

Introcution to JavaScript

Nils Twelker

March 2023

What learned we last Week?

- HTML Basics `<h1>My Title</h1>`
- CSS Basics `h1 { color: red; }`
- Developer Tools (Pressing F12 in Browser)
- DOM (Document Object Model) `document.body`
- Searching the DOM `document.getElementById("title")`
- Manipulating the DOM `myElement.style.color = "red"`
- Adding / Removing Elements `document.createElement("h1")` +
- Events `myElement.onclick = () => { ... }`

Goals of this week

- More about Events
 - More Events
 - Bubbling
 - Dispatching Events
- Displaying and Interacting with Data
 - Lists
 - Tables
 - Forms

More Events

- `click` (left mouse button)
- `contextmenu` (right mouse button)
- `dblclick` (double click)
- `mouseover` `mouseout` (hovering)
- `mousemove` (moving mouse)
- `keydown` `keyup` (keyboard)
- `focus` `blur` (focus)
- `submit` (submitting a form)

on<event> Attribute

```
<button onclick="console.log('clicked')">Click me</button>
<input type="text" onkeydown="console.log('key pressed')">

<script>
function enterDiv() {
    console.log("Mouse entered div")
}
</script>

<div onmouseover="enterDiv()"></div>
```

Event Bubbling

```
<div id="root" onclick="console.log('Root')">  
  <div id="c1" onclick="console.log('Child 1')">  
    <div id="c1-1" onclick="console.log('Child 1-1')"></div>  
  </div>  
  <div id="c2" onclick="console.log('Child 2')"></div>  
</div>
```

Event Bubbling

```
<div id="root" onclick="console.log('Root')">  
  <div id="c1" onclick="console.log('Child 1')">  
    <div id="c1-1" onclick="console.log('Child 1-1')">  
    </div>  
  </div>  
  <div id="c2" onclick="console.log('Child 2')">  
  </div>  
</div>
```

Clicking #c1-1 :

Child 1-1
Child 1
Root

Clicking #c2 :

Child 2
Root

Clicking #c1 :

Child 1
Root

event.target

```
<div id="root">  
  <div id="child"></div>  
</div>
```

```
const root = document.getElementById("root")  
// When clickin on #child  
root.onclick = function(event) {  
  console.log(this.id) // "root"  
  console.log(event.target.id) // "child"  
}
```

Stopping bubbling

```
<div id="root">
  <div id="child"></div>
</div>
```

```
const root = document.getElementById("root")
const child = document.getElementById("child")

child.onclick = (event) => {
  event.stopPropagation()
} // #root will not receive event when clicking on #child
root.onclick = () => console.log("root")
```

Dispatching Events

```
<div id="root">  
  <div id="child"></div>  
</div>
```

```
const root = document.getElementById("root")  
const child = document.getElementById("child")  
  
root.onclick = () => console.log("root")  
  
const event = new Event("click")  
child.dispatchEvent(event)
```

Dispatching Custom Events

```
<div id="myElement"></div>
```

```
const myElement = document.getElementById("myElement")
```

```
document.addEventListener("myEvent", (event) => {  
  console.log("myEvent:" + event.detail.message)  
}) // myEvent: Hello World
```

```
let event = new CustomEvent("myEvent", { detail: { message: "Hello World" } })  
myElement.dispatchEvent(event)
```

Displaying Data (Lists)

```
<ul>
  <li>Item A</li>
  <li>Item B</li>
  <li>Item C</li>
</ul>
```

- Item A
- Item B
- Item C

```
<ol>
  <li>Item A</li>
  <li>Item B</li>
  <li>Item C</li>
</ol>
```

1. Item A
2. Item B
3. Item C

Displaying Data (Tables)

```
<table>
  <tr>
    <th>Name</th>
    <th>Age</th>
  </tr>
  <tr>
    <td>John Doe</td>
    <td>54</td>
  </tr>
</table>
```

Name	Age
John Doe	54

Interacting with Data (Forms)

```
<form id="myForm">
  <label for="name">Name</label>
  <input type="text" id="name">
  <br/><br/>
  <label for="age">Age</label>
  <input type="number" id="age">
  <br/><br/>
  <button type="submit">Submit</button>
</form>
```



The image shows a rendered version of the HTML form. It consists of a 'Name' label followed by a text input field, an 'Age' label followed by a number input field with a spinner, and a 'Submit' button below them.

Interacting with Data (Forms)

```
const form = document.getElementById("myForm")

form.onsubmit = (event) => {
  event.preventDefault()
  const name = document.getElementById("name").value
  const age = document.getElementById("age").value
  console.log(name, age)
}
```

Tasks and Points

Goal is to get 100 Points.

- adding-removing (35 Points)
- basic-events (35 Points)
- hello-browser (20 Points)
- manipulating-dom (35 Points)
- searching-elements (35 Points)
- to-do-list (65 Points)