

Introcution to JavaScript

Nils Twelker

March 2023

Overview

- 12 Sessions
- every Wednesday, 11:10 - 12:40
- Room: APB/E065

Questions after the lesson?

- Matrix: ...
- GitHub
- Discord?

Verbal Language?

Is there someone that only speaks English fluently?

Goals

- Learn the basics of JavaScript
- Learn to manipulate the Browser
- Complete a small project of your choice

Who am I?

- Nils Twelker
- mail@nitwel.de
- Hobbys: Cycling, Programming, Gaming
- Favorite Game: Minecraft
- I also contribute a lot to Open Source Software

Who are you?

- Name?
- Hobbys?
- Favorite Game?
- Do you know JavaScript?
- Prior experience with programming?
- Do you have a goal you want to achieve with this course?

What is JavaScript?

- Created by Brendan Eich in 1995
- Used in web browsers
- Used in Node.js

Tipp: JavaScript is not related to Java

What can you build with JavaScript?

- An interactive website
- A game
- An app of mobile devices
- A desktop app

And much more!

Getting started

You need the following

- VS Code for writing code
- Node.js for running JavaScript outside a browser
- Git for Version Control
- GitHub Account for sharing your code

Running your first JavaScript code

- Create A folder
- Open VS Code
- Click on `File > Open Folder`
- Select the folder you created
- Create a new file called `index.js`
- Write `console.log("Hello World")` and save
- Open a terminal in VS Code
- Run `node index.js`

Variables

```
let myAge = 16  
const myName = "Ben"
```

- Store Data
- `let` can be changed
- `const` can not be changed

Data Types

```
let myAge = 16  
let myName = "Ben"  
let isAdult = true
```

- `number` for numbers
- `string` for text
- `boolean` for true or false

Converting Data Types

```
const a = String(16)    // a = "16"  
const b = Number("16")  // b = 16  
const c = Boolean(16)    // c = true  
const d = Boolean(0)     // d = false  
const e = Boolean(1)     // e = true
```

Tipp:

`String()` converts to a string

`Number()` converts to a number

`Boolean()` converts to a boolean

Special Data Types

```
let yourAge = null  
let yourName
```

- `null` for empty values
- `undefined` for variables that are not yet defined

Basic Operators

```
let a = 9
let b = 4

let sum = a + b           // 13
let difference = a - b    // 5
let product = a * b       // 36
let quotient = a / b      // 2.25
let remainder = a % b     // 1
```

Modify in place

```
let a = 9
let b = 4

a += b // a = 13
a -= b // a = 5
a *= b // a = 36
a /= b // a = 2.25
a %= b // a = 1
```

Tipp: Remainder is the rest of a division

Increment and Decrement

```
let a = 9
let b = 4
let c

a++
console.log(a) // 10
c = b--
console.log(b) // 3
console.log(c) // 4
```

Tipp: `++` and `--` can be written before or after the variable.

The difference is that the variable is changed before or after the value is used.

`c = --b` would have set c to 3 aswell.

Comments


```
// This is a single line  
comment
```

```
/*  
This is a  
multi line  
comment  
*/
```

Tipp: Comments can help explain the code and ignored by the computer

Git

- Version Control
- Share your code
- Collaborate with others

 Use the Source Control section in VS Code to commit and push your changes.

Tipp: `Commit` is a bundle of changes like “added a new button to the page”

Tipp: `Push` is the process of sending your commits to a remote repository like GitHub

Platforms: `GitHub` or `GitLab` to share your code with others.

GitHub Classrooms

- Assignments for you to complete
- Submit your solutions
- Get feedback
- Track your progress

GitHub Classrooms

After you have setup your GitHub account, you can join the classroom using the following link:

<https://classroom.github.com/a/Z69I9ZBQ>

- Auth github classroom
- refresh page
- accept assignment
- click on `Open in VS Code`

Assignments

- 1 assignment each lesson
- easy, medium and hard tasks
- More points for harder tasks
- Goal are 100 Points per lesson

Assignment Structure

- `/slide` - contains the slides for the lesson
- `/tasks` - contains the tasks to complete
- `/tests` - contains the tests for the tasks

Each task will have a test associated with it. Try to pass the tests in order to complete the task.

Run `npm i` to install the dependencies.


Run `npm run test-[name]` where `[name]` is the name of the task. The test will tell you if you passed or failed.

Helpful Links

- [JavaScript.info](https://javascript.info)
- [MDN Web Docs](https://developer.mozilla.org/)
- [W3Schools](https://www.w3schools.com/)

Tipp: Or just google it, google is your friend!

After finishing a task

-  Open the Source Control section in VS Code
- Write a commit message
- Commit your changes
- Push your changes

Tipp: The name of the commit could be the name of the task