

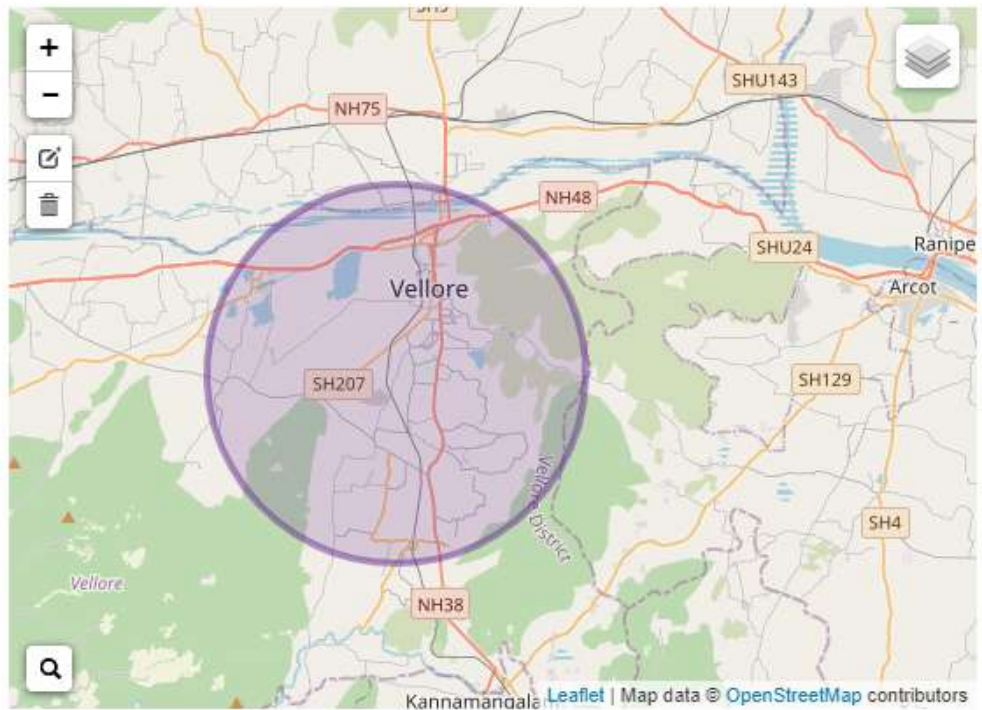
### Edit geofence node

Delete

Cancel

Done

#### Properties



\_ Floor

ground

\_ Ceiling

infinity

Action

add "inarea" property

Enable output of zones to WorldMap node



Name

Geofence name

help



> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

> n

### geofence

A simple g

It supports

all messag

the region

selected m

This node

the followi

- `msg.p`

- `msg.l`

- `msg.la`

Alternative

to the msg

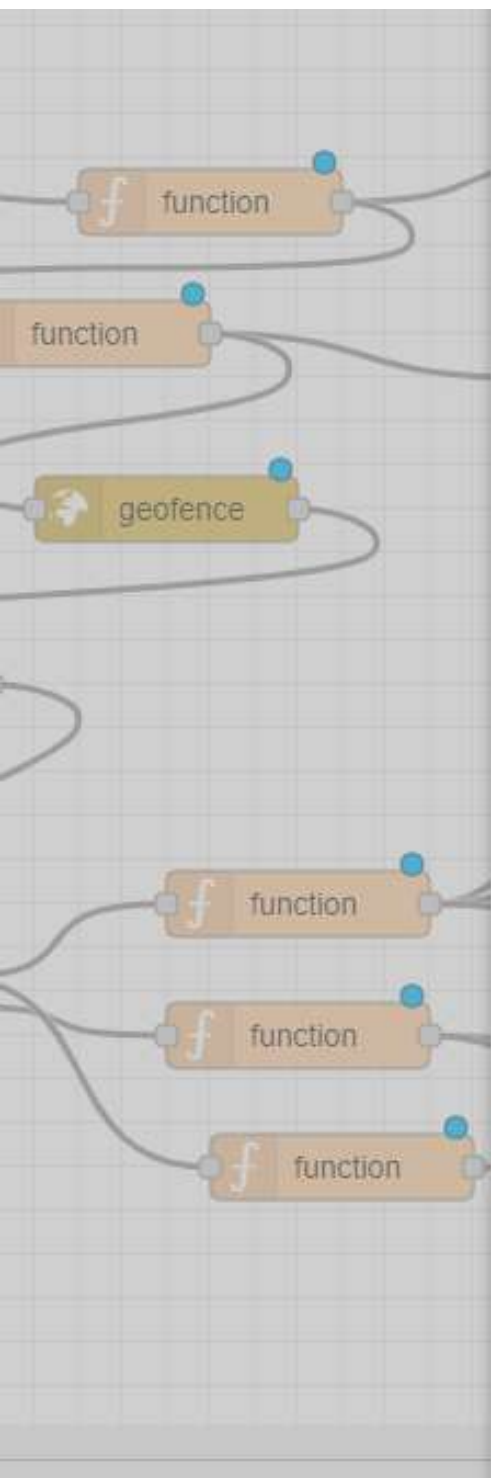
In this mod

**msg.locat**

a list of na

of and **ms**

object of n



Delete Cancel Update

### Properties

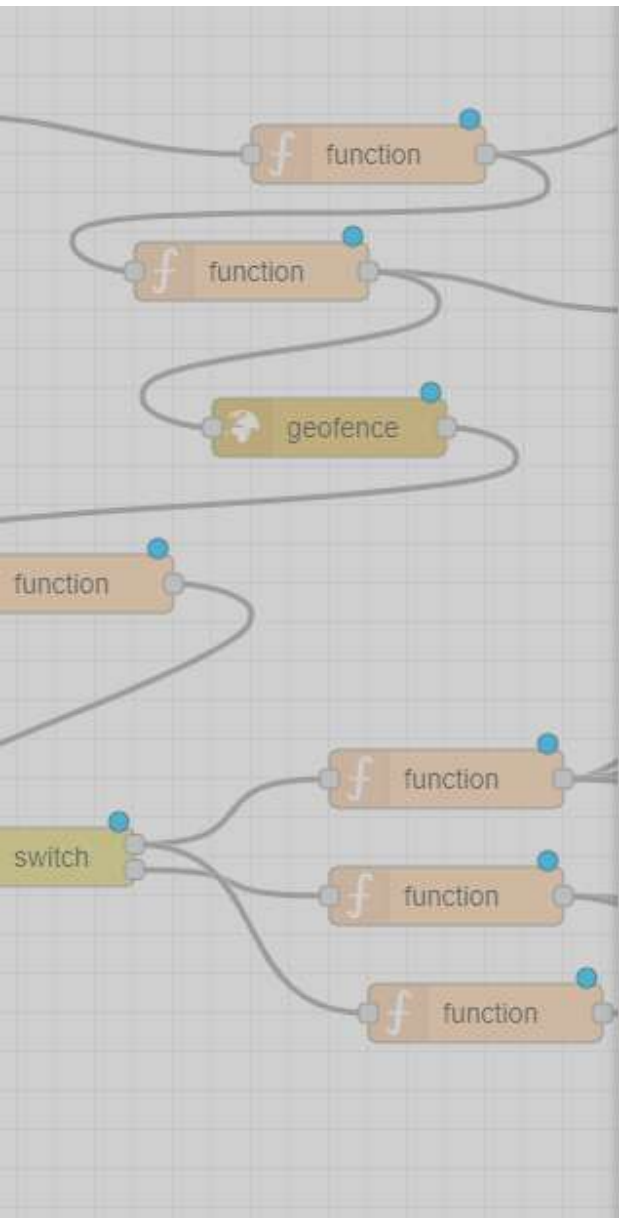
**Host**

**Username**

**Password**

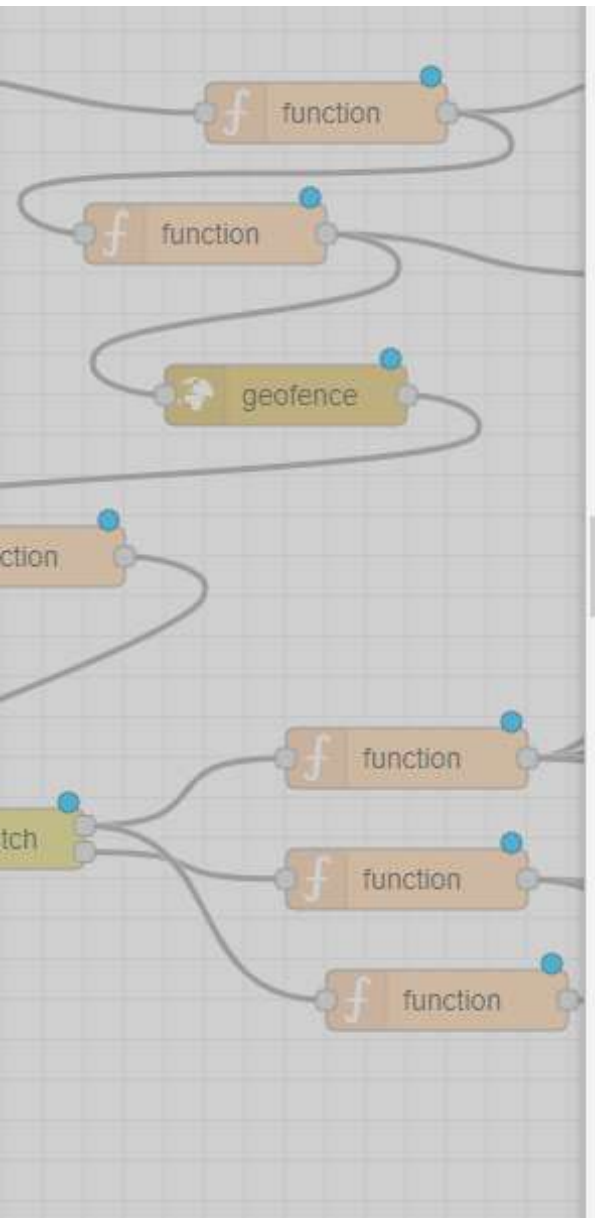
**Name**

Enabled 1 node uses this config On all flows



## ⚙ Properties

Service	External cloudant or couchdb service	⌵
🏠 Server	https://apikey-v2-ggd2ua4kz511yzfjs4h	⌵ ✎
📁 Database	child	
🔑 Operation	insert	⌵
	<input type="checkbox"/> Only store msg payload object?	
🏷 Name	Name	



## ⚙ Properties



☰ 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

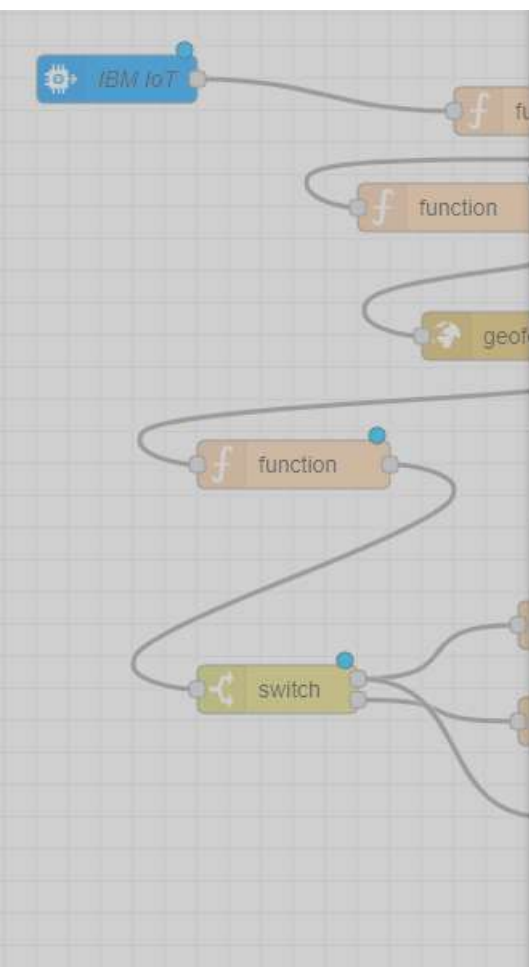
← Return

a UTF-8 string



🔑 Name

Name



## Properties

Name

Name

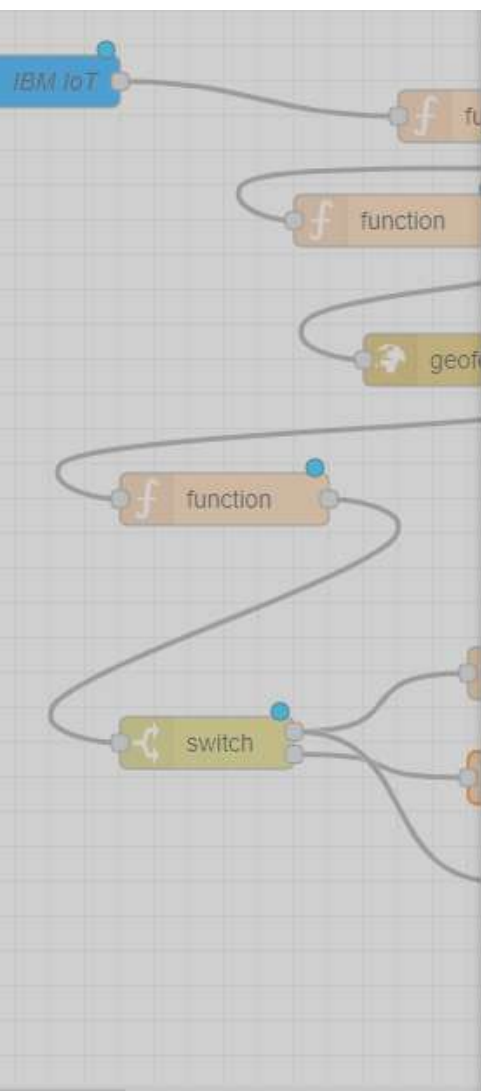
Setup

On Start

On Message

On Stop

```
1 var d = new Date();
2
3 var utc = d.getTime() + (d.getTimezoneOffset()*60000);
4
5 var offset =5.5;
6
7 newDate=new Date(utc + (3600000*offset));
8
9 msg.payload ={
10   "message":"Entry",
11   "Time":newDate.toLocaleString(),
12   "name":global.get("name"),
13   "lat":global.get("latitude"),
14   "lon":global.get("longitude")
15 };
16 };
17 return msg;
```



## Properties

Name

Name

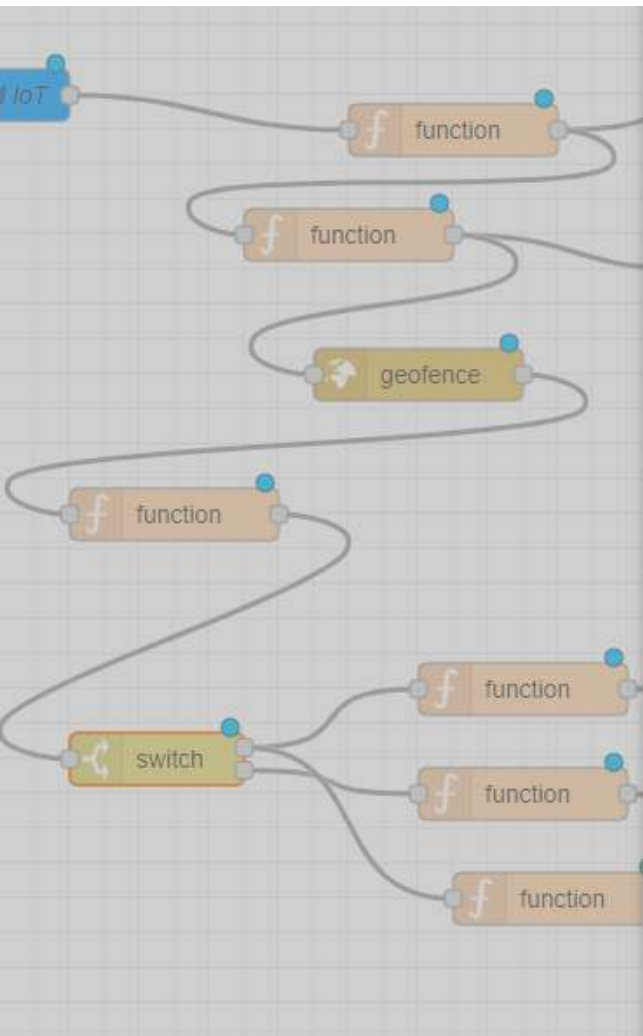
Setup

On Start

On Message

On Stop

```
1 var d = new Date();
2
3 var utc = d.getTime() + (d.getTimezoneOffset()*60000);
4
5 var offset =5.5;
6
7 newDate=new Date(utc + (3600000*offset));
8
9 msg.payload ={
10   "message":"Exit",
11   "Time":newDate.toLocaleString(),
12   "name":global.get('name'),
13   "lat":global.get('latitude'),
14   "lon":global.get('longitude')
15 };
16 ^};
17 return msg;
```



## Properties

Name

Name

Property

msg. payload



is false



→ 1



is true



→ 2

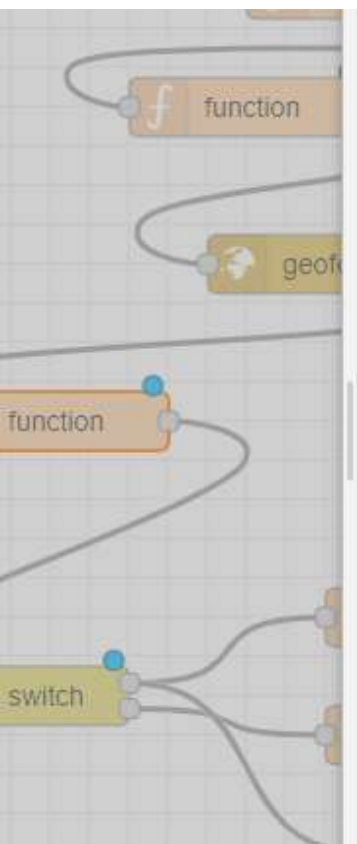


+ add

checking all rules



☐ recreate message sequences

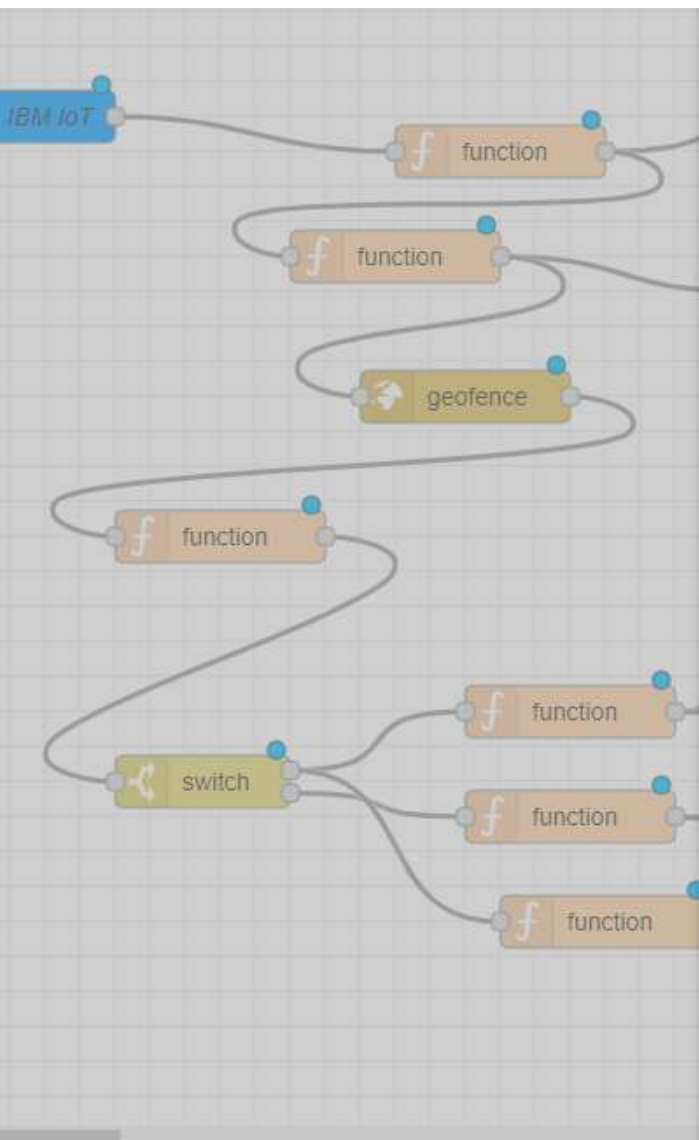


▼ Name      Name

⚙ Setup	On Start	<b>On Message</b>	On Stop
---------	----------	-------------------	---------

```
1 msg.payload=msg.location.inarea
2 return msg;
3
```





### Properties

**Group** [safety] child safety

**Size** auto

**Start**

Latitude	Longitude	Zoom
12.871807	79.126633	16

**Map list** 7 selected

**Base map** ESRI Satellite

**Overlays** 5 selected

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

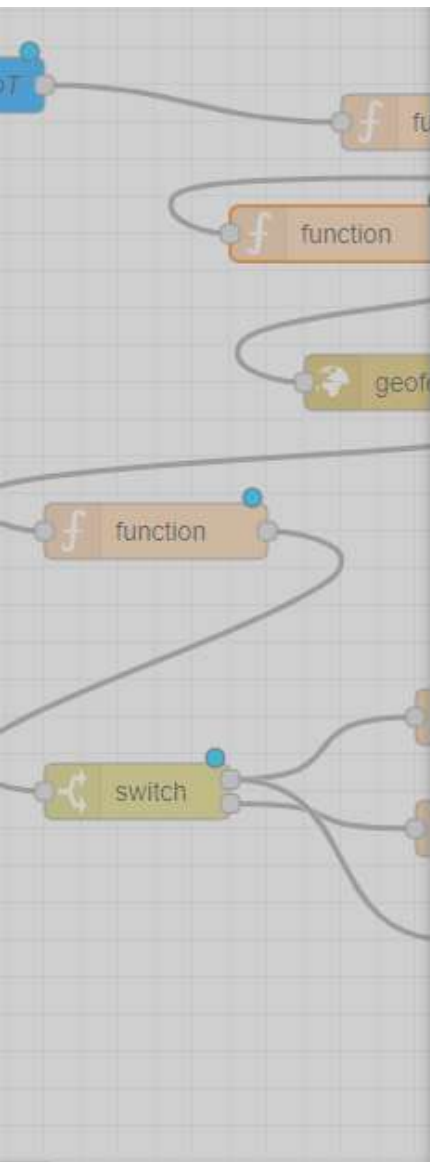
**Max age** Remove markers after 600 seconds

**User menu** Show **Layer menu** Show

**Lock map** False **Lock zoom** False

**Auto-pan** Disable **Right click** Disable

**Co-ordinates** Not shown **Graticule** Not shown



## Properties

Name

Name

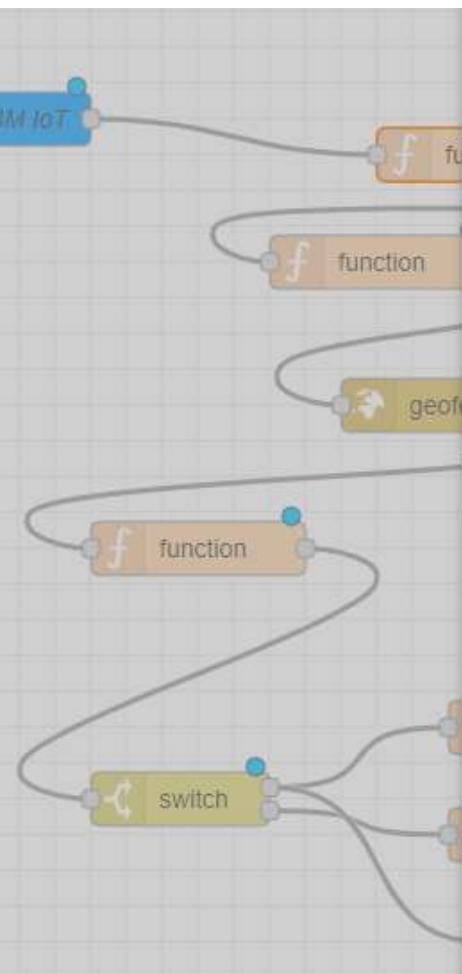
Setup

On Start

On Message

On Stop

```
1 msg.payload={
2   'name':global.get('name'),
3   'lat':global.get('latitude'),
4   'lon':global.get('longitude')
5 };
6 return msg;
```



## Properties

Name

Name

Setup

On Start

On Message

On Stop

```
1 var name=msg.payload.name
2 var lat=msg.payload.lat
3 var lon=msg.payload.lon
4 global.set("latitude",lat)
5 global.set("longitude",lon)
6 global.set("name",name)
7 return msg;
```