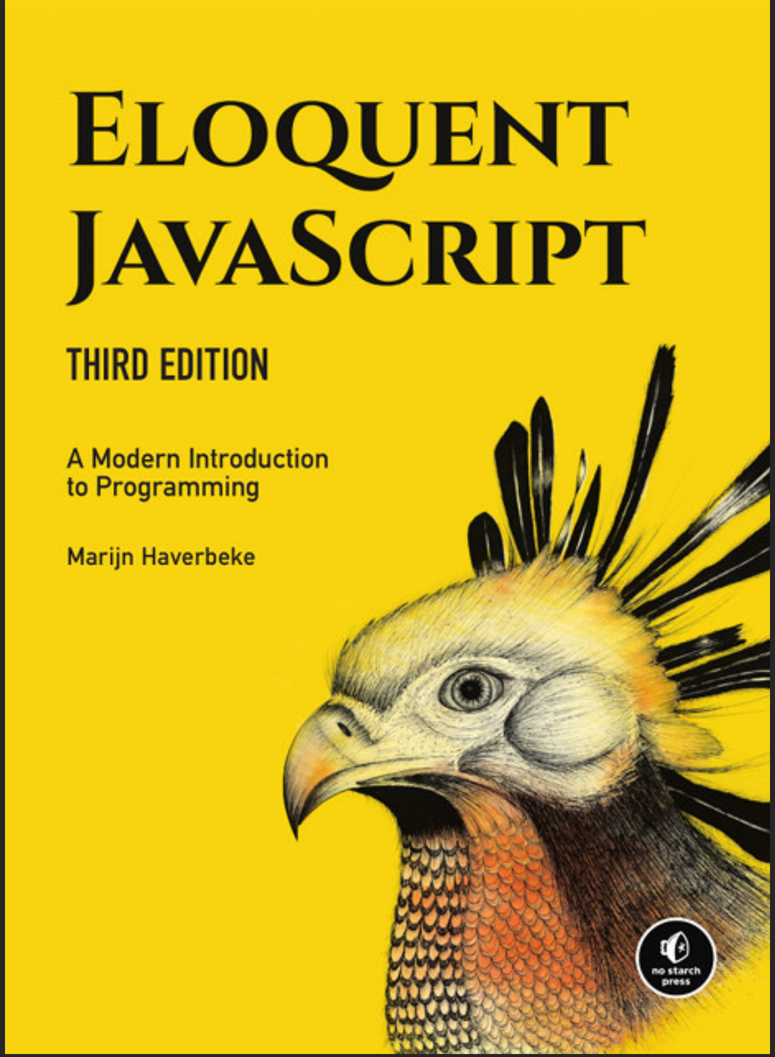
Programming JS

Margit Tennosaar

Course cookbook



https://eloquentjavascript.net/

Code editors

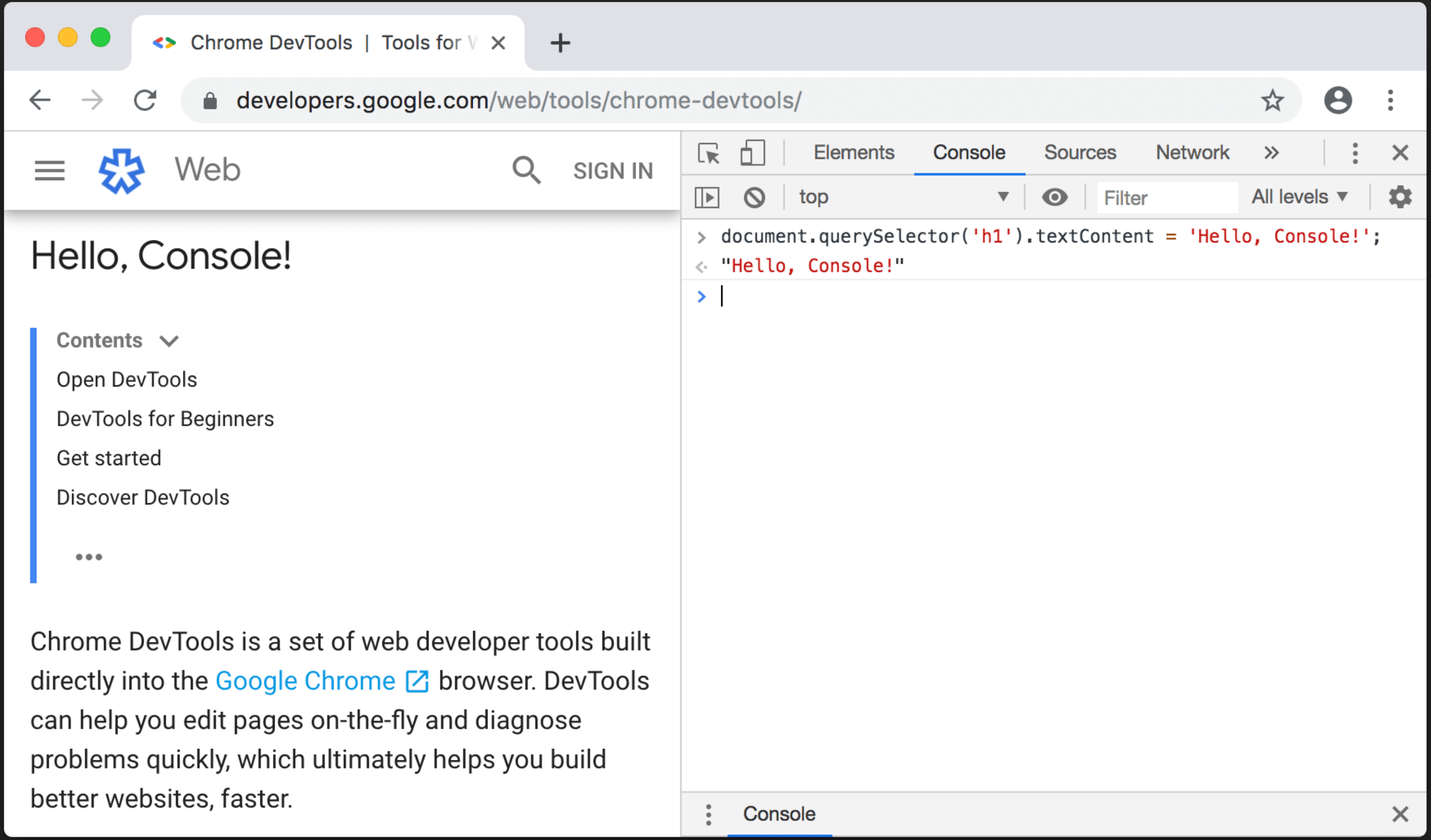
IDEs

- Visual Studio Code (cross-platform, free)
- WebStorm (cross-platform, paid)

Lightweight editors

- Atom (cross-platform, free)
- Sublime Text (cross-platform, shareware)
- Notepad++ (Windows, free)

Developer console



Chrome DevTools | Tools for W
developers.google.com/web/tools/chrome-devtools/
Web
SIGN IN
Elements
Console
Sources
Network
top
Filter
All levels
Console

Hello, Console!

Contents

Open DevTools

DevTools for Beginners

Get started

Discover DevTools

Chrome DevTools is a set of web developer tools built directly into the Google Chrome browser. DevTools can help you edit pages on-the-fly and diagnose problems quickly, which ultimately helps you build better websites, faster.

```
> document.querySelector('h1').textContent = 'Hello, Console!';
< "Hello, Console!"
>
```

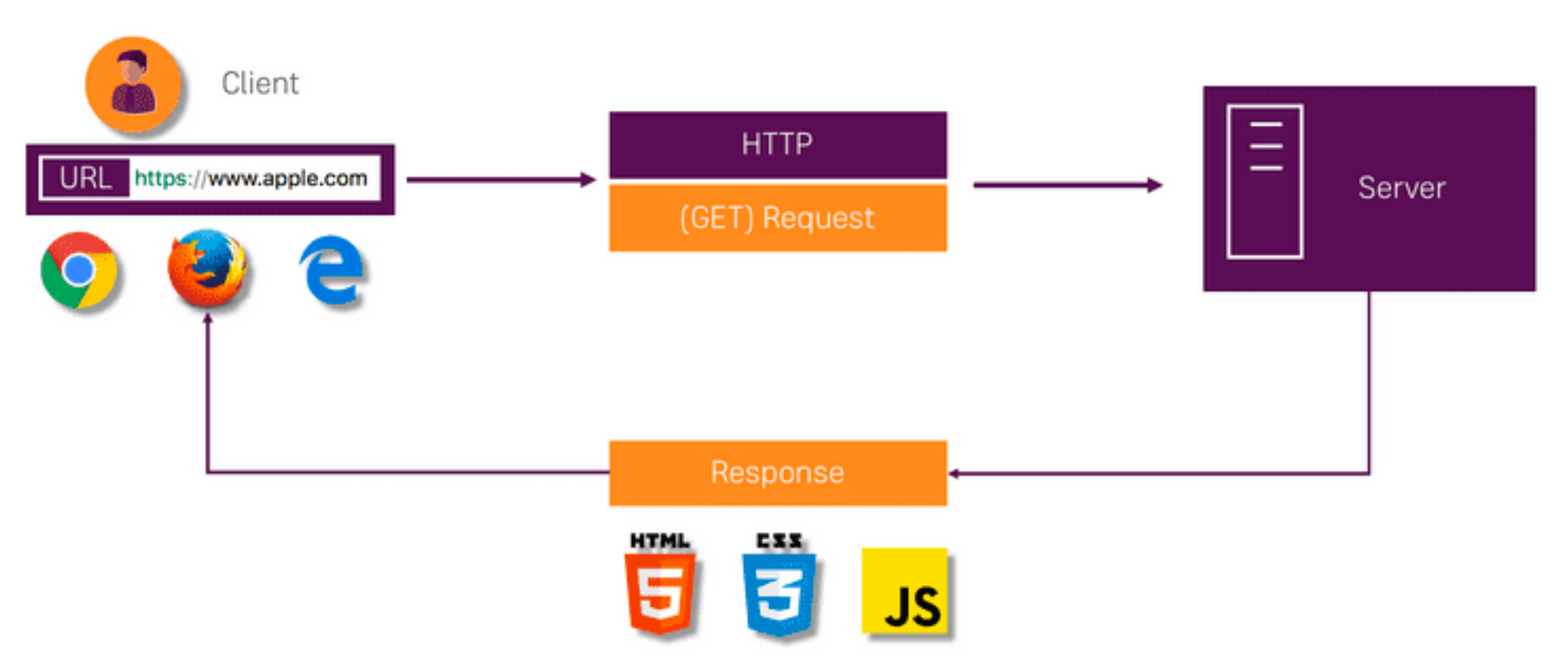
What is JavaScript?

JavaScript is a scripting language, one of the three core languages used to develop websites. Whereas HTML and CSS give a website structure and style, JavaScript lets you add functionality and behaviours to your website, allowing visitors to interact with content in imaginative ways.

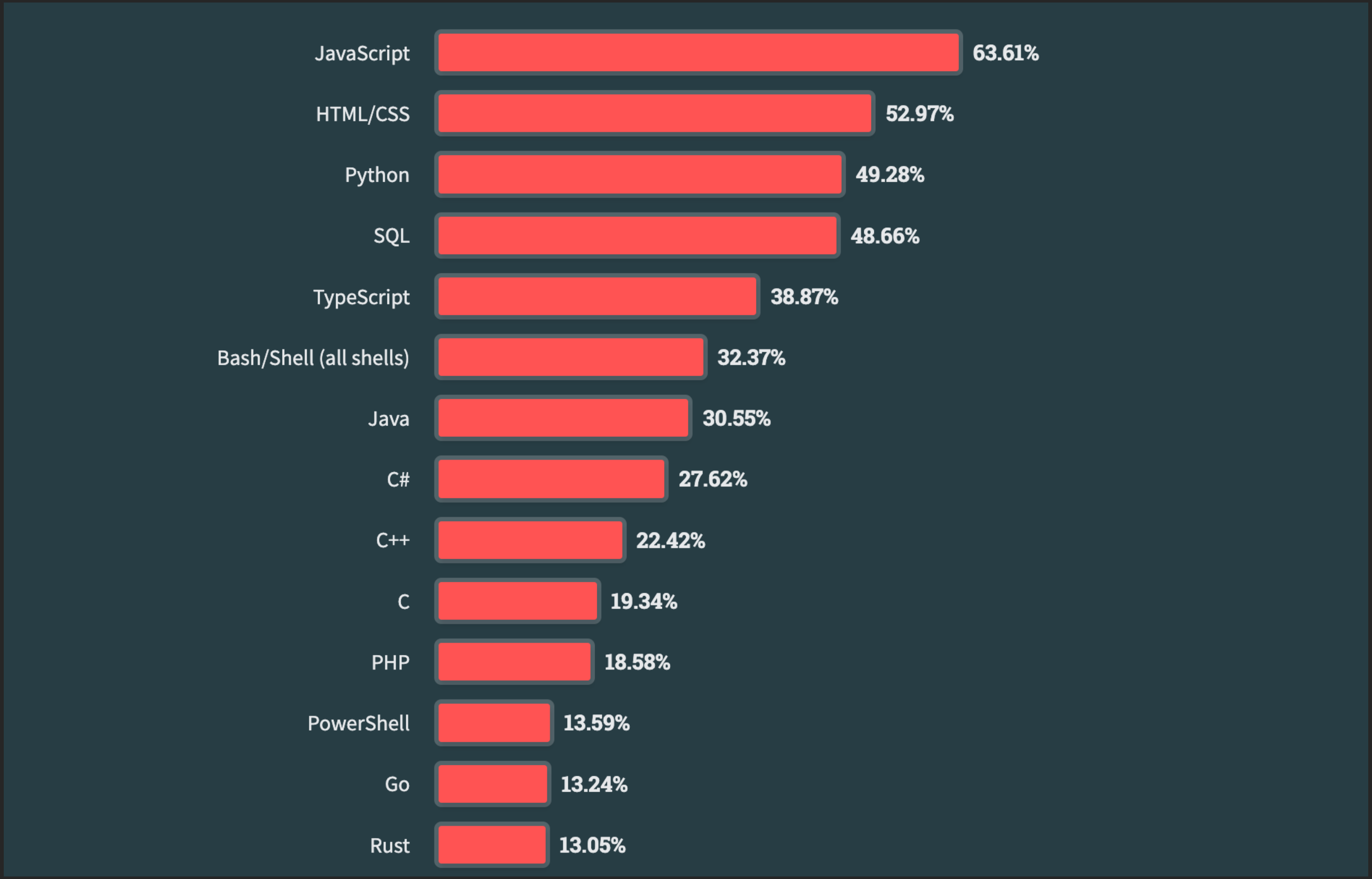
What is the difference between Java and JavaScript?

Same as between a car and a carpet...

Feature	JavaScript	Java	C/C++	Python	PHP
Paradigm	Multi-paradigm: Event-driven, functional	Object-oriented, class-based	Procedural, object-oriented (C++)	Multi-paradigm: Object-oriented, procedural	Server-side scripting
Typing	Dynamic	Static	Static	Dynamic	Dynamic
Execution Environment	Browser, server (Node.js)	Java Virtual Machine (JVM)	System-level, standalone applications	Server-side, standalone applications	Server-side (Web development)
Syntax	Prototype-based, less formal	Strict, verbose	Low-level, verbose	Easy to read, less formal	Embedded in HTML
Primary Use	Web development, real-time applications	Enterprise applications, Android apps	System/Software development, applications	Data science, web development, automation	Web development


Client
URL
https://www.apple.com
HTTP
(GET) Request
Server
Response
HTML
5
CSS
3
JS

Most popular programming, scripting, and markup languages



https://survey.stackoverflow.co/2023/#technology

ECMAscript

JavaScript was invented by Brendan Eich in 1995. It was developed for Netscape 2. After Netscape handed JavaScript over to ECMA, the Mozilla foundation continued to develop JavaScript for the Firefox browser.

JavaScript standard is commonly used for client-side scripting.

Release year	Version
1997	ES1
1998	ES2
1999	ES3
2003 /never released/	ES4
2009	ES5
2015	ES2015 (ES6)
2016	ES2016 (ES7)
2017	ES2017 (ES8)
2018	ES2018 (ES9)
2019	ES2019 (ES10)
2020	ES2020 (ES11)
2021	ES2021 (ES21)
2022	ES2022 (ES22)
2023	ES2023 (ES23)

1995: Created by Brendan Eich at Netscape; initially called LiveScript, renamed to JavaScript.

1997: JavaScript is standardized as ECMAScript to ensure consistent behavior across browsers.

2000s: Adoption of AJAX enables dynamic web applications; jQuery simplifies cross-browser scripting.

2009: Introduction of Node.js allows JavaScript to run server-side, expanding its capabilities beyond the browser.

2010s: The rise of front-end frameworks (Angular, React, Vue.js) revolutionizes web application development.

2015-onwards: ECMAScript updates (ES6 and beyond) introduce new features like arrow functions, promises, classes, and modules, enhancing the language's capabilities.

Today: JavaScript is a versatile, widely-used language for both client-side and server-side applications, with a rich ecosystem of libraries and tools.

https://developer.mozilla.org/en-US/docs/Web/JavaScript

How to connect .js files

JavaScript can be inserted almost anywhere into an HTML document using the <script> tag.

```
<script>
   alert( 'Hello, world!' );
 </script>
```

```
<script src="/path/to/script.js"></script>
```

It is recommended to add external js file in end of <body>

Task 1

Create a webpage which shows load an alert message:

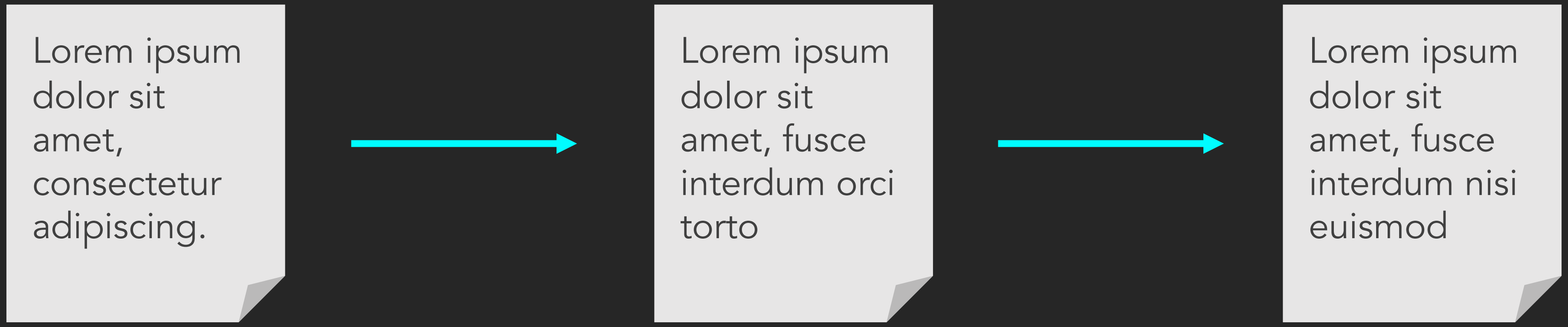
“Welcome to learning JavaScript!” and in the console “, Learning is fun!”

Use external JS file.

Version control

Version control

Version control refers to the methods by which files (or other repositories) are kept in memory, what they have been like at any stage, who has modified them, when and how (and often free-form descriptions of why).



Lorem ipsum dolor sit amet, consectetur adipiscing.

Lorem ipsum dolor sit amet, fusce interdum orci torto

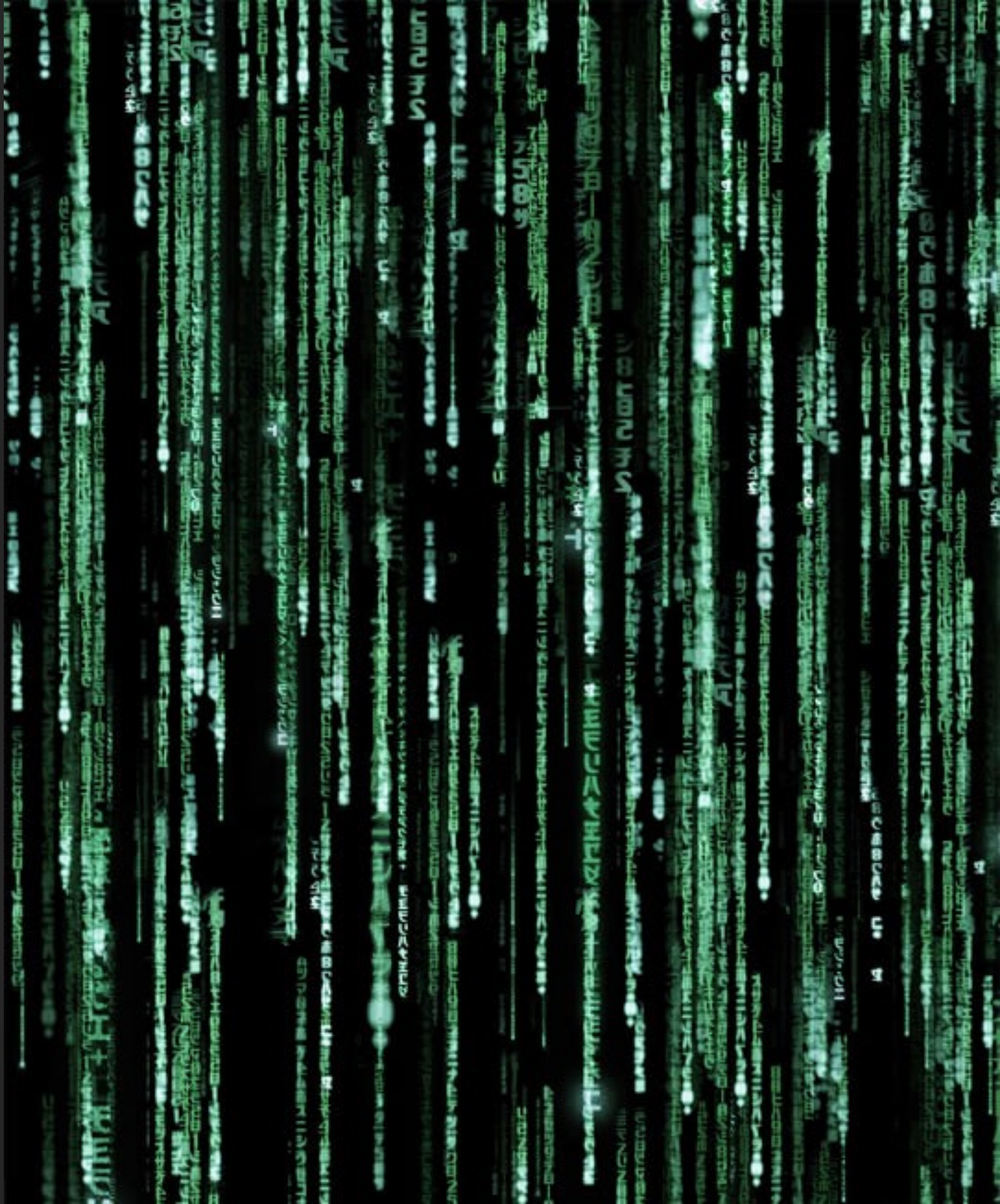
Lorem ipsum dolor sit amet, fusce interdum nisi euismod

First things first - tools

Git

CLI

GitHub



Navigating directories

```
pwd
Print Working Directory - shows the current directory path.

ls
Lists files and directories in the current directory.

cd [directory]
Change Directory - moves to the specified directory.

cd ..
Moves up one directory level.
```

Managin files and directories

```
mkdir [directory_name]
Make Directory - creates a new directory.

touch [file_name]
Creates a new file if it doesn't exist.

rm [file_name]
Removes a file.

rm -r [directory_name]
Recursively removes a directory and its contents.

cp [source] [destination]
Copies files or directories.

mv [source] [destination]
Moves files or directories, or renames them.
```

Working with Git

```
git init
Initializes a new Git repository.

git clone [url]
Clones a repository into a new directory.

git status
Shows the status of changes as untracked, modified, or staged.

git add [file/directory]
Adds a file or directory to the staging area.

git commit -m "[commit message]”
Records file snapshots in the version history.

git pull
Fetches changes from the remote server and merges them into the current
branch.

git push
Uploads all local branch commits to the remote repository.
```

Reading about GitHub

https://guides.github.com/

https://github.com/git-school/visualizing-git

https://learngitbranching.js.org/

https://github.com/margittennosaar/github_study_materials

Pre-task

1. Create an account in GitHub

2. Install Git

3. Configure Git

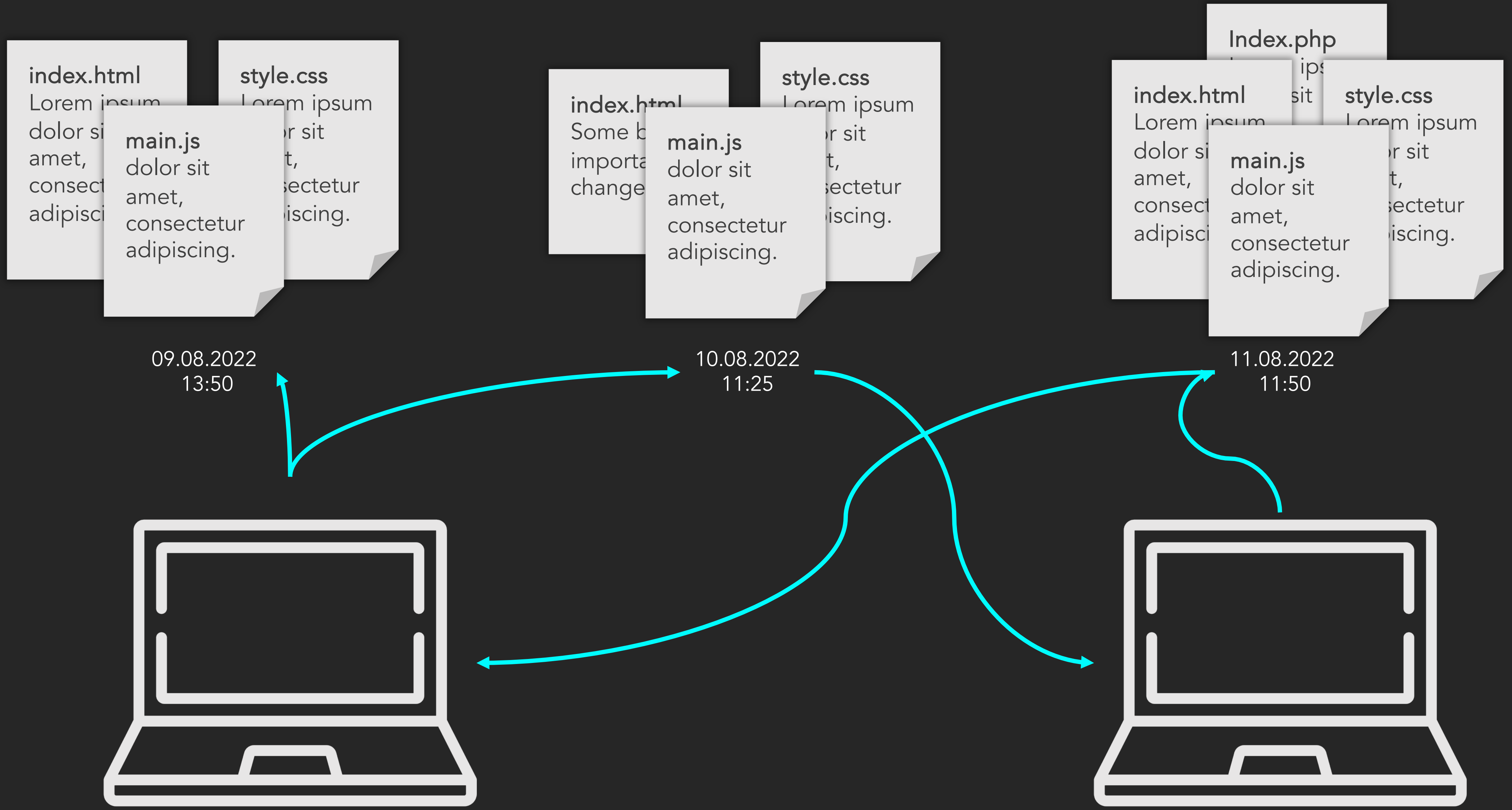
```
git config --global user.name "Your Name"

git config --global user.email "your.email@example.com"
```

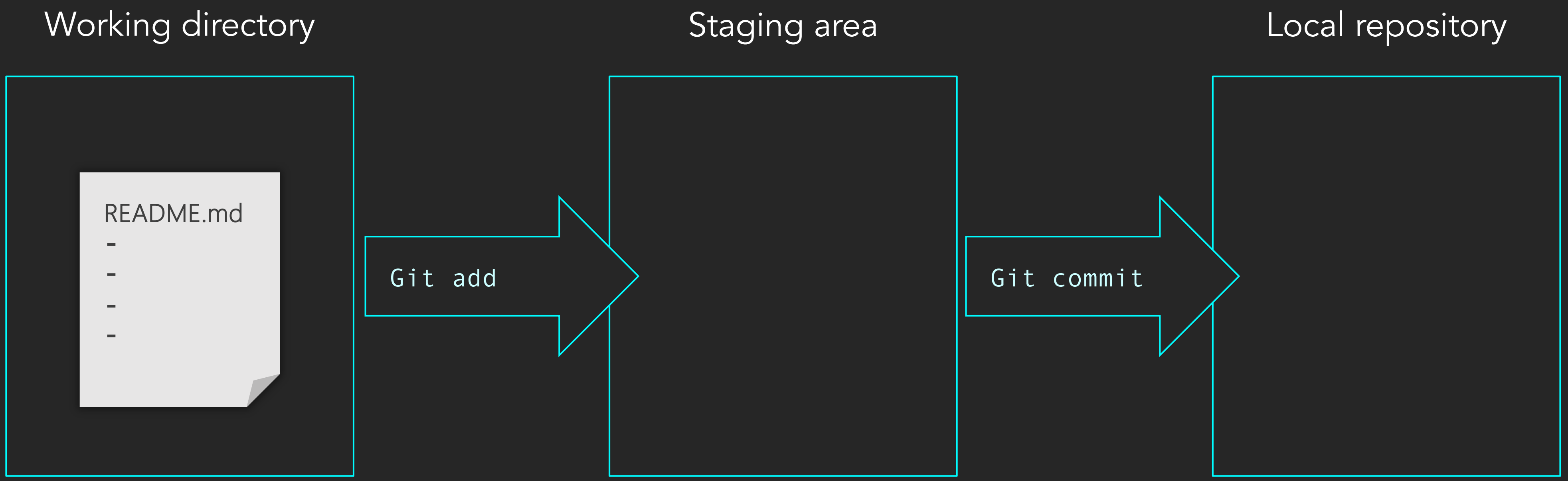
3. Install and set up VSc

4. Create a logical folder system where you start collect your school projects

Git - repository



Three stages



Tools you might need install

Node - https://nodejs.org/en/download/

Xcode (command line tools)

Brew

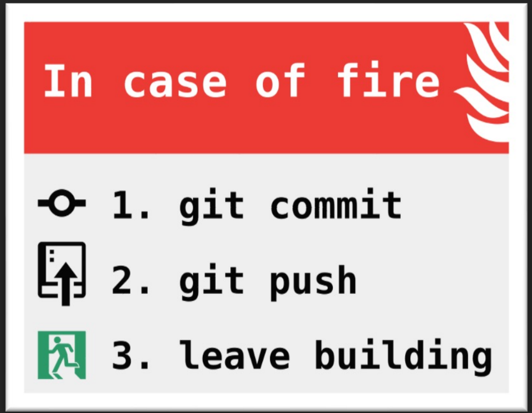
GIT

Github connection

Create a new repository in GitHub for JS projects.

Connect your JS lessons folder and GitHub repository

- Open terminal and locate JS folder
- git init
- git add .
- git commit -m „first commit“
- git remote add origin URL
- git push origin master



In case of fire

1. git commit
2. git push
3. leave building

https://kbroman.org/github_tutorial/pages/init.html