

# JavaScript

Das umfassende Handbuch

# DAS INHALTS- VERZEICHNIS

» Hier geht's  
direkt  
zum Buch

---

# Auf einen Blick

1	Grundlagen und Einführung .....	29
2	Erste Schritte .....	55
3	Sprachkern .....	85
4	Mit Referenztypen arbeiten .....	243
5	Webseiten dynamisch verändern .....	381
6	Ereignisse verarbeiten und auslösen .....	439
7	Mit Formularen arbeiten .....	491
8	Browser steuern und Browserinformationen auslesen .....	521
9	Inhalte einer Webseite dynamisch nachladen .....	549
10	Aufgaben vereinfachen mit jQuery .....	595
11	Bilder und Grafiken dynamisch erstellen .....	641
12	Moderne Web-APIs verwenden .....	673
13	Objektorientierte Programmierung .....	773
14	Funktionale Programmierung .....	817
15	Den Quelltext richtig strukturieren .....	831
16	Asynchrone Programmierung und weitere fortgeschrittene Features verwenden .....	853
17	Serverseitige Anwendungen mit Node.js erstellen .....	887
18	Mobile Anwendungen mit JavaScript erstellen .....	935
19	KI-Anwendungen mit JavaScript .....	959
20	Einen professionellen Entwicklungsprozess aufsetzen .....	1001

---

# Inhalt

Materialien zum Buch .....	23
Vorwort .....	25

## **1 Grundlagen und Einführung** 29

---

<b>1.1 Grundlagen der Programmierung</b> .....	29
1.1.1 Mit dem Computer kommunizieren .....	30
1.1.2 Programmiersprachen .....	31
1.1.3 Hilfsmittel für den Programmentwurf .....	39
<b>1.2 Einführung in JavaScript</b> .....	45
1.2.1 Historie .....	45
1.2.2 Anwendungsgebiete .....	47
<b>1.3 Zusammenfassung</b> .....	53

## **2 Erste Schritte** 55

---

<b>2.1 Einführung in JavaScript und die Webentwicklung</b> .....	55
2.1.1 Der Zusammenhang zwischen HTML, CSS und JavaScript .....	55
2.1.2 Das richtige Werkzeug für die Entwicklung .....	59
<b>2.2 JavaScript in eine Webseite einbinden</b> .....	63
2.2.1 Eine geeignete Ordnerstruktur vorbereiten .....	64
2.2.2 Eine JavaScript-Datei erstellen .....	64
2.2.3 Eine JavaScript-Datei in eine HTML-Datei einbinden .....	65
2.2.4 JavaScript direkt innerhalb des HTML definieren .....	68
2.2.5 Platzierung und Ausführung der <script>-Elemente .....	70
2.2.6 Den Quelltext anzeigen .....	74
<b>2.3 Eine Ausgabe erzeugen</b> .....	77
2.3.1 Standarddialogfenster anzeigen .....	77
2.3.2 Auf die Konsole schreiben .....	78
2.3.3 Bestehende UI-Komponenten verwenden .....	82
<b>2.4 Zusammenfassung</b> .....	84

---

<b>3</b>	<b>Sprachkern</b>	85
<b>3.1</b>	<b>Werte in Variablen speichern</b>	85
3.1.1	Variablen definieren	85
3.1.2	Gültige Variablenamen verwenden	88
3.1.3	Konstanten definieren	96
<b>3.2</b>	<b>Die verschiedenen Datentypen verwenden</b>	96
3.2.1	Zahlen	97
3.2.2	Zeichenketten	100
3.2.3	Boolesche Werte	106
3.2.4	Arrays	106
3.2.5	Objekte	112
3.2.6	Besondere Datentypen	114
3.2.7	Symbole	115
<b>3.3</b>	<b>Die verschiedenen Operatoren einsetzen</b>	116
3.3.1	Operatoren für das Arbeiten mit Zahlen	117
3.3.2	Operatoren für das einfachere Zuweisen	118
3.3.3	Operatoren für das Arbeiten mit Zeichenketten	120
3.3.4	Operatoren für das Arbeiten mit booleschen Werten	121
3.3.5	Operatoren für das Arbeiten mit Bits	128
3.3.6	Operatoren für das Vergleichen von Werten	129
3.3.7	Der Optional Chaining Operator	132
3.3.8	Die Logical Assignment Operatoren	134
3.3.9	Operatoren für spezielle Operationen	136
<b>3.4</b>	<b>Den Ablauf eines Programms steuern</b>	137
3.4.1	Bedingte Anweisungen definieren	137
3.4.2	Verzweigungen definieren	139
3.4.3	Den Auswahloperator verwenden	146
3.4.4	Mehrfachverzweigungen definieren	147
3.4.5	Zählschleifen definieren	154
3.4.6	Kopfgesteuerte Schleifen definieren	162
3.4.7	Fußgesteuerte Schleifen definieren	165
3.4.8	Schleifen und Schleifeniterationen vorzeitig abbrechen	167
<b>3.5</b>	<b>Wiederverwendbare Codebausteine erstellen</b>	176
3.5.1	Funktionen definieren	176
3.5.2	Funktionen aufrufen	180
3.5.3	Funktionsparameter übergeben und auswerten	180
3.5.4	Rückgabewerte definieren	189
3.5.5	Standardwerte für Parameter definieren	191

---

3.5.6	Elemente aus einem Array als Parameter verwenden .....	193
3.5.7	Funktionen über Kurzschreibweise definieren .....	195
3.5.8	Zeichenketten über Funktionen verändern .....	197
3.5.9	Funktionen im Detail .....	199
3.5.10	Funktionen aufrufen durch Nutzerinteraktion .....	207
<b>3.6</b>	<b>Auf Fehler reagieren und sie richtig behandeln .....</b>	<b>208</b>
3.6.1	Syntaxfehler .....	209
3.6.2	Laufzeitfehler .....	209
3.6.3	Logische Fehler .....	210
3.6.4	Das Prinzip der Fehlerbehandlung .....	211
3.6.5	Fehler fangen und behandeln .....	213
3.6.6	Fehler auslösen .....	215
3.6.7	Fehler und der Funktionsaufruf-Stack .....	220
3.6.8	Bestimmte Anweisungen unabhängig von aufgetretenen Fehlern aufrufen .....	222
<b>3.7</b>	<b>Den Quelltext kommentieren .....</b>	<b>228</b>
<b>3.8</b>	<b>Den Code debuggen .....</b>	<b>229</b>
3.8.1	Einführung .....	229
3.8.2	Ein einfaches Codebeispiel .....	229
3.8.3	Haltepunkte definieren .....	230
3.8.4	Variablenbelegungen einsehen .....	233
3.8.5	Ein Programm schrittweise ausführen .....	233
3.8.6	Mehrere Haltepunkte definieren .....	236
3.8.7	Die verschiedenen Typen von Haltepunkten verwenden .....	237
3.8.8	Den Funktionsaufruf-Stack einsehen .....	237
<b>3.9</b>	<b>Zusammenfassung .....</b>	<b>239</b>
<b>4</b>	<b>Mit Referenztypen arbeiten .....</b>	<b>243</b>
<b>4.1</b>	<b>Unterschied zwischen primitiven Datentypen und Referenztypen .....</b>	<b>243</b>
4.1.1	Das Prinzip von primitiven Datentypen .....	243
4.1.2	Das Prinzip von Referenztypen .....	244
4.1.3	Primitive Datentypen und Referenztypen als Funktionsargumente .....	246
4.1.4	Den Typ einer Variablen ermitteln .....	247
4.1.5	Ausblick .....	250
<b>4.2</b>	<b>Zustand und Verhalten in Objekten kapseln .....</b>	<b>251</b>
4.2.1	Einführung objektorientierte Programmierung .....	251
4.2.2	Objekte erstellen über die Literal-Schreibweise .....	252

4.2.3	Objekte erstellen über Konstruktorfunktionen .....	254
4.2.4	Objekte erstellen unter Verwendung von Klassen .....	257
4.2.5	Objekte erstellen über die Funktion »Object.create()« .....	261
4.2.6	Auf Eigenschaften zugreifen und Methoden aufrufen .....	265
4.2.7	Objekteigenschaften und Objektmethoden hinzufügen oder überschreiben .....	271
4.2.8	Objekteigenschaften und Objektmethoden löschen .....	276
4.2.9	Objekteigenschaften und Objektmethoden ausgeben .....	278
4.2.10	Symbole zur Definition eindeutiger Objekteigenschaften verwenden .....	282
4.2.11	Änderungen an Objekten verhindern .....	283
<b>4.3</b>	<b>Mit Arrays arbeiten</b> .....	<b>286</b>
4.3.1	Arrays erzeugen und initialisieren .....	287
4.3.2	Auf Elemente eines Arrays zugreifen .....	290
4.3.3	Elemente einem Array hinzufügen .....	292
4.3.4	Elemente aus einem Array entfernen .....	296
4.3.5	Einen Teil der Elemente aus einem Array kopieren .....	299
4.3.6	Arrays sortieren .....	302
4.3.7	Arrays als Stack verwenden .....	305
4.3.8	Arrays als Queue verwenden .....	307
4.3.9	Elemente in Arrays finden .....	308
4.3.10	Elemente innerhalb eines Arrays kopieren .....	311
4.3.11	Arrays in Zeichenketten umwandeln .....	312
<b>4.4</b>	<b>Werte aus Arrays und Objekten extrahieren</b> .....	<b>313</b>
4.4.1	Werte aus Arrays extrahieren .....	313
4.4.2	Werte aus Objekten extrahieren .....	317
4.4.3	Werte innerhalb einer Schleife extrahieren .....	320
4.4.4	Argumente einer Funktion extrahieren .....	322
4.4.5	Objekteigenschaften in ein anderes Objekt kopieren .....	323
4.4.6	Objekteigenschaften aus einem anderen Objekt kopieren .....	324
<b>4.5</b>	<b>Mit Zeichenketten arbeiten</b> .....	<b>325</b>
4.5.1	Der Aufbau einer Zeichenkette .....	326
4.5.2	Die Länge einer Zeichenkette ermitteln .....	326
4.5.3	Innerhalb einer Zeichenkette suchen .....	327
4.5.4	Teile einer Zeichenkette extrahieren .....	330
<b>4.6</b>	<b>Maps verwenden</b> .....	<b>333</b>
4.6.1	Maps erstellen .....	333
4.6.2	Grundlegende Operationen .....	334
4.6.3	Über Maps iterieren .....	336
4.6.4	Weak Maps verwenden .....	338

---

<b>4.7</b>	<b>Sets verwenden</b> .....	340
4.7.1	Sets erstellen .....	340
4.7.2	Grundlegende Operationen von Sets .....	341
4.7.3	Über Sets iterieren .....	343
4.7.4	Weak Sets verwenden .....	344
<b>4.8</b>	<b>Sonstige globale Objekte</b> .....	345
4.8.1	Mit Datum und Zeit arbeiten .....	345
4.8.2	Komplexe Berechnungen durchführen .....	348
4.8.3	Wrapperobjekte für primitive Datentypen .....	348
<b>4.9</b>	<b>Mit regulären Ausdrücken arbeiten</b> .....	349
4.9.1	Reguläre Ausdrücke definieren .....	349
4.9.2	Zeichenketten gegen einen regulären Ausdruck testen .....	350
4.9.3	Zeichenklassen verwenden .....	353
4.9.4	Anfang und Ende begrenzen .....	356
4.9.5	Quantifizierer verwenden .....	359
4.9.6	Nach Vorkommen suchen .....	364
4.9.7	Alle Vorkommen innerhalb einer Zeichenkette suchen .....	365
4.9.8	Auf einzelne Teile eines Vorkommens zugreifen .....	366
4.9.9	Nach bestimmten Zeichenketten suchen .....	368
4.9.10	Vorkommen innerhalb einer Zeichenkette ersetzen .....	368
4.9.11	Nach Vorkommen suchen .....	369
4.9.12	Zeichenketten zerteilen .....	369
<b>4.10</b>	<b>Funktionen als Referenztypen</b> .....	370
4.10.1	Funktionen als Argumente verwenden .....	370
4.10.2	Funktionen als Rückgabewert verwenden .....	373
4.10.3	Standardmethoden jeder Funktion .....	375
<b>4.11</b>	<b>Zusammenfassung</b> .....	379
<b>5</b>	<b>Webseiten dynamisch verändern</b> .....	381
<b>5.1</b>	<b>Aufbau einer Webseite</b> .....	381
5.1.1	Document Object Model .....	381
5.1.2	Die verschiedenen Knotentypen .....	382
5.1.3	Der Dokumentknoten .....	386
<b>5.2</b>	<b>Elemente selektieren</b> .....	387
5.2.1	Elemente per ID selektieren .....	390
5.2.2	Elemente per Klasse selektieren .....	392
5.2.3	Elemente nach Elementnamen selektieren .....	396

5.2.4	Elemente nach Namen selektieren .....	397
5.2.5	Elemente per Selektor selektieren .....	399
5.2.6	Das Elternelement eines Elements selektieren .....	405
5.2.7	Die Kindelemente eines Elements selektieren .....	408
5.2.8	Die Geschwisterelemente eines Elements selektieren .....	412
5.2.9	Selektionsmethoden auf Elementen aufrufen .....	415
5.2.10	Elemente nach Typ selektieren .....	417
<b>5.3</b>	<b>Mit Textknoten arbeiten</b> .....	417
5.3.1	Auf den Textinhalt eines Elements zugreifen .....	418
5.3.2	Den Textinhalt eines Elements verändern .....	419
5.3.3	Das HTML unterhalb eines Elements verändern .....	420
5.3.4	Textknoten erstellen und hinzufügen .....	421
<b>5.4</b>	<b>Mit Elementen arbeiten</b> .....	421
5.4.1	Elemente erstellen und hinzufügen .....	422
5.4.2	Elemente und Knoten entfernen .....	425
5.4.3	Die verschiedenen Typen von HTML-Elementen .....	426
<b>5.5</b>	<b>Mit Attributen arbeiten</b> .....	431
5.5.1	Den Wert eines Attributs auslesen .....	432
5.5.2	Den Wert eines Attributs ändern oder ein neues Attribut hinzufügen .....	434
5.5.3	Attributknoten erstellen und hinzufügen .....	435
5.5.4	Attribute entfernen .....	435
5.5.5	Auf CSS-Klassen zugreifen .....	435
<b>5.6</b>	<b>Zusammenfassung</b> .....	437
<b>6</b>	<b>Ereignisse verarbeiten und auslösen</b> .....	439
<hr/>		
<b>6.1</b>	<b>Das Konzept der ereignisgesteuerten Programmierung</b> .....	439
<b>6.2</b>	<b>Auf Ereignisse reagieren</b> .....	440
6.2.1	Einen Event-Handler per HTML definieren .....	443
6.2.2	Einen Event-Handler per JavaScript definieren .....	445
6.2.3	Event-Listener definieren .....	447
6.2.4	Mehrere Event-Listener definieren .....	449
6.2.5	Argumente an Event-Listener übergeben .....	451
6.2.6	Event-Listener entfernen .....	453

---

6.2.7	Event-Handler und Event-Listener per Helferfunktion definieren .....	454
6.2.8	Auf Informationen eines Ereignisses zugreifen .....	455
<b>6.3</b>	<b>Die verschiedenen Typen von Ereignissen .....</b>	<b>457</b>
6.3.1	Ereignisse bei Interaktion mit der Maus .....	458
6.3.2	Ereignisse bei Interaktion mit der Tastatur .....	463
6.3.3	Ereignisse beim Arbeiten mit Formularen .....	465
6.3.4	Ereignisse bei Fokussieren von Elementen .....	466
6.3.5	Allgemeine Ereignisse der Nutzerschnittstelle .....	467
6.3.6	Ereignisse bei mobilen Endgeräten .....	470
<b>6.4</b>	<b>Den Ereignisfluss verstehen und beeinflussen .....</b>	<b>471</b>
6.4.1	Die Event-Phasen .....	471
6.4.2	Den Ereignisfluss unterbrechen .....	478
6.4.3	Standardaktionen von Events verhindern .....	484
<b>6.5</b>	<b>Ereignisse programmatisch auslösen .....</b>	<b>486</b>
6.5.1	Einfache Ereignisse auslösen .....	486
6.5.2	Ereignisse mit übergebenen Argumenten auslösen .....	487
6.5.3	Standardereignisse auslösen .....	488
<b>6.6</b>	<b>Zusammenfassung .....</b>	<b>489</b>
 <b>7 Mit Formularen arbeiten</b>		<b>491</b>
<hr/>		
<b>7.1</b>	<b>Auf Formulare und Formularfelder zugreifen .....</b>	<b>492</b>
7.1.1	Auf Formulare zugreifen .....	492
7.1.2	Auf Formularelemente zugreifen .....	495
7.1.3	Den Wert von Textfeldern und Passwortfeldern auslesen .....	497
7.1.4	Den Wert von Checkboxes auslesen .....	499
7.1.5	Den Wert von Radiobuttons auslesen .....	499
7.1.6	Den Wert von Auswahllisten auslesen .....	501
7.1.7	Die Werte von Mehrfachauswahllisten auslesen .....	503
7.1.8	Auswahllisten per JavaScript mit Werten befüllen .....	504
<b>7.2</b>	<b>Formulare programmatisch abschicken und zurücksetzen .....</b>	<b>505</b>
<b>7.3</b>	<b>Formulareingaben validieren .....</b>	<b>508</b>
<b>7.4</b>	<b>Zusammenfassung .....</b>	<b>519</b>

---

<b>8</b>	<b>Browser steuern und Browserinformationen auslesen</b>	521
<b>8.1</b>	<b>Das Browser Object Model</b>	521
<b>8.2</b>	<b>Auf Fensterinformationen zugreifen</b>	523
8.2.1	Die Größe und Position eines Browserfensters ermitteln	523
8.2.2	Die Größe und Position eines Browserfensters ändern	525
8.2.3	Auf Anzeigeeigenschaften der Browserleisten zugreifen	527
8.2.4	Allgemeine Eigenschaften ermitteln	528
8.2.5	Neue Browserfenster öffnen	529
8.2.6	Browserfenster schließen	531
8.2.7	Dialoge öffnen	532
8.2.8	Funktionen zeitgesteuert ausführen	533
<b>8.3</b>	<b>Auf Navigationsinformationen der aktuellen Webseite zugreifen</b>	535
8.3.1	Auf die einzelnen Bestandteile der URL zugreifen	535
8.3.2	Auf Querystring-Parameter zugreifen	536
8.3.3	Eine neue Webseite laden	536
<b>8.4</b>	<b>Den Browserverlauf einsehen und verändern</b>	538
8.4.1	Im Browserverlauf navigieren	538
8.4.2	Browserverlauf bei Single Page Applications	539
8.4.3	Einträge in den Browserverlauf hinzufügen	540
8.4.4	Auf Änderungen im Browserverlauf reagieren	543
8.4.5	Den aktuellen Eintrag im Browserverlauf ersetzen	543
<b>8.5</b>	<b>Browser erkennen und Browserfeatures bestimmen</b>	544
<b>8.6</b>	<b>Auf Informationen des Bildschirms zugreifen</b>	547
<b>8.7</b>	<b>Zusammenfassung</b>	548
<b>9</b>	<b>Inhalte einer Webseite dynamisch nachladen</b>	549
<b>9.1</b>	<b>Das Prinzip von Ajax</b>	549
9.1.1	Synchrone Kommunikation	549
9.1.2	Asynchrone Kommunikation	550
9.1.3	Typische Anwendungsfälle für die Verwendung von Ajax	552
9.1.4	Verwendete Datenformate	555
<b>9.2</b>	<b>Das XML-Format</b>	555
9.2.1	Der Aufbau von XML	555
9.2.2	XML und die DOM API	557

---

9.2.3	Zeichenketten in XML-Objekte umwandeln .....	558
9.2.4	XML-Objekte in Zeichenketten umwandeln .....	559
<b>9.3</b>	<b>Das JSON-Format</b> .....	<b>561</b>
9.3.1	Der Aufbau von JSON .....	561
9.3.2	Unterschied zwischen JSON und JavaScript-Objekten .....	563
9.3.3	Objekte in das JSON-Format umwandeln .....	564
9.3.4	Objekte aus dem JSON-Format umwandeln .....	565
<b>9.4</b>	<b>Anfragen per Ajax stellen</b> .....	<b>566</b>
9.4.1	Das »XMLHttpRequest«-Objekt .....	566
9.4.2	HTML-Daten per Ajax laden .....	573
9.4.3	XML-Daten per Ajax laden .....	577
9.4.4	JSON-Daten per Ajax laden .....	581
9.4.5	Daten per Ajax an den Server schicken .....	584
9.4.6	Formulare per Ajax abschicken .....	585
9.4.7	Daten von anderen Domains laden .....	586
9.4.8	Die neuere Alternative zu »XMLHttpRequest«: die Fetch API .....	589
<b>9.5</b>	<b>Zusammenfassung</b> .....	<b>593</b>
<b>10</b>	<b>Aufgaben vereinfachen mit jQuery</b> .....	<b>595</b>
<hr/>		
<b>10.1</b>	<b>Einführung</b> .....	<b>595</b>
10.1.1	jQuery einbinden .....	596
10.1.2	jQuery über ein CDN einbinden .....	597
10.1.3	jQuery verwenden .....	598
10.1.4	Aufgaben mit jQuery vereinfachen .....	599
<b>10.2</b>	<b>Mit dem DOM arbeiten</b> .....	<b>601</b>
10.2.1	Elemente selektieren .....	601
10.2.2	Auf Inhalte zugreifen und diese verändern .....	606
10.2.3	Ausgewählte Elemente filtern .....	610
10.2.4	Auf Attribute zugreifen .....	612
10.2.5	Auf CSS-Eigenschaften zugreifen .....	613
10.2.6	Zwischen Elementen navigieren .....	614
10.2.7	Effekte und Animationen verwenden .....	616
<b>10.3</b>	<b>Auf Ereignisse reagieren</b> .....	<b>618</b>
10.3.1	Event-Listener registrieren .....	618
10.3.2	Auf allgemeine Ereignisse reagieren .....	619
10.3.3	Auf Mausereignisse reagieren .....	620

10.3.4	Auf Tastaturreignisse reagieren .....	621
10.3.5	Auf Formularereignisse reagieren .....	622
10.3.6	Auf Informationen von Ereignissen zugreifen .....	623
<b>10.4</b>	<b>Ajax-Anfragen erstellen .....</b>	<b>625</b>
10.4.1	Ajax-Anfragen erstellen .....	625
10.4.2	Auf Ereignisse reagieren .....	628
10.4.3	HTML-Daten per Ajax laden .....	629
10.4.4	XML-Daten per Ajax laden .....	631
10.4.5	JSON-Daten per Ajax laden .....	632
<b>10.5</b>	<b>Zusammenfassung .....</b>	<b>633</b>

## **11 Bilder und Grafiken dynamisch erstellen** 641

---

<b>11.1</b>	<b>Bilder zeichnen .....</b>	<b>641</b>
11.1.1	Die Zeichenfläche .....	641
11.1.2	Der Rendering-Kontext .....	642
11.1.3	Rechtecke zeichnen .....	644
11.1.4	Pfade verwenden .....	647
11.1.5	Texte zeichnen .....	653
11.1.6	Farbverläufe zeichnen .....	654
11.1.7	Speichern und Wiederherstellen des Canvas-Zustands .....	656
11.1.8	Transformationen anwenden .....	658
11.1.9	Animationen erstellen .....	661
<b>11.2</b>	<b>Vektorgrafiken einbinden .....</b>	<b>663</b>
11.2.1	Das SVG-Format .....	663
11.2.2	SVG in HTML einbinden .....	665
11.2.3	Das Aussehen von SVG-Elementen mit CSS beeinflussen .....	668
11.2.4	Das Verhalten von SVG-Elementen mit JavaScript beeinflussen .....	669
<b>11.3</b>	<b>Zusammenfassung .....</b>	<b>671</b>

## **12 Moderne Web-APIs verwenden** 673

---

<b>12.1</b>	<b>Über JavaScript kommunizieren .....</b>	<b>675</b>
12.1.1	Unidirektionale Kommunikation mit dem Server .....	675
12.1.2	Bidirektionale Kommunikation mit einem Server .....	677
12.1.3	Vom Server ausgehende Kommunikation .....	679

---

<b>12.2 Nutzer wiedererkennen</b> .....	684
12.2.1 Cookies verwenden .....	684
12.2.2 Cookies anlegen .....	686
12.2.3 Cookies auslesen .....	687
12.2.4 Ein Beispiel: Einkaufswagen auf Basis von Cookies .....	689
12.2.5 Nachteile von Cookies .....	693
<b>12.3 Den Browserspeicher nutzen</b> .....	693
12.3.1 Werte im Browserspeicher speichern .....	694
12.3.2 Werte aus dem Browserspeicher lesen .....	695
12.3.3 Werte im Browserspeicher aktualisieren .....	695
12.3.4 Werte aus dem Browserspeicher löschen .....	696
12.3.5 Auf Änderungen im Browserspeicher reagieren .....	696
12.3.6 Die verschiedenen Typen von Browserspeichern .....	697
12.3.7 Ein Beispiel: Einkaufswagen auf Basis des Browserspeichers .....	699
<b>12.4 Die Browserdatenbank nutzen</b> .....	700
12.4.1 Öffnen einer Datenbank .....	701
12.4.2 Erstellen einer Datenbank .....	702
12.4.3 Erstellen eines Objektspeichers .....	703
12.4.4 Hinzufügen von Objekten zu einem Objektspeicher .....	704
12.4.5 Lesen von Objekten aus einem Objektspeicher .....	708
12.4.6 Löschen von Objekten aus einem Objektspeicher .....	709
12.4.7 Aktualisieren von Objekten in einem Objektspeicher .....	710
12.4.8 Verwendung eines Cursors .....	711
<b>12.5 Auf das Dateisystem zugreifen</b> .....	712
12.5.1 Auswählen von Dateien per Dateidialog .....	713
12.5.2 Auswählen von Dateien per Drag and Drop .....	714
12.5.3 Lesen von Dateien .....	716
12.5.4 Den Lesefortschritt überwachen .....	719
<b>12.6 Komponenten einer Webseite mit Drag and Drop verschieben</b> .....	721
12.6.1 Ereignisse einer Drag-and-Drop-Operation .....	721
12.6.2 Verschiebbare Elemente definieren .....	722
12.6.3 Verschieben von Elementen .....	724
<b>12.7 Aufgaben parallelisieren</b> .....	726
12.7.1 Das Prinzip von Web Workern .....	727
12.7.2 Web Worker verwenden .....	728
<b>12.8 Den Standort von Nutzern ermitteln</b> .....	730
12.8.1 Auf Standortinformationen zugreifen .....	730
12.8.2 Kontinuierlich auf Standortinformationen zugreifen .....	732

<b>12.9 Den Batteriestand eines Endgeräts auslesen</b> .....	733
12.9.1 Auf Batterieinformationen zugreifen .....	733
12.9.2 Auf Ereignisse reagieren .....	734
<b>12.10 Sprache ausgeben und Sprache erkennen</b> .....	736
12.10.1 Sprache ausgeben .....	737
12.10.2 Sprache erkennen .....	739
<b>12.11 Animationen erstellen</b> .....	741
12.11.1 Verwendung der API .....	741
12.11.2 Steuern einer Animation .....	744
<b>12.12 Mit der Kommandozeile arbeiten</b> .....	745
12.12.1 Auswahl und Inspektion von DOM-Elementen .....	745
12.12.2 Analyse von Events .....	747
12.12.3 Debugging, Monitoring und Profiling .....	750
<b>12.13 Mehrsprachige Anwendungen entwickeln</b> .....	754
12.13.1 Begriffserklärungen .....	755
12.13.2 Die Internationalization API .....	756
12.13.3 Vergleich von Zeichenketten .....	758
12.13.4 Formatierung von Datums- und Zeitangaben .....	761
12.13.5 Formatierung von Zahlenwerten .....	764
<b>12.14 Übersicht über verschiedene Web-APIs</b> .....	767
<b>12.15 Zusammenfassung</b> .....	771
<b>13 Objektorientierte Programmierung</b> .....	773
<hr/>	
<b>13.1 Die Prinzipien der objektorientierten Programmierung</b> .....	773
13.1.1 Klassen, Objektinstanzen und Prototypen .....	774
13.1.2 Prinzip 1: Abstraktes Verhalten definieren .....	776
13.1.3 Prinzip 2: Zustand und Verhalten kapseln .....	777
13.1.4 Prinzip 3: Zustand und Verhalten vererben .....	778
13.1.5 Prinzip 4: Verschiedene Typen annehmen .....	779
13.1.6 JavaScript und die Objektorientierung .....	780
<b>13.2 Prototypische Objektorientierung</b> .....	780
13.2.1 Das Konzept von Prototypen .....	781
13.2.2 Von Objekten ableiten .....	781
13.2.3 Methoden und Eigenschaften vererben .....	782
13.2.4 Methoden und Eigenschaften im erbenden Objekt definieren .....	783

---

13.2.5	Methoden überschreiben .....	784
13.2.6	Die Prototypenkette .....	785
13.2.7	Methoden des Prototyps aufrufen .....	786
13.2.8	Prototypische Objektorientierung und die Prinzipien der Objektorientierung .....	787
<b>13.3</b>	<b>Pseudoklassische Objektorientierung</b> .....	<b>787</b>
13.3.1	Konstruktorfunktionen definieren .....	788
13.3.2	Objektinstanzen erzeugen .....	788
13.3.3	Methoden definieren .....	788
13.3.4	Vererbung mit Prototypen .....	789
13.3.5	Konstruktor der »Oberklasse« aufrufen .....	793
13.3.6	Methoden überschreiben .....	793
13.3.7	Methoden der »Oberklasse« aufrufen .....	793
13.3.8	Pseudoklassische Objektorientierung und die Prinzipien der Objektorientierung .....	794
<b>13.4</b>	<b>Objektorientierung mit Klassensyntax</b> .....	<b>794</b>
13.4.1	Klassen definieren .....	795
13.4.2	Objektinstanzen erzeugen .....	797
13.4.3	Getter und Setter definieren .....	797
13.4.4	Private Eigenschaften und private Methoden definieren .....	799
13.4.5	Von »Klassen« ableiten .....	803
13.4.6	Methoden überschreiben .....	807
13.4.7	Methoden der »Oberklasse« aufrufen .....	809
13.4.8	Statische Methoden definieren .....	810
13.4.9	Statische Eigenschaften definieren .....	812
13.4.10	Klassensyntax und die Prinzipien der Objektorientierung .....	814
<b>13.5</b>	<b>Zusammenfassung</b> .....	<b>815</b>
<b>14</b>	<b>Funktionale Programmierung</b> .....	<b>817</b>
<hr/>		
<b>14.1</b>	<b>Prinzipien der funktionalen Programmierung</b> .....	<b>817</b>
14.1.1	Prinzip 1: Funktionen sind Objekte erster Klasse .....	817
14.1.2	Prinzip 2: Funktionen arbeiten mit unveränderlichen Datenstrukturen .....	818
14.1.3	Prinzip 3: Funktionen haben keine Nebeneffekte .....	818
14.1.4	Prinzip 4: Funktionale Programme sind deklarativ .....	818
<b>14.2</b>	<b>Imperative Programmierung und funktionale Programmierung</b> .....	<b>819</b>
14.2.1	Iterieren mit der Methode »forEach()« .....	819
14.2.2	Werte abbilden mit der Methode »map()« .....	822

14.2.3	Werte filtern mit der Methode »filter()« .....	823
14.2.4	Mehrere Werte zu einem Wert reduzieren mit der Methode »reduce()« .....	825
14.2.5	Kombination der verschiedenen Methoden .....	828
<b>14.3</b>	<b>Zusammenfassung</b> .....	<b>829</b>

## **15 Den Quelltext richtig strukturieren** 831

---

<b>15.1</b>	<b>Namenskonflikte vermeiden</b> .....	<b>831</b>
<b>15.2</b>	<b>Module definieren und verwenden</b> .....	<b>835</b>
15.2.1	Das Module-Entwurfsmuster .....	835
15.2.2	Das Revealing-Module-Entwurfsmuster .....	839
15.2.3	AMD .....	843
15.2.4	CommonJS .....	845
15.2.5	Native Module .....	847
<b>15.3</b>	<b>Zusammenfassung</b> .....	<b>850</b>

## **16 Asynchrone Programmierung und weitere fortgeschrittene Features verwenden** 853

---

<b>16.1</b>	<b>Asynchrone Programmierung verstehen und anwenden</b> .....	<b>853</b>
16.1.1	Das Callback-Entwurfsmuster verwenden .....	854
16.1.2	Promises verwenden .....	858
16.1.3	Async Functions verwenden .....	867
<b>16.2</b>	<b>Das Iterieren über Datenstrukturen kapseln</b> .....	<b>871</b>
16.2.1	Das Prinzip von Iteratoren .....	871
16.2.2	Iteratoren verwenden .....	872
16.2.3	Einen eigenen Iterator erstellen .....	872
16.2.4	Ein iterierbares Objekt erstellen .....	874
16.2.5	Über iterierbare Objekte iterieren .....	875
<b>16.3</b>	<b>Funktionen anhalten und fortsetzen</b> .....	<b>876</b>
16.3.1	Eine Generatorfunktion erstellen .....	876
16.3.2	Einen Generator erstellen .....	876
16.3.3	Über Generatoren iterieren .....	877
16.3.4	Unendliche Generatoren erstellen .....	878
16.3.5	Generatoren mit Parametern steuern .....	878

---

<b>16.4</b>	<b>Den Zugriff auf Objekte abfangen</b> .....	879
16.4.1	Das Prinzip von Proxies .....	880
16.4.2	Proxies erstellen .....	880
16.4.3	Handler für Proxies definieren .....	880
<b>16.5</b>	<b>Zusammenfassung</b> .....	885
<b>17</b>	<b>Serverseitige Anwendungen mit Node.js erstellen</b> .....	887
<hr/>		
<b>17.1</b>	<b>Einführung Node.js</b> .....	887
17.1.1	Die Architektur von Node.js .....	887
17.1.2	Installation von Node.js .....	889
17.1.3	Eine einfache Anwendung .....	890
<b>17.2</b>	<b>Node.js-Packages verwalten</b> .....	891
17.2.1	Den Node.js Package Manager installieren .....	891
17.2.2	Packages installieren .....	891
17.2.3	Eigene Packages initialisieren .....	894
<b>17.3</b>	<b>Ereignisse verarbeiten und auslösen</b> .....	898
17.3.1	Ein Event auslösen und abfangen .....	898
17.3.2	Ein Event mehrfach auslösen .....	901
17.3.3	Auf ein Event genau einmal reagieren .....	901
17.3.4	Mehrere Event-Listener für ein Event registrieren .....	902
<b>17.4</b>	<b>Auf das Dateisystem zugreifen</b> .....	903
17.4.1	Dateien lesen .....	903
17.4.2	Dateien schreiben .....	904
17.4.3	Dateiinformatioen auslesen .....	905
17.4.4	Dateien löschen .....	906
17.4.5	Mit Verzeichnissen arbeiten .....	906
<b>17.5</b>	<b>Einen Webserver erstellen</b> .....	908
17.5.1	Einen Webserver starten .....	908
17.5.2	Dateien per Webserver zur Verfügung stellen .....	909
17.5.3	Einen Client für einen Webserver erstellen .....	910
17.5.4	Routen definieren .....	910
17.5.5	Das Web-Framework Express.js verwenden .....	911
<b>17.6</b>	<b>Auf Datenbanken zugreifen</b> .....	916
17.6.1	Installation von MongoDB .....	917
17.6.2	MongoDB-Treiber für Node.js installieren .....	917
17.6.3	Verbindung zur Datenbank herstellen .....	917

17.6.4	Eine Collection erstellen .....	918
17.6.5	Objekte speichern .....	919
17.6.6	Objekte lesen .....	920
17.6.7	Objekte aktualisieren .....	922
17.6.8	Objekte löschen .....	923
<b>17.7</b>	<b>Mit Streams arbeiten</b> .....	924
17.7.1	Einführung und Arten von Streams .....	925
17.7.2	Anwendungsfälle von Streams .....	925
17.7.3	Daten mit Streams lesen .....	926
17.7.4	Daten mit Streams schreiben .....	928
17.7.5	Streams mithilfe von Piping kombinieren .....	929
17.7.6	Fehlerbehandlung beim Piping .....	932
<b>17.8</b>	<b>Zusammenfassung</b> .....	933
<b>18</b>	<b>Mobile Anwendungen mit JavaScript erstellen</b> .....	935
<b>18.1</b>	<b>Die unterschiedlichen Arten mobiler Anwendungen</b> .....	935
18.1.1	Native Anwendungen .....	935
18.1.2	Mobile Webanwendungen .....	936
18.1.3	Hybridanwendungen .....	938
18.1.4	Vergleich der verschiedenen Ansätze .....	939
<b>18.2</b>	<b>Mobile Anwendungen mit React Native erstellen</b> .....	941
18.2.1	Das Prinzip von React Native .....	942
18.2.2	Installation und Projektinitialisierung .....	942
18.2.3	Die Anwendung starten .....	945
18.2.4	Das Grundgerüst einer React-Native-Anwendung .....	948
18.2.5	UI-Komponenten verwenden .....	950
18.2.6	Kommunikation mit dem Server .....	955
18.2.7	Bauen und Veröffentlichen von Anwendungen .....	956
<b>18.3</b>	<b>Zusammenfassung</b> .....	956
<b>19</b>	<b>KI-Anwendungen mit JavaScript</b> .....	959
<b>19.1</b>	<b>Grundlagen und Konzepte</b> .....	960
19.1.1	Wie arbeitet ein LLM? .....	962
19.1.2	Tokens – die Währung eines LLMs .....	962

<b>19.2 Arbeiten mit LLMs auf dem Server</b> .....	965
19.2.1 Vorbereitung: Lokales LLM installieren .....	967
19.2.2 Über Bibliotheken mit dem LLM kommunizieren .....	968
19.2.3 Direkt über die REST-Schnittstelle mit einem LLM kommunizieren .....	972
<b>19.3 Praktische Anwendung von LLMs</b> .....	977
19.3.1 Chatbot mit JavaScript .....	977
19.3.2 Bilderkennung mit TensorFlow.js .....	983
<b>19.4 LangChain – Ein Architekturframework für KI-Applikationen</b> .....	991
19.4.1 Eine erste Kette mit LangChain .....	991
<b>19.5 Zusammenfassung</b> .....	999

## 20 Einen professionellen Entwicklungsprozess aufsetzen

1001

<b>20.1 Den Workflow automatisieren</b> .....	1001
20.1.1 Überblick: Moderne Tools für die Automatisierung .....	1002
20.1.2 Codequalität prüfen mit ESLint .....	1004
20.1.3 Quelltext automatisiert testen mit Vitest .....	1007
20.1.4 Quelltext bündeln und optimieren mit esbuild .....	1020
20.1.5 Zusammenspiel im npm-Workflow .....	1023
<b>20.2 Versionsverwaltung des Quelltexts</b> .....	1024
20.2.1 Einführung in die Versionsverwaltung .....	1024
20.2.2 Das Versionsverwaltungssystem Git installieren und konfigurieren .....	1028
20.2.3 Ein neues lokales Repository anlegen .....	1030
20.2.4 Ein bestehendes Repository klonen .....	1030
20.2.5 Änderungen in den Staging-Bereich übertragen .....	1031
20.2.6 Änderungen in das lokale Repository übertragen .....	1031
20.2.7 Die verschiedenen Zustände in Git .....	1033
20.2.8 Änderungen in das Remote Repository übertragen .....	1034
20.2.9 Änderungen aus dem Remote Repository übertragen .....	1035
20.2.10 In einem neuen Branch arbeiten .....	1036
20.2.11 Änderungen aus einem Branch übernehmen .....	1037
20.2.12 Übersicht über die wichtigsten Befehle und Begriffe .....	1038
<b>20.3 Zusammenfassung</b> .....	1042

<b>Anhang</b>	1043
<b>A JavaScript-Referenz</b> .....	1045
<b>B DOM-Referenz und HTML-Erweiterungen</b> .....	1103
<b>C BOM und Ajax</b> .....	1193
<b>D HTML5-Web-APIs-Referenz</b> .....	1231
Index .....	1281