

HANSER



Leseprobe

zu

„WordPress 5“

von Jolantha Belik

Print-ISBN: 978-3-446-43944-3
E-Book-ISBN: 978-3-446-44083-8
E-Pub-ISBN: 978-3-446-46704-0

Weitere Informationen und Bestellungen unter
<http://www.hanser-fachbuch.de/978-3-446-43944-3>

sowie im Buchhandel

© Carl Hanser Verlag, München

Inhalt

Vorwort	XVII
1 WordPress auf dem eigenen Server installieren	1
1.1 WordPress und seine zahlreichen Versionen	1
1.1.1 WP MU, WP MS, .com oder .org?	3
1.1.2 Welche Version ist die richtige für mich?	3
1.2 WordPress auf Server installieren	6
1.2.1 WordPress über Provider installieren lassen	7
1.2.2 WordPress per FTP hochladen und installieren	7
1.3 Mehrere WordPress-Installationen in einer Datenbank	13
1.4 Multi-Site: Mehrere Websites mit einer WordPress-Installation	13
1.4.1 Eine Multi-Site in WordPress aktivieren	13
1.4.2 Multi-Site deaktivieren und alle Websites löschen	17
2 WordPress lokal installieren	21
2.1 Testserver mit XAMPP Portable	21
2.1.1 Download von XAMPP Portable	22
2.1.2 XAMPP Portable installieren oder nur entpacken?	23
2.1.2.1 Einrichten von XAMPP Portable auf einem USB-Stick ...	24
2.1.3 Control Panel, Apache und MySQL starten	25
2.1.3.1 Port-Probleme auf Windows 10 lösen	27
2.1.4 Datenbank und Benutzer erstellen	28
2.2 WordPress im htdocs-Ordner	32
2.2.1 Fatal error: maximum execution time exceeded	38
2.2.1.1 Maximum execution time erhöhen	38
2.2.1.2 Maximum execution time exceeded während der Installation	39
2.2.2 Passwort vergessen?	40
2.2.3 Datenbank mit zwei WordPress-Installationen	41

3	Migration einer WordPress-Website/-Installation	43
3.1	Migration von WordPress.com zum eigenen Server	43
3.1.1	Daten von WordPress.com exportieren	44
3.1.2	Daten auf eigenem Server importieren	46
3.1.2.1	Fehlerbehebung: Leere Seite nach Import	49
3.1.3	Problemereiche nach dem Import	50
3.1.3.1	Formular fehlt	51
3.1.3.2	Videos werden nicht angezeigt	51
3.1.4	Probleme beheben	51
3.1.4.1	Formular und Videos	52
3.2	Migration von Server zu Server	53
3.2.1	Vorbereitungsarbeiten	53
3.2.2	Vom lokalen Test-Server zum eigenen Server	53
3.2.2.1	Nur eigenes Theme und eigenes Plugin übertragen	54
3.2.2.2	Gesamte Installation auf Server übertragen	58
3.2.3	Migration vom alten Server zum neuen Server	63
3.3	WordPress über Hauptdomain aufrufen	64
3.4	Von HTTP auf HTTPS umstellen	66
3.4.1	Von HTTP zu HTTPS	67
3.4.2	Das Mixed-Content-Problem	72
3.5	Mehr Sicherheit für Ihre Website	75
4	WordPress aktualisieren	77
4.1	Updates, Updates, Updates ...	77
4.1.1	Welche Aktualisierungen gibt es?	77
4.1.2	Wo werden Aktualisierungen angezeigt?	78
4.1.3	Wie werden Aktualisierungen durchgeführt?	82
4.1.3.1	Automatisch durchgeführte Aktualisierungen	82
4.1.3.2	One-Click-Updates via Dashboard	83
4.1.3.3	Manuelle Aktualisierungen via FTP	90
4.1.4	Fehlgeschlagene Aktualisierung – was nun?	91
5	Grundlegende Einstellungen	93
5.1	WordPress Dashboard	94
5.1.1	Ihr Profil anpassen	95
5.1.1.1	Bereich Persönliche Optionen	96
5.1.1.2	Bereich Name	97
5.1.1.3	Bereiche Kontaktinfo und Über Dich	98
5.1.1.4	Bereich Benutzerkonten-Verwaltung	99
5.1.2	Ihr Dashboard im Detail	100
5.1.2.1	Die Werkzeugleiste (A)	100
5.1.2.2	Die Menüleiste (B)	102
5.1.2.3	Die Dashboard-Boxen (C)	104
5.1.2.4	Dashboard-Box entfernen	108

5.2	Allgemeine Einstellungen	111
5.2.1	Einstellungen – Allgemein	111
5.2.2	Einstellungen – Schreiben	114
5.2.3	Einstellungen – Lesen	116
5.2.4	Einstellungen – Diskussion	117
5.2.5	Einstellungen – Medien	120
5.2.6	Einstellungen – Permalinks	121
5.3	Datenschutz gemäß DSGVO	123
5.3.1	Einstellungen – Datenschutz	123
5.3.2	Personenbezogene Daten	124
5.3.2.1	Personenbezogene Daten exportieren	124
5.3.2.2	Personenbezogene Daten löschen	129
5.4	Website-Zustand	131
6	Design anpassen	133
6.1	Theme mit dem Customizer anpassen	133
6.1.1	Arbeiten mit dem Customizer	134
6.1.1.1	Customizer öffnen, ausblenden und schließen	134
6.1.1.2	Änderungen veröffentlichen, speichern und planen	135
6.1.1.3	Untermenü und Customizer schließen	136
6.1.1.4	Ausgabegerät für Vorschau auswählen	136
6.2	Twenty Nineteen im Customizer anpassen	137
6.2.1	Im Customizer Theme wechseln	137
6.2.2	Website-Informationen im Customizer	138
6.2.2.1	Site-Logo	138
6.2.2.2	Titel und Untertitel	141
6.2.2.3	Website-Icon	141
6.2.3	Farben im Customizer	142
6.2.4	Menüs im Customizer	142
6.2.4.1	Neues Menü im Customizer erstellen	143
6.2.5	Widgets im Customer	149
6.2.6	Startseiten-Einstellungen im Customizer	151
6.2.7	Zusätzliches CSS im Customizer	151
6.3	Customizer anpassen	153
6.3.1	Einzelne Sections aus Customizer entfernen	153
6.3.2	Pinsel & Customizer entfernen	155
6.3.3	Customizer aus Menü Design entfernen	156
7	Sidebars & Widgets	159
7.1	Widgets-Bereiche (Sidebars)	159
7.1.1	Einen Widgets-Bereich definieren	160
7.1.2	Neuer Widgets-Bereich in Template-Datei	162
7.2	Widgets	164
7.2.1	Standard-Widgets in WordPress-Core	164

7.2.2	Widgets im Customizer	165
7.2.3	Widgets in der Widgets-Verwaltung	165
7.2.3.1	Die Widgets-Verwaltung im Standardmodus	165
7.2.3.2	Die Widgets-Verwaltung im Zugänglichkeitsmodus	167
7.2.4	WordPress Core – Standard-Widgets	169
7.2.4.1	Standard-Widget Archiv	169
7.2.4.2	Standard-Widget Audio	170
7.2.4.3	Standard-Widget Bild	170
7.2.4.4	Standard-Widget Galerie	170
7.2.4.5	Standard-Widget HTML	171
7.2.4.6	Standard-Widget Kalender	172
7.2.4.7	Standard-Widget Kategorien	172
7.2.4.8	Standard-Widget Meta	172
7.2.4.9	Standard-Widget Navigationsmenü	173
7.2.4.10	Standard-Widget Neueste Beiträge	173
7.2.4.11	Standard-Widget Neueste Kommentare	174
7.2.4.12	Standard-Widget RSS	174
7.2.4.13	Standard-Widget Schlagwörter-Wolke	174
7.2.4.14	Standard-Widget Seiten	175
7.2.4.15	Standard-Widget Suche	175
7.2.4.16	Standard-Widget Text	176
7.2.4.17	Standard-Widget Video	176
7.2.5	Altes Standard-Widget Links	177
8	Der Gutenberg Block-Editor	179
8.1	Die Block-Editor-Oberfläche	180
8.1.1	Block-Editor: Werkzeugleiste (A)	181
8.1.1.1	Block-Editor: Tastaturkürzel	184
8.1.1.2	Block-Editor: kompletten Inhalt kopieren	185
8.1.1.3	Block-Editor: Ansicht anpassen	186
8.1.1.4	Block-Editor: Block-Manager	187
8.1.2	Block-Editor: Einstellungen-Sidebar (B)	188
8.1.2.1	Block-Editor: Dokument – Status und Sichtbarkeit	189
8.1.2.2	Block-Editor: Permalinks	189
8.1.2.3	Block-Editor: Dokument – Beitragsbild	191
8.1.2.4	Block-Editor: Dokument – Diskussion	193
8.1.2.5	Block-Editor: Dokument – Seitenattribute	193
8.1.2.6	Block-Editor: Dokument – Revisionen	193
8.1.2.7	Block-Editor: Dokument – Kategorien	196
8.1.2.8	Block-Editor: Dokument – Schlagwörter	196
8.1.2.9	Block-Editor: Dokument – Textauszug	197
8.1.3	Block-Editor: Inhaltsteil (D)	197
8.1.3.1	Block-Editor: Titel (C)	197

8.2	Block-Editor: Arbeiten mit Blöcken	198
8.2.1	Block-Editor: Neuen Block hinzufügen	198
8.2.2	Block-Editor: Blöcke verschieben	198
8.2.3	Block-Editor: Block-Typ ändern	199
8.2.3.1	Block-Editor: Block-Typ Überschrift umwandeln	199
8.2.3.2	Block-Editor: weitere Block-Typen umwandeln	200
8.2.4	Block-Editor: Block löschen	201
8.2.5	Block-Editor: Wiederverwendbare Blöcke	201
8.2.5.1	Block-Editor: Wiederverwendbaren Block erstellen	202
8.2.5.2	Block-Editor: wiederverwendbaren Block bearbeiten	203
8.2.5.3	Block-Editor: wiederverwendbaren Block in „normalen“ Block umwandeln	203
8.2.5.4	Block-Editor: wiederverwendbare Blöcke verwalten	204
8.3	Arbeiten im Visuellen Editor	207
8.3.1	Block-Editor: Text einfügen und formatieren	207
8.3.1.1	Block-Editor: Text einfügen	207
8.3.1.2	Block-Editor: Sonderzeichen einfügen	208
8.3.1.3	Block-Editor: Einfache Text-Formatierungen	208
8.3.1.4	Block-Editor: Schriftgrößen	209
8.3.1.5	Block-Editor: Schriftfarbe und Hintergrund ändern	211
8.3.1.6	Block-Editor: Initialbuchstabe	212
8.3.2	Block-Editor: Absatzausrichtung ändern	213
8.3.2.1	Block-Editor: Blocksatz definieren	213
8.3.3	Block-Editor: Hyperlink einfügen	214
8.3.3.1	Block-Editor: internen Link einfügen	214
8.3.3.2	Block-Editor: Externen Link einfügen	214
8.3.3.3	Block-Editor: Hyperlink bearbeiten oder entfernen	215
8.3.4	Block-Editor: Anker-Links einfügen	215
8.3.4.1	Anker im Block-Editor setzen	215
8.3.4.2	Anker-Text im Block-Editor verlinken	216
8.3.4.3	Link „Nach oben“ im Block-Editor setzen	216
8.4	Blöcke: Bereich Allgemeine Blöcke	217
8.4.1	Allgemeine Blöcke: Absatz-Block	217
8.4.2	Allgemeine Blöcke: Audio-Block	218
8.4.2.1	Audio-Datei mit Audio-Block einfügen	218
8.4.2.2	Audio-Datei mit Classic-Block einfügen	220
8.4.2.3	Audio-Playlist mit Classic-Block einfügen	222
8.4.3	Allgemeine Blöcke: Bild-Block	226
8.4.3.1	Bild-Datei in Bild-Block einfügen	227
8.4.3.2	Bild im Bild-Block bearbeiten	232
8.4.3.3	Exkurs: Bilder online bearbeiten	235
8.4.4	Allgemeine Blöcke: Cover-Block	242
8.4.5	Allgemeine Blöcke: Datei-Block	244
8.4.6	Allgemeine Blöcke: Galerie-Block	246

8.4.7	Allgemeine Blöcke: Liste-Block	249
8.4.8	Allgemeine Blöcke: Überschrift-Block	251
8.4.9	Allgemeine Blöcke: Video-Block	251
8.4.9.1	Video mit Video-Block einfügen	252
8.4.9.2	Video mit Classic Block einfügen	254
8.4.9.3	Exkurs: Videountertitel erstellen (VTT-Datei)	256
8.4.10	Allgemeine Blöcke: Zitat-Block	257
8.5	Blöcke: Bereich Formatierung	258
8.5.1	Bereich Formatierung: Classic Block	258
8.5.2	Bereich Formatierung: Code-Block	259
8.5.3	Bereich Formatierung: HTML-Block	259
8.5.4	Bereich Formatierung: Pullquote-Block	260
8.5.5	Bereich Formatierung: Tabelle-Block	261
8.5.6	Bereich Formatierung: Vers-Block	264
8.5.7	Bereich Formatierung: Vorformatiert-Block	265
8.6	Blöcke: Bereich Layout-Elemente	266
8.6.1	Bereich Layout-Elemente: Abstandhalter-Block	266
8.6.2	Bereich Layout-Elemente: Button-Block	267
8.6.3	Bereich Layout-Elemente: Medien-und-Text-Block	267
8.6.4	Bereich Layout-Elemente: Mehr-Block	268
8.6.5	Bereich Layout-Elemente: Seitenumbruch-Block	269
8.6.6	Bereich Layout-Elemente: Spalten-Block	270
8.6.7	Bereich Layout-Elemente: Trennzeichen-Block	271
8.7	Blöcke: Bereich Widgets	271
8.7.1	Bereich Widgets: Archive-Block	272
8.7.2	Bereich Widgets: Kalender-Block	272
8.7.3	Bereich Widgets: Kategorien-Block	273
8.7.4	Bereich Widgets: Neue Beiträge-Block	273
8.7.5	Bereich Widgets: Neue Kommentare-Block	274
8.7.6	Bereich Widgets: RSS-Block	274
8.7.7	Bereich Widgets: Schlagwörter-Wolke-Block	275
8.7.8	Bereich Widgets: Shortcodes-Block	275
8.7.9	Bereich Widgets: Suchen-Block	276
8.8	Blöcke: Bereich Einbettungen	276
8.8.1	Bereich Einbettungen: Facebook-Block	277
8.8.1.1	Facebook-Video einbetten	277
8.8.1.2	Facebook-Beitrag einbetten	278
8.8.2	Bereich Einbettungen: YouTube-Block	279
8.8.2.1	YouTube-Video mit YouTube-Block einbetten	279
8.8.2.2	YouTube-Video im erweiterten Datenschutzmodus im HTML-Block einbetten	280
8.8.3	Bereich Einbettungen: Twitter-Block	282
8.8.4	Bereich Einbettungen: Crowdsignal-Block	284
8.8.5	Bereich Einbettungen: Amazon-Kindle-Block	285

8.9	Blöcke im Block-Editor erweitern	286
8.9.1	Theme-spezifische Bildgröße definieren	286
8.9.2	Mit Editor-Styles Rahmen um Bild anzeigen	289
8.9.2.1	CSS-Datei für Editor Styles	289
8.9.2.2	Theme-Support aktivieren und CSS-Datei einbinden	290
8.9.3	Site-spezifische Block-Stile	291
8.9.3.1	Plugin für Block-Stile erstellen	292
	Weitere Beispiele für eigene Block-Stile im Block-Editor	296
9	Weiterarbeiten mit dem Classic Editor TinyMCE	301
9.1	Arbeiten mit Classic Editor ohne Plugin	301
9.1.1	Der Classic Block in Gutenberg	302
9.1.2	Classic Block in Blöcke umwandeln	303
9.1.2.1	Überschriften und Absätze in Blöcke umwandeln	303
9.1.2.2	iFrame in Block umwandeln	304
9.1.2.3	Tabelle in Block umwandeln	304
9.1.2.4	Bild mit Bildunterschrift in Block umwandeln	304
9.1.2.5	Fehlermeldung: Unerwarteter oder ungültiger Inhalt	305
9.2	Das Classic Editor Plugin	306
9.2.1	Classic Editor Plugin installieren und aktivieren	306
9.2.2	Standard-Editor festlegen	306
9.2.2.1	Standard-Editor für alle Benutzer festlegen	307
9.2.2.2	Standard-Editor individuell festlegen	307
9.2.3	Zwischen Classic Editor und Gutenberg wechseln	308
9.2.3.1	Editor-Wechsel bei bestehenden Seiten/Beiträgen	308
9.2.3.2	Editorauswahl bei neuen Seiten/Beiträgen	308
9.2.3.3	In Blöcke umgewandelte Classic-Block-Inhalte mit Classic Editor bearbeiten	309
10	Seiten erstellen und verwalten	311
10.1	Die Seiten-Verwaltung	311
10.1.1	Bestehende Seiten bearbeiten	312
10.1.1.1	Bestehende Seiten in Seiten-Verwaltung bearbeiten	312
10.1.1.2	Bestehende Seiten vom Frontend aus bearbeiten	313
10.2	Seite erstellen und bearbeiten	314
10.2.1	Seiten Home, Aktuelles und Kontakt	316
10.2.1.1	Leere Seite Home (Variante B1)	317
10.2.1.2	Seite Home mit Inhalt (Variante B2 und B3)	318
10.2.1.3	Seite Aktuelles (ohne Inhalt)	318
10.2.1.4	Seite Kontakt	318
10.3	Inhalte auf mehrere Seiten aufteilen	318
10.3.1	Automatisch erstellte Paginierung (im Frontend)	319
10.3.2	Benutzerdefinierte Paginierung	320
10.3.2.1	Benutzerdefinierte Paginierung („Kurzversion“)	320
10.3.2.2	Benutzerdefinierte Paginierung („Langversion“)	321

10.4	Meta Description als Eigenes Feld	321
10.4.1	Benutzerdefiniertes Feld erstellen	321
10.4.1.1	Neues Benutzerdefiniertes Feld	321
10.4.1.2	Bestehendes Benutzerdefiniertes Feld	322
10.4.1.3	Weiteres neues Benutzerdefiniertes Feld	323
10.4.2	Custom Field als HTML-Meta-Tag definieren	324
10.4.3	Benutzerdefinierte Felder innerhalb des Loops	325
11	Menüs	327
11.1	Die Menü-Verwaltung	327
11.1.1	Neues Menü erstellen	328
11.1.2	Bestehendes Menü bearbeiten	332
11.1.3	Menüs und Menüpositionen	332
11.1.4	Menüpunkt entfernen und Menü löschen	333
11.2	Menüs in Widgets	333
11.3	Menüposition im Theme definieren	334
12	Medienverwaltung	337
12.1	Mediathek	337
12.1.1	Die Mediathek in der Kachel-Ansicht	338
12.1.1	Medien in der Kachel-Ansicht bearbeiten	339
12.1.2	Die Mediathek in der Listen-Ansicht	340
12.2	Dateien verwalten	342
12.2.1	Dateien hochladen	342
12.2.2	Dateien löschen	343
13	Kategorien und Schlagwörter	345
13.1	Kategorien vs. Tags	345
13.2	Kategorien	346
13.2.1	Neue Standardkategorie erstellen	346
13.2.2	Bestehende Kategorie bearbeiten	348
13.2.3	Kategorie löschen	349
13.2.4	Neue Kategorie in der Box Kategorie erstellen	350
13.2.5	Neue Unterkategorie in der Box erstellen	351
13.2.6	Kategorien in Widget anzeigen	352
13.2.7	Kategorie-Wolke anzeigen	353
13.2.8	Kategorie-Wolke in Widget	353
13.2.9	Beiträge aus bestimmter Kategorie anzeigen	354
13.3	Tags oder Schlagwörter?	356
13.3.1	Schlagwörter hinzufügen bzw. entfernen	356
13.3.2	Schlagwörter verwalten	357
13.3.3	Schlagwörter-Wolke Widget	358
13.4	Kategorie- und Schlagwort-Konverter	359
13.4.1	Schlagwörter in Kategorien umwandeln	359
13.4.2	Kategorien in Schlagwörter umwandeln	360

14	Beiträge und Kommentare	361
14.1	Beiträge erstellen und verwalten	361
14.1.1	Neuen Beitrag erstellen	361
14.1.1.1	Bedienfeld Status und Sichtbarkeit	362
14.1.1.2	Bedienfeld Permalink	367
14.1.1.3	Bedienfeld Revisionen	367
14.1.1.4	Bedienfeld Kategorien und Schlagwörter	367
14.1.1.5	Bedienfeld Beitragsbild (Featured Image)	367
14.1.1.6	Bedienfeld Textauszug	368
14.1.1.7	Bedienfeld Diskussion	369
14.1.1.8	Bedienfeld Beitrags-Attribute	370
14.2	Beiträge verwalten	370
14.2.1	Beiträge bearbeiten	372
14.2.1.1	Beitrag im Editor bearbeiten	373
14.2.1.2	Beitrag im QuickEdit-Fenster bearbeiten	373
14.2.2	Weiterlesen-Link	374
14.2.2.1	Weiterlesen-Link einfügen	374
14.2.2.2	Individuellen Weiterlesen-Link erstellen	374
14.3	Kommentare verwalten	375
15	Benutzerverwaltung	377
15.1	Rollen und Rechte	377
15.1.1	Abonnent (Subscriber)	381
15.1.2	Mitarbeiter (Contributor)	382
15.1.2.1	Mitarbeiter erstellt neuen Beitrag	383
15.1.2.2	Revision und Veröffentlichung	384
15.1.3	Autor (Author)	385
15.1.4	Redakteur (Editor)	386
15.1.5	Administrator	387
15.2	Benutzer verwalten	388
15.2.1	Neuen Benutzer hinzufügen	389
15.2.2	Benutzerdaten und Rolle ändern	390
15.2.3	Benutzer löschen	391
15.3	Rollen einschränken	392
15.3.1	Menüeintrag ausblenden	392
15.3.1.1	Menüeintrag Jetpack ausblenden	392
15.3.1.2	Menüeintrag Formular (Contact Form 7) ausblenden	393
15.4	Rollen erweitern	394
15.4.1	Das Hochladen von Bildern gestatten	394
15.5	Eigene Info-Box auf dem Dashboard	395
16	Plugins	397
16.1	Die Plugins-Verwaltung	397
16.1.1	Plugins installieren und aktivieren	399

16.1.2	Plugins deaktivieren und löschen	400
16.1.3	Der Plugin-Editor	401
16.2	Ausgewählte Plugins	402
16.2.1	Plugin Akismet	402
16.2.1.1	Akismet DSGVO-Hinweis für Kommentarformulare	403
16.2.1.2	Akismet-Widget	404
16.2.2	Plugin Captcha Code	404
16.2.3	Plugin Disable REST API	405
16.2.4	Plugin Disable XML-RPC Pingback	405
16.2.5	Plugins für Google Maps	406
16.2.6	Plugin Google XML Sitemaps	406
16.2.7	Jetpack	406
16.2.7.1	Jetpack-Module	407
16.2.7.2	Jetpack-Blöcke für den Block-Editor	411
16.2.7.3	Jetpack-Einstellungen im Block-Editor	423
16.2.7.4	Jetpack-Widgets und zusätzliche Funktionen	424
16.2.8	Plugin miniOrange 2 Factor Authentication	433
16.2.9	Plugin WP Updates Notifier	434
16.2.10	Mehrsprachige Internetauftritte	434
16.2.10.1	Fallbeispiel: Kunjana.net	434
16.2.10.2	Anforderungen an ein Multilingual-Plugin	437
16.2.10.3	Ausgewählte Plugins	437
17	Site-spezifisches Plugin	445
17.1	Erste Schritte zum Custom Plugin	445
17.1.1	Ordnerstruktur des Custom Plugins	446
17.1.2	Der Header der Plugin-Datei	446
17.1.2.1	Plugin-Datei erstellen	447
17.1.2.2	Custom Plugin in WordPress installieren und aktivieren	448
17.1.2.3	Fataler Fehler im Code, was nun?	449
17.2	Funktionen für Ihr Custom Plugin	450
17.2.1	Dashboard Widgets entfernen	450
17.2.2	Eigenes Logo für die Login-Seite	453
17.2.2.1	Verlinkung auf dem Logo ändern	453
17.2.2.2	Tooltip bei Logo-Link vs. headertext	453
17.2.2.3	Eigenes Logo anzeigen	455
18	Bestehendes Theme in responsives Theme umwandeln	457
18.1	Voraussetzungen & Vorbereitungen	457
18.1.1	Voraussetzungen für ein responsives Layout	458
18.1.2	Erforderliche Änderungen	458
18.1.2.1	Problem mit Menü-Positionen in alten Themes	459
18.1.2.2	Mögliche Probleme mit der Datei header.php	459
18.1.3	Vorbereitung und Anpassungen	461
18.1.3.1	Vorbereitungsarbeiten	461
18.1.3.2	Das „alte“ Theme wird responsive	461

19	Arbeiten mit Child-Themes	479
19.1	Child-Theme einrichten	479
19.1.1	Die style.css im Child-Theme	480
19.1.2	Die functions.php im Child-Theme	481
19.1.3	Das Child-Theme aktivieren	482
19.2	Änderungen im Child-Theme	484
19.2.1	Neue Footer-Zeile	484
19.2.1.1	Neue ©-Zeile im Footer	484
19.2.1.2	Anmelde-Link im Footer	487
19.2.1.3	RSS-Link im Footer	488
19.2.2	Site-Logo größer anzeigen	489
19.2.3	Textformatierungen anpassen	491
19.2.3.1	Größe Widget-Titel anpassen	491
19.2.3.2	Textgröße Widgets	491
19.2.3.3	Titelgröße verändern	492
20	Eigenes Theme erstellen	493
20.1	Das Beispiel-Theme	493
20.1.1	Aufbau des Layouts	494
20.1.2	Projekt-Ordner, layout.html und style.css	495
20.1.2.1	Beispiel-Layout: layout.html	495
20.1.2.2	Beispiel-Layout: style.css	500
20.1.3	Responsives Layout	528
20.1.3.1	Viewport	528
20.1.3.2	Media Queries	528
20.1.4	WordPress-spezifische CSS-Regeln	532
20.2	Aufbau von WordPress-Themes	539
20.2.1	Template-Dateien erstellen	540
20.2.2	Die Template-Dateien bearbeiten	542
20.2.2.1	Die Datei style.css	543
20.2.2.2	Die Datei functions.php	545
20.2.2.3	Die Datei header.php	560
20.2.2.4	Die Datei footer.php	564
20.2.2.5	Die Datei index.php	566
20.2.2.6	Die Template-Parts-Dateien	568
20.2.2.7	Die Datei page.php	572
20.2.2.8	Die Datei single.php	573
20.2.2.9	Die Datei 404.php	573
20.2.2.10	Die Template-Datei für das One-Page-Layout	574
20.2.2.11	Beispiel-Theme installieren und aktivieren	576
	Index	579

Vorwort

WordPress hat mit Version 5.0 nicht nur einen neuen Editor bekommen, auch das Gesamtpaket wurde weiterentwickelt. In diesem Buch wird WordPress ab der Version 5.2 vorgestellt, von der Installation und Migration auf dem eigenen Server bis hin zum Anpassen eines Themes im Customizer und Arbeiten mit Child-Themes, Erstellen eines eigenen Themes from scratch auf XAMPP als Testserver bis hin zum Erstellen eines Site-spezifischen Plugins sowie Arbeiten mit WordPress als Anwenderin und Anwender. Es werden auch zahlreiche Anpassungsmöglichkeiten aus der Praxis gezeigt, die ohne tiefere PHP- und JavaScript-Kenntnisse durchführbar sind. Damit ist das Buch nicht nur für WordPress-Neulinge und Umsteiger von WordPress.com auf den eigenen Server geeignet, sondern auch für Hobby-Programmiererinnen und Programmierer, die sich in die Welt der Template-Tags, Loops und Widgets wagen. Das Buch richtet sich aber auch an Profis, die dem Block-Editor bisher eher ablehnend gegenübergestanden sind. Ich war anfangs vom Block-Editor auch nicht begeistert, inzwischen sind mir die aktuellen Grenzen bewusst, dennoch bin ich vom Block-Editor sehr angetan. Wer weiterhin lieber den Classic Editor (TinyMCE) weiterverwenden möchte, findet eine detaillierte Beschreibung im Buch, wie dies möglich gemacht sowie ein Wechsel bei Bedarf zwischen den beiden Editoren erlaubt und durchgeführt werden kann.

Für alle anderen widmet sich ein ausführliches Kapitel dem neuen Block-Editor – vom Kennenlernen der Blöcke bis zum Erweitern bzw. Einschränken von Anpassungsoptionen für Userinnen und User, Definieren von Theme-spezifischen Farben und Schriftgrößen sowie eigenen Block-Stilen. Als ich mit dem Gutenberg-Kapitel begann, war ursprünglich geplant, die einzelnen Schritte parallel zum Kapitel mit dem Classic Editor (TinyMCE) durchzuarbeiten. Bereits nach kurzer Zeit wurde aber klar, dass dies nicht sinnvoll war. Der Block-Editor funktioniert nicht nur anders, es entsteht ein vollkommen anderer Workflow. Sobald man sich mit der Oberfläche auskennt und weiß, wie man neue Blöcke hinzufügen kann, ist – meiner Erfahrung mit zahlreichen WordPress-Grundkurs-Teilnehmerinnen und -Teilnehmern nach – das Arbeiten mit dem Block-Editor intuitiver, einfacher und schneller.

Soweit möglich, werden bei den Beispielen im Buch keine Plugins, sondern On-Board-Mittel verwendet, um den Funktionsumfang von WordPress Core zu erweitern. Ich vertrete die Ansicht, je weniger Plugins, umso sicherer und schneller ist WordPress (die Anfang 2020 entdeckten Sicherheitslücken in einigen großen beliebten Plugins bestärken mich in dieser Meinung). Mit anderen Worten, auf gut Wienerisch gesagt, man braucht nicht für jeden i-Punkt ein extra Plugin. Für manche Bereiche allerdings, wie beispielsweise Mehrsprachig-

keit, ist der Einsatz eines Plugins wohl unumgänglich, wobei gerade das Problem einer mehrsprachigen Website nach wie vor nicht zur vollen Zufriedenheit gelöst ist. Mein Praxisbeispiel *Kunjana* zeigt die Problematik auf, wenn ein Plugin nicht mehr weiterentwickelt und ein Umstieg auf ein anderes kaum möglich bis unmöglich wird. Wer einen mehrsprachigen Internetauftritt realisieren möchte, sollte sich genauestens überlegen, welches Plugin dafür verwendet werden soll. Die Übersicht und Beschreibung der aktuellen Free-, Free-mium- und Premium-Multi-Language-Plugins in diesem Buch sind dabei sicherlich eine hilfreiche Entscheidungsgrundlage.

Eine häufig unbeachtete Funktion betrifft die Benutzerverwaltung. WordPress Core verfügt über fünf Benutzerrollen, die registrierten Benutzern unterschiedliche Rechte zuweisen. In diesem Buch werden nicht nur die einzelnen Rollen und deren Rechte detailliert beleuchtet. Sie erfahren auch, wie Sie im Bedarfsfall den Rollen neue Rechte hinzufügen und auch einschränken können.

Abschließend möchte ich mich bei allen bedanken, die mit Ideen und Fragen aus der Praxis zur inhaltlichen Gestaltung beigetragen und mich immer wieder motiviert haben. Danke an die Firma *Techsmith* für das Zurverfügungstellen von *Snagit* und *Camtasia* zum Erstellen und Bearbeiten der Screenshots und an *Automattic* für *Jetpack* Premium. Schlussendlich geht ein spezieller herzlicher Dank an meine Betreuerin beim Hanser Verlag für ihre schier unendliche Geduld.

Die Website zum Buch unter

<https://www.wp5buch.net>

verwendet das nach dem Mobile-First-Ansatz entwickelte Beispiel-Theme aus dem letzten Kapitel. Die im Buch erstellten JSON-Dateien (wiederverwendbare Blöcke) sowie das Site-spezifische Plugin und die Dateien des Beispiel-Themes können Sie kostenlos herunterladen (Passwort: jwp5buch).

In zahlreichen Kapiteln finden Sie detaillierte Schritt-für-Schritt-Beschreibungen mit Screenshots passend zu den einzelnen Schritten. Ich wünsche viel Spaß beim Nacharbeiten dieser Beispiele sowie viel Erfolg für Ihr WordPress-Projekt.

Wien, im Juli 2020

Jolantha „Jola“ Belik

19

Arbeiten mit Child-Themes



In diesem Kapitel erfahren Sie ...

- ... wie Sie ein Child-Theme erstellen,
- ... wie Sie ein Child-Theme aktivieren,
- ... wie Sie das Standard-Theme Twenty Nineteen mit einem Child-Theme anpassen.

Manchmal möchte man an einem Theme weitere Anpassungen vornehmen, als jene mittels Customizer. Wenn Sie Änderungen in Original-Theme-Dateien durchführen, werden diese beim nächsten Theme-Update überschrieben. Um dies zu verhindern, werden Änderungen in sogenannte *Child-Themes* geschrieben. Wenn ein Child-Theme vorhanden ist, schaut WordPress zuerst in der Child-Theme-Datei nach und überschreibt mit den dort gefundenen Angaben die Original-Datei. Anders bei der `functions.php` im Child Theme. Diese Datei wird unmittelbar vor dem Parent-Theme `functions.php` geladen, die Funktionen etc. werden dem Parent-Theme `functions.php` hinzugefügt.

Für Änderungen in einzelnen Template-Dateien kopiert man die jeweilige Datei in den Child-Theme-Ordner und führt dort die Änderung durch. Bei einem Update bleiben die Child-Theme-Dateien unberührt. Dies hat allerdings auch den Nachteil, dass Korrekturen von etwa Fehlern oder Sicherheitslücken mit dem nächsten Update im Parent-Theme korrigiert und behoben werden können, nicht jedoch automatisch auch im Child-Theme.

■ 19.1 Child-Theme einrichten

Für ein *Child-Theme* benötigen Sie im Ordner `wp-content/themes` einen neuen Ordner mit dem Namen des Original-Themes gefolgt von `-child`. Achten Sie darauf, dass sich keine Leerzeichen im Namen befinden, sonst erhalten Sie Fehlermeldungen. Im Beispiel sollen Änderungen an *Twenty Nineteen* vorgenommen werden, somit heißt der Child-Ordner `twentyineteen-child`. In diesem Ordner benötigen Sie zwei neue leere Dateien, `functions.php` und `style.css` (siehe Bild 19.1).

Anmerkung: Sie können den Child-Theme-Ordner theoretisch beliebig benennen, solange Sie nur Kleinbuchstaben und keine Sonderzeichen verwenden. Praktisch empfehle ich besonders am Anfang, das Parent-Theme im Namen beizubehalten.

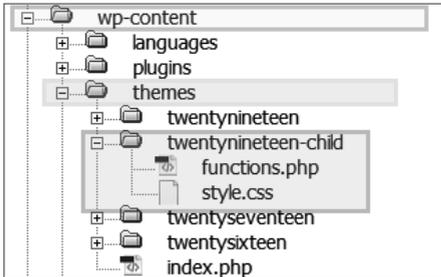


Bild 19.1
Ordnerstruktur bei Verwendung eines Child-Themes

19.1.1 Die style.css im Child-Theme

Im CSS-Header, d.h. am Anfang der Datei `style.css`, müssen zumindest die Angaben *Theme Name* und *Template* enthalten sein. Was bei *Theme Name* angegeben wird, erscheint in der Theme-Verwaltung als Name Ihres neuen Themes. Neben *Template* muss der Name des Eltern-Themes angegeben werden und zwar in der korrekten Schreibweise. *Author*, also Ihren Namen, die *Version* und *Description*, eine kurze Beschreibung, würde ich auch jedenfalls angeben. Demnach sieht der CSS-Header meiner Beispiel-Datei folgendermaßen aus (siehe Listing 19.1):

Listing 19.1 Der CSS-Header des Child-Themes

```

1 /*
2 Theme Name:   Mein neues Twenty Nineteen Child-Theme
3 Description:  WP5-Buch 2019 Twenty Nineteen Child-Theme
4 Author:       Jola Belik
5 Template:     twentyineteen
6 Version:      1.0.0
7 */

```

Im Folgenden finden Sie eine Auflistung aller Angaben, die erwartet werden, wenn Sie Ihr Theme anderen zur Verfügung stellen möchten. Die Zahlen in Klammern zeigen die Position der Angaben in der Themes-Verwaltung in Bild 19.3 und Bild 19.4:

- *Theme name* (1) – Der Name Ihres Child-Themes, Angabe zwingend erforderlich.
- *Theme URI* – Angabe einer Website, in der das Theme beschrieben wird bzw. Beispiel-Anwendungs-Seite, diese Angabe oder die Angabe der Website des Autors bzw. der Autorin ist erforderlich, damit Ihr Theme in der WordPress directory akzeptiert wird.
- *Description* (2) – Kurze Beschreibung des Themes, wird angezeigt, wenn Sie in der Themes-Verwaltung auf **THEME DETAILS** klicken.
- *Author* (3) – Name des Autors bzw. der Autorin, also Ihr Name.
- *Author URI* – Die Adresse Ihrer Website (siehe Theme URI).

- *Template* – Hier steht der Name des Parent-Themes und zwar der Name des Ordners. Diese Angabe ist zwingend erforderlich. Achten Sie auf die korrekte Schreibweise, diese Angabe ist case-sensitiv. Ist dieser Wert falsch angegeben, erhalten Sie Fehlermeldungen statt eines funktionierenden Child-Themes.
- *Version (4)* – Die Versionsnummer beginnt üblicherweise mit 1.0.0.
- *License* – Üblicherweise werden Themes in der WordPress directory unter einer GPL license veröffentlicht. Am besten ist es, Sie übernehmen die Lizenz des Parent-Themes (z.B. GNU General Public License v2 or later).
- *License URI* – Die Adresse, unter der eine Beschreibung der verwendeten Lizenz zu finden ist. Auch hier sollten Sie sich an die Angabe des Parent-Themes halten (z. B. <http://www.gnu.org/licenses/gpl-2.0.html>).
- *Tags* – Stichwörter, sogenannte Tags, erleichtern das Finden von passenden Themes in der WordPress directory. Geben Sie hier aussagekräftige Stichwörter an, die Ihr Thema beschreiben (z.B. one-column, two-columns, responsive, blau, custom-header etc.).
- *Text domain* – Die Angabe des Theme-Slugs wird für die Internationalisierung verwendet, damit Ihr Theme in andere Sprachen übersetzt werden kann (für das Beispiel-Child-Theme wäre es `twentyineteen`).

Fügen Sie den Code aus Listing 19.1 – mit Ihren individuellen Angaben versehen – am Beginn der `style.css` im Ordner `twentyineteen-child` ein und speichern Sie die Datei.

19.1.2 Die `functions.php` im Child-Theme

Damit ein Child-Theme funktionieren kann, müssen die Stylesheets im Child-Theme und im Parent-Theme miteinander verknüpft werden. Dies wurde in früheren Versionen mit `@import` in der Child-CSS gemacht. Diese Vorgangsweise ist *nicht* mehr State of the Art, im WordPress Codex wird stattdessen die Verwendung von `wp_enqueue_script()` und `wp_enqueue_style()` im Child-Theme `functions.php` empfohlen.

Fügen Sie deshalb folgenden Code in die `functions.php` im Child-Theme-Ordner ein (siehe Listing 19.2): Achten Sie darauf, dass sich vor dem öffnenden PHP-Tag `<?php` weder eine Leerzeile noch ein Leerzeichen befindet!

Listing 19.2 Verknüpfen von Child- und Parent-Stylesheets

```
01 <?php
02 function jwp5buch_enqueue_styles() {
03     $parent_style = 'parent-style';
04     wp_enqueue_style( $parent_style, get_template_directory_uri()
05         . '/style.css' );
06     wp_enqueue_style( 'child-style',
07         get_stylesheet_directory_uri() . '/style.css',
08         array( $parent_style ),
09         wp_get_theme()->get('Version')
10     );
11 }
12 add_action( 'wp_enqueue_scripts', 'jwp5buch_enqueue_styles' );
```

19.1.3 Das Child-Theme aktivieren

Das Child-Theme ist nun bereit, aktiviert zu werden. Es ist zwar nicht unbedingt erforderlich, jedoch empfehlenswert und praktischer, wenn Sie – wie für jedes Theme üblich – eine `screenshot.png`-Datei erstellen und in den Child-Theme-Ordner geben (siehe Bild 19.2, rechts). Mit dem Vorschaubild Ihres Child-Themes soll das Theme auf den ersten Blick erkennbar sein. Die erforderliche Mindestgröße, in der diese Vorschaudatei in der Theme-Verwaltung in der Themes-Übersicht angezeigt wird, ist 387×290 Pixel. Allerdings wird die Datei bei der Ansicht der Theme-Details abhängig von der Screen-Größe des Ausgabegeräts deutlich größer angezeigt. Bis vor etwa zwei Jahren war die Empfehlung deshalb 880×660 Pixel. Damit die Vorschau auch auf hochauflösenden Ausgabegeräten gut aussieht, wird aktuell im Codex eine Größe von 1200×900 Pixel empfohlen. Da im oberen und im unteren Teil der Grafik Teile in der Theme-Verwaltung abgedeckt sind, ist es ratsam, wichtige Logos etc. auf dem Screenshot weiter mittig zu platzieren, dass sie sichtbar bleiben. Sie können dafür eine beliebige PNG-Datei oder beispielsweise einen Screenshot der bisher angepassten Seite *Twenty Nineteen* erstellen. Ich verwende einen Screenshot in der Größe von 880×660 Pixel (siehe Bild 19.2, links).

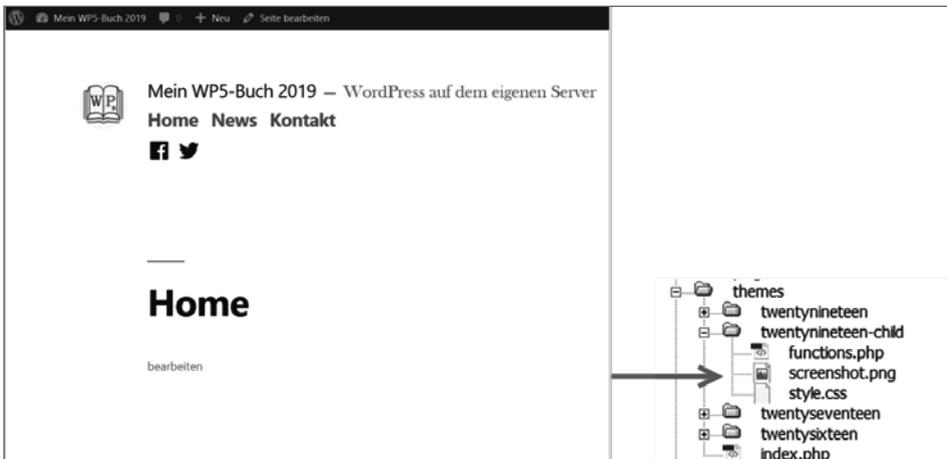


Bild 19.2 Die Datei `screenshot.png` im Child-Theme-Ordner (rechts)

Öffnen Sie nun die *Themes-Verwaltung* auf dem Dashboard über *Design/Themes*. An erster Stelle ist immer das aktuell aktivierte Theme zu sehen. Das neue Child-Theme *Mein neues Twenty Nineteen Child Theme* wird im Beispiel gleich daneben angezeigt (siehe Bild 19.3).

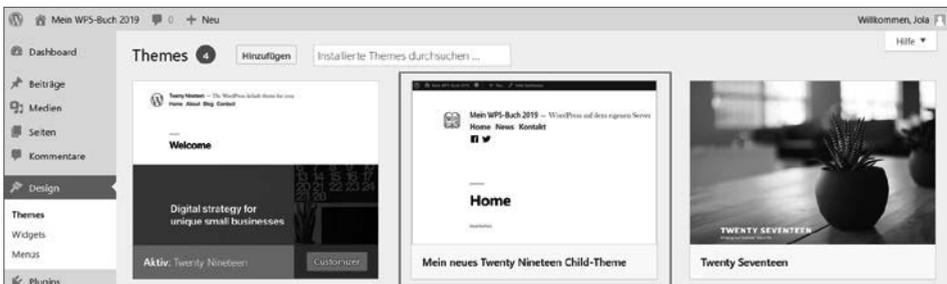


Bild 19.3 Das neue Child-Theme (rechts) in der Themes-Verwaltung

Fahren Sie mit der Maus über das Vorschaubild und klicken Sie auf **THEME-DETAILS**. Sie sehen nun die Angaben, die im CSS-Header des Child-Themes eingefügt wurden: *Theme Name* (1), *Versionsnummer* (4), *Autorname* (3) und die *Beschreibung* (2). Außerdem erscheint der Hinweis, dass es sich um ein *Child-Theme von Twenty Sixteen* handelt (siehe Bild 19.4).

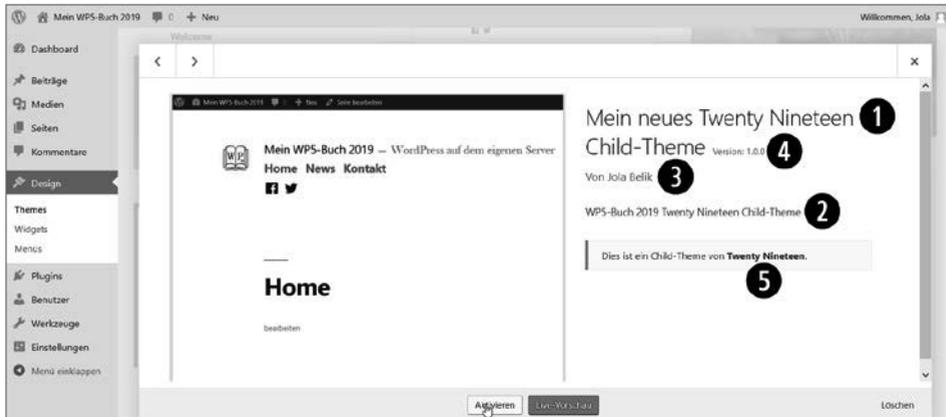


Bild 19.4 Aktivieren Sie das neue Child-Theme

Sie können nun das neue Child-Theme aktivieren. Klicken Sie unterhalb des Vorschaubilds auf **AKTIVIEREN** (siehe Bild 19.4). Warten Sie, bis das neue Theme aktiviert wurde, das kann ein Weilchen dauern. Wenn alles erfolgreich durchgeführt werden konnte, erscheint oben auf der Seite der Themes-Verwaltung eine Erfolgsmeldung (siehe Bild 19.5).

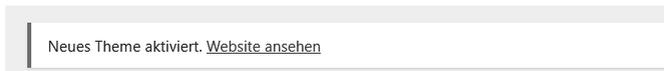


Bild 19.5 Das neue Child-Theme wurde erfolgreich aktiviert!

Gratulation, Sie können nun mit dem Child-Theme arbeiten und weitere Anpassungen vornehmen. Die Freude währt allerdings nur so lange, bis Sie sich das Frontend anschauen. Statt den bisher vorgenommenen Änderungen und Anpassungen erscheint der Widget-Titel wieder riesengroß, zudem ist das Website-Logo verschwunden (siehe Bild 19.6).



Bild 19.6 Das Frontend nach der Aktivierung des Child-Themes

Nein, es ist kein Fehler in Ihrem Code, der das verursacht. Es ist lediglich die übliche Problematik beim Aktivieren eines neuen Themes. Sie müssen in den meisten Fällen nicht nur die bereits erwähnten „verschwundenen“ Anpassungen wieder durchführen. Wenn im neuen Theme Menü- und Widgets-Positionen anders benannt sein sollten (was im Beispiel-Child-Theme nicht der Fall ist), müssen zusätzlich auch die Menüs neu zugewiesen und die Widgets neu hinzugefügt werden.

■ 19.2 Änderungen im Child-Theme

Nachdem unser Child-Theme aktiviert wurde, kann das Theme nach Belieben angepasst werden. Als Beispiele in diesem Kapitel werden die Copyright-Zeile mit Anmelde-Link im Footer und RSS-Link samt RSS-Icon sowie diverse Änderungen bei Textformatierungen und Anzeige eines größeren Site-Logos gezeigt. In einigen anderen Kapiteln werden weitere Änderungen bzw. Erweiterungen des Themes beschrieben wie beispielsweise das Hinzufügen eines neuen Widgets-Bereichs (siehe Abschnitt 7.1) und Hinzufügen einer benutzerdefinierten Bildgröße in der Mediathek. Vorher wurden die im Customizer (Kapitel 6) gemachten Änderungen bezüglich Site-Logo und Farben nochmals durchgeführt – als primäre Farbe wurde nun Rot statt Blau gewählt. Das Site-Logo muss nicht nochmals hochgeladen werden, da es sich in der Mediathek befindet.

19.2.1 Neue Footer-Zeile

Als Erstes werden einige Änderungen im *Footer*, konkreter gesagt in der Datei `footer.php` im Child-Theme-Ordner, vorgenommen. Im Folgenden wird eine neue Copyright-Zeile mit diversen Links erstellt.

19.2.1.1 Neue ©-Zeile im Footer

Der Footer mit der Zeile *Mein WP5-Buch 2019, Stolz präsentiert von WordPress.* und der Verlinkung mit `wordpress.org` mag ja nett sein (siehe Bild 19.7, oben). Wenn Sie unter *Einstellungen/Datenschutz* die Seite für die Datenschutzerklärung festgelegt und diese Seite unter *Seiten* bearbeitet und veröffentlicht haben, so wird im Footer auch der Link zur *Datenschutzerklärung* angezeigt (siehe Bild 19.7, oben). Sinnvoller fände ich eine Fußzeile mit einem Copyright-Hinweis samt Namen und Jahreszahl und ohne Verlinkung zu *wordpress.org* sowie den Link zur Datenschutzerklärung und einen Link zum Anmelden (siehe Bild 19.7, unten).



Bild 19.7 Fußzeile in Twenty Nineteen ohne (oben) und mit veröffentlichter Seite Datenschutzerklärung (unten)

Link zur Datenschutzerklärung Ob der Link zur *Datenschutzerklärung* im Footer oder an einer anderen Stelle platziert wird, ist individuell zu entscheiden. Mit folgenden Code-Zeilen wird der Link automatisch an der gewünschten Stelle eingefügt. Zuerst wird überprüft, ob eine Seite für den Datenschutz definiert wurde (siehe Listing 19.3, Zeile 2). Falls ja, so wird der Link zur Datenschutzerklärung mit `the_privacy_policy_link(' | ', ' | ')` eingefügt. Die beiden Parameter definieren, was vor und nach dem Link eingefügt werden soll, im Beispiel jeweils ein Separator zum vorherigen bzw. nachfolgenden Inhalt. Beachten Sie, dass es sich hier jeweils um einzelne Hochkommas handelt.

Listing 19.3 Code zum Einfügen des Links zur Datenschutzerklärung

```
1 <?php
2   if ( function_exists( 'the_privacy_policy_link' ) ) {
3       the_privacy_policy_link( ' | ', ' | ' );
4   }
5 ?>
```

Änderungen in der Datei footer.php Die Änderungen im Footer werden nicht in der Originaldatei durchgeführt! Kopieren Sie die Datei `footer.php` aus dem Parent-Theme-Ordner in den Child-Theme-Ordner. In dieser Datei wird nun der Quellcode verändert. Nach dem Speichern der Datei (wenn Sie nicht auf XAMPP arbeiten, nach dem Hochladen der Datei in den Child-Theme-Ordner!) wird die Originaldatei im Parent-Theme beim Laden der Seite überschrieben.

1. Kopieren Sie die Datei `footer.php` aus dem Ordner `themes/twentyineteen` in den Child-Theme-Ordner `twentyineteen-child`.
2. Öffnen Sie die Datei `footer.php` und suchen Sie die Zeile mit `<div class="site-info">` (Zeile 20 in Bild 19.8). Darunter befinden sich die einzelnen Code-Blöcke für die Fußzeile und zwar:

In Zeile 21 – 24 der *Name der Website* verlinkt mit der Startseite (1 in Bild 19.8) – *kann gelöscht werden*. Im Beispiel wird dies durch einen Copyright-Hinweis ersetzt.

In Zeile 25 – 30 der Text *Stolz präsentiert von WordPress* mit einer Verlinkung zu `WordPress.org` und dem Theme *Twenty Nineteen* (2 in Bild 19.8) – *kann gelöscht werden*.

In Zeile 31 – 35 der Link zur Seite mit der *Datenschutzerklärung*, falls diese Seite definiert und bereits veröffentlicht wurde (3 in Bild 19.8) – nur dann löschen, falls Sie die Datenschutzerklärung nicht im Footer verlinken möchten. Meine Empfehlung: *nicht löschen*.

In Zeile 36 beginnt der Code für die *Menü-Position Footer* (4 in Bild 19.8) – *nicht löschen!*

```

17
18 <footer id="colophon" class="site-footer">
19 <?php get_template_part( 'template-parts/footer/footer', 'widgets' ); ?>
20 <div class="site-info">
21 <?php $blog_info = get_blogo_info( 'name' ); ?>
22 <?php if ( ! empty( $blog_info ) ) : ?>
23 <a class="site-name" href="<?php echo esc_url( home_url( '/' ) ); ?>" rel="home"><?php
bloginfo( 'name' ); ?></a>,
24 <?php endif; ?>
25 <a href="<?php echo esc_url( __( 'https://wordpress.org/', 'twentyineteen' ) ); ?>" class=
"imprint">
26 <?php
27 /* translators: %s: WordPress. */
28 printf( __( 'Proudly powered by %s.', 'twentyineteen' ), 'WordPress' );
29 ?>
30 </a>
31 <?php
32 if ( function_exists( 'the_privacy_policy_link' ) ) {
33 the_privacy_policy_link( ' ', '<span role="separator" aria-hidden="true"></span>' );
34 }
35 ?>
36 <?php if ( has_nav_menu( 'footer' ) ) : ?>
37 <nav class="footer-navigation" aria-label="<?php esc_attr_e( 'Footer Menu',
'twentyineteen' ); ?>">
38 <?php

```

Bild 19.8 Website-Name (1), Link zu WordPress (2), Datenschutzerklärung (3) sowie Beginn der Menü-Position Footer (4) in der Datei footer.php

- Da nur mehr eine benutzerdefinierte Zeile angezeigt werden soll, löschen Sie die Zeilen 21 bis 30, d.h. von `<?php $blog_info` in Zeile 21 bis inklusive `` in Zeile 30. Übrig bleiben ein öffnendes DIV mit der Klasse `.site-info` in Zeile 20 und – mit einigen Leerzeilen dazwischen – der PHP-Block für die Datenschutzerklärung und darunter die Definition für die Menü-Position *Footer* (siehe Bild 19.9).

```

17
18 <footer id="colophon" class="site-footer">
19 <?php get_template_part( 'template-parts/footer/footer', 'widgets' ); ?>
20 <div class="site-info">
21
22
23 <?php
24 if ( function_exists( 'the_privacy_policy_link' ) ) {
25 the_privacy_policy_link( ' ', '<span role="separator" aria-hidden="true"></span>' );
26 }
27 ?>
28 <?php if ( has_nav_menu( 'footer' ) ) : ?>
29 <nav class="footer-navigation" aria-label="<?php esc_attr_e( 'Footer Menu',

```

Bild 19.9 Blog-Info und Link zu WordPress wurden gelöscht

- Fügen Sie das Copyright-Symbol ©, Ihren *Namen* und die aktuelle *Jahreszahl* sowie *Alle Rechte vorbehalten* in die Zeile unterhalb des öffnenden DIV-Tags ein. Da das Copyright-Symbol ein Sonderzeichen ist, wird es maskiert und als Entity `©` eingefügt. Die aktuelle Jahreszahl soll automatisch eingelesen werden, dafür verwenden Sie den PHP-Codeblock `<?php echo date('Y'); ?>`. Zudem ergänze ich noch die Jahreszahl, in der die Website (oder Firma) gestartet wurde. Meine Zeile lautet somit `© J. Belik 2018-<?php echo date('Y'); ?> Alle Rechte vorbehalten.` (siehe Zeile 21 in Bild 19.10, oben). Das Ergebnis im Frontend zeigt Bild 19.10, unten.

```

20 <div class="site-info">
21 scopy; J. Belik 2018-<?php echo date('Y'); ?> Alle Rechte vorbehalten. | ←
22 <?php
23     if ( function_exists( 'the_privacy_policy_link' ) ) {

```

© J. Belik 2018-2019 Alle Rechte vorbehalten. | Datenschutzerklärung

Bild 19.10 Die neue Fußzeile im Child-Theme

19.2.1.2 Anmelde-Link im Footer

Ich möchte statt des Widgets *Meta* einen Anmelde-Link mit *Anmelden* in die Fußzeile geben. Wenn ich eingeloggt bin, soll *Abmelden* angezeigt werden. Ist man angemeldet, so wird im Widget *Meta* neben *Abmelden* auch ein Link *Website-Administration* angezeigt. Mit Klick auf diesen Link gelangt man direkt aufs Dashboard. Dies soll auch im Footer angezeigt werden.

1. Damit man sich über einen Link in der Fußzeile einloggen kann, wird `<?php wp_loginout(); ?>` in Zeile 27 direkt oberhalb des Code-Blocks für die *Menü-Position Footer* hinzugefügt (siehe Bild 19.11, oben). Damit erscheint im Frontend *Anmelden* bzw. *Abmelden* in der Fußzeile (siehe Bild 19.11, Mitte und unten).

```

21 scopy; J. Belik 2018-<?php echo date('Y'); ?> Alle Rechte vorbehalten. |
22 <?php
23     if ( function_exists( 'the_privacy_policy_link' ) ) {
24         the_privacy_policy_link( '', '<span role="separator" aria-hidden="true"></span>' );
25     }
26     ?>
27     <?php wp_loginout(); ?> ←
28     <?php if ( has_nav_menu( 'footer' ) ) : ?>

```

© J. Belik 2018-2019 Alle Rechte vorbehalten. | Datenschutzerklärung Anmelden ←

© J. Belik 2018-2019 Alle Rechte vorbehalten. | Datenschutzerklärung Abmelden ←

Bild 19.11 Anmelden und Abmelden im Footer

2. Zum einfacheren Administrieren wäre es praktischer, wenn nicht nur der Link *Abmelden*, sondern auch *Website-Administration*, wie sie im Widget *Meta* angezeigt wird, erscheint, sobald man angemeldet ist. Dazu wird `<?php wp_register(' | ', ''); ?>` verwendet. Beachten Sie, dass die Schaltfläche *Registrieren* nur dann verfügbar ist, wenn die Option *Jeder darf sich registrieren* aktiviert wurde. Ist man eingeloggt, so erscheint der Direkt-Link *Website-Administration* zum Dashboard unabhängig davon, ob man sich registrieren darf oder nicht.

Defaultmäßig ist der erste Parameter in der Klammer ein öffnender ``-Tag, der zweite Parameter ein schließender ``-Tag. Die Schaltfläche soll jedoch nicht als Listenelement (mit einem Listenpunkt davor) in der Fußzeile angezeigt werden. Außerdem soll sie zwecks besserer Optik durch einen senkrechten Strich | von Abmelden getrennt werden. Deshalb wird der erste Parameter als senkrechter Strich und ein Leerzeichen ' | ' davor und dahinter definiert und der zweite Parameter als leeres Element '' angegeben. Beachten Sie, dass es sich jeweils um einfache Hochkommas handelt. Nach dem Speichern der Datei (und Hochladen per FTP, wenn Sie nicht auf XAMPP arbeiten)

kann man sich das Ergebnis im Frontend anschauen. Nach dem Anmelden erscheinen die Links *Abmelden* und *Website-Administration* (siehe Bild 19.12, unten).

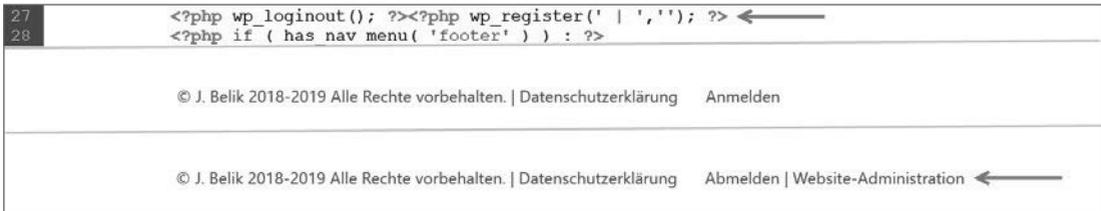


Bild 19.12 Ergänzter Code für Link Anmelden und Registrieren (oben), Anmelden (Mitte) sowie Abmelden und Website-Administration (unten)

19.2.1.3 RSS-Link im Footer

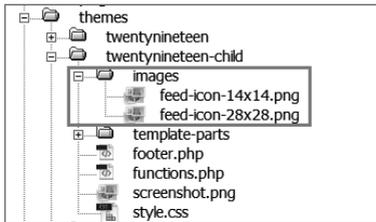
Wenn Sie häufig Beiträge veröffentlichen, ist es sinnvoll und üblich, einen *RSS-Link* zu Beiträgen in die Sidebar oder in die Fußzeile zu geben. Da das *Twenty Nineteen* Theme keine Sidebar hat, kommt der Link in den Footer. Die Adresse des Beiträge-Feeds hat die Form `www.meinedomain.xyz/feed/`. Sie könnten einen Link natürlich auch hard-coded mit `Beitrags-Feed (RSS)` einfügen. Falls Sie später von HTTP auf HTTPS umsteigen, haben Sie einen unsicheren Link auf Ihren Seiten und Sie müssen nachträglich den Link anpassen (siehe Kapitel 3). Daher ist es empfehlenswert, den Link zum Feed mit `<?php bloginfo('rss2_url'); ?>` automatisch ausgeben zu lassen. Somit lautet der Link `<a href="<?php bloginfo('rss2_url'); ?>" title="RSS Feed"> Beitrags-Feed (RSS)`. Bild 19.13, oben, zeigt das Ergebnis im Frontend.



Bild 19.13 RSS-Beitrags-Link ohne (oben) und mit RSS-Icon (unten)

Ansprechender und professioneller wirkt der RSS-Link mit dem passenden Icon (siehe Bild 19.13, unten). Unter der Adresse `http://www.feedicons.com/` können Sie die offiziellen *RSS-Icons* in diversen Grafikformaten und Farben downloaden. Ich verwende das *Standard Icon Bundle* mit zwei PNG-Dateien in der Größe 28×28 px und 14×14 px, die beiden Dateien heißen `feed-icon-14x14.png` und `feed-icon-28x28.png`. Diese wurden in den `images`-Ordner im Child-Theme-Ordner entpackt.

1. Laden Sie die gewünschten RSS-Icons auf Ihren Rechner.
2. Falls noch nicht vorhanden, erstellen Sie einen Ordner mit dem Namen `images` in Ihrem Child-Theme-Ordner.
3. Entpacken Sie die RSS-Icons in den Ordner `images` im Child-Theme-Ordner.

**Bild 19.14**

Die RSS-Icons im images-Ordner im Child-Theme-Ordner

- Die Grafik soll vor dem Link angezeigt werden, dies wird mit einer Klasse in der Datei `style.css` im Child-Theme-Ordner definiert. Die Klasse bekommt im Beispiel den Namen `.rss`, die Grafik wird als Hintergrundgrafik festgelegt (siehe Zeile 5 in Listing 19.4). Öffnen Sie die Datei `style.css` im Child-Theme-Ordner und fügen Sie die CSS-Regel aus Listing 19.4 ein. Speichern Sie die Datei.

Listing 19.4 Klasse `.rss` in der `style.css` des Child-Themes

```
1 /* RSS-Icon */
2 .rss {
3     margin-left: 2em;
4     padding: 0 0 0 20px;
5     background: url("images/feed-icon-14x14.png") no-repeat 0 50%;
6 }
```

- Wechseln Sie in die Datei `footer.php` in Ihrem Child-Theme-Ordner. Um den Link herum wird ein ``-Tag mit der Klasse `.rss` gelegt. So lautet der Link mit dem RSS-Icon im Footer nun `<a href="<?php bloginfo('rss2_url'); ?>" title="RSS Feed">RSS-Beiträge`. Fügen Sie diesen Link nach dem PHP-Block mit `wp_register` ein (siehe Zeile 28 in Bild 19.15). Das Ergebnis im Frontend zeigt Bild 19.13, unten.

```
27     <?>
28     <?php wp_loginout(); ?><?php wp_register(' | ', ''); ?> <span class="rss"><a
29     href="<?php bloginfo('rss2_url'); ?>" title="RSS Feed">Beitrags-Feed (RSS)</a></span>
30     <?php if ( has_nav_menu( 'footer' ) ) : ?>
```

Bild 19.15 Fügen Sie den RSS-Link mit RSS-Icon in Zeile 28 ein

19.2.2 Site-Logo größer anzeigen

Das *Site-Logo* wird im Theme *Twenty Nineteen* links oben neben Titel und Navigationsleiste in einem runden Ausschnitt angezeigt. Bis zu einer Browserfenster-Breite von mindestens 768 Pixel hat es eine Größe von 50×50 px, auf schmälere Monitoren 64×64 px. Dies ist meiner Meinung nach besonders auf großen Monitoren viel zu klein (siehe Bild 19.16, links). Deshalb möchte ich das Site-Logo doppelt so groß anzeigen lassen (siehe Bild 19.16, rechts).



Bild 19.16 Ursprüngliche Größe des Site-Logos (links) und neue Größe (rechts)

Dazu wird eine bestehende CSS-Regel aus dem Parent-Theme im Child-Theme neu definiert und zwar die Klassen `.site-logo` und `.custom-logo-link`. Am einfachsten ist es, die bestehende Regel aus der Datei `style.css` im Parent-Theme-Ordner in die Datei `style.css` im Child-Theme-Ordner zu kopieren und dort wunschgemäß zu bearbeiten.

Mit Hilfe von *Media Queries* können beim Design von responsiven Layouts CSS-Regeln für unterschiedliche Ausgabemedien definiert werden. Im Beispiel soll die CSS-Regel für das größere Site-Icon nur auf dem Monitor und nur ab einer Mindestbreite von 768 Pixel gelten.

Listing 19.5 Neue Größe 100 × 100px für Site-Logo für Screens ab 768px Breite

```

1 /* Site-Logo 100x100px */
2 @media only screen and (min-width: 768px) {
3   .site-logo .custom-logo-link {
4     width: 100px;
5     height: 100px;
6   }
7 }

```

Da die „große“ Version des Site-Logos nur auf Screens ab mindestens 768px Breite angezeigt werden soll, wird ein *Media Query* in der Form `@media only screen and (min-width: 768px) { ... }` um die CSS-Regel herum gelegt (siehe Zeile 2 und 7 in Listing 19.5). Die CSS-Regel wird zwischen die geschwungenen Klammern `{ }` gesetzt. Beachten Sie, dass es nun zwei geschwungenen Klammern `}` gibt, die erste von der CSS-Regel, die zweite von der Media Query. Fügen Sie die CSS-Regel samt Media Query aus Listing 19.5 in die `style.css` im Child-Theme-Ordner ein (siehe Bild 19.17).

```

9 /* RSS Icon */
10 .rss {
11   margin-left: 2em;
12   padding: 0 0 0 20px;
13   background: url("images/feed-icon-14x14.png") no-repeat 0 50%;
14 }
15
16 /* Site-Logo 100x100px */
17 @media only screen and (min-width: 768px) {
18   .site-logo .custom-logo-link {
19     width: 100px;
20     height: 100px;
21   }
22 }
23

```

Bild 19.17 RSS-Icon und neue Größe für Site-Logo in der `style.css`

Index

Symbole

<!--more-->-Tag 368
\$wp_admin_bar 155
\$wp_customize 153

A

Abmelden 93
Administration 93
Akismet 397
Aktivität 107
Aktualisierungen
– manuell 90
– One-Click-Updates 83
Allgemeine Blöcke 217
An-/Abmelden Link 565
Anhang-Details 139, 232
Anmelden 93
Anpassen 134
Ansicht anpassen 104
Ansichtsmodus 372
Apache Server 25
API Schlüssel
– Akismet 403
Attachments 232
Auf einen Blick 105
Auszug 368
– automatisch 368
– manuell 368
Automatic 1
Autor 366
AutoSave 195
Avatar 114

B

Bearbeiten-Link 313, 572
Beitrag
– löschen 366
– Verwaltung 370
Beitrags-Attribute 370
Beitragsbild 367, 555
Beitragsformat 370
Beitragsnavigation 572
Benutzer
– Abonnent 381
– Autor 385
– hinzufügen 389
– löschen 391
– Rechte 377
– Redakteur 386
– Rolle ändern 391
– Rollen erweitern 394
– verwalten 388
Benutzerdefinierte Felder *siehe* Eigene Felder
Benutzerdefiniertes Feld
– außerhalb Loop 324
– innerhalb Loop 325
Benutzerkonto hinzufügen 30
Benutzername 12
Benutzerprofil 95
Bild
– Alternativtext 234
– Beschreibung 234
– Bildunterschrift 234
– Original-Datei wiederherstellen 238
– skalieren 241
– Titel 234
– Unwiderruflich löschen 233
Bild-Block
– Bild-Einstellungen 227
– Link-Einstellungen 227

Block-Editor

- Absatz linksbündig 213
- Absatz rechtsbündig 213
- Absatz zentriert 213
- Absatz-Block 217
- Anker 215
- Anker-Links 215
- Archive-Block 272
- Audio-Block 218
- Block entfernen 201
- Blocksatz 213
- Block-Stile 291
- Block-Typ ändern 199
- Button-Block 267
- Cover-Block 242
- Crowdsignal-Block 284
- Datei-Block 244
- Editor Styles 289
- Facebook-Block 277
- Farbeinstellungen 212
- Farb-Palette 211
- Hyperlinks 214
- Initialbuchstabe 212
- JSON-Blöcke 205
- Kalender-Block 272
- Kategorien-Block 273
- Layout-Elemente 266
- Liste-Block 249
- Medien-und-Text-Block 267
- Mehr-Block 268
- Neue Beiträge Block 273
- Overlay-Farbe 242
- Pullquote-Block 260
- RSS-Block 274
- Schlagwörter 196
- Schlagwörter-Wolke-Block 275
- Seitenumbruch-Block 269
- Shortcode-Block 275
- Spalten-Block 270
- Suchen-Block 276
- Tabelle-Block 261
- Textauszug 197
- Titel 197
- Trennzeichen-Block 271
- Überschrift-Block 251
- Vers-Block 264
- Video-Block 252
- Vorformatiert-Block 265
- Werkzeuggestreife 181
- wiederverwendbare Blöcke 201
- YouTube-Block 279

Blöcke-Verwaltung 204

Block-Manager 183

Block-Navigation 182

Blogroll 177, 358

Brute Force Attacke 75

C

Calendly 412

Child-Theme 479

- aktivieren 482
- CSS-Header 480
- einrichten 479
- footer.php 485
- functions.php 481

Cipher Suites 66

Classic Block 301

Classic Editor 301

Classic Editor Plugin 306

Color Picker 551

Control Panel 25

Copyright-Zeile 484

CSS Font Stack 505

CSS Shape Generator 297

Cues 257

Custom Fields *siehe* Eigene Felder

customize_register 153

Customizer 133, 134

- Section entfernen 153
- Vorschau 136

Customizer entfernen 156

Custom Plugin *siehe* Site-spezifisches Plugin

D

Dashboard 93

Dashicons 507

Daten importieren 46

Datenschutzerklärung 123

Datenschutz-Hinweis 403

Debug-Modus

- aktivieren 450

DeepL 443

Diskussion 369

Dokument-Bedienfelder 315

dunkle Editor Stile 553

E

Editor

- Themes 157

Eigene Felder 315, 321

Einstellungen

- Allgemein 111

- Permalinks 121
- Persönliche Optionen 96
- Sichtbarkeit für Suchmaschinen 117
- Sprache der Website 112
- Standardkategorie 114
- Startseite zeigt 116
- EV-TLS-Zertifikat 67
- Excerpt 197, 368
- Exportieren
 - von WordPress.com 44

F

- Farbnamen 511
- Farb-Palette 551
 - deaktivieren 551
 - Theme-spezifische 551
- Farbschema 96
- Farb-Triade 517
- Fatal Error Recovery Mode 91
- Featured Image *siehe* Beitragsbild

G

- Github 436
- Google Maps 406
- Grafik-Block 304
- Gutenberg
 - Spotlight-Modus 183
 - Vollbild-Modus 183

H

- Hellip 368
- Hello Dolly 397
- HSL-Farbraum 516
- HSTS-Header 72
- HTML-Block 304
- httpd.conf 27
- httpd-ssl.conf 27
- HTTPS 66
- HTTPS Checker 73

I

- Icon Fonts 507
- iFrame 304
- individuelle Farben *siehe* Color-Picker
- individuelles Menü *siehe* Navigationsmenü
- individuelle Schriftgröße
 - deaktivieren 553
- Info-Box 395

J

- Jetpack
 - Diesen Beitrag teilen 424
 - Kurzlink 424
 - Likes und Teilen 424
 - Sichtbarkeit von Widgets 425
- Jetpack-Block
 - Abonnementformular 412
 - Ähnliche Beiträge 422
 - Calendly 412
 - Diashow 412
 - Einfaches-Bezahlen-Button 413
 - Eventbrite-Bezahlungsfunktion 414
 - Folgebesuche 414
 - Formular 415
 - Gekachelte Galerie 417
 - Geschäftszeiten 421
 - GIF 418
 - Google Calendar 418
 - Karte 419
 - Kontaktinfo 419
 - Mailchimp 420
 - Markdown 420
 - OpenTable 421
 - Pinterest 421
 - Revue 422
 - Sterne-Bewertung 423
 - Wiederkehrende Zahlungen 422
- Jetpack-Widget
 - Autoren 425
 - Blog-Abonnements 427
 - Blogstatistik 427
 - Einfache Bezahlung 427
 - EU-Cookies-Banner 425
 - Facebook-Seiten-Plugin 428
 - Flickr 428
 - Goodreads 429
 - Google Translate 429, 438
 - Gravatar-Profil 429
 - Internet Defense League 430
 - Kommende Veranstaltungen 430
 - Kontakt-Info & Karte 430
 - MailChimp Abonnenten-Popup 431
 - Meilenstein 431
 - Meine Community 431
 - RSS-Links 431
 - Soziale Icons 432
 - Top Beiträge und Seiten 432
 - Twitter Timeline 433
 - WordPress-Beiträge anzeigen 433
- JPEG-Komprimierung deaktivieren 560

K

- Kategorie 346
 - bearbeiten 348
 - Bedienfeld 350
 - Beiträge anzeigen 354
 - Beitrags-Zähler 352
 - erstellen 346
 - Hauptkategorie 351
 - Quickedit 349
 - Unterkategorie 351
 - Widget 352
 - Wolke 353
- Kategorien 367
- Kategorie-Schlagwort-Konverter 359
- Kategorie-Verwaltung 346
- Kollation 29
 - utf8_general_ci 29
- Kommentar
 - Formular 370
- Kontaktformular 52, 318

L

- Lesedauer 569
- Let's Encrypt 67
- Libtheora 255
- Lingotek 440
- Link-Manager 358
- Link-Optionen 214
- Link-Text 330
- Linkziel 329
- Listenansicht 372
- Location Identifier 334
- Login-Formular
 - eigenes Logo 453

M

- MariaDB 22
- Maßeinheiten
 - absolute 502
 - relative 502
- MD5 Hash 41
- Media Queries 490, 528
- Mediathek
 - Datei hochladen 342
 - Datei löschen 343
 - Kachel-Ansicht 338
 - Listen-Ansicht 340
- Mehr-Block 374
- mehrere WordPress-Installationen 13, 41

Menü

- erstellen 328
- löschen 333
- Menü-Position
 - registrieren 334
- Menüpunkte hinzufügen 329
- Meta-Boxen 108
- Meta-Box entfernen 109
- Meta-Daten 321
- Meta-Tag Description 321
- Mindestanfordernisse 6
- Mixed Content 72
- Mobile-first 500
- Multi-Site 3, 13
 - aktivieren 13
 - deaktivieren 17
- Multi-User 3
- MySQL 25

N

- Nag-Boxen 108
- Navigationsmenü 173
- Netstat 27

O

- Öffentlicher Name 97
- OpenStreetMap 406

P

- Paginierung 319
- personenbezogene Daten 124
 - exportieren 125
 - löschen 129
- phpMyAdmin 28
 - Datenbank 29
- PID 4 27
- Pingbacks 369
- Pinsel entfernen 155
- Plugin
 - Akismet 404
 - Akismet Datenschutz-Hinweis 404
 - bearbeiten 401
 - Captcha Code 405
 - Disable REST API 405
 - Disable XML-RPC Pingback 405
 - Google XML Sitemaps 406
 - Google XML Sitemaps v3 for qTranslate 435
 - GTranslate Free 438
 - installieren 56

- Jetpack 406
- miniOrange 2 Factor Authentication 434
- Polylang 439
- qTranslate X 434
- qTranslate-XT 436
- TranslatePress 442
- Translate WordPress with GTranslate 438
- Widget GTranslate 438
- WPGlobus 441
- WP Updates Notifier 434
- Plugin-Editor 401
- Plugin installieren 448
- Plugin Repository *siehe* Plugin-Verzeichnis
- Plugin-Verwaltung 397
- Plugin-Verzeichnis 402
- Port 80 27
- Port 443 27
- Port-Probleme 27
- Post Formats *siehe* Beitragsformate
- Priorität 155

Q

QuickEdit 312, 372

R

remove_control() 153
 remove_menu() 155
 remove_panel() 153
 RGB-Farbraum 510
 RSS-Icons 488

S

Schlagwörter 356, 367

- Quickedit 357
- verwalten 357
- Widget 358
- Wolke 358

 Schnellobearbeitung *siehe* QuickEdit
 Schneller Entwurf 106
 Schriftgrößen

- Theme-spezifisch 554

 screenshot.png 482
 Seitenumbruch 318
 Seitenumbruch-Block 318
 Seiten-Verwaltung 311
 Sichtbarkeit 362
 Sidebar 159
 Site Health Check *siehe* Website-Zustand
 Site-Icon 550

Site-Logo 138, 489, 496, 550
 Site-spezifisches Plugin 445
 Skip to Content Link 560
 Slug 346
 Sprache festlegen 96
 SSL 67
 SSL Check 72
 Standard-Beitragsformat 370
 Standard-Editor

- zum Block-Editor wechseln 309
- Box 309
- zum Classic Editor wechseln 309
- festlegen 307
- individuell festlegen 307
- wechseln 308

 Standard-Kategorie 346
 Standard-Widget

- Archive 169
- Audio 170
- Bild 170
- Galerie 170
- HTML 171
- Kalender 172
- Kategorien 172
- Links 177
- Meta 172
- Neueste Beiträge 173
- Neueste Kommentare 174
- RSS 174
- Seiten 175
- Standard-Widget Schlagwörter-Wolke 174
- Suche 175
- Text 176
- Video 176

 Startseiten-Einstellungen 151
 Status und Sichtbarkeit 362
 Sticky Post 364, 373
 Syntaxfehler 449

T

Tabellen-Block 304
 Tags *siehe* Schlagwörter
 Tag Cloud *siehe* Schlagwörter-Wolke
 Teaser 368, 374
 Tellerrand 94
 Template-Hierarchie 540
 Template-Partials 541
 Textauszug-Ansicht 372
 the_content() 368
 the_excerpt() 368
 Theme

- auf Test-Server 54

- installieren 55
- screenshot.png 54
- wechseln 137
- Theme-Editor 157
- Theme-Support 547
 - alignwide und alignfull 555
 - automatische RSS-Feed Links 548
 - Beitragsformate 548
 - benutzerdefinierter Hintergrund 548
 - Color-Picker deaktivieren 551
 - Custom Title 549
 - Default-Header 549
 - dunklen Editor Stil anzeigen 553
 - editor-styles 551
 - Farb-Palette deaktivieren 551
 - HTML5 Markup 555
 - Individuelle Schriftgrößen deaktivieren 553
 - Post Thumbnails 555
 - Responsive Einbettungen 554
 - Schriftgrößen-Vorgaben 554
 - Selektiver Refresh im Customizer 550
 - Standard-Stile anzeigen 553
 - Starter Content 555
 - Theme-Logos 549
 - Theme-spezifische Farb-Palette 551
- Theora 255
- Titel 189, 496
- TLS 67
- TLS-Handshake 66
- Trackbacks 369
- Triadisches Farbschema 517
- Twenty Nineteen 133

U

- Unerwarteter oder ungültiger Inhalt 305
- Universal-Selektoren 502
- Untertitel 256, 496

V

- Veranstaltungen und Neuigkeiten 107

W

- Web-Fonts 506
- Website-Icon 138, 141
- Website-Informationen 138
- Website-Zustand 131
- Weiterlesen Link 374
- Werkzeugleiste 94
- White Screen of Death 91
- Widget
 - Akismet 404
 - Meta 93
 - Navigationsmenü 333
- Widgets-Bereich 159, 498
- Widgets-Verwaltung
 - Standardmodus 165
 - Zugänglichkeitsmodus 167
- Wiederherstellungsmodus 91
- Wildcard TLS-Zertifikat 67
- WordPress Importer 47
- wp_loginout() 487
- WPML 442
- wp_register() 487

X

- XAMPP 21
- XAMPP Control Panel 60

Z

- Zugänglichkeitsmodus 167
- Zusätzliches CSS 52, 151