

Inhalt

Vorwort	15
1 Einleitung	17
1.1 Konzeption	20
1.2 Feedback	23
2 Grundpfeiler der Webentwicklung	25
2.1 Das weltweite Netz	26
2.2 Das HTTP-Protokoll	30
2.3 Hypertext Markup Language (HTML)	34
2.4 Anbieter und Anwender	36
2.4.1 Anbieter/Ressourceninhaber	36
2.4.2 Nutzer Ihres Systems	39
3 Installation	41
3.1 Microsoft Windows	43
3.1.1 Installation des XAMPP-Basispaketes	43
3.1.2 Installation von XAMPP Lite	46
3.1.3 Starten und Beenden der Server	47
3.2 Linux	48
3.3 Konfiguration von XAMPP	50
3.3.1 Sicherheitslücken schließen	50
3.3.2 Konfigurationsdateien anpassen	53
3.4 Aktualisierung der Komponenten	60
4 Einführung in PHP	63
4.1 Strukturen einer PHP-Seite	64
4.2 Variablen	66
4.2.1 Grundlegende Syntax	66
4.2.2 Datentypen	68
4.2.3 Namenskonventionen	90
4.3 Konstanten	91
4.4 Kommentare	92
4.5 Funktionen	94

4.5.1	Syntax	96
4.5.2	Gültigkeitsbereiche	97
4.5.3	Namenskonventionen	99
4.5.4	Anonyme Funktionen	99
4.6	Kontrollkonstrukte	101
4.6.1	Bedingte Entscheidungen	102
4.6.2	Wiederholungen	110
4.6.3	Sprunganweisungen	119
4.7	Vordefinierte Informationen	121
4.7.1	Superglobale Arrays	122
4.7.2	Vordefinierte Konstanten	131
4.8	Einbinden externer Dateien	134

5 Objektorientierung in PHP 139

5.1	Die Modellierungssprache UML	139
5.2	Klassen und Objekte	141
5.2.1	Konstruktoren und Destruktoren	143
5.2.2	Zugriffsmodifizierer	144
5.2.3	Funktionen oder Methoden	148
5.2.4	Die Implementierung der Klasse Fahrzeug	149
5.2.5	Magische Methoden	150
5.3	Klassenbeziehungen	155
5.3.1	Vererbung	155
5.3.2	Klonen	161
5.4	Automatisches Laden von Klassen	163
5.5	Klassenattribute und -methoden überladen	164
5.6	Namensräume	169
5.7	Objektorientierte Fehlerbehandlung	174

6 Einführung in MySQL 179

6.1	Relationale Datenbanksysteme	184
6.2	MySQL und SQL	187
6.2.1	Eine Server-Verbindung aufbauen	188
6.2.2	Grundlegende SQL-Kommandos	195
6.2.3	Datentypen	213
6.3	Zugriffswerkzeuge	228
6.3.1	MySQL Administrator	228
6.3.2	MySQL Query Browser	230
6.3.3	MySQL Migration Toolkit	232

6.3.4	MySQL Workbench	233
6.3.5	phpMyAdmin	233
7	MySQLi	237
7.1	MySQLi in PHP einsetzen	237
7.2	MySQLi-Klassen	239
7.2.1	mysqli	239
7.2.2	mysqli_result	254
7.2.3	mysqli_stmt	264
8	Wichtige PHP-Funktionalitäten	271
8.1	Datum- und Zeitfunktionen	271
8.1.1	Erstellung eines Datums	272
8.1.2	Erstellung von Zeitstempeln	273
8.1.3	Mikrosekunden	274
8.1.4	Umgangssprachliche Zeitkalkulation	276
8.2	Datei- und Verzeichnisfunktionen	278
8.2.1	Auslesen und Schreiben von Dateien	278
8.2.2	Arbeiten mit Verzeichnissen	282
8.2.3	Prüfungen im Dateisystem	284
8.3	Reguläre Ausdrücke	285
8.3.1	Syntax	287
8.3.2	Reguläre Ausdrücke in PHP	293
8.3.3	Reguläre Ausdrücke in der Praxis	296
8.4	PEAR und PECL	299
8.4.1	PEAR	300
8.4.2	PECL	301
8.5	Datenabstraktion	302
8.5.1	Abstraktion im Kleinen: DBX	303
8.5.2	PDO	309
8.6	SQLite	328
8.7	Standard-PHP-Bibliothek	338
8.7.1	Iteratoren	339
8.7.2	Datenstrukturen	342
8.7.3	Exceptions	343
8.8	PHP-Archive	345

9 Fortgeschrittenes MySQL 359

9.1	Benutzerverwaltung	359
9.2	Kontrollfluss und Aggregationen	364
9.2.1	Bedingte Auswertung	364
9.2.2	Aggregationen	366
9.3	Performerter Datenbankzugriff	372
9.3.1	JOIN-Syntax	372
9.3.2	Indizes	376
9.4	Metadaten	382
9.4.1	INFORMATION_SCHEMA	382
9.4.2	Metadaten-Anweisungen	387
9.5	Views	388
9.5.1	Anlegen	388
9.5.2	Editierbare und erweiterbare Sichten	392
9.5.3	Ändern und löschen	395
9.5.4	Ein praktisches Beispiel	395
9.6	Stored Procedures	397
9.6.1	Anlegen	398
9.6.2	Aufrufen	401
9.6.3	Ändern und Löschen	403
9.6.4	Variablen	403
9.6.5	Kontrollstrukturen	404
9.7	Trigger	409
9.7.1	Anlegen	409
9.7.2	Wozu sind Trigger aber notwendig?	412
9.7.3	Löschen	415
9.8	Partitionierung	415
9.8.1	Partitionstypen	419
9.8.2	Subpartitionierung	426
9.8.3	Verwaltung von Partitionen	428
9.9	Events	429
9.9.1	Anlegen	430
9.9.2	Ändern	436
9.9.3	Löschen	437

10 MySQL Storage Engines 439

10.1	MyISAM	441
10.2	InnoDB	445
10.2.1	Transaktionen	446
10.2.2	Referentielle Integrität	450

10.3	Memory	452
10.4	Archive	454
10.5	Maria	456
10.6	CSV	458
10.7	MySQL Pluggable Storage Engines	460

11 Sicherheit	463
----------------------------	------------

11.1	Formulardaten und Validierung	465
11.2	Verschlüsselung	470
11.2.1	Ein-Weg-Verschlüsselung	470
11.2.2	Zwei-Wege-Verschlüsselung	474
11.2.3	SSL	478
11.3	Angriffsmethoden und Schutzmaßnahmen	479
11.3.1	Cross-Site-Scripting (XSS)	480
11.3.2	SQL Injection	483
11.3.3	Angriffe auf Sitzungen	484
11.3.4	HTTP Response Splitting	486
11.3.5	Fazit	488
11.4	Filter	488
11.4.1	Verfügbare Filter	489
11.4.2	Filterfunktionen	490
11.4.3	Datenvalidierung	498
11.4.4	Datenbereinigung	506

12 Datenbankentwurf	513
----------------------------------	------------

12.1	ERM	514
12.2	Normalisierung	518
12.2.1	Normalformen	518
12.2.2	Denormalisierung	523

13 Extensible Markup Language (XML)	525
--	------------

13.1	XML-Grundlagen	525
13.2	XPath	527
13.3	SimpleXML	531
13.4	MySQL-XML-Funktionalität	542

14 Ein Basissystem mit PHP und MySQL 553

14.1 Konfigurationsdateien	554
14.1.1 common.php	554
14.1.2 paths.php	556
14.1.3 settings.php	557
14.1.4 includeAllClasses.php	559
14.2 Die Klasse für HTML	559
14.3 Datenbankverbindungen	562
14.3.1 Die Datenbankverbindungsklasse MySQL	563
14.3.2 SQLite-Verbindungsklasse	569
14.3.3 Abstraktionsschicht mittels PHP Data Objects	572
14.3.4 Verwendung der PDO-Klasse	577
14.4 Sicherheitsklasse	578
14.5 Logging	584
14.5.1 Die Logging-Klasse	584
14.5.2 Verwendung des Loggings	589
14.6 Debugging	590
14.6.1 Die Klasse DebugConsole	591
14.6.2 Hilfsdateien für die Debug-Konsole	600
14.6.3 Verwendung der Klasse DebugConsole	601

15 Sichere Webanwendungen 603

15.1 Benutzer authentifizieren	604
15.1.1 Klasse Login	604
15.1.2 Login-Klasse anwenden	608
15.2 Sitzungen mit der Datenbank verwalten	610
15.2.1 Die Klasse der Sitzungsverwaltung	610
15.2.2 Sitzungsverwaltung anwenden	617
15.2.3 Probleme mit der Sitzungsverwaltung	618
15.2.4 »Race Hazard« bei datenbankbasierter Sitzungsverwaltung	619
15.2.5 Benutzerstatus abfragen	620
15.2.6 Benutzer abmelden	622
15.3 Passwörter sicher gestalten	623
15.3.1 Passwortstrategie	623
15.3.2 Zufalls-Passwörter generieren	624
15.3.3 Passwort-Syntax überprüfen	626
15.4 Logging realisieren	628
15.4.1 Daten speichern	629

15.4.2 Klasse Log	630
15.4.3 Daten mittels JGraph darstellen	633
15.4.4 Klasse Chart	634
15.4.5 Daten als PDF-Dokument archivieren	637
15.4.6 Klasse PDFMaker	639
15.4.7 PDFMaker-Klasse anwenden	645
15.5 Einfache »Intrusion Detection« implementieren	647
15.5.1 Konfigurationsdatei für das »Intrusion Detection Login«	649
15.5.2 Klasse für »Intrusion Detection«	650
15.6 Sichere Formulare	656
15.7 Eigene Fehlerbehandlung einbauen	661
15.7.1 Konfigurationsdatei für Fehlerbehandlung	663
15.7.2 Fehlerbehandlungsklasse	664
15.7.3 Fehlerbehandlung in das Basissystem integrieren	670

16 Mehrbenutzersysteme	671
-------------------------------------	------------

16.1 Das Hauptproblem: 2 Benutzer – 1 Datensatz	672
16.1.1 Szenario 1: Wer zuerst kommt ... ein Änderungsschlüssel	672
16.1.2 Szenario 2: Datensätze explizit sperren	672
16.2 Sperren von MySQL-Datensätzen	673
16.2.1 Die Klasse Locks	674
16.2.2 Beispianwendung mit Sperren versehen	679
16.3 Transaktionen im praktischen Einsatz	682
16.3.1 Klasse Bank	684
16.3.2 Sichere und unsichere »Banktransaktionen« verwenden	688
16.4 Mehrsprachige Weboberflächen	689
16.4.1 Klasse LanguageSupport	691
16.4.2 Mehrsprachige Benutzeroberflächen realisieren	695
16.4.3 Erweiterungsmöglichkeiten	696
16.5 Mehrsprachige Weboberflächen mit gettext	699
16.5.1 Vorbereiten einer PHP-Datei	700
16.5.2 Dateiformat	701
16.5.3 Ordnerstruktur von »locale«	703
16.5.4 Klasse Gettext	704
16.5.5 Software für die Erstellung von Locales-Dateien	706

17 Web 2.0-Technologien 713

17.1	JavaScript Object Notation	713
17.2	Ajax	716
17.2.1	Beispiel: Blog->Suchmaschine<	716
17.2.2	Klasse AJAX	717
17.2.3	AJAXJavaScript.js	725
17.2.4	PHP-Skripte für das Ajax-Beispiel	731
17.2.5	PHPLiveX im Einsatz	734
17.3	Web-API	743
17.3.1	Webservices	744
17.3.2	REST-Services	745
17.3.3	Interface für unsere eigene Web-API	747
17.3.4	REST-API	748
17.3.5	JSON-API	758
17.3.6	Verwendung und Aufruf der API-Klassen	766
17.3.7	Klasse Resolver	769
17.3.8	Klasse Request	772
17.3.9	index.php	777
17.4	jQuery	779
17.5	Web-Feeds	785
17.5.1	Technologie hinter Web-Feeds	786
17.5.2	Erstellen von Feed und Einträgen	787
17.5.3	RSS-Feed zur Verfügung stellen	799
17.5.4	Einsatz des RSS-Feeds	803

18 Blogs und Wikis 805

18.1	Blog	806
18.1.1	Klasse Blog	807
18.1.2	Blog in der praktischen Anwendung	818
18.2	Ein konkretes Mehrbenutzersystem: Wiki	823
18.2.1	Die Klasse Wiki	825
18.2.2	Wiki in der Praxis	839

19 Bildergalerien 843

19.1	Standard-Bildergalerie	843
19.1.1	Klassenübersicht: Bildergalerie	843
19.1.2	Klasse AbstractGallery	847
19.1.3	Klasse Admin	848

19.1.4 Klasse Gallery	870
19.1.5 Klasse Picture	873
19.2 Bildergalerie als Flash-Variante	876
19.2.1 Klasse FlashGallery	877
19.2.2 Klasse Picture erweitern	880
20 Dauerhafte Objektspeicherung	883
20.1 Persistenz	883
20.2 Umsetzung persistenter Objekte	883
20.2.1 Klasse Attribute	884
20.2.2 Klasse Object	886
20.3 Gültigkeitsprüfung von Parametern	903
20.3.1 Konfigurationsdatei der Gültigkeitsprüfung	903
20.3.2 Gültigkeitsprüfungsklasse	907
20.3.3 Gültigkeitsprüfung in die Klasse Object einbauen	921
21 Automatische Formularerstellung	923
21.1 Klasse SimpleAutomaticFormular	926
21.2 Automatische Formulargenerierung anwenden	942
21.3 Verbesserungsvorschläge	943
22 Model View Controller	945
22.1 Installation und Konfiguration von CakePHP	948
22.2 Prinzipien in CakePHP	950
22.3 MVC mit CakePHP umsetzen	951
22.3.1 HABTM-Modell	956
22.3.2 Validierung von Modell-Attributen	959
22.3.3 Methoden zum Controller hinzufügen	960
22.3.4 Views anpassen (selbst backen)	962
22.4 Ausblick	965
Anhang	967
A PHP-Referenz	969
B MySQL-Referenz	1003
C Open Source in der Praxis: Lizenzen	1025
C.1 GPL	1025
C.2 LGPL	1027

C.3	BSD	1028
C.4	PHP License	1029
C.5	MySQL-Lizenz	1029
C.6	Lizenzen im Überblick	1030
D	Glossar	1031
E	Inhalt der CD-ROM	1037
	Index	1039