

JavaScript Basics

pre-hack Tutorial

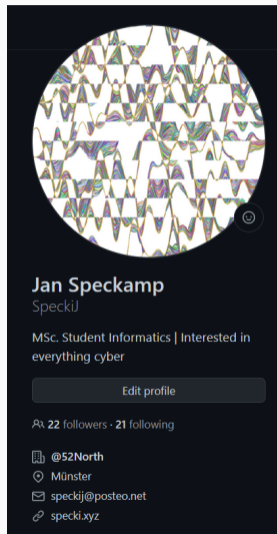
Jan Speckamp

31. März 2022

52°North Spatial Information Research GmbH & Westfälische Wilhelms-Universität Münster

Jan Speckamp

- Student MSc. Informatik
- Vorher: BSc. Geoinformatik @ifgi
- Java Backend Developer @52North (seit 2017)
- Vorher IT-Support & Tutor @ifgi (bis 2019)
- @speckij auf allen Plattformen



Javascript Basics

- Was ist Javascript?
- Wie benutze ich Javascript?
- Crash-Kurs Syntax
- Besonderheiten

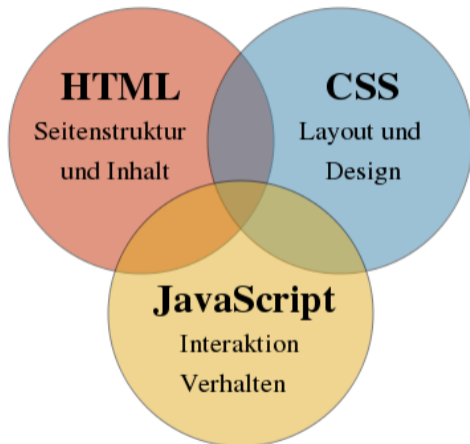
Was ist Javascript?

- Programmiersprache
- Eingesetzt auf 97% aller Webseiten
- de-facto Standard für interaktive Webseiten
- Multi-Paradigmen
- Vormalig nur Webseiten, mittlerweile auch andere Anwendungen
 - NodeJS (Geosoftware I & II)



Quelle: [1], MIT License

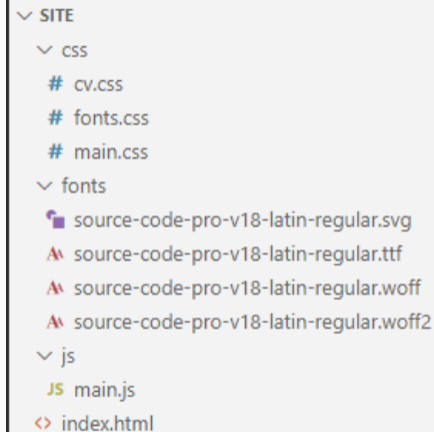
Was ist Javascript?



Quelle: [2], CC-BY-SA 3.0

Wie benutze ich Javascript?

- Interpretierte Sprache
→ Dateien mit Text (Code)
- Verteilung auf mehrere Dateien & Ordner zwecks Übersichtlichkeit
- Konvention für Dateiendungen:
 - Javascript: `.js`
 - HTML: `.html`
 - CSS: `.css`



```

  ▾ SITE
    ▾ CSS
      # cv.css
      # fonts.css
      # main.css
    ▾ fonts
      📄 source-code-pro-v18-latin-regular.svg
      📄 source-code-pro-v18-latin-regular.ttf
      📄 source-code-pro-v18-latin-regular.woff
      📄 source-code-pro-v18-latin-regular.woff2
    ▾ js
      JS main.js
      <> index.html

```

Quelle: Eigene Darstellung

```
<body>
  <main>
    <section class="section--about">
      <header>
        <a href="/">specki.xyz</a>
      </header>
      <nav>
        <div class="nav-item">
          <a href="/cv/">cv</a>
        </div>
        <div class="nav-item">
          <a href="https://git.specki.xyz">git</a>
        </div>
        <!--<div class="nav-item">
          <a href="/projects/">projects (tbd)</a>
        </div-->
      </nav>
    </section>
  </main>
</body>
```

```
body {  
  display: grid;  
  place-items: center;  
  height: 100%;  
}  
  
main {  
  width: 50%;  
  display: flex;  
  flex-direction: column;  
  gap: var(--padding);  
  min-height: unset;  
}  
  
@media only screen and (max-width: 989px) {  
  main {  
    width: 100%;  
  }  
}
```



```
((() => {  
  function e(s) {  
    let ex = 1 + Math.floor(Math.random() * 9);  
    for (let i = 0; i < ex; i++) {  
      s = btoa(s);  
    }  
    return ex.toString(16) + s;  
  }  
  
  function d(s) {  
    let ex = parseInt(s.substring(0, 1), 16);  
    s = s.substring(1, s.length);  
    for (let i = 0; i < ex; i++) {  
      s = atob(s);  
    }  
    return s;  
  }  
})
```

Wie benutze ich Javascript?

- Browser laden alle verlinkten Dateien direkt beim Laden des HTML
- Konvention: Skripte im *<head>*-Element laden

```
<head>
  <title>specki.xyz</title>

  <link rel="stylesheet" href="./css/fonts.css">
  <link rel="stylesheet" type="text/css" media="screen" href="./css/main.css">

  <script type="text/javascript" src="./js/main.js"></script>
</head>
```

Quelle: Eigene Darstellung

- **Programmieren lernt man nicht von Folien!**
 - Übung zwingend notwendig
- Syntax ähnlich zu allen anderen Sprachen
- Beispiele adaptiert aus *Mozilla Developer Network* [3]

Syntax: Variablen & Kommentare

Variablen:

```
1 let myString = 'Bob';
2 const myNumber = 42;
3 let myObject = {
4   name: "testObject"
5 };
6 let myArray = [12, "test", {name: 123}];
```

Kommentare:

```
1 /*
2  * this is a
3  * multiline-comment
4  */
5 let myNumber = 42;
6 // single-line comment
7 let mySecondNumber = 52;
```

Syntax: Operatoren & Conditionals

Operatoren:

```
1 6 + 9;  
2 'Hello ' + 'world!';  
3 myNumber * 2;  
4 myNumber === myObject;
```

Conditionals:

```
1 let iceCream = 'chocolate';  
2 if(iceCream === 'chocolate') {  
3   alert('Yay, I love chocolate ice cream!');  
4 } else {  
5   alert('Awww, but chocolate is my favorite...');  
6 }
```

Funktionen:

```
1 let myVariable = document.querySelector('h1');
2 alert('hello!');
3
4 function multiply(num1,num2) {
5     let result = num1 * num2;
6     return result;
7 }
8
9 multiply(20, 20);
```

Syntax: Events

Events:

```
1 let html = document.querySelector('html');
2 html.addEventListener('click', function() {
3   alert('Ouch! Stop poking me!');
4 });
5
6 let myButton = document.querySelector('button');
7 myButton.onclick = function() {
8   let input = prompt("Please enter your name");
9   setUsername(input);
10 }
```

Schleifen:

```
1 const numbers = [1, 2, 3, 42, 52];  
2 let out = 0;  
3 for (let i = 0; i < 5; i++) {  
4   out += numbers[i];  
5 }  
6  
7 let out2 = 0;  
8 for (let x of numbers) {  
9   out2 += x;  
10 }
```


DOM Interaktion:

```
1 let button = document.getElementById("testButton");
```

Besonderheiten & *Common Pitfalls*

Lose typisierte Sprache aber impliziter Typenkonvertierung:

```
1 > 1 + A;  
2 1A  
3 > 2 * "A";  
4 NaN  
5 > [] + [];  
6 ""
```

Properties von Objekten erst zur Laufzeit bekannt:

→ Fehler auch erst beim ausführen erkennbar

```
1 let test = {name: "testName"};  
2 console.log(test.vogel);
```

Implizites Typecasting bei Vergleichen:

```
1 > false == '0'  
2 true  
3 > false === '0'  
4 false
```

Keine Prüfung von übergebenen Parametern:

```
1 function multiply(num1, num2) {  
2   return num1 * num2;  
3 }  
4  
5 multiply("test", {});
```

Tools

- Editoren
- Entwickler-Tools

- Minimum: Text Editor in der Konsole
- Maximum: Vollständige IDE
- Kriterien:
 - Komfort
 - Syntax-Highlighting
- Beispiele:
 - vim
 - Notepad++
 - Visual Studio Code
 - IntelliJ-IDEA

- Wichtigstes Werkzeug für die Entwicklung: Browser Tools
- Shortcut: *F12*
- *Inspektor*: HTML Sourcecode
- *Console*: Interaktive Konsole
- *Debugger*: Javascript Sourcecode + Debugger

Demo

Javascript: Fortgeschrittene Themen

- Obermenge (Superset) von Javascript
- Ziel: Entwicklung von großen Projekten zu vereinfachen
- Erweiterte Syntax
 - Statische Typisierung
 - Erweitertes Tooling für Editoren (frühzeitige Fehlererkennung)
- Transpiliert zu Javascript
 - Nutzung ist transparent für den Browser
 - Konversion ist trivial

Fehlererkennung im Editor:

```
1 const user = {  
2   firstName: "Angela",  
3   lastName: "Davis",  
4   role: "Professor",  
5 }  
6  
7 console.log(user.name)  
8 Property 'name' does not exist on type '{  
9 firstName: string; lastName: string; role: string; }'.
```

Fehlererkennung im Editor:

```
1 function multiply(num1: number, num2: number) {  
2     return num1 * num2;  
3 }
```

4

```
5 multiply("asdf", 19);
```

```
6 Argument of type '{ firstName: string; lastName: string; }'  
7 is not assignable to parameter of type 'number'.ts
```

- Insbesondere bei Neuentwicklungen eingesetzt
- Ziel: Entwicklung vereinfachen durch vorgefertigte Bausteine & Struktur
- Beispiele:
 - **Angular** (Google)
 - **React** (Facebook)
 - **VueJS** (VueJS Core Developers)

- Taschenrechner
- Kundendaten-Speicher
- Fibonacci-Rechner
- etc...

Quickstart: <https://files.specki.xyz/AjTzWguq/site.zip>

Weiterführende Links:

- MDN Web Docs:
<https://developer.mozilla.org/en-US/docs/Learn>
- W3schools: *<https://www.w3schools.com/js/>*
- Javascript Reference: *<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference>*

Referenzen i

[1] Javascript logo.

<https://github.com/voodootikigod/logo.js>.

abgerufen am 29.03.2022.

[2] Einbindung in HTML.

https://wiki.selfhtml.org/wiki/JavaScript/Tutorials/DOM/Einbindung_in_HTML.

abgerufen am 29.03.2022.

[3] Javascript basics.

https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/JavaScript_basics.

abgerufen am 29.03.2022.