


Diese Leseprobe haben Sie beim
 edv-buchversand.de heruntergeladen.
Das Buch können Sie online in unserem
Shop bestellen.

[Hier zum Shop](#)

Inhalt

Materialien zum Buch	26
Geleitwort	27
Vorwort	29

1 Wissenswertes über Webseiten 33

1.1 Webseiten sehen bei jedem Benutzer anders aus	33
1.2 Webseiten bestehen aus Quelltext	34
1.3 Quelltext besteht aus HTML, CSS und JavaScript	36
1.3.1 Der Inhalt: HTML ist nicht hübsch, aber flexibel	36
1.3.2 Das Styling: CSS gestaltet das HTML	37
1.4 Webseiten werden von einem Browser dargestellt	39
1.4.1 Die bekanntesten Browser: Chrome, Firefox, Safari, Edge und Co	39
1.4.2 Viele Browser sind miteinander verwandt	40
1.4.3 Besonderheiten: Browser unter iOS und Internet Explorer	40
1.5 Editoren zum Schreiben und Bearbeiten von Quelltext	41
1.6 Referenzen und Nachschlagewerke zu HTML und CSS	42
1.6.1 SelfHTML – das deutschsprachige Urgestein	42
1.6.2 Die »MDN Web Docs« – best in English	43
1.6.3 Anlaufstelle für Fragen zur Browserunterstützung: »caniuse.com«	44
1.7 Auf einen Blick	44

2 HTML kennenlernen: Die erste Webseite erstellen 45

2.1 Webseiten bestehen aus rechteckigen Kästchen	45
2.2 HT-M-L: die »HyperText Markup Language«	46
2.2.1 HT wie »Hypertext«: Hyperlinks erstellen	47
2.2.2 M wie »Markup«: Etiketten kleben	47

2.2.3	L wie »Language«: Vokabeln und Grammatikregeln	47
2.2.4	Der Unterschied zwischen »HTML-Element« und »HTML-Tag«	47
2.3	Jede Webseite hat ein HTML-Grundgerüst	48
2.3.1	Die Datei »index.html« im Editor erstellen und speichern	49
2.3.2	Eine gute Angewohnheit: <!-- Kommentare -->	49
2.3.3	Das HTML-Grundgerüst für die Startseite erstellen	50
2.4	Der <!doctype> und das Stammelement <html>	51
2.4.1	Die Dokumenttyp-Definition <!doctype html>	52
2.4.2	Das Stammelement: <html> und </html> umschließen den Quelltext	52
2.5	HTML-Elemente können im Anfangs-Tag Attribute enthalten	52
2.6	<head> enthält wichtige Infos über die Webseite	53
2.6.1	Die Angabe des Zeichensatzes: <meta charset="utf-8">	54
2.6.2	Bitte nicht verkleinern: <meta name="viewport" ...>	54
2.6.3	Der Seitentitel steht zwischen <title> und </title>	56
2.6.4	Die Seitenbeschreibung mit <meta name="description">	57
2.7	<body> enthält den sichtbaren Bereich der Webseite	57
2.8	Der Kopfbereich <header> mit Überschrift und Slogan	59
2.9	Entwicklerwerkzeuge im Browser: HTML untersuchen	60
2.9.1	Die Entwicklerwerkzeuge in Firefox	61
2.9.2	Die Entwicklerwerkzeuge in Chrome	62
2.10	Auf einen Blick	63
3	CSS kennenlernen: Die erste Webseite gestalten	65
3.1	Jeder Browser hat ein fest eingebautes Stylesheet	65
3.2	Das HTML für <body> als schematische Darstellung	67
3.3	Das erste eigene Stylesheet: »style.css«	68
3.3.1	Schritt 1: Einen Unterordner und ein Stylesheet erstellen	68
3.3.2	Schritt 2: HTML-Datei und CSS-Datei verbinden mit <link>	69
3.4	Die erste eigene CSS-Regel: Hintergrund- und Schriftfarbe für <body>	70
3.4.1	Auch in CSS eine gute Angewohnheit: /* Kommentare */	70
3.4.2	Hintergrund- und Schriftfarbe für <body> ändern	70

3.5	Den Kopfbereich <header> selektieren und gestalten	72
3.5.1	Hintergrund- und Schriftfarbe für <header> ändern	72
3.5.2	Etwas Abstand zwischen Text und Rand einfügen mit »padding«	73
3.6	Entwicklerwerkzeuge: CSS im Browser untersuchen	74
3.7	Auf einen Blick	76
4	HTML-Elemente für Text: Überschriften, Absätze, Hervorhebungen und Listen	77
4.1	Überschriften strukturieren den Inhalt: <h1> bis <h6>	78
4.1.1	HTML kennt sechs Ebenen für Überschriften	78
4.1.2	Eine <h2>-Überschrift im Inhaltsbereich einfügen	79
4.2	Absätze und Hervorhebungen: <p>, , 	80
4.2.1	Absätze mit <p> und Hervorhebungen mit und 	80
4.2.2	Absätze und Hervorhebungen auf der Übungswebsite einfügen	80
4.2.3	HTML-Elemente verschachteln – zuerst geöffnet, zuletzt geschlossen	81
4.3	Webseiten von Anfang an in unterschiedlichen Viewports testen	82
4.4	Listen erstellen mit , und 	84
4.4.1	Ungeordnete Listen mit einem Aufzählungszeichen: und 	84
4.4.2	Geordnete Listen mit einer Nummerierung: und 	86
4.5	Listen verschachteln: Eine Liste in einer Liste	87
4.6	Über Blockelemente, Inline-Elemente und die Eigenschaft »display«	89
4.6.1	Blockelemente werden so breit, wie es geht	89
4.6.2	Inline-Elemente werden nur so breit wie ihr Inhalt	89
4.7	Auf einen Blick	90
5	Hyperlinks – das Besondere am Web	91
5.1	Das Standardverhalten von Hyperlinks	91
5.2	Anatomie eines Hyperlinks: Linktext	92
5.3	Hyperlinks und sinnvolle Linktexte: das »Klicken-Sie-hier«-Syndrom	94

5.4	Hyperlinks in neuem Tab oder Fenster öffnen	95
5.5	Eine Navigation ist eine Liste mit Links	96
5.6	Eine grundlegende Gestaltung für die Navigation	97
5.6.1	Schritt 1: Die Listenelemente nebeneinanderstellen	98
5.6.2	Schritt 2: Den Navigationsbereich und die Liste gestalten	100
5.6.3	Schritt 3: Die Links und den Linktext gestalten	101
5.7	Im Fußbereich einen Link »Nach oben« einfügen	102
5.7.1	Schritt 1: Das HTML für einen Link nach oben auf derselben Seite	102
5.7.2	Schritt 2: Eine grundlegende Gestaltung für den Footer und den Link	104
5.8	Besondere Links: Dateien, E-Mail und Telefon	106
5.8.1	Hyperlinks auf Dateien, die keine Webseiten sind: PDF & Co	106
5.8.2	Links auf E-Mail-Adressen	107
5.8.3	Links auf Telefonnummern	108
5.9	Auf einen Blick	108
6	HTML-Elemente für Bilder, Audio und Video	109
6.1	Über Grafikformate im Web: JPEG, GIF, PNG, SVG & Co	109
6.2	Ein Bild als Logo einbinden mit 	111
6.2.1	Das Element und seine wichtigsten Attribute	111
6.2.2	Ein Logo auf der Übungswebsite einfügen mit 	112
6.2.3	Fine-Tuning: Die Abstände um Logo und Slogan anpassen	113
6.3	Pixelbilder und hochauflösende Bildschirme	114
6.3.1	Das Problem: Das Logo ist auf hochauflösenden Bildschirmen unscharf	115
6.3.2	Die einfache Lösung: Eine doppelt so große Grafik einbinden	116
6.3.3	Die optimale Lösung: Je nach Pixeldichte unterschiedliche Dateien einbinden	116
6.3.4	Testen: Die korrekte Einbindung der Grafiken im Browser überprüfen	118
6.4	Bilder mit flexibler Breite: »max-width: 100%«	119
6.4.1	Das Problem: Pixelbilder haben eine feste Breite	120
6.4.2	Die Lösung: Flexible Bilder mit »max-width: 100%«	120

6.5	Abbildungen beschriften: <figure> und <figcaption>	122
6.5.1	Ein Foto mit einer Beschriftung: <figure> und <figcaption> im Einsatz	122
6.5.2	Die Einrückung von <figure> entfernen und die Beschriftung zentrieren	123
6.6	»Lazy Loading«: Seiten mit vielen Bildern optimieren	124
6.7	Let there be sound: Audiodateien einbinden mit <audio>	125
6.7.1	Audioformate, Browserunterstützung und Audioplayer	125
6.7.2	Die Einbindung von Sound-Dateien mit <audio>	126
6.7.3	Audiodateien beschriften mit <figure> und <figcaption>	127
6.8	Bewegte Bilder einbinden mit <video>	127
6.8.1	Videoformate und Browserunterstützung im Überblick	128
6.8.2	Die Einbindung von Videodateien mit <video>	128
6.8.3	Flexible Videos per CSS mit »max-width: 100%«	130
6.9	Auf einen Blick	131
7	HTML-Elemente zur Strukturierung von Webseiten und Inhalten	133
7.1	Die semantischen Strukturelemente auf einen Blick	134
7.2	Kopfbereiche auszeichnen mit <header>	134
7.2.1	Das Element <header> kann auf einer Seite mehrfach vorhanden sein.	134
7.2.2	Den Kopfbereich auf der Übungswebsite um eine Klasse erweitern	136
7.3	Navigationsbereiche erstellen mit <nav>	137
7.3.1	<nav> für die Site-Navigation auf der Übungswebsite	137
7.3.2	Den Navigationsbereich auf der Übungswebsite um eine Klasse erweitern	138
7.3.3	<nav> kann in der HTML-Struktur auch an anderen Positionen stehen	139
7.4	Der Hauptinhalt einer Webseite steht in <main>	140
7.4.1	Das Element <main> für den Hauptinhalt einer Webseite	140
7.4.2	Den Inhaltsbereich der Übungswebsite um eine Klasse erweitern	141

7.5	Fußbereiche auszeichnen mit <footer>	141
7.5.1	Der Fußbereich <footer> auf der Übungswebsite	141
7.5.2	Den Fußbereich auf der Übungswebsite um eine Klasse erweitern	142
7.6	Inhaltliche Abschnitte erstellen mit <section>	142
7.7	In sich geschlossene, eigenständige Blöcke mit <article>	144
7.7.1	Grundlegende Gestaltung für den Abschnitt und die Infoboxen	146
7.8	Bereiche mit zusätzlichen Informationen: <aside>	148
7.9	Elemente mit <div> semantisch neutral gruppieren	150
7.10	Auf einen Blick	152

8 Weitere HTML-Elemente zur Auszeichnung von Text 153

8.1	Zitate auszeichnen mit <blockquote> und <cite>	153
8.1.1	Das HTML für Blockzitate: <blockquote> und <cite>	154
8.1.2	Ein Blockzitat mit Quellenangabe	154
8.1.3	Eine grundlegende Gestaltung für ein Zitat mit Quellenangabe	156
8.2	Einen Zeilenumbruch erzwingen mit
	157
8.3	Kontaktinformationen auszeichnen mit <address>	157
8.3.1	Eine Kontaktadresse auszeichnen mit <address>	158
8.3.2	Eine grundlegende Gestaltung für eine Kontaktadresse	158
8.4	Zeitangaben für Menschen und Maschinen: <time>	159
8.4.1	Datumsangaben mit <time>	159
8.4.2	Die Uhrzeit mit <time>	160
8.5	Ausklappbare Inhalte: <details> und <summary>	161
8.5.1	Das HTML für ausklappbare Inhalte: <details> und <summary>	161
8.5.2	Eine grundlegende Gestaltung für <details> und <summary>	163
8.6	Änderungen am Text dokumentieren: und <ins>	164
8.6.1	Das HTML für Änderungen am Text	164
8.6.2	Eine grundlegende Gestaltung für Änderungen am Text	165
8.7	Kurz vorgestellt: , <hr> und <small>	166
8.7.1	 ist ein semantisch neutrales Inline-Element	166
8.7.2	<hr> markiert einen inhaltlichen Bruch innerhalb eines Abschnitts	167

8.7.3	Das sprichwörtliche Kleingedruckte mit <small>	167
8.8	Weitere Inline-Elemente in der Übersicht	167
8.9	Know-how: Zeichensätze und Sonderzeichen	169
8.9.1	UTF-8: Wissenswertes über Zeichensätze	169
8.9.2	Die Kodierung von Sonderzeichen in HTML	170
8.10	Auf einen Blick	172

9 HTML-Elemente zum Erstellen von Formularen 173

9.1	Interaktion mit Besuchern basiert auf HTML-Formularen	174
9.2	Das Element <form> definiert ein Formular	174
9.3	Einzeilige Eingabefelder erstellen und beschriften: <input> und <label>	176
9.3.1	Ein einzeiliges Eingabefeld für Text: <input type="text">	176
9.3.2	Die Beschriftung eines Formularfeldes mit <label>	176
9.3.3	Ein Eingabefeld für E-Mail-Adressen: <input type="email">	178
9.3.4	Pflichtfelder definieren: das Attribut »required«	179
9.4	Mehrzeilige Eingabefelder erstellen und beschriften: <textarea> und <label>	180
9.5	Zum Anklicken: eckige Kontrollkästchen, runde Optionsfelder und ausklappbare Auswahllisten	181
9.5.1	Eckige Kontrollkästchen mit <input type="checkbox">	181
9.5.2	Runde Optionsfelder mit <input type="radio">	182
9.5.3	Auswahllisten mit <select> und <option>	183
9.6	Formularfelder gruppieren mit <fieldset> und die Gruppe beschriften mit <legend>	185
9.7	Ein Button zum Abschicken der Formulardaten	186
9.8	Ein DSGVO-kompatibles Kontaktformular erstellen	187
9.8.1	Schritt 1: Das HTML für die Eingabefelder	188
9.8.2	Schritt 2: DSGVO-Einverständnis per Kontrollkästchen	189
9.8.3	Schritt 3: Eine grundlegende Gestaltung für das Formular	190
9.8.4	Schritt 4: Beschriftung und Formularfelder ausrichten	191
9.9	Auf einen Blick	193

10	HTML-Elemente zum Erstellen von Tabellen	195
10.1	Eine einfache HTML-Tabelle: <code><table></code> , <code><tr></code> und <code><td></code>	195
10.2	Tabellenüberschriften stehen in <code><th></code>	197
10.3	Tabellen strukturieren: <code><thead></code> , <code><tbody></code> und <code><tfoot></code>	198
10.4	Zellen verbinden mit » <code>colspan</code> « und » <code>rowspan</code> «	199
10.5	HTML-Tabellen erstellen und gestalten – ein Beispiel	200
10.5.1	Schritt 1: Das HTML für die Beispieltabelle	201
10.5.2	Schritt 2: Eine grundlegende Gestaltung für die Beispieltabelle	202
10.5.3	Schritt 3: Zwischenraum kontrollieren mit » <code>border-spacing</code> « und » <code>border-collapse</code> «	204
10.6	Auf einen Blick	204
11	Von der Webseite zur Website	205
11.1	Fine-Tuning für die Startseite	205
11.1.1	Eine Klasse für die Seite: <code><body class="startseite"></code>	206
11.1.2	»Sie sind hier«: Den aktuellen Menüpunkt hervorheben	206
11.1.3	Im Footer: Links zu Impressum und Datenschutz einfügen	208
11.2	Das HTML überprüfen mit dem HTML-Validator	210
11.3	Die Seiten »News«, »Über uns« und »Kontakt« erstellen	212
11.3.1	Die Seite »News« erstellen und anpassen	213
11.3.2	Die Seiten »Über uns« und »Kontakt« erstellen und anpassen	215
11.4	Inhalte für die Seite »News« hinzufügen	216
11.4.1	Einen neuen Abschnitt hinzufügen: <code><section class="beitragsliste"></code>	217
11.4.2	Einen Bereich mit Linklisten erstellen: <code><aside class="linklisten"></code>	218
11.4.3	Eine grundlegende Gestaltung für die Inhalte auf der Seite »News«	220
11.5	Ein Bild auf der Seite »Über uns« einfügen	220
11.6	Kontaktdaten und Formular für die Seite »Kontakt«	222
11.6.1	Den Abschnitt »Kontaktdaten« hinzufügen	223
11.6.2	Einen Abschnitt mit einem Kontaktformular hinzufügen	224

11.7	Die Seiten »Impressum« und »Datenschutz«	225
11.8	Auf einen Blick	226
12	CSS kennenlernen: Syntax, Box-Modell, Farbwerte und Einheiten	227
12.1	Überblick: Webseiten gestalten per CSS	227
12.2	Wichtige Vokabeln: Der Aufbau einer CSS-Regel	228
12.3	Es gibt drei verschiedene Möglichkeiten, CSS zu speichern	229
12.3.1	Externes Stylesheet: CSS-Regeln in einer eigenen Datei	229
12.3.2	»Style-Block«: CSS-Regeln mit <code><style></code> im <code><head></code> einer Webseite	230
12.3.3	»Inline-Styles«: Deklarationen mit dem Attribut » <code>style</code> « im Element	230
12.3.4	Die empfohlene Vorgehensweise: CSS so viel wie möglich in externen Dateien speichern	231
12.4	Das Box-Modell kennenlernen: » <code>padding</code> «, » <code>border</code> « und » <code>margin</code> «	231
12.4.1	Der Inhalt sollte möglichst flexibel bleiben	232
12.4.2	Der Innenabstand » <code>padding</code> « schafft Platz zwischen Inhalt und Rand	233
12.4.3	Die Rahmenlinien drumherum: » <code>border</code> «	233
12.4.4	Der Außenabstand » <code>margin</code> « regelt den Abstand zu anderen Boxen	234
12.4.5	Der Unterschied zwischen Abständen mit » <code>padding</code> « und » <code>margin</code> «	235
12.4.6	Das Box-Modell im Browser visualisieren	235
12.5	Farbnamen in CSS: Einfach, aber nicht sehr flexibel	236
12.6	Weit verbreitet: Hexadezimale Farbangaben mit <code>#rrggbb</code>	238
12.6.1	Der Aufbau eines hexadezimalen Farbwertes	238
12.6.2	Die hexadezimale Kurzschreibweise: <code>#rgb</code>	238
12.6.3	Übersicht: Einige Farbnamen und ihre HEX-Werte	239
12.6.4	HEXen und blaufärben: Farbnamen auf der Übungswebsite ändern	239
12.7	Die Alternative: Farben mit <code>rgb()</code> – auch mit Transparenz	240
12.8	Nützliche Werkzeuge und Websites zur Arbeit mit Farben	242
12.8.1	Firefox hat in den Entwicklerwerkzeugen einen Farbwähler	242
12.8.2	Ausführliche Farbauswahl in den Entwicklerwerkzeugen der Browser	243

12.9	Wichtige Einheiten: px, em, rem, % & Co	245
12.9.1	Die Einheit »px«: Ein Pixel ist nicht immer ein Pixel	245
12.9.2	Die Einheit »em« hat verschiedene Berechnungsgrundlagen	246
12.9.3	Die Einheit »rem« entspricht der Standardschriftgröße des Browsers	247
12.9.4	Die Einheit »%« (Prozent) ist meist relativ zum Elternelement	247
12.10	Auf einen Blick	248
13	Die wichtigsten Selektoren und Spezifität	249
13.1	Einfache Selektoren: Elemente, Gruppierung und »*«	250
13.1.1	»Der Name der Kiste« – einfache Elementselektoren	250
13.1.2	Mehrere Kästchen zugleich: Selektoren mit einem Komma gruppieren	250
13.1.3	Alle Kästchen auswählen: der Universalselektor »*«	251
13.2	Klassen sind klasse: Der Selektor mit dem Punkt	251
13.2.1	Beispiele für den Einsatz von Klassen auf der Übungswebsite	251
13.2.2	Gebundene Klassen: Klassen auf einen Elementtyp beschränken	252
13.2.3	Ein HTML-Element kann mehrere Klassennamen haben	253
13.3	IDs sind einmalig: Der Selektor mit der Raute	253
13.4	Überblick: Die HTML-Elemente im DOM-Baum	254
13.5	Nachfahren auswählen: Der Selektor mit dem Leerzeichen	256
13.6	Selektoren zum Auswählen von Kindelementen	256
13.6.1	Der Kindselektor: Der Selektor mit dem »>« (Größer-als-Zeichen)	256
13.6.2	Praktisch: Die Pseudoklassen »:first-child« und »:last-child«	257
13.6.3	Der Zauberstab zum Auswählen von Kindern: »:nth-child()«	259
13.7	Nachbar- und Geschwisterselektoren mit + und ~	260
13.8	Attributselektoren haben eckige Klammern: [attribut]	261
13.8.1	Der Selektor [attribut] prüft nur, ob es das Attribut gibt	262
13.8.2	Attributselektoren mit einem Gleichheitszeichen: [attribut="wert"]	262
13.8.3	Attributselektoren mit Tilde und Gleichheitszeichen: [attribut~="wert"]	263

13.8.4	Attributselektoren mit senkrechtem Strich und Gleichheitszeichen: [attribut = "wert"]	263
13.8.5	Attributselektoren mit Hütchen und Gleichheitszeichen: [attribut^="wert"]	264
13.8.6	Attributselektoren mit Dollar und Gleichheitszeichen: element[attribut\$="zeichen"]	264
13.8.7	Attributselektoren mit Sternchen und Gleichheitszeichen: element[attribut*="zeichen"]	265
13.9	Quellen zum Nachschlagen von weiteren Selektoren	265
13.10	Spezifität: Einige Selektoren sind wichtiger als andere	266
13.10.1	Einer wird gewinnen: So funktioniert Spezifität	266
13.10.2	Ganz sparsam benutzen: »!important« macht Anweisungen WICHTIG!	267
13.11	Auf einen Blick	268
14	Der Browser und das CSS: Kaskade, Vererbung oder Standardwert	269
14.1	Die Kaskade: Wichtigkeit, Spezifität und Reihenfolge	269
14.1.1	Die Kaskade hilft dem Browser, genau eine Anweisung zu finden	270
14.1.2	Die Ausgangssituation: Das Beispiel im Überblick	270
14.2	Intuitiv: Die Reihenfolge im CSS entscheidet	271
14.3	Ungewohnt: Spezifität ist wichtiger als Reihenfolge	272
14.4	Ausnahme: »!important« gewinnt immer	274
14.5	Nichts gefunden? Vererbung oder Standardwert	275
14.5.1	»Vererbung« macht ein Stylesheet übersichtlicher	275
14.5.2	Bestimmte Eigenschaften werden nicht vererbt	276
14.5.3	Falls er gar nichts findet, nimmt der Browser den »Standardwert«	277
14.6	Die Kaskade im Browser analysieren	277
14.7	Übersicht: Kaskade, Vererbung oder Standardwert	278
14.8	Auf einen Blick	280

15	Schrift und Text gestalten per CSS	281
15.1	Klassische Schriftarten mit und ohne Serifen im Web	281
15.1.1	Es gibt Schriftarten mit und ohne »Serifen«	282
15.1.2	Sehr praktisch: Die Schriftgestaltung in Firefox analysieren	282
15.2	Die Schriftart definieren mit »font-family«	283
15.2.1	Bitte eine Schriftart ohne Serifen mit »font-family«	283
15.2.2	Generische Schriftfamilien im Überblick	285
15.3	Die Systemschrift des Geräts: Gut lesbar und echt schnell	285
15.4	Webfonts – die Schriftart gleich mitliefern	287
15.5	Schnell und einfach: Google Fonts direkt von Google	288
15.5.1	Schritt 1: Schriftart und Schriftschnitte auswählen	289
15.5.2	Schritt 2: Den Code für die Schriftart kopieren und einfügen	290
15.6	Auf der sicheren Seite: Google Fonts selbst ausliefern	291
15.6.1	Schritt 1: Schriftart und Zeichensatz auswählen	292
15.6.2	Schritt 2: Die gewünschten Styles festlegen	292
15.6.3	Schritt 3: Den Code für die Schriftarten kopieren und einfügen	293
15.6.4	Schritt 4: Schriftdateien herunterladen und im Ordner »font« speichern	293
15.7	Gut lesbarer Text: »font-size« und »line-height«	295
15.7.1	Schriftgröße definieren mit »font-size« und einer Längeneinheit	295
15.7.2	Die Schriftgröße für die Überschriften ändern	296
15.7.3	Schriftgröße definieren mit »font-size« und einem Wort	297
15.7.4	Wichtig für die Lesbarkeit: Der Zeilenabstand mit »line-height«	298
15.8	Listen: Aufzählungszeichen gestalten per CSS	299
15.8.1	Die Gestaltung von Listen in den Browser-Stylesheets	300
15.8.2	Aufzählungszeichen gestalten mit »list-style« & Co	301
15.8.3	Aufzählungszeichen gestalten mit dem Pseudoelement »::marker«	302
15.9	Hyperlinks: Unterstreichungen gestalten mit »text-decoration«	303
15.9.1	Zusätzliche Eigenschaften zur Unterstreichung von Links	303
15.9.2	Die Unterstreichung der Links gestalten	304
15.10	Hyperlinks: Linkzustände gestalten mit Pseudoklassen	305
15.10.1	Besuchte und nicht besuchte Hyperlinks mit »:link« und »:visited«	305
15.10.2	Benutzeraktionen gestalten mit »:hover«, »:focus« und »:active«	307

15.11	Externe Hyperlinks kennzeichnen mit »::after«	309
15.11.1	Schritt 1: Externe Hyperlinks auswählen mit einem Attributselektor	309
15.11.2	Schritt 2: Das Pseudoelement »::after« und die Eigenschaft »content«	309
15.11.3	Schritt 3: Links kennzeichnen mit einem Unicode-Symbol	310
15.11.4	Alternative: Externe Links mit »target="_blank"« selektieren	311
15.12	Weitere Eigenschaften zur Gestaltung von Schrift und Text	312
15.12.1	Die wichtigsten Eigenschaften zur Schrift- und Textgestaltung	312
15.12.2	Schrift gestalten: fett, kursiv, Kapitälchen und Zeichenabstand	312
15.12.3	Die Kurzschreibweise »font«	313
15.12.4	Text ausrichten und die erste Zeile einrücken	313
15.12.5	Schatten im Text: »text-shadow«	314
15.13	Auf einen Blick	315
16	Abstände gestalten mit dem Box-Modell	317
16.1	Das klassische Box-Modell für Blockboxen	317
16.2	Die Breite begrenzen: »min-width« und »max-width«	318
16.3	Der Abstand zum Rand: »padding«	320
16.3.1	Das »padding« für den Kopfbereich der Seite	320
16.3.2	Das »padding« für die Navigation und den Fußbereich	321
16.3.3	Das »padding« für den Inhaltsbereich	322
16.4	Rahmenlinien gestalten mit »border« und »border-radius«	323
16.4.1	Die Eigenschaften zum Gestalten von Rahmenlinien	323
16.4.2	Abgerundete Ecken mit »border-radius«	326
16.5	Blockboxen horizontal zentrieren mit »margin: auto«	327
16.6	Abstände zwischen den Elementen mit »margin«	328
16.7	»Collapsing Margins«: Vertikale Außenabstände kollabieren	329
16.7.1	Praktisch: Vertikale Außenabstände aufeinanderfolgender Elemente kollabieren	329
16.7.2	Problematisch: Außenabstände kollabieren nicht nur bei aufeinanderfolgenden Elementen	331

16.7.3	Beispiel: Ein Kopfbereich mit Überschrift und »Collapsing Margins« ...	331
16.7.4	»padding-top« für den Kopfbereich verhindert das Kollabieren der Außenabstände	332
16.7.5	Nützlich: Eine CSS-Regel zur Vermeidung von »Collapsing Margins«	333
16.8	Das intuitivere Box-Modell: »box-sizing: border-box«	335
16.8.1	Das Border-Box-Modell in der Übersicht	335
16.8.2	Das Border-Box-Modell aktivieren mit »box-sizing: border-box«	336
16.9	Das Box-Modell für Inline-Boxen	337
16.10	Inline-Block: Blockboxen, aber nebeneinander	338
16.11	Auf einen Blick	339
17	Boxen gestalten per CSS	341
17.1	Hintergrundgrafiken per CSS einbinden und gestalten	341
17.1.1	Hintergrundgrafiken einbinden: »background-image«	342
17.1.2	Hintergrundgrafiken wiederholen: »background-repeat«	343
17.1.3	Hintergrundgrafiken positionieren: »background-position«	344
17.1.4	Hintergrundgrafiken fixieren: »background-attachment«	345
17.1.5	Die Größe der Hintergrundgrafik definieren: »background-size«	345
17.1.6	Die Kurzschreibweise: »background«	347
17.1.7	Das Box-Modell und die dritte Dimension	348
17.2	Lineare Farbverläufe: »background-image« und »linear-gradient()«	348
17.3	Schattenboxen mit »box-shadow«	350
17.4	Gestalten mit dem Box-Modell: Zitate als Kundenstimmen	352
17.4.1	Das HTML: »section« und »blockquote«	352
17.4.2	Zitate gestalten mit den Box-Modell-Eigenschaften	353
17.5	»Call to Action«: Hyperlinks in Buttons verwandeln	355
17.5.1	Die Ausgangssituation: Zwei ganz normale Hyperlinks	356
17.5.2	Schritt 1: Die grundlegende Gestaltung für beide Links	356
17.5.3	Schritt 2: Die Unterschiede – primäre und sekundäre Buttons	358
17.5.4	Schritt 3: Die Linkzustände der Buttons gestalten	358
17.5.5	Schritt 4: Einen sanften Übergang mit »transition« hinzufügen	359

17.6	Boxen am Bildschirm ausblenden: »visually-hidden«	360
17.6.1	Schritt 1: Die Klasse »visually-hidden« erstellen	361
17.6.2	Schritt 2: Den Elementen die Klasse »visually-hidden« zuweisen	362
17.7	Auf einen Blick	363
18	Ordnung halten: Stylesheets organisieren	365
18.1	Stylesheets strukturieren mit Kommentaren	366
18.1.1	Der Kommentar am Anfang des Stylesheets	366
18.1.2	Ein Stylesheet mit Kommentaren in Abschnitte unterteilen	367
18.2	Verschiedene Schreibweisen für CSS-Regeln	367
18.2.1	Übersichtlich und weit verbreitet: Auf jeder Zeile eine Deklaration	367
18.2.2	Kurze Regeln: Alles in einer Zeile	368
18.2.3	Übersichtlich: Mehrere Selektoren auf Zeilen verteilen	369
18.2.4	Reihenfolge der Deklarationen: 1. Am Box-Modell orientieren	369
18.2.5	Reihenfolge der Deklarationen: 2. Am Alphabet orientieren	370
18.3	CSS überprüfen mit dem CSS-Validator	371
18.4	Modulbauweise: Ein zentrales Stylesheet erleichtert die Entwicklung	372
18.4.1	Während der Entwicklung: Modulbauweise mit mehreren Stylesheets	372
18.4.2	Für die Live-Site: Alles wieder in einem Stylesheet vereinen	373
18.5	Die Stylesheets für die einzelnen Module erstellen	374
18.5.1	Schritt 1: Die einzelnen Stylesheets erstellen	374
18.5.2	Schritt 2: Stylesheets mit @import in »style.css« einbinden	375
18.6	Das Modul »basis.css« ist das Fundament	376
18.6.1	Der Abschnitt »Globale Einstellungen für die gesamte Website«	376
18.6.2	Der Abschnitt »Grundlegende Gestaltung von Schrift und Text«	377
18.6.3	Der Abschnitt »Nützliche, allgemeine Klassen«	379
18.7	Das Modul »layout.css« für Seitenlayout und Layoutbereiche	379
18.7.1	Der Abschnitt für das Seitenlayout	380
18.7.2	Die Abschnitte zur Gestaltung der Layoutbereiche	380
18.8	Das Modul »navi-inline.css« für die Navigation	381

18.9	Das Modul »content.css« zur Gestaltung der Inhalte	382
18.9.1	Links im Inhaltsbereich gestalten	382
18.9.2	Weitere Inhalte gestalten	382
18.10	Das Modul »forms.css« für Kontaktdaten und Formulare	383
18.11	Ein neues Modul für ein modernes Layout	384
18.11.1	Schritt 1: Das HTML anpassen – die Dopplung mit »div«	385
18.11.2	Schritt 2: Das Stylesheet »layout-modern.css« hinzufügen	387
18.11.3	Schritt 3: Fine-Tuning für die Infoboxen auf der Startseite	389
18.12	Auf einen Blick	391
19	Media Queries und responsives Webdesign	393
19.1	»Responsives Webdesign«: Das Web wird flexibel	393
19.2	Medientypen definieren das Ausgabemedium	394
19.2.1	Die Medientypen in der Übersicht	394
19.2.2	Eine Druckversion für die Übungswebsite mit »@media print«	395
19.3	Media Queries = Medientypen + Medieneigenschaften	397
19.3.1	Die Syntax: »@media Medientyp and (Eigenschaft: Wert)«	398
19.3.2	Eine Media Query zur Änderung der Hintergrund- und Schriftfarbe	398
19.3.3	Weitere Beispiele für Media Queries	400
19.3.4	Die wichtigsten Medieneigenschaften im Überblick	400
19.4	Media Queries brauchen den »Meta-Viewport«	401
19.4.1	Media Queries und die virtuelle Viewportbreite mobiler Browser	401
19.4.2	Der Meta-Viewport definiert die Viewportbreite für mobile Browser neu	402
19.5	Media Queries und die Suche nach dem Breakpoint	403
19.5.1	Weit verbreitet: Breakpoints für »Mobile«, »Tablet« und »Desktop«	403
19.5.2	Breakpoints sollte man vom Layout ableiten, nicht von Geräten	404
19.6	Auf einen Blick	404

20	Der Flow und die Eigenschaft »position«	405
20.1	Blockboxen, der Flow und »position: static«	405
20.1.1	Der »Block Formatting Context« mit Block- und Inline-Boxen	406
20.1.2	Mit »position: static« stehen Blockboxen immer untereinander	406
20.2	Versetzt weiterfließen mit »position: relative«	407
20.3	Raus aus dem Flow mit »position: absolute«	408
20.4	Der Trick: »absolute« und »relative« kombinieren	409
20.5	Wie ein Fels in der Brandung – »position: fixed«	412
20.6	Scrollen und dann stehen bleiben: »position: sticky«	413
20.7	Positionierte Boxen und der »z-index«	414
20.8	Auf einen Blick	417
21	Schweben und schweben lassen: »float«	419
21.1	Text um eine Abbildung fließen lassen mit »float«	419
21.1.1	Die Ausgangssituation: ein <figure> mit Bild und Beschriftung	419
21.1.2	Das <figure>-Element nach rechts floaten mit »float: right«	420
21.1.3	Gefloatete Boxen in einem schmalen Viewport überprüfen	422
21.1.4	Die umgebenden Elemente reichen bis unter die gefloatete Box	422
21.2	Floats beenden mit »clear: both«	423
21.3	Floats umschließen mit »display: flow-root«	424
21.3.1	Das Problem: Floats ragen nach unten aus dem Elternelement heraus	424
21.3.2	Die Lösung: Floats umschließen mit »display: flow-root«	424
21.4	Praktisch: Klassen zum Floaten und Clearen	425
21.5	Das Umschließen von Floats mit »@supports«	427
21.6	Auf einen Blick	428

22 Flexbox: Mehrspaltige Layouts mit »display: flex«	429
22.1 Flexbox und Grid – das moderne CSS-Layout	429
22.1.1 Der »Block Formatting Context« ist für Layouts nur bedingt geeignet	430
22.1.2 Flexbox und Grid: Jenseits vom »Block Formatting Context«	430
22.2 Flex-Container erstellen mit »display: flex«	431
22.2.1 Die Ausgangsposition – eine einfache Navigation	431
22.2.2 Eine Flexbox definieren mit »display: flex«	432
22.2.3 Layouten per Flexbox: Neue Möglichkeiten ohne alte Probleme	433
22.3 Die Fließrichtung von Flex-Items kontrollieren mit »flex-flow«	434
22.3.1 Jede Flexbox hat eine »Hauptachse« und eine »Querachse«	434
22.3.2 »flex-direction« ändert die Fließrichtung: von »row« zu »column«	435
22.3.3 »flex-wrap« ermöglicht einen Zeilenumbruch in der Flexbox	436
22.3.4 Shorthand: »flex-flow« ist kurz für »flex-direction« und »flex-wrap«	437
22.4 Flex-Items an der Hauptachse ausrichten mit »justify-content«	438
22.5 Flex-Items an der Querachse ausrichten mit »align-items« und »align-self«	439
22.5.1 Flex-Items an der Querachse ausrichten mit »align-items«	439
22.5.2 Einzelne Flex-Items an der Querachse ausrichten mit »align-self«	441
22.6 Automatische Abstände für Flex-Items mit »margin«	442
22.6.1 Flex-Items am Ende des Flex-Containers ausrichten mit »margin«	442
22.6.2 Elemente horizontal und vertikal zentrieren mit »margin: auto«	443
22.7 Flexibilität für Flex-Items: Die Zauberformel »flex: 1«	444
22.7.1 »Lieber Browser, bitte mach alle Flex-Items gleich groß.«	444
22.7.2 »flex« ist kurz für »flex-grow«, »flex-shrink« und »flex-basis«	445
22.7.3 Überraschung: »flex-grow« in einer mehrzeiligen Flexbox	446
22.8 Flexbox in Aktion: Den Footer am unteren Rand des Browserfensters platzieren	447
22.8.1 Schritt 1: <body> wird Flex-Container, die Layoutbereiche Flex-Items	447
22.8.2 Schritt 2: Die Zauberformel »flex: 1« für den Inhaltsbereich	448

22.9 Die Reihenfolge von Flex-Items ändern	450
22.10 Auf einen Blick	451
23 Eine responsive Navigation erstellen	453
23.1 Die responsive Navigation im Überblick	453
23.1.1 Die mobile Navigation in schmalen Viewports hat einen »Menü«-Button	453
23.1.2 Barrierefreiheit: Was passiert, wenn JavaScript nicht verfügbar ist?	454
23.1.3 In breiteren Viewports erscheint eine horizontale Navigation	455
23.1.4 »Progressive Enhancement«: Die Navigation schrittweise verbessern	456
23.1.5 Die Erstellung der responsiven Navigation im Überblick	456
23.2 Schritt 1: Prüfen, ob JavaScript im Browser aktiv ist	457
23.3 Schritt 2: Die mobile Navigation gestalten	459
23.3.1 Das Stylesheet »navi-responsiv.css« erstellen und einbinden	459
23.3.2 Mobile First – die grundlegende Gestaltung der Navigation für schmale Viewports	459
23.4 Schritt 3: Den Menübutton im HTML einfügen	461
23.5 Schritt 4: Den Menübutton per CSS gestalten	464
23.5.1 Kein JavaScript? Menübutton ausblenden und Navigation anzeigen	464
23.5.2 Menübutton per CSS gestalten	465
23.5.3 Burger-Menü: Mit »::before« vor dem Button eine Grafik einfügen	466
23.6 Schritt 5: Die Navigation per CSS ausblenden	468
23.7 Schritt 6: Die Navigation mit dem Menübutton einblenden	470
23.8 Schritt 7: Eine horizontale Navigation für breitere Viewports	471
23.9 Die Navigation im Fußbereich gestalten	473
23.10 Auf einen Blick	474

24	CSS-Grid: Mehrspaltige Layouts erstellen mit »display: grid«	477
24.1	Ein »Grid« ist ein Raster und schafft Ordnung	477
24.2	Mehrspaltiges Layout nur für moderne Browser: »@supports«	478
24.3	Das erste Grid-Layout: Drei Boxen nebeneinander	479
24.3.1	Ein Blick auf das HTML für den Abschnitt mit den Infoboxen	480
24.3.2	Schritt 1: Einen Grid-Container definieren mit »display: grid«	480
24.3.3	Schritt 2: Ein Grid-Layout erstellen mit »grid-template-columns« und der Einheit »fr«	482
24.3.4	Schritt 3: Den Zwischenraum kontrollieren mit »grid-gap«	483
24.4	Flexbox und Grid sind ein gutes Team	485
24.5	Grid-Items manuell platzieren mit nummerierten Grid-Linien	488
24.5.1	Ein Blick auf das HTML für den Abschnitt mit den Kundenstimmen	489
24.5.2	Einen Grid-Container definieren und das Grid-Layout erstellen	489
24.5.3	Lösung 1: Die HTML-Struktur für das Layout ändern	490
24.5.4	Lösung 2: Grid-Items manuell platzieren mit Grid-Linien und der Eigenschaft »grid-column«	492
24.6	Grid-Items manuell platzieren mit benannten Grid-Bereichen	493
24.6.1	Die HTML-Struktur für den Inhaltsbereich der Seite »News«	494
24.6.2	Grid-Container definieren und die benannten Grid-Bereiche erstellen	495
24.7	Die Grid-Zauberformel: Responsiv ohne Media Query	497
24.7.1	Die Ausgangsposition: HTML und CSS für die Teamvorstellung	498
24.7.2	Schritt 1: »repeat()« erzeugt mit »auto-fit« beliebig viele Spalten	499
24.7.3	Schritt 2: Die Funktion »minmax()« macht das responsive Grid perfekt	501
24.8	Die wichtigsten Grid-Eigenschaften in der Übersicht	502
24.9	Auf einen Blick	504

25	Flexible Icons und responsive Bilder	505
25.1	Flexible Icons: Skalierbare Symbole mit SVG	505
25.2	SVG-Icons mit als Datei einbinden	507
25.3	SVG-Icons kann man im Editor bearbeiten	509
25.4	SVG-Icons inline direkt im HTML einfügen	510
25.5	Unterschiedliche Bilder je nach Pixeldichte des Bildschirms	512
25.6	Unterschiedliche Bilder je nach Viewportbreite	513
25.6.1	Tausche X gegen W: , »srcset w« und »sizes«	514
25.6.2	Das Attribut »sizes« kann die Breite des Viewports abfragen	516
25.7	Auf einen Blick	518
	Index	519

Materialien zum Buch

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

► **alle Übungsdateien**

Gehen Sie auf www.rheinwerk-verlag.de/5560. 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.