

Doporučuje časopis

PC WORLD

PHP 5

začínáme programovat

Jiří Bráza

- Úspěšná učebnice, rozšířená o novinky PHP 5
- Cesta k profesionálnímu dynamickému webu
- Seznámení s jazykem PHP
- Řada zajímavých příkladů
- Podrobný popis objektového modelu
- Software pro vývoj aplikací v PHP



GRADA

Upozornění pro čtenáře a uživatele této knihy

Všechna práva vyhrazena. Žádná část této tištěné či elektronické knihy nesmí být reprodukována a šířena v papírové, elektronické či jiné podobě bez předchozího písemného souhlasu nakladatele. Neoprávněné užití této knihy bude **trestně stíháno**.

Používání elektronické verze knihy je umožněno jen osobě, která ji legálně nabyla a jen pro její osobní a vnitřní potřeby v rozsahu stanoveném autorským zákonem. Elektronická kniha je datový soubor, který lze užívat pouze v takové formě, v jaké jej lze stáhnout s portálu. Jakékoliv neoprávněné užití elektronické knihy nebo její části, spočívající např. v kopírování, úpravách, prodeji, pronajímání, půjčování, sdělování veřejnosti nebo jakémkoliv druhu obchodování nebo neobchodního šíření je zakázáno! Zejména je zakázána jakákoliv konverze datového souboru nebo extrakce části nebo celého textu, umístování textu na servery, ze kterých je možno tento soubor dále stahovat, přitom není rozhodující, kdo takovéto sdílení umožnil. Je zakázáno sdělování údajů o uživatelském účtu jiným osobám, zasahování do technických prostředků, které chrání elektronickou knihu, případně omezují rozsah jejího užití. Uživatel také není oprávněn jakkoliv testovat, zkoušet či obcházet technické zabezpečení elektronické knihy.





Copyright © Grada Publishing, a.s.

Obsah

Předmluva	11
Používané konvence	13
1. Základy	15
1.1 Co je PHP?	15
1.1.1 Server vs. klient	16
1.1.2 Interpretovaný vs. kompilovaný kód	16
1.2 Historie PHP	17
1.2.1 Manuál	18
1.3 Instalace a konfigurace	19
1.3.1 PHP	20
1.3.2 Webový server	21
Instalace Apache	22
1.3.3 Databáze	23
1.3.4 Svatá PHP trojice	25
1.4 Editory	25
1.4.1 PHPEd	26
1.4.2 HTML-Kit	28
1.4.3 PSPad	28
1.4.4 Další editory	29
2. Jazyk PHP	31
2.1 Vkládání PHP do HTML	31
2.1.1 Kombinace HTML a PHP	33
2.2 První skript	33
2.3 Základní syntaxe	34
2.3.1 Struktura	35
2.4 Komentáře	37
2.4.1 Vnořování komentářů	37
2.5 Proměnné	38
2.5.1 Datové typy	38
2.5.2 Odkaz na proměnnou	39
2.5.3 Proměnná v proměnné	39

2.6 Pole	40
2.6.1 Vícerozměrná pole	42
2.6.2 Asociativní pole	42
2.7 Řetězce	43
2.7.1 Únikové sekvence	44
2.7.2 Interpretace proměnných	46
2.7.3 Syntaxe HEREDOC	47
2.8 Výrazy	48
2.8.1 Matematické operátory	49
2.8.2 Operace s řetězci	51
2.8.3 Operátory inkrementace a dekrementace	52
2.8.4 Operátory porovnávání	53
2.8.5 Logické operátory	53
Negace	54
Konjunkce	54
Disjunkce	55
Exkluzivní disjunkce	55
2.8.6 Bitové operátory	56
2.8.7 Složené operátory	56
2.8.8 Podmíněný operátor	57
2.8.9 Priority operátorů	57
2.9 Funkce	58
2.9.1 Deklarace vlastních funkcí	59
2.9.2 Platnost proměnných	61
2.9.3 Předávání parametrů odkazem	63
2.9.4 Variabilní funkce	64
2.9.5 Funkce s proměnným počtem parametrů	64
2.9.6 Standardní hodnoty parametrů	67
3. Objekty	69
3.1 Deklarace vlastností	72
3.2 Vytváření instancí	72
3.3 Práce s vlastnostmi objektu	73
3.4 Deklarace metod	74
3.5 Volání metod	75
3.6 Statické metody	76
3.7 Konstruktor	77

3.8 Dědičnost.....	78
3.8.1 final, zamezení překrytí	79
3.8.2 Volání metod předka	80
3.9 Abstraktní třídy a metody	80
3.10 Rozhraní.....	83
3.11 Závěrem.....	85
4. Řízení běhu programu.....	87
4.1 Jednoduchá podmínka: if.....	87
4.2 Volitelná část else	89
4.3 Podmínka v podmínce – elseif	90
4.3.1 Na co si dát pozor	92
4.4 Rozcestník switch	93
4.5 Cyklus for.....	96
4.5.1 Alternativní syntaxe	98
4.6 Průchod polem pomocí foreach	98
4.7 Cyklus while	99
4.8 Cyklus do-while	100
4.9 Příkazy break a continue	102
4.9.1 break	102
4.9.2 continue.....	104
4.9.3 Opuštění více cyklů najednou	104
4.10 Zpracování výjimek.....	105
4.10.1 Mechanismus výjimek	105
4.10.2 Druhy výjimek.....	105
4.10.3 Výjimky v PHP	106
4.10.4 Objekt Exception	107
4.10.5 Shrnutí.....	107
4.11 Příkaz exit	108
4.12 Vkládání souborů	109
4.12.1 require a include.....	109
4.12.2 require_once a include_once	112
4.12.3 virtual.....	113
4.12.4 readfile.....	113

5. Formuláře.....	115
5.1 Formuláře v HTML.....	115
5.1.1 Element form	116
5.1.2 Element input	117
Textové pole	118
Zadávání hesla	119
Zaškrťovací políčko.....	120
Přepínač	121
Skryté pole.....	122
Odesílání formuláře	122
Výmaz zadaných hodnot	123
Obrázek jako tlačítko	124
Přenos souborů	125
Obecné tlačítko.....	125
5.1.3 Výběr	126
5.1.4 Textové pole	127
5.1.5 Kombinované tlačítko	128
5.1.6 Skupiny voleb	128
5.1.7 Aktivní popisky.....	129
5.1.8 Bloky vstupních polí	130
5.2 Zpracování formuláře v PHP	131
5.2.1 Za bezpečnější prostředí.....	132
5.2.2 Jednoduché předávání proměnných	134
5.2.3 Sám na sebe	135
5.2.4 Předání polí	138
5.2.5 Práce s nahranými soubory.....	140
5.3 Rady pro tvorbu formulářů	147
6. Práce se soubory	159
6.1 Přístupová práva	159
6.2 Bezpečnostní nastavení	160
6.3 Manipulace s celými soubory.....	160
6.4 Manipulace s obsahem souborů	161
6.4.1 Čtení dat ze souboru	162
6.4.2 Zápis do souboru.....	164
6.4.3 Posuny ukazatele pozice.....	165
6.4.4 Uzavření souboru	165
6.4.5 Příklad	165

7. Databáze.....	169
7.1 Základy.....	170
7.2 Jazyk SQL	170
7.2.1 Vytvoření databáze	171
7.2.2 Vytvoření tabulky	171
7.2.3 Modifikace struktury tabulky	174
7.2.4 Smazání tabulky	175
7.2.5 Vkládání záznamů	176
7.2.6 Mazání záznamů	177
7.2.7 Úprava záznamů	178
7.2.8 Výběr dat – příkaz SELECT	179
7.3 Práce s MySQL v PHP	181
7.3.1 Připojení k databázovému serveru	181
7.3.2 Perzistentní připojení	182
7.3.3 Výběr databáze	182
7.3.4 Položení dotazu	183
7.3.5 Zpracování výběrového dotazu	184
7.4 Práce s ODBC databází v PHP	190
7.4.1 Instalace ovladačů pro ODBC	190
7.4.2 ODBC v PHP	192
7.5 Databáze SQLite	196
8. Co se jinač nevešlo.....	199
8.1 Cookies	199
8.1.1 Poslání cookie	200
8.1.2 Práce s cookies	201
8.1.3 Jednoduché počítadlo přístupů	201
8.2 Session proměnné	202
8.2.1 Používání session proměnných	203
8.2.2 Jednoduchá autentizace	204
8.3 Elektronická pošta.....	207
8.3.1 Odesílání pošty	207
8.3.2 IMAP funkce	208
8.4 Práce s XML	209
8.4.1 PHP v XML	209
8.4.2 Zpracování XML dokumentů	212

8.4.3 XML parser	213
8.4.4 Práce se styly	216
9. Ladění aplikací	219
9.1 Chybová hlášení PHP	220
9.2 Nejčastější chyby a jejich příčiny	221
9.2.1 Neinicializované proměnné	222
9.2.2 Syntaktické chyby	223
9.2.3 Logické chyby	225
9.3 Obsluha chyb v PHP	227
9.3.1 Nastavení PHP	227
9.3.2 Zachycení chyb	228
9.3.3 Vlastní funkce na obsluhu chyb	229
9.3.4 Vlastní chyby	231
9.4 Použití ladících nástrojů	231
10. Zdroje informací na internetu	233
10.1 PHP.NET	234
10.2 České programátorské servery	234
10.3 Zahraniční programátorské servery	235
10.3.1 HotScripts	235
10.3.2 PHPClasses	235
10.3.3 PHP Builder	235
10.3.4 Web Developer's Virtual Library	236
10.4 Diskuzní servery	236
Rejstřík	237

Předmluva

Držíte v ruce upravenou učebnici jazyka PHP. Jazyk PHP vznikl před několika lety jako jednoduchý projekt pro vylepšení webových stránek a dnes se jedná o velmi komplexní programovací jazyk pro dynamické webové aplikace. PHP se neustále rozvíjí a umožňuje svým uživatelům spolupracovat s mnoha dalšími technologiemi, včetně například databází, XML, XSLT, umožňuje upravovat a vytvářet grafické soubory či dokumenty ve formátu PDF. Zkrátka a dobře, dovolí vám prakticky vše, co budete na svém webu potřebovat. Díky tomu uživatelé vašich stránek budou mít přístup k nejaktuálnějším informacím ve vhodném grafickém kabátě.

Tato kniha samozřejmě není první knihou o PHP a pravděpodobně není ani poslední. Před dvěma lety vyšla tato učebnice pro jazyk PHP 4. Nyní je k dispozici pátá verze, a tedy také upravená verze učebnice PHP.

Právě proto, že se PHP neustále vyvíjí a umožňuje využívat další a další technologie, stávají se některé starší publikace zároveň zastaralými. Některé principy platí, ale některé již také ne. Trendem je zachovat v co největší míře zpětnou kompatibilitu, ale protože se PHP stále častěji nasazuje na velmi rozsáhlé projekty, je zároveň snaha přiblížit jeho možnosti jiným moderním programovacím jazykům. Z původně skriptovacího jazyka doplňu-

jícího dynamické možnosti do statických HTML stránek vyrostl komplexní programovací jazyk, s jehož pomocí lze vytvářet rozsáhlé aplikace i knihovny objektů. Přitom si z velké části zachoval svou jednoduchost a stručnost zápisu.

Většina změn a novinek se tedy nedotkne dříve vytvořeného kódu a prakticky všechny aplikace psané pro PHP 4 by měly fungovat také pod PHP 5. Naopak už to samozřejmě nefunguje, a proto u všech vlastností, jež jsou v PHP 5 nové, bude čtenář upozorněn, že ve starších verzích nebudou fungovat.



Tato učebnice je zaměřena na PHP 5. V této nové verzi jazyka byl především podstatně změněn objektový model a přidány některé další nové vlastnosti. Základ jazyka však zůstal stejný, a proto většinu textu lze bez problémů použít i pro učení se jazyku PHP, který si budete zkoušet na starší verzi, tj. PHP 4. Vždy budete upozorněni, pokud je některá vlastnost novinkou v PHP 5.

Tato publikace vás naučí základy jazyka PHP způsobem, který by měl pochopit i začátečník bez jakýchkoli zkušeností s jazykem PHP. Postupně se naučíte pracovat s proměnnými, funkcemi, ale i objekty a dalšími prvky jazyka. Na těchto základech pak staví další kapitoly, které se zabývají několika nejčastěji používanými oblastmi, ve kterých je PHP nasazováno. Jedná se především o zpracování formulářů a přístup k databázím, ale najdete zde také mnohé další oblasti, jako například manipulaci se soubory, práci s XML dokumenty i šablonami XSLT, cookies, session proměnné či elektronickou poštu. Veškerý výklad je samozřejmě doplněn jednoduchými okomentovanými příklady, na nichž poznáte, jak se v PHP programuje.

Neváhejte si jednotlivé ukázky vyzkoušet, a především upravit podle svých vlastních potřeb. Popis instalace všech produktů potřebných či užitečných pro vyvíjení vlastních aplikací v PHP v knize najdete také.



Pouhým opisováním příkladů se programovat nenaučíte, ale je to dobrý začátek. Vždy se snažte předkládaný program pochopit (měly by vám k tomu dopomoci bohaté komentáře) a později se jej pokuste upravit a rozšířit. Hlavně je třeba, abyste programovali vy sami. Stejně jako se čtením nenaučíte psát, ani čtením programů se nenaučíte programovat.

PHP jako jazyk je také poměrně specifické tím, že programy psané v PHP jsou obvykle součástí webových stránek, a jsou tedy prováděné na webu. Kvůli tomu je poněkud problematictější ladění aplikací, a tomuto tématu je věnována samostatná kapitola.

Přeji vám, aby vám kniha byla užitečným pomocníkem nejen při vašich začátcích v PHP, ale i později, kdy si budete chtít doplnit informace v oblastech, na něž jste v PHP dosud nenarazili.

Jiří Bráza

Používané konvence

V knize se setkáte s několika ikonami, které označují zvláštní typy odstavců:

Symbolem poznámky bude uveden odstavec, který rozšiřuje probíranou problematiku o nějakou zajímavost či výjimečnost.



Symbol vykřičníku upozorňuje na úskalí, se kterým se můžete při práci setkat, a je proto dobré zůstat stále ve střehu.



Tato ikona nabízí zpravidla nějaké usnadnění nebo tip, kterým si oproti standardnímu postupu můžete práci ulehčit.



1.

Základy

V úvodní kapitole si nejprve stručně řekneme, co je vlastně PHP zač, v čem a jak nám může pomoci, či s jakými technologiemi ho můžeme srovnávat. Dozvíte se, jak PHP vzniklo i jak jej nainstalovat. Kromě samotného PHP budeme pracovat ještě s dalšími aplikacemi – pravděpodobně se neobejdete bez webového serveru a databáze, a stejně tak bude vhodné vybrat si dobrý editor pro psaní kódu.

Naučíme se také vkládat kód PHP do stránek HTML a ukážeme si první „program“ vytvořený v PHP. V následující kapitole se potom dozvíte veškeré náležitosti pro tvorbu programů v PHP.

1.1 Co je PHP?

PHP je hypertextový preprocesor, který na serveru interpretuje stránky HTML s vlastními příkazy před jejich odesláním ke klientovi (obvykle je jím webový prohlížeč). To znamená, že PHP umožňuje vkládat vlastní skripty (krátké úseky kódu, ale i celé programy) přímo do hypertextových stránek. To není nic neobvyklého – do hypertextových stránek lze vkládat

také například kód v JavaScriptu. Existuje zde však několik podstatných rozdílů. Jednak je PHP interpretováno na serveru, zatímco JavaScript je jazyk interpretovaný teprve klientem. Oba způsoby se v některých ohledech velmi podstatně liší.

1.1.1 Server vs. klient

Je nesmyslné řešit, zdali je lepší programový kód provádět na serveru či na straně klienta, neboť obojí má svá pro i proti. Provádění kódu na serveru přináší především tyto výhody:

- ✓ Snadná interakce s dalšími aplikacemi na serveru, není třeba přenášet data po internetu, snáze se zajišťuje bezpečnost celého systému.
- ✓ Nenáročnost na hardware či software klienta – výstupem je obvykle čisté HTML, které dokáže interpretovat i starší prohlížeč. Není potřeba, aby na klientovi byly prováděny skripty či rozsáhlé výpočty, aby podporoval zásuvné moduly atd.
- ✓ Menší objem přenesených dat – serverové skripty, jež do kódu vložíte, jsou ještě na serveru interpretovány a ve výstupu se již neobjeví, zatímco skripty určené pro klienta ano.
- ✓ Výrazně vyšší ochrana zdrojových textů programů – díky interpretaci kódu na serveru se klient nemůže dostat ke zdrojovým textům, zatímco v případě klientských technologií jsou mu zdrojové texty přímo zaslány.

Naopak provádění kódu u klienta má také několik výhod:

- ✓ Menší zátěž serveru. Server pouze dostane požadavek a odešle stránku, zatímco v případě serverových technologií musí ještě spustit interpret jazyka, předat mu stránku a klientovi zasílá až výsledek. PHP může běžet také jako modul webového serveru, čímž se výrazně snižuje reakční doba i paměťové nároky, takže vliv interpretace PHP kódu na rychlost odezvy serveru je minimální.
- ✓ Možnost interakce s objekty na webové stránce. Klientské technologie vznikají především z tohoto důvodu. Jen u klienta je možné reagovat na stisk kláves, pohyb či klepnutí myši apod.

Z výše uvedeného přehledu také jasně vyplývá, kdy je vhodné které technologie použít. Zatímco pro manipulace s objekty uvnitř webové stránky je vhodné použít např. JavaScript, PHP se velmi dobře hodí pro spolupráci s databázemi, zpracování formulářů a náročnější úlohy jako manipulace s grafikou či soubory PDF.

1.1.2 Interpretovaný vs. kompilovaný kód

Několikrát jsem v textu zmínil termín **interpret PHP**. Jazyk PHP je totiž interpretovaný, což znamená, že až do okamžiku svého použití (spuštění) je uchováváno ve zdrojovém tvaru. Interpret jazyka tento kód vezme a překládá jej do strojového kódu pro počítač, na kterém PHP běží. To je samozřejmě pomalejší než v případě **kompilovaného kódu**, který je k dispozici již přímo ve spustitelném tvaru – cílový počítač soubor vezme, zde uvedeným instrukcím rozumí a může je ihned provádět. Na internetu se můžete setkat s oběma přístupy. Jistě jste si všimli, že na některých webových serverech se vám občas (dnes již skutečně spíše výjimečně) v adrese objeví soubor s příponou `.exe` či `.dll` – neklamná

známka toho, že server běží na operačním systému *Windows*. Ale i na jiných operačních systémech můžete nalézt CGI aplikace vytvořené například v jazyce C.

Na druhou stranu jsou na internetu velmi oblíbené interpretované jazyky – zpočátku se většina „dynamických“ stránek vytvářela na unixových systémech pomocí skriptů pro tzv. shell (obdobu dávkových souborů na počítačích MS-DOS a Windows). Později se začaly prosazovat o něco sofistikovanější interpretované jazyky, především *Perl*.

Výhodou kompilovaných jazyků je jejich vyšší rychlost, neboť pro své spuštění nevyžadují „tlumočníka“. Na druhou stranu tyto programy nejsou přenositelné mezi různými operačními systémy. Výhodou interpretovaných jazyků je též snazší možnost úprav – stačí provést úpravu ve zdrojovém souboru, který může být neustále k dispozici. V prvním případě je po provedení změny ještě třeba program znovu přeložit, aby změna byla skutečně provedena.

Na rozhraní mezi těmito technologiemi je jazyk Java (a nově také C#). Jde v podstatě o **předkompilovaný** kód, který se snaží vzít si to lepší z obou technologií (zároveň s tím však má i některé nevýhody obou). V podstatě se jedná o to, že kód je kompilovaný do strojové podoby pro jakýsi virtuální stroj, nezávislý na konkrétním operačním systému. Což znamená, že výsledný „program“ je libovolně přenositelný. Na druhou stranu tento kód není připraven ke spuštění na žádném počítači a pro jeho provedení je třeba mezivrstvy, která jej překládá do příkazů pro konkrétní počítačový systém (interpretuje). Kde je tedy výhoda? Překlad z binárního kódu nezávislého na platformě do kódu pro konkrétní počítač je výrazně rychlejší než interpretace příkazů programovacího jazyka (přesto se však nemůže srovnávat s kompilací do nativních instrukcí určité platformy).

PHP je interpretovaný jazyk, podobně jako třeba Perl. Oproti Perlu (i dalším skriptovacím jazykům) však nabízí podstatně více možností, pohodlnější práci s proměnnými i objekty, výbornou podporu mnoha databázových systémů, a také přehlednější syntaxi. Než se tímto jazykem v další kapitole začneme zabývat podrobněji, podíváme se nejen na to, jak vznikl, ale také co budeme potřebovat pro jeho využití.

1.2 Historie PHP

U zrodu celého systému byl původně soukromý program Rasmuse Lerdorfa. Napsal si jednoduchý systém pro své vlastní webové stránky, především pro sledování návštěvnosti. Kvůli zvýšení výkonosti jej později přeprogramoval v C a uvolnil k používání pro několik svých známých. Těm se velmi zalíbil a požadovali stále nové vlastnosti a chodili s novými připomínkami, čímž udrželi systém při životě a při vývoji. Při jeho uvolnění pro používání mu byl dán název *Personal Home Page*, PHP. Velké obliby se dočkal především ve své druhé verzi, obohacené především o velmi snadnou manipulaci s daty z formulářů – *PHP/FI 2.0*. FI v této zkratce znamenalo právě *Form Interpreter*. Od roku 1998 byla k dispozici verze 3.0, oproti předchozím podstatně zrychlená a obohacená o nové funkce, především podporu mnoha databázových systémů, objekty, cookies atd. Tou dobou už rozhodně nelze mluvit o PHP jako o nástroji pro domovské stránky uživatelů, PHP je nasazováno především na tisících velkých serverů poskytovatelů obsahu – zpravodajské servery, archivy softwaru atd. Proto se již vžilo označení PHP a původní význam této zkratky zapadl.

V současné době je již několik let k dispozici PHP verze 4, jež dosáhla skutečně masového rozšíření a je stále rozvíjena (nyní existuje verze 4.3.9). Využívá zcela nové jádro Zend a kromě zvýšení výkonu byl jazyk opět obohacen o mnoho nových vlastností a funkcí.

Letos v létě (2004) byla po roce betatestování uvolněna první finální verze PHP 5. Nová verze jazyka přináší velmi podstatné změny, avšak bez větších dopadů na zpětnou

kompatibilitu. Především byl kompletně přepracován objektový model jazyka, a třídy a objekty již nejsou pouhým výstřelkem, ale integrální součástí jazyka, nabízející téměř možnosti srovnatelné s objektově orientovanými jazyky (režimy přístupu, konstruktory a destruktory, volání odkazem, rozhraní atd.). Mezi další důležité změny nepochybně patří zavedení mechanismu výjimek a jmenné prostory. Vzhledem k tomu, že PHP 5 je zatím stále ještě horká novinka, která není masově nasazována na úkor PHP 4, budeme se s touto starší verzí ještě dost dlouho setkávat. A protože, jak jsem říkal, PHP 5 je téměř ve všech ohledech kompatibilní s předchozí verzí, je možné tuto knihu bez problémů využít i pro seznámení se starší verzí PHP, například pokud k nové verzi nemáte přístup. Pokud bude v knize pojednáváno o vlastnostech, jež jsou k dispozici výhradně ve verzi nové, budete na to vždy upozorněni.



Knihu je možné použít i pro seznámení se s vlastnostmi starší verze PHP. V případě nějakých odlišností na to bude čtenář vždy upozorněn.

1.2.1 Manuál

Je třeba se také zmínit o dokumentaci. Ta doznala během vývoje snad ještě větších změn než samotné PHP. Může to znít paradoxně, ale během překotného vývoje PHP, až do verze 3 včetně, dokumentace za samotným jazykem značně zaostávala. Manuál byl plný prázdných stránek místo popisu konkrétních funkcí nebo zmínek o tom, že daná funkce bude popsána později.

V současné době je dokumentace k PHP kompletní, i když poněkud trpí tím, že je již v oběhu několik verzí jazyka, jež se svými vlastnostmi mohou lišit. Je k dispozici v mnoha formátech a dokonce i v mnoha jazycích, včetně češtiny. Je pravda, že v českém manuálu občas najdete anglický popis funkce nebo anglický nadpis s českým popisem, ale už se vám rozhodně nestane, že by manuál ke konkrétní funkci zcela chyběl.



Občas se však stane, že v české verzi manuálu některé důležité informace chybí, proto není na škodu podívat se také do anglické verze.

Výborné také je, že je manuál doplněn připomínkami a poznámkami uživatelů. Nezapomeňte se do nich podívat v případě, že vám vysvětlení podané v manuálu nebude postačovat. Obvykle tam najdete nějaký zajímavý příklad či vyřešený instalační nebo konfigurační problém. Samotný manuál máte k dispozici v několika verzích:

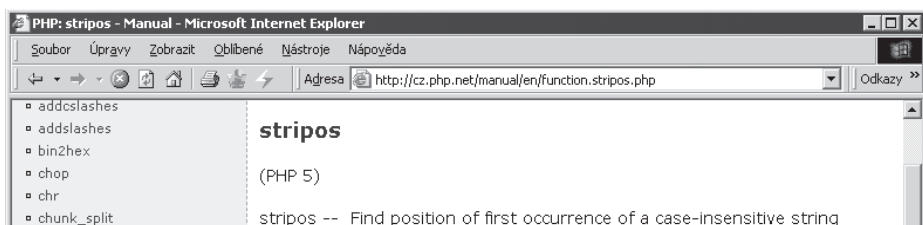
- ✓ **Online na webu** na adrese www.php.net/manual/en (anglicky), resp. www.php.net/manual/cs (česky).
- ✓ Odkaz na **verzi pro tisk** najdete na stránkách www.php.net/docs.php.
- ✓ Nemáte-li trvalé připojení k internetu, jistě budete mít zájem o stažení **offline verze** manuálu k PHP. Ke stažení je na stránce www.php.net/download-docs.php k dispozici několik verzí v různých jazykových mutacích:
 - ✓ kompletní manuál jako jedna stránka HTML,
 - ✓ manuál sestávající z mnoha (několik tisíc) souborů stejně jako na webu,

- ✓ jako kompilovaný HTML help pro Windows (formát CHM) – k dispozici je jeden soubor se samotným manuálem a druhý s připomínkami uživatelů (pouze v angličtině).

Máte-li přímé připojení k internetu, není třeba nejprve navštívit hlavní stránku, ze které je možné přímo vyhledávat v seznamu funkcí. Hledáte-li nápovědu ke konkrétní funkci, zkuste ji zadat do URL ihned za adresu serveru, např. tedy `www.php.net/trim`. Tuto možnost lze využít i pokud nehledáme přímo určitou funkci, ale třeba kompletní tematickou oblast – `www.php.net/array`. Můžete se také podívat na stránku `www.php.net/tips.php`, kde najdete několik užitečných tipů pro ještě rychlejší hledání v PHP manuálu z různých problémech.



Všimněte si, že u každé funkce je hned pod jejím názvem závorka, ve které najdete informace o kompatibilitě, tj. v jakých verzích PHP je zmíněná funkce k dispozici. Může se samozřejmě stát, že třeba díky rozšiřujícímu modulu budete mít příslušnou funkci k dispozici i v jiné (starší) verzi.



Obr. 1.1: Manuál k PHP – nápověda k jednotlivé funkci

1.3 Instalace a konfigurace

Než začneme povídku o vlastnostech jazyka PHP, podíváme se na to, co všechno budeme pro práci s PHP potřebovat, kde to seženeme, jak nainstalujeme a jak upravíme podle svých potřeb.

V úplně minimální verzi (pro praktické použití zcela nevhodné) si vystačíme pouze s **interpretrem jazyka PHP**. Zdrojové texty programů můžeme psát v libovolném textovém editoru, včetně *Poznámkového bloku* a PHP využijeme pouze jako interpret spouštěný z příkazového řádku. S webem však toto řešení nemá nic společného. Také tvorba stránek vám půjde lépe od ruky v **editoru**, který vám bude asistovat poněkud více než pouze ukládat stisknuté znaky. Více informací o editorech pro tvorbu kódu v PHP najdete v následující podkapitole.

Hlavním důvodem nasazení jazyka PHP je jeho použití v dynamických webových stránkách. To však znamená, že kromě samotného PHP potřebujeme ještě **webový server**. Stačí některý drobný lokální webový server pro vaši potřebu, nemusíte jej otevírat do světa.

Velmi často budete navíc chtít na webu pracovat s **databází**. PHP dokáže spolupracovat s množstvím nejrůznějších databázových systémů. Díky obsažené podpoře standardu ODBC se množina databází, s nimiž můžete z PHP komunikovat, ještě zvyšuje.

Když to shrneme, potřebujeme obvykle následující komponenty:

- ✓ PHP pro interpretaci kódu PHP uvnitř vašich stránek,
- ✓ webový server pro automatizaci spouštění PHP a prezentaci stránek na internetu, intranetu, či samostatném domácím serveru používaném především pro vývoj,
- ✓ databázový systém.

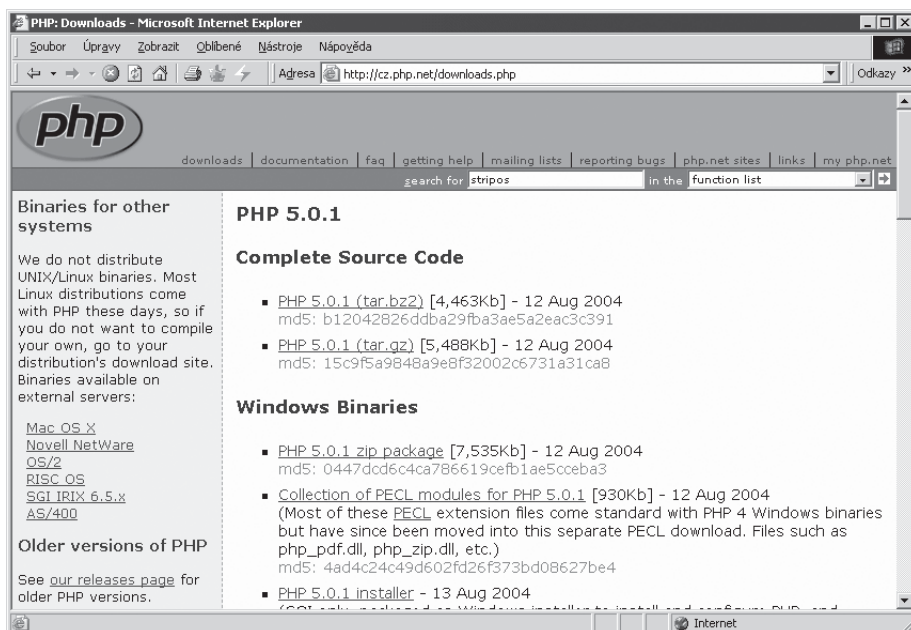
Postupně se podíváme na instalaci všech jednotlivých součástí.



Máte-li již zmíněné programy nainstalované (např. pokud využíváte webhosting s podporou PHP), můžete tuto podkapitulu bez problémů přeskočit, případně se k ní vrátit tebdy, až budete jeden či více zmínovaných produktů potřebovat nainstalovat.

1.3.1 PHP

Instalace PHP je velmi jednoduchá a nemusíte z ní mít strach. Začnete tím, že ze serveru `www.php.net` stáhnete vybranou verzi PHP. Můžete si zobrazit přímo stránku `www.php.cz/downloads.php`, kde najdete několik verzí ke stažení:



Obr. 1.2: Stažení PHP

- ✓ kompletní zdrojové soubory – pro nadšence, kteří si chtějí PHP upravovat nebo mají natolik obskurní systém, že pro něj není možné sehnat binární soubory.
- ✓ Binární soubory pro *Windows* – nejčastěji stahovaná položka. Operační systémy založené na Linuxu již obvykle PHP obsahují a operační systémy *Windows* jsou

dalším nejrozšířenějším systémem (a nepochybně nejrozšířenějším na osobních počítačích).

- ✓ Rozdílové verze (*patches*) pro jednoduchý upgrade z několika posledních minor-verzí.
- ✓ Binární soubory pro některé další operační systémy (*Mac OS, Novell, klony Unixu*).

Máte-li operační systém *Linux*, pravděpodobně již máte PHP v distribučních balíčcích. Pokud jej tedy nemáte přímo nainstalované, stáhněte si příslušný balíček a nainstalujte jej. Pracujete-li ve *Windows*, bude dobrou volbou stažení binárních souborů pro *Windows* (cca 5MB). Existuje ještě výrazně menší varianta, instalační soubor PHP, který dokáže nainstalovat PHP v minimální konfiguraci a ještě jej propojit s několika podporovanými webovými servery (máte-li je nainstalované). Na druhou stranu tato varianta vůbec neobsahuje žádná rozšíření (např. šifrovací funkce, podpora PDF, XML, XSL a dalších).

Stáhnete-li si kompletní binární distribuci, probíhá instalace takto:

1. Stažený soubor `.zip` rozbalte do nějakého adresáře (obvykle se používá `c:\php`).
2. Zkopírujte soubor `php.ini-dist` do vašeho adresáře s *Windows* a přejmenujte jej na `php.ini`.
3. Upravte soubor `php.ini` – některým nastavením se budeme věnovat v průběhu celé knihy. Nastavení v tomto souboru ovlivňují, kde jsou hledány soubory různého druhu (rozšíření, session proměnné, vkládané soubory a knihovny atd.), jak se chová PHP i některá instalovaná rozšíření.
4. Nyní by mělo PHP fungovat, spustíte-li jej z příkazového řádku. Až nainstalujete webový server, bude ještě třeba jej nastavit tak, aby pro soubory `.php` volal interpret jazyka PHP.

Pokud jste si stáhli pouze instalační balíček, je instalace ještě jednodušší. Začněte tím, že nainstalujete vybraný webový server (*Apache, IIS, PWS, Pinknet WebServer, Xitami* apod.), zvláště je-li zařazen mezi servery podporované instalačním programem. Poté spusťte instalaci a řiďte se pokyny. Vyberte složky, kde budete chtít spouštět PHP soubory. Po proběhnutí instalace by mělo být vše připraveno k vaší práci s PHP (možná bude potřeba restartovat webový server či celý počítač).

Máte-li funkční PHP, které potřebujete pouze pro ladění aplikací, ani webový server nepotřebujete. Vaše soubory můžete testovat buď přímo na serveru, kam mají být později umístěny, nebo lépe v editoru, který zvládne jejich interpretaci či dokonce ladění (takovým editorem je například *PHPEd*, ale dá se tak nastavit i výborný český freewarový editor *PSPad*).

1.3.2 Webový server

Pokud si na váš počítač nainstalujete webový server, budete schopni komfortně spouštět soubory PHP. V současné době již je k dispozici celá řada webových serverů i pro operační systémy *Windows*, především samozřejmě pro osobní použití.

Pro osobní použití máte na výběr celou řadu webových serverů, z nichž řada zde již byla zmíněna:

- ✓ *Personal Web Server* od společnosti *Microsoft*. Na osobní webový server (o jeho reálném nasazení ani neuvažujte) jde o pořádného otesánka. Nemáte-li jej k dispozici

na CD, rozhodně nevažujte o jeho stahování z internetu. Řada jiných osobních webových serverů je podstatně menší a udělá vám přinejmenším stejnou službu.

- ✓ *IIS* od společnosti *Microsoft* je webový server určený pro nasazení v reálném prostředí internetu.
- ✓ *Apache* je nejoblíbenějším webovým serverem na všech platformách. Je k dispozici zdarma, v současné době je již také velmi dobře zdokumentován a jeho zprovoznění je daleko snazší, než dříve. Zároveň díky jeho velkému rozšíření snadno seženete nápovědu v případě problémů. Instalaci webového serveru *Apache* se budeme blíže zabývat za chvíli.
- ✓ *Pinknet Webserver* je miniaturní webový server pro *Windows*. Instalace je velmi rychlá, správa jednoduchá (prostřednictvím jediného dialogového okna), nabízené možnosti dostatečné a cena také sympatická (zdarma). Je k dispozici v českém jazyce, ale již řadu let se nevyvíjí. Najdete jej na adrese <http://pnws.pinknet.cz>. Celý server zabere pár set kilobajtů a snadno se spravuje.
- ✓ *Xitami* je další miniaturní webový server, na rozdíl od předchozího pouze v angličtině.
- ✓ *OmniHTTPd* je také často používaný osobní webový server. Jednoduchá instalace, jednoduchý provoz.

Podrobněji se podíváme na instalaci webservru *Apache*, použitelného i pro reálný webový server.

Instalace Apache

Začněte stažením binárních souborů pro *Windows* z webových stránek <http://httpd.apache.org>. Chcete-li si zřídit *Apache* pro *Linux*, najdete jej nepochybně v distribučních balíčcích. V závislosti na volbách při instalaci jej dokonce nejspíš již máte nainstalován. Pro *Windows* je možné *Apache* získat ve dvou variantách: jako archiv `.zip` nebo přímo jako instalační soubor pro *Windows Installer* (přípona `.msi`).

K dispozici jsou v současné verze dvě stabilní vývojové větve, starší platforma 1.3.x a novější 2.0.x. Pro domácí server vám pravděpodobně postačí server „jedničkové“ řady, jehož výhodou je i mnohem vyšší stabilita při spolupráci s modulem PHP.

Samotná instalace serveru by měla proběhnout bez problémů. Po spuštění serveru byste na adrese `http://localhost` měli nalézt funkční webovou stránku informující o tom, že server *Apache* je v pořádku nainstalován a funkční. Je však třeba doplnit podporu pro PHP. Standardně je na *Windows* PHP používáno jako běžná CGI aplikace, ale pro *Apache* (a některé další webové servery) existuje i ve variantě webového modulu.

Pro novou (2.0.x) distribuci *Apache* je podpora modulu pouze experimentální, takže máte na vybranou mezi vyšším výkonem (SAPI) a vyšší stabilitou (CGI). Chcete-li spouštět PHP jako aplikaci CGI, vložte do konfiguračního souboru *Apache* (soubor `httpd.conf` v podsložce `conf` složky *Apache*) tyto řádky:

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