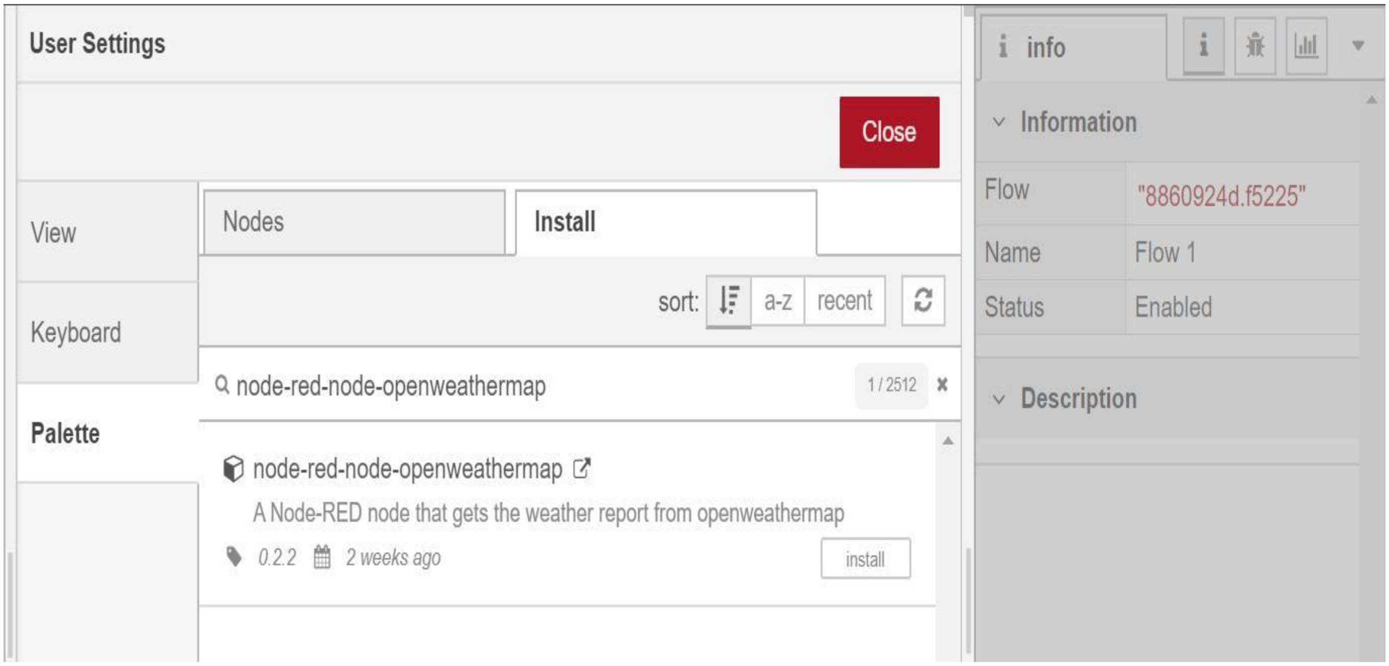
DEVELOP A WEB APPLICATION USING NODE -RED :

1. Double-click the tab with the flow name, and call it `Earthquake Details`.
2. Click the hamburger menu, and then click **Manage palette**. Look for **node-red-node-openweathermap** to install these additional nodes in your palette.



node-red-node-openweathermap 1 / 2512

node-red-node-openweathermap

A Node-RED node that gets the weather report from openweathermap

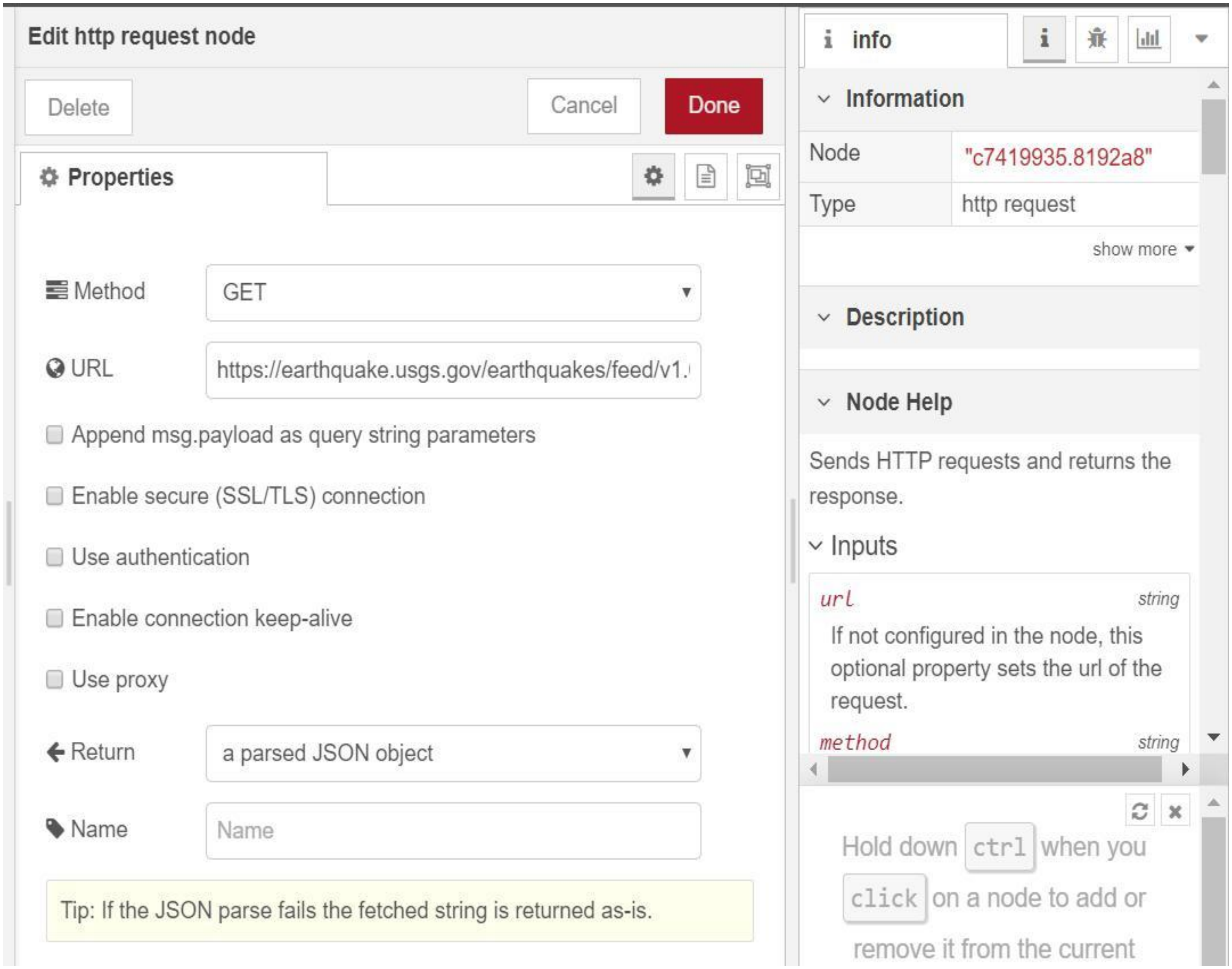
0.2.2 2 weeks ago

Flow	"8860924d.f5225"
Name	Flow 1
Status	Enabled

Add an **HTTP input** node to your flow.

Double-click the node to edit it. Set the method to `GET` and set the URL to `/earthquakeinfo-hr`.

1. Add an **HTTP response** node, and connect it to the previously added **HTTP input** node. All other nodes introduced in this sub-section is to be added between the **HTTP input** node and the **HTTP response** node.
2. Add an **HTTP request** node and set the *URL* to `https://earthquake.usgs.gov/earthquakes/feed/v1.0/summary/all_hour.geojson`, the *Method* to **GET** and the *Return* to **a parsed JSON object**. This will allow extracting all earthquakes that occurred within the last hour. Name this node `Get Earthquake Info from USGS`.



Method: GET

URL: https://earthquake.usgs.gov/earthquakes/feed/v1.

☐ Append msg.payload as query string parameters

☐ Enable secure (SSL/TLS) connection

☐ Use authentication

☐ Enable connection keep-alive

☐ Use proxy

Return: a parsed JSON object

Name: Name

Tip: If the JSON parse fails the fetched string is returned as-is.

Node	"c7419935.8192a8"
Type	http request

Sends HTTP requests and returns the response.

url *string*

If not configured in the node, this optional property sets the url of the request.

method *string*

Hold down `ctrl` when you `click` on a node to add or remove it from the current

Add a **change** node. Double-click the node to modify it. Name this node `Set Earthquake Info`. In the **Rules** section, add rules to *Delete* `msg.topic`, `msg.headers`, `msg.statusCode`, `msg.responseUrl` and `msg.redirectList` and *Set* `msg.payload` `payload.features.`

```
{
   "type":properties.type,
   "magnitude": properties.mag,
   "location": properties.place,
   "longitude":geometry.coordinates[0],
   "latitude":geometry.coordinates[1],
   "depth":geometry.coordinates[2],
   "timestamp": $fromMillis(
       properties.time,
       '[H01]:[m01]:[s01] [z]',
       '+0400'
   ),
   "source": properties.net
}
```

the following JSONata expression.