Introcution to JavaScript

Nils Twelker

March 2023

What learned we last Week?

- Objects `let person = {name: "Dan", age: 43}`
- Accessing Properties `person.name // "Dan"`
- Adding Properties `person.height = 1.80`
- Removing Properties `delete person.age`
- “this”, “in” and “for in” keywords
- Object references and Garbage Collection
- Methods on Primitives `"Hello World".length // 11`
- Arrays Methods `[2, 3, 1, 4].sort() // [1, 2, 3, 4]`

Goals of this week

- Basic introduction to HTML
- Basic introduction to CSS
- Developer Tools
- JavaScript in the Browser
- DOM (Document Object Model)
- Searching & Manipulating the DOM
- Adding / Removing Elements
- Events

HTML (Hyper Text Markup Language)

```
<!DOCTYPE html>
<html>
    <head>
        <title>My Website</title>
    </head>
    <body>
        <h1>My Website</h1>
    </body>
</html>
```

HTML Elements

```
<h1 id="title" class="heading large">My Website</h1>
```

`h1` - Type of the element (tag)

`id` - Only one id per element

`class` - Multiple `classes per element` and `elements per class`

`My Website` - Content of the element

CSS (Cascading Style Sheets)

```
body { // tag selector
    background-color: black;
}
#title { // id selector
    color: red;
}
.large { // class selector
    font-size: 20px;
}
```

CSS (Cascading Style Sheets)

```
<!DOCTYPE html>
<html>
    <head>
        <title>My Website</title>
        <style> h1 { color: red; }</style>
    </head>
    <body>
        <h1>My Website</h1>
    </body>
</html>
```

Developer Tools

- Press F12 or Ctrl+Shift+I
- `Elements / Inspector` Tab
  - See the current HTML & CSS
  - Change the HTML & CSS
- `Console` Tab
  - Execute JavaScript
  - See Errors

JavaScript in the Browser

```
<!DOCTYPE html>
<html>
    <head><title>My Website</title></head>
    <body>
        <h1 id="title">My Website</h1>
        <script>
            document.getElementById("title").style.color = "red";
        </script>
    </body>
</html>
```

DOM (Document Object Model)

```
<!DOCTYPE html>
<html>
    <head>
        <title>My Website</
title>
    </head>
    <body>
        <h1>My Website</h1>
    </body>
</html>
```

```
document.body // <body>
document.body.children// [<h1>]
```

Searching the DOM

```
<h1 id="title" class="heading large">My Website</h1>
<button id="button">Click me!</button>
<span>First Text</span>
<span>Second Text</span>
```

```
document.getElementById("title") // <h1>
document.getElementsByClassName("heading") // [<h1>]
document.getElementsByTagName("h1") // [<h1>]

document.querySelector("#button") // <button>
document.querySelector(".large") // <h1>
document.querySelectorAll("span") // [<span>, <span>]
```

Manipulating the DOM

```
<h1 id="title">My Website</h1>
```

```
let myTitle = document.getElementById("title") // <h1>

console.log(myTitle.innerHTML) // "My Website"

myTitle.innerHTML = "My new Website"
myTitle.style.color = "red"
myTitle.style.fontSize = "20px"
```

Adding Elements

```
<h1 id="title">My Website</h1>
```

```
let myTitle = document.getElementById("title") // <h1>

let myText = document.createElement("span") // <span>
myText.innerHTML = "This is some text"

document.body.appendChild(myText)
```

```
<h1 id="title">My Website</h1>
<span>This is some text</span>
```

Removing Elements

```
<h1 id="title">My Website</h1>
<span>This is some text</span>
```

```
let myText = document.querySelector("span") // <span>

document.body.removeChild(myText)
```

```
<h1 id="title">My Website</h1>
```

Events

```
<h1 id="title">My Website</h1>
<button id="button">Click me!</button>
```

```
let myButton = document.getElementById("button") // <button>

myButton.onclick = function () {
    console.log("Button clicked!")
}
```

Events

```
<h1 id="title">My Website</h1>
<button id="button">Click me!</button>
```

```
let myButton = document.getElementById("button") // <button>

myButton.addEventListener("click", function () {
    console.log("Button clicked!")
})
myButton.addEventListener("click", function () {
    console.log("Button clicked again!")
})
```

Tasks and Points

Goal is to get 100 Points.

- `array-methods` (25 Points)
- `basic-objects` (25 Points)
- `in-keyword` (25 Points)
- `object-references` (25 Points)
- `graph-calculator` (50 Points)
- `login-system` (50 Points)