HOCHSCHULE FULDA
UNIVERSITY OF APPLIED SCIENCES

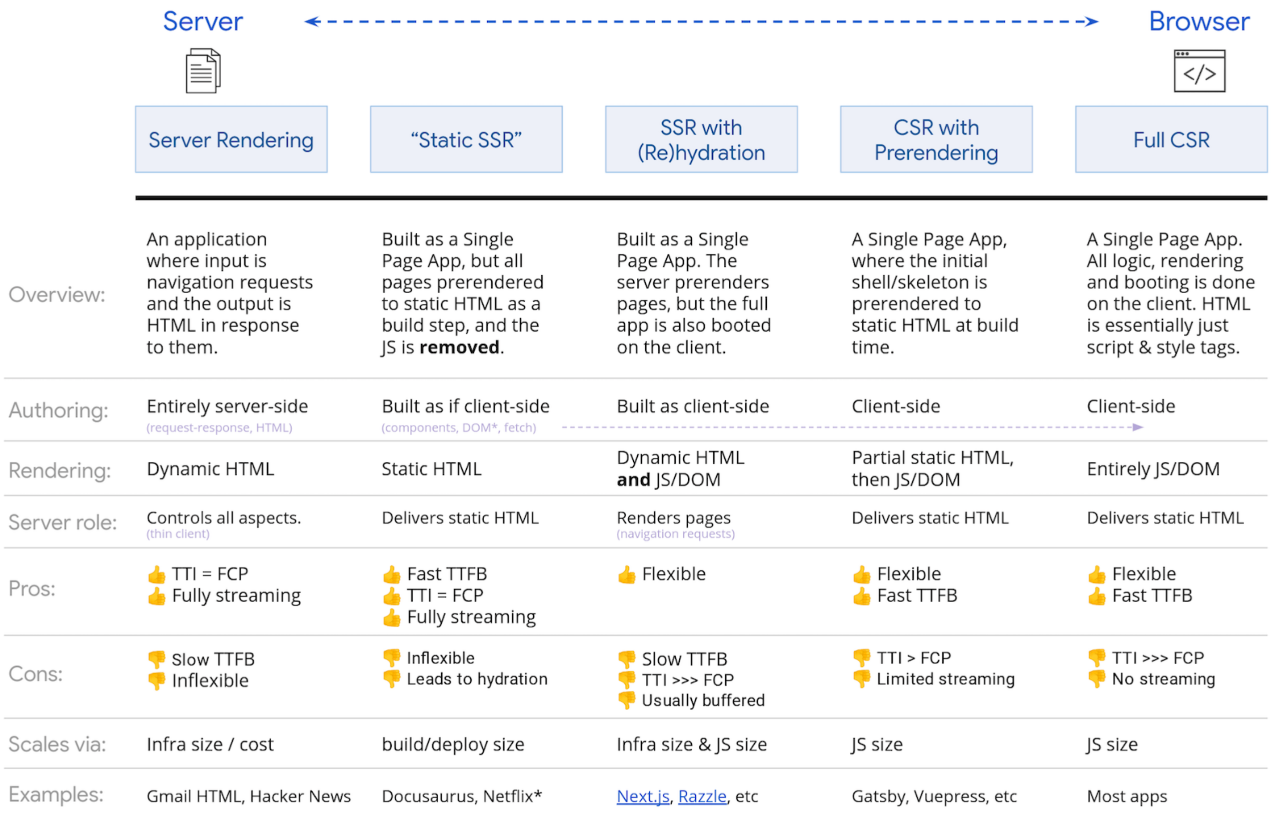
WEB-APPLIKATIONEN

JAVASCRIPT IM BROWSER

AI
Manuel Richardt, Dominik Rupprecht
FB **ANGEWANDTE INFORMATIK**

SERVER VS. CLIENT RENDERING

WER MACHT WAS



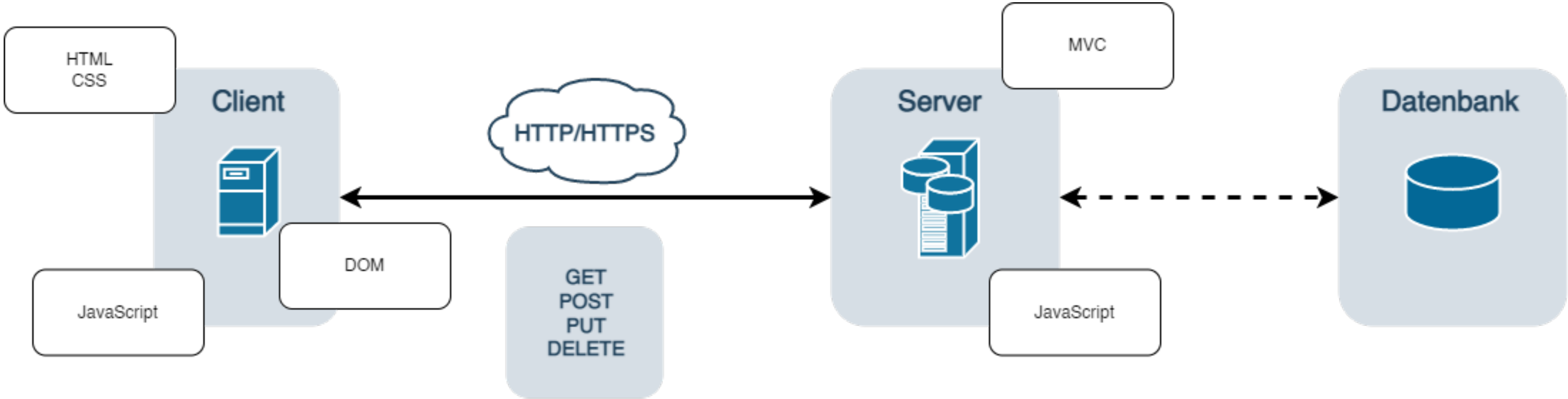
Server
Browser

	Server Rendering	"Static SSR"	SSR with (Re)hydration	CSR with Prerendering	Full CSR
Overview:	An application where input is navigation requests and the output is HTML in response to them.	Built as a Single Page App, but all pages prerendered to static HTML as a build step, and the JS is **removed**.	Built as a Single Page App. The server prerenders pages, but the full app is also booted on the client.	A Single Page App, where the initial shell/skeleton is prerendered to static HTML at build time.	A Single Page App. All logic, rendering and booting is done on the client. HTML is essentially just script & style tags.
Authoring:	Entirely server-side (request-response, HTML)	Built as if client-side (components, DOM*, fetch)	Built as client-side	Client-side	Client-side
Rendering:	Dynamic HTML	Static HTML	Dynamic HTML **and** JS/DOM	Partial static HTML, then JS/DOM	Entirely JS/DOM
Server role:	Controls all aspects. (thin client)	Delivers static HTML	Renders pages (navigation requests)	Delivers static HTML	Delivers static HTML
Pros:	👍 TTI = FCP 👍 Fully streaming	👍 Fast TTFB 👍 TTI = FCP 👍 Fully streaming	👍 Flexible	👍 Flexible 👍 Fast TTFB	👍 Flexible 👍 Fast TTFB
Cons:	👎 Slow TTFB 👎 Inflexible	👎 Inflexible 👎 Leads to hydration	👎 Slow TTFB 👎 TTI >>> FCP 👎 Usually buffered	👎 TTI > FCP 👎 Limited streaming	👎 TTI >>> FCP 👎 No streaming
Scales via:	Infra size / cost	build/deploy size	Infra size & JS size	JS size	JS size
Examples:	Gmail HTML, Hacker News	Docusaurus, Netflix*	Next.js, Razzle, etc	Gatsby, Vuepress, etc	Most apps

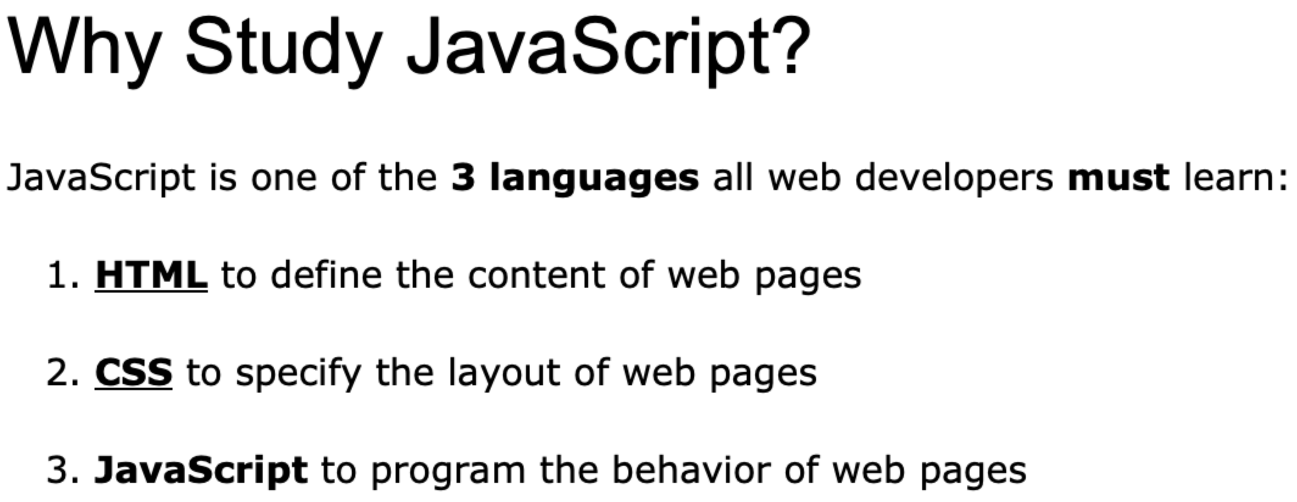
AI

Manuel Richardt, Dominik Rupprecht
FB **ANGEWANDTE INFORMATIK**

JAVASCRIPT IM SERVER UND IM CLIENT



WARUM JAVASCRIPT?



Why Study JavaScript?

JavaScript is one of the **3 languages** all web developers **must** learn:

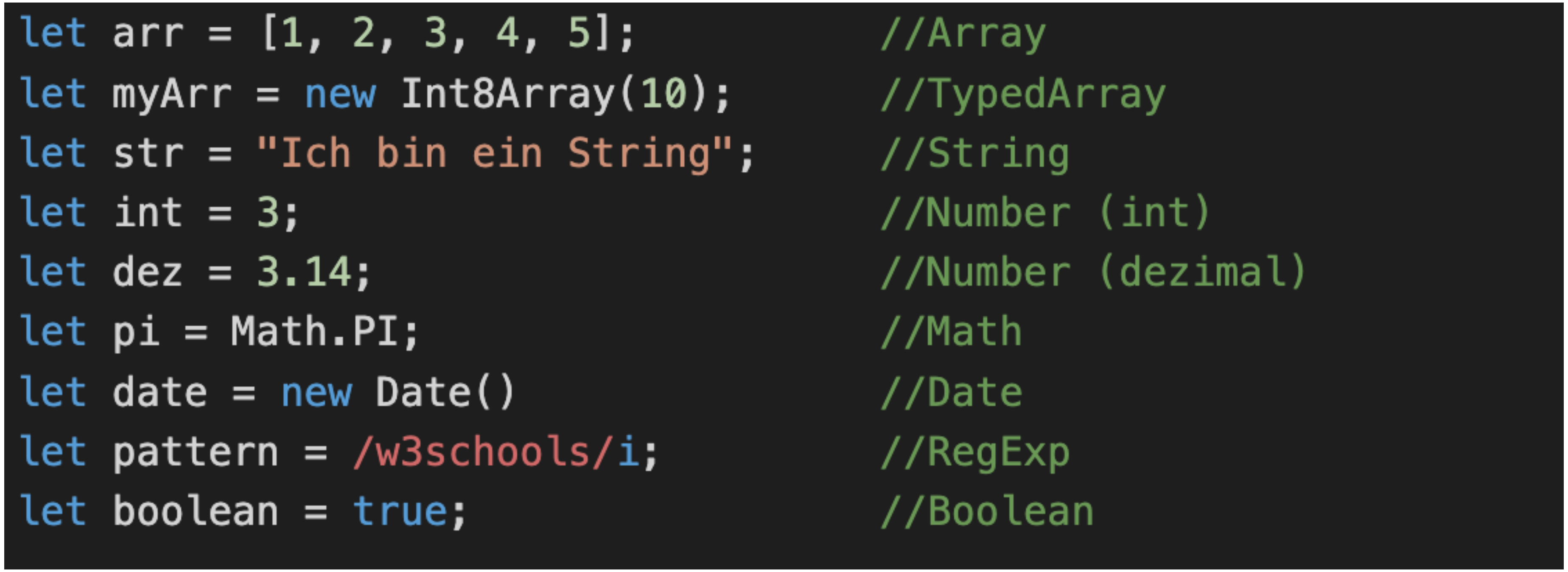
1. **HTML** to define the content of web pages
2. **CSS** to specify the layout of web pages
3. **JavaScript** to program the behavior of web pages

Quelle: https://www.w3schools.com/js/default.asp, Abgerufen am 15.06.2023

JAVASCRIPT IM BROWSER

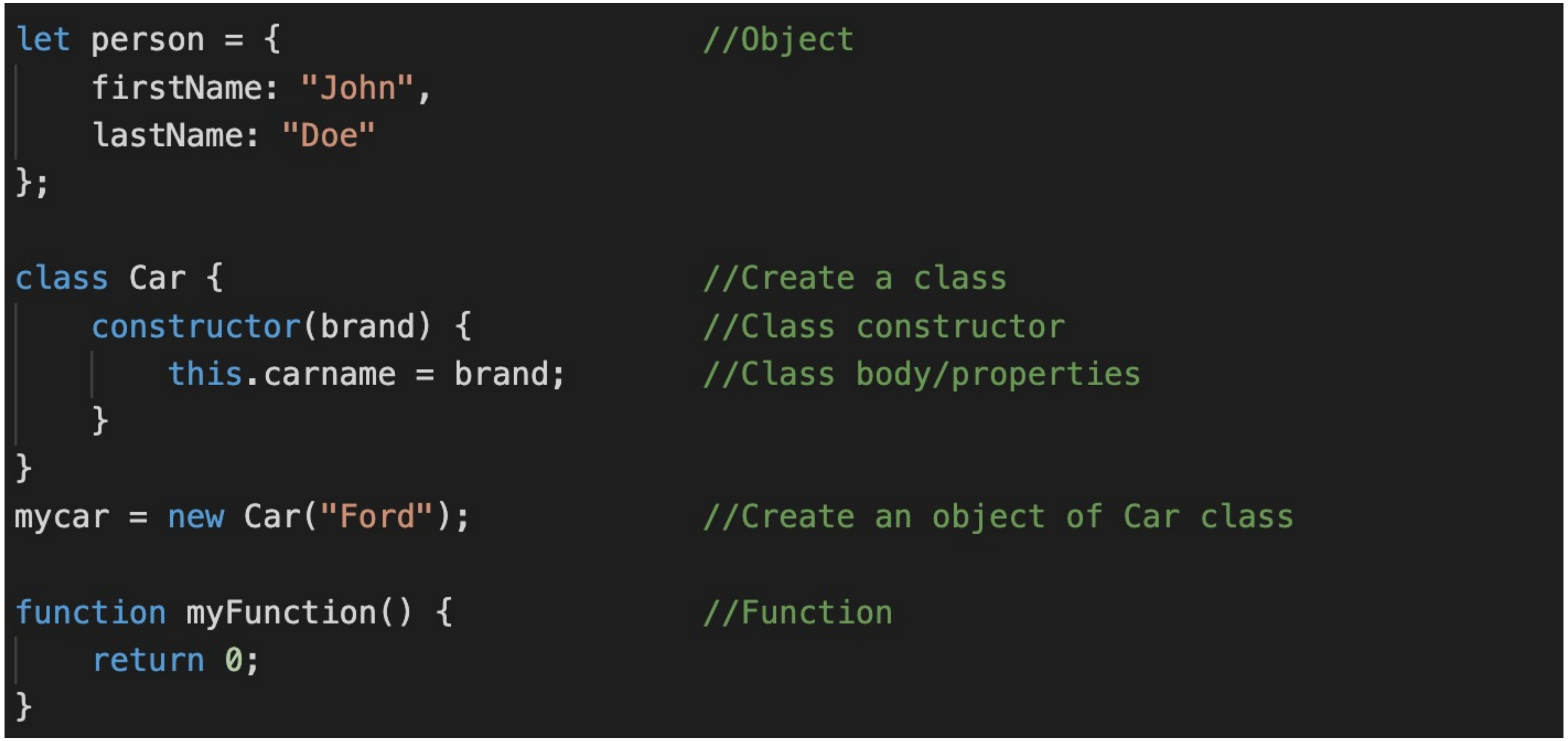
1. JAVASCRIPT OBJECTS
2. WINDOW OBJECTS
3. HTML DOM OBJECTS
4. WEBAPIS
5. HTML ELEMENTS

DATENTYPEN IN JAVASCRIPT



```
let arr = [1, 2, 3, 4, 5];          //Array
let myArr = new Int8Array(10);      //TypedArray
let str = "Ich bin ein String";     //String
let int = 3;                        //Number (int)
let dez = 3.14;                     //Number (dezimal)
let pi = Math.PI;                   //Math
let date = new Date()               //Date
let pattern = /w3schools/i;         //RegExp
let boolean = true;                 //Boolean
```

OBJEKT UND KLASSEN IN JAVASCRIPT

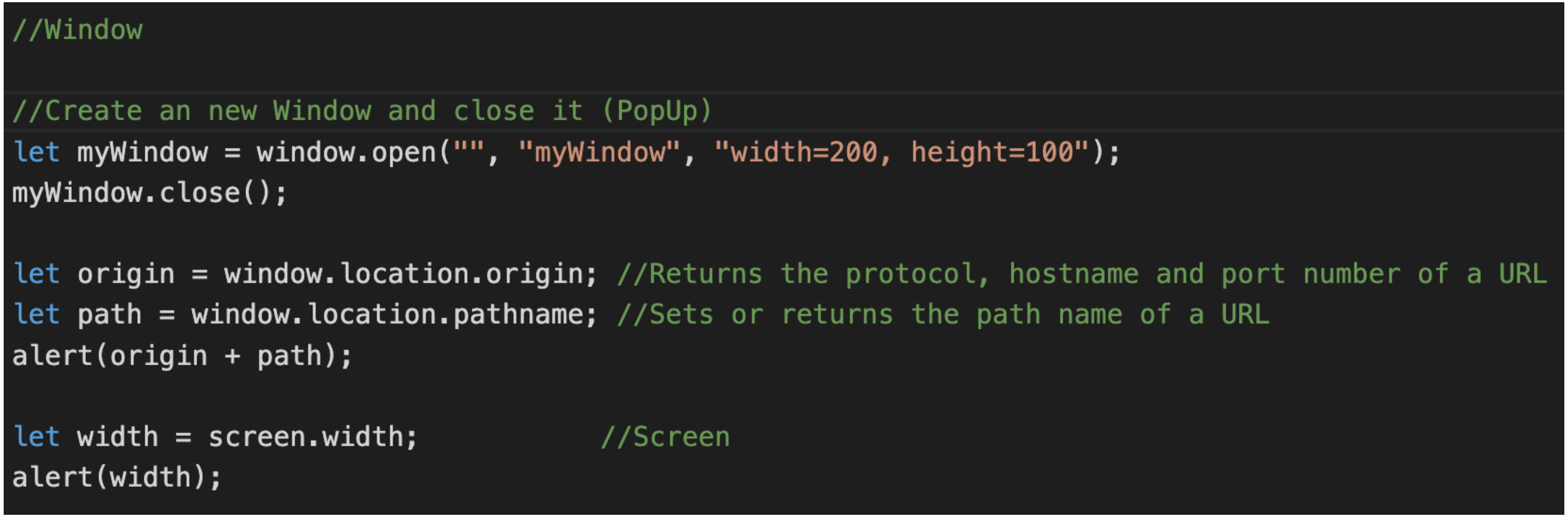


```
let person = {                      //Object
    firstName: "John",
    lastName: "Doe"
};

class Car {                         //Create a class
    constructor(brand) {            //Class constructor
        this.carname = brand;       //Class body/properties
    }
}
mycar = new Car("Ford");            //Create an object of Car class

function myFunction() {             //Function
    return 0;
}
```

JS: WINDOW UND SCREEN



```
//Window

//Create an new Window and close it (PopUp)
let myWindow = window.open("", "myWindow", "width=200, height=100");
myWindow.close();

let origin = window.location.origin; //Returns the protocol, hostname and port number of a URL
let path = window.location.pathname; //Sets or returns the path name of a URL
alert(origin + path);

let width = screen.width;           //Screen
alert(width);
```

DOCUMENT OBJECT

1. WENN EIN HTML-DOKUMENT IN EINEN WEBBROWSER GELADEN WIRD, WIRD ES ZU EINEM DOKUMENTOBJEKT (DOCUMENT).
2. DAS DOKUMENTOBJEKT IST DER WURZELKNOTEN DES HTML-DOKUMENTS.
3. DAS DOKUMENTOBJEKT IST EINE EIGENSCHAFT DES FENSTEROBJEKTS (WINDOW).

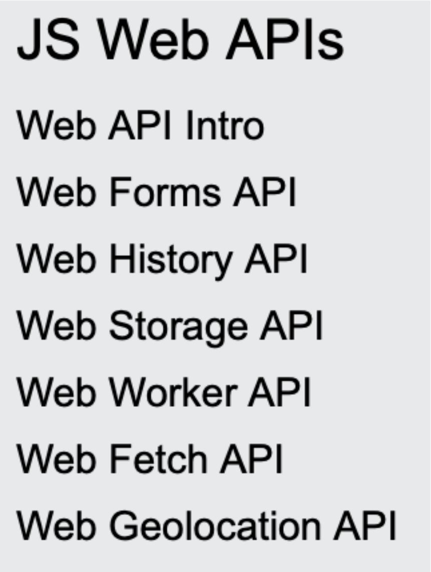
DOCUMENT METHODS - AUSWAHL

Method	Description
addEventListener()	Attaches an event handler to the document
cookie	Returns all name/value pairs of cookies in the document
createAttribute()	Creates an attribute node
createElement()	Creates an Element node
getElementByID()	Returns the element that has the ID attribute with the specified value
getElementByClassName()	Returns an HTMLCollection containing all elements with the specified class name
getElementByName()	Returns an live NodeList containing all elements with the specified name
getElementByTagName()	Returns an HTMLCollection containing all elements with the specified tag name
hasFocus()	Returns a Boolean value indicating whether the document has focus
URL	Returns the full URL of the HTML document

ELEMENT METHODS - AUSWAHL

Method	Description
addEventListener()	Attaches an event handler to an element
appendChild()	Adds (appends) a new child node to an element
attributes	Returns a NamedNodeMap of an element's attributes
childNodes	Returns a NodeList of an element's child nodes
children	Returns an HTMLCollection of an element's child elements
click()	Simulates a mouse-click on an element
getAttribute()	Returns the value of an element's attribute
innerHTML	Sets or returns the content of an element
innerText	Sets or returns the text content of a node and its descendants
style	Sets or returns the value of the style attribute of an element

WEB APIS



JS Web APIs

Web API Intro

Web Forms API

Web History API

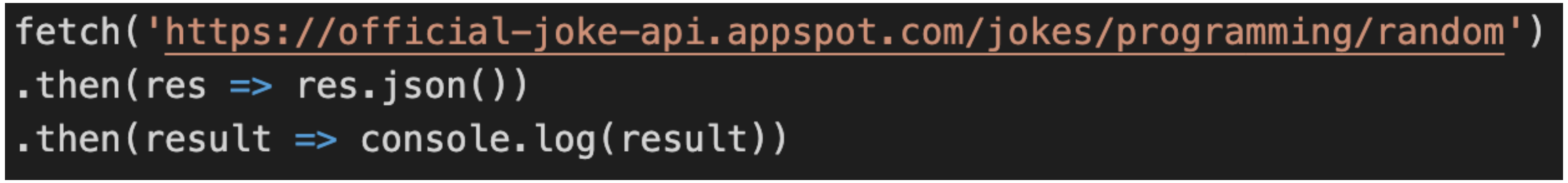
Web Storage API

Web Worker API

Web Fetch API

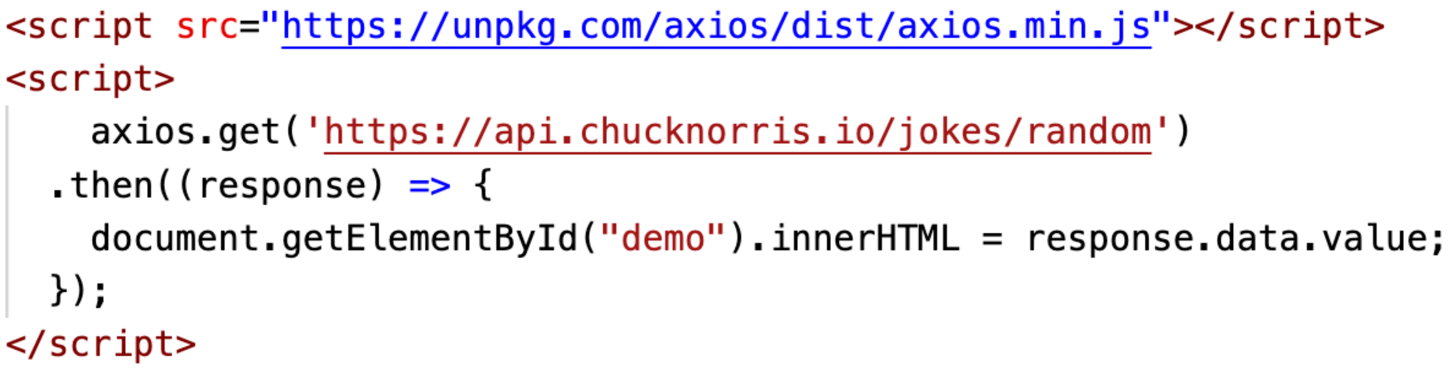
Web Geolocation API

- JavaScript FETCH



```
fetch('https://official-joke-api.appspot.com/jokes/programming/random')
.then(res => res.json())
.then(result => console.log(result))
```

- Axios



```
<script src="https://unpkg.com/axios/dist/axios.min.js"></script>
<script>
    axios.get('https://api.chucknorris.io/jokes/random')
  .then((response) => {
    document.getElementById("demo").innerHTML = response.data.value;
  });
</script>
```

ZUSAMMENFASSUNG

1. JAVASCRIPT KANN NICHT NUR IM SERVER, SONDERN AUCH IM CLIENT (BROWSER) EINGESETZT WERDEN
2. ERMÖGLICHT UMFANGREICHE MANIPULATION VON HTML-SEITEN
3. ZUGRIFF AUF HTML-ELEMENTE ÜBER `document.getElement…`