Webudvikling Frontend

JavaScript, Forms & Intro to UI Frameworks

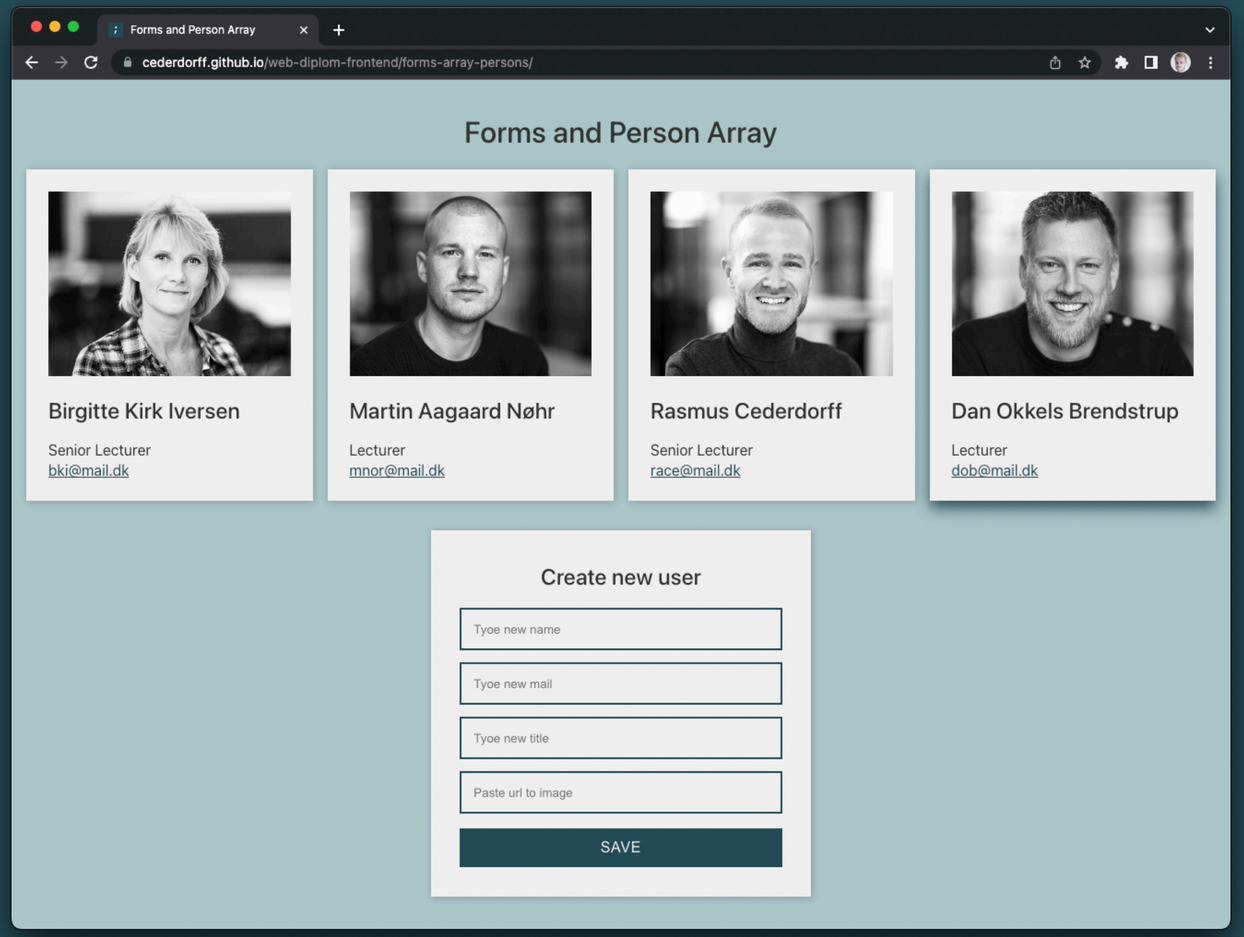
Webudvikling

kea
KØBENHAVNS ERHVERVSAKADEMI

Agenda

- Person Array & JS-koncepter
- Forms, Events & Submit
- Functions & Create New Person
- Evt Intro to Bootstrap

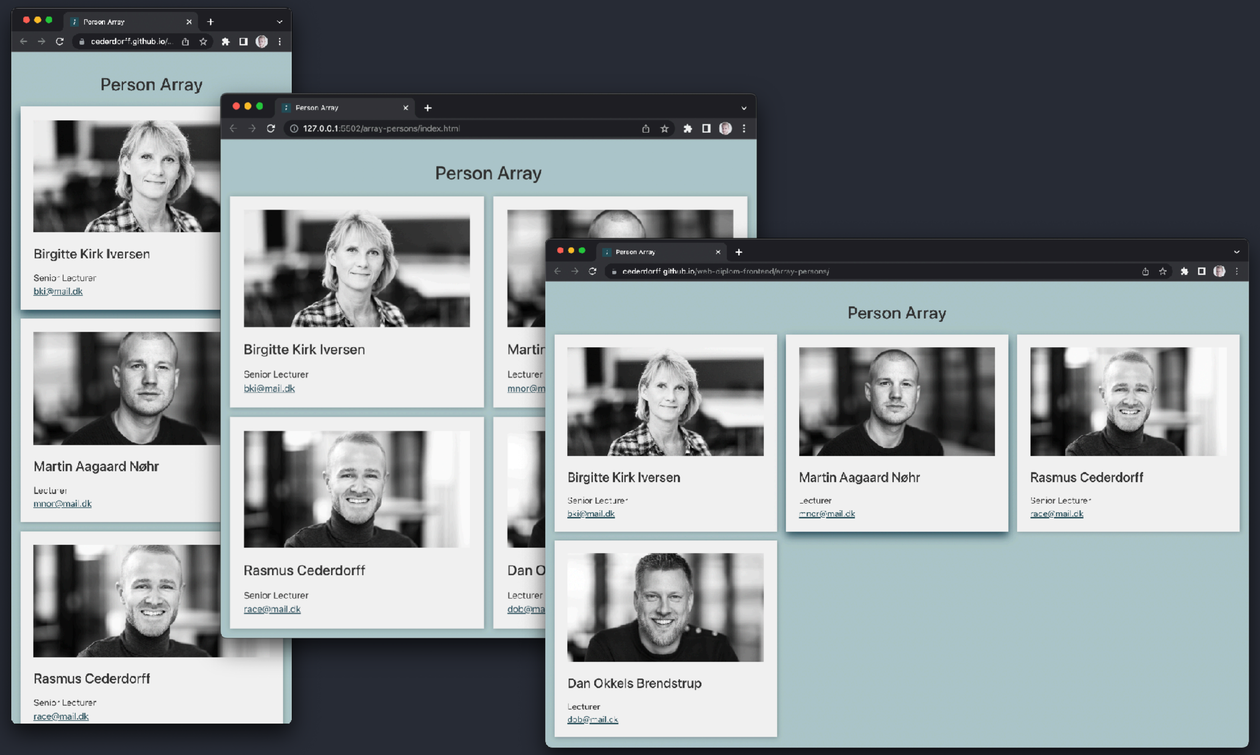
Todays End Goal



https://cederdorff.github.io/web-diplom-frontend/forms-array-persons/

Person Array with CSS Grid

https://race.notion.site/Person-Array-with-CSS-Grid-667a504e43d14c17a8a6f2d1c4a4975c

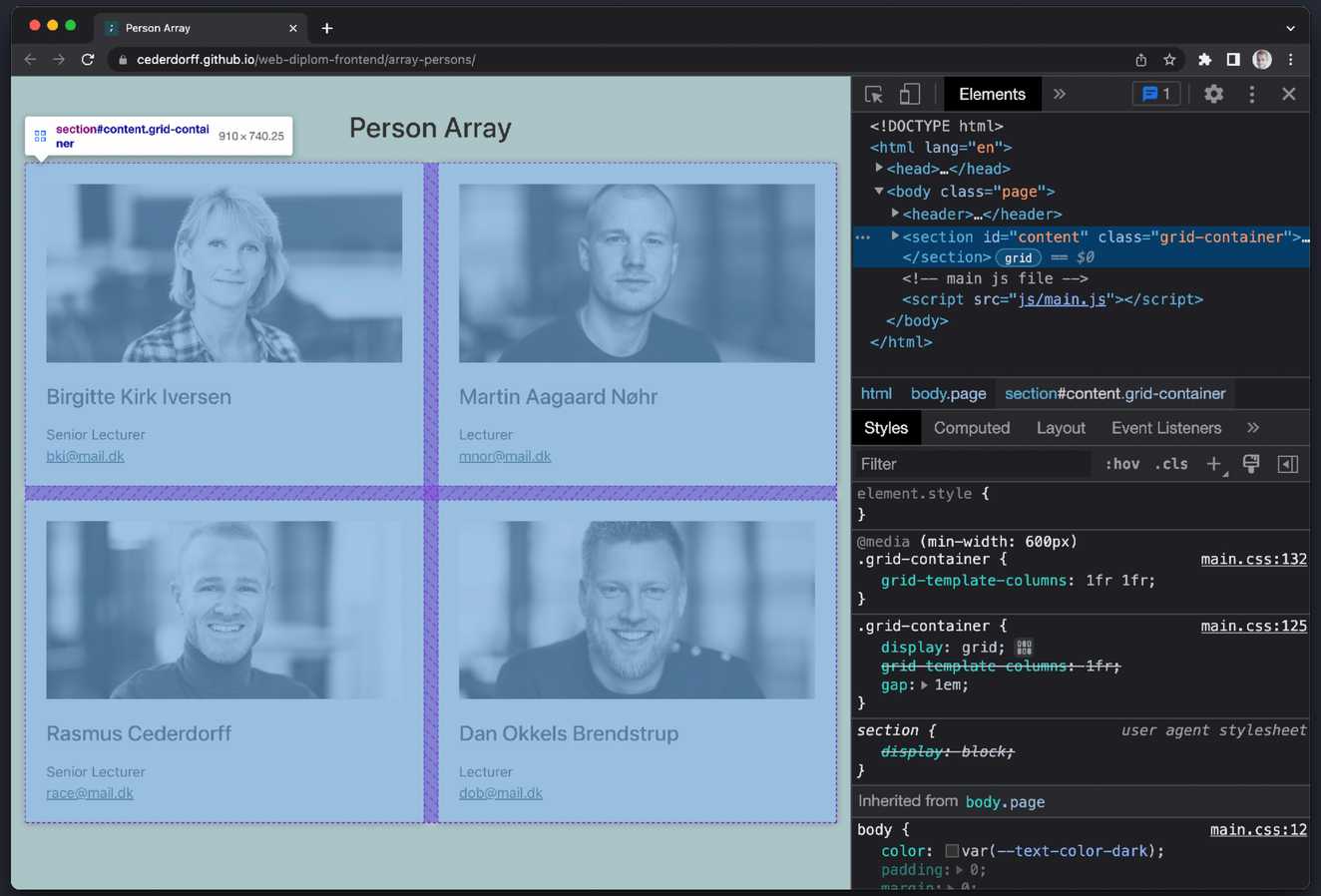


- Variables, Objects & Arrays
- Loops & For-of-loop
- Template String
- DOM-Manipulation

- If/else
- Improvements, naming & best practice

array-persons

CSS Grid




```html
<section id="content" class="grid-container"></section>
```

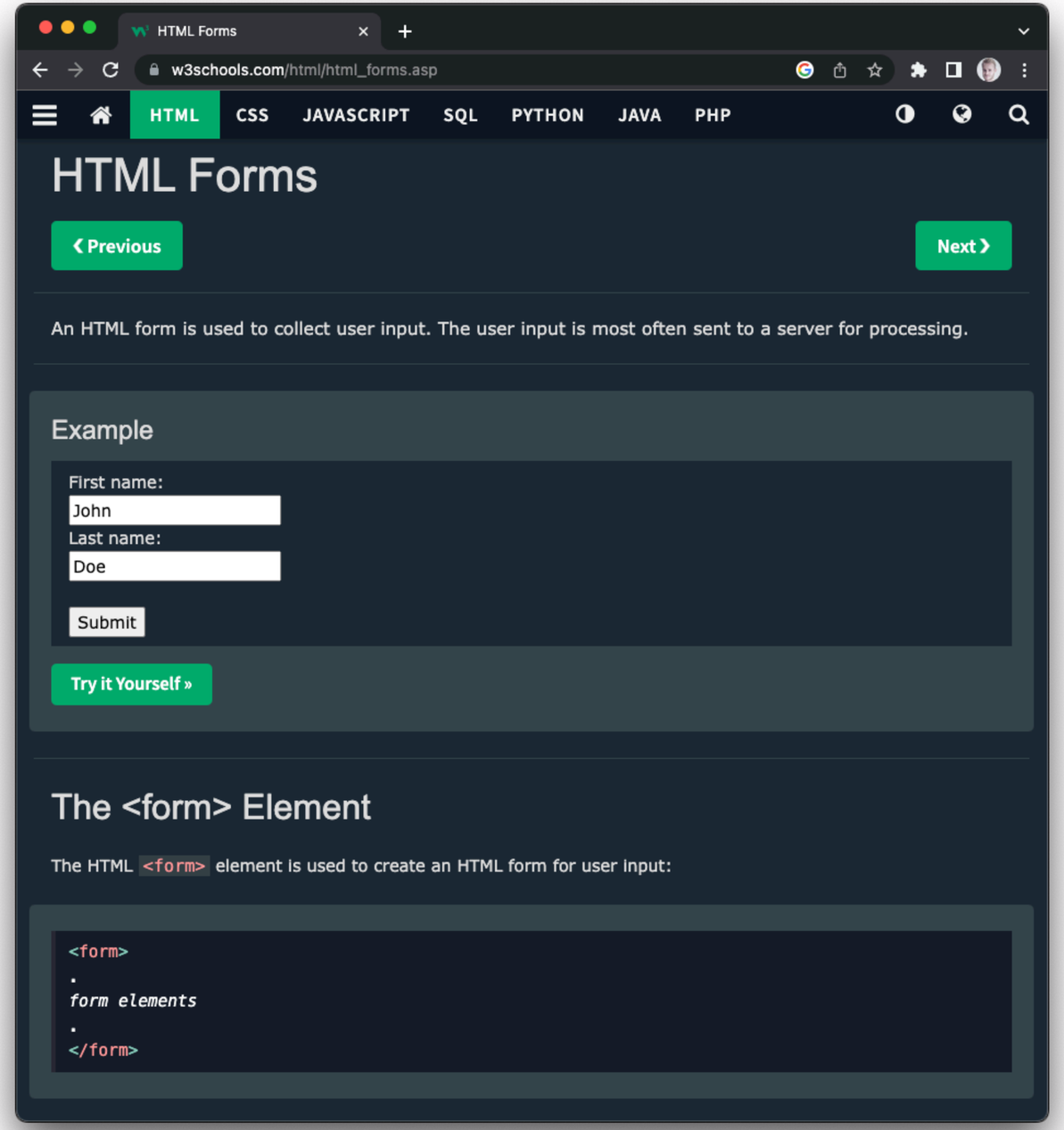
```css
/* ---------- grid container styling ---------- */
.grid-container {
    display: grid;
    grid-template-columns: 1fr;
    gap: 1em;
}

@media (min-width: 600px) {
    .grid-container {
        grid-template-columns: 1fr 1fr;
    }
}

@media (min-width: 992px) {
    .grid-container {
        grid-template-columns: 1fr 1fr 1fr;
    }
}
```

Forms

Collect user inputs

In HTML we use the `<form>` element

with `<input>` fields to create forms



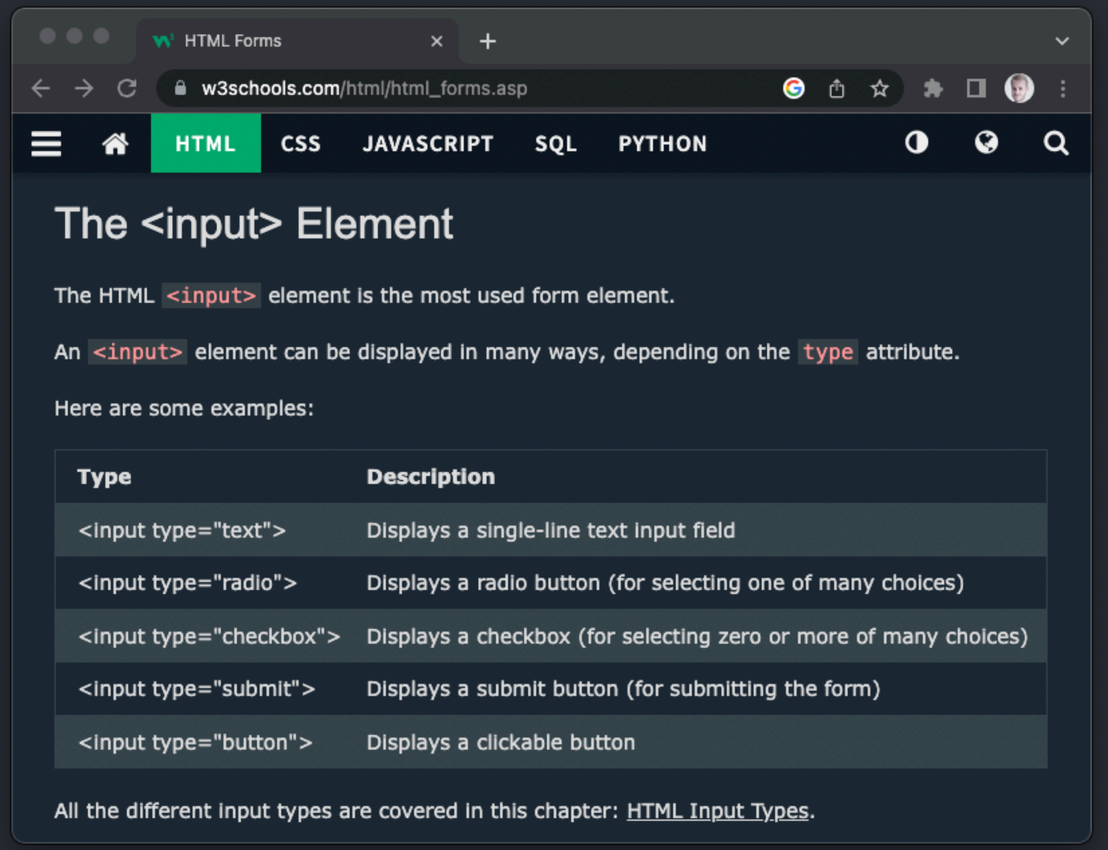
https://www.w3schools.com/html/html_forms.asp

Inputs

Mostly used inside the `<form>` element

`<input>` fields can be displayed in different ways with different types.

See all types: HTML Input Types




```
<form>
  <input type="text" name="name" placeholder="Type new name">
  <input type="email" name="mail" placeholder="Type new mail">
  <input type="text" name="title" placeholder="Type new title">
  <input type="url" name="url" placeholder="Paste url to image">
  <button>Save</button>
</form>
```

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

when you ask Rasmus
for help and he says
"Read documentation"

Forms & Inputs

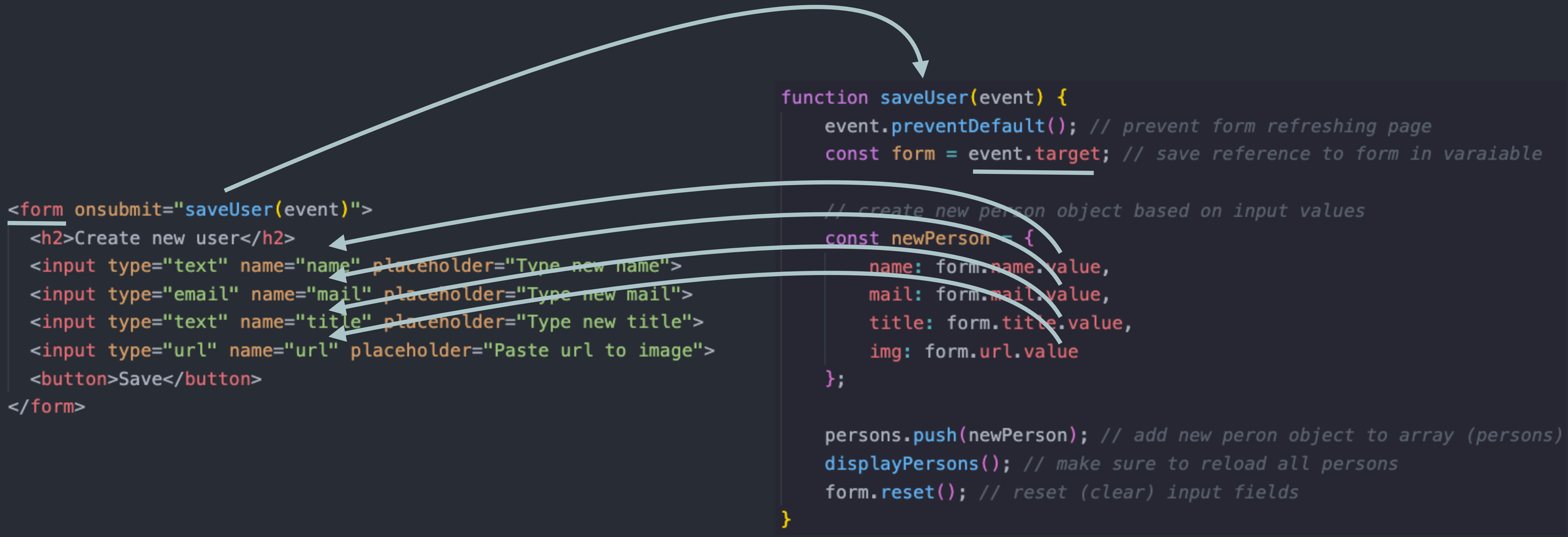
```html
<form onsubmit="saveUser(event)">
  <h2>Create new user</h2>
  <input type="text" name="name" placeholder="Type new name">
  <input type="email" name="mail" placeholder="Type new mail">
  <input type="text" name="title" placeholder="Type new title">
  <input type="url" name="url" placeholder="Paste url to image">
  <button>Save</button>
</form>
```

HTML

```javascript
function saveUser(event) {
    event.preventDefault(); // prevent form refreshing page
    const form = event.target; // save reference to form in varaiable

    // create new person object based on input values
    const newPerson = {
        name: form.name.value,
        mail: form.mail.value,
        title: form.title.value,
        img: form.url.value
    };

    persons.push(newPerson); // add new peron object to array (persons)
    displayPersons(); // make sure to reload all persons
    form.reset(); // reset (clear) input fields
}
```

JavaScript

Forms & Inputs



```html
<form onsubmit="saveUser(event)">
  <h2>Create new user</h2>
  <input type="text" name="name" placeholder="Type new name">
  <input type="email" name="mail" placeholder="Type new mail">
  <input type="text" name="title" placeholder="Type new title">
  <input type="url" name="url" placeholder="Paste url to image">
  <button>Save</button>
</form>
```

```javascript
function saveUser(event) {
    event.preventDefault(); // prevent form refreshing page
    const form = event.target; // save reference to form in varaiable

    // create new person object based on input values
    const newPerson = {
        name: form.name.value,
        mail: form.mail.value,
        title: form.title.value,
        img: form.url.value
    };

    persons.push(newPerson); // add new peron object to array (persons)
    displayPersons(); // make sure to reload all persons
    form.reset(); // reset (clear) input fields
}
```

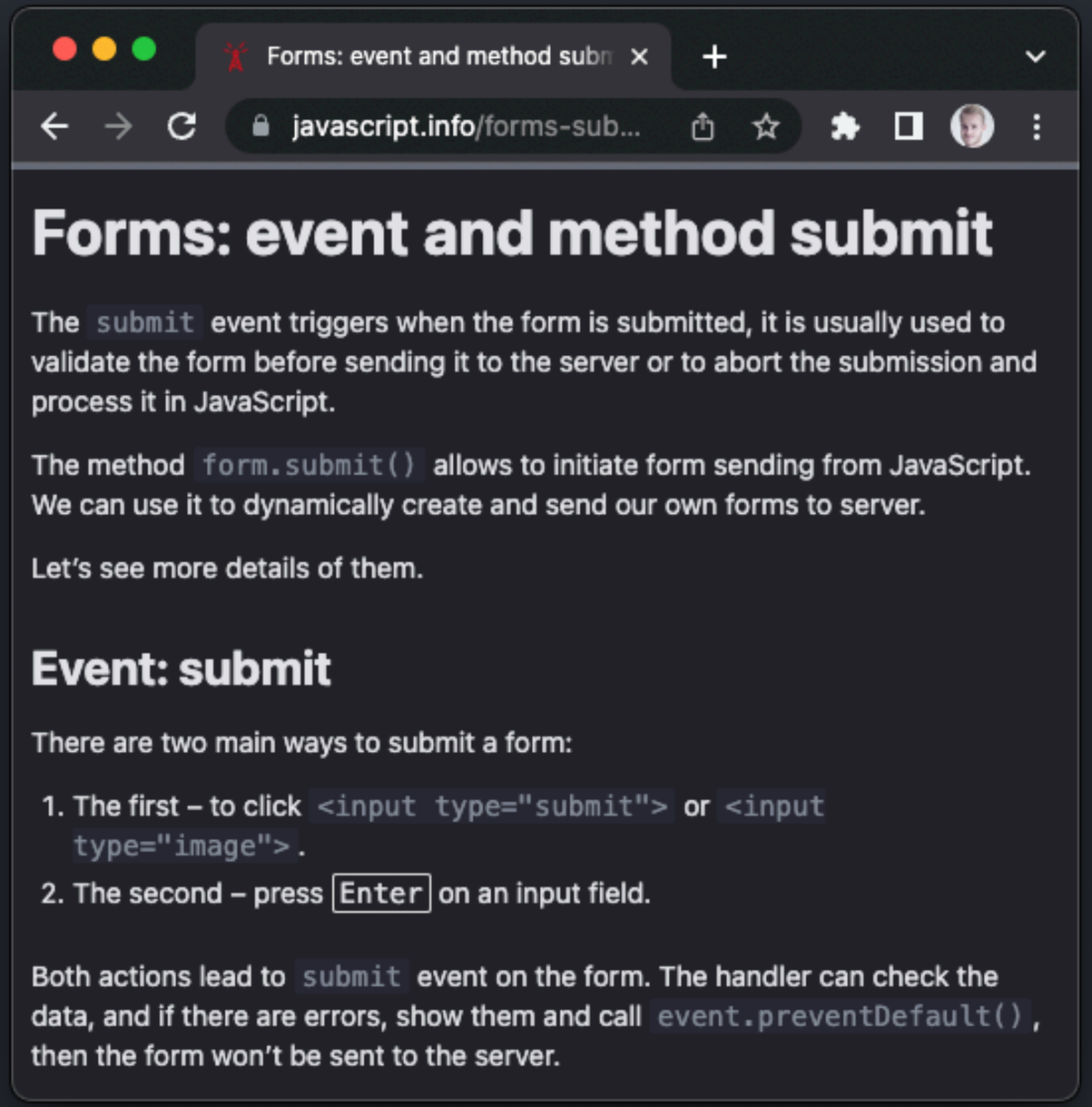
HTML

JavaScript

Submit event

```html
<form onsubmit="saveUser(event)">
  <h2>Create new user</h2>
  <input type="text" name="name" placeholder="Type new name">
  <input type="email" name="mail" placeholder="Type new mail">
  <input type="text" name="title" placeholder="Type new title">
  <input type="url" name="url" placeholder="Paste url to image">
  <button>Save</button>
</form>
```

```js
function saveUser(event) {
    event.preventDefault(); // prevent form refreshing page
    const form = event.target; // save reference to form in varaiable

    // create new person object based on input values
    const newPerson = {
        name: form.name.value,
        mail: form.mail.value,
        title: form.title.value,
        img: form.url.value
    };

    persons.push(newPerson); // add new peron object to array (persons)
    displayPersons(); // make sure to reload all persons
    form.reset(); // reset (clear) input fields
}
```

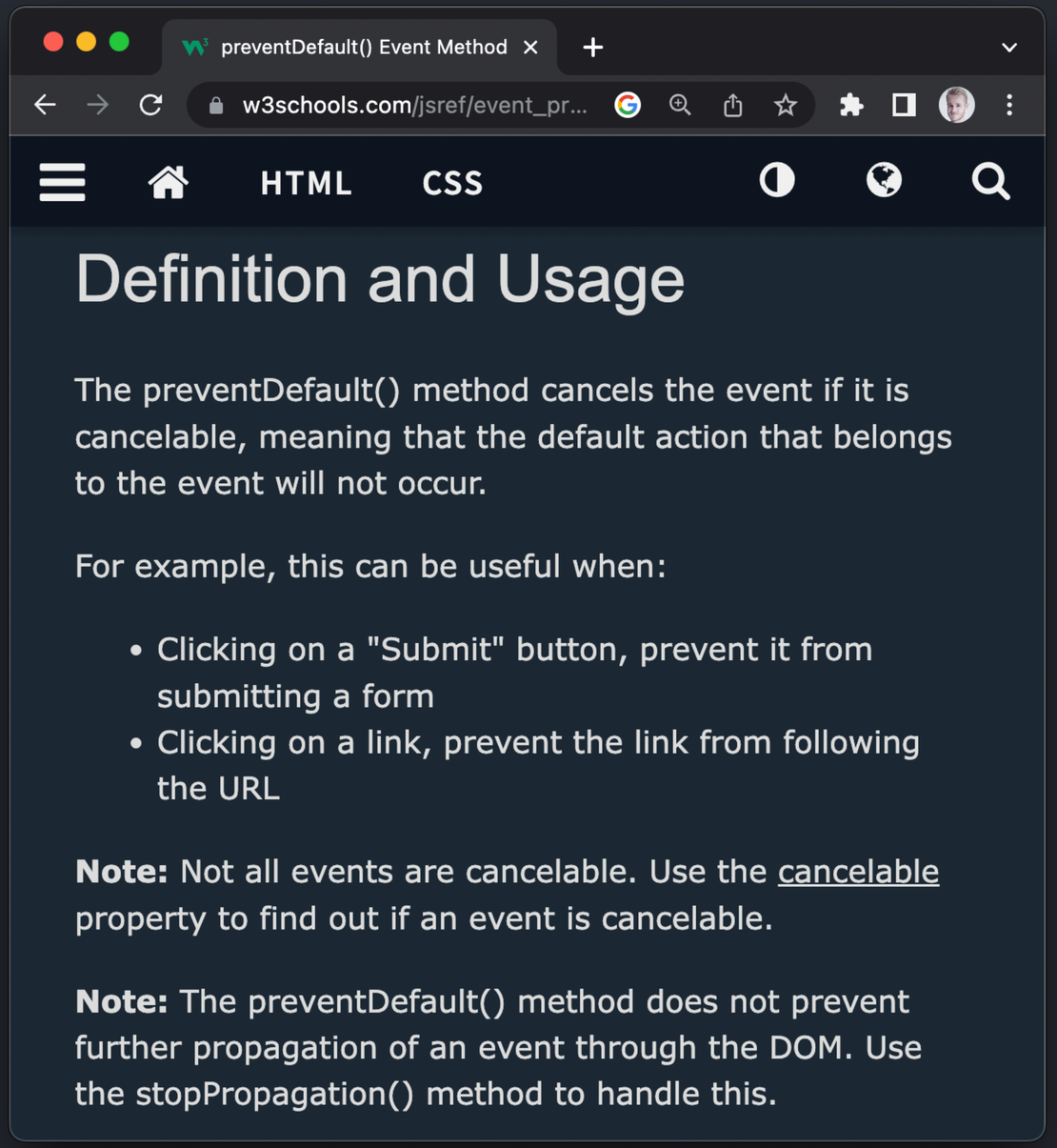


https://javascript.info/forms-submit#event-submit

Prevent Default

```html
<form onsubmit="saveUser(event)">
  <h2>Create new user</h2>
  <input type="text" name="name" placeholder="Type new name">
  <input type="email" name="mail" placeholder="Type new mail">
  <input type="text" name="title" placeholder="Type new title">
  <input type="url" name="url" placeholder="Paste url to image">
  <button>Save</button>
</form>
```

```js
function saveUser(event) {
    event.preventDefault(); // prevent form refreshing page
    const form = event.target; // save reference to form in varaiable

    // create new person object based on input values
    const newPerson = {
        name: form.name.value,
        mail: form.mail.value,
        title: form.title.value,
        img: form.url.value
    };

    persons.push(newPerson); // add new peron object to array (persons)
    displayPersons(); // make sure to reload all persons
    form.reset(); // reset (clear) input fields
}
```



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

Events & EventListener

Assign an onclick event to a button element:

```
<button onclick="displayDate()">Try it</button>
```

Assign an onclick event to a button element:

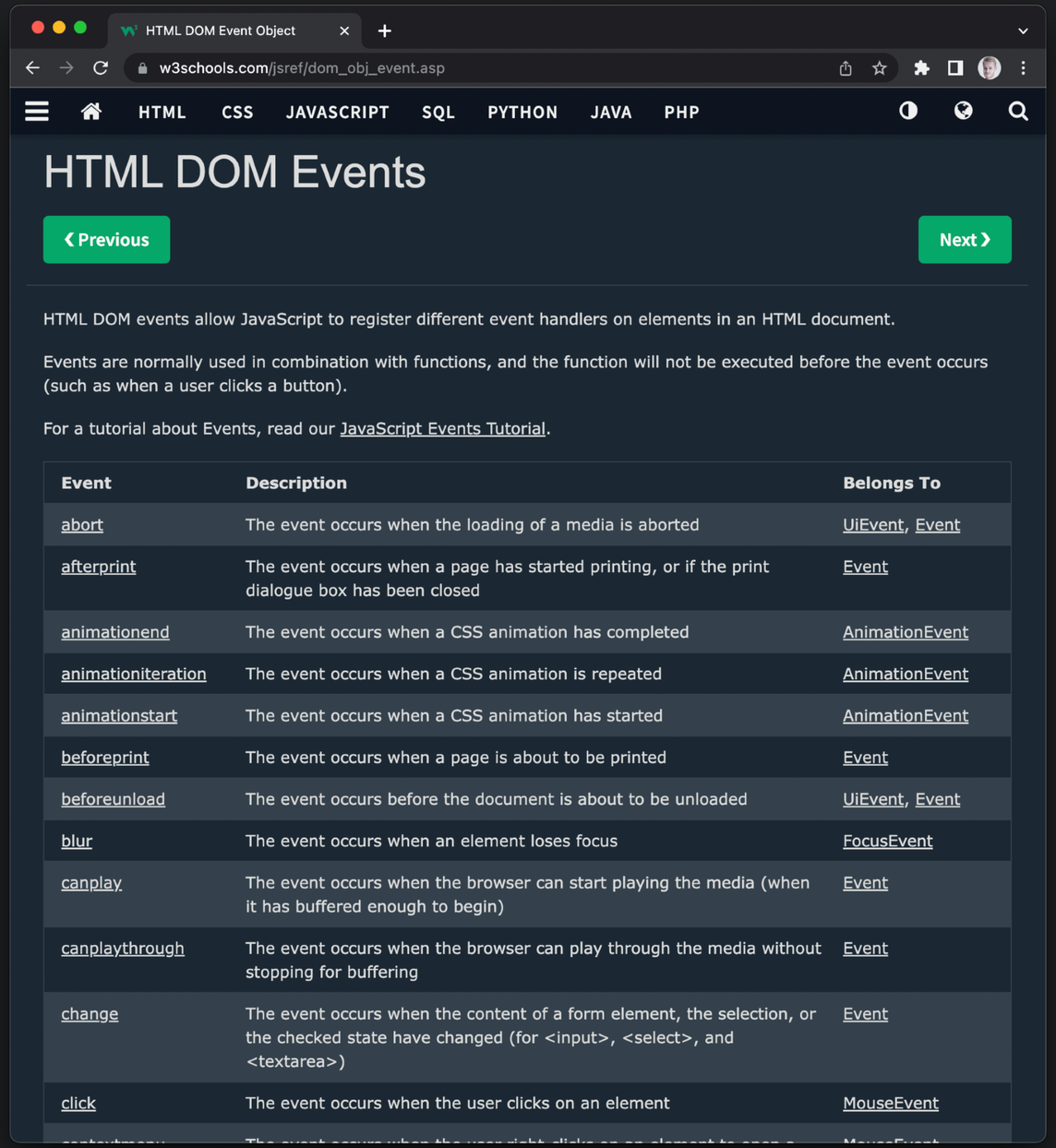
```
<script>
document.getElementById("myBtn").onclick = displayDate;
</script>
```

Add an event listener that fires when a user clicks a button:

```
document.getElementById("myBtn").addEventListener("click", displayDate);
```

```
function displayDate() {
    document.getElementById("demo").innerHTML = Date();
}
```

https://www.w3schools.com/js/js_htmldom_events.asp
https://www.w3schools.com/js/js_htmldom_eventlistener.asp



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

UI FRAMEWORK

COLLECTION OF HTML, CSS & JAVASCRIPT

STYLING, UI ELEMENTS & ‘READY TO USE’ COMPONENTS

STANDARDISED UI

TYPOGRAPHY, GRID SYSTEM, RESPONSIVE UTILITIES, NAVIGATION, ICONS, FONTS, UI ELEMENTS, ETC.


Onsen UI


Semantic UI
User Interface is the language of the web.


Framework
7
Framework7


Skeleton
A dead simple, responsive boilerplate.


UIkit


MATERIAL-UI


Foundation
Start here, build everywhere.


P
CSS


IONIC

WHY USE A UI FRAMEWORK?

PREBUILD LAYOUT, STYLING & FUNCTIONALITY

SIMPLYFIES & SPEEDS UP THE DEV PROCESS

FOCUS ON REQUIREMENTS & USER-NEEDS

BROWSER COMPABILITY

STRUCTURE, SEMANTICS & STANDARDIZATION

DEMO, TEMPLATES & THEMES

RAPID PROTOTYPING

MINIMAL CUSTOM STYLING

WHY USE A UI FRAMEWORK?

BEST PRACTICE

UI & UX

WELL-TESTED

PROTOTYPING

UI FRAMEWORK

BEHAVIOR

.JS

.CSS

UI ELEMENTS

FONTS

ICONS

THEMES

TEMPLATES

PROJECT TEMPLATE

YOUR WEB APP

INDEX.HTML

MAIN.CCS

MAIN.JS

BOOTSTRAP

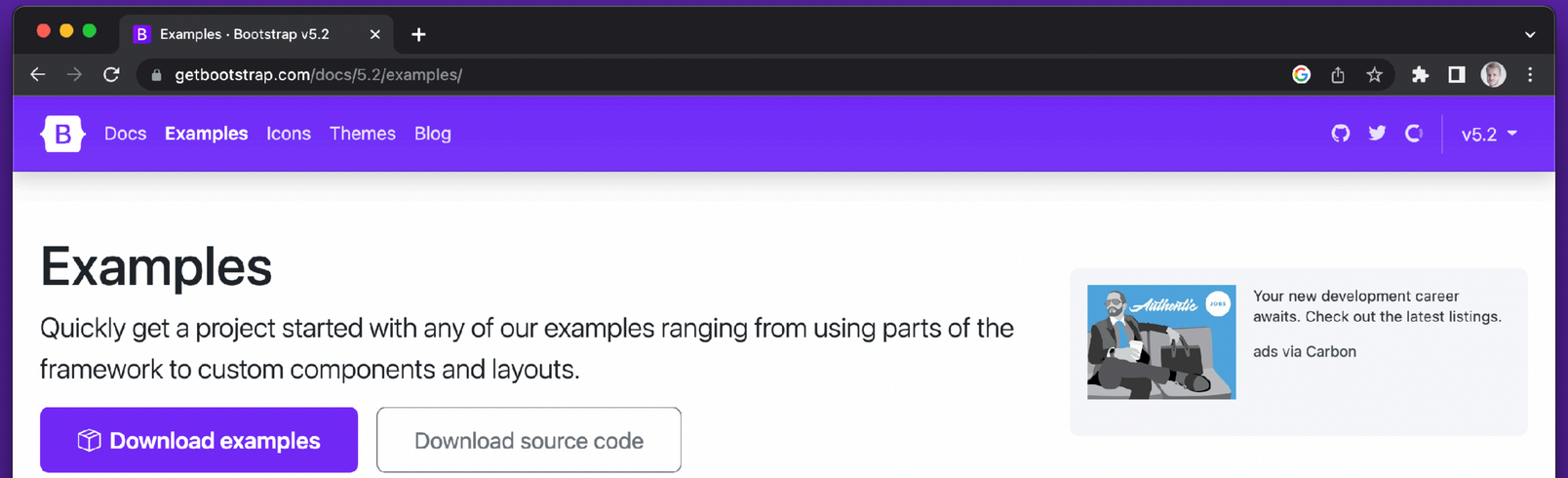
THE MOST POPULAR FRAMEWORK FOR BUILDING RESPONSIVE &
MOBILE-FIRST WEB APPS WITH HTML, CSS & JAVASCRIPT

BOOTSTRAP

CSS CLASSES, COMPONENTS & FUNTIONALITY

PREBUILD CSS & STYLING

TYPOGRAPHY, COLORS, STYLES, CONTAINERS, ETC

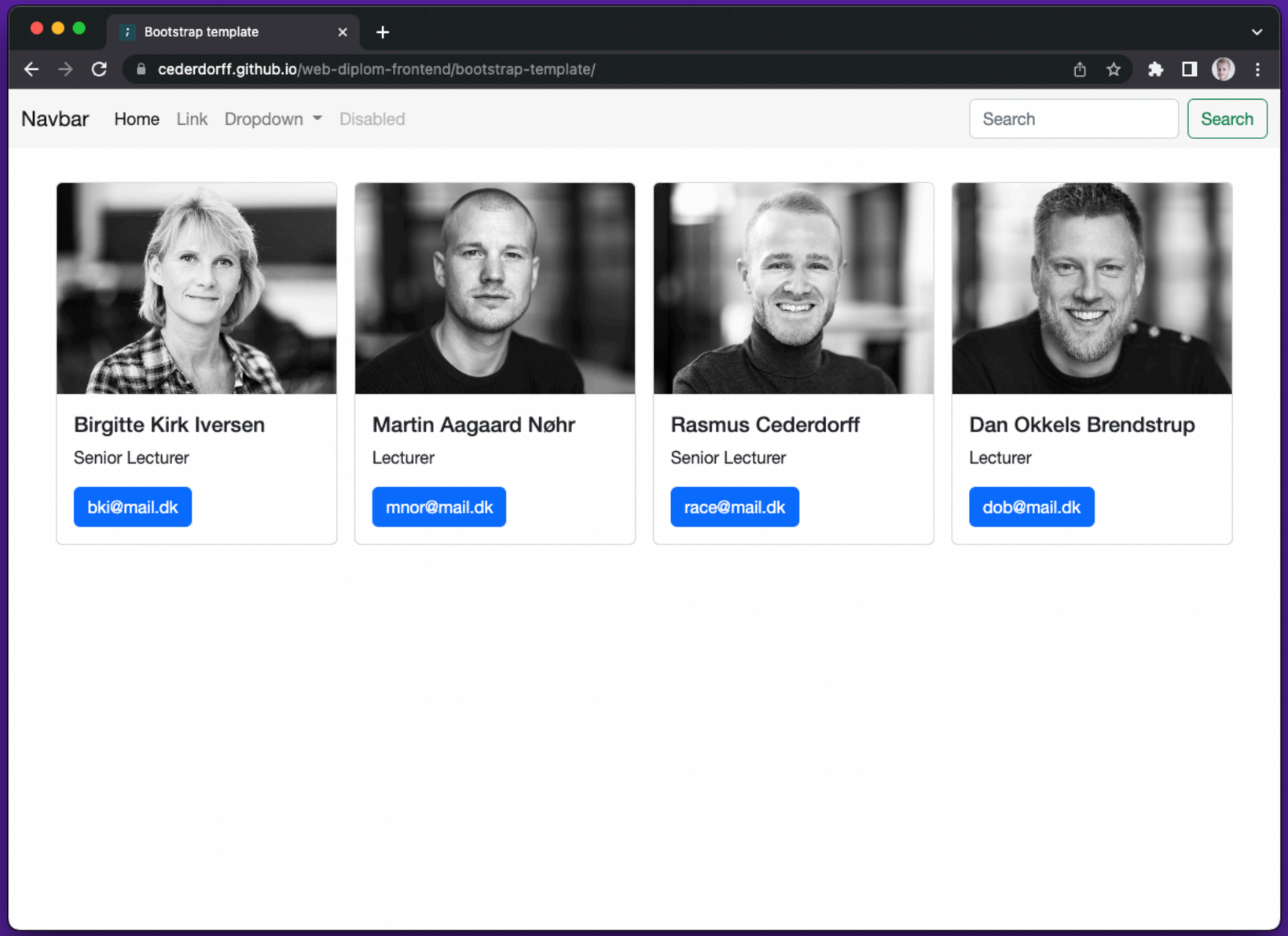


https://getbootstrap.com/docs/5.2/examples/

USE THE DOCS

... COPY AND PASTE

https://getbootstrap.com/docs/5.2/getting-started/introduction/



https://cederdorff.github.io/web-diplom-frontend/bootstrap-template/