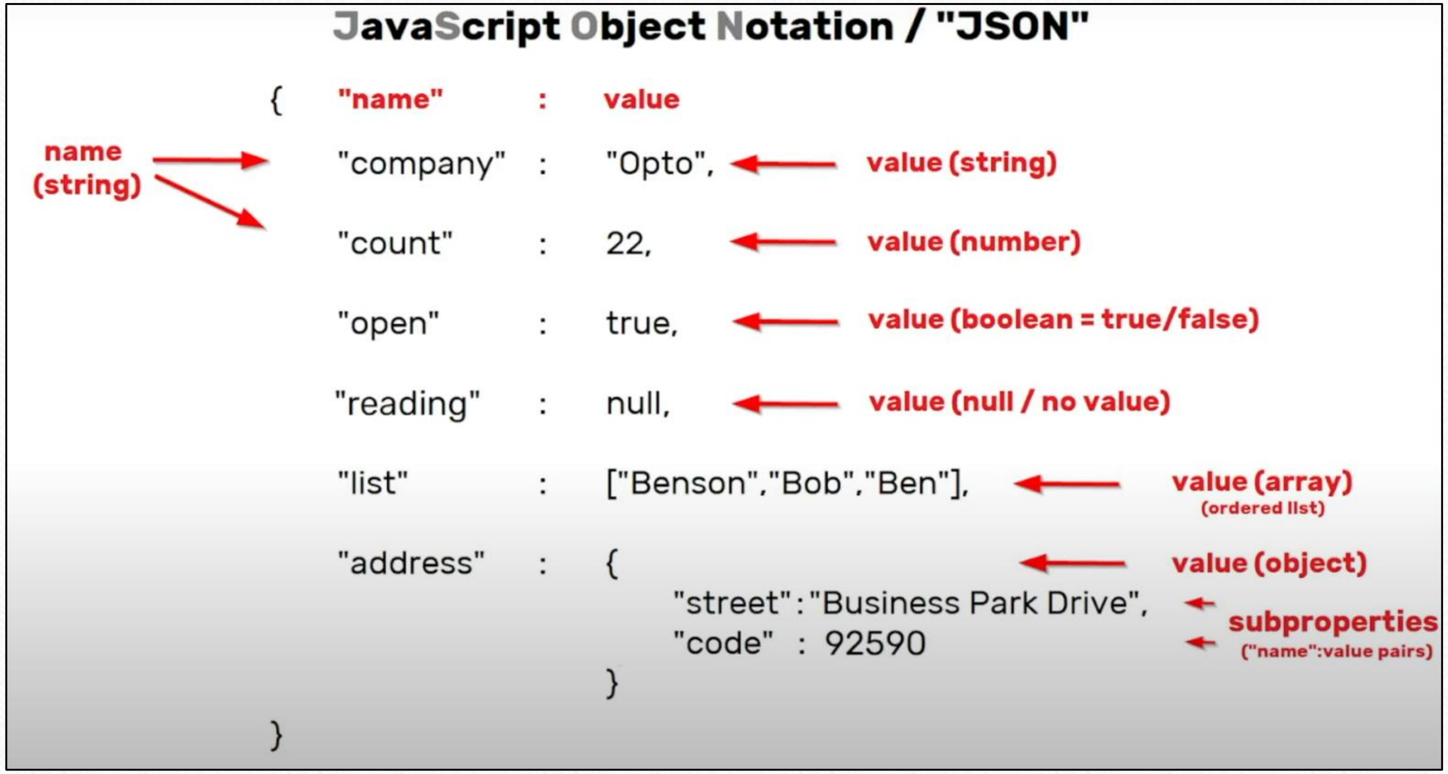
# 物聯網實務

(四)

廖裕評

# {key:value}



JavaScript Object Notation / "JSON"

```
{  "name"      :  value
   "company"   :  "Opto",
   "count"     :  22,
   "open"      :  true,
   "reading"   :  null,
   "list"      :  ["Benson","Bob","Ben"],
   "address"   :  {
                      "street" : "Business Park Drive",
                      "code"   :  92590
                  }
}
```

name (string)

value (string)

value (number)

value (boolean = true/false)

value (null / no value)

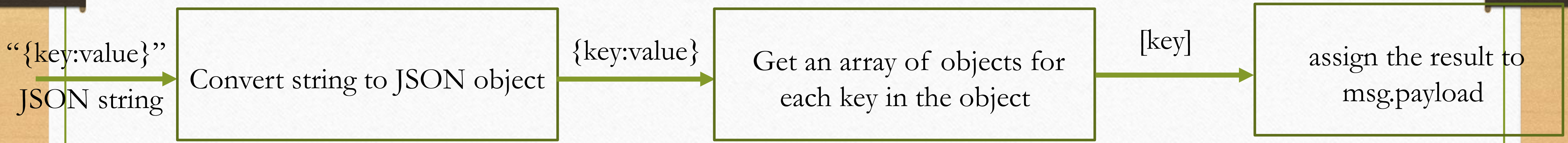
value (array)
(ordered list)

value (object)

subproperties
("name":value pairs)

https://www.youtube.com/watch?v=pbzwYKbGPC4

# Processing Data



"{key:value}"

JSON string

Convert string to JSON object

{key:value}

Get an array of objects for each key in the object

[key]

assign the result to msg.payload

# JSON.parse()

- A common use of JSON is to exchange data to/from a web server.
- When receiving data from a web server, the data is always a string.
- Parse the data with JSON.parse(), and the data becomes a JavaScript object.

```
'{"name":"John", "age":30, "city":"New York"}'
```

↓

```
{"name":"John", "age":30, "city":"New York"}
```

# Object.keys()

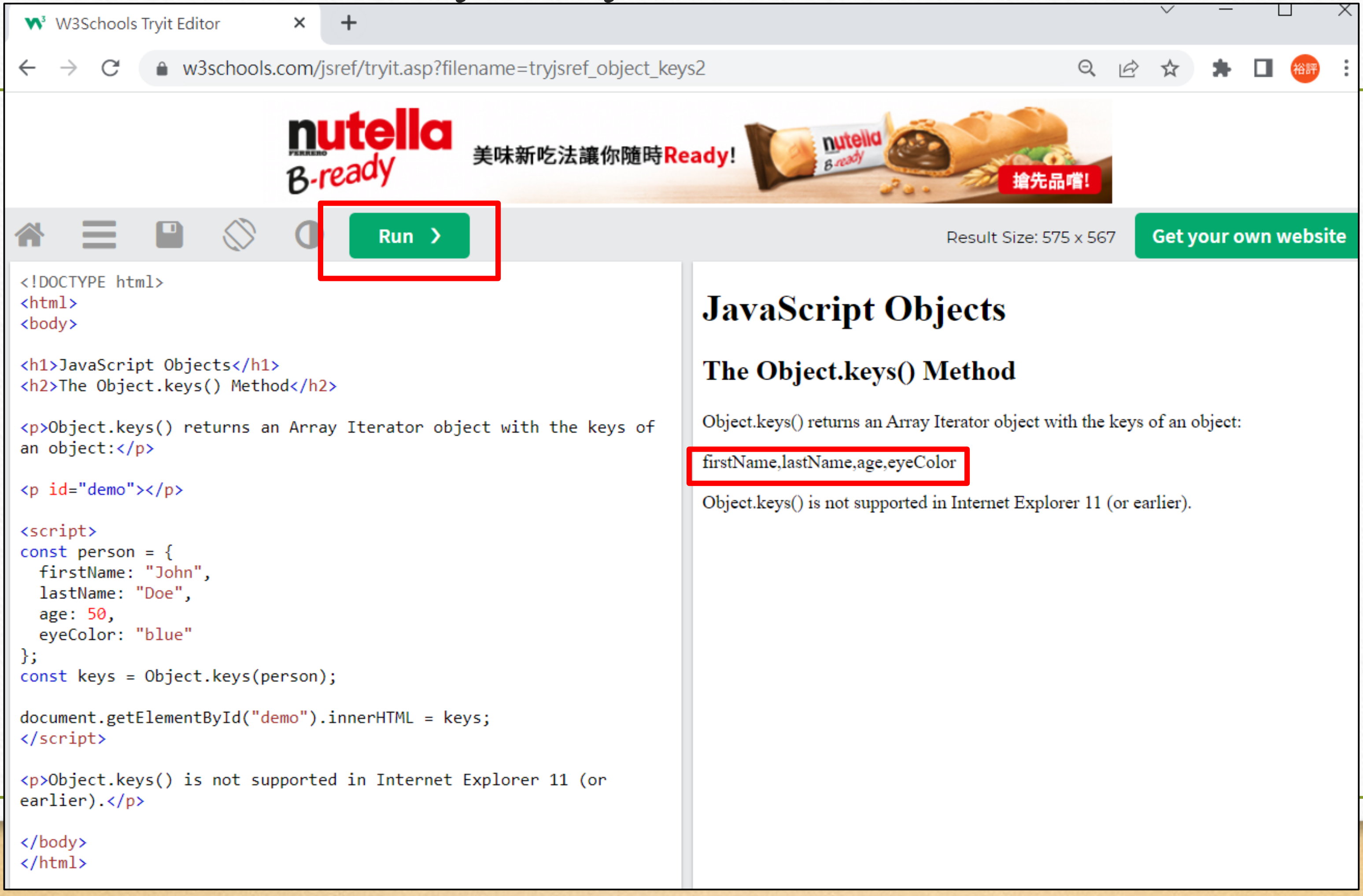
- The Object.keys() method returns an Array Iterator object with the keys of an object.
- The Object.keys() method does not change the original object.

```
const person = {
  firstName: "John",
  lastName: "Doe",
  age: 50,
  eyeColor: "blue"
};
const keys = Object.keys(person);
```

→ Keys=[ ??????????????????? ]

https://www.w3schools.com/jsref/jsref_object_keys.asp https://www.w3schools.com/jsref/tryit.asp?filename=tryjsref_object_keys2

# Try it yourself



```
<!DOCTYPE html>
<html>
<body>

<h1>JavaScript Objects</h1>
<h2>The Object.keys() Method</h2>

<p>Object.keys() returns an Array Iterator object with the keys of
an object:</p>

<p id="demo"></p>

<script>
const person = {
  firstName: "John",
  lastName: "Doe",
  age: 50,
  eyeColor: "blue"
};
const keys = Object.keys(person);

document.getElementById("demo").innerHTML = keys;
</script>

<p>Object.keys() is not supported in Internet Explorer 11 (or
earlier).</p>

</body>
</html>
```

## JavaScript Objects

### The Object.keys() Method

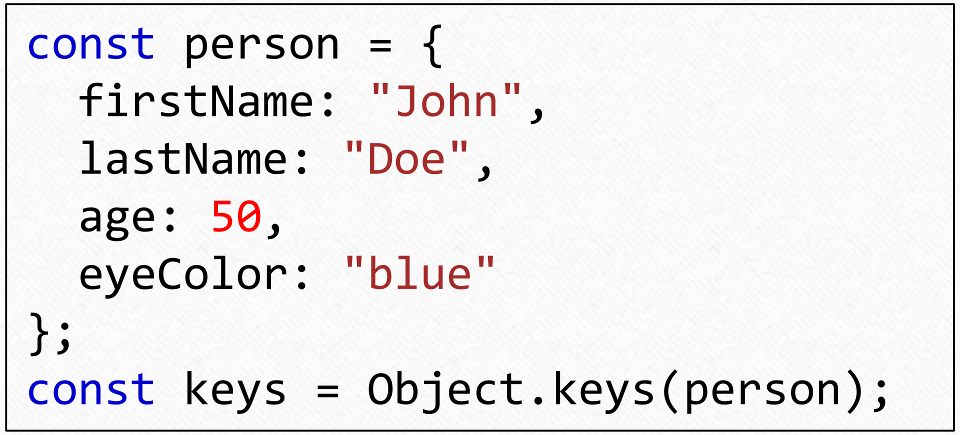
Object.keys() returns an Array Iterator object with the keys of an object:

firstName,lastName,age,eyeColor

Object.keys() is not supported in Internet Explorer 11 (or earlier).

# Object.keys()

- The Object.keys() method returns an Array Iterator object with the keys of an object.
- The Object.keys() method does not change the original object.



```
const person = {
  firstName: "John",
  lastName: "Doe",
  age: 50,
  eyeColor: "blue"
};
const keys = Object.keys(person);
```

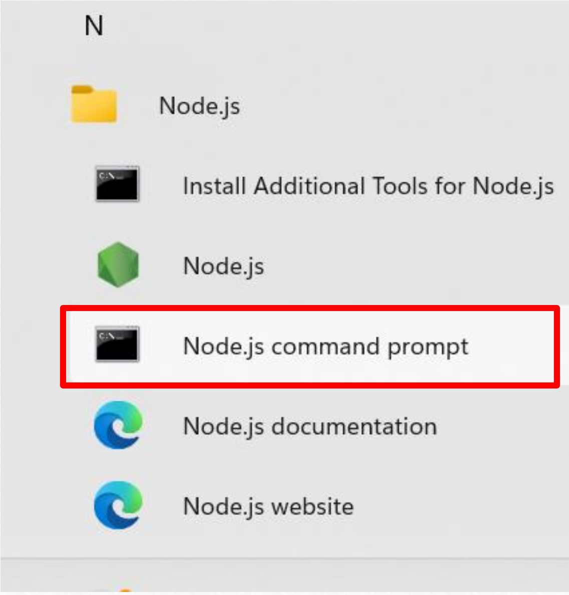
[firstName,lastName,age,eyeColor ]

https://www.w3schools.com/jsref/jsref_object_keys.asp

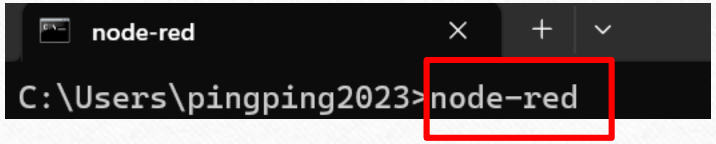
# Exercise 4-1

- Test Object.keys() in Node-RED

Step 1 :

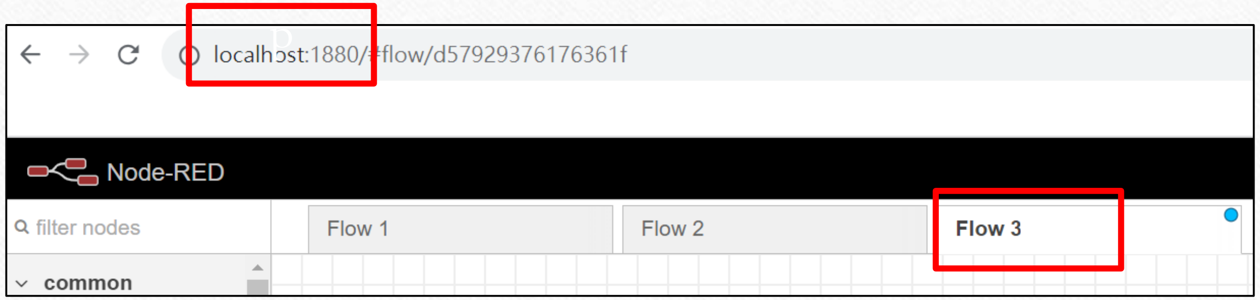


Step 2 :

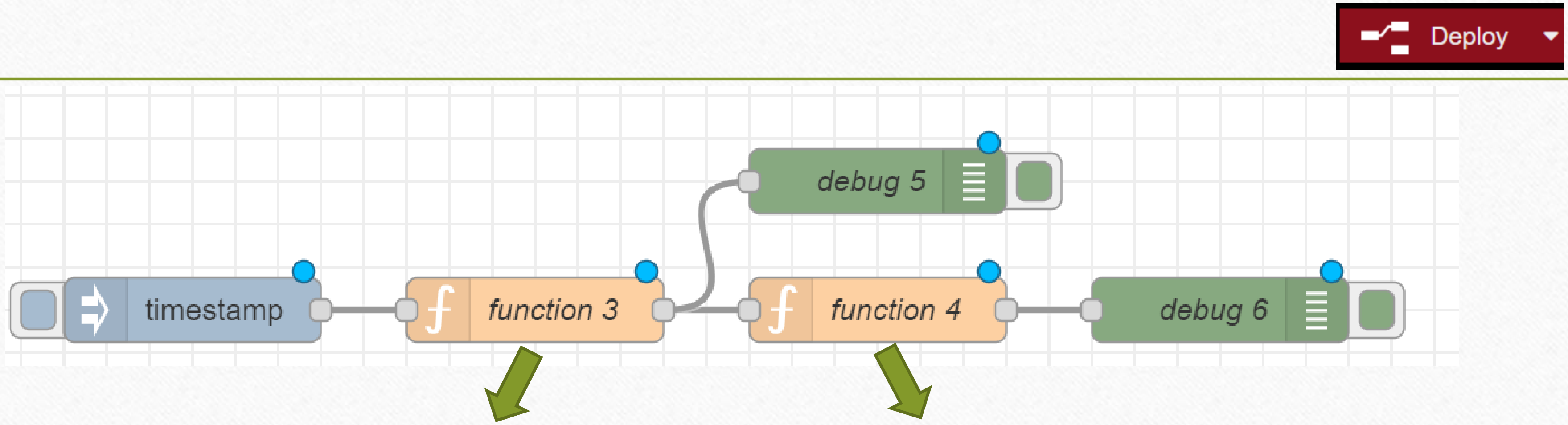


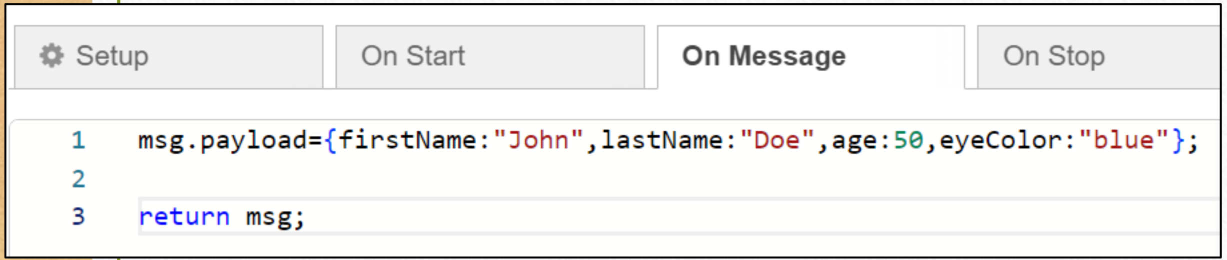
```
C:\Users\pingping2023>node-red
```

Step 3 :

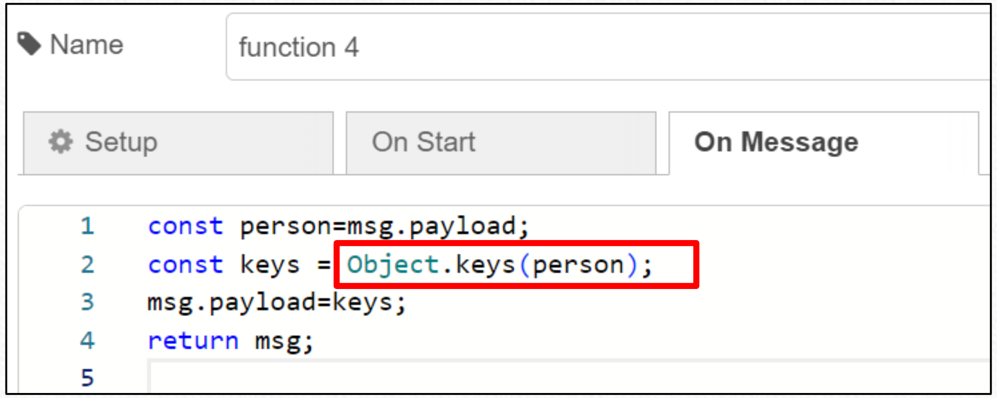


# Test Object.keys() in Node-RED





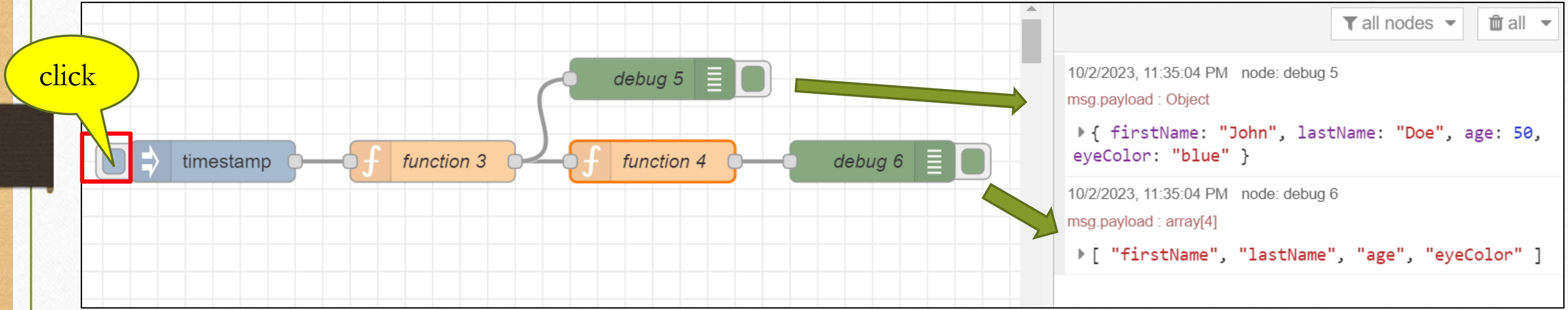
```
msg.payload={firstName:"John",lastName:"Doe",age:50,eyeColor:"blue"};

return msg;
```



Name function 4

```
const person=msg.payload;
const keys = Object.keys(person);
msg.payload=keys;
return msg;

```

# Trigger



```
10/2/2023, 11:35:04 PM   node: debug 5
msg.payload : Object
▸{ firstName: "John", lastName: "Doe", age: 50,
eyeColor: "blue" }
10/2/2023, 11:35:04 PM   node: debug 6
msg.payload : array[4]
▸[ "firstName", "lastName", "age", "eyeColor" ]
```

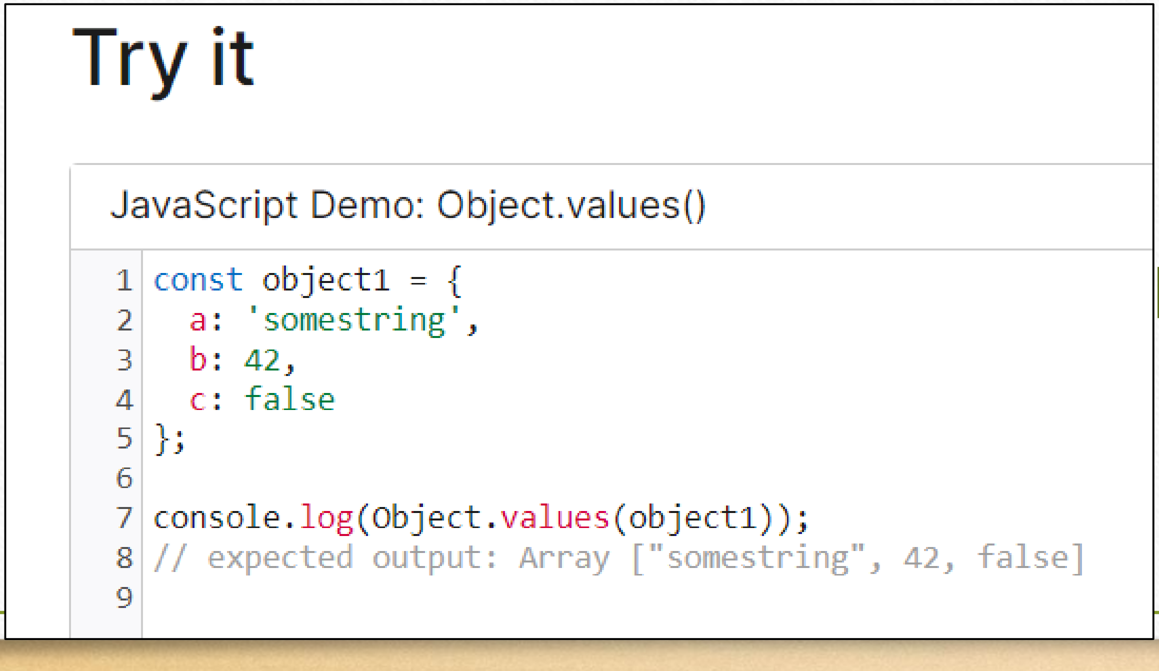
Object

```
{
firstName:"John",
lastName:"Doe",
age:50,
eyeColor:"blue“
}
```

→ Object.keys( Object ) → ["firstName", "lastName", "age", "eyeColor"]

# Object.values()

- The **Object.values()** method returns an array of a given object's own enumerable property values



## Try it

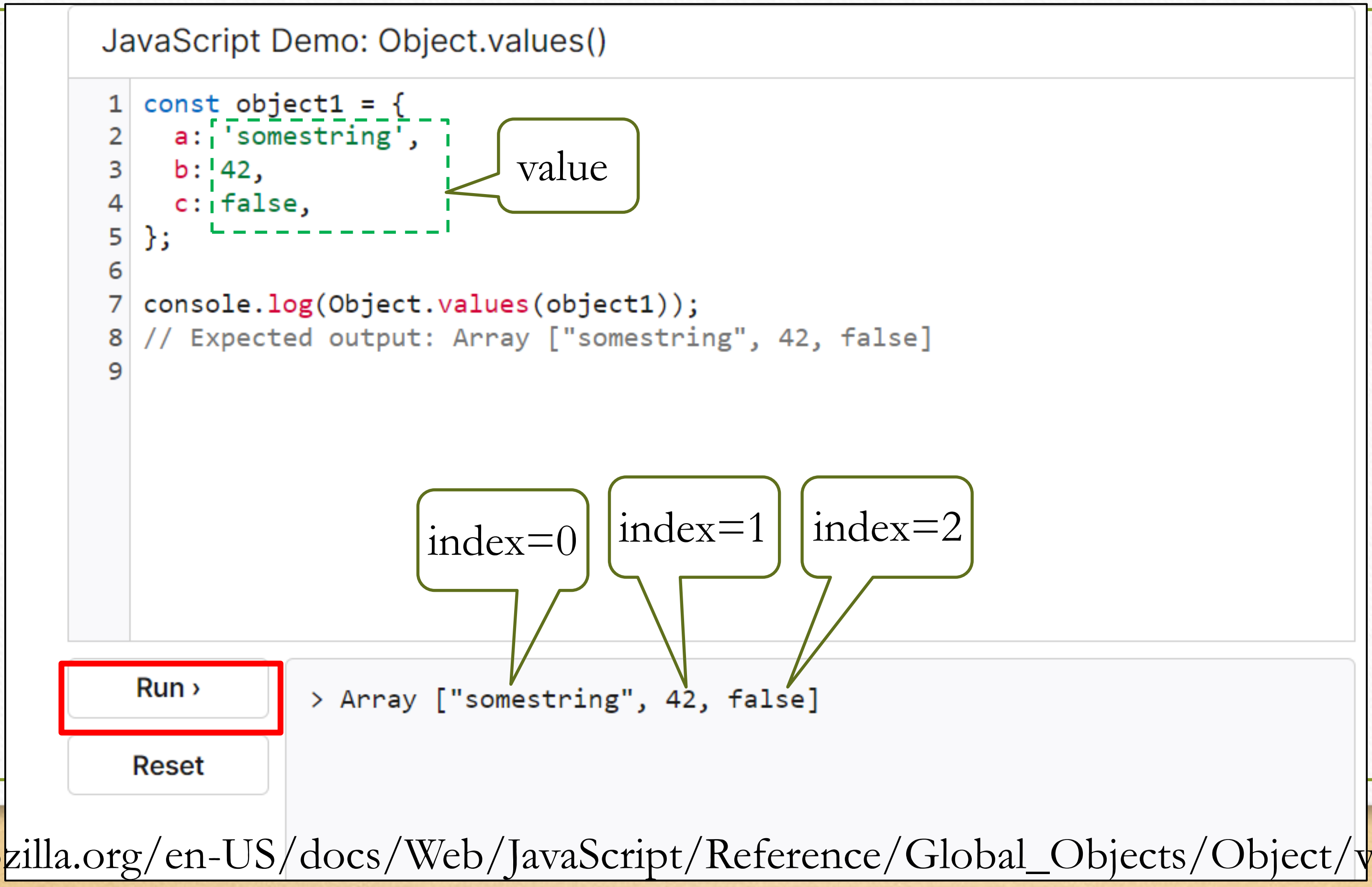
JavaScript Demo: Object.values()

```
const object1 = {
  a: 'somestring',
  b: 42,
  c: false
};

console.log(Object.values(object1));
// expected output: Array ["somestring", 42, false]

```

[ ??????????????????? ]

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Object/values

# Object.values()



JavaScript Demo: Object.values()

```
const object1 = {
  a: 'somestring',
  b: 42,
  c: false,
};

console.log(Object.values(object1));
// Expected output: Array ["somestring", 42, false]
```

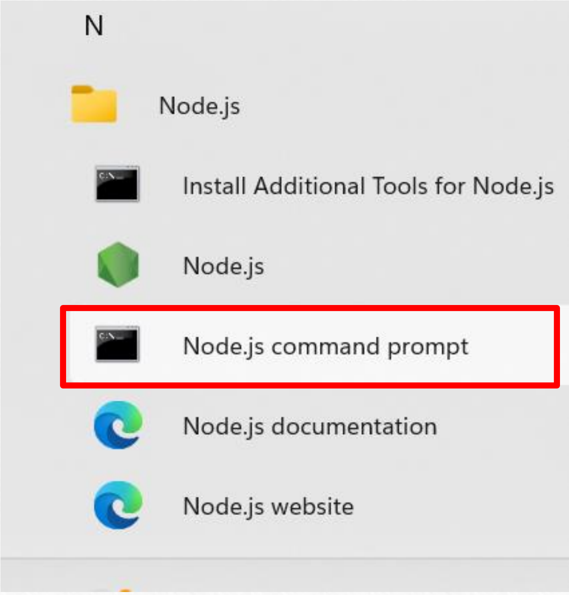
```
> Array ["somestring", 42, false]
```

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Object/values

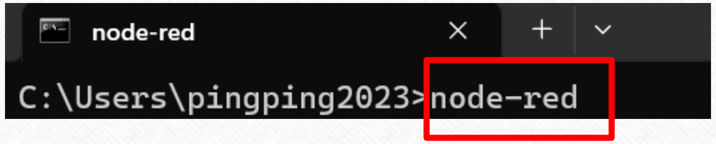
# Exercise 4-2

- Test Object.values() in Node-RED

Step 1 :

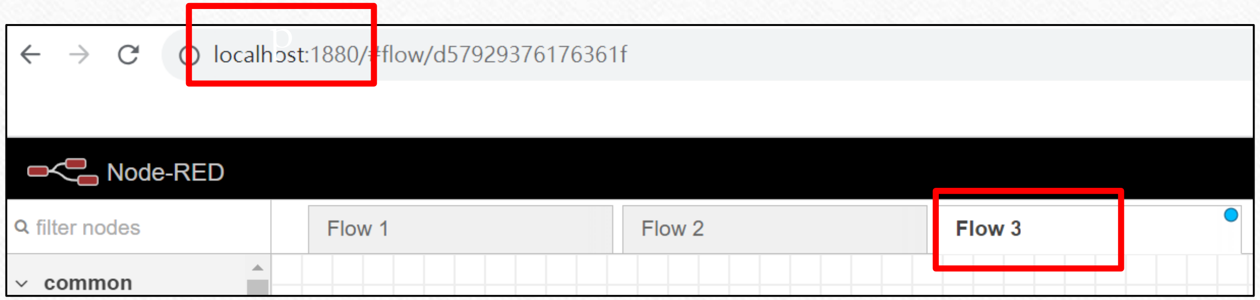


Step 2 :



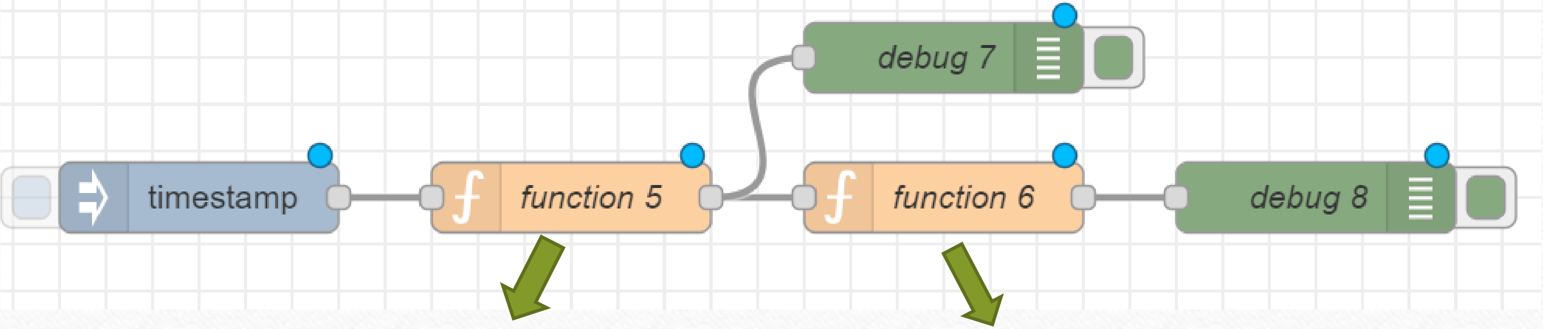
```
C:\Users\pingping2023>node-red
```

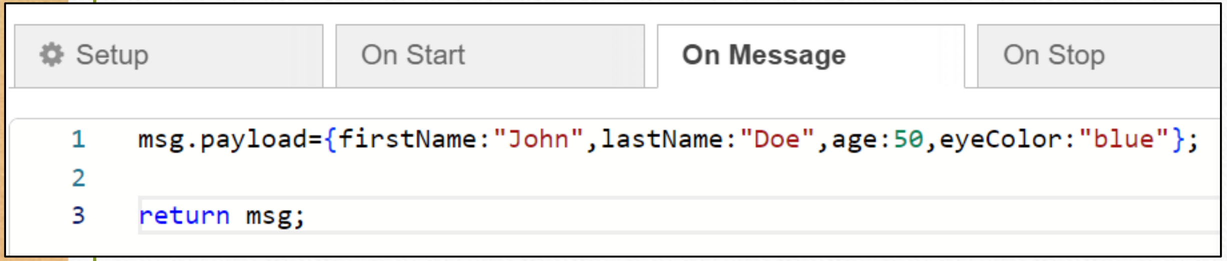
Step 3 :



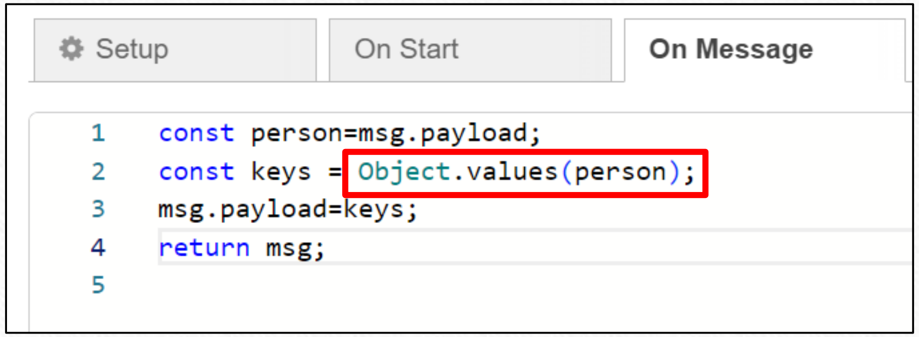
localhost:1880/#flow/d57929376176361f

# Test Object.values() in Node-RED



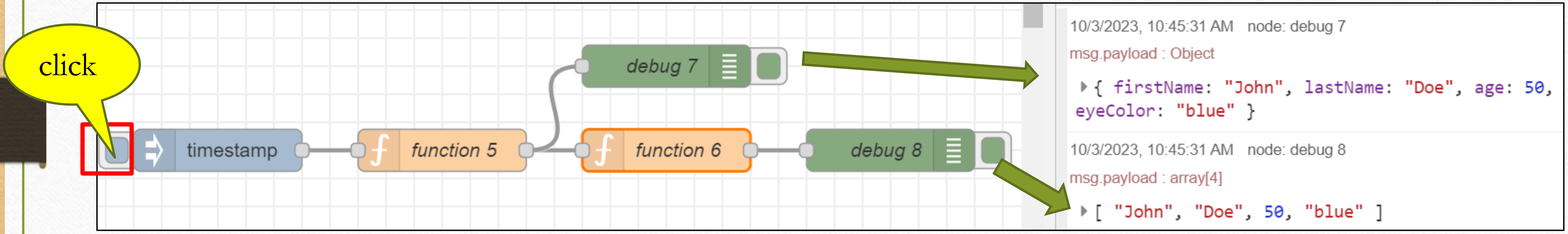


```
msg.payload={firstName:"John",lastName:"Doe",age:50,eyeColor:"blue"};

return msg;
```

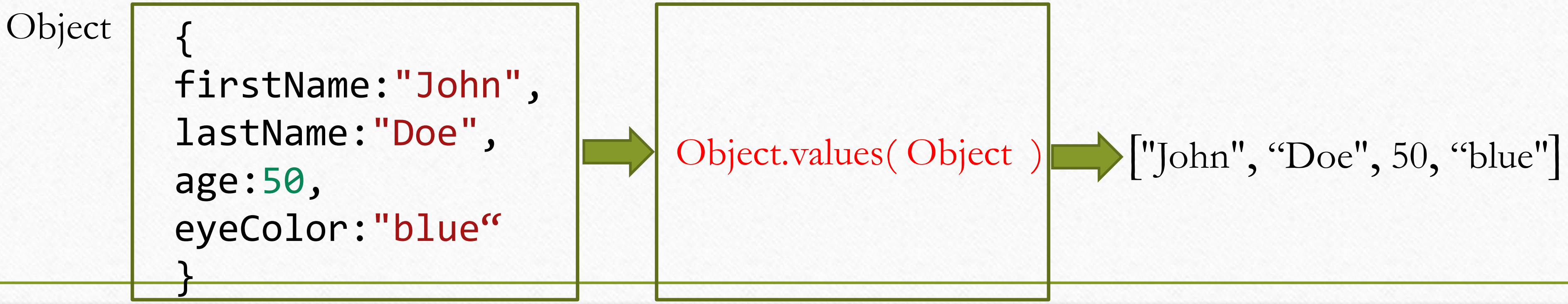


```
const person=msg.payload;
const keys = Object.values(person);
msg.payload=keys;
return msg;

```

# Trigger



```
10/3/2023, 10:45:31 AM   node: debug 7
msg.payload : Object
▸{ firstName: "John", lastName: "Doe", age: 50,
eyeColor: "blue" }

10/3/2023, 10:45:31 AM   node: debug 8
msg.payload : array[4]
▸[ "John", "Doe", 50, "blue" ]
```



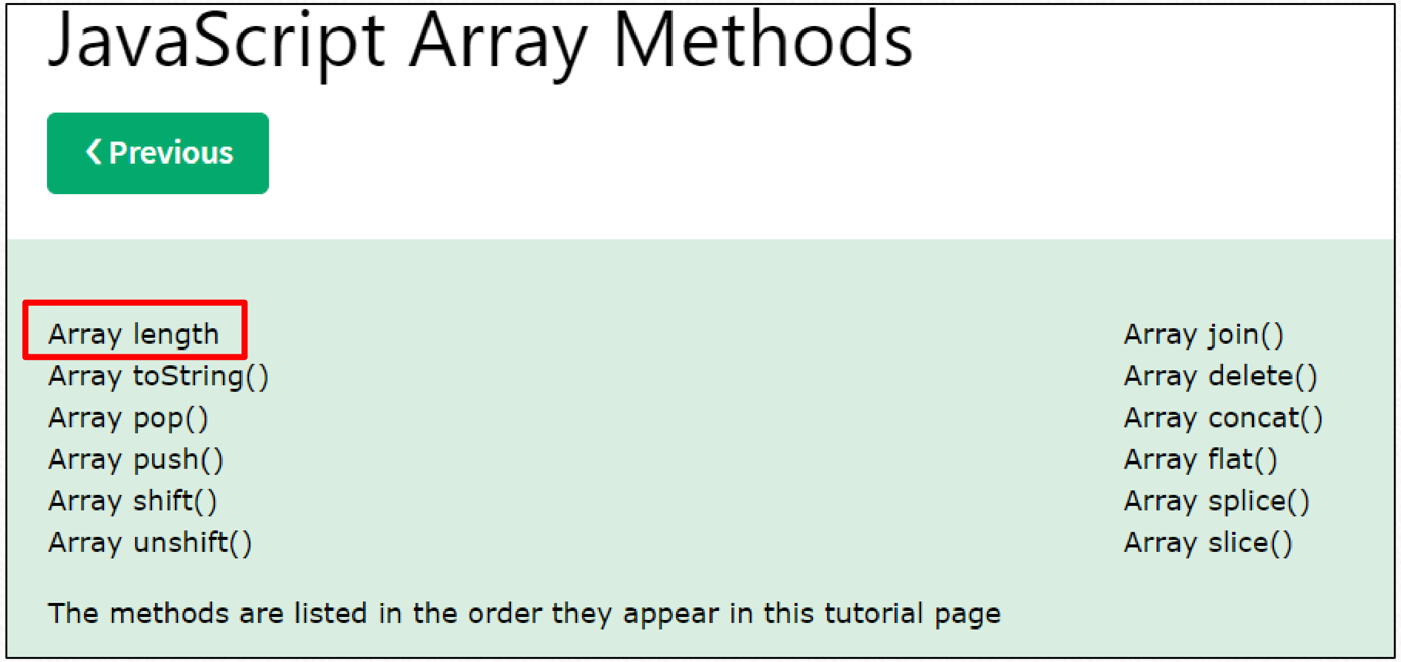
Object

```
{
firstName:"John",
lastName:"Doe",
age:50,
eyeColor:"blue“
}
```

Object.values( Object )

["John", “Doe", 50, “blue"]

# JavaScript Array Methods



## JavaScript Array Methods

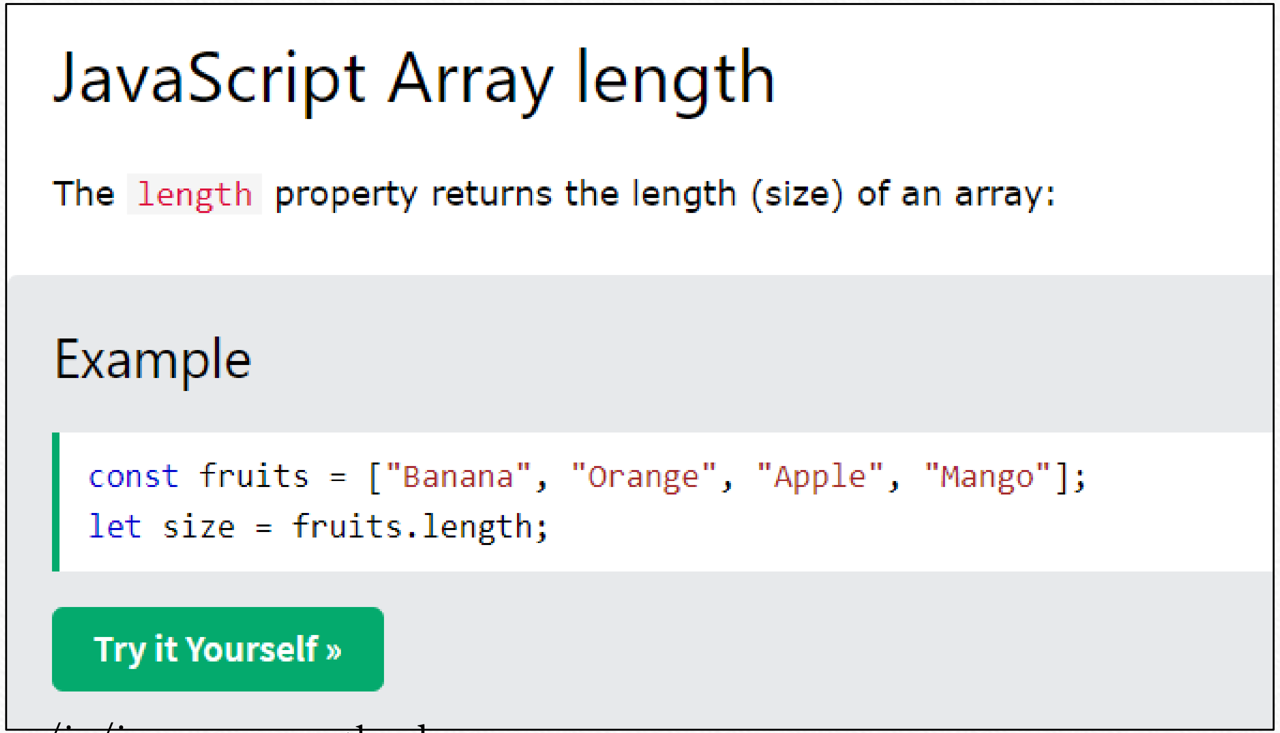
Array length
Array toString()
Array pop()
Array push()
Array shift()
Array unshift()

Array join()
Array delete()
Array concat()
Array flat()
Array splice()
Array slice()

The methods are listed in the order they appear in this tutorial page

https://www.w3schools.com/js/js_array_methods.asp

# JavaScript Array length



## JavaScript Array length

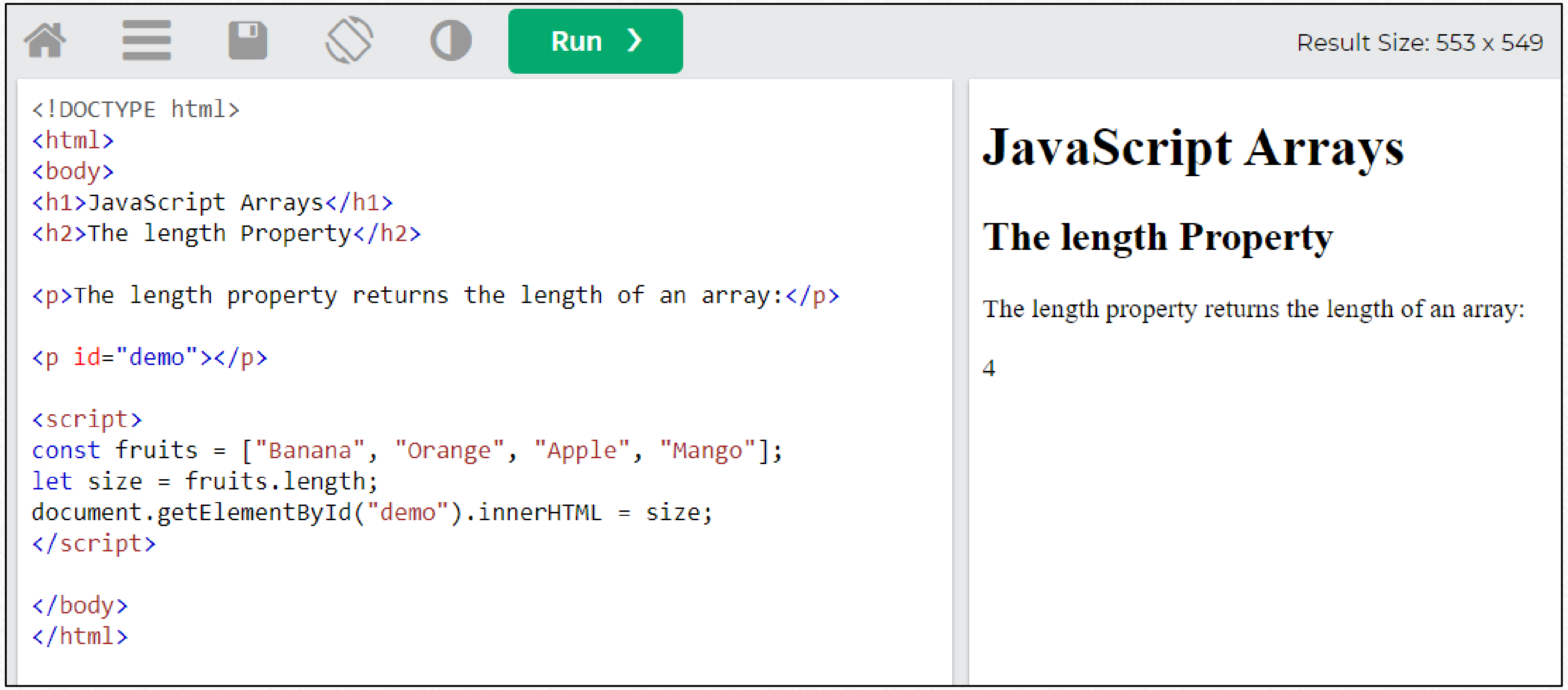
The `length` property returns the length (size) of an array:

### Example

```
const fruits = ["Banana", "Orange", "Apple", "Mango"];
let size = fruits.length;
```

https://www.w3schools.com/js/js_array_methods.asp

# JavaScript Array length



```
<!DOCTYPE html>
<html>
<body>
<h1>JavaScript Arrays</h1>
<h2>The length Property</h2>

<p>The length property returns the length of an array:</p>

<p id="demo"></p>

<script>
const fruits = ["Banana", "Orange", "Apple", "Mango"];
let size = fruits.length;
document.getElementById("demo").innerHTML = size;
</script>

</body>
</html>
```

**JavaScript Arrays**

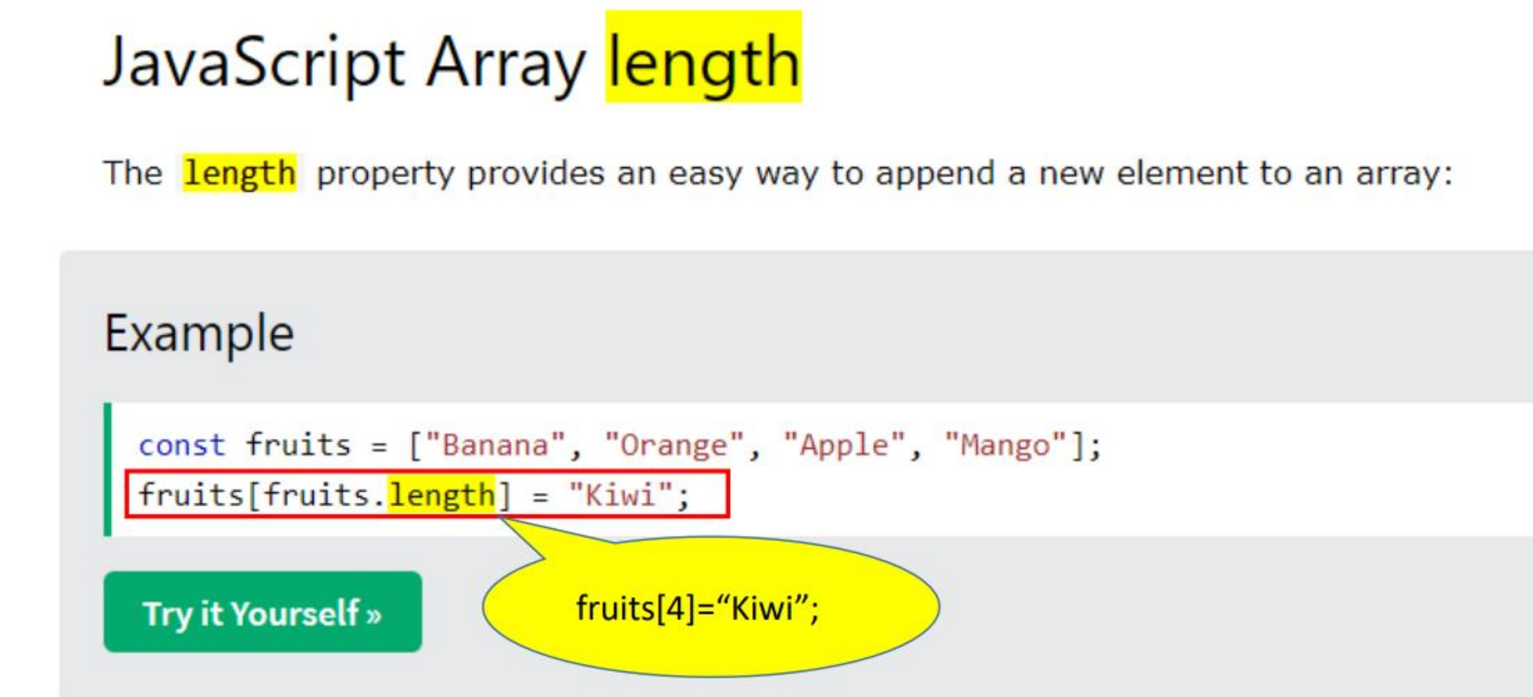
**The length Property**

The length property returns the length of an array:

4

https://www.w3schools.com/js/tryit.asp?filename=tryjs_array_length

# JavaScript Array length



## JavaScript Array length

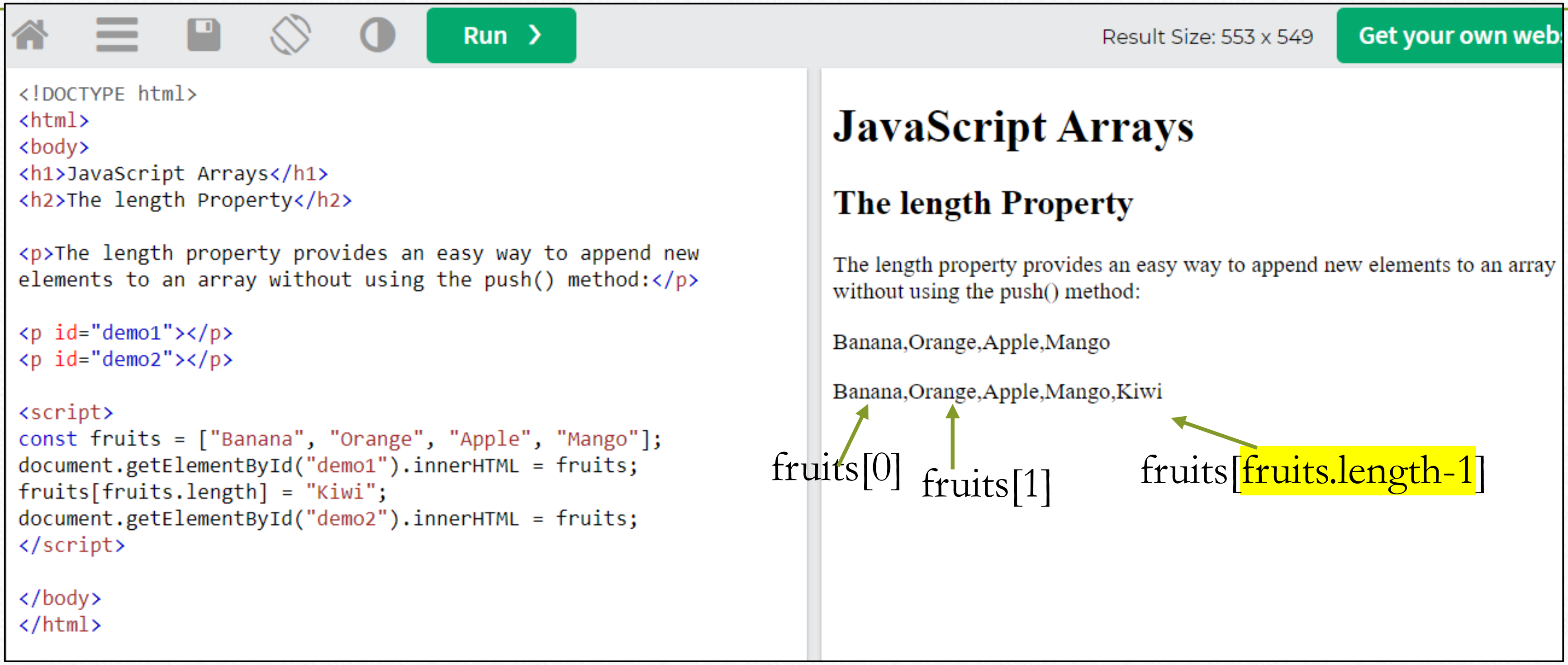
The length property provides an easy way to append a new element to an array:

### Example

```
const fruits = ["Banana", "Orange", "Apple", "Mango"];
fruits[fruits.length] = "Kiwi";
```

fruits[4]=“Kiwi”;

# JavaScript Array length



```
<!DOCTYPE html>
<html>
<body>
<h1>JavaScript Arrays</h1>
<h2>The length Property</h2>

<p>The length property provides an easy way to append new
elements to an array without using the push() method:</p>

<p id="demo1"></p>
<p id="demo2"></p>

<script>
const fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo1").innerHTML = fruits;
fruits[fruits.length] = "Kiwi";
document.getElementById("demo2").innerHTML = fruits;
</script>

</body>
</html>
```

## JavaScript Arrays

### The length Property

The length property provides an easy way to append new elements to an array without using the push() method:

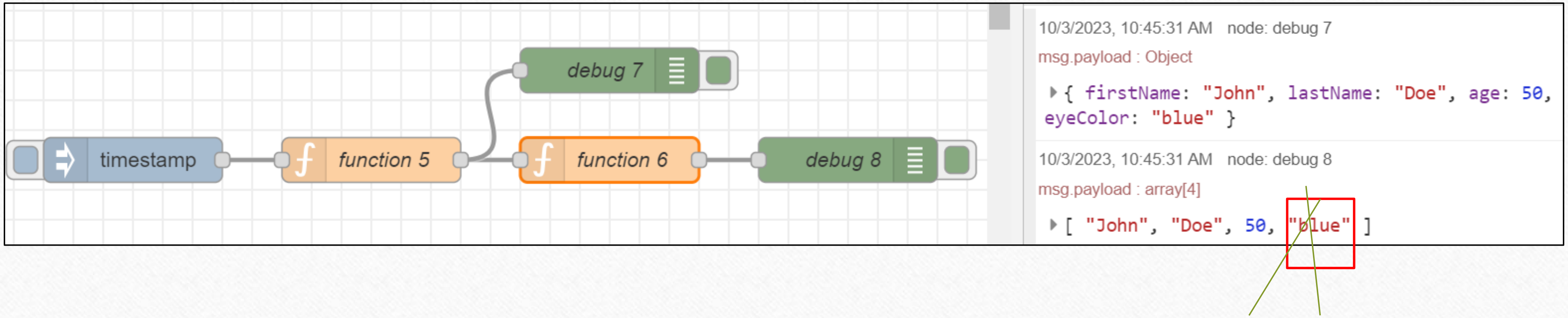
```
Banana,Orange,Apple,Mango

Banana,Orange,Apple,Mango,Kiwi
```

fruits[0] fruits[1] fruits[fruits.length-1]

https://www.w3schools.com/js/tryit.asp?filename=tryjs_array_change_add

# Homework 4-1

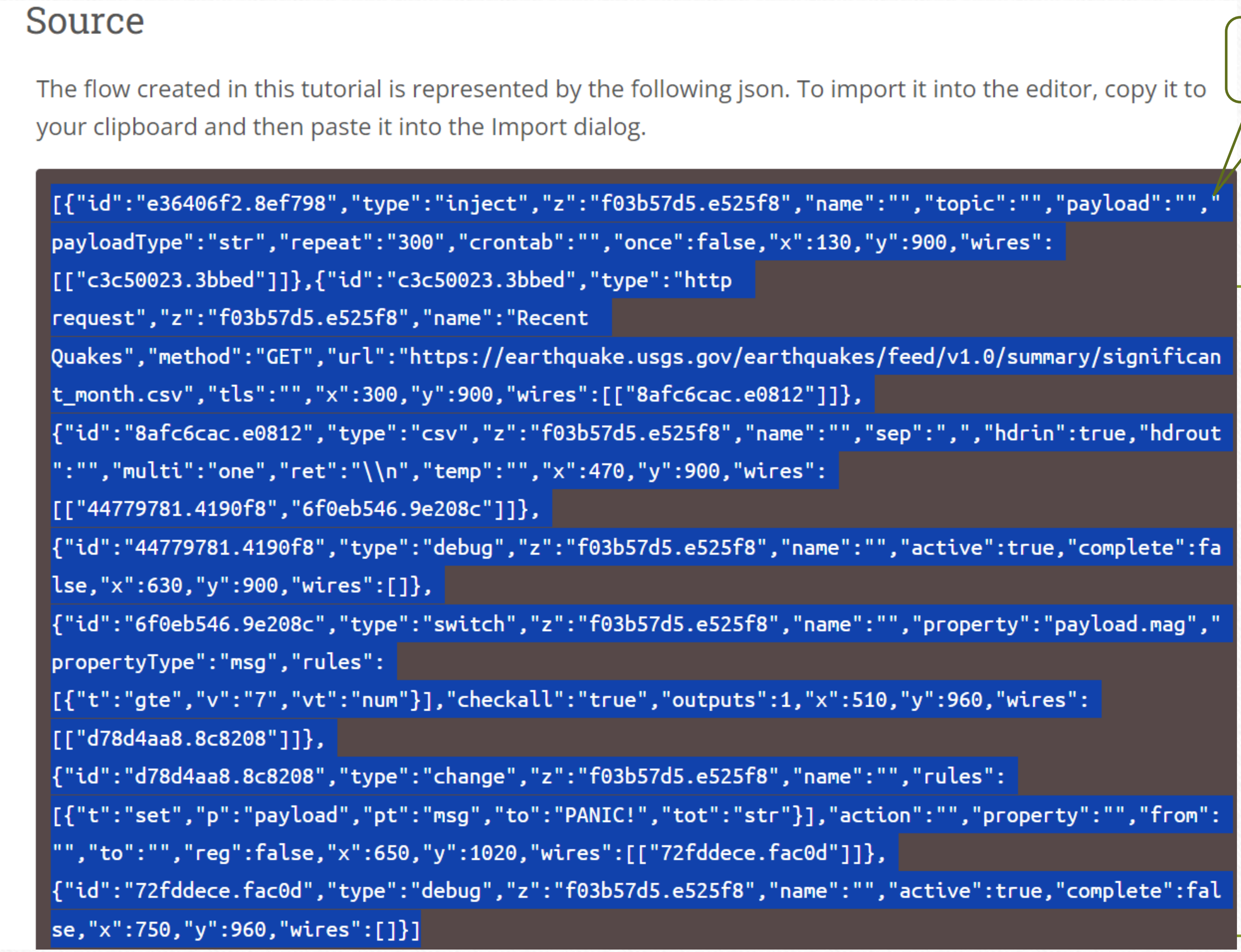
- For an object : `{firstName:"John",lastName:"Doe",age:50,eyeColor:"blue"}`
- Change the last one value to "red"



```
10/3/2023, 10:45:31 AM   node: debug 7
msg.payload : Object
▸{ firstName: "John", lastName: "Doe", age: 50,
eyeColor: "blue" }
10/3/2023, 10:45:31 AM   node: debug 8
msg.payload : array[4]
▸[ "John", "Doe", 50, "blue" ]
```

# Homework 4-2

- Print the value of mag data when the mag > 4



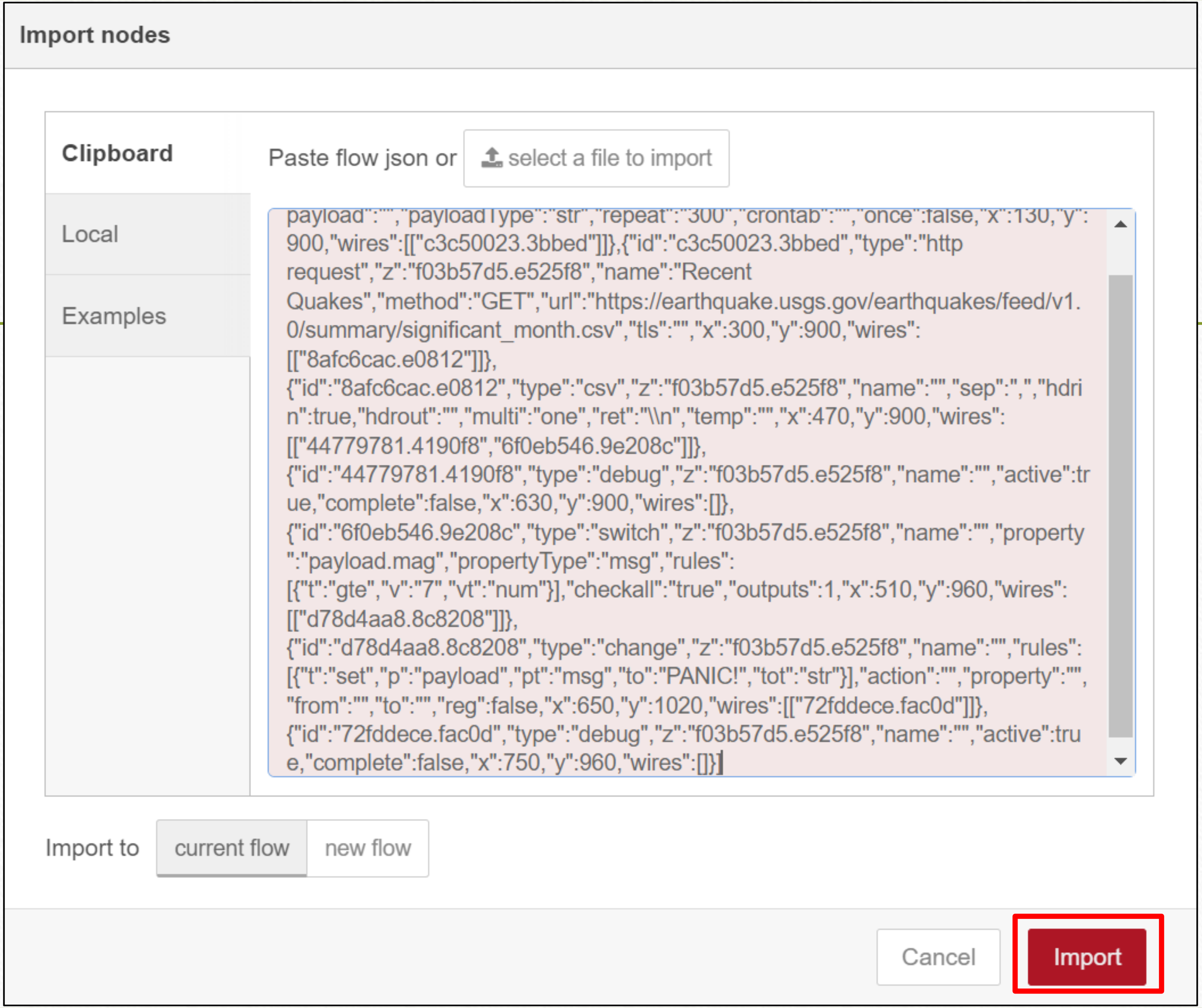
## Source

The flow created in this tutorial is represented by the following json. To import it into the editor, copy it to your clipboard and then paste it into the Import dialog.

```
[{"id":"e36406f2.8ef798","type":"inject","z":"f03b57d5.e525f8","name":"","topic":"","payload":"","
payloadType":"str","repeat":"300","crontab":"","once":false,"x":130,"y":900,"wires":
[["c3c50023.3bbed"]]},{"id":"c3c50023.3bbed","type":"http
request","z":"f03b57d5.e525f8","name":"Recent
Quakes","method":"GET","url":"https://earthquake.usgs.gov/earthquakes/feed/v1.0/summary/significan
t_month.csv","tls":"","x":300,"y":900,"wires":[["8afc6cac.e0812"]]},
{"id":"8afc6cac.e0812","type":"csv","z":"f03b57d5.e525f8","name":"","sep":",","hdrin":true,"hdrout
":"","multi":"one","ret":"\\n","temp":"","x":470,"y":900,"wires":
[["44779781.4190f8","6f0eb546.9e208c"]]},
{"id":"44779781.4190f8","type":"debug","z":"f03b57d5.e525f8","name":"","active":true,"complete":fa
lse,"x":630,"y":900,"wires":[]},
{"id":"6f0eb546.9e208c","type":"switch","z":"f03b57d5.e525f8","name":"","property":"payload.mag","
propertyType":"msg","rules":
[{"t":"gte","v":"7","vt":"num"}],"checkall":"true","outputs":1,"x":510,"y":960,"wires":
[["d78d4aa8.8c8208"]]},
{"id":"d78d4aa8.8c8208","type":"change","z":"f03b57d5.e525f8","name":"","rules":
[{"t":"set","p":"payload","pt":"msg","to":"PANIC!","tot":"str"}],"action":"","property":"","from":
"","to":"","reg":false,"x":650,"y":1020,"wires":[["72fddece.fac0d"]]},
{"id":"72fddece.fac0d","type":"debug","z":"f03b57d5.e525f8","name":"","active":true,"complete":fal
se,"x":750,"y":960,"wires":[]}]
```

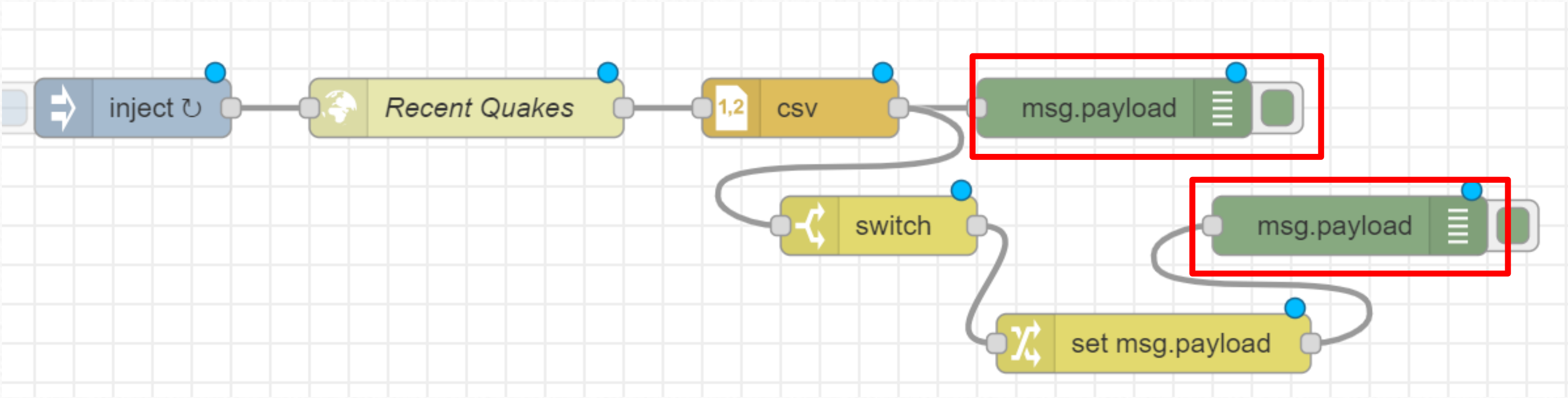
copy

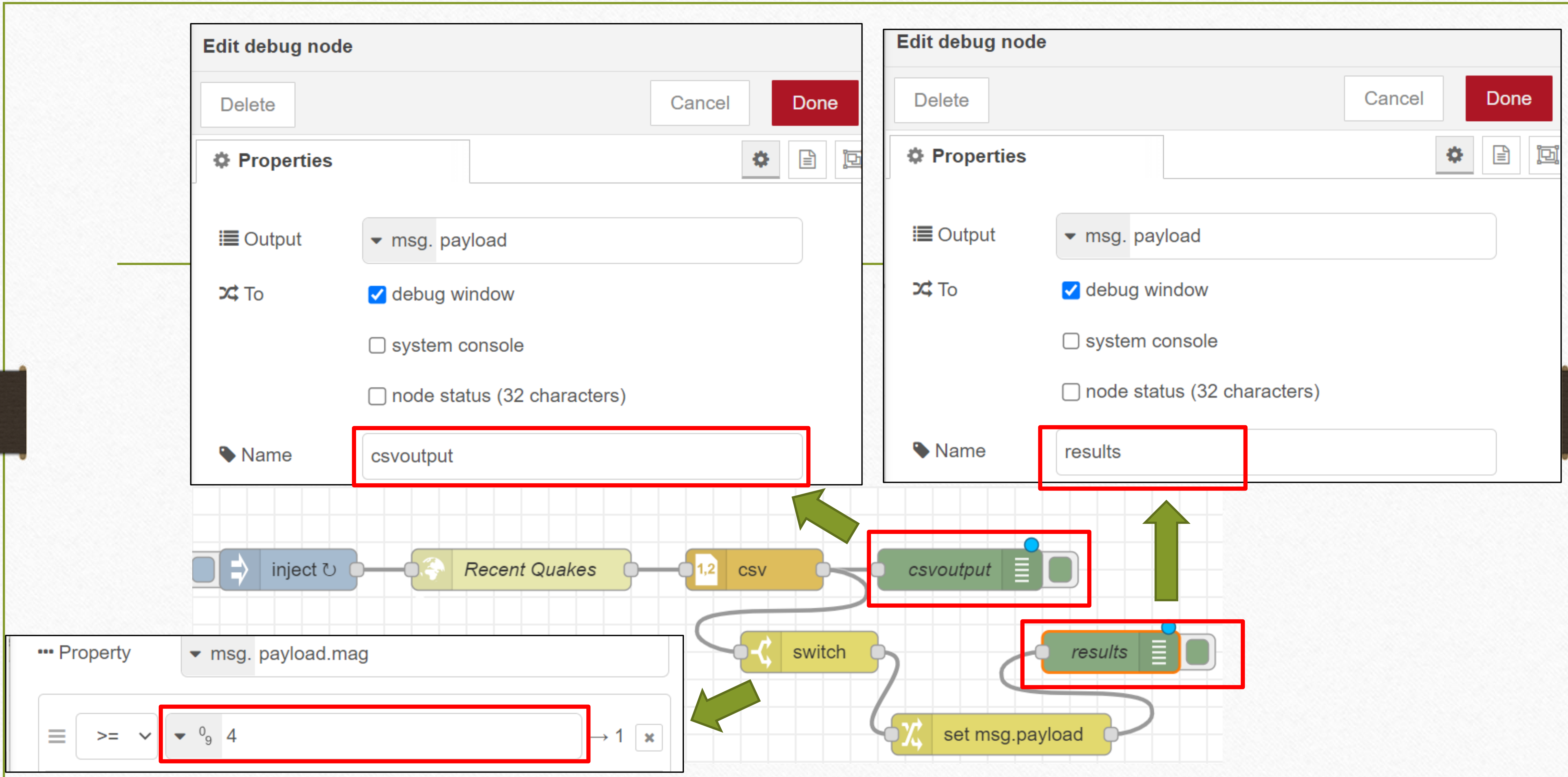
https://nodered.org/docs/tutorials/second-flow



payload":"","payloadType":"str","repeat":"300","crontab":"","once":false,"x":130,"y":900,"wires":[["c3c50023.3bbed"]]},{"id":"c3c50023.3bbed","type":"http request","z":"f03b57d5.e525f8","name":"Recent Quakes","method":"GET","url":"https://earthquake.usgs.gov/earthquakes/feed/v1.0/summary/significant_month.csv","tls":"","x":300,"y":900,"wires":[["8afc6cac.e0812"]]},
{"id":"8afc6cac.e0812","type":"csv","z":"f03b57d5.e525f8","name":"","sep":",","hdrin":true,"hdrout":"","multi":"one","ret":"\\n","temp":"","x":470,"y":900,"wires":[["44779781.4190f8","6f0eb546.9e208c"]]},
{"id":"44779781.4190f8","type":"debug","z":"f03b57d5.e525f8","name":"","active":true,"complete":false,"x":630,"y":900,"wires":[]},
{"id":"6f0eb546.9e208c","type":"switch","z":"f03b57d5.e525f8","name":"","property":"payload.mag","propertyType":"msg","rules":[{"t":"gte","v":"7","vt":"num"}],"checkall":"true","outputs":1,"x":510,"y":960,"wires":[["d78d4aa8.8c8208"]]},
{"id":"d78d4aa8.8c8208","type":"change","z":"f03b57d5.e525f8","name":"","rules":[{"t":"set","p":"payload","pt":"msg","to":"PANIC!","tot":"str"}],"action":"","property":"","from":"","to":"","reg":false,"x":650,"y":1020,"wires":[["72fddece.fac0d"]]},
{"id":"72fddece.fac0d","type":"debug","z":"f03b57d5.e525f8","name":"","active":true,"complete":false,"x":750,"y":960,"wires":[]}]

Deploy


inject ↻
Recent Quakes
csv
msg.payload
switch
msg.payload
set msg.payload



**Edit debug node**

Delete | Cancel | Done

Properties

Output: msg. payload

To: ☑ debug window

□ system console

□ node status (32 characters)

Name: csvoutput

**Edit debug node**

Delete | Cancel | Done

Properties

Output: msg. payload

To: ☑ debug window

□ system console

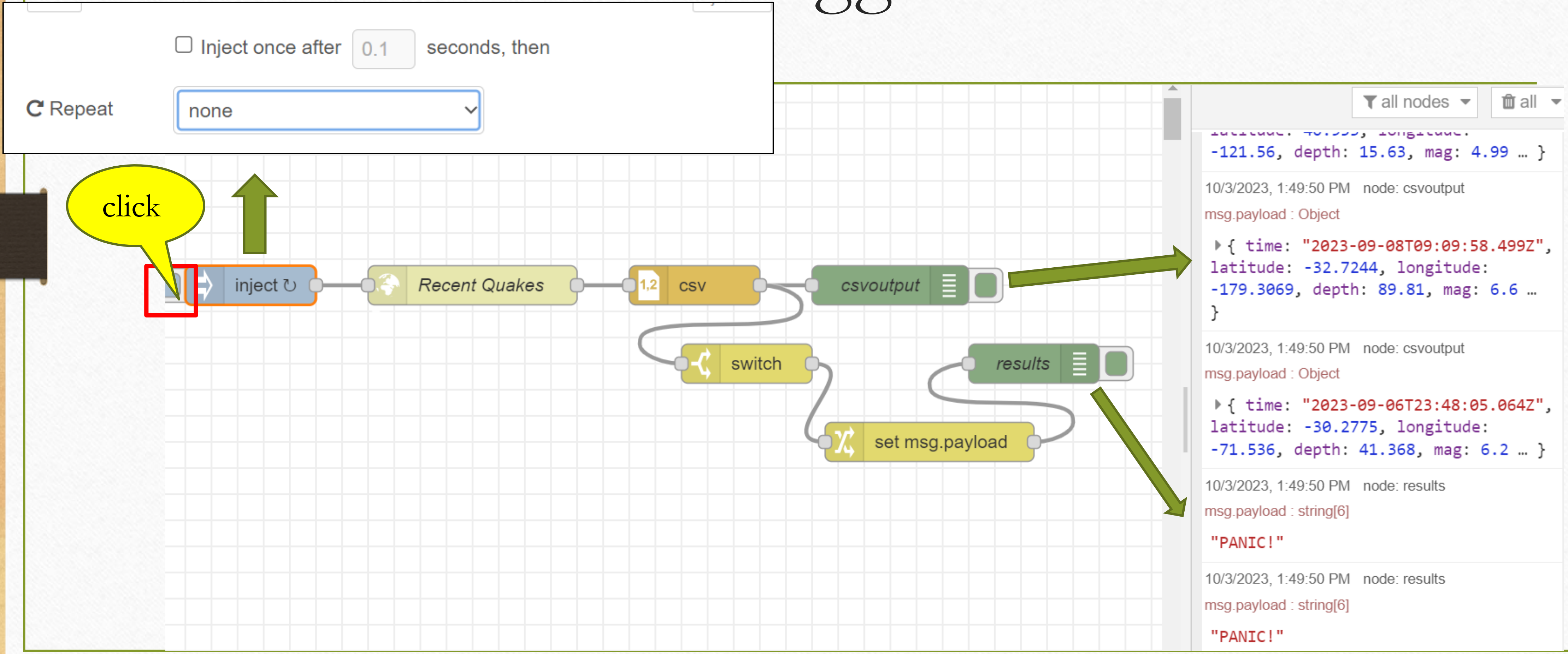
□ node status (32 characters)

Name: results

Property: msg. payload.mag

>= 4 → 1

# Trigger

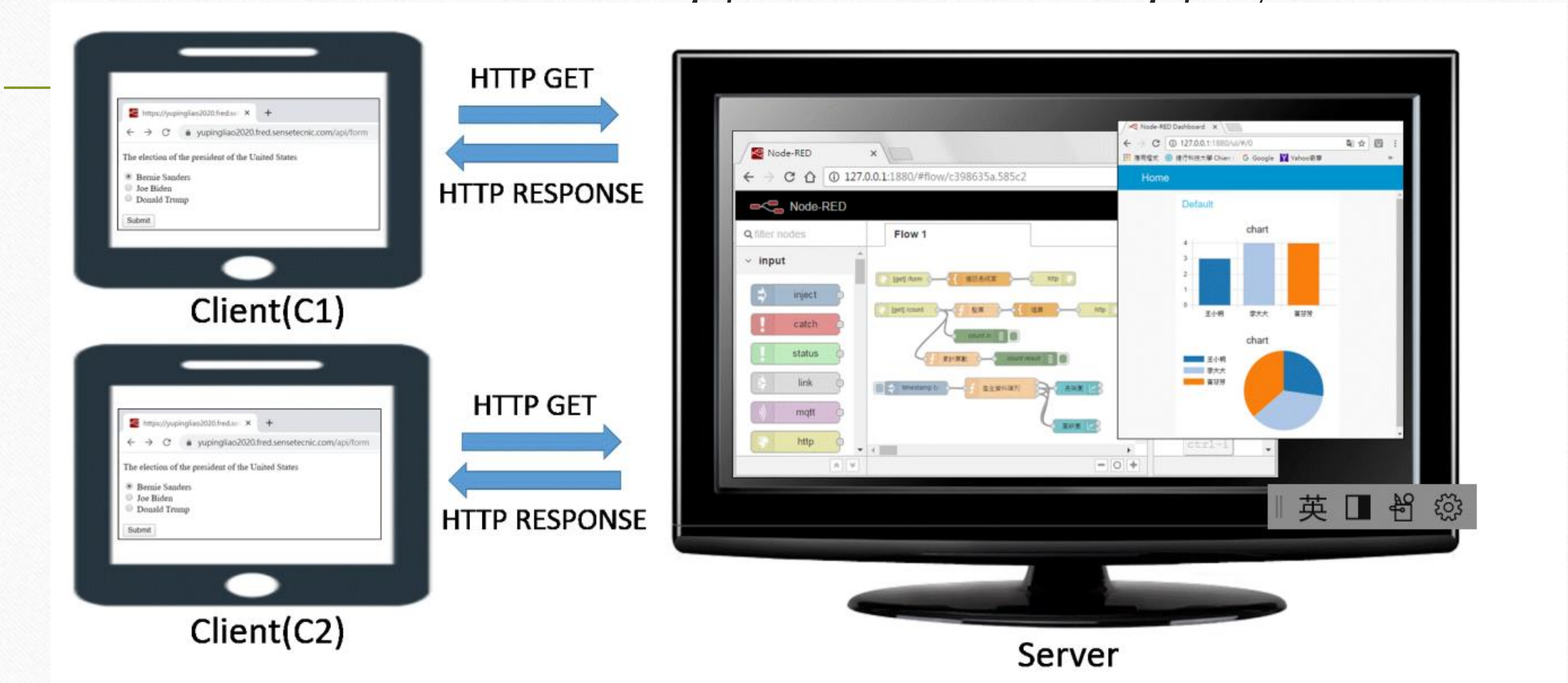


☐ Inject once after 0.1 seconds, then

Repeat: none

```
-121.56, depth: 15.63, mag: 4.99 … }

10/3/2023, 1:49:50 PM   node: csvoutput
msg.payload : Object
▸{ time: "2023-09-08T09:09:58.499Z",
latitude: -32.7244, longitude:
-179.3069, depth: 89.81, mag: 6.6 …
}

10/3/2023, 1:49:50 PM   node: csvoutput
msg.payload : Object
▸{ time: "2023-09-06T23:48:05.064Z",
latitude: -30.2775, longitude:
-71.536, depth: 41.368, mag: 6.2 … }

10/3/2023, 1:49:50 PM   node: results
msg.payload : string[6]
"PANIC!"

10/3/2023, 1:49:50 PM   node: results
msg.payload : string[6]
"PANIC!"
```

# Exercise 4-3 Design of a voting system



https://yupingliao2020.fred.se
yupingliao2020.fred.sensetecnic.com/api/form

The election of the president of the United States

● Bernie Sanders
○ Joe Biden
○ Donald Trump

Submit

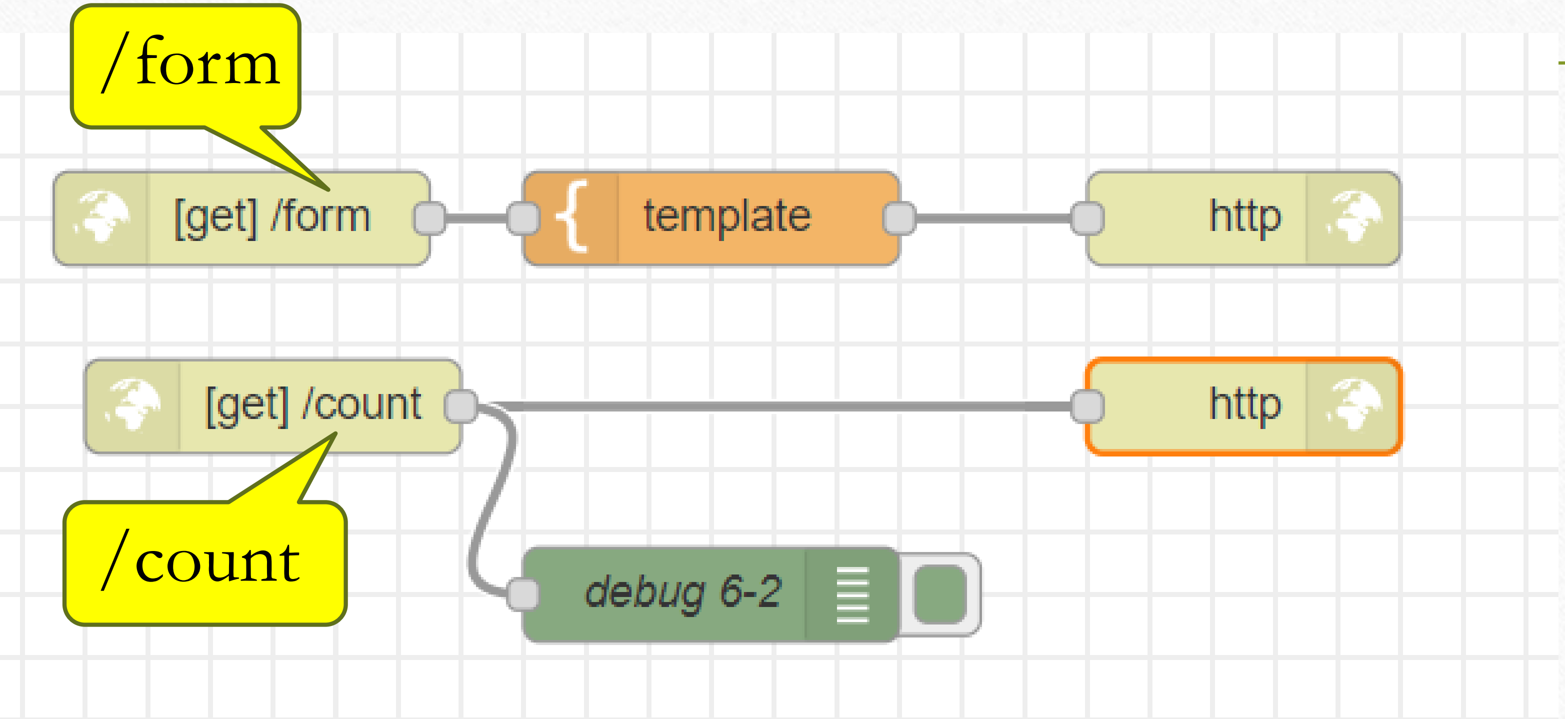
https://yupingliao2020.fred.se
yupingliao2020.fred.sensetecnic.com/api/form

The election of the president of the United States

● Bernie Sanders
○ Joe Biden
○ Donald Trump

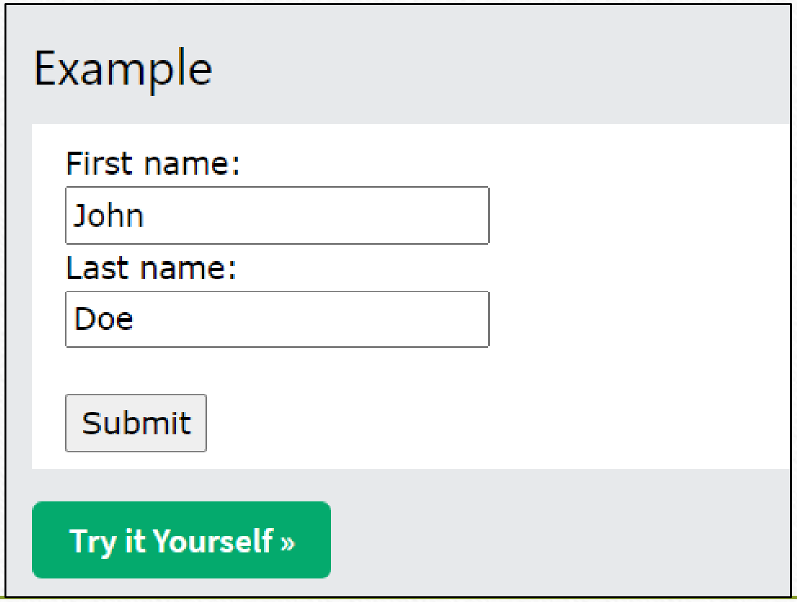
Submit

# Step 1



# HTML Forms

- An HTML form is used to collect user input. The user input is most often sent to a server for processing.



Example

First name:

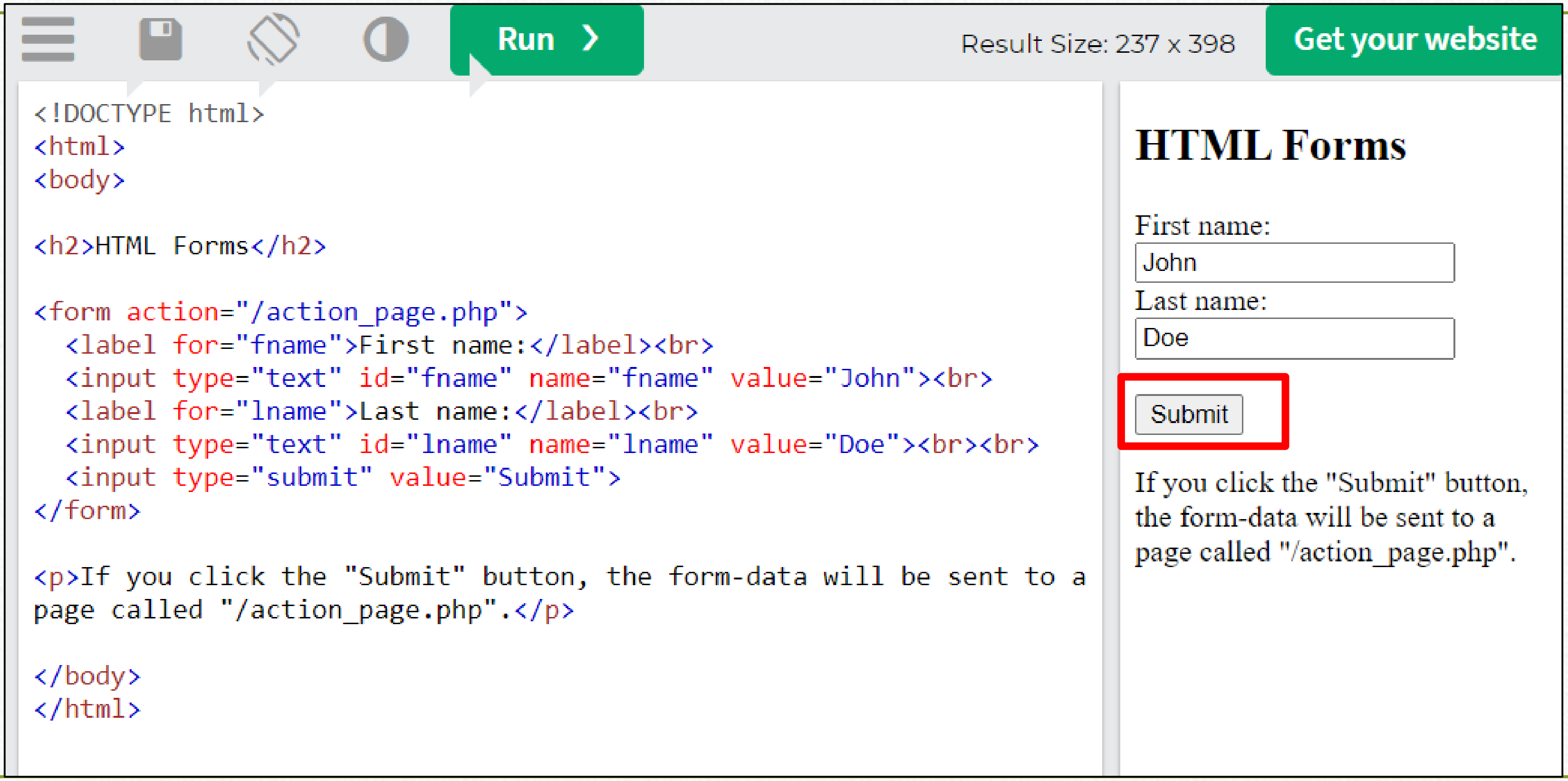
John

Last name:

Doe

https://www.w3schools.com/html/html_forms.asp

# Try it yourself



```
<!DOCTYPE html>
<html>
<body>

<h2>HTML Forms</h2>

<form action="/action_page.php">
  <label for="fname">First name:</label><br>
  <input type="text" id="fname" name="fname" value="John"><br>
  <label for="lname">Last name:</label><br>
  <input type="text" id="lname" name="lname" value="Doe"><br><br>
  <input type="submit" value="Submit">
</form>

<p>If you click the "Submit" button, the form-data will be sent to a
page called "/action_page.php".</p>

</body>
</html>
```

## HTML Forms

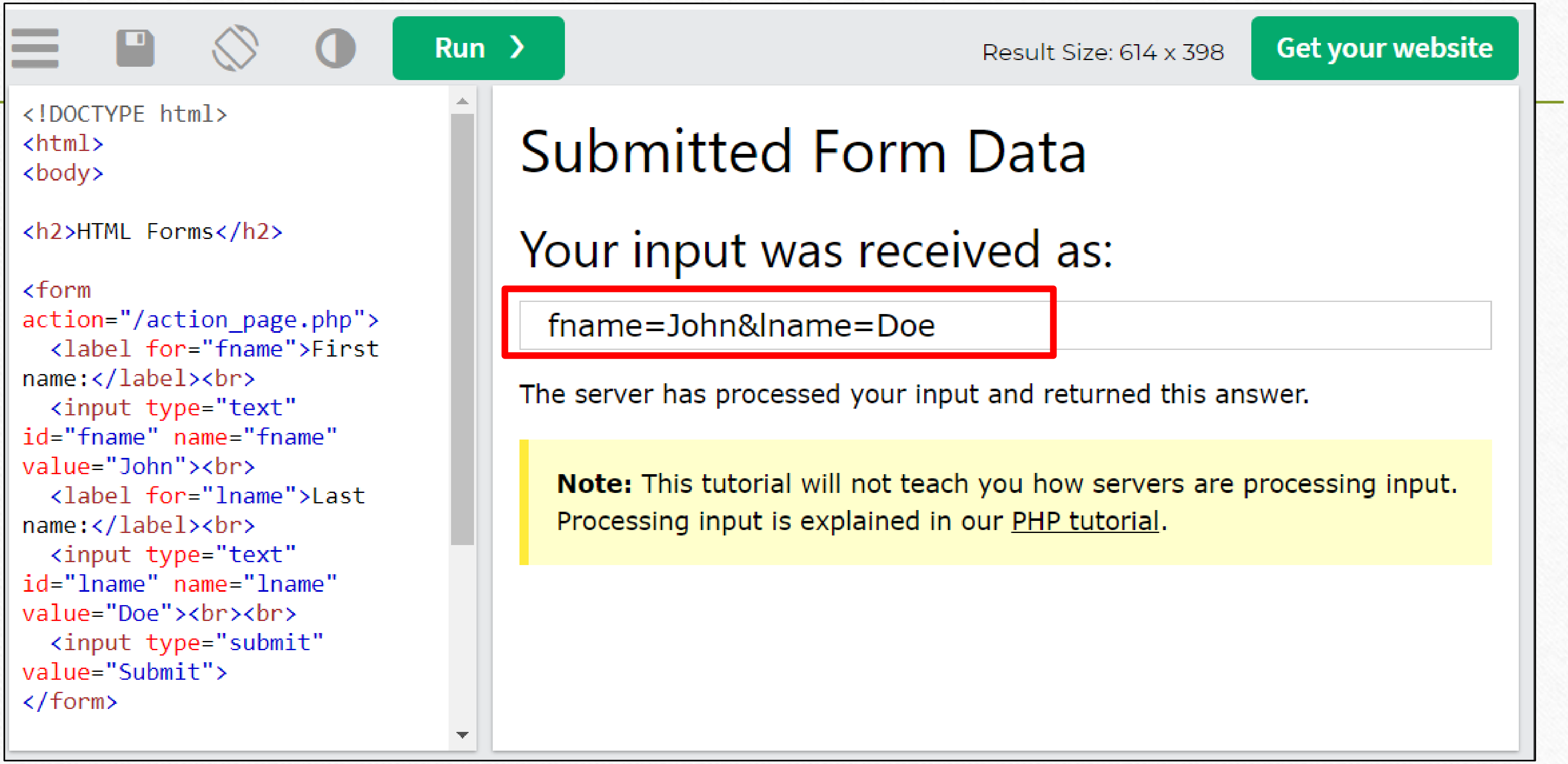
First name:
John

Last name:
Doe

Submit

If you click the "Submit" button, the form-data will be sent to a page called "/action_page.php".

https://www.w3schools.com/html/tryit.asp?filename=tryhtml_form_submit



```
<!DOCTYPE html>
<html>
<body>

<h2>HTML Forms</h2>

<form
action="/action_page.php">
  <label for="fname">First
name:</label><br>
  <input type="text"
id="fname" name="fname"
value="John"><br>
  <label for="lname">Last
name:</label><br>
  <input type="text"
id="lname" name="lname"
value="Doe"><br><br>
  <input type="submit"
value="Submit">
</form>
```

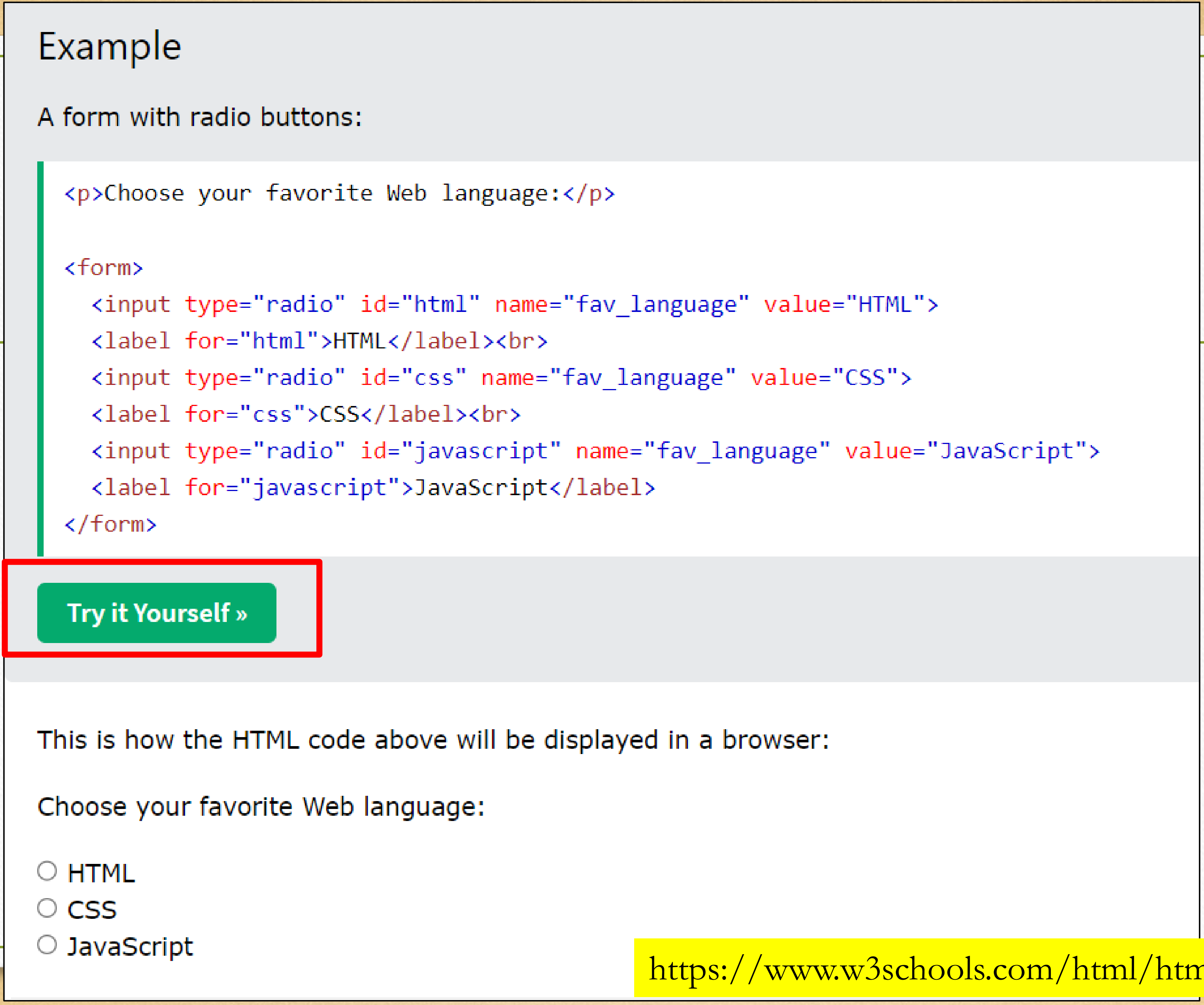
# Submitted Form Data

## Your input was received as:

```
fname=John&lname=Doe
```

The server has processed your input and returned this answer.

**Note:** This tutorial will not teach you how servers are processing input. Processing input is explained in our PHP tutorial.



## Example

A form with radio buttons:

```
<p>Choose your favorite Web language:</p>

<form>
  <input type="radio" id="html" name="fav_language" value="HTML">
  <label for="html">HTML</label><br>
  <input type="radio" id="css" name="fav_language" value="CSS">
  <label for="css">CSS</label><br>
  <input type="radio" id="javascript" name="fav_language" value="JavaScript">
  <label for="javascript">JavaScript</label>
</form>
```

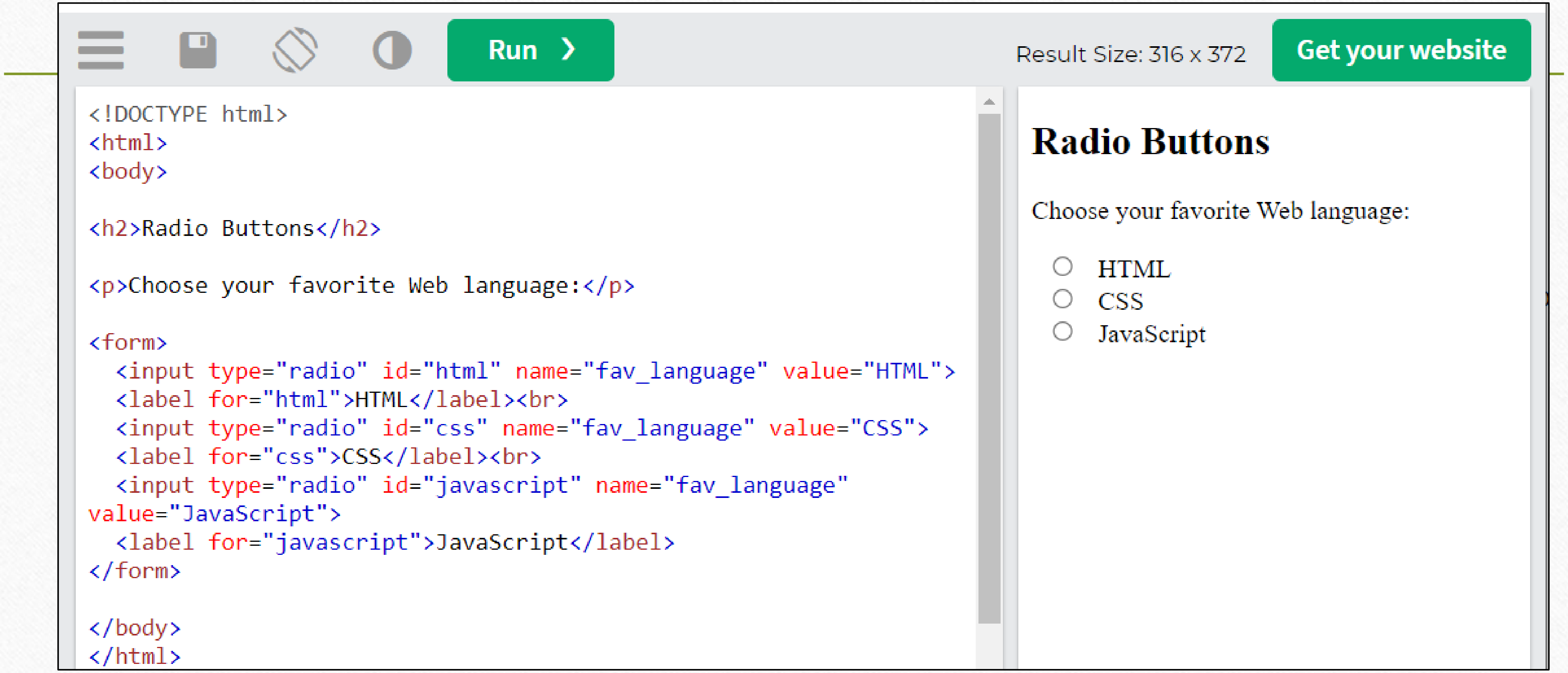
This is how the HTML code above will be displayed in a browser:

Choose your favorite Web language:

○ HTML
○ CSS
○ JavaScript

https://www.w3schools.com/html/html_forms.asp

# Try it yourself



```
<!DOCTYPE html>
<html>
<body>

<h2>Radio Buttons</h2>

<p>Choose your favorite Web language:</p>

<form>
  <input type="radio" id="html" name="fav_language" value="HTML">
  <label for="html">HTML</label><br>
  <input type="radio" id="css" name="fav_language" value="CSS">
  <label for="css">CSS</label><br>
  <input type="radio" id="javascript" name="fav_language"
value="JavaScript">
  <label for="javascript">JavaScript</label>
</form>

</body>
</html>
```

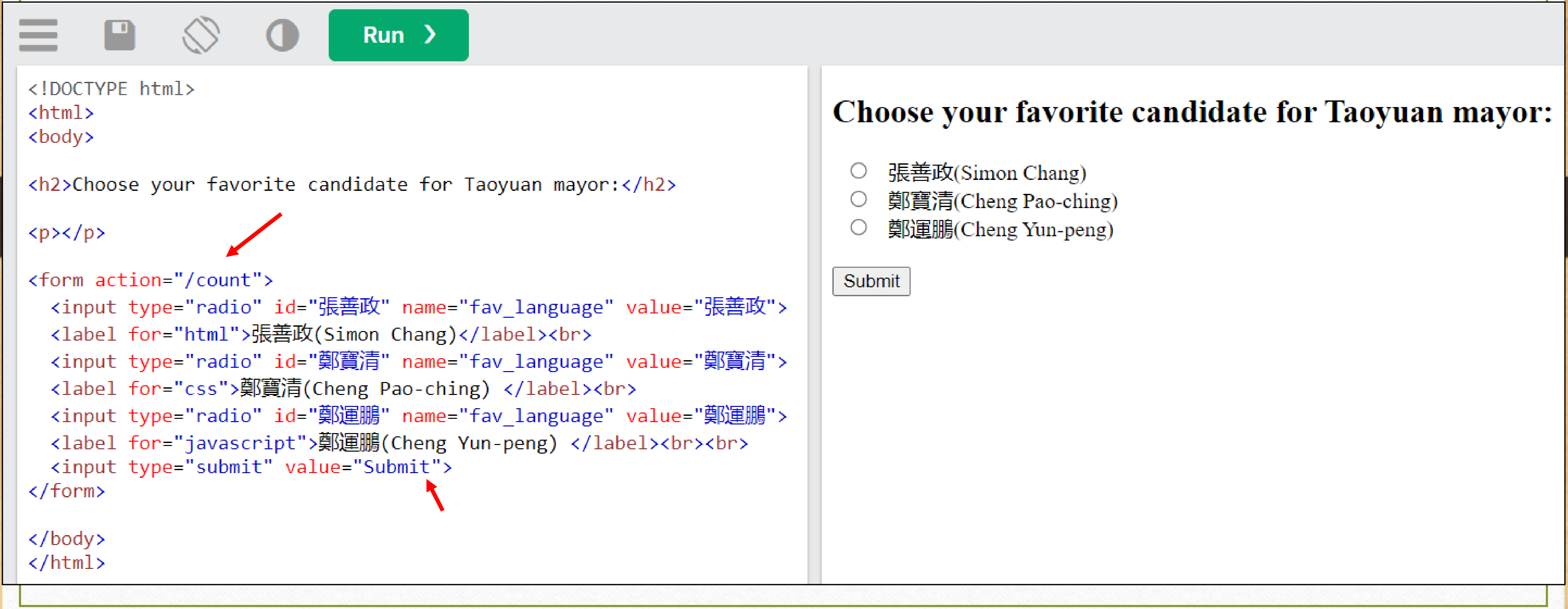
**Radio Buttons**

Choose your favorite Web language:

○ HTML
○ CSS
○ JavaScript

https://www.w3schools.com/html/tryit.asp?filename=tryhtml_form_radio

# Edit



```html
<!DOCTYPE html>
<html>
<body>

<h2>Choose your favorite candidate for Taoyuan mayor:</h2>

<p></p>

<form action="/count">
  <input type="radio" id="張善政" name="fav_language" value="張善政">
  <label for="html">張善政(Simon Chang)</label><br>
  <input type="radio" id="鄭寶清" name="fav_language" value="鄭寶清">
  <label for="css">鄭寶清(Cheng Pao-ching) </label><br>
  <input type="radio" id="鄭運鵬" name="fav_language" value="鄭運鵬">
  <label for="javascript">鄭運鵬(Cheng Yun-peng) </label><br><br>
  <input type="submit" value="Submit">
</form>

</body>
</html>
```

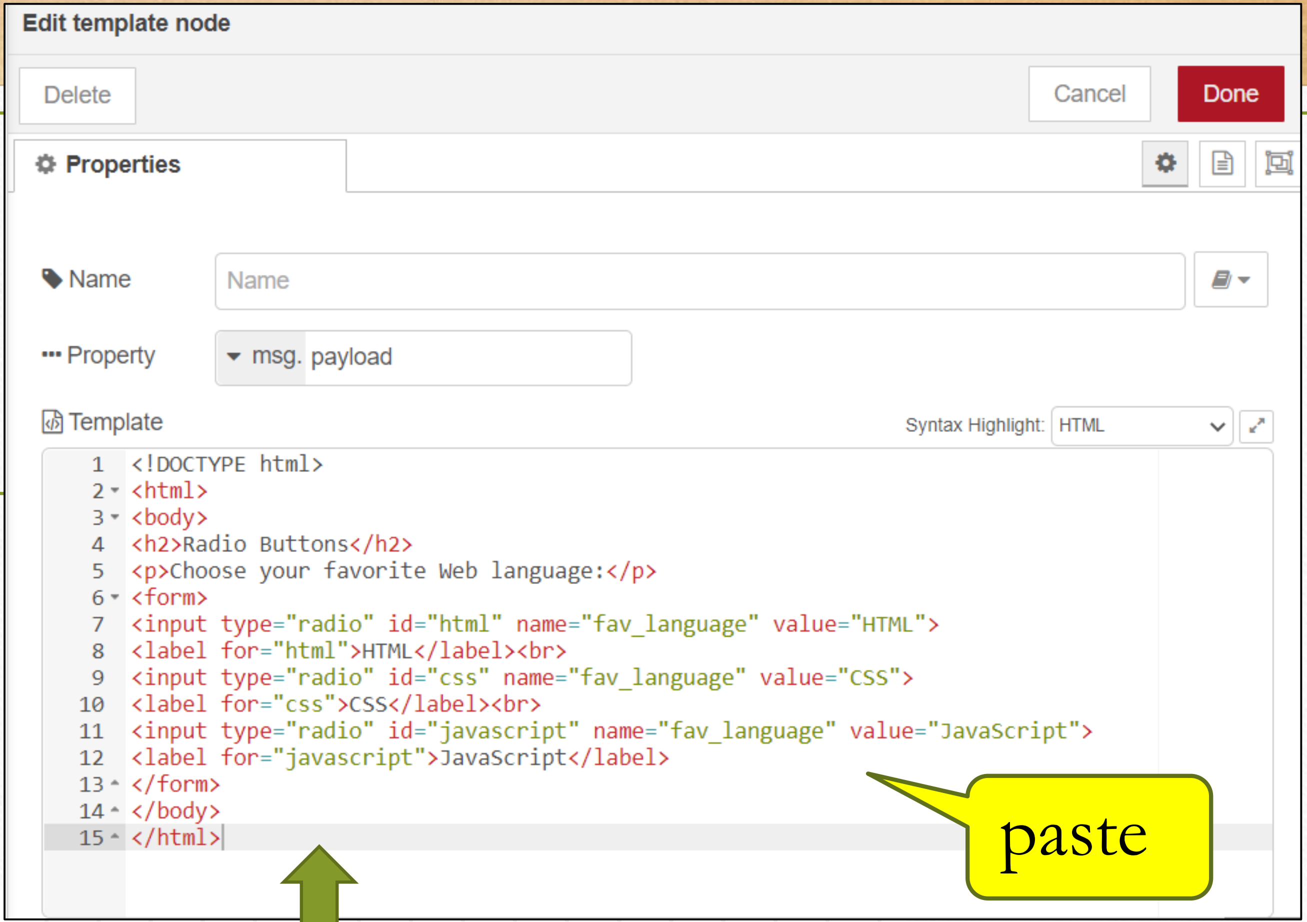
## Choose your favorite candidate for Taoyuan mayor:

○ 張善政(Simon Chang)
○ 鄭寶清(Cheng Pao-ching)
○ 鄭運鵬(Cheng Yun-peng)

Submit

```html
<!DOCTYPE html>
<html>
<body>

<h2>Radio Buttons</h2>

<p>Choose your favorite candidate for Taoyuan mayor:</p>

<form action="/count">
  <input type="radio" id="張善政" name="fav_language" value="張善政">
  <label for="html">張善政(Simon Chang)</label><br>
  <input type="radio" id="鄭寶清" name="fav_language" value="鄭寶清">
  <label for="css">鄭寶清(Cheng Pao-ching) </label><br>
  <input type="radio" id="鄭運鵬" name="fav_language" value="鄭運鵬">
  <label for="javascript">鄭運鵬(Cheng Yun-peng) </label><br><br>
  <input type="submit" value="Submit">
</form>

</body>
</html>
```

copy



Name: Name

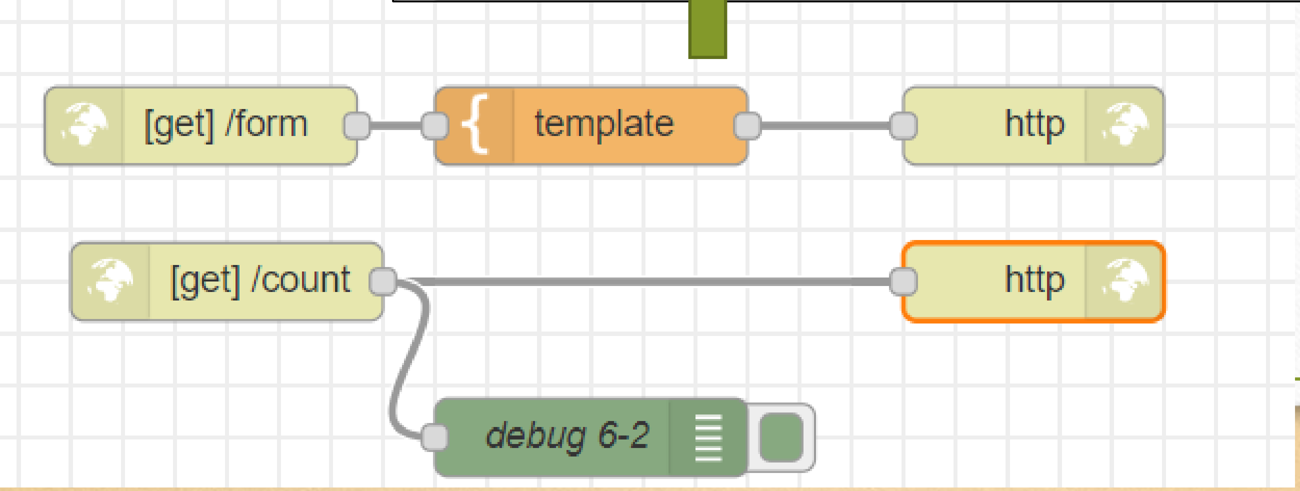
Property: msg. payload

Template

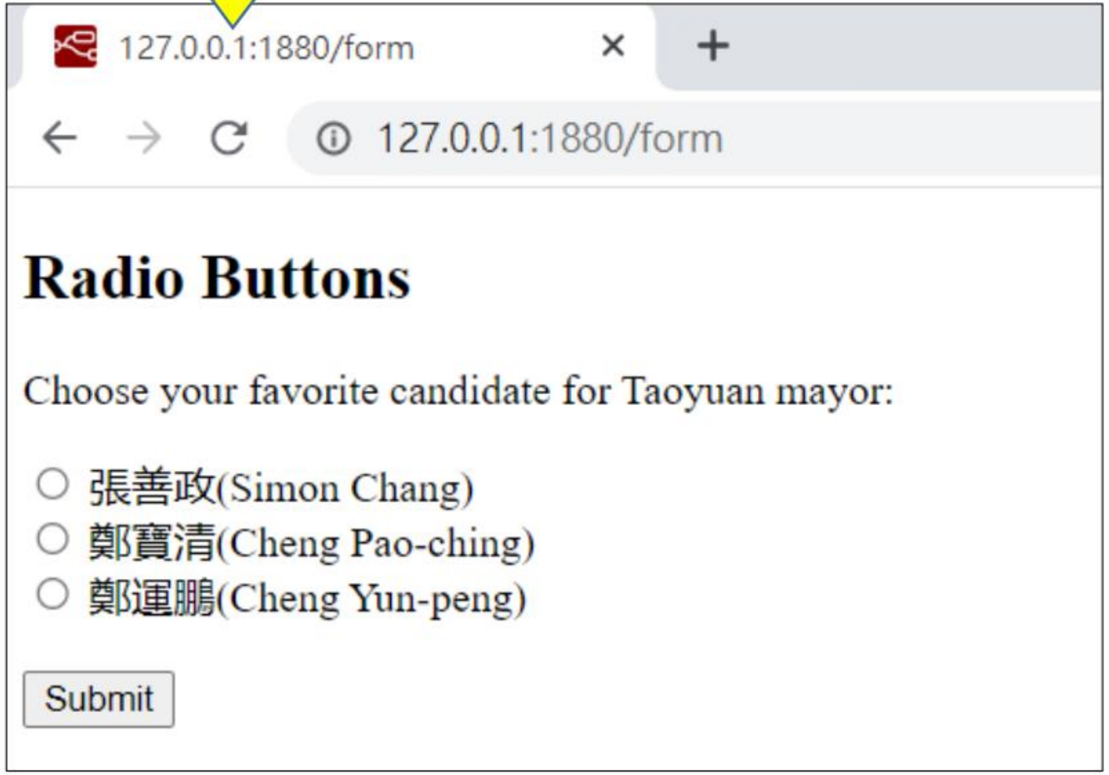
Syntax Highlight: HTML

```html
<!DOCTYPE html>
<html>
<body>
<h2>Radio Buttons</h2>
<p>Choose your favorite Web language:</p>
<form>
<input type="radio" id="html" name="fav_language" value="HTML">
<label for="html">HTML</label><br>
<input type="radio" id="css" name="fav_language" value="CSS">
<label for="css">CSS</label><br>
<input type="radio" id="javascript" name="fav_language" value="JavaScript">
<label for="javascript">JavaScript</label>
</form>
</body>
</html>
```

paste



127.0.0.1:1880/form

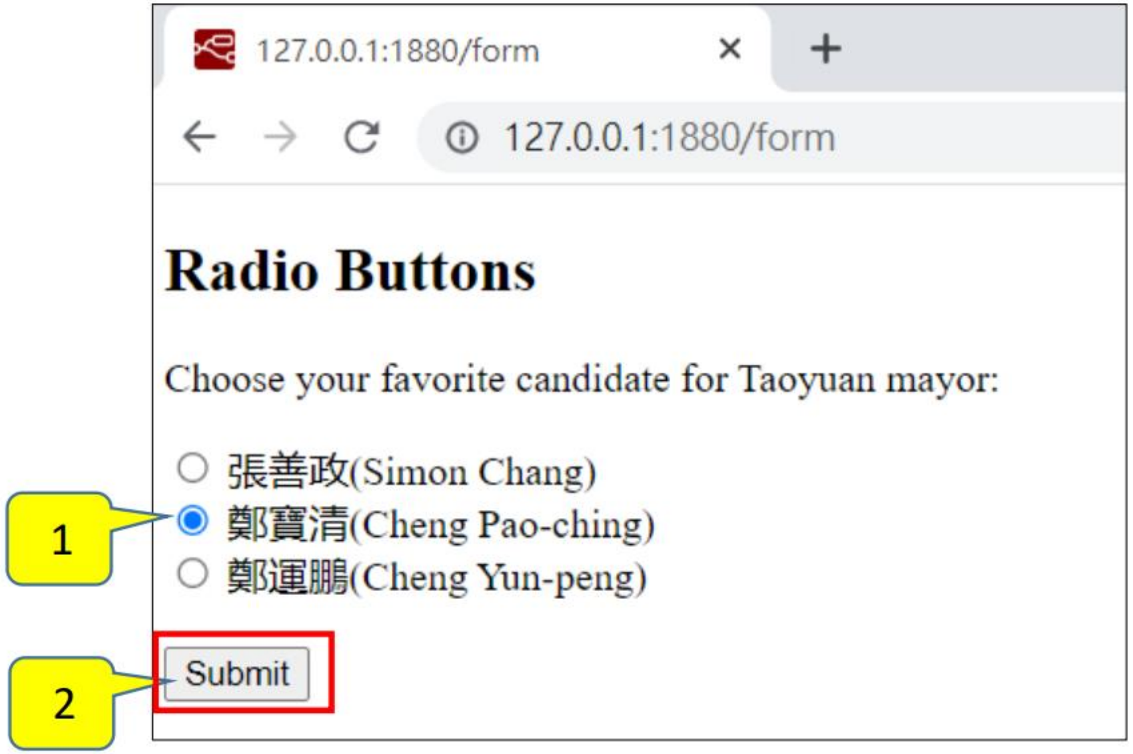


## Radio Buttons

Choose your favorite candidate for Taoyuan mayor:

○ 張善政(Simon Chang)
○ 鄭寶清(Cheng Pao-ching)
○ 鄭運鵬(Cheng Yun-peng)

# Select -> Submit



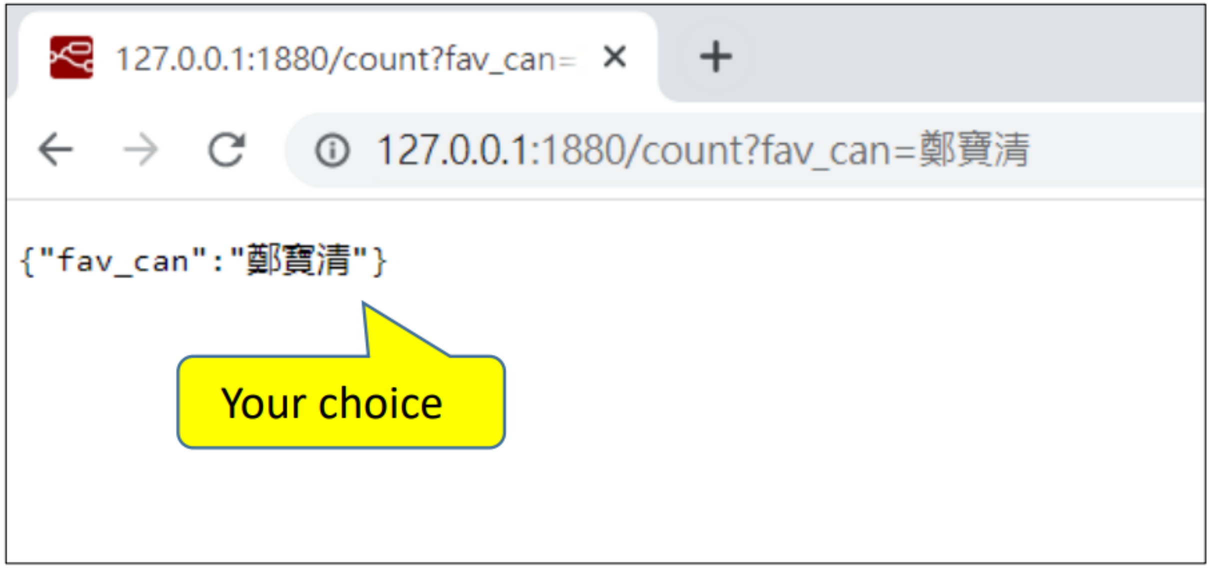
127.0.0.1:1880/form

## Radio Buttons

Choose your favorite candidate for Taoyuan mayor:

○ 張善政(Simon Chang)
● 鄭寶清(Cheng Pao-ching)
○ 鄭運鵬(Cheng Yun-peng)

# Showing your choice

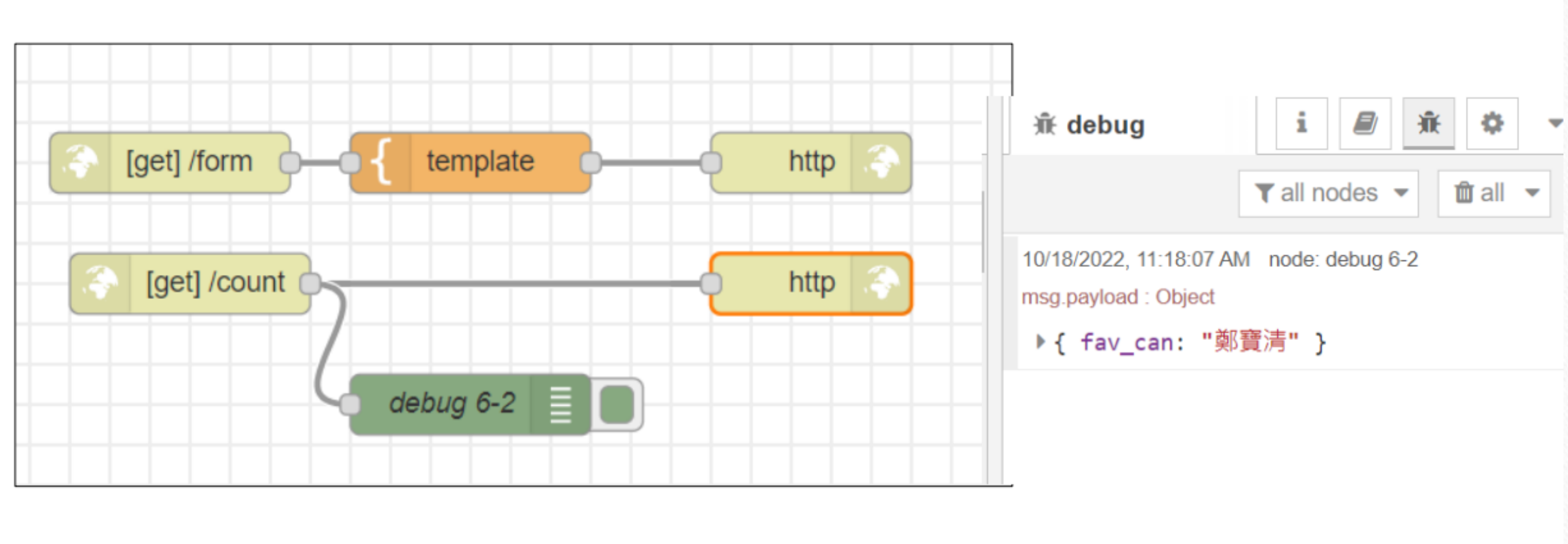


127.0.0.1:1880/count?fav_can=
127.0.0.1:1880/count?fav_can=鄭寶清

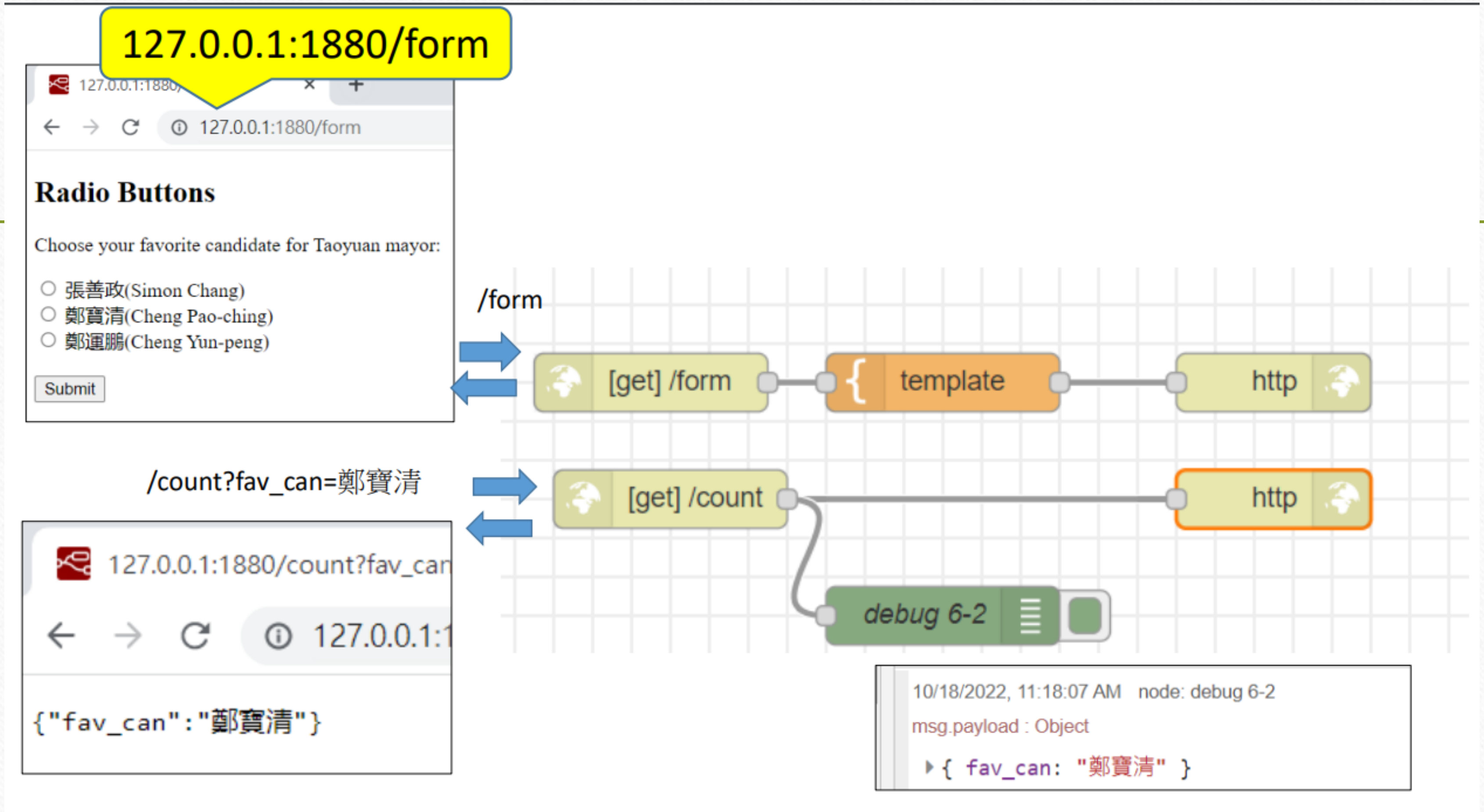
```
{"fav_can":"鄭寶清"}
```

Your choice

# debug



```
10/18/2022, 11:18:07 AM   node: debug 6-2
msg.payload : Object
▸{ fav_can: "鄭寶清" }
```



127.0.0.1:1880/form

127.0.0.1:1880/form

### Radio Buttons

Choose your favorite candidate for Taoyuan mayor:

○ 張善政(Simon Chang)
○ 鄭寶清(Cheng Pao-ching)
○ 鄭運鵬(Cheng Yun-peng)

Submit

/form

/count?fav_can=鄭寶清

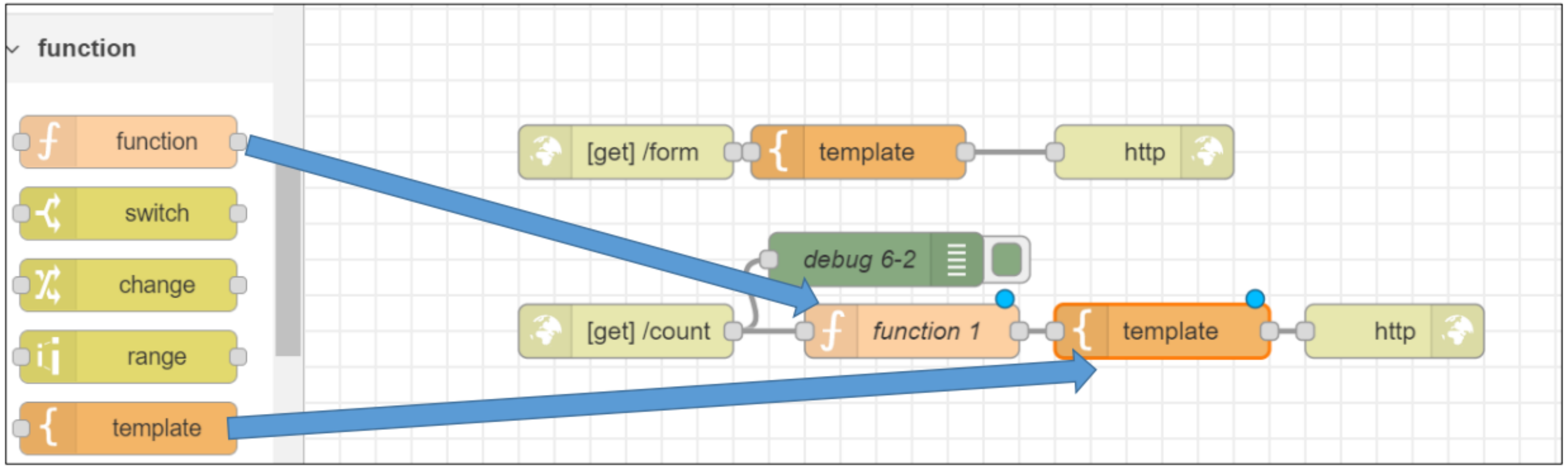
127.0.0.1:1880/count?fav_can

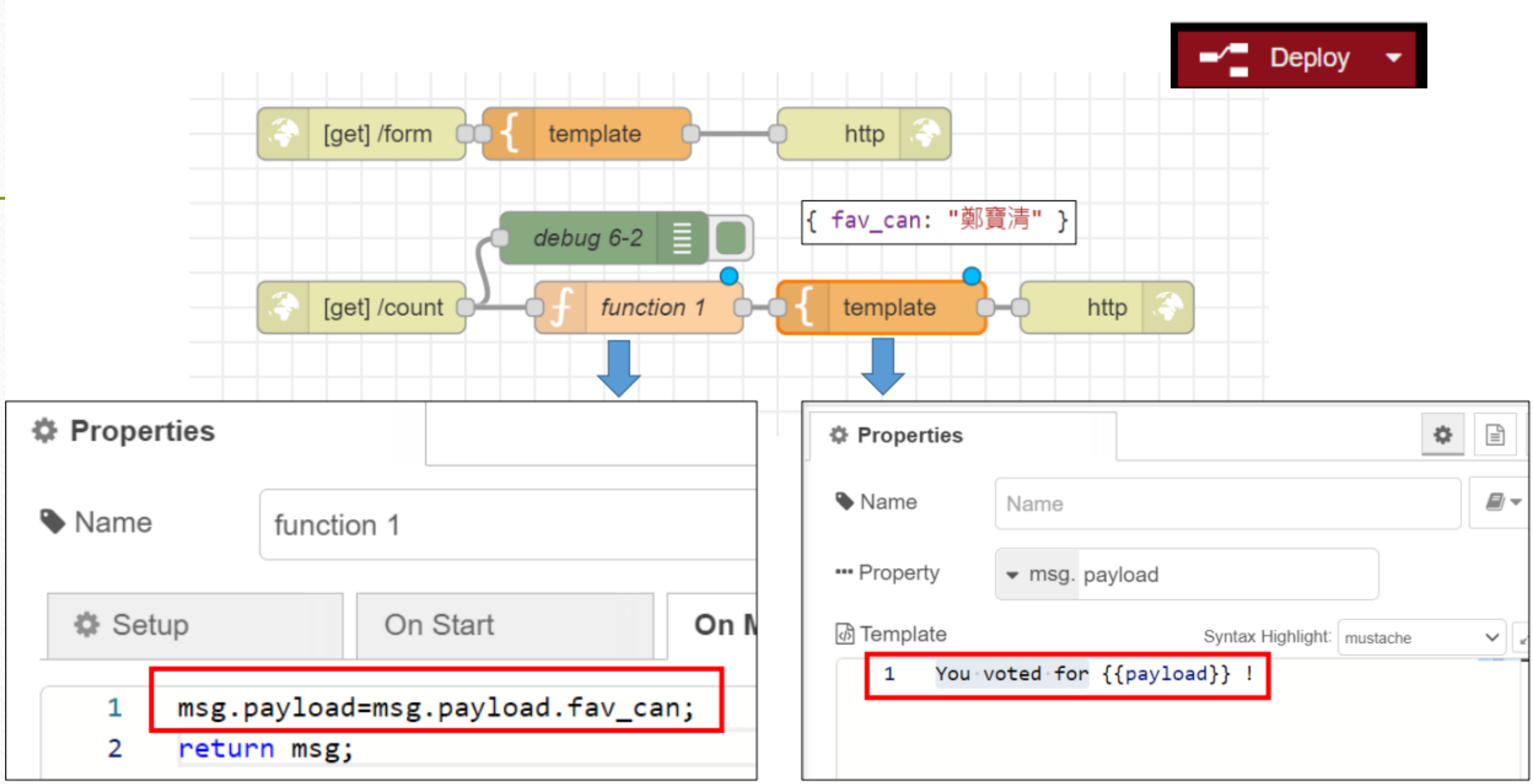
127.0.0.1:1

```
{"fav_can":"鄭寶清"}
```

```
10/18/2022, 11:18:07 AM   node: debug 6-2
msg.payload : Object
▸{ fav_can: "鄭寶清" }
```

# add a function node & a template node





```
{ fav_can: "鄭寶清" }
```

Properties

Name: function 1

Setup | On Start | On M

```
msg.payload=msg.payload.fav_can;
return msg;
```

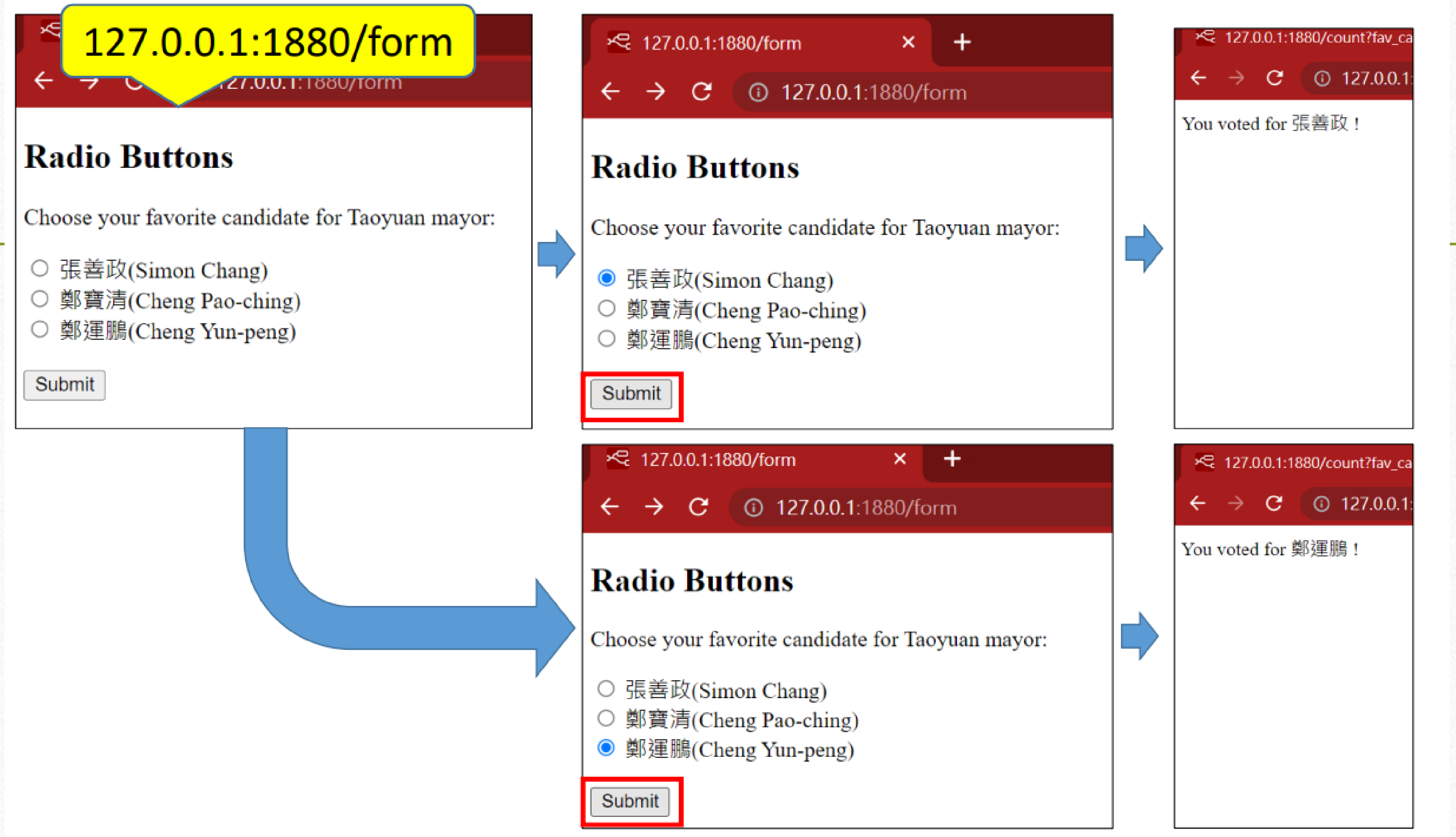
Properties

Name: Name

Property: msg. payload

Template — Syntax Highlight: mustache

```
You voted for {{payload}} !
```



127.0.0.1:1880/form

127.0.0.1:1880/form

## Radio Buttons

Choose your favorite candidate for Taoyuan mayor:

○ 張善政(Simon Chang)
○ 鄭寶清(Cheng Pao-ching)
○ 鄭運鵬(Cheng Yun-peng)

Submit

127.0.0.1:1880/form

## Radio Buttons

Choose your favorite candidate for Taoyuan mayor:

● 張善政(Simon Chang)
○ 鄭寶清(Cheng Pao-ching)
○ 鄭運鵬(Cheng Yun-peng)

Submit

127.0.0.1:1880/count?fav_ca

You voted for 張善政！

127.0.0.1:1880/form

## Radio Buttons

Choose your favorite candidate for Taoyuan mayor:

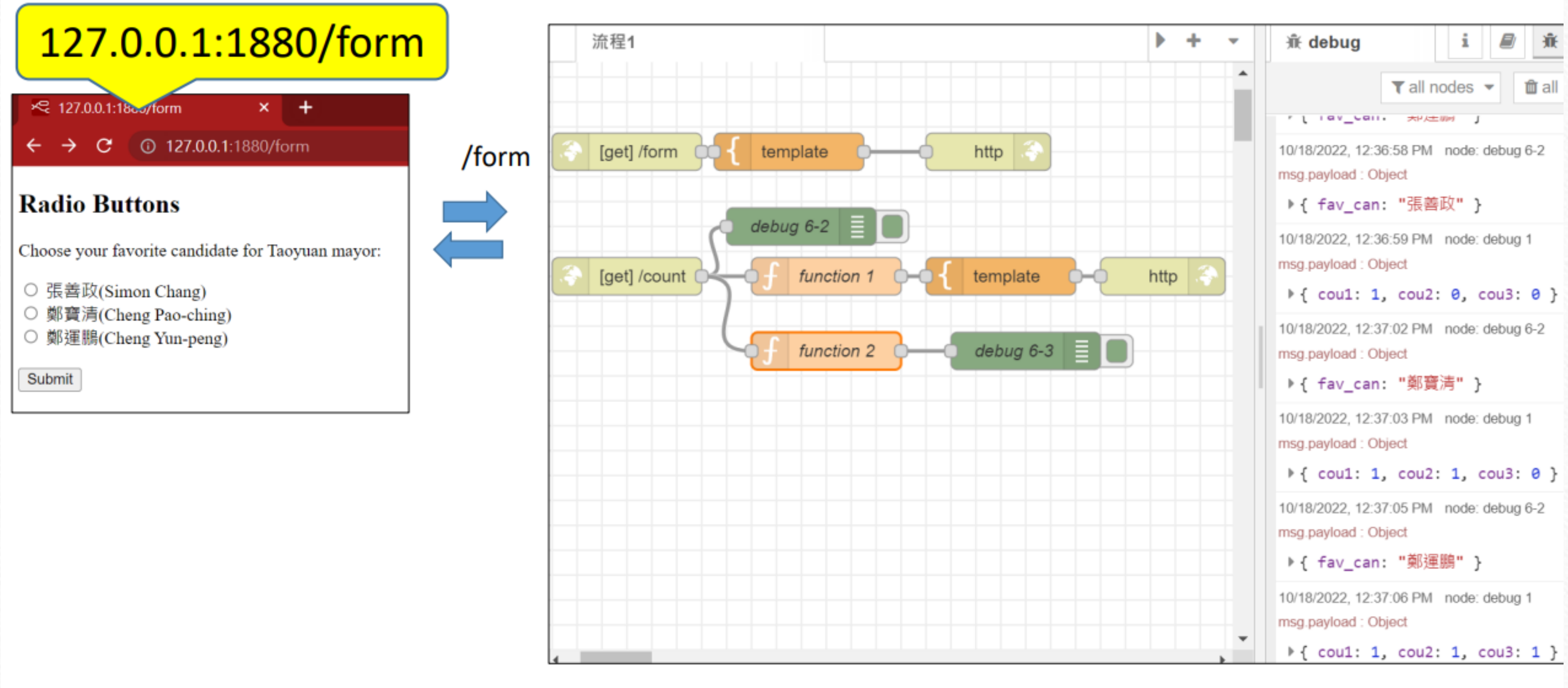
○ 張善政(Simon Chang)
○ 鄭寶清(Cheng Pao-ching)
● 鄭運鵬(Cheng Yun-peng)

Submit

127.0.0.1:1880/count?fav_ca

You voted for 鄭運鵬！

# Exercise 4-4 Evaluate the voting results



127.0.0.1:1880/form

127.0.0.1:1880/form

**Radio Buttons**

Choose your favorite candidate for Taoyuan mayor:

○ 張善政(Simon Chang)
○ 鄭寶清(Cheng Pao-ching)
○ 鄭運鵬(Cheng Yun-peng)

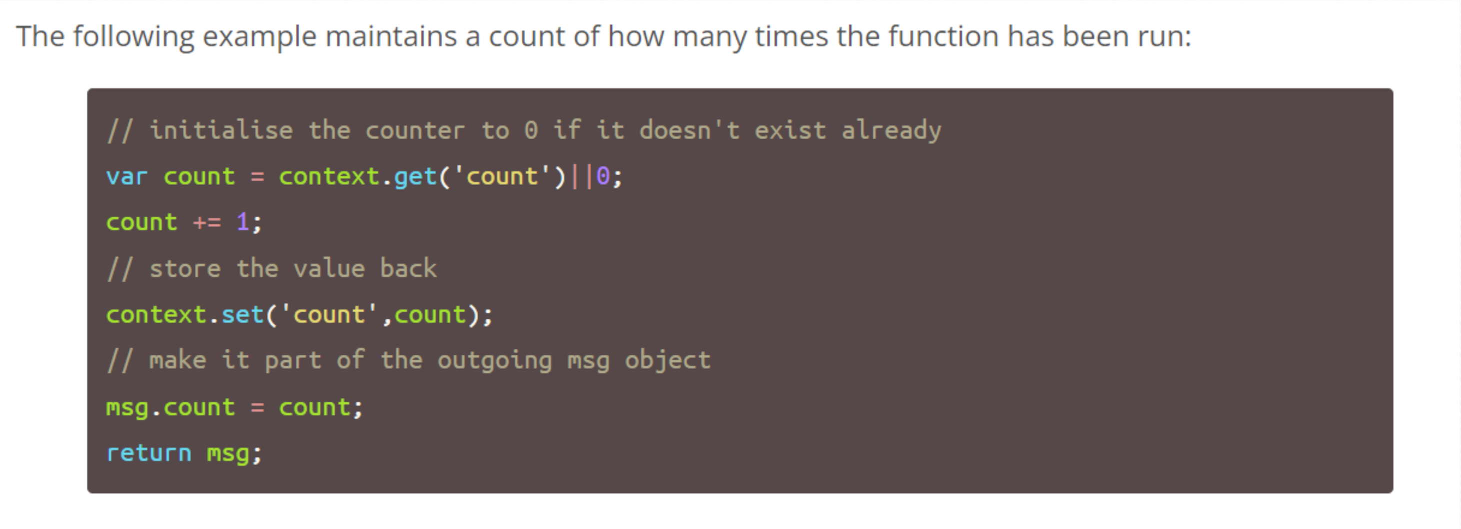
Submit

/form

```
10/18/2022, 12:36:58 PM   node: debug 6-2
msg.payload : Object
▸{ fav_can: "張善政" }
10/18/2022, 12:36:59 PM   node: debug 1
msg.payload : Object
▸{ cou1: 1, cou2: 0, cou3: 0 }
10/18/2022, 12:37:02 PM   node: debug 6-2
msg.payload : Object
▸{ fav_can: "鄭寶清" }
10/18/2022, 12:37:03 PM   node: debug 1
msg.payload : Object
▸{ cou1: 1, cou2: 1, cou3: 0 }
10/18/2022, 12:37:05 PM   node: debug 6-2
msg.payload : Object
▸{ fav_can: "鄭運鵬" }
10/18/2022, 12:37:06 PM   node: debug 1
msg.payload : Object
▸{ cou1: 1, cou2: 1, cou3: 1 }
```

# What is context?

- Node-RED provides a way to store information that can be shared between different nodes without using the messages that pass through a flow. This is called ‘context’.

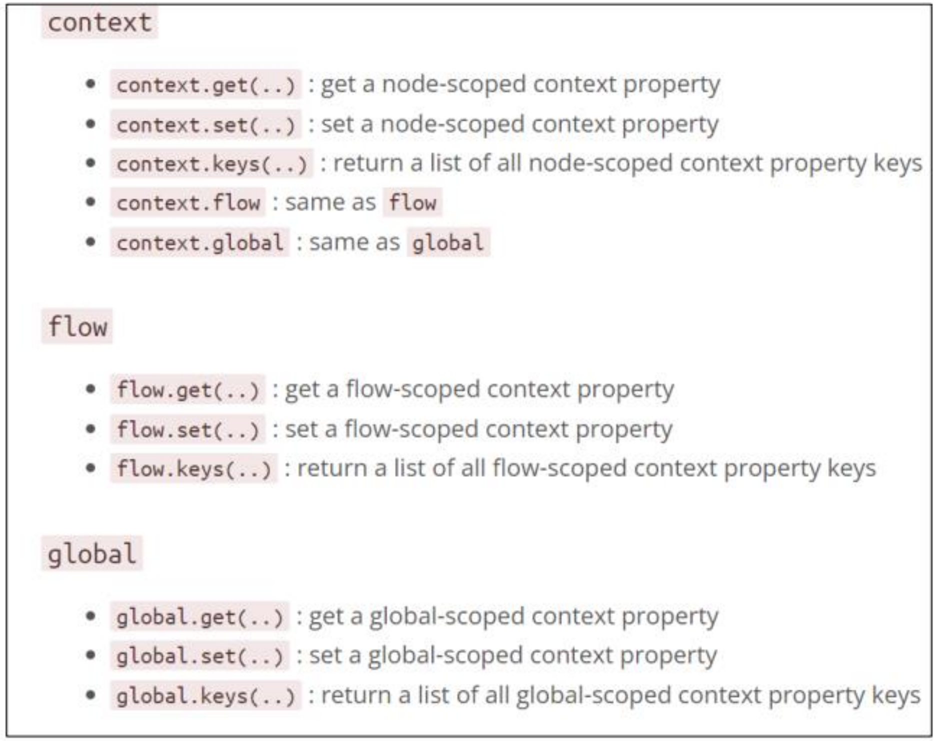


The following example maintains a count of how many times the function has been run:

```
// initialise the counter to 0 if it doesn't exist already
var count = context.get('count')||0;
count += 1;
// store the value back
context.set('count',count);
// make it part of the outgoing msg object
msg.count = count;
return msg;
```

https://nodered.org/docs/user-guide/writing-functions

# API Reference



`context`

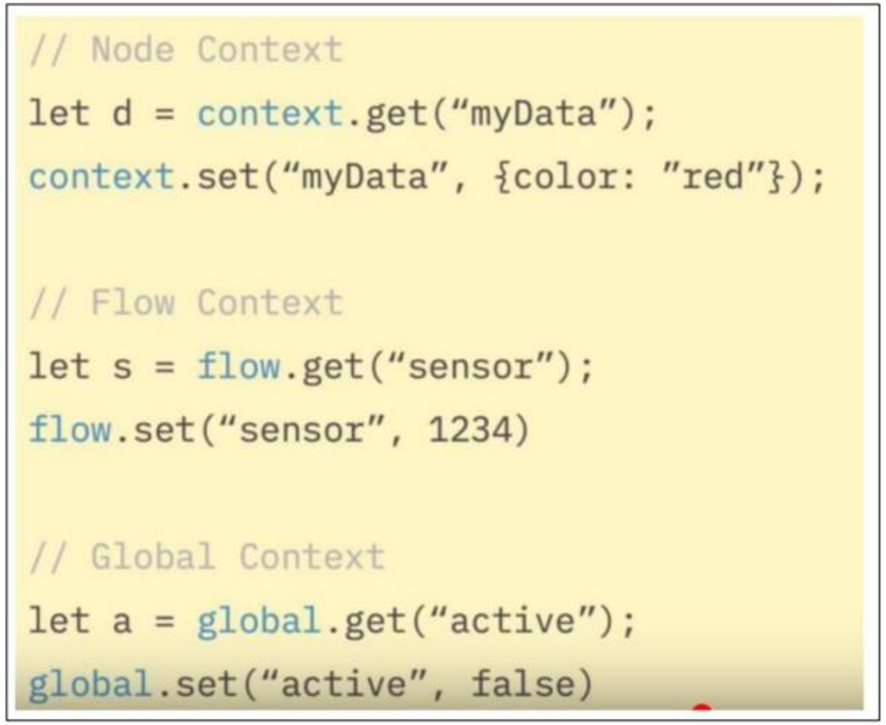
- `context.get(..)` : get a node-scoped context property
- `context.set(..)` : set a node-scoped context property
- `context.keys(..)` : return a list of all node-scoped context property keys
- `context.flow` : same as `flow`
- `context.global` : same as `global`

`flow`

- `flow.get(..)` : get a flow-scoped context property
- `flow.set(..)` : set a flow-scoped context property
- `flow.keys(..)` : return a list of all flow-scoped context property keys

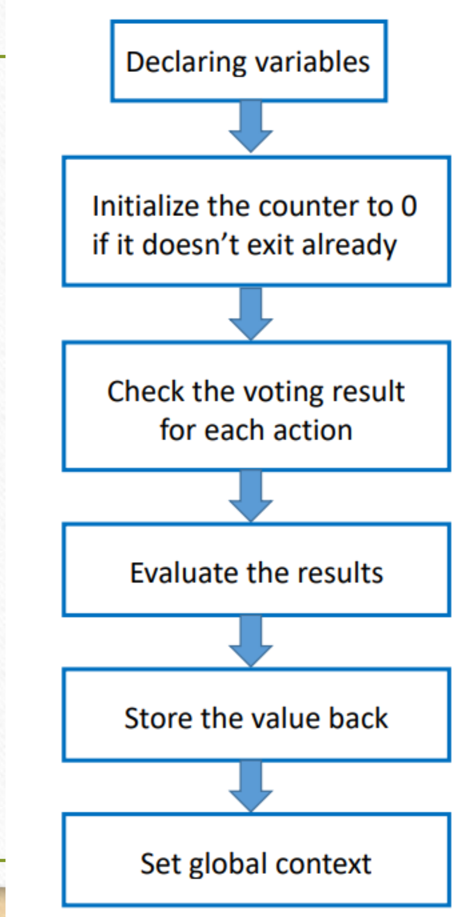
`global`

- `global.get(..)` : get a global-scoped context property
- `global.set(..)` : set a global-scoped context property
- `global.keys(..)` : return a list of all global-scoped context property keys



```
// Node Context
let d = context.get(“myData”);
context.set(“myData”, {color: ”red”});

// Flow Context
let s = flow.get(“sensor”);
flow.set(“sensor”, 1234)

// Global Context
let a = global.get(“active”);
global.set(“active”, false)
```

# Flow chart



Declaring variables

Initialize the counter to 0 if it doesn’t exit already

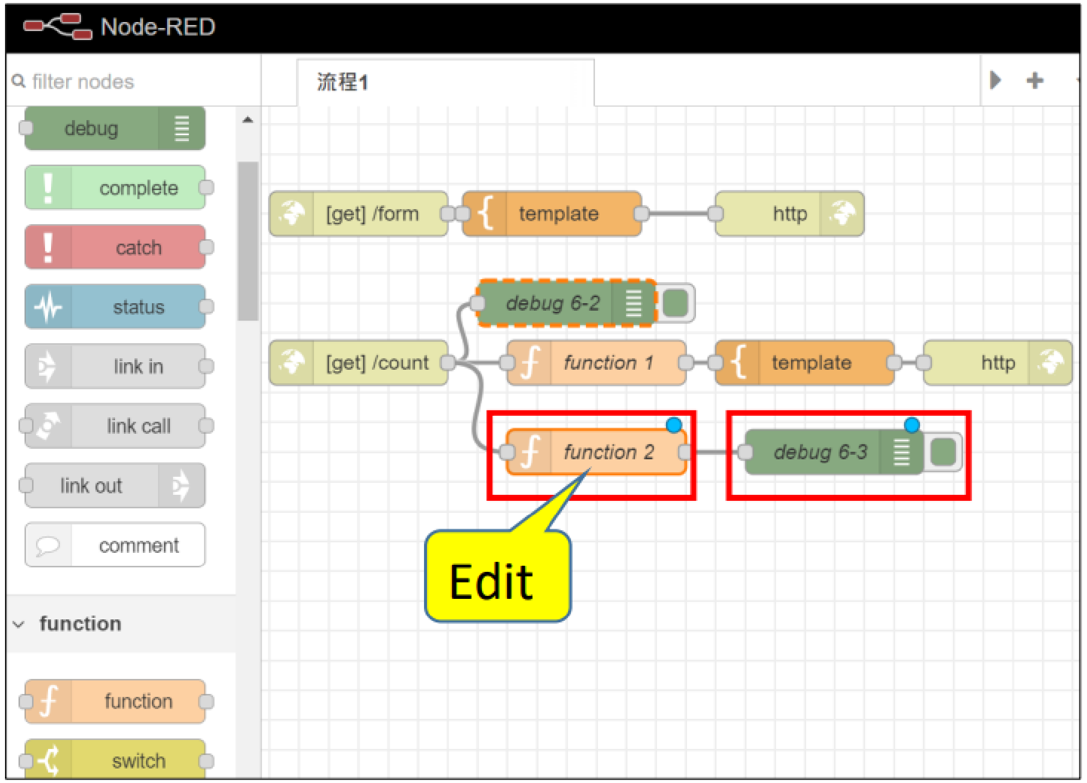
Check the voting result for each action

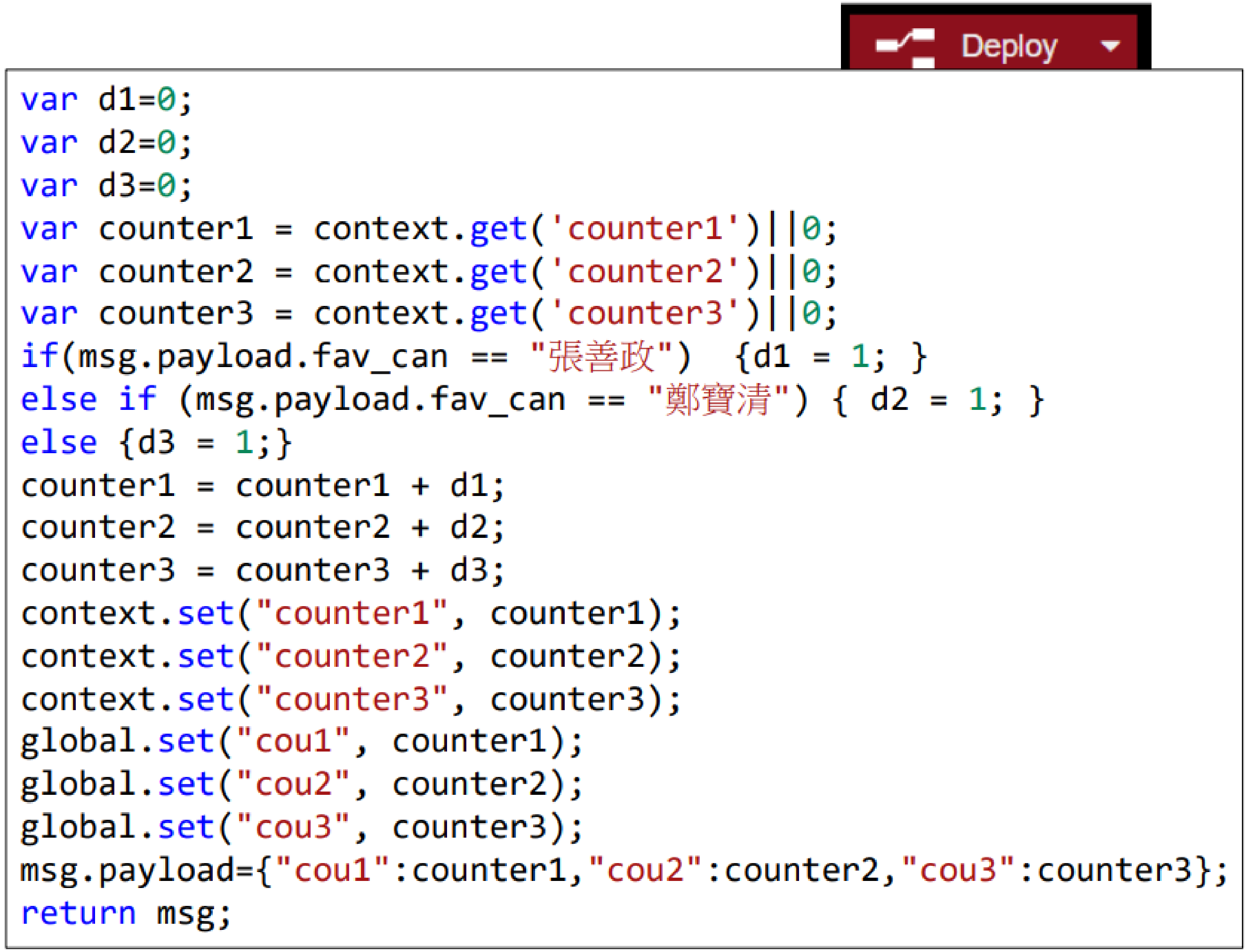
Evaluate the results

Store the value back

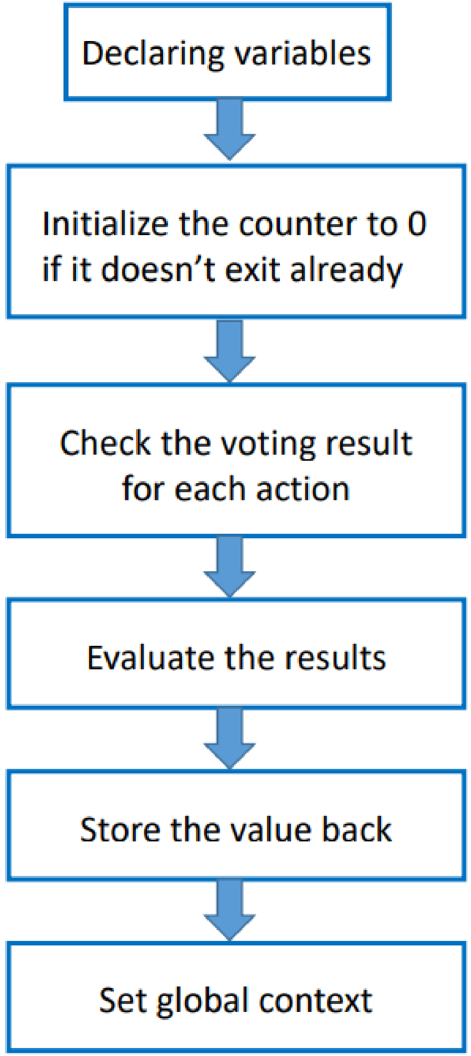
Set global context

# add a function node & a debug node





```
var d1=0;
var d2=0;
var d3=0;
var counter1 = context.get('counter1')||0;
var counter2 = context.get('counter2')||0;
var counter3 = context.get('counter3')||0;
if(msg.payload.fav_can == "張善政")  {d1 = 1; }
else if (msg.payload.fav_can == "鄭寶清") { d2 = 1; }
else {d3 = 1;}
counter1 = counter1 + d1;
counter2 = counter2 + d2;
counter3 = counter3 + d3;
context.set("counter1", counter1);
context.set("counter2", counter2);
context.set("counter3", counter3);
global.set("cou1", counter1);
global.set("cou2", counter2);
global.set("cou3", counter3);
msg.payload={"cou1":counter1,"cou2":counter2,"cou3":counter3};
return msg;
```



Declaring variables

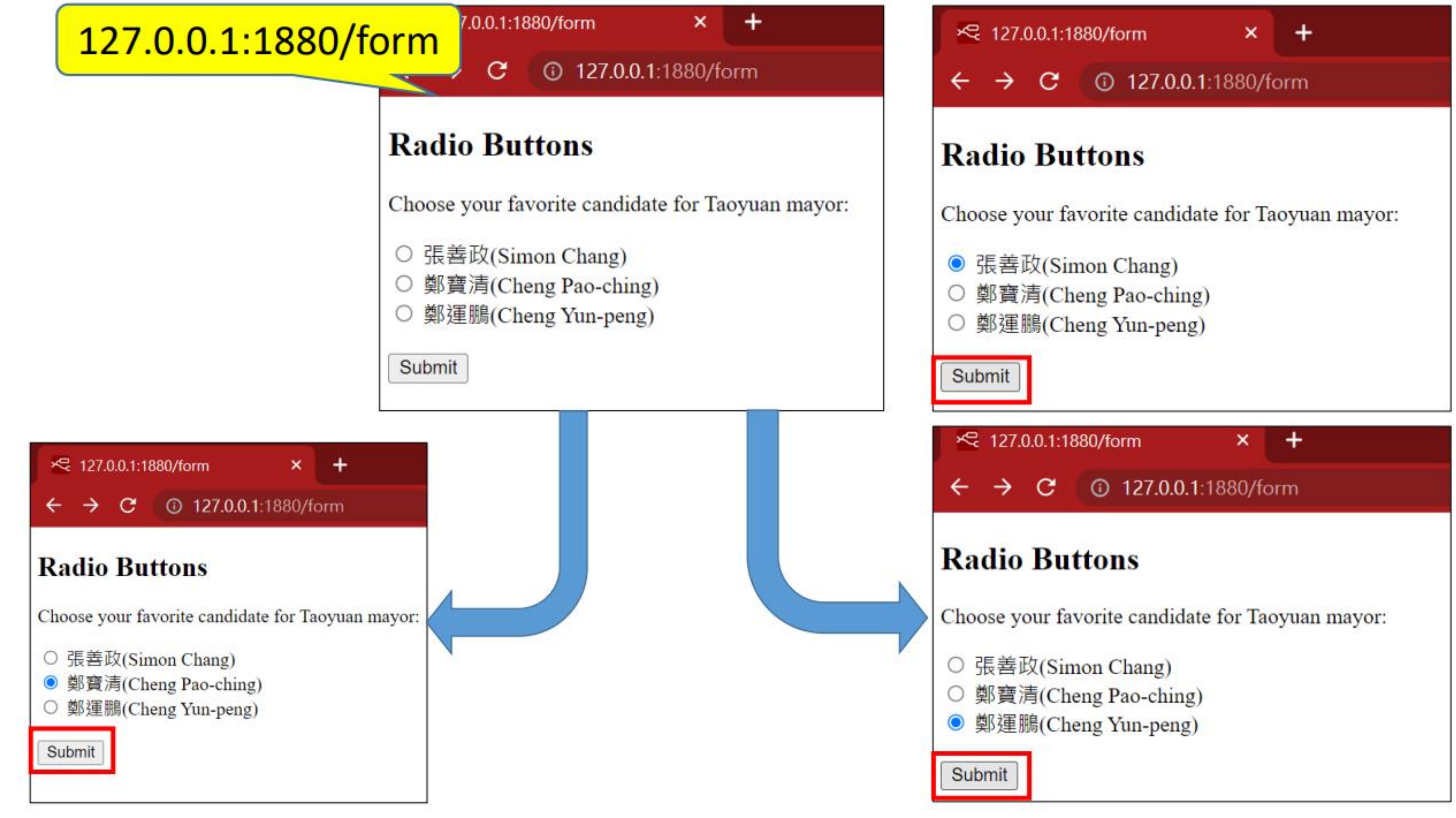
Initialize the counter to 0
if it doesn't exit already

Check the voting result
for each action

Evaluate the results

Store the value back

Set global context



127.0.0.1:1880/form

127.0.0.1:1880/form

## Radio Buttons

Choose your favorite candidate for Taoyuan mayor:

○ 張善政(Simon Chang)
○ 鄭寶清(Cheng Pao-ching)
○ 鄭運鵬(Cheng Yun-peng)

127.0.0.1:1880/form

## Radio Buttons

Choose your favorite candidate for Taoyuan mayor:

● 張善政(Simon Chang)
○ 鄭寶清(Cheng Pao-ching)
○ 鄭運鵬(Cheng Yun-peng)

127.0.0.1:1880/form

## Radio Buttons

Choose your favorite candidate for Taoyuan mayor:

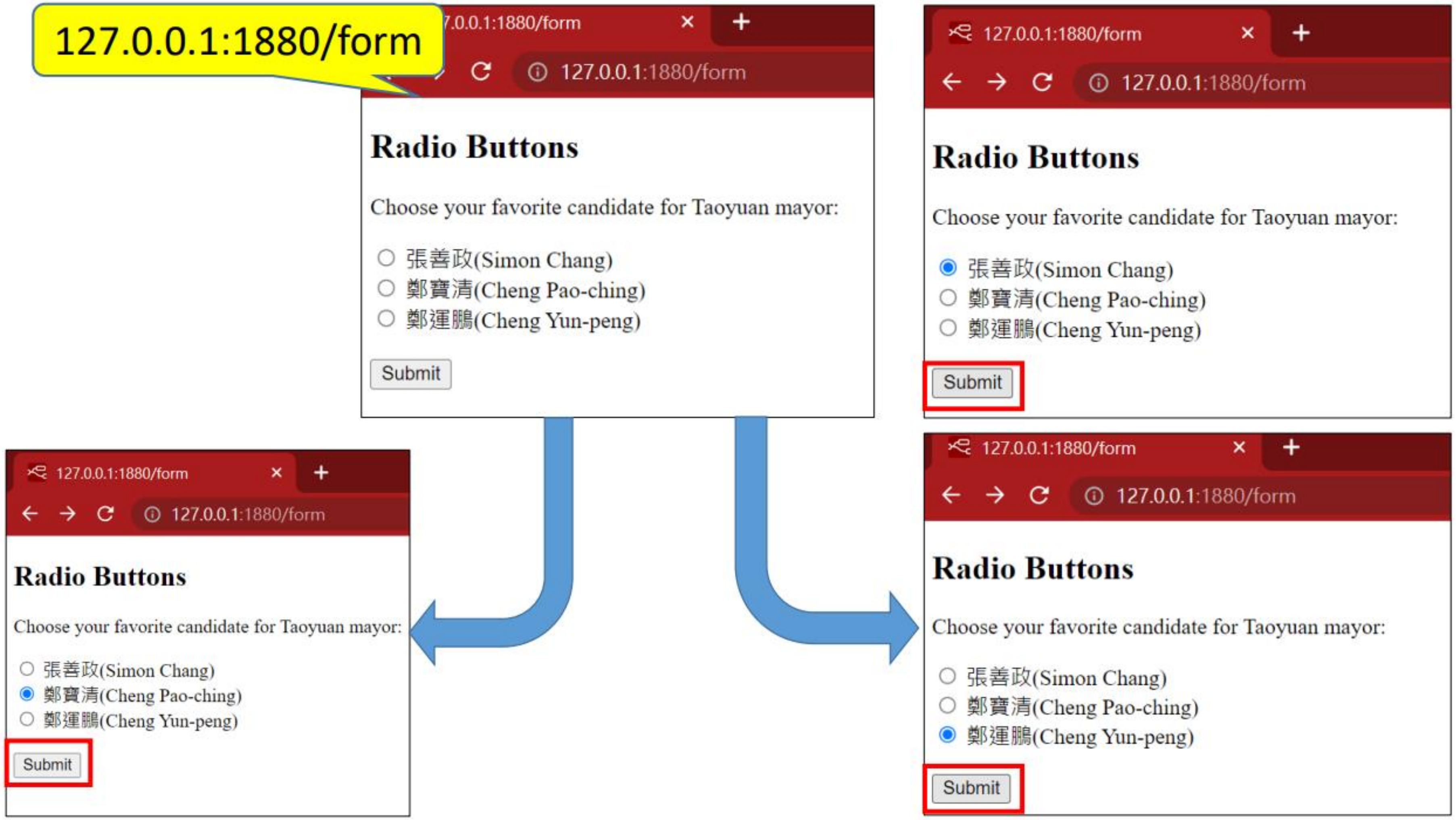
○ 張善政(Simon Chang)
● 鄭寶清(Cheng Pao-ching)
○ 鄭運鵬(Cheng Yun-peng)

127.0.0.1:1880/form

## Radio Buttons

Choose your favorite candidate for Taoyuan mayor:

○ 張善政(Simon Chang)
○ 鄭寶清(Cheng Pao-ching)
● 鄭運鵬(Cheng Yun-peng)



127.0.0.1:1880/form

## Radio Buttons

Choose your favorite candidate for Taoyuan mayor:

○ 張善政(Simon Chang)
○ 鄭寶清(Cheng Pao-ching)
○ 鄭運鵬(Cheng Yun-peng)

## Radio Buttons

Choose your favorite candidate for Taoyuan mayor:

● 張善政(Simon Chang)
○ 鄭寶清(Cheng Pao-ching)
○ 鄭運鵬(Cheng Yun-peng)

## Radio Buttons

Choose your favorite candidate for Taoyuan mayor:

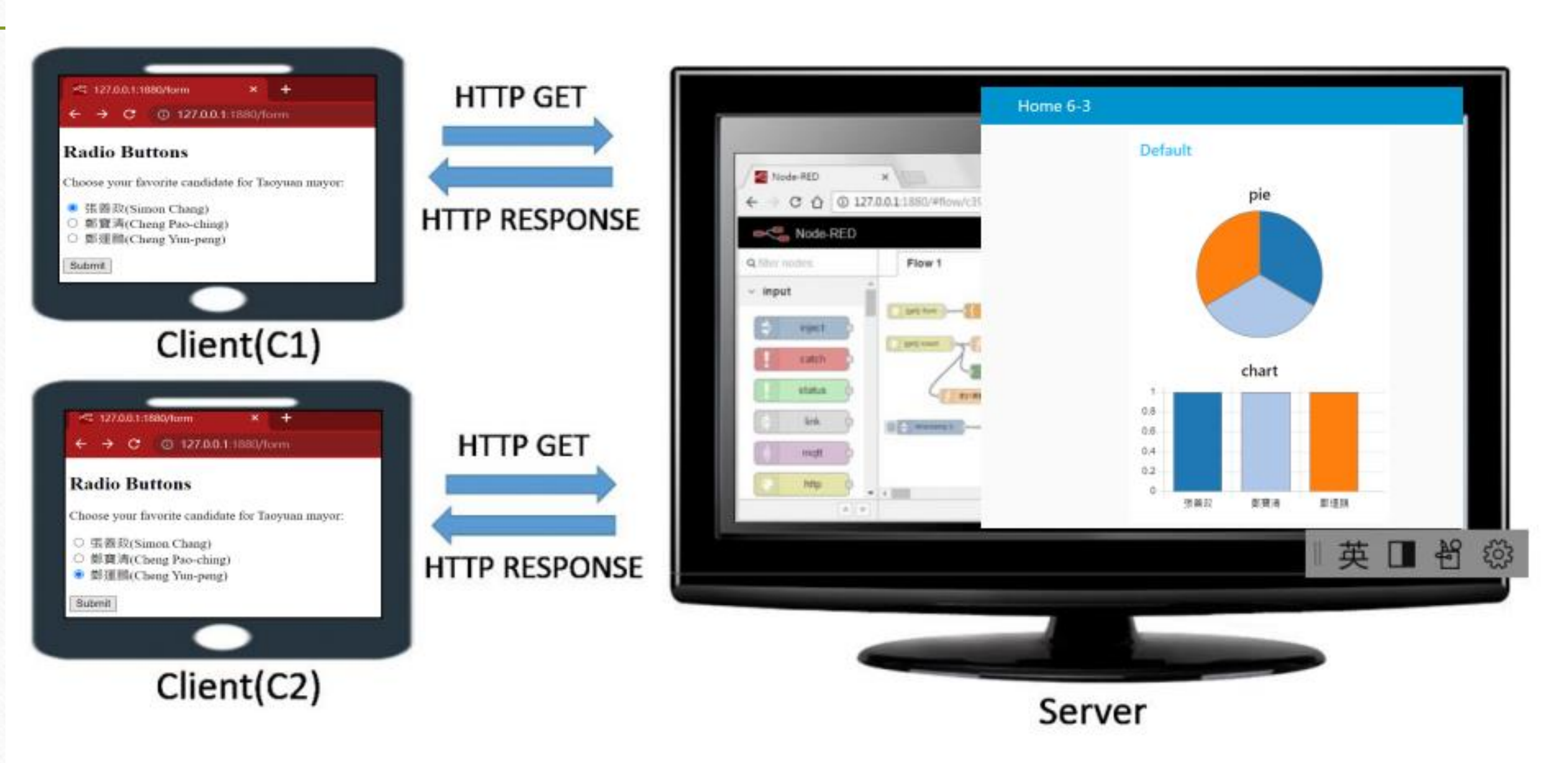
○ 張善政(Simon Chang)
● 鄭寶清(Cheng Pao-ching)
○ 鄭運鵬(Cheng Yun-peng)

## Radio Buttons

Choose your favorite candidate for Taoyuan mayor:

○ 張善政(Simon Chang)
○ 鄭寶清(Cheng Pao-ching)
● 鄭運鵬(Cheng Yun-peng)

# Exercise 4-5 Visualize the voting results



Client(C1):

**Radio Buttons**

Choose your favorite candidate for Taoyuan mayor:

● 張善政(Simon Chang)
○ 鄭寶清(Cheng Pao-ching)
○ 鄭運鵬(Cheng Yun-peng)

Submit

Client(C2):

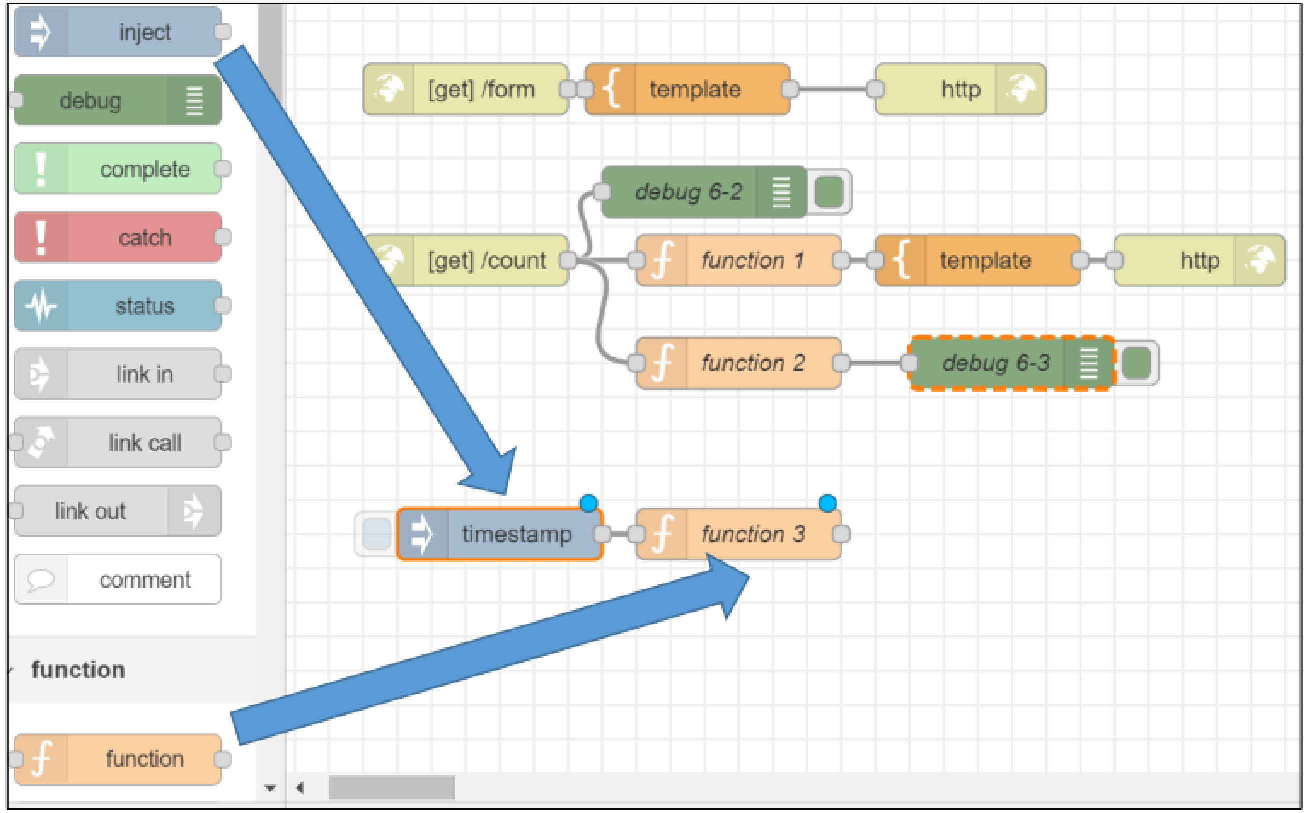
**Radio Buttons**

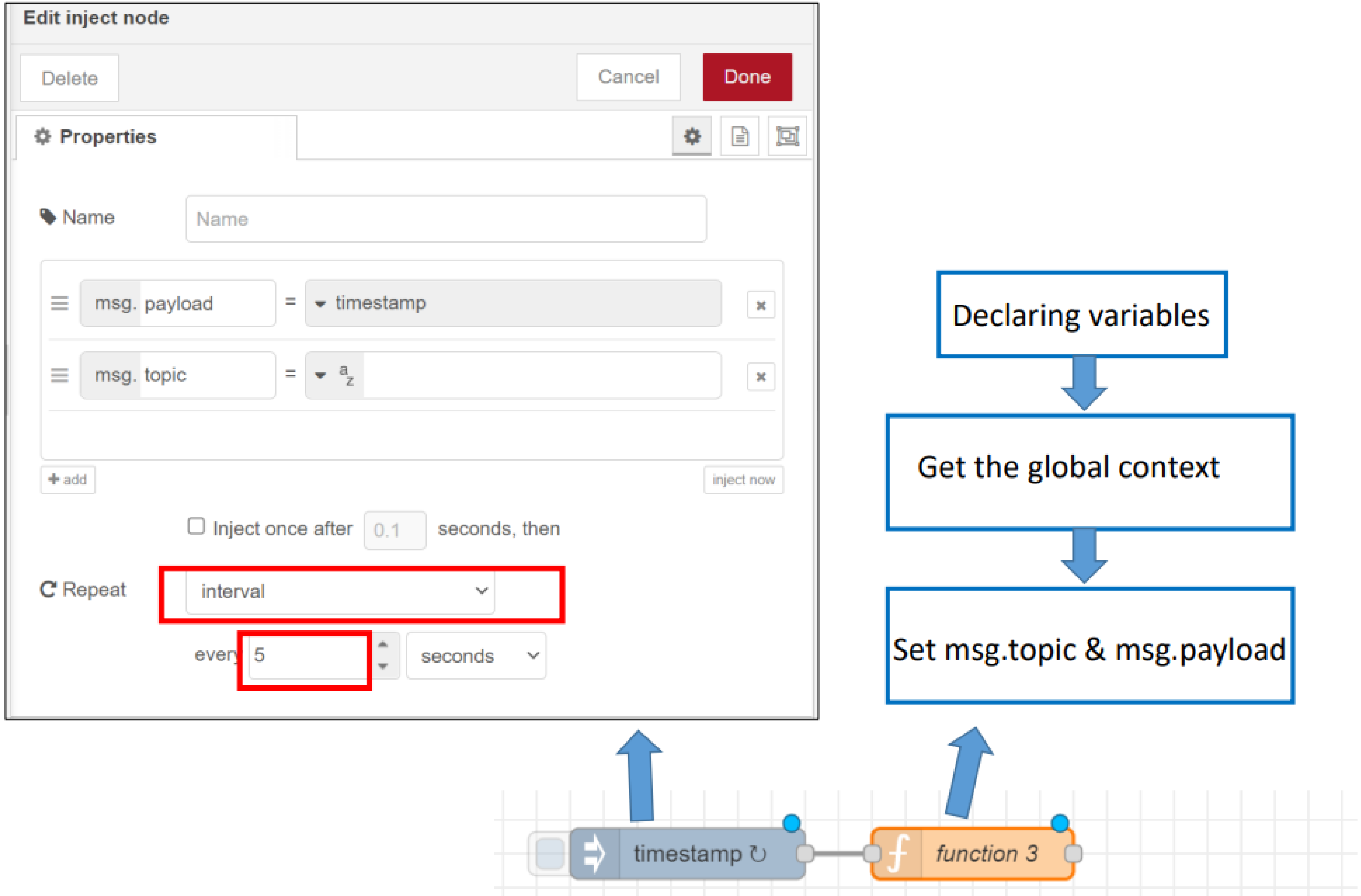
Choose your favorite candidate for Taoyuan mayor:

○ 張善政(Simon Chang)
○ 鄭寶清(Cheng Pao-ching)
● 鄭運鵬(Cheng Yun-peng)

Submit

# add a inject node & a function node





**Edit inject node**

Delete | Cancel | Done

Properties

Name: Name

msg. payload = timestamp

msg. topic = ᵃz

+ add | inject now

☐ Inject once after 0.1 seconds, then

Repeat: interval

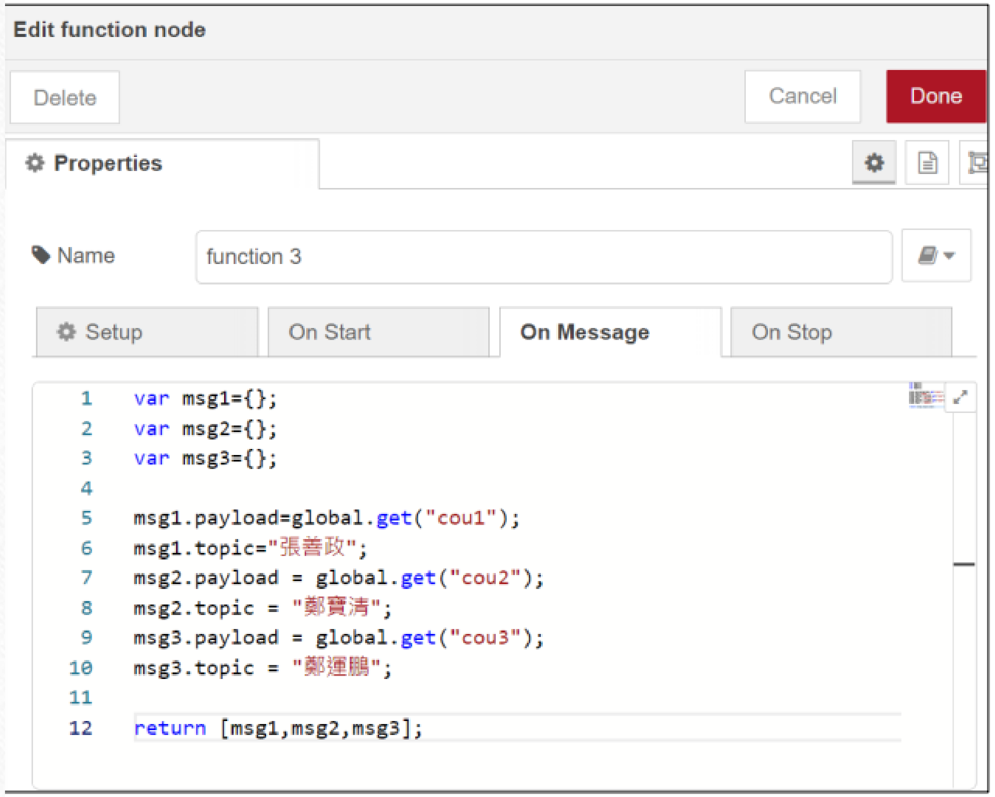
every 5 seconds

Declaring variables

Get the global context

Set msg.topic & msg.payload

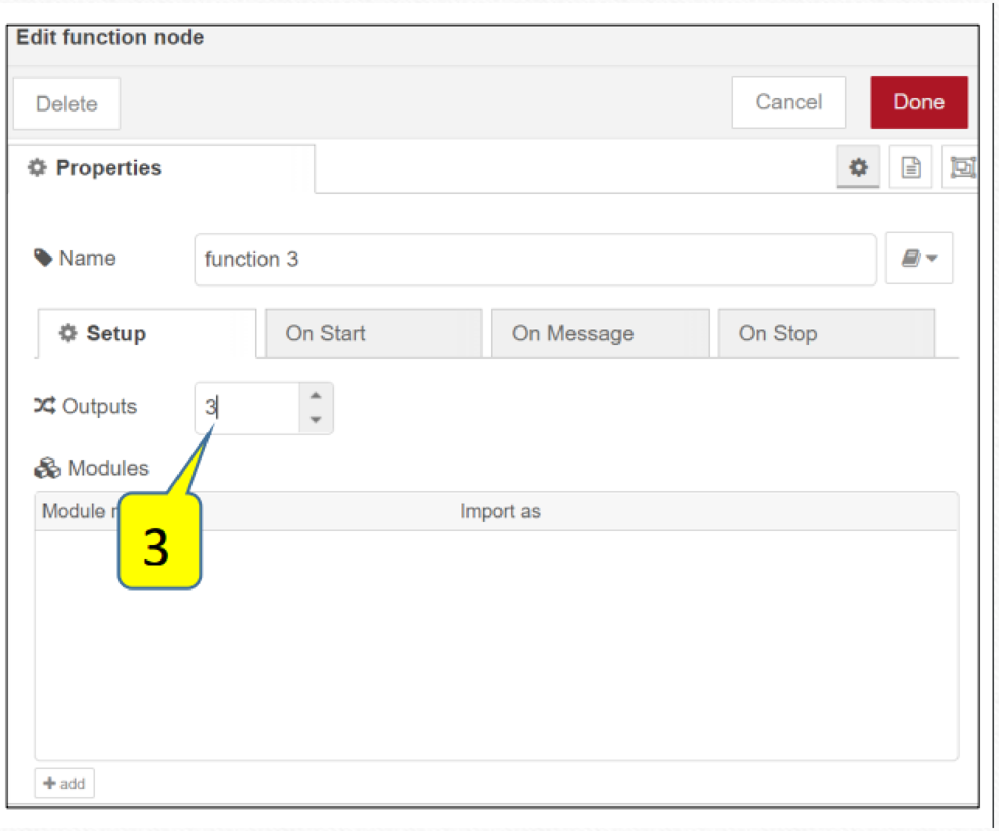
# Edit function node



Edit function node

Name: function 3

```
var msg1={};
var msg2={};
var msg3={};

msg1.payload=global.get("cou1");
msg1.topic="張善政";
msg2.payload = global.get("cou2");
msg2.topic = "鄭寶清";
msg3.payload = global.get("cou3");
msg3.topic = "鄭運鵬";

return [msg1,msg2,msg3];
```

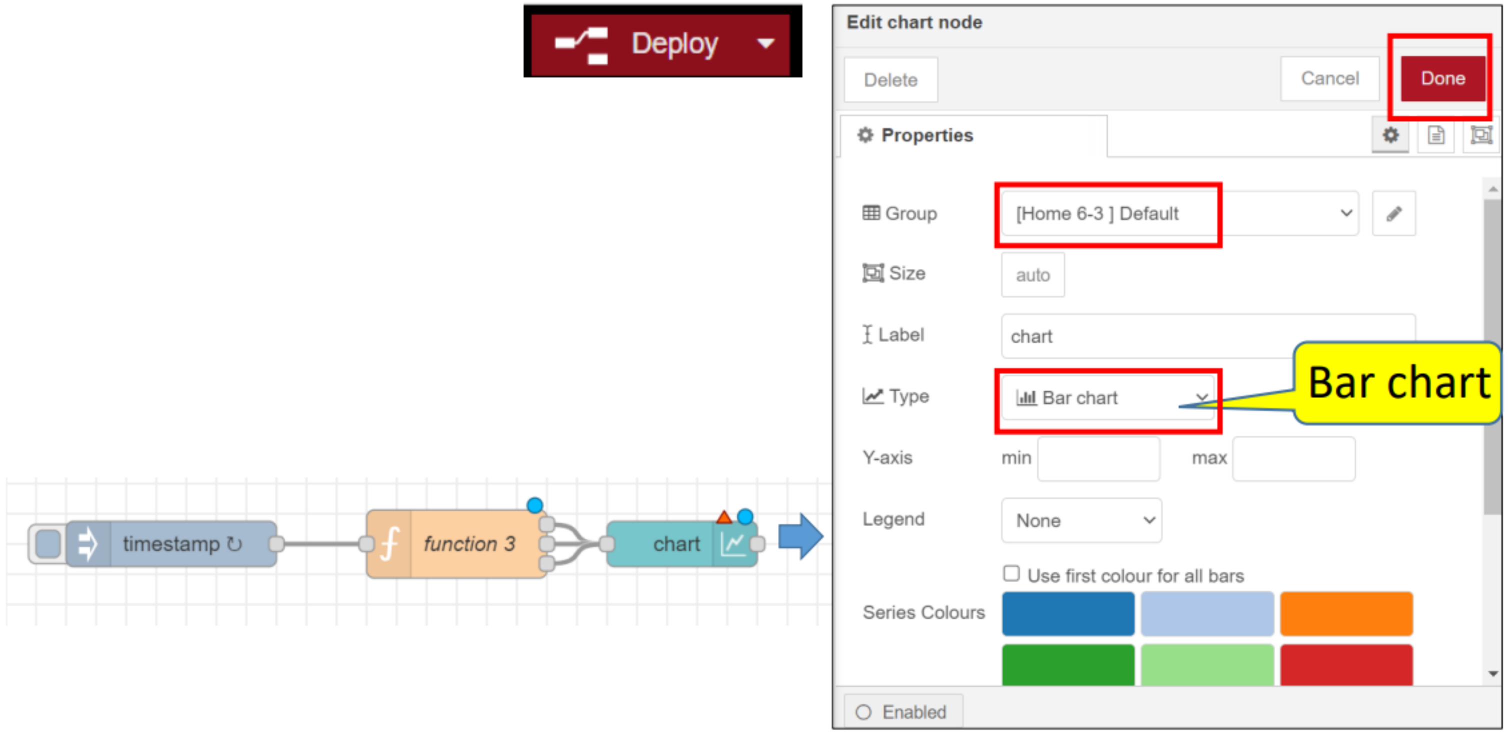


Edit function node

Name: function 3

Outputs: 3

# Add chart



Edit chart node

Group [Home 6-3 ] Default

Size auto

Label chart

Type Bar chart

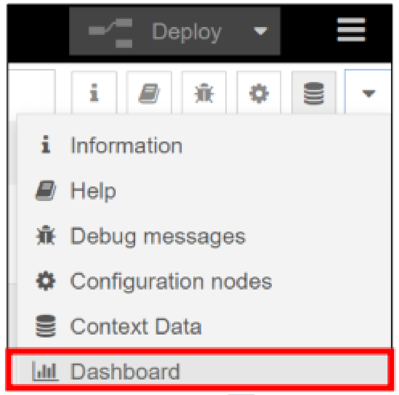
Y-axis min max

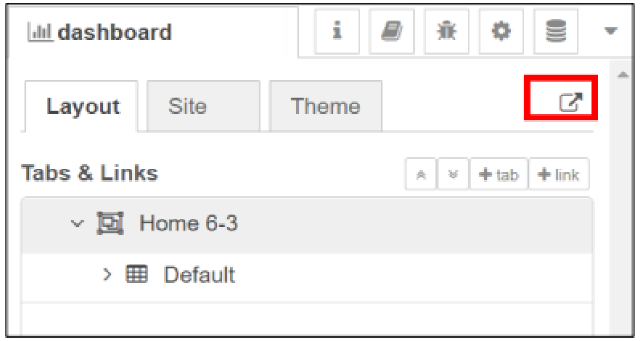
Legend None

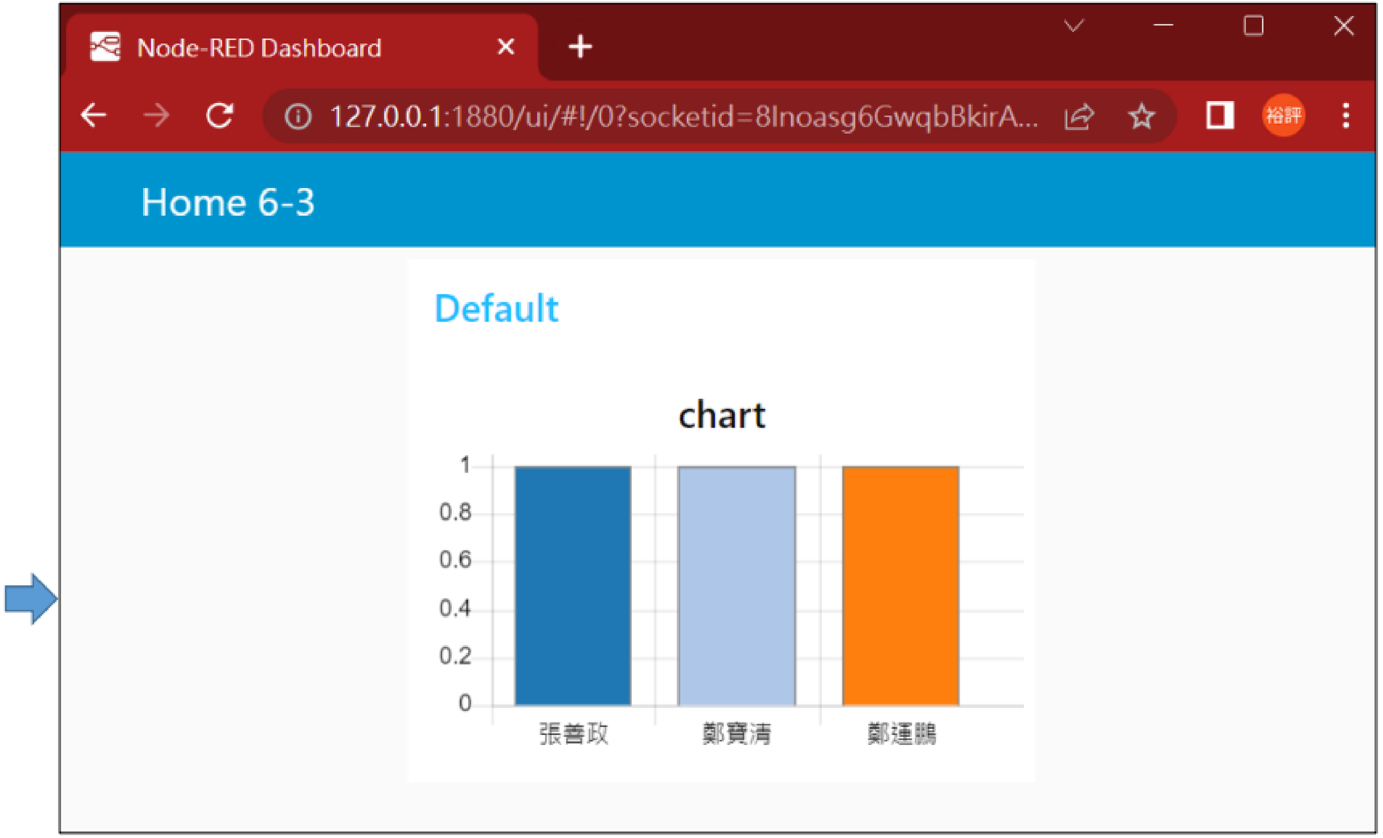
☐ Use first colour for all bars

Series Colours

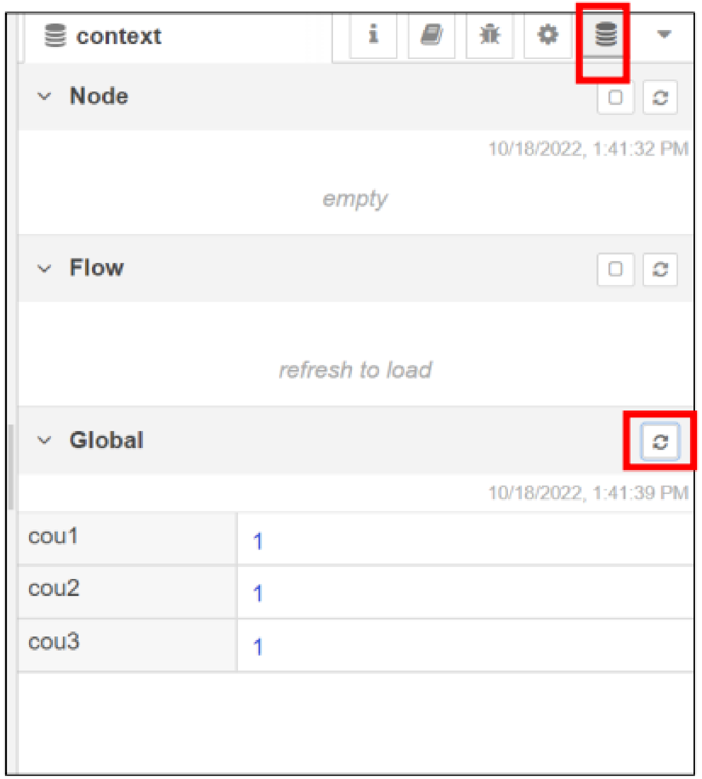
Bar chart


Deploy
Information
Help
Debug messages
Configuration nodes
Context Data
Dashboard


dashboard
Layout
Site
Theme
Tabs & Links
tab
link
Home 6-3
Default


Node-RED Dashboard
127.0.0.1:1880/ui/#!/0?socketid=8lnoasg6GwqbBkirA...
裕評
Home 6-3
Default
chart
1
0.8
0.6
0.4
0.2
0
張善政
鄭寶清
鄭運鵬

# Context data



Node

10/18/2022, 1:41:32 PM

*empty*

Flow

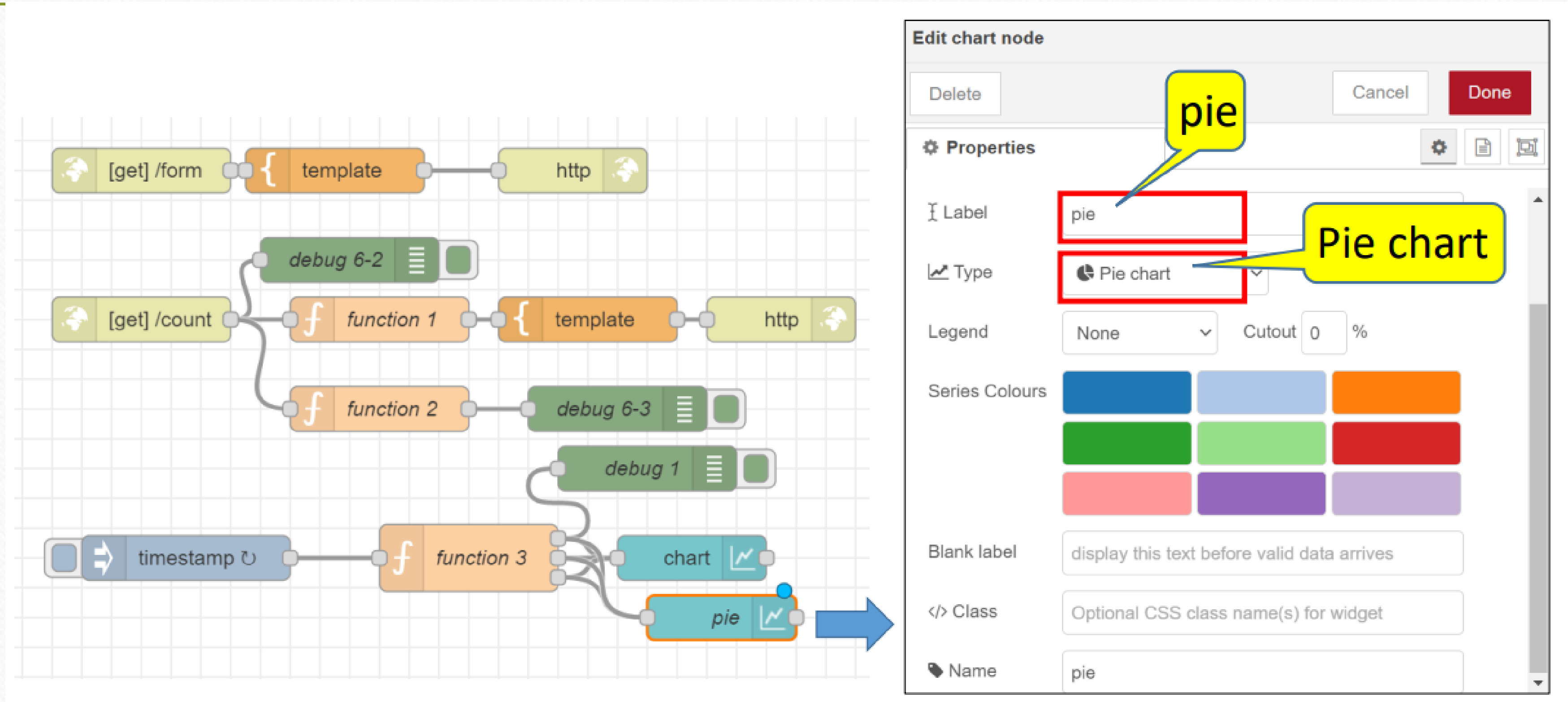
*refresh to load*

Global

10/18/2022, 1:41:39 PM

| | |
|---|---|
| cou1 | 1 |
| cou2 | 1 |
| cou3 | 1 |

# add another chart node



Edit chart node

Delete Cancel Done

Properties

Label: pie

Type: Pie chart

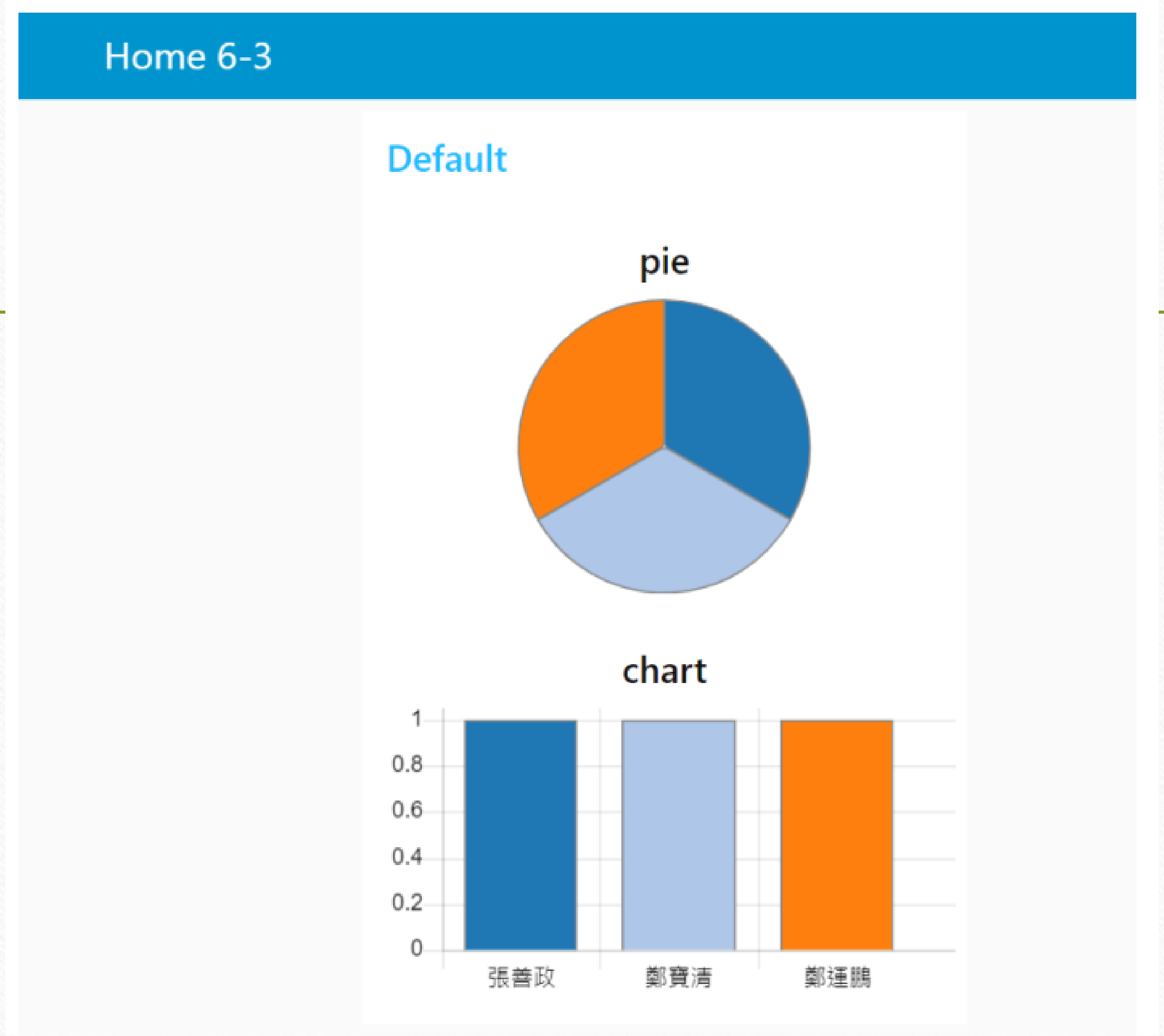
Legend: None   Cutout: 0 %

Series Colours

Blank label: display this text before valid data arrives

Class: Optional CSS class name(s) for widget

Name: pie



Home 6-3

Default

pie

chart

# Homework 4-3

- Design a voting system for four candidates.