

Inhalt

Vorwort	15
1 Einleitung	17
1.1 Konzeption	21
1.2 Feedback	23
2 Grundpfeiler der Webentwicklung	25
2.1 Das weltweite Netz	26
2.2 Das HTTP-Protokoll	31
2.3 Hypertext Markup Language (HTML)	34
2.4 Anbieter und Anwender	36
2.4.1 Anbieter/Ressourceninhaber	37
2.4.2 Nutzer Ihres Systems	39
3 Installation	43
3.1 Microsoft Windows	45
3.1.1 Installation des XAMPP-Basispakets	45
3.1.2 Installation von XAMPP Lite	49
3.1.3 Starten und Beenden der Server	49
3.2 Linux	51
3.3 Konfiguration von XAMPP	52
3.3.1 Sicherheitslücken schließen	52
3.3.2 Konfigurationsdateien anpassen	55
3.4 Aktualisierung der Komponenten	61
4 Einführung in PHP	65
4.1 Strukturen einer PHP-Seite	66
4.2 Variablen	68
4.2.1 Grundlegende Syntax	68
4.2.2 Datentypen	70
4.2.3 Namenskonventionen	92
4.3 Konstanten	93
4.4 Kommentare	94
4.5 Funktionen	96

4.5.1	Syntax	98
4.5.2	Gültigkeitsbereiche	99
4.5.3	Namenskonventionen	101
4.5.4	Anonyme Funktionen	102
4.6	Kontrollkonstrukte	104
4.6.1	Bedingte Entscheidungen	104
4.6.2	Wiederholungen	113
4.6.3	Sprunganweisungen	122
4.6.4	Superglobale Arrays	124
4.6.5	Vordefinierte Konstanten	134
4.7	Einbinden externer Dateien	137

5 Objektorientierung in PHP 141

5.1	Die Modellierungssprache UML	141
5.2	Klassen und Objekte	143
5.2.1	Konstruktoren und Destruktoren	145
5.2.2	Zugriffsmodifizierer	146
5.2.3	Funktionen oder Methoden	150
5.2.4	Die Implementierung der Klasse Fahrzeug	151
5.2.5	Magische Methoden	152
5.3	Klassenbeziehungen	157
5.3.1	Vererbung	157
5.3.2	Klonen	163
5.4	Automatisches Laden von Klassen	165
5.5	Klassenattribute und -methoden überladen	166
5.6	Namensräume	171
5.7	Objektorientierte Fehlerbehandlung	176

6 Einführung in MySQL 181

6.1	Relationale Datenbanksysteme	186
6.2	MySQL und SQL	189
6.2.1	Eine Serververbindung aufbauen	190
6.2.2	Grundlegende SQL-Kommandos	197
6.2.3	Datentypen	216
6.3	Zugriffswerkzeuge	231
6.3.1	MySQL Administrator	231
6.3.2	MySQL Query Browser	233
6.3.3	MySQL Migration Toolkit	234
6.3.4	MySQL Workbench	236
6.3.5	phpMyAdmin	238

7	MySQLi	241
7.1	MySQLi in PHP einsetzen	241
7.2	MySQLi-Klassen	243
7.2.1	mysqli	243
7.2.2	mysqli_result	258
7.2.3	mysqli_stmt	268
8	Wichtige PHP-Funktionalitäten	277
8.1	Datums- und Zeitfunktionen	277
8.1.1	Erstellung eines Datums	278
8.1.2	Erstellung von Zeitstempeln	279
8.1.3	Mikrosekunden	280
8.1.4	Umgangssprachliche Zeitkalkulation	282
8.2	Datei- und Verzeichnisfunktionen	284
8.2.1	Auslesen und Schreiben von Dateien	284
8.2.2	Arbeiten mit Verzeichnissen	288
8.2.3	Prüfungen im Dateisystem	290
8.3	Reguläre Ausdrücke	291
8.3.1	Syntax	293
8.3.2	Reguläre Ausdrücke in PHP	299
8.3.3	Reguläre Ausdrücke in der Praxis	303
8.4	PEAR und PECL	306
8.4.1	PEAR	306
8.4.2	PECL	308
8.5	Datenabstraktion	309
8.5.1	Abstraktion im Kleinen: DBX	310
8.5.2	PDO	316
8.6	SQLite	335
8.7	Standard-PHP-Bibliothek	345
8.7.1	Iteratoren	346
8.7.2	Datenstrukturen	349
8.7.3	Exceptions	351
8.8	PHP-Archive	353
9	Fortgeschrittenes MySQL	367
9.1	Benutzerverwaltung	367
9.2	Kontrollfluss und Aggregationen	372
9.2.1	Bedingte Auswertung	372
9.2.2	Aggregationen	375

9.3	Performer Datenbankanfrage	380
9.3.1	JOIN-Syntax	381
9.3.2	Indizes	385
9.4	Metadaten	390
9.4.1	INFORMATION_SCHEMA	391
9.4.2	Metadatenanweisungen	396
9.4.3	PERFORMANCE_SCHEMA	397
9.5	Views	398
9.5.1	Anlegen	399
9.5.2	Editierbare und erweiterbare Sichten	403
9.5.3	Ändern und löschen	406
9.5.4	Ein praktisches Beispiel	406
9.6	Stored Procedures	408
9.6.1	Anlegen	409
9.6.2	Aufrufen	413
9.6.3	Ändern und Löschen	414
9.6.4	Variablen	414
9.6.5	Kontrollstrukturen	415
9.6.6	Error Handling	420
9.7	Trigger	426
9.7.1	Anlegen	426
9.7.2	Wozu aber sind Trigger notwendig?	429
9.7.3	Löschen	431
9.8	Partitionierung	432
9.8.1	Partitionstypen	436
9.8.2	Subpartitionierung	445
9.8.3	Verwaltung von Partitionen	447
9.9	Events	449
9.9.1	Anlegen	449
9.9.2	Ändern	456
9.9.3	Löschen	457

10 MySQL Storage Engines 459

10.1	MyISAM	461
10.2	InnoDB	465
10.2.1	Transaktionen	467
10.2.2	Referenzielle Integrität	470
10.3	Memory	473
10.4	Archive	475
10.5	Maria	477

10.6	CSV	478
10.7	MySQL Pluggable Storage Engines	481
10.8	MySQL Forks	483
10.8.1	Percona XtraDB	484
10.8.2	OurDelta	485
10.8.3	MariaDB	486
10.8.4	Drizzle	486

11 Sicherheit 489

11.1	Formulardaten und Validierung	491
11.2	Verschlüsselung	496
11.2.1	Ein-Weg-Verschlüsselung	496
11.2.2	Zwei-Wege-Verschlüsselung	500
11.2.3	SSL	504
11.3	Angriffsmethoden und Schutzmaßnahmen	505
11.3.1	Cross-Site-Scripting (XSS)	506
11.3.2	SQL Injection	509
11.3.3	Angriffe auf Sitzungen	510
11.3.4	HTTP Response Splitting	513
11.3.5	Fazit	514
11.4	Filter	514
11.4.1	Verfügbare Filter	515
11.4.2	Filterfunktionen	516
11.4.3	Datenvalidierung	525
11.4.4	Datenbereinigung	532

12 Datenbankentwurf 541

12.1	ERM	542
12.2	Normalisierung	546
12.2.1	Normalformen	546
12.2.2	Denormalisierung	551
12.3	Datenbankentwurf mit phpMyAdmin	552

13 Extensible Markup Language 555

13.1	XML-Grundlagen	555
13.2	XPath	557
13.3	SimpleXML	561
13.4	MySQL-XML-Funktionalität	573

14 Ein Basissystem mit PHP und MySQL 583

14.1	Konfigurationsdateien	584
14.1.1	common.php	584
14.1.2	paths.php	586
14.1.3	settings.php	587
14.1.4	includeAllClasses.php	589
14.2	Die Klasse für HTML	589
14.3	Datenbankverbindungen	592
14.3.1	Die Datenbankverbindungsklasse MySQL	593
14.3.2	SQLite-Verbindungsklasse	599
14.3.3	Abstraktionsschicht mittels PHP Data Objects	602
14.3.4	Verwendung der PDO-Klasse	607
14.4	Sicherheitsklasse	608
14.5	Logging	614
14.5.1	Die Logging-Klasse	614
14.5.2	Verwendung des Loggings	620
14.6	Debugging	621
14.6.1	Die Klasse DebugConsole	621
14.6.2	Hilfsdateien für die Debug-Konsole	630
14.6.3	Verwendung der Klasse DebugConsole	631

15 Sichere Webanwendungen 635

15.1	Benutzer authentifizieren	636
15.1.1	Klasse Login	636
15.1.2	Login-Klasse anwenden	640
15.2	Sitzungen mit der Datenbank verwalten	642
15.2.1	Die Klasse der Sitzungsverwaltung	642
15.2.2	Sitzungsverwaltung anwenden	649
15.2.3	Probleme mit der Sitzungsverwaltung	650
15.2.4	»Race Hazard« bei datenbankbasierter Sitzungsverwaltung	651
15.2.5	Benutzerstatus abfragen	652
15.2.6	Benutzer abmelden	654
15.3	Passwörter sicher gestalten	655
15.3.1	Passwortstrategie	655
15.3.2	Zufalls-Passwörter generieren	656
15.3.3	Passwort-Syntax überprüfen	658
15.4	Logging realisieren	660
15.4.1	Daten speichern	661

15.4.2	Klasse Log	662
15.4.3	Daten mittels JGraph darstellen	665
15.4.4	Klasse Chart	666
15.4.5	Daten als PDF-Dokument archivieren	669
15.4.6	Klasse PDFMaker	671
15.4.7	PDFMaker-Klasse anwenden	678
15.5	Einfache »Intrusion Detection« implementieren	679
15.5.1	Konfigurationsdatei für das »Intrusion Detection Login«	681
15.5.2	Klasse für »Intrusion Detection«	682
15.6	Sichere Formulare	688
15.7	Eigene Fehlerbehandlung einbauen	694
15.7.1	Konfigurationsdatei für Fehlerbehandlung	695
15.7.2	Fehlerbehandlungsklasse	697
15.7.3	Fehlerbehandlung in das Basissystem integrieren	702

16 Mehrbenutzersysteme 703

16.1	Das Hauptproblem: 2 Benutzer – 1 Datensatz	704
16.1.1	Szenario 1: Wer zuerst kommt ... ein Änderungsschlüssel	704
16.1.2	Szenario 2: Datensätze explizit sperren	704
16.2	Sperren von MySQL-Datensätzen	706
16.2.1	Die Klasse Locks	707
16.2.2	Beispielanwendung mit Sperren versehen	711
16.3	Transaktionen im praktischen Einsatz	715
16.3.1	Klasse Bank	717
16.3.2	Sichere und unsichere »Banktransaktionen« verwenden	720
16.4	Mehrsprachige Weboberflächen	723
16.4.1	Klasse LanguageSupport	724
16.4.2	Mehrsprachige Benutzeroberflächen realisieren	728
16.4.3	Erweiterungsmöglichkeiten	729
16.5	Mehrsprachige Weboberflächen mit gettext	732
16.5.1	Vorbereiten einer PHP-Datei	732
16.5.2	Dateiformat	734
16.5.3	Ordnerstruktur von »locale«	735
16.5.4	Klasse Gettext	737
16.5.5	Software für die Erstellung von Locales-Dateien	739

17 Web 2.0-Technologien	745
17.1 JavaScript Object Notation	745
17.2 Ajax	748
17.2.1 Beispiel: Blog-»Suchmaschine«	748
17.2.2 Klasse AJAX	749
17.2.3 AJAXJavaScript.js	757
17.2.4 PHP-Skripte für das Ajax-Beispiel	763
17.2.5 PHPLiveX im Einsatz	766
17.3 Web-API	775
17.3.1 Webservices	776
17.3.2 REST-Services	777
17.3.3 Interface für unsere eigene Web-API	780
17.3.4 REST-API	781
17.3.5 JSON-API	791
17.3.6 Verwendung und Aufruf der API-Klassen	799
17.3.7 Klasse Resolver	802
17.3.8 Klasse Request	805
17.3.9 index.php	811
17.4 jQuery	812
17.5 Web-Feeds	819
17.5.1 Technologie hinter Web-Feeds	819
17.5.2 Erstellen von Feed und Einträgen	821
17.5.3 RSS-Feed zur Verfügung stellen	833
17.5.4 Einsatz des RSS-Feeds	837
18 Blogs und Wikis	839
18.1 Blog	840
18.1.1 Klasse Blog	841
18.1.2 Blog in der praktischen Anwendung	852
18.2 Ein konkretes Mehrbenutzersystem: Wiki	857
18.2.1 Die Klasse Wiki	860
18.2.2 Wiki in der Praxis	873
19 Bildergalerien	877
19.1 Standard-Bildergalerie	877
19.1.1 Klassenübersicht: Bildergalerie	878
19.1.2 Klasse AbstractGallery	881
19.1.3 Klasse Admin	883

19.1.4	Klasse Gallery	904
19.1.5	Klasse Picture	907
19.2	Bildergalerie als Flash-Variante	910
19.2.1	Klasse FlashGallery	911
19.2.2	Klasse Picture erweitern	914

20 Dauerhafte Objektspeicherung 917

20.1	Persistenz	917
20.2	Umsetzung persistenter Objekte	917
20.2.1	Klasse Attribute	918
20.2.2	Klasse Object	920
20.3	Gültigkeitsprüfung von Parametern	937
20.3.1	Konfigurationsdatei der Gültigkeitsprüfung	937
20.3.2	Gültigkeitsprüfungsklasse	941
20.3.3	Gültigkeitsprüfung in die Klasse Object einbauen	956

21 Automatische Formularerstellung 959

21.1	Klasse SimpleAutomaticFormular	962
21.2	Automatische Formulargenerierung anwenden	978
21.3	Verbesserungsvorschläge	979

22 Model View Controller 981

22.1	Installation und Konfiguration von CakePHP	984
22.2	Prinzipien in CakePHP	986
22.3	MVC mit CakePHP umsetzen	987
22.3.1	HABTM-Modell	992
22.3.2	Validierung von Modell-Attributen	995
22.3.3	Methoden zum Controller hinzufügen	996
22.3.4	Views anpassen (selbst backen)	998
22.4	Ausblick	1001

Anhang..... 1003

A	PHP-Referenz	1005
B	MySQL-Referenz	1037
C	Open Source in der Praxis: Lizenzen	1059
C.1	GPL	1059
C.2	LGPL	1061

Inhalt

C.3	BSD	1062
C.4	PHP License	1063
C.5	MySQL-Lizenz	1063
C.6	Lizenzen im Überblick	1064
D	Glossar	1065
E	Inhalt der CD-ROM	1071
Index		1073