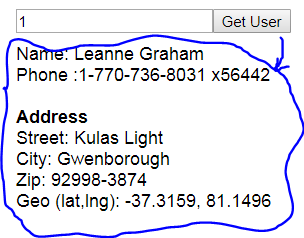
## Dynamic UI manipulation using data obtained via fetch

*Take this as a small, very first “getting started” tutorial, on how to use the* ***fetch-API*** *to fetch data*

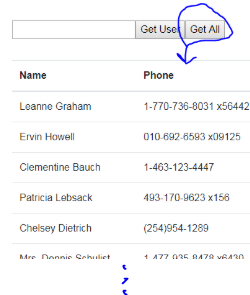
Enter this URL in a browser and observe the result: <https://jsonplaceholder.typicode.com/users/1>

What is it, you get back?

Change the number at the end of the URL to any number <= 10 and observe the result.

**1)** Implement a page, as sketched in this figure, which should fetch the requested user, and render his data.

See hint below before you start.



**2)** Enter this URL in a browser and observe the result: <https://jsonplaceholder.typicode.com/users>

Use this URL, and add a new button to the page as sketched in this figure.

When pressed, it should fetch all persons and render name + phone in a table

**Hints:**

Communication with a REST/JSON-based backend, is what we will do, almost on a daily basis for the rest of the semester. So for this flow just accept the hints below as something you do. *Details and much more info will follow in flow-2*.

The only conceptual difference between this exercise and the other DOM-exercises is that data is fetched from a remote endpoint.

* For the first part you need to create the URL like this:

let url = "[https://jsonplaceholder.typicode.com/users/](https://jsonplaceholder.typicode.com/users/1)" **+ ID-FROM-INPUT**

* For the second part, there are no arguments, so you just use this url each time “Get All” is pressed: <https://jsonplaceholder.typicode.com/users>

This is how you fetch the data (using the **url** variable declared above)

fetch(**url**)

.then(res => res.json()) //in flow1, just do it

.then(data => {

// Inside this callback, and only here, the response data is available

console.log("data",data);

/\* data now contains the response, converted to JavaScript

Observe the output from the log-output above

Now, just build your DOM changes using the data

})