

# Stručný obsah

<b>1. Co je nového v PHP 5?</b>	<b>33</b>
<b>2. Základy jazyka PHP 5</b>	<b>45</b>
<b>3. PHP 5 – objektově orientovaný jazyk</b>	<b>81</b>
<b>4. Pokročilé objektové programování a návrhové vzory v PHP 5</b>	<b>107</b>
<b>5. Jak v PHP naprogramovat webovou aplikaci</b>	<b>131</b>
<b>6. Databáze v PHP 5</b>	<b>165</b>
<b>7. Ošetřování výjimek</b>	<b>207</b>
<b>8. Použití XML v PHP 5</b>	<b>233</b>
<b>9. Tradiční doplňky</b>	<b>271</b>
<b>10. Pracujeme s repositářem PEAR</b>	<b>347</b>
<b>11. Důležité balíčky v repositáři PEAR</b>	<b>385</b>
<b>12. Tvorba komponent pro PEAR</b>	<b>403</b>
<b>13. Upgrade na PHP 5</b>	<b>431</b>
<b>14. Výkon</b>	<b>445</b>
<b>15. Úvod do tvorby vlastních doplňků</b>	<b>477</b>
<b>16. Skriptování z příkazového řádku</b>	<b>499</b>
<b>A. Přehled balíčků PEAR a PECL</b>	<b>519</b>
<b>B. Příručka nástroje phpDocumentor</b>	<b>605</b>
<b>C. Začínáme s aplikací Zend Studio</b>	<b>633</b>
<b>Rejstřík</b>	<b>645</b>



# Obsah

<b>Předmluva</b>	<b>25</b>
<b>Úvod</b>	<b>27</b>
<b>Kapitola 1</b>	
<b>Co je nového v PHP 5?</b>	<b>33</b>
<b>Úvod</b>	<b>33</b>
<b>Jazykové funkce</b>	<b>33</b>
Nový objektově orientovaný model	33
Nové objektově orientované funkce	35
Další nové funkce	39
<b>Všeobecné změny v PHP</b>	<b>40</b>
Jazyk XML a webové služby	40
Základy	40
SAX	40
DOM	40
XSLT	41
SimpleXML	41
SOAP	42
Nový doplněk MySQLi (MySQL Improved)	42
Doplněk SQLite	42
Doplněk Tidy	42
Doplněk Perl	43
<b>Další nové funkce v PHP 5</b>	<b>43</b>
Nový správce paměti	43
Ukončení podpory systému Windows 95	43
<b>Shrnutí</b>	<b>43</b>
<b>Kapitola 2</b>	
<b>Základy jazyka PHP 5</b>	<b>45</b>
<b>Úvod</b>	<b>45</b>
<b>Jak vkládat kód v PHP do stránek HTML</b>	<b>46</b>
<b>Komentáře</b>	<b>46</b>
<b>Proměnné</b>	<b>47</b>
Nepřímé odkazy na proměnné (proměnné s proměnnými názvy)	48
Správa proměnných	48
Funkce isset()	48

Funkce unset()	49
Funkce empty()	49
Superglobální proměnné	50
<b>Základní datové typy</b>	<b>50</b>
Celá čísla	50
Reálná čísla (čísla s plovoucí čárkou)	51
Řetězce	51
Uvozovky	51
Apostrofy	52
Syntaxe Here-doc	53
Jak použít indexy řetězců	53
Logické hodnoty	53
Prázdné hodnoty (null)	54
Prostředky (resources)	55
Pole (array)	55
Funkce array()	55
Práce s prvky pole	56
Úpravy a tvorba nových prvků pole	56
Čtení hodnot polí	57
Práce s vnořenými poli (nebo s vícerozměrnými poli)	57
Procházení polí pomocí příkazu foreach	57
Procházení polí pomocí funkcí list() a each()	59
Funkce reset()	59
Funkce each()	59
Funkce list()	60
Další metody pro procházení polí	60
Konstanty	60
<b>Operátory</b>	<b>61</b>
Binární operátory	62
Číselné operátory	62
Operátor spojení (.)	62
Operátory přiřazení	62
Operátor přiřazení odkazem	63
Operátory porovnání	63
Logické operátory	64
Zkrácené vyhodnocování	64
Bitové operátory	64
Unární operátory	65
Operátory negace	65
Inkrementační a dekrementační operátory	65
Inkrementace řetězců	66
Operátory přetypování	67
Operátor ticha	67
Jediný ternární operátor	67
<b>Řídicí struktury</b>	<b>68</b>
Podmínkové řídicí struktury	68
Příkazy if	68
Příkazy switch	69

Cyklické řídicí struktury	70
Cyklus while	70
Řízení cyklů: příkazy break a continue	71
Cykly do...while	71
Cykly for	72
Řídicí struktury pro zahrnování kódu	73
Příkazy include a přátelé	73
Funkce eval()	75
<b>Funkce</b>	<b>75</b>
Uživatelsky definované funkce	76
Kontext funkce	76
Vracení výsledků hodnotou	77
Vracení výsledků odkazem	77
Deklarace argumentů funkce	78
Argumenty předávané hodnotou	78
Argumenty předávané odkazem	78
Implicitní argumenty	79
Statické proměnné	79
<b>Shrnutí</b>	<b>80</b>

## Kapitola 3

# PHP 5 – objektově orientovaný jazyk **81**

---

<b>Úvod</b>	<b>81</b>
<b>Objekty</b>	<b>82</b>
<b>Deklarace třídy</b>	<b>83</b>
<b>Tvorba objektů</b>	<b>83</b>
<b>Destruktory</b>	<b>84</b>
<b>Metody a vlastnosti</b>	<b>85</b>
Veřejné, chráněné a soukromé vlastnosti	86
Veřejné, chráněné a soukromé metody	87
Statické vlastnosti	88
Statické metody	89
<b>Konstanty třídy</b>	<b>90</b>
Duplikace objektů	91
<b>Polymorfismus</b>	<b>92</b>
<b>Identifikátory parent:: a self::</b>	<b>95</b>
<b>Operátor instanceof</b>	<b>95</b>
<b>Abstraktní metody a třídy</b>	<b>96</b>
<b>Rozhraní</b>	<b>98</b>
<b>Dědičnost rozhraní</b>	<b>99</b>
<b>Konečné metody</b>	<b>100</b>
<b>Konečné třídy</b>	<b>100</b>
<b>Metoda __toString()</b>	<b>100</b>
<b>Ošetření výjimek</b>	<b>101</b>
<b>Funkce __autoload()</b>	<b>104</b>

<b>Definice typů v argumentech funkcí</b>	<b>106</b>
<b>Shrnutí</b>	<b>106</b>

## Kapitola 4

<b>Pokročilé objektové programování a návrhové vzory v PHP 5</b>	<b>107</b>
--	------------

<b>Úvod</b>	<b>107</b>
<b>Možnosti přetěžování</b>	<b>107</b>
Přetěžování vlastností a metod	107
Přetěžování syntaxe pro přístup k poli	110
<b>Iterátory</b>	<b>111</b>
<b>Návrhové vzory</b>	<b>115</b>
Vzor strategie	116
Vzor singleton	118
Vzor výrobní třídy	119
Vzor pozorovatel	122
<b>Reflexe</b>	<b>124</b>
Úvod	124
Aplikační rozhraní pro reflexi	124
Příklady reflexe	126
Jednoduchý příklad	126
Implementace vzoru delegování pomocí reflexe	128
<b>Shrnutí</b>	<b>129</b>

## Kapitola 5

<b>Jak v PHP naprogramovat webovou aplikaci</b>	<b>131</b>
---	------------

<b>Úvod</b>	<b>131</b>
<b>Kód PHP v souborech HTML</b>	<b>132</b>
<b>Uživatelský vstup</b>	<b>134</b>
<b>Bezpečný uživatelský vstup</b>	<b>136</b>
Časté chyby	136
Globální proměnné	136
Skriptování mezi servery	137
Vkládání kódu SQL	138
<b>Techniky „zabezpečení“ skriptů</b>	<b>139</b>
Ověřování zadaných informací	139
Ověřování HMAC	141
Třída PEAR::Crypt_HMAC	142
Vstupní filtr	145
Práce s hesly	145
Ošetření chyb	147
<b>Soubory cookie</b>	<b>148</b>

<b>Uživatelské relace</b>	<b>152</b>
<b>Odesílání souborů</b>	<b>155</b>
Jak zpracovat přijatý soubor	155
<b>Architektura</b>	<b>159</b>
Jeden skript obslouží všechna volání	160
Jeden skript pro každou funkci	161
Oddělení logiky od rozvržení	161
<b>Shrnutí</b>	<b>162</b>

## Kapitola 6

# Databáze v PHP 5 165

---

<b>Úvod</b>	<b>165</b>
<b>MySQL</b>	<b>166</b>
Síla a slabiny MySQL	166
Přednost: Velká obliba na trhu	166
Přednost: Snadný začátek	166
Přednost: Licence otevřeného zdrojového kódu téměř pro všechny uživatele	166
Přednost: Rychlost	167
Přednost: Přiměřená možnost růstu	167
Rozhraní PHP	167
Data příkladů	167
Datová spojení	167
Dotazy s vyrovnávací pamětí a dotazy bez ní	170
Databázové dotazy	170
Vícepříkazové dotazy	171
Režimy čtení řádků	172
Připravené příkazy	172
Vázání proměnných	172
Práce s binárními daty (BLOB)	174
Ukládání binárních dat do databáze	174
Načítání binárních dat z databáze	175
<b>SQLite</b>	<b>176</b>
Přednosti a nedostatky databázového stroje SQLite	176
Výhoda: samostatnost a nezávislost, obejdete se bez serveru	176
Výhoda: snadné začátky	176
Výhoda: je součástí PHP 5	176
Výhoda: jednoduchost a rychlost	176
Výhoda: procedurální a objektové rozhraní	177
Nevýhoda: Neexistence serverového procesu	177
Nevýhoda: Soubory nejsou binárně bezpečné	177
Nevýhoda: Transakce zamykají soubory	177
Nejlepší možnosti využití	177
Rozhraní PHP	177
Nastavení databází	178
Jednoduché dotazy	178
Ošetření chyb	179
Jednodušší dotazy a transakce	180

Spouštěcí procedury	181
Uživatelsky definované funkce (UDF)	181
Další funkce pro spouštění dotazů	183
Načítání dat	184
Iterátory	185
Vlastní způsoby procházení	186
Další funkce pro práci se záznamy ve výsledných sadách	188
Agregační uživatelsky definované funkce	188
Kódování znaků	189
Vyladění	189
Další triky	190
Moudrá slova	191
<b>PEAR DB</b>	<b>192</b>
Jak získat balíček PEAR DB	192
Klady a zápory databázové abstrakce	192
Jaké funkce jsou abstrahovány?	193
Databázová spojení	193
Zdroje dat (DSN)	193
Navázání spojení	194
Konfigurační možnosti	195
Spouštění dotazů	195
Metoda query()	195
Metoda limitQuery()	196
Metody prepare() a execute()	197
Metoda simpleQuery()	198
Čtení řádků ve výsledcích	198
Režimy čtení řádků	198
Konfigurace režimu čtení řádků	199
Metoda fetchRow()	199
Metoda fetchInto()	199
Použijte vlastní třídu výsledku	199
Posloupnosti	200
Přenosnost	201
Počet odstraněných řádků (DB_PORTABILITY_DELETE_COUNT)	202
Počet řádků (DB_PORTABILITY_NUMROWS)	202
Převod na malá písmena (DB_PORTABILITY_LOWERCASE)	202
Oseknutí koncových mezer (DB_PORTABILITY_RTRIM)	202
Ošetření prázdných řetězců (DB_PORTABILITY_NULL_TO_EMPTY)	202
Skutečné chyby přenosnosti (DB_PORTABILITY_ERRORS)	202
Abstrahované chyby	202
Chybové kódy balíčku DB	203
Elegantní ošetření chyb	204
Pomocné metody	204
Metoda getOnce()	204
Metoda getRow()	205
Metoda getCol()	205
Metoda getAssoc()	205
Metoda getAll()	206



## Kapitola 7

**Ošetřování výjimek****207**

<b>Úvod</b>	<b>207</b>
<b>Typy chyb</b>	<b>208</b>
Programátorské chyby	208
Syntaktické chyby	208
Funkce eval()	208
Klíčová slova include a require	209
Nedefinované symboly	209
Proměnné a konstanty	210
Indexy pole	211
Funkce a třídy	211
Logické chyby	212
Chyby přenosnosti	212
Rozdíly v operačních systémech	212
Konfigurační rozdíly	212
Direktiva register_globals	213
Direktiva register_argc_argv	213
Direktivy magic_quotes_gpc a magic_quotes_runtime	213
Direktiva y2k_compliance	213
Direktiva unserialize_callback_func	213
Direktiva arg_separator.input	213
Direktiva allow_url_fopen	213
Rozdíly SAPI	214
Jak se vypořádat s přenosností	215
Nástroje pro zlepšení přenosnosti	215
Chyby při běhu programu	216
Chyby způsobené funkcemi jazyka PHP	216
Úrovně oznamování chyb	216
Oznamování chyb	217
Vlastní rutiny pro ošetření chyb	218
Umlčení chyb	219
<b>Chyby rozhraní PEAR</b>	<b>220</b>
Zachycování chyb	221
Metoda PEAR::isError()	221
Způsobování chyb	221
Metoda PEAR::throwError()	223
Metoda PEAR::raiseError()	223
Třída PEAR_Error	224
Konstruktor třídy PEAR_Error	224
Metoda PEAR_Error::addUserInfo()	225
Metoda PEAR_Error::getBacktrace([ rámec])	225
Metoda PEAR_Error::getCallback()	225
Metoda PEAR_Error::getCode()	225
Metoda PEAR_Error::getMessage()	225
Metoda PEAR_Error::getMode()	225
Metoda PEAR_Error::getType()	226

Metoda PEAR_Error::getUserInfo()	226
Jak ošetřit chyby typu PEAR	226
Metoda PEAR::setErrorHandling()	226
Režimy ošetření chyb typu PEAR	227
PEAR_ERROR_RETURN	227
PEAR_ERROR_PRINT	227
PEAR_ERROR_DIE	227
PEAR_ERROR_TRIGGER	227
PEAR_ERROR_CALLBACK	227
Elegantní způsob ošetření chyb	227
Metoda PEAR::pushErrorHandling()	227
Metoda PEAR::popErrorHandling()	228
Metoda PEAR::expectError()	228
Metoda PEAR::popExpect()	229
Metoda PEAR::delExpect()	230
<b>Výjimky</b>	<b>230</b>
Co jsou to výjimky?	230
Konstrukce try...catch...throw	230
<b>Shrnutí</b>	<b>232</b>

## Kapitola 8

### Použití XML v PHP 5

**233**


---

<b>Úvod</b>	<b>233</b>
<b>Slovníček pojmů</b>	<b>234</b>
<b>Zpracování dokumentu XML</b>	<b>236</b>
SAX	236
DOM	239
Pracujeme s jazykem XPath	242
Jak vytvořit strom modelu DOM	242
<b>SimpleXML</b>	<b>244</b>
Tvorba objektu typu SimpleXML	244
Procházíme objekty typu SimpleXML	245
Jak se objekty typu SimpleXML ukládají	246
<b>PEAR</b>	<b>246</b>
Třída XML_Tree	247
XML_RSS	248
<b>Převody souborů XML</b>	<b>251</b>
XSLT	251
<b>Komunikace prostřednictvím XML</b>	<b>255</b>
XML-RPC	256
Zprávy	256
Požadavek	257
Odpověď	257
Chyba	258
Klient	258
Retrospekce	261

Server	261
SOAP	263
PEAR::SOAP	263
Rozšíření PHP SOAP	267

## **Shrnutí** **269**

## Kapitola 9

# **Tradiční doplňky** **271**

## **Úvod** **271**

### **Soubory a proudy** **271**

Přístup k souborům	272
Vstup a výstup programu	274
Funkce popen()	274
Funkce proc_open()	274
Popisovače souborů	275
Roury (pipes)	275
Soubory	276
Vstupní a výstupní proudy	276
Kompresní proudy	278
Uživatelské proudy	279
Metoda stream_open()	279
Metoda stream_close()	279
Metoda stream_read()	280
Metoda stream_write()	280
Metoda stream_eof()	280
Metoda stream_tell()	280
Metoda stream_seek()	280
Metoda stream_flush()	280
Proudy URL	281
Zamykání	285
Přejmenování a přesouvání souborů	286
Dočasné soubory	287

### **Regulární výrazy** **287**

Syntaxe	288
Syntaxe vzoru	288
Metaznaky	289
Příklad 1	292
Příklad 2	293
Sekvence změny	294
Další příklady	296
Odložené porovnávání	297
Modifikátory	297
Funkce	301
Vyhledávací funkce	301
Funkce nahrazení	303
Rozdělování řetězců	306

## **Zpracování kalendářních dat** **309**

Načítání informací o datu a času	309
Formátování data a času	311
Příklad 2: Letní čas	317
Příklad 3: Zobrazení místního času v jiných časových pásmech	319
Analýza a zpracování formátu data a času	320
<b>Práce s grafickou knihovnou GD</b>	<b>321</b>
Případ 1: Formuláře s ochranou proti robotům	322
Případ 2: Sloupcový graf	327
Rozšíření Exif	331
<b>Vícebajtové řetězce a znakové sady</b>	<b>334</b>
Konverze znakových sad	335
Další funkce pro manipulaci s vícebajtovými znakovými sadami	341
Národní nastavení	344
<b>Shrnutí</b>	<b>345</b>

## Kapitola 10

### **Pracujeme s repositářem PEAR** **347**

---

<b>Úvod</b>	<b>347</b>
<b>Základní koncepce</b>	<b>348</b>
Balíčky	348
Verze	349
Čísla verzí	349
Formát číslování verzí	349
Porovnávání čísel verzí	350
<b>Jak získat PEAR</b>	<b>351</b>
Instalace z unixové/linuxové distribuce PHP	351
Windows	352
Instalace prostřednictvím instalační služby systému Windows	353
Server go-pear.org	353
Vstupní předpoklady	353
Aktivujeme místní repositář PEAR	354
<b>Instalace balíčků</b>	<b>357</b>
Jak pracovat s příkazem pear	357
Přepínače	358
<b>Konfigurační parametry</b>	<b>360</b>
<b>Vnitřní příkazy instalátoru PEAR</b>	<b>366</b>
pear install	366
pear list	370
pear info	371
pear list-all	372
pear list-upgrades	373
pear upgrade	373
pear upgrade-all	374
pear uninstall	375
pear search	375
pear remote-list	376

pear remote-info	377
pear download	377
pear config-get	378
pear config-set	378
pear config-show	378
Zkratky (shortcuts)	379

## **Uživatelská rozhraní instalátoru** **380**

Instalátor s rozhraním příkazového řádku	380
Instalátor Gtk	380
Práce s instalátorem Gtk	380

## **Shrnutí** **383**

## Kapitola 11

# **Důležité balíčky v repositáři PEAR** **385**

## **Úvod** **385**

### **Databázové dotazy** **385**

Systémy šablon	385
Základní terminologie	386
Balíček HTML_Template_IT	386
Syntaxe zástupců (placeholders syntax)	386
Příklad: Základní šablona IT	386
Syntaxe bloků	387
Příklad: Šablona IT s bloky	387
HTML_Template_Flexy	389
Příklad: Základní šablona Flexy	389
Příklad: Balíček Flexy s bloky	390
Formát značkování podle balíčku Flexy	391
Zpracování atributů HTML v interpretu Flexy	392
Interpretace prvku HTML ve Flexy	392

### **Ověřování** **394**

Stručný přehled	394
Příklad: Ověřování na základě souboru hesel	394
Příklad: Ověřování pomocí balíčku DB a uživatelských dat	395
Úvahy o zabezpečení	397
Bezpečnostní tip 1: Zakažte funkci session.trans_sid	398
Bezpečnostní tip 2: Používejte balíček Auth_HTTP	398
Bezpečnostní tip 3: Používejte zabezpečený protokol HTTPS	398
Úvahy na téma rozšiřitelnosti	398
1. způsob rozšíření ověřování: Vyvážení zatížení pomocí ID relace	399
2. způsob rozšíření ověřování: Zachovejte relaci na jednom serveru	399
3. způsob rozšíření ověřování: Společné úložiště relací	399
Shrnutí tématu balíčku Auth	399

### **Zpracování formulářů** **399**

Balíček HTML_QuickForm	400
Příklad: přihlašovací formulář	400
Příjem dat	400

<b>Mezipaměť</b>	<b>400</b>
Balíček Cache_Lite	401
Příklad: Ukládání výstupu do mezipaměti	401
<b>Shrnutí</b>	<b>402</b>

## Kapitola 12

# Tvorba komponent pro PEAR **403**

---

<b>Úvod</b>	<b>403</b>
<b>Standardy PEAR</b>	<b>403</b>
Pojmenování symbolů	404
Konstanty	404
Globální proměnné	404
Funkce	404
Třídy	405
Metody	405
Členské proměnné	405
Odsazení	406
<b>Číslování verzí</b>	<b>407</b>
<b>Prostředí příkazového řádku</b>	<b>408</b>
<b>Základy</b>	<b>409</b>
Kdy a jak zahrnovat externí soubory	410
Ošetření chyb	410
<b>Tvorba balíčků</b>	<b>411</b>
Příklad PEAR: HelloWorld	411
Tvorba komprimovaného archivu	413
Ověřování	414
Regresní testy	414
<b>Formát souboru package.xml</b>	<b>415</b>
Informace o balíčku	416
Prvek <package>	416
Prvek <name>	416
Prvek <summary>	416
Prvek <description>	416
Prvek <license>	416
Prvek <maintainers>	416
Prvek <user>	417
Prvek <email>	417
Prvek <role>	417
Prvek <release>	417
Prvek <changelog>	417
Informace o verzi	418
Prvek <version>	418
Prvek <license>	418
Prvek <state>	418
Prvek <date>	418
Prvek <notes>	418

Prvek <filelist>	419
Prvek <dir>	419
Prvek <file>	419
Atribut name	419
Atribut role	420
Atribut platform	420
Atribut md5sum	420
Atribut install-as	420
Atributy debug a zts	421
Atributy phpapi a zendapi	421
Atribut format	421
Prvek <provides>	421
Atribut name	421
Atribut type	421
<b>Závislosti</b>	<b>421</b>
Prvek <deps>	422
Prvek <dep>	422
Atribut name	422
Atribut type	422
Atribut rel	422
Atribut optional	423
Typy závislosti	423
Závislost na verzi interpretu jazyka PHP	423
Závislost na rozšíření	423
Závislost na jiném balíčku PEAR	423
Závislost na externím programu	423
Závislost na operačním systému	424
Proč se závislostem vyhýbat	424
Nepovinné závislosti	425
Několik příkladů	425
<b>Náhrada řetězců</b>	<b>425</b>
Prvek <replace>	425
Příklady	426
<b>Jak zahrnovat kód v jazyce C</b>	<b>426</b>
Prvek <configureoptions>	426
Prvek <configureoption>	426
Atribut name	427
Atribut default	427
Atribut prompt	427
<b>Publikování balíčků</b>	<b>427</b>
<b>Proces publikování</b>	<b>427</b>
<b>Sestavení balíčku</b>	<b>428</b>
Analýza zdrojových souborů	428
Generování kontrolního součtu MD5	429
Inovace souboru package.xml	429
Tvorba komprimovaného archivu	430
<b>Odeslání na server</b>	<b>430</b>
Odeslání verze	430

Hotovo!	430
<b>Shrnutí</b>	<b>430</b>

## Kapitola 13

# Upgrade na PHP 5 431

---

<b>Úvod</b>	<b>431</b>
<b>Objektový model</b>	<b>431</b>
<b>Jak předávat objekty funkcím</b>	<b>432</b>
<b>Režim kompatibility</b>	<b>433</b>
Přetypování objektů	433
Porovnávání objektů	434
<b>Další změny</b>	<b>434</b>
Přiřazení hodnoty proměnné \$this	435
Funkce get_class()	437
<b>Direktiva E_STRICT</b>	<b>438</b>
Automatická tvorba objektů	438
Modifikátory var a public	438
Konstruktory	438
Zděděné metody	439
Definujte třídy ještě před jejich užitím	440
<b>Další problémy s kompatibilitou</b>	<b>440</b>
Rozhraní příkazového řádku	440
Tokeny komentářů	440
MySQL	441
<b>Změny funkcí</b>	<b>442</b>
Funkce array_merge()	442
Funkce strrpos() a strripos()	443
<b>Shrnutí</b>	<b>443</b>

## Kapitola 14

# Výkon 445

---

<b>Úvod</b>	<b>445</b>
<b>Návrh s ohledem na výkon</b>	<b>445</b>
Návrhový tip č. 1: Hlídejte si stav	446
Stav relace	446
Izolace stavu	447
Návrhový tip č. 2: Mezipaměť!	447
Mezipaměť pro databázové dotazy a jejich výsledky	449
Mezipaměť pro volání funkcí	452
Přeložené šablony	452
Mezipaměť pro výstup skriptů	452
Návrhový tip č. 3: Jen to s tím návrhem nepřehnat!	452
Objektově orientované obálky pro integrované funkce	452
Při zobecňování buďte opatrní	453
Nesnažte se předstírat, že PHP je Java!	453



<b>Porovnávací testy</b>	<b>453</b>
Pracujeme s nástrojem ApacheBench	453
Pracujeme s nástrojem Siege	454
Testování versus skutečný provoz	455
<b>Profilování pomocí profilovacího nástroje vývojového prostředí Zend Studio</b>	<b>455</b>
<b>Profilování pomocí APD</b>	<b>457</b>
Instalace nástroje APD	457
Analýza protokolu o sledování výkonu	458
<b>Profilování pomocí Xdebug</b>	<b>461</b>
Instalace nástroje Xdebug	461
Trasování běhu skriptu	462
Práce s nástrojem KCachegrind	463
<b>Pracujeme s APC (Advanced PHP Cache)</b>	<b>466</b>
<b>Pracujeme se sadou ZPS</b>	<b>466</b>
Automatická optimalizace	467
Mezipaměť přeloženého kódu	467
Mezipaměť pro dynamický obsah	468
Kompresie obsahu	471
<b>Optimalizace kódu</b>	<b>472</b>
Mikrotesty	472
Převedte kód z jazyka PHP do jazyka C	475
Objektově orientovaný, nebo procedurální kód?	475
<b>Shrnutí</b>	<b>476</b>

## Kapitola 15

# Úvod do tvorby vlastních doplňků **477**

<b>Úvod</b>	<b>477</b>
<b>Stručný průvodce</b>	<b>478</b>
Správa paměti	482
Vracení hodnot z funkcí PHP	483
Dokončení návrhu funkce self_concat()	484
Shrnutí příkladu	485
Zapouzdření knihoven jiných dodavatelů	485
Pohnutky	485
Typ resource	487
Registrace typů prostředků	487
Tvorba a registrace nových prostředků	489
Souborové funkce	489
První vlastní funkce pro rozšíření PHP	490
Jak použít registrovaný prostředek	491
Odebrání prostředku	492
Makra používaná pro přístup k hodnotám typu zval	493
Test nového rozšíření	494
Globální proměnné	494
Jak přidávat vlastní konfigurační direktivy	496

Makra správce prostředků přizpůsobených pro souběžné užití ve více vláknech 497

**Shrnutí** **498**

## Kapitola 16

### **Skriptování z příkazového řádku** **499**

**Úvod** **499**

**Skripty příkazového prostředí** **499**

Čím se modul CLI liší od modulu CGI 500

Implicitní parametry 500

Další přepínače 501

Název a umístění konfiguračního souboru 501

Další rozdíly 502

Prostředí skriptování na příkazovém řádku 502

Uživatelský vstup 502

Doba trvání běhu 502

Zpracování magických znaků #! 503

Zpracování přepínačů příkazového řádku 503

Osvědčené postupy 506

Zpráva o způsobu užití 506

Ukončovací kód 507

Chybové zprávy 507

Řízení toku 508

Procesy 508

Rozvětvení 508

Spouštění 508

Signály 510

Příklady 511

Filtr PHP 511

**Shrnutí** **517**

## Příloha A

### **Přehled balíčků PEAR a PECL** **519**

**Čísla** **519**

**Databáze** **520**

**Datum a čas** **529**

**Elektronická pošta** **530**

**HTML** **532**

**HTTP** **542**

**Komponenty Gtk** **545**

**Konzola** **545**

**Konfigurace** **547**

**Matematika** **547**

**Mezinárodní** **550**

**Mezipaměť** **552**

<b>Nástroje a utility</b>	<b>553</b>
<b>Obrázky</b>	<b>556</b>
<b>Ověřování</b>	<b>559</b>
<b>PEAR</b>	<b>562</b>
<b>PHP</b>	<b>564</b>
<b>Platby</b>	<b>570</b>
<b>Protokolování</b>	<b>571</b>
<b>Proudy</b>	<b>571</b>
<b>Sítě</b>	<b>573</b>
<b>Souborové formáty</b>	<b>584</b>
<b>Souborový systém</b>	<b>587</b>
<b>Šifrování</b>	<b>589</b>
<b>Struktury</b>	<b>590</b>
<b>Systémové</b>	<b>592</b>
<b>Testování výkonu</b>	<b>593</b>
<b>Text</b>	<b>593</b>
<b>Vědecké</b>	<b>594</b>
<b>Webové služby</b>	<b>595</b>
<b>XML</b>	<b>596</b>
<b>Zpracování</b>	<b>604</b>

## Příloha B

### **Příručka nástroje phpDocumentor** **605**

---

<b>Úvod</b>	<b>605</b>
<b>Dokumentační komentáře</b>	<b>605</b>
<b>Seznam klíčových slov</b>	<b>607</b>
Klíčové slovo abstract	607
Klíčové slovo access	608
Klíčové slovo author	609
Klíčové slovo category	609
Klíčové slovo copyright	610
Klíčové slovo deprecated	610
Klíčové slovo example	610
Klíčové slovo filesource	611
Klíčové slovo final	611
Klíčové slovo global	612
Klíčové slovo ignore	613
Klíčové slovo inheritdoc (vložené)	613
Klíčové slovo internal, internal (vložené)	613
Klíčové slovo licence	614
Klíčové slovo link	614
Klíčové slovo link (vložené)	614
Klíčové slovo name	615
Klíčové slovo package	615

Klíčové slovo param	617
Klíčové slovo return	618
Klíčové slovo see	618
Klíčové slovo since	619
Klíčové slovo static	619
Klíčové slovo staticvar	619
Klíčové slovo subpackage	620
Klíčové slovo todo	620
Klíčové slovo uses	620
Klíčové slovo var	621
Klíčové slovo version	622

**Tabulka klíčových slov** **622**

**Jak používat nástroj phpDocumentor** **623**

## Příloha C

### **Začínáme s aplikací Zend Studio** **633**

---

**Verze 3.5.x** **633**

**O příručce** **633**

**O společnostech Zend** **633**

**Zend Studio Client: Přehled** **634**

Komponenty vývojového prostředí 634

Konfigurace klient/server 635

Instalace a registrace 635

**Editace souborů** **637**

Editace souboru 637

**Práce s projekty** **638**

Výhody práce v projektech 638

Jak vytvořit projekt 638

**Spuštění ladicího programu** **639**

Interní ladicí program 639

Vzdálený ladicí program 640

Ladění adres URL 640

**Konfigurace vzdáleného ladění a profilování** **641**

**Spuštění profilovacího programu** **642**

**Podpora produktu** **643**

Jak získat podporu 643

**Hlavní funkce** **644**

**Rejstřík** **645**

---