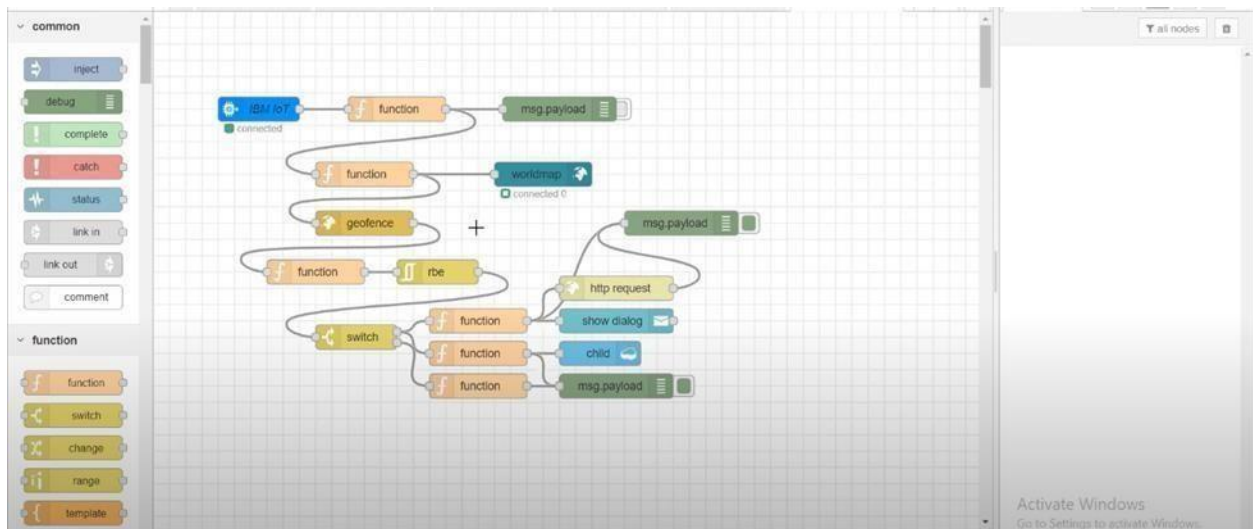


## PREREQUISITES

### NODE RED SERVICE

Step 1: Connect the blocks.



```
import json
import wiotp.sdk.device
import time

myConfig = {
  "identity": {
    "orgId": "h35fmy",
    "typeId": "NodeMCU",
    "deviceId": "12345"
  },
  "auth": {
    "token": "12345678"
  }
}

client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()

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

  latitude= 17.4225176
  longitude= 78.5458842

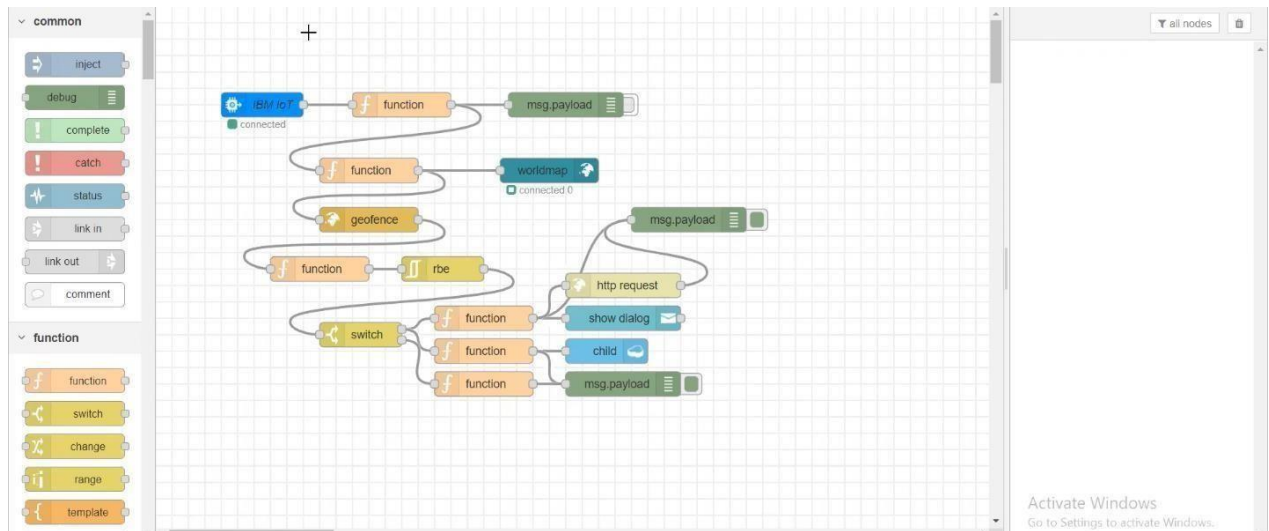
  #out area location

  #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 platform: ",myData)
  time.sleep(5)

client.disconnect()
```

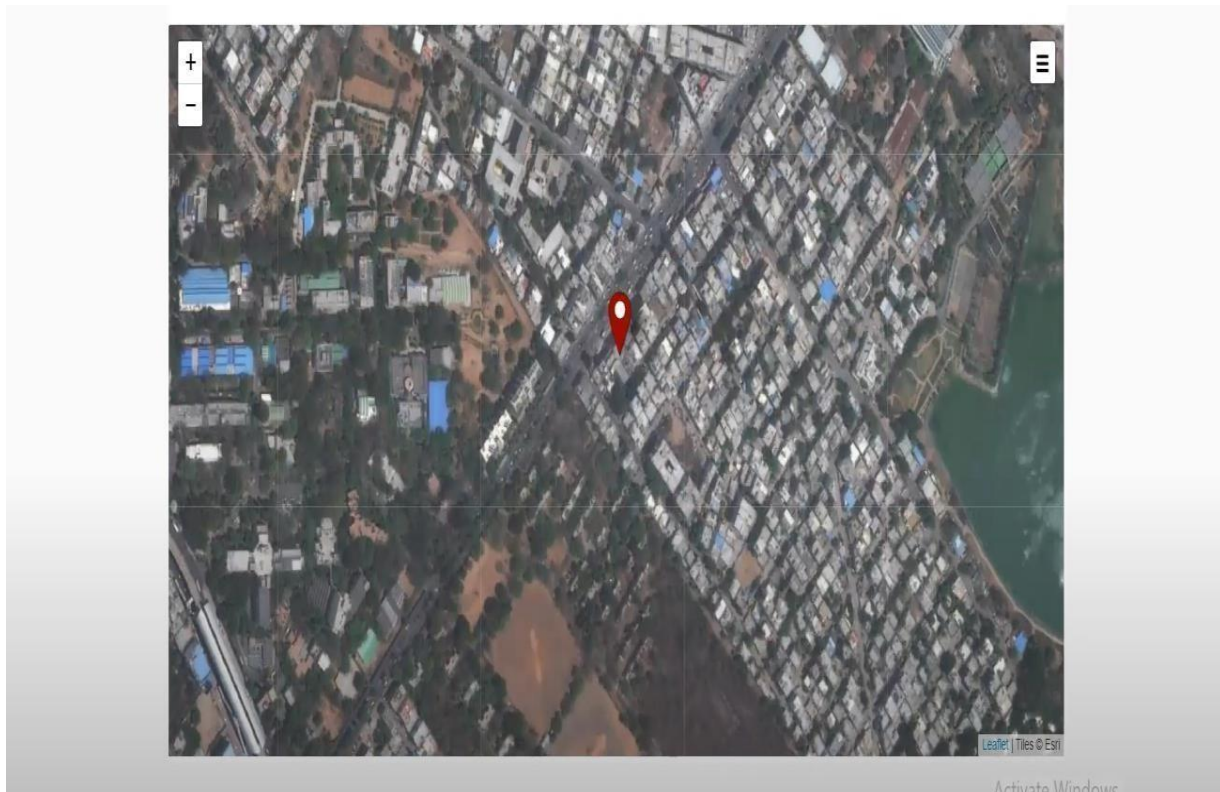
Step 2: Create python code.

Step 3: Click the geo-fence node.

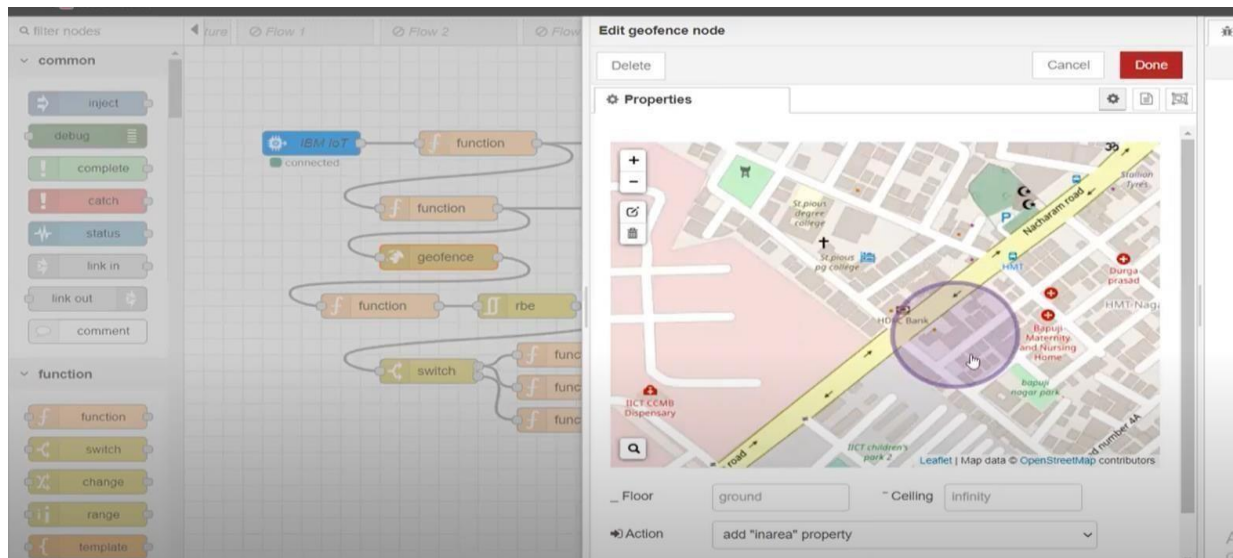


Step 4: Create the geo-fence area in the map.

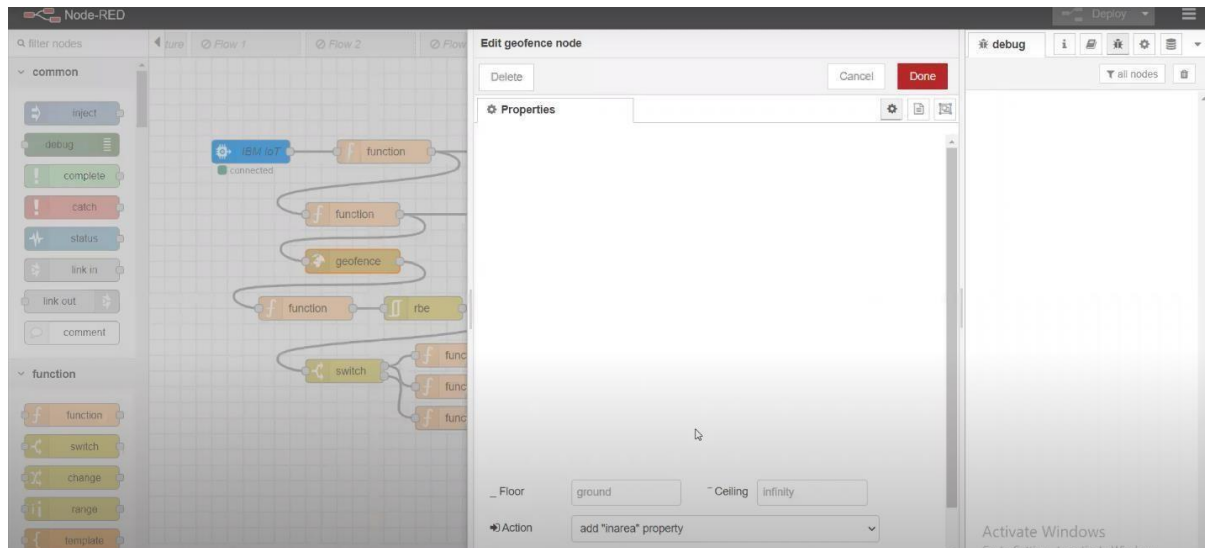
Step



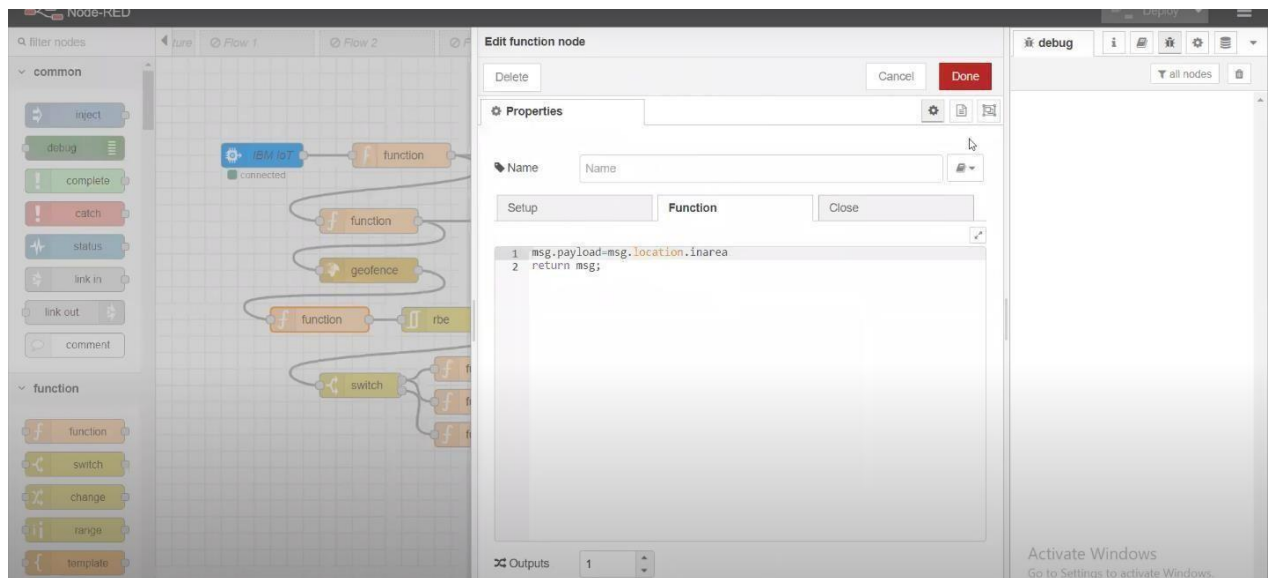
5: Create geo-fence in a particular area.



Step 6: Select the function block.

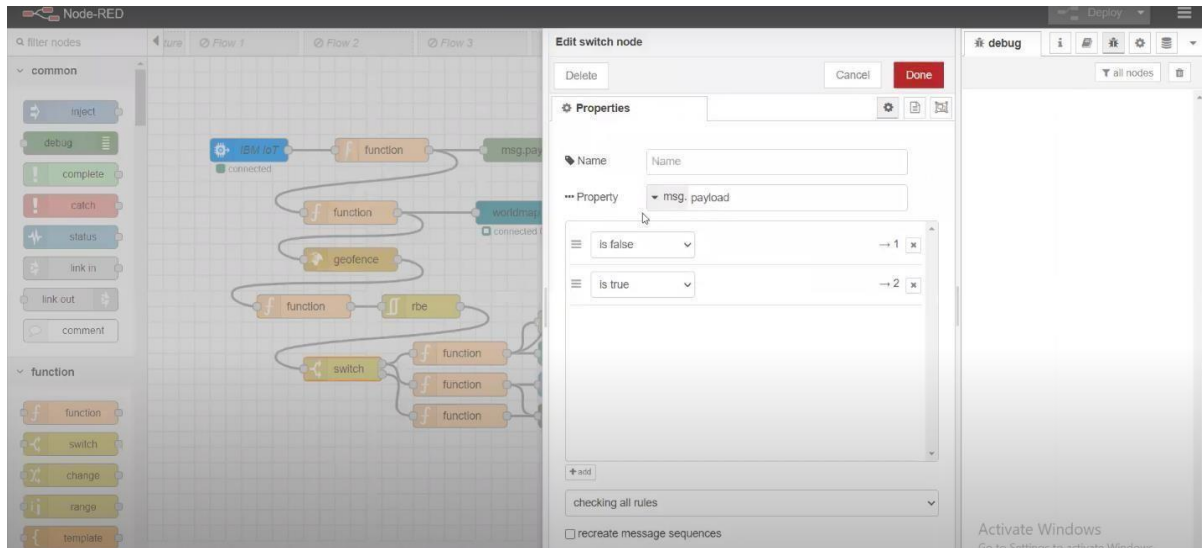


Step 7: Select the message payload.

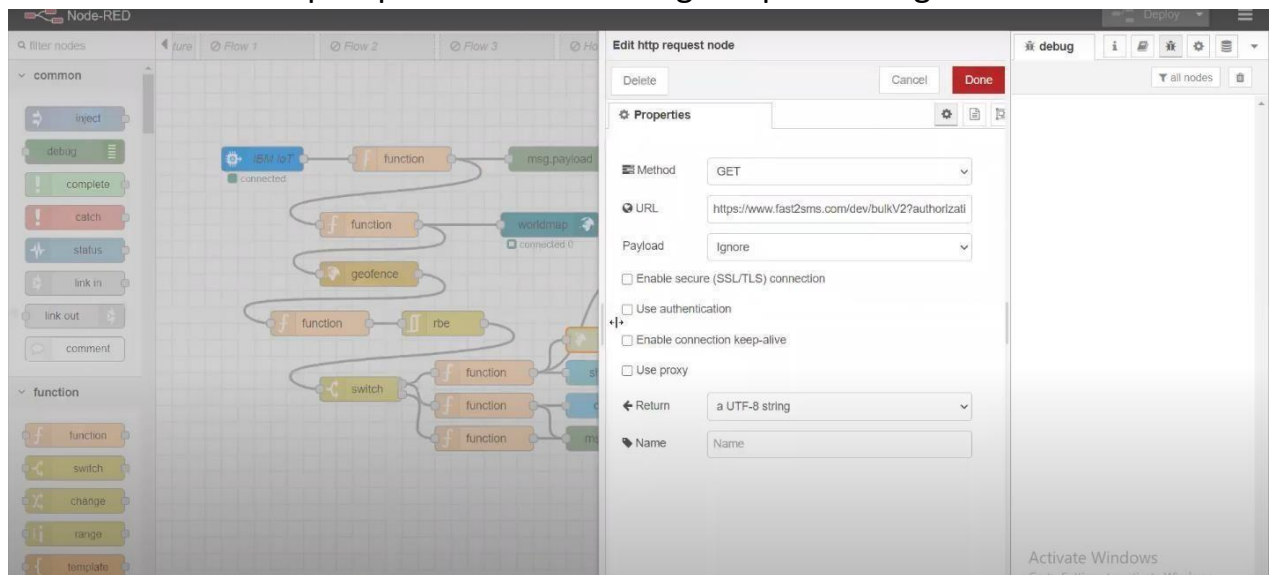


Step 8: To identify the person in area.

Step



9: Select the http request to send message to parent or guardian.

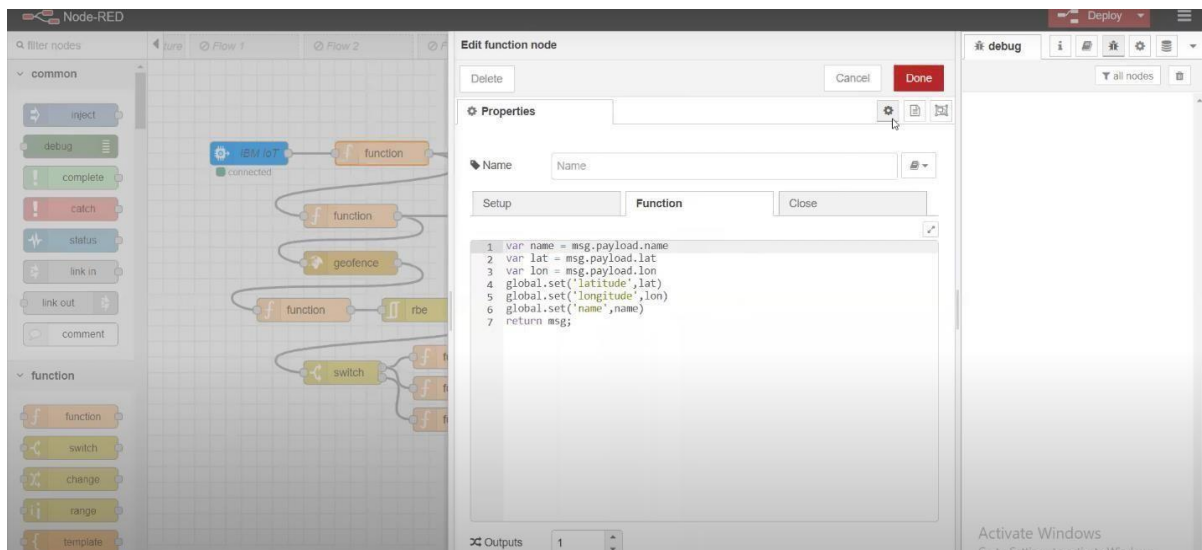


Step 10: For sending the message with time.



Step

10: Click show dialog for notifying the popup alert.





## Step

The screenshot displays the Node-RED web interface. On the left, the 'common' and 'function' node palettes are visible. The central workspace shows a flow diagram with nodes including 'inject', 'debug', 'complete', 'catch', 'status', 'link in', 'link out', 'comment', 'function', 'switch', 'rbe', 'worldmap', and 'msg.payload'. The right-hand panel is titled 'Edit notification node' and contains the following settings:

- Delete** button
- Cancel** and **Done** buttons
- Properties** section with icons for settings, save, and help.
- Layout** dropdown menu set to 'OK / Cancel Dialog'.
- Send to all browser sessions** checkbox, currently unchecked.
- Default action label** text input field containing 'OK'.
- Secondary action label** text input field containing '(optional label for Cancel button)'.
- Accept raw HTML/JavaScript input in msg.payload to format popup** checkbox, currently unchecked.
- Topic** text input field containing '[msg.topic]'.
- Name** text input field containing 'Name'.
- Note:** checking **Accept raw HTML/JavaScript** can allow injection of code. Ensure the input comes from trusted sources.

At the bottom right of the interface, there is a message: 'Activate Windows. Go to Settings to activate Windows.'



Step

Step 11: Create another payload and to pass the data to geo-fence and world map.

Step

12: Click the world map to see the location.

