

Auf einen Blick

	Einführung.....	35
Teil 1	Grundlagen.....	41
Teil 2	HTML – Die Sprache des Internets	123
Teil 3	CSS – Layout fürs Internet	261
Teil 4	JavaScript.....	371
Teil 5	Perl – dynamisch und interaktiv I	475
Teil 6	PHP – dynamisch und interaktiv II.....	595
Teil 7	MySQL – Der Datenspeicher	721
Teil 8	Dynamische Bildgenerierung mit PHP und Perl	791
Teil 9	XML.....	863
Teil 10	Workshops	929
	Anhang	987
	Index.....	1125

Inhalt

Einführung	35
An wen richtet sich dieses Buch?	35
Was Sie schon können sollten	36
Was Sie mit diesem Buch lernen	36
Schreibweisen in diesem Buch	37
Danksagungen	38
Internetangebot	38
Vorwort zur 2. Auflage	38
Teil 1 Grundlagen	41
1 Arbeitsmittel	43
1.1 Editoren	43
1.2 Webserver	44
1.3 Perl	45
1.4 PHP	46
1.5 MySQL	46
1.6 phpMyAdmin	46
2 Das Internet	49
2.1 Entwicklung	49
2.2 Das statische Web	50
2.3 Das Plug-in-Web	50
2.3.1 JavaApplets	51
2.3.2 ActiveX-Controls	52
2.3.3 Macromedia Flash	52
2.4 Das dynamische Web	53
2.5 Internetdienste	55
2.5.1 E-Mail	55
2.5.2 Suchmaschinen	56
2.5.3 FTP	58

2.5.4	Telnet	59
2.5.5	Whois	59

3 Die »Netzüberflieger« 61

3.1	Internet Explorer	61
3.2	Netscape Navigator	62
3.3	Opera	64
3.4	Mozilla	65
3.5	FireFox alias Firebird alias Phönix	66
3.6	Camino	67
3.7	Konqueror	67
3.8	Safari	69
3.9	Lynx	70

4 Gute Vorbereitung ist wichtig 71

4.1	WAMP	71
4.1.1	Apache installieren	71
4.1.2	MySQL installieren	74
4.1.3	PHP installieren	76
4.1.4	Perl installieren	80
4.1.5	PhpMyAdmin installieren	81
4.2	LAMP unter SUSE 9.0	82
4.2.1	Apache2-Installation	83
4.2.2	MySQL installieren	86
4.3	XAMPP für Windows – ApacheFriends	87
4.4	XAMPP für Linux – ApacheFriends	89

5 Der Weg ins Internet 93

5.1	Wahl des Providers	93
5.1.1	AOL, T-Online & Co.	97
5.2	Upload der Seite	97
5.2.1	Grafischer Client unter Windows	97
5.2.2	Kommandozeilenbasierter Client unter Windows und UNIX/Linux	101
5.3	Suchmaschinen, Webkataloge und anderes	103

6 Adressierung 107

6.1	TCP/IP	107
6.1.1	Aufbau und Struktur	108
6.2	DNS	110
6.3	Adressierung, URIs	111
6.3.1	Ports	113
6.3.2	Protokolle	114
6.3.3	URI	114
6.4	Webserver zu Hause	115
6.4.1	Einwahl in das Internet	115
6.4.2	Routerkonfiguration	116
6.4.3	Server konfigurieren	117
6.5	Ausblick auf IPv6	118
6.5.1	Jedem Rechner eine öffentliche IP-Adresse	119
6.6	Umlaut-Domains	120

Teil 2 HTML – Die Sprache des Internets 123

7 HTML – Die Sprache des Internets 125

7.1	Am Anfang war HTML	125
7.2	HTML-Versionen	126
7.2.1	HTML 1	126
7.2.2	HTML 2	126
7.2.3	HTML 3.2	126
7.2.4	HTML 4 und 4.01	127
7.3	Das erste HTML-Dokument	127
7.4	Elemente und Tags	128
7.5	Hierarchie	129
7.6	Attribute	130
7.7	Kommentare und Quelltext-Strukturierung	131
7.8	Entities	132
7.9	Dokumententyp-Definition	133
7.9.1	Sprachvarianten	133
7.9.2	DTDs für andere HTML-Versionen und für XHTML	134
7.9.3	HTML-Vorlage	135
7.10	Zusammenfassung	135
7.11	Fragen und Übungen	135

8 Textstrukturierung 137

8.1	Absätze	137
8.2	Textausrichtung	138
8.3	Zeilenumbrüche	140
8.4	Vorformatierter Text	141
8.5	Überschriften	142
8.6	Physische Textauszeichnung	144
8.7	Logische Textauszeichnung	146
8.8	Schriftformatierung	148
8.8.1	Vererbung	149
8.8.2	Schriftart	150
8.8.3	Schriftgröße	151
8.8.4	Schriftfarbe	151
8.9	Dateiweite Farben	155
8.9.1	Hintergrund und Textfarbe	155
8.9.2	Hyperlink-Farben	155
8.10	Listen	156
8.10.1	Numerierte Listen	156
8.10.2	Aufzählungslisten	158
8.10.3	Definitionslisten	160
8.10.4	Listenattribute	162
8.11	Zusammenfassung	163
8.12	Fragen und Übungen	164

9 Tabellen 165

9.1	Aufbau und Strukturierung	165
9.2	Die wichtigsten Elemente	166
9.3	Größe, Rahmen und Abstände	169
9.4	Tabellen und Inhalte ausrichten	170
9.4.1	Tabelle horizontal ausrichten	171
9.4.2	Zelleninhalt horizontal ausrichten	171
9.4.3	Zelleninhalt vertikal ausrichten	172
9.4.4	Ausrichtung an Dezimalzeichen	172
9.4.5	Beispiel für die Ausrichtung von Zelleninhalten	173
9.5	Farben und Schrift	174
9.5.1	Tabellenhintergrund	174
9.5.2	Zellenhintergrund	175
9.5.3	Schriftformatierung	175
9.5.4	Farb- und Schriftbeispiel	176
9.6	Zellenverbund	177

9.6.1	Über- und nebeneinander liegende Zellen verbinden	183
9.7	Gruppierungen	185
9.7.1	Spaltengruppen	185
9.7.2	Logische Gruppierung	186
9.8	Tabellen als Designmittel	188
9.9	Zusammenfassung	191
9.10	Fragen und Übungen	191

10 Grafiken und Multimedia 193

10.1	Grafikformate	193
10.1.1	GIF	193
10.1.2	JPEG	195
10.1.3	PNG	196
10.1.4	SVG	197
10.2	Grafiken einbinden	197
10.2.1	Alternativ-Text	198
10.2.2	Breite und Höhe	199
10.2.3	Rahmen	199
10.2.4	Grafiken ausrichten	200
10.3	Transparente Grafiken	200
10.3.1	Blind- oder Fake-GIFs	202
10.4	Hintergründe	202
10.5	Flash-Filme	203
10.6	JavaApplets	206
10.6.1	Das applet-Element	206
10.6.2	Das object-Element	208
10.7	Grafiken mit dem object-Element einbinden	208
10.8	Zusammenfassung	210
10.9	Fragen und Übungen	211

11 Geknüpftes Netz 213

11.1	Aufbau einer Verknüpfung	213
11.2	Lokale Links	213
11.2.1	Gleiches Verzeichnis	214
11.2.2	Übergeordnete Verzeichnisse	214
11.2.3	Anderer Verzeichniszweig	215
11.3	Globale Links	215
11.4	E-Mail & Co.	215
11.4.1	Andere Webdienste	217
11.5	Grafiken als Links	217

Teil 3 CSS – Layout fürs Internet **261**

15 CSS – Layout fürs Internet **263**

15.1	CSS – Make-up fürs Web ...	263
15.2	Wie funktioniert CSS?	263
15.3	CSS-Versionen	264
15.4	Direkte Formatierung	265
15.5	Das erste Stylesheet	266
15.6	Zentrale Formatierung	267
15.7	Externe Formatierung	270
15.8	Ausgabemedien	272
15.9	Einheiten	273
15.10	Zusammenfassung	274
15.11	Fragen und Übungen	275

16 Textformatierung **277**

16.1	Farben in CSS	277
16.2	Schriftformatierung	280
16.2.1	Schriftart bzw. -typ	280
16.2.2	Schriftgröße	282
16.2.3	Schriftneigung und -variante	283
16.2.4	Schriftdicke	284
16.2.5	Schriftfarbe	286
16.2.6	Wort- und Zeichenabstände	287
16.2.7	Textdekoration	288
16.2.8	Texttransformation	290
16.2.9	Kurznotation zur Schriftformatierung	291
16.3	Schriftartendateien	292
16.4	Spezielle Formatierungen	296
16.4.1	Klassen	300
16.4.2	Verschachtelte Elemente	301
16.4.3	Individuelle Formate	303
16.5	Hintergrundfarbe	305
16.6	Hintergrundbilder	306
16.7	Zusammenfassung	310
16.8	Fragen und Übungen	311

17 Allgemeine Formatierung 313

17.1	Die Elemente div und span	313
17.2	Außenabstand	315
17.3	Innenabstand	317
17.4	Rahmen	319
17.4.1	Detaillierte Variante	321
17.5	Positionierung	323
17.6	Anzeige	325
17.7	Zusammenfassung	329
17.8	Fragen und Übungen	329

18 Tabellen und Listen 331

18.1	Tabellenformatierung	331
18.1.1	Tabellenlayout	331
18.1.2	Rahmenlayout	333
18.1.3	Ausrichtung der Überschrift	335
18.1.4	Rahmenabstand	336
18.1.5	Darstellung leerer Zellen	337
18.2	Listenformatierung	339
18.2.1	Aufzählungszeichen	339
18.2.2	Einrückung	341
18.2.3	Ein Bild als Aufzählungszeichen	343
18.2.4	Eigenschaften zusammenfassen	344
18.3	Zusammenfassung	345
18.4	Fragen und Übungen	346

19 Pseudoformate 347

19.1	Verweise	347
19.2	Absätze	349
19.3	Automatischer Text	350
19.4	Automatische Nummerierung	354
19.5	Zusammenfassung	357
19.6	Fragen und Übungen	357

20 Was sonst noch wichtig ist 359

20.1	Cursor	359
20.2	Scrollbar	360
20.3	Special Effects	362
20.3.1	Alpha	362
20.3.2	Blur	363
20.3.3	Chroma	363
20.3.4	DropShadow	363
20.3.5	FlipH	364
20.3.6	FlipV	364
20.3.7	Glow	364
20.3.8	Gray	364
20.3.9	Invert	364
20.3.10	Mask	365
20.3.11	Shadow	365
20.3.12	Wave	365
20.3.13	XRay	366
20.3.14	Weitere Informationen	366
20.4	Printmedien	366
20.4.1	Medientyp festlegen	366
20.4.2	Seitengröße und -ränder	367
20.4.3	Seitenumbruch	367
20.4.4	Allein stehende Zeilen	368
20.4.5	Schnittmarken	368
20.4.6	Linke, rechte und erste Seite	368
20.5	Zusammenfassung	369
20.6	Fragen und Übungen	369

Teil 4 JavaScript 371

21 Grundlagen 373

21.1	Java und JavaScript	374
21.2	JavaScript in HTML	375
21.3	Das erste JavaScript	376
21.4	JavaScript in Dateien	378
21.5	Kommentare	379
21.6	Zusammenfassung	380
21.7	Fragen und Übungen	380

22 Variablen, Werte und Operatoren 381

22.1	Variablen	381
22.2	Werte	382
22.3	Operatoren	383
22.4	Einfacher Passwortschutz	387
22.5	Bedingte Anweisungen – if	388
22.6	Fallunterscheidung – switch	390
22.7	Zusammenfassung	391
22.8	Fragen und Übungen	392

23 Funktionen und Schleifen 393

23.1	Funktionen definieren	393
23.2	Funktionen aufrufen	394
23.3	Funktionen und Parameter	394
23.4	Rückgabewerte	396
23.5	Schleifen	397
23.5.1	Die while-Schleife	397
23.5.2	Die do-while-Schleife	399
23.5.3	Die for-Schleife	399
23.6	Zusammenfassung	400
23.7	Fragen und Übungen	401

24 Objektorientierung 403

24.1	Eigenschaften	404
24.2	Methoden	405
24.3	Objekthierarchie	406
24.4	Objekte instanziiieren	406
24.5	Mehrere Anweisungen	409
24.6	Eigene Klassen	410
24.7	Zusammenfassung	412
24.8	Fragen und Übungen	412

25 Datums- und Zeitfunktionen 413

25.1	Formatierte Datumsausgabe	413
25.1.1	Arrays	413
25.1.2	Monatsnamen ermitteln	414
25.1.3	Wochentag ermitteln	415
25.1.4	Vierstellige Jahreszahl	415
25.1.5	Kombinierte Ausgabe	416
25.2	Führende Nullen	417
25.3	Besuchsdauer	418
25.4	Countdown	420
25.4.1	Verbesserungen	422
25.5	Datum validieren	422
25.5.1	Schaltjahr	422
25.5.2	Zahlenbereich	423
25.6	Das Jahr 2000	425
25.7	Zusammenfassung	426
25.8	Fragen und Übungen	426

26 Informationen vom Browser 427

26.1	Welcher Browser?	427
26.1.1	Detailinformationen	428
26.1.2	Browser identifizieren	430
26.2	Bildschirm	431
26.2.1	Farb- und Pixeltiefe	434
26.3	Plug-ins	436
26.3.1	Auf ein Plug-in prüfen	438
26.4	Zusammenfassung	440
26.5	Fragen und Übungen	440

27 Zeichenkettenfunktionen 441

27.1	Länge	441
27.2	Groß- und Kleinschreibung	442
27.3	Zeichenposition	442
27.4	Teilzeichen	444
27.5	Teilstring	445
27.5.1	Variante 1	445
27.5.2	Variante 2	445

27.6	String zerlegen	446
27.7	Zusammenfassung	446
27.8	Fragen und Übungen	447

28 DHTML 449

28.1	HTML wird dynamisch	449
28.2	Kompatibilität	449
28.3	Internet Explorer-DOM	450
28.4	Netscape-DOM	451
28.5	W3C-DOM	452
28.6	Problematik und Lösung	455
28.7	Ausklappbare Navigationsleiste	456
28.8	Zusammenfassung	458
28.9	Fragen und Übungen	458

29 Was sonst noch wichtig ist 459

29.1	Ereignisbehandlung	459
29.1.1	onAbort	460
29.1.2	onBlur	461
29.1.3	onChange	461
29.1.4	onClick	461
29.1.5	onDbClick	462
29.1.6	onError	462
29.1.7	onFocus	462
29.1.8	onKeyDown	462
29.1.9	onKeyPress	462
29.1.10	onKeyUp	462
29.1.11	onLoad	463
29.1.12	onMouseDown	463
29.1.13	onMouseMove	463
29.1.14	onMouseout und onMouseover	464
29.1.15	onMouseup	464
29.1.16	onReset und onSubmit	464
29.1.17	onResize	465
29.1.18	onSelect	465
29.1.19	onUnload	465
29.2	Cookies	466
29.2.1	Cookie schreiben	466
29.2.2	Cookie auslesen	467
29.2.3	Verfallsdatum	467
29.2.4	Persönliche Seitenbesuche zählen	468
29.3	Fehlerbehandlung	471

29.4	Zusammenfassung	474
29.5	Fragen und Übungen	474

Teil 5 Perl – dynamisch und interaktiv I **475**

30 Perl **477**

30.1	Perl ist nicht CGI	478
30.2	Das erste Perl-Script	478
30.3	Textausgabe	479
30.4	Die erste Zeile	480
30.5	Notwendige Anweisungen	481
30.6	Kommentare und Quellstrukturierung	482
30.7	Zusammenfassung	482
30.8	Fragen und Übungen	483

31 Variablen und Operatoren **485**

31.1	Skalare	485
31.2	Arrays	486
31.2.1	Zugriff auf einzelne Elemente	488
31.3	Hashes	489
31.3.1	Zugriff auf einzelne Elemente	489
31.3.2	Unbekannte Schlüssel ermitteln	490
31.4	Vordefinierte Variablen	491
31.5	Operatoren	492
31.5.1	Zuweisungsoperator	492
31.5.2	Berechnungsoperatoren	492
31.5.3	Vergleichsoperatoren	493
31.5.4	Logische Operatoren	495
31.5.5	Zeichenkettenoperator	496
31.6	Zusammenfassung	496
31.7	Fragen und Übungen	497

32 Subroutinen **499**

32.1	Was sind Subroutinen?	499
32.2	Subroutinen erstellen	499
32.3	Subroutinen aufrufen	500

32.4	Local und my	501
32.4.1	Doppeldefinition	503
32.5	Parameter übergeben	505
32.5.1	Elegantere Methode	506
32.6	Rückgabewerte	506
32.7	Zusammenfassung	508
32.8	Fragen und Übungen	508

33 Ablaufsteuerung 509

33.1	If und unless	509
33.1.1	elsif	510
33.1.2	unless	511
33.2	do-Schleife	511
33.2.1	Variante mit while	512
33.3	for-Schleife	512
33.4	while-Schleife	513
33.4.1	while und Hashes	513
33.5	foreach-Schleife	514
33.5.1	Spezielle Notation	515
33.6	Schleifensteuerung	516
33.6.1	Durchlauf wiederholen	516
33.6.2	Schleife abbrechen	516
33.6.3	Durchlauf überspringen	517
33.7	Zusammenfassung	517
33.8	Fragen und Übungen	517

34 Standardfunktionen 519

34.1	Zeichenkettenfunktionen	519
34.1.1	Groß- und Kleinbuchstaben	519
34.1.2	Zeichenwerte	520
34.1.3	Länge einer Zeichenkette	520
34.1.4	Teilzeichenkette suchen	521
34.1.5	Teilzeichenkette extrahieren	521
34.1.6	Zeichenkette verschlüsseln	522
34.2	Hash- und Listenfunktionen	523
34.2.1	Wert löschen	523
34.2.2	Werte und Schlüssel eines Hashs	524
34.2.3	Elemente aus einem Hash entfernen	524
34.2.4	Überprüfen, ob ein Element existiert	525
34.2.5	Alle Elemente eines Hashs ausgeben	525

34.2.6	Ein Element am Anfang einer Liste hinzufügen oder löschen	526
34.2.7	Ein Element am Ende einer Liste hinzufügen oder löschen	526
34.2.8	Eine Liste sortieren oder umkehren	527
34.2.9	Liste in Zeichenkette konvertieren und umgekehrt	527
34.2.10	Eine Zeichenkettenfunktion auf alle Listenelemente anwenden	528
34.3	Datums- und Zeitfunktionen	528
34.4	Mathematische Funktionen	533
34.4.1	Zufallszahlen	533
34.4.2	Sinus und Kosinus	534
34.4.3	Wurzelberechnung	535
34.5	Umwandlungsfunktionen	535
34.5.1	Hexadezimal in dezimal	535
34.5.2	Oktalzahl in dezimal	536
34.5.3	Zeichenkette in Zahlenwert umwandeln	536
34.5.4	Absoluter Zahlenwert	538
34.6	Zusammenfassung	538
34.7	Fragen und Übungen	538

35 Ein- und Ausgabe 539

35.1	Parameterübergabe	539
35.1.1	Modularisierung	542
35.2	Formulare	543
35.2.1	Daten dekodieren	546
35.3	Cookies	548
35.3.1	Ein Cookie schreiben	550
35.3.2	Ein Cookie lesen	551
35.4	Zusammenfassung	552
35.5	Fragen und Übungen	552

36 Dateisystem 553

36.1	Verzeichnisliste	553
36.2	Rekursion	556
36.3	Weitere Verzeichnisfunktionen	561
36.3.1	Verzeichnis wechseln	561
36.3.2	Verzeichnis erstellen	561
36.3.3	Verzeichnis löschen	562
36.4	Datei lesen	562
36.4.1	Praxisanwendung	563
36.5	Datei schreiben	565
36.6	Datei-Uploads	569

36.7	Zusammenfassung	572
36.8	Fragen und Übungen	573

37 Reguläre Ausdrücke 575

37.1	Was sind reguläre Ausdrücke?	575
37.2	Reguläre Ausdrücke verwenden	575
37.3	Metazeichen	576
37.4	Quantifier	578
37.5	Gruppierung	580
37.6	Zeichenklassen	580
37.6.1	Abkürzungen	581
37.7	Alternativen	582
37.8	Flags	582
37.9	Suchen und Ersetzen	583
37.10	Zusammenfassung	584
37.11	Fragen und Übungen	584

38 Was sonst noch wichtig ist 585

38.1	CGI und SSI	585
38.1.1	Ausgabe von Variablen	585
38.1.2	Konfigurationen	587
38.1.3	Programm ausführen	589
38.2	CGI-Umgebungsvariablen	591
38.3	Zusammenfassung	593
38.4	Fragen und Übungen	593

Teil 6 PHP – dynamisch und interaktiv II 595

39 PHP 597

39.1	Das erste PHP-Script	598
39.2	PHP und die Dateieindungen	599
39.3	PHP in eigenen Dateien	600
39.4	Quellstrukturierung und Kommentare	601
39.5	Textausgabe	601
39.5.1	Formatierte Textausgabe	602
39.6	Alternative PHP-Tags	603

39.7	Zusammenfassung	604
39.8	Fragen und Übungen	604

40 Variablen und Operatoren 605

40.1	Variable	605
40.1.1	Variablen bezeichnen	605
40.2	Datentypen	606
40.2.1	Integer	606
40.2.2	Fließkommazahl	607
40.2.3	Boolean	607
40.2.4	String	607
40.2.5	Nützliche Funktionen	609
40.2.6	Typumwandlung	609
40.3	Arrays	610
40.4	Resource und NULL	612
40.5	Operatoren	612
40.5.1	Arithmetische Operatoren	612
40.5.2	Vergleichsoperatoren	614
40.5.3	Zeichenkettenoperatoren	615
40.5.4	Logische Operatoren	616
40.6	Zusammenfassung	616
40.7	Fragen und Übungen	617

41 Programmsteuerung 619

41.1	if-Bedingung	619
41.2	do...while-Schleife	620
41.3	while-Schleife	621
41.4	for-Schleife	622
41.5	foreach-Schleife	623
41.6	switch-Anweisung	624
41.7	Einbinden von Dateien	626
41.8	Schleifen steuern	627
41.9	Alternative Syntax	628
41.9.1	if	628
41.9.2	while	628
41.9.3	for	629
41.9.4	foreach	629
41.9.5	switch	629
41.10	Zusammenfassung	630
41.11	Fragen und Übungen	630

42 Objekte und Klassen 631

42.1	Objekte	631
42.2	Funktionen	631
42.3	Klassen	633
42.4	Objekte instanzieren	635
42.5	Konstruktoren	635
42.6	Vererbungslehre	637
42.7	Zusammenfassung	638
42.8	Fragen und Übungen	639

43 Standardfunktionen 641

43.1	Zeichenkettenfunktionen	641
43.1.1	Groß- und Kleinschreibung	641
43.1.2	Zeichenwerte	642
43.1.3	Länge einer Zeichenkette	642
43.1.4	Teilzeichenkette suchen	643
43.1.5	Teilzeichenkette auslesen	644
43.1.6	Teilzeichenkette ersetzen	644
43.1.7	Zeichenkette verschlüsseln	645
43.2	Array-Funktionen	645
43.2.1	Größe eines Arrays	646
43.2.2	Arrays sortieren	646
43.2.3	Array mit Wertebereich	647
43.2.4	Element am Ende hinzufügen oder löschen	648
43.2.5	Elemente am Anfang hinzufügen oder löschen	649
43.2.6	Array durchsuchen	650
43.2.7	Schlüssel und Werte	650
43.3	Datums- und Zeitfunktionen	651
43.3.1	Der aktuelle Zeitpunkt	651
43.3.2	Ausgabe formatieren	653
43.3.3	Datum umwandeln	654
43.4	Mathematische Funktionen	654
43.4.1	Zufallsgenerator	655
43.4.2	Zahlensysteme	656
43.4.3	Sinus und Kosinus	656
43.5	Zusammenfassung	657
43.6	Fragen und Übungen	657

44 Ein- und Ausgabe 659

44.1	Parameterübergabe	659
44.2	Formulare	662
44.2.1	GET-Methode	662
44.2.2	POST-Methode	663
44.2.3	Checkboxen, Radiobuttons und Auswahllisten	664
44.3	Cookies	666
44.3.1	Cookie schreiben	666
44.3.2	Cookie lesen	667
44.4	Datei-Upload	667
44.5	Zusammenfassung	671
44.6	Fragen und Übungen	671

45 Dateisystem 673

45.1	Verzeichnisliste	673
45.1.1	Verzeichnis oder Datei	674
45.1.2	Alternative	675
45.1.3	Weitere Verzeichnisfunktionen	676
45.2	Rekursion	676
45.3	Datei schreiben	678
45.4	Datei lesen	679
45.5	Dateieigenschaften	682
45.6	Dateisystemoperationen	684
45.7	Zusammenfassung	684
45.8	Fragen und Übungen	685

46 Sitzungen 687

46.1	Was sind Sessions?	687
46.2	Session erzeugen	688
46.3	Mit Variablen arbeiten	689
46.4	Session beenden	690
46.5	Weitere Session-Funktionen	690
46.6	Zusammenfassung	691
46.7	Fragen und Übungen	691

47 Was sonst noch wichtig ist 693

47.1	Server-Informationen	693
47.2	Netzwerkfunktionen	694
47.2.1	IP-Adressen und DNS	694
47.2.2	Verbindungen zu anderen Servern	694
47.2.3	HTTP-Verbindungen	696
47.2.4	POP3-Verbindung	697
47.3	Perl-kompatible reguläre Ausdrücke in PHP	698
47.4	PDF-Dokumente erzeugen	700
47.4.1	Dokument mit Inhalt füllen	702
47.5	Zusammenfassung	703
47.6	Fragen und Übungen	703

48 PHP 5 – Was ist neu? 705

48.1	Objektorientierung	705
48.2	Konstruktor und Destruktor	707
48.3	Fehlerbehandlung	710
48.3.1	Die herkömmlichen Varianten	711
48.3.2	Die elegante Lösung	712
48.3.3	Eigene Fehlertypen	714
48.4	SimpleXML	717
48.5	Sonstiges	720
48.6	Zusammenfassung	720
48.7	Fragen und Übungen	720

Teil 7 MySQL – Der Datenspeicher 721

49 MySQL – Der Datenspeicher für Ihre Internetseiten 723

49.1	Mein SQL gib mir heute	723
49.2	Datenbanktypen	723
49.2.1	Relationale Datenbanksysteme	724
49.2.2	Objektorientierte Datenbanksysteme	724
49.3	Redundanz und Inkonsistenz	725
49.4	Zusammenfassung	726
49.5	Fragen und Übungen	727

50 Die Sprache SQL 729

50.1	Einstieg	729
50.2	Namenskonventionen	730
50.3	Datenbank erstellen, löschen oder auswählen	731
50.4	Tabellen erstellen und löschen	732
50.5	Tabellen verändern	734
50.6	MySQL-Datentypen	735
50.6.1	Numerische Typen	735
50.6.2	Zeichen- und Zeichenketten-Typen	735
50.6.3	Vermischte Typen	736
50.6.4	Was ist ein BLOB?	736
50.6.5	Optionen	736
50.7	Zusammenfassung	737
50.8	Fragen und Übungen	738

51 Datenaustausch 739

51.1	SELECT	739
51.1.1	WHERE	742
51.1.2	ORDER BY	745
51.1.3	GROUP BY	746
51.1.4	DISTINCT	747
51.1.5	Numerische Ausdrücke	748
51.2	INSERT	749
51.3	UPDATE	750
51.4	DELETE	751
51.5	Tabellen importieren und exportieren	751
51.5.1	Export	751
51.5.2	Import	752
51.6	Tabellenverknüpfung	753
51.6.1	Aliase	754
51.6.2	JOIN	755
51.7	Zusammenfassung	755
51.8	Fragen und Übungen	756

52 MySQL und Perl 757

52.1	Vorbereitungen	757
52.2	Datenbankverbindung herstellen	758
52.3	Anfragen stellen	760

52.4	Ergebnisverarbeitung bei SELECT-Anweisungen	763
52.4.1	Datensätze lesen	763
52.5	Ergebnisverarbeitung bei Anweisungen, die keine Ergebnismenge liefern	766
52.6	Datentypen	768
52.6.1	Zeichenketten	769
52.6.2	Datums- und Zeitangaben	770
52.7	Fehlerbehandlung	771
52.8	Metainformationen	772
52.9	Zusammenfassung	773
52.10	Fragen und Übungen	773

53 MySQL und PHP 775

53.1	Datenbankverbindung herstellen	775
53.2	Datenbank auswählen	776
53.2.1	Konfigurationsdatei	777
53.3	Anfragen stellen	778
53.4	Ergebnisverarbeitung bei SELECT oder SHOW	780
53.5	Ergebnisverarbeitung bei anderen Anweisungen	782
53.6	Datentypen	783
53.6.1	Zeichenketten	783
53.6.2	Datums- und Zeitangaben	784
53.7	Fehlerbehandlung	785
53.8	Metainformationen	786
53.8.1	Datenbanken	786
53.8.2	Tabellen	787
53.8.3	Felder bzw. Spalten	787
53.9	Zusammenfassung	789
53.10	Fragen und Übungen	790

Teil 8 Dynamische Bildgenerierung mit PHP und SVG 791

54 Dynamische Bildgenerierung – PHP-Variante 793

54.1	Die GD-Library und PHP	793
54.2	Eine erste dynamische Grafik	794
54.2.1	Farben	794
54.2.2	Text einfügen	795
54.2.3	Ausgabe des Bildes	796

54.3	Grafiken verändern	798
54.3.1	Abmessungen	798
54.4	Zeichnen	800
54.4.1	Punkte und Linien	800
54.4.2	Rechtecke	801
54.4.3	Kreise und Kreisbögen	802
54.4.4	Flächen füllen	805
54.4.5	Transparenz	806
54.5	Erweiterte Textausgabe	807
54.6	Thumbnails erzeugen	810
54.7	Anwendungsbeispiele	814
54.7.1	Kreisdiagramm	814
54.7.2	Zeichensalat	819
54.8	Zusammenfassung	822
54.9	Fragen und Übungen	823

55 Dynamische Bildgenerierung – Perl-Variante 825

55.1	Die GD-Library und Perl	825
55.1.1	Manuelle Installation	827
55.2	Erste Versuche	827
55.2.1	Farben	828
55.2.2	Text einfügen	829
55.2.3	Ausgabe des Bildes	831
55.3	Grafiken verändern	834
55.4	In einer Grafik zeichnen	836
55.4.1	Punkte und Linien	836
55.4.2	Rechtecke	838
55.4.3	Kreise und Kreisbögen	838
55.4.4	Flächen füllen	841
55.4.5	Transparenz	843
55.5	Erweiterte Textausgabe	845
55.6	Thumbnails erzeugen	848
55.7	Anwendungsbeispiele	851
55.7.1	Verlaufdiagramm	851
55.7.2	Grafik als Textdatei	857
55.8	Zusammenfassung	862
55.9	Fragen und Übungen	862

Teil 9 XML 863

56 XML – Eine Einleitung 865

56.1	Was ist XML?	865
56.2	Elemente und Attribute	866
56.2.1	Processing Instructions	867
56.3	Strukturierung, Bezeichnung und Kommentare	867
56.3.1	Bezeichnung	868
56.3.2	Kommentierung	870
56.4	DTD und Schema	870
56.5	Zusammenfassung	871
56.6	Fragen und Übungen	871

57 XHTML – Die nächste Generation von HTML 873

57.1	Neu – und doch altbekannt	873
57.2	Erforderliche Angaben	875
57.3	XHTML validieren	876
57.4	Zusammenfassung	878
57.5	Fragen und Übungen	878

58 SVG – Scalable Vector Graphics 879

58.1	Was sind Vektoren?	879
58.2	Das erste SVG-Dokument	881
58.3	SVG in HTML	882
58.4	Grundformen	884
58.4.1	Text ausgeben	884
58.4.2	Linien	886
58.4.3	Rechtecke	887
58.4.4	Kreise und Ellipsen	888
58.5	Farben	889
58.5.1	Transparenz	889
58.5.2	Lineare Farbverläufe	890
58.5.3	Radiale Farbverläufe	891
58.6	Pfade	892
58.6.1	Einfache Pfade	893
58.6.2	Kurven	894
58.6.3	Kreisbögen	895
58.7	SVG mit PHP	899

58.8	Zusammenfassung	904
58.9	Fragen und Übungen	905

59 RDF/RSS 907

59.1	Was ist RDF/RSS?	907
59.2	RDF/RSS-Dokumente einsetzen	908
59.3	Zusammenfassung	911
59.4	Fragen und Übungen	911

60 Webservices 913

60.1	Was sind Webservices?	913
60.2	Die unterschiedlichen Varianten	913
60.2.1	XML-RPC	913
60.2.2	SOAP	914
60.2.3	WSDL	916
60.3	Webservices nutzen	916
60.3.1	eBay Developer Program	916
60.3.2	Grundlagen	918
60.3.3	Beispiel	921
60.4	Zusammenfassung	926
60.5	Fragen und Übungen	927

Teil 10 Workshops 929

61 Workshop »Content Management System« 931

61.1	Das Ziel	931
61.2	Wählen Sie jetzt!	934
61.3	Strukturierung	935
61.3.1	Verzeichnisschutz	935
61.4	Zusammenfassung	938

62 Die Datenbank 939

62.1	Datenbankstruktur planen und festlegen	939
62.2	Tabellenstruktur planen und erzeugen	939
62.2.1	Planung	939
62.2.2	Erzeugen	942

62.3	Erste Einträge anlegen	943
62.4	Zusammenfassung	943

63 Das Design 945

63.1	Die Struktur	945
63.2	HTML-Vorlage	946
63.3	Zusammenfassung	948

64 Die Programmierung 949

64.1	Die Konfigurationsdateien	949
64.1.1	base.inc.php	949
64.1.2	database.inc.php	949
64.1.3	Sicherheit	949
64.1.4	functions.inc.php	950
64.2	Die index.php	950
64.3	Die Datei functions.inc.php	953
64.4	Die Datei base.inc.php	955
64.5	Die Datei database.inc.php	956
64.6	Das Projekt im Einsatz	956

65 Administrationsbereich 957

65.1	Verzeichnisse und Dateien	957
65.2	Anmelden im Adminbereich	958
65.2.1	Das Script login.php	960
65.2.2	Abmelden	963
65.3	Anmerkungen	964

66 Der Schluss 965

66.1	Verbesserungsmöglichkeiten	965
66.2	Das Projekt installieren	965

67 Workshop »Newsfeed« 967

67.1	Die newsfeed-Klasse	967
67.1.1	Der Konstruktor	968

67.1.2	Die Methode addItem	973
67.1.3	Die Methode getItem	974
67.1.4	Die Methode getItem	974
67.1.5	Die Methode getItemSelective	975
67.2	Die Klasse einsetzen	976
67.3	Anmerkungen	977

68 Workshop »Weblog« 979

68.1	Was sind Weblogs?	979
68.2	Einen Weblog einrichten	980
68.2.1	Serendipity installieren	980
68.2.2	Serendipity nutzen	982
68.2.3	Serendipity verwalten	984
68.3	Anmerkungen	985

Anhang 987

A Antworten und Lösungen 989

A.1	Teil 2 – HTML	989
A.1.1	Lösungen zu Abschnitt 7.11	989
A.1.2	Lösungen zu Abschnitt 8.12	990
A.1.3	Lösungen zu Abschnitt 9.10	991
A.1.4	Lösungen zu Abschnitt 10.9	996
A.1.5	Lösungen zu Abschnitt 11.10	997
A.1.6	Lösungen zu Abschnitt 12.9	998
A.1.7	Lösungen zu Abschnitt 13.8	1001
A.2	Teil 3 CSS – Layout fürs Internet	1001
A.2.1	Lösungen zu Abschnitt 15.11	1001
A.2.2	Lösungen zu Abschnitt 16.8	1002
A.2.3	Lösungen zu Abschnitt 17.8	1003
A.2.4	Lösungen zu Abschnitt 18.4	1005
A.2.5	Lösungen zu Abschnitt 19.6	1006
A.2.6	Lösungen zu Abschnitt 20.6	1006
A.3	Teil 4 – JavaScript	1007
A.3.1	Lösungen zu Abschnitt 21.7	1007
A.3.2	Lösungen zu Abschnitt 22.8	1008
A.3.3	Lösungen zu Abschnitt 23.7	1009
A.3.4	Lösungen zu Abschnitt 24.8	1010
A.3.5	Lösungen zu Abschnitt 25.8	1011
A.3.6	Lösungen zu Abschnitt 26.5	1012
A.3.7	Lösungen zu Abschnitt 27.8	1013
A.3.8	Lösungen zu Abschnitt 28.9	1014
A.3.9	Lösungen zu Abschnitt 29.5	1014

A.4	Teil 5 – Perl – dynamisch und interaktiv I	1015
A.4.1	Lösungen zu Abschnitt 30.8	1015
A.4.2	Lösungen zu Abschnitt 31.7	1016
A.4.3	Lösungen zu Abschnitt 32.8	1017
A.4.4	Lösungen zu Abschnitt 33.8	1018
A.4.5	Lösungen zu Abschnitt 34.7	1019
A.4.6	Lösungen zu Abschnitt 35.5	1021
A.4.7	Lösungen zu Abschnitt 36.8	1022
A.4.8	Lösungen zu Abschnitt 37.11	1025
A.4.9	Lösungen zu Abschnitt 38.4	1026
A.5	Teil 6 – PHP – dynamisch und interaktiv II	1027
A.5.1	Lösungen zu Abschnitt 39.8	1027
A.5.2	Lösungen zu Abschnitt 40.7	1028
A.5.3	Lösungen zu Abschnitt 41.11	1029
A.5.4	Lösungen zu Abschnitt 42.8	1030
A.5.5	Lösungen zu Abschnitt 43.6	1032
A.5.6	Lösungen zu Abschnitt 44.6	1034
A.5.7	Lösungen zu Abschnitt 45.8	1035
A.5.8	Lösungen zu Abschnitt 46.7	1037
A.5.9	Lösungen zu Abschnitt 47.6	1038
A.5.10	Lösungen zu Abschnitt 48.7	1040
A.6	Teil 7 – MySQL – Datenspeicher	1041
A.6.1	Lösungen zu Abschnitt 49.5	1041
A.6.2	Lösungen zu Abschnitt 50.8	1042
A.6.3	Lösungen zu Abschnitt 51.8	1042
A.6.4	Lösungen zu Abschnitt 52.10	1044
A.6.5	Lösungen zu Abschnitt 53.10	1047
A.7	Teil 8 – Dynamische Bildgenerierung mit PHP und Perl	1050
A.7.1	Lösungen zu Abschnitt 54.9	1050
A.7.2	Lösungen zu Abschnitt 55.9	1052
A.8	Teil 9 – XML	1054
A.8.1	Lösungen zu Abschnitt 56.6	1054
A.8.2	Lösungen zu Abschnitt 57.5	1055
A.8.3	Lösungen zu Abschnitt 58.9	1055
A.8.4	Lösungen zu Abschnitt 59.4	1058
A.8.5	Lösungen zu Abschnitt 60.5	1059

B Rechtliche Fragen rund um Webseiten 1061

B.1	Urheberrecht	1061
B.2	Zivilrecht	1062
B.3	Das Recht am eigenen Bild	1063
B.4	Teledienstegesetz	1064
B.5	Strafrecht	1065
B.6	Anmerkungen	1065

C Referenztabelle **1067**

C.1	ASCII-Zeichentabelle	1067
C.2	Zahenumrechnungstabelle	1069
C.2.1	Binärsystem	1070
C.2.2	Oktalsystem	1071
C.2.3	Hexadezimalsystem	1072
C.3	Rechnen mit bitweisen Operatoren	1073
C.4	Farbworte	1076
C.5	Websichere Farben	1079
C.6	MIME-Typen	1080
C.7	Sprachenkürzel	1083
C.8	MySQL-Fehlernummern	1085

D Reservierte Wörter **1093**

D.1	JavaScript	1093
D.2	Perl	1094
D.3	PHP	1096

E Installationsanleitungen **1099**

E.1	Microsoft Personal Webserver	1099
E.1.1	Den Server installieren	1099
E.1.2	PHP installieren	1100
E.1.3	Perl installieren	1101
E.1.4	MySQL installieren	1101
E.2	Microsoft Internet Information Server	1102
E.2.1	Den Server installieren	1102
E.2.2	PHP installieren	1103
E.2.3	Perl installieren	1103
E.2.4	MySQL installieren	1104
E.3	OmniHTTP	1105
E.3.1	Den Server installieren	1105
E.3.2	Perl installieren	1106
E.3.3	MySQL installieren	1107
E.4	Xitami-Webserver	1107
E.4.1	Den Server installieren	1107
E.4.2	PHP installieren	1108
E.4.3	Perl installieren	1109
E.4.4	MySQL installieren	1110
E.5	MySQL	1110

F	Weitere Informationen	1113
F.1	WWW-Links	1113
F.2	Buch-Tipps	1113
F.3	Quellenverzeichnis	1114
G	Inhalt der CD-ROM	1115
H	Glossar	1117
I	Schlusswort	1123
	Index	1125

Einführung

*Die Softwarebranche ist binär, du bist die 1 oder 0, tot oder lebendig.
– Tim Robbins als Gary Winston im Film »StartUp«*

Im Jahre 1990 stellte Tim Berners-Lee die erste Verbindung von einem Browser zu einem Server her, um ein dort abgelegtes Dokument im HTML-Format zu öffnen. Seitdem ist das Internet unaufhörlich in alle möglichen Bereiche vorgestoßen, und es ist aus dem heutigen Leben kaum noch wegzudenken – seien es nun Privatleute, die eine kleine Seite ins Internet gestellt haben, um auch dabeisein zu können, oder ganze E-Business-Systeme, die den Eigentümern bares Geld beschern bzw. bescherten. Jedoch sind eine gewisse Flaute und der Einbruch der so genannten New Economy nicht von der Hand zu weisen. Nun beginnt die Aufbauarbeit, diesmal jedoch wesentlich überlegter und langsamer.

Ohne Dynamik kommt kaum eine Webseite aus, denn Internetbenutzer verlangen heutzutage einfach mehr als nur einfache Seiten mit alten Informationen. Die Informationen müssen aktuell, informativ und nach Möglichkeit schön verpackt sein. Bei kleinen Projekten ist das nicht weiter problematisch, sind HTML, CSS und JavaScript doch probate Mittel, dies auch zu realisieren. Ändern sich die Informationen jedoch täglich oder werden sie gar stündlich erweitert, ist es mühselig, dies alles noch in kleine handliche HTML-Pakete zu schnüren. Hier wird der Einsatz von serverseitigen Scriptsprachen wie Perl und PHP, die im Laufe der Jahre zu ernst zu nehmenden Programmiersprachen herangewachsen sind, äußerst interessant und hilfreich. Gerade PHP schickt sich in der vor kurzem veröffentlichten neuen Version 5 an, seinen Platz an der Spitze weiter auszubauen.

Dynamik

Ob Sie nun zur Gruppe der Privatleute gehören, die auch sagen wollen: »Ich habe meine eigene Internetseite«, oder Ihren Verein im Internet präsentieren wollen, um noch mehr Personen und somit potenzielle Mitglieder erreichen zu können, spielt dabei eine eher nebensächliche Rolle. Eventuell steht sogar Ihr Chef eines Morgens in Ihrem Büro – angestachelt durch die Werbung eines alteingesessenen Unternehmens mit drei großen blauen Buchstaben – und teilt Ihnen mit, dass Ihre Firma ebenfalls ins Internet muss, damit neue Vertriebswege erschlossen werden können. Für Sie stellt sich nun in jedem Fall die Frage: Aber wie? Diese Frage werde ich Ihnen in diesem Buch Schritt für Schritt beantworten.

Private oder geschäftliche Website?

An wen richtet sich dieses Buch?

In erster Linie richtet sich dieses Buch an Einsteiger, die bisher lediglich auf Seiten der User Erfahrungen mit dem Medium Internet gemacht haben und die nun den

Einsteiger

Umgang mit HTML, CSS, JavaScript oder den anderen Internettechnologien wie PHP, Perl und MySQL erlernen möchten.

Fort-
geschrittene

Aber auch erfahrene Leser, die bereits Kenntnisse in HTML oder Perl haben und nun weitere Mittel und Wege kennen lernen möchten, um Webprojekte mit serverseitigem Scripting zu erstellen, werden in diesem Buch nicht außen vor gelassen. Gerade die Zusammenarbeit all dieser Sprachen birgt des Öfteren Stolpersteine, die es zu überwinden gilt.

Was Sie schon können sollten

Programmier-
erfahrung ist
hilfreich

Grundsätzlich sind Erfahrungen mit der einen oder anderen Programmiersprache wie der C-Familie oder Java hilfreich. Vor allem syntaktische Elemente aus diesen Sprachen werden Sie in JavaScript, Perl und PHP antreffen. Außerdem erleichtern Kenntnisse in einer Programmiersprache auch das Verständnis des programmatischen Ablaufs. Dies ist jedoch nicht Voraussetzung, denn ich werde Ihnen zu jeder Sprache die syntaktischen Eigenheiten der Variablen, Schleifen und Programmsteuerungen eingehend erläutern.

Kenntnisse des
Betriebssystems

Generell sollten Sie sich jedoch schon einmal ausführlicher mit Windows oder Linux beschäftigt haben und sich mit der Verwendung der Betriebssystemumgebung und den spezifischen Eigenheiten auskennen. Sowohl zum Erstellen der Dateien und zum anschließenden Ablegen auf dem Webserver (auf Ihrem eigenen Rechner oder auf einem Server im Internet) als auch zur Installation und Konfiguration eines funktionierenden Webservers auf Ihrem Rechner ist dieses Wissen notwendig. Aber auch hierbei werde ich Ihnen Hilfestellung leisten.

Was Sie mit diesem Buch lernen

Ich werde Sie in diesem Buch in die Auszeichnungssprache HTML und die Formatierungssprache CSS einführen. Die clientseitige Sprache JavaScript wird Ihnen in Verbindung mit HTML an praxisorientierten Beispielen nähergebracht. Aber auch die serverseitigen Scriptsprachen Perl und PHP werden im Zusammenhang mit der Datenbankabfragesprache SQL an alltäglichen Beispielen erläutert.

Von Besucher-
zählern bis
Datenbanken

So lernen Sie das Programmieren von Gästebüchern, die Funktionsweise von Besucherzählern und werden am Ende auch eigene, dynamische Grafiken – egal ob als JPG, PNG oder gar SVG – erzeugen können. Dabei werden die Daten sowohl aus Dateien als auch aus Datenbanken geholt.

Neu in der 2. Auflage ist der zusätzliche Teil zu XML. Nachdem zu Zeiten der 1. Auflage der Einsatz von XML noch relativ selten erfolgte, wird die Bedeutung durch Webservices oder RSS/RDF immer größer, und dem soll selbstverständlich Rechnung getragen werden.

Schlussendlich werden Sie in den Entwicklungs- und Programmierprozess einer vollständig dynamischen Webseite eingeführt, indem Sie ein solches Projekt von A bis Z planen, entwickeln und programmieren – natürlich immer mit praxistauglichen Beispielen.

Schreibweisen in diesem Buch

Da Sie in diesem Buch sowohl auf Erklärungen, Quellcodebeispiele als auch auf programmiersprachenspezifische Schlüsselwörter stoßen werden, werden zur besseren Unterscheidung und auch der Übersicht halber verschiedene Formatierungen verwendet.

Da Quelltexte einen großen Teil in diesem Buch ausmachen, wurden sie auch mit den augenscheinlichsten Formatierungen versehen. Diese Quelltexte werden Listings genannt und heben sich vom normalen Text durch eine nichtproportionale Schrift ab.

```
// Dies ist ein Beispiel für einen Quelltext  
// bzw. ein Listing.
```

Quelltexte

Auch Variablen- und Funktionsbezeichner, Schlüsselwörter einer Programmiersprache und ähnliche Begriffe werden in der nichtproportionalen Schrift dargestellt.

Schlüsselwörter

Dateinamen und Verzeichnispfade sowie Internet- und E-Mail-Adressen werden **fett gedruckt** formatiert.

Neben diesen Formatierungen werden Sie auch auf verschiedene Symbole am Seitenrand treffen, die Textabschnitte mit wichtigen Inhalten, Warnungen oder Verweise auf die dem Buch beiliegende CD-ROM beinhalten.

Symbole

Textabschnitte mit diesem Symbol enthalten Tipps und Hinweise für den Umgang im Programmieralltag.



Wenn dieses Symbol am Seitenrand auftaucht, enthält der Textabschnitt eine Warnung.



Sicherheitsprobleme und Fallen der Programmiersprache werden mit diesem Symbol markiert.



Textabschnitte mit Verweisen auf die beiliegende CD-ROM sind mit dem nebenstehenden Symbol markiert.



Danksagungen

Kein Buch entsteht in völliger Eigenregie, und somit gibt es immer Personen, die direkt oder auch indirekt an einem solchen Werk mitgearbeitet haben. Diesen Personen möchte ich an dieser Stelle danken.

Meiner Familie und insbesondere meinem Vater danke ich für die Inspiration zu diesem Buch und dafür, dass sie mich während der Arbeit daran immer wieder aufs Neue motiviert haben.

Meinem guten Freund Kolja Engelmann danke ich für sein Wissen um Perl und Linux und die Bereitschaft, das Buch immer wieder Probe zu lesen, und dafür, dass er mit Tipps und Ratschlägen zur Verbesserung dieses Buchs beigetragen hat. Ich werde mich noch lange Zeit an sein »Willst du das wirklich so schreiben?« erinnern.

Meiner wunderbaren Freundin Conny danke ich für ihre Hilfe beim Finden der richtigen Ausdrucksweise und für ihre unermüdliche Kraft, mich zu motivieren.

Natürlich sei auch Stephan Mattescheck und seinem gesamten Team von Galileo Computing gedankt, sowohl für das Interesse an diesem Projekt als auch für die vielen Hilfestellungen.

Internetangebot

Auf den Webseiten von Galileo Press und natürlich auf meiner Webseite finden Sie zusätzliche Informationen rund um dieses Buch. Dort können Sie auch mit mir oder dem Verlag in Kontakt treten. Ich würde mich über Anregungen und konstruktive Kritik freuen und stehe Ihnen im Forum von MyGalileo auch für Fragen zur Verfügung, sobald Sie dieses Buch dort registriert haben. Alternativ können Sie auch gern das Forum auf meiner Webseite verwenden.

Webseiten Galileo Press: <http://www.galileo-press.de>

Galileo Computing: <http://www.galileocomputing.de>

Meine Seite: <http://www.mark-lubkowitz.de>

Forum: <http://www.mark-lubkowitz.de/forum>

Vorwort zur 2. Auflage

Vor gut anderthalb Jahren – im Mai 2003 – ist die 1. Auflage dieses Buches erschienen. Seit diesem Zeitpunkt ist das Buch mehrere tausend Male verkauft worden, und wenn ich ehrlich bin, habe ich mit diesem Erfolg nicht gerechnet. Im Gegenteil, als blutiger Neuling im Schreiben von Büchern habe ich meine Erwartungen gering gehalten. Desto erfreuter und überraschter war ich über die große Zahl an Zuschriften

und die durchweg positive Kritik. Es gab zwar auch die eine oder andere negative Anmerkung, jedoch war auch diese in der Regel angebracht und nachvollziehbar. Ganz besonders bedanken möchte ich mich an dieser Stelle bei Tanja Credner. Die vielen E-Mails haben mir die Überarbeitung dieses Buches wesentlich erleichtert. Nachdem ich mit einigen Lesern auch einen regen elektronischen Schriftwechsel führen konnte, hat mich am meisten die unterschiedliche Mischung der Leser überrascht. Im Altersbereich zwischen 13 und 60 Jahren war jede Gruppe vertreten, ob Studenten oder Hobbyisten. Euch allen meinen Dank.

Seit der 1. Auflage hat sich in Hinblick auf die Internettechnologien einiges getan. Die finale Version 5 von PHP und die Version 4 von MySQL sind erschienen, in vielen Fachzeitschriften wird verstärkt auf Webservices eingegangen, und langsam, aber sicher scheint sich auch in Deutschland das Bloggen zu etablieren. Darüber hinaus finden sich auf vielen Seiten immer häufiger kleine orange Icons mit der Aufschrift RSS oder RDF. PHP 5 und MySQL 4 wurden in dieser 2. Auflage ebenso berücksichtigt wie die steigende Bedeutung von Webservices, und was es mit Bloggen und RSS/RDF auf sich hat, werden Sie in diesem Buch ebenfalls erfahren.

Aufgrund der Erweiterungen wurde das Buch einigen Änderungen unterzogen, die gerade Ihnen, dem Leser, zugute kommen. Das Buch selbst hat ein breiteres Format. Bei einer hohen Seitenzahl erleichtert es das Lesen, da Sie das Buch problemlos aufgeklappt auf den Tisch legen können. Andererseits ist das Format auch in der Lage, mehr Informationen zu fassen. Ich hoffe, dass Ihnen dieses Format genauso gut gefällt wie mir.

Ach ja, Käufer eines Exemplars der 1. Auflage oder eines der Nachdrucke können sich unter MyGalileo den neu hinzugekommenen XML-Teil als PDF herunterladen.

So, genug geredet, nun heißt es:

<start>

Ihr Mark Lubkowitz

4 Gute Vorbereitung ist wichtig

Um einen Zufall herbeizuführen, bedarf es viele Vorbereitung. Um eine gute improvisierte Rede zu halten, braucht man mindestens drei Wochen.
– Mark Twain

Bevor Sie alle Beispiele aus diesem Buch auch in vollem Umfang testen können, sollten Sie sich auf Ihrem Rechner einen Webserver mit allen erforderlichen Erweiterungen wie PHP, Perl und MySQL einrichten. Da dies nicht ohne weiteres möglich ist, folgen in diesem Kapitel ein paar Schritt-für-Schritt-Anleitungen sowohl für Windows als auch für Linux. Weitere Anleitungen finden Sie im Anhang in Anhang D.

4.1 WAMP

WAMP ist die Abkürzung für eine gebräuchliche Installationsvariante, in der als Webserver der Apache HTTPD-Server (daher das A) zum Einsatz kommt. Als Datenbanksystem wird MySQL (dafür steht das M) verwendet. Das P steht für Perl und/oder PHP. Das W ergibt sich aus dem Betriebssystem, auf dem diese Installation läuft: Windows.

4.1.1 Apache installieren

Zuallererst muss der Apache HTTPD-Server installiert werden. Die aktuellste Version des Servers können Sie unter der Adresse <http://nagoya.apache.org/dist/httpd/binaries/win32/> herunterladen. Ansonsten finden Sie sowohl die Version 1.3.31 (die letzte stabile Version der 1er-Reihe) als auch die Version 2.0.50 des Servers auf der beiliegenden CD-ROM im Verzeichnis `x:\server\apache\win32`. Die nachfolgende Installationsanleitung bezieht sich auf die Version 2.0.x des Servers.



Öffnen Sie nun die Datei `x:\server\apache\win32\ apache_2.0.50-win32-x86-no_ssl.msi` entweder durch einen Doppelklick auf die Datei oder mittels **Start · Ausführen....** Sollte Ihnen der Microsoft Installer nun melden, dass Sie eine neuere Version des MSI benötigen, finden Sie diese ebenfalls auf der CD-ROM im Verzeichnis `x:\misc\win32`. Für Windows NT/2000/XP ist es `inmsint.exe` und für Windows 95/98/Me `inmsi9x.exe`.

Microsoft
Installer

Apache heißt Sie im ersten Schritt zur Installation willkommen. Klicken Sie auf **Next >**, und akzeptieren Sie im nächsten Schritt die Lizenzvereinbarung, indem Sie »I accept the terms in the license agreement« markieren. Nach zwei weiteren Klicks auf **Next >** möchte das Setup-Programm einige Informationen zur Installation von Ihnen haben.



Abbildung 4.1 Eingeben der Server-Informationen

Server-Information

Im Eingabefeld »Network Domain« erwartet das Setup-Programm die Angabe einer Domäne. Wie Sie die Domäne genau nennen, bleibt Ihnen überlassen. »Server Name« erwartet den Namen des Servers in der angegebenen Domäne und »Administrator's Email Address« die E-Mail-Adresse des Administrators der Domäne. Auch diese beiden Angaben können Sie beliebig auswählen. Die Tabelle 4.1 enthält ein paar Vorschläge.

Network Domain	Server Name	Email Address
somenet.com	http://www.somenet.com	admin@somenet.com
mynetwork.local	http://www.mynetwork.local	admin@mynetwork.local
ihrname.local	http://www.ihrname.local	admin@ihrname.local

Tabelle 4.1 Beispiel-Konfiguration des Servers

Welche Daten Sie eingeben, ist für das Funktionieren des Servers unerheblich. Diese Daten werden jedoch ausgegeben, falls Fehler oder Ähnliches auftreten.

Zusätzlich haben Sie noch die Wahl zwischen »for All Users« und »only for the Current User«. Die erste Option bedeutet, dass jeder, der sich auf dem Rechner anmeldet, den installierten Server nutzen kann, aber auch alle Nutzer, die von außerhalb auf den Rechner zugreifen. Die zweite Option bedeutet, dass nur der Benutzer, der gerade angemeldet ist, den Server nutzen kann und dass alle anderen Benutzer, ob von außen oder vom gleichen Rechner aus, keinen Zugriff auf den Server erhalten.

Nach einem erneuten Klick auf **Next >** fordert das Setup-Programm Sie auf, den Installationstyp auszuwählen. Markieren Sie »Typical«, und klicken Sie auf **Next >**.

Auf dem nächsten Bildschirm bietet sich Ihnen die Möglichkeit, das Zielverzeichnis für die Installation auszuwählen. Wenn Sie auf die Schaltfläche **Change** klicken, können Sie dieses Verzeichnis ändern. Bestätigen Sie das Verzeichnis, indem Sie auf **Next >** und im nächsten Schritt auf **Install** klicken. Die Installation wird nun gestartet. Sie dauert in der Regel weniger als zwei Minuten.

Zielverzeichnis

Wenn Sie auf Ihrem Rechner eine Firewall installiert haben, kann es sein, dass diese sich nun meldet. Erlauben Sie Apache Verbindungen über den Port 80. Andernfalls wird der Server keine Anfragen bearbeiten können, da die Firewall den Zugriff sperrt. Mit einem abschließenden Klick auf **Finish** wird die Installation abgeschlossen.

Abschluss

Wenn die Installation erfolgreich durchgeführt worden ist, finden Sie in der Systray neben der Uhr nun ein neues Icon. Mit einem Doppelklick auf dieses Icon öffnet sich der Apache Service Monitor.

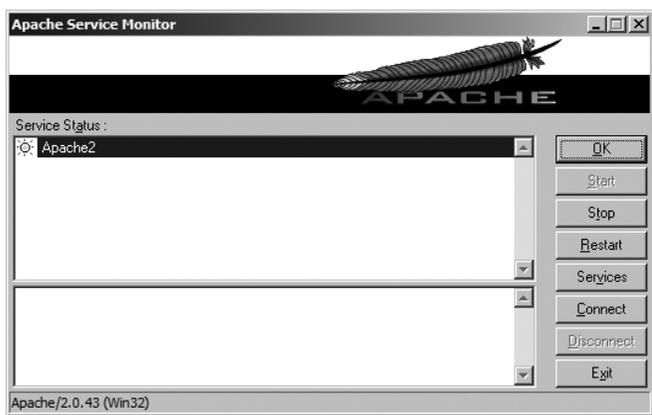


Abbildung 4.2 Apache Service Monitor

Ein grüner Kreis vor dem Eintrag »Apache2« bedeutet, dass der Server funktioniert und aktiv ist. Im Gegensatz dazu bedeutet ein roter Kreis, dass der Server nicht funktioniert. Sollte der Apache Service Monitor nach einem Neustart des Systems immer noch einen roten Kreis anzeigen, sollten Sie die Konfiguration der Firewall überprüfen oder die Installation ein weiteres Mal durchführen.

Sollte auf Ihrem Rechner ein Windows der 9x-Reihe installiert sein, also Windows 95 / 98 / 98SE oder ME, dann wird Ihnen der Apache-Service-Monitor neben dem Eintrag »Apache2« eine rote Lampe zeigen. Dies liegt daran, dass Windows 9x keine so genannten Dienste (engl. services) kennt. Somit kann Apache2 auch nicht als Dienst gestartet und betrieben werden. Rufen Sie aus dem Startmenü aus dem Apache-Ordner den Eintrag **Start Apache** aus. Es erscheint ein schwarzes Fenster. Solange dieses Fenster geöffnet ist, läuft der Server.



Erster Test Starten Sie nun Ihren Webbrowser, und geben Sie als Adresse entweder `http://localhost/` oder `http://127.0.0.1/` ein. Der Browser sollte nun eine Webseite mit der Überschrift »Es klappt! Der Apache-Webserver ist auf dieser Website installiert!« anzeigen.

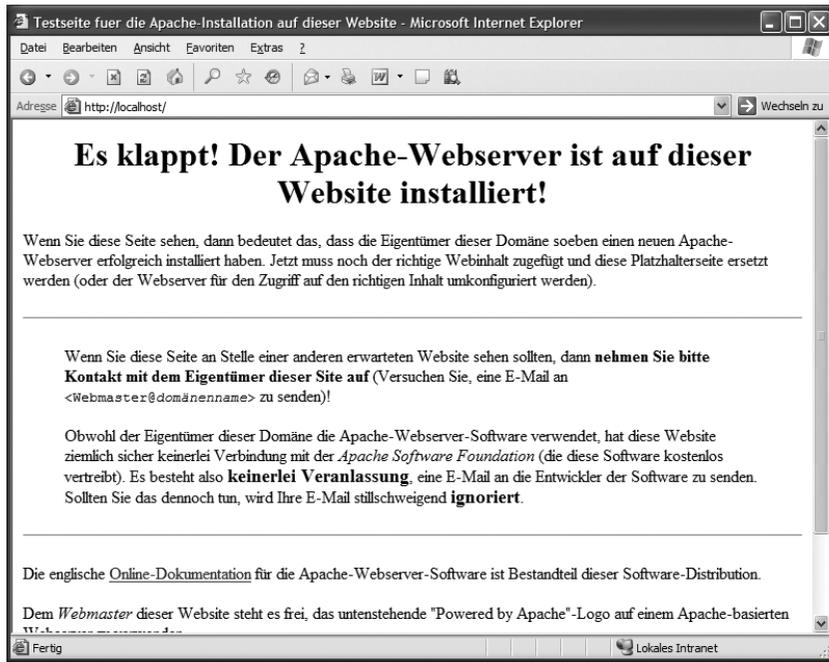


Abbildung 4.3 Der Apache-Webserver wurde erfolgreich installiert.

4.1.2 MySQL installieren



Der erste Schritt ist nun also getan. Jetzt folgt die Installation von MySQL. Auch MySQL ist in verschiedenen Versionen auf der CD-ROM vorhanden, und zwar im Verzeichnis `x:\server\mysql\win32`. Eine neuere Version können Sie unter der Adresse `http://www.mysql.de/` herunterladen. Das ZIP-Archiv müssen Sie nun entpacken. Die nachfolgende Installationsanleitung bezieht sich auf die Version 3.23, lässt sich aber auf alle 3.x- und 4.x-Versionen anwenden.

Nach dem Entpacken müssen Sie die Datei `setup.exe` ausführen. Klicken Sie nun zweimal auf **Next >**. Wenn Sie nicht möchten, dass MySQL im Verzeichnis `C:\mysql` installiert wird, können Sie dies mit einem Klick auf **Browse...** ändern. Klicken Sie anschließend wieder auf **Next >**, und wählen Sie als Installationstyp »Typical« aus. Starten Sie dann die Installation mit einem weiteren Klick auf **Next >**. Mit einem Klick auf **Finish** können Sie das Setup-Programm beenden.

Nun muss zum einen getestet werden, ob die Installation erfolgreich war, und zum anderen müssen noch ein paar Einstellungen vorgenommen werden. Wenn Sie MySQL im Verzeichnis `c:\mysql` installiert haben, klicken Sie nun auf **Start · Ausführen...**, und geben Sie `c:\mysql\bin\winmysqladmin.exe` und  ein. Haben Sie MySQL in einem anderen Verzeichnis installiert, dann ersetzen Sie `c:\mysql` durch den Installationspfad. Beim ersten Start von WinMySQLAdmin erscheint eine Maske zum Eingeben eines Benutzernamens und eines Passwortes.



Abbildung 4.4 WinMySQLAdmin erwartet beim ersten Start einen Benutzernamen und ein Passwort.

Tragen Sie im Feld »User name« einen beliebigen Benutzernamen und im Feld »Password« ein beliebiges Passwort ein. Dies sind dann gleichzeitig die Benutzerdaten für den späteren Zugriff auf die Datenbanken. Bestätigen Sie die Eingaben durch einen Klick auf **OK**. In der Systray neben der Uhr taucht nun ein neues Symbol auf, das eine Ampel darstellt. Klicken Sie auf dieses Symbol, und wählen Sie im aufklappenden Kontextmenü **Show me** aus.

Die Oberfläche von WinMySQLAdmin wird nun angezeigt. Klicken Sie auf die Registerkarte **my.ini Setup**.

Klicken Sie als Erstes auf die Schaltfläche **Pick-up and Edit my.ini values**. Im Memo-feld auf der rechten Seite sehen Sie nun den Inhalt der **my.ini**. Ersetzen Sie die Zeile

Sprache
auswählen

```
#language=c:/mysql/share/your language directory
```

durch die Zeile

```
language=c:/mysql/share/german
```

und zwar angepasst an das Installationsverzeichnis, das Sie gewählt haben.

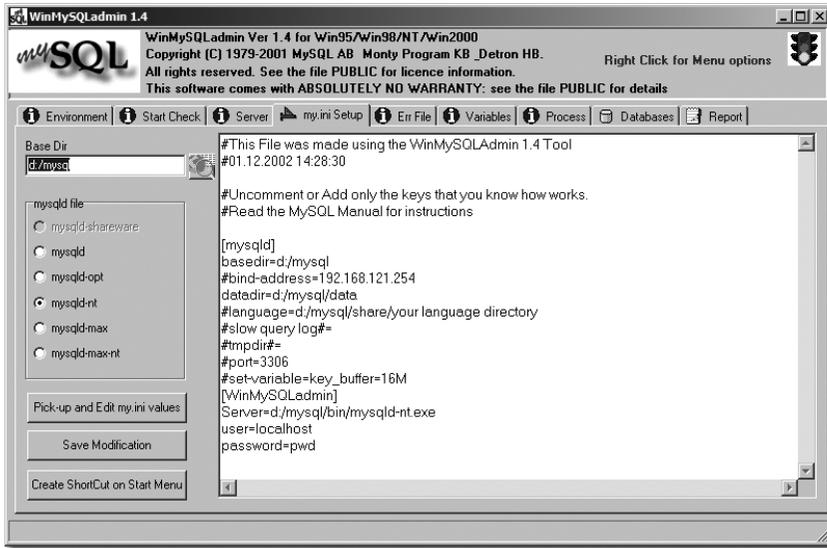


Abbildung 4.5 Konfiguration von MySQL

Port festlegen Entfernen Sie außerdem das Raute-Zeichen in der Zeile:

```
#port=3306
```

Klicken Sie nun auf die Schaltfläche **Save Modification**. Jetzt sollten Sie erst einmal einen Neustart des Rechners durchführen. Dies ist zwar nicht zwingend erforderlich, es ist aber die beste Überprüfung, ob alles korrekt funktioniert.

Rufen Sie nach dem Neustart wieder WinMySQLAdmin auf. Wenn das Symbol in der Systray neben der Uhr eine grüne Ampel anzeigt, wurde MySQL korrekt installiert und ist nun bereit für Verbindungen. Andernfalls sollten Sie zuerst die Firewall überprüfen (falls vorhanden) und gegebenenfalls Zugriffe auf den Port 3306 zulassen.

4.1.3 PHP installieren



Die Installation von PHP ist in wenigen Schritten erledigt. Die Version 4.3.8 von PHP finden Sie im Verzeichnis `x:\server\php\win32\ php-4.3.8-Win32.zip` als komprimiertes ZIP-Archiv. Eine aktuelle Version von PHP können Sie unter der Adresse <http://www.php.net/> herunterladen. Sie sollten dann aber auf jeden Fall das ca. 5 MB große ZIP-Archiv wählen. Als Ziel des Entpackvorgangs sollten Sie das Verzeichnis PHP und am besten das Laufwerk wählen, auf dem Sie auch schon Apache oder MySQL installiert haben, also z.B. `c:\php`.

Das ebenfalls in dem Verzeichnis liegende Archiv `php-5.0.0-Win32.zip` ist übrigens die nagelneue Version 5 von PHP für Windows. Gehen Sie bei dessen Installation genauso vor, wie hier für Version 4.3.8 beschrieben, beachten Sie aber unbedingt die folgenden Hinweise:

1. Seit der PHP-Version 5 ist die Bibliothek `php_pdf.dll` nicht mehr verfügbar. Sie dürfen das Semikolon in der entsprechenden Zeile also nicht entfernen, da Ihnen PHP sonst die Meldung ausgibt, dass die Bibliothek – weil sie fehlt – nicht geladen werden konnte.
2. In der `httpd.conf` müssen Sie anstatt `php.exe` die Interpreter-Version `php-cgi.exe` angeben. Andernfalls quittiert Ihnen der Apache 2 Webserver jeglichen Versuch, ein PHP-Skript auszuführen, mit einem *500 Internal Server Error*.

Öffnen Sie nun das Verzeichnis, in das Sie das ZIP-Archiv entpackt haben. Dort finden Sie eine Datei mit dem Namen `php.ini-recommended`. Kopieren Sie diese Datei in Ihr Windows-Verzeichnis. Dies ist entweder `C:\WINDOWS` oder `C:\WINNT`. Benennen Sie die Datei in `php.ini` um, und öffnen Sie diese mit einem Doppelklick. Es sollte nun der Windows-Editor gestartet und der Inhalt der `php.ini` angezeigt werden. Suchen Sie nun zuerst die Zeile

php.ini
konfigurieren

```
error_reporting = E_ALL
```

und ändern Sie diese in

```
error_reporting = E_ALL & ~E_NOTICE
```

um. Anschließend ändern Sie die Zeile

```
display_errors = Off
```

in:

```
display_errors = On
```

Jetzt suchen Sie die Zeile

```
extension_dir = ./
```

ersetzen diese durch das PHP-Installationsverzeichnis und hängen das Verzeichnis **extensions** an. Das Verzeichnis muss in doppelten Anführungsstrichen notiert werden, und anstelle des Backslashes `\` müssen Sie einen normalen Schrägstrich `/` verwenden.

```
extension_dir = "c:/php/extensions"
```

Neben diesen Änderungen sollten Sie außerdem noch die beiden Zeilen

```
;extension=php_gd2.dll  
;extension=php_pdf.dll
```

durch

```
extension=php_gd2.dll  
extension=php_pdf.dll
```

ersetzen. Achten Sie darauf, das Semikolon wegzulassen.

Apache konfigurieren

Der nächste Schritt ist die Konfiguration vom Apache Webserver in Bezug auf PHP. Klicken Sie auf **Start • Programme • Apache HTTP Server 2.0.x • Configure Apache Server • Edit the Apache httpd.conf Configuration File**. Sollten Sie nun gefragt werden, mit welchem Programm die Datei geöffnet werden soll, dann wählen Sie am besten **notepad.exe** aus. Suchen Sie in der Datei nun die Stelle

```
<Directory />  
    Options FollowSymLinks  
    AllowOverride None  
</Directory>
```

Ersetzen Sie diese vier Zeilen durch die folgenden:

```
<Directory />  
    Options FollowSymLinks ExecCGI  
    AllowOverride None  
</Directory>
```

Suchen Sie nun die Zeile:

```
<Directory "C:/Programme/Apache Group/Apache2/htdocs">
```

Diese Zeile kann variieren, je nachdem, in welchem Verzeichnis Sie Apache installiert haben. Nach dieser Zeile sollte die Zeile

```
Options Indexes FollowSymLinks
```

folgen. Unter Umständen finden Sie zuvor einige Zeilen mit einem Raute-Zeichen am Zeilenanfang. Hängen Sie an die Zeile `ExecCGI` an. Die Zeile sollte nun folgendermaßen aussehen:

```
Options Indexes FollowSymLinks ExecCGI
```

Als Nächstes suchen Sie die Zeile

```
DirectoryIndex index.html index.html.var
```

und ersetzen diese durch:

```
DirectoryIndex index.html index.htm index.php index.php3
```

Zum Schluss müssen Sie noch drei neue Zeilen in die `httpd.conf` einfügen:

```
ScriptAlias /php/ "c:/php/"
AddType application/x-httpd-php .php .php3
Action application/x-httpd-php "/php/php.exe"
```

Der Pfad `c:/php/` sollte natürlich Ihrem Installationspfad von PHP angepasst werden und auf das Verzeichnis zeigen, in dem die Datei `php.exe` liegt. Speichern Sie die Änderungen ab, und starten Sie den Apache Server neu. Dies geht am einfachsten über den Apache Service Monitor, indem Sie auf **Restart** klicken.

Um nun zu testen, ob die Installation und die Konfiguration von PHP erfolgreich waren, sollten Sie die Datei `x:\misc\phpinfo.php` in das Dokumentenverzeichnis von Apache kopieren. Dieses finden Sie im Installationsverzeichnis von Apache. Wurde der Apache Server im Verzeichnis `C:\Programme\Apache Group` installiert, heißt das Dokumentenverzeichnis `C:\Programme\Apache Group\Apache2\htdocs`. Dort liegen bereits verschiedene Dateien. Starten Sie nun den Browser, und geben Sie als Adresse `http://localhost/phpinfo.php` ein. Sie sollten dann eine ähnliche Ausgabe erhalten, wie sie in Abbildung 4.6 zu sehen ist. Test

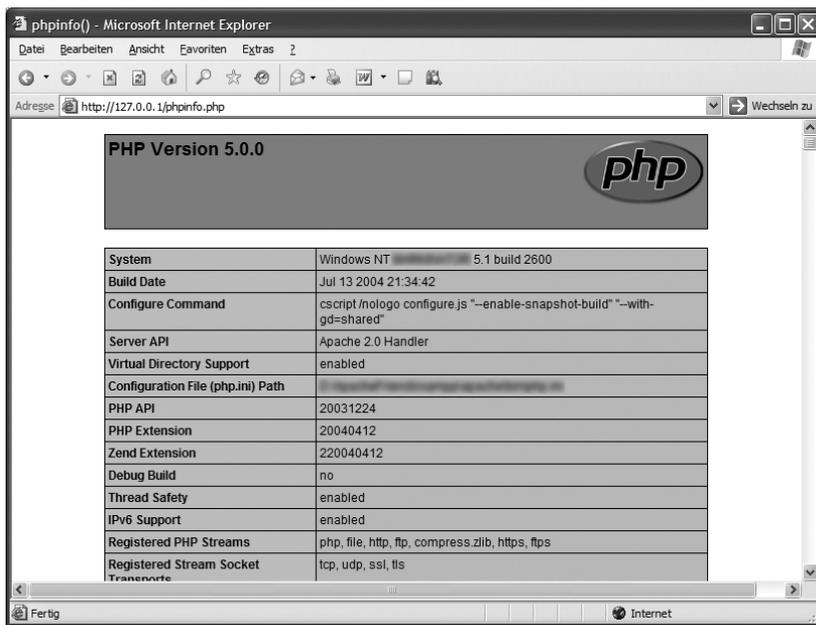


Abbildung 4.6 Testausgabe zum Überprüfen, ob PHP erfolgreich installiert wurde

Wenn Sie diese Ausgabe nicht erhalten, überprüfen Sie noch einmal die Änderungen an der `php.ini` und `httpd.conf`. Versuchen Sie außerdem einen weiteren Neustart des Webservers und notfalls des gesamten Computers.

4.1.4 Perl installieren



Die Version 5.8.0 von Perl finden Sie auf der CD-ROM im Verzeichnis `x:\server\perl\win32\ActivePerl-5.8.0.806-MSWin32-x86.msi`. Eine aktuellere Version erhalten Sie unter <http://www.activestate.com/>. Um die Installation zu starten, doppelklicken Sie auf die Datei oder klicken auf **Start** · **Ausführen...** und geben `x:\server\perl\win32\ActivePerl-5.8.0.806-MSWin32-x86.msi`  ein.

Klicken Sie auf **Next >**, und akzeptieren Sie die Lizenzvereinbarung, indem Sie die Option »I accept the terms in the License Agreement« markieren und ein weiteres Mal auf **Next >** klicken. Wenn Sie das Installationsverzeichnis von Perl ändern möchten, klicken Sie nun auf die Schaltfläche **Browse** und geben ein anderes Verzeichnis an. Bestätigen Sie das Verzeichnis mit einem Klick auf **Next >**.

PPM₃ Deaktivieren Sie die Option »Enable PPM3 to send profile info to ASPN«, da diese nicht weiter erforderlich ist. Weitere Informationen erhalten Sie unter http://www.activestate.com/Products/ASPN_Perl/. Klicken Sie erneut auf **Next >**, und markieren Sie die beiden Optionen »Add Perl to the PATH environment variable« und »Create Perl file extension association«. Bestätigen Sie auch dies mit einem Klick auf **Next >**.

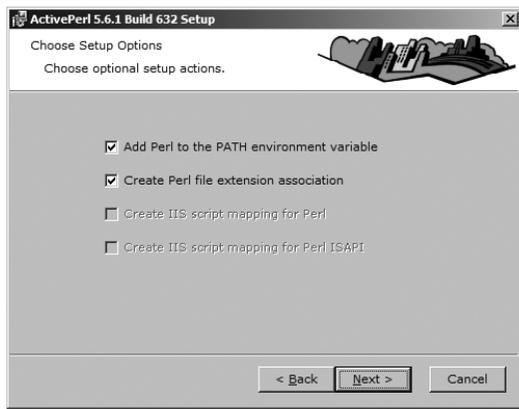


Abbildung 4.7 Perl in die PATH-Umgebungsvariable eintragen und eine Dateiverknüpfung erzeugen

Installationsabschluss

Um den Installationsvorgang zu starten, klicken Sie auf **Next >** und auf **Install**. Nachdem die Dateien kopiert und alle Konfigurationen vorgenommen wurden, können Sie die Installation mit **Finish** abschließen.

Testen

Um die Installation zu testen, öffnen Sie das Installationsverzeichnis von Apache und wechseln in das Verzeichnis `cgi-bin`, z. B. `C:\Programme\Apache Group\Apache2\cgi-bin`. In dem Verzeichnis finden Sie eine Datei mit dem Namen `printenv.pl`.

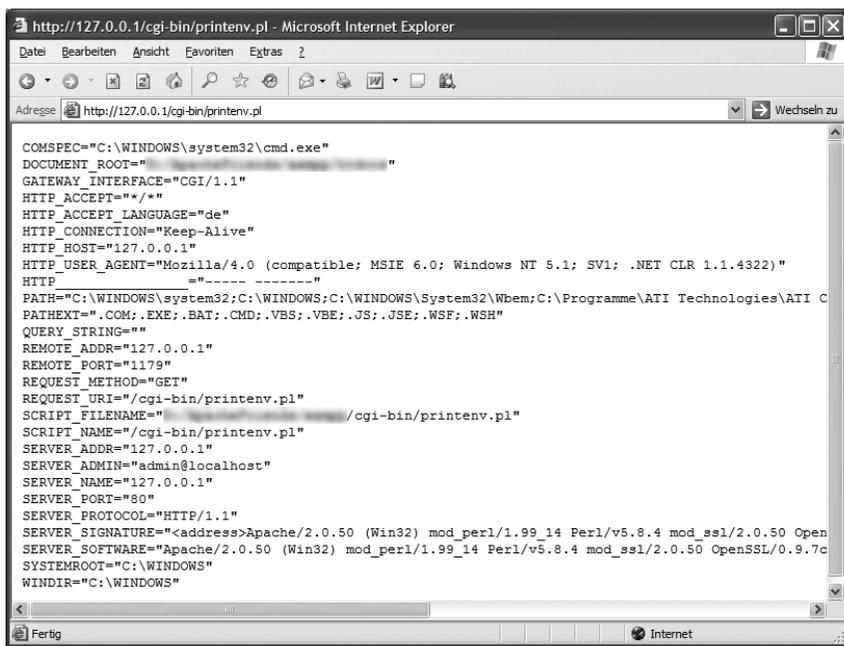
Öffnen Sie diese Datei, und ersetzen Sie die erste Zeile

```
#!C:/Perl/bin/Perl.exe
```

durch das Installationsverzeichnis von Perl. Haben Sie Perl im Verzeichnis C:\Perl installiert, müssen Sie keine Änderungen vornehmen. Wurde Perl aber z. B. auf Laufwerk D: installiert, muss die neue Zeile nun

```
#!D:/Perl/bin/Perl.exe
```

lauten. Speichern Sie die Änderungen, und öffnen Sie Ihren Browser. Geben Sie als Adresse <http://localhost/cgi-bin/printenv.pl> ein. Wenn Sie die gleiche oder eine ähnliche Ausgabe wie in Abbildung 4.8 erhalten, war die Installation erfolgreich. Sollte dies nicht der Fall sein, überprüfen Sie die erste Zeile der Datei und den Pfad zur perl.exe.



```

http://127.0.0.1/cgi-bin/printenv.pl - Microsoft Internet Explorer
Datei Bearbeiten Ansicht Favoriten Extras 2
Adresse http://127.0.0.1/cgi-bin/printenv.pl
COMSPEC="C:\WINDOWS\system32\cmd.exe"
DOCUMENT_ROOT=""
GATEWAY_INTERFACE="CGI/1.1"
HTTP_ACCEPT="*/*"
HTTP_ACCEPT_LANGUAGE="de"
HTTP_CONNECTION="Keep-Alive"
HTTP_HOST="127.0.0.1"
HTTP_USER_AGENT="Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)"
HTTP_=""
PATH="C:\WINDOWS\system32;C:\WINDOWS;C:\WINDOWS\System32\Wbem;C:\Programme\ATI Technologies\ATI C
PATHEXT=".COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH"
QUERY_STRING=""
REMOTE_ADDR="127.0.0.1"
REMOTE_PORT="1179"
REQUEST_METHOD="GET"
REQUEST_URI="/cgi-bin/printenv.pl"
SCRIPT_FILENAME=""
SCRIPT_NAME="/cgi-bin/printenv.pl"
SERVER_ADDR="127.0.0.1"
SERVER_ADMIN="admin@localhost"
SERVER_NAME="127.0.0.1"
SERVER_PORT="80"
SERVER_PROTOCOL="HTTP/1.1"
SERVER_SIGNATURE="<address>Apache/2.0.50 (Win32) mod_perl/1.99_14 Perl/v5.8.4 mod_ssl/2.0.50 Open
SERVER_SOFTWARE="Apache/2.0.50 (Win32) mod_perl/1.99_14 Perl/v5.8.4 mod_ssl/2.0.50 OpenSSL/0.9.7c
SYSTEMROOT="C:\WINDOWS"
WINDIR="C:\WINDOWS"
Fertig Internet

```

Abbildung 4.8 Browserausgabe des Scripts printenv.pl bei erfolgreicher Installation von Perl

4.1.5 PhpMyAdmin installieren

Die Version 2.5.7 von phpMyAdmin finden Sie auf der CD-ROM im Verzeichnis x:\server\phpmyadmin\phpMyAdmin-2.5.7-pl1.zip als komprimiertes ZIP-Archiv. Eine neuere Version erhalten Sie unter der Adresse <http://phpmyadmin.sourceforge.net/>.



Entpacken Sie das Archiv in das `htdocs`-Verzeichnis des Apache Webservers, z.B. `C:\Programme\Apache Group\Apache2\htdocs`. Wenn das Archiv entpackt worden ist, sollte sich in `htdocs` ein neues Verzeichnis mit dem Namen `phpMyAdmin-2.5.7` befinden.

Testen Zum Testen reicht es nun, den Browser zu starten und die Adresse `http://localhost/phpMyAdmin-2.5.7/` einzugeben. Die Ausgabe sollte dann der in Abbildung 4.9 entsprechen.

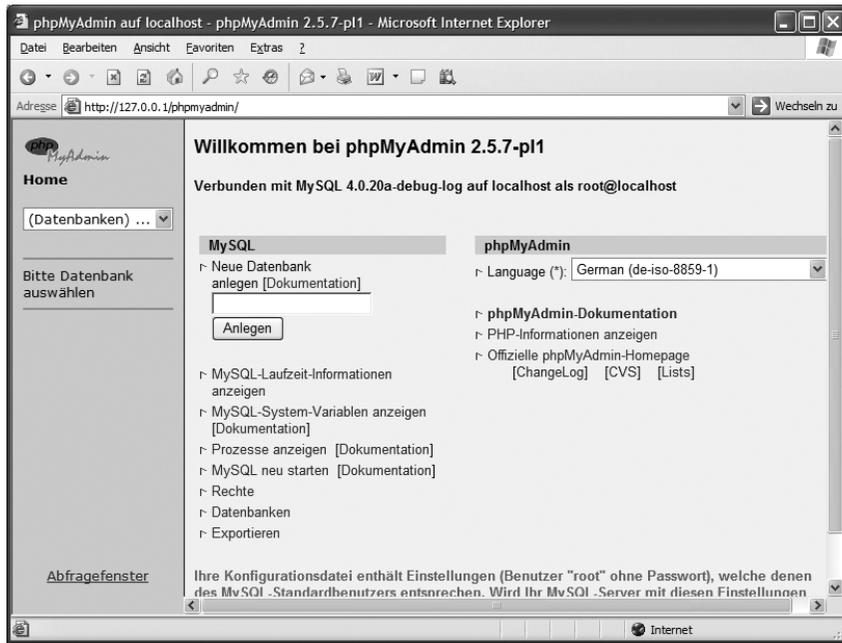


Abbildung 4.9 PhpMyAdmin nach der »Installation«

Dies ist auch eine gute Möglichkeit, um die Installation von MySQL zu testen, da phpMyAdmin sofort eine Verbindung aufbaut und auf der linken Seite die vorhandenen Datenbanken anzeigt.

4.2 LAMP unter SUSE 9.0

Die Installation eines LAMP-Systems (Linux-Apache-MySQL-PHP-Perl) ist je nach eingesetzter Linux-Distribution sehr unterschiedlich – vor allem, wenn es darum geht, die Pakete selbst zu kompilieren. Da SUSE eine der verbreitetsten Distributionen auf dem Markt ist, wird die Installation über den Package Manager an dieser Version dargestellt.

```
<body>Hallo WorldWideWeb!</body>
</html>
```

Listing 7.1 Hallo WorldWideWeb!

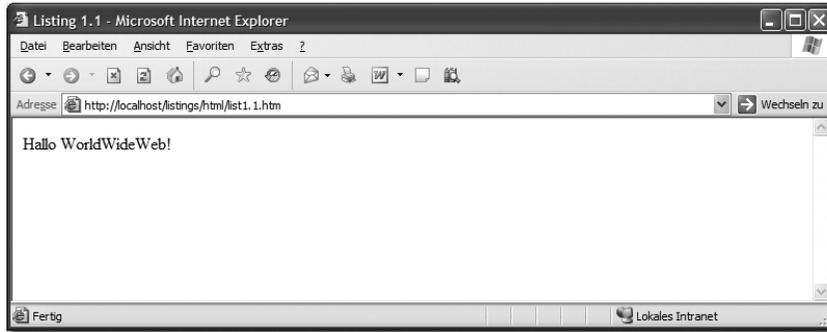


Abbildung 7.1 Darstellung des Listings 1.1 im Internet Explorer 6.0

Herzlichen Glückwunsch! Dies ist das erste von Ihnen erstellte HTML-Dokument.

7.4 Elemente und Tags

Wenn Sie sich das genauer ansehen, entdecken Sie gewisse Regelmäßigkeiten. Sie können Schlüsselnamen erkennen, die in spitzen Klammern geschrieben wurden. Diese Schlüsselnamen mit den spitzen Klammern nennt man **Tags** (dt. *Etiketten*), und sie unterteilen sich in Start- und Ende-Tags. Letztere kann man von den Start-Tags durch einen vorangestellten Schrägstrich / unterscheiden. Die Schlüsselnamen ergeben sich aus der Bezeichnung des Elements. Was jedoch sind Elemente? Ein HTML-Dokument besteht aus einer Vielzahl unterschiedlicher Elemente. Diese Elemente weisen gewisse Eigenschaften und Verhaltensformen auf und haben somit unterschiedliche spezifische Aufgaben, z.B. erzeugt das Element `h1` eine Überschrift der ersten Ordnung. Der zwischen den Tag-Paaren (Start- und Ende-Tag) stehende Text (auch Gültigkeitsbereich genannt) hebt sich deutlich vom restlichen Inhalt ab. In können Sie insgesamt vier dieser Elemente ausmachen: `html`, `head`, `title` und `body`. Diese werden Sie auch in jedem anderen HTML-Dokument wiederfinden, da sie die Grundausstattung eines HTML-Dokuments bilden. Ich nenne sie deshalb Basiselemente.

Kopf und Körper Sehen Sie sich die Zeilen `<title>Listing 1.1</title>` und `<body>Hallo WorldWideWeb!</body>` genauer an. Die jeweiligen Start- und Ende-Tags der Elemente `title` und `body` umrahmen unterschiedlichen Text. In Abbildung 7.1 können Sie diese Texte wiedererkennen: Der Text im Gültigkeitsbereich von `title` (also zwischen Start- und Ende-Tag) steht in der Titelzeile des Browserfensters, der Text im Gültigkeitsbereich von `body` im Anzeigebereich des Browsers.

Somit können Sie die Aufgaben dieser beiden Elemente schon festlegen. Das `title`-Element definiert den Dokument-Titel, und das `body`-Element enthält den darzustellenden Inhalt eines Dokuments. Das Pendant zum `body`-Element ist das `head`-Element. Es enthält alle wichtigen Dokumentinformationen (wie den Titel), die nicht im Anzeigebereich des Browsers dargestellt werden sollen. Im Gültigkeitsbereich eines Elements können auch weitere Elemente stehen. Dies ist insbesondere am `html`-Element zu erkennen. Alle Elemente eines Dokuments müssen in den Gültigkeitsbereich des `html`-Elements geschrieben werden.

Die typische syntaktische Notation für ein Element, das einen beliebigen Text enthält, sieht also folgendermaßen aus:

```
<Elementname>Text</Elementname>
```

Um in einem Element ein weiteres Element notieren zu können (auch als Verschachtelung bezeichnet), ersetzen Sie den Text einfach durch das weitere Element:

```
<Elementname1>
  <Elementname2>Text</Elementname2>
</Elementname1>
```

Eine Verschachtelung lässt sich beliebig lange fortsetzen, bis das hierarchisch letzte Element kein weiteres Element, sondern nur Text enthält.

7.5 Hierarchie

Sie kennen nun die vier Basiselemente. Diese und auch alle anderen Elemente unterliegen einer gewissen Ordnung. Sie legt fest, wie oft und in welcher Reihenfolge ein Element im HTML-Dokument vorkommen darf. Wenn Sie ein HTML-Dokument mit einem Haus vergleichen, dann stellt das `html`-Element das Haus dar. Die Elemente `head` und `body` bilden unterschiedliche Stockwerke, z.B. wäre `body` das Erdgeschoss und `head` der 1. Stock. Das `title`-Element kommt nur im `head`-Element vor und steht sinnbildlich für einen Raum im 1. Stock. Text, der im Gültigkeitsbereich des `title`-Elements notiert wird, ist dann gleichsam die Innenausstattung dieses Raums im 1. Stock des Hauses.

Wenn Sie nun diese Hierarchie (Ordnung) anhand des Hauses zurückverfolgen, stellen Sie fest, dass das `title`-Element dem `head`-Element untergeordnet ist, genauso wie die Elemente `head` und `body` dem `html`-Element untergeordnet sind. Man spricht auch von Eltern- und Kindelementen. Das `title`-Element ist das Kindelement des `head`-Elements, und anders herum ist `head` das Elternelement des `title`-Elements.

Die Felder mit den drei Punkten in Abbildung 7.2 stehen für weitere Kindelemente, die innerhalb von `head` oder `body` vorkommen dürfen. Versuchen Sie sich diese Hierarchie gut einzuprägen, da sie essenziell für das weitere Verständnis von HTML ist.

11 Geknüpftes Netz

Eine Gunst hört auf, eine Gunst zu sein, wenn Bedingungen an sie geknüpft sind.

– Thornton Wilder, US-amerikanischer Schriftsteller und Dramatiker

Neben der Menge an Informationen, die man bekommen kann – obgleich es eher als chaotische Informationsschwemme zu bezeichnen ist –, ist die Verknüpfung von Informationen das eigentlich Besondere am Internet. Von einer Seite im World Wide Web aus ist es möglich, eine weitere aufzurufen, die über thematisch ähnlichen Inhalt verfügt oder auch ganz anderen Inhalt bietet. Eine Verknüpfung zu einer anderen Seite nennt man **Hyperlink**. Sie werden in diesem Kapitel die verschiedenen Arten dieser Hyperlinks kennen lernen.

11.1 Aufbau einer Verknüpfung

Um in einem HTML-Dokument eine Verknüpfung zu einem anderen Dokument anzulegen, wird das `a`-Element verwendet. `a` steht in diesem Fall als Abkürzung für das englische Wort »anchor«, das ins Deutsche übersetzt »Anker« heißt. Wie Sie feststellen können, gibt es für Verknüpfungen die unterschiedlichsten Bezeichnungen; sowohl »Hyperlink« und die Kurzform »Link« als auch »Anker« oder »Verknüpfung« sind gebräuchlich.

Um nun einen Link auf ein anderes HTML-Dokument zu setzen, müssen Sie wissen, wo genau bzw. unter welcher Adresse das Dokument zu finden ist. Diesen Pfad übergeben Sie dem `a`-Element im Attribut `href` (engl. *hyper reference*, dt. *Hyperreferenz*). Den Titel für den Link notieren Sie im Gültigkeitsbereich des `a`-Elements. Dieser Titel ist dann für den Benutzer sicht- und anklickbar. Ein Link, der auf das HTML-Dokument `link.htm` verweisen und als Titel »Test« zugewiesen bekommen soll, würde folgendermaßen notiert werden:

```
<a href="link.htm">Test</a>
```

Das eben gezeigte Beispiel weist eine kleine Schwäche auf: Der Link würde nur funktionieren, wenn sich die Datei `link.htm` im gleichen Verzeichnis wie das Ausgangsdokument befinden würde. Läge es in einem Unterverzeichnis oder gar auf einem anderen Server, würde der Link nicht funktionieren. Umgangssprachlich wäre das ein »broken link« (gebrochene Verknüpfung).

11.2 Lokale Links

Von lokalen Links wird gesprochen, wenn auf ein Dokument (bzw. eine Datei) verwiesen wird, das auf dem gleichen Server wie das Ausgangsdokument liegt. Es gibt aber auch hier verschiedene Fälle, beispielsweise kann das Zieldokument im gleichen



Verzeichnis liegen, in einem untergeordneten Verzeichnis, in einem übergeordneten Verzeichnis oder in einem ganz anderen Verzeichniszweig.

11.2.1 Gleiches Verzeichnis

Liegt das Zieldokument im gleichen Verzeichnis, müssen Sie lediglich den Dateinamen des Dokuments (inklusive Dateierweiterung) im Attribut `href` notieren. Achten Sie aber auf Groß- und Kleinschreibung. Dies macht zwar in einer Windows-Umgebung keinen Unterschied, wohl aber in einer UNIX/Linux-Umgebung, und letztere ist die am häufigsten anzutreffende bei Webservern.

Untergeordnetes Verzeichnis

Würde das Zieldokument in einem untergeordneten Verzeichnis liegen, müssten Sie zusätzlich den Verzeichnisnamen zum Dateinamen des Zieldokuments hinzufügen. Gehen Sie einmal von folgendem Beispiel aus:

Das Quelldokument `quelle.htm` mit der Verknüpfung liegt im Verzeichnis `html`. Das Zieldokument `ziel.htm` liegt in einem Unterverzeichnis des `html`-Verzeichnisses namens `unterver`. Die Notation würde dann wie folgt aussehen:

```
<a href="unterver/ziel.htm">Zieldokument</a>
```



Beachten Sie bitte immer, dass Sie anstelle eines rückwärts geneigten Schrägstrichs (engl. *backslash*) »\« den normalen Schrägstrich (engl. *slash*) »/« verwenden. Dies hängt ebenfalls mit der Betriebssystem-Umgebung des Webserver zusammen. UNIX/Linux kennen den Backslash nicht als Trennzeichen für Verzeichnisse, da er Windows-typisch ist. Windows kommt aber auch mit dem normalen Schrägstrich als Verzeichnistrennzeichen klar.

Würde das Zieldokument in einem weiteren Unterverzeichnis des `unterver`-Verzeichnisses liegen, müssten Sie jedes weitere Unterverzeichnis ebenfalls mit in das Attribut `href` übernehmen.

```
<a href="unterver1/unterver2/ziel.htm">Zieldokument</a>
```

11.2.2 Übergeordnete Verzeichnisse

Gehen Sie vom vorherigen Beispiel aus. Sie möchten nun im Dokument `ziel.htm` einen Link zum ursprünglichen Dokument `quelle.htm` setzen. Die Angabe

```
<a href="html/quelle.htm">Zurück zum Quelldokument</a>
```

wäre jedoch falsch, da sie bedeuten würde, dass das Dokument `quelle.htm` im Unterverzeichnis `html` des Verzeichnisses `unterver` liegen würde. Es liegt aber in einem übergeordneten Verzeichnis, daher müssen Sie Folgendes notieren:

```
<a href="../quelle.htm">Zurück zum Quelldokument</a>
```



Die Zeichenfolge »..« steht für das übergeordnete Verzeichnis, d.h., Sie gehen in der Verzeichnisstruktur eine Ebene höher und sind wieder im Verzeichnis **html**. Für jede weitere Ebene, die Sie nach oben wechseln möchten, müssen Sie erneut »..« notieren – natürlich immer getrennt durch einen Schrägstrich. Drei Ebenen höher würden so notiert werden: ../.././quelle.htm.

11.2.3 Anderer Verzeichniszweig

Um einen Link auf ein Dokument in einem anderen Verzeichniszweig setzen zu können, müssen Sie die Schreibweisen für über- und untergeordnete Verzeichnisse kombinieren. Sie geben zuerst an, wie viele Ebenen Sie nach oben wechseln wollen, und geben dann die Unterverzeichnisse an.

```
<a href="../../html/unterver/ziel.htm">Zieldokument</a>
```

Würden Sie diesen Link im Dokument **ziel.htm** notieren, würden Sie wieder im gleichen Verzeichnis landen, da Sie zwei Ebenen nach oben gewechselt haben und dann wieder zwei Ebenen tiefer in die Verzeichnisse **html** und **unterver**.

11.3 Globale Links

Lokale Links funktionieren natürlich nur auf dem gleichen Server. Möchten Sie ein Dokument referenzieren, das auf einem anderen Server liegt, müssen Sie diesen explizit angeben.

```
<a href="http://www.kress.de/index.asp">Kress.De-Startseite</a>
```

Dieser Link würde auf die Startseite der Webseite des Mediendienstes Kress.de führen. Alternativ hätte man auch nur **http://www.kress.de/** im Attribut **href** notieren können. Der Webserver hätte in diesem Fall automatisch das Standard-Dokument in diesem Verzeichnis aufgerufen und an den Browser gesendet.

Genau wie bei lokalen Links können Sie auch bei globalen Links direkt in ein spezielles Unterverzeichnis wechseln, um ein bestimmtes Dokument zu referenzieren. An die Serveradresse werden dann einfach das Unterverzeichnis und der Dateiname angehängt.

```
<a href="http://www.galileo-press.de/foren/">Galileo-Press Foren</a>
```

Dieser Link würde das Standard-Dokument im Unterverzeichnis **foren** auf dem Galileo Press-Webserver aufrufen.

Unter-
verzeichnisse

11.4 E-Mail & Co.

Prinzipiell können Sie durch Verknüpfungen auf jeden möglichen Server, jedes Verzeichnis und jeden Dateityp verlinken – egal ob auf Ihrem Server oder einem anderen,

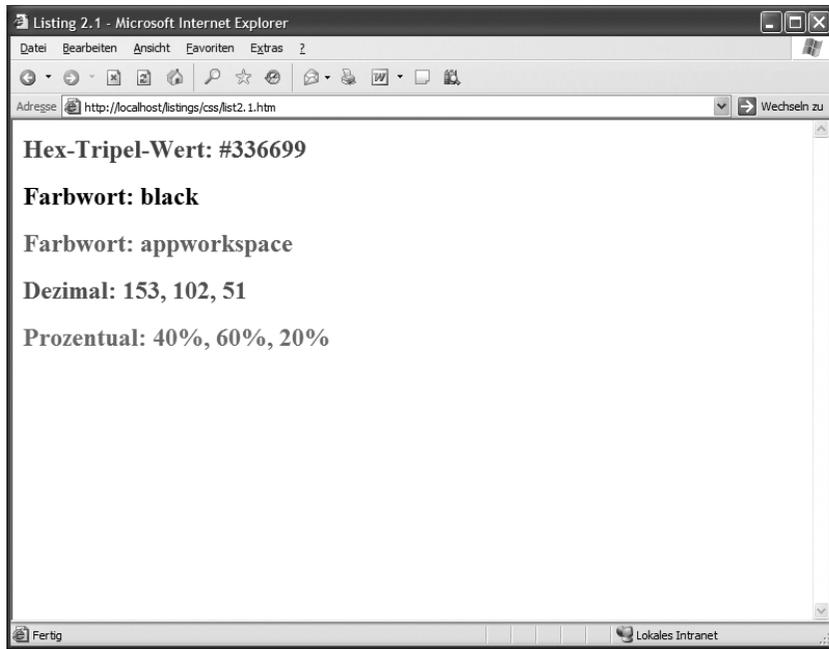


Abbildung 16.1 Darstellung des Listings 16.1 im Internet Explorer 6.0

16.2 Schriftformatierung

Schriftformatierung bedeutet, dass Sie die Schriftart, -farbe, -größe etc. eines Textes verändern können. Es würde also keinerlei Sinn machen, ein `img`-Element mit Eigenschaften zur Schrift zu formatieren, da die Aufgabe des `img`-Elements das Referenzieren einer Grafik und nicht das Darstellen von Text ist.

16.2.1 Schriftart bzw. -typ

Sie haben bereits einige Eigenschaften kennen gelernt, mit denen Sie das Aussehen eines Textes verändern können, unter anderem auch die Eigenschaft `font-family`. Mit dieser Eigenschaft können Sie die Schriftart bzw. den Schrifttyp eines Textes festlegen.

Angabe nach
Namen

Um explizit eine Schriftart zu wählen, müssen Sie den Namen der Schriftart notieren. Da bei vielen Schriftarten auch Leerzeichen enthalten sind, sollten Sie den Namen in einfache oder doppelte Anführungszeichen setzen, z.B. "Times New Roman" oder 'Times New Roman'.

```
font-family:Arial;  
font-family:'Times New Roman';  
font-family:"Times New Roman";
```

Achten Sie darauf, dass Sie nicht beide Arten von Anführungszeichen miteinander mischen, also zu Beginn des Namens der Schriftart ein doppeltes Anführungszeichen verwenden, zum Schluss aber ein einfaches.



Es kann jedoch immer einmal vorkommen, dass die gewünschte Schriftart auf dem Rechner des Benutzers nicht verfügbar ist. Die Schriftart hängt vom Betriebssystem, seiner Version und den installierten Programmen ab. Aus diesem Grund können Sie auch Schrifttypen angeben. Die bereits bekannten Schrifttypen aus HTML können Sie auch in CSS verwenden:

- ▶ `serif`
eine Schrift mit Serifen, z.B. Times New Roman
- ▶ `sans-serif`
eine Schrift ohne Serifen, z.B. Arial
- ▶ `cursive`
eine Schreibschrift, z.B. Comic Sans
- ▶ `fantasy`
eine Phantasieschrift
- ▶ `monospace`
eine Schrift mit nichtproportionalen Zeichen, z.B. Courier

```
font-family:serif;
font-family:sans-serif;
font-family:monospace;
```

Das Besondere ist, dass Sie immer Alternativen angeben können, falls es eine Schriftart oder einen Schrifttyp nicht gibt. Ein fehlender Schrifttyp ist aber sehr unwahrscheinlich. Die Alternativen werden dann in Reihenfolge ihrer Priorität, durch Kommata getrennt, notiert.

```
font-family:Arial,Geneva,Tahoma,sans-serif;
font-family:"Times New Roman",Garamond,serif;
```

Wenn Sie eine bestimmte Schriftart verwenden möchten, ist dies natürlich Ihre Entscheidung. Geben Sie als Alternative jedoch immer einen entsprechenden Schrifttyp an.



Zum Schluss folgt nun ein vollständiges Beispiel mit Abbildung:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<!-- Schriften /-->
<html>
  <head>
    <title>Listing 2.2</title>
  </head>
```

```

<body>
  <h2 style="font-family:Arial">Schriftart Arial</h2>
  <h2 style="font-family:Garamond">Schriftart
    Garamond</h2>
  <h2 style="font-family:cursive">Schrifttyp cursive
</h2>
  <h2 style="font-family:monospace">Schrifttyp
    monospace</h2>
</body>
</html>

```

Listing 16.2 Beispiele für Schriftarten in CSS

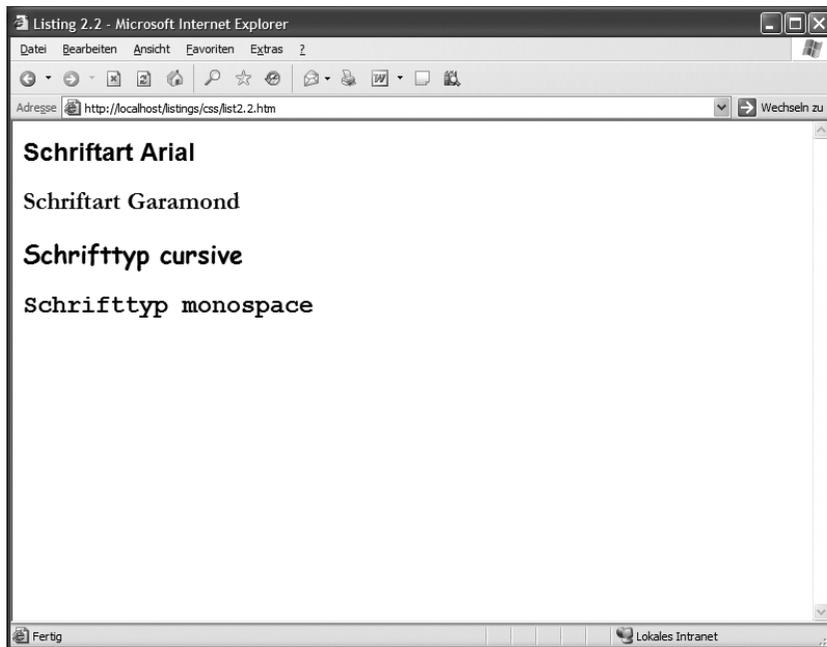


Abbildung 16.2 Darstellung des Listing 16.2 im Internet Explorer 6.0

Das Beispiel aus Listing 16.2 demonstriert die Verwendung von Schriftarten und -typen. In diesem Fall werden die Schriftarten Arial und Garamond und die Typen *cursive* und `monospace` verwendet.

16.2.2 Schriftgröße

Um die Schriftgröße nach Ihrem Ermessen zu verändern, müssen Sie die Eigenschaft `font-size` verwenden. Die Angaben sind in allen möglichen Maßen erlaubt.

```
font-size:20px;
font-size:16pt;
font-size:1cm;
```

Durch CSS ist noch eine weitere Möglichkeit gegeben. Dabei wird anstelle einer numerischen Angabe mit Maßeinheit einfach eine ungefähre Größe definiert.

Schlüsselwort	Beschreibung
xx-large	sehr, sehr groß
x-large	sehr groß
large	groß
medium	mittel
small	klein
x-small	sehr klein
xx-small	sehr, sehr klein
larger	größer als normal
smaller	kleiner als normal

Tabelle 16.2 Ungefähre Schriftgrößenangabe

```
font-size:xx-large;
font-size:smaller;
font-size:medium;
```

Diese ungefähren Größen tragen nicht umsonst diesen Namen. Sie sind abhängig vom Betriebssystem, dem Browser und den Einstellungen des Desktops.



16.2.3 Schriftneigung und -variante

Mit der Eigenschaft `font-style` können Sie die Schriftneigung und mit der Eigenschaft `font-variant` eine Schriftvariante definieren.

Mögliche Angaben für `font-style` sind:

- ▶ `italic` – kursiv
- ▶ `oblique` – kursiv
- ▶ `normal` – normal

Neigung

Die Angaben `italic` und `oblique` unterscheiden sich in der optischen Darstellung durch die Browser nicht.

Variante Mögliche Angaben für `font-variant` sind:

- ▶ `small-caps` – Kapitälchen
- ▶ `normal` – normal

Kapitälchen erzeugen bei einem normal geschriebenen Text mit Groß- und Kleinbuchstaben nur Großbuchstaben, wobei echte Großbuchstaben im Text größer dargestellt werden.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<!-- Schriftneigung und -variante //-->
<html>
  <head>
    <title>Listing 2.3</title>
  </head>
  <body>
    <h2 style="font-style:italic">Schriftstil italic</h2>
    <h2 style="font-style:oblique">Schriftstil oblique
  </h2>
    <h2 style="font-variant:small-caps">Schriftvariante
    small-caps</h2>
  </body>
</html>
```

Listing 16.3 Beispiele für Schriftstil und `-variante`

In Abbildung 16.3 können Sie sehr gut erkennen, dass zwischen `italic` und `oblique` kein Unterschied in der Darstellung zu sehen ist.

16.2.4 Schriftdicke

Während Sie in HTML darauf beschränkt sind, entweder einen Text normal oder fett darzustellen, bietet CSS mehrere Stufen, um die Schriftdicke zu verändern. So können Sie die Schrift sogar dünner als normal darstellen. Die entsprechende Eigenschaft lautet `font-weight`. Dabei können Sie sowohl eine numerische Angabe machen, die aus dem DTP¹-Bereich kommt, als auch ein Schlüsselwort verwenden.

Für numerische Angaben können Sie die Zahlen 100, 200, 300, 400, 500, 600, 700, 800 und 900 verwenden, wobei 100 extra dünn, 500 normal und 900 extra fett entspricht. Das einzige Problem ist, dass nicht jede Schriftart diese Angaben unterstützt.

Infolgedessen sollten Sie die Schlüsselwörter `bold`, `bolder`, `lighter` und `normal` verwenden. Während `bold` die Schrift einfach nur fett darstellt, erhöhen oder verringern die Angaben `bolder` und `lighter` die Schriftdicke. Um einen Normalzustand der Schrift zu erhalten, verwenden Sie schließlich `normal`.

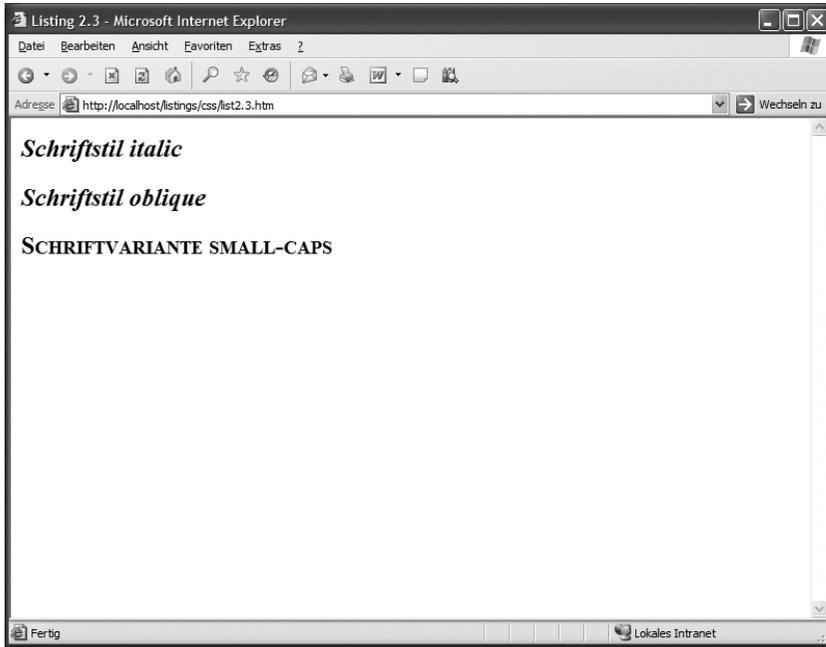


Abbildung 16.3 Darstellung des Listings 16.3 im Internet Explorer 6.0

```
font-weight:bold;
font-weight:lighter;
font-weight:bolder;
font-weight:100;
font-weight:500;
```

Zum Schluss folgt nun ein vollständiges Beispiel mit Abbildung:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<!-- Schriftneigung und -variante //-->
<html>
  <head>
    <title>Listing 2.4</title>
  </head>
  <body>
    <h2 style="font-weight:bold">Schriftdicke bold</h2>
    <h2 style="font-weight:lighter">Schriftdicke lighter</h2>
    <h2 style="font-weight:100">Schriftdicke 100</h2>
    <h2 style="font-weight:500">Schriftdicke 500</h2>
  </body>
</html>
```

Listing 16.4 Beispiele für font-weight



Auch hier fehlt noch die korrekte Unterstützung durch die Browser. In der Praxis können Sie momentan ausschließlich `normal` und `bold` bzw. `500` und `700` verwenden.

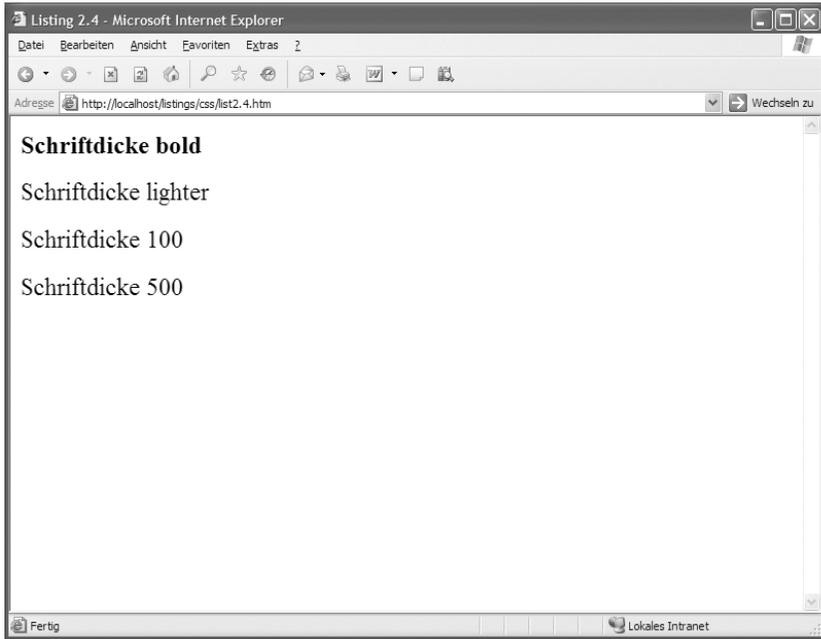


Abbildung 16.4 Darstellung des Listing 16.4 im Internet Explorer 6.0

16.2.5 Schriftfarbe

Mit der Eigenschaft `color` können Sie die Farbe der Schrift verändern. Dabei dürfen Sie sowohl Farbwerte (normale und benutzerspezifische) als auch alle drei Tripel-Werte verwenden.

```
color:black;  
color:appworkspace;  
color:#336699;  
color:rgb(140,140,0);  
color:rgb(50%,50%,100%);
```

Ein entsprechendes Beispiel finden Sie in Listing 16.1 und Abbildung 16.1.



Wegen der Lesbarkeit der Formatierungen sollten Sie nach Möglichkeit maximal zwei der fünf Möglichkeiten verwenden, um einer Schrift eine Farbe zuzuweisen.

16.2.6 Wort- und Zeichenabstände

Normalerweise verfügt ein HTML-Dokument über einen festgelegten Abstand zwischen den Zeichen und Wörtern. Der Wortabstand wird z.B. automatisch verändert, wenn Sie einen Text als Blocksatz ausrichten. Dies können Sie aber manuell über die Eigenschaften `letter-spacing` (Zeichenabstand) und `word-spacing` (Wortabstand) einstellen. Erlaubt sind numerische Angaben mit Ausnahme von prozentualen Angaben.

Beachten Sie, dass einige Browser noch Probleme mit der Interpretation haben, vor allem die Versionen von IE und Netscape vor der Version 6.



```
letter-spacing:0.5cm;
letter-spacing:3px;
word-spacing:1in;
word-spacing:20pt;
```

Zum Schluss folgt nun ein vollständiges Beispiel mit Abbildung:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<!-- Wort- und Zeichenabstände //-->
<html>
  <head>
    <title>Listing 2.5</title>
  </head>
  <body>
    <p style="letter-spacing:0.5cm; word-spacing:1in">
      Textabsatz mit Buchstabenabstand 0.5 cm und
      Wortabstand 1 Zoll.</p>
    <p style="letter-spacing:3px; word-spacing:20pt">
      Textabsatz mit Buchstabenabstand 3 Pixel
      und Wortabstand 20 Punkte.</p>
  </body>
</html>
```

Listing 16.5 Beispiel für `letter-spacing` und `word-spacing`

Der erste Textabsatz hat einen Buchstabenabstand von 0,5 Zentimetern und einen Wortabstand von 1 Zoll zugewiesen bekommen. Der Textabsatz ist dadurch sehr schwer zu lesen. Verwenden Sie also am besten geringere Abstände, damit der Benutzer den Text ohne größere Anstrengungen lesen kann.



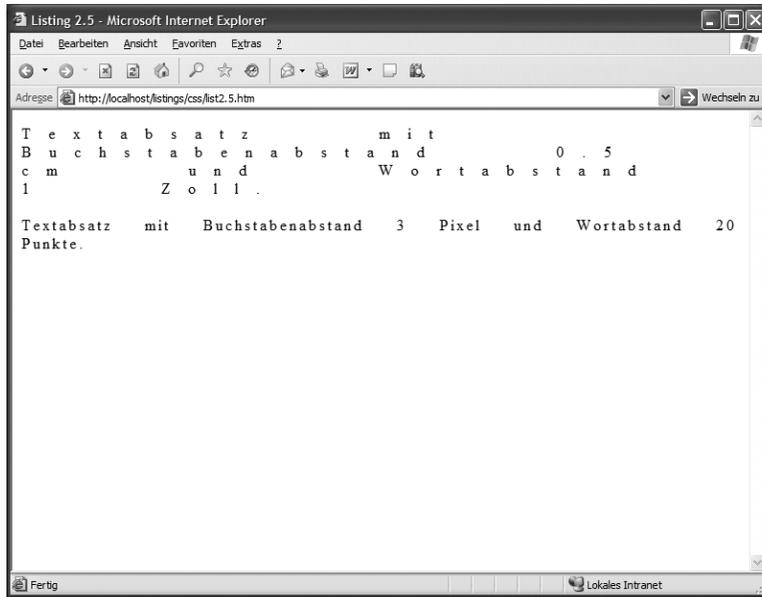


Abbildung 16.5 Darstellung des Listing 16.5 im Internet Explorer 6.0

16.2.7 Textdekoration

Es gibt verschiedene Arten, einen Text zu dekorieren. CSS versteht darunter hauptsächlich das Unter-, Über- oder Durchstreichen von Text. Verwenden Sie die Eigenschaft `text-decoration`, um einen Text mit einer der folgenden Formatierungen zu belegen.

- ▶ `underline`
Text wird unterstrichen.
- ▶ `overline`
Text wird überstrichen (Netscape interpretiert diese Angabe nicht).
- ▶ `line-through`
Text wird durchgestrichen.
- ▶ `blink`
Text blinkt (der IE interpretiert diese Angabe nicht).
- ▶ `none`
bedeutet keine Textdekoration.

Einige kurze Beispiele:

```
text-decoration:underline;
text-decoration:line-through;
text-decoration:none;
```

Zum Schluss folgt nun ein vollständiges Beispiel mit Abbildung:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<!-- Schriftdekoration //-->
<html>
  <head>
    <title>Listing 2.6</title>
  </head>
  <body>
    <h2 style="text-decoration:underline">Unterstrichen
  </h2>
    <h2 style="text-decoration:line-through">
      Durchgestrichen</h2>
    <h2 style="text-decoration:overline">
      &Uuml;berstrichen</h2>
    <h2 style="text-decoration:blink">Blinkend</h2>
  </body>
</html>
```

Listing 16.6 Beispiel für text-decoration

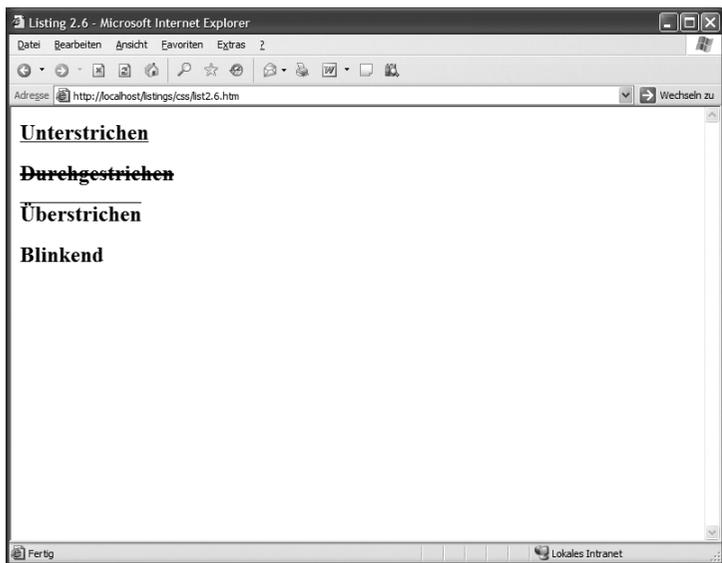


Abbildung 16.6 Darstellung des Listing 16.6 im Internet Explorer 6.0

Auch hier sollten Sie wieder mit Bedacht vorgehen. Ein unter-, über- oder durchgestrichener Text ist sehr schwer zu lesen, genauso wie ein blinkender Text. Denken Sie auch daran, dass unterstrichene Texte vom Benutzer schnell für einen Link gehalten werden können. Achten Sie also darauf, unterstrichene Texte und unterstrichene Hyperlinks z.B. farbig anders zu formatieren.



16.2.8 Texttransformation

Ähnlich wie mit der Eigenschaft `font-variant` können Sie mit `text-transform` die Groß- und Kleinschreibung eines Textes verändern, unabhängig davon, wie er im HTML-Dokument notiert wurde. Dadurch kann auch ein nur in Großbuchstaben geschriebener Text im Browser in Kleinbuchstaben dargestellt werden.

- ▶ `uppercase`
Text wird nur in Großbuchstaben dargestellt.
- ▶ `lowercase`
Text wird nur in Kleinbuchstaben dargestellt.
- ▶ `capitalize`
Wortanfänge werden groß dargestellt.
- ▶ `normal`
Keine Änderungen.

Die Angabe `capitalize` wird vom IE 4.x nicht interpretiert, wohl aber vom IE 6.

```
text-transform:uppercase;  
text-transform:lowercase;  
text-transform:normal;
```

Zum Schluss folgt nun ein vollständiges Beispiel mit Abbildung:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01  
Transitional//EN">  
<!-- Texttransformation //-->  
<html>  
  <head>  
    <title>Listing 2.7</title>  
  </head>  
  <body>  
    <h2 style="text-transform:uppercase">kleiner text  
    gross dargestellt</h2>  
    <h2 style="text-transform:lowercase">GROSSER TEXT  
    KLEIN DARGESTELLT</h2>  
    <h2 style="text-transform:capitalize">alles klein  
    geschrieben, trotzdem satzanfaenge gross</h2>  
  </body>  
</html>
```

Listing 16.7 Beispiele für `text-transform`



Ein sinnvoller Verwendungszweck für diese Formatierungsmöglichkeiten ist sehr schwer zu finden, da Sie den Text auch in der gewünschten Schreibweise im HTML-Dokument notieren können.

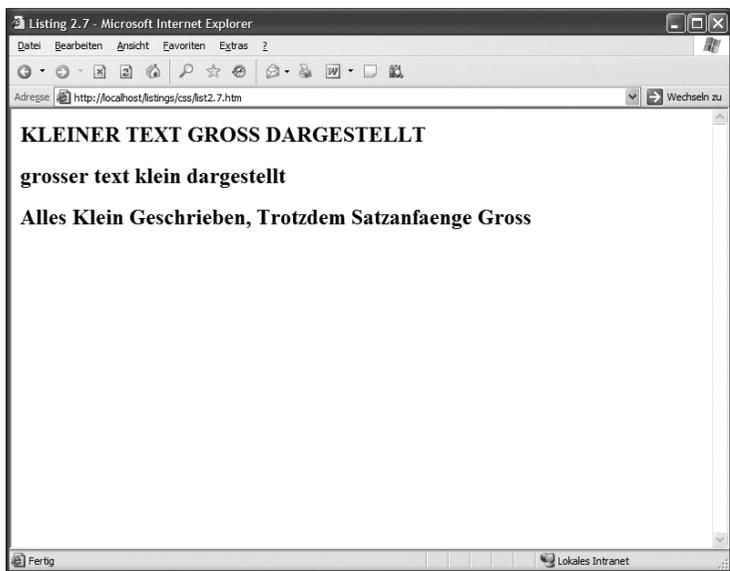


Abbildung 16.7 Darstellung des Listing 16.7 im Internet Explorer 6.0

16.2.9 Kurznotation zur Schriftformatierung

Es kann ziemlich mühselig werden, für jedes HTML-Element die gewünschten Eigenschaften zur Schriftformatierung einzeln anzugeben und zu verändern. Aus diesem Grund können Sie mit der Eigenschaft `font` die Eigenschaften `font-family`, `font-size`, `font-variant` und `font-weight` zusammenfassen. Dabei werden die einzelnen Angaben nicht durch Kommata getrennt, sondern durch Leerzeichen. Um einem Text also die Schrift Arial, die Größe 16 Pixel und kursiv zuzuweisen, wäre folgende Angabe möglich:

```
font:italic 14px Arial;
```

Weitere Beispiele:

```
font:bold 1cm Tahoma;
font:lighter 12pt monospace;
```

Zum Schluss folgt nun ein vollständiges Beispiel mit Abbildung:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<!-- Kurznotation //-->
<html>
  <head>
    <title>Listing 2.8</title>
  </head>
  <body>
```

```

<h2 style="font:bold 1cm Tahoma">Tahoma, 1cm, bold
</h2>
<h2 style="font:lighter 12pt monospace">Monospace,
  12 Punkte, lighter</h2>
</body>
</html>

```

Listing 16.8 Beispiele für font



Achten Sie auf die Reihenfolge. Notieren Sie zuerst die Schriftdicke und dann die Schriftvariante, gefolgt von der Schrifthöhe. Zum Schluss notieren Sie dann die Schriftart oder den Schrifttyp.

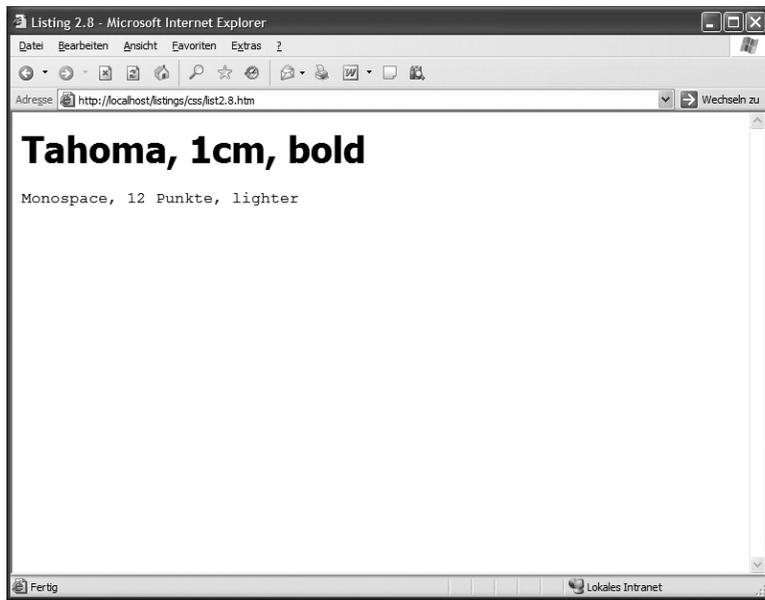


Abbildung 16.8 Darstellung des Listings 16.8 im Internet Explorer 6.0

16.3 Schriftartendateien

Ein sehr häufig anzutreffendes Problem ist folgendes: Ein Autor einer Webseite hat tagelang geschuftet, gebastelt und geschrieben und muss am Ende feststellen, dass die schöne Schriftart, die er verwendet hat, bei fast niemandem auf dem Rechner vorhanden ist. Das Ergebnis dieses »Missgriffs« ist eine vollkommen falsche Darstellung seiner Webseite im Browser. Natürlich könnte er nun anfangen, alle möglichen Texte in ein Bild umzuwandeln und diese dann in die HTML-Dokumente einzubinden. Die Download-Zeit eines solchen HTML-Dokuments würde jedoch sprunghaft in die Höhe schnellen, und die Besucher wären eher ungehalten, anstatt sich über eine so toll gewählte Schrift zu freuen und die langen Ladezeiten dankend in Kauf zu neh-

39 PHP

Aber für was ist das gut?

– Ingenieur der Advanced Computing Systems Division von IBM 1968 zum Microchip

PHP wurde ursprünglich im Herbst 1994 von Rasmus Lerdorf entwickelt und bestand anfänglich aus einer Sammlung von Perl-Skripten, die es Lerdorf ermöglichten, Zugriffe auf Webseiten zu erfassen und auszuwerten. Die Abkürzung PHP steht für »Personal Home Page Tools«.

Als Rasmus Lerdorf jedoch mehr Funktionalität benötigte, entwickelte er Mitte 1995 den ersten Parser und nannte das Paket PHP/FI Version 2. FI ist die Abkürzung für »Form Interpreter«. Bereits diese Version verfügte über große Ähnlichkeiten mit PHP, wie es heute verwendet wird. Es existierten bereits Variablen, Formularvariablen wurden automatisch interpretiert (FI = engl. *form interpreter*), und die Version beherrschte eine in HTML eingebettete Syntax. Zwar wies die Syntax große Affinitäten mit Perl auf, sie war jedoch sehr viel einfacher strukturiert. PHP/FI

Bis 1997 erfuhr PHP/FI immer wieder Verbesserungen und Änderungen und wurde mittlerweile auf ca. 50.000 Servern eingesetzt (ca. 1 % Marktanteil). Zu diesem Zeitpunkt war es allerdings noch immer ein Ein-Mann-Projekt.

Dies änderte sich 1997, als Zeev Suraski und Andi Gutmans den Parser vollständig neu programmierten, da ihnen PHP/FI zu schwach für die Entwicklung ihrer eigenen E-Commerce-Applikationen war. Sie einigten sich mit Lerdorf, und der neue Parser wurde als PHP 3.0 – nach 9 Monaten Testphase – Mitte 1998 freigegeben. Ende 1998 fand sich PHP 3.0 auf ca. 10 % der weltweiten Webserver. PHP 3.0

Im Mai 2000 wurde dann PHP 4.0 freigegeben, das auf der neuen Scripting-Engine Zend (gebildet aus den Vornamen Zeev und Andi) basierte. Viele Erweiterungen fanden Einzug in die neue Version, z.B. eine stark verbesserte Leistung, HTTP-Sessions und neue Sprachkonstrukte. Diese Verbesserungen wurden entsprechend honoriert, was dazu führte, dass auf ca. 20 % der Domains PHP installiert wurde und ca. 100.000 Entwickler mit PHP arbeiteten. Zend

Zum Zeitpunkt der Drucklegung dieses Buches liegt PHP in der finalen Version 4.3.8 und 5.0.2 vor. Mehr Informationen zu PHP 5 erhalten Sie in Kapitel 48, PHP 5 – *Was ist neu?* PHP 5.0

39.1 Das erste PHP-Script

PHP funktioniert ähnlich wie Perl, jedoch mit einem gravierenden Unterschied: PHP kann, ähnlich wie JavaScript, in HTML-Dokumente eingebettet werden. Die Einbettung ist jedoch nicht zwingend erforderlich. PHP kann genauso gut verwendet werden, um HTML-Code zu erzeugen.

PHP-Tags Bei beiden Varianten – Einbettung in den Code und das Erzeugen des Codes – werden die Start- und Ende-Tags für die Scripts gleich verwendet. PHP-Quellcode wird immer zwischen den Tags

```
<?php  
und  
?>
```

notiert, egal, ob eine Verwendung innerhalb eines HTML-Dokuments oder als eigenständiges Script erfolgt.

```
<html>  
  <head>  
    <title>Listing 1.1</title>  
  </head>  
  <body>  
    <?php  
      echo "Hallo WorldWideWeb!";  
    ?>  
  </body>  
</html>
```

Listing 39.1 Erstes PHP-Script

Das PHP-Script aus Listing 39.1 ist ein Beispiel für PHP-Code, der in ein HTML-Dokument eingebettet wurde. Kopieren Sie dieses Script in eine neue Datei, und speichern Sie es im Dokumenten-Verzeichnis Ihres Webservers ab.



Beachten Sie, dass Sie als Dateiendung **.php** wählen müssen, auch wenn das Script äußerlich einem HTML-Dokument gleicht. In einigen Fällen kann es auch sein, dass die Endung der Datei **.php3** oder **.php4** lauten muss. Dies hängt von der Webserver-Konfiguration ab. Häufig kommt auch die Endung **.phtml** oder **.phtm** zum Einsatz.

Den Grund dafür werde ich Ihnen im nachfolgenden Kapitel 40, *Variablen und Operatoren*, noch genauer erläutern. Zurück zum Listing 39.1. Alles, was außerhalb des Tag-Paares `<?php` und `?>` notiert wurde, wird als normaler HTML-Code interpretiert und ganz normal im Browser ausgegeben. Alles, was innerhalb des PHP-Tag-Paares notiert

wurde, wird als PHP-Code interpretiert und entsprechend vom PHP-Interpreter verarbeitet. Die Anweisung

```
echo "Hallo WorldWideWeb!";
```

weist den Interpreter an, die Ausgabe `Hallo WorldWideWeb!` zu erzeugen. Der Befehl `echo` erzeugt also eine Ausgabe. Nach `echo` kann entweder eine feste Zeichenkette (wird in doppelten oder einfachen Anführungsstrichen notiert) oder eine Variable folgen. Anschließend wird die vollständige Anweisung mit einem Strichpunkt bzw. Semikolon abgeschlossen (;).

Rufen Sie das Listing 39.1 nun in Ihrem Browser auf. Die Ausgabe des Scripts sollte folgendermaßen aussehen:

```
Hallo WorldWideWeb!
```

Herzlichen Glückwunsch, dies war Ihr erstes PHP-Script.

39.2 PHP und die Dateierendungen

Was hat es aber mit der Endung `.php` auf sich? Genauso wie die Endungen `.htm` oder `.html` eine Datei als HTML-Dokument ausweisen, definiert die Endung `.php` eine Datei als PHP-Script. Wenn nun PHP-Code in einem HTML-Dokument eingebettet wurde, muss dem Webserver klar gemacht werden, dass innerhalb dieses Dokuments spezielle Anweisungen enthalten sind, die vor der Übertragung der Datei an den Browser interpretiert werden müssen. Andernfalls würde der PHP-Code einfach mit dem Rest an den Browser gesendet werden. Damit kann der Benutzer freilich nicht viel anfangen.

Damit der Server nun aber den richtigen Interpreter wählt, müssen Sie die Endung `.php` verwenden. In der Regel ist diese Dateierendung mit dem PHP-Interpreter verknüpft, der sich dann um die Ausführung des PHP-Codes und die anschließende Ausgabe im Browser kümmert. Der Interpreter kann nun anhand der PHP-Tags erkennen, ob es sich um HTML-Code oder um PHP-Code handelt.

Im Fall des Listing 39.1 findet er zu Beginn lediglich HTML-Code und sendet diesen ohne Umwege an den Browser. Erst wenn er auf das Tag `<?php` stößt, beginnt seine eigentliche Aufgabe. Den nachfolgenden Code interpretiert er nun als PHP-Code und führt die Anweisungen entsprechend aus; und zwar so lange, bis er auf das Tag `?>` stößt. Alles, was danach folgt, wird wieder als HTML-Code bewertet und direkt zum Browser gesendet.

Dies bedeutet jedoch nicht, dass das PHP-Tag-Paar nur einmal in einer Datei verwendet werden darf. Ein Beispiel:

Verknüpfung
mit PHP-Inter-
preter

```

<html>
  <head>
    <title>Listing 1.2</title>
  </head>
  <body>
    <?php
      echo "Hallo WorldWideWeb!";
    ?>
    <?php
      echo "</body></html>";
    ?>

```

Listing 39.2 Mehrere PHP-Tag-Paare in einem Script

Der Unterschied zum Listing 39.1 liegt im zweiten PHP-Tag-Paar. Anstatt einfach den restlichen HTML-Code im Browser auszugeben, erfolgt dies mit der `echo`-Anweisung. Das Ergebnis beider Listings (sowohl Listing 39.1 als auch Listing 39.2) ist das gleiche, wobei das Listing 39.2 eher unsinnig ist.

39.3 PHP in eigenen Dateien

Wie bereits erwähnt wurde, kann ein PHP-Script auch lediglich aus PHP-Code bestehen. Der HTML-Code muss dann aber mit der `echo`-Anweisung ausgegeben werden.

```

<?php
  echo "<html>";
  echo "<head>";
  echo "<title>Listing 1.3</title>";
  echo "</head>";
  echo "<body>";
  echo "Hallo WorldWideWeb!";
  echo "</body>";
  echo "</html>";
?>

```

Listing 39.3 PHP-Script, das ausschließlich PHP-Code enthält



Anders als Perl sendet PHP automatisch den korrekten MIME-Typ an den Browser, also `text/html`. Sie müssen diesen Header nur dann ausgeben, wenn die nachfolgenden Daten nicht zu einem HTML-Dokument, sondern zu einer Grafik oder Ähnlichem gehören.

In Listing 39.3 wird das gesamte HTML-Dokument mit PHP ausgegeben. Auch bei diesem Beispiel fehlt der praktische Nutzen. Es verdeutlicht aber, dass auch in einem solchen PHP-Script das PHP-Tag-Paar notiert werden muss, da der Interpreter sonst den reinen Quelltext an den Browser sendet.

Würden Sie die Tags `<?php` und `?>` aus dem Listing 39.3 entfernen, sähe die Ausgabe im Browser folgendermaßen aus:

```
echo ""; echo ""; echo ""; echo ""; echo "";
    echo "Hallo WorldWideWeb!"; echo ""; echo "";
```

39.4 Quellstrukturierung und Kommentare

Auch in PHP ist eine vernünftige Quellstrukturierung und -kommentierung äußerst wichtig. Der Grund dafür dürfte mittlerweile hinlänglich bekannt sein.

Kommentieren können Sie in PHP auf drei Arten, entweder einzeilig oder mehrzeilig. Für einzeilige Kommentare notieren Sie einfach zwei Schrägstriche `//`. Alles was nach diesen beiden Schrägstrichen bis zum Zeilenende folgt, wird dann als Kommentar interpretiert und vom PHP-Interpreter ignoriert. Anstelle der Schrägstriche können Sie aber auch das Raute-Zeichen `#` verwenden.

Zwei einzeilige
Kommentar-
arten

Für mehrzeilige Kommentare müssen Sie `/*` und `*/` verwenden. Dabei kennzeichnet `/*` den Beginn eines mehrzeiligen Kommentars und `*/` das Ende des Kommentars.

Eine mehr-
zeilige
Kommentarart

```
echo "Hallo WordWideWeb!"; // ein einzeiliger Kommentar
echo "Hallo Welt!";         /* Mehrzeiliger
                             Kommentar */
echo "Hallo Alle!";        # ebenfalls einzeiliger Kommentar
```

39.5 Textausgabe

In vielen Fällen ist die Ausgabe von Text mit der `echo`-Anweisung ungenügend, gerade in Bezug auf Zeilenumbrüche. Zwar können Sie einen optischen Zeilenumbruch im Browser mit dem Element `br` bewirken, wenn Sie aber Text ausgeben möchten, der mit dem `pre`-Element formatiert werden soll, hat das `br`-Element keinerlei Auswirkungen.

Dafür benötigen Sie das Steuerzeichen `\n`. Dies erzeugt im Quelltext des HTML-Dokuments einen Zeilenumbruch (`n` = engl. *new line* = dt. *neue Zeile*). An welcher Stelle Sie das Steuerzeichen notieren, ist egal.

Zeilen-
umbrüche

```
echo "Hallo Welt!\n";
```

Die Anweisung erzeugt einen im Browser nicht sichtbaren Zeilenumbruch nach der Zeichenkette `Hallo Welt!`.

```
echo "<pre>Hallo \n Welt!</pre>";
```

Diese Anweisung hingegen erzeugt sowohl im Quelltext als auch im Browser einen sichtbaren Zeilenumbruch.

print-Anweisung Analog zur `echo`-Anweisung können Sie auch die `print`-Anweisung verwenden. Sowohl die Syntax als auch die Wirkungsweise beider Anweisungen sind gleich. Die Anweisung `print` kommt noch aus den Anfangszeiten von PHP, woran auch die Abstammung von Perl deutlich wird.

```
echo "Hallo Welt!<br>";  
print "Hallo Welt!<br>";
```

Die Ausgabe lautet:

```
Hallo Welt!  
Hallo Welt!
```

39.5.1 Formatierte Textausgabe

Anders sieht es bei der Funktion `printf` aus. Sie ist stark an die C-Syntax¹ angelehnt. Mit dieser Funktion ist es möglich, Daten für die Ausgabe speziell zu formatieren. Vor allem bei der Ausgabe von Zahlenwerten ist dies hilfreich.

Dabei wird die Ausgabe mit Hilfe von Platzhaltern gesteuert.

```
int printf(string format [, mixed args...])
```

Die Formatierungszeichenkette mit den Platzhaltern wird als erster Parameter an die Funktion übergeben, und anschließend folgen die einzelnen Daten. Die Formatierung kann dabei sowohl normalen Text als auch die Platzhalter enthalten.

Platzhalter	Erklärung
<code>%b</code>	Binäre Darstellung von Ganzzahlen
<code>%c</code>	Darstellung einer Ganzzahl als ASCII-Zeichen
<code>%d</code>	Darstellung einer Ganzzahl als Dezimalzahl
<code>%e</code>	Fließkommazahl in Exponentialdarstellung
<code>%f</code>	Fließkommadarstellung
<code>%g</code>	Entweder <code>%e</code> oder <code>%f</code> , je nachdem, was kürzer ist
<code>%o</code>	Darstellung einer Ganzzahl als Oktalzahl
<code>%s</code>	Darstellung als Zeichenkette
<code>%x, %X</code>	Darstellung einer Ganzzahl in Hexadezimaldarstellung, entweder in Kleinbuchstaben (<code>%x</code>) oder Großbuchstaben (<code>%X</code>)

Tabelle 39.1 Platzhalter zur Formatierung mit der Funktion `printf`

Hier sehen Sie ein Beispiel für die unterschiedlichen Ausgabeformatierungen:

```
<?php
printf("Binär: %b<br>",167);
printf("ASCII: %c<br>",167);
printf("Dezimal: %d<br>",167);
printf("Exponential: %e<br>",167);
printf("Fließkomma: %.2f<br>",167);
printf("Oktal: %o<br>",167);
printf("Zeichenkette: %s<br>",167);
printf("Hexadezimal: %x<br>",167);
printf("Hexadezimal: %X<br>",167);
?>
```

Listing 39.4 Formatierte Ausgabe von Daten

Im Browser erzeugt das Listing 39.4 die folgende Ausgabe:

```
Binär: 10100111
ASCII: §
Dezimal: 167
Exponential: 1.67000
Fließkomma: 167.00
Oktal: 247
Zeichenkette: 167
Hexadezimal: a7
Hexadezimal: A7
```

39.6 Alternative PHP-Tags

Neben der Verwendung der Tags `<?php` und `?>` können Sie noch weitere Schreibweisen verwenden.

Bei der Kurzform werden die Tags `<? und ?>` verwendet.

```
<? echo "Hallo Welt!"; ?>
```

Die Langform verwendet das HTML-Element `script` zur Einbindung von Scripts.

```
<script language="php">
  echo "Hallo Welt!";
</script>
```

Bei der ASP-Form werden die Tags `<%` und `%>` verwendet, die normalerweise bei Active Server Pages Anwendung finden.

```
<% echo "Hallo Welt!"; %>
```



Sowohl die Kurzform als auch die ASP-Form müssen nicht mit jeder PHP-Installation funktionieren, da dies nur Optionen sind, die in der Datei **php.ini** unter Umständen aktiviert werden müssen. Bei den meisten Providern haben Sie jedoch keinen Zugriff auf diese Datei. Aus diesem Grund sollten Sie die bisher verwendete Form `<?php ... ?>` benutzen.

39.7 Zusammenfassung

- ▶ PHP-Code kann sowohl in HTML eingebettet werden als auch in eigenen Dateien notiert werden.
- ▶ Generell müssen Sie PHP-Tags verwenden, innerhalb derer der PHP-Code notiert wird. Die gängigste Variante ist `<?php ... ?>`.
- ▶ Ausgaben können mit den Anweisungen `echo` oder `print` oder mit der Funktion `printf` erfolgen.
- ▶ Einzeilige Kommentare werden durch `//` eingeleitet, und mehrzeilige Kommentare werden von `/*` und `*/` umgeben.
- ▶ Die Endung eines PHP-Scripts sollte `.php` lauten.

39.8 Fragen und Übungen

1. Worin besteht der Unterschied zwischen der Anweisung `echo` und der Funktion `printf` für die Ausgabe im Browser?
2. Mit welchem Steuerzeichen können Sie im HTML-Quelltext einen Zeilenumbruch erzeugen?
3. Schreiben Sie ein PHP-Script, das ein vollständiges HTML-Dokument erzeugt und zusätzlich eine Überschrift der 1. Ordnung mit dem Text »Dynamische Webseiten mit PHP« ausgibt. Strukturieren Sie den HTML-Quelltext mit Zeilenumbrüchen.

Index

`$_` 505, 514
`$_COOKIE` 667
`$_ENV` 693
`$_FILES` 668
`$_GET` 662
`$_POST` 662, 663
`$_SERVER` 661, 693
`$o` 491
`$ENV'CONTENT_LENGTH'` 544
`$ENV'DOCUMENT_ROOT'` 558
`$ENV'QUERY_STRING'` 539
`$HTTP_COOKIE_VARS` 667
`$HTTP_GET_VARS` 662, 664
`$HTTP_POST_FILE` 668
`$HTTP_POST_VARS` 662, 664
`$HTTP_SERVER_VARS` 660
\$-Zeichen 605
&-Operator 975
.xhtml 876
<!--_..._--> 870
<?...?> 867
@ 486, 776, 798
@_ 505, 506
@font-face 293
@import 273
@-Operator 710
@page 366
__construct 708
__destruct 709
|-Operator 977
500 Internal Server Error 77

A

abbr 146
acronym 146
ActiveX 52, 62, 253, 255
 Kalender 256
 WindowsMediaPlayer 256
addslashes 784
adminfunction.inc.php 957
Administrationsbereich 957
Adobe 881
a-Element (HTML) 213

_blank 247
_parent 218, 248
_self 248
_top 218, 248
active 347
Dokumentintern 217
focus 347
globale Links 215
Grafiken 217
hover 347
href 213
link 347
lokale Links 213
mailto 216
name 218
target 218, 247
visited 347
AGB 1063
alert 388, 393
AllowOverride 936
ALTER TABLE 734
 ADD 734
 ADD PRIMARY KEY 734
 CHANGE 734
 DROP 734
 DROP PRIMARY KEY 734
 MODIFY 734
alternative Syntax-/Kontrollstrukturen 628
Anbieterkennzeichnung 1064
AND 743
and 495
Apache Server 44
ApacheFriends 87
applet 206
 code 206
 height 206
 width 206
area 219
alt 219
circle 219, 220
coords 219
href 219
poly 220

- rect 220
- shape 219
- Arpanet 49
- array_pop 648
- array_push 648
- array_search 650
- array_shift 649
- array_unshift 649
- Array-Klasse 413
 - new 413
- Arrays 413, 486
 - Index 414
- as 623
- Attributbedingte Formatierung 297
- Attribute 130
- Ausgabemedien 272
- Außenabstand 315
- Auth & Auth 918
- AuthConfig 936
- AUTO_INCREMENT 732, 736

B

- background 309
 - attachment 307
 - fixed 308
 - scroll 308
 - color 278, 305
 - image 307
 - position 308
 - repeat 307
 - no-repeat 307
 - repeat 307
 - repeat-x 307
 - repeat-y 307
- base_convert 656
- Bedingungsprüfung 619
- b-Element (HTML) 144, 235
- Bereichsauflösungsoperator 638
- Bezeichnungsregeln 868
- bgsound 258
- big 144
- BINARY 736
- binmode 569, 831
- Bitweise Operatoren 975, 1073
- Blog 979
- bloggen 979

- BMP 193, 194
- body 128, 130, 145
 - alink 155
 - background 202
 - bgcolor 155, 203
 - leftmargin 258
 - link 155, 217
 - marginheight 259
 - marginwidth 259
 - text 155
 - topmargin 258
 - vlink 155
- border 320, 333
 - collapse 333
 - seperate 333
- border-bottom 320
- border-color 322
- border-left 320
- border-right 320
- border-spacing 336
- border-style 322
- border-top 320
- border-width 322
- bottom 323
- br 140
- break 390, 391, 621, 625
- button 229
 - accesskey 237
 - reset 229
 - submit 229
 - type 229

C

- ca-bundle.crt 926
- Camino 67
- caption 166
 - align 166
 - bottom 166
 - top 166
- caption-side 335
- Cascading Style Sheets 263
- case 390, 625
- case sensitive 605, 873
- catch 713
- CGI 478, 480

- channel 909
 - description 909
 - language 909
 - link 909
 - title 909
- chdir 561, 676
- chomp 854
- chr 520, 642
- circle 888
 - cx 888
 - cy 888
 - fill 888
 - r 888
 - stroke 888
 - stroke-width 888
- cite 146
- class 206, 410
- close 563
- closedir 553, 554, 675
- CMS 931
- code 146
- col 173, 185
 - align 173
 - char 173
 - width 185
- colgroup 185
 - span 186
 - width 186
- color 267, 277, 286
- content 350, 354
 - counter 354
- Content Management System 931
- Content Syndication 907
- continue 627
- Cookies 54, 548
 - expires 549
 - name 549
 - path 549
 - value 549
- copy 684
- cos 534, 657, 903
- count 646
- counter-increment 355
- counter-reset 355
- CREATE DATABASE 731

- CREATE TABLE 732
- crypt 522, 645, 937
- CSS 263, 270
 - Ausgabemedien 272
 - Browserunterstützung 264
 - Einheiten 273
 - Selektor 267
 - Zentrale Formatierung 264
- cURL 921
- curl_errno 925
- curl_error 925
- curl_exec 925
- curl_init 924
- curl_setopt 924, 926
- CURLOPT_HTTPHEADER 925
- CURLOPT_POST 924
- CURLOPT_POSTFIELDS 924
- CURLOPT_RETURNTRANSFER 924
- cursor 359

D

- Date-Klasse 406, 418
 - getDate 409
 - getDay 415
 - getFullYear 415
 - getHours 408, 417
 - getMinutes 408, 417
 - getMonth 409, 415
 - getSeconds 408, 417
 - getTime 419
 - getYear 408, 415
- Datenbank-Abfragespache 723
- Datentypen 606
 - Arrays 610
 - assoziative 611
 - einfache 610
 - mehrdimensionale 611
 - Boolean 607
 - Fließkommazahl 607
 - Integer 606
 - NULL 612
 - Resource 612
 - String 607
 - Typumwandlung 609
- DBH
 - do 767

- err 772
- errstr 772
- execute 761
- mysql_insertid 761, 767
- prepare 761
- quote 769
- DBI 758
 - connect 758, 759
 - disconnect 760
 - do 760
 - err 771
 - errstr 771
- dd 160
- DEFAULT 736
- default 391, 625
- define 971, 975
- deg2rad 900, 903
- DELETE 751
 - FROM 751
 - WHERE 751
- delete 524
- demilitarisierte Zone 117
- DeNIC 60
- deprecated 148, 162, 339
- Destruktor 707, 709
- dfn 146
- DHCP 118
- DHTML 449
- die 834
- dir 673
 - close 673
 - dir 673
 - handle 673
 - path 673
 - read 673
 - rewind 673
- div 313
- dl 160, 921
- DMZ 117
- DNS 110
- document 403, 406
 - all 450
 - bgColor 404
 - cookie 466
 - getElementById 453
 - getElementsByName 453
 - getElementsByTagName 453
 - lastModified 403
 - title 403
 - URL 404
 - write 376, 400
- Document Type Definition 133, 870
- DOM 450
 - Assoziierte Knoten 454
 - firstChild 454
 - innerHTML 454
 - innerText 454
 - Internet Explorer 450
 - Knoten 453
 - Netscape 451
 - nodeValue 454
 - Testen 455
 - W3C 452
- Domäne 112
- DROP DATABASE 732
- DROP TABLE 733
- dt 160
- DTD 870
 - HTML 133
 - XML 875
- DynDNS 116

E

- each 513, 650
- eBay Developer Program 916
- eBay-API 916, 918
- echo 599
- ECMAScript 373
- Eigene Fehlertypen 714
- Eigenschaften 404
- Einbinden von Dateien 626
- Eingabefelder 225
 - Checkboxes 232
 - einzeilig 225
 - Passwort 225
 - Radiobuttons 231
 - Schaltflächen 228
- Elemente 128
 - Hierarchie 129
 - Vererbung 149

- ellipse 888
 - rx 888
 - ry 888
- else 389, 509, 619
- elseif 510, 620
- em 146, 263
- eMacs 44
- E-Mail 55
- embed 205, 258
- empty-cells 337
- Entities 132
- ENUM 960
- envelope 914
- eot 294
- eq 494
- ErrorLevel 920
- Escape-Sequenzen 608
- Escape-Zeichen 608
- eval 472
- Exception 713
 - getMessage 713
- exists 525
- EXPLAIN 733, 787
- explode 661
- extends 637

- F**
- Farben
 - fill-opacity 889
 - stroke-opacity 889
- Farbworte 151, 277
- fclose 678
- feof 680
- fgets 680, 817
- fieldset 235
- file 953
- file_exists 680
- filemtime 683
- fileowner 683
- filesize 683
- filetype 683
- fill 882
- filter 362
 - Alpha 362
 - Blur 363
 - Chroma 363
 - DropShadow 363
 - FlipH 364
 - FlipV 364
 - Glow 364
 - Gray 364
 - Invert 364
 - Mask 365
 - Shadow 365
 - Wave 365
 - XRay 366
- finally 472
- Firebird 66
- FireFox 66
- Flags 1075
- Flash 880
- FliaI 117
- float 350
- font 142, 148, 174, 175, 291
 - color 150, 151
 - face 150
 - size 150, 151
- font-family 267, 280, 882
 - cursive 281
 - fantasy 281
 - monospace 281
 - sans-serif 281
 - serif 281
- font-size 267, 282, 882
- font-style 283
 - italic 283
 - oblique 283
- font-variant 283
 - normal 284
 - small-caps 284
- font-weight 284
 - bold 284
 - bolder 284
 - lighter 284
 - normal 284
- fopen 678
- form 223
 - action 223, 224
 - enctype 570
 - get 224

- mailto 224
- method 223, 224
- post 224
- Formatvorlagen 263
- Formulare 223
- FoxServ 87
- fpassthru 695
- fputs 678
- frame 241, 875
 - border 248
 - frameborder 248
 - framespacing 248
 - name 242
 - noresize 247
 - scrolling 248
 - src 241
- Frames 239
- frameset 240
 - cols 240
 - rows 240
- fsockopen 694
- FTP 58
 - ascii 102
 - binary 103
 - bye 103
 - cd 102
 - delete 102
 - dir 102
 - get 102
 - help 103
 - Kommandozeilenclient 101
 - lcs 102
 - ls 102
 - mget 102
 - mput 102
 - status 103
 - Webseite auf Server übertragen 97
 - WS FTP 97
- Führende Nullen 417
- function 393, 410, 631
- Funktion 631
 - Parameter 632
 - Rückgabewerte 633

G

- GD 827, 828, 829, 830, 831, 834, 835, 836, 838, 839, 841, 843, 844, 845, 846, 847, 848, 850, 857, 858
- ge 495
- getcwd 676
- gethostbyaddr 694
- gethostbyname 694
- gethostbyname1 694
- gettype 609
- GIF 193
 - Animation 194
 - Blind-GIF 202
 - Fake-GIF 202
 - Interlacing 194
 - Transparenz 194
- global 959
- gmtime 532
- Goldene Regeln 259
- Gruppenformatierungen 296
- gt 494
- Gültigkeitsbereich 501
- Gutmans, Andi 597

H

- h1-h6 142
 - align 143
- hashbang 480, 482, 500
- Hashes 489
 - Unbekannte Schlüssel 490
 - Zugriff auf Elemente 489
- head 128, 130
- header 480
- Header modifizieren 918
- height 325
- hex 535
- Hex-Tripel 152, 277
- Hierarchie, XML 868
- hr 257
 - align 257
 - color 257
 - noshade 257
 - size 257
 - width 257
- ht 936

htaccess 936, 949

htm 127

HTML 125, 127, 128

 Kommentare 131

 Version 1.0 126

 Version 2.0 126

 Version 3.2 126

 Version 4.0/4.01 127

htpasswd 937

httpd.conf 936

I

i 144

if 385, 388, 509, 619

iframe 250

 height 251

 name 251

 src 251

 width 251

IIS 45

image/svg+xml 882

imageAlphaBlending 806

imageArc 803, 804, 814, 899

imageColorAllocate 794, 806, 808

imageColorAt 819, 821

imageColorResolveAlpha 806, 808

imageColorsForIndex 819, 821

imageCopyResized 810, 812

imageCreate 794, 798, 806

imageCreateFromJPEG 798, 812

imageCreateFromPng 798, 812

imageCreateTrueColor 806, 813

imagedashedline 800

imageDestroy 797

imageEllipse 802, 804

imageFill 805

imageFilledArc 814, 818

 IMG_ARC_CHORD 814

 IMG_ARC_EDGED 814

 IMG_ARC_NOFILL 814

 IMG_ARC_PIE 814, 818

imageFilledEllipse 802

imageFilledRectangle 801, 802

imageFontHeight 795

imageFontWidth 795

imageJPEG 796

imageline 800

imageLoadFont 807

Imagemaps 219

imagePng 796

imageRectangle 801

imagesetpixel 800

imageString 795

imageStringUp 809

imagesx 799

imagesy 799

imageTTFText 807

img 197

 align 200

 alt 198

 Alternativ-Text 198

 border 199

 height 199

 src 197

 Transparenz 200

 usemap 219

 width 199

implode 968

Impressum 1064

include 626, 678

index 521, 540

Individual API License 917

Individual Tier 917

Individuelle Formate 303

Inkonsistenz 725

Innenabstand 317

input 225

 accept 570

 accesskey 237

 checkbox 232

 checked 232

 file 570

 maxlength 225, 570

 name 229

 password 225

 radio 231

 readonly 226

 reset 228

 size 225

 submit 228

- tabindex 236
- type 225
- value 225, 229, 232
- INSERT 749
 - FROM_UNIXTIME 770, 784
 - VALUES 749
- int 536
- Internet Explorer 61
- Interpreter 477
- IPv4 119
- IPv6 118
- is_dir 674
- is_executable 683
- is_readable 683
- is_uploaded_file 669
- is_writable 683
- ISAPI 478
- isset 609
- item 910
 - comments 910
 - description 910
 - link 910
 - pubDate 910
 - title 910

J

- Jahr-2000-Fehler 425
- Java 51
- JavaApplets 51, 206, 449
- JavaScript 373, 395
 - Zugriff auf Elemente 419
- join 527
- jpe 195
- JPEG 195
 - Auflösung 195
 - JPEG2000 195
 - Kompressionsstufe 195
- js 379
- JScript 375

K

- kbd 146, 263
- keys 490, 524
- KHTML 69

- Klassen 300, 633
 - Eigenschaften 634
 - Konstruktoren 635
 - Methoden 634
 - Objekte instanzieren 635
 - Vererbung 637
- Kommentare
 - XML 867, 870
- Kompatibilitätslevel 919
- Konkatenationsoperator 615
- Konqueror 67
- Konstruktor 707

L

- Larry Wall 45, 477
- last 516
- layer 450, 451
 - clear 452
 - close 452
 - open 452
 - write 452
- Layer-Technik 450
- lc 519
- le 495
- left 323
- legend 235
 - accesskey 237
- length 520
- Lerdorf, Rasmus 46, 597
- letter-spacing 287
- li 157, 159
 - value 163
- line 886
 - stroke 886
 - stroke-width 886
 - x1 886
 - x2 886
 - y1 886
 - y2 886
- linearGradient 890
- link 270
 - href 271
 - rel 271
- list 650

- Listen 486
 - Zugriff auf Elemente 488
- list-style 344
- list-style-image 343
- list-style-position 341
- list-style-type 339
- LiveScript 373
- LOAD DATA 752
 - FIELDS TERMINATED BY 753
 - INFILE 752
 - INTO TABLE 752
 - LINES TERMINATED BY 753
- local 501, 502
- localtime 529, 651
- lt 494
- Lynx 70

- M**
- Macromedia Flash 52, 203, 881
- map 219, 528
- margin 315
- margin-bottom 316
- margin-left 316
- margin-right 316
- margin-top 316
- marks 368
- Math-Objekt
 - pow 435
 - round 419
- maxheight 326
- maxwidth 326
- md5 962
- MDStV 1064
- Mediendienstestaatsvertrag 1064
- Mensch-zu-Maschine 913
- meta 253
 - author 253
 - content 255
 - date 254
 - description 253
 - http-equiv 255
 - keywords 254
 - robots 255
- META-Tags 253
 - Automatische Weiterleitung 255

- Dublin-Core 254
- Methoden 405
 - definieren 411
- microtime 656
- MIME-Typ 882, 918
 - application/java 208
 - application/java-vm 208
 - image/gif 209
 - image/jpeg 209, 480
 - image/png 209
 - image/svg+xml 882, 902
 - multipart/form-data 570
 - text/css 268
 - text/html 480
- minheight 326
- minwidth 326
- mkdir 561, 684
- mktime 654
- Module 477
- Modulo 384, 492
- move_uploaded_file 669
- Mozilla 65
- my 485, 501
- MySQL 46
- mysql_affected_rows 782
- mysql_close 776
- mysql_connect 775
- mysql_errno 785
- mysql_error 785
- mysql_fetch_array 779, 781
- mysql_fetch_object 781
- mysql_fetch_row 779, 781, 786
- mysql_field_flags 788
- mysql_field_len 788
- mysql_field_name 788
- mysql_field_type 788
- mysql_free_result 780
- mysql_insert_id 783
- mysql_list_dbs 786
- mysql_list_fields 787
- mysql_list_tables 787
- mysql_num_fields 780, 789
- mysql_num_rows 780
- mysql_query 778
- mysql_result 780

- mysql_select_db 776, 778
- MySQL-Datentypen 735
 - BIGINT 735
 - BLOB 736
 - CHAR 735
 - DATE 736
 - DATETIME 736
 - DECIMAL 735
 - DOUBLE 735
 - ENUM 736
 - FLOAT 735
 - INTEGER 735
 - LONGBLOB 735
 - LONGTEXT 735
 - MEDIUMBLOB 735
 - MEDIUMINT 735
 - MEDIUMTEXT 735
 - numerische 735
 - SET 736
 - SMALLINT 735
 - TEXT 735
 - TIME 736
 - TINYBLOB 735
 - TINYINT 735
 - TINYTEXT 735
 - VARCHAR 735
 - vermischte 736
 - Zeichen- und Zeichenketten 735
- MySQL-Konsole 729
 - Befehlsübersicht 729

N

- nachgestellte Bedingungsprüfung 559
- nachprüfende Schleifen 620
- Namenskonventionen 730
- Namespace 915
- NaN 423
- NAT 119
- navigator-Objekt 427
 - appName 429
 - appVersion 430
 - cookieEnabled 430
 - language 430
 - platform 430

- plugins 436, 439
 - plugins.description 438
 - plugins.length 437
 - plugins.name 437
 - userAgent 430
 - userLanguage 430
- Netiquette 259
- Netscape Navigator 62
- new 407
- next 517
- noframes 247
- NOT 496, 743
- NOT NULL 737
- Notepad 43
- NSFnet 49
- NULL 435, 737

O

- object 204, 208, 256, 883
 - align 205, 209
 - classid 204, 208
 - codebase 205
 - codetype 208
 - data 209, 883
 - height 205, 883
 - type 883
 - width 205, 883
- Objekte 631
- objektorientierte Datenbanksysteme 724
- oct 536
- ol 156
 - A 163
 - a 163
 - l 162
 - i 162
 - start 163
 - type 162
- OmniHTTpd 45
- onAbort 460
- onBlur 461
- onChange 461
- onClick 461
- onDbClick 462
- onError 462
- onFocus 462

onKeyDown 462
 onKeyPress 462
 onKeyUp 462
 onLoad 394, 420, 463
 onMouseDown 463
 onMouseMove 463
 onMouseOut 464
 onMouseOver 458
 onMouseover 464
 onMouseup 464
 onReset 464
 onResize 465
 onSelect 465
 onSubmit 464
 onUnload 465
 OOP 631
 open 562, 854
 opendir 553, 675
 Opera 64
 Operatoren 383, 385, 494, 565, 612, 743

- 384, 492, 612, 748
- 387, 493, 613
- ! 496, 616
- != 385, 494
- % 384, 489, 492, 613
- % ... %> 603
- %= 493
- %ENV 491, 524, 539, 591
- && 386, 495, 616
- * 384, 492, 612, 748
- ** 492, 535
- **= 493
- *= 387, 493, 614
- + 384, 386, 492, 612, 748
- ++ 387, 493, 613
- += 387, 493, 614
- .= 615
- / 384, 492, 613, 748
- /* ... */ 380, 601
- // 379, 601
- /= 387, 493, 614
- = 387, 493, 614
- = 383, 385, 407, 492, 494, 743
- == 385, 493
- => 489, 611, 623
- =~ 576
- > 634
- > 385, 494, 565, 743
- >= 385, 494, 743
- >> 565
- ?> 598
- ^ 495
- || 386, 495, 616
- AND 616
- arithmetische 612
- logische 616
- OR 616
- Trinität 615
- Vergleich 614
- XOR 616
- Zeichenketten 615

 option 232

- selected 233
- value 233

 OR 743
 or 495
 ord 520, 642
 orphans 368
 overflow 326

- auto 326
- hidden 326
- scroll 327
- visible 327

P
 p

- align 138
- first-letter 349
- first-line 349
- padding 317
- padding-bottom 318
- padding-left 318
- padding-right 318
- padding-top 318
- page-break-after 367
- page-break-before 367
- param 205, 208, 256, 546
- parseInt 423, 431
- path 893, 903
 - d 893

- H, h 893
- L 893, 898
- large-arc-flag 896
- M 893
- Q, q 893
- Q,_q 894
- sweep-flag 896
- V, v 893
- z 893
- pdf_begin_page 701
- pdf_close 702
- pdf_end_page 702
- pdf_findfont 702
- pdf_get_buffer 702
- pdf_new 701
- pdf_open_file 701
- pdf_set_info 701
- pdf_set_parameter 702
- pdf_show_xy 702
- Perl 45, 475, 480
 - h 481
 - U 481
 - V 481
 - v 481
 - W 481
 - w 480, 481
 - X 481
- Perl Package Manager 757
- Personal Webserver 45
- pfr 294
- Phönix 66
- PHP 46, 595, 599, 936
- php.ini 77, 793
- PHP/FI 597
- php_curl.dll 921
- php_gd.dll 793
- php_pdf.dll 77
- phpinfo 693, 808
- PHP-Interpreter 599
- PhpMyAdmin 46
- PHP-Tags 598
- pi 867
- pl 478, 480
- Platzhalter 602
- Plug-Ins
 - Flash-Player 204
 - JavaApplets 206
- PNG 196
- polyline 886
 - fill 886
 - points 886
- pop 526
- Portforwarding 117
- position 323
 - absolute 323
 - relative 323
 - static 323
- Potenz 492
- PPM 757
- pre 141
 - align 142
- preg_match 698
- preg_match_all 699
- preg_replace 699
- preg_split 700
- Presserecht 1064
- Primärschlüssel 726
- PRIMARY KEY 733, 737
- print 479, 486, 602
- printf 602
- Printmedien 366
 - Alleinstehende Zeilen 368
 - Größe und Ränder 367
 - Medientyp festlegen 366
 - Schnittmarken 368
 - Seitenumbruch 367
- private 706
- Processing Instructions 867, 875, 882
- prompt 388, 396
- protected 706
- Pseudoformate 347
 - Absätze 349
 - Verweise 347
- public 705
- Punkt-vor-Strich 384, 492
- push 526

R

- radialGradient 891
 - id 891

- Rahmenabstand 336
- rand 533, 655
- range 647
- Rastergrafik 879
- RDF 907
- read 544
- readdir 553, 554, 675
- Recht am eigenen Bild 1063
- rect 887
 - fill 887
 - height 887
 - rx 887
 - ry 887
 - stroke 887
 - width 887
 - x 887
 - y 887
- redo 516
- Redundanz 725
- Reguläre Ausdrücke 575, 698
 - \$ 578
 - * 579
 - + 578
 - /./ 575
 - ? 578
 - ^ 577
 - | 582
- Bindungsoperator 576
- c 582
- Escape-Zeichen 577
- Flags 582
- g 582
- Gruppierung 580
- Häufigkeit 579
- i 582
- m 582
- o 582
- Quantifier 578
- s 582
- s/// 583
- Whitespacezeichen 581
- x 582
- Zeichenklassen 580

Rekursion 556, 676

Relation 725

- relationale Datenbanksysteme 724
- Remote Procedure Calls 913
- rename 684
- request 920
- require 626
- Resource Description Framework 907
- return 397, 506, 633
- reverse 527
- rewinddir 676
- rgb 279
- right 323
- rmdir 562, 684
- round 903
- RPC 914
- rsort 647
- RSS 907, 909
 - 0.90 907
 - 0.91 907
 - 0.92 907
 - 0.94 907
 - 1.0 907

S

- s/strike 144
- Safari 69
- samp 147
- Sandbox 920
 - Keys 918, 920
- Schemata 870, 871
- Schleifen
 - do...while 620
 - do..until 511
 - do..while 399, 512
 - for 399, 512, 622
 - foreach 623
 - spezielle Notation 515
 - while 397, 513, 621
- Schriftartendateien 292
- Schriftformatierung 280
- screen-Objekt 431
 - availHeight 433
 - availWidth 433
 - colorDepth 434
 - pixelDepth 434
 - width 432

- script 375
 - src 379
 - type 375
- Scriptsprachen 53
- scrollbar-* 360
- SELECT 739
 - * 740
 - Aliase 754
 - AS 747
 - ASC 745
 - AVG 748
 - BETWEEN 744
 - COUNT 741, 748
 - DESC 745
 - DISTINCT 747
 - FIELDS TERMINATED BY 752
 - FROM 740
 - GROUP BY 746
 - IN 744
 - INTO_OUTFILE 751
 - JOIN 755
 - LEFT JOIN 755
 - LIKE 744
 - LIMIT 741, 742
 - LINES TERMINATED BY 752
 - MAX 748
 - MIN 748
 - ORDER BY 745
 - SUM 748
 - Tabellen verknüpfen 753
 - UNIX_TIMESTAMP 770, 785
 - WHERE 742
- select 232
 - accesskey 237
 - multiple 233
 - size 233
- Semantik 869
- semantisch korrekt 869
- Serendipity 980
- session_destroy 690, 964
- session_get_cookie_params 691
- session_id 688
- session_is_registered 690
- session_register 689
- session_set_cookie_params 691
- session_start 688, 959, 962
- session_unregister 690
- Sessions 687
 - Cookies 687
 - ID 687
- setcookie 666
- SGML 125, 865, 873
- shift 506, 526, 559
- SHOW DATABASES 786
- SHOW TABLES 787
- sht 585
- shtm 585
- shtml 585
- SimpleXML 717, 977
- simplexml_load_file 718
- sin 534, 657, 903
- Skalare 485
 - auflösen 486
- small 144
- SMIL 871
- SOAP 914
 - Envelope 915
- sort 527, 646
 - SORT_NUMERIC 646
 - SORT_REGULAR 646
 - SORT_STRING 646
- span 313, 352
- split 528, 541
- sqrt 535
- srand 534, 655
- SSH 59
- SSI 585
 - cgi 589
 - config 587
 - echo 585
 - timefmt 587
 - var 585
- SSL 926
- Statische IP-Adresse 116
- STDIN 544
- STH
 - err 772
 - errstr 772
 - execute 762
 - fetch 764

- fetchrow_array 764
- finish 762
- NAME 773
- NAME_LC 773
- NAME_UC 773
- NULLABLE 773
- NUM_OF_FIELDS 772
- PRECISION 773
- rows 763
- SCALE 773
- TYPE 772
- stop 890, 891
 - offset 890
 - stop-color 890
- Strafrecht 1065
- strftime 653
- strict 875
- Strings
 - charAt 444
 - indexOf 431, 442
 - length 441
 - split 446
 - substr 445
 - toLowerCase 431, 441, 442
 - toUpperCase 431, 442
- stripslashes 953
- strlen 642
- strong 147
- strpos 643
- strrpos 812
- strtolower 641
- strtoupper 641
- style
 - Attribut 265
 - Element 267
 - media 272
 - type 268
- sub 145, 499
- Subdomain 112
- Subroutinen 499
 - definieren 499
- substr 521, 644
- substr_count 643
- substr_replace 644

- Suchmaschinen 56
 - Kataloge 57
 - Roboter 57
- sup 145
- Suraski, Zeev 597
- SVG 197, 865, 879, 881, 882, 903
- SVG Viewer 881
- switch 390, 624
- Syntax-Highlighting 44

T

- Tabbed Browsing 63
- Tabellen 751
 - exportieren 751
 - importieren 752
- Tabellen als Designmittel 188
- Tabellenformatierung 331
- Tabellenverknüpfung 753
- table 166, 170
 - align 171
 - bgcolor 174
 - border 169
 - cellpadding 169
 - cellspacing 169
 - height 169
 - rules 187
 - width 169
- table-layout 331
 - auto 331
 - fixed 331
- Tags 128
- tbody 186
- TCP/IP 107
 - Aufbau und Struktur 108
 - Ports 113
 - Protokolle 114
- td 166
 - align 171
 - bgcolor 175
 - colspan 177
 - valign 172
- TDG 1064
- Teledienstegesetz 1064
- Telekommunikationsgesetz 1064
- Telnet 59

- Templates 933
- text 882, 884
 - fill 884
 - font-family 884
 - font-size 884
 - font-style 885
 - font-weight 885
 - stroke 885
 - stroke-width 885
 - style 884
 - x 884
 - y 884
- textarea 225, 226
 - accesskey 237
 - cols 226
 - name 226
 - rows 226
- text-decoration 288
 - blink 288
 - line-through 288
 - none 288
 - overline 288
 - underline 288
- Textformatierung 277
- Textpad 44
- text-transform 290
 - capitalize 290
 - lowercase 290
 - normal 290
 - uppercase 290
- tfoot 186
- th 166
 - align 171
 - bgcolor 175
 - valign 172
- thead 186
- this 410
- throw 713
- Thumbnails 810
- Tim Berners-Lee 35, 49
- time 529, 571, 651
- title 128
- TKG 1064
- Token 918
- top 323
- Top-Level-Domain 111
- tr 166
 - bgcolor 175
 - rowspan 180
- transitional 875
- trim 817
- try 713
- try..catch 471, 710
- tt 145
- txt 480

U

- u 144
- uc 519
- ul 158
 - circle 162
 - disc 162
 - square 162
 - type 162
- Umlaut-Domains 120
- undef 523
- undefined 434
- UNIQUE 737, 960
- Universalattribut
 - class 300
 - id 303
- unless 511
- unlink 684
- unset 609
- unshift 526
- UNSIGNED 736, 737
- UPDATE 750
 - SET 750
 - WHERE 750
- upload_max_filesize 668
- Urheberrecht 1061
- URI 111, 114
- USE 732
- use 481, 730
 - qw(:standard) 481
- UserLand Software 907

V

- Validator 877
- validieren 876
- values 491, 524
- var 147, 382

Variablen bezeichnen 605
VBScript 62, 375
vektorbasierte Grafiken 880
Vektoren 879
Verzeichnisschutz 936
vi 43
Voice over IP 118
VoIP 118
vorprüfende Schleifen 621

W

W3C 877
Wahrheitswerte 383
WAMP 71
 Apache installieren 71
 Apache konfigurieren 78
 MySQL installieren 74
 MySQL konfigurieren 75
 Perl installieren 80
 PHP installieren 76
 PhpMyAdmin installieren 81, 1079, 1080,
 1083, 1085, 1100, 1101, 1103, 1104, 1105,
 1106, 1107, 1108, 1109, 1110
 WinMySQLAdmin 75
Web
 dynamisch 53
 PlugIn 50
 statisch 50
Web Service Description Language 916
Weblogs 979
Webserver zu Hause 115
Webservices 913
Whois 59
widows 368
width 325
window 405, 406
 confirm 405
 setTimeout 420
with 409
WML 871
Wohlgeformtheit 869
word-spacing 287
WSDL 916
WYSIWYG 44

X

x 496
XAMPP 87
XHTML 873
Xitami 45
XML 865
 Strukturierung 867
xml
 lang 876
XML_OPTION_CASE_FOLDING 969
XML_OPTION_SKIP_WHITE 969
xml_parse_into_struct 969
xml_parser_create 968
xml_parser_free 970
xml_parser_set_option 969
XML-Parser 968
XML-Parser-Objekt 968
XML-Processing-Instruction 875, 882, 902
XML-RPC 913
xor 495

Y

Y2K-Bug 425
Yellow Pages 918

Z

Zeichenketten 383
Zend 597
ZEROFILL 737
z-index 323
Zugriffsmodi 679
Zugriffsschutz 705
Zwangstrennung 115