JavaScript

Luca Berres

Allgemines zu JavaScript

Syntax

Einbindung JavaScript

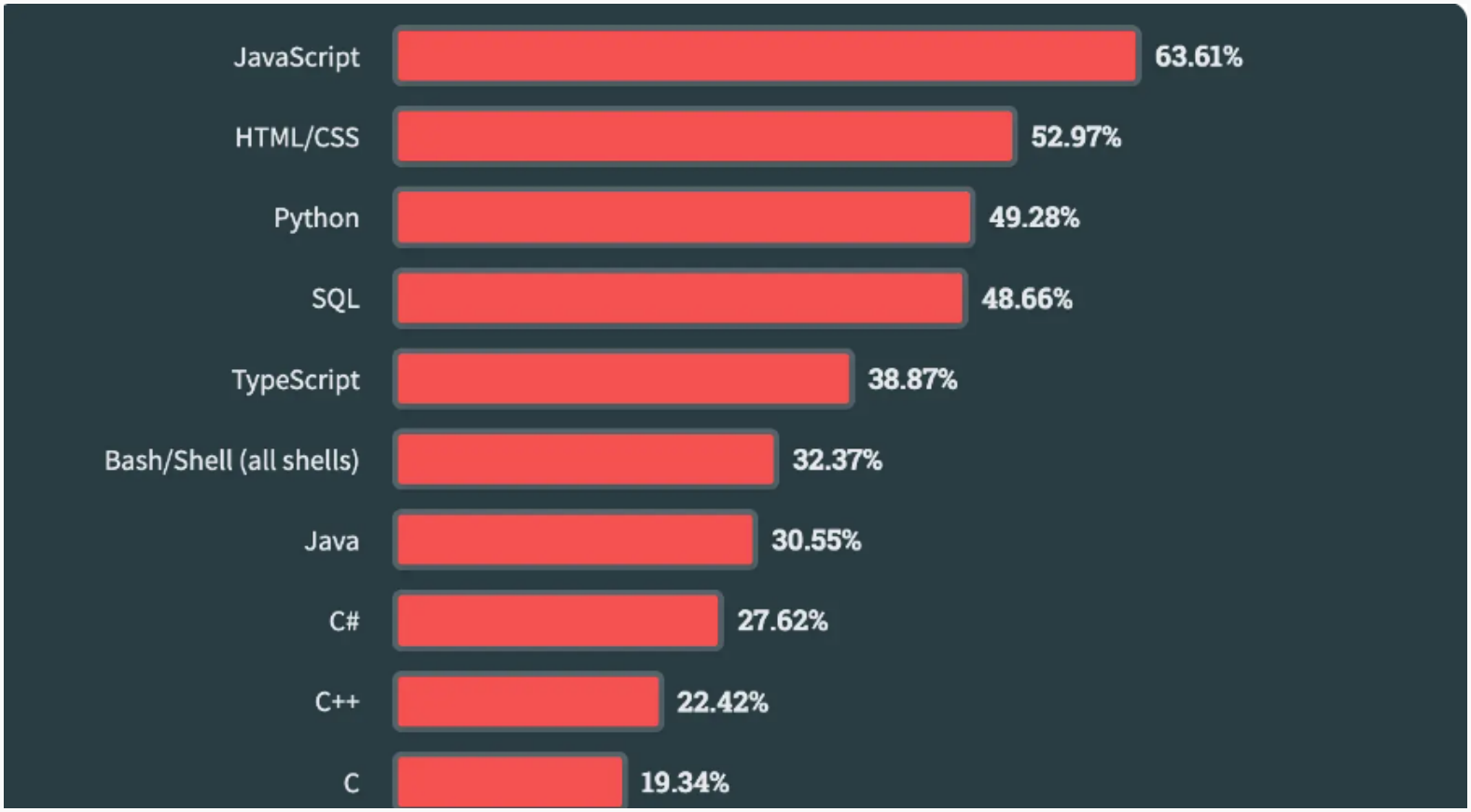
Allgemine Programmierkonzepte

JavaScript im Browser

Allgemines zu JavaScript

Allgemeines

- entwickelt 1995 von Brendan Eich um Webseiten mit Interaktion auszustatten
- Eine der beliebtesteten Programmiersprachen
- Trotz Namensähnlichkeit nicht mit JAVA verwandt, aber beide orientieren sich von der Syntax an C



Wer steht hinter JavaScript

ECMA International (früher: European Computer Manufacturers Association)

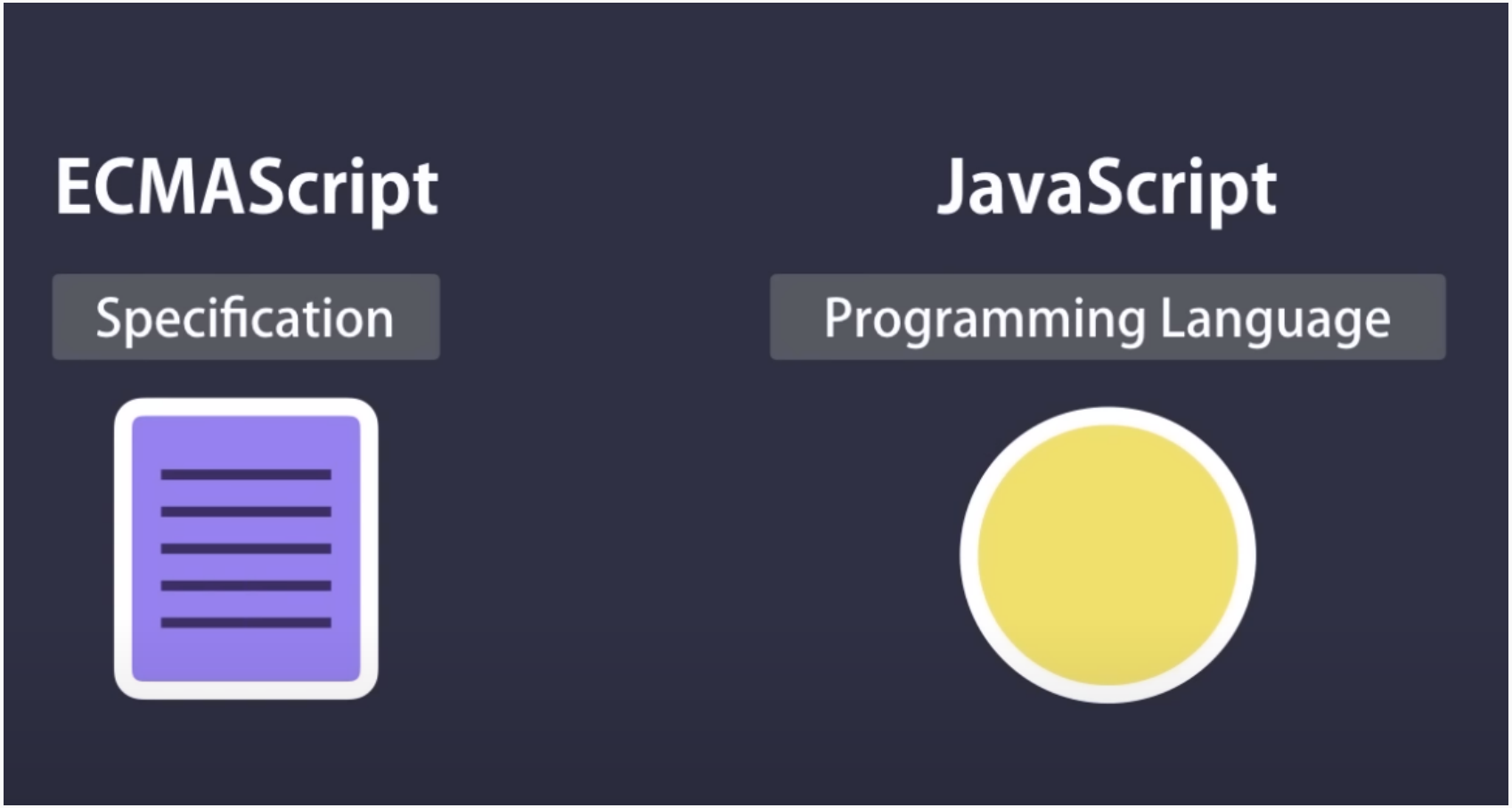


Figure 1: ECMA

Wo läuft JavaScript?

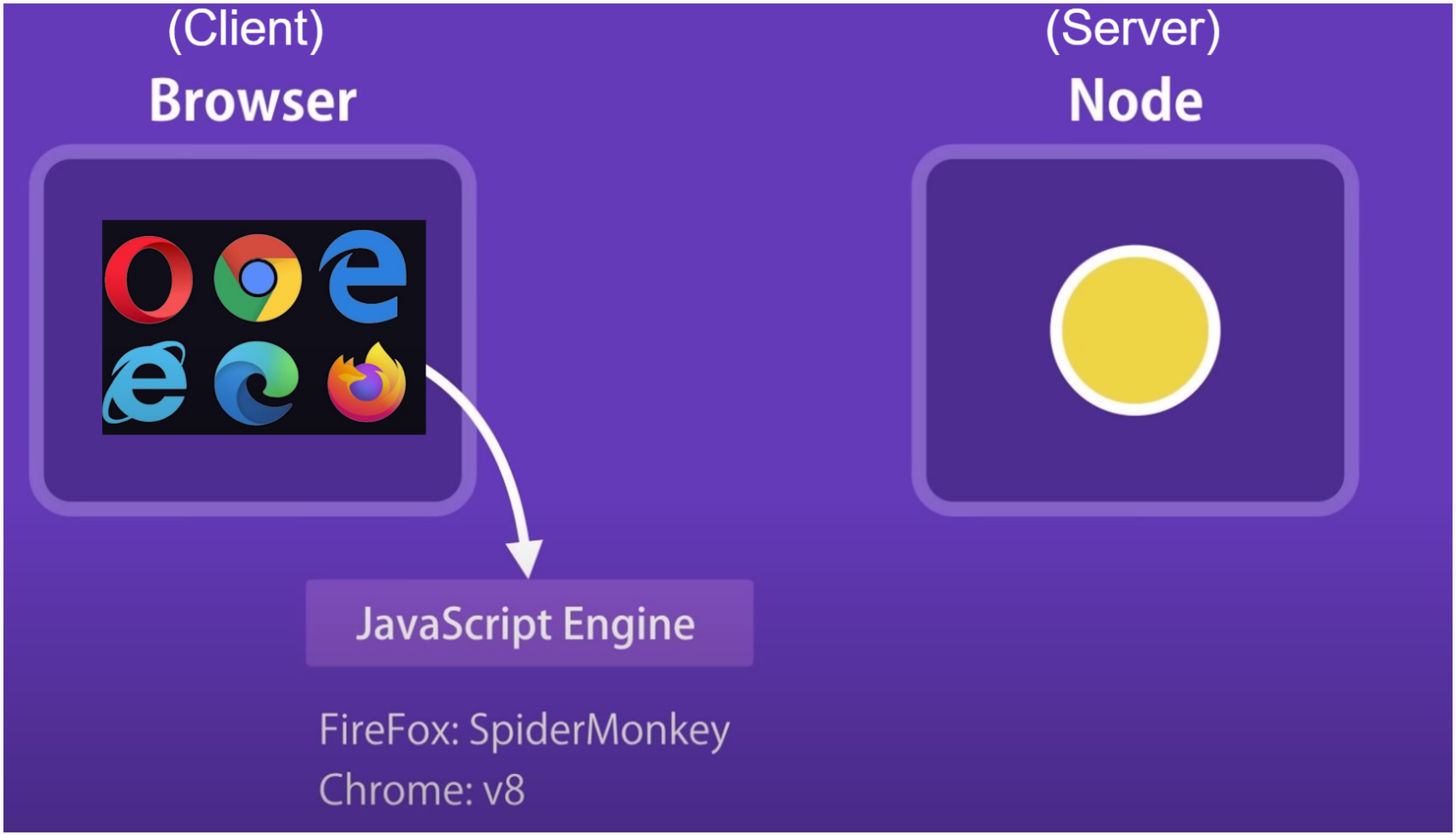


Figure 2: JavaScript Runtime

Auf welchen Plattformen läuft JavaScript?

- Server Applikationen-> Node.js
- Desktop Applikationen -> Electron
- Mobile Applikationen -> React Native oder Ionic

Syntax

Einbindung JavaScript

Eingebettetes im HTML

```html
<!DOCTYPE html>
<html lang="de">
  <head>
    <meta charset="UTF-8" />
    <title>Meine Webseite</title>
  </head>
  <body>
    <h1>Willkommen auf meiner Webseite</h1>
    <script>
      console.log("Hallo Welt");
    </script>
  </body>
</html>
```

Extern referenziert im HTML

1. Erstelle eine Datei namens script.js mit folgendem Inhalt:

```
alert("Hallo, Welt!");
```

2. Binde die externe Datei in dein HTML-Dokument ein

```
<!DOCTYPE html>
<html lang="de">
  <head>
    <meta charset="UTF-8" />
    <title>Meine Webseite</title>
  </head>
  <body>
    <h1>Willkommen auf meiner Webseite</h1>
    <script src="script.js"></script>
  </body>
</html>
```

Konsole

1. Öffne die Entwicklertools in deinem Browser: In Chrome: Rechtsklick -> “Untersuchen” -> Tab “Konsole” oder F12
2. Führe JavaScript-Code direkt in der Konsole aus:

```
console.log("Hallo, Welt!");
```

Standalone

1. Erstelle eine Datei mit folgendem Inhalt und speicher sie als Test.js ab

```
console.log("Hallo, Welt!");
```

2. Öffne sie mit einem Browser
3. Öffne die Konsole wie zuvor gezeigt

Allgemine Programmierkonzepte

Kommenatere

Variables

Übersicht

- speichern Daten temporär
- Analogie: beschrifteter Karton mit Inhalt

Komponente (DE)	**Komponente (EN)**	**Beschreibung**	**Beispiel**
Bezeichner	Identifier	Name der Variable, die ihren Wert bezeichnet.	test
Literal	Literal	Wert, der der Variable zugewiesen wird.	42, “Text”
Schlüsselwort	Keyword	Reserviertes Wort in der Programmiersprache für Deklaration oder Steuerung.	let, const, var

Keyword in JavaScript

- var (veraltet, weil globaler scope(Geltungsbereich))
- let (block-scoped -> Geltungsbereich ist eine näheste von geschweiften Klammern umschlossenen Syntaxen, z.B. if statement)
- const (block-scoped, kann nicht nochmals zugewiesen werden)

Numerische und Boolsche Literals

Typ	Beispiel
Hexadezimale Konstanten	`var test = 0x12f`
Binäre Konstanten	`var test = 0b011101`
Oktale Konstanten	`var test = 0o767`
Ganzzahlenkonstanten	`var test = 123456`
Gleitkommazahlen	`var test = 12.34`
	`var test = 12.34e2`
Boolesche Konstanten	`var test = true`
	`var test = false`

Zeichenketten/Strings Literals

```
var jsString = `Das ist ein String`; // Backticks
var jsString = "Das ist ein String"; // einfache Anführungs
var jsString = "Das ist ein String"; // doppelte Anführungs
```

Vorteil von Backticks:

```
var jsString = `half of 100 is ${100 / 2}`
console.log(jsString)
// -> half of 100 is 50
```

Operatoren

Operator	Bedeutung	Beispiel
+, +=	Addition	x+=3
-, -=	Subtraktion	x=x-5
, =	Multiplikation	a=b*c
/, /=	Division	z=e/5
%	Modulus	m=5 % 3
++, –	Inkrement, Dekrement	x++ oder y–
«, «=	Bitweise Linksschieben	x « 4
», »=	Bitweise Rechtsschieben	y » 5
»>	Bitweise Linksschieben mit Nullfüllung	a »> b
&	Bitweise UND	a & b
|	Bitweise ODER	a | b
^	Bitweise Negieren	a ^ b

Short Circuit Evaluation

Erklärung

Short Circuit Evaluation ist eine Programmiertechnik, bei der der Auswertungsprozess eines logischen Ausdrucks frühzeitig beendet wird, sobald das Ergebnis feststeht.

Beispiele in JavaScript

Logisches UND (&&)

```
const a = false;
const b = true;
const result = a && b; // result ist false,
```

JavaScript im Browser