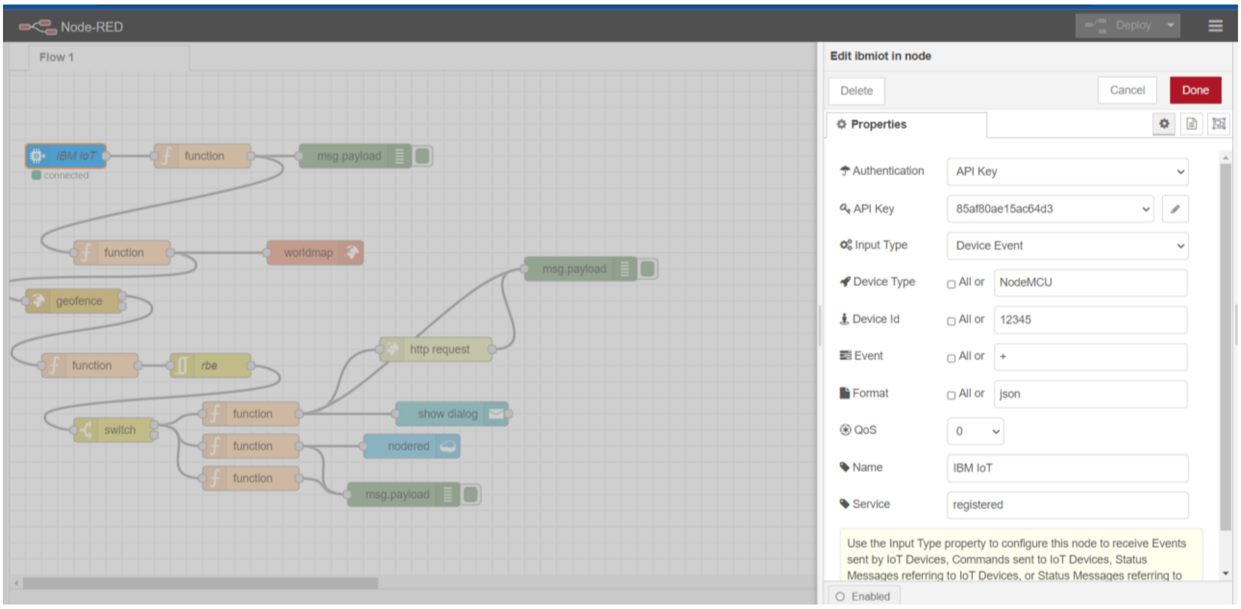
TEAM ID: PNT2022TMID04578

Creating Node-Red service and connecting with IBM cloud

Creating Node-Red service:



Codes in each Node:



Edit ibmiot in node

Delete Cancel Done

Properties

Authentication: API Key

API Key: 85af80ae15ac64d3

Input Type: Device Event

Device Type: □ All or NodeMCU

Device Id: □ All or 12345

Event: □ All or +

Format: □ All or json

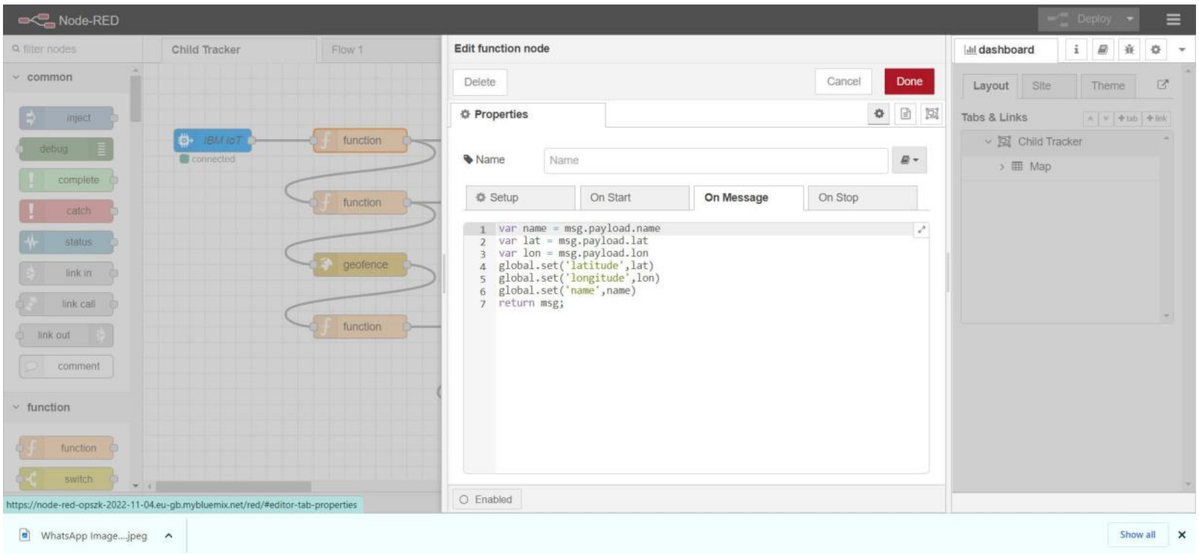
QoS: 0

Name: IBM IoT

Service: registered

Use the Input Type property to configure this node to receive Events sent by IoT Devices, Commands sent to IoT Devices, Status Messages referring to IoT Devices, or Status Messages referring to

○ Enabled



Edit function node

Delete | Cancel | Done

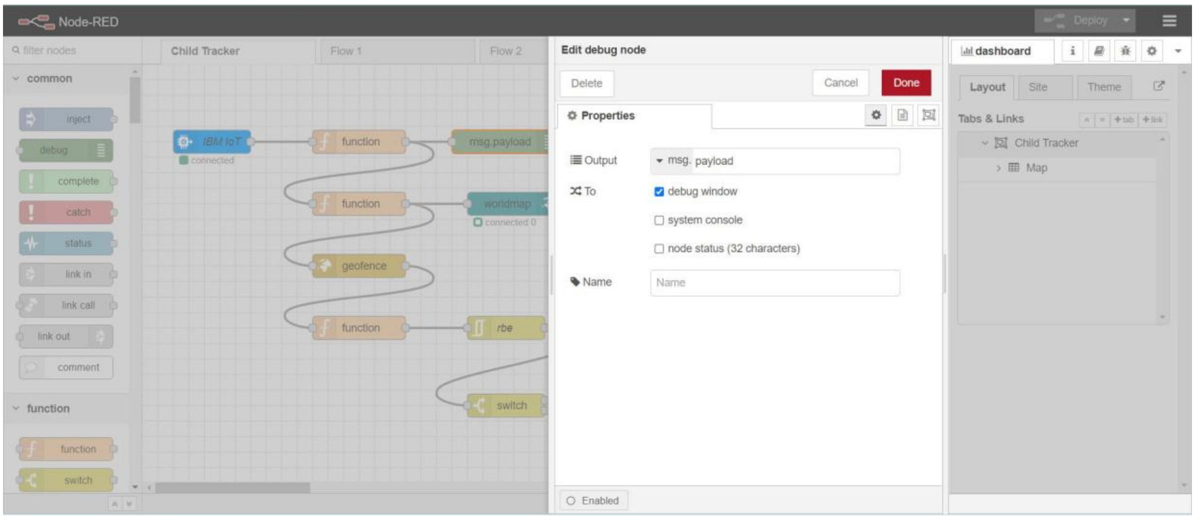
Properties

Name: Name

Setup | On Start | On Message | On Stop

```
var name = msg.payload.name
var lat = msg.payload.lat
var lon = msg.payload.lon
global.set('latitude',lat)
global.set('longitude',lon)
global.set('name',name)
return msg;
```

○ Enabled



Edit debug node

Delete | Cancel | Done

Properties

Output: msg. payload

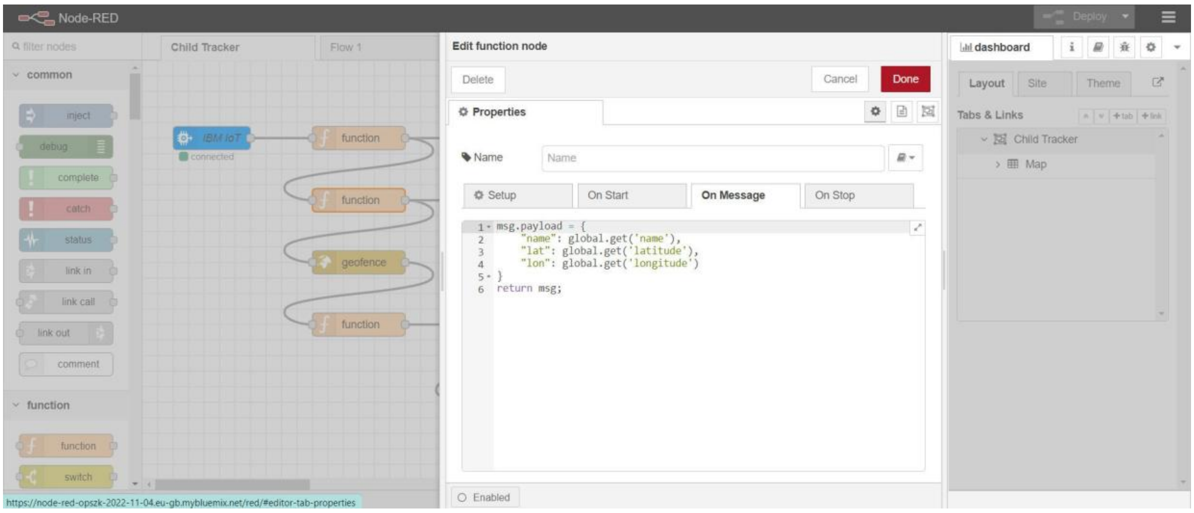
To: ☑ debug window

□ system console

□ node status (32 characters)

Name: Name

○ Enabled



Edit function node

Delete | Cancel | Done

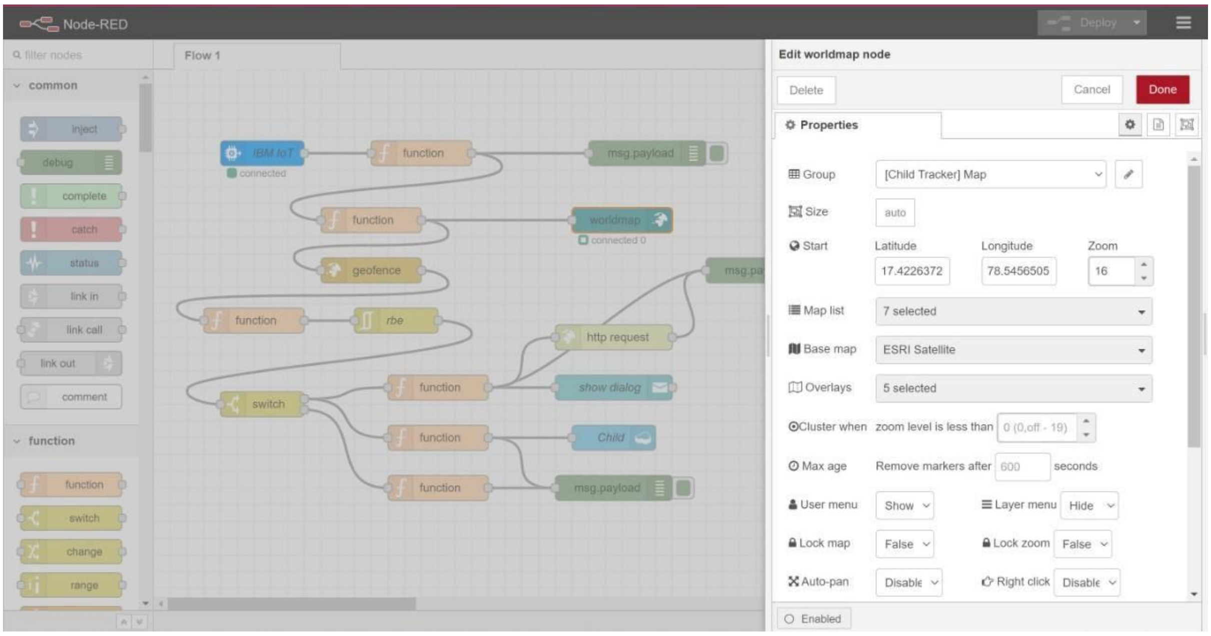
Properties

Name: Name

Setup | On Start | On Message | On Stop

```
msg.payload = {
    "name": global.get('name'),
    "lat": global.get('latitude'),
    "lon": global.get('longitude')
}
return msg;
```

○ Enabled



Edit worldmap node

Group: [Child Tracker] Map

Size: auto

Start — Latitude: 17.4226372 — Longitude: 78.5456505 — Zoom: 16

Map list: 7 selected

Base map: ESRI Satellite

Overlays: 5 selected

Cluster when zoom level is less than 0 (0,off - 19)

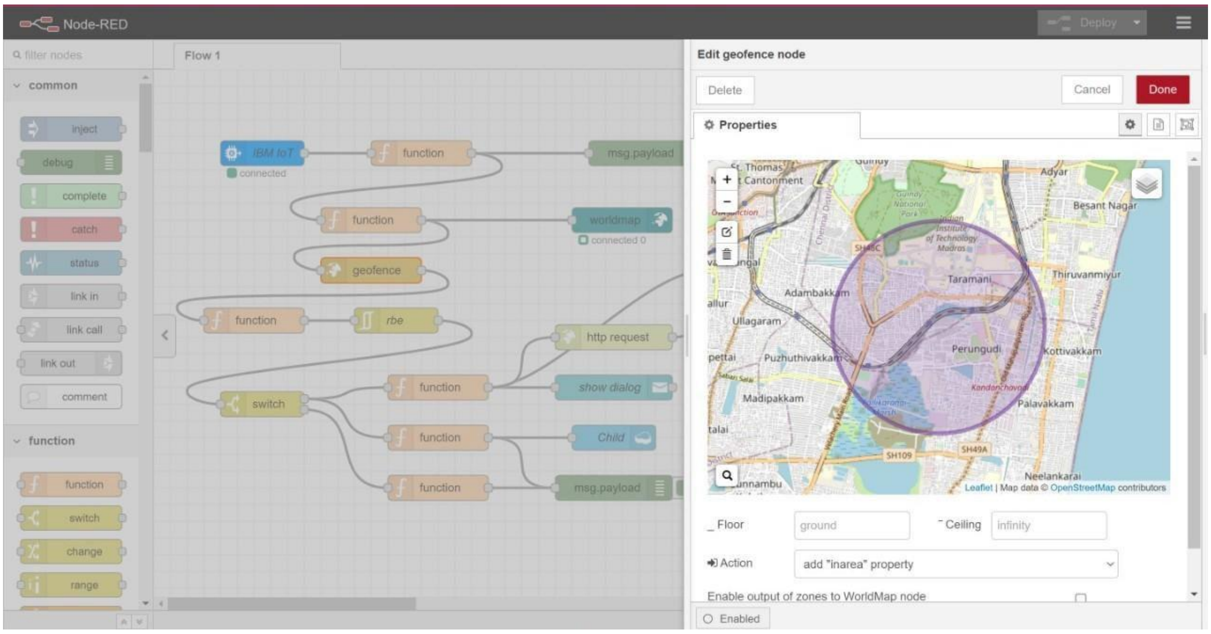
Max age: Remove markers after 600 seconds

User menu: Show — Layer menu: Hide

Lock map: False — Lock zoom: False

Auto-pan: Disable — Right click: Disable

○ Enabled



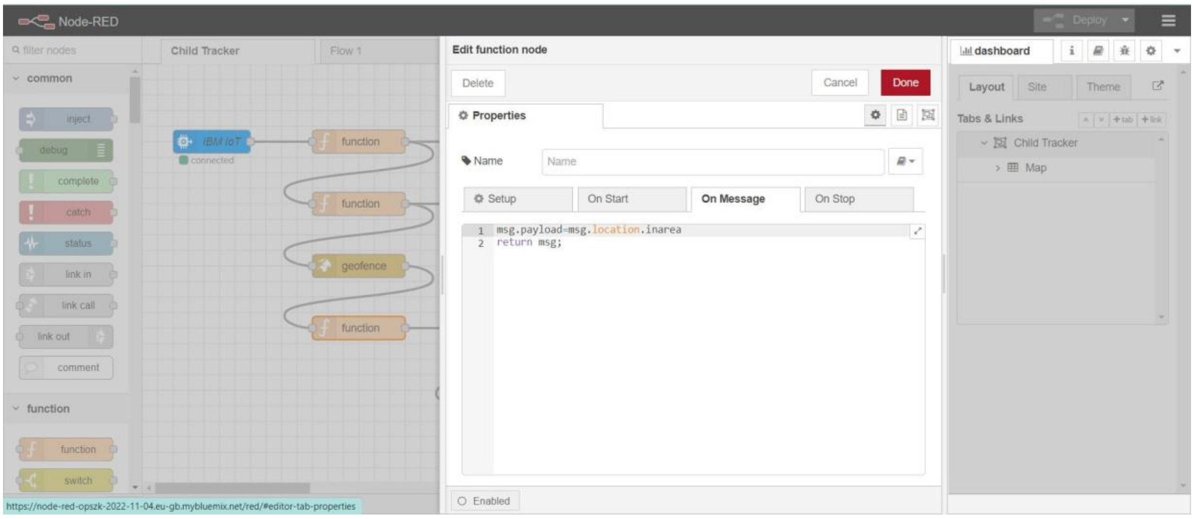
Edit geofence node

Floor: ground — Ceiling: infinity

Action: add "inarea" property

Enable output of zones to WorldMap node □

○ Enabled



Edit function node

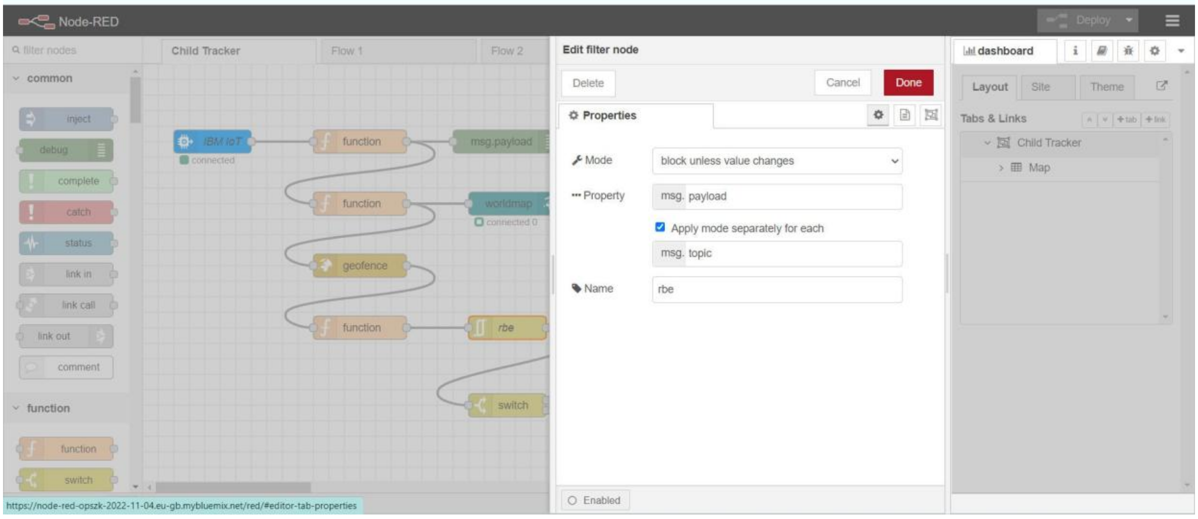
Name: Name

On Message

```
msg.payload=msg.location.inarea
return msg;
```

○ Enabled

https://node-red-opszk-2022-11-04.eu-gb.mybluemix.net/red/#editor-tab-properties



Edit filter node

Mode: block unless value changes

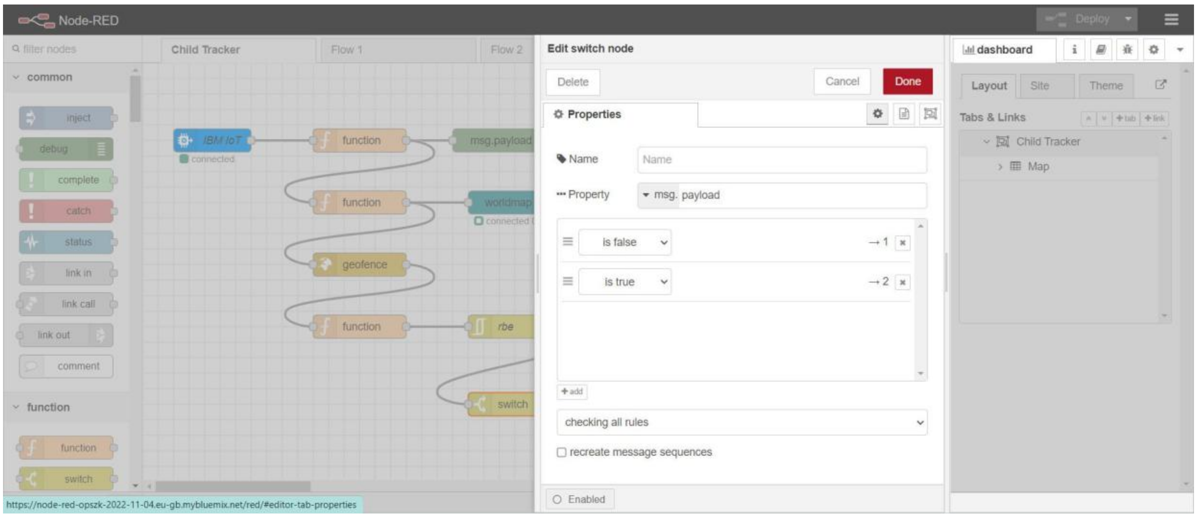
Property: msg. payload

☑ Apply mode separately for each

msg. topic

Name: rbe

○ Enabled



Edit switch node

Name: Name

Property: msg. payload

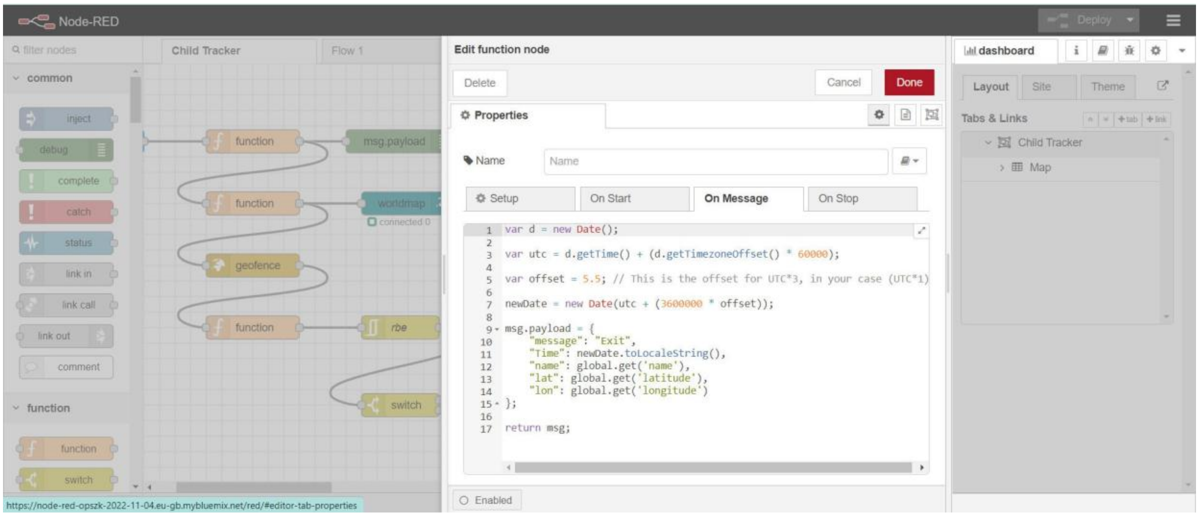
is false → 1

is true → 2

checking all rules

□ recreate message sequences

○ Enabled



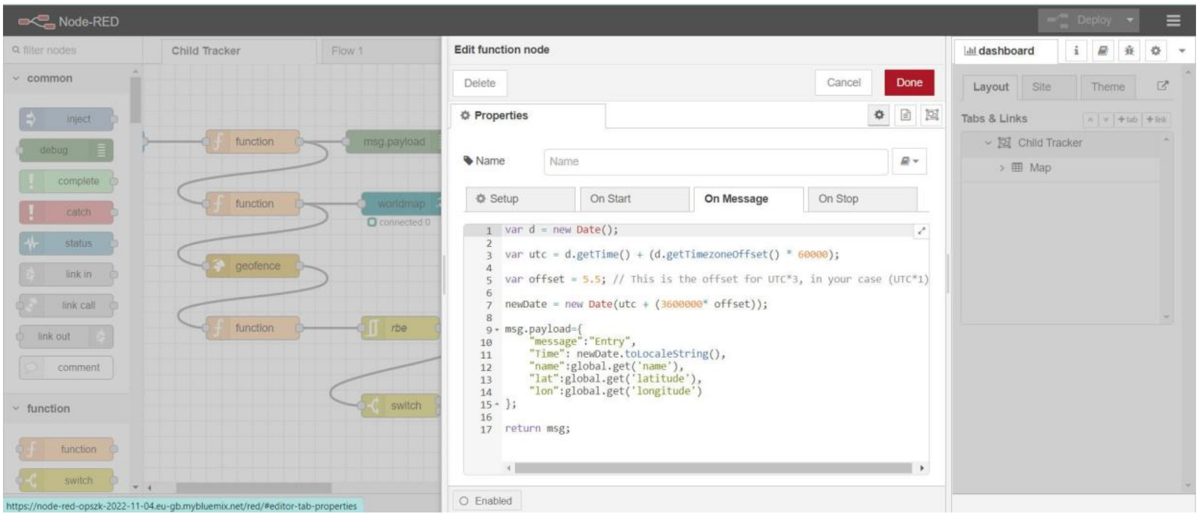
Edit function node

Name: Name

On Message

```
var d = new Date();

var utc = d.getTime() + (d.getTimezoneOffset() * 60000);

var offset = 5.5; // This is the offset for UTC*3, in your case (UTC*1)

newDate = new Date(utc + (3600000 * offset));

msg.payload = {
    "message": "Exit",
    "Time": newDate.toLocaleString(),
    "name": global.get('name'),
    "lat": global.get('latitude'),
    "lon": global.get('longitude')
};

return msg;
```

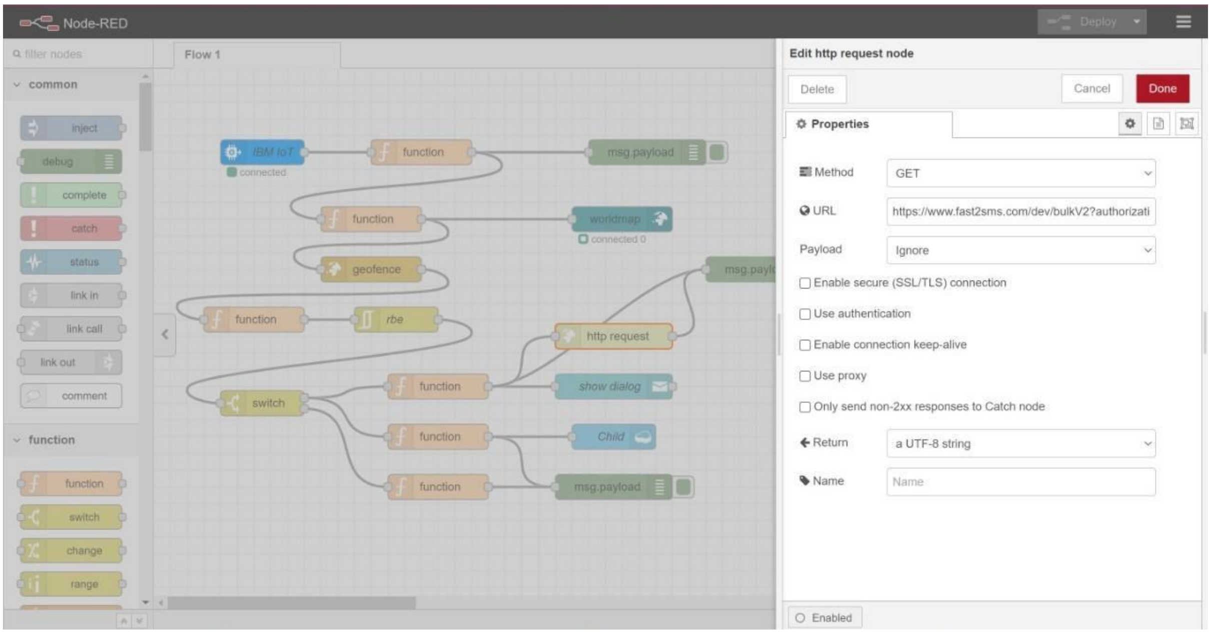
○ Enabled



Edit function node

Name: Name

```
var d = new Date();

var utc = d.getTime() + (d.getTimezoneOffset() * 60000);

var offset = 5.5; // This is the offset for UTC*3, in your case (UTC*1)

newDate = new Date(utc + (3600000* offset));

msg.payload={
    "message":"Entry",
    "Time": newDate.toLocaleString(),
    "name":global.get('name'),
    "lat":global.get('latitude'),
    "lon":global.get('longitude')
};

return msg;
```



Edit http request node

Method: GET

URL: https://www.fast2sms.com/dev/bulkV2?authorizati

Payload: Ignore

□ Enable secure (SSL/TLS) connection

□ Use authentication

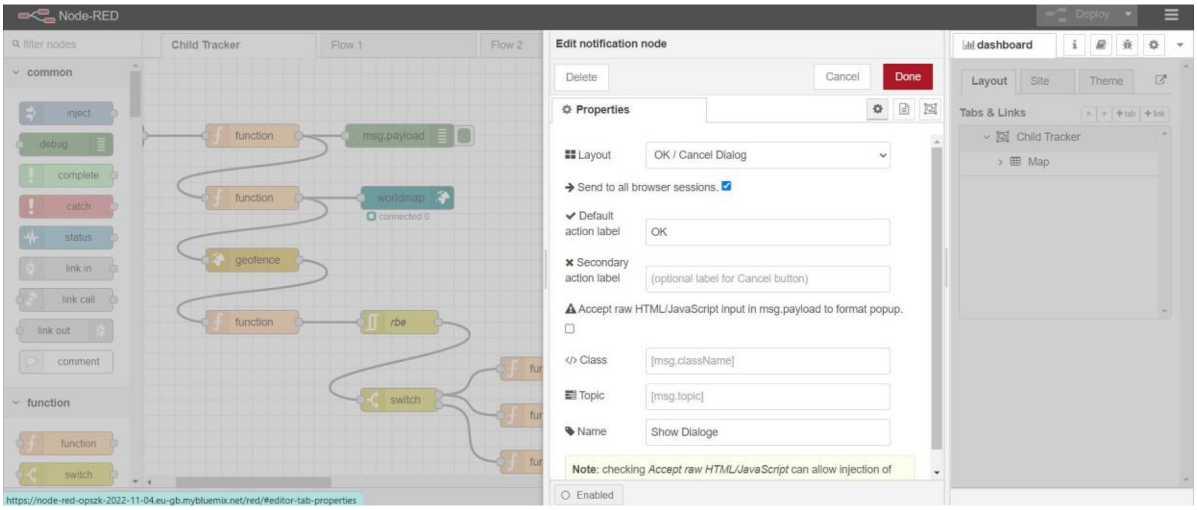
□ Enable connection keep-alive

□ Use proxy

□ Only send non-2xx responses to Catch node

Return: a UTF-8 string

Name: Name



Edit notification node

Layout: OK / Cancel Dialog

Send to all browser sessions. ☑

Default action label: OK

Secondary action label: (optional label for Cancel button)

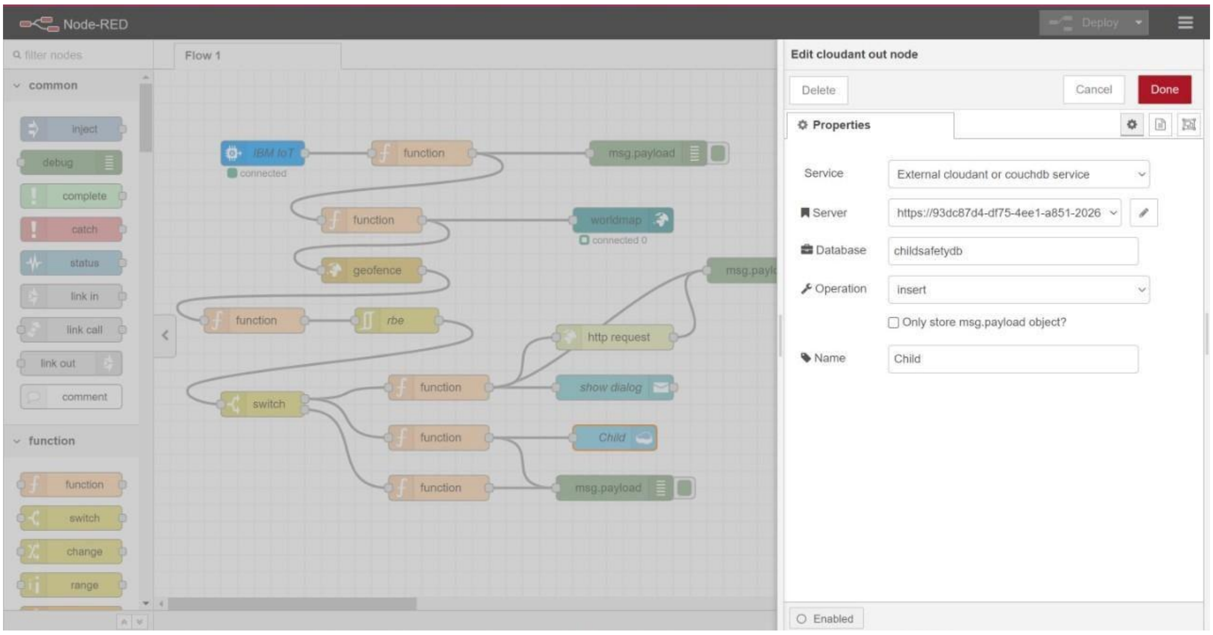
Accept raw HTML/JavaScript input in msg.payload to format popup. □

Class: [msg.className]

Topic: [msg.topic]

Name: Show Dialoge

Note: checking *Accept raw HTML/JavaScript* can allow injection of



Edit cloudant out node

Service: External cloudant or couchdb service

Server: https://93dc87d4-df75-4ee1-a851-2026

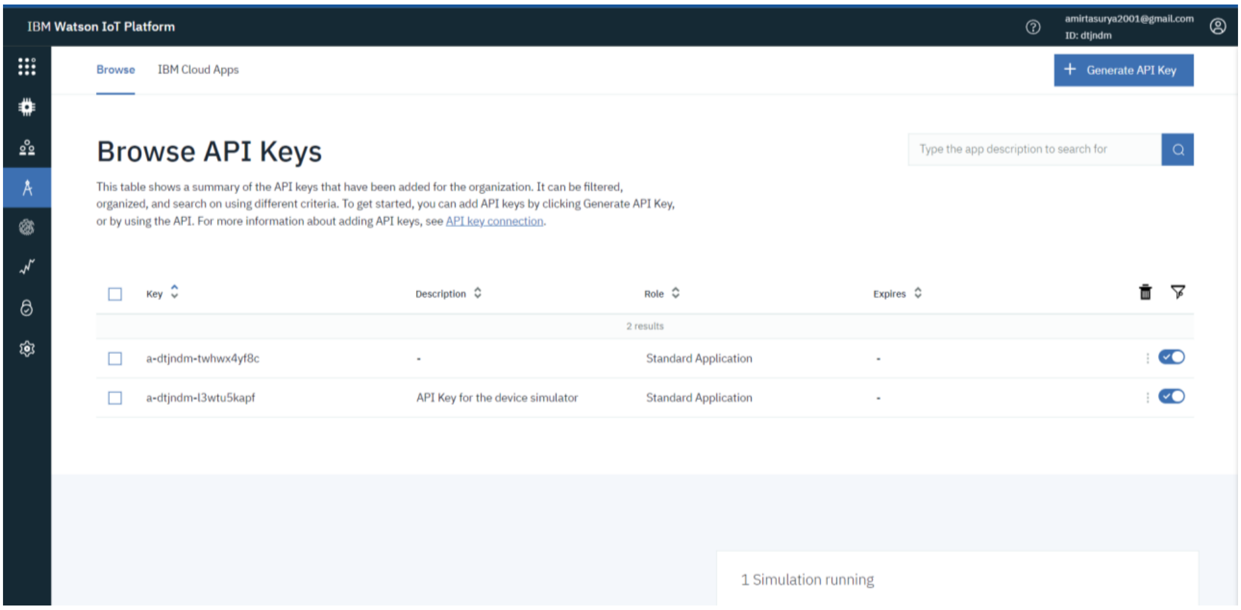
Database: childsafetydb

Operation: insert

☐ Only store msg.payload object?

Name: Child

Connecting with IBM Cloud: Using IBM IOT node through the API key

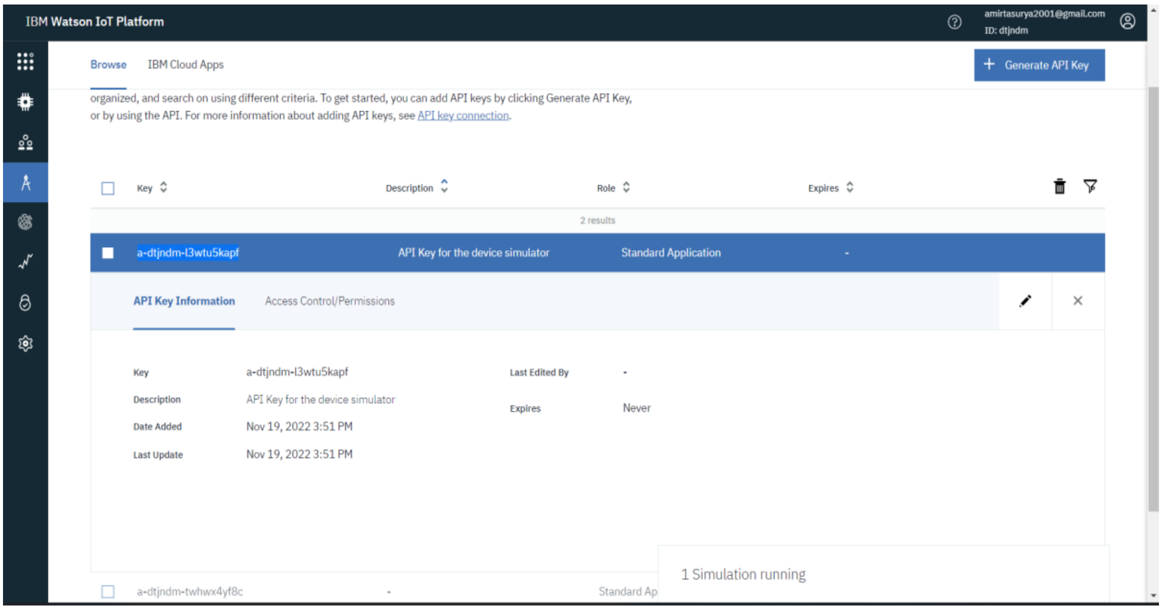


Browse API Keys

This table shows a summary of the API keys that have been added for the organization. It can be filtered, organized, and search on using different criteria. To get started, you can add API keys by clicking Generate API Key, or by using the API. For more information about adding API keys, see API key connection.

☐	Key	Description	Role	Expires
		2 results		
☐	a-dtjndm-twhwx4yf8c	-	Standard Application	-
☐	a-dtjndm-l3wtu5kapf	API Key for the device simulator	Standard Application	-

1 Simulation running



organized, and search on using different criteria. To get started, you can add API keys by clicking Generate API Key, or by using the API. For more information about adding API keys, see API key connection.

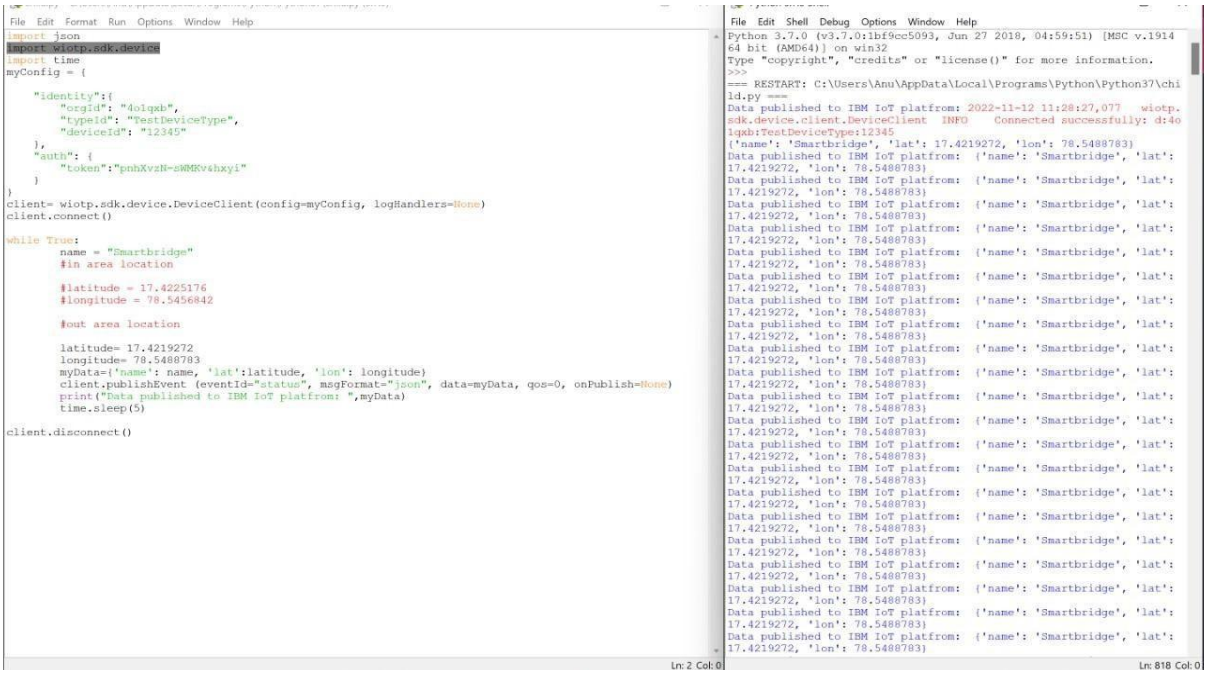
☐	Key	Description	Role	Expires
		2 results		
■	a-dtjndm-l3wtu5kapf	API Key for the device simulator	Standard Application	-
☐	a-dtjndm-twhwx4yf8c	-	Standard Ap	

API Key Information   Access Control/Permissions

Key	a-dtjndm-l3wtu5kapf	Last Edited By	-
Description	API Key for the device simulator	Expires	Never
Date Added	Nov 19, 2022 3:51 PM		
Last Update	Nov 19, 2022 3:51 PM		

1 Simulation running

Transferring values from Python Code:

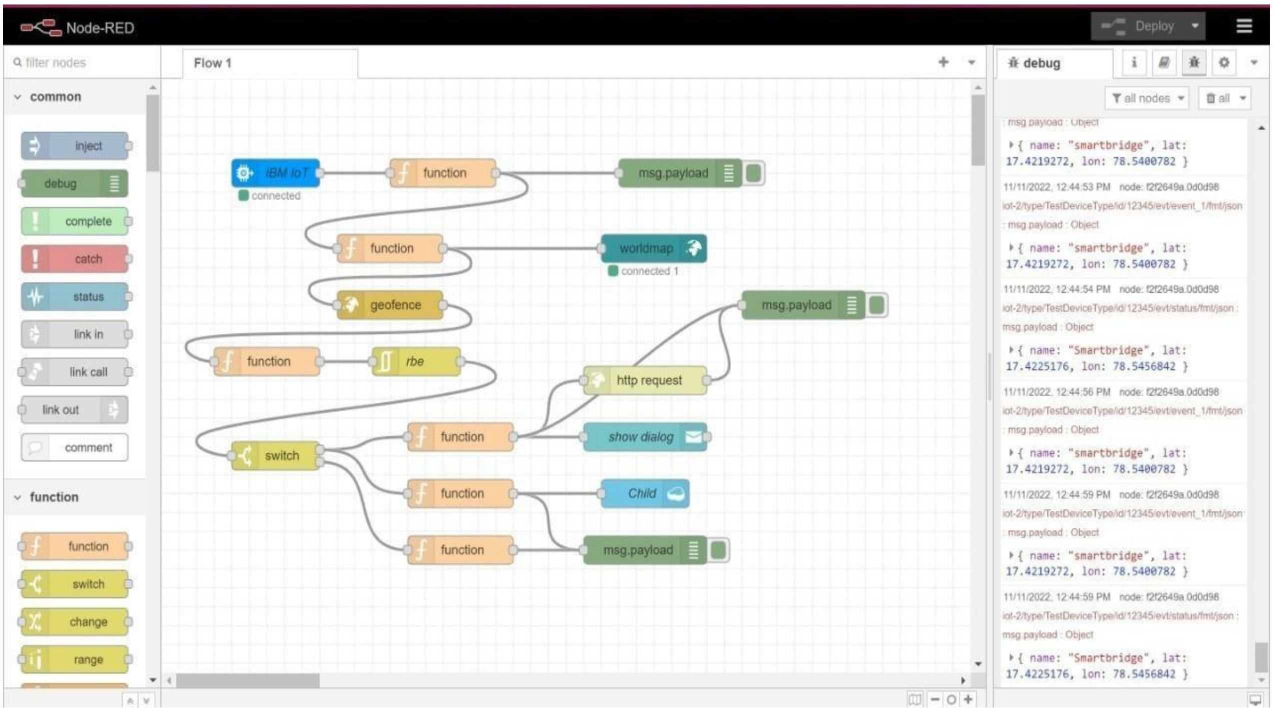


File Edit Format Run Options Window Help
File Edit Shell Debug Options Window Help
Ln: 2 Col: 0
Ln: 818 Col: 0

```
import json
import wiotp.sdk.device
import time
myConfig = {

    "identity":{
        "orgId": "4o1qxb",
        "typeId": "TestDeviceType",
        "deviceId": "12345"
    },
    "auth": {
        "token":"pnhXvzN-sWMKv&hxyi"
    }
}
client= wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()

while True:
        name = "Smartbridge"
        #in area location

        #latitude = 17.4225176
        #longitude = 78.5456842

        #out area location

        latitude= 17.4219272
        longitude= 78.5488783
        myData={'name': name, 'lat':latitude, 'lon': longitude}
        client.publishEvent (eventId="status", msgFormat="json", data=myData, qos=0, onPublish=None)
        print("Data published to IBM IoT platfrom: ",myData)
        time.sleep(5)

client.disconnect()
```

```
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) [MSC v.1914
64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
=== RESTART: C:\Users\Anu\AppData\Local\Programs\Python\Python37\chi
ld.py ===
Data published to IBM IoT platfrom: 2022-11-12 11:28:27,077   wiotp.
sdk.device.client.DeviceClient  INFO    Connected successfully: d:4o
1qxb:TestDeviceType:12345
{'name': 'Smartbridge', 'lat': 17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platfrom:  {'name': 'Smartbridge', 'lat':
17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platfrom:  {'name': 'Smartbridge', 'lat':
17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platfrom:  {'name': 'Smartbridge', 'lat':
17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platfrom:  {'name': 'Smartbridge', 'lat':
17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platfrom:  {'name': 'Smartbridge', 'lat':
17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platfrom:  {'name': 'Smartbridge', 'lat':
17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platfrom:  {'name': 'Smartbridge', 'lat':
17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platfrom:  {'name': 'Smartbridge', 'lat':
17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platfrom:  {'name': 'Smartbridge', 'lat':
17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platfrom:  {'name': 'Smartbridge', 'lat':
17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platfrom:  {'name': 'Smartbridge', 'lat':
17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platfrom:  {'name': 'Smartbridge', 'lat':
17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platfrom:  {'name': 'Smartbridge', 'lat':
17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platfrom:  {'name': 'Smartbridge', 'lat':
17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platfrom:  {'name': 'Smartbridge', 'lat':
17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platfrom:  {'name': 'Smartbridge', 'lat':
17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platfrom:  {'name': 'Smartbridge', 'lat':
17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platfrom:  {'name': 'Smartbridge', 'lat':
17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platfrom:  {'name': 'Smartbridge', 'lat':
17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platfrom:  {'name': 'Smartbridge', 'lat':
17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platfrom:  {'name': 'Smartbridge', 'lat':
17.4219272, 'lon': 78.5488783}
```

Node-Red:



```
: msg.payload : Object
▸{ name: "smartbridge", lat:
17.4219272, lon: 78.5400782 }
11/11/2022, 12:44:53 PM   node: f2f2649a.0d0d98
iot-2/type/TestDeviceType/id/12345/evt/event_1/fmt/json
: msg.payload : Object
▸{ name: "smartbridge", lat:
17.4219272, lon: 78.5400782 }
11/11/2022, 12:44:54 PM   node: f2f2649a.0d0d98
iot-2/type/TestDeviceType/id/12345/evt/status/fmt/json :
msg.payload : Object
▸{ name: "Smartbridge", lat:
17.4225176, lon: 78.5456842 }
11/11/2022, 12:44:56 PM   node: f2f2649a.0d0d98
iot-2/type/TestDeviceType/id/12345/evt/event_1/fmt/json
: msg.payload : Object
▸{ name: "smartbridge", lat:
17.4219272, lon: 78.5400782 }
11/11/2022, 12:44:59 PM   node: f2f2649a.0d0d98
iot-2/type/TestDeviceType/id/12345/evt/event_1/fmt/json
: msg.payload : Object
▸{ name: "smartbridge", lat:
17.4219272, lon: 78.5400782 }
11/11/2022, 12:44:59 PM   node: f2f2649a.0d0d98
iot-2/type/TestDeviceType/id/12345/evt/status/fmt/json :
msg.payload : Object
▸{ name: "Smartbridge", lat:
17.4225176, lon: 78.5456842 }
```