

Martin Mikulák

Programování WWW stránek *pro úplné začátečníky*



*Názorná příručka krok za krokem
Od úplných základů po hotové aplikace
Formuláře, fotogalerie, kniha návštěv
Tvorba administrace webu a databáze*

computer
press

Martin Mikulák

Programujeme WWW stránky pro úplné začátečníky

**Computer Press
Brno
2013**

Programujeme WWW stránky pro úplné začátečníky

Martin Mikulák

Obálka: Martin Sodomka

Odpovědný redaktor: Martin Domes

Technický redaktor: Jiří Matoušek

Objednávky knih:

<http://knihy.cpress.cz>

www.albatrosmedia.cz

eshop@albatrosmedia.cz

bezplatná linka 800 555 513

ISBN 978-80-251-3252-4

Vydalo nakladatelství Computer Press v Brně roku 2013 ve společnosti Albatros Media a. s. se sídlem Na Pankráci 30, Praha 4. Číslo publikace 16664.

© Albatros Media a. s. Všechna práva vyhrazena. Žádná část této publikace nesmí být kopírována a rozmnožována za účelem rozšiřování v jakékoli formě či jakýmkoli způsobem bez písemného souhlasu vydavatele.

Dotisk 1. vydání

ALBATROS  **MEDIA** a.s.

Obsah

Úvodem	9
Zpětná vazba od čtenářů	10
Zdrojové kódy ke knize	10
Errata	10
Kapitola 1	
Než začneme	11
Dynamické vs. statické stránky	11
Co je a k čemu slouží PHP	12
Instalace potřebného softwarového vybavení	13
Instalace webového serveru	13
Spuštění serveru Xampp	15
Kontrola správné instalace	17
Zkopírování zdrojových kódů ke knize na lokální webový server	18
Adresářová struktura a přístup k souborům	18
Textové editory se zvýrazněním syntaxe	19
Instalace textového editoru pro skripty	19
Jak k práci využít textový editor pro skripty	20
Výchozí podoba ukázkového webu	20
Kapitola 2	
Dynamické zobrazování obsahu	25
Přetypování statických souborů	25
Ahoj světe!	27
Ukládáme údaje do proměnných	28
Volání externích souborů při generování dynamické stránky	29
Volání externích souborů – princip fungování	30
Části stránky v samostatných souborech a jejich volání	32
Parametry v URL adrese	36
Načítání hodnot parametrů z URL adresy	37

Obrana proti vkládání škodlivého kódu přes parametr URL adresy	38
Volání externích skriptů s texty a ošetření všech možností	39
Rozšiřujeme své znalosti	42
Správné programátorské návyky	43
Podmínky a jejich vyhodnocování	43
Vytváříme cykly	44

Kapitola 3

Tvoříme kontaktní formulář	49
Fungování formulářů	50
První formulář	51
Metody POST a GET	53
Tlačítka ve formulářích	55
Vstupní pole – různé druhy	55
Klasické jednořádkové textové pole	55
Textová oblast	58
Pole pro heslo	59
Skrytá pole	59
Výběr z možností	60
Roletová nabídka	62
Zaškrťovací políčko	63
Formulář i výkonný skript v jediném souboru	65
Návrh a spuštění kontaktního formuláře	66
Obsah kontaktního formuláře	66
HTML kód formuláře	67
Ověření údajů vložených do formuláře	68
Ověření dat v kontaktním formuláři	76
Odesílání e-mailů	78
Fungování kontaktního formuláře	83
Ochrana formuláře proti spamům	85

Kapitola 4

Návrh databází	87
Jak vypadá databáze	87
Datové typy	87
Číselné typy	88
Textové (řetězcové) typy	88
Dvě funkce identifikátoru	89
Práce v prostředí PHPMyAdmin	90
Vytvoření nové databáze	91

Vytvoření nové tabulky v databázi	93
Záložky Struktura a Projít	95
Vkládání záznamu do tabulky	95
Zobrazování záznamů v tabulce	96
Úprava záznamů v tabulce	97
Smazání záznamu z tabulky	97
Úprava nebo smazání více záznamů současně	98
Export a import databází a tabulek	99

Práce s databází v jazyce PHP **103**

Připojení k databázi	104
Možné chyby při připojování k databázi a jejich řešení	106
Výběr dat z databáze	106
Vkládání nových záznamů do tabulky	119
Úprava existujících záznamů	120
Mazání záznamů z tabulky	122
Doplňující informace	123

Kapitola 5

Dynamický obsah načítaný z databáze **125**

Vytvoření databáze a tabulky pro texty	126
Skript pro přípravu textů	126
Jak to funguje?	129
Proč vyplňovat značky unikátními texty	131
Co když požadovaná stránka neexistuje?	132

Kapitola 6

Připravujeme administrační zónu **135**

Relace – pamatování informací do okamžiku zavření prohlížeče	135
Soubory cookie – „koláčky“, které si informace pamatují i mezi relacemi	138
Nastavení času vypršení cookie	140
Zrušení platnosti souboru cookie	142
Zobrazení souborů cookie přes prohlížeče	142
Práce s časem a datem	148
Jednoduchá správa obsahu – administrační zóna	150
Princip fungování	150
Co dělají jednotlivé skripty	151
Administrační zóna v praxi	158
Změna hesla přímo v administrační zóně	163
Tipy na vylepšení administračního rozhraní	175

Kapitola 7

Tvorba fotogalerie a úprava obrázků

177

Nahrávání souborů na webový server

177

Přesun souborů po nahrání na server

178

Zjištění dalších informací o souboru

182

Vytváření obrázků pomocí PHP

186

Míchání barev

188

Jednoduché tvary

188

Průhlednost

191

Ochrana formuláře proti spamerům

193

Práce s obrázky, které odesílají uživatelé

196

Přetypování obrázků

196

Využíváme dostupné filtry

199

Zmenšování obrázků

205

Vkládáme vodoznaky

208

Mazání souborů

209

Praktická realizace fotogalerie

210

Mazání obrázků přímo v administrační zóně

216

Zobrazení fotogalerie v části pro uživatele

220

Kapitola 8

Kniha návštěv

223

Analýza požadavků

223

Návrh databáze a skriptů

224

Skript pro obsluhu knihy návštěv

224

Další potřebné úpravy

227

Přidání textu/stránky do databáze

227

Testování a ladění

230

Správa knihy návštěv v administrační zóně

233

Zobrazení, úprava a mazání příspěvků

233

Testování a ladění administrační zóny

237

Možná vylepšení

238

Kapitola 9

Testování a oprava chyb

239

Chyby způsobené překlepy

239

Nalezen neočekávaný znak

239

Očekávaný znak nenalezen

239

Chyby ve funkcích	240
Hlavička odeslána	241
Logické chyby	241
Chyby v proměnných	241
Nenastartované relace	242
Soubory Cookie	242
Cykly a podmínky	242
Nevykonání příkazů uvnitř podmínky	242
Zakomentování částí skriptu	243
Testování a ladění	243
Jak testovat	243
Závěrem	245
Příloha A	
Časté konstrukce	247
PHP	247
Větvění (rozhodovací struktury)	247
Cykly	250
MySQL	252
Připojení k databázi	252
Vkládání nových záznamů do databáze	252
Výběr a výpis dat z databáze	253
Aktualizace záznamu	254
Mazání záznamů	254
Příloha B	
Přehled funkcí a příkazů použitých v knize	255
Všeobecné funkce	255
Práce s textem	255
Práce s poli	256
Práce se soubory	256
Práce s časem	257
Odesílání e-mailů	257
Práce s obrázky	257
Rejstřík	259

Úvodem

V dnešní době to někdy vypadá, že Internet vládne světu. Dnes už není problém vyřídit mnoho běžných záležitostí přímo od počítače, aniž bychom museli někam chodit, a to během pár minut.

Když potřebujeme něco koupit, na Internetu si vyhledáme informace o dostupných řešeních. Zjistíme si reference lidí na jednotlivá řešení a firmy, které je poskytují. Až pak se rozhodneme, že konkrétní firmu navštívíme osobně a využijeme jejích služeb. Mnohdy však není potřeba ani to.

Internet nabízí mnoho možností: můžeme nakupovat zboží, psát vlastní názory na blogu nebo komunikovat s přáteli, kteří jsou třeba na opačném konci světa. Vše okamžitě a bez problémů.

Pravděpodobně i vás zajímá, jak vytvořit kvalitní dynamické stránky, které budou snadno rozšiřitelné o nové funkce a obsah. To je totiž to, co mají lidé na Internetu rádi – kvalitní obsah podaný v přehledné formě. A zajímavé funkce k tomu.

V této knize si ukážeme, jak přetvořit běžné (X)HTML statické stránky do dynamické podoby. Po přečtení této knihy budete umět psát vlastní skripty a navrhopvat a používat databáze. Budete si umět vytvořit vlastní administrační zónu a díky tomu pro vás bude přidávání nového obsahu na stránky hračkou.

Celá kniha je napsaná snadno srozumitelným jazykem a prakticky při všem používá ukázkové příklady. Na začátku jednotlivých kapitol se nejdříve naučíme používat nové funkce a získané znalosti pak použijeme na konkrétním řešení, které implementujeme do bývalých statických stránek.

Původní jednoduché stránky tak rozšíříme o spojení s databází, o fotogalerii s automatickou úpravou fotek, kontaktní formulář s odesíláním zpráv na e-mail, knihu návštěv a také administrační rozhraní pro správu obsahu.

Díky znalostem, které získáte, pro vás nebude problém vytvořit vlastní funkcionalitu nebo modul.

V této knize se budeme věnovat pouze jazykům PHP a MySQL – a to od úplných základů. Nebudeme se však zabývat vysvětlováním jazyka (X)HTML a jeho značek ani tvorbou stylů pomocí CSS. Předpokládáme, že tyto znalosti již máte.

Celá kniha je členěna do devíti kapitol a obsahuje také dvě přílohy.

V první kapitole si nainstalujeme potřebné softwarové vybavení a podíváme se na statické stránky, které budeme předělávat.

V dalších sedmi kapitolách se budeme postupně učit psát skripty v jazycích PHP a MySQL a budeme předělávat původní statické stránky.

V deváté kapitole naleznete užitečné rady a tipy, jak řešit případné problémy a jak testovat a ladit vaše webové aplikace.

V příloze A pak uvádíme přehled nejčastěji používaných konstrukcí jazyků PHP a MySQL i s krátkými příklady použití.

A nakonec v příloze B najdete seznam funkcí jazyka PHP, které budeme v knize používat při tvorbě našich skriptů.

Dost bylo slov – vrhněme se na tvorbu dynamických stránek!

Zpětná vazba od čtenářů

Nakladatelství a vydavatelství Computer Press stojí o zpětnou vazbu a bude na vaše podněty a dotazy reagovat. Můžete se obrátit na následující adresy:

redakce PC literatury

Computer Press

Spielberk Office Centre

Holandská 3

639 00 Brno

nebo

sefredaktor.pc@cpress.cz

Computer Press neposkytuje rady a konzultace ohledně programování.

Zdrojové kódy ke knize

Z adresy <http://knihy.cpress.cz/k1868> si po klepnutí na odkaz Soubory ke stažení můžete přímo stáhnout archiv s ukázkovými kódy.

Errata

Přestože jsme udělali maximum pro to, abychom zajistili přesnost a správnost obsahu, chybám se úplně vyhnout nedá. Pokud v některé z našich knih najdete chybu, ať už chybu v textu nebo v kódu, budeme rádi, pokud nám ji nahlásíte. Ostatní uživatelé tak můžete ušetřit frustrace a pomoci nám zlepšit následující vydání této knihy.

Veškerá existující errata zobrazíte na adrese <http://knihy.cpress.cz/k1868> po klepnutí na odkaz Soubory ke stažení.

Kapitola 1

Než začneme

Předtím, než naše statické stránky předěláme do dynamické podoby, bychom si měli říct, co to vlastně dynamické stránky jsou, co všechno umožňují nám jako tvůrcům a co návštěvníkům našich stránek. V této kapitole si také připravíme a nainstalujeme všechno, co budeme později při psaní a ladění dynamických stránek potřebovat.

Dynamické vs. statické stránky

Základní rozdíl mezi statickými a dynamickými stránkami je, že statické stránky jsou neměnné. Celý web je „vyskládan“ z mnoha souborů, přičemž každý z nich obsahuje HTML soubor, ve kterém je kompletně celý zdrojový kód dané podstránky.

Dynamické weby však mají v jednotlivých souborech pouze části celkové stránky (jeden soubor obsahuje zdrojový kód hlavičky, další patičky, další soubor třeba kontaktní formulář a tak dále). Výsledná stránka, kterou pak uživatel vidí na svém monitoru, je vyskládaná z těchto částí.



Obrázek 1.1 Statický web má celé stránky v jediném souboru, a pokud máme stránek více, je obtížné změnit třeba menu na každé z nich. Dynamické stránky jsou však vyskládány jako mozaika z několika souborů, a tak se změny v jediném souboru projeví na všech stránkách celého webu.

Tento přístup má výhodu v tom, že pokud máme menu v samostatném souboru a chceme ho změnit, stačí, když tak učiníme pouze v jediném souboru. Změněné menu se pak jako část skládačky použije na celém webu a na každé podstránce, kterou vidí uživatel ve výsledku na svém monitoru. Tento přístup použijeme i v našem ukázkovém projektu, hned v příští kapitole.

Dynamické stránky však umožňují daleko více než jen snadné úpravy a údržbu. Jejich další vlastností je, že jsou většinou propojené s databází a mohou měnit svůj obsah podle toho, co konkrétní uživatel vyžaduje.

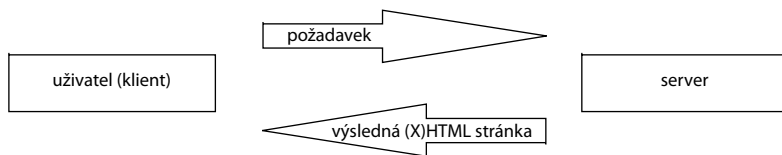
Dobrym příkladem je vyhledávání. Každý uživatel bude na našich stránkách hledat něco jiného, ale nám stačí napsat pouze jeden skript, který projde databází a ukáže uživateli to, co chtěl najít na stránce s vyhovujícími články. A to vše dynamicky.

Jak přesně dynamické stránky fungují, si vysvětlíme dále.

Co je a k čemu slouží PHP

PHP je open source skriptovací jazyk, který se používá k vytvoření dynamických stránek. Dá se snadno naučit, a přitom má široké možnosti využití: můžeme ho použít pro malé a jednoduché osobní stránky nebo blogy, ale také pro elektronické obchody či prezentaci velké společnosti.

PHP jazyk se používá pro programování aplikací na bázi klient–server (obrázek 1.2).



Obrázek 1.2 Schéma fungování aplikací na bázi klient–server

1. Uživatel někde na Internetu klepne na odkaz vedoucí na naši stránku nebo její adresu napíše do řádku pro URL adresu a stiskne klávesu Enter.
2. Uživatelův prohlížeč odešle požadavek serveru, na kterém jsou naše stránky hostovány. Požadavek obsahuje informaci o tom, kterou stránku si chce uživatel zobrazit.
3. Server spustí náš skript, který dynamicky vygeneruje požadovanou stránku. Vybere z databáze požadované informace a spojí části stránky (hlavička, menu, hlavní text, patička) do jediného (X)HTML zdrojového kódu výsledné stránky. Tento kód už vypadá jako kód běžné statické stránky – jen byl vytvořen dynamicky.
4. Tento (X)HTML kód je poslán prohlížeči uživatele, který jej zobrazí a vykreslí, tak jako běžnou statickou stránku.

Jak je vidět, podstata je v tom, že všechno „dynamické“ se děje na straně serveru, který na základě požadavku od uživatele vygeneruje statickou stránku a tu pošle prohlížeči uživatele.

Výhody tohoto klient-server přístupu jsou v tom, že všechny výpočty, dotazy do databáze a generování kódu (skládání mozaiky) probíhají na serveru (což je vlastně velmi výkonný počítač) a uživateli se posílá statický zdrojový kód, který prohlížeč pouze zobrazí – tak jako každou jinou statickou stránku.

Využití skriptů napsaných v jazyce JavaScript

Součástí dynamických stránek jsou často i skripty v jazyce JavaScript. Tyto skripty se však posílají spolu s (X)HTML kódem prohlížeči uživatele a tento prohlížeč teprve vykonává instrukce obsažené v uvedených skriptech. JavaScript neběží na straně serveru jako PHP, ale až na straně uživatele.

V praxi se obvykle využívá kombinace těchto dvou jazyků. PHP skript na serveru vygeneruje statickou stránku třeba s fotogalerií i s vloženým javascriptovým kódem. Výsledek se pošle prohlížeči uživatele. JavaScript pak umožní, že fotogalerie už není jen nudný, statický „výčet“ fotek.

Taková fotogalerie pak vypadá velmi efektně díky tomu, že jednotlivé fotky mohou dynamicky vystoupit do popředí, různě rolovat a podobně. Právě díky tomu, že JavaScript běží na straně uživatele, se nezatěžuje server, na kterém jsou naše stránky hostovány.

JavaScript však umožňuje mnohem více než jen vytvoření efektních fotogalerií. Díky němu je možné měnit a aktualizovat některé části (bloky) stránky, aniž by bylo nutné znovu načítat celou stránku. To však není obsahem naší knížky, proto se tomuto tématu dál věnovat nebudeme.

Instalace potřebného softwarového vybavení

Abychom se mohli naučit programovat dynamické stránky v jazyce PHP, musíme si na náš počítač nainstalovat webový server. Slouží k tomu, abychom přímo na našem vlastním počítači mohli spouštět PHP skripty a vytvářet tak naše dynamické stránky. To je nezbytné, abychom nemuseli naše stránky testovat a vyvíjet na reálném, vzdáleném serveru, kde už běží naše statické stránky. Bylo by totiž hodně nepraktické při každé změně kopírovat upravené soubory na vzdálený server. Nemluvě o tom, že pokud by byl skript špatný (třeba se zacyklil do nekonečné smyčky), zbytečně by server zatěžoval. I v praxi se webové aplikace vyvíjí na vlastním počítači a na server se nasazuje už otestovaná aplikace.

Instalace webového serveru

Jako u téměř každého softwaru, i zde existuje více rozdílných možností. My si nainstalujeme Xampp server, protože se instaluje velice snadno, obsahuje webový Apache i databázový MySQL server (využijeme ho později) a má pohodlné ovládání. Je to open source freeware řešení, takže si nemusíme nic kupovat.



Poznámka: Pro instalaci budeme potřebovat administrátorská práva, takže se musíme přihlásit jako administrátor.

Existuje několik možností, jak si tento server nainstalovat. Záleží také na tom, jaký operační systém používáme. Všeobecně platí, že si nejdříve musíme otevřít stránku <http://www.apachefriends.org/en/xampp.html>, na které najdeme vše potřebné.

Instalace ve Windows pomocí instalátoru (doporučujeme):

1. Xampp server je podporován prakticky všemi dnes používanými operačními systémy od společnosti Microsoft: Windows 2000, Windows XP, Windows Vista i Windows 7.
2. Klepneme na možnost **Xampp for Windows** a dostaneme se na stránku <http://www.apachefriends.org/en/xampp-windows.html>.
3. Rolujme stránku níže, až dojdeme k bloku **Download**. V době psaní této knihy je aktuální verze **XAMPP Windows 1.7.4**.
4. Klepneme na možnost **Installer** a stáhneme si jej. Měl by mít velikost kolem 66 MB.
5. Po stáhnutí instalátoru na něj poklepeme a spustíme ho.
6. Vybereme si jazyk **English**.
7. V dalším kroku pouze klepneme na **Next**.
8. Vybereme si cílový adresář na disku, do kterého chceme server nainstalovat. Doporučujeme si zvolit adresář **C:\xampp**, protože dále v knize budeme používat právě toto umístění. Instalace serveru vyžaduje asi 460 MB volného místa na disku. Klepneme na **Next**.
9. V dalším kroku můžeme označit, jaké ikony chceme vytvořit. Možnosti **Create a XAMPP desktop icon** a **Create an apache friends xampp folder in the start menu** doporučujeme nechat zaškrtnuté.



Poznámka: Možnosti **Install apache as service** a **Install mysql as service** můžeme zaškrtnout také. Znamenají, že chceme, aby se webový server a server pro databáze nainstalovaly jako služby operačního systému. Pokud je tak nainstalujeme, budou se automaticky spouštět již při startu počítače. Tato možnost je vhodná, pokud se programování stránek budete věnovat často. Nebudete tak muset startovat tyto servery ručně pokaždé, když je budete potřebovat.

10. Klepněme na **Install**.
11. Pokud instalace proběhla správně, zobrazí se okno s tlačítkem **Finish**. Klepneme na něj.
12. Pokud vyskočilo okno *Firewall*, klepneme na možnost **Unblock** (neblokovat).
13. Instalace je dokončená.

Instalace ve Windows bez instalátoru:

Pokud jste zkušenější uživatel, můžete si Xampp server nainstalovat i bez instalátoru. Ze stránky <http://www.apachefriends.org/en/xampp-windows.html> si v sekci **Download** stáhněte ZIP archiv. Rozbalte ho do adresáře podle svého výběru. Poté spusťte soubor *setup_xampp.bat* a server si nakonfigurujte podle instrukcí na obrazovce.



Důležité: Pokud si vyberete jako cílový adresář **C:**, nemůžete spustit soubor *setup_xampp.bat*. Způsobovalo by to problémy.

Instalace v Linuxu:

1. Stáhneme si nejnovější verzi ze stránky <http://www.apachefriends.org/en/xampp-linux.html>.
2. V terminálovém okně zadáme příkaz `su` a přihlásíme se jako **root**.
3. Spustíme následující příkaz: `tar xvfz xampp-linux-1.7.4.tar.gz -C /opt`.



Poznámka: Verze 1.7.4. je aktuální v době psaní této knihy. Pokud jste si stáhli novější verzi, pozměňte příkaz tak, aby vyhovoval názvu staženého archivu.



Důležité: Používejte pro instalaci pouze tento příkaz. Na rozbalení archivu nepoužívejte žádné jiné nástroje, jinak se instalace nezdaří.



Důležité: Po spuštění tohoto příkazu se přepíše původní (pokud byla) instalace/verze serveru Xampp.

Pro nastartování serveru spustíme příkaz `/opt/lampp/lampp start`.

Pokud je na obrazovce následující text, server se úspěšně nastartoval:

```
Starting XAMPP 1.7.4...
LAMPP: Starting Apache...
LAMPP: Starting MySQL...
LAMPP started.
```

Teď už zbývá jen zkontrolovat správnost instalace.

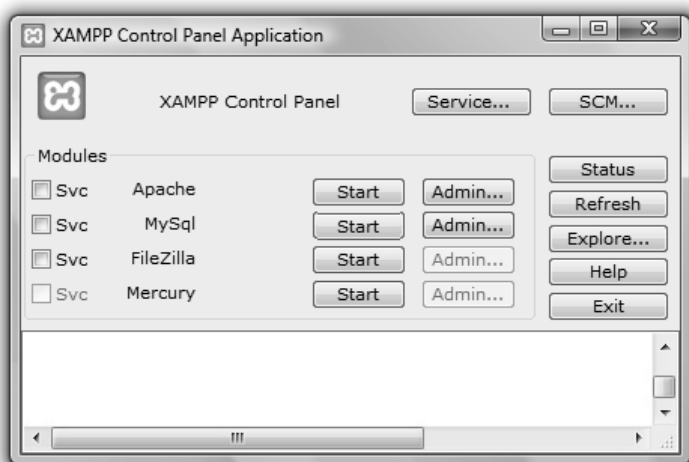
Spuštění serveru Xampp

Pokud jsme server Xampp nenainstalovali jako službu, a není tudíž ještě spuštěný, klepneme na ikonu **Xampp Control Panel**. Pokud ji nemáme na ploše, vyhledejme ji v nabídce **Start** → **XAMPP for Windows** → **Xampp Control Panel**. Otevře se ovládací panel serveru (jako na obrázku 1.3).

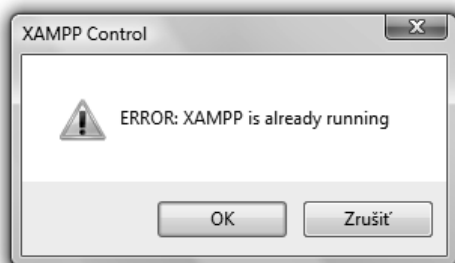


Řešení problému: Pokud systém při pokusu o zobrazení kontrolního panelu vypisuje hlášku jako na obrázku 1.4 s textem **ERROR: XAMPP control panel is already running** (XAMPP control panel už běží), zkuste si tento kontrolní panel zobrazit z nabídky běžících programů, kterou najdete v pravém dolním rohu obrazovky (jako na obrázku 1.5). Xampp Control Panel má oranžovou ikonku s velkým písmenem X obtaženým bílou barvou.

Pokud v levé části okna vedle popisku **Apache** nesvítí zelený obdélník s textem **Running**, klepneme na tlačítko **Start** ve stejném řádku, jako je popisek **Apache**. Totéž můžeme udělat i s druhým řádkem pro **MySQL**. Pokud jsou oba servery spuštěny, svítí vedle jejich názvů dva zelené obdélníky s textem **Running** (viz obrázek 1.6).



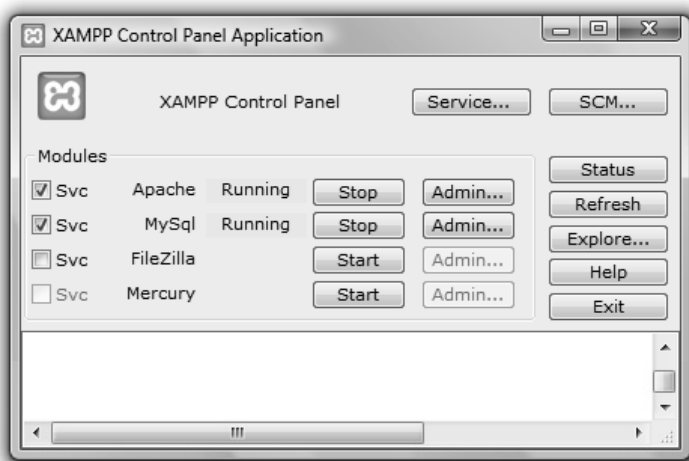
Obrázek 1.3 Xampp Control Panel



Obrázek 1.4 Chybová hláška: kontrolní panel serveru Xampp už běží



Obrázek 1.5 Zobrazení kontrolního panelu serveru Xampp z nabídky běžících programů

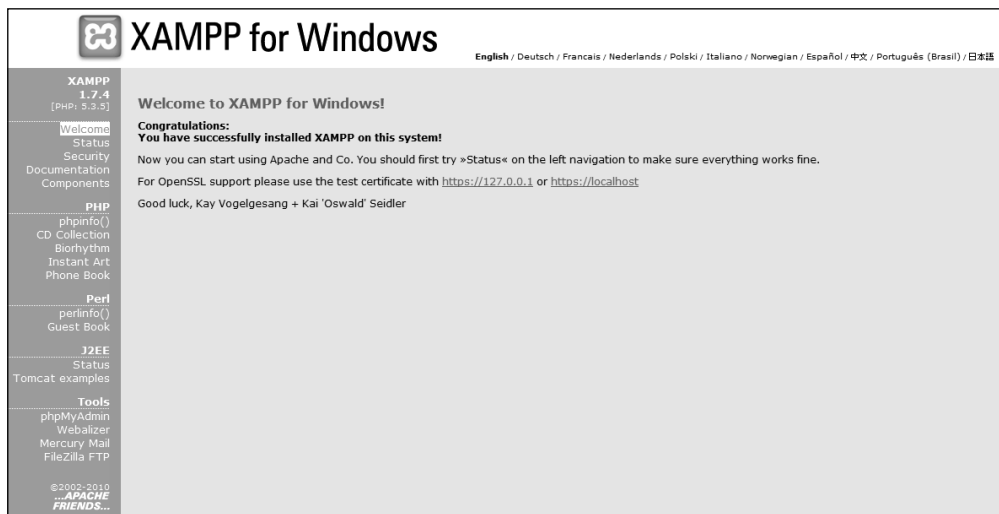


Obrázek 1.6 Xampp Control Panel ve stavu, kdy běží webový (Apache) i databázový (MySQL) server

Kontrola správné instalace

Pokud už máme nainstalovaný a spuštěný webový server, můžeme zkontrolovat jeho funkčnost. Otevřeme si svůj oblíbený internetový prohlížeč a v něm adresu <http://localhost/>.

Pokud se zobrazila stránka s velkým logem Xampp uprostřed a možností volby jazyka, máme server nainstalován správně a také správně běží. Klepneme na možnost **English** a dostaneme se na další stránku. Měli bychom vidět hlášku: **Congratulations: You have successfully installed XAMPP on this system!**, která potvrzuje správnou instalaci. Obrazovka by teď měla vypadat jako na obrázku 1.7.



Obrázek 1.7 Stav obrazovky po úspěšné instalaci a nastartování webového serveru

Můžeme si zobrazit také stránku <http://localhost/phpmyadmin/>. Na této adrese se zobrazí uživatelské rozhraní pro práci s databázemi. Těm se budeme věnovat později.



Řešení problému: Pokud se na vašem počítači nezobrazila hláška o úspěšném zprovoznění webového serveru, může to být způsobeno hlavně těmito skutečnostmi:

- Pokud na obrazovce máte hlášku prohlížeče „**Stránka nenalezena**“, zkontrolujte, jestli máte nastartován webový server, podle odstavce **Spuštění Xampp serveru** výše v této kapitole.
- Pokud server nejde nastartovat nebo se ani nezobrazí **Xampp Control Panel**, zkuste celý server přeinstalovat.
- Pokud **Xampp Control Panel** vypadá jako na obrázku 1.6 – vedle řádku s textem **Apache** je zelený obdélník s textem **Running** – zkontrolujte, jestli se pokoušíte zobrazit správnou stránku: <http://localhost/> nebo <http://127.0.0.1/>.
- Pokud nic nepomáhá, zkuste server znovu přeinstalovat.

Zkopírování zdrojových kódů ke knize na lokální webový server

Teď si nakopírujeme zdrojové kódy ke knize na náš právě vytvořený lokální webový server, abychom mohli sledovat všechny postupy a řešení, tak jak budou popsány v knize.

Stáhneme si zdrojové kódy dostupné ke knize z webové stránky <http://knihy.cpress.cz/k1868> a rozbalíme je. Otevřeme si adresář `C:\xampp\htdocs\`. Vytvoříme si nový adresář *kniha* a všechny soubory, které jsme si před chvílí stáhli, nakopírujeme do tohoto adresáře. Kódy k jednotlivým kapitolám najdeme v příslušném podadresáři – např. pro tuto první kapitolu jsou v adresáři `C:\xampp\htdocs\kniha\01\`.

Adresářová struktura a přístup k souborům

Ze začátku bude možná trochu matoucí, jak najít požadovaný soubor na disku (třeba pro editaci) a jak zapsat jeho adresu do prohlížeče tak, aby webový server vykonal instrukce skriptu zapsané v daném souboru.

Pri práci se soubory a skripty budeme postupovat následovně:

- Pokud budeme chtít najít soubor na disku (třeba pro editaci), hledejme ho v adresáři např. takto: `C:\xampp\htdocs\kniha\01\index.html`.
- Pokud si budeme chtít tento soubor zobrazit v prohlížeči (tak aby webový server vykonal skript, který soubor obsahuje), zapišme do něj adresu `http://localhost/kniha/01/index.html`.

Jediný rozdíl v přístupu k souboru pak spočívá v začátku cesty k němu: buď `C:\xampp\htdocs\...`, nebo `http://localhost/...`. Záleží na tom, co s tímto souborem chceme dělat.



Poznámka: Heslo pro uživatele **root** pro přístup k databázi můžete nastavit na stránce <http://localhost/security/xamppsecurity.php>. Podrobnější informace o bezpečnostních nastaveních pak naleznete na stránce <http://localhost/security/>. Pro naše výukové účely a později i pro vývoj vlastních stránek a aplikací na lokálním počítači tato nastavení nemusíte měnit.

Omezená bezpečnost Xampp serveru

Xampp server je určen téměř výhradně pro vývojáře webových aplikací na jejich lokálních počítačích a tomu odpovídají i jeho nastavení. Je konfigurován jako co nejvíce otevřený a s minimem bezpečnostních omezení, proto je nevhodný pro produkční (ostrý) server. Je navržen tak, aby umožňoval vše, co návrhář potřebuje.

Za všechny bezpečnostní hrozby uveďme alespoň jednu. Pro přístup k databázi se používá standardní uživatel (root), který nemá nastavené žádné heslo. Heslo pro tohoto uživatele se sice dá nastavit, ale pořád zbývají ostatní hrozby.

Proto Xampp server využívejte pouze na svém lokálním počítači a jen pro vývoj webových stránek a aplikací. Pro hotové stránky, které pak budete chtít „zavěsit na Internet“, využijte služby poskytovatelů webhostingu, kteří se poskytování těchto služeb věnují. Pokud byste chtěli provozovat vlastní webový sever, rozhodně budete potřebovat nastudovat více informací.

Textové editory se zvýrazněním syntaxe

HTML stránky i skripty jazyka PHP lze psát prakticky v libovolném textovém editoru (třeba i v Poznámkovém bloku nebo editoru Vi), ale existuje obrovské množství editorů, jež umožňují zvýrazňování a některé i doplňování značek kódu konkrétního jazyka. Využít můžete také velké množství placených, ale rovněž freeware editorů, které se dost liší kvalitou i uživatelskou přívětivostí.

Pro jakéhokoli vývojáře nebo programátora je velmi vhodné pracovat s editorem, který dokáže zvýraznit (a třeba i doplnit) značky daného jazyka.

Pro nás je výhodné, když jsou značky HTML kódu zvýrazněné jinou barvou jako proměnné jazyka PHP a když jsou navíc ještě odlišené jinou barvou třeba od funkcí tohoto jazyka. Velmi příjemnou vlastností je také zvýrazňování začátku a konce bloků mezi závorkami / kulatými () i složenými {}, třeba při cyklech nebo podmínkách jazyka PHP. Častá je také funkce automatického odsazování a zpřehlednění zdrojového kódu.

Všechny tyto funkce editoru výrazně usnadňují práci programátorům a vývojářům, kteří rychleji najdou chybu ve svém skriptu nebo se v něm výrazně rychleji zorientují.

Instalace textového editoru pro skripty

Pro naše účely je velmi vhodný PSPad editor, který je freeware.

1. Otevřeme si stránku <http://www.pspad.com/cz/>.
2. Klepneme na možnost **Stažení PSPadu** v levém menu.
3. Stáhneme si *Instalátor*. Je to malý soubor (asi 3,5 MB), v době psaní knihy je aktuální verze 4.5.4.
4. Poté na *Instalátor* poklepeme a zahájíme proces instalace.
5. Na první kartě pouze klepneme na **Další**.
6. Na druhé kartě musíme souhlasit s licenčními podmínkami, poté klepneme na **Další**.
7. V třetím kroku si vybereme, kam chceme editor nainstalovat. Cílový adresář nemusíme (ale můžeme) změnit. Klepneme na **Další**.
8. V dalším kroku si vybereme typ instalace. Klidně můžeme ponechat možnost **Plná instalace**, poněvadž zabere pouze necelých 12 MB prostoru na disku. Klepneme na **Další**.
9. V tomto kroku vytvoříme zástupce v nabídce **Start**. Opět můžeme ponechat beze změn. Klepneme na **Další**.
10. Na následující kartě si zvolíme další požadované úlohy podle toho, co budeme od editoru požadovat. Můžeme je nechat nezměněné, některé možnosti lze ubrat nebo také zaškrtnout položku **Otevírat soubory TXT programem PSPad**. Klepneme na **Další**.
11. Klepneme na **Instalovat**.
12. Instalace je ukončená, klepneme na **Dokončit**.

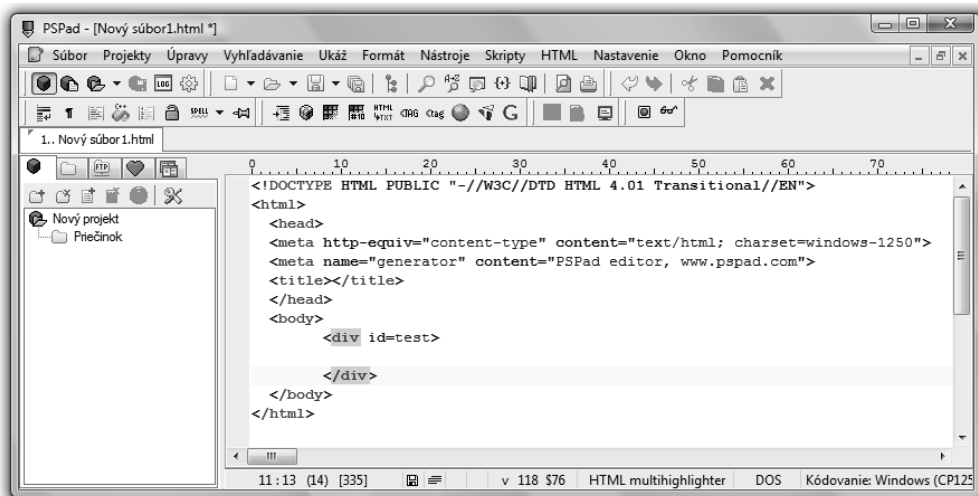
Jak k práci využít textový editor pro skripty

Spustíme si teď nově nainstalovaný editor – buď klepnutím na ikonku PSPad na ploše, nebo po vyhledání v nabídce **Start** → **PSPad**.

Klepneme na příkaz **Soubor** → **Nový**. Otevře se okno, ve kterém je poměrně dlouhý seznam podporovaných typů souborů. Vyberme si třeba HTML multihighlighter. Otevře se nový soubor a v něm připravená základní šablona. Napišme mezi značky `<body>` a `</body>` následující kód:

```
<div id="test">
    testovací text
</div>
```

Teď klepneme do značky `</div>`. Jak vidíme na obrázku 1.8, značka `<div id="test">` a k ní patřící uzavírací značka `</div>` se podbarví modře. Zvýrazňování syntaxe je velmi užitečné, a když si na něj zvyknete, pravděpodobně už nebudete chtít používat editor bez této funkce.



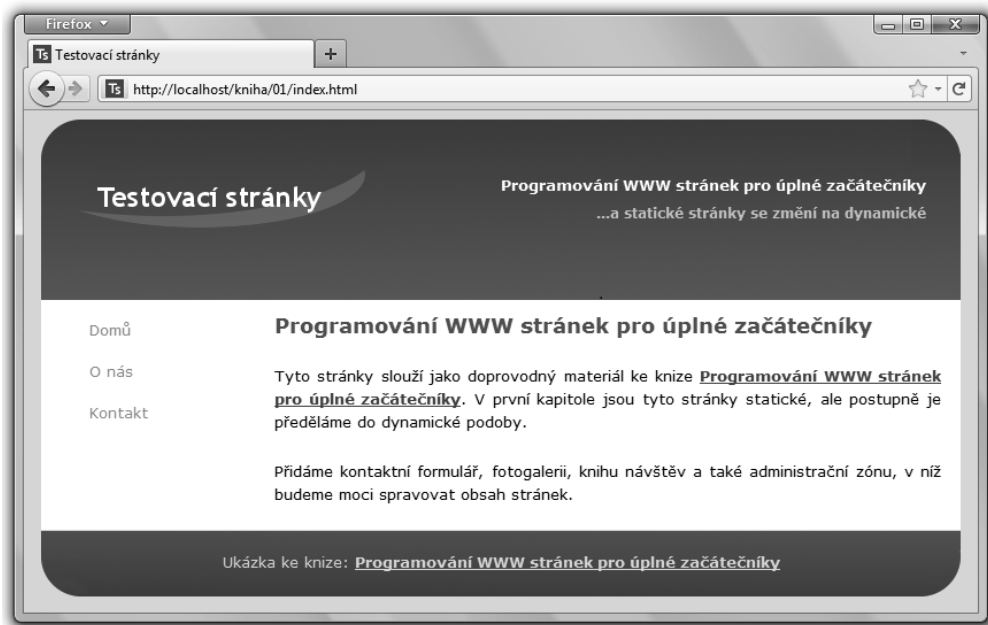
Obrázek 1.8 Zvýrazňování syntaxe ulehčuje práci vývojářům

Výchozí podoba ukázkového webu

Podívejme se na ukázkové testovací stránky, které postupně předěláme do dynamické podoby. Najdeme je v adresáři /01/ mezi zdrojovými kódy doprovázejícími tuto knihu. Pokud jste si server nainstalovali stejně jako já, je cesta k těmto stránkám `C:\xampp\htdocs\kniha\01\`. Jak vidíme, v tomto adresáři máme:

- tři statické stránky: *index.html*, *o-nas.html* a *kontakt.html*,
- adresář obrázky, ve kterém jsou obrázky použité na těchto stránkách,
- a adresář *css*, v němž je jenom jeden soubor *style.css*, který popisuje vzhled stránek.

Otevřeme si v prohlížeči soubor *index.html*. Měl by vypadat jako na obrázku 1.9.



Obrázek 1.9 Soubor *index.html* zobrazený v prohlížeči

Teď si rychle prohlédneme zdrojový kód souboru *index.html*, abychom přesně poznali stránky, které budeme předělávat do dynamické podoby.

Nejdříve definujeme typ souboru, kódování, titulek stránky, připojíme soubor s definicí stylu, ikonku, informace pro roboty procházející stránky na Internetu a prozatím prázdné značky pro metapopis a klíčová slova:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <meta http-equiv="Content-Type" content="text/html;
      charset=utf-8" />
    <title>Testovací stránky</title>
    <style type="text/css" media="all">
      @import "css/style.css";
    </style>
    <link rel="shortcut icon" href="obrazky/favicon.ico" />
    <meta name="robots" content="index,follow" />
    <meta name="description" content="" />
    <meta name="keywords" content="" />
  </head>
```

Teď následuje samotné tělo stránky. Uspořádání jednotlivých bloků ukazuje obrázek 1.10.



Obrázek 1.10 Uspořádání bloků ukázkové stránky

Celá stránka je zabalená v bloku `obalStranky`. Nahoře je hlavička, která obsahuje dva bloky – jeden vlevo pro logo a druhý vpravo pro hlavní nadpis a podnadpis stránek.

```
<body>
  <div id="obalStranky">
    <div id="hlavicka">
      <div id="logo">
        <a href="index.html" />
          
        </a>
      </div>
      <div id="nadpisy">
        <h1>Programování WWW stránek pro úplné
        začátečníky</h1>
        <h2>...a statické stránky se změni na
        dynamické</h2>
      </div>
    </div>
  </div>
```

Střední část stránky obsahuje také dva bloky. Vlevo je sloupec pro menu a případně další malé bloky. Vpravo se zobrazuje samotný obsah stránek – hlavně texty.

```
<div id="stred">
  <div id="levySloupec">
    <div id="menu">
      <a href="index.html">
        <div class="menuLink">Domů</div>
      </a>
      <a href="o-nas.html">
```

```

        <div class="menuLink">O nás</div>
    </a>
    <a href="kontakt.html">
        <div class="menuLink">Kontakt</div>
    </a>
</div>
</div>
<div id="obsah">
    <h3>Programování WWW stránek pro úplné začátečníky</h3>
    <p>Tyto stránky slouží jako podpůrný materiál ke knize
    <a href="http://knihy.cpress.cz/k1868" target="_blank">
    <strong>Programování WWW stránek pro úplné
    začátečníky</strong></a>. V první kapitole jsou tyto
    stránky statické, ale postupně je předeíláme do
    dynamické podoby.</p>
    <p>Přidáme kontaktní formulář, fotogalerii, knihu
    návštěv a také administrační zónu, v které budeme
    umět spravovat obsah stránek.</p>
</div>
<div style="clear:both;"></div>
</div>

```

Nakonec přidáme patičku, zavřeme párové značky a stránka je hotová.

```

    <div id="paticka">
        Ukázka ke knize: <a href="http://knihy.cpress.cz/k1868"
        target="_blank"><strong>Programování WWW stránek
        pro úplné začátečníky</strong></a>
    </div>
</div>
</body>
</html>

```

Tím jsme dokončili první kapitolu a můžeme se pustit do další, v níž se budeme věnovat dynamickému zobrazení obsahu webové stránky.

Kapitola 2

Dynamické zobrazování obsahu



Jak jsme si již řekli, mezi základní vlastnosti dynamických stránek patří možnost dynamického generování výsledného kódu, který se odesílá uživateli. Také jsme už zmínili, že jednotlivé části stránky jsou obvykle uloženy v samostatných souborech. Tento přístup má několik výhodných vlastností:

- Každou část stránky (hlavička, menu, patička) máme uloženu v samostatném souboru, a je tudíž velice snadné udělat jakoukoliv změnu.
- Změnu stačí udělat v jediném souboru a ta se projeví na všech stránkách, kde se tento soubor používá jako část výsledného zdrojového kódu (mozaiky).
- Díky tomuto dělení jsou i samotné soubory kratší a přehlednější.
- Při vývoji se tak snadněji hledají chyby a později provádějí změny a úpravy a doplňují nové funkce.

Přetypování statických souborů

Prvním krokem k dynamickému webu je přetypování souborů z typu *.html na typ, ve kterém jsou skripty jazyka PHP – *.php. Je to nezbytné, aby server věděl, jak má k souboru přistupovat – aby vykonal skripty zapsané v souboru a nebral je pouze jako statickou stránku.



Poznámka: Všimněte si adresáře `C:\xampp\htdocs\kniha\ts\`. Ts odpovídá zkratce Testovací stránky. V tomto adresáři jsou stejné soubory, jako byly v adresáři `C:\xampp\htdocs\kniha\01\`. Soubory jsou v *.html formátu.

Přetypování souborů je velmi jednoduché. Následující změny budeme vykonávat na souborech v adresáři `C:\xampp\htdocs\kniha\ts\`.

1. Otevřeme si soubor `index.html`. Můžeme použít editor PSPad, Poznámkový blok nebo jiný textový editor. Je to běžná statická stránka.
2. Klepneme na možnost **Soubor** a **Uložit jako**.
3. Otevře se nové okno, jak je vidět na obrázku 2.1. V nejspodnějším řádku (možnost **Uložit ve formátu**) zvolíme možnost **Všechny soubory *.***. V řádku pro **Název souboru** napíšeme `index.php`. Klepneme na **Uložit**.