

HTML und CSS

Das umfassende Handbuch

DAS INHALTS- VERZEICHNIS

» Hier geht's
direkt
zum Buch

Inhalt

Materialien zum Buch	29
Vorwort	31

1 Einführung in das HTML-Universum 37

1.1 Ist dieses Buch überhaupt etwas für mich?	37
1.2 Die verschiedenen Typen von Websites	38
1.2.1 Webpräsenz	38
1.2.2 Blog/Onlinemagazin/Portfolio	39
1.2.3 E-Commerce-Websites – Geschäfte ohne Öffnungszeiten	40
1.2.4 Landingpage/Microsite	41
1.2.5 Webplattform – sich ein eigenes soziales Netzwerk bauen	41
1.2.6 Web-Apps	42
1.3 Dynamische und statische Websites	42
1.3.1 Statische Websites	42
1.3.2 Dynamische Websites	44
1.4 Sprachen für die Gestaltung und Entwicklung im Web	46
1.4.1 HTML – die textbasierte Hypertext-Auszeichnungssprache	46
1.4.2 CSS – die Gestaltungssprache Cascading Stylesheets	46
1.4.3 JavaScript – die clientseitige Skriptsprache des Webbrowsers	47
1.4.4 Die serverseitigen Skriptsprachen und Datenbanken	48
1.4.5 WebAssembly	49
1.5 Was brauche ich, um hier anzufangen?	50
1.5.1 (HTML-)Editor zum Schreiben von HTML-Dokumenten	50
1.5.2 Webbrowser für die Anzeige der Website	52
1.5.3 Schritt für Schritt: eine Webseite erstellen und im Webbrowser betrachten	53
1.5.4 Geschriebenes HTML überprüfen	55
1.5.5 Gute Gründe, den HTML-Code (trotzdem) zu validieren	57
1.6 Verwendete Konventionen im Buch	59
1.7 Zusammenfassung	59

2	Grundlegender Aufbau von HTML(-Dokumenten)	61
2.1	Syntax und Aufbau von HTML(-Dokumenten)	61
2.1.1	Wie wird in HTML ein Dokument strukturiert?	61
2.1.2	Baumstruktur mit dem DOM-Inspektor betrachten	64
2.1.3	Was sind HTML-Tags und HTML-Elemente?	65
2.1.4	HTML-Elemente verschachteln und die hierarchische Struktur	66
2.1.5	Falsche Verschachtelung von HTML-Elementen vermeiden	67
2.1.6	Das Ende-Tag eines HTML-Elements weglassen?	68
2.1.7	Allein stehende HTML-Tags ohne Ende-Tag	69
2.1.8	Zusätzliche HTML-Attribute für HTML-Elemente	70
2.1.9	Kommentare in HTML-Dokumenten verwenden	71
2.2	Ein einfaches HTML-Dokument-Grundgerüst	72
2.2.1	Der HTML-Dokumenttyp <code><!doctype></code>	72
2.2.2	Der Anfang und das Ende eines HTML-Dokuments mit <code><html></code>	73
2.2.3	<code><head></code> – der Kopf eines HTML-Dokuments	74
2.2.4	<code><body></code> – der sichtbare Bereich eines HTML-Dokuments	74
2.2.5	Vorlage eines HTML-Grundgerüsts in der Praxis	74
2.3	Zusammenfassung	75
3	Die Kopfdaten eines HTML-Dokuments	77
3.1	Die HTML-Elemente für den Kopf in der Übersicht	77
3.2	<code><title></code> – die Überschrift der HTML-Seite	78
3.3	Exkurs: Namenskonvention und Referenzierung	80
3.3.1	Gültige und gute Dateinamen für ein HTML-Dokument	80
3.3.2	Gültige Verzeichnisnamen und sinnvolle Verzeichnisstrukturen	81
3.3.3	Referenz auf eine Datenquelle notieren	82
3.4	Die Basis-URL einer Webseite mit <code><base></code> definieren	85
3.5	Beziehung zu einem externen Dokument mit <code><link></code>	87
3.6	Dokumentglobale CSS-Stile mit <code><style></code> notieren	91
3.7	Skripte in Webseiten einbinden mit <code><script></code>	92
3.8	Metainformationen für das Dokument mit <code><meta></code>	94
3.8.1	Die Verwendung von Metaangaben	95
3.8.2	Den Viewport einstellen	96
3.8.3	Nützliche Metadaten für einen Webcrawler angeben	98
3.8.4	Hilfreiche Metadaten für Suchmaschinen	98

3.8.5	Nützliche Metadaten für den Webbrowser	100
3.8.6	Allgemeine Metadaten verwenden	101
3.8.7	Meine Empfehlung: Diese Metadaten gehören in das HTML-Grundgerüst	101
3.8.8	Die HTML-Attribute für das HTML-Element <meta>	102
3.9	Zusammenfassung	103
4	Der sichtbare Bereich eines HTML-Dokuments	105
4.1	HTML-Elemente für Seitenstrukturierung	106
4.1.1	<body> – der darstellbare Inhaltsbereich eines HTML-Dokuments	106
4.1.2	Die Sektionselemente von HTML	107
4.1.3	Inhalt in themenbezogene Abschnitte mit <section> einteilen	107
4.1.4	Inhalt in einen in sich geschlossenen Block mit <article> einteilen	108
4.1.5	Inhalte mit zusätzlichen Informationen mit <aside> ergänzen	109
4.1.6	Einen Inhalt mit <nav> zu einer Seiten-Navigationsleiste erklären	110
4.1.7	Überschriften mit den HTML-Elementen von <h1> bis <h6>	112
4.1.8	Ein Kopfbereich mit <header> und ein Fußbereich mit <footer>	114
4.1.9	Kontaktinformationen mit <address> kennzeichnen	116
4.2	HTML-Elemente für Textstrukturierung	117
4.2.1	Textabsätze mit <p> hinzufügen	117
4.2.2	Zeilenumbruch erzwingen mit 	118
4.2.3	Einen optionalen Zeilenumbruch mit <wbr> hinzufügen	119
4.2.4	Leerzeichen erzwingen und Umbruch verhindern mit » «	120
4.2.5	Thematische Trennung mit <hr> hinzufügen	121
4.2.6	Absätze bzw. Zitate mit <blockquote> hinzufügen	122
4.2.7	Einen allgemeinen Bereich mit <div> gruppieren	122
4.2.8	<main> – ein HTML-Element für den Hauptinhalt	124
4.2.9	Gesonderte Beschriftung von Inhalten mit <figure> und <figcaption>	125
4.2.10	Ungeordnete Listen mit und 	126
4.2.11	Geordnete Listen mit und 	127
4.2.12	Nummerierung einer geordneten Liste umkehren	128
4.2.13	Nummerierung einer geordneten Liste ändern	128
4.2.14	Listen ineinander verschachteln	129
4.2.15	Eine Beschreibungsliste mit <dl>, <dt> und <dd> erstellen	130
4.3	HTML-Elemente für Textauszeichnungen	132
4.3.1	Abkürzungen oder Akronyme mit <abbr> kennzeichnen	134
4.3.2	Text als Quelle eines Arbeitstitels mit <cite> markieren	135
4.3.3	Darstellung von Computercode mit <code> und <pre> auszeichnen	135
4.3.4	Tastatureingabe mit <kbd> und Programmausgabe mit <samp>	137

4.3.5	Einen Text mit <dfn> als eine Definition auszeichnen	137
4.3.6	Text als Variable mit <var> auszeichnen	138
4.3.7	Textrichtung mit <bdo> und <bdi> ändern	138
4.3.8	Text betonen bzw. hervorheben mit , , <i> und 	140
4.3.9	Einen Text mit <mark> hervorheben	141
4.3.10	Text zwischen Anführungsstriche setzen mit <q>	142
4.3.11	Text unter- bzw. durchstreichen mit <u> und <s>	144
4.3.12	Änderungen von Text mit <ins> und markieren	145
4.3.13	Einen Text hochstellen bzw. tiefstellen mit <sup> und <sub>	146
4.3.14	Datums- und Zeitangaben mit <time> kennzeichnen	147
4.3.15	Das Kleingedruckte mit <small> kennzeichnen	150
4.3.16	<ruby>, <rp> und <rt> für eine Anmerkung der Aussprache	150
4.3.17	Bereich von einzelnen Textpassagen mit zusammenfassen	152
4.4	HTML und die Semantik	152
4.5	Barrierearmes Webdesign mit WAI-ARIA	154
4.5.1	Die Rolle eines HTML-Elements	155
4.5.2	Zustand und Attribute eines HTML-Elements mit WAI-ARIA beschreiben	157
4.5.3	Verwenden Sie einen sinnvollen Namen für das HTML-Element	158
4.6	Exkurs: Zeichencodierung	158
4.7	Zeichenentitäten in HTML	160
4.8	Zusammenfassung	162

5 Tabellen und Hyperlinks 163

5.1	Daten in einer Tabelle strukturieren	163
5.1.1	Eine einfache Tabellenstruktur mit <table>, <tr>, <td> und <th>	164
5.1.2	Spalten bzw. Zeilen mit »colspan« bzw. »rowspan« zusammenfassen	166
5.1.3	HTML-Attribute für die Tabellenelemente	169
5.1.4	Tabellen mit <thead>, <tbody> und <tfoot> strukturieren	169
5.1.5	Spalten einer Tabelle gruppieren mit <colgroup> und <col>	172
5.1.6	Tabellen beschriften mit <caption> bzw. <figcaption>	173
5.2	»Elektronische« Verweise aka Hyperlinks mit <a>	175
5.2.1	Links zu anderen HTML-Dokumenten der eigenen Website einfügen	177
5.2.2	Links zu anderen Websites einfügen	179
5.2.3	Links mit dem »target«-Attribut in einem neuen Fenster öffnen	180
5.2.4	E-Mail-Links mit »href=mailto:...«	181
5.2.5	Links zu anderen Inhaltstypen setzen	182
5.2.6	Downloadlinks mit dem »download«-Attribut hinzufügen	183

5.2.7	Links zu bestimmten Teilen einer Webseite setzen	185
5.2.8	Links auf Telefonnummern erstellen	188
5.2.9	Die HTML-Attribute für das HTML-Element <code><a></code>	189
5.3	Zusammenfassung	191
6	Grafiken und Multimedia	193
6.1	Bilder für das Web	194
6.1.1	Das JPEG-Format	194
6.1.2	Das GIF-Format	195
6.1.3	Das PNG-Format	195
6.1.4	SVG für Vektorgrafiken	196
6.1.5	Das WebP-Format	196
6.1.6	Bilder für das Web optimieren	197
6.2	Bilder mit <code></code> einbinden	197
6.2.1	Bilder einem HTML-Dokument hinzufügen	198
6.2.2	Höhe und Breite für die Grafik angeben	201
6.2.3	Bilder beschriften mit <code><figure></code> und <code><figcaption></code>	203
6.2.4	Natives Lazy Loading	205
6.2.5	Alternative Bildgrößen mit »srcset« und »sizes«	207
6.2.6	Die HTML-Attribute für das HTML-Element <code></code>	211
6.3	Das passende Bild mit <code><picture></code> laden	212
6.3.1	Bilder im passenden Seitenverhältnis bzw. den passenden Ausschnitt liefern	212
6.3.2	Verschiedene Dateiformate abfragen und ausliefern	215
6.3.3	HTML-Attribute von <code><source></code>	215
6.4	Ein Icon für die Website hinzufügen (Favicon)	216
6.5	Vektorgrafiken in HTML-Dokumenten verwenden	218
6.5.1	SVG als Grafikreferenz hinzufügen mit <code></code>	219
6.5.2	SVG direkt in die Webseite einbetten mit <code><svg></code>	220
6.5.3	SVG-Tags für die Vektorgrafiken	220
6.5.4	Übersicht über die grafischen Elemente von SVG	221
6.5.5	Weitere Hinweise zur Verwendung von SVG	222
6.6	Grafiken zeichnen mit <code><canvas></code>	223
6.7	Videos mit dem HTML-Element <code><video></code> abspielen	224
6.7.1	Die HTML-Attribute für das HTML-Element <code><video></code>	227
6.7.2	Dem Video Untertitel mit <code><track></code> hinzufügen	228
6.7.3	Videos über YouTube abspielen lassen	231

6.8	Audiodateien mit dem HTML-Element <audio> abspielen	232
6.8.1	Die HTML-Attribute für das HTML-Element <audio>	233
6.9	Andere aktive Inhalte einbinden	234
6.9.1	Das HTML-Element <embed>	235
6.9.2	Das HTML-Element <object>	236
6.9.3	Das HTML-Element <iframe>	236
6.10	Zusammenfassung	238

7 HTML-Formulare und interaktive Elemente 239

7.1	Einen Bereich für Formulare definieren	240
7.2	Die HTML-Eingabefelder für Formulare	242
7.2.1	Ein einzeliges Texteingabefeld mit <input type="text">	242
7.2.2	Ein Passwortheingabefeld mit <input type="password">	243
7.2.3	Ein mehrzeiliges Texteingabefeld mit <textarea>	243
7.2.4	Eine Auswahlliste bzw. Dropdown-Liste mit <select>	244
7.2.5	Eine Gruppe von Radiobuttons mit <input type="radio"> erstellen	246
7.2.6	Ein Textlabel mit <label> hinzufügen	247
7.2.7	Checkboxen mit <input type="checkbox"> verwenden	248
7.2.8	Verschiedene Schaltflächen im Überblick	248
7.2.9	Ein verstecktes Eingabefeld mit <input type="hidden"> verwenden	249
7.3	Spezielle Typen von Eingabefeldern	250
7.3.1	Felder für Dateiapload mit <input type="file"> verwenden	251
7.3.2	Ein Eingabefeld für Farben mit <input type="color">	251
7.3.3	Ein Eingabefeld für ein Datum mit <input type="date">	252
7.3.4	Ein Eingabefeld für eine Uhrzeit mit <input type="time">	253
7.3.5	Eingabefelder für Datum und Uhrzeit	253
7.3.6	Eingabefelder für den Monat und die Woche	253
7.3.7	Ein Eingabefeld für die Suche mit <input type="search">	254
7.3.8	Ein Eingabefeld für E-Mail-Adressen mit <input type="email">	254
7.3.9	Ein Eingabefeld für eine URL mit <input type="url">	255
7.3.10	Ein Eingabefeld für Telefonnummern mit <input type="tel">	255
7.3.11	Ein Eingabefeld für Zahlen mit <input type="number">	255
7.3.12	Ein Eingabefeld für Zahlen eines bestimmten Bereichs	256
7.3.13	Ausgabe von Werten und Berechnungen mit <output>	256
7.4	Die HTML-Attribute für Eingabefelder	257
7.4.1	Den Eingabefokus mit dem HTML-Attribut »autofocus« setzen	257
7.4.2	Autovervollständigung (de)aktivieren mit dem Attribut »autocomplete«	258

7.4.3	Eine Liste mit Vorschlägen mit dem HTML-Attribut »list« und <datalist>	258
7.4.4	Minimale und maximale Werte und die Schrittweite festlegen	259
7.4.5	Das Auswählen oder die Eingabe mehrerer Werte mit »multiple«	259
7.4.6	Reguläre Ausdrücke für Eingabefelder mit »pattern«	259
7.4.7	Ein Platzhalter für ein Eingabefeld mit »placeholder«	260
7.4.8	Eine Eingabe erforderlich machen mit dem Attribut »required«	260
7.4.9	Fehlermeldungen für Eingabefelder steuern	260
7.5	Weitere nützliche Helferlein für Eingabefelder	263
7.5.1	Formularelemente mit dem HTML-Attribut »disabled« deaktivieren	264
7.5.2	Bei Eingabefeldern mit dem Attribut »readonly« nur Lesen erlauben	264
7.5.3	Formularelemente gruppieren mit <fieldset> und <legend>	265
7.5.4	Fortschrittsanzeige mit <progress>	266
7.5.5	Werte visualisieren mit <meter>	266
7.6	Formulardaten mit PHP versenden	267
7.6.1	Die Daten vom Webbrowser zur Weiterverarbeitung übertragen	268
7.6.2	Die POST-Methode	269
7.6.3	Die GET-Methode	270
7.6.4	Die Daten mit einem PHP-Skript verarbeiten	271
7.7	Interaktive HTML-Elemente	274
7.7.1	Inhalte auf-/zuklappen mit <details> und <summary>	274
7.7.2	Eine Dialogbox mit <dialog>	275
7.8	Zusammenfassung	277
8	Einführung in Cascading Stylesheets (CSS)	279
8.1	Das grundlegende Anwendungsprinzip von CSS	280
8.1.1	Aufbau einer CSS-Regel	282
8.1.2	Die Deklaration eines Selektors	283
8.1.3	Kommentare für CSS-Code verwenden	284
8.1.4	Ein paar Hinweise zur Codeformatierung von CSS-Code	285
8.2	Einbindungsmöglichkeiten von CSS in HTML	286
8.2.1	Stilanweisungen direkt im HTML-Tag mit dem HTML-Attribut »style«	286
8.2.2	Stilanweisungen im Dokumentkopf mit dem HTML-Element <style>	287
8.2.3	Stilanweisungen aus einer externen CSS-Datei mit <link> einbinden	288
8.2.4	Kombinieren von CSS-Regeln im Kopfbereich und in externen CSS-Dateien	289
8.2.5	Empfehlung: Trennen Sie HTML und CSS	290
8.2.6	Stilanweisungen aus einer externen CSS-Datei mit »@import« einbinden ...	290

8.2.7	Medienspezifische Stylesheets für bestimmte Ausgabegeräte	291
8.2.8	Medienspezifische Stylesheets mit CSS	292
8.3	CSS im Webbrowser analysieren	293
8.4	Werte an CSS-Eigenschaften übergeben	294
8.4.1	Die verschiedenen Maßeinheiten in CSS	294
8.4.2	Zeichenketten und Schlüsselwörter als Wert für CSS-Eigenschaften	297
8.4.3	Die vielen Möglichkeiten, eine Farbe in CSS zu verwenden	297
8.4.4	Winkelmaße in CSS	304
8.4.5	Werte mit Kurzschreibweise an CSS-Eigenschaft übergeben	305
8.5	Zusammenfassung	306
9	Die Selektoren von CSS	307
9.1	Die einfachen Selektoren von CSS	309
9.1.1	HTML-Elemente mit dem Typselektor ansprechen	309
9.1.2	Klassenselektor – Elemente mit einem bestimmten »class«-Wert ansprechen	312
9.1.3	ID-Selektor – ein Element mit einem bestimmten »id«-Wert ansprechen	315
9.1.4	Universalselektor – alle Elemente in einem Dokument ansprechen	316
9.1.5	Elemente anhand der Attribute mit dem Attributselektor ansprechen	317
9.1.6	Attributselektor für Attribute mit einem bestimmten Attributwert	319
9.1.7	Attributselektor für Attribute mit einem bestimmten Teilwert	322
9.1.8	CSS-Pseudoklassen, die Selektoren für bestimmte Eigenschaften	323
9.1.9	Die komfortablen Struktur-Pseudoklassen von CSS	327
9.1.10	Weitere nützliche Pseudoklassen	333
9.1.11	Pseudoelemente, die Selektoren für nicht vorhandene Elemente	334
9.2	Kombinatoren – die Selektoren verketten	337
9.2.1	Der Nachfahrenselektor (E1 E2)	339
9.2.2	Der Kindselektor (E1 > E2)	340
9.2.3	Der Nachbarselektor (E1 + E2)	341
9.2.4	Der Geschwisterselektor (E1 ~ E2)	342
9.3	Empfehlung: So verwenden Sie effizientes und einfaches CSS	343
9.3.1	So schreiben Sie performantes CSS	343
9.3.2	Empfehlung: Halten Sie den CSS-Code so einfach wie möglich	345
9.4	Zusammenfassung	346

10 Die Vererbung und die Kaskade	349
10.1 Das Prinzip der Vererbung in CSS	349
10.1.1 Vorsicht bei der Verwendung von relativen Eigenschaften	353
10.1.2 Es wird nicht alles weitervererbt	353
10.1.3 Vererbung erzwingen mit »inherit«	354
10.1.4 Den Standardwert einer CSS-Eigenschaft wiederherstellen (»initial«)	356
10.1.5 Vererben erzwingen oder Wert wiederherstellen (»unset«)	356
10.1.6 Vererben erzwingen oder Werte wiederherstellen für alle Eigenschaften	356
10.2 Das Regelsystem der Kaskade verstehen	358
10.2.1 Die Herkunft eines Stylesheets	359
10.2.2 Die Priorität einer CSS-Eigenschaft mit »!important« erhöhen	360
10.2.3 Sortierung nach Wichtigkeit und Herkunft	360
10.2.4 Sortieren nach Gewichtung der Selektoren (Spezifität)	362
10.2.5 Zusammenfassung zum Regelsystem der Kaskade	366
10.2.6 Die Kaskade im Browser analysieren	366
10.3 Zusammenfassung	367
11 Das Box-Modell von CSS	369
11.1 Das klassische Box-Modell von CSS	370
11.1.1 Den Inhaltsbereich mit »width« und »height« vorgeben	370
11.1.2 Den Innenabstand mit »padding« angeben	374
11.1.3 Den Rahmen mit »border« erstellen	375
11.1.4 Den Außenabstand mit »margin« einrichten	375
11.1.5 »margin«, »border«, »padding« und »content« mit den Entwicklertools analysieren	377
11.1.6 Collapsing Margins (zusammenfallende vertikale Außenabstände)	378
11.1.7 Gesamtbreite einer Box ermitteln	380
11.2 Das alternative Box-Modell von CSS	382
11.2.1 Das Box-Modell »box-sizing« verwenden	383
11.2.2 Das alternative Box-Modell einsetzen	383
11.3 Logische Eigenschaften	386
11.4 Das Box-Modell für Inline-Elemente	389
11.5 Boxen gestalten	390
11.5.1 Einen Rahmen mit der Eigenschaft »border« hinzufügen und gestalten	390
11.5.2 Hintergrundfarbe mit »background-color« festlegen	394
11.5.3 Hintergrundgrafiken verwenden	394

11.5.4	Die Boxen durchsichtig gestalten	401
11.5.5	Einen Farbverlauf hinzufügen	402
11.5.6	Einen Schlagschatten mit der Eigenschaft »box-shadow« hinzufügen	405
11.5.7	Runde Ecken mit der CSS-Eigenschaft »border-radius« hinzufügen	406
11.6	Exkurs: Webbrowser-Präfixe (CSS Vendor Prefixes)	409
11.7	Zusammenfassung	411

12 CSS-Positionierung 413

12.1	Der normale Elementfluss	413
12.1.1	Block-Elemente	414
12.1.2	Inline-Elemente	414
12.1.3	Mit »display« die Art der Box festlegen	414
12.2	Positionierung mit der CSS-Eigenschaft »position«	416
12.2.1	Die normale Positionierung (»position: static«)	417
12.2.2	Platzierung von Elementen mit »top«, »right«, »bottom« und »left«	418
12.2.3	Die relative Positionierung (»position: relative«)	419
12.2.4	Die absolute Positionierung (»position: absolute«)	420
12.2.5	Die fixe Positionierung (»position: fixed«)	424
12.2.6	Die haftende Positionierung (»position: sticky«)	426
12.3	Übereinanderstapeln regeln mit »z-index«	428
12.4	Schwebende Boxen für die Positionierung mit »float«	432
12.4.1	Das Umfließen beenden	435
12.4.2	Floats zu einer Einheit zusammenfassen mit »flow-root«	436
12.4.3	Nicht rechteckige Formen umfließen lassen	438
12.4.4	Die CSS-Eigenschaft »clip-path«	443
12.5	Zusammenfassung	447

13 Flexboxen 449

13.1	Den Flexbox-Container erstellen	450
13.1.1	Die Fließrichtung anpassen	452
13.1.2	Flex-Items im Flexbox-Container in der Zeile umbrechen	454
13.2	Flex-Items anpassen	455
13.2.1	Wachstumsfaktor von Flex-Items mit »flex-grow« festlegen	455
13.2.2	Schrumpffaktor von Flex-Items mit »flex-shrink« festlegen	456

13.2.3	Die Basisbreite von Flex-Items mit »flex-basis« festlegen	456
13.2.4	Die Reihenfolge der Flex-Items mit »order« bestimmen	460
13.3	Die Flex-Items im Flex-Container ausrichten	460
13.3.1	Flex-Items an der Hauptachse des Flexbox-Containers ausrichten	461
13.3.2	Ausrichtung der Flex-Items entlang der Querachse des Flexbox-Containers ..	462
13.3.3	Extrabehandlung für ein Flex-Item im Flexbox-Container	464
13.3.4	Flex-Items bei einem mehrzeiligen Flexbox-Container ausrichten	466
13.3.5	Leerraum zwischen den Flex-Items hinzufügen	467
13.4	Beispiel: Navigationsbar mit Flexboxen	468
13.5	Zusammenfassung	471
14	Responsives Webdesign und Media Queries	473
14.1	Grundlagenwissen zum responsiven Webdesign	473
14.1.1	Verwenden Sie immer die Viewport-Meta-Angabe	475
14.1.2	Die Medienabfragen für Medieneigenschaften	478
14.1.3	Layoutumbrüche (Breakpoints)	485
14.1.4	Flexible und responsive Bilder	490
14.1.5	Typografie	491
14.2	Die Druckversion einer Website	492
14.2.1	Inhalte ausblenden	493
14.2.2	Weblinks für den Druck sichtbar machen	493
14.2.3	Seitenumbrüche beim Drucken steuern	494
14.2.4	Layout anpassen bzw. vereinfachen	495
14.3	Auf Benutzereinstellungen reagieren	496
14.3.1	Der Dark Mode mit »prefers-color-scheme«	496
14.3.2	Stärkere Kontraste anbieten mit »prefers-contrast«	501
14.3.3	Erzwungene Farben mit »forced-colors« ermitteln	503
14.3.4	Verzicht auf Bewegung mit »prefers-reduced-motion«	505
14.3.5	Datenverbrauch mit »prefers-reduced-data« reduzieren	506
14.4	Container Queries	507
14.5	Berechnungen mit CSS und der »calc()«-Funktion	512
14.6	Zusammenfassung	513

15 CSS Grid 515

15.1 Grid-Container und Rasterelemente verwenden	516
15.1.1 Einen Grid-Container mit Spalten und Zeilen erzeugen	517
15.1.2 Grid-Items im Raster platzieren	520
15.1.3 Layoutänderung leichtgemacht	527
15.1.4 Abstände zwischen den Rasterzeilen	528
15.1.5 Raster im Webbrowser überprüfen	528
15.2 Responsives Grid ohne Media Queries	529
15.2.1 »auto-fill« und die »minmax()«-Funktion	531
15.2.2 auto-fit vs. auto-fill	532
15.2.3 Grid-Items aufspannen und Lücken füllen	533
15.3 Ausrichten von Grid-Items	535
15.3.1 Vertikale Ausrichtung mit »align-items« und »align-self« kontrollieren	536
15.3.2 Horizontale Ausrichtung mit »justify-items« und »justify-self« kontrollieren	537
15.3.3 Gesamten Inhalt im Grid-Container mit »align-content« und »justify-content« ausrichten	538
15.4 Grid-Bereiche benennen	539
15.5 Zusammenfassung	542

16 Stylen mit CSS 543

16.1 Schriftfamilien und -formatierung mit »font«	543
16.1.1 Schriftarten mit »font-family« auswählen	544
16.1.2 Schriftgröße mit »font-size« festlegen	548
16.1.3 Typografische Auszeichnungen mit »font-style« und »font-weight«	555
16.1.4 Zeilenabstand mit »line-height« definieren	556
16.1.5 Kapitälchen mit »font-variant« erstellen	557
16.1.6 Die Kurzschreibweise der Schriftformatierung mit »font«	558
16.2 »@font-face«, Webfonts und variable Fonts	559
16.2.1 Verschiedene Dateiformate für Webfonts	560
16.2.2 Schriften von Google in die Website einbinden	561
16.2.3 Ladestatus von Schriften mit »font-display« überprüfen	564
16.2.4 Performance-Tipps für Webfonts	565
16.2.5 Variable Fonts	566
16.3 Weitere Anpassungsmöglichkeiten von Text	567
16.3.1 Buchstaben- und Wortabstände mit »letter-spacing« und »word-spacing« festlegen	567

16.3.2	Die Textausrichtung mit »text-align« festlegen	569
16.3.3	Den Text mit »text-indent« einrücken	570
16.3.4	Unter- und Durchstreichen von Text mit »text-decoration«	571
16.3.5	Groß- und Kleinschreibung von Text mit »text-transform«	572
16.3.6	Dem Text Schlagschatten mit »text-shadow« hinzufügen	573
16.3.7	Text mit »column-count« in mehrere Spalten aufteilen	574
16.4	Listen mit CSS gestalten	576
16.4.1	Aufzählungspunkte mit »list-style-type« anpassen	576
16.4.2	Bilder als Aufzählungszeichen mit »list-style-image« verwenden	578
16.4.3	Aufzählungsliste mit »list-style-position« positionieren	578
16.4.4	Die Kurzschreibweise »list-style« für die Listengestaltung	579
16.4.5	Das ::marker-Pseudoelement	580
16.5	Schönere Tabellen mit CSS gestalten	582
16.5.1	Tabellen mit fester Breite erstellen	582
16.5.2	Allgemeines Rezept: Tabelle schön mit CSS gestalten	583
16.5.3	Kollabierende Rahmen für Tabellenzellen mit »border-collapse«	585
16.5.4	Abstand zwischen den Zellen mit »border-spacing« einstellen	585
16.5.5	Leere Tabellenzellen anzeigen mit »empty-cells«	586
16.5.6	Tabellenbeschriftung mit »caption-side« positionieren	587
16.6	Bilder mit CSS gestalten	588
16.6.1	Das Bild in einem Container mit »object-fit« und »object-position« anpassen	588
16.6.2	Flächendeckende Hintergrundbilder verwenden	591
16.6.3	Festes Seitenverhältnis für Bilder und Objekte festlegen	594
16.6.4	Filtereffekte mit CSS	596
16.7	Elemente transformieren mit CSS	598
16.7.1	HTML-Elemente skalieren mit »transform: scale()«	598
16.7.2	HTML-Elemente drehen mit »transform: rotate()«	599
16.7.3	HTML-Elemente neigen mit »transform: skew()«	599
16.7.4	HTML-Elemente verschieben mit »transform: translate()«	599
16.7.5	Verschiedene Transformationen kombinieren	600
16.7.6	Es geht auch mit anderen HTML-Elementen	601
16.8	Übergänge und Animationen mit CSS erstellen	601
16.9	HTML-Formulare mit CSS stylen	605
16.9.1	Ein HTML-Formular ordentlich strukturieren	605
16.9.2	Ausrichten der Formularelemente mit CSS	607
16.9.3	Gestalten der Formularelemente mit CSS	610
16.10	Zusammenfassung	614

17	Der CSS-Präprozessor Sass und SCSS	615
17.1	Sass oder SCSS	616
17.2	Von Sass/SCSS zu CSS	617
17.3	Einführung in Sass	617
17.3.1	Sass installieren und einrichten	617
17.3.2	Variablen mit Sass verwenden	622
17.3.3	Verschachtelungen mit Sass	624
17.3.4	Mixins (@mixin, @include)	626
17.3.5	Extend (@extend)	629
17.3.6	Media Queries und @content	631
17.3.7	Operatoren	634
17.3.8	Farben und Helligkeit anpassen	635
17.3.9	Kontrollstrukturen von Sass	638
17.3.10	Funktionen @function	641
17.3.11	@import	643
17.3.12	Kommentare	644
17.4	Custom Properties mit reinem CSS	644
17.4.1	Was sind Custom Properties?	645
17.4.2	Verwendung von Custom Properties mit »var()«	646
17.4.3	Erlaubte Werte mit Custom Properties verwenden	646
17.4.4	Werte aufteilen und kurzfristig ändern	647
17.4.5	JavaScript und Custom Properties	648
17.4.6	Fazit zu den Custom Properties	650
17.5	Verschachteltes CSS	651
17.5.1	Wie man die Verschachtelung anwendet	651
17.5.2	Media Queries mit verschachteltem CSS	656
17.5.3	Nicht verwandte Selektoren	658
17.5.4	Achtung bei Deklarationen nach der Verschachtelung	659
17.6	Zusammenfassung	659
18	Eine kurze Einführung in JavaScript	661
18.1	JavaScript in der Webentwicklung	662
18.2	JavaScript-Programme schreiben und ausführen	664
18.2.1	Eine JavaScript-Datei in eine HTML-Datei einbinden	665
18.2.2	JavaScript innerhalb von HTML schreiben	667

18.2.3	Position des JavaScripts und seine Ausführung im HTML-Dokument	668
18.2.4	Attribute, um das Ladeverhalten von JavaScript zu beeinflussen (async, defer)	669
18.2.5	Das <noscript>-Element für kein JavaScript	670
18.3	Ausgabe von JavaScript	670
18.3.1	Standarddialoge (und Eingabedialog)	670
18.3.2	Ausgabe auf die Konsole	671
18.3.3	Ausgabe auf die Webseite	674
18.3.4	JavaScript ohne Webbrowser ausführen	675
18.3.5	Den JavaScript-Code mit Anmerkungen kommentieren	677
18.4	Die Verwendung von Variablen in JavaScript	677
18.4.1	Konstanten definieren	680
18.4.2	Strikter Modus mit »use strict«	681
18.5	Übersicht über die JavaScript-Datentypen	681
18.5.1	Number-Datentyp (Zahlen)	682
18.5.2	String-Datentypen (Zeichenketten)	682
18.5.3	Template-Strings	685
18.5.4	Boolean-Datentyp	685
18.5.5	Undefined- und Null-Datentypen	686
18.5.6	Objekte	686
18.5.7	Datentypen konvertieren	687
18.6	Arithmetische Operatoren zum Rechnen in JavaScript	688
18.7	Bedingte Anweisungen in JavaScript	690
18.7.1	»true« oder »false«: boolescher Wahrheitswert	691
18.7.2	Die verschiedenen Vergleichsoperatoren in JavaScript verwenden	692
18.7.3	Die »if«-Verzweigung verwenden	693
18.7.4	Den Auswahloperator verwenden	694
18.7.5	Logische Operatoren	695
18.7.6	Nullish Coalescing Operator	696
18.7.7	Mehrfachverzweigung mit »switch«	697
18.8	Programmanweisungen mehrmals wiederholen mit Schleifen	698
18.8.1	Inkrement- und Dekrementoperator	698
18.8.2	Die kopfgesteuerte »for«-Schleife	699
18.8.3	Die kopfgesteuerte »while«-Schleife	700
18.8.4	Die fußgesteuerte »do-while«-Schleife	701
18.8.5	Den Anweisungsblock mit »break« beenden	701
18.8.6	Mit »continue« zum Schleifenanfang springen	702
18.9	Zusammenfassung	702

19 Arrays, Funktionen und Objekte in JavaScript	703
19.1 Funktionen in JavaScript	703
19.1.1 Die verschiedenen Möglichkeiten, eine Funktion in JavaScript zu definieren	704
19.1.2 Funktionen aufrufen und Funktionsparameter	706
19.1.3 Rückgabewert einer Funktion	709
19.1.4 Der Gültigkeitsbereich von Variablen in einer Funktion	710
19.1.5 Funktionen in der Kurzschreibweise definieren (Arrow Functions)	712
19.1.6 Funktion in einer Webseite verwenden	713
19.2 Arrays	715
19.2.1 Zugriff auf die einzelnen Elemente im Array	717
19.2.2 Mehrdimensionale Arrays	718
19.2.3 Einem Array neue Elemente hinzufügen bzw. entfernen	719
19.2.4 Arrays sortieren	724
19.2.5 Suchen in Arrays	726
19.2.6 Weitere Methoden für Arrays	726
19.3 Zeichenketten und reguläre Ausdrücke	727
19.3.1 Nützliche Funktionen für Zeichenketten	728
19.3.2 Reguläre Ausdrücke auf Zeichenketten anwenden	728
19.4 Objektorientierte Programmierung in JavaScript	729
19.4.1 Was sind Objekte genau?	729
19.4.2 Objekte über Konstruktorfunktionen erstellen	731
19.4.3 Objekte mit der Klassensyntax erzeugen	732
19.4.4 Zugriff auf die Objekteigenschaften und -methoden – Setter und Getter	733
19.4.5 Das Schlüsselwort »this«	735
19.5 Weitere globale Objekte	736
19.5.1 Das oberste Objekt »Object«	737
19.5.2 Objekte für die primitiven Datentypen Number, String und Boolean	737
19.5.3 »Function«-Objekt	739
19.5.4 »Date«-Objekt	739
19.5.5 »Math«-Objekt	740
19.5.6 »Map«-Objekt	740
19.5.7 »Set«-Objekt	740
19.6 Zusammenfassung	741

20 Webseiten dynamisch ändern	743
20.1 Einführung in das DOM eines HTML-Dokuments	743
20.2 Das »document«-Objekt	745
20.3 Die Programmierschnittstelle von DOM	745
20.4 Auf Elemente im DOM zugreifen	747
20.4.1 Ein HTML-Element mit einem bestimmten »id«-Attribut suchen	748
20.4.2 HTML-Elemente mit einem bestimmten Tag-Namen suchen	749
20.4.3 HTML-Elemente mit einem bestimmten »class«-Attribut suchen	752
20.4.4 HTML-Elemente mit einem bestimmten »name«-Attribut suchen	752
20.4.5 »querySelector()« und »querySelectorAll()« verwenden	754
20.4.6 Weitere Objekt- und Eigenschaftensammlungen	756
20.5 Auf JavaScript-Events reagieren	759
20.6 Mit dem Event-Handler die Events behandeln	761
20.6.1 Event-Handler als HTML-Attribut im HTML-Element einrichten (veraltet)	761
20.6.2 Event-Handler als Eigenschaft eines Objekts einrichten	762
20.6.3 Einen Event-Handler mit »addEventListener()« einrichten (Best Practice)	763
20.7 Gängige JavaScript-Events in der Übersicht	765
20.7.1 Die JavaScript-Events der Benutzeroberfläche (Window-Events)	765
20.7.2 JavaScript-Events, die in Verbindung mit der Maus auftreten können	767
20.7.3 JavaScript-Events für Geräte mit einem Touchscreen	768
20.7.4 JavaScript-Events, die in Verbindung mit der Tastatur auftreten	769
20.7.5 JavaScript-Events für HTML-Formulare	770
20.7.6 JavaScript-Events für die Web-APIs	770
20.8 Standardaktion von Events unterdrücken	770
20.9 Der Event-Fluss (Event-Propagation)	771
20.9.1 Mehr zur Bubbling-Phase	772
20.9.2 Bubbling mit der Methode »stopPropagation()« abbrechen	773
20.9.3 Eingreifen in den Event-Fluss während der Capturing-Phase	775
20.9.4 Mehr Informationen zu Capturing-Phase und Bubbling-Phase	776
20.10 Weitere Informationen zu Events mit dem »event«-Objekt	776
20.11 HTML-Attribut oder den Textknoten ändern	779
20.11.1 Den Inhalt von HTML-Elementen mit »innerHTML« ändern	780
20.11.2 Den Wert eines HTML-Attributs ändern	781
20.12 HTML-Elemente hinzufügen, ändern, entfernen	783
20.12.1 Neues HTML-Element und Inhalt erzeugen und hinzufügen	784
20.12.2 HTML-Elemente noch gezielter im DOM-Baum ansteuern	786

20.12.3	Ein neues HTML-Element gezielter dem DOM-Baum hinzufügen	790
20.12.4	Vorhandenes HTML-Element vom DOM-Baum löschen	792
20.12.5	Ein HTML-Element im DOM-Baum durch ein anderes ersetzen	793
20.12.6	Einen Knoten oder ganze Fragmente des DOM-Baumes klonen	794
20.12.7	Verschiedene Methoden, die HTML-Attribute zu manipulieren	795
20.12.8	Das <template>-HTML-Tag	798
20.13	CSS und JavaScript	801
20.13.1	Direktformatierung über Inline-Styles	801
20.13.2	Mehrere Styles auf einmal anwenden	802
20.13.3	CSS-Klassen mit JavaScript ändern	803
20.14	HTML-Formulare und JavaScript	804
20.14.1	Texteingabefelder mit JavaScript einlesen	805
20.14.2	Auswahllisten mit JavaScript einlesen	806
20.14.3	Radioschaltflächen und Checkboxes mit JavaScript einlesen	807
20.14.4	Schaltflächen mit JavaScript abfangen	809
20.14.5	Die Fortschrittsanzeige <progress> mit JavaScript steuern	810
20.15	Zusammenfassung	811
21	Einführung in die Web-APIs (Browser-APIs)	813
21.1	Video- und Audio-Media-API	814
21.1.1	Ein Video mit JavaScript und der Media-API steuern	815
21.1.2	Eine Audiodatei mit JavaScript und der Media-API steuern	819
21.2	Den Standort ermitteln mit der Geolocation-API	821
21.2.1	Die Geolocation-API in einem HTML-Dokument verwenden	822
21.2.2	Fehler und Zugriffsrechte der Geolocation-API behandeln	824
21.2.3	Feintuning mit weiteren Optionen der Geolocation-API	826
21.2.4	Die Position des Benutzers dauerhaft überwachen	828
21.2.5	Die Position des Benutzers auf einer Karte anzeigen	829
21.3	Web Storage – Datenbank für Offlineanwendungen	833
21.3.1	Cookies mit JavaScript realisieren	837
21.3.2	Cookies vs. Web Storage	842
21.3.3	Weitere Offlinetechnologien im Überblick	842
21.3.4	Die Internetverbindung des Benutzers prüfen	843
21.4	Web Workers – die Helfer im Hintergrund	845
21.5	Aktualisieren ohne Anfrage mit Server-Sent Events	853

21.6 Weitere interessante Web-APIs im Schnelldurchlauf	857
21.6.1 Kommunikation in Echtzeit mit den WebSockets	857
21.6.2 Sprach- und Videotelefonie mit WebRTC	859
21.6.3 3D-Grafiken und Spiele mit WebGL	860
21.6.4 Der Umgang mit lokalen Dateien mit der File-API	860
21.6.5 Übersicht aller Web-APIs	863
21.7 Zusammenfassung	864
22 Eine Einführung in Ajax und die Fetch-API	865

22.1 Eine Einführung in die Ajax-Programmierung	865
22.1.1 Ein einfaches Ajax-Beispiel bei der Ausführung	867
22.1.2 Das »XMLHttpRequest«-Objekt erzeugen	869
22.1.3 Eine Anfrage an den Server stellen	869
22.1.4 Daten mitsenden	870
22.1.5 Den Status des »XMLHttpRequest«-Objekts ermitteln	871
22.1.6 Die Antwort vom Server weiterverarbeiten	873
22.1.7 Das Ajax-Beispiel bei der Ausführung	873
22.1.8 Ein komplexeres Ajax-Beispiel mit XML und DOM	874
22.1.9 Das JSON-Datenformat mit Ajax	879
22.2 Die Fetch-API	883
22.2.1 Die Fetch-API verwenden	883
22.2.2 Die Fetch-API mit Promises verwenden	885
22.2.3 POST-Anfragen mit der Fetch-API verwenden	886
22.3 Zusammenfassung	887
23 Das Bootstrap-Framework einsetzen	889

23.1 Bootstrap herunterladen und integrieren	890
23.2 Das Layout mit dem Bootstrap-Framework erstellen	893
23.3 Die Komponenten des Bootstrap-Frameworks verwenden	905
23.4 Übersicht über weitere beliebte CSS-Frameworks	915
23.5 Zusammenfassung	916

24 Einführung in die grundlegenden Konzepte von React	917
24.1 Was ist React?	917
24.2 Die Umgebung für React einrichten	918
24.3 Ein React-Projekt einrichten und starten	919
24.4 Hallo Welt in React	922
24.5 JSX (JavaScript XML)	924
24.5.1 JSX für die UI verwenden	924
24.5.2 JSX bei HTML-Attributen	927
24.5.3 Allein stehende Elemente müssen geschlossen werden	928
24.5.4 Fallunterscheidung	928
24.5.5 Listen mit React	929
24.6 Rendern von Elementen	932
24.7 React-Komponenten und Props	932
24.7.1 Funktionskomponente	934
24.7.2 Klassenkomponenten	937
24.7.3 Props	939
24.8 Statusmanagement in React	941
24.8.1 Das »state«-Objekt in der Klassenkomponente verwenden	943
24.8.2 Das »state«-Objekt verändern	944
24.8.3 Children-Props	946
24.9 Ereignisse und deren Handler	948
24.9.1 Argumente an Event-Handler übergeben	949
24.10 Lifecycle von Komponenten	949
24.10.1 Mounting	950
24.10.2 Updating	955
24.10.3 Unmounting	957
24.10.4 React.Fragment	959
24.11 Die React-Hooks-API	960
24.11.1 Lokaler State in Funktionskomponenten mit »useState«	961
24.11.2 Der Komponenten-Lifecycle mit »useEffect«	963
24.11.3 Zugriff auf den Kontext mit »useContext«	965
24.11.4 Custom Hooks	967
24.12 CSS und React	968
24.12.1 Inline-Styling mit dem »style«-Attribut	970
24.12.2 CSS-Klassen in JSX verwenden	973

24.12.3 CSS-Module in React	975
24.12.4 CSS in JavaScript – Styled Components	976
24.13 Formulare in React	979
24.13.1 Controlled Components	980
24.13.2 Uncontrolled Components	982
24.13.3 Zugriff auf das native DOM über eine Referenz	983
24.14 So geht es weiter	984

25 Einführung in die grundlegenden Konzepte von Angular

987

25.1 Die nötigen Werkzeuge	987
25.2 Hallo Welt in Angular	988
25.2.1 Übersicht der Projektdateien	990
25.2.2 Übersicht der ng-Befehle	991
25.3 Eine kleine Einführung in TypeScript	992
25.3.1 Typisierung	994
25.3.2 Interfaces	997
25.3.3 Export und Import	999
25.3.4 Klassen	1000
25.3.5 Weiteres	1005
25.4 Komponenten	1005
25.4.1 Komponenten erstellen	1006
25.4.2 Komponenten mit CLI erstellen	1011
25.5 Komponenten binden	1013
25.5.1 String-Interpolation	1014
25.5.2 Property-Binding	1016
25.5.3 Event-Binding	1018
25.5.4 Two-Way-Binding	1023
25.6 Pipes	1025
25.6.1 Vordefinierte Pipes	1026
25.6.2 Eigene Pipes	1030
25.7 Der Datenfluss zwischen den Komponenten	1034
25.7.1 Input-Property (Eingehende Daten)	1037
25.7.2 Output-Property (Ausgehende Daten)	1039
25.8 Styles für Komponenten	1041
25.8.1 Style als externe Datei mit »styleUrls« in »@Component«	1042

25.8.2	»styles«-Eigenschaft in »@Component«	1042
25.8.3	Inline-Style im HTML-Template	1043
25.8.4	Exkurs: Das Bootstrap-Framework in Angular verwenden	1043
25.9	Der Lebenszyklus einer Komponente	1044
25.9.1	Der Konstruktor	1045
25.9.2	Reagieren auf Änderungen – »ngOnChanges«	1046
25.9.3	Die Komponente initialisieren – »ngOnInit«	1047
25.9.4	In den ChangeDetection-Lauf eingreifen – »ngDoCheck«	1047
25.9.5	ViewChild und ContentChild	1048
25.9.6	Aufräumarbeiten mit »ngOnDestroy«	1048
25.10	Direktiven	1048
25.10.1	Strukturdirektiven	1049
25.10.2	Attributdirektiven »ngClass« und »ngStyle«	1057
25.10.3	Eigene Direktiven erstellen	1061
25.11	Wie geht es nun weiter?	1063

26 Beispielprojekt erstellen, testen und optimieren 1065

26.1	Projektplanung	1066
26.1.1	Wer ist Ihre Zielgruppe?	1066
26.1.2	Den Inhalt planen	1066
26.2	Grundgerüst mit Inhalt erstellen	1068
26.3	Das Layout der Website festlegen und erstellen	1070
26.3.1	Das Layout planen	1070
26.3.2	Das grundlegende Layout erstellen	1071
26.4	Typografie – Auswahl der Schriften	1077
26.5	Navigation	1080
26.6	Farben für das Projekt festlegen	1082
26.7	Grafiken und Bilder	1083
26.8	Website auf Funktionalität testen	1087
26.8.1	Browstertests	1087
26.8.2	Layoutumbrüche testen	1089
26.8.3	HTML und CSS validieren	1090
26.9	Website auf Barrierefreiheit hin testen	1091

26.10 Performance der Website testen und verbessern	1092
26.10.1 Speed-Tests	1092
26.10.2 Ungenutzten Code entfernen	1094
26.10.3 Serveranfragen reduzieren	1095
26.10.4 Dateigröße reduzieren	1096
26.11 Nachhaltige Webentwicklung	1097
26.11.1 CO2-Bilanz einer Website messen	1098
26.11.2 So geht nachhaltige Webentwicklung	1099
26.11.3 Nachhaltig bei Design und Inhalt	1099
26.11.4 Nachhaltig bei der Webentwicklung	1103
26.11.5 Nachhaltig beim Webhosting	1105
26.11.6 Fazit zur Nachhaltigkeit bei der Webentwicklung	1106
26.12 Zusammenfassung	1106
 Index	 1107

Materialien zum Buch

Auf der Webseite zu diesem Buch stehen folgende Materialien für Sie zum Download bereit:

- ▶ **alle Beispielprogramme**
- ▶ **Bonuskapitel »Verweissensitive Grafiken aka Image-Maps erstellen« und die Anhänge A »HTML-Referenz«, B »CSS-Referenz«, C »JavaScript- und Browserobjekte«**

Gehen Sie auf www.rheinwerk-verlag.de/5767. Klicken Sie auf den Reiter MATERIALIEN. Sie sehen die herunterladbaren Dateien samt einer Kurzbeschreibung des Dateiinhalts. Klicken Sie auf den Button HERUNTERLADEN, um den Download zu starten. Je nach Größe der Datei (und Ihrer Internetverbindung) kann es einige Zeit dauern, bis der Download abgeschlossen ist.