

Node-RED

home

about

blog

documentation

forum

flows

github

Node-RED

Low-code programming for event-driven applications

Latest version: v3.0.2 (npm)

Node-RED is a programming tool for wiring together hardware devices, APIs and online services in new and interesting ways.

It provides a browser-based editor that makes it easy to wire together flows using the wide range of nodes in the palette that can be deployed to its runtime in a single-click.

Features

Get Started

Community



Node-RED はじめてのいっぽ for Windows

CC BY @taisukef



<https://nodered.org/>

Node-RED ハンズオン

Node-RED

home

about

blog

documentation

forum

flows

github

Node-RED

Low-code programming for event-driven applications

Latest version: v3.0.2 (npm)

Node-RED is a programming tool for wiring together hardware devices, APIs and online services in new and interesting ways.

It provides a browser-based editor that makes it easy to wire together flows using the wide range of nodes in the palette that can be deployed to its runtime in a single-click.

Features

Get Started

Community



Node-RED

<https://nodered.org/>

Node.jsベース
イケテル
ローコード
ツール

Node-REDをWindowsにインストール

1. Node.js をインストール

<https://nodejs.org/ja/>

2. Node-REDをインストール

Windows+R、"cmd"とタイプしコマンドプロンプト起動

下記コマンドを入れて、インストール

```
> npm i -g --unsafe-perm node-red
```

3. Node-REDの起動

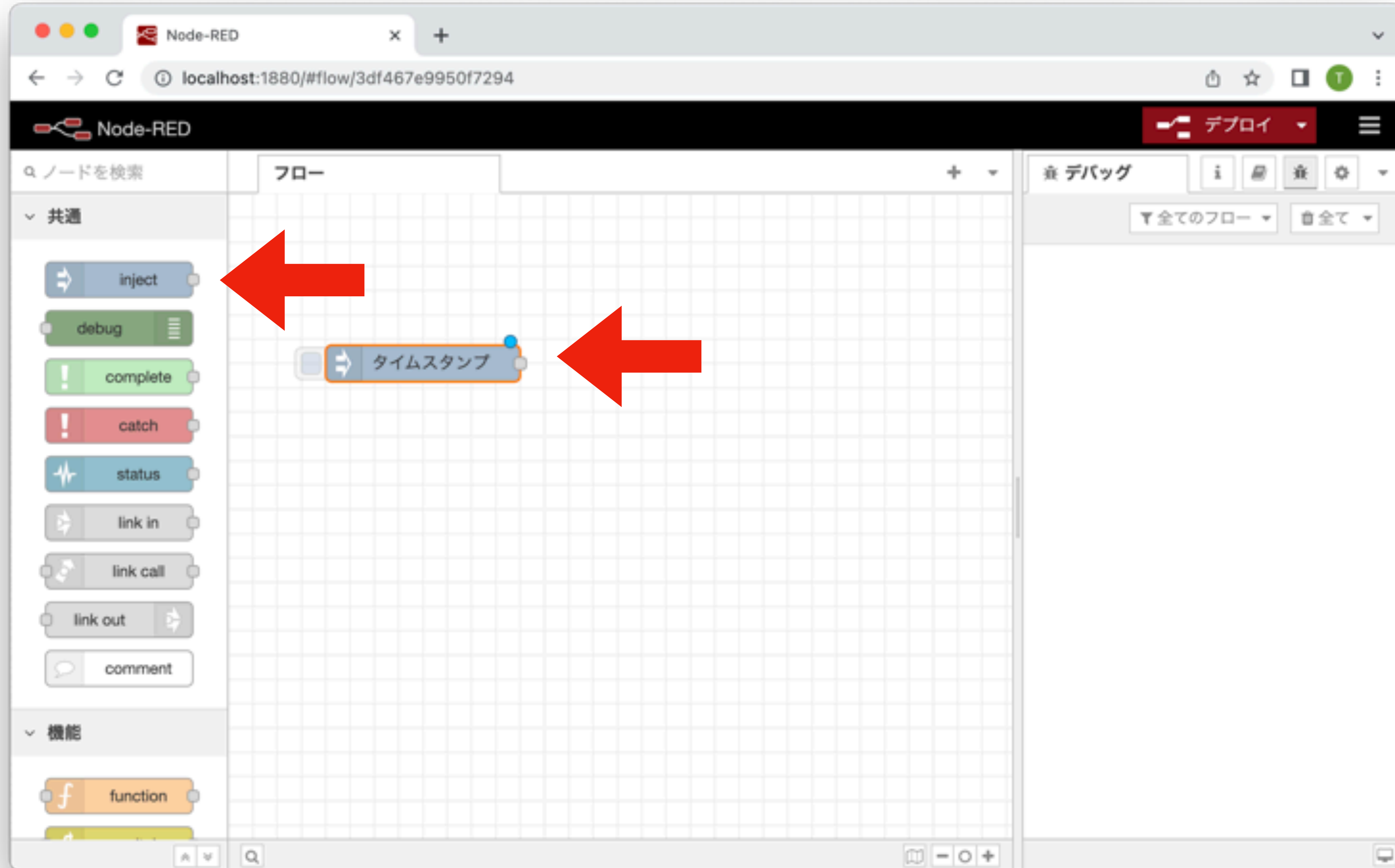
下記コマンドでNode-RED を起動する

```
> node-red
```

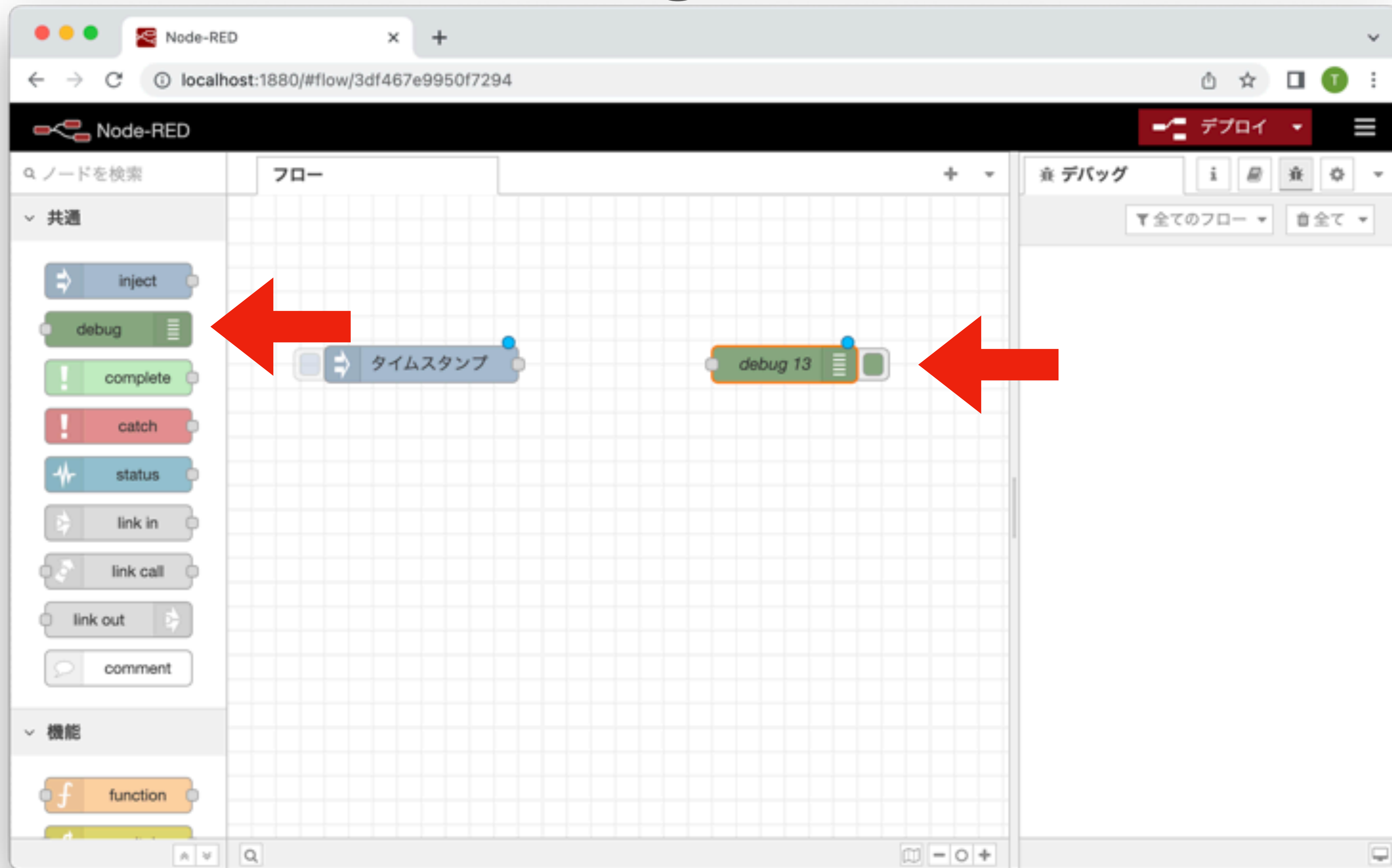
ブラウザで、 <http://localhost:1880/> を開く

Node-RED はじめのいっぽ

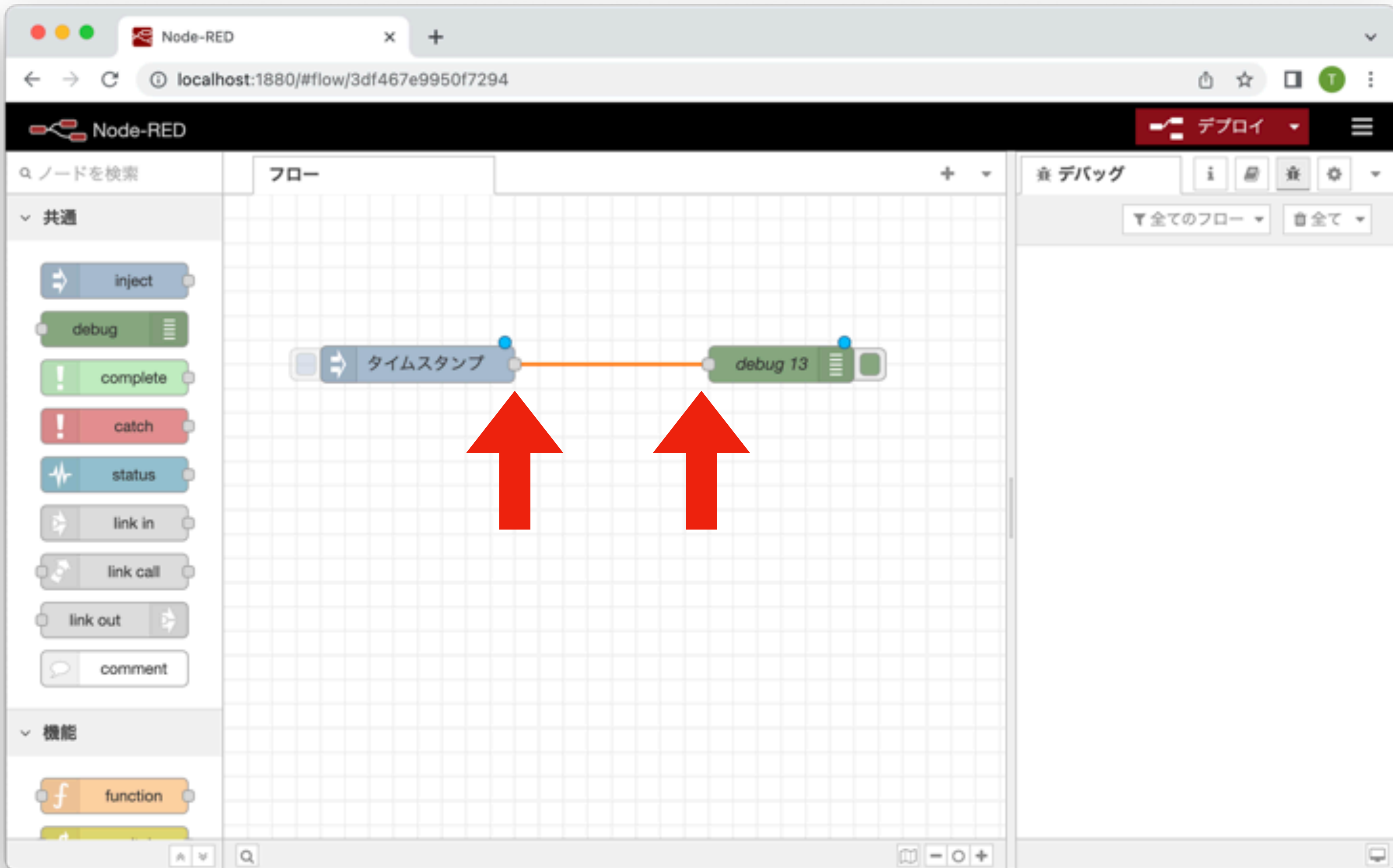
injectを置く (タイムスタンプになる)



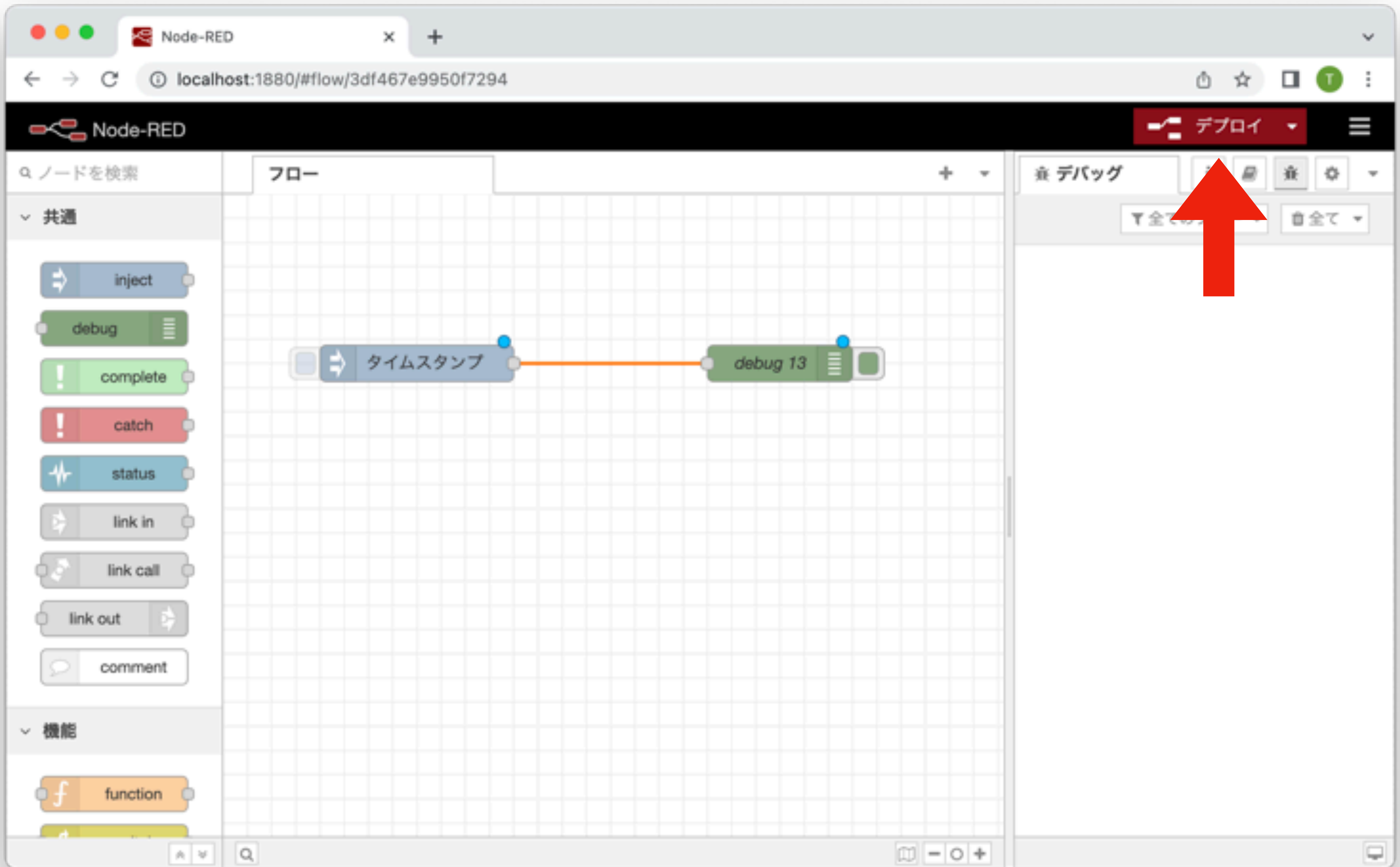
debugを置く



ドラッグして、つなぐ



デプロイする



タイムスタンプの左のボタンを押す

The screenshot shows the Node-RED web interface in a browser. The main workspace contains a flow with two nodes: 'タイムスタンプ' (Timestamp) and 'debug 13'. A red arrow points to the left button of the 'タイムスタンプ' node. On the right side, the 'デバッグ' (Debug) console is open, showing the output of the 'debug 13' node: '2022/8/12 12:16:20 ノード: debug 13' and 'msg.payload : number 1660274180844'. Another red arrow points from the 'debug 13' node to the debug console.

タイムスタンプの左のボタンを押す

デバッグ表示
される

はじめのいっぽ達成！

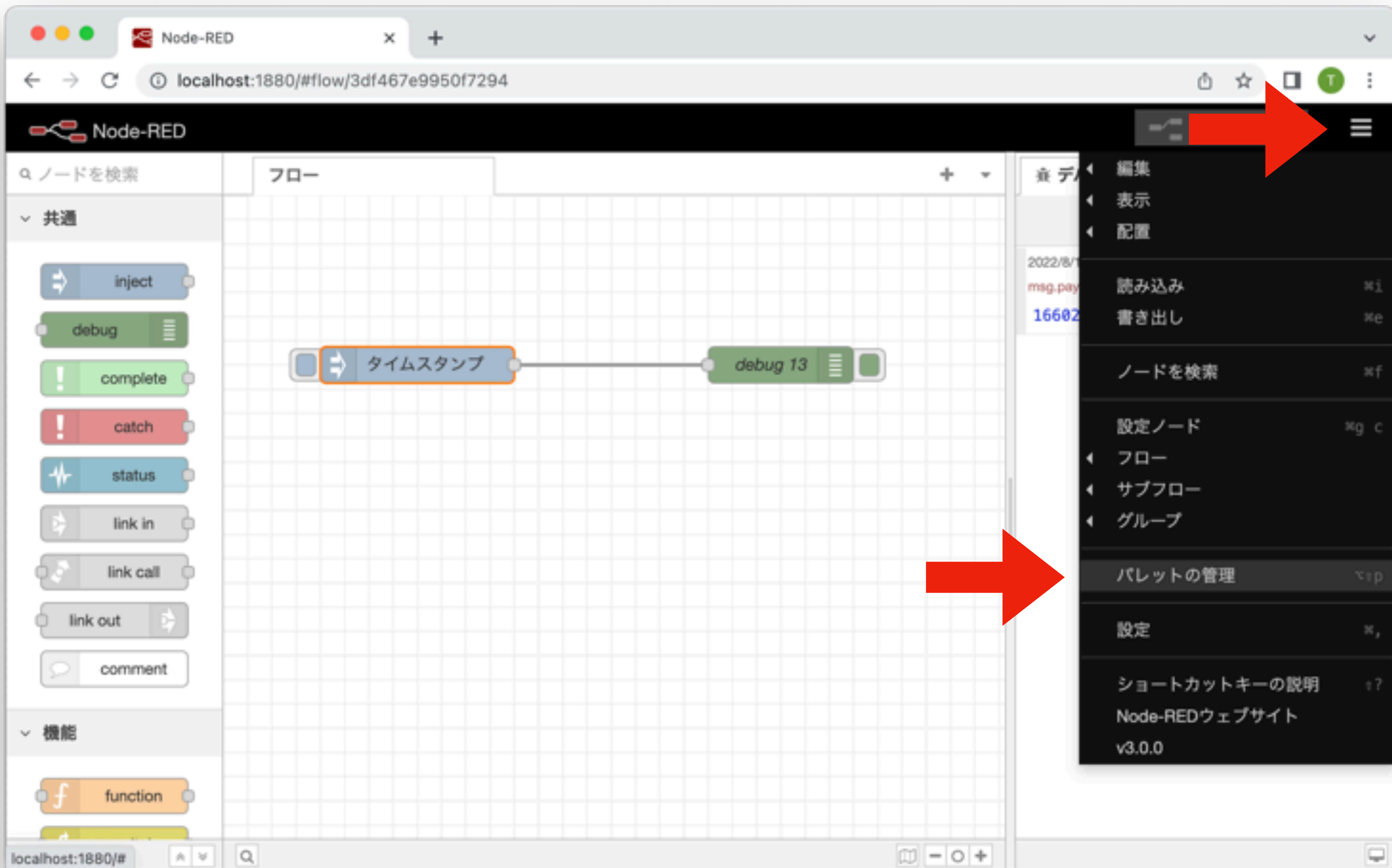
つくる→デプロイ→試す

健康観察アプリをつくらう！

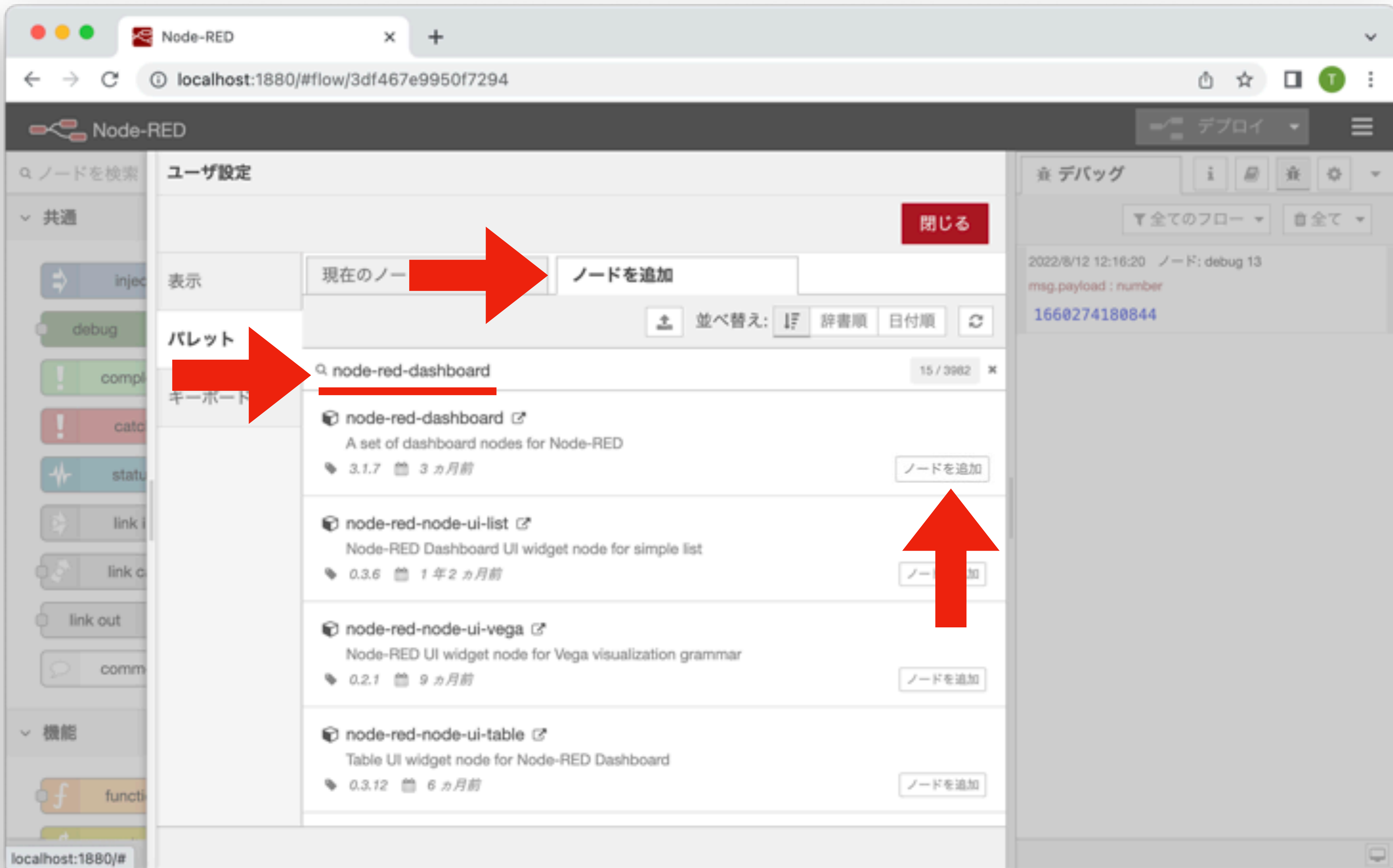
健康観察アプリ

名前、日付、体温をフォーム入力
ファイルに記録していく

右上メニュー、パレットの管理



ノードを追加、node-red-dashboard、ノードを追加

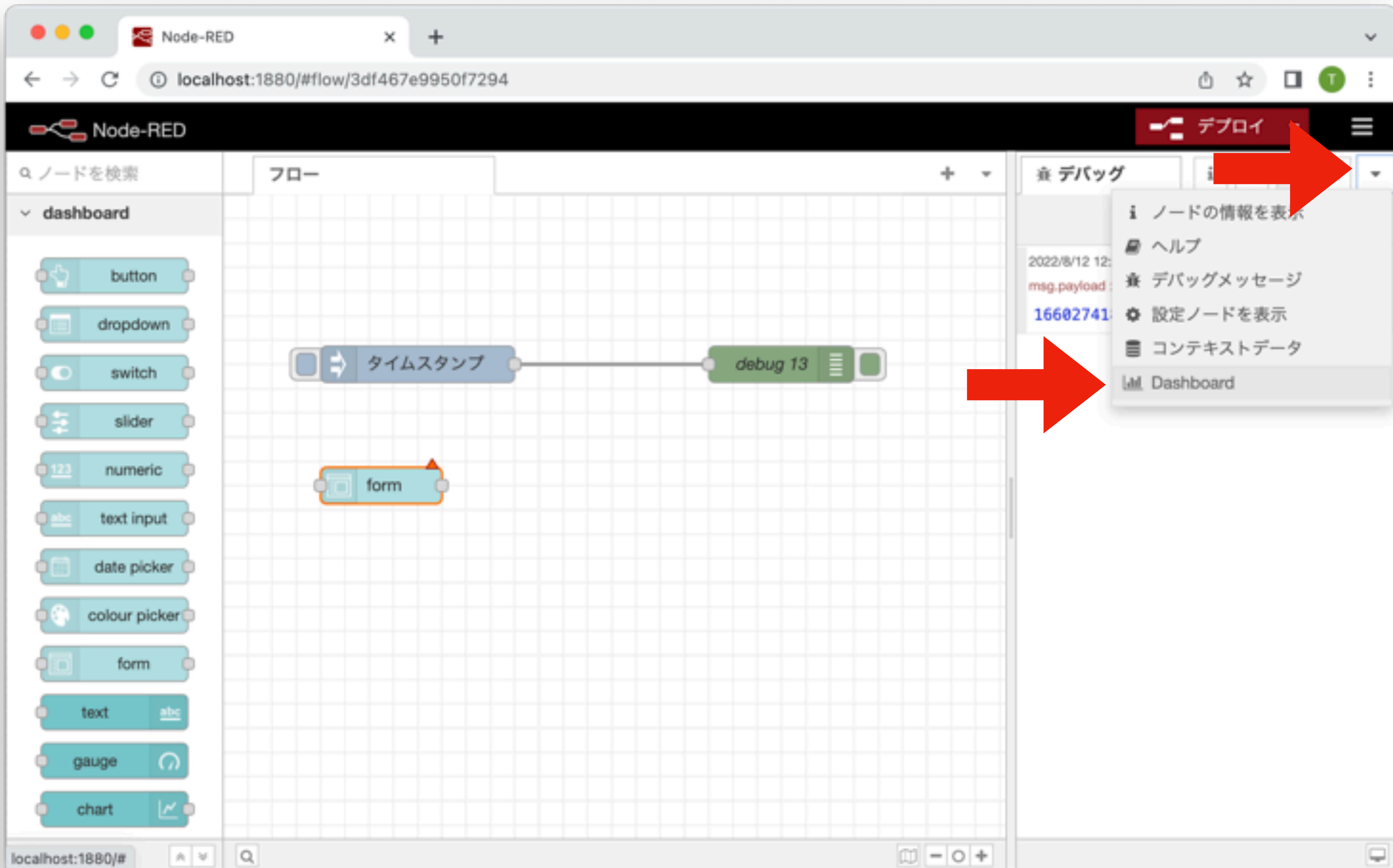


左パレットから”form”を探し、フローに置く

The screenshot shows the Node-RED web interface in a browser window. The address bar indicates the URL is `localhost:1880/#flow/3df467e9950f7294`. The interface is divided into three main sections:

- Left Palette (ノードを検索):** A list of nodes under the 'dashboard' category. The 'form' node is highlighted with a red arrow pointing to it from the left. A tooltip for the 'form' node is visible, stating:
form
ユーザインターフェイスにフォームを追加します。
node-red-dashboard : ui_form
- Center Canvas (フロー):** The workspace where a flow is being built. It contains a 'タイムスタンプ' (Timestamp) node connected to a 'debug 13' node. A red arrow points from the 'form' node in the left palette to the center canvas.
- Right Panel (デバッグ):** The debug console shows a message from 'debug 13' with a payload of 'number' and the value '1660274180844'.

右のメニューから”Dashboard”を選択



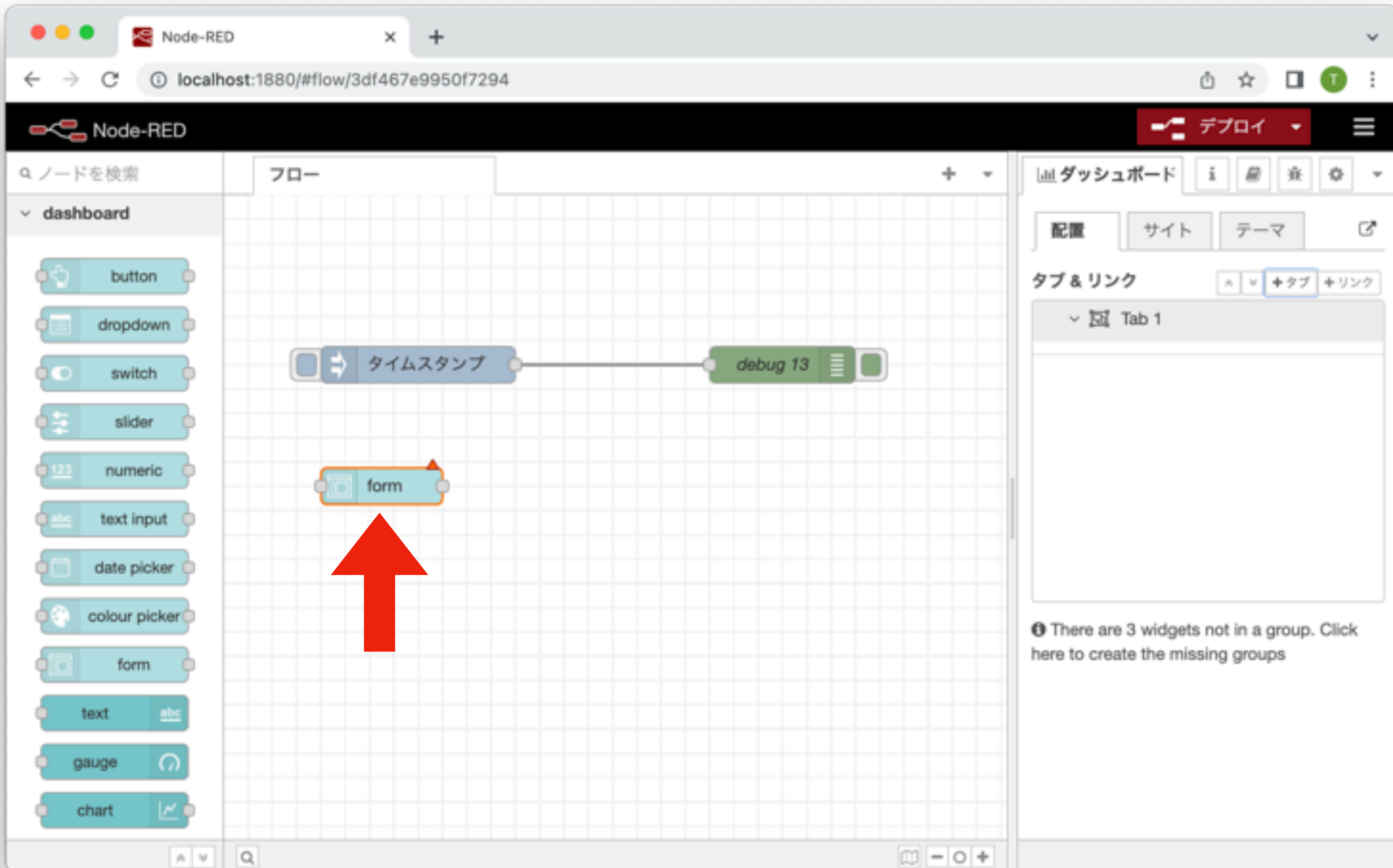
タブを一つ追加する

The screenshot shows the Node-RED web interface in a browser window. The address bar indicates the URL is `localhost:1880/#flow/3df467e9950f7294`. The interface is divided into three main sections:

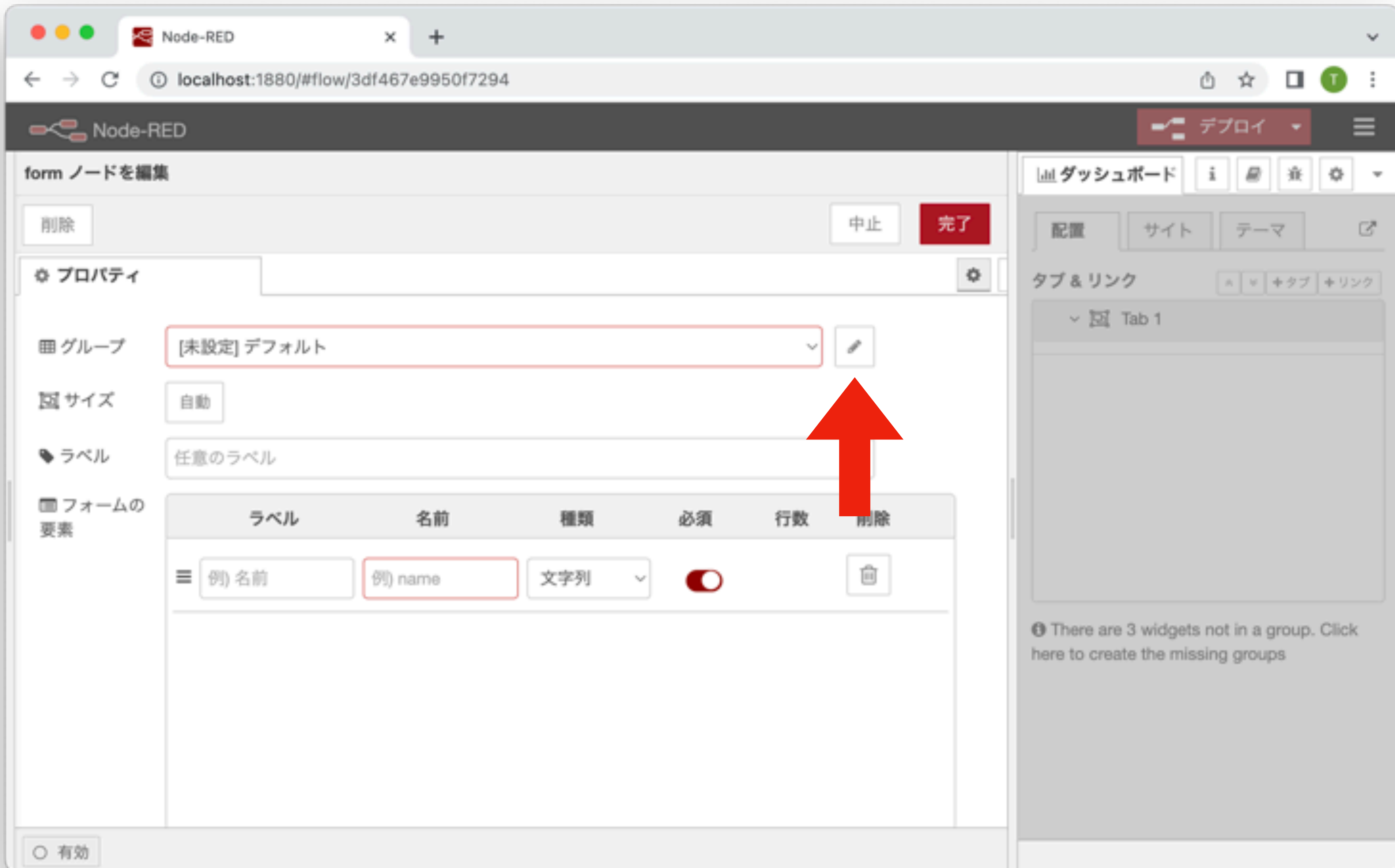
- Left Panel (Node Palette):** Contains a search bar and a list of widgets under the "dashboard" category. The widgets listed are: button, dropdown, switch, slider, numeric, text input, date picker, colour picker, form, text, gauge, and chart.
- Center Panel (Flow Canvas):** Displays a flow with two nodes: a "タイムスタンプ" (Timestamp) node and a "debug 13" node, connected by a wire. A "form" node is also present but not connected.
- Right Panel (Dashboard Settings):** Contains tabs for "ダッシュボード" (Dashboard), "サイト" (Site), and "テーマ" (Theme). Under the "ダッシュボード" tab, there is a "タブ & リンク" (Tabs & Links) section. A red arrow points to the "+タブ" (Add Tab) button in this section. Below the button, there is a list of existing tabs, currently showing "Tab 1".

At the bottom of the right panel, there is a message: "There are 3 widgets not in a group. Click here to create the missing groups".

“form”をダブルクリックする



グループの「」マークをクリック



The screenshot shows the Node-RED web interface in a browser window. The address bar indicates the URL is `localhost:1880/#flow/3df467e9950f7294`. The interface is in Japanese. The main area is titled 'form ノードを編集' (Edit form node). On the left, there are tabs for 'プロパティ' (Properties) and 'フォームの要素' (Form elements). The 'プロパティ' tab is active, showing a 'グループ' (Group) dropdown menu with the value '[未設定] デフォルト' (Not set, Default). A red arrow points to the pencil icon next to this dropdown. Below the group dropdown, there are fields for 'サイズ' (Size) set to '自動' (Automatic) and 'ラベル' (Label) set to '任意のラベル' (Arbitrary label). The 'フォームの要素' (Form elements) tab is also visible, showing a table with columns: ラベル (Label), 名前 (Name), 種類 (Type), 必須 (Required), 行数 (Rows), and 削除 (Delete). The table contains one row with the following values: ラベル: 例) 名前 (Example) Name, 名前: 例) name (Example) name, 種類: 文字列 (Text), 必須: checked, 行数: 1, and 削除: a trash icon. On the right side of the interface, there is a 'ダッシュボード' (Dashboard) section with tabs for '配置' (Layout), 'サイト' (Site), and 'テーマ' (Theme). Below these tabs, there is a 'タブ & リンク' (Tabs & Links) section with a 'Tab 1' tab. At the bottom of the dashboard, there is a message: 'There are 3 widgets not in a group. Click here to create the missing groups'.

Node-RED

localhost:1880/#flow/3df467e9950f7294

Node-RED

デプロイ

form ノードを編集

削除 中止 完了


プロパティ

グループ [未設定] デフォルト

サイズ 自動

ラベル 任意のラベル

フォームの要素

ラベル	名前	種類	必須	行数	削除
例) 名前	例) name	文字列	<input checked="" type="checkbox"/>	1	

ダッシュボード

配置 サイト テーマ

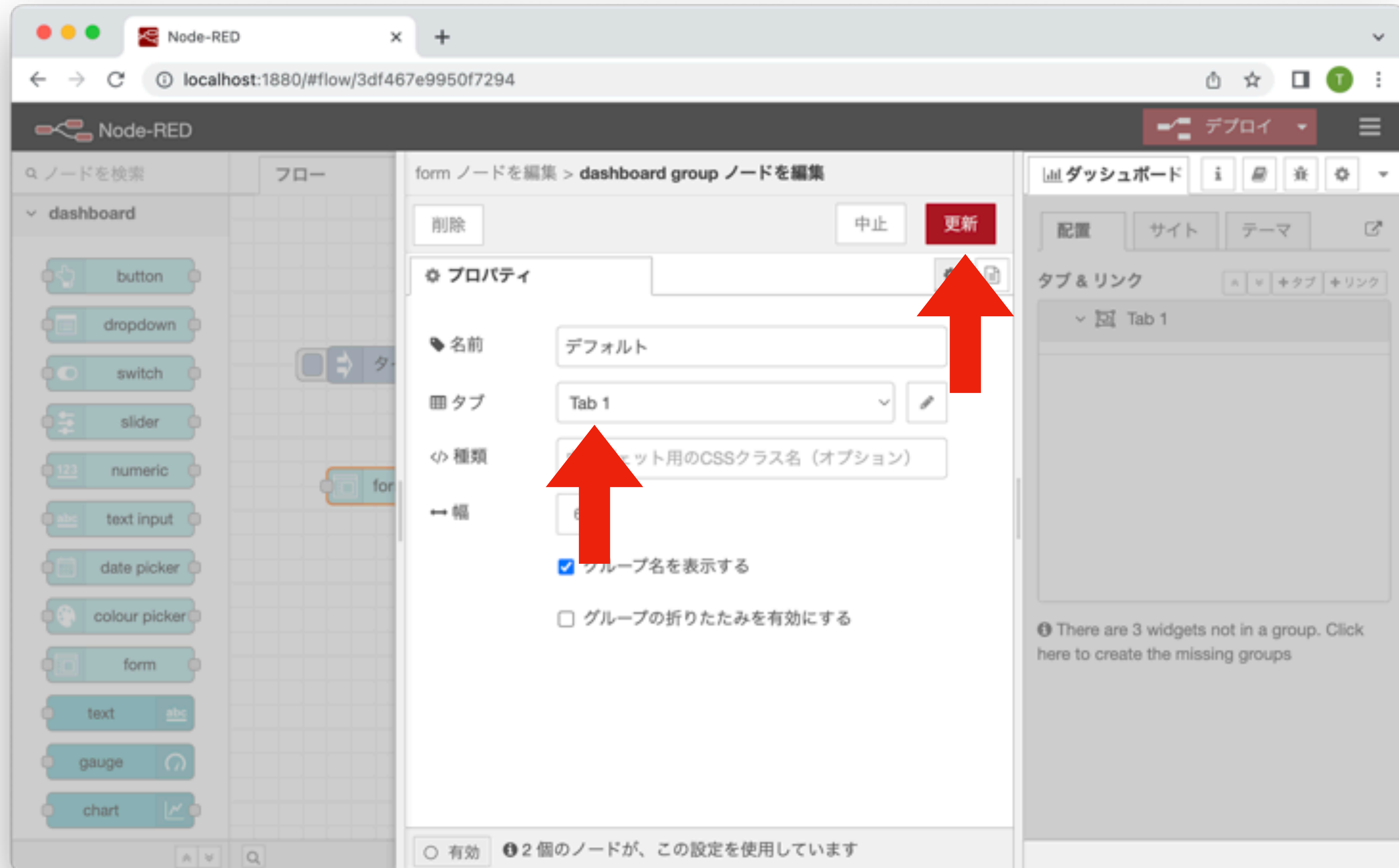
タブ & リンク

Tab 1

There are 3 widgets not in a group. Click here to create the missing groups

有効

タブからDashboardで追加した”Tab 1”を選択、更新



ラベルと名前に”名前”と記述し、完了

The screenshot shows the Node-RED web interface for editing a form. The browser tabs are 'Node-RED' and 'Node-RED ダッシュボード'. The address bar shows 'localhost:1880/#'. The interface is in Japanese.

On the left, the 'form ノードを編集' (Edit form node) panel is active. It contains a 'プロパティ' (Properties) section with fields for 'グループ' (Group) set to '[Tab 1] デフォルト', 'サイズ' (Size) set to '自動' (Automatic), and 'ラベル' (Label) set to '任意のラベル' (Arbitrary label). Below this is a table for 'フォームの要素' (Form elements).

ラベル	名前	種類	必須	行数	削除
名前	名前	文字列	<input checked="" type="checkbox"/>		

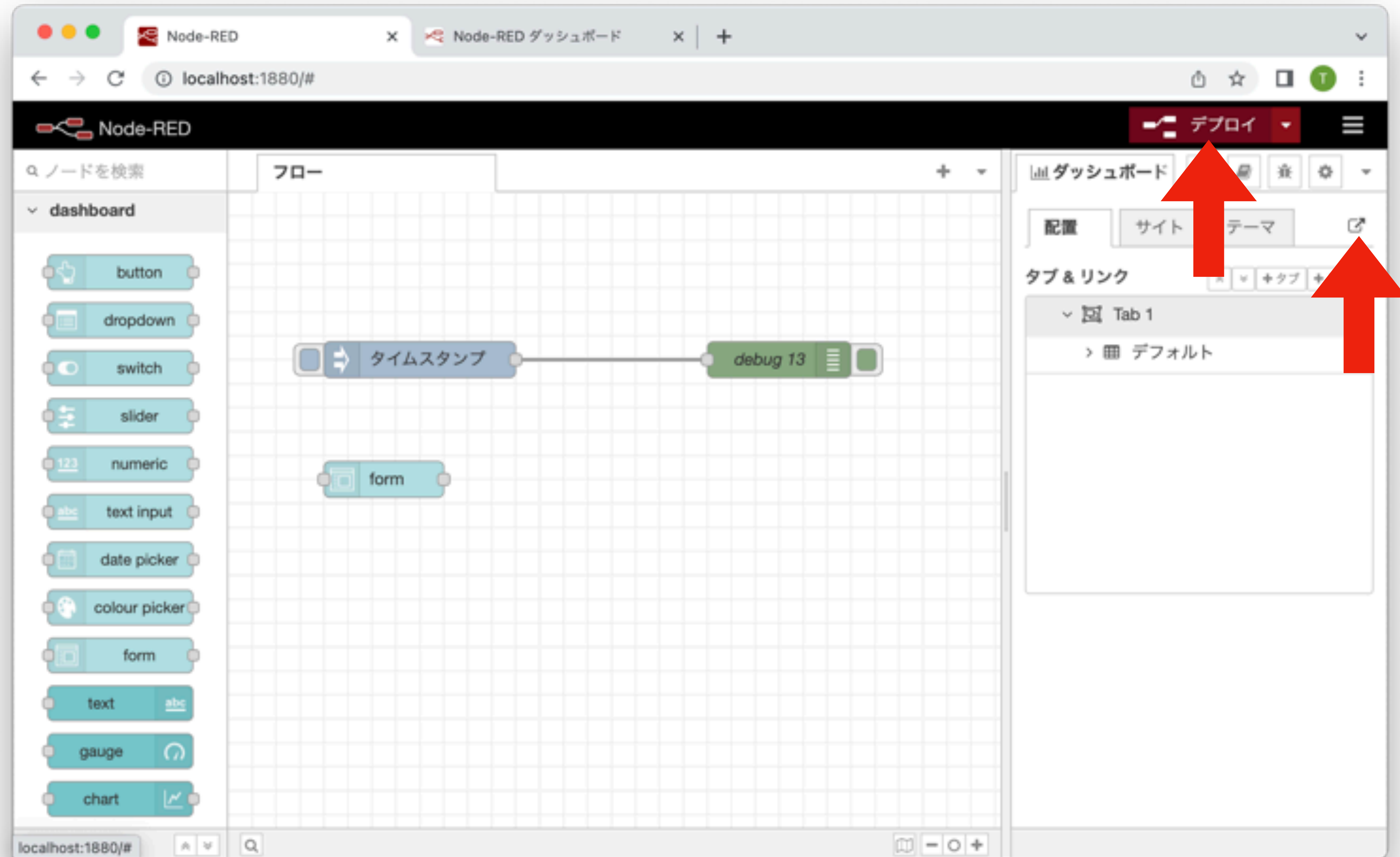
Two red arrows point to the '名前' (Name) field in the first row of the table. The '必須' (Required) checkbox is checked.

At the top right of the form editor, there are buttons: '削除' (Delete), '中止' (Cancel), and '完了' (Done). A red arrow points to the '完了' button.

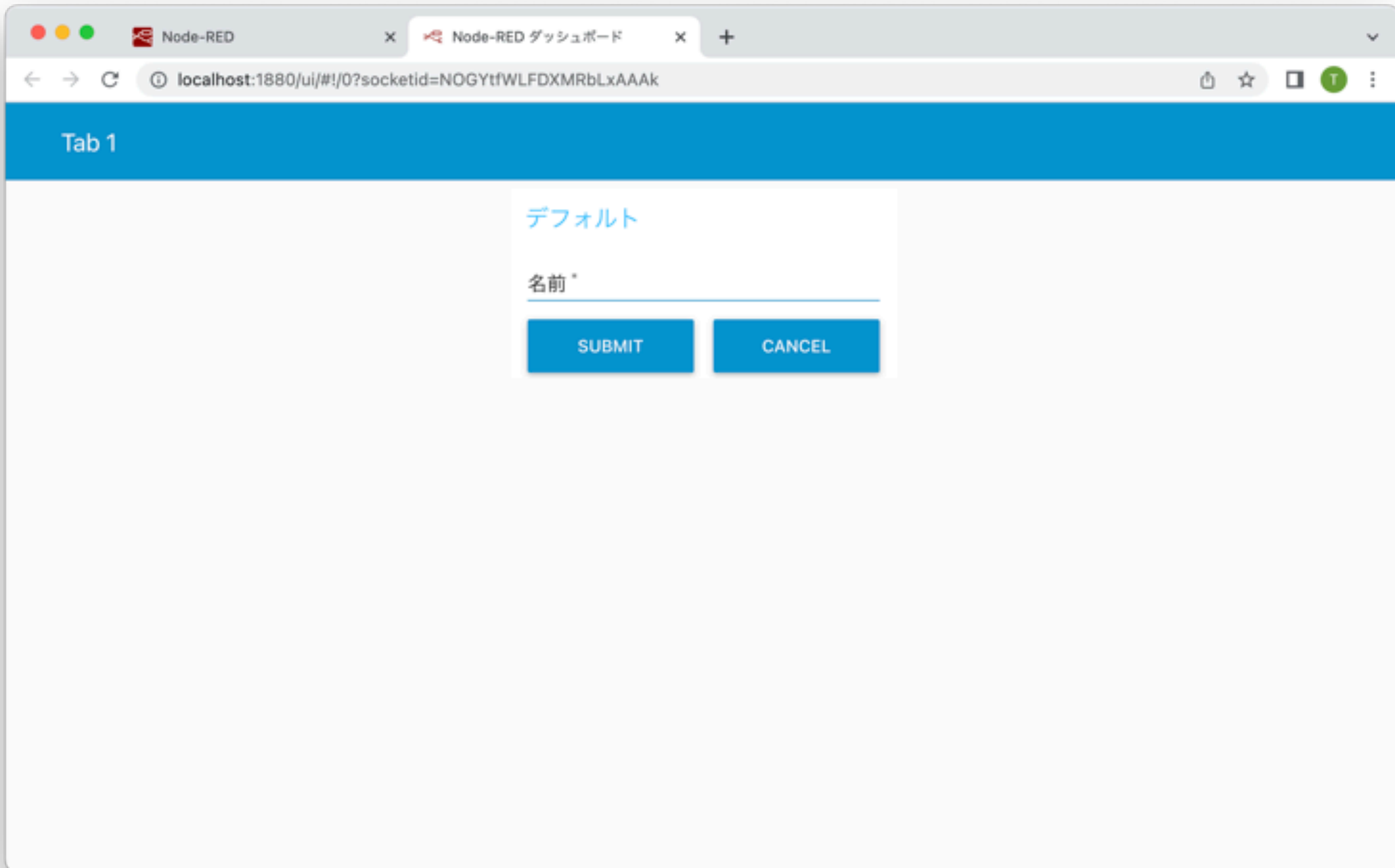
On the right, the 'ダッシュボード' (Dashboard) panel is visible, showing a 'タブ & リンク' (Tabs & Links) section with a 'Tab 1' tab and a 'デフォルト' (Default) group.

At the bottom left, there is a checkbox labeled '有効' (Enabled) which is currently unchecked.

デプロイして、Dashboardを開く



フォームで「名前」が入力できる



The screenshot shows a web browser window with two tabs. The active tab is titled "Node-RED ダッシュボード". The address bar shows the URL "localhost:1880/ui/#!/0?socketid=NOGYtfWLFDXMRbLxAAAk". The page has a blue header bar with the text "Tab 1". The main content area is white and contains a form titled "デフォルト" in blue text. The form has a text input field labeled "名前*" (Name*) with a blue underline. Below the input field are two blue buttons: "SUBMIT" and "CANCEL".

Node-RED

Node-RED ダッシュボード

localhost:1880/ui/#!/0?socketid=NOGYtfWLFDXMRbLxAAAk

Tab 1

デフォルト

名前*

SUBMIT CANCEL

“+要素”を押し、フォームに日付と体温を追加し、完了

The screenshot shows the Node-RED web interface for editing a form. The browser tabs are 'Node-RED' and 'Node-RED ダッシュボード'. The address bar shows 'localhost:1880/#'. The interface has a dark header with 'Node-RED' and a 'デプロイ' button. The main area is titled 'form ノードを編集'. On the left, there are buttons for '削除', '中止', and '完了'. Below these is a 'プロパティ' section and a 'フォームの要素' section. The 'フォームの要素' section contains a table with three rows of form elements. A red arrow points to the '+要素' button at the bottom of this section. Another red arrow points to the '完了' button. A third red arrow points to the '体温' field in the table.

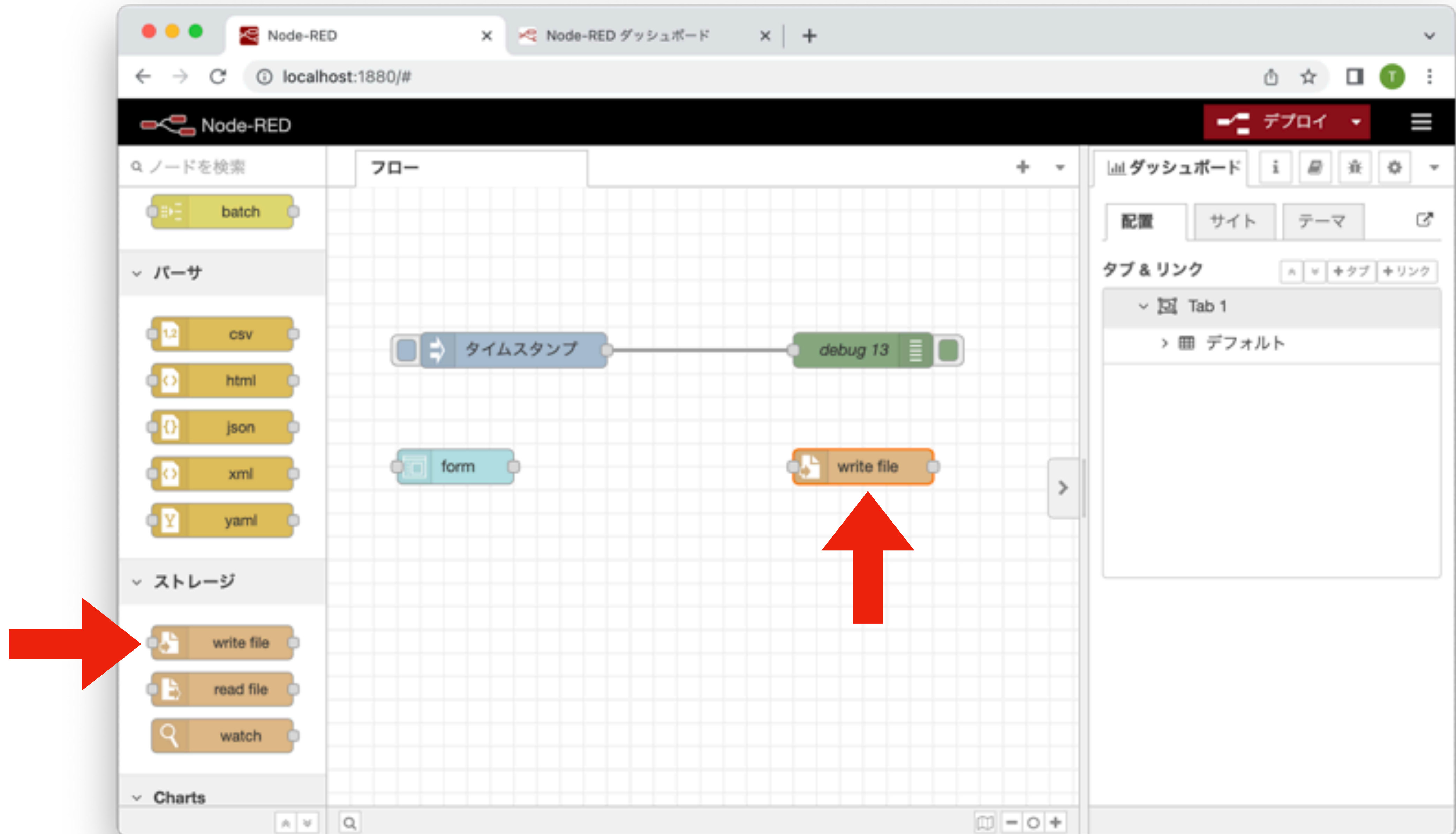
ラベル	名前	種類	必須	行数	削除
名前	名前	文字列	<input checked="" type="checkbox"/>		
日付	日付	日付	<input checked="" type="checkbox"/>		
体温	体温	数値	<input checked="" type="checkbox"/>		

+要素

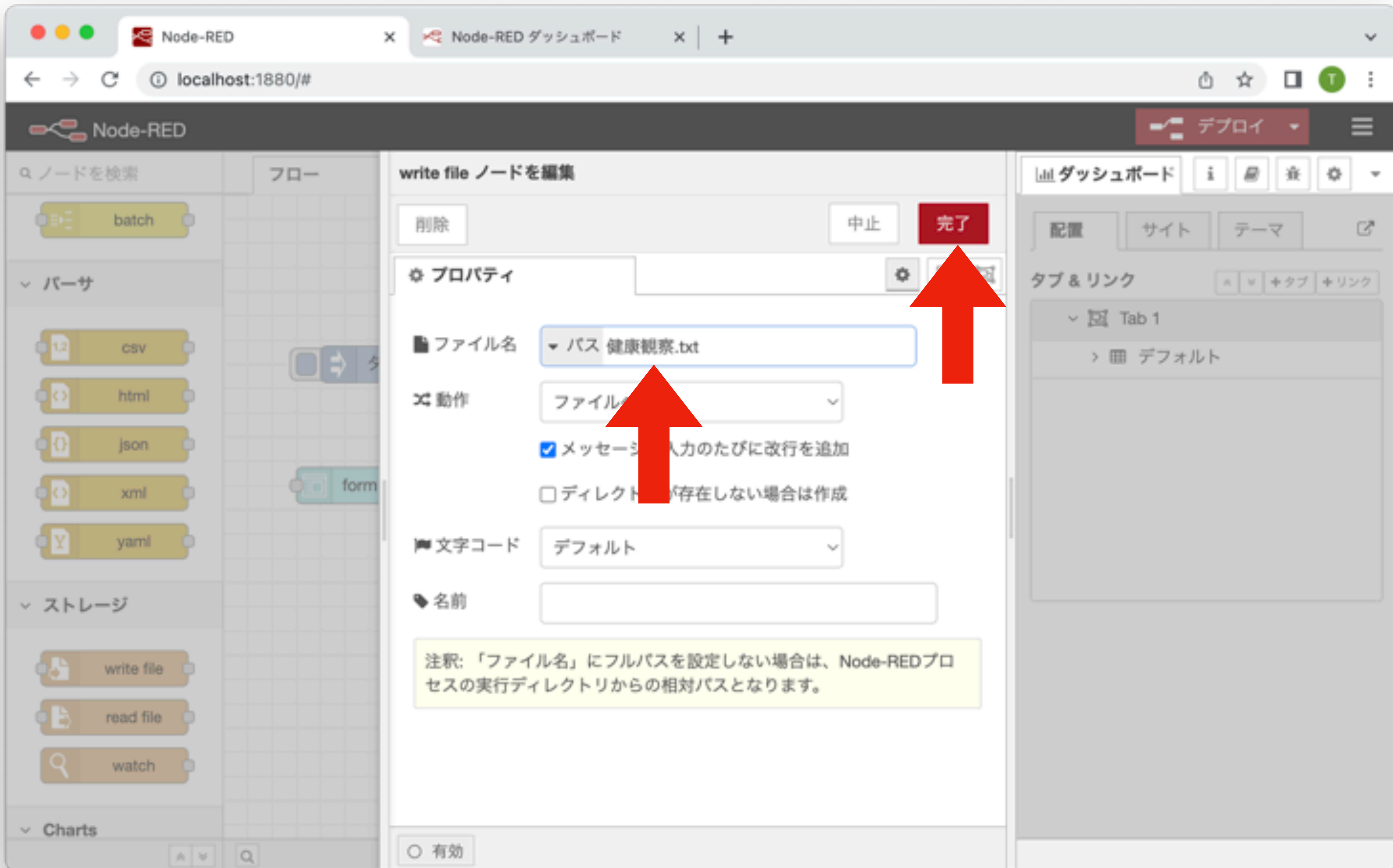
ボタン submit cancel

有効

ストレージ”write file”を置き、ダブルクリック



パスにファイル名”健康観察.txt”を書き、完了

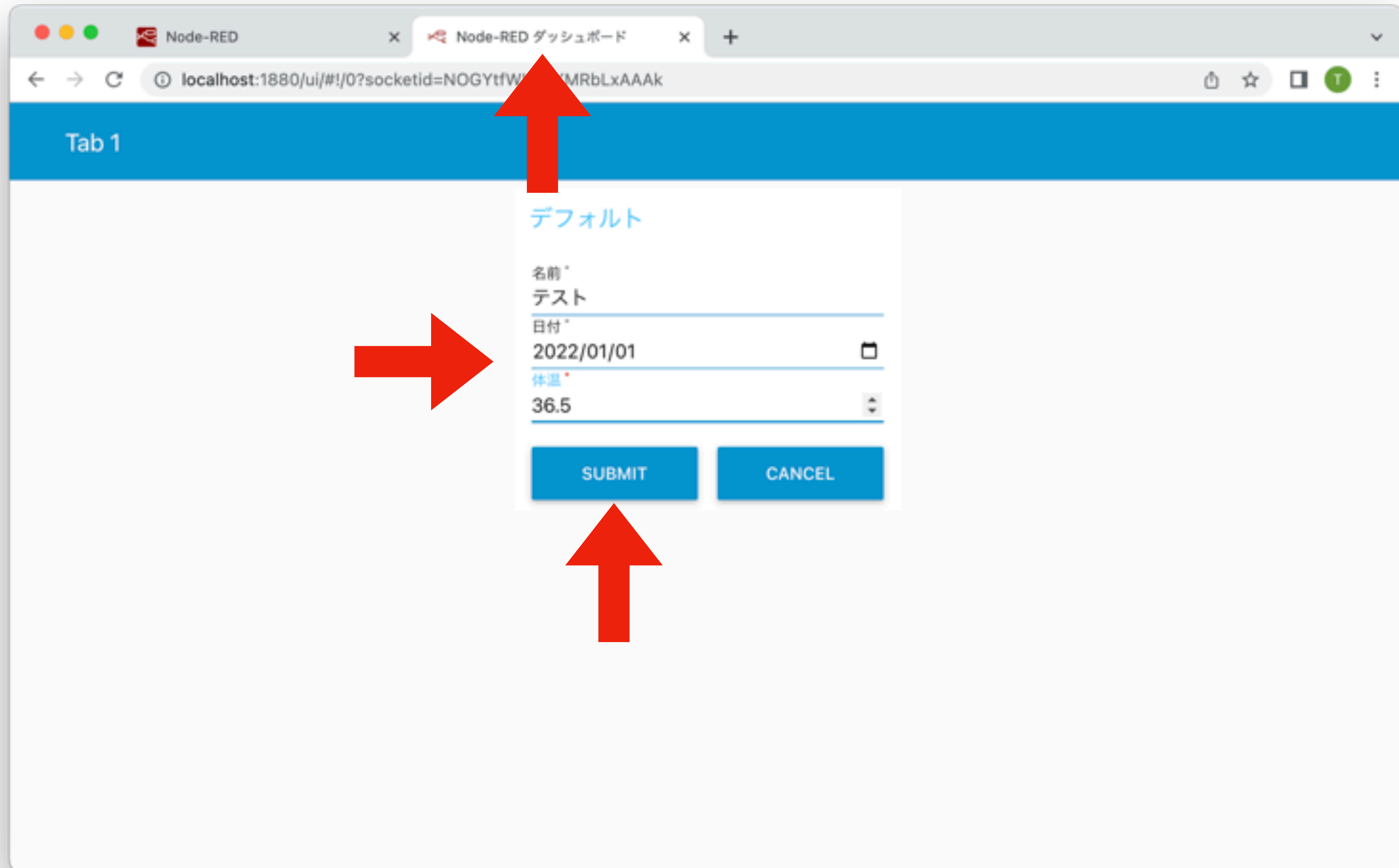


“form”と”健康観察.txt”をつなぎ、デプロイ

The screenshot shows the Node-RED web interface in a browser window. The address bar indicates the URL is `localhost:1880/#`. The interface is divided into several sections:

- Left Panel (Node Palette):** Contains a search bar and categories of nodes. The 'Storage' (ストレージ) category is expanded, showing nodes like 'write file', 'read file', and 'watch'. Other categories include 'Batch' (batch), 'Parsers' (パーサ) with 'csv', 'html', 'json', 'xml', and 'yaml', and 'Charts'.
- Center Canvas (Flow):** Displays a workflow with two parallel horizontal flows. The top flow consists of a 'Timestamp' (タイムスタンプ) node connected to a 'debug 13' node. The bottom flow consists of a 'form' node connected to a '健康観察.txt' (Health Observation.txt) node. A large red arrow points upwards from the bottom of the canvas towards the 'form' node.
- Right Panel (Dashboard):** Features a 'Deploy' (デプロイ) button at the top, highlighted by a red arrow. Below it are tabs for 'Configuration' (配置), 'Site' (サイト), and 'Form' (フォーマット). The 'Form' tab is currently selected. Under the 'Tabs & Links' (タブ & リンク) section, there is a 'Tab 1' dropdown menu showing a 'Default' (デフォルト) view.

Dashboardに切り替えて、記述し、SUBMIT



SUBMITする毎に健康観察.txtにデータが追記される

健康観察.txt



“テキストエディット”で開く

```
{"名前":"テスト","日付":"2021-12-31T15:00:00.000Z","体温":36.5}  
{"名前":"テスト2","日付":"2022-01-01T15:00:00.000Z","体温":36.9}
```

健康観察アプリ達成！

自分のIPアドレスを使ってフォームを公開できる！

（使用者はNode-RED不要、ブラウザだけでOK）

→ EVER/IPを使えばセキュアにグローバルにつながる

Node-RED の管理パスワードをつける


コマンドプロンプトで、パスワードのハッシュを作成する

> node-red admin hash-pw

Password: で聞かれるので、パスワードを入力、ハッシュをコピー

設定ファイルを開いて、下記の用に編集用のID/PWを設定する

```
adminAuth: {  
  type: "credentials",  
  users: [{  
    username: "admin",  
    password: "$2b$08$YG9hV13Uc/HHEtu9Nxq0aumN9V83zAMD8kSao8c8By_E_9L7Fc03y",  
    permissions: "*"   
  }]  
},
```



コピーしたハッシュを設定する

Node-RED EVER-IPを使用する場合

設定ファイルを開いて IPv6 で使えるように編集する

settings.js を開く

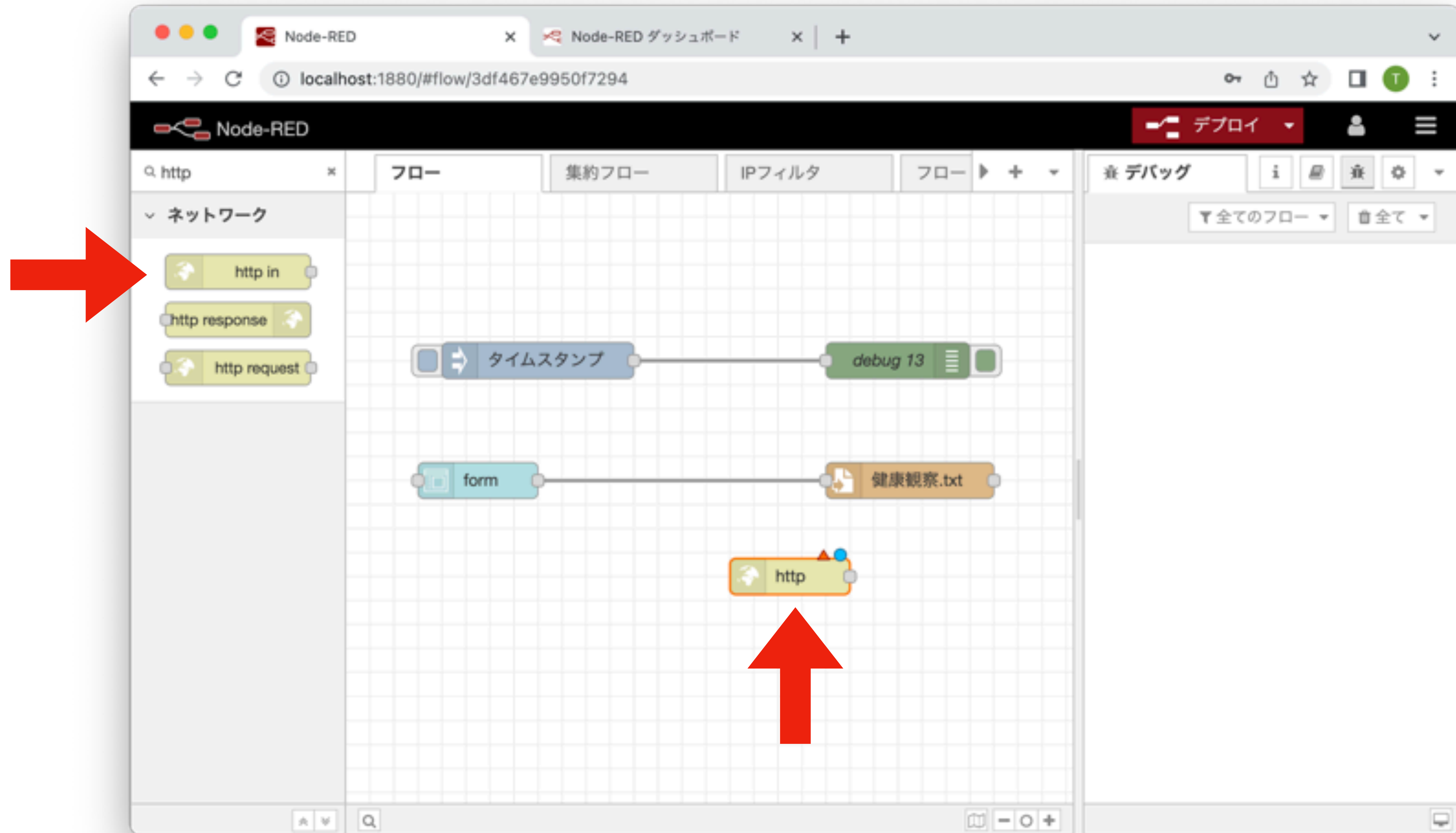
下記の記述を探す

```
//uiHost: "127.0.0.1",
```

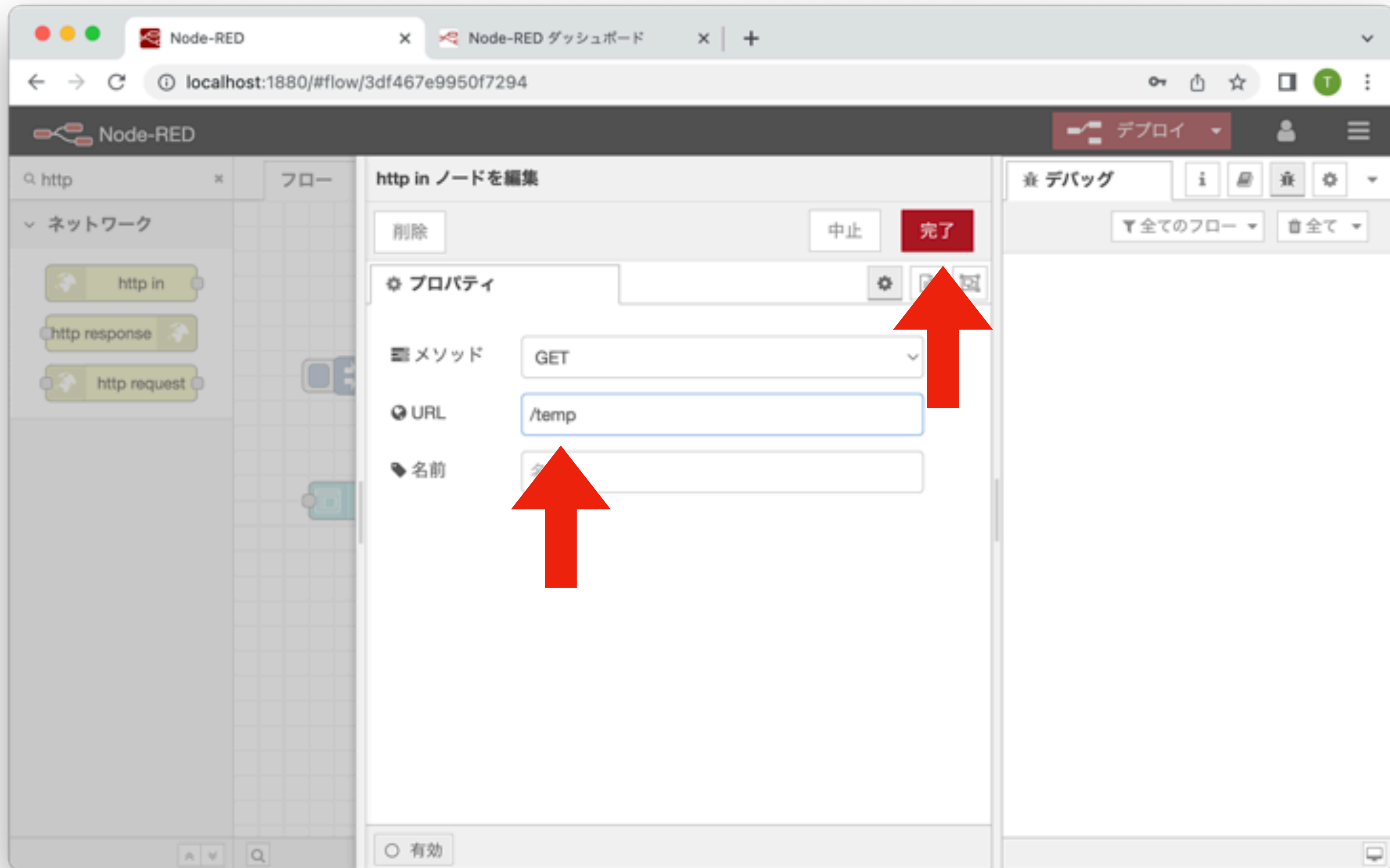
“//” を削除し、“127.0.0.1” を “::” と変更する

```
> uiHost: "::",
```

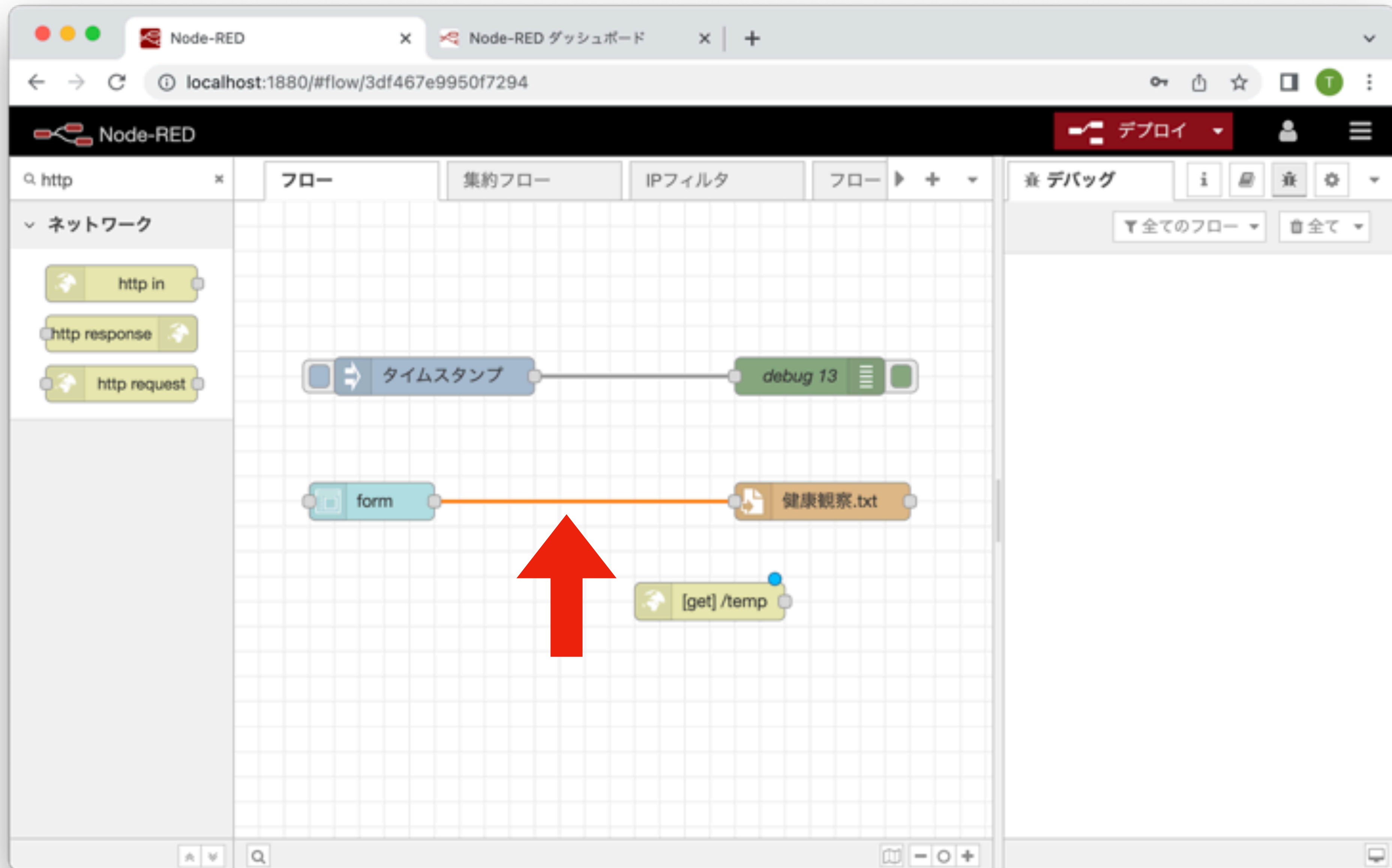
受信するノード”http in”を置き、ダブルクリック



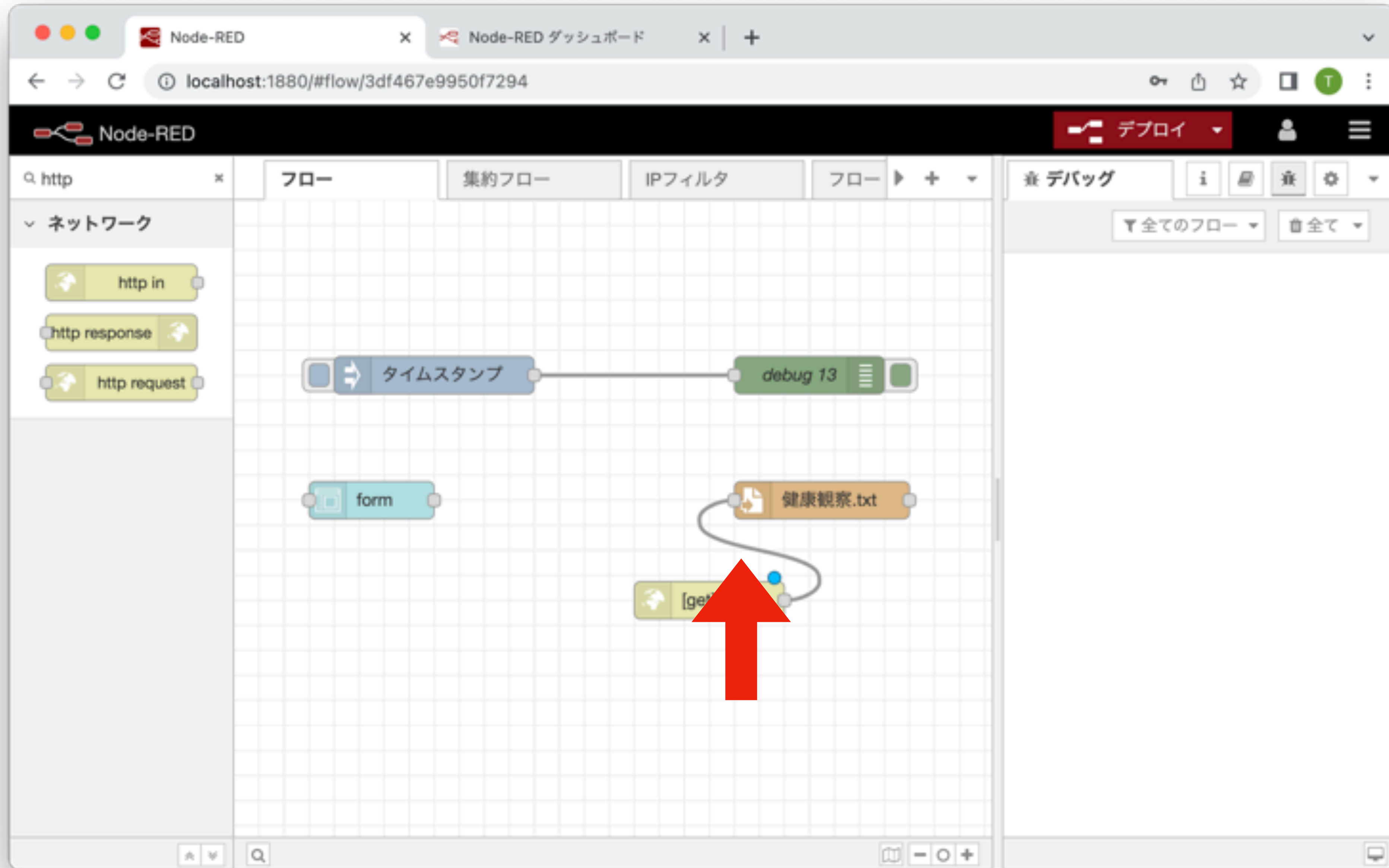
URLに “/temp” と設定し、完了



“form”と”健康観察.txt”をつなぐ線を選択して、delete



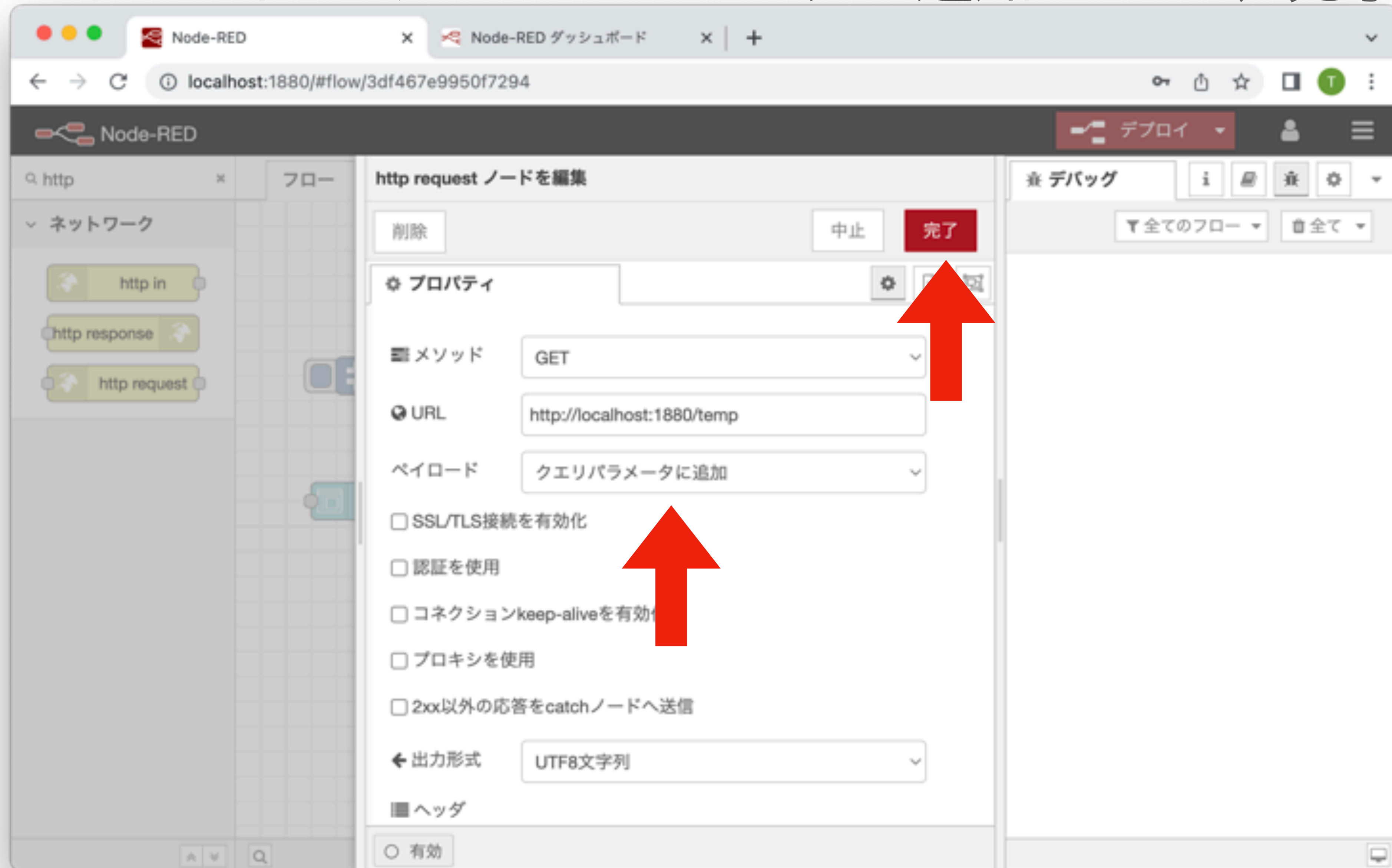
“http in”と”健康観察.txt”をつなぐ



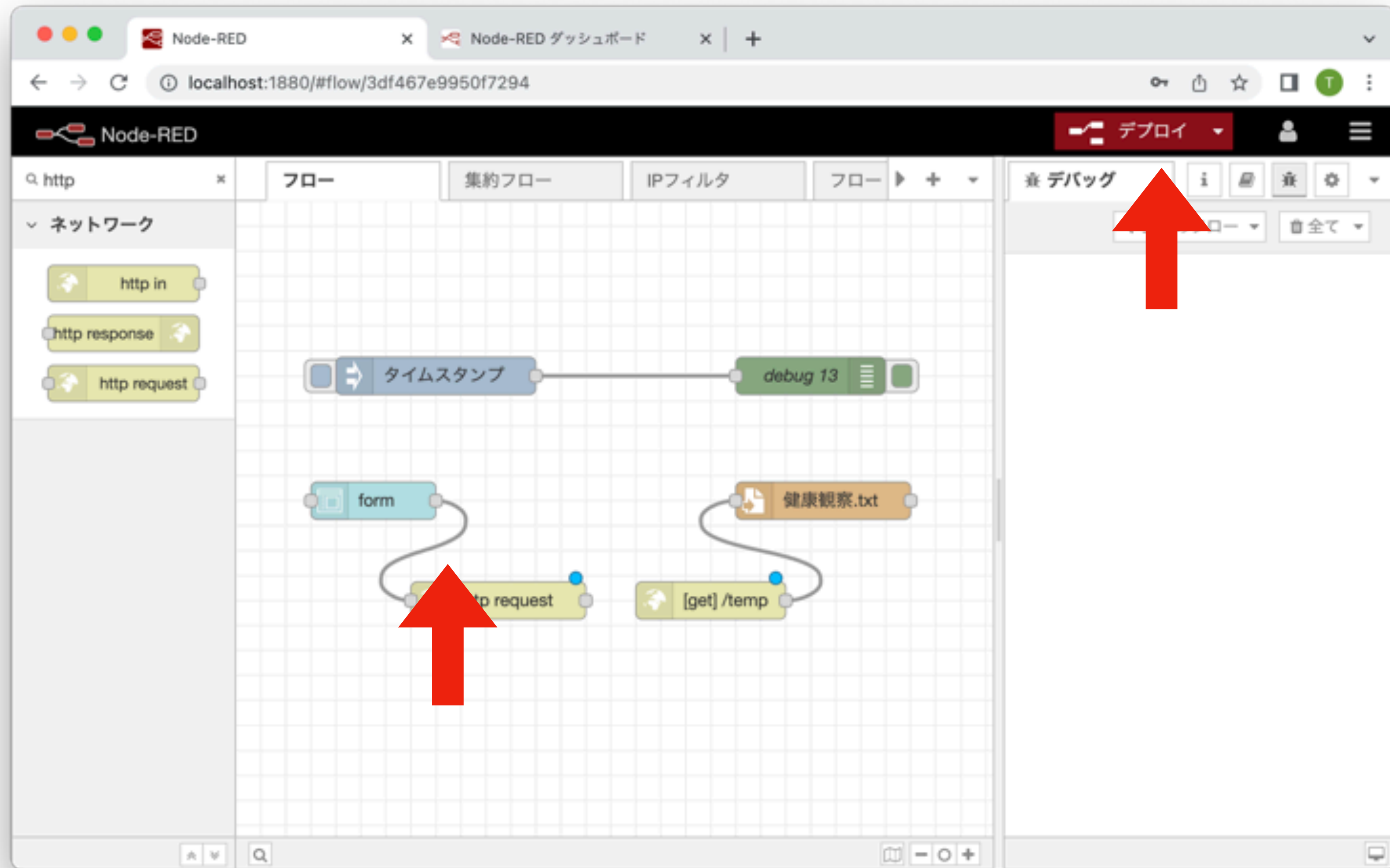
通信するノード”http request”を置き、ダブルクリック



URLに通信先 “http://localhost:1880/temp” 設定し、
ペイロードを”クエリパラメータに追加”にして、完了

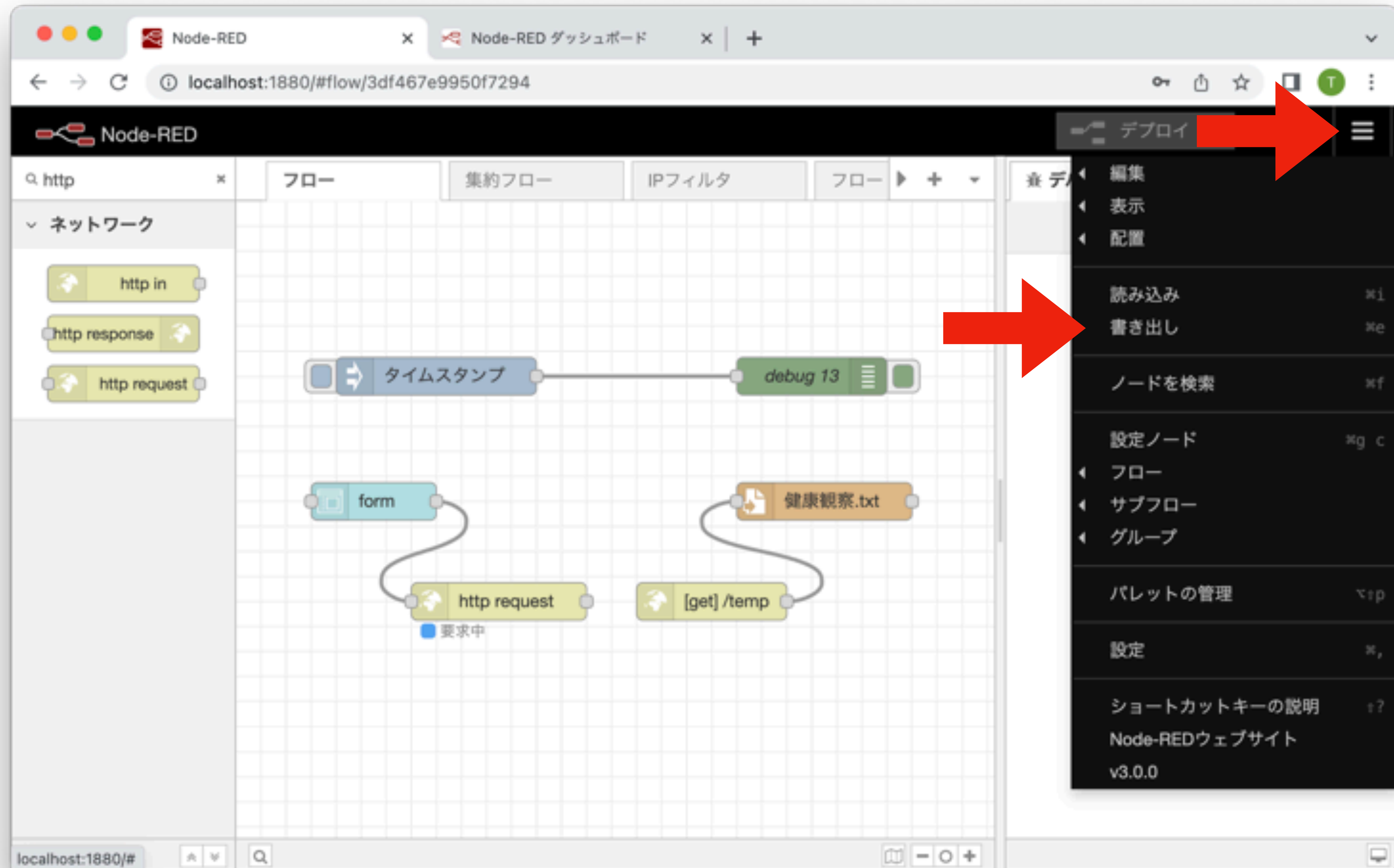


“form”と“http request”をつなぎ、デプロイ、確認する

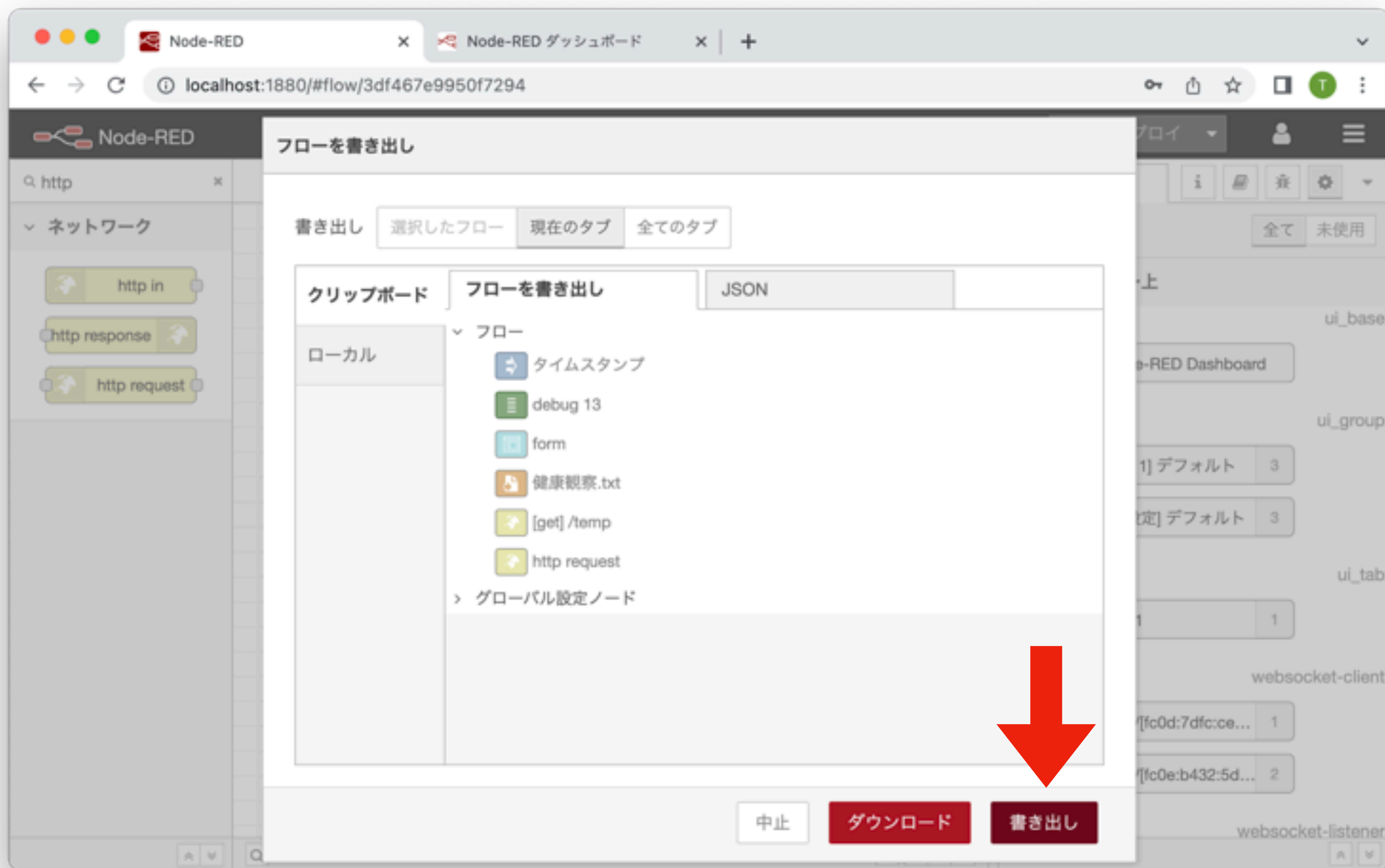


複数のコンピューターを
つないで動かす

フローを書き出すために、右メニュー、書き出し

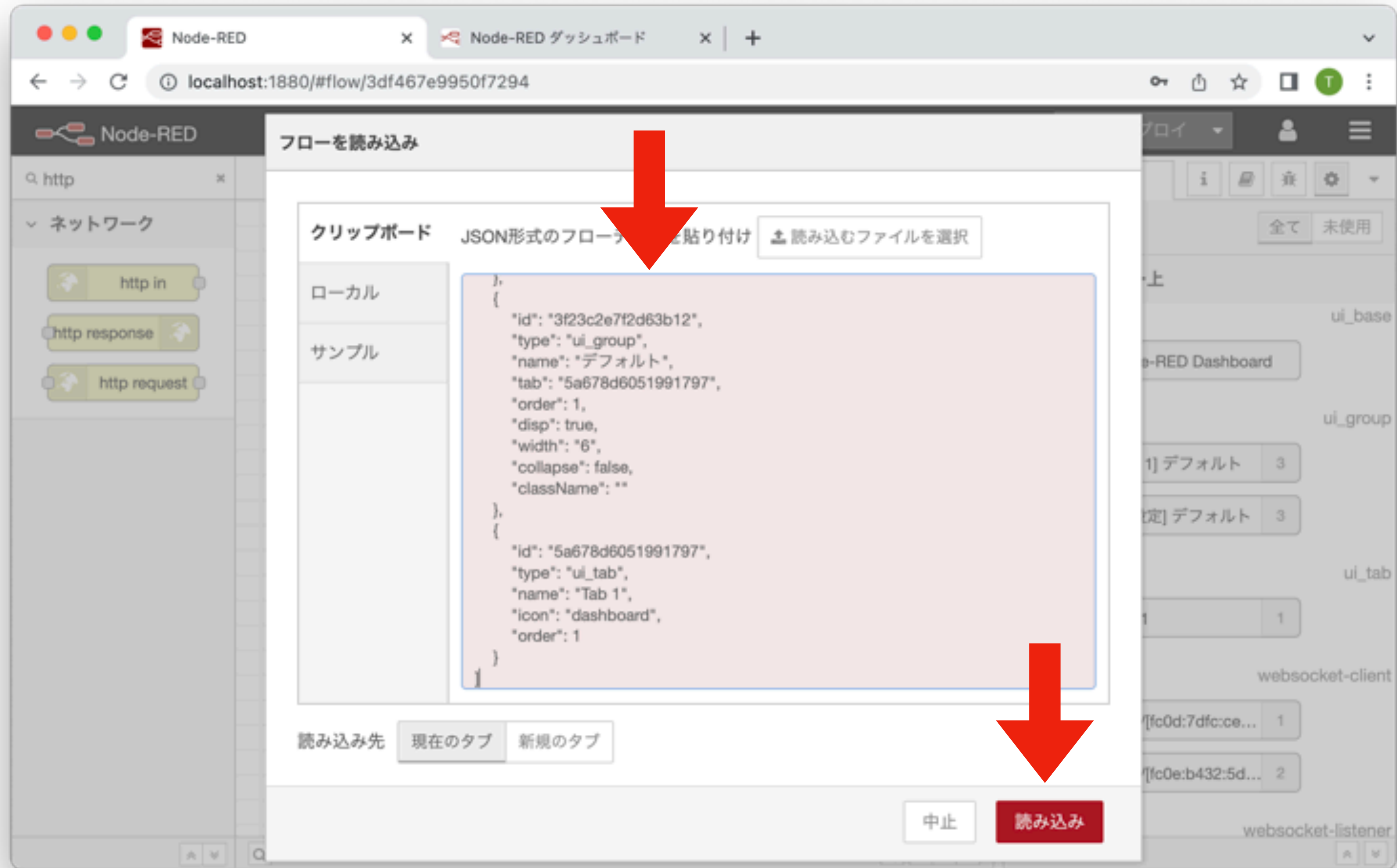


“書き出し”を選択、フローが、クリップボードにコピーされる

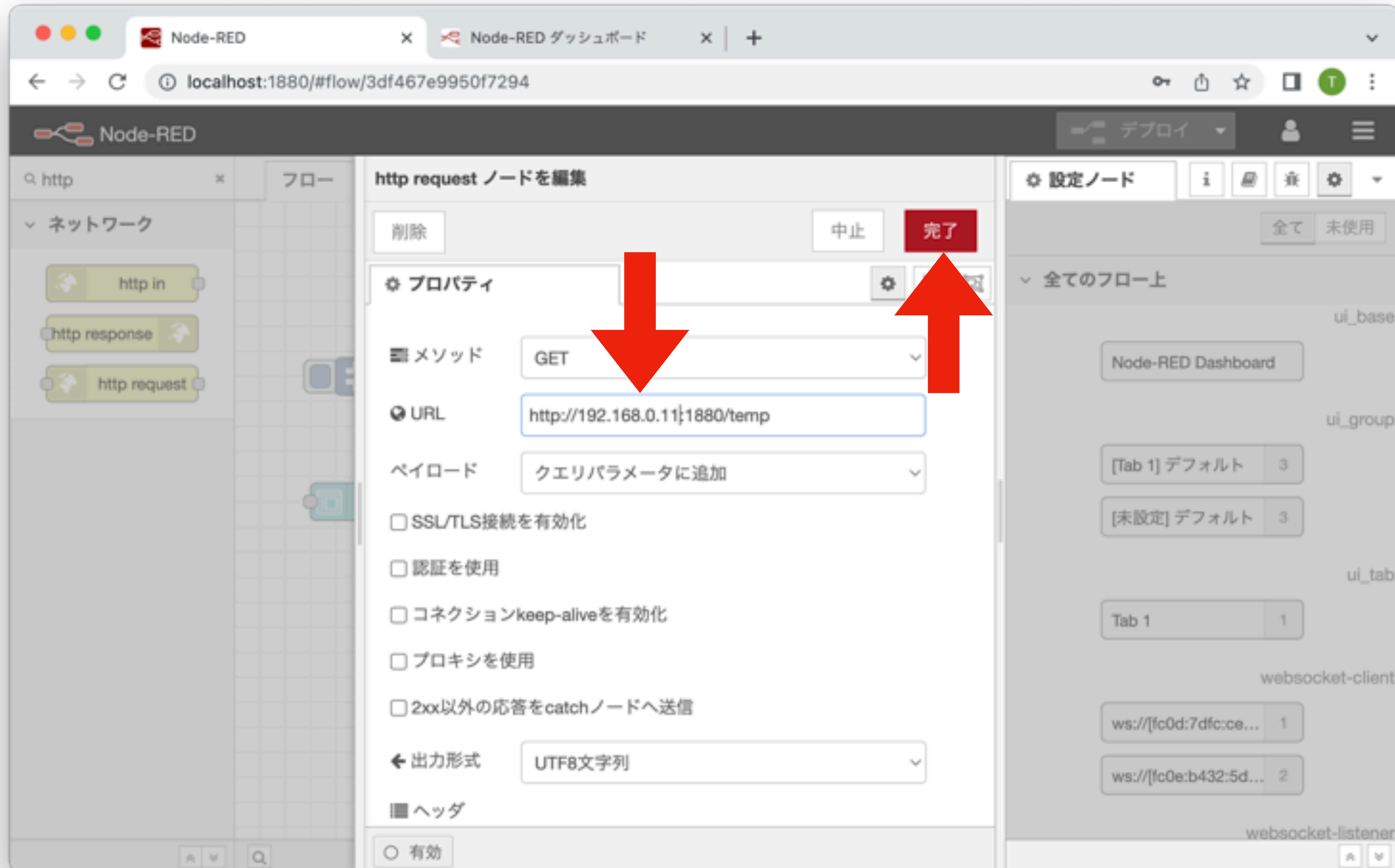


別のコンピューターに
設定する

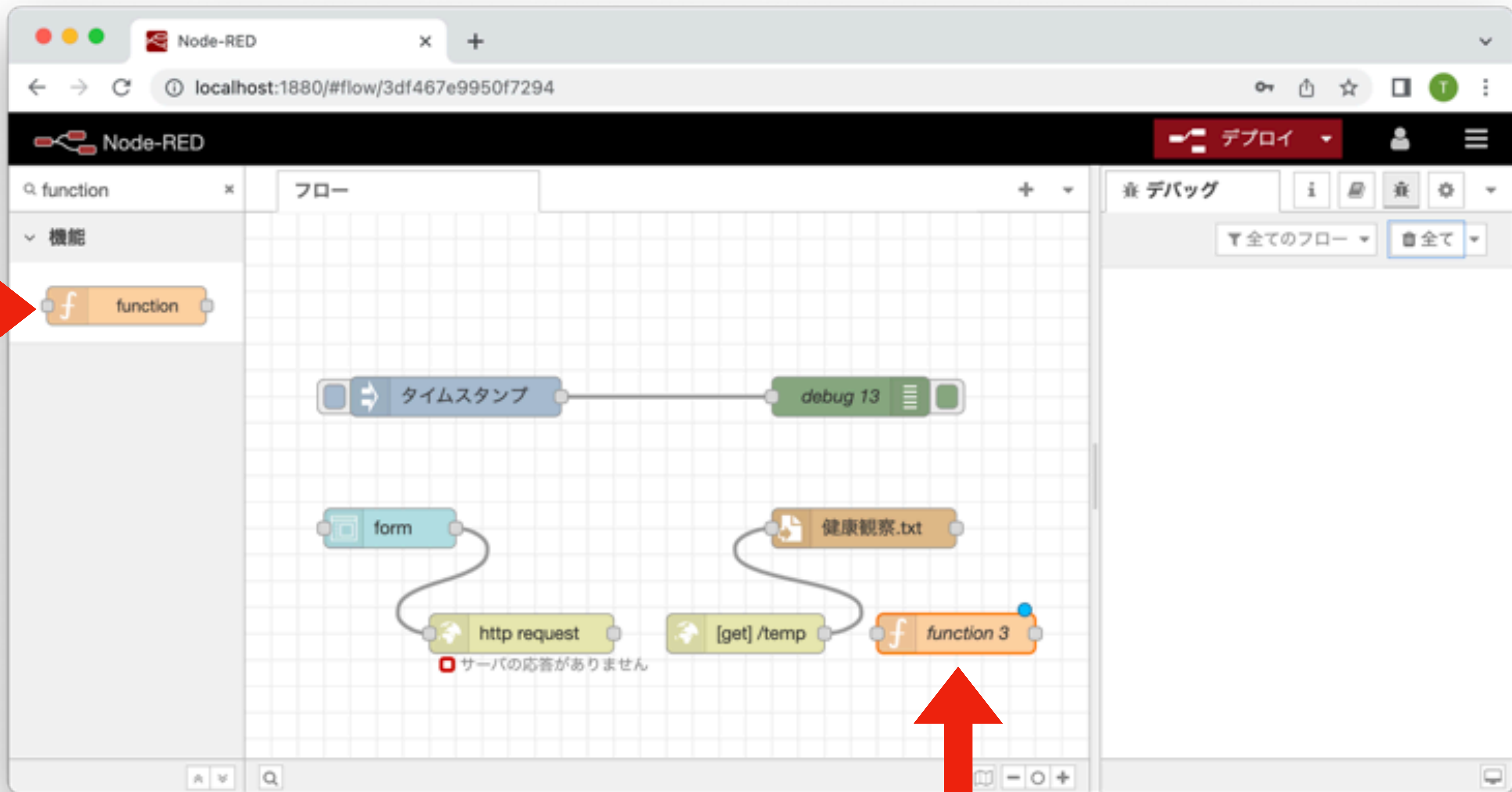
“読み込み”を選択、クリップボードから貼り付け、読み込み



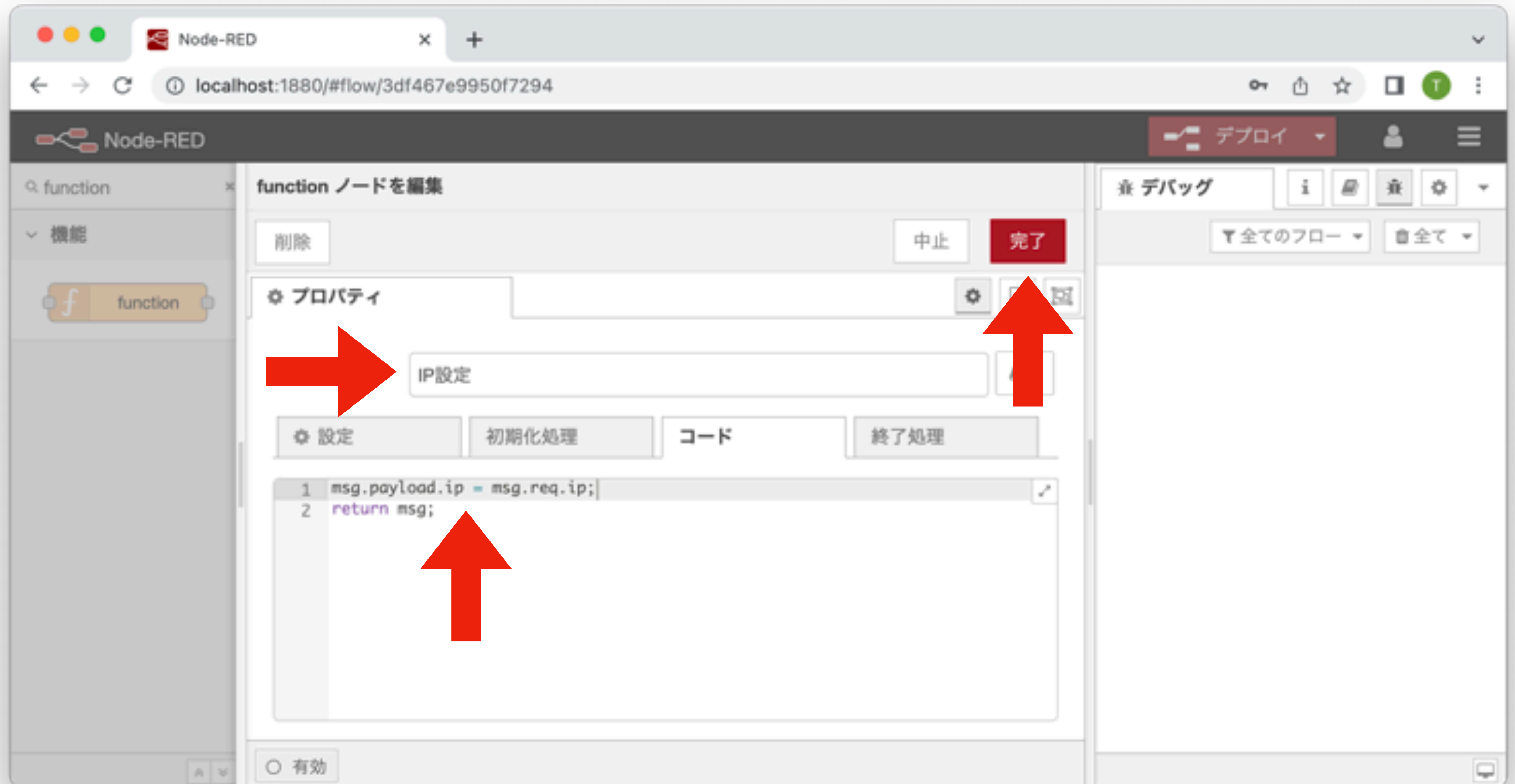
“http request” の送信先IPアドレスを変更して確認する



“function”を置き、ダブルクリック



名前を”IP設定”、`msg.payload = msg.req.ip` を追加し、完了



“[get]/temp”と”健康観察.txt”の間に”IP設定”をいれ、デプロイ

The screenshot shows the Node-RED web interface in a browser window. The address bar indicates the URL is `localhost:1880/#flow/3df467e9950f7294`. The interface is divided into several sections:

- Left Panel (Palette):** Contains a search bar with "http" entered. Under the "ネットワーク" (Network) category, the following nodes are listed: "http in", "http response", and "http request".
- Center Canvas:** Displays a flow configuration. It starts with a "タイムスタンプ" (Timestamp) node connected to a "debug 13" node. Below this, a "form" node is connected to an "http request" node. This "http request" node is connected to a "[get] /temp" node, which is then connected to an "IP設定" (IP Setting) node. A "健康観察.txt" (Health Observation.txt) node is also connected to the "IP設定" node. A red error message "サーバの応答がありません" (Server response not received) is visible below the "http request" node.
- Right Panel (Sidebar):** Contains a "デプロイ" (Deploy) button at the top. Below it, there are tabs for "情報" (Info), "フロー" (Flow), and "設定" (Settings). The "情報" tab is selected, showing a search bar "ノードを検索" (Search nodes) and a list of nodes: "集約ノード" (Aggregate node), "IPフィルタ" (IP filter), "フロー 15", "サブフロー" (Subflow), and "グローバル設定ノード" (Global settings node). The "フロー" tab is also visible, showing the current flow ID "3df467e9950f7294". At the bottom of the sidebar, there is a message: "フロー内の「ノードの設定」は、サイドバーに一覧表示できます。メニューから呼び出すか `%g` `c` を入力してください。" (The settings for nodes in the flow can be displayed in a list in the sidebar. Please enter `%g` `c` to call it from the menu).

Two red arrows are overlaid on the image: one pointing to the "情報" (Info) tab in the sidebar, and another pointing to the "IP設定" (IP Setting) node in the center canvas.

Internet3の場合

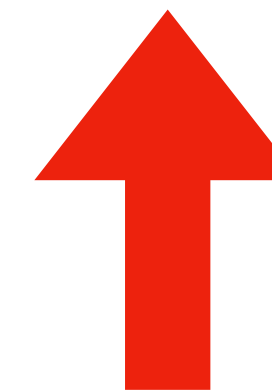
信頼できるIPアドレスが記載される

健康観察.txt



"テキストエディット"で開く

```
{"名前":"テストさんA","日付":"2022-12-30T15:00:00.000Z","体温":"35.3","ip":"????:????:????:????:????:????"}  
{"名前":"テストさんB","日付":"2022-12-30T15:00:00.000Z","体温":"36.6","ip":"????:????:????:????:????:????"}
```



switch を使ってIPアドレスによる制限を実装できる

The image shows the Node-RED web interface in a browser window. The address bar indicates the URL is `localhost:1880/#flow/2dd20844330429f5`. The main workspace contains a flow with the following nodes: a `[get] /test` node, an `IP設定` (IP Setting) function node, and an `IPフィルタ` (IP Filter) switch node. The switch node is configured with three rules: `NG`, `ローカル` (Local), and `OK!`. The left sidebar shows the '機能' (Functions) and 'dashboard' sections. The right sidebar is open to the 'switch ノードを編集' (Edit Switch Node) panel.

switch ノードを編集

削除 中止 完了

プロパティ

名前 IPフィルタ

プロパティ `msg.req.ip`

<code>==</code>	<code>::ffff:172.20.10.1</code>	→ 1	x
<code>==</code>	<code>::1</code>	→ 2	x
	その他	→ 3	x

+ 追加

創ろう！

安心便利なインターネット社会

ディスカッション

小学校の体育館の鍵を毎回公民館まで取りにきてもらっている

→ 遠隔カメラと遠隔鍵解除にする

→ 個人認証したアプリで解錠できるようにする

行政管理の人口や橋梁のデータを手間なくオープンデータにしたい

→ システムから自動連携する（公開できないデータを隠す）

庁内の他の課が管理するデータの更新をすばやく知りたい

→ ファイルの更新をチェックして、ダッシュボード上でグラフ化する