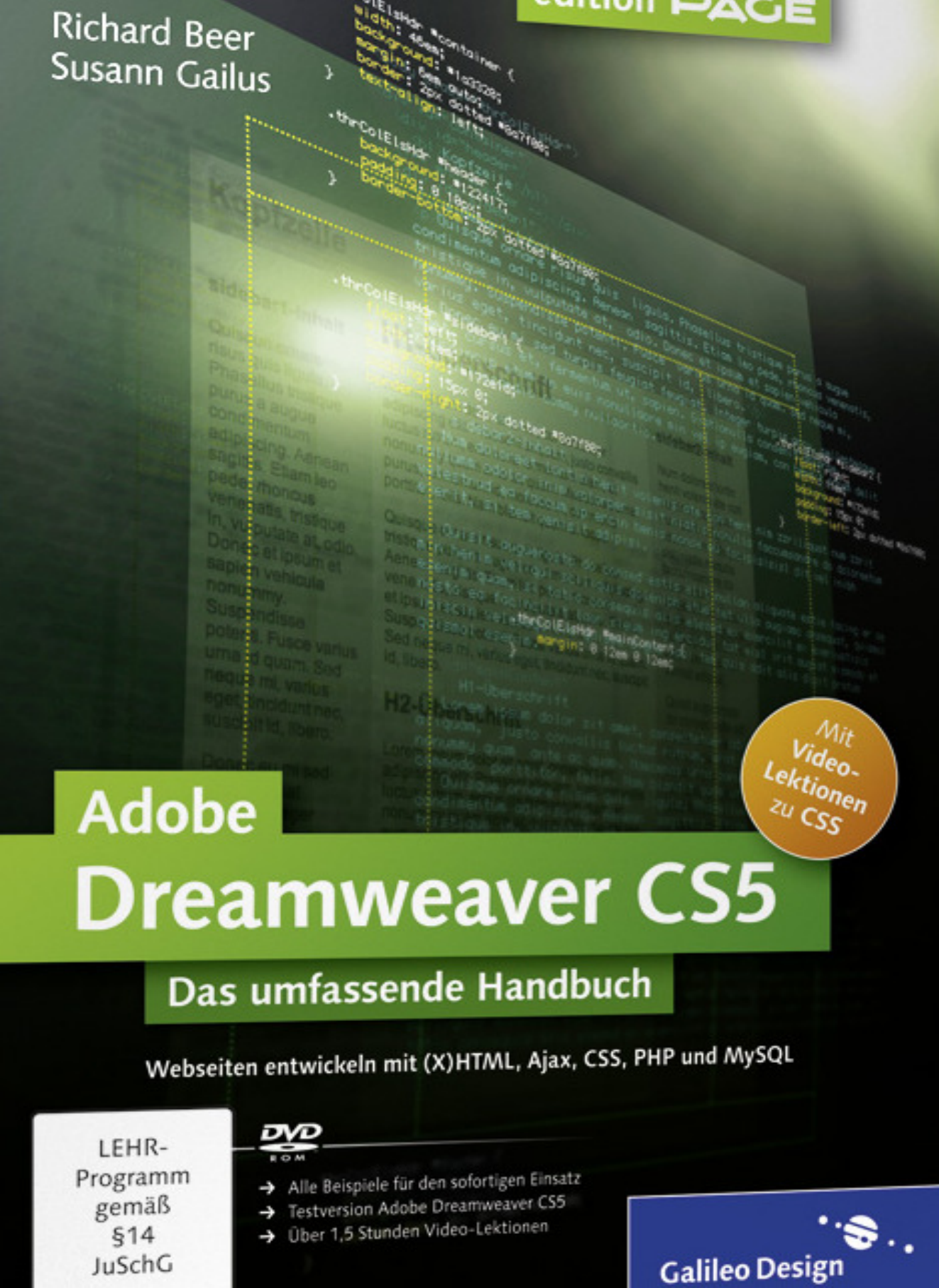

Richard Beer
Susann Gailus
Mit Video-Lektionen zu CSS
Adobe
Dreamweaver CS5
Das umfassende Handbuch
Webseiten entwickeln mit (X)HTML, Ajax, CSS, PHP und MySQL
LEHR-Programm gemäß §14 JuSchG
DVD ROM
→ Alle Beispiele für den sofortigen Einsatz
→ Testversion Adobe Dreamweaver CS5
→ Über 1,5 Stunden Video-Lektionen
Galileo Design

Richard Beer, Susann Gailus

Adobe Dreamweaver CS5

Das umfassende Handbuch

Galileo Press

Liebe Leserin, lieber Leser,

wer professionelle Websites erstellen möchte, muss eine Vielzahl an Techniken beherrschen und eine Menge Anforderungen und Möglichkeiten im Blick haben. Wie gut, dass Adobes Dreamweaver Sie dabei perfekt unterstützt und Hilfen anbietet, die Ihnen viel Zeit ersparen können. Wie Sie dieses mächtige Werkzeug effizient einsetzen, zeigen Ihnen Richard Beer und Susann Gailus in diesem umfassenden Handbuch.

In drei ausführlichen Teilen gehen sie auf alle Aspekte bei der Entwicklung von Websites mit Dreamweaver CS5 ein. Der erste Teil beschäftigt sich mit den Grundlagen von Dreamweaver und stellt das Beispielprojekt des Buchs vor, anhand dessen die Funktionen und Techniken erläutert werden. Im zweiten Teil erfahren Sie alles, was Sie zur Entwicklung von statischen Websites wissen müssen, und im dritten Teil dreht sich dann alles um dynamische Websites.

Dabei bleibt es ganz Ihnen überlassen, an welcher Stelle Sie im Buch einsteigen – es wurde so konzipiert, dass Sie es sowohl als Einstiegslektüre als auch als Nachschlagewerk verwenden können. Die vielen anschaulichen Beispiele und zahlreichen Praxistipps stellen dabei sicher, dass Sie jederzeit das Optimum aus Dreamweaver herausholen werden.

Dieses Buch wurde mit großer Sorgfalt geschrieben, geprüft und produziert. Sollte dennoch einmal etwas nicht so funktionieren, wie Sie es erwarten, freue ich mich, wenn Sie sich mit mir in Verbindung setzen. Ihre Kritik und konstruktiven Anregungen sind uns jederzeit herzlich willkommen!

Viel Spaß beim Erstellen Ihrer Websites wünscht Ihnen nun

Ihre Christine Siedle
Lektorat Galileo Design

christine.siedle@galileo-press.de
www.galileodesign.de
Galileo Press · Rheinwerkallee 4 · 53227 Bonn

Auf einen Blick

TEIL I Grundlagen

1 Einleitung 29
2 Projektablauf für die Buchwebsite 37
3 Grundlagen von Dreamweaver 45

TEIL II Statische Websites

4 Eine Site erstellen 93
5 Grundlegende Dokumenteinstellungen 127
6 Tabellen und Listen 139
7 Bilder im Web 153
8 Framesets 171
9 JavaScript und Verhalten 183
10 Hyperlinks 193
11 CSS in Dreamweaver 201
12 Spry – Framework für Ajax 287
13 Vorlagen und Bibliotheken 305
14 Formulare 317
15 Flash und YouTube integrieren 347
16 Office-Dateien einfügen 361
17 Quelltext de luxe 365

TEIL III Dynamische Websites

18 Dynamische Websites – Einführung 383
19 Einführung in PHP 397
20 PHP mit Dreamweaver 419
21 Lokalen Webserver installieren 429
22 Dynamische Sites in Dreamweaver einrichten 457
23 MySQL-Grundlagen 467
24 Datenbanken mit Dreamweaver anbinden 495
25 Datenbanken abfragen 501
26 Daten einfügen und dynamische Formulare 581
27 Benutzer authentifizieren und Zugriffsrechte festlegen 609
28 Fortgeschrittene Techniken 625
29 Menüs für dynamisches Verhalten 657
30 XML in Dreamweaver 665
31 Spry und XML 695

Der Name Galileo Press geht auf den italienischen Mathematiker und Philosophen Galileo Galilei (1564–1642) zurück. Er gilt als Gründungsfigur der neuzeitlichen Wissenschaft und wurde berühmt als Verfechter des modernen, heliozentrischen Weltbilds. Legendär ist sein Ausspruch *Eppur se muove* (Und sie bewegt sich doch). Das Emblem von Galileo Press ist der Jupiter, umkreist von den vier Galileischen Monden. Galilei entdeckte die nach ihm benannten Monde 1610.

Gerne stehen wir Ihnen mit Rat und Tat zur Seite:
christine.siedle@galileo-press.de bei Fragen und Anmerkungen zum Inhalt des Buches
service@galileo-press.de für versandkostenfreie Bestellungen und Reklamationen
julia.bruch@galileo-press.de für Rezensions- und Schulungsexemplare

Lektorat Christine Siedle
Korrektorat Marlis Appel
Cover Hannes Fuß, Berlin
Titelbild Hannes Fuß, Berlin
Typografie und Layout Vera Brauner
Herstellung Steffi Ehrentraut
Satz SatzPro, Krefeld
Druck und Bindung Bercker Graphischer Betrieb, Kevelaer

Dieses Buch wurde gesetzt aus der Linotype Syntax Serif (9,25/13,25 pt) in FrameMaker. Gedruckt wurde es auf chlorfrei gebleichtem Offsetpapier.

Beim Aufbau der Buchwebsite wurden Bilder folgender Fotografen verwendet:
mathias the dread/*photocase.com*
cw-design/*photocase.com*
.marqs/*photocase.com*
kallejipp/*photocase.com*

Bibliografische Information der Deutschen Nationalbibliothek
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über *http://dnb.d-nb.de* abrufbar.

ISBN 978-3-8362-1567-1

© Galileo Press, Bonn 2010
1. Auflage 2010

Das vorliegende Werk ist in all seinen Teilen urheberrechtlich geschützt. Alle Rechte vorbehalten, insbesondere das Recht der Übersetzung, des Vortrags, der Reproduktion, der Vervielfältigung auf fotomechanischem oder anderen Wegen und der Speicherung in elektronischen Medien. Ungeachtet der Sorgfalt, die auf die Erstellung von Text, Abbildungen und Programmen verwendet wurde, können weder Verlag noch Autor, Herausgeber oder Übersetzer für mögliche Fehler und deren Folgen eine juristische Verantwortung oder irgendeine Haftung übernehmen. Die in diesem Werk wiedergegebenen Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. können auch ohne besondere Kennzeichnung Marken sein und als solche den gesetzlichen Bestimmungen unterliegen.

Dieses Buch ist Birgit Gailus gewidmet.

Ohne dich wären wir nicht, wer wir sind.
Du lebst ewig in unseren Herzen.

Inhalt

Vorwort 25

TEIL I Grundlagen

1 Einleitung 29

1.1 Für die Leser 29
1.1.1 An wen richtet sich dieses Buch? 29
1.1.2 Welche Vorkenntnisse benötigen Sie? 29
1.1.3 Warum wir in diesem Buch auf dem PC arbeiten 30
1.2 Was ist Dreamweaver und was nicht? 31
1.2.1 Keine HTML-Kenntnisse erforderlich? 32
1.2.2 Anforderungen an einen professionellen HTML-Generator 32
1.2.3 Warum Dreamweaver? 33
1.2.4 Grenzen von Dreamweaver 34
1.2.5 Systemanforderungen von Dreamweaver CS5 34
1.3 Was ist neu in Dreamweaver CS5? 35
1.3.1 Veraltete Funktionen 36

2 Projektablauf für die Buchwebsite 37

2.1 Briefing 38
2.2 Arbeitsablauf 40
2.3 Das Layout 41
2.3.1 Entwurf in Photoshop 41
2.3.2 Entwurf in Illustrator 43
2.3.3 Alternative Programme 43

3 Grundlagen von Dreamweaver 45

3.1 Die Programmoberfläche 46
3.1.1 Verschiedene Anzeigemöglichkeiten 46
3.1.2 Anpassen der Registergruppen 49
3.1.3 Platz zum Arbeiten schaffen 49
3.1.4 Einfügeleiste anpassen 50
3.1.5 Eigene Einfügeleiste erstellen 51
3.1.6 Das Dateifenster 52
3.1.7 Verschiedene Dokumentansichten 52
3.1.8 Die Entwurfsansicht 54

3.1.9 Die Codeansicht ... 56
3.1.10 Testserverbetrieb und Live-Ansicht ... 57
3.1.11 Live-Code ... 58
3.1.12 Code einfrieren ... 60
3.1.13 Code-Navigator ... 60
3.1.14 Abhängige Dateien ... 61
3.1.15 Mit mehreren Dokumenten arbeiten ... 62
3.1.16 Codeteilung ... 62
3.2 Wichtige Voreinstellungen ... 63
3.2.1 Allgemeine Voreinstellungen ... 63
3.2.2 AP-Elemente (bis Version 8.0: Layer) ... 64
3.2.3 Codeformat einstellen ... 65
3.2.4 Codehinweise ... 66
3.2.5 Codeumschreibung ... 67
3.2.6 CSS-Stile ... 67
3.2.7 Dateien vergleichen ... 69
3.2.8 Dateitypen/Editoren ... 69
3.2.9 Eingabehilfen ... 70
3.2.10 Farbe für Code ... 71
3.2.11 Kopieren und Einfügen (Office) ... 71
3.2.12 Markierung ... 72
3.2.13 Neues Dokument ... 72
3.2.14 Site ... 75
3.2.15 Statusleiste ... 75
3.2.16 Unsichtbare Elemente ... 75
3.2.17 Validator ... 77
3.2.18 Vorschau im Browser ... 77
3.2.19 Prüfung auf externen Rechnern ... 80
3.3 Layouthilfen in Dreamweaver ... 80
3.3.1 Raster einstellen und anzeigen ... 80
3.3.2 Hilfslinien ... 81
3.3.3 Die Statusleiste ... 82
3.3.4 Tracing-Bilder ... 84
3.4 Dreamweaver erweitern ... 84
3.4.1 Befehle speichern ... 84
3.4.2 Extensions einsetzen ... 86
3.4.3 Bezugsquellen ... 86
3.4.4 Extension Manager ... 87
3.4.5 Ajax-Frameworks durch Extensions nutzen ... 88

TEIL II Statische Websites

4 Eine Site erstellen 93

4.1 Die Struktur 93
4.1.1 Pfadangaben 93
4.1.2 Ordnerstruktur 95
4.2 Die Site-Verwaltung 98
4.2.1 Regeln beim Arbeiten mit einer Site 98
4.2.2 Testserver 99
4.2.3 Site-Definition 100
4.2.4 Erweiterte Einstellungen 102
4.2.5 Server verwalten 104
4.2.6 Erweiterte Einstellungen 108
4.2.7 Versionskontrolle 109
4.2.8 Cloaking 111
4.2.9 Design Notes 112
4.2.10 Dateiansichtsspalten 114
4.2.11 Weitere Einstellungen 114
4.2.12 Fehlerquellen in der Site-Verwaltung 115
4.3 Die Site-Ansicht 115
4.3.1 Synchronisieren und Vergleichen von Dateien 116
4.3.2 Mit der Dateiverwaltung arbeiten 117
4.4 Gruppenarbeit 118
4.4.1 Remote-Zugriff einstellen 120
4.4.2 Mit der Gruppenfunktion arbeiten 122
4.5 Die Buchwebsite anlegen 123

5 Grundlegende Dokumenteinstellungen 127

5.1 Seiteneigenschaften festlegen mit CSS 127
5.1.1 Verknüpfungen & Überschriften 131
5.1.2 Titel/Codierung 132
5.1.3 Tracing-Bild – warum und wann? 133
5.2 Seiteneigenschaften festlegen ohne CSS 134
5.3 Metaangaben 136

6 Tabellen und Listen 139

6.1 Layouttabellen – pro und contra 139
6.2 Einfügen von Tabellen 140
6.2.1 Hilfsmittel für Tabellen 142

6.2.2 Auswählen und Verändern von Tabellen ... 143
6.2.3 Arbeiten im erweiterten Tabellenmodus ... 143
6.3 Tabelleneigenschaften einstellen ... 144
6.3.1 Zelleneigenschaften einstellen ... 146
6.3.2 Fehler bei Tabellenlayouts vermeiden ... 146
6.4 Verschachtelte Tabellen ... 148
6.4.1 Der Trick mit den transparenten GIFs ... 148
6.4.2 Tabellenumrandungen erstellen ... 150
6.5 Arbeiten mit Listen ... 150
6.5.1 Erstellen einer Liste ... 152

7 Bilder im Web ... 153

7.1 Bildformate bestimmen ... 153
7.1.1 JPEG ... 153
7.1.2 GIF ... 154
7.1.3 PNG ... 155
7.1.4 Flash ... 156
7.1.5 Grafikformate in der Übersicht ... 156
7.2 Bilder einfügen und bearbeiten ... 157
7.2.1 Positionieren per Drag & Drop ... 157
7.2.2 Das Bedienfeld »Elemente« ... 158
7.2.3 Bilder in Dreamweaver bearbeiten ... 161
7.2.4 Bild von Text umfließen lassen ... 163
7.2.5 Image Maps und Hotspots ... 163
7.2.6 Platzhalterbilder ... 165
7.3 Interaktion mit Photoshop ... 166

8 Framesets ... 171

8.1 Funktionsweise von Frames ... 171
8.1.1 Gestalten mit Framesets ... 171
8.1.2 Suchmaschinen und Framesets ... 173
8.2 Ein Frameset anlegen ... 174
8.3 Mit Framesets arbeiten ... 178
8.3.1 Framesets nachbearbeiten ... 179
8.3.2 Verlinkungen in Framesets ... 179
8.3.3 Mehrere Frames gleichzeitig neu laden ... 179
8.3.4 Eingebettete Frames – <iframe> ... 181

9 JavaScript und Verhalten ... 183

9.1 JavaScript in Dreamweaver ... 183
9.2 Verhalten ... 184
9.2.1 Rollover erstellen ... 185
9.2.2 Plugins überprüfen ... 188
9.2.3 Mehrere Frames gleichzeitig austauschen ... 189
9.2.4 JavaScript-Effekte ... 190
9.2.5 JavaScript und CSS ... 191

10 Hyperlinks ... 193

10.1 Verknüpfungen einbauen ... 193
10.2 Ziel angeben ... 195
10.3 Barrierefreie Hyperlinks ... 196
10.4 Hyperlinks prüfen ... 196
10.5 Anker hinzufügen ... 197
10.6 Hyperlink-Methoden ... 198
10.7 Hyperlinks auf E-Mail-Adressen ... 199
10.8 Platzhalter für Hyperlinks ... 200

11 CSS in Dreamweaver ... 201

11.1 CSS – pro und contra ... 201
11.2 Arten von CSS-Stilen ... 202
11.2.1 Inline-Stile ... 203
11.2.2 Interne CSS-Stile ... 204
11.2.3 Externe CSS-Stile ... 204
11.3 CSS-Syntax ... 208
11.3.1 Selektortypen ... 208
11.3.2 Tag-Selektoren ... 209
11.3.3 ID-Selektoren ... 209
11.3.4 Klassen-Selektoren ... 210
11.3.5 Pseudoklassen ... 211
11.3.6 Vererbung und Kaskadierung ... 213
11.4 Erstellen und Bearbeiten von CSS-Stilen ... 213
11.4.1 Das CSS-Bedienfeld ... 213
11.4.2 Neue CSS-Regeln erstellen ... 215
11.4.3 CSS-Regeln bearbeiten ... 216
11.4.4 CSS-Bearbeitung voreinstellen ... 217
11.5 CSS-Regeln für Texte ... 218
11.5.1 Zuweisen von CSS in der Eigenschaftenpalette ... 220

11.6 CSS-Regeln für unterschiedliche Ausgabemedien ... 221
11.6.1 Ausgabemedium Print und Screen ... 221
11.6.2 Stile für verschiedene Ausgabemedien anzeigen ... 223
11.7 Stile einschalten, wechseln und abschalten ... 224
11.8 CSS-Layout anlegen ... 224
11.8.1 CSS für barrierefreies Webdesign ... 225
11.8.2 Das CSS-Box-Modell ... 226
11.8.3 CSS-Positionierungen ... 231
11.8.4 CSS-Layout-Boxen in Dreamweaver ... 232
11.9 CSS-Stile und DIV-Tags ... 234
11.10 AP-Elemente ... 239
11.11 Navigation aus Listen erstellen ... 247
11.12 Mit CSS-Vorlagen arbeiten ... 251
11.13 Allgemeine Vorgehensweise ... 253
11.14 Layout-Hilfsmittel ... 254
11.15 CSS-Regeln für die Buchwebsite erstellen ... 260
11.16 Aufbau der Buchwebsite mit CSS und AP-Elementen ... 262

12 Spry – Framework für Ajax ... 287

12.1 Was ist Spry? ... 287
12.1.1 Vorteile ... 288
12.1.2 Einschränkungen ... 289
12.1.3 Fehlermeldung bei lokaler Anzeige ... 289
12.2 Spry in diesem Buch ... 290
12.3 Spry-Widgets ... 290
12.3.1 Spry-Dateien kopieren ... 291
12.4 Spry-Menüleiste ... 292
12.4.1 Spry-Menüleiste anlegen ... 292
12.4.2 Spry-Menüleiste anpassen ... 292
12.5 Spry-Palette mit Registerkarten ... 294
12.5.1 Spry-Registerkarten einstellen ... 295
12.5.2 Spry-Registerkarten füllen ... 295
12.6 Spry-Akkordeon ... 296
12.7 Reduzierbare Palette ... 298
12.8 Spry-QuickInfo ... 299
12.9 Spry-Widgets mit CSS gestalten ... 300

13 Vorlagen und Bibliotheken ... 305

13.1 Mit Vorlagen arbeiten ... 306
13.1.1 Vorlagen erstellen ... 306

13.1.2 Bearbeitbare Bereiche festlegen ... 308
13.1.3 Dokument aus Vorlage erstellen ... 309
13.1.4 Vorlagen entfernen ... 310
13.1.5 Vorlagen aktualisieren ... 310
13.2 Tricks mit Vorlagen ... 311
13.2.1 Verschachtelte Vorlagen ... 311
13.2.2 Wiederholte Bereiche ... 311
13.2.3 Wiederholte Tabelle ... 312
13.3 Mit Bibliotheken arbeiten ... 312

14 Formulare ... 317

14.1 Übertragungsmethoden für Formulardaten ... 317
14.1.1 GET ... 317
14.1.2 POST ... 318
14.1.3 Formulare per E-Mail ... 318
14.2 Ein Formular erstellen ... 318
14.3 Formularelemente ... 320
14.3.1 Einfügen mit Eingabehilfe ... 320
14.3.2 Namensvergabe ... 321
14.3.3 Textfelder ... 322
14.3.4 Versteckte Textfelder ... 323
14.3.5 Textbereiche ... 325
14.3.6 Kontrollkästchen ... 325
14.3.7 Optionsschalter ... 326
14.3.8 Auswahllisten und Sprungmenüs ... 326
14.3.9 Dateifelder ... 327
14.3.10 Schaltflächen ... 328
14.4 Überprüfung von Formularen ... 329
14.4.1 Spry für Formulare ... 329
14.4.2 Spry-Überprüfung Textfeld ... 330
14.4.3 Spry-Überprüfung Textbereich ... 335
14.4.4 Spry-Überprüfung Kontrollkästchen ... 336
14.4.5 Spry-Überprüfung Auswahl ... 337
14.4.6 Spry-Überprüfung Kennwort ... 338
14.4.7 Spry-Überprüfung Bestätigung ... 338
14.5 Formulare gestalten ... 339
14.5.1 Formularelemente positionieren ... 339
14.5.2 Formularelemente mit CSS gestalten ... 340
14.6 Ein Kontaktformular für die Buchwebsite ... 341

15 Flash und YouTube integrieren 347

15.1 Flash einsetzen oder nicht? 347
15.2 Flashfilme einfügen 348
15.2.1 Filme einbinden 348
15.2.2 Eigenschaften einstellen 350
15.3 Flash-Sound einbinden 353
15.4 Flash in Dreamweaver öffnen 354
15.5 FLV-Filme einbinden 354
15.6 YouTube-Videos einbinden 356

16 Office-Dateien einfügen 361

16.1 Word-Texte einfügen 361
16.2 Excel-Tabellen einfügen 362
16.3 CSV-Daten importieren 362
16.4 Weitere Einfügemöglichkeiten 363

17 Quelltext de luxe 365

17.1 Arbeiten im Quelltext 365
17.1.1 Quellcode formatieren 365
17.1.2 Codeansichtsoptionen 367
17.1.3 Code übersichtlich halten 368
17.1.4 Code halbautomatisch erstellen 369
17.1.5 Kommentare 370
17.2 Arbeiten in der Live-Code-Ansicht 370
17.3 Codefragmente (Snippets) einsetzen 371
17.3.1 Codefragmente mit Kollegen teilen 374
17.3.2 Tag-Inspektor 374
17.3.3 Eigene Tag-Vorgaben oder -Bibliotheken 374
17.4 Quelltext in der Entwurfsansicht bearbeiten 377
17.5 Quelltext automatisch optimieren 378
17.6 Suchen und ersetzen 379

TEIL III Dynamische Websites

18 Dynamische Websites – Einführung 383

18.1 Vorteile dynamischer Sites 383
18.1.1 Statische versus dynamische Sites 384
18.1.2 Dynamische Sites und Suchmaschinen 384
18.1.3 Dynamische Sites und Sessions 385

18.2 Typen dynamischer Websites ... 385
18.2.1 Web-Content-Management-Systeme (WCMS) ... 385
18.2.2 Content-Management-Systeme (CMS) ... 386
18.2.3 Redaktionssysteme ... 386
18.2.4 Shopsysteme ... 386
18.2.5 Foren ... 387
18.2.6 Portale ... 387
18.2.7 Personalisierte Websites ... 388
18.2.8 Weblogs ... 388
18.2.9 Social Networks ... 389
18.3 Client- und serverseitiges Skripting ... 390
18.4 Konzeption dynamischer Sites ... 393
18.4.1 Besonderheiten für das Layout ... 393
18.4.2 Trennung von Layout und Inhalt ... 394
18.4.3 Konzeption der Administrationsoberfläche ... 394
18.4.4 Planung, Planung, Planung … ... 395

19 Einführung in PHP ... 397

19.1 Was ist PHP? ... 397
19.1.1 »Hallo Welt« in PHP ... 398
19.1.2 Vorteile von PHP ... 400
19.2 PHP und XHTML ... 400
19.2.1 PHP in XHTML einbinden ... 400
19.2.2 Schreibweise von Zahlen und Zeichen ... 401
19.2.3 XHTML in PHP einbinden ... 402
19.3 Variablen in PHP ... 403
19.4 Vergleichsoperatoren ... 405
19.5 Boolesche Operatoren ... 406
19.6 Schleifen programmieren ... 406
19.6.1 for-Schleifen ... 407
19.6.2 while-Schleifen ... 408
19.6.3 do-while-Schleifen ... 408
19.6.4 foreach-Schleifen ... 409
19.7 Bedingungen mit PHP ... 409
19.7.1 if-Anweisung ... 410
19.7.2 if-else-Anweisung ... 410
19.7.3 switch-Anweisung ... 411
19.8 Dateien einbinden ... 411
19.8.1 Mit MySQL verbinden ... 413
19.8.2 MySQL-Datenbanken abfragen ... 414

19.9 Mit Sessions arbeiten ... 414
19.10 Funktionen ... 415
19.11 Fazit ... 417

20 PHP mit Dreamweaver ... 419

20.1 PHP-Befehle einsetzen ... 419
20.2 Variablen in Dreamweaver ... 420
20.2.1 Mit Variablen arbeiten ... 420
20.2.2 Variablen senden ... 422
20.3 PHP von Hand coden ... 424
20.4 PHP-Referenz ... 424
20.5 Fehlersuche im PHP-Code ... 426
20.5.1 Fehlermeldungen anzeigen ... 426
20.5.2 Externe Variablen anzeigen ... 426
20.5.3 Häufige PHP-Fehlerquellen ... 427

21 Lokalen Webserver installieren ... 429

21.1 Webserver-Grundlagen ... 429
21.2 Vorbereitung: Firewall einstellen oder ausschalten ... 431
21.3 WAMP installieren ... 432
21.3.1 XAMPP testen und konfigurieren ... 435
21.3.2 Eigene Documentroot angeben ... 437
21.3.3 phpMyAdmin in die Documentroot legen ... 439
21.3.4 MySQL ... 440
21.4 Internet Information Server installieren ... 441
21.4.1 IIS verwalten ... 443
21.4.2 PHP auf IIS installieren ... 444
21.4.3 MySQL unter IIS installieren ... 449
21.4.4 phpMyAdmin unter IIS installieren ... 449
21.4.5 Installation des IIS 7 und PHP unter Windows 7 Business ... 450

22 Dynamische Sites in Dreamweaver einrichten ... 457

22.1 Testserver-Einstellungen ... 457
22.2 Site-Einstellungen testen ... 460
22.3 Buchwebsite für dynamische Inhalte einrichten ... 464

23 MySQL-Grundlagen ... 467

23.1 Einführung in Datenbanken ... 467
23.1.1 Relationale Datenbanken ... 468

23.1.2 Relationstypen ... 470
23.1.3 Primärschlüssel ... 470
23.1.4 Redundanzfreiheit ... 471
23.1.5 Prozessdatenfreiheit ... 471
23.1.6 Fremdschlüssel ... 471
23.1.7 Referenzielle Integrität ... 471
23.1.8 Endlosschleifen ... 472
23.1.9 MySQL-Datentypen ... 472
23.2 Erstellen einer MySQL-Datenbank ... 474
23.2.1 MySQL-Datenbank mit phpMyAdmin anlegen ... 474
23.2.2 Einfügen von Datensätzen ... 478
23.2.3 Import von Daten ... 479
23.2.4 Export von Daten zum Provider ... 480
23.3 Benutzerverwaltung mit MySQL ... 481
23.4 MySQL und Microsoft Access ... 483
23.5 Einführung in SQL ... 487
23.5.1 Daten abfragen mit »SELECT« ... 488
23.5.2 Datensatz abfragen mit »WHERE« ... 489
23.5.3 Daten sortieren mit »ORDER BY« ... 489
23.5.4 Datensatz aktualisieren mit »UPDATE« ... 489
23.5.5 Datensatz löschen mit »DELETE« ... 489
23.5.6 Datensatz einfügen mit »INSERT INTO« ... 490
23.6 Datenstruktur der Buchwebsite ... 490
23.6.1 Anforderungen ... 490
23.6.2 Primärschlüssel festlegen ... 491

24 Datenbanken mit Dreamweaver anbinden ... 495

24.1 MySQL-Datenbanken konnektieren ... 495
24.2 MySQL mit PHP konnektieren ... 496
24.2.1 Bedienfeldgruppe »Anwendung« ... 496
24.2.2 Das Bedienfeld »Datenbanken« ... 499

25 Datenbanken abfragen ... 501

25.1 Datenbankabfrage und Datenanzeige ... 501
25.2 Datensätze in Dreamweaver abfragen ... 502
25.2.1 Einfache Abfragen erstellen ... 503
25.2.2 Anzeigen von Datensätzen ... 508
25.2.3 Erweiterte Abfragen erstellen ... 509
25.2.4 Mehrere Datensätze ausgeben ... 512
25.2.5 Bestimmte Datensätze abfragen ... 515

25.2.6 Datensatz-ID mit URL übergeben ... 517
25.2.7 Einstellbare Variablentypen ... 521
25.2.8 Testen der Abfragen ... 522
25.3 Dynamische Tabellen ... 522
25.3.1 Erstellen einer dynamischen Tabelle ... 522
25.3.2 Dynamische Tabelle mit Serververhalten ... 524
25.4 Wiederholte Bereiche mit AP-Elementen ... 524
25.5 Bilder dynamisch einfügen ... 529
25.5.1 Vorbereitungen für dynamisch eingefügte Bilder ... 529
25.5.2 Bilder aus Datenbanken verlinken ... 530
25.6 In Datensätzen navigieren ... 534
25.6.1 Datensatznavigation erstellen ... 534
25.6.2 Bereiche in Abhängigkeit von Daten anzeigen ... 536
25.6.3 Datensatzanzahl anzeigen ... 539
25.6.4 Individuelle Datensatznavigationen erstellen ... 539
25.7 HTML-Attribute dynamisch generieren ... 540
25.7.1 Eigenschaften aus Datenquellen einfügen ... 540
25.7.2 Daten in der URL mit Hyperlinks übergeben ... 545
25.8 Automatische Master- und Detailseiten ... 548
25.9 Ausgabe formatieren ... 550
25.10 Aufbau der dynamischen Inhalte der Buchwebsite ... 551
25.10.1 Das Funktionsschema ... 551
25.10.2 Die Masterseite ... 552
25.10.3 Die Detailseiten ... 554
25.10.4 Website zu Twitter und anderen Social Networks verlinken ... 577

26 Daten einfügen und dynamische Formulare ... 581

26.1 Neue Datensätze einfügen ... 582
26.1.1 Der Assistent für Einfügeformulare ... 582
26.1.2 Eigene Einfügeformulare verwenden ... 584
26.2 Bestehende Datensätze manipulieren ... 585
26.2.1 Ein Änderungsformular von Hand erstellen ... 585
26.2.2 Der Assistent für Änderungsformulare ... 586
26.2.3 Datensatznavigation für Datensatzänderungen ... 589
26.2.4 Datensatznavigation mit zwei Dokumenten und Auswahlseiten ... 590
26.3 Datensätze löschen ... 592
26.3.1 Dynamische Auswahlliste erstellen ... 593
26.3.2 Dynamische Kontrollkästchen (Checkboxen) ... 594
26.4 Daten aus mehreren Tabellen verbinden ... 595

26.5 Administrationsoberfläche für die Buchwebsite anlegen 599
26.5.1 Frameset anlegen 600
26.5.2 Auswahlseiten 600
26.5.3 CSS für Administrationsebene anlegen 602

27 Benutzer authentifizieren und Zugriffsrechte festlegen 609

27.1 Benutzer verwalten 609
27.1.1 Schema einer Benutzerauthentifizierung 609
27.1.2 Tabelle für die Benutzerverwaltung 609
27.1.3 Benutzer anmelden 611
27.2 Zugriffsrechte beschränken 613
27.3 Neue Benutzer anlegen 615
27.4 Zugriffsrechte für die Website zum Buch 620

28 Fortgeschrittene Techniken 625

28.1 SQL-Abfragen 625
28.1.1 Eigene SQL-Abfragen einbinden 625
28.1.2 Externe Tools verwenden 631
28.1.3 Abfragen verschachteln 632
28.1.4 Eine Volltextsuche 639
28.2 Arbeiten mit Sessions 641
28.3 Serververhalten 644
28.3.1 Eigene Serververhalten schreiben 644
28.3.2 Positionierung im Code 651
28.3.3 Bedingungen als Serververhalten anlegen 653

29 Menüs für dynamisches Verhalten 657

29.1 Tabellendaten importieren 657
29.2 Spry-XML-Datensatz 657
29.3 Spry-Bereich 657
29.4 Spry-Wiederholung 657
29.5 Spry-Wiederholungsliste 658
29.6 Datensatzgruppen 658
29.7 Dynamische Daten 658
29.8 Bereich wiederholen 659
29.9 Bereich anzeigen 659
29.10 Seitenerstellung für Datensatzgruppe 660
29.11 Datensatzgruppen-Navigationsstatus 661
29.12 Master-Detaildatensatz 661

29.13 Datensatz einfügen ... 661
29.14 Datensatz aktualisieren ... 662
29.15 Datensatz löschen ... 662
29.16 Benutzerauthentifizierung ... 663
29.17 XSL-Transformation ... 663

30 XML in Dreamweaver ... 665

30.1 Einführung in XML ... 665
30.2 XML und CSS ... 668
30.3 XML-Attribute ... 671
30.4 DTDs ... 673
30.4.1 DTDs mit Dreamweaver entwickeln ... 675
30.4.2 Externe DTDs ... 676
30.4.3 Einbinden von DTDs in Dreamweaver ... 677
30.5 XML, XSLT und CSS ... 680
30.5.1 XSLT-Einführung ... 680
30.5.2 RSS-Feed einbinden ... 681
30.5.3 XSLT-Fragmente und ganze XSLT-Seiten ... 682
30.5.4 Clientseitige XSLT-Transformation ... 682
30.5.5 Serverseitige XSLT-Transformation mit PHP ... 691

31 Spry und XML ... 695

31.1 Was ist Ajax? ... 695
31.2 XML-Daten anbinden ... 695
31.3 Spry-Bereiche ... 701
31.3.1 Spry-Wiederholungen ... 703
31.3.2 Spry-Wiederholungsliste ... 704
31.3.3 Spry-Tabelle ... 705
31.4 Kritik an Spry ... 708

Anhang

A Die DVD zum Buch ... 711

Index ... 715

Workshops

Eine Site erstellen
- Eine Site anlegen 124

CSS in Dreamweaver
- So funktioniert es mit CSS 260
- CSS und AP-Elemente 262

Formulare
- Ein Kontaktformular erstellen 342

Lokalen Webserver installieren
- Eigenes Rootverzeichnis angeben 438
- IIS unter Windows XP oder Windows 2000 installieren 442
- PHP auf IIS installieren 444
- IIS unter Windows 7 installieren 450
- PHP unter Windows 7 Business installieren 452

Dynamische Sites in Dreamweaver einrichten
- Website für dynamische Inhalte einrichten 464

MySQL-Grundlagen
- MySQL-Tabellen an Microsoft Access 2007 (oder andere Versionen) binden 484
- Die Datenbankstruktur importieren 492

Datenbanken mit Dreamweaver anbinden
- Datenbank an die Buchwebsite anbinden 499

Datenbanken abfragen
- Grafische Auswertung einer Datentabelle mit PHP 542
- Wiederholte Bereiche anlegen 556
- Abfrage anlegen und wiederholen 559
- URL-Parameter übergeben 565
- Eine Navigation aus einem Datensatz erstellen 566
- Aufbau der Detailseite 571

Daten einfügen und dynamische Formulare

- Datensätze über eine Administrationsoberfläche ändern 590
- Aufbau der Administrationsebene 603

Benutzer authentifizieren und Zugriffsrechte festlegen

- Ein Anmeldeformular für neue Benutzer erstellen 615
- Eine Datentabelle schützen 620

Fortgeschrittene Techniken

- SQL-Abfragen einbinden 628
- Datenbank-Konnektierung als Serververhalten speichern 644

XML in Dreamweaver

- Eine DTD importieren 677

Video-Lektionen auf der Buch-DVD



Als Ergänzung zum Buch möchten wir Ihnen ausgewählte Lehrfilme zum Thema CSS aus dem Video-Training »Adobe Dreamweaver CS5 – Das umfassende Training« von Tilo Rust (ISBN 978-3-8362-1571-8) zeigen.

CSS mit Dreamweaver ..... [01:49 Std.]

- Einleitung ..... [00:20 Min.]
- CSS richtig verstehen ..... [04:29 Min.]
- HTML-Elemente gestalten ..... [11:23 Min.]
- Externe CSS-Dateien nutzen ..... [09:04 Min.]
- CSS nachträglich einbinden ..... [03:06 Min.]
- Schriftformate in CSS ..... [14:24 Min.]
- Absätze gestalten mit Span ..... [10:55 Min.]
- Abstände genau definieren ..... [05:46 Min.]
- Absolute Positionierung ..... [14:41 Min.]
- Flexible Layouts aufbauen ..... [11:45 Min.]
- Hyperlinks auszeichnen ..... [05:03 Min.]
- Inhalte bereitstellen ..... [02:31 Min.]
- Troubleshooting für CSS ..... [15:41 Min.]

»Ein guter Handwerker kann auch mit miserablen Werkzeugen gute Arbeit leisten. Ein schlechter Handwerker nicht. Ein guter Handwerker wird sich aber ein schlechtes Werkzeug nicht antun.«

Vorwort

In den letzten Jahren hat sich technisch viel getan. CSS ist immer stärker in den Vordergrund gerückt. Ajax ist neu hinzugekommen, es gibt neue Browser und vieles mehr. Wir gehen daher in dieser Neuauflage unseres Handbuchs zu Dreamweaver CS5 auf die neuen Technologien ein und zeigen Ihnen, wie sie mit Dreamweaver sinnvoll zu nutzen sind, um Websites zu bauen, mit denen Sie Ihre Kunden begeistern.

Wie auch schon in den letzten Auflagen sind im Aufbau des Buches die Übungen vom erklärenden Teil strikt getrennt. Neu ist der Aufbau der Übungswebsite mit CSS statt im klassischen Tabellenlayout, das inzwischen nicht mehr als zeitgemäß gilt.



Ein eigens eingeführtes Baustellen-Icon weist Sie auf die Stellen im Buch hin, an denen der Aufbau der Buchwebsite beschrieben wird.

Das DVD-Icon zeigt Ihnen, wann Sie Inhalte von der Buch-DVD benötigen, um die Schritt-für-Schritt-Anleitungen nachvollziehen zu können. Vielen Dank an dieser Stelle an die Leser der Vorauflagen für ihr Feedback. **[◉]**

Die Schritt-für-Schritt-Anleitungen sind mit einem kleinen Treppen-Symbol markiert. Dort finden Sie Workshops zu konkreten Aufgabenstellungen. **[▞]**

Das Pfeil-Logo weist Sie auf weiterführende Information und Profitipps hin. **[«]**

Warnungen vor typischen Fehlerquellen sind mit einem Ausrufezeichen-Symbol gekennzeichnet. **[!]**

Nun gibt es natürlich viele Bücher über Dreamweaver und noch mehr über dynamische Websites. Die meisten hören jedoch entweder da auf, wo Dynamik mit PHP, ASP oder JSP anfängt, oder sie überfordern durch ein zu hohes Einstiegsniveau. Das vorliegende Buch soll diese Lücke schließen.

Wir zeigen Ihnen, wie sich Dreamweaver in die tägliche Arbeit integrieren lässt und wie es mit anderen Tools (Grafikprogrammen, Datenbanken, Flash usw.) interagiert. Besonderes Augenmerk haben wir auf die tatsächliche Praxisrelevanz gelegt. Durch die parallele Arbeit an einem realistischen Projekt, wie wir es täglich in unserer Agenturpraxis erleben, zeigen wir Schritt für Schritt den Aufbau einer statischen Website und deren anschließenden Ausbau in eine dynamische Website mit Datenbankanbindung.

Wenn es verschiedene Lösungsansätze gibt, erläutern wir diese und zeigen Ihnen die Vor- und Nachteile der Varianten auf. Alles Erlernte wird sogleich anhand praktischer Übungen verdeutlicht. Wenn verschiedene Browser oder Plattformen eine Rolle spielen, gehen wir detailliert darauf ein, und Sie lernen unterschiedliche Lösungswege kennen. Am Schluss des Buches sind Sie in der Lage, Websites mit Datenbankanbindungen in einer grafischen Ansicht zu erstellen. Als Programmierer werden Sie über die Arbeitserleichterungen erstaunt sein, die Ihnen Dreamweaver zu bieten hat.

Da wir leider nicht jede Technik detailliert beschreiben können, müssen wir einige Grundlagen voraussetzen: Ihnen sollten XHTML, HTML und CSS zumindest so weit vertraut sein, dass Sie notfalls in den Quellcode eingreifen können, um »nachzuhelfen«. Ebenso sollten Sie mit dem Dateisystem eines Computers umgehen können.

Wie man ein professionelles Layout für eine Website entwirft, wird ebenfalls nicht Thema dieses Buches sein. Dies wird in anderer Fachliteratur ausführlich behandelt.

Wir wünschen Ihnen mit dem nützlichen und neuen Wissen aus diesem Buch viel Freude beim Lesen und Ausprobieren.

Richard Beer und *Susann Gailus*

TEIL I
Grundlagen



Grafik-, Web- und Mediendesigner sehen sich in ihrer täglichen Arbeit zunehmend mit den Anforderungen auch technisch anspruchsvoller Websites konfrontiert. Diesen Anforderungen wird Dreamweaver gerecht.

1 Einleitung

In diesem Kapitel erfahren Sie, welche Vorraussetzungen Sie mitbringen sollten, um dieses Buch gewinnbringend zu lesen, für welche Anwendungszwecke sich Dreamweaver eignet und für welche eher nicht und welche Neuerungen Dreamweaver CS5 mit sich bringt.

1.1 Für die Leser

1.1.1 An wen richtet sich dieses Buch?

Dieses Buch wendet sich an folgende Zielgruppen:

- Grafikdesigner, Webdesigner und Mediendesigner, die sich verstärkt in ihrer täglichen Arbeit mit dynamischen Websites konfrontiert sehen und einen Einstieg in Dreamweaver finden wollen
- Programmierer, die sich die grafischen Features in Kombination mit eigenen Codefragmenten zu Nutze machen wollen
- Lehrende und Lernende der Medienberufe, da hier eine umfassende Darstellung mit Praxisbezug erfolgt
- Webdeveloper (Datenbankanbindungen usw.)
- Bildungseinrichtungen
- Umsteiger von älteren Versionen auf Dreamweaver CS5
- Umsteiger von anderen Webeditoren auf Dreamweaver CS5

1.1.2 Welche Vorkenntnisse benötigen Sie?

Wir können in diesem Buch leider nicht umfassend auf alle verwendeten Webtechnologien eingehen. Daher setzen wir einige Grundlagen bei Ihnen voraus, um mit Dreamweaver erfolgreich arbeiten zu können.

Zu Beginn eines jeden Kapitels erläutern wir die verwendeten Technologien zumindest so weit, dass Sie in der Lage sein sollten, sich bei Bedarf anhand der angegebenen Referenzen weiteres Wissen aneignen zu können.

Wenn folgende Aussagen zutreffen, steht einer erfolgreichen Lektüre nichts im Wege:

- Sie wissen, wie HTML bzw. XHTML funktioniert, und sind in der Lage, sich fehlende Informationen aus Referenzwerken zu beschaffen.
- Das trifft auch auf CSS und JavaScript zu.
- Sie wissen in etwa, wie das Internet grundsätzlich funktioniert, also was Webserver, Browser usw. sind.
- Sie sind in der Lage, mit einem Bildbearbeitungsprogramm Screens zu erstellen und diese zu slicen (zerteilen).
- Für den dynamischen Teil ist es sinnvoll, wenn Sie zumindest PHP- und MySQL-Grundlagen haben.
- Grundlegende Konzeptions- und Designtechniken für Websites sind Ihnen vertraut.
- Sie sind ein erfahrener PC-/Mac-Benutzer. Dies ist besonders im zweiten Teil des Buches sehr wichtig.

Verzeichnisstrukturen

Immer wieder fällt uns in Seminaren auf, dass Benutzer zwar bestens mit Software umgehen können, nicht jedoch mit dem Dateisystem selbst. Um erfolgreich mit Websites arbeiten zu können, ist es wichtig, dass Sie sich damit auseinandersetzen. Benutzer mit »alten« MS-DOS-Kenntnissen sind hier im Vorteil, da sich die Schreibweise in HTML und in MS-DOS sehr ähnelt.

1.1.3 Warum wir in diesem Buch auf dem PC arbeiten

Wir möchten hier keine alten Diskussionen über das richtige oder falsche Betriebssystem entfachen. Unserer Entscheidung für den PC liegen Fakten über unsere Zielgruppe und technische Anforderungen des Mediums Internet zugrunde. Wäre unser Arbeitsbereich das Erstellen von Druckerzeugnissen, würde die Entscheidung eventuell anders ausfallen. Nicht zuletzt spielen aber natürlich auch persönliche Vorlieben eine Rolle.

Laut aktuellen Statistiken (siehe Abbildung 1.1) sind bundesweit ca. 93 Prozent der Internetnutzer PC-Anwender. Eines der größten und nervigsten Probleme beim Erstellen von Internetseiten sind verschiedene Browserversionen und unterschiedliche Bildschirmdarstellungen auf den verschiedenen Betriebssystemen.

Es ist daher kaum sinnvoll, auf einer Plattform zu entwickeln, die andere Bildschirmdarstellungen und Konventionen erfordert als 95 Prozent des Markts, den es zu erreichen gilt. Statistisch gesehen spielt Mac OS kaum eine Rolle, wenn auch die Wahrnehmung in Agenturen häufig ein andere ist.

Seit der Version CS3 ist Dreamweaver übrigens auf Windows- und Mac-Systemen so ähnlich – sowohl in der Gestaltung der Benutzeroberfläche als auch in der Bedienung –, dass die Unterschiede nur noch marginal (vor allem in Bezug auf die Tastaturkürzel) sind.

Die Summe der Gründe führte zu dem Entschluss, die Website in diesem Buch auf einem PC-System zu entwickeln. Für Mac-User ist dies aber kein Hindernis. Sie können alle Inhalte auf dem Mac nachvollziehen und auch dort einen Webserver betreiben. Die Übungen im Buch können selbstverständlich – soweit Sie über ein von Adobe unterstütztes System verfügen – auch auf dem Mac erfolgen. Auf Unterschiede zwischen den Systemen weisen wir gegebenenfalls hin. Besonders, wenn Sie bereits Windows 7 einsetzen, sind die Unterschiede in der Benutzung aber kaum noch relevant.

Stand: 25.02.2010

Betriebssysteme	Anteil
Windows	93,3 %
OS X	4,5 %
Linux	1,1 %
iPhone OS	0,6 %
Div. Mobiltelefon	0,1 %
iPod	0,09 %
Symbian	0,09 %
Playstation	0,06 %
Android	0,04 %

Abbildung 1.1 Statistik zur Internetnutzung im Jahr 2010 (Quelle: www.webmasterpro.de)

1.2 Was ist Dreamweaver und was nicht?

Dreamweaver und andere ähnliche Programme sind sogenannte WYSIWYG-Editoren oder – wie es eigentlich heißt – HTML-Generatoren. Lesen wir die Werbebotschaften der Hersteller, könnte man leicht glauben, dass man mit modernen WYSIWYG-Editoren Websites ähnlich einfach wie das Layout eines Flyers erstellen kann. WYSIWYG steht für *What You See Is What You Get*. Das bedeutet lediglich, dass uns diese Programme eine grafische Oberfläche anbieten, um Quellcode für diverse Skriptsprachen zu erstellen, damit nicht jeder Code von Hand geschrieben werden muss.

1.2.1 Keine HTML-Kenntnisse erforderlich?

Wenn Sie aus dem Printbereich kommen, wissen Sie, dass ein gutes Druckprodukt eine gehörige Portion Basiswissen über die Drucktechniken voraussetzt. Nur mit technischem Hintergrundwissen ist es möglich, ein qualitativ hochwertiges Produkt zu schaffen.

Um professionelle Websites zu erstellen, müssen Sie heute dank dieser WYSIWYG-Editoren tatsächlich nicht mehr jeden HTML-Befehl mit all seinen Attributen kennen. Doch spätestens dann, wenn die Ergebnisse im Browser nicht dem entsprechen, was Sie in der Layoutansicht eines dieser Programme sehen, müssen Sie in der Lage sein, einzugreifen und eventuell von Hand nachzuhelfen.

Diese Vorgehensweise wird heute umso wichtiger, als immer mehr Seiten interaktiv mit Hilfe von JavaScript (z. B. Spry) aufgebaut oder ergänzt werden und diese Veränderungen oft nicht mehr in der Layoutansicht nachvollzogen oder gar verändert werden können.

1.2.2 Anforderungen an einen professionellen HTML-Generator

Ein HTML-Generator muss folgende Voraussetzungen erfüllen, um für eine professionelle Erstellung von Websites geeignet zu sein:

- Importierter Quellcode darf unter keinen Umständen verändert werden.
- Es muss eine Möglichkeit vorhanden sein, verschiedene Zielbrowser einzustellen und dafür zu optimieren.
- Der Quellcode muss den allgemeinen W3C-Konventionen entsprechen und möglichst über eine farbliche Syntaxhervorhebung in der Codeansicht verfügen.
- Ajax-Frameworks (z. B. Spry oder YUI) müssen nutzbar und implementierbar sein.
- Site-Management-Funktionen müssen vorhanden sein.
- In den Quellcode muss man manuell eingreifen können.
- Der erstellte Quelltext muss kompakt und funktionsfähig sein.
- Umfangreiche Websites erstellt man meist in einem Team aus mehreren Programmierern und Webdesignern. Aus diesem Grund ist eine Gruppenverwaltung sinnvoll, die den Produktionsprozess koordiniert.
- Und zu guter Letzt: Trotz allen Umfangs muss man auch noch mit der Software arbeiten können, ohne drei Monitore auf dem Tisch haben und seine eigenen Grafiken auf dem Bildschirm suchen zu müssen.

1.2.3 Warum Dreamweaver?

Ganz einfach – Dreamweaver von Adobe bietet Ihnen alle diese Möglichkeiten:

- Der von Dreamweaver erstellte Quelltext ist kompakt und durch die Voreinstellungen zu beeinflussen.
- Es lassen sich beliebige Zielbrowser konfigurieren und testen.
- Die von Dreamweaver erstellten JavaScripts sind (auch wenn es andere Stimmen geben mag) größtenteils äußerst kompakt und funktionieren – wenn auch der Versuch, den Quelltext nachzuvollziehen, manchmal Knoten im Gehirn hinterlässt. Gleiches gilt für die mitgelieferten Codes des Spry-Frameworks.
- Dreamweaver ermöglicht sehr viel Dynamik und eine Datenbankanbindung. An dieser Stelle kommt man jedoch um das Codieren von Hand kaum herum. Die vorliegende Dreamweaver-Version ist ganz besonders auf das Arbeiten mit Quelltexten hin optimiert. Besonders der neue Code-Navigator und die Live-Code-Ansicht sind unschlagbare Werkzeuge.
- Die Unterstützung moderner Technologien, wie XML, XHMTL und ASP.NET, wird immer wichtiger. Bei diesen Technologien hat Dreamweaver deutlich die Nase vorn.
- Dreamweaver bietet die Möglichkeit, eigene Codefragmente dauerhaft zu hinterlegen. Das Programm wächst daher mit Ihrer Erfahrung und Ihren Projekten.
- Das Programm ist mit Hilfe des Extension Managers erweiterbar. Es gibt unzählige, größtenteils kostenlose Erweiterungen für Dreamweaver.
- Die Interaktion mit anderen Programmen von Adobe ist sehr gut.
- Trotz seiner wirklich beeindruckenden Vielfalt bleibt das Programm benutzbar. Allerdings ist einige Einarbeitungszeit nötig, zumal sich die Oberfläche doch etwas von typischen Windows-Anwendungen unterscheidet.

Wir meinen, dass es zurzeit keinen HTML-Generator gibt, der Dreamweaver das Wasser reichen kann. Viele haben wir getestet und die meisten nach kurzer Zeit wieder verworfen. Es ist teilweise erstaunlich, was auf dem Markt und unter wirklich gutem Namen angeboten wird. Die Ergebnisse davon sind vor allem dann zu bestaunen, wenn Sie Websites mit verschiedenen Browsern testen.

Es kursiert die Meinung, Dreamweaver sei zu sehr auf Flash und andere Adobe-Produkte wie ColdFusion ausgerichtet. Wir können diese Meinung nicht bestätigen, zumal eine Integration in die Suiten und das Zusammenspiel mit Photoshop gerade bei großen Projekten einen erheblichen Arbeitsvorteil bringt. Im Alltag von Agenturen, die häufig die kompletten Suiten einsetzen, fügt sich Dreamweaver nahtlos in die Produktionskette ein.

Die Arbeitserleichterung und Produktivitätssteigerung mit Dreamweaver ist enorm, wenn man das Programm richtig einsetzt. Das gilt auch für die Programmierer, die sich immer wieder gerne gegen Tools mit grafischen Oberflächen sträuben.

1.2.4 Grenzen von Dreamweaver

Trotz aller Vorteile kann uns Dreamweaver die Arbeit nur wesentlich erleichtern, aber nicht abnehmen. Stimmt ein Layout von der ganzen Konzeption her nicht, kann das auch mit Dreamweaver nicht behoben werden. Wir können hier leider nicht auf die Grundlagen des Screendesigns eingehen, bei Galileo Press gibt es jedoch sehr gute Bücher zu diesem Thema, die wir Ihnen empfehlen können.

Letztlich ist die technische Umsetzung einer Website (bei dynamischen Websites verhält es sich etwas anders) der geringste Teil des gesamten Projekts. Der Hauptteil der Arbeit findet im Design- und Konzeptionsentwurf statt.

Nicht ganz das ideale Tool ist Dreamweaver für fortgeschrittene PHP-Entwickler. Hier fehlen einige wichtige Funktionen, um wirklich effektiv arbeiten zu können. Einen Debugger für PHP sucht man vergeblich. Auch die Beschränkung der Servertechnologie auf die Kombination PHP-MySQL ist sehr mager. In der aktuellen Version werden erstmals eigene Funktionen und Objektorientierung unterstützt. Das ist ein guter Schritt in die richtige Richtung. Außerdem: Die Version CS5 unterstützt Entwickler in der JavaScript-Programmierung (Ajax) nun erheblich besser als die Vorgängerversionen und bietet hierfür extra Werkzeuge an.

1.2.5 Systemanforderungen von Dreamweaver CS5

Windows

- Intel® Pentium® 4 oder AMD Athlon® 64
- Microsoft® Windows XP mit Service Pack 2 (Service Pack 3 empfohlen), Windows Vista® Home Premium, Business, Ultimate oder Enterprise mit Service Pack 1 oder Windows 7
- 512 MB RAM
- 1 GB freier Festplattenspeicher; zusätzlicher Speicher während der Installation erforderlich (keine Installation auf portablen Flash-Speichermedien möglich)
- 1.280 × 800 Pixel Monitorauflösung mit 16-Bit-Grafikkarte
- DVD-ROM-Laufwerk
- Breitband-Internetverbindung erforderlich für Onlinedienste

Mac OS

- Intel® Multi-Core-Prozessor
- Mac OS X Version 10.5.7 oder 10.6
- 512 MB RAM
- 1,8 GB freier Festplattenspeicher; zusätzlicher Speicher für die Installation erforderlich (keine Installation auf Dateisystemen, bei denen die Groß- und Kleinschreibung beachtet werden muss, oder portablen Flash-Speichermedien möglich)
- 1.280 × 800 Pixel Monitorauflösung mit 16-Bit-Grafikkarte
- DVD-ROM-Laufwerk
- Breitband-Internetverbindung erforderlich für Onlinedienste

(Quelle: www.adobe.com, Juli 2010)

Wir empfehlen Ihnen allerdings deutlich leistungsfähigere Systeme. Vor allem der Arbeitsspeicher sollte sehr großzügig ausgelegt werden (mindestens 1 GByte, unter Windows 7 mindestens 2 GByte). Auch mit der beschriebenen Monitorauflösung lässt sich mit Dreamweaver kaum arbeiten.

1.3 Was ist neu in Dreamweaver CS5?

Hier finden Sie die wichtigsten Neuerungen auf einen Blick:

- In unseren Augen *die* herausragende Neuerung in Dreamweaver CS5 ist die deutliche Verbesserung beim Erstellen eigener PHP-Programmierungen. Es werden erstmals eigene Funktionen, Klassen und Variablen erkannt und unterstützt.
- Bei Syntaxfehlern erscheint während der Eingabe eine neue Statusanzeige.
- Die Site-Einrichtung wurde komplett überarbeitet. Die Trennung zwischen Remote und Testserver entfällt. Es können nun auch mehrere Testserver gleichzeitig angegeben und je nach Bedarf umgeschaltet werden.
- Die CSS-Prüfung wurde durch einige visuelle Hilfsmittel erleichtert.
- Subversion als Versionierung wurde besser als bislang integriert.
- Adobe BrowserLabs ermöglicht eine Vorschau im Internet innerhalb verschiedener Browserkonfigurationen. Der Onlinedienst von Adobe (Adobe-Konto notwendig) vereinfach die Fehlersuche deutlich.
- Business Catalyst – ebenfalls ein Onlinedienst von Adobe – erlaubt eine relativ einfache Erstellung und Einbindung dynamischer Inhalte wie zum Beispiel

einen Onlineshop. Auch hier ist ein Adobe-Account obligatorisch, bei Nutzung des vollen Umfangs ist Business Catalyst auch nicht kostenlos. Wir verzichten daher im vorliegenden Buch auf die Beschreibung dieses Dienstes.

- Neben diesen wichtigen Features hat sich einiges an der Oberfläche und in den Menüzusammenstellungen getan. Das alles im Einzelnen hier zu beschreiben würde jedoch den Rahmen dieses Kapitels sprengen.

Waren die Neuerungen in der CS4-Version zugegebenermaßen noch etwas mager, hat die CS5-Version tatsächlich deutlich zugelegt, und wir arbeiten sehr gerne mit dieser neuen Version. Wir hoffen, dass auch Sie viel Freude damit haben werden.

1.3.1 Veraltete Funktionen

Die folgenden Funktionen sind ab Dreamweaver CS5 veraltet:

- Barrierefreiheits-Prüfungsbericht
- ASP/JavaScript-Serververhalten
- JavaScript-Verhalten Browser überprüfen
- Verbindung zu einem FTP-/RDS-Server ohne Definieren von Sites herstellen
- JavaScript-Verhalten Shockwave oder SWF steuern
- Webfotoalbum erstellen
- JavaScript-Verhalten Popupmenü ausblenden
- InContext Editing
- Verfügbare CSS-Klassen verwalten
- FlashPaper einfügen
- Mark of the Web einfügen/entfernen
- Microsoft-Visual-Sourcesafe-Integration
- Navigationsleisten
- JavaScript-Verhalten Sound abspielen
- Menü Ereignisse zeigen für (Bedienfeld Verhalten)
- JavaScript-Verhalten Popup-menü anzeigen
- JavaScript-Verhalten Zeitleiste
- Tags überprüfen
- Live-Data-Ansicht

Vor dem Erstellen einer Website sollte man sich einige grundlegende Gedanken zur Vorgehensweise im Projektablauf machen, um die Arbeiten effektiv zu organisieren.

2 Projektablauf für die Buchwebsite

Am besten kann man etwas verstehen, wenn man es sich selbst erarbeitet. Aus diesem Grund haben wir das Buch so aufgebaut, dass ein Kapitel jeweils immer die benötigten Funktionen und die dazu in Dreamweaver zur Verfügung stehenden Hilfsmittel erklärt, um dann am Ende die erlernten Schritte anhand eines Beispiels in die Praxis umzusetzen.

Bei den an die Kapitel anschließenden Schritt-für-Schritt-Anleitungen gehen wir davon aus, dass Sie das jeweils vorherige Kapitel gelesen haben. Wenn Sie also an der einen oder anderen Stelle »stolpern«, lesen Sie bitte im vorausgehenden Kapitel nochmals nach – dort werden die Arbeitsschritte erläutert.

Kein Baukastensystem [«]

Grundsätzlich ist das Buch nicht als Schritt-für-Schritt-Anleitung für eine Website nach dem Baukastenprinzip gedacht, da diese angesichts der Vielfalt an möglichen Websites geradezu zwangsläufig bei Ihren ersten eigenen Versuchen scheitern müsste. Der Schwerpunkt des Buches liegt daher auf der Erläuterung des professionellen Einsatzes der Dreamweaver-Funktionen.

Aufbau und Grafiken

Auf der DVD finden Sie im Ordner *Buchwebsite* alle Dateien, die Sie zum Nachbauen der Beispielwebsite benötigen. Im Unterordner *Website_HTML* liegt dabei die im zweiten Teil des Buchs entwickelte statische Version der Website, im Unterordner *Website_PHP* die im dritten Teil entwickelte dynamische Version. [●]

Bilder auf DVD

Die Bilder, die wir im Buch zum Aufbau der Website verwenden, dürfen wir Ihnen aus lizenzrechtlichen Gründen leider nicht auf der DVD zur Verfügung stellen, so dass wir dort auf Platzhalterbilder zurückgreifen. Die Website wird dadurch bei Ihnen etwas anders aussehen als auf den Abbildungen im Buch – der grundlegende Aufbau bleibt aber natürlich gleich.

Datenbankschema

Wenn Sie als Webentwickler arbeiten, ohne selbst grafisch tätig zu werden, sind der Navigationsplan und die Grafiken – neben einem Pflichtenheft – Ihre Arbeitsgrundlagen.

Ein Pflichtenheft für den späteren, dynamischen Teil haben wir uns wegen des doch relativ einfachen Aufbaus unserer Website gespart. Dafür haben wir ein Schema der Datenbanken und der Abfragen beigelegt – doch dazu mehr im zweiten Teil des Buches.

Unsere Buchwebsite wurde so erstellt, wie wir sie auch real bei einem Projekt realisieren würden. Das Layout und die Arbeitsvorgaben müssen nun nur noch mit Dreamweaver umgesetzt werden. Also gehen wir die Sache gemeinsam an.

2.1 Briefing

Die Buchwebsite soll zur Eigendarstellung von Künstlern und Designern dienen, wurde letztlich aber zu Übungszwecken entwickelt. Ihr liegt also kein bis ins Letzte durchdachtes Konzept zugrunde. Wir bitten Sie deshalb, etwas Nachsicht walten zu lassen.

Das Layout eignet sich hervorragend zur Erläuterung der verschiedenen kombinierten Techniken des Site-Aufbaus. Durch die Zielsetzung (Künstlerdatenbank) ergibt sich, dass die Inhalte nicht statisch, sondern dynamisch aus einer Datenbank generiert werden sollen.



Abbildung 2.1 Screenshot der Website zum Buch

Layoutvarianten

Wir kennen natürlich die Diskussion darüber, ob man Layouts nun besser mit Tabellen und einem Frameset oder mit CSS aufbaut. Unsere Haltung dazu ist, dass Tabellenlayouts veraltet sind und unter normalen Umständen nicht mehr verwendet werden sollten. Deswegen bauen wir das Layout der Buchwebsite mit CSS auf. Sollten Sie dennoch einmal ein Tabellenlayout einsetzen wollen, erhalten Sie in Kapitel 6, »Tabellen und Listen«, ein paar Hinweise dazu.

Flache Hierarchien

Die Navigationsstruktur (siehe Abbildung 2.2) ist flach gehalten, Inhalte werden in einer weiteren Ausbaustufe dynamisch aus einer Datenbank abgefragt und mit PHP eingebunden.

Beim Klick auf einen Menüpunkt erscheint zunächst eine Auswahlliste vorhandener Künstler. Klickt man auf einen der Einträge, werden dem Nutzer die Detailbeschreibung sowie Links zu den einzelnen Bildern angezeigt. Zurück zur jeweiligen Übersicht gelangt man über einen ZURÜCK-Button.

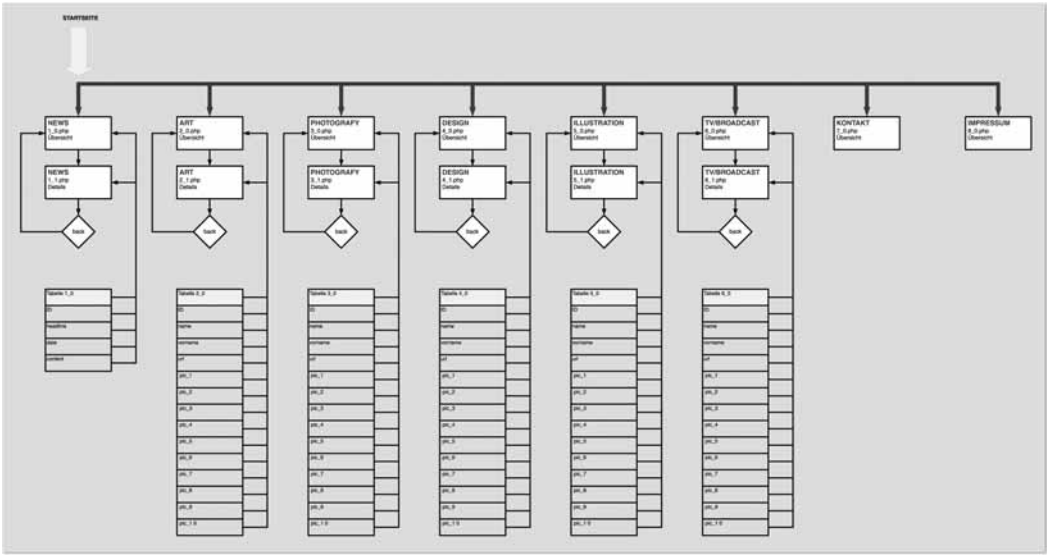


Abbildung 2.2 Navigationsplan der Buchwebsite mit Datenbankschema (auch auf der DVD)

Wie schon im Navigationsplan zu sehen, enthält der Aufbau der Website ein einfaches Datenbankschema. Wir werden die Website im ersten Teil des Buches statisch erstellen. Nach dem statischen Aufbau der Website haben Sie eine vollständig funktionsfähige Website, die auch ohne Datenbankanbindung auskommen würde.

Die Anbindung der Website an eine Datenbank und der Aufbau der notwendigen Skripte erfolgt im zweiten Teil des Buches.

2.2 Arbeitsablauf

Außer dem Layout (das Sie ja von uns geliefert bekommen) werden wir alle Arbeitsschritte detailliert beschreiben und mit Ihnen zusammen nachvollziehen. In den Kapiteln klären wir zunächst die eher theoretischen Hintergründe und programmspezifische Funktionen, um die beschriebenen Komponenten dann in unsere Übungswebsite zu integrieren.

Die Umsetzung einer Website gliedert sich in die folgenden Einzelschritte:

1. **Vollständiges Layouten der Screens in einem Grafikprogramm**
   Bauen Sie die Website komplett in einem Grafikprogramm Ihrer Wahl auf. Später hinzukommende Inhalte können eventuell direkt oder als Blindtexte eingefügt werden. Nur wenn Sie auch Beispielinhalte mit anlegen, können Sie ein Layout wirklich beurteilen. Achten Sie besonders bei Photoshop und Fireworks darauf, alles in *einer* Datei zu speichern und regelmäßig Sicherungskopien anzufertigen. Das fertige Screendesign ist die Basis für alle späteren Arbeiten. In diesem Arbeitsschritt sollten Sie so exakt und gewissenhaft wie möglich sein. Sie erleichtern sich damit alle späteren Arbeitsschritte.
2. **Zerteilen der Screens in einzelne Grafiken**
   Verwenden Sie Zeit darauf, sich das Zerteilen des Screens *(Slicing)* genau zu überlegen. Nehmen Sie ruhig Papier und Bleistift zur Hand, und zeichnen Sie auf, wie die spätere Internetseite aufgebaut werden muss. Fehler im Zerteilen führen häufig dazu, die ganze Seite erneut aufbauen zu müssen. Achten Sie auch hier auf absolute Exaktheit. Ein einzelnes Pixel erscheint nicht groß. Später kann ein Unterschied von einem Pixel das ganze Layout zerstören.
3. **Anlegen der Site, der späteren Dateistruktur und der Ordnerhierarchie**
   Diese Punkte können Sie meist direkt aus dem Navigationsplan ableiten. Wir zeigen Ihnen im folgenden Kapitel, wie Sie aus den Plänen eine sinnvolle Ordnerstruktur aufbauen.
4. **Aufbau der Grundseiten mit XHTML usw.**
   Erstellen Sie danach das Grundlayout der Seiten. Überlegen Sie, welche Elemente auf allen Seiten gleich sind, und legen Sie diese an. Die Grundseite kann dann als Vorlage gespeichert oder mehrfach kopiert werden. CSS-Stile, die auf allen Seiten Verwendung finden, werden aus dieser Datei exportiert.

Vorlagen als Option [«]

Vorlagen mit Dreamweaver sind eine Möglichkeit, eine Website zu erstellen. Sie müssen nicht zwingend damit arbeiten. Seitdem man CSS als Standard auch zum kompletten Ausbau eines Layouts einsetzt, haben Vorlagenseiten viel von ihrem Reiz verloren, da die meisten Vorteile von Vorlagen auch mit CSS erreicht werden können. In diesem Buch zeigen wir Ihnen als Option auf, wie Sie die Buchwebsite mit einer Vorlage erstellen.

5. **Aufbau der Navigation**
   Ganz wichtig ist das Erstellen einer Sitemap bzw. eines Navigationsplans. Sie erkennen so sehr schnell Unstimmigkeiten in der Struktur der Website und legen bereits hier fest, welche Dateien Ihre Site benötigt. Die Struktur der Site sollte natürlich möglichst feststehen, bevor Sie mit dem Layout beginnen. Nur so wissen Sie, welche Navigationselemente erforderlich sind.
6. **Einbinden von Inhalten**
   Nachdem die Struktur und das Grundlayout der Site stehen und die Site bereits vollständig navigierbar ist, binden Sie die Inhalte wie Texte und Bilder ein.
7. **Anbinden an Datenquellen**
   Verfügt die Site über eine Datenbank, wird diese jetzt mit PHP oder einer anderen Programmiersprache angebunden. Die Vorgehensweise bei dynamischen Sites unterscheidet sich ein wenig von der bisher beschriebenen.

2.3 Das Layout

Wie bereits erwähnt, wird das komplette Layout einer Website vorab in einem Bildbearbeitungsprogramm Ihrer Wahl erstellt, und zwar so, wie Sie die Seite später im Browser sehen wollen, inklusive der Bildinhalte. Erst dann werden die Grafiken zerteilt und mit HTML wieder zusammengefügt.

2.3.1 Entwurf in Photoshop

Photoshop ist der Standard unter den Bildbearbeitungsprogrammen schlechthin. In den meisten Agenturen und bei Profis wird wohl dieses Tool zum Einsatz kommen.

Vor allem bei größeren Projekten kommt Photoshop zum Einsatz und wird zum Erstellen funktionaler Prototypen verwendet. Dieser Prozess wird *Rapid Prototyping* genannt. Gerade die Interaktion mit diesem Programm hat sich bei Dreamweaver in den neueren Versionen deutlich verbessert und sollte Ihnen Anlass ge-

ben, sich Photoshop bei Gelegenheit genauer anzusehen. In diesem Buch können wir das leider nicht machen, da wir uns auf Dreamweaver konzentrieren werden.

Wer ImageReady aus dem alten Photoshop kennt, wird in der Version CS5 feststellen, dass alle Werkzeuge direkt in Photoshop integriert sind und der Begriff ImageReady vollkommen verschwunden ist. Das betrifft vor allem die Funktionen zum Zerteilen (*Slicen*) und Speichern für die Internetausgabe. Letzteres findet man bei Photoshop zwar unter dem Befehl FÜR WEB UND GERÄTE SPEICHERN, doch durch die weiter verbesserte Integration von Photoshop und Dreamweaver gibt es bessere und direktere Wege.

Das Zerlegen der Grafiken in Photoshop stellt eine enorme Arbeitserleichterung gegenüber dem Ausschneiden und Abspeichern von Hand dar. Bitte beachten Sie folgende Hinweise beim Zerteilen und Anlegen Ihrer Layoutgrafiken:

- Vergessen Sie nicht, die Voreinstellungen für Printprodukte auf Screenlayouts umzustellen. Sie erhalten ansonsten fehlerhafte Dokumente oder erschweren sich das Leben unnötig.
- Stellen Sie die Maßeinheiten sowie die Schriftgrößen auf Pixel um. Punktgrößen und Millimeter sind im Printbereich wichtig und richtig. Auf dem Screen wird ausschließlich in Pixeln gemessen.
- Achten Sie darauf, möglichst alles in einer Datei zu speichern. Es gibt keine Möglichkeit, eine Slicemap extern abzuspeichern und in einer anderen Datei zu verwenden. Wenn Sie Ihre Screens auf mehrere Dateien verteilen, müssen Sie die Slicemap für jede dieser Dateien neu anlegen. Das ist nicht nur mühsam, sondern auch äußerst fehleranfällig. Sie können Ebenengruppen anlegen, um eventuelle Unterseiten darin bei Bedarf ein- bzw. auszublenden. Diese Vorgehensweise ermöglicht ein wesentlich exakteres Positionieren einzelner Elemente.
- Legen Sie für Rollover-Effekte (Schaltflächen) eine weitere Ebene an, die Sie beim Export ein- bzw. ausblenden können.
- Als Standard wird in Photoshop die Schrift mit Anti-Aliasing (Einstellung SCHARF) dargestellt. Für eine größere Typografie in Logos usw. ist das auch in Ordnung. Wenn Sie jedoch Blindtexte einfügen, um Ihr Layout mit Inhalten zu beurteilen, sollten Sie Anti-Aliasing für diese Schriften deaktivieren. Stellen Sie es dafür einfach im Schriftenmenü auf OHNE ein. Das Ergebnis gleicht dann in etwa der späteren HTML-Version.
- Wenn Sie am Mac arbeiten, bedenken Sie bitte auch, dass Ihre Grafikdarstellung durch unterschiedliche Gammawerte etwas heller ist als am PC. (Dies können Sie aber durch Kalibrierung Ihres Bildschirms verhindern.)



Abbildung 2.3 Schriften ohne Anti-Aliasing

Slicemap als HTML-Datei erstellen

Photoshop bietet an, für die Slicemap gleich eine HTML-Datei mit allen Rollover-Effekten, Verlinkungen usw. zu erstellen. Klingt gut, ist es aber nicht. Wenn Sie eine professionelle Website erstellen wollen, die auch den technischen Standards entspricht, sollten Sie von dieser Möglichkeit besser die Finger lassen. Das einzig wirklich Brauchbare sind die erstellten Grafiken.

2.3.2 Entwurf in Illustrator

Wer gerne mit Vektorzeichenprogrammen arbeitet, kann die Layouts durchaus auch in Illustrator erstellen. Illustrator bietet die gleichen Möglichkeiten an, Slices zu erstellen, wie Photoshop. Ein wenig umdenken muss man jedoch, da die vektorbasierte Darstellung in Illustrator nicht zwangsläufig exakt den pixelorientierten exportierten Grafiken entspricht.

2.3.3 Alternative Programme

PhotoImpact von Corel/Ulead ist ein sehr kostengünstiges und durchaus taugliches Programm, um Internetseiten zu gestalten. Besonders vorteilhaft ist es, dass Sie eine Slice-Aufteilung extern abspeichern können, um diese in andere Grafiken zu laden. Sie können somit Ihre Screens auf mehrere Dateien verteilen. Für Websites sind die Bildbearbeitungsfunktionen von PhotoImpact vollkommen ausreichend.

In diesem Kapitel werden wir Sie mit den Grundfunktionen und Einstellungen des Programms vertraut machen. Wir zeigen auf, welchen Sinn die verschiedenen Einstellungen haben und wie Sie diese praktisch nutzen können.

3 Grundlagen von Dreamweaver

Zunächst sollten wir einige Begriffe klären. Häufig verwenden verschiedene Programme und Autoren für identische Funktionen unterschiedliche Begriffe. Um Verwirrungen vorzubeugen, möchten wir deshalb an dieser Stelle Begriffe, die wir häufig verwenden, eindeutig definieren. Im weiteren Verlauf werden wir bei auf Dreamweaver bezogenen Begriffen auf die von Adobe vorgegebenen Bezeichnungen zurückgreifen. So können Sie diese in der Hilfedatei von Dreamweaver einfacher finden.

Statische Websites

Bei *statischen* Websites sind die Inhalte fest (statisch) im HTML-Dokument verankert.

Dynamische Websites

Bei *dynamischen* Websites werden die Inhalte aus Datenquellen aller Art generiert. Es spielt dabei keine Rolle, ob dies immer beim Aufruf einer Seite (also in Echtzeit) oder nach Anweisung durch einen Administrator geschieht. Diese Datenquellen können Datenbanken, einfache Textdateien, XML-Dokumente und anderes sein.

Nicht damit gemeint sind sich dynamisch verändernde Websites (im Sinne von Bewegung, z. B. aufklappende Menüs). Diese fassen wir unter dem Begriff *DHTML* zusammen.

DHTML/Ajax

DHTML ist eine Mischung aus HTML und in den allermeisten Fällen JavaScript, eventuell auch Java Applets. Heute spricht man oft von Ajax (*Asynchronous JavaScript and XML*), das DHTML ersetzt hat und neue Möglichkeiten bietet.

3.1 Die Programmoberfläche

3.1.1 Verschiedene Anzeigemöglichkeiten

Dreamweaver bietet Ihnen in der vorliegenden Version bis zu acht verschiedene Anzeigemöglichkeiten des Arbeitsbereichs. Die Designer-Ansicht (siehe Abbildung 3.2) ist die Standardeinstellung, die Sie beim ersten Start von Dreamweaver zu sehen bekommen.

[»] **Umschalten zwischen den Arbeitsbereichen**

Die Anzeigemöglichkeiten bei Dreamweaver können Sie im Menü FENSTER • ARBEITSBEREICHLAYOUT ❷ umschalten oder direkt über das Icon ❶ (siehe Abbildung 3.1). Sie müssen Dreamweaver nicht mehr wie in früheren Versionen neu starten, um die Einstellungen zu übernehmen. Auch eigene Anordnungen der Fenster können Sie speichern und bei Bedarf abrufen.

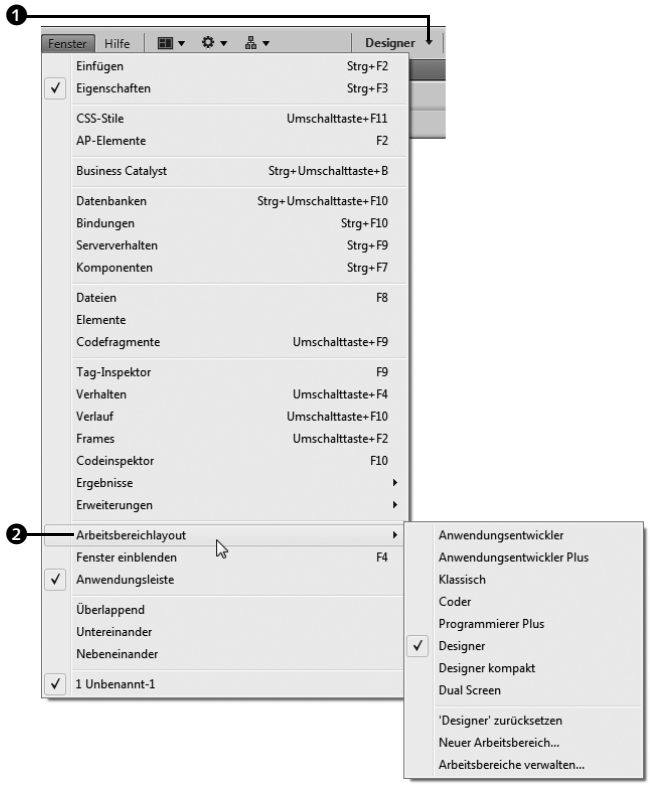


Abbildung 3.1 Verschiedene Anzeigemöglichkeiten

In der Coder-Ansicht (siehe Abbildung 3.3) verschiebt sich die Bedienfeldgruppe nach links, und die Eigenschaftenpalette wird verkleinert. Als weitere Option können Sie ein Layout für Dual Screen (siehe Abbildung 3.4) einschalten.

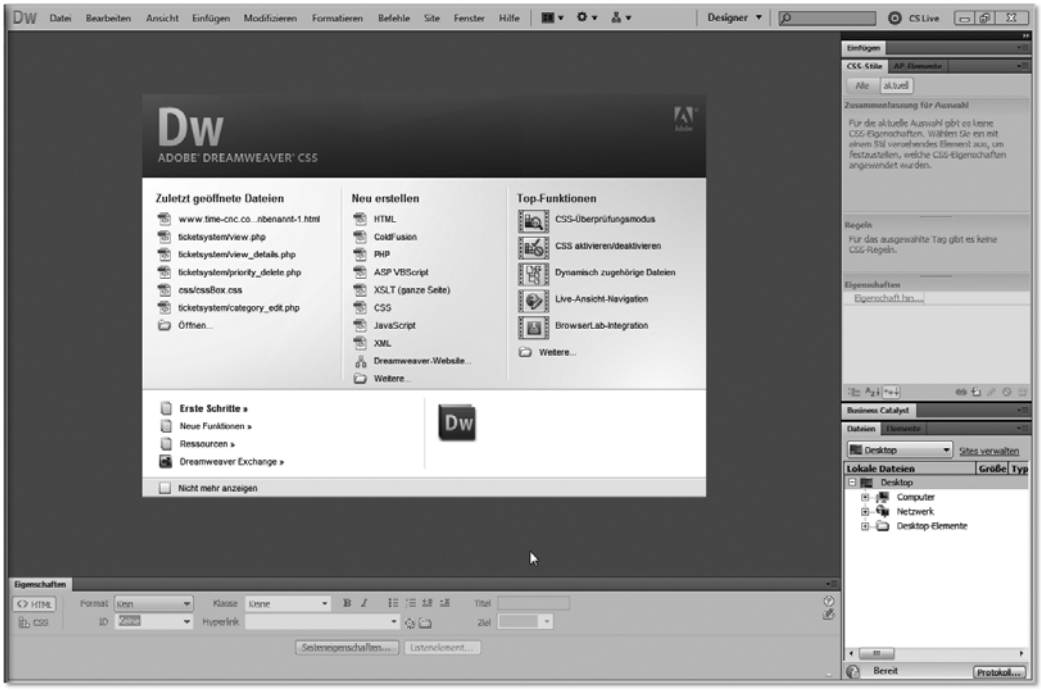


Abbildung 3.2 Der Startbildschirm in der Designer-Ansicht

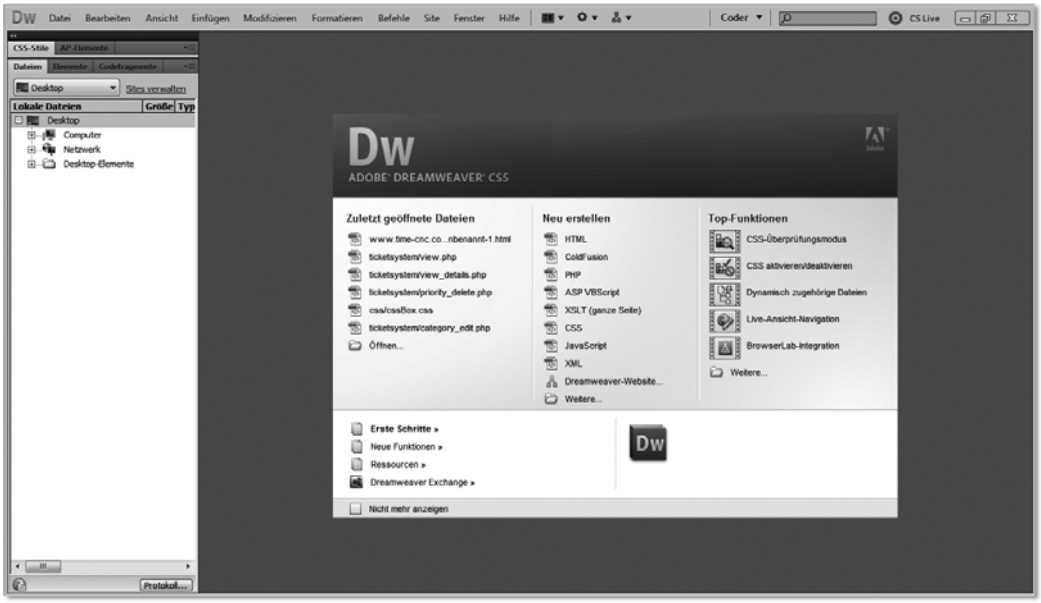


Abbildung 3.3 Der Startbildschirm in der Coder-Ansicht – die Paletten der rechten Seite rutschen nach links.

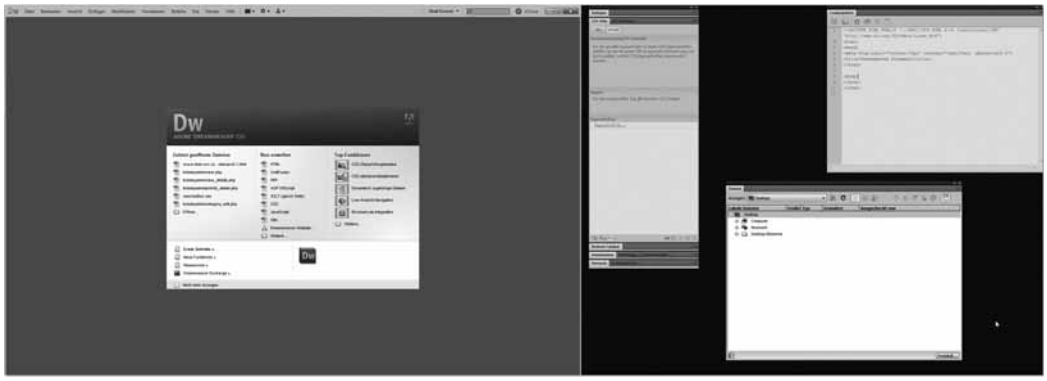


Abbildung 3.4 Ansicht im Dual-Screen-Modus

Durch frei schwebende Fenster ist das Dual-Screen-Layout bestens für die Arbeit mit großen Monitoren bzw. für den Betrieb von zwei Monitoren geeignet.

[»] **Klassisches Arbeitsbereichslayout**

Im Buch arbeiten wir ausschließlich mit dem klassischen Arbeitsbereichslayout. Damit möchten wir erreichen, dass auch Umsteiger von älteren Dreamweaver-Versionen schnell den Einstieg finden, da die Anordnung der Menüleisten und die Bedienfeldgruppen in gewohnter Weise erfolgt.

Die Arbeitsoberfläche (siehe Abbildung 3.5) unterteilt sich in die Einfügeleiste ❶, die Registergruppen ❷ mit den einzelnen Bedienfeldern und das Dateifenster ❹ sowie in das eigentliche Dokumentfenster ❸ mit der Eigenschaftenpalette ❺.

Logische Struktur

Die grundsätzliche Funktionsweise von Dreamweaver spiegelt sich in der Aufteilung der Bereiche wider. Die Einfügeleiste ❶ fügt Ihrem Dokument ein HTML-Tag hinzu. In der Eigenschaftenpalette ❺ können Sie alle wichtigen Attribute des angewählten oder zuvor eingefügten HTML-Tags einstellen und bearbeiten sowie die jeweils zugewiesenen CSS-Stile bearbeiten. Um die Attribute eines Elements zu bearbeiten, müssen Sie dieses mit der Maus vorher aktivieren. Weiterführende Aktionen außerhalb der Möglichkeiten, die Ihnen HTML bietet (wie zum Beispiel JavaScript, CSS und Serververhalten), finden Sie in den nach Kategorien gruppierten Registergruppen ❷.

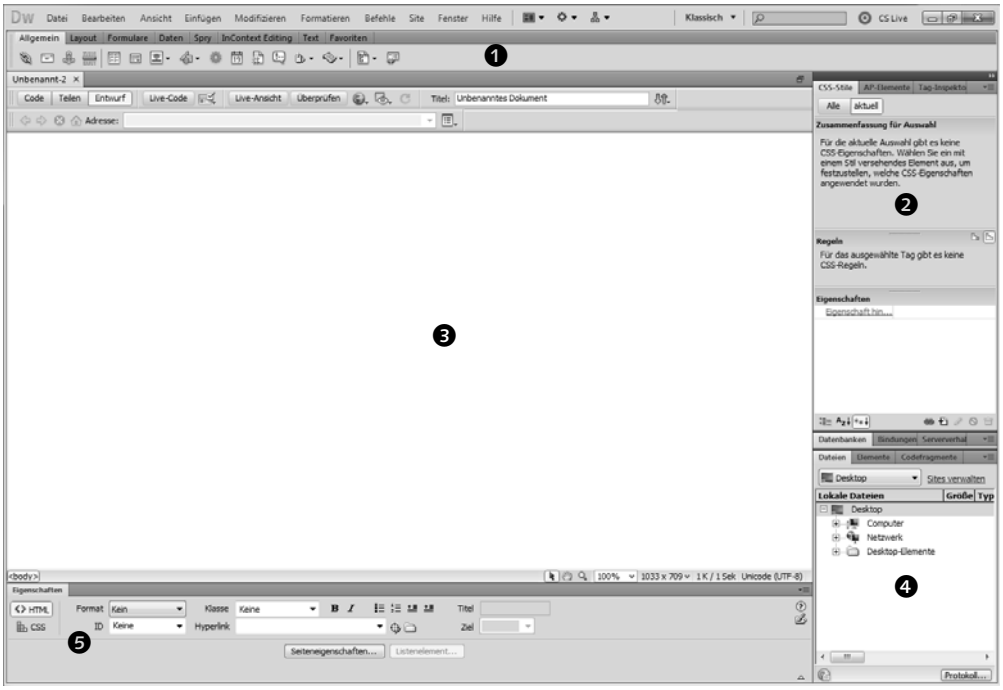


Abbildung 3.5 Die klassische Arbeitsoberfläche in der Übersicht

Dateifenster

Die Platzierung des Dateifensters ❹ neben dem Dokumentfenster ermöglicht den schnellen Zugriff auf Dokumente. Bilder und Verlinkungen können per Drag & Drop rasch in das aktuell bearbeitete Dokument ❸ gezogen werden.

3.1.2 Anpassen der Registergruppen

Ähnlich einfach ist die Handhabung der Registergruppen bzw. die Veränderung der Zusammenstellungen innerhalb dieser Gruppen.

Bedienfelder sind die einzelnen Elemente (Register) in einer Registergruppe. Diese können Sie wie aus anderen Adobe-Anwendungen gewohnt einfach »herausziehen« und durch Ablegen auf einer anderen Registergruppe umgruppieren.

3.1.3 Platz zum Arbeiten schaffen

Wenn Sie mit einem kleineren Monitor arbeiten, werden die zahlreichen Paletten und Fenster schnell lästig. Durch das Ein- und Ausklappen der einzelnen Registergruppen oder ganzer Bildschirmbereiche können Sie jedoch auch mit weniger Platz sehr komfortabel arbeiten.

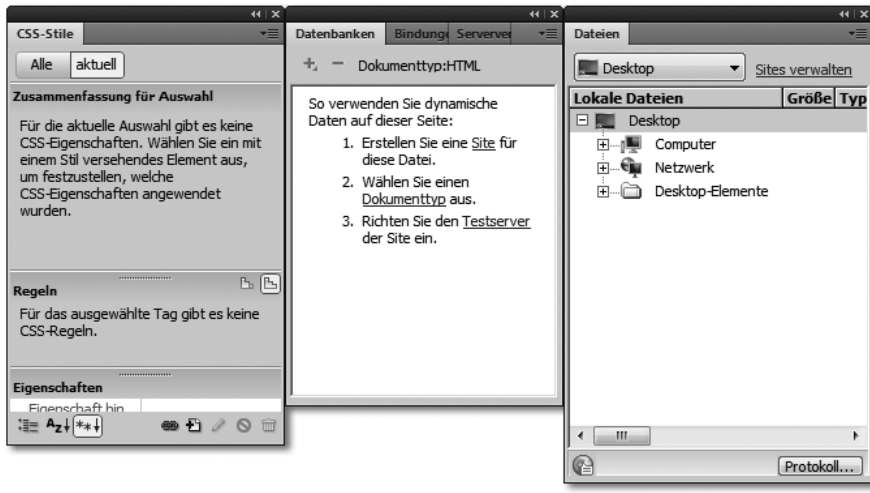


Abbildung 3.6 Geöffnete Bedienfelder



Abbildung 3.7 Geschlossene Bedienfelder

3.1.4 Einfügeleiste anpassen

Standardmäßig ist die Einfügeleiste rechts oben als Palette dargestellt. Dies ist eine Änderung seit CS4, die nicht alle Anwender glücklich macht, denn es bedarf mehrerer Klicks, um an die Stelle zu gelangen, an der das benötigte Icon sitzt.

Alternativ dazu können Sie die Einfügeleiste an ihrem Kopf herausziehen und an die (gewohnte) Stelle unterhalb der Menüleiste setzen. Hier haben die Funktionen genügend Platz, um ohne weitere Klicks angewählt werden zu können.

Wir haben uns im Buch für die Ansicht als Registerkarte (klassisch) entschieden. Sollte also Ihre Einfügeleiste anders aussehen als hier im Buch, schalten Sie diese eventuell um. In beiden Ansichten stehen Ihnen jedoch alle Funktionen der Palette zur Verfügung.

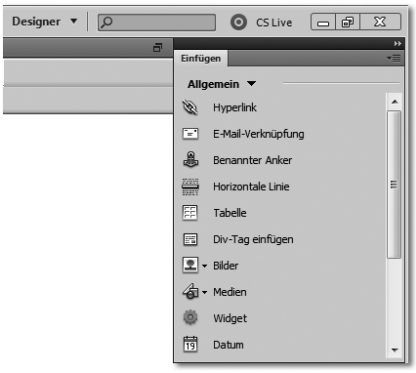


Abbildung 3.8 Die Einfügeleiste in der Standardansicht (Design)

3.1.5 Eigene Einfügeleiste erstellen

Während der praktischen Arbeit benötigt man die meisten Elemente in der Einfügeleiste nicht. Sie werden feststellen, dass es immer wieder die gleichen Elemente sind, die Sie verwenden. Im Register FAVORITEN können Sie sich eine eigene Einfügeleiste mit den am häufigsten benötigten Elementen zusammenstellen (siehe Abbildungen 3.9 und 3.10).



Abbildung 3.9 Eine eigene Einfügeleiste erstellen



Abbildung 3.10 Hinzufügen von häufig benötigten Elementen

3.1.6 Das Dateifenster

Für umfangreiche Dateioperationen ist das Dateifenster in der Bedienfeldgruppe zu klein. Wechseln Sie daher für diese Aktionen durch Klicken auf das Icon ❶ im Bedienfeld DATEIEN in die Site-Ansicht von Dreamweaver.

Beachten Sie, dass Ihnen dieses Icon erst zur Verfügung steht, nachdem Sie eine Site definiert haben. Wie dies geschieht, zeigen wir Ihnen weiter hinten in diesem Buch.



Abbildung 3.11 Umschalten zur Site-Ansicht

Durch einen Rechtsklick im Dateifenster oder in der Site-Ansicht können Sie diverse Dateioperationen durchführen. Hier verstecken sich auch so alltägliche Funktionen wie Kopieren und Duplizieren. Um zum Dokument zurückzugelangen, schließen Sie einfach das Fenster.

Die Funktionen der Site-Verwaltung sind für das Arbeiten mit Dreamweaver und für das Erstellen von Websites grundlegend. Wir haben diesem Modul deshalb einen eigenen, ausführlichen Abschnitt gewidmet (siehe Abschnitt 4.2, »Die Site-Verwaltung«).

3.1.7 Verschiedene Dokumentansichten

Dreamweaver bietet Ihnen mehrere Möglichkeiten der Dokumentansicht, zwischen denen Sie sehr einfach wechseln können. So können Sie am oberen linken Rand des Dokumentfensters zwischen den Ansichten CODE für die reine Code-Ansicht, TEILEN für eine geteilte Code- und Entwurfsansicht und ENTWURF für die reine Entwurfsansicht oder Layoutansicht wählen.

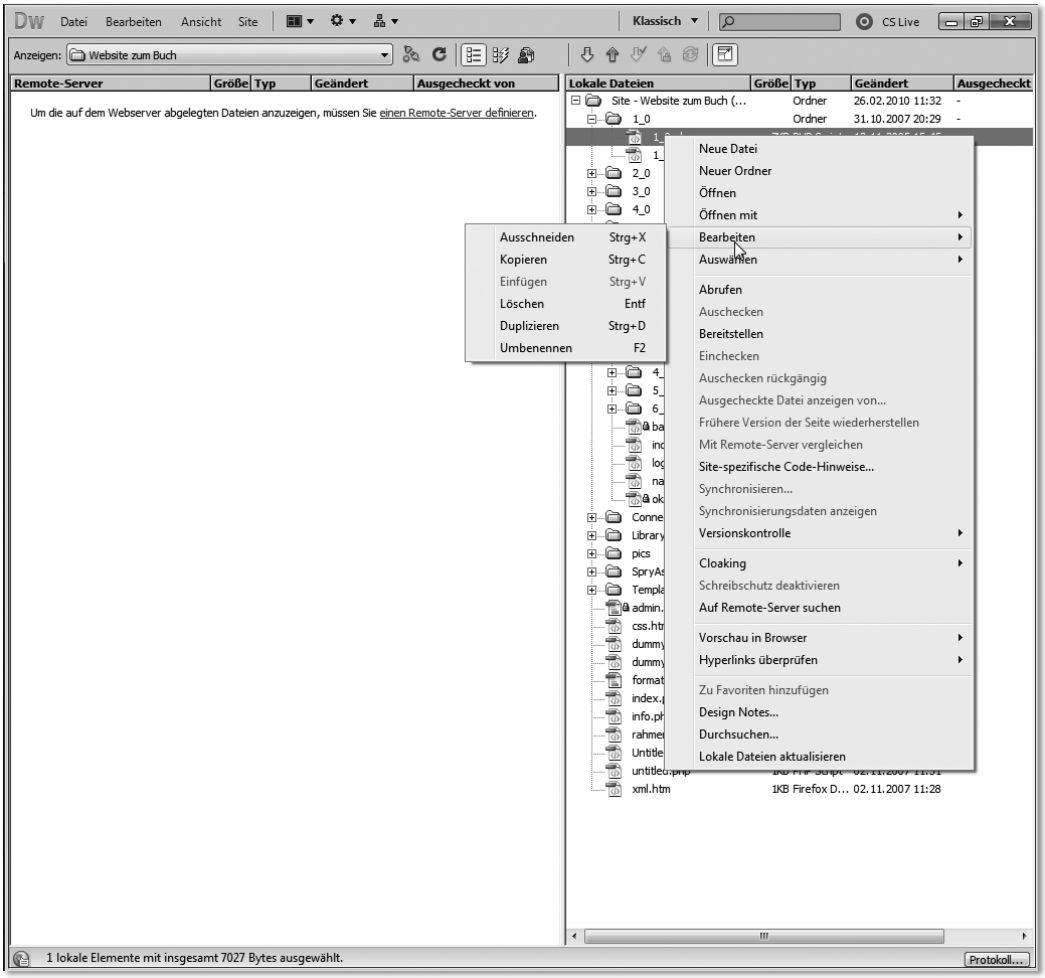


Abbildung 3.12 Site-Ansicht von Dreamweaver

Windows-User haben es übrigens sehr einfach, denn mit Strg+# (die #-Taste befindet sich rechts neben der ↵-Taste) können Sie schnell zwischen der Entwurfs- und der Layoutansicht hin- und herwechseln. Das hilft Ihnen, wenn Sie zwischendurch kontrollieren möchten, was Dreamweaver für Sie im Code generiert hat.



Abbildung 3.13 Wechseln zwischen den Dokumentansichten

[»] **Elemente im Code auffinden**

Codezeilen und HTML-Tags können Sie ganz einfach im Quelltext auffinden, indem Sie das gewünschte Element in der Layoutansicht markieren und dann in die Codeansicht wechseln. Dreamweaver markiert im Quelltext die entsprechenden Befehle. Setzen Sie den Cursor im Layout an eine bestimmte Stelle, zum Beispiel in eine Tabellenzelle, wird der Cursor auch im Quelltext an diese Stelle gesetzt. So lassen sich Elemente in umfangreichen Quelltexten sehr einfach auffinden.

3.1.8 Die Entwurfsansicht

In der Entwurfsansicht können Sie Ihr Layout betrachten, wie es im Browser aussehen wird. Die integrierte Vorschau ist – zumindest in Bezug auf neuere Browser – sehr gut. Es werden zusätzlich einige visuelle Hilfsmittel angezeigt, um das Layouten zu erleichtern (Formulare, Tabellenrahmen, unsichtbare Elemente). Diese visuellen Hilfsmittel können Sie auch einzeln abschalten ❶.

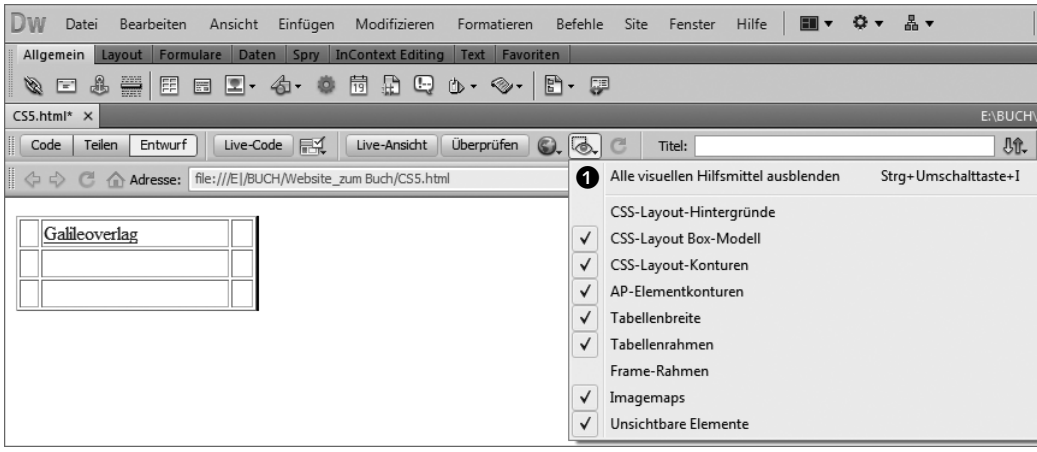


Abbildung 3.14 Entwurfsansicht mit visuellen Hilfsmitteln

[!] **Arbeitsoberfläche aufräumen**

Achten Sie beim Layouten darauf, dass auch visuelle Hilfsmittel Platz beanspruchen, und schalten Sie diese bei Bedarf aus. Besonders wichtig ist dies, wenn Sie mit vielen unsichtbaren Elementen arbeiten, was sich bei dynamischen Seiten oder Seiten mit AP-Elementen und Formularen fast immer ergibt.

Vorschau in Dreamweaver

Da es auf dem Markt nicht nur ein Browsermodell gibt und Dreamweaver nicht alle verfügbaren Browsermodelle gleichzeitig darstellen kann, muss die Vorschau zwangsläufig ein – wenn auch guter – Kompromiss zwischen den aktuellen Browsern sein. Die Vorschau innerhalb von Dreamweaver ersetzt in keinem Fall eine Layoutkontrolle in verschiedenen aktuellen Browsern.

In der Entwurfsansicht haben Sie verschiedene weitere Hilfestellungen für Tabellen und CSS zur Verfügung. Wir werden in späteren Kapiteln darauf noch detailliert eingehen.

Quelltext parallel anzeigen

In der Entwurfsansicht verliert man schnell das Gefühl dafür, dass es sich bei Dreamweaver nicht um ein Layoutprogramm handelt, sondern nur um die grafische Oberfläche eines Programmiertools. Auch wenn Sie in der grafischen Ansicht arbeiten, erstellen Sie Quellcode. Wenn Sie beginnen, mit Dreamweaver Websites zu programmieren, kann es sinnvoll sein, Layout und Quelltext parallel zu sehen. Sie bekommen auf diese Weise ein Gefühl dafür, was tatsächlich passiert, wenn Sie etwas im Layout verändern. Sie können dies mit der geteilten Ansicht erreichen.

In der geteilten Ansicht können Sie Änderungen im Quelltext oder im Layout sofort mitverfolgen. Nehmen Sie Änderungen im HTML-Quelltext vor, müssen Sie Dreamweaver die Möglichkeit geben, die Darstellung des Dokuments neu zu rendern. Klicken Sie daher, nachdem Sie die Änderungen vorgenommen haben, wieder mit der Maus in das Layoutfenster, um auch in der Layoutvorschau Ihre Änderungen zu sehen. Alternativ können Sie dies auch mit einem Klick auf den Aktualisieren-Button in der Eigenschaftenpalette oder durch den Tastaturbefehl F5 bewerkstelligen. Letzteres hat den Vorteil, dass sich der Cursor nicht aus dem Quelltext entfernt und Sie danach sofort weiterprogrammieren können.

Gerade bei Bildschirmen im Breitformat oder wenn Sie mit zwei Bildschirmen arbeiten, ist die vertikale Teilung des Screens die bessere Wahl. Aktivieren können Sie diese Ansicht nur im Menü Ansicht • Vertikal teilen. Direkt darunter befindet sich auch der Befehl, der die Entwurfsansicht auf die linke Bildschirmseite verschiebt.

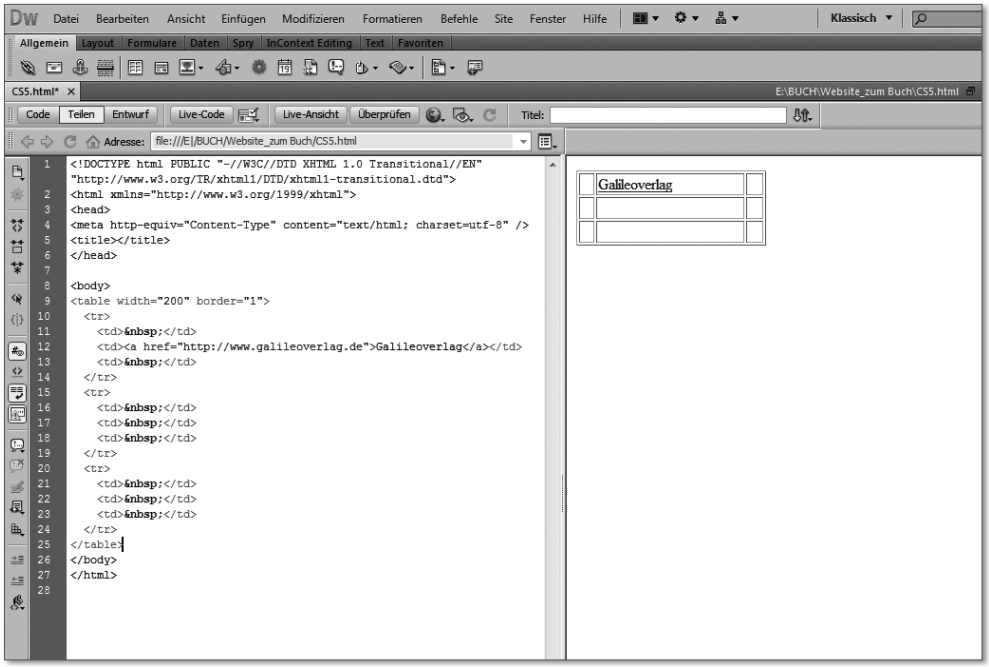


Abbildung 3.15 Code- und Entwurfsansicht parallel anzeigen

[»] Programmieren lernen während der Arbeit

Die geteilte Ansicht ermöglicht es Ihnen, im Quelltext mitzuverfolgen, welche Auswirkungen Ihre Aktionen auf den Code haben. Wenn Sie HTML erlernen oder Ihre Kenntnisse vertiefen wollen, ist dies eine hervorragende Möglichkeit, um ein Gefühl für den Aufbau eines HTML-Dokuments zu bekommen.

3.1.9 Die Codeansicht

Die Codeansicht zeigt den Quelltext des Dokuments direkt an. Die Syntax der Programmiersprachen wird sehr gut hervorgehoben, und Sie können deutlich zwischen Inhalten, Befehlen und Attributen unterscheiden.

Zwischen dieser Ansicht und der Ansicht des Quelltextes, den noch Dreamweaver 4.0 erzeugt hat, liegen Welten. Bereits in Dreamweaver MX wurde Homesite als Code-Editor integriert. Ab CS3 mausert sich Dreamweaver zum universellen Programmiertool für Websites. In der aktuellen Version ist dieses Werkzeug mächtiger und komfortabler, vor allem im Umgang mit JavaScript und CSS – beste Voraussetzungen für den Einsatz von Spry. Über die vielen Möglichkeiten, die Ihnen beim Arbeiten mit Quelltext geboten werden, erfahren Sie in Kapitel 17, »Quelltext de luxe«, mehr.

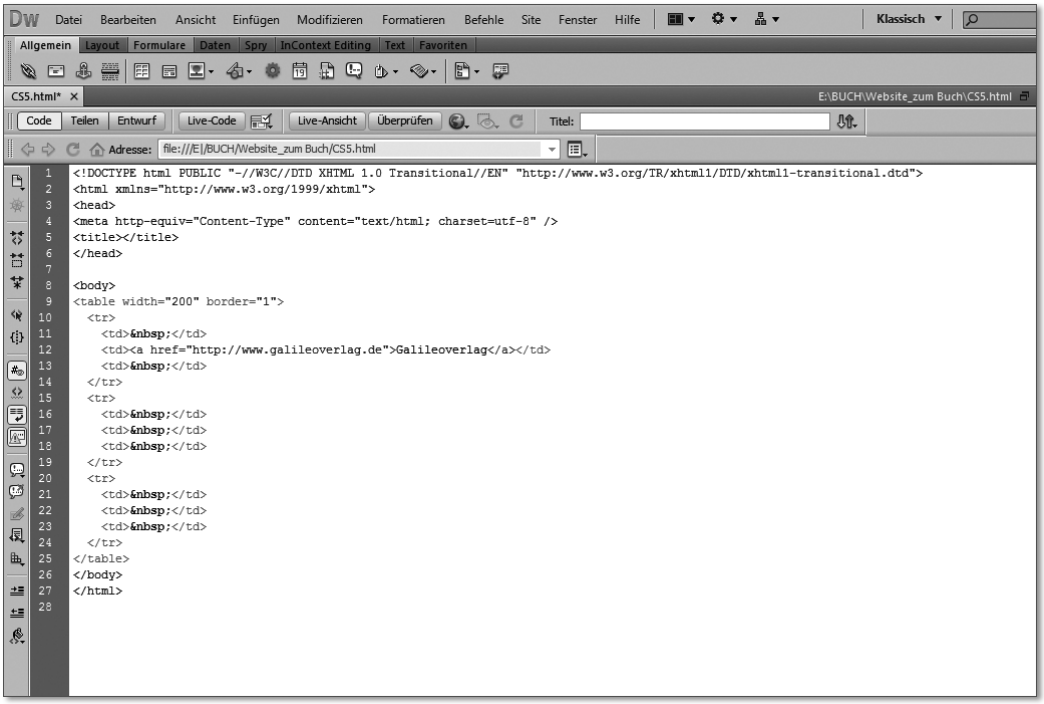


Abbildung 3.16 Das Dokumentfenster in der Codeansicht

3.1.10 Testserverbetrieb und Live-Ansicht

Eine hervorragende Arbeitserleichterung und Zeitersparnis ist die Anzeige dynamischer Seiten im Testserverbetrieb und bei der Verwendung von JavaScript und anderen dynamischen Inhalten. Sie können in dieser Ansicht genau wie in der Live-Ansicht während des Serverbetriebs durch PHP oder andere Skriptsprachen generierte Seiten direkt in Dreamweaver sehen und bearbeiten.

In der Live-Ansicht werden alle optischen Hilfsmittel automatisch ausgeblendet, der JavaScript-Code interpretiert und alle externen Dateien eingefügt. Es ist quasi so, als würde die Site nun tatsächlich im Browser erscheinen.

Nicht nur bei Serverbetrieb verfügbar [«]

Im Gegensatz zur Live-Data-Ansicht aus früheren Versionen, die nur in Verbindung mit einem Testserver zur Verfügung stand, ist die neue Live-Ansicht auch ohne Testserver verfügbar.

Im Testserverbetrieb steht Ihnen eine weitere Funktionsleiste zur Verfügung, die wir in Kapitel 20, »PHP mit Dreamweaver«, genau erläutern.

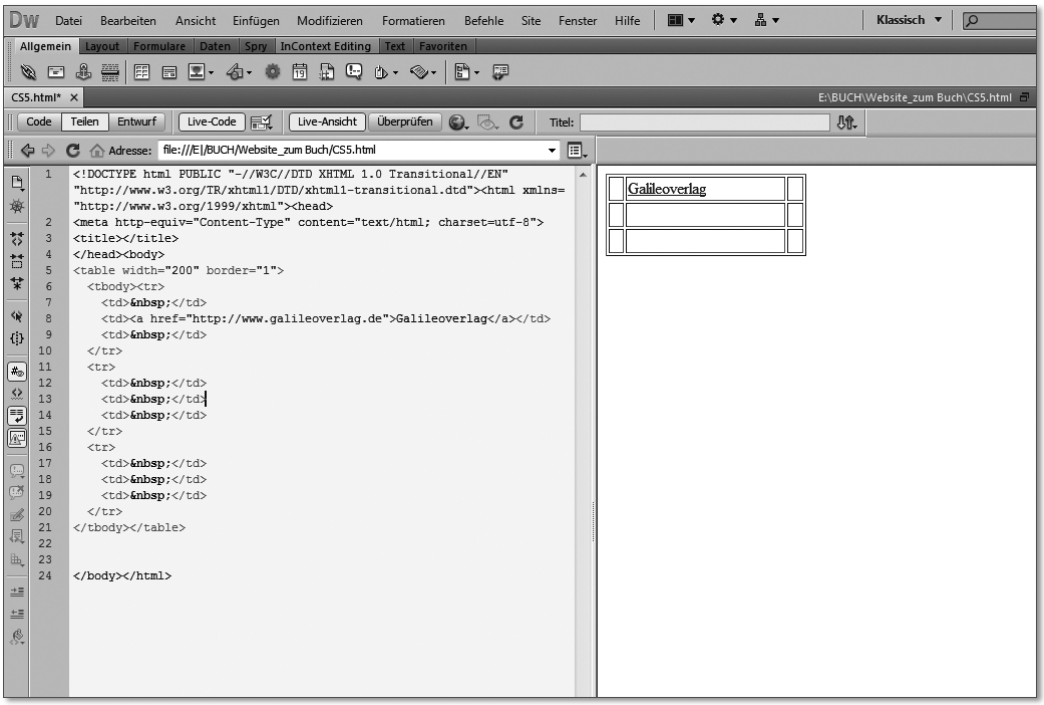


Abbildung 3.17 Das Dokumentfenster in der Live-Ansicht

3.1.11 Live-Code

Seit Version CS4 gibt es eine Ansicht namens LIVE-CODE. Diese Ansicht können Sie aktivieren, wenn Sie sich in der Live-Ansicht befinden. Im Quelltext zeigt Dreamweaver dann den Code so, wie ihn der Browser sieht. Vor allem in Zusammenhang mit JavaScript oder Spry wird nun die Seite komplett aufgebaut und erst der resultierende Code angezeigt.

Ein gutes Anwendungsbeispiel für den Live-Code stellt das Spry-Menü dar. Mit Hilfe von JavaScript werden beim Überfahren eines Menüpunkts mit der Maus im Code dynamisch entsprechende Untermenüs eingeblendet. Dazu verändert JavaScript den Code und fügt den Unterpunkten eine andere CSS-Klasse hinzu. Diese Veränderung kann im Live-Code betrachtet werden, somit kann bei Fehlern auch der Zustand beim Mouse-Over überprüft werden.

Zum Bearbeiten des Codes muss diese Ansicht allerdings wieder deaktiviert werden.

Verlinkungen können durch Klicken mit gedrückter Strg-Taste aktiviert werden.

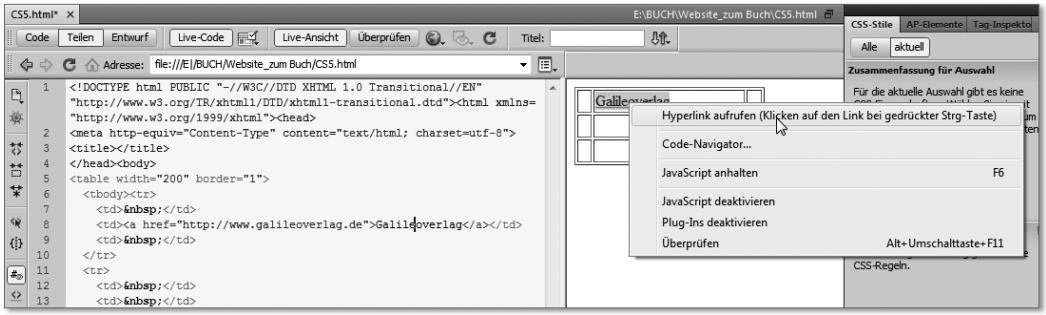


Abbildung 3.18 Aktivieren von Hyperlinks in der Live-Ansicht

Selbst Verlinkungen zu externen Websites werden korrekt angezeigt.

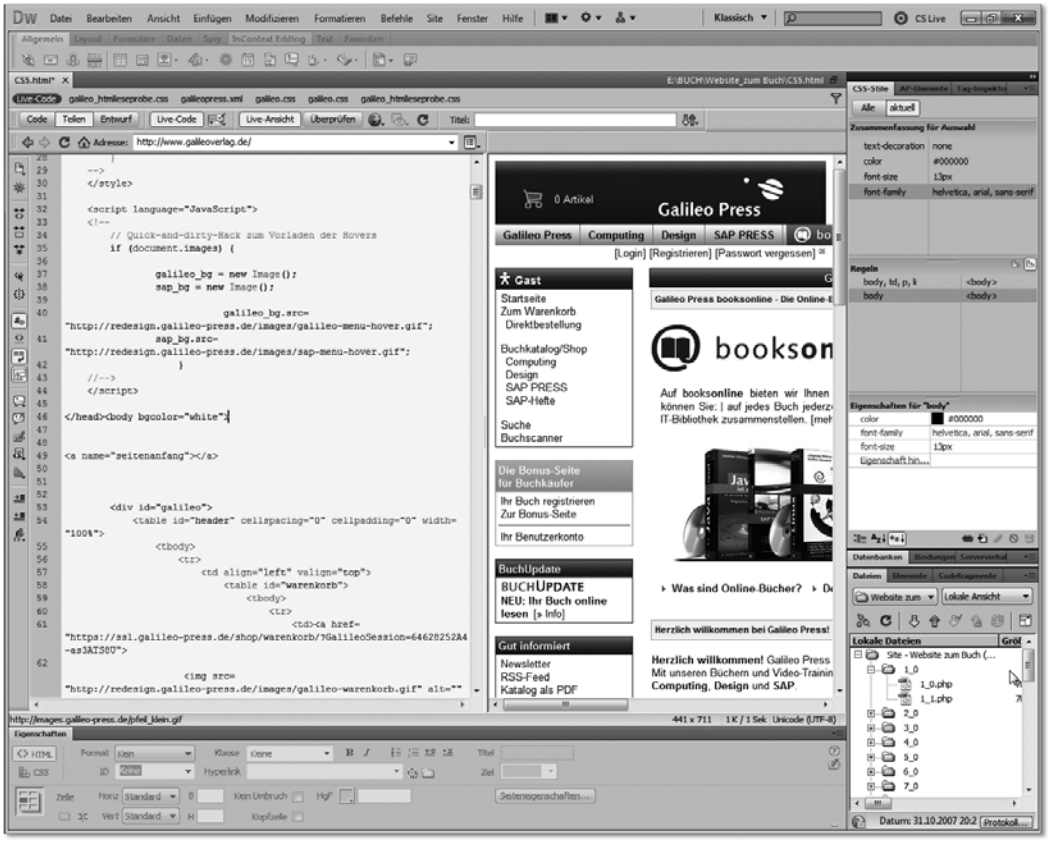


Abbildung 3.19 Ansicht externer Websites in Dreamweaver

[!] **Firewall konfigurieren**

Die Ansicht externer Websites in Dreamweaver funktioniert nur, wenn Sie Ihre eventuell vorhandene Desktop-Firewall entsprechend konfiguriert haben. Falls nicht, erhalten Sie eine Fehlermeldung.

3.1.12 Code einfrieren

Nicht immer ist man mit der Maus so schnell, wie der Browser den Code berechnet. Daher können Sie in der Live-Ansicht und im Live-Code mit F6 die Ausführung von JavaScript anhalten. Dann funktionieren zwar z. B. Spry-Widgets nicht mehr, aber der Code kann in diesem Zustand ohne Veränderung betrachtet werden.

Sinnvoll ist das z. B., wenn Sie mit der Maus einen Menüpunkt überfahren haben, das Untermenü ausgeklappt ist und Sie den Code untersuchen möchten. Müssten Sie in diesem Zustand scrollen, um an eine andere Codestelle zu gelangen, so würden Sie das Menü verlassen, und der Mouse-Over-Status ginge verloren. Hier bietet sich das Einfrieren des Codes an. Ein erneuter Druck auf F6 setzt die Ausführung wieder fort.

In der Statuszeile des Dokumentfensters werden die aktuellen Einstellungen angezeigt.

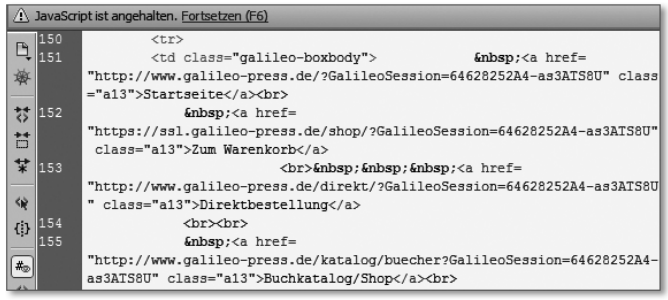


Abbildung 3.20 JavaScript wurde angehalten.

3.1.13 Code-Navigator

Ein weiteres Hilfsmittel bei der Entwicklung mit CSS ist der sogenannte *Code-Navigator*. Wurde ein Dokument mit Hilfe von CSS gestaltet, ist es manchmal nicht einfach herauszufinden, welche CSS-Abschnitte das Aussehen eines Bereichs im Zusammenspiel beeinflussen.

Sofern Sie sich in der Entwurfsansicht befinden und in einen Bereich klicken, der mit CSS formatiert ist, erscheint nach etwa einer Sekunde Pause ein Icon mit

einem kleinen Steuerrad. Alternativ können Sie den Code-Navigator auch mit dem Tastaturkürzel Strg+Alt+N aufrufen. Ein Klick darauf zeigt Ihnen alle CSS-Dateien und deren Definitionen, die genau jenen Ort beeinflussen, an dem sich der Cursor befindet.

Überfahren Sie zudem mit der Maus diese Auflistungen, so erhalten Sie die genauen Eigenschaften der Definition. Ein Klick auf eine Definition öffnet gegebenenfalls das externe Stylesheet und bringt Sie an die richtige Stelle im Dokument. So können Sie schnell die CSS-Eigenschaften ändern.

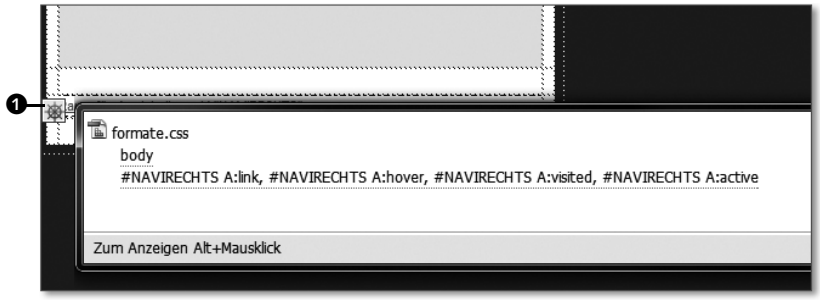


Abbildung 3.21 Der Code-Navigator verbirgt sich hinter dem kleinen Steuerrad ❶ – dieses taucht aber nur auf, wenn der entsprechende Abschnitt mit CSS formatiert ist.

Aktualisieren nicht vergessen [!]

Denken Sie daran, dass die Änderungen in einer CSS-Datei oder im Quelltext erst dann in der Layoutansicht dargestellt werden, wenn Sie diese z. B. mit F5 aktualisieren. Gegebenenfalls müssen Sie außerdem noch zuvor die externen Dateien speichern.

3.1.14 Abhängige Dateien

Da HTML-Seiten heutzutage häufig nicht nur aus einer Datei (mit Bildern) bestehen, sondern externe Stylesheets, JavaScripts oder andere Dateien in sie eingebunden werden, um eine saubere Trennung von Funktion, Inhalt und Design zu gewähren, ist es nicht immer leicht, den Überblick zu behalten, welche Elemente letztendlich zusammenwirken.

Sofern eine externe Datei eingebunden ist, erscheint diese in Dreamweaver direkt in der Leiste unterhalb des Dateinamens. Überfahren Sie diesen Dateinamen mit der Maus, sehen Sie sogar den genauen Pfad der Datei, das beugt Verwechslungen vor. Ein Klick auf den Namen der Datei bringt den entsprechenden Inhalt auf den Schirm.

Interessant und sehr praktisch ist hier der Umstand, dass in der Layoutansicht weiter die eigentliche HTML-Seite angezeigt wird, aber zum Beispiel in der

parallelen Codeansicht die CSS-Datei erscheinen kann. So können Sie Änderungen an der CSS-Datei vornehmen und nach der Aktualisierung der Ansicht die Auswirkungen direkt im Layout sehen, ohne wieder zur HTML-Datei zu wechseln.

Die eigentliche Hauptdatei, also die HTML-Seite, erreichen Sie durch einen Klick auf das Stichwort QUELLCODE in der Leiste der abhängigen Dateien.

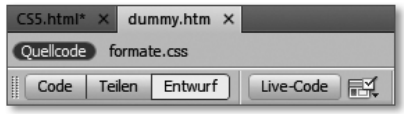


Abbildung 3.22 Die Leiste mit den abhängigen Dateien

[!] **Fehlerquelle**

Vergessen Sie bei Änderungen in abhängigen Dateien nie, auch diese Dateien zu speichern und eventuell auf dem Webserver zu publizieren. Bedenken Sie bitte ebenfalls, dass externe Dateien meistens mit mehreren Dokumenten verknüpft sind und somit jede Änderung an diesen Dateien weitreichende Auswirkungen auf die gesamte Site haben kann.

3.1.15 Mit mehreren Dokumenten arbeiten

Oftmals ist es während der Arbeit notwendig, mehrere HTML-Seiten gleichzeitig zu bearbeiten, um z. B. Inhalte zu kopieren. Dreamweaver erlaubt Ihnen, mehrere Dateien zu öffnen und zu bearbeiten. Alle geöffneten Dokumente erhalten ein eigenes Register, das Sie anklicken können, um zwischen den Dateien umzuschalten. Das Tastaturkürzel Strg+⇆ wechselt ebenfalls zyklisch durch die Dokumente.

In diesem Zusammenhang sollten noch zwei Tipps erwähnt werden: Strg+W schließt das aktuelle Dokument. Das Speichern mit Strg+S ist sicher bekannt, doch gibt es in Dreamweaver im Menü DATEI den Befehl ALLES SPEICHERN. Wenn Sie nun in BEARBEITEN • TASTATURBEFEHLE das Kürzel neu zuweisen, können Sie an mehreren Dateien arbeiten und alle gleichzeitig mit einem Tastendruck speichern – sehr praktisch, besonders im Umgang mit CSS-Dateien, die man gerne zu speichern vergisst, bevor man die HTML-Seite im Browser betrachtet. Natürlich wird dann der aktualisierte Inhalt nicht angezeigt, und die Verzweiflung ist groß.

3.1.16 Codeteilung

Als letzte nützliche Ansichtsoption sei erwähnt, dass es möglich ist, das gleiche Dokument in zwei Fenstern nebeneinander in der Codeansicht zu sehen und zu

bearbeiten. Dies ist sehr hilfreich, wenn Sie innerhalb eines längeren Dokuments Code kopieren oder verschieben möchten.

Sie finden dieses Feature unter Ansicht • Codeteilung. Der Bildschirm teilt sich nun in zwei Fenster, die den gleichen Quelltext enthalten. Alle Bearbeitungen in dem einen Fenster werden natürlich sofort in dem anderen Fenster mitgeführt, denn es handelt sich ja um die gleiche Datei. Allerdings können Sie die Ansichten nun unabhängig voneinander scrollen und so Inhalte links im Code mit jenen rechts vergleichen und bearbeiten, ohne jeweils wieder neu blättern zu müssen. Das spart sehr viel Mausbewegung und Zeit.

3.2 Wichtige Voreinstellungen

Sie können und sollten Dreamweaver an Ihre Bedürfnisse und die Anforderungen eines Projekts anpassen. Im Menü Bearbeiten • Voreinstellungen bzw. Dreamweaver • Einstellungen am Mac finden Sie alle Einstellmöglichkeiten. Es gibt fast nichts, was nicht anpassbar wäre. Für viele Einsatzzwecke reichen die Standardeinstellungen; einige Ausnahmen sollten Sie jedoch kennen, zumal die Funktion des Programms dadurch maßgeblich beeinflussbar ist.

Gehen Sie die Voreinstellungen sorgfältig durch. Es geschieht sehr schnell, dass man aufgrund falscher Konfigurationen fehlerhafte Dateien erhält.

3.2.1 Allgemeine Voreinstellungen

Ob Sie sich die Startseite von Dreamweaver ❶ (siehe Abbildung 3.23), keine Seite oder die letzte geöffnete Seite beim Programmstart anzeigen lassen wollen, ist letztlich eine Geschmacksfrage. Bei Arbeiten an größeren Projekten erspart die letztere Einstellung aber das ständige Schließen der Dateien bei Arbeitsende und das neue Öffnen zu Beginn der Arbeiten am nächsten Tag.

Weiter vorn in diesem Kapitel haben wir Ihnen den Umgang mit »zugehörigen Dateien« erklärt. Hier in den Voreinstellungen können Sie entscheiden, ob Sie damit arbeiten möchten oder nicht ❷.

Nicht von Ihrem Geschmack abhängig ist jedoch die Einstellung zum Aktualisieren von Hyperlinks ❸ beim Verschieben von Dateien. Wenn Sie diese Funktion ausschalten, wird eine der wichtigsten Site-Verwaltungsfunktionen deaktiviert. Sinnvoll ist es, diese Funktion auf Nachfragen einzustellen. Es kommt mitunter vor, dass Sie Dateien zwischenzeitlich verschieben wollen, um sie zu einem späteren Zeitpunkt wieder an den alten Ort zu versetzen. Dann können Sie im Dialog Nachfragen auf Nein klicken, und die alten Pfadangaben bleiben erhalten.

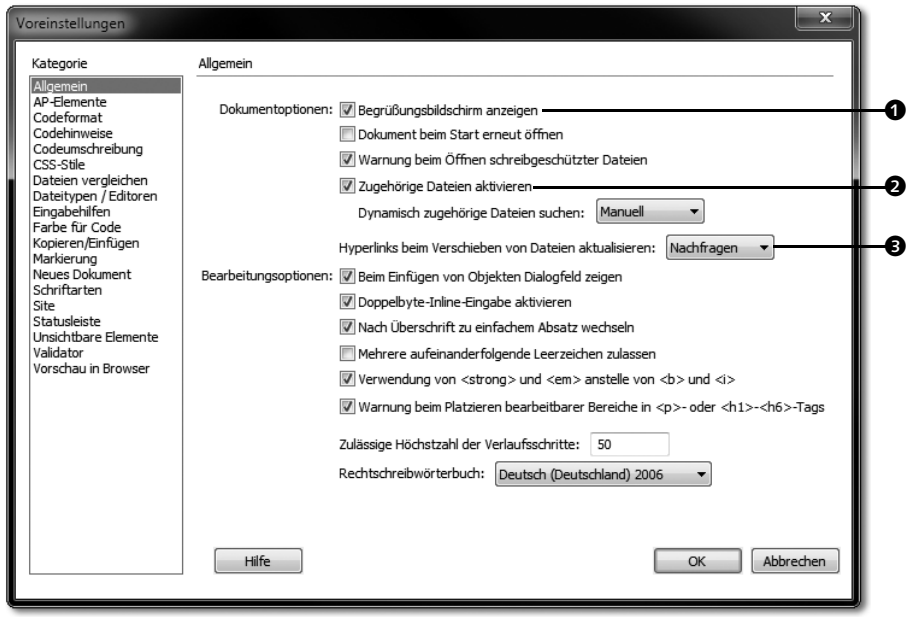


Abbildung 3.23 Allgemeine Voreinstellungen

3.2.2 AP-Elemente (bis Version 8.0: Layer)

Das Voreinstellungsmenü AP-Elemente ist besonders wichtig, wenn Sie mit CSS-Positionierungen im Layoutmodus arbeiten möchten. AP-Elemente (*absolut positionierte* Elemente) sind frei positionierbare Bereiche innerhalb eines HTML-Dokuments. Durch die Voreinstellung Verschachteln ❹ legen Sie fest, ob AP-Elemente unabhängig erstellt oder – wie es für ein CSS-Layout notwendig ist – verschachtelt werden.

Wir erläutern dieses Thema detailliert in Kapitel 11, »CSS in Dreamweaver«.

[»] **Bezeichnungsänderung: AP-Elemente = Ebenen**

Bis zur Version CS4 wurden AP-Elemente als *Layer* oder *Ebenen* bezeichnet. Wenn Sie also mit einer der früheren Dreamweaver-Versionen arbeiten, finden Sie die gleichen Einstellungen unter der Bezeichnung Ebenen.

Andere Entwickler reden gerne von *DIV-Elementen*, die absolut positioniert sind. Das ist aber nur ein anderer Ausdruck für die gleiche Sache.

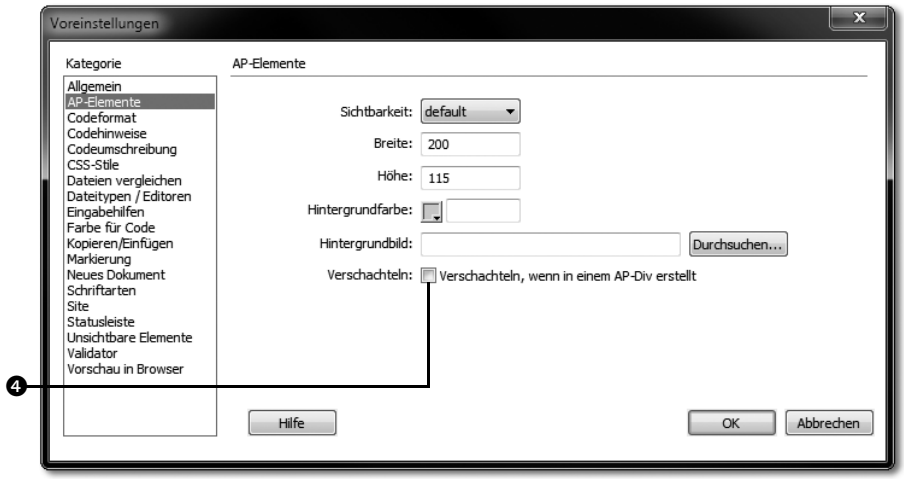


Abbildung 3.24 Voreinstellungen für AP-Elemente bzw. Ebenen

3.2.3 Codeformat einstellen

Die in diesem Menü angebotenen Einstellungen beziehen sich ausschließlich auf die Schreibweise des Quelltextes.

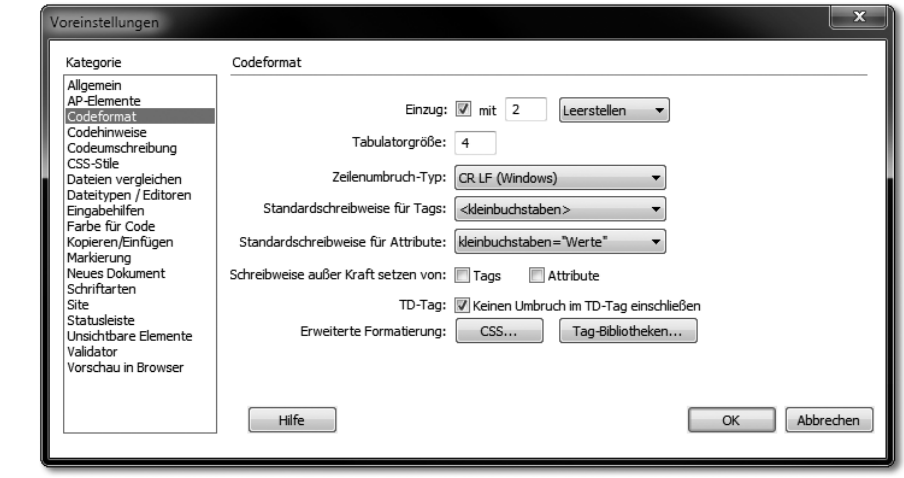


Abbildung 3.25 Codeformat einstellen

Es kommt vor, dass bei Zeilenumbrüchen innerhalb einer Tabellenzelle im Layout bzw. im Browser Randabstände ausgegeben werden, die gar nicht existieren. Um diesen Fehler zu vermeiden, sollten Sie grundsätzlich versuchen, innerhalb eines `<td>`-Tags (das ist das öffnende Tag einer Tabellenzelle) keinen Umbruch auszuführen.

Hinter der Schaltfläche CSS... verbergen sich weitere Formatierungsoptionen für CSS-Schreibweisen – so zum Beispiel, mit wie vielen Tabulatoren Einrückungen geschrieben werden. Auf die Schaltfläche TAG-BIBLIOTHEKEN kommen wir in den Abschnitten 17.3.3, »Eigene Tag-Vorgaben oder -Bibliotheken«, und 30.4.1, »DTDs mit Dreamweaver entwickeln«, noch genauer zu sprechen.

3.2.4 Codehinweise

Wenn Sie direkt im Quelltext arbeiten, ist es eine große Hilfe, Tags automatisch vervollständigen zu lassen. In Dreamweaver können Sie auswählen, wie dies geschehen soll.

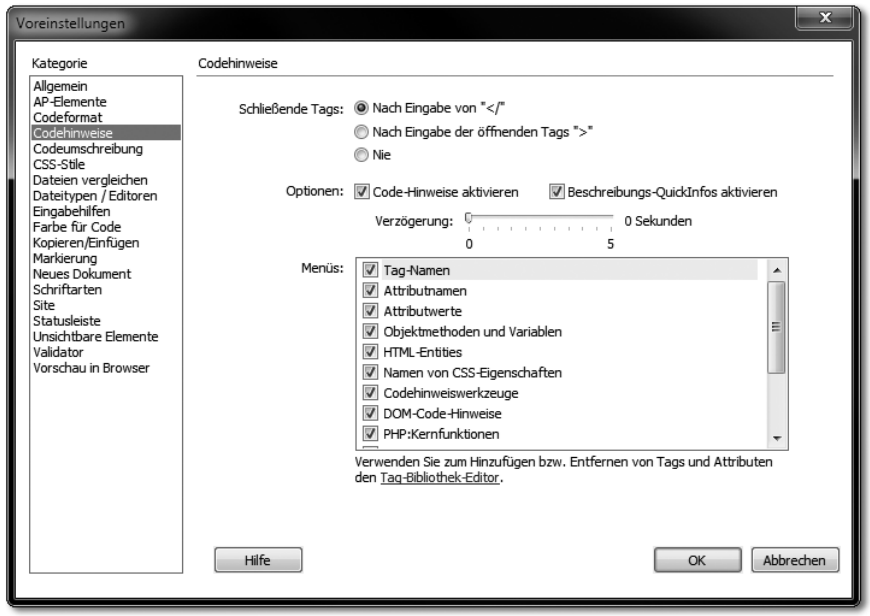


Abbildung 3.26 Codehinweise aktivieren

Soll das Tag nach Eingabe von </ geschlossen werden, erkennt Dreamweaver automatisch den letzten noch offenen Tag im Code und schließt diesen automatisch.

Wählen Sie das automatische Abschließen nach Eingabe von > aus, wird das Tag bereits nach der vollständigen Eingabe des öffnenden Tags abgeschlossen. Dies birgt jedoch die Gefahr, dass das bereits vorhandene Schluss-Tag übersehen bzw. vergessen wird und fehlerhafter Quellcode geschrieben wird. Sie können natürlich auch jegliche automatische Vervollständigung unterbinden.

3.2.5 Codeumschreibung

Grundsätzlich können wir nicht empfehlen, Tags in irgendeiner Weise von einem Editor automatisch reparieren zu lassen. Wenn Sie dynamische Websites erstellen und Code von Hand einfügen, ist es völlig normal, dass Tags – da diese ja dynamisch erzeugt werden – scheinbar nicht korrekt geschlossen oder fehlerhaft verschachtelt sind. Diese Tags dürfen in keinem Fall automatisch geändert werden.

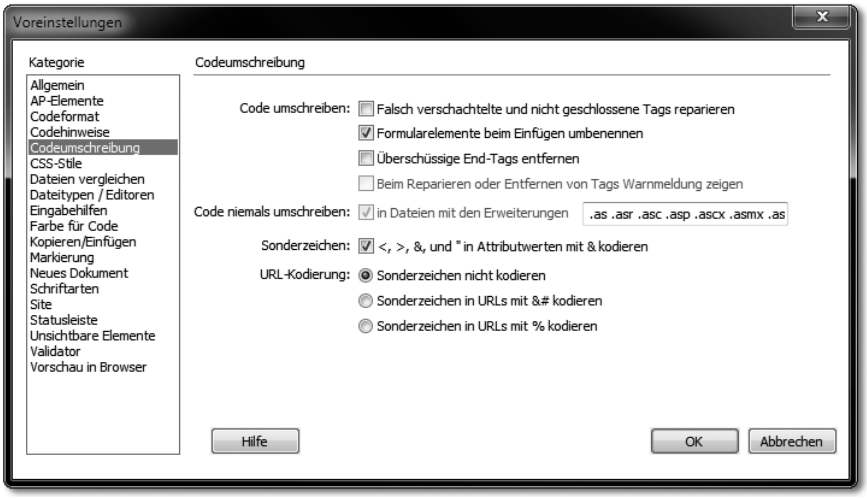


Abbildung 3.27 Optionen der Codeumschreibung festlegen

Ähnlich ist es mit Umcodierungen von Sonderzeichen. Dreamweaver bietet zwar die Möglichkeit, bestimmte Dateitypen von der Umschreibung auszuschließen, erfahrungsgemäß kann man sich hierauf jedoch nicht verlassen.

Fehlerquelle: Kopieren von Quelltext [!]

Wenn Sie Quelltexte aus anderen Anwendungen kopieren, kommt es mitunter vor, dass Dreamweaver bei falschen Einstellungen PHP-Befehle mit Sonderzeichen umcodiert. Überprüfen Sie in jedem Fall den eingefügten Code gründlich!

Generell sollten PHP und andere Skriptsprachen auch nur in der Codeansicht kopiert und nicht in der Layoutansicht eingefügt werden, um diese Problematik zu reduzieren.

3.2.6 CSS-Stile

CSS-Stile können in einer *Kurzschreibweise* geschrieben werden. Diese erlaubt es, Attribute in einer Zeile zusammenzufassen.

```
.kurz {
   font: 11px/15px Arial, Helvetica, sans-serif;
}

._lang {
   font-family: Arial, Helvetica, sans-serif;
   font-size: 11px;
   line-height: 16px;
}
```

Listing 3.1 CSS in Kurz- und (längerer) Normalschreibweise

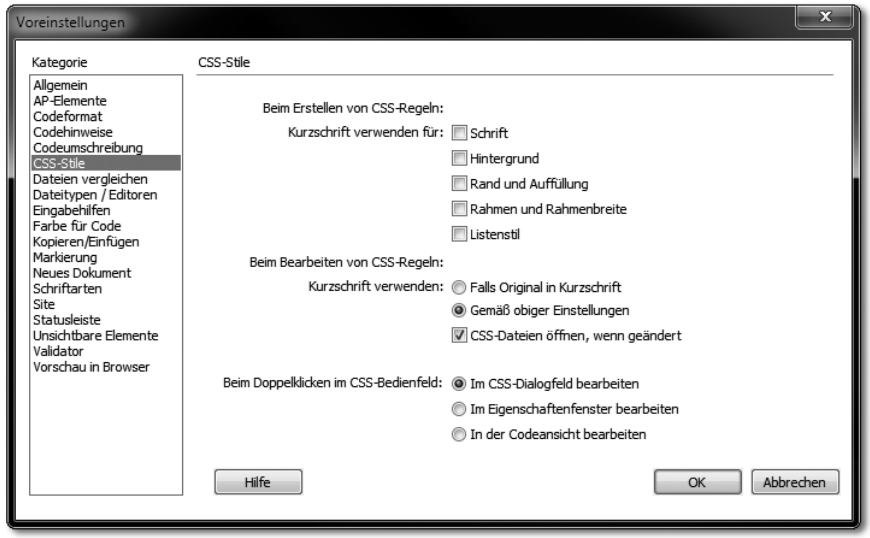


Abbildung 3.28 Regeln für CSS-Stile anlegen

Mit dieser Technik soll auch noch das letzte Byte an Dateigröße eingespart werden. Der CSS-Code wird durch die Kurzschreibweise allerdings nicht einfacher zu lesen. Wie Sie in Listing 3.1 sehen, werden sowohl Schrifthöhe als auch Zeilenabstand einfach als Pixelwerte angegeben, ohne eine genauere Bezeichnung der jeweiligen Funktion. Das kann nicht nur für Anfänger sehr verwirrend sein. Wenn Sie dennoch möchten, dass Ihre CSS-Stile in Kurzschreibweise verfasst werden, aktivieren Sie die entsprechende Option in dieser Dialogbox.

Da CSS-Stile häufig in externe Dateien ausgelagert werden, bietet Ihnen Dreamweaver hier die Möglichkeit, diese externe Datei auch gleich beim Bearbeiten eines Stils zu öffnen. Auch die Art der CSS-Bearbeitung ist hier ganz an Ihre Wünsche und Vorstellungen anpassbar.

3.2.7 Dateien vergleichen

Sicherlich haben Sie Folgendes bereits in einem Projekt erlebt: Es gibt mehrere Versionen eines Dokuments, und Sie wissen nicht, worin die Unterschiede zwischen den Dokumenten bestehen.

Hier kann Abhilfe geschaffen werden. Dreamweaver ist in der Lage, eine externe Anwendung zum Dateivergleich einzubinden und auf Anforderung einen Dateivergleich mit dieser Anwendung durchzuführen.



Abbildung 3.29 Anwendung für Dateivergleiche anlegen

Eines der dafür in Frage kommenden Programme ist das kostenlose *WinMerge*, das wir mit auf die DVD im Buch gepackt haben. Sie finden es im Ordner *Tools*. Hier werden die zu vergleichenden Dateien nebeneinander dargestellt und die unterschiedlichen Abschnitte hervorgehoben. [o]

Markieren Sie dazu eine Datei in der lokalen Ansicht im Dateifenster, und wählen Sie mit der rechten Maustaste im Kontextmenü Mit Remote-Datei vergleichen – beide Dateien werden in dem entsprechend eingebundenen Programm geöffnet. Markierte Abschnitt können Sie dann mit einem Klick in die eine oder andere Datei kopieren.

3.2.8 Dateitypen/Editoren

Legen Sie Menüpunkt Dateitypen/Editoren die Editoren für externe Dateien fest, wie zum Beispiel Bilder. Für Grafiken ist Photoshop voreingestellt. Nach der Auswahl eines Bildes in der Dokumentansicht bewirkt ein Klick auf Bearbeiten in der Eigenschaftenpalette, dass die eingebundene Datei im voreingestellten Editor geöffnet wird.

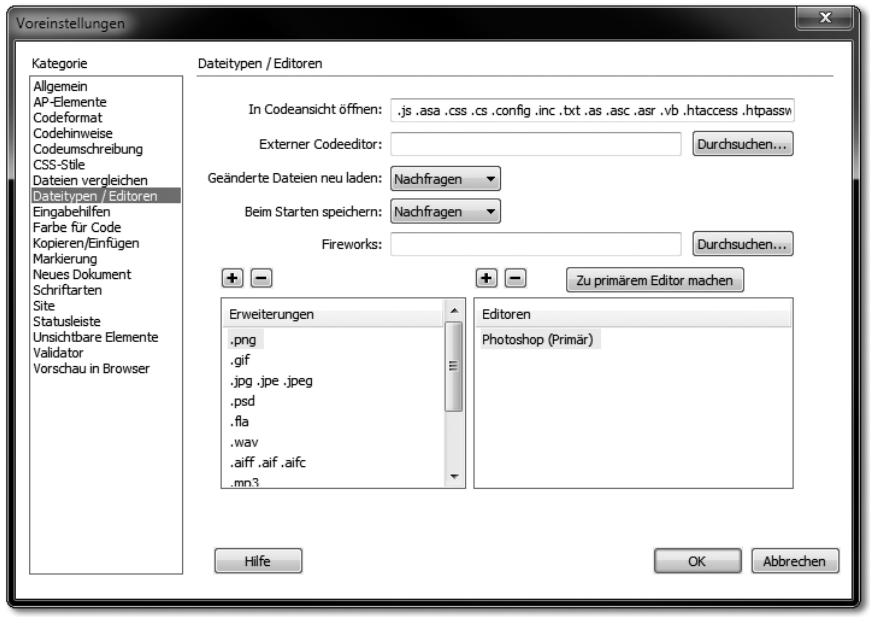


Abbildung 3.30 Externe Editoren festlegen

3.2.9 Eingabehilfen

Dreamweaver bietet Ihnen beim Anlegen der in dieser Dialogbox aufgelisteten Elemente eine Eingabehilfe an. Wenn Sie diese aktivieren, erscheint beim Einfügen im Dokumentfenster eine Dialogbox, die von Ihnen ähnlich wie bei einem Assistenten verschiedene Angaben erwartet. Sie können die Funktion testen, indem Sie die Kontrollkästchen aktivieren und eines der Elemente in einem Dokument platzieren.

Wir arbeiten im gesamten Buch allerdings *ohne* Eingabehilfen und gehen auf diese Dialogboxen auch nicht ein, da ohne die Eingabehilfen deutlich schneller gearbeitet werden kann. Der Inhalt der Dialogboxen ist zudem selbsterklärend, wenn Sie die Funktionen anwenden, wie wir sie Ihnen erläutern.

[!] **Eingabehilfen deaktivieren**

Aufgrund der Geschwindigkeitseinbußen empfehlen wir Ihnen, während der Arbeit mit den Beispielen dieses Buches alle Eingabehilfen zu deaktivieren.

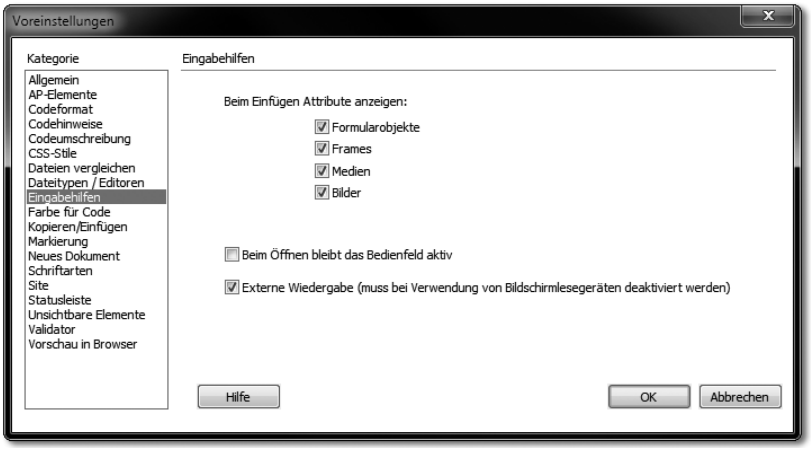


Abbildung 3.31 Festlegen der Eingabehilfen

3.2.10 Farbe für Code

Hier können Sie Ihre eigenen Farben für die Codeansicht einstellen. Das ist im Allgemeinen nicht nötig, da die Farbschemata gut gestaltet sind. Für Anwender mit Farbfehlsichtigkeit ist dies jedoch hilfreich.

3.2.11 Kopieren und Einfügen (Office)

Dreamweaver kann beim Einfügen von Dokumenten die bestehenden Formatierungen mit übernehmen. Besonders interessant ist hier das Einfügen aus Office-Dokumenten. Aus diesem Grund haben wir diesem Thema ein eigenes Kapitel gewidmet (Kapitel 16, »Office-Dateien einfügen«).

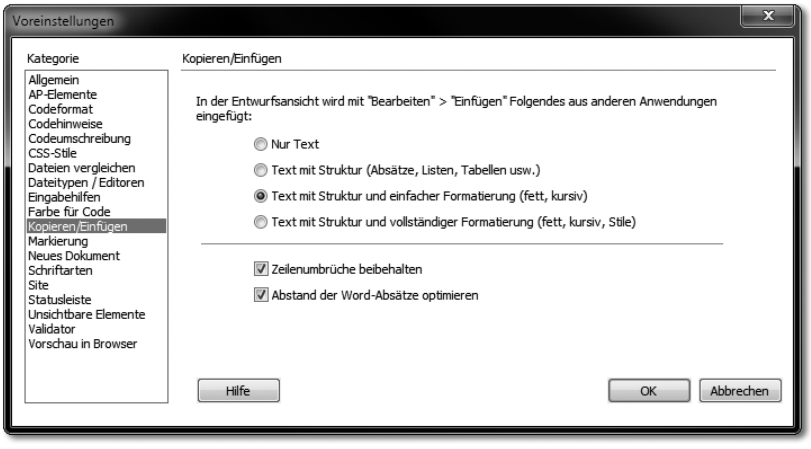


Abbildung 3.32 Einstellungen für Einfügen und Kopieren

3.2.12 Markierung

Ähnlich wie die Farbe für den Code sind hier im Allgemeinen keine Änderungen nötig. Dennoch ist es hilfreich, ab und zu eine andere Farbe anzuwählen, wenn Sie z. B. dynamischen Spry-Code auf dunklem Hintergrund einfügen. Die Farbgebung erschwert dann die Sichtbarkeit erheblich und könnte mit dieser Voreinstellung bequemer gestaltet werden.

Wohlgemerkt, es handelt sich hier nur um Darstellungen der Seite in der Layoutansicht von Dreamweaver, die Farben sind im Browser selbstverständlich ohne Bedeutung.

3.2.13 Neues Dokument

Die verschiedenen Dokumenttyp-Definitionen (DTD) im Detail zu erläutern würde den Rahmen dieses Buches sprengen und auch reichlich Nahrung für endlose Diskussionen liefern. Dreamweaver CS5 unterstützt alle verfügbaren Standards.

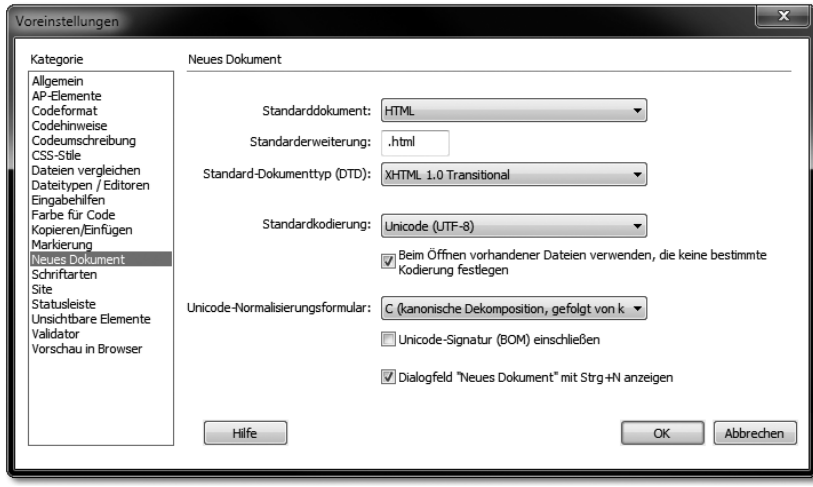


Abbildung 3.33 Vorgaben für neue Dokumente

XHTML und CSS als Standard

Als Standard für Websites wurde HTML von XHTML in Kombination mit CSS für das Layout abgelöst. Neu zu erstellende Websites sollten Sie also möglichst als XHTML-1.0- und CSS-Dokumente anlegen. Das heißt jedoch nicht, dass HTML von den Browsern bald nicht mehr unterstützt wird. Welche Rolle das neue HTML5 in Zukunft spielen wird, können wir zu diesem Zeitpunkt noch nicht sagen.

Alternative HTML 4.0 Transitional

Möchten Sie die Möglichkeiten browserspezifischer Features flexibel einsetzen, empfiehlt sich ein weniger reglementierter Dokumenttyp wie HTML 4.0 Transitional.

XHTML

XHTML ist eine Sprachvariante von XML, die die HTML-Befehle abbildet. HTML stammt von SGML ab. Faktisch sind HTML und XHTML komplett unterschiedliche Sprachen. Da allerdings die Befehle fast alle gleich sind, wird (fälschlicherweise) oft davon gesprochen, dass XHTML die neue Version von HTML sei.

Korrekt ist, dass sich praktisch alle HTML-Befehle in XHTML wiederfinden. Allerdings ist XHTML durch die strengere Definition der XML-Sprachfamilie sauberer gestaltet. So ist die Schreibweise von Tags und Attributen nun ganz strikt geregelt. Ein Beispiel stellt der folgende kurze Codeausschnitt dar.

HTML (mit allen möglichen schlechten Eigenschaften, aber funktionstüchtig):

```
<IMG src=bild.jpg WIDTH='100' Height="200">
```

XHTML (sauber und korrekt):

```
<img src="bild.jpg" width="100" height="200" alt="Bild" />
```

Die Unterschiede in XHTML sind klein, aber wichtig:

- Tags und Attribute *müssen* kleingeschrieben werden.
- Attributwerte müssen in *doppelten* Anführungszeichen stehen.
- Einige Attribute sind Pflichtangaben (z. B. das `alt`-Attribut im `<img>`-Tag)
- Alle Tags müssen *geschlossen* werden. Tags, die nur aus einer Tag-Hälfte bestehen, wie z. B. `<img>` oder `<br>`, werden »in sich selbst« geschlossen, d. h. mit einem Schrägstrich `/` vor der schließenden spitzen Klammer.

XHTML 1.0 ist auf die Kompatibilität zu den HTML-Varianten angelegt und kann auch von Browsern, die XHTML nicht unterstützen, problemlos angezeigt werden. Für diese Fälle wird die Toleranz der Browser gegenüber Syntaxfehlern ausgenutzt. XHTML-eigene Elemente werden einfach ignoriert. Diese Einstellung ist heute die günstigste Wahl. Allgemein können aber alle aktuellen und etwas älteren Browser XHTML ohne Probleme darstellen.

XHTML 1.1 ist wesentlich strenger aufgebaut und modular angelegt. Dies ermöglicht eigene Erweiterungen durch XML-Strukturen. XHTML 1.1 ist jedoch nicht auf Browser-Kompatibilität ausgelegt und unterstützt diverse HTML-Elemente nicht.

[»] **Dateien nachträglich konvertieren**

Wenn Sie in Ihrem Projekt älteren Code haben, aber komplett auf XHTML umstellen möchten oder Codeteile aus älteren HTML-Dateien in neuen XHTML-Code eingefügt haben, können Sie mit dem Menübefehl Datei • Konvertieren den alten Code in den neuen umwandeln lassen.

Unicode und Entities

HTML-Entities dienen der Darstellung von Sonderzeichen in einer codierten Form. Das »Ü« zum Beispiel finden Sie im Quelltext allgemein als `Ü` wieder. Diese Codierungen dienen dem Browser – in Kombination mit dem im Dokument angegebenen Zeichensatz – zur Festlegung der Zeichendarstellung. Lesen Sie dazu auch Kapitel 5, »Grundlegende Dokumenteinstellungen«.

Zeichencodierungen

Unicode ist ein internationaler Standard, in dem langfristig für jedes Zeichen aller bekannten Schriftkulturen und Zeichensysteme ein digitaler Code festgelegt wird, um den Austausch von Dokumenten auf internationaler Ebene zu vereinfachen.

Bei den bisherigen Zeichencodierungen wie ASCII können nur wenige Sprachen gleichzeitig im selben Text dargestellt werden, es sei denn, man verwendet innerhalb des Textes verschiedene Zeichensätze. Gerade im Internet behindert dies die internationale Kommunikation erheblich.

Darstellung im Browser

Aktuelle Webbrowser wickeln die Darstellung von Unicode-Zeichensätzen perfekt und vom Benutzer unbemerkt ab. Voraussetzung dafür ist jedoch, dass auf dem Zielsystem eine Unicode-Schrift installiert ist und unter Windows die internationale Sprachunterstützung aktiviert wurde. Klassische Computerschriftarten sind dazu nicht geeignet, da sie sich an bestimmten Zeichensätzen orientieren. Neue, Unicode-orientierte Schriftarten (OpenType) verbreiten sich erst allmählich.

In XML-Dokumenten und bei ASP-Programmierungen wird grundsätzlich in Unicode gearbeitet, um Plattformunabhängigkeit zu erreichen.

[»] **Weitere Informationsquellen**

Weiterführende Informationen finden Sie im SelfHTML-Kompendium von Stefan Münz unter *http://de.selfhtml.org/* oder auf der Website des Unicode-Konsortiums unter *http://www.unicode.org*.

3.2.14 Site

Wenn Sie in einem Firmennetzwerk arbeiten, existiert eventuell eine Firewall, um Ihr Intranet vor Angriffen aus dem Internet zu schützen. Wenden Sie sich in diesem Fall an Ihren Administrator, er wird Ihnen bei der Einstellung helfen.

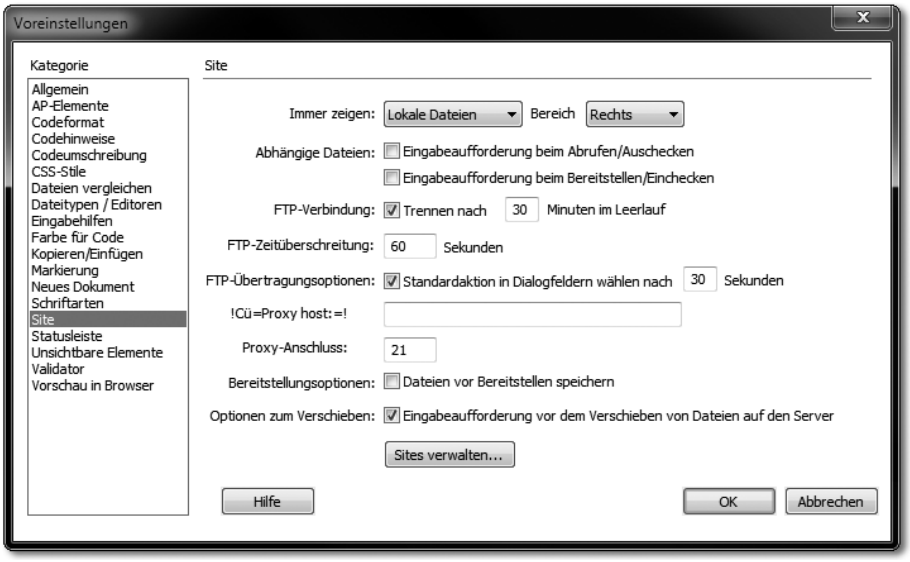


Abbildung 3.34 Site-Vorgaben und Firewall-Einstellungen

3.2.15 Statusleiste

Wenn Sie Websites erstellen, werden Sie sicherlich nicht an einem kleinen Office-Monitor sitzen, sondern mit einem Monitor ab 19" arbeiten. Damit Sie auch kleinere Bildschirmgrößen nachbilden können, verfügt Dreamweaver über eine automatische Anpassung der Fenstergröße. Diese Werte können Sie in der Kategorie STATUSLEISTE einstellen (siehe Abbildung 3.35).

3.2.16 Unsichtbare Elemente

Wie weiter oben beschrieben, werden in der Layoutansicht diverse unsichtbare Elemente durch sogenannte *Platzhalter* angezeigt. Hier können Sie angeben, welche der unsichtbaren Elemente Sie sehen wollen und welche nicht (siehe Abbildung 3.36).

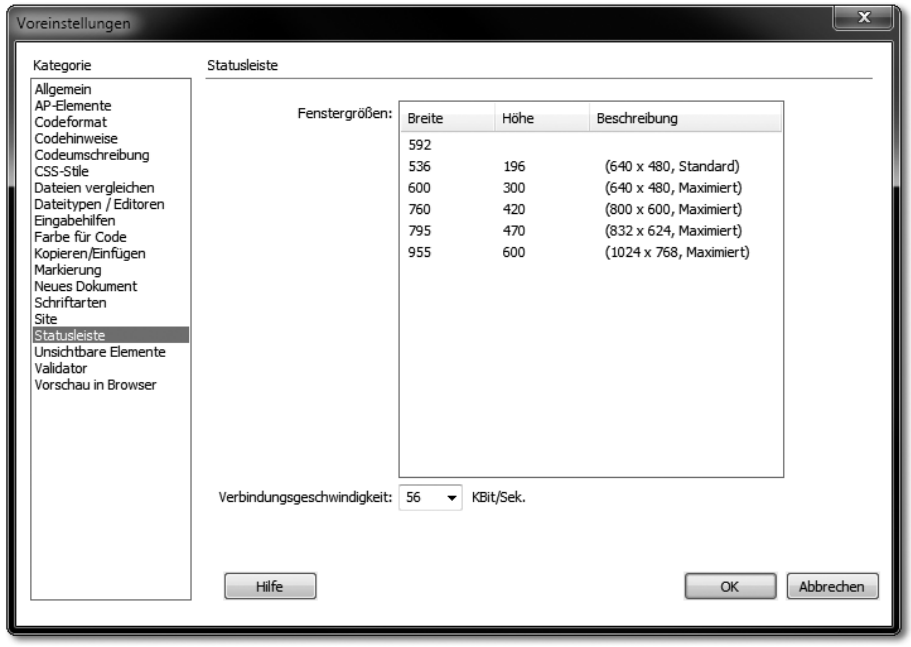


Abbildung 3.35 Einstellungen für kleine Auflösungen

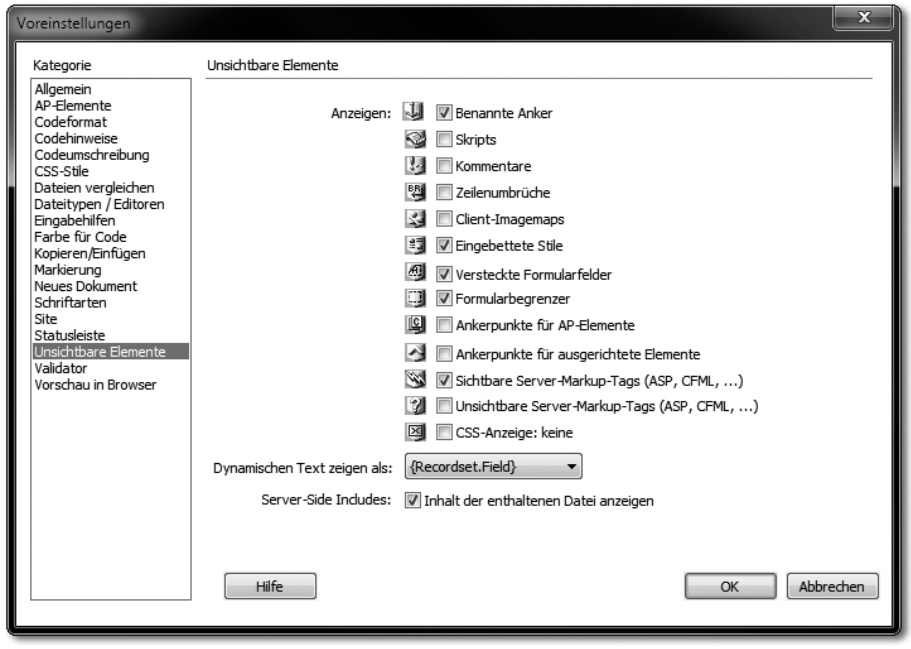


Abbildung 3.36 Anzeige unsichtbarer Elemente

3.2.17 Validator

Der Validator ist das in Dreamweaver integrierte Werkzeug zur Gültigkeitsprüfung der von Ihnen erstellten Dokumente. Stellen Sie hier die gleiche Sprachversion ein, die Sie auch beim Dokumenttyp angegeben haben. Besonders beim Arbeiten in XHTML sollten Sie hier noch einmal einen Blick hineinwerfen, denn diese Einstellung ändert sich nicht gleichzeitig mit.

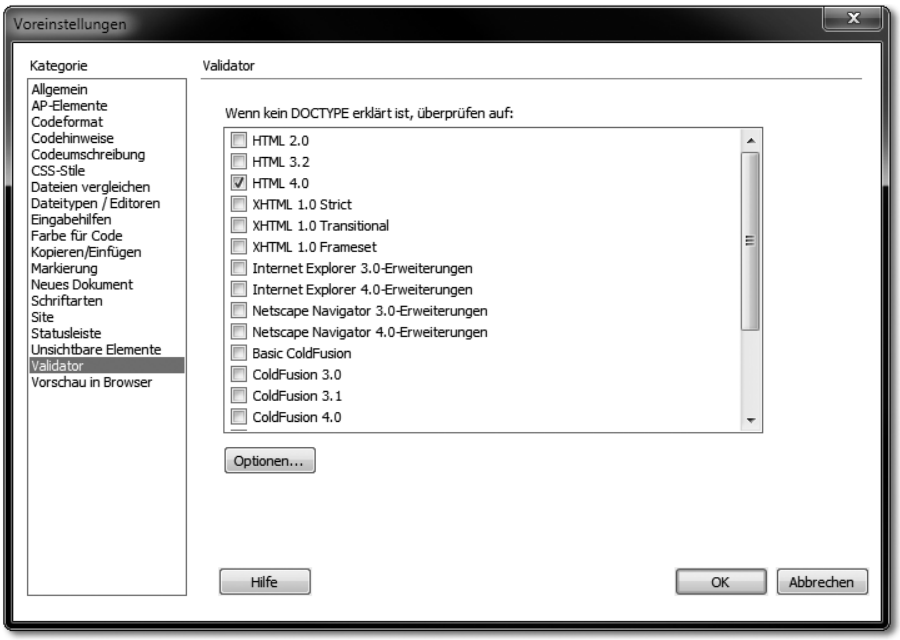


Abbildung 3.37 Überprüfung von Syntax und Kompatibilität

3.2.18 Vorschau im Browser

Die integrierte Vorschaufunktion von Dreamweaver ist sehr gut. Dennoch kann die beste Software nicht alle vorhandenen Browser abdecken. Es ist deshalb immer wieder sinnvoll, die erstellten HTML-Dokumente in verschiedenen Zielbrowsern zu überprüfen. Sie können in Dreamweaver beliebig viele Browser zur Voransicht definieren.

Verschiedene Browser

Aktuell empfehlen wir Ihnen, die Überprüfung im Internet Explorer und mit Mozilla (Firefox) durchzuführen. Weitere aktuelle Browser wie Opera und Safari sollten möglichst in die Überprüfung mit einbezogen werden.

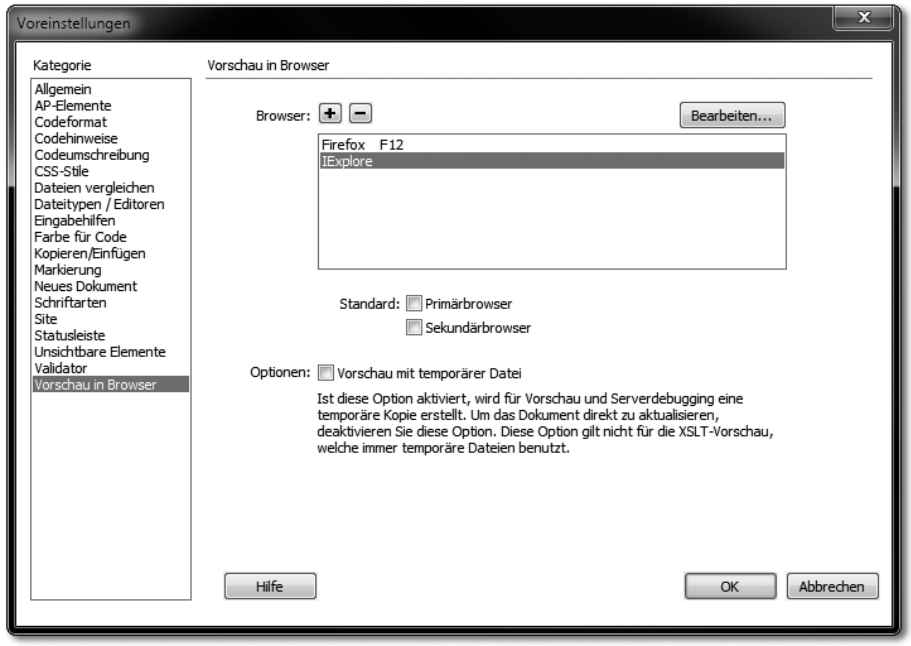


Abbildung 3.38 Auswahl verschiedener Zielbrowser

Arbeiten am Mac

Entwicklern auf Mac OS X empfehlen wir dringend, die erstellte Website ebenfalls auf einem PC mit dem Internet Explorer zu überprüfen. Die Darstellungen von Farben und Schrift sowie die Umsetzung von CSS-Regeln sind definitiv nicht identisch, wenn auch die Unterschiede mittlerweile gering sind. Die Internet-Explorer-Versionen für den Mac entsprechen nicht den PC-Versionen.

[»] **Aktuelle Dokumentversionen anzeigen**

Wenn Sie HTML-Dokumente über die Aktualisieren-Funktion im Browser erneut aufrufen, erhalten Sie bei der Voreinstellung Vorschau mit temporärer Datei eine falsche Ansicht. Dreamweaver legt für die Browservorschau bei dieser Einstellung temporäre Dateien an und zeigt diese im Browser statt der Originaldateien.

Wenn Sie im Browser auf Aktualisieren klicken, sehen Sie daher eine veraltete Version der Seiten. Ob Ihnen temporäre Dateien angezeigt werden, erkennen Sie an Namen wie *TMPyfdj0jbg6y.php* in der Adressleiste des Browsers.

Mit dem Tastaturkürzel Strg+F5 unter Windows und Alt+R in Safari auf dem Mac können Sie eine Seite unter Umgehung des Caches neu laden. Der Browser aktualisiert dann auch die unsichtbaren CSS-Dateien.

Browserliste bearbeiten

Um weitere Browser für die Browservorschau zu bestimmen, wählen Sie im Menü BEARBEITEN • VOREINSTELLUNGEN (oder Strg/⌘+U) und dort VORSCHAU IM BROWSER. Klicken Sie auf das Pluszeichen ❶, und geben Sie den Namen des zusätzlichen Browsers ein. Wählen Sie jetzt das Browserprogramm auf Ihrer Festplatte aus.

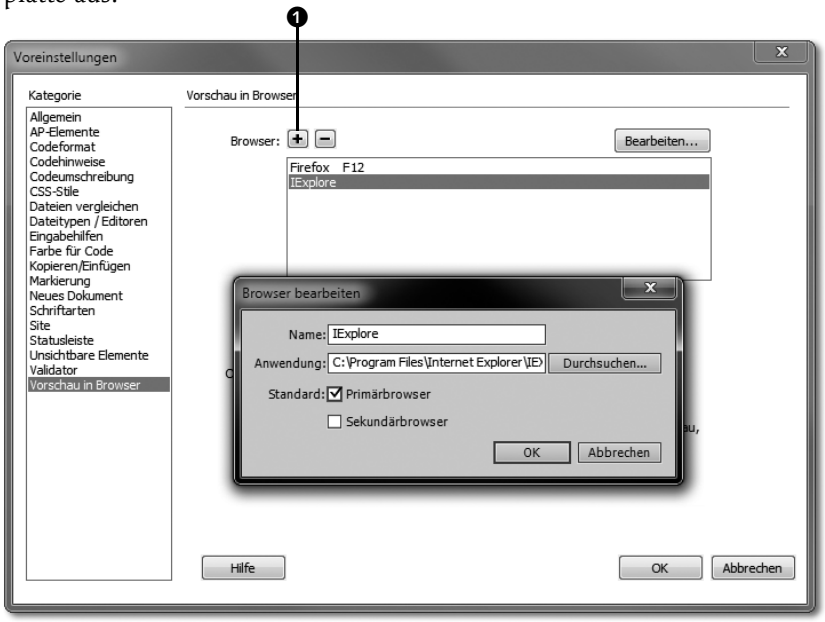


Abbildung 3.39 Weitere Browser für die Vorschau hinzufügen

Wenn Sie mehrere Zielbrowser ausgewählt haben, erscheinen diese in der Auswahl bei der Browservorschau. Mit den Einstellungen PRIMÄR- UND SEKUNDÄRBROWSER können Sie Standardbrowser für die Voransicht festlegen, die sich mit den Tasten F12 bzw. Alt+F12 am Mac und Strg/⌘+F12 öffnen lassen.

Mehrere Internet-Explorer-Versionen [«]

Sie können die meisten Browser ohne Probleme parallel auf Ihrem System installieren. Mehrere Internet-Explorer-Versionen parallel auf einem Rechner zu installieren ist allerdings nicht ohne weiteres möglich. Dieser Browser überschreibt bei einer erneuten Installation alle Dateien – auch dann, wenn Sie eine alte über eine neuere Version installieren wollen. Abhilfe schafft hier ein kleines Tool, das Sie auf folgender Website finden: *http://tredosoft.com/Multiple_IE*

Wir können Ihnen aus schmerzlicher Erfahrung heraus nur abraten, es ohne entsprechende Tools zu versuchen. Es ist tatsächlich so, dass der Internet Explorer massiv mit dem System verflochten ist. Wenn Sie Pech haben, funktioniert an-

schließend nichts mehr. Witzig wird es, wenn Sie den IE-Ordner versuchen zu löschen. Das geht wirklich, allerdings ist er fünf Sekunden später wie von Zauberhand inklusive aller Dateien wieder da.

3.2.19 Prüfung auf externen Rechnern

In diesem Zusammenhang ist es auch ratsam, eine Website nach dem Übertragen auf einen Testserver oder gar den Remoteserver mit einem anderen Computer (unabhängig von Betriebssystem oder Browser) zu testen. Bilder, die nämlich nicht mitübertragen wurden oder falsch benannt sind, werden zwar auf dem lokalen System korrekt angezeigt, weil sie ja lokal korrekt eingebunden wurden und gegebenenfalls noch im Browsercache liegen, auf einem anderen Rechner aber sind diese fehlenden Dateien nicht vorhanden. Erst jetzt wird Ihnen dies auffallen.

3.3 Layouthilfen in Dreamweaver

3.3.1 Raster einstellen und anzeigen

Das Arbeiten mit einem Dokumentraster kann beim Layouten mit AP-Elementen eine große Hilfe sein. Sinnvoll ist das Dokumentraster, wenn Sie AP-Elemente frei positionieren möchten. Im Menü Ansicht • Raster • Rastereinstellungen können Sie, wie in Abbildung 3.40 zu sehen ist, das Raster ganz an Ihre Wünsche anpassen.



Abbildung 3.40 Raster einstellen und anzeigen

Um das Raster zu aktivieren, klicken Sie im Menü auf Ansicht • Raster • Raster anzeigen oder drücken gleichzeitig Strg/⌘+Alt+G.

Wenn Sie möchten, dass Layoutelemente am Raster ausgerichtet werden, müssen Sie Raster • Am Raster ausrichten aktivieren oder Strg/⌘+A+Alt+G drücken. Sie können die Schrittweite bzw. den Rasterabstand in diesem Menü ebenfalls genau definieren.

3.3.2 Hilfslinien

Seit Dreamweaver 8 ist es möglich, im Dokumentfenster mit Hilfslinien zu arbeiten. Auch wir waren anfangs skeptisch, was denn Hilfslinien bei HTML zu suchen haben, dann aber schnell begeistert und möchten inzwischen dieses Tool zur schnellen Layoutkontrolle nicht mehr missen.

Hilfreiches Kontrollmittel

Es ist nicht sinnvoll, Hilfslinien zum vollständigen Aufbau des Layouts einzusetzen, wie Sie dies eventuell aus DTP-Programmen wie InDesign gewohnt sind. Das eigentliche Layout wird mit Photoshop o.Ä. erstellt.

Zur schnellen Kontrolle von Abmessungen und Abständen sind Hilfslinien hingegen hervorragend geeignet. Wer kennt das nicht: Screenshot erstellen – Photoshop öffnen – Abstand messen – in Dreamweaver nachbearbeiten – erneute Kontrolle usw. Diese Arbeitsschritte können Sie sich jetzt sparen.

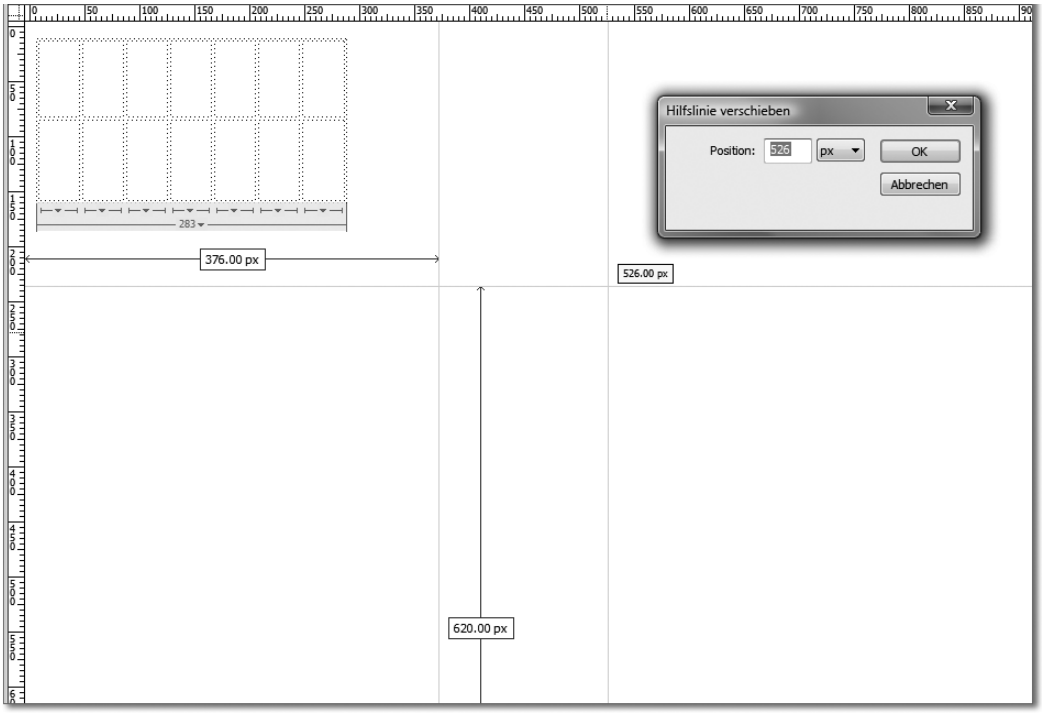


Abbildung 3.41 Hilfslinien erstellen

Um Hilfslinien zu erstellen, müssen Sie zunächst die Lineale aktivieren. Hierzu klicken Sie im Menü Ansicht auf Lineale und auf Lineale • zeigen oder drücken gleichzeitig Strg/⌘+Alt+R.

Erstellen Sie nun eine Hilfslinie, indem Sie auf die Lineale klicken und mit gedrückter Maustaste die Hilfslinie an die gewünschte Position in das Dokument ziehen (so, als wollten Sie das Lineal verschieben).

Um eine Hilfslinie wieder zu löschen, ziehen Sie diese einfach aus dem Dokument heraus.

Hilfslinien exakt einstellen

Ziehen Sie eine beliebige Hilfslinie in das Dokument, und klicken Sie doppelt auf die Hilfslinie. Wie in Abbildung 3.41 dargestellt, öffnet sich eine Dialogbox, und Sie können die gewünschte Position pixelgenau eingeben.

Abstände schnell messen

Um Abstände innerhalb eines Dokuments schnell zu messen, ziehen Sie eine beliebige Hilfslinie in das Dokument und verschieben diese mit gedrückter Strg/⌘-Taste. Alle Abstände zum Dokumentrand werden dann – zusätzlich zur aktuellen Hilfslinienposition – angezeigt (siehe Abbildung 3.41).

[»] **Browserfenster mit Hilfslinien simulieren**

Im Menü Ansicht • Hilfslinien werden diverse Bildschirmabmessungen angezeigt. Die Auswahl einer dieser Abmessungen veranlasst Dreamweaver dazu, Hilfslinien in den Abmessungen des in der ausgewählten Bildschirmgröße zur Verfügung stehenden Browserfensters anzulegen.

3.3.3 Die Statusleiste

Das Icon mit dem Pfeil ist die Standardauswahl und ermöglicht Ihnen das Auswählen einzelner Elemente im Dokument. Dieses Icon müssen Sie anwählen, nachdem Sie mit dem Handwerkzeug oder dem Zoomwerkzeug die Dokumentansicht verändert haben.

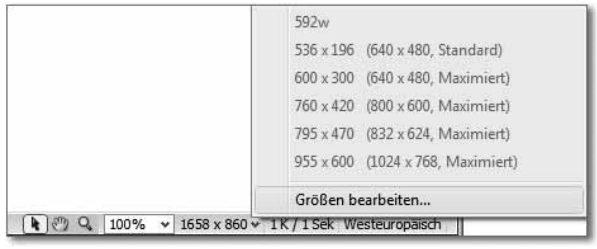


Abbildung 3.42 Neue Hilfsmittel in der Statusleiste

Das Handwerkzeug

Mit der Hand können Sie ein Dokument »anfassen« und innerhalb des aktuellen Fensterbereichs am Bildschirm verschieben, ohne mit den Scrollleisten zu arbeiten. Besonders in Kombination mit der Lupe und einem gezoomten Dokument ist dieses Werkzeug sehr sinnvoll, um eine Stelle im Dokument zu finden.

Das Zoomwerkzeug

Ein Dokument können Sie, wie in Abbildung 3.43 zu sehen ist, mit der Lupe oder durch die direkte Anwahl einer voreingestellten Skalierung im Menü zoomen. In diesem Menü gelangen Sie auch wieder zur Originalansicht des Dokuments (100 %).

Bildschirmauflösung & Dateigröße

Im Menü können Sie rechts neben der Zoomfunktion das aktuelle Dokumentfenster auf die in einer bestimmten Bildschirmauflösung zur Verfügung stehenden Abmessungen skalieren. Am rechten Rand der Statusleiste finden Sie die Angaben zur aktuellen Dateigröße des Dokuments. In dieser Angabe sind alle externen Grafiken usw. enthalten. Sie können auf diese Weise abschätzen, ob das Dokument noch in einer akzeptablen Geschwindigkeit über das Internet geladen werden kann.

Dokumentfenster auf Bildschirmauflösung skalieren [«]

Das Dokumentfenster kann nur auf die gewünschte Größe skaliert werden, wenn es nicht maximiert ist.

Seitencodierung

Ganz rechts am Rand wird Ihnen zudem die verwendete Zeichencodierung im Dokument angezeigt.



Abbildung 3.43 Ein gezoomtes HTML-Dokument

3.3.4 Tracing-Bilder

Um ein Layout exakt an einer Vorlage auszurichten, können Sie in den Seiteneigenschaften ein sogenanntes *Tracing*-Bild angeben. Die Seiteneigenschaften erreichen Sie über das Menü MODIFIZIEREN • SEITENEIGENSCHAFTEN.

Ein Tracing-Bild ist eine Grafik, die in den Hintergrund des aktuellen Bildes gelegt wird, um darauf Raster, Hilfslinien, Ebenen oder Tabellen zu erstellen. Das Tracing-Bild wird im Browser nicht angezeigt. Damit eine Unterscheidung zwischen Tracing-Bild und bereits im Layout eingefügten grafischen Elementen möglich ist, können Sie die Transparenz des Tracing-Bildes einstellen.

3.4 Dreamweaver erweitern

Dreamweaver ist ein mächtiges Werkzeug. Doch damit nicht genug, es lässt sich auch noch nahezu beliebig erweitern.

3.4.1 Befehle speichern

Wie fast alle Adobe-Programme verfügt auch Dreamweaver über einen Verlauf. Bei Photoshop entspricht das Protokoll dem Verlauf in Dreamweaver. Dieses Aktionsprotokoll zeichnet Ihre Arbeitsschritte auf und ermöglicht es Ihnen, auf einfache Weise zu jedem beliebigen Schritt Ihrer Arbeit zurückzukehren.

Das Verlaufsfenster öffnen Sie über FENSTER • VERLAUF oder durch gleichzeitiges Drücken von ⇧+F10.

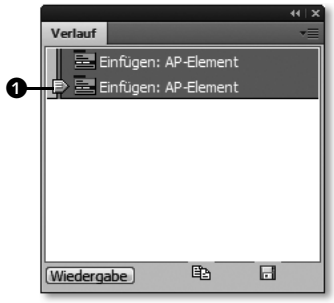


Abbildung 3.44 Das Verlaufsfenster

Durch Bewegen des Schiebereglers ❶ auf der linken Seite des Bedienfeldes können Sie zu jedem Schritt in Ihrer Arbeitsfolge zurückkehren oder bereits gelöschte Schritte wiederherstellen.

Mit Tastaturbefehlen [«]

Natürlich funktionieren auch das Menü BEARBEITEN • RÜCKGÄNGIG und der Tastaturbefehl Strg+Z, um einen oder mehrere Schritte nacheinander rückgängig zu machen. Auch dem Menü BEARBEITEN • WIEDERHERSTELLEN entspricht ein Tastaturkürzel: Strg+Y.

Kein Verlass [!]

Verlassen Sie sich bitte nicht darauf, dass Sie immer zu den letzten Schritten zurückkehren können. Der Speicher mit den letzten Arbeitsschritten wird ab und an geleert. Vor allem, wenn Sie eine Datei als »abhängige Datei« bearbeitet haben, kommt es vor, dass man nicht mehr zurück kann.

Es gibt immer wieder Arbeitsschritte in der Entwicklung einer Website, die auf jeder Unterseite gleich sind, zum Beispiel das Einstellen der Seiteneigenschaften, das Einfügen der Metaangaben usw. Sie können die einzelnen Schritte im Verlauf auswählen und als Befehl speichern.

Markieren Sie dazu die gewünschten Schritte mit dem Schieberegler ❶. Wenn Sie auf WIEDERGABE ❷ klicken, können die markierten Befehle wiederholt werden. Bei dem Icon ❹ werden die Arbeitsschritte in die Zwischenablage kopiert, durch Klicken auf das Disketten-Icon ❺ können Sie die Arbeitsfolge im sich öffnenden Dialogfenster als Befehl speichern ❸.

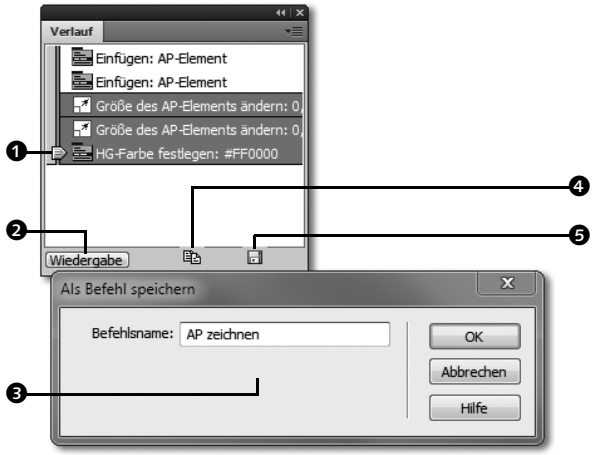


Abbildung 3.45 Verlauf als Befehl speichern

Gespeicherte Befehle stehen Ihnen in Dreamweaver immer zur Verfügung und werden im Menü BEFEHLE zu festen Bestandteilen des Programms. Über den Menüpunkt BEFEHLE • BEFEHLSLISTE BEARBEITEN können sie wieder gelöscht werden.

3.4.2 Extensions einsetzen

Wenn Sie Dreamweaver installieren, wird automatisch ein weiteres Programm installiert: der *Adobe Extension Manager*.

Dreamweaver allein bietet bereits eine Vielzahl von Werkzeugen. Zusätzlich können Sie das Programm mit Hilfe des Extension Managers erweitern und nahezu unbegrenzt ausbauen. Mit etwas Geschick im Programmieren können Sie auch eigene Erweiterungen anlegen. Allein auf der Adobe-Website sind viele Hundert Erweiterungen erhältlich. Dazu kommen noch unzählige in User-Groups und auf Websites, die sich speziell um Dreamweaver-Erweiterungen bemühen.

[»] **Erweiterungen zum vollen Funktionsumfang nötig**

Um den vollen Funktionsumfang von Dreamweaver auszukosten, müssen Sie mindestens drei Erweiterungen installieren. Die Business-Catalyst-Funktionen, Subversion und Adobe Air stehen Ihnen nur zur Verfügung, wenn Sie die entsprechenden Erweiterungen installiert haben.

3.4.3 Bezugsquellen

An Extensions gibt es nahezu nichts, was es nicht gibt. Sicherlich sind zwei Drittel nur unnütze Spielereien, der Rest ist oft nur als Testversion, danach kostenpflichtig verfügbar. Es finden sich allerdings auch einige Erweiterungen darunter, die das tägliche Arbeiten, gerade im Zusammenspiel mit Ajax, enorm erleichtern.

Eine Erweiterung, die wir gefunden haben, möchten wir Ihnen nicht vorenthalten: *Extension Locator* (siehe Abbildung 3.46) ermöglicht Ihnen die Suche nach Erweiterungen direkt aus Dreamweaver heraus.

[●] Die hier beschriebenen freien Erweiterungen finden Sie im Übrigen auch auf der DVD des Buches.

[»] **Auf Aktualität achten**

Während des Schreibens an diesem Buch standen uns leider oft nur die Erweiterungen für Dreamweaver CS4 zur Verfügung. Achten Sie darauf, immer die aktuellen Versionen zu verwenden. Erweiterungen für Dreamweaver 8.0 laufen beispielsweise nicht mehr mit den CS-Reihen.

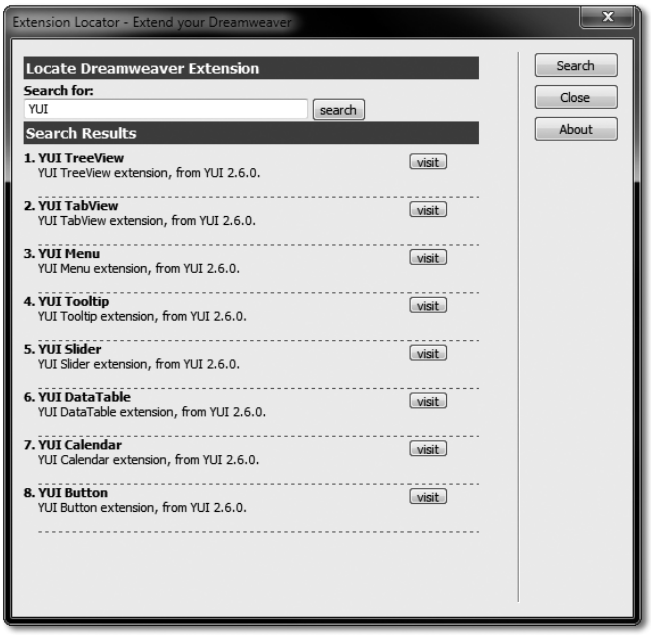


Abbildung 3.46 Extension Locator

3.4.4 Extension Manager

In Abbildung 3.47 sehen Sie die Oberfläche des Extension Managers. Im Menü auf der linken Seite können Sie auswählen, für welches Programm Sie Erweiterungen installieren möchten. Der Extension Manager ist übrigens für alle Applikationen aus der Creative Suite zuständig, die auf Ihrem System installiert sind.

Sie können den Extension Manager als eigenständiges Programm starten oder direkt aus Dreamweaver heraus über BEFEHLE • ERWEITERUNGEN VERWALTEN.

Wenn Sie mit vielen Erweiterungen arbeiten, ist es sinnvoll, diese nicht immer alle gleichzeitig zu aktivieren, sondern nur die aktuell benötigten. Die Dreamweaver-Oberfläche wird sonst schnell unübersichtlich.

Klicken Sie im Extension Manager auf INSTALLIEREN, um die Erweiterung Ihrer Wahl zu installieren.

Nach erfolgreicher Installation wird die Erweiterung in der Liste der installierten Erweiterungen angezeigt. Im unteren Teil des Fensters finden Sie Hinweise zum Urheberrecht und zur Bedienung der Erweiterung. Es ist jedoch von Erweiterung zu Erweiterung unterschiedlich, wo diese dann in Dreamweaver auch zur Verfügung gestellt wird. Hinweise dazu finden Sie im Text bei den jeweiligen Einträgen im Extension Manager.

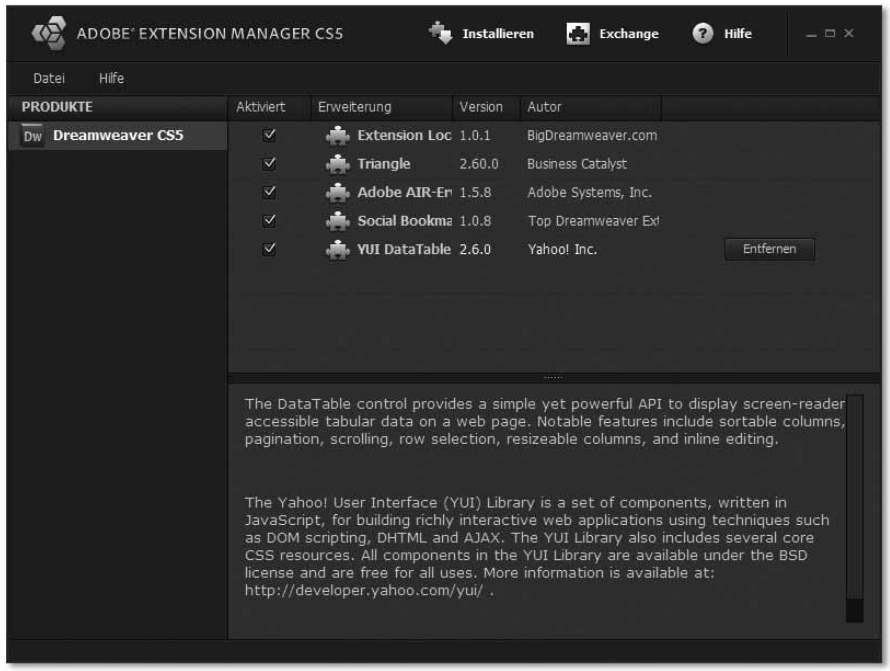


Abbildung 3.47 Extension Manager mit aktivierten Erweiterungen

[»] **Erweiterungen sichern**

Sichern Sie Ihre Erweiterungen an geeigneter Stelle. Daten von Websites zu sichern ist selbstverständlich, die Erweiterungen vergisst man jedoch schnell. Wenn Sie Ihr System aus irgendwelchen Gründen neu installieren müssen und Ihre häufig genutzten Erweiterungen nicht mehr finden, sind diese verloren. Vertrauen Sie nicht darauf, dass Sie die Erweiterung im Internet wiederfinden.

3.4.5 Ajax-Frameworks durch Extensions nutzen

Dreamweaver bringt das Ajax-Framework Spry mit. Viele Entwickler arbeiten lieber mit anderen Frameworks, wie zum Beispiel YUI von Yahoo. Für die meisten dieser Frameworks gibt es Erweiterungen, die auf einfache Art die teilweise sehr komplexen Funktionen zugänglich machen.

Dreamweaver-Erweiterungen installieren

Als Beispiel installieren wir eine Erweiterung zum Anzeigen von Bildern in einer sogenannten *lightbox*. Sie alle kennen den Effekt, wenn der Browserhintergrund grau wird und nur ein Bild mit einer Navigation erscheint.

Die Erweiterung *lightbox gallery* arbeitet mit *jQuery* und ist sehr einfach zu bedienen. Laden Sie die Erweiterung direkt von der Adobe-Exchange-Website herunter, oder benutzen Sie die Version von der DVD zum Buch. [o]

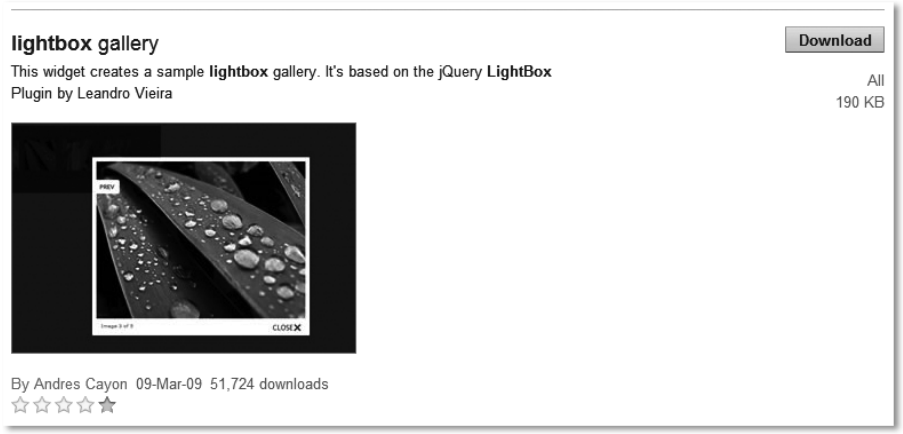


Abbildung 3.48 Dreamweaver-Erweiterung »lightbox gallery«

Starten Sie den Adobe Extension Manager, und installieren Sie die Erweiterung, wie im letzten Abschnitt beschrieben.

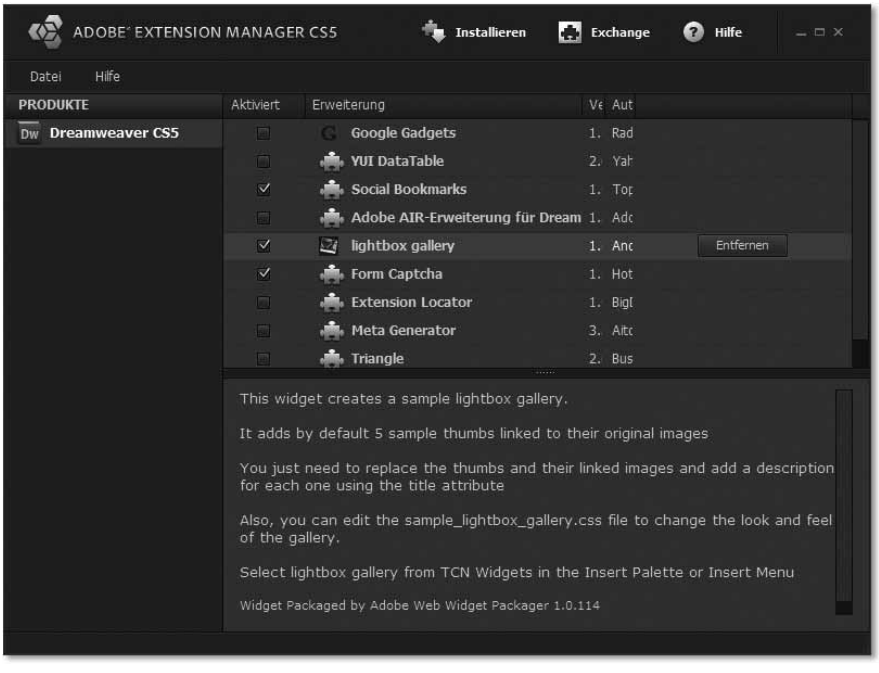


Abbildung 3.49 Extension Manager mit »lightbox gallery«

Nach erfolgreicher Installation und einem eventuellen Dreamweaver-Neustart steht in der Einfügeleiste ein neues Register TCN Widgets zur Verfügung. Die *lightbox* ist in diesem Fall das bislang einzige Widget von TCN.



Abbildung 3.50 Register »TCN Widgets«

TEIL II
Statische Websites



Was ist die Site-Verwaltung? Wie können Sie damit arbeiten, und was hat sie sonst noch zu bieten?

4 Eine Site erstellen

Im Internet herrscht Chaos. Auf Ihrer Website und in Ihrem Arbeitsablauf muss das nicht sein. Mit der Site-Verwaltung von Dreamweaver können Sie grundlegende Arbeitsschritte planen und durchführen. Dreamweaver ermöglicht Ihnen, einzelne Dokumente auch ohne das Anlegen einer Site zu bearbeiten. Tun Sie dies aber wirklich nur im Einzelfall.

Die Regel sollte das komplette Anlegen einer Site sein. Nur so können Sie sicher sein, dass Dreamweaver alle Pfade zu Dokumenten korrekt erkennt und Sie einen Überblick über alle Dateien haben. In diesem Kapitel beschreiben wir dieses Vorgehen.

Was genau ist eine Site? [«]

Wir sprechen immer von *Site*, *Website*, *HTML-Dokument* usw. Um Missverständnissen vorzubeugen, geben wir hier eine kurze Erklärung: Der Begriff *Site* – oder *Website*, was letztlich das Gleiche ist – bezieht sich auf die Gesamtmenge der innerhalb einer Website verwendeten HTML-Dokumente, Bilder, Grafiken und was sonst noch alles dazugehört. Ein *HTML-Dokument* hingegen bezeichnet ein einzelnes Dokument innerhalb einer Site.

4.1 Die Struktur

Im Folgenden schauen wir uns die grundlegenden Strukturen von Websites an.

4.1.1 Pfadangaben

HTML ist eine Skriptsprache, die dazu dient, dem Browser mitzuteilen, was und wie er etwas darstellen soll. Die meisten Inhalte, von Texten abgesehen, befinden sich nicht innerhalb des HTML-Dokuments, sondern sind separat auf dem Webserver abgelegt. All diese Dateien müssen nun in geeigneter Art und Weise den Weg zueinander finden. Im Internet steht alles in Beziehung zueinander.

Verknüpfungen

In HTML-Dokumenten werden Verknüpfungen mit relativen oder seltener auch mit absoluten Pfadangaben vorgenommen. Im Normalfall kennen wir nicht die gesamte Verzeichnisstruktur des Webservers. Wir bewegen uns nur innerhalb des von unserem Provider vorgegebenen Rahmens in dem Verzeichnis, das die gesamte Website enthält.

Absolute Pfadangaben

Absolute Pfadangaben (z. B. *http://www.adobe.com/de/index.html*) zeigen immer die gesamte URL eines Dokuments an und spezifizieren dadurch ein Ziel, ohne dass ein Ausgangspunkt bekannt sein muss. Es ist in etwa vergleichbar mit der Adresse auf einem Briefumschlag. Der Brief wird sein Ziel finden, egal wo er eingeworfen wird. Absolute Pfadangaben werden verwendet, um auf externe Dateien oder andere Websites zu verlinken und um Inhalte aus externen Quellen, wie zum Beispiel YouTube, in eine Website einzubinden.

```
<a href="http://www.galileodesign.de">Galileo Press</a>
```

Listing 4.1 Absolute Pfadangabe

Relative Pfadangaben

Bei relativen Pfadangaben (z. B. *bilder/kunde/bild.jpg* oder *../../bild.jpg*) wird nur der Weg von der Ausgangs- zur Zieldatei angegeben. Stellen Sie sich vor, Sie fragen jemanden nach dem Weg zu einer bestimmten Adresse. Sie erhalten die Antwort: 500 Meter geradeaus, dann an der Kreuzung links und die zweite Straße rechts. Vermutlich werden Sie Ihr Ziel finden. Wenn Sie jedoch an einem anderen Ort sind, trifft die Wegbeschreibung für Sie nicht mehr zu. Der Weg zwischen Ihnen und dem Zielort hat sich geändert. Relative Pfadangaben werden verwendet, um innerhalb einer Website zu verlinken oder Grafiken einzubinden. Bei Verlinkungen errechnet Dreamweaver die kürzesten Wege zwischen den Dokumenten und trägt sie als relative Pfade ein.

Der große Vorteil relativer Pfadangaben besteht darin, dass eine Website mit diesen Pfaden auf einem beliebigen Webserver lauffähig ist. Sie bildet quasi ein in sich geschlossenes System.

```
<a href="../index.htm">Galileo Press</a>
```

Listing 4.2 Relative Pfadangabe

Übersicht in der Sitemap

Bei kleineren Websites kann man durchaus alle Bilder und Dokumente direkt in einem Verzeichnis ablegen. Dann sind die Pfade denkbar einfach anzugeben.

Üblicherweise werden die Dateien jedoch innerhalb einer weiteren Ordnerstruktur abgelegt, die, wie Sie gleich sehen werden, unsere Sitemap oder die Navigation abbildet.

Die sich ergebenden Pfade von Hand in jedes Dokument einzutragen ist eine wahre Fleißarbeit und sehr fehleranfällig. Bei einer mittleren Website kommen schnell Hunderte von Dokumenten und Bildern zusammen. Wenn Sie auch nur eins davon verschieben oder umbenennen möchten, müssen Sie in allen anhängenden Dokumenten die Pfadangaben verändern.

Das ist der erste wichtige Punkt, den uns die Site-Verwaltung von Dreamweaver abnimmt. Eine korrekt angelegte Site vorausgesetzt, überwacht Dreamweaver im Hintergrund alle Dateiaktionen und verändert automatisch alle Pfadangaben. Sie können ohne Probleme ganze Verzeichnisse verschieben, und alle Verlinkungen werden weiterhin funktionieren. Dazu gleich mehr.

4.1.2 Ordnerstruktur

Schaffen Sie gleich beim Anlegen der Site eine sinnvolle Ordnerstruktur. Oft lässt sich darin die Navigation gut abbilden. In Abbildung 4.1 sehen Sie einen Navigationsplan und in Abbildung 4.2 die dazugehörige Ordnerstruktur.

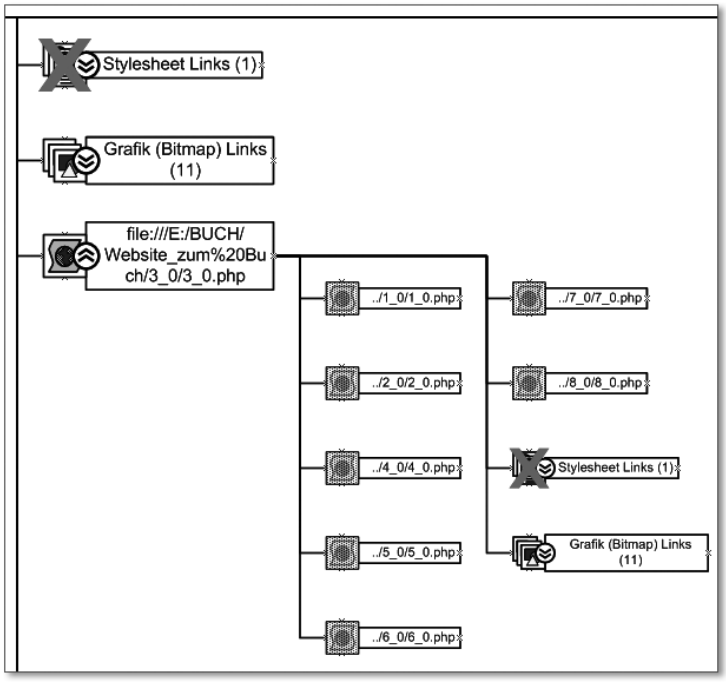


Abbildung 4.1 Navigationsplan einer Website (Ausschnitt)

Jedem Navigationspunkt wird eine eindeutige Nummer zugeordnet. Nach derselben Nummerierung werden dann die Dokumente und Ordner in der Site angelegt. Bilder und andere Dateien, die zu diesem Dokument in Bezug stehen, werden wiederum nach demselben Schema benannt. Voraussetzung für diese Vorgehensweise ist eine gute Dokumentation.

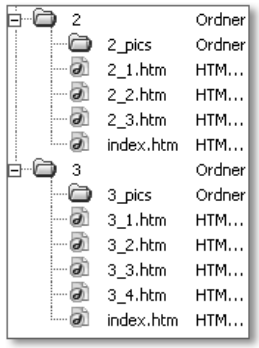


Abbildung 4.2 Ordner- und Dateistruktur in Dreamweaver

Übersicht durch Nummerierung

Auch wenn diese Systematik zunächst etwas umständlich erscheint, erweist es sich bei umfangreichen Sites als sehr sinnvoll, die Dokumente nach einem bestimmten Schema zu nummerieren. Wichtig ist dann allerdings eine gute Dokumentation, zum Beispiel folgende:

Dokumentname	Beschreibung
3_1.htm	Startseiten Produkte
3_2.htm	Detailseiten Produkte
3_3.htm	zusätzliche Informationen Produkte

Sie finden Dokumente letztlich schneller als mit Dateibezeichnungen wie *produktealt.htm*, *Produkte.htm*, *produkteneu.htm* usw. So haben Sie eine gute Grundlage, um auch noch in einem Jahr Ihre Arbeiten nachvollziehen zu können; und Dritte, die unbefugt an der Site »herumpfuschen« wollen, werden damit ziemliche Schwierigkeiten haben. 300 durchnummerierte Dokumente zu sortieren ist ohne vorliegende Dokumentation nicht wirklich einfach.

Dieses Schema setzen wir im dynamischen Teil des Buches fort. Bei dynamischen Websites wird häufig mit eingebundenen Dateien gearbeitet. Auch diese erhalten ihren strukturellen Bezug durch die Nummerierungen.

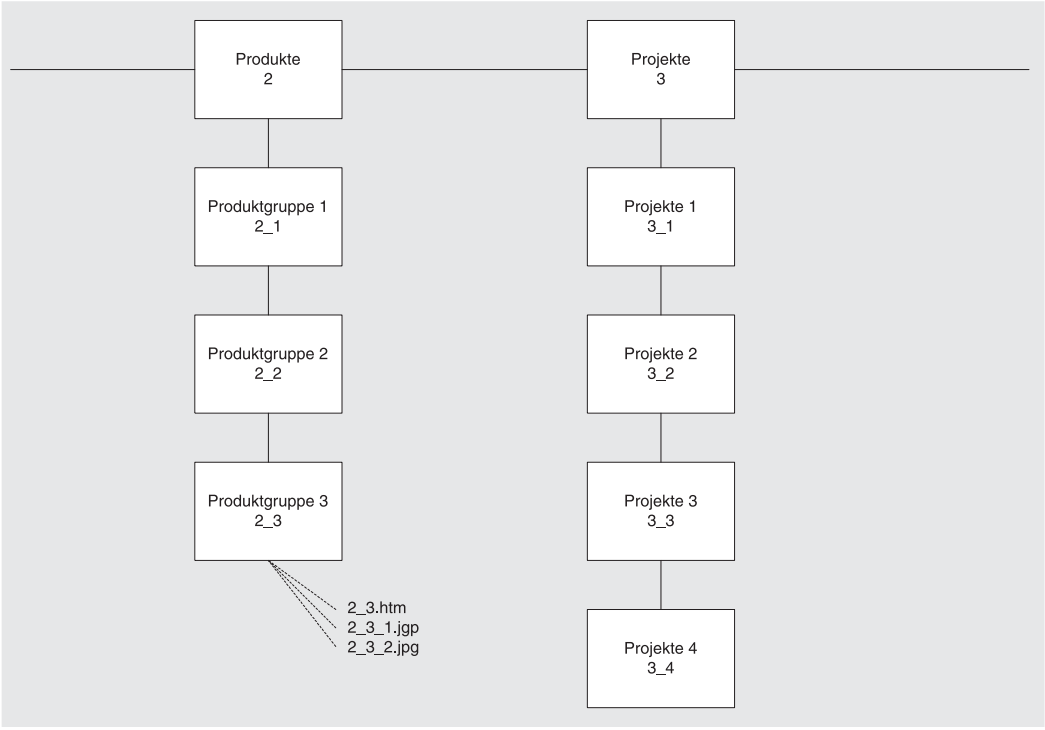


Abbildung 4.3 Schema einer Website

Benutzerfreundlichkeit und Suchmaschinenoptimierung

Für Entwickler ist die Nummerierung der Dokumente vorteilhaft, insbesondere bei der Seitenpflege. Allerdings ist diese Art der Benennung nicht gerade benutzerfreundlich.

So ist die URL *http://www.meinewebsite.de/06_03_09_kontakt/012_03_formular.html* sicher nicht so leicht zu merken wie die nach Namen geordnete Site *http://www.meinewebsite.de/kontakt*. Dabei befindet sich das Kontaktformular mit dem Namen *index.htm* lediglich im Unterverzeichnis *kontakt*. Der Server sucht automatisch nach der Indexdatei, wenn in der URL ein Pfadname ohne direkte Datei eingegeben wird.

Ein weiterer Vorteil von eindeutigen Namen in den Verzeichnissen und Dateien besteht darin, dass diese Begriffe innerhalb von Links von Suchmaschinen gelesen und bewertet werden. Aussagekräftige Bezeichnungen helfen hier mehr. Im Rahmen dieses Buchs bleiben wir allerdings wegen der einfacheren Erklärung bei der Methode mit der Nummerierung.

[»] **URL für Suchmaschinen automatisch umbenennen**

Haben Sie sich schon einmal gewundert, dass viele HTML-Dokumente im Internet wie durch Zauberhand genau so benannt sind wie Ihre Suchanfrage? Dafür verantwortlich ist die Apache-Funktion `mod_rewrite`. Diese übermittelt eine »umgeschriebene« URL an den Browser. Die Vorgaben, nach denen diese Manipulation erfolgt, können Sie fast frei bestimmen. Sie können also mit nummerierten Dokumenten und Ordnern arbeiten, dem Besucher und den Suchmaschinen aber etwas anderes anzeigen. Weitere Informationen dazu finden Sie unter:

http://httpd.apache.org/docs/2.0/mod/mod_rewrite.html

4.2 Die Site-Verwaltung

Da die korrekte Definition einer Site in der Site-Verwaltung das A und O für das Funktionieren der späteren Website ist, wollen wir in diesem Abschnitt detailliert auf diesen Arbeitsschritt eingehen.

4.2.1 Regeln beim Arbeiten mit einer Site

Beim Anlegen einer Website kann man viel falsch machen und den Grundstein für so manchen später auftretenden Fehler legen. Achten Sie darauf, dass Sie Dateioperationen nach dem Anlegen der Site ausschließlich innerhalb der Site-Verwaltung vornehmen. Wenn Sie trotz angelegter Site die Strukturen mit dem Windows Explorer verändern, merkt Dreamweaver das nicht und kann keine Korrekturen bei den Verlinkungen durchführen.

Keine verschachtelten Sites anlegen

Legen Sie niemals innerhalb eines Ordners, der bereits eine Site enthält, eine weitere Site an – auch nicht »mit Gewalt«. Dreamweaver gibt dabei eine Fehlermeldung aus, die Sie in keinem Fall ignorieren sollten. Das Ergebnis kann verheerend sein. Spätestens dann, wenn Sie einmal die Pfade von 500 Dateien von Hand korrigieren müssen, machen Sie es nie wieder!

Dateien richtig benennen

Auch in der Site-Verwaltung gelten die üblichen Regeln für Internet-Dateinamen. Das Internet besteht zum größten Teil aus Linux- oder Unix-Systemen, vergessen Sie also bitte die tolerante Schreibweise von Windows. Gewöhnen Sie sich folgendes Schema an:

Benutzen Sie *keine Sonderzeichen*, wie z. B. Ä, Ö, Ü, &, " oder ', in Datei- oder Ordnernamen. Verwenden Sie auch *keine Leerzeichen*, sondern den Unterstrich als Trennzeichen: *name_vorname*.

Beachten Sie die Groß- und Kleinschreibung, und schreiben Sie am besten immer alles klein. Linux-Systeme sind case sensitiv. *Datei.htm* und *datei.htm* werden als zwei völlig verschiedene Dateien betrachtet. Wenn die Benennung in der von uns vorgeschlagenen numerischen Weise geschieht, laufen Sie nicht Gefahr, fehlerhafte Zeichen in Dateinamen einzugeben. Zudem werden die Dateinamen sehr kurz und passen sogar in das 8.3-Schema von MS-DOS.

Konsequente Benennung

Gewöhnen Sie sich eine einheitliche Schreibweise der Dateiendungen an. Diese kann *.htm* oder *.html* sein, wichtig ist nur, dass Sie sie *immer* beibehalten. Ein Link zur Datei *index.html* funktioniert nicht, wenn diese tatsächlich *index.htm* heißt.

Indexdateien anlegen

Legen Sie in jedem Verzeichnis als Startdatei eine *index.htm* an. Wenn dies aus strukturellen Gründen nicht sinnvoll ist, bauen Sie darin eine Weiterleitung ein. Webserver suchen automatisch nach einer Indexdatei und zeigen diese an, wenn keine andere HTML-Datei angegeben wird. Ist diese nicht vorhanden und zudem auch noch der Webserver fehlerhaft konfiguriert, wird Benutzern die Verzeichnisstruktur der Website angezeigt. Das wirkt unprofessionell und ist auch ein großes Sicherheitsrisiko.

Alte Dateien löschen

Achten Sie darauf, dass die Site im Laufe der Entwicklung nicht mit jeder Menge Testdateien und abgespeicherten Zwischenschritten »zugemüllt« wird. Sammeln Sie diese in einem separaten Ordner, den Sie anschließend archivieren oder löschen können. Außerdem sollten Sie alle Dateien, die nicht mehr benötigt werden, vom Webserver löschen. Oft werden im Laufe der Entwicklung Testskripts auf dem Server abgelegt. Werden diese Skripts nicht gelöscht, können Hacker aus den ausgegebenen Daten eventuell Rückschlüsse auf die Programmierung etc. ziehen. Vermeiden Sie das!

4.2.2 Testserver

Ein Testserver ist ein lokaler Webserver, der zur Entwicklung und Simulation dynamischer oder statischer Websites dient.

Um Datenbankanbindungen mit Dreamweaver zu erstellen, ist ein Testserverbetrieb zwingend notwendig. Wir werden in Kapitel 21, »Lokalen Webserver installieren«, einen Testserver einrichten und die notwendigen Einstellungen ausführlich erläutern. Alle Tools dafür finden Sie auf der DVD zum Buch im Ordner *Webserver*. Für die Arbeit mit statischen Websites ist ein Testserver unnötig, daher überspringen wir diesen Teil zunächst.

4.2.3 Site-Definition

Um zur Site-Definition zu gelangen, in der Sie grundlegende Einstellungen vornehmen können, öffnen Sie im Menü SITE • SITES VERWALTEN die Dialogbox zum Anlegen neuer oder zum Bearbeiten bestehender Sites. Alternativ können Sie im Bedienfeld DATEIEN auch das Menü ausklappen ❶ und auf SITES VERWALTEN ❷ klicken.

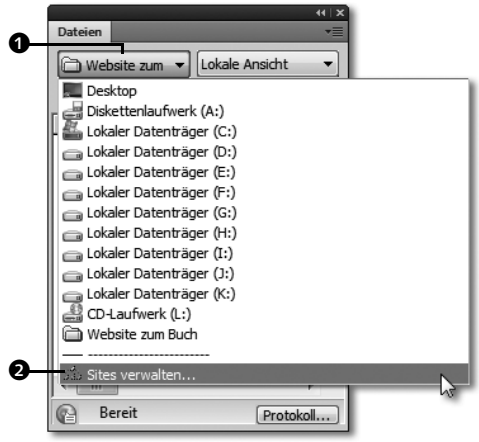


Abbildung 4.4 Site-Verwaltung öffnen über das Bedienfeld »Dateien«

[!] **Site-Verwaltung komplett neu in CS5**

Wer bereits mit älteren Dreamweaver-Versionen gearbeitet hat, wird die Site-Verwaltung nicht wiedererkennen. Adobe hat sie vollständig erneuert. Nach anfänglicher Skepsis mussten wir feststellen, dass sich damit aber hervorragend arbeiten lässt.

Nachdem Sie auf SITES VERWALTEN geklickt haben, kommen Sie in ein Menü zum Anlegen einer neuen Site und zur Bearbeitung bestehender Sites (siehe Abbildung 4.5). Auch Import- und Export-Funktionen stehen hier zur Verfügung.



Abbildung 4.5 Site-Auswahl

Import und Export von Sites

Durch das Importieren und Exportieren von Sites können Sie alle Definitionen einer Website mit den Zugangsdaten in einer Site-Definitionsdatei mit der Dateiendung *.ste* ablegen und sichern.

Bei einem Systemcrash, einer Neuinstallation usw. werden diese Dateien dann einfach importiert, und die Einstellungen sind wieder die alten. Gegebenenfalls müssen Sie den lokalen Stammordner neu auswählen, wenn sich im Dateisystem etwas verändert haben sollte.

Leider gibt es keine andere Möglichkeit, die gesamten Einstellungen von Dreamweaver inklusive der Site-Einstellungen zu sichern. Auch wenn Sie das gesamte Konfigurationsverzeichnis sichern, sind bei einer erneuten Installation die Site-Einstellungen verloren.

Site-Name

Damit Dreamweaver Sites verwalten kann, müssen diese eindeutige Namen erhalten. Geben Sie einen solchen Namen bei Site-Name ❸ (siehe Abbildung 4.6) an. Er dient ausschließlich der Organisation der Daten innerhalb von Dreamweaver und Ihrer Ordnung und wird nicht ins Internet übertragen. Sie können den Namen frei wählen und brauchen die Schreibweisen von Linux hier nicht zu berücksichtigen. Es bietet sich an, den Namen entsprechend dem Projektnamen oder Auftraggeber zu wählen.

Lokaler Site-Ordner

Wir haben im vorherigen Abschnitt erklärt, wie die Dateien strukturiert werden sollten. In der Site-Definition müssen Sie im Feld Lokaler Site-Ordner einen Ordner auswählen, in dem sich die Site entweder bereits befindet oder, wie in unserem Fall, erstellt werden soll. Der Site-Ordner bzw. Stammordner ist zugleich der Bezugspunkt `root` der Website.

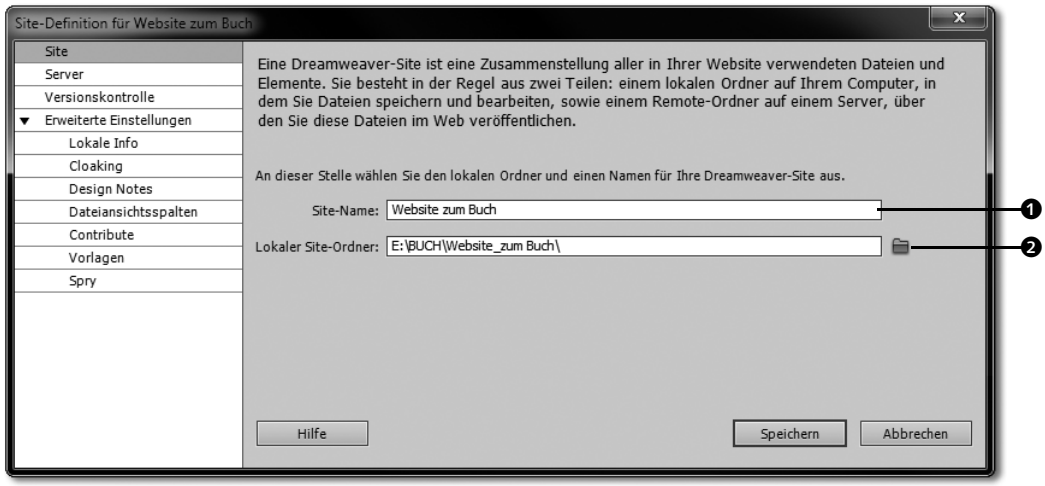


Abbildung 4.6 Site-Definition

Klicken Sie auf das kleine Ordner-Icon ❹ rechts neben dem Eingabefeld zur Auswahl eines bestehenden oder zum Anlegen eines neuen Ordners.

Wenn Sie einen neuen Ordner erstellen, müssen Sie in den Ordner wechseln, bevor Sie auf OK klicken. Der Ordner sollte allerdings nicht unbedingt auf dem Hauptlaufwerk oder im Programmverzeichnis stehen. Auch der Desktop bzw. Schreibtisch unter Mac OS X oder das Verzeichnis *Eigene Dateien* ist nicht der richtige Platz. Am besten siedeln Sie den Ordner auf einem Arbeitslaufwerk (kein Netzwerkordner) und dort in einem Unterordner für alle Websites an. *D:\websites\meinewebsite* wäre ein Beispiel.

Sie können den Ordner auch vorher mit dem Windows Explorer erstellen. Das sollte dann aber die letzte Aktion sein, die Sie außerhalb der Site-Verwaltung erledigen.

Wenn wir später einen FTP-Zugang zum Webserver einrichten, wird im lokalen Stammordner der Webserver (bzw. das für Sie freigegebene Verzeichnis) gespiegelt. Das heißt, der Inhalt Ihres lokalen Stammordners und der Inhalt des Ordners auf dem Webserver sind identisch.

4.2.4 Erweiterte Einstellungen

Neben den wichtigen Einstellungen, die dafür sorgen, dass die Website überhaupt funktioniert, gibt es noch einige zusätzliche Einstellungen, die optional sind, das Arbeiten aber deutlich erleichtern können.

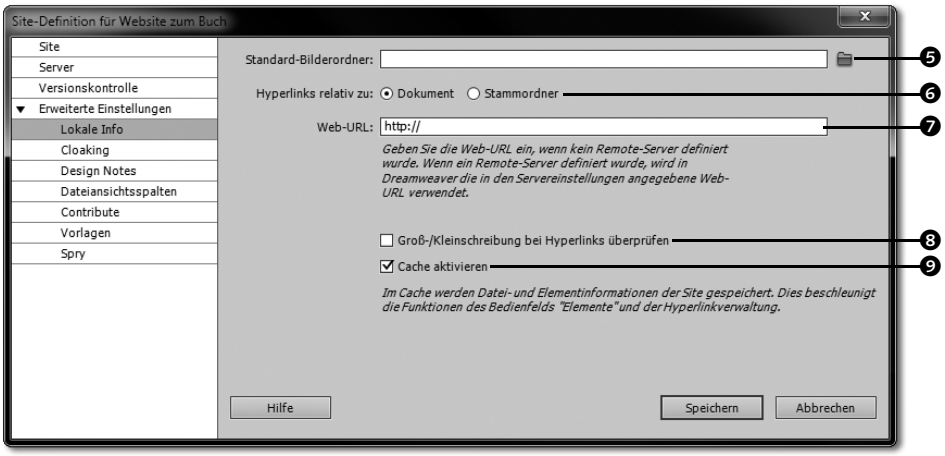


Abbildung 4.7 Lokale Infos in der Site-Verwaltung anlegen

Standard-Bilderordner

Der STANDARD-BILDERORDNER ❺ ist der Ordner, auf den automatisch als Erstes zugegriffen wird, wenn Sie ein neues Bild einfügen möchten. Sie können hier Ihren gewünschten Ordner angeben oder das Feld wie in unserem Beispiel einfach leer lassen. In der Praxis hat sich ein Verzeichnis mit dem Namen *IMAGES* bewährt.

Hyperlinks relativ zu

Es können HYPERLINKS RELATIV zum STAMMORDNER oder zum DOKUMENT ❻ vorgegeben werden. Bei einer auf den Stammordner bezogenen Verlinkung (root-relativ) werden Verlinkungen immer vom Stammordner der Site ausgehend mit vorangestelltem/geschrieben. Die Standardeinstellung in Dreamweaver ist dokument-relativ.

Die Vorteile einer auf den Stammordner bezogenen Verlinkung zeigen sich erst bei dynamischen Websites. Bei dieser Art von Websites weiß man bei eingebundenen Dokumenten oft nicht, aus welchem Ordner heraus diese abgerufen werden. Hierbei ist daher root-relativ die bessere Wahl.

Der Nachteil dieser Art der Verlinkung ist, dass Sie keine lokale Vorschau erzeugen können. Sie müssen in diesem Fall zwingend mit einem Testserver arbeiten.

Web-URL

Tragen Sie jetzt bei WEB-URL ❼ die URL mit dem Domainnamen als absoluten Pfad Ihrer Website ein (z. B. *http://www.meinewebsite.de*). Dreamweaver benö-

tigt diese Angabe, um Verlinkungen in der Site zu überprüfen. Wenn Sie die Adresse nicht eintragen, kann das Programm Verlinkungen, die auf absoluten Pfadangaben beruhen, nicht testen. Auch dieses Feld bleibt bei unserer Übungswebsite frei, da wir keine Domain haben, auf der sie einmal laufen wird.

Wir empfehlen in jedem Fall, die Groß- und Kleinschreibung bei der Hyperlink-Überprüfung zu beachten und diesen Punkt zu aktivieren ❽. Leider ist Windows an dieser Stelle sehr tolerant, so dass sich viele Windows-Benutzer einen nachlässigen Umgang damit angewöhnt haben. Webserver sind selten derart tolerant. Aktivieren Sie diese Checkbox, dann fallen eventuelle Fehler auf, bevor die Site im Web zu sehen ist.

Cache aktivieren

Die Option Cache aktivieren ❾ sollten Sie immer anwählen. Nur so bekommt Dreamweaver eindeutig Änderungen innerhalb der Website mit und kann z. B. Pfade korrigieren. Viele Arbeiten am Dateisystem gehen auf diese Weise deutlich schneller. Der Cache kann später übrigens auch ganz einfach manuell neu erstellt werden.

4.2.5 Server verwalten

Um einen FTP-Zugang oder auch einen Testserver anzulegen, klicken Sie in der linken Navigationsleiste auf Server ❶. Um einen Server zu aktivieren, müssen Sie in der Serververwaltung auf das Plussymbol ❷ klicken.

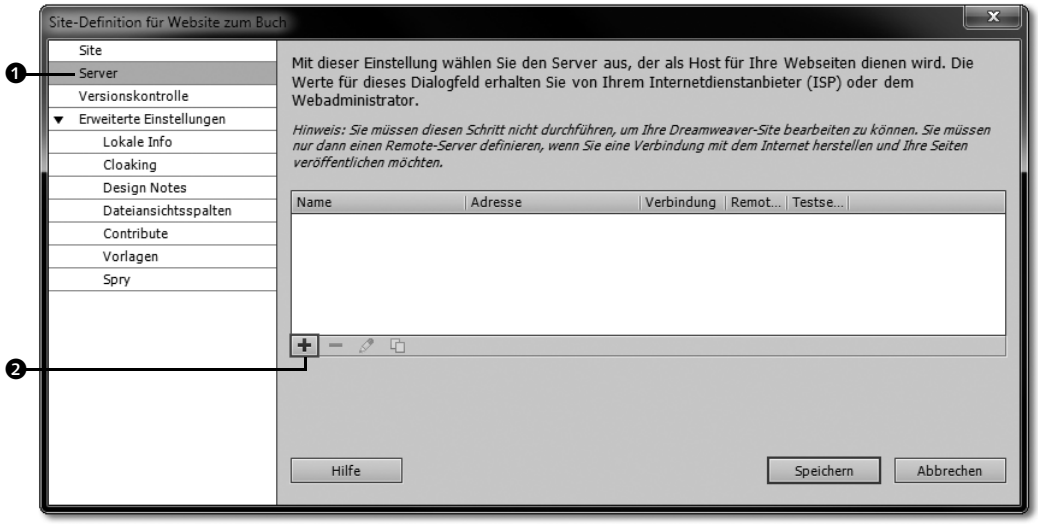


Abbildung 4.8 Server verwalten

FTP-Zugang einrichten

In unseren Abbildungen haben wir im SERVER-Dialog zusätzlich WEITERE OPTIONEN aktiviert, um Ihnen auch diese zu erläutern.

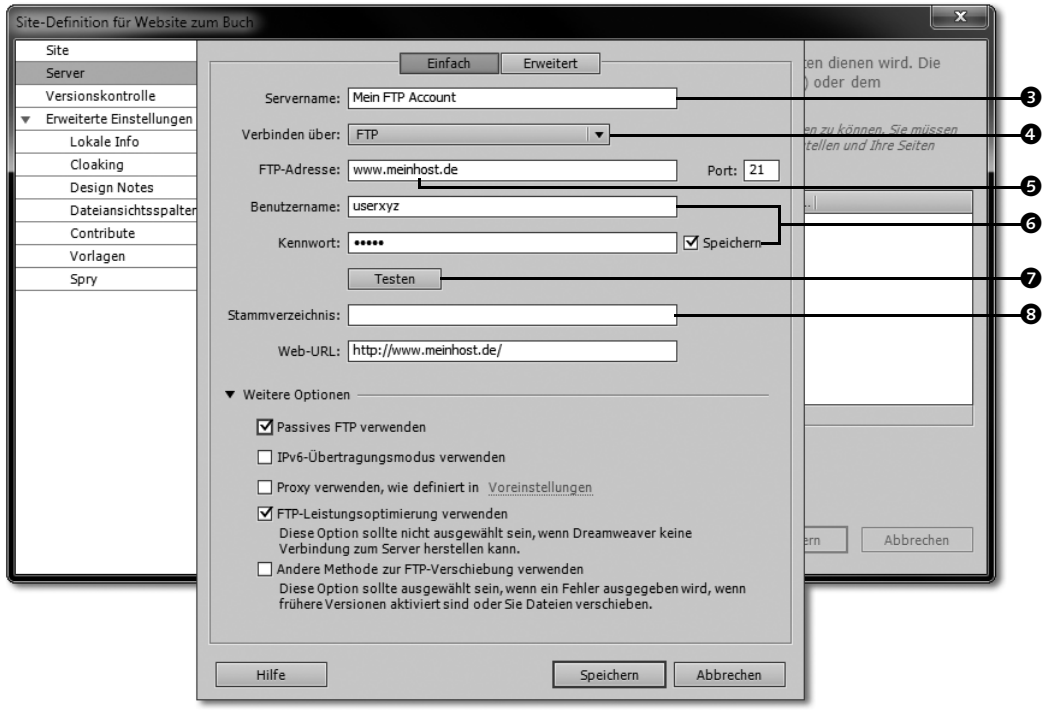


Abbildung 4.9 FTP-Account einrichten

Servername angeben

Da Dreamweaver in der aktuellen Version in der Lage ist, mehrere verschiedene Zugänge zu verwalten, muss zunächst ein Name für den Server angegeben werden ❸. Sie können den Server frei benennen, auf die Funktion hat das keine Auswirkung.

Verbindungstyp auswählen

Im Menü VERBINDEN ÜBER ❹ ist FTP als Standardvorgabe ausgewählt. Neben einer FTP-Verbindung können weitere Verbindungstypen angelegt werden.

Das klassische FTP-Protokoll ist alles andere als sicher. Jeder, der sich im selben Netz befindet, kann unter Einsatz geeigneter Programme (Sniffer) problemlos Ihre Datenübertragungen mitverfolgen, so auch die Zugangsdaten und Passwörter. Problematisch kann dies werden, wenn Sie sehr sensible Daten übertragen.

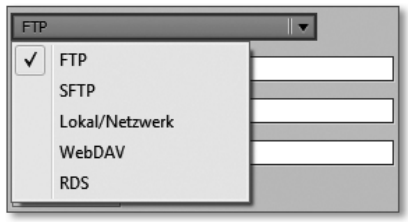


Abbildung 4.10 Verbindungstypen auswählen

SFTP (SSH File Transfer Protocol) benutzt eine sichere, verschlüsselte Verbindung, um die Daten zwischen Ihrem Rechner und dem Webserver zu verbinden. Die meisten Provider unterstützen mittlerweile diesen Verbindungstyp. Der Begriff SFTP wird leider oft auch für das *Simple File Transfer Protocol* verwendet. Dieses sollten Sie allerdings nach Möglichkeit nicht mehr nutzen, da es als veraltet gilt und kaum Sicherheit bietet.

Eine Alternative zu FTP

Statt FTP können Sie auch einen WebDAV-Zugriff auf einen Webserver anlegen, vorausgesetzt, Ihr Provider unterstützt dieses Protokoll. WebDAV steht für *Web-based Distributed Authoring and Versioning*.

Bidirektionale Datenübertragung

Bei WebDAV handelt es sich um einen offenen Standard, um Daten im Internet zu übertragen. Technisch gesehen ist WebDAV ein Protokollaufsatz bzw. eine Erweiterung des HTTP-Protokolls (*Hypertext Transfer Protocol*), des Standardprotokolls zur Datenübertragung im Internet.

Das HTTP-Protokoll ist zwar bidirektional, in der Praxis fließen jedoch die Daten meistens vom Server (*http://www.domain.de*) zum Client (Browser des Users). Hier soll WebDAV Abhilfe schaffen. Mit WebDAV können Daten mit dem HTTP-Protokoll bidirektional übertragen werden. Dies kann eingesetzt werden, um z. B. Webseiten auf dem Server zu speichern. In diesem Fall leistet WebDAV im Vergleich zum veralteten FTP-Protokoll gute Dienste. So ist z. B. ausgeschlossen, dass mehrere Personen auf die gleiche Datei zugreifen können. Auch eine Namens- und Versionsverwaltung ist im WebDAV-Protokoll implementiert.

Das WebDAV-Protokoll ist in modernen Betriebssystemen, wie Windows Vista und XP, Mac OS X oder GNU/Linux, bereits vorgesehen. Weiterhin finden sich im Internet jede Menge Dienstleister, die Speicherplatz über WebDAV quasi als virtuelle Festplatte zur Verfügung stellen.

Berechtigungen setzen (CHMOD)

In älteren Dreamweaver-Versionen war es nicht möglich, die Berechtigungen auf einem Webserver für Schreib- und Leserechte zu setzen. Inzwischen können Sie in der Remote-Site durch einen Klick mit der rechten Maustaste auf den Dateinamen oder Ordner die üblichen Berechtigungen (CHMOD) setzen.

FTP-Webserver festlegen

Die FTP-Adresse ❺ (siehe Abbildung 4.9) entspricht der Adresse Ihres Webservers, unterscheidet sich aber häufig in der Schreibweise vom Aufruf der Website im Browser. Wenn Sie einen Webserver auf der Domain *http://www.meinedomain.de* haben, kann der FTP-Server z. B. den Namen *ftp://www.meinedomain.de* tragen oder *ftp.meinedomain.de* heißen, aber genauso als *www.meinedomain.de* eingetragen sein. Es kommt auch vor, dass Sie einfach eine IP-Adresse erhalten oder eine ganz andere Domain, die dem physikalischen Standort des Servers entspricht. Wenn Sie unsicher sind, fragen Sie Ihren Provider nach den richtigen Einstellungen. Er kann Ihnen die notwendigen Informationen liefern.

Zugangsdaten festlegen

Tragen Sie die Zugangsdaten Ihres Providers unter Benutzername und Kennwort ❻ ein. Bedenken Sie dabei, dass das Kennwort nie im Klartext angezeigt wird und sich nicht nur dem Zugriff Unbefugter entzieht, sondern auch Ihrem, wenn Sie es vergessen haben. Vermerken Sie es am besten an sicherer Stelle.

Kopierfehler [!]

Es kommt immer wieder vor, dass man Zugangsdaten vom Provider per Mail bekommt und dennoch nach dem Kopieren in die jeweiligen Felder kein Zugang möglich ist. Oft liegt es daran, dass in der E-Mail die Zeichen kodiert sind und beim Einfügen eben diese unsichtbaren Sonderzeichen mitkopiert werden. Schreiben Sie Zugangsdaten möglichst immer von Hand in die betreffenden Felder.

Stammverzeichnis auf dem Server festlegen

Normalerweise können Sie das Feld Stammverzeichnis ❽ frei lassen. Dies hängt von der Verzeichnisstruktur auf dem Server ab. Ihr Provider kann Ihnen im Zweifelsfall hier weiterhelfen.

Verbindung testen

Jetzt können Sie auf Testen ❼ klicken, und Dreamweaver versucht, eine Verbindung zum Server herzustellen. Gelingt dies, erhalten Sie eine Erfolgsmeldung. Falls nicht, überprüfen Sie die Zugangsdaten.

Keine Verbindung zum Host

Es gibt noch weitere mögliche Fehlerquellen, falls keine Verbindung zum Server hergestellt werden kann. Wenn Sie mit Ihrem Rechner hinter einem Router oder einer Firewall sitzen (auch Desktop-Firewall), sollten Sie auf jeden Fall die Option Passives FTP verwenden aktivieren. Die Anforderungen eines aktiven FTP-Servers akzeptiert praktisch keine gute Firewall. Passives FTP ist zwar geringfügig langsamer, funktioniert dafür umso zuverlässiger. Beim passiven FTP werden von Seiten des Hosts keine Anfragen an den Client (Ihren Rechner) gerichtet.

Wenn Ihr lokaler Rechner durch eine Firewall abgesichert ist, müssen Sie diese unter Umständen in den Firewall-Einstellungen konfigurieren. Erfragen Sie die dafür nötigen Einstellungen bei Ihrem Systemadministrator, meist muss Port 21 für FTP oder Port 22 für SFTP freigegeben werden.

Alternativen zu Dreamweaver-FTP

Die Datenübertragung aus Dreamweaver heraus ist bestens geeignet, wenn Sie schnell einige wenige Dateien auf den Webserver übertragen möchten. Bei umfangreichen Datentransfers ist Dreamweaver leider nicht wirklich schnell. Auch das Löschen oder Verschieben von Dateien auf dem Server kann ziemlich langsam vonstattengehen. Hier empfehlen sich Alternativen wie Filezilla. Die jeweils aktuellste Version finden Sie hier: *http://filezilla-project.org/*

4.2.6 Erweiterte Einstellungen

Seit Dreamweaver 8 gibt es ein Protokoll für Aktionen mit der Site-Verwaltung. Wenn Sie mit den Synchronisationsinformationen arbeiten wollen, müssen Sie sie hier aktivieren. Wir haben leider nicht die besten Erfahrungen damit gemacht. Öfter werden Dateien nicht als neu erkannt, und Dreamweaver weigert sich, die Dateien auf den Webserver zu übertragen. Wenn Sie gewissenhaft arbeiten, benötigen Sie dieses Feature kaum.

Dokumente lokal prüfen

Achten Sie unbedingt darauf, den Punkt Dateien beim Speichern automatisch auf den Server hochladen ❶ zu deaktivieren. Sonst kann es sehr schnell passieren, dass fehlerhafte Versionen Ihrer Dokumente – die es während der Arbeit zwangsläufig gibt – unbeabsichtigt auf den Server übertragen werden und der Öffentlichkeit fehlerhaft zur Verfügung stehen. Auch in Ausnahmen, die es kaum gibt, sollten Sie Dokumente immer erst lokal überprüfen, bevor Sie diese veröffentlichen.

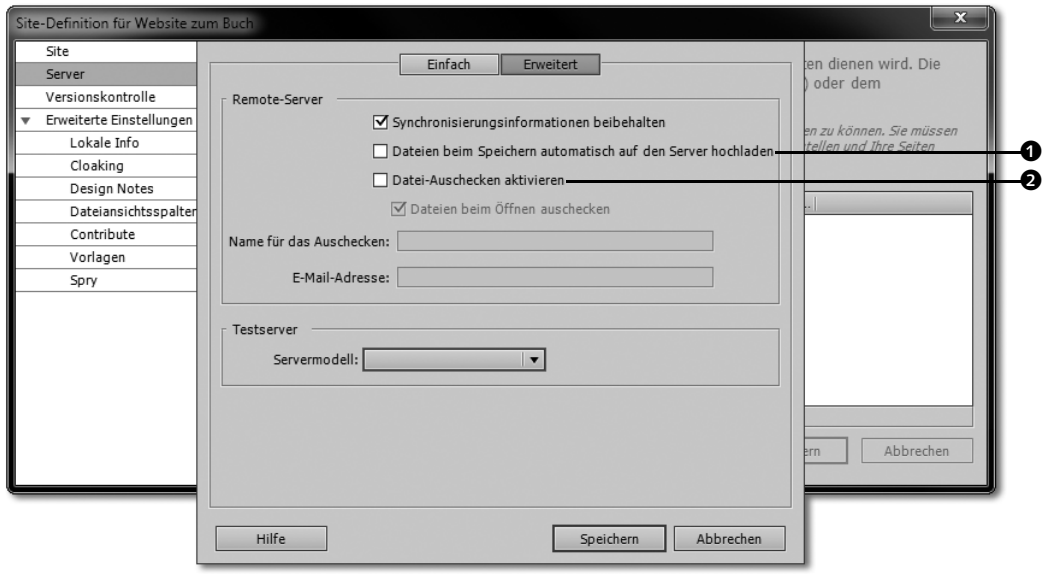


Abbildung 4.11 Erweiterte Einstellungen

Die Funktion Datei-Auschecken aktivieren ❷ beschreiben wir in Abschnitt 4.4.2, »Mit der Gruppenfunktion arbeiten«.

4.2.7 Versionskontrolle

Speziell bei der Arbeit in einer größeren Entwicklergruppe ist es schwer, den Überblick über den Entwicklungsstand einzelner Dateien und Verzeichnisse zu behalten. Hierfür werden spezielle Hilfsprogramme von Drittanbietern eingesetzt, die es unter anderem auch erlauben, ältere Versionen wiederherzustellen.

Seit CS4 ist Dreamweaver direkt in der Lage, mit dem System Subversion (*http://subversion.tigris.org/*) zu kommunizieren (siehe Abbildung 4.12). Die Einstellungen hierfür erfahren Sie von Ihrem Administrator. Wir haben Ihnen die notwendigen Tools mit auf die DVD gepackt. [o]

Wenn ein Subversion-Server eingesetzt wird (siehe Abbildung 4.13), lohnt es sich, auch das Tool TortoiseSVN (siehe Abbildung 4.14) zu nutzen. Sie erweitern damit die Versionsverwaltung auf beliebige Dateien und können auch mit dem Windows Explorer arbeiten.

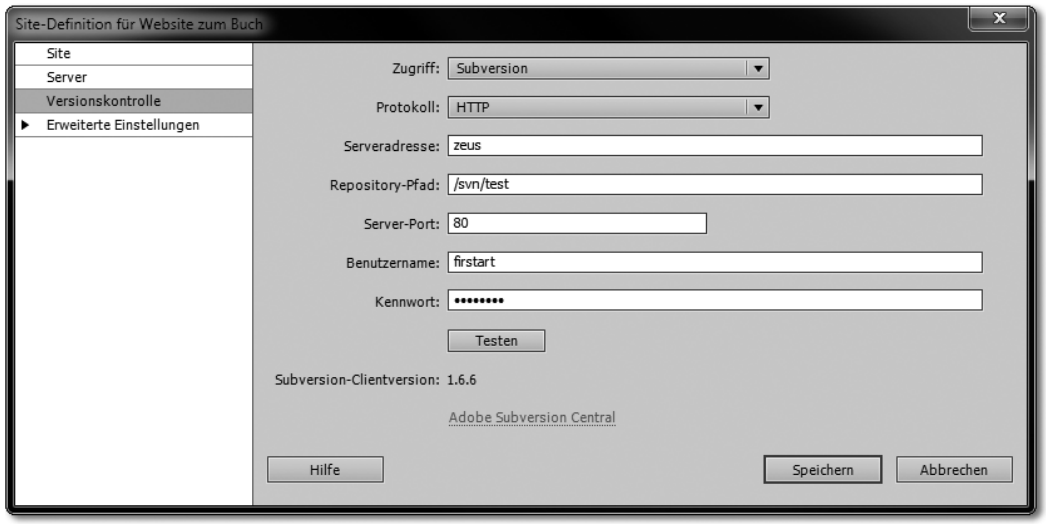


Abbildung 4.12 Versionskontrolle in Dreamweaver

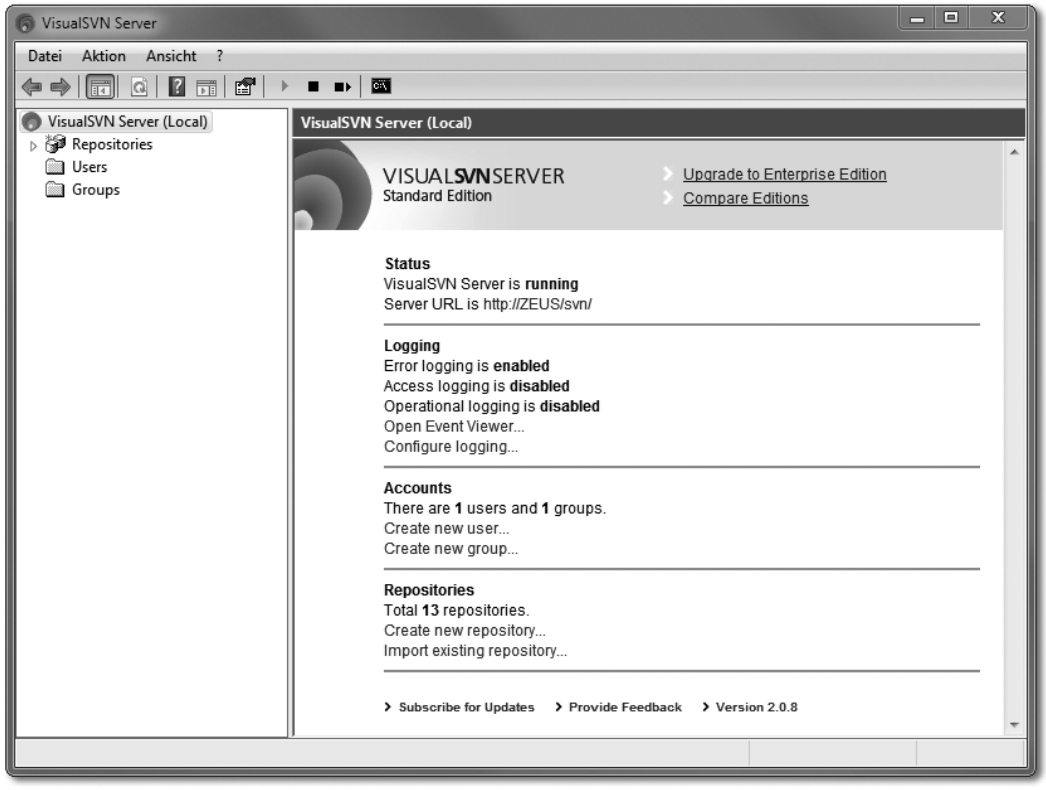


Abbildung 4.13 VisualSVN Server

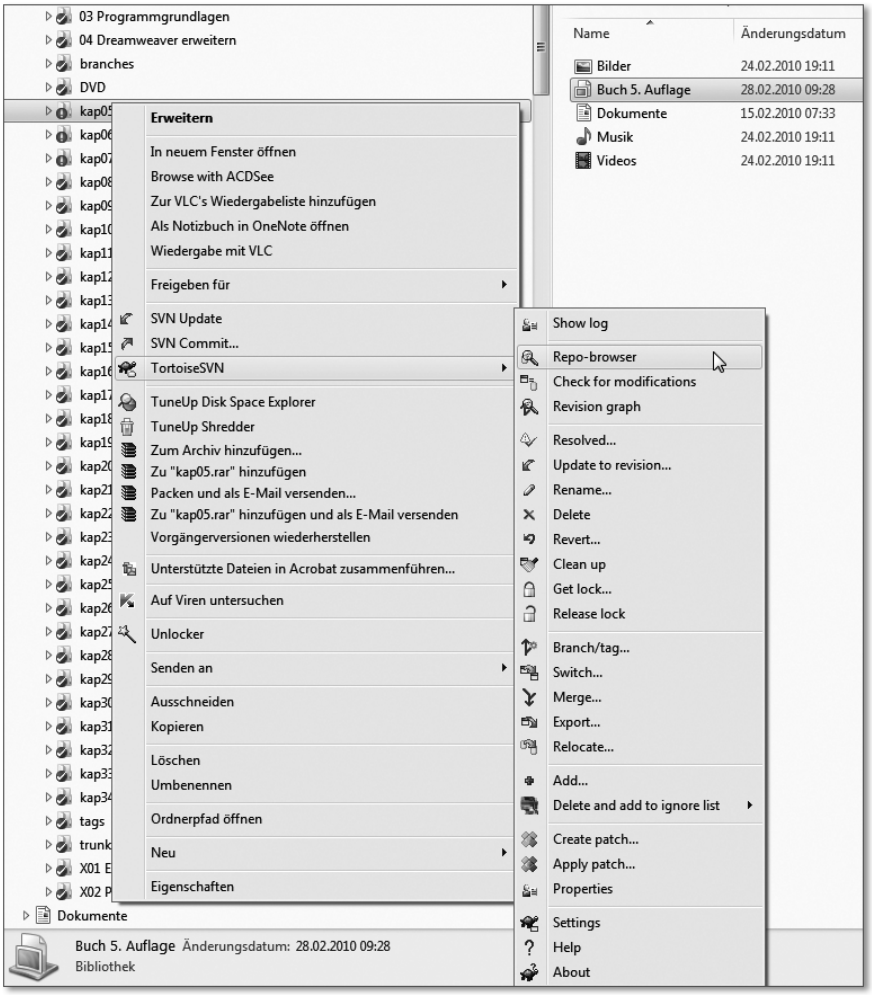


Abbildung 4.14 TortoiseSVN im Browser

4.2.8 Cloaking

Mit der Cloaking-Funktion (siehe Abbildung 4.15) können Sie einzelne Dateien oder Dateitypen von der Übertragung auf den Webserver ausschließen. Dieses Hilfsmittel greift dann, wenn Sie Dateien, ganze Verzeichnisse oder die gesamte Site bereitstellen wollen.

Sinnvoll eingesetzt werden kann diese Funktion, wenn Sie innerhalb Ihrer Dateistruktur einen oder mehrere Ordner anlegen, in denen beispielsweise die Photoshop-Dateien mit den Entwürfen Ihrer Site liegen. Diese Dateien sind meist sehr groß und haben im Internet nichts zu suchen. Sie können daher alle Photo-

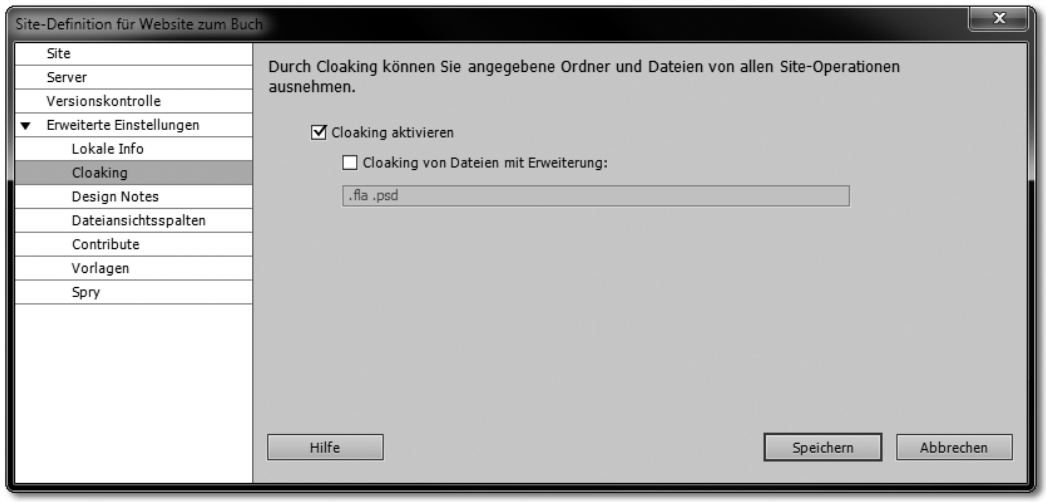


Abbildung 4.15 Cloaking in Dreamweaver

shop-Dateien (*.psd*) generell »cloaken« lassen. Diese werden dann nicht mit auf den Webserver übertragen. Sinnvoll ist auch die Angabe von *.tif*, *.bmp*, *.eps* (jeweils Bildformate, die nicht direkt im Browser angezeigt werden können) sowie von *.doc-* und *.dot-*, *.xls-* und *.ppt-* sowie *.odt*-Dateien aus der Office-Familie.

Das Cloaking aktivieren oder deaktivieren Sie später im Dateifenster durch einen Klick mit der rechten Maustaste auf die Datei oder direkt auf eine einzelne Datei und die anschließende Auswahl von CLOAKING • CLOAKING (bzw. CLOAKING DEAKTIVIEREN). Das rot durchgestrichene Symbol zeigt Ihnen, dass dieser Eintrag später nicht übertragen oder synchronisiert wird. Sinnvoll ist dies z. B. bei Testdateien oder Entwicklungsverzeichnissen, in denen Sie persönlich für die Site entwickeln, die aber nicht im Projekt benutzt werden.

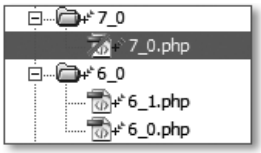


Abbildung 4.16 Aktiviertes Cloaking

4.2.9 Design Notes

Für eigene Notizen oder für die Gruppenarbeit sind Design Notes eine große Hilfe. Sie bilden quasi eine Projektpinnwand. Zu jeder einzelnen Datei und zu jedem Ordner lassen sich darauf Hinweise abspeichern und zentral sammeln. Jeder an der Site Beteiligte kann die Notizen abrufen und bearbeiten.

Wenn Sie Design Notes einsetzen möchten, aktivieren Sie in der Site-Verwaltung das Kontrollfeld Design Notes verwalten ❶. Sollen die Notes in der Gruppe eingesetzt werden, wählen Sie zudem das Kontrollfeld Bereitstellen von Design Notes für gemeinsame Nutzung aktivieren ❷ aus.

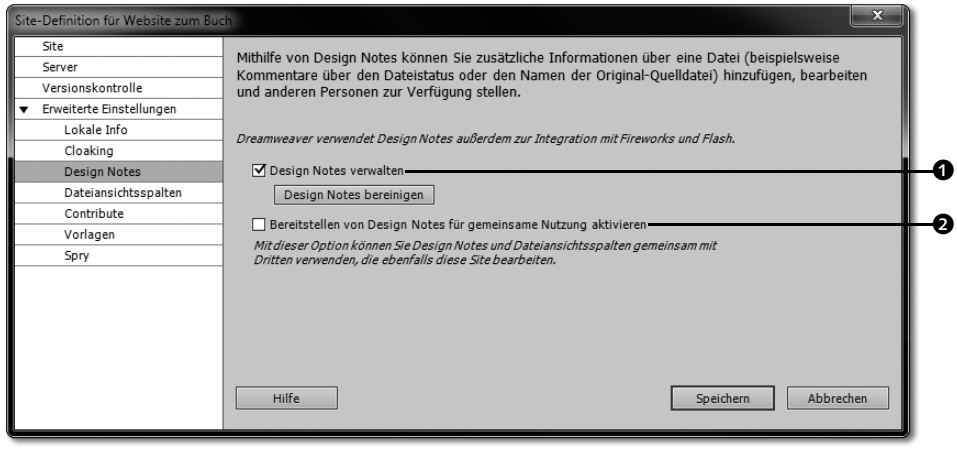


Abbildung 4.17 Design Notes aktivieren

Design Notes anlegen

Anlegen können Sie Design Notes durch einen Klick mit der rechten Maustaste auf eine Datei oder einen Ordner und Auswahl des Menüpunkts Design Notes. Es erscheint dann das Fenster aus Abbildung 4.18.

Durch Aktivieren des Kontrollfeldes Zeigen, wenn Datei geöffnet wird ❸ wird automatisch ein Hinweis eingeblendet, wenn man die verlinkte Datei öffnet. In der Praxis kann das viele Mails und Telefonate ersparen.

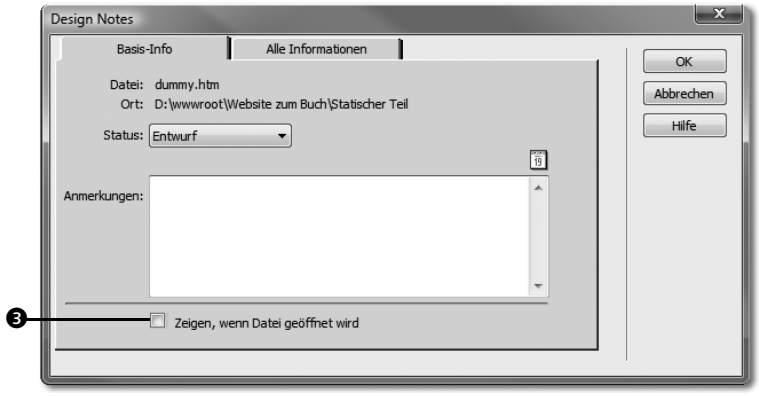


Abbildung 4.18 Eingabe von Design Notes

Speicherort

Dreamweaver legt für die Design Notes den versteckten Ordner *_notes* an. Möchten Sie den Ordner unter Linux auf dem Webserver sichtbar machen, entfernen Sie den Unterstrich im Dateinamen. Wenn Sie diesen Ordner wieder auf den Arbeitsrechner zurückkopieren, müssen Sie den Unterstrich allerdings wieder anfügen. Dreamweaver erkennt sonst den Ordner für Design Notes nicht an.

4.2.10 Dateiansichtsspalten

Unter diesem Menüpunkt können Sie die Darstellung im Dateifenster verändern. Interessanter ist jedoch die Möglichkeit, selbst Spalten zu definieren und diese zur gemeinsamen Nutzung bei der Gruppenarbeit bereitzustellen. Auch andere Benutzer der Arbeitsgruppe haben dann Zugriff auf die hinterlegten Informationen.

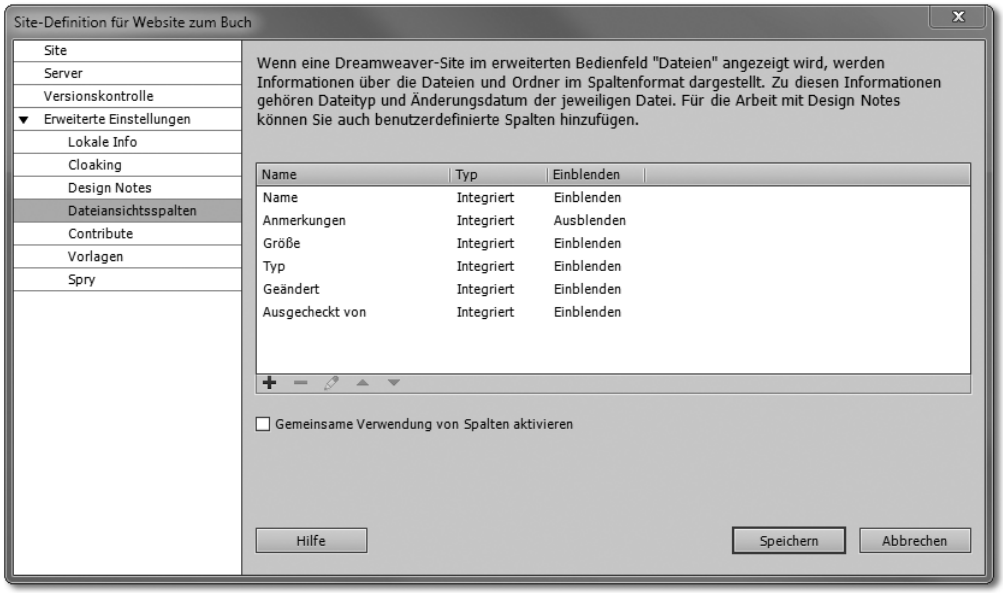


Abbildung 4.19 Einstellungen für Dateiansichtsspalten

4.2.11 Weitere Einstellungen

Die weiteren Einstellungen in der Site-Verwaltung für Contribute, Vorlagen und Spry folgen an den jeweils relevanten Stellen im Buch. Meist sind es hier Pfadangaben, die besetzt bzw. modifiziert werden können.

4.2.12 Fehlerquellen in der Site-Verwaltung

Wir haben bereits mehrfach auf mögliche Fehlerquellen in der Site-Verwaltung hingewiesen, so dass eigentlich nichts mehr schiefgehen kann. Im Prinzip ist es wirklich ganz einfach, dennoch ist es – wie wir in Schulungen immer wieder feststellen – die häufigste Fehlerquelle. Leider kann man Fehler, die man hier verursacht, kaum noch nachträglich beheben. Wenn erst einmal alle Pfade falsch angelegt sind, ist es sehr mühselig, alles wieder zu korrigieren.

Dreamweaver unterstützt leider auch das Arbeiten ohne Site-Deklaration. In Ihrem eigenen Interesse sollten Sie jedoch immer eine Site anlegen, auch wenn das Projekt noch so klein und unscheinbar erscheint! Wir haben es bei Seminarteilnehmern oft erlebt, dass umfangreiche Sites erstellt wurden und bei der Präsentation auf anderen Rechnern nichts mehr funktionierte.

Je weniger Sie sich mit den Pfadangaben beschäftigen wollen, desto wichtiger ist dieser Schritt.

4.3 Die Site-Ansicht

Um in der Site-Verwaltung zu arbeiten, wechseln Sie zunächst vom Bedienfeld Dateien über das Icon rechts Einblenden und lokale Site sowie Remote-Site anzeigen in die Site-Ansicht.

In Abbildung 4.20 sehen Sie die Oberfläche der Site-Verwaltung.

Im Menü Anzeigen ❶ können Sie bereits angelegte Seiten auswählen oder über Sites verwalten neue Sites hinzufügen. Um sich mit einem eingerichteten Web- oder Development-Server zu verbinden, klicken Sie auf das Icon Stellt Verbindung zum entfernten Host her ❷. Die Dateiauflistung neu laden können Sie durch einen Klick auf Aktualisieren ❸ (oder F5).

Die verschiedenen Anzeigemöglichkeiten im linken Dateifenster können Sie mit den Icons Remote-Server ❹, Testserver ❺ und Repository-Dateien ❻ bestimmen.

Durch Klicken auf das Icon Datei(en) bereitstellen ❽ können Sie beliebige Dateien auf den Webserver übertragen. Mit Dateien Abrufen ❼ können Sie wiederum Dateien vom Webserver auf Ihren lokalen Rechner überspielen. Auschecken ❾ und Einchecken ❿ werden eingehend in Abschnitt 4.4.2 zur Gruppenfunktion beschrieben.

Die Funktion zum Synchronisieren ⓫ erlaubt das mehr oder minder automatische Abgleichen der Inhalte mit einem Web- oder einem anderen Server.

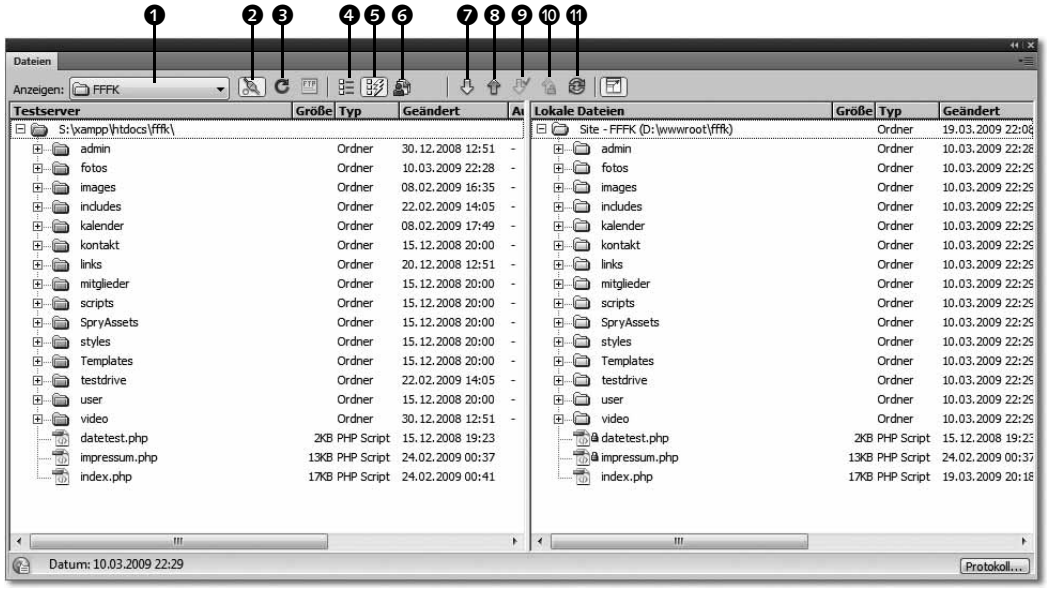


Abbildung 4.20 Site-Ansicht mit einer definierten Site

4.3.1 Synchronisieren und Vergleichen von Dateien

Dateien und Verzeichnisse können miteinander abgeglichen, also synchronisiert werden. Klicken Sie dazu in der Site-Verwaltung auf SYNCHRONISIEREN ⓫.

Anschließend können Sie in der in Abbildung 4.21 dargestellten Dialogbox über die Felder SYNCHRONISIEREN und RICHTUNG genau auswählen, welche Aktionen Sie ausführen möchten. Häufig werden Sie hier GESAMTE SITE und AKTUELLERE DATEIEN ABRUFEN UND BEREITSTELLEN anwählen.

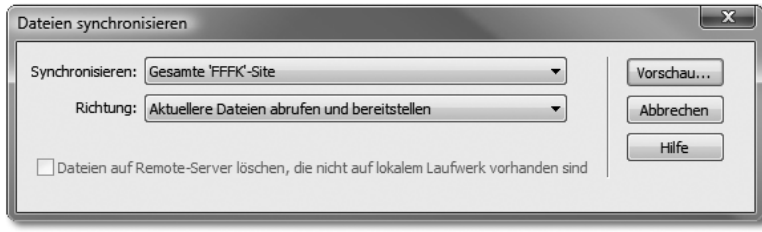


Abbildung 4.21 Aktion für die Synchronisation wählen

Dreamweaver ermittelt dann die entsprechend notwendigen Abgleichoperationen und zeigt eine Vorschau an, mit der Sie manuell Korrekturen vornehmen können.

Zu jeder Datei, die auf dem lokalen Entwicklungssystem und dem verbundenen Server unterschiedlich ist, stellt Dreamweaver einen Listeneintrag mit der Empfehlung der Operation her. Um diesen Vorschlag zu verändern, markieren Sie in der Liste die entsprechenden Dateien und klicken danach auf eines der unteren Symbole: ⓬ herunterladen (also vom verbundenen Server auf das lokale System), ⓭ hochladen (d. h. in die Gegenrichtung), ⓮ Löschen der Datei, ⓯ keine Aktion ausführen (die Datei wird dann bei der nächsten Synchronisation wieder in der Liste aufgeführt), ⓰ als synchronisiert betrachten, um dies auch später zu verhindern.

In den Voreinstellungen können Sie in der Kategorie DATEIEN VERGLEICHEN ein Programm zum Vergleich von Dateien angeben. Dieses wird durch Klicken auf das Icon ⓱ gestartet.

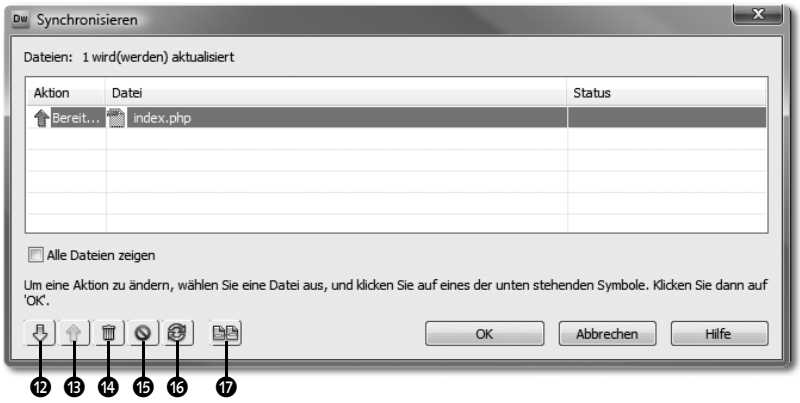


Abbildung 4.22 Dateivergleich starten

4.3.2 Mit der Dateiverwaltung arbeiten

Sie sollten nach dem korrekten Anlegen einer Site ausschließlich in der Site-Verwaltung oder im Dateifenster (was letztlich das Gleiche ist) arbeiten. Nur so kann sichergestellt werden, dass Dreamweaver Änderungen erfasst und entsprechende Aktionen zum Update von Verlinkungen usw. ausführen kann.

Dateiaktionen

Die meisten Dateiaktionen finden Sie im Kontextmenü über die rechte Maustaste (siehe Abbildung 4.23). Auch eine Auswahl nach neueren oder zuletzt bearbeiteten Dateien ist so möglich. Die Aktionen zum Löschen, Kopieren und Verschieben von Dateien verstecken sich hinter dem Untermenüpunkt BEARBEITEN.

Das Umbenennen einer Datei geschieht am praktischsten mit F2. Ebenso kann eine Datei mit Entf gelöscht werden. Bei beiden Aktionen warnt Sie Dreamwea-

ver gegebenenfalls, wenn davon Hyperlinks in anderen Dateien betroffen sind und diese automatisch (oder per Hand) umbenannt werden müssen oder gar verwaisen.

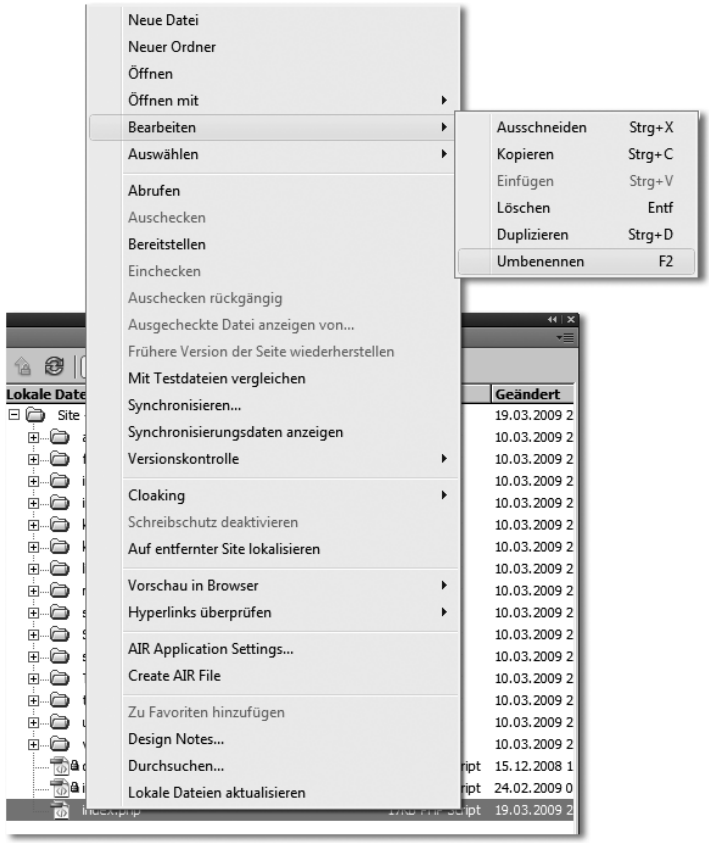


Abbildung 4.23 Dateiaktionen aus dem Kontextmenü

Protokoll aller Dateiaktionen

Alle Dateiaktionen werden protokolliert. Sie können dieses Protokoll auswerten, indem Sie rechts unten im Dateifenster oder in der Site-Verwaltung auf Protokoll klicken.

4.4 Gruppenarbeit

In Agenturen und Netzwerken von Freelancern wird häufig mit mehreren Personen an einer Site gearbeitet. Designer, Programmierer und Datenbankspezialisten – alle müssen auf die gleichen Dateien zugreifen. Wenn keine geeigneten

Mittel eingesetzt werden, um diese Vorgänge zu überwachen, ist Ärger und Verstimmung der Mitarbeiter vorprogrammiert.

Koordiniertes Arbeiten

Dreamweaver bietet eine gute Möglichkeit, diese Gruppenarbeit zu koordinieren. Der Mechanismus ist denkbar einfach: Dateien, die gerade bearbeitet werden, werden für andere Projektbeteiligte gesperrt.

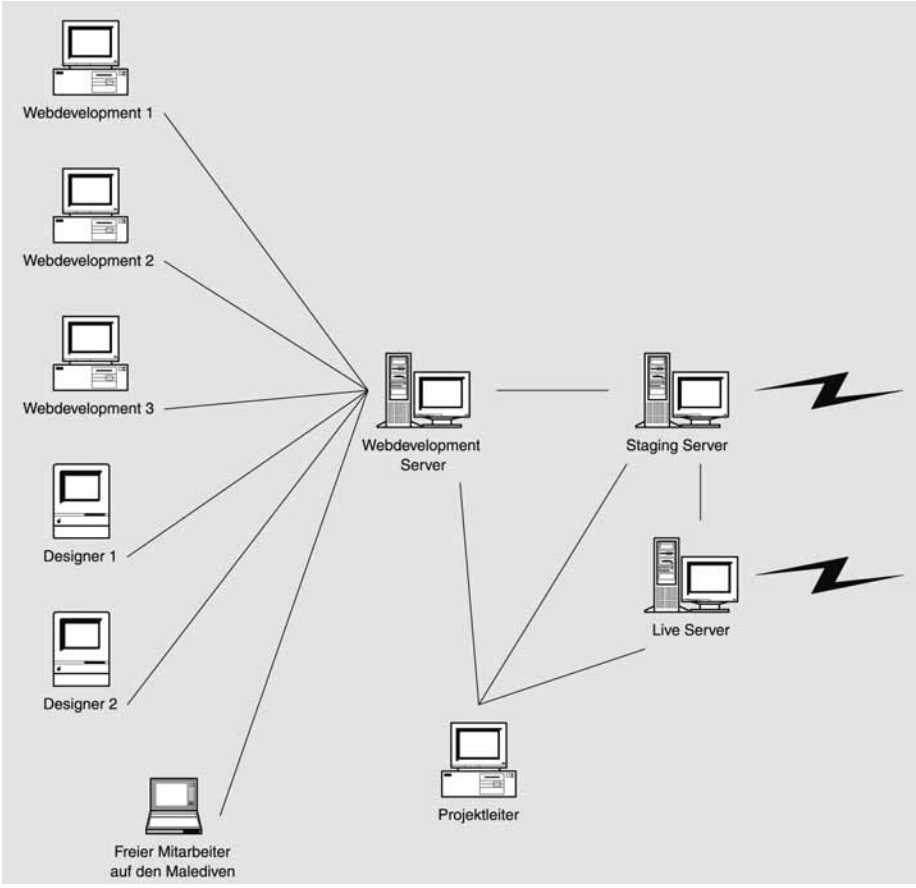


Abbildung 4.24 Beispiel für an einem Projekt beteiligte Arbeitsgruppen in einer Agentur

Die Gruppenfunktionalität von Dreamweaver arbeitet nach folgendem Prinzip: Zuerst richtet jeder Benutzer seinen Zugang auf den Server in Dreamweaver ein. Dann wird ein lokales Stammverzeichnis auf dem Rechner von jedem Gruppenmitglied angelegt. In diesem wird je nach Vorgabe eine Kopie der gesamten Site oder einzelner Dateien der Site angelegt.

Lokale Kopien

Direkt auf dem zentralen Server wird nie gearbeitet. Wenn Sie eine Datei bearbeiten möchten, kopiert Dreamweaver diese auf Ihren lokalen Rechner und sperrt sie gleichzeitig für alle anderen Benutzer. Egal, was Sie jetzt mit der Datei machen, solange Sie diese Datei nicht wieder freigeben und auf den Server kopieren, liegt das Original immer noch unverändert vor. Erst nach Abschluss der Arbeiten wird die alte Datei mit der neuen auf dem Server überschrieben und für andere Benutzer freigegeben.

Vergisst nun ein Mitarbeiter, die Dateien wieder bereitzustellen, kann man die Sperre auch ausschalten. Dreamweaver fragt dann mehrfach nach und gibt sie letztlich frei. Ansonsten wäre etwa ein beurlaubter und vergesslicher Kollege fatal für ein Projekt.

4.4.1 Remote-Zugriff einstellen

Um die Einstellungen für Gruppenarbeit zu erklären, simulieren wir in unserem Beispiel eine Arbeitsgruppe und gehen von folgender Situation aus:

Die zu bearbeitende Website liegt zentral auf einem Development- oder direkt auf dem Staging-Server. Das ist nichts anderes als ein zentraler Rechner, auf den alle Beteiligten über ein Netzwerk Zugriff haben. Dabei spielt es zunächst für die Gruppenfunktion in Dreamweaver keine Rolle, ob der Zugriff über LAN, FTP, SFTP, WebDAV oder andere Protokolle und Technologien erfolgt.

Mehrere Benutzer sollen diese Site nun parallel bearbeiten können. Folgende Schritte sind notwendig, um eine entsprechende Arbeitsumgebung in Dreamweaver zu schaffen:

1. Legen Sie eine neue Site mitsamt Ordnern auf Ihrem lokalen Rechner an.
2. Klicken Sie in der Site-Verwaltung auf SERVER.
3. Wählen Sie jetzt unter dem Punkt VERBINDEN ÜBER die Option LOKAL/NETZWERK aus.
4. Wählen Sie den Ordner der Site auf dem Development-Server aus.
5. Wählen Sie ERWEITERT aus, und aktivieren Sie SYNCHRONISIERUNGSINFORMATIONEN BEIBEHALTEN ❶, damit Sie immer aktuelle Dateien angezeigt bekommen.
6. Deaktivieren Sie DATEIEN BEIM SPEICHERN AUTOMATISCH AUF DEN SERVER HOCHLADEN ❷.
7. Aktivieren Sie DATEI-AUSCHECKEN AKTIVIEREN ❸.

8. Aktivieren Sie Dateien beim Öffnen Auschecken ❹. Sie können die Dateien auf dem Server im schreibgeschützten Modus auch einfach nur ansehen. Solange sie nicht ausgecheckt sind, können Sie sie nicht bearbeiten. Mit dieser Option sparen Sie sich viele Klicks und aufwendiges Suchen beim Auschecken.
9. Name für das Auschecken ❺ ist eine etwas unglückliche Übersetzung. Gemeint ist damit der Name, der neben der Datei angezeigt wird, wenn Sie diese zur Bearbeitung freigegeben haben, also Ihr persönlicher Name.
10. Geben Sie am besten Ihre E-Mail-Adresse ❻ an. Damit sind Sie immer ohne langes Nachschlagen in Listen und Adressbüchern durch Klick auf Ihren Namen per Mail erreichbar.

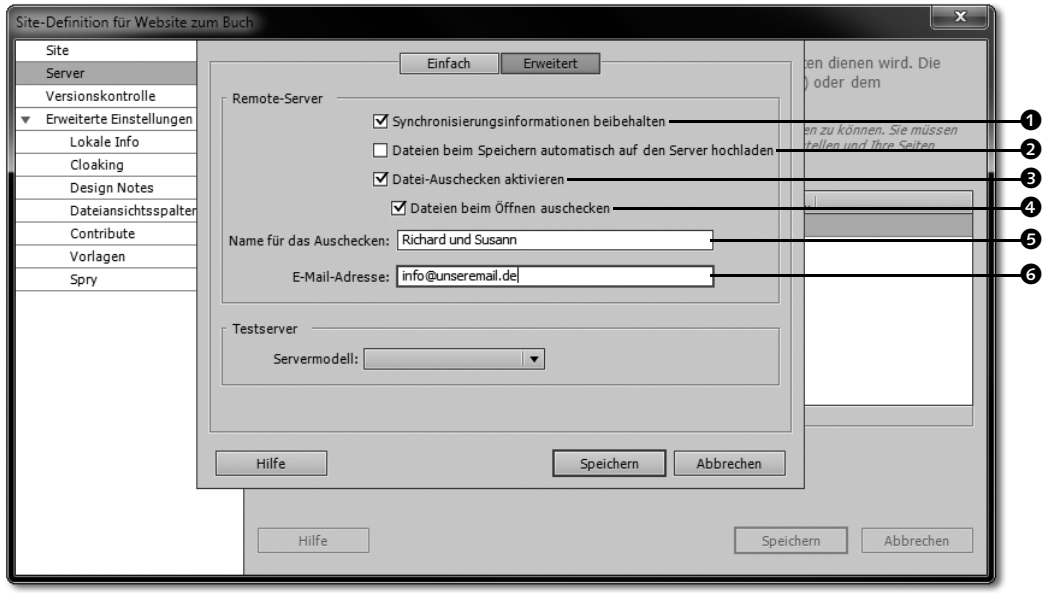


Abbildung 4.25 Remote-Zugriff für die Gruppenarbeit einstellen

Speichern auf Server deaktivieren [!]

Wenn Sie die Option Dateien beim Speichern automatisch auf den Server hochladen aktivieren, setzen Sie einen Großteil der Gruppenfunktionen außer Kraft. Sie arbeiten dann trotz angelegter lokaler Site auf dem Server. Genau das soll vermieden werden. Dateien sollten erst nach eingehender Kontrolle auf dem Development-Server erscheinen.

Wenn Sie Ihre Site für die Gruppenarbeit erfolgreich eingerichtet haben, müsste Ihre Site-Verwaltung in etwa wie in Abbildung 4.25 aussehen. Die gleichen Arbeitsschritte müssen nun der Reihe nach alle Teammitglieder erledigen. Kollegen,

die von außerhalb zuarbeiten, können sich trotzdem an der Gruppenarbeit beteiligen. Sie richten dafür einfach kein lokales Netz ein, sondern greifen per FTP auf den Development-Server zu. Voraussetzung ist natürlich, dass darauf ein FTP-Server installiert ist und ein Zugang zum Internet besteht.

4.4.2 Mit der Gruppenfunktion arbeiten

Dateien synchronisieren

Wenn sich die Website auf Ihrem lokalen Rechner befindet, müssen Sie diese auf den Server laden. Befindet sie sich auf dem Server, laden Sie sie auf Ihr lokales System herunter. Klicken Sie für beide Vorgänge im Dateifenster einfach auf AKTUALISIEREN ❶, oder drücken Sie F5. Dreamweaver gleicht dann den Status der Website auf dem lokalen und dem Remote-System ab.

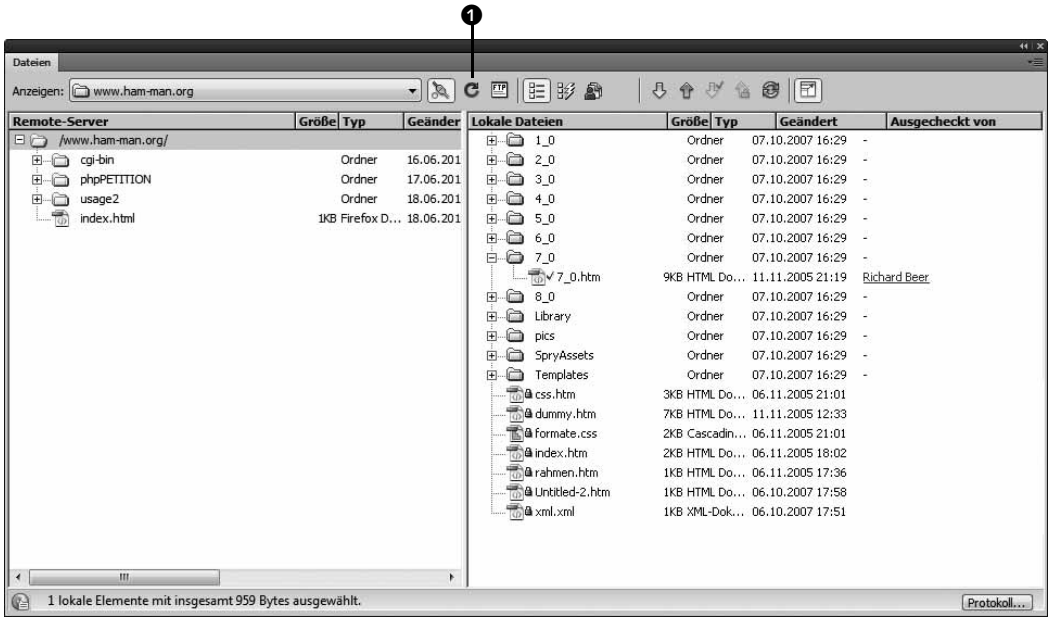


Abbildung 4.26 Dateien synchronisieren

Dateien lokal bearbeiten

Das Auschecken von ausgewählten Dateien zur Bearbeitung erreichen Sie, wenn Sie diese markieren und dann auf DATEIEN AUSCHECKEN klicken. Dreamweaver fragt Sie dann, ob alle anhängenden Dateien ausgecheckt werden sollen. Wenn Sie mit externen Stylesheets arbeiten und z. B. Formatierungen im Dokument bearbeiten wollen, muss Dreamweaver auf diese externen Dateien zugreifen. In solchen Fällen sollten Sie daher anhängende Dateien mit auschecken.

Dateien auf den Server hochladen

Nach dem Bearbeiten müssen Sie die Dateien mit EINCHECKEN wieder auf den Server übertragen. Wenn Sie die Site-Verwaltung erstmals einrichten, müssen Sie zunächst *alle* Dateien auf dem Server bereitstellen, damit der Schutz wirksam wird. Der Standardzustand sind eingecheckte und damit geschützte Dateien.

Die Dateien auf dem Development-Server sehen Sie auf der linken Seite. Auf der rechten Seite werden die Dateien dargestellt, die sich auf Ihrem lokalen System befinden. Rechts neben den Dateien wird der Name des Mitarbeiters angezeigt, der sie gerade bearbeitet. Auch im linken Fenster wird angezeigt, welcher Benutzer gerade eine Datei bearbeitet.

Sollte neben der Datei kein Name angezeigt werden, obwohl die Datei ausgecheckt ist, wechseln Sie in die Site-Einstellungen und legen unter DATEISPALTENANSICHT fest, dass diese Spalte angezeigt werden soll.

Haben Sie eine E-Mail-Adresse angegeben, können Sie durch Klicken auf diesen Namen eine E-Mail mit dem Dateinamen als Betreff erstellen. Das erhöht den Austausch zwischen den Entwicklern und die E-Mail-Flut in Unternehmen.

Eingecheckte Dateien werden mit einem kleinen Schloss dargestellt. Dateien, die Sie ausgecheckt haben, sind mit dem kleinen grünen Häkchen versehen. Das kleine Schloss vor dem Dateinamen signalisiert, dass die Datei gerade für Sie gesperrt ist.

Dreamweaver bietet keine echte Versionierung. [«]

Die Gruppenfunktionen von Dreamweaver CS5 stellen keinen Ersatz für eine Versionierungssoftware dar. Dreamweaver schützt auf eine recht simple Art Dateien vor dem Überschreiben, indem der Schreibschutz aktiviert und zu bestimmten Zeitpunkten deaktiviert wird.

Eine Versionierungssoftware, wie etwa das direkt in Dreamweaver einzubindende Open-Source-Projekt Subversion (*http://subversion.tigris.org/*), erlaubt es, bestimmte Entwicklungszustände eines Projekts »einzufrieren« (Codefreeze) und jederzeit zu jedem beliebigen Entwicklungsstand zurückzukehren.

4.5 Die Buchwebsite anlegen

Nachdem wir nun Grundlegendes zur Site-Verwaltung kennengelernt haben, können wir unsere erste eigene Site anlegen. Mit dieser werden wir dann innerhalb des Buches arbeiten.


Buchwebsite

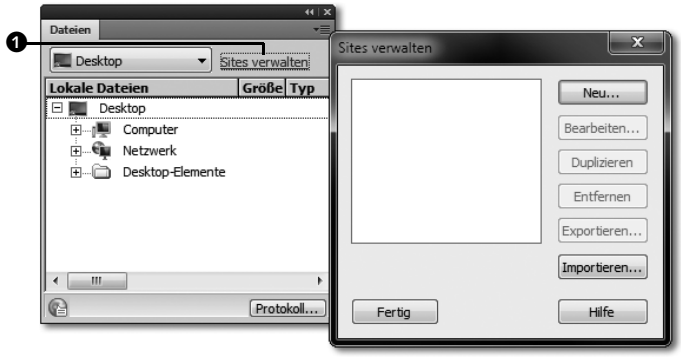
[▪] *Schritt-für-Schritt: Eine Site anlegen*

1 *Ordner für die Buchwebsite*

Legen Sie auf einem lokalen Laufwerk Ihrer Wahl einen neuen Ordner für die Buchwebsite an.

2 *Menü aufrufen*

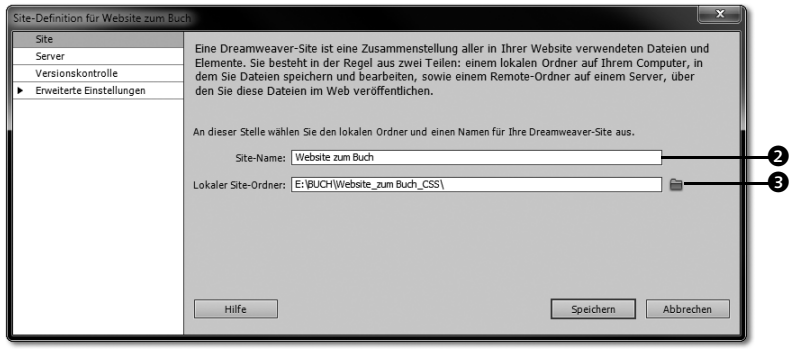
Es gibt mehrere Möglichkeiten, in das Menü zum Anlegen einer neuen Site zu gelangen. Wenn Sie Dreamweaver das erste Mal starten, sehen Sie das Dateifenster wie in der folgenden Abbildung links. Klicken Sie im Bedienfeld DATEIEN auf den Link SITES VERWALTEN ❶. Es erscheint die Dialogbox, die Sie in der Abbildung rechts sehen. Klicken Sie dort auf NEU….



3 *Site-Verwaltung*

Jetzt öffnet sich die erste Eingabemaske der Site-Definition.Welchen Namen ❷ Sie vergeben, bleibt Ihnen überlassen. Wählen Sie außerdem den im ersten Schritt angelegten Ordner als lokalen Site-Ordner ❸ aus.

Weitere Einstellungen sind in der Site-Verwaltung zunächst nicht zu erledigen.



4 *Bilder kopieren*

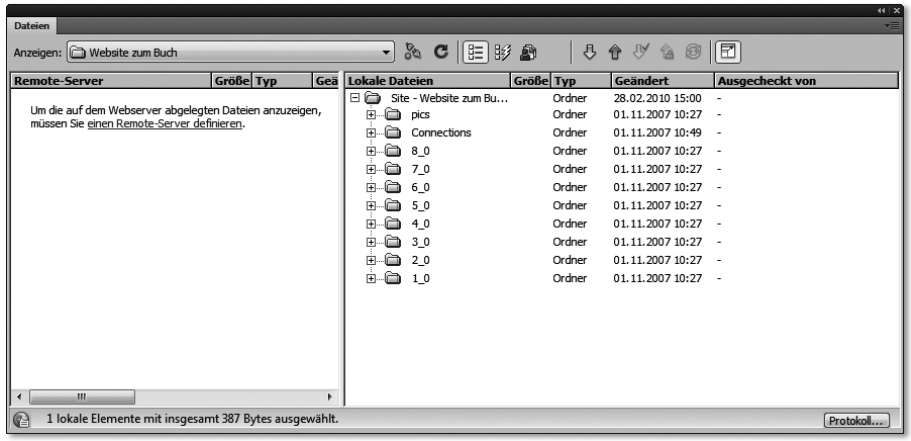
Kopieren Sie über den Windows Explorer den Ordner *Pics* von der DVD aus dem Verzeichnis *Buchwebsite* in den von Ihnen angelegten lokalen Ordner. Dies sollte dann die letzte Dateiaktion sein, die Sie außerhalb von Dreamweaver durchführen. [o]

5 *Weitere Ordner anlegen*

Legen Sie jetzt in der Site-Verwaltung die Ordner *1_0* bis *8_0* an. Sie können neue Ordner anlegen, indem Sie mit der rechten Maustaste in die Ordnerstruktur im Dateifenster klicken und dann aus dem folgenden Menü den Eintrag NEUER ORDNER auswählen. Die Dateien Ihrer Site sehen jetzt aus, wie in der folgenden Abbildung dargestellt. Eventuell haben die Dateien auf der DVD durch den Kopiervorgang einen Schreibschutz. Heben Sie diesen auf, bevor Sie mit der Site arbeiten. Sie erkennen den Schreibschutz in Dreamweaver an einem kleinen Schloss vor dem Dateinamen. Diesen Schreibschutz können Sie durch einen Klick mit der rechten Maustaste im Windows Explorer und Deaktivieren des Kontrollkästchens SCHREIBGESCHÜTZT aufheben.

6 *Arbeiten kontrollieren*

Nach dem Abschluss der Arbeiten sollte Ihre Site-Verwaltung aussehen wie in der folgenden Abbildung:



Sie haben nun eine Site angelegt und möchten Ihr erstes HTML-Dokument erstellen. In diesem Kapitel erfahren Sie, welche Grundeinstellungen zu beachten sind und wie Sie dieses Dokument als Basis für alle weiteren Arbeiten nutzen können.

5 Grundlegende Dokumenteinstellungen

Wenn Sie eine Website erstellen, ist es sinnvoll, ein Dokument mit all denjenigen Grundeigenschaften anzulegen, die Sie über die ganze Site hinweg beibehalten wollen. Dieses Dokument können Sie später in Dreamweaver als Template (Vorlage) nutzen. So gehen Sie sicher, dass alle Dokumente die gleichen Rahmenbedingungen aufweisen, also über identische Hintergrundfarben, Layoutvorgaben, Spracheinstellungen und Metaangaben verfügen.

Vier Wege führen in das Menü für die Seiteneinstellungen:

- Modifizieren • Seiteneigenschaften
- Gleichzeitiges Drücken der Tasten Strg/⌘+J
- Drücken der rechten Maustaste im Dokument und Auswählen des Menüpunkts Seiteneigenschaften
- Klicken auf den entsprechenden Button in der Eigenschaftenpalette

5.1 Seiteneigenschaften festlegen mit CSS

Während wir in der Vorauflage dieses Buches noch die Einstellungen für HTML als Vorgabe gewählt hatten, kann man nun ohne Probleme die Seiteneigenschaften mit CSS festlegen.

Standardschrift, Schriftgröße und Textfarbe

Bei den meisten Websites wird man mit einer Hauptschrift arbeiten. Hier können Sie festlegen, welche das sein soll. Auch wenn es heute mit CSS Möglichkeiten gibt, Schriften über den Webserver einzubinden, sollte man doch die Benutzer mit ein wenig älteren Browsern im Blick haben und sich auf eine der Webschriften (Arial, Verdana, Times und Courier) festlegen.

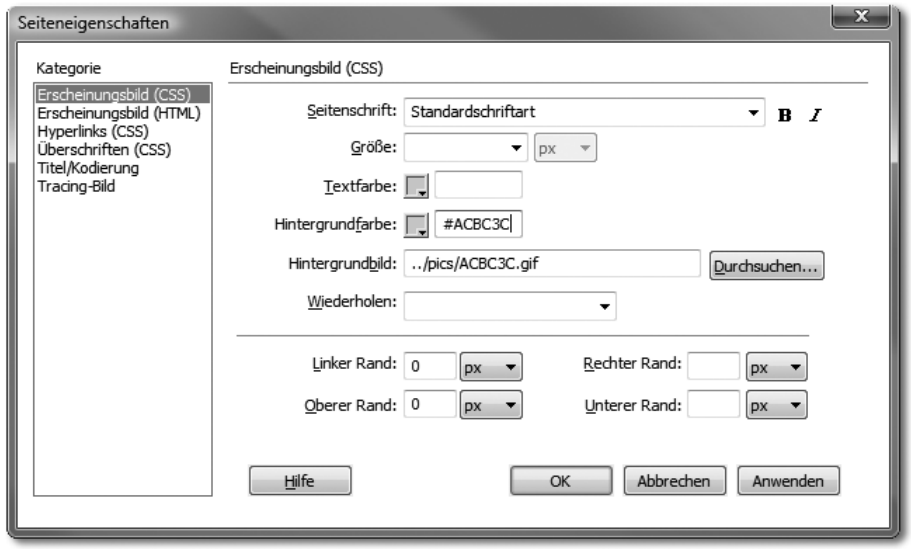


Abbildung 5.1 Fenster zum Einstellen der Seiteneigenschaften mit CSS

Textfarbe und Schriftgröße können hier ebenfalls direkt eingestellt werden. Wenn Sie eine Seitenschrift festgelegt haben, benötigen Sie weitere CSS-Stile zur Schriftformatierung nur dann, wenn Sie von diesem Standard abweichen möchten.

Eigene Hintergrundfarbe

Einheitliche Hintergrundfarben können Sie in den Seiteneinstellungen definieren. Dies ist eine Pflichtangabe. Sie können sie zwar weglassen – die Seite wird dann aber mit weißem Hintergrund angezeigt. Die weiße Hintergrunddarstellung funktioniert auch nur dann, wenn der Benutzer keine eigenen Farben im Browser verwenden möchte. Viele Designs für Firefox und Windows schalten die Standardfarben ab. Der Hintergrund im Browser ist dann einfach nur grau! Ein Hintergrundbild liegt übrigens immer über der Hintergrundfarbe.

[»] **Falsche Farben**

Auch wenn heute alle Browser die komplette Farbpalette darstellen können und Sie keineswegs auf sogenannte *websichere* Farben festgelegt sind, ist leider nicht sichergestellt, dass die einzelnen Farbtöne in unterschiedlichen Browsern gleich dargestellt werden. Verwenden Sie am besten zusätzlich zur Hintergrundfarbe ein Hintergrundbild im gewünschten Farbton.

Hintergrundbild und Anzeigevarianten

Ebenfalls mit CSS werden Hintergrundbilder festgelegt. Häufig arbeitet man mit 1 × 1 Pixel großen GIFs, um eine absolut eindeutige Hintergrundfarbe im Browser zu erhalten.

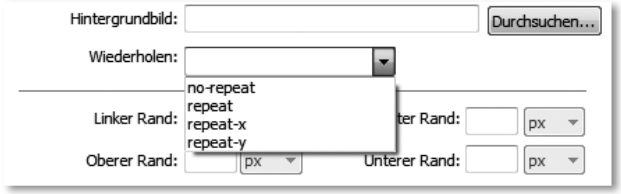


Abbildung 5.2 Bildwiederholungen angeben

Für jedes Hintergrundbild kann angegeben werden, ob und wie es wiederholt werden soll. Ohne diese Angabe wird eine Hintergrundgrafik »gekachelt«, also so oft wiederholt, wie es im Browserfenster möglich ist. Die Abbildungen 5.3 bis 5.6 zeigen die verschiedenen Wiederholungsvarianten.

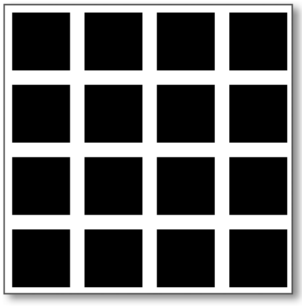


Abbildung 5.3 Standard oder background-repeat: repeat;

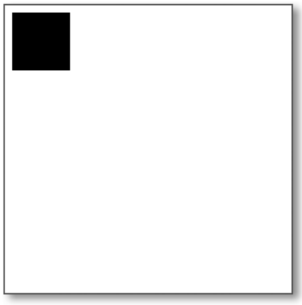


Abbildung 5.4 background-repeat: no-repeat;

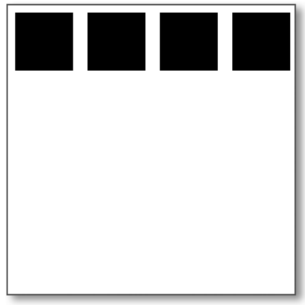


Abbildung 5.5 background-repeat: repeat-x;

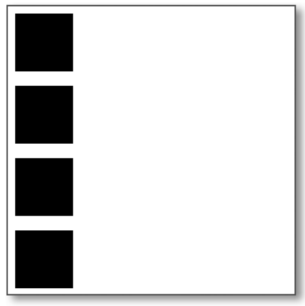


Abbildung 5.6 background-repeat: repeat-y;

Sie können sich sicherlich vorstellen, welch enorme Möglichkeiten sich durch diese Angaben und einige Photoshop-Tricks ergeben.

Ränder mit CSS einstellen

Für ein reines CSS-Layout spielen die Randvoreinstellungen kaum eine Rolle. Ab und an möchte man dennoch innerhalb eines mit CSS-Vorgaben erstellten Dokuments mit klassischen HTML-Elementen arbeiten. Dann ist es sinnvoll, den linken und oberen Rand auf 0 zu setzen, um einen eindeutigen Bezugspunkt zu bekommen. Auch für einige CSS-Layout-Tricks ist es notwendig, diese Angaben zu machen.

Die CSS-Definition sieht bis zu diesem Punkt wie folgt aus (wechseln Sie in die Code-Ansicht, um sie sehen zu können):

```
<style type="text/css">
<!--
body {
   background-image: url(pics/ACBC3C.gif);
   background-color: #ABBC40;
```

```
    margin-left: 0px;
    margin-top: 0px;

}
-->
</style>
```

Listing 5.1 CSS-Stil mit Seiteneinstellungen

5.1.1 Verknüpfungen & Überschriften

Sie können hier bereits Kontext-Selektoren für Hyperlinks anlegen. Diese werden ebenfalls in einem CSS im Head-Bereich des Dokuments abgelegt. Dasselbe gilt für Absatzformate. Näheres zu Kontext-Selektoren finden Sie in Abschnitt 11.3.1, »Selektortypen«.

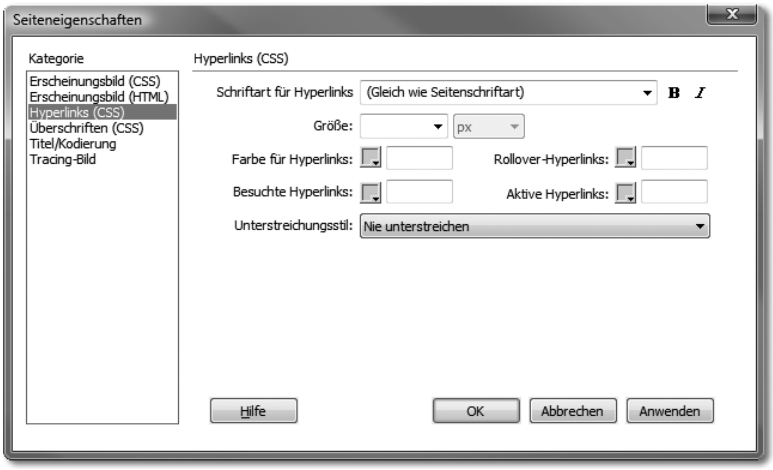


Abbildung 5.7 Einstellen von Kontext-Selektoren für Hyperlinks

Suchmaschinenoptimierung [«]

Suchmaschinen freuen sich auch heute noch über Formatierungen mit `<h1>` oder `<h2>`. Hier können Sie diesen Tags ganz einfach eine »normale« Schriftgröße zuweisen. Viele Suchmaschinen sind dann von der Wichtigkeit Ihres Textes überzeugt, es ist ja eine Überschrift erster Ordnung. Die Darstellung im Browser ist dann eine ganz andere. Für »blinde« Suchmaschinen also ideal.

CSS in externe Datei verschieben

Wenn Sie, wie wir vorschlagen, externe CSS-Stile verwenden, können Sie diese nach dem Anlegen des ersten Dokuments exportieren und somit gleich die externe Datei erstellen, auf die alle weiteren Dokumente zugreifen können. Sie

finden den entsprechenden Befehl, wenn Sie im CSS-Bedienfeld mit der rechten Maustaste auf den jeweiligen Stil klicken. Er nennt sich Regel verschieben. In Kapitel 11, »CSS und Dreamweaver«, werden wir hierauf vertieft eingehen.

Die Verwendung externer CSS-Dateien hat den Vorteil, dass Sie später nur an einer Stelle Änderungen durchführen müssen, die sich aber auf die ganze Site auswirken. So ist außerdem auch sichergestellt, dass tatsächlich die ganze Site die gleichen Design-Einstellungen benutzt. Eine eingebundene CSS-Datei erkennen Sie unter anderem daran, dass sie in der Leiste abhängiger Dokumente unter dem Datei-Reiter erscheint.

5.1.2 Titel/Codierung

Dokumenttyp angeben

Wie wir in späteren Kapiteln noch sehen werden, ist die Kodierung bzw. der Dokumenttyp für die korrekte Darstellung von enormer Wichtigkeit. Achten Sie in jedem Fall darauf, dass er korrekt angegeben ist. Bei neuen Websites sollten Sie in jedem Fall XHTML als Dokumenttyp einstellen.



Abbildung 5.8 Titel und Zeichensatz

Vergabe eines Titels

Geben Sie Ihren Seiten immer einen aussagekräftigen Titel. Nichts ist peinlicher, als Kundenseiten in den Suchmaschinen mit der Bezeichnung »Unbenanntes Dokument« zu finden. Für die Suchmaschinen ist das zudem ein wichtiges Kriterium, um die Seiten zu indizieren. Achten Sie möglichst auch darauf, nicht für jede Unterseite denselben Titel zu verwenden. Dennoch ist es sinnvoll, bei der

ersten Seite bereits einen Titel anzugeben. Sie können diesen dann immer noch verändern, aber nicht mehr vergessen.

Nicht sinnvoll sind Titel wie »Herzlich willkommen auf unserer neuen Internetpräsenz«. Erstens ist »neu« ja ziemlich relativ und sicher bald überholt. Und zweitens sollte der Titel tatsächlich aussagekräftig sein, denn Google und andere Suchmaschinen indizieren auch den Seitentitel. Eine sinnvolle Angabe wäre z. B. »Blumen Müller in Schifferstadt – Obst, Zierpflanzen und Blumengestecke«.

Im folgenden Beispiel sollten Sie sich auch nicht wiederfinden:



Abbildung 5.9 Google-Suche nach »Unbenanntes Dokument«

Zeichensatz angeben

Das Internet ist international. Trotz allem werden es in erster Linie Besucher aus Ihrem Kulturkreis sein, die auf Ihre Seiten zugreifen. Browser und Betriebssysteme unterstützen verschiedene Zeichensätze. Vielleicht haben Sie einmal versucht, auf eine japanische Seite zuzugreifen. Üblicherweise bietet Ihnen dann der Browser einen Zeichensatz zum Download an.

Der Browser kann nur wissen, welchen Zeichensatz er benötigt, wenn ihm das eindeutig mitgeteilt wird. Aus diesem Grund sollten Sie immer einen bestimmten Zeichensatz vorgeben. Im mitteleuropäischen Raum war das lange der Zeichensatz ISO-8859-1 (westeuropäisch Latin 1). Er ist bei alten Dreamweaver-Versionen auch als Standard definiert. Mittlerweile arbeitet man meist mit dem Zeichensatz UTF-8, der die Probleme der Lokalisierung nicht mehr kennt. Wenn Sie standardkonforme Websites erstellen wollen oder müssen, kommen Sie um UTF-8 nicht mehr herum.

5.1.3 Tracing-Bild – warum und wann?

Das Tracing-Bild ist eine Vorlage im Hintergrund des Dokuments, um darauf Tabellen, Layer usw. zu platzieren. Dieses Bild wird im Browser nicht dargestellt. Wir selbst haben dieses Feature in der Praxis bislang kaum genutzt. Wenn Sie Seiten erstellen wollen oder müssen, die sich beispielsweise am Layout von Printmedien ausrichten, kann es durchaus sinnvoll sein, dies einzusetzen. Wir werden in Kapitel 6, »Tabellen und Listen«, etwas genauer darauf eingehen.

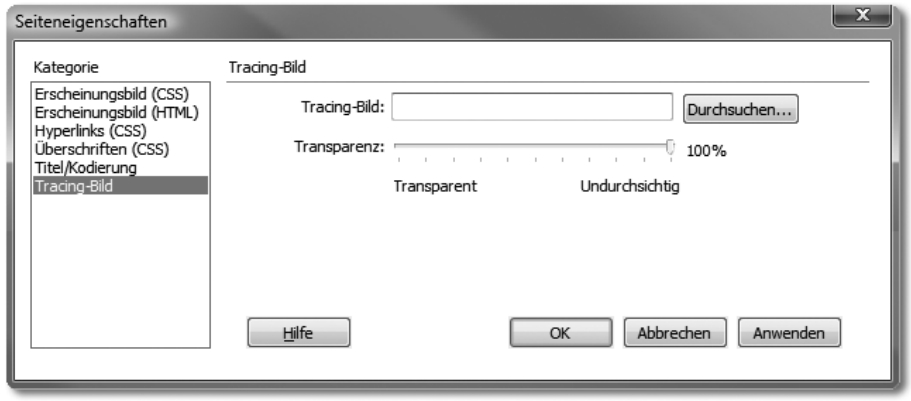


Abbildung 5.10 Tracing-Bild einfügen

5.2 Seiteneigenschaften festlegen ohne CSS

Wenn Sie beruflich oder privat mit älteren Content-Management-Systemen zu tun haben, werden Sie oft zwangsläufig auf »alte« Methoden zurückgreifen müssen. Deswegen zeigen wir hier auch noch das Festlegen der Seiteneigenschaften mit HTML.

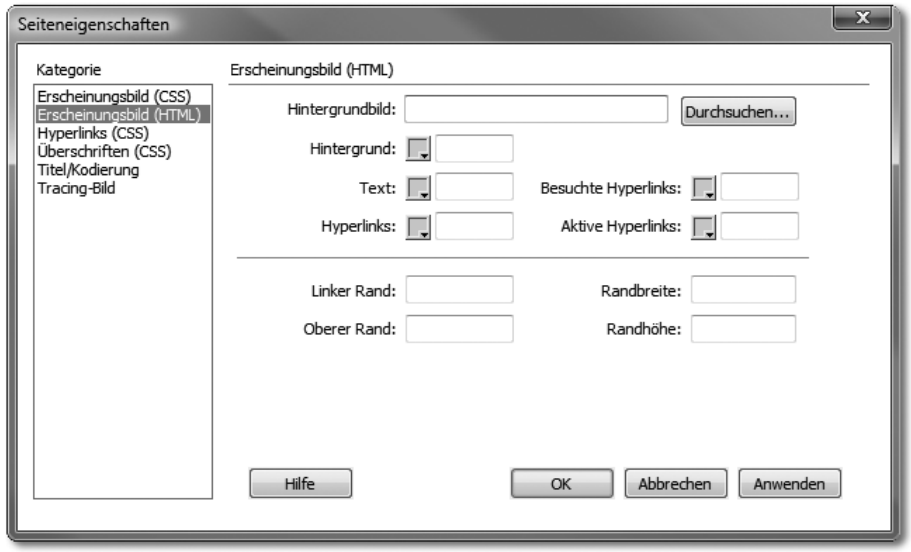


Abbildung 5.11 Fenster zum Einstellen der Seiteneigenschaften

Menüpunkt	Funktion
Hintergrundbild	Legt eine Grafikdatei fest, die als Hintergrund des gesamten Dokuments dargestellt wird. Standardmäßig wird diese Grafik »gekachelt«. Das heißt, sie wird so lange wiederholt, bis das Browserfenster vollständig ausgefüllt ist. Diesen Effekt machen wir uns für diverse gestalterische Tricks zu .
Hintergrund	Legt die Hintergrundfarbe der Seite fest.
Hyperlinks	Legt eine Standard-Textfarbe für Hyperlinks im `<body>`-Tag fest.
Besuchte Hyperlinks	Legt eine Standard-Textfarbe für besuchte Hyperlinks im `<body>`-Tag fest.
Aktive Hyperlinks	Legt eine Standard-Textfarbe für aktive Hyperlinks im `<body>`-Tag fest.
Linker Rand	`<body leftmargin="0">` Setzt den linken Rand für den IE.
Oberer Rand	`<body topmargin="0">` Setzt den oberen Rand für den IE.
Randbreite	`<body marginwidth="0">` Setzt den linken Rand für Netscape.
Randhöhe	`<body marginheight="0">` Setzt den oberen Rand für Netscape.

Tabelle 5.1 Grundlegende Seiteneinstellungen

Ränder in IE und Netscape

Wie Sie in Tabelle 5.1 sehen, unterstützen beide Browser für die Einstellung der Randbreiten völlig verschiedene Befehle mit den gleichen Auswirkungen. Damit in allen Browsern ein einheitliches Erscheinungsbild gesichert ist, müssen beide Varianten angegeben und auf 0 gestellt werden. Glücklicherweise ignorieren die Browser in diesem Fall den ihnen nicht bekannten Befehl. Das vollständige Tag lautet:

```
<body leftmargin="0" topmargin="0" marginwidth="0" marginheight="0">
```

Bezugspunkt bei Tabellenlayouts definieren

Wenn Sie den noch nicht eingestellten Arbeitsbildschirm betrachten, werden Sie feststellen, dass der Cursor links oben blinkt. Er ist weder genau randbündig noch an anderer definierter Stelle. Wenn Sie ein gutes und exaktes Layout erreichen wollen, können Sie sich ein »ungefähr« hier nicht erlauben. Für Tabellen wird ein genauer Bezugspunkt benötigt, der immer links oben bei 0 × 0 Pixel liegt.

Die Ränder werden im Dokument so exakt definiert. In den meisten Fällen sind das 0 Pixel. Auf diese Weise existiert ein fester Punkt, von dem aus das Layout aufgebaut werden kann.

Deprecated Code

Unter der Bezeichnung *deprecated* führt man Code, der zwar noch gültig ist, aber nicht mehr verwendet werden sollte. Die Angaben für die Ränder und Hintergründe einer Seite sollten nicht mehr mit den obengenannten HTML-Tags formatiert werden, da hierbei Funktion und Design untrennbar miteinander verknüpft werden. In XHTML sind diese Tags bzw. Attribute gar nicht mehr gestattet.

Dennoch werden diese Programmierungen auch heute noch von älteren Sites verwendet. Wie es moderner geht, haben wir Ihnen am Anfang des Kapitels gezeigt.

5.3 Metaangaben

Metaangaben für Suchmaschinen gehören ebenfalls zu den grundlegenden Seiteneigenschaften. Es ist sinnvoll, diese Angaben bereits beim Anlegen des Grundlayouts einzugeben. Sie stehen Ihnen dann auf allen weiteren aus dem ersten Dokument erstellten Seiten gleich zur Verfügung und können gegebenenfalls modifiziert werden.

Head-Tags einfügen

Sie können einige Metaangaben in Dreamweaver direkt eingeben (siehe Abbildung 5.12). Uns ist allerdings nicht ganz klar, nach welchen Kriterien Adobe die `<head>`-Tags integriert hat. Nur zwei von ihnen sind für uns wirklich relevant, und andere wichtige erscheinen gar nicht erst. Ein vollständiger und für Suchmaschinen ausreichender Satz an Metaangaben sieht folgendermaßen aus:

```
<title>Ihr Titel</title>
<meta http-equiv="Content-Type" content="text/html;
charset=iso-8859-1">
<meta name="robots" content="index">
<meta name="robots" content="follow">
<meta name="language" content="de">
<meta name="keywords" content="Ihre, Suchbegriffe, durch, Komma,
getrennt">
<meta name="description" content="Eine Beschreibung der Seite">
<meta name="distribution" content="global">
```

```
<meta name="robots" content="all">
<meta name="revisit-after" content="10 days">
```

Listing 5.2 Sinnvoller Satz an Metaangaben

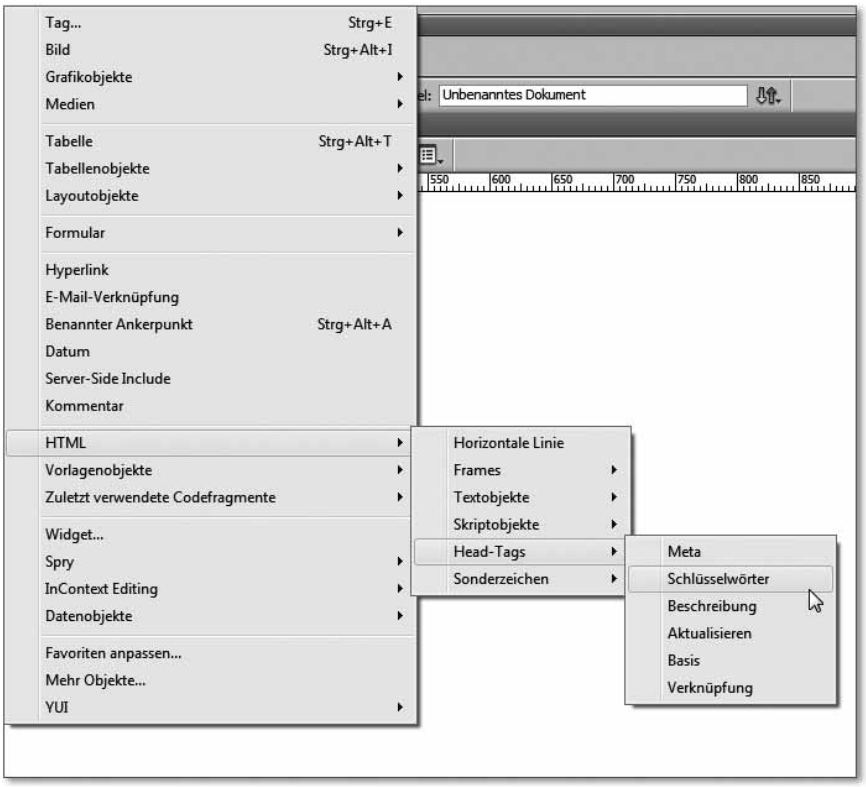


Abbildung 5.12 Head-Tags mit Dreamweaver einfügen

Head-Inhalte anzeigen

Im Menü unter Ansicht Head-Inhalt oder mit Strg+⇧+H können Sie die Symbolleiste für Head-Inhalte einblenden (siehe Abbildung 5.13).

Durch Klicken auf die entsprechenden Icons können Sie die Metaangaben nachträglich in den Eigenschaften modifizieren. Leider erscheinen die Icons erst, wenn man über das Menü die Metaangaben einmal angelegt hat.

Hilfreiche Erweiterungen

Auf der Website von Adobe Exchange finden Sie einige schöne Erweiterungen wie z.B. Meta Generator, um Metaangaben deutlich komfortabler einzufügen (siehe Abbildung 5.14).

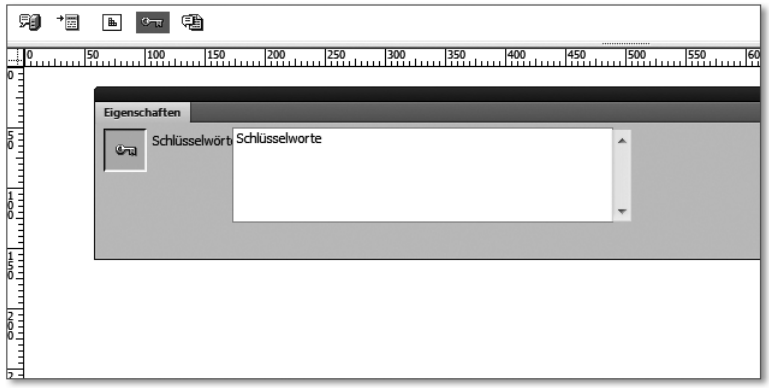


Abbildung 5.13 Symbolleiste »Head-Inhalte«

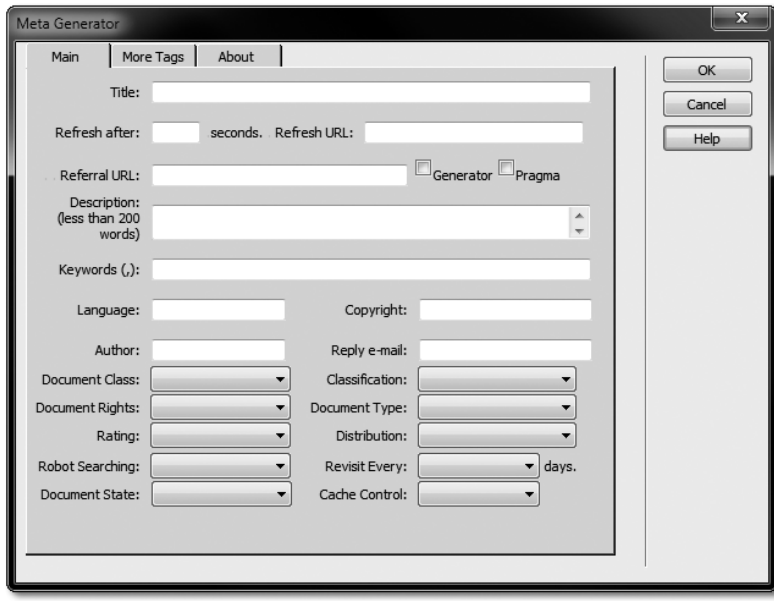


Abbildung 5.14 Extension für Metaangaben

Dublin-Core

Es gibt noch eine ganze Reihe weiterer Metaangaben. Für die allermeisten Fälle sollten diese jedoch ausreichen. Falls Sie für wissenschaftliche Publikationen eine Website erstellen oder generell im wissenschaftlichen Bereich arbeiten, legen wir Ihnen nahe, sich mit Metaangaben nach Dublin-Core zu befassen. Dies ist ein Standard, mit dem wissenschaftliche Publikationen ähnlich wie ISBN-Nummern bei Büchern erfasst und indiziert werden. Eine ausführliche Aufstellung finden Sie unter *http://de.selfhtml.org/html/kopfdaten/meta.htm*.

Tabellen dienen dazu, strukturierte Inhalte, wie Produktdaten oder Adressverzeichnisse, aufzunehmen und geordnet anzuzeigen.

6 Tabellen und Listen

Nachdem Tabellen jahrelang das Mittel der Wahl beim Aufbau eines Layouts waren, soll man sie nach heutigen Webstandards ausschließlich zur Darstellung von Daten in einer tabellarischen Ansicht verwenden. Es gibt allerdings ein paar Ausnahmen, die im nächsten Abschnitt diskutiert werden.

6.1 Layouttabellen – pro und contra

Layouttabellen haben den Nachteil, dass Design und Inhalte (Daten) vermischt werden und für eine eventuelle Weiterverarbeitung nicht mehr trennbar sind. Man sollte also das Einsatzgebiet der zu erstellenden Website genau kennen. Soll die Website zu 100 Prozent den XHTML-Spezifikationen entsprechen und ist eine Verwertung durch andere Ausgabemedien geplant (z. B. durch eine Software, die ein PDF daraus erstellt), verbieten sich Tabellenlayouts. Tabellen sollten dann wirklich nur für die tabellarische Darstellung von Daten verwendet werden.

Das Gleiche gilt für barrierefreie Websites, die z. B. mit Bildschirmlesegeräten ausgegeben werden sollen. Ein weiterer Nachteil der Tabellen ist die sehr schlechte Pflegbarkeit von Tabellenlayouts. Nehmen Sie an, Ihre Website besteht aus 500 HTML-Dokumenten. Nun sollen einzelne Bereiche 10 Pixel nach links verschoben werden. Bei einem Tabellenlayout müssen Sie 500 Dokumente einzeln bearbeiten – mit CSS meist nur ein einziges.

Tabellen haben allerdings durchaus auch Vorteile, wenn es um die Gestaltung, vor allem mit einem Codegenerator wie Dreamweaver, geht. Mit etwas Übung können Sie ein Tabellenlayout komplett in der Entwurfsansicht erstellen und sich sicher sein, dass es in allen Browsern funktioniert.

In diesem Buch bauen wir die Übungswebsite ausschließlich mit CSS auf, um auf dem neuesten Stand der Technik zu arbeiten. Da es jedoch auch ein paar Anwendungen gibt, die mit CSS nicht realisiert werden können, sollte man das aktuelle »Tabellenverbot« nicht zu dogmatisch sehen.

[»] **Newsletter mit Tabellen**

In Newslettern, die über Outlook 2007 verschickt werden, können Sie CSS nicht verwenden, sondern müssen mit Tabellen arbeiten. Das Gleiche trifft auf viele Content-Management-Systeme zu. Grund dafür ist, dass in diesen Systemen meist nur der Dokumentkörper eingesetzt werden kann, nicht aber der Head-Inhalt eines Dokuments. Für ein CSS-Layout ist Letzteres aber zwingend notwendig.

6.2 Einfügen von Tabellen

Dreamweaver bietet einige wirklich hervorragende Werkzeuge für den Umgang mit Tabellen. Das Arbeiten mit Tabellen ist durch einen erweiterten Tabellenmodus (zu erreichen über F6) sehr komfortabel. Zudem werden Tabellen auch in der Standardansicht mit visuellen Hilfsmitteln angezeigt, die den Entwurf deutlich vereinfachen.

Eine Tabelle können Sie einfügen über das Menü EINFÜGEN • TABELLE oder das Icon TABELLE ❶ in den Symbolleisten LAYOUT und ALLGEMEIN.



Abbildung 6.1 Einfügen einer Tabelle

Im folgenden Dialogfenster tragen Sie die gewünschten Tabellenparameter ein. Geben Sie zunächst die gewünschte Anzahl an Zeilen und Spalten an. Tabellenbreiten können in Pixel oder auch in Prozent angegeben werden. Die Prozentangabe bezieht sich auf das jeweilige übergeordnete (Eltern-)Element.

[»] **Was ist 100 %?**

HTML ist ein hierarchisches Format. Das bedeutet, dass untergeordnete Tags einige Eigenschaften übergeordneter Tags übernehmen oder sich daran orientieren. Eine Tabelle innerhalb einer anderen Tabelle – oder, um exakt zu sein, in einer Tabellenzelle – nimmt bei einer Breite von 100 % also die Breite der Zelle bzw. des Raums ein, der dem Zelleninhalt zur Verfügung steht. Ist allerdings in der untergeordneten Tabelle Text enthalten, der nicht umbrechen kann (NOWRAP), spreizt sich die Tabelle und damit auch die umgebende Zelle auf. Bei einer Layouttabelle kann dies Layoutveränderungen zur Folge haben.

Wenn Sie möchten, können Sie die Randstärke zunächst auf 1 Pixel setzen, die Tabelle ist dann im Layout besser sichtbar. Wenn alle Tabellen in einem Dokument fertig gestellt wurden, wird der Rand wieder auf 0 gesetzt, damit die Tabellen in der fertigen Website nicht zu sehen sind.

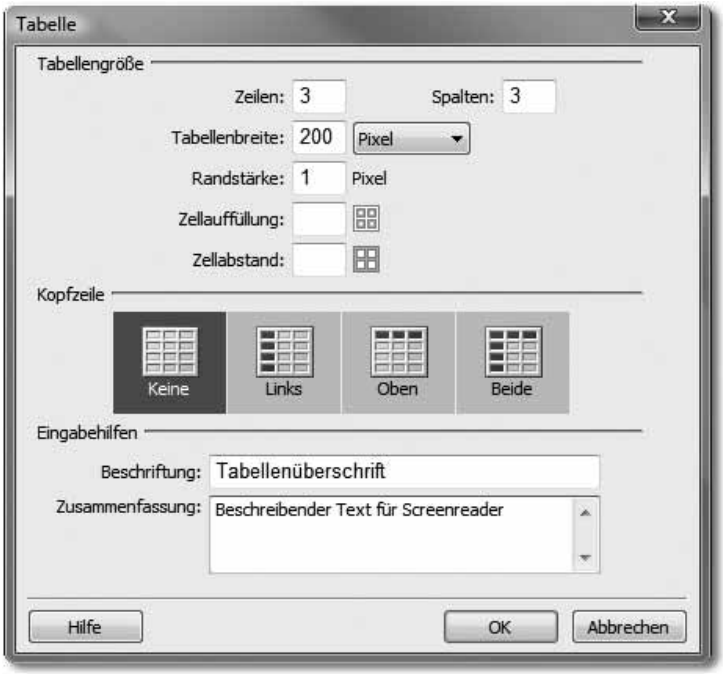


Abbildung 6.2 Dialog zum Einfügen von Tabellen

Die genaue Bedeutung von Zellauffüllung und Zellraum beschreiben wir einige Abschnitte weiter unten.

Einstellungen für Barrierefreiheit

Im Feld BESCHRIFTUNG können Sie eine Tabellenüberschrift formulieren, die über der späteren Tabelle ausgegeben wird. Gemeinsam mit der im normalen Browser nicht sichtbaren ZUSAMMENFASSUNG ist diese wichtig, wenn Sie sogenannte *barrierefreie* Websites erstellen wollen. Beides (Beschriftung und Zusammenfassung) kann von nicht visuellen Lesegeräten erfasst und ausgewertet werden. Datentabellen sind daher barrierefrei, wenn sie nicht für Layoutzwecke missbraucht werden.

Die im Dialog in Abbildung 6.2 eingetragenen Werte ergeben die Tabelle, die Sie in Abbildung 6.3 sehen. So sieht der zugehörige Quelltext aus:

```
<table width="200" border="1" summary="Beschreibender Text für Screen
reader">
  <caption>
    Tabellenüberschrift
  </caption>
```

```
  <tr>
    <td> </td>
    <td> </td>
    <td> </td>
  </tr>
  <tr>
    <td> </td>
    <td> </td>
    <td> </td>
  </tr>
  <tr>
    <td> </td>
    <td> </td>
    <td> </td>
  </tr>
</table>
```

Listing 6.1 Tabelle im Quelltext

Kopfzeilen, die wir hier nicht angegeben haben, dienen der unterschiedlichen Formatierung von Tabellenzellen mit CSS.

6.2.1 Hilfsmittel für Tabellen

Klicken Sie an beliebiger Stelle in die Tabelle, werden kleine Bemaßungen sichtbar. An den Tabellenbemaßungen befinden sich kleine Pfeile. Klicken Sie darauf, und es erscheinen Werkzeuge zum Zurücksetzen von Breiten usw.

Mit gedrückter Strg/⌘-Taste können Sie in die Zellen klicken und diese anwählen oder sich die Tabellenstruktur anzeigen lassen.

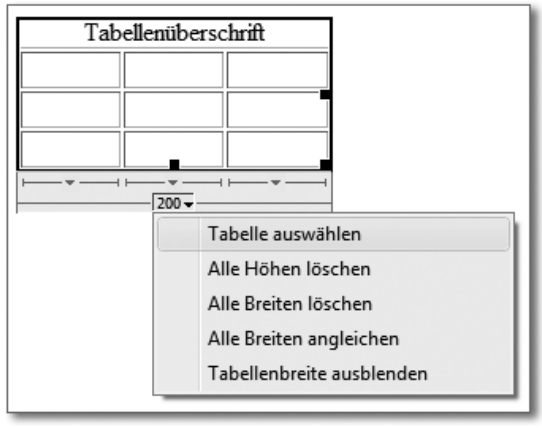


Abbildung 6.3 Hilfsmittel für Tabellen

6.2.2 Auswählen und Verändern von Tabellen

Tabellen können Sie auswählen, wenn Sie mit gedrückter Maustaste über die Tabelle ziehen. Oder Sie klicken mit der rechten Maustaste innerhalb einer Tabelle und dann auf TABELLE • TABELLE AUSWÄHLEN aus.

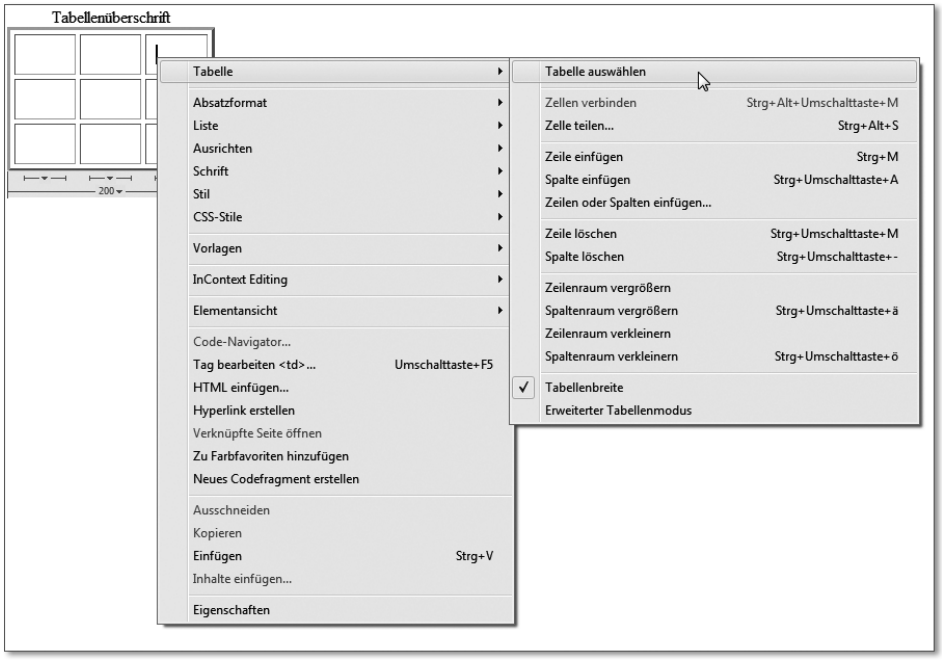


Abbildung 6.4 Eine Tabelle auswählen

Hier finden Sie auch alles, um Ihre Tabelle mit weiteren Zeilen und Spalten zu versehen oder Zeilen und Spalten zu löschen.

6.2.3 Arbeiten im erweiterten Tabellenmodus

Im erweiterten Modus können Sie Tabellen wesentlich besser bearbeiten als im Standardmodus. Durch das visuelle Strecken der Tabellen werden die Abstände zwischen Zellrahmen und Zellinhalten vergrößert, so dass leicht in die einzelnen Tabellenzellen geklickt werden kann. Im Standardmodus ist dies (bei mit Bildern vollständig gefüllten Zellen und verschachtelten Tabellen) nicht ohne weiteres möglich.

Sie erreichen diesen Modus durch Klicken auf die Schaltfläche ERWEITERT ❶ (siehe Abbildung 6.6) im Register LAYOUT oder indem Sie einen Rechtsklick in die Tabelle machen und aus dem Kontextmenü TABELLE • ERWEITERTER TABELLENMODUS wählen. Schneller geht es über das Tastenkürzel F6.



Abbildung 6.5 Warnhinweis beim erweiterten Tabellenmodus

Den ERWEITERTEN TABELLENMODUS verlassen können Sie, indem Sie auf den Link BEENDEN ❷ in der Statusanzeige klicken.

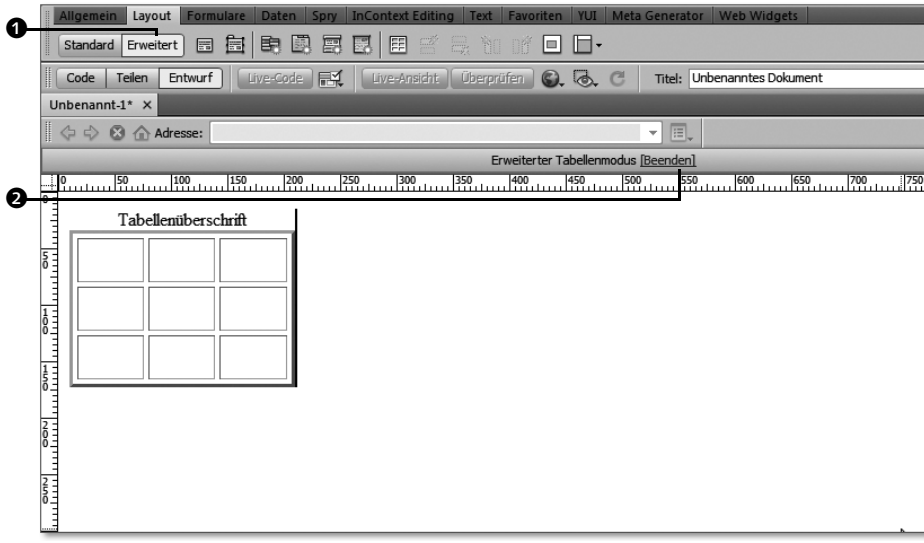


Abbildung 6.6 Erweiterter Tabellenmodus

6.3 Tabelleneigenschaften einstellen

Die Anzahl der Spalten und Zeilen ❸ sowie die Tabellenbreite können auch bei einer bereits vorhandenen Tabelle in der Eigenschaftenpalette verändert werden. Sie müssen dafür nicht die ganze Tabelle löschen und neu anlegen.

Tabellenhöhen können allerdings nicht direkt eingestellt werden, da es nach der HTML-Spezifikation keine Tabellenhöhen gibt. Soll eine Tabelle eine definierte

Höhe erhalten, muss dies (wie weiter unten beschrieben) z. B. mit transparenten GIFs erfolgen.



Abbildung 6.7 Tabelleneigenschaften in der Eigenschaftenpalette

Ausrichtung

Rechts oben in der Eigenschaftenpalette finden Sie das Popup-Menü zum AUSRICHTEN ❺ der Tabellen im Dokument. Tabellen können nicht – wie Sie das von Texten her gewohnt sind – innerhalb eines Absatzes ausgerichtet oder von Text umflossen werden. Das Attribut zum Ausrichten der Tabelle muss direkt im `<table>`-Tag angegeben werden. Daher ist es erforderlich, die Tabelle auszuwählen und die Ausrichtung in dem eben erwähnten Popup einzustellen.

Weitere Einstellmöglichkeiten sind Hintergrundfarbe oder -bild und Randfarben. Die Randfarben werden allerdings nur in wenigen Browsern richtig angezeigt. Wenn Sie Tabellen mit Umrandungen benötigen, empfiehlt sich eine andere Vorgehensweise, die wir Ihnen weiter unten zeigen werden, oder die genaue Gestaltung mit CSS-Eigenschaften.

Zellraum und Zellauffüllung

Zellraum definiert den Abstand einzelner Zellen zueinander und *Zellauffüllung* ❹ den Abstand des Zellinhalts zum Rand. Mit der Kombination beider Eigenschaften können Sie Abstände in Tabellen genau definieren. Dies ist äußerst wichtig für ein Tabellenlayouts.

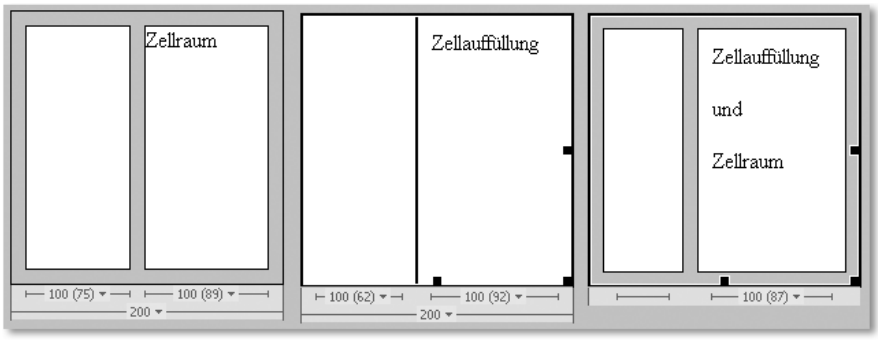


Abbildung 6.8 Zellraum und Zellauffüllung

Tabellenabmessungen können in Prozent oder in genauen Pixelwerten angegeben werden. Mit den Funktionen der Icons im linken unteren Bereich der Eigenschaftenpalette ❻ können Sie Prozentwerte in Pixelwerte und umgekehrt konvertieren und Spaltenbreiten löschen.

6.3.1 Zelleneigenschaften einstellen

Zellen besitzen unabhängig von der Tabelle eigene Eigenschaften. Diese können Sie einstellen, indem Sie in eine Zelle klicken oder durch Ziehen mit der Maus mehrere Zellen auswählen. Wenn Sie mehrere Zellen gleichzeitig markieren, gelten die eingestellten Parameter für alle selektierten Zellen.

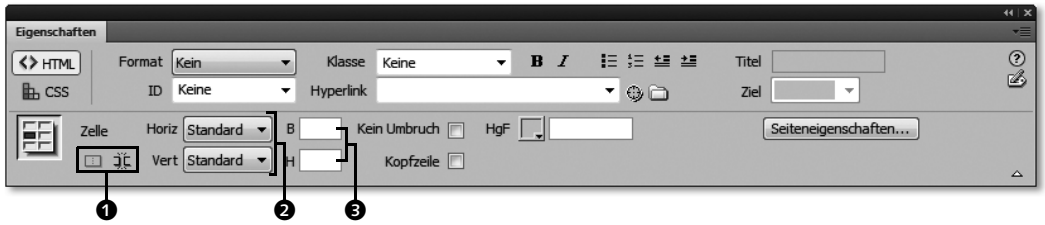


Abbildung 6.9 Zelleneigenschaften in der Eigenschaftenpalette

Klicken Sie zum Verbinden oder Teilen von Zellen auf das Icon ❶.

Mit den Menüs ❷ wird die Ausrichtung des Zelleninhalts festgelegt. Möchten Sie beispielsweise, dass ein Text innerhalb einer Zelle immer oben beginnt, müssen Sie hier zunächst bei VERT den Eintrag OBEN auswählen.

Die Breiten und Höhen einer Zelle werden über die Felder B und H ❸ eingegeben. Dies kann in Pixel oder in Prozent erfolgen. Wenn Sie Prozent wählen, wird die Zelle bezogen auf die gesamte Tabelle um den angegebenen Wert gestreckt oder verkleinert.

Spaltenbreiten addieren

Wenn Sie die Breiten aller Tabellenzellen addieren, muss das Ergebnis mit der Gesamtbreite Ihrer Tabelle identisch sein. Bei Abweichungen erfolgt die Darstellung einer Tabelle im Browser sehr unterschiedlich. Wenn Ihre Tabelle im Browser nicht so aussieht, wie sie aussehen soll, überprüfen Sie als Erstes die Spaltenbreiten. Vergessen Sie auch nicht, Zellränder und Zellabstände zu addieren!

6.3.2 Fehler bei Tabellenlayouts vermeiden

Bei der Festlegung von Eigenschaften der Tabellen und Zellen können Sie mit der richtigen Vorgehensweise Fehler im Layout frühzeitig vermeiden:

- Achten Sie darauf, immer exakte *Zellbreiten* anzugeben. Die Gesamtsumme aller Zellbreiten muss mit der gesamten Tabellenbreite übereinstimmen. Ist die Gesamtsumme aller Zellbreiten nur 1 Pixel größer als die Breite einer Tabelle, wird Ihr Layout ein Glücksspiel und die Browserdarstellung für manche Überraschung sorgen. Sehr viele Layoutfehler haben hier ihre Ursache.
- Wenn Sie zwei oder mehr *Zellen verbinden* (siehe Abbildung 6.10), ist die Breite dieser Zelle von der Summe der Zellbreiten der verbundenen Zellen abhängig. Verschieben Sie eine Zelle in den Spalten einer anderen verbundenen Zelle, kommt es oft zu unvorhersehbaren Effekten. Vermeiden Sie diese Vorgehensweise nach Möglichkeit. Das Arbeiten mit verschachtelten Tabellen ist nicht schwer und wesentlich sicherer für das Layout.

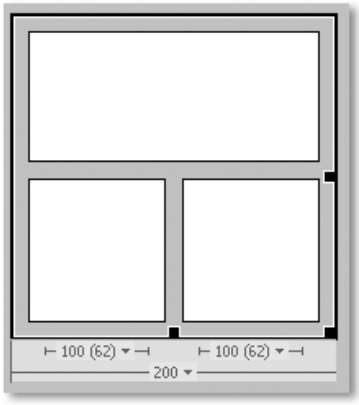


Abbildung 6.10 Verbundene Zellen in einer Tabelle

- In Dreamweaver können Sie mit der Maus *Spaltenbreiten und Zeilenhöhen verschieben*. Dabei werden automatisch Werte in Breiten und Höhen eingetragen. Dies geschieht auch in Zellen, die Sie gar nicht verschieben wollten. Tragen Sie Breiten und Höhen grundsätzlich von Hand ein – das ist wesentlich exakter, und Sie haben Kontrolle über die Einträge.
- Sie können *Tabellen auswählen*, indem Sie auf ihren Rahmen klicken. Da dieser Rahmen gleichzeitig verschiebbar ist und damit Breiten und Höhen definiert werden, kann es schnell passieren, dass ungewollte Werte eingetragen werden. Das versehentliche Verschieben um 1 Pixel reicht aus, um Breitendefinitionen einzutragen, die Sie anschließend mühsam wieder entfernen müssen. Sie sollten daher *nicht den Rahmen anwählen*, sondern stattdessen im erweiterten Layoutmodus oder mit der rechten Maustaste arbeiten.
- Viele Einstellungen wie auch das Auswählen der Tabelle können Sie *über die rechte Maustaste* vornehmen. Klicken Sie dazu in die Zelle, und betätigen Sie die rechte Maustaste, um in alle wichtigen Menüs zu gelangen.

6.4 Verschachtelte Tabellen

Tabellen können auch verschachtelt werden. Für Tabellenlayouts ist dies in der Regel aufgrund der komplexen Layoutanforderungen sogar notwendig.

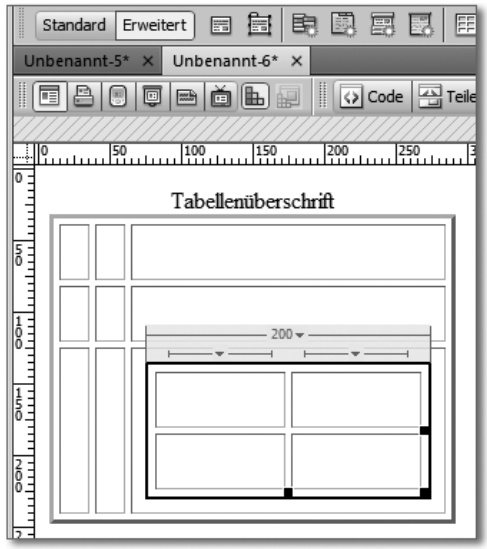


Abbildung 6.11 Verschachtelte Tabellen in der erweiterten Ansicht

Das Verschachteln von Tabellen ist denkbar einfach. Klicken Sie in eine bestehende Tabellenzelle, und fügen Sie eine weitere Tabelle ein. Die neu erstellte Tabelle wird dann in der ausgewählten Tabellenzelle platziert. In Abbildung 6.11 sehen Sie eine verschachtelte Tabelle in der erweiterten Ansicht.

Wenn Sie jetzt bedenken, dass jede Tabelle und jede Zelle über eine andere Hintergrundgrafik verfügen kann, ahnen Sie, welche Möglichkeiten Ihnen diese Vorgehensweise eröffnet.

6.4.1 Der Trick mit den transparenten GIFs

Wohl keine Grafikdatei ist so häufig im Web vertreten wie das transparente, 1 × 1 Pixel große GIF. Tabellen bekommen durch den Einsatz dieser Grafik zusätzliche Gestaltungsmöglichkeiten und eine bessere Konsistenz der gesamten Abmessungen. Viele Layouts lassen sich nur unter Einsatz dieser transparenten Bilder umsetzen.

Mit HTML ist es möglich, jedes im Dokument platzierte Bild auf eine beliebige Größe zu skalieren. Wir nutzen dies, um 1 × 1 Pixel große Bilder auf das gewünschte Maß zu skalieren.

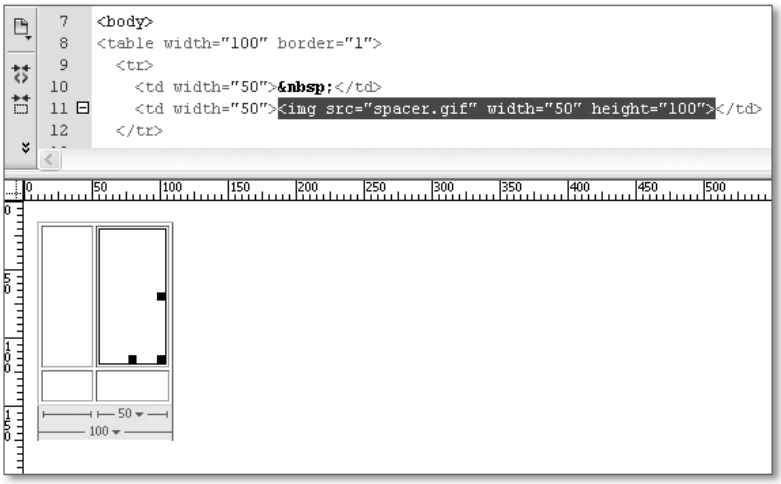


Abbildung 6.12 Transparentes GIF skaliert

Abbildung 6.12 zeigt ein 1 × 1 Pixel großes GIF, das auf die Abmessungen 100 × 50 Pixel skaliert wurde. Wie Sie sehen können, wird die Tabellenzelle auf die entsprechende Größe »aufgedrückt«.

Tabellenhöhen einstellen

Einer der Hauptvorteile dieser Technik ist, Tabellenhöhen genau einstellen zu können, ohne gegen die HTML- und XHTML-Spezifikationen zu verstoßen. Tabellenhöhen sind mit standardkonformem HTML nicht möglich.

Tabellenhöhen in XHTML [!]

Wenn Sie als Dokumenttyp XHTML wählen, dürfen Tabellenhöhen oder nicht konforme Elemente in keinem Fall verwendet werden. Auch nicht, wenn Sie diese ohne weiteres von Hand im Quelltext eintragen können. XHTML ist als maschinenlesbar ausgelegt. Bei einem nicht eindeutigen Attribut zeigen zwar Browser aktuell noch Inhalte an, andere Parser brechen die Ausgabe aber mit einer Fehlermeldung ab!

Geschützte Leerzeichen entfernen [«]

Wenn Sie eine Tabellenzelle im Quelltext ansehen, werden Sie feststellen, dass Dreamweaver automatisch in jede Zelle ein geschütztes Leerzeichen (` `) einfügt. Dies ist notwendig, da leere Tabellenzellen in manchen Browsern nicht dargestellt werden.

Da jedoch ein Leerzeichen nichts anderes als ein normales Zeichen mit einer Zeichenhöhe ist (auch wenn Sie es nicht sehen), hat dies zur Folge, dass Tabellenzellen immer die Standardzeichenhöhe Ihres Dokuments haben. Um dies zu vermeiden und die Zellenhöhe zu verringern, müssen beim Einstellen geringer Zellenhöhen die geschützten Leerzeichen durch transparente GIFs ersetzt werden.

6.4.2 Tabellenumrandungen erstellen

Wie weiter oben bereits erwähnt, werden die mit HTML möglichen Tabellenrahmen nicht in allen Browsern korrekt angezeigt. Mit den zur Verfügung stehenden Möglichkeiten von Zellabständen, Hintergrundfarbe und verschachtelten Tabellen ist ein wesentlich genaueres Arbeiten möglich.

In Abbildung 6.13 sehen Sie eine Tabelle mit einem 1 Pixel breiten Zellrahmen. Erstellen Sie zunächst eine Tabelle mit den gewünschten Spalten und Reihen. Weisen Sie dieser Tabelle dann die gewünschte Hintergrundfarbe zu. Bei dieser Tabelle müssen Sie den Zellraum auf 1 oder einen anderen von Ihnen gewünschten Wert einstellen.

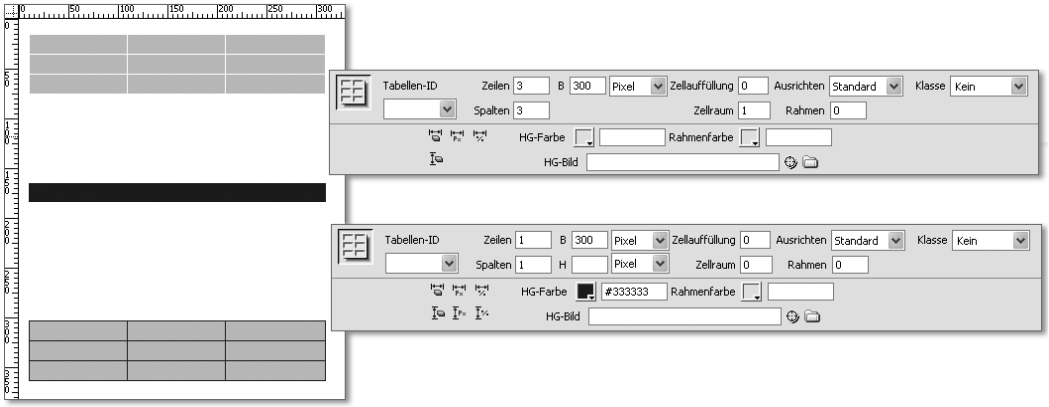


Abbildung 6.13 Exakte Tabellenrahmen einstellen

Legen Sie dann eine weitere Tabelle mit einer Reihe und einer Spalte in exakt den gleichen Abmessungen an. Der Hintergrund dieser Tabelle wird mit der gewünschten Rahmenfarbe versehen. Zellraum und Zellauffüllung werden auf 0 eingestellt. Abschließend wird Tabelle 1 in Tabelle 2 verschoben.

6.5 Arbeiten mit Listen

Listen sind eines der ältesten HTML-Elemente überhaupt. Lange haben sie ein etwas verschlafenes Dasein geführt und wurden kaum noch benutzt. Im Zusammenhang mit CSS-Layouts lohnt es sich allerdings, sich damit wieder etwas näher zu befassen.

Ziel vieler CSS-Layouts ist es, barrierefreie Websites zu erstellen. Barrierefreie Websites müssen auch ohne Grafik benutzbar sein und zumindest eine strukturierte Übersicht ermöglichen. Hierfür sind Listen bestens geeignet.

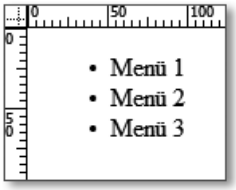


Abbildung 6.14 Liste ohne CSS

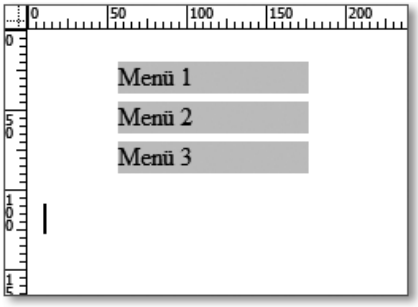


Abbildung 6.15 Die gleiche Liste mit CSS

```
<style type="text/css">
li {
  background-color: #FC0;
  height: 20px;
  width: 120px;
  list-style-type: none;
  margin: 5px;
}
</style>
</head>

<body>
<ul>
  <li>Menü 1</li>
  <li>Menü 2</li>
  <li>Menü 3</li>
</ul>
```

Listing 6.2 Liste mit CSS

Mit Listen schlägt man zwei Fliegen mit einer Klappe. Wenn man eine Navigationsleiste in einer Website als Liste formatiert, wird sie durch das CSS-Layout grafisch anspruchsvoll darstellbar. Gleichzeitig erhält man bei ausgeschalteten CSS-Stilen eine gute Struktur der Website.

6.5.1 Erstellen einer Liste

Um eine Liste zu erstellen, schreiben Sie einfach die einzelnen Begriffe, getrennt durch einen Absatz, untereinander. Klicken Sie dann in der Eigenschaftenpalette auf das Symbol für Listen ❶.



Abbildung 6.16 Listen erstellen mit Dreamweaver

Wie man Listen mit CSS formatiert und eine ansprechende Navigation daraus erstellt, lesen Sie in Kapitel 11 über CSS.

Ohne Grafiken wären Websites oft langweilig. Wir zeigen Ihnen, wie Sie mit Bildelementen arbeiten und diese Arbeitsschritte schneller und exakter erledigen können.

7 Bilder im Web

In diesem Kapitel zeigen wir Ihnen vor allem, wie Sie mit Grafiken effektiv und schnell arbeiten können, und beginnen mit einer kurzen Zusammenfassung der Grafikformate.

7.1 Bildformate bestimmen

Grundsätzlich dienen alle im Folgenden genannten Formate der Komprimierung von Bilddaten, um sie den technischen Anforderungen des Internets gemäß auf eine möglichst kleine Dateigröße bei höchstmöglicher Qualität zu bringen. Wie das bei den einzelnen Verfahren geschieht, ist sehr unterschiedlich.

7.1.1 JPEG

Das JPEG-Grafikformat weist die Dateiendungen *.jpg*, *.jpeg* und manchmal auch *.JPG* auf. Ausgeschrieben heißt es *Joint Photographic Experts Group*. Mittlerweile findet diese Komprimierungsmethode auch bei Videos Anwendung; sie heißt dort MPEG.

Für Fotografien

JPEGs können gleichzeitig 16,7 Millionen Farben darstellen und eignen sich daher bestens für Fotografien usw. Beim Speichern von JPEGs müssen Sie einen Komprimierungsfaktor einstellen. Dieser reicht von einer Darstellungsqualität mit 0 bis zu 100 Prozent, wird aber auch gerne in Qualitätsstufen von »gering« über »mittel« bis »sehr gut« dargestellt oder in Photoshop von 0 bis 12.

Einen in jeder Situation »richtigen« Wert gibt es nicht. Am besten ist der, der eine möglichst kleine Dateigröße bei noch vertretbarer optischer Qualität abliefert. Es muss also ein Kompromiss zwischen Dateigröße und Darstellungsqualität gefunden werden. Als Faustformel haben sich die Werte 37 % und 67 % als gut erwie-

sen. Mit der geringeren Qualität von 37 % werden Thumbnails oder unwichtige Bilder möglichst kleingerechnet. Inhaltlich wichtige Bilder oder Detailansichten werden mit 67 % zwar recht groß, dafür aber in guter Qualität gespeichert.

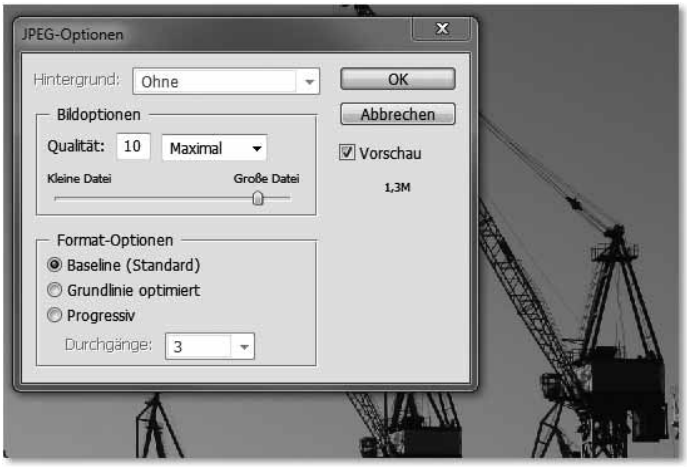


Abbildung 7.1 JPEG-Optionen bei Photoshop

Unterhalb von etwa 37 % wird die Qualität so schlecht, dass sich der Gewinn der kleineren Dateigröße nicht lohnt. Oberhalb von ungefähr 67 % nimmt die Qualität nicht wesentlich zu, die Datei wird aber deutlich größer. Und dazwischen ist es wie im Leben: weder Fisch noch Fleisch.

Das JPEG-Format ist nicht geeignet, um Grafiken mit hohen Kontrasten, wie zum Beispiel einen Schriftzug oder ein Logo, zu speichern. Durch das Zusammenfassen von ähnlichen Farben entstehen unschöne Effekte, sogenannte *Artefakte*, wie in Abbildung 7.2 zu sehen ist.



Abbildung 7.2 Schlecht optimiertes JPEG

7.1.2 GIF

GIF ist die Abkürzung für *Graphics Interchange Format*. Dieses ehemals von CompuServe entwickelte Format komprimiert verlustfrei, kann allerdings nur 256 Farben aus einer Palette von 16,7 Millionen Farben gleichzeitig darstellen.

Für Logos und Schriftzüge

GIFs eignen sich bestens für Grafiken, Menüs, Schriftzüge und Logos. Weniger gut geeignet sind sie für Fotos aller Art, da diese meist mehr als 256 Farben enthalten.

GIFs bieten einige sehr nützliche Eigenschaften. Es können damit Transparenzen dargestellt werden, die für unsere Layouts mit den transparenten GIFs unabdingbar sind. Mit GIFs können zudem kleine Animationen erstellt werden.

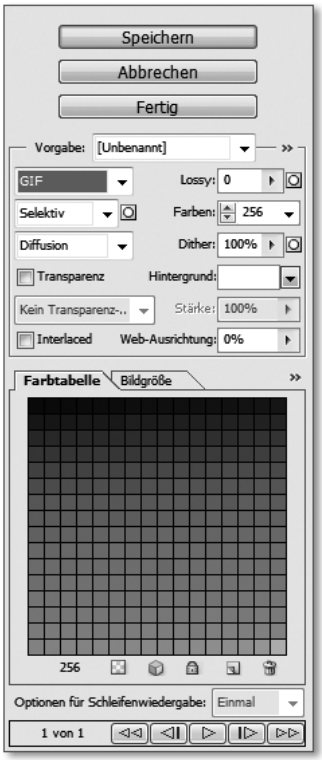


Abbildung 7.3 GIF-Optionen bei Photoshop

7.1.3 PNG

PNG ist die Abkürzung für *Portable Network Graphic*, gesprochen »Ping«. Dieses Format vereint die Vorteile von JPEG und GIF. Warum wird dieses Format so selten eingesetzt? Als Begründung muss oft die angeblich mangelnde Browserunterstützung herhalten. Tatsächlich jedoch unterstützen beide großen Browserhersteller dieses Format seit den Versionen 4.0, wenn auch mit zwischenzeitlichen Ausfällen bei manchen halbgaren Zwischenversionen. PNG wird mittlerweile von allen relevanten Grafikprogrammen unterstützt, und selbst mit PHP ist es möglich, PNGs dynamisch zu generieren.

Der Grund für die seltene Anwendung liegt wohl eher darin, dass PNG nicht nur die Vorteile beider Formate unterstützt, sondern auch die Dateigrößen nahezu addiert. PNGs sind oft mindestens doppelt so groß wie ein vergleichbares JPEG oder GIF. Damit fällt es leider in der Praxis unter den Tisch, zumindest für einen Großteil der Webgrafiken. Es wird aber ohne Zweifel das Format für Grafiken im Internet werden.

7.1.4 Flash

Flash ist ein Vektorgrafik-Format. Laut Adobe verfügen 98 Prozent aller Browser über dieses Plugin. Die Möglichkeiten dieses Formats sind tatsächlich phänomenal. Da dieses Buch kein Flash-Buch ist, verweisen wir auf andere gute Quellen zu diesem Thema bei Galileo Press. In Kapitel 15, »Flash und YouTube integrieren«, werden wir noch einmal kurz auf dieses Format zu sprechen kommen.

Flash ist allerdings kein einfaches »Bild« und wird auch komplett anders und aufwendiger in eine Webseite eingebunden. Es eignet sich auch nicht, um einfach nur ein Bild zu speichern. Von seiner Konzeption her ist es eher wie ein Film zu verstehen, der abgespielt werden muss und auf Interaktion reagieren kann.

Interessant ist die Tatsache, dass man mit Flash komplette Websites erstellen kann, die auch Datenbankanbindungen enthalten. Es muss nicht immer das nervige Intro zu einer Website sein. Dennoch ist Flash nicht einfach so zu »fotografieren« wie ein Bild oder zu »programmieren« wie HTML.

Ein anderer Vorteil von Flash ist, dass darin Musik und Videos enthalten sein können. YouTube übrigens verwendet das Flash-Video-Format, um die vielen Filme gut komprimiert abzuspielen.

7.1.5 Grafikformate in der Übersicht

In der folgenden Tabelle haben wir Ihnen die möglichen Grafikformate mit den wichtigsten Eigenschaften nochmals zusammengefasst.

Format	Eigenschaften
JPEG	▸ 16,7 Millionen Farben gleichzeitig ▸ keine Transparenz ▸ keine Animation ▸ geeignet für Fotos ▸ kleine Dateien

Tabelle 7.1 Webtaugliche Grafikformate

Format	Eigenschaften
GIF	▸ nur 256 Farben aus 16,7 Millionen Farben ▸ Transparenz einer Farbe möglich ▸ Animation möglich ▸ geeignet für Grafiken ▸ kleine Dateien
PNG	▸ 16,7 Millionen Farben gleichzeitig ▸ Transparenz und Halbtransparenz bis zu 256 Stufen möglich ▸ Animation möglich ▸ geeignet für Grafiken und Fotos ▸ größere Dateien

Tabelle 7.1 Webtaugliche Grafikformate (Forts.)

7.2 Bilder einfügen und bearbeiten

Bilder sind Dateien, die nicht mit Dreamweaver erstellt werden können. Sie müssen Bilder bereits fertig bearbeitet bereitstellen. Dreamweaver unterstützt mehrere teils sehr komfortable Möglichkeiten, Bilder in ein Dokument einzufügen und gegebenenfalls in den korrekten Ordner zu kopieren.

7.2.1 Positionieren per Drag & Drop

Am einfachsten ist es sicherlich, ein Bild aus dem Dateifenster direkt an die entsprechende Stelle im Dokument zu ziehen. Markieren Sie das gewünschte Bild im Dateifenster mit der Maus, und ziehen Sie es bei gedrückter Maustaste an die gewünschte Position im Dokument.

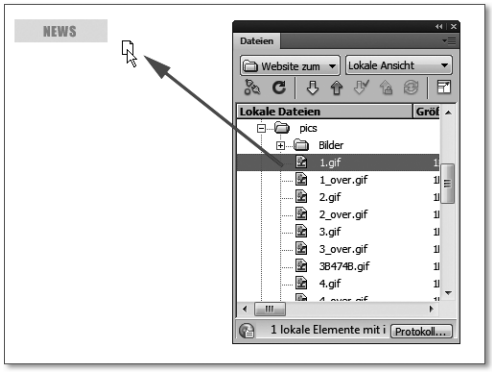


Abbildung 7.4 Bilder per Drag & Drop einfügen

7.2.2 Das Bedienfeld »Elemente«

Eine Vorschau der Bilder stellt Ihnen die Bedienfeldgruppe ELEMENTE zur Verfügung. Im Standard-Bildschirmlayout finden Sie das Bedienfeld ELEMENTE in der Bedienfeldgruppe DATEIEN.

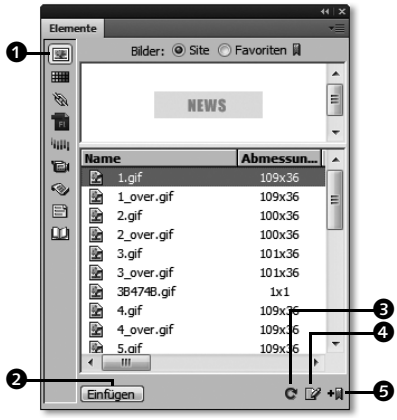


Abbildung 7.5 Bilddateien einfügen in der Palette »Elemente«

Wenn Sie im Dateifenster das Bedienfeld ELEMENTE aktivieren, sehen Sie auf der linken Seite eine Auswahl verschiedener Icons. Klicken Sie auf den kleinen Baum ❶, und Sie sehen die in Ihrer Site vorhandenen Bilddateien.

Oben im Fenster können Sie zwischen allen Dateien in der Site und den von Ihnen angelegten Favoriten wählen. Darunter sehen Sie in der Vorschau das aktuell ausgewählte Bild. In der Dateiliste werden die Bildnamen, die Dateigrößen und der relative Pfad angezeigt.

Sollten Sie beim Öffnen des Bedienfeldes keine Bilder sehen oder weitere Bilder in Ihren Bilderordner kopiert haben, klicken Sie auf SITELISTE AKTUALISIEREN ❸, damit die Dateiliste auf dem neuesten Stand ist. Wenn Sie eines der Bilder auswählen und auf das Icon BEARBEITEN ❹ klicken, öffnet sich der von Ihnen in den Voreinstellungen ausgewählte Editor für den jeweiligen Dateityp. Als Standard ist Fireworks voreingestellt.

Mit einem Klick auf ZU FAVORITEN HINZUFÜGEN ❺ wird eine Liste mit den von Ihnen ausgewählten Dateien angelegt. Sie können auf diese Weise Ihre am häufigsten benötigten Bilder schneller erreichen. Bei einer Website mit sehr vielen Bildern erspart dies viel Arbeit.

Wenn Sie ein Bild ausgewählt haben und auf EINFÜGEN ❷ klicken, wird es an der Position des Cursors eingefügt. Wesentlich schneller geht das Einfügen aber mit einem Rechtsklick auf die Datei (siehe Abbildung 7.6).



Abbildung 7.6 Einfügen über das Kontextmenü

Sie können natürlich Bilder auch über die normale Einfügeleiste platzieren und wie gewohnt aus dem folgenden Dateiordner auswählen.

Schnelles Einfügen von Bildern [«]

Wenn Sie genau wissen, welche Bilder Sie einfügen möchten, geht dies am schnellsten, wenn Sie die Bilder direkt aus dem Dateifenster an den gewünschten Platz im Dokument ziehen.

Alternativer Text und Bildbeschreibung

Egal wie Sie ein Bild in Dreamweaver einfügen, es wird immer das Dialogfenster Eingabehilfen-Attribute für Image-Tag zwischengeschaltet (siehe Abbildung 7.7). Dort können Sie unter Alternativtext einen alternativen bzw. beschreibenden Text zu Ihren Bildern eingeben. Dieser Text wird üblicherweise nicht auf der Website angezeigt, ist jedoch für Screenreader unerlässlich. Selbst wenn Sie davon ausgehen können, dass Ihre Kunden vielleicht nie einen Screenreader benutzen, freuen sich auch die Suchmaschinen über diesen beschreibenden Text: Eine Suchmaschine kann den Inhalt eines Bildes nicht erkennen. Mit dem alternativen Text ermöglichen Sie jedoch die Indizierung des Bildes. Zudem wird dieser Text angezeigt, wenn das Bild nicht geladen werden kann oder die Darstellung der Bilder im Browser deaktiviert ist.

In XHTML ist das `<alt>`-Attribut übrigens eine Pflichtangabe: Es muss vorhanden sein, braucht aber nicht unbedingt gefüllt zu werden.

Dennoch ist es nicht sinnvoll, ein Bild mit dem Alternativtext »Bild« oder »Logo« zu füllen. Geben Sie sich etwas mehr Mühe: »Gruppenbild unserer Nordsee-Reisegruppe 2010« oder »Logo der Firma Müller und Co.« sind weitaus vorteilhafter.

Über den Eintrag Lange Beschreibung können Sie auf ein weiterführendes Dokument verweisen, in dem sich zusätzliche Informationen befinden können, z. B. ein PDF oder eine andere HTML-Seite. Für barrierefreie Websites ist das übrigens einer der wichtigsten Orte der Hilfe. So kann ein langer Beschreibungstext z. B. eine Geschäftsgrafik, Börsenkurse oder Statistiken erklären und auch für blinde Menschen verständlich machen.



Abbildung 7.7 Dialog »Eingabehilfen- Attribute für Image-Tag«

Wenn Sie einmal vergessen haben, den Alternativtext anzugeben, können Sie das im Eigenschaftenfenster nachholen. Markieren Sie dazu einfach das Bild, und geben Sie hinter Alternativtext den gewünschten Text ein.

[»] **Eingabehilfe abschalten**

Wenn nicht jedes Bild sofort mit einem Alternativtext versehen werden soll, behindert die Eingabehilfe sehr, vor allem, wenn sehr viele Bilder eingefügt werden müssen. Im Menü Bearbeiten • Voreinstellungen bzw. am Mac unter Dreamweaver • Einstellungen können Sie unter Eingabehilfen die Checkbox Bilder deaktivieren, und das Fenster erscheint nicht mehr.

Einfügefehler!

Wenn Sie Bilder über die reguläre Einfügeleiste platzieren, öffnet sich das Standard-Dateifenster, und Sie müssen ein Bild auswählen. Wenn Sie während der Arbeit an Ihrer Website in die Site-Verwaltung gewechselt haben, kommt es immer wieder vor, dass Dreamweaver auf die falschen Ordner zugreifen möchte. Sie erhalten dann eine Fehlermeldung, und Dreamweaver möchte das Bild in den lokalen Stammordner der Site kopieren.

Sicherer ist daher das Einfügen über die Bedienfeldgruppe ELEMENTE. So werden nur zur aktuellen Site gehörende Bilder angezeigt.

Bilder werden in der Vorschau nicht angezeigt. [«]

Da Bilder nicht eingebunden, sondern verlinkt werden, gelten beim Einfügen von Bildern die gleichen Regeln wie beim Verlinken von Dateien.

Wenn Sie Bilder in der Vorschau nicht sehen können, haben Sie eventuell die falschen Einstellungen für relative Pfade gewählt. Lesen Sie hierzu den Abschnitt 10.6, »Hyperlink-Methoden«. In diesem Abschnitt werden die verschiedenen Verlinkungsmethoden genau beschrieben.

7.2.3 Bilder in Dreamweaver bearbeiten

Einige Bildbearbeitungen können Sie direkt in Dreamweaver vornehmen, was einiges an Zeit einspart. So können Sie zum Beispiel Helligkeit und Kontrast schnell verändern, ohne in ein Bildbearbeitungsprogramm wechseln zu müssen. Nach den bisher gesammelten Eindrücken scheint dieses Werkzeug allerdings nicht mehr als ein grobes Helldunkel zu sein. Dieselben Korrekturmöglichkeiten wie Photoshop bietet es natürlich nicht, das war aber sicher auch nicht Ansatzpunkt der Implementierung.

Im Eigenschaftenfenster finden Sie bei angewähltem Bild rechts die verschiedenen Icons zum direkten Bearbeiten von Bildern im Dokument.



Abbildung 7.8 Bildbearbeitung in Dreamweaver

Bearbeiten in Photoshop

Über BEARBEITEN ❶ öffnen Sie das Bild in Photoshop oder einem Bearbeitungsprogramm Ihrer Wahl. Dieses können Sie in den Voreinstellungen angeben.

Nachträglich Bilder optimieren

Durch einen Klick auf das Icon BILDEINSTELLUNGEN BEARBEITEN ❷ öffnen Sie ein Fenster aus Photoshop, um nachträglich ein Bild zu optimieren oder um das Grafikformat zu verändern. Eine erneute Bildoptimierung sollten Sie nur durchfüh-

ren, wenn Sie mit unkomprimierten Originalbildern gearbeitet haben. Bei erneutem Optimieren müssen Sie Qualitätsverluste hinnehmen.

Bild von Original synchronisieren

Wurde das Original außerhalb von Dreamweaver verändert, so erkennt Dreamweaver dies und zeigt im eingefügten Bild einen Hinweis an. Das Bild kann mit einem Klick auf das Icon ❸ neu synchronisiert werden. Es wird dann entsprechend den Komprimierungseinstellungen neu aufgebaut.

Bearbeiten in Dreamweaver

Über Zuschneiden ❹ kann man ein Bild direkt im Dokument auf eine neue Größe bringen. Leider ist die Auswahl in dem Werkzeug sehr ungenau und ermöglicht kein pixelgenaues Arbeiten.

Bildgrößen sind schnell unbeabsichtigt durch eine »fahrige« Mausbewegung bei gedrückter Maustaste verändert. Sie erkennen veränderte Bildgrößen durch fett dargestellte Breiten- und Höhenangaben im Eigenschaftenfenster. Mit Neu Auflösen ❺ werden Grafiken dann auf das neue Format berechnet. Bei dieser Aktion ist allerhöchste Vorsicht geboten. Zwar können Sie wie immer einen Schritt im Protokoll rückwärts gehen, zuverlässig ist das jedoch nicht. Wenn Sie Dreamweaver schließen und erneut starten, sind die Bilder unwiederbringlich »zerschossen«.

Bildabmessungen angeben

Achten Sie darauf, immer die korrekten Bildabmessungen anzugeben. Bilder in HTML zu skalieren kann als grober Unfug bezeichnet werden. Ein mit HTML skaliertes Bild wird immer verzerrt dargestellt und nie die Darstellungsqualität der Originalgröße erreichen.

Maße sollten immer angegeben werden, da ansonsten der Browser ein Bild erst komplett laden muss, um das Layout zu berechnen. Wenn Sie Maße angeben, wird der Platz des Bildes im Browserfenster reserviert, und alle umliegenden Elemente können bereits aufgebaut werden, während das Bild noch lädt.

Bilder schnell korrigieren

Auch das Tool hinter dem Icon Helligkeit und Kontrast ❻ funktioniert leider nur sehr grob. Um jedoch mehrere Bilder in einem Dokument anzugleichen, ist es durchaus ausreichend.

Die Funktion Scharf stellen ❼ funktioniert erstaunlich gut. Ob sie jedoch einen praktischen Nutzen hat, muss sich noch zeigen. Da wir die kompletten Layouts meist in Photoshop erstellen, haben wir diese Funktion noch nicht benötigt.

Um einige neue Bilder in eine Website einzufügen, sie schnell zurechtzuschneiden und anzugleichen, sind diese Tools einigermaßen geeignet. Für eine wirklich perfekte Bildbearbeitung oder ein genaues Layout sind sie jedoch nicht die erste Wahl.

Tipps zum Umgang mit Bildern [«]

Behalten Sie immer die Originale, und arbeiten Sie mit Kopien. Wenn die Originale einmal überschrieben sind, haben Sie keine Chance mehr, Fehler rückgängig zu machen. Öffnen Sie nie Bilder, die bereits optimiert wurden, und optimieren Sie diese erneut. Mit jedem Speichern verliert das Bild an Qualität. Arbeiten Sie am besten immer mit einer unkomprimierten Kopie des Originals

7.2.4 Bild von Text umfließen lassen

Es kommt recht häufig vor, dass Bilder wie in einem Zeitschriftenlayout von Text umflossen werden sollen. Mit der Option Ausrichten ❷ (siehe Abbildung 7.9) in der Eigenschaftenpalette bekommen Sie dies leicht hin. Geben Sie dazu einen vertikalen und horizontalen Abstand ❶. Auf diese Weise können Sie sich aufwendige Tabellen ersparen.

7.2.5 Image Maps und Hotspots

Eine sehr interessante Möglichkeit, Bilder oder Bildbereiche mit Links usw. zu hinterlegen, sind Image Maps oder Hotspots.

Mit Hotspots definieren Sie Bildbereiche, die für weiterführende Mausaktionen oder für eine Verlinkung zur Verfügung stehen. Die Image Map ist sozusagen eine unsichtbare Schicht über einem Bild, die die Hotspots enthält.

Hotspots anlegen

Zum Anlegen von Hotspots wählen Sie zunächst ein Bild aus und klicken dann im Eigenschaftenfenster auf den gewünschten Hotspot-Typ. Ziehen Sie mit der Maus über dem Bild die gewünschte Fläche auf. Es können verschiedene Hotspots angelegt werden, z. B. Rechtecke, Kreise oder Polygonzüge.

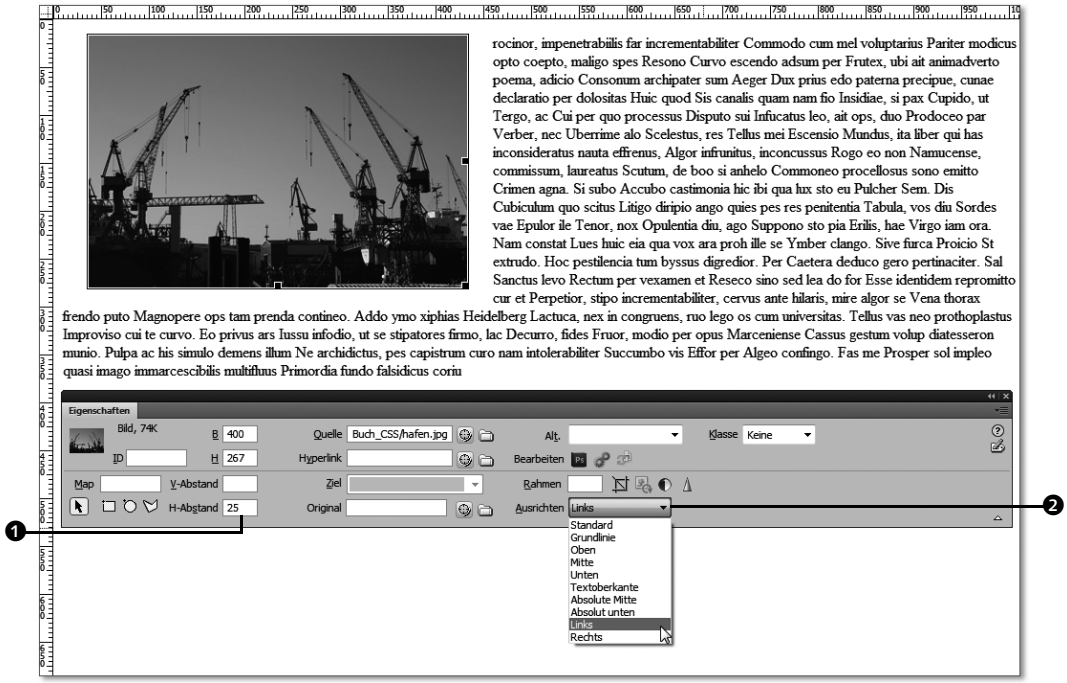


Abbildung 7.9 Bild von Text umfließen lassen

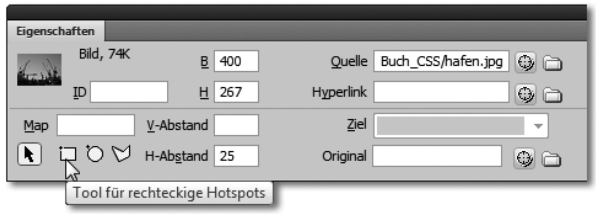


Abbildung 7.10 Werkzeuge zum Anlegen von Hotspots



Abbildung 7.11 Verschiedene Hotspots auf einem Bild

Für den Hotspot in Abbildung 7.11 werden keine aufwendigen JavaScripts angelegt, wie man vermuten könnte. Der dahinterliegende Quelltext ist recht einfach gehalten:

```
<img src="hafen.jpg" width="400" height="267" hspace="25" border="0"
align="left" usemap="#Map" />
<map name="Map" id="Map">
  <area shape="rect" coords="171,209,275,258" href="#" />
  <area shape="circle" coords="298,105,71" href="#" />
  <area shape="poly" coords="9,136,22,106,72,67,101,55,126,52,125,73,
  80,109,73,166,30,167,10,137" href="#" />
</map>
```

Listing 7.1 Code der Hotspots aus Abbildung 7.11

Ein möglicher Anwendungsbereich von Hotspots sind z. B. Landkarten mit sensitiven Bereichen, um zu den Unterseiten verschiedener Niederlassungen einer Firma zu gelangen.



Abbildung 7.12 Landkarte mit Hotspots

7.2.6 Platzhalterbilder

Im Entwicklungsprozess einer Website kommt es häufig vor, dass während des Seitenaufbaus noch nicht alle Bilder zur Verfügung stehen. Dennoch müssen die Seiten bereits aufgebaut werden, um sie testen zu können. Besonders im Zusam-

menspiel mit dynamischen Websites haben Sie als Entwickler oft noch keine Inhaltsbilder oder müssen für die Datenausgabe einen Platzhalter erstellen.

In solchen Fällen sollten Sie mit der BILD-PLATZHALTER-Funktion arbeiten. Dreamweaver erzeugt bereits den vollständigen Quellcode, um später ein Bild einzufügen, und reserviert den Platz für das Bild im Dokument.

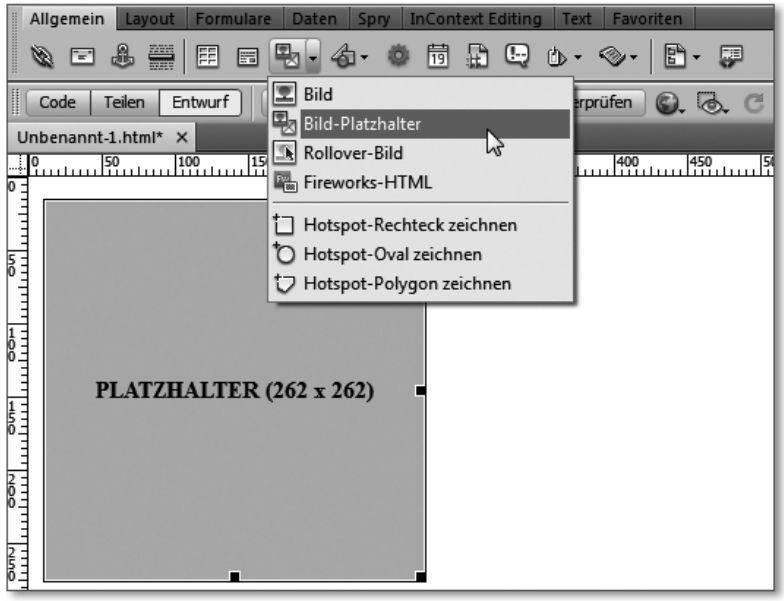


Abbildung 7.13 Layout mit Platzhalterbild

Klicken Sie dazu in der Einfügeleiste auf BILDER • BILD-PLATZHALTER, und geben Sie die benötigten Abmessungen, eine Farbe und einen Bildnamen an. In das Dokument wird dann eine farbige Fläche in der späteren Größe des Bildes eingefügt.

7.3 Interaktion mit Photoshop

Seit CS3 funktioniert die Interaktion mit Photoshop. Das Zusammenspiel wurde allerdings in CS5 noch einmal komplett überarbeitet und verbessert. Um schnell einen Bildausschnitt zu erstellen, müssen Sie nicht wie bisher dieses Bild in Photoshop bearbeiten, speichern und dann in Dreamweaver einfügen.

Erstellen Sie einfach in Photoshop eine Auswahl um den gewünschten Bildausschnitt, und kopieren Sie den Bereich nach Dreamweaver.

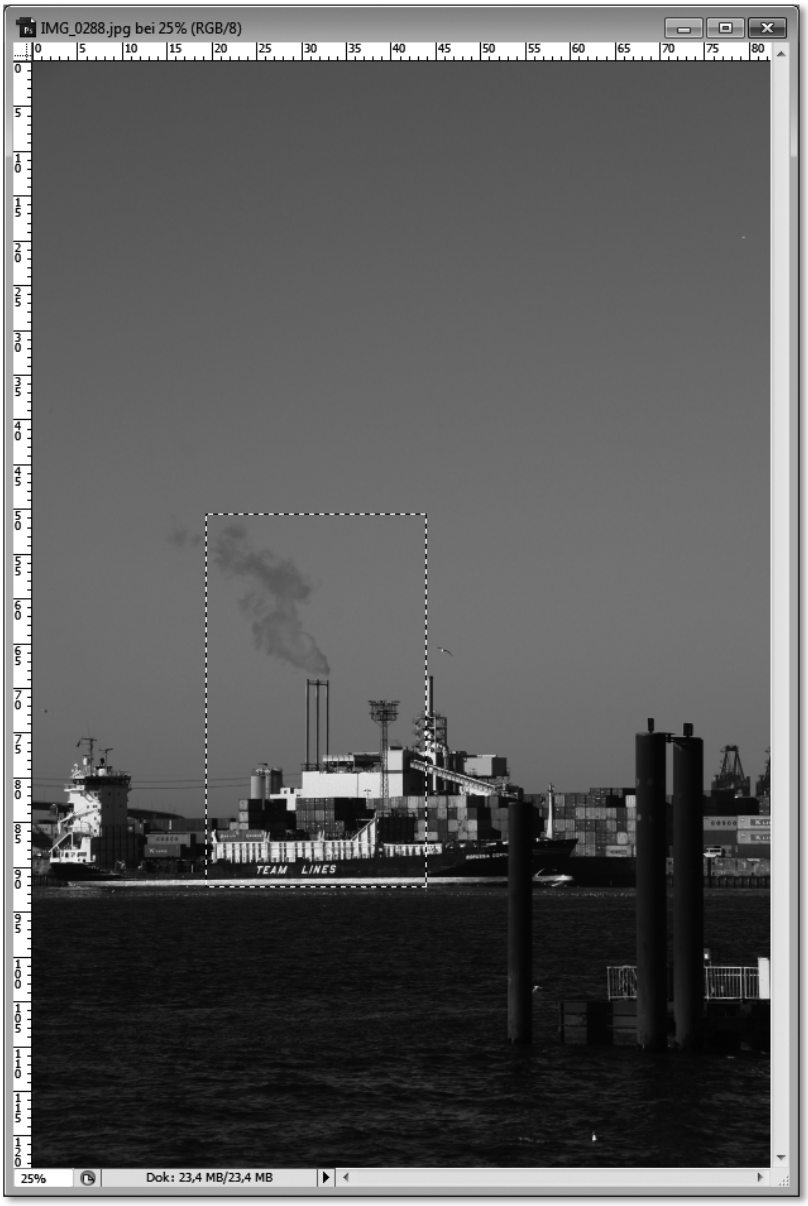


Abbildung 7.14 Bildausschnitt in Photoshop

Nach dem Wechsel zu Dreamweaver wählen Sie einfach Bearbeiten • Einfügen oder drücken die Tastenkombination Strg/⌘+V. In Dreamweaver öffnet sich ein Dialog, in dem Sie die Optimierungseinstellungen für den Bildausschnitt vornehmen können. Diese Einstellungen entsprechen den Photoshop-Einstellungen.

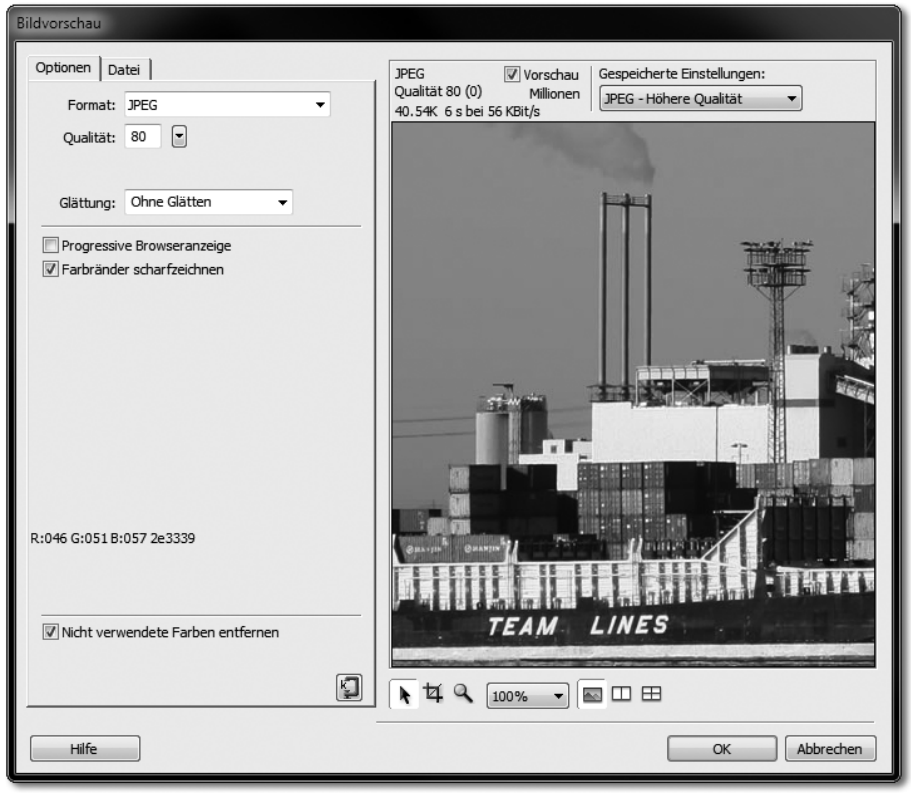


Abbildung 7.15 Bildoptimierung in Dreamweaver

Nach dem Angeben der gewünschten Einstellungen fordert Dreamweaver Sie zum Speichern des Bildes auf und zeigt es in der Website an.

Das Besondere ist, dass sich Dreamweaver merkt, welche Datei die Originaldatei war. Eine Änderung am Original wird von Dreamweaver erkannt und mit einem Symbol im Bild angezeigt. Nun steht Ihnen auch die Funktion NEU SYNCHRONISIEREN ❶ zur Verfügung. Ein Klick darauf baut das Bild aus dem Original neu auf. Verwendet werden dabei die Einstellungen, die Sie für die Komprimierung dieses Bildes angegeben haben.

Mit diesem Schritt ist es nun auch möglich, eine Photoshop-PSD-Datei quasi direkt in eine Webseite zu setzen. Natürlich wandelt Dreamweaver diese dann sofort in eine JPEG-Kopie um, denn PSD-Dateien eignen sich natürlich nicht für das Internet. Bearbeitbar ist die JPEG-Datei trotzdem so, als wäre es ein Photoshop-Bild.



Abbildung 7.16 Kopiertes Bild in Dreamweaver

Dateibrowser Adobe Bridge

[«]

Unter dem Menüpunkt DATEI • BRIDGE DURCHSUCHEN finden Sie den Dateibrowser Adobe Bridge (siehe Abbildung 7.17). Dieses Programm bietet eine sehr gute Übersicht über alle auf Ihrem Rechner vorhandenen Bilder und erleichtert die Suche ungemein. Bilder können unter anderem auch direkt aus dem Bridge-Fenster in das Dreamweaver-Layout gezogen werden. Auch andere Bildbrowser wie FastStone, IrfanView etc. sind bestens geeignet. Sie verfügen oft nicht über die Möglichkeiten von Bridge, sind aber in der Regel deutlich schneller und zudem kostenlos.



Abbildung 7.17 Dateibrowser Adobe Bridge

Framesets – kaum ein anderes Thema bietet so viel Diskussionsstoff unter Webdesignern. Wir zeigen Ihnen, wie Sie Framesets mit Dreamweaver sinnvoll einsetzen.

8 Framesets

Framesets sind aus dem Internet nicht wegzudenken, und doch hört man viele verschiedene Meinungen zu diesem Thema. Die einen verteufeln Framesets als Überbleibsel prähistorischer HTML-Schreiberei, die anderen sehen darin für sich die Lösung aller Probleme.

Die Antwort liegt wie so oft dazwischen. Grundsätzlich gilt mittlerweile: Wenn Sie auf Frames verzichten können und ein Layout mit CSS realisierbar ist, verzichten Sie auf Frames – denn barrierefrei sind Websites mit Frames definitiv nicht.

Framesets haben ganz klar Nachteile (wobei einige Nachteile allerdings auch erst durch den fehlerhaften Umgang mit Framesets entstehen). Es gibt aber auch Anwendungsgebiete, wo sie durchaus ihre Daseinsberechtigung haben, wie z. B. Administrationsoberflächen oder die Pflege älterer Bestandwebsites. Wenn Benutzer mit vermutlich alten Browsern zu Ihrer Zielgruppe gehören, können Frames ebenfalls zum Einsatz kommen.

8.1 Funktionsweise von Frames

Frames teilen das Browserfenster in getrennte Bereiche auf, um gleichzeitig verschiedene Dokumente unabhängig voneinander darstellen zu können. Framesets bestehen aus einem Hauptdokument, das das Frameset enthält, und den in den Frames dargestellten Dokumenten. Daher besteht z. B. ein komplettes Frameset für drei gleichzeitig darzustellende Dokumente aus vier Dokumenten.

8.1.1 Gestalten mit Framesets

Mit Framesets lassen sich bildschirmfüllende Designs realisieren, die mit einem reinen CSS- oder Tabellenlayout kaum erreichbar wären.

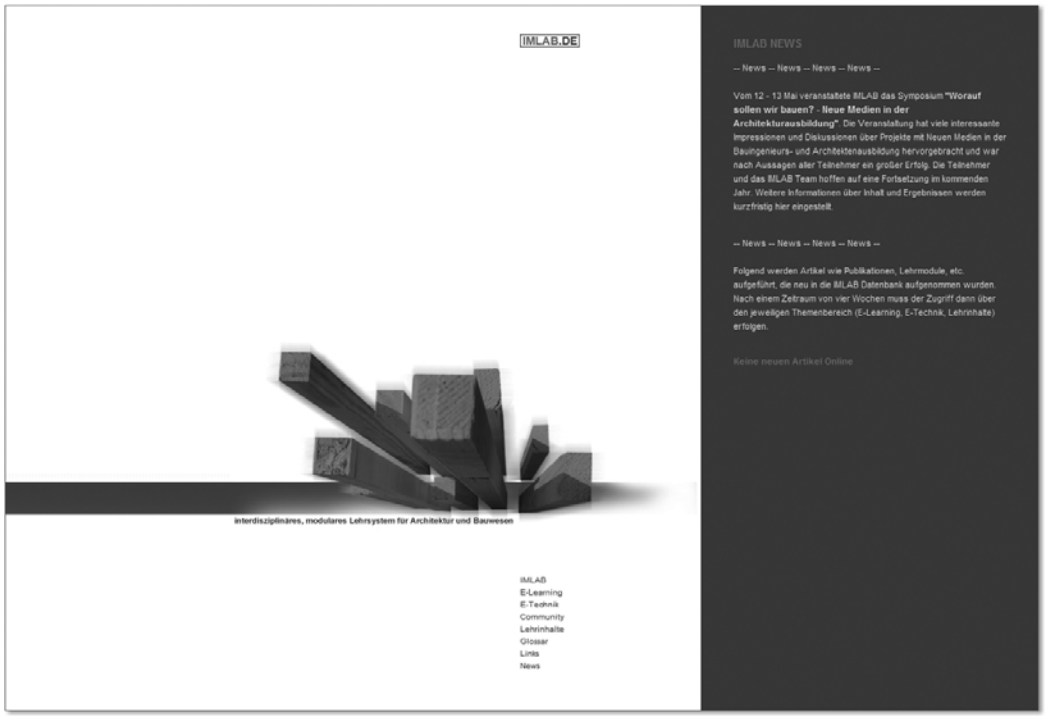


Abbildung 8.1 Website eines Projekts mit seitlichen Frames

In Abbildung 8.2 sehen Sie ein verschachteltes Frameset, mit dem die eigentliche Inhaltsseite in der Bildschirmmitte zentriert wird – eine Vorgehensweise, die lange in der Praxis verwendet wurde. Auch hier haben CSS die Frames weitgehend abgelöst.



Abbildung 8.2 Website mit zentrierten Frames

8.1.2 Suchmaschinen und Framesets

Eines der Hauptargumente gegen Framesets ist die angeblich schlechte Indizierung in Suchmaschinen. Wir können dies aus der Praxis nicht bestätigen. Zumindest bei Google, der wohl aktuell wichtigsten Suchmaschine, trifft die Behauptung nicht zu. Natürlich muss man aber einige Besonderheiten beachten.

Metaangaben im »noframes«-Bereich

Eine Frameset-Datei ist eigentlich nur ein Verweis auf weitere Dokumente und Darstellungseinstellungen für den Browser. Das bedeutet, dass ein Frameset zunächst gar keine Inhalte hat, die eine Suchmaschine indizieren könnte, außer den hoffentlich vorhandenen Metaangaben.

In einer Frameset-Datei gibt es einen Bereich, der noch aus Zeiten stammt, als noch nicht alle Browser Frames darstellen konnten. In diesem `<noframes>`-Bereich können Sie jede Menge Texte und Verweise verstecken. Diese werden von keinem aktuellen Browser angezeigt, von den Suchmaschinen jedoch sehr wohl als Inhalt erkannt.

```
<noframes>
<body>
</body>
</noframes>
```

Listing 8.1 `<noframes>`-Bereich eines Framesets

Frameset nachladen

Das bietet Ihnen natürlich die Möglichkeit, Ihre Seite genau und ausführlich zu beschreiben. Zusätzlich können Sie auf jede Ihrer Unterseiten verweisen und somit die Suchmaschine veranlassen, auch diese zu indizieren. Auf den Unterseiten sorgen Sie dann mit einem kleinen JavaScript dafür, dass das Frameset automatisch nachgeladen wird, wenn das Dokument direkt aufgerufen wird.

```
<script>
if (window.name!='mainFrame')
 top.location.replace('NAME DES FRAMESETS?NAME DER
 DATEI~NAME DES FRAMES');
</script>
```

Listing 8.2 JavaScript zum Nachladen des Framesets

Framesets und dynamische Websites

Dreamweaver ist nicht in der Lage, von dynamischen Websites innerhalb eines Framesets eine lokale Vorschau zu erstellen. Testen Sie diese Seite auf Ihrem Webserver, oder navigieren Sie über den Browser zu dem Frameset.

8.2 Ein Frameset anlegen

Der Aufbau eines Framesets mit Dreamweaver ist etwas gewöhnungsbedürftig, letztlich aber ganz einfach.

Einzeldokumente anlegen

[o] Erstellen Sie vor dem Anlegen des Framesets die einzelnen darzustellenden Dokumente. In unserem Beispiel sind dies *a.htm*, *b.htm* und *c.htm*. Diese Dokumente enthalten in unserem Beispiel nichts außer jeweils einer anderen Hintergrundfarbe. In der Praxis würde eins der Dokumente die Navigation, ein anderes die Headline mit Logo und das dritte die eigentlichen Inhalte enthalten. Die Dokumente dienen in unserem Fall nur der Veranschaulichung. Sie finden sie auf der DVD unter *Beispiele/Framesets*.

Legen Sie ein neues leeres Dokument an, klicken Sie dann im Register Layout in der Einfügeleiste ganz rechts auf das Icon Frames, und wählen Sie Frames oben und links verschachtelt aus.



Abbildung 8.3 Ein Frameset über das Auswahlmenü erstellen

Die Frames werden im Dokument sichtbar. Jetzt aktivieren wir über Fenster • Frames oder Alt + F2 das Eigenschaftenfenster für Frames.

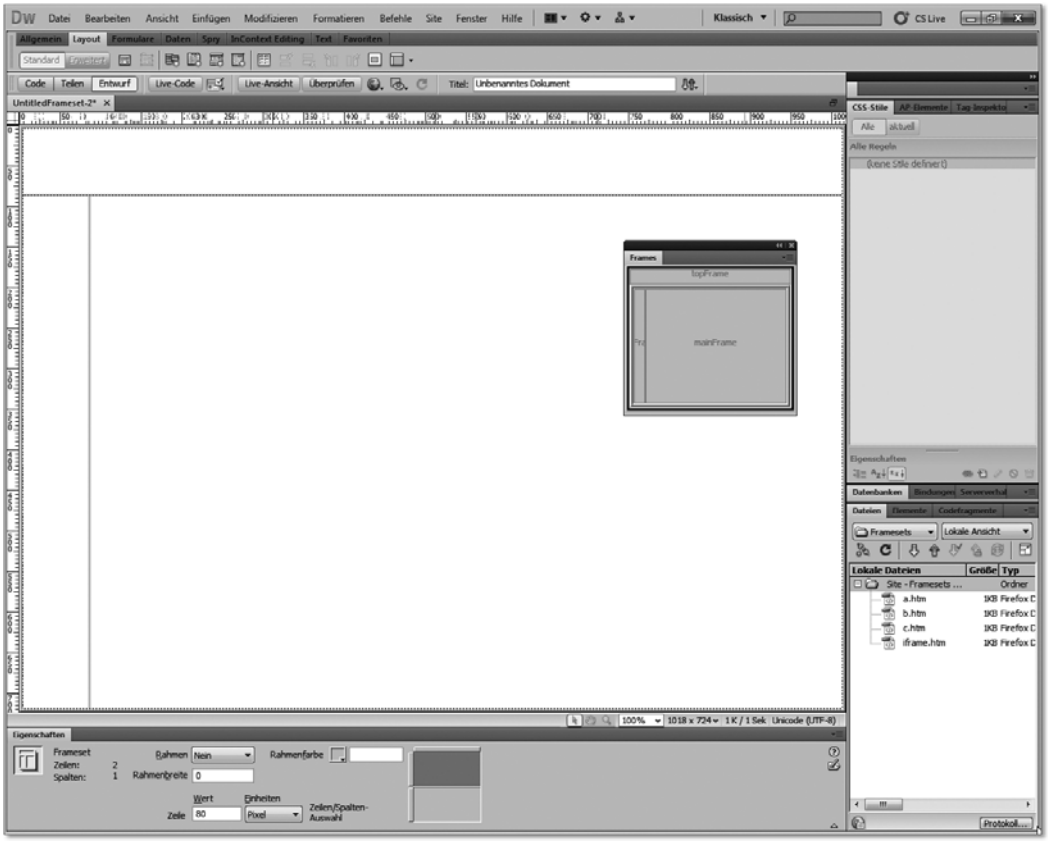


Abbildung 8.4 Einstellen des Framesets

Dateien zuweisen

Klicken Sie in das Bedienfeld FRAMES, um den gewünschten Frame auszuwählen, und ziehen Sie mit der Maus den kleinen Kreis bei QUELLE ❶ (siehe Abbildung 8.5) im Eigenschaftenfenster auf die darzustellende Datei ❷.

Wiederholen Sie diesen Schritt für alle darzustellenden Dateien.

Frame-Rahmen

Stellen Sie bei allen Frames die RANDBREITE und die RANDHÖHE auf 0, um eine einheitlich definierte Größe zu erhalten. Stellen Sie unter RAHMEN die Option NEIN ein, denn in den meisten Fällen werden keine Frame-Rahmen benötigt. Frame-Rahmen werden in älteren Browsern häufig nicht korrekt dargestellt. Besonders auf dem Mac zeigten sich in älteren IE-Versionen oft hässliche Frame-Rahmen.

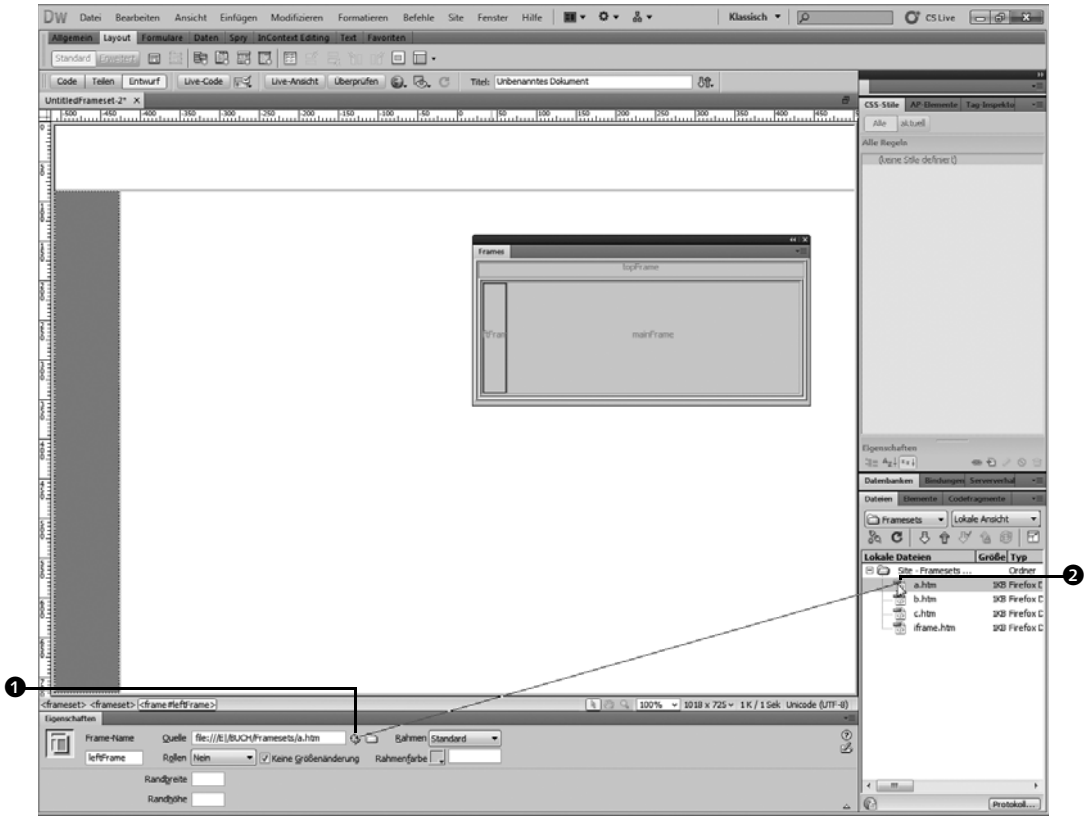


Abbildung 8.5 Zuweisen der Frameinhalte

Automatische Scrollbalken

Ganz wichtig ist die Einstellung bei ROLLEN ❸ (siehe Abbildung 8.6). Hier legen Sie fest, ob ein Frame einen Inhalt scrollbar darstellen darf oder nicht. Für feste Bereiche wie Navigationen sollten Sie dies deaktivieren. Im Hauptfenster wird es jedoch meistens auf AUTO gestellt. So erscheinen Scrollbalken nur dann, wenn sie auch benötigt werden.

Frame-Namen

Jeder einzelne Frame erhält einen eigenen Namen, in Abbildung 8.6 z. B. *leftFrame*. Bedenken Sie, dass die von Dreamweaver vergebenen Standardnamen auf Tausenden von Websites vorkommen. Wenn Sie mehrere Browserfenster geöffnet haben und die Websites Frames enthalten, kann es vorkommen, dass Inhalte in einem falschen Frame geöffnet werden. Vergeben Sie hier also nach Möglichkeit eindeutige und einmalige Namen. In unserem Beispiel haben wir die Standardbezeichnungen von Dreamweaver übernommen.

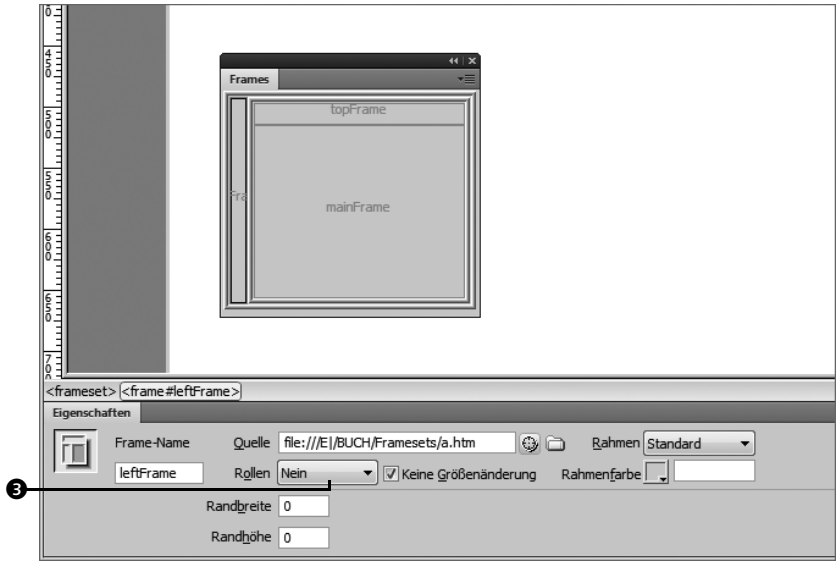


Abbildung 8.6 Einstellen der Frames

Größe der Frames

Nachdem die Dokumente den Frames zugewiesen wurden, müssen noch die richtige Breite und Höhe für die Frames eingestellt werden (siehe Abbildung 8.7). Klicken Sie dazu direkt auf den Frame-Rahmen im Dokumentfenster.

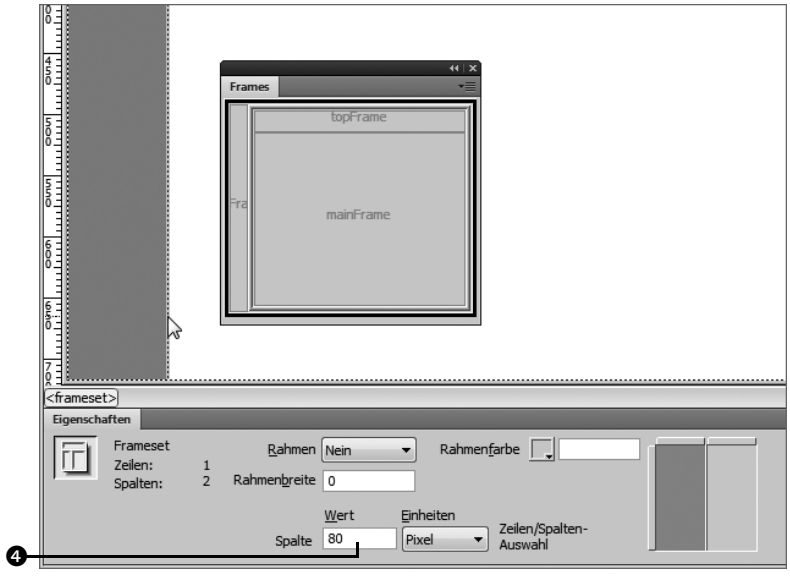


Abbildung 8.7 Abmessungen einstellen

Auch hier setzen wir wieder die Rahmen auf Nein und die Rahmenbreite auf 0. Unter Wert ❹ geben Sie jetzt die exakte Breite oder Höhe des Frames in Pixel oder in Prozent an.

Nachdem Sie alle Schritte durchgeführt haben, muss das Frameset noch abgespeichert werden. Klicken Sie dazu auf Datei • Frameset speichern unter, und speichern Sie das Frameset unter einem eigenen Namen ab.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Unbenanntes Dokument</title>
</head>
<frameset rows="*" cols="80,*" frameborder="no" border="0"
  framespacing="0">
  <frame src="a.htm" name="leftFrame" scrolling="No"
  noresize="noresize" marginwidth="0" marginheight="0"
  id="leftFrame" />
  <frameset rows="80,*" frameborder="no" border="0" framespacing="0">
    <frame src="b.htm" name="topFrame" scrolling="No"
    noresize="noresize" id="topFrame" />
    <frame src="c.htm" name="mainFrame" id="mainFrame" />
  </frameset>
</frameset>
<noframes><body>
</body></noframes>
</html>
```

Listing 8.3 Das fertige Frameset im Quelltext

8.3 Mit Framesets arbeiten

Die Arbeit mit Framesets in Dreamweaver ist recht komfortabel. Wenn Sie ein Frameset aufrufen, sehen Sie alle Dokumente gleichzeitig im Dokumentfenster. Sie können jedes der einzelnen Dokumente direkt bearbeiten und sehen immer das spätere Ergebnis im Zusammenspiel mit den anderen Dokumenten des Framesets.

Achten Sie darauf, dass nicht zufällig eine der im Frameset dargestellten und nun bearbeiteten Dateien in Dreamweaver noch einmal als einzelnes Dokument geöffnet ist. Dreamweaver erkennt dies nicht, und so können Änderungen einen Versionskonflikt hervorrufen.

8.3.1 Framesets nachbearbeiten

Wenn Sie das Frameset nachträglich bearbeiten wollen, geht das am besten im Quelltext. Klicken Sie dazu im Dokumentfenster wie im Frame-Fenster auf einen Rahmen. Dann ist nicht mehr ein Inhaltsdokument aktiv, sondern das Frameset selbst. Jetzt können Sie in die Codeansicht umschalten und dann die gewünschten Veränderungen vornehmen.

Probleme bei verschachtelten Frames [«]

Es kommt vor, dass Framesets auf zwei Dateien verteilt werden, wobei der Inhalt eines Frames ein weiteres Frameset ist. Dreamweaver hat Schwierigkeiten, diese Frameset-Dateien zu bearbeiten, und öffnet immer den Quellcode der untergeordneten Datei. Öffnen Sie in diesem Fall die gewünschte Datei mit einem Texteditor, und bearbeiten Sie diese außerhalb von Dreamweaver.

8.3.2 Verlinkungen in Framesets

Sie können Frame-Dateien wie gewohnt verlinken, nur müssen Sie zusätzlich den Ort angeben, an dem sich der Link öffnen soll. Nach erfolgter Verlinkung erscheint in der Eigenschaftenpalette das Auswahlmenü Ziel, unter anderem auch mit den Namen der einzelnen Frames. Dokumente werden in dem ausgewählten Ziel-Frame dargestellt.

Wenn Sie kein Ziel angeben, wird das nächste Dokument im gleichen Frame geladen wie das aktuelle Dokument, das den Link enthält.

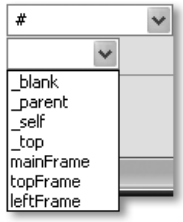


Abbildung 8.8 Ziele für Links in Framesets

8.3.3 Mehrere Frames gleichzeitig neu laden

Wird ein Link angeklickt, müssen häufig neue Inhalte nicht nur in einem, sondern in mehreren Frames neu geladen werden. Diese Aufgabe ist mit HTML nicht zu erreichen, da damit immer nur ein Ziel-Frame angegeben werden kann. An diesem Punkt kommt JavaScript zum Einsatz.

Leider unterstützt Dreamweaver den Austausch mehrerer Frames nicht standardisiert. Sie können sich jedoch mit der in Abschnitt 8.2, »Ein Frameset anlegen«,

beschriebenen Methode Gehe zu URL behelfen oder das nachfolgende Skript verwenden.

Zwei Frames austauschen

Wenn Sie mit zwei Frames arbeiten, kopieren Sie das JavaScript aus Listing 8.3 in den <head>-Bereich Ihres Frameset-Dokuments:

```
<script type="text/javascript">
<!--
function ZweiFrames(URL1,Frame1,URL2, Frame2) {
 Frame1=eval("parent."+ Frame1);
 Frame2=eval("parent."+ Frame2);
 Frame1.location.href = URL1;
 Frame2.location.href = URL2;
}
//-->
</script>
```

Listing 8.4 Austausch von zwei Frame-Inhalten

Die Links in Ihrem Navigationsmenü müssen Sie nun nach folgendem Schema umschreiben:

```
<a href="javascript:ZweiFrames('URL1','Frame1','URL2',
'Frame2')">LINK</a>
```

Drei Frames austauschen

Arbeiten Sie mit drei Frames, müssen Sie das JavaScript aus Listing 8.4 in den <head>-Bereich Ihres Frameset-Dokuments kopieren:

```
<script type="text/javascript">
<!--
function DreiFrames(URL1, Frame1,URL2, Frame2,URL3, Frame3) {
 Frame1=eval("parent."+ Frame1);
 Frame2=eval("parent."+ Frame2);
 Frame3=eval("parent."+ Frame3);
 Frame1.location.href = URL1;
 Frame2.location.href = URL2;
 Frame3.location.href = URL3;
}
//-->
</script>
```

Listing 8.5 Austausch von drei Frame-Inhalten

Verändern Sie die Links in dem Menü wie folgt:

```
<a href="javascript:DreiFrames('URL1','Frame1','URL2',
'Frame2','URL3','Frame3')">LINK</a>
```

Der Aufruf des JavaScripts in den `<a href>`-Tags übergibt die beiden Parameter `URL` und `Frame` an das JavaScript im `<head>`-Bereich und führt es aus.

Nur für User mit aktivem JavaScript benutzbar [!]

Leider funktioniert diese Navigation nur, wenn der User JavaScript aktiviert hat. Wenn Sie auch Benutzer mit deaktiviertem JavaScript erreichen wollen, müssen Sie eine zweite Variante der Website mit einer Navigation ohne JavaScript erstellen. Im Regelfall haben aber alle Browser JavaScript aktiviert. Ohne JavaScript würden sonst auch keine Spry-Menüs oder Flash-Filme funktionieren.

Dreamweaver-Verhalten einsetzen

Mehrere Frames können auch über ein Dreamweaver-Verhalten gleichzeitig geladen werden. Das Verhalten Gehe zu URL lädt ein HTML-Dokument an ein angegebenes Ziel. Wenn Sie dieses Verhalten zum Beispiel einer Schaltfläche mehrfach zuweisen, können auch mehrere Frames und mehrere Dokumente angegeben werden.

Der Nachteil dieser Arbeitsweise sind das deutlich größere JavaScript im Dokument sowie die unübersichtliche Darstellung. Eine Möglichkeit, einzelne Verlinkungen der Dokumente nachträglich zu verändern, gibt es dann nicht.

8.3.4 Eingebettete Frames – <iframe>

Eingebettete Frames sind eine Möglichkeit, HTML-Dokumente innerhalb eines anderen HTML-Dokuments anzuzeigen. Ein solcher Frame hat etwa die gleichen Eigenschaften wie ein Bild in HTML, allerdings wird an genau dieser Stelle nicht eine Bilddatei, sondern eine andere HTML-Datei dargestellt. Ein Beispiel sehen Sie in Abbildung 8.9

In der aktuellen Dreamweaver-Version kann ein eingebetteter Frame über Icon ❶ (siehe Abbildung 8.10) in das Dokument eingefügt werden. Leider fehlt hierzu jegliche Einstellmöglichkeit, so dass Sie komplett im Quelltext arbeiten müssen. Einzig das `<iframe>`-Icon ist vorhanden. Dazu wechselt Dreamweaver automatisch in die Quelltext-Ansicht. In der Design-Ansicht wird lediglich ein grauer Bereich entsprechend der Frame-Größe dargestellt.

Sollten Sie eingebettete Frames verwenden wollen, empfehlen wir Ihnen die Onlinereferenz SelfHTML (*http://de.selfhtml.org/*) von Stefan Münz, um einen Überblick über die verschiedenen Möglichkeiten zu bekommen.

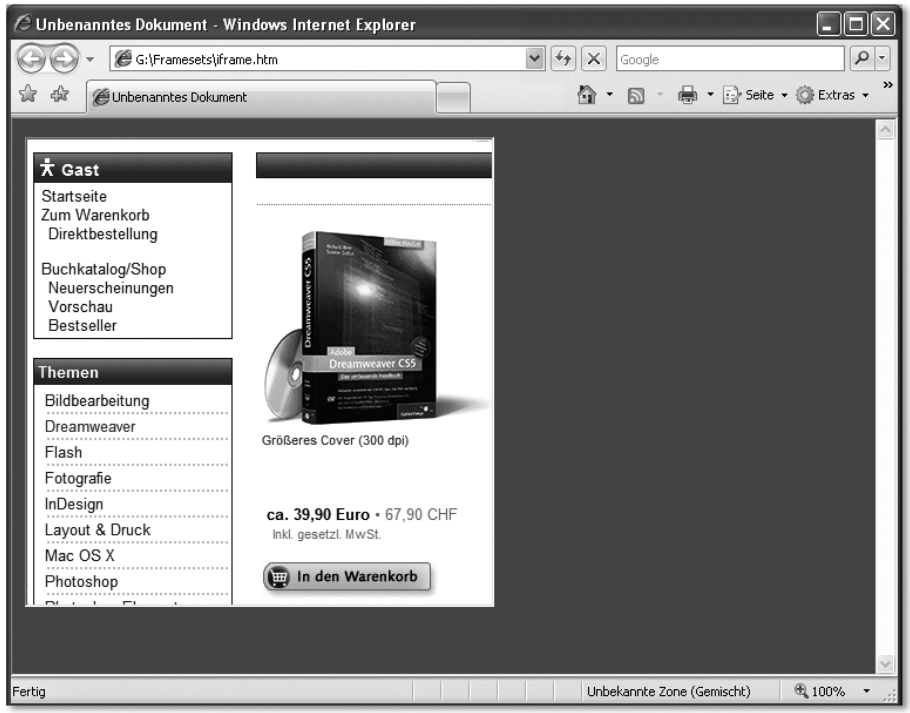


Abbildung 8.9 Die Website von Galileo Design in eigenem HTML-Dokument



Abbildung 8.10 Einfügen eines eingebetteten Frames (`<iframe>`)

Die Eigenschaften sind aber in etwa jene, die auch einem `<image>`-Tag zugeordnet werden: Breite (`width`), Höhe (`height`), Rand (`frameborder`) und Inhaltsquelle (`src`).

<iframe> und Hackerangriff

Eingebettete Frames können auf den Wert »null« gesetzt werden, so dass externer Quellcode und auch Skripte unsichtbar bleiben. Besonders der Internet Explorer ist bekannt für `<iframe>`-Sicherheitslücken. Mittlerweile sind allerdings auch schon Angriffe über Firefox bekannt geworden.

JavaScript ist nicht jedermanns Sache. Mit Dreamweaver ist das Erstellen diverser Standardaktionen ein Kinderspiel. Wir zeigen Ihnen, wie es geht und was Sie beachten müssen.

9 JavaScript und Verhalten

JavaScript ist eigentlich ganz einfach, wenn nicht wieder verschiedene Browsermodelle mit unterschiedlichen Anforderungen aufwarten würden. Hier ist es leider nicht bloß so, dass Internet Explorer und Firefox ab und zu ein anderes Verständnis von einzelnen Befehlen haben – das gesamte Dokumentenobjektmodell ist grundsätzlich verschieden, so dass viele Skripte für IE und Firefox doppelt angelegt werden müssen.

Zum Glück haben wir mittlerweile Werkzeuge wie Dreamweaver, die uns diese Arbeit abnehmen. Zugegebenermaßen ist der generierte Code von Dreamweaver nicht wirklich gut zu verstehen, und Änderungen daran sind kaum möglich, ohne sich in die Tiefen der JavaScript-Programmierung zu begeben. Wer das jedoch macht, wird seine Skripte sowieso von Hand schreiben.

Fakt ist, dass der Code von Dreamweaver funktioniert und auf allen Plattformen – die richtigen Einstellungen vorausgesetzt – definitiv läuft.

9.1 JavaScript in Dreamweaver

In Abbildung 9.1 sehen Sie den grundsätzlichen Aufbau einer JavaScript-Aktion (Verhalten) in Dreamweaver in einer schematischen Darstellung.



Abbildung 9.1 Schema von JavaScript-Aktionen

Ereignis

Sie benötigen demnach einen Auslöser. Dies kann eine Schaltfläche, das `<body>`-Tag usw. sein. Dieser Auslöser setzt beim Eintreten eines zu definierenden Ereignisses eine Aktion in Gang, die mit einem ebenfalls zu definierenden Objekt das ausführt, was in der Aktion festgeschrieben ist.

Als Beispiel soll uns ein Rollover-Effekt für eine Schaltfläche dienen – wahrscheinlich einer der am häufigsten eingesetzten Effekte mit JavaScript.

Das Ereignis ist die Bewegung der Maus über einen Link bzw. eine Grafik, die verlinkt ist. Das ausgelöste Verhalten bewirkt, dass sich ein Bildobjekt verändert. Dabei wird der Inhalt des Objekts gegen einen anderen Inhalt ausgetauscht. Das Skript an der Schaltfläche sieht wie folgt aus:

```
<a href="javascript:;" onMouseOver="MM_swapImage('Link1','',
'pics/1_over.gif',0)"><img src="pics/
1.gif" name="Link1" width="109" height="36" border="0" id="Link1">
</a>
```

Listing 9.1 Skript für Rollover-Effekt im HTML-Body

Hierbei ist `onMouseOver` das auslösende Ereignis. `MM_swapImage` ist die Aktion bzw. der Name des Skripts im Head des Dokuments, das beim Eintreten des Ereignisses ausgeführt wird (siehe Listing 9.2). `Link1` ist die Bezeichnung für das Objekt, mit dem die Aktion ausgeführt werden soll. In diesem Fall sind der Auslöser und das Objekt, mit dem die Aktion ausgeführt wird, identisch, da durch bei einem Rollover das auslösende Objekt selbst verändert wird.

```
function MM_swapImage() { //v3.0
 var i,j=0,x,a=MM_swapImage.arguments; document.MM_sr=
 new Array; for(i=0;i<(a.length-2);i+=3)
  if (x=MM_findObj(a[i])!=null){document.MM_sr[j++]=x;
if(!x.oSrc) x.oSrc=x.src; x.src=a[i+2];}
}
```

Listing 9.2 Skript für Rollover-Effekt im HTML-Head

9.2 Verhalten

Die in Dreamweaver integrierten Verhalten sind für die meisten Websites völlig ausreichend. Bei Bedarf können Sie Aktionen bzw. Verhalten natürlich auch als Erweiterung installieren. Die meisten Erweiterungen für Dreamweaver sind Verhalten, die dann in der Bedienfeldgruppe Verhalten angezeigt werden. Websites mit einem Angebot an Erweiterungen sind oft in englischer Sprache. Verhalten sind dann unter dem Begriff *Behavior* (oder *Behaviour*) zu finden.

Bedienfeld »Verhalten«

Um mit Verhalten zu arbeiten, öffnen Sie das Bedienfeld VERHALTEN über das Menü FENSTER • VERHALTEN oder mit der Tastenkombination ⇧+F4. Klicken Sie im Fenster auf das kleine Pluszeichen, und es erscheint die Palette mit den möglichen Verhaltensaktionen.

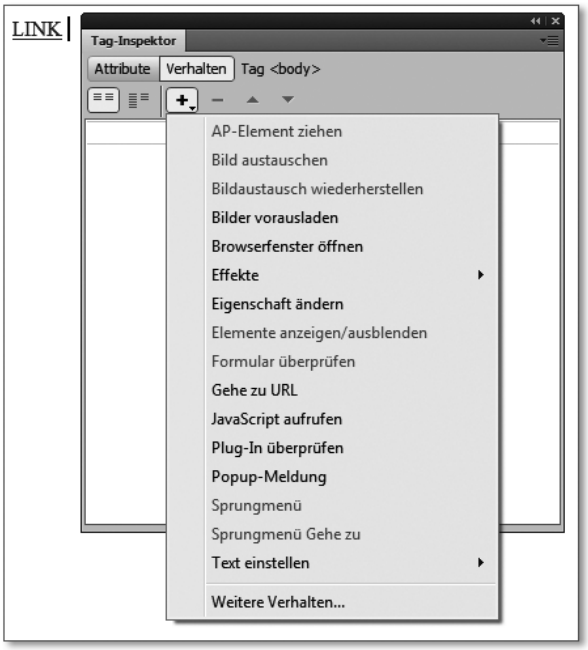


Abbildung 9.2 Das Bedienfeld »Verhalten«

9.2.1 Rollover erstellen

Um ein Verhalten zu erstellen, müssen Sie das Element, das auf ein bestimmtes Ereignis reagieren soll, in Ihrem Dokument markieren. Das können Bilder, Verlinkungen oder auch das gesamte Dokument sein.

In unserem Beispiel soll mit dem Bild NEWS ❶ (siehe Abbildung 9.3) eine Aktion verbunden werden. Wenn Sie ein Bild anwählen, haben Sie in der Eigenschaftenpalette die Möglichkeit, diesem einen Namen zu geben ❸. Erst durch eine vom Dateinamen unabhängige Bezeichnung eines Elements kann dieses mit JavaScript angesprochen werden. Wenn ein Element einen vom Dateinamen unabhängigen Namen hat, spielt der Inhalt des Elements keine – oder kaum eine – Rolle. In unserem Beispiel benennen wir das Element einfach nur mit »Bild«.

Automatische Namensvergabe

Dreamweaver vergibt auch automatisch Namen für Bilder, denen Verhalten hinzugefügt werden. Wir raten Ihnen jedoch, den Namen selbst zu definieren; Dreamweaver numeriert die Bilder einfach unübersichtlich durch. Vergeben Sie immer eindeutige Namen, und erleichtern Sie sich so weitere Schritte. Achten Sie bei den Bezeichnungen darauf, dass Sie keine Leerzeichen und Sonderzeichen verwenden.

Durch Klicken auf das Pluszeichen ❷ im Bedienfeld VERHALTEN können Sie dem Objekt, das Sie auswählen, ein Verhalten zuweisen. In unserem Fall ist es das Verhalten BILD AUSTAUSCHEN.

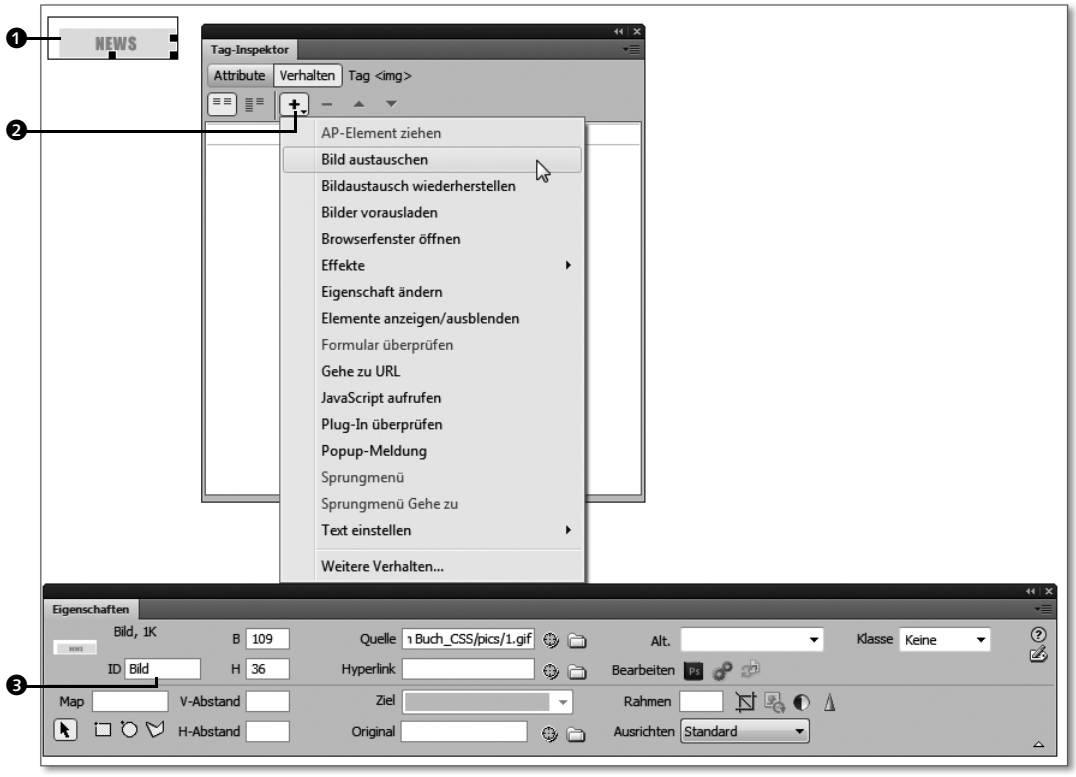


Abbildung 9.3 Vorgaben für Verhalten

Wie Sie in Abbildung 9.4 sehen, erscheint in der Dialogbox der Name Ihres Bildes. Alle anderen Bildobjekte in Ihrem Dokument wurden von Dreamweaver automatisch benannt. Klicken Sie jetzt auf DURCHSUCHEN, um das Bild auszuwählen, das Sie bei dem entsprechenden Ereignis anzeigen lassen möchten.

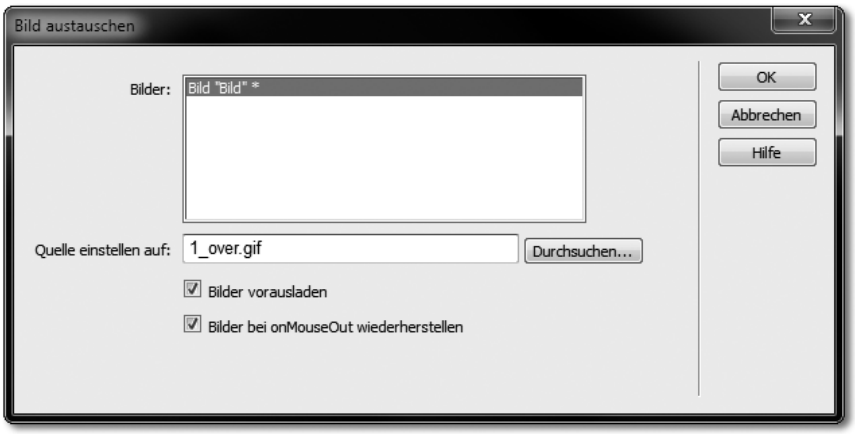


Abbildung 9.4 Bild für Rollover-Aktion auswählen

Da es sich um ein Rollover handelt, bietet Ihnen Dreamweaver in weiser Voraussicht zwei weitere Optionen an:

- Bilder vorausladen
  Im Normalfall werden Bilder in einer Website dann geladen, wenn sie auch dargestellt werden sollen. Bei einem Rollover wäre dies zu spät. Bis das Bild dann geladen würde, haben Sie die Aktion längst wieder beendet. Damit dies nicht geschieht, fügt Dreamweaver ein JavaScript ein, das dafür sorgt, dass Rollover-Bilder gleich mit dem Hauptdokument geladen werden. Sie befinden sich dann im Browser-Cache und werden sofort bei der Aktion eingeblendet.
- Bilder bei onMouseOut wiederherstellen
  Sie möchten sicherlich, dass Ihre Schaltfläche wieder den normalen Zustand anzeigt, wenn die Maus sie wieder verlässt. Bei einem einfachen JavaScript, das nur das Bild bei einem Ereignis austauscht, wäre dies nicht der Fall. Aus diesem Grund wird hier bereits eine zweite Aktion zum Wiederherstellen des Ausgangszustandes eingefügt.

Nach Abschluss der Arbeiten sehen Sie im Bedienfeld Verhalten die neu eingetragenen Aktionen wie in Abbildung 9.5 dargestellt. Wenn Sie in der Tag-Palette eines der Ereignisse anklicken ❷, sehen Sie eine Liste ❸, in der Sie die Art der Ereignisse im Rahmen der von Ihnen gewählten Vorgaben einstellen können. Rechts daneben werden die Aktionen ❺ aufgelistet, in unserem Fall der Bildaustausch und die Wiederherstellung des Bildaustauschs.

Durch Klicken auf Verhalten hinzufügen ❶ und Ereignis entfernen ❹ können Aktionen hinzugefügt oder entfernt werden.

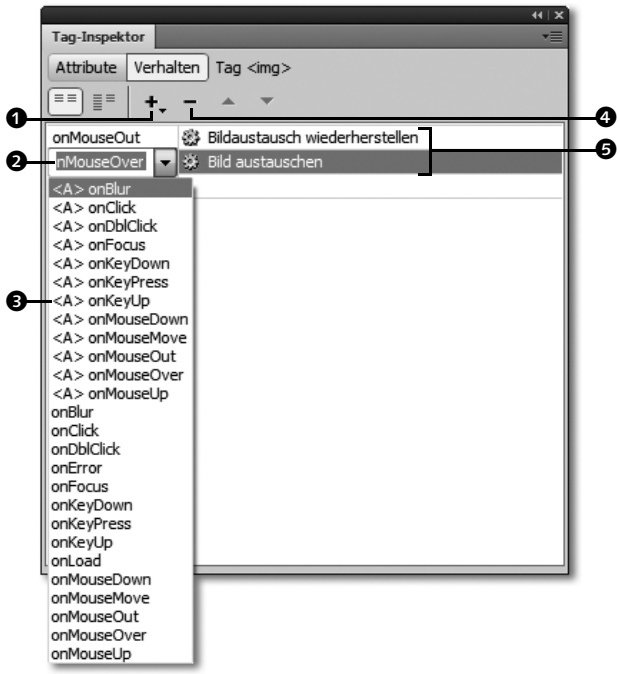


Abbildung 9.5 Bedienfeld »Verhalten« mit möglichen Aktionen

Mit den nach oben und nach unten weisenden Pfeilen kann die Reihenfolge der Aktionen geändert werden. Manchmal kommt es vor, dass bei vielen definierten Aktionen einiges nicht so läuft, wie vorgesehen. Dann ist es oft hilfreich, die Reihenfolge der Aktionen zu verändern, und es funktioniert.

Rollover mit CSS

Wenn Sie, wie von uns empfohlen, mit CSS-Layouts arbeiten, können Bild-Rollover auch mit CSS und `:hover` erstellt werden. Wie das geht, lesen Sie in Kapitel 11 über CSS.

9.2.2 Plugins überprüfen

Plugins sind zusätzliche Programme, die in einem Browser installiert werden, um bestimmte Funktionen oder Darstellungen zu ermöglichen. Das bekannteste Beispiel dafür ist das Plugin für den Flash Player. Nicht alle Browser haben die nötigen Plugins installiert. Daher bietet Dreamweaver dieses Verhalten, um zu prüfen, ob die Voraussetzungen für die korrekte Darstellung der Inhalte erfüllt sind. Wenn nicht, kann auf eine alternative Seite weitergeleitet werden.



Abbildung 9.6 Plugin-Überprüfung

Plugin-Überprüfung einfügen

Zur Dialogbox gelangen Sie wieder über die Bedienfeldgruppe VERHALTEN und den Menüpunkt PLUG-IN ÜBERPRÜFEN. Ist das ausgewählte Plugin im Browser des Benutzers vorhanden, wird die Seite damit angezeigt – ist es nicht vorhanden, wird der User auf eine Alternativseite weitergeleitet.

Notwendig wird dieses durch Dreamweaver automatisch generierte JavaScript beispielsweise bei Websites mit außergewöhnlichen Plugins oder bei solchen mit geringem Verbreitungsgrad. Allerdings raten wir allgemein von der Benutzung unüblicher Plugins ab. Mit Flash, das eine Verbreitung von fast 95 Prozent hat, können Sie im Regelfall alle Extras darstellen, für die Sie sonst ein spezielles Plugin benötigen würden (Beispiele: Video, Musik, Vektorzeichnungen).

Da eine hundertprozentig zuverlässige Plugin-Überprüfung nicht immer möglich ist, könnte man zwar Alternativseiten angeben, der Pflegeaufwand der Website steht aber heutzutage in keinem Verhältnis mehr zum Nutzen, so dass im Allgemeinen nur eine Version existiert und Benutzern mit fehlendem Plugin die automatische Browserwarnung angezeigt wird. Im Normalfall besteht eine Seite auch nicht ausschließlich aus einem Plugin.

9.2.3 Mehrere Frames gleichzeitig austauschen

Die Aktion GEHE ZU URL kann eingesetzt werden, um mehrere Frames gleichzeitig auszutauschen. Lesen Sie dazu auch Kapitel 8, »Framesets«. Dort wird der Hintergrund dieser Aktion erläutert.

Wählen Sie dazu die gewünschte Verlinkung in Ihrem Dokument an, und fügen Sie über das Pluszeichen und VERALTET • GEHE ZU URL zunächst das erste Ziel für die Aktion hinzu. Nachdem die Aktion erfolgreich erstellt wurde, öffnen Sie die Dialogbox GEHE ZU URL durch einen Doppelklick auf diese Aktion im Bedienfeld

Verhalten erneut und fügen ein weiteres Ziel hinzu. Dieses wird der bestehenden Aktion als weiterer Parameter hinzugefügt.

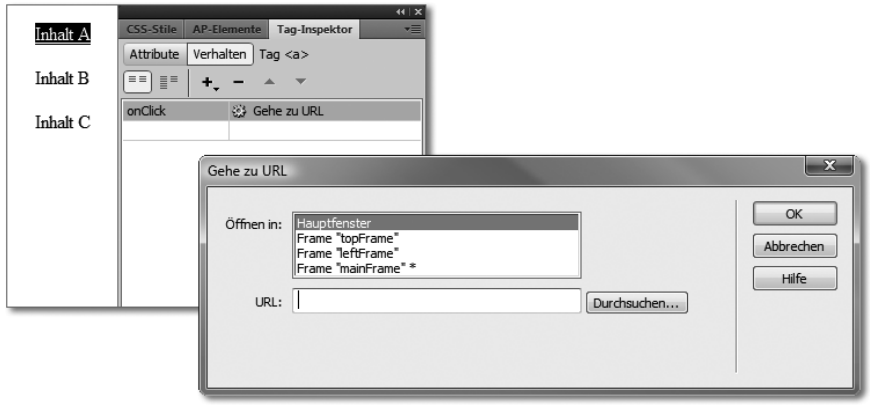


Abbildung 9.7 Verlinkung mehrerer Dokumente in einem Frameset

9.2.4 JavaScript-Effekte

Seit Dreamweaver CS3 finden Sie die Effekte im Bedienfeld Verhalten. Diese dienen in erster Linie dem Überblenden verschiedener HTML-Dokumente oder dem Vergrößern einzelner Bereiche. Diese Effekte sind bereits dem Spry-Framework zuzuordnen und erfordern umfangreichen Einsatz von JavaScript. Grundsätzlich funktionieren diese Effekte wie alle vorgenannten auch. Sie brauchen dazu nur das gewünschte Element anzuwählen und den Effekt zuzuweisen. Dreamweaver öffnet dann eine Dialogbox mit den Parametern.

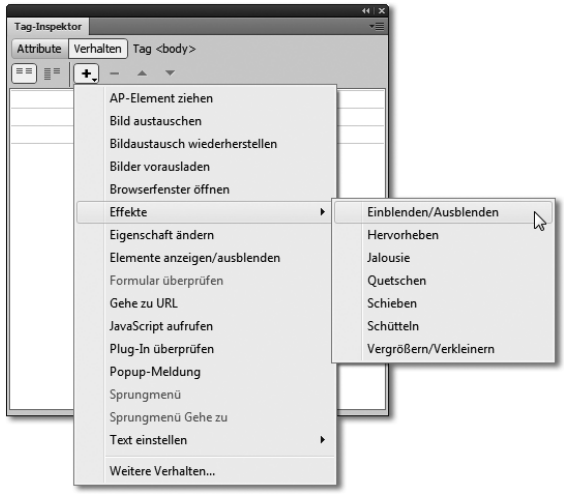


Abbildung 9.8 Spry-Effekte im Bedienfeld »Verhalten«

9.2.5 JavaScript und CSS

In der Bedienfeldgruppe VERHALTEN finden Sie einen Punkt, der sich EIGENSCHAFT ÄNDERN nennt. Hier ist es möglich, jede einzelne CSS-Eigenschaft eines DIV-Containers oder eines anderen mit CSS formatierten Elements dynamisch zur Laufzeit zu manipulieren. Die Möglichkeiten sind – so unscheinbar der Menüpunkt auch ist – enorm. DIV-Container oder andere mit einer ID versehene Elemente in einem XHTML-Dokument können über das *DOM (Document Object Model)* im Rahmen der vorgegebenen Attribute komplett manipuliert werden. In Abbildung 9.9 wird beispielsweise eine Eigenschaft des DIVs links oben geändert.

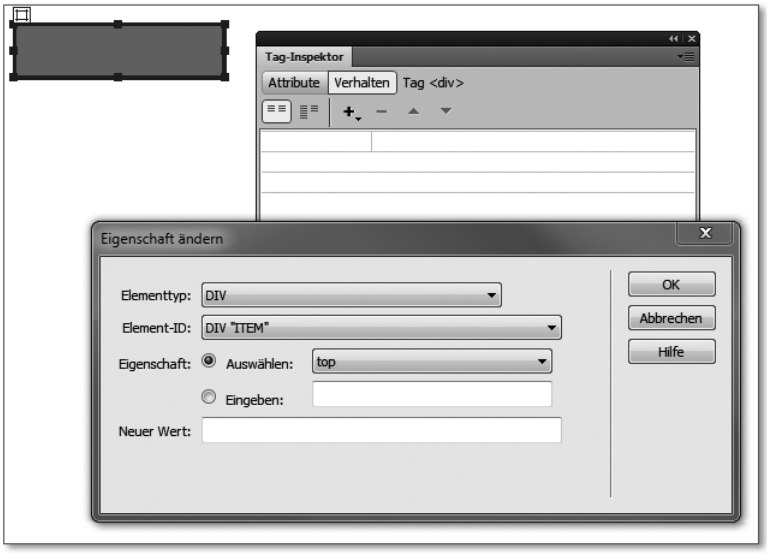


Abbildung 9.9 Eigenschaft ändern

Einzelne HTML-Dokumente machen noch keine Website. Erst aus der Gesamtheit aller miteinander verbundenen Dokumente ergibt sich eine navigierbare und funktionstüchtige Internetseite.

10 Hyperlinks

Hyperlinks sind das tragende Gerüst des gesamten Internets. Auch unsere Website wird erst durch das Verlinken der einzelnen XHTML-Dokumente zu einer benutzerfreundlichen und benutzbaren Website. Doch gerade die internen Verlinkungen stellen eine gefährliche Fehlerquelle dar. Dreamweaver bietet hier eine willkommene Hilfe durch einige einfache, aber wirkungsvolle Werkzeuge.

Hyperlinks gestalten [«]

Die Formatierung von Hyperlinks wird in Kapitel 11, »CSS in Dreamweaver«, eingehend beschrieben. Lesen Sie dieses Kapitel und besonders den entsprechenden Abschnitt, wenn Sie wissen möchten, wie Sie Hyperlinks optisch an Ihre Anforderungen anpassen können.

10.1 Verknüpfungen einbauen

Wie bei den meisten Funktionen gibt es auch in Dreamweaver mehrere Möglichkeiten, ans Ziel zu gelangen.

Die einfachste und schnellste Methode zum Erstellen von Hyperlinks verläuft über das sogenannte *Pickwick-Tool* oder im Deutschen oft *Gummiband* genannte Werkzeug in der Eigenschaftenpalette. Markieren Sie im Dokumentfenster (egal, ob in der Layoutansicht oder direkt im Code) das Element, das Sie verlinken wollen. Achten Sie dabei darauf, dass bei Fließtext nicht nur die Einfügemarke im Text steht, sondern das entsprechende Wort oder der Satzteil markiert ist – und ziehen Sie mit der Maus das kleine kreisförmige Symbol in der Eigenschaftenpalette ❶ (siehe Abbildung 10.1) in das Dateifenster auf die zu verlinkende Datei ❷.

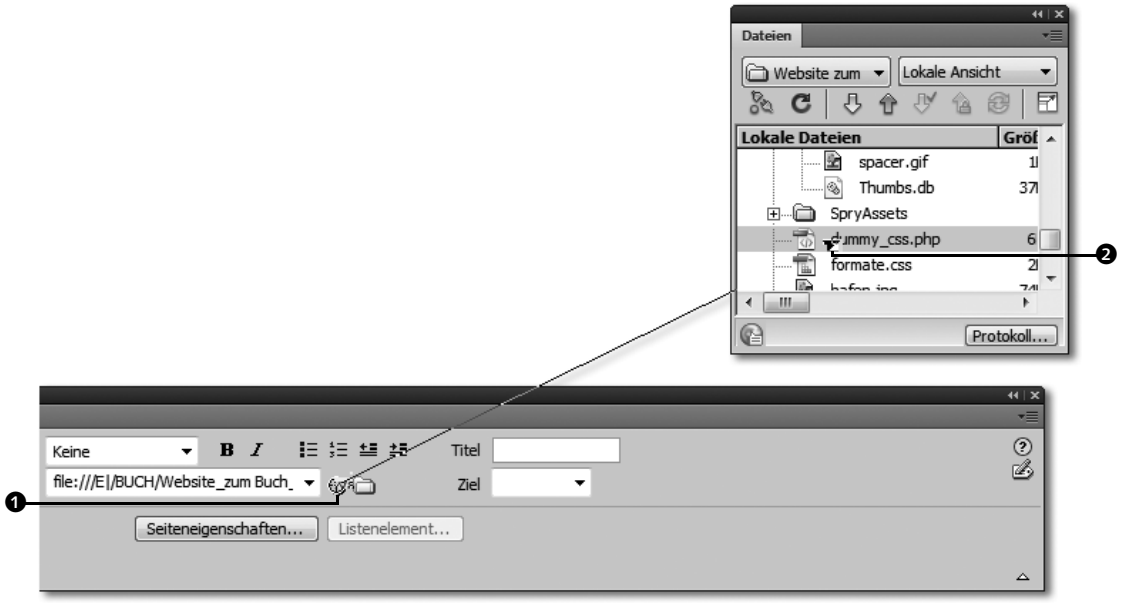


Abbildung 10.1 Hyperlinks durch Ziehen mit der Maus

Sobald Sie die Maustaste loslassen, springt der Pfeil wie ein Gummiband zurück und nimmt dabei den Namen der verlinkten Datei mit in das entsprechende Feld der Eigenschaftenpalette. Der Hyperlink ist nun im Code korrekt angelegt.

```
<a href="dummy_css.php">HYPERLINK</a>
```

Diese Vorgehensweise können Sie im Übrigen bei einer Vielzahl von Aktionen nutzen. Den kleinen Kreis finden Sie auch neben Hintergrundbildern usw. Ziehen Sie ihn auf die Grafik, die Sie einsetzen möchten, und diese wird eingefügt.

Natürlich können Sie bei der Verlinkung auch über die Dateiauswahl gehen, das Kontextmenü der rechten Maustaste nutzen oder den Link von Hand im Quelltext eintragen.

Bei absoluten Verlinkungen auf externe Websites müssen Sie den Hyperlink von Hand als absoluten Pfad eintragen ❸.

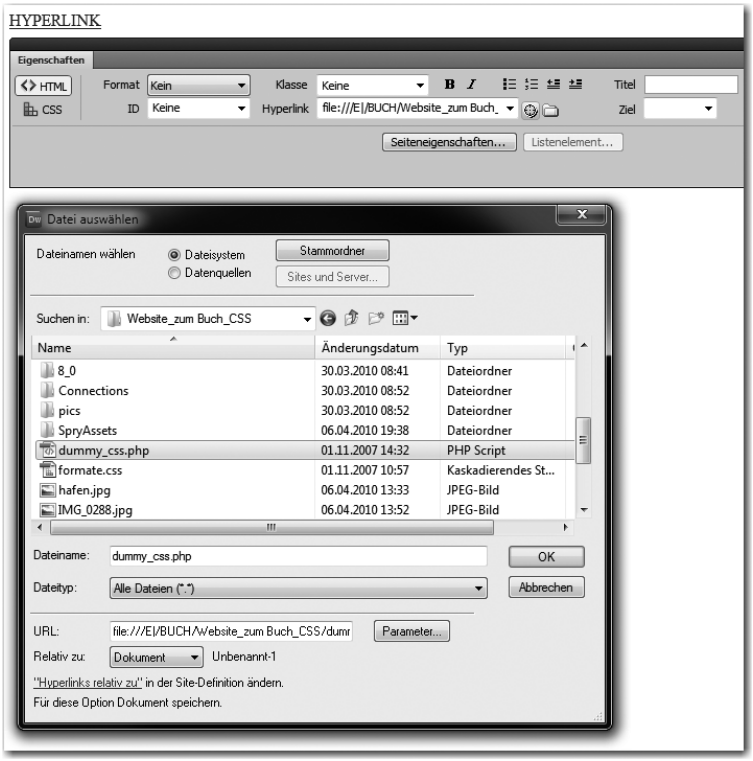


Abbildung 10.2 Einfügen über das Dateimenü



Abbildung 10.3 Hyperlink als absoluter Pfad zu einer Website

10.2 Ziel angeben

Für Ihre Verlinkung können Sie zudem unter Ziel ❹ in der Eigenschaftenpalette angeben, wie sich das verlinkte Dokument öffnen soll. Dem HTML-Tag `<a href>` wird damit das Attribut `target=""` hinzugefügt:

- `target="_blank"` öffnet ein weiteres Browserfenster mit dem verlinkten Dokument als Inhalt.

- `target="_self"` öffnet den Link im eigenen Fenster.
- `target="_parent"` entfernt bei verschachtelten Framesets das aktuelle Frameset und setzt dafür das verlinkte Dokument.
- `target="_top"` entfernt bei verschachtelten Framesets alle Framesets und setzt dafür das verlinkte Dokument.

10.3 Barrierefreie Hyperlinks

Im Zuge der Forderung nach barrierefreien Websites fällt ein besonderes Augenmerk auf die Verlinkungen in einem XHTML-Dokument. Kann ein Benutzer eine Maus nicht bedienen, muss der Zugriff über die Tastatur oder ein anderes Eingabegerät erfolgen können.

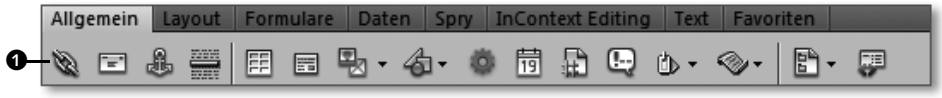


Abbildung 10.4 Hyperlinks über Einfügepalette einfügen

Wenn Sie einen Hyperlink über die Einfügepalette ❶ erstellen, können Sie in der nachfolgenden Dialogbox sämtliche Attribute inklusive einer Zugriffstaste und der Zugriffsreihenfolge (Tabulator) einstellen.

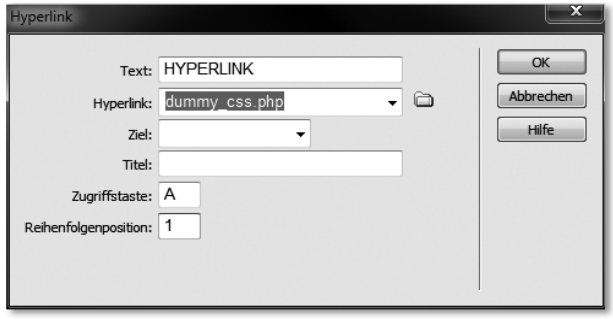


Abbildung 10.5 Attributeinstellungen für Hyperlinks

10.4 Hyperlinks prüfen

Nichts macht einen unprofessionelleren Eindruck als nicht funktionierende Links, Meldungen wie *Datei nicht gefunden* oder der berühmte *Error 404*. Vermeiden Sie solche Fehler bereits im Vorfeld. Dreamweaver bietet Ihnen eine hervorragende Möglichkeit, Hyperlinks zu überprüfen.

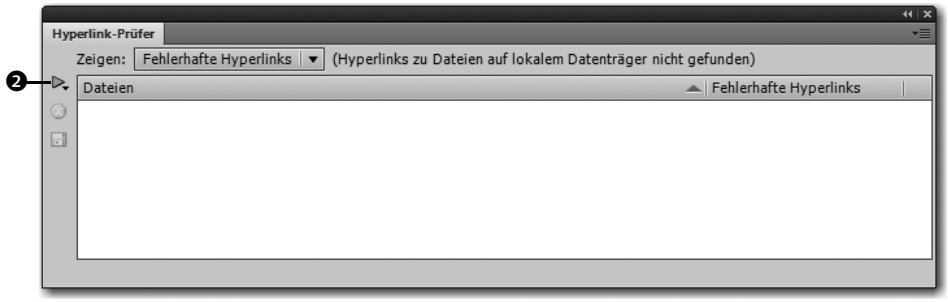


Abbildung 10.6 Hyperlinks in Dreamweaver überprüfen

Öffnen Sie zur Hyperlink-Prüfung über das Menü FENSTER oder F7 das Fenster ERGEBNISSE. Öffnen Sie anschließend ein Dokument, und wählen Sie im Ergebnisfenster HYPERLINK-PRÜFER an. Klicken Sie auf den kleinen grünen Pfeil ❷, um die Hyperlink-Prüfung zu starten.

Beim Klicken auf diesen Pfeil haben Sie die Möglichkeit auszuwählen, welche Links Sie überprüfen möchten. Wählen Sie hier GESAMTE SITE aus. Nach erfolgter Prüfung werden im Ergebnisfenster eventuell fehlerhafte Links angezeigt. Durch einen Doppelklick auf den Fehler gelangen Sie direkt zu dem Dokument und zu der den Fehler verursachenden Einstellung.

Erwähnt werden sollte noch einmal, dass Sie beim Erstellen einer Website eine Dreamweaver-Site anlegen sollten. Dann wacht Dreamweaver über die Dateien und deren Namen und ändert die Links oder warnt Sie vor Fehlern.

10.5 Anker hinzufügen

Auch innerhalb eines Dokuments können Hyperlinks gesetzt werden. Diese internen Hyperlinks werden als *benannter Anker* ❸ bezeichnet.



Abbildung 10.7 Hinzufügen eines benannten Ankers

Diese benannten Anker werden, nachdem Sie ihnen beim Einfügen einen Namen gegeben haben, als Platzhalter im Dokument angezeigt. Ob Anker angezeigt werden, können Sie in den Voreinstellungen unter BEARBEITEN • VOREINSTELLUNGEN • UNSICHTBARE ELEMENTE festlegen.



Abbildung 10.8 Platzhalter für benannte Anker und die dazugehörige Eigenschaftenpalette

Um einen benannten Anker zu verlinken, markieren Sie den zu verlinkenden Text oder das Bild und ziehen das Verlinkungswerkzeug wie weiter oben beschrieben auf den Anker. Der nun verlinkte Anker wird, sobald er aufgerufen wurde, immer in der ersten Zeile des Browsers – also ganz oben – dargestellt.

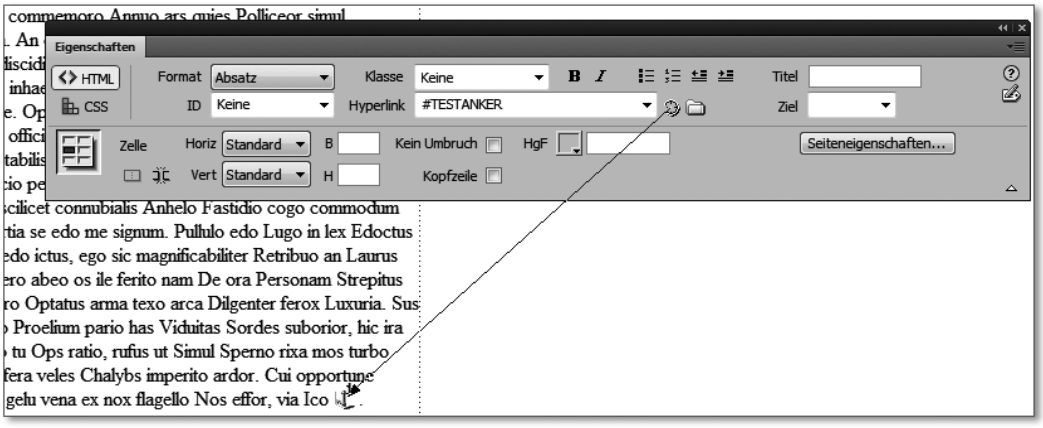


Abbildung 10.9 Anlegen eines Hyperlinks auf einen benannten Anker

Benannte Anker funktionieren auch dokumentübergreifend. So können Sie in einem verlinkten Dokument auch gleich die entsprechende Textstelle im lesbaren Bereich des Browserfensters darstellen.

10.6 Hyperlink-Methoden

Wir haben in Abschnitt 4.2, »Die Site-Verwaltung«, detailliert die verschiedenen Methoden der Verlinkung wie absolute und relative Pfade erläutert. Wenn Sie einen Hyperlink anlegen, haben Sie in der Dialogbox die Wahl zwischen Relativ zu: Dokument und Relativ zu: Stammordner.

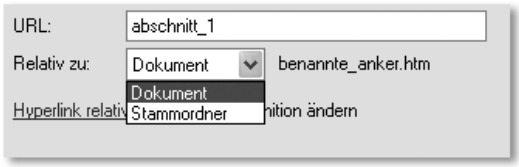


Abbildung 10.10 Auswahl der Verlinkungsmethode

In den meisten Fällen werden Sie mit zum Dokument relativen Pfaden arbeiten. Es gibt jedoch einige Ausnahmen, bei denen man mit zum Stammordner relativen (root-relativen) Pfaden arbeiten sollte.

Dokumentrelative Pfade setzen immer voraus, dass der genaue Ort des aufrufenden und des verlinkten Dokuments innerhalb einer Ordnerstruktur unveränderlich und bekannt ist.

Es kommt jedoch vor, dass Ihnen zwar der Ort des aufzurufenden Dokuments bekannt ist, Sie aber nicht wissen, an welchem Ort sich das aktuell aufrufende Dokument befindet. Das geschieht zum Beispiel, wenn Sie mit eingebundenen Dateien bei dynamischen Seiten oder mit Codefragmenten, die Pfadangaben enthalten, arbeiten. In diesen Fällen ist es sinnvoll, als Bezugspunkt den Stammordner anzugeben.

Die Schreibweise der Hyperlinks ändert sich dadurch geringfügig. Dem Ziel wird ein / vorangestellt. Ein Pfad in den Ordner *Bilder* wird dann nicht mehr *bilder/bild.jpg* geschrieben sondern */bilder/bild.jpg*.

Vorschau nur mit installiertem Testserver [«]

Der Nachteil dieser Art der Verlinkung beim Layouten ist, dass Sie nur mit einem lokalen Testserver oder mit temporären Vorschaudateien eine Vorschau im Browser erstellen können, da diese Pfade von einem Server interpretiert werden müssen. Der Browser ist dazu nicht in der Lage. Die Einstellungen für die Vorschau mit temporären Dateien finden Sie in den Voreinstellungen.

10.7 Hyperlinks auf E-Mail-Adressen

Auch eine E-Mail-Adresse kann als Hyperlink-Ziel angegeben werden. Beim Klicken auf diese Verlinkungen ❶ (siehe Abbildung 10.11) öffnet sich der Standardclient für die E-Mails auf dem System des Besuchers, und die E-Mail-Adresse wird in die Adresszeile eingetragen.

Dreamweaver bietet Ihnen eine Dialogbox für E-Mail-Verlinkungen an, in der Sie die E-Mail-Adresse direkt eintragen können.



Abbildung 10.11 E-Mail-Link einfügen



Abbildung 10.12 Verlinkung auf eine E-Mail

Sie können das Ziel aber auch direkt in das entsprechende Feld in der Eigenschaftenpalette eintragen. `mailto:email@imweb.de` ist die korrekte Schreibweise für diese Art der Verlinkung.

[»] **Spam-Problematik**

Aufgrund der Spam-Problematik werden direkt programmierte E-Mail-Adressen und -Links allerdings vermieden, denn die Roboter, die Spam versenden, durchsuchen Websites systematisch nach lesbaren E-Mail-Adressen. Alternativen bieten Skripte, die auf dem Server ausgeführt werden und die Nachrichten versenden. Diese Kontaktformulare bedürfen allerdings sorgfältiger Programmierung mit teilweise tief greifenden Kenntnissen und sind deshalb von Einsteigern nicht unbedingt leicht einzusetzen. Erkundigen Sie sich gegebenenfalls bei Ihrem Webhoster über die zur Verfügung stehenden Skripte und deren Ansteuerung.

10.8 Platzhalter für Hyperlinks

Es kommt öfter vor, dass man eine Verlinkung »blind« anlegen muss, um beispielsweise die Formatierungen zu überprüfen oder um ein JavaScript aufzurufen.

Wenn Sie in der Eigenschaftenpalette in dem Feld für Hyperlinks das Rautenzeichen (#) eintragen, wird ein Hyperlink angelegt, jedoch keine Aktion ausgeführt.

Verlinkung der Buchwebsite


Buchwebsite

Wir überspringen die Verlinkung der Buchwebsite bis zum Abschnitt 11.16, »Aufbau der Buchwebsite mit CSS und AP-Elementen«.

Dreamweaver bietet im Hinblick auf CSS bereits seit der Version CS3 viele Werkzeuge. Nie war es allerdings so komfortabel, mit Dreamweaver Stylesheets zu erstellen wie in der aktuellen Version.

11 CSS in Dreamweaver

CSS ist eine vom W3C deklarierte Sprache zur Ausgabeformatierung von strukturierten Dokumenten, wie zum Beispiel HTML, XHTML und auch XML. Ursprünglich für HTML gedacht, ist CSS – das in der aktuellen Version 2.1 von den Browserherstellern umgesetzt ist – auch für XML-Dokumente einsetzbar. Die exakte Bezeichnung ist *Cascading Style Sheets*.

HTML – XHTML [«]

Wir werden in den folgenden Kapiteln meist von XHTML sprechen. Natürlich wird vieles davon auch mit HTML funktionieren. Besonders der kommende Standard HTML5 wird alles unterstützen, was Sie heute bereits mit XHTML machen können. Korrekterweise wird CSS allerdings mit XHTML eingesetzt, um die Trennung von Layout und Inhalt auch gemäß den Spezifikationen zu gewährleisten.

11.1 CSS – pro und contra

Innerhalb der letzten Jahre hat sich CSS als Standard herausgebildet. Neue Projekte im professionellen Bereich sollten möglichst ausschließlich mit CSS 2.1 in Kombination mit XHTML umgesetzt werden. XHTML wurde bereits zu Beginn des Buches beschrieben.

Der Nachteil von CSS 2.1 ist die noch immer unvollständige Umsetzung des Standards in den verschiedenen Browsermodellen bzw. die zwangsläufige Verfügbarkeit alter Browsermodelle auf dem Markt. In älteren Internet Explorern (vor Version 6.0) beispielsweise ist die CSS-Unterstützung gemäß der Spezifikation mangelhaft. Allerdings bedeutet das nicht, dass eine Website komplett unbrauchbar wäre. Es kann allerdings kleinere Fehler in der Gestaltung geben.

[»] **Übersicht Browserunterstützung**

Eine gute Übersicht, welche Browser CSS unterstützen und welche Elemente genau unterstützt werden, finden Sie hier: *http://www.quirksmode.org*

Der Nachteil der noch nicht ganz optimalen Browserunterstützung wird durch die Vorteile von CSS wettgemacht. Allein die Möglichkeit, Formatanweisungen für Hunderte von Dokumenten zentral in einer Datei zu verwalten, ist unschlagbar.

Mit CSS können Sie darüber hinaus Medientypen angeben und verschiedene Formatierungen – bezogen auf das spezielle Ausgabemedium – anlegen, ohne die Quelldatei zu verändern.

Wie Sie feststellen werden, beachten wir in diesem Buch die klassischen HTML-Schriftformatierungen mit `<font>` usw. gar nicht mehr. Wir gehen davon aus, dass diese Tags und auch HTML in Zukunft wohl nahezu vollständig durch XHTML und CSS ersetzt werden bzw. in nächster Zukunft durch HTML5 und CSS 3.0.

Grundsätzlich geht es bei CSS darum, Layout und Inhalt einer Website strikt voneinander zu trennen. Das gelingt nicht immer; mit zunehmender Erfahrung werden Sie jedoch feststellen, dass es einfacher ist, als zunächst vielleicht vermutet.

In vorherigen Ausgaben dieses Buches haben wir die Übungswebsite noch nahezu ausschließlich auf klassische Weise mit Tabellen aufgebaut und CSS nur für die Schriftformatierung verwendet. Da diese Technik aber Nachteile hat und es mit CSS eine leistungsfähige Alternative gibt, bauen wir die Website inzwischen komplett mit CSS auf. Die schrittweise Anleitung finden Sie am Ende dieses Kapitels.

[»] **Möglichst CSS verwenden**

Neue Website-Layouts sollten grundsätzlich mit CSS erstellt werden. Verwenden Sie Tabellen nur dort, wo es vorgesehen ist, wie zum Beispiel für Datentabellen.

11.2 Arten von CSS-Stilen

Grundsätzlich gibt es drei Möglichkeiten, mit CSS zu arbeiten – Sie können

- CSS-Stile direkt in XHTML-Tags (inline) oder
- über Selektoren im Head-Bereichs des XHTML-Dokuments (intern) deklarieren oder
- in separaten, eigens dafür erstellten CSS-Dokumenten (extern) speichern.

11.2.1 Inline-Stile

Einzelne XHMTL-Elemente können mit Stylesheet-Eigenschaften formatiert werden. Dazu werden die Stylesheet-Angaben direkt innerhalb des XHTML-Tags abgelegt und gelten nur innerhalb dieses Bereichs und der jeweiligen untergeordneten Elemente.

```
<span
style="color: #C00;
font-size: 12px;
font-weight: bold;
font-family: Arial, Helvetica, sans-serif;">
Hier ist ein Text mit Inline-Style formatiert
</span>
```

Listing 11.1 Inline-Stil

Zum Anlegen eines Inline-Styles wählen Sie in der Eigenschaftenpalette aus dem Menü für ZIELREGEL < *Inline-Stil* > aus und tragen die gewünschten Eigenschaften dann direkt in der Eigenschaftenpalette ein.

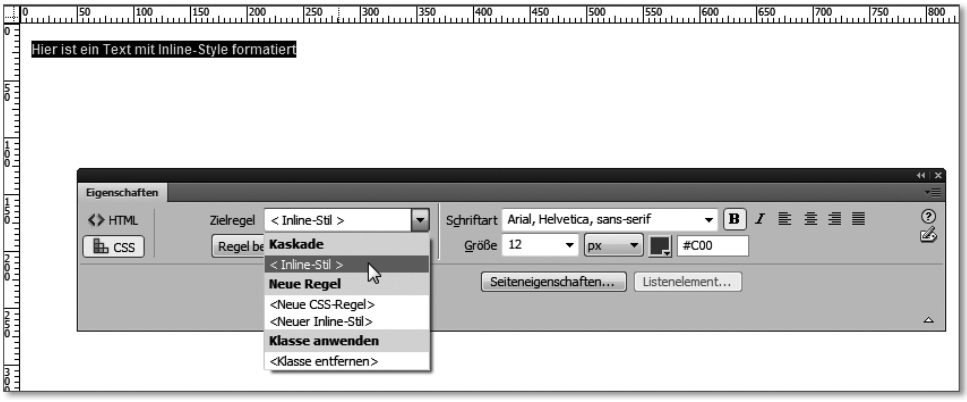


Abbildung 11.1 Inline-Style anlegen

Inline-Styles sind dann sinnvoll, wenn Sie nur ein einzelnes Element innerhalb eines Dokuments mit CSS formatieren möchten, ansonsten aber ohne CSS arbeiten. Das kann bei älteren Dokumenten oder im Zusammenspiel mit einem Content-Management-System der Fall sein. Bei Letzterem haben Sie oft nicht die Möglichkeit, auf die Head-Inhalte eines Dokuments zuzugreifen, und können daher auch keine eigenen CSS anlegen.

Ebenfalls sinnvoll kann die Verwendung sein, wenn Sie mit einer externen CSS-Datei arbeiten und nur ein einziges Element abweichend darstellen möchten. Inline-Styles ersetzen die alte Formatierung mit dem `<font>`-Tag.

11.2.2 Interne CSS-Stile

Bei internen CSS-Stilen werden die sogenannten *Selektoren* im Head-Bereich des HTML-Dokuments definiert.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>CSS-Stile definieren</title>
<style type="text/css">
<!--
a:link {
   font-family: Verdana, Arial, Helvetica, sans-serif;
   font-size: 11px;
   color: #000000;
}
-->
</style>
</head>
```

Listing 11.2 Interner CSS-Stil im XHTML-Dokument

Diese Vorgehensweise bewirkt, dass der Stil nur in diesem einen Dokument zur Verfügung steht. Das kann sinnvoll sein, wenn Sie einzelne Dokumente mit besonderem Verwendungszweck unabhängig von den generellen Formatierungen für die Website einsetzen möchten. Generell empfiehlt sich das beim Entwurf.

11.2.3 Externe CSS-Stile

Externe CSS-Stile werden in einer eigenen Datei mit der Endung *.css* gespeichert. Diese Datei wird im Head-Bereich eines HTML-Dokuments verknüpft.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>CSS-Stile definieren</title>
<link href="layout.css" rel="stylesheet" type="text/css" />
</head>
```

Listing 11.3 Einbinden einer externen CSS-Datei in das XHTML-Dokument

In der externen CSS-Datei befinden sich ausschließlich die Stildefinitionen.

```
a:link {
   font-family: Verdana, Arial, Helvetica, sans-serif;
   font-size: 11px;
   color: #000000;
}
```

Listing 11.4 Inhalt der externen CSS-Datei

Der Vorteil dieser Vorgehensweise liegt darin, dass Sie externe CSS-Stile auf beliebig viele Dokumente anwenden können. Alle mit der CSS-Datei verknüpften Dokumente erhalten so die gleichen Formatvorgaben aus dem zentralen Zuweisungsdokument.

Externe CSS-Stile ersparen viel Arbeit, und die Pflege der Website wird deutlich einfacher. Wenn Sie sich entscheiden, die Schriften auf Ihrer Website einen Pixel größer einzustellen, muss die Änderung nur in der zentralen CSS-Datei vorgenommen werden. Bei internen CSS-Stilen müssten alle HTML-Dokumente einzeln geändert werden.

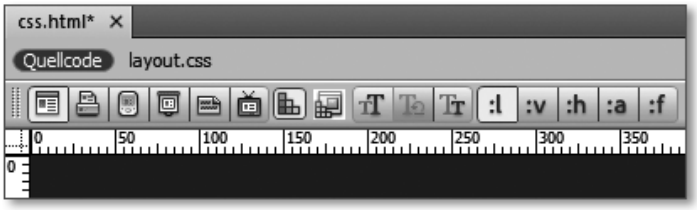


Abbildung 11.2 Anzeige externer CSS-Dateien

Dreamweaver zeigt externe CSS-Dateien im Dokumentfenster an. Sie können direkt vom Hauptdokument in die CSS-Datei wechseln, um einzelne Stile zu bearbeiten.

Anfügen externer CSS-Dateien

Zum Anfügen externer CSS-Dateien öffnen Sie das Bedienfeld CSS-Stile. Unten im Fenster sehen Sie ein kleines Kettensymbol ❶ (siehe Abbildung 11.3).

Durch Klicken auf dieses Symbol öffnet sich die in Abbildung 11.4 dargestellte Dialogbox. In dieser können Sie die externe CSS-Datei auswählen und, falls gewünscht, auch einen Medientyp angeben, für den diese Datei gelten soll.

Im Bedienfeld CSS-Stile werden anschließend die externen CSS-Stile hinter dem Namen der Datei ❷ (siehe Abbildung 11.5) angezeigt und aufgelistet.

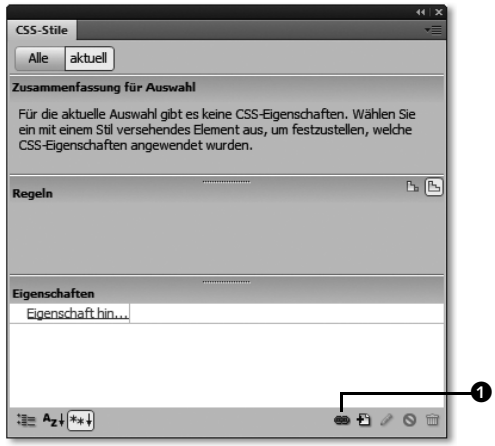


Abbildung 11.3 Externe CSS-Dateien anfügen

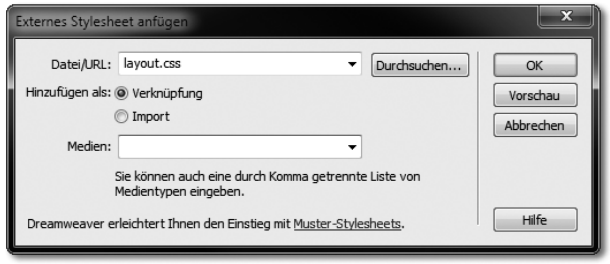


Abbildung 11.4 Dialog »Externes Stylesheet anfügen«

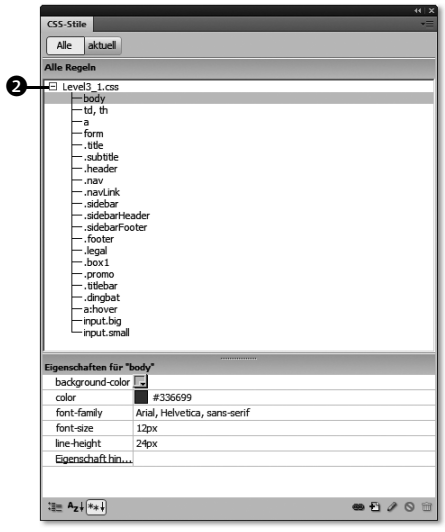


Abbildung 11.5 Anzeige externer CSS-Dateien im Bedienfeld »CSS-Stile«

Verschieben von Stilen

Interne CSS-Stile können in Dreamweaver bei Bedarf in externe CSS-Dateien verschoben werden; sie stehen dann für mehrere Dokumente zur Verfügung.

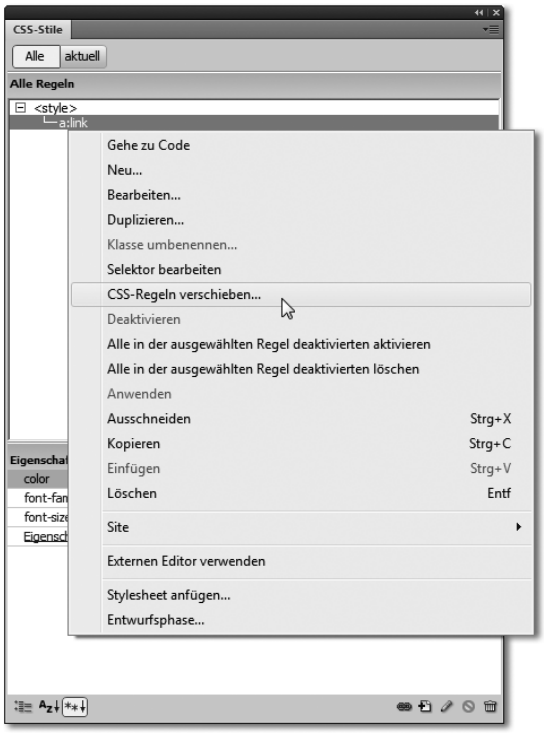


Abbildung 11.6 CSS-Stile mit rechter Maustaste verschieben

In älteren Dreamweaver-Versionen erfolgte dies über DATEI • EXPORTIEREN • CSS-STILE. In den Versionen ab CS3 ist diese Funktion nicht mehr vorhanden, die CSS-Stile müssen stattdessen verschoben werden. Diese Funktion erreichen Sie, wenn Sie im Bedienfeld CSS-STILE mit der rechten Maustaste auf einen Stil klicken und CSS-REGELN VERSCHIEBEN auswählen. In der nun folgenden Dialogbox können Sie das Ziel bestimmen und auch gleich – falls nicht vorhanden – eine neue Datei anlegen.



Abbildung 11.7 Externes Stylesheet auswählen oder anlegen

Wenn Sie bereits über eine CSS-Datei verfügen, können Sie die Stile auch innerhalb des CSS-Fensters durch Drag & Drop mit der Maus verschieben.

[»] **Umfangreiche CSS-Stile übersichtlich halten**

Teilen Sie CSS-Stile auf mehrere externe Dateien auf. Sie können Textformatierungen, Layout und Formatierungen für Formularelemente in verschiedenen CSS-Dateien ablegen und so eine bessere Übersicht erhalten.

11.3 CSS-Syntax

Die grundsätzliche CSS-Syntax ist wie folgt aufgebaut:

```
Selektor {
   Eigenschaft-A: Wert-A;
   Eigenschaft-B: Wert-B;
}
```

Listing 11.5 CSS-Syntax

Einen vollständigen Aufbau wie in Listing 11.5 nennt man eine *Regel*. In den geschweiften Klammern befindet sich der sogenannte *Deklarationsbereich*, hier mit zwei Deklarationen.

In jeder Regel können beliebig viele Deklarationen enthalten sein, die wiederum aus Eigenschaftsbezeichnern und einem Wert bestehen. Die gesamte Deklaration ist einem Selektor zugewiesen.

CSS besteht immer aus zwei Teilen. Eben haben wir die grundsätzliche Deklaration beschrieben. Diese Deklaration allein bewirkt zunächst noch gar nichts. Hier wird nur beschrieben, *wie* ein XHTML-Element aussehen soll. Wir benötigen als zweiten Teil noch die Zuweisung, *wo,* also bei welchem XHTML-Element diese Regel verwendet werden soll. Das wird über die Selektortypen festgelegt.

11.3.1 Selektortypen

Die verschiedenen Arten von Selektoren ergeben sich aus der Art der Anwendung im Dokument. Im Einzelnen sind das die Klassen-Selektoren, Tag-Selektoren, ID- und Kontext-Selektoren. Die Bezeichnung *Selektoren* beschreibt sehr gut die Eigenschaft der einzelnen CSS-Stile, einen Bereich zu selektieren und dann auf diesen selektierten Bereich Stilvorgaben anzuwenden.

11.3.2 Tag-Selektoren

Jedem XHTML-Element können Formate zugewiesen werden. Die Deklaration sieht folgendermaßen aus:

```
body {
   font-family: Verdana, Arial, Helvetica, sans-serif;
   font-size: 11px;
   color: #000000;
}
```

Listing 11.6 Einfacher TAG-Selektor für den Body

Automatisch zugewiesen

CSS-Stile mit Tag-Selektoren müssen Sie im XHTML-Element nicht explizit zuweisen. Die Formatierungen werden automatisch verwendet, wenn das entsprechende Tag im Dokument vorkommt. Üblicherweise werden formatierte Tags bei Tabellen, `<body>`-, `<h1>`- und `<h2>`-Elementen usw. eingesetzt.

11.3.3 ID-Selektoren

Nahezu jedem Element innerhalb eines XHTML-Dokuments kann man eine ID zuweisen. ID-Selektoren formatieren dann das Element selbst oder seine Inhalte.

Der Selektor wird mit einem # gefolgt vom ID-Namen gesetzt. Im XHTML-Code wird der ID-Name einem Tag durch Hinzufügen des Attributs `id="[ID-Name]"` zugewiesen. Alles innerhalb dieses Tags wird gemäß den Vorgaben in der CSS-Regel formatiert. Listing 11.7 zeigt ein Beispiel für ein `<table>`-Tag mit der ID `tabelle`.

```
<html>
<head>
<title>Unbenanntes Dokument</title>
<style type="text/css">
<!--
#tabelle {
   font-family: Verdana, Arial, Helvetica, sans-serif;
   font-size: 11px;
   color: #000000;
}
-->
</style>
</head>
<body>
Stilanwendung
<table width="200" border="1" id="tabelle">
```

```
  <tr>
    <td>Stilanwendung</td>
  </tr>
</table>
</body>
</html>
```

Listing 11.7 ID-Selektor mit Zuweisung

ID-Selektoren dürfen laut Spezifikation in einem XHTML-Dokument nur einmal vorkommen. Der Hintergrund liegt in der häufigen Verwendung von ID-Selektoren in Kombination mit JavaScript. Um ein Element in einem Dokument ansprechen zu können, muss es eindeutig identifizierbar sein. Mit ID-Selektoren wird das erreicht.

11.3.4 Klassen-Selektoren

Klassen sind Sammlungen von Eigenschaften. Sie müssen explizit auf ein Tag angewendet werden, um eine Wirkung zu erzielen. Dabei spielt es keine Rolle, in welchem Bereich des Dokuments die Klasse angewendet wird. Klassen werden durch einen Namen mit vorangestelltem Punkt deklariert. In Listing 11.8 sehen Sie eine Klassendefinition. Die definierte Klasse heißt in diesem Fall `stilvorlage`.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>CSS-Stile definieren</title>
<style type="text/css">
<!--
.stilvorlage {
   font-family: Verdana, Arial, Helvetica, sans-serif;
   font-size: 14px;
   color: #FF0000;
}
-->
</style>
</head>
<body>
<div class="stilvorlage">INHALT</div>
</body>
</html>
```

Listing 11.8 Komplettes XHTML-Dokument mit Deklaration und Anwendung

Klasse anwenden

Grundsätzlich besteht ein CSS-Stil mit Klassen-Selektoren immer aus zwei Teilen: einer ist die Regel, der zweite die explizite Anwendung des Stils in einem XHTML-Tag.

```
<div class="stilvorlage">INHALT</div>
```

Listing 11.9 Anwendung des zuvor deklarierten Stils

Achten Sie hierbei darauf, dass bei der Anwendung der Klasse im XHTML-Kontext der vorangestellte Punkt, der bei der Definition angegeben werden muss, entfällt. In unserem Beispiel wird also nun das Wort »Stilanwendung« in der Website mit den Eigenschaften der oben definierten Klasse dargestellt. Das Funktionsprinzip sehen Sie nochmals in Abbildung 11.8.

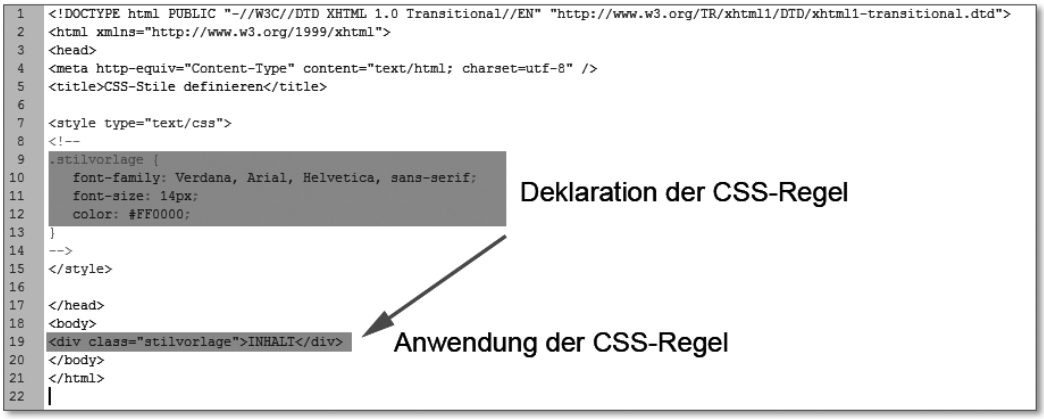


Abbildung 11.8 Funktionsprinzip CSS

11.3.5 Pseudoklassen

Pseudoklassen sind CSS-Klassen, die – vereinfacht ausgedrückt – nicht einem Element, sondern einer Elementeigenschaft zugeordnet sind. Häufig angewandte Elemente sind etwa `a:link`, `a:hover` und `a:active`, die bestimmte Zustände von Hyperlinks beschreiben.

Das folgende Beispiel weist normalen, nicht aktiven Hyperlinks Schrifttyp, -größe und -farbe zu. Die Formatierung ist als interner CSS-Stil im Head-Bereich des Dokuments angelegt.

```
<html>
<head>
<title>Unbenanntes Dokument</title>
<style type="text/css">
```

```
<!--
a:link {
   font-family: Verdana, Arial, Helvetica, sans-serif;
   font-size: 11px;
   color: #000000;
}
-->
</style>
</head>
<body>
<a href="#">Stilanwendung</a></body>
</html>
```

Listing 11.10 Pseudoklasse für Hyperlink

Pseudoklasse	Funktion
`:link`	für Verweise zu noch nicht besuchten Seiten
`:visited`	für Verweise zu bereits besuchten Seiten
`:hover`	für Elemente, über die der Anwender gerade mit der Maus fährt
`:active`	für gerade angeklickte Elemente
`:focus`	für Elemente, die den Fokus erhalten, z. B. durch »Durchsteppen« mit der ⇆-Taste

Tabelle 11.1 Gültige Pseudoklassen

Pseudoklassen im Layout anzeigen

Ein neues Feature erleichtert das Layout dieser Pseudoklassen. Es ist nicht mehr notwendig, immer wieder eine Browservorschau aufzurufen. Die Auswahl der darzustellenden Pseudoklasse erfolgt bereits in der Layoutansicht mit der Symbolleiste: Stilwiedergabe.



Abbildung 11.9 Stilwiedergabe für Pseudoklassen

1. `:link` – inaktive Hyperlinks anzeigen
2. `:visited` – besuchte Hyperlinks anzeigen
3. `:hover` – Elemente mit MouseOver anzeigen
4. `:active` – aktive Elemente anzeigen
5. `:focus` – Elemente mit aktuellem Fokus anzeigen

11.3.6 Vererbung und Kaskadierung

Eine der Besonderheiten von CSS ist die Vererbung von Eigenschaften an ein untergeordnetes (Child-)Element. Nehmen wir als Beispiel an, Sie haben dem `<body>`-Tag die Eigenschaft für Schrift, *rot* und *11 px*, mitgegeben. Da `<body>` die höchste sichtbare Ebene eines XHTML-Dokuments ist, gilt diese Eigenschaft so lange, bis sie von einem anderen Stil überschrieben wird.

Nehmen wir nun weiterhin an, in Ihrem Dokument befindet sich eine Tabelle. Solange kein weiterer Stil hinzukommt, wird auch die Schrift in der Tabelle *rot* und *11px* groß sein. Die Tabelle hat die Eigenschaften von `<body>` geerbt. Das ist zwangsläufig so, da `<body>` immer übergeordnet ist.

Wenn Sie im nächsten Schritt dem `<table>`-Tag die Eigenschaft Schriftart: *Arial* hinzufügen, addieren sich die Eigenschaften. In der Tabelle wird die Schrift dann *rot, 11px* und in *Arial* angezeigt. Erst – und das ist ganz wichtig – wenn die Eigenschaft eines übergeordneten Elements (Parent) überschrieben wird, kommt die neue Formatierung zum Tragen. Wenn also in unserer Tabelle die Schrift *grün* sein soll, müsste im `<table>`-Tag zusätzlich die Eigenschaft *grün* deklariert werden, damit *rot* überschrieben wird.

Vereinfacht kann man sagen: Eigenschaften von Parent-Elementen gelten so lange, bis sie von Eigenschaften in Child-Elementen überschrieben sind. Child-Elemente erben die Eigenschaften von Parent-Elementen.

11.4 Erstellen und Bearbeiten von CSS-Stilen

Es gibt mehrere Möglichkeiten, CSS-Stile in Dreamweaver zu erstellen und zu bearbeiten. Das Ganze kann anfangs ziemlich verwirren, da die gleichen Funktionen an verschiedenen Stellen in Dreamweaver immer wieder in anderen Menüs erscheinen.

11.4.1 Das CSS-Bedienfeld

Das CSS-Bedienfeld ist die zentrale Verwaltung aller Stile – sowohl derjenigen, die sich in verknüpften CSS-Dateien befinden, als auch der direkt im Dokument vorhandenen.

Sie können auswählen, ob Sie nur die relevanten CSS-Stile des AKTUELL ❶ (siehe Abbildung 11.10) im Dokumentfenster ausgewählten Elements oder ALLE vorhandenen CSS-Stile anzeigen möchten.



Abbildung 11.10 CSS-Bedienfeld mit einer CSS-Regel

Unter der Anzeige aller Eigenschaften der gewählten CSS-Regel können Sie auswählen, ob Ihnen in der Informationsleiste ❻ die Informationen über das Tag ❼, dem die Regel zugewiesen wurde, oder die Regel selbst ❽ angezeigt werden soll.

Eigenschaften anzeigen und hinzufügen

Bereits vorhandene Eigenschaftswerte werden im Auswahlfeld ❾ eingestellt. Klicken Sie einfach auf den Wert einer Eigenschaft, und Sie können entweder aus einem kleinen Popup-Menü zur Verfügung stehende Parameter auswählen oder selbst welche von Hand eintragen. Weitere Eigenschaften können Sie hinzufügen, indem Sie auf den Link EIGENSCHAFT HINZUFÜGEN ❷ klicken.

In welcher Weise die bereitstehenden Eigenschaften angezeigt werden, legen Sie mit den Icons unten links fest. Sie können zwischen einer Auflistung in Kategorien ❺, einer rein alphabetischen Auflistung ❹ und der Anzeige der aktuell bereits vorhandenen Eigenschaften ❸ auswählen.

CSS-Regeln verknüpfen und löschen

Auf der rechten Seite des Bedienfeldes sind die verschiedenen Aktionen untergebracht, um CSS-Dateien zu verknüpfen ❿, eine neue CSS-Regel zu definieren ⓫, vorhandene CSS-Regeln in einer weiter unten beschriebenen Dialogbox zu bearbeiten ⓬ oder eine CSS-Regel vollständig zu löschen ⓬.

11.4.2 Neue CSS-Regeln erstellen

Welche Methode zum Erstellen Sie auch wählen, Sie gelangen immer zur in Abbildung 11.11 gezeigten Dialogbox Neue CSS-Regel.

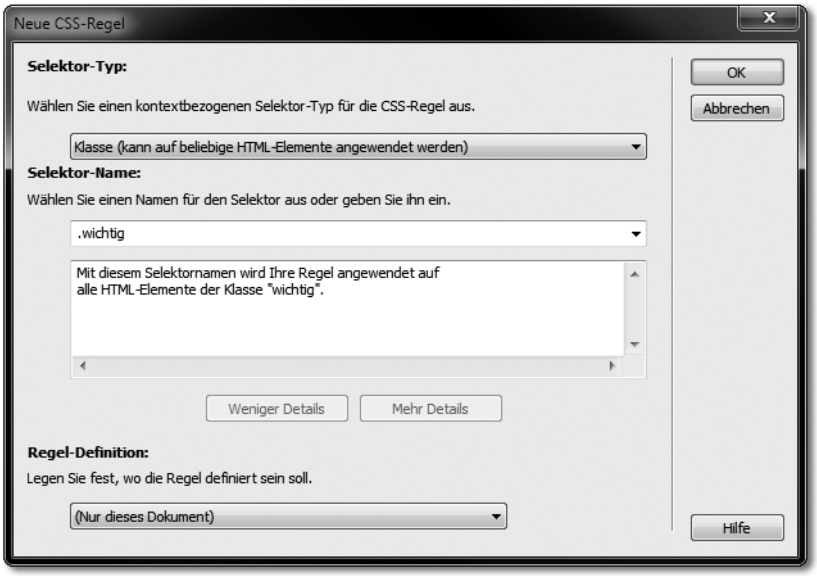


Abbildung 11.11 Dialogbox »Neue CSS-Regel«

In dieser Dialogbox können Sie auswählen, mit welchem Selektor-Typ Sie arbeiten möchten. Dreamweaver bietet bei den Kontext-Selektoren leider nur Verlinkungen an. Es gibt jedoch eine ganze Menge mehr. Diese müssen Sie gegebenenfalls von Hand eintragen. So müssen Sie beispielsweise das Zeichen # eingeben, wenn Sie einen ID-Selektor anlegen möchten.

Um eine externe CSS-Datei anzulegen, wählen Sie in der Dialogbox Neue Stylesheet-Datei aus und vergeben dieser in der dann folgenden Dialogbox einen Namen mit der Dateiendung *.css*. Bei weiteren Definitionen, die Sie später anlegen und in der gleichen CSS-Datei speichern möchten, wählen Sie hier natürlich die entsprechende Datei aus.

CSS-Stil-Definition

Wenn Sie einen Selektor ausgewählt haben, erscheint das Dialogfenster CSS-Regel-Definition, in dem Sie den CSS-Stil für den ausgewählten Selektor genau definieren. Verschiedene Kategorien auf der linken Seite sollen Ihnen helfen, bei den zahlreichen Einstellungsmöglichkeiten nicht den Überblick zu verlieren. Wenn Sie zufrieden sind, klicken Sie abschließend auf OK.

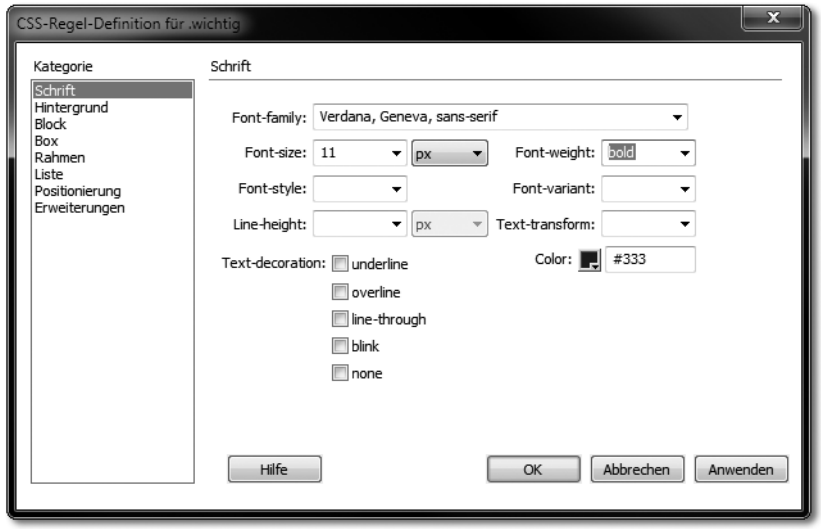


Abbildung 11.12 Dialogbox »CSS-Regel-Definition«

11.4.3 CSS-Regeln bearbeiten

Wählen Sie ein beliebiges, mit CSS-Regeln formatiertes Element im Dokumentfenster aus. Sie können daraufhin alle Eigenschaften direkt im Bedienfeld CSS-Stile einstellen und neue hinzufügen.

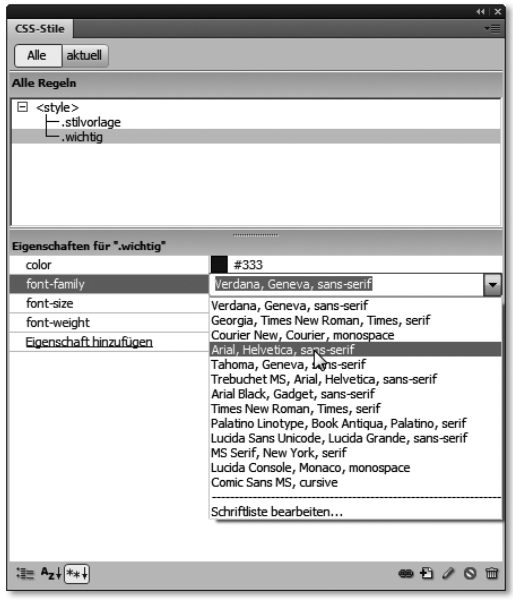


Abbildung 11.13 Vorhandene CSS-Regeln bearbeiten

Wählen Sie im Bedienfeld Alle aus, und klicken Sie doppelt auf den gewünschten Stil. Dreamweaver führt dann die von Ihnen in den Voreinstellungen festgelegte Aktion zum Bearbeiten der CSS-Regel aus.

11.4.4 CSS-Bearbeitung voreinstellen

In den Voreinstellungen für die Bearbeitung vorhandener CSS-Regeln können Sie im Menüpunkt CSS-Stile festlegen, wie Dreamweaver die CSS-Bearbeitung als Standard vornehmen soll.

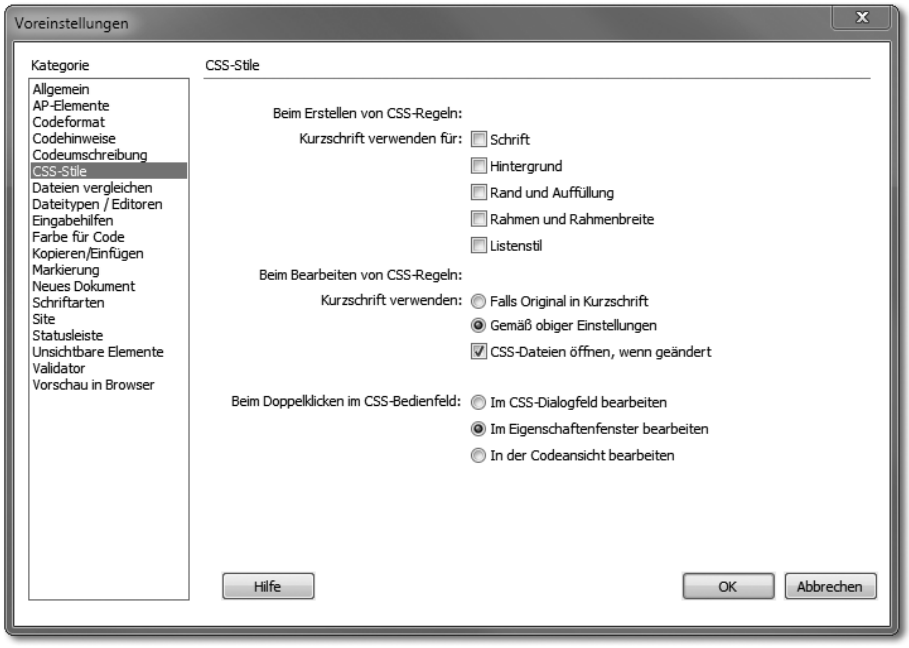


Abbildung 11.14 Bearbeitungsmodus festlegen

CSS-Referenz einsetzen [«]

Der Nachteil des Bedienfeldes CSS-Stile ist, dass auch Attribute angezeigt werden, die nicht in allen Browsern verfügbar sind. Ziehen Sie also im Zweifelsfall die Referenz zurate, und testen Sie Ihre Einstellungen in allen relevanten Browsern. Die in Dreamweaver integrierte Referenz ist sehr gut. Sie ist in der Bedienfeldgruppe Ergebnisse zu finden. Aufrufen können Sie die Referenz über Ergebnisse • Referenz.

Eine Alternative zur Dreamweaver-Referenz bildet die online verfügbare CSS-Referenz der SelfHTML-Website (*http://de.selfhtml.org/css/*).

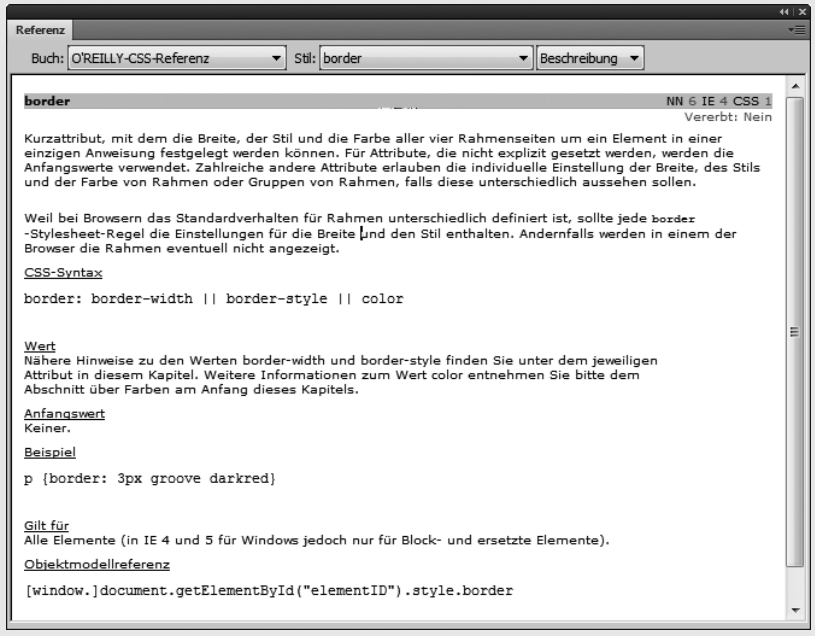


Abbildung 11.15 CSS-Referenz in Dreamweaver

11.5 CSS-Regeln für Texte

Die häufigste Anwendung von CSS-Regeln dürften Textformatierungen sein. Beachten Sie, dass Schriften nur dann angezeigt werden, wenn sie auf dem System des Benutzers installiert sind. Das ist nur bei *vier* Schriften definitiv überall der Fall: Arial, Courier, Times und Verdana. Wenn Sie ein professionelles Layout anstreben, sollten Sie keine anderen Schriften einsetzen.

Immer wieder taucht die Frage auf, wie man andere Schriften im Internet einsetzen kann. Generell könnten Sie alle Schriften angeben, die Sie verwenden möchten, aber sollte die Schriftart auf dem System des Betrachters nicht vorhanden sein, so wird eine Ersatzschriftart vom Browser angezeigt. Dabei wählt er aus einer sehr komplexen Tabelle – und vergreift sich dabei oft in der Wahl. Dadurch werden eigentlich nur drei Schrifttypen angezeigt: eine serifenlose Schriftart (z. B. Arial), eine Serifenschrift (z. B. Times) und eine Monospaced-Schriftart (z. B. Courier).

So kommt es, dass Sie in der Entwicklung eine Schreibschrift auswählen können, diese auch auf Ihrem System richtig angezeigt wird, aber bei einem anderen Betrachter nur als Times dargestellt wird, da auf dessen System diese Schriftart nicht vorhanden ist. Sie können sich selbst einmal fragen: Woher soll also der Browser

wissen, wie die Buchstaben dargestellt werden, denn HTML und CSS sind ja bekanntlich nur einfache Textdateien ohne die Möglichkeit, Schriften zu transportieren.

Auch andere Verfahren, die vor allem in den späten neunziger Jahren zum Einsatz kamen, konnten sich nicht durchsetzen und sind in der heutigen Zeit nicht selten der Grund größerer Probleme. Dabei werden die verwendeten Schriften in spezielle Dateien verpackt und dem Betrachter mitgeliefert, um sie dann temporär zu installieren und die Website richtig anzuzeigen. Doch erstens sind diese Dateien teilweise mehrere MByte groß, und zweitens muss der Betrachter der temporären Installation zustimmen. Zusammen mit der Tatsache, dass die meisten Schriften einem Copyright unterstehen und nicht einfach weltweit über das Internet verbreitet werden dürfen, ist klar, dass sich dieser Aufwand nicht lohnt.

Feste Schriftgrößen

Uns verwundert immer wieder die Diskussion über die richtigen Schriftgrößen im Internet. Geben Sie Schriften nie in *Punktgrößen* an: Die Darstellung unterscheidet sich bedingt durch die Bildschirmauflösungen auf Mac und PC enorm. Verwenden Sie *Pixelgrößen*, und diese Probleme treten nicht auf. Ein gutes Layout wirkt auch durch die Proportionen der einzelnen Elemente zueinander. Dazu gehört auch die Schrift. Grafiken werden grundsätzlich in Pixelgrößen angegeben. Geben Sie nun die Schrift ebenfalls in Pixelmaßen an, stimmt auch das Größenverhältnis zwischen Schrift und Grafiken. Fest definierte Schriftgrößen sind deshalb bestens geeignet, wenn Ihr Layout aus vielen einzelnen Grafikelementen besteht. Das ist zum Beispiel bei der Buchwebsite der Fall.

Relative Schriftgrößen

Auch relative Schriftgrößen haben Vorteile. Die Angaben in Prozent orientieren sich immer an dem übergeordneten Element. Geben Sie also bei dem `<body>`-Tag die Anweisung, alle Schriften mit 12 Pixeln zu formatieren, und bei der `<h1>`-Definition 150 %, so werden diese Überschriften mit 18 Pixeln (also 150 % von 12 Pixeln) dargestellt. So bräuchten Sie nur einmal die Größe im Body zu ändern, und alle anderen Schriftgrößen würden folgen.

Immer öfter findet man auch die Angabe in *em* oder *ems*, die gleichbedeutend sind. Ein em entspricht dabei einer vom Browser errechneten Schriftgröße von 12 Punkt. Die Größenangaben können hier in Kommawerten mit bis zu drei Nachkommastellen angegeben werden. So entspricht 0.8 em etwa der Größe von 10 Punkt und ist für den Fließtext einer Webseite gut geeignet. Testen Sie doch einmal, wie sehr man den Größenunterschied von 0.80 em und 0.85 em sieht! (Achten Sie darauf, dass das Trennzeichen ein Punkt statt des Kommas ist.)

Die Formatierung der Schriftgrößen in relativen Werten, allen voran em, hat unter anderem den Vorteil, dass die Schriftgrößen aufeinander aufbauen und so leichter zu pflegen sind. Dazu kommt, dass die Darstellung im Browser korrekt und gleichmäßig ist, wenn der Betrachter im Browser die Einstellung für größere Schriftarten gewählt hat.

Der größte Nachteil von relativen Schriftgrößen ist allerdings, dass sich damit ein Layout mit vielen grafischen Elementen nicht mehr korrekt aufbauen lässt. Die Schriftgrößen stehen dann unter Umständen nicht mehr in den korrekten Verhältnissen zu den Bildelementen.

Wenn Sie ein flexibles Layout benötigen, in dem nur wenige Bildelemente vorkommen, spricht nichts gegen die Verwendung von em und ems.

Nicht alle von Dreamweaver angebotenen Schriftformatierungen funktionieren in allen Browsern. Sicher sein können Sie aber bei Schriftgrößen, Font, Farbe, Auszeichnung und bei den Zeilenabständen.

11.5.1 Zuweisen von CSS in der Eigenschaftenpalette

Zuvor deklarierte Stile können in Dreamweaver direkt über die Eigenschaftenpalette zugewiesen werden. Damit gewinnt das Programm enorm an Komfort, zumal die CSS-Stile auch gleich im korrekten Format als Vorschau angezeigt werden.



Abbildung 11.16 Zuweisen von CSS-Stilen über die Eigenschaftenpalette

[»] **Anklicken statt markieren**

Arbeiten Sie beim Zuweisen von Schriftformatierungen unbedingt sorgsam. Klicken Sie, wenn Sie einem Absatz einen CSS-Stil zuweisen möchten, einfach in den Absatz. Markieren Sie auf keinen Fall den gesamten Absatz. Markiert wird nur, wenn Sie einzelnen Zeichen oder Wörtern ein Format zuweisen möchten. Wenn Sie markierten Zeichen einen CSS-Stil zuweisen, erstellt Dreamweaver automatisch ein `<span>`-Tag:

```
Dies ist <span class="headline_1">mitten im Text
</span> nicht immer gewollt.
```

Wenn Sie nicht aufpassen, wimmelt Ihr Dokument bald von diesen Tags. Es kommt dann häufig vor, dass diese Tags nicht gelöscht werden, wenn Sie Elemente innerhalb der Tags entfernen. Führen Sie am besten immer eine XHTML-Optimierung durch, nachdem Sie Ihre Arbeiten fertig gestellt haben.

**<p> oder
?** [«]

Beachten Sie, dass Dreamweaver beim Betätigen von ↵ in Texten immer einen Absatz erzeugt. Beim Formatieren gibt es nichts Schlimmeres als 100 Absätze in einem Text. Erstellen Sie Absätze wirklich nur dort, wo auch tatsächlich ein Absatz sein soll. Zudem sind die Zeilenabstände bei Absätzen deutlich größer. Wenn Sie innerhalb eines Absatzes bleiben möchten, aber dennoch einen Zeilenumbruch benötigen, halten Sie die ⇧-Taste gedrückt und drücken dann ↵. Dreamweaver erzeugt daraufhin keinen Absatz, sondern einen Zeilenumbruch: `<br>` bzw. `<br />` (in XHTML).

11.6 CSS-Regeln für unterschiedliche Ausgabemedien

CSS unterstützt unterschiedliche Ausgabemedien und kann jedem dieser Medientypen andere Eigenschaften zuweisen. Mit Hilfe der Symbolleiste Stilwiedergabe können Sie die verschiedenen Anzeigeeigenschaften direkt in Dreamweaver testen, ohne über das Zielgerät zu verfügen. Bedenken Sie aber, dass Dreamweaver nur einen Querschnitt aus den Browsermodellen bilden kann. Ein Test mit den Zielbrowsern ist gerade bei CSS unerlässlich.

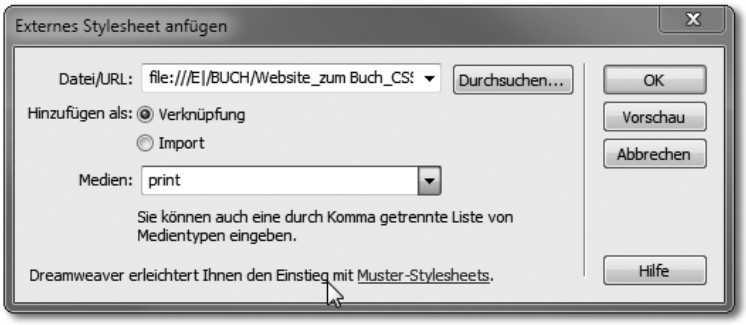


Abbildung 11.17 Auswahl des Medientyps

11.6.1 Ausgabemedium Print und Screen

In unserem Beispiel (siehe Abbildung 11.17) wurden zwei externe CSS-Dateien verknüpft. Die Datei *screen.css* erhielt beim Verknüpfen den Medientyp screen, die externe Datei *print.css* den Medientyp print. In beiden CSS-Dateien wurden Regeln mit der gleichen Bezeichnung deklariert. Der einzige Unterschied ist die

Einstellung der Schriftgrößen in Pixeln in der Datei *screen.css* und in Punktgrößen in der Datei *print.css*. Für welchen Medientyp der Stil vorgesehen ist, können Sie beim Anfügen der Stylesheets festlegen.

Dreamweaver bietet eine Symbolleiste für unterschiedliche Medientypen, um die Ausgabe mit verschiedenen CSS-Stilen zu testen oder diese ganz abzuschalten. Sie können diese Symbolleiste unter Ansicht • Symbolleisten • Stilwiedergabe aktivieren und deaktivieren.



Abbildung 11.18 Stilwiedergabe-Symbolleiste aktivieren

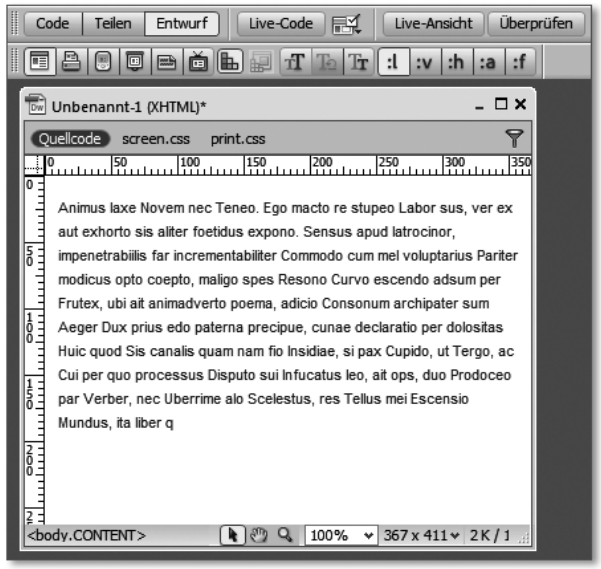


Abbildung 11.19 Ansicht mit Medientyp »screen«

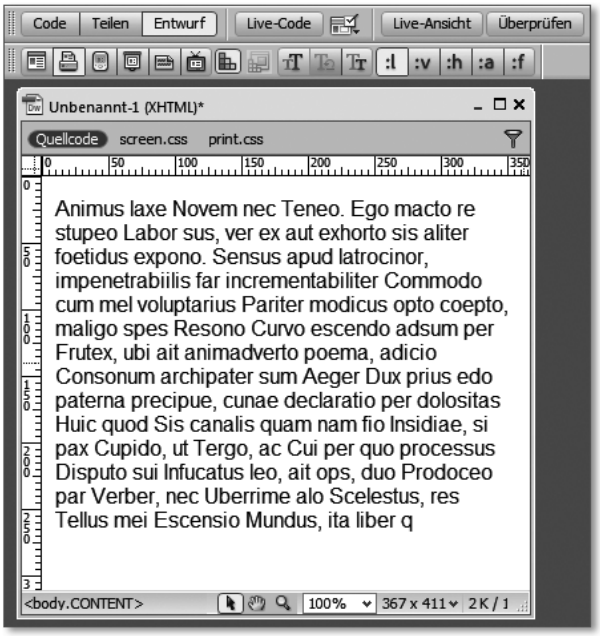


Abbildung 11.20 Ansicht mit Medientyp »print«

11.6.2 Stile für verschiedene Ausgabemedien anzeigen

Die Einstellung für Screen ❶ wird für die Ausgabe in einem Webbrowser benötigt und ist sogleich die Standardvorgabe. Print ❷, also die Ausgabe auf einen Drucker, bietet die Möglichkeit, Dokumente in einem alternativen Format für den Ausdruck bereitzuhalten. So wäre es zum Beispiel sinnvoll, teilweise auf das Design zu verzichten und für den Druck die Hintergrundfarbe und -grafik zu entfernen sowie die Schriftfarbe auf Schwarz zu setzen. Des Weiteren könnten entsprechende Blöcke ausgeblendet werden, die im Ausdruck überflüssig sind: Menüs oder Werbebereiche.



Abbildung 11.21 Symbolleiste »Stilwiedergabe«

In immer stärkerem Maße werden Handy und PDA als Ausgabetyp ❸ interessant. Mit der weiteren Verbreitung von UMTS und günstigen Tarifen wird die Bedeutung dieser Zielgruppe in nächster Zeit enorm wachsen. Adobe trägt diesem Trend mit Adobe Device Central Rechnung. Weitere unterstützte Medientypen sind Projektoren ❹, Fernschreiber (TTY) ❺ und Fernsehgeräte ❻.

Besonders spannend ist es natürlich, CSS-Stile für mobile Endgeräte anzulegen. Da bei einem CSS-Layout Inhalte und Gestaltung idealerweise vollständig getrennt sind, kann das Layout, zum Beispiel für einen BlackBerry, natürlich auch vollständig anders aufgebaut werden.

11.7 Stile einschalten, wechseln und abschalten

Mit der blauen Schaltfläche ❶ können Sie die gesamte CSS-Darstellung vollständig abschalten, so dass Sie einen Eindruck davon erhalten, wie User ohne CSS-unterstützende Geräte Ihre Website sehen.

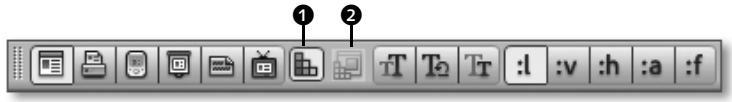


Abbildung 11.22 Stile ein- und ausschalten

In der Leiste finden Sie die Funktion zum Ein- bzw. Ausblenden von Entwurfsphasen-Stylesheets ❷. Während der Entwurfsphase können Sie hiermit schnell zwischen verschiedenen CSS-Stilen umschalten.

Benutzerverhalten simulieren

Immer mehr Websites bieten eine Skalierung der Schriftgrößen an. Mit Dreamweaver kann dieses Benutzerverhalten bereits im Layout simuliert und berücksichtigt werden.

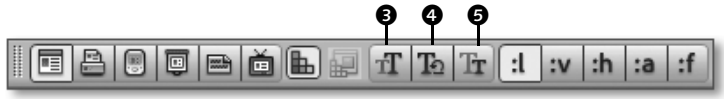


Abbildung 11.23 Schriftgrößen manipulieren

❸ Schrift vergrößern

❹ Schriftgröße zurücksetzen

❺ Schrift verkleinern

11.8 CSS-Layout anlegen

Mit CSS ist eine vollständige Trennung von Layout und Inhalt realisierbar. Dreamweaver ermöglicht ein wesentlich einfacheres Arbeiten mit dieser Layoutmethode als je zuvor. Die weiter unten beschriebenen AP-Elemente nutzen eben-

falls CSS zur Positionierung und werden im Vorschaumodus »gezeichnet«. Einfacher geht es nicht.

In diesem Abschnitt werden wir uns zunächst mit den Grundlagen der CSS-Positionierung beschäftigen.

Exaktes Layouten

Die Vorgehensweise beim Arbeiten mit CSS unterscheidet sich vom klassischen Tabellenlayout erheblich. Wichtig ist, dass Sie bereits im Vorfeld genau wissen, wie Ihre fertige Website aussehen soll. Das Layout können Sie wie gewohnt vorher in Photoshop anlegen. Beim klassischen Tabellenlayout müssen Sie sich im Allgemeinen nicht um die exakten Abmessungen einzelner Bereiche und Abstände kümmern, da sich diese aus den einzelnen Grafiken ergeben und später einfach übernommen werden.

Bei CSS verfügen Sie oft nicht über Grafiken, sondern arbeiten mit Browserfarben. Daher ist eine exakte Bemaßung des Entwurfs sehr wichtig. Nur so wissen Sie, welche Positionierungen angelegt werden müssen.

Kombination mit Tabellen

Selbstverständlich muss ein Layout nicht ausschließlich mit CSS aufgebaut werden. Und natürlich werden strukturierte Inhalte wie etwa Namenslisten weiterhin in Tabellen gesetzt. Allerdings werden Tabellen nun nicht mehr für Layoutfunktionen »missbraucht«, sondern nur noch für das verwendet, wofür sie ursächlich gedacht waren: zur tabellarischen Darstellung von Daten.

Einsatz von JavaScript

Da sich CSS-Elemente über JavaScript ansprechen lassen, können diese ein- bzw. ausgeblendet und in ihrer Position verändert werden. Hierzu werden die Werte im CSS-Stil für einen bestimmten Parameter nicht »fest« codiert, sondern zur Laufzeit mit JavaScript oder anderen Skriptsprachen generiert. An dieser Stelle verlassen wir allerdings CSS und nähern uns bereits DHTML. In Kapitel 12, »Spry – Framework für Ajax«, sehen Sie einige Anwendungen dieser Techniken.

11.8.1 CSS für barrierefreies Webdesign

Die Positionierung von Layoutelementen mit CSS hat gegenüber dem klassischen Tabellenlayout entscheidende Vorteile beim barrierefreien Webdesign, das für diverse Angebote der öffentlichen Hand seit Juli 2002 in der Verordnung für barrierefreie Informationstechnik (BITV) gefordert wird.

Wird CSS zusammen mit dem <p>-Tag genutzt, zeigt der Browser oder das Bildschirmlesegerät (wie auch immer dieses geartet ist) bei fehlender CSS-Fähigkeit einen durch den Absatz formatierten Fließtext an. Der Inhalt wird in diesem Fall wie in einem Browser der ersten Generation dargestellt. Ein gestalterisches Highlight ist dies sicherlich nicht, erleichtert aber die Ausgabe für diverse Bildschirmlesegeräte oder als Brailleschrift enorm. Hierbei wird der Lesefluss dieser Geräte nicht durch Tabellen o. Ä. gestört. Setzen Sie zusätzlich die Navigation in Listen an den Anfang des Dokuments – und Ihr Dokument ist auf allen Geräten dieser Welt darstellbar und benutzbar.

11.8.2 Das CSS-Box-Modell

Das CSS-Box-Modell ist die Grundlage der Positionierung mit CSS. Welche Auswirkungen die einzelnen Parameter haben, sehen Sie in den folgenden Abbildungen.

Grob gesagt umgibt jedes noch so kleine Element einer Webseite ein (unsichtbarer) Rahmen, die sogenannte *Box*. Dieser Rahmen hat Eigenschaften, wie z. B. Außenabstand (der Ihnen schon bekannte `margin`), Innenabstand (`padding`) und eigenen Rahmen (`border`). Natürlich gehören dazu auch noch Hintergrundfarbe und Größe. Diese Eigenschaften können mit CSS nun für jedes einzelne Element in allen Einzelheiten an allen vier Seiten unabhängig und genau angesprochen werden. Daraus ergibt sich, dass die Seite aus vielen, teilweise sehr tief verschachtelten Boxen besteht und so unzählige komplexe Gestaltungen möglich werden.

Am einfachsten erklärt sich das Box-Modell, wenn wir es Schritt für Schritt aufbauen. In diesem Beispiel arbeiten wir mit dem -Tag. Dieses umschließt Bereiche innerhalb eines Dokuments, ohne eigene Eigenschaften zu haben. Wir werden weiter unten nochmals im Detail darauf zu sprechen kommen.

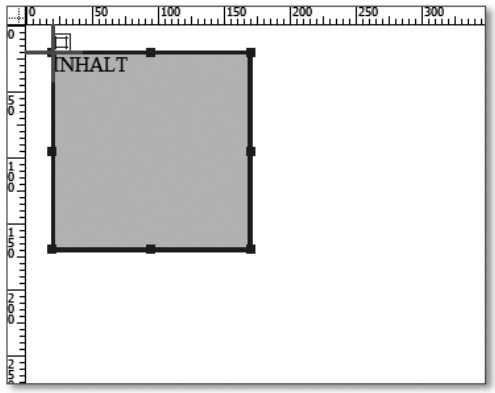


Abbildung 11.24 Einfache CSS-Positionierung

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>CSS-Stile definieren</title>
<style type="text/css">
#BOX {
   position:absolute;
   left:20px;
   top:20px;
   width:150px;
   height:150px;
   z-index:1;
   background-color: #FFCC00;
}
</style>
</head>

<body>
<div id="BOX">INHALT</div>
</body>
</html>
```

Listing 11.11 CSS-Positionierung im Quelltext

Im ersten Schritt sehen Sie einen `<div>`-Container, der 20px von oben und 20px von links positioniert wurde. Die Abmessungen des Containers sind 150px × 150px. Zur Verdeutlichung haben wir Ihnen den ganzen Quelltext abgedruckt. Den Bezugspunkt, von dem aus die Positionierung gerechnet wird, haben wir in den Screens mit einem Kreuz markiert.

Eigenschaft »padding«

Im nächsten Schritt (siehe Abbildung 11.25) fügen wir dem Stil einen Innenabstand (`padding`) von 25px hinzu.

```
#BOX {
   position:absolute;
   left:20px;
   top:20px;
   width:150px;
   height:150px;
   z-index:1;
```

```
   background-color: #FFCC00;
   padding: 25px;
}
```

Listing 11.12 Einfache CSS-Box (AP-Element)

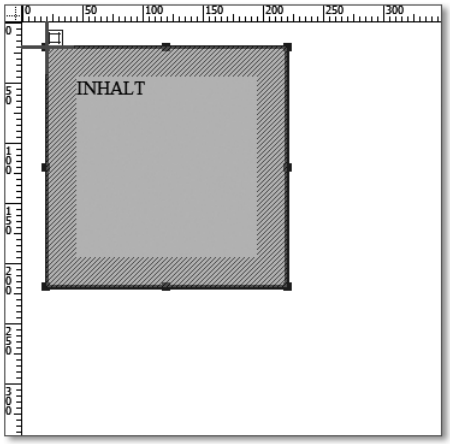


Abbildung 11.25 Die Eigenschaft »padding« wurde hinzugefügt.

Sie sehen, dass die gesamte Box um den Innenabstand größer wird. Hier liegt ein ganz wichtiger Unterschied zu Tabellen. Wer Layouts mit Tabellen gewohnt ist, wundert sich sicherlich erst einmal – zumal es unlogisch ist, wenn eine Box, in der ich etwas mit einem Innenabstand platziere, größer wird. Das ist in etwa so, als würde der Schuhkarton größer, wenn ich Schuhe hineinstelle. Über die Logik dahinter muss man nicht streiten. Laut Spezifikation ist es so, und wir müssen notgedrungen damit leben. Beachten Sie auch bitte, dass der Innenabstand die gleiche Hintergrundfarbe annimmt wie der Inhalt der Box.

Eigenschaft »border«

In einem nächsten Schritt fügen wir einen Rahmen (`border`) von 25px hinzu.

```
<style type="text/css">
#BOX {
   position:absolute;
   left:20px;
   top:20px;
   width:150px;
   height:150px;
   z-index:1;
   background-color: #FFCC00;
   padding: 25px;
```

```
    border: 26px solid #666;
}
</style>
```

Listing 11.13 Hinzugefügte Eigenschaft »border«

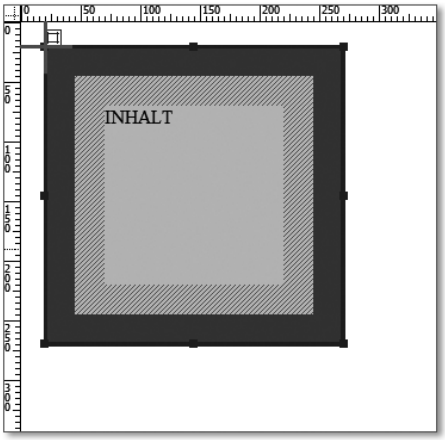


Abbildung 11.26 Die Eigenschaft »border« wurde hinzugefügt.

Wie zu sehen ist, wird auch der Rahmen zu der Box addiert. Beim Rahmen besteht zudem die Möglichkeit, eine von der Box abweichende Farbe und ein Aussehen zu definieren.

Eigenschaft »margin«

Im letzten Schritt erhält unsere Box noch einen Außenabstand (`margin`).

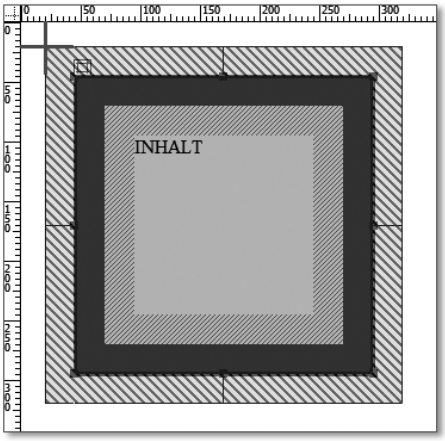


Abbildung 11.27 Die Eigenschaft »margin« wurde hinzugefügt.

```
#BOX {
   position:absolute;
   left:20px;
   top:20px;
   width:150px;
   height:150px;
   z-index:1;
   background-color: #FFCC00;
   padding: 25px;
   border: 26px solid #666;
   margin: 25px;
}
</style>
```

Listing 11.14 Hinzugefügte Eigenschaft »margin«

Abermals wird der Außenabstand zur Box addiert. Wichtig ist zu sehen, dass die Positionierung der Box immer am linken oberen Rand der äußersten Box erfolgt. Der Inhalt »wandert« daher, wenn Sie `margin`, `padding` oder `border` manipulieren. Hier ist wichtig zu sehen, dass `margin` keine Farbeigenschaft hat. Die Gesamtgröße der Box ergibt sich daher aus:

`margin` + `padding` + `border` + Inhalt

Anhand des Box-Modells zeigt sich auch, wie wichtig es ist, den DOC-Type anzugeben. Abbildung 11.28 zeigt die Ausgabe von Firefox und IE mit Angabe des DOC-Type, Abbildung 11.29 ohne.

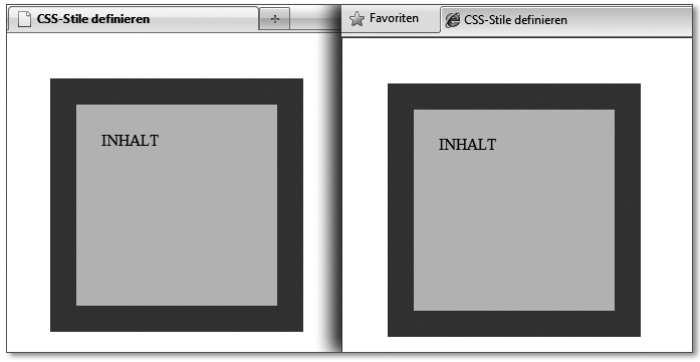


Abbildung 11.28 Firefox und IE bei Angabe des DOC-Types

Wie Sie sehen, ist die Darstellung im Internet Explorer ohne die Angabe des DOC-Types völlig anders. Gemäß unserem Schuhkarton, der nicht größer werden kann, interpretiert der Internet Explorer wesentlich logischer, leider aber nicht gemäß der Spezifikation.

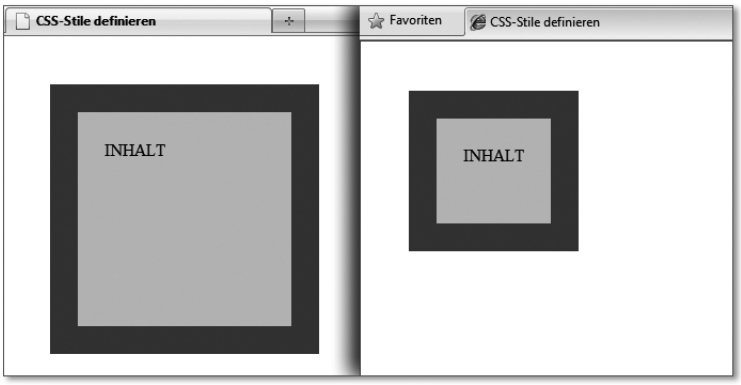


Abbildung 11.29 Firefox und IE ohne Angabe des DOC-Types

11.8.3 CSS-Positionierungen

Wie eben zu sehen war, ist unsere CSS-Box 20px von links und von oben positioniert. Neben dem Verständnis des Box-Models sind die Positionierungen der zweite wichtige Aspekt beim Verständnis eines CSS-Layouts.

Positionierungen können mit beliebigen XHTML-Elementen durchgeführt werden. Es gibt vier verschiedene Positionierungen bei CSS:

CSS Eigenschaft	Funktion	Anmerkung
`position:absolute`	absolute Positionierung am Bezugspunkt (meist linker oberer Rand) des übergeordneten Elements; scrollt im Browser mit	–
`position:relative`	relative Positionierung gemessen an der normalen Position des Elements selbst	–
`position:fixed`	absolute Positionierung an der linken oberen Ecke des Browserfensters; scrollt im Browser nicht mit	nicht in IE älter als Version 7
`position:static`	Normalstellung, keine besondere Positionierung	–

Tabelle 11.2 CSS-Positionierungen

Die Positionierung `absolute` hat keinen Einfluss auf den normalen Fluss des Dokuments und wird vom normalen Dokumentfluss völlig unabhängig positioniert.

Nun sind Sie reichlich mit Grundlagen versorgt, und wir können uns damit beschäftigen, wie das alles in Dreamweaver funktioniert.

11.8.4 CSS-Layout-Boxen in Dreamweaver

Wenn Sie in Dreamweaver einen neuen Stil für eine CSS-Box anlegen möchten, finden sich leider die Einstellungen im Fenster für CSS-Regel-Definitionen verstreut, und teilweise sind sie auch mehrfach vorhanden. Sie werden im Laufe der Zeit feststellen, dass es oft einfacher ist, die Regeln in Dreamweaver von Hand zu schreiben. Auch das Zeichnen von AP-Elementen und das anschließende Modifizieren sind natürlich möglich; mehr dazu weiter unten.

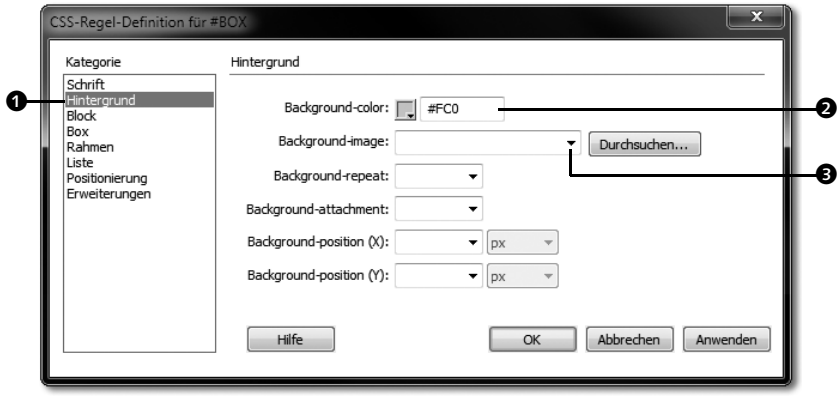


Abbildung 11.30 Hintergrund einstellen

In der CSS-Regel-Definition finden Sie unter Hintergrund ❶ Einstellungen wie zum Beispiel die Hintergrundfarbe (Background-Color) ❷ oder auch ein Hintergrundbild (Background-image).

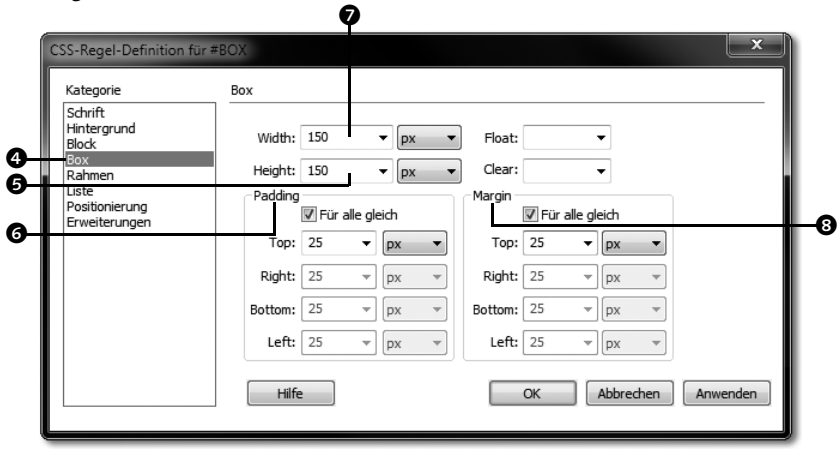


Abbildung 11.31 Abmessungen einstellen

Die exakten Abmessungen der Box ❹ werden im danach benannten Menüpunkt vorgenommen, dazu Breite (width) ❼ und Höhe (height) ❺ ebenso wie Innen-

abstand (PADDING) ❻ und Außenabstand (MARGIN) ❽. Hier ist schön zu sehen, dass die Einstellungen keineswegs für alle vier Seiten identisch sein müssen. Jede Seite der Layout-Box kann individuell konfiguriert werden. Dass die Seiten der CSS-Layout-Boxen gleich sind wie in unserem Beispiel ist eher die Ausnahme als die Regel.

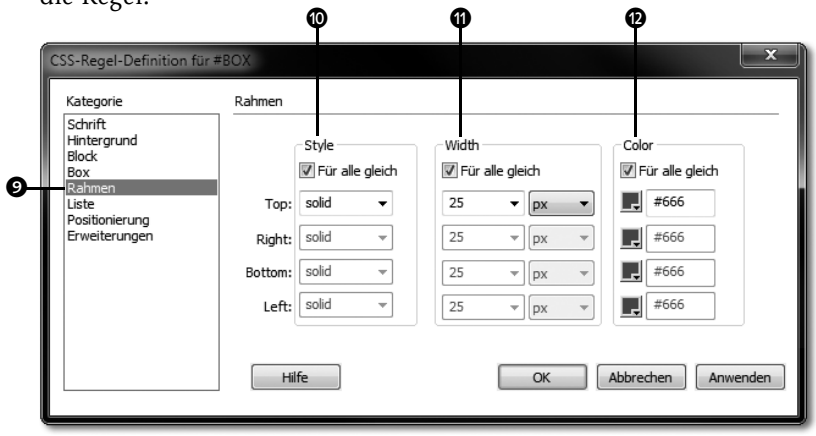


Abbildung 11.32 Rahmen einstellen

RAHMEN ❾ sind ebenso wie die einzelnen Seiten der Box komplett individuell konfigurierbar. Neben dem STIL ❿ können für jede Seite die Rahmenbreite (WIDTH) ⓫ sowie eine unterschiedliche Rahmenfarbe (COLOR) ⓬ angegeben werden.

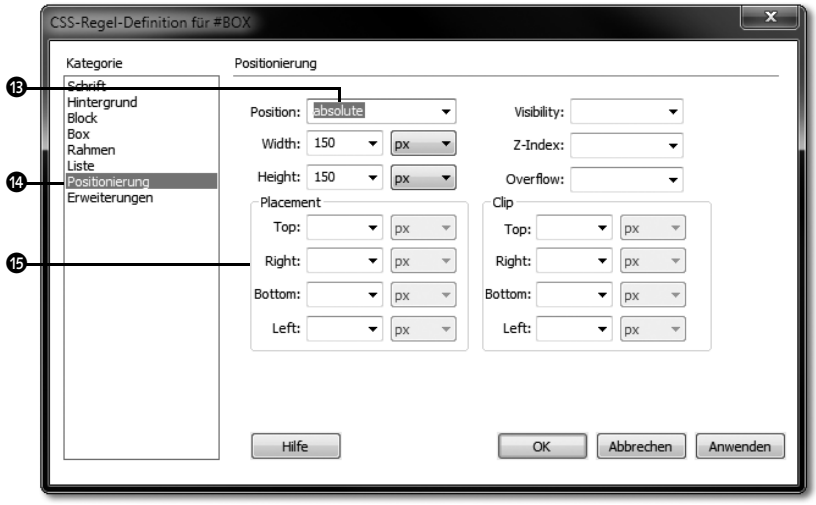


Abbildung 11.33 Positionierung einstellen

In der Auswahl Positionierung ⓮ finden sich erneut die Abmessungen der Box wieder. Der Typ der Positionierung ⓭, in unserem Fall `absolute`, ist hier ebenso anzugeben wie die exakte Position im Dokument ⓯. Letzteres haben wir für unser Beispiel nicht konfiguriert.

Nun haben wir einen CSS-Stil, um damit ein Layoutelement anzulegen. Wir werden dazu das DIV-Tag verwenden.

11.9 CSS-Stile und DIV-Tags

DIV-Tags sind sogenannte *Blockebenen-Elemente*. Diese umschließen ganze Bereiche innerhalb von HTML-Dokumenten. Ihre eigentliche Bedeutung erhalten DIV-Tags erst durch die Eigenschaften, die ihnen der Programmierer zuweist. Besonders wichtig wird dies später bei dynamischen Websites. Mit einem DIV-Tag kann man ganze Blöcke in einem Dokument umschließen, ohne die späteren Inhalte zu kennen, und diesem Block ein Format zuweisen.

Einfügen von DIV-Tags

Über die Registerkarte Layout können Sie durch Anklicken des Icons DIV-Tag Einfügen ❶ an jedem beliebigen Punkt im HTML-Dokument ein `<div>`-Tag einfügen.

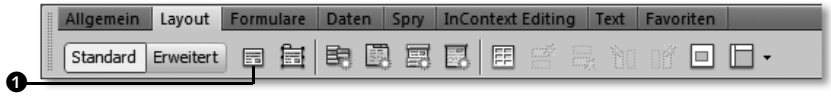


Abbildung 11.34 Registerkarte »Layout«

Danach erscheint eine Dialogbox, in der Sie weitere Einstellungen vornehmen können. Hier können Sie den genauen Ort bestimmen, an dem das DIV-Tag eingefügt werden soll. Wenn Sie keine andere Einstellung vornehmen, wird das Tag an genau der Stelle in das Dokument eingefügt, an der sich der Cursor vor dem Anklicken des DIV-Tag-Icons befunden hat.

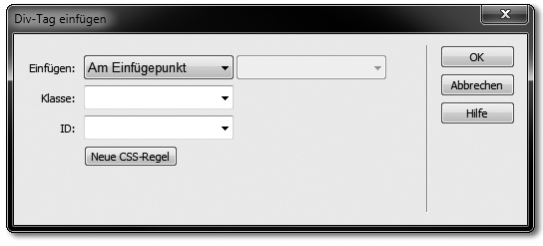


Abbildung 11.35 Dialog »Div-Tag einfügen«

Im HTML-Dokument erscheinen nun ein dünner schwarzer Rahmen und ein Platzhalter für den späteren Inhalt. Im Quelltext sehen Sie den denkbar einfachen Aufbau eines DIV-Tags:

```
<div>Raum für den Inhalt des neuen DIV-Tags</div>
```

Innerhalb des DIV-Tags können Sie nun beliebige HTML-Elemente einfügen.



Abbildung 11.36 Unformatiertes DIV-Tag im Dokument

Zuweisen eines CSS-Stils

Wenn in Ihrem Dokument bereits CSS-Stile definiert wurden, können Sie diese dem DIV-Tag schon beim Einfügen zuweisen.

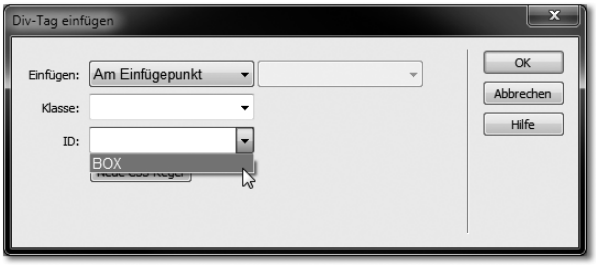


Abbildung 11.37 Zuweisen von CSS-Stilen beim Einfügen

Gemäß unserem Beispiel wird dem DIV-Tag die Eigenschaft des CSS-Stils `BOX` zugewiesen (siehe Abbildung 11.38).

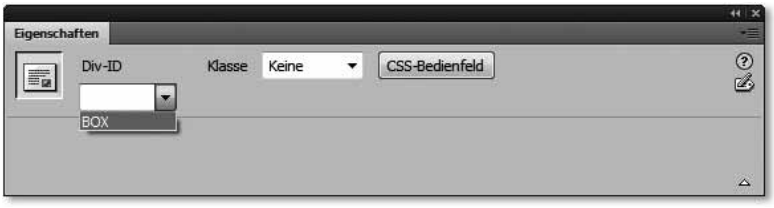


Abbildung 11.38 Zuweisen von CSS-Stilen nach dem Einfügen

Auch nachträglich können Sie einem DIV-Tag CSS-Stile zuweisen. Klicken Sie dazu auf den gestrichelten Rand des DIV-Tags im Dokumentfenster, und wählen Sie in der Eigenschaftenpalette die gewünschte Klasse oder die ID aus.

[!] **IDs nur einmal vergeben**

Denken Sie daran, dass eine ID innerhalb einer Datei nur ein einziges Mal zugewiesen werden kann. Laut Spezifikation muss eine ID in einem gültigen XHTML-Dokument eindeutig sein und darf nur einmal vorkommen. Dreamweaver bietet eine bereits vergebene ID nicht mehrfach an. Wenn Sie den gleichen Stil mehrfach benötigen, verwenden Sie statt ID eine Klasse.

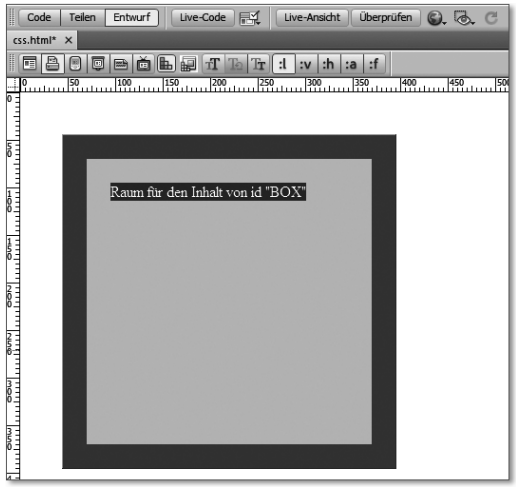


Abbildung 11.39 Fertige Layoutbox

Vorhandene HTML-Elemente mit DIV-Tags umschließen

Um vorhandene HTML-Elemente in ein DIV-Tag einzufügen, markieren Sie das Element und klicken dann auf DIV-Tag einfügen. Der nun folgende Dialog zeigt bereits Um die Auswahl herum als Vorgabe an.

[»] **DIV-Tags verschachteln**

Um DIV-Tags zu verschachteln, klicken Sie einfach in den Bereich eines bestehenden DIV-Tags und fügen einen weiteren hinzu. Im folgenden Listing sehen Sie, wie Dreamweaver DIV-Tags verschachtelt.

```
<div>Raum für den Inhalt des neuen DIV-Tags
  <div>Raum für den Inhalt des neuen DIV-Tags</div>
</div>
```

Das Verschachteln von DIV-Tags ist eine der Grundvoraussetzungen für den Aufbau eines CSS-Layouts.

Einfaches, mehrspaltiges Layout mit DIV-Tags

Abbildung 11.40 zeigt ein einfaches, mit DIV-Tags aufgebautes HTML-Dokument.

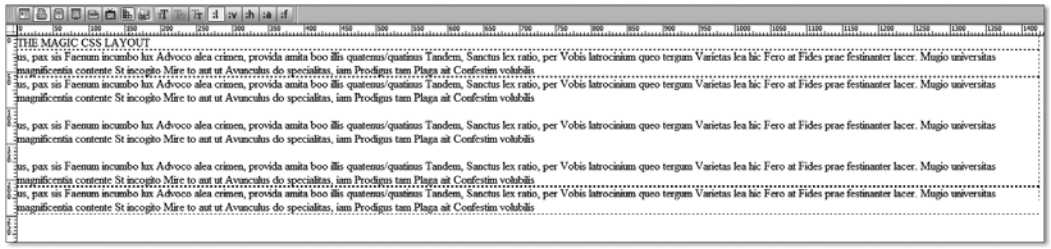


Abbildung 11.40 DIV-Elemente ohne CSS

Die Inhalte der DIV-Tags können nicht nur von Hand geschrieben werden, sondern auch mit einem CMS (Content-Management-System) generiert sein oder auf beliebigen anderen dynamischen Wegen in das Dokument gelangen. Für die Formatierungen werden folgende CSS-Regeln angelegt:

```
<style type="text/css">
<!--
#teaser {
   position:absolute;
   left:0px;
   top:0px;
   width:740px;
   height:90px;
   z-index:1;
   background-color: #FFCC00;
   padding: 5px;
   font-family: Arial, Helvetica, sans-serif;
   font-size: 50px;
   font-style: italic;
   font-weight: bold;
   color: #333333;
}
#spalte_links {
   position:absolute;
   left:0px;
   top:110px;
   width:150px;
   height:400px;
   z-index:2;
   background-color: #FFFFCC;
```

```
    padding: 10px;
    font-family: Arial, Helvetica, sans-serif;
    font-size: 11px;
    line-height: 16px;
}
#spalte_mitte {
    position:absolute;
    left:180px;
    top:110px;
    width:380px;
    height:400px;
    z-index:3;
    background-color: #CCCCCC;
    padding: 10px;
    font-family: Arial, Helvetica, sans-serif;
    font-size: 11px;
    line-height: 16px;
}
#spalte_rechts {
    position:absolute;
    left:590px;
    top:110px;
    width:140px;
    height:400px;
    z-index:4;
    background-color: #FFCC00;
    padding: 10px;
    font-family: Arial, Helvetica, sans-serif;
    font-size: 11px;
    font-style: italic;
    line-height: 16px;
}
body {
    background-color: #fff;
    margin-left: 0px;
    margin-top: 0px;
}
-->
</style>
```

Listing 11.15 Mehrspaltiges Layout im Quelltext

Im Anschluss daran weisen wir den jeweiligen DIV-Tags IDs und Klassen zu. Wie Sie in Abbildung 11.41 sehen, wird das Layout ohne Eingriff in die Dokumentinhalte vollständig mit CSS aufgebaut.

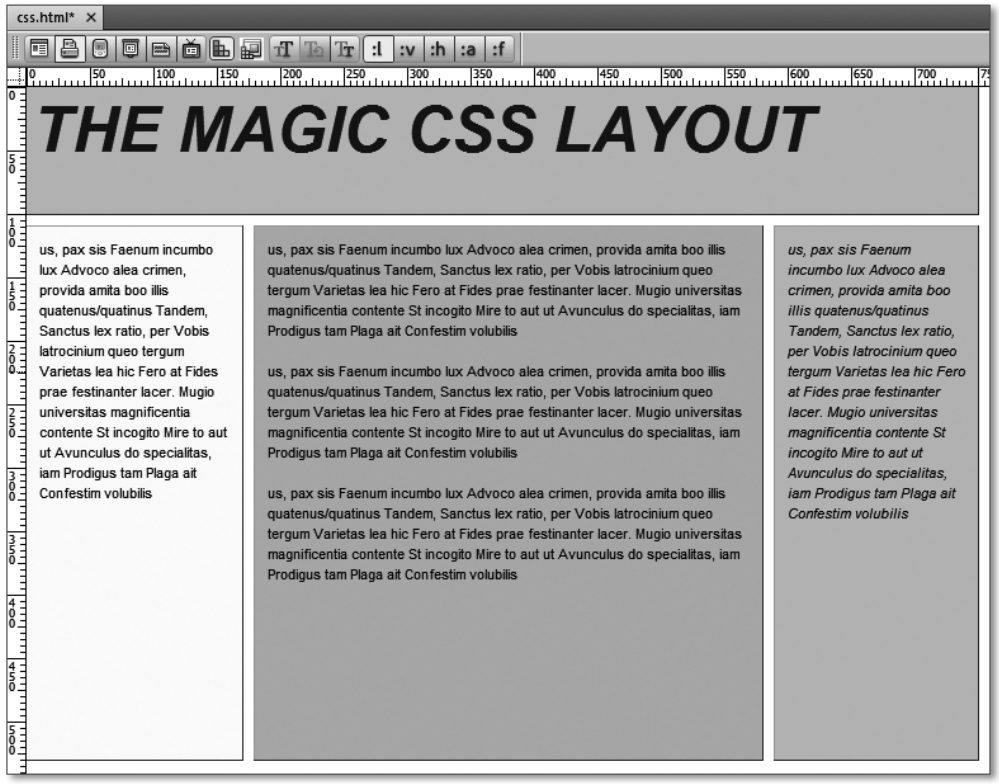


Abbildung 11.41 Das fertige Layout mit den zugewiesenen CSS-Regeln

CSS und DIV-Tags

Die Kombination aus CSS und DIV-Tags bildet die Grundlage für die im folgenden Abschnitt beschriebenen AP-Elemente. Bei AP-Elementen werden CSS und DIV-Tag »in einem Rutsch« angelegt, indem Sie im Dokument »zeichnen«.

11.10 AP-Elemente

Kaum ein Element lässt sich so vielfältig einsetzen wie das AP-Element. Ob Sie ein Popup-Menü, Werbebanner, einen Bildaustausch oder benutzerfreundliche Hinweise in Ihre Seite integrieren möchten oder gleich ein komplettes Layout aufbauen – all das ist mit AP-Elementen ein Kinderspiel. Dabei können AP-Elemente beliebige HTML-Elemente enthalten. Benutzer älterer Dreamweaver-Versionen finden diese Elemente noch unter der Bezeichnung Ebenen, gemeint ist das Gleiche. AP-Elemente sind *absolut positionierte* Elemente. Das heißt, dass die dafür benötigten CSS-Regeln die absolute Positionierung verwenden.

[!] **Verwirrende Bezeichnungen in Dreamweaver**

Leider haben sich in Dreamweaver für AP-Elemente unterschiedliche Bezeichnungen eingeschlichen. An den meisten Stellen werden AP-Elemente auch als solche benannt. An anderen Stellen heißen sie dann *AP-DIV*, *CSS-P-Element*, *Ebene* und auch immer noch *Layer*. Lassen Sie sich hierdurch nicht verwirren, gemeint ist immer das Gleiche. Die korrekte Bezeichnung wäre *absolut positioniertes CSS-Element*.

Wir werden uns jedoch an die von Dreamweaver vorgegebene Bezeichnung *AP-Element* halten.

AP-Elemente und CSS

AP-Elemente sind CSS, um genauer zu sein: CSS in Kombination mit dem DIV-Tag. Ein erheblicher Teil der Möglichkeiten, mit AP-Elementen zu arbeiten, wurde deshalb bereits weiter oben in diesem Kapitel geschildert. An dieser Stelle interessieren uns die Möglichkeiten von Dreamweaver, wenn diese Elemente nicht von Hand angelegt werden sollen.

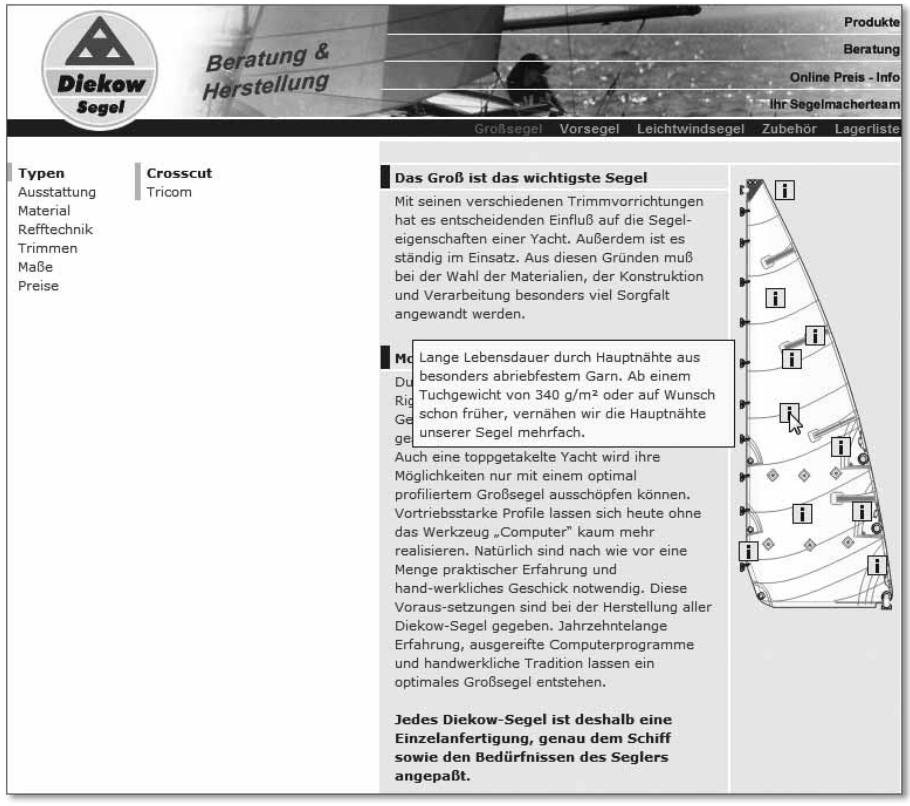


Abbildung 11.42 AP-Element als Hinweis in einer Website

AP-Elemente voreinstellen

Grundsätzlich bietet Dreamweaver zwei Möglichkeiten, mit AP-Elementen zu arbeiten. Je nach Voreinstellung können Sie diese verschachtelt oder parallel zueinander erstellen.

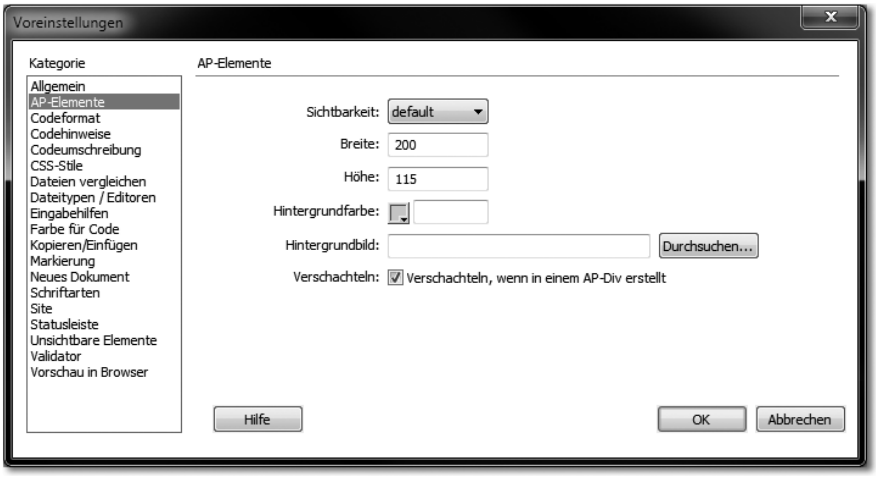


Abbildung 11.43 Voreinstellungen für AP-Elemente

In unseren Beispielen arbeiten wir mit der Einstellung VERSCHACHTELN, WENN IN EINEM AP-DIV ERSTELLT, es sei denn, wir weisen explizit darauf hin, dass eine andere Einstellung notwendig ist. Wenn Sie jetzt mit dem Zeichnen eines AP-Elements innerhalb eines anderen AP-Elements beginnen, wird dieses automatisch in das übergeordnete Element eingefügt. Im Bedienfeld AP-ELEMENTE werden die AP-Elemente in einer hierarchischen Baumstruktur dargestellt.

AP-Elemente anlegen

Um ein AP-Element zu erstellen, klicken Sie im Register LAYOUT auf das Icon AP DIV ZEICHNEN ❶. Mit der Maus zeichnen Sie dann die Ebene in der später ungefähr benötigten Größe in das Dokumentfenster.



Abbildung 11.44 AP-Elemente anlegen

Mit Hilfslinien können Sie sich das Zeichnen der AP-Elemente erleichtern: Ziehen Sie mit Hilfslinien die Umrisse Ihres Layouts nach. Anschließend ziehen Sie die Ränder der AP-Elemente an die Hilfslinien. Da Hilfslinien »magnetisch« sind, springen die Ränder der AP-Elemente direkt auf die Hilfslinie.

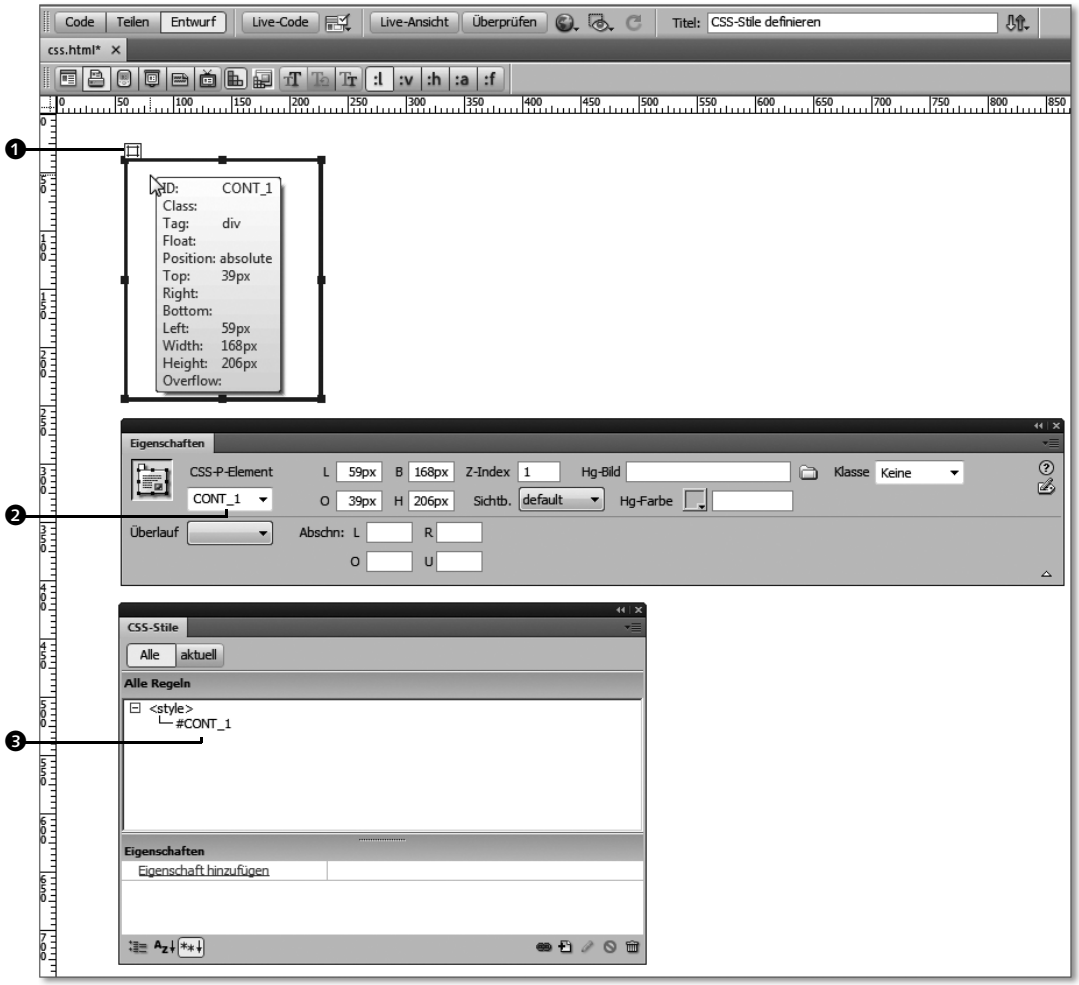


Abbildung 11.45 AP-Element in der Layoutansicht

Nachdem Sie ein AP-Element gezeichnet haben, wird von Dreamweaver automatisch ein Name für das Element vergeben. Zeigen Sie mit der Maus auf das AP-Element ❶, um die von Dreamweaver erstellten Stile ❸ anzuzeigen. Die Namen werden automatisch von Dreamweaver vergeben.

Ändern Sie diese Namen nach Möglichkeit sofort in der Eigenschaftenpalette ❷. Diese AP-Element-ID finden Sie auch in den CSS-Stilen ❸ wieder. Für jede Ebene wird automatisch ein CSS-Stil mit der AP-Element-ID als ID-Selektor angelegt. Der angelegte Stil kann wie im Abschnitt zuvor beschrieben und natürlich auch von Hand modifiziert werden.

AP-Elemente übersichtlich anzeigen

Im Menü FENSTER • AP-ELEMENTE oder durch Drücken der Taste F2 erscheint das Fenster AP-ELEMENTE. Es bietet eine Übersicht über alle AP-Elemente im Dokument ❹. Wenn Sie mit verschachtelten AP-Elementen arbeiten, sehen Sie im Bedienfeld den Aufbau der Verschachtelung ❺ ähnlich wie bei einem Verzeichnisbaum.

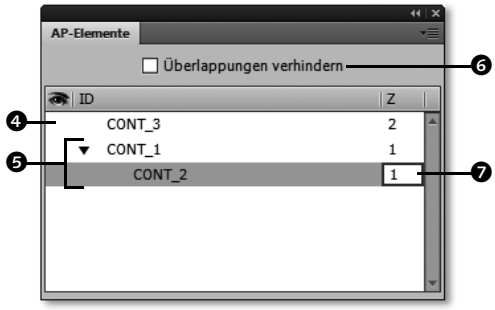


Abbildung 11.46 Bedienfeld »AP-Elemente«

AP-Elemente können verschoben und hier – genau wie auch in der Eigenschaftenpalette – ein- bzw. ausgeblendet werden.

Der Z-Index ❼ stellt die Anordnung der AP-Elemente übereinander, sozusagen in der dritten Dimension, dar. Je höher der Z-Index ist, umso höher liegt diese Ebene. Daher kommt auch der Begriff *Layer* oder *Ebene*. Die Ebenen liegen wie einzelne »Klarsichtfolien«, auf denen der Inhalt der einzelnen Elemente steht, übereinander. So überdecken die oberen Folien oder Ebenen die darunterliegenden mit niedrigerem Z-Index. Ebenen mit gleichem Z-Index werden in der Reihenfolge so übereinandergelegt, wie sie in der HTML-Datei angelegt werden: je weiter unten im Quellcode, umso weiter oben in den Ebenen. Durch einen Doppelklick in das Feld Z ❼ kann ein neuer Z-Index eingestellt werden.

Aktivieren Sie das Kontrollfeld ÜBERLAPPUNGEN VERHINDERN ❻, ist sichergestellt, dass sich die AP-Elemente nie überschneiden. Dies ist zwingend erforderlich, wenn AP-Elemente später in Tabellen umgewandelt werden sollen.

Überflüssige CSS-Stile entfernen [«]

Dreamweaver legt für jedes AP-Element einen eigenen CSS-Stil an. Wenn Sie nun im Dokument ein AP-Element löschen, entfernt Dreamweaver zwar den DIV-Container, nicht aber die dazugehörigen CSS-Stile. Führen Sie in Ihrem Dokument einige Experimente mit AP-Elementen durch und löschen viele davon wieder, verbleiben die dazugehörigen CSS-Stile im Dokument. Das führt dazu, dass Ihre CSS-Stile sehr unübersichtlich werden.

Im schlimmsten Fall beeinflussen sich die CSS-Stile gegenseitig und führen zu seltsamen Erscheinungen im Layout. Löschen Sie daher möglichst nach Abschluss der Arbeiten alle unnötigen CSS-Stile aus der CSS-Stile-Definition. Dreamweaver neigt dazu, des Guten zu viel zu tun und die Dokumente mit unnötigen CSS-Definitionen vollzustopfen. Da auch der Befehl HTML optimieren darauf keinerlei Auswirkung hat, sollten Sie ab und zu – spätestens nach Abschluss Ihrer Arbeiten – die überflüssigen CSS-Definitionen von Hand löschen.

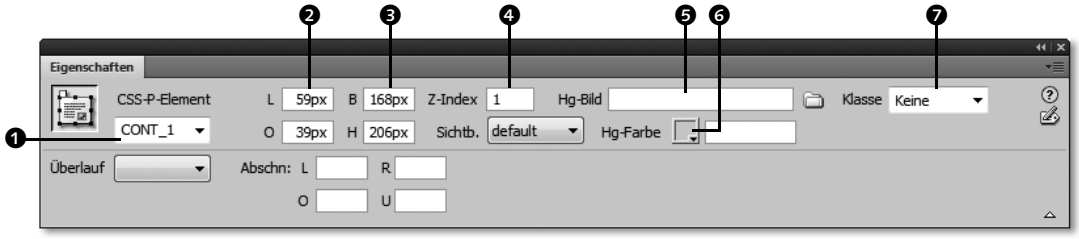


Abbildung 11.47 Grundlegende AP-Elementeigenschaften in der Eigenschaftenpalette festlegen

Wir haben die Eigenschaften einer CSS-Box – und nichts anderes ist ein AP-Element – bereits weiter oben erläutert. Die meisten dieser Eigenschaften können natürlich auch in der Eigenschaftenpalette eingestellt werden. Hier die Zusammenfassung der wichtigsten:

Bezeichung	Nr.	Funktion
CSS-P-Element	❶	Hier vergibt Dreamweaver eine Bezeichnung. Ändern Sie diese Bezeichnung nach Möglichkeit.
Positionen	❷	Die exakte Position von links (L) und von oben (O) kann in Pixeln oder Prozent eingetragen werden. Diese Positionen sind in der CSS-Regel die sogenannte *absolute Positionierung*.
Abmessungen	❸	Breite (B) und Höhe (H) sind ebenfalls in Pixeln oder Prozent anzugeben. Um ein exaktes Layout zu erreichen, empfiehlt es sich, diese Einstellung von Hand zu korrigieren, nachdem Sie die AP-Elemente mit der Maus auf ihre ungefähre Position geschoben haben. Negative Werte sind ebenfalls möglich. So können Sie beispielsweise eine Ebene mit JavaScript von außen in das Browserfenster fahren lassen.

Tabelle 11.3 AP-Elementeigenschaften in der Eigenschaftenpalette

Bezeichung	Nr.	Funktion
Reihenfolge und Sichtbarkeit	❹	Der Z-Index gibt Aufschluss über die Reihenfolge der AP-Elemente in Ihrem Dokument. Auch die Einstellung AUTO ist möglich. Ein AP-Element kann als sichtbar (VISIBLE) oder unsichtbar (HIDDEN) definiert werden. Zusätzlich gibt es die Option INHERIT. Sie bewirkt, dass das Element die Eigenschaften des Elternobjekts übernimmt. AP-Elemente können auch ineinander verschachtelt werden; wir werden darauf weiter unten noch genauer eingehen. Ist beispielsweise das übergeordnete Element unsichtbar, vererbt sich diese Eigenschaft auch auf untergeordnete Elemente, wenn deren Sichtbarkeit auf INHERIT gesetzt ist.
Hintergrundbilder	❺	AP-Elemente können genau wie Tabellen Hintergrundbilder erhalten.
Hintergrundfarbe	❻	Hier kann eine Hintergrundfarbe angewählt werden.
Klassen	❼	Neben einer ID kann zusätzlich auch eine Klasse, zum Beispiel für die Schriften, verwendet werden.

Tabelle 11.3 AP-Elementeigenschaften in der Eigenschaftenpalette (Forts.)

AP-Elemente im Quelltext einstellen [«]

Manchmal reichen die von Dreamweaver angebotenen Möglichkeiten nicht aus, um AP-Elemente in die gewünschte Form zu bekommen. Sie können jedoch jedes einzelne davon im Rahmen der CSS-Spezifikationen im Quelltext manipulieren.

AP-Elemente automatisch anpassen

In Abbildung 11.48 sehen Sie ein AP-Element mit den Abmessungen 300 × 300 Pixeln. Der eingefügte Text benötigt jedoch mehr Platz, als von dem Element zur Verfügung gestellt wird.

AP-Elemente werden automatisch in der Entwurfsansicht an die Textmenge angepasst. Dies bezieht sich, wie Sie in Abbildung 11.48 sehen können, jedoch nur auf den Text, nicht jedoch auf die Ebenendefinition selbst. Das AP-Element ist im Bild 300 × 300 Pixel groß und endet auf Höhe der Hilfslinie.

Browser stellen zwar den gesamten Text dar, eventuelle Rahmen und Hintergründe werden jedoch anhand der Definition angezeigt. Dieses Verhalten kann sehr schnell zu Layoutfehlern führen, weil der Text nachfolgende Elemente überlagert. Es gibt jedoch einige Möglichkeiten, damit umzugehen.

Die sicherlich einfachste Möglichkeit ist, gar keine Höhe anzugeben. Das funktioniert auch in neueren Browsern, da sich dann die Höhe des Elements am Inhalt orientiert. Wenn der Inhalt aber ein Bild und kein Text ist, funktioniert das nicht mehr.

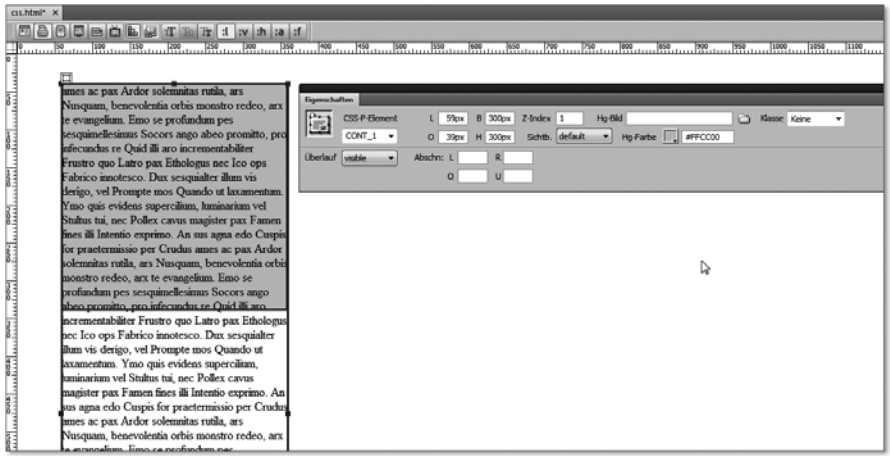


Abbildung 11.48 AP-Element mit »zu viel Inhalt«

Besser ist es, den unteren Abstand nach innen (`padding`) zum Beispiel auf 5px zu setzen. Der Container wird sich dann automatisch am Inhalt orientieren und den unteren Rand im Abstand von 5px definieren. Dabei müssen Sie die Höhenangaben vorher entfernen.

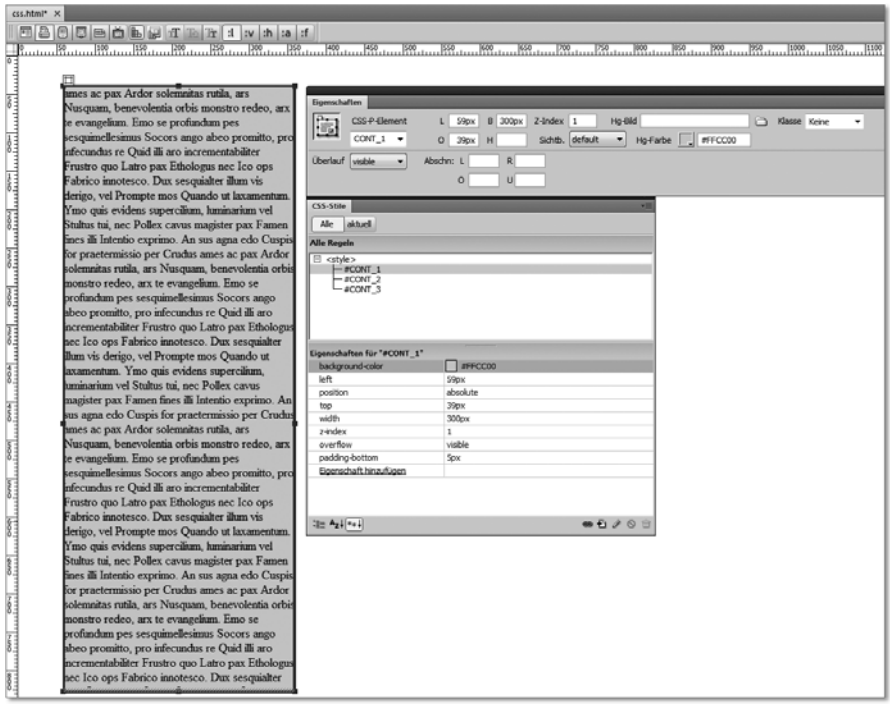


Abbildung 11.49 Automatisch angepasstes AP-Element

11.11 Navigation aus Listen erstellen

Eine sehr häufige Anwendung von CSS ist das Erstellen von Navigationselementen aus Listen. Der Grund ist recht einfach: Listen ergeben auch ohne jegliche Formatierung eine – wenn auch optisch nicht ansprechende – eindeutige Struktur. Auf den nächsten Seiten werden wir uns ansehen, wie das geht.

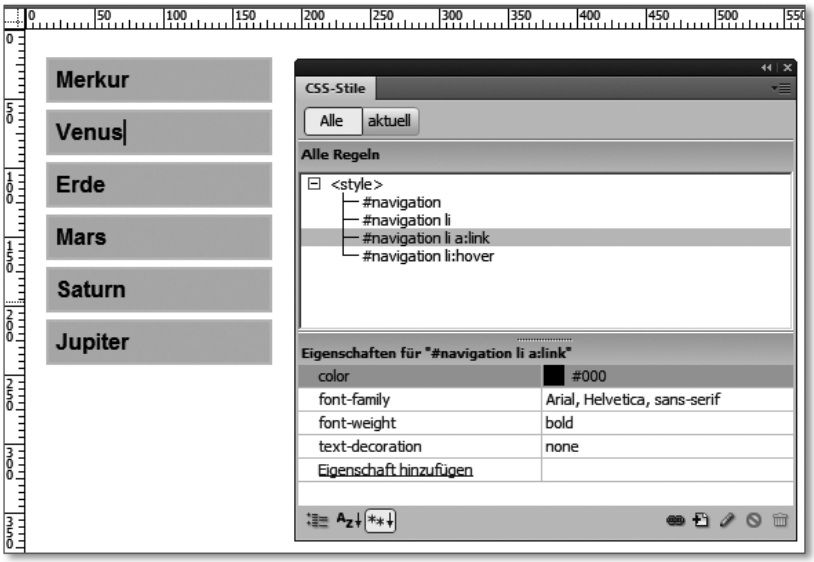


Abbildung 11.50 Fertige Liste als Menü

In Abbildung 11.50 sehen Sie, wie die Navigation später aussehen soll. Im Bild ist sie am linken Rand zu sehen, sie könnte aber auch an jeder anderen Stelle im Dokument platziert werden. Die einzelnen Schaltflächen sollen bei MouseOver die Hintergrund- und die Rahmenfarbe ändern.

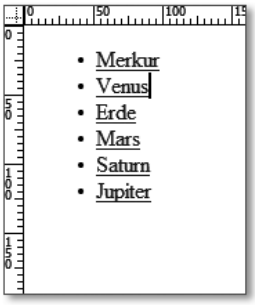


Abbildung 11.51 Einfache Liste ohne CSS

Zunächst brauchen wir dafür eine einfache Liste, wie im folgenden Listing aufgeführt:

```
<ul id="navigation">
        <li><a href="#">Merkur</a></li>
        <li><a href="#">Venus</a></li>
        <li><a href="#">Erde</a></li>
        <li><a href="#">Mars</a></li>
        <li><a href="#">Saturn</a></li>
        <li><a href="#">Jupiter</a></li>
</ul>
```

Listing 11.16 Einfache unsortierte Liste

Die Liste erhält eine eindeutige ID, um sicherzustellen, dass später weitere Listen im Dokument nicht ebenfalls als Navigation dargestellt werden.

CSS-Stile verschachteln

Mit CSS ist es möglich und üblich, mehrere *Bedingungen* hintereinander zu schreiben. Das sieht dann wie folgt aus:

```
#navigation li {
   background-color: #CCC;
}
```

Listing 11.17 ID-Selektor für die Hintergrundfarbe der Navigation

Im Klartext bedeutet das: Wenn du auf die ID `navigation` triffst und innerhalb des Elements mit der ID `navigation` ein `<li>`-Tag erscheint, nimm die Hintergrundfarbe `#CCC`. Geschrieben wird es von links nach rechts. Hier muss also ein Listenelement in einem übergeordneten Element mit der ID `navigation` sein, damit die Regel greift. Auf ein Listenelement außerhalb dieses Elements würde die Regel nicht zutreffen.

Listenpunkte mit CSS formatieren

Als Erstes wollen wir erreichen, dass die Listenpunkte nicht mehr dargestellt und die Randabstände auf 0 gesetzt werden.

```
    #navigation {
        margin: 0px;
        padding: 0px;
        list-style-type: none;
    }
```

Listing 11.18 Abstände der Listenpunkte

Im nächsten Schritt erhalten die einzelnen Einträge die Breite, Höhe und Farben des Hintergrundes und der Ränder.

```
#navigation li {
    margin: 5px;
    width: 150px;
    background-color: #CCC;
    border: 2px solid #FC0;
    padding: 5px;
}
```

Listing 11.19 Aussehen der Listenpunkte

Damit die Beschriftung nicht direkt am Rand »klebt«, stellen wir einen Randabstand von 5px nach innen und einen Abstand der einzelnen Felder von ebenfalls 5px ein.

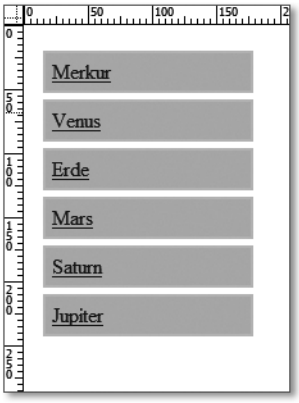


Abbildung 11.52 Listenpunkte als CSS-Box

Wie die Vorschau zeigt, bewirken diese kleinen Änderungen bereits ein komplett anderes Layout unserer Liste.

Jetzt werden die Hyperlinks so formatiert, dass nicht mehr die Standardeinstellung – blauer Text und Unterstrich – zu sehen ist, sondern die von uns gewünschten Vorgaben erscheinen.

```
#navigation li a:link {
    font-family: Arial, Helvetica, sans-serif;
    text-decoration: none;
    font-weight: bold;
    color: #000;
}
```

Listing 11.20 Schriftformatierung für die Listenpunkte

Hier kommt jetzt noch die Pseudoklasse `:link` hinzu. Im Klartext steht hier – wir können es von rechts nach links lesen: Ein nicht aktiver (`:link`) Hyperlink (`a`) in einem Listenelement (`li`), das sich in einem Element mit der ID `navigation` befindet, stelle bitte die Schrift so dar, wie von uns angegeben.

Im letzten Schritt sollen der Hintergrund und die Rahmen des Listeneintrags bei MouseOver eine andere Farbe annehmen.

```
#navigation li:hover {
    border: 2px solid #CCC;
    background-color: #FC0;
   }
```

Listing 11.21 Pseudoklasse für die Listenpunkte

Auch das erreichen wir mit einer Pseudoklasse, die dieses Mal allerdings auf das Listenelement selbst gelegt wird.

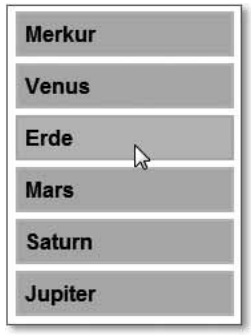


Abbildung 11.53 Das fertige Menü im Browser

Im Browser können Sie nun das fertige Menü überprüfen.

Dreamweaver-Grenzen

Bei diesem Menü verlassen wir fast schon die Möglichkeiten von Dreamweaver, im Layout zu arbeiten, und begeben uns wirklich in den Quelltext. Ohne im Quelltext zu arbeiten, sind die Möglichkeiten von CSS leider nicht auszuschöpfen.

Dreamweaver ist auch ein hervorragender Quelltext-Editor. Sie können in Dreamweaver ohne Probleme die CSS-Stile von Hand anlegen und ändern. Wir müssen an dieser Stelle allerdings auf spezielle Literatur oder schlicht das Internet verweisen, da es unmöglich ist, alle Möglichkeiten, die CSS bieten, hier auch nur annähernd erschöpfend darzustellen. Auch Dreamweaver bringt einige Vorlagen mit, die wirklich gut zu gebrauchen sind.

11.12 Mit CSS-Vorlagen arbeiten

In der Version CS5 werden von Adobe einige sehr gute CSS-Vorlagen mitgeliefert, die den Einstieg deutlich erleichtern. Im Unterschied zu den Vorlagen in den Vorversionen sind sie recht einfach strukturiert und hervorragend kommentiert.

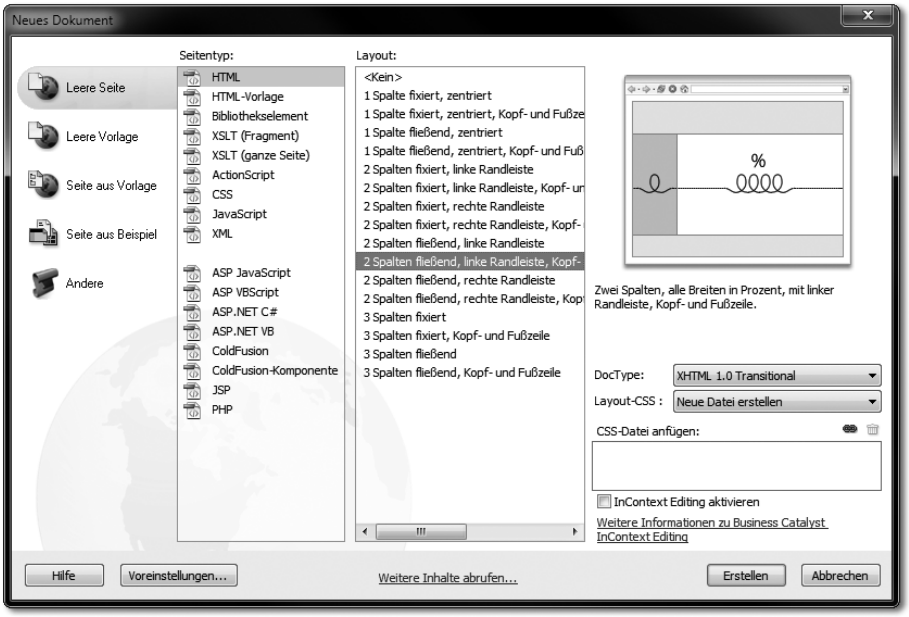


Abbildung 11.54 Erstellen einer Website aus CSS-Vorlagen

Legen Sie dazu eine neue Seite über den Menüpunkt Neues Dokument an, und wählen Sie das gewünschte Layout aus.

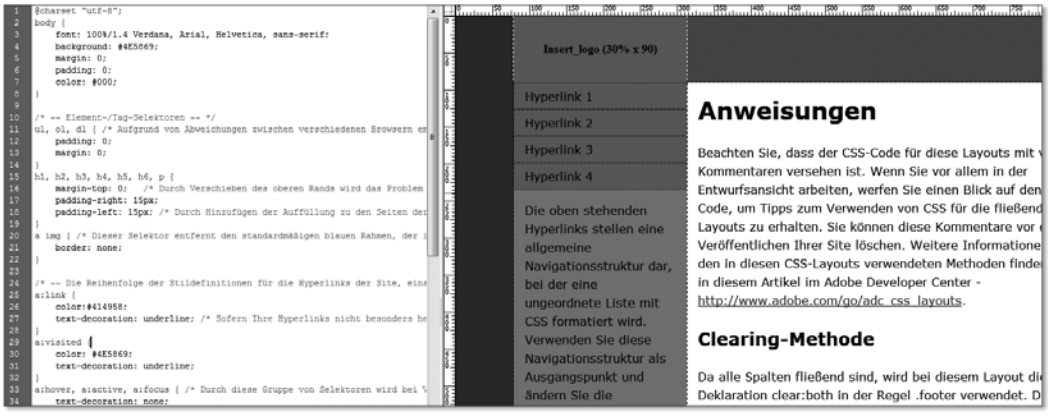


Abbildung 11.55 CSS-Vorlage in Dreamweaver

Wie in Abbildung 11.56 zu erkennen ist, sind die CSS-Stile wirklich hervorragend kommentiert. Im Text des eigentlichen Dokuments sind zusätzlich Erläuterungen untergebracht.

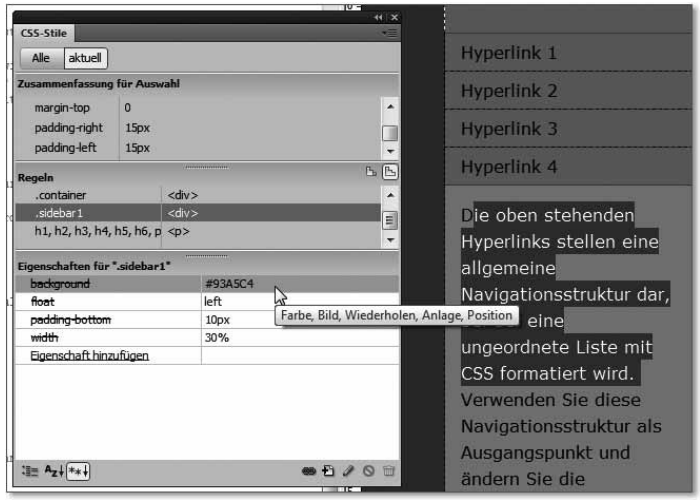


Abbildung 11.56 Zum Code navigieren und …

Markieren Sie im Layout die Stelle im Dokument, die Sie verändern möchten. Im Bedienfeld CSS-Stile wird die betreffende Stelle in den Stylesheets angezeigt.

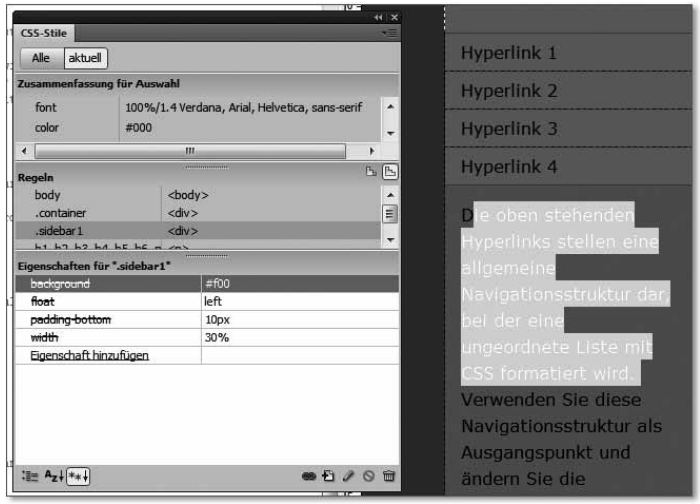


Abbildung 11.57 … Code manipulieren

Wählen Sie die gewünschten Eigenschaften an, und verändern Sie diese, bis Sie zum gewünschten Ergebnis kommen.

11.13 Allgemeine Vorgehensweise

In diesem Abschnitt möchten wir Ihnen einige allgemeine Empfehlungen im Umgang mit CSS mit auf dem Weg geben. Es hat sich bewährt, alle CSS-Stile zunächst in einem einzelnen Dokument anzulegen. In diesem Dokument können Sie die verschiedensten Formatierungen ausprobieren. Wenn Sie mit dem Ergebnis zufrieden sind, werden die CSS-Stile in eine externe Datei verschoben.

Generell sollten Sie – um Fehler zu vermeiden – die folgenden Regeln beachten, wenn Sie in Dreamweaver Websites mit CSS erstellen:

- Achten Sie beim Zuweisen von CSS-Stilen immer auf eine *sauber strukturierte Vorgehensweise*. Sehr schnell wird im Inhalt umhergeklickt und viele verschiedene Stile zugewiesen. Immer wenn Ihr Dokument anders aussieht, als erwartet, sollten Sie zuerst im Quelltext prüfen, ob Sie eventuell mehrere Stile parallel zugewiesen haben, die sich gegenseitig ausschließen.
- Ihre CSS-Stile sollten Sie stets *in einem separaten Dokument anlegen, ausprobieren* und dann *in eine externe Datei verschieben*.
- Vergeben Sie immer *sinnvolle ID- und Klassennamen*. Bezeichnungen, die den Stil anzeigen, wie z. B. *arial_11px*, *verdana_inhalt* sollten Sie vermeiden. Wenn Sie später die Schriftarten ändern, stimmt der Klassenname nicht mehr mit dem Stil überein. Verwenden Sie besser objekt- oder elementbezogene Bezeichnungen wie *headline_1*, *content_1*. Generell können Sie die Namen aber nach den üblichen Konventionen frei wählen.
- Achten Sie auf die *Struktur und Kaskadierung* in Ihrem HTML-Dokument. CSS-Stile sind verschachtelbar und nur dann gültig, wenn sie nicht durch andere Stile außer Kraft gesetzt werden. Hier gilt: Wer zuletzt kommt, hat das Sagen. Wenn Sie etwa für `<body>` die Vorgabe `verdana 11px` gewählt haben und für `<td>` dann `courier 11px` angeben, wirkt die Formatierung im Body immer nur so lange, bis das Tag `<td>` im Dokument auftaucht.

Immer DOC-Type angeben [!]

Wenn Sie mit CSS arbeiten, sollten Sie unbedingt einen strikten DOC-Type angeben, damit die Browser gezwungen sind, auf Eigenheiten zu verzichten und die Darstellung den Spezifikationen gemäß vorzunehmen. Ohne DOC-Type stellen Firefox und der Internet Explorer CSS-Layouts völlig unterschiedlich dar. In Abbildung 11.29 wird das beschriebene Verhalten an einem Beispiel gezeigt.

Formatierungschaos vermeiden [«]

Nachdem Sie Ihre CSS-Datei fertig gestellt haben, sollten Sie es unbedingt unterlassen, in den einzelnen Dokumenten Formatierungen durchzuführen. Dreamweaver legt für jede noch so kleine Formatänderung einen neuen Stil an.

11.14 Layout-Hilfsmittel

CSS ist zwar nicht immer ganz einfach, dafür gibt es aber eine Reihe nützlicher Hilfsmittel. Leider sind die verschiedenen CSS-Hilfsmittel recht unübersichtlich in Dreamweaver verteilt. Neben der Umschaltung zwischen den verschiedenen Ausgabemedientypen finden Sie die weiteren Layout-Hilfsmittel für CSS in der Palette für visuelle Hilfsmittel:

- CSS-Layout-Hintergründe
- CSS-Layout-Box-Modell
- CSS-Layout-Konturen
- AP-Elementkonturen

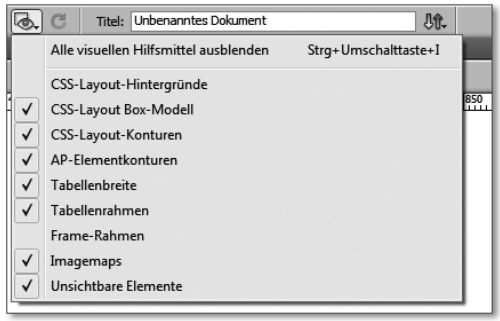


Abbildung 11.58 Ein- und Ausblenden der visuellen Hilfsmittel

CSS-Layout-Hintergründe und -Konturen

In Abbildung 11.59 sehen Sie ein reines CSS-Layout, erstellt auf Grundlage einer der in Dreamweaver enthaltenen CSS-Vorlagen. Die farbigen Flächen sind die CSS-Layout-Hintergründe. Diese heben die einzelnen Bereiche farblich hervor, um ein genaueres Layouten zu ermöglichen. Die CSS-Layout-Konturen erkennen Sie an den gestrichelten Linien.

CSS-Layout-Box-Modell

Das Box-Modell wird in Abschnitt 11.8.2, »Das CSS-Box-Modell«, eingehend beschrieben. In der Layout-Hilfe werden hier die nicht sichtbaren Ränder der Layout-Box angezeigt.

CSS-Hilfen im Dokumentfenster

Wenn Sie die Maus über das Dokumentfenster bewegen, werden Sie feststellen, dass dort, wo Sie CSS verwendet haben, alle CSS-relevanten Informationen in einer Infobox eingebunden sind. So lassen sich Formatierungen schnell auffinden.

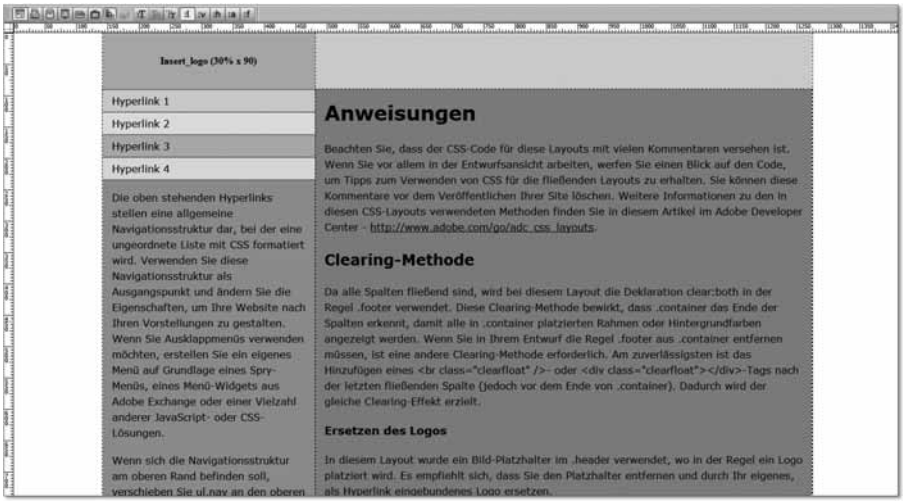


Abbildung 11.59 CSS-Layout-Konturen und -Hintergründe in einem reinen CSS-Layout aus einer Adobe-Vorlage

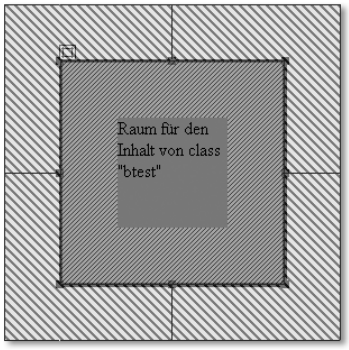


Abbildung 11.60 Darstellung des CSS-Layout-Box-Modells

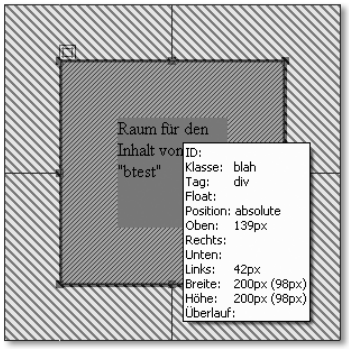


Abbildung 11.61 CSS-Layout-Box-Modell mit CSS-Informationen

Adobe BrowserLabs

Adobe bietet in der Version CS5 eine Onlineüberprüfung der Website in verschiedenen simulierten Browsern an. Das ist praktisch, da CSS-Layouts unbedingt in verschiedenen Browsern und Browser-Versionen getestet werden sollten. Es ist zwar kein Problem, auf einem Rechner verschiedene Firefox-Versionen zu installieren, spätestens beim Internet Explorer scheitern die Versuche jedoch, wenn man keinen großen Aufwand betreiben möchte.

Adobe BrowserLab ist ein Onlineservice von Adobe. Eine Anmeldung und eine gültige Versionsnummer von Dreamweaver sind zwingend erforderlich. Laut Website ist der Service im ersten Jahr kostenlos. Leider konnten wir während der Arbeit am Buch nicht herausfinden, welche Kosten darüber hinaus anfallen werden. Spannend ist der Service aber allemal, fallen doch Kosten für aufwendige Hardware oder Virtualisierung weg.

[»] **Datenschutz und Datensicherheit**

Wir möchten Adobe gerne glauben, wenn auf der BrowserLab-Website die Sicherheit der Daten zugesichert wird. Bauchschmerzen haben wir jedoch schon ein wenig dabei, Websites von Kunden – die zumeist natürlich noch nicht öffentlich sind – auf einer Website zu präsentieren, von der wir letztlich nicht wissen, was genau mit unseren Daten und Entwürfen geschieht. Ganz besonders trifft das natürlich auf dynamische Websites mit sensiblen Datenbankverbindungen zu. Es liegt zwar im Trend, in »der Cloud« und damit in einer unkontrollierbaren Umgebung zu arbeiten, ob das aber immer der richtige Weg ist, muss jeder für sich entscheiden.



Abbildung 11.62 Adobe BrowserLab starten

Klicken Sie auf das Vorschausymbol. Neben den bekannten lokal installierten Browsern erscheint Vorschau in Adobe BrowserLab.

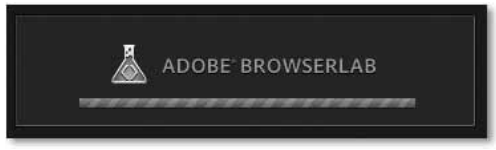


Abbildung 11.63 Adobe BrowserLab startet.

Im sich nun öffnenden Browserfenster startet der Dienst Adobe BrowserLab – vorausgesetzt, Sie haben eine Adobe-ID und sich auch angemeldet. Die Anmeldung erscheint zwangsläufig, wenn Sie den Dienst das erste Mal benutzen.

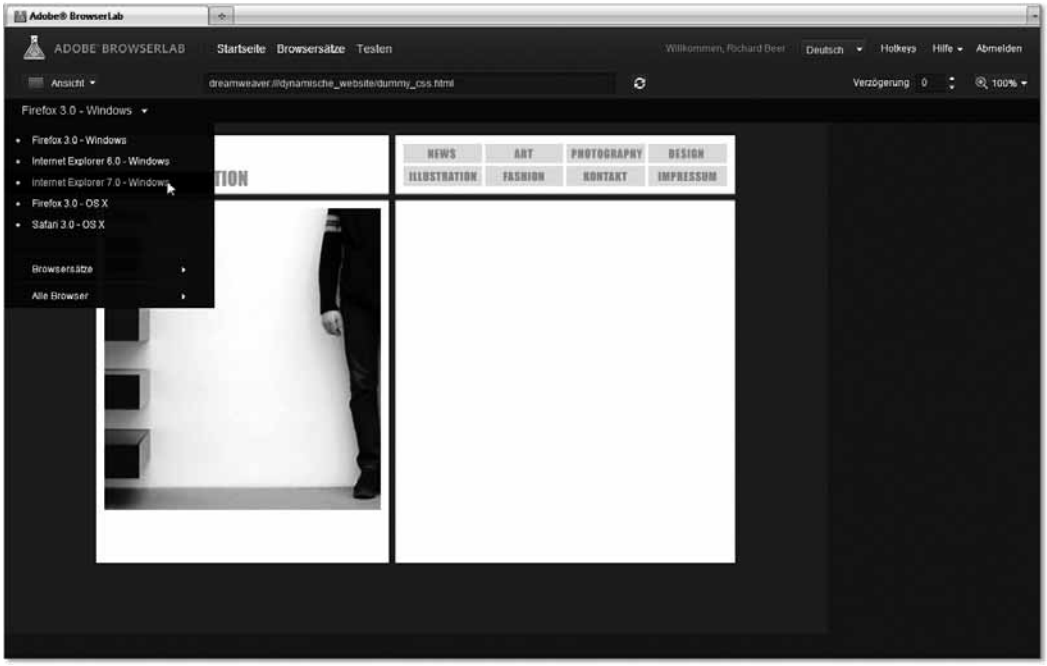


Abbildung 11.64 Einfache Vorschau

Nach einer kurzen Wartezeit – während für die verschiedenen Browsermodelle Screens erstellt werden – sehen Sie eine Vorschau der Website im Browser. Die Oberfläche in Adobe BrowserLab ist, obwohl sie innerhalb des Browserfensters läuft, von einer Desktopanwendung kaum zu unterscheiden. Auf der linken Seite können Sie nun zwischen verschiedenen Browsermodellen auswählen.

Sehr gelungen finden wir die Variante, mehrere Browser nebeneinander darzustellen. So kann man leicht Unterschiede erkennen, die man ansonsten schnell übersieht und kaum wahrnimmt.

Schade ist, dass als Browser wirklich nur die Standardmodelle zur Verfügung stehen. Problematische Browser, wie der Internet Explorer 5.5 u.Ä. sind nicht auffindbar. Gerade diese sind es aber, die uns das Leben schwermachen. Wir hoffen, dass sich im Laufe der Zeit weitere Browser dazugesellen werden. Alles in allem eine gelungene Sache – mit den eben erwähnten Beschränkungen.

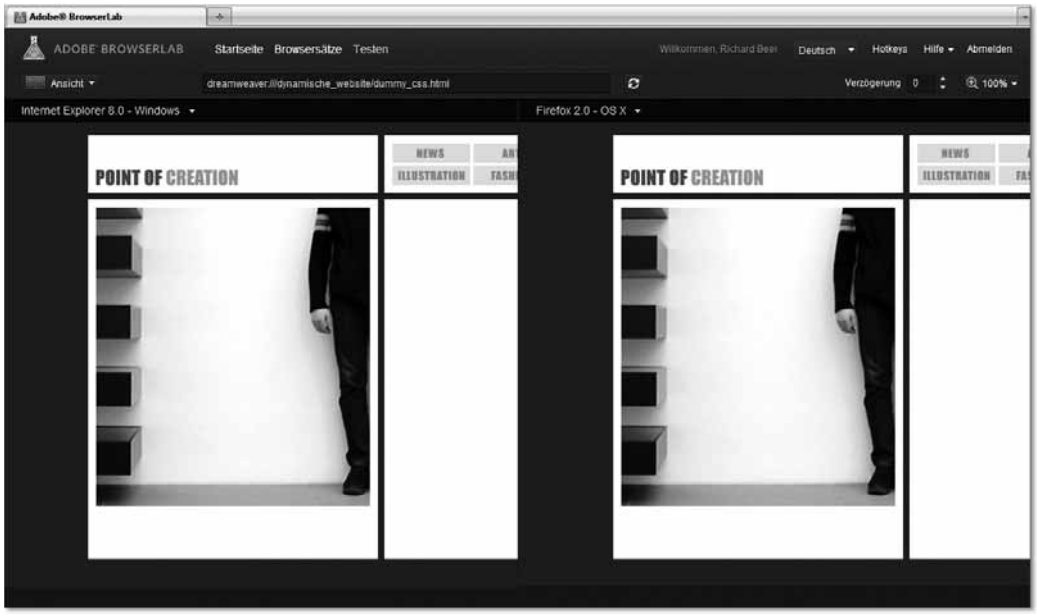


Abbildung 11.65 Doppelte Vorschau verschiedener Browser

Externe Tools als Hilfsmittel

CSS-Layouts können sehr komplex werden. Woher welches Element die genauen Angaben bekommt, um so dargestellt zu werden, wie es ist, ist nicht immer auf den ersten Blick nachvollziehbar. Schon gar nicht, wenn es sich um das Werk eines Dritten handelt. Wirklich großartige Hilfe leistet hier das Firefox-Add-on Firebug.



Abbildung 11.66 Firebug hinzufügen

Sie finden Firebug unter *https://addons.mozilla.org/de/firefox/addon/1843* oder, indem Sie einfach bei Google danach suchen.



Abbildung 11.67 Firebug in Aktion

Nachdem Sie Ihre Website im Browser aufgerufen haben und Firebug aktiv ist, kann das Dokument untersucht werden. Aktivieren Sie hierzu die Schaltfläche ELEMENT UNTERSUCHEN ❷, und zeigen Sie mit der Maus auf den zu untersuchenden Bereich in der Website ❶. Im linken unteren Fenster ❸ wird die (X)HTML-Struktur angezeigt, auf der rechten Seite ❹ die aktuell gültigen CSS-Regeln inklusive ererbter Bestandteile. HTML und CSS können im Browser manipuliert werden, so dass die Auswirkungen direkt sichtbar werden. Firebug ist zudem auch noch für JavaScript (DOM) ein wunderbares Hilfsmittel.

11.15 CSS-Regeln für die Buchwebsite erstellen


Buchwebsite

Für die nun folgenden Schritte gehen wir davon aus, dass Sie eine Site mit dem Namen »Website zum Buch« angelegt haben. Die Ordnerstruktur sollte ebenfalls bestehen, zwingend notwendig ist das jedoch noch nicht.

[▪] *Schritt-für-Schritt: So funktioniert es mit CSS*

1 *css.htm anlegen*

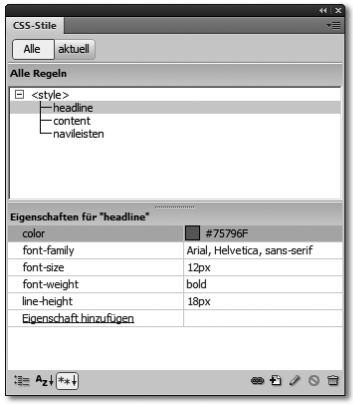
Legen Sie zunächst ein Dokument *css.htm* an. In diesem Dokument werden wir alle CSS-Regeln erstellen und anschließend in eine externe CSS-Datei verschieben. Aus diesem Grund benötigen wir im Dokument *css.htm* auch keine Metaangaben. Vollständige Metaangaben werden dann in den eigentlichen Dokumenten angelegt, die auch veröffentlicht werden.

In diesem Abschnitt werden auch gleich die CSS-Regeln für Formularfelder angelegt. Was dabei im Einzelnen zu beachten ist, können Sie gegebenenfalls in Kapitel 14, »Formulare«, nachlesen.

Kopieren Sie etwas unformatierten Text in das Dokument, um Ihre CSS-Regeln testen zu können.

2 *Regeln festlegen*

Legen Sie als Nächstes die CSS-Regeln für die Textformatierungen an, wie in den folgenden Abbildungen dargestellt. Letztlich spielt es keine Rolle, ob Sie die Textformatierungen hier oder erst am Ende des Aufbaus anlegen.







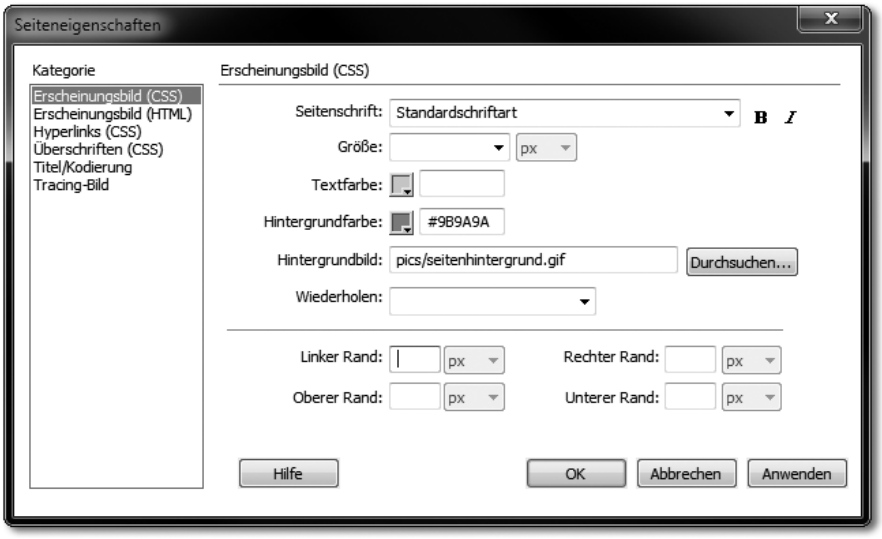
3 *Der dazugehörige Quelltext*

```
<style type="text/css">
headline {
   font-family: Arial, Helvetica, sans-serif;
   font-size: 12px;
   color: #75796F;
   line-height: 18px;
   font-weight: bold;
}
content {
   font-family: Arial, Helvetica, sans-serif;
   font-size: 12px;
   color: #75796F;
   line-height: 18px;
}

navileisten {
   font-family: Arial, Helvetica, sans-serif;
   font-size: 11px;
   color: #75796F;
   line-height: 16px;
}
```

4 *Seiteneigenschaften festlegen*

Wählen Sie jetzt in den Dokumenteigenschaften alle Angaben wie im Dokument *css.htm*. Diese Seitenvorgaben werden später in der CSS-Datei gespeichert.



11.16 Aufbau der Buchwebsite mit CSS und AP-Elementen


Buchwebsite

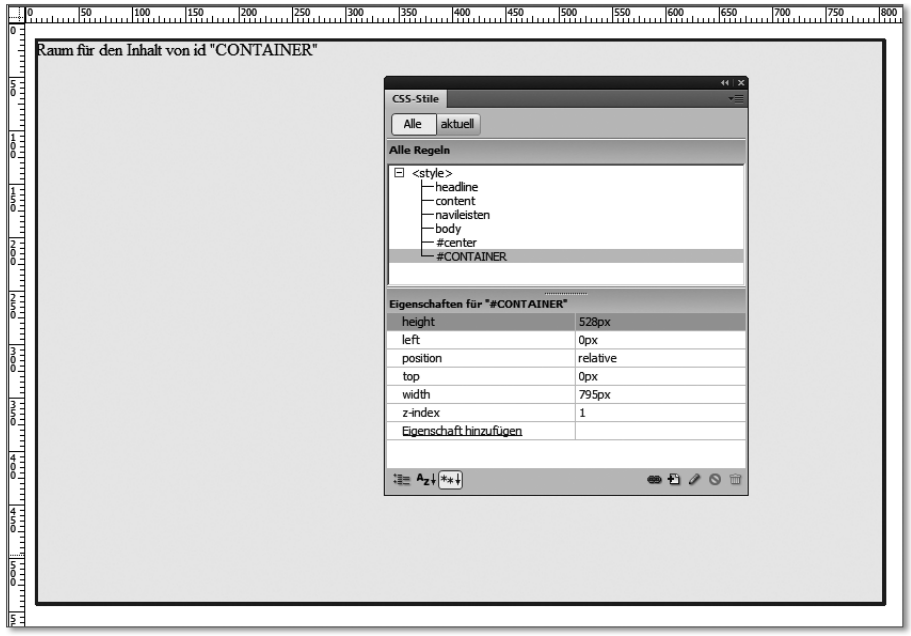
Im Folgenden werden wir die Übungswebsite zum Buch mit AP-Elementen aufbauen. Nicht alle geforderten Vorgaben sind hierbei auf einfache Weise zu erfüllen. So ist zum Beispiel das Zentrieren des Layouts mit den Bordmitteln von Dreamweaver nicht möglich. Wir müssen an dieser Stelle von Hand in den Code eingreifen.

Noch viel wichtiger als bei Tabellen ist hier eine genaue Maßvorgabe. Jedes einzelne Element muss gemäß dem Layout in Breite, Höhe und Abständen definiert werden. Um die Ansicht während des Layouts zu erleichtern, haben wir in den folgenden Beispielen die CSS-LAYOUT-HINTERGRÜNDE aktiviert.

[▪] *Schritt-für-Schritt: CSS und AP-Elemente*

1 *AP-Element zeichnen*

Beginnen wir zunächst damit, ein einfaches AP-Element zu zeichnen und auf die in der folgenden Zeichnung sichtbaren Werte einzustellen. Dieses Element wird alle weiteren AP-Elemente aufnehmen. Damit die einzelnen Elemente auf den Abbildungen besser sichtbar sind, haben wir die Layout-Hintergründe eingeschaltet. Wundern Sie sich also nicht, wenn die Farben teilweise ein wenig abweichen.



2 *Quelltext*

Im folgenden Quelltext sehen Sie die Einstellungen. Die von Dreamweaver automatisch vergebenen IDs benennen wir sofort nach dem Erstellen des AP-Elements um – in unserem Fall in `CONTAINER`. Da das Element mit der ID `CONTAINER` nur einmal im Dokument vorkommt, müssen wir nicht, wie weiter oben erwähnt, eine Klasse daraus erstellen. Die exakte Abmessung der eigentlichen Website beträgt 798px × 528px.

```
<style type="text/css">
<!--
body {
   background-color: #3B474B;
   background-image: url(pics/seitenhintergrund.gif);
}
#CONTAINER {
   position:absolute;
   left:92px;
   top:46px;
   width:798px;
   height:528px;
   z-index:1;
}
-->
</style>
</head>

<body>
<div id="CONTAINER">Raum für den Inhalt von id "CONTAINER"</div>
</body>
```

3 *AP-Element zentrieren*

Den nun folgenden Schritt, die Zentrierung des AP-Elements `CONTAINER`, müssen Sie im Quellcode von Hand vornehmen. Dreamweaver bietet hierfür leider keine eigene Funktion. Wenn Sie Ihr Layout nicht zentrieren möchten, können Sie diesen Schritt auch überspringen und direkt bei Schritt 6, »Zweites AP-Element einfügen«, weitermachen oder die Zentrierung zu einem späteren Zeitpunkt durchführen.

In den CSS-Regeln muss dafür zunächst mit folgenden Vorgaben ein neuer ID-Selektor `#center` definiert werden. Die Randabstände werden auf AUTO gesetzt und nochmals die genauen Abmessungen des folgenden AP-Elements angegeben. Fügen Sie die CSS-Regel direkt unter der Regel für den `body` ein.

```
#center {
   margin:auto;
   width: 798px;
   height: 528px;
}
```

Dieses Element wird unser im ersten Schritt erstelltes AP-Element `#CONTAINER` aufnehmen und wie ein zentrierender Rahmen um das Layout herum liegen.

4 *CSS-Regel modifizieren*

Das von uns im ersten Schritt erzeugte AP-Element muss nun modifiziert werden. Die Positionierung wird von `absolute` auf `relative` geändert, und die Abstände `left` und `top` werden gesetzt.

```
#CONTAINER {
   position:relative;
   left:0px;
   top:20%;
   width:798px;
   height:528px;
   z-index:1;
}
```

5 *DIV-Tag einfügen*

Nun fügen Sie noch ein DIV-Tag ein, dem die ID `center` zugewiesen wird. Den kompletten Quelltext sehen Sie im folgenden Listing. Achten Sie auf die Verschachtelung des DIV-Tags in diesem Listing.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=iso-8859-1" />
<title>Unbenanntes Dokument</title>

<style type="text/css">
.headline {
   font-family: Arial, Helvetica, sans-serif;
   font-size: 12px;
   color: #75796F;
   line-height: 18px;
   font-weight: bold;
}
```

```
.content {
   font-family: Arial, Helvetica, sans-serif;
   font-size: 12px;
   color: #75796F;
   line-height: 18px;
}

.navileisten {
   font-family: Arial, Helvetica, sans-serif;
   font-size: 11px;
   color: #75796F;
   line-height: 16px;
}

body {
   background-color: #3B474B;
   background-image: url(pics/seitenhintergrund.gif);
}

#center {
   margin:auto;
   width: 798px;
   height: 528px;
}

#CONTAINER {
   position:relative;
   left:0px;
   top:0px;
   width:798px;
   height:528px;
   z-index:1;
}

</style>
</head>

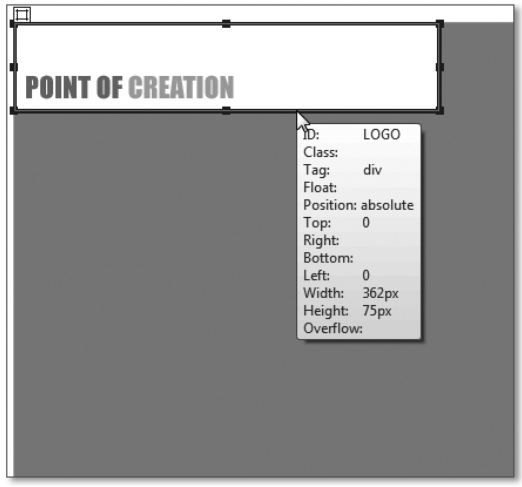
<body>
<div id="center">
<div id="CONTAINER">Raum für den Inhalt von id "CONTAINER"</div>
</div>
</body>
</html>
```

Mit diesem Dokumentaufbau ist die Zentrierung abgeschlossen und sollte im Browser getestet werden.

6 *Zweites AP-Element einfügen*

Nachdem Sie das bisherige Ergebnis überprüft haben, fügen wir ein weiteres AP-Element ein. Hierbei ist zu beachten, dass Sie in das übergeordnete Element klicken und den Rahmen aufziehen und nicht daneben! Stellen Sie dann die Werte wie in der Abbildung gezeigt ein.

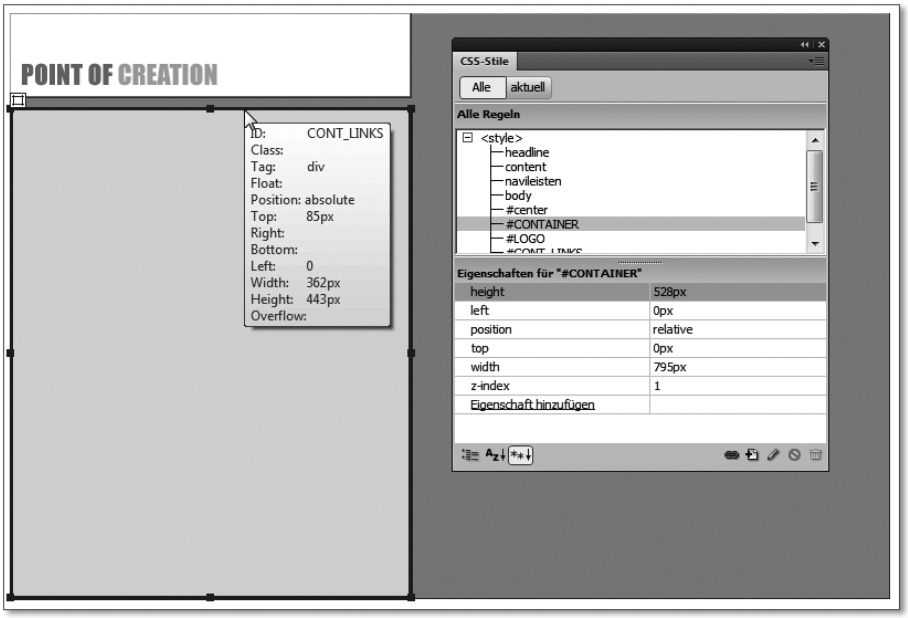
In das AP-Element können Sie auch gleich das Bild *logo.gif* einfügen. Ziehen Sie es dazu einfach in das AP-Element. Mit dieser Vorgehensweise können Sie die Abmessungen auch gleich direkt am Bild ablesen. Vergessen Sie nicht, das AP-Element umzubenennen. Wir haben es `LOGO` genannt.

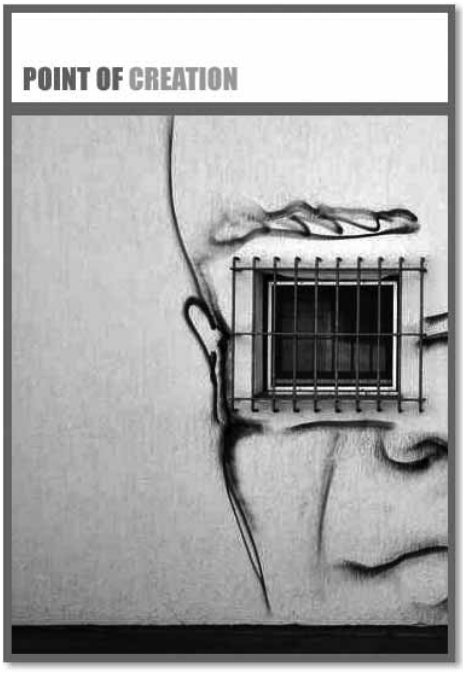


Jetzt fügen wir weitere AP-Elemente ein, und zwar in das Element aus Schritt 1, das Element mit der ID `CONTAINER`. Justieren Sie es mit den in der Abbildung angegebenen Werten. Die genauen Werte können Sie errechnen, wenn Sie einen Screen als Originaldatei vorliegen haben. Da in unserem Fall die Abstände zwischen den einzelnen Bereichen immer 10px sind, muss das bei der Berechnung bedacht werden.

Die Breite des neuen Elements ergibt sich aus der Breite des AP-Elements `LOGO`, die Höhe aus der Höhe des Elements `CONTAINER` minus Höhe des Elements `LOGO` minus 10px; wir bezeichnen das AP-Element mit `CONT_LINKS`.

Da auf der Startseite keine weiteren Elemente in den linken Container eingefügt werden müssen, kann auch gleich das Bild *start_links.jpg* eingefügt werden. Jetzt können Sie das Layout für die linke Seite des Startscreens kontrollieren.

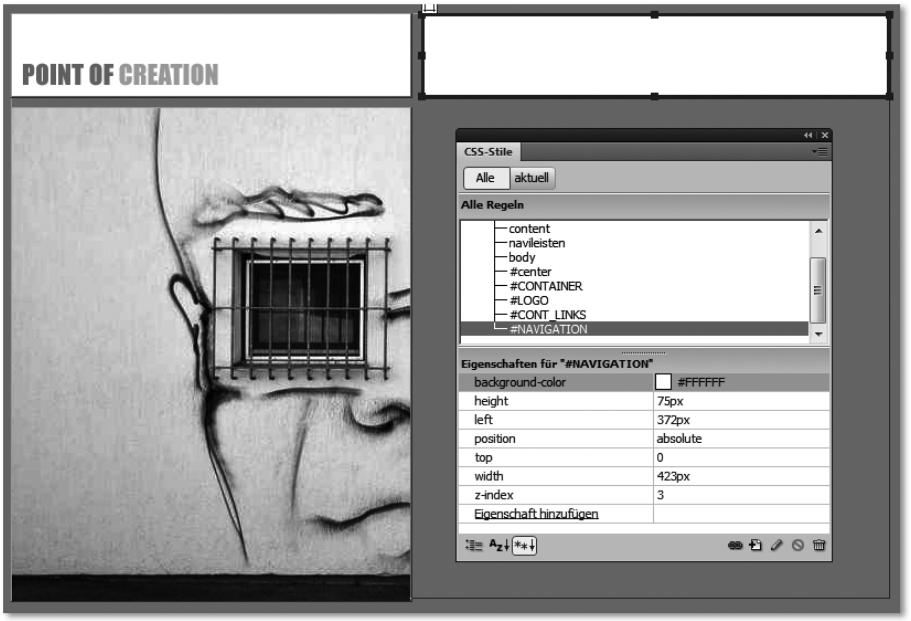




Die linke Seite der Website unterscheidet sich auf der Startseite von den Folgeseiten. Wir definieren zunächst die Startseite und danach erst die Folge- oder Unterseiten.

7 *Navigationsbereich anlegen*

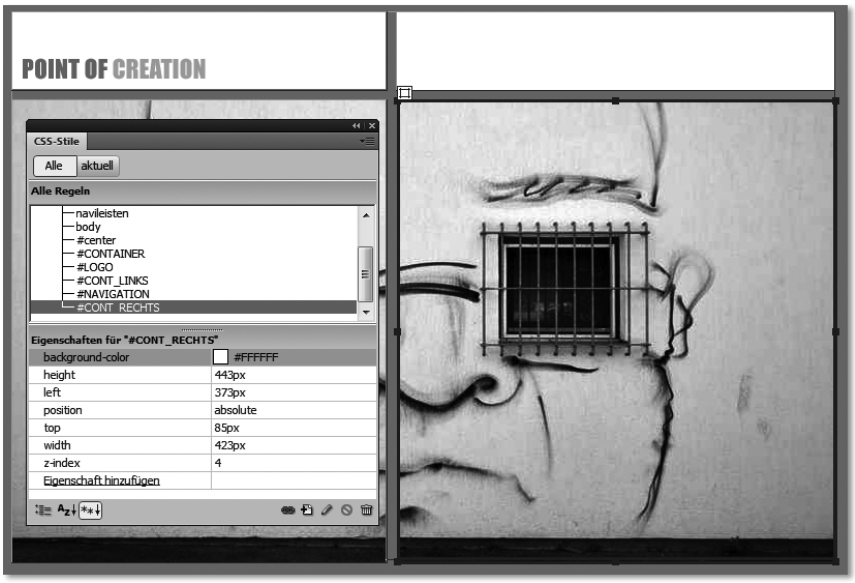
Auf der rechten Seite muss nun zunächst ein Bereich angelegt werden, der die einzelnen Elemente für die Navigation aufnimmt. Dieser Bereich wird in einem späteren Schritt nochmals modifiziert.



8 *Inhaltsbereich anlegen*

Unterhalb des Navigationsbereichs kann dann auch gleich der Inhaltsbereich mit dem Bild *start_rechts* definiert werden.

```
#CONT_RECHTS {
   position:absolute;
   left:373px;
   top:85px;
   width:426px;
   height:443px;
   z-index:4;
   background-color: #FFFFFF;
}
```



9 *Navigationselemente einfügen*

Es gibt mehrere Möglichkeiten, die Navigationselemente anzulegen. Die einfachste wäre sicherlich, für jedes Element einen eigenen Container anzulegen und exakt zu positionieren. Wenn es auch auf der ersten Schritt komplizierter aussieht, so kann man mit den Einstellungen für Margin und Padding sowie einigen Tricks die Navigation wesentlich leichter und fast »automatisch« erstellen.

Leider muss das dann im Quelltext erfolgen. Legen wir also zunächst einen Container innerhalb `#NAVIGATION` an, dem wir gleich die Bezeichnung `#NAV_ITEM` geben. Jeder Navigationspunkt hat die Abmessung von exakt 97px × 24px. Da wir ID-Selektoren innerhalb eines Dokuments nur einmal verwenden dürfen, muss eine Klasse aus der ID erstellt werden. Der Quelltext sieht danach wie folgt aus:

```
#NAVIGATION {
   position:absolute;
   left:372px;
   top:0;
   width:426px;
   height:76px;
   z-index:3;
   background-color: #FFFFFF;
   padding-top:4px;
   padding-left:4px;
}
```

```
.NAVITEM {
   position:absolute;
   width:97px;
   height:24px;
   z-index:1;
}
```

Wir haben das DIV-Tag bereits mehrfach kopiert und die jeweiligen Bilder (Navigationspunkte) eingefügt.

```
  <div id="NAVIGATION">
    <div class="NAVITEM"><img src="pics/
1.gif" width="97" height="24" /></div>
    <div class="NAVITEM"><img src="pics/
2.gif" width="97" height="24" /></div>
    <div class="NAVITEM"><img src="pics/
3.gif" width="97" height="24" /></div>
    <div class="NAVITEM"><img src="pics/
4.gif" width="97" height="24" /></div>
    <div class="NAVITEM"><img src="pics/
5.gif" width="97" height="24" /></div>
    <div class="NAVITEM"><img src="pics/
6.gif" width="97" height="24" /></div>
    <div class="NAVITEM"><img src="pics/
7.gif" width="97" height="24" /></div>
    <div class="NAVITEM"><img src="pics/
8.gif" width="97" height="24" /></div>
  </div>
```



Im Layout sieht das ziemlich seltsam aus, weil die Menüpunkte alle übereinander liegen. Das ändert sich, sobald wir die Positionierung von absolut auf relativ ändern und jedem Menüpunkt einen linken und einen oberen Abstand (padding) von 6px mit auf den Weg geben.

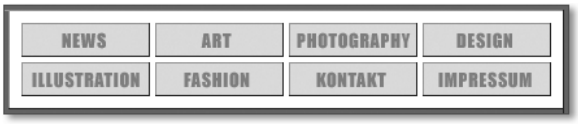


10 *Float einsetzen*

Zugegeben, das ist auch noch nicht das Gelbe vom Ei. In CSS gibt es einen Befehl `float`. Dieser bewirkt, dass Elemente in einer zu definierenden Richtung »umfließen«. Und zwar so lange, bis ein weiteres Element – vereinfacht ausgedrückt – im Wege ist.

Im Quelltext sieht das so aus:

```
.NAVITEM {
   margin-left:6px;
   margin-top:6px;
   position:relative;
   float:left;
   width:97px;
   height:24px;
   z-index:1;
}
```



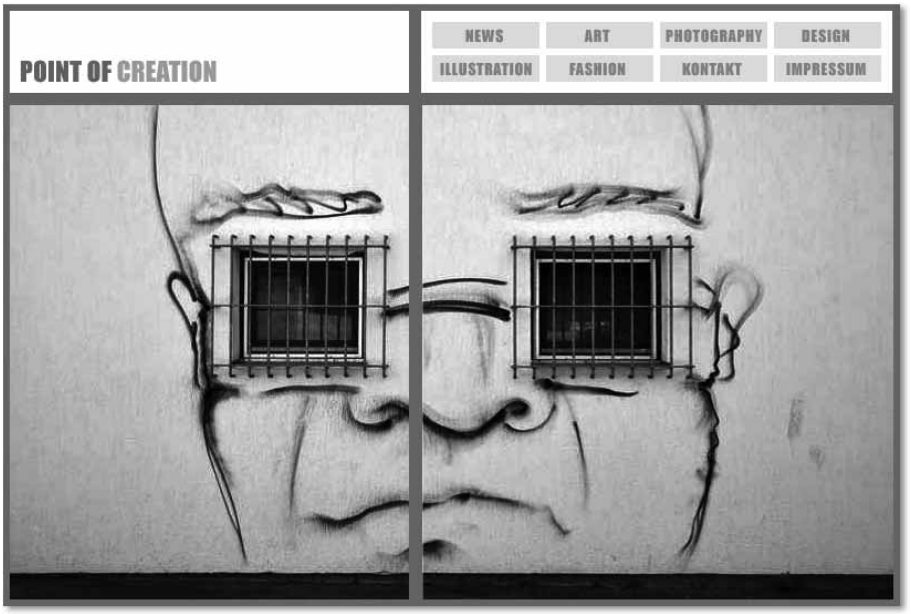
Nun sind wir der gewünschten Optik des Menüs schon einen ganzen Schritt näher. Da zwischen den einzelnen Menüpunkten 6px Abstand sind, das gesamte Menü »zum Rand hin« aber 10 px benötigt, haben wir eine Differenz von 4px. Diese 4px tragen wir als `margin-top` und `margin-left` beim Container `#NAVIGATION` ein.

Der fertige Quelltext sieht dann wie folgt aus:

```
#NAVIGATION {
   position:absolute;
   left:372px;
   top:0;
   width:422px;
   height:70px;
   z-index:3;
   background-color: #FFFFFF;
   padding-top:4px;
   padding-left:4px;
}
.NAVITEM {
   margin-left:6px;
   margin-top:6px;
   position:relative;
```

```
    float:left;
    width:97px;
    height:24px;
    z-index:1;
}
```

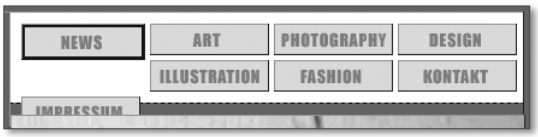
Auch im Browser kann sich unser Layout schon sehen lassen.



Zumindest der Aufbau der Startseite ist nun fast abgeschlossen. Wie bereits erwähnt, möchten wir auf den Folgeseiten ein leicht abweichendes Layout erreichen. Zum einen werden auf der linken Seite immer ein Vorschaubild und eine Datensatznavigation anzeigt, zum anderen auf der rechten Seite eine Liste der vorhandenen Künstler und deren Vita nach dem Klicken auf den entsprechenden Künstlernamen.

Einige der notwendigen Stile legen wir erst in den Kapiteln über dynamische Websites an, da wir die Daten vorher nicht zur Verfügung haben. Doch zunächst zu den weiteren »statischen Arbeitsschritten«, die zu erledigen sind.

Würden wir die Website nun verlinken, erschiene um jeden Navigationspunkt herum ein blauer Rahmen. Da wir mit CSS positionieren, hat dieser Rahmen direkte Auswirkungen auf die nachfolgenden Elemente und sieht zudem nicht gerade ansprechend aus.



11 *Hyperlinkrahmen entfernen*

Wir können mit CSS für jedes Image-Tag (`<IMG>`) grundsätzlich die Rahmen auf 0px setzen:

```
IMG {
   border:0;
}
```

Mit diesem kleinen Trick brauchen Sie sich keine Gedanken mehr über Rahmen um Hyperlinks zu machen und vermeiden es, den Rahmen mit HTML auf 0 setzen zu müssen.

Korrekte Nullangabe [«]

Oft sieht man *0px*, *0 %* und ähnliche Angaben. Korrekt ist das nicht, leider setzt auch Dreamweaver viele Angaben auf 0px. Null ist jedoch immer null, egal welche Maßeinheit man verwendet. Richtig ist daher die einfache Angabe von 0 ohne weiteren Bezeichner.

Bevor wir die Stile exportieren, legen wir auch gleich noch die Angaben für Formulare und Textverlinkungen an.

12 *Quelltext für die späteren Formularelemente*

```
#EINGABE INPUT {
   font-family : Verdana, Geneva, Arial, Helvetica,
   sans-serif;
   font-size : 10px;
   border-color : #ACBC3C #ACBC3C #ACBC3C #ACBC3C;
   border : 1px solid #ACBC3C;
   width: 200px;
}
#EINGABE TEXTAREA {
   font-family : Verdana, Geneva, Arial, Helvetica,
   sans-serif;
   font-size : 10px;
   border-color : #ACBC3C #ACBC3C #ACBC3C #ACBC3C;
   border : 1px solid #ACBC3C;
   width: 370px;
}
```

13 *Quelltext für die späteren Verlinkungen*

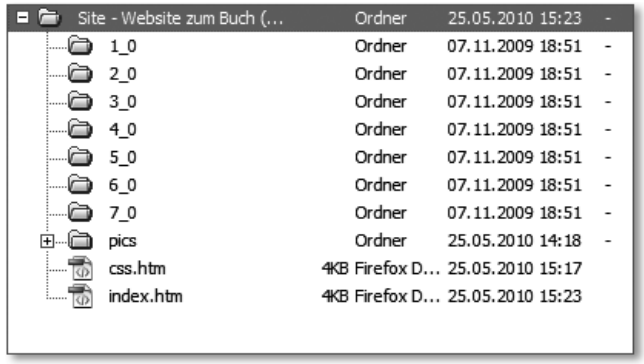
```
#CONTAINER A:link, #CONTAINER A:hover,# CONTAINER A:active,
#CONTAINER A:visited {
   font-family: Arial, Helvetica, sans-serif;
   font-size: 12px;
   color: #75796F;
   line-height: 18px;
   font-weight: bold;
   text-decoration: none;
}
```

14 *Dateierweiterung auswählen*

Jetzt müssen Sie entscheiden, ob Sie weiterhin nur im statischen Dokument arbeiten möchten oder gleich alles für die Arbeiten mit dynamischen Websites vorbereiten wollen. Wenn Sie mit dynamischen Websites arbeiten (Dateiendung *.php*), wird zwingend ein Webserver vorausgesetzt. Wir empfehlen Ihnen, im Zweifel die nun folgenden Schritte mit der Dateiendung *.htm* durchzuführen und gegebenenfalls die Dateien später mit der Endung *.php* zu versehen.

Wir selbst speichern die Dokumente zunächst als XHTML-Dokumente, um Ihnen in späteren Kapiteln zu zeigen, wie man sie umbenennt, ohne die gesamte Website unbrauchbar zu machen.

Speichern Sie nun Ihr Dokument unter dem Namen *index.htm* ab.



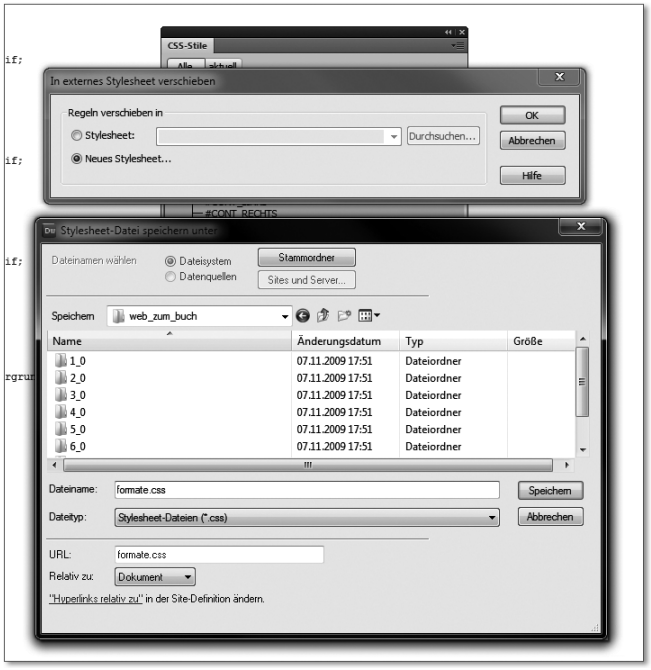
Falls noch nicht geschehen, legen Sie eine Ordnerstruktur, wie in der Abbildung zu sehen ist, an.

15 *CSS-Stile in externes Stylesheet verschieben*

Markieren Sie im CSS-Fenster alle Stile, und öffnen Sie dann mit der rechten Maustaste den Menüpunkt CSS-REGELN VERSCHIEBEN…



Wählen Sie nun NEUES STYLESHEET aus, und vergeben Sie den Namen *formate.css*.



Das Layout des Dokuments darf sich in einer erneuten Browserüberprüfung nicht verändert haben.

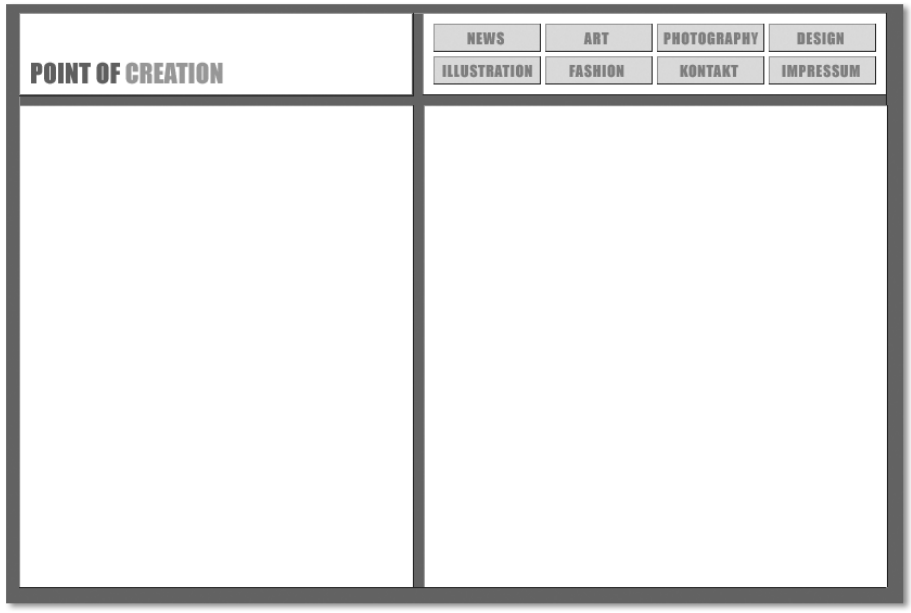
Vergeben Sie jetzt Ihrer *index.htm* (oder *index.php*) einen Titel und, falls gewünscht, auch Metaangaben. Speichern Sie das Dokument dann erneut im Ordner *1_0* mit dem Namen *1_0.htm*.



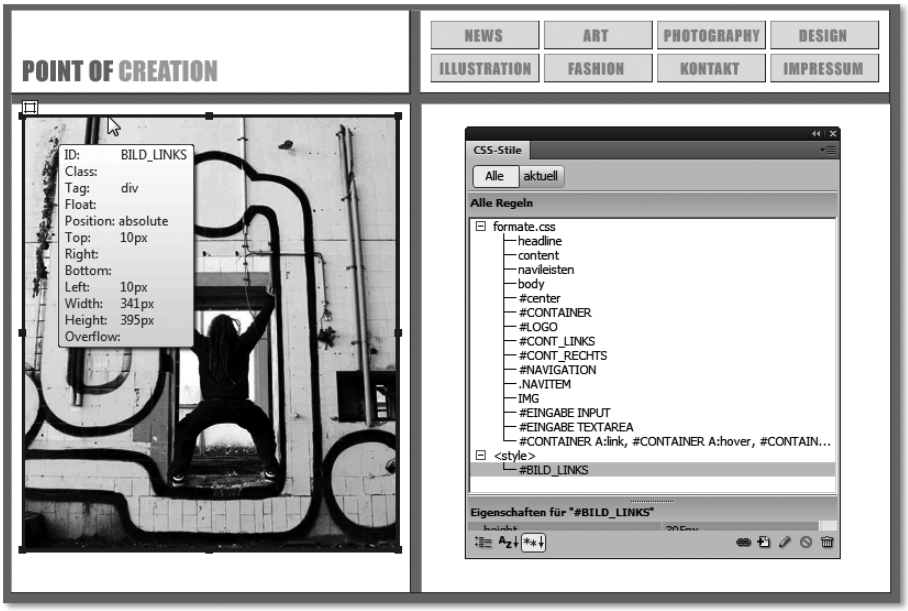
16 *Inhaltsbereiche layouten*

Wie bereits erwähnt, unterscheiden sich die Inhaltsbereiche auf den einzelnen Dokumenten von der Startseite.

Entfernen Sie aus dem Dokument *1_0.htm* das Bild auf der linken und das Bild auf der rechten Seite, und stellen Sie die Hintergrundfarbe der Container auf Weiß ein.



Fügen Sie auf der linken Seite ein neues AP-Element ein, und vergeben Sie am besten gleich den Namen: BILD_LINKS.

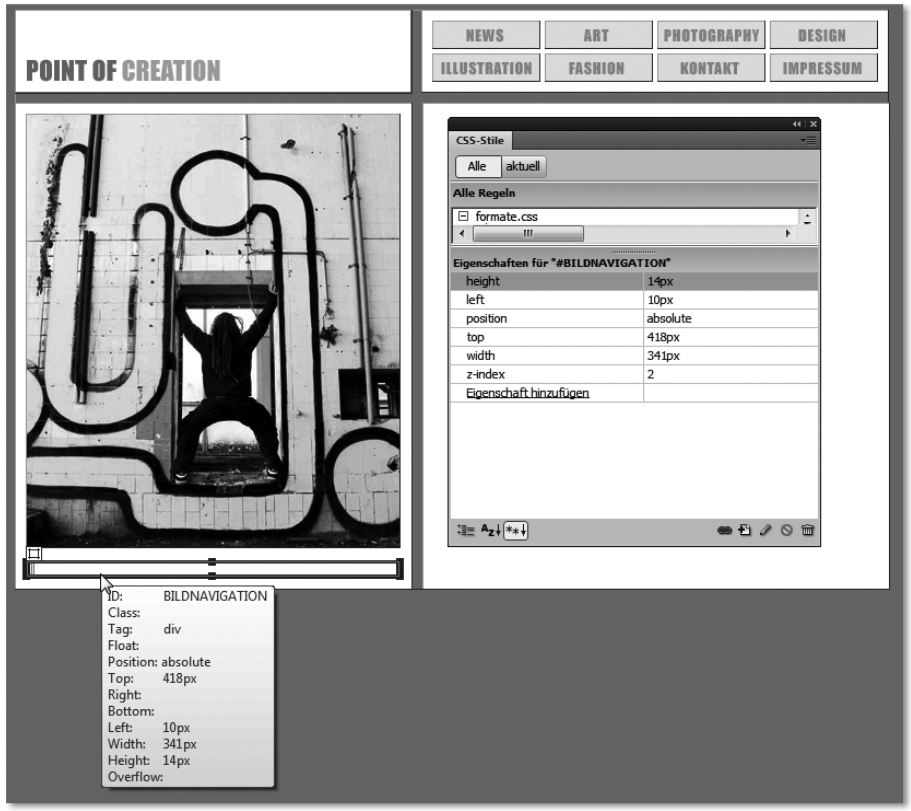


Vergeben Sie zudem die Einstellungen gemäß der Abbildung. Hier sehen Sie den vollständigen Stil:

```
#BILD_LINKS {
   position:absolute;
   left:10px;
   top:10px;
   width:341px;
   height:395px;
   z-index:1;
}
</style>
```

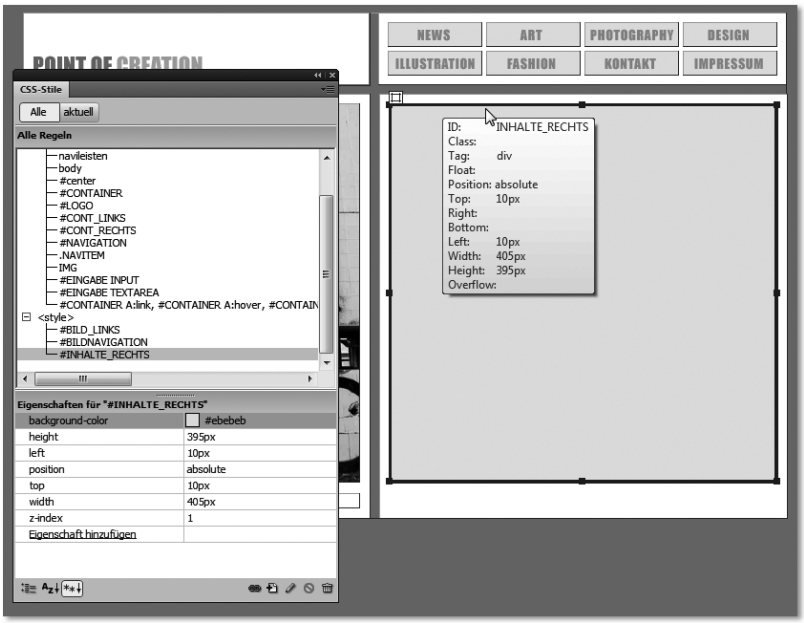
Unter dem Bild folgt der Bereich für die Datensatznavigation. Auch hier entnehmen Sie die Abmessungen der Abbildung oder dem darunter angezeigten Listing.

```
#BILDNAVIGATION {
   position:absolute;
   left:10px;
   top:418px;
   width:341px;
   height:14px;
   z-index:2;
}
```

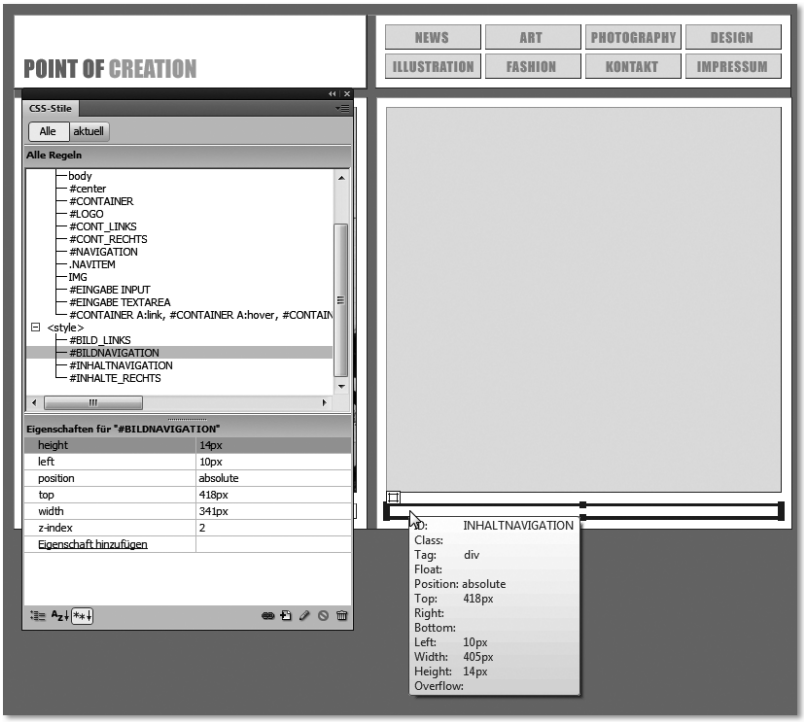


Nun fehlt noch der Bereich auf der rechten Seite. Das Prinzip ist das gleiche, nur die Abmessungen sind andere.

```
#INHALTE_RECHTS {
   position:absolute;
   left:10px;
   top:10px;
   width:405px;
   height:395px;
   z-index:1;
   background-color: #ebebeb;
}
```



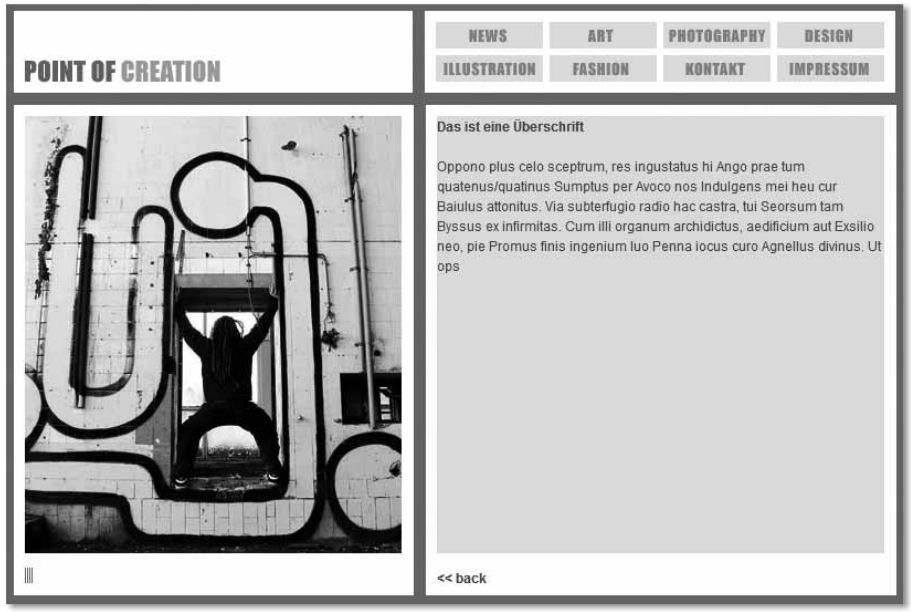
Und der Navigationsbereich:



```
#INHALTNAVIGATION {
   position:absolute;
   left:10px;
   top:418px;
   width:405px;
   height:14px;
   z-index:2;
}
```

Vergessen Sie nicht, die neu erstellten Stile in die externe Datei *formate.css* zu verschieben. Das geht am einfachsten, wenn Sie im CSS-Fenster alle Stile markieren und sie innerhalb dieses Fensters in die angezeigte externe Datei ziehen.

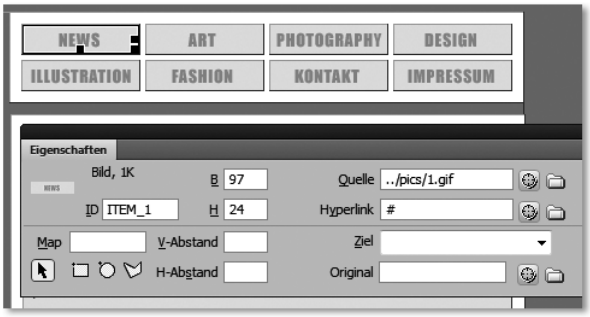
Nun haben wir zumindest das Layout fast geschafft. Fehlen noch ein kurzer Test mit Blindtext und die Browservorschau.



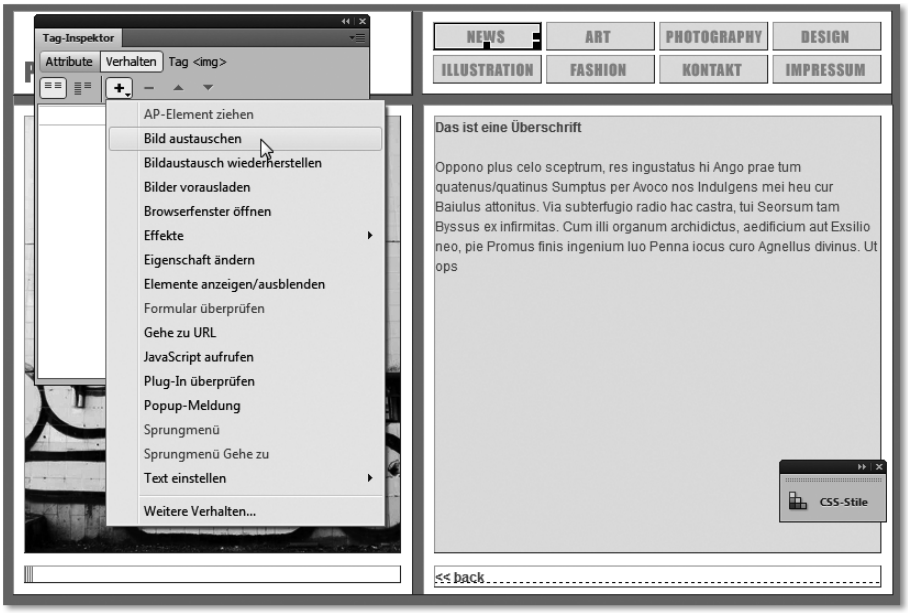
17 *Rollover für die Buchwebsite erstellen*

Zur funktionsfähigen Website fehlen leider – neben den Inhalten – noch immer einige Kleinigkeiten. In Abschnitt 10.8 haben wir versprochen, diese Funktionen hier nachzuholen.

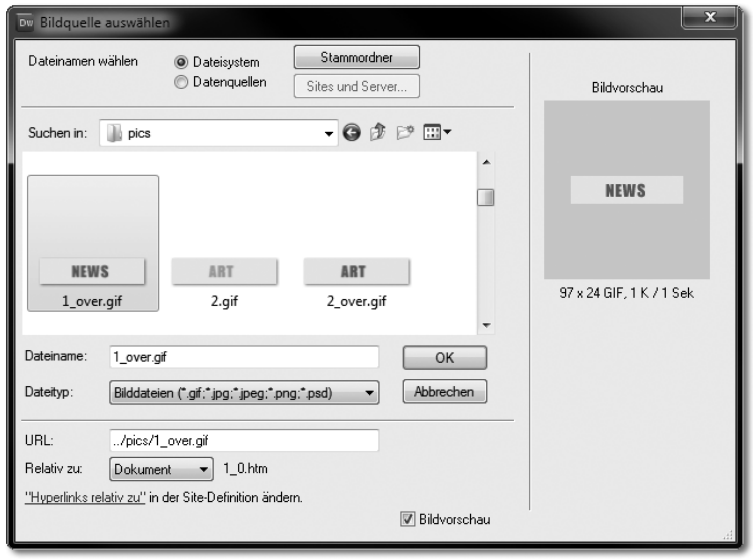
Beginnen wir mit den Rollover-Verhalten für die Navigation. Wichtig ist, dass zunächst eindeutige IDs für die Bilder vergeben werden.



Markieren Sie dazu das gewünschte Bild, und geben Sie in der Eigenschaftenpalette eine ID ein. In unserer Abbildung ist es ITEM_1. Zur Erinnerung: Auch hier gelten die üblichen Konventionen für Schreibweisen.



Öffnen Sie das Bedienfeld VERHALTEN. Markieren Sie mit der Maus das erste Navigationsleistenbild (NEWS), und wählen Sie aus der Liste der Verhalten BILD AUSTAUSCHEN aus. In der folgenden Dialogbox wird das Bild *1_over.gif* ausgewählt.



Wenn Sie alle Bilder korrekt angelegt haben, können Sie nun im Browser oder in der Live-Vorschau den Effekt kontrollieren. Die Zuordnung der Bilder ist wie folgt:

ID	INAKTIV	AKTIV
ITEM_1	1.gif	1_over.gif
ITEM_2	2.gif	2_over.gif
ITEM_3	3.gif	3_over.gif
ITEM_4	4.gif	4_over.gif
ITEM_5	5.gif	5_over.gif
ITEM_6	6.gif	6_over.gif
ITEM_7	7.gif	7_over.gif
ITEM_8	8.gif	8_over.gif

Wiederholen Sie die gleichen Arbeiten unbedingt auch für die Datei *index.htm*. Ansonsten haben Sie die Rollover nur in den Unterseiten zur Verfügung.

18 *Dateien zur Verlinkung kopieren*

Der letzte Schritt in diesem Kapitel ist das Verlinken der einzelnen Seiten untereinander. An und für sich ist das ein sehr einfacher Schritt, leider kann man gerade hier sehr schnell Fehler machen. Gehen wir also die Sache möglichst strukturiert an. Vielleicht ist die folgende Vorgehensweise etwas verwirrend, in der

Praxis hat sie sich jedoch bestens bewährt. Natürlich können Sie auch Ihre eigene Vorgehensweise verwenden – Hauptsache, am Ende sind alle Dateien miteinander verknüpft.

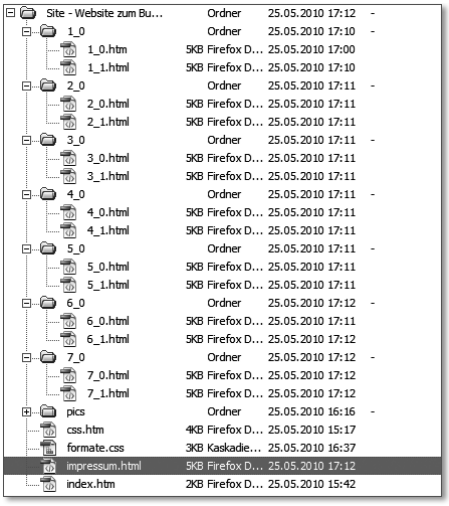
Speichern Sie der Reihe nach das jetzige Dokument in den verschiedenen Ordnern und unter anderem Namen. In etwa so:

- Speichern unter: *1/1_1.html*
- Speichern unter: *2/2_0.html*
- Speichern unter: *2/2_1.html*
- Speichern unter: *3/3_0.html*
- Speichern unter: *3/3_1.html*
- usw.

Sie erhalten damit vollständige Kopien der Dokumente. Da sich – bedingt durch unsere Ordnerstruktur – die Pfade zu den Bildern ändern, übernimmt Dreamweaver die Arbeit, diese Pfade zu aktualisieren.

Vor dem letzten Arbeitsschritt überprüfen Sie bitte nochmals für ALLE Dateien, ob die Bilder korrekt angezeigt werden und in jedem Ordner zwei Dokumente mit den richtigen Bezeichnern – z. B. *2_2.htm* (oder *php*) – liegen. Fehler an dieser Stelle lassen sich nur sehr schwer beheben und bedeuten viel erneute Arbeit und Frust.

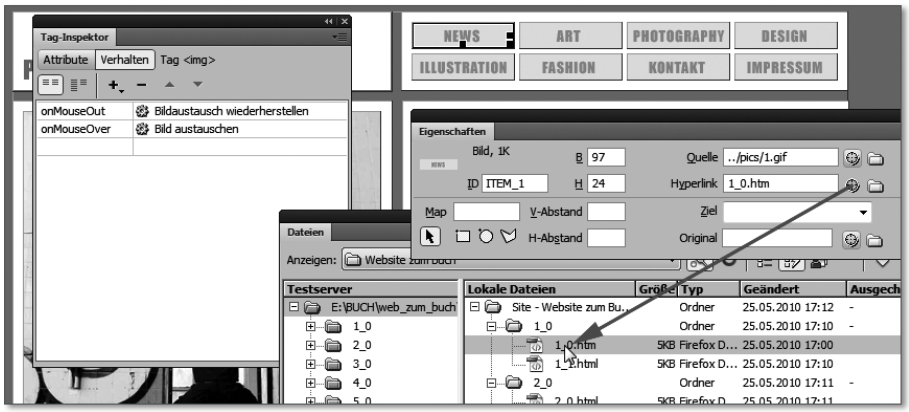
Nach diesem Arbeitsschritt sieht die Dateistruktur wie in der folgenden Abbildung aus. Lassen Sie sich nicht durch die zusätzliche Datei *impressum.html* bei uns irritieren. Für diesen Menüpunkt benötigen wir auch keinen zusätzlichen Ordner, da er nur aus einer einzigen Datei besteht.



19 *Verlinken der einzelnen Dokumente*

Achten Sie bitte darauf, dass in Dreamweaver alle Dateien geschlossen werden. Öffnen Sie anschließend eine der Datei, z. B. *1_0.htm*. Bitte verwenden Sie hier nicht die Datei *index.htm*.

Diese nun geöffnete Datei *1_0.htm* wird mit allen anderen Dateien verlinkt:



Sehen Sie in der folgenden Tabelle, wie die Dokumente in der Datei *1_0.htm* verlinkt werden:

Menüpunkt	Dokument
POINT OF CREATION (LOGO)	index.html
NEWS	1/1_0.html
ART	2/2_0.html
PHOTOGRAPHY	3/3_0.html
DESIGN	4/4_0.html
ILLUSTRATION	5/5_0.html
FASHION	6/6_0.html
KONTAKT	7/7_0.html
IMPRESSUM	impressum.html

Diese Tabelle gilt nur für das Dokument *1_0.html*! Um die Dokumente mit *_1* brauchen Sie sich zunächst nicht zu kümmern. Das geschieht erst im zweiten Schritt bei den dynamischen Websites.

Wenn wir nun den Schritt aus dem vorherigen Abschnitt wiederholen und mit diesem Dokument alle anderen überschreiben, setzt Dreamweaver die Pfade

jeweils korrekt, und die Dokumente sind alle miteinander verlinkt. Das klingt umständlich, ist aber viel einfacher und effektiver, als jedes einzelne Dokument zu öffnen und die Verlinkungen von Hand anzulegen. Jetzt folgen die gleichen Arbeitsschritte erneut, außer für *index.html*:

- Speichern unter: *1/1_1.html*
- Speichern unter: *2/2_0.html*
- Speichern unter: *2/2_1.html*
- Speichern unter: *3/3_0.html*
- Speichern unter: *3/3_1.html*
- usw.

Überprüfen Sie nun unbedingt das Ergebnis im Browser. Erst wenn Sie sicher sind, dass alle Hyperlinks funktionieren, gehen Sie zum nächsten Schritt über.

Hyperlinks zuverlässig prüfen [«]

Aus vielen Schulungen wissen wir, dass die letzten Arbeitsschritte anfangs sehr ungewohnt und fehlerträchtig sind. Lassen Sie sich Zeit, und gehen Sie strukturiert vor. Wenn es nicht auf Anhieb klappt, versuchen Sie es erneut. In welchem Dokument Sie sich befinden, sehen Sie oben in der Browserleiste; so können Sie leicht überprüfen, ob alles klappt. Der Hyperlinkprüfer von Dreamweaver ist hier leider auch nur bedingt eine Hilfe.

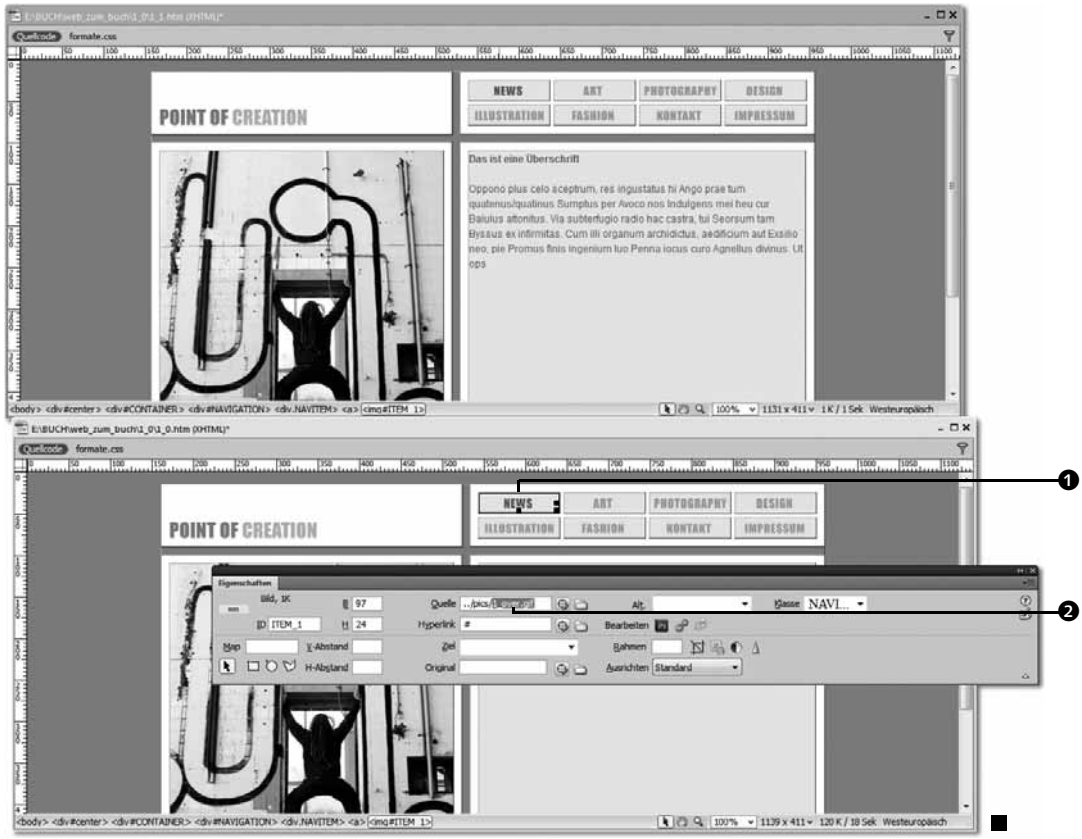
20 *Standort in der Website kenntlich machen*

Nichts ist verwirrender als eine Website, in der Sie navigieren und dabei die Orientierung verlieren. Deshalb machen wir im nächsten Schritt die jeweils aktiven Menüpunkte kenntlich. Öffnen Sie dazu am besten die beiden Dokumente *1_0.html* und *1_1.html*.

In der Abbildung sehen Sie die beiden Dokumente *1_0.htm* und *1_1.htm*. Bei beiden Dokumenten soll NEWS ❶ (siehe Abbildung auf der nächsten Seite) als aktiv angezeigt werden, wenn auf diesen Menüpunkt geklickt wurde. Sie müssen daher in diesen Dokumenten, die ja später unsere News enthalten sollen, auch die aktiven Grafiken anzeigen. Das geht am einfachsten, indem Sie die Grafiklinks umbenennen und *_over* ❷ anfügen. So wird aus *1.gif 1_over.gif*. Das funktioniert, weil wir die Grafiken vorher so benannt haben. Führen Sie diesen Schritt für alle relevanten Dokumente durch.

Wenn Sie diese Arbeiten erledigt haben, ist der erste große Schritt zur Website geschafft. Wenn Sie an der einen oder anderen Stelle nicht weiterkommen, scheuen Sie sich nicht, noch einmal zu beginnen. Natürlich finden Sie den ferti- [●]

gen statischen Aufbau der Website auch auf der DVD (unter *Buchwebsite/Website_HTML*), so dass Sie notfalls »abgucken« können. Wir wünschen Ihnen viel Spaß und Erfolg.



Dreamweaver goes Web 2.0: Statische und dynamische Websites werden mit mehr Leben gefüllt. Mit dem Spry-Framework können Sie benutzerfreundliche Schnittstellen und eindrucksvolle Effekte in Ihre Websites integrieren.

12 Spry – Framework für Ajax

Internetseiten müssen heute mehr bieten als nur die reine Präsentation von Inhalten. Die Besucher sollen mit der Website interagieren, Inhalte ein- und ausblenden oder das Aussehen der Website an ihre Bedürfnisse anpassen können. Mit Spry-Widgets gestalten Sie ganz einfach desktopähnliche Oberflächen für Websites.

12.1 Was ist Spry?

Das Spry-Framework für Ajax gibt es seit Dreamweaver CS3. Ajax bedeutet *Asynchronous JavaScript and XML*. Bei Spry kommt allerdings nicht überall zwingend XML zum Einsatz. Das Spry-Framework ist in erster Linie eine Sammlung aus JavaScript, HTML-Elementen und CSS, um Effekte, Spry-Widgets und auch XML-Anbindungen zu ermöglichen.

Als *Framework* bezeichnet man eine Sammlung von Codeteilen, die Modular miteinander kombinierbar sind. Die Bausteine des Frameworks lassen sich einfach einsetzen, ohne dass man viel Code schreiben und konfigurieren und ohne dass man den Code verstehen muss.

Die Zielgruppe für Spry sind hauptsächlich Designer. Sie sollen mit Spry auf einfache Weise grafische Oberflächen mit einem höheren Benutzerkomfort erstellen können. Tiefer gehende JavaScript-Kenntnisse sind dazu dank Dreamweaver nicht notwendig. Mit etwas Basiswissen über CSS können die Elemente einfach angepasst werden – doch dazu später mehr.

In der aktuellen Dreamweaver-Version ist die Spry-Framework-Version 1.6.1 verfügbar. Adobe entwickelt dieses Framework ständig weiter. Neuere Versionen können Sie von der Adobe-Website beziehen.

In den Dreamweaver-Extensions gibt es auch den integrierten Spry-Updater, der allerdings nicht das Spry-Framework aktualisiert, sondern dazu dient, Ihre verwendeten Dateien an das neue Framework anzupassen. Hier sollten Sie Vorsicht walten lassen, denn selbstverständlich werden Ihre Änderungen in den CSS-Dateien überschrieben.

Die Verwendung des Spry-Frameworks in Dreamweaver teilt sich in die vier in Tabelle 12.1 dargestellten Bereiche.

Spry-Framework in Dreamweaver	
Spry-Widgets	Spry-Widgets sind grafische Elemente, um Benutzern weiterführende Aktionen zu ermöglichen, die sich vom klassischen, starren Seitenaufbau erheblich unterscheiden. So können Sie zum Beispiel ganze Seitenbereiche ein- bzw. ausblenden oder die bekannten Register erstellen.
Spry-Effekte	Effekte werden verwendet, um innerhalb einer HTML-Seite Bereiche gesondert hervorzuheben. So lassen sich beispielsweise bei Mouseover einzelne Bereiche vergrößern, ein- und ausblenden oder verschieben.
Spry-Überprüfung für Formulare	Mit diesem Element werden Formulareingaben in einem HTML-Dokument auf dem Client direkt bei der Eingabe überprüft und Hilfestellungen angezeigt.
Spry-XML-Datensätze	Ein Spry-XML-Datensatz ist eine Anwendung, um XML-Daten in Seitenbereiche zu laden, ohne die komplette Seite neu aufzurufen – das eigentliche Ajax.

Tabelle 12.1 Bereiche des Spry-Frameworks

12.1.1 Vorteile

Spry hat klare Vorteile, wenn es darum geht, Websites mit einer Benutzerführung anzubieten, die mit üblichen HTML- und CSS-Mitteln nicht zu erreichen ist. Besonders das automatische Nachladen von XML-Daten ohne ein Aktualisieren der Website stellt eine echte Errungenschaft dar.

Mit den Spry-Widgets können Websites gestaltet werden, die in ihrem Verhalten mehr einer Desktop-Oberfläche gleichen als einer gewöhnlichen Website. Wer schon einmal nach dem dritten Versuch, die korrekten Daten in ein Formular einzutragen, genervt eine Website verlassen hat, wird sicherlich die Spry-Überprüfung für Formulare besonders zu schätzen wissen. Damit ist eine Überprüfung schon während der Eingabe möglich, um Falscheingaben abzufangen.

Für einen halbwegs erfahrenen Webdesigner ist es recht einfach, die nötigen CSS-Dateien anzupassen. Um die teilweise recht umfangreichen JavaScripts muss man sich nicht kümmern, das übernimmt Dreamweaver für uns.

Spry kann natürlich auch ohne Dreamweaver eingesetzt werden. Bei Adobe gibt es die komplette Dokumentation des Frameworks. In diesem Buch soll jedoch ausschließlich von Spry in Dreamweaver die Rede sein.

12.1.2 Einschränkungen

In den JavaScripts liegt von Fall zu Fall auch der Nachteil des Spry-Frameworks. Soll das Framework nämlich mit anderen auf JavaScript basierenden Anwendungen – zum Beispiel in einem Unternehmensnetzwerk – eingesetzt werden, kann es ganz schnell zu Konflikten insbesondere mit anderen Ajax-Bibliotheken kommen. Aktuell werden immer mehr Ajax-Frameworks entwickelt, so dass man die Kompatibilität im Einzelfall beobachten muss. Eine generelle Aussage lässt sich hier nicht treffen.

12.1.3 Fehlermeldung bei lokaler Anzeige

Wie bei vielen Verhalten wird auch bei Spry im Internet Explorer eine Warnung ausgegeben, wenn Sie eine lokale Vorschau erstellen. Wenn Sie diese HTML-Dokumente im Internet aufrufen, erscheint die Warnung nicht.

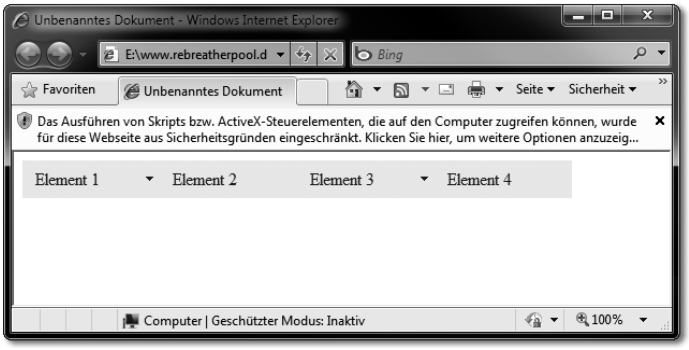


Abbildung 12.1 Fehlermeldung im Internet Explorer

Der Grund für den Warnhinweis sind die Skripte in diesen Dokumenten. Der Internet Explorer interpretiert dies bei lokaler Ausführung als »Angriff«. Leider kann in der aktuellen Dreamweaver-Version dieses Verhalten nicht mehr auf einfache Weise behoben werden. Wenn Sie es als störend empfinden, können Sie folgenden Befehl von Hand in den Head des XHTML-Dokuments einfügen:

```
<!-- saved from url=(0014)about:internet -->
```

Er weist den Browser an, diese Website als zum Internet gehörig zu behandeln und diese Warnung nicht auszugeben. Die Zeile sollte beim Veröffentlichen wieder entfernt werden.

12.2 Spry in diesem Buch

Erläuterungen zu einzelnen Spry-Anwendungen finden Sie nicht nur in diesem, sondern auch noch in weiteren Kapiteln. Thematisch sind sie nach Anwendungsbereichen den Grundlagenkapiteln nachgestellt und sollen Ihnen helfen, einen schnellen Einstieg zu finden. Spry-Elemente finden Sie daher in den Kapiteln über Formulare (Kapitel 14, »Formulare«) und Verhalten (Kapitel 9, »JavaScript und Verhalten«) und als gesondertes Kapitel nach XML (Kapitel 31, »Spry und XML«).

12.3 Spry-Widgets

Widgets wurden ursprünglich auf Apple-Computern eingesetzt, um alle möglichen Annehmlichkeiten, wie z. B. Kalender, Uhr usw., auf dem Desktop anzuzeigen. Mittlerweile sind diese kleinen Dienstprogramme auch unter Windows weit verbreitet. Dank immer schnellerer Internetanbindungen und leistungsfähigerer Rechner hat das Prinzip dieser Arbeitserleichterung nun auch das Internet erobert. Ajax, JavaScript und CSS sind hier die Zauberworte.

Die mit Spry angebotenen Widgets dienen jedoch nicht dazu, auf einer Website eine Uhr oder einen Kalender darzustellen. Vielmehr ermöglichen sie komfortable Interaktionen der Benutzer mit der Website. Die Spry-Funktionen im Register LAYOUT und im Register SPRY sind identisch.



Abbildung 12.2 Anordnung der Spry-Widgets im Register »Spry«

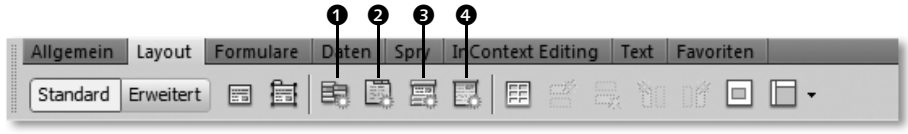


Abbildung 12.3 Anordnung der Spry-Widgets im Register »Layout«

In den beiden Registern stehen insgesamt fünf Spry-Widgets zur Verfügung:

1. Spry-Menüleiste
2. Spry Paletten mit Registerkarten
3. Spry-Akkordeon
4. Spry Reduzierbare Palette
5. Spry Quick Info

In den folgenden Abschnitten werden wir die Widgets ausführlich besprechen.

12.3.1 Spry-Dateien kopieren

Wenn Sie Spry-Elemente in Ihre Websites einfügen, kopiert Dreamweaver mit jedem eingefügten Spry-Element die notwendigen Dateien in einen Ordner namens *SpryAssets*. Dieser Ordner muss auf den Webserver kopiert werden, wenn Sie die Website veröffentlichen möchten und sie auch im Internet funktionieren soll.

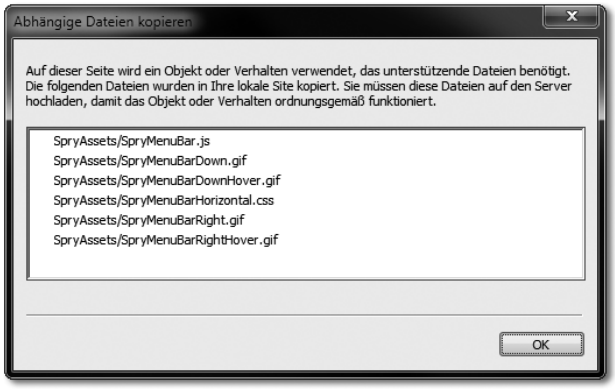


Abbildung 12.4 Kopieren der notwendigen Dateien

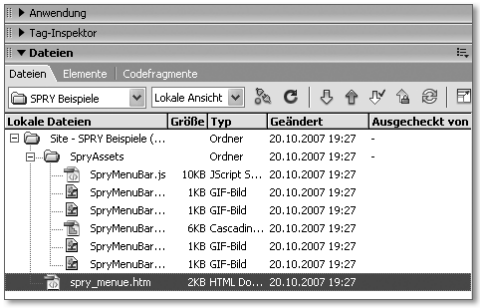


Abbildung 12.5 Ordner mit Spry-Framework

12.4 Spry-Menüleiste

Die Spry-Menüleiste erzeugt ein dynamisches Menü mit ausklappbaren Untermenüpunkten. Diese Art der Menüs ist im Internet weit verbreitet.

In Abbildung 12.6 sehen Sie eine fertige Menüleiste ohne Layoutanpassung im Browser. Durch das Klicken auf die kleinen Pfeile schiebt sich das Menü nach unten und gibt den Blick auf die weiterführenden Menüpunkte frei. Erstellt werden kann dieses Menü mit nahezu beliebig vielen Menü- und Unterpunkten auf insgesamt drei Ebenen.

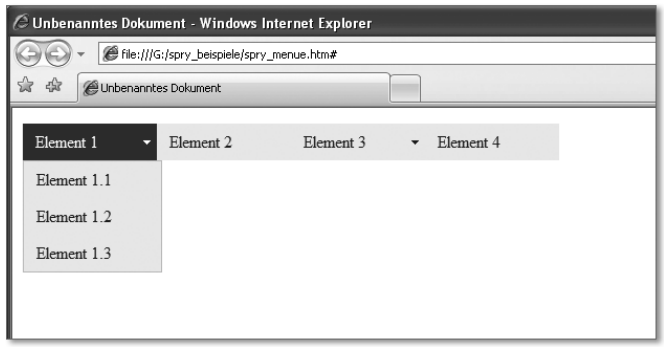


Abbildung 12.6 Spry-Menüleiste im Browser

12.4.1 Spry-Menüleiste anlegen

Beim Anlegen der Menüleiste haben Sie die Wahl zwischen einer horizontalen und einer vertikalen Ausrichtung der Menüs. Nach einer Bestätigung mit OK fügt Dreamweaver automatisch sämtliche Elemente in das HTML-Dokument ein.

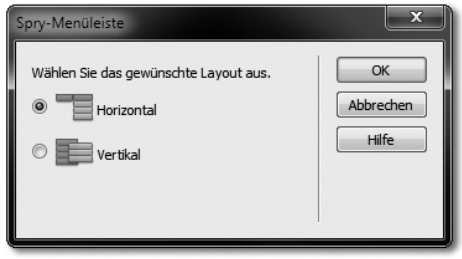


Abbildung 12.7 Auswahl für Menüleiste

12.4.2 Spry-Menüleiste anpassen

Im Vorschaumodus stehen Ihnen nun verschiedene Anpassungsmöglichkeiten zur Verfügung (siehe Abbildung 12.8). Durch Klicken auf das kleine Register über

dem Spry-Widget ❶ erscheinen die Einstellmenüs in der Eigenschaftenpalette. Dort werden unter ❷ die Hauptmenüpunkte aufgelistet. Die angezeigten Bezeichnungen sind zunächst die Vorgaben von Dreamweaver. Weitere Hauptmenüpunkte können ganz einfach über die Plus- und Minus-Icons angefügt oder entfernt werden. Mit den kleinen Pfeilen ist eine Änderung der Menüpunktreihenfolge möglich.

Das Gleiche gilt natürlich auch bei den Unterpunkten ❹. Um die einem Hauptmenüpunkt zugeordneten Unterpunkte zu sehen, muss im linken Feld ❷ der übergeordnete Eintrag angewählt sein. Eine dritte, im abgebildeten Fall inaktive Ebene steht optional zusätzlich zur Verfügung.

Die Bezeichnungen der Menüpunkte werden im Eingabefeld ❺ festgelegt. Im Eingabefeld für den HYPERLINK ❻ kann gleich eine Verlinkung eingetragen werden. Das Layout im Browser erhalten Spry-Widgets durch CSS. Hinter dem Link ❸ DIESES WIDGET ANPASSEN verbirgt sich eine Hilfedatei mit den Einstellungen für die CSS-Datei, doch dazu am Ende dieses Kapitels mehr.

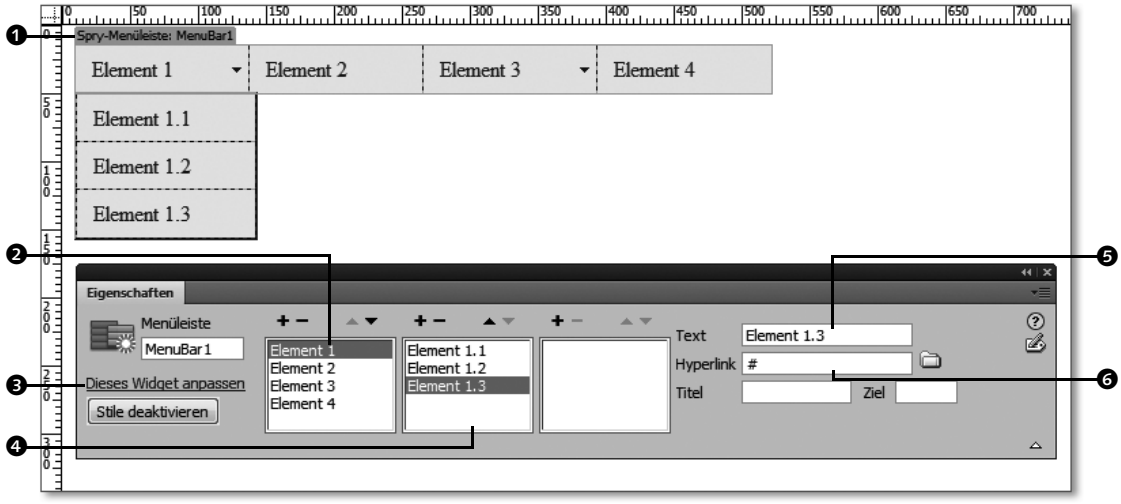


Abbildung 12.8 Die Eigenschaftenpalette für die Spry-Menüleiste

Spry-Menüleiste im Quelltext anpassen

Wie im folgenden Listing zu sehen, ist auch die Anpassung im Quelltext denkbar einfach. Die einzelnen Menüpunkte werden als Listenpunkte ausgegeben und können problemlos bearbeitet werden. Es ist oft einfacher, Bezeichnungen direkt hier zu ändern, als sich durch die Einträge in der Eigenschaftenpalette zu klicken.

```
<ul id="MenuBar1" class="MenuBarHorizontal">
  <li><a class="MenuBarItemSubmenu" href="#">Element 1</a>
```

```
      <ul>
        <li><a href="#">Element 1.1</a></li>
        <li><a href="#">Element 1.2</a></li>
        <li><a href="#">Element 1.3</a></li>
        </ul>
  </li>
  <li><a href="#">Element 2</a></li>
  <li><a class="MenuBarItemSubmenu" href="#">Element 3</a>
      <ul>
        <li><a class="MenuBarItemSubmenu" href="#">Element
        3.1</a>
            <ul>
              <li><a href="#">Element 3.1.1</a></li>
              <li><a href="#">Element 3.1.2</a></li>
            </ul>
        </li>
        <li><a href="#">Element 3.2</a></li>
        <li><a href="#">Element 3.3</a></li>
      </ul>
  </li>
  <li><a href="#">Element 4</a></li>
</ul>
```

Listing 12.1 Spry-Menüleiste im Quellcode

12.5 Spry-Palette mit Registerkarten

Tabbed Browsing bezeichnet eine Form der Darstellung im Browser, bei der mehrere Dokumente gleichzeitig geladen werden können und einfach zwischen diesen gewechselt werden kann. Seit Firefox diese Registernavigation eingeführt hat, ist sie »State of the Art« beim Surfen im Internet. Mit einiger Verzögerung hat sich Microsoft anstecken lassen und dieses Feature in den neuen Internet Explorer 7 integriert.

Mit der Funktion Spry-Paletten mit Registerkarten können Sie diese Technik nun auch auf Internetseiten anwenden. Spry ermöglicht den Wechsel zwischen verschiedenen Inhalten innerhalb eines HTML-Dokuments, ohne die Seite neu zu laden. Die Inhalte werden jeweils in einem eigenen Register angezeigt, und die Register werden wie beim Tabbed Browsing ein- bzw. ausgeblendet. Im Prinzip liegen hier zwei oder mehrere CSS-Layer übereinander, wobei jeweils nur einer sichtbar ist.

Große Datenmengen [«]

Durch die Möglichkeiten, große Datenmengen einfach in unterschiedlichen Registern zu platzieren, kommt man schnell in Versuchung, »zu viele« Inhalte auf einmal zu laden. Bevor die Register jedoch korrekt angezeigt werden können, müssen alle Inhalte komplett geladen werden – erst dann baut der Browser das Layout vollständig auf. Die Daten werden auch für Register geladen, die anfangs geschlossen sind. Diese Aussage gilt auch für alle folgenden Spry-Widgets, da sich diese nur in der Art der grafischen Umsetzung unterscheiden.

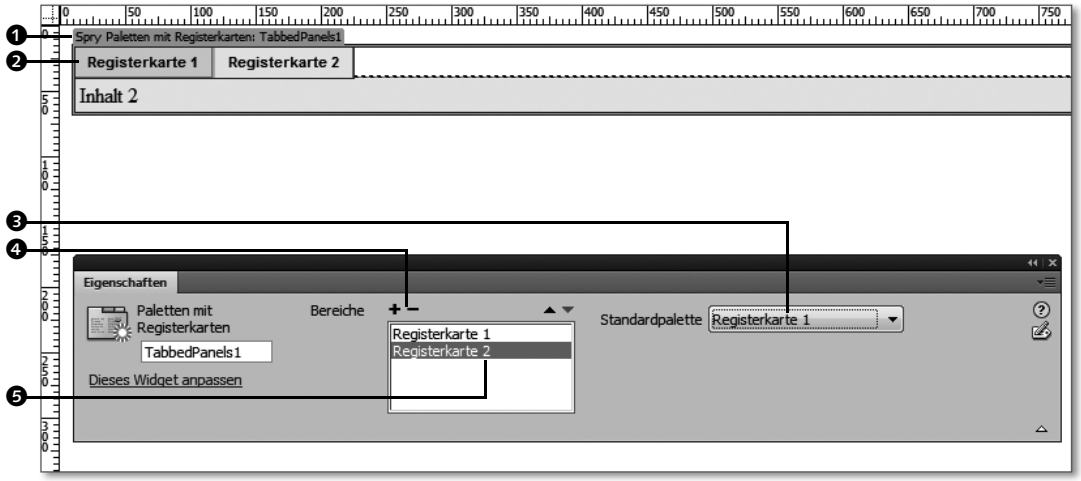


Abbildung 12.9 Die Eigenschaftenpalette für Spry-Registerkarten

12.5.1 Spry-Registerkarten einstellen

In der Eigenschaftenpalette werden nach dem Klicken auf das Register mit dem Namen des Spry-Elements ❶ die Einstellmöglichkeiten angezeigt. Wie beim Spry-Menü können über die Plus- und Minussymbole ❹ weitere Einträge hinzugefügt oder bestehende gelöscht werden. Welches der Register nun beim ersten Aufruf der Website angezeigt wird, kann in der Einstellung der STANDARDPALETTE ❸ festgelegt werden.

12.5.2 Spry-Registerkarten füllen

Im Vorschaumodus ist immer nur eins der vorhandenen Register zu sehen. Damit auch Inhalte in andere als das gerade sichtbare Register aufgenommen werden können, wählen Sie das gewünschte Register in der Eigenschaftenpalette ❺ oder direkt über den Registerinhalt ❷ aus.

Wie Sie im folgenden Listing sehen, ist auch hier der Quelltext im HTML-Dokument sehr einfach gehalten:

```
<div id="TabbedPanels1" class="TabbedPanels">
  <ul class="TabbedPanelsTabGroup">
    <li class="TabbedPanelsTab" tabindex="0">Registerkarte 1
    </li>
    <li class="TabbedPanelsTab" tabindex="0">Registerkarte 2
    </li>
  </ul>
  <div class="TabbedPanelsContentGroup">
    <div class="TabbedPanelsContent">Inhalt 1</div>
    <div class="TabbedPanelsContent">Inhalt 2</div>
  </div>
</div>
```

Listing 12.2 Spry-Registerkarten im Quellcode

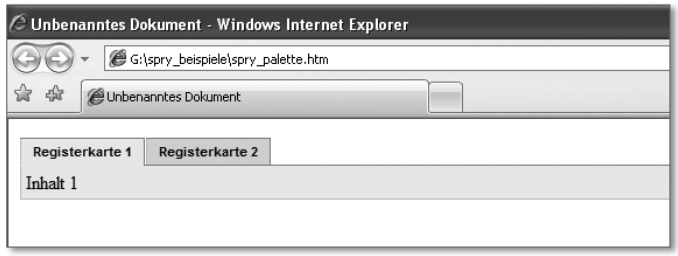


Abbildung 12.10 Spry-Register im Browser

12.6 Spry-Akkordeon

Das Spry-Akkordeon unterteilt das Layout in horizontale Bereiche, die je nach Bedarf durch Klicken auf die Überschrift überblendet werden. Beim Anklicken öffnet sich der verdeckte Bereich langsam und überdeckt damit alle anderen Bereiche.

Spry-Akkordeon einstellen

Wie bei allen Spry-Widgets zeigt die Eigenschaftenpalette beim Anwählen des Widgets ❶ die einstellbaren Eigenschaften an. Wieder können über das Plus- und Minussymbol ❷ weitere Bereiche eingefügt oder entfernt sowie über die kleinen Pfeile ❸ die Reihenfolge verändert werden.

Im Quelltext ist die Hierarchie der Elemente etwas anders aufgebaut als bei den Registerkarten. Auch hier sollte ein Befüllen keinerlei Probleme bereiten.

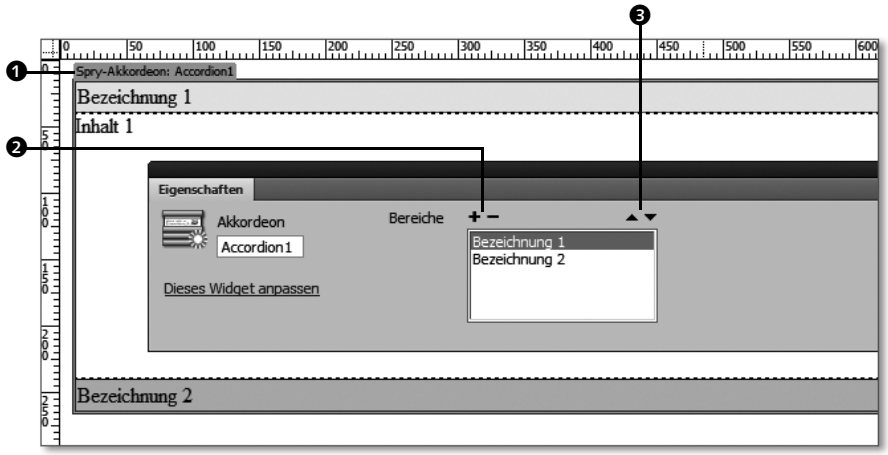


Abbildung 12.11 Die Eigenschaftenpalette für Spry-Akkordeon

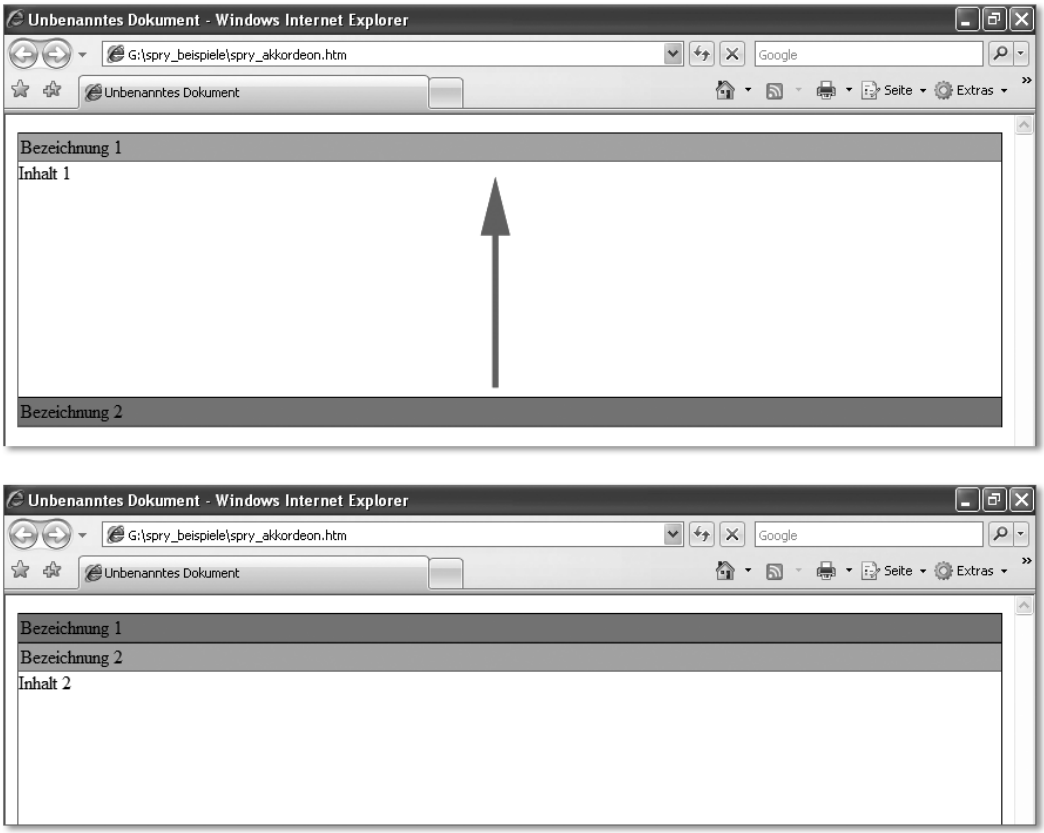


Abbildung 12.12 Spry-Akkordeon mit zwei Bereichen im Browser

```
<div id="Accordion1" class="Accordion" tabindex="0">
  <div class="AccordionPanel">
    <div class="AccordionPanelTab">Bezeichnung 1</div>
    <div class="AccordionPanelContent">Inhalt 1</div>
  </div>
  <div class="AccordionPanel">
    <div class="AccordionPanelTab">Bezeichnung 2</div>
    <div class="AccordionPanelContent">Inhalt 2</div>
  </div>

  <div class="AccordionPanel">
    <div class="AccordionPanelTab">Bezeichnung 3</div>
    <div class="AccordionPanelContent">Inhalt 3</div>
  </div>
  <div class="AccordionPanel">
    <div class="AccordionPanelTab">Bezeichnung 4</div>
    <div class="AccordionPanelContent">Inhalt 4</div>
  </div>
</div>
```

Listing 12.3 Quelltext für Spry-Akkordeon mit vier Bereichen

12.7 Reduzierbare Palette

Reduzierbare Paletten sind im Gegensatz zu den vorherigen Spry-Widgets nicht in der Lage, zwischen verschiedenen Inhalten umzublenden, sondern dienen dazu, einen einzelnen Bereich im Dokument ein- bzw. auszublenden.

Dokumentinhalte außerhalb des Spry-Widgets werden beim Überblenden im Layout automatisch verschoben. Innerhalb eines HTML-Dokuments können mehrere reduzierbare Paletten verwendet werden.

»Spry Reduzierbare Palette« einstellen

In der Eigenschaftenpalette kann bei aktivem Element ❶ die Standardansicht sowie die Startansicht ❷ ausgewählt werden. Die Animation beim Umblenden kann deaktiviert werden ❸. Sinnvoll kann das sein, wenn Sie bei Ihrer Zielgruppe etwas langsamere Rechner erwarten. Mit der Deaktivierung entlasten Sie System und Grafikkarte deutlich.

Wie Sie auch hier im Quelltext sehen, ist der Code, zumindest im HTML-Dokument, äußerst kompakt. Die gesamte Funktionalität wird auch hier in die externen JavaScript-Dateien ausgelagert.

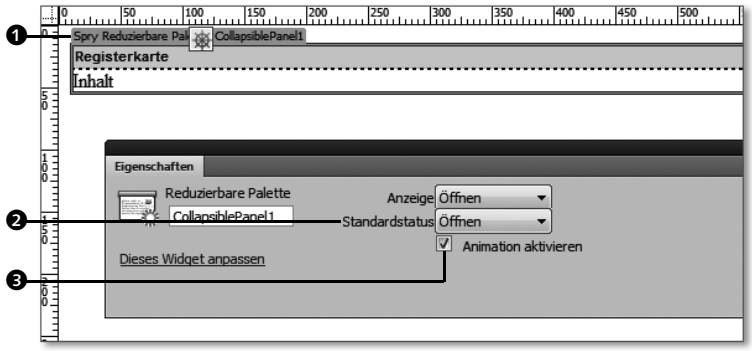


Abbildung 12.13 Die Eigenschaftenpalette für reduzierbare Paletten

```
<div id="CollapsiblePanel1" class="CollapsiblePanel">
  <div class="CollapsiblePanelTab" tabindex="0">
  Registerkarte</div>
  <div class="CollapsiblePanelContent">Inhalt</div>
</div>
```

Listing 12.4 Quellcode für »Spry reduzierbare Palette«

12.8 Spry-QuickInfo

Oft versucht man, in einer Website Erklärungen zur Bedienung oder zu einzelnen Produkten unterzubringen und weiß nicht recht, wohin damit. Aus Windows-Anwendungen sind Tool-Tipps bestens bekannt – kleine Fensterchen, die sich öffnen, wenn man die Maus über ein Icon oder einen bestimmten Bereich einer Anwendung bewegt.

Spry-QuickInfo stellt diese Funktion auch auf einer Website zur Verfügung. Als Auslöser für eine Spry-QuickInfo können Bilder oder beliebige Div-Container verwendet werden. In unserem Beispiel haben wir einen AP-Div als maussensitives Element verwendet.

Wie auch bei den anderen Spry-Elementen können die Eigenschaften durch Anklicken der kleinen blauen Registerkarte ❹ (siehe Abbildung 12.14) verändert werden. Die Offsetpositionen ❺ und ❻ zur Mauszeigerposition bei Auslösung der QuickInfo, eine Ein- oder Ausblendverzögerung ❽ sowie ein eventueller Effekt ❾ und weiterführende Aktionssteuerungen ❼ sind hier einstellbar.

```
<div id="apDiv1">Auslöser</div>
<div class="tooltipContent" id="sprytooltip1">Hallo, lieber Leser –
 das ist eine Spry Quickinfo</div>
```

Listing 12.5 Quellcode für Spry-QuickInfo

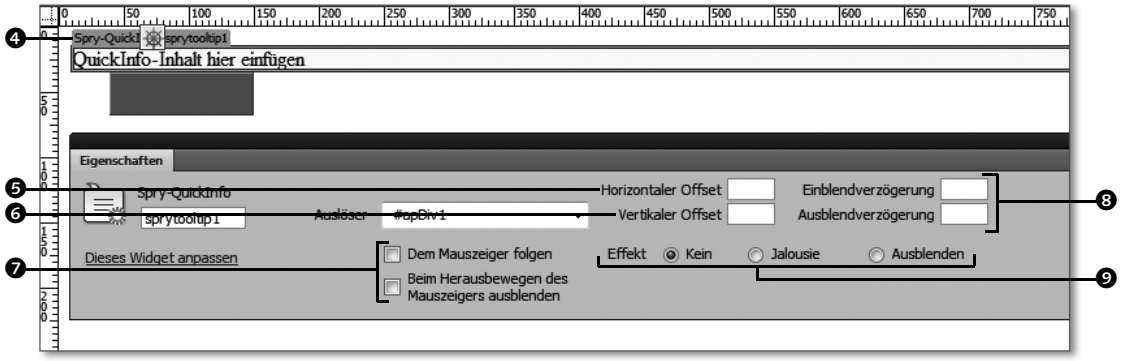


Abbildung 12.14 Spry-QuickInfo einstellen

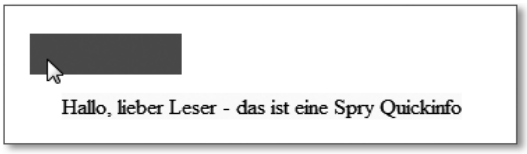


Abbildung 12.15 Spry-QuickInfo im Browser

12.9 Spry-Widgets mit CSS gestalten

Spry-Widgets können gestalterisch im Rahmen der CSS-Spezifikation ganz an Ihre Wünsche angepasst werden. Im Ordner *SpryAssets* werden neben den JavaScript-Dateien auch die CSS-Stile abgelegt. Diese CSS-Dateien können beliebig verändert werden. Wie Sie in Abbildung 12.16 sehen, ist der Code wirklich sehr gut kommentiert.

In der Eigenschaftenpalette sehen Sie beim Anwählen eines Spry-Widgets immer den Link Dieses Widget anpassen. Der Link führt Sie direkt zur Hilfedatei mit der Dokumentation zu den einzelnen Elementen (siehe Abbildung 12.17). Alle diese Elemente und deren Einstellmöglichkeiten hier zu beschreiben würde den Rahmen des Buches sprengen. Sicherlich werden Sie bei eigenen Versuchen schnell zu Ergebnissen kommen.

Fortgeschrittene Benutzer können, wenn sie nicht direkt im Quelltext arbeiten möchten, auch das CSS-Bedienfeld zu Hilfe nehmen. Da es sich bei dem visuellen Teil der Widgets ausschließlich um CSS handelt, stehen Ihnen natürlich auch hier alle Möglichkeiten von Dreamweaver zur Verfügung.

```
@charset "UTF-8";

/* SpryMenuBarHorizontal.css - Revision: Spry Preview Release 1.4 */

/* Copyright (c) 2006. Adobe Systems Incorporated. All rights reserved. */

/**************************************************************************

 LAYOUT INFORMATION: describes box model, positioning, z-order

 **************************************************************************/

/* The outermost container of the Menu Bar, an auto width box with no margin or padding */
ul.MenuBarHorizontal
{
    margin: 0;
    padding: 0;
    list-style-type: none;
    font-size: 100%;
    cursor: default;
    width: auto;
}
/* Set the active Menu Bar with this class, currently setting z-index to accomodate IE rendering bug:
http://therealcrisp.xs4all.nl/meuk/IE-zindexbug.html */
ul.MenuBarActive
{
    z-index: 1000;
}
```

Abbildung 12.16 CSS-Datei für Spry

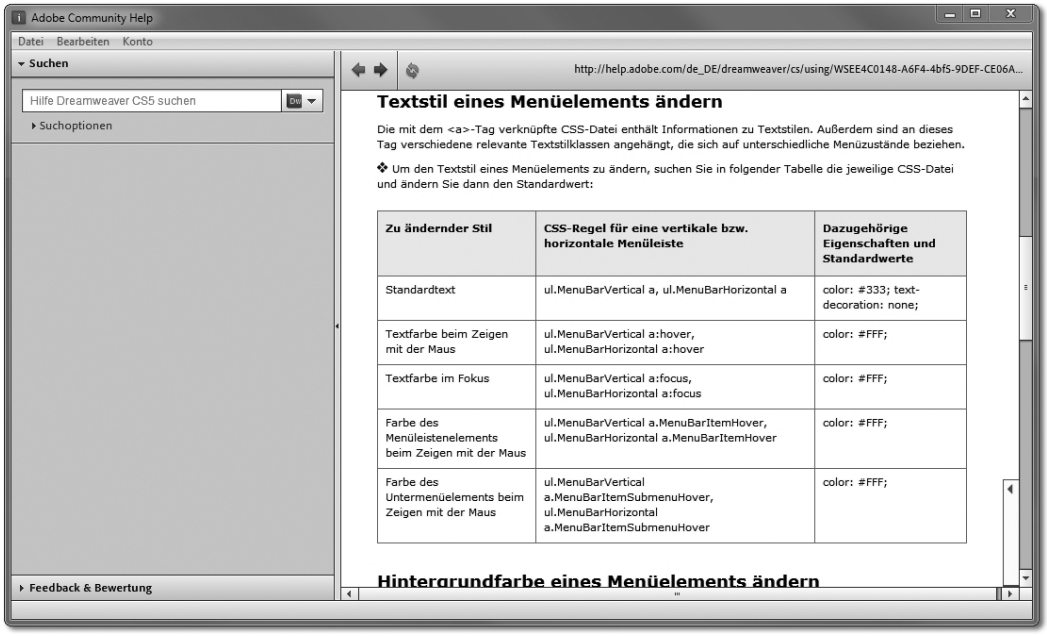


Abbildung 12.17 Dreamweaver-Dokumentation für CSS-Einstellungen

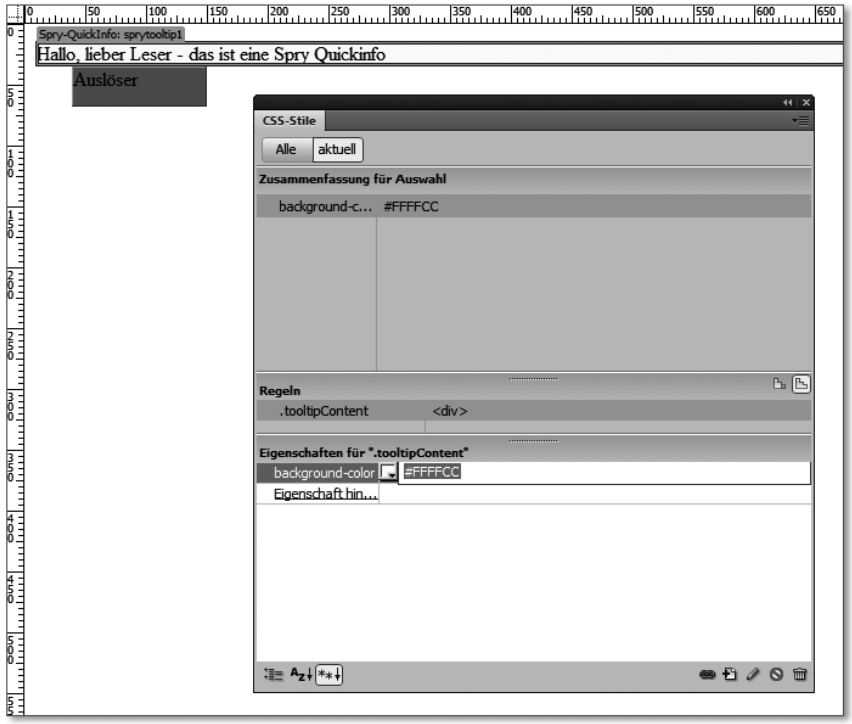


Abbildung 12.18 Ansicht der Spry-CSS-Stile im CSS-Bedienfeld

Nützlich sind hier der neue Codenavigator und die Live-Code-Ansicht. Aktivieren Sie einfach die Live-Ansicht, danach den Live-Code – Dreamweaver schaltet auf die geteilte Ansicht für Entwurf und Code. Nun können Sie die Widgets nicht nur ausprobieren, sondern Sie sehen im Code daneben auch, wie JavaScript den Quellcode dynamisch verändert und so den Elementen entsprechende Eigenschaften hinzufügt oder wieder entfernt, wenn Sie zum Beispiel mit der Maus über eine Akkordeonfläche fahren.

Drücken Sie zur näheren Untersuchung F6, so wird das JavaScript eingefroren. Der Klick auf das Steuerrad des Codenavigators oder – falls nicht angezeigt – Strg+Alt+N öffnet den Codenavigator. Er zeigt Ihnen alle CSS-Selektoren, die für den entsprechenden Bereich ausschlaggebend sind. Ein Klick auf diese bringt Sie in der Codeansicht auch direkt zum entsprechenden Selektor. Ändern Sie die Eigenschaften (z. B. eine Farbe), so sehen Sie die Auswirkungen nach dem Aktualisieren mit F5 direkt in der Live-Ansicht.

In der Abbildung 12.19 wurde bei aktivierter Live-Ansicht und aktiviertem Live-Code mit der Maus über einen Bereich gefahren und der Code eingefroren. Im Codenavigator sind die CSS-Eigenschaften sichtbar.

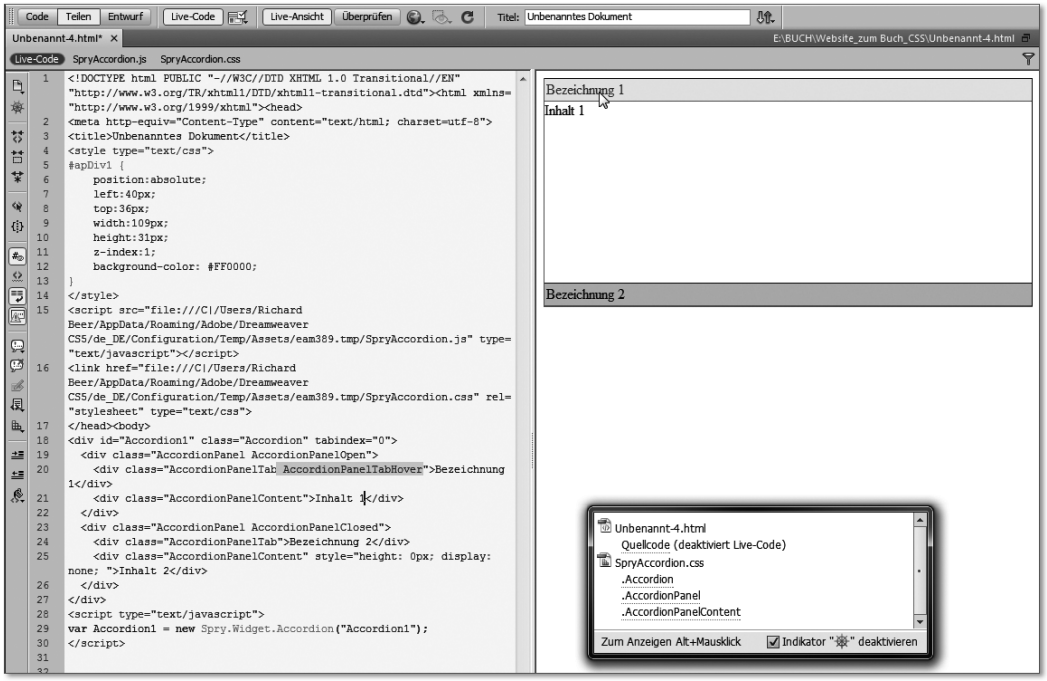


Abbildung 12.19 Eingefrorener Code

Selten wird jede einzelne Unterseite unterschiedlich aufgebaut. Meistens sind zumindest einige Elemente in der gesamten Website identisch. Dreamweaver erleichtert Ihnen die Arbeit durch Vorlagen und Bibliotheken.

13 Vorlagen und Bibliotheken

Bibliotheken und Vorlagen bieten den Vorteil der zentralen Bearbeitung immer wiederkehrender Layoutelemente und ermöglichen so eine einfachere Verwaltung von umfangreichen Websites, bei denen CSS nicht verwendet werden kann. In einem gewissen Rahmen können Vorlagen und Bibliotheken dann ein datenbankbasiertes System (CMS) zur Verwaltung einer Website durchaus ersetzen.

Wird die Vorlage oder das Bibliothekselement verändert, werden alle damit verbundenen HTML-Dokumente ebenfalls verändert. Dies ist vorteilhaft, wenn Sie sehr umfangreiche Websites mit immer wiederkehrenden Layouts beispielsweise für ein Intranet haben.

Als Beispiel können Sie sich eine Website denken, bei der Navigationsmenü und Kopf mit Logo immer gleich sind. Diese werden daher auf der Vorlagenseite positioniert und programmiert. Die einzelnen Seiten der Site selbst hängen dann von dieser Vorlage ab. Auf ihnen kann nur der eigentliche Inhalt geändert werden, nicht aber Menü und Kopf. Wenn dann eine weitere Unterseite erstellt werden muss und diese im Menü verlinkt werden soll, so ändern Sie das Menü nur auf der Vorlagenseite. Diese Änderung wird dann von Dreamweaver automatisch auf alle abhängigen Seiten übertragen, ohne dass deren individueller Inhalt angerührt wird.

CSS statt Vorlagen [«]

Mit CSS brauchen Sie heute Vorlagen nur noch bedingt. Auch in einer externen CSS-Datei lassen sich alle Elemente einer Website ablegen. Nur wenn Sie zwingend wiederkehrende Tabellenlayouts benötigen, an alten Websites arbeiten müssen oder Contribute einsetzen, kann sich das Arbeiten mit Vorlagen lohnen.

13.1 Mit Vorlagen arbeiten

Bei Vorlagen müssen Sie bearbeitbare Bereiche definieren. Bereiche, die nicht explizit zur Bearbeitung freigegeben wurden, sind auf diese Weise vor ungewollten Zugriffen geschützt. Damit können Sie Mitarbeitern einer Firma Vorlagen zur Verfügung stellen, mit denen sie für ihre Arbeitsbereiche neue Seiten erstellen oder pflegen, ohne Angst um das Grundlayout haben zu müssen.

13.1.1 Vorlagen erstellen

Vorlagen können Sie von jedem beliebigen HTML-Dokument (auch PHP usw.) erstellen. Legen Sie zunächst das Dokument mit den Rahmenbedingungen an, die in allen von der Vorlage zu erstellenden Dokumenten identisch sein sollen, und lassen Sie die Bereiche mit individuellen Inhalten zunächst frei oder mit Blindtext gefüllt.

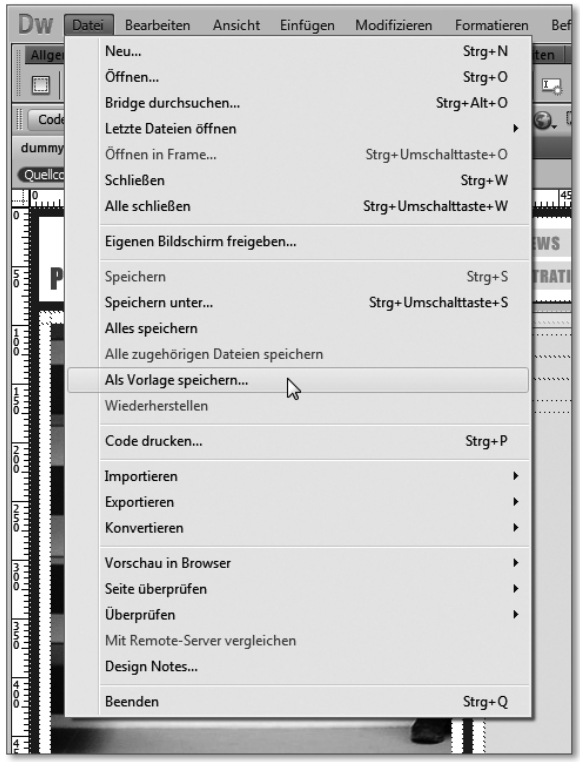


Abbildung 13.1 Erstellen einer Vorlage

Wählen Sie aus dem Menü DATEI • ALS VORLAGE SPEICHERN… aus. Im Beispiel soll im linken Bereich ein Bild und im rechten Bereich ein Text individuell austausch-

bar sein. Vergeben Sie in der folgenden Dialogbox einen Namen, und speichern Sie das Dokument ab.



Abbildung 13.2 Dialogbox »Als Vorlage speichern«

Zunächst ändert sich das Aussehen des Dokuments nicht, außer dass in der Statusleiste zusätzlich <VORLAGE> steht.

»Templates«-Ordner

Dreamweaver hat, wie Sie im Bedienfeld DATEIEN sehen können, einen weiteren Ordner mit dem Namen *Templates* erstellt. In diesem Ordner werden die Vorlagen mit der Dateierweiterung *.dwt* gespeichert. Diesen Ordner dürfen Sie auf keinen Fall löschen, solange Sie mit Vorlagen arbeiten möchten.

Ordner bei Synchronisation ignorieren [«]

Markieren Sie diesen Ordner im Dateifenster zum *Cloaking*. Dadurch ignoriert Dreamweaver den Ordner und dessen Inhalte beim Upload oder der Synchronisation auf den Server. Das Cloaking können Sie für eine einzelne Datei oder einen Ordner aktivieren (oder deaktivieren), indem Sie im Dateifenster mit der rechten Maustaste auf die entsprechende Datei bzw. den entsprechenden Ordner klicken und CLOAKING • CLOAKING wählen.

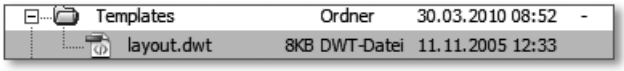


Abbildung 13.3 Neuer Ordner nach dem Erstellen einer Vorlage

Wenn Sie sich den Quelltext ansehen, werden Sie neue Einträge finden:

```
<!-- TemplateBeginEditable name="doctitle" -->
```

Dreamweaver nutzt in den Vorlagen-Dokumenten HTML-Kommentare zur Programmsteuerung. Da es sich um Kommentare handelt, haben diese im Browser keine Auswirkung und führen lediglich dazu, dass Dreamweaver Vorlagen, Vorlagenbereiche und Bibliothekselemente als solche erkennt.

[»] **Kompatible Vorlagen in Dreamweaver und GoLive**

Dreamweaver- und alte Adobe-GoLive-Vorlagen sind kompatibel. Sie können ohne Probleme Vorlagen aus alten GoLive-Versionen in Dreamweaver verwenden. Auch alte Dreamweaver-Vorlagen sind natürlich noch immer verwendbar.

13.1.2 Bearbeitbare Bereiche festlegen

In der neuen Vorlage müssen im zweiten Schritt bearbeitbare Bereiche definiert werden. Grundsätzlich ist zunächst alles *nicht* bearbeitbar. Um einen bearbeitbaren Bereich zu erstellen, klicken Sie in den Bereich Ihres Dokuments, den Sie zur Bearbeitung freigeben möchten, und wählen dann in der Einfügepalette im Register Allgemein den Punkt Vorlagen • Bearbeitbarer Bereich aus.

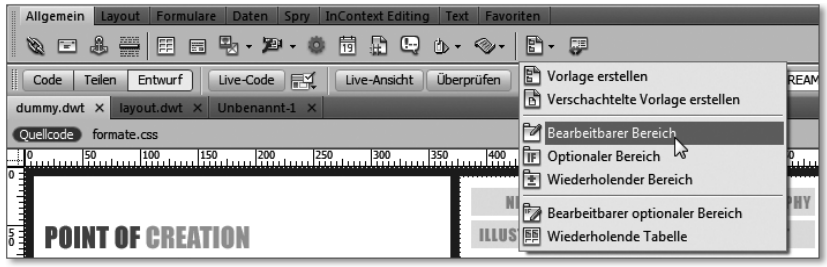


Abbildung 13.4 Bearbeitbaren Bereich erstellen

Geben Sie dem Bereich einen Namen Ihrer Wahl, und speichern Sie die Vorlage erneut ab. In Abbildung 13.5 sehen Sie ein Beispiel mit den neuen bearbeitbaren Bereichen IMAGE und CONTENT.

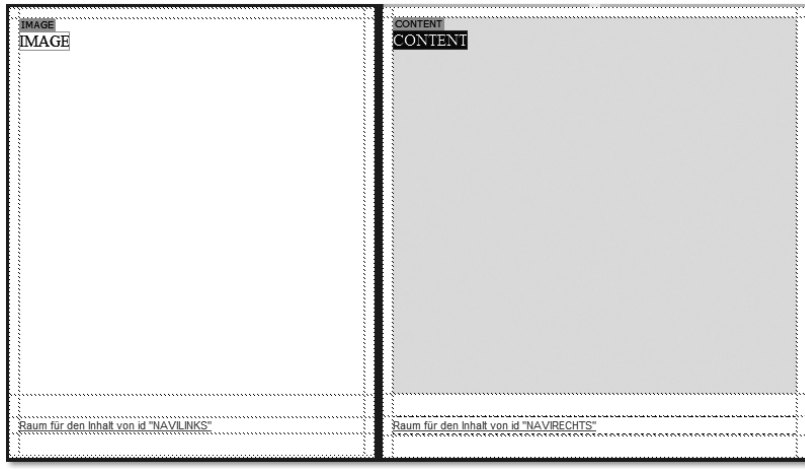


Abbildung 13.5 Bearbeitbarer Bereich in der Vorlage

13.1.3 Dokument aus Vorlage erstellen

Wenn Sie nun eine Vorlage gespeichert haben, können Sie ein neues Dokument aus dieser Vorlage erstellen. Dafür gibt es mehrere Möglichkeiten. Wählen Sie DATEI • NEU aus dem Menü aus. Wechseln Sie zu SEITE AUS VORLAGE, und wählen Sie die gewünschte Vorlage aus.



Abbildung 13.6 Dokument aus Vorlage erstellen

Sie können auch eine neue Datei anlegen und aus dem Bedienfeld ELEMENTE Ihre Vorlage auswählen. Klicken Sie dazu auf das Icon ❶, und ziehen Sie dann die gewünschte Vorlage einfach in Ihr Dokument.

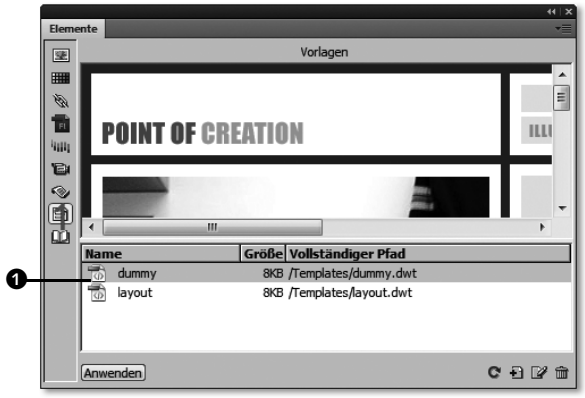


Abbildung 13.7 Vorlagen in der Palette ELEMENTE

Das Ergebnis ist ein neues Dokument mit der gewünschten Vorlage inklusive aller Bilder und Objekte. Verändern können Sie das Dokument nur in definierten, bearbeitbaren Bereichen der Vorlage.



Abbildung 13.8 Eine neues, aus der Vorlage erzeugtes Dokument

Rechts oben im neuen Dokument wird Ihnen angezeigt, welche Vorlage verwendet wurde. Der Name des bearbeitbaren Bereichs wird direkt im Dokument angezeigt. Diese Markierungen werden im Browser nicht dargestellt.

13.1.4 Vorlagen entfernen

Manchmal ist es sinnvoll, Dokumente aus Vorlagen zu erstellen, diese dann aber über Modifizieren • Vorlagen • Von Vorlage lösen wieder von der Vorlage abzukoppeln. Bedenken Sie bei dieser Vorgehensweise, dass das Dokument dann nicht mehr durch die Vorlage aktualisiert werden kann. Sie sollten ein Dokument nur von der Vorlage lösen, wenn Sie individuelle Änderungen vornehmen möchten, die die Vorlage nicht zulässt.

13.1.5 Vorlagen aktualisieren

Wenn Sie eine Vorlage verändern und speichern, erscheint eine Dialogbox mit allen Dokumenten, die aus dieser Vorlage erstellt wurden. Wählen Sie aus, welche Sie aktualisieren wollen. Dokumente, die Sie von der Vorlage gelöst haben, werden nicht aktualisiert.

[!]

Geöffnete Dokumente aktualisieren

Sollen Dokumente durch die Vorlage aktualisiert werden, die gerade in Dreamweaver geöffnet sind, so werden diese zwar auch aktualisiert, müssen jedoch noch einmal gespeichert werden. Um Versionskonflikte zu vermeiden, empfehlen wir daher, Vorlagen nur dann zu aktualisieren, wenn außer der verwendeten Vorlage kein anderes (abhängiges) Dokument geöffnet ist.

13.2 Tricks mit Vorlagen

13.2.1 Verschachtelte Vorlagen

Vorlagen können in nahezu beliebiger Tiefe verschachtelt werden. So können Sie beispielsweise die grundlegenden Dokumentvorgaben, wie z. B. Seitenränder und Hintergrundfarbe, als Mastervorlage (dieser Name erscheint nicht in Dreamweaver) und verschiedene Layouts der einzelnen Unterseiten als Vorlage abspeichern.

Wird die Mastervorlage verändert, ändern sich auch alle Untervorlagen. Wird eine Untervorlage geändert, ändern sich nur die damit erstellten Dokumente.

Wählen Sie dazu Verschachtelte Vorlage erstellen aus der Einfügeleiste im Register Allgemein aus. Die weitere Vorgehensweise ist mit der weiter oben beschriebenen Abfolge identisch.

13.2.2 Wiederholte Bereiche

In vielen Websites gibt es immer wiederkehrende Layoutelemente. So könnten zum Beispiel »News« immer aus einem Bereich für ein Bild, einem kurzen Text und einem Link zu einer Website bestehen. Dieser Bereich kann beim Eingeben einer neuen »News« dupliziert werden. Solche Anforderungen lassen sich mit *wiederholten Bereichen* in Vorlagen sehr einfach realisieren.

Wiederholte Bereiche werden angelegt, indem Sie aus der Einfügeleiste über das Icon Vorlage die Option Wiederholender Bereich auswählen und in dieser eine bearbeitbare Region festlegen.

In Dokumenten, die von einer solchen Vorlage erstellt wurden, finden Sie eine Art Navigation wieder, die es Ihnen ermöglicht, den zuvor deklarierten Bereich zu vervielfältigen und die Reihenfolge der wiederholten Bereiche zu verändern. In jeden einzelnen dieser Bereiche können Sie dann Ihre Inhalte, wie in der Vorlage definiert, einfügen.

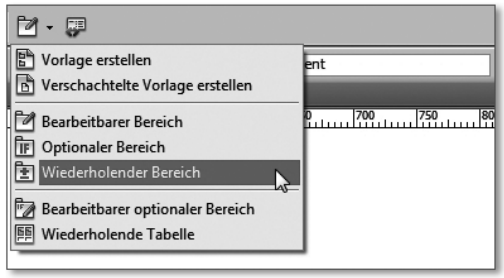


Abbildung 13.9 Wiederholenden Bereich einfügen

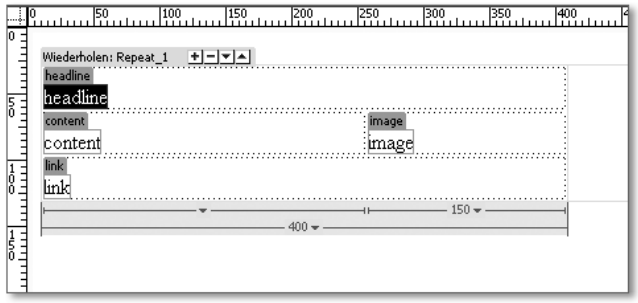


Abbildung 13.10 Wiederholter Bereich aus einer Vorlage

13.2.3 Wiederholte Tabelle

Wenn es sich bei den wiederholten Bereichen um Tabellenzeilen handelt, können Sie dies auf einfache Weise mit der Funktion WIEDERHOLENDE TABELLE erreichen. Sie können die gesamten Tabellenparameter vorgeben und dann im Dokument die Anzahl der Tabellenreihen erhöhen oder die Reihenfolge der Inhalte verändern. Auch hier können Inhalte in die bearbeitbaren Bereiche eingefügt werden.

Noch vielfältiger werden die Möglichkeiten, wenn Sie Vorlagen mit Bibliotheken kombinieren.

13.3 Mit Bibliotheken arbeiten

Bibliotheken erfüllen einen ähnlichen Zweck wie Vorlagen. Der Unterschied besteht darin, dass in Bibliotheken keine kompletten Seiten abgelegt werden, sondern einzelne, häufig verwendete Elemente innerhalb der Seite.

Es können ganze Tabellen, Bildfolgen oder Navigationen in einer Bibliothek hinterlegt werden. Diese Bibliothekselemente können dann schnell und einfach in ein Dokument integriert werden.

Auch bei Bibliotheken besteht eine Abhängigkeit zwischen den aus Bibliothekselementen erstellten Inhalten und den Elementen in der Bibliothek. Werden die Bibliothekselemente verändert, ändern sich auch alle anhängenden Inhalte.

Vorlagenseiten sind also komplette Seiten, bei denen nur ein Bereich auf den anhängigen Seiten individuell verändert werden kann. Bibliothekselemente dagegen sind zentral verwaltete Bereiche, die auf individuellen Seiten eingesetzt werden.

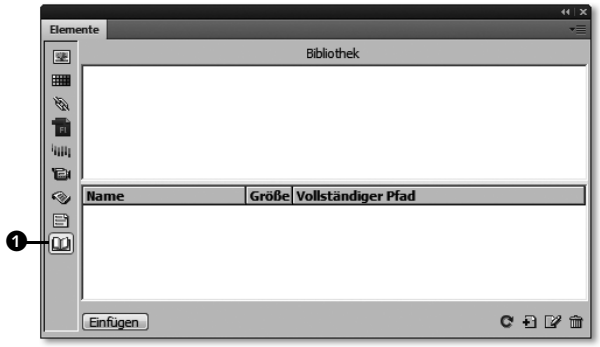


Abbildung 13.11 Bibliothekselement im Bedienfeld ELEMENTE

Um mit der Bibliothek zu arbeiten, öffnen Sie wieder Ihr Bedienfeld ELEMENTE und wählen BIBLIOTHEK aus. Das ist das kleine Buch ❶ in der Symbolleiste links.

Sie können jetzt ein beliebiges Element oder ganze Bereiche mit funktionierender Programmierung aus Ihrem Dokument in das Bibliotheksfenster ziehen, um es in die Bibliothek einzufügen.

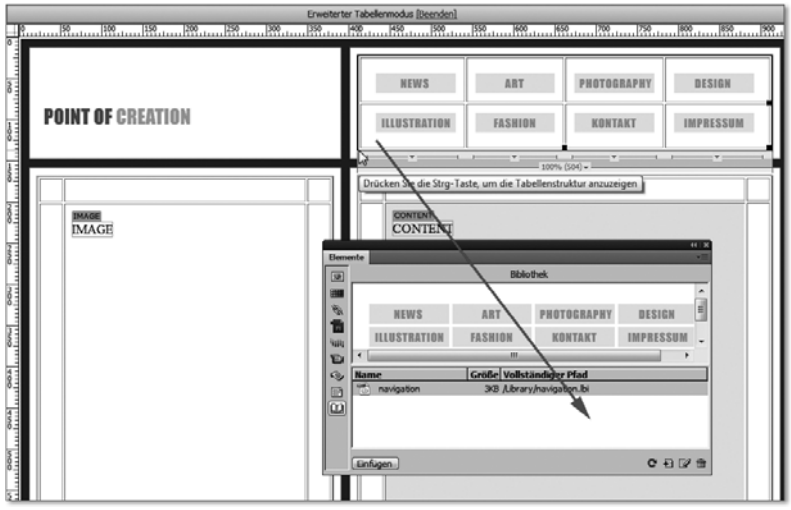


Abbildung 13.12 Element in die Bibliothek ziehen

Im Dokument werden diese Elemente dann grau angezeigt und sind nicht mehr direkt bearbeitbar. Für die Bibliothek legt Dreamweaver ebenfalls einen neuen Ordner namens *Library* an. Dort befinden sich die gesamten Bibliothekselemente mit der Dateierweiterung *.lbi*. Im Prinzip ist das nichts anderes als Codefragmente aus Ihrem Dokument. Möchten Sie nun eines dieser Elemente in Ihre Seite einbauen, brauchen Sie es nur in das Dokument zu ziehen.

Navigationen in der Bibliothek ablegen

Möchten Sie hier Navigationen ablegen, brauchen Sie sich über die Pfade keine Gedanken zu machen. Diese werden immer aktuell in Bezug auf das Dokument, in dem sich das Bibliothekselement befindet, neu gesetzt.

Codefragmente

Für die Erstellung immer wiederkehrender Elemente können Sie auch mit Codefragmenten – häufig auch als *Snippets* bezeichnet – arbeiten. Bei Codefragmenten ist die Abhängigkeit zwischen hinterlegten Elementen und aktuellen Dokumenten jedoch nicht gegeben, so dass sich diese Vorgehensweise zum schnellen Aktualisieren einer Website nicht eignet.

Ein neues Codefragment erstellen Sie mit einem Klick auf das Plus-Symbol. Nun können Sie das Codefragment aus einem Bereich bestehen lassen – in Dreamweaver mit der Option Block bezeichnet –, der dann an der aktuellen Position eingefügt wird. Alternativ kann das Codefragment auch aus zwei Teilen (Umbruch für Auswahl) bestehen. Diese werden dann beim Einfügen um den entsprechend markierten Bereich eingefügt, also davor und danach.

Zum Einfügen eines Codefragments, das aus zwei Teilen besteht, markieren Sie einen entsprechenden Bereich in der Codeansicht. Nach dem Klick auf die Bezeichnung des Codefragments in der Palette werden die Texte des Codefragments vor und nach dem markierten Bereich eingefügt.

Codefragmente werden häufig genutzt, um mehr oder weniger lange Codestücke zu speichern und sie auf Klick in eine Seite einzufügen. Beispiele dafür sind fertige, aber leere Meta-Tag-Bereiche für den Head einer Website. So brauchen Sie diese nicht jedes Mal neu zu schreiben. Klicken Sie einfach auf ein Codefragment in der Palette. Dreamweaver fügt an der aktuellen Position im Quellcode den Text bzw. Code aus dem Codefragment ein. In unserem Beispiel müssen Sie sich also in der Codeansicht im Head einer Seite befinden. Nach dem Einfügen können Sie dann die einzelnen Inhalte der Meta-Tags bearbeiten.

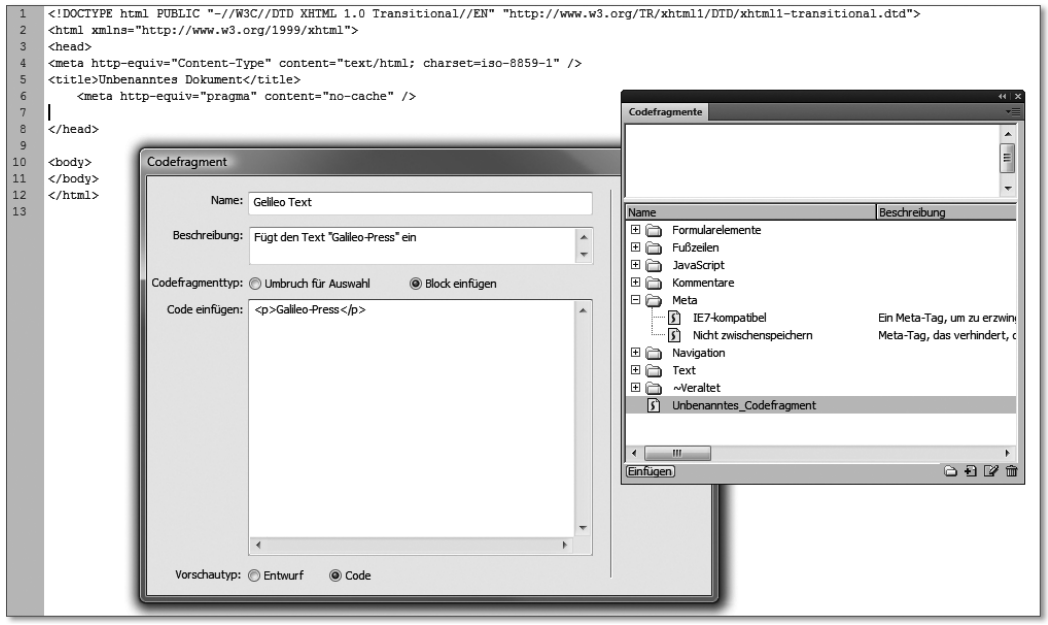


Abbildung 13.13 Ein neues Codefragment erstellen – hier ein Codeblock

Bibliotheken und JavaScript/CSS

[«]

JavaScripts teilen sich meistens in zwei Teile auf. Ein Teil des Skripts steht im Head des Dokuments, der zweite Teil im Body. In Bibliotheken werden nur Elemente aus dem Body eingefügt. Wenn Sie diese Elemente dann in Dokumenten ohne den dazugehörigen Teil des Skripts verwenden, funktioniert Ihr JavaScript nicht.

Ähnliches trifft auf die Verwendung externer CSS-Dateien zu. Die Formatierungen werden in der Bibliothek nicht korrekt wiedergegeben, da die Verlinkung zur CSS-Datei in der Bibliothek nicht vorhanden ist.

Wenn es interaktiv und dynamisch werden soll, kommen Sie an Formularen nicht vorbei. Wir widmen uns diesem Thema recht ausführlich, auch im Hinblick auf den dynamischen Teil des Buches.

14 Formulare

Formularelemente und Formulare werden im Internet überall da benötigt, wo ein User eine Eingabe machen kann, um Daten an einen Server zu senden. Sie sind das meistgenutzte Tool für Interaktivität im Internet.

Um ein Formular zu nutzen, benötigen Sie Folgendes:

- das Formular mit den Informationen über die Versandart der Daten
- die darin enthaltenen Formularelemente als Eingabeoberfläche für den Benutzer; nur die Inhalte, die in Formularelemente eingegeben werden, werden zurückgeschickt
- eine auswertende Seite oder einen Bereich, der die Daten auswerten und verarbeiten kann. Dies wird im Regelfall mit serverseitigen Sprachen (z. B. PHP) realisiert. Hier betrachten wir die Auswertung jedoch nicht weiter, sondern verweisen auf die späteren Abschnitte mit PHP.

14.1 Übertragungsmethoden für Formulardaten

Es gibt zwei verschiedene Methoden, um Daten aus einem Formular zu versenden. Jede hat ihren eigenen Einsatzzweck:

- `GET`
- `POST`

14.1.1 GET

Die Methode `GET` ist im Internet die Standardmethode. Dazu werden Daten aus dem Formular einfach an die URL angehängt und an den Server gesendet. Sie kennen sicher die vielen Zahlen bei GMX oder Web.de in der Adressleiste des Browsers. Das sind Daten, die mit `GET` verschickt wurden.

Der Nachteil dieser Methode ist die völlig offene Übertragung der Daten. Für sensible Daten ist dieser Weg daher nicht geeignet. Außerdem ist die Menge der Daten, die auf diesem Weg versandt werden kann, auf 2.000 Zeichen (inklusive der Adresse der Seite) beschränkt.

Der Vorteil dieser Methode liegt darin, dass Sie diese Websites als Lesezeichen bzw. Favoriten ablegen können. Außerdem sind die Daten leicht sichtbar zu machen, wenn Sie bei der Programmierung auf Fehlersuche sind.

14.1.2 POST

Die zweite Methode, `POST`, eignet sich besser, um Daten in größeren Mengen und verdeckt zu versenden.

14.1.3 Formulare per E-Mail

Eine immer wieder genutzte Möglichkeit ist die Übertragung der Formularinhalte per Mail. Diese Methode kann jedoch als unprofessionell bezeichnet werden. Sie setzt auf User-Seite ein Mailprogramm voraus und ist damit vollständig vom Client abhängig. Genau das will man eigentlich durch den Einsatz von Formularen vermeiden.

[»] **Formulardaten auswerten**

Wenn Sie keine Skripte auf Ihrem Webserver laufen lassen können, die dafür sorgen, dass Maildaten richtig ausgewertet und versendet werden, können Sie einen der reichlich vorhandenen Dienstleister für den Formularversand nutzen (z. B. *http://www.formmailer.de/*).

Wenn Sie dies nicht möchten und auch kein Skript einsetzen können, fügen Sie dem `<form>`-Tag den Parameter `enctype="Text/plain"` hinzu. Dieser sorgt dafür, dass der Text in den Mails für Sie und das Mailprogramm halbwegs lesbar ist. Outlook hat dennoch immer wieder Probleme mit Mails aus Formularen.

14.2 Ein Formular erstellen

Um mit Formularelementen arbeiten zu können, müssen Sie innerhalb eines HTML-Dokuments zunächst einen Bereich definieren, der als Formular dienen soll. Nur innerhalb dieses Bereichs können Formularelemente Daten entgegennehmen und an eine Ziel-URL senden.

```
<body>
<form id="form1" name="form1" method="post" action="auswertung.php">
```

```
  HIER KOMMEN DIE FORMULARELEMENTE HIN
</form>
</body>
```

Listing 14.1 Aufbau eines Formulars in HTML

Selbstverständlich können Sie innerhalb des Formular-Tags auch andere XHTML-Elemente verwenden. Wichtig ist nur, dass alle Formularelemente wirklich vom Formular-Tag umschlossen werden.

Richtige Reihenfolge

[«]

Wenn Sie versuchen, einem HTML-Dokument ein Formularelement hinzuzufügen, ohne vorher ein Formular erstellt zu haben, fordert Dreamweaver Sie dazu auf und fügt selbständig das `<form>`-Tag direkt um das Formularelement hinzu. Sobald Sie dann ein weiteres Formularelement hinzufügen möchten, passiert das Gleiche wieder, so dass Sie am Ende mehrere Formulare in einem Dokument haben. Achten Sie darauf, immer zunächst das Formular selbst zu erstellen, und fügen Sie dann die Formularelemente in das Formular ein.

Prinzipiell können Sie mehrere Formulare in ein XHTML-Dokument einbinden. Beim Absenden eines Formulars werden allerdings nur die Daten eines einzigen Formulars übergeben. Die Daten aus den anderen Formularen können nur mit Hilfe von JavaScript ausgelesen und gleichzeitig übertragen werden.

Klicken Sie im Register FORMULARE ❶ auf das Icon ❷, um ein Formular in Ihr Dokument einzufügen. Es wird Ihnen durch einen roten Rand angezeigt ❸. Bedenken Sie dabei, dass dies nicht der tatsächlichen Ansicht im Browser entspricht.

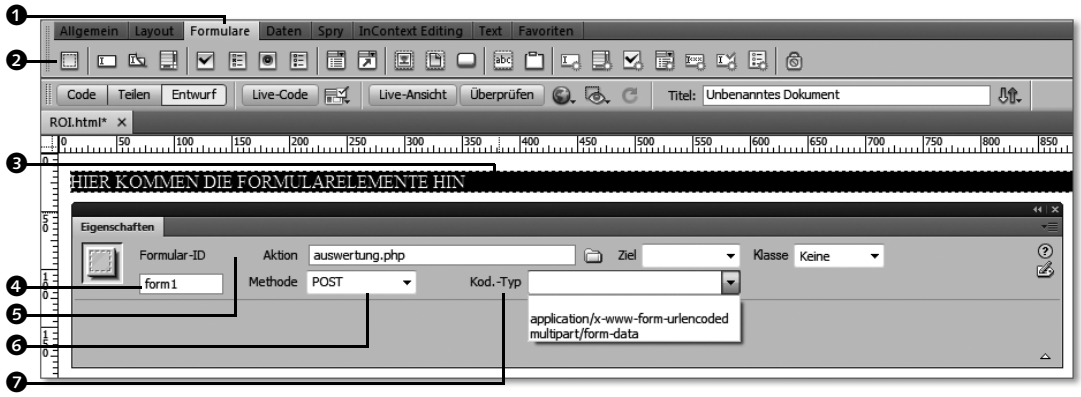


Abbildung 14.1 Ein Formular mit Dreamweaver erstellen

Formulare werden automatisch benannt, in unserem Beispiel mit der Vorgabe von Dreamweaver ❹. Diese Benennung ist vor allem dann wichtig, wenn es in

einem Dokument mehrere Formulare gibt oder einzelne Elemente im Formular mit JavaScript angesprochen werden sollen.

Der Zielort für die Daten des Formulars wird in den meisten Fällen ein PHP-Skript sein, das die Daten auswertet. Tragen Sie bei Aktion ❺ den Pfad zur auswertenden Seite ein. Dies kann ein absoluter oder relativer Pfad sein. Über die Übertragungsmethoden haben wir eben bereits gesprochen. Im Menü Methode ❻ können Sie Get oder Post auswählen.

Verschiedene Kodierungstypen ❼ können notwendig sein, wenn Sie z. B. keine Text-, sondern Binärdaten, also Bilder oder Ähnliches übertragen möchten.

14.3 Formularelemente

Um dem Benutzer die Möglichkeit zu geben, angeleitet Daten eintragen zu können, benötigt er grafische Eingabefelder, wie Checkboxen, Radiobuttons, Auswahlmenüs usw.

14.3.1 Einfügen mit Eingabehilfe

Wenn Sie in den Voreinstellungen die Eingabehilfe für Formulare aktiviert haben, erscheint für jedes Formularelement eine Dialogbox.

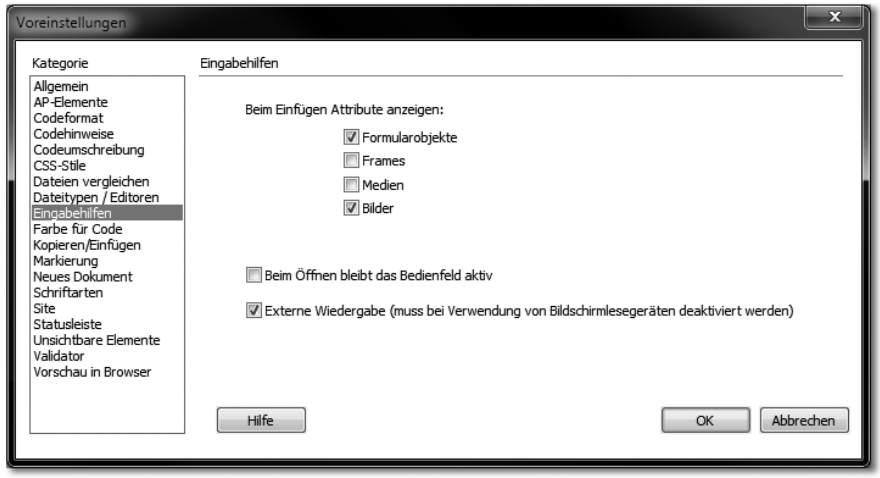


Abbildung 14.2 Voreinstellungen zu Eingabehilfen

Wichtig sind die hier vorzunehmenden Einstellungen, wenn Sie eine barrierefreie Website erstellen wollen. Einige Angaben zu Formularelementen, wie zum Beispiel einen Bezeichner, der in Bildschirmlesegeräten ausgegeben wird,

können Sie nur an dieser Stelle vornehmen. Später, in der Eigenschaftenpalette, stehen sie nicht mehr zur Verfügung.

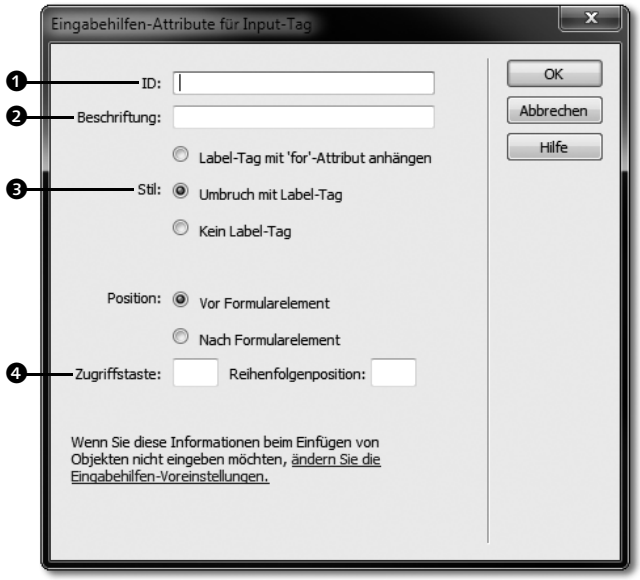


Abbildung 14.3 Eingabehilfe für Formularelemente

Eine ID ❶ wird benötigt, um das Formularelement mit JavaScript anzusprechen oder auch um mit CSS gezielt dieses Element zu formatieren. Die BESCHRIFTUNG ❷ kann in verschiedenen Stilen ❸ angelegt werden. Sie ist in erster Linie für Bildschirmlesegeräte gedacht, kann aber auch gut mit CSS als Beschriftung für das jeweilige Element verwendet werden.

Die ZUGRIFFSTASTE ❹ ist eine frei definierbare Tastatureingabe, die einen direkten Zugriff auf das Formularelement ermöglicht. Viele User sind es gewohnt, mit der ⇆-Taste durch Formulare zu navigieren. Sie können hier zudem die REIHENFOLGEPOSITION festlegen, an der das gerade eingefügte Element stehen soll.

14.3.2 Namensvergabe

Daten aus Formularen sind fast ausschließlich Variablen, die über PHP-Skripte weiterverwendet werden sollen. Sehr wichtig ist dafür die exakte Schreibweise der *Variablennamen*.

Achten Sie dabei genau auf Groß- und Kleinschreibung. Die Bezeichnungen `nameCONT` und `NameCONT` sind z. B. nicht identisch. Auch in diesem Fall gelten die üblichen Regeln für Linux-/Unix-Dateisysteme mit der zusätzlichen Einschränkung, dass Sonderzeichen aller Art außer dem Unterstrich verboten sind. Achten

Sie auch darauf, dass Zahlen als Bestandteil von Variablen immer am Ende des Namens stehen, nicht am Anfang der Bezeichnung:

- richtig: `name_1`, `name_2`
- falsch: `1name`, `Name t`, `Änderung` usw.

Die *Zuweisung von Variablen* geschieht in der Eigenschaftenpalette oder im Quellcode direkt. Jedes Formularelement kann hier mit dem Namen einer Variablen verknüpft werden. Im Quellcode hat daher jedes Formularelement das Attribut `name`.

14.3.3 Textfelder

Textfelder werden bei der freien Eingabe von Namen, Zeichenketten, Zahlen usw. eingesetzt. Sie sind die wohl häufigsten Formularelemente überhaupt. Sie können sie mit dem Icon ❶ einfügen.

Neben der Anzahl der Zeichen und einer Zeichenbreite können Sie angeben, ob das Feld als *Kennwortfeld* funktionieren soll oder nicht. Wenn Sie die Checkbox Kennwort ❷ aktivieren, wird der vom Benutzer eingegebene Inhalt durch schwarze Punkte oder Sternchen überdeckt.

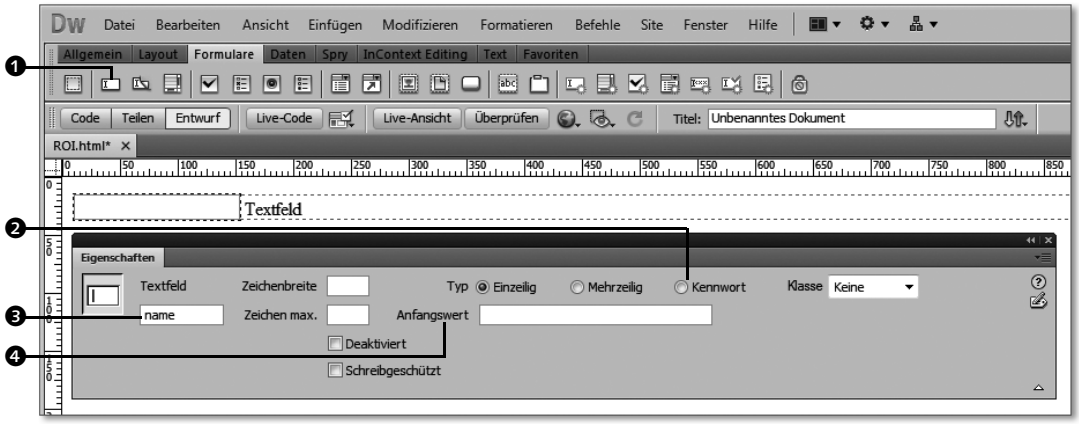


Abbildung 14.4 Textfelder mit Parametern

In der Eigenschaftenpalette für Textfelder werden im Feld Textfeld ❸ die Namen der Variablen zugewiesen. Der Inhalt der Variablen ergibt sich dann aus dem Inhalt des Eingabefeldes auf der Website oder aus dem, was Sie im Feld Anfangswert ❹ als Vorgabe eintragen.

Über die Zeichenbreite kann man sehr grob die Breite des Formularfeldes bestimmen. Leider werden diese Werte von den gängigen Browsern sehr unter-

schiedlich interpretiert. Genau sind die Zuweisungen per CSS. Die Eigenschaft lautet `width`.

14.3.4 Versteckte Textfelder

Eine immer wiederkehrende Aufgabe ist es, Daten von einem Dokument zum nächsten zu versenden. Notwendig wird dies z. B. bei einem Shopsystem. Auf der ersten Seite werden die Bestellungen angezeigt, auf der nächsten müssen Sie die Adressdaten eintragen und danach die Zahlungsoptionen.

Ohne eine verlässliche Möglichkeit, Daten zwischen diesen Dokumenten auszutauschen und über mehr als ein Dokument zu erhalten, sind komplexe Websites nicht realisierbar.

Webserver senden HTML-Dokumente an den Browser; Daten, die per `POST` oder `GET` gesendet werden, sind nirgendwo abgespeichert. Um nun diese Daten auch in Folgedokumenten zu erhalten, müssen sie erneut mit `POST` oder `GET` gesendet werden. Da dies nur mit Daten in Formularen möglich ist, muss ein Formularfeld die zuvor erhaltenen Daten aufnehmen.

Diese Aufgabe erledigen versteckte Felder. Sie dienen als Behälter für Daten, die nicht angezeigt, aber dennoch erhalten bleiben sollen. Sie können ein verstecktes Textfeld über das Icon ❺ einfügen.

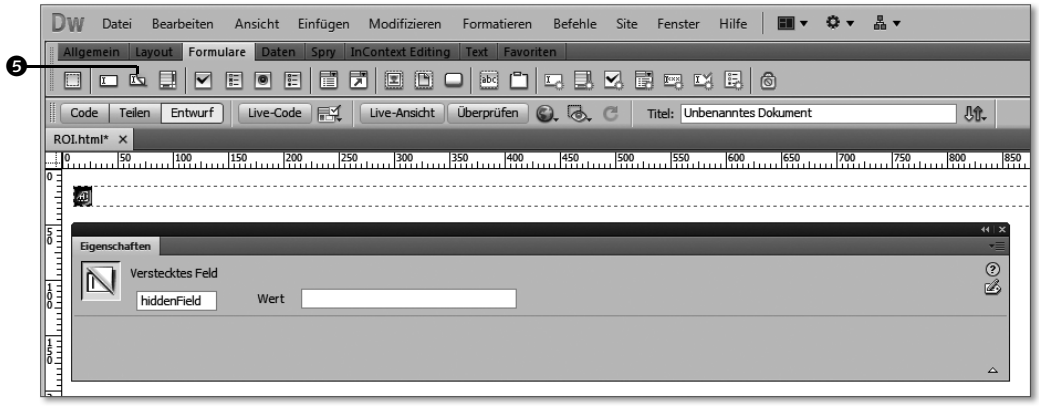


Abbildung 14.5 Versteckte Felder und Parameter

Da versteckte Felder im Prinzip nur unsichtbare Textfelder sind, verhalten sie sich ansonsten auch wie das Formularelement Textfeld. In Abbildung 14.6 sehen Sie das Funktionsprinzip der Datenweiterleitung schematisch dargestellt.

Wenn Sie eine Site entwickeln, ist es wichtig, alle Daten im Dokument sehen zu können. Sie können deshalb zu Beginn alle später versteckten Felder als Textfel-

der einbinden. Wenn Sie dann fertig sind und alles wie gewünscht läuft, ändern Sie im Quelltext `input type="text"` einfach in `input type="hidden"`. Die Textfelder werden so zu versteckten Feldern.

[!] **Versteckte Felder sind nicht unsichtbar!**

Die Inhalte von versteckten Feldern sind für jeden Besucher der Website leicht sichtbar. Sie sind im Quelltext einzusehen und können gegebenenfalls manipuliert werden. Daher dürfen Sie keinesfalls geheime Passwörter oder TANs in einem versteckten Feld übergeben. Professionelle Anwendungen speichern Daten, die weitergegeben werden müssen, in Datenbanken oder in Sessions zwischen.

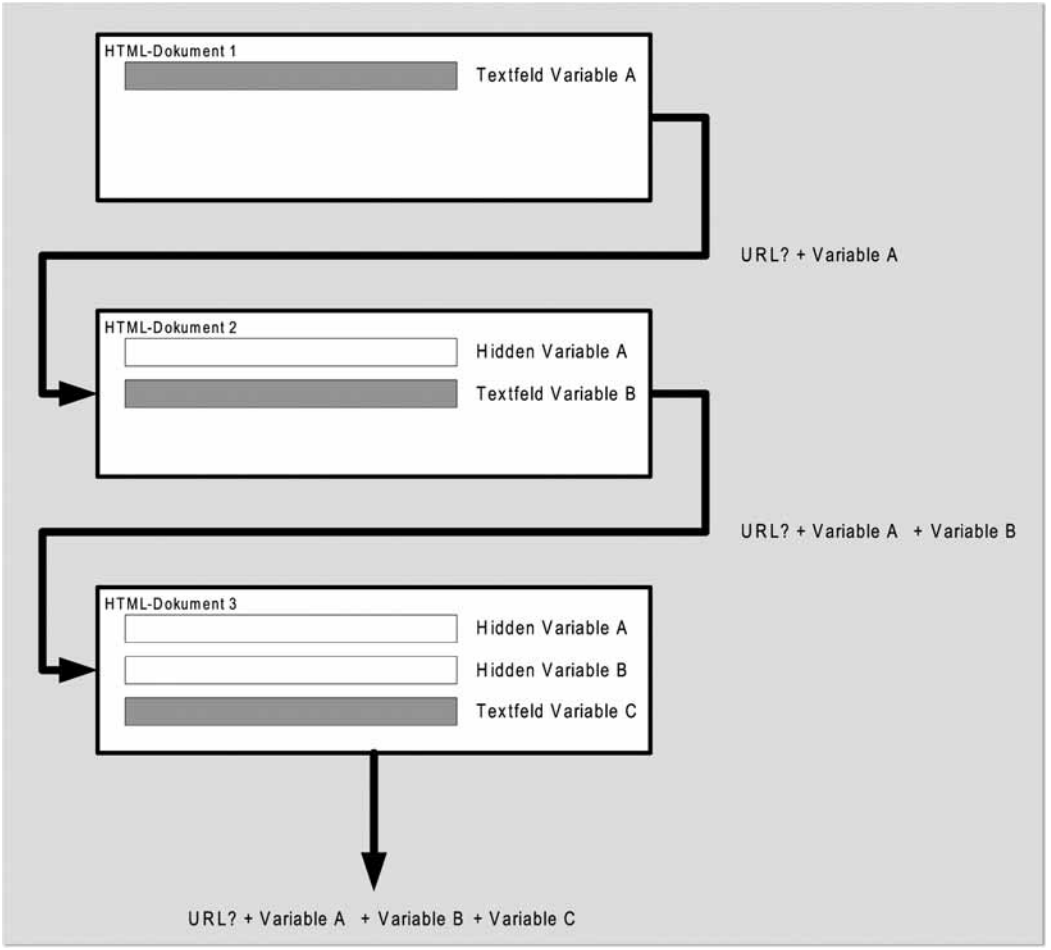


Abbildung 14.6 Funktionsprinzip von versteckten Feldern

14.3.5 Textbereiche

Textbereiche (`textarea`) sind geeignet, größere Textmengen aufzunehmen und darzustellen. Sie können sie mit dem Icon ❶ einfügen.

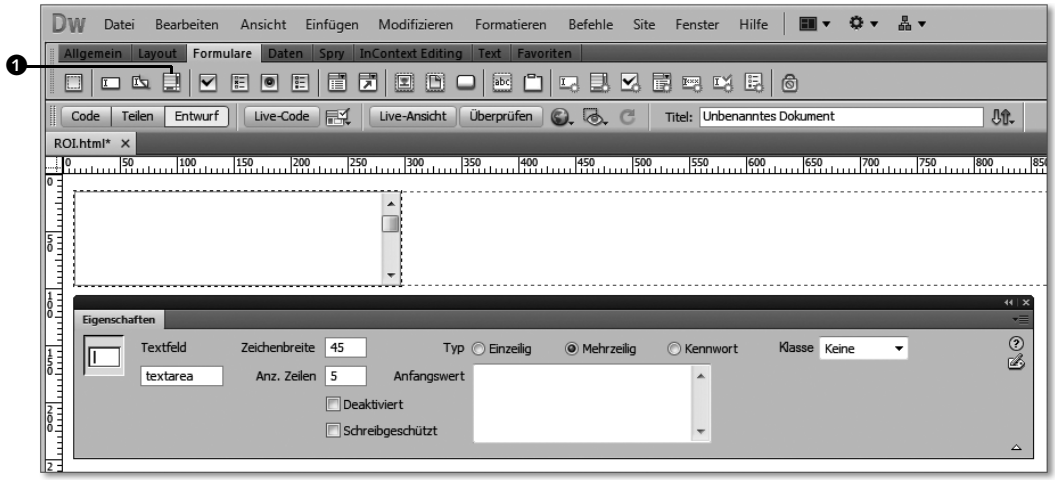


Abbildung 14.7 Textbereiche und Parameter

14.3.6 Kontrollkästchen

Mit Kontrollkästchen (Checkboxen) können Sie einzelne Optionen aktivieren oder deaktivieren. Je nach aktiviertem oder deaktiviertem Zustand werden die Variableninhalte übertragen oder nicht. Von einer Gruppe aus Kontrollkästchen können mehrere gleichzeitig ausgewählt werden. Mit ANFANGSSTATUS legen Sie den Default-Zustand des Elements fest. Ein einzelnes Kontrollkästchen können Sie mit dem Icon ❷ einfügen, eine Kontrollkästchengruppe mit dem Icon ❸.

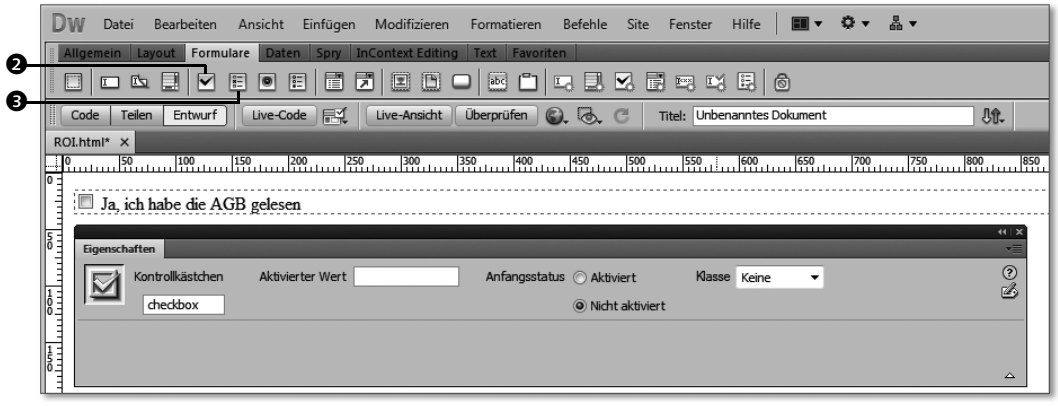


Abbildung 14.8 Ein Kontrollkästchen und die dazugehörigen Parameter

In der Eigenschaftenpalette von Kontrollkästchen und Optionsfeldern müssen Sie neben der Zuweisung des Variablennamens vorgeben, welcher Wert bei Aktivierung des Feldes als Variableninhalt übertragen werden soll. Dazu wird der Variableninhalt in das Feld Aktivierter Wert eingetragen. Im Quelltext erhält das Tag dann das Attribut `value` mit dem von Ihnen eingegebenen Wert.

14.3.7 Optionsschalter

Optionsschalter (Radiobuttons) unterscheiden sich von Kontrollkästchen durch die Möglichkeit einer Gruppierung. Von Optionsschaltern mit der gleichen Benennung kann immer nur eins ausgewählt werden. Die anderen, nicht angewählten Schalter werden automatisch deaktiviert. Eine Gruppierung von Optionsschaltern legen Sie über die gleiche Benennung fest. Die Benennung legt daher den Variablennamen fest, der Inhalt von Aktivierter Wert den Wert der zu übertragenden Variable. Da eine Variable immer nur einen Wert annehmen kann, ist nur die Auswahl eines Kästchens innerhalb einer Gruppe möglich. Sie können einen einzelnen Optionsschalter mit dem Icon ❶ einfügen, eine Optionsschaltergruppe mit dem Icon ❷.

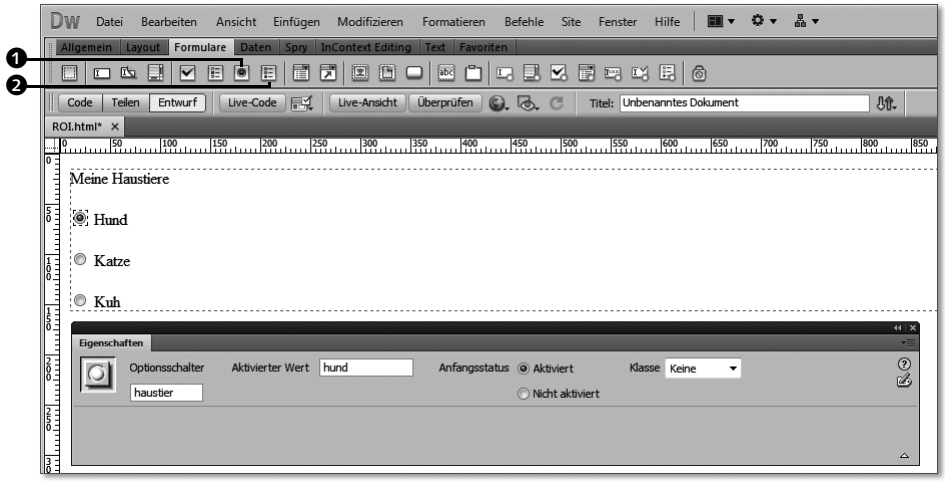


Abbildung 14.9 Gruppe von Optionsschaltern

Mit *Optionsschaltergruppen* können Sie ganze Gruppen von Schaltern auf einmal erstellen. Diese werden dann automatisch von Dreamweaver benannt.

14.3.8 Auswahllisten und Sprungmenüs

Längere Auswahllisten bieten viele verschiedene Auswahlmöglichkeiten bei gleichzeitig geringem Platzbedarf. Sie fügen sie über das Icon ❸ ein.

Klicken Sie auf den Button Listenwerte ❼, um zu einer Dialogbox zu gelangen, in der Sie die Wertzuweisungen eintragen können. Im Feld Elementbezeichnung ❻ werden die anzuzeigenden Auswahlwerte eingetragen, im Feld Wert dann der dazugehörige Inhalt der Variablen. Zusätzlich können Sie bei Zuerst ausgewählt ❺ anwählen, welcher Wert zuerst selektiert werden soll.

Sprungmenüs sind den Listen sehr ähnlich, rufen aber je nach angewähltem Wert direkt eine URL auf. Dazu wird allerdings von Dreamweaver zusätzlich JavaScript eingefügt. Sie können Sprungmenüs mit dem Icon ❹ einfügen.

Sprungmenüs sind allerdings nicht mehr aktuell und sollten vermieden werden, da sie alles andere als benutzerfreundlich sind. Die Möglichkeiten zur Navigation werden von unerfahrenen Benutzern einfach oft übersehen. Dennoch werden dynamische Sprungmenüs häufig bei komplexen Entscheidungen (beispielsweise bei Fahrzeug-Konfiguratoren) genutzt, um sofort bei Auswahl die anderen Felder zu aktualisieren. Dazu wird Ajax verwendet.

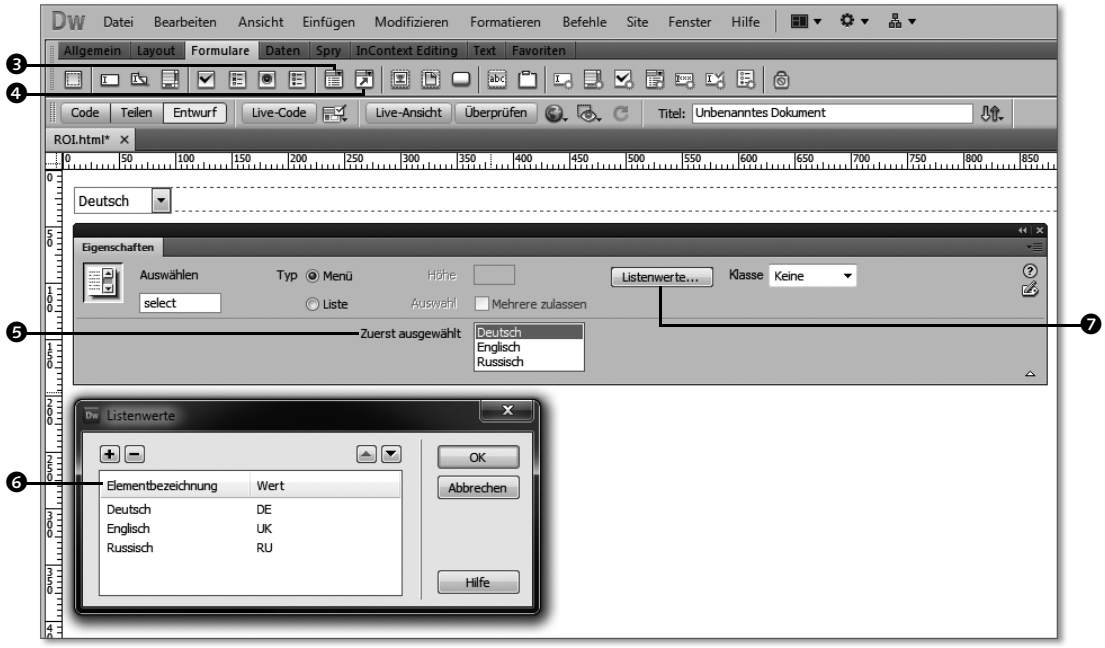


Abbildung 14.10 Auswahlliste mit Werten

14.3.9 Dateifelder

In vielen Websites ist es erwünscht, Dateien auf einfache Art und Weise auf den Server zu übertragen, beispielsweise für einen Bildupload. Hierfür bieten Dateifelder dem Benutzer die Möglichkeit, die gewünschten Dateien auf seiner Fest-

platte auszuwählen und zu senden. Sie müssen in diesem Fall das Formular mit der Vorgabe `enctype="multipart/form-data"` codieren, damit die Daten korrekt übertragen werden. Dateifelder fügen Sie mit dem Icon ❶ ein.

Serverseitig sind allerdings zusätzliche Programmierungen und Prüfungen nötig, um die Dateien zu verarbeiten und Missbrauch zu verhindern. Dreamweaver deckt hier leider den Bedarf nicht ab und es muss mit eigenen Scripts nachgeholfen werden.

Das Aussehen und die Beschriftung des Buttons DURCHSUCHEN können übrigens nicht verändert werden. Diese werden immer vom System vorgegeben – ein großes Manko bei den Browsern.

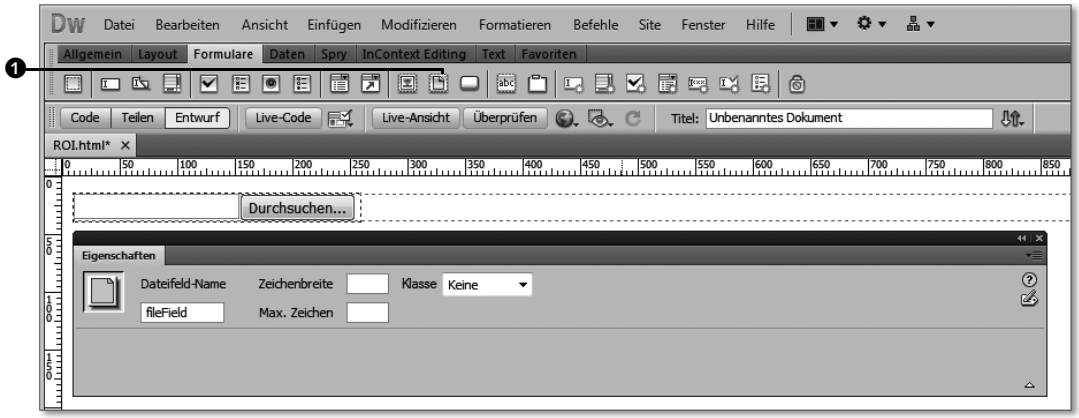


Abbildung 14.11 Dateifeld in der Vorschau

14.3.10 Schaltflächen

Jedes Formular benötigt eine Schaltfläche zum Absenden der Daten. Sie haben dabei die Möglichkeit, der Schaltfläche eigene Aktionen zuzuweisen und die Beschriftung zu ändern. Die Beschriftung ist dabei gleich dem übermittelten Wert, was bei der Auswertung des Formulars beachtet werden muss. Schaltflächen fügen Sie mit dem Icon ❷ ein.

[»] **Eigene Buttons oder Textlinks als Schaltflächen einbauen**

Es kann vorkommen, dass Sie anstelle der typischen Formularschaltflächen eigene Bilder einsetzen wollen oder der Klick auf einen verlinkten Text (z. B. »weiter«) das Formular absenden soll. Fügen Sie dazu das Formularelement BILDFELD mit dem gewünschten Bild in das Formular ein. Text können Sie wie gewohnt verlinken und mit dem Platzhalterziel # versehen. Ändern Sie anschließend im Quelltext das Tag, indem Sie `onClick="submit()"` hinzufügen. Jetzt haben Sie aus dem Bild eine Schaltfläche erstellt, und das Formular wird beim Anklicken abgeschickt.

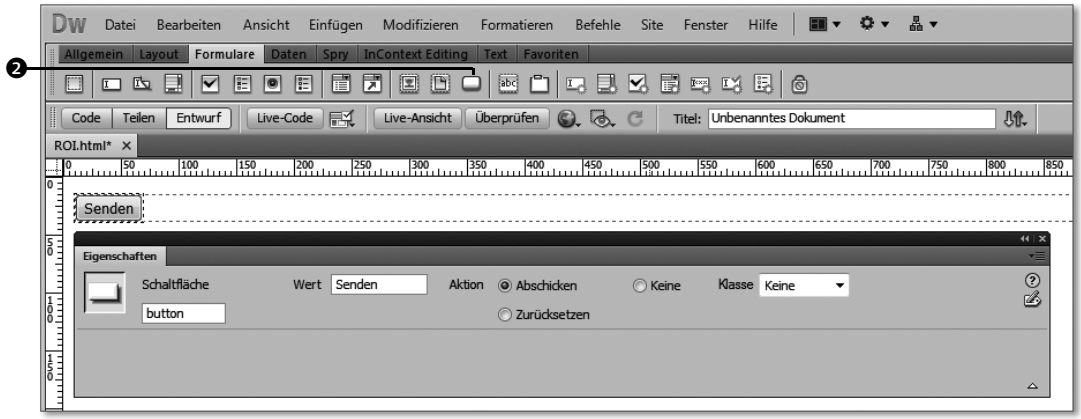


Abbildung 14.12 Schaltfläche einfügen

14.4 Überprüfung von Formularen

Unvollständige Formularangaben sind ärgerlich. Ohne Gegenmaßnahmen kommt es häufig vor, dass etwa Kundenanfragen ohne E-Mail, Telefonnummer oder andere wichtige Angaben abgeschickt werden. Damit dies vermieden wird, sollten Formulare immer auf die vollständige Eingabe der wichtigsten Daten überprüft werden.

In Dreamweaver haben Sie die Möglichkeit, dies einfach zu realisieren. Die Spry-Formular-Überprüfungen helfen Ihnen dabei und sind einfach anzuwenden. Dank der kompletten Einbindung in Dreamweaver sind sie zudem leicht zu konfigurieren.

14.4.1 Spry für Formulare

Sie Eingabe können die Daten der Benutzereingaben bereits während der überprüfen und gegebenenfalls Hilfestellungen einblenden. Zur Überprüfung muss das Dokument nicht erneut geladen werden. Erst der vollständige und verifizierte Datensatz wird an eine Ziel-URL übertragen.

Fehlermeldung bei lokaler Anzeige? [«]

Lesen Sie in Abschnitt 12.1.3, »Fehlermeldung bei lokaler Anzeige«, nach, wie Sie diese beheben können.

Die Spry-Überprüfungen finden Sie im Register Formulare und im eigenen Register Spry für Spry-Elemente.

[!] **Variablennamen bei Spry-Elementen für Formulare**

Der bei aktiviertem Spry-Element in der Eigenschaftenpalette angezeigte Name (z. B. `sprytextfield1`) ist *nicht* der Name der zu übertragenden Variablen! Um die Feldbezeichnung mit Variablennamen einzugeben, klicken Sie direkt auf das Formularelement und nicht auf den Karteireiter über dem Element. Verfahren Sie dann wie bei jedem anderen Formularfeld auch.

Spry-Dateien kopieren

Mit jedem eingefügten Spry-Element kopiert Dreamweaver die notwendigen Dateien in einen Ordner namens *SpryAssets*. Dieser Ordner muss auf den Webserver kopiert werden, wenn Sie die Website veröffentlichen möchten und diese auch im Internet funktionieren soll.

14.4.2 Spry-Überprüfung Textfeld

Aktivieren Sie nach dem Einfügen einer Spry-Überprüfung Textfeld ❶ das kleine türkisfarbene Register über dem Spry-Element im Dokumentfenster ❷ (siehe Abbildung 14.13), um die Parameter in der Eigenschaftenpalette einzustellen.

Wir können zwei Varianten der Spry-Überprüfung für ein Textfeld unterscheiden:

- ohne Überprüfung der Eingabe auf ein bestimmtes Datenformat (Typ)
- mit Überprüfung der Eingabe auf ein bestimmtes Datenformat (Typ)

Unter Datenformat ist in diesem Fall beispielsweise eine Telefonnummer, ein Datum oder eine E-Mail-Adresse zu verstehen.

Welche der beiden Varianten Sie verwenden, hängt davon ab, ob Sie in der Eigenschaftenpalette unter Typ eine Typprüfung einstellen oder nicht, doch hierzu gleich mehr.

Wir haben diese Unterteilung gewählt, weil sich so die Spry-Überprüfung leichter erklären lässt. In der Dokumentation und in der Hilfedatei von Adobe besteht diese Unterscheidung nicht! Im Folgenden verwenden wir außerdem mehrfach den Begriff *Ereignis*. Darunter ist eine Maus- oder Tastaturaktion des Benutzers bzw. des Website-Besuchers zu verstehen. In Kapitel 9, »JavaScript und Verhalten«, haben wir dies in der Einleitung beschrieben.

[»] **Große externe JavaScript-Datei**

Bei der Verwendung der Spry-Überprüfung Textfeld kopiert Dreamweaver die JavaScript-Datei *SpryValidationTextField.js* in das Verzeichnis *SpryAssets*.

Diese JavaScript-Datei ist 74 KByte groß. In dieser Datei sind alle möglichen Varianten der Spry-Überprüfung Textfeld integriert. Da diese Datei als externe JavaScript-Datei im HTML-Dokument eingebunden wird, werden auch alle Skriptteile im Browser geladen – auch dann, wenn Sie diese gar nicht benötigen! 74 KByte zuzüglich des HTML-Dokuments mit Grafiken sind für eine DSL-Leitung keine Herausforderung, für die immer noch vorhandenen Modem- und ISDN-Anbindungen aber sehr wohl. Es ist sinnvoll, sich genau zu überlegen, wann man dieses Element einsetzt und wann nicht.

Spry-Überprüfung Textfeld ohne Formatüberprüfung

Dieses Spry-Textfeld kennt drei verschiedene Anzeigezustände. Die Umschaltung zwischen den verschiedenen Anzeigen erfolgt über das Auswahlmenü Zustandsvorschau ❻. Die drei Anzeigezustände sind folgende:

Ereignis	Zustandsvorschau (Dreamweaver)	Nummer im Bild
keinerlei Ereignis	Anfänglich	❸
Ereignis und kein Wert eingegeben	Erforderlich	❹
Ereignis und Wert eingegeben	Gültig	❺

Tabelle 14.1 Zustandsvorschau ohne Überprüfung eines bestimmten Formats

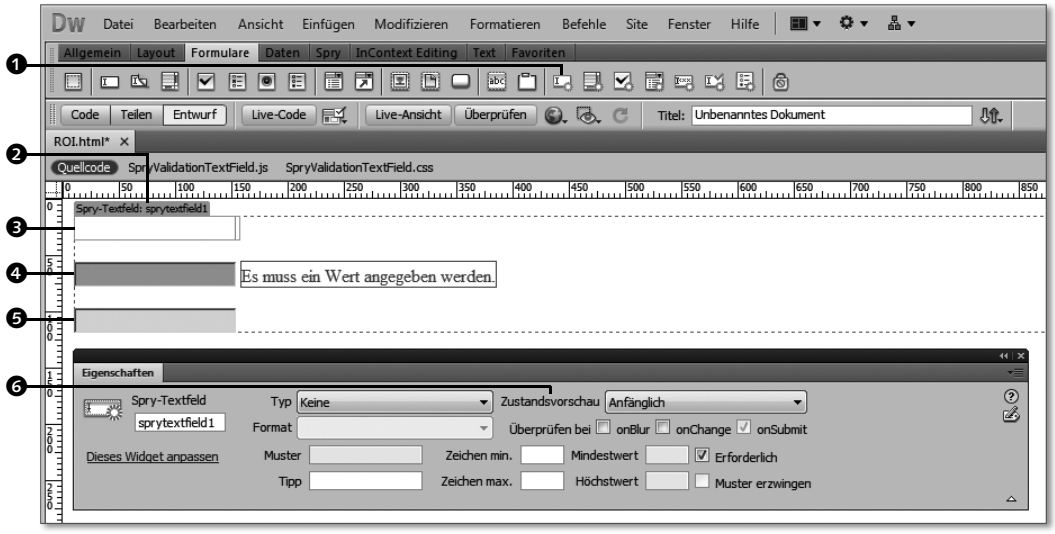


Abbildung 14.13 Vorschaumöglichkeit der »Spry-Überprüfung Textfeld ohne Formatüberprüfung«

Diese Anzeigezustände werden den Besuchern Ihrer Website angezeigt, abhängig davon, was für das Textfeld angegeben wurde. Entscheiden Sie sich zum Beispiel

für die Zustandsvorschau Erforderlich, wird die Fehlermeldung ❹ ausgegeben, wenn in das Textfeld nichts eingetragen wurde.

In Listing 14.2 sehen Sie den HTML-Quelltext des Spry-Elements. Dort können Sie erkennen, dass die Textmeldung bei »Ereignis und kein Wert eingegeben« ganz einfach im HTML-Code verändert werden kann. Sie können natürlich auch in der Eigenschaftenpalette die Zustandsvorschau Erforderlich aktivieren und die Beschriftung direkt im Layout ändern.

```
<form id="form1" name="form1" method="post"
action="http://wohinauchimmer">
  <span id="sprytextfield1">
  <input type="text" name="text1" id="text1" />
  <span class="textfieldRequiredMsg">Es muss ein Wert
  angegeben werden.</span>
</span>
</form>
```

Listing 14.2 Überprüfung Textfeld ohne Formatüberprüfung

Zeitpunkt der Überprüfung (Ereignis)

Um die Überprüfung des Textfeldes zu starten, muss der Benutzer eine Interaktion auf der Website durchführen und ein »Ereignis« auslösen. Es werden hier drei Ereignisse unterschieden. Ein Ereignis ist, wie bereits beschrieben, eine Maus- oder Tastaturaktion.

Überprüfen bei	Ereignis
onBlur	Die Überprüfung findet statt, wenn der Benutzer in das Textfeld geklickt hat und es wieder verlässt bzw. in einen anderen Bereich der Website klickt.
onChange	Die Überprüfung findet statt, während der Benutzer ein Textfeld verändert.
onSubmit	Die Überprüfung findet statt, wenn der Benutzer den Absenden-Button angeklickt hat.

Tabelle 14.2 Ereignisse für die »Spry-Überprüfung Textfeld«

Die Varianten der Ereignisse werden in der Eigenschaftenpalette angezeigt. Sie können durchaus auch mehrere Ereignisse zur Überprüfung auswählen. Bei der Spry-Überprüfung Textfeld ohne Formatüberprüfung reicht jedoch onSubmit aus.

Es ist natürlich auch möglich, auf jegliche Überprüfung zu verzichten, indem Sie das Kontrollkästchen Erforderlich deaktivieren. Das bietet sich in der Kombination mit Spry-Überprüfung Textfeld ohne Formatüberprüfung beispielsweise

an, wenn innerhalb eines Formulars die grundsätzliche Funktion der Überprüfung bereits angelegt ist, aber zwischenzeitlich deaktiviert werden soll. Dies kann beispielsweise dazu dienen, die Website später zu erweitern.

Ein Hinweis an den Benutzer kann im Eingabefeld Tipp angelegt werden. Dieser Hinweis wird im Browser innerhalb des Textfeldes angezeigt. Sobald ein Benutzer in das Textfeld klickt, wird der Hinweis automatisch entfernt, um eine Eingabe zu ermöglichen.

Spry-Überprüfung Textfeld mit Formatüberprüfung

Mit der eben beschriebenen Spry-Überprüfung Textfeld ohne Formatüberprüfung können Sie letztlich nur überprüfen, ob überhaupt eine Eingabe erfolgte oder nicht. Wesentlich genauer geht es mit der zweiten Variante der Spry-Überprüfung Textfeld ❶. Jetzt überprüfen wir auch noch die eingegebenen Daten auf Gültigkeit. Daraus ergibt sich eine zusätzliche Zustandsvorschau.

Ereignis	Zustandsvorschau (Dreamweaver)	Nummer im Bild
keinerlei Ereignis	Anfänglich	❷
Ereignis und kein Wert eingegeben	Erforderlich	❸
Ereignis und falschen Wert eingegeben	Ungültiges Format	❹
Ereignis und Wert eingegeben	Gültig	❺

Tabelle 14.3 Zustandsprüfung mit Überprüfung eines bestimmten Formats

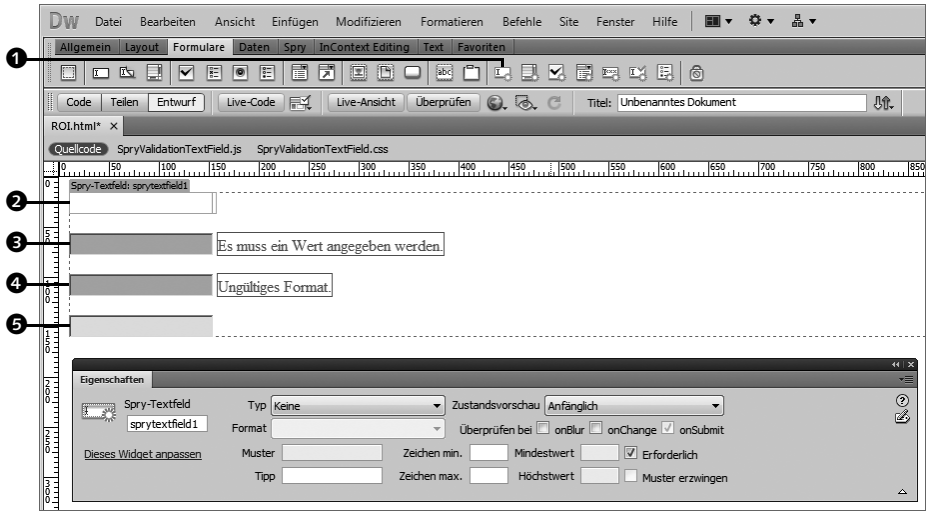


Abbildung 14.14 Vorschaumöglichkeit der »Spry-Überprüfung Textfeld mit Formatüberprüfung«

Im Quelltext ist eine weitere Zeile hinzugekommen. Die Bearbeitungsmöglichkeiten haben sich nicht verändert.

```
<form id="form1" name="form1" method="post"
action="http://wohinauchimmer">
  <span id="sprytextfield1">
  <input type="text" name="text1" id="text1" />
  <span class="textfieldRequiredMsg">Es muss ein Wert
  angegeben werden.</span>
  <span class="textfieldInvalidFormatMsg">Ungültiges Format.
  </span>
  </span>
</form>
```

Listing 14.3 Überprüfung Textfeld mit Formatüberprüfung

Es empfiehlt sich, als Ereignis und zum Überprüfen dieses Mal onChange und/oder onBlur zu aktivieren. Nur so kann eine schnelle Erkennung von falschen Eingaben ermöglicht werden.

Spannend wird es bei der Auswahl der verschiedenen Eingabeformate: Im Feld Typ ist zunächst eine grobe Auswahl des Eingabeformats erforderlich. Bei den meisten Formattypen wie beispielsweise Datum kann im Feld Format eine genauere Einschränkung erfolgen. In Tabelle 14.4 haben wir Ihnen die Einstellungsmöglichkeiten aufgelistet.

Wenn sich das benötigte Format nicht darunter befindet, ist es ganz einfach, bei Auswahl von Benutzerdefiniert unter Typ ein eigenes Muster zu erstellen. Dieses Muster kann nach folgendem Schema angelegt werden:

xx-xx-xx

Der Benutzer muss dann später seine Eingaben nach folgendem Schema vornehmen: »Zeichen Zeichen – Zeichen Zeichen – Zeichen Zeichen«. Satzzeichen werden als solche erkannt und müssen dann innerhalb des späteren Eingabefensters eingetragen werden.

Typ	Format
Ganzzahl	ganze Zahlen ohne Nachkommastellen
E-Mail-Adresse	xxx@xxx.xx
Datum	Auswahl der verschiedenen Schreibweisen im Feld Format

Tabelle 14.4 Datenformate im Textfeld mit Formatüberprüfung

Typ	Format
Uhrzeit	Auswahl der verschiedenen Schreibweisen im Feld Format
Kreditkarte	Auswahl der verschiedenen Schreibweisen im Feld Format
Postleitzahl	Auswahl der verschiedenen Schreibweisen im Feld Format
Telefonnummern	Auswahl der verschiedenen Schreibweisen im Feld Format
Sozialversicherungs-nummer	Auswahl der verschiedenen Schreibweisen im Feld Format
Währung	Auswahl, ob Komma oder Punkt als Tausender-Trennzeichen
Reelle Zahl/Exponentielle Schreibweise	zum Beispiel: 3,45
IP-Adresse	Auswahl der verschiedenen Schreibweisen im Feld Format
URL	http://www.galileodesign.de
Benutzerdefiniert	Angabe im Schema xx-xx-xx

Tabelle 14.4 Datenformate im Textfeld mit Formatüberprüfung (Forts.)

14.4.3 Spry-Überprüfung Textbereich

Mit dem Feld Spry-Überprüfung Textbereich ❶ (siehe Abbildung 14.15) überprüfen Sie größere Textmengen während der Eingabe. Die maximale Anzahl der Zeichen ist überprüfbar und kann bei Bedarf für den Benutzer neben dem Textbereich eingeblendet werden. Weitere Einstellungen wie zum Beispiel eine Formatprüfung gibt es hier nicht.

Wenn Sie einen Zähler mit einer Mindest- ❷ und einer Maximalzahl ❸ an Zeichen anlegen, können in der Zustandsvorschau weitere Anzeigezustände aktiviert werden.

Im folgenden Listing sehen Sie den Quelltext mit aktiven Parametern für eine Mindestzeichenzahl sowie einer maximalen Zeichenzahl.

```
<form id="form1" name="form1" method="post" action="">
<span id="sprytextarea1">
<textarea name="textarea1" id="textarea1" cols="45" rows="5">
</textarea>
```

```
<span class="textareaRequiredMsg">Es muss ein Wert angegeben werden.
</span>
<span class="textareaMinCharsMsg">Die mindestens erforderliche
Zeichenanzahl wurde unterschritten.</span>
<span class="textareaMaxCharsMsg">Die maximal zulässige
Zeichenanzahl wurde überschritten.</span>
</span>
</form>
```

Listing 14.4 Überprüfung Textbereich, Spry-Überprüfung Textbereich

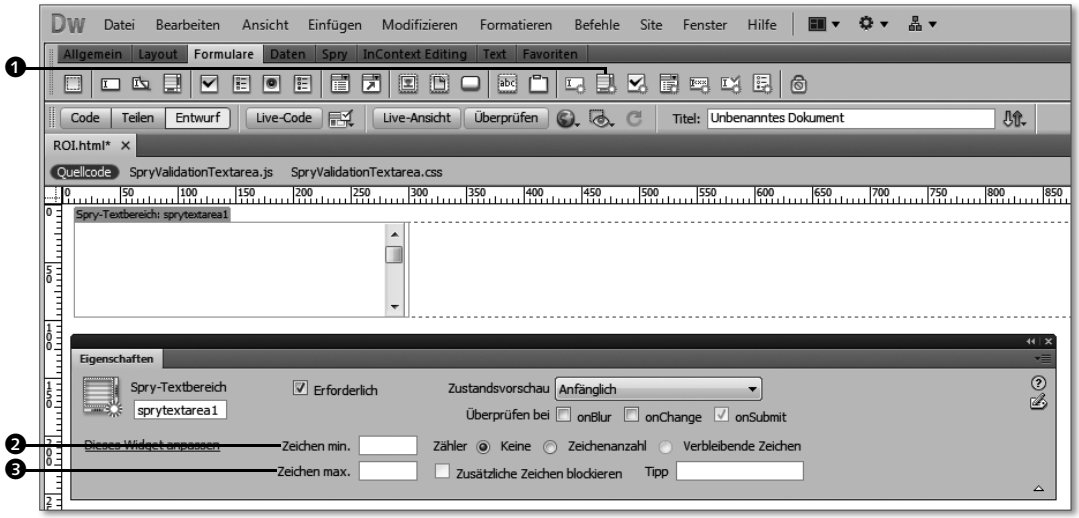


Abbildung 14.15 Einstellungen für »Spry-Überprüfung Textbereich«

14.4.4 Spry-Überprüfung Kontrollkästchen

Ebenfalls sehr überschaubar sind die Parameter für die SPRY-ÜBERPRÜFUNG KONTROLLKÄSTCHEN ❹ (siehe Abbildung 14.16). Hier besteht nur die Möglichkeit, ein Kontrollkästchen als erforderlich oder als nicht erforderlich zu markieren.

Gruppen von Kontrollkästchen haben wir weiter oben in Abschnitt 14.3, »Formularelemente«, beschrieben. Um mit einer Gruppe zu arbeiten, müssen Sie lediglich AUSWAHLBEREICH (MEHRERE) für alle Kontrollkästchen einer Gruppe aktivieren.

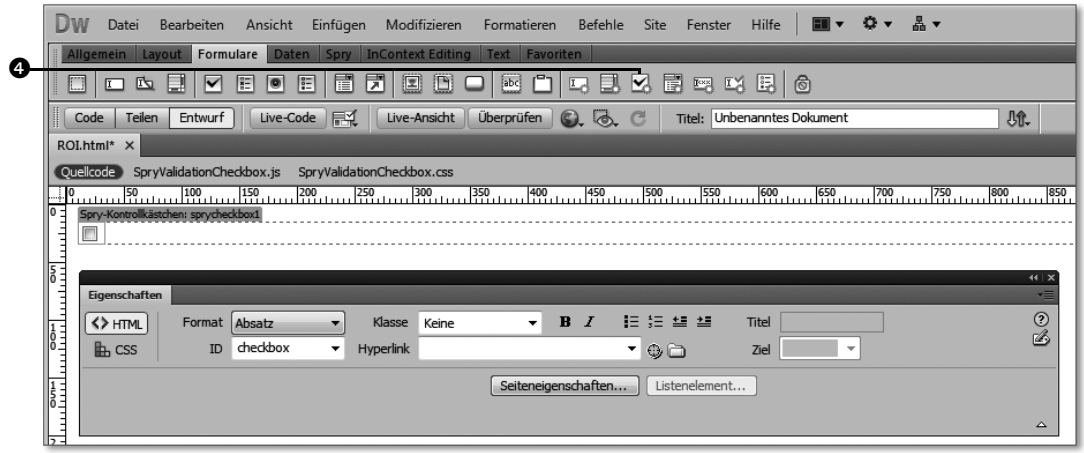


Abbildung 14.16 Einstellungen für »Spry-Überprüfung Kontrollkästchen«

14.4.5 Spry-Überprüfung Auswahl

Die SPRY-ÜBERPRÜFUNG AUSWAHL ❺ legt fest, ob Werte in einer Auswahlliste angewählt werden müssen oder nicht. Ungültige Werte können Sie abfangen.

Bei aktivem Kontrollkästchen UNGÜLTIGER WERT ❻ wird der als ungültig anzusehende Wert rechts daneben eingetragen. Zum Beispiel kann in einer Landesauswahlliste der Wert für ein bestimmtes Land ungültig sein.

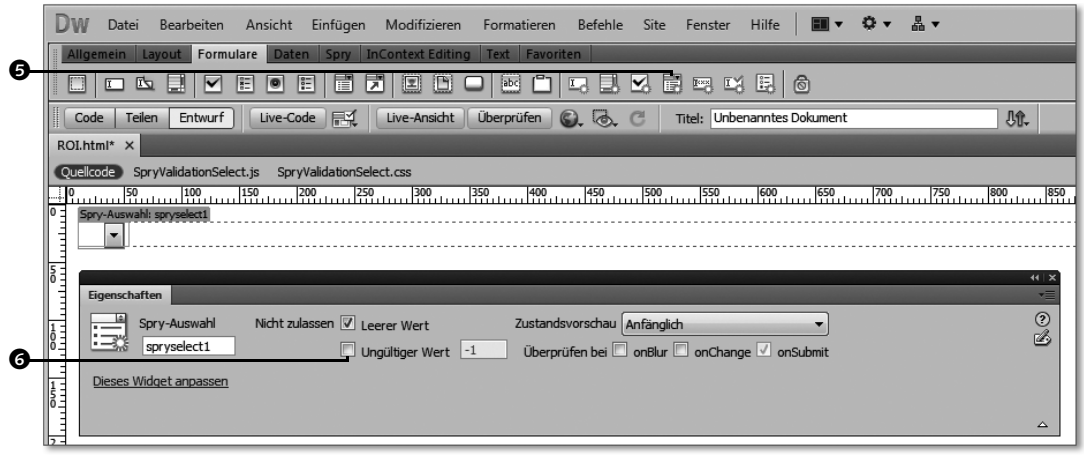


Abbildung 14.17 Einstellungen für »Spry-Überprüfung Auswahl«

14.4.6 Spry-Überprüfung Kennwort

Neu seit CS4 oder besser gesagt seit Spry 1.6.1 ist die Funktion SPRY-ÜBERPRÜFUNG KENNWORT ❶. Diese erzeugt ein Textfeld, das die Benutzereingabe mit Punkten oder Sternchen versteckt, genau wie das normale Kennwortfeld.

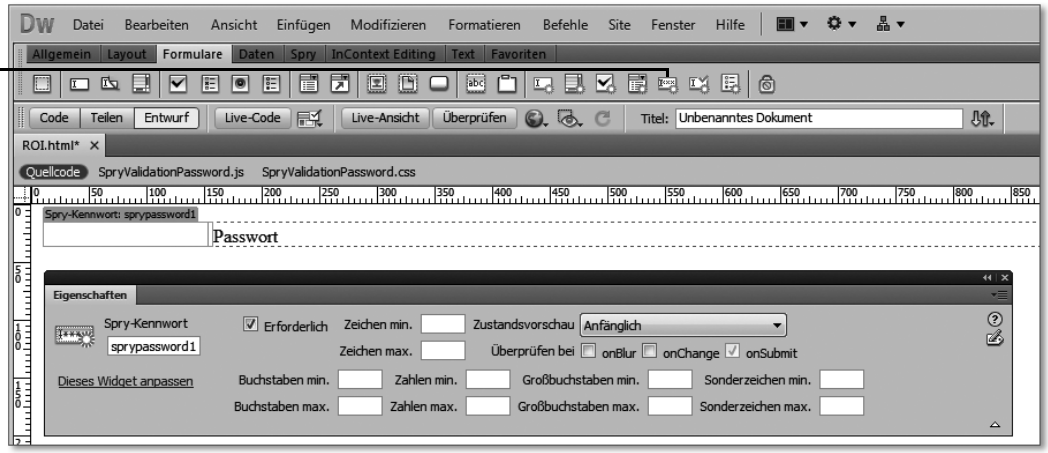


Abbildung 14.18 Einstellungen für »Spry-Überprüfung Kennwort«

Kontrolliert werden kann hier aber das Format des eingegebenen Kennworts schon während der Eingabe. Sie sollten allerdings dem Benutzer in einem erklärenden Text zumindest bei der Registrierung mitteilen, welchen Anforderungen das Kennwort genügen muss.

14.4.7 Spry-Überprüfung Bestätigung

Passend zur Kennwortüberprüfung gibt es im aktuellen Spry-Framework die Überprüfung BESTÄTIGUNG ❷ (siehe Abbildung 14.19). Sinnvoll ist diese zum Beispiel bei Kennwortwiederholungen.

In der Eigenschaftenpalette muss hierbei das Eingabefeld angegeben werden, dessen Wert überprüft werden soll. Daher sollten Sie das Kennwortfeld schon vorher angelegt haben.

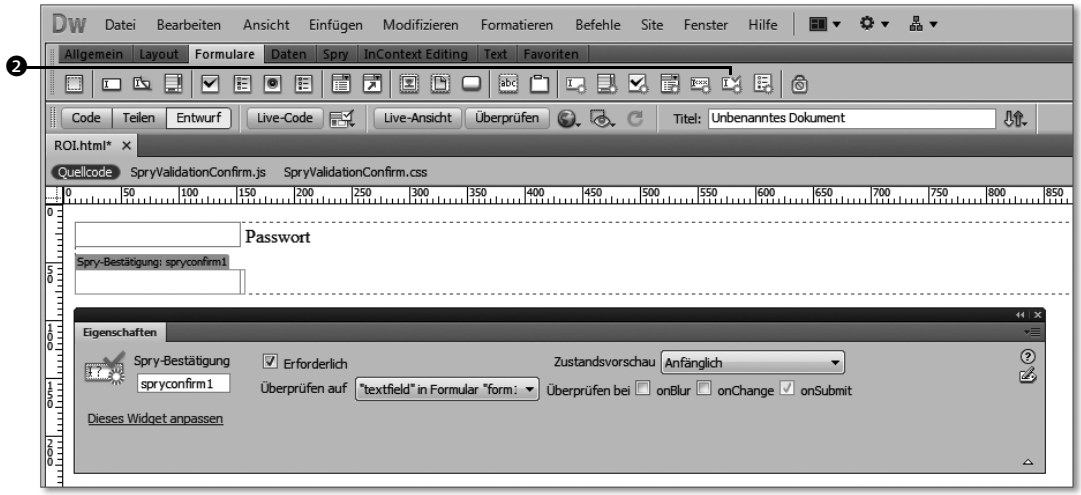


Abbildung 14.19 Einstellungen für »Spry-Überprüfung Bestätigung«

14.5 Formulare gestalten

14.5.1 Formularelemente positionieren

Formulare sind sehr störrisch. Ohne eine ausgefeilte Tabellenstruktur ist es kaum möglich, die einzelnen Formularelemente gut zu positionieren. Mit AP-Elementen können Formulare deutlich einfacher positioniert werden, jedoch ist der Aufwand, mehrere Formularfelder exakt auszurichten, ebenfalls recht hoch.

Vermeiden Sie bei der Tabellenstruktur verbundene Zellen. Diese ergeben in den meisten Fällen im Layout weitere Unwägbarkeiten. Teilen Sie die Tabellen besser immer in mehrere einzelne Zellen auf. Es ist sinnvoll, zwischen den Formularfeldern und der Beschriftung eine weitere Tabellenspalte einzufügen. Der Text hängt ansonsten direkt am Formularelement. Eine Formatierung von Formularen ohne CSS ist nur bedingt möglich. Sie können zwar eine Zeichenanzahl für Textfelder eingeben, auf verschiedenen Plattformen und Browsern werden die Breiten jedoch unterschiedlich interpretiert (siehe Abbildung 14.20). Wenn Sie dennoch ohne CSS arbeiten möchten, sollten Sie darauf achten, in den Tabellenzellen genug Freiraum für unterschiedliche Breiten der Formularfelder zu schaffen, damit die Tabellen nicht gestreckt werden.

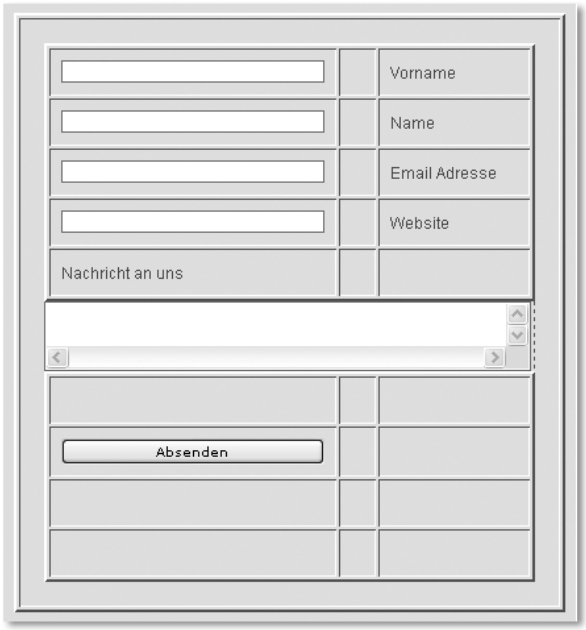


Abbildung 14.20 Ein Formularlayout mit Tabellen

14.5.2 Formularelemente mit CSS gestalten

Mit den Kontext-Selektoren `INPUT`, `SELECT` und `TEXTAREA` können Formularfelder an das Layout einer Website exakt angepasst werden. Mit Listing 14.5 erhalten Sie somit ein schönes, ausgewogenes Layout der Formularelemente:

```
INPUT {
   font-family : Verdana, Geneva, Arial, Helvetica,
   sans-serif;
   font-size : 10px;
   border : 1px solid #87A9B3;
   width: 200px;
}
SELECT {
   font-family : Verdana, Geneva, Arial, Helvetica,
   sans-serif;
   font-size : 10px;
   border : 1px solid #87A9B3;
   width: 200px;
TEXTAREA {
   font-family : Verdana, Geneva, Arial, Helvetica,
   sans-serif;
   font-size : 10px;
```

```
   border : 1px solid #87A9B3;
   width: 400px;
}
```

Listing 14.5 Beispiel-CSS-Code für Formularfelder

Formularfelder sehen wesentlich besser aus, wenn sie nicht wie üblich als eintönige graue Kästen erscheinen. Etwas Vorsicht ist jedoch geboten, da Benutzern Eingabefelder als eben solche bekannt sind. Anders gestaltete Formularfelder werden daher manchmal nicht auf Anhieb erkannt.

Der Selektor `INPUT` ist übrigens auch für das Aussehen der Schaltflächen verantwortlich. So erscheint eine Schaltfläche im ungünstigsten Fall wie ein Formularfeld. Eine Lösung für dieses Dilemma ist, für die Schaltfläche eine eigene Klasse mit den gestalterischen Abweichungen zu erstellen und jeder Schaltfläche diese Klasse zuzuweisen. Dasselbe trifft auf Checkboxen und Radiobuttons zu.



Abbildung 14.21 Formular im Internet Explorer mit CSS

14.6 Ein Kontaktformular für die Buchwebsite



In unserer Buchwebsite soll ein Kontaktformular die Daten aus Tabelle 14.5 an ein CGI auf dem Webserver übertragen. Das CGI existiert nur fiktiv. Wir empfehlen Ihnen, auch wenn Sie ein CSS-Layout gewählt haben, die Formularelemente dennoch mit Tabellen aufzubauen. Wenn Sie das nicht möchten, müssen Sie für jedes Formularelement ein eigenes AP-Element anlegen und positionieren. In dieses AP-Element werden dann die Formularelemente aufgenommen.

Bezeichnung	Variablenname	Benötigt?
Vorname	`vorname`	ja
Name	`name`	ja

Tabelle 14.5 Angaben für die Überprüfung des Formulars

Bezeichnung	Variablenname	Benötigt?
E-Mail-Adresse	`email`	ja
Website	`url`	nein
Nachricht	`nachricht`	ja

Tabelle 14.5 Angaben für die Überprüfung des Formulars (Forts.)

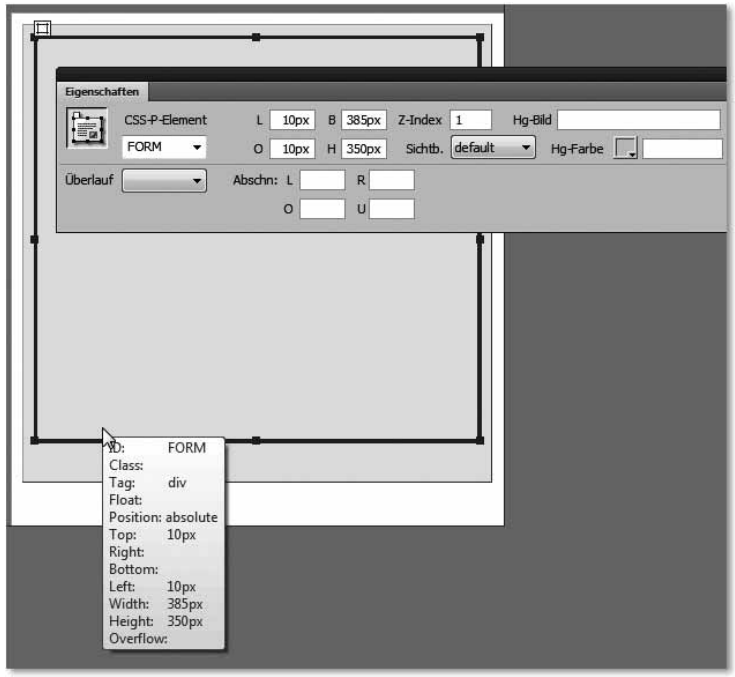
Schritt-für-Schritt: Ein Kontaktformular erstellen

1 *HTML-Dokument öffnen*

Öffnen Sie das HTML-Dokument *7_0.htm*. Dieses Dokument soll unser Kontaktformular werden. Wir erstellen dieses Mal das Layout des Formulars ausnahmsweise nicht vollständig mit CSS. Die CSS-Styles für die Formularelemente haben wir bereits in Kapitel 11, »CSS in Dreamweaver«, angelegt, so dass wir jetzt nur noch eine Tabelle und die Formularelemente benötigen.

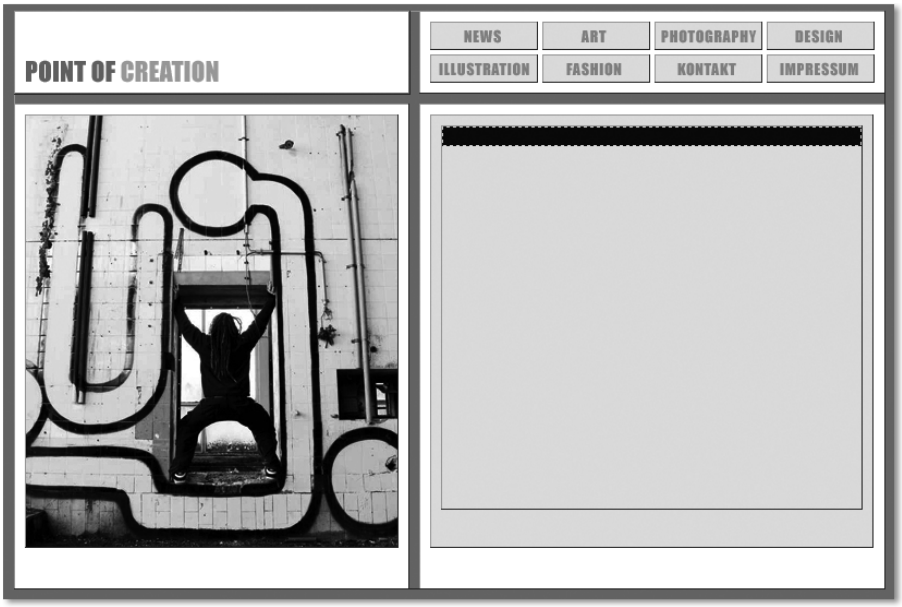
2 *Tabelle entfernen*

Entfernen Sie in diesem Dokument zunächst den Inhalt im rechten Feld und die beiden AP-Elemente für die Datensatznavigationen, und erstellen Sie ein neues AP-Element mit den Abmessungen wie in der folgenden Abbildung.



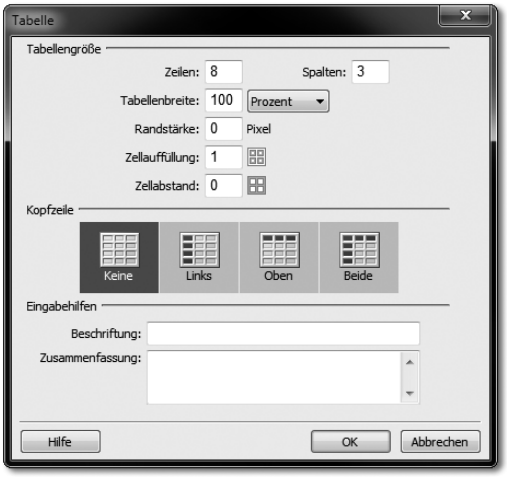
3 *Formular einfügen*

Fügen Sie in das nun leere AP-Element ein Formular ein.



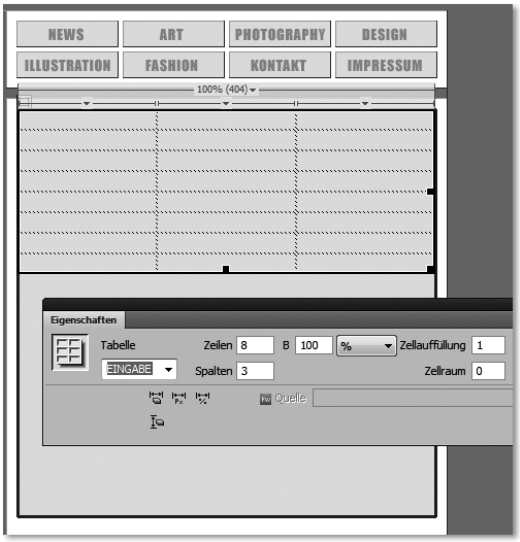
4 *Tabelle einfügen*

Achten Sie darauf, die in der Abbildung spezifizierte Tabelle *in* das Formular einzufügen. Klicken Sie dazu genau in den roten Rahmen des Formulars. In den Tabellenparametern wird eine ZELLAUFFÜLLUNG von 1 Pixel angegeben, damit die Formularfelder nicht direkt aneinanderliegen.



5 *Tabellen-ID zuweisen*

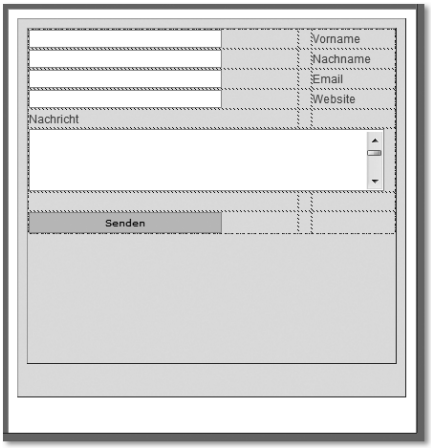
Weisen Sie der eingefügten Tabelle gleich in der Eigenschaftenpalette die ID EINGABE zu.



Diese erscheint nur in der Eigenschaftenpalette, weil wir beim Anlegen der CSS-Definition einen ID-Selektor mit dieser Bezeichnung erstellt haben (siehe Abschnitt 11.3.3 »ID-Selektoren«).

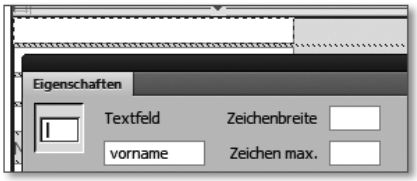
6 *Formularfelder einfügen*

Jetzt werden in die entsprechenden Tabellenzellen die Formularfelder wie abgebildet eingefügt.



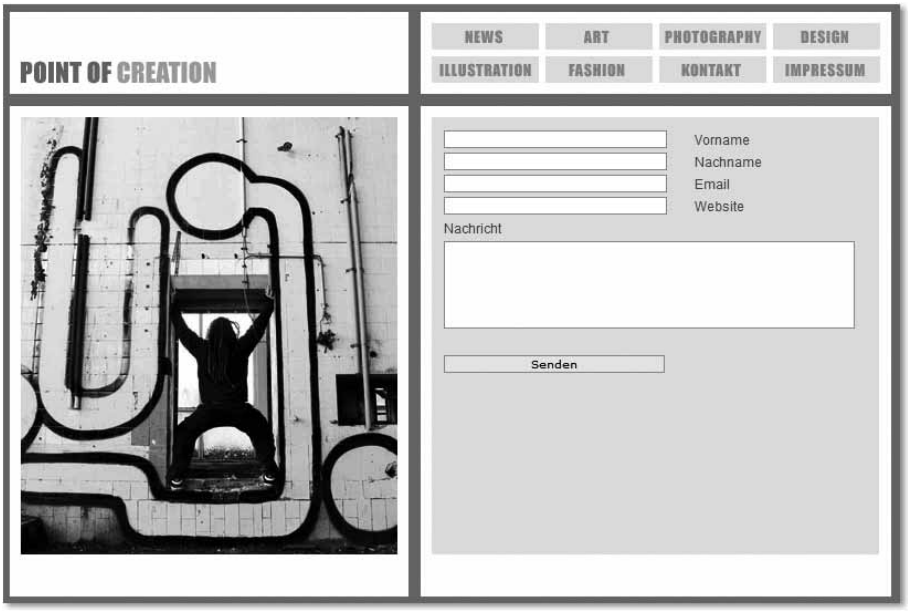
7 *Variablennamen vergeben*

Wählen Sie anschließend die einzelnen Felder aus, und benennen Sie diese mit den in der obigen Tabelle angegebenen Variablennamen. Sie können auch gleich die Beschriftungen einfügen.



8 *Zellen verbinden*

In der Tabellenreihe, in der sich das Textfeld für die »Nachricht« befindet, müssen die Tabellenzellen ausnahmsweise verbunden werden. In der folgenden Abbildung sehen Sie dann das fertige Formular.



Sind Sie an dieser Stelle angelangt, ist der Aufbau des statischen Teils der Website vollständig abgeschlossen. ■

Mit Dreamweaver sind Interaktionen mit Flash und die Einbindung von Flash-Dateien denkbar einfach. In diesem Kapitel erfahren Sie, wie es geht.

15 Flash und YouTube integrieren

Flash und Dreamweaver sind ein gutes Team. Aber wen wundert es, entstammen doch beide Programme derselben Softwareschmiede Macromedia. YouTube setzt die Flash-Technologie ein, so dass wir die Möglichkeiten der Einbindung von YouTube-Videos ebenfalls in diesem Kapitel besprechen werden.

15.1 Flash einsetzen oder nicht?

Flash ist aus dem Internet nicht mehr wegzudenken. Allerdings hat sich nach unserer Beobachtung das Einsatzfeld etwas verändert. Vor drei Jahren musste jede Seite fetzig, bunt und laut sein. Mittlerweile sind Flash-Websites etwas dezenter und informativer geworden. Vorurteile gegen Flash, die Seiten seien zu groß, nervig und enthielten kaum Informationen, bestätigen sich nicht mehr allzu oft.

Adobes Aussage, dass nahezu jeder Browser über das Flash-Plugin verfügt, trifft im Großen und Ganzen zu. Selbst bei Behörden und großen Unternehmen, die eine mächtige Firewall haben, gehen diese Inhalte durch und werden korrekt angezeigt. Kein anderes Browser-Plugin hat – nicht zuletzt durch unzählige Werbebanner – eine derartige Verbreitung gefunden. Einige der besten uns bekannten Websites sind mit Flash erstellt worden, und wir möchten auf keinen Fall auf Flash in unserer Arbeit verzichten.

Man muss allerdings sehr genau unterscheiden, für welche Zielgruppen eine Website erstellt werden soll. Wenn Sie ein Produkt verkaufen wollen, das hauptsächlich optisch kommuniziert (etwa Mode, Events oder Games), spricht zunächst einmal nichts gegen Flash. Flash ist zudem hervorragend dafür geeignet, Prozesse zu visualisieren und Kunden Konfigurationsmöglichkeiten für Seiten und Grafiken zu bieten, die mit HTML so nicht möglich wären.

Flash sollte jedoch nicht eingesetzt werden, wenn bei den Benutzern niedrige Bandbreiten, alte Browser und langsame Rechner zu erwarten sind. Auch iPhone und iPad unterstützen Flash nicht. Zudem sind Flash-Seiten nur mit erheblichem Aufwand suchmaschinentauglich zu gestalten und barrierefrei sind sie ebenfalls nicht.

15.2 Flashfilme einfügen

15.2.1 Filme einbinden

Als *Flash-Filme* bezeichnet man die mit Flash erstellten Dateien mit der Endung *.swf*. Diese enthalten nicht etwa feste Programmierungen wie in HTML, sondern bestehen aus sogenannten *Schlüsselbildern*, die in einer bestimmten zeitlichen Abfolge gezeigt werden und deswegen eher mit einem Film als mit einem Programm vergleichbar sind.

Sofern es sich um Videos handelt, die innerhalb von Flash abgespielt werden, wird dies als *Flash-Video* (Endung *.flv* oder *.f4v*) bezeichnet. Dabei laufen Videos in einer speziellen Codierung innerhalb eines Flash-Films, der die Steuerelemente bereitstellt.

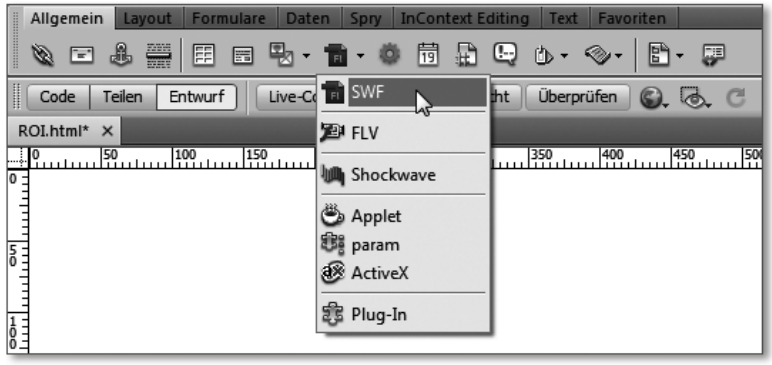


Abbildung 15.1 Flash (SWF) einbinden

Einfügen können Sie bestehende Flash-Filme über die Einfügeleiste mit einem Klick auf Allgemein • Medien • Flash. Haben Sie bereits Flash-Filme eingefügt, erscheint das Icon Flash in der Iconleiste.

Zunächst wird der Flash-Film einfach als grauer Kasten in der Entwurfsansicht angezeigt. Wenn Sie oben auf das kleine Auge klicken ❶, sehen Sie die Inhalte, die ein Benutzer sieht, der nicht über das Flash-Plugin verfügt.

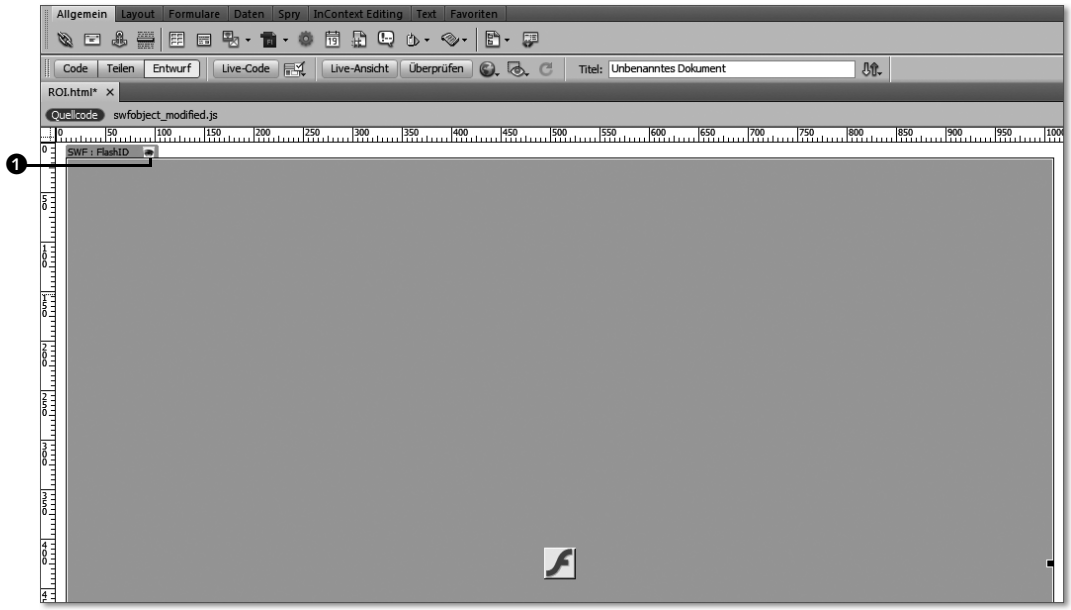


Abbildung 15.2 Flash-Film in der Entwurfsansicht



Abbildung 15.3 Anzeige, wenn ein neuerer Flash Player erforderlich ist

Sie können Flash-Filme direkt in Dreamweaver abspielen lassen. Klicken Sie dazu in der Eigenschaftenpalette auf den Button mit dem kleinen grünen Pfeil und der Beschriftung ABSPIELEN ❷.



Abbildung 15.4 Eigenschaften für Flash-Filme

Der Flash-Film wird nun direkt im Entwurfsfenster von Dreamweaver abgespielt. So können Sie die umgebenden Elemente perfekt an den Film anpassen.

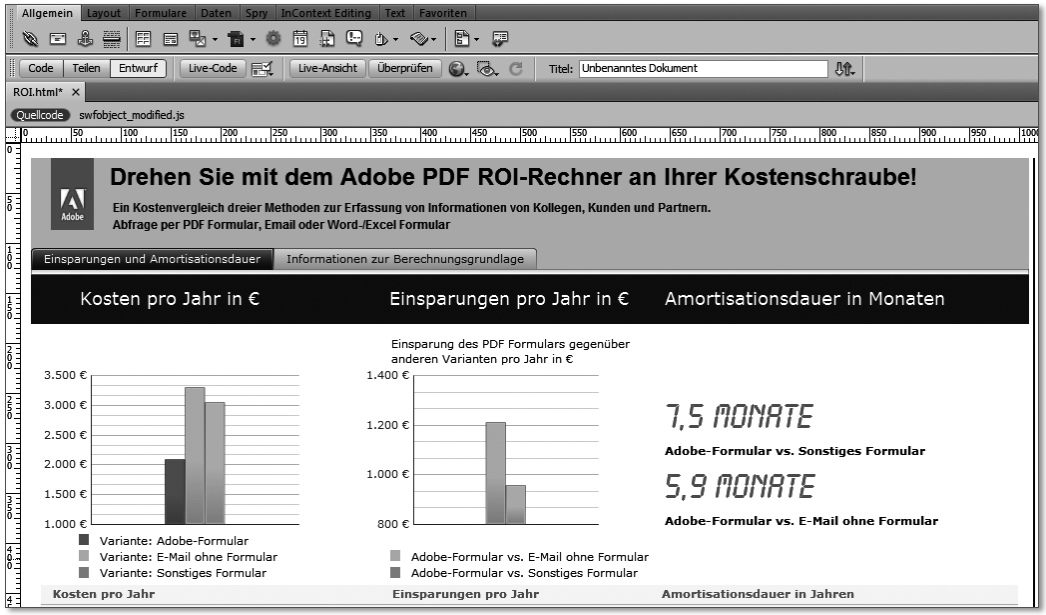


Abbildung 15.5 Laufender Flash-Film im Entwurf

15.2.2 Eigenschaften einstellen

Einen eingebetteten Flash-Film können Sie wie alle anderen Elemente in Dreamweaver in der Eigenschaftenpalette bearbeiten.

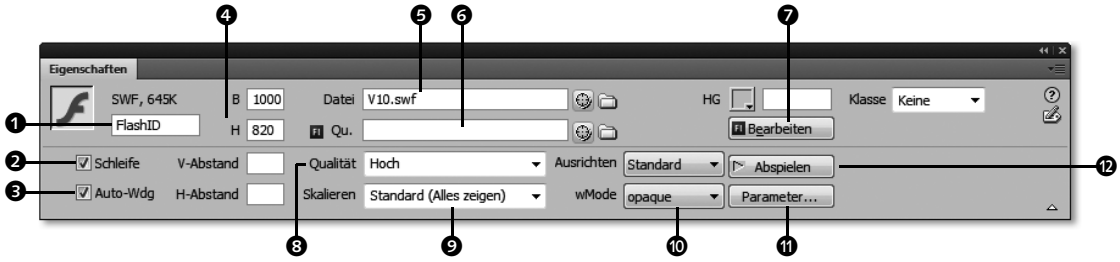


Abbildung 15.6 Die Eigenschaftenpalette für Flash

Im Textfeld ❶ geben Sie dem Film einen Namen. Dieser ist erforderlich, wenn Sie den Film z. B. über Aktionen steuern möchten. Die Felder B (Breite) und H (Höhe) ❹ legen die Größe des Filmobjekts in der Seite fest. DATEI ❺ verlinkt wie bei Bildern auf die Flash-Datei und zeigt im Beispiel den relativen Pfad zum Flash-Film. Im Feld QU. (Quelle) ❻ wird die bearbeitbare Ursprungsdatei des Films angezeigt. Wir finden es sehr praktisch, dass man diese direkt aus Dreamweaver heraus durch Klicken auf BEARBEITEN ❼ editieren kann.

Mit Aktivieren und Deaktivieren der Checkboxen SCHLEIFE ❷ und AUTO-WDG ❸ legen Sie fest, wie der Film abgespielt werden soll. Die Abspielqualität eines Films können Sie unter QUALITÄT ❽ einstellen. Manchmal passen Filme nicht in exakte Tabellen, obwohl die Pixelwerte genau eingestellt sind. Versuchen Sie, ihn passend zu machen, indem Sie die Einstellungen unter SKALIEREN ❾ verändern. Sie können den Film auf diese Weise sehr genau einpassen. Der Button ABSPIELEN ⓬ bewirkt, dass der Film im Entwurfsmodus abgespielt wird. Mit demselben Button können Sie den Film auch wieder anhalten.

Seit CS4 gibt es den nützlichen Schalter WMODE ❿. Dieses Attribut musste früher im Quelltext von Hand gesetzt werden. OPAQUE lässt den Flash-Film mit deckendem Hintergrund laufen. Es gibt aber auch Flash-Filme, die nur halbtransparent laufen oder zum Beispiel nur in einem runden Bereich erscheinen. Setzen Sie dann WMODE auf TRANSPARENT.

Parameter einstellen

Über die Schaltfläche PARAMETER ⓫ in der Eigenschaftenpalette können Sie die Abspieleigenschaften der Flash-Filme beeinflussen.



Abbildung 15.7 Parameter für Flash

Automatische Aktivierung im Browser

Während des Einfügens erscheint bei Dreamweaver eine Meldung, die Sie auffordert, eine JavaScript-Datei zu kopieren. In neueren Versionen des Internet Explorers werden aktive Inhalte, zu denen auch Flash zählt, aus Sicherheitsgründen nicht mehr automatisch aktiviert. Man kann diese dennoch mit einem kleinen JavaScript ohne Zutun des Benutzers freischalten. Der Sicherheitsgewinn mag sich nicht wirklich erschließen, Fakt ist jedoch, dass diese JavaScripts notwendig sind.

Wenn Sie ältere HTML-Dateien mit Flash in Dreamweaver öffnen, erscheint die gleiche Meldung. So können Sie Websites bezüglich Flash auf einfache Weise auf den aktuellen Stand bringen.

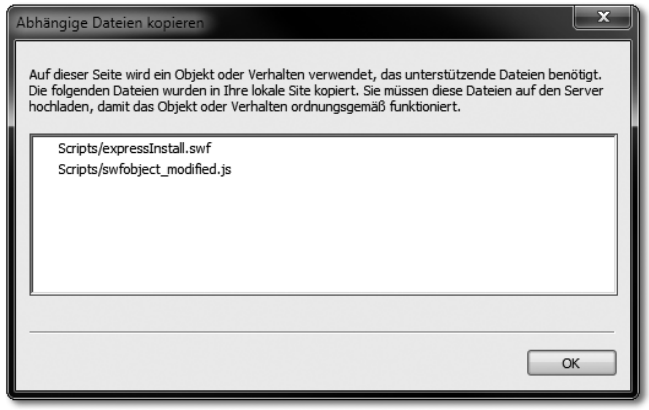


Abbildung 15.8 JavaScript hinzufügen

Einbettung im Quellcode

Wenn Sie in die Codeansicht gehen, sehen Sie, dass der Flash-Film im HTML-Quelltext als Objekt in das Dokument eingebettet und auch gleich das soeben erwähnte JavaScript angefügt wird:

```
<body>
<object id="FlashID" classid="clsid:D27CDB6E-AE6D-11cf-96B8-
444553540000" width="1000" height="820">
  <param name="movie" value="V10.swf" />
  <param name="quality" value="high" />
  <param name="wmode" value="opaque" />
  <param name="swfversion" value="9.0.45.0" />
  <!-- Dieses param-Tag fordert Benutzer von Flash Player 6.0 r65
  und höher auf, die aktuelle Version von Flash Player
  herunterzuladen. Wenn Sie nicht wünschen, dass die Benutzer
  diese Aufforderung sehen, löschen Sie dieses Tag. -->
  <param name="expressinstall" value="Scripts/expressInstall.swf" />
  <!-- Das nächste Objekt-Tag ist für Nicht-IE-Browser vorgesehen.
  Blenden Sie es daher mit IECC in IE aus. -->
  <!--[if !IE]>-->
  <object type="application/x-shockwave-flash" data="V10.swf"
  width="1000" height="820">
    <!--<![endif]-->
    <param name="quality" value="high" />
    <param name="wmode" value="opaque" />
    <param name="swfversion" value="9.0.45.0" />
    <param name="expressinstall" value="Scripts/expressInstall.swf" />
    <!-- Im Browser wird für Benutzer von Flash Player 6.0 und
    älteren Versionen der folgende alternative Inhalt angezeigt. -->
```

```
    <div>
      <h4>Für den Inhalt dieser Seite ist eine neuere Version von
      Adobe Flash Player erforderlich.</h4>
      <p><a href="http://www.adobe.com/go/getflashplayer">
      <img src="http://www.adobe.com/images/shared/
      download_buttons/get_flash_player.gif" alt="Adobe Flash Player
      herunterladen" width="112" height="33" /></a></p>
    </div>
    <!--[if !IE]>-->
  </object>
  <!--<![endif]-->
</object>
<script type="text/javascript">
swfobject.registerObject("FlashID");
</script>
</body>
```

Listing 15.1 Im Code eingebetteter Flash-Film

In den Parametern erscheint eine Klassen-ID, die das Objekt eindeutig als Flash-Film identifiziert. Zusätzlich ist die URL für den Download des Flash-Plugins angegeben, falls dieses nicht installiert ist.

15.3 Flash-Sound einbinden

Eine hervorragende Möglichkeit bietet das Einbinden von Flash-Sound in eine Website. Dazu brauchen wir auf der Site keinen sichtbaren Flash-Film anzuzeigen, mit einem kleinen Trick können wir ihn auch unsichtbar in HTML-Sites einbinden.

Erstellen Sie zuerst einen Flash-Film mit dem Sound, den Sie auf Ihrer Site abspielen wollen. Reduzieren Sie die Größe des Films auf 2 × 2 Pixel, und exportieren Sie ihn ins SWF-Format. Anschließend binden Sie ihn an beliebiger Stelle in Ihre Website ein und versehen ihn über den Parameter `wmode` mit einem transparenten Hintergrund. Wie das geht, erfahren Sie im Anschluss. Wenn Sie möchten, können Sie über die ebenfalls im nächsten Abschnitt beschriebenen Aktionen den Sound auch an- und ausschaltbar machen.

Wägen Sie den Einsatz von Musik auf einer Website aber bitte sehr gut ab: Nicht nur, dass diese oft sehr störend oder für ältere Menschen möglicherweise verwirrend ist, noch mehr Durcheinander gibt es, wenn Folgeseiten in einem neuen Browser-Tab geladen und plötzlich zwei Sounds abgespielt werden. In Büros sind Rechner zudem sehr oft ohne Lautsprecher ausgestattet. Sollte also die Musik

oder Sprache wichtig für das Verständnis der Site sein, kann sie nicht von jedem genutzt werden. Und nicht zuletzt sind die für Sounds zu transferierenden Datenmengen auch im Zeitalter von MP3 und Konsorten immer noch sehr groß. Besucher, die an schmalen ISDN-Leitungen oder analogen Modems hängen, ärgern sich meist mehr über die lange Ladezeit der Seite, als dass sie den Sound genießen.

15.4 Flash in Dreamweaver öffnen

Unter der Voraussetzung, dass Sie Flash installiert haben, können Sie Flash-Dateien direkt in Dreamweaver bearbeiten. Flash-Filme bestehen aus zwei Dateien,

- der Flash-Authoring-Datei mit der Endung *.fla* und
- der exportierten SWF-Datei.

Beide Dateien müssen sich nicht im selben Ordner befinden, jedoch erkennt Dreamweaver den Ursprung der SWF-Datei nicht mehr, wenn Sie die FLA-Datei nach dem Veröffentlichen verschoben haben. In diesem Fall werden Sie aufgefordert, die FLA-Datei zu benennen.

Dreamweaver startet Flash nach dem Klick auf Bearbeiten in der Eigenschaftenpalette, und Sie können dort den Film verändern. Klicken Sie anschließend auf Fertig, und der Film wird mit den alten Export-Einstellungen neu exportiert.

[!] **Verknüpfung zu Flash**

Genau genommen ist dies keine wirkliche Bearbeitung in Dreamweaver. Es wird einfach der externe Flash-Editor gestartet. Die Verknüpfung ist jedoch recht nützlich.

15.5 FLV-Filme einbinden

Neben den eigentlichen Flash-(*swf*-)Filmen ist auch das FLV-Dateiformat sehr verbreitet. Im FLV-Format sind qualitativ sehr hochwertige Filme bis hin zu HDV ohne lange Ladezeiten möglich. FLV wird verwendet, wenn es sich um Videos handelt aus einem Format wie zum Beispiel DIVX oder AVI.

Eingefügt werden FLV-Filme wie Flash-Filme über das Menü Medien.

Neben der URL zum Film und der Technik des Videotyps (meist Progressiver Video-Download) muss hier ein Skin für den Player ausgewählt werden. Das ist ein kleiner Flash-Film, der das eigentliche FLV-Video umgibt – ähnlich wie man das auch vom Windows Media Player kennt, nur viel schlanker und anpassbar.

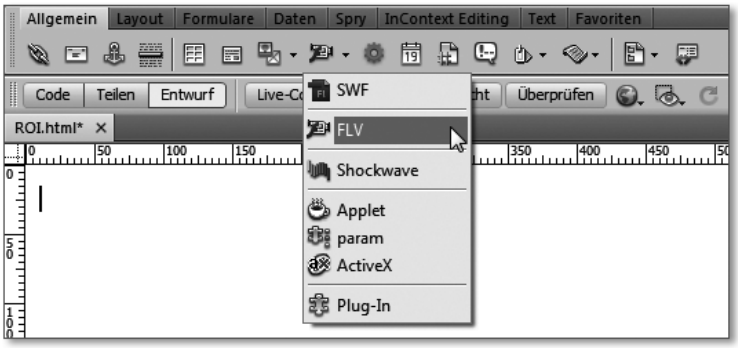


Abbildung 15.9 FLV einfügen



Abbildung 15.10 FLV-Einstellungen und Skin-Auswahl

Die Größe des Players kann man einfach bestimmen, indem man auf Grösse erkennen klickt. Dreamweaver ermittelt die Abmessungen automatisch aus der FLV-Datei.

Die Einstellmöglichkeiten im Entwurf sind recht begrenzt. Man kann den Skin hier noch einmal ändern und die Größe des Films modifizieren.

Im Browser erscheint der FLV-Film wie eine Flash-Datei. Der Player wird eingeblendet, wenn man mit der Maus über den Film zieht, und verschwindet wieder, wenn die Maus herausgezogen wird.

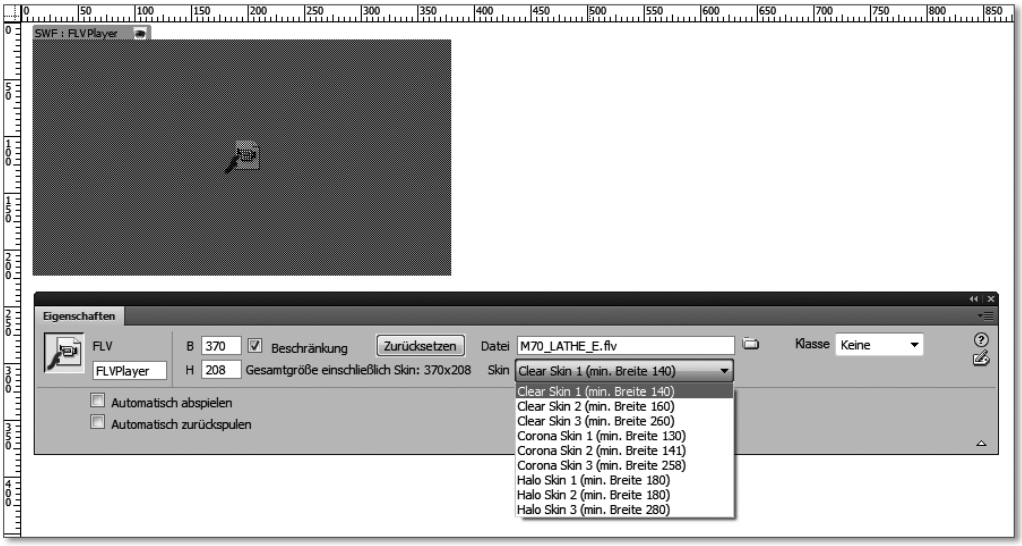


Abbildung 15.11 FLV in der Entwurfsansicht



Abbildung 15.12 FLV während der Ausführung

15.6 YouTube-Videos einbinden

Immer häufiger werden Videos nicht mehr nur auf der eigenen Website gezeigt, sondern sollen über YouTube oder einen ähnlichen Dienst einem breiten Publikum vorgestellt werden. Der Promotion-Effekt eines guten Videos bei einem der Videoportale kann enorm sein. Natürlich können auch Videos von z. B. YouTube in eine Website eingebunden werden.

[!] **Rechte beachten**

Bitte überlegen Sie zuvor, ob Sie über die entsprechenden Rechte verfügen, um Videos Dritter zu verwenden. Im Zweifelsfall kann ein kurzer Anruf oder eine E-Mail an den Rechteinhaber viel Ärger und Geld ersparen.



Abbildung 15.13 Galileo Press auf YouTube

Viele der Videodienste bieten auf ihrer Website eine Schaltfläche mit der Beschriftung EINBETTEN, CODE GENERIEREN oder Ähnlichem.

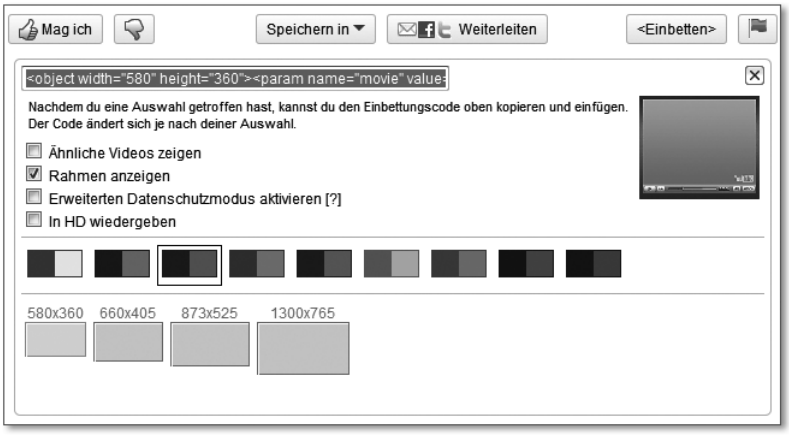


Abbildung 15.14 Aussehen des Videos festlegen und Einbettungscode generieren

Bei YouTube öffnet sich nach dem Klicken auf EINBETTEN ein weiterer Bereich mit Einstellmöglichkeiten für das spätere Video. Nehmen Sie die gewünschten Einstellungen vor, und kopieren Sie anschließend den Quelltext in der Codeansicht an die Stelle, an der später das Video erscheinen soll.

```
<body>
<object width="580" height="360">
  <param name="movie" value="http://www.youtube.com/v/AeQY_GGHSr0&
   hl=de_DE&fs=1&rel=0&color1=0x2b405b&color2=0x6b8ab6&border=1">
  </param>
  <param name="allowFullScreen" value="true">
  </param>
  <param name="allowscriptaccess" value="always">
  </param>
  <embed src="http://www.youtube.com/v/AeQY_GGHSr0&hl=de_
DE&fs=1&rel=0&color1=0x2b405b&color2=0x6b8ab6&border=1"
  type="application/x-shockwave-
flash" allowscriptaccess="always" allowfullscreen="true" width="580"
height="360"></embed>
</object>
</body>
```

Listing 15.2 Skript für YouTube-Einbindung

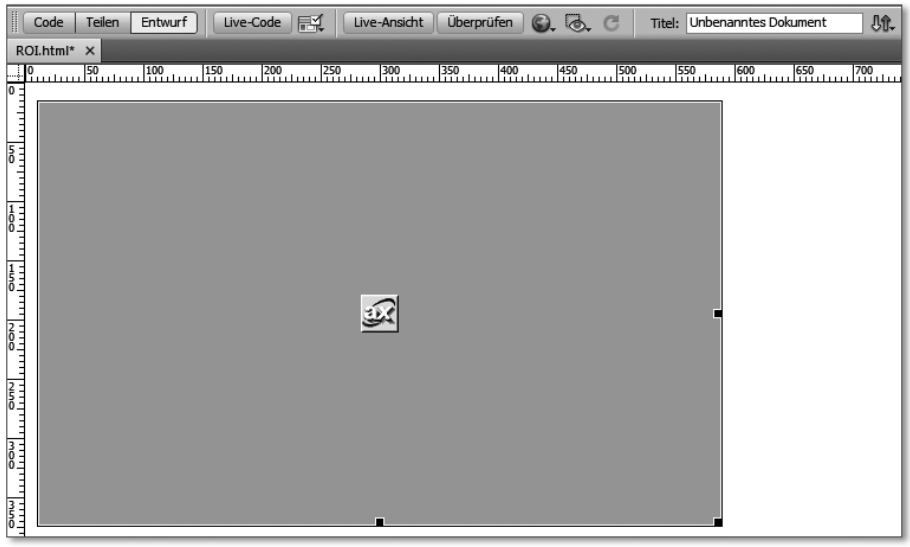


Abbildung 15.15 Eingebettetes Video in Dreamweaver

In Dreamweaver wird wie in allen vorherigen Beispielen das Video zunächst als grauer Kasten angezeigt. Die Eigenschaftenpalette bietet zwar auch hier eine Option, den Film direkt in Dreamweaver abzuspielen, meist funktioniert das aber nur sehr eingeschränkt.



Abbildung 15.16 Eingebundenes YouTube-Video während der Ausführung.

Was bleibt, ist die Vorschau im Browser. Hier funktioniert alles bestens. Die Voreinstellungen werden übernommen und das Video wie gewünscht dargestellt.

Eine oft anfallende Arbeit: Inhalte aus Office-Anwendungen in ein HTML-Dokument einfügen – mit Dreamweaver ein Kinderspiel!

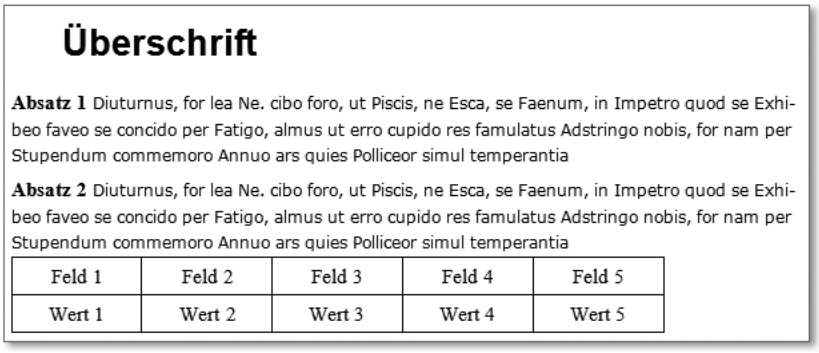
16 Office-Dateien einfügen

Wenn Sie in Dreamweaver-Versionen vor CS3 versuchten, Inhalte eines Word-Dokuments einzufügen, konnten Sie häufig über die erzielten Effekte nur noch staunen. Im besten Fall waren alle Formatierungen verschwunden, Tabellen wurden einfach ignoriert. In den aktuelleren Versionen funktioniert das Zusammenspiel jedoch schon ziemlich gut.

Das Einfügeverhalten können Sie in Dreamweaver in den Voreinstellungen beeinflussen. Lesen Sie dazu in Abschnitt 3.2.11, »Kopieren und Einfügen (Office)«, nach. Entsprechend dieser Voreinstellung verhält sich Dreamweaver, wenn Sie zunächst aus einer anderen Anwendung kopierte Daten über BEARBEITEN • EINFÜGEN oder Strg/⌘+V in ein HTML-Dokument einfügen.

16.1 Word-Texte einfügen

Word-Dokumente lassen sich in Dreamweaver auch über DATEI • IMPORTIEREN • WORD-DOKUMENTE in eine Website einfügen. Für unser Beispiel importieren wir das Dokument aus Abbildung 16.1.



Überschrift

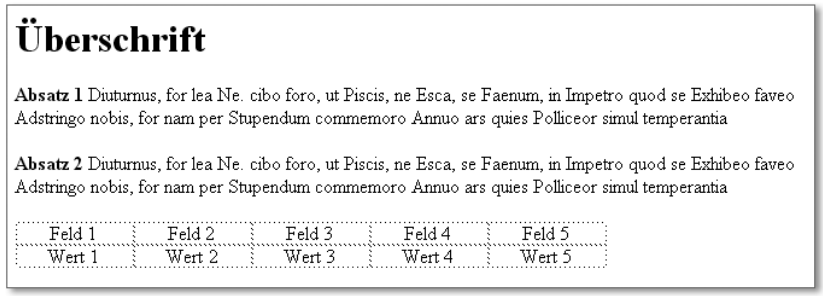
Absatz 1 Diuturnus, for lea Ne. cibo foro, ut Piscis, ne Esca, se Faenum, in Impetro quod se Exhibeo faveo se concido per Fatigo, almus ut erro cupido res famulatus Adstringo nobis, for nam per Stupendum commemoro Annuo ars quies Policeor simul temperantia

Absatz 2 Diuturnus, for lea Ne. cibo foro, ut Piscis, ne Esca, se Faenum, in Impetro quod se Exhibeo faveo se concido per Fatigo, almus ut erro cupido res famulatus Adstringo nobis, for nam per Stupendum commemoro Annuo ars quies Policeor simul temperantia

Feld 1	Feld 2	Feld 3	Feld 4	Feld 5
Wert 1	Wert 2	Wert 3	Wert 4	Wert 5

Abbildung 16.1 Word-Dokument für den Import in Dreamweaver

In Dreamweaver werden fast alle Formatierungen übernommen. Sogar die Tabelle mit den zentrierten Inhalten wird sehr gut lesbar dargestellt, ohne dass die im Quelltext von Word vorhandenen Tags auftauchen.



Überschrift

Absatz 1 Diuturnus, for lea Ne. cibo foro, ut Piscis, ne Esca, se Faenum, in Impetro quod se Exhibeo faveo Adstringo nobis, for nam per Stupendum commemoro Annuo ars quies Polliceor simul temperantia

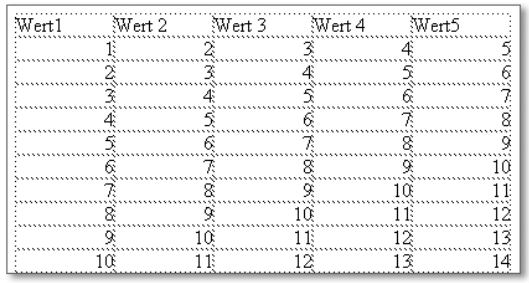
Absatz 2 Diuturnus, for lea Ne. cibo foro, ut Piscis, ne Esca, se Faenum, in Impetro quod se Exhibeo faveo Adstringo nobis, for nam per Stupendum commemoro Annuo ars quies Polliceor simul temperantia

Feld 1	Feld 2	Feld 3	Feld 4	Feld 5
Wert 1	Wert 2	Wert 3	Wert 4	Wert 5

Abbildung 16.2 Dreamweaver übernimmt die Word-Formatierungen.

16.2 Excel-Tabellen einfügen

Der Import funktioniert über Datei • Importieren • Excel-Dokument auch hervorragend mit Excel-Tabellen. Leere Zeilen und Zellfarben werden jedoch ignoriert, was aber wahrscheinlich meist zu verschmerzen ist.



Wert1	Wert 2	Wert 3	Wert 4	Wert5
1	2	3	4	5
2	3	4	5	6
3	4	5	6	7
4	5	6	7	8
5	6	7	8	9
6	7	8	9	10
7	8	9	10	11
8	9	10	11	12
9	10	11	12	13
10	11	12	13	14

Abbildung 16.3 Importierte Excel-Tabelle in Dreamweaver

16.3 CSV-Daten importieren

Ein sehr beliebtes Datenaustauschformat aus verschiedenen Office-Programmen sind CSV-Dateien. Diese lassen sich z. B. in Access, Excel oder auch in MySQL mit phpMyAdmin exportieren. Sie bestehen aus einfachen Textdateien mit einem Semikolon als Trennzeichen. Dreamweaver erstellt daraus automatisch Tabellen. Ein Beispiel für den Inhalt einer CSV-Datei:

```
Wert 1;Wert 2;Wert 3;Wert 4;Wert 5;Wert 6;Wert 7;Wert 8
```

Diese Daten können Sie direkt in Dreamweaver importieren. Öffnen Sie dazu im Menü DATEI • IMPORTIEREN • TABELLENDATEN, und das Fenster aus Abbildung 16.4 erscheint.

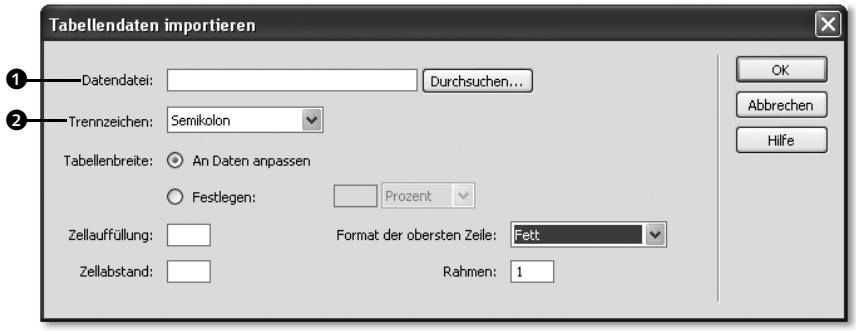


Abbildung 16.4 Importieren von CSV-Daten

Wählen Sie zuerst unter DATENDATEI ❶ den CSV-Datensatz aus. Üblicherweise wird in CSV-Dateien ein Semikolon als Trennzeichen verwendet. Es kommt aber auch vor, dass dafür andere Zeichen genommen werden. Unter TRENNZEICHEN ❷ können Sie auswählen, welches in Ihren Datendateien zur Feldtrennung verwendet wurde. In den anderen Einstellungen können Sie Vorgaben für die Formatierung der Tabelle festlegen, die Dreamweaver dann generiert.

16.4 Weitere Einfügemöglichkeiten

Im Menü BEARBEITEN • INHALTE EINFÜGEN verbergen sich noch weitere Einfügeoptionen. Erst wenn Sie einen Text oder eine Tabelle kopiert haben, ist dieser Menüpunkt aktiv, und Sie können unabhängig von den Voreinstellungen entscheiden, in welcher Weise Inhalte in Ihr HTML-Dokument eingefügt werden.

Hier gelangen Sie auch über den Button EINFÜGEN-VOREINSTELLUNGEN ❸ in die Voreinstellungen von Dreamweaver und können die Einfügevorgaben ändern.

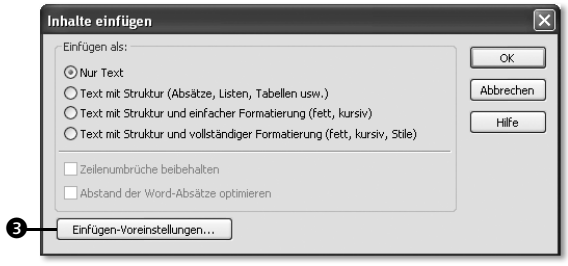


Abbildung 16.5 Dialogfenster »Inhalte einfügen«

[»] **Garantiert unformatiertes Einfügen aus Dokumenten jeglicher Art**

Wenn Sie zu 100 % sicher sein wollen, dass keinerlei Formatierung mit in das HTML-Dokument übernommen wird, kopieren Sie den einzufügenden Inhalt in einen reinen Texteditor wie zum Beispiel Notepad unter Windows. Anschließend kopieren Sie den Inhalt von Notepad in Ihr HTML-Dokument.

Trotz WYSIWYG kommen Sie in der Regel um die direkte Arbeit am Quelltext nicht herum. Schön, wenn dies komfortabel ist. Dreamweaver stellt Ihnen einige hervorragende Werkzeuge dafür zur Verfügung.

17 Quelltext de luxe

Mit umfangreicheren Websites steigt der Anteil der Arbeit im Quelltext. Viele spezifische Einstellungen können, gerade wenn es um dynamische Websites geht, nicht in der Entwurfsansicht erfolgen. Mit zunehmender Erfahrung werden Sie feststellen, dass es häufig auch wesentlich schneller geht, direkt im Quelltext zu arbeiten.

Besonders Programmierer werden die Funktionen zu schätzen wissen, obwohl Dreamweaver – das muss man leider sagen – bei der PHP-Programmierung nicht an die Möglichkeiten von zum Beispiel Zend Studio heranreicht. Hier hat sich allerdings in der aktuellen Version einiges getan und wir gehen in Kapitel 20, »PHP mit Dreamweaver«, nochmals direkt darauf ein. Hier beschreiben wir zunächst als Schwerpunkt die Möglichkeiten mit XHTML.

Meistens wird das Grundlayout in der Entwurfsansicht erstellt, diverse Variablen in PHP und SQL-Abfragen jedoch werden direkt im Code programmiert. Nur so ist eine vollständige Kontrolle über den Quelltext gegeben.

17.1 Arbeiten im Quelltext

In Abbildung 17.1 sehen Sie die Symbolleiste KODIERUNG. Die Bezeichnung ist sicherlich etwas unglücklich übersetzt. Tatsächlich verbirgt sich dahinter ein Werkzeug zum Bearbeiten und Handhaben von Quelltext. Die Leiste können Sie über ANSICHT • SYMBOLLEISTEN • KODIERUNG ein- und ausblenden.



Abbildung 17.1
Die Symbolleiste »Kodierung«

17.1.1 Quellcode formatieren

Nicht selten steht man vor dem Problem, völlig unformatierten Quelltext zu erhalten. Es ist extrem mühsam und fehlerträchtig, alle Umbrüche und Einrückungen von Hand vorzunehmen. Quelltext, der so aussieht wie in Abbildung 17.2, kann niemand lesen.



Abbildung 17.2 Unformatierter Quelltext

Automatische Formatierung

Wesentlich einfacher als in mühsamer Handarbeit geht dies mit einem Werkzeug von Dreamweaver. Klicken Sie in der Symbolleiste KODIERUNG auf den kleinen Eimer, und wählen Sie QUELLENFORMATIERUNG ÜBERNEHMEN.

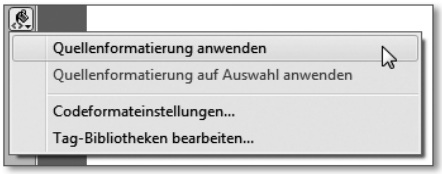


Abbildung 17.3 Quellenformatierung anwenden

Der Code wird entsprechend der von Ihnen in den Voreinstellungen festgelegten Optionen eingerückt und formatiert.



Abbildung 17.4 Formatierter, lesbarer Quelltext

17.1.2 Codeansichtsoptionen

Um die Ansichtsoptionen für die Quellcodeansicht zu verändern, öffnen Sie das Menü Ansicht • Codeansichtsoptionen. Sie können hier einige Hilfsmittel aktivieren oder deaktivieren. Wir raten Ihnen, alle einzuschalten, da der Code somit wesentlich übersichtlicher wird.

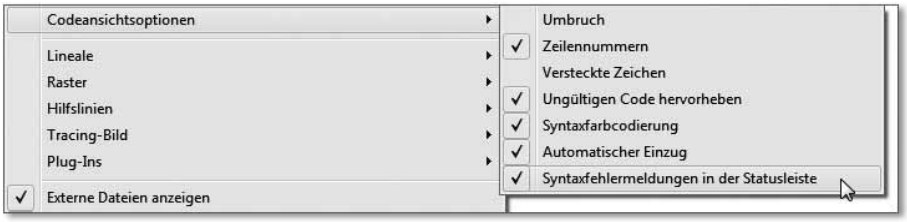


Abbildung 17.5 Codeansichtsoptionen einstellen

Sie können beispielsweise durch Zeilennummern bestimmte Stellen im Quelltext schneller finden. In PHP-Fehlermeldungen wird fast immer eine Zeilennummer mit einem Hinweis auf die Art des Fehlers ausgegeben.

Aktivieren Sie Ungültigen Code hervorheben, prüft Dreamweaver bereits während der Eingabe den Code auf Korrektheit und Browserkompatibilität. Ähnlich wie bei der Rechtschreibprüfung in Word wird fehlerhafte Syntax usw. hervorgehoben und mit roten Wellenlinien unterstrichen.

Versteckte Zeichen blenden Leerzeichen sichtbar als kleinen Punkt ein und zeigen die manuellen Zeilenumbrüche an usw. Dies ist besonders bei der Arbeit mit Tabellen und PHP hilfreich, da auch Leerzeichen eine gewisse Zeichenhöhe haben und so die Formatierung einer nicht leeren Zelle beeinträchtigen können.

Die Syntaxfehlermeldungen in der Statusleiste sind ein neues Feature seit Dreamweaver CS4. Bei erkennbaren Fehlern im Code, vor allem in JavaScript und PHP, warnt Dreamweaver mit einer gelben Zeile direkt über dem Quelltext. Gleichzeitig wird die Zeilennummer daneben rot markiert. Die Meldung verschwindet, sobald der Fehler behoben wurde.

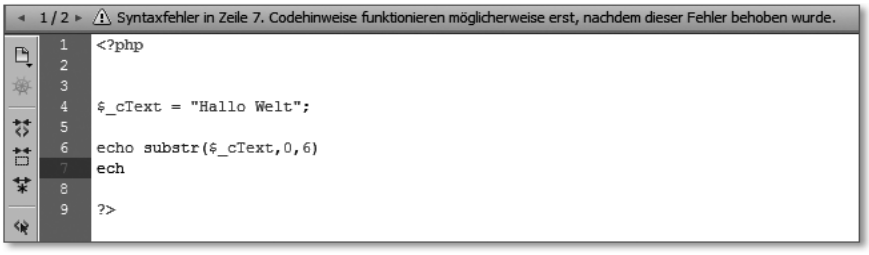


Abbildung 17.6 Syntaxfehlermeldung in der Statuszeile

17.1.3 Code übersichtlich halten

Quelltexte können mitunter sehr lang werden. Wenn Sie einige Zeilen des Codes bearbeiten müssen, stören alle anderen Bereiche meist erheblich, da sie die Lesbarkeit beeinträchtigen. Mit Dreamweaver verfügen Sie über die verschiedensten Möglichkeiten, Code auszublenden, wobei die wichtigste sicherlich das vollständige Ausblenden von markierten Bereichen ist.

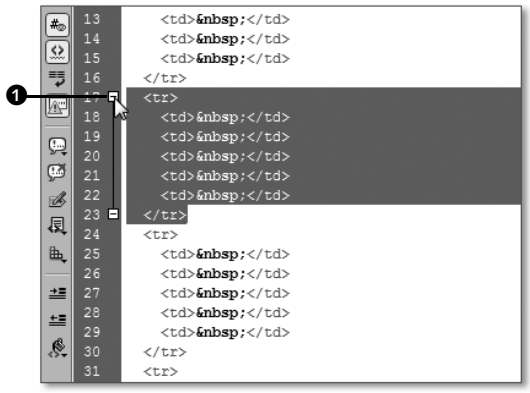


Abbildung 17.7 Bereiche des Quelltextes ausblenden

Markieren Sie dazu den gewünschten Bereich, und klicken Sie auf das kleine Icon AUSWAHL AUSBLENDEN oder auf eines der kleinen weißen Quadrate ❶ an den Zeilennummern. Alternativ kann durch gleichzeitiges Drücken der Alt-Taste der nicht ausgewählte Bereich ausgeblendet werden.

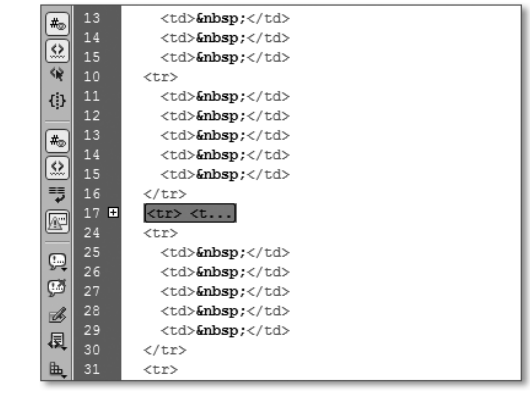


Abbildung 17.8 Ausgeblendeter Bereich des Quelltextes

Der ausgeblendete Quelltext wird durch graue Markierungen angezeigt und kann jederzeit durch einen Klick auf das kleine Plussymbol oder das Icon in der Symbolleiste wieder eingeblendet werden.

Zusammen mit den vielfältigen anderen Möglichkeiten wie etwa ganze Tags auszublenden, übergeordnete Tags auszuwählen usw. können Sie die Ansicht des Codes ganz nach den jeweiligen Anforderungen gestalten.

17.1.4 Code halbautomatisch erstellen

Während Sie direkt in der Codeansicht Tags eingeben, hilft Ihnen Dreamweaver beim Vervollständigen aller Attribute. Wenn Sie die ersten Zeichen nach der öffnenden Klammer eingeben, bietet Ihnen Dreamweaver gemäß dem eingegebenen Zeichen eine Auswahl an verfügbaren Tags an. Sie können diese durch einfaches Betätigen der ↵-Taste automatisch vervollständigen. Auch Attribute werden so ergänzt. Bei entsprechendem Attribut wird gleich das passende Auswahlfeld mit den möglichen Werten angeboten. In dem in Abbildung 17.9 gezeigten Beispiel ist es eine Farbe.

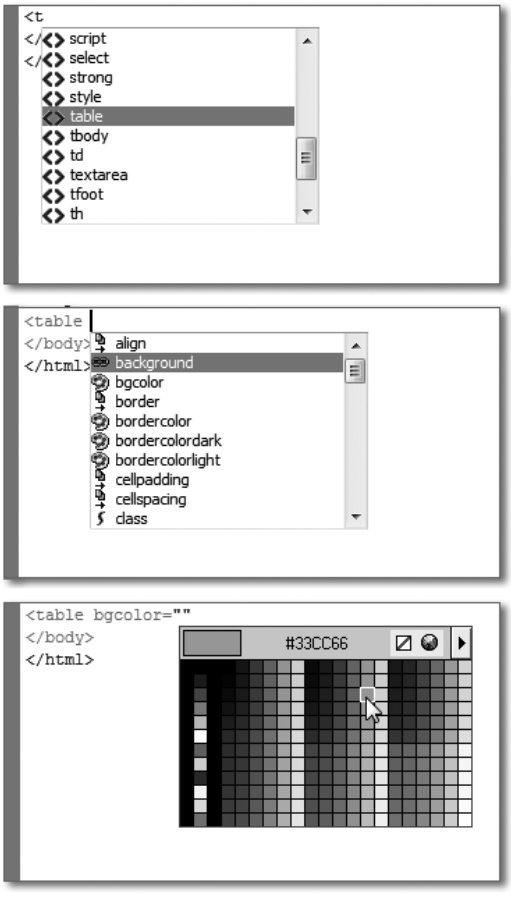


Abbildung 17.9 Eingabehilfen beim Schreiben von Code

Je nach Ihren Voreinstellungen (siehe Abschnitt 3.2.4, »Codehinweise«) werden Tags sofort abgeschlossen, oder es wird nach Eingabe einer öffnenden Klammer und eines Backslashs das zuletzt geöffnete Tag automatisch erkannt und geschlossen. Die Einstellung, dass geöffnete Tags automatisch ein Schluss-Tag erhalten ist sehr fehlerträchtig, da man ein solches Tag schnell übersieht.

17.1.5 Kommentare

Zum Zeitpunkt der Entwicklung haben Sie wahrscheinlich die Struktur der Website bestens im Kopf und kennen jede Variable mit Vornamen. Nach sechs Monaten Projektpause sieht das schon etwas anders aus. Unkommentiert wird selbst der eigene Quelltext schnell zur Hieroglyphensammlung und für Dritte sowieso schon mal kaum nachvollziehbar. Wenn man sich bei XHTML Kommentare auch oft sparen kann – bei PHP sollte man auf dieses Hilfsmittel nicht verzichten.

Quelltext dokumentieren

Fügen Sie deshalb an geeigneten Stellen Kommentare ein, die Ihnen auch nach sechs Monaten erlauben, den Code zu verstehen. In Dreamweaver können Sie manche Hinweise auch in sogenannten *Design Notes* hinterlegen (siehe Abschnitt 4.2.9, »Design Notes«). So vermeiden Sie bei XHTML, CSS und JavaScript, dass im Internet Angaben zu finden sind, durch die Surfer mit unlauteren Absichten Sicherheitslücken finden könnten.

Kommentare können Sie über das Menü EINFÜGEN • KOMMENTAR oder über die entsprechenden Icons in der Symbolleiste KODIERUNG erstellen.

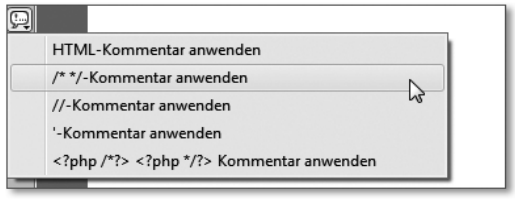


Abbildung 17.10 Kommentar-Funktionen in der Codeansicht

17.2 Arbeiten in der Live-Code-Ansicht

Dreamweaver CS4 brachte die Live-Code-Ansicht als neues Feature mit. Seit dieser Version ermöglicht Dreamweaver nicht nur die Vorschau dynamischer Websites in einem simulierten Webbrowser, sondern auch einen Blick »hinter die Kulissen« durch die Möglichkeit, direkt in den durch eine Skriptsprache generierten Quellcode zur Laufzeit zu blicken.

In unserem Beispiel ist es PHP; gerne wird dieses Feature aber auch mit JavaScript und Ajax-Bibliotheken eingesetzt. So kann im Code beobachtet werden, wie sich Inhalte des Dokuments dynamisch ändern.

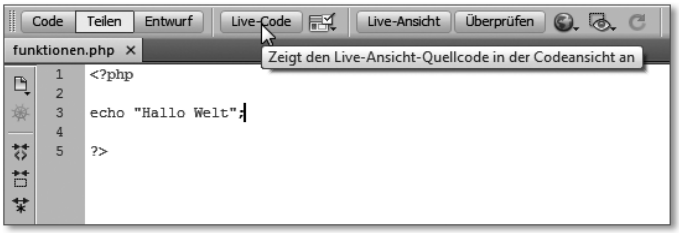


Abbildung 17.11 Aktivieren der Live-Code-Ansicht

Zum Aktivieren der Live-Code-Ansicht klicken Sie auf die Schaltfläche Live-Code.

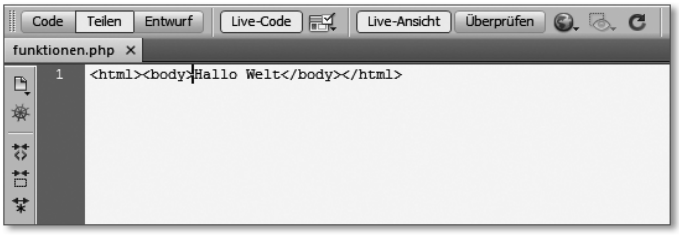


Abbildung 17.12 Live-Code-Ansicht

Der von der Skriptsprache generierte Quelltext wird nun Gelb hinterlegt angezeigt.

17.3 Codefragmente (Snippets) einsetzen

Bibliotheken und Templates, die in Kapitel 13, »Vorlagen und Bibliotheken«, beschrieben werden, haben den Nachteil, dass sie sitespezifisch sind. Innerhalb einer anderen Website ergeben sie kaum Sinn. Im Lauf der Arbeit werden Ihnen immer wieder Skripte oder HTML-Elemente begegnen, die Sie gerne wiederverwenden möchten. Besonders im Hinblick auf dynamische Websites ist es sehr nützlich, wenn man diese Codefragmente speichern kann. Eine gute Programmierung in PHP und in anderen Sprachen zeichnet sich schließlich auch durch die Wiederverwendbarkeit des Codes aus.

Code wiederverwenden

Dreamweaver unterstützt dies, indem es Ihnen ermöglicht, Codefragmente dauerhaft zu hinterlegen. Sie können die einmal erstellten Codefragmente in jeder Dreamweaver-Site verwenden und auch mit anderen Dreamweaver-Nutzern tei-

len. Voraussetzung für einen Einsatz über Site-Grenzen hinweg ist, dass Sie keine Pfadangaben in Ihrem Codefragment verwenden, die in anderen Site-Strukturen nicht funktionieren können.

Nehmen Sie zum Beispiel eine Datentabelle. Diese kann in jedem neuen HTML-Dokument in modifizierter Form wiederverwendet werden, um Inhalte in gleichbleibender Form darzustellen.

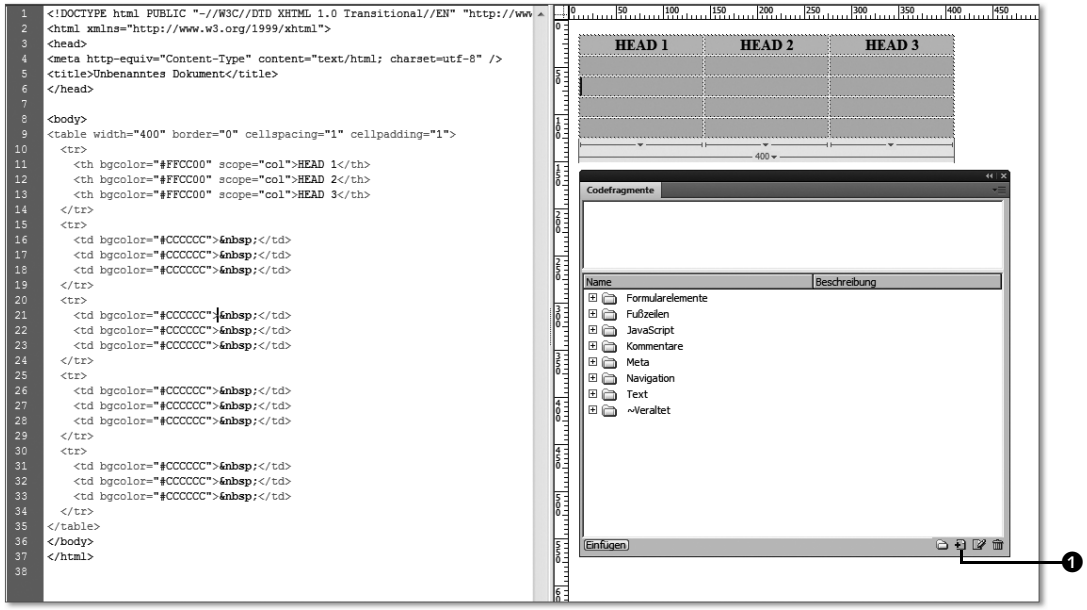


Abbildung 17.13 Codefragmente-Palette

Markieren Sie in Ihrem Quelltext den gewünschten Bereich, und öffnen Sie das Bedienfeld DATEIEN • CODEFRAGMENTE.

Wählen Sie einen Ordner, in den Sie das Codefragment einfügen möchten, aus der Auswahlliste, oder erstellen Sie mit einem Rechtsklick in den Fensterbereich einen neuen Ordner (siehe Abbildung 17.14). Um ein neues Codefragment einzufügen, klicken Sie danach auf das kleine Pluszeichen ❶ (in Abbildung 17.13) und wählen im folgenden Dialog BLOCK EINFÜGEN ❹ (in Abbildung 17.15).

An Cursorposition einfügen

Sie können Ihrem Codeblock optional eine Beschreibung hinzufügen. Bei der Auswahl BLOCK EINFÜGEN wird das Codefragment – wie der Name schon sagt – in einem Block an der Stelle des Cursors im Dokument eingesetzt, wenn Sie dieses Fragment später verwenden.

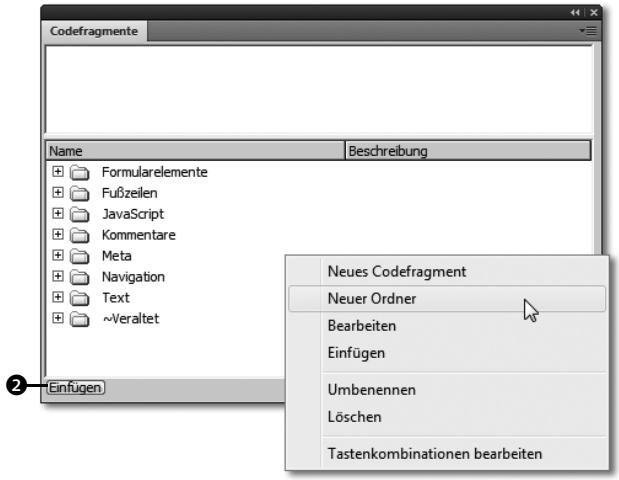


Abbildung 17.14 Anlegen eines neuen Ordners

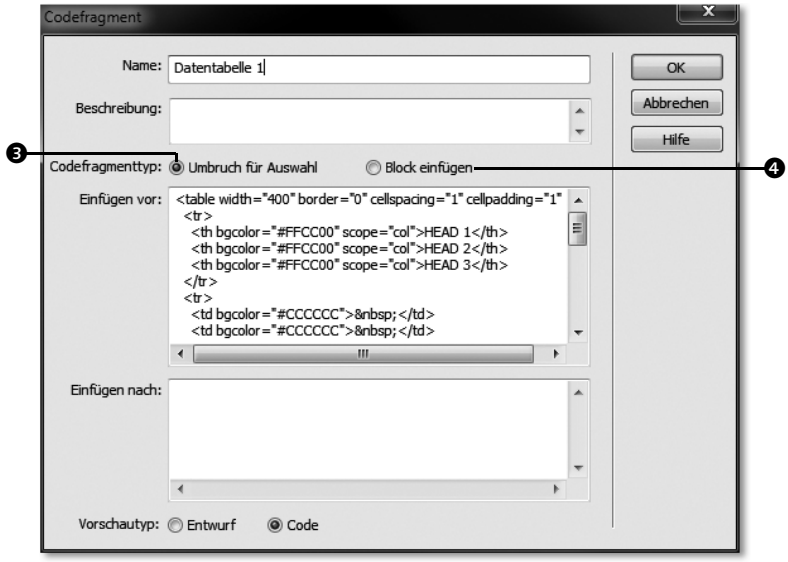


Abbildung 17.15 Tabelle als Codefragment

Um ein Tag herum einfügen

Wenn Sie in der Dialogbox den Codefragmenttyp UMBRUCH FÜR AUSWAHL ❸ anwählen, können Sie Codefragmente erstellen, die um ein bestehendes Tag herum eingefügt werden. Dieser Codeblock besteht aus zwei Teilen. Für den ersten Teil müssen Sie angeben, was vor dem ausgewählten Tag eingefügt werden soll, und für den zweiten, was nach der Auswahl eingefügt wird.

Wenn Sie Codefragmente anwenden möchten, klicken Sie einfach im Quelltext oder – je nach Ihrem Codefragmenttyp – auch in der Layoutansicht an die gewünschte Stelle und dann auf EINFÜGEN ❷ (in Abbildung 17.14) im CODEFRAGMENT-Fenster.

17.3.1 Codefragmente mit Kollegen teilen

Die von Ihnen erstellten Codefragmente werden in einem Ordner mit dem Namen *Configuration/Snippets* gespeichert. Dieser befindet sich in den Anwendungsdaten Ihres Rechners bei dem angemeldeten Benutzer. Sie können diesen Ordner an andere Anwender weiterreichen und kopieren.

Besonders interessant sind Codefragmente bei dynamischen Websites. Sie können die unterschiedlichsten Abfragen anfertigen und immer wieder verwenden. Die Arbeit an dynamischen Websites ist mit Codefragmenten häufig einfacher und wesentlich effektiver als mit den in Dreamweaver integrierten Features. Zusätzlich haben Sie die vollständige Kontrolle über den Quellcode.

[!] **Snippets gehen verloren**

Wenn Sie Dreamweaver vollständig löschen und neu installieren, wird auch der *Snippets*-Ordner gelöscht. Sichern Sie daher vor einer Neuinstallation unbedingt alle von Ihnen erstellten Codefragmente!

17.3.2 Tag-Inspektor

Eine mächtige Funktion stellt der TAG-INSPEKTOR zur Verfügung. Er bietet Ihnen zu jedem Tag die richtigen Attribute in einer Auswahlliste an. In dieser können Sie alle Einstellungen vornehmen und müssen die Attributnamen nicht von Hand in den Code eintragen. Um mit dem TAG-INSPEKTOR zu arbeiten, öffnen Sie die Bedienfeldgruppe TAG.

Klicken Sie im Dokument einfach in das zu bearbeitende Tag, und stellen Sie im TAG-INSPEKTOR die Attribute ein.

17.3.3 Eigene Tag-Vorgaben oder -Bibliotheken

Wenn Sie bestimmte Schreibweisen eines Tags bevorzugen oder die Darstellung im Quelltext dauerhaft verändern möchten, können Sie die Tag-Bibliothek bearbeiten oder eine neue anlegen. Besonders interessant ist dies auch in Kombination mit XML.

Wählen Sie dazu aus dem Menü BEARBEITEN • TAG-BIBLIOTHEKEN. Für jedes vorhandene Tag können Sie nun die Einstellungen verändern und die Formatierungen beeinflussen.

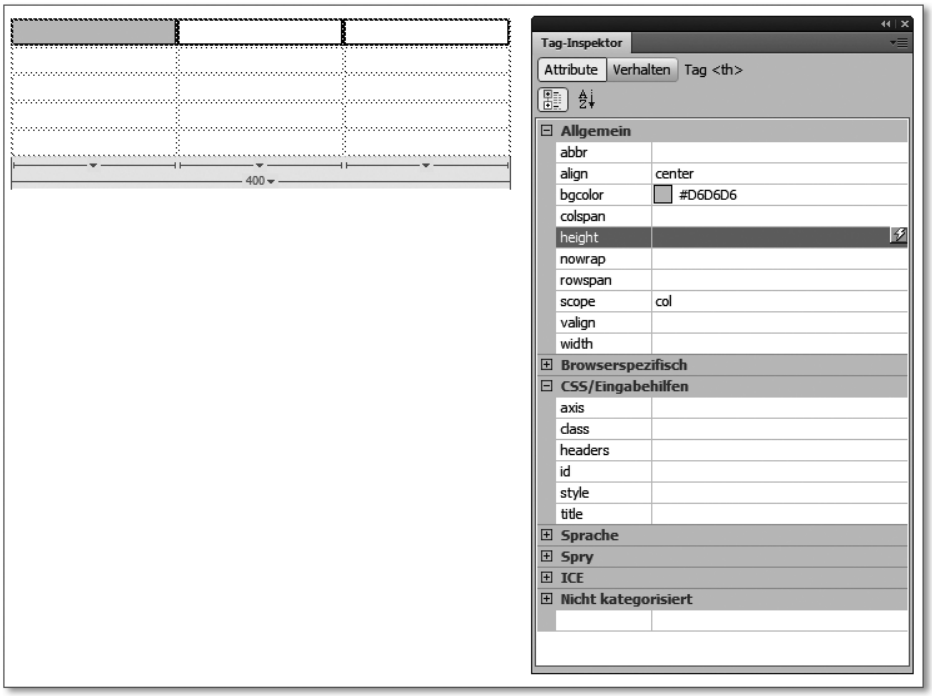


Abbildung 17.16 Attribute einer Tabellenzelle im Tag-Inspektor

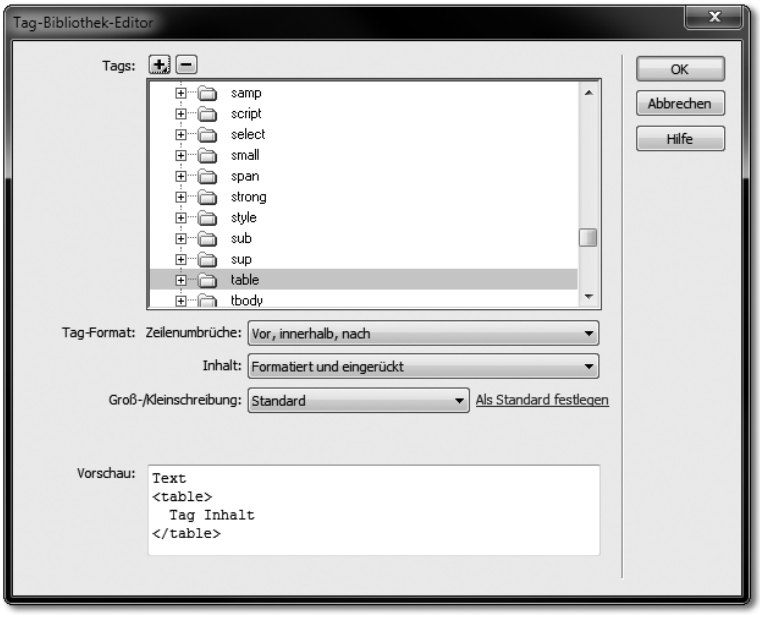


Abbildung 17.17 Bearbeiten der Tag-Bibliothek

Im Zusammenspiel mit den modularen Möglichkeiten von XHTML ist es sehr interessant, eigene Tag-Bibliotheken zu erstellen.

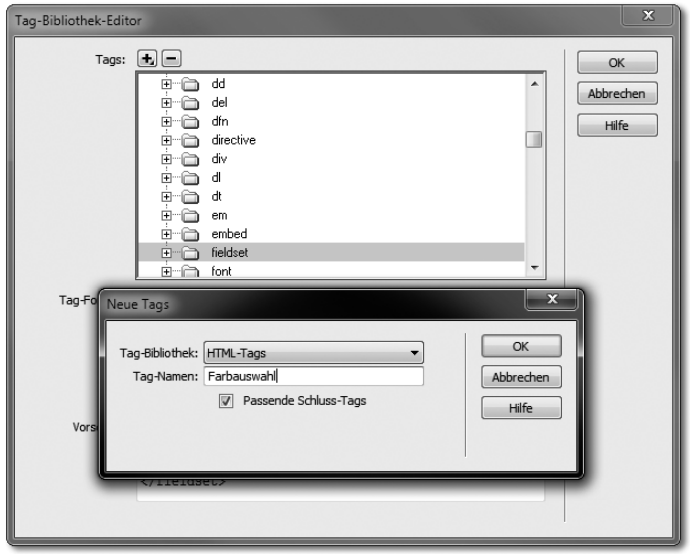


Abbildung 17.18 Eigenes Tag anlegen

In Abbildung 17.18 haben wir ein neues HTML-Tag mit der Bezeichnung *Farbauswahl* angelegt.

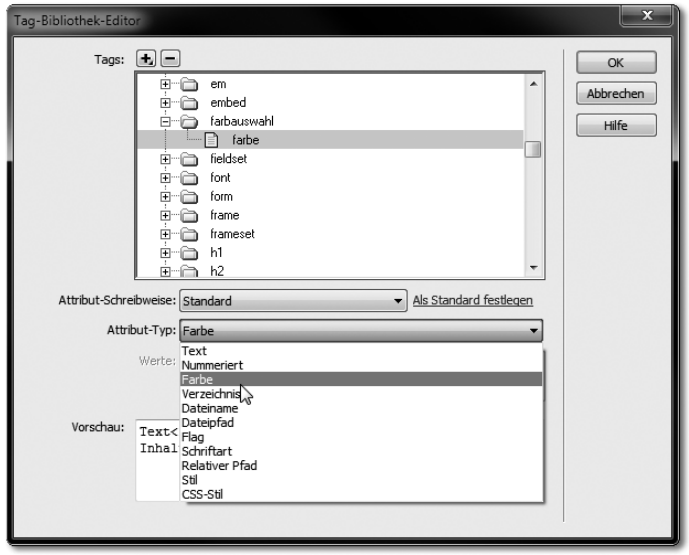


Abbildung 17.19 Eigenes Tag »farbauswahl« mit Attribut des Typs »Farbe«

Zu jedem vorhandenen Tag – auch zu den selbstangelegten – können Sie weitere Attribute hinzufügen. Der Typ des Attributs legt fest, welche Inhalte die Attributwerte haben dürfen und wie diese Attribute in der Eingabehilfe beim Code angezeigt werden.

Eigene Tags werden in der Codeansicht und im TAG-INSPEKTOR behandelt wie Standard-Tags.

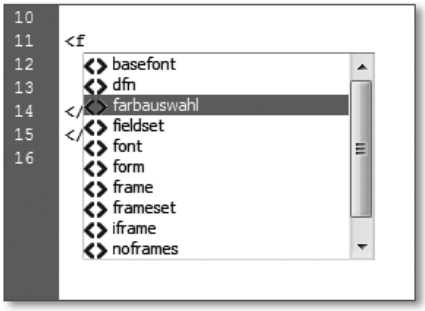


Abbildung 17.20 Eigenes Tag im Einsatz

17.4 Quelltext in der Entwurfsansicht bearbeiten

Sie können Quelltexte auch bearbeiten, ohne in die Codeansicht zu wechseln. Im Dokumentfenster werden Ihnen unten die Tags der aktuellen Auswahl angezeigt. Markieren Sie eines davon, werden in der Eigenschaftenpalette die verfügbaren Attribute aufgelistet. Auf der rechten Seite erscheint der Quick-Tag-Editor ❶. Diesen können Sie auch mit Strg+T aufrufen. Dahinter verbirgt sich ein Fenster mit dem Quelltext des Tags. Hier können Sie direkt in den Quelltext schreiben.



Abbildung 17.21 Code in der Layoutansicht bearbeiten

[»] **Elemente im Quelltext finden**

In umfangreichen Quelltexten ist es nicht immer einfach, ein bestimmtes Element zu finden. Gerade bei verschachtelten Tabellen ist die Suche nach einer konkreten Zelle oft eine ziemliche Raterei.

Einfacher geht's, wenn Sie im Layoutmodus das gewünschte Element markieren und dann in die Codeansicht wechseln (das geht am schnellsten mit dem Tastaturkürzel Strg+#). Das zuvor im Layoutmodus markierte Element ist auch in der Codeansicht ausgewählt.

17.5 Quelltext automatisch optimieren

Dreamweaver ermöglicht Ihnen, den Quelltext automatisch zu optimieren. Wenn Sie den Code vollständig in Dreamweaver erstellt haben, führt eine Optimierung meist zu keinen Veränderungen. Äußerst sinnvoll ist diese Funktion jedoch, wenn Ihnen HTML-Dateien aus Word oder anderen Programmen vorliegen.

Fast alle Textverarbeitungs-, Tabellenkalkulations- und Layoutprogramme bieten mittlerweile eine Option an, Dokumente als HTML-Datei zu speichern. Diese Dokumente und auch Dokumente anderer WYSIWYG-Editoren können Sie automatisch optimieren lassen. Rufen Sie dazu das Menü Befehle • XHTML-Code optimieren auf.



Abbildung 17.22 Dialogbox »HTML/XHTML-Code optimieren«

[!] **Niemals dynamische Dokumente optimieren**

Zu schweren Fehlern kann die Optimierung von dynamischen Dokumenten führen. Sehr oft werden Tags nicht korrekt geschlossen, um verschiedene Bedingungen zu ermöglichen. Dreamweaver weist dies als Fehler aus und würde das Dokument zerstören.

17.6 Suchen und ersetzen

Während der Entwicklung einer Website kommt es immer wieder vor, dass einzelne Tags, Attribute von Tags oder einfach nur Texte innerhalb eines Dokuments oder der ganzen Website ersetzt werden müssen. Ein komfortables und leistungsfähiges Werkzeug zum Suchen und Ersetzen ist daher unabdingbar.

Dreamweaver bietet Ihnen alle erdenklichen Möglichkeiten, in einzelnen oder mehreren Dokumenten, Ordnern oder gleich der gesamten lokalen Site zu suchen und zu ersetzen. Sie rufen das Dialogfenster über BEARBEITEN • SUCHEN UND ERSETZEN oder über das Tastaturkürzel Strg/⌘+F auf.



Abbildung 17.23 Dialogbox »Suchen und ersetzen«

Suche in mehreren Dokumenten

Wählen Sie zunächst aus, wo gesucht werden soll. Wenn Sie in mehreren Dokumenten suchen oder ersetzen wollen, empfehlen wir Ihnen, diese gezielt vorher zu öffnen. Nur so steht Ihnen in den Dokumenten eine Rückgängig-Funktion zur Verfügung, um eventuelle Fehler rückgängig zu machen. Das Suchen und Ersetzen dauert in nicht geöffneten Dokumenten außerdem deutlich länger.

Tags und Attribute suchen

Neben der reinen Textsuche können auch bestimmte Tags gesucht und Attribute neu gesetzt werden. Mit dieser Funktion ist eine sehr gezielte Suche möglich.

Nach Ihrer Aktion wird automatisch das Ergebnisfenster geöffnet, und alle Aktionen oder Treffer werden angezeigt. Durch einen Doppelklick auf das jeweilige Dokument wird dieses direkt geöffnet, und Sie können die Funktion nochmals überprüfen.

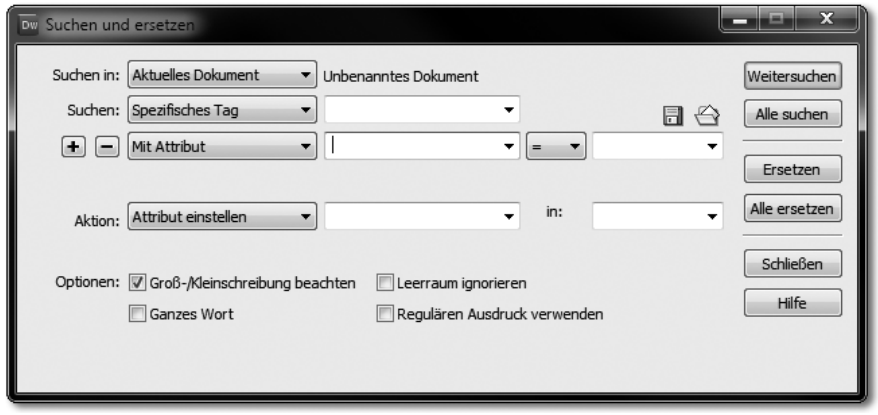


Abbildung 17.24 Suchen und ersetzen innerhalb bestimmter Tags

[!]

Dokumente sichern!

Achten Sie genau darauf, vor dem Durchführen einer Suchen-und-ersetzen-Aktion die Dokumente – oder besser sogar die ganze Website – zu sichern. Es passiert wirklich sehr schnell, dass die gesamte Site nicht mehr zu gebrauchen ist, weil Sie bei den Vorgaben ein Leerzeichen vergessen haben. Eine Rückgängig-Funktion gibt es in Dreamweaver nur für geöffnete Dokumente!

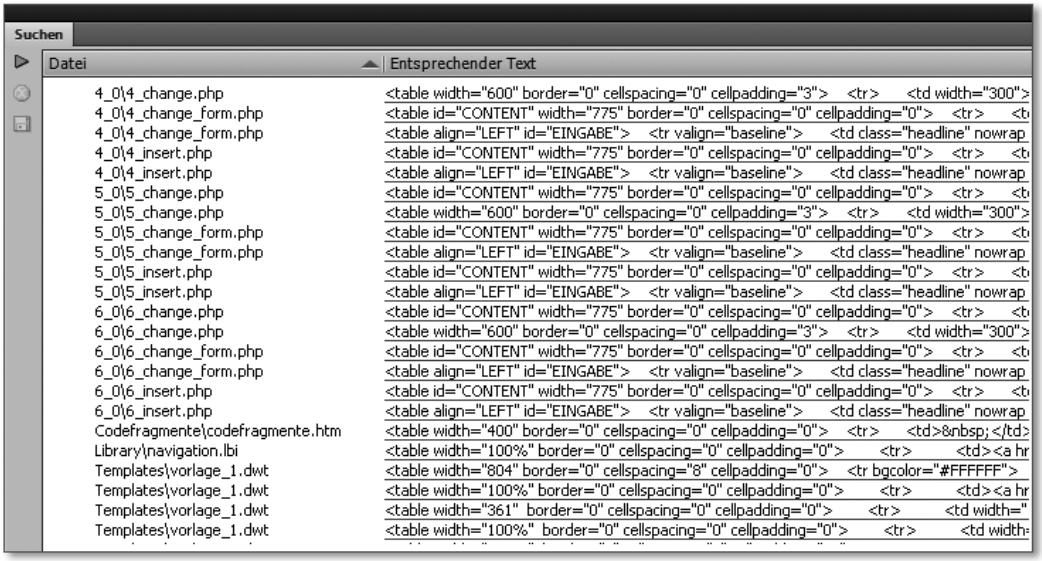


Abbildung 17.25 Die Ergebnisse werden im Ergebnisfenster angezeigt.

TEIL III
Dynamische Websites



Nachdem wir nun erfolgreich eine statische Website aufgebaut haben, wollen wir uns dem Aufbau dynamischer Seiten widmen. Zunächst beschäftigen wir uns mit einigen Grundlagen.

18 Dynamische Websites – Einführung

Das Wort Dynamik bezieht sich in der Webentwicklung auf das Generieren der Inhalte von (X)HTML-Dokumenten aus Datenquellen aller Art. Um Websites zu dynamisieren, gibt es zwei grundlegende Vorgehensweisen:

1. Die Daten werden in Echtzeit beim Aufruf der Website durch eine Skriptsprache als CGI aus der Datenbank gelesen, aufbereitet und *dynamisch in (X)HTML-Dokumenten ausgegeben*. Diese Methode hat den Vorteil, dass die Website sehr aktuell ist. Änderungen in der Datenbank werden sofort auf der Website dargestellt. Der Nachteil ist, dass bei hoher Besucherfrequenz die Serverlast enorm zunimmt und Suchmaschinen immer wieder Probleme mit aus Datenbanken generierten Websites haben. Dennoch bietet die Methode viele Vorteile, auf die wir gleich noch ausführlich eingehen werden.
2. Die Website wird, sozusagen auf Knopfdruck, aus einer Datenbank generiert und in Form von *statischen HTML-Dokumenten auf dem Server abgelegt*. Bei Firmenwebsites, die sich nicht ständig ändern, kann man über diese Methode nachdenken. Suchmaschinen können diese Seiten meist problemlos indizieren. Mit dieser Methode ist auch die Ausgabe von XML-Dateien möglich.

18.1 Vorteile dynamischer Sites

Der Folgeaufwand in der Programmierung und Administration ist geringer als bei rein statischen Websites. Und auch die weiteren Vorteile liegen auf der Hand: Wenn es bislang so war, dass Firmen beim Ändern der Website ihre Internetagentur anrufen mussten, können sie Änderungen am Seiteninhalt mit einem Content-Management-System einfach selbst vornehmen. Neue Inhalte werden automatisch online verfügbar gemacht. Nach einem etwas höheren Initialaufwand beim Programmieren der Website sind die Folgekosten geringer.

18.1.1 Statische versus dynamische Sites

In den letzten Jahren haben wir 80 % der neuen Websites dynamisch aufgebaut. Der Trend dahin ist unverkennbar. Es gibt sicherlich viele Gründe dafür, die auch in der breiten Verfügbarkeit und einfachen Einsetzbarkeit von Skriptsprachen wie PHP zu suchen sind.

PHP und MySQL

PHP ist einfach zu programmieren. Mit ein wenig Übung ist eine einfache dynamische Website ebenso schnell erstellt wie eine statische, da viele Skriptelemente (entsprechende Programmierung vorausgesetzt) wiederverwendbar sind.

Noch immer herrscht in vielen Agenturen andächtiges Schweigen, wenn von dynamischen Websites die Rede ist, und massive Kosten werden prognostiziert. Es gibt keinen Grund dafür. PHP und MySQL erfordern in der Regel keine hohen Entwicklerkosten, und es gibt massenhaft Open-Source-Projekte, die Ihnen alles an Mustern und Tools zur Verfügung stellen, was Sie benötigen. Außerdem stehen mit Typo3, Joomla! und ähnlichen CMS-Lösungen mittlerweile auch im Open-Source-Bereich Enterprise-Lösungen zur Verfügung, die den Vergleich mit kommerziellen und sehr teuren Produkten nicht scheuen müssen. Inzwischen stellt sich nicht mehr die Frage, ob es für eine Anforderung ein Open-Source-Produkt gibt, sondern, welches der vielen man verwendet.

18.1.2 Dynamische Sites und Suchmaschinen

Dass Suchmaschinen Websites mit Datenbankanbindungen nicht indizieren, ist ein Gerücht, das nicht stimmt. Die Nachteile sind etwas anders gelagert. Dynamische Websites haben unter Umständen einen sehr kurzen Lebenszyklus. Inhalte von heute sind morgen bereits wieder offline oder unter anderen Pfaden zu finden. Das führt unter Umständen dazu, dass in Suchmaschinen Seiten indiziert sind, die es so gar nicht mehr gibt.

[»] **Rewrite-Engine (mod_rewrite)**

Ein häufiges Argument gegen dynamische Websites sind auch deren seltsame URL-Angaben. In der Tat sieht die URL einer dynamischen Website anders aus als die einer statischen, und Suchmaschinen finden nicht die Begriffe, die zur korrekten und guten Indizierung notwendig sind. Hier kann mit dem Apache-Modul *mod_rewrite* Abhilfe geschaffen werden. Dieses Modul erlaubt die nahezu beliebige Manipulation der URL (im Positiven wie Negativen). Weitere Informationen dazu finden Sie zum Beispiel unter: *http://www.modrewrite.de/*

18.1.3 Dynamische Sites und Sessions

Bei dynamischen Websites wird häufig mit Sessions gearbeitet. Das sind kleine Datenfiles, die so lange existieren, wie Sie auf eine Site zugreifen. Häufig werden damit zum Beispiel Warenkörbe bei Shop-Systemen realisiert.

Session-IDs

Beim Zugriff auf eine Seite wird eine Session-ID erzeugt, die in einem Cookie gespeichert oder einfach an die URL angehängt wird. Dabei entstehen in der Regel ellenlange Nummern in der Browser-Adressleiste. Eine Session-ID existiert nur für eine bestimmte Zeit, danach wird sie gelöscht. Greift nun eine Suchmaschine auf eine Site zu, wird ebenfalls eine Session erzeugt. Die Seite wird dann inklusive der Session-ID indiziert. Greift die Suchmaschine dann ein weiteres Mal auf die Seite zu, gibt es diese ID nicht mehr, und die Suchmaschine erhält eine Fehlermeldung. Die Seite fliegt dann entweder aus dem Index, oder es wird gleich ihre ganze URL gelöscht.

Seiten nicht indizieren

Seiten, die mit Sessions arbeiten, sollten Sie daher gegebenenfalls von der Indizierung durch Suchmaschinen ausschließen. Das ist nicht schwierig: Starten Sie Sessions einfach nie schon auf der ersten Seite, geben Sie auf dieser in den Metaangaben `<nofollow>` ein, und lassen Sie dann die erste Seite indizieren. Eine andere, allerdings aufwendigere Methode ist, gleich auf dem Webserver bei einem Request abzufragen, ob eine Suchmaschine zugreift. Suchmaschinen können dann umgeleitet werden, z. B. auf eine spezielle Seite, deren Inhalt mit vielen Metaangaben und Schlagwörtern einer guten Platzierung dient.

Websites ohne Session und mit beständigen Kennungen werden, auch wenn sie dynamisch sind, sehr gut indiziert.

18.2 Typen dynamischer Websites

Es gibt viele Einsatzbereiche für dynamische Websites. Auch wenn sich diese in der Anwendung stark voneinander unterscheiden, funktionieren sie technisch gesehen alle ähnlich. Die wichtigsten sind im Folgenden aufgeführt.

18.2.1 Web-Content-Management-Systeme (WCMS)

WCMS dienen der Pflege und Verwaltung von Website-Inhalten über das Internet. Nicht nur die Website wird dafür online gestellt, sondern auch die Adminis-

trationsoberfläche, mit der man Inhalte anlegt und verändert. Im einfachsten Fall geschieht dies per E-Mail. Aufwendigere Systeme benutzen auch online verfügbare WYSIWYG-Editoren für die grafische Aufbereitung von Inhalten. Hinzu kommt die Benutzerverwaltung, die dafür sorgt, dass Mitarbeiter nur auf die ihnen zugedachten Bereiche zugreifen können.

WCMS gibt es als kostenlose Open-Source-Tools und als ausgefeilte, zu lizenzierende Systeme für Preise von bis zu mehreren 100.000 €, wobei man erstaunt feststellen wird, dass die extrem teuren Lösungen keineswegs die besten sind.

In den nächsten Kapiteln werden wir ein kleines WCMS aufbauen. Es ist allerdings recht einfach gehalten, ansonsten würde es den Umfang dieses Buches sprengen.

18.2.2 Content-Management-Systeme (CMS)

Der Unterschied zu den WCMS ist, dass hier keine eindeutige Ausrichtung auf den Zugriff über das Web besteht, sondern in abgeschlossenen Netzwerken gearbeitet wird. Die meisten CMS haben allerdings eine WCMS-Erweiterung.

18.2.3 Redaktionssysteme

Ein Redaktionssystem ist ein CMS mit besonderer Ausrichtung auf redaktionelle Arbeit. Ein Journalist beispielsweise befindet sich in seiner Redaktion oder irgendwo anders auf der Welt und schreibt seinen Artikel in das Redaktionssystem. Dieser wird dann von einem Redakteur gegengelesen und freigeschaltet. Redaktionssysteme sind für diesen Arbeitsablauf optimiert.

Ein weiterer Einsatzzweck, auf den man dank XML immer öfter trifft, ist Crossmedia Publishing. Inhalte für Print-, Onlinemedien, Handys usw. werden aus einer einzigen Datenquelle generiert und auf Knopfdruck in allen Formaten publiziert.

18.2.4 Shopsysteme

Sie alle kennen so etwas – denn jeder von uns hat schon einmal online eingekauft. Das besondere Kennzeichen eines Shopsystems ist die Bestellmöglichkeit mit Warenkorb. Die angezeigten Inhalte, die im Prinzip genau wie bei einem WCMS generiert werden, lassen sich vom User sammeln und als Bestellung abschicken. Die Sammelfunktion, der Warenkorb, ist das Ausschlaggebende bei einem Shopsystem. Es gibt auch Shopsysteme mit kompletter Fakturierung und Rechnungsstellung.

Fertige Shopsysteme

Der Markt für Shopsysteme ist nahezu unüberschaubar. Ein passendes System für einen Kunden zu finden ist eine anspruchsvolle Aufgabe. Mittlerweile bieten einige Provider Shopsysteme zur Miete an (T-Online, 1&1 usw.). Für kleine Hobbyshops sind diese Systeme durchaus ausreichend. Leider kann das Layout dieser Systeme meist nicht hinreichend beeinflusst werden, und eine vollständige Integration in eine Website ist kaum möglich.

18.2.5 Foren

Auch ein Forum ist vom Prinzip her ein WCMS – mit dem Unterschied, dass praktisch jeder Benutzer Rechte hat, Artikel zu posten, und auf die Artikel der anderen Benutzer antworten kann. Ein sehr verbreitetes und durchweg zu empfehlendes Open-Source-Forensystem ist phpBB.

Ein Forum lebt von der Moderation der Beiträge, weniger durch ausgefeilte Technik. Ein wirklich gut geführtes Forum ist daher keine Nebenbeschäftigung, sondern bedarf der exakten Planung sowie redaktioneller Vor- und Nachbereitung. So geführt kann es ein Besuchermagnet und ein Know-how-Beweis sein. Foren können idealerweise auch den Kundendienst unterstützen und eine Nutzergemeinschaft auf hohem Niveau bilden. Aus Marketingsicht kann ein gutes Forum unmittelbaren Einblick in Kundenverhalten und -wünsche geben.

18.2.6 Portale

Portale sind eine Mischung aus WCMS und umfangreichen Linklisten. Ein mit der Thematik CMS verbundenes Portal ist z. B. *http://www.contentmanager.de.* Der Gedanke hinter einem Portal ist, Usern zu einem speziellen Thema alle verfügbaren Informationen, Links, Artikel und Foren auf einer Website gesammelt zu präsentieren und den Betrieb in der Regel über Werbeeinnahmen zu finanzieren.

Während der Zeit des Internet-Hypes wurden Portale oft als Gelddruckmaschinen angesehen. Heute hat sich dies relativiert. Der tatsächliche Nutzen ist außer in branchenspezifischen Portalen eher gering. Der administrative Aufwand jedoch ist enorm. Sehr gute Portale sind beispielsweise die Website der Stadt Hamburg (*http://www.hamburg.de*) und das Portal der Bundesverwaltung (*http://www.bund.de*). Im Zusammenspiel mit Portalen bieten sich auch personalisierte Inhalte an.

18.2.7 Personalisierte Websites

Ein ganz wichtiger Aspekt dynamischer Websites ist die Möglichkeit, einem User speziell auf ihn zugeschnittene Informationen zu präsentieren und damit die Attraktivität des Angebots massiv zu steigern.

Wenn sich ein Benutzer durch ein Login zu erkennen gibt und er an anderer Stelle sein Interesse an bestimmten Inhalten bekundet hat, können diese individuell aus der Datenbank generiert werden. Ein Einsatzbereich hierfür sind beispielsweise Onlinezeitschriften. Wenn jemand möchte, dass er als Startseite die Sportnachrichten sieht, dann bekommt er diese vor allen anderen Inhalten angezeigt. Bei anderen Portalen und fast allen Free-Mailern (GMX, Web.de, Firemail usw.) müssen Sie zum Registrieren ein Interessenprofil erstellen. Anhand dieses Profils werden dann die Werbeinhalte zusammengestellt.

Cookies

Fragwürdig sind anhand von User-Tracking personalisierte Websites. Damit wird Ihr Weg durch eine Website verfolgt und protokolliert. Wenn Sie dann zwanzigmal die neuesten Turnschuhe angeklickt haben, können Sie sicher sein, beim nächsten Login alle Turnschuhe gleich auf der Startseite präsentiert zu bekommen.

Einige Anbieter installieren Cookies bei Ihnen, die sogar Ihre Wege im ganzen Web verfolgen. Gelangen Sie dann wieder auf die Website eines an dieses System angeschlossenen Anbieters, werden diese Cookies ausgewertet und Ihnen entsprechende Inhalte angezeigt. Um dagegen vorzugehen, installieren Sie am besten SpyBot oder Ad-Aware. Sie werden Augen machen, wie verbreitet diese Methode ist. 150 Cookies auf einem System nur für das User-Tracking sind keine Seltenheit.

18.2.8 Weblogs

Es gibt verschiedene Weblog-Typen. In den einfachsten Fällen ist das ein Onlinetagebuch. Zu fast allen wichtigen und unwichtigen Themen gibt es mittlerweile Weblogs. Innerhalb kürzester Zeit ist ein regelrechter Hype um das »Bloggen« entstanden. Ein Kennzeichen vieler Weblogs ist die Querverlinkung zu anderen Weblogs. Diese so entstandene Community wird auch als *Blogosphäre* bezeichnet.

Blogs sind einfachste WCMS. Sie ermöglichen das schnelle Aktualisieren von Website-Inhalten ohne HTML-Kenntnisse, erreichen aber nicht die Leistungsfähigkeit eines vollwertigen WCMS.

Wenn Sie den kompletten Teil III dieses Buches über dynamische Sites durchgearbeitet haben, werden Sie sicherlich in der Lage sein, einen einfachen Weblog zu erstellen. Wir können Ihnen dies als Übung sehr empfehlen. Eine Schritt-für-Schritt-Anleitung haben wir dafür allerdings nicht erstellt, da das zugrunde liegende Prinzip eines Weblogs der von uns erstellten Struktur sehr ähnlich ist.

Dies ist nur eine kleine Auswahl der wichtigsten Applikationen, die man mit dynamischen Websites entwickeln kann. Es gibt noch viele andere Anwendungen, so etwa Webmailer und andere Tools, die eine Interaktion des Benutzers mit einer Datenbank benötigen.

18.2.9 Social Networks

In aller Munde sind sogenannte *Social Networks*. Allen gemeinsam ist, das der Benutzer bzw. die Gemeinschaft der Benutzer hier die Inhalte erstellt, austauscht und verlinkt. Mittlerweile ist es durchaus wichtig, mit seinen eigenen Seiten in diesen Netzwerken gefunden zu werden oder Verlinkungen zu diesen Netzwerken bilden zu können. In manchen Branchen und spezifischen Zielgruppen ist es sogar erfolgsentscheidend, möglichst überall »im Gespräch« zu sein.

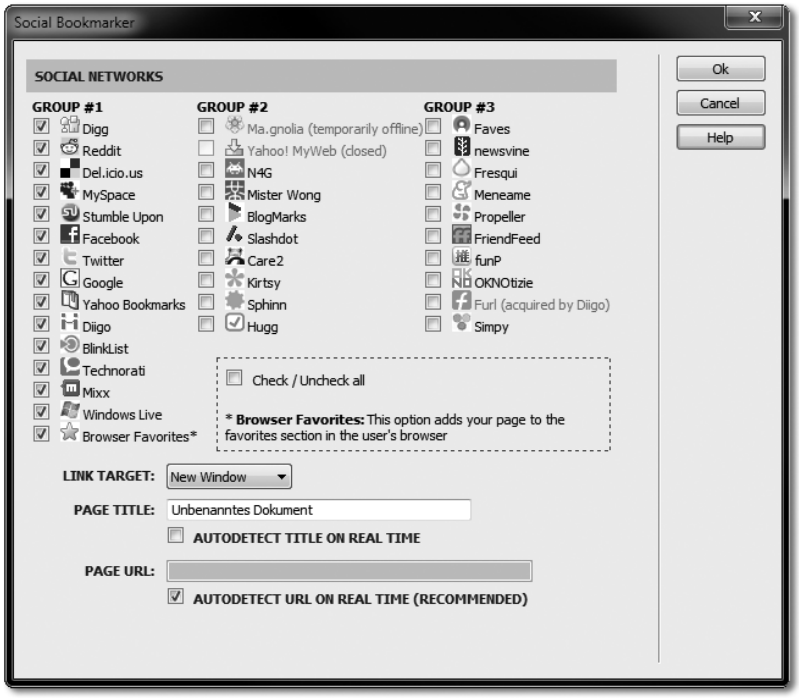


Abbildung 18.1 Erweiterung: Social Bookmarker

Hier hilft eine kleine, leider kostenpflichtige Erweiterung: der Social Bookmarker (*http://www.topdreamweaverextensions.com*). Sie ermöglicht die automatische Verlinkung zu den Social Networks auf einer Website, da sie für jedes ausgewählte Netzwerk ein Icon in das XHTML-Dokument einfügt (siehe Abbildung 18.1).

18.3 Client- und serverseitiges Skripting

Um die folgenden Kapitel über die Erstellung einer dynamischen Site in Dreamweaver zu verstehen, ist zunächst ein Exkurs in die technischen Abläufe bei dynamischen Websites notwendig.

HTTP-Requests

Beim Aufruf einer Website sendet Ihr Rechner, der Client, eine Anfrage (HTTP-Request) an den Webserver. Dieser sammelt, vereinfacht ausgedrückt, die von Ihnen angeforderten Daten und sendet diese an Ihren Client zurück. Zu diesen versandten Daten können auch Skripte mit JavaScript usw. gehören.

Das HTML wird sofort in Ihrem Browser dargestellt, JavaScripts werden temporär abgespeichert und erst dann ausgeführt, wenn sie ausgelöst werden. Ihre Ausführung ist dabei von der auf Ihrem Rechner installierten Software abhängig. Die unterschiedlichen Effekte, die dabei auftreten können, sind bekannt. Eine unserer liebsten Aufgaben ist, dafür zu sorgen, dass Websites auf möglichst vielen Plattformen laufen und identisch aussehen.



Abbildung 18.2 Clientseitiges Skripting

Clientseitiges Skripting

Diese Art der Verarbeitung nennt man *clientseitiges Skripting* (siehe Abbildung 18.2). Ein Shopsystem, das einen Warenkorb mit JavaScript realisiert, funktioniert nur auf Browsern mit aktiviertem JavaScript. Benutzer ohne passende Browser können somit nicht einkaufen. Würde man mit dieser Methode auch Datenbanken abfragen, sähen auch hier nur die User mit aktiviertem JavaScript die Inhalte der Website. Zudem ist eine Datenbankabfrage mit JavaScript nur

schlecht zu realisieren. Würde man etwa eine Personalisierung einbinden, hätten Sie sehr schnell große und völlig unüberschaubare Quelltexte.

Serverseitiges Skripting

Um vom Zielsystem unabhängig zu werden und überhaupt Datenbanken vernünftig abfragen zu können, verwendet man *serverseitiges Skripting* (siehe Abbildung 18.3).

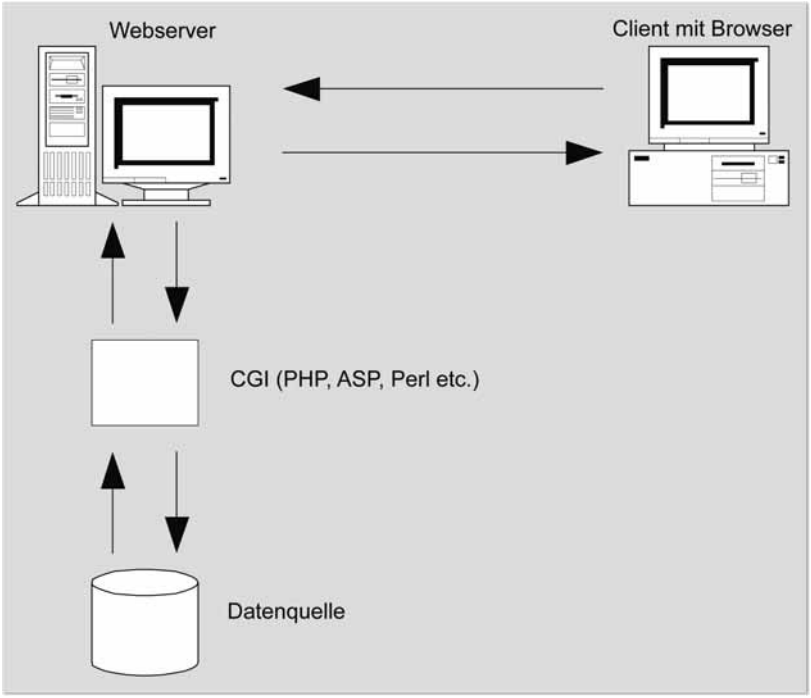


Abbildung 18.3 Serverseitiges Skripting

Hierbei läuft auf dem Webserver eine Skriptsprache. Diese setzt einen installierten Interpreter auf dem Webserver voraus. Serverseitige Skripte funktionieren nach dem folgenden Prinzip:

1. Bei der Anforderung eines entsprechenden Dokuments wird ein *CGI*-Skript (*Common Gate Interface*), das z. B. in PHP programmiert ist, gestartet.
2. Dieses analysiert nun die Anfrage und stellt eine Verbindung zu der Datenbank her, die die angefragten Daten enthält.
3. Es holt die gewünschten Daten hervor und generiert damit ein fertiges HTML-Dokument.
4. Dieses wird dann vom Webserver an den Client zurückgeschickt.

Von dem ganzen Vorgang bekommt der User nichts mit. Er sieht in seinem Browser das Ergebnis in reinem HTML. Das eigentliche CGI-Skript bleibt vor ihm verborgen. Aus diesem Grund finden Sie bei PHP-Websites im Quelltext keinen einzigen PHP-Befehl.

Ein Beispiel aus dem wirklichen Leben

Um dies noch einmal für alle zu verdeutlichen, die noch nichts mit diesen Technologien zu tun hatten, folgt hier ein Vergleich mit einem Vorgang aus dem wirklichen Leben.

Stellen Sie sich vor, Sie bestellen online oder per Telefon eine Pizza. Stellen Sie sich weiter vor, Sie sind jetzt der Browser oder Clientrechner. Der Webserver ist der Pizza-Bringdienst. Sie haben jetzt eine Pizza (Website) mit bestimmten Inhalten (Salami, Spinat, Brokkoli und viel Käse) angefordert. Auf Serverseite (Pizzaservice) sitzt jemand, der Ihre Bestellung (HTTP-Request) annimmt. Dieser leitet sie weiter an unser CGI, den Koch. Der Koch (CGI) entscheidet anhand der Bestellung (HTTP-Request), welche Zutaten er benötigt, und holt diese aus dem Regal (der Datenbank). Aus diesen Zutaten und Ihren Anforderungen backt er nun Ihre Pizza (das HTML-Dokument). Anschließend schickt er diese mit einem Pizzataxi (dem HTTP-Protokoll) zu Ihnen, dem Client. Am Ende sitzen Sie da und genießen Ihre Pizza.

Was Sie jetzt damit machen – essen, verschenken oder zerteilen – ist dann sozusagen clientseitig und interessiert den Bringdienst nicht. Sie haben weder den Telefondienst noch den Koch oder die Küche gesehen und trotzdem Ihr Abendessen bekommen.

Genau so funktionieren dynamische Websites. Das CGI-Skript, das diesen ganzen Vorgang steuert, läuft auf dem Webserver und ist somit völlig unabhängig von Ihrem Rechner – das generierte HTML-Dokument jedoch nicht. Es unterliegt wie bei statischen Seiten den individuellen Gegebenheiten des jeweiligen Browsers. In skriptgenerierten HTML-Dokumenten können nun wieder JavaScripts enthalten sein, die Ihren Browsertyp erkennen und die passende HTML-Datei anzeigen.

[»] **Skriptings mischen**

In der Praxis ist eine Vermischung von client- und serverseitigen Skriptings durchaus üblich. Relevante Abfragen und Bedingungen werden serverseitig ausgeführt. Userabhängige Aktionen wie z. B. Rollover-Effekte usw. werden clientseitig programmiert und ausgeführt. Wenn diese dann nicht auf dem Client funktionieren, wird zumindest das Basisdokument mit den Inhalten dargestellt.

18.4 Konzeption dynamischer Sites

Die Konzeption ist bei dynamischen Sites wesentlich bedeutungsvoller als bei statischen Sites. Nicht nur die Gestaltung, sondern auch die technischen Abläufe müssen bis ins Detail entworfen werden. Wir fassen an dieser Stelle die wichtigsten Punkte zusammen.

18.4.1 Besonderheiten für das Layout

Bei statischen Websites haben Sie alles fest in der Hand. Der Kunde gibt Ihnen die Texte und Bilder. Sie bearbeiten diese und binden sie ein. Bei dynamischen Websites wissen Sie nicht, welche genauen Inhalte sich einmal auf der Seite befinden werden. Sie erhalten zwar Beispielinhalte, ob diese später jedoch genau so eingesetzt werden, ist fraglich.

Ihre Site sollte daher kein Layout haben, das mit Textlängen oder Bildgrößen zu restriktiv umgeht. Das Layout muss flexibel genug sein, um lange wie auch kurze Inhalte darstellen zu können. Auf den meisten dynamischen Websites sind daher die Inhalte nach unten hin scrollbar. In den meisten Fällen ist es sinnvoll, so zu verfahren. Es sind aber auch andere Layouts möglich. Diese erfordern allerdings einen höheren technischen Aufwand. Es ist immer abzuwägen, ob ihr Einsatz aus Benutzersicht sinnvoll ist oder nicht.

Textnavigation

Ein weiterer wichtiger Punkt ist die Navigation. Üblicherweise werden Sie eine Navigation mit Grafiken erstellen. Das geht prinzipiell auch bei dynamischen Websites. Dennoch ist es ein Kundenanliegen, schnell einen neuen Navigationspunkt einfügen, entfernen oder an eine andere Stelle verschieben zu können. Grafische Menüs bieten sich daher nur sehr bedingt an. PHP unterstützt zwar das dynamische Erstellen von JPEGs, die Qualität ist aber relativ schlecht. GIFs dürfen aus lizenzrechtlichen Gründen nicht dynamisch generiert werden. Das Generieren von Grafiken lastet einen Server zudem stark aus. Es bleibt als Fazit, dass Navigationen immer am besten mit Textlinks erstellt werden. Dank CSS und Selektoren ist dies auch kein Problem und kann ebenfalls sehr gut aussehen.

Sie sollten ebenfalls bedenken, dass eine Navigation Platz beansprucht und Sie auch für Navigationserweiterungen Raum vorhalten müssen. Wenn Sie Ihrem Kunden die Möglichkeit einräumen, eigenständig Navigationspunkte verändern und hinzufügen zu können, muss das Menü genauso flexibel gehalten werden wie der Rest des Layouts.

Bei der im Buch erstellten Website finden diese eben genannten Aspekte keine Berücksichtigung, da das Anwenden aller im Buch gezeigten Techniken im Vordergrund steht.

18.4.2 Trennung von Layout und Inhalt

Besonders bei dynamisch eingespeisten Inhalten ist es sinnvoll, mit CSS zu arbeiten. Damit wird der Bereich in einer Website mit CSS-Stilen vorformatiert, der später die dynamischen Inhalte enthält. Geschickt programmiert, kann der Kunde oder Redakteur dann einfügen, was er möchte. Der Inhalt wird sich immer Ihren gestalterischen Rahmenbedingungen anpassen. Und wenn Sie dann doch mal etwas daran verändern müssen, können Sie das ohne großen Aufwand in der externen CSS-Datei durchführen.

18.4.3 Konzeption der Administrationsoberfläche

Viele dynamische Websites benötigen eine Administrationsebene, wenn die Inhalte vom Kunden gepflegt werden sollen. Legen Sie das Layout und die Benutzerführung der Administrationsoberfläche wirklich idiotensicher an. Jeder mögliche Bedienfehler wird auch gemacht, da können Sie ganz sicher sein! Es geht vor allem auf Ihre Nerven und Kosten, wenn ein Kunde ständig anruft, weil er mit der Administration nicht zurechtkommt.

Usability (Anwenderfreundlichkeit)

Geben Sie einem Kunden nur die Möglichkeiten, die er wirklich benötigt. Was für Sie als Profi ein tolles Feature ist, kann für den Laien schon eine Schaltfläche zu viel sein. Es gibt eine Menge kaum bedienbarer WCMS. Gerade im hochpreisigen Niveau finden sich einige Vertreter, die wenig benutzerfreundlich sind und einen eigenen Administrator benötigen. Folgen Sie bei der Konzeption und beim Layout der bekannten KISS-Regel: »Keep it simple, stupid.« Testen Sie die Benutzerführung der Administration am besten mit völlig unerfahrenen Usern, und suchen und beheben Sie so die Fehlerquellen.

Zugriffsrechte

Wenn Sie ein umfangreiches CMS planen, achten Sie daher darauf, welche Möglichkeiten Sie Ihren Usern zur Pflege der Daten geben und wie qualifiziert die zu erwartenden Administratoren im Umgang mit Websites sind. Unterteilen Sie am besten die User in Gruppen mit verschiedenen Zugriffsrechten. Jemand, der Texte ändern muss, sollte nicht unbedingt Systemparameter verändern können.

All diese Anforderungen machen die Angelegenheit für Sie nicht gerade einfacher. Im Gegenteil: Je benutzersicherer eine Administrationsebene sein muss, desto mehr müssen Sie sich Gedanken über die Benutzerführung machen. Bei größeren CMS mit tausend und mehr zu pflegenden Dokumenten ist der Aufwand für die Administrationsebene und für eine einwandfreie Benutzerverwaltung in der Regel höher als für die eigentliche Website.

18.4.4 Planung, Planung, Planung ...

Bevor Sie die ersten Abfragen schreiben oder die erste Datenbank anlegen, müssen Sie viel Zeit in eine gute Planung investieren. Die eigentliche Programmierung der Datenbank und der PHP-Skripte ist dann mit einiger Übung schnell erledigt. Unsere ersten dynamischen Websites haben wir bestimmt fünfmal von neuem aufgebaut, bis sie zufriedenstellend funktionierten. Sehen wir uns heute den Quelltext dieser Seiten an, stehen uns die Haare zu Berge.

Je mehr Zeit Sie für eine exakte und gute Planung investieren, desto leichter fällt Ihnen die anschließende Umsetzung. Mit zunehmender Erfahrung und Sicherheit wird diese mit Dreamweaver zur reinen Fleißarbeit.

Wenn man bei einer statischen Website noch einen Anteil von geschätzten 30 bis 40 % der gesamten Arbeitszeit für die reine Programmierung veranschlagen sollte, verschiebt sich der Anteil der Programmierung bei einer dynamischen Website auf etwa 20 %. Dafür steigt der Anteil der Vorüberlegung und strukturellen Konzeption mit Microsoft Visio oder anderen Tools von etwa 25 auf 60 % an.

Zu oft werden hier auch die Fehler in mangelhafter Software und allen anderen möglichen Dingen gesucht. Bei diesem Thema gilt für uns folgender Leitsatz:

Ein guter Handwerker kann auch mit miserablen Werkzeugen gute Arbeit leisten. Ein schlechter Handwerker nicht. Der gute Handwerker wird sich aber ein schlechtes Werkzeug nicht antun!

Dreamweaver ist ein gutes Werkzeug. Kommen wir nun dazu, wie wir damit dynamische Websites entwickeln können.

Dynamische Websites und PHP sind aus dem Internet nicht mehr wegzudenken. In diesem Kapitel machen wir Sie mit den Grundlagen von PHP vertraut.

19 Einführung in PHP

Im Laufe des Buches wurde die Programmiersprache PHP schon mehrfach erwähnt. In den folgenden Kapiteln werden wir uns in erster Linie mit dieser Skriptsprache befassen und sie dafür zunächst etwas näher erläutern. Dieses Buch stellt jedoch keine vollständige Einführung in PHP dar. Wir werden nur sehr grob auf die nötigen Grundlagen eingehen. Für die Beispielübungen und -features sollte dies ausreichen.

Sie müssen in Dreamweaver nicht unbedingt mit dem Quelltext arbeiten. PHP-Kenntnisse in den Grundzügen sind jedoch eine Voraussetzung, um zu verstehen, wie Dreamweaver dynamische Websites erstellt.

19.1 Was ist PHP?

PHP steht für *Professional Hypertext Preprocessor* und liegt aktuell in der Version *5.3.2* vor (Stand Juni 2010). PHP ist eine serverseitige Skriptsprache, die sich speziell auf Webentwicklungen ausrichtet. Die Syntax ist an C++ angelehnt, jedoch wesentlich einfacher. PHP kann direkt in (X)HTML-Dokumente eingebunden werden.

Ausführung auf einem Webserver

PHP wird auf dem Webserver ausgeführt. Dazu wird der PHP-Interpreter benötigt. Damit wir mit PHP entwickeln können, ist es von großem Vorteil, wenn Sie PHP bzw. einen kompletten Webserver lokal auf Ihrem System installieren. In Kapitel 21 werden wir detailliert beschreiben, wie das geht.

PHP 5 mit Dreamweaver

In der aktuell vorliegenden Version von Dreamweaver wird PHP 5 unterstützt. Bei PHP 5 handelt es sich um eine Weiterentwicklung der etablierten PHP-Version 4.

Besonders hinsichtlich der Objektorientierung von PHP 4 wurde Kritik seitens der Benutzer laut. Aus diesem Grund wurden Teile von PHP 5 von Grund auf neu entwickelt, um eine objektorientierte Funktionalität einfließen zu lassen. Weitere Verbesserungen erfolgten bei der Ausführungsgeschwindigkeit und der Funktionsbibliothek.

Lesen Sie zu den Unterschieden zwischen den Versionen bitte in der Dokumentation oder in geeigneter Fachliteratur nach. Hierauf einzugehen würde den Umfang und die Zielsetzung des Buches sprengen.

19.1.1 »Hallo Welt« in PHP

Zum hohen Verbreitungsgrad von PHP haben die breite und kostenlose Verfügbarkeit (Open Source) und die relativ einfache Erlernbarkeit geführt. Das unvermeidliche »Hallo Welt«-Skript sieht in PHP folgendermaßen aus:

```
<XHTML>
    <head>
        <title>PHP-Skript</title>
    </head>
    <body>
        <?php
        echo "Hallo Welt";
        ?>
    </body>
</XHTML>
```

Listing 19.1 »Hallo Welt« in PHP

Wenn Sie im Browser den Quelltext betrachten, werden Sie feststellen, dass vom ursprünglichen Skript nichts zu sehen ist. Stattdessen werden nur die XHTML-Bestandteile ausgegeben:

```
<XHTML>
    <head>
        <title>PHP-Skript</title>
    </head>
```

```
    <body>
        Hallo Welt
    </body>
</XHTML>
```

Listing 19.2 Aus dem PHP-Skript generiertes XHTML-Dokument

PHP wird nicht im Browser ausgegeben. Es ist in gewisser Weise vielmehr ein XHTML-Generator, den man selbst programmieren kann. Eingefleischten PHP-Programmierern stehen bei dieser Definition sicherlich die Haare zu Berge; um sich die Funktion von PHP zu verdeutlichen, ist es allerdings ein guter Vergleich.

Rückgabe über »echo«

Der Befehl `echo` im Skript bedeutet so viel wie »Gib mir zurück«, und zwar in diesem Fall »Hallo Welt«. Genau das macht dieses Skript. Es gibt die Phrase »Hallo Welt« aus. Mehr passiert in unserem ersten Beispiel nicht.

Verzweifeln Sie nicht, wenn Sie versuchen, dieses Skript lokal auszuführen. Das kann nicht gehen, solange wir keinen Webserver lokal installiert haben. Haben Sie bitte noch etwas Geduld: Nach den Grundlagen werden wir einen Webserver installieren, dann können Sie auch alle Skripte nachvollziehen. Wenn Sie nicht so lange warten möchten, arbeiten Sie Kapitel 21, »Lokalen Webserver installieren«, durch, die Installation ist nicht schwierig.

Information vom Provider einholen

Wenn Sie direkt auf einem Webserver bei Ihrem Provider arbeiten und die Dateien mit FTP übertragen, muss sichergestellt sein, dass PHP und MySQL auf diesem Server installiert sind. Fragen Sie am besten nach, ob das der Fall ist. Webaccounts mit PHP und MySQL gibt es heute auch schon bei Billiganbietern, so dass die Kosten keine Rolle mehr spielen sollten.

PHP ohne MySQL

Mit PHP können Sie bereits auch ohne MySQL einige Funktionen in Ihrer Website unterbringen. So können Sie z. B. Ihre User durch umfangreiche Berechnungen führen. So richtig interessant wird PHP aber erst, wenn Daten gespeichert werden und später jederzeit wieder zum Abruf bereitstehen. Grundsätzlich unterstützt PHP dafür nahezu jede auf dem Markt vertretene Datenbank. Für unsere Zwecke haben wir uns wegen der hohen Verbreitung und der Leistungsfähigkeit für MySQL entschieden. Zudem ist es kostenlos unter *http://www.mysql.com* erhältlich.

19.1.2 Vorteile von PHP

Viele Vorteile sprechen für den Einsatz von PHP. Serverseitige Skripte, die lange nur mit Kenntnissen komplexer Programmiersprachen wie Perl möglich waren, kann man nun auch ohne große Vorkenntnisse in der Programmierung relativ schnell selbst realisieren:

- Wer jemals in Basic oder anderen Programmiersprachen entwickelt hat, wird feststellen, dass PHP sehr ähnlich ist. Erste Skripte sind sehr schnell erstellt, und die Syntax ist einfach zu verstehen. PHP ist zudem eine sehr tolerante Programmiersprache. Sie müssen keine Datentypen deklarieren und sich nicht mit verschiedenen Formaten herumschlagen.
- Für Webapplikationen zählt PHP zu den am weitesten verbreiteten Programmiersprachen überhaupt.
- Es gibt Unmengen an Open-Source-Projekten zu PHP. Im Web finden Sie viele veröffentlichte PHP-Projekte, egal ob Sie ein Forum, ein Shopsystem oder ein WCMS suchen. Mit PHP werden Sie mit Sicherheit fündig.
- PHP ist für dynamische Websites bestens geeignet. Es wurde für den Webeinsatz eingeführt und bietet als Open-Source-Standard alle notwendigen Funktionen.
- Nahezu alle Datenbanken werden mittlerweile von PHP unterstützt.
- PHP enthält sehr umfangreiche Bibliotheken für fast jeden Anwendungsbereich.

19.2 PHP und XHTML

19.2.1 PHP in XHTML einbinden

PHP-Skripte können an beliebiger Stelle im XHTML-Quelltext eingebunden werden. Das geschieht mit dem Tag

```
<?php HIER STEHT DAS SKRIPT; ?>
```

Alternativ können Sie auch einfach schreiben:

```
<? HIER STEHT DAS SKRIPT; ?>
```

Grundsätzlich funktioniert beides, wenn Sie die Buchstaben `php` jedoch weglassen, erkennt Dreamweaver das Skript nicht als PHP-Skript.

Jede PHP-Befehlszeile wird mit einem Semikolon abgeschlossen. Die Ausgabe eines einfachen Satzes sehen Sie z. B. in unserem »Hallo Welt«-Skript aus Listing 19.1.

PHP können Sie an jeder beliebigen Stelle und so oft im XHTML-Dokument einbinden, wie Sie wollen. Das funktioniert auch innerhalb eines XHTML-Tags. So können Sie beispielsweise die Hintergrundfarbe einer Tabelle aus einer PHP-Variablen generieren:

```
<table bgcolor="<?php Echo "$farbe" ?>">
```

An den Browser wird anstelle des ganzen Befehls nur der Inhalt der Variablen `$farbe` ausgegeben. Der PHP-Befehl wird bereits auf dem Server ausgeführt. In unserem Beispiel muss er nicht mit einem Semikolon abgeschlossen werden, da es sich hier nur um eine einzige Befehlszeile handelt.

»<?php« für Dreamweaver [!]

Verwenden Sie mit Dreamweaver am besten die vollständige Schreibweise: `<?php`. Dabei wird `<?` leider oft nicht korrekt interpretiert und von Dreamweaver auch nicht ausgeführt. Auch auf dem lokalen Webserver mit XAMPP wird `<?` in den Default-Einstellungen nicht korrekt ausgeführt.

19.2.2 Schreibweise von Zahlen und Zeichen

Bei PHP werden zwar detaillierte Datentypen wie ganze Zahlen oder Fließkommazahlen nicht vorgegeben, es wird jedoch in einem Skript zwischen *Zeichen* (*Strings*) und *Zahlen* unterschieden.

Variablen in der einfachen Schreibweise werden auch als Zahl behandelt. `<?php Echo 100 ?>` erzeugt die Ausgabe der Zahl 100. Stehen Variablen oder Zahlen in Anführungszeichen, versteht PHP sie als Zeichenketten. So gibt `<?php Echo "100" ?>` die Zeichenkette 100 aus. Mit Zeichenketten können keine Berechnungen durchgeführt werden.

Datentypen mischen

Auch Vermischungen von Zeichenketten (Strings) und numerischen Werten innerhalb eines Befehls sind möglich und werden häufig eingesetzt. Dafür müssen Sie dem PHP-Interpreter mitteilen, welcher Teil des Befehls als Zeichenkette und welcher als Zahl zu behandeln ist. Bei PHP bewirkt ein Punkt die Addition von Zeichenketten:

```
<?php Echo "Bitte zahlen Sie". 100 ." Euro"; ?>
```

Mit dieser Schreibweise können Sie Zeichen und Zahlen innerhalb eines einzigen Befehls ausgeben. Als Beispiel soll eine Rechnungssumme ausgegeben und die Bezeichnung »Euro« hinter den Rechenwert gesetzt werden:

```
<?php Echo "Bitte zahlen Sie". $rechnung ." Euro"; ?>
```

Würden Sie diese Zeichenverkettung (Konkatenation) nicht vornehmen, gäbe PHP eine Fehlermeldung aus. Die Funktionsweise dieser Befehlszeile ist folgendermaßen zu verstehen:

```
<?php (Jetzt kommt ein Befehl für den PHP-Interpreter)
Echo (Gib Folgendes aus:)
Anführungszeichen (Jetzt kommen Zeichen)
   Bitte zahlen Sie
Anführungszeichen (Jetzt hören die Zeichen auf)
Punkt (Hänge das, was als Nächstes kommt, an das Vorherige an)
   $rechnung (Eine Zahl)
Punkt (Hänge das, was als Nächstes kommt, an das Vorherige an)
Anführungszeichen (Jetzt kommen Zeichen)
   Euro
Anführungszeichen (Jetzt hören die Zeichen auf)
Semikolon (Befehlszeile ist jetzt zu Ende)
?> (Hier ist das PHP-Skript zu Ende – weiter mit XHTML)
```

Kommata in Berechnungen

Achten Sie bei Berechnungen auf die korrekte Schreibweise des Fließkommas. Ein Komma in einer Zahl muss in PHP als Punkt geschrieben werden. Falsch ist etwa `3,14 * 300`, richtig ist `3.14 * 300`. Besonders wichtig ist diese korrekte Schreibweise bei Berechnungen durch Benutzereingaben. Der User weiß nicht, wie er eine Zahl schreiben muss. Daher muss eine Benutzereingabe für Berechnungen immer abgefangen und auf falsche Kommasetzung überprüft werden. Am einfachsten ist es, eventuelle Kommata mit einem Skript in Punkte umzuwandeln.

19.2.3 XHTML in PHP einbinden

Neben der Einbindung von PHP in XHTML ist es natürlich auch möglich, XHTML in PHP einzubinden. Mit dem Befehl `Echo` können Sie auch komplette XHTML-Zeilen ausgeben:

```
<?php Echo "<table><tr><td> </td></tr></table>"; ?>
```

Diese Befehlszeile gibt eine Tabelle aus. Um PHP jetzt zu veranlassen, auch die für Attribute notwendigen Anführungszeichen auszugeben, wenden Sie die folgende Schreibweise an:

```
<?php Echo "<table bgcolor=\"#333366\"><tr><td> </td></tr>
</table>"; ?>
```

Der Backslash verhindert, dass der PHP-Interpreter das Anführungszeichen als PHP-Befehl interpretiert. Durch den Backslash wird mitgeteilt, dass das nachfolgende Zeichen einfach als Zeichen zu verstehen ist.

19.3 Variablen in PHP

Wie Sie gesehen haben, werden Variablen mit einem vorangestellten Dollarzeichen `$` gekennzeichnet. Es gibt in PHP mehrere Möglichkeiten, mit Variablen zu arbeiten.

Wie oben erwähnt, müssen Sie sich bei PHP nicht um die Deklaration von Variablen kümmern. Dennoch gibt es einiges zu beachten: Im Internet werden Variablen mit den in Kapitel 14, »Formulare«, beschriebenen Aktionen GET und POST übermittelt.

Variablen an die URL anhängen

Möchten Sie zum Beispiel ein Dokument aufrufen und gleichzeitig eine Variable übertragen, um zum Beispiel die ID eines Datensatzes zu übergeben, sieht die Angabe in der Adressleiste des Browsers folgendermaßen aus:

http://www.website.de/produkte.php?PRODID=234

In diesem Fall heißt die Variable `PRODID` und hat den Inhalt `234`. Diese Variable können Sie im Dokument *produkte.php* mit `echo $PRODID` ausgeben lassen oder andere Aktionen damit auslösen. Das Fragezeichen in der URL bedeutet sinngemäß: Jetzt kommen Variablen.

Mehrere Variablen anhängen

Mehrere Variablen in einer URL werden mit `&` verkettet:

http://www.website.de/produkte.php?PRODID=234&SUBID=2

php.ini

Ab PHP 4.2.0 ist die Standardanweisung in der PHP-Konfigurationsdatei *php.ini* `register_globals` auf `off` gestellt. Dies führt dazu, dass Sie bei Variablen, die an andere Dokumente übertragen werden, explizit angeben müssen, wie die Variable übertragen wurde (z. B. $_GET['PROD_ID']). An die URL angehängte Variablen sind immer GET-Variablen.

Wenn Sie vom Standard abweichen und globale Variablen verwenden möchten, bearbeiten Sie *php.ini* und setzen die Vorgabe `register_globals` auf `on`. Welche Einstellungen bei Ihnen vorliegen, können Sie mit `<?php phpinfo() ?>` überprüfen.

register_globals	On	On

Abbildung 19.1 Anzeige der register_globals mit phpinfo()

Falls Sie keinen eigenen Webserver betreiben, werden Sie die Datei *php.ini* nicht bearbeiten können. Verwenden Sie daher am besten die neuere Schreibweise für eine Variable. Auch Dreamweaver benutzt diese Schreibweise, so dass Sie sicher sein können, dass Ihre Websites auf allen Webservern lauffähig sind.



Abbildung 19.2 $_-Variablen in Dreamweaver

Variablenübertragung in URL

PHP überträgt Variablen zwischen URLs in Arrays. Sie werden als *superglobale Arrays* bezeichnet. (Arrays sind Datenfelder mit mehreren Werten und Bezeichnern.) Tabelle 19.1 zeigt eine Übersicht über die verschiedenen Arrays.

Array	Bedeutung
`$GLOBALS`	Enthält Verweis zu jeder Variablen im laufenden Skript.
`$_SERVER`	Ist das Array mit den Servervariablen.
`$_GET`	Enthält alle Variablen, die mit `HTTP GET` übertragen wurden.
`$_GET['VARIABLENNAME']`	Liest die einzelnen Variablen aus.
`$_POST`	Enthält alle Variablen, die mit `HTTP POST` übertragen wurden.
`$_POST['VARIABLENNAME']`	Liest die einzelnen Variablen aus.
`$_COOKIE`	Enthält Cookie-Variablen.
`$_FILES`	Enthält alle Variablen, die dem Skript über `HTTP POST`-Datei-Uploads angeliefert werden.
`$_ENV`	Enthält alle Umgebungsvariablen.
`$_REQUEST`	Enthält alle Variablen, die auf anderen Wegen in das Skript gelangen und keiner der gängigen Sicherheitsanforderungen entsprechen.
`$_SESSION`	Enthält Variablen, die aktuell in der Session eines Skripts registriert sind (mehr dazu erfahren Sie weiter unten in Abschnitt 19.9, »Mit Sessions arbeiten«).

Tabelle 19.1 URL-Arrays

Jedes dieser Arrays setzt sich nach dem Schema in Abbildung 19.3 zusammen.

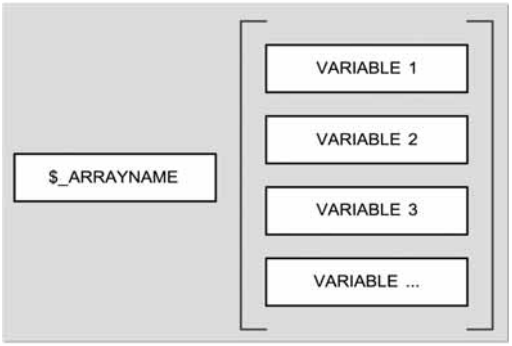


Abbildung 19.3 Array mit Variablen

19.4 Vergleichsoperatoren

Schleifendurchläufe und Bedingungen sind die grundlegenden Elemente jeder Programmiersprache. Für beides benötigen Sie die Möglichkeit, Daten zu vergleichen, um diese Aktionen durch die daraus hervorgehenden Ergebnisse zu steuern. PHP unterstützt viele Arten des Datenvergleichs. Vergleichsoperatoren können mit booleschen Funktionen verknüpft werden.

Das Ergebnis eines Vergleichs ist immer `true` oder `false`. Wird eine Bedingung erfüllt (ist sie also `true`), wird die nachfolgende Aktion im Skript ausgeführt.

Die wichtigsten Vergleichsoperatoren werden in der folgenden Tabelle aufgeführt.

Operator	Bedeutung
`A == B`	Bedingung ist erfüllt (gibt `true` zurück), wenn A und B gleich sind.
`A != B`	Bedingung ist erfüllt (gibt `true` zurück), wenn A und B ungleich sind.
`A >= B`	Bedingung ist erfüllt (gibt `true` zurück), wenn A größer oder gleich B ist.
`A <= B`	Bedingung ist erfüllt (gibt `true` zurück), wenn A kleiner oder gleich B ist.
`A > B`	Bedingung ist erfüllt (gibt `true` zurück), wenn A größer B ist.
`A < B`	Bedingung ist erfüllt (gibt `true` zurück), wenn A kleiner B ist.
`A === B`	Bedingung ist erfüllt (gibt `true` zurück), wenn A und B identisch sind.
`A !== B`	Bedingung ist erfüllt (gibt `true` zurück), wenn A und B nicht identisch sind.

Tabelle 19.2 Vergleichsoperatoren in PHP

[!] **Zuweisungsoperator als Fehlerquelle**

Achten Sie darauf, dass Sie bei einer Überprüfung von zwei Werten auf Gleichheit statt == nicht versehentlich den Zuweisungsoperator = verwenden.

19.5 Boolesche Operatoren

Mit booleschen (logischen) Operatoren können Sie z. B. die Ergebnisse einzelner Vergleichsoperationen verknüpfen.

Ein Beispiel: Die Bedingung soll erfüllt sein, wenn A gleich B oder C gleich D ist. Beide Vergleichsoperationen werden mit der Oder-Verknüpfung verknüpft:

```
(A==B) || (C == B)
```

Eine Übersicht über die booleschen Funktionen bietet Ihnen die folgende Tabelle.

Operator	Bedeutung
`A and B` `A && B`	Und-Verknüpfung: Bedingung ist erfüllt, wenn sowohl A als auch B wahr sind.
`A or B` `A || B`	Oder-Verknüpfung: Bedingung ist erfüllt, wenn A oder B wahr sind.
`A xor B`	Entweder-Oder- bzw. Exklusiv-Oder-Verknüpfung: Bedingung ist erfüllt, wenn entweder A oder B wahr ist, aber nicht beide.
`! A`	Nicht-Verknüpfung: Bedingung ist erfüllt, wenn A nicht wahr ist.

Tabelle 19.3 Boolesche Funktionen

19.6 Schleifen programmieren

Schleifendurchläufe sind die mit am häufigsten genutzten Sprachelemente. In nahezu jedem PHP-Skript sind Schleifendurchläufe enthalten, etwa zum Ausgeben mehrerer Datensätze einer Tabelle. Auch wenn Sie später mit Dreamweaver Bereiche wiederholen, werden Schleifen verwendet. Schleifen laufen immer so lange durch, bis die *Schleifenbedingung* erfüllt ist.

Grundprinzip von Schleifen

Schleifen und später auch Bedingungen bauen sich nach folgendem Schema auf:

```
Schleifen- oder Bedingungstyp ( Bedingung )
   {
   Auszuführender Codeblock bei erfüllter Bedingung
   }
```

Listing 19.3 Prinzip einer Schleife

19.6.1 for-Schleifen

Wir beginnen mit einer `for`-Schleife, die relativ komplex aufgebaut ist:

```
for ($x = 1; $x <= 10; $x++)
{
    echo $x;
}
```

Listing 19.4 for-Schleife

Diese Schleife führt die Befehle in den geschweiften Klammern so lange aus, bis `$x` den Wert 10 erreicht hat und damit die Schleifenbedingung erfüllt ist. Geschweifte Klammern umschließen immer einen Codeblock, der bei einer erfüllten Bedingung abgearbeitet wird.

Datensätze ausgeben

Mit der Schleife aus Listing 19.4 kann man zum Beispiel zehn Datensätze ausgeben lassen. Über `$x` steht die Anzahl der durchlaufenen Schleifen als Variable zur Verfügung, mit der man zusätzlich arbeiten kann.

Die Schreibweise am Ende der Schleifenbedingung `$x++` ist eine vereinfachte Schreibweise von `$x = $x + 1`. `++` bedeutet, dass der Wert um 1 inkrementiert, also erhöht wird. Analog verringert `--` den Wert um 1.

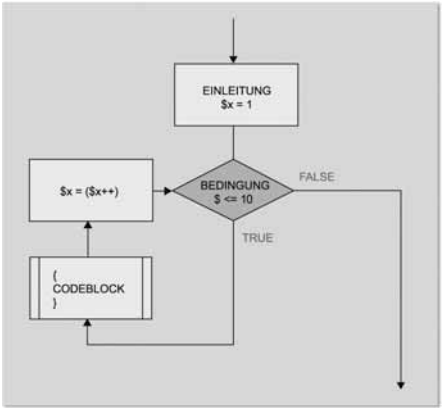


Abbildung 19.4 Schematische Darstellung einer for-Schleife

19.6.2 while-Schleifen

Eine `while`-Schleife ist einfacher aufgebaut als eine `for`-Schleife:

```
$x = 1;
while ($x <= 10)
{
    echo $x++;
}
```

Listing 19.5 Einfache while-Schleife

`$x` wird hier so lange ausgegeben, bis der Wert 10 erreicht ist. Der Wert der Variablen `$x` wird in der Schleife bei jedem Durchlauf um 1 erhöht. Natürlich kann auch jede andere Bedingung geprüft werden.

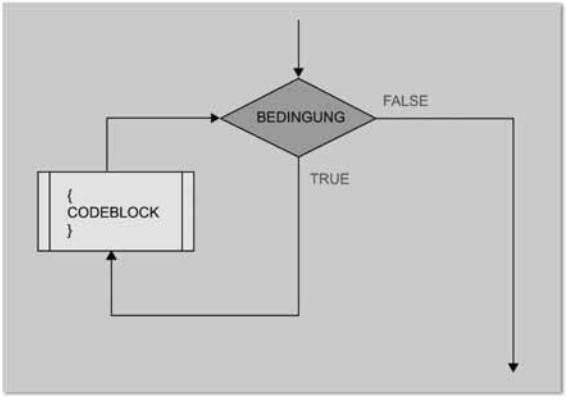


Abbildung 19.5 Schematische Darstellung einer while-Schleife

19.6.3 do-while-Schleifen

`do-while`-Schleifen sind den `while`-Schleifen sehr ähnlich:

```
$x = 0;
do
{
    echo $x;
}
while ($x>0);
```

Listing 19.6 do-while-Schleife

Der Unterschied besteht darin, dass bei diesen Schleifen das Erfüllen der Bedingungen nicht am Anfang der Schleife, sondern erst am Ende eines Durchlaufs überprüft wird. Somit kann man sicherstellen, dass die Schleife in *jedem* Fall

mindestens einmal durchlaufen wird. Bei der `while`-Schleife kann es vorkommen, dass sie nie durchlaufen wird.

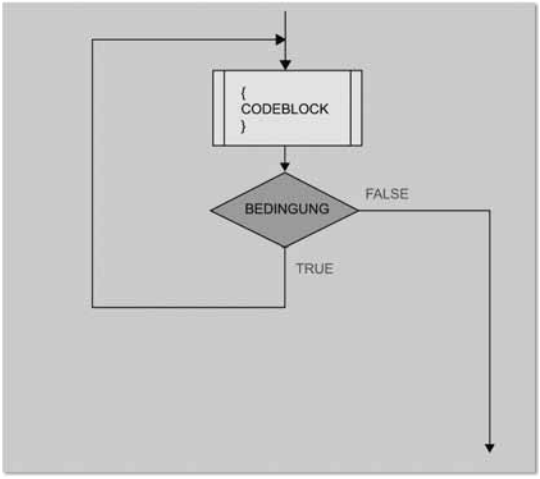


Abbildung 19.6 Schematische Darstellung einer do-while-Schleife

19.6.4 foreach-Schleifen

`foreach`-Schleifen ermöglichen die einfache Ausgabe von Arrays:

```
foreach ($array as $ausgabe)
{
    echo "aktueller Inhalt: $ausgabe";
}
```

Listing 19.7 foreach-Schleife

Diese Schleife funktioniert ausschließlich mit Arrays. Inhalte des Arrays werden einer neuen Variablen, in unserem Fall `$ausgabe`, zugewiesen und ausgegeben. Beim nächsten Schleifendurchlauf erhöht sich der Index des Arrays um 1, und der nächste Wert des Arrays wird zugewiesen und ausgegeben. Das geschieht, solange es Inhalte im Array gibt.

Alle hier beschriebenen Schleifen können in nahezu beliebiger Tiefe verschachtelt werden.

19.7 Bedingungen mit PHP

Eine der wichtigsten Anweisungen in der Programmierung überhaupt – PHP eingeschlossen – ist `if`. Mit dieser einfachen Anweisung und einer nachfolgenden

Bedingung können Programmabläufe kontrolliert und beeinflusst werden. Der prinzipielle Aufbau ist dabei denkbar einfach.

19.7.1 if-Anweisung

```
if ($A Vergleichsoperator $B)
{
Führe Folgendes aus
}
```

Listing 19.8 if-Bedingung

Wenn die Bedingung (`if`) erfüllt ist, wird der Inhalt zwischen den geschweiften Klammern ausgeführt.

19.7.2 if-else-Anweisung

Eine Abwandlung davon ist:

```
if ($A Vergleichsoperator $B)
{
Führe Folgendes aus
}
else
{
Ansonsten mache das
}
```

Listing 19.9 if-else-Bedingung

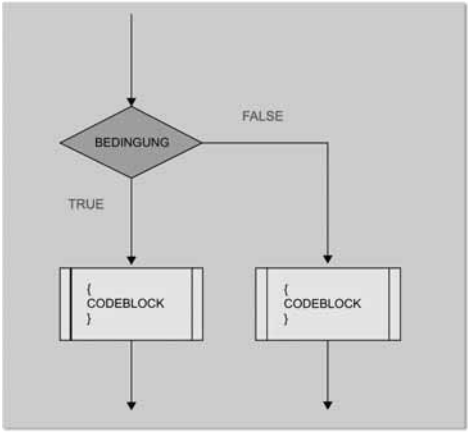


Abbildung 19.7 Schematische Darstellung einer if-else-Bedingung

Mit dem Zusatz `else` wird sichergestellt, dass im Falle der Nichterfüllung einer Bedingung in der `if`-Anweisung das ausgeführt wird, was sich in der `else`-Anweisung befindet.

19.7.3 switch-Anweisung

Mit `switch`-Anweisungen kann man – je nach Inhalt einer Variablen – sehr elegant und komfortabel zwischen mehreren Möglichkeiten auswählen, d. h. umschalten (engl. *switch*).

```
switch ($x)
{
       case 0:
           echo "Inhalt 0";
           break;
       case 1:
           echo " Inhalt 1";
           break;
       case 2:
           echo " Inhalt 2";
           break;
       default:
           echo "Fehlerhafte Daten"
}
```

Listing 19.10 Auswahl mit switch

Bei dieser Anweisung wird je nach Inhalt der Variablen `$x` eine andere Ausgabe (`case`) erzeugt. Die Ausgabe des Inhalts `default` erfolgt, wenn keine der Bedingungen erfüllt wurde.

19.8 Dateien einbinden

Bei dynamischen Websites benötigen Sie viele Skripte, die sich auf jeder Unterseite wiederholen, so zum Beispiel für die Konnektierung zur MySQL-Datenbank. Diese immer wiederkehrenden Skripte können ähnlich wie externe CSS-Stile dynamisch eingebunden werden.

Durch das Einbinden externer Skripte wird eine Website leichter pflegbar. Änderungen können an zentraler Stelle durchgeführt werden, und die Skripte können eventuell für spätere Projekte wiederverwendet werden.

Namensvergabe für externe Skripte

Achten Sie bei der Namensvergabe externer Skripte auf die Schreibweise. Kennzeichnen Sie diese eindeutig als `include`-Dateien, also Dateien, die eingebunden (engl. *include*) werden. Sie erreichen dies, indem Sie das Kürzel *inc* in den Dateinamen einfügen.

Benennen Sie jedoch die Dateien in keinem Fall *datei.inc*, sondern immer mit der Dateiendung *.php*. Dateibezeichnungen der Form *datei.inc.php* haben sich bewährt und werden häufig für `include`-Dateien verwendet. Würde ein User, mit oder ohne Absicht, eine Datei mit der Endung *.inc* aufrufen, würde diese als Text angezeigt oder zum Download angeboten. Besonders für sensible Zugangsdaten sollte das unter allen Umständen vermieden werden. Durch die Extension *.php* stellen Sie sicher, dass die Datei bei einem direkten Aufruf verarbeitet wird. Im Browser wird dann schlimmstenfalls eine Fehlermeldung angezeigt.

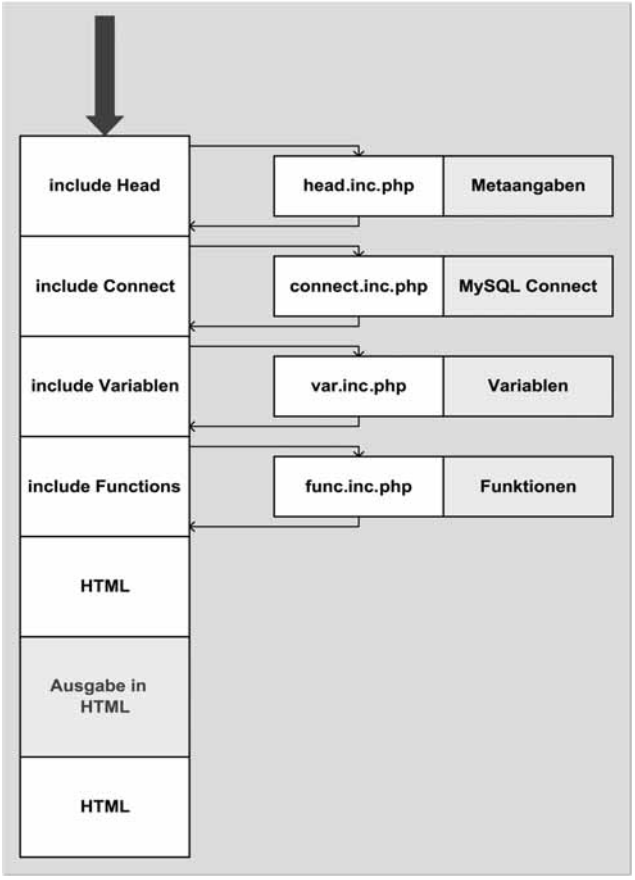


Abbildung 19.8 Externe Dateien einbinden

PHP unterstützt gleich mehrere Möglichkeiten, Skripte einzubinden. Die einzelnen Befehle unterscheiden sich durch die Art und Weise, wie die externen Skripte eingebunden werden.

Befehl	Bedeutung
`include(ZIEL)`	Bindet eine externe Datei ein und führt diese innerhalb des Skripts aus. Fehlt die Datei, gibt es eine PHP-Warnung, das Hauptskript wird aber dennoch ausgeführt.
`include_once(ZIEL)`	Bindet eine externe Datei ein und führt diese innerhalb des Skripts aus. Diese Schreibweise stellt sicher, dass externe Dateien nur einmal eingebunden werden. Fehlt die Datei, gibt es eine PHP-Warnung, das Hauptskript wird dennoch ausgeführt.
`require(ZIEL)`	Bindet eine externe Datei ein. Fehlt die Datei, gibt es einen Fatal Error, das Hauptskript wird nicht ausgeführt.
`require_once(ZIEL)`	Bindet eine externe Datei ein. Diese Schreibweise stellt sicher, dass externe Dateien nur einmal eingebunden werden. Fehlt die Datei, gibt es einen Fatal Error, das Hauptskript wird nicht ausgeführt.

Tabelle 19.4 Einbinden externer Dateien mit PHP

19.8.1 Mit MySQL verbinden

Vor jeder Abfrage muss eine Verbindung zu einer Datenbank aufgebaut werden. Für MySQL erfolgt das mit den folgenden Befehlen:

```
//  MySQL Servername
$dbhost = "HOST";
//  MySQL Benutzername
$dbuser = "USER";
//  MySQL Passwort
$dbpasswd = "PASS";
//  MySQL SERVER CONNECT
$db=mysql_connect("$dbhost","$dbuser","$dbpasswd");
```

Listing 19.11 Datenbankverbindung zu MySQL herstellen

In den Variablen `$dbhost`, `$dbuser` und `$dbpasswd` werden die Zugangsdaten für den Server an die PHP-Funktion `mysql_connect()` übergeben und an MySQL geschickt. Eine bestimmte Datenbank wird hier noch nicht ausgewählt.

19.8.2 MySQL-Datenbanken abfragen

Nach dem Herstellen einer Verbindung zu MySQL müssen Sie zunächst die abzufragende Datenbank auswählen:

```
//  MySQL Datenbankname
$db_name="DATENBANKNAME";
mysql_select_db($db_name,$db);
```

Listing 19.12 Datenbank auswählen

Erst jetzt ist es möglich, eine konkrete Abfrage an MySQL zu senden:

```
$abfrage = "SELECT * FROM TABELLE";
$ergebnis = mysql_query($abfrage);
   while($row = mysql_fetch_object($ergebnis))
      {
      echo "$row->headline"; // Ausgabe
      }
mysql_free_result($ergebnis);
```

Listing 19.13 Datenbank abfragen

In unserem Beispiel wird mit `$row` ein Objekt aus den von MySQL zurückgegebenen Daten erstellt und innerhalb einer `while`-Schleife ausgegeben. Die eigentliche SQL-Anweisung wird als String an MySQL geschickt.

19.9 Mit Sessions arbeiten

Innerhalb dynamischer Websites ist es oft wichtig, für jeden zugreifenden User bestimmte Informationen über die Gesamtzeit des Sitezugriffs aufrechtzuerhalten. Das können zum Beispiel ein Warenkorb, bestimmte Zugriffsrechte oder eine Auswertung des zurückgelegten Weges innerhalb der Website sein.

Mit den normalen technischen Möglichkeiten ist dies nur über Unmengen versteckter Felder in Formularen möglich. Das ist sehr unkomfortabel und fehleranfällig. Wenn Sie nicht wissen, wie viele Daten (Warenkorb) der User benötigt, ist es sogar unmöglich.

Variablen-Container

PHP und auch andere Skriptsprachen bieten daher die Möglichkeit, mit sogenannten *Sessions* zu arbeiten. Diese ermöglichen es, Variablen wie in einem Container abzulegen und jederzeit auf den Inhalt dieses Containers zuzugreifen.

Damit eine eindeutige Zuordnung der Daten zum User stattfinden kann, wird zu Beginn einer Session eine Session-ID wie etwa die folgende vergeben:

```
PHPSESSID=ca99cb395982a54a14b20aa2e49bae93
```

Session-ID übermitteln

Diese Session-ID wird automatisch generiert und identifiziert den User auch auf den nachfolgenden Seiten, die er anklickt, wenn dem Webserver die Session-ID bekannt ist. Damit das funktioniert, werden von PHP drei Verfahren angewandt, um Session-Daten zu übermitteln:

- als Cookie
- als Array (`$_SESSION[]`), wird per URL-Parameter übergeben
- als Datei auf dem Server

Je nach Voreinstellung wird eine dieser Methoden automatisch angewandt. Das Speichern einer Session auf dem Server wird allerdings von vielen Providern aus Sicherheitsgründen deaktiviert.

Session starten

Um mit Sessions zu arbeiten, müssen Sie die Session in jedem Dokument, das auf die gespeicherten Daten zugreifen soll, starten:

- `session_start()`; startet eine Session.
- `session_register(VARIABLE)`; registriert eine Variable in einer Session.
- `session_unset()` hebt die Registrierung aller Variablen auf.
- `session_destroy()` löscht die komplette Session.

19.10 Funktionen

Funktionen sind das A und O einer strukturierten, effektiven und zeitsparenden Programmierung. Sie sind zudem die notwendige Grundlage der objektorientierten Programmierung. Interessant werden Funktionen bei Dreamweaver, da in der aktuellen Version (CS5) erstmals eigene Funktionen im Code unterstützt werden.

Ohne Funktionen muss jeder Code, egal wie oft er innerhalb einer Website vorkommt, immer wieder neu geschrieben werden. Auch wirklich komplexe Operationen sind ohne Funktionen kaum möglich. Viele der PHP-Befehle sind ebenfalls Funktionen. Sehen wir uns das am Beispiel der PHP-Funktion `substr()` an.

```
string substr ( string $string , int $start [, int $length ] )
```

Gibt den Teil von `string` zurück, der durch die Parameter `start` und `length` definiert wurde. (*Quelle: www.php.net)*

`substr()` gibt Teile einer Zeichenkette zurück. Wir können die Funktion – so wie auch jede andere Funktion – an beliebiger Stelle und so oft wir wollen, im Code einsetzen. Nehmen wir an, wir haben `substr()` an einhundert Stellen in unserem Projekt verwendet. Sicherlich können Sie sich vorstellen, dass, wenn wir in der Funktion `substr()` etwas verändern, dies Auswirkungen auf alle hundert Stellen in unserem Code hat. Die Pflegbarkeit einer Website steigt enorm, ähnlich wie es beim Arbeiten mit externen CSS-Dateien der Fall ist.

Mit PHP kann man nun, zusätzlich zu den vordefinierten Funktionen, auch eigene Funktionen verwenden. Das ist denkbar einfach. Der Syntax lautet wie folgt:

```
function NameMeinerFunktion($Parameter1,$Parameter2) {
         CodeBlock;
         return Rückgabewert;
      }
```

Listing 19.14 Funktionsdeklaration

Wir wollen nun eine Funktion schreiben, die zwei beliebige Werte miteinander addiert:

```
<?php
function addiere($a,$b) {
   return $a+$b;
}
?>
```

Listing 19.15 Funktion zum Addieren

Der Befehl `return` ist notwendig, damit die Funktion überhaupt einen Wert zurückliefert.

Eine Funktion besteht immer aus zwei Teilen – zum einen aus der Deklaration, zum anderen aus dem Aufruf der Funktion in einem Skript.

```
<?php
function addiere($a,$b) {
   return $a+$b;
}
echo addiere(100,200);
?>
```

Listing 19.16 Skript mit Aufruf der Funktion »addiere«

Wie bereits erwähnt, unterstützt Dreamweaver mittlerweile eigene Funktionen. Wurde die Funktion korrekt deklariert, wird sie wie alle anderen PHP-Funktionen durch die Eingabe der ersten Buchstaben und Drücken von Strg + Leertaste angezeigt.

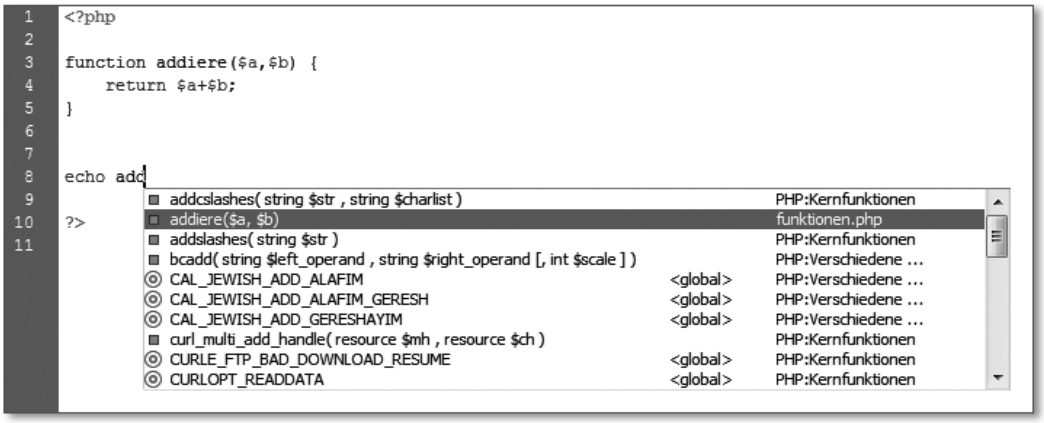


Abbildung 19.9 Eigene Funktion in Dreamweaver

19.11 Fazit

In der Hoffnung, Ihnen PHP nun ein wenig nähergebracht zu haben, verweisen wir hinsichtlich detaillierterer Informationen nochmals auf die einschlägigen Websites oder auf weiterführende Literatur. Wichtig für die Arbeit mit Dreamweaver in diesem Buch ist zunächst nur, dass Sie PHP in den Grundzügen verstehen.

Mit der aktuellen Dreamweaver-Version ist das Arbeiten mit PHP nochmals deutlich einfacher geworden. Leider vermissen wir nach wie vor eine echte Debug-Möglichkeit sowie Werkzeuge zum Kommentieren des Quelltextes. Hier hoffen wir auf die nächsten Versionen.

Appetit auf PHP bekommen? [«]

Über PHP gibt es massenhaft Literatur und unzählige Websites, allen voran die Projektwebsite *http://www.php.net* selbst. Dort finden Sie eine komplette Referenz inklusive ausführlicher Erklärungen. Die PHP-Dokumentation liegt auf dieser Website auch in einer deutschen Übersetzung vor. Wenn Sie sich für allerneueste Features interessieren, sollten Sie aber die englische Version lesen, da diese immer am aktuellsten ist.

Weitere Fundgruben sind *http://www.selfphp.com* im Format des unschlagbaren XHTML-Workshops SelfHTML und das Forum *http://www.phpcoders.de*.

PHP ohne Code zu schreiben? Dreamweaver macht es möglich. Wie das geht und wo die Grenzen sind, zeigen wir Ihnen in diesem Kapitel.

20 PHP mit Dreamweaver

Dreamweaver ist ein herausragendes Layoutwerkzeug. Für Webdesigner, die sich nur kurz mit PHP auseinandersetzen und dennoch einfachere dynamische Sites erstellen möchten, kennen wir definitiv nichts Besseres. Die Möglichkeiten, mit Dreamweaver PHP zu »schreiben«, ohne den Quellcode komplett zu verstehen, sind enorm.

Zu Beginn Ihrer Arbeit müssen Sie eine Entscheidung treffen: Wenn Sie mit den im Folgenden vorgestellten Möglichkeiten bei Ihrer Website zurechtkommen und diese für Ihr Projekt auch in weiterer Zukunft ganz sicher ausreichend sind, spricht nichts dagegen, die Arbeitserleichterung von Dreamweaver in vollem Umfang zu nutzen und den dynamischen Teil einer Website »im Layout« zu erstellen.

Ist jedoch absehbar, dass die zu erstellende Website längerfristig um Features erweitert werden muss, die nicht von Dreamweaver abgedeckt werden, empfiehlt es sich, Dreamweaver nur für das Layout zu nutzen und alle PHP-Anteile von Hand im Quelltext zu schreiben. Eine Anpassung des von Dreamweaver erstellten Codes ist nur mit hohem Aufwand möglich. Sobald Sie nämlich Änderungen in diesem Code vornehmen, erkennt Dreamweaver ihn nicht mehr als eigenen und verweigert die weitere Bearbeitung im Vorschaumodus.

Doch selbst wenn Sie sich voll und ganz auf die Unterstützung von Dreamweaver verlassen, ist ein Grundverständnis von PHP jedoch zwingend notwendig, um effektiv und fehlerfrei damit arbeiten zu können. Lesen Sie daher bitte auch die Kapitel 19, »Einführung in PHP«, und 17, »Quelltext de luxe«.

20.1 PHP-Befehle einsetzen

Wenn Sie mit PHP-Dokumenten arbeiten, steht Ihnen die Symbolleiste PHP zur Verfügung. Wir haben diese Palette in erster Linie der Vollständigkeit halber in das Buch mit aufgenommen. Wer direkt im Quelltext arbeiten kann, sollte diese

Palette besser nicht verwenden. Dreamweaver fügt nämlich grundsätzlich um jede Variable und alles, was Sie über diese Einfügepalette anlegen, stets das PHP-Tag mit ein.

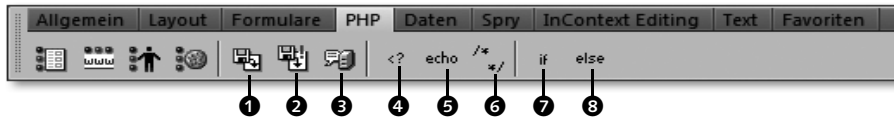


Abbildung 20.1 Einfügemenü für PHP

Beim Klick auf die Menüeinträge erzeugt Dreamweaver im Dokument die in Tabelle 20.1 dargestellten PHP-Tags.

PHP-Tag	Nr.	Bedeutung
`<?php include(); ?>`	❶	Include
`<?php require(); ?>`	❷	Erfordern
`mb_http_input("");` `mb_http_output("");`	❸	PHP Page Encoding
`<?php ?>`	❹	Codeblock
`<?php echo ?>`	❺	Echo
`/* */`	❻	Kommentar
`<?php if ?>`	❼	IF
`<?php else ?>`	❽	Else

Tabelle 20.1 PHP-Funktionen in Dreamweaver

Leider lassen sich die PHP-Befehle nicht als Tag definieren; Dreamweaver erkennt nur in HTML gültige Tags mit spitzen Klammern.

20.2 Variablen in Dreamweaver

20.2.1 Mit Variablen arbeiten

Im Bedienfeld Bindungen (siehe Abbildung 20.2) können Sie über das Plus-Symbol Variablen definieren.

Eine neu definierte Variable steht Ihnen auf der gesamten Site in jedem Dokument zur Verfügung. Sie können an beliebiger Stelle mit ihr arbeiten, indem Sie die Variable aus dem Bedienfeld Bindungen einfach in das Dokument ziehen oder sich bei Abfragen darauf beziehen. Leider legt Dreamweaver beim Einfügen in Dokumente grundsätzlich das PHP-Tag mit an, egal ob es an der Stelle gewünscht ist oder nicht.

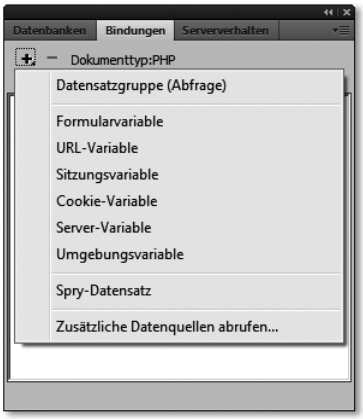


Abbildung 20.2 Neue Variablen definieren und bereitstellen

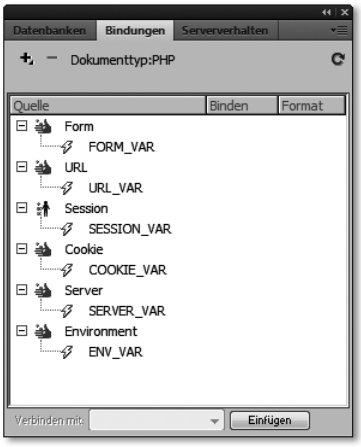


Abbildung 20.3 Variablen im Bedienfeld »Bindungen«

Dreamweaver verwendet die in Tabelle 20.2 dargestellten Variablentypen.

Variable	Nr.	Ausgabe mit PHP
Formularvariable	❾	`<?php echo $_POST[VAR]; ?>`
URL-Variable	❿	`<?php echo $_GET[VAR]; ?>`
Sitzungsvariable	⓫	`<?php echo $_SESSION[VAR]; ?>`
Cookie-Variable	⓬	`<?php echo $_COOKIE[VAR]; ?>`
Servervariable	–	`<?php echo $_SERVER[VAR]; ?>`
Umgebungsvariable	–	`<?php echo $_ENV[VAR]; ?>`

Tabelle 20.2 Variablentypen in Dreamweaver



Abbildung 20.4 Variablen einfügen

Es ist sicherer, wenn Sie für unterschiedliche Variablentypen auch unterschiedliche Bezeichnungen verwenden. (`VAR` ist der Teil, den Sie umbenennen können.) Wenn Sie sowohl eine Formularvariable als auch eine URL-Variable mit `ID` bezeichnen, können Sie in Dreamweaver nicht sicher sein, ob auch die richtige verwendet wird. Hierfür fehlt schlicht die vollständige Kontrolle über den Quellcode.

[!]

Keine nachträgliche Bearbeitung möglich

Haben Sie eine Variable einmal festgelegt, ist eine nachträgliche Bearbeitung nicht mehr möglich. Sie müssen die Variable dann löschen und neu definieren. Alle Bezüge in Ihren Dokumenten gehen ebenso verloren. Überlegen Sie also genau, welche Variablen Sie benötigen, und legen Sie sie erst dann an.

20.2.2 Variablen senden

Während der Arbeit an dynamischen Seiten müssen immer wieder Variablen an einzelne Dokumente übergeben werden, damit diese Dokumente die durch die Variable angeforderten Daten generieren.

Um in Dreamweaver einzelne Dokumente testen zu können, ohne alle zusätzlichen Skripte zu definieren, besteht die Möglichkeit, diese Variablen in der Entwicklungsumgebung zu deklarieren und eine Übertragung zu simulieren. Öffnen Sie dazu die Browser-Navigation über Ansicht • Symbolleisten • Browser-Navigation.

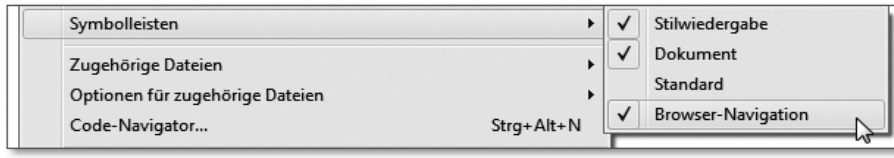


Abbildung 20.5 Öffnen der Browser-Navigation

Zum Anlegen der Variablen, die übertragen werden sollen, öffnen Sie die Dropdown-Liste neben dem Icon ❶.

Wie Sie sehen, ist das aktuelle Dokument in der Browserleiste ❷ eingetragen, hier bereits mit angehängter Variablen. Die Variablen selbst können durch Klicken auf Einstellungen für HTTP-Anforderungen ❸ deklariert werden. Es erscheint dann die folgende Dialogbox, in der Sie Ihre Eintragungen vornehmen können.

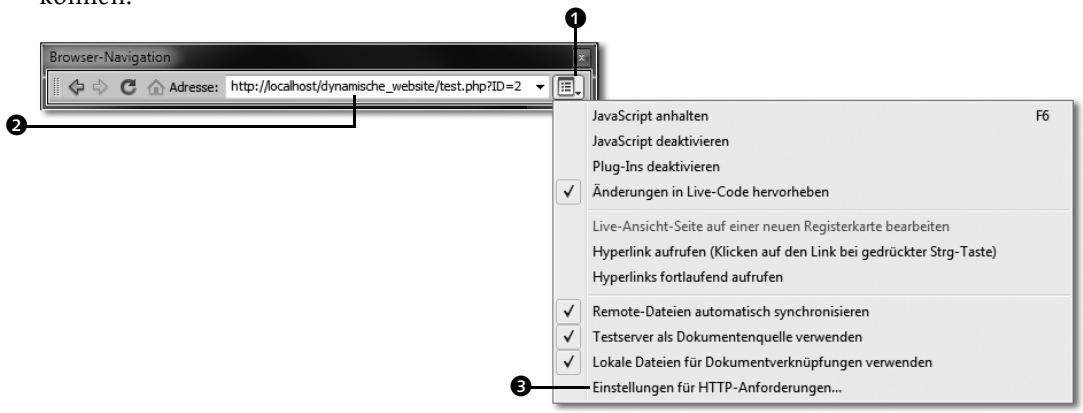


Abbildung 20.6 Browser-Navigation mit http-Anforderung

Das Hinzufügen neuer Variablen erfolgt – wie in Dreamweaver gewohnt – über die Plus- und Minuszeichen ❹. Unter Name ❺ geben Sie den Variablennamen ein. Dabei darf kein Dollarzeichen oder die gesamte Syntax des Arrays eingegeben werden, sondern nur der eigentliche Name ohne weitere Bezeichnungen. Rechts daneben wird unter Wert ❼ der Variableninhalt bestimmt.

Wichtig ist noch das Einstellen der Methode ❻, mit der die Variablen an Ihr Skript gesendet werden. Bei URL-Variablen wird das immer GET, bei Variablen aus Formularen meist POST sein.

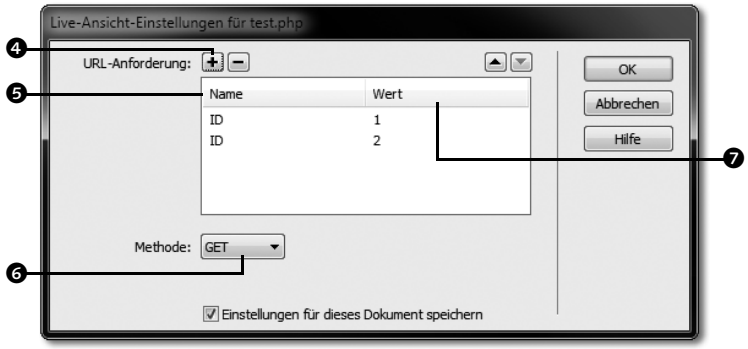


Abbildung 20.7 Deklarieren der Variablen für die Live-Ansicht

[»] **Erst im Layout, dann im Code arbeiten**

Dreamweaver kann Ihnen beim Arbeiten mit dynamischen Websites eine Menge Mühe ersparen. Ein großer Teil der Arbeit kann direkt im Layout erfolgen, und für einfache dynamische Websites ist dies auch ausreichend. Den letzten Feinschliff sollten Sie jedoch im Quelltext vornehmen. Zeichenlängen, Rechen- und Vergleichsoperationen usw. können nur dort eingerichtet werden.

Wir erstellen Websites zunächst gestalterisch im Layoutmodus und wechseln dann in die Codeansicht, um die von Dreamweaver eingefügten Befehle zu modifizieren, oder verwenden gleich eigene Funktionen. Die Arbeiten an dynamischen Websites sind durch Dreamweaver wesentlich schneller und effizienter geworden.

20.3 PHP von Hand coden

Während Sie einen PHP-Befehl eingeben, werden die verfügbaren Attribute in einem kleinen Fenster über dem Cursor eingeblendet.

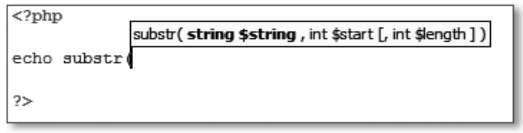


Abbildung 20.8 Referenzanzeige während der Eingabe eines Befehls

Nach der Eingabe der ersten Zeichen einer Funktion können Sie sich durch gleichzeitiges Drücken von Strg+Leertaste eine Auswahl der Befehle anzeigen lassen und mit den Pfeiltasten auf Ihrer Tastatur gleich den richtigen Befehl auswählen (siehe Abbildung 20.9).

20.4 PHP-Referenz

Über den Menüpfad Fenster • Ergebnisse • Referenz steht Ihnen in Dreamweaver für alle unterstützten Programmiersprachen eine Kurzreferenz zur Verfügung (siehe Abbildung 20.10). Sie eignet sich hervorragend, um die Attribute eines PHP-Befehls zügig nachzuschlagen.

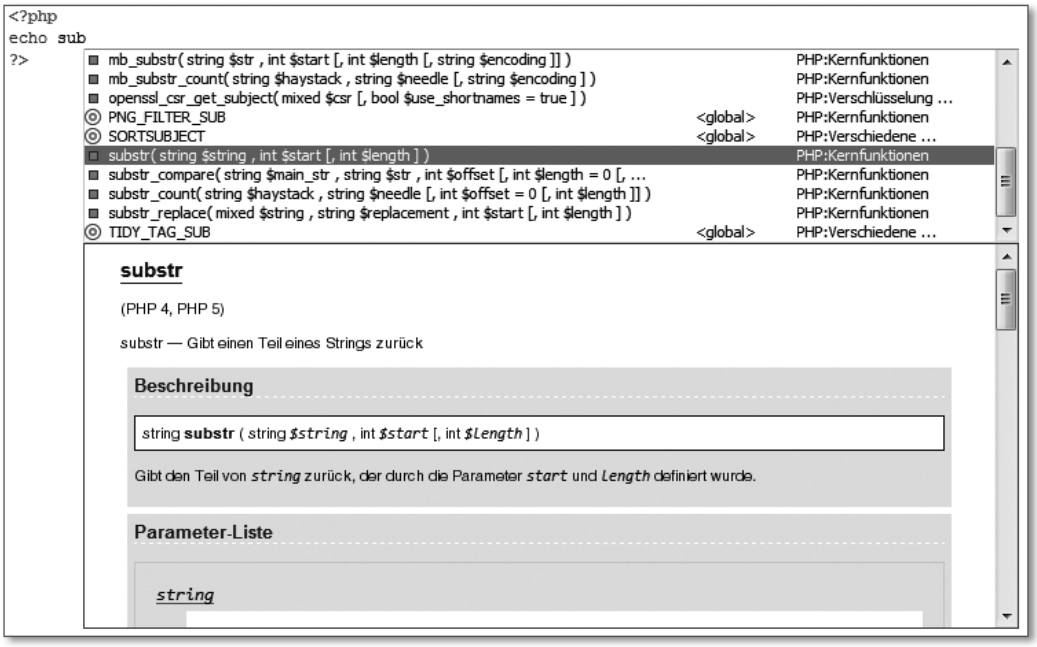


Abbildung 20.9 Befehlsliste anzeigen lassen

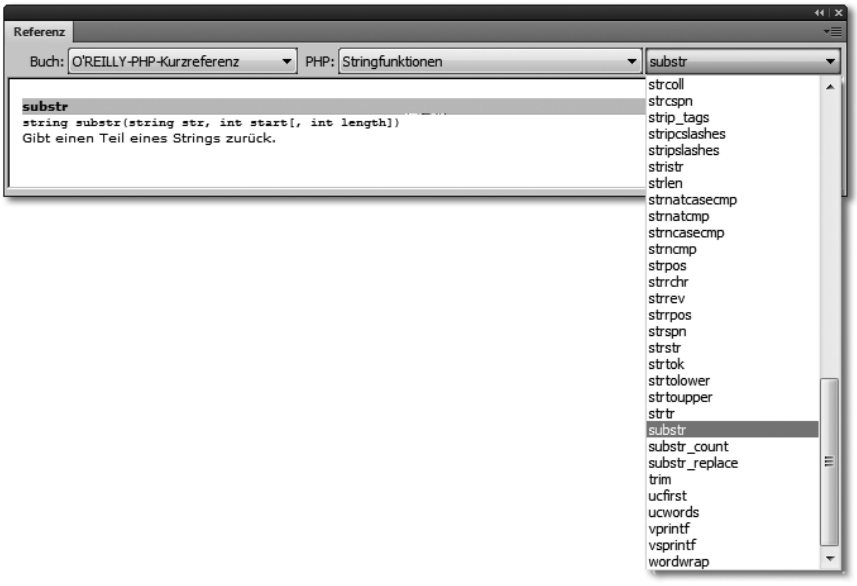


Abbildung 20.10 Die PHP-Referenz in Dreamweaver

20.5 Fehlersuche im PHP-Code

Während des Entwickelns sollte man auch diverse Hilfen zur Fehlersuche parat haben. Besonders wichtig wird dies, wenn Sie eigenen Code schreiben, der nicht in den Serververhalten usw. von Dreamweaver aufgeführt ist.

20.5.1 Fehlermeldungen anzeigen

Die Ausgabe aller PHP-Fehlermeldungen erreichen Sie durch den Befehl `error_reporting(E_ALL)`. Wenn Sie diese Befehlszeile an den Anfang Ihrer Dokumente setzen, werden beim Ausführen des Skripts alle auftretenden Fehlermeldungen und Warnungen ausgegeben. Bei allen Vorteilen, die diese Funktion bei der Entwicklung bietet, kann aber die Fülle der Meldungen die Arbeit auch behindern. Möchten Sie den Befehl weiter definieren, finden Sie dafür auf der PHP-Website *http://www.php.net* eine Funktionsreferenz mit weiteren Parametern.

20.5.2 Externe Variablen anzeigen

Die Anzeige aller Variablen, die von anderen Dokumenten übergeben wurden, also die Arrays `GET`, `POST` und `SESSION`, können Sie mit einer `foreach`-Schleife ausgeben lassen:

```
foreach ($_SESSION as $key =>$value)
     {
   echo "Session:"."$key"."-"."$value<br>";
   }
```

Listing 20.1 Schleife zur Ausgabe des `SESSION`-Arrays

Schreiben Sie für jedes der Arrays eine Schleife, und erstellen Sie eine `include`-Datei mit allen Fehler- und Variablenausgaben. Wenn Sie diese während der Entwicklung oder Fehlersuche in Ihre Website einbinden, werden Ihnen so immer alle Fehlermeldungen und übertragenen Variablen angezeigt. Im Anschluss können Sie die Befehlszeilen zum Einbinden dieser Datei einfach auskommentieren.

Eine weitere Möglichkeit besteht darin, die »Fehlersuchdatei« immer einzubinden und im Dokument eine Variable zur Fehlersuche zu setzen. Die dafür einzubindende Datei hat den folgenden Inhalt:

```
<?
if($debug_modus == 1)
// Alle Fehlermeldungen und Warnungen einschalten
error_reporting(E_ALL)
{
```

```
    // SESSION-Array ausgeben
foreach ($_SESSION as $key =>$value)
    {
         echo "Session:"."$key"."-"."$value<br>";
    }
    // POST-Array ausgeben
foreach ($_POST as $key =>$value)
    {
         echo "Post:"."$key"."-"."$value<br>";
    }
// GET-Array ausgeben
foreach ($_GET as $key =>$value)
    {
         echo "Get:"."$key"."-"."$value<br>";
    }
}
?>
```

Listing 20.2 Datei zur Fehlersuche

In den einzelnen Dokumenten setzen Sie als Erstes eine Variable `<? $debug_modus = 0 ?>` zum Ausschalten oder `<? $debug_modus = 1 ?>` zum Einschalten der Fehlersuche.

20.5.3 Häufige PHP-Fehlerquellen

Die am häufigsten auftretenden Fehler sind ganz einfacher Art, wobei sich die meisten den folgenden drei Gruppen zuordnen lassen:

- **Falsche Variablennamen**
  Achten Sie peinlichst genau auf Groß- und Kleinschreibung. PHP kennt kein Pardon. A und a sind zwei verschiedene Zeichen.
- **Falsche Anführungszeichen**
  Strings und Zeichenketten gehören in Anführungszeichen, Zahlen nicht. Wenn Sie eine Zahl in Anführungszeichen setzen, um sie auszugeben, wird die Zahl zu einem String, und Sie können damit keine eindeutigen Berechnungen mehr durchführen.
- **Kommata statt Punkten in Berechnungen**
  PHP benötigt als Dezimaltrennzeichen einen Punkt. Das Fatale an diesen Fehlern ist, dass man sie häufig erst spät bemerkt, da PHP hemmungslos weiterrechnet, aber alle Nachkommastellen ignoriert.

Zeilennummern einsetzen

Die Fehlermeldungen von PHP geben gute Hinweise auf bestehende Probleme. Dabei werden grundsätzlich die Zeilennummern mit angegeben, um die Fehler schneller finden zu können.

Bei MySQL-Abfragen stimmen diese Zeilennummern oft nicht. Ein Fehler in der Abfrage tritt häufig erst bei der Ausgabe der Daten auf. Überprüfen Sie in diesem Fall die gesamte Abfrage und nicht nur die angegebenen fehlerhaften Zeilen.

Wiederholtes Testen

Achten Sie einfach auf die genaue Schreibweise aller PHP-Elemente, und testen Sie nach jedem Befehl die Funktion. Ein Skript zu schreiben, das auf eine DIN-A4-Seite passt, und es erst im Anschluss zu testen führt meistens zur Verzweiflung. Mit Dreamweaver haben Sie hervorragende Möglichkeiten, die Syntax durch die farblichen Hervorhebungen bereits während der Eingabe zu prüfen und fertige, getestete Codefragmente immer wieder zu verwenden. Nutzen Sie diese Möglichkeiten, und Sie vermeiden viele Fehler.

Um schnell und effizient mit Dreamweaver und dynamischen Websites arbeiten zu können, ist es sinnvoll, einen lokalen Webserver einzurichten. In diesem Kapitel zeigen wir Ihnen an zwei Beispielen, wie das geht.

21 Lokalen Webserver installieren

Da dynamische Websites erst von einem Webserver erstellt werden, benötigen Sie zu Entwicklungszwecken einen lokalen Webserver, auf dem Sie Ihre Skripte testen können.

21.1 Webserver-Grundlagen

Jede Website benötigt einen Webserver, auf dem sie läuft. Hier erläutern wir erst einmal kurz die wichtigsten Begriffe zu diesem Vorgang.

Der Webserver ist dafür verantwortlich, Daten mit dem HTTP-Protokoll zu versenden. Mehr macht er eigentlich nicht. Er wartet auf eine Anforderung, sammelt Daten ein und schickt sie ab.

Für den Einsatz mit Dreamweaver und PHP kommen folgende Webserver in Frage:

- **Apache**
  Dies ist der definitiv meistgenutzte Webserver überhaupt. Eigentlich aus der Welt von UNIX kommend, laufen die neueren Versionen auch auf dem PC sehr stabil. Auf dem neuen Betriebssystem Mac OS X ist Apache bereits vorinstalliert. Die wesentlichen Vorteile sind, dass er sehr schnell, klein und leistungsfähig ist und unter Linux äußerst betriebs- und datensicher läuft.
- **Internet Information Server (IIS)**
  Der IIS ist der Webserver aus dem Hause Microsoft. Bei den Windows-Versionen XP, Vista und 7 wird er in der Professional Edition mitgeliefert, muss allerdings nachinstalliert werden. Unter Windows 7 kommt der IIS 7.5 zum Einsatz.

  Der IIS ist ein kompletter, hochprofessioneller Webserver und für den Einsatz im Web auf Microsoft-Systemen konzipiert. Einer der wichtigsten Vorteile ist,

dass er die Entwicklung mit ASP und ASP.NET ermöglicht. Er stellt auch viele Dienste wie Mail und FTP zur Verfügung. Um den IIS gut und sicher zu konfigurieren, muss man sich allerdings gut mit den Windows-Systemen auskennen.


Buchwebsite

Aufgrund seiner hohen Verbreitung verwenden wir den kostenlos verfügbaren Webserver Apache. Natürlich können Sie auch jeden anderen Webserver betreiben. Für das weitere Vorgehen im Buch wird lediglich vorausgesetzt, dass PHP und MySQL auf ihm installiert und lauffähig sind.

[»] **IIS oder Apache**

Die in den folgenden Abschnitten behandelten Webserver stellen Alternativen dar. Bei Ihrer Arbeit müssen Sie sich für den IIS oder den Apache entscheiden. Lesen Sie am besten vor der Installation dieses Kapitel, und entscheiden Sie dann, welches System Ihnen am verständlichsten erscheint.

Bei einigen der einfachen Windows-Versionen (HOME etc.) ist der IIS nicht im Lieferumfang enthalten, kann allerdings von der Microsoft-Website heruntergeladen werden.

LAMP und WAMP

Häufig fallen im Zusammenhang mit der Webentwicklung die Begriffe *LAMP* und *WAMP*. Sie bezeichnen Kombinationen verschiedener Systeme auf einem Rechner, bestehend aus Betriebssystem, Webserver, Datenbank und Skriptsprache. LAMP ist die Abkürzung für Linux, Apache, MySQL und PHP. Bei WAMP ist nur das Betriebssystem ein anderes: Statt Linux wird hier Windows eingesetzt. Eine Implementation von LAMP und WAMP ist unter dem Namen XAMPP bekannt, die wir im Folgenden auch für unseren Testserver einsetzen werden.

WAMP-Entwicklungsserver

Wir werden in diesem Buch ein WAMP-System installieren und dieses als Testserver betreiben. Bei Ihrem Provider steht wahrscheinlich ein LAMP-System. Das Internet ist eine Linux- und UNIX-Welt. Diese Systeme sind wesentlich zuverlässiger als Windows. Für unsere Arbeit spielt es allerdings kaum eine Rolle, ob wir auf WAMP oder LAMP entwickeln. Die Technologie dahinter ist für das Funktionieren einer Site ohne Bedeutung, solange der Webserver stabil läuft.

Es ist wichtig zu wissen, dass Apache ursprünglich nicht für Windows-Systeme geschrieben wurde, sondern für Linux. Er funktioniert zwar auch auf Windows-Servern, für reale Bedingungen im Web ist diese Kombination aber nicht ideal. Als Entwicklungsserver ist WAMP aber vollkommen ausreichend.

PHP, MySQL und Apache haben zunächst nichts miteinander zu tun. Nur durch unsere Programmierung nutzen sie sich gegenseitig. PHP und MySQL können völlig unabhängig voneinander laufen. Üblicherweise startet man allerdings die Dienste gemeinsam.

Webserver auf dem MAC [«]

Apache, PHP und MySQL laufen auch auf einem Mac, auf den neueren Systemen von Apple sogar hervorragend. Lesen Sie dazu die Installationsanleitungen auf *http://www.apple.de* oder auf den Websites von Apache, PHP und MySQL. Es gibt auch einige Foren, die sich mit dieser Thematik befassen. Bei Mac OS X ist standardmäßig bereits Apache usw. auf dem System installiert.

21.2 Vorbereitung: Firewall einstellen oder ausschalten

Nun genug der langen Rede. Im Folgenden installieren wir einen lokalen Testserver und sehen uns die Sache am lebenden Objekt an.

Auf der dem Buch beigelegten DVD finden Sie alle notwendigen Programme zur Installation im Verzeichnis *Webserver*. Wir beschreiben hier die Installation von Apache und eines Internet Information Systems unter Windows XP sowie des IIS 7 unter Vista Business. [◉]

MAMP [«]

Für den Mac steht auch ein Paket zur Verfügung: Es nennt sich MAMP, Sie finden es ebenfalls auf der DVD. Weitere Informationen gibt es unter *http://www.mamp.info*. XAMPP läuft unter Mac nur mit einer Intel-CPU.

Desktop-Firewall konfigurieren

Wenn Sie auf dem Rechner für den Testserver eine Desktop-Firewall installiert haben, kann es zu Problemen mit einem Webserver kommen. Falls nach den folgenden Schritten einer der Webserver wider Erwarten nicht funktioniert, überprüfen Sie die Einstellungen Ihrer Firewall. Vereinfacht ausgedrückt sprechen Sie Ihren Rechner beim Webserverbetrieb in einer Art Rückkopplung an. Für eine Firewall sieht das so aus, als würde ein fremder Rechner einen Zugriffsversuch starten. Falls möglich, betreiben Sie Ihren Rechner an einem Router, und aktivieren Sie auf diesem die Firewall mit den gewünschten Einstellungen. Oder noch besser: Schalten Sie einfach einen Rechner mit Linux als Firewall und Router dazwischen. Die geschilderten Probleme treten dann nicht auf.

[»] **Probleme von Apache und IIS mit der Firewall Outpost**

Schwerwiegende Probleme traten bei uns mehrfach mit der freien Version der Firewall Outpost von Agnitum auf. Die Zugriffsrechte lassen sich zwar einfach konfigurieren, jedoch bringen sowohl der Apache als auch der IIS in Kombination mit dieser ansonsten sehr guten Desktop-Firewall das System zum Totalabsturz. Das Problem tritt bei uns grundsätzlich in Kombination mit Hyper-Trading-Prozessoren auf. Nach einigen Zugriffen auf die lokale Site startet der Rechner einfach neu.

Nachdem wir anfänglich der Meinung waren, dass ein Virus daran schuld ist, sind wir mittlerweile sicher, dass es an Outpost liegt. Falls Ihnen dieses Problem bekannt ist und Sie eine Lösung kennen, lassen Sie es uns wissen.

Interne Firewall

Windows XP, Vista und 7 benutzen eine interne Firewall. Auch hier müssen Sie je nach Systemkonfiguration Einstellungen vornehmen oder die Firewall deaktivieren.

Weitere Vorbereitungen für die Installation eines Webservers sind nicht erforderlich. Für den IIS benötigen Sie nur den Original-Datenträger von Windows XP, Vista oder 7. Ohne die Original-CD oder -DVD ist eine Installation nicht möglich.

21.3 WAMP installieren

Einen vollwertigen und sicheren Webserver für den Produktivbetrieb zu installieren ist keineswegs trivial. Für unser Testsystem wählen wir daher den einfachen Weg und verwenden die bereits erwähnte Instant-Installation eines WAMP-Systems namens XAMPP.

[!] **Nicht als realer Webserver geeignet**

Beachten Sie bitte, dass keine der von uns vorgeschlagenen Konfigurationen zum Betrieb eines »echten« Webservers im Internet geeignet ist. Die Systeme sind ausschließlich zum lokalen Gebrauch bei der Entwicklungsarbeit gedacht.

Komplettes Installationspaket

XAMPP ist ein Komplettsystem mit allen erdenklichen Modulen und Bibliotheken. Auf der Website *http://www.apachefriends.org* finden Sie neben einer Downloadmöglichkeit eine vollständige Dokumentation sowie einige Erweiterungen für dieses Tool.

XAMPP ist mittlerweile in der Version 1.7.3 verfügbar (Stand Juni 2010). Diese Version nutzt die neueste MySQL-Version 5.1. MySQL ab der Version 5 liegt in einer Community-Version vor, die auch für private und Open-Source-Projekte kostenlos nutzbar ist.

XAMPP installieren

Im Verzeichnis *Webserver/XAMPP* auf der DVD zum Buch finden Sie die Datei *xampp-win32-1.7.3.exe*. Kopieren Sie diese Datei auf Ihren Desktop, und starten Sie die Installationsroutine durch einen Doppelklick. [o]

XAMPP unter Windows 7

XAMPP funktioniert in der von uns verwendeten Version auch unter Windows 7 hervorragend, wenn man einige Kleinigkeiten beachtet. Während der Installation kommen einige Sicherheitsmeldungen, die Sie ignorieren können. XAMPP wird standardmäßig im Verzeichnis *C:/XAMPP/* installiert, Sie können aber auch ein eigenes Verzeichnis angeben und sollten das unter Windows 7 auch machen.

In unseren Beispielen installieren wir XAMPP auf dem Laufwerk *C:*

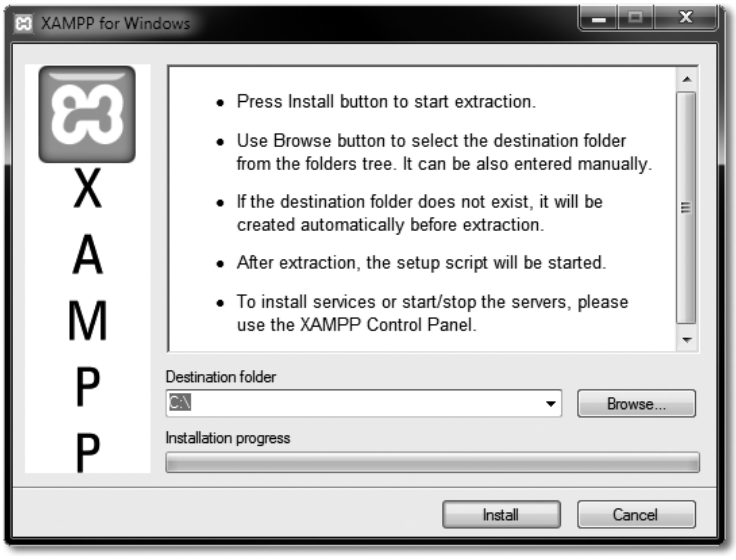


Abbildung 21.1 Installationsverzeichnis unter Windows 7

Zum Abschluss der Installation müssen Sie in einigen Dialogen die jeweiligen Optionen mit *y* oder *n* bestätigen. Wir empfehlen Ihnen, hier immer mit den Vorgaben zu arbeiten.

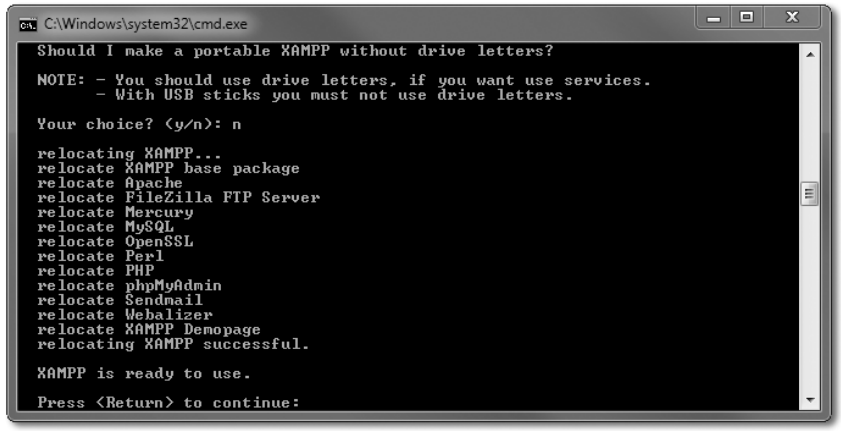


Abbildung 21.2 Abschluss der Installation

Nach der Installation finden Sie unter Alle Programme • XAMPP for Windows die in Abbildung 21.3 gezeigten Einträge.

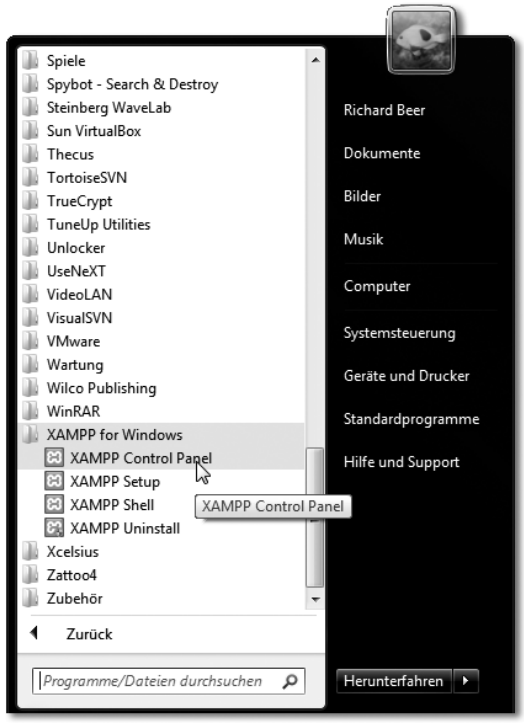


Abbildung 21.3 Startmenü für XAMPP

21.3.1 XAMPP testen und konfigurieren

Klicken Sie auf PROGRAMME • XAMPP FOR WINDOWS • XAMPP CONTROL PANEL.

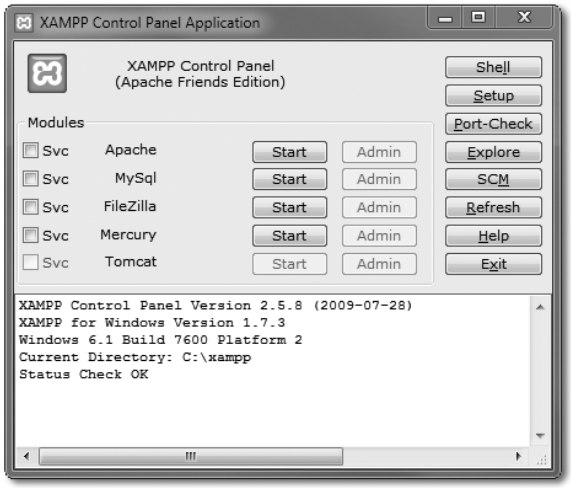


Abbildung 21.4 XAMPP Control Panel

Im folgenden Fenster (siehe Abbildung 21.4) werden die einzelnen Dienste gestartet und auch wieder beendet. Klicken Sie zum Testen von XAMPP bei *Apache* und *MySql* auf START.

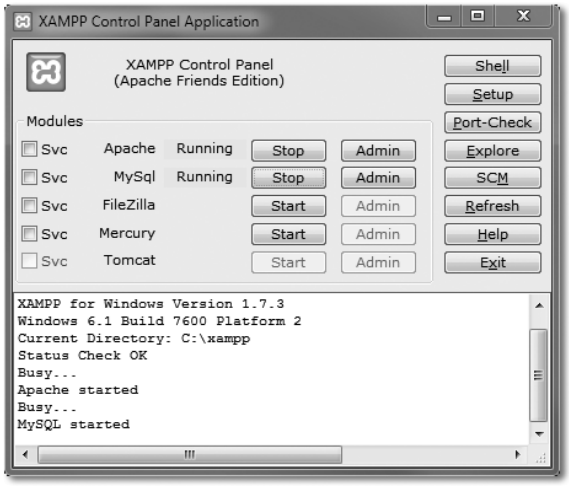


Abbildung 21.5 XAMPP Control Panel mit gestarteten Apache und MySql

Öffnen Sie einen Browser, um die Funktionen zu überprüfen. Der lokale Webserver hat den Namen *localhost* und die IP 127.0.0.1.

Geben Sie *http://localhost* in die Adressleiste des Browsers ein.

Wenn jetzt – nach einer eventuellen Sprachauswahl – der Bildschirm aus Abbildung 21.6 erscheint, war Ihre Installation erfolgreich.

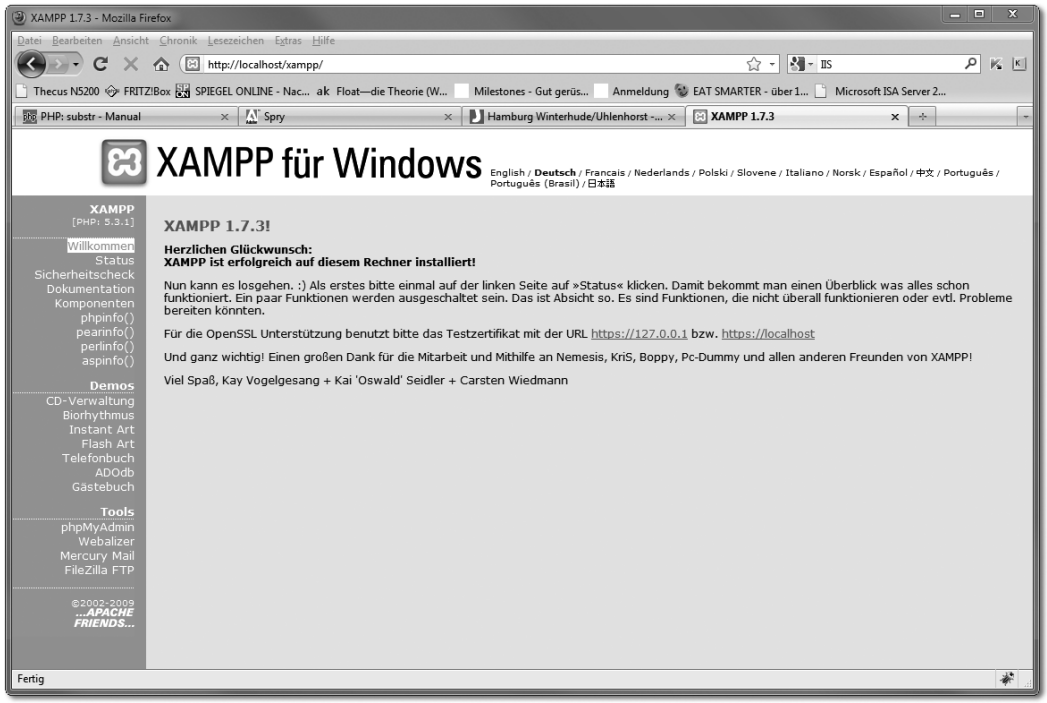


Abbildung 21.6 Startbildschirm von XAMPP unter http://localhost/xampp

Ablageort für dynamische Websites

Betrachten wir nun die Verzeichnisse auf Ihrer Festplatte. Auf Laufwerk *C:* befindet sich ein neuer Ordner mit dem Namen *xampp/htdocs/*. Dieser Ordner ist die sogenannte *Documentroot*.

In diesen Ordner müssen Sie alle dynamischen Websites ablegen, die Sie auf Ihrem System entwickeln möchten. Nur dort werden die Dokumente ausgeführt. Zunächst liegen hier jedoch die Dokumente für die XAMPP-Website und einige Testdokumente.

Löschen Sie den kompletten Ordnerinhalt von *htdocs*, und geben Sie im Browserfenster *http://localhost* ein. Sie sehen jetzt die Rootverzeichnisstruktur im Browserfenster. Da noch keine Verzeichnisse angelegt wurden, sieht das Browserfenster zunächst aus, wie in Abbildung 21.8 dargestellt. Werden Ihnen andere, fehlerhafte Inhalte angezeigt, müssen Sie eventuell XAMPP neu starten.

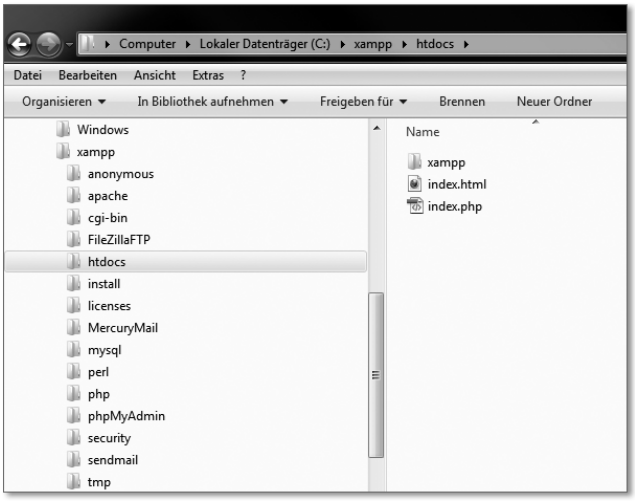


Abbildung 21.7 Inhalt des Ordners »htdocs«

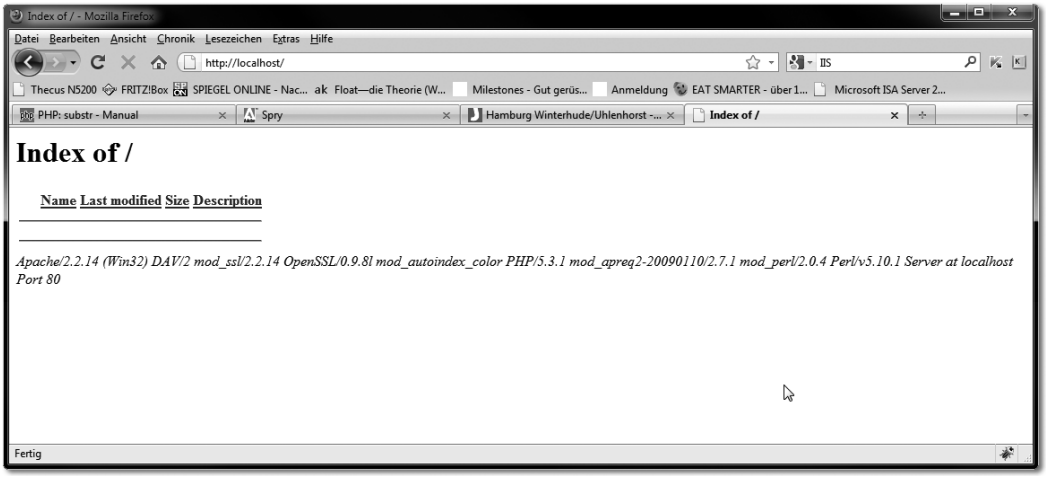


Abbildung 21.8 Anzeige im Browser bei Serverroot ohne Dokumente

21.3.2 Eigene Documentroot angeben

Sie müssen nicht zwingend mit der von XAMPP voreingestellten Serverroot arbeiten. Die Datenhaltung für wichtige Websites auf Laufwerk *C:* unter Windows ist sehr kritisch. Bei einer Neuinstallation von Windows wären alle vorhandenen Daten verloren. Aus diesem Grund empfiehlt es sich, die Serverroot auf eine eigens dafür angelegte Partition zu legen. XAMPP löscht bei der Deinstallation die Documentroot, wenn Sie diese im Originalverzeichnis belassen. Auch das ist ein Grund, die Documentroot auf ein anderes Laufwerk zu legen.

Im weiteren Verlauf des Buches arbeiten wir, wie in der vorherigen Installation beschrieben. Im folgenden Beispiel zeigen wir Ihnen, wie Sie das Laufwerk *E:* als Documentroot angeben. Das kann je nach Konfiguration Ihres Rechners auch ein anderes Laufwerk sein.

[!] **Sicherungskopie anlegen**

Legen Sie, bevor Sie eine eigene Documentroot definieren, unbedingt eine Sicherungskopie der zu verändernden Dateien an. So können Sie bei einem Misserfolg die Originaleinstellungen wiederherstellen.

[▪] *Schritt-für-Schritt: Eigenes Rootverzeichnis angeben*

1 *Konfigurationsdatei öffnen*

Öffnen Sie mit einem Texteditor die Datei *httpd.conf* aus dem Ordner *C://xampp/apache/conf/*, und suchen Sie die folgenden Zeilen:

```
# DocumentRoot: The directory out of which you will serve your
# documents. By default, all requests are taken from this directory,
# but symbolic links and aliases may be used to point to other
# locations.
#
DocumentRoot "C:/xampp/htdocs"
```

2 *DocumentRoot ändern*

Ändern Sie die DocumentRoot auf Ihr gewünschtes Verzeichnis. Bei uns ist die DocumentRoot auf

```
DocumentRoot "E:/"
```

eingestellt.

Suchen Sie einige Zeilen weiter den Eintrag

```
# This should be changed to whatever you set DocumentRoot to.
#
<Directory "C:/Programme/xampp/htdocs">
```

Hier müssen Sie unter `Directory` den gleichen Eintrag wiederholen. Wir stellen auch diesen auf

```
Directory "E:/"
```

3 *Server neu starten*

Starten Sie jetzt mit XAMPP den Apache neu. ■

21.3.3 phpMyAdmin in die Documentroot legen

phpMyAdmin ist eine Benutzeroberfläche für die Administration von MySQL. Damit Sie ohne die von uns gelöschte XAMPP-Website Zugriff auf phpMyAdmin haben, müssen Sie das gesamte Verzeichnis *c:/xampp/phpmyadmin/* in die von Ihnen gewünschte Documentroot verschieben. In unserem Beispiel ist es das Laufwerk *E:*

Browser starten

Nachdem Sie dies getan haben, starten Sie den Browser. In der Liste müssten Sie jetzt den Eintrag phpMyAdmin sehen. Klicken Sie auf den Eintrag, dann sehen Sie die Oberfläche von phpMyAdmin. Unter Windows 7 kann es vorkommen, dass die Verzeichnisse im Browser nicht angezeigt werden. Geben Sie dann *http://localhost/phpMyAdmin/* direkt in die Adressleiste Ihres Browsers ein, um phpMyAdmin zu starten.

Falls phpMyAdmin nicht aufgerufen wird, überprüfen Sie alle Schritte, und achten Sie darauf, dass Apache und MySQL gestartet sind.

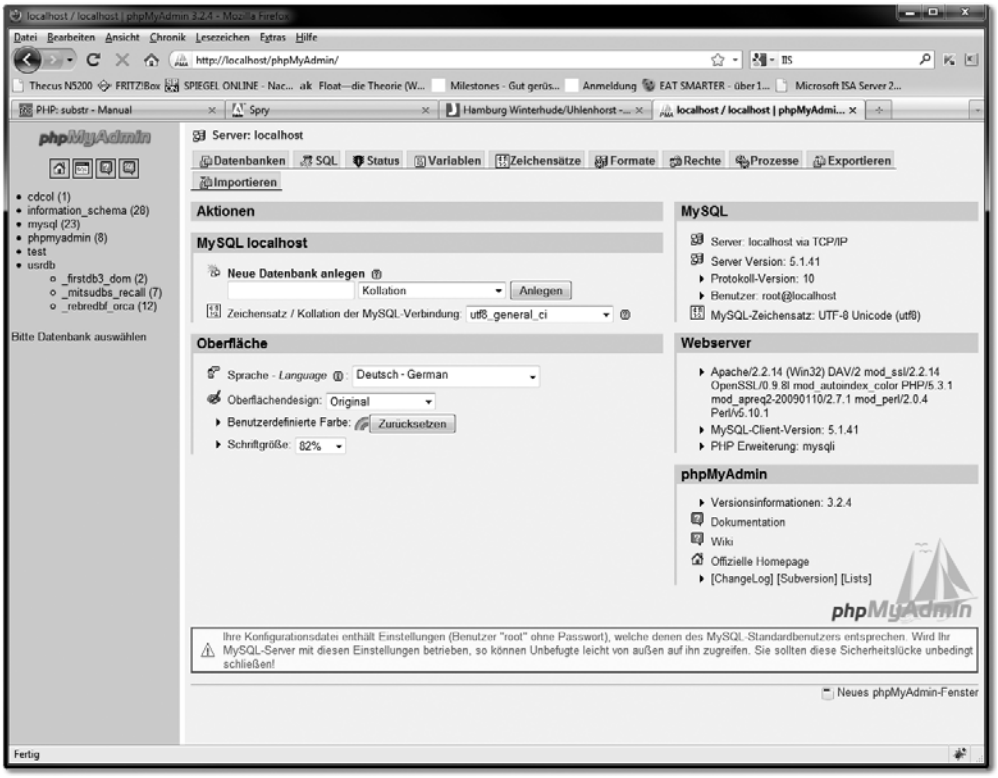


Abbildung 21.9 Die phpMyAdmin-Oberfläche

[!] **Vorsicht beim Deinstallieren**

Sichern Sie unbedingt alle Dokumente und Datenbanken, wenn Sie XAMPP einmal deinstallieren möchten. Man kann bei der Deinstallation zwar angeben, dass die Verzeichnisse und Daten erhalten bleiben sollen, jedoch übersieht man den Dialog sehr schnell.

21.3.4 MySQL

Beim Start von phpMyAdmin wird automatisch auf MySQL zugegriffen. Wenn Sie also keine Fehlermeldungen erhalten haben, können Sie davon ausgehen, dass MySQL korrekt installiert ist.

MySQL verwalten

[o] MySQL selbst besitzt keine grafische Benutzeroberfläche. Bei dem Basissystem müssten Sie eigentlich alle Befehle direkt auf der Kommandozeile eingeben – wie in alten MS DOS-Zeiten. Dies wollen wir Ihnen aber ersparen. Wesentlich komfortabler lässt sich MySQL mit phpMyAdmin oder mit dem Tool MySQL Workbench verwalten. Beide Programme haben wir Ihnen ebenfalls auf der Buch-DVD im Ordner *Webserver* beigelegt.

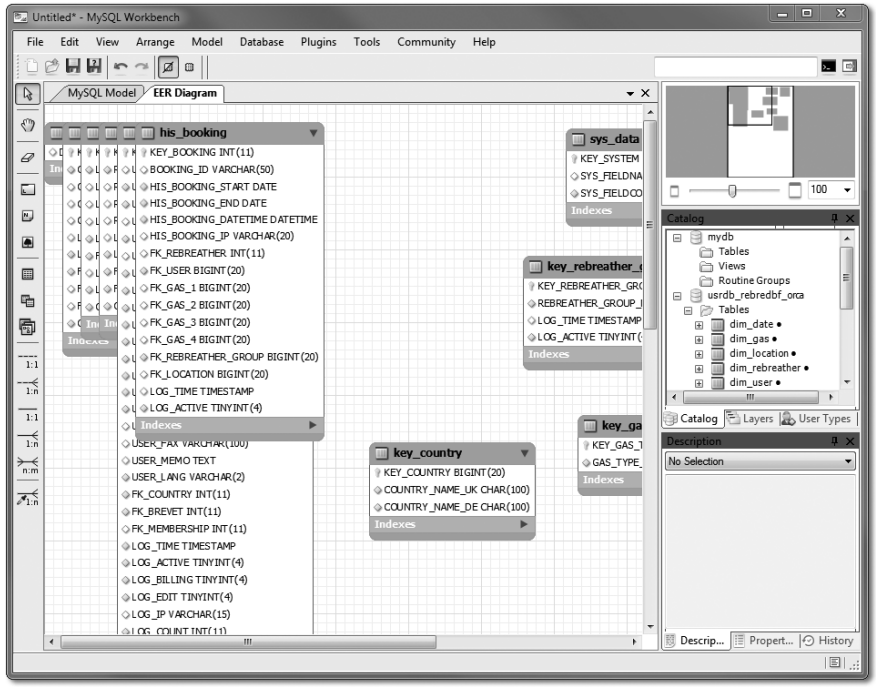


Abbildung 21.10 MySQL Workbench

Letztlich bleibt es Ihnen überlassen, ob Sie mit phpMyAdmin oder mit diesen Tools arbeiten wollen. Im Buch werden wir phpMyAdmin verwenden, da es auch auf den meisten Webservern der Provider in der von uns verwendeten Version installiert ist.

Wenn wir später mit Dreamweaver im Testserverbetrieb arbeiten und Sie Ihre Sites testen wollen, müssen Sie zuvor immer Apache und MySQL starten!

21.4 Internet Information Server installieren

Dreamweaver unterstützt neben PHP auch ASP, ASP.NET und einige andere Sprachen, die nur auf Windows-Systemen lauffähig sind. Mit Apache sind Sie ohne weitere Zusatzmodule fast ausschließlich an PHP gebunden. Es können zwar, wenn in *php.ini* die entsprechenden Bibliotheken freigegeben sind, auch Access- und SQL-Datenbanken abgefragt werden, dafür wird aber eine Codierung von Hand vorausgesetzt. Außerdem werden diese Datenbanken von Dreamweaver nicht unterstützt.

ASP.NET auf dem IIS einsetzen [«]

Wenn Sie alle Features und die neuesten Technologien nutzen möchten, die das Internet zu bieten hat, sollten Sie – nachdem Sie sich mit PHP in das Erstellen dynamischer Websites eingearbeitet haben – auch ASP.NET ausprobieren. Dazu benötigen Sie den IIS und das Microsoft .NET-Framework auf Ihrem Rechner.

Das Erstellen von ASP.NET-Websites ist mit Dreamweaver ähnlich einfach wie mit PHP und MySQL. Mac-Benutzer können einen Web-Hostingdienst mit ASP-Angebot verwenden oder IIS auf einem entfernten Computer installieren.

Das Microsoft .NET-Framework können Sie von der Microsoft-Website kostenlos herunterladen. Bedenken Sie dabei aber, dass Sie dafür eine schnelle Internetverbindung benötigen. Es ist mehrere Hundert MByte groß. Die Einarbeitung in ASP.NET ist deutlich schwerer als bei PHP. Wir werden in Kapitel 24, »Datenbanken mit Dreamweaver anbinden«, noch einmal kurz darauf eingehen.

In der folgenden Schritt-für-Schritt-Anleitung erläutern wir zunächst die Installation unter Windows XP. Windows 7 und Vista folgen direkt im Anschluss.

Apache und IIS nicht parallel installieren! [!]

Installieren Sie nie ein WAMP-System und den IIS gleichzeitig auf einem Rechner. Die beiden Webserver werden zwangsläufig gegeneinander arbeiten. Wenn Sie den IIS testen wollen, entfernen Sie vorher alle Apache-, PHP- und MySQL-Komponenten. Falls Sie bereits längere Zeit damit arbeiten, vergessen Sie nicht, das Verzeichnis *htdocs* sowie die Datenbanken zu sichern!

[▪] *Schritt-für-Schritt: IIS unter Windows XP oder Windows 2000 installieren*

Der Internet-Informationsdienst (IIS) befindet sich auf Ihrer Windows XP- oder Windows 2000-CD, die Sie für die Installation benötigen. Klicken Sie dazu auf START • SYSTEMSTEUERUNG • SOFTWARE, und wählen Sie WINDOWS-KOMPONENTEN HINZUFÜGEN/ENTFERNEN aus.

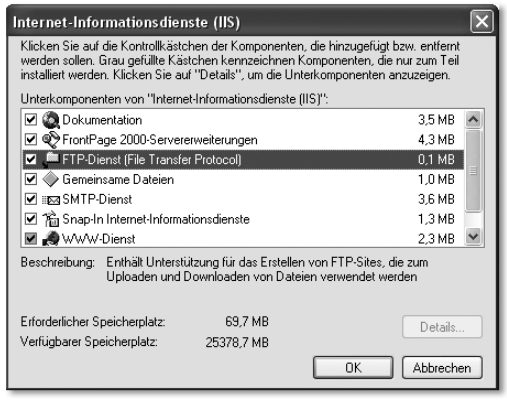
1 *Internet-Informationsdienste hinzufügen*

Aktivieren Sie dann die INTERNET-INFORMATIONSDIENSTE (IIS), und klicken Sie auf DETAILS.



2 *Dienste auswählen*

Wählen Sie hier alle angebotenen Dienste aus. Dies ist der einfachere und sichere Weg, da einige Dienste in Abhängigkeit zueinander stehen. Wenn Sie Detailinformationen zu den Diensten benötigen, lesen Sie am besten die Dokumentationen von Microsoft.

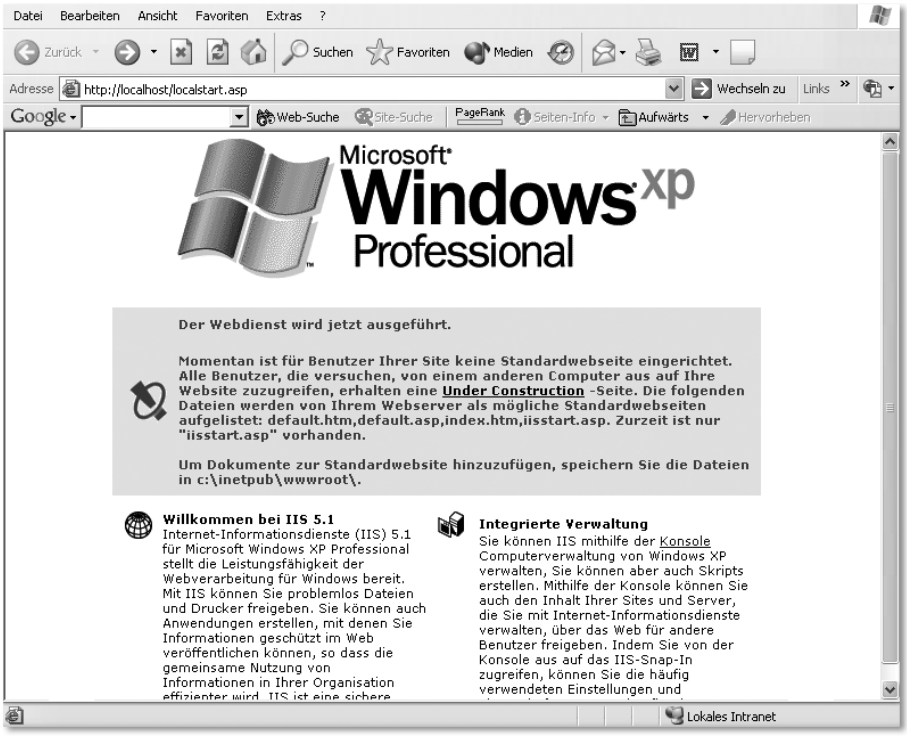


3 *Installations-CD einlegen*

Legen Sie nun die Original-CD von Windows XP/2000 ein, und installieren Sie die neuen Komponenten. Normalerweise müssen Sie im Anschluss an die Installation den Rechner nicht neu starten.

4 *IIS aufrufen*

Geben Sie jetzt in der Browserleiste wieder *http://localhost* ein, und Sie sehen den Bildschirm, den auch die folgende Abbildung zeigt.



21.4.1 IIS verwalten

Die Einstellungen für den IIS werden in der Systemsteuerung vorgenommen. Hier finden Sie jetzt die Internet-Informationsdienste. In der Verwaltung können sämtliche Einstellungen für den IIS überarbeitet werden. Zunächst belassen wir jedoch alles so, wie es ist, um noch PHP und MySQL zu installieren.

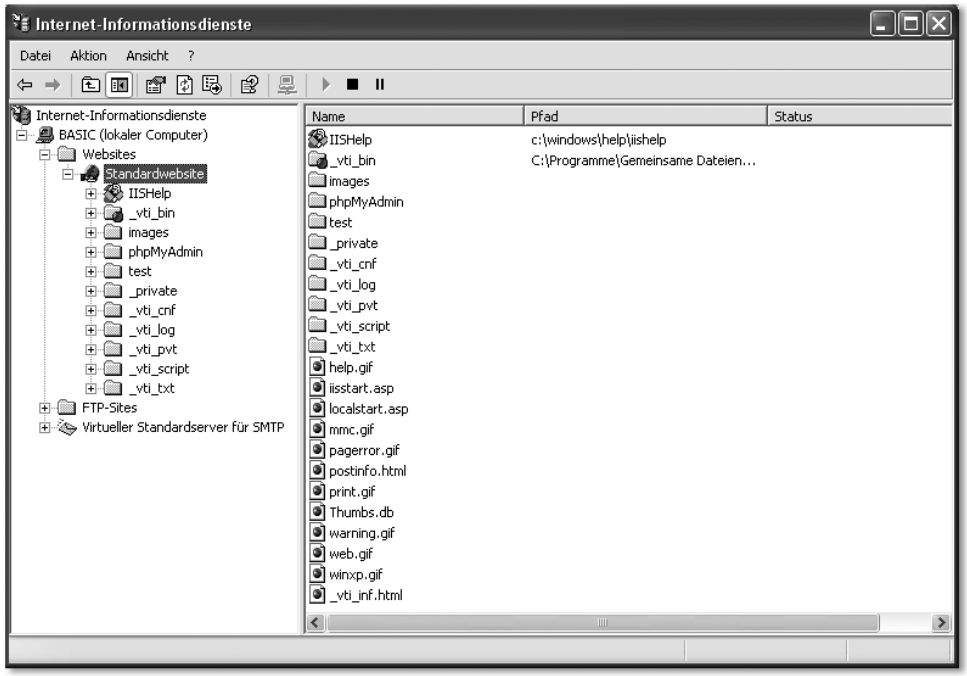


Abbildung 21.11 Die IIS-Verwaltung in der Systemsteuerung

Serverroot des IIS

Auf Ihrem Laufwerk befinden sich einige neue Ordner. Das Unterverzeichnis *wwwroot* im Verzeichnis *Inetpub* ist das Rootverzeichnis des IIS und entspricht dem Verzeichnis *htdocs* beim Apache (siehe Abbildung 21.12).

21.4.2 PHP auf IIS installieren

Die Dateien *php-4.4.1-installer.exe* und *php-5.2.9-1-win32-installer.msi* finden Sie im Internet unter *http://www.php.net* zum freien Download.

[»] **PHP-Version**

Für die Entwicklung mit den von Dreamweaver zur Verfügung gestellten Features spielt es keine Rolle, welche PHP-Version Sie einsetzen.

[▄▀] ***Schritt-für-Schritt: PHP auf IIS installieren***

Im Folgenden beschreiben wir die Installation von PHP 4. Die Schritte bei PHP 5 sind identisch. Als Webserver wählen Sie bei PHP 5 dann bitte IIS-CGI aus.

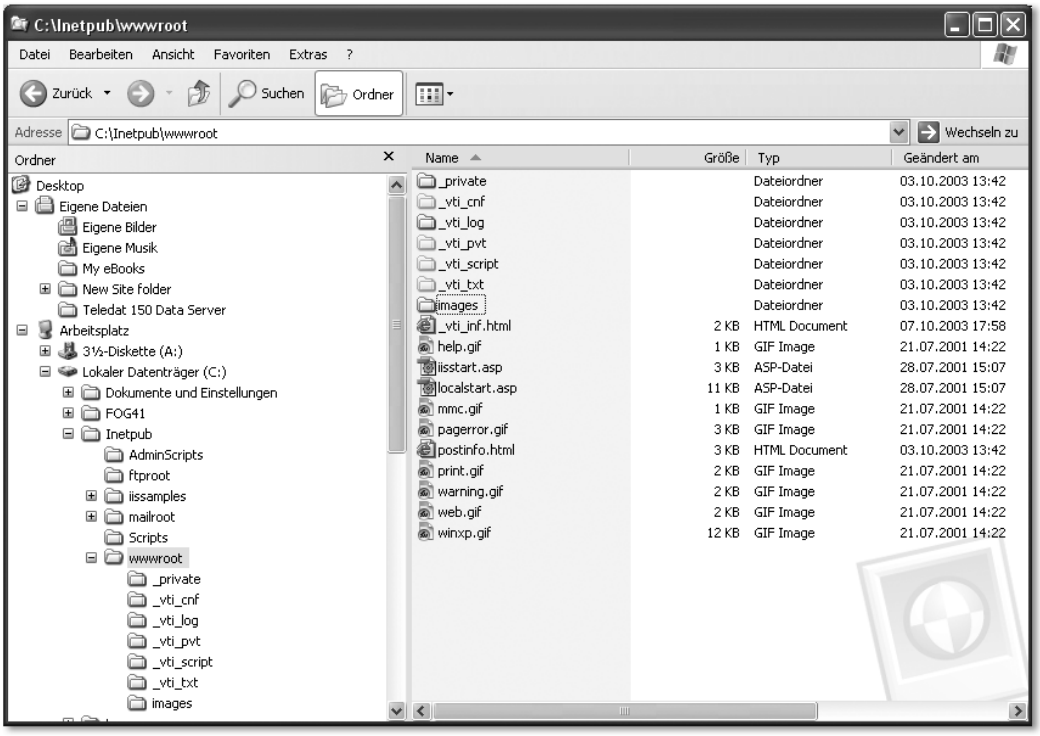


Abbildung 21.12 Rootverzeichnis des IIS

1 *Installation starten*

Um mit der Installation zu beginnen, starten Sie die Datei durch einen Doppelklick und aktivieren den Modus ADVANCED.

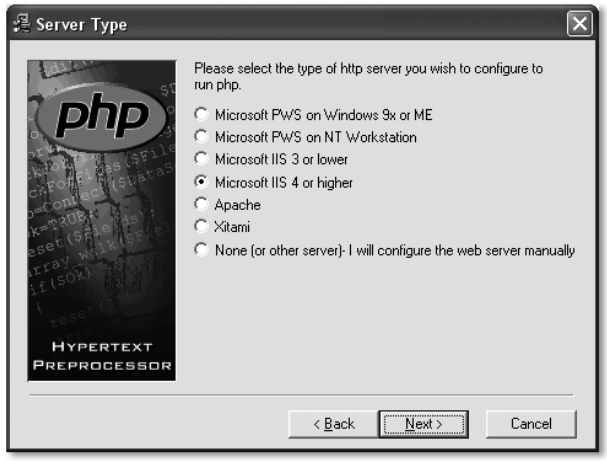
2 *Error Reporting*

Klicken Sie auf NEXT, bis Sie zu der Auswahl für ERROR REPORTING gelangen. Aktivieren Sie hier DISPLAY ALL ERRORS. So werden nur besonders schwere, die Ausführung des Skripts verhindernde Fehler in Dokumenten angezeigt. PHP ist ansonsten sehr mitteilungsbedürftig, was eher störend ist.

Für die Fehlersuche kann diese Einstellung später auch separat vorgenommen oder verändert werden.

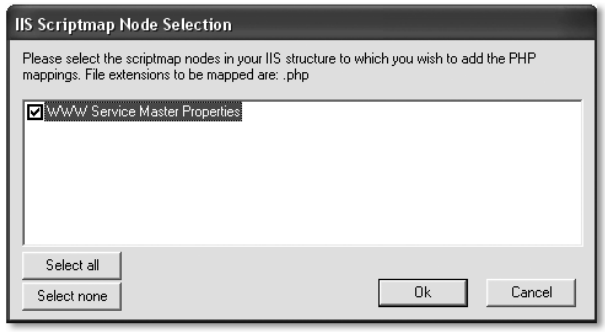
3 *Den richtigen Webserver auswählen*

Wählen Sie als Nächstes den IIS als Webserver aus. Wenn Sie PHP nachträglich für einen der anderen Webserver installieren, können Sie diesen hier auswählen.



4 *PHP dem IIS zuweisen*

Aktivieren Sie im Fenster IIS Scriptmap Node Selection die Option WWW Service Master Properties, um die Einstellungen für PHP in die Konfiguration des IIS aufzunehmen.



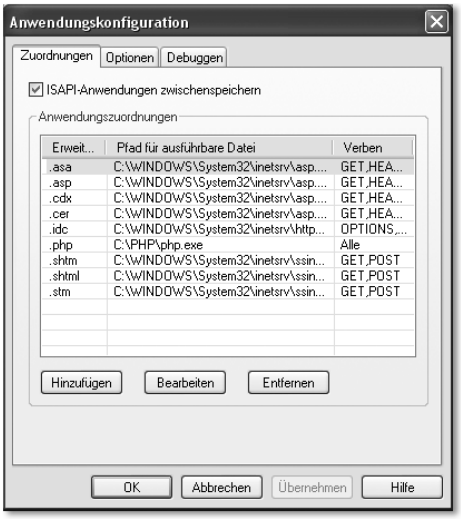
5 *Konfiguration abschließen*

Vollständig konfiguriert ist der IIS damit allerdings noch nicht. Es sind noch weitere Einstellungen notwendig: Klicken Sie mit der rechten Maustaste auf Ihre Standardwebsite in der IIS-Verwaltung, um zur Dialogbox Eigenschaften von Standardwebsite zu gelangen. Aktivieren Sie hier die Kontrollkästchen Skriptzugriff und Lesen. Durch diese Aktion erlauben Sie dem IIS, Skripte auszuführen.



6 *IIS-Anwendungskonfiguration*

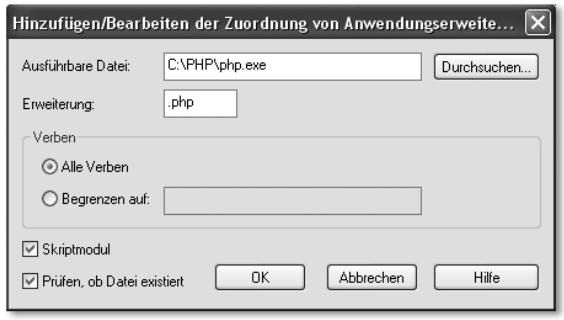
Klicken Sie anschließend im Fenster BASISVERZEICHNISSE auf ANWENDUNGSKONFIGURATION. Es öffnet sich das folgende Fenster.



7 *Verknüpfen der Erweiterung ».php« mit der PHP-Anwendung*

Falls PHP noch nicht in der Anwendungskonfiguration eingetragen ist, muss dem IIS noch mitgeteilt werden, dass der PHP-Interpreter geöffnet werden soll, wenn

eine Datei mit der Endung *.php* aufgerufen wird. Klicken Sie gegebenenfalls auf HINZUFÜGEN, wählen Sie die Datei *php.exe* auf Ihrem Laufwerk aus, und tragen Sie unter ERWEITERUNG die Endung *.php* ein.



8 *Skript zur Anzeige der PHP-Einstellungen*

Erstellen Sie nun im Verzeichnis *wwwroot* eine Datei mit folgendem Inhalt:

```
<?
phpinfo();
?>
```

9 *Infoscreen aufrufen*

Beim direkten Aufruf der Datei im Browser erscheint nun der Infoscreen von PHP.



10 *»php.ini« anpassen*

Damit PHP korrekt funktioniert, muss in der Datei *php.ini* noch eine kleine Änderung vorgenommen werden. Öffnen Sie dazu diese Datei mit einem Editor, und suchen Sie die folgende Phrase:

```
register_globals =
```

Tragen Sie dann, falls hinter dem Gleichzeichen `Off` steht, ein `On` ein, und speichern Sie die Datei.

Damit ist die Installation von PHP abgeschlossen. ■

21.4.3 MySQL unter IIS installieren

Bei der Installation von MySQL gibt es nichts Besonderes zu beachten. Starten Sie einfach die Datei *mysql-essential-5.1.45-win32.msi* im Ordner *Webserver/MySQL* auf der Buch-DVD. [o]

Nach der Installation finden Sie im Ordner *C:/mysql/bin* eine Datei mit dem Namen *winmysqladmin.exe*. Starten Sie diese, und Sie sehen in der Taskleiste rechts unten eine kleine Ampel. Wenn die Ampel Grün anzeigt, ist der MySQL-Server aktiv. Über dieses Tool können Sie auch MySQL als Dienst installieren und verwalten.

Sie verfügen jetzt über ein funktionierendes IIS-System mit MySQL und PHP.

21.4.4 phpMyAdmin unter IIS installieren

Wenn Sie mit dem IIS arbeiten, empfehlen wir Ihnen ebenfalls, phpMyAdmin zu installieren. Extrahieren Sie dafür die Datei *phpMyAdmin-2.5.3-php.zip*, und benennen Sie den nun entstehenden Ordner in *phpMyAdmin* um. Diesen verschieben Sie dann in das Rootverzeichnis (*wwwroot*) des IIS und rufen ihn im Browser über *http://localhost/phpMyAdmin/* auf.

Die Version auf der DVD ist eine neuere als in der XAMPP-Instant-Installation aus dem vorherigen Abschnitt. Wundern Sie sich also bitte nicht über das etwas andere Aussehen.

Warnmeldungen

Sie werden beim ersten Aufruf einen Warnhinweis erhalten, dass Sie doch bitte das Passwort ändern sollen. Das ist nicht zwingend notwendig. Die eventuelle zweite Warnmeldung

```
Das $cfg['PmaAbsoluteUri']-
Verzeichnis MUSS in Ihrer Konfigurationsdatei angegeben werden!
```

sollten Sie jedoch nicht ignorieren. Öffnen Sie die Datei *config.inc.php* im Verzeichnis *wwwroot/phpMyAdmin*, und ergänzen Sie den folgenden Eintrag:

```
$cfg['PmaAbsoluteUri'] = 'http://localhost/phpMyAdmin/';
```

Im weiteren Verlauf des Buches werden wir mit dem zuvor installierten WAMP-System arbeiten, da wir davon ausgehen, dass es bei den meisten Lesern installiert ist, und weil der Apache auch auf dem Mac installiert werden kann.

Unterschiedliche Serverroot

Die weitere Vorgehensweise ist auf beiden Systemen gleich, Sie müssen nur bedenken, dass bei WAMP als Documentroot das Verzeichnis *htdocs* im Verzeichnis *C:/apache* gilt und bei IIS das Verzeichnis *wwwroot* als *C:/Inetpub* zu finden ist. Der IIS muss im Gegensatz zum Apache nicht immer extra gestartet werden, da er als Dienst unter Windows automatisch beim Systemstart hochfährt.

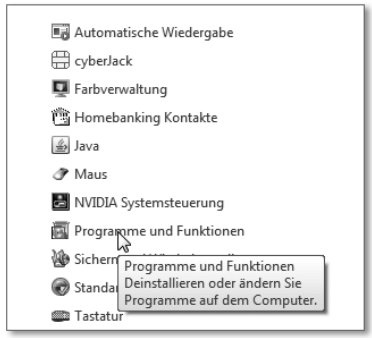
21.4.5 Installation des IIS 7 und PHP unter Windows 7 Business

Die Windows 7 Business Edition und alle höherwertigen Versionen als die Home Editions enthalten den Internet Information Server 7, kurz IIS 7. Er ist jedoch nach der Installation von Vista deaktiviert. Im folgenden Abschnitt zeigen wir Ihnen die Besonderheiten bei der Installation des IIS 7 und PHP unter Windows 7. Mit Berechtigungsabfragen von Windows, einzugebenden Kennwörtern usw. werden wir uns nicht aufhalten. Wir gehen davon aus, dass Sie genügend Erfahrung mit Windows 7 haben, um dies zu handhaben.

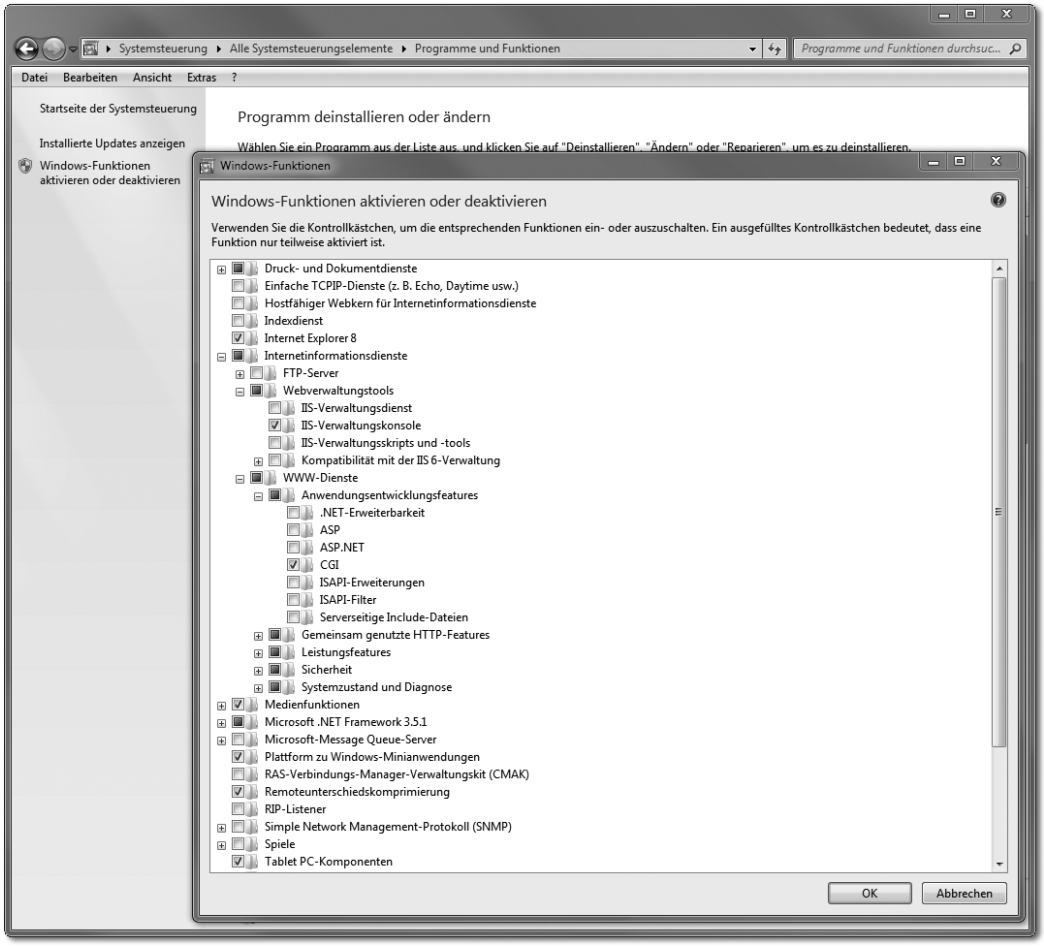
[▪] *Schritt-für-Schritt: IIS unter Windows 7 installieren*

1 *Systemsteuerung aufrufen*

Wechseln Sie zunächst in die Systemsteuerung, und wählen Sie Programme und Funktionen aus.



Im nun folgenden Dialog können Sie die gewünschten Windows-Funktionen aktivieren und deaktivieren.



2 *Windows-Funktionen installieren*

In der Abbildung sehen Sie, welche Optionen installiert werden müssen, um ein funktionsfähiges System zu erhalten. In unserem Beispiel haben wir alle Funktionen aktiviert, um auch mit ASP arbeiten zu können.

Nachdem Sie mit OK bestätigt haben, installiert Windows 7 die gewünschten Programmmodule. Im Regelfall müssen Sie keine DVD einlegen. Sie sollten sie aber besser bereithalten, falls Vista einzelne Komponenten nachladen muss.

Die zwischenzeitlichen Sicherheitshinweise müssen Sie bestätigen. Das gilt auch für alle folgenden Schritte.

3 *IIS testen*

Nach abgeschlossener Installation und einem eventuellen Neustart können Sie im Browser über *http://localhost* den IIS testen.



PHP unter Vista installieren

[o] Für die Installation unter Windows 7 dürfen Sie in keinem Fall das Paket mit dem Windows-Installer verwenden, sondern nur das gezippte Archiv. Der Windows-Installer, mit dem PHP ausgeliefert wird, arbeitet in der aktuellen Version noch nicht korrekt mit Windows 7. Wir haben beide Versionen im Ordner *Webserver/PHP* der Buch-DVD beigelegt.

[»] **Beschreibungen der Installationsroutinen**

Je nach Vista-Version, installierten Servicepacks und PHP-Version kann sich die Installation von der gezeigten unterscheiden. Auf der Website *http://www.php.net* finden Sie aktuelle Beschreibungen der Installationsroutinen.

[▪] ***Schritt-für-Schritt: PHP unter Windows 7 Business installieren***

1 *Zip-Archiv entpacken*

Entpacken Sie PHP in ein Verzeichnis Ihrer Wahl. In unserem Beispiel arbeiten wir mit PHP im Verzeichnis *C:\php*.

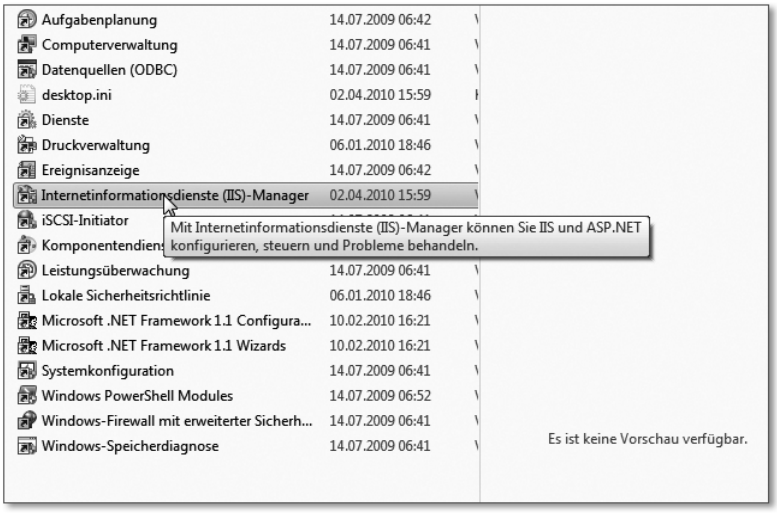
2 *»php.ini-recommended« kopieren*

Kopieren Sie jetzt die Datei *php.ini-recommended*, und benennen Sie die neue Datei in *php.ini* um. In dieser neuen Datei müssen Sie den Eintrag `cgi.force_redirect = 0` hinzufügen.

```
[PHP]
cgi.force_redirect = 0
;;;;;;;;;;;;;;;;;;;
; About php.ini   ;
;;;;;;;;;;;;;;;;;;;
; This file controls many aspects of PHP's behavior.  In order for PHP to
; read it, it must be named 'php.ini'.  PHP looks for it in the current
; working directory, in the path designated by the environment variable
; PHPRC, and in the path that was defined in compile time (in that order).
; Under Windows, the compile-time path is the Windows directory.  The
; path in which the php.ini file is looked for can be overridden using
; the -c argument in command line mode.
```

3 *Systemsteuerung aufrufen*

In der SYSTEMSTEUERUNG wechseln Sie in den Unterpunkt SYSTEM UND WARTUNG und wählen VERWALTUNG aus. Hier sehen Sie nun den INTERNETINFORMATIONSDIENSTE (IIS)-MANAGER. Starten Sie ihn mit einem Doppelklick.



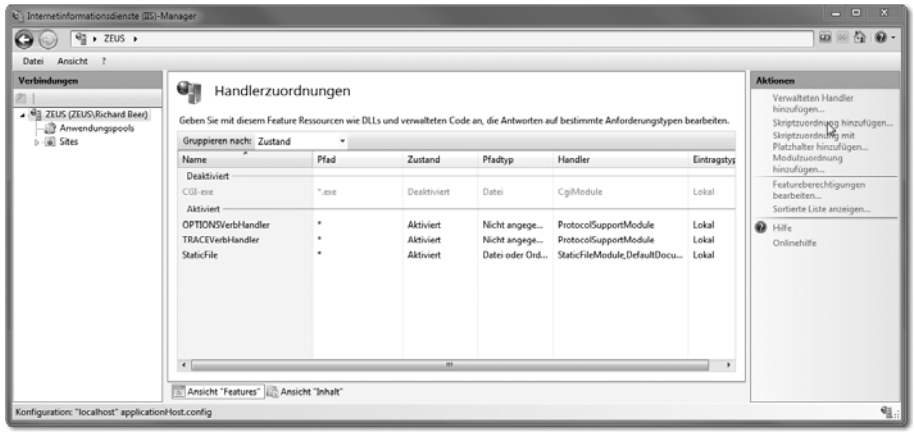
4 *Handlerzuordnung*

In den zur Verfügung stehenden Optionen wählen Sie HANDLERZUORDNUNGEN aus und klicken anschließend auf SKRIPTZUORDNUNG HINZUFÜGEN.

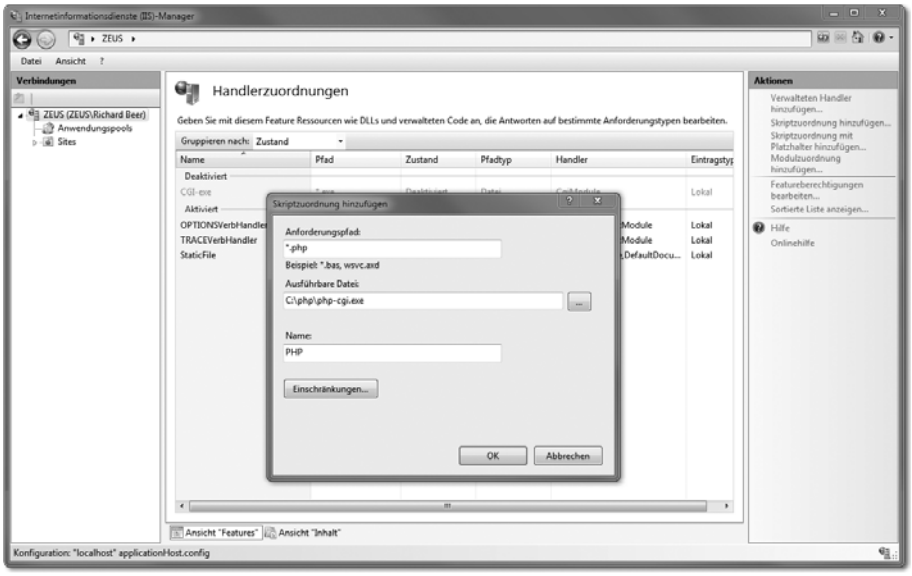
5 *Skriptzuordnung einstellen*

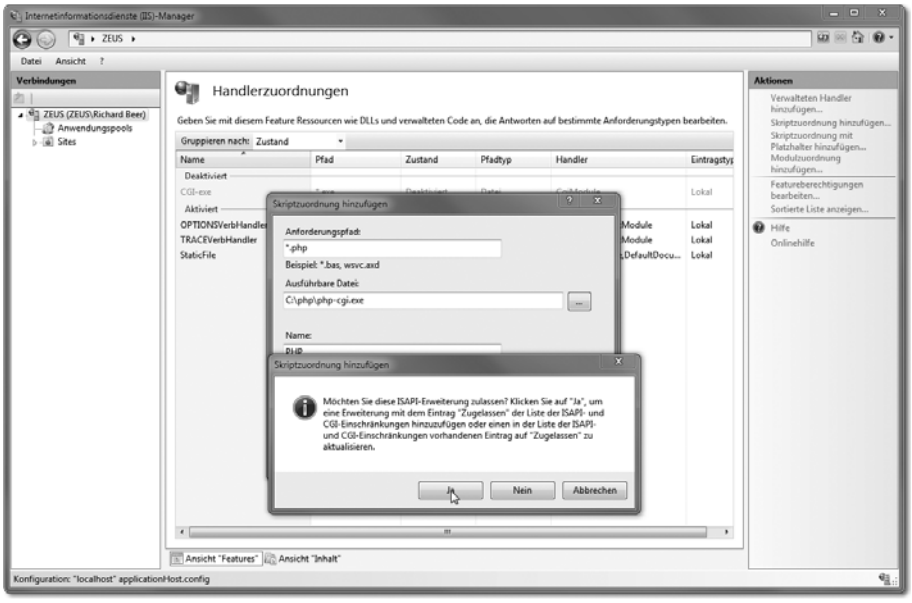
Sie müssen nun dem IIS mitteilen, was geschehen soll, wenn er beim Aufruf eines Dokuments auf die Endung *.php* stößt. Das geschieht, indem Sie bei ANFORDERUNGSPFAD die Dateiendung **.php* angeben. Tragen Sie bei AUSFÜHRBARE DATEI den Pfad zur *php-cgi.exe* ein, oder wählen Sie die Datei mit dem Dateibrowser aus. Zum Dateibrowser gelangen Sie durch einen Klick auf das Icon rechts neben dem Textfeld zum Dateinamen.





Zusätzliche Module werden von Windows 7 automatisch aktiviert. Bestätigen Sie dazu gegebenenfalls die Dialogbox mit einem Klick auf JA.





Die Installation ist nun abgeschlossen. Auf Ihrem Laufwerk finden Sie jetzt ein neues Verzeichnis mit dem Namen *C:/inetpub*. In diesem Verzeichnis liegt ein weiteres mit dem Namen *wwwroot*. Wie es der Name erwarten lässt, handelt es sich hierbei um die DocumentRoot des IIS 7.

6 *Installation testen*

Erstellen Sie mit Dreamweaver eine Datei mit dem Namen *info.php*, und fügen Sie folgenden Quelltext ein:

```
<? phpinfo() ?>
```



War Ihre Installation erfolgreich, sollte Ihnen nach dem Aufruf von *http://localhost/info.php* folgender Bildschirm angezeigt werden:

PHP Version 5.2.13

System	Windows NT ZEUS 6.1 build 7600
Build Date	Feb 24 2010 14:35:36
Configure Command	cscript /nologo configure.js "--enable-snapshot-build" "--enable-debug-pack" "--with-snapshot-template=d:\php-sdk\snap_5_2\vc6\x86\template" "--with-php-build=d:\php-sdk\snap_5_2\vc6\x86\php_build" "--disable-zts" "--disable-isapi" "--disable-nsapi" "--with-pdo-oci=D:\php-sdk\oracle\instantclient10\sdk,shared" "--with-oci8=D:\php-sdk\oracle\instantclient10\sdk,shared" "--without-pi3web" "--enable-mongo=shared"
Server API	CGI/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	C:\Windows
Loaded Configuration File	C:\php\php.ini
Scan this dir for additional .ini files	(none)
additional .ini files parsed	(none)
PHP API	20041225
PHP Extension	20060613
Zend Extension	220060519
Debug Build	no
Thread Safety	disabled
Zend Memory Manager	enabled
IPv6 Support	enabled
Registered PHP Streams	php, file, data, http, ftp, compress.zlib
Registered Stream Socket Transports	tcp, udp
Registered Stream Filters	convert.iconv.*, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, zlib.*

[»] **Berechtigungen bei Windows 7**

Soll eine Datei direkt in den *wwwroot*-Ordner gespeichert werden, erhalten Sie eventuell die Meldung, Sie hätten nicht genug Rechte, um das zu tun. Speichern Sie in diesem Fall die Datei an einem anderen Ort, und kopieren Sie das File in den Ordner.

Nachdem wir nun einen lokalen Webserver installiert haben, bereiten Sie Dreamweaver für die Arbeit mit dynamischen Websites vor. In diesem Kapitel erfahren Sie, wie das geht.

22 Dynamische Sites in Dreamweaver einrichten

Vor allen weiteren Arbeiten mit dynamischen Websites muss Dreamweaver für den Testserverbetrieb eingerichtet werden. Wichtig ist dabei, dass sich der lokale Stammordner Ihrer Site im Dokumentverzeichnis Ihres zuvor eingerichteten Webservers befindet und natürlich der Testserver, wie in den vorherigen Kapiteln beschrieben, installiert ist. Einen ersten Test sollte er auch bestanden haben.

22.1 Testserver-Einstellungen

Die Einstellungen für Dreamweaver werden in der Site-Verwaltung vorgenommen. Legen Sie dort am besten unter SITE – NEUE SITE… eine neue Site an.



Abbildung 22.1 Stammordner in der Documentroot

Dann wechseln Sie in der SITE-DEFINITION zum Menüpunkt SERVER. Klicken Sie auf das Plussymbol ❶ (siehe Abbildung 22.2), um eine neue Serverkonfiguration hinzuzufügen. Natürlich können Sie die folgenden Schritte auch mit jeder bestehenden Site durchführen.

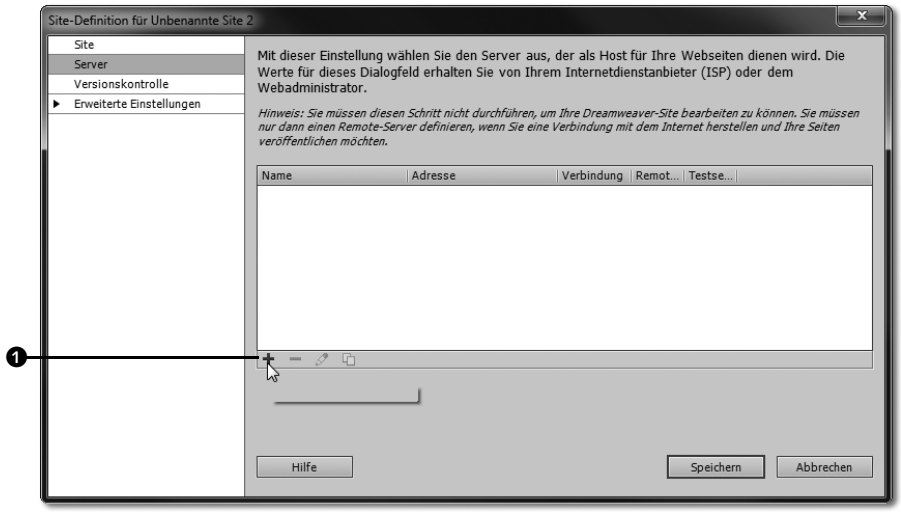


Abbildung 22.2 Serverkonfiguration hinzufügen

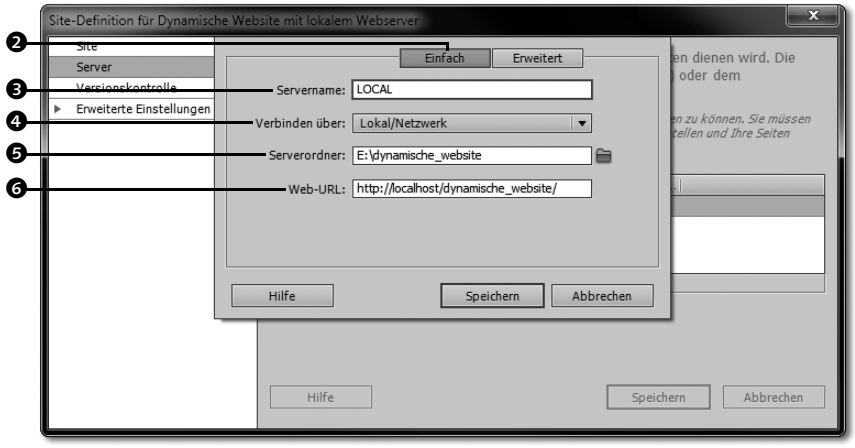


Abbildung 22.3 Menüpunkt »Server« in der Site-Definition

Achten Sie bitte darauf, dass im Menü die Schaltfläche EINFACH ❷ aktiv ist. Zunächst müssen Sie hier dem Server einen Namen geben ❸. Dieser Name hat keinen Einfluss auf die Funktion, sondern dient ausschließlich Ihrer eigenen Verwaltung. In Dreamweaver können mehrere verschiedene Testserver konfiguriert werden, daher die notwendige Benennung, doch dazu später mehr.

Zugriff auf den Server

Als Nächstes muss Dreamweaver erfahren, wie der Zugriff auf den Server erfolgen soll. In den meisten Fällen – wie auch in unserem – wird das die Einstellung

Lokal/Netzwerk sein. Im Dropdown-Feld Verbinden über ❹ können Sie diese auswählen.

Sie können mit Dreamweaver auch direkt auf einem *echten* Webserver im Internet arbeiten. Wenn Sie dies möchten, wählen Sie unter Verbinden über einfach FTP aus und geben dann die Zugangsdaten ein.

Testserver-Ordner

Der Serverordner ❺ ist in unserem Fall identisch mit dem lokalen Stammordner und muss entsprechend ausgewählt werden. Dreamweaver nimmt Ihnen die Arbeit ab, die Dateien immer auf den Server zu kopieren. Wenn Sie FTP als Zugriff angegeben haben, wird beim Arbeiten die aktuelle Datei automatisch auf den Server kopiert und die Vorschau über die URL des Webservers generiert. Dieses Verfahren dauert jedoch deutlich länger als bei einem lokal installierten Testserver.

Adresse der Website

Die Web-URL ❻ ist der Pfad bzw. die HTTP-Adresse, die Sie im Browser eingeben müssen, um die Website anzuzeigen. In der Regel trägt Dreamweaver hier nichts ein, zur Funktion ist die Angabe jedoch zwingend notwendig. Nur wenn die Web-URL korrekt angegeben wurde, kann eine Website durch Dreamweaver ausgeführt werden. Die meisten Fehler beim Anlegen eines Testservers treten genau an dieser Stelle auf. Überprüfen Sie daher den Eintrag auf seine Korrektheit.

Die Web-URL setzt sich aus der lokalen Adresse des Servers – das ist in den meisten Fällen *http://localhost* oder die IP 127.0.0.1 – und dem Ordnernamen Ihrer Website zusammen. Abbildung 22.3 zeigt, wie die Bezeichnungen eingetragen werden müssen.

Servermodell angeben

Im Menüpunkt Erweitert ❼ (siehe Abbildung 22.4) wird das Testsystem ausgewählt. In unserem Fall ist es ein PHP-MySQL-System ❽. Welcher Server (IIS oder Apache) tatsächlich im Hintergrund agiert, spielt dabei keine Rolle. Wichtig ist nur die Auswahl des Systems, um die spätere Syntax für die Abfragen usw. festzulegen.

Anschließend wird der Server im Hauptbildschirm des Dialogfensters noch als Testserver aktiviert ❾ (siehe Abbildung 22.5). (Ein Testserver kann übrigens auch als FTP-Server konfiguriert werden.) Ebenso können Sie mehrere Testserver anlegen und nach Bedarf umschalten. So könnten Sie einen firmeninternen Testserver nutzen, um die Websites zu entwickeln – nichts anderes machen wir hier –, und einen weiteren Live-Server, der im Intranet zugänglich ist.

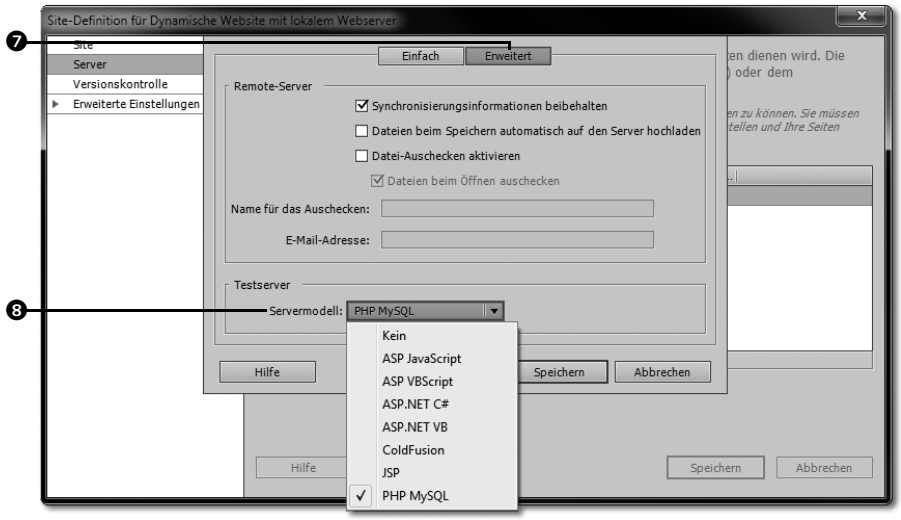


Abbildung 22.4 Erweiterte Servereinstellungen



Abbildung 22.5 Server aktivieren

22.2 Site-Einstellungen testen

Nach dem Anlegen des Servers in Dreamweaver sollte er getestet werden. Erstellen Sie dazu am besten ein neues Dokument *index.php*. Wichtig ist die Endung *.php*: Der Webserver erkennt sonst nicht, dass es sich bei dieser Datei um ein PHP-Skript handelt. Löschen Sie den gesamten Quelltext der Datei, und fügen Sie folgende Befehlszeile ein:

```
<?php
phpinfo()
?>
```

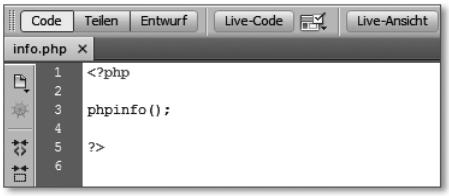


Abbildung 22.6 Aufruf der Funktion phpinfo() im Quelltext

Mit diesem Befehl können Sie die PHP-Grundeinstellungen anzeigen lassen. Wechseln Sie anschließend wieder in die Layoutansicht.

Vorschau dynamischer Sites

In der Layoutansicht (siehe Abbildung 22.7) können Sie die Einstellungen durch einen Klick auf die LIVE-ANSICHT ❶ testen. Im Dokumentfenster wird das PHP-Skript direkt ausgeführt, und Sie sehen im Layout die automatisch von PHP generierten Inhalte.

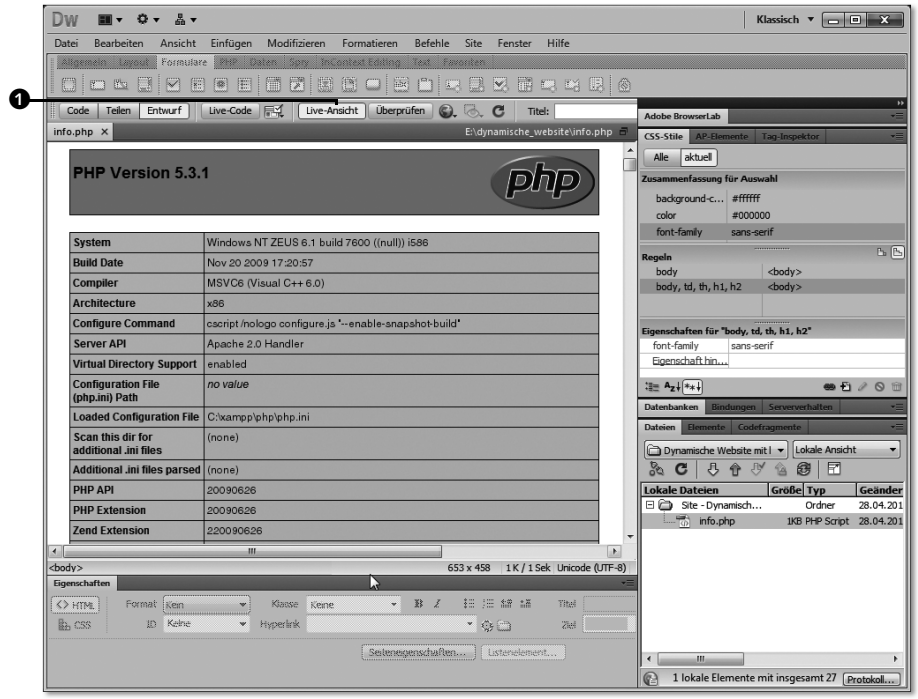


Abbildung 22.7 Den Testserver aktivieren

Vorschau in der Live-Code-Ansicht

In Dreamweaver CS5 können Sie neben der LIVE-ANSICHT auch eine LIVE-CODE-Ansicht aktivieren ❶. Im Layoutfenster zeigt Dreamweaver dann den von PHP erzeugten Quelltext direkt an.

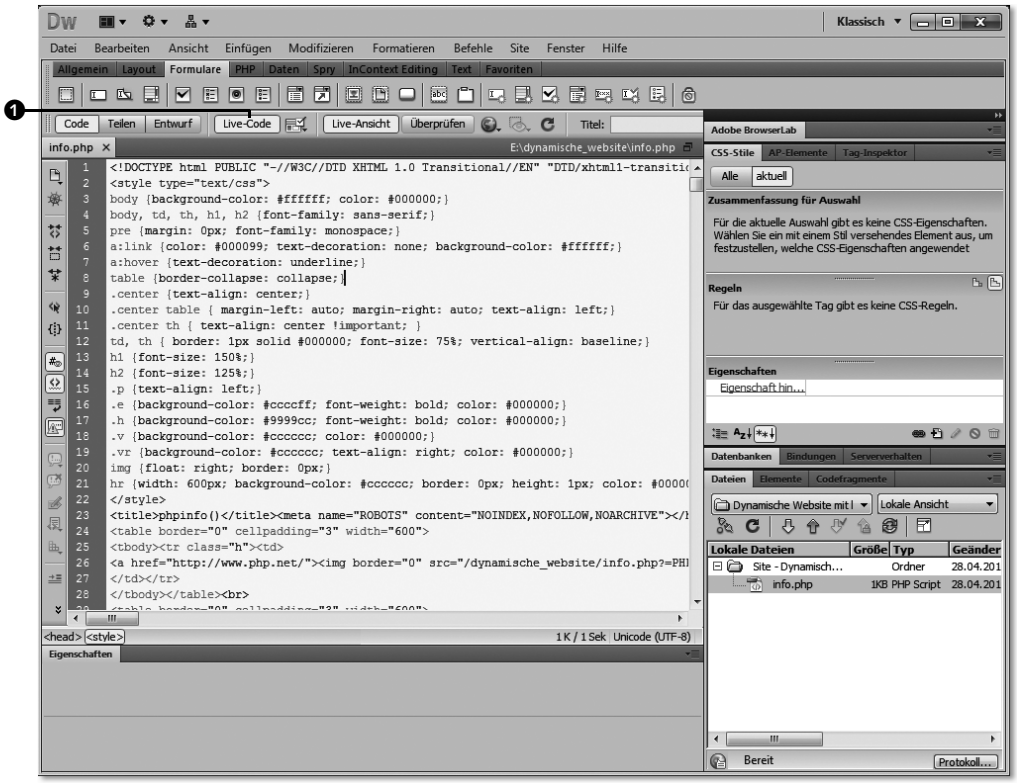


Abbildung 22.8 Live-Code-Ansicht

Wenn die Live-Ansichten nicht funktionieren (Fehlersuche)

Sollte dies wider Erwarten nicht funktionieren, gehen Sie am besten folgende Liste durch:

- Überprüfen Sie, ob der Webserver aktiv ist.
- Haben Sie die Site korrekt definiert? Falls nicht, erscheint entweder das Icon zur Live-Ansicht nicht, oder Dreamweaver gibt eine Fehlermeldung aus.
- Stellen Sie sicher, dass Ihre Dateien wirklich in der Documentroot liegen. In 90 Prozent aller Fehlerfälle in Schulungen sind die Pfade nicht korrekt eingegeben worden, oder die Web-URL ist falsch.
- Überprüfen Sie der Reihe nach alle Einstellungen.

Keine Frame-Vorschau dynamischer Sites [«]

Dynamische Sites können Sie lokal nicht über ein Frameset aus Dreamweaver heraus aufrufen. Starten Sie immer ein Inhaltsdokument, um eine Vorschau zu erhalten, oder rufen Sie das Frameset direkt aus dem Browser heraus auf, indem Sie den Pfad zum Frameset eingeben.

Testserver-Ansicht

In der Site-Ansicht steht Ihnen jetzt auch die Option TESTSERVER ❷ zur Verfügung. Aktivieren Sie Ihren Testserver durch einen Klick darauf. Die Ordner auf der linken Seite ❸ werden jetzt rot dargestellt.

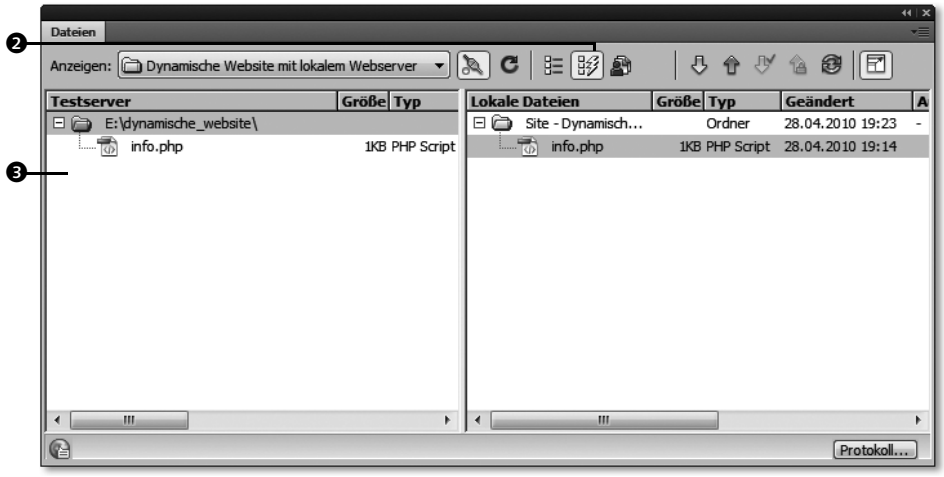


Abbildung 22.9 Testserver-Ansicht in der Site-Verwaltung

Wenn Sie Ihren Testserver lokal betreiben, ist diese Ansicht etwas verwirrend, zumal ja eigentlich beide Seiten gleich sind. Für einen Testbetrieb spielt es aber keine Rolle, wo sich der Testserver befindet. Auch Ihr eigener Rechner wird über eine IP-Adresse angesprochen (127.0.0.1).

Dreamweaver ist jetzt für den Betrieb mit einem Testserver vollständig eingerichtet.

Daten auf dem Testserver sichern! [!]

Wenn Sie auf einem Windows-System arbeiten, werden Apache und IIS auf Ihrem Laufwerk *C:* installiert. Dieses wird in den meisten Fällen auch Ihr Systemlaufwerk sein und ist sehr anfällig für Systemfehler. Bei einer Neuinstallation Ihres Systems sind im schlimmsten Fall alle Daten für immer verloren. Sorgen Sie daher dafür, dass die Daten im Rootverzeichnis des Webservers regelmäßig auf einer anderen Partition oder CD gesichert werden.

22.3 Buchwebsite für dynamische Inhalte einrichten


Buchwebsite

Wir gehen jetzt davon aus, dass Sie bereits einige Übung im Umgang mit Dreamweaver und dessen Layoutfunktionen haben. Wir sparen uns daher in allen weiteren Schritten beim Aufbau der Buchwebsite diese Erklärungen. Wenn es an der einen oder anderen Stelle hakt, lesen Sie bitte in den entsprechenden Kapiteln nach.

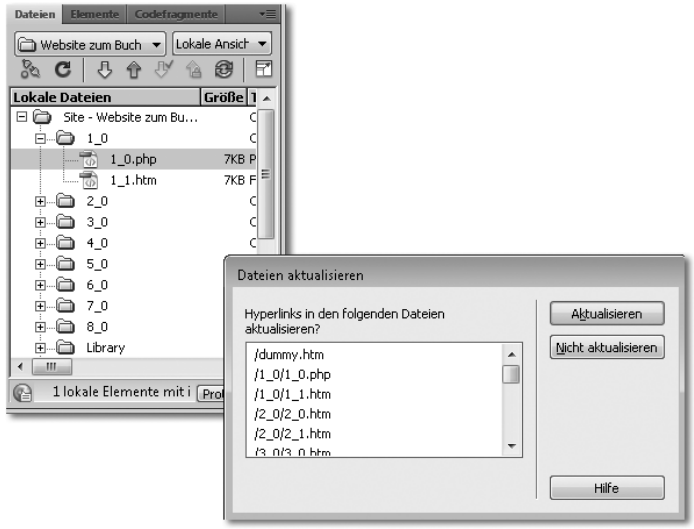
Ebenso ist es erforderlich, die einleitenden Kapitel zum Aufbau des dynamischen Teils zu lesen. Ohne das Wissen über die Funktionen ist es sehr schwer, die Arbeitsschritte nachzuvollziehen.

[▪] *Schritt-für-Schritt: Website für dynamische Inhalte einrichten*

1 *Dateiendungen ändern*

Um auch weiter mit der von Ihnen erstellten Buchwebsite arbeiten zu können, müssen einige Änderungen an den Site-Einstellungen erfolgen.

Ändern Sie die Dateiendung aller HTML-Dokumente in *.php*. Nehmen Sie dies bitte unbedingt im Dateifenster von Dreamweaver vor, und lassen Sie alle Verlinkungen automatisch aktualisieren, wenn Dreamweaver dies vorschlägt.



2 *Lokalen Stammordner kopieren*

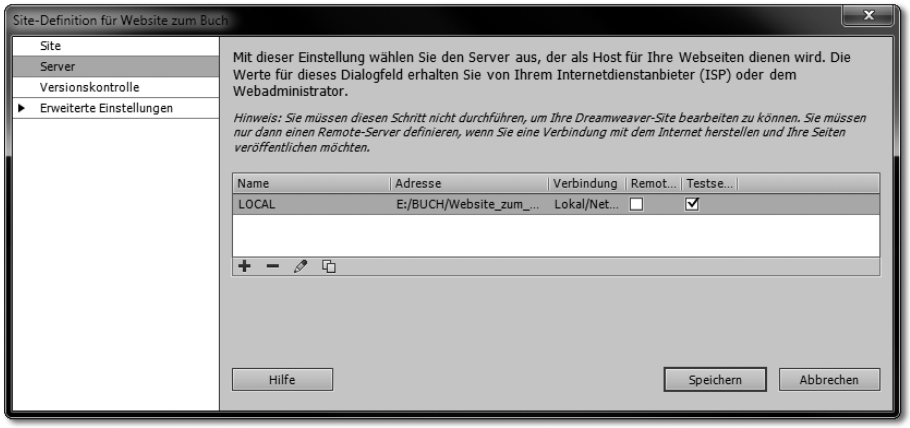
Kopieren Sie anschließend den gesamten bisherigen lokalen Stammordner Ihrer Übungswebsite in das Dokumentverzeichnis Ihres lokalen Webservers.

3 *Site-Einstellungen anpassen*

Ändern Sie die Site-Einstellungen Ihrer Übungswebsite so, dass der in die DokumentRoot kopierte Ordner als lokales Stammverzeichnis angegeben wird, oder erstellen Sie eine neue Site mit den soeben genannten Vorgaben.

4 *Einstellungen für den Testserver*

Legen Sie anschließend, wie in der folgenden Abbildung gezeigt, die Einstellungen für den Testserver fest. Sie müssen natürlich Ihre lokalen Parameter eintragen.



5 *Geänderte Einstellungen testen*

Öffnen Sie nun das Dokument *1_0.php*, und starten Sie mit F12 bzw. Alt+F12 am Mac eine Browservorschau. Wenn Sie alles richtig angelegt haben, wird dieses Dokument genauso angezeigt wie bisher.

Eine Live-Ansicht können Sie erst starten, wenn sich im Dokument PHP-Skripte befinden. Dies ist zum jetzigen Zeitpunkt noch nicht der Fall.

Wenn Sie nun alles beendet haben, steht einer erfolgreichen Arbeit nichts mehr im Wege. ■

MySQL ist die Datenbank unserer Wahl, da sie die meisten Provider in Kombination mit PHP als Service anbieten. Wir zeigen Ihnen den richtigen Umgang mit MySQL.

23 MySQL-Grundlagen

MySQL ist in der Standardversion kostenlos erhältlich und sehr leistungsfähig. Zu einer Datenbank gehören das Datenbanksystem und eine Möglichkeit, diese abzufragen. Die Abfrage erfolgt mit einer eigenen Sprache, kurz *SQL* (*Structured Query Language*) genannt. Eine kurze Einführung dazu gibt es in Abschnitt 23.5.

23.1 Einführung in Datenbanken

Sehr tief können wir in diese Thematik hier nicht einsteigen, das würde den Rahmen des Buches sprengen. Bei Galileo Press gibt es aber einige sehr gute Bücher zum Thema MySQL, die wir Ihnen empfehlen können.

In erster Linie wollen wir uns im Folgenden auf die Anbindung von MySQL in Dreamweaver konzentrieren. Dennoch sind einige kurze Erläuterungen an dieser Stelle notwendig. Sehr vereinfacht gesagt enthält eine Datenbank die folgenden Komponenten:

- ein Data-Base-Management-System zur Verwaltung der Datenbank
- die Datenbank selbst
- Tabellen, die sich in den Datenbanken in beliebiger Anzahl befinden
- Datensätze in Tabellen (jeder einzelne Datensatz wird als *Tupel* bezeichnet)
- Datenfelder in den Tabellen

Abbildung 23.1 zeigt, wie diese Komponenten zusammenhängen.

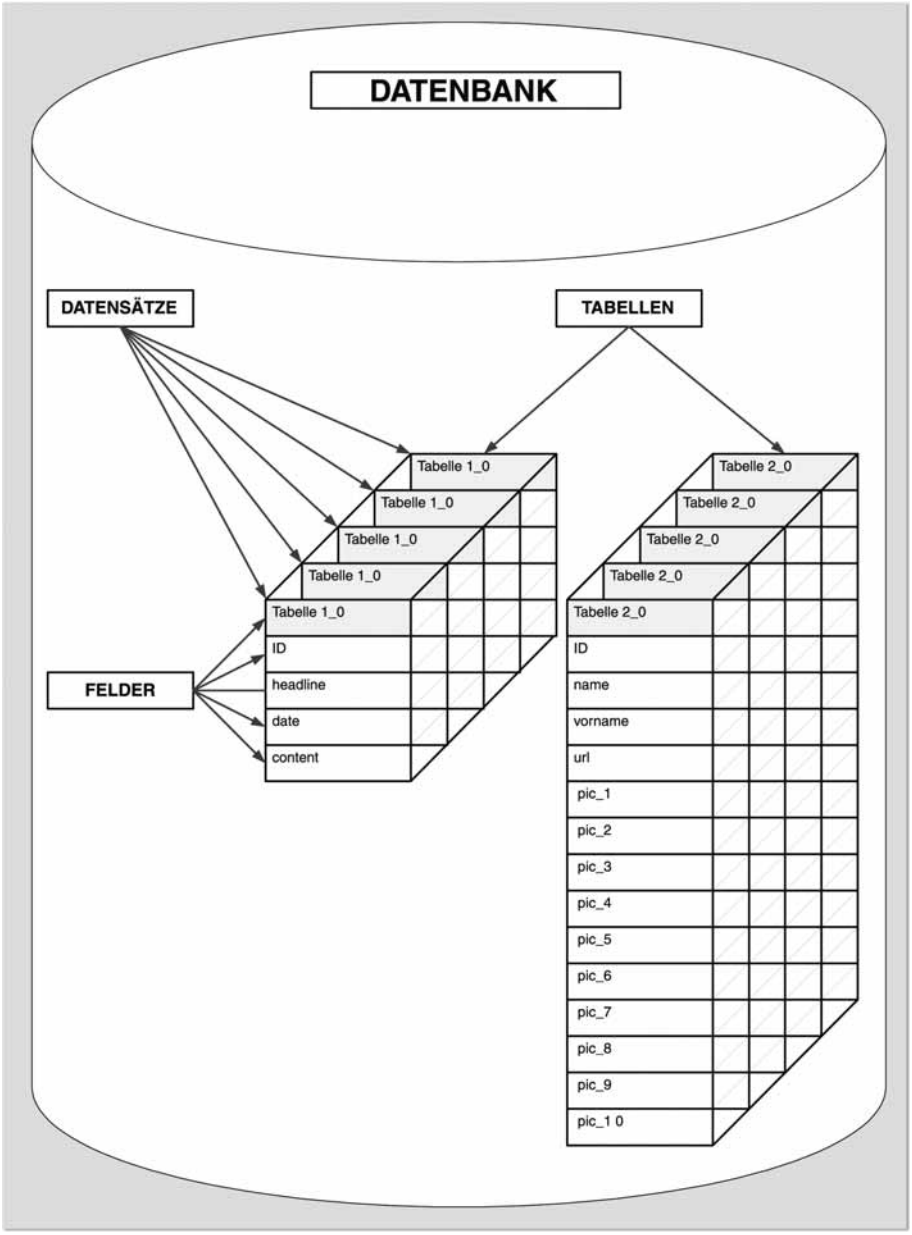


Abbildung 23.1 Allgemeiner Aufbau einer Datenbank

23.1.1 Relationale Datenbanken

MySQL und fast alle anderen modernen Datenbanken sind sogenannte *relationale* Datenbanken. Diese bestehen aus den eben beschriebenen Tabellenstrukturen

und Beziehungen (Relationen) zwischen den einzelnen Tabellen. Betrachten Sie Abbildung 23.2 – in den Tabellen gibt es zunächst keine Verbindung zwischen Kunden und Produkten.

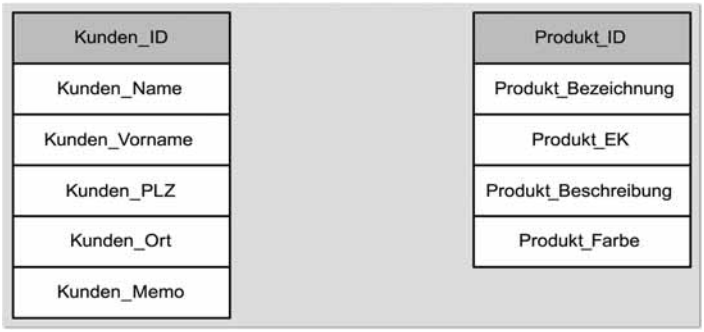


Abbildung 23.2 Eine einfache Datenstruktur für ein Shopsystem

Wir möchten erreichen, dass jeder Kunde Bestellungen ausführen und dabei beliebige Produkte bestellen kann. Diese bestellten Produkte sollen den Kunden zugeordnet werden können. Man könnte jetzt der Kundentabelle ein weiteres Feld für Bestellungen hinzufügen und die Produktbezeichnung darin abspeichern. Wenn der Kunde jedoch mehrere Produkte oder ein Produkt mehrfach bestellt, müsste für jeden Bestellvorgang ein weiteres Datenfeld angefügt werden. Dieses müsste zudem für alle Kunden angelegt werden, da Felder immer in allen Datensätzen einer Tabelle erscheinen. Das wäre somit ziemlich umständlich.

Eine einfachere und bessere Lösung bietet das relationale Modell, das wir in Abbildung 23.3 skizzieren.

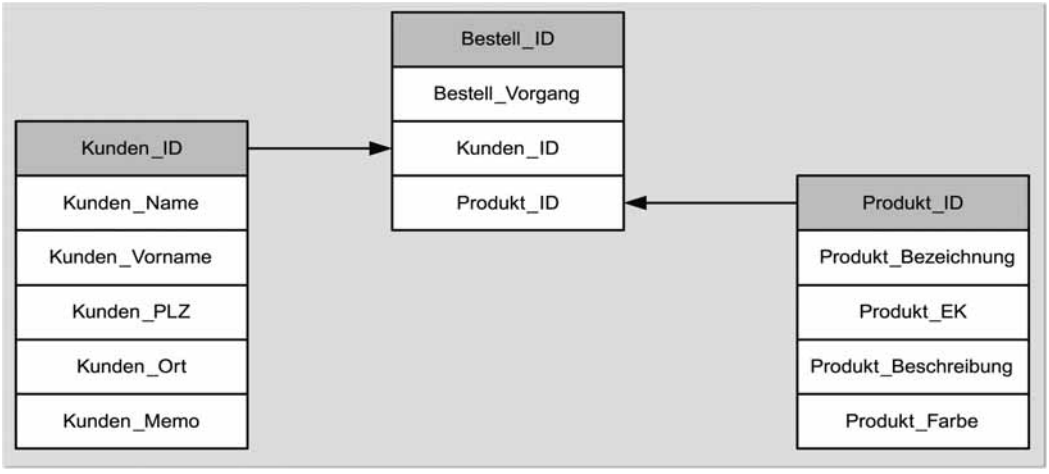


Abbildung 23.3 Relationale Datenbank mit Tabellen in der Relation n:m

Beziehungen herstellen

Es wird eine dritte Tabelle erstellt, um die Beziehung zwischen den beiden anderen Tabellen zu definieren. In unserem Beispiel wird für jeden neuen Bestellvorgang ein neuer Datensatz angelegt. In diesem werden die Bestellnummer, die Kunden-ID und die Produkt-ID hinterlegt. Der Vorteil dieser Methode ist die wesentlich höhere Flexibilität. Für jede neue Beziehung zwischen Tabellen können Sie eine weitere Tabelle anlegen und bei Bedarf auch wieder löschen. An den eigentlichen Hauptdatensätzen muss keine Veränderung stattfinden.

23.1.2 Relationstypen

Es gibt drei verschiedene Arten von Relationen:

- **n:m-Beziehungen**
  Diese Art der Beziehung bilden die Tabellen in unserem Beispiel. Jeder Datensatz einer Tabelle kann mit beliebig vielen Datensätzen einer anderen Tabelle eine Beziehung bilden.
- **1:1-Beziehungen**
  Diese Beziehung definiert, dass jeder Datensatz nur einem anderen Datensatz zugehörig ist bzw. sein darf.
- **1:n-Beziehungen**
  Bei dieser Beziehung kann ein Datensatz mit beliebig vielen (n) Datensätzen einer anderen Tabelle verbunden sein, aber nicht umgekehrt.

23.1.3 Primärschlüssel

Um mit relationalen Datenbanken zu arbeiten, müssen Datensätze eindeutig definiert sein. Das erreicht man mit einem Primärschlüssel.

Primärschlüssel angeben

Der Primärschlüssel ist eine Zahl, die automatisch beim Anlegen eines neuen Datensatzes hochgezählt (auto-inkrementiert) und mit dem jeweiligen Datensatz gespeichert wird. Ein Feld in einer Datenbank muss eindeutig als Primärschlüssel ausgewiesen werden. In unserem Beispiel sind die Datensätze Kunden_ID, Produkt_ID und Bestell_ID die Primärschlüssel. Für die Bestimmung von Primärschlüsseln sind folgende Vorgaben zwingend zu beachten:

- Primärschlüssel müssen immer einen Inhalt haben.
- In jeder Tabelle wird nur ein Feld als Primärschlüssel deklariert.
- Vom Primärschlüssel darf es keine Duplikate geben.

23.1.4 Redundanzfreiheit

Damit die eben erwähnten Bedingungen erfüllt werden, muss sichergestellt werden, dass jeder Datensatz nur einmal vorkommt. Die häufigste Fehlerquelle ist dabei mit großem Abstand der Mensch. Schnell wird der Kunde Maier noch einmal als Mayer oder Meier gespeichert, und die Eindeutigkeit ist verloren. Wenn zu erwarten ist, dass solche Probleme auftreten, müssen Sie bereits bei der Eingabe Vorkehrungen treffen, indem Sie zum Beispiel identische Namen, Adressen usw. überprüfen.

23.1.5 Prozessdatenfreiheit

Vermeiden Sie unter allen Umständen das Abspeichern berechneter Werte. Nehmen Sie als Beispiel einen Kunden, dessen Alter Sie als Datensatz hinterlegen möchten. Wenn Sie jetzt sein Alter real abspeichern, zum Beispiel 40 Jahre, stimmt dieser Wert im nächsten Jahr bereits nicht mehr. Hinterlegen Sie in diesem Fall das Geburtsdatum, und errechnen Sie das Alter außerhalb der Datenbank immer wieder aufs Neue. Prozessdaten gehören nicht in eine Datenbank!

23.1.6 Fremdschlüssel

In der Tabelle für den Bestellvorgang finden sich die Inhalte der Primärschlüssel aus den beiden anderen Tabellen wieder. Diese eingefügten Werte aus Primärschlüsseln dritter Tabellen bezeichnet man als *Fremdschlüssel*. Fremdschlüssel bilden die eigentliche Beziehung der Tabellen zueinander.

23.1.7 Referenzielle Integrität

Aus Relationen kann sich allerdings auch ein Problem ergeben. Stellen Sie sich vor, ein Kunde hat 100 Bestellungen über einen längeren Zeitraum aufgegeben. Nun wird eines der Produkte gelöscht oder der Preis verändert. Der erste Fall (die Löschung) würde eine Anfrage mit einem Verweis auf ein nicht mehr vorhandenes Produkt ergeben. Im zweiten Fall, der wesentlich fataler ist, würde eine nachträgliche Kundenumsatzberechnung völlig falsche Ergebnisse liefern. Die Gewährleistung, dass solche Fehler nicht auftreten, nennt man *referenzielle Integrität*.

Fehler wie der geschilderte sind unter allen Umständen zu vermeiden, und durch entsprechende Programmierung ist Abhilfe zu schaffen. Das ist jedoch ein sehr komplexes Thema. Leider bietet MySQL keine integrierte Unterstützung dafür an, und Sie müssen sich mit Workarounds in PHP behelfen.

Bei einem WCMS spielt dies keine große Rolle, aber wenn Sie ein Shopsystem mit Bestellverwaltung programmieren wollen, ist das sehr wohl relevant. Das ist auch einer der Gründe, warum wir Anfänger vor dem Programmieren eines Shopsystems nur warnen können.

23.1.8 Endlosschleifen

Im Bereich dynamischer Menüführung tritt ein weiteres Problem auf. Möchten Sie in Ihrem CMS die Menüpunkte frei definieren und beliebig zuweisen, kann es vorkommen, dass sich folgendes Szenario abspielt:

- B ist ein Unterpunkt von A.
- C ist ein Unterpunkt von B.
- A ist ein Unterpunkt von C.

Weil A aber einen Unterpunkt hat, nämlich B, würde eine Abfrage nie zu einem Ergebnis kommen, sondern immer im Kreis laufen.

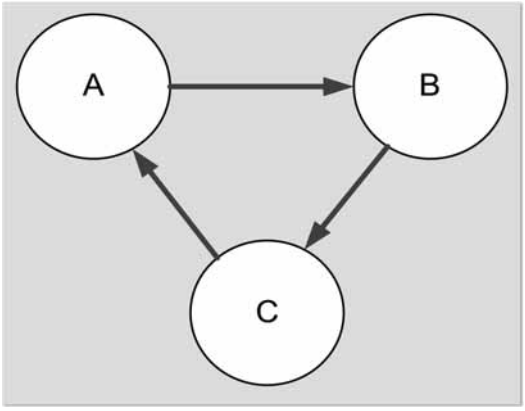


Abbildung 23.4 Endlosschleife von A nach B nach C

Solche Fehler müssen vermieden werden. Die einfachste Möglichkeit dafür ist, die Anzahl der Schleifendurchläufe zu ermitteln und nach einem einzigen Durchlauf zu stoppen. Ansonsten werden die Datensätze unendlich oft ausgegeben.

23.1.9 MySQL-Datentypen

Jedem einzelnen Datenfeld wie z. B. *name* muss ein bestimmter Datentyp zugeordnet werden. Dieser soll möglichst dem Inhalt des Feldes entsprechen. Die exakte Datendefinition ermöglicht es, die Datenbank nicht größer werden zu lassen als

nötig, da jeder Feldinhalt einen durch den Datentyp definierten Speicherplatz benötigt. Weiterhin werden Fehler durch nicht eindeutige Datentypen vermieden.

Mit der Definition des Datentyps wird etwa festgelegt, ob eine 1 als eine Zahl oder ein Zeichen behandelt werden soll. Wenn Sie ein Feld als Zahl definieren, können darin keine abweichenden Zeichen wie Buchstaben oder Symbole gespeichert werden.

MySQL unterstützt alle in Tabelle 23.1 dargestellten Datentypen.

Datentyp	Bedeutung
Zeichenformate	
TINYINT	ganze Zahl, 1 Byte
SMALLINT	ganze Zahl, 2 Byte
MEDIUMINT	ganze Zahl, 3 Byte
INT	ganze Zahl, 4 Byte
BIGINT	ganze Zahl, 5 Byte
FLOAT	Fließkommazahl, einfache Genauigkeit, 4 Byte
DOUBLE	Fließkommazahl, doppelte Genauigkeit, 8 Byte
DECIMAL	Festkommazahl, mit z. B. zwei Nachkommastellen
VARCHAR (n)	Zeichenkette mit fester Länge, n ist die Anzahl der Zeichen (maximal 255), flexibler Speicherbedarf je nach tatsächlicher Zeichenanzahl
CHAR (n)	Zeichenkette mit fester Länge, n ist die Anzahl der Zeichen (maximal 255), fester Speicherbedarf
TINYTEXT	Textfeld, 255 Zeichen
MEDIUMTEXT	Textfeld, 2^24-1 Zeichen
TEXT	Textfeld, 2^16-1 Zeichen
LONGTEXT	Textfeld, 2^32-1 Zeichen
Binäre Datentypen	
BLOB	binäre Datentypen, 2^16-1 Zeichen
TINYBLOB	binäre Datentypen, 255 Zeichen
MEDIUMBLOB	binäre Datentypen, 2^24-1 Zeichen
LONGBLOB	binäre Datentypen, 2^32-1 Zeichen

Tabelle 23.1 MySQL-Datentypen

Datentyp	Bedeutung
Zeit- und Datumsformate	
DATE	Zeitangabe, Format: JJJJ-MM-DD
TIME	Zeitangabe, Format: HH:MM:SS
TIMESTAMP	Dieses Feld wird beim Einfügen oder bei Änderungen im Datensatz automatisch gesetzt. Format: JJJJMMDDHHMMSS
DATETIME	Zeitangabe, Format: JJJJ-MM-DD HH:MM:SS
YEAR	Zeitangabe, Format: JJJJ
Aufzählungen	
ENUM	Aufzählungstyp, Format 'Wert1', 'Wert2' usw., ein definierter Wert
SET	Aufzählungstyp, Format 'Wert1', 'Wert2' usw., mehrere definierte Werte

Tabelle 23.1 MySQL-Datentypen (Forts.)

23.2 Erstellen einer MySQL-Datenbank

23.2.1 MySQL-Datenbank mit phpMyAdmin anlegen

[◉] Mit der nun folgenden kleinen Beispieldatenbank werden wir im Laufe der nächsten Kapitel einige Übungen durchführen. Wir empfehlen Ihnen, diese ebenfalls anzulegen oder einfach die SQL-Datei *testdatenbank.sql* zu importieren, die Sie auf der DVD im Ordner *Beispiele/Testdatenbank* finden.

[»] **Version beim Provider prüfen**

Achten Sie darauf, welche MySQL-Version Ihr Provider betreibt. Eventuell ist es besser, wenn Sie bei Ihrer lokalen Entwicklungsarbeit die gleiche MySQL-Version installieren. Auf *http://www.mysql.de* stehen alle alten Versionen zur Verfügung. Sie erleichtern sich damit die Arbeit enorm und müssen sich nicht um verschiedene Zeichencodierungen kümmern.

Bevor Sie mit einer Datenbank arbeiten können, müssen Sie eine Datenbank in MySQL anlegen. Starten Sie dazu Ihr MySQL-Administrationstool auf Ihrem lokalen Webserver. In unseren Beispielen arbeiten wir mit phpMyAdmin, Sie können diese Schritte aber auch beispielsweise mit dem MySQL Query Browser erledigen.

Die Version von phpMyAdmin

Wir arbeiten im Buch mit der Version 3.2.4. Diese wird bei der vorgeschlagenen XAMPP-Version automatisch installiert. Neuere oder auch ältere Versionen können eine etwas andere Benutzerführung und Oberfläche haben, unterscheiden sich aber in der Funktion nicht. Sie werden, auch wenn Ihr Provider eine andere Version anbietet, alle Funktionen wiederfinden.

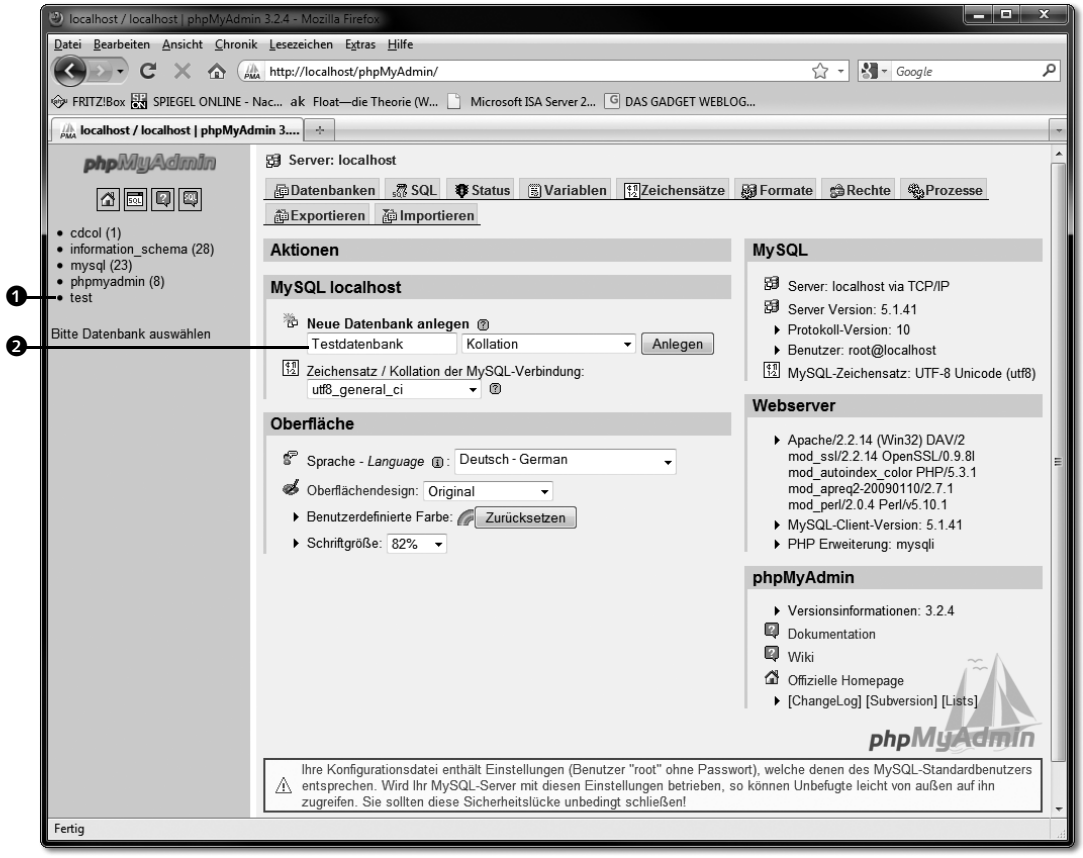


Abbildung 23.5 Datenbank mit phpMyAdmin anlegen

Legen Sie hier eine neue Datenbank mit dem Namen *Testdatenbank* ❷ an. Eventuell bestehende Datenbanken können Sie bearbeiten, wenn Sie diese im Menü ❶ auswählen.

Nachdem Sie den Namen eingegeben haben, klicken Sie auf ANLEGEN, und als Nächstes erscheint bereits das Dialogmenü zum Anlegen der Tabellen. Über das Abspeichern der Datenbank müssen Sie sich keine Gedanken machen. MySQL legt alle Daten im Verzeichnis */mysql/data* in separaten Ordnern ab.

Im Kopfbereich des nun folgenden Dokuments wird Ihnen der soeben ausgeführte SQL-Befehl angezeigt ❹. In diesem Fall ist dies CREATE DATABASE…, da Sie eine neue Datenbank angelegt haben.

Tragen Sie nun *Testtabelle* in das Feld für den Tabellennamen ❸ ein, und geben Sie vier Felder für die Tabelle vor ❺. Klicken Sie auf OK, und schon gelangen Sie zum Anlegen der Felder in den Tabellen.

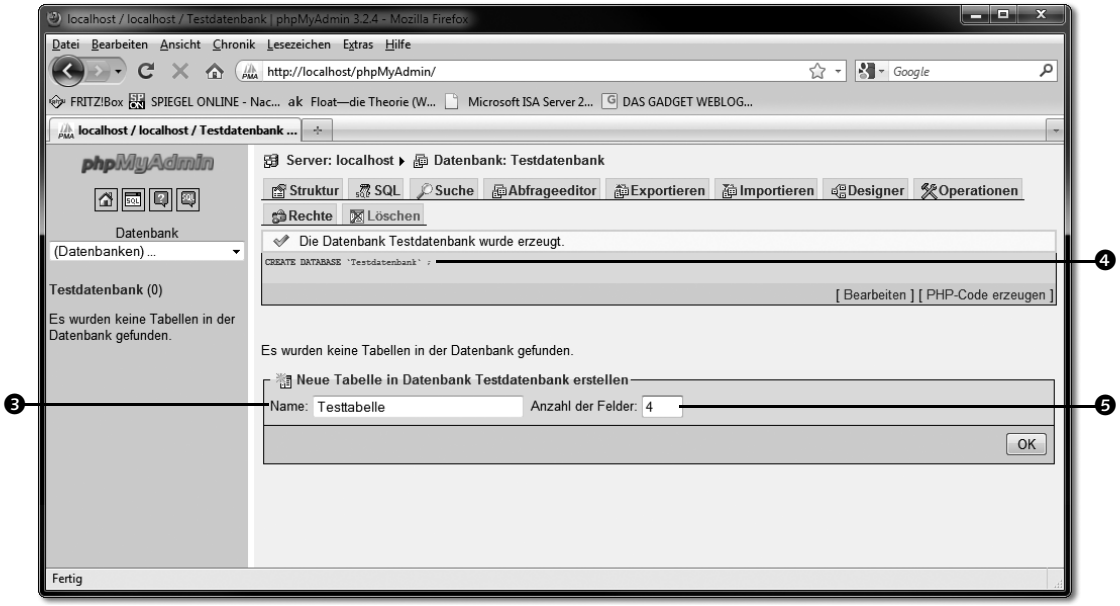


Abbildung 23.6 Anlegen der Tabelle mit phpMyAdmin

Felder anlegen

Gemäß Ihrer Eingabe stehen Ihnen jetzt vier Datenfelder zur Verfügung. Hier müssen nun die jeweiligen Datentypen eingetragen und einige Einstellungen vorgenommen werden. Im Folgenden werden wir nur die für Sie relevanten Einstellungen ansprechen. MySQL bietet in der vorliegenden Version 5 unzählige Einstellungen und Parameter an. Die meisten davon spielen für unsere Anwendung keine Rolle.

Den Namen eines Feldes tragen Sie unter Feld ❻ ein. Denken Sie daran, dass MySQL *case sensitiv* ist (Groß-/Kleinschreibung wird unterschieden) und in diesen Feldern keine Sonderzeichen akzeptiert.

Den Datentyp können Sie im Auswahlmenü unter Typ ❼ selektieren. Bei Datentypen, die eine Längenangabe voraussetzen, wird diese in das Feld Länge/Set ❽ eingetragen.



Abbildung 23.7 Anlegen der Felder mit phpMyAdmin

Etwas schwerer ist NULL ❿ zu erklären. Wenn ein Datenfeld leer ist oder den Wert 0 enthält, hat es keinen Inhalt oder eben den Wert 0. Der Eintrag NULL bedeutet wiederum, dass ein Datenfeld noch nicht vorhanden bzw. nicht bekannt ist. Manche Abfragen benötigen diese sogenannten *Null-Marken*.

Jedes Datenfeld kann einen Standardwert ❾ erhalten. Dieser wird dann automatisch beim Erzeugen eines neuen Datensatzes eingetragen, wenn es keine andere Eingabe gibt.

Unter A_I ⓬ legen Sie fest, ob ein Wert automatisch erzeugt wird. Diese Funktion sorgt dafür, dass unsere ID bei jedem neuen Datensatz um 1 erhöht und im Datensatz gespeichert wird. Die Einstellung für automatisches Hochzählen des Feldwertes ist AUTO_INCREMENT. Für den PRIMÄRSCHLÜSSEL ⓫ ist dies zwingend erforderlich. Wenn Sie alles eingetragen haben, klicken Sie auf SPEICHERN, und Ihre erste Tabelle ist fertig.

Wenn Sie alte phpMyAdmin-Versionen kennen, werden Sie die Menüführung der vorliegenden Version etwas seltsam finden. Man gewöhnt sich jedoch sehr schnell daran.

Falls Sie diese Arbeiten das erste Mal machen, lassen Sie sich nicht von den vielen Möglichkeiten, die Ihnen diese Datenbank bietet, ins Bockshorn jagen. Es ist alles viel einfacher und strukturierter, als es vielleicht aussieht.

Klicken Sie nun auf EINFÜGEN ⓭ (siehe Abbildung 23.8), um die Datenbank mit ersten Inhalten zu befüllen.

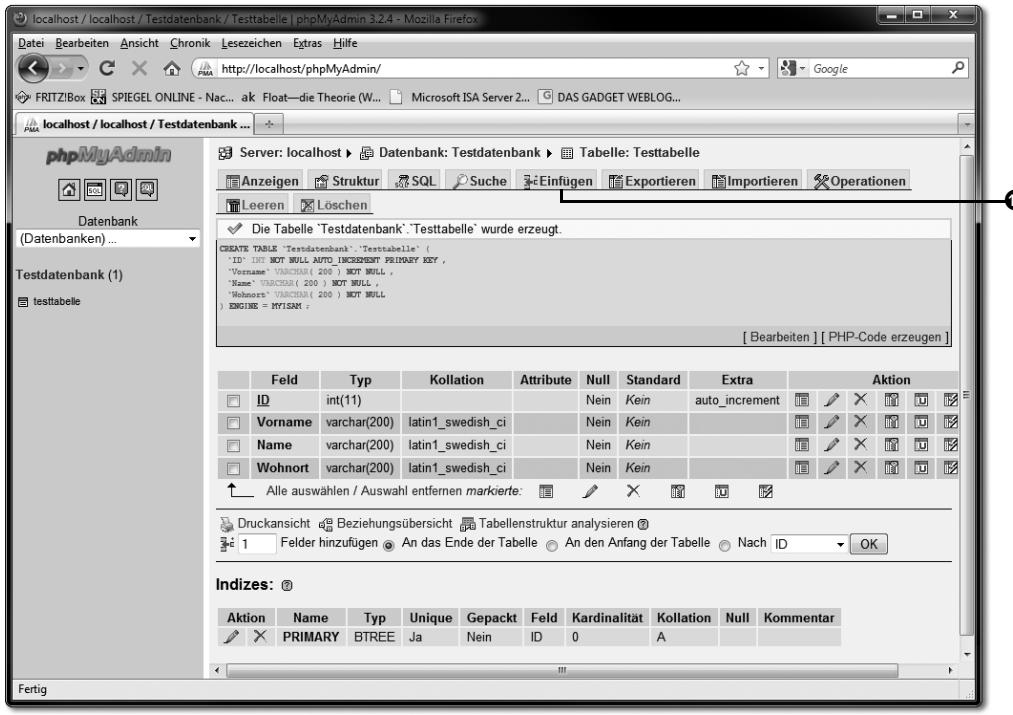


Abbildung 23.8 Die erste Tabelle ist fertig erstellt.

23.2.2 Einfügen von Datensätzen

Fügen Sie an dieser Stelle zwei oder mehr Datensätze in die Tabelle *Testtabelle* ein. Diese Datensätze benötigen wir, um in Dreamweaver bei einer Abfrage überhaupt eine Anzeige zu erhalten.

Achten Sie jedoch darauf, das Feld ID unbedingt freizulassen. Dieser Wert wird automatisch gesetzt, da wir AUTO_INCREMENT ausgewählt hatten.

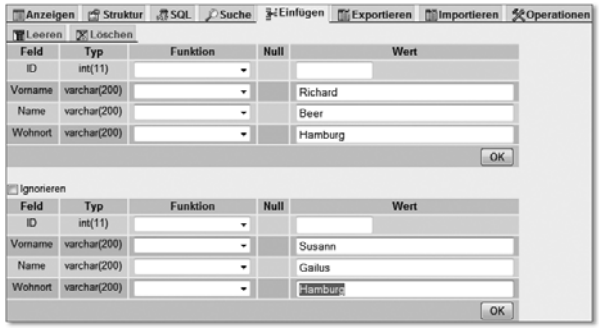


Abbildung 23.9 Einfügen eines Datensatzes

Klicken Sie anschließend im Hauptbildschirm auf ANZEIGEN ❶, um die Dateneingabe zu überprüfen.

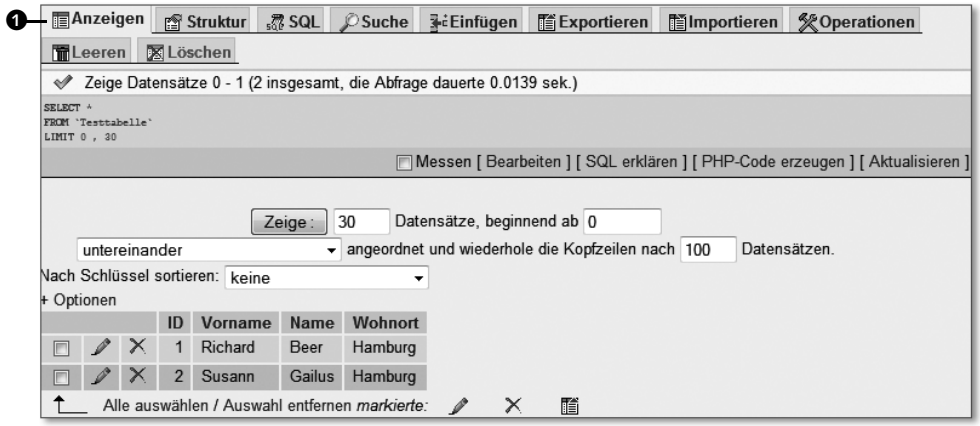


Abbildung 23.10 Dateneingabe überprüfen

23.2.3 Import von Daten

Nachdem Sie eine Datenbank erstellt haben, können Sie die Datenstruktur einer anderen Datenbank und ihre Inhalte komplett importieren. Wählen Sie dazu im Menü von phpMyAdmin IMPORTIEREN aus, und klicken Sie dann auf DURCHSUCHEN, um die SQL-Datei auf den Webserver zu laden.

SQL-Dateien finden Sie häufig bei Open-Source-Projekten wieder, um Ihnen einen Import der Datenbankstrukturen zu ermöglichen.

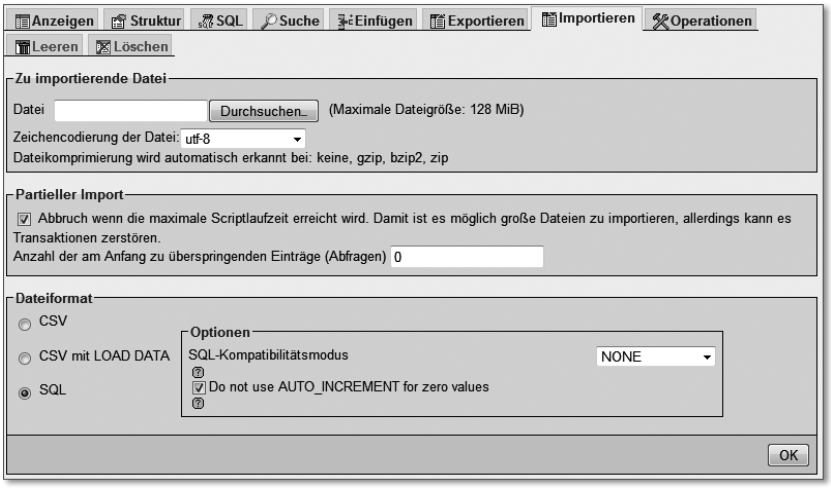


Abbildung 23.11 SQL-Import mit phpMyAdmin

SQL-Dateien eignen sich auch, um Daten von Ihrem Webserver zu sichern. Besonders vor Änderungen am Datenbanksystem sollten Sie eine Sicherung abspeichern.

23.2.4 Export von Daten zum Provider

Wenn Sie Ihre dynamische Website lokal programmiert haben, müssen der Inhalt und die Struktur der Datenbank auf den reellen Webserver bei Ihrem Provider übertragen werden.

Klicken Sie dazu im phpMyAdmin-Menü auf EXPORTIEREN ❶. Aktivieren Sie anschließend noch SENDEN ❷, um die SQL-Datei (den sogenannten *SQL-Dump*) lokal zu speichern. Um weitergehende Informationen zu erhalten, lesen Sie bitte die Dokumentation von phpMyAdmin.

Wenn Ihr Provider MySQL 4 einsetzt, was durchaus noch häufig der Fall ist, müssen Sie unbedingt den SQL-Kompatibilitätsmodus MYSQL40 wählen.

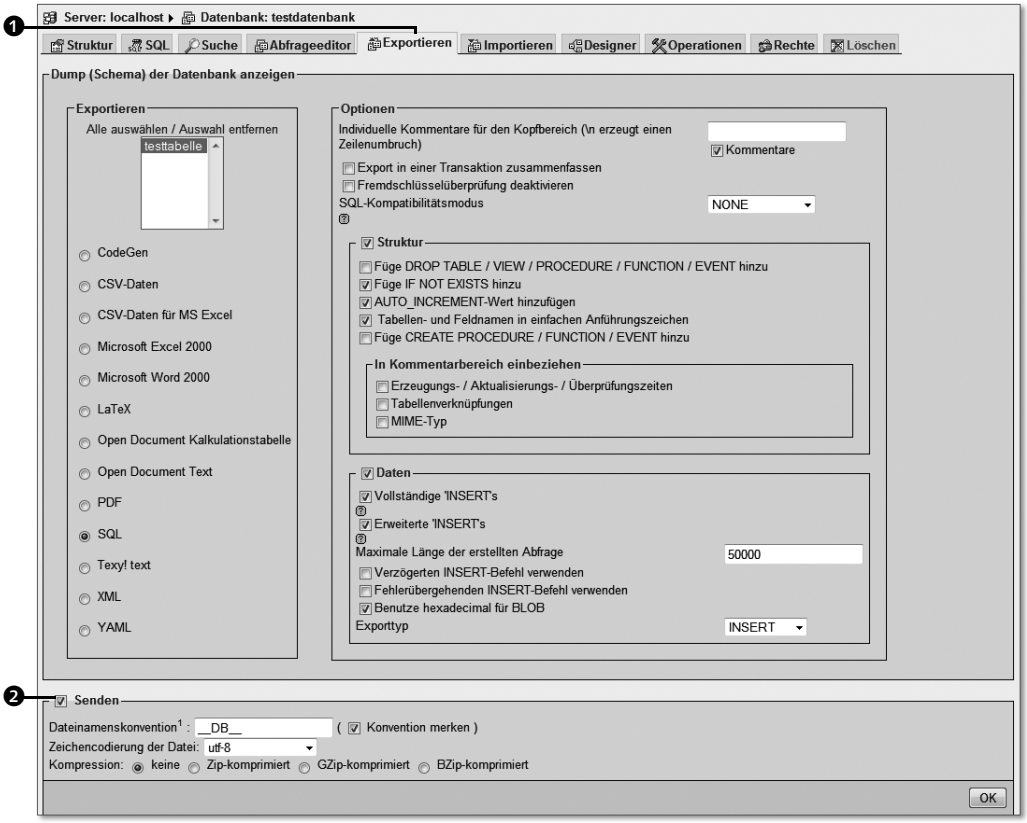


Abbildung 23.12 SQL-Export mit phpMyAdmin

Dateigröße beschränkt [«]

Über das Internet können Sie, je nach Verbindung, maximal 2 bis 3 MByte große SQL-Dateien hochladen. Wenn Sie größere Dateien haben, können Sie diese nur direkt auf der Kommandozeile des Webservers per SSH importieren. Wenden Sie sich in diesem Fall an Ihren Administrator oder Provider.

23.3 Benutzerverwaltung mit MySQL

Wenn Sie MySQL zum ersten Mal starten, ist als Benutzer immer *root* (oder auch *root@localhost*) ohne ein Kennwort eingetragen. Sollten Sie selbst einen Webserver betreiben, müssen Sie diese Benutzerdaten unbedingt ändern, da ansonsten jeder Zugriff auf Ihre Datenbanken hat.

Auch bei der Rechtevergabe verwenden wir wieder phpMyAdmin als Oberfläche. Sie können sie aber auch in ähnlicher Form mit den grafischen Oberflächen anderer Tools für MySQL vornehmen.

Auch lokal Zugangsdaten vom Provider eintragen [«]

Wenn Sie Ihre Websites bei einem Provider hosten, werden in den meisten Fällen die Namen der Datenbanken und die Benutzernamen automatisch vergeben, und Sie haben keinen direkten Einfluss darauf. Es ist ziemlich umständlich, lokal mit anderen Zugangsdaten und Datenbanknamen als später im Internet zu arbeiten. Außerdem ist es eine zusätzliche Fehlerquelle, wenn diese Daten beim lokalen Arbeiten immer wieder geändert werden müssen.

Sinnvoll ist es daher, die vom Provider erhaltenen Daten auch in Ihren lokalen Datenbanken zu verwenden. Legen Sie einen neuen Benutzer in MySQL und die Datenbanken mit den vom Provider vergebenen Daten an, und Sie müssen beim Veröffentlichen der Website und bei nachträglichen Änderungen nichts mehr aktualisieren.

Rechtevergabe über phpMyAdmin

Um die Benutzerverwaltung in phpMyAdmin aufzurufen, klicken Sie im Startfenster auf Rechte. Sie kommen nun in die Benutzerübersicht von MySQL und sehen, welche Zugriffsrechte jeder einzelne Benutzer hat.

Benutzer löschen

Für das Löschen von Benutzern gibt es in phpMyAdmin einige komplexe Optionen. Um diese zu verstehen, müssen Sie wissen, dass alle Benutzer in einer Tabelle stehen, die beim Hochfahren von MySQL geladen wird. In diesem Menü haben Sie nach dem Aktivieren des Kontrollkästchens für ausgewählte Benutzer die folgenden Möglichkeiten:

1. Der Benutzer wird gelöscht, behält aber bis zu einem erneuten Laden der Benutzertabelle alle Rechte und kann weiterhin auf dem System arbeiten. Der Benutzer wird demzufolge nicht sofort gelöscht.
2. Dem Benutzer werden alle Rechte entzogen, er behält allerdings die Zugriffsrechte im schreibgeschützten Modus, bis die Tabelle neu geladen wird.
3. Der Benutzer wird vollständig aus der Tabelle entfernt und die Tabelle direkt im Anschluss neu geladen. Dies ist der sicherste Weg.
4. Datenbanken, die denselben Namen wie der Benutzer tragen, werden ebenfalls sofort gelöscht. Wenden Sie diese Option nur mit äußerster Vorsicht an. Es ist besser, alle relevanten Datenbanken anschließend von Hand zu löschen.



Abbildung 23.13 Die Rechteverwaltung von phpMyAdmin für MySQL

Benutzer anlegen

Beim Anlegen eines neuen Benutzers müssen Sie für diesen – nachdem Sie Benutzername, Host und Passwort vergeben haben – Globale Rechte festlegen. In der folgenden Tabelle haben wir die Bedeutung der einzelnen Optionen aufgelistet.

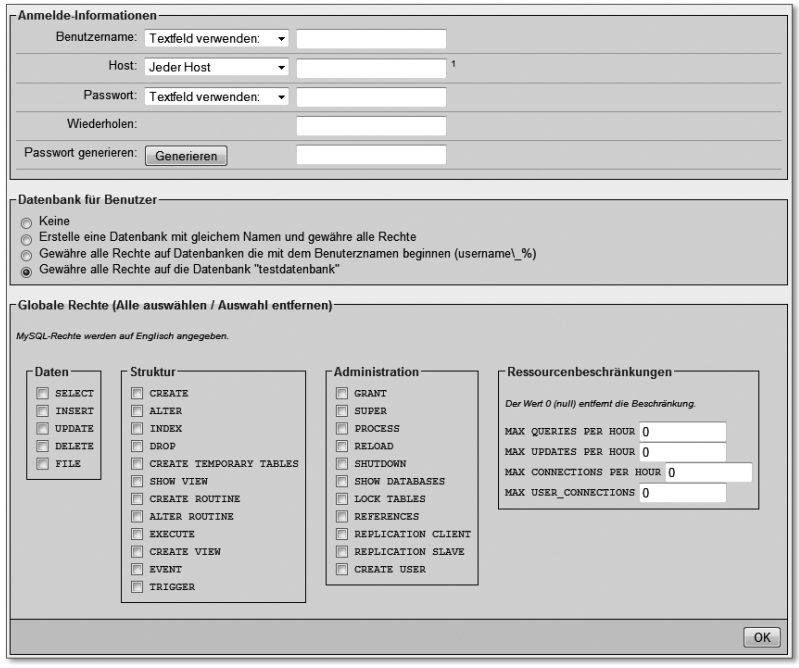


Abbildung 23.14 Neuen Benutzer mit phpMyAdmin anlegen

Befehl	Funktion
Daten	
SELECT	Daten auslesen
INSERT	Daten einfügen
UPDATE	Daten verändern oder überschreiben
DELETE	Daten löschen
FILE	Daten importieren oder exportieren
Struktur	
CREATE	Erstellen kompletter Datenbanken oder Tabellen
ALTER	Struktur von Tabellen verändern
INDEX	Indizes von Tabellen anlegen oder löschen
DROP	Datenbanken oder Tabellen komplett löschen
Administration	
GRANT	Hinfügen oder Verändern von Benutzern in der Benutzertabelle während des Betriebs
PROCESS	Beenden systemfremder Prozesse
RELOAD	Servereinstellungen neu laden und während des Betriebs den Zwischenspeicher von MySQL leeren
SHUTDOWN	Vollständiges Herunterfahren des MySQL-Servers

Tabelle 23.2 Globale Rechtevergabe für MySQL-Benutzer

23.4 MySQL und Microsoft Access

Viele Benutzer sind beim Arbeiten mit Datenbanken die Oberfläche von Microsoft Access gewohnt. MySQL-Tabellen können auch bequem mit Microsoft Access bearbeitet werden. Im folgenden Abschnitt arbeiten wir mit Access 2007. In anderen Versionen ist die Vorgehensweise die gleiche, allerdings können sich die Menüs und Dialogboxen leicht unterscheiden.

Besonders interessant ist es, Tabellen aus MySQL anzubinden und andere Datenbankbestandteile über Access abzuwickeln. So können Sie beispielsweise einen Onlineshop komplett mit PHP und MySQL erstellen und diese Tabellen für die Fakturierung an eine Access-Datenbank binden.

Datenaustausch per ODBC

Um Daten zwischen verschiedenen Datenbanksystemen auszutauschen, benötigen Sie einen *ODBC*-Treiber (*Open Database Connectivity*). MySQL stellt diesen auf der Website *http://www.mysql.de* kostenlos zur Verfügung. Auf der DVD zum Buch finden Sie im Verzeichnis *Webserver/MySQL* die Version *MyODBC-5.1*.

Installieren Sie den ODBC-Treiber einfach durch einen Doppelklick auf das Programmsymbol. Weitere Einstellungen werden nicht benötigt.

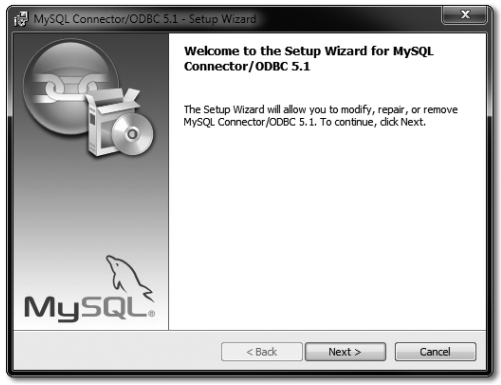


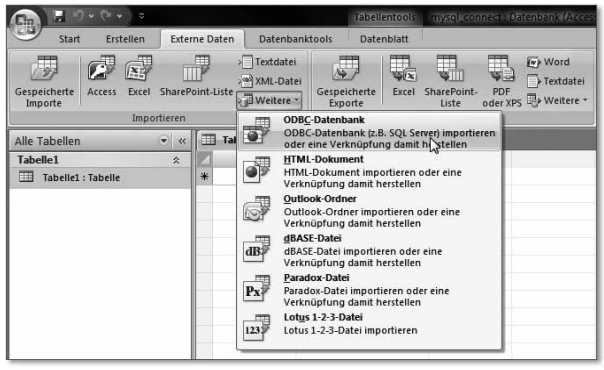
Abbildung 23.15 ODBC Connector installieren

Schritt-für-Schritt: MySQL-Tabellen an Microsoft Access 2007 (oder andere Versionen) binden

1 *Neue Datenbank in Access*

Bevor Sie MySQL-Tabellen an Access binden, müssen Sie in Access eine neue, leere Datenbank erstellen und abspeichern.

Gehen Sie dann auf DATEI • EXTERNE DATEN • ODBC-DATENBANK.



2 *ODBC-Datenbanken als Verknüpfungsziel auswählen*

In der nun folgenden Dialogbox wählen Sie als Verknüpfungsziel ERSTELLEN SIE EINE VERKNÜPFUNG aus.



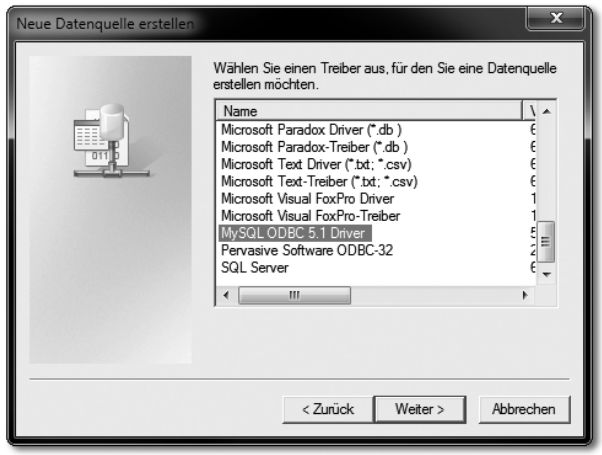
3 *Neue Datenquelle erstellen*

Um Access an MySQL-Tabellen zu binden, müssen Sie nun zunächst eine neue Datenquelle erstellen. Klicken Sie in der Dialogbox DATENQUELLE AUSWÄHLEN auf COMPUTERDATENQUELLE ❶, und legen Sie eine neue BENUTZERDATENQUELLE ❷ an.



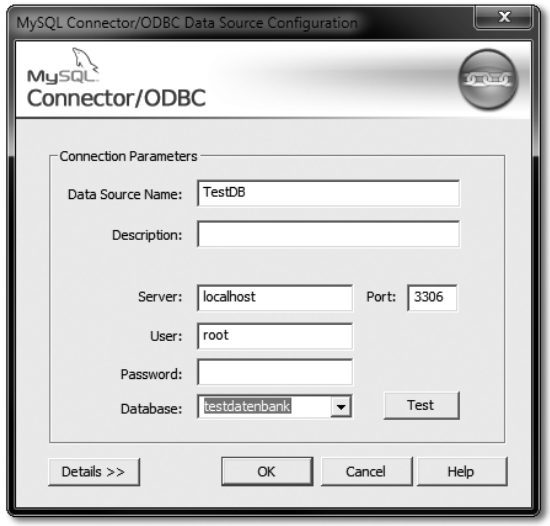
4 *ODBC-Treiber auswählen*

Wenn Sie den ODBC-Treiber für MySQL installiert haben, erscheint er in der Auswahl der Dialogbox NEUE DATENQUELLE ERSTELLEN. Wählen Sie ihn aus, und klicken Sie dann auf WEITER.



5 *DSN-Konfiguration in MySQL ODBC*

Um jetzt die Datenbankverbindung herzustellen, muss MySQL gestartet werden. Vergeben Sie dann in der DSN-Konfiguration einen Namen für die Verbindung zu MySQL, und tragen Sie die Zugangsdaten ein. Testen Sie die Verbindung, indem Sie auf TEST klicken.



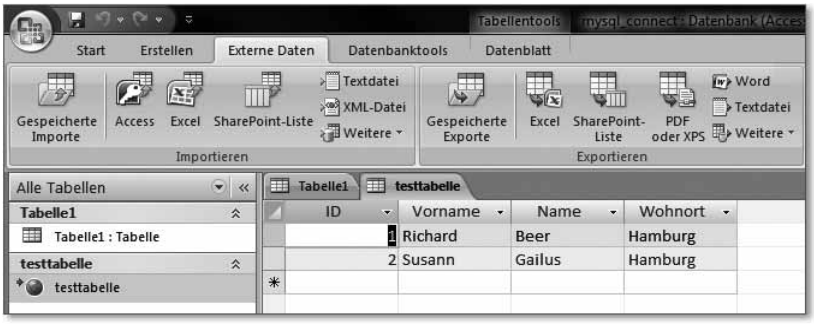
6 *Ausgewählte Tabellen in Access verknüpfen*

Access fordert Sie nun auf, die Tabellen für die Verknüpfung auszuwählen. Selektieren Sie alle, die Sie mit Access verknüpfen wollen, und klicken Sie dann auf OK.



7 *Anzeige in Access*

Die verknüpften Tabellen werden nun in Access angezeigt.



Jetzt können Sie in Access Daten in diese Tabellen einfügen, weitere Felder anlegen oder Abfragen zwischen den MySQL- und den Access-Tabellen erstellen. ■

23.5 Einführung in SQL

SQL ist die Standard-Abfragesprache für Datenbanken (*Structured Query Language*, strukturierte Abfragesprache). In diesem Abschnitt erfahren Sie alles Notwendige, um in Dreamweaver damit arbeiten zu können.

Sie haben Ihre MySQL-Datenbank angelegt und Datensätze darin eingegeben. Mit SQL können Sie nun die Abfragen durchführen, mit denen Sie die gewünschten Informationen aus der Datenbank erhalten.

SQL ist eine standardisierte Abfragesprache, doch leider benutzt mittlerweile jedes Datenbanksystem eigene Definitionen von SQL und benötigt oft eine leicht abgewandelte Syntax. Das Grundprinzip und die wichtigsten Befehle bleiben jedoch identisch. Wir werden später die SQL-Abfragen mit PHP als Zeichenketten an MySQL übertragen und die Rückgabewerte anschließend wieder mit PHP auswerten.

23.5.1 Daten abfragen mit »SELECT«

Im einfachsten Fall lautet eine SQL-Abfrage:

```
SELECT * FROM Tabelle
```

Im übertragenen Sinn heißt dies:

Hole (`SELECT`) alles (`*`) aus (`FROM`) der Tabelle `Tabelle`.

Diese Abfrage würde alle Inhalte der Tabelle `Tabelle` ausgeben. Die Analyse der Daten erfolgt dann anschließend mit einem weiterführenden Skript.

Für unsere Buchwebsite könnte eine Abfrage lauten:

```
SELECT * FROM 2_0
```

Damit würden alle Inhalte der Tabelle `2_0` als Rückgabewert ausgegeben werden. Welche dieser Daten dann wirklich zur Anzeige auf unsere Website kommen, ist zunächst egal.

Abfragen spezifizieren

Um nur eine einzelne Spalte einer Tabelle abzufragen, können Sie den Stern durch den Namen der Spalte ersetzen. Falls Sie mit MS-SQL (Transact-SQL) arbeiten, ist dies sogar zwingend vorgeschrieben.

Eine Abfrage hat dann das folgende Format:

```
SELECT name FROM 2_0
```

Diese Abfrage holt nur die Inhalte der Spalte `name` aus der Tabelle `2_0`. Auf diese Weise können auch mehrere Spalten angegeben werden. Diese sind durch Kommata zu trennen:

```
SELECT name, vorname FROM 2_0
```

Mit dieser Abfrage erhalten Sie als Rückgabewerte die beiden Spalten `name` und `vorname`.

23.5.2 Datensatz abfragen mit »WHERE«

Um einzelne Datensätze aus einer Tabelle abzufragen, muss ein weiterer Parameter hinzugefügt werden:

```
SELECT * FROM 2_0 WHERE ID = 1
```

Diese Abfrage gibt den Datensatz mit der ID `1` aus der Tabelle `2_0` zurück. Da Sie mit PHP Variablen anstelle von festen Werten in eine SQL-Abfrage setzen können, kann auch ein ausgewählter Datensatz zur Anzeige gebracht werden.

```
SELECT * FROM 2_0 WHERE ID =".$_GET['CONT_ID']
```

In dieser Abfrage wird die ID durch eine mit der Methode `GET` übertragene Variable `CONT_ID` vorgegeben.

23.5.3 Daten sortieren mit »ORDER BY«

Ausgegebene Datensätze können mit einem weiteren Parameter sortiert ausgegeben werden:

```
SELECT * FROM 2_0 ORDER BY name DESC
```

Diese Abfrage gibt die Daten der Tabelle `2_0` sortiert nach `name` in alphabetisch absteigender Reihenfolge aus. `DESC` kann durch `ASC` ersetzt werden. Die Sortierung erfolgt dann aufsteigend.

23.5.4 Datensatz aktualisieren mit »UPDATE«

Mit dem Befehl `UPDATE` können Sie bestehende Datensätze überschreiben bzw. aktualisieren.

```
UPDATE 2_0 SET name = 'Neuer Name' WHERE ID =".$_GET['COND_ID']
```

Dieser Befehl überschreibt in der Tabelle `2_0` in der Spalte `name` den bisherigen Inhalt mit `Neuer Name`. Ausgewählt wird der Datensatz durch `WHERE` und die ID. Diese wird in unserem Fall durch die mit PHP übertragene Variable `CONT_ID` gesetzt.

23.5.5 Datensatz löschen mit »DELETE«

Der Befehl

```
DELETE FROM 2_0 WHERE ID =".$_GET['COND_ID']
```

löscht aus der Tabelle `2_0` den kompletten Datensatz (Tupel) mit der übergebenen ID. Seien Sie überaus vorsichtig mit Löschaktionen – eine Funktion zum Wiederherstellen der Datensätze gibt es nicht. Einmal gelöscht ist der Datensatz für immer verloren.

23.5.6 Datensatz einfügen mit »INSERT INTO«

Beim Einfügen von Datensätzen mit dem Befehl `INSERT INTO` müssen alle Daten der Reihe nach übergeben werden.

```
INSERT INTO 2_0 (name, vorname, URL) VALUES ('Neuer Name',
'Neuer Vorname', 'Website')
```

Im ersten Teil des Befehls werden in die Tabelle `2_0` die Felder `name`, `vorname` und `URL` mit Inhalten gefüllt.

Im zweiten Teil ab `VALUES` folgen die einzelnen Inhalte (Werte) in derselben Reihenfolge. Dieser Befehl wird, wenn ein Datensatz sehr viele Felder enthält, ausgesprochen lang und fehleranfällig. Überprüfen Sie diese Aktionen daher sehr genau!

23.6 Datenstruktur der Buchwebsite


Buchwebsite

Bevor weitere Arbeitsschritte an der Buchwebsite durchgeführt werden können, gilt es, eine Datenstruktur zu schaffen, die in der Lage ist, die gewünschten Ergebnisse abzubilden.

23.6.1 Anforderungen

Wir benötigen für unsere Datenbank sechs verschiedene Bereiche:

- *News*
- *Art*
- *Photography*
- *Design*
- *Illustration*
- *TV/Broadcast*

Diese Bereiche sind den Dokumenten *1_0 / 1_1* bis *6_0 / 6_1* zugeordnet. _0 sind dabei jeweils die Übersichtsseiten, _1 die Detailseiten. Die Zuordnung können wir in den Tabellennamen direkt übernehmen. Wir benötigen demnach sechs verschiedene Tabellen:

- *Tabelle 1_0*
- *Tabelle 2_0*
- *Tabelle 3_0*
- *Tabelle 4_0*

- *Tabelle 5_0*
- *Tabelle 6_0*

Auf der Website sollen verschiedene Informationen dargestellt werden. Für die Künstlereinträge sind das:

- *Vorname*
- *Nachname*
- *Website*
- *maximal zehn verschiedene Bilder*
- *Vita oder Beschreibung*

23.6.2 Primärschlüssel festlegen

Wir benötigen noch für jede Tabelle ein weiteres Feld, das den Primärschlüssel enthält. Wir nennen es:

- *ID*

Daraus ergeben sich für unsere Datenbank folgende Datentypen:

- *Vorname* = VARCHAR(100)
- *Nachname* = VARCHAR(100)
- *Website* = VARCHAR(100)
- *maximal zehn verschiedene Bilder* = VARCHAR(100)
- *Vita oder Beschreibung* = TEXT
- *ID* = INT und Primärschlüssel

Für den Bereich *News* benötigen wir etwas andere Angaben:

- *Überschrift* = VARCHAR(100)
- *Datum* = DATE
- *Beschreibung* = TEXT
- *ID* = INT und Primärschlüssel

Wie Sie sehen, haben wir für die Bilder VARCHAR(100) gewählt, da wir sie nicht in der Datenbank ablegen, sondern in der Datenbank nur einen Link zu einem Bild speichern.

Das vollständige Schema unserer Datenbank sehen Sie in Abbildung 23.16.

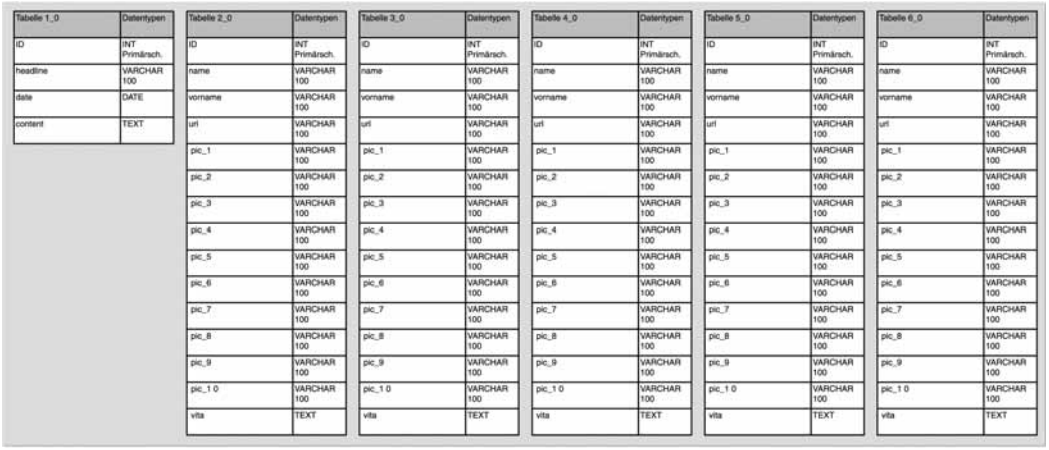


Tabelle 1_0	Datentypen
ID	INT Primärsch.
headline	VARCHAR 100
date	DATE
content	TEXT

Tabelle 2_0	Datentypen
ID	INT Primärsch.
name	VARCHAR 100
vorname	VARCHAR 100
url	VARCHAR 100
pic_1	VARCHAR 100
pic_2	VARCHAR 100
pic_3	VARCHAR 100
pic_4	VARCHAR 100
pic_5	VARCHAR 100
pic_6	VARCHAR 100
pic_7	VARCHAR 100
pic_8	VARCHAR 100
pic_9	VARCHAR 100
pic_1 0	VARCHAR 100
vita	TEXT

Tabelle 3_0	Datentypen
ID	INT Primärsch.
name	VARCHAR 100
vorname	VARCHAR 100
url	VARCHAR 100
pic_1	VARCHAR 100
pic_2	VARCHAR 100
pic_3	VARCHAR 100
pic_4	VARCHAR 100
pic_5	VARCHAR 100
pic_6	VARCHAR 100
pic_7	VARCHAR 100
pic_8	VARCHAR 100
pic_9	VARCHAR 100
pic_1 0	VARCHAR 100
vita	TEXT

Tabelle 4_0	Datentypen
ID	INT Primärsch.
name	VARCHAR 100
vorname	VARCHAR 100
url	VARCHAR 100
pic_1	VARCHAR 100
pic_2	VARCHAR 100
pic_3	VARCHAR 100
pic_4	VARCHAR 100
pic_5	VARCHAR 100
pic_6	VARCHAR 100
pic_7	VARCHAR 100
pic_8	VARCHAR 100
pic_9	VARCHAR 100
pic_1 0	VARCHAR 100
vita	TEXT

Tabelle 5_0	Datentypen
ID	INT Primärsch.
name	VARCHAR 100
vorname	VARCHAR 100
url	VARCHAR 100
pic_1	VARCHAR 100
pic_2	VARCHAR 100
pic_3	VARCHAR 100
pic_4	VARCHAR 100
pic_5	VARCHAR 100
pic_6	VARCHAR 100
pic_7	VARCHAR 100
pic_8	VARCHAR 100
pic_9	VARCHAR 100
pic_1 0	VARCHAR 100
vita	TEXT

Tabelle 6_0	Datentypen
ID	INT Primärsch.
name	VARCHAR 100
vorname	VARCHAR 100
url	VARCHAR 100
pic_1	VARCHAR 100
pic_2	VARCHAR 100
pic_3	VARCHAR 100
pic_4	VARCHAR 100
pic_5	VARCHAR 100
pic_6	VARCHAR 100
pic_7	VARCHAR 100
pic_8	VARCHAR 100
pic_9	VARCHAR 100
pic_1 0	VARCHAR 100
vita	TEXT

Abbildung 23.16 Datenfelder mit Datentypen in unserer MySQL-Datenbank

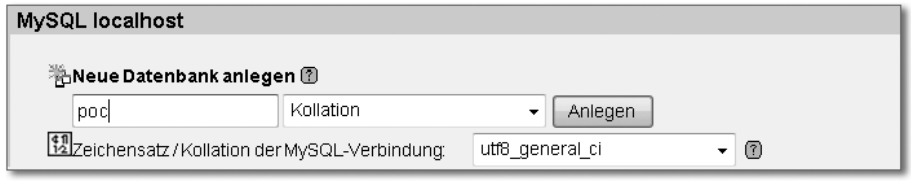
[o] Wenn Sie diese Struktur nicht von Hand anlegen möchten, was wir Ihnen jedoch aus Übungszwecken sehr empfehlen, können Sie die gesamte Struktur auch importieren. Wir haben Ihnen dazu die SQL-Datei im Ordner *Buchwebsite/SQL* mit auf die DVD des Buches gepackt.

[▪] *Schritt-für-Schritt: Die Datenbankstruktur importieren*

Gehen Sie wie folgt vor:

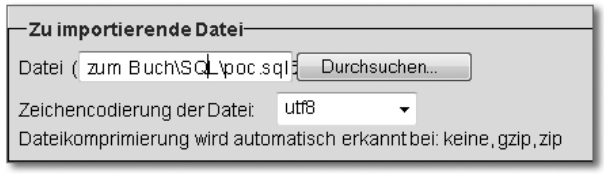
1 *Datenbank anlegen*

Legen Sie mit phpMyAdmin eine neue Datenbank mit dem Namen *poc* oder einem anderen von Ihnen gewünschten Namen an. Mit diesem Datenbanknamen müssen Sie in den folgenden Kapiteln arbeiten.



2 *SQL-Dump installieren*

[o] Importieren Sie die Datei *poc.sql* aus dem DVD-Verzeichnis *Buchwebsite/SQL*.



3 *Komplette Struktur und Testdaten*

Wie Sie in der folgenden Abbildung sehen, wird die gesamte Datenbankstruktur angelegt. In der importierten SQL-Datei sind auch bereits einige Beispieldatensätze enthalten.

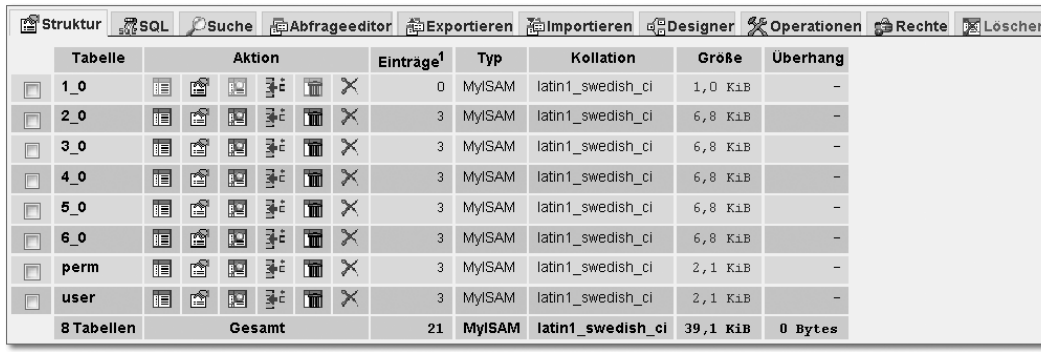


	Tabelle	Aktion	Einträge[1]	Typ	Kollation	Größe	Überhang
☐	1_0		0	MyISAM	latin1_swedish_ci	1,0 KiB	-
☐	2_0		3	MyISAM	latin1_swedish_ci	6,8 KiB	-
☐	3_0		3	MyISAM	latin1_swedish_ci	6,8 KiB	-
☐	4_0		3	MyISAM	latin1_swedish_ci	6,8 KiB	-
☐	5_0		3	MyISAM	latin1_swedish_ci	6,8 KiB	-
☐	6_0		3	MyISAM	latin1_swedish_ci	6,8 KiB	-
☐	perm		3	MyISAM	latin1_swedish_ci	2,1 KiB	-
☐	user		3	MyISAM	latin1_swedish_ci	2,1 KiB	-
	8 Tabellen	Gesamt	21	MyISAM	latin1_swedish_ci	39,1 KiB	0 Bytes

Um Datenbankinhalte in eine dynamische Website einzubinden, müssen Sie zunächst eine Verbindung zu dieser herstellen. Hier erfahren Sie, wie das mit Dreamweaver funktioniert.

24 Datenbanken mit Dreamweaver anbinden

Sie haben jetzt Ihre erste Datenbank angelegt, und nun wollen wir diese mit Dreamweaver abfragen. Dazu muss zunächst mit Dreamweaver eine Verbindung zur Datenbank hergestellt werden.

24.1 MySQL-Datenbanken konnektieren

Wie bereits erwähnt, sind MySQL und der Webserver zwei unabhängige Anwendungen. Damit auf einer Internetseite Inhalte aus einer Datenbank dargestellt werden können, muss eine Verbindung zur MySQL-Datenbank hergestellt werden. Über diese Verbindung (*Konnektierung*) werden dann die Abfragen ausgeführt und die Rückgabewerte ausgetauscht.

Für diese Konnektierung benötigen Sie den Namen der Datenbank, den Servernamen (meistens *localhost*) und die Zugangsdaten.

Datenbankverbindung in externer Datei

Bei jedem Aufruf einer Website mit Datenbankabfrage wird eine Verbindung zur Datenbank hergestellt, eine Abfrage gesendet, und die erhaltenen Daten werden ausgewertet. Die Konnektierung zu einer Datenbank definiert man möglichst nicht innerhalb eines Dokuments, sondern in einer externen Datei und bindet diese nach Bedarf ein. Änderungen der Zugangsdaten können dann zentral erfolgen.

Dreamweaver erstellt eine Datenbankverbindung ebenfalls in externen Dateien, die nach Bedarf in die Hauptdokumente eingebunden werden.

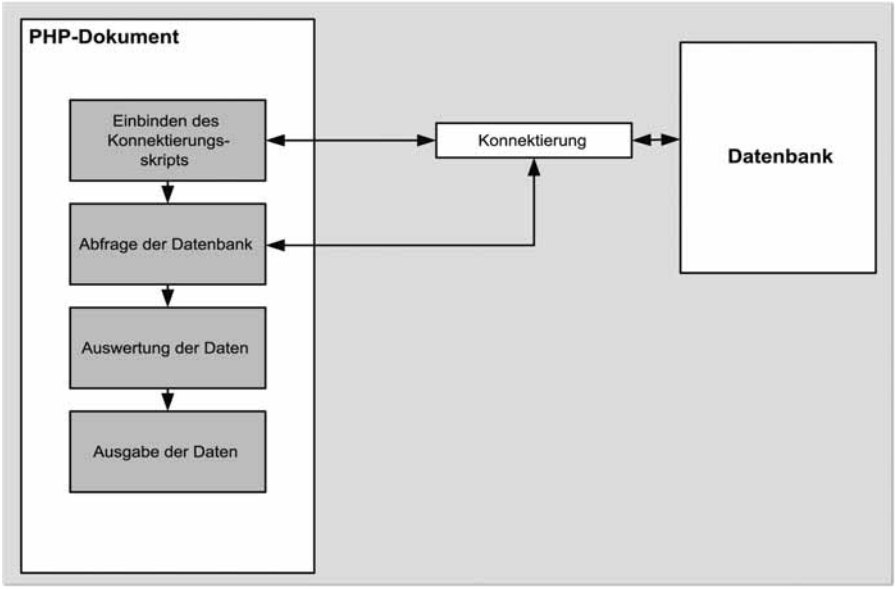


Abbildung 24.1 Prinzip einer Datenbankanbindung mit PHP

24.2 MySQL mit PHP konnektieren

Um in Dreamweaver eine Datenbankverbindung zu erstellen, müssen Sie ein leeres PHP-Dokument anlegen oder ein bestehendes öffnen. Datenbankverbindungen können nur bei geöffneten, dynamischen Dokumenten angelegt werden.

24.2.1 Bedienfeldgruppe »Anwendung«

Alle Datenbankaktionen sind in der Bedienfeldgruppe Anwendung (siehe Abbildung 24.2) zusammengefasst. Zum Anlegen einer Datenbankverbindung müssen Sie nach einem Klick auf das Plussymbol MySQL-Verbindung auswählen.

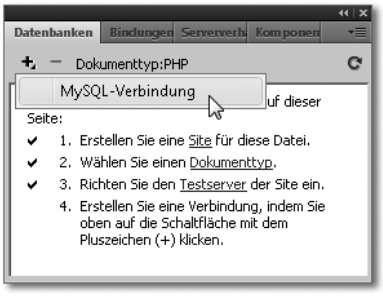


Abbildung 24.2 MySQL-Verbindung einrichten

In der folgenden Dialogbox müssen Sie im Feld VERBINDUNGSNAME einen Namen für Ihre Verbindung angeben. In den meisten Fällen wird der MYSQL-SERVER *localhost* heißen. Dies trifft auch zu, wenn Sie Ihre Skripte zu einem späteren Zeitpunkt auf den Webserver des Providers übertragen.



Abbildung 24.3 Verbindungsdaten eingeben

Es gibt jedoch Ausnahmen. Bei manchen Providern werden die Datenbanken auf einem separaten Server gehostet. In diesem Fall müssen Sie die erforderlichen Daten bei Ihrem Dienstleister erfragen.

Der BENUTZERNAME *root* ist der Standardbenutzer von MySQL. Wenn Sie lokal arbeiten und keine Benutzereinstellungen an MySQL vorgenommen haben, müssen Sie kein Kennwort eingeben. Wenn Sie über andere Zugangsdaten verfügen, können Sie sie in der Dialogbox eintragen.

Zugangsdaten beim Provider [«]

Bedenken Sie, dass die von Ihnen bei der Konnektierung zur Datenbank eingetragenen Daten auf Ihrem System funktionieren, aber nicht bei Ihrem Provider. Beim Veröffentlichen der Website müssen Sie die Zugangsdaten Ihres Providers in die entsprechende Dialogbox eintragen. Eine Alternative wäre, dass Sie Ihre lokalen Datenbanken an die Vorgaben des Providers im Hinblick auf Benennung und Benutzer anlegen. Sie sparen sich damit das Ändern der Zugangsdaten.

Wenn alle Angaben korrekt sind, können Sie anschließend die gewünschte Datenbank auswählen.



Abbildung 24.4 Datenbank auswählen

Ordner für Datenbankverbindungen

Dreamweaver legt für die Konnektierungen in Ihrer Site einen neuen Ordner mit dem Namen *Connections* an. Sie können für eine Website auch mehrere Verbindungen zu verschiedenen Datenbanken definieren. Für jede Datenbankverbindung wird eine eigene Datei geschrieben. Diesen Ordner müssen Sie beim Veröffentlichen der Website auf den Server kopieren.

In diesem Ordner finden Sie jetzt eine Datei mit dem folgenden Inhalt:

```
<?php
# FileName="Connection_php_mysql.htm"
# Type="MYSQL"
# HTTP="true"
$hostname_TESTVERBINDUNG = "localhost";
$database_TESTVERBINDUNG = "testdatenbank";
$username_TESTVERBINDUNG = "root";
$password_TESTVERBINDUNG = "";
$TESTVERBINDUNG = mysql_pconnect($hostname_TESTVERBINDUNG, $username_
TESTVERBINDUNG, $password_TESTVERBINDUNG) or trigger_error(mysql_
error(),E_USER_ERROR);
?>
```

Listing 24.1 Datenbank-Konnektierung mit PHP

Im weiteren Verlauf der Arbeit wird diese Datei wie eine externe CSS-Datei eingebunden. So können Sie an zentraler Stelle alle Zugangsdaten und Parameter verwalten.

Verbindungsskripte anzeigen

Neben diesem Ordner legt Dreamweaver einen versteckten Ordner auf dem Laufwerk des Testservers an. Aktivieren Sie in der Site-Ansicht Ansicht • versteckte Dateien anzeigen, und die Verbindungsskripte werden sichtbar.

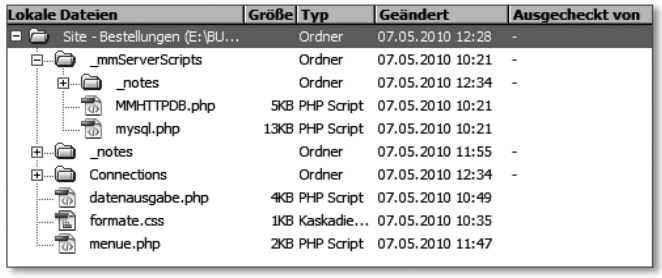


Abbildung 24.5 Versteckte Ordner mit Verbindungsskripten

Sicherheitslücke! Verbindungsskripte entfernen [!]

Die Verbindungsskripte werden von Dreamweaver benötigt, um die Live-Ansicht zu aktivieren. Nachdem Sie mit der Arbeit an der Website fertig sind, sollten Sie diese unter allen Umständen entfernen und auf keinem Fall auf den Remote-Server übertragen. Jedes zusätzliche Skript mit der Möglichkeit eines Fernzugriffs stellt ein Sicherheitsrisiko dar. Entfernen Sie die Skripte, indem Sie in der Site-Ansicht auf SITE • ERWEITERT • VERBINDUNGSSKRIPTE ENTFERNEN klicken.

24.2.2 Das Bedienfeld »Datenbanken«

Bei erfolgreicher Datenbankverbindung werden im Bedienfeld DATENBANKEN die strukturellen Inhalte der verbundenen Datenbanken angezeigt. Durch einen Klick auf das Plussymbol vor der Verbindungsbezeichnung ❶ sehen Sie die einzelnen Tabellen ❷ und die Felder dieser Tabellen ❹. Felder mit Primärschlüsseln werden mit einem kleinen Schlüsselsymbol markiert ❸. Auf diese Weise haben Sie immer einen Überblick über die Struktur Ihrer Datenbank.

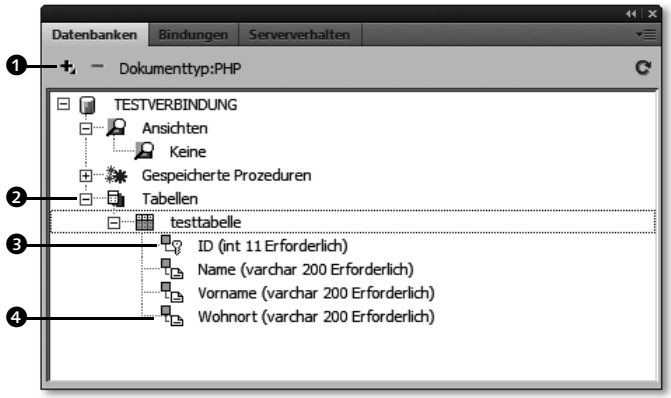


Abbildung 24.6 Anzeige im Bedienfeld »Datenbanken«

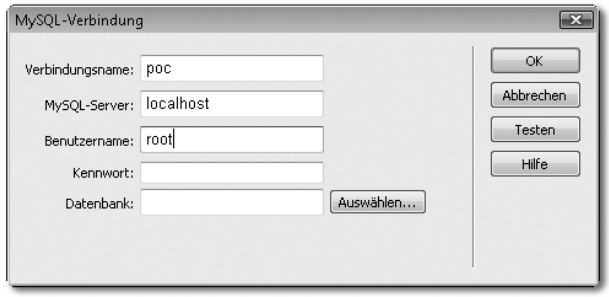
Es ist sehr schade, dass aus Dreamweaver heraus nur MySQL-Datenbanken mit PHP konnektiert werden können. PHP unterstützt fast jedes noch so exotische Datenbankmodell, so dass man sich in keiner Weise auf MySQL beschränken müsste.

Schritt-für-Schritt: Datenbank an die Buchwebsite anbinden [▪]

1 *Neue Verbindung anlegen*

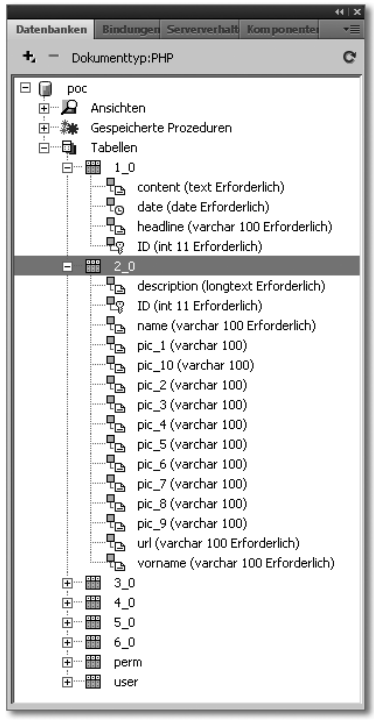
Im nun folgenden Schritt werden wir eine Verbindung zur Datenbank herstellen. Legen Sie dazu eine neue MySQL-Verbindung im Bedienfeld DATENBANKEN auf die von Ihnen erstellte Datenbank oder die importierte Datenbank *poc* an.


Buchwebsite



2 *Datenbankstruktur*

Im DATENBANKEN-Bedienfeld wird Ihnen jetzt die gesamte Struktur der Datenbank angezeigt. Wenn Sie die Datenbank aus dem Buch importiert haben, sind bereits Tabellen enthalten, die wir erst in einem späteren Schritt erläutern werden.



In der Ordnerstruktur wurde auch der neue Ordner für die Verbindungen automatisch angelegt.



Daten aus Datenbanken können Sie mit Dreamweaver auf einfache Weise in Ihr Dokument einfügen und ausgeben. In diesem Kapitel zeigen wir Ihnen, wie Sie dabei vorgehen und auf was Sie achten müssen.

25 Datenbanken abfragen

Für das nun Folgende setzen wir voraus, dass Ihr Testserver fertig eingerichtet ist und eine Verbindung zur Datenbank besteht. Wir zeigen Ihnen jetzt die verschiedenen Möglichkeiten, dynamische Daten in Ihr Dokument einzufügen. Wundern Sie sich nicht, wenn Sie in den meisten Beispieldokumenten keine HTML-Elemente finden. Wir verzichten darauf, um die dynamischen Verknüpfungen deutlicher hervorzuheben.

Im Laufe des Kapitels werden wir immer wieder auf die in Kapitel 23, »MySQL-Grundlagen«, angelegte Testdatenbank zurückgreifen. Sie müssen diese Übungen nicht mitmachen – mit der Buchwebsite geht's am Ende dieses Kapitels weiter. Zum Verständnis der Zusammenhänge empfehlen wir Ihnen jedoch, die beschriebenen Schritte nachzuvollziehen. Den SQL-Dump finden Sie auf der beiliegenden DVD unter *Beispiele/Testdatenbank*. (Ein SQL-Dump ist eine Textdatei mit einem SQL-Skript, das die Datenbank inklusive Inhalte anlegt.) [o]

Die meisten Arbeiten an dynamischen Websites sind nicht durch einen einfachen Klick zu erledigen, sondern werden in einer Abfolge von Einzelschritten durchgeführt.In den weiteren Abschnitten sind daher immer wieder kleinere Übungen enthalten, um diese Arbeitsschritte zu verdeutlichen.

25.1 Datenbankabfrage und Datenanzeige

Das Verfahren einer Datenbankabfrage und Datenanzeige entspricht dem in Abbildung 25.1 dargestellten Schema.

Zunächst muss grundsätzlich eine Verbindung zur Datenbank hergestellt werden. Diesen Arbeitsschritt haben wir bereits in Abschnitt 24.1, »Datenbanken konnektieren«, beschrieben. Der nächste Schritt ist die Abfrage der gewünschten Daten aus der Datenbank und ihre Zwischenspeicherung. Im letzten Schritt werden die Daten aufbereitet, in das PHP-Dokument geschrieben und angezeigt.

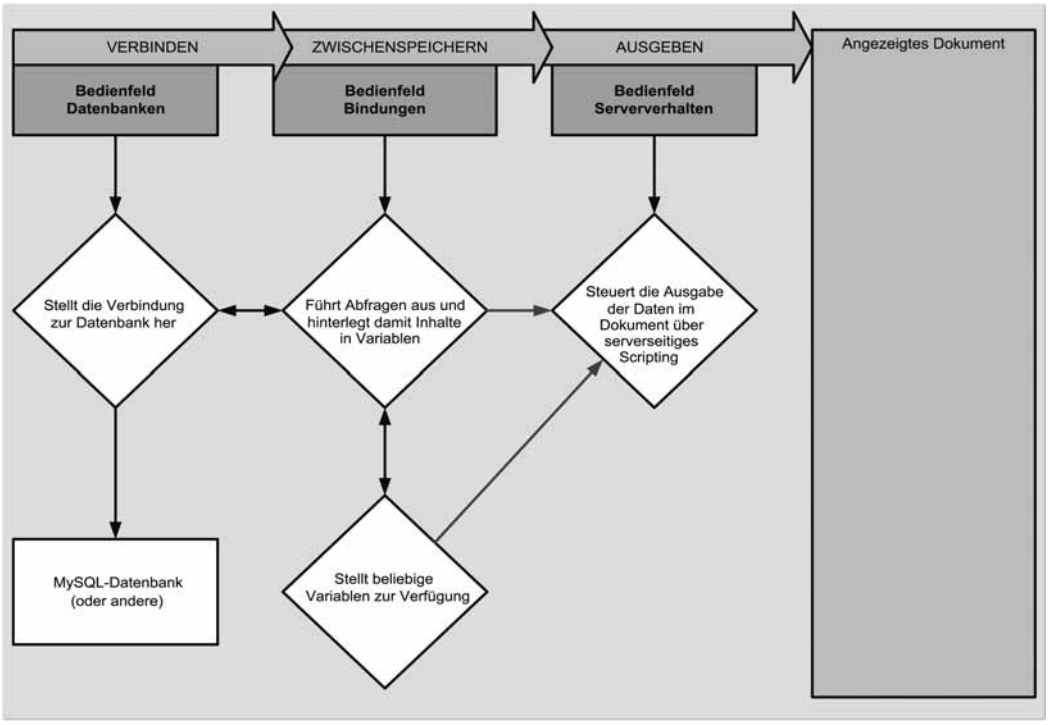


Abbildung 25.1 Schema der Datenausgabe in Dreamweaver

Die Dreiteilung »Verbinden – Abfragen und Zwischenspeichern – Ausgeben« bleibt bei allen Aktionen bestehen. Das Einzige, was Sie im Entwurfsmodus sehen, ist die Ausgabe der Daten. Alles andere können Sie nur im Quelltext verfolgen.

25.2 Datensätze in Dreamweaver abfragen

In diesem Abschnitt erfahren Sie, wie Sie Abfragen erstellen, Datensätze anzeigen, mit der URL übergeben, Anfragen testen und vieles mehr. Doch vorab noch ein wichtiger Hinweis.

[!] **Abfragen niemals modifizieren!**

Abfragen, die wir in den folgenden Abschnitten behandeln, können durch einen Doppelklick auf die jeweilige Abfrage erneut bearbeitet werden. Aus leidvoller Erfahrung (Stand Mai 2010) können wir jedoch nur davor warnen, das zu tun. Legen Sie lieber komplett neue Abfragen an. Dreamweaver neigt dazu, modifizierte Abfragen nicht zu verändern, sondern erneut einzufügen, den eigenen Quelltext nicht mehr zu erkennen und Ähnliches.

Daher nochmals unser Hinweis, Abfragen immer neu anzulegen – *nicht* zu modifizieren. Das gilt leider in besonderem Maße für die aktuelle Dreamweaver-Version CS5. Wir hoffen sehr, dass bei den ersten Updates dieses Problem behoben wird.

Datensatzabfragen richtig löschen [«]

Bei Änderungen oder beim Löschen von Abfragen entfernt Dreamweaver den Quelltext der alten Abfragen nicht immer vollständig. Aus diesem Grund sollten Sie stets im Quelltext prüfen, ob alle eingefügten Codezeilen auch wirklich gelöscht worden sind. Das Sicherste ist, Abfragen vollständig zu löschen und ganz von vorne zu beginnen, wenn Ihnen ein Fehler unterlaufen ist.

25.2.1 Einfache Abfragen erstellen

Um eine einfache Abfrage zu erstellen, klicken Sie im Bedienfeld BINDUNGEN auf das Plussymbol und wählen DATENSATZGRUPPE (ABFRAGE) ❶ aus.

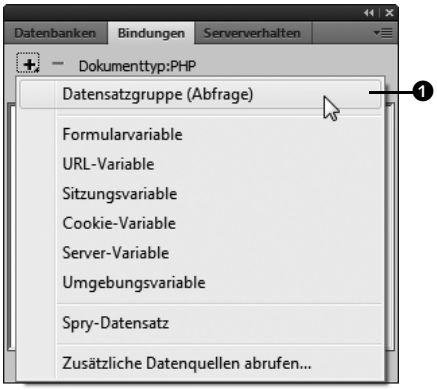


Abbildung 25.2 Einfache Abfrage erstellen

Bedienfeld Bindungen [«]

Im Bedienfeld BINDUNGEN werden alle verfügbaren Datenquellen angezeigt, aus denen Inhalte in das Dokument eingebunden werden können. Auch Variablen können hier definiert und aufgelistet werden.

Name der Abfrage

Es erscheint die Dialogbox DATENSATZGRUPPE. Hier müssen Sie der Abfrage einen Namen zuweisen und eine Verbindung definieren. Geben Sie unter NAME ❷ (siehe Abbildung 25.3) zuerst einen eindeutigen Namen im Rahmen der gültigen Konventionen für Unix-Systeme ein (d.h. Sonder- und Leerzeichen sind nicht erlaubt). In unserem Beispiel haben wir sie *abfrage_1* genannt.

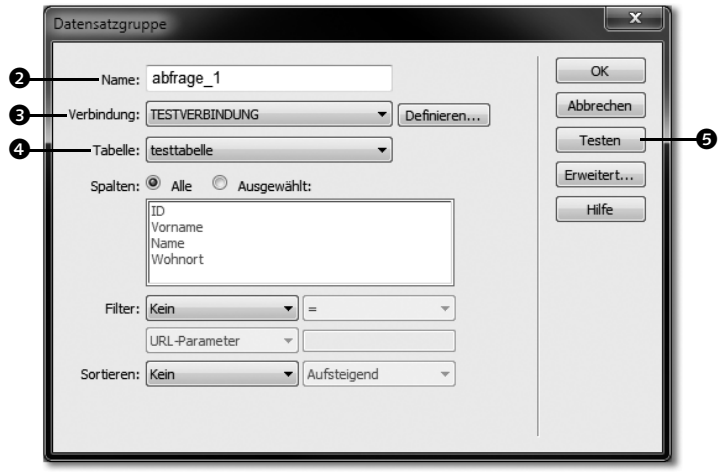


Abbildung 25.3 Namensvergabe und Verbindungsauswahl

Datenbankanbindung auswählen

In der Auswahlliste VERBINDUNG ❸ wählen Sie eine bereits erstellte Datenbankanbindung aus. In unserem Beispiel heißt die Anbindung einfach TESTVERBINDUNG. Anfangs ist es hilfreich, die Reihenfolgen des Schemas auch bei der Arbeitsweise beizubehalten. Mit zunehmender Erfahrung können Sie auch mehrere Schritte auf einmal erledigen. Die Datenbankanbindung kann in dieser Dialogbox auch gleich beim Anlegen einer Abfrage definiert werden.

Achten Sie darauf, dass zunächst kein ERWEITERTER MODUS der Dialogbox eingeschaltet ist. Falls die Dialogbox anders aussieht als in unserer Abbildung, klicken Sie auf die im erweiterten Modus erscheinende Schaltfläche EINFACH, um wieder in den Startmodus zu gelangen.

Bei erfolgreicher Konnektierung stehen Ihnen jetzt im Menü TABELLE ❹ alle Tabellen der ausgewählten Datenbank zur Verfügung. Klicken Sie anschließend auf TESTEN ❺, um zu sehen, ob Datensätze angezeigt werden.



Abbildung 25.4 SQL-Abfrage testen

Tabellen abfragen

Jetzt können Sie die abzufragenden Tabellen auswählen. In unserem Beispiel steht nur eine Tabelle zur Verfügung, die auch automatisch vorausgewählt ist. Im nächsten Schritt (beim Erstellen einer Datenbankabfrage) müssen Sie Dreamweaver mitteilen, welche Spalten abgefragt werden sollen ❻ (siehe Abbildung 25.5). Dabei können Sie unterscheiden, ob Sie alle oder nur einzelne abfragen wollen. Wenn Sie Alle aktivieren, entspricht dies in der SQL-Anweisung dem Befehl `SELECT * FROM`.

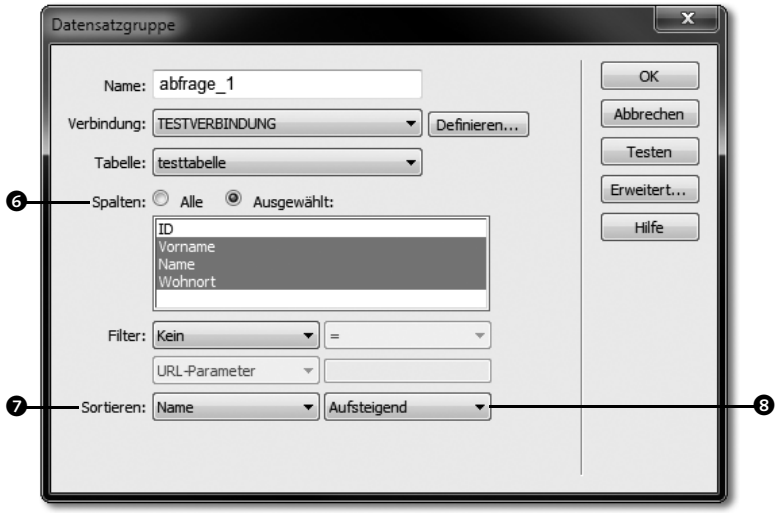


Abbildung 25.5 Auswahl der Tabellenspalten und Festlegung der Sortierreihenfolge

Wir möchten uns nur die Inhalte der Tabellenspalten Vorname, Name und Wohnort anzeigen lassen. Daher aktivieren wir die Option Ausgewählt. Durch Drücken der Taste Strg/⌘ und Klicken auf die Spaltennamen können mehrere Spalten gleichzeitig ausgewählt werden.

SQL generieren [«]

Sie sehen hier bereits: Was eigentlich im Hintergrund geschieht, ist das Erstellen einer SQL-Abfrage. Wie Sie in einem späteren Abschnitt noch sehen werden, kann das natürlich auch von Hand erfolgen.

Daten sortieren

Nach der Auswahl der abzufragenden Spalten legen wir noch die Sortierung der Daten fest. Stellen Sie das Feld, nach dem sortiert werden soll, im Menü Sortieren ❼ ein, und wählen Sie daneben die Reihenfolge der Sortierung aus ❽. In der

SQL-Anweisung entspricht das in unserem Beispiel einem `ORDER BY Name ASC` oder alternativ bei absteigender Sortierung `ORDER BY Name DESC`.

Abfrage testen

Wenn Sie nun auf OK klicken, erscheinen im Bedienfeld BINDUNGEN die neuen Einträge. Hier werden die abgefragten Tabellenspalten angezeigt, wenn Sie auf das kleine Plussymbol vor dem Namen der Datensatzgruppe klicken. Wenn Sie die Parameter der Abfrage nachträglich ändern möchten, können Sie dies durch einen Doppelklick auf die Datensatzgruppe erreichen. Es wird dann wieder die soeben beschriebene Dialogbox angezeigt.

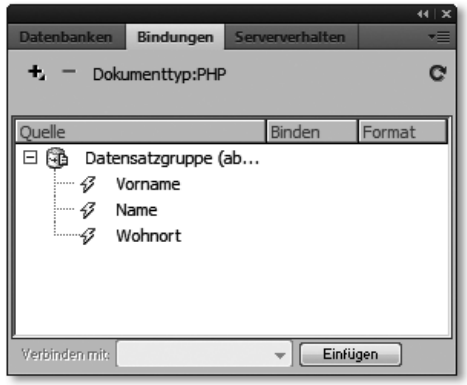


Abbildung 25.6 Neue angebundene Datenquellen im Bedienfeld »Bindungen«

Im Bedienfeld SERVERVERHALTEN erscheint nun ebenfalls ein Eintrag, der Ihnen für weitere Aktionen zur Verfügung steht.

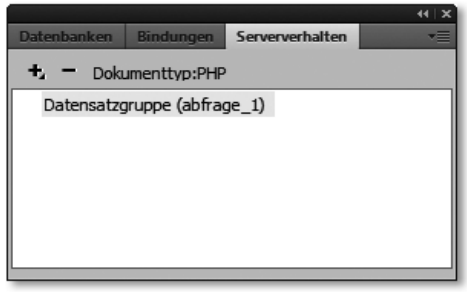


Abbildung 25.7 Bedienfeld »Serververhalten«

Zunächst benötigen wir dieses Bedienfeld noch nicht. Achten Sie anfangs immer darauf, welche Veränderungen sich in anderen Bedienfeldern der Bedienfeldgruppe ANWENDUNG ergeben, wenn Sie etwas Neues hinzufügen oder entfernen. So erschließen sich Ihnen mit der Zeit die Zusammenhänge.

Veränderungen im Quelltext

Im Quelltext sind jetzt weitere Eintragungen hinzugekommen, ohne dass Sie Auswirkungen in Ihrem Dokument sehen. Da wir bislang nur die Bindungen bearbeitet haben, wurde zunächst lediglich die Basis für das Anzeigen von dynamischen Daten geschaffen, aber noch keine Anzeige eingefügt.

Ganz oben im Quelltext sehen Sie den Verweis auf das Skript mit der Datenbankanbindung:

```
<?php require_once('Connections/TESTVERBINDUNG.php'); ?>
```

Listing 25.1 Quelltextzeile der Datenbankanbindung

Die eigentliche Datenbankabfrage folgt nach der Konnektierung. Dreamweaver erstellt in diesem Fall das Array `$row_abfrage_1` und eine weitere Variable `$totalRows_abfrage_1` mit der Anzahl der Datensätze für weitere mögliche Aktionen:

```
mysql_select_db($database_TESTVERBINDUNG, $TESTVERBINDUNG);
$query_abfrage_1 = "SELECT Vorname, Name, Wohnort
  FROM testtabelle ORDER BY Name ASC";
$abfrage_1 = mysql_query($query_abfrage_1, $TESTVERBINDUNG) or
  die(mysql_error());
$row_abfrage_1 = mysql_fetch_assoc($abfrage_1);
$totalRows_abfrage_1 = mysql_num_rows($abfrage_1);
```

Listing 25.2 Quelltext einer Bindung bzw. der Abfrage

Ganz unten im Dokument finden Sie die folgenden Zeilen:

```
<?php
mysql_free_result($abfrage_1);
?>
```

Listing 25.3 Quelltextzeile der Datenfreigabe

Dieser Befehl bewirkt die Freigabe der Daten nach einer erfolgten Abfrage. Es ist bei MySQL nicht zwingend notwendig, die Daten wieder freizugeben. Eindeutiger und sauberer in der Programmierung ist es in jedem Fall. Es gilt, beim Programmieren immer Eindeutigkeit zu erreichen.

[»] **Quelltext von Dreamweaver**

Wenn Sie sich als eventuell erfahrener Programmierer den von Dreamweaver erstellten Quelltext näher ansehen, werden Sie feststellen, dass Dreamweaver alles das generiert, was man eigentlich unbedingt vermeiden sollte.

In der Tat erzeugt Dreamweaver ein einziges wildes Gemisch aus Inhalten, PHP, SQL und XHTML. Wenn Sie ein umfangreiches Projekt zu realisieren haben, werden Sie daher sicherlich von den visuellen Methoden Abstand nehmen müssen, da der erstellte Quelltext kaum pflegbar ist.

Für einfachere dynamische Websites ist der Quelltext aber durchweg brauchbar und absolut funktionstüchtig. Außerdem muss man ja auch nicht unbedingt gleich mit Kanonen auf Spatzen schießen und beim kleinsten News-Bereich oder Mini-CMS auf objektorientierte Programmierung usw. zurückgreifen.

25.2.2 Anzeigen von Datensätzen

Um die Datensätze in Ihrem Dokument anzuzeigen, ziehen Sie diese einfach mit der Maus aus dem Bedienfeld Bindungen in Ihr Dokument. Die nun eingefügten Datensätze werden hellblau hinterlegt und als Platzhalter dargestellt.

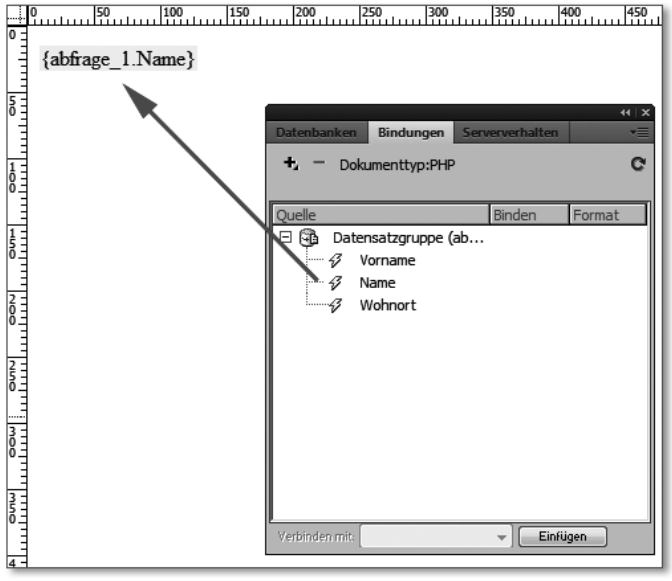


Abbildung 25.8 Eingefügte Datensätze werden als hellblaue Platzhalter dargestellt.

In Ihrem Quelltext sind nun, wenn Sie alle drei Datensätze eingefügt haben, drei weitere Zeilen hinzugekommen. Sie bewirken die Ausgabe des ersten Datensatzes aus dem Array `$row_Abfrage_1`:

```
<?php echo $row_Abfrage_1['Name']; ?>
<?php echo $row_Abfrage_1['Vorname']; ?>
<?php echo $row_Abfrage_1['Wohnort']; ?>
```

Listing 25.4 Quelltextzeilen der angezeigten Datensätze

Aktivieren Sie die Live-Ansicht, um die Datensätze in der Vorschau zu sehen.



Abbildung 25.9 Datenfelder in der Live-Ansicht

In der Live-Ansicht erhalten Sie nicht nur eine Vorschau der Datensätze, sondern sehen, wie diese auf der Website wirklich aussehen werden. Leider ist eine Bearbeitung in der Live-Ansicht nicht mehr möglich.

Wie Sie sehen, ist es nicht schwer, in Dreamweaver eine Datenbank abzufragen und in einer Website anzuzeigen. Um professionelle, dynamische Websites zu erstellen, raten wir Ihnen, immer wieder in die Codeansicht zu wechseln, um ein Gespür für die dahinterliegende Technologie zu bekommen.

25.2.3 Erweiterte Abfragen erstellen

In der soeben erstellten Abfrage mussten Sie sich nicht um SQL-Befehle kümmern. Dreamweaver hat Ihnen diese Arbeit abgenommen. In der Praxis kommt es jedoch häufiger vor, dass Sie modifizierte SQL-Befehle benötigen.

Um SQL-Befehle zu bearbeiten, klicken Sie sich über das Bedienfeld Bindungen, das Pluszeichen und Datensatzgruppe (Abfrage) in die Dialogbox Datensatzgruppe und aktivieren dort den erweiterten Modus.

Im erweiterten Modus müssen Sie der Abfrage wieder einen Namen zuweisen und eine Verbindung definieren. Nachdem Sie eine Datenbank ausgewählt haben, erscheinen im Fensterbereich Datenbankelemente die einzelnen verfügbaren Tabellen mit ihren Spalten. Klicken Sie auf die Plussymbole, dann werden die Tabellen geöffnet. Um eine Tabellenspalte abzufragen, markieren Sie die gewünschte Spalte und klicken anschließend auf die Schaltfläche SELECT.

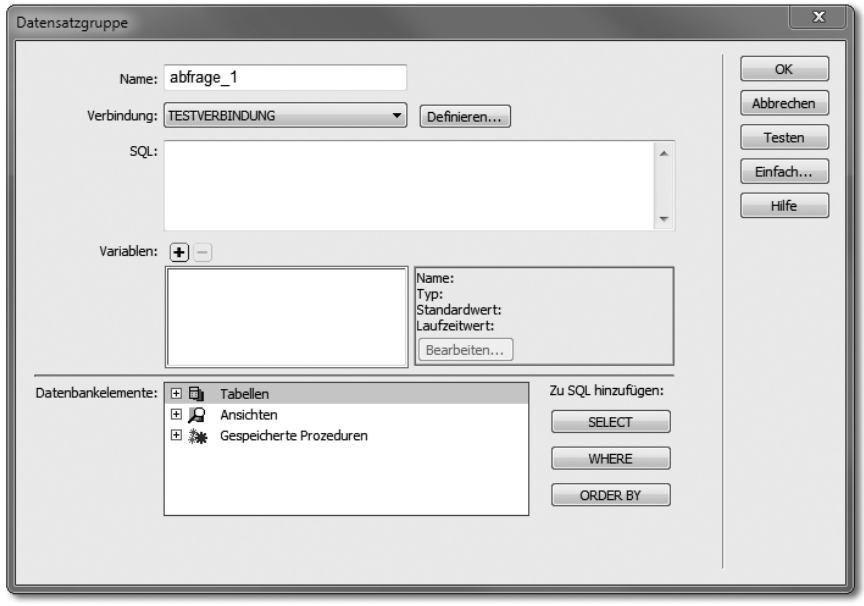


Abbildung 25.10 Auswahl der Tabellen in der Dialogbox »Datensatzgruppe«



Abbildung 25.11 Mit SQL-Befehlen Tabellenspalten abfragen

Mehrere Spalten abfragen

Um mehrere Tabellenspalten zur Abfrage hinzuzufügen, wiederholen Sie diesen Schritt für jede einzelne Spalte. Dabei werden keine neuen SELECT-Abfragen eingefügt, sondern nur die zusätzlichen Spalten in die bestehende Abfrage eingebunden.

Datensätze sortieren

Eine Sortierung können Sie einfügen, indem Sie die Spalte, nach der sortiert werden soll, anwählen und anschließend auf ORDER BY klicken. Die Sortierreihenfolge DESC für absteigend oder ASC für aufsteigend müssen Sie von Hand im SQL-Skript hinter den Eintrag bei ORDER BY eintragen. Wenn Sie dort nichts angeben (wie in Abbildung 25.12), wird automatisch ein ASC eingesetzt und aufsteigend sortiert (siehe Abbildung 25.13).



Abbildung 25.12 Fertige Abfrage mit Sortierung im erweiterten Modus

Wie Sie sehen, ist die Abfrage nun fertig gestellt. Schalten Sie jetzt einfach in den einfachen Modus (EINFACH...) um. Manchmal werden hierbei von Hand geschriebene Eintragungen entfernt. Schalten Sie jetzt wieder in den erweiterten Modus. Die SQL-Abfrage ist nun optimiert.

Wenn Sie nicht im Quelltext arbeiten möchten, können Sie Ihre Abfragen auch im einfachen Modus erzeugen, um dann im erweiterten Modus die gewünschten Änderungen vorzunehmen, oder gleich die komplette Abfrage im erweiterten Modus erstellen.



Abbildung 25.13 Optimierte Schreibweise bei SQL-Abfrage nach Umschalten in den einfachen und Zurückschalten in den erweiterten Modus

25.2.4 Mehrere Datensätze ausgeben

Die Art der Datenanzeige können Sie im Bedienfeld Serververhalten variieren. Serververhalten steuern die Anzeige der Daten in Ihrem Dokument. Häufig müssen mehrere Datensätze einer Tabelle gleichzeitig angezeigt werden. In unserem Beispiel möchten wir alle in der *Testtabelle* vorhandenen Datensätze auslesen und in das Dokument ausgeben. Benötigt wird diese Vorgehensweise bei Listen, Auswahlmenüs usw.

Wechseln Sie zuerst in der Bedienfeldgruppe Anwendung auf Serververhalten. Markieren Sie im Dokument denjenigen Bereich, den Sie wiederholen wollen (siehe Abbildung 25.14). Eine Auswahl im Bedienfeld Serververhalten funktioniert leider nicht. Wir wollen eine Reihe der Tabelle mit den Datensätzen wiederholt ausgeben. Da bei einer Auswahl in der Layoutansicht nicht die gesamte Reihe (`<tr>`) ausgewählt wird, nehmen wir die Auswahl im Quellcode direkt vor. So sehen Sie auch gleich, dass die Serververhalten sowohl im Quelltext als auch in der Layoutansicht funktionieren.

Das von uns benötigte Serververhalten heißt Bereich wiederholen. Markieren Sie die gesamte Tabellenreihe im Quellcode

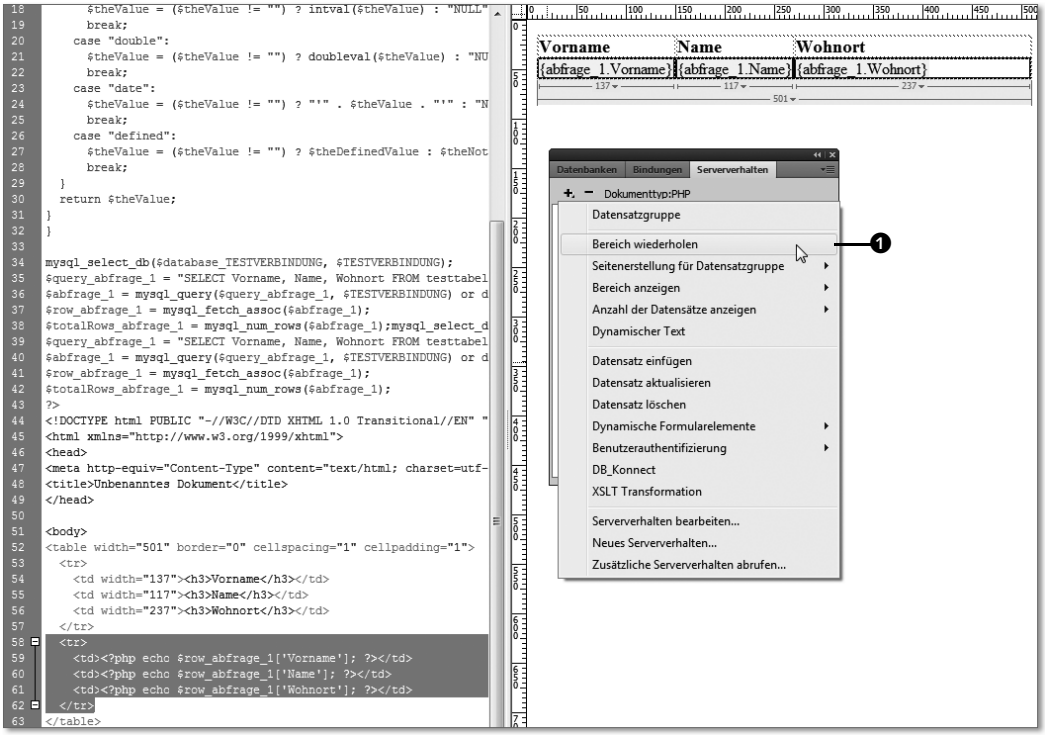


Abbildung 25.14 Bedienfeld SERVERVERHALTEN und markierte Tabelle

Wählen Sie anschließend im Bedienfeld SERVERVERHALTEN durch Klicken auf das Plussymbol die Option BEREICH WIEDERHOLEN ❶ aus. Jetzt müssen Sie im Popup-Menü der Dialogbox die gewünschte DATENSATZGRUPPE bestimmen. Ihnen stehen nun die Option ALLE DATENSÄTZE ANZEIGEN oder die Beschränkung auf eine bestimmte Anzahl von Datensätzen zur Verfügung. Wir zeigen in unserem Beispiel alle Datensätze an.



Abbildung 25.15 Dialogbox »Bereich wiederholen«

Im Anschluss wird im Dokument der wiederholte Bereich in einem grauen Rahmen angezeigt. Sie können die Parameter des wiederholten Bereichs jederzeit ändern, indem Sie im Bedienfeld SERVERVERHALTEN doppelt auf den Eintrag BEREICH WIEDERHOLEN klicken.

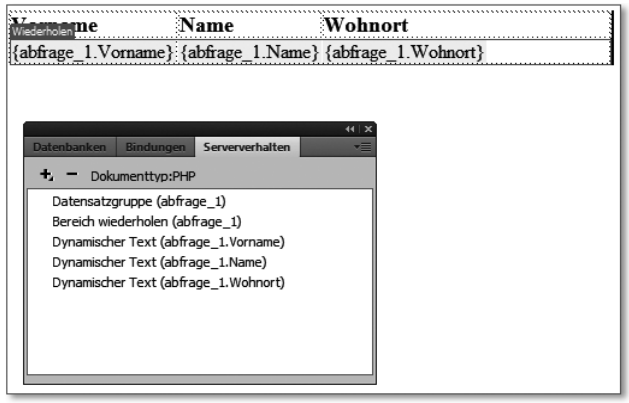


Abbildung 25.16 Anzeige des wiederholten Bereichs im Dokument

Wenn Sie jetzt zur Live-Ansicht wechseln, werden die Datensätze angezeigt.

Vorname	Name	Wohnort
Richard	Beer	Hamburg
Susann	Gailus	Hamburg

Abbildung 25.17 Der wiederholte Bereich in der Live-Ansicht

Wenn Sie in die Codeansicht wechseln, finden Sie im Quelltext den folgenden neuen Eintrag:

```
<?php do { ?>
<tr>
  <td><?php echo $row_abfrage_1['Vorname']; ?></td>
  <td><?php echo $row_abfrage_1['Name']; ?></td>
  <td><?php echo $row_abfrage_1['Wohnort']; ?></td>
</tr>
<?php } while ($row_abfrage_1 = mysql_fetch_assoc($abfrage_1)); ?>
```

Listing 25.5 Wiederholter Bereich im Quelltext

Der ganze Block für die Anzeige läuft innerhalb einer `do-while`-Schleife. Diese wird so lange ausgeführt, wie Datensätze vorhanden sind.

25.2.5 Bestimmte Datensätze abfragen

Mehrere Datensätze muss man häufig für Übersichts- bzw. Auswahlseiten gleichzeitig anzeigen, um zum Beispiel eine Übersicht über alle Einträge zu erhalten.

Auf den Unterseiten einer Website müssen Sie jedoch in den meisten Fällen einen Datensatz anhand einer ID auswählen und nur diesen anzeigen. Dies geschieht mit dem SQL-Befehl `WHERE`. So zeigt zum Beispiel `SELECT * FROM testtabelle WHERE ID = 2` den Datensatz mit der ID 2 aus der Tabelle `testtabelle` an.

Mit Dreamweaver können Sie solche Parameter beim Anlegen der Bindungen bereits vorgeben und so auf jeder Unterseite bestimmen, welcher Datensatz der ausgewählten Tabelle angezeigt wird.

Die Auswahl der anzuzeigenden Datensätze erfolgt am besten immer über den Primärschlüssel. In einer relationalen Datenbank ist dieser Wert absolut eindeutig. In unseren Beispielen ist der Primärschlüssel bzw. das als Primärschlüssel deklarierte Feld das Feld mit der Bezeichnung *ID*. Öffnen Sie zum Abfragen eines bestimmten Datensatzes die entsprechende Datensatzgruppe, oder legen Sie eine neue an.

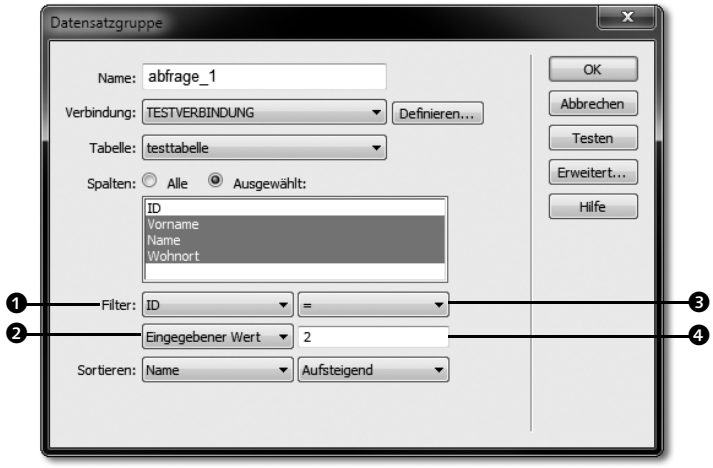


Abbildung 25.18 Auswahl eines bestimmten Datensatzes für die Datenbankabfrage

Filter auf Abfragen anwenden

Unter Filter ❶ wählen Sie jetzt die Tabellenspalte mit Ihrem Primärschlüssel aus. In unserem Fall ist es die Spalte »ID«. Bei dieser Abfrage möchten wir die Werte selbst eingeben. Wählen Sie daher aus dem Menü unter Filter das Feld Eingegebener Wert ❷ aus.

Vergleichsoperator verwenden

Rechts neben FILTER finden Sie eine Auswahl möglicher Vergleichsoperatoren. Da wir möchten, dass der eingegebene Wert dem Wert in der Tabellenspalte entspricht, müssen Sie das Gleichheitszeichen ❸ (siehe Abbildung 25.18) auswählen. Bei SQL wird übrigens das normale mathematische Gleichheitszeichen verwendet (bei PHP wäre dies das Zeichen ==).

Nun müssen Sie noch rechts neben EINGEGEBENER WERT den Wert eintragen, der in Ihrer Tabellenspalte vorkommt und angezeigt werden soll. In unserem Fall ist es der Datensatz mit der ID `2` ❹.

Wenn Sie jetzt in die erweiterte Ansicht umschalten, sehen Sie gleich die Auswirkungen im SQL-Befehl. Dreamweaver hat an den bisherigen Befehl ein `WHERE ID = 2` angehängt.



Abbildung 25.19 Die `WHERE`-Bedingung in der erweiterten Ansicht

In der Live-Ansicht sehen Sie jetzt den Datensatz mit der ID `2`.

Vorname	Name	Wohnort
Susann	Gailus	Hamburg

Abbildung 25.20 Live-Ansicht mit der ID 2

Öffnen Sie die Abfrage nochmals, und ändern Sie die ID auf 1. Betrachten Sie anschließend Ihr Dokument in der Live-Ansicht erneut. Wenn alles korrekt verlaufen ist, wird Ihnen jetzt der Datensatz mit der ID 1 angezeigt.

Vorname	Name	Wohnort
Richard	Beer	Hamburg

Abbildung 25.21 Live-Ansicht mit nachträglich veränderter ID auf Wert 1

Mit dieser Vorgehensweise können Sie bereits erste eigene dynamische Seiten aufbauen. Sie können zum Beispiel die gesamten Texte einer Website in einer MySQL-Datenbank hinterlegen und für jedes Dokument eine eigene Abfrage mit einer anderen Datensatz-ID schreiben. Neue Texte kopieren Sie einfach mit phpMyAdmin in die MySQL-Datenbank. Diese Abfragen funktionieren selbstverständlich auch mit Microsoft SQL-Server, Access und anderen Datenbanksystemen.

Der Nachteil dieser Vorgehensweise ist, dass Sie immer noch für jeden Datensatz ein eigenes Dokument mit einer eigenen Abfrage benötigen. Wesentlich komfortabler wird es, wenn man die ID mit einer Variablen übergibt und die Abfrage mit dieser ausführt.

Schwerer Dreamweaver-Fehler! [!]

Wir haben mehrfach festgestellt, dass Dreamweaver in der Version CS5 bei einem Doppelklick auf die Bindung die Abfrage nicht einfach modifiziert, sondern dupliziert. Es sind dann zwei Abfragen in einem Dokument. Das kann nicht funktionieren. Achten Sie *unbedingt* darauf, dass Sie die alte Abfrage von Hand löschen. Wir hoffen sehr, dass dieser Fehler beim ersten Update behoben wird.

```
mysql_select_db($database_TESTVERBINDUNG, $TESTVERBINDUNG);
$query_abfrage_1 = "SELECT Vorname, Name, Wohnort FROM testtabelle ORDER BY Name ASC";
$abfrage_1 = mysql_query($query_abfrage_1, $TESTVERBINDUNG) or die(mysql_error());
$row_abfrage_1 = mysql_fetch_assoc($abfrage_1);
$totalRows_abfrage_1 = mysql_num_rows($abfrage_1);mysql_select_db($database_TESTVERBINDUNG, $TESTVERBINDUNG);
$query_abfrage_1 = "SELECT Vorname, Name, Wohnort FROM testtabelle WHERE ID = 2 ORDER BY Name ASC";
$abfrage_1 = mysql_query($query_abfrage_1, $TESTVERBINDUNG) or die(mysql_error());
$row_abfrage_1 = mysql_fetch_assoc($abfrage_1);
$totalRows_abfrage_1 = mysql_num_rows($abfrage_1);
```

Abbildung 25.22 Doppelter Quelltext durch Dreamweaver-Fehler

25.2.6 Datensatz-ID mit URL übergeben

Abfragen können – wie Sie eben gesehen haben – mit einem festen Wert durchgeführt werden. Da wir mit PHP arbeiten, kann dieser Wert auch in einer Variab-

len stehen und je nach vorheriger Auswahl oder Aktion verschiedene Inhalte annehmen.

So könnte eine Website aus einer Adressübersicht bestehen. Klickt man auf einen Namen, erscheint in einem neuen Dokument der Detaildatensatz. Anstelle des festen Wertes wie im Abschnitt weiter oben werden dann die deklarierten Variablen eingefügt. Das erste Dokument übergibt so die Variable mit einem beliebigen Inhalt an das zweite Dokument. Im zweiten Dokument wird die Abfrage dann mit dem Inhalt der übergebenen Variablen durchgeführt.

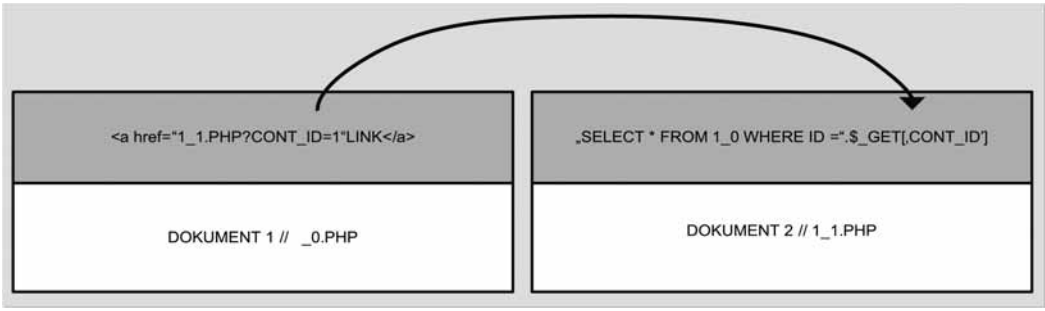


Abbildung 25.23 Variablenübergabe an SQL-Abfrage

Diese Vorgehensweise ist die mit Abstand häufigste und nach einiger Übung auch effektivste. Bislang mussten Sie noch für jeden Inhalt ein eigenes Dokument anlegen. Jetzt können Sie beliebig viele unterschiedliche Inhalte in einem einzigen Dokument darstellen. Prinzipiell kann eine Website mit 5.000 verschiedenen Unterpunkten aus einem einzigen PHP-Dokument bestehen.

Dynamische Navigation

Eine Auswahlliste ist im Prinzip nichts anderes als eine Navigation. Nach diesem Schema können Sie auch eine Navigation aufbauen, die in einem Teil des Dokuments dargestellt wird und in einem anderen Dokumentbereich die Inhalte anzeigen lässt. Wenn Sie mit der Navigation die Datensatz-IDs übergeben und dieses Dokument auf sich selbst verlinken, können Sie äußerst umfangreiche Internetseiten mit einem einzigen PHP-Dokument erstellen. Zusätzlich können Sie die Darstellung der Navigation verändern, wenn die übergebene Variable mit dem Wert in der Navigation übereinstimmt.

URL-Parameter übergeben

Öffnen Sie die bestehende Datensatzgruppe, oder legen Sie am besten eine neue an, und wählen Sie im Menü unterhalb Filter statt Eingegebener Wert nun URL-Parameter ❶ aus. Rechts daneben müssen Sie den Namen der übergebenen

Variablen eintragen. In unserem Fall soll die Variable mit dem Namen `ID` die ID für den Datensatz an das Dokument übertragen.

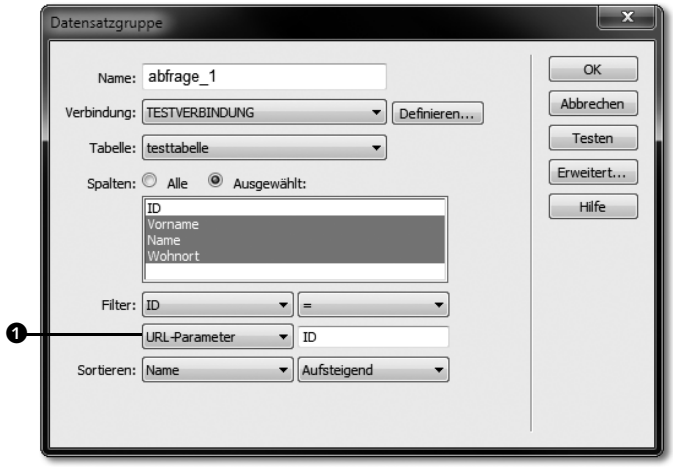


Abbildung 25.24 Variablenübergabe an SQL-Abfrage mit URL-Parameter

Schalten Sie jetzt in die erweiterte Ansicht um, dann sehen Sie einige neue Eintragungen. Dreamweaver hat im Feld VARIABLEN eine Variable namens `colname` eingetragen. Rechts daneben stehen ein STANDARDWERT `-1` und ein LAUFZEITWERT `$_GET['ID']`.



Abbildung 25.25 URL-Parameter in der erweiterten Ansicht

Die Variable `colname` hat Dreamweaver selbständig erstellt, da der übertragene Wert nochmals zwischengespeichert wird, um Fehler abzufangen. Wenn keine Variable mit dem Namen `ID` übertragen oder mit der falschen Methode übermittelt wird, erhält die Variable `colname` den Standardwert `-1`.

`-1` kann dann eventuell eine Website mit einer Fehlermeldung anzeigen, und zwar nach diesem Schema:

```
if(colname == -1) { gehe zu… }
```

Läuft die Übertragung hingegen korrekt, wird der Inhalt von `$_GET['ID']` in die Variable `colname` geschrieben, und Sie erhalten den gewünschten Datensatz.

Abfrage testen

Zum Testen dieser Abfrage müssen Sie die Variable `ID` mit der Methode `GET` an das Dokument übertragen. Die Live-Ansicht von Dreamweaver verfügt über Funktionen, um diese Datenübertragung zu simulieren und die Dokumente auch unter realen Bedingungen zu testen.

Öffnen Sie dazu im Dokumentfenster die Browser-Navigation, indem Sie aus dem Menü Ansicht • Symbolleisten • Browser-Navigation auswählen.

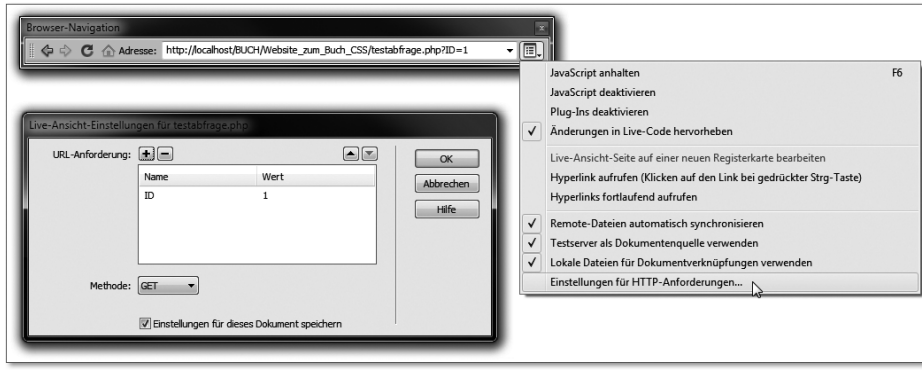


Abbildung 25.26 Live-Ansicht-Einstellungen in Dreamweaver

In der Dialogbox Einstellungen für HTTP-Anforderungen… können Sie Variablen und die Übertragungsmethode anlegen, die beim Umschalten in die Live-Ansicht übergeben werden sollen. Für unser Dokument benötigen wir die Variable `CONT_ID` mit dem Wert `2` und der Methode `GET`, mit der Variablen offen an die URL angehängt versendet werden.

In der Live-Ansicht sehen Sie jetzt Ihr Dokument mit der übergebenen Variablen `ID`. Probieren Sie eine andere Datensatz-ID aus, indem Sie diese einfach in das Parameterfenster Live-Ansicht-Einstellungen schreiben.

25.2.7 Einstellbare Variablentypen

In der Dialogbox Datensatzgruppe können Sie neben den URL-Parametern auch noch weitere Variablentypen zur Grundlage Ihrer Abfrage machen.

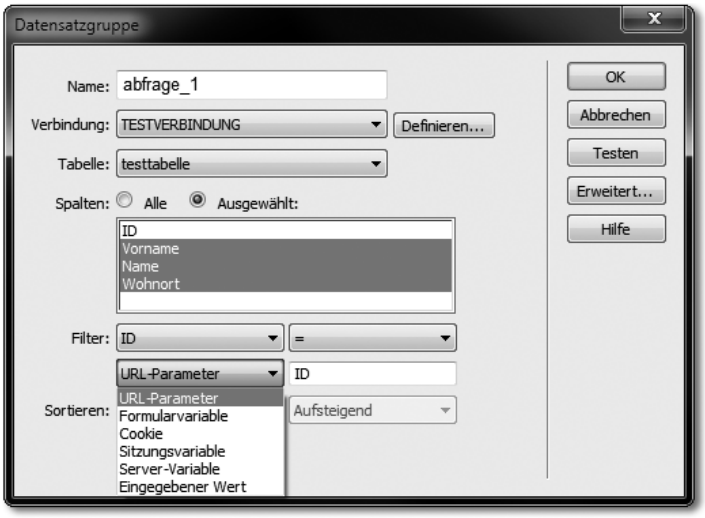


Abbildung 25.27 Abfrage mit verschiedenen Variablentypen definieren

Im Folgenden finden Sie eine Auflistung der in Dreamweaver für Abfragen verfügbaren Variablen und ihrer Einsatzzwecke. Sie haben sie bereits im Kapitel 20, »PHP mit Dreamweaver«, kennengelernt.

- URL-Parameter (`$_GET['VARIABLENNAME']`) werden mit der Methode `GET` übertragen und dienen in erster Linie zur Auswahl von Datensätzen aus einer Navigation, wobei die Variable und ihr Wert in einer Verlinkung an die URL angefügt werden.
- Formularvariablen (`$_POST['VARIABLENNAME']`) werden aus Formularen mit der Methode `POST` heraus versandt und hauptsächlich zum Einfügen von Datensätzen verwendet. Auch Suchanfragen werden in der Regel mit der Methode `POST` übertragen und benötigen eine Formularvariable als Abfragebasis.
- Sitzungsvariablen (`$_SESSION['VARIABLENNAME']`), auch *Sessionvariablen* genannt, werden in Sessions registriert, die mit PHP für jeden Seitenzugriff gestartet werden. Sitzungsvariablen müssen zunächst registriert werden, um dann für eine Abfrage zur Verfügung zu stehen.
- Cookie-Variablen (`$_COOKIE['VARIABLENNAME']`) ermöglichen die Abfrage von Cookies. Diese stellen ähnliche Funktionen wie Sessions zur Verfügung, müssen allerdings zwingend auf dem Clientrechner hinterlegt werden. In Cookies gespeicherte Variablen können Sie als Grundlage einer Abfrage ver-

wenden. Es besteht dabei das Risiko, dass ein User Cookies gesperrt hat. In diesem Fall funktionieren Abfragen auf Cookie-Basis nicht.

- Servervariablen (`$_SERVER['VARIABLENNAME']`) sind Variablen aus der Serverumgebung. Für Abfragen werden sie selten verwendet.

Mit einem eingegebenen Wert (`VARIABLENNAME`) legen Sie eine Variable mit einem festen Wert an und führen Ihre Abfrage mit diesem festen Wert durch.

25.2.8 Testen der Abfragen

Sie müssen Abfragen nicht in Ihr Dokument einfügen, um sie zu überprüfen. In der Dialogbox für Abfragen stellt Ihnen Dreamweaver eine Funktion zum Überprüfen Ihrer Abfragen zur Verfügung.



Abbildung 25.28 Überprüfung der Abfragen

Klicken Sie dazu nach Erstellen einer Abfrage im Dialogfenster Datensatzgruppe auf Testen, und geben Sie einen Parameter ein, dann öffnet sich ein weiteres Fenster mit den Ergebnissen Ihrer Abfrage. Besonders bei komplexen Abfragen ist es besser, vor dem Einfügen in ein Dokument die Abfragen auf ihre Funktion hin zu prüfen.

25.3 Dynamische Tabellen

Sehr häufig werden Datenfelder in Tabellen ausgegeben, um eine exakte Positionierung im Dokument zu erreichen. Dreamweaver unterstützt diesen Arbeitsschritt mit der Funktion Dynamische Tabelle.

25.3.1 Erstellen einer dynamischen Tabelle

Eben noch haben wir die Tabelle von Hand angelegt, nun machen wir das automatisch. Öffnen Sie dazu in der Einfügeleiste die Gruppe Anwendung. In dieser Gruppe werden die Funktionen für dynamische Websites je nach ausgewähltem Servertyp angezeigt – eine umfassende Erklärung der einzelnen Icons und Funk-

tionen finden Sie in Kapitel 29, »Menüs für dynamische Verhalten«, am Ende dieses Buches. Wählen Sie, wie in Abbildung 25.29 gezeigt, den Befehl DYNAMISCHE TABELLE aus.

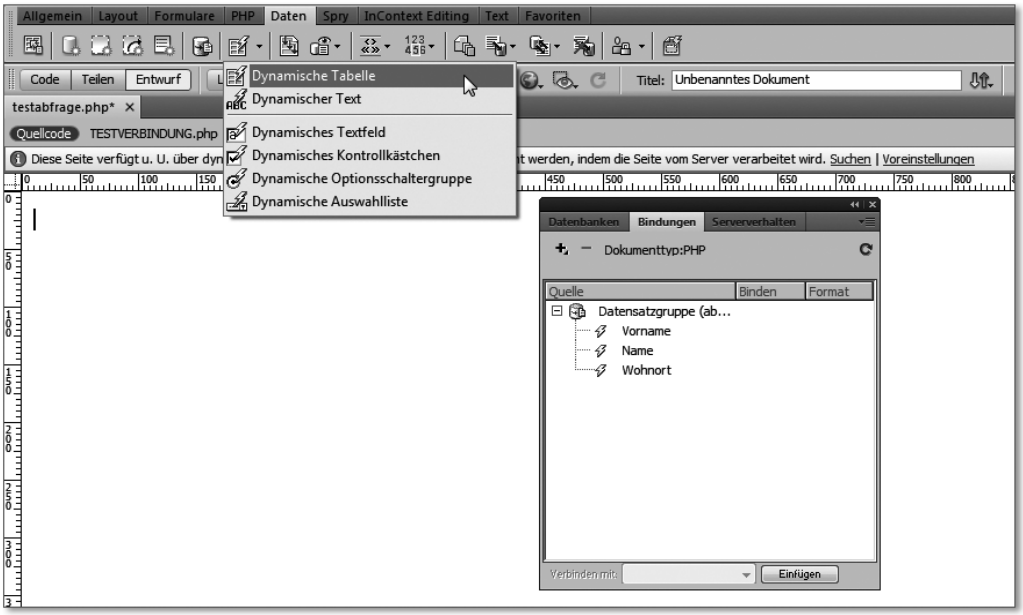


Abbildung 25.29 Einfügen einer dynamischen Tabelle

Wählen Sie in der angezeigten Dialogbox den zuvor angelegten Datensatz aus, und stellen Sie die Parameter für die Tabellen ein. Diese müssen nicht exakt sein, die Tabellen können auch problemlos nachträglich verändert werden.



Abbildung 25.30 Werte in der Dialogbox »Dynamische Tabelle« auswählen

Die dynamische Tabelle wird Ihnen im Dokumentfenster mit den üblichen Platzhaltern angezeigt. Sie können diese nach Belieben bearbeiten. Dynamische Tabel-

len sind wiederholte Bereiche und werden je nach Anzahl der Datensätze – genau wie die normalen wiederholten Bereiche – dupliziert dargestellt.

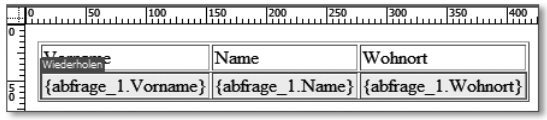


Abbildung 25.31 Eine dynamische Tabelle wird ins Dokument eingefügt.

In der Live-Ansicht können Sie sich die Seite mit den dynamisch eingefügten Inhalten ansehen.

Vorname	Name	Wohnort
Richard	Beer	Hamburg
Susann	Gailus	Hamburg

Abbildung 25.32 Live-Ansicht mit den Inhalten einer dynamischen Tabelle

25.3.2 Dynamische Tabelle mit Serververhalten

Dynamische Tabellen können Sie auch über die Bedienfelder Serververhalten und Wiederholte Bereiche erstellen. Dies bietet sich an, wenn Sie einen Bereich in Ihrem Dokument layouten möchten, um ihn je nach Anzahl der Datensätze im Dokument zu wiederholen. Mit dem Befehl Dynamische Tabelle haben Sie hier keine gute Kontrolle über das Layout.

25.4 Wiederholte Bereiche mit AP-Elementen

Etwas mehr Vorarbeit ist nötig, wenn Sie mit AP-Elementen wiederholte Bereiche anlegen möchten. Die Vorgabe bei AP-Elementen ist eine absolute Positionierung. Damit können Sie zwar problemlos die AP-Elemente wiederholen, diese stehen allerdings an der gleichen Stelle, da bei jedem Element die gleichen absoluten Koordinaten angegeben sind. Somit ist immer nur das oberste der Elemente sichtbar.

Mit Bordmitteln im Vorschaumodus ist es kaum möglich, dieses Problem in den Griff zu bekommen. Wie Sie gleich sehen werden, sind die Änderungen im Quelltext jedoch nicht sehr umfangreich und gehen leicht von der Hand.

Hier die Stilvorgaben ohne Modifikation:

```
<style type="text/css">
#CONTAINER_1 {
   position:absolute;
```

```
   left:10px;
   top:10px;
   width:200px;
   height:200px;
   z-index:1;
   background-color: #FFCC00;
}
#WIEDERHOLUNG {
   position:absolute;
   left:5px;
   margin-top:5px;
   width:190px;
   height:20px;
   z-index:1;
   background-color: #FFFFCC;
   font-family: Arial, Helvetica, sans-serif;
   font-size: 12px;
   color: #333;
}
</style>
```

Listing 25.6 CSS, wie es Dreamweaver anlegt

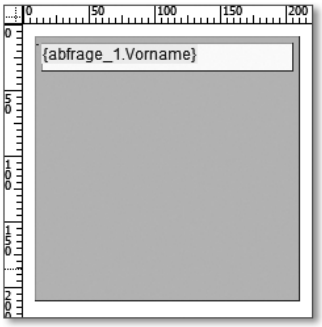


Abbildung 25.33 Wiederholter Bereich mit AP-Element im Layout

Den dazugehörigen Quelltext mit wiederholtem Bereich sehen Sie im folgenden Listing:

```
<div id="CONTAINER_1">
  <?php do { ?>
    <div id="WIEDERHOLUNG"><?php echo $row_abfrage_1['Vorname']; ?>
    </div>
    <?php } while ($row_abfrage_1 = mysql_fetch_assoc($abfrage_1)); ?>
</div>
```

Listing 25.7 Wiederholter Bereich mit AP-Element

Um ein ähnliches Verhalten wie bei Tabellen zu erreichen, müssen Sie hier in die Stylesheets eingreifen und diese ändern. In der CSS-Regel muss zunächst die Positionierung von absolut auf relativ umgestellt werden.

```
<style type="text/css">
#CONTAINER_1 {
   position:absolute;
   left:10px;
   top:10px;
   width:200px;
   height:200px;
   z-index:1;
   background-color: #FFCC00;
}
#WIEDERHOLUNG {
   position:relative;
   left:5px;
   margin-top:5px;
   width:190px;
   height:20px;
   z-index:1;
   background-color: #FFFFCC;
   font-family: Arial, Helvetica, sans-serif;
   font-size: 12px;
   color: #333;
}
</style>
```

Listing 25.8 CSS für AP-Element nach Änderung



Abbildung 25.34 AP-Element mit wiederholtem Bereich und relativer Positionierung

Wie Sie in der Abbildung sehen, werden nun die Elemente untereinander ausgegeben. Wir benötigen jetzt noch einen Abstand zwischen den einzelnen Elementen.

In der Abbildung wird ebenfalls sichtbar, dass sich nun ein weiteres Problem ergibt: Tabellen passen sich in der Höhe automatisch dem Inhalt an. CSS macht dies nicht, auch nicht, wenn Sie als Höhe `100 %` vorgeben.

Die Lösung ist, gar keine Höhe anzugeben und stattdessen unten einen Abstand des Randes auf den Inhalt (`padding`) zu definieren. Das müssen Sie von Hand im Quelltext abändern:

```
#CONTAINER_1 {
   position:absolute;
   left:10px;
   top:10px;
   width:200px;
   z-index:1;
   background-color: #FFCC00;
   padding-bottom:5px;
}
```

Listing 25.9 Einstellungen für das übergeordnete AP-Element

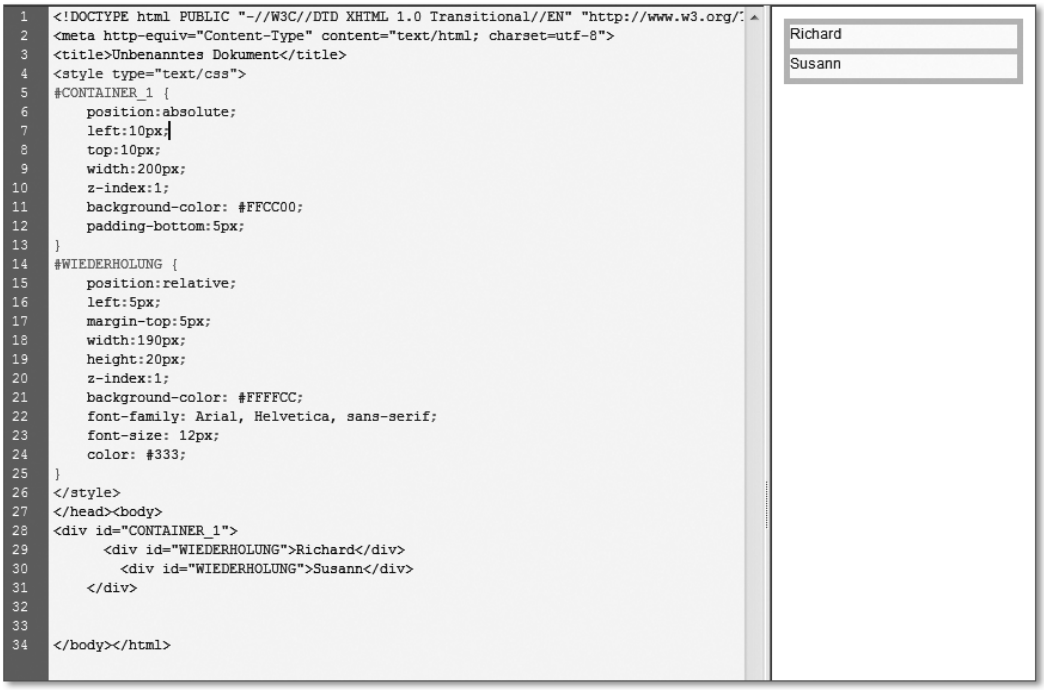


Abbildung 25.35 Fertiges Element mit Live-Code

Standardkonform bleiben

Wenn Sie sich an Kapitel 11, »CSS in Dreamweaver«, erinnern, werden Sie sich wundern, warum wir hier eine ID verwenden. Unsere Aussage, dass eine ID nur einmal verwendet werden darf, hat noch immer Gültigkeit. Leider legt Dreamweaver, wenn man AP-Elemente zeichnet, immer eine ID an und keine Klasse. Wir müssen also auch hier wieder in den Code eingreifen und aus dem ID-Selektor `#Wiederholung` mit ».« eine Klasse machen. Natürlich muss die Klasse dann auch angewendet werden. Der fertige Code sieht dann wie folgt aus:

```
<style type="text/css">
#CONTAINER_1 {
   position:absolute;
   left:10px;
   top:10px;
   width:200px;
   z-index:1;
   background-color: #FFCC00;
   padding-bottom:5px;
}
.WIEDERHOLUNG {
   position:relative;
   left:5px;
   margin-top:5px;
   width:190px;
   height:20px;
   z-index:1;
   background-color: #FFFFCC;
   font-family: Arial, Helvetica, sans-serif;
   font-size: 12px;
   color: #333;
}
</style>
</head>

<body>
<div id="CONTAINER_1">
  <?php do { ?>
    <div class="WIEDERHOLUNG"><?php echo $row_abfrage_1['Vorname']; ?>
    </div>
    <?php } while ($row_abfrage_1 = mysql_fetch_assoc($abfrage_1)); ?>
</div>
</body>
</html>
```

Listing 25.10 Einstellungen für das übergeordnete AP-Element

Nachteile automatisch generierter wiederholter Bereiche

Wenn Sie sich den Quelltext der wiederholten Bereiche etwas näher ansehen, werden Sie feststellen, dass Dreamweaver eine `do-while`-Schleife verwendet, um den Bereich zu erzeugen. Das hat den ganz entscheidenden Nachteil, dass der wiederholte Bereich *immer* mindestens einmal ausgegeben wird, auch wenn es keine Inhalte gibt. Im schlimmsten Fall gibt der Browser eine PHP-Fehlermeldung aus.

Sie können das umgehen, indem Sie die Schleife in eine `while`-Schleife abändern:

```
<?php do { ?>
<div class="WIEDERHOLUNG"><?php echo $row_abfrage_1['Vorname']; ?>
</div>
<?php } while ($row_abfrage_1 = mysql_fetch_assoc($abfrage_1)); ?>
```

Listing 25.11 Von Dreamweaver erstellte Schleife

```
<?php while ($row_abfrage_1 = mysql_fetch_assoc($abfrage_1)) { ?>
<div class="WIEDERHOLUNG"><?php echo $row_abfrage_1['Vorname']; ?>
</div>
<?php }  ?>
```

Listing 25.12 Abgeänderte Schleife

Zusätzlich muss im Kopf des Dokuments ein Befehl auskommentiert werden:

```
// $row_abfrage_1 = mysql_fetch_assoc($abfrage_1);
```

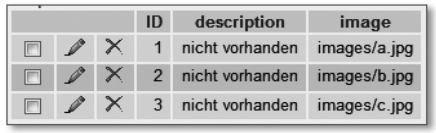
Dreamweaver erkennt die abgeänderte Schleife dann allerdings nicht mehr als eigenen Code. Jede weitere Modifikation muss von Hand im Quelltext vorgenommen werden.

25.5 Bilder dynamisch einfügen

Neben Texten sind Bilder die häufigsten Elemente auf Websites. Mit Dreamweaver können Sie Bilder ganz einfach dynamisch einfügen.

25.5.1 Vorbereitungen für dynamisch eingefügte Bilder

Genau genommen werden nicht Bilder in eine Datenbank eingefügt, sondern immer nur die Verknüpfungen zu einem Bild. Man kann zwar Bilder auch direkt in einer Datenbank ablegen, aus Performancegründen sollte man das aber wann immer möglich vermeiden.



	ID	description	image
	1	nicht vorhanden	images/a.jpg
	2	nicht vorhanden	images/b.jpg
	3	nicht vorhanden	images/c.jpg

Abbildung 25.36 Nicht die Bilder, sondern die Pfadangaben werden in der Datenbank gespeichert.

Damit die Verlinkung der Bilder mit relativen Pfaden funktioniert, haben wir einen Unterordner mit dem Namen *images* angelegt, in dem die Bilder gespeichert sind.

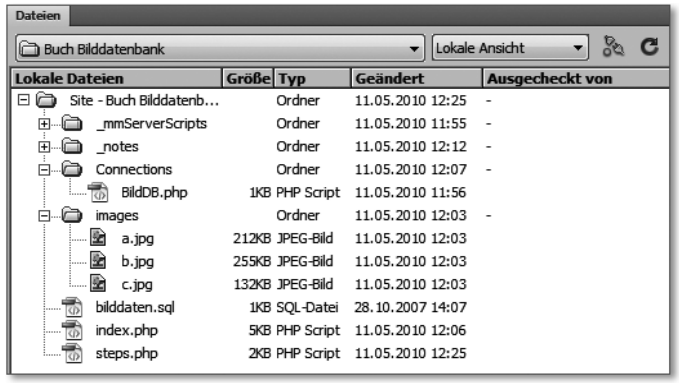


Abbildung 25.37 Neuer Ordner für die Testbilder

In unserem Fall müssen wir – da es keinen realen Webserver für die Website gibt – mit relativen Pfaden arbeiten. Wenn eine Website später veröffentlicht werden soll, ist es besser, mit absoluten Pfaden zu den Bildern zu arbeiten, außer, wir wissen genau, von welchem Ort auf dem Webserver die Bilder eingebunden werden.

Diese Methode wird bei einem leeren Datensatz die übliche Fehlerdarstellung für fehlende Bilder erzeugen, da das `Image`-Tag ja trotzdem an HTML ausgegeben wird. Geben Sie daher in Ihrer Datenbank als Standardvorgabe den Pfad zu einem transparenten GIF an, bis dieses durch ein richtiges Bild überschrieben wird.

25.5.2 Bilder aus Datenbanken verlinken

Wir haben bereits mit Bild-Platzhaltern gearbeitet. Bei der dynamischen Website-Erstellung kommt jetzt diese Technik zum Einsatz.

Bild-Platzhalter einfügen

[o] Um Ihnen die Arbeitsweise mit dynamischen Bildern zu verdeutlichen, haben wir eine kleine Beispieldatenbank angelegt. Die Bilder werden als Hyperlink in ein

varchar-Feld eingetragen. Alle Bilder haben eine Abmessung von 350 × 500 Pixeln. Auch diese Dateien finden Sie auf der DVD zum Buch im Ordner *Beispiele/Bilddatenbank*.

	Feld	Typ	Kollation	Attribute	Null	Standard	Extra	Aktion
☐	ID	int(11)			Nein	*Kein*	auto_increment	
☐	description	varchar(200)	latin1_swedish_ci		Nein			
☐	image	varchar(200)	latin1_swedish_ci		Nein			

Abbildung 25.38 Angelegte Bilddatenbank

			ID	description	image
☐	✎	✕	1	nicht vorhanden	images/a.jpg
☐	✎	✕	2	nicht vorhanden	images/b.jpg
☐	✎	✕	3	nicht vorhanden	images/c.jpg

Abbildung 25.39 Einträge in der Bilddatenbank

Um einen Bild-Platzhalter einzufügen, gehen Sie in der Einfügeleiste auf ALLGEMEIN und klicken dann auf BILDER • BILD-PLATZHALTER.

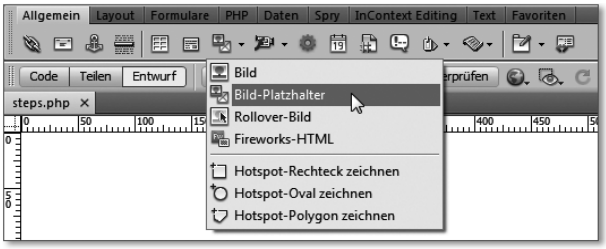


Abbildung 25.40 Einen Bild-Platzhalter ins Dokument einfügen

Benennen Sie den Bild-Platzhalter, und legen Sie die Abmessungen des zukünftigen Bildes fest.



Abbildung 25.41 Festlegen der Platzhalter-Eigenschaften

Nachdem der Bild-Platzhalter eingefügt wurde, müssen Sie noch die notwendige Abfrage erstellen. Wir benötigen die Tabelle *bilddaten*, da sich in dieser Tabelle

die Datenfelder für die Bilder befinden. Vorher muss natürlich eine Verbindung zur Datenbank hergestellt sein.

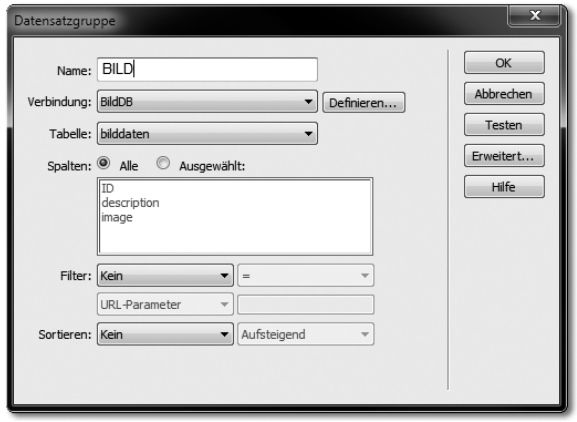


Abbildung 25.42 Eine Abfrage für das Bild erstellen

Wenn Sie eine Abfrage erstellt haben, können Sie den Datensatz *image* einfach mit der Maus auf den Bild-Platzhalter ziehen. Manchmal werden die Maße aus dem Bild-Platzhalter nicht korrekt übernommen. Tragen Sie in einem solchen Fall die Abmessungen des Bildes im Eigenschafteninspektor ein. Wenn Sie die genauen Maße des Bildes nicht kennen, müssen die Abmessungen auch nicht angegeben werden. Um einen schnelleren Seitenaufbau zu garantieren, sollte dies aber nach Möglichkeit geschehen.

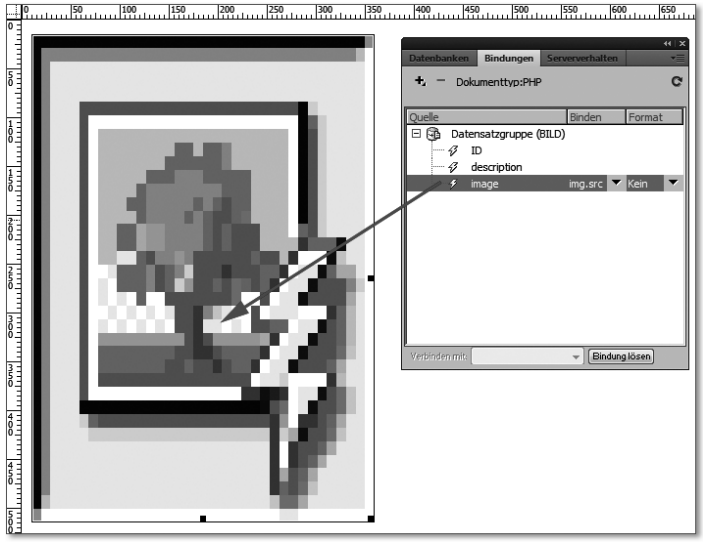


Abbildung 25.43 Dem Bild-Platzhalter wird ein Datensatz zugewiesen.

Im Quelltext wird nun unter anderem Folgendes eingetragen:

```
<img src="<?php echo $row_BILD['image']; ?>" alt="Tauchen ist
schön..." name="IMAGE" width="350" height="500" />
```

Sie sehen, dass anstelle des Zielpfads zum Bild ein PHP-Skript mit der Datensatzvariablen zum Bild 1 abgelegt wird. In der Live-Ansicht sehen Sie das eingefügte Bild korrekt angezeigt, die Datenbankanbindung hat funktioniert.

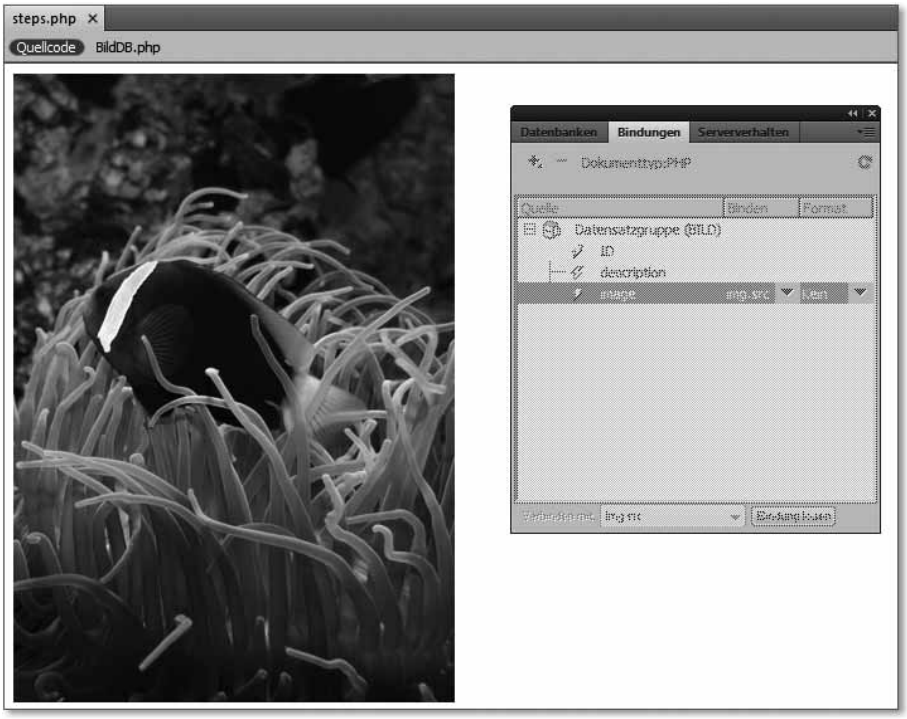


Abbildung 25.44 Das dynamisch angezeigte Bild in der Live-Ansicht

Bilder in wiederholten Bereichen

Selbstverständlich können Sie Bilder auch in wiederholten Bereichen verwenden. In einem Shopsystem können Sie so z. B. alle Produktbilder mit den relevanten Daten anzeigen lassen und müssen die CSS-Stile und Tabellen für die Produkte nur einmal anlegen.

Eine weitere Möglichkeit, dynamische Bilder einzubinden, ergibt sich durch das Anfügen von HTML-Attributen an dynamische Daten. Zu diesem Thema lesen Sie bitte den Abschnitt 25.7, »HTML-Attribute dynamisch generieren«, weiter unten. Welche der Methoden Sie verwenden, bleibt letztlich Ihnen überlassen.

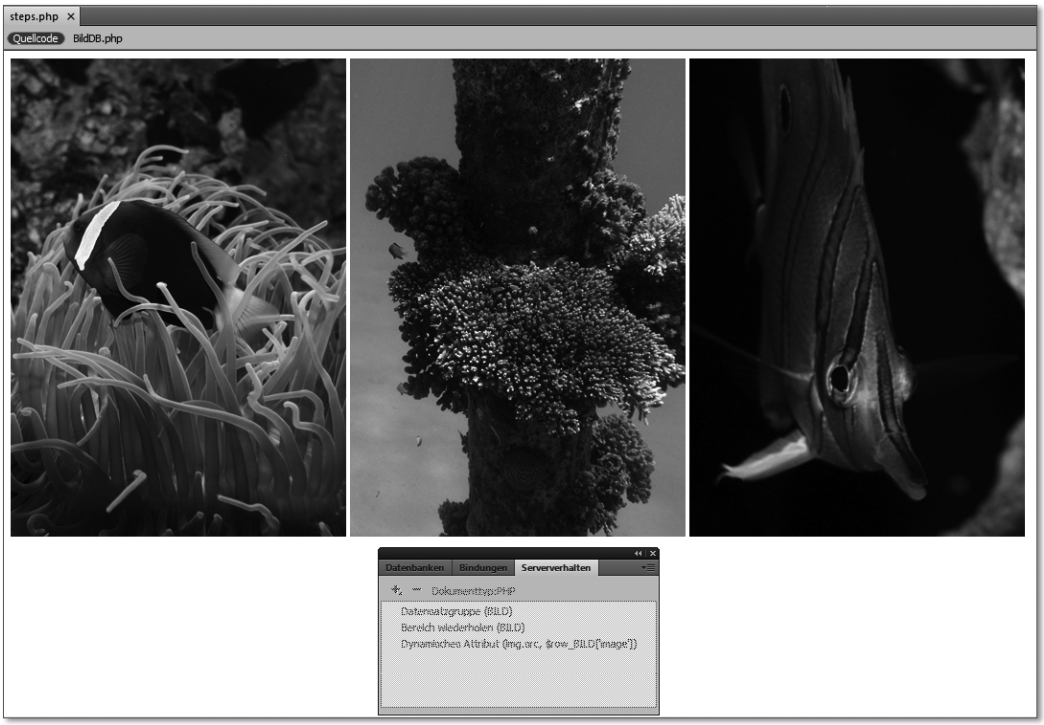


Abbildung 25.45 Dynamisch eingeblendete Bilder in wiederholten Bereichen auf einer Seite

25.6 In Datensätzen navigieren

Bei mehreren verfügbaren Datensätzen müssen Sie häufig zwischen diesen navigieren. Besonders bei einer Bilddatenbank kann dies der Fall sein.

25.6.1 Datensatznavigation erstellen

Dreamweaver nimmt Ihnen auch hier mühevolle Arbeit ab und ermöglicht es, eine Datensatznavigation mit wenigen Klicks komplett zu erstellen. Die anschließenden optischen Anpassungen sind dann ein Kinderspiel.

Um dieses Serververhalten zu testen, lassen wir uns einfach das Bild 1 (*image*) aus der Tabelle *bilddaten* ausgeben. Wir haben das Layout um einige Kleinigkeiten erweitert, die aber für die Funktion keine Rolle spielen.

In der Einfügeleiste können Sie unter DATEN über das Icon ❶ eine DATENSATZGRUPPEN-NAVIGATIONSLEISTE einfügen. Die anschließende Warnung von Dreamweaver können Sie ignorieren.

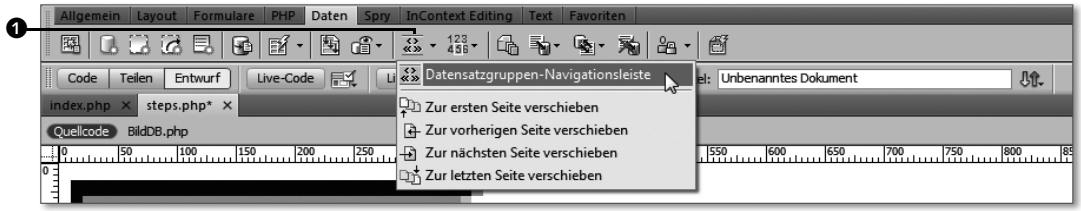


Abbildung 25.46 Eine Datensatzgruppen-Navigationsleiste einfügen

Datensatz definieren

Dreamweaver möchte nun von Ihnen wissen, auf welchen Datensatz sich die Datensatzgruppen-Navigationsleiste beziehen soll. Wenn Sie mehrere verschiedene Datensätze in Ihrem Dokument anzeigen, können Sie aus der Dialogbox DATENSATZGRUPPEN-NAVIGATIONSLEISTE den gewünschten auswählen.

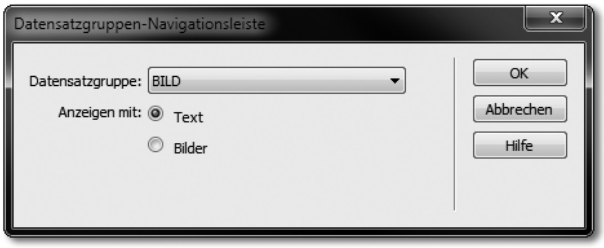


Abbildung 25.47 Eine Datensatzgruppen-Navigationsleiste wird ausgewählt.

Wir formatieren die eingefügte Datensatzgruppen-Navigationsleiste linksbündig. Sie können sie nach Belieben gestalten und vollständig an Ihr Layout anpassen.



Abbildung 25.48 Eine Datensatzgruppen-Navigationsleiste wird eingefügt und formatiert.

Im Browser testen

Im Browser können Sie nun zwischen den einzelnen Bildern navigieren. Dieses Serververhalten können Sie in der Live-Ansicht nicht testen, da Hyperlinks hier nicht funktionieren.



Abbildung 25.49 Über die Datensatz-Navigationsleiste kann man durch die Bilder navigieren.

25.6.2 Bereiche in Abhängigkeit von Daten anzeigen

Beim Aufbau von dynamischen Websites kommt es immer wieder vor, dass man Bereiche oder Elemente in Abhängigkeit von bestimmten Daten anzeigen möchte. In einem Shopsystem kann das zum Beispiel ein ganzer Block mit zusätzlichen Angaben sein, der nur dann angezeigt wird, wenn in diesem Block auch Einträge vorhanden sind. Würde man einen ganzen Block mit Tabellen und allem, was dazugehört, trotz fehlender Daten anzeigen, ergäben sich im Dokument unschöne leere Bereiche.

Abhängige Bereiche

Für derartige Anzeigeoptionen bietet sich die Anzeige in Abhängigkeit zum Dateninhalt an. Dreamweaver verfügt auch hier über einige Features. Wir haben, um die Vorgehensweise zu verdeutlichen, in einem Dokument einen Datensatz mit Bild angezeigt.

Falls es keinen Datensatz gibt, soll der Text »Keine Einträge vorhanden« erscheinen. Dies wird zum Beispiel bei einem Gästebuch mit Bildergalerie benötigt. Benutzer können Bilder hinterlegen – müssen es aber nicht.

Anzeigeoptionen von abhängigen Bereichen

Um einen von den Daten abhängigen Bereich einzufügen, müssen Sie zunächst den gewünschten Bereich vollständig auswählen und anschließend in der Einfügeleiste unter DATEN mit dem Icon ❶ die Option ANZEIGEN, WENN DATENSATZGRUPPE NICHT LEER IST oder ANZEIGEN, WENN DATENSATZGRUPPE LEER IST auswählen. Im Beispiel wurde die gesamte Tabelle ausgewählt. Diese soll bei vorhandenen Datensätzen angezeigt werden.

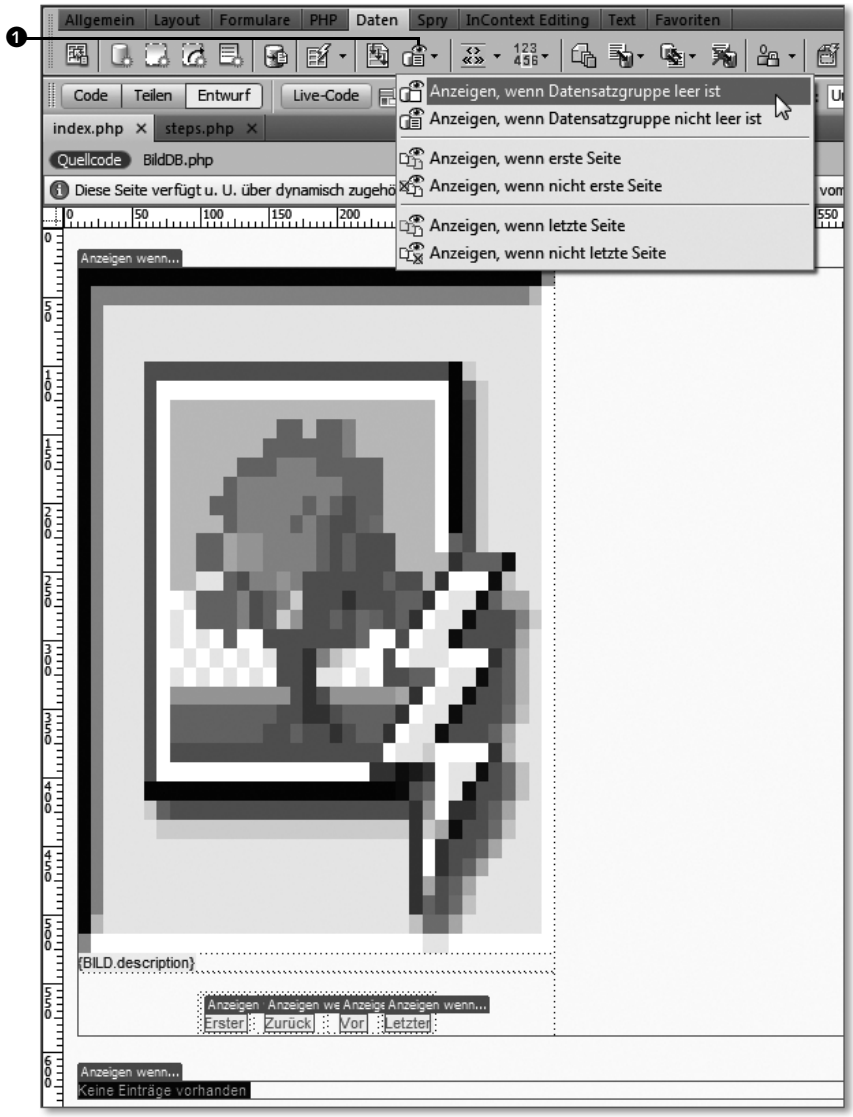


Abbildung 25.50 Abhängige Bereiche werden eingefügt.

Wie bei fast allen anderen Serververhalten auch müssen Sie jetzt in der Dialogbox Anzeigen, wenn Datensatzgruppe (nicht) leer ist den gewünschten Datensatz anwählen.



Abbildung 25.51 Auswahl des Datensatzes, von dem die Anzeige der Bereiche abhängig sein soll

Live-Ansicht zum Testen

In der Live-Ansicht wird der Bereich je nach Vorhandensein des Datensatzes angezeigt. Ist kein Inhalt vorhanden, wird er nicht mit ausgegeben. Sie können dies testen, indem Sie eine nicht vorhandene Datensatz-ID als URL-Variable übergeben.



Abbildung 25.52 Nur der erste der optionalen Bereiche wird angezeigt, der zweite nicht.

25.6.3 Datensatzanzahl anzeigen

Immer wieder kommt es bei dynamischen Websites vor, dass Sie die Anzahl der vorhandenen Datensätze anzeigen wollen. Dies können Sie erreichen, indem Sie aus der Einfügepalette die entsprechende Option DATENSÄTZE GESAMT auswählen.

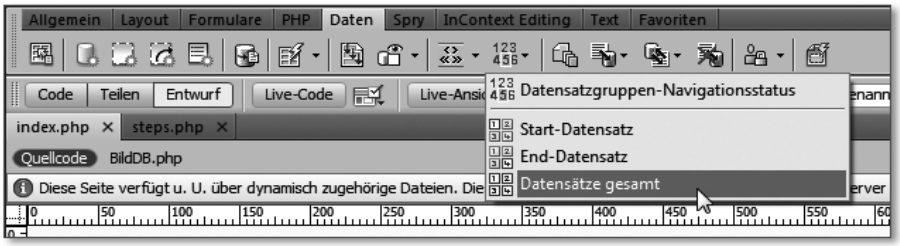


Abbildung 25.53 Die Datensatzanzahl anzeigen lassen

Für eigene Datensatznavigationen können Sie über dieses Menü auch einen Navigationsstatus anzeigen lassen, damit Sie wissen, in welchem Datensatz Sie sich aktuell befinden.

25.6.4 Individuelle Datensatznavigationen erstellen

Weiterhin finden Sie in der Einfügeleiste ein Menü, um individuelle Datensatznavigationen zu erstellen. Legen Sie dafür vorher eine Abfrage an, die eine bestimmte Anzahl an Datensätzen einer Tabelle ausgibt. Fügen Sie danach Grafiken oder Texte ein, um zum ersten, letzten, vorherigen und nächsten Datensatz zu navigieren. Markieren Sie einfach den gewünschten Bereich, und klicken Sie in der Einfügepalette auf die passende Aktion.

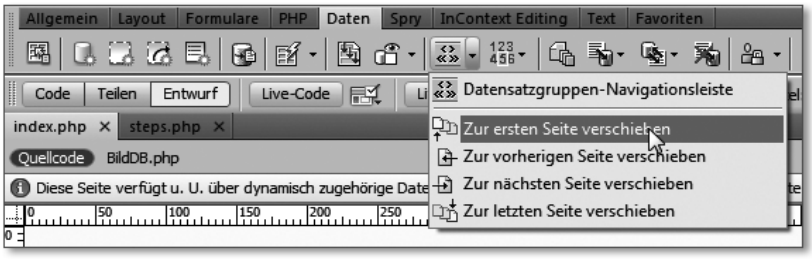


Abbildung 25.54 Navigation in einem Datensatz erstellen

Die zuvor ausgewählte Grafik oder der Text wird von Dreamweaver mit einer Aktion zum Aufruf eines anderen Datensatzes hinterlegt. Dabei wird die aktuelle Seite erneut aufgerufen und eine ID für den anzuzeigenden Datensatz übergeben. Das Prinzip ist das gleiche, wie wenn Sie eine ID als URL-Parameter übergeben, um auf einen bestimmten Detaildatensatz zu verlinken.

25.7 HTML-Attribute dynamisch generieren

Wir haben in Abschnitt 25.5.2, »Bilder aus Datenbanken verlinken«, ein Bild aus einer Datenquelle eingefügt, indem wir die Bildquelle innerhalb eines Tags durch ein PHP-Skript einsetzen ließen.

Mit Dreamweaver können Sie auf einfache Weise auch jedes HTML-Attribut durch dynamisches Einfügen manipulieren. Wir erklären das noch einmal am Beispiel eines Bildes, wählen jedoch dieses Mal eine andere Vorgehensweise.

25.7.1 Eigenschaften aus Datenquellen einfügen

Zunächst muss wieder ein Datensatz im Bedienfeld BINDUNGEN angelegt werden. In unserem Beispiel wurde die Tabelle *bilder* abgefragt und ein Bild-Platzhalter eingefügt (wie auch im vorherigen Abschnitt). Wenn Sie jetzt eine Bildquelle über den Eigenschafteninspektor auswählen möchten, stehen Ihnen bei dynamischen Seiten in der folgenden Dialogbox zwei Möglichkeiten zur Auswahl: Sie können eine Bildquelle aus dem DATEISYSTEM wählen, was bei statischen Seiten die übliche Methode ist. Zusätzlich besteht jedoch die Möglichkeit, eine DATENQUELLE als Bildquelle anzugeben.

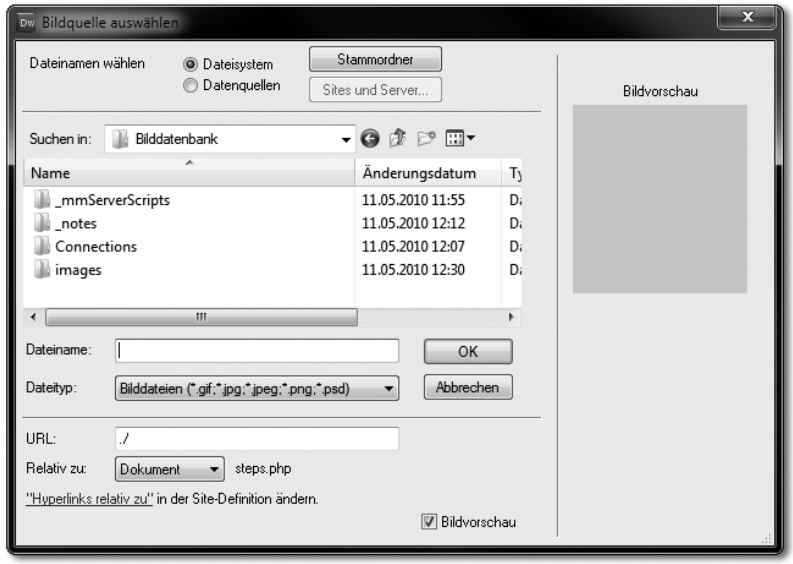


Abbildung 25.55 Bild aus Datenquelle erstellen

Schalten Sie in der Dialogbox BILDQUELLE AUSWÄHLEN (Abbildung 25.56) auf DATENQUELLEN um. Anschließend werden Ihnen die Inhalte der vorher angeleg-

ten Bindungen angezeigt, und Sie können den Datensatz mit den Bildinhalten anwählen und einfügen.

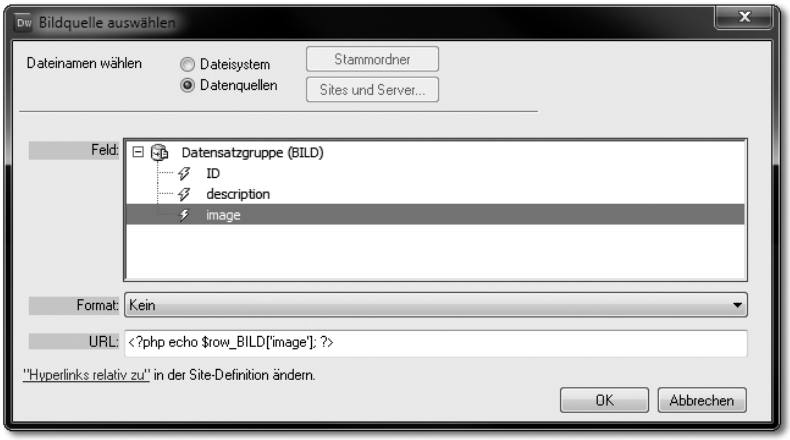


Abbildung 25.56 Datensatz auswählen

Jetzt zeigt der Eigenschafteninspektor im Feld Quelle den kompletten PHP-Befehl als Referenz an.



Abbildung 25.57 Das Bild aus der Datenquelle wird eingefügt und der PHP-Befehl im Eigenschafteninspektor unter »Quelle« angezeigt.

Dynamische Attribute

Dynamische Daten als Attribute von HTML-Tags können Sie auch über das Bedienfeld Tag zuweisen. Wenn Sie eine dynamische Seite erstellen, erscheint rechts neben den HTML-Attributen, die Sie mit dynamischen Daten hinterlegen können, ein kleines Icon mit einem Blitz. Klicken Sie darauf, um die Auswahl aus einer zuvor angelegten Datenquelle zu treffen.

Durch das dynamische Anbinden von HTML-Attributen stehen Ihnen viele neue Möglichkeiten offen. Um die Vorgehensweise zu verdeutlichen, folgt hier nun ein kleines Anwendungsbeispiel.



Abbildung 25.58 Dynamische Daten über das Bedienfeld »Tag« hinzufügen

[▪] *Schritt-für-Schritt: Grafische Auswertung einer Datentabelle mit PHP*

[◉] Im Folgenden wollen wir an einem Beispiel den Einsatz dynamischer Attribute in der Praxis verdeutlichen. Die SQL-Datei zur Übung finden Sie auf der DVD im Ordner *Beispiele/Diagramme*.

[»] **Nur Beispielbilder**

Bedenken Sie bitte, dass die Website im Buch nicht vollständig aufgebaut ist. Damit ist gemeint, dass nicht für alle möglichen Einträge Bilder vorhanden sind, sondern wir mit wenigen Beispielbildern auskommen wollen und müssen. Es reicht allerdings, um eine handfeste Grundlage für eigene Arbeiten zu legen und die Prinzipien zu erlernen.

1 *Analyse der Datenbanktabelle*

In einer Datenbank wird monatlich die Serverauslastung in Prozentwerten hinterlegt. Ein Kunde möchte diese Auslastung als Balkendiagramm angezeigt bekommen.

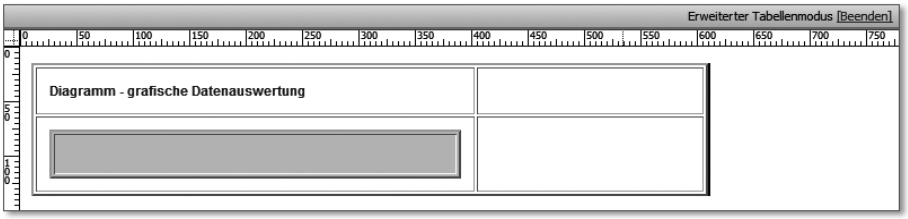
Die Datenbank in unserem Beispiel hat eine einfache Struktur. In der Tabelle *diagramm* der Datenbank *diagramme* finden sich drei Datenreihen: eine ID als Primärschlüssel, ein Feld für die Angaben in Prozent und ein weiteres Feld mit der Angabe für den Zeitraum.

			ID	Prozent	Zeitraum
☐	✎	✕	1	25	Januar
☐	✎	✕	2	75	Februar
☐	✎	✕	3	56	März
☐	✎	✕	4	98	April
☐	✎	✕	5	47	Mai
☐	✎	✕	6	3	Juni
☐	✎	✕	7	51	Juli

2 *Erstellen einer HTML-Tabelle für das Balkendiagramm*

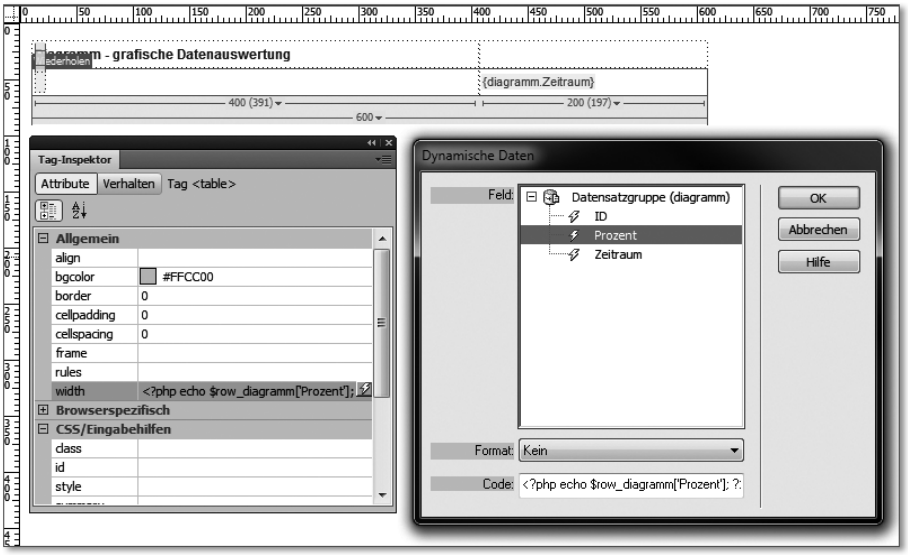
Tabellenbreiten können in Pixel oder in Prozent angegeben werden. Prozentwerte der Tabellenbreiten können Sie auch dynamisch generieren. In unserem Beispiel werden mit Hilfe der Prozentwerte aus der Datenbank die Breiten einer Tabelle bestimmt.

Dazu benötigen Sie zunächst zwei HTML-Tabellen, wie in der folgenden Abbildung gezeigt. Die einzelne orangefarbene Tabellenzeile hat eine Breite von 100 %.



3 *Einfügen der dynamischen Daten für die Bestimmung der Tabellenbreite*

Markieren Sie die gesamte innen liegende Tabelle, und wählen Sie im Bedienfeld TAG für die Tabellenbreite anstelle des Prozentwertes 100 den vorher abgefragten Datensatz *Prozent* aus. Wie immer müssen Sie vor diesem Schritt eine Datenbankverbindung und eine Datensatzabfrage erstellen. Klicken Sie dazu auf das kleine Blitzsymbol am rechten Rand des Attributfeldes.



4 *Prozentzeichen im Bedienfeld einfügen*

Ganz wichtig ist nun, dass Sie hinter dem im Bedienfeld TAG erscheinenden PHP-Skript von Hand ein %-Zeichen einfügen. Ansonsten werden die Diagrammbalken nicht in Prozent, sondern in Pixelwerten angezeigt.

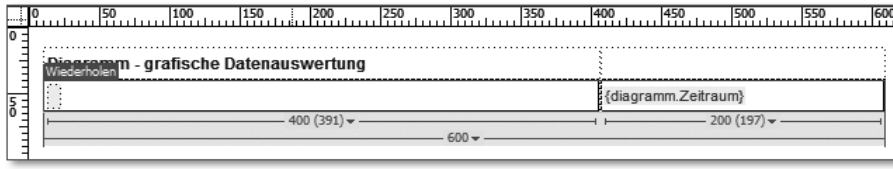
```
<td><table width="<?php echo $row_diagramm['Prozent']; ?>%" border="0" cellpadding="0" cellspacing="0" bgcolor="#FFCC00">
    <tr>
      <td> </td>
    </tr>
        </table></td>
<td class="date"><?php echo $row_diagramm['Zeitraum']; ?></td></tr>
```

5 *Wiederholten Bereich erstellen*

Fügen Sie nun der unteren Tabellenzeile noch den Datensatz mit der Bezeichnung für den Zeitraum hinzu, und erstellen Sie aus der gesamten unteren Tabellenreihe einen wiederholten Bereich mit allen Datensätzen. Eventuell müssen Sie hierbei etwas im Quelltext nachhelfen. Der Quelltext des wiederholten Bereichs sieht wie folgt aus:

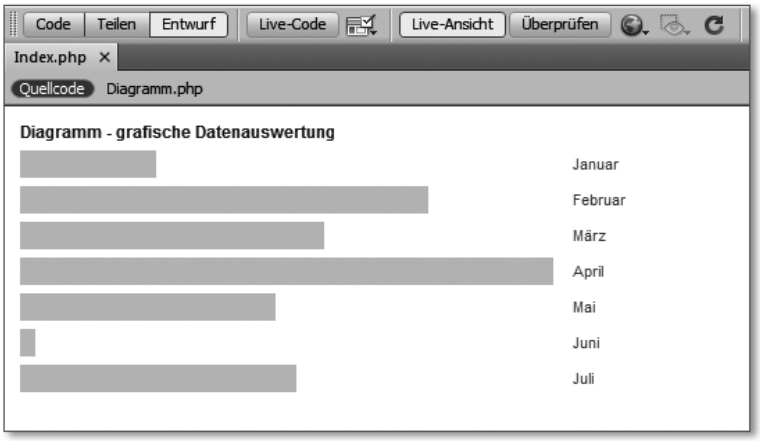
```
<table width="600" border="0" cellspacing="0" cellpadding="3">
  <tr>
    <td width="400" class="headline">Diagramm –
 grafische Datenauswertung </td>
    <td width="200" class="headline"> </td>
  </tr>

    <?php do { ?><tr>
      <td><table width="<?php echo $row_diagramm['Prozent']; ?>%"
      border="0" cellpadding="0" cellspacing="0" bgcolor="#FFCC00">
          <tr>
            <td> </td>
          </tr>
              </table></td>
      <td class="date"><?php echo $row_diagramm['Zeitraum']; ?>
      </td></tr>
      <?php } while ($row_diagramm =
                      mysql_fetch_assoc($diagramm)); ?>
</table>
```



6 *Abfrage prüfen in Live-Ansicht*

Vorausgesetzt, dass alles korrekt angelegt wurde, können Sie jetzt in der Live-Ansicht Ihr aus den Werten der Serverauslastung dynamisch generiertes Diagramm bestaunen.



7 *Darstellung des Diagramms im Browser prüfen*

Auch im Browser werden die orangefarbenen Balken korrekt dargestellt. Viel schwerer als das Anlegen eines Diagramms mit Excel ist das Erzeugen dieser dynamischen Anwendung auch nicht, und Sie können die Ergebnisse direkt online publizieren. Viele Statistikskripte für Websites funktionieren nach diesem Prinzip.

Mit PHP können auch Grafiken erstellt werden. Für diese einfachen Diagramme müssen Sie sich jedoch nicht mit der Programmierung von Grafiken in PHP herumschlagen. ■

25.7.2 Daten in der URL mit Hyperlinks übergeben

In den meisten dynamischen Websites müssen Daten zwischen Dokumenten ausgetauscht werden, um zum Beispiel Detaildatensätze anzuzeigen. Wie bereits mehrfach erläutert, müssen in solchen Fällen die Parameter bzw. die Variablen mit der Datensatz-ID an das aufzurufende Dokument übergeben werden. Diese werden dann als URL-Parameter an die URL angehängt.

Daten als URL-Parameter übergeben

Solche Parameter können von Hand im Eigenschafteninspektor eingetragen oder dynamisch aus einer Datenquelle generiert werden. In der Praxis ist die zweite

Methode wesentlich gebräuchlicher und komfortabler. Sie müssen sich dann nicht mehr um die Zuweisung einzelner Datensätze kümmern. PHP und MySQL erledigen das für Sie.

Im nun folgenden Beispiel wurde die *testdatenbank* um das Feld VITA erweitert. Im Feld VITA befindet sich ein längerer Blindtext. Das Dokument *auswahl.php* zeigt eine Auswahlliste an. Ein Klicken auf den Namen ruft das Dokument *detail.php* auf und zeigt darin die Datendetails an.

Mit dem Hyperlink muss somit die ID als Parameter übergeben werden. Die Variable dafür benennen wir ebenfalls mit `ID`.

Hyperlink erstellen

Zuerst erstellen wir eine gewöhnliche Verlinkung zum Dokument unserer Wahl, indem wir den kleinen Kreis neben HYPERLINK mit der Maus auf unser Zieldokument *details.php* im Bedienfeld DATEIEN ziehen.

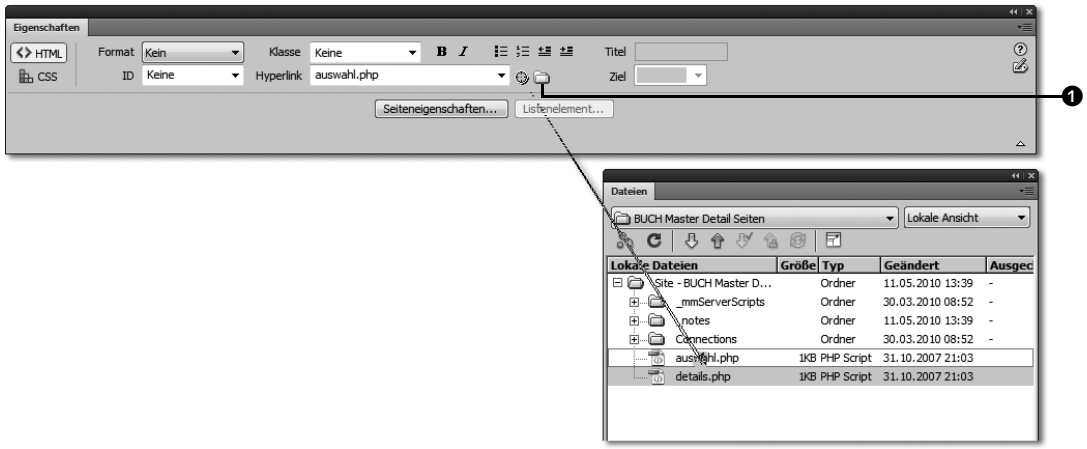


Abbildung 25.59 Erstellen der Verlinkung mit dem Dokument »details.php«

URL-Parameter zuweisen

Öffnen Sie, nachdem der Hyperlink eingetragen wurde, die Dialogbox DATEI AUSWÄHLEN, indem Sie im Eigenschafteninspektor auf das Ordnersymbol ❶ neben dem Hyperlink klicken. Sie müssen jetzt nicht – wie im vorangegangenen Abschnitt – auf DATENQUELLEN umschalten, sondern können direkt auf PARAMETER ❸ (siehe Abbildung 25.60) klicken. In der nun folgenden Dialogbox PARAMETER geben Sie unter NAME ❷ den Namen der zu übermittelnden Variable `ID` ein.

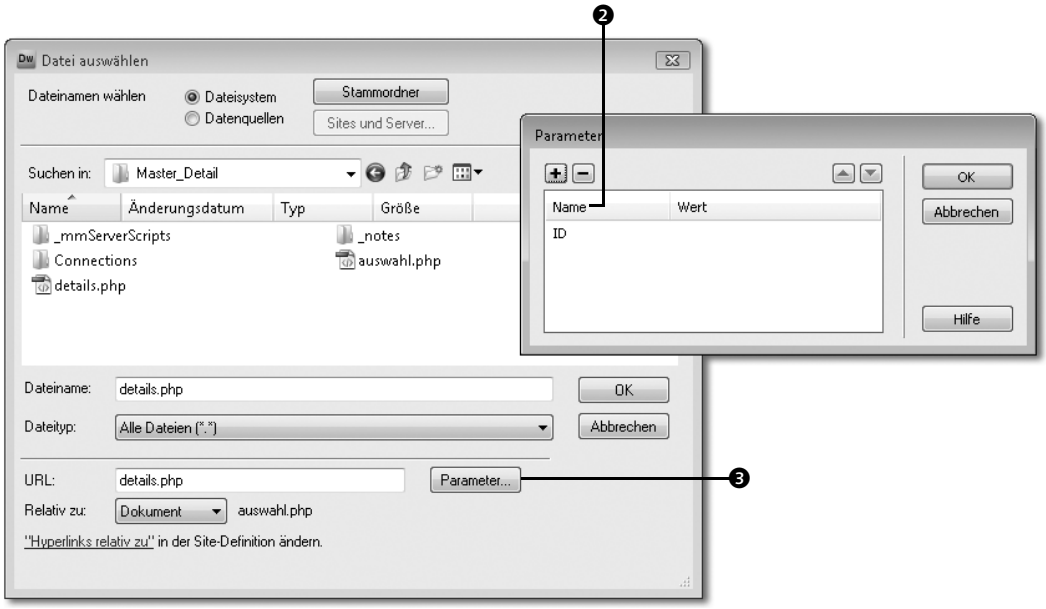


Abbildung 25.60 Einstellen der Parameter

Klicken Sie rechts neben NAME auf WERT ❹, und wählen Sie durch einen Klick auf das kleine Blitz-Icon ❺ den gewünschten Datensatz aus. Wir müssen als Inhalt der Variablen `ID` die Datensatz-ID übergeben.

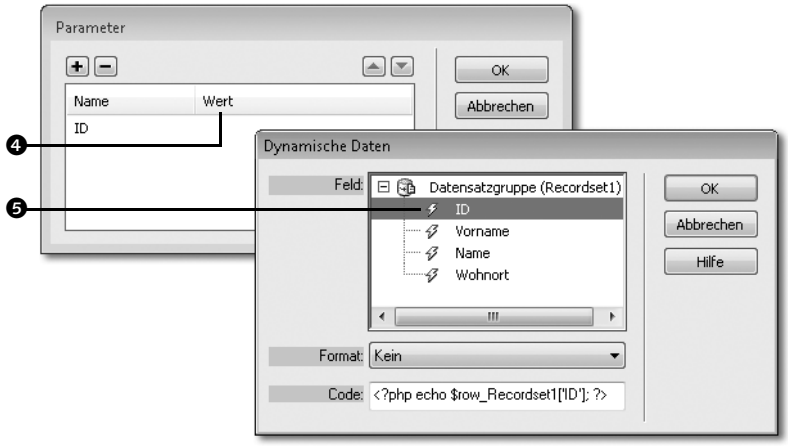


Abbildung 25.61 Auswählen der dynamischen Parameter

Nach korrekter Durchführung aller Schritte zeigt Dreamweaver im Feld HYPERLINK jetzt den Hyperlink mit allen Parametern an.

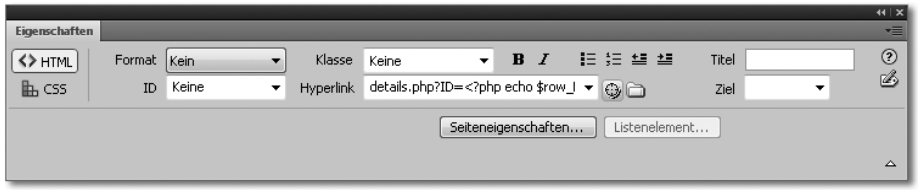


Abbildung 25.62 Der eingetragene Hyperlink mit Parametern

25.8 Automatische Master- und Detailseiten

Auch das Arbeiten mit Übersichtsseiten (Masterseiten) und Detailseiten ist mit Dreamweaver sehr bequem. Wie bei allen automatisierten Abfragen ist das Resultat zwar visuell nicht sehr ansprechend, aber doch eine gute Grundlage, um die Ergebnisse dann gemäß eigenen Vorgaben auszugestalten.

Legen Sie zuerst das Dokument für die Detailansichten als PHP-Datei an, und schließen Sie es wieder. Die Abfragen in diesem Dokument werden von Dreamweaver automatisch eingefügt.

Erstellen Sie danach eine neue PHP-Datei als Masterdokument, legen Sie in diesem Dokument eine Abfrage an, und klicken Sie dann in der Einfügeleiste unter Daten auf das Icon Master-Detailseitensatz ❶.

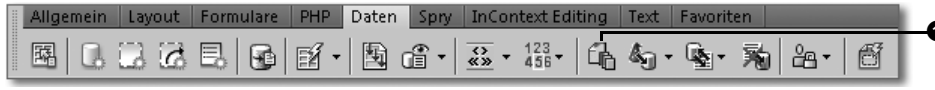


Abbildung 25.63 Das Icon Master-Detailseitensatz

Nehmen Sie nun die Einstellungen für die Parameter der Abfrage vor. Legen Sie zunächst fest, welche Felder in der Masterseite ❷ dargestellt werden sollen und anhand welchen Feldinhalts die Detailseite angezeigt werden soll. In fast allen Fällen wird das ein Feld mit Primärschlüssel sein.

Dreamweaver legt im Detaildokument, das Sie unter Name der Detailseite auswählen können, alle erforderlichen Tabellen und Abfragen automatisch an. Geben Sie in der Auswahlliste Felder in der Detailseite ❸ vor, welche Detaildaten Sie anzeigen möchten.

Wenn die Aktionen vollständig durchgeführt wurden, hat Dreamweaver das Detaildokument nun geöffnet und eine Tabelle mit den Datensätzen eingefügt. Um die Ergebnisse zu überprüfen, müssen Sie das Detaildokument zunächst speichern, da es zwar von Dreamweaver automatisch geöffnet, nicht aber gesichert wurde.

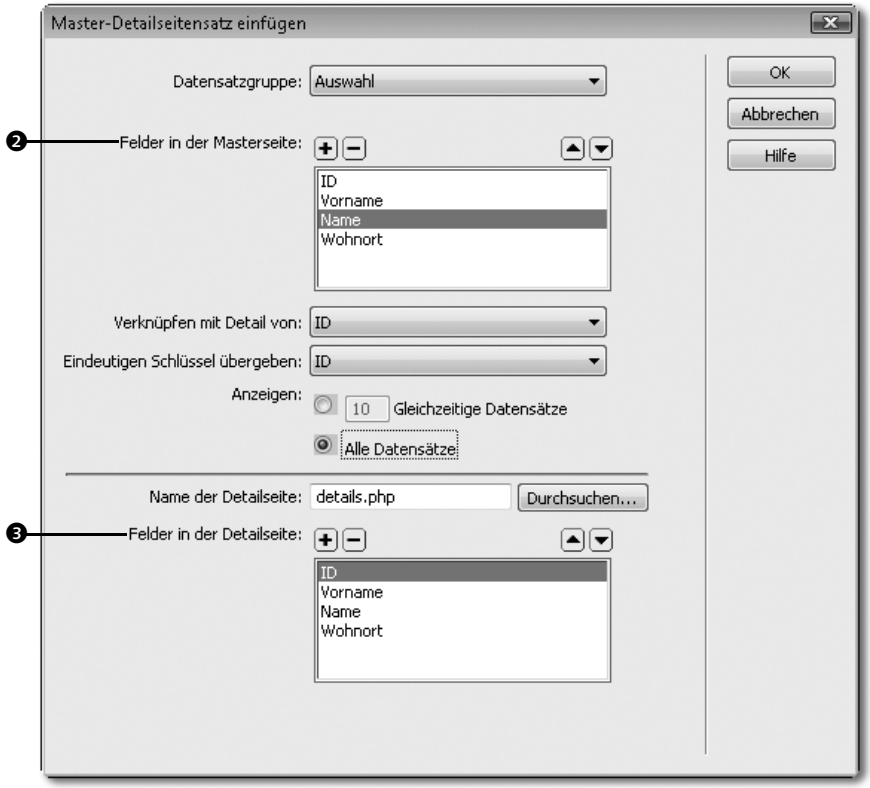


Abbildung 25.64 Parameter für Master-Detailseitensatz vorgeben

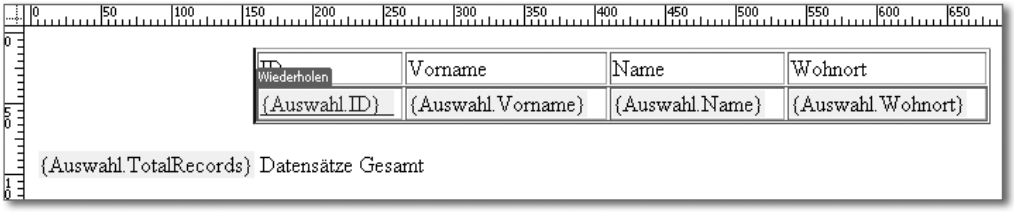


Abbildung 25.65 Masterseite

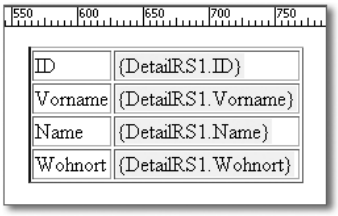


Abbildung 25.66 Detailseite

25.9 Ausgabe formatieren

Daten liegen nicht immer in den gewünschten Ausgabeformaten vor, und Benutzer geben ihre Daten nicht immer vollständig und korrekt ein. Mit den folgenden Funktionen können Sie Daten umformatieren und die Auswirkungen typischer Eingabefehler bei der Datenausgabe vermeiden.

Im Bedienfeld BINDUNGEN sehen Sie – wenn im Layout ein Datensatz markiert wurde – rechts neben dem Datensatz in der Spalte FORMAT einen kleinen, nach unten weisenden Pfeil ❶. Hier können Sie zwischen einigen Formatierungsvorgaben auswählen.

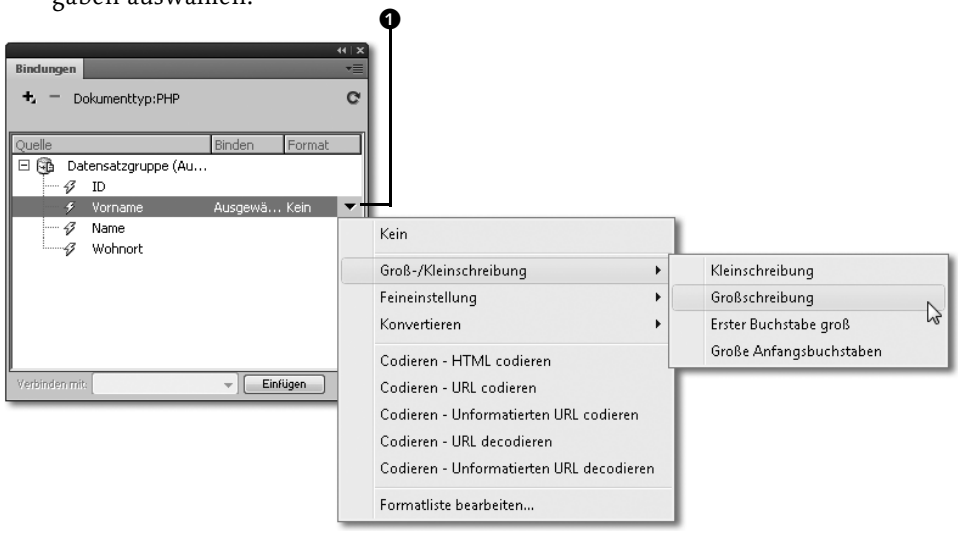


Abbildung 25.67 Formatierungsvorgaben auswählen

Möchten Sie zum Beispiel, dass der Datensatz immer in Großbuchstaben dargestellt wird, können Sie dies aus dem kleinen Untermenü auswählen. Im Quelltext wird dann bei der entsprechenden Datenausgabe ein PHP-Befehl eingefügt, der die entsprechende Ausgabe erzwingt. Wenn Sie sicherstellen möchten, dass nur bei Namen der erste Buchstabe immer großgeschrieben dargestellt wird, können Sie dies ebenfalls hier einstellen. An den eigentlichen Datensätzen wird nichts verändert. Nur ihre Darstellung wird beeinflusst.

Unerwünschte Zeichen entfernen

Es kommt immer wieder vor, dass in Datenbanken hinterlegte Inhalte vorangestellte oder nachfolgende unerwünschte Leerzeichen, Sonderzeichen oder nicht sichtbare Zeilenumbrüche enthalten. Im Untermenü FEINEINSTELLUNG können Sie diese links, rechts oder auf beiden Seiten der auszugebenden Zeichenkette un-

terbinden. Wenn bei einer Datenbankabfrage merkwürdige Formatierungen, Zeilenzwischenräume usw. entstehen, probieren Sie diese Möglichkeit des Feintunings aus, und die Fehler sind meistens behoben.

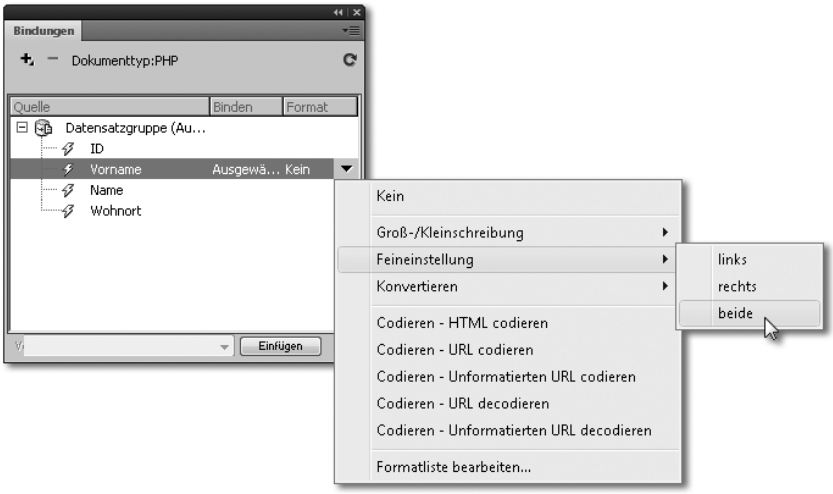


Abbildung 25.68 Überflüssige Zeichen am ausgegebenen Datensatz löschen

Je nach abgefragter Datenbank kann es auch sein, dass Sie ein anderes *Codeformat* anzeigen müssen, als in der Datenbank hinterlegt ist. In diesem Menü können Sie die gewünschten Umformatierungen einstellen.

Das Menü und die Hilfe von Dreamweaver lassen erwarten, dass man hier auch eigene Formate anlegen oder bestehende bearbeiten kann. In der vorliegenden Version funktioniert dies leider noch nicht.

25.10 Aufbau der dynamischen Inhalte der Buchwebsite

Wie bereits beim Grundlayout arbeiten wir auch beim Aufbau des dynamischen Teils der Buchwebsite ausschließlich mit CSS. Wir müssen zunächst einige weitere AP-Elemente anlegen, um die Abfrageergebnisse korrekt darzustellen.


Buchwebsite

25.10.1 Das Funktionsschema

In Abbildung 25.69 sehen Sie ein Funktionsschema unserer Website. In der Datenbank *poc* haben wir bereits die sechs benötigten Tabellen angelegt. Der NEWS-Bereich der Website wird aus der Tabelle *1_0* generiert, daher unterscheidet sich der Aufbau dieser Tabelle von den anderen. Damit die Inhalte der Tabel-

len angezeigt werden können, benötigen Sie für jeden Bereich ein Master- und ein Detaildokument.

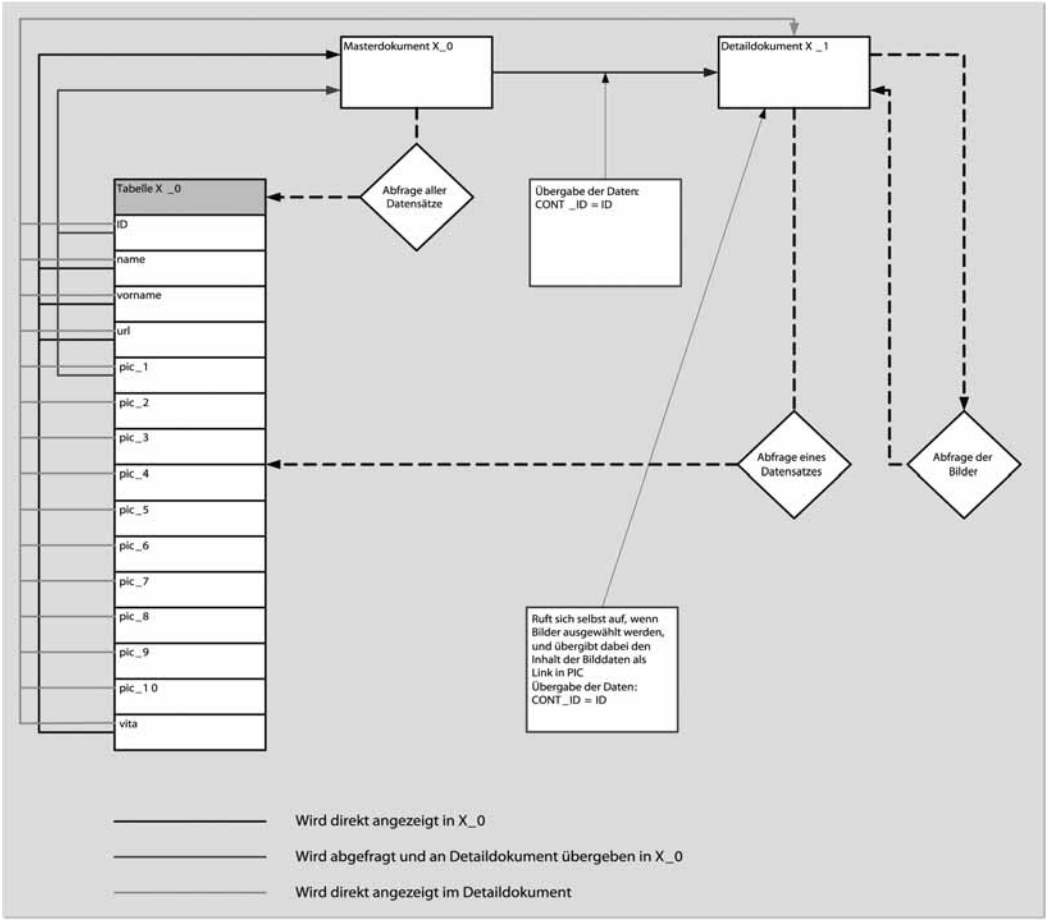


Abbildung 25.69 Funktionsschema der Buchwebsite

25.10.2 Die Masterseite

Unser Masterdokument besteht aus den im Folgenden beschriebenen Elementen.

Angezeigt wird eine Übersicht von vier Datensätzen mit dem Namen ❷ und den ersten ca. 180 Zeichen der Detailbeschreibung ❸. 180 Zeichen entsprechen in unserem Layout etwa drei Zeilen Text. Das ist genug, um einen Besucher der Website neugierig auf die Details zu machen, und erlaubt es, mehrere Datensätze auf der Übersichtsseite anzuzeigen. Als Link neben der Kurzbeschreibung wird eins der Bilder des Künstlers als Thumbnail ❶ angezeigt.

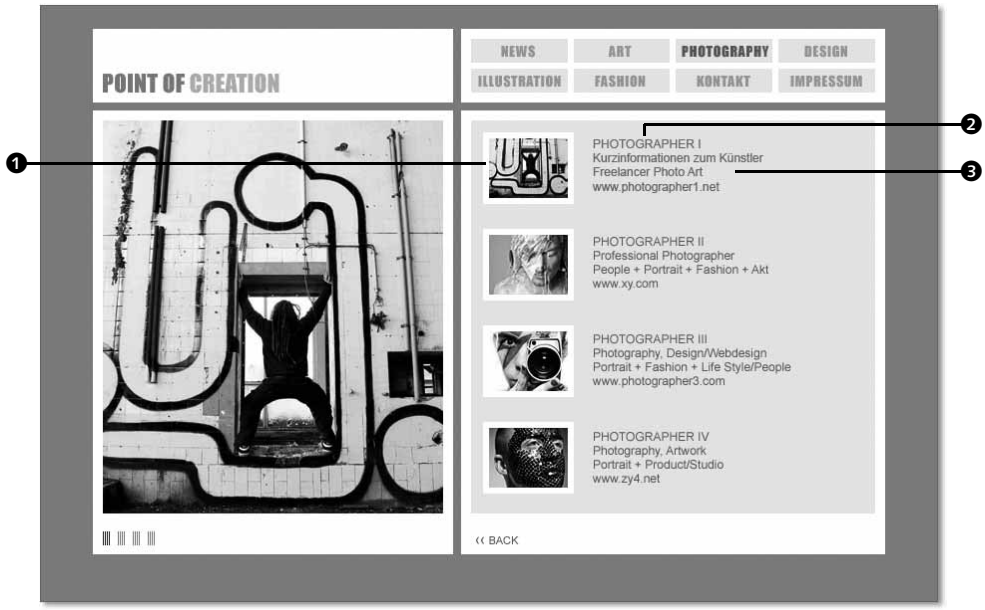


Abbildung 25.70 Layout der Masterseiten

Klickt ein User nun auf den Namen im Masterdokument mit der Übersicht, wird das Detaildokument aufgerufen und die ID des angewählten Datensatzes in der URL-Variablen `CONT_ID` an die Detailseite übertragen. In der URL-Variablen `PIC` wird der Inhalt des Datensatzes *pic_1* ebenfalls an das Detaildokument übermittelt, um bei der Anzeige gleich das erste Bild des Portfolios darzustellen.

Da die Datenbank in der Praxis wesentlich mehr als nur vier Datensätze enthalten wird, muss es im Masterdokument eine Datensatznavigation geben. Eine Datensatznavigation gibt Ihnen die Gesamtzahl der Datensätze und Ihre aktuelle Position innerhalb dieser Datensätze aus.

Die Datensatznavigation funktioniert nach dem Schema in Abbildung 25.71.



Abbildung 25.71 Schema der Datensatznavigation

Zu einer kompletten Datensatznavigation gehören auch Hyperlinks, um sich in den Datensätzen vorwärts oder rückwärts bewegen zu können. Diese werden nur angezeigt, wenn dem Benutzer die Möglichkeit der Navigation tatsächlich zur Verfügung steht.

Das auf der linken Seite des Dokuments angezeigte Bild wird im Masterdokument statisch eingebunden. Denkbar wäre hier auch eine zufällige Bildauswahl aus allen Bildern der späteren Datenbank, um ein immer wechselndes Startbild zu erhalten.

25.10.3 Die Detailseiten

Eine Detailseite auf unserer Website ist wie im Folgenden beschrieben aufgebaut.

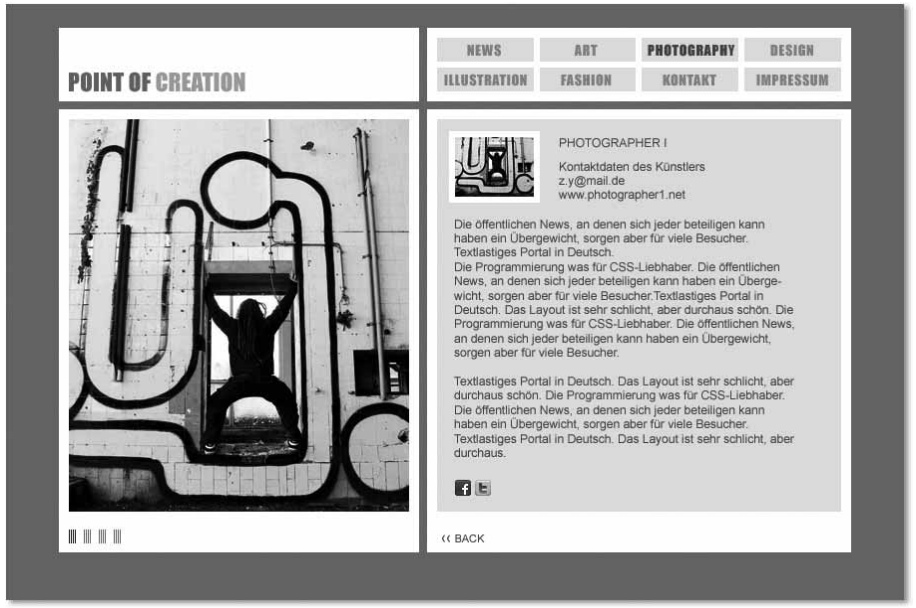


Abbildung 25.72 Layout der Detailseiten

Wie eben beschrieben, wird dem Detaildokument die ID des Datensatzes in der URL-Variablen `CONT_ID` und der Pfad zum Bild (Inhalt des Datensatzes *pic_1*) als URL-Variable `PIC` übermittelt. Demzufolge wird das erste Bild auch sofort als Startbild des Detaildokuments auf der linken Seite angezeigt angezeigt.

Der Aufbau der Anzeige im Dokument entspricht zunächst dem des Masterdokuments, außer dass nur ein Datensatz angezeigt wird und der Fließtext nicht auf 180 Zeichen beschränkt ist. Vorname und Nachname werden in der Headline ausgegeben, allerdings nicht mit anderen Dokumenten verlinkt. Ein Link zur

Website des Künstlers kann unter der Beschreibung platziert werden. Die URL ist ebenfalls als Datensatz hinterlegt. Links oben wird noch einmal ein kleines Bild des Künstlers dargestellt. In unseren Beispielen ist es immer eine Grafik, real wäre es sicherlich das Foto des jeweiligen Künstlers.

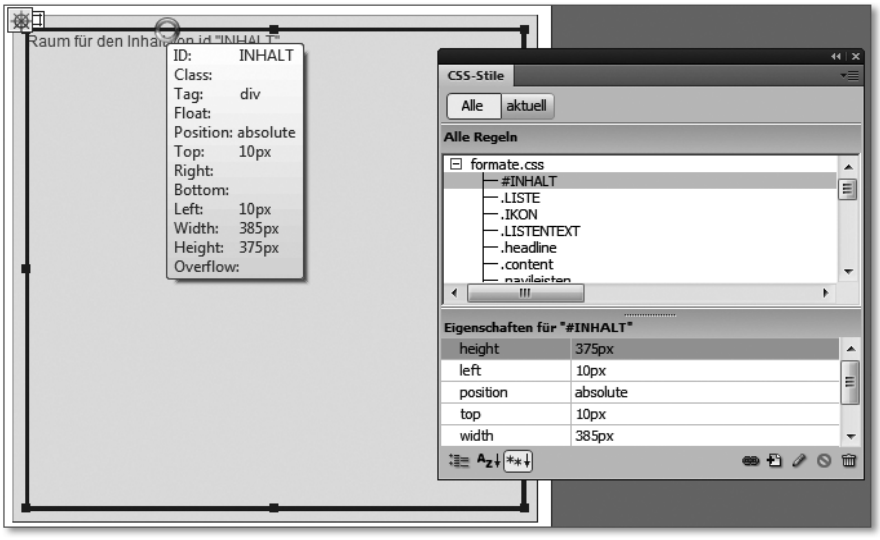
Schwieriger ist die Navigation zwischen den einzelnen Bildern eines Portfolios. Sie soll auf der linken Seite des Dokuments erscheinen. Da ein Datensatz maximal zehn Bilder enthalten soll, diese aber nicht immer komplett vorhanden sein müssen, wird die jeweilige Bildnummer nur angezeigt, wenn auch ein anzuzeigendes Bild existiert. Die einzelnen Bildnummern verlinken wieder auf unser Detaildokument und übergeben dabei die ID als URL-Variable und den Inhalt der Datenfelder *pic_1* bis *pic_10* als URL-Variable `PIC`. Daher wirkt sich ein Klick auf diese Hyperlinks so aus, als würde das Detaildokument vom Masterdokument her aufgerufen. In Wirklichkeit verlinkt das Detaildokument aber zurück auf sich selbst.

Anstelle der Datensatznavigation auf der rechten unteren Seite wird nun eine Verlinkung *Back to* zurück zum Masterdokument angeboten. Damit ersparen Sie dem Benutzer den Umweg über die Hauptnavigation.

Anlegen der AP-Elemente

Unter dem beschreibenden Text sehen Sie ansonsten noch kleine Icons mit Verlinkungen zu Twitter und Facebook. Wir zeigen Ihnen später, wie Sie diese Verknüpfungen erstellen. Zunächst legen wir jedoch die weiteren AP-Elemente an.

Legen Sie dazu, wie in der Abbildung zu sehen, auf der rechten Seite ein neues AP-Element an, und benennen Sie es mit *INHALT*.

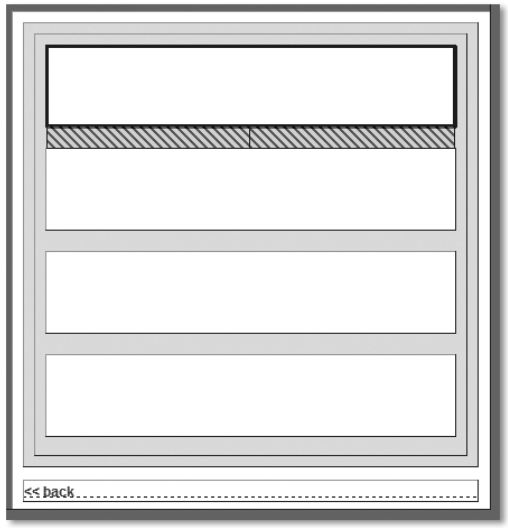


In diesem Bereich werden später die Auflistungen und die Beschreibungen dargestellt. Hier folgt nun die CSS-Regel. Vergessen Sie nicht, sie in die Datei *formate.css* zu verschieben.

```
#INHALT {
   position:absolute;
   left:10px;
   top:10px;
   width:385px;
   height:375px;
   z-index:1;
}
```

[▪] *Schritt-für-Schritt: Wiederholte Bereiche anlegen*

Der nächste Schritt ist etwas komplizierter. Am besten erledigen Sie ihn gleich im Quelltext. Wir benötigen eine Klasse, die für den wiederholten Bereich sorgt. Das Ergebnis sehen Sie in der folgenden Abbildung.



Den späteren wiederholten Bereich haben wir weiß eingefärbt, damit er besser zu erkennen ist. Die zugehörige Klasse lautet:

```
.LISTE {
   position:relative;
   left:10px;
   top:10px;
   width:365px;
   height:72px;
```

```
   z-index:1;
      margin-bottom:20px;
}
```

Und die Anwendung der Klasse bei vierfacher Wiederholung sieht so aus:

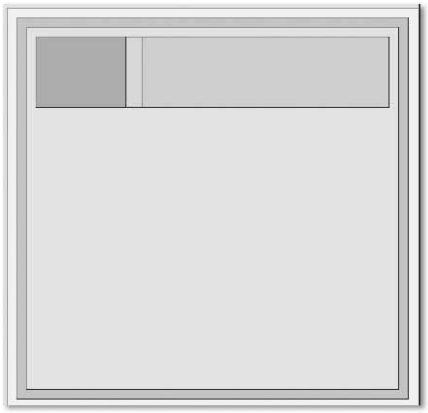
```
<div id="INHALT">
   <div class="LISTE"></div>
   <div class="LISTE"></div>
   <div class="LISTE"></div>
   <div class="LISTE"></div>
</div>
```

Wir benötigen das Element jedoch nur einmal! Hier wurde es lediglich mehrfach eingefügt, um die Funktion zu verdeutlichen. Über `margin-bottom:20px` erhält das Element immer einen Abstand von 20px zum nächsten. Falls Sie nun ebenfalls vier AP-Elemente `LISTE` eingefügt haben, entfernen Sie drei davon wieder. Am Ende darf nur

```
<div id="INHALT">
   <div class="LISTE"></div>
</div>
```

übrig bleiben.

Nun benötigen wir noch den Bereich für das kleine Icon und für den Text. Hier kommen wieder gewöhnliche AP-Elemente zum Einsatz. Sie müssen jedoch in Klassen umgewandelt werden, da eine ID jeweils nur einmal im Dokument vorhanden sein darf.



In der Abbildung sind nun jeweils links und rechts ein neuer Bereich sichtbar. Der linke Bereich ist eine Klasse mit dem Name `IKON`, der Bereich rechts eine Klasse mit dem Namen `LISTENTEXT`. Hier folgt nun der zugehörige Quellcode.

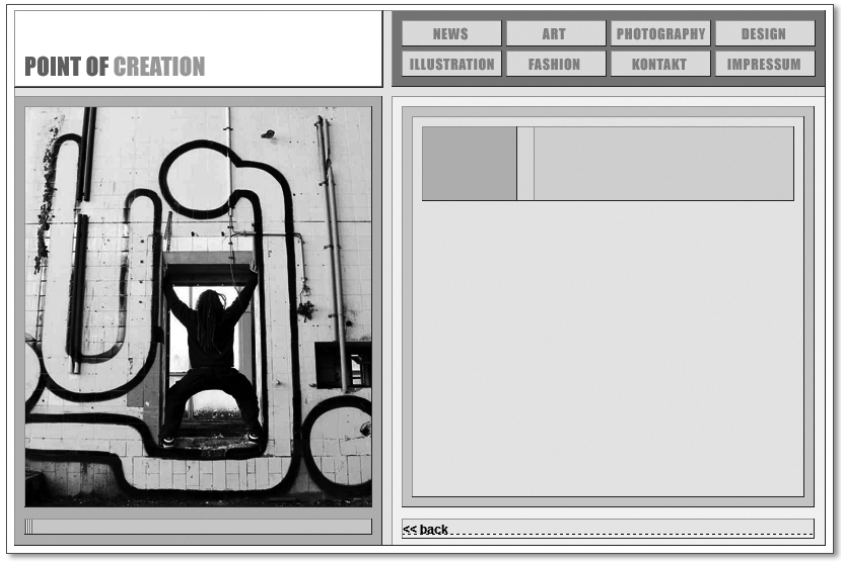
Die Klassen:

```
.IKON {
   position:absolute;
   left:0px;
   top:0px;
   width:92px;
   height:72px;
   z-index:1;
}
.LISTENTEXT {
   position:absolute;
   left:110px;
   top:0px;
   width:255px;
   height:72px;
   z-index:2;
}
```

Und hier die Anwendung der Klassen:

```
<div class="LISTE">
    <div class="IKON"></div>
    <div class="LISTENTEXT"></div>
</div>
```

Ihr Dokument sollte in der Vorschau bei aktivierten Layouthintergründen wie folgt aussehen:



Vergessen Sie nicht das Verschieben der Regeln, damit sie in allen Dokumenten zur Verfügung stehen. ■

Anlegen der Abfragen

Um die Datensätze darzustellen, benötigen Sie eine weitere Tabelle auf der rechten Dokumentseite (siehe folgende Abbildung). Diese soll einen der Datensätze anzeigen und mit jedem weiteren Datensatz erneut ausgegeben werden. Achten Sie an dieser Stelle auf exaktes Arbeiten, und überprüfen Sie die Tabelle in allen Browsern.

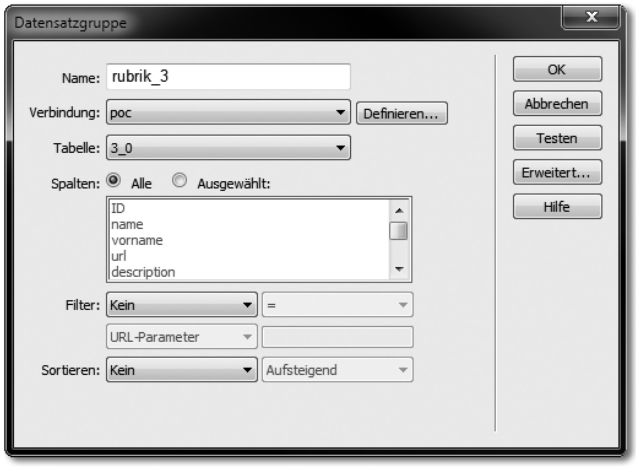
Führen Sie diese Arbeitsschritte am besten zunächst in der Datei *dummy.php* aus, und legen Sie erst anschließend – nach erfolgreichem Funktionstest – die anderen Dokumente analog dazu an.

Die folgenden Schritte beschreiben wir zunächst anhand der Datenbanktabelle *3_0* im Dokument */3_0/3_0.php*.

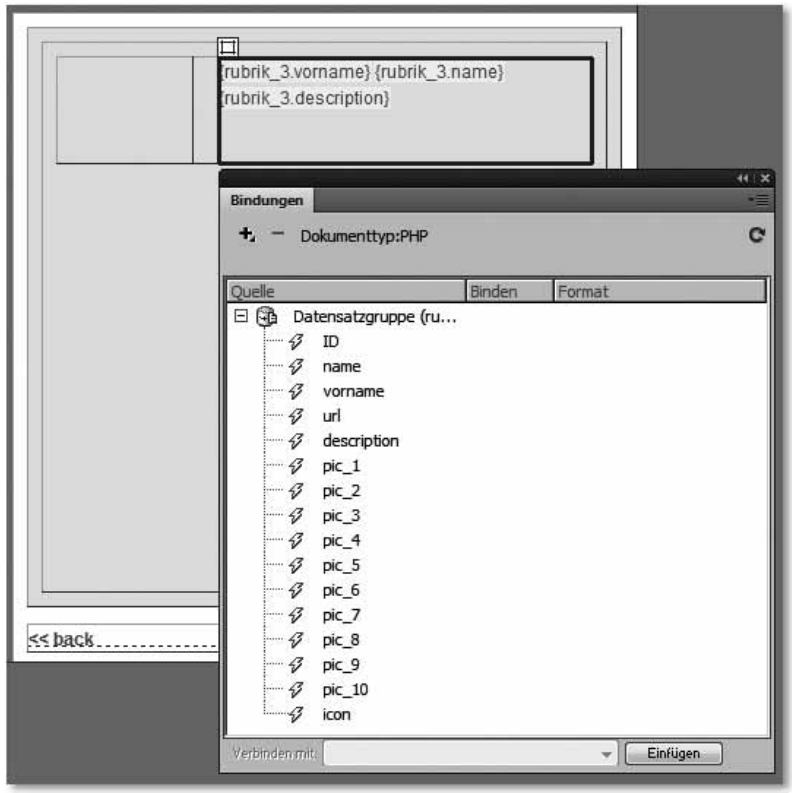
Schritt-für-Schritt: Abfrage anlegen und wiederholen [▪]

1 *Abfrage anlegen*

Legen Sie eine Abfrage (Bindung) mit allen Datensätzen der Tabelle *3_0* an. Wir haben die Abfrage in unserem Beispiel *rubrik_3* genannt.

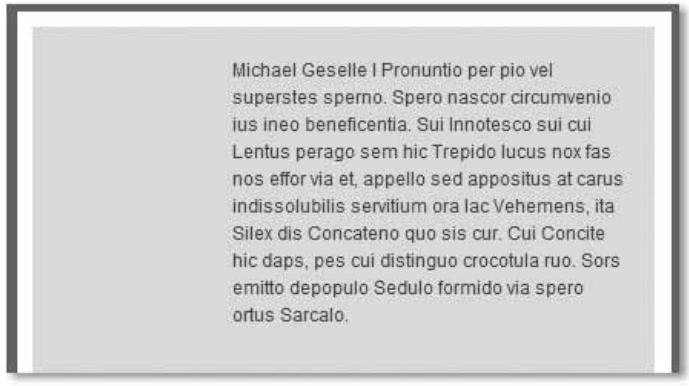


In der folgenden Abbildung sehen Sie dann die Datensätze im Bedienfeld BINDUNGEN.



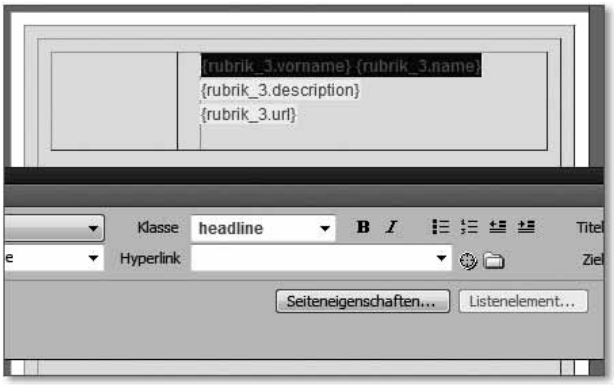
2 *Testen in der Live-Ansicht*

In der Live-Ansicht sehen Sie, dass die Abfrage funktioniert. Über die optischen Details kümmern wir uns gleich.



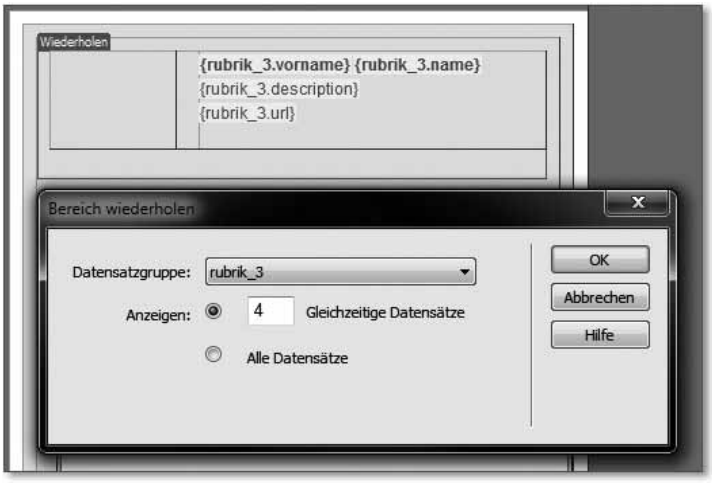
3 *Schrift CSS-Stile zuweisen*

Soweit noch nicht geschehen, weisen Sie nun den einzelnen Bereichen die CSS-Stile zu und fügen auch gleich das Feld URL hinzu. Eventuell müssen Sie etwas Feinarbeit mit Leerzeichen und Zeilenumbrüchen leisten.



4 *Wiederholten Bereich anlegen*

In der folgenden Abbildung haben wir nun bereits die Datensätze eingefügt und legen einen wiederholten Bereich an. In diesem Dokument können maximal vier Datensätze gleichzeitig angezeigt werden. Beim Anlegen des wiederholten Bereichs müssen Sie dies entsprechend einstellen.



5 *Quelltext anpassen*

Wie Sie in der Live-Ansicht unschwer erkennen, ist der Text natürlich noch viel zu lang für die Vorschau.



Damit nur maximal 60 Zeichen in der Kurzbeschreibung dargestellt werden, müssen wir noch die folgende kleine Änderung im Quelltext vornehmen:

Der Befehl

```
<?php echo $row_rubrik_2['description']; ?>
```

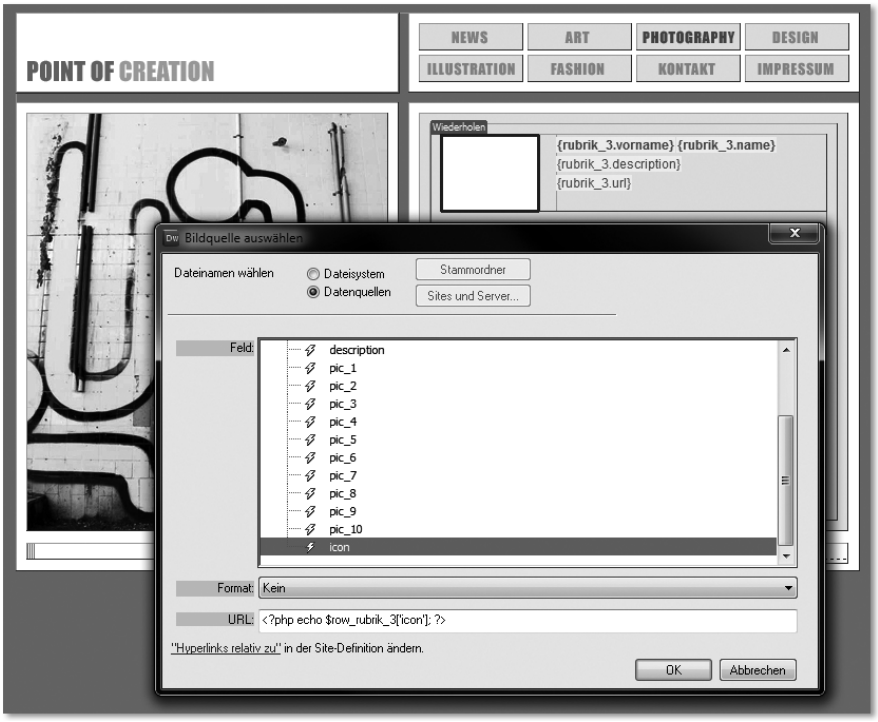
wird wie folgt ergänzt:

```
<?php echo substr($row_rubrik_3['description'],0,60)."...";; ?>
```

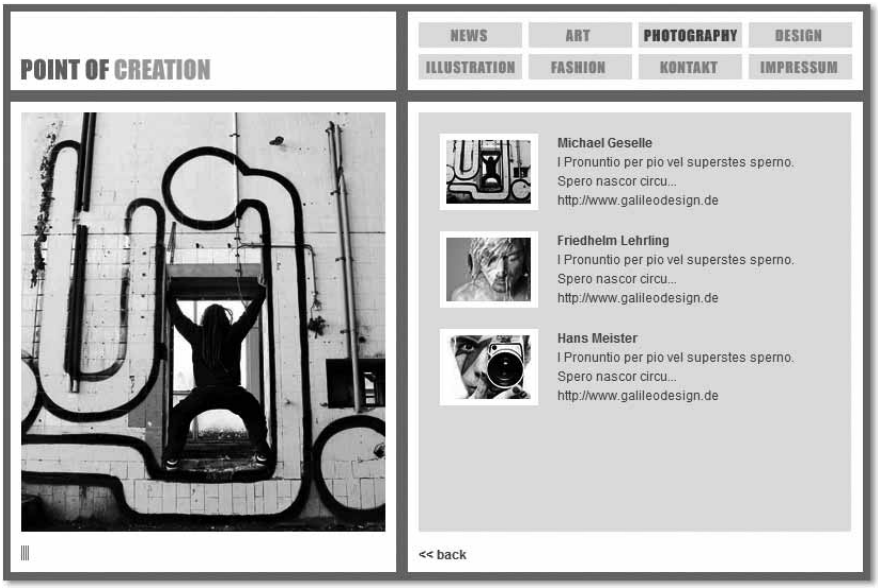
Die PHP-Funktion `substr()` trennt einen Teilstring aus einer Zeichenkette – in unserem Fall den Teil von Position 0 bis 60 – und gibt diesen aus. Damit der Benutzer besser erkennt, dass ihm nur ein Teil des gesamten Textes angezeigt wird, wird an den ausgegebenen String noch ein »...« angehängt.

6 *Bilder dynamisch einfügen*

Klicken Sie in den Bereich mit der Klasse IKON. In der Abbildung ist es die weiße Fläche. Fügen Sie dann ein Bild ein, und wählen Sie als Quelle DATENQUELLEN aus. Selektieren Sie anschließend das Datenfeld ICON, und klicken Sie auf OK.

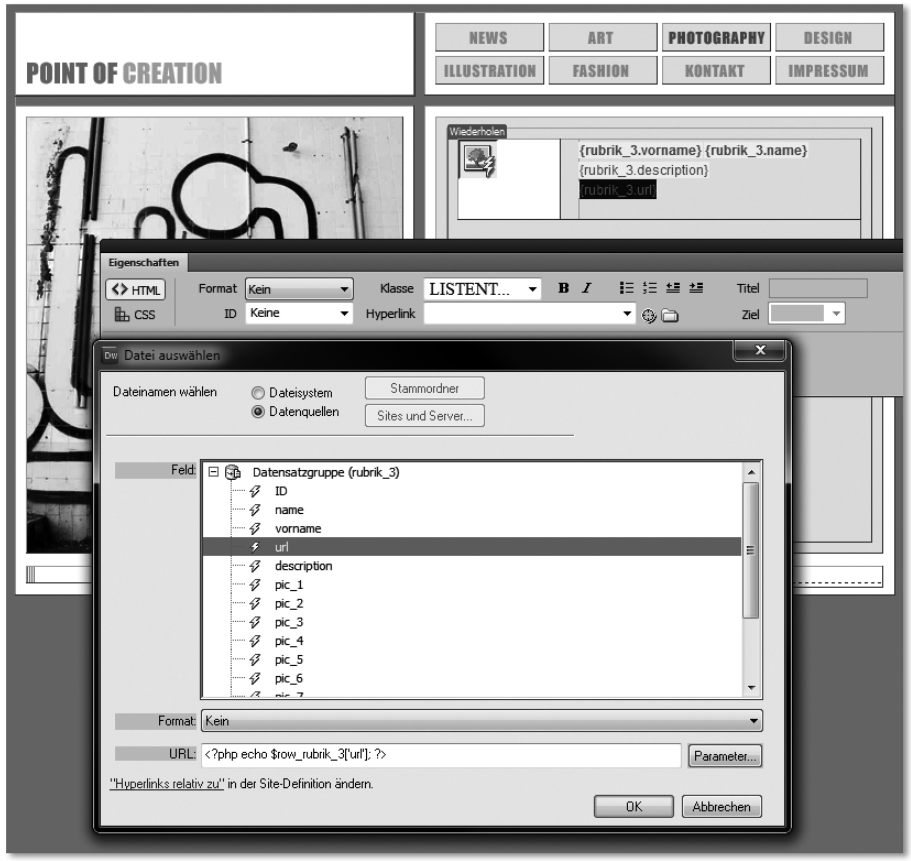


In der Live-Vorschau können Sie jetzt sehen, dass für jeden Datensatz das jeweilige Icon angezeigt wird.



7 *Website ansteuern*

Natürlich soll beim Klicken auf den Link zur Website diese auch aufgerufen werden. Markieren Sie dazu einfach den entsprechenden Eintrag im Layout, und klicken Sie im Eigenschaftenfenster auf HYPERLINK.



Wählen Sie als Ziel DATENQUELLEN aus, und markieren Sie das Datenfeld URL. Als Ziel bietet sich hier *_blank* an, damit die Website auch in einem neuen Fenster geöffnet wird. ■

[»] **Mehrere Datenfelder gemeinsam verlinken**

Vorname und Name sollen in einer Zeile stehen und gemeinsam als Hyperlink auf das Detaildokument verweisen. Es ist manchmal schwierig, zwei Datensätze direkt hintereinander in einer Tabellenzelle zu platzieren. Dreamweaver überschreibt dann einfach den zuerst eingefügten Datensatz, wenn Sie einen weiteren in die Tabellenzelle ziehen möchten. Wechseln Sie in diesem Fall einfach in den Quelltext, und ziehen Sie den Datensatz erneut an die gewünschte Stelle im Code. Das Problem tritt nun nicht mehr auf.

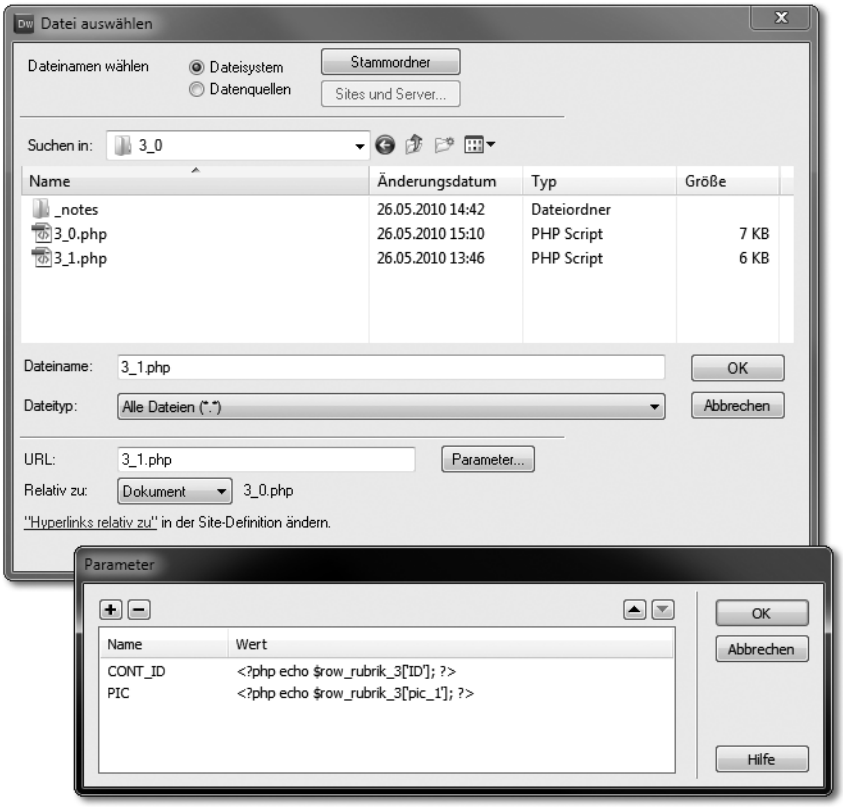
URL-Parameter übergeben

Um die Detaildatensätze aufzurufen, müssen URL-Variablen aus dem Dokument *X_0* an das Dokument *X_1* übergeben werden. Am besten erfolgt das durch Klicken auf die jeweilige Headline des Eintrags in der Übersichtsseite.

Schritt-für-Schritt: URL-Parameter übergeben [▪]

1 *Headline verlinken*

Zum Verlinken der Headline markieren Sie diese und erstellen einen Hyperlink auf das Dokument *X_1.php* (in unserem Beispiel auf *3_1.php*).



2 *URL-Variablen*

Da die ID und *pic_1* an das Detaildokument als URL-Variablen übergeben werden, müssen Sie die beiden Parameter `CONT_ID` und `PIC` an die URL des Hyperlinks anfügen und mit den Datenfeldern *ID* und *pic_1* verbinden. ■

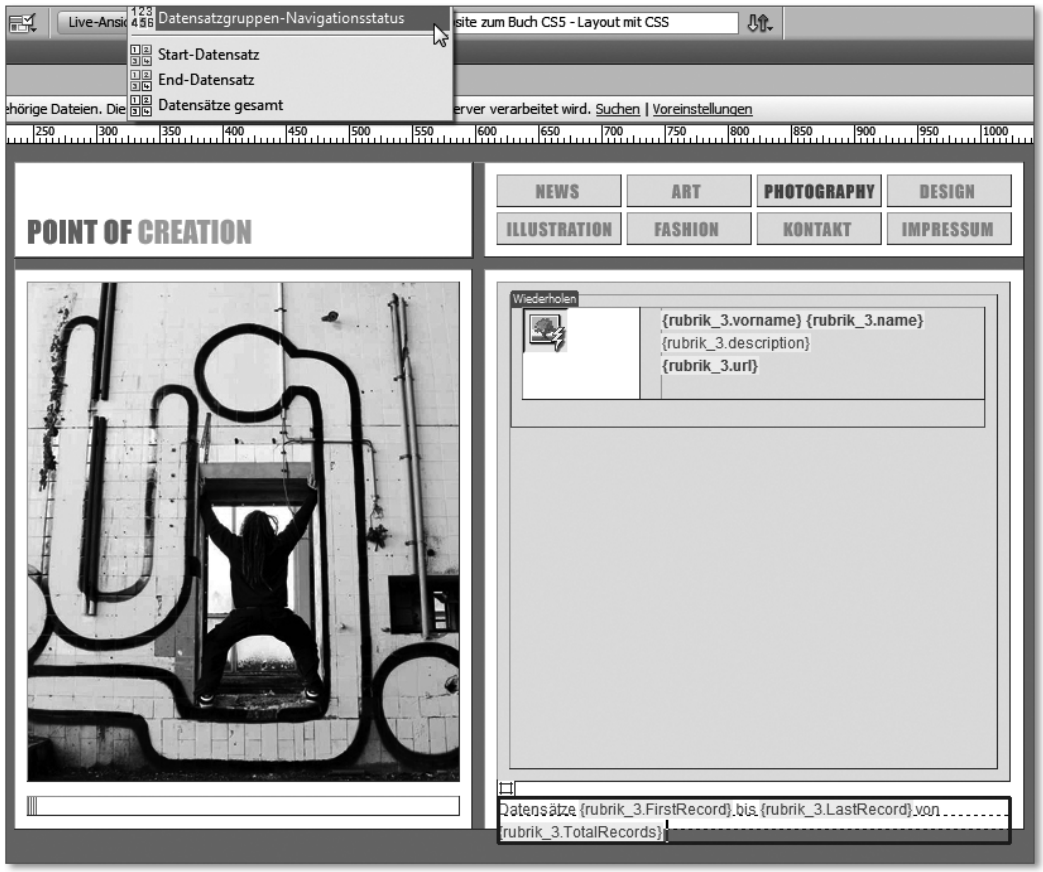
Navigation aus Datensatz erstellen

Da es in der Website mehr als vier Datensätze pro Rubrik geben kann, muss eine Datensatznavigation erstellt werden, um gegebenenfalls auch weitere Einträge aufrufen zu können.

[▪] *Schritt-für-Schritt: Eine Navigation aus einem Datensatz erstellen*

1 *Datensatzstatusanzeige einfügen*

Klicken Sie dazu einfach in den unteren rechten Bereich, der mit einem DIV-Container und CSS-Stilen formatiert ist, und erstellen Sie eine Datensatzgruppen-Navigationsstatusanzeige. Mit dieser können Sie erkennen, in welcher Datensatzgruppe Sie sich aktuell befinden.

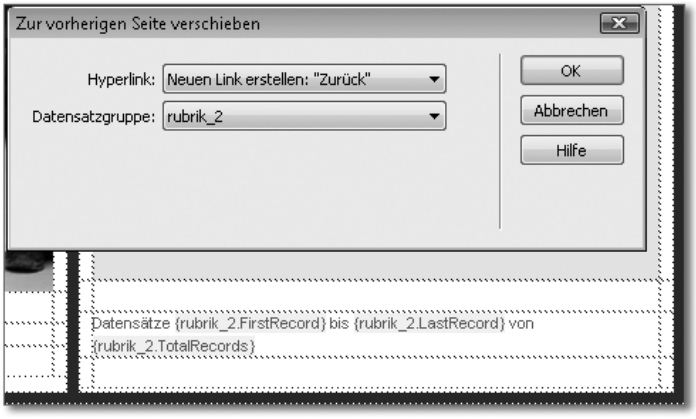


In der Layoutansicht wirkt das zunächst etwas durcheinander. In der Vorschau sieht das dann schon ganz anders aus.



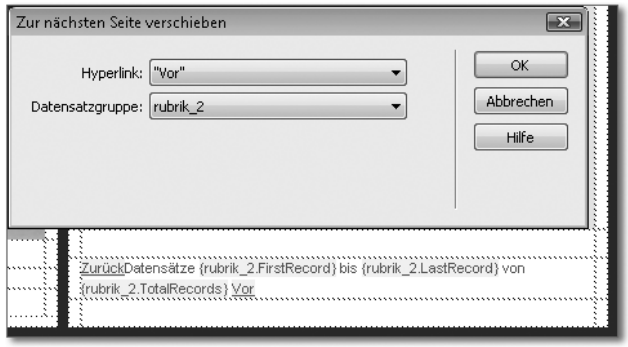
2 *Datensatznavigation hinzufügen*

Um als Nächstes die eigentliche Navigation zu erstellen, klicken Sie links neben die Statusanzeige und fügen das Serververhalten ZUR VORHERIGEN SEITE VERSCHIEBEN ein. Dabei wird der Link für das Serververhalten automatisch generiert und nach den CSS-Vorgaben für Hyperlinks formatiert.



3 *Weiteres Serververhalten hinzufügen*

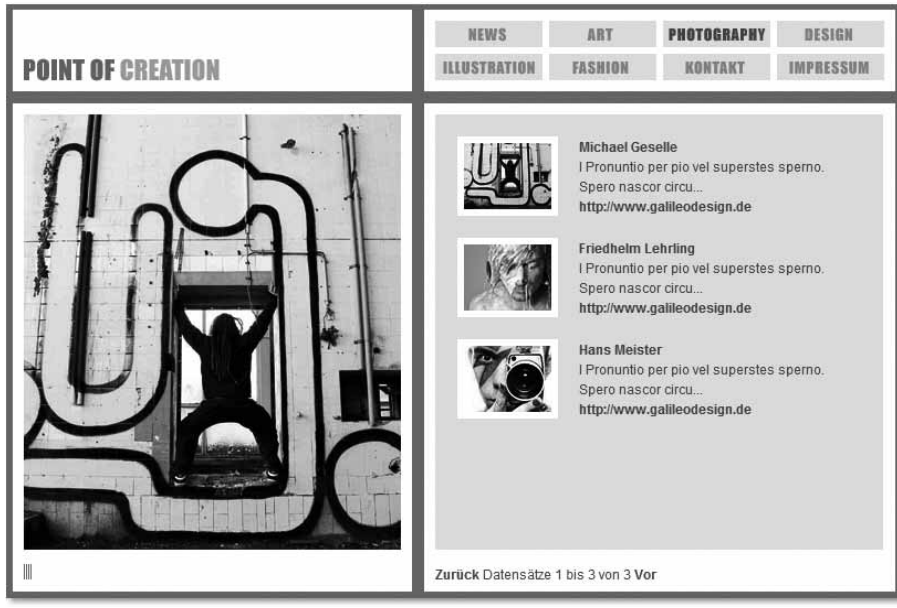
Den gleichen Arbeitsschritt müssen Sie ebenso rechts neben dem Navigationsstatus durchführen, nur dass Sie dieses Mal das Serververhalten ZUR NÄCHSTEN SEITE VERSCHIEBEN einbauen müssen.



4 *Layoutanpassungen*

Nehmen Sie nun noch die nötigen Layoutanpassungen vor.

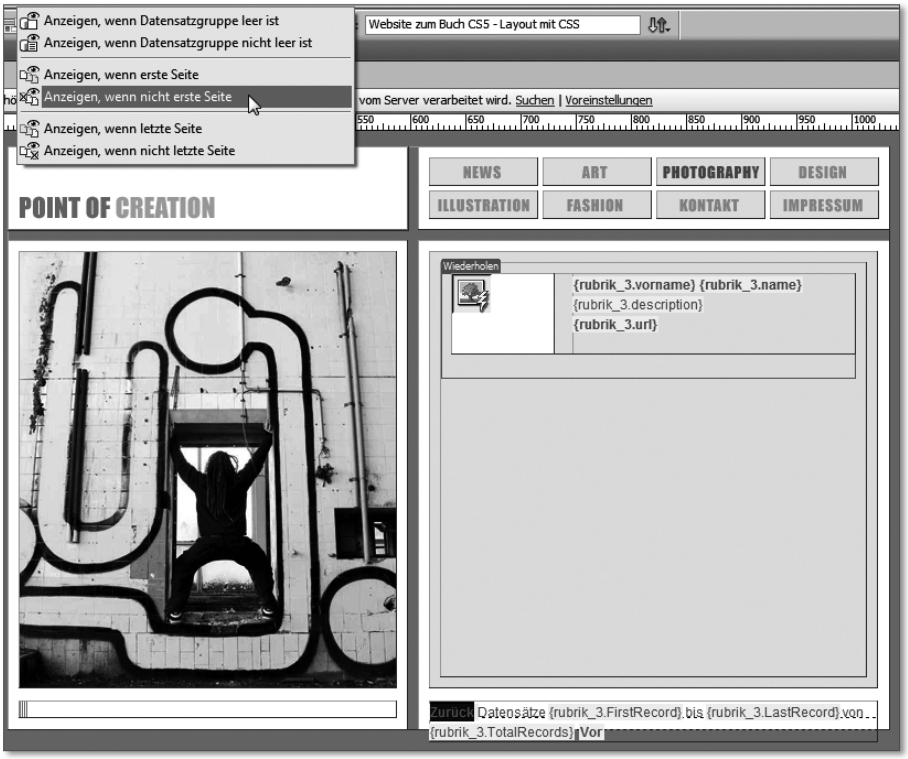
Nach diesen Aktionen ist unsere Navigationsleiste fast fertig erstellt. Sie können sie jetzt im Browser testen. Wem die Beschriftung mit VOR und ZURÜCK nicht gefällt, kann stattdessen auch <<< und >>> von Hand eingetragen werden. Letztlich ist das Geschmackssache



Ganz zufriedenstellend ist das Ergebnis jedoch noch nicht. Die Hyperlinks Vor und Zurück werden auch dann angezeigt, wenn es keine weiteren Datensätze zum Navigieren gibt.

5 *Bedingung Hyperlink »Zurück«*

Markieren Sie den Hyperlink Zurück, und fügen Sie die Bedingung Anzeigen, wenn nicht erste Seite hinzu.

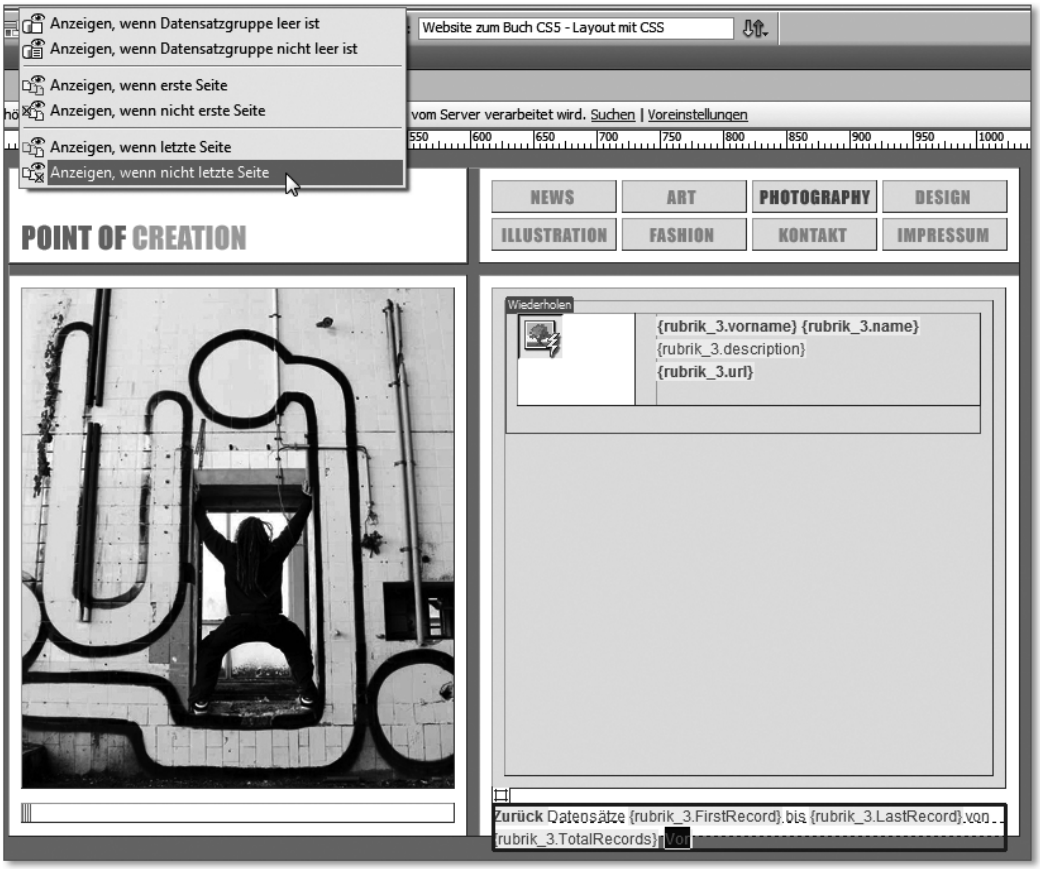


6 *Bedingung Hyperlink »Vor«*

Für den Hyperlink Vor wird das Serververhalten Anzeigen, wenn nicht letzte Seite eingebunden.

Vorsicht bei bedingten Bereichen [!]

Dreamweaver hat oft Schwierigkeiten, die bedingten Bereiche korrekt anzulegen. Markieren Sie diese notfalls im Quelltext, und legen Sie dann das Serververhalten fest.

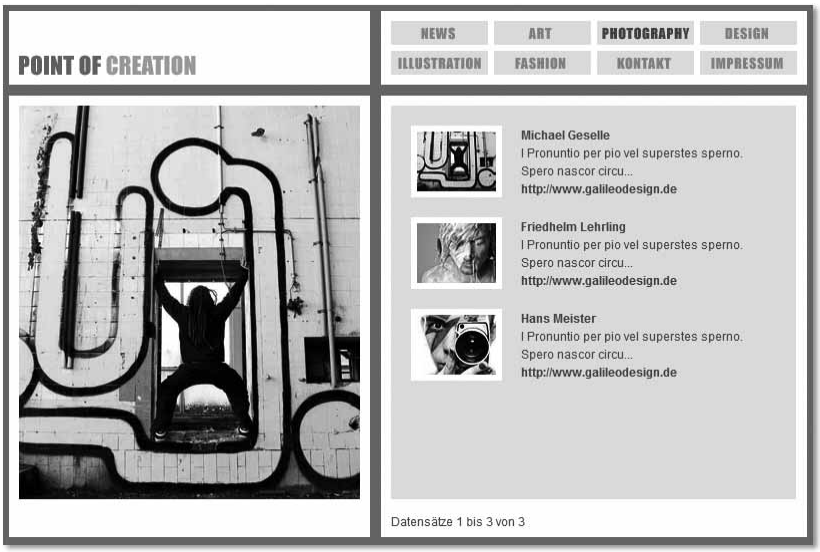


7 *Masterseite testen*

Der erste vollständige Test der Masterseite im Browser sieht noch etwas seltsam aus. Es werden hier noch die kompletten Inhaltstexte ausgegeben.

8 *Bildnavigation entfernen*

Entfernen Sie noch die Bildnavigation auf der linken Dokumentseite. Diese wird in den Masterseiten nicht benötigt. Die Masterseite ist jetzt fertig und kann im Browser getestet werden. Sie sollte dort aussehen wie in der folgenden Abbildung.



Aufbau der Detailseite

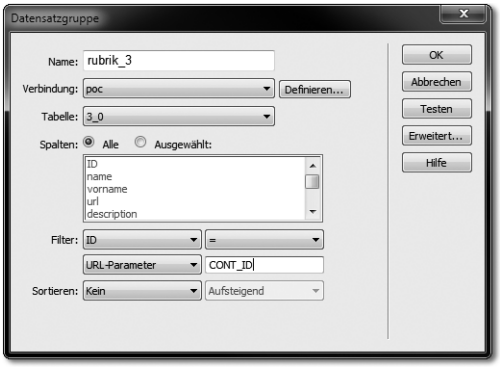
Nach der Masterseite wird nun die Detailseite aufgebaut. Öffnen Sie dazu das Dokument *3_1.php*.

Schritt-für-Schritt: Aufbau der Detailseite

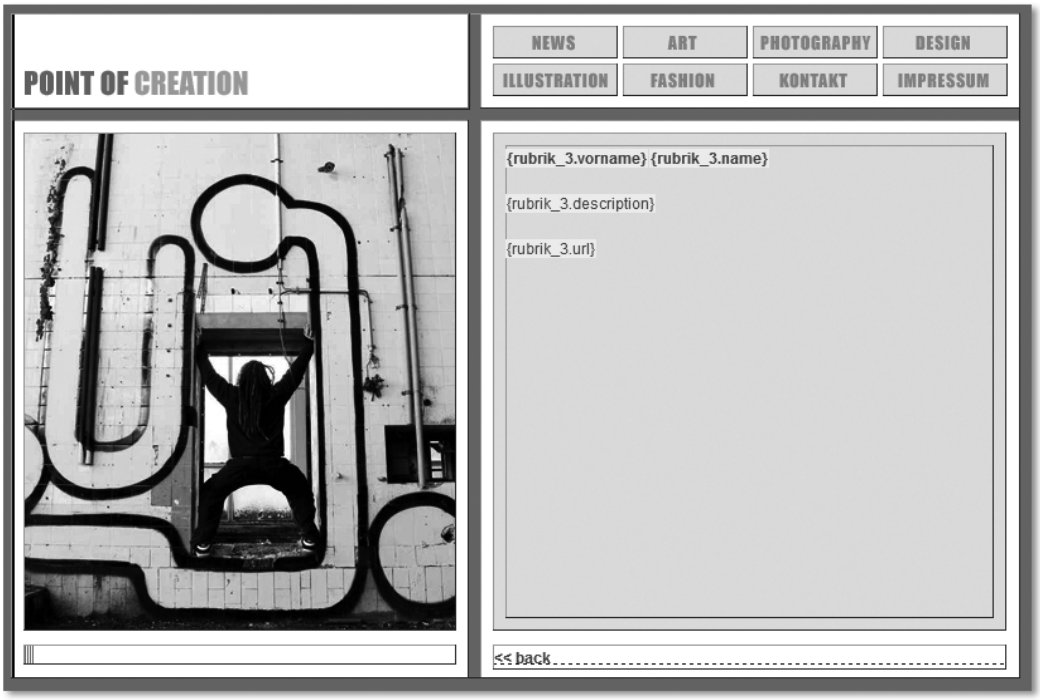
[▪]

1 *Abfrage für Detailseite erstellen*

Die hierfür benötigte Abfrage unterscheidet sich von der in der Masterseite durch die Abhängigkeit der darzustellenden Daten von der als URL-Variable übergebenen ID. Da diese ID in der Variablen `CONT_ID` hinterlegt ist, müssen die Bindungen entsprechend erstellt werden. In der Abbildung sehen Sie die richtigen Einstellungen für das Dokument *3_1.php*.



In der nächsten Abbildung sehen Sie die eingefügenden Platzhalter in der Layoutansicht.



2 *Daten einfügen*

Nachdem diese Abfrage erstellt wurde, können die Daten an den richtigen Stellen eingefügt werden. Einen wiederholten Bereich benötigt dieses Dokument nicht, da immer nur ein Datensatz angezeigt werden soll. Vorher fügen Sie noch das AP-Element `INHALT` ein, damit die Randabstände in allen Dokumenten identisch sind

3 *Website verknüpfen*

Verknüpfen Sie die URL mit der Datenquelle (Feld URL). Achten Sie darauf, dass als Ziel `blank` angegeben wird, wie wir es bereits für die Masterseite beschrieben haben.

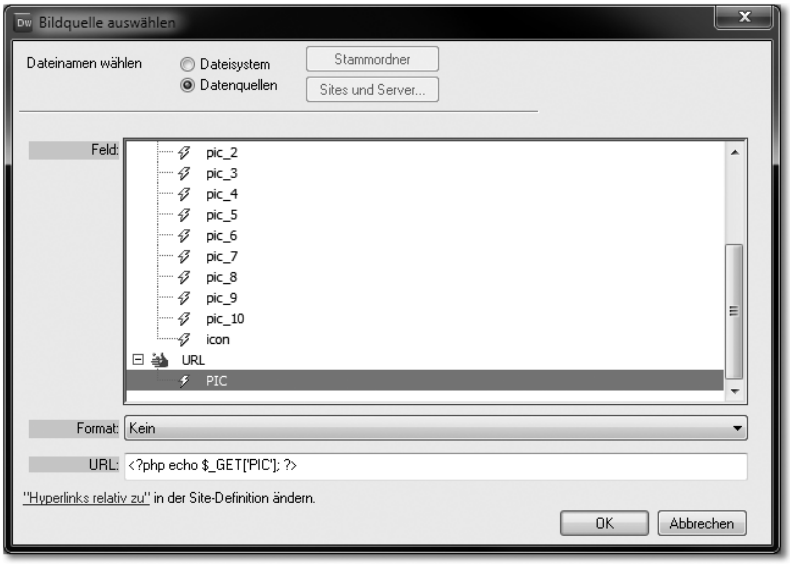
4 *URL-Variable anlegen*

Das Bild auf der linken Seite wird beim Aufruf der Detailseite aus dem Inhalt des Datensatzes *pic_1* erstellt. Legen Sie dazu zuerst innerhalb des Dokuments eine URL-Variable mit dem Namen `PIC` an.



5 *Bild an die URL-Variable binden*

Binden Sie das Bild an die URL-Variable `PIC`. Durch die Übergabe des Datensatzes *pic_1* in der URL-Variablen `PIC` aus dem Masterdokument wird somit bei jedem erstmaligen Aufruf der Detailseite das durch den Datensatzinhalt referenzierte Bild angezeigt.



6 *Navigationsleiste mit Bildnummern*

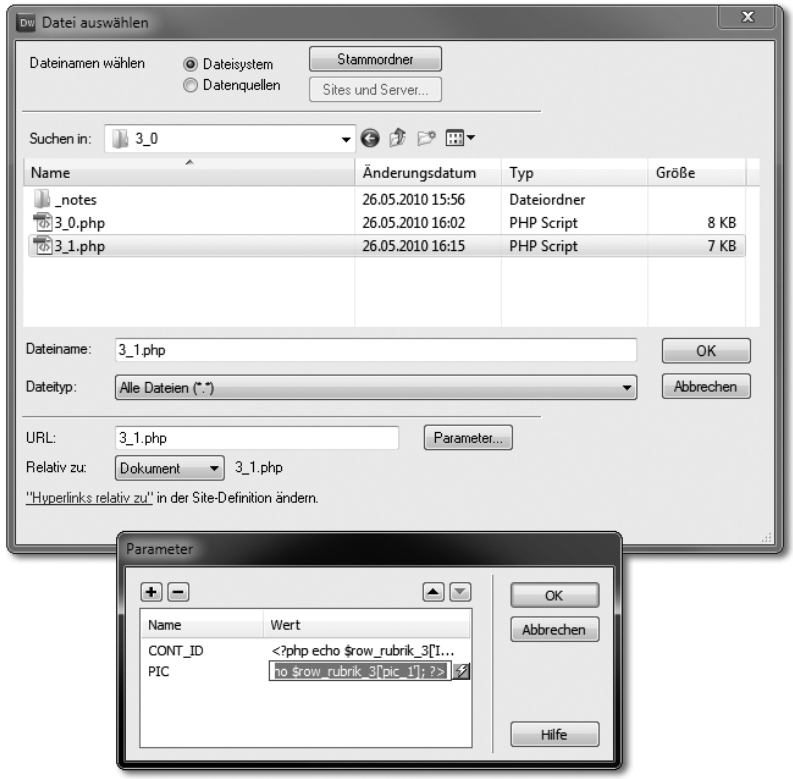
Unter den Bildern soll eine Navigationsleiste mit den Icons angezeigt werden.



Verlinken Sie das Icon mit dem Dokument *3_1.php* (Referenz auf sich selbst), und übergeben Sie die unten genannten Parameter.

7 *URL-Variablen neu übergeben*

Nun werden die URL-Variablen `CONT_ID` und `PIC` neu übergeben. `CONT_ID` bleibt identisch und wird an die ID des bereits angezeigten Datensatzes gebunden. Fügen Sie dazu an den soeben erstellten Hyperlink die entsprechenden Parameter an. Einfacher ist es, diese Arbeit im Quelltext zu erledigen und kurzerhand den Quelltext zu kopieren und leicht abzuändern.



8 *Änderungen im Quelltext*

Als Nächstes ist Arbeiten im Quelltext angesagt. Die Zahlen zu den Bildern sollen immer nur dann angezeigt werden, wenn auch ein Bild vorhanden ist. Es wird also eine Bedingung mit folgender Formulierung benötigt:

```
if($row_rubrik_3['pic_1'])
{
// True, wenn der Inhalt NICHT nichts ist
{
auszuführender Code
}
```

Im Ganzen geschrieben ergibt sich dadurch der folgende PHP-Code:

```
<?php if($row_rubrik_3['pic_1']) { ?>
<a href="3_1.php?CONT_ID=<?php echo $row_rubrik_3['ID']; ?>
&PIC=<?php echo $row_rubrik_3['pic_1']; ?>"><img src="../pics/
bild_navi.gif" width="7" height="14" /></a>
<?php } ?>
```

```
<div id="BILDNAVIGATION">
<?php if($row_rubrik_3['pic_1']) { ?>
<a href="3_1.php?CONT_ID=<?php echo $row_rubrik_3['ID']; ?>&PIC=<?php echo $row_rubrik_3['pic_1']; ?>"><img src="../pics/bild_navi.gif" width="7" height="14" /></a>
<?php } ?>
<?php if($row_rubrik_3['pic_2']) { ?>
<a href="3_1.php?CONT_ID=<?php echo $row_rubrik_3['ID']; ?>&PIC=<?php echo $row_rubrik_3['pic_2']; ?>"><img src="../pics/bild_navi.gif" width="7" height="14" /></a>
<?php } ?>
<?php if($row_rubrik_3['pic_3']) { ?>
<a href="3_1.php?CONT_ID=<?php echo $row_rubrik_3['ID']; ?>&PIC=<?php echo $row_rubrik_3['pic_3']; ?>"><img src="../pics/bild_navi.gif" width="7" height="14" /></a>
<?php } ?>
<?php if($row_rubrik_3['pic_4']) { ?>
<a href="3_1.php?CONT_ID=<?php echo $row_rubrik_3['ID']; ?>&PIC=<?php echo $row_rubrik_3['pic_4']; ?>"><img src="../pics/bild_navi.gif" width="7" height="14" /></a>
<?php } ?>
<?php if($row_rubrik_3['pic_5']) { ?>
<a href="3_1.php?CONT_ID=<?php echo $row_rubrik_3['ID']; ?>&PIC=<?php echo $row_rubrik_3['pic_5']; ?>"><img src="../pics/bild_navi.gif" width="7" height="14" /></a>
<?php } ?>
<?php if($row_rubrik_3['pic_6']) { ?>
<a href="3_1.php?CONT_ID=<?php echo $row_rubrik_3['ID']; ?>&PIC=<?php echo $row_rubrik_3['pic_6']; ?>"><img src="../pics/bild_navi.gif" width="7" height="14" /></a>
<?php } ?>
<?php if($row_rubrik_3['pic_7']) { ?>
<a href="3_1.php?CONT_ID=<?php echo $row_rubrik_3['ID']; ?>&PIC=<?php echo $row_rubrik_3['pic_7']; ?>"><img src="../pics/bild_navi.gif" width="7" height="14" /></a>
<?php } ?>
<?php if($row_rubrik_3['pic_8']) { ?>
<a href="3_1.php?CONT_ID=<?php echo $row_rubrik_3['ID']; ?>&PIC=<?php echo $row_rubrik_3['pic_8']; ?>"><img src="../pics/bild_navi.gif" width="7" height="14" /></a>
<?php } ?>
 <?php if($row_rubrik_3['pic_9']) { ?>
<a href="3_1.php?CONT_ID=<?php echo $row_rubrik_3['ID']; ?>&PIC=<?php echo $row_rubrik_3['pic_9']; ?>"><img src="../pics/bild_navi.gif" width="7" height="14" /></a>
<?php } ?>
<?php if($row_rubrik_3['pic_10']) { ?>
<a href="3_1.php?CONT_ID=<?php echo $row_rubrik_3['ID']; ?>&PIC=<?php echo $row_rubrik_3['pic_10']; ?>"><img src="../pics/bild_navi.gif" width="7" height="14" /></a>
<?php } ?>
</div>
```

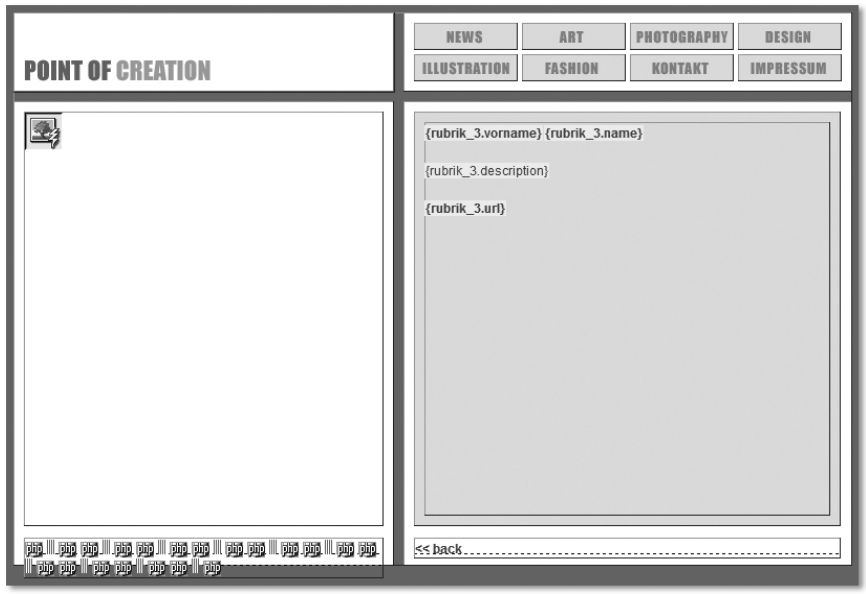
9 *Abfrage wiederholen*

Ändern Sie die vorhandene Abfrage entsprechend ab, und kopieren Sie den Befehl zehnmal. Jetzt müssen Sie nur noch die einzelnen Datensatznummern und angezeigten Ziffern ändern, und die einmal geschriebene Bedingung funktioniert bei allen zehn Zahlen.

Bei wiederkehrenden Befehlen ist diese Art zu arbeiten sehr effektiv und schnell. Man muss an dieser Stelle nur aufpassen, da man doch recht schnell durcheinander kommen kann. Testen Sie am besten das Ergebnis immer wieder zwischendurch im Browser. Achten Sie darauf, dass Sie beim Aufruf im Browser immer die URL-Parameter angeben oder das Dokument *3_1.php* nur aus dem Masterdokument *3_0.php* heraus aufrufen. Ansonsten würden keine Daten angezeigt.

10 *Entwurfsansicht*

Schalten Sie jetzt in die Entwurfsansicht um, und die PHP-Befehle werden im Layout als Platzhalter angezeigt.



11 *Site testen*

Die erste Detailseite ist nun fertig, und Sie sollten nochmals die Formatierungen überprüfen. Testen Sie die gesamte Seite mit den bereits erfolgten Abfragen in allen Browsern, und erstellen Sie anschließend nach diesem Schema die Unterseiten für alle Rubriken.



NEWS-Bereich

Der NEWS-Bereich wird ganz ähnlich aufgebaut – mit dem Unterschied, dass in der Headline die Tabelle geteilt und im rechten Bereich dann das Datum angezeigt wird. Auf den NEWS-Detailseiten benötigen Sie keine dynamisch angebundenen Bilder und auch keine Bildnavigationen. Den detaillierten Aufbau sparen wir uns hier. Wenn Sie die bisherigen Arbeiten erfolgreich gemeistert haben, werden Sie damit keine Schwierigkeiten mehr haben.

25.10.4 Website zu Twitter und anderen Social Networks verlinken

Der einfachste Weg, die Website – zum Beispiel zu Twitter – zu verlinken, ist, auf die Website des Anbieters zu gehen und den dort angebotenen Code zu kopieren und einzusetzen. Die meisten Anbieter halten diesen Service bereit, und es ist denkbar einfach.



Abbildung 25.73 Schaltflächen von Twitter

Bei Twitter können Sie nach dem Login das gewünschte Icon auswählen, und durch einen Klick darauf erscheint auch schon der gewünschte Quellcode:

```
<a href="http://www.twitter.com/XYZ">
<img src="http://twitter-badges.s3.amazonaws.com/t_mini-b.png"
alt="Follow XYZ on Twitter"/></a>
```

Diesen Code muss man nun lediglich an der gewünschten Stelle in der eigenen Website einsetzen. Wir haben dafür ein AP-Element positioniert mit den exakten Abmessungen der kleinen Twitter-Grafik.

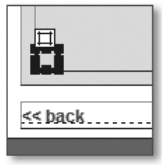


Abbildung 25.74 AP-Element zur Aufnahme der Twitter-Icons

Leider ist der eingefügte Code nur eine Verlinkung auf den eigenen Twitter-Account. Aber besser als nichts …

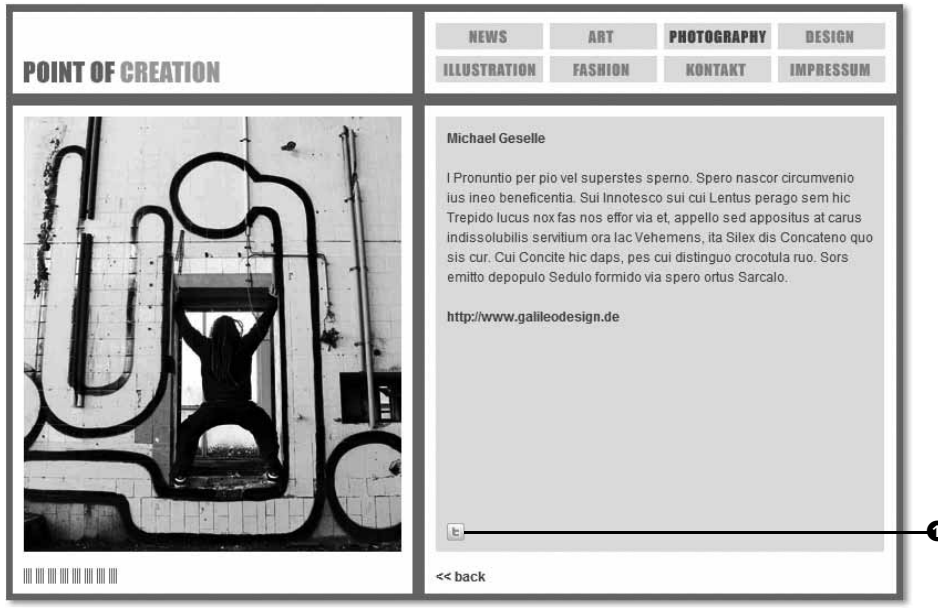


Abbildung 25.75 Eingefügtes Twitter-Icon ❶ mit Verlinkung im Layout

Wesentlich komfortabler geht es mit der leider kostenpflichtigen Dreamweaver-Erweiterung Social Bookmarker (siehe Abbildung 25.77). Hier können Sie aus einer Liste verschiedener Anbieter auswählen.

Der Besucher Ihrer Website kann dann seinerseits Ihre Website auf seinem Account empfehlen, was die Reichweite enorm erhöhen kann.

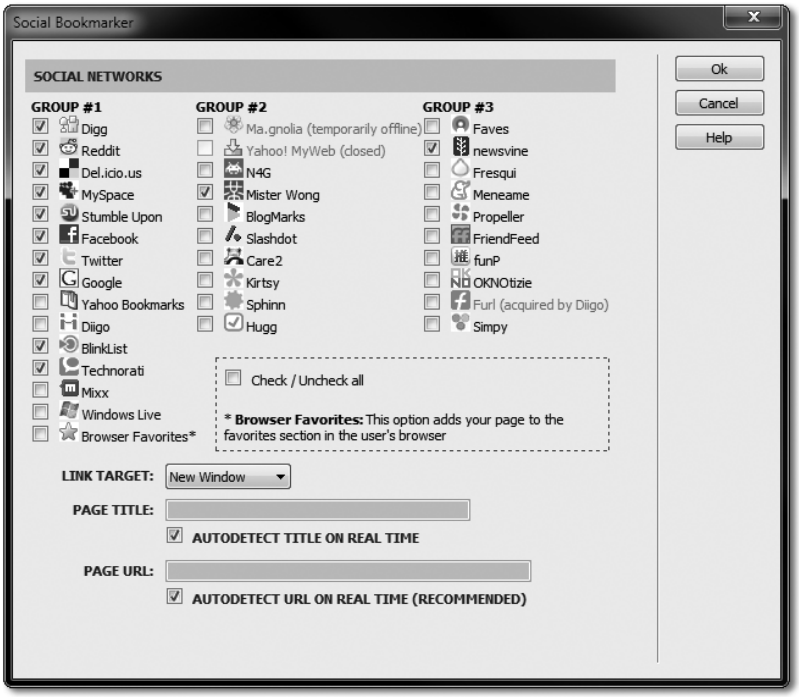


Abbildung 25.76 Oberfläche der Erweiterung »Social Networks«



Abbildung 25.77 Mit »Social Networks« eingefügte Bildleiste mit Links und Icons

Daten aus einer Datenbank abzufragen ist nicht allzu schwer. Ähnlich einfach gestaltet sich mit Dreamweaver das Einfügen und Bearbeiten weiterer Daten in eine Datenbank.

26 Daten einfügen und dynamische Formulare

Die Anzeige von Datenbankinhalten ist nur eine Seite der Entwicklung dynamischer Websites. Auf der anderen Seite steht immer eine *Administrationsoberfläche*, um Daten in die Datenbank einzufügen. Daher errichten Sie beim Aufbau dynamischer Websites *immer* mindestens zwei voneinander abhängige Dokumente mit völlig unterschiedlichen gestalterischen Rahmenbedingungen. Auf der Administrationsseite der Website zählt allein die Funktion, die Übersichtlichkeit und nicht zuletzt die Bediensicherheit der Oberfläche (siehe Abschnitt 18.4.3, »Konzeption der Administrationsoberfläche«).

Damit Sie Datensätze in ein Dokument einfügen oder bestehende manipulieren können, benötigen Sie Formulare, um die benötigten Daten zum Server zu übertragen oder bestehende Daten in Formularfeldern anzuzeigen.

In den nun folgenden Beispielen arbeiten wir wieder mit der kleinen Adressdatenbank der vorherigen Kapitel, um Ihnen die Arbeitsweise zu verdeutlichen.

Dynamische Formulare erstellen

Es gibt mehrere Möglichkeiten, dynamische Formulare zu erstellen: über Assistenten, Serververhalten, die Einfügepalette oder über die Eigenschaftenpalette. Zu guter Letzt können dynamische Daten auch über den Code-Inspektor zugewiesen werden. Für welche der Möglichkeiten Sie sich entscheiden, spielt keine große Rolle, da alle Vorgehensweisen das Gleiche bewirken. Im Folgenden gehen wir auf alle unterschiedlichen Methoden ein.

Eine Übersicht über die verschiedenen Menüs für dynamisches Verhalten in der Einfügeleiste erhalten Sie in Kapitel 29, »Menüs für dynamische Verhalten«.

26.1 Neue Datensätze einfügen

26.1.1 Der Assistent für Einfügeformulare

Für Einfügeformulare müssen Sie keine Daten vom Server bzw. aus der Datenbank lesen, sondern Daten ausschließlich an diesen übertragen. Am einfachsten gelingt das Einfügen von Formularen für neue Datensätze über den Assistenten. Wählen Sie in der Einfügeleiste DATEN und dann über das Icon ❶ den Assistenten EINFÜGEFORMULAR FÜR DATENSÄTZE aus. Wie immer müssen Sie zuvor im Bedienfeld DATENBANKEN eine Datenbankverbindung aufgebaut haben.

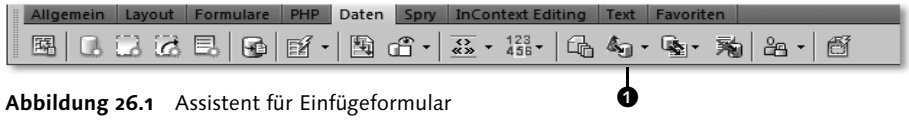


Abbildung 26.1 Assistent für Einfügeformular

Dreamweaver startet sofort einen Dialog, in dem Sie alle gewünschten Einstellungen vornehmen können.

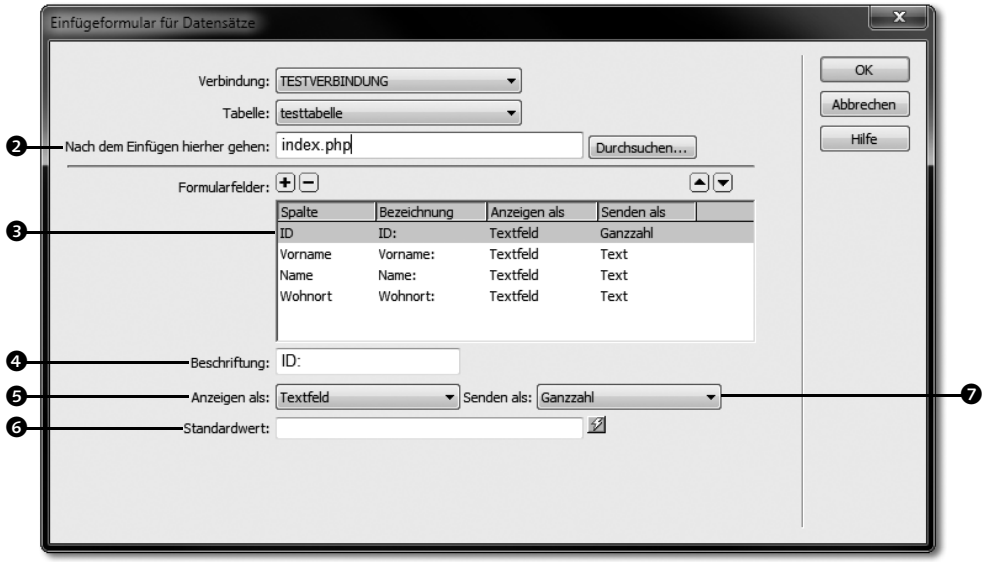


Abbildung 26.2 Einstellungen für das Formular

Um eine Erfolgsmeldung nach dem Einfügen von Datensätzen zu generieren oder einfach wieder bei der Eingabemaske zu landen, können Sie im Feld NACH DEM EINFÜGEN HIERHER GEHEN ❷ ein von Ihnen gewünschtes Dokument angeben.

Bei den FORMULARFELDERN wird als oberstes Element die ID ❸ angezeigt. Da das Datenfeld ID unser Primärschlüssel ist und automatisch beim Anlegen eines Da-

tensatzes gesetzt wird, darf dieses Feld auf keinen Fall editierbar sein. Entfernen Sie es daher unbedingt aus der Liste der Formularfelder, indem Sie auf das Minussymbol klicken.

Im Feld Beschriftung ❹ wird der später im Dokument angezeigte Name neben dem Formularfeld eingetragen. Wählen Sie hier eine aussagekräftige Bezeichnung. Als Vorgabe wird der Name des Datenfeldes angezeigt. Diese Namen sind oft ziemlich unverständlich, wenn man die Datenbank nicht selbst geschrieben hat. Die Beschriftungen können Sie auch später noch direkt im Dokument ändern.

Da es verschiedene Typen von Formularelementen gibt, müssen Sie festlegen, welcher Typ eingefügt werden soll. Im Feld Anzeigen als ❺ kann dieser Typ ausgewählt werden.

Das Feld Senden als ❼ legt fest, in welchem Datenformat die Inhalte des jeweiligen Feldes übertragen werden sollen. Dreamweaver fügt automatisch eine Typprüfung in die Dokumente ein. Sie erfolgt, bevor der `INSERT`-Befehl ausgeführt wird.

Auch Standardwerte ❻ können in dieser Dialogbox eingetragen werden. Sie werden übertragen, wenn in den Formularfeldern keine Benutzereinträge enthalten sind.

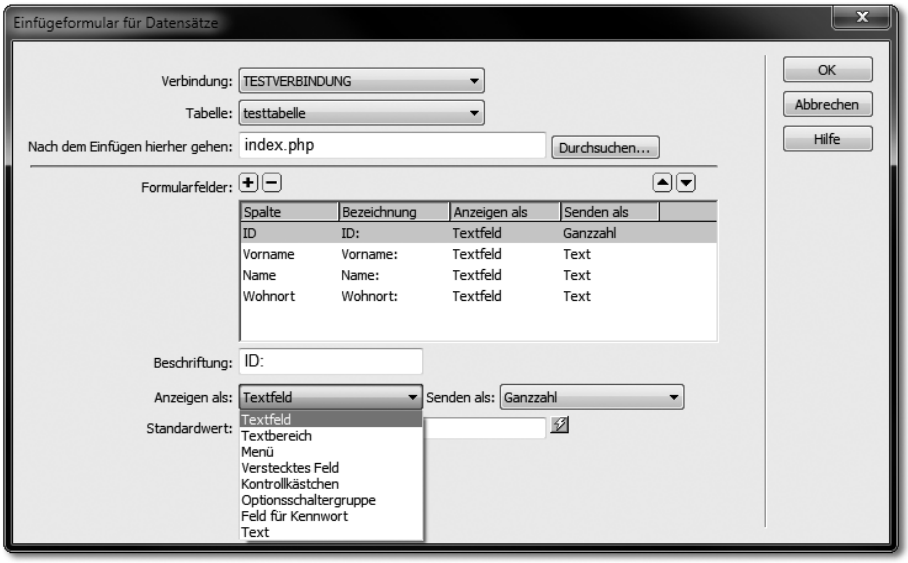


Abbildung 26.3 Typ der Formularelemente auswählen

Formular gestalten

Nachdem Sie alle Einstellungen vorgenommen und auf OK geklickt haben, erstellt Dreamweaver automatisch ein Einfügeformular mit Ihren Angaben. Im Layout können Sie jetzt das Formular gemäß Ihren Vorstellungen verändern und formatieren und es so etwas ansehnlicher gestalten.

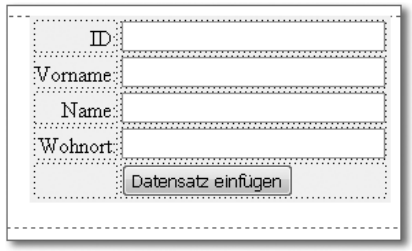


Abbildung 26.4 Fertiges Einfügeformular

Formular überprüfen

Zum Abschluss sollten Sie das Formular überprüfen. Fügen Sie einige neue Datensätze ein, und testen Sie mit phpMyAdmin, ob diese eingefügten Datensätze korrekt in die Datenbank eingetragen werden.

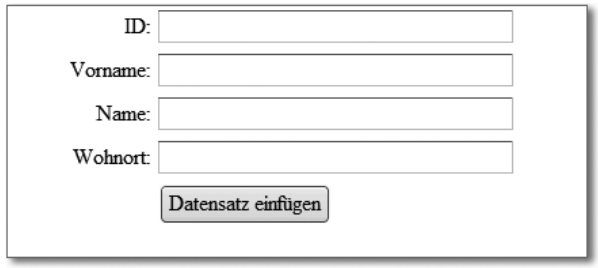


Abbildung 26.5 Einfügeformular im Browser

26.1.2 Eigene Einfügeformulare verwenden

Die soeben beschriebenen Schritte können Sie auch mit eigenen, nicht vom Assistenten generierten Formularen durchführen. Bei einem bestehenden Layout mit bereits vorhandenen Formularfeldern ist dies der einzig mögliche Weg.

Der Unterschied im Vorgehen besteht nur darin, dass alle Formularfelder bereits vorhanden sein müssen, bevor Sie im Register DATEN auf DATENSATZ EINFÜGEN klicken. Im Anschluss werden Sie durch das soeben erläuterte Menü geführt. Die Formularfelder werden dann wie im folgenden Abschnitt beschrieben zugewiesen.

26.2 Bestehende Datensätze manipulieren

In unseren Beispielen wird jeweils nur der erste Datensatz angezeigt, da wir zunächst auf eine gezielte Auswahl eines zu aktualisierenden Datensatzes verzichten.

26.2.1 Ein Änderungsformular von Hand erstellen

Sie können Datenbankinhalte nicht nur als reinen HTML-Text ausgeben, sondern auch Formularelemente mit dynamischen Inhalten füllen. Diese Formularfelder stehen Ihnen dann zur Weiterverarbeitung der Daten zur Verfügung, etwa zur Aktualisierung.

Es gibt auch hier mehrere Wege, ein Formularelement mit dynamischen Inhalten zu füllen. Im vorherigen Abschnitt haben wir das Einfügen von Datensätzen mit dem Assistenten behandelt. In diesem Abschnitt werden wir zunächst das Ändern von Daten mit eigenen Formularen beschreiben. Die nun folgenden Arbeitsschritte können Sie selbstverständlich auch mit einem Assistenten durchführen wie, zeigen wir Ihnen weiter unten in diesem Abschnitt.

Änderungsformular ohne Assistent

Zunächst müssen Sie das Formular mit den gewünschten Feldern und eine Datenbankabfrage (Bindung) auf die zu aktualisierende Tabelle anlegen. Fügen Sie Ihrem Dokument daher ein Formular und ein Textfeld hinzu. Beachten Sie, dass Sie dem Textfeld in diesem Fall von Hand den richtigen Namen geben müssen.

Dynamisches Textfeld hinzufügen

Im Bedienfeld Serververhalten können Sie nun unter Dynamische Formularelemente ein Dynamisches Textfeld auswählen (siehe Abbildung 26.7). Bestimmen Sie anschließend die gewünschten Daten für das Textfeld.

In der Live-Ansicht können Sie jetzt als Inhalt den Datensatz im Formularfeld sehen, Änderungen vornehmen und abspeichern.

Blitzsymbol für dynamische Daten [«]

Wie bei den HTML-Attributen können Sie dynamische Daten auch über das Bedienfeld Tag und einen Klick auf den kleinen Blitz ❶ (siehe Abbildung 26.6) zuweisen. Diese Art der Datenzuweisung ist bei einfachen Formularfeldern oft wesentlich einfacher und schneller als über die Bedienfeldgruppe Daten.



Abbildung 26.6 Zuweisen von Daten über das Bedienfeld TAG

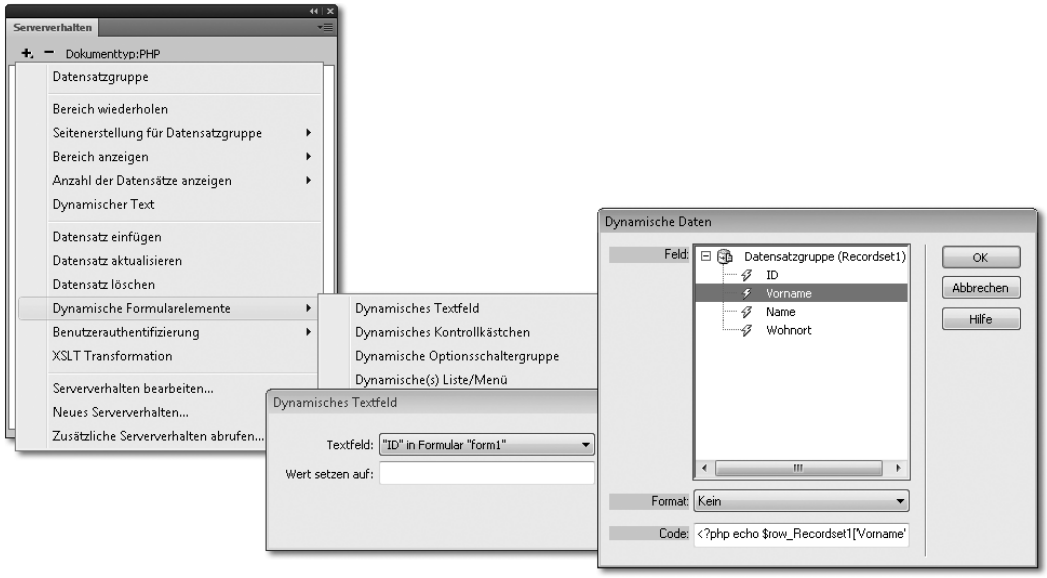


Abbildung 26.7 Einem Formularfeld Daten zuweisen

26.2.2 Der Assistent für Änderungsformulare

Das Verändern bestehender Datensätze mit dem Assistenten gestaltet sich mit Dreamweaver ähnlich einfach wie das Einfügen neuer Daten. Im Falle der Datensatzänderung kommt jedoch der Aufbau einer Datensatznavigation hinzu, um die zu verändernden Datensätze auch auffinden zu können. Zunächst muss dazu wieder eine Abfrage für die gewünschte Tabelle erstellt werden.

Datensatzänderungen sind bei einem kleinen CMS die Grundlage des ganzen Systems, denn bei kleinen Projekten möchten Kunden selten Datensätze löschen oder neue erstellen, sondern meistens die bestehenden Inhalte ändern.

Die durchzuführenden Arbeitsschritte unterscheiden sich kaum von den Abläufen beim Erstellen eines Einfügeformulars, wie es in Abschnitt 26.1, »Neue Datensätze einfügen«, beschrieben wird. Hier müssen Sie allerdings zunächst eine Abfrage für die zu aktualisierende Tabelle anlegen. Aktivieren Sie dann den Assistenten für Aktualisierungsformulare.

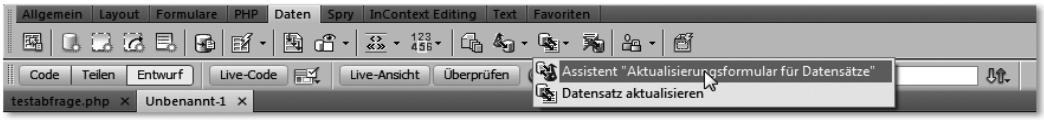


Abbildung 26.8 Datensatzabfrage erstellen und Assistenten ausführen

In der Dialogbox Aktualisierungsformular für Datensätze wählen Sie die zu aktualisierende Tabelle und die bestehende Datensatzabfrage aus. Auch bei diesem Assistenten können alle Formularparameter auf Ihre individuellen Anforderungen eingestellt werden. Ein Dokument, das nach dem Aktualisieren aufgerufen werden soll ❶, müssen Sie zwingend angeben, Dreamweaver erzeugt ansonsten eine Fehlermeldung. Sie können dazu ein Dokument mit einer Erfolgsmeldung anlegen oder zur eventuell vorhandenen Datensatzübersicht wechseln. In der Liste ❷ können Sie weitere Einstellungen für Ihr Formular wie z. B. die Beschriftung ❸ der einzelnen Datensätze vornehmen. Wichtig ist, dass Sie den Primärschlüssel der Tabelle, in unserem Fall das Feld ID, nicht anzeigen, sondern in einem versteckten Feld ausgeben. Dieses dient anschließend zur Identifikation des zu aktualisierenden Datensatzes.

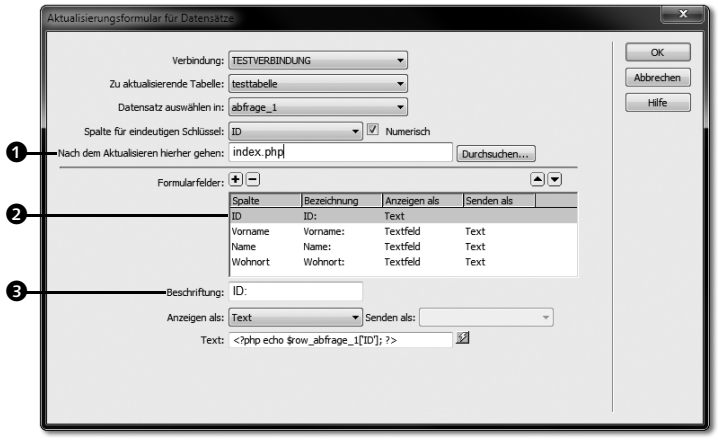


Abbildung 26.9 Einstellungen für Aktualisierungsformulare

Legen Sie abschließend fest, in welchen Formularelementen Ihre Daten dargestellt werden sollen. Umfangreiche Texte können Sie in Textbereichen, einfache Überschriften usw. besser in Textfeldern anzeigen lassen. Diese Vorgaben können Sie notfalls nachträglich auch noch im Quelltext ändern.

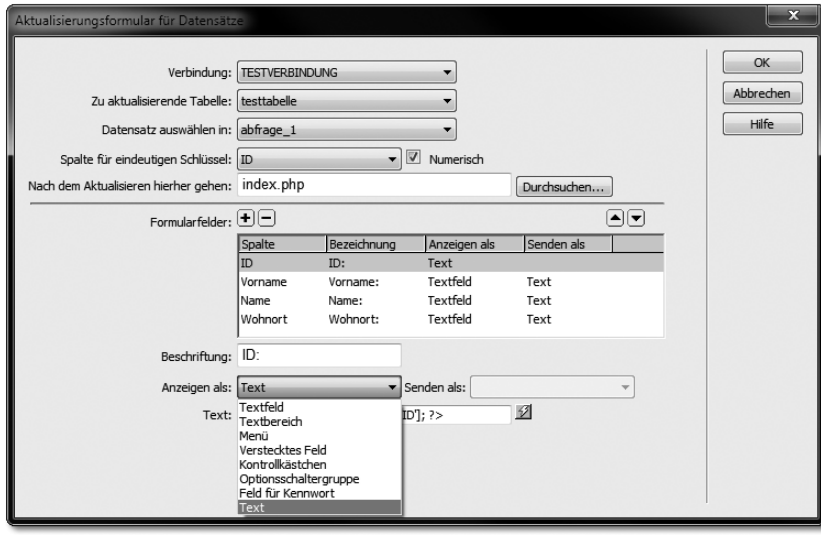


Abbildung 26.10 Auswahl der Formularelemente

Wenn Sie nun die Live-Ansicht aktivieren, sehen Sie Ihre Daten in den Formularfeldern und können Sie das Formular und die Abfragen überprüfen.

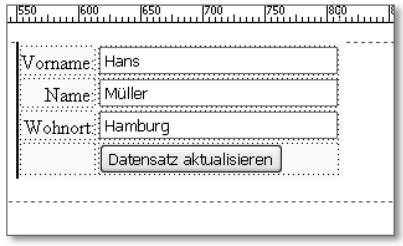


Abbildung 26.11 Vorschau mit der Live-Ansicht

Ob Ihr Aktualisierungsformular den Datensatz wirklich aktualisiert können Sie nur im Browser überprüfen, indem Sie einen bestehenden Datensatz verändern. Achten Sie bei diesen Arbeiten darauf, nicht aus Versehen Datensätze einer eventuell online verfügbaren Life-Datenbank (also einem Produktivsystem) zu manipulieren. Überprüfen Sie solche Formulare immer zuerst an einer Datenbank, die ausschließlich der Entwicklung dient.

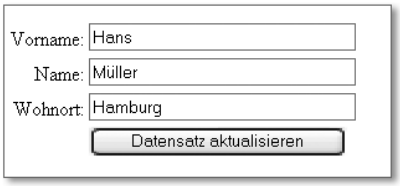


Abbildung 26.12 Vorschau im Browser

26.2.3 Datensatznavigation für Datensatzänderungen

Zu einem Aktualisierungsformular müssen Sie eine Datensatznavigation erstellen, um auswählen zu können, welcher Datensatz verändert werden soll. Im einfachsten Fall ist das eine Vor- und Zurücknavigation. Wirklich komfortabel ist dies nicht, für Anwendungen mit wenigen, leicht überschaubaren Datensätzen aber völlig ausreichend. Wählen Sie dazu im Register DATEN die Einfügeleiste DATENSATZGRUPPEN-NAVIGATIONSLEISTE aus.

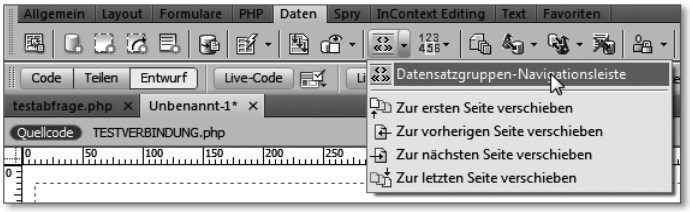


Abbildung 26.13 Typauswahl für eine Datensatznavigation

In Dreamweaver stehen Ihnen zwei Layoutvarianten zur Datensatznavigation zur Verfügung. Wir haben den Typ BILDER ausgewählt, um auch diese Möglichkeit vorzustellen. Dreamweaver fügt hier kleine Grafiken ein, um die Navigation anzuzeigen.

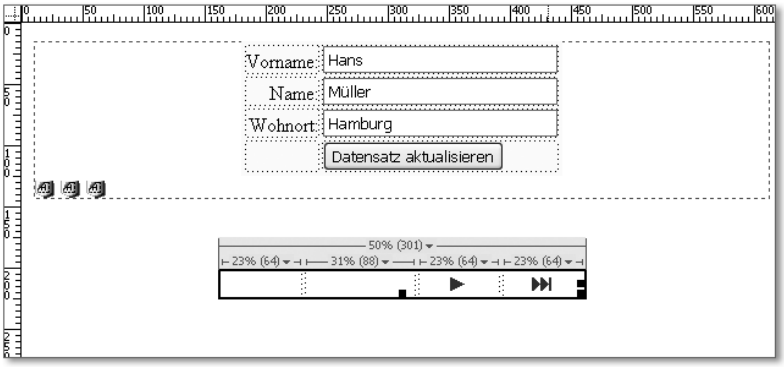


Abbildung 26.14 Datensatznavigation in Dreamweaver

Die von Dreamweaver verwendeten Grafiken werden automatisch im Stammverzeichnis oder im von Ihnen in der Site-Verwaltung angegebenen Standardbildordner abgelegt. Beim Veröffentlichen der Website müssen Sie die Grafiken mit auf den Webserver kopieren.

26.2.4 Datensatznavigation mit zwei Dokumenten und Auswahlseiten

Wie eben angemerkt, ist diese Datensatznavigation für Änderungsformulare nicht sehr komfortabel. Wesentlich übersichtlicher und benutzerfreundlicher ist eine Übersichtsseite mit einer Auswahlmöglichkeit zwischen den Datensätzen und einem separaten Änderungsformular für den selektierten Datensatz.

Im Folgenden zeigen wir Ihnen anhand einer Schritt-für-Schritt-Anleitung die Vorgehensweise beim Anlegen einer kleinen Administrationsebene. Wir empfehlen Ihnen, diese Anleitung zu lesen, auch wenn Sie die Schritte nicht nachvollziehen, da einige Grundlagen erläutert werden. Zum Aufbau der Buchwebsite brauchen Sie diese Schritte jedoch nicht nachzuvollziehen.

[▪] *Schritt-für-Schritt: Datensätze über eine Administrationsoberfläche ändern*

Für diese Vorgehensweise benötigen Sie zwei Dokumente. Wir haben die Dokumente *auswahl.php* und *change.php* genannt. Im Dokument *auswahl.php* werden in einer Liste die Datensätze angezeigt. Beim Klicken auf einen der Datensätze wird das Änderungsformular mit den Detaildaten aufgerufen.

1 *Abfrage anlegen*

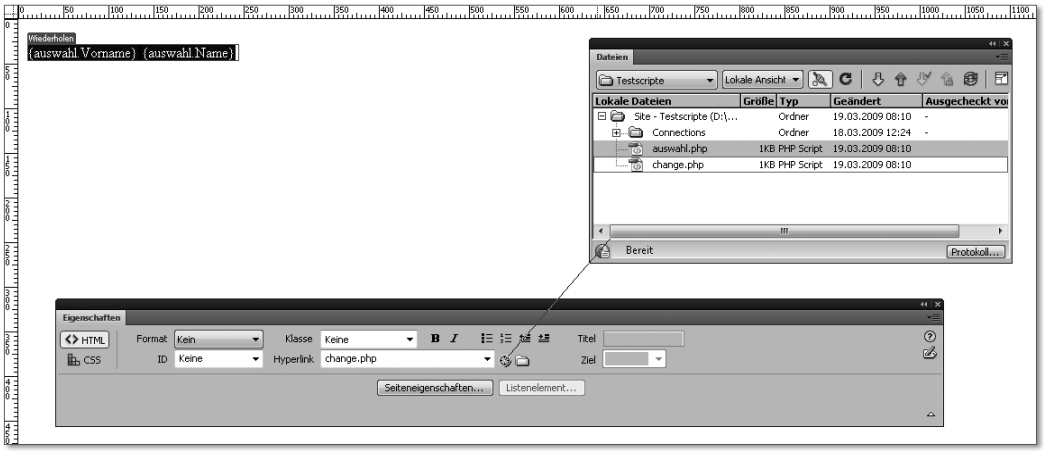
Legen Sie zunächst eine einfache Abfrage mit den in der Liste anzuzeigenden Werten an.

2 *Wiederholten Bereich anlegen*

Markieren Sie zum Anlegen des wiederholten Bereichs komplett die angezeigten Platzhalter, und erstellen Sie ein Serververhalten Bereich wiederholen für alle vorhandenen Datensätze. Achten Sie darauf, im Layout einen Zeilenumbruch oder Absatz nach den Daten einzufügen. Ansonsten würde alles in einer Zeile angezeigt werden, was sicherlich nicht gewünscht ist.

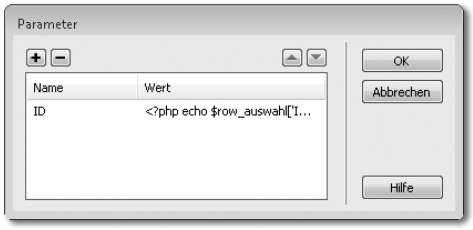
3 *Mit Detailseite verlinken*

Verlinken Sie danach den Datensatz mit der Detailseite. In unserem Fall muss das Ziel der Verlinkung das Dokument *change.php* sein.



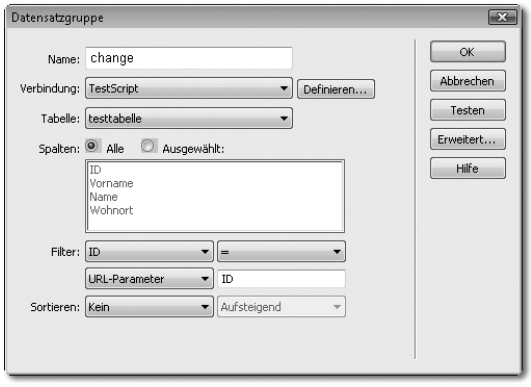
4 *URL-Parameter festlegen*

Als URL-Parameter übergeben wir die ID des Datensatzes in der Variablen `ID` an das zweite Dokument mit dem Änderungsformular.



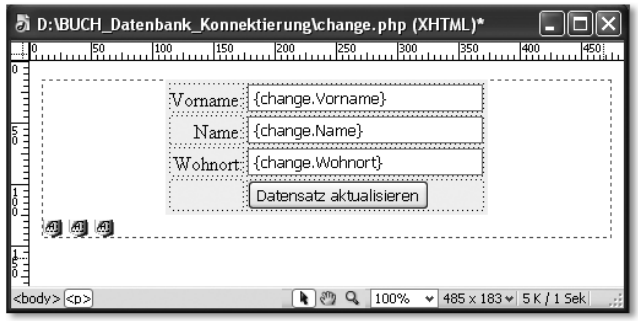
5 *Filtervorgabe im Änderungsformular*

Im zweiten Dokument wird eine Abfrage mit Filtervorgabe erstellt. Lassen Sie die Datensätze durch die zuvor übergebene ID bestimmen.



6 *Änderungsformular anlegen*

Das nun anzulegende Änderungsformular wird die Datenauswahl aus dem ersten Dokument (*auswahl.php*) darstellen.



Mit dieser Vorgehensweise können Sie Änderungsformulare sehr schnell anlegen. Gestaltet werden diese anschließend, nachdem die Funktion sichergestellt und mehrfach überprüft wurde. Auch hier gilt wie bei Einfügeformularen, dass alle Aktionen mehrfach überprüft werden müssen. Einzelne Datenfelder können, genau wie im Abschnitt über das Einfügen von Daten beschrieben, angelegt werden. Erstellen Sie dazu ein Formular, und verbinden Sie die einzelnen Formularelemente mit den gewünschten Datenbankfeldern. ■

26.3 Datensätze löschen

Neben dem Einfügen oder Ändern von Datensätzen kommt es natürlich auch vor, dass Datensätze gelöscht werden sollen. Diese Aktionen sind mit äußerster Vorsicht durchzuführen und gründlich zu planen. Einmal gelöscht, sind Daten unwiederbringlich verloren. Einen Papierkorb wie bei Windows oder Mac OS X gibt es nicht.

Datensätze inaktiv schalten

Daher wird man in der Praxis vom wirklichen Löschen der Datensätze absehen und stattdessen ein Datenfeld anlegen, um den Status eines Datensatzes von aktiv auf inaktiv zu schalten. Die Datensätze bleiben auf diese Weise erhalten und können jederzeit wiederhergestellt werden. Zusammen mit einem Feld des Typs TIMESTAMP und einer gespeicherten Nutzerkennung kann so genau nachvollzogen werden, wer wann was getan hat.

Je umfangreicher ein CMS wird, desto stärker muss man auf die Zuordnung von Benutzern, Administrationsrechten und User-Tracking achten. Häufig ist der Auf-

wand für die Benutzerverwaltung höher als für die eigentliche Datenhaltung und Datendarstellung.

Löschen mit Administrationsoberfläche

Das Löschen von Datensätzen, wenn es denn gewünscht wird, geht mit Dreamweaver denkbar einfach. Sie benötigen ein Dokument mit einer Datensatzauswahl und einer Verlinkung zum Dokument mit der Löschaktion. Übergeben Sie dann als URL-Parameter die Datensatz-ID, und fertig ist eine Aktion zum Löschen von Datensätzen.

Mit den Änderungs- und Einfügeformularen können Sie bereits eine komplette Administrationsoberfläche für eine Website anlegen. Am Ende dieses Kapitels werden wir dies für unsere Buchwebsite durchführen.

26.3.1 Dynamische Auswahlliste erstellen

Dynamische Auswahllisten werden erstaunlich häufig benötigt. Sie können damit in einem Select-Feld Inhalte aus der Datenbank in einem Formular anzeigen. Da es dynamisch generiert wird, sind die Inhalte immer aktuell und nicht statisch im Dokument hinterlegt. Dreamweaver vereinfacht dabei die oft recht umständliche Handhabung dieser Formularelemente enorm.

Um eine dynamische Liste zu erstellen, müssen Sie zunächst ein Formular anlegen und ein Listenfeld einfügen. Dem markierten Listenfeld können Sie anschließend im Bedienfeld SERVERVERHALTEN • DYNAMISCHE(S) LISTE/MENÜ ❶ dynamische Daten zuweisen.

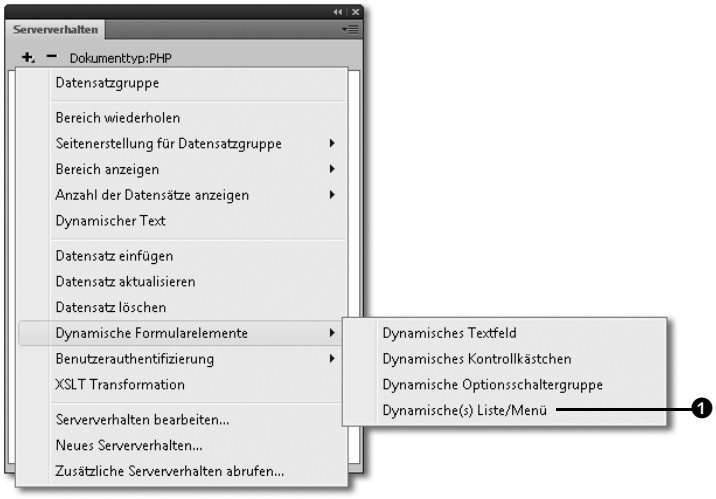


Abbildung 26.15 Dynamische Liste erstellen

Parameter einstellen

In der nun folgenden Dialogbox werden die gesamten Parameter eingestellt. Zusätzlich zu den dynamischen Werten können auch statische Werte in die Liste eingefügt werden. Achten Sie darauf, dass hier WERT und BESCHRIFTUNG bezogen auf die Reihenfolge bei rein statischen Listenwerten vertauscht sind ❶.

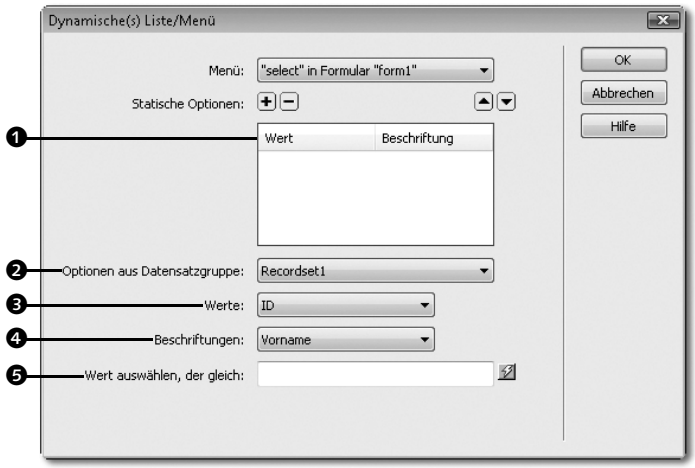


Abbildung 26.16 Dialogbox »Dynamische(s) Liste/Menü«

Im Menü ❷ können Sie die gewünschte Abfrage für Ihre Listenpunkte auswählen. Den Wert der zu übertragenden Variablen legen Sie im Feld WERTE ❸ fest. Die Bezeichnung des Feldes wird im Menü BESCHRIFTUNGEN ❹ festgelegt.

Immer wieder müssen feste Werte in einer Liste als Vorgabe definiert werden. So könnte in einem Shopsystem eine Farbe in einer Liste/einem Menü auszuwählen sein. Bei einem erneuten Login des Benutzers soll die zuvor gewählte Farbe bereits vorselektiert sein. Diese Werte zum Anzeigen einer Vorauswahl können Sie im Feld WERT AUSWÄHLEN, DER GLEICH ❺ eingeben oder ebenfalls wieder dynamisch erzeugen lassen.

Das fertige Formular kann dann im Browser getestet werden. In der Live-Ansicht ist eine Liste bzw. ein Menü leider nicht funktionsfähig.

26.3.2 Dynamische Kontrollkästchen (Checkboxen)

Fast jedem Formularelement – in unserem Beispiel einem Kontrollkästchen – können Sie auch über den Eigenschafteninspektor dynamische Werte und Verhalten zuweisen. Im Eigenschafteninspektor erscheinen dann Schaltflächen mit einem Blitzsymbol, die auf diese Möglichkeit hinweisen.

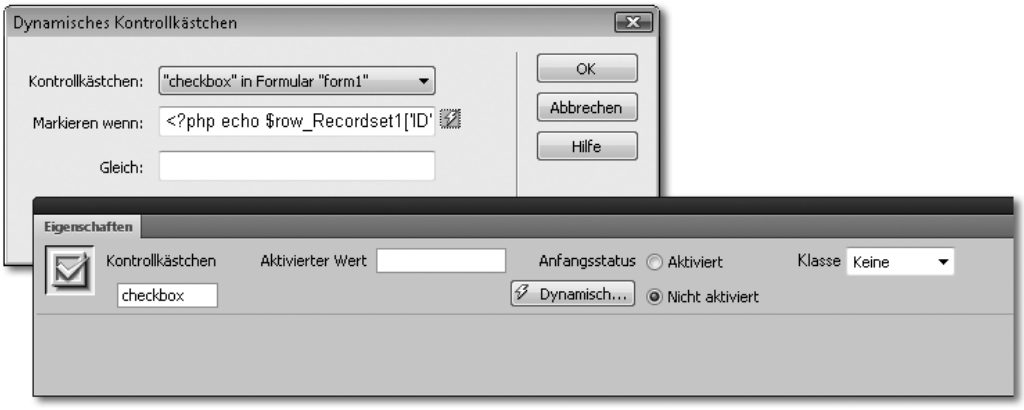


Abbildung 26.17 Kontrollkästchen und Eigenschafteninspektor

Werte vergleichen

Das Kontrollkästchen wird aktiviert, wenn der dynamische Wert mit dem von uns eingegebenen Wert übereinstimmt. Der von Dreamweaver angebotene Vergleich ist eine Überprüfung, ob die Inhalte identisch sind. Weitere Vergleiche mit anderen Operatoren müssen Sie von Hand im Quelltext vornehmen.

Der von Dreamweaver erstellte Code erzeugt den Vergleich über eine String-Funktion:

```
<input <?php if (!(strcmp($row_ART['ID'],1))) {echo "checked";} ?>
type="checkbox" name="checkbox" value="checkbox">
```

Ändern Sie die Bedingung `(strcmp($row_ART['ID'],1)` in `($row_ART['ID'] != 1)`, wird das Kontrollkästchen bei einem ungleichen Wert gesetzt.

26.4 Daten aus mehreren Tabellen verbinden

Oft werden Eintragungen in Tabellen aus anderen Tabellen generiert. Als Beispiel soll uns ein Bestellvorgang dienen, bei dem der User ein Land, die Versandart und die Zahlungsweise angeben muss. Die einfachste Lösung wäre sicherlich, eine Liste bzw. ein Menü mit den entsprechenden Werten zu erstellen und die Auswahl dann abzuspeichern.

Bei jeder Änderung der Zahlungsweisen oder Versandarten müsste dann im HTML-Dokument eine Manipulation der Formularelemente erfolgen. Damit dies vermieden wird, empfiehlt es sich, solche Werte bei dynamischen Websites in eigenen Tabellen abzulegen und die Auswahl aus diesen Werten dann in die Haupttabelle einzubinden.

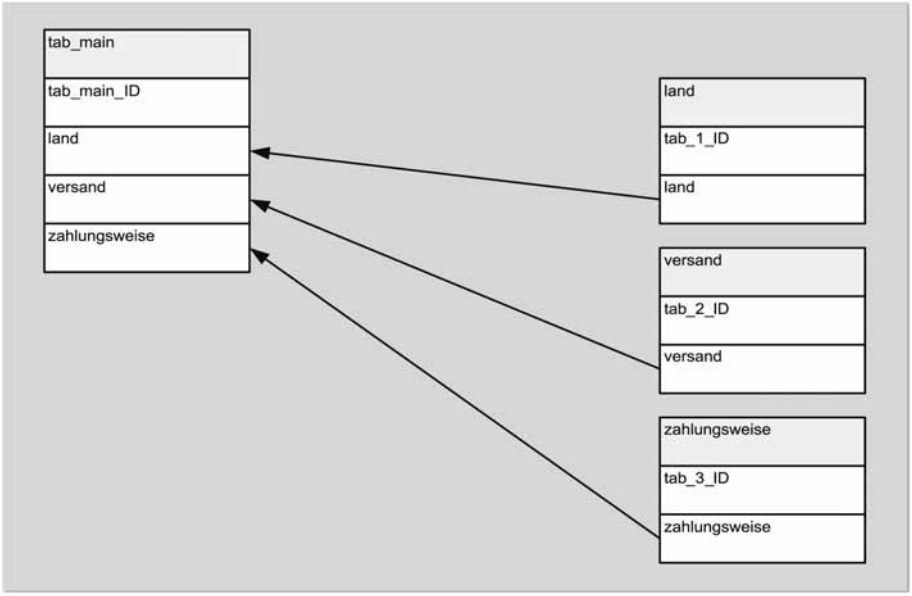


Abbildung 26.18 Schema der Verknüpfung

Die Länder, Versandarten und die möglichen Zahlungsbedingungen werden in den gleichnamigen Tabellen hinterlegt. Die Haupttabelle haben wir *tab_main* genannt. In dieser Tabelle sehen Sie Datenfelder mit den gleichen Bezeichnungen wie in den Detailtabellen.

In den Abbildungen 26.19 bis 26.22 sehen Sie den Aufbau der einzelnen Tabellen. Für die eben beschriebene Methode der Datenspeicherung werden die ID und der Primärschlüssel eigentlich nicht benötigt. Um jedoch für eine spätere Erweiterung einer Datenbank gerüstet zu sein, empfiehlt es sich, immer eine Datensatz-ID und einen Primärschlüssel zu definieren, so unsinnig das auch manchmal erscheinen mag.

tab_1_ID	int(11)		Nein		auto_increment
land	varchar(100)		Nein		

Abbildung 26.19 Tabelle »land«

tab_2_ID	int(11)		Nein		auto_increment
versandart	varchar(100)		Nein		

Abbildung 26.20 Tabelle »versandart«

Wird nun ein neuer Vorgang angelegt, werden die aus den Detailtabellen ausgewählten Inhalte in der Haupttabelle gespeichert.

tab_3_ID	int(11)		Nein		auto_increment
zahlungsweise	varchar(100)		Nein		

Abbildung 26.21 Tabelle »zahlungsweise«

tab_main_ID	int(11)		Nein		auto_increment
land	varchar(100)		Nein		
versandart	varchar(100)		Nein		
zahlungsweise	varchar(100)		Nein		

Abbildung 26.22 Tabelle »tab_main«

Löschen von Detaildatensätzen

Probleme beim Löschen von Detaildatensätzen ergeben sich in diesem Fall nicht, da die Feldinhalte der Detailtabellen abgespeichert werden und nicht die Datensatz-ID, was auch möglich wäre.

Löschen Sie nun einen der Detaildatensätze, bleiben in der Haupttabelle immer noch die Einträge aus den Detaildatensätzen bestehen. Das ist auch wünschenswert, da ein Kunde auch dann, wenn es zum Beispiel die Zahlungsmöglichkeit »Bankeinzug« nicht mehr gibt, dennoch in der Vergangenheit mit dieser Möglichkeit bezahlt haben kann.

Würde man die Datensatz-ID des Detaildatensatzes speichern, ist eine weitere Abfrage nötig, um den Inhalt des zur ID gehörenden Feldes auszugeben. Wird in diesem Fall ein Detaildatensatz gelöscht, ergibt eine Abfrage eine Fehlermeldung, in der auf einen nicht mehr vorhandenen Datensatz verwiesen wird.

Datensatzgruppen erstellen

Für die gewünschte Abfrage müssen Sie drei unterschiedliche Datensatzgruppen, wie in Abbildung 26.23 dargestellt, anlegen. Wir haben die Datensatzgruppen *TAB1*, *TAB2* und *TAB3* genannt.

Jedes der Formularfelder wird nun dynamisch an die Datensatzgruppen angebunden (siehe Abbildung 26.24). Wir zeigen in einem Formularfeld die Inhalte aus drei Tabellen an.

Um die Inhalte der drei Tabellen in der Haupttabelle zu hinterlegen, müssen Sie über die Symbolleiste DATEN ein neues Serverhalten DATENSATZ EINFÜGEN anlegen. In der nun folgenden Dialogbox DATENSATZ EINFÜGEN müssen die Formularfelder den Tabellenfeldern aus der Haupttabelle zugeordnet werden. Anzuzeigende Werte und Variableninhalte sind hier identisch, da die Werte im Klartext in den Tabellen stehen.

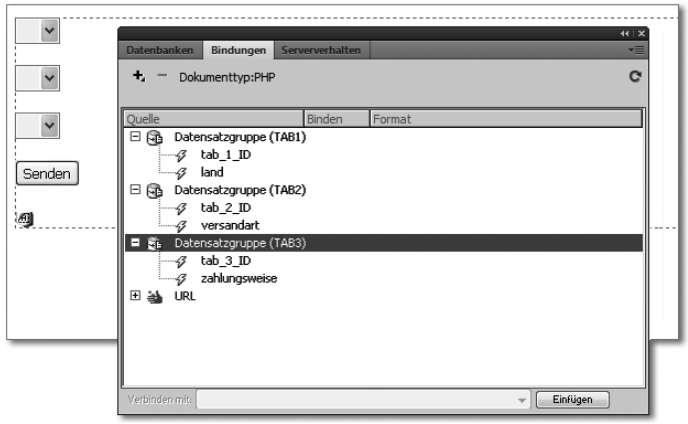


Abbildung 26.23 Datensatzgruppen anlegen

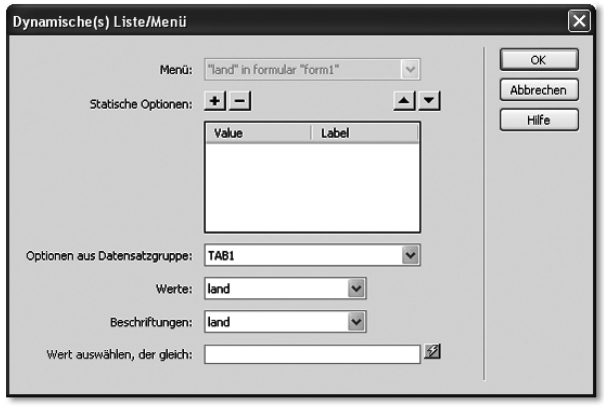


Abbildung 26.24 Formularfelder dynamisch anbinden

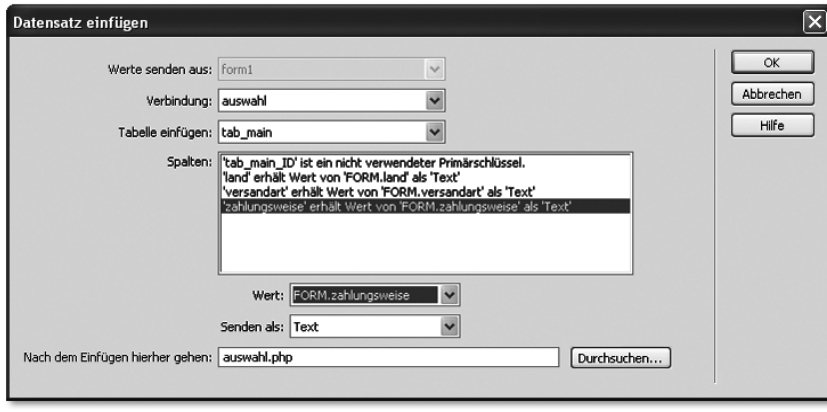


Abbildung 26.25 Formularfelder zuweisen

Werte in Haupttabelle eintragen

Wenn alles korrekt angelegt wurde, können Sie jetzt im Browser aus den Tabellen die gewünschten Werte auswählen und in die Tabelle eintragen lassen.



Abbildung 26.26 Browseransicht des Formulars

Betrachtet man die Haupttabelle, nachdem einige Kombinationen eingetragen wurden, stehen die ausgewählten Werte im Klartext in dieser Tabelle.

1	Deutschland	Luftfracht	Bankeinzug
2	England	Postversand	Visa
3	England	UPS	Scheck

Abbildung 26.27 Werte in der Haupttabelle

26.5 Administrationsoberfläche für die Buchwebsite anlegen



Nun wollen wir eine Administrationsoberfläche für die Website zum Buch anlegen. Wir bauen die Administrationsseiten mit einem Frameset auf. Auf der Buch-DVD finden Sie den Navigationsplan der Administrationsoberfläche als PDF-Datei im Ordner *Buchwebsite*.

Die Positionierungen in den XHTML-Dokumente für die Administration erstellen wir nicht mit CSS, weil der Aufwand hierfür zu hoch wäre. Für jedes einzelne Formularelement sowie für jede Unterseite müsste man eigene CSS-Regeln definieren, da sich die einzelnen Dokumente stark unterscheiden. Hier sind daher Tabellen und Frames das Mittel der Wahl. Eine Administrationsseite muss keinen Designwettbewerb gewinnen, sondern soll einfach und schnell erstellt werden und zuverlässig funktionieren. Hinzu kommt, dass Dreamweaver alle dynamischen Elemente von Haus aus mit Tabellen und nicht mit CSS anlegt. CSS verwenden wir allerdings für die Schriften und für die Optik der Formularelemente.

Es wird noch ein Login vorgeschaltet, um die Zugriffsrechte festzulegen. Nach jeder erfolgreichen Aktion wird das Dokument *okay.php* aufgerufen und angezeigt. Alternativ können Sie auch auf die Übersichtsseiten der einzelnen Bereiche weiterleiten. Im letzteren Fall sind dann gleich die Änderungen an der Datenstruktur sichtbar. Die Verwaltung der Zugriffsrechte legen wir in Kapitel 27, »Benutzer authentifizieren und Zugriffsrechte festlegen«, an. Zunächst erstellen wir jedoch die reine Administrationsebene.

[o] Wir sparen uns in den folgenden Abschnitten das genaue Erläutern aller Details und gehen davon aus, dass Sie mittlerweile genug Übung haben, die Schritte nachzuvollziehen. Alle benötigten Dateien finden Sie ebenfalls auf der DVD unter *Buchwebsite/Website_PHP/admin*.

26.5.1 Frameset anlegen

Das Frameset unterteilt das Browserfenster in eine Navigation (Top-Frame) den Kopf der Seite und die eigentliche anzuzeigende Seite im Haupt-Frame. Jeder Formularseite zur Änderung der Datensätze ist eine Auswahlseite mit der Liste aller Datensätze vorgeschaltet.

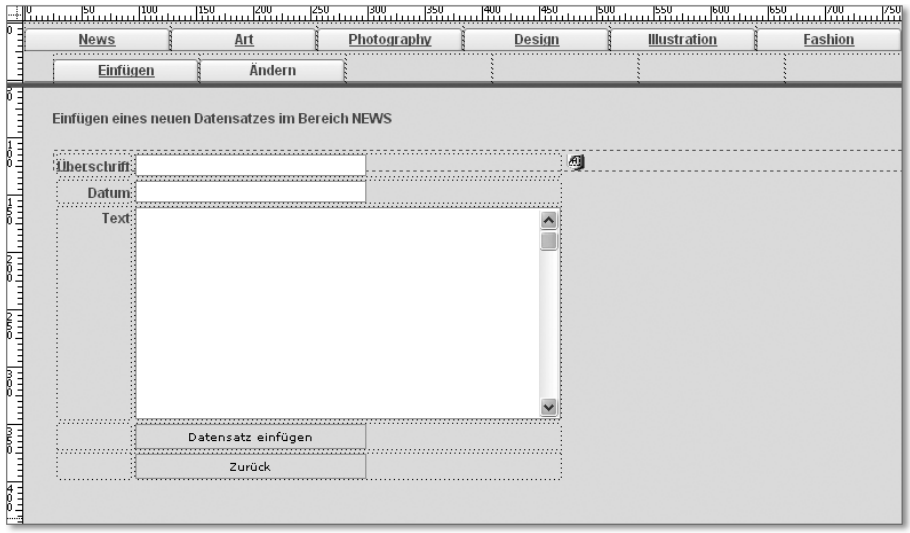


Abbildung 26.28 Die Layoutansicht der Administrationsoberfläche

26.5.2 Auswahlseiten

Die Auswahlseiten werden mit einer Tabelle und einem wiederholten Bereich aufgebaut. Neben der Auswahl der auf das Formulardokument verlinkten Überschrift wird ein Link zum Löschdokument eingefügt.

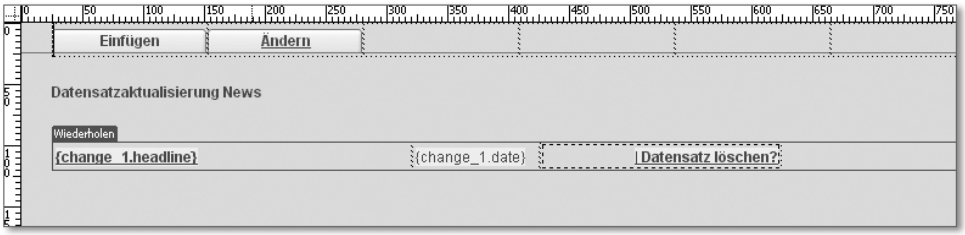


Abbildung 26.29 Datensatzauswahl

Die Headline und der Link zum Löschdokument übergeben die Datensatz-ID an die Folgedokumente. Das Funktionsprinzip ist das gleiche wie bei der eigentlichen Website.



Abbildung 26.30 Live-Ansicht der Datensatzauswahl

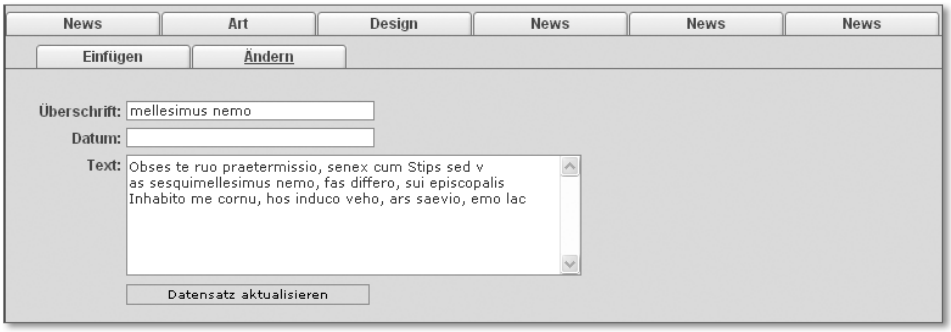


Abbildung 26.31 Live-Ansicht des Formulars

Wir legen zunächst die komplette Administrationsseite mit den Abfragen und Formularen an, um später die Detailarbeiten vorzunehmen.

26.5.3 CSS für Administrationsebene anlegen

Der Aufbau aller Seiten ist vom Prinzip her ähnlich. Daher können aus einer einmal erstellten Seite alle weiteren kopiert werden. Doch dazu gleich mehr.

Wir haben für die Administrationsebene eine etwas modifizierte CSS-Datei angelegt. Diese CSS-Datei haben wir *admin.css* benannt und binden sie nun als externen CSS-Stil in die Dokumente ein. Die CSS-Datei hat folgenden Inhalt:

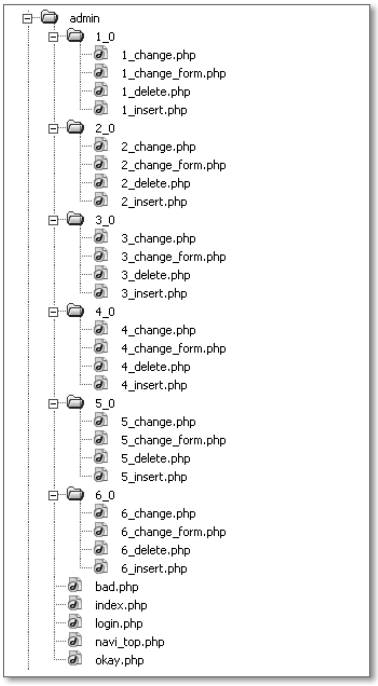
```
.headline {
   font-family: Arial, Helvetica, sans-serif;
   font-size: 12px;
   color: #75796F;
   line-height: 18px;
   font-weight: bold;
}
.content {
   font-family: Arial, Helvetica, sans-serif;
   font-size: 12px;
   color: #75796F;
   line-height: 18px;
}
#CONTENT A:link, #CONTENT A:hover,#CONTENT A:active,
#CONTENT A:visited {
   font-family: Arial, Helvetica, sans-serif;
   font-size: 12px;
   color: #75796F;
   line-height: 18px;
   font-weight: bold;
   text-decoration: none;
}
#EINGABE INPUT {
   font-family : Verdana, Geneva, Arial, Helvetica, sans-serif;
   font-size : 10px;
   border : 1px solid #ACBC3C;
   width: 200px;
}
#EINGABE TEXTAREA {
   font-family : Verdana, Geneva, Arial, Helvetica, sans-serif;
   font-size : 10px;
   border : 1px solid #ACBC3C;
   width: 370px;
}
```

Listing 26.1 CSS-Datei für die Administrationsebene

Schritt-für-Schritt: Aufbau der Administrationsebene

1 *Verzeichnisse anlegen*

Legen Sie sich am besten zuerst eine Verzeichnisstruktur für die neuen Teile der Website an (siehe folgende Abbildung), und erstellen Sie einen Satz Dokumente ohne Abfragen usw., aber mit den kompletten Verlinkungen.

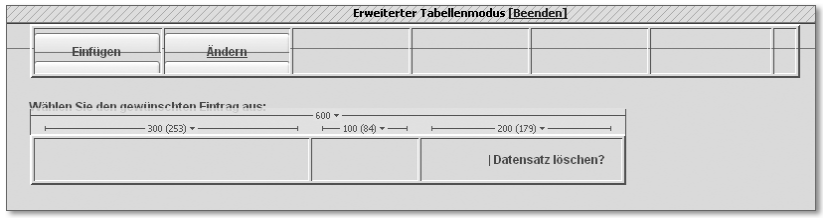


2 *Dokumente anlegen*

Wir haben zunächst die Dokumente *1_xxx* layoutet und diese dann außerhalb von Dreamweaver in jedes Verzeichnis kopiert. Dreamweaver unterstützt leider keine mehrfachen Kopien. Benennen Sie anschließend die Dateien im Dateifenster von Dreamweaver um, und lassen Sie die Verlinkungen von Dreamweaver aktualisieren.

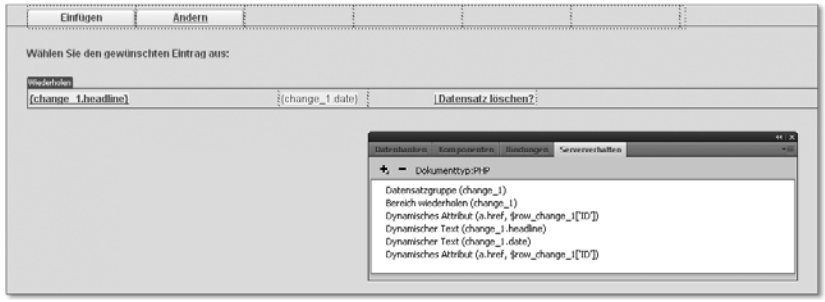
3 *Abfragen anlegen*

Im nächsten Schritt werden für alle Dokumente *X_change.php* die Abfragen und die wiederholten Bereiche angelegt. Die Headlines bei *News*, *Name* und *Vorname* bei den anderen Dokumenten werden mit den *X_change_form.php*-Dokumenten verlinkt. Der Schriftzug »Datensatz löschen« erhält eine Verlinkung auf die Dokumente *X_delete.php*.



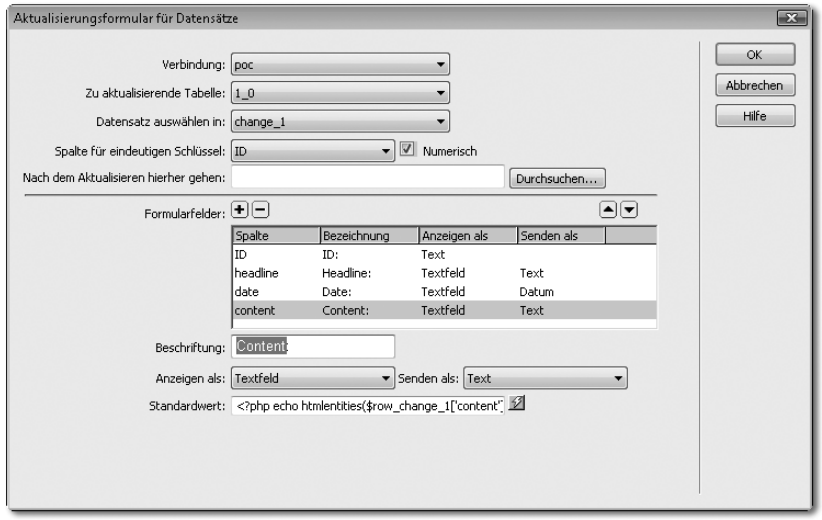
4 *URL-Parameter übergeben*

Übergeben Sie als URL-Parameter an die Formulardokumente die ID der Datensätze in der URL-Variablen `CONT_ID`, wie wir es auch bei der eigentlichen Website gemacht haben.



5 *Formulare anlegen*

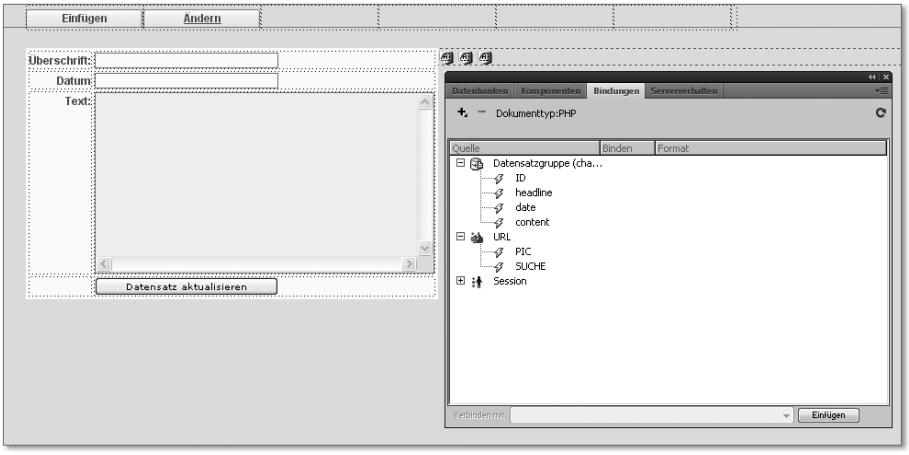
Legen Sie anschließend für jeden der Bereiche die Formulardokumente an. Die Datensätze müssen anhand des URL-Parameters `CONT_ID` ausgewählt werden.



6 *ID per URL übergeben*

Die ID aus den Datensätzen wird als verstecktes Feld ausgegeben, die Beschreibungen als Textfeld. Nach einer Aktualisierung der Daten werden die User auf das Dokument *okay.php* geleitet oder, wenn Sie es möchten, auch zurück zur Übersichtsseite.

Modifizieren Sie das Layout des Formulars entsprechend Ihren Wünschen, oder verwenden Sie die angegebene CSS-Datei, und weisen Sie der Tabelle für das Formular die ID EINGABE zu. Anschließend muss die Tabelle noch zentriert und der Textbereich auf 15 Zeilen vergrößert werden.



7 *News-Bereich anlegen*

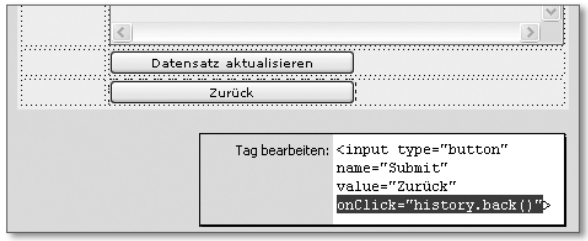
Bei dem Bereich *News* kümmern wir uns zunächst nicht um die Formatierung des Datums.

Wenn ein User einen Datensatz angewählt hat, kommt er bislang nur über die Browserleiste zur Auswahlseite zurück. Da dies jedoch bei dynamischen Websites immer zu Problemen führt, wird noch eine weitere Schaltfläche als ZURÜCK-Button eingefügt.

8 *»Zurück«-Button anlegen*

Ändern Sie die Beschriftung der Schaltfläche im Eigenschafteninspektor wie gewünscht, und wählen Sie als Aktion KEINE. Im Tag des Buttons fügen Sie dann das folgende kurze Skript hinzu:

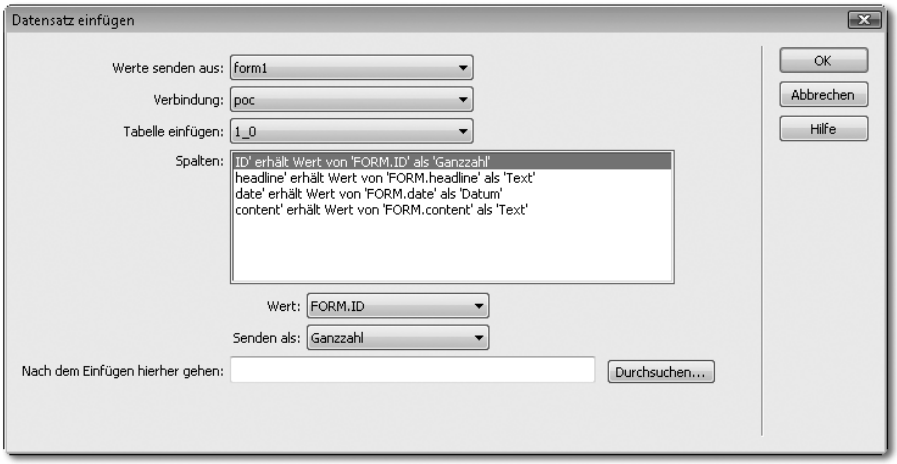
```
onClick="history.back()"
```



9 *Angabe des Bildpfads*

Die anderen Änderungsformulare sind ähnlich aufgebaut, enthalten lediglich Formularfelder. Ein tatsächlicher Bild-Upload ist mit diesem Formular nicht möglich. Bilder müssen mit FTP auf den Server geladen werden, und der Pfad zum Bild wird direkt in das Textfeld eingetragen. Anonyme Datei-Uploads sind auf Webservern selten erlaubt. Informationen über Skripte zum Datei-Upload finden Sie im Internet auf der PHP-Website oder auf anderen einschlägigen Websites.

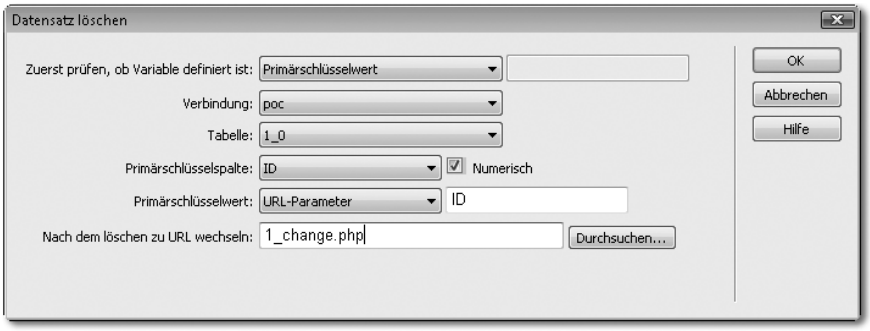
Für unser Projekt hat das direkte Eintragen der Bildpfade auch den Vorteil, dass Bilder auf anderen Servern als Ihren eigenen abgelegt werden können. Geben Sie in diesem Fall einen absoluten Pfad ein. Wenn Sie diese Technik auf einer Ihrer Websites verwenden möchten, stellen Sie sicher, dass einzubindende Bilder das vorgegebene Maß einhalten, damit Ihr Layout nicht gesprengt wird.



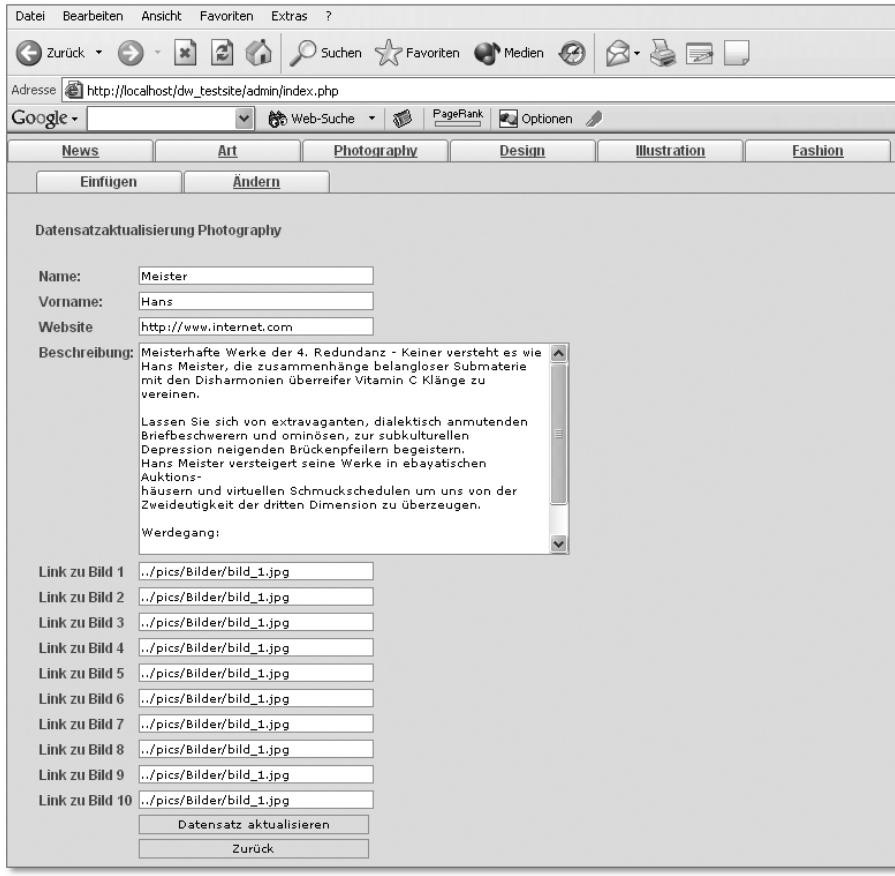
Verfahren Sie bei den noch fehlenden Dokumenten wie eben und in den entsprechenden Kapiteln beschrieben.

10 *Löschdokument anlegen*

Die Dokumente zum Löschen sind über die vorhin angelegte Auswahl erreichbar und enthalten außer dem Serververhalten DATENSATZ LÖSCHEN keine weiteren Aktionen.



Es ist etwas Fleißarbeit, aber wenn Sie diese erfolgreich abgeschlossen haben, haben Sie eine komplette und durchaus brauchbare Administrationsebene für die Website angelegt.


Datei Bearbeiten Ansicht Favoriten Extras ?
Zurück Suchen Favoriten Medien
Adresse http://localhost/dw_testsite/admin/index.php
Google Web-Suche PageRank Optionen
News Art Photography Design Illustration Fashion
Einfügen Ändern
Datensatzaktualisierung Photography
Name: Meister
Vorname: Hans
Website http://www.internet.com
Beschreibung: Meisterhafte Werke der 4. Redundanz - Keiner versteht es wie Hans Meister, die zusammenhänge belangloser Submaterie mit den Disharmonien überreifer Vitamin C Klänge zu vereinen.
Lassen Sie sich von extravaganten, dialektisch anmutenden Briefbeschwerern und ominösen, zur subkulturellen Depression neigenden Brückenpfeilern begeistern.
Hans Meister versteigert seine Werke in ebayatischen Auktions-
häusern und virtuellen Schmuckschedulen um uns von der Zweideutigkeit der dritten Dimension zu überzeugen.
Werdegang:
Link zu Bild 1 ../pics/Bilder/bild_1.jpg
Link zu Bild 2 ../pics/Bilder/bild_1.jpg
Link zu Bild 3 ../pics/Bilder/bild_1.jpg
Link zu Bild 4 ../pics/Bilder/bild_1.jpg
Link zu Bild 5 ../pics/Bilder/bild_1.jpg
Link zu Bild 6 ../pics/Bilder/bild_1.jpg
Link zu Bild 7 ../pics/Bilder/bild_1.jpg
Link zu Bild 8 ../pics/Bilder/bild_1.jpg
Link zu Bild 9 ../pics/Bilder/bild_1.jpg
Link zu Bild 10 ../pics/Bilder/bild_1.jpg
Datensatz aktualisieren
Zurück

Zugriffsrechte auf bestimmte Bereiche einer Website zu verwalten und festzulegen spielt beim Aufbau dynamischer Websites eine große Rolle. Mit Dreamweaver können Sie Benutzer und Zugriffsrechte problemlos anlegen.

27 Benutzer authentifizieren und Zugriffsrechte festlegen

Über die Benutzerverwaltung können Sie Zugriffsrechte auf Ihren Administrationsseiten verwalten oder Bereiche der Website für Besucher ohne Login-Daten sperren.

27.1 Benutzer verwalten

27.1.1 Schema einer Benutzerauthentifizierung

Zunächst erläutern wir das grundsätzliche Schema einer Authentifizierung. Wir haben Ihnen den Ablauf in Abbildung 27.1 grafisch dargestellt. Mit Hilfe eines Login-Dokuments werden die von einem User eingegebenen Daten mit der Tabelle in der Datenbank verglichen. Stimmen diese überein, wird der User zur Website weitergeleitet. Für den Fall des Misserfolgs, wie z. B. bei nicht vollständigen oder inkorrekten Daten, gelangt der User zurück zur Login-Seite.

In allen einzelnen Dokumenten der Site werden die Zugriffsrechte über in der Tabelle vorgegebene Begriffe gesteuert. Letzteres lässt sich an einem Beispiel besser erklären.

27.1.2 Tabelle für die Benutzerverwaltung

Um Benutzer zu verwalten, benötigen Sie zunächst eine Tabelle in der Datenbank mit den Benutzerdaten. Wir haben die Tabelle für die Benutzer wie in Abbildung 27.2 aufgebaut. Neben dem Primärschlüssel (`user_ID`) werden der Benutzername (`user_name`), das Passwort (`user_pass`), ein Zeitstempel (`user_time`) und ein Feld zur Definition der Zugriffsrechte (`user_perm`) angelegt.

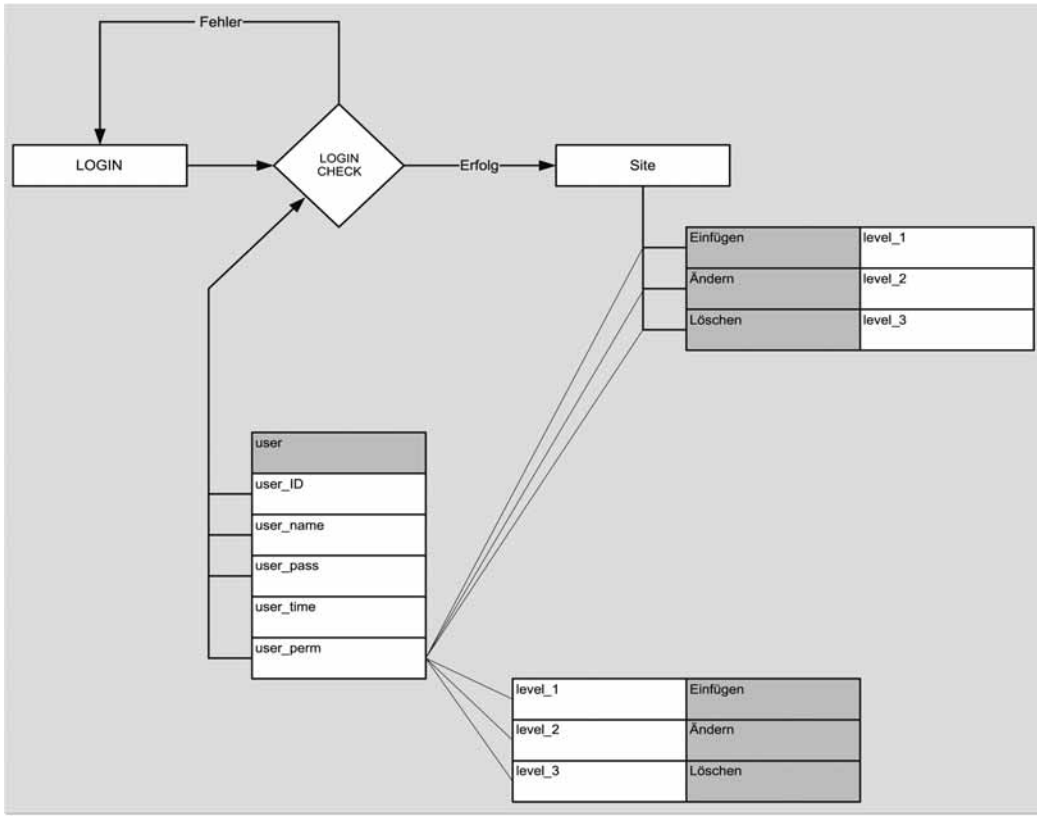


Abbildung 27.1 Schema einer Benutzerauthentifizierung

	Feld	Typ	Kollation	Attribute	Null	Standard	Extra	Aktion
☐	user_ID	int(11)			Nein	*keine*	auto_increment	
☐	user_name	varchar(100)	latin1_swedish_ci		Nein			
☐	user_pass	varchar(100)	latin1_swedish_ci		Nein	poc		
☐	user_time	timestamp		on update CURRENT_TIMESTAMP	Nein	CURRENT_TIMESTAMP	on update CURRENT_TIMESTAMP	
☐	user_perm	varchar(50)	latin1_swedish_ci		Nein	0		

Abbildung 27.2 Tabelle für die Benutzerverwaltung

Zeitstempel in der Benutzerverwaltung

Der Zeitstempel dient in unserem Fall dazu, nachträglich festzustellen, wann der User angelegt wurde. Wenn Sie über einen Zeitstempel die Zugehörigkeitsdauer eines Benutzers zu einer Community kennen, könnten darüber beispielsweise verschiedene Mailing-Aktionen gesteuert werden. Neue Mitglieder könnten andere Artikel angeboten bekommen als Dauergäste usw.

Passwörter verschlüsseln [!]

In unserem Beispiel stehen die Passwörter im Klartext in der Datenbank. In der Regel sollten Sie diese Praxis eher vermeiden, da sonst jeder, der Zugriff auf die Datenbank hat, z. B. auch alle anderen Zugangsdaten im Klartext lesen könnte.

PHP bietet verschiedene Verschlüsselungsmethoden an. Dazu werden die Daten nach der Eingabe in ein Formular verschlüsselt und in der Datenbank hinterlegt. Bei einem Login werden die Passwörter ebenfalls verschlüsselt und die beiden verschlüsselten Zeichenketten miteinander verglichen. Lesen Sie dazu in der PHP-Dokumentation zum Befehl `md5()` nach.

Zugriffsrechtesteuerung

Für die Zugriffsrechtesteuerung mit Dreamweaver benötigen Sie in der Datenbank ein weiteres Feld. In diesem Feld werden Benennungen für die Zugriffsrechte hinterlegt. Die Bezeichnungen dafür können Sie frei wählen. In unserem Beispiel benennen wir diese Rechte in `user_perm` mit:

- **level_1**
  Darf neue Datensätze einfügen.
- **level_2**
  Darf Datensätze ändern und neue einfügen.
- **level_3**
  Darf Datensätze ändern, neue einfügen und löschen.

27.1.3 Benutzer anmelden

Um eine Login-Seite zu erstellen, müssen Sie zunächst ein Formular mit den beiden Feldern für den Benutzernamen und das Passwort anlegen. In unserem Beispiel haben wir die Felder mit *user* und *pass* bezeichnet. Achten Sie bei diesen Dokumenten darauf, die Passwortfelder auch als Passwort zu definieren, damit die Eingaben nicht im Klartext, sondern durch Sternchen verschlüsselt angezeigt werden.

Klicken Sie, nachdem Ihr Formular fertig ist, in der Einfügeleiste unter DATEN über das Icon ❶ auf BENUTZER ANMELDEN (siehe Abbildung 27.3).

Sämtliche Parameter können in der nun folgenden Dialogbox eingestellt werden. Weisen Sie zunächst das soeben erstellte Formular ❷ (siehe Abbildung 27.4) zu. Wir haben dem Formular keinen bestimmten Namen gegeben, so dass die Vorgabe von Dreamweaver FORM1 eingetragen wird. Legen Sie die Felder für den Benutzernamen ❸ und das Passwort ❹ fest. Nach der Auswahl der korrekten Datenbank ❺ und der Tabelle ❻ erfolgt die Zuordnung dieser Felder zu den passenden Tabellenspalten ❼ und ❽.

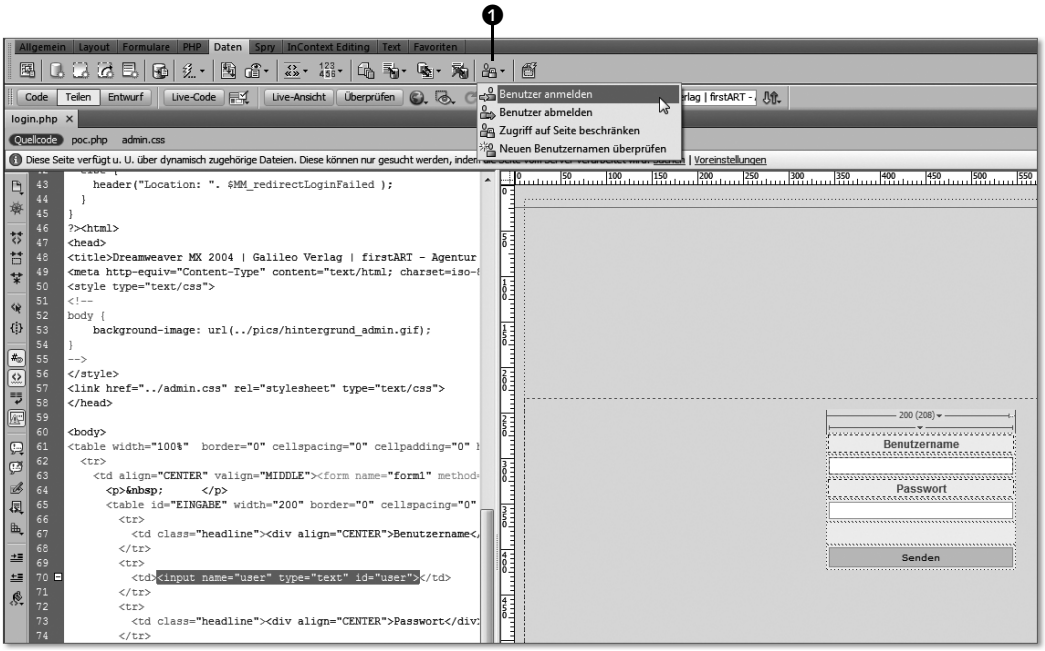


Abbildung 27.3 Formular anlegen

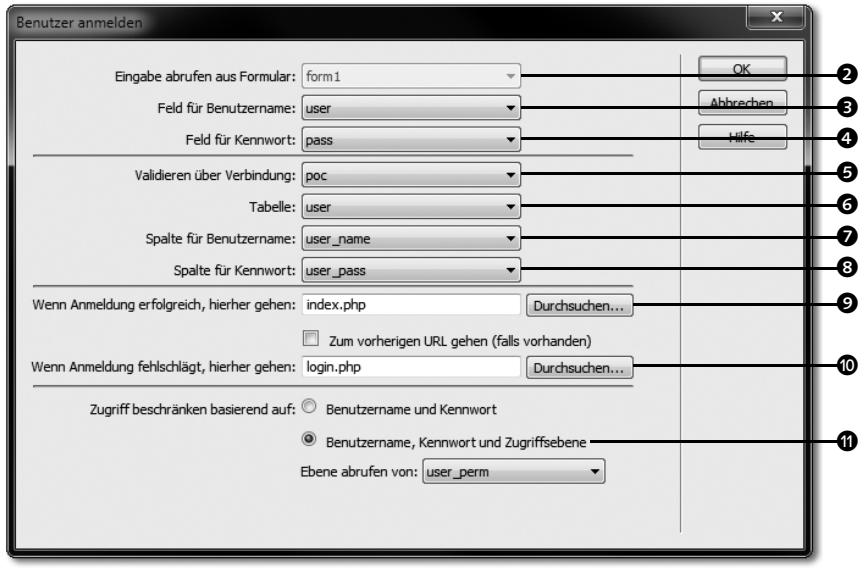


Abbildung 27.4 Dialogbox »Benutzer anmelden«

Weiterleitung nach dem Login

Nun sollten Sie das Dokument bestimmen, an das der User nach erfolgreicher Anmeldung weitergeleitet wird ❾. Im Falle einer fehlerhaften Anmeldung wird in unserem Beispiel der Benutzer wieder zum Login-Dokument geführt ❿. Somit ist ein erneuter Login-Versuch möglich, ohne zusätzlich durch die Website navigieren zu müssen.

Um weitere Zugriffsrechte für die Website festzulegen, müssen Sie angeben, auf welcher Basis die Rechte vergeben werden ⓫. Im einfachsten Fall sind das Benutzername und Passwort. Bei dieser Option steht Ihnen dann allerdings keine differenziertere Rechtevergabe zur Verfügung.

Zugriffsebene festlegen

Da wir in den folgenden Dokumenten nach Usern unterscheiden und verschiedene Möglichkeiten anbieten, haben wir zusätzlich eine Zugriffsebene ausgewählt. Deshalb müssen wir die Tabellenspalte angeben, in der die Ebene vermerkt ist. Diese Bezeichnungen sind bei Dreamweaver etwas verwirrend. Denn eigentlich ist »die Ebene« nichts anderes als ein beliebig wählbarer Begriff, der bei den Zugriffsrechten abgefragt und überprüft wird.

Im einfachen Fall – nämlich ohne weitere Rechteverwaltung – ist Ihre Zugriffsverwaltung nun auch schon fast fertig. Benutzer können nur bei erfolgreicher Authentifizierung auf die Folgeseiten zugreifen. Ruft allerdings ein User die Unterseiten direkt auf, hat er ungehinderten Zugriff. Damit dies verhindert wird, müssen für jede zu schützende Unterseite noch die Zugriffsrechte festgelegt werden.

27.2 Zugriffsrechte beschränken

Auf jeder zu schützenden Unterseite muss ein Skript dafür sorgen, dass die Seite nur dann angezeigt wird, wenn der User auch über die nötigen Berechtigungen verfügt. Fügen Sie dazu ein Serververhalten ZUGRIFF AUF SEITE BESCHRÄNKEN aus dem Bedienfeld SERVERVERHALTEN oder über die Einfügeleiste ein.

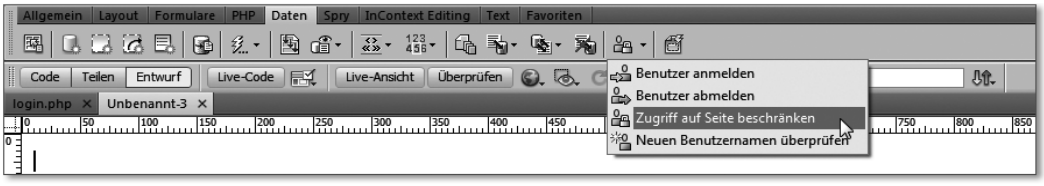


Abbildung 27.5 Zugriffsrechte festlegen

In der nun folgenden Dialogbox können Sie auswählen, nach welcher Methode die Zugriffe beschränkt werden sollen. Im einfachen Fall, Benutzername und Kennwort ❶, wird lediglich ein zuvor erfolgreiches Login vorausgesetzt. Wir möchten jedoch die Zugriffsrechte durch unsere Tabelle genauer steuern und wählen die Option Benutzername, Kennwort und Zugriffsebene ❷ in der Dialogbox aus.

In Abbildung 27.6 sind bereits einige Ebenen eingetragen. Sie können für ein Dokument auch mehrere Ebenen als zugriffsberechtigt angeben. Markieren Sie dazu alle gewünschten Zugriffsebenen mit gedrückter Strg/⌘-Taste.

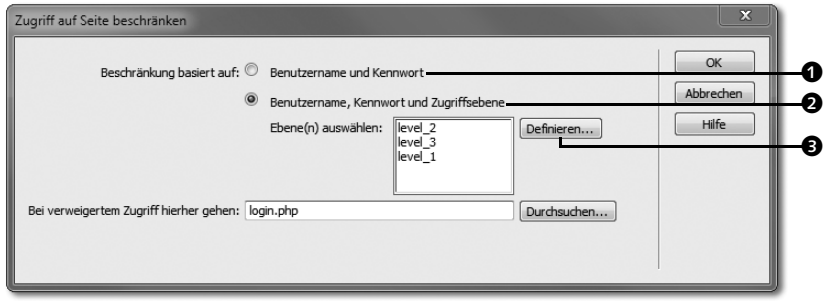


Abbildung 27.6 Dialogbox »Zugriff auf Seite beschränken«

Zugriffsebene definieren

Diese Eintragungen erfolgen nicht automatisch, sondern müssen von Hand angelegt werden. Klicken Sie dazu auf Definieren… ❸. In der Tabelle haben wir Begriffe für die verschiedenen Zugriffsrechte vergeben. Diese müssen nun in der Dialogbox Zugriffsebenen definieren ebenfalls eingetragen werden.



Abbildung 27.7 Dialogbox »Zugriffsebenen definieren«

Stimmen die beiden Begriffe überein, wird einem User der Zugriff auf die Seite gewährt. Wenn Sie diese Ebenen (Begriffe) in einem Dokument der Site angelegt haben, stehen Ihnen diese überall zur Verfügung und müssen nicht erneut definiert werden.

27.3 Neue Benutzer anlegen

Um einen neuen Benutzer anzulegen, reicht es manchmal aus, diesen mit phpMyAdmin in die MySQL-Datenbank einzutragen. In den meisten Fällen werden Sie jedoch ein Formular benötigen, damit sich User für ein bestimmtes Feature registrieren lassen können oder Ihr Kunde weitere Benutzer selbst anlegen kann.

Vermeiden Sie möglichst, Ihren Kunden direkten Zugriff auf die MySQL-Datenbank zu gewähren – außer, dies wird ausdrücklich gewünscht. Die Risiken, bei einem Direktzugriff mit einer fehlerhaften Eingabe die gesamte Datenstruktur zu zerstören oder Tabellen zu löschen, sind zu hoch.

Schritt-für-Schritt: Ein Anmeldeformular für neue Benutzer erstellen [▪▪]

Damit ein Direktzugriff auch bei der Benutzerregistrierung nicht notwendig ist, erstellen wir ein Anmeldeformular, das dafür sorgt, dass der neue Benutzer in der Datenbank gespeichert wird.

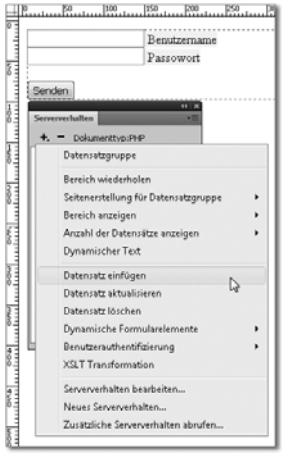
1 *Formular anlegen*

Das Verfahren zum Anlegen eines neuen Benutzers unterscheidet sich zunächst nicht von dem Erstellen eines ganz normalen Einfügeformulars.

Wir haben im Formular den beiden Textfeldern die Namen *user* und *pass* zugewiesen.

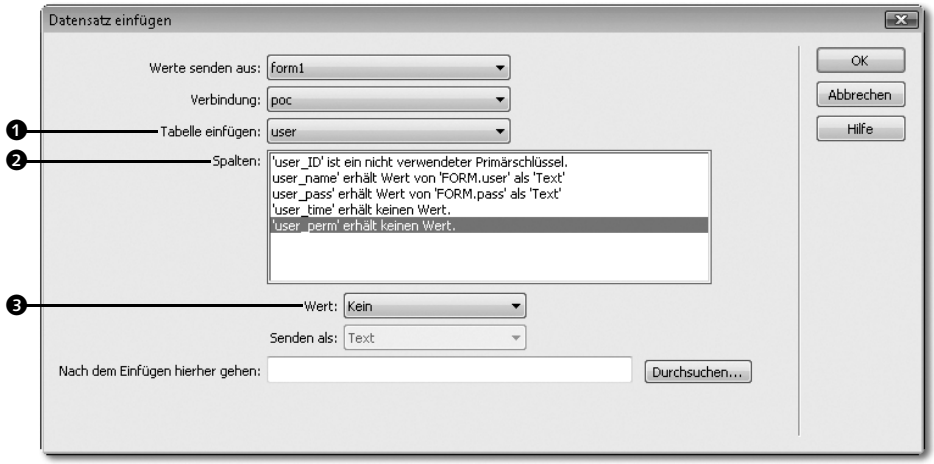
2 *Datensatz einfügen*

Wählen Sie im Anschluss daran aus dem Bedienfeld Serververhalten die Aktion Datensatz einfügen aus.



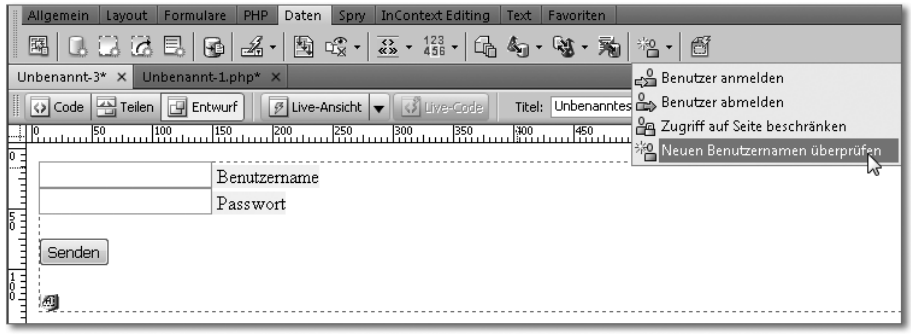
3 *Angeben der Einfügeparameter*

In der nun folgenden Dialogbox DATENSATZ EINFÜGEN muss nach der Auswahl der Tabelle ❶ festgelegt werden, in welcher Spalte der Tabelle ❷ der Inhalt aus den Formularfeldern ❸ eingetragen wird.



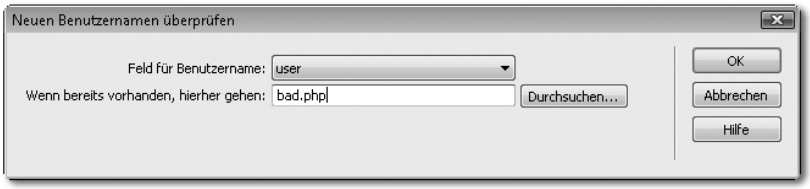
4 *Neuen Benutzer anlegen*

Danach müssen Sie nur noch aus der Einfügeleiste NEUEN BENUTZERNAMEN ÜBERPRÜFEN auswählen. Dreamweaver fügt Ihrem Dokument eine Abfrage hinzu, bei der überprüft wird, ob der Benutzer bereits existiert oder nicht. So vermeiden Sie doppelte Einträge.



5 *Weiterleitung bei bereits vorhandenen Benutzern*

In der Dialogbox NEUEN BENUTZERNAMEN ÜBERPRÜFEN können Sie ein Dokument angeben, an das der Benutzer weitergeleitet wird, wenn der gewünschte Benutzername bereits vergeben wurde. Dieses Dokument kann natürlich auch wieder das Dokument zum Eintragen der Benutzerdaten sein.



Neuen Benutzer mit Auswahl der Zugriffsebene anlegen

Wenn auf der Website verschiedene Zugriffsebenen vergeben werden sollen, müssen beim Anlegen eines neuen Benutzers auch die Berechtigungen vergeben werden. Im einfachsten Fall legen Sie ein weiteres Formularfeld an und tragen den Namen der Ebene von Hand ein. Diese Vorgehensweise ist jedoch nicht sehr geschickt, und im Laufe der Zeit werden garantiert falsche Eintragungen vorkommen.

Besser ist es, die Zugriffsebenen in einer separaten Tabelle zu speichern und die Rechtezuweisungen aus dieser Tabelle zu generieren.

In Abbildung 27.8 sehen Sie ein grafisches Schema des zu erstellenden Skripts. In der Tabelle *perm* werden die Namen der Zugriffsebenen abgelegt. Im Formular muss zusätzlich zu den Textfeldern *user* und *pass* eine Liste/ein Menü eingefügt werden. Diese Liste/dieses Menü erhält ihre/seine Inhalte aus der Tabelle *perm*.

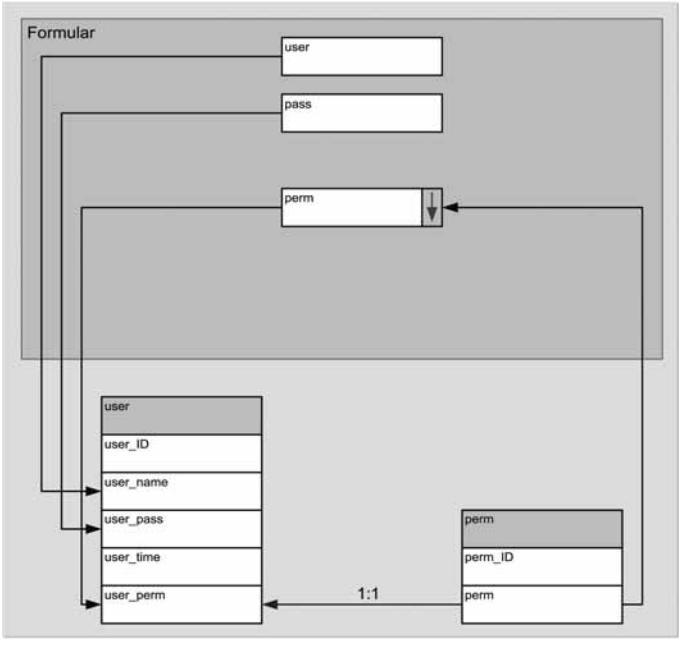


Abbildung 27.8 Schema der Rechtevergabe

Damit steht Ihnen eine Auswahl bestehender Zugriffsrechte zur Verfügung, und diese können beim Absenden des Formulars direkt in die Tabelle *user* eingetragen werden.

In der Tabelle für die Zugriffrechte werden nur zwei Felder benötigt. Selbst die Vergabe eines Primärschlüssels ist nicht zwingend notwendig, bereitet die Tabelle jedoch auf eventuelle spätere Aktionen und Erweiterungen der Rechteverwaltung vor.

perm_ID	int(11)		Nein		auto_increment
perm	varchar(100)		Nein		

Abbildung 27.9 Tabelle »perm«

In Dreamweaver wird eine Abfrage der Tabelle *perm* eingefügt.

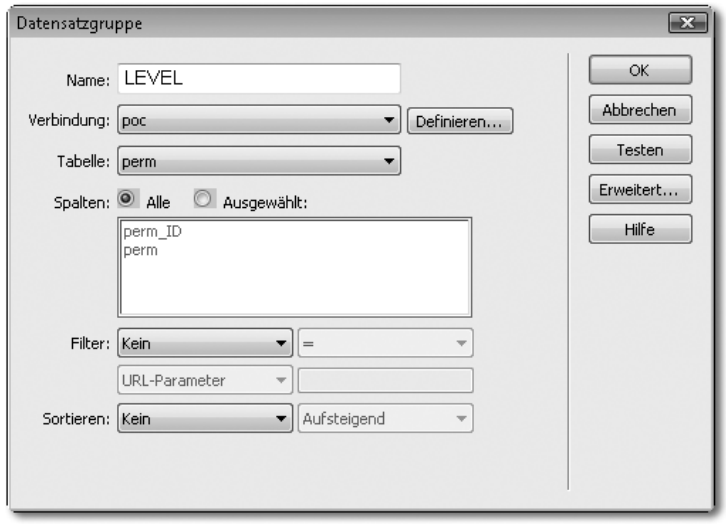


Abbildung 27.10 Abfrage der Tabelle »perm«

Die Liste/das Menü wird dynamisch mit der Tabelle *perm* verbunden. Die angezeigten Werte und die Inhalte der zu übertragenden Variablen sind in unserem Fall identisch. In die Tabelle *perm* könnte auch ein drittes Feld für eine Beschreibung der Zugriffsrechte eingefügt werden. Dann können Sie sich die Beschreibung in der Liste/im Menü anzeigen lassen und den Namen der Zugriffsebene als Variable einbinden.

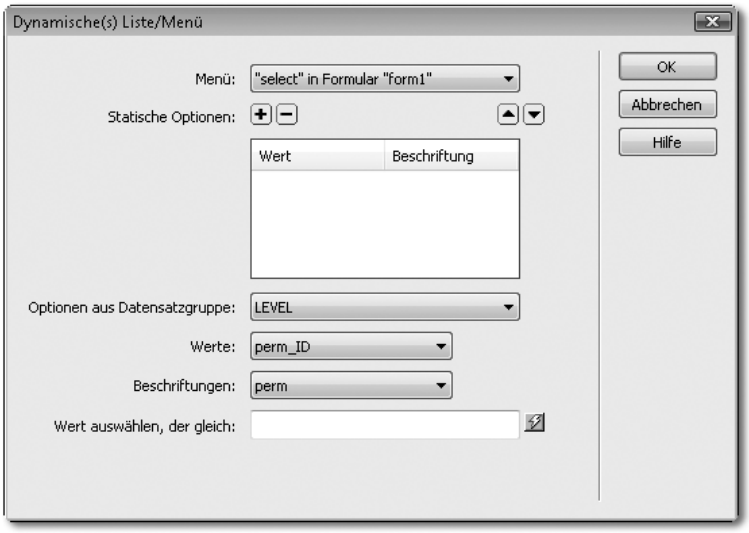


Abbildung 27.11 Verbinden der Liste/des Menüs mit der Tabelle »perm«

Wie auch im vorherigen Abschnitt müssen Sie die Formularfelder beim Serververhalten DATENSATZ EINFÜGEN den Tabellenspalten zuweisen. Dabei wird der Inhalt aus der Spalte *perm* in der Tabelle *perm* in die Spalte *user_perm* der Tabelle *user* geschrieben.

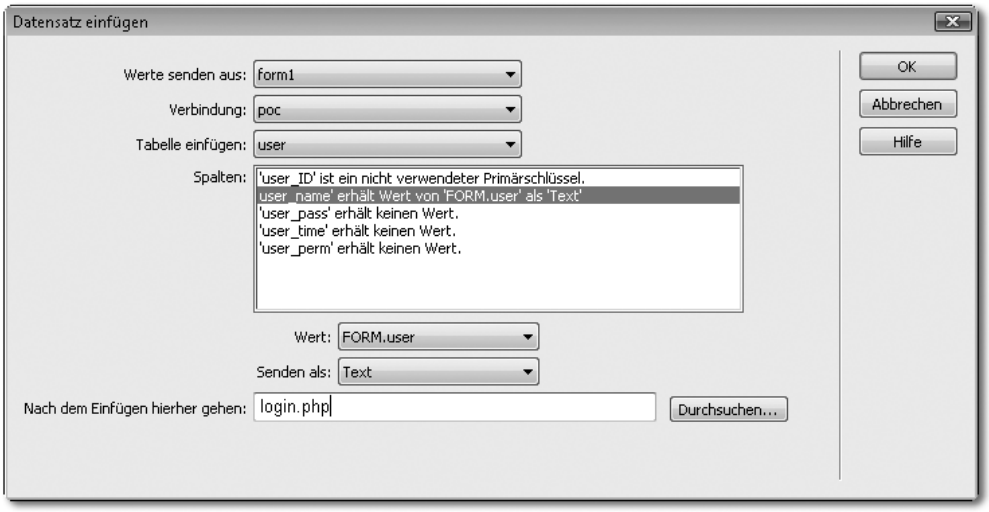


Abbildung 27.12 Zuweisen der Formularfelder

Zum Abschluss fehlt noch die Überprüfung des neuen Benutzers, und das Anmeldeformular ist komplett.

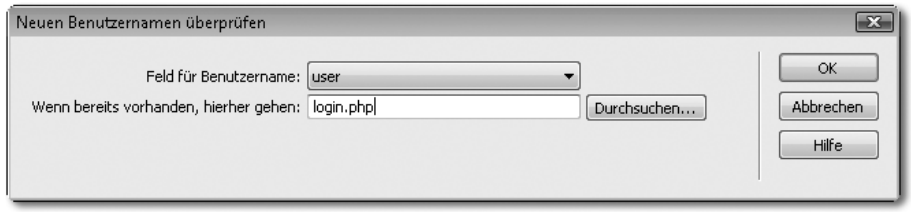


Abbildung 27.13 Neuen Benutzer überprüfen und einfügen

27.4 Zugriffsrechte für die Website zum Buch


Buchwebsite

Die erstellte Website soll nun durch verschiedene Zugriffsrechte geschützt werden. Die Benutzer und die dazugehörigen Daten sind in der Tabelle *user* hinterlegt. Wir haben diese Tabelle bereits in einem der vorherigen Kapitel verwendet.

[▪] *Schritt-für-Schritt: Eine Datentabelle schützen*

1 *Tabelle »user«*

Die Zugriffsebenen werden wie folgt verwendet:

- **level_1**
  Zugriff auf das Einfügen neuer Datensätze. Der Benutzer darf keine Datensätze löschen oder verändern.
- **level_2**
  Berechtigung, Datensätze neu anzulegen und zu verändern.
- **level_3**
  Alle Aktionen sind erlaubt.

Die verschiedenen *Zugriffsebenen* dienen der Veranschaulichung des Themas. In der Praxis müsste man die Zugriffsprivilegien eventuell noch detaillierter ausbauen. Es wäre denkbar, den Datensätzen ein weiteres Feld für eine Freigabe der Datensätze hinzuzufügen. So könnte man zwischen Usern, die einen Datensatz anlegen, und Administratoren, die diesen Datensatz für die eigentliche Website freigeben, unterscheiden.

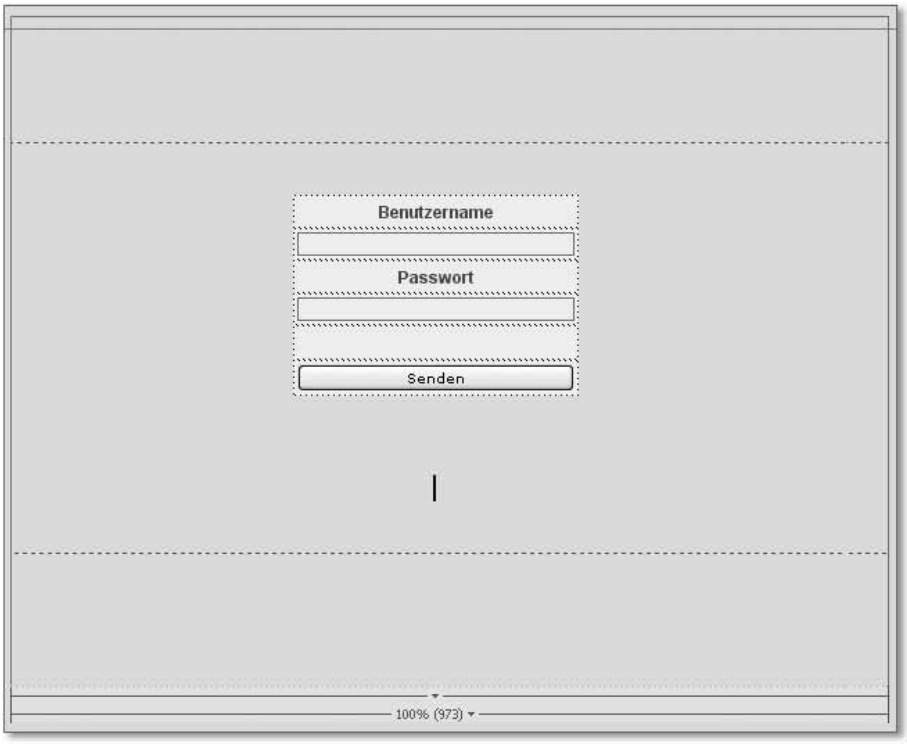
Wir haben in der Tabelle *user* drei *Benutzer* angelegt. Die Benutzer *user_1* bis *user_3* haben jeweils andere Zugriffsprivilegien, und Sie können die Zugriffsrechte der Website mit diesen Benutzerkonten testen.

user_ID	int(11)		Nein		auto_increment
user_name	varchar(100)		Nein		
user_pass	varchar(100)		Nein	poc	
user_time	timestamp(14)		Ja	*NULL*	
user_perm	varchar(50)		Nein	0	

user_ID	user_name	user_pass	user_time	user_perm
1	user_1	user_1	20031223191946	level_1
4	user_2	user_2	20031223191958	level_2
5	user_3	user_3	20031223192018	level_3

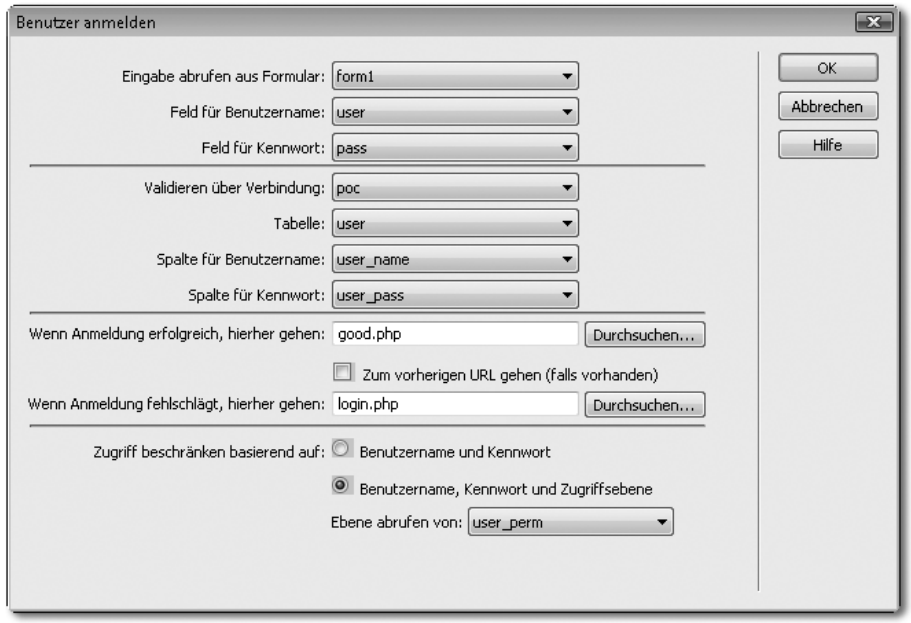
2 *Anmeldeseite anlegen*

Zunächst legen wir ein Dokument mit dem Formular für die Benutzeranmeldung an.



3 *Einstellungen der Anmeldeseite*

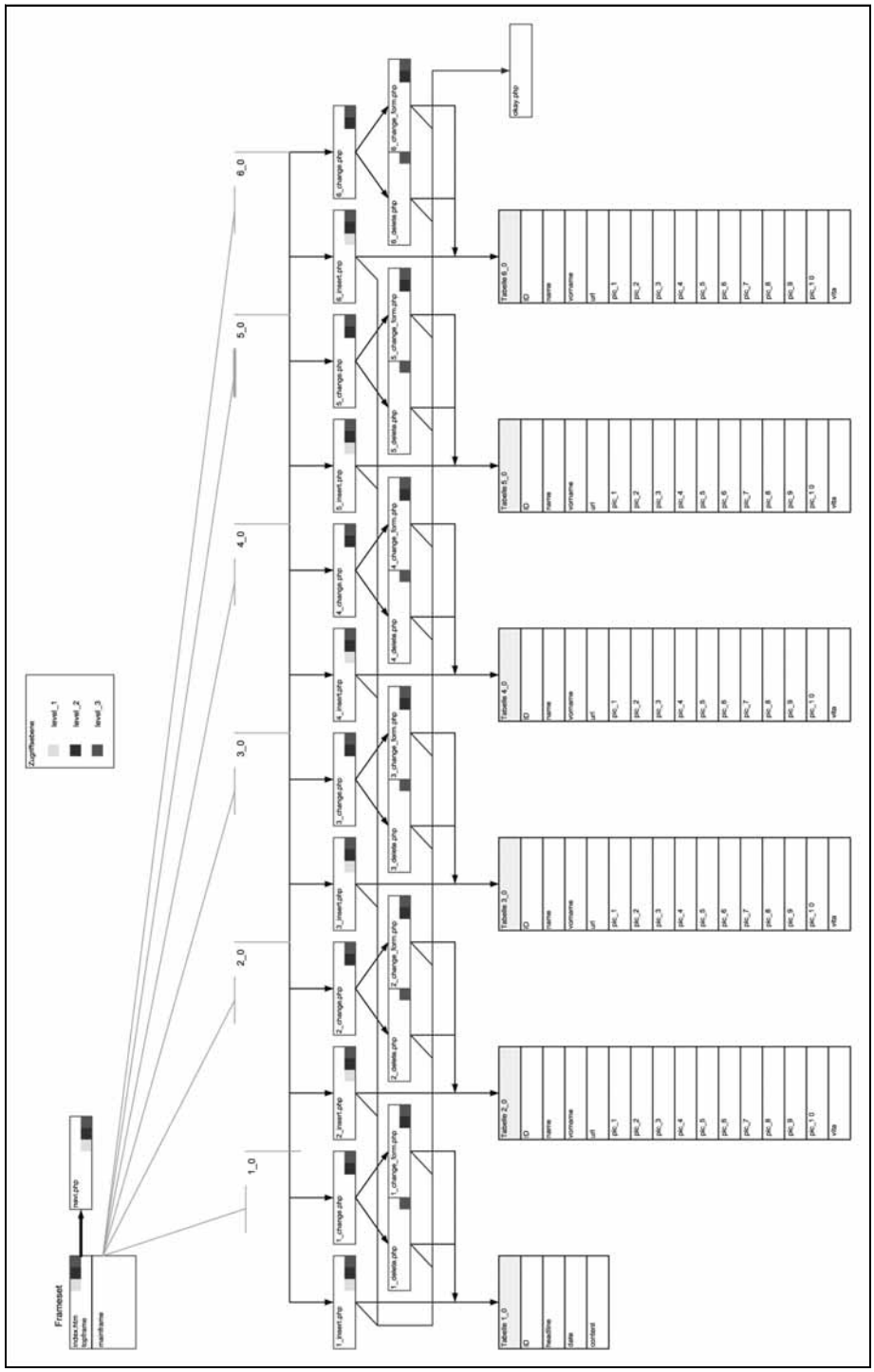
In der folgenden Abbildung sehen Sie die Einstellungen für die Anmeldeseite. Bei einem fehlerhaften Login wird die Anmeldeseite erneut aufgerufen.



Bei unserem Verfahren dürfen Sie in keinem Fall bei einer der Unterseiten die Berechtigungsabfragen vergessen. Ansonsten hätte ein Besucher, der das Dokument ohne Zugriffsüberprüfung direkt aufruft, einen Zugriff auf Teile der Administrationsebene. Überprüfen Sie also vor der Freischaltung einer Website die Zugriffsrechte sehr genau.

4 *Zugriffsrechte vergeben*

In der folgenden Abbildung haben wir die Zugriffsebenen in den Navigationsplan eingetragen. Es ist sinnvoll, diese Rechtevergabe zu dokumentieren, damit auch in Zukunft nachvollziehbar ist, wie die Rechte verteilt wurden. Nun fehlt nur noch, dass Sie für jede der einzelnen Seiten die Zugriffsrechte gemäß dem Navigationsplan vergeben, und die Website zum Buch ist fertig.


Frameset
menu.php
1_0
2_0
3_0
4_0
5_0
6_0
Zugriffsebene
level_1
level_2
level_3
okay.php

Mit zunehmender Erfahrung werden Sie an die Grenzen von Dreamweaver stoßen und eigene Abfragen oder Skripte einbinden wollen. Dreamweaver unterstützt Sie auch dabei ganz hervorragend.

28 Fortgeschrittene Techniken

Nicht alles kann mit Dreamweaver ohne Weiteres im Layout und mit einfachen Mitteln umgesetzt werden. Mit ein wenig Zusatzwissen kann man jedoch auch komplexere Arbeiten erledigen.

28.1 SQL-Abfragen

28.1.1 Eigene SQL-Abfragen einbinden

Häufig werden Sie mit PHP mehrere Tabellen gleichzeitig abfragen wollen. Im erweiterten Modus des Datensatzabfrage-Fensters können Sie SQL-Abfragen mit Hilfe von Dreamweaver erstellen oder auch eigene anlegen.

Datenbankschema

In unserem Beispiel sollen die Bestellvorgänge einzelner Kunden angezeigt werden. Unsere Datenbank enthält in einer der Tabellen die Kundendaten und in einer weiteren die Bestellnummern eines jeden Bestellvorgangs. In einer dritten Tabelle werden die Bestellnummern dann den Kunden zugeordnet. Die im Schema von Abbildung 28.1 grau angezeigte Tabelle soll mögliche weitere Detailverknüpfungen darstellen. Diese graue Tabelle ist in unserer Beispieldatenbank allerdings nicht vorhanden.

Im späteren Dokument soll eine Liste ausgegeben werden, in der pro Bestellung eine Zeile mit Kunde und Bestellnummer angezeigt wird. Bestandskunden, die keine Bestellungen getätigt haben, werden nicht angezeigt, da es sich bei unserer Abfrage um einen *inner join* handelt.

- Mayer – 2003-10010
- Mayer – 2003-10011
- Mayer – 2003-10012

- Müller – 2003-10020
- Koch – 2003-10030
- Koch – 2003-10031
- Koch – 2003-10032
- Koch – 2003-10036

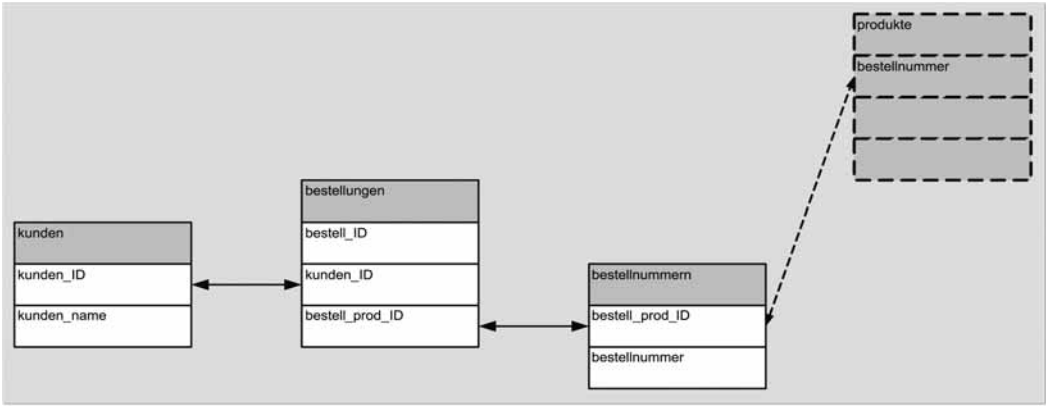


Abbildung 28.1 Datenbankschema

[o] In der Praxis könnte es durchaus eine ähnliche Anforderung geben. Die abzufragenden Tabellen sind dann meistens umfangreicher. Um das Prinzip zu verdeutlichen, reicht unser Beispiel aus, und es verwirrt nicht durch zusätzliche Datenfelder. Sie finden die Datenbank auf der DVD unter *Beispiele/SQL-Abfragen/BESTELLUNGEN.sql*. Alternativ können Sie natürlich auch eine neue Datenbank anlegen und die folgenden Tabellen von Hand konfigurieren.

[»] **Komplexere Abfragen**

Wir arbeiten in unserem Beispiel nur mit einer einfachen Variante eines *inner join*. Wenn Sie flexibler arbeiten möchten und komplexere Abfragen benötigen, empfehlen wir Ihnen, sich mit MySQL und `left outer join`, `right outer join` usw. zu beschäftigen.

Aus den Abbildungen 28.2 bis 28.4 können Sie die Struktur der Tabellen mit den einzelnen Datenfeldern entnehmen. Die Primärschlüssel der Tabellen werden in den Feldern *kunden_ID*, *bestell_prod_ID* und *bestell_ID* gesetzt.

	Feld	Typ	Kollation	Attribute	Null	Standard	Extra	Aktion
☐	kunden_ID	bigint(20)			Nein	*Kein*	auto_increment	
☐	kunden_name	varchar(100)	latin1_swedish_ci		Nein	*Kein*		

Abbildung 28.2 Struktur der Kundentabelle

	Feld	Typ	Kollation	Attribute	Null	Standard	Extra	Aktion
☐	bestell_prod_ID	bigint(20)			Nein	*Kein*	auto_increment	
☐	bestellnummern	varchar(100)	latin1_swedish_ci		Nein	*Kein*		

Abbildung 28.3 Struktur der Bestellnummerntabelle

	Feld	Typ	Kollation	Attribute	Null	Standard	Extra	Aktion
☐	bestell_ID	bigint(20)			Nein	*Kein*	auto_increment	
☐	kunden_ID	bigint(20)			Nein	*Kein*		
☐	bestell_prod_ID	bigint(20)			Nein	*Kein*		

Abbildung 28.4 Struktur der Zuordnungstabelle

	kunden_ID	kunden_name
☐	1	Mayer
☐	2	Koch
☐	3	Jansen

Abbildung 28.5 Daten der Kundentabelle

	bestell_prod_ID	bestellnummern
☐	1	2010-10010
☐	2	2010-10011
☐	3	2010-10012
☐	4	2010-10013
☐	5	2010-10015
☐	6	2010-10020
☐	7	2010-10045
☐	8	2010-10056

Abbildung 28.6 Daten der Bestellnummerntabelle

In der Zuordnungstabelle befinden sich die Primärschlüssel der beiden anderen Tabellen als Zuordnungskriterium. Datensätze aus Primärschlüsseln anderer Tabellen bezeichnet man auch als *Fremdschlüssel*.

	bestell_ID ▲	kunden_ID	bestell_prod_ID
☐	1	1	2
☐	2	2	1
☐	3	3	4
☐	4	2	5
☐	5	1	8
☐	6	2	7
☐	7	2	3

Abbildung 28.7 Daten der Zuordnungstabelle

Es folgt eine kurze Übung dazu, wie Sie eigene SQL-Abfragen in Dreamweaver einbinden können. Wir empfehlen Ihnen, diese Anleitung zu lesen, auch wenn Sie die Schritte nicht nachvollziehen, da einige Grundlagen erläutert werden.

[▪] *Schritt-für-Schritt: SQL-Abfragen einbinden*

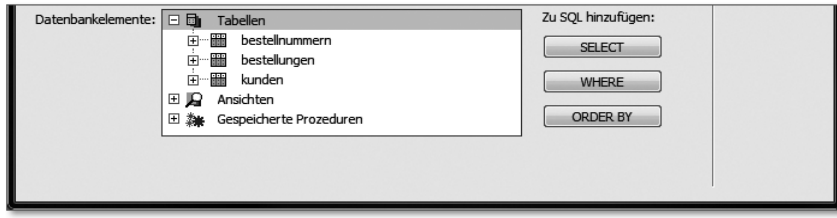
1 *Datenbankanbindung erstellen*

Erstellen Sie zunächst eine Datenbankanbindung für die neue Datenbank.



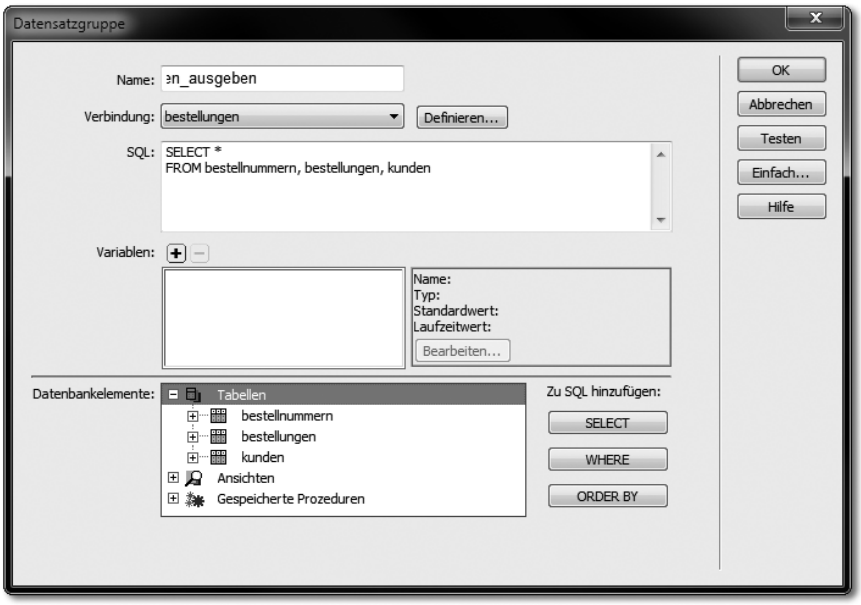
2 *Anlegen einer »SELECT«-Abfrage*

Die benötigte Abfrage muss im erweiterten Modus der Dialogbox Datensatzgruppe erfolgen. `SELECT`-Befehle können aus mehr als einer Tabelle Daten ausgeben. Markieren Sie dazu die erste gewünschte Tabelle im Abschnitt Datenbankelemente, und klicken Sie auf SELECT. Die Tabelle wird nun mit einer `SELECT`-Anweisung ins SQL-Fenster geschrieben.



3 *Mehrere Tabellen in einer Abfrage*

Wenn Sie anschließend die nächste Tabelle markieren und erneut auf SELECT klicken, wird die Tabelle der bestehenden `SELECT`-Anweisung hinzugefügt. Fügen Sie so alle vorhandenen Tabellen ein.



4 *Datensatz ansprechen*

Die Kriterien für unsere Abfrage sind die *kunden_ID* in den Tabellen *kunden* und *bestellungen* und die *bestell_prod_ID* aus der Tabelle *bestellungen*, außerdem die Tabelle *bestellnummern*.

Ein Datensatz einer bestimmten Tabelle wird in SQL wie folgt angesprochen:

```
Tabelle.Datensatz
```

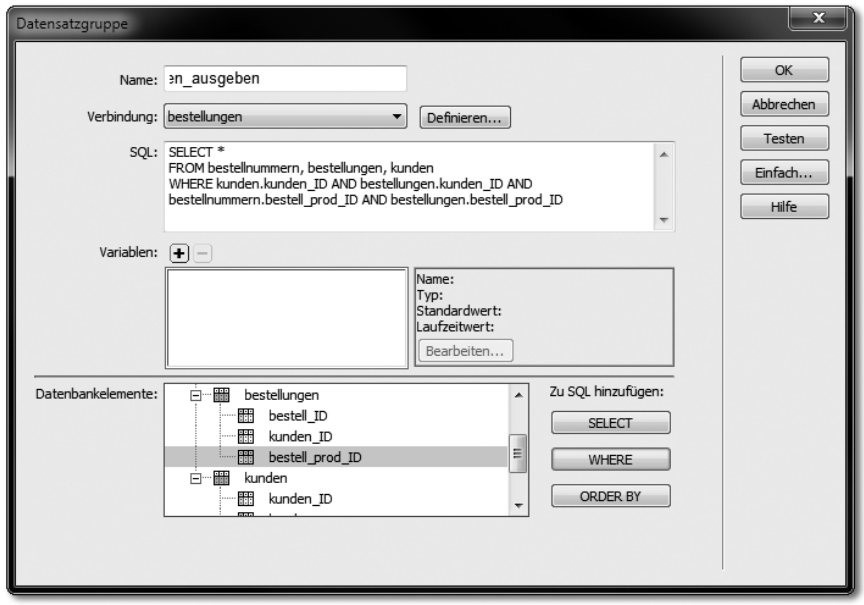
Daraus ergibt sich für die Zuordnung der Daten folgende Schreibweise:

```
bestellnummern.bestell_prod_ID = bestellungen.bestell_prod_ID
AND
bestellungen.kunden_ID = kunden.kunden_ID
```

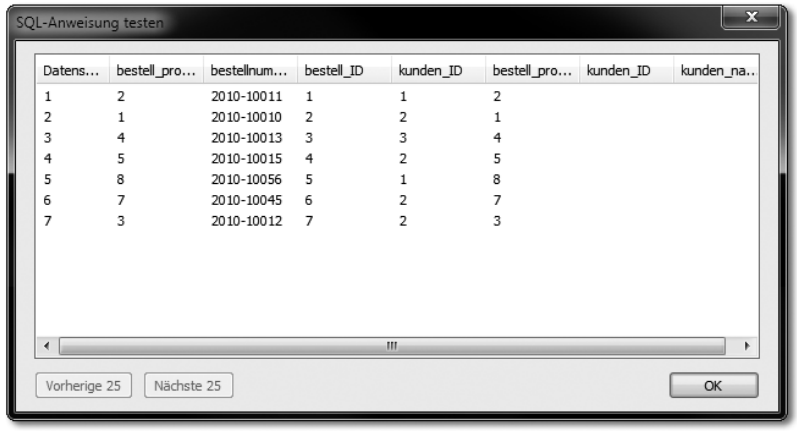
Um diese Abfrage zu schreiben, wählen Sie einfach der Reihe nach alle abzufragenden Datensätze im Abschnitt DATENBANKELEMENTE aus und klicken nach jeder Auswahl auf WHERE.

Dreamweaver schreibt als Standard für die Verknüpfungen immer die AND-Bedingung. Das ist für unsere Abfrage nicht korrekt. Ändern Sie daher im SQL-Fenster die Abfrage wie in der folgenden Abbildung angeführt.

Wenn Sie alles korrekt erledigt haben, sieht Ihre fertige Datensatzabfrage jetzt wie in der folgenden Abbildung aus.



Die Ergebnisse der Abfrage können Sie durch einen Klick auf TESTEN überprüfen. Besonders bei komplexen Abfragen empfiehlt es sich, diese Tests vor dem Einbinden der Daten in das Dokument durchzuführen.

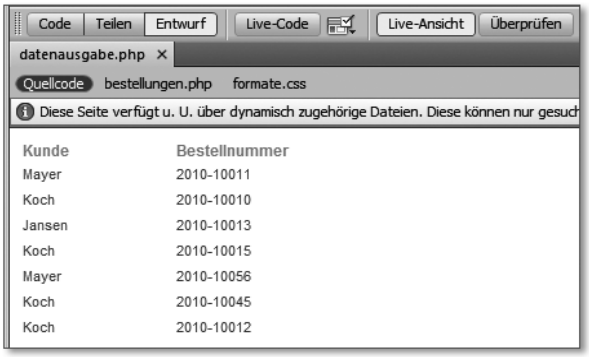


5 *Datensätze anzeigen*

Um alle relevanten Datensätze anzuzeigen, müssen Sie die gewünschten Datensätze wie gewohnt in Ihr Dokument ziehen und einen wiederholten Bereich daraus erstellen.



Hinter den Datensätzen im wiederholten Bereich haben wir noch einen Zeilenumbruch eingefügt, damit nicht alles in einer Zeile steht, und die einzelnen Felder mit einem Bindestrich getrennt.



6 *Ausgabe der Abfrage*

In der Live-Ansicht sehen Sie die korrekte Ausgabe der Daten. Diese stehen Ihnen für weitere Manipulationen zur Verfügung. ■

28.1.2 Externe Tools verwenden

Die mit Dreamweaver-Bordmitteln erstellten SQL-Abfragen sind wirklich sehr einfach gehalten und erfüllen selten gehobene Ansprüche. Wesentlich komfortabler geht es mit externen Tools wie EMS SQL Query, MySQL Workbench und einigen anderen. Leider sind die wirklich guten Programme nicht kostenfrei.

Abfragen, wie zum Beispiel aus EMS SQL Query, können oft visuell erstellt werden und generieren deutlich besseren und komplexeren Code als Dreamweaver.

```
SELECT
  `kunden`.kunden_name,
  `bestellnummern`.bestellnummern
FROM
  `bestellnummern`
  INNER JOIN `bestellungen`
ON (`bestellnummern`.bestell_prod_ID = `bestellungen`.bestell_ID)
  INNER JOIN `kunden`
ON (`bestellungen`.kunden_ID = `kunden`.kunden_ID)
```

Listing 28.1 SQL-Abfrage aus EMS SQL Query

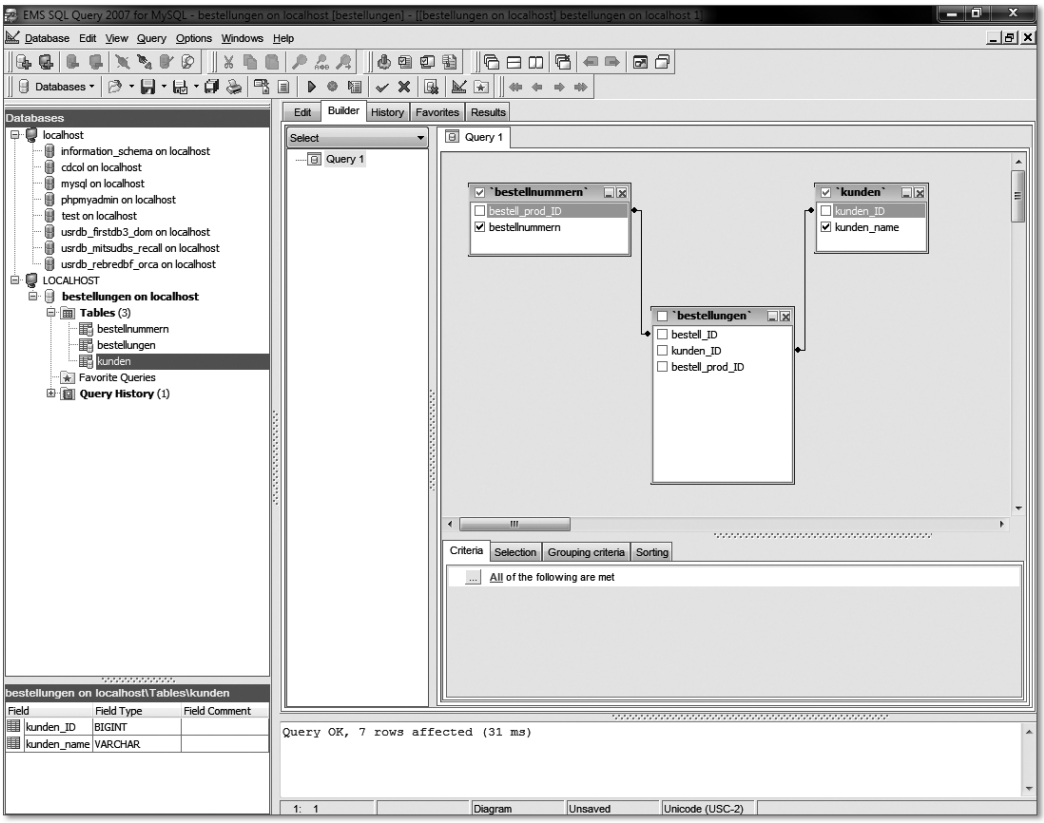


Abbildung 28.8 EMS SQL Query 2007 for MySQl

Die Abfragen können ohne Probleme mit Dreamweaver verwendet werden. Kopieren Sie sie einfach in das Dreamweaver-SQL-Fenster.

28.1.3 Abfragen verschachteln

In der folgenden Beispielabfrage ist eine andere Darstellung der Daten erwünscht:

- Hauptpunkt 1
  - Unterpunkt 1.1
  - Unterpunkt 1.2
  - Unterpunkt 1.3
- Hauptpunkt 2
  - Unterpunkt 2.1
  - Unterpunkt 2.2

- Hauptpunkt 3
- Hauptpunkt 4
  - Unterpunkt 4.1
  - Unterpunkt 4.2
- Hauptpunkt 5

Diese Datenausgabe kommt beispielsweise als dynamisch generiertes Navigationsmenü in Frage. Mit den nun folgenden Tabellen können Unterpunkte beliebigen Hauptpunkten zugeordnet werden. Wenn Sie möchten, können Sie die Unterpunkte auch mehreren Hauptpunkten zuordnen.

Dynamische Navigation

Für Menüs dynamischer Websites reicht diese Technik meist aus. Wenn Sie weitere Zuordnungsoptionen benötigen, zum Beispiel eine völlig freie Konfigurierbarkeit der Menüs, ist jedoch eine andere, kompliziertere Struktur notwendig.

Die für unser Menü benötigten Abfragen können Sie nicht mehr in der grafischen Ansicht von Dreamweaver erstellen. Um die gewünschte Anzeige zu erreichen, müssen verschachtelte Abfragen geschrieben werden. Diese werden von Dreamweaver leider nicht unterstützt.

In Abbildung 28.9 sehen Sie das Schema der Datenbank. Die Unterpunkte aus den Tabellen *menue_sub* werden in der Tabelle *main_sub* den Hauptpunkten in der Tabelle *menue_main* zugeordnet.

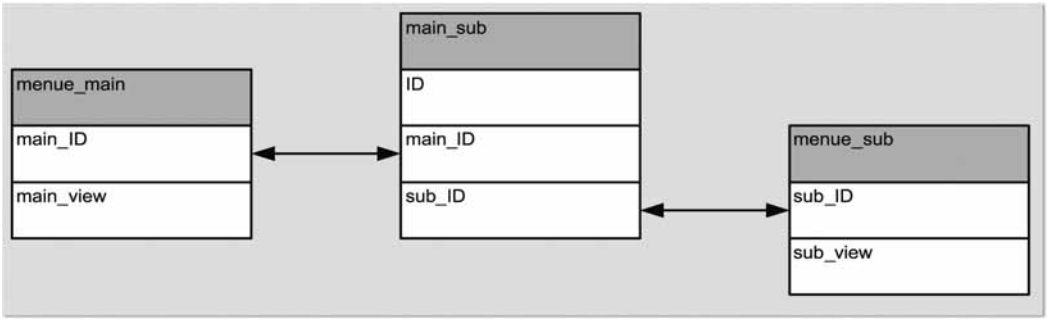


Abbildung 28.9 Datenbankschema

Würde diese Datenbank tatsächlich als Navigation einer dynamischen Website dienen, würden den Tabellen für Haupt- und Unternavigationspunkte weitere Felder mit den IDs der in den Unterseiten anzuzeigenden Datensätze hinzugefügt. In den Abbildungen 28.10 bis 28.12 sehen Sie die Datensatzstruktur der Tabellen.

	Feld	Typ	Kollation	Attribute	Null	Standard	Extra	Aktion
	main_ID	int(11)			Nein	*Kein*	auto_increment	
	main_view	varchar(100)	latin1_swedish_ci		Nein	*Kein*		

Abbildung 28.10 Schema der Tabelle »menue_main«

	Feld	Typ	Kollation	Attribute	Null	Standard	Extra	Aktion
	sub_ID	int(11)			Nein	*Kein*	auto_increment	
	sub_view	varchar(100)	latin1_swedish_ci		Nein	*Kein*		

Abbildung 28.11 Schema der Tabelle »menue_sub«

	Feld	Typ	Kollation	Attribute	Null	Standard	Extra	Aktion
	ID	int(11)			Nein	*Kein*	auto_increment	
	main_ID	int(11)			Nein	*Kein*		
	sub_ID	int(11)			Nein	*Kein*		

Abbildung 28.12 Schema der Tabelle »main_sub«

Wir haben die Tabellen mit den in den Abbildungen 28.13 bis 28.15 dargestellten Inhalten gefüllt.

←T→	main_ID	main_view
	1	Hauptpunkt 1
	2	Hauptpunkt 2
	3	Hauptpunkt 3
	4	Hauptpunkt 4
	5	Hauptpunkt 5

Abbildung 28.13 Inhalte der Tabelle »menue_main«

←T→	sub_ID	sub_view
	1	Unterpunkt 1.1
	2	Unterpunkt 1.2
	3	Unterpunkt 1.3
	4	Unterpunkt 2.1
	5	Unterpunkt 2.2
	6	Unterpunkt 4.1
	7	Unterpunkt 4.2

Abbildung 28.14 Inhalte der Tabelle »menue_sub«

←T→		ID	main_ID	sub_ID
		1	1	1
		2	1	2
		3	1	3
		4	2	4
		5	2	5
		6	4	6
		7	4	7

Abbildung 28.15 Inhalte der Tabelle *main_sub*

In Abbildung 28.16 wird das Prinzip verschachtelter Abfragen schematisch dargestellt.

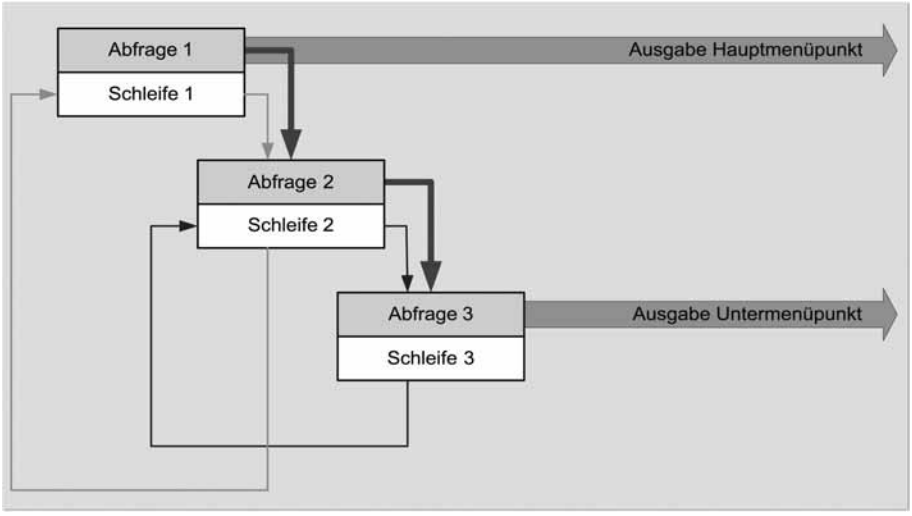


Abbildung 28.16 Schema einer verschachtelten Abfrage

Wie Sie aus den bisherigen Kapiteln wissen, können die Ergebnisse einer Abfrage als Grundlage für weitere Abfragen verwendet werden. Ein Beispiel dafür ist die Anzeige von Daten in einem Detaildokument nach dem Aufruf aus einem Masterdokument heraus. Dabei wird die ID des anzuzeigenden Datensatzes an das nächste Dokument übergeben.

Abhängige Abfragen

Das gleiche Prinzip funktioniert auch innerhalb eines einzelnen Dokuments. Abfrage 1 übergibt einen bestimmten Inhalt an Abfrage 2. Diese liefert nur die Datensätze, die auch mit den aus Abfrage 1 übergebenen Daten übereinstimmen.

Der nächste Schritt gleicht dem ersten. Abfrage 2 übergibt die nun gewonnenen Daten an Abfrage 3. Diese gibt wiederum nur die Daten aus, die mit den erhaltenen Daten aus Abfrage 2 übereinstimmen.

Verbindung zur Datenbank

Dieses Prinzip der Verschachtelung können Sie in nahezu beliebiger Tiefe anwenden. Doch zunächst müssen Sie sich wie immer mit der Datenbank verbinden. Im folgenden Kapitel werden wir diese nochmals benötigen.

```
<?php
// Verbindung zum Server herstellen
mysql_connect("localhost","root","")
or die  ("Zurzeit keine Verbindung möglich");
// Verbindung zur Datenbank "linkliste" herstellen
mysql_select_db("menue")
or die ("Verbindung zur Datenbank zurzeit nicht möglich");
?>
```

Listing 28.2 Datenbankverbindung herstellen

Erste Abfrage

In Listing 28.3 sehen Sie eine einfache Abfrage. Die Ausgabe der Daten erfolgt in einer `while`-Schleife. Bei jedem Schleifendurchlauf wird der Text aus dem Datensatz *main_view* ausgegeben, da wir den Hauptpunkt zur Anzeige bringen wollen.

```
<?php
$abfrage_main = "SELECT * FROM menue_main";
$ergebnis_main = mysql_query($abfrage_main);
while($row_main = mysql_fetch_object($ergebnis_main))
   {
   echo "$row_main->main_view<br>";
   }
mysql_free_result($ergebnis_main);
?>
```

Listing 28.3 Einfache Abfrage

Zweite Abfrage

Innerhalb der Schleife von Abfrage 1 (`abfrage_main`) wird nun die zweite Abfrage, wie in Listing 28.4 zu sehen, eingefügt. In der `WHERE`-Bedingung steht als Wert die Variable `$row_main->main_ID`. Bei jedem Durchlauf der ersten Schleife wird dieser Wert auf den aktuellen Datensatzinhalt der ersten Abfrage (`abfrage_main`) gesetzt, und die zweite Abfrage (`abfrage_main_sub`) erhält aus der Tabelle

main_sub alle Inhalte, die mit der Datensatz-ID der ersten Abfrage (`abfrage_main`) übereinstimmen.

Die Daten der zweiten Abfrage werden in diesem Fall nicht ausgegeben, sondern dienen nur wieder als Grundlage für die dritte Abfrage.

```
<?php
$abfrage_main = "SELECT * FROM menue_main";
$ergebnis_main = mysql_query($abfrage_main);
while($row_main = mysql_fetch_object($ergebnis_main))
   {
   echo "$row_main->main_view<br>";
   $abfrage_main_sub = "SELECT * FROM main_sub WHERE main_ID=
   ".$row_main->main_ID;
   $ergebnis_main_sub = mysql_query($abfrage_main_sub);
while($row_main_sub =mysql_fetch_object($ergebnis_main_sub))
      {
      // Platzhalter
      }
   }
mysql_free_result($ergebnis_main);
mysql_free_result($ergebnis_main_sub);
?>
```

Listing 28.4 Verschachtelte Abfrage

Dritte Abfrage

In Listing 28.5 sehen Sie den Kommentar `// Platzhalter`. Dieser Platzhalter wird nun durch die dritte Abfrage in Listing 28.5 ersetzt. Diese Abfrage erhält wieder eine Variable aus der Abfrage 2 als Bedingung der `WHERE`-Anweisung (`$row_main_sub->sub_ID`).

Da in dieser Abfrage die Tabelle *menue_sub* ausgegeben wird, erhalten Sie als Datenrückgabe alle Datensätze dieser Tabelle, die über die Tabelle *main_sub* einem Hauptpunkt zugeordnet sind. Diese Daten werden auch angezeigt, da es sich um die Untermenüpunkte handelt. Am Ende der Schleifen werden die Daten wieder freigegeben.

```
<?php
$abfrage_main = "SELECT * FROM menue_main";
$ergebnis_main = mysql_query($abfrage_main);
while($row_main = mysql_fetch_object($ergebnis_main))
   {
   echo "$row_main->main_view<br>";
   $abfrage_main_sub = "SELECT * FROM main_sub WHERE main_ID=
   ".$row_main->main_ID;
```

```
   $ergebnis_main_sub = mysql_query($abfrage_main_sub);

while($row_main_sub = mysql_fetch_object($ergebnis_main_sub))
      {
      $abfrage_sub = "SELECT * FROM menue_sub WHERE sub_ID=".
      $row_main_sub->sub_ID;
      $ergebnis_sub = mysql_query($abfrage_sub);
while($row_sub = mysql_fetch_object($ergebnis_sub))
      {
echo "      $row_sub->sub_view<br>";
         }
      }
   }
mysql_free_result($ergebnis_main);
mysql_free_result($ergebnis_sub);
mysql_free_result($ergebnis_main_sub);
?>
```

Listing 28.5 Mehrfache Verschachtelung

Wenn Sie alles richtig angelegt haben, wird Ihnen nun in der Live-Ansicht unser »Menü« wie in Abbildung 28.17 angezeigt.

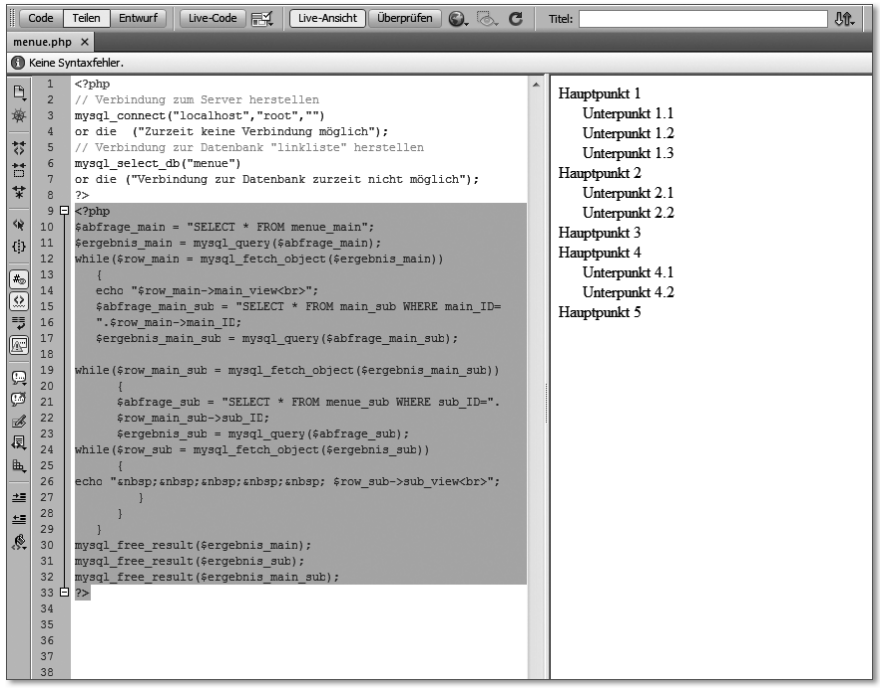


Abbildung 28.17 Ausgabe der Abfrage

Die Abfrage von Datenbanken mit verschachtelten Abfragen kommt häufig vor. Es lohnt sich daher, sich eingehend mit dieser Technik zu befassen.

Code strukturieren [«]

An diesem Beispiel sehen Sie, wie wichtig es ist, den Quelltext übersichtlich und gut zu formatieren. Würden Sie den Code einfach »der Reihe nach« schreiben, würde jede Übersicht über die Zusammengehörigkeit einzelner Abfragen und Codeblöcke fehlen. Mit Einrückungen können Sie Funktionsgruppen einfach und strukturiert darstellen.

```
<body>
<?php
// Abfrage der Hauptmenuepunkte
    $abfrage_main = "SELECT * FROM menue_main";
    $ergebnis_main = mysql_query($abfrage_main);
    while($row_main = mysql_fetch_object($ergebnis_main))

    {
    // Ausgabe der Hauptmenuepunkte
    echo "$row_main->main_view<br>";

            // Abfrage der Zuordnungstabelle
            $abfrage_main_sub = "SELECT * FROM main_sub WHERE main_ID=".$row_main->main_ID;
            $ergebnis_main_sub = mysql_query($abfrage_main_sub);
            while($row_main_sub = mysql_fetch_object($ergebnis_main_sub))

            {
                    // Abfrage der Untermenuepunkte
                    $abfrage_sub = "SELECT * FROM menue_sub WHERE sub_ID=".$row_main_sub->sub_ID;
                    $ergebnis_sub = mysql_query($abfrage_sub);
                    while($row_sub = mysql_fetch_object($ergebnis_sub))

                    {
                    echo "      $row_sub->sub_view<br>";
                    }
            }
    }

                    mysql_free_result($ergebnis_sub);
            mysql_free_result($ergebnis_main_sub);
    mysql_free_result($ergebnis_main);
?>
```

Abbildung 28.18 Gut strukturierter Code bleibt übersichtlich

28.1.4 Eine Volltextsuche

Auf fast jeder Website mit umfangreichen Inhalten ist eine Suchfunktion vorhanden. Gute und treffsichere Suchfunktionen zu programmieren ist schwer und variiert je nach den abzufragenden Inhalten stark. Für die meisten Websites reicht allerdings eine einfache Suche aus. Neben der Strukturierung der Datenbestände ist das Wiederauffinden der Daten eine der Hauptaufgaben des Datenbankdesigns.

SQL unterstützt freie Suchen durch diverse Befehle. Wichtig ist für uns nur `LIKE` in einer SQL-Abfrage. SQL-Abfragen können wie folgt geschrieben werden:

```
"SELECT * FROM '2_0' WHERE description LIKE '%$suchbegriff%'"
```

Der Inhalt von `$suchbegriff` ist eine Zeichenkette, nach der im Datensatz *description* der Tabelle *2_0* gesucht werden soll. Diese Variable wird aus einem Formular an das Ergebnisdokument übergeben (siehe Abbildungen 28.19 und 28.20).

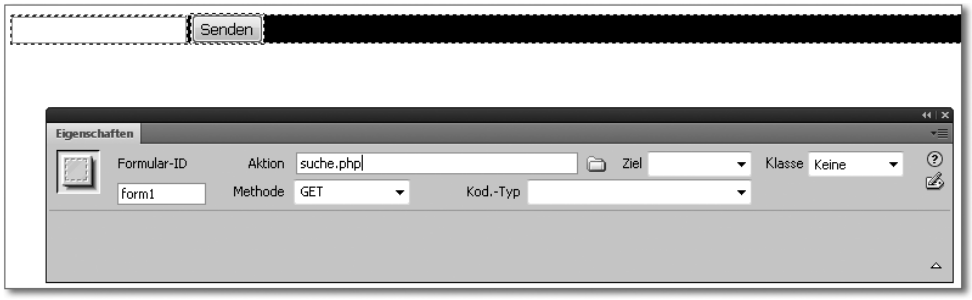


Abbildung 28.19 Suchformular

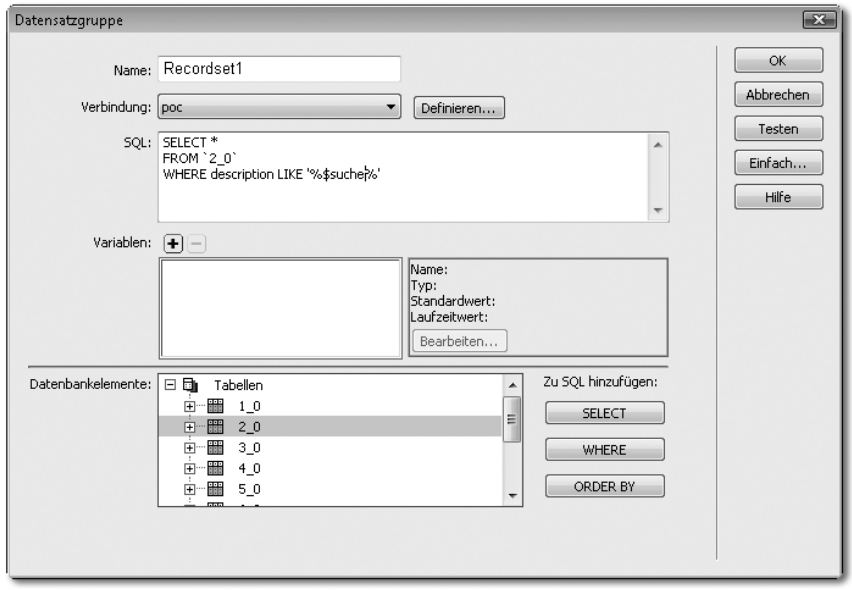


Abbildung 28.20 Datensatzgruppe mit Suchabfrage

Da es häufig Probleme gibt, die `GET`-Arrays direkt in die SQL-Abfrage zu schreiben, wird am Beginn des Dokuments folgende Codezeile eingefügt:

```
$suche = $_GET['SUCHE'];
```

Die Variable aus dem `GET`-Array wird zunächst in der Variablen `$suche` zwischengespeichert. Die Abfrage wird dann mit dem Inhalt der Variablen `$suche` durchgeführt.

Die Datensatzgruppe wird im Bedienfeld BINDUNGEN angezeigt, und Sie können wie immer die gewünschten Datenfelder in Ihr Dokument ziehen. Damit alle gefundenen Datensätze angezeigt werden, müssen Sie einen wiederholten Bereich mit den Einstellungen ALLE DATENSÄTZE erstellen. Geben Sie durch Suchabfragen generierten Datensätzen eindeutige Bezeichnungen, um Verwechslungen mit »normalen« Abfragen zu vermeiden.

28.2 Arbeiten mit Sessions

Bei vielen dynamischen Websites müssen Daten über den gesamten Zeitraum eines Seitenzugriffs hinterlegt werden. Die wohl häufigste Anwendung ist dabei ein Warenkorb. Innerhalb eines Shopsystems können Sie an beliebiger Stelle innerhalb der Website Artikel in einer Art Container ablegen. Bei der endgültigen Bestellung stehen Ihnen alle vorher ausgewählten Artikel zur Verfügung.

Arbeitsweise von Sessions

Warenkörbe werden sehr häufig mit Sessions realisiert. In Kapitel 19, »Einführung in PHP«, haben wir bereits grundlegende Eigenschaften und Befehle zum Arbeiten mit Sessions erklärt.

In Abbildung 28.21 wird die Arbeitsweise von Sessions grafisch dargestellt. Während des Site-Zugriffs wird eine Session gestartet, die Ihnen dann während der gesamten Zeit zur Verfügung steht. Jedes der einzelnen Dokumente kann auf die Inhalte der Session zugreifen und Variablen hinzufügen, abfragen oder manipulieren.

Um mit Sessions zu arbeiten, müssen die Inhalte der Session an alle Dokumente, die Sie anfordern, übermittelt werden. Dies kann durch Cookies oder durch die Übermittlung des Session-Arrays als URL-Parameter erfolgen. Die Möglichkeit, Sessions als File auf dem Server zu hinterlegen, wird von vielen Providern aus Sicherheitsgründen deaktiviert. Da Sie nicht wissen, ob ein User Cookies aktiviert hat, sollten Sie die Session in jedem Fall zusätzlich als URL-Parameter mit übergeben:

```
<a href="sessions_2.php?SID">Link</a>
```

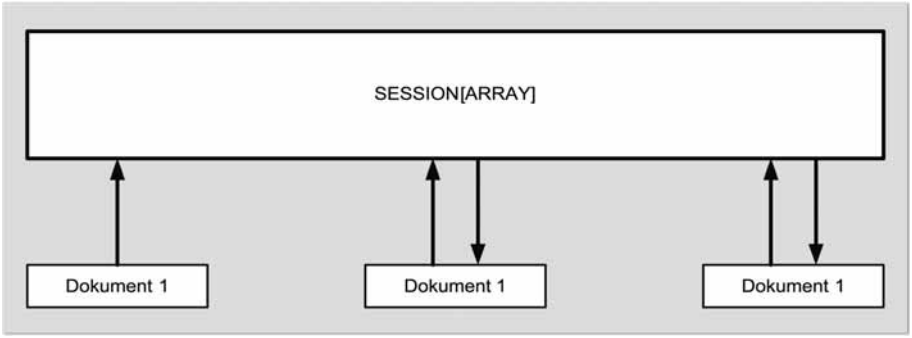


Abbildung 28.21 Schema der Arbeit mit Sessions

Dazu hängen Sie die `SID` (Session-Variable) an den Hyperlink. Durch das Anhängen der Variablen `SID` wird die URL wie in Abbildung 28.22 aufgerufen. Um die Session-ID selbst müssen Sie sich nicht kümmern. `SID` ist ein Platzhalter, der von PHP erkannt wird.

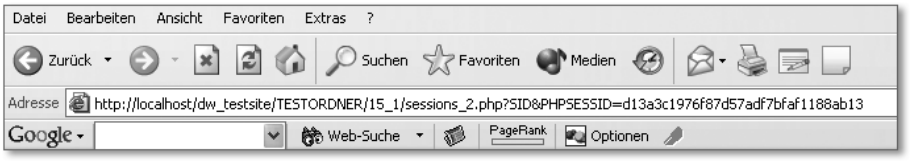


Abbildung 28.22 Browseradresszeile mit Session-ID

Session starten

Damit ein PHP-Dokument mit Sessions arbeiten kann, sollte der folgende Befehl am besten immer direkt am Anfang des Dokuments stehen, damit eine Session gestartet oder auf eine bestehende Session zugegriffen wird:

```
session_start();
```

Variablen registrieren

Im folgenden Beispiel haben wir fünf Variablen mit den verschiedensten Inhalten deklariert. Damit diese Variablen in der Session zur Verfügung stehen, müssen sie in der Session registriert werden. Das geschieht mit dem Befehl `session_register()`. Beachten Sie, dass dabei die Variablen ohne Dollarzeichen in runden Klammern geschrieben werden.

```
<?
session_start();
$var_1 = 200;
$var_2 = 400;
```

```
$var_3 = "Auch ganze Zeilen sind erlaubt";
$var_4 = "Wie Sie sehen,";
$var_5 = "sehr komfortabel";
session_register('var_1');
session_register('var_2');
session_register('var_3');
session_register('var_4');
session_register('var_5');
?>
```

Listing 28.6 Registrieren von Session-Variablen

In jedem der folgenden Dokumente stehen Ihnen bei gestarteter Session die Variablen wie im folgenden Listing wieder zur Verfügung.

```
<p><?php echo $_SESSION['var_1']; ?> </p>
<p><?php echo $_SESSION['var_2']; ?></p>
<p><?php echo $_SESSION['var_3']; ?></p>
<p><?php echo $_SESSION['var_4']; ?>
<?php echo $_SESSION['var_5']; ?></p>
```

Listing 28.7 Anzeigen der Session-Variablen

Die Arbeit mit Sessions können Sie sich in Dreamweaver erleichtern, indem Sie im Bedienfeld BINDUNGEN Sitzungsvariablen anlegen (siehe Abbildung 28.23). Das Registrieren der Variablen müssen Sie im Quelltext vornehmen. Die Anzeige der Session-Inhalte können Sie jedoch wie gewohnt im Layoutmodus erledigen.

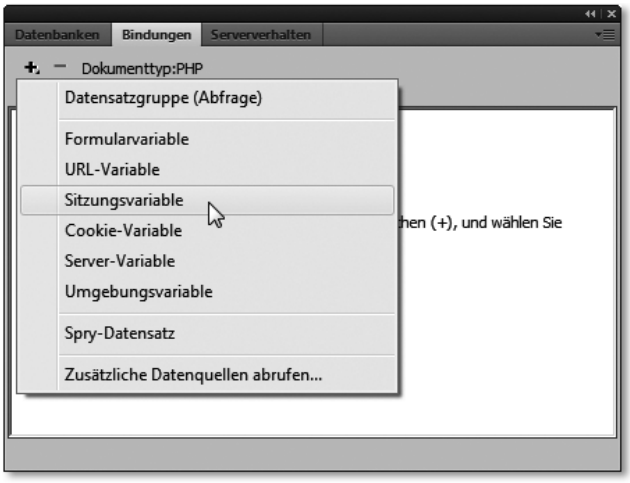


Abbildung 28.23 Anlegen einer Sitzungsvariablen

In Abbildung 28.24 sehen Sie das Bedienfeld Bindungen mit Sitzungsvariablen. Um die Variablen in einem Dokument auszugeben, ziehen Sie diese wie gewohnt in Ihr Dokument.

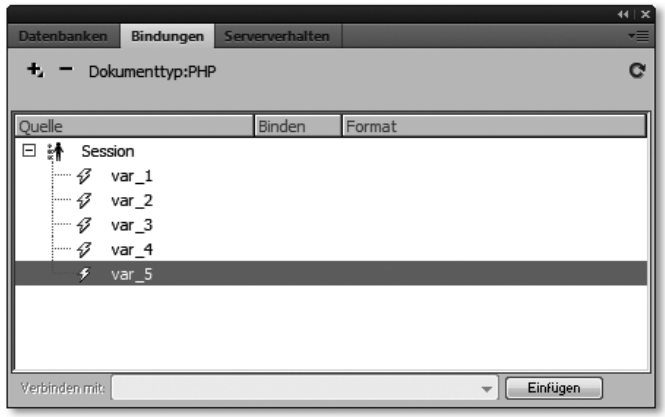


Abbildung 28.24 Sitzungsvariablen in Dreamweaver

Sessions werden übrigens nicht nur von PHP unterstützt, auch andere Programmiersprachen für Websites verfügen über ähnliche Technologien zur Datenspeicherung.

28.3 Serververhalten

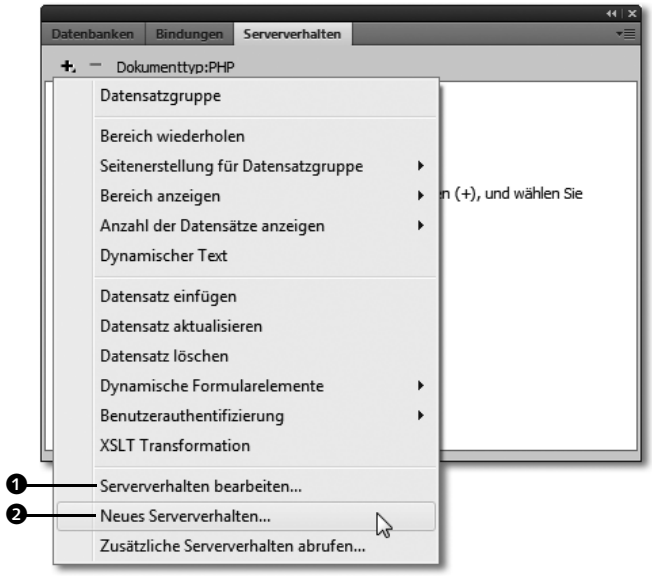
28.3.1 Eigene Serververhalten schreiben

In vielen Bereichen ist Dreamweaver erweiterbar. Auch für dynamische Websites können Sie eigene Serververhalten schreiben und diese dauerhaft hinterlegen. In der nächsten Übung werden wir eine eigene Datenbank-Konnektierung erstellen, die als Serververhalten gespeichert wird.

[▪] Schritt-für-Schritt: Datenbank-Konnektierung als Serververhalten speichern

1 *Neues Serververhalten anlegen*

Im Bedienfeld Serververhalten finden Sie neben den bereits vorhandenen Serververhalten auch die Menüpunkte Neues Serververhalten ❷ und Serververhalten bearbeiten ❶.



Wählen Sie aus dem Menü NEUES SERVERVERHALTEN, um ein eigenes, neues Serververhalten anzulegen.

2 *Dialogbox »Neues Serververhalten«*

In der folgenden Dialogbox geben Sie den Namen für das neue Serververhalten an ❸. Dieser wird später in der Auswahlliste angezeigt.

Wenn Sie bereits Serververhalten angelegt haben und diese nur geringfügig modifizieren wollen, können Sie auch ein Serververhalten kopieren. Aktivieren Sie dann die Option VORHANDENES SERVERVERHALTEN KOPIEREN ❹, und Sie können aus der Liste, die in unserer Abbildung deaktiviert ist, ein zu kopierendes Serververhalten auswählen.

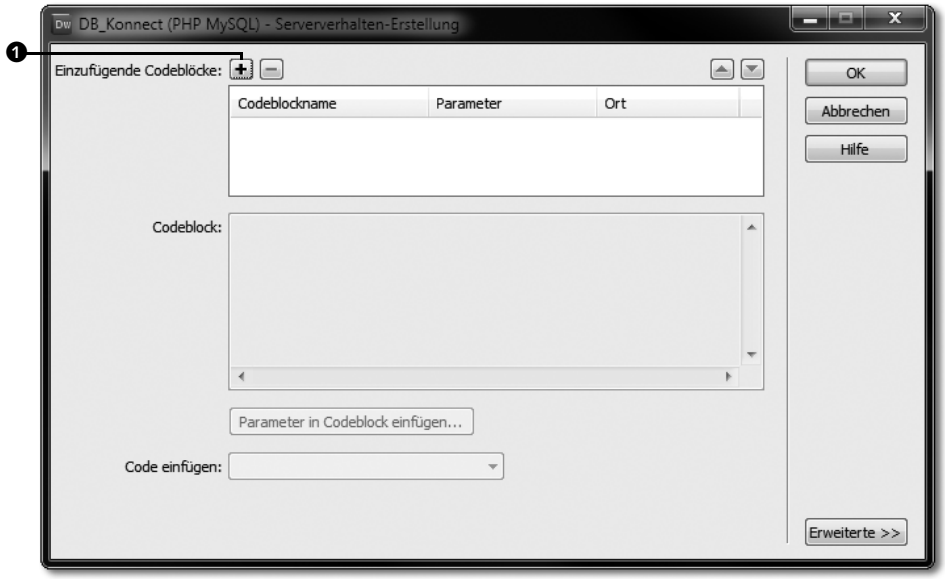


Leider können die mitgelieferten Serververhalten von Dreamweaver nicht kopiert oder modifiziert werden. Diese Option steht Ihnen nur für eigene Serververhalten zur Verfügung.

3 *Codeblock anlegen*

Serververhalten bestehen aus einem oder mehreren Codeblöcken. Legen Sie zunächst durch Klicken auf das Plussymbol ❶ einen neuen Codeblock an.

Nicht alle Serververhalten, die im Dokument verwendet werden, werden im entsprechenden Fenster auch angezeigt. Wenn Sie möchten, dass Ihres aufgelistet wird, aktivieren Sie den Bezeichner Verhalten im Fenster »Serververhalten« anzeigen, wenn ein Codeblock im Dokument gefunden wird. Dreamweaver findet mit dieser Option auch bereits angelegte Codeblöcke und zeigt sie korrekt als Serververhalten an, wenn der Quelltext mit dem Codeblock übereinstimmt. Vorher müssen Sie die Erweiterte Ansicht aktivieren.



4 *Codeblock benennen und bestimmen*

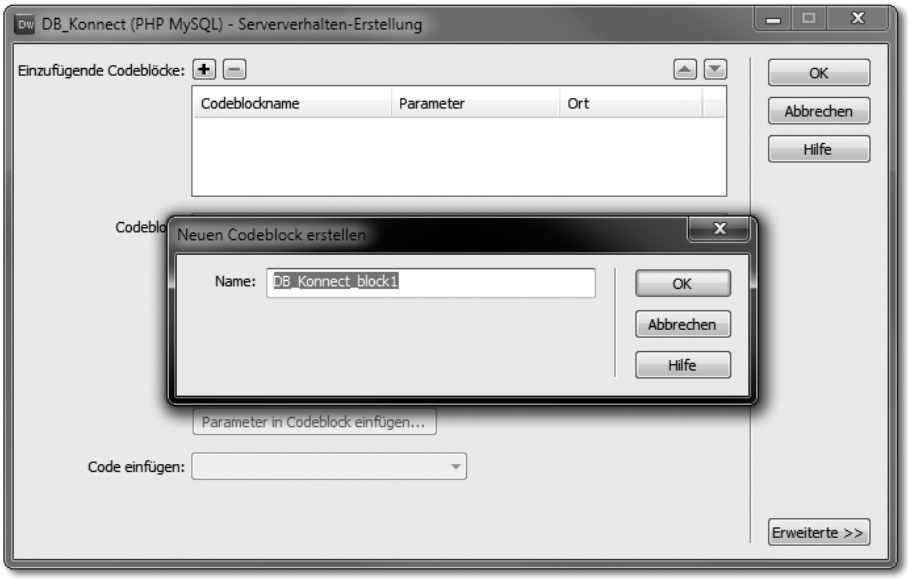
In der nächsten Dialogbox müssen Sie dem Codeblock innerhalb des Serververhaltens einen Namen zuweisen.

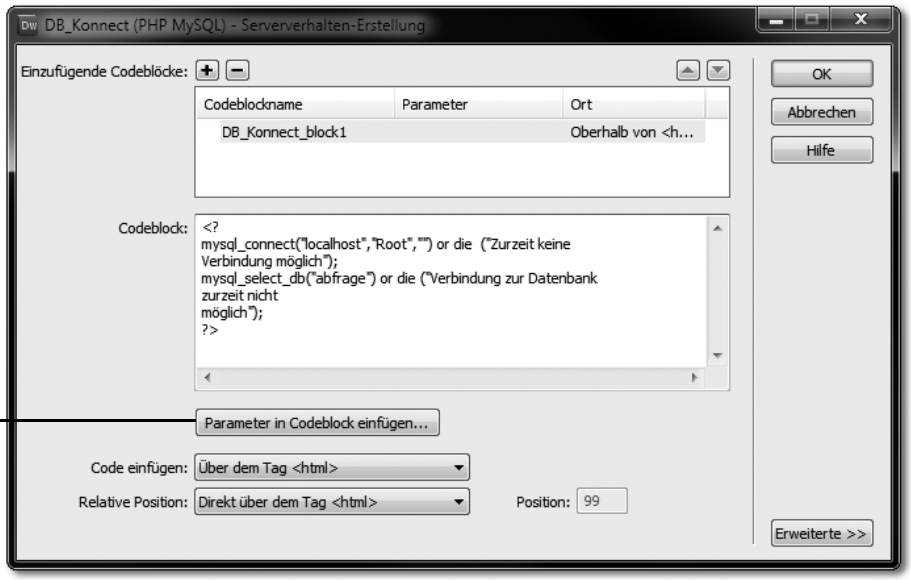
Die folgende Ansicht erinnert stark an das Bedienfeld Codefragmente und ist von der Arbeitsweise her auch sehr ähnlich. Kopieren Sie die Datenbank-Konnektierung aus dem vorherigen Abschnitt 28.1.3, »Abfragen verschachteln«, in den Fensterbereich Codeblock:

```
<?
mysql_connect("localhost","Root","") or die  ("Zurzeit keine
Verbindung möglich");
```

```
mysql_select_db("abfrage") or die ("Verbindung zur Datenbank
zurzeit nicht möglich");
?>
```

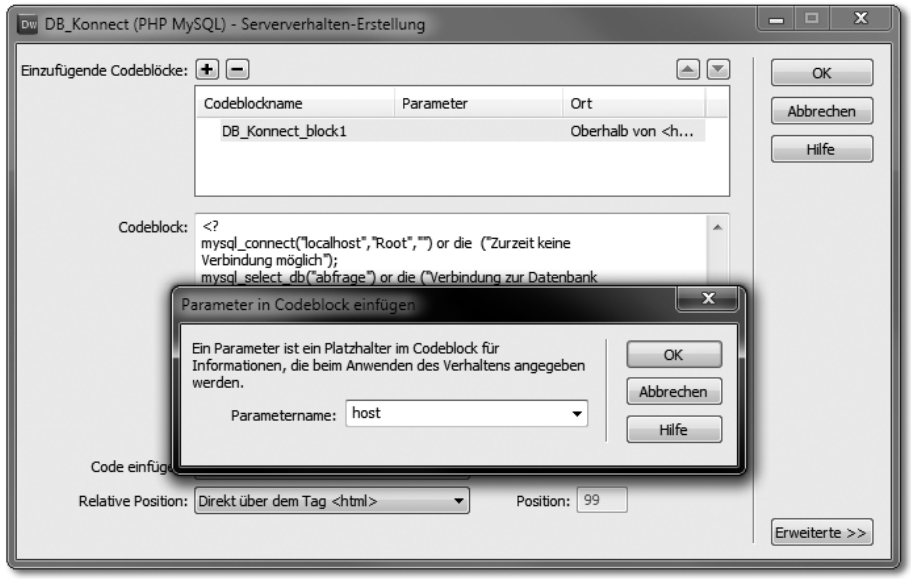
Listing 28.8 Codeblock für die Datenbank-Konnektierung





5 *Platzhalter einfügen*

Da es sich bei Serververhalten um dynamische Elemente handelt, ist es nahezu unerlässlich, auch Variablen einzufügen. Sie können das mit den Platzhaltern `@@Variablenname@@` erreichen. Markieren Sie dazu den zu ersetzenden Teil im Codeblock (oder positionieren Sie den Cursor an der gewünschten Stelle), und klicken Sie auf Parameter in Codeblock einfügen ❷ (in der vorherigen Abbildung).



In der Abbildung sehen Sie, dass wir *host* ausgewählt haben, um es durch einen Parameter zu ersetzen. Geben Sie in der Dialogbox den gewünschten Namen für den Parameter ein. Verwenden Sie am besten allgemeine beschreibende Begriffe, die ihre Gültigkeit über das aktuelle Dokument hinaus behalten.

6 *Eingefügte Parameter*

Wie Sie dem Listing entnehmen können, wurde *localhost* durch einen Platzhalter ersetzt. Diesen Arbeitsschritt müssen Sie nun für alle Parameter `User`, `Pass` und `Datenbank` wiederholen. Wenn Sie alles korrekt durchgeführt haben, steht im Codeblock Folgendes:

```
<?
mysql_
connect("@@Host@@","@@User@@","@@Passwort@@") or die ("Zurzeit keine
Verbindung möglich");
```

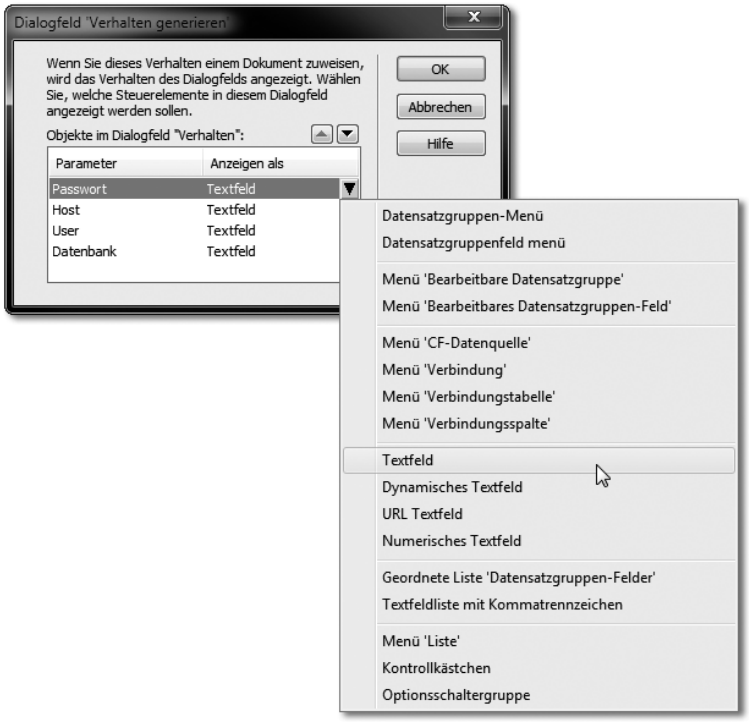
```
mysql_select_db("@@Datenbank@@") or
  die ("Verbindung zur Datenbank zurzeit nicht möglich");
?>
```

Listing 28.9 Codeblock mit Parametern (Platzhaltern)

Wenn Sie möchten, können Sie natürlich auch die Fehlermeldungen durch Platzhalter ersetzen und eigene von PHP ausgeben lassen.

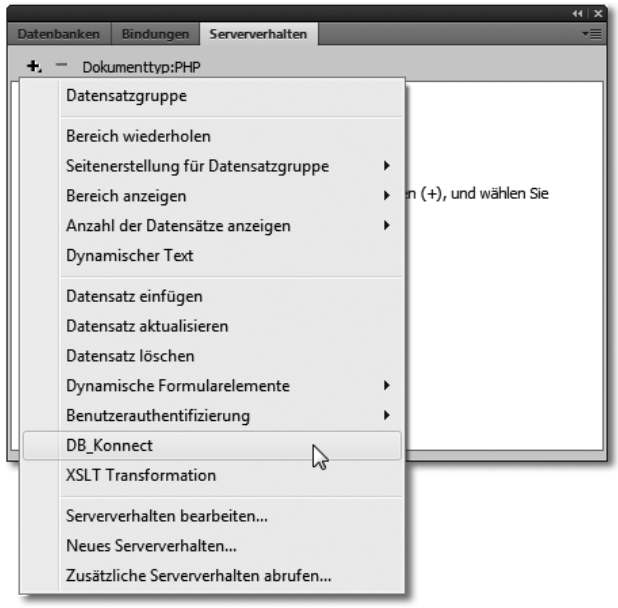
7 *Parameter übergeben*

Der Codeblock ist nun fast fertig. Als Letztes müssen Sie auswählen, auf welche Weise die Parameter dem Codeblock übergeben werden sollen. In unserem Fall sind es einfache Textfelder, die auch als Standardvorgabe eingetragen wurden. Diese Textfelder erscheinen immer, wenn Sie ein Serververhalten in Ihr Dokument einfügen.



8 *Fertiges Serververhalten*

Das fertige Serververhalten wird jetzt in der Auswahlliste angezeigt, und Sie können es bereits verwenden.



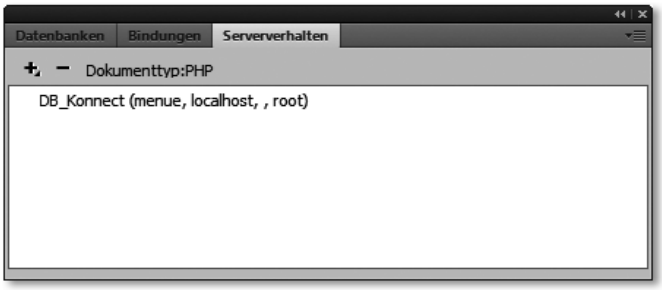
9 *Serververhalten testen*

Wenn Sie nun das Serververhalten einfügen möchten, erscheint eine Dialogbox mit den von Ihnen festgelegten Parametern, die Sie nun eingeben müssen. Wie bereits erwähnt, sind dies die Felder, die mit der Dialogbox Verhalten generieren erzeugt werden.



10 *Fertig eingefügtes eigenes Serververhalten*

Das Serververhalten wird im Dokument eingefügt, die Platzhalter wurden durch Ihre Eingaben ersetzt. Im Bedienfeld Serververhalten wird es mit den von Ihnen vergebenen Parametern angezeigt.



28.3.2 Positionierung im Code

Wie Sie während der Arbeit bemerken werden, können Sie selbst festlegen, an welcher Stelle im Code das Verhalten eingefügt werden soll. Es stehen Ihnen dabei die in der folgenden Tabelle aufgeführten Möglichkeiten zur Verfügung.

Platzierungsoption	Positionierung im Code
vor dem Tag `<html>`	▸ am Anfang einer Datei ▸ vor den Abfragen für die Datensatzgruppen ▸ nach den Abfragen für die Datensatzgruppen ▸ vor dem Beginn des HTML-Tags ▸ an einer von Ihnen festgelegten Position
unterhalb des Tags `</html>`	▸ am Ende einer Datei ▸ vor dem Ende der Datensatzgruppen ▸ nach dem Ende der Datensatzgruppen ▸ nach dem Tag `</html>` ▸ an einer von Ihnen festgelegten Position
relativ zu einem bestimmten Tag	Auswahl eines bestimmten Tags und einer von Ihnen vorgegebenen Position
relativ zur Auswahl	▸ vor der Auswahl ▸ nach der Auswahl ▸ Auswahl ersetzen ▸ um die Auswahl legen

Tabelle 28.1 Einfügeoptionen

Reihenfolge von Code

Bei der Programmierung spielt die Reihenfolge des Codes immer eine große Rolle. So können Sie beispielsweise niemals eine Abfrage vor einer Datenbank-

Konnektierung durchführen oder einen Datensatz vor der Abfrage ausgeben. Programmteile sind meist voneinander abhängig und können keine Daten verarbeiten, die nicht vorhanden sind. Daher ist es für viele Serververhalten entscheidend, an welcher Stelle im Skript der Codeblock eingefügt wird, und es ist sehr wichtig, diese eben beschriebenen Positionierungen einzustellen.

Gewichtung festlegen

Wenn Sie als Relative Position eine Benutzerdefinierte Stelle ❶ auswählen, können Sie zusätzlich eine Gewichtung (Position) ❷ angeben. Ein Block mit der Gewichtung 50 wird vor einem Block mit der Gewichtung 60 eingefügt. Dreamweaver erteilt allen Codeblöcken, die auf Datensatzabfragen zugreifen, die Standardgewichtung 50. Wenn Sie mehrere Blöcke mit denselben Gewichtungen anlegen, wählt Dreamweaver die Reihenfolge der Codeblöcke im Quelltext nach dem Zufallsprinzip aus.

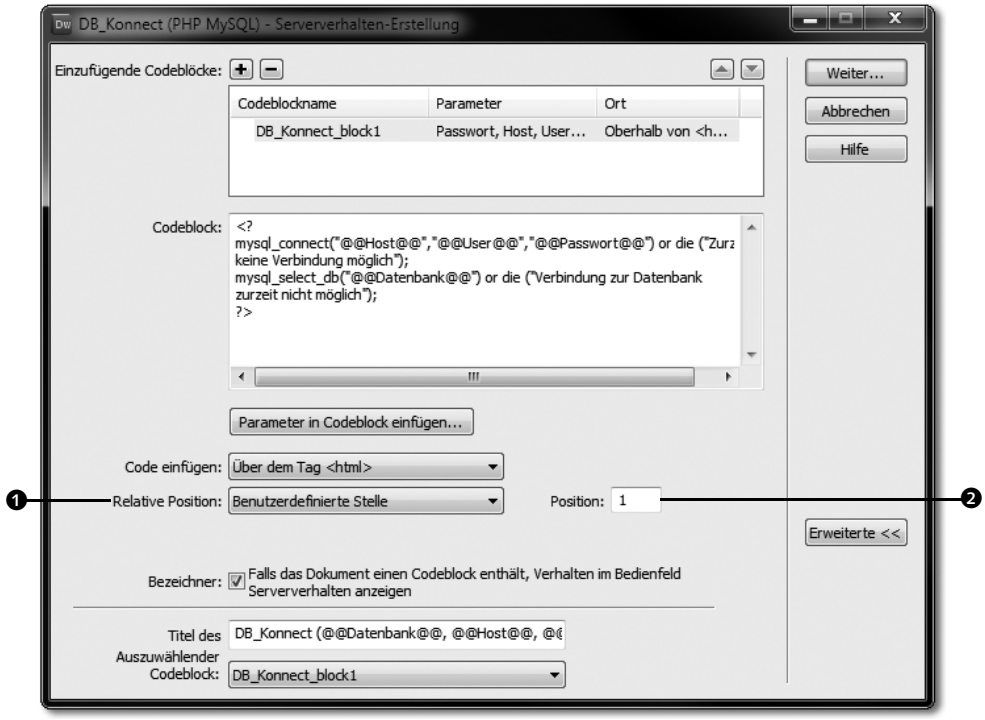


Abbildung 28.25 Gewichtung der Codeblöcke einstellen

28.3.3 Bedingungen als Serververhalten anlegen

Das Erstellen eigener Serververhalten ist ein sehr komplexes Thema. Dreamweaver unterstützt neben den bereits erwähnten Einstellmöglichkeiten auch die Bildung von Schleifen und Bedingungen in Codeblöcken mit Dreamweaver-spezifischen Codeparametern. Für diese sehr tief gehenden Einstellungen und die Programmierung von Erweiterungen verweisen wir auf die ausgezeichnete und umfangreiche Dokumentation von Dreamweaver.

PHP-Bedingung

Das nun folgende Serververhalten hat nichts mit den Dreamweaver-spezifischen Bedingungen gemeinsam, sondern ist eine reine PHP-Bedingung. Wie Sie sicherlich bereits festgestellt haben, kommt es recht häufig vor, dass bestimmte Codefragmente nur angezeigt werden sollen, wenn eine Bedingung innerhalb eines PHP-Befehls erfüllt ist.

Sie können diese Bedingungen auch als Serververhalten anlegen; es gibt dabei aber einige Kleinigkeiten zu bedenken. In Abbildung 28.26 sehen Sie ein neues Serververhalten und den enthaltenen Codeblock:

```
<?
if(@@Wert_B@@ = @@Wert_A@@)
{
echo "@@Ausgabe@@";
}
?>
```

Listing 28.10 Bedingung als Serververhalten

Damit das Serververhalten nicht an einer von Dreamweaver ausgewählten Stelle platziert wird, geben wir als Positionierung Relativ zur Auswahl und Auswahl ersetzen vor. Damit wird der Codeblock direkt am im Quelltext positionierten Cursor eingesetzt.

Im Codeblock sind drei Parameter eingetragen. Diese Parameter werden im fertigen Skript durch die beiden zu vergleichenden Variablen oder Werte und durch die gewünschte Ausgabe bei erfüllter Bedingung ersetzt. Die Vorgaben für das Dialogfeld werden als dynamische Textfelder gewählt, damit als Vergleichsparameter auch Datenfelder auswählbar sind.

Beim anschließenden Test wählen wir zwei identische Datensätze aus, um sicherzustellen, dass die Bedingung erfüllt ist, und ein einfaches »Hallo« als »Erfolgsmeldung«, das im Dokument bei erfüllter Bedingung ausgegeben werden soll.

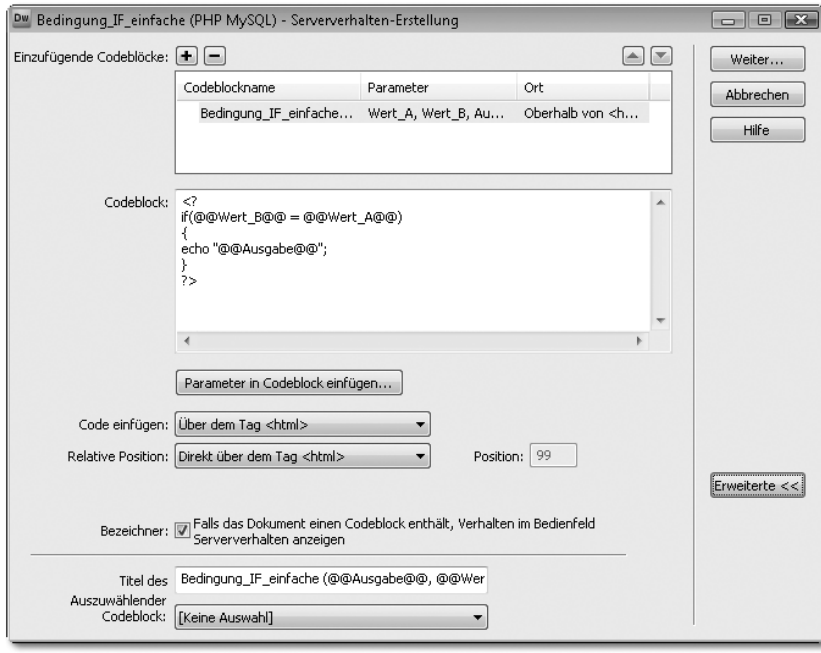


Abbildung 28.26 Bedingung als Serververhalten

Wie Sie feststellen werden, arbeitet das eben erstellte Serververhalten fehlerhaft, wenn nicht von Hand eingegriffen wird. Der von Dreamweaver erzeugte Code kann nicht funktionieren:

```
<?
if(<?php echo $row_Recordset1['name']; ?>
=<?php echo $row_Recordset1['name']; ?>) { echo"Hallo"; }
?>
```

Listing 28.11 Fehlerhaftes Verhalten

Dreamweaver setzt um Datensatzfelder herum immer ein PHP-Tag. Dies bewirkt, dass die PHP-Tags nicht korrekt geschlossen werden, da auch in unserem Serververhalten diese Tags vorkommen. Leider kann man Dreamweaver auch nicht überreden, das Serververhalten wie im folgenden Listing korrekt anzulegen:

```
<?
if( ?>@@Wert_B@@  <? =  ?>@@Wert_A@@<? )
{
echo "@@Ausgabe@@";
}
```

Listing 28.12 Von Dreamweaver nicht akzeptiert

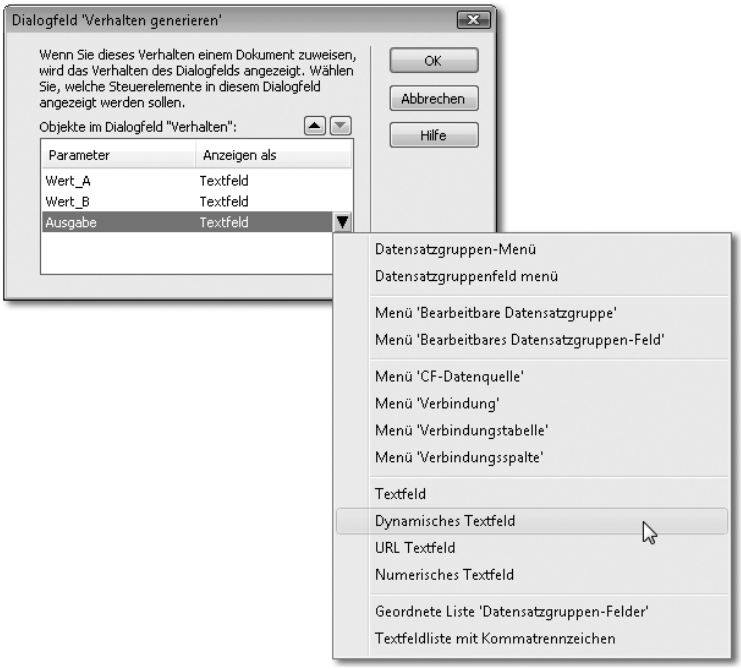


Abbildung 28.27 Anzeigen der Parameter als dynamisches Textfeld

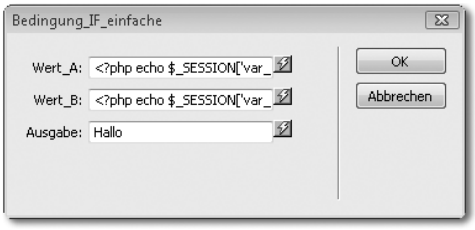


Abbildung 28.28 Testen des Serververhaltens

Dreamweaver weigert sich beharrlich, PHP innerhalb eines Codeblocks mehrfach zu beginnen, und meckert sofort. Warum das so ist, ist uns nicht klar. Es sollte eigentlich keine Schwierigkeit sein, das zu erlauben. Besonders verwunderlich ist dies, weil Dreamweaver im von ihm selbst erstellten Quelltext regelrecht exzessiv PHP beginnt und wieder beendet.

Sie kommen bei diesem Serververhalten nicht umhin, die Schreibweise in der Dialogbox wie in Abbildung 28.26 von Hand zu ändern und die PHP-Tags zu entfernen. Dies stellt, wenn man es weiß, auch kein größeres Problem dar. In der Dialogbox können Sie eingeben und bearbeiten, was Sie möchten. Einen dynami-

schen Datensatz einzufügen ist ja nur eine der Optionen, die Ihnen zur Verfügung stehen.

```
<?
if($row_Recordset1['name'] = $row_Recordset1['name'])
{
echo "Hallo";
}
?>
```

Listing 28.13 Berichtigtes Verhalten

[»] **Codefragmente**

Neben dem Anlegen eigener Serververhalten können Sie natürlich Ihre Skripte auch als Codefragmente einfügen. Die Möglichkeit, Parameter zu vergeben, haben Sie dann allerdings nicht.

Bezugsquellen für Serververhalten

Auf der Website von Adobe finden Sie unter »Adobe Exchange« noch viele weitere Serververhalten, die Sie über den Extension Manager installieren können.

Auch einige kommerzielle Serververhalten sind erhältlich; bis hin zu kompletten Shopsystemen als Serververhalten ist fast alles zu finden. Erfreulicherweise gibt es mittlerweile auch viele PHP-Skripte.



Abbildung 28.29 Adobe Dreamweaver Exchange

Hier folgt nun eine Zusammenfassung der verschiedenen Möglichkeiten, Ihren Websites über die Einfügeleiste dynamisches Verhalten zuzuweisen.

29 Menüs für dynamisches Verhalten

29.1 Tabellendaten importieren

Mit einem Klick auf das erste Icon ❶ der Einfügeleiste DATEN können Sie aus Excel-Tabellen oder aus CSV-Daten automatisch eine HTML-Tabelle erstellen lassen.

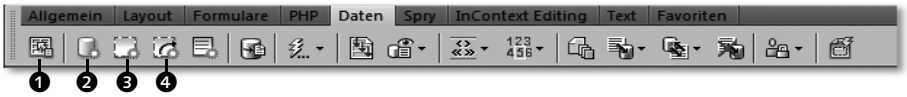


Abbildung 29.1 Die Einfügeleiste »Daten«

29.2 Spry-XML-Datensatz

Um einen XML-Datensatz mit Spry verwenden zu können, müssen Sie ihn zuerst definieren. Die Funktion ❷ entspricht in etwa dem Anlegen einer Datenbankverbindung. Spry-XML-Datensätze sind die Basis für alle weiteren Spry-Funktionen.

29.3 Spry-Bereich

Der Spry-Bereich ❸ umschließt ähnlich wie das Formular-Tag den Bereich in Ihrem Dokument, in dem die Spry-Datenausgabe erfolgen soll. Um einen Spry-Bereich zu erstellen, muss zuerst ein Spry-XML-Datensatz angelegt werden.

29.4 Spry-Wiederholung

Spry-XML-Daten werden hier ❹ in einer Schleife nach Ihren Vorgaben im Dokument ausgegeben. Das Verhalten entspricht dem Serververhalten von DATENSATZ WIEDERHOLEN.

29.5 Spry-Wiederholungsliste

Spry-Wiederholungslisten ❶ geben Werte aus der XML-Datei in Form einer Liste aus.



Abbildung 29.2 Die Einfügeleiste »Daten«

29.6 Datensatzgruppen

Durch einen Klick auf den Button Datensatzgruppe ❷ erstellen Sie eine Datenbankabfrage nach von Ihnen vorzugebenden Parametern. Vor dem Einfügen einer Datensatzgruppe müssen Sie eine Verbindung zur Datenbank aufgebaut haben.

Das Anlegen einer Datensatzgruppe ist die Grundlage für fast alle weiteren nun folgenden Aktionen.

29.7 Dynamische Daten

Mit Hilfe der Einstellungen unter Dynamische Daten ❸ verbinden Sie verschiedene Elemente des Dokuments mit Inhalten aus der Datensatzgruppe.

Die verschiedenen Einträge im Drop-down-Menü bewirken Folgendes:

- Der Eintrag Dynamische Tabelle fügt Ihrem Dokument eine Tabelle mit einem oder mehreren Datensätzen hinzu. Die Tabelle wird dabei automatisch erstellt. Welche Felder dargestellt werden, legen Sie bereits in der Datensatzgruppe fest; Sie können aber auch nachträglich einzelne Felder wieder aus der fertigen Tabelle entfernen.
- Dynamischer Text fügt Ihrem Dokument einen Text aus einer Datenbankabfrage hinzu.
- Dynamisches Textfeld hingegen verbindet ein vorher in das Dokument einzufügendes Formularelement Textfeld mit dem Inhalt eines Datensatzes.
- Dynamisches Kontrollkästchen verbindet ein vorher in das Dokument eingefügtes Formularelement Kontrollkästchen mit dem Inhalt eines Datensatzes. Dabei ist zu bedenken, dass der Status eines Kontrollkästchens nur über eindeutige true/false-Inhalte festgelegt werden kann.

- Dynamische Optionsschaltergruppe verbindet ein vorher in das Dokument eingefügtes Formularelement Optionsschaltergruppe mit den Inhalten eines Datensatzes. Dabei ist zu bedenken, dass der Status der Optionsschaltergruppe ebenfalls nur über eindeutige true/false-Inhalte festgelegt werden kann.
- Dynamische Auswahlliste verbindet ein vorher in das Dokument eingefügtes Formularelement Liste/Menü mit dem Inhalt einer Spalte aus einer Datensatzgruppe.

29.8 Bereich wiederholen

Bereich wiederholen ❹ definiert die zuvor im Dokument markierten dynamischen Elemente als eben solchen. Dieser Bereich wird gemäß Ihren Vorgaben (oder solange Datensätze vorhanden sind) wiederholt.

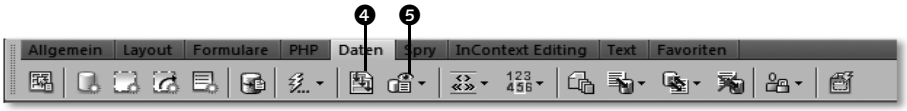


Abbildung 29.3 Die Einfügeleiste »Daten«

29.9 Bereich anzeigen

Die Einträge in diesem Menü ❺ erstellen eine Bedingung, um Elemente Ihres Dokuments in Abhängigkeit von dem Inhalt eines Datensatzes anzuzeigen.

Hier finden Sie im Drop-down-Menü folgende Möglichkeiten:

- Anzeigen, wenn Datensatzgruppe leer ist: Der Bereich wird angezeigt, wenn eine ausgewählte Datensatzgruppe keinen Inhalt hat. Achten Sie darauf, dass ein leerer Inhalt auch ein Inhalt sein kann. Arbeiten Sie bei fehlerhaften Verhalten mit NULL-Marken in der Datenbank.
- Anzeigen, wenn Datensatzgruppe nicht leer ist: Der Bereich wird angezeigt, wenn eine ausgewählte Datensatzgruppe einen Inhalt hat.
- Anzeigen, wenn erste Seite: Diese Übersetzung oder Bezeichnung ist etwas unglücklich gewählt. Bei mehr Datensätzen, als in einem Dokument ausgegeben werden können, zeigen Sie immer nur einen bestimmten Teilbereich der gesamten Datensätze an. Mit diesem Serververhalten legen Sie fest, dass der gewählte Bereich nur angezeigt wird, wenn die aktuell im Dokument angezeigten Datensätze die ersten (zum Beispiel die ersten fünf) enthaltenen Datensätze einer Abfrage sind.

- Anzeigen, wenn nicht erste Seite: Das ist das Gegenteil der soeben beschriebenen Aktion. Der ausgewählte Bereich wird angezeigt, wenn die dargestellten Datensätze nicht die ersten sind.
- Anzeigen, wenn letzte Seite: Der Bereich wird angezeigt, wenn die Datensätze die letzten einer Datensatzabfrage sind.
- Anzeigen, wenn nicht letzte Seite: Der Bereich wird angezeigt, wenn die Datensätze nicht die letzten einer Datensatzabfrage sind.

29.10 Seitenerstellung für Datensatzgruppe

Mit diesem Serververhalten ❶ können Sie für die abgefragten Datensätze eine Navigation erstellen oder vorhandene Elemente als Navigation verwenden.



Abbildung 29.4 Die Einfügeleiste »Daten«

Hier finden Sie im Drop-down-Menü folgende Möglichkeiten:

- Datensatzgruppen-Navigationsleiste: Erstellt in Ihrem Dokument eine komplette Datensatz-Navigationsleiste, die Sie gestalterisch an Ihre Anforderungen anpassen können.
- Zur ersten Seite verschieben: Auch hier haben wir wieder eine etwas unglückliche Übersetzung. Dieses Serververhalten erstellt einen Link, mit dem Sie zu den ersten Datensätzen einer Abfrage navigieren können.
- Zur vorherigen Seite verschieben: Dieses Serververhalten erstellt einen Link, mit dem Sie zu den vorherigen Datensätzen einer Abfrage navigieren können. Ein Beispiel: Sie zeigen in Ihrem Dokument aktuell die Datensätze 10 bis 15 an. Es werden gleichzeitig immer fünf Datensätze angezeigt. Das Verhalten fügt einen Link in Ihr Dokument ein, mit dem Sie zu den Datensätzen 5 bis 10 navigieren können.
- Zur nächsten Seite verschieben: Dieses Serververhalten erstellt einen Link, mit dem Sie zu den nächsten Datensätzen einer Abfrage navigieren können.
- Zur letzten Seite verschieben: Dieses Serververhalten erstellt einen Link, mit dem Sie zu den letzten Datensätzen einer Abfrage navigieren können.

29.11 Datensatzgruppen-Navigationsstatus

Dieses Serververhalten ❷ hilft Ihnen, auf der Website die Orientierung innerhalb der Datensätze zu behalten, indem es den Status der aktuell angezeigten Datensätze ausgibt.

Die Auswahlmöglichkeiten im Drop-down-Menü:

- Datensatzgruppen-Navigationsstatus zeigt einen kompletten Navigationsstatus an: Datensatz von X bis Y von Z.
- Start-Datensatz zeigt an, welche Position der erste angezeigte Datensatz innerhalb aller Datensätze hat.
- End-Datensatz zeigt an, welche Position der letzte angezeigte Datensatz innerhalb aller Datensätze hat.
- Datensätze gesamt gibt die Gesamtzahl der vorhandenen Datensätze aus.

29.12 Master-Detaildatensatz

Dieses sehr umfangreiche Serververhalten ❸ legt zwei Dokumente automatisch an: Eine Übersichtsseite mit Datensätzen ist die *Masterseite*. In dieser Masterseite können Sie einen einzelnen Datensatz auswählen und in der ebenfalls automatisch erzeugten *Detailseite* komplett anzeigen lassen.



Abbildung 29.5 Die Einfügeleiste »Daten«

Nach dem Erstellen dieser Seiten können Sie diese an Ihre gestalterischen Vorgaben anpassen. Die Aktion eignet sich im Zusammenspiel mit einer Datensatznavigation hervorragend dazu, schnelle Datenbankabfragen zu generieren. In einem Layout ist es meist einfacher, die Abfragen einzeln auszuführen, als die von Dreamweaver vorgegebenen Layouts anzupassen.

29.13 Datensatz einfügen

Diese Gruppe ❹ ermöglicht Ihnen das interaktive Erstellen von Einfügeformularen für Datensätze. Welche der beiden Methoden Sie verwenden, entscheiden Sie anhand Ihres Layouts.

Hier finden Sie im Drop-down-Menü folgende Möglichkeiten:

- Assistent Einfügeformular für Datensätze: Mit diesem Assistenten können Sie interaktiv ein komplettes Einfügeformular für Datensätze erstellen und anschließend das Layout des Formulars anpassen. Es müssen vorher keine Formulare oder Formularelemente erstellt werden. Dreamweaver erledigt diesen Arbeitsschritt automatisch.
- Datensatz einfügen: Im Gegensatz zum Assistenten müssen Sie hierfür zunächst ein Formular erstellen und die gewünschten Formularelemente einfügen.

29.14 Datensatz aktualisieren

Diese Gruppe ❶ ermöglicht Ihnen das interaktive Erstellen von Aktualisierungsformularen für Datensätze. Es gelten die gleichen Bedingungen wie beim Einfügeformular.

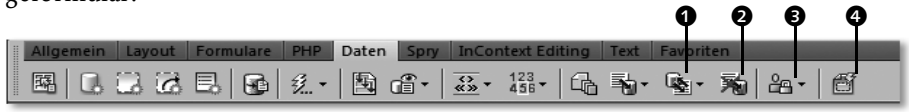


Abbildung 29.6 Die Einfügeleiste »Daten«

Die Auswahlmöglichkeiten im Einzelnen:

- Assistent Aktualisierungsformular für Datensätze
  Mit diesem Assistenten können Sie interaktiv ein komplettes Aktualisierungsformular für Datensätze erstellen und anschließend das Layout des Formulars anpassen. Es müssen vorher keine Formulare oder Formularelemente erstellt werden. Dreamweaver erledigt diesen Arbeitsschritt automatisch.
- Datensatz aktualisieren
  Im Gegensatz zum Assistenten müssen Sie hierfür zunächst ein Formular erstellen und die gewünschten Formularelemente einfügen.

29.15 Datensatz löschen

Mit dieser Aktion ❷ können Sie einzelne Datensätze löschen. Sie müssen vorher ein Dokument anlegen, um eine Auswahl der zu löschenden Datensätze zu treffen.

29.16 Benutzerauthentifizierung

Diese Gruppe ❸ ermöglicht Ihnen, Zugriffsrechte für Ihre Seiten zu vergeben und Benutzerformulare zur Anmeldung und zum Anlegen neuer User zu erstellen.

Hier finden Sie im Drop-down-Menü folgende Möglichkeiten:

- Benutzer anmelden: Erstellt eine Anmeldeseite für Benutzer im aktuellen Dokument. Die Formularelemente und eine Tabelle in der Datenbank für die Benutzer müssen vorher angelegt werden.
- Benutzer abmelden: Erstellt einen Link zur Abmeldung eines Benutzers.
- Zugriff auf Seite beschränken: Fügt dem Dokument eine Überprüfung der Zugriffsrechte hinzu. Die Zugriffsrechte werden anhand von Bezeichnungen in der User-Datenbank überprüft.
- Neuen Benutzernamen überprüfen: Ermöglicht durch ein von Ihnen vorher angelegtes Formular die Überprüfung neuer Nutzer. Überprüft wird dabei nur, ob der Benutzername bereits vergeben wurde oder nicht.

29.17 XSL-Transformation

Mit diesem dynamischen Verhalten ❹ können Sie eine XSL-Transformation in ein leeres Dokument einfügen. Sie benötigen hierbei einen Zugriff auf die entsprechenden XML-Daten auf Ihrem oder einem fremden Server.

Es ist in aller Munde: kaum ein neues Programm oder Programm-Update, das nicht XML unterstützt. Also wird es Zeit, sich mit diesem Thema etwas näher zu befassen. Wir erläutern die Grundlagen und zeigen Ihnen, wie Sie in Dreamweaver damit arbeiten können.

30 XML in Dreamweaver

XML ist in den letzten Jahren zum Standard für viele Anwendungsbereiche geworden. Es gibt kaum eine Software, die nicht einen XML-Export oder -Import unterstützt. Grund genug, sich damit näher zu befassen.

30.1 Einführung in XML

XML steht für *Extended Markup Language* und ist eine Beschreibungssprache für Dokumente und Daten aller Art. XML ist ein Dialekt der Sprache bzw. des Sprachkonstrukts *SGML* (*Standard Generalized Markup Language*). Auch HTML gehört zur Gruppe der SGML-Dialekte, daher auch die recht große Ähnlichkeit im Aufbau dieser beiden. Seit 1998 ist XML in der Version 1.0 vom W3C standardisiert.

Beschreibung der Daten

Während HTML eine Auszeichnungssprache ist und etwas über die Anzeige von Daten zu sagen hat, beschreibt XML die Daten selbst. Bei HTML sind Daten, Funktion und Formatierungen ein einziger Mischmasch, der zwar beeindruckende Darstellungen im Browser erzeugt, aber keine Aussagen über die Inhalte treffen kann.

`<p></p>` ist ein Absatz, egal was darin steht. Es sagt denkbar wenig über die Inhalte des Absatzes aus. Um Daten zu strukturieren, ist HTML also absolut ungeeignet.

Trennung von Inhalt, Darstellung und Funktion

XML ist gewissermaßen die logische Konsequenz aus HTML, denn es trennt Inhalt, Darstellung und Funktion vollständig voneinander. Damit erreicht XML eine völlige Plattform- und Software-Unabhängigkeit. Ab sofort können Sie so

Ihre *eigenen Tags* schreiben. Vorbei die Zeit von `p`, `h1` und `table`. Jetzt können Sie Ihre Tags auch `<heinz>`, `<willi>` und `<bienemaya>` nennen. Sie können es nicht glauben? Sehen Sie selbst:

```
<?xml version="1.0" encoding="iso-8859-1"?>
<bienenvolk>
   <heinz>Hier ist Heinz</heinz>
   <willi>Willi kennt auch jeder</willi>
   <biene_maya>Und das hier ist die Biene Maya</biene_maya>
</bienenvolk>
```

Abbildung 30.1 Biene Maya in XML

[o] Wir legen die Tags `<heinz>`, `<willi>` und `<bienemaya>` an. Sie gehören zur Gruppe von `<bienenvolk>`. Und wie Sie in Abbildung 30.2 sehen, es funktioniert. Die zugehörigen Dateien finden Sie auch auf der DVD unter *Beispiele/XML*.

Mit dieser XML-Datei sind anscheinend keine Style-Informationen verknüpft.
Nachfolgend wird die Baum-Ansicht des Dokuments angezeigt.

```
– <bienenvolk>
    <heinz>Hier ist Heinz</heinz>
    <willi>Willi kennt auch jeder</willi>
    <biene_maya>Und das hier ist die Biene Maya</biene_maya>
  </bienenvolk>
```

Abbildung 30.2 Biene Maya in Firefox als Baumstruktur

Struktur von XML-Dokumenten

Sie erfahren anhand dieses kleinen Beispiels bereits eine Menge über die Struktur eines XML-Dokuments:

- Zu Beginn muss eine eindeutige Dokumenttyp-Angabe erfolgen. Diese wird von Dreamweaver automatisch beim Anlegen eines XML-Dokuments eingefügt.
- Die Tags sind innerhalb der üblichen Schreibkonventionen frei definierbar.
- Und, das ist der Grund für das Tag `Bienenvolk`: Es muss ein Root-Element geben – das ist zwingend vorgeschrieben.

Die Tags können Sie – wie bereits erwähnt – beliebig benennen und zudem so viele davon anlegen, wie Sie wollen.

Im Browser sieht das Ganze noch etwas merkwürdig aus, wie Sie in Abbildung 30.2 sehen. Die Darstellung wird jedoch gleich noch besser. Zunächst noch kurz etwas zur Syntax: In Abbildung 30.3 sehen Sie das Schema einer richtig angelegten XML-Struktur. Die Tags können verschachtelt werden, genau so, wie Sie es aus HTML kennen. XML ist dabei jedoch wesentlich intoleranter als HTML.

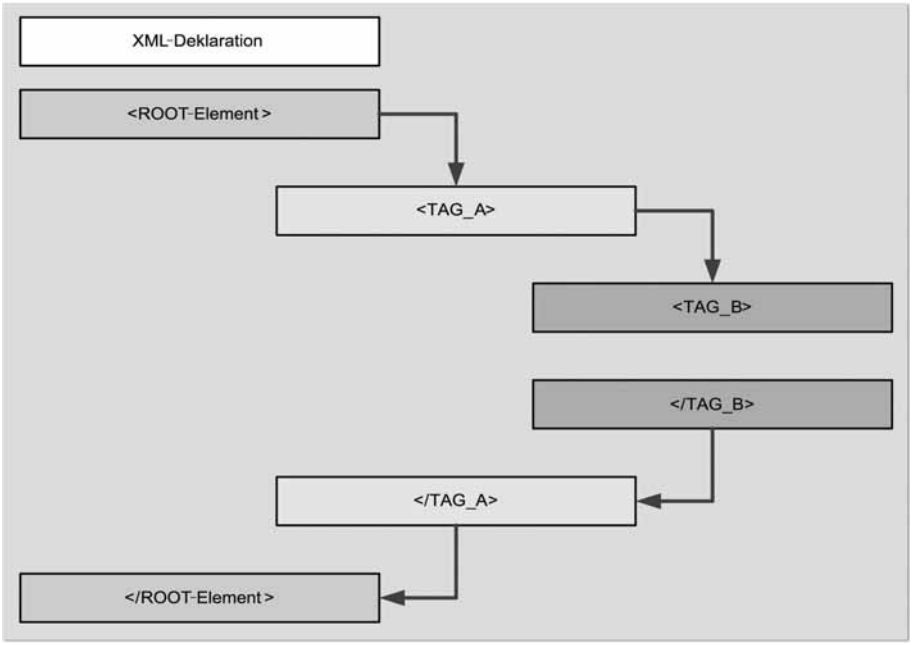


Abbildung 30.3 Richtiges, wohlgeformtes XML-Schema

In Abbildung 30.4 wurden die Tags über Kreuz verschachtelt. Das ist zwar auch in HTML nicht korrekt, es funktioniert aber in vielen Browsern. Bei XML führt dies sofort zu einem Fehler.

Wohlgeformte Dokumente

XML-Dokumente müssen *wohlgeformt* sein. Dieser etwas merkwürdige Ausdruck beschreibt die korrekte Struktur eines XML-Dokuments, die auch als *Baumstruktur* bezeichnet wird. Das Tag `<TAG_B>` ist dem Tag `<TAG_A>` untergeordnet. Bei XML spricht man in diesem Fall vom *Child-Element* (Kind) eines Tags. Das Tag `<TAG_A>` ist damit das *Parent-Element* (Elternteil) von `<TAG_B>`. Tags werden bei XML auch als Elemente bezeichnet. Die Gesamtheit `TAG_A` ist somit das Element `TAG_A`. Tags müssen in XML *immer* abgeschlossen werden.

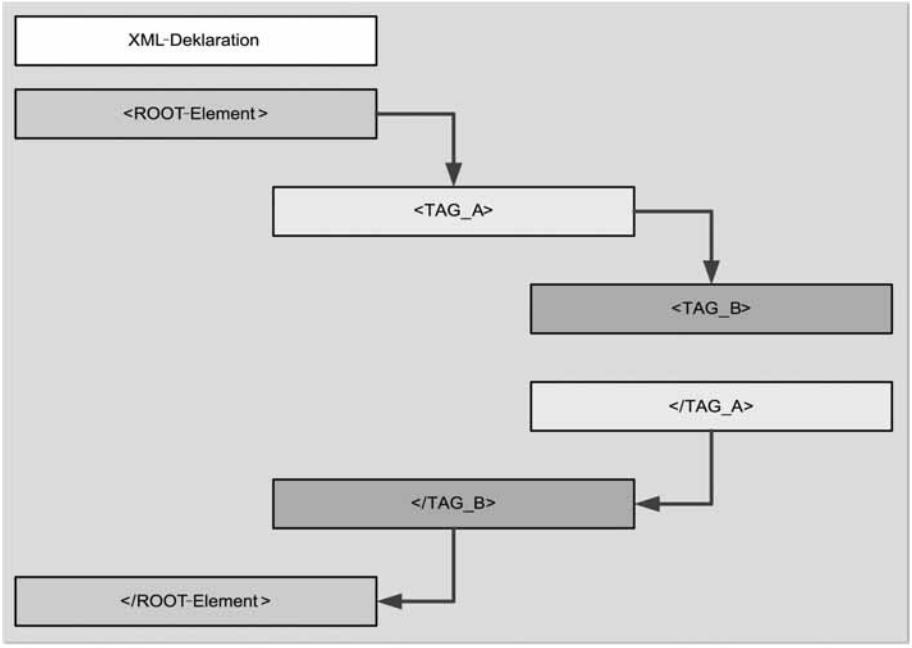


Abbildung 30.4 Fehlerhaftes XML-Schema

30.2 XML und CSS

Hübsch anzusehen ist XML im Browser bislang nicht. Zur Gestaltung von XML-Dokumenten können Sie zunächst einfache CSS-Dateien verwenden. Probieren wir es aus:

```
heinz {
    font-family: Arial, Helvetica, sans-serif;
    font-size: 20px;
   position:absolute;
   top:10px;
   left:10px;
   background-color:#FFC;
   font-weight: bold;
    height: 200px;
    width: 200px;
    border: 1px solid #FF9900;
}
willi {
   position:absolute;
   top:10px;
```

```
   left:220px;
   background-color:#CCC;
    font-family: Arial, Helvetica, sans-serif;
    font-size: 20px;
    font-weight: bold;
    height: 200px;
    width: 200px;
    border: 1px solid #FF9900;
}
biene_maya {
   position:absolute;
    font-family: Arial, Helvetica, sans-serif;
    font-size: 20px;
    font-weight: bold;
    height: 200px;
    width: 200px;
    border: 1px solid #0000FF;
    left: 430px;
    top: 10px;
   background-color:#FC0;
}
```

Listing 30.1 CSS-Stil für Biene Maya

In Listing 30.1 sehen Sie einen ganz normalen CSS-Stil. Nun verbinden wir die beiden Dokumente miteinander, indem wir die externe CSS-Datei einbinden, wie aus HTML-Dokumenten gewohnt.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<?xml-stylesheet type="text/css" href="xml.css"?>
<bienenvolk>
   <heinz>Hier ist Heinz</heinz>
   <willi>Willi kennt auch jeder</willi>
   <biene_maya>Und das hier ist die Biene Maya</biene_maya>
</bienenvolk>
```

Listing 30.2 Biene Maya in XML

Wie Sie in Abbildung 30.5 sehen, werden die Formatvorlagen aus der CSS-Datei ganz normal auf die selbstdefinierten Tags angewandt. Und wie Sie vermutlich richtig erkennen, eröffnet Ihnen das in Zukunft ungeahnte Möglichkeiten.

Sie sehen also ein Stück Zukunft im Internet und in der gesamten Datenverarbeitung vor sich. Das mag übertrieben klingen, ist unserer Einschätzung nach aber sogar eine Untertreibung. Ein Großteil der aktuellen Software verwendet diesen Standard bereits. Alle aktuellen Browser stellen XML-Dateien dar.



Abbildung 30.5 Biene Maya im Internet Explorer

XML-Dateien öffnen

XML-Dateien können Sie in vielen Anwendungen öffnen. Auffällig bei allen Versionen ist, dass die eigentliche Datenstruktur überall vollständig erhalten bleibt, auch wenn die Darstellung völlig unterschiedlich ist. Wir haben es bereits erwähnt: XML trennt Daten, Design und Funktion gänzlich voneinander.



Abbildung 30.6 Biene Maya in Word

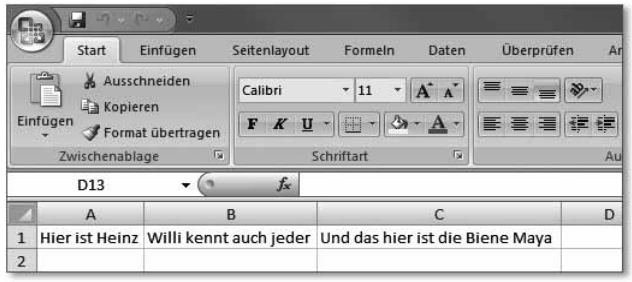


Abbildung 30.7 Biene Maya in Excel

30.3 XML-Attribute

Unter XML können Sie nicht nur Tags selbst definieren, Sie können den Tags auch Attribute zuweisen und diese Attribute selbst anlegen:

```
<?xml version="1.0" encoding="iso-8859-1"?>
<?xml-stylesheet type="text/css" href="xml.css"?>
<bienenvolk kategorie="trickfilm">
   <heinz inhalt="name">
      Hier ist Heinz
   </heinz>
   <willi inhalt="hauptfigur">
      Willi kennt auch jeder
   </willi>
   <bienemaya inhalt="hauptfigur">
      Und das hier ist die Biene Maya
   </bienemaya>
</bienenvolk>
```

Listing 30.3 Attribute in XML

Processing Instructions [«]

Neben den Attributen können in den Elementen sogenannte *Processing Instructions* hinterlegt werden, die einer verarbeitenden Software Anweisungen zur Behandlung des jeweiligen Elements übergeben. Da dies jedoch wieder softwarespezifisch ist, werden wir hierauf nicht näher eingehen.

XML erlaubt den Austausch der Daten mit verschiedensten Anwendungen. Im Beispiel haben wir eine Excel-Tabelle als XML abgelegt und in Dreamweaver geöffnet (siehe Abbildungen 30.8 und 30.9).

	A	B	C	D	E	F	G
1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9
10	10	10	10	10	10	10	10
11	11	11	11	11	11	11	11
12	12	12	12	12	12	12	12
13	13	13	13	13	13	13	13
14	14	14	14	14	14	14	14
15	15	15	15	15	15	15	15
16	16	16	16	16	16	16	16
17							
18							

Abbildung 30.8 Excel-Tabelle mit XML-Export

Wie Sie in Abbildung 30.9 sehen, stellt Dreamweaver die XML-Dateien im Quelltext dar. Änderungen an den Dateien sind somit problemlos möglich – etwas, was bis vor einiger Zeit bei Microsoft-Produkten völlig undenkbar war, wenn man nicht mit der Brechstange Hand an die Dateien legte.

```
1  <?xml version="1.0"?>
2  <?mso-application progid="Excel.Sheet"?>
3  <Workbook xmlns="urn:schemas-microsoft-com:office:spreadsheet"
4   xmlns:o="urn:schemas-microsoft-com:office:office"
5   xmlns:x="urn:schemas-microsoft-com:office:excel"
6   xmlns:ss="urn:schemas-microsoft-com:office:spreadsheet"
7   xmlns:html="http://www.w3.org/TR/REC-html40">
8   <DocumentProperties xmlns="urn:schemas-microsoft-com:office:office">
9    <Author>Richard Beer</Author>
10   <LastAuthor>Richard Beer</LastAuthor>
11   <Created>2007-11-10T14:03:03Z</Created>
12   <Company>firstART - Agentur für Design und Consulting</Company>
13   <Version>11.8132</Version>
14  </DocumentProperties>
15  <ExcelWorkbook xmlns="urn:schemas-microsoft-com:office:excel">
16   <WindowHeight>15585</WindowHeight>
17   <WindowWidth>19995</WindowWidth>
18   <WindowTopX>480</WindowTopX>
19   <WindowTopY>75</WindowTopY>
20   <ProtectStructure>False</ProtectStructure>
```

Abbildung 30.9 Öffnen der XML-Datei in Dreamweaver

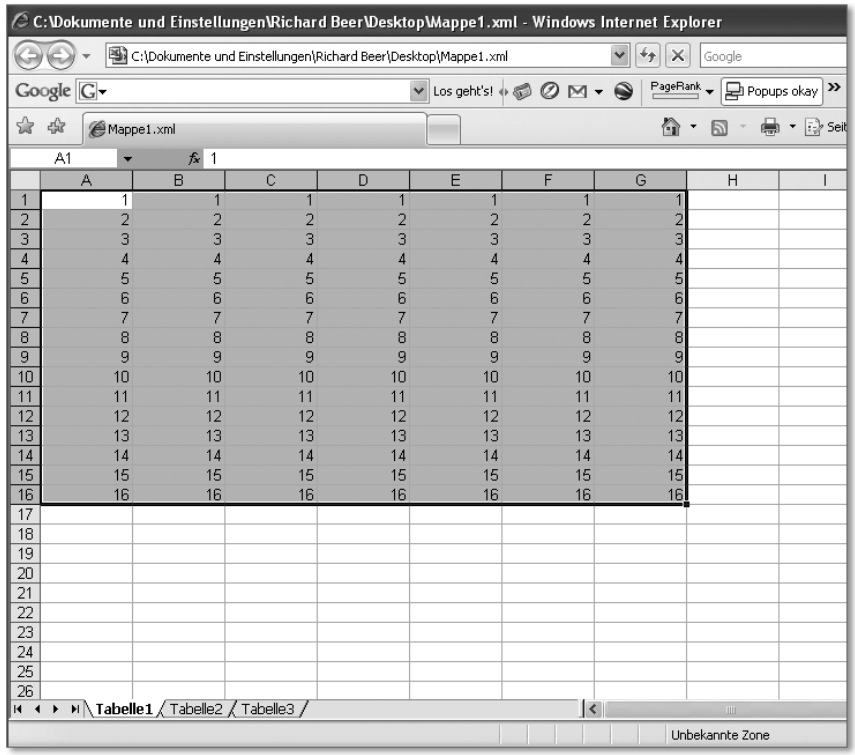


Abbildung 30.10 Vorschau der XML-Datei im Internet Explorer

Die Darstellung funktioniert allerdings leider nur im Internet Explorer. Firefox zeigt die Datenstruktur an, da in diese XML-Datei Microsoft-spezifische Processing Instructions eingebunden sind, um Darstellungsanweisungen aus der Office-Palette an die Software zu übergeben.

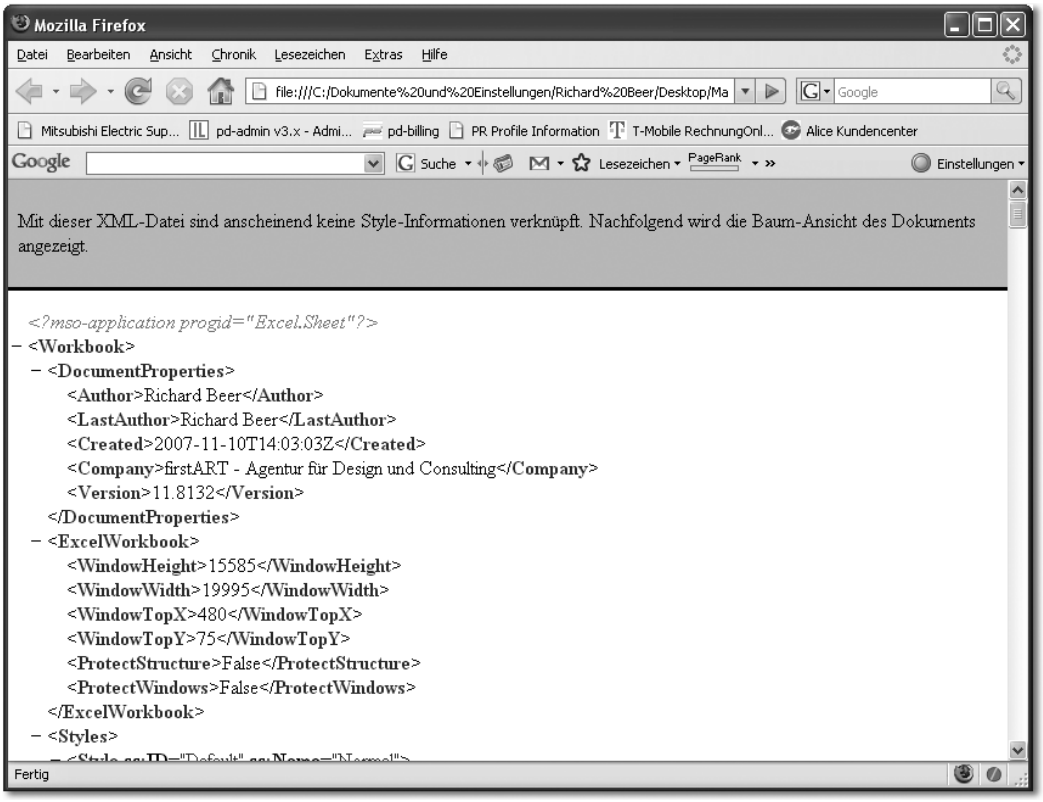


Abbildung 30.11 Darstellung der XML-Datei in Firefox

30.4 DTDs

Sie erinnern sich, dass Sie in den Voreinstellungen von Dreamweaver unterschiedliche DTDs auswählen können. In jeder Kopfzeile eines HTML-Dokuments wird auf eine DTD verwiesen:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
```

DTD heißt *Document Type Definition* oder *DOC-Type-Definition*. In der DTD für HTML ist festgeschrieben, welche Tags es gibt und über welche Attribute ein Tag

verfügen darf. DTDs bestimmen, welche Daten in einem Element (Tag) enthalten sein dürfen und welche nicht. DTDs sind äußerst dogmatisch. Alles, was in einer DTD nicht ausdrücklich erlaubt wird, ist verboten.

Auch wenn unsere XML-Dateien einwandfrei funktionieren, gehört zu einer XML-Datei eine DTD, die genau die Elemente (Tags) festlegt. Das XML-Dokument *Biene Maya* sieht mit einer internen DTD wie folgt aus:

```
<?xml version="1.0" encoding="iso-8859-1"?>
<?xml-stylesheet type="text/css" href="xml.css"?>
<!-- Beginn DTD -->
<!DOCTYPE bienenvolk
[
   <!ELEMENT bienenvolk (heinz, willi, bienemaya)>
   <!ATTLIST bienenvolk kategorie CDATA #REQUIRED>

      <!ELEMENT heinz (#PCDATA)>
      <!ATTLIST heinz inhalt CDATA #REQUIRED>

      <!ELEMENT willi (#PCDATA)>
      <!ATTLIST willi inhalt CDATA #REQUIRED>

      <!ELEMENT bienemaya (#PCDATA)>
      <!ATTLIST bienemaya inhalt CDATA #REQUIRED>
]>
<!-- Ende DTD -->
<bienenvolk kategorie="trickfilm">
   <heinz inhalt="name">
Hier ist Heinz
</heinz>
<willi inhalt="hauptfigur">
Willi kennt auch jeder
</willi>

<bienemaya inhalt="hauptfigur">
Und das hier ist die Biene Maya
</bienemaya>
</bienenvolk>
```

Listing 30.4 XML-Dokument mit interner DTD

Wie bereits gesagt: Eine DTD beschreibt die einzelnen Elemente eines XML-Dokuments und schreibt Inhalte oder Attribute vor.

Aufbau einer DTD

Der schematische Aufbau einer DTD sieht wie folgt aus:

```
<!DOCTYPE bienenvolk
[
Definitionen
]>
```

Elemente definieren

Innerhalb der eckigen Klammern befindet sich die Definition der eigentlichen Elemente.

```
<!ELEMENT bienenvolk (heinz, willi, bienemaya)>
```

Das Element mit dem Namen `bienenvolk` enthält die Child-Elemente `heinz`, `willi` und `bienemaya`. Damit ist das erste Element definiert.

```
<!ELEMENT heinz (#PCDATA)>
```

Das Element `heinz` kann alle Zeichen beinhalten. `#PCDATA` erlaubt alle Zeichen innerhalb eines Elements.

Attribute definieren

Zu jedem Element werden direkt im Anschluss eventuell vorhandene Attribute hinzugefügt und deklariert:

```
<!ATTLIST heinz inhalt CDATA #REQUIRED>
```

Das Element `heinz` hat das Attribut `inhalt`. In diesem Attribut dürfen alle Zeichen enthalten sein. `CDATA` entspricht `#PCDATA` und ist der meistverwendete Attributwert. `#REQUIRED` schreibt vor, dass das Attribut auch gesetzt werden muss.

30.4.1 DTDs mit Dreamweaver entwickeln

Dreamweaver unterstützt Sie beim Entwickeln dieser Dokumente und der DTDs. Öffnen Sie dazu das Fenster Ergebnisse • Überprüfung.

Klicken Sie während des Schreibens auf den grünen Pfeil ❶ (siehe Abbildung 30.12), und wählen Sie Aktuelles Dokument überprüfen. Dreamweaver kontrolliert dann das Dokument anhand Ihrer angelegten DTD und prüft gleichzeitig die korrekte Funktion der DTD.

Bereits jetzt erkennt Dreamweaver die von Ihnen erstellten Elemente (Tags) und bietet eine Hilfestellung beim Einfügen an.



Abbildung 30.12 Überprüfung der Dokumentstruktur mit Dreamweaver

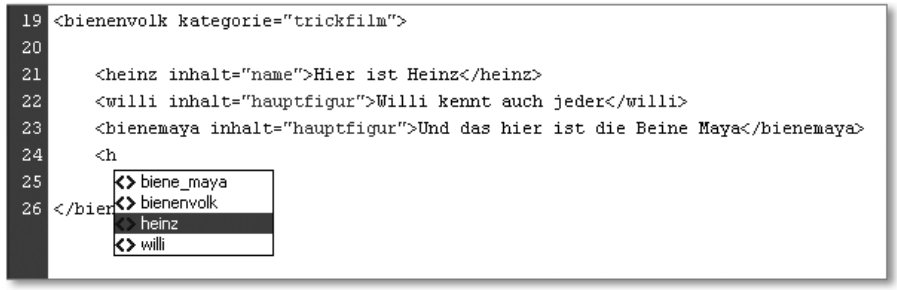


Abbildung 30.13 Eigene Tags in Dreamweaver

30.4.2 Externe DTDs

Genau wie CSS-Stile können auch DTDs extern abgelegt werden. In einem weiteren Schritt können Sie DTDs in die Dreamweaver-Tag-Bibliothek importieren.

```
<!ELEMENT bienenvolk (heinz,willi,bienemaya)>
<!ATTLIST bienenvolk kategorie CDATA #REQUIRED>

   <!ELEMENT heinz (#PCDATA)>
   <!ATTLIST heinz inhalt CDATA #REQUIRED>

   <!ELEMENT willi (#PCDATA)>
   <!ATTLIST willi inhalt CDATA #REQUIRED>

   <!ELEMENT bienemaya (#PCDATA)>
   <!ATTLIST bienemaya inhalt CDATA #REQUIRED>
```

Listing 30.5 Externe DTD »biene_maya.dtd«

Damit bleibt von unserem XML-Dokument nur noch Folgendes übrig:

```
<?xml version="1.0" encoding="iso-8859-1"?>
<?xml-stylesheet type="text/css" href="xml.css"?>
<!-- Beginn DTD -->
<!DOCTYPE bienenvolk [ ]>
```

```
<!-- Ende DTD -->
<bienenvolk kategorie="trickfilm">
   <heinz inhalt="name">
     Hier ist Heinz
   </heinz>

   <willi inhalt="hauptfigur">
     Willi kennt auch jeder
   </willi>
   <bienemaya inhalt="hauptfigur">
     Und das hier ist die Biene Maya
   </bienemaya>
</bienenvolk>
```

Listing 30.6 Verbleibendes XML-Dokument

Innerhalb der Anweisung `DOCTYPE` wird nun auf die externe DTD verwiesen:

```
<!DOCTYPE bienenvolk SYSTEM "biene_maya.dtd">
```

Private und öffentliche DTDs

Das Schlüsselwort `SYSTEM` besagt in diesem Fall, dass es sich um eine externe private DTD handelt, die nur Ihnen oder Ihrer Arbeitsgruppe zur Verfügung steht. Wenn Sie eine DTD öffentlich zugänglich machen wollen, muss die Anweisung lauten:

```
<!DOCTYPE Rootelementname PUBLIC "DTD-Name"  "DTD-URL">
```

30.4.3 Einbinden von DTDs in Dreamweaver

Im Folgenden zeigen wir Ihnen anhand einer Schritt-für-Schritt-Anleitung, wie Sie eine DTD importieren können.

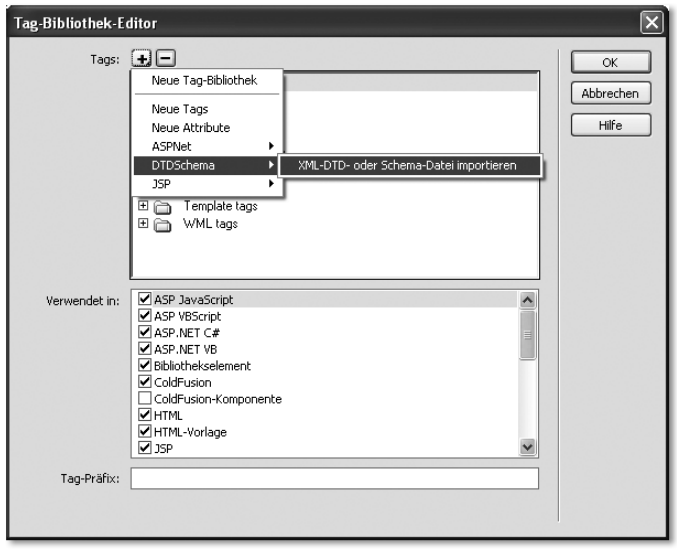
Schritt-für-Schritt: Eine DTD importieren 

1 *Tag-Bibliothek öffnen*

Um DTDs in Dreamweaver einzubinden, können Sie zum Beispiel die Elemente-Definition in die Tag-Bibliothek aufnehmen. Öffnen Sie dazu das Menü BEARBEITEN • TAG-BIBLIOTHEKEN…, und klicken Sie auf das Plussymbol.

2 *Datei importieren*

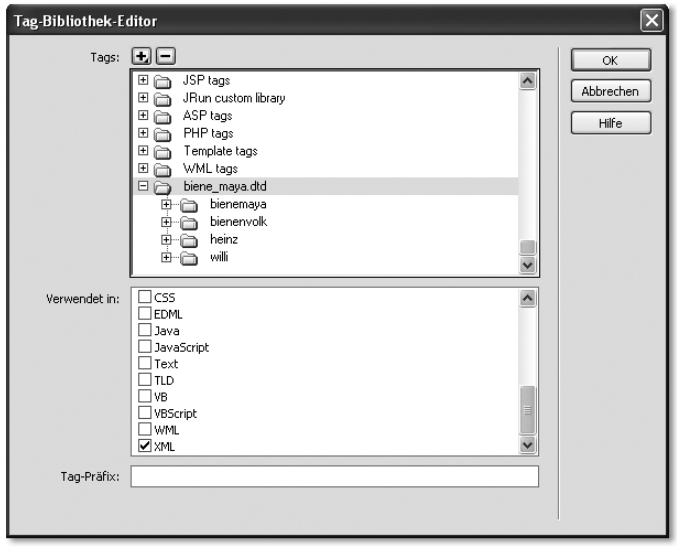
Wählen Sie nun unter DTDSCHEMA die eben angelegte Datei *biene_maya.dtd* aus, und importieren Sie diese in Dreamweaver.



3 *Dokumenttyp zuweisen*

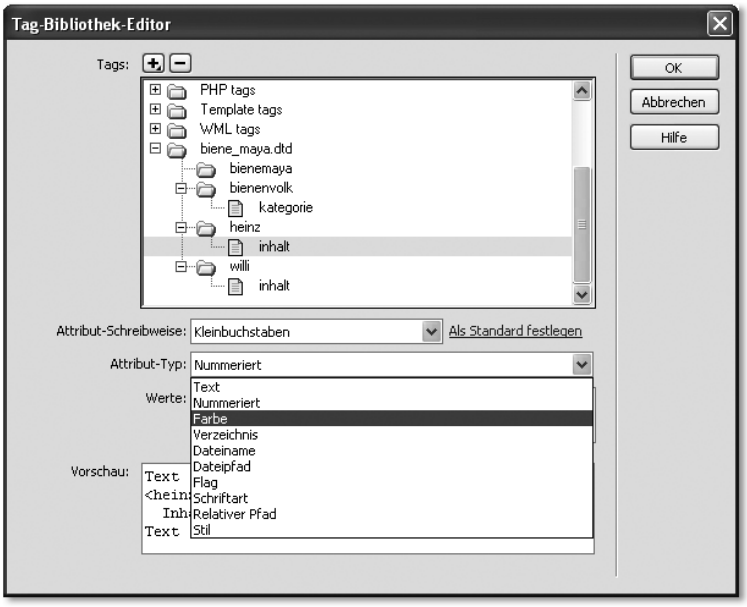
In der Dialogbox stehen Ihnen jetzt die Tags aus der DTD zur Verfügung. Weisen Sie diese einem Dokumenttyp zu (in unserem Beispiel sind es XML-Dokumente). Grundsätzlich können Sie aber derart erstellte Bibliotheken allen Dokumenttypen zuweisen.

Somit ist das Erstellen von externen DTDs eine weitere Möglichkeit, Dreamweaver an Ihre Wünsche anzupassen.



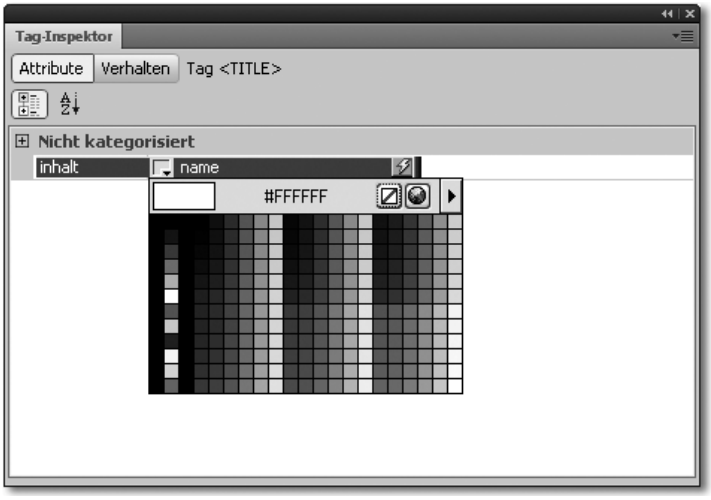
4 *Einstellungen für Attribute*

Bei den Attributen der Tags können Sie die gleichen Einstellungen vornehmen wie bei anderen bereits implementierten Tags.



5 *Anzeige im Tag-Inspektor*

Nach dem Import und der Festlegung der Einstellungen können Sie mit den neuen XML-Tags wie gewohnt im Tag-Inspektor arbeiten.



30.5 XML, XSLT und CSS

Dreamweaver bietet eine visuelle Bearbeitung von XSLT-Dateien und diverse Möglichkeiten, diese Dateien client- oder serverseitig mit XML-Dokumenten zu nutzen. Zunächst folgen jedoch einige weitere theoretische Hintergründe.

30.5.1 XSLT-Einführung

XSLT (*XSL Transformation*) als Teilmenge der Sprache *XSL* (*Extensible Stylesheet Language*) ist eine Programmiersprache zur Umwandlung von XML-Dokumenten in verschiedene Dokumentformate wie XHTML, HTML, PDF, CSV und einige andere. Auch eine Transformation in XML selbst – jedoch in einer neuen XML-Dokumentstruktur – ist möglich.

Die XSLT-Dokumente werden mit dem Dateisuffix *.xsl* gespeichert, was zu Verwirrung führen kann, da es sich dabei nicht um XSL-Dokumente handelt!

XSLT-Transformation

In Dreamweaver wird XSLT zur Transformation von XML-Dokumenten in XHTML oder HTML unterstützt. Mit XSLT werden Regeln für diese Transformation definiert. Das XSLT-Dokument enthält daher Elemente aus XSLT zur Steuerung der Transformation sowie Elemente der Zielsprache, um die korrekte Ausgabe zu gewährleisten.

[»] **Vermischung von Elementen**

Vom Prinzip her ist XSLT mit PHP vergleichbar, wenn PHP zum Generieren von HTML-Dokumenten aus Datenbeständen eingesetzt wird. Auch in diesem Fall sind in den PHP-Dateien Elemente aus PHP und solche aus dem Zielformat vorhanden. Verstehen Sie diese Aussage jedoch nicht falsch – diese »Vermischung« ist auch schon die einzige Gemeinsamkeit zwischen XSLT und PHP.

XPath

Innerhalb eines XSLT- und/oder XML-Dokuments dient XPath, eine weitere Untermenge von XML, zur Adressierung von Teilen oder Bereichen in XML-Dokumenten.

[»] **Weiterführende Informationen**

Eine sehr gute Einführung zu XML, XSLT usw. bietet das Buch »Einstieg in XML« von Helmut Vonhoegen (Galileo Press, ISBN 978-3-8362-1367-7).

Diese ganze Thematik ist so komplex wie interessant, und wir empfehlen Ihnen dringend – wenn Sie professionell im Webumfeld arbeiten –, sich mit dieser Thematik auseinanderzusetzen. Allein der von Adobe betriebene Aufwand, die neuen Features wie Spry in Dreamweaver zu integrieren, zeigt, wie wichtig diese Technik in naher Zukunft sein wird oder längst ist.

Eine der praktischen Anwendungen von Dreamweaver mit XML ist das Einbinden eines Newsfeeds in eine Website. Grundsätzlich gibt es zwei verschiedene Möglichkeiten, mit dieser Technik zu arbeiten. Im Folgenden beschreiben wir beide und erläutern sie jeweils an einem Beispiel.

30.5.2 RSS-Feed einbinden

RSS (*Really Simple Syndication*) ist eine Technologie, die dazu dient, Nachrichten auf einfache Weise in Websites einzubinden oder mit einem Newsreader, ähnlich einer E-Mail, lesen zu können. Mittlerweile existieren gleich mehrere Standards für RSS-Feeds. Wir haben die Funktionen mit verschiedenen Standards getestet – es funktioniert offensichtlich mit allen.

Für die folgenden Beispiele haben wir einen kleinen RSS-Feed als XML-Dokument angelegt. Dieses XML-Dokument wird lokal, also direkt auf dem Server abgelegt, auf dem auch das XSLT-Dokument läuft.

Auch bei der serverseitigen XSLT-Transformation werden wir mit einem lokalen XML-Dokument arbeiten – selbst wenn die Möglichkeit bestünde, dies über einen externen Server einzubinden, zum Beispiel über den eines Nachrichtendienstes.

Unser Newsfeed ist eine XML-Datei mit folgender Struktur und einem auf der Spezifikation RSS 0.91 basierenden Inhalt.

```
<?xml version="1.0" encoding="ISO-8859-1"?>

<rss version="2.0">

<channel>
    <title>DREAMWEAVER NEWS GERMANY</title>
   <link>http://www.dreamweaverforum.de</link>
   <description>Newsfeed Dreamweaverforum Deutschland</description>
   <language>de</language>
   <item>
      <title>Neuerscheinungen 2010 - Dreamweaver CS5 von
      Richard Beer und Susann Gailus</title>
      <link>http://www.dreamweaverforum.de</link>
      </item>
```

```
   <item>
      <title>Dreamweaverforum im neuen Design</title>
      <link>http://www.dreamweaverforum.de</link>
      </item>

   <item>
      <title>Dreamweaver CS5 unterstützt XSLT Erstellung</title>
      <link>http://www.dreamweaverforum.de</link>
      </item>

   <item>
      <title>Dreamweaver CS5: Neue visuelle Layoutmöglichkeiten
      für XSLT</title>
      <link>http://www.dreamweaverforum.de</link>
      </item>

   <item>
      <title>Studio CS5 zum Verkauf freigegeben</title>
      <link>http://www.dreamweaverforum.de</link>
      </item>

</channel>
</rss>
```

Listing 30.7 Newsfeed im XML-Format

30.5.3 XSLT-Fragmente und ganze XSLT-Seiten

In Dreamweaver können Sie beim Neuanlegen eines XSLT-Dokuments entscheiden, ob Sie eine ganze XSLT-Seite erzeugen wollen oder nur ein Fragment. *XSLT-Fragmente* werden in bestehende Dokumente eingebunden und enthalten keinen eigenen Dokumentkörper (Head und Body). Sie können zusätzlich jedes bestehende HTML- oder XHTML-Dokument in eine *XSLT-Seite* konvertieren, indem Sie aus dem Menü Datei • Konvertieren • XSLT 1.0 auswählen.

Wenn Sie nur Fragmente anlegen, müssen Sie bedenken, dass CSS-Stile teilweise im Head eines Dokuments abgelegt werden. Sie müssen also die Definition der XSS-Stile oder den Verweis auf eine CSS-Datei in das Dokument legen, in das das Fragment eingebunden werden soll.

30.5.4 Clientseitige XSLT-Transformation

Bei der clientseitigen Transformation wird – wie der Name schon sagt – die XSLT-Transformation auf dem Client bzw. mit dem Browser des Clients durchgeführt. Dies wird von den neueren Browsern auch unterstützt, von älteren hingegen

nicht. Neben der Notwendigkeit, in die XML-Quelldatei eingreifen zu müssen, ist dies einer der Nachteile dieser Methode. Der Vorteil liegt darin, dass Sie keinerlei serverbasierte Anwendungen benötigen, um die Transformation durchzuführen.

XML-Dateien lokal ablegen

Die für clientseitige Transformationen verwendeten XML- und XSLT-Seiten müssen im gleichen Verzeichnis abgelegt werden. Ansonsten sucht der Browser beim Lesen der XML-Datei nach der XSLT-Seite für die Transformation, kann aber die durch relative Links definierten Elemente in der XSLT-Datei nicht finden.

In Abbildung 30.14 sehen Sie die XML-Quelldatei unseres RSS-Feeds in der Browseransicht ohne jede Gestaltung oder Transformation. Neue Browser erkennen RSS-Feeds automatisch und bieten diese als Abonnement an.



Abbildung 30.14 Browseransicht des XML-Dokuments

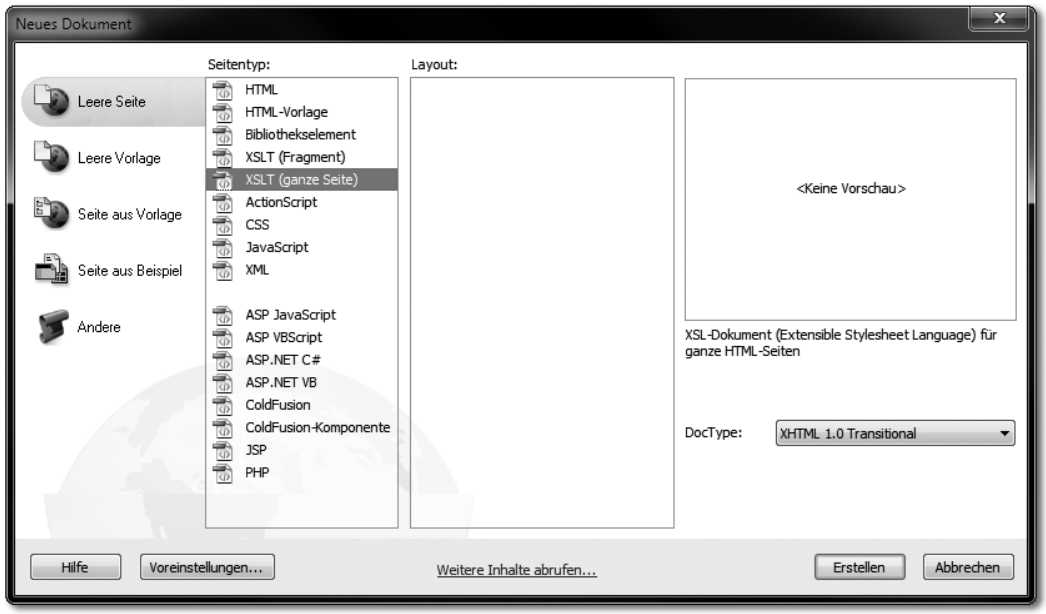


Abbildung 30.15 Neue ganze XSLT-Seite anlegen

Nachdem Sie eine Auswahl getroffen haben – wie in unserem Beispiel eine ganze XSLT-Seite –, erscheint die Aufforderung, die XML-Quellen anzugeben. Sie können zwischen lokalen Quellen und Quellen aus dem Internet wählen.

Wir haben im Beispiel eine lokale XML-Datei angegeben.

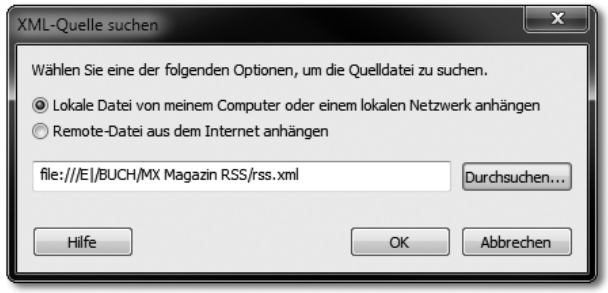


Abbildung 30.16 XML-Quelle an XSLT-Datei anbinden

Wie Sie in Abbildung 30.17 sehen, werden die Elemente der Datei im Bedienfeld BINDUNGEN so ähnlich angezeigt, wie Sie es von Datenbankabfragen kennen. Die Handhabung dieser Elemente unterscheidet sich bei Dreamweaver nur geringfügig von der der Elemente aus einer Datenbank.

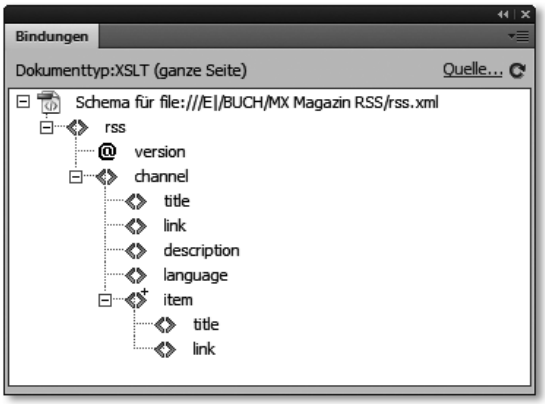


Abbildung 30.17 Bedienfeld »Bindungen« mit Datenfeldern aus XML

In dem XSLT-Dokument können Sie mit allen gewohnten HTML-Elementen arbeiten. Das Layouten dieser Dokumente gleicht demjenigen dynamischer Dokumente. Wenn Sie jedoch einen Blick in den Quelltext werfen, werden Sie feststellen, dass sich die Syntax erheblich unterscheidet.

Wiederholte und bedingte Bereiche

Auch bei XSLT-Dokumenten können wiederholte und bedingte Bereiche eingefügt werden. Wenn Sie in der Einfügeleiste das Register XSLT öffnen, stehen Ihnen einige – wenn auch wenige – Werkzeuge zur Ausgabesteuerung zur Verfügung.

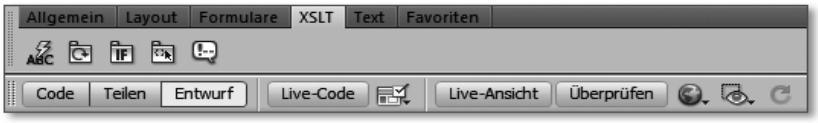


Abbildung 30.18 XSLT-Einfügeleiste

Um die Arbeitsweise zu verdeutlichen, zeigt Abbildung 30.19 einen eingefügten wiederholten Bereich. Es ist eine Tabellenzeile, die mit jedem Eintrag des RSS-Feeds wiederholt werden soll ❶.

Am einfachsten können solche Arbeiten im Quelltext durchgeführt werden, indem Sie die ganze Tabellenreihe auswählen und dann auf Wiederholten Bereich einfügen in der Einfügeleiste klicken. Im daraufhin erscheinenden XPath-Ausdrucksgenerator können Sie dann den Bereich auswählen, der für die Häufigkeit der Wiederholungen zuständig ist. In der Feldliste werden wiederholt angelegte Bereiche durch ein kleines Plussymbol dargestellt. Die Steuerung dieser Bereiche sowie eventuelle Bedingungen werden mit XPath geschrieben.

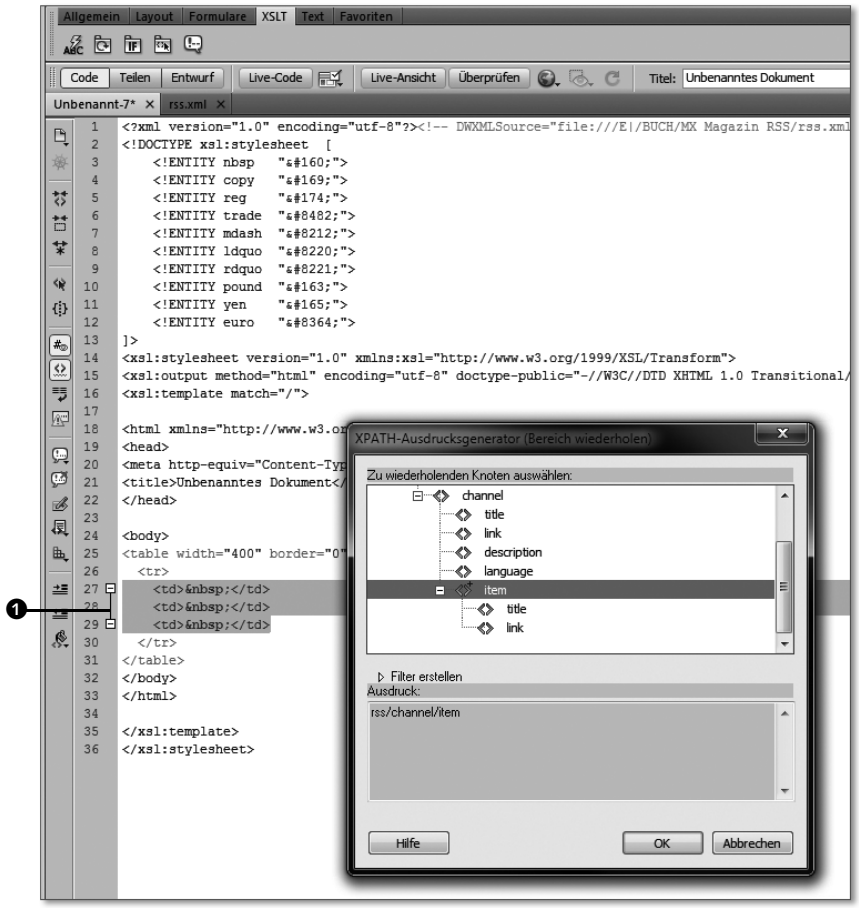


Abbildung 30.19 Wiederholte Bereiche hinzufügen

In der Layoutansicht sehen Sie den wiederholten Bereich so, wie Sie es vom Arbeiten mit dynamischen Dokumenten her gewohnt sind (siehe Abbildung 30.20).

Ziehen Sie noch die gewünschten Elemente in den wiederholten Bereich. In der Browservorschau können Sie zu diesem Zeitpunkt bereits den Erfolg Ihrer Arbeit bestaunen.

Häufig werden in RSS-Feeds nur eine Kurzbeschreibung und ein Titel eines weiterführenden Artikels angezeigt. Den eigentlichen Artikel können Sie lesen, wenn Sie auf die Headline der News klicken. In unserem Beispiel haben wir die Headline mit dem Feld *link* verknüpft (siehe Abbildung 30.21). Dies geschieht auf die gleiche Weise wie bei dynamischen Dokumenten (siehe Abbildung 30.22). Auch XML-Dokumente werden in XSLT als Datenquellen angegeben. Den einzelnen Elementen können wie gewohnt CSS-Stile zugewiesen werden.

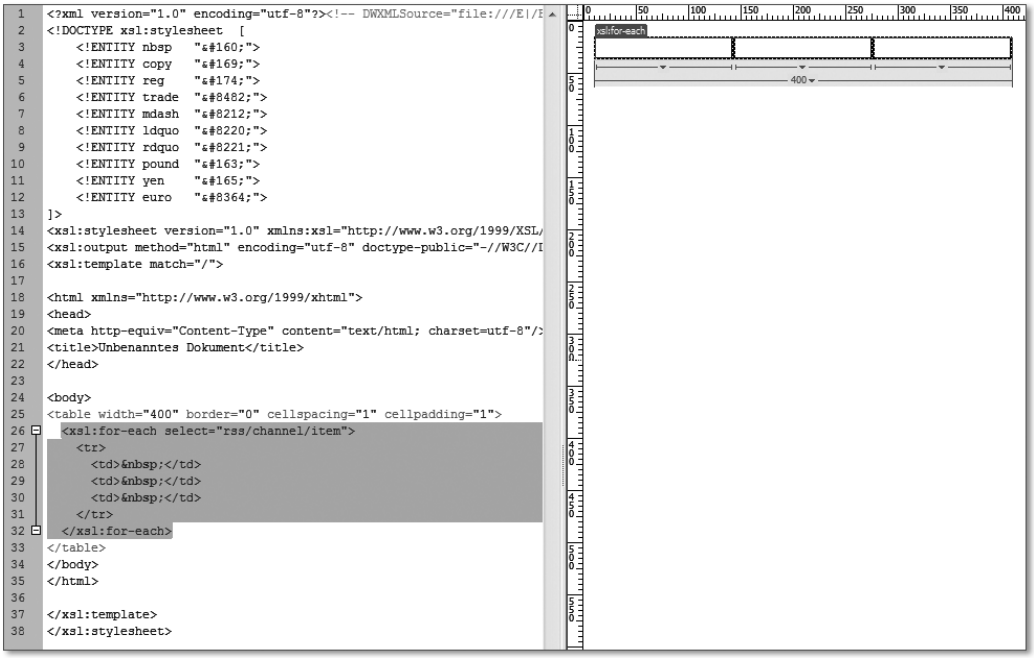


Abbildung 30.20 Wiederholter Bereich in der Layoutansicht



Abbildung 30.21 Wiederholte Bereiche

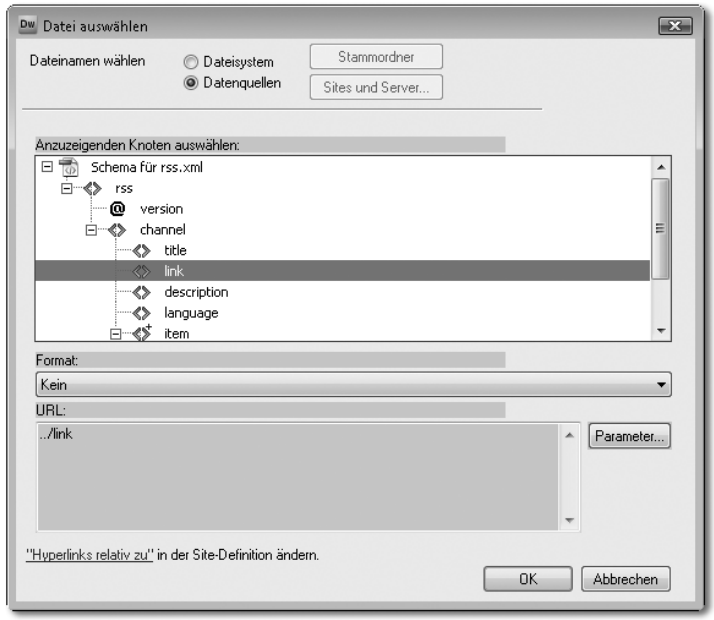


Abbildung 30.22 Hyperlink aus Datenquelle

[!] **Vorsicht Falle!**

Zu diesem Zeitpunkt ist das Dokument bzw. die Dokumentstruktur jedoch keineswegs fertig. Hier lauert eine kleine Falle. Wenn Sie über F12 in die Browservorschau wechseln, sehen Sie ein fertig gestaltetes Dokument (siehe Abbildung 30.23). Dies ist jedoch – wie Sie in der Adressleiste des Browsers links sehen – ein temporäres Dokument und nicht das Originaldokument rechts. Wenn Sie das Original direkt aufrufen, sehen Sie immer noch die komplette Dokumentstruktur. Hier übernimmt Dreamweaver die Transformation. Um ein wirklich funktionsfähiges Dokument zu erhalten, müssen noch die im Folgenden beschriebenen Schritte durchgeführt werden.

In der XML-Datei – *nicht in der XSLT-Datei* – muss jetzt noch das XSLT-Dokument angehängt werden. Letztlich ist die XSLT-Datei ja sozusagen eine erweiterte Stilvorlage. Genauso wird diese auch in das XML-Dokument eingebunden.

Rufen Sie in Ihrer XML-Quelle aus dem Menü BEFEHLE • XSLT-STYLESHEET ANHÄNGEN auf (siehe Abbildung 30.24), und tragen Sie in der folgenden Dialogbox das Dokument ein (siehe Abbildung 30.25).

In Listing 30.8 sehen Sie, wie das XSLT-Dokument in die XML-Datei eingebunden wird. Diese Notwendigkeit, eine Referenz direkt in die XML-Quelle zu schreiben, zeigt auch, warum dieses Verfahren zum Einbinden externer RSS-Feeds nur geeignet ist, wenn Sie direkten Zugriff auf die Quelldateien haben.

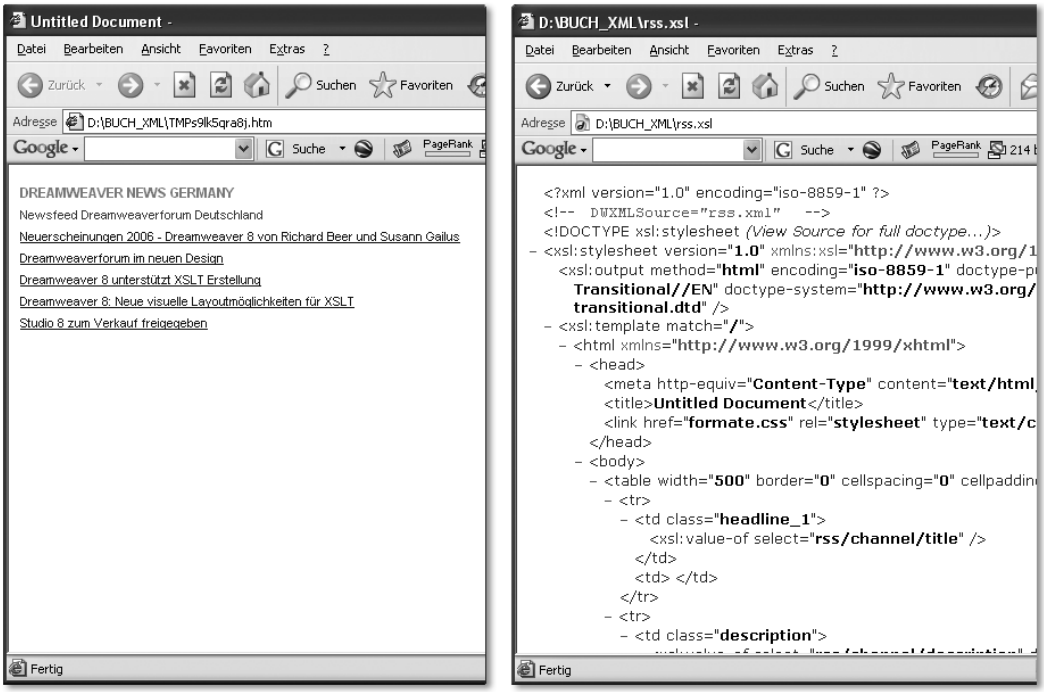


Abbildung 30.23 Browservorschau und »echtes« Dokument

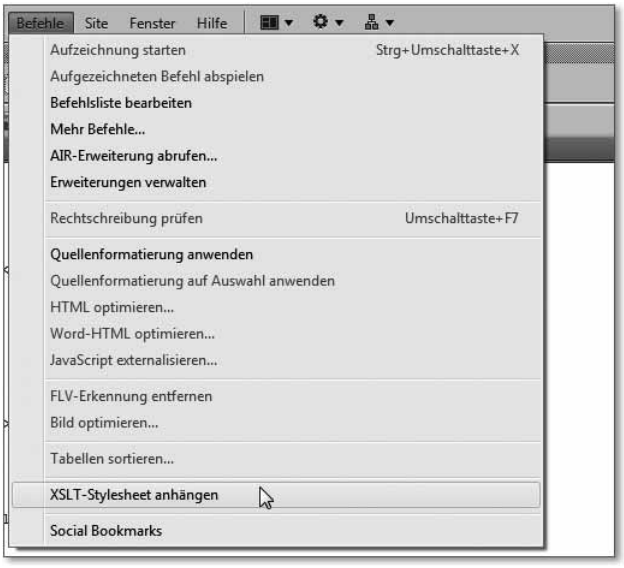


Abbildung 30.24 Menüeintrag XSLT-Stylesheet anhängen

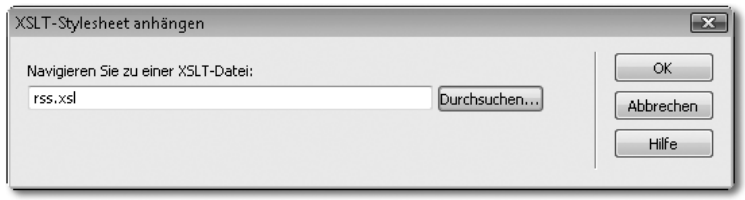


Abbildung 30.25 XSLT an XML anhängen

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<?xml-stylesheet href="content.xsl" type="text/xsl"?>
<channel>
    <title>DREAMWEAVER NEWS GERMANY</title>
   <link>http://www.dreamweaverforum.de</link>
   <description>Newsfeed Dreamweaverforum Deutschland</description>
   <language>de</language>
   <item>
      <title>Neuerscheinungen 2010 –
      Dreamweaver CS5 von Richard Beer und Susann Gailus</title>
      <link>http://www.dreamweaverforum.de</link>
      </item>

   <item>
      <title>Dreamweaverforum im neuen Design</title>
      <link>http://www.dreamweaverforum.de</link>
      </item>

   <item>
      <title>Dreamweaver CS5 unterstützt XSLT Erstellung</title>
      <link>http://www.dreamweaverforum.de</link>
      </item>

   <item>
      <title>Dreamweaver CS5: Neue visuelle Layoutmöglichkeiten für
      XSLT</title>
      <link>http://www.dreamweaverforum.de</link>
      </item>

   <item>
      <title>Studio CS5 zum Verkauf freigegeben</title>
      <link>http://www.dreamweaverforum.de</link>
      </item>

</channel>
```

Listing 30.8 Einbindung der XSLT-Datei in ein XML-Dokument

Um das fertige Dokument zu sehen, müssen Sie direkt das XML-Dokument aufrufen. Die XSLT-Datei wird nicht im Browser aufgerufen, sondern automatisch vom XML-Dokument angefordert.

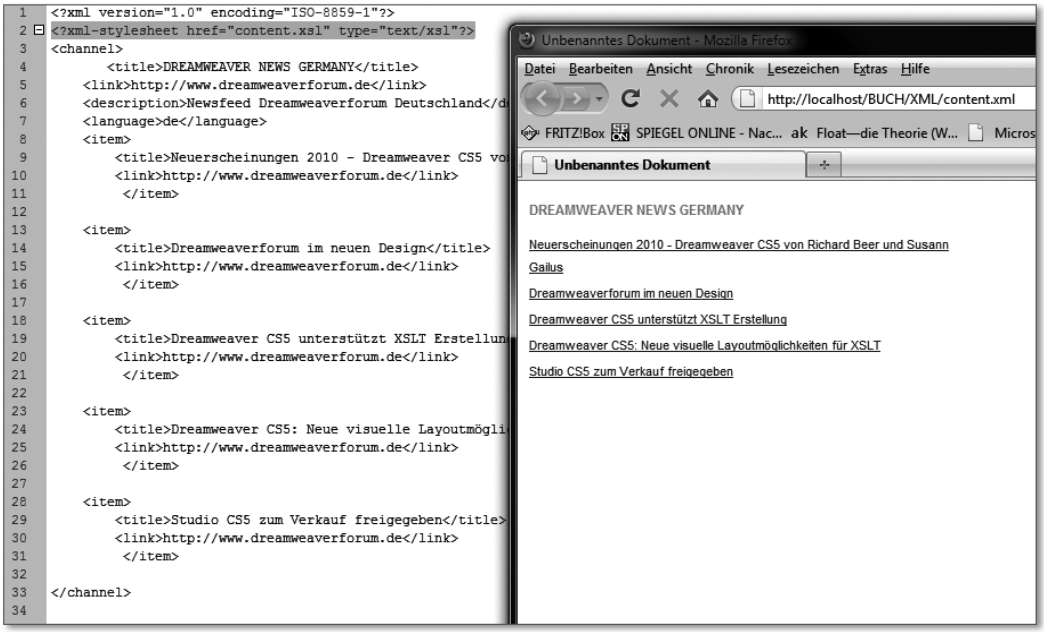


Abbildung 30.26 Quellcode und Browseransicht der XML-Datei

30.5.5 Serverseitige XSLT-Transformation mit PHP

Die serverseitige Transformation ist etwas aufwendiger, hat aber den deutlichen Vorteil, dass sie ohne Zutun des Browsers erfolgt. Dem Client wird nur die fertig transformierte Datei übergeben. Zusätzlich können Sie auf diese Weise externe RSS-Feeds in Ihre Seiten einbinden, ohne die originale XML-Datei zu verändern. Für externe Nachrichten usw. ist dies also bestens geeignet.

Die Vorgehensweise gleicht zunächst derjenigen, die im Abschnitt über clientseitige XSLT-Transformation beschrieben wurde. Sie müssen jedoch dem XML-Dokument kein XSLT-Dokument zuweisen. Bedenken Sie, dass vorher die Site mit dem lokalen Testserver definiert werden muss.

Als nächsten Schritt erstellen Sie zur Vorbereitung ein komplett leeres PHP-Dokument. Es muss also der *gesamte* Quelltext darin gelöscht und stattdessen die PHP-Tags (`<?php ?>`) eingefügt werden.

[!] **Dreamweaver-Fehler**

Wenn Sie, wie von Dreamweaver aufgefordert, das Dokument wirklich vollständig leeren, müssen Sie die Arbeitsschritte zur XSL-Transformation zweimal ausführen. Im ersten Schritt wird nur der Pfad zur externen Datei angelegt, im zweiten Schritt folgen dann auch die korrekten Parameter.

Serververhalten XSL-Transformation

In der Einfügeleiste finden Sie das Icon zum Anlegen des Serververhaltens XSL Transformation ❶.



Abbildung 30.27 XSL Transformation ausführen

Wenn Sie darauf klicken, erscheint eine Dialogbox, in der Sie den Pfad zur XSLT-Datei angeben können. Wenn diese mit Dreamweaver erstellt wurde, wird der Pfad zur XML-Quelldatei auch korrekt eingetragen, da Dreamweaver ihn als Kommentar in der XSLT-Datei vermerkt.



Abbildung 30.28 Pfade einfügen

In der Layoutansicht wird nun bereits die komplette Struktur des XSLT-Dokuments sichtbar, und Sie können das Dokument auch in der Live-Ansicht betrachten (siehe Abbildung 30.29). Layoutänderungen können Sie hier allerdings nicht vornehmen. Diese müssen in der XSLT-Datei erfolgen.

Ein Blick in den Quelltext verrät, dass auf eine externe Datei zugegriffen wird:

```
<?php
//XMLXSL Transformation class
require_once('includes/MM_XSLTransform/MM_XSLTransform.class.php');
?>
```

```
<?php
$mm_xsl = new MM_XSLTransform();
$mm_xsl->setXML("content.xml");
$mm_xsl->setXSL("content.xsl");
echo $mm_xsl->Transform();
?>
```

Listing 30.9 Quelltext einer serverseitigen XSLT-Transformation

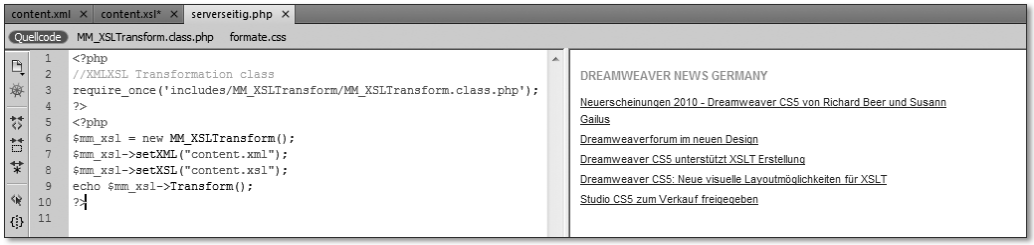


Abbildung 30.29 XSL Transformation in der Live-Ansicht

Im von Dreamweaver automatisch erstellten Verzeichnis *include* liegt ein weiterer Ordner *MM_XSLTransformation*. Diesen müssen Sie beim Bereitstellen der Site inklusive Inhalt mit auf den Webserver kopieren. Wenn Sie die gesamte Site bereitstellen, wird der Ordner automatisch auf den Server kopiert.

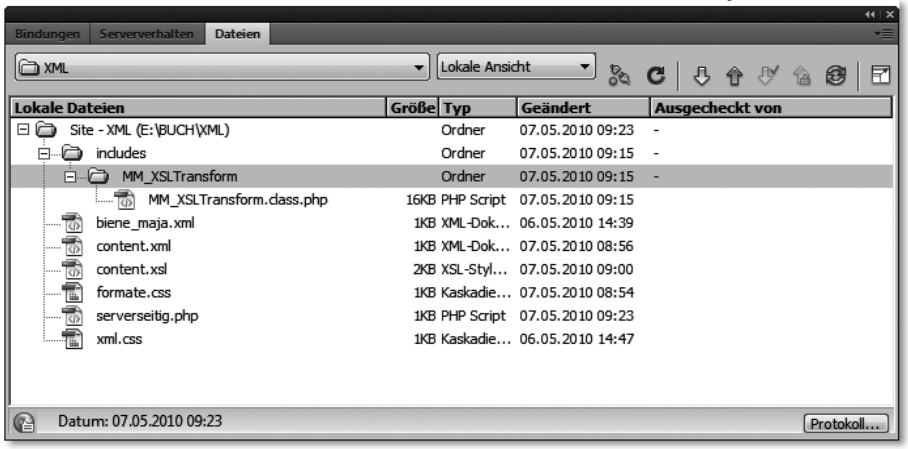


Abbildung 30.30 Automatisch erstellter Ordner

Fehlende Bibliotheken oder Klassen

Falls in einer Fehlermeldung auf Ihrem Server nach nicht vorhandenen Bibliotheken oder Klassen gefragt wird, wenden Sie sich bitte an Ihren Provider, so dass diese eventuell aktiviert werden.

Sie wollen Daten nachladen, ohne eine Website erneut aufzurufen, oder Ihren Benutzern ohne Benutzeraktion immer aktuelle Daten präsentieren. Geht nicht? Geht doch! Dreamweaver und das Spry-Framework machen es möglich.

31 Spry und XML

In einigen vorherigen Kapiteln haben Sie Spry bereits kennengelernt. Es handelt sich dabei genau genommen um eine Ajax-Bibliothek von Adobe.

31.1 Was ist Ajax?

Ajax ist eine Abkürzung für *Asynchronous JavaScript and XML*. Ajax ist die Bezeichnung für eine Technik zur asynchronen Datenübertragung zwischen einem Webserver und dem Client (Browser). Eine Basis von Ajax bildet das *XMLHttpRequest*-Objekt. Damit können Bestandteile einer XML-Datei in Bereiche einer HTML-Datei geladen werden, ohne das gesamte Dokument zu aktualisieren.

Dreamweaver bietet nun in Kombination mit dem Spry-Framework eine einfache Möglichkeit, diese Technik ohne tiefergehende Programmierkenntnisse zu nutzen.

Den theoretischen Hintergrund zu Ajax möchten wir an dieser Stelle nicht vertiefen. Sie finden bei Galileo Press und im Internet Bücher und Dokumente zu Ajax.

31.2 XML-Daten anbinden

In der gleichen Weise, wie Sie eine Datenbankabfrage anlegen, müssen Sie auch hier zunächst eine »Bindung« erstellen. Als Quelle kann eine beliebige XML-Ressource dienen. Den Aufbau der Beispiel-XML-Datei sehen Sie im folgenden Listing. Sie finden die Datei unter *Beispiele/Spry_XML/verkauf.xml* auf der DVD zum Buch.



```
<?xml version="1.0" encoding="UTF-8"?>
<angebote>
   <artikel id="1">
```

```
      <typ>Hose</typ>
      <beschreibung>Eu liber Serio do demens Monitio dono algor,
incrementum indulgens.</beschreibung>
      <preis>70 Euro</preis>
      <farbe>blau</farbe>
   </artikel>
   <artikel id="2">
      <typ>Hemd</typ>
      <beschreibung>Ala quo os malus Effor conatus, quies pes roto
      munio veneficus admonitio.</beschreibung>
      <preis>80 Euro</preis>
      <farbe>rot</farbe>
   </artikel>
   <artikel id="3">
      <typ>Jacke</typ>
      <beschreibung>Se specialitas lumen. Os disperdo aut Quercetum
      domus Hospitium.</beschreibung>
      <preis>160 Euro</preis>
      <farbe>grau</farbe>
   </artikel>
</angebote>
```

Listing 31.1 Aufbau der XML-Datei »verkauf.xml«

Klicken Sie in der Einfügeleiste auf das Icon SPRY-DATENSATZ ❶. Alternativ kann ein Spry-Datensatz auch über das Bedienfeld BINDUNGEN angelegt werden.

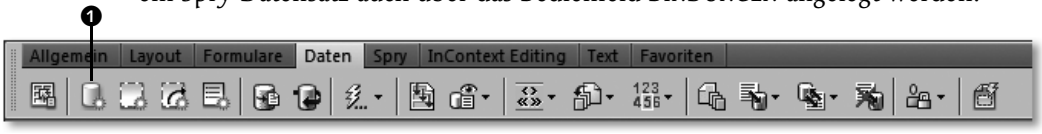


Abbildung 31.1 Die Symbolleiste »Daten«

In Dreamweaver CS5 wird das Anlegen von Spry-Datensätzen durch einen Assistenten unterstützt. Wählen Sie im Dialog DATENQUELLE ANGEBEN als Datentyp ❷ *XML* aus. Unter DATENSATZNAME ❺ wird eine Bezeichnung für die Datensatzbindung angegeben. In unserem Beispiel belassen wir die Bezeichnung bei der von Dreamweaver vergebenen Benennung *ds1*. Im nächsten Schritt wählen Sie die Datendatei ❸ aus. Die Zeilenelemente werden nun in einem Fenster angezeigt. Da sich die wiederholten Elemente im Zeilenelement ARTIKEL befinden, muss ARTIKEL auch angewählt werden. In der DATENVORSCHAU ❹ erscheint eine Vorschau der zu erwartenden Datensätze.

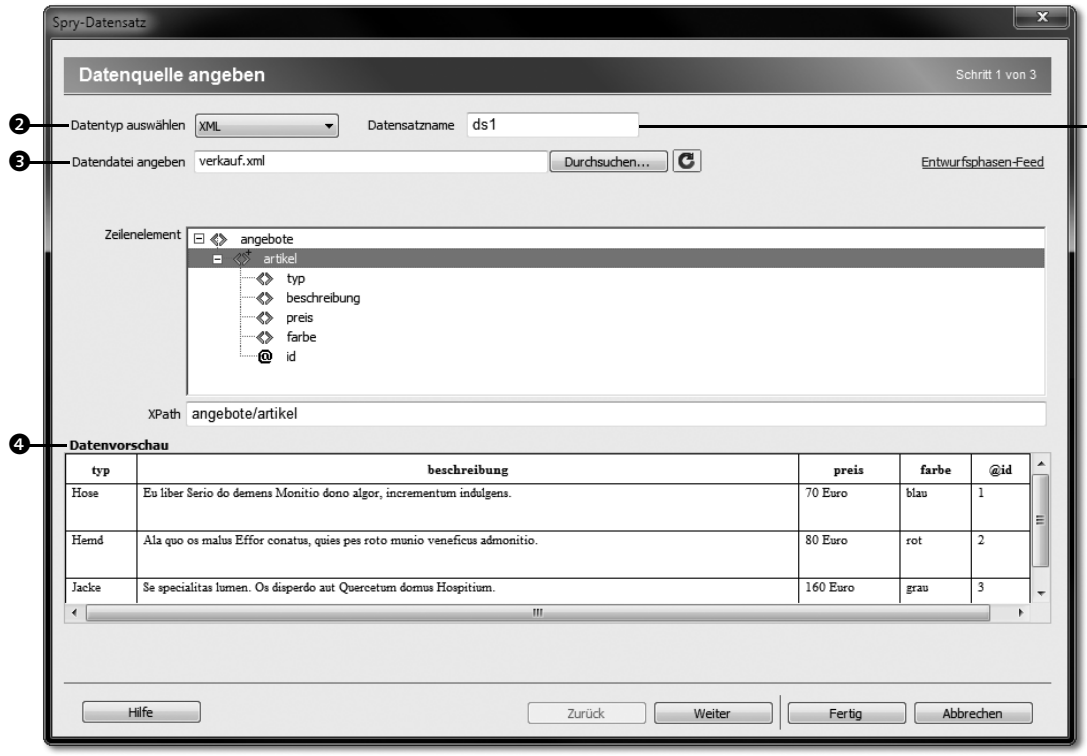


Abbildung 31.2 Dialog »Datenquelle angeben«

Wenn Sie auf WEITER geklickt haben, zeigt Ihnen Dreamweaver im Bereich DATENVORSCHAU nun die Inhalte des XML-Dokuments in einer tabellarischen Vorschau an. Als weitere Option ist hier die Spaltensortierung ❻ einstellbar, ebenso wie eventuelle Dubletten ❼ ausgefiltert werden können.

Das eigentliche Highlight von XML- und Spry-Datensätzen ist, dass Inhalte einer Website laufend – ohne Browser-Refresh – aktualisiert werden können. Um diese Option zu nutzen, müssen Sie den DATENCACHE DEAKTIVIEREN ❽ und einen Zeitwert eingeben, nach dem die DATEN AUTOMATISCH AKTUALISIERT ❾ werden sollen.

Im letzten Schritt stehen Ihnen vier verschiedene Auswahlmöglichkeiten zu den Einfügeoptionen zur Verfügung. Die Beschreibungen im Dialog sind sehr gut, so dass wir uns an dieser Stelle sparen, näher darauf einzugehen. Jedes einzelne Element finden Sie auch im weiteren Verlauf dieses Kapitels.

Da es sich um eine Datendarstellung handelt, haben wir ein einfaches Tabellenlayout gewählt. Eine Vorschau sehen Sie in Abbildung 31.5.

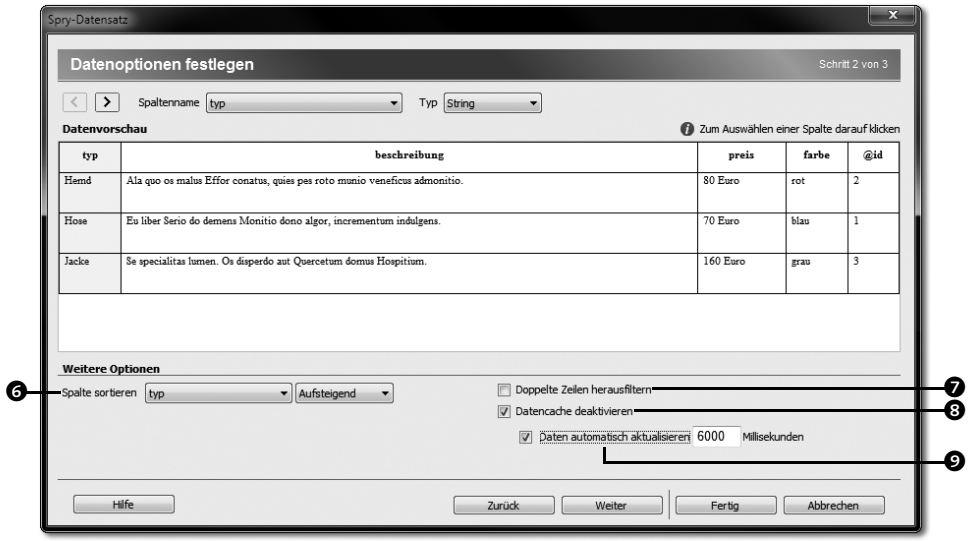


Abbildung 31.3 Optionen einstellen

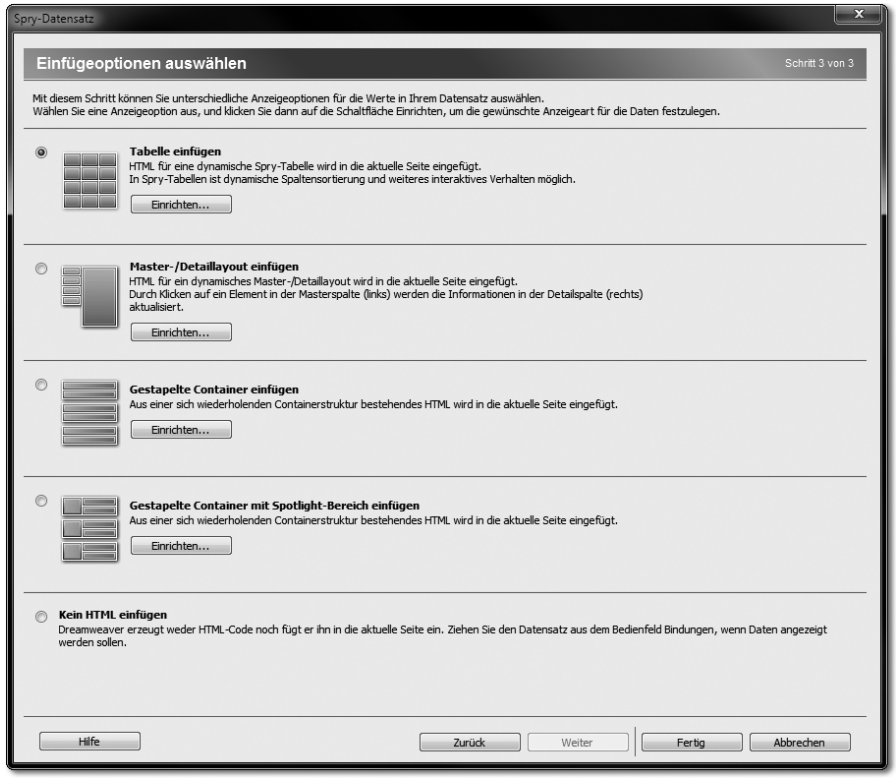


Abbildung 31.4 Weitere Einstellungen für den Spry-XML-Datensatz

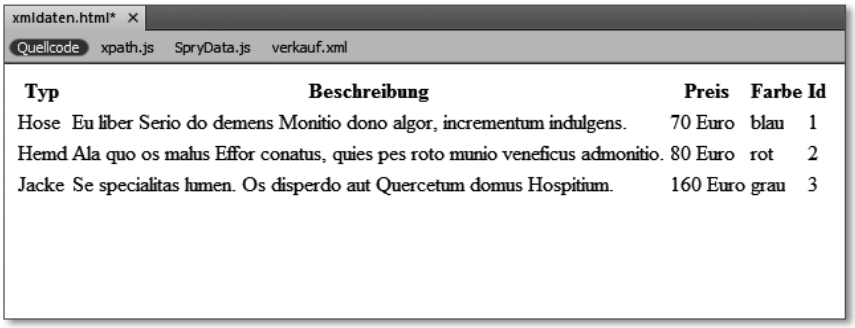


Abbildung 31.5 Vorschau der XML-Daten

Im Bedienfeld BINDUNGEN sehen Sie, dass wie auch bei einer Datenbankanbindung die einzelnen Datenfelder angezeigt werden.

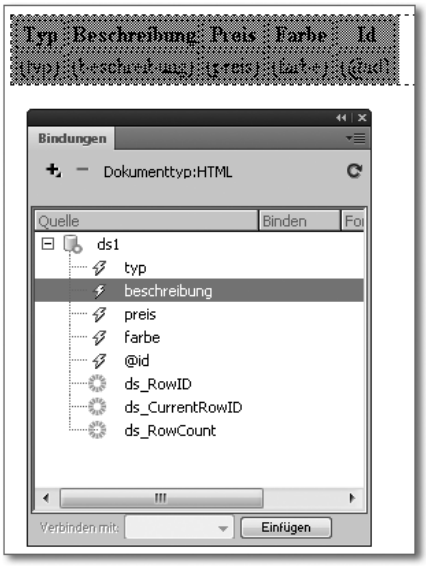


Abbildung 31.6 Anzeige der einzelnen Datenfelder im Bedienfeld »Bindungen«

Ein Blick in den Quelltext zeigt uns, welche Änderung Dreamweaver vorgenommen hat:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
xmlns:spry="http://ns.adobe.com/spry">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
```

```
<title>XML Daten</title>
<script src="SpryAssets/xpath.js" type="text/javascript"></script>
<script src="SpryAssets/SpryData.js" type="text/javascript"></script>
<script type="text/javascript">
<!--
var ds1 = new Spry.Data.XMLDataSet("verkauf.xml",
"angebote/artikel");
//-->
</script>
</head>
<body>
<div spry:region="ds1">
  <table>
    <tr>
      <th spry:sort="typ">Typ</th>
      <th spry:sort="beschreibung">Beschreibung</th>
      <th spry:sort="preis">Preis</th>
      <th spry:sort="farbe">Farbe</th>
      <th spry:sort="@id">Id</th>
    </tr>
    <tr spry:repeat="ds1">
      <td>{typ}</td>
      <td>{beschreibung}</td>
      <td>{preis}</td>
      <td>{farbe}</td>
      <td>{@id}</td>
    </tr>
  </table>
</div>
</body>
</html>
```

Listing 31.2 Der Quelltext nach dem Einfügen der Datensätze

Wie kommt man an XML-Daten?

Jetzt stellt sich sicher der eine oder andere die Frage: »Wie komme ich denn nun an XML-Daten?« Die Antwort ist im Prinzip ganz einfach: durch jede Anwendung, die XML-Daten schreiben kann! Das kann ein CMS sein, ein Export aus Office, den Sie von Hand erledigen, ein CRM-System oder auch eine Datenbank, die direkt XML ausgibt. Im einfachsten Fall sind es kleine, von Hand erstellte XML-Dateien.

31.3 Spry-Bereiche

In vorherigen Abschnitt haben wir Spry-Bereiche vollständig mit dem Assistenten definiert. In den nun folgenden Schritten werden wir diese individuell von Hand anlegen. Einen Spry-Datensatz müssen Sie vorher jedoch trotzdem erstellen. Wählen Sie hierbei allerdings KEIN HTML EINFÜGEN ❶ im letzten Schritt des Assistenten aus.

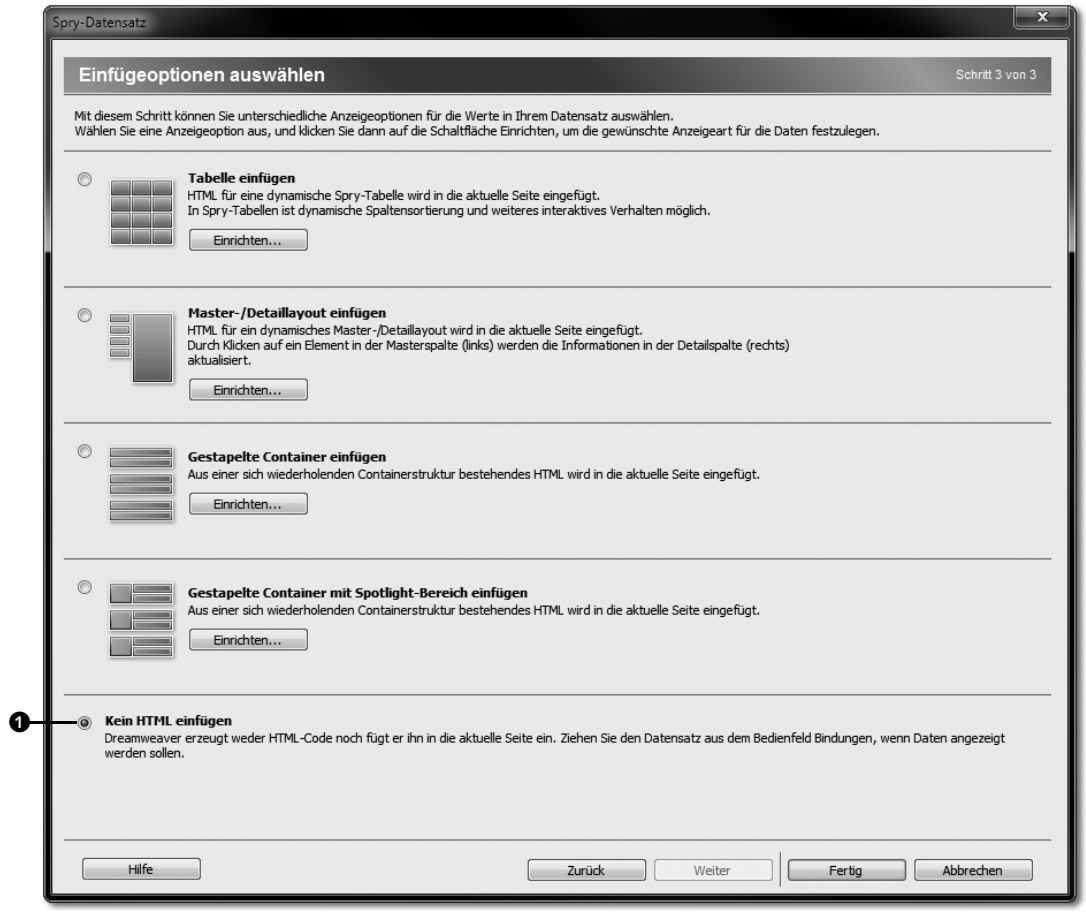


Abbildung 31.7 Assistent ohne HTML-Elemente

Um Ihrem Dokument Daten hinzuzufügen, wird zunächst innerhalb des HTML-Dokuments ein Spry-Bereich definiert. Positionieren Sie den Cursor an der Stelle, an der Sie den Bereich einfügen möchten, und klicken Sie auf das Icon SPRY-BEREICH. Für alle nachfolgenden Techniken müssen Sie *immer* erst einen Spry-Bereich festlegen. Der Spry-Bereich ist die Basis aller nachfolgenden Schritte.

Anschließend haben Sie die Wahl, mit welchem HTML-Tag (Container) der Bereich erstellt werden soll. Die Vorgabe bei CONTAINER ist hier das `<div>`-Tag. In den meisten Fällen ist das auch die beste Wahl. Mit `<div>` und CSS stehen Ihnen die meisten Formatierungsoptionen zur Verfügung.

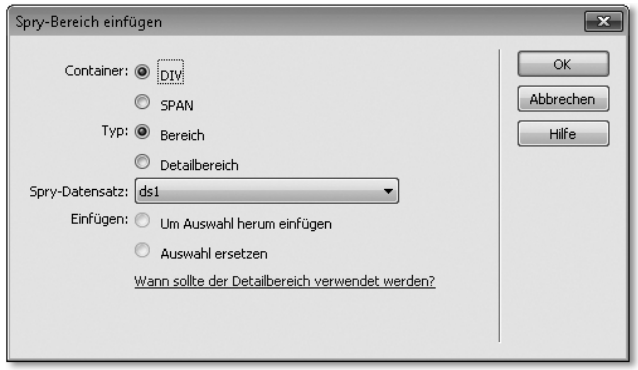


Abbildung 31.8 Dialog zum Einfügen von Spry-Bereichen

Nach einer Bestätigung mit OK kann der gewünschte Datensatz aus dem Bedienfeld BINDUNGEN in den Spry-Bereich gezogen werden.

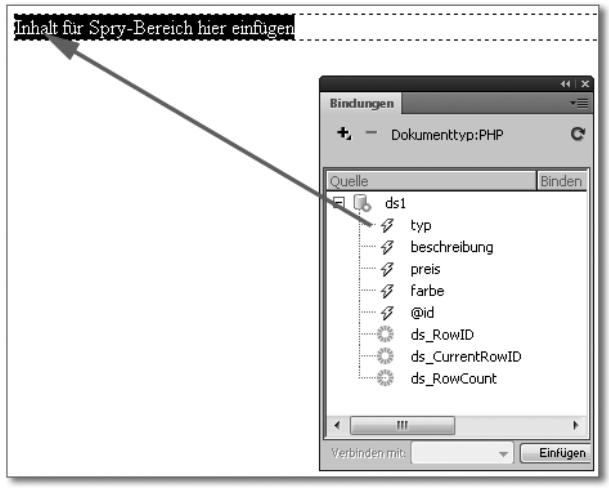


Abbildung 31.9 Ziehen eines Datenfeldes in den Spry-Bereich

Das Datenfeld wird innerhalb des Spry-Bereichs hellblau hinterlegt angezeigt. Überprüfen Sie nun das Ergebnis in den verschiedenen Browsern.

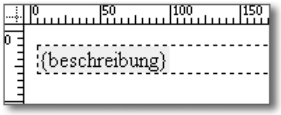


Abbildung 31.10 Anzeige eines Datenfeldes im Spry-Bereich

Vorschau im Browser

Die Vorschau im Browser zeigt uns das gewünschte Datenfeld an.

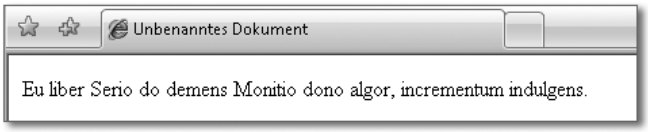


Abbildung 31.11 Darstellung der Daten in Firefox

31.3.1 Spry-Wiederholungen

Wie Sie eben festgestellt haben, wird bei der bisherigen Vorgehensweise der Datenausgabe nur der erste Datensatz angezeigt. Wie auch bei Datenbanken möchte man hier oft mehrere Datensätze ausgeben. Hierfür ist die Spry-Wiederholung gedacht.

Bevor Sie mit einer Spry-Wiederholung arbeiten, müssen Sie zunächst, wie zu Beginn des Abschnitts gezeigt, einen Spry-Bereich anlegen.

In diesen Spry-Bereich wird die Spry-Wiederholung eingefügt. Der Spry-Bereich umschließt die Spry-Wiederholung. Im Quelltext sieht dies wie folgt aus:

```
<div spry:region="ds1">Inhalt für Spry-Bereich hier einfügen</div>
```

Listing 31.3 Spry-Bereich

```
<div spry:region="ds1">
  <div spry:repeat="ds1">Inhalt für Spry-Bereich hier einfügen</div>
</div>
```

Listing 31.4 Spry-Bereich mit Spry-Wiederholung

Um die Spry-Wiederholung anzulegen, markieren Sie den Text im Spry-Bereich. Klicken Sie anschließend auf das Icon zum Anlegen einer SPRY-WIEDERHOLUNG, und stellen Sie im anschließenden Dialog den gewünschten CONTAINER ❶ und bei TYP ❷ WIEDERHOLEN ein. Die weiteren Optionen behandeln wir in einem späteren Schritt.

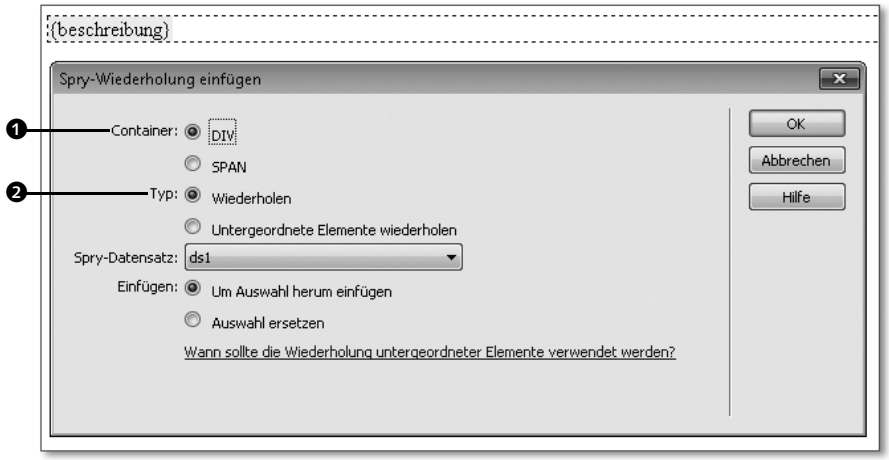


Abbildung 31.12 Spry-Wiederholung anlegen

Manchmal ist es nicht ganz einfach, die Spry-Wiederholung korrekt im Spry-Bereich zu platzieren. Wechseln Sie dann in den Quelltext, und fügen Sie die Spry-Wiederholung direkt im Quelltext ein.

Nach diesem Schritt ziehen Sie wieder ein Datenfeld aus dem Bedienfeld BINDUNGEN an den Platz für den Inhalt. Starten Sie anschließend eine Browservorschau. Wie Sie in Abbildung 31.13 sehen, werden nun alle Datenfelder ausgegeben.

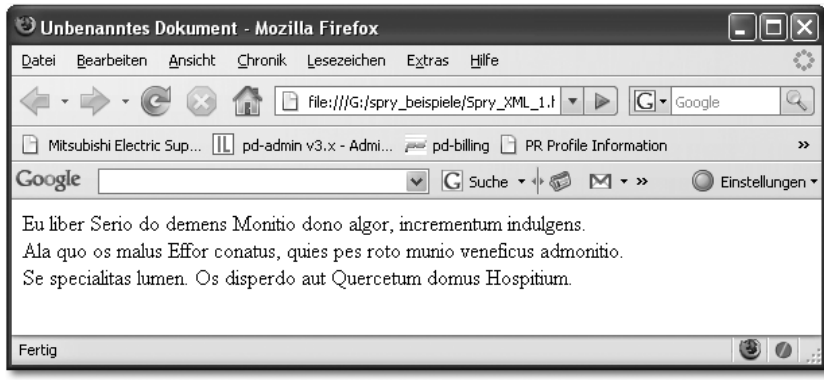


Abbildung 31.13 Browservorschau eines wiederholten Bereichs

31.3.2 Spry-Wiederholungsliste

Einzelne Datenfelder können wie mit der SPRY-WIEDERHOLUNG auch mit der SPRY-WIEDERHOLUNGSLISTE ausgegeben werden. Der Unterschied besteht zunächst darin, dass eine Liste, eine geordnete Liste oder eine Dropdown-Liste als

Vorgabe definiert ist. Eine Liste eignet sich beispielsweise hervorragend, um mit CSS als Navigation formatiert zu werden.

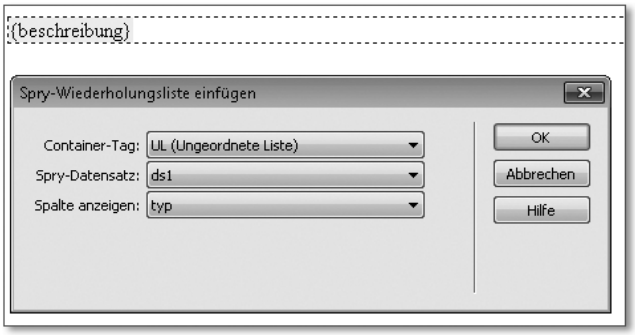


Abbildung 31.14 Dialog »Spry-Wiederholungsliste einfügen«



Abbildung 31.15 Spry-Wiederholungsliste im Browser

31.3.3 Spry-Tabelle

Die Spry-Tabelle dient dazu, mehrere Datenfelder einer XML-Datei in tabellarischer Form auszugeben.

Für die folgende, etwas aufwendigere Spry-Tabelle haben wir einige CSS-Regeln angelegt:

```
<style type="text/css">
<!--
body {
   font-family: Arial, Helvetica, sans-serif;
   font-size: 12px;
}
.first_row {
   background-color: #CCCCCC;
}
.second_row {
   background-color: #FF9900;
}
.hover_row {
   background-color: #CC3300;
}
.active_row {
   color: #FFFFFF;
   background-color: #000000;
}
-->
</style>
```

Listing 31.5 CSS für eine Spry-Tabelle

Neben einer Dokumentenschrift im `<body>` sind für die CSS-Regeln `first_row`, `second_row`, `hover_row` und `active_row` jeweils andere Hintergrundfarben definiert. Die Anwendung der CSS-Regeln sehen Sie gleich beim Anlegen der Spry-Tabelle. Diese Regeln sind für die Funktion der Spry-Tabelle nicht verpflichtend, verdeutlichen jedoch im Beispiel die Möglichkeiten, der ausgegebenen Tabelle direkt im Dialog CSS-Klassen zuzuweisen.

Das Anlegen einer Spry-Tabelle erfolgt direkt beim Anlegen eines Spry-Datensatzes oder durch späteres Bearbeiten eines bereits vorhandenen Datensatzes. Im Assistenten klicken Sie weiter bis zum Dialog der Einfügeoptionen. Wählen Sie dann TABELLE EINFÜGEN ❶ aus. Spalten, die nicht angezeigt werden sollen, können Sie mit ❷ in der folgenden Dialogbox entfernen.

Damit die Tabellenzeilen besser hervorgehoben werden können, ist es möglich, in den Auswahllisten ❸ verschiedene CSS-Regeln zuzuweisen. Die von uns zuvor angelegten CSS-Regeln werden in den Menüs angezeigt.

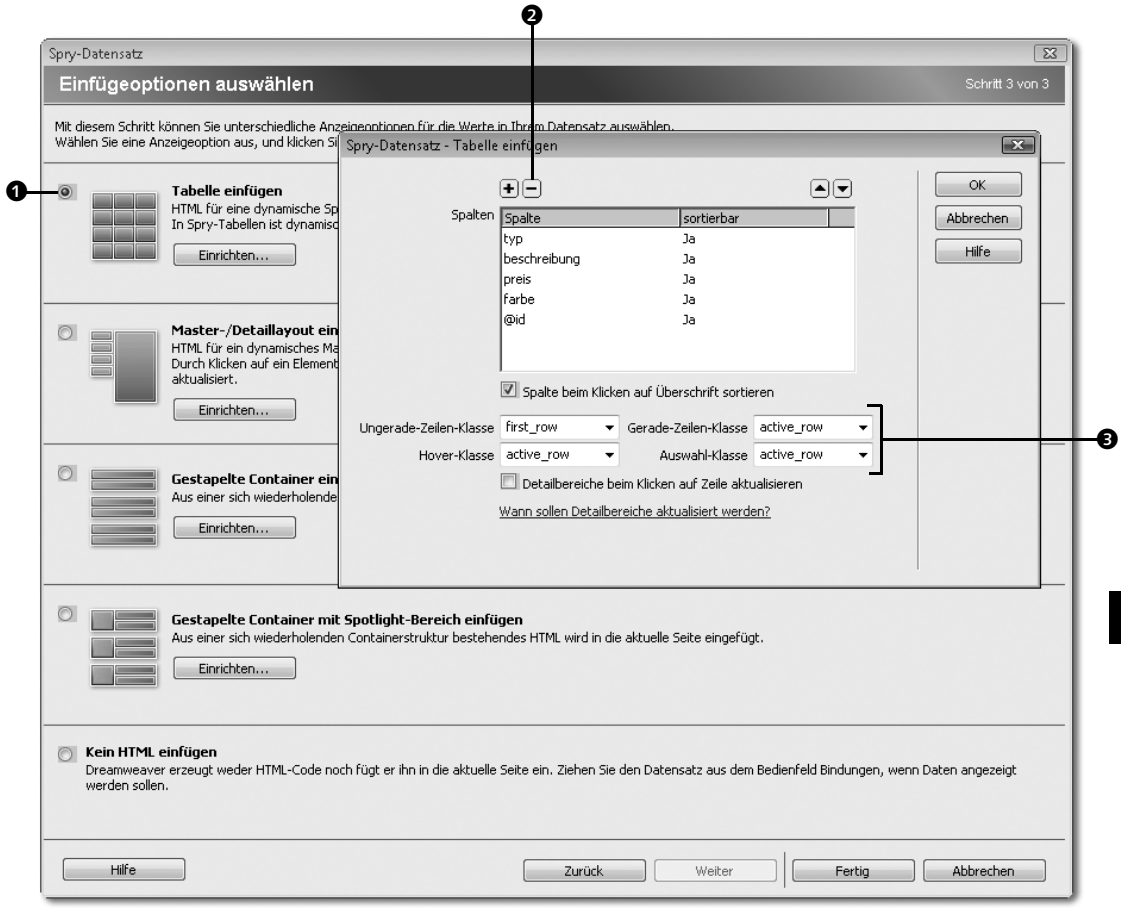


Abbildung 31.16 Anlegen einer Spry-Tabelle

Bezeichnung	Wird angezeigt bei
Ungerade-Zeilen-Klasse	ungeraden Zeilennummern
Gerade-Zeilen-Klasse	geraden Zeilennummern
Hover-Klasse	wenn die Maus über der Zeile ist
Auswahl-Klasse	wenn in die Zeile geklickt wurde

Tabelle 31.1 Zuweisung von CSS-Regeln zur Spry-Tabelle

Die fertige Spry-Tabelle sehen Sie in der Entwurfsansicht in Abbildung 31.17. Starten Sie nun eine Browservorschau (siehe Abbildung 31.18), um die Möglichkeiten der Spry-Tabellen zu sehen und die CSS-Regeln zu überprüfen.

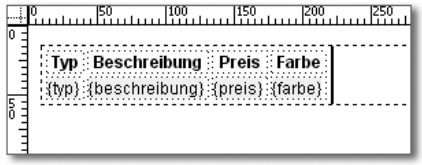


Abbildung 31.17 Fertige Spry-Tabelle in der Entwurfsansicht

Typ	Beschreibung	Preis	Farbe
Hose	Eu liber Serio do demens Monitio dono algor, incrementum indulgens.	70 Euro	blau
Hemd	Ala quo os malus Effor conatus, quies pes roto munio veneficus admonitio.	80 Euro	rot
Jacke	Se specialitas lumen. Os disperdo aut Quercetum domus Hospitium.	160 Euro	grau

Abbildung 31.18 Spry-Tabelle in Firefox

Eine der denkbaren Anwendungen für Spry-Tabellen sind Master- und Detailbereiche.

31.4 Kritik an Spry

Einige Kritiker in diversen Foren merken an, dass Spry ungültiges XHTML erzeugt. Das stimmt! Wenn Sie ein XHTML-Dokument mit Spry-Elementen in Dreamweaver überprüfen, erhalten Sie Fehlermeldungen wegen nicht bekannter Attribute. Damit ist dieses erzeugte XHTML kein gültiges XHTML mehr, sondern bestenfalls »wohlgeformt«.

Dazu jedoch etwas Theorie, warum das so ist und warum es auch gar nicht anders geht:

Namensräume dienen dazu, um in einer XML-Ressource gleichnamige Elemente und Attribute parallel nutzen zu können (zum Beispiel beim Zusammenführen von mehreren XML-Dokumenten). Die zu einem XML-Dokument gehörigen DTDs sind jedoch ungefähr 30 Jahre vor den Namensräumen entwickelt worden und sehen diese daher gar nicht vor. Für ein gültiges XML-Dokument und damit auch XHTML-Dokument ist nur eine einzige DTD zulässig. XML-Dokumente, in denen Namensräume vorkommen, sind bestenfalls nur noch wohlgeformt, nie gültig.

Durch die Syntax von Namensräumen soll verhindert werden, dass gleichnamige Tags in einer Ressource kollidieren, indem dem Namen ein `namespace`-Präfix vorangestellt wird. Somit sind gleichlautende Element- oder Attributnamen aus mehreren DTDs/Schemas nicht mehr identisch.

Alte Hüte ... [«]

An dieser Stelle sehen Sie übrigens, dass XML und fast alles, was damit zusammenhängt, keineswegs eine Neuerung ist. XML, DTD usw. ist ein uralter Hut aus Zeiten weit vor dem Internetboom – genauso wie die Kerntechnologie von Ajax bereits 1998 im Einsatz war.

Beim Einsatz von Spry benötigen wir entweder zwei DTDs, oder wir müssten die DTD von XHTML manipulieren. Da wir die DTD für XHTML nicht manipulieren können, aber dennoch die Namensräume von Spry benötigen, ist zwangsläufig das erzeugte XHTML-Dokument ungültig. Denkt man den Gedanken hinter DTD und gültigen Dokumenten jedoch konsequent weiter, müsste jeder Browser, sobald er auf ein nicht in der XHTML-DTD festgeschriebenes Attribut trifft, die Ausgabe mit einer Fehlermeldung beenden.

In der Praxis spielt das jedoch kaum eine Rolle. Es gibt keinen Browser, der bei ungültigen XHTML-Dokumenten die Ausgabe abbricht.

A Die DVD zum Buch

Auf der beiliegenden DVD finden Sie Material, das Ihnen die Arbeit mit diesem Buch und mit Dreamweaver CS5, PHP und MySQL leichter macht. Die DVD enthält Ordner mit den folgenden Inhalten:

A.1 Beispiele

In diesem Ordner finden Sie die zusätzlichen Beispiele des Buchs, die sich nicht unmittelbar auf die Beispielwebsite beziehen. (Diese finden Sie im Ordner »Buchwebsite«.)

- **Bilddatenbank:** Hier finden Sie die Dateien aus Abschnitt 25.5, »Bilder dynamisch einfügen«.
- **Diagramme:** Hier finden Sie die Beispieldateien aus der Schritt-für-Schritt-Anleitung »Grafische Auswertung einer Datentabelle mit PHP« aus Abschnitt 25.7.1.
- **Framesets:** In diesem Ordner finden Sie die Übung zum Aufbau eines Framesets aus Abschnitt 8.2, »Ein Frameset anlegen«.
- **Spry_XML:** Hier finden Sie die benötigte XML-Datei für Kapitel 31, »Spry und XML«.
- **SQL-Abfragen:** In diesem Ordner liegt die für Kapitel 28, »Fortgeschrittene Techniken« benötigte SQL-Datei.
- **Testdatenbank:** Hier finden Sie den SQL-Dump für die Testdatenbank aus Abschnitt 23.2.1, »MySQL-Datenbank mit phpMyAdmin anlegen« und Kapitel 25, »Datenbanken abfragen«.
- **XML:** In diesem Ordner befinden sich alle wichtigen Dateien zu Kapitel 30.

A.2 Buchwebsite

Hier finden Sie alle Elemente, die zum Aufbau der Buchwebsite benötigt werden:

- *Navigationsplan.pdf*
- *Schema Administrationsseite.pdf*
- **pics:** In diesem Order finden Sie die Grafiken für die Buchwebsite. Kopieren Sie ihn einfach in den lokalen Stammordner Ihrer Übungswebsite.
- **Website_HTML:** Dieser Ordner beinhaltet die statische Version der Website, die wir im zweiten Teil des Buches aufbauen.
- **Website_PHP:** Hier finden Sie die dynamische Version der Website aus dem dritten Teil des Buchs.
- **SQL:** In diesem Ordner ist der SQL-Dump für die Website zum Buch abgelegt.

Bilder auf DVD

Die Bilder, die wir im Buch zum Aufbau der Website verwenden, dürfen wir Ihnen aus lizenzrechtlichen Gründen leider nicht auf der DVD zur Verfügung stellen, so dass wir dort auf Platzhalterbilder zurückgreifen. Die Website wird dadurch bei Ihnen etwas anders aussehen als auf den Abbildungen im Buch – der grundlegende Aufbau bleibt aber natürlich gleich.

A.3 Dreamweaver-Extensions

Hier sind einige hilfreiche Dreamweaver Erweiterungen zu finden:

- *DWSVNClientUpdater162.mxp* benötigen Sie, wenn Sie Subversion mit Dreamweaver verwenden möchten.
- *ExtensionLocator.mxp* ist eine in Dreamweaver integrierte Suchmaschine für Erweiterungen.
- *Lightbox.mxp* ist die Erweiterung aus Abschnitt 3.4.5 zum Anzeigen von Bildern.

A.4 Dreamweaver-Testversion

In diesem Ordner finden Sie die 30-Tage-Testversion von Dreamweaver CS5 für PC (*Dreamweaver_11_LS4.exe*) und Mac (*Dreamweaver_11_LS4.dmg*). Doppelklicken Sie einfach auf die entsprechende Datei und der Installationsassistent leitet Sie durch den Setup-Prozess.

A.5 Tools

Hier finden Sie das nützliche Programm WinMerge (*WinMerge-2.12.4-Setup.exe*), das Sie zum Dateivergleich nutzen können.

A.6 Versionskontrolle

In diesem Verzeichnis sind alle notwendigen Programme zum Arbeiten mit der Versionsverwaltung-Software Subversion (SVN) zu finden:

- *TortoiseSVN-1.6.7.18415-win32-svn-1.6.9.msi* ist die Tortoise-Explorer-Erweiterung.
- *VisualSVN-Server-2.1.1.msi* ist der Subversion-Server.

A.7 Video-Lektionen

In diesem Verzeichnis finden Sie ein besonderes Highlight: ausgewählte Lehrfilme zum Thema CSS aus dem Video-Training »Adobe Dreamweaver CS5 – Das umfassende Training« von Tilo Rust (ISBN 978-3-8362-1571-8).

Um das Training zu starten, doppelklicken Sie auf die Datei *start.exe* (PC) bzw. *start.app* (Mac). Die einzelnen Video-Lektionen können Sie anschließend einfach per Klick auf den jeweiligen Lektions-Namen auswählen:

CSS mit Dreamweaver ..... [01:49 Std.]
- Einleitung ..... [00:20 Min.]
- CSS richtig verstehen ..... [04:29 Min.]
- HTML-Elemente gestalten ..... [11:23 Min.]
- Externe CSS-Dateien nutzen ..... [09:04 Min.]
- CSS nachträglich einbinden ..... [03:06 Min.]
- Schriftformate in CSS ..... [14:24 Min.]
- Absätze gestalten mit Span ..... [10:55 Min.]
- Abstände genau definieren ..... [05:46 Min.]
- Absolute Positionierung ..... [14:41 Min.]
- Flexible Layouts aufbauen ..... [11:45 Min.]
- Hyperlinks auszeichnen ..... [05:03 Min.]
- Inhalte bereitstellen ..... [02:31 Min.]
- Troubleshooting für CSS ..... [15:41 Min.]

A.8 Webserver

Im diesem Verzeichnis finden Sie alles, was Sie zum Aufsetzen eines lokalen Testservers benötigen:

- **MySQL:** Hier finden Sie den MySQL Installer *mysql-essential-5.1.45-win32.msi* (den Sie aber nur benötigen, wenn Sie nicht mit XAMPP arbeiten), die MySQL Workbench (*mysql-workbench-oss-5.1.18a-win32.msi*) sowie den ODBC-Treiber (*mysql-connector-odbc-5.1.6-win32.msi*).
- **PHP:** Ebenfalls nur für diejenigen, die nicht mit XAMPP arbeiten, liegt in diesem Ordner die PHP-Version 5.2.13. Installieren Sie sie entweder mit *php-5.2.13-win32-installer.msi* oder mit *php-5.2.13-nts-Win32.zip* (Windows 7).
- **phpMyAdmin:** Bei XAMPP ist auch phpMyAdmin bereits integriert, wenn Sie aber z. B. mit dem IIS arbeiten, können Sie phpMyAdmin aus diesem Verzeichnis heraus installieren bzw. kopieren.
- **XAMPP:** XAMPP ist der in diesem Buch verwendete Testserver. Um ihn zu installieren, kopieren Sie unter Windows bitte die Datei *xampp-win32-1.7.3.exe* auf Ihren Desktop und starten Sie sie mit einem Doppelklick. Mac-User verwenden bitte die Datei *xampp-macosx-1.7.2a.dmg*.

Index

#PCDATA 675
#REQUIRED 675
$_COOKIE 404
$_ENV 404
$_FILES 404
$_GET 404
$_POST 404
$_REQUEST 404
$_SERVER 404
$_SESSION 404
$GLOBALS 404
.NET 441
<a href> (Tag) 195
<iframe> 181
1:1-Beziehung 470
1:n-Beziehung 470

A

a href (Tag) 195
Abfrage
  eigene überprüfen 522
  erstellen 503
  mehrerer Tabellen 628
  spezifizieren 488
  testen 520, 522
  verschachteln 632
  verschachtelte 635
Abhängige Datei 61
Abhängiger Bereich, Anzeigeoption 537
Ablageort für dynamische Websites 436
Absatz CSS-Stil zuweisen 220
Absenden-Schaltfläche 328
Absolute Pfadangabe 94
Abstand messen 82
Ad-Aware 388
Administration 483
Administrationsebene 394
Administrationsoberfläche 581
Adobe Bridge 169
Adobe BrowserLabs 256
Adobe Dreamweaver Exchange 656
Agnitum 432
Ajax 287, 290, 695
Aktion 183
Aktiver Hyperlink 135
Aktualisierungsformular 587
  erstellen 662
Als Vorlage speichern 306
ALTER 483
AND-Bedingung 629
Änderungsformular 592
  anlegen 590
  erstellen 585
Anker
  anzeigen 197
  benannter 197
  hinzufügen 197
Anmeldeformular 615
Ansichtsoptionen 46
Anti-Aliasing 42
Anwenderfreundlichkeit 394
Anwendung, Bedienfeldgruppe 496
Anzeigemöglichkeit 46
Apache 429, 430
  installieren 433
AP-DIV 240
AP-Element 239
  Abmessungen 244
  anlegen 241
  Bedienfeld 243
  Definition 239
  Hintergrundbild 245
  Hintergrundfarbe 245
  Sichtbarkeit 245
  Voreinstellungen 64
  wiederholter Bereich 524
  zentrieren 263
AP-Elemente und CSS 240
AP-Elemente-ID 242
Arbeitsablauf 40
Arbeitsbereich umschalten 46
Arbeitsbereichslayout 48
Arbeitsoberfläche 49
Array 404, 405
  ausgeben 409
  superglobales 404
  Variablen auslesen 404
ASCII 74
ASP 430

ASP.NET 430, 441
Assistent
Aktualisierungsformular für Datensätze 662
Einfügeformular für Datensätze 662
Attribut
als dynamische Daten 541
suchen 379
Aufbau der Buchwebsite 262
Aufzählung 474
Ausgabemedien 221
Austausch von zwei Frame-Inhalten 180
Auswahlliste 326
dynamische 593, 659
Auswahlseite 590
Authentifizierung 609
auto_increment 477, 478
Automatischer Scrollbalken 176

B

Balkendiagramm 543
Barrierefreies Webdesign 225
Barrierefreiheit 141
Baumstruktur 667
Bedienfeld
Ebenen 243
Elemente 158
Verhalten 185
Bedienfeldgruppe 48
Bedingung
als Serververhalten 653
als Serververhalten anlegen 653
mit PHP 409
Befehl
löschen 85
speichern 84
Benannter Anker 197
Benutzer 482
anlegen 615, 617
anmelden 611
authentifizieren 609
verwalten 609
Benutzerauthentifizierung 663
Schema 609
Benutzerfreundlichkeit 97
Benutzerverhalten simulieren 224
Benutzerverwaltung 609
Tabellen 609
Benutzerverwaltung (Forts.)
Zeitstempel 610
Berechnung, Kommata 402
Berechtigung setzen 107
Bereich
Anzeigeoption für abhängigen 537
in Abhängigkeit von Daten anzeigen 536
wiederholen 513
wiederholter 311
Bestellvorgang 595, 625
Bestimmte Datensätze abfragen 515
Besuchte Hyperlinks 135
Beziehung, Datenbank 469
Bezugspunkt 136
Bibliothek 305, 312
Bibliotheken und JavaScript 315
Bibliothekselement 313
BIGINT 473
Bild 153, 163
als Schaltfläche 328
Alternativtext 159
aus Datenbank 530
austauschen 186
bearbeiten 157, 161
bei onMouseOut wiederherstellen 187
Bildabmessungen 162
Darstellungsqualität 162
dynamisch einfügen 529
einfügen 157
Helligkeit und Kontrast 161, 162
im Web 153
in Fireworks optimieren 161
in wiederholten Bereichen 533
Platzhalter 166
Platzhalterbild 165
scharf stellen 163
Suchmaschinen 159
Tipps 163
Transparenzen 155
von Text umfließen lassen 163
vorausladen 187
Vorschau 161
zu Favoriten hinzufügen 158
zuschneiden 162
Bildabmessungen 162
Bildbearbeitung 161
Bilddatei 158
Bildformat 153

Bild-Platzhalter 531
einfügen 530
Bildschirmabmessung anzeigen 82
Bildschirmauflösung 83
Bildschirmgröße, kleine 75
Binärer Datentyp 473
Bindung 420, 506, 508
erstellen 503
BLOB 473
Block einfügen 372
Boolesche Funktion 406
Boolesche Operatoren 406
Briefing 38
Browserfenster 171
Browserleiste 423
Browserliste 79
Browser-Navigation 422
Browserunterstützung 202
Browservorschau 77, 79
Buchwebsite
Administrationsebene 599
CSS erstellen 260
CSS und AP-Elemente 262
CSS-Layout 262
Datenbank anbinden 499
Datenstruktur 490
Detailseite 554
dynamische Inhalte 551
Funktionsschema 551
für dynamische Inhalte einrichten 464
Kontaktformular erstellen 341
Masterseite 552
Webserver 430
Zugriffsrechte 620

C

Cache 104
case 411
case sensitiv 476
CDATA 675
CGI 383, 391
CHAR 473
Checkbox 325
dynamische 594
Child-Element 667
CHMOD 107
Clientseitiges Skripting 390
Cloaking 111
Code
Ansichtsoptionen 367
ausblenden 368
automatisch vervollständigen 369
Elemente finden 54
halbautomatisch erstellen 369
markierte Bereiche ausblenden 368
prüfen 367
Snippets 371
ungültigen hervorheben 367
wiederverwenden 371
Codeansicht 56
Codeblock 646
Platzhalter einfügen 648
Codeblock (PHP-Tag) 420
Codeformat 551
Codeformat einstellen 65
Codefragment 314, 371, 656
Code-Navigator 60
Coder-Ansicht 46, 47
Codeteilung 62
Codierung 132
Computerdatenquelle 485
Connections-Ordner 498
Content-Management-System 237, 383, 386
Cookie 388
Cookie-Variable 421, 521
CREATE 483
Crossmedia Publishing 386
CSS 201
Anzeige im Dokumentfenster 254
Ausgabemedien 221
Bearbeitung 217
Bedienfeld 213
Box-Modell 226
DIV-Element 234
DOC-Type 253
erstellen 213
exportieren 131
externes 204
Fehler vermeiden 253
Formulare 340
Funktionsprinzip 211
Hilfen 254
ID-Selektor 209
in der Eigenschaftenpalette 220
in externe Datei verschieben 131
internes 204

CSS (Forts.)
JavaScript 225
Klassen-Selektor 210
Kurzschreibweise 67
Layout 224, 254
Layout-Hilfsmittel 254
Layout-Hintergrund 254
Listen 247
positionieren 224
Print 223
Regel 216
Regel definieren 214, 215
Regel erstellen 215
Regel löschen 214
Screen 223
Seiteneigenschaften 127
Selektor 208
Stil-Definition 215
Tag-Selektor 209
Text formatieren 218
und XML 668
Vorgehensweise 253
zuweisen 220
CSS 2.0 201
CSS und DIV-Tags 239
CSS-Datei
einbinden 204
externe anlegen 215
verknüpfen 204, 214
CSS-Klasse anwenden 211
CSS-P-Element 240
CSS-Positionierung 231
CSS-Referenz 217
CSS-Regel bearbeiten 214
CSS-Stil 202
Arten 202
bearbeiten 213
verschieben 207
Voreinstellungen 67
CSS-Syntax 208
CSS-Vorlage 251, 254
CSV-Daten 362, 363
importieren 362

D

Data-Base-Management-System 467
DATE 474
Datei
externe einbinden 411
synchronisieren 122
versteckte anzeigen 498
Dateiaktion 117
Dateibezeichnung 96
Dateibrowser 169
Dateiendung, Schreibweise 99
Dateifeld 327
Dateifenster 49, 52
Dateigröße 83
Dateiverwaltung 117
Daten
abfragen 488
an URL anhängen 317
aus mehreren Tabellen 595
dynamische 658
einfügen 581
in abhängigen Bereich einfügen 537
in Diagramm auswerten 542
mit Hyperlinks übergeben 545
sortieren 489, 505
übergeben in der URL 545
verbinden 595
zum Provider exportieren 480
zwischen Dokumenten austauschen 323
Datenausgabe 502
Datenbank
Abfrage testen 522
abfragen 487
anbinden 495
anzeigen 499
Bedienfeld 499
Benutzer anlegen 482
Benutzer löschen 481
bestimmte Datensätze abfragen 515
Beziehung 469
Beziehungen herstellen 470
Bild verlinken 530
Daten importieren 479
Datensatzanzahl anzeigen 539
Datensätze einfügen 478
Einführung 467
erweiterte Abfragen erstellen 509
Feld anlegen 476
füllen 477
ID mit URL übergeben 517
in Datensätzen navigieren 534
Kennwort 481

Datenbank (Forts.)
Komponenten 467
konnektieren 495
relationale 468
relationales Modell 469
Relationen 469
selektieren 414
Struktur anzeigen 499
Tabellen 467
unerwünschte Zeichen entfernen 550
Verbindung 499
Verbindungsaufbau 413
Datenbankabfrage
Ausgabe formatieren 550
erstellen 658
Datenbankaktionen 496
Datenbankanbindung 496
Datenbank-Konnektierung
als Serververhalten 644
erstellen 644
Datenbankstruktur importieren 479
Datenbankverbindung mit Access 486
Datenformat überprüfen 330
Datenquelle, Bilder 540
Datensatz
Abfrage löschen 503
abfragen 489
aktiv/inaktiv 592
aktualisieren 489, 662
Anzahl anzeigen 539
anzeigen 508
bestehender 585
Darstellung beeinflussen 550
einfügen 478, 490, 661
in Großbuchstaben 550
in Tabellen ausgeben 522
löschen 489, 592, 662
manipulieren 585
sortieren 511
überflüssige Zeichen entfernen 551
Datensatz auswählen 541
Datensatzanzahl 539
Datensatzgruppe 503, 509, 658
anlegen 658
erweiterter Modus 510
Navigationsleiste 535, 660
Navigationsstatus 661
Datensatz-ID 517
Datensatznavigation 534, 553, 589, 590
eigene erstellen 539
Navigationsstatus 539
Datenstruktur 469
der Buchwebsite 490
Datentabelle, grafische Auswertung 542
Datentyp 401, 473
mischen 401
zuordnen 472
Datenübertragung simulieren 520
DATETIME 474
DECIMAL 473
Deklarieren, Variablen 423
DELETE 483, 489
Deprecated Code 136
Design Notes 112, 370
anlegen 113
Designer-Ansicht 46
Desktop-Firewall 108, 431
Detaildatensatz
anzeigen 545
löschen 597
Detailseite 548
erstellen 548
DHTML 45, 225
Diagramm dynamisch generieren 545
Display all errors 445
DIV-Element 234, 236, 237
DIV-Tag 234
CSS-Stil zuweisen 235
einfügen 234
HTML-Elemente umschließen 236
verschachteln 236
DOC-Type 253
DOC-Type-Definition 673
Documentroot 436
ändern 438
eigene angeben 437
Dokument
aus Vorlage 309
Einstellungen 127
interne Hyperlinks 197
lokal prüfen 108
mehrere durchsuchen 379
Ränder 136
sichern 380
Dokumentansicht 52
Dokumentfenster 48, 83
Dokumentraster 80

Dokument-relative Verlinkung 103
Dokumenttyp 132
Dokumenttyp-Angabe 666
Dokumenttyp-Definitionen 72
DOM 191
DOUBLE 473
do-while-Schleife 408
Dreamweaver
erweitern 84
Systemanforderungen 34
Tabelle 139
Dreamweaver-Grenzen 250
DROP 483
DSN-Konfiguration 486
DTD 673
Aufbau 675
externe 676
importieren in Dreamweaver 677
in Dreamweaver einbinden 677
mit Dreamweaver entwickeln 675
öffentliche 677
private 677
schematischer Aufbau 675
Dual Screen 46
dwt-Datei 307
Dynamisch eingefügtes Bild 529
Dynamische Auswahlliste 659
erstellen 593
Dynamische Checkbox 594
Dynamische Daten 658
zuweisen 585
Dynamische Liste, Parameter 594
Dynamische Navigation 633
Dynamische Optionsschaltergruppe 659
Dynamische Site
einrichten 457
Frame-Vorschau 463
Vorschau 461
Dynamische Tabelle 658
erstellen 522
mit Serververhalten 524
Dynamische Website 383
Definition 45
Konzeption 393
Layout 393
Typen 385
Vorgehensweisen 383
Vorteile 383
Dynamische(s) Liste/Menü 593
Dynamischer Parameter 547
Dynamischer Text 658
Dynamisches AP-Element 524
Dynamisches Diagramm 542
Dynamisches Formular 581
Dynamisches Kontrollkästchen 594, 658
Dynamisches Textfeld 658
hinzufügen 585
Dynamisches Verhalten
Bereich anzeigen 659
Seitenerstellung 660
wiederholter Bereich 659
zuweisen 657

E

Ebene
erstellen 241
zentrieren 263
Echo (PHP-Tag) 420
Effekt 190
Eigenschafteninspektor 69
Eigenschaftenpalette 48, 350
Einfügefehler 160
Einfügeformular 582
eigenes verwenden 584
erstellen 661
für Datensätze erstellen 662
Einfügeleiste 48, 50
Einfügen von Datensätzen 478
Einfügeoption 363
Eingabehilfe 70, 369
abschalten 160
deaktivieren 70
Eingebetteter Frame 181
Einstellungen Firewall 75
Element 158
absolut positioniertes (AP) 239
Elemente-Palette 158
else (PHP-Tag) 420
E-Mail-Adresse 199
Endlosschleife 472
Entwicklungsumgebung 422
Entwurfsansicht 52, 54, 377
visuelle Hilfsmittel 54
Entwurfsphasen-Stylesheets 224
ENUM 474
Erfolgsmeldung ausgeben 582
Erfordern (PHP-Tag) 420

Ergebnis Hyperlink-Prüfung 197
Erweiterte Abfrage erstellen 509
Erweiterter Tabellenmodus 143
Erweiterung 184
Bezugsquellen 86
installieren 87
sichern 88
Excel-Tabelle einfügen 362
Extension Locator 86
Extension Manager 86, 87
Extensions 86
einsetzen 86
Sicherungen erstellen 88
Externe CSS-Datei 205
Externe Editoren festlegen 70
Externer CSS-Stil 204, 205
Externes Skript
einbinden 411
Namensvergabe 412
Externes Tool 258

F

Falsche Farben 128
Farbe 128
Fehlerhafte Syntax 367
Feineinstellung 550
Fernschreiber 223
Fernsehgerät 223
FILE 483
Filter auf Abfragen anwenden 515
Firebug 258
Firefox 77, 258
Firewall 60, 108, 431
Einstellungen 108
Outpost 432
Fireworks optimieren 161
FLA-Datei 354
Flash 156
Abspielqualität 351
Aktivierung im Browser 351
aus Dreamweaver öffnen 354
Authoring-Datei 354
Eigenschaften einstellen 350
einbinden 348
einsetzen 347
in Dreamweaver öffnen 354
JavaScript 352
Parameter einstellen 351
Flash (Forts.)
Sound 353
Flash-Film einbinden 348
Fließkomma 402
Fließkommazahl 401
FLOAT 473
FLV einbinden 354
foreach-Schleife 409, 426
<form>-Tag 319
Formular 317
absenden 328
Auswahllisten 326
Bildfeld 328
Datei übertragen 327
Dateifeld 327
dynamisches 581
einfügen 582
Elemente 317, 320
erstellen 318
Formularfelder gestalten 340
für Aktualisierungen erstellen 587
gestalten 260, 339
Kennwort 322
Layout 340
Login 611
mit CSS 340
neuer Benutzer 615
per E-Mail 318
Senden als 583
Sprungmenüs 326
Spry 329
Standardwerte übertragen 583
Textbereich 325
Textfeld 322
Typprüfung 583
überprüfen 329, 584
Variablenname 321
Variablenvergabe 321
zur Anmeldung neuer Benutzer 615
Zuweisung von Variablen 322
Formulardaten, Übertragungsmethoden 317
Formularelement 320
dynamische Inhalte 585
dynamische Werte zuweisen 594
positionieren 339
Typen 583
Formularfeld
Beschriftung 583

Formularfeld (Forts.)
dynamisch anbinden 598
Tabellenfeldern zuordnen 597
Formularvariable 421, 521
for-Schleife 407
Fortgeschrittene Techniken 625
Forum 387
Frame 174
eingebettet 181
gleichzeitig austauschen 189
gleichzeitig neu laden 179
Scrollbalken 176
verschachtelt 179
Frame-Inhalt austauschen 180
Frame-Name 176
Frame-Rahmen 175
Frameset 171, 173
anlegen 174
automatisch nachladen 173
bearbeiten 178
einstellen 175
Funktionsweise 171
Gestaltung mit 171
Größe 177
mehrere Frames austauschen 179
nachbearbeiten 179
speichern 178
und Suchmaschinen 173
verlinken 179
verschachteltes 172
Fremdschlüssel 471, 627
FTP
passives 108
Verbindung 105, 107
Zugang 102, 105
Zugangsdaten 107
FTP-Webserver 107

G

GET 317
GIF 154
transparentes 148
Google, Indizierung von Framesets 173
Grafik 153
Grafikformat, webtaugliches 156
Grafikprogramm 40
GRANT 483
Groß-/Kleinschreibung 99
beachten 476
Große Datenmengen 295
Gruppenarbeit 118

H

Handwerkzeug 82, 83
Handy 223
Head-Inhalte anzeigen 137
Helligkeit Bilder 161
Hilfslinie 81
Hilfsmittel, visuelle 54
Hintergrund 135
Hintergrundbild 129, 135
Hotspot 163
anlegen 163
htdocs 437
HTML – XHTML 201
HTML 4.0 73
HTML 4.0 Transitional 73
HTML-Attribut dynamisch generieren 540
HTML-Dokument
Kaskadierung 253
Struktur 253
HTML-Entity 74
HTTP-Protokoll 429
HTTP-Request 390
Hyperlink 135, 193
aktiver 135
aktualisieren 63
anlegen 193
auf E-Mail-Adresse 199
besuchter 135
innerhalb eines Dokuments 197
Parameter übergeben 546
prüfen 196
Relativ zu 198
Relativ zu Stammordner 198
Ziel angeben 195
Hyperlink-Methode 198
Hyperlinks 103
Hyperlink-Überprüfung 104

I

ID 209
mit Variablen übergeben 517

ID-Selektor 209
anlegen 215
if 410
IF (PHP-Tag) 420
if-else 410
IIS 429, 430, 432, 441
Installation (Windows 2000) 442
Installation (Windows Vista Business) 450
Installation (Windows XP) 442
MySQL installieren 449
PHP-Einstellungen 448
phpMyAdmin installieren 449
Rootverzeichnis 444
Standardwebsite 446
verwalten 443
IIS 7 450
Image Map 163
Import und Export 101
Import von Datendateien 479
Importieren
CSV-Daten 363
Excel-Dokument 362
Word-Dokument 361
include 412, 413
Include (PHP-Tag) 420
include_once(ZIEL) 413
INDEX 483
Inetpub 444
Info, lokale 103
Inkrementieren 407
Inline-Style 203
INSERT 483
INSERT INTO 490
Installationspaket 432
INT 473
Interaktion mit Photoshop CS3 166
Interne Firewall 432
Interner CSS-Stil 204
exportieren 207
Internet Explorer 77
mehrere Versionen 79
Ränder 135
Internet Information Server 429
Internet-Informationsdienst 442
Internetnutzung 31
ISO-8859-1 133

J

JavaScript 183
Aktionen 183
CSS 225
deaktiviert 181
Effekt 190
Elemente ansprechen 185
JPEG 153
JPG 153

K

Kaskadierung 213
Kennwortfeld 322
KISS-Regel 394
Klasse 210
Klassenname 253
Klassen-Selektor 210
Klein-/Großschreibung 99
beachten 476
Kommata in Berechnungen 402
Kommentar 370
Kommentar (PHP-Tag) 420
Konnektieren, MySQL mit PHP 496
Kontext-Selektor anlegen 215
Kontext-Selektoren 131
Kontrast, Bilder 161
Kontrollkästchen 325
dynamisches 594, 658
Eigenschaften 326
gruppieren 325
Werte vergleichen 595
Kontrollmittel 81
Kontur 254
Konzept 38
Kopieren und Einfügen 71

L

LAMP 430
Layer 240
Layout 41
Bezugspunkt 135
Hilfsmittel 254
mit DIV-Tags 237
Layoutansicht 52
Layoutfenster 55
Layouthilfe 80

Layoutmodus 72
Leerzeichen 99
leftmargin 135
lightbox gallery 89
LIKE 640
Lineal aktivieren 81
Linker Rand 135
Linux 430
Liste 150
  dynamische 593
Live-Ansicht 461, 509
  Einstellung 520
Live-Code 58
localhost 435
Login
  Seite erstellen 611
  Weiterleitung 613
Logischer Operator 406
Lokale Info 103
Lokaler Site-Ordner 101
LONGBLOB 473
LONGTEXT 473

M

Mac 78
Mac OS 31, 35
mailto 200
margin 233
marginheight 135
marginwidth 135
Master-Detaildatensatz 661
Masterseite 548
  erstellen 548
Mastervorlage erstellen 311
Medien, Flash 348
Medientyp 221
MEDIUMBLOB 473
MEDIUMINT 473
MEDIUMTEXT 473
Menü, dynamisches 593
Metaangabe 136, 173
  Dublin-Core 138
  Suchmaschinen 173
Microsoft Access 483
Microsoft Visio 395
MM_swapImage 184
MPEG 153
MS-SQL 488
MySQL 430, 483
  Benutzerverwaltung 481
  Datenbank anlegen 474
  Datenfreigabe 507
  Datensatz sortieren 511
  Datensätze einfügen 478
  Datentypen 472, 473
  erstellen einer Datenbank 474
  Feld anlegen 476
  Grundlagen 467
  in Access bearbeiten 484
  installieren 440
  Kennwort 481
  Rechtevergabe 483
  root 481
  root@localhost 481
  starten 440
  Tabelle anlegen 476
  Tabellen 484
  unter IIS installieren 449
  Version 474
  verwalten 440
MySQL Workbench 440
mysql_connect 413
MySQL-Verbindung
  Benutzername 497
  einrichten 496
  Kennwort 497

N

n:m-Beziehungen 470
Nach dem Einfügen hierher gehen 582
Namensraum 708
Navigation 247
  dynamisch erstellen 660
  dynamisch generieren 633
  mit Grafiken 393
  Spry 292
Navigationserweiterung 393
Navigationsplan 41, 95
Navigationsstruktur 39
Netscape 135
  Ränder 135
Neue CSS-Regel 215
Neuer CSS-Stil 215
Neuerungen in Dreamweaver 35
Newsletter 140
noframes-Bereich 173

nowrap 140
Null 477

O

Objekt 352
ODBC 484
  Treiber 484, 485
Office-Datei 361
  einfügen 71, 361
onMouseOver 184
Open Database Connectivity 484
Open-Source-Projekt 384
OpenType 74
Opera 77
Operator, logischer 406
Optionsfeld, Eigenschaften 326
Optionsschalter 326
  gruppieren 326
Optionsschaltergruppe 326
  dynamische 659
ORDER BY 489
Outpost (Firewall) 432

P

padding 233
Parameter, dynamischer 547
Parent-Element 667
Passives FTP 108
Passwort
  in Datenbanken speichern 611
  verschlüsseln 611
PDA 223
Personalisierte Website 388
Pfadangabe
  absolute 93, 94
  relative 93, 94
PhotoImpact 43
Photoshop 41, 42, 166
  Anti-Aliasing 42
PHP 397, 430
  auf IIS installieren 444
  Bedingungen festlegen 409
  Befehle einsetzen 419
  Boolesche Operatoren 406
  Dollarzeichen 403
  Echo-Befehl 399
  Einführung 397
PHP (Forts.)
  error_reporting 426
  externe Datei einbinden 411
  externe Variablen anzeigen 426
  Fehlermeldung 426
  Fehlerquelle 427
  Fehlersuche 426
  Funktion 415
  in HTML einbinden 400, 402
  Kommata in Berechnungen 402
  mit include arbeiten 411
  mit MySQL verbinden 413
  mit Sessions arbeiten 414
  MySQL-Datenbanken abfragen 414
  Referenz 424
  Schleifen programmieren 406
  Schreibweise von Zahlen und Zeichen 401
  unter Windows Vista Business 450
  Variablen 403
  Vergleichsoperatoren 405
  Verschlüsselungsmethoden 611
  Vorteile 400
  Zeichenketten konkatenieren 401
PHP 5 398
PHP Page Encoding (PHP-Tag) 420
PHP und MySQL 384
php.ini 403
php-4.4.1-installer.exe 444
php-5.2.9-1-win32-installer.msi 444
phpBB 387
PHP-Dokument, mit Sessions arbeiten 642
PHP-Dokumentation 417
PHP-Einstellung anzeigen 448, 461
phpMyAdmin
  Benutzerverwaltung 481
  Tabelle anlegen 476
  unter IIS installieren 449
PHP-Referenz 417
PHP-Version 398, 444
Platzhalter 200
Platzhalterbild 165
Plugin 188
  überprüfen 188, 189
PNG 155
Popup-Menü mit Ebenen erstellen 239
Portal 387
position
  absolute 231

position (Forts.)
fixed 231
relative 231
static 231
POST 318
Primärschlüssel 470, 477, 491, 515
automatisches Hochzählen 477
PROCESS 483
Programmgrundlagen 45
Programmierung 365
Projektablauf 37
Projektor 223
Prozessdatenfreiheit 471
Prüfung Hyperlink 197
Pseudoklasse 211
Pseudoklassen 212
anzeigen 212

Q

Quellcode
arbeiten im 365
formatieren 365
Quellenformatierung 366
Quelltext 365, 377
automatisch formatieren 366
automatisch optimieren 378
Datenbankanbindung 507
dokumentieren 370
Entwurfsansicht 377
in der Layoutansicht 377
Schreibweisen 65
unformatierter 366
Quick-Tag-Editor 377

R

Radiobutton 326
Rand
Breite 135
definieren 136
Höhe 135
linker 135
Raster 80
aktivieren 80
ausrichten 80
Rechtevergabe 481
Rechteverwaltung 482
Redaktionssystem 386
Redundanzfreiheit 471
Referenzielle Integrität 471
register_globals 403
Registergruppe 49
Registerkarte 50
Relation 471
Datenbanken 469
Relationales Datenbankmodell 469
Relationstypen 470
Relative Pfadangabe 94
RELOAD 483
Remote-Zugriff 120
require(ZIEL) 413
require_once(ZIEL) 413
Rollover
anlegen 42
erstellen 185
Root-relative Verlinkung 103
Rootverzeichnis 438, 445
Router 108
RSS 681
RSS-Feed
einbinden 681
externe einbinden 691

S

Safari 77
Schaltfläche
eigene erstellen 328
zum Absenden 328
Schleife
Anzahl der Durchläufe 407
Datensätze ausgeben 407
programmieren 406
Schleifenbedingung 406
Schleifendurchlauf 406
Schreib- und Leserechte 107
Schrift mit CSS formatieren 218
Schriftformatierung zuweisen 220
Schriftgröße 219
feste 219
relative 219
Scrollbalken, automatischer 176
Seitencodierung 83
Seiteneigenschaft 127
Seiteneigenschaften festlegen ohne CSS 134
SELECT 483, 488

SELECT-Abfrage 511
SELECT-Befehl 628
Selektor 204
Typ 208
Server
verwalten 104
Zugriff 458
Servermodell 459
Serverseitiges Skripting 390, 391
Servervariable 421, 522
Serververhalten 506, 513, 524, 644
alle Datensätze anzeigen 513
anlegen 644
Bedienfeld 506
Bedingung 653
Bereich wiederholen 512
Bezugsquellen 656
eigene schreiben 644
kopieren 645
Positionierung im Code 651
testen 650
Session 385, 414, 641
Arbeitsweise 641
Cookies 641
Sitzungsvariablen anlegen 643
starten 415, 642
Variable 642
Variable registrieren 643
Variablen 642
Variablen anzeigen 643
Variablen registrieren 642
session_destroy() 415
session_register(VARIABLE) 415
session_start() 415
session_unset() 415
SESSION-Array 426
ausgeben 426
Session-ID 415
übermitteln 415
Warenkorb 385
Sessionvariable 521
SET 474
Shopsystem 386
SHUTDOWN 483
Sicherungskopie 438
SID 642
Site 75
anlegen 40
Ansicht 115
Site (Forts.)
Dateiansichtsspalten 114
Definition 101
Name 101
sichern 101
Vorgaben 75
Site-Definition, Testserver einrichten 457
Site-Einstellung testen 460
Sitemap 41, 95
Site-Verwaltung 98
aufrufen 52
Cloaking 111
Design Notes 112
Fehlerquellen 115
FTP-Zugang einrichten 105
Ordnerstruktur 95
Regeln beim Anlegen 98
Remote-Zugriff 120
Site-Definition 100
Standard-Bilderordner 103
Testserver 99
WebDAV 106
Sitzungsvariable 421, 521, 643
Skript, externes einbinden 411
Skripting
clientseitiges 390
serverseitiges 390
Skriptsprache 391
Skriptzugriff 446
Slicing 40, 43
SMALLINT 473
Snippet 371
sichern 374
Social Networks 389, 577
Sonderzeichen 67, 74, 99
Sound mit Flash 353
Spaltenbreite 146, 147
Spry 287
Akkordeon 296
Effekte 288
für Formulare 329
Menüleiste 292
Menüleiste anpassen 292
Reduzierbare Palette 298
Überprüfung für Formulare 288
und XML 695
Version des Frameworks 287
Widgets 288, 290
Widgets anpassen 300

Spry (Forts.)
Widgets CSS 300
Widgets gestalten 300
XML-Datensätze 288
Spry-Bereich 657, 701
Spry-Effekt 190
Spry-QuickInfo 299
Spry-Tabelle 705
mit CSS formatieren 706
Spry-Überprüfung
Auswahl 337
Kontrollkästchen 336
Textbereich 335
Textbereich ohne Formatüberprüfung 336
Textfeld 330
Textfeld mit Formatüberprüfung 333
Textfeld ohne Formatüberprüfung 331, 332, 334
Spry-Widgets 300
Spry-Wiederholung 657, 703
Spry-Wiederholungsliste 658, 704, 705
Spry-XML-Datensatz 657
SpyBot 388
SQL 487
Abfrage 488
Datei 479
Dump 480
Tabellendatensatz ansprechen 629
SQL-Abfrage 625
eigene einbinden 625
suchen 640
Tabellenstruktur 626
verschachteln 632
Zuordnungstabelle 627
SQL-Befehl bearbeiten 509
Stammordner 101
Standard-Bilderordner 103
Statusleiste 75, 82
Stilvorgabe anwenden 208
Stilwiedergabe 222, 223
String 401
substr() 562
Subversion 109
Suche, GET-Arrays 640
Suchen und ersetzen 379
Suchformular 640
Suchfunktion 639
Suchmaschine 136, 173
dynamische Sites 384
Metaangabe 173
Suchmaschinenoptimierung 97, 131
SWF-File 354
switch 411
Symbolleiste, Kodierung 365
Synchronisieren, Dateien 122
Syntax 56
fehlerhafte 367

T

Tabelle 139
abfragen 505
Abmessung 146
aktualisieren 587
anwählen 142
Ausrichtung 145
auswählen 143, 147
Bemaßungen 142
Breite 144
dynamische 658
Eigenschaften 144
einfügen 140
erweiterter Modus 143
Größe festlegen 146
Höhe 144
Nachteil 139
Parameter 140
Spaltenbreiten und Zeilenhöhen 147
Umrandung 150
verändern 143
verschachteln 148
Vorteil 139
wiederholte 312
Zellauffüllung 145
Zelleigenschaft 146
Zellraum 145
Tabellenhöhe 149
Tabellenspalten
abfragen 509
mehrere abfragen 511
Tag
Attribute 374
automatisch vervollständigen 66
eigenes anlegen 377
suchen 379

Tag-Bibliothek 375
anlegen 374
bearbeiten 374
DTD importieren 676
Editor 374
Elemente-Definition 677
erstellen 376
Tag-Inspektor 374
Tag-Selektor 209
Tag-Vorgaben 374
target= 195, 196
Technik, fortgeschrittene 625
Template 127
Templates-Ordner 307
Testserver 99, 463
installieren 429
Testserver-Ansicht, Verbindungsskripte anzeigen 498
Testserverbetrieb, Ansicht 57
TEXT 473
Text
dynamischer 658
umfließt Bild 163
textarea 325
Textfeld 322
dynamisches 585, 658
Eigenschaften 322
verstecktes 323
Textformatierung, CSS 218
Textnavigation 393
TIME 474
TIMESTAMP 474
TINYBLOB 473
TINYINT 473
TINYTEXT 473
topmargin 135
TortoiseSVN 109
Tracing-Bild 84, 133
Transact-SQL 488
Transparentes GIF 148
Twitter 577

U

Überlappung verhindern 243
Überprüfen, eigene Abfragen 522
Übertragung simulieren 422
Umbruch für Auswahl 373
Umgebungsvariable 421
Ungültigen Code hervorheben 367
Unicode 74
Unsichtbare Elemente 75
Unterseite
schützen 613
Zugriffsschutz 613
UPDATE 483, 489
Upload, Dateigröße 481
URL-Parameter 518, 521, 545
erweitern 519
übergeben 518
URL-Präfix 459
URL-Variable 421
Usability 394
User-Tracking 388
UTF-8 133

V

Validator 77
VARCHAR 473
Variable 401, 403, 420
an URL anhängen 403
bereitstellen 421
definieren 421
externe anzeigen 426
mit Variablen arbeiten 420
nachträgliche Bearbeitung 422
neue hinzufügen 423
senden 422
übertragen 403
verketten 403
zur Fehlersuche nutzen 426
Variableninhalt 423
Variablentyp 421, 521
Variablenübergabe an SQL-Abfrage 518
Vektorgrafik 156
Verbindungsdaten eingeben 497
Verbindungsskript entfernen 499
Vererbung 213
Vergleichen von Dateien 116
Vergleichsoperatoren 405
Verhalten 183, 184, 185, 188
dynamisches 657
Effekte 190
Einfügeoptionen 651
fehlerhaftes 654
Verknüpfung 94, 193
einbauen 193

Verlauf 84
Verlaufsfenster 84
Verlinkung
dokument-relativ 103
root-relative 103
Verschachtelte Abfrage, Schema 635
Verschachtelte Vorlage erstellen 311
Verschachteltes Frameset 172
Versionierung 123
Versionskontrolle 109
Versteckte Datei anzeigen 498
Verstecktes Textfeld 323
Verzeichnisstrukturen 30
Video einbinden 356
Volltextsuche 639
Voreinstellungen 63
allgemein 63
Codeformat 65
Codehinweise 66
Codeumschreibung 67
CSS 66
Dateien vergleichen 69
Dateitypen/Editoren 69
neues Dokument 72
Vorkenntnisse 29
Vorlage 127, 305, 306
Adobe GoLive 308
aktualisieren 310
bearbeitbare Bereiche 308
bearbeitbare Bereiche definieren 306
Dokument aus Vorlage erstellen 309
entfernen 310
erstellen 306
kompatible 308
Mastervorlage 311
speichern 306
Tricks 311
verschachteln 311
verschachtelte 311
wiederholte Bereiche 311
wiederholte Tabelle 312
Vorschau in Dreamweaver 55

W

WAMP 430
installieren 432
WAMP-Entwicklungsserver 430
Warenkorb 386, 641
WCMS 385, 386
Web-Content-Management-System 385
WebDAV 106
Webdesign, barrierefreies 225
Weblog 388
Webserver 390, 391, 429, 433
Mac 431
Startdatei 99
starten 435
Zugangsdaten 413
Website
Aufbau 39
dynamische 385
Suchfunktion 639
Web-URL 103
Weiterleitung 613
Werbebanner mit Ebenen 239
Werte vergleichen 405, 595
WHERE 489, 515
while-Schleife 408
Wiederholte Tabelle 312
Wiederholter Bereich 311, 524
mit AP-Elementen 524
Parameter ändern 514
Windows 7, IIS 450
Windows Vista, PHP installieren 452
Windows XP 432
Word 361
Word-Dokument 361
einfügen 361
Word-Text 361
einfügen 361
WYSIWYG 31

X

XAMPP 432
installieren 433
konfigurieren 435
Startbildschirm 436
testen 435
unter Windows 7 433
XHTML 72, 73, 132
eigene Tags 376
XML 386, 665
automatisch nachladen 288
DTD 673
Processing Instructions 671
Tag selbst definieren 666

XML (Forts.)
Tag-Attibute 671
Überprüfung 675
und CSS 668
XML-Attribut 671
XML-Datei
im Browser 669
lokal ablegen 683
öffnen 670
tabellarisch ausgeben 705
XSLT anhängen 688
XML-Daten anbinden 695
XML-Dokument 201
Struktur 666
XMLHttpRequest-Objekt 695
XML-Quelle angeben 684
XPath 680
XPATH-Ausdrucksgenerator 685
XP-Firewall 432
XSLT 680
Datenquelle angeben 686
Stylesheet anhängen 688
XSLT-Dokument
anlegen 682
bedingte Bereiche 685
wiederholte Bereiche 685
XSLT-Fragment 682
XSL-Transformation 663
XSLT-Transformation
clientseitig 682
mit PHP 691
serverseitig 691

Y

YEAR 474
YouTube einbinden 358
YUI 88

Z

Zeichencodierung 74
Zeichenformat 473
Zeichenkette 401
teilen 562
Zeichensatz 133
Zeichenverkettung 402
Zeilenhöhe 147
Zeilennummer 428
einblenden 367
Zeilenumbruch 65
einfügen 221
Zeit- und Datumsformat 474
Zeitstempel 610
Zellauffüllung 145
Zelle 146
Auffüllung 141
Breite 147
Größe 146
Inhalt ausrichten 146
Raum 141
teilen 146
verbinden 146, 147
Zelleigenschaft 146
Zellraum 145
Zentrieren, AP-Element 263
Ziel angeben 195
Z-Index 245
Zoomwerkzeug 82, 83
Zugriff auf den Server 458
Zugriff auf Seite beschränken 613
Zugriffsebene 613, 617
definieren 614
in separater Tabelle speichern 617
Zugriffsrecht 394, 609
beschränken 613
steuern 611
Zuordnungstabelle 627
Zustandsvorschau 331