



Peter Lavin

PHP objektorientiert

Konzepte, Techniken, Code

HANSER

Inhalt.....	5
Einführung.....	15
Was kann Ihnen dieses Buch bieten?.....	15
Für wen ist dieses Buch Pflichtlektüre?.....	16
Anforderungen.....	16
Software.....	16
Wissen.....	17
Inhaltsübersicht.....	17
Webseite zum Buch.....	20
Quellen.....	20
Quellen im Internet.....	20
Literatur.....	21
1 Verwirrendes WWW.....	23
1.1 Brauchen wir wirklich Objekte?.....	23
Nur eine Skriptsprache.....	24
Objektorientierung ist eine Sache für große Softwarehersteller.....	25
Lassen wir es bei „gut genug“ bleiben.....	25
Erhöhte Komplexität.....	26
1.2 PHP Kultur.....	26

2	Grundlagen objektorientierter Programmierung.....	28
2.1	Klassen.....	28
	Vergleich zwischen Klassen und Datensätzen.....	28
	Konsequente Modelle.....	29
	Objekte sind Instanzen.....<.....	29
2.2	Objekte benötigen Zugriffsmodifikatoren.....	29
2.3	Objekt-Wiederverwendung und Vererbung.....	30
	Mehrfachvererbung.....	31
	Gewinner auf allen Ebenen.....	31
2.4	Wie es weiter geht.....	32
3	Die neuen, objektorientierten Features in PHP 5.....	33
3.1	Zugriffsmodifikatoren.....	34
3.2	Eingebundene Klassen.....	34
	Ausnahmen.....	35
	Datenbankklassen.....	35
	Webdienste.....	35
	Reflection-Klassen.....	36
	Iterator.....	36
3.3	Abwärtskompatibilität.....	36
	Pass By Reference.....	37
	Prognose.....	37
3.4	Was Sie jetzt tun können.....	38
	Einsatz von PHP 5.....	39
	Kompromiss.....	39
4	Eine kleine Klasse.....	41
4.1	Entwurf.....	42
	Definition des Problems.....	42
4.2	Kein Da Vinci Code.....	43
	Der Konstruktor.....	44

Instanzvariablen referenzieren.....	44
Wrapper-Methoden.....	45
4.3 Erstellung einer Instanz.....	46
4.4 Was haben Sie erreicht?.....	48
4.5 Aber wird es laufen?.....	48
5 Die Klasse anpassen.....	51
5.1 Auf PHP 5 aktualisieren.....	51
Zugriffsmodifikatoren.....	51
Der Konstruktor.....	54
5.2 Änderungen an Ihrer Klasse.....	55
Umschreiben des Konstruktors.....	55
Daten filtern.....	58
Das Array zurücksetzen.....	59
5.3 Zusammenfassung der Änderungen.....	60
6 Die Miniaturbild-Klasse.....	63
6.1 Was macht ein Webdesigner?.....	63
6.2 Die Klasse ThumbnailImage.....	65
Datenmitglieder.....	65
Dekonstruktion des Konstruktors.....	66
Zwei Möglichkeiten zum Konstruieren eines Objekts.....	67
Internes Verhalten - Private Methoden.....	67
6.3 Anzeige als Vorschaubild.....	72
6.4 Das Ganze zusammenbauen.....	73
6.5 Wie es weiter geht.....	74
7 Aufbau der PageNavigator-Klasse.....	75
7.1 Wie wird sich der Navigator verhalten?.....	75
7.2 Wie wird sie aussehen?.....	76

Inhalt

7.3	Code.....	77
	Kein falsches Verhalten zulassen.....	80
	Sonstige Aufrufmethoden des Konstruktors.....	80
7.4	Die Methode getNavigator.....	82
	Hauptprogramm des Navigators.....	83
	Move Next und Move Last.....	84
	Aktuelle Seite und Gesamtseitenanzahl.....	84
7.5	Wie es weitergeht.....	85
8	Verwendung der PageNavigator-Klasse.....	87
8.1	Änderung an der Klasse DirectoryItems.....	87
8.2	CSS und Wiederverwendbarkeit.....	88
8.3	Seitenweise navigieren.....	90
	Teil eines Arrays anzeigen.....	91
	Das Seitennavigations-Objekt erstellen.....	92
8.4	Wie es weiter geht.....	93
9	Die Datenbankklasse.....	95
9.1	Ihr Wissen verwenden.....	95
9.2	Zwei Klumpen oder Einer?.....	96
9.3	Die MySQLConnect-Klasse.....	96
	Eine klassenbewusste Variable.....	97
	Weitere Verbindungen erstellen.....	98
	Sie können nur von dort nach hier gelangen.....	98
9.4	Die Klasse MySQLResultSet.....	99
9.5	Die seitenweise Navigation verwenden.....	100
	Sortieren, Filtern, Auswählen.....	102
	Durch die Ergebnisse laufen.....	102
	Der Navigator braucht eine Richtung.....	103
9.6	Wie es weiter geht mit dem Navigator.....	104

10	Verbesserungen durch Vererbung.....	105
10.1	Die PHP Standard Bibliothek.....	106
10.2	Eine Klasse durch Vererbung erweitern.....	106
	Die Klasse Exception.....	106
	protected.....	107
	final.....	108
	Noch mehr magische Methoden.....	108
	__toString.....	108
	__clone.....	109
10.3	Fehler durch Ausnahmen ersetzen.....	109
10.4	Die Klasse MySQLException.....	110
	Methoden überschreiben.....	111
10.5	Die Klasse MySQLConnect anpassen.....	112
	Die Klasse in Aktion versetzen.....	113
10.6	Ausnahmen abfangen.....	114
10.7	Eine Schnittstelle implementieren.....	115
	Etwas über die Iterator-Schnittstelle lernen.....	116
	Implementierung.....	117
	Eine Methode undefiniert belassen.....	120
	Implementierung und Zugriff.....	120
	Durch MySQLResultSet iterieren.....	121
10.8	Wie es weitergeht.....	121
11	Fortgeschrittene Konzepte objektorientierter Programmierung.....	123
11.1	Abstrakte Klassen.....	123
	Private Methoden können nicht abstrakt sein.....	124
	Schnittstelle oder reine abstrakte Klasse?.....	124
11.2	Polymorphismus.....	125
	Steuern, wie Funktionen eingesetzt werden.....	125
11.3	Statische Klassen.....	126
	11.3.1 Statische Math-Klassen.....	127

Inhalt

11.3.2	Instanzen statischer Klassen	128
11.3.3	Instanzierung einer statischen Klasse verhindern	128
11.4	Entwurfsmuster	129
	Singleton-Muster	129
	Welche Implementierung?	130
11.5	Wie geht es weiter?	131
12	Frisch halten leicht gemacht	133
12.1	SimpleXML	134
	XML	134
	RSS	135
12.2	Struktur einer RSS-Datei	135
	DenFeed lesen	136
12.3	Sitespezifische Suche	137
	Die Google-API	138
	AJAX	138
	SOAP installieren	138
	Die SOAP-Erweiterung	139
	Ein SOAP-Client	139
12.4	Funktionalität testen	142
	Ergebnisse mit AJAX anzeigen	143
12.5	Komplexe Aufgaben leicht gemacht	144
12.6	Wollen Sie das alles prozedural machen?	145
13	Noch mehr Zauberei	147
13.1	<code>_get</code> und <code>_set</code>	148
	Ist das den Aufwand wert?	149
13.2	<code>__isset</code> und <code>__unset</code>	150
13.3	<code>_call</code>	151
13.4	<code>autoload</code>	152

13.5	__sleep und __wakeup.....!	153
13.6	_clone.....	153
	Und wo ist Waldo?.....	154
	clone.....	155
	Aggregationsklassen.....!	156
	Eine Get-Methode für Objektdatenmitglieder einer Aggregationsklasse.....	158
13.7	Klonen verboten.....	159
13.8	Anmerkung zum Überladen.....	159
14	Dokumentation mit Reflection-Klassen erstellen.....	161
14.1	Was sind Reflection-Klassen?.....	162
14.2	Die Reflection-Klassengruppe.....	162
	Die Klasse Reflection.....	163
	Die Klasse ReflectionClass.....	164
	ReflectionMethod und ReflectionParameter.....	165
	Integrierte Funktionen.....	166
14.3	Welches Format wünschen Sie?.....	166
14.4	Die Klasse Documenter.....	166
	Beschreiben der Klasse Documenter.....	167
	Methoden und Datenmitglieder beschreiben.....	168
	Der Konstruktor.....	169
	Methoden- und Datenmitgliedsmodifizierer.....	169
14.5	Die Klasse Documenter verwenden.....	170
	Seitenleiste mit Klassen und Schnittstellen erstellen.....	170
	Detaillierte Dokumentation formatieren.....	171
	Kommentare für Documenter formatieren.....	173
	Methoden dokumentieren.....	173
14.6	Reflektieren über die Reflexion.....	174
15	SQLite erweitern.....	177
15.1	Überblick.....	177

Inhalt

15.2	Verzeichnisstruktur.....;	179
15.3	Wie es funktioniert.....	179
15.4	Erste Schritte.....	180
15.5	Eine Tabelle erstellen.....	181
15.6	Views.....	182
15.7	Trigger.....	183
15.8	Die PHP-Implementierung von SQLite.....	184
15.9	SQLiteDatabase erweitern.....	184
15.10	Abfragemethoden außer Kraft setzen.....	185
	Fehlermeldungen.....	186
	Abfragemethoden.....	187
15.11	Hilfsmethoden.....	190
	Metadaten holen.....	193
	Metadaten verwenden.....	193
15.12	Benutzerdefinierte Funktionen.....	195
15.13	Anwendungsmöglichkeiten und Beschränkungen von SQLite.....	197
16	PDO verwenden.....	199
16.1	Pro und Kontra.....	199
16.2	Konvertierung der SQLite-Anwendung.....	200
	Änderungen am Code.....	200
16.3	Weitere Funktionalitäten von PDO.....	204
	Die Klasse PDO.....	204
	PDOStatement.....	204
16.4	Einschätzung.....	207
	Der Stein der Weisen?.....	208
	Anhang A - PHP Einrichten.....	209
	php.ini-Einstellungen.....	210

E_STRICT.....211

Escapen Sie nicht zweimal.....211

Anhang B - Konvertierungstabelle PHP 4 nach PHP 5.....213

Register.....217