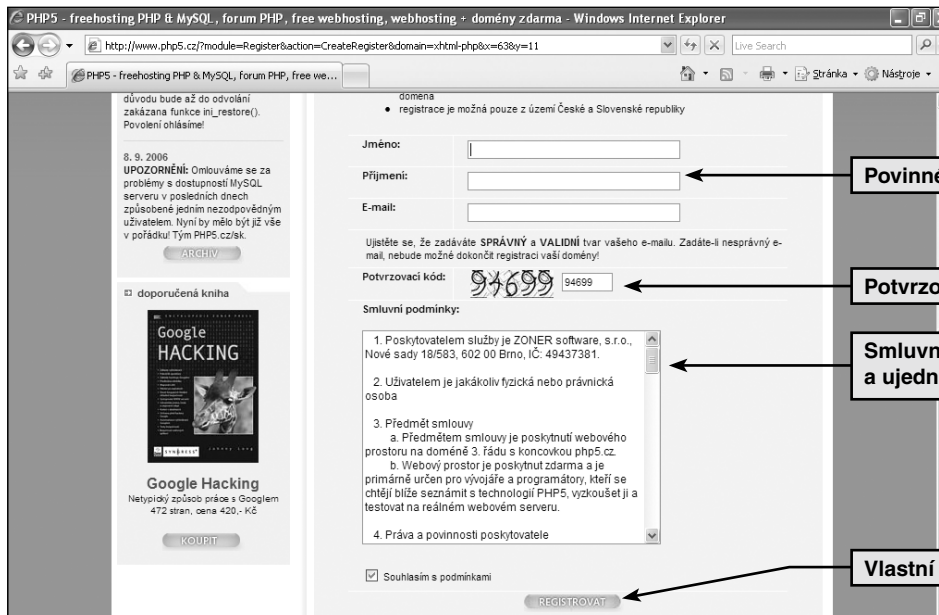
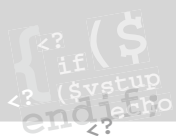


<b>1. Základní kostra HTML dokumentu</b>	<b>1</b>
<b>2. Členění dokumentu</b>	<b>2</b>
<b>3. Kaskádové styly - CSS</b>	<b>3</b>
<b>4. Pozadí stránky</b>	<b>4</b>
<b>5. Vkládání obrázků</b>	<b>5</b>
<b>6. Orámování objektů</b>	<b>6</b>
<b>7. Tabulky</b>	<b>7</b>
<b>8. Odkazy</b>	<b>8</b>
<b>9. Index, menu, submenu</b>	<b>9</b>
<b>10. Publikování na internetu</b>	<b>10</b>
<b>11. Úvod do PHP</b>	<b>11</b>
<b>12. Include</b>	<b>12</b>
<b>13. Validace stránek</b>	<b>13</b>
<b>14. Formátování textu pomocí CSS</b>	<b>14</b>
<b>15. Vlastní ikona webové stránky</b>	<b>15</b>
<b>16. Seznamy s odrážkami</b>	<b>16</b>
<b>17. Číslované seznamy</b>	<b>17</b>
<b>18. Formuláře</b>	<b>18</b>
<b>19. Datum a čas</b>	<b>19</b>
<b>20. Funkce</b>	<b>20</b>
<b>21. Proměnné</b>	<b>21</b>
<b>22. Získání proměnné z formuláře</b>	<b>22</b>
<b>23. Funkce mail()</b>	<b>23</b>
<b>24. Switch</b>	<b>24</b>
<b>25. If</b>	<b>25</b>
<b>26. Cykly</b>	<b>26</b>
<b>27. Generování náhodných čísel</b>	<b>27</b>
<b>28. Session - relace</b>	<b>28</b>
<b>29. Práce s řetězcí</b>	<b>29</b>
<b>30. Databáze MySQL</b>	<b>30</b>
<b>31. Zasílání novinek</b>	<b>31</b>
<b>32. Heslovaný přístup</b>	<b>32</b>



Bude-li váš návrh přijat, vyplňte na následující obrazovce jméno, příjmení, platnou e-mailovou adresu, na kterou vám budou zaslány přístupové údaje, a potvrzovací kód. Nezapomeňte si přečíst smluvní podmínky a potvrdit souhlas. Nakonec znovu klepněte na tlačítko **REGISTROVAT**.



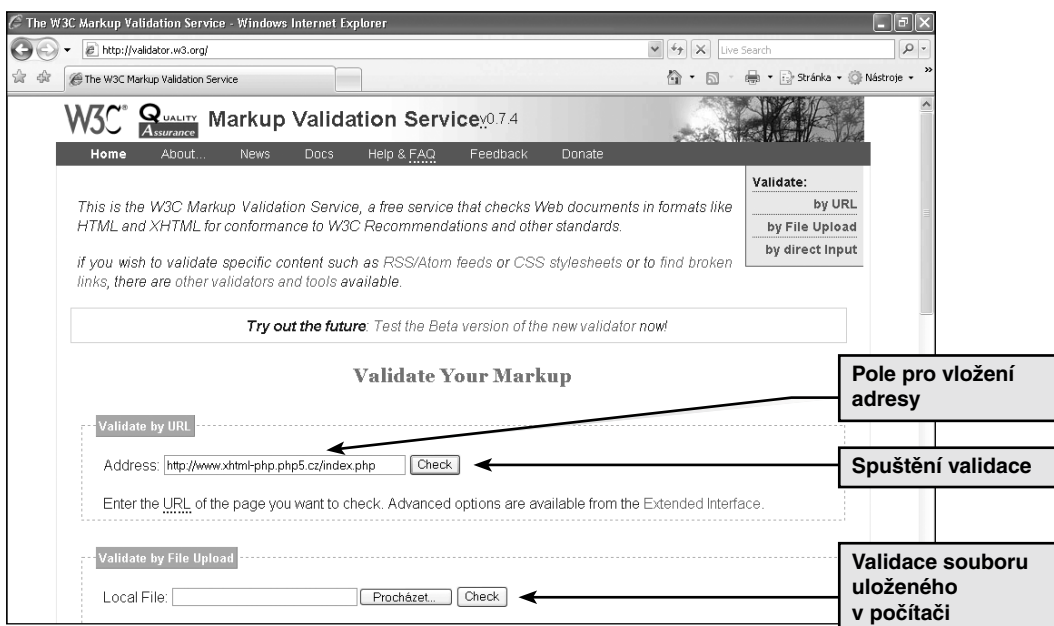


## 13. Validace stránek

Prohlížeče jsou velmi tolerantní k různým chybám, avšak správný webmaster by se měl snažit o naprostou čistotu svého kódu. Zaručí se tím mimo jiné funkčnost stránek v různých prohlížečích. Přes veškerou snahu se tu a tam nějaká chyba může vloudit. Abyste měli jistotu, že vaše stránky vyhovují všem pravidlům definovaným v DTD (ke kterým jste se zavázali specifikací dokumentu popsanou v úvodní kapitole), můžete je nechat překontrolovat validátorem.

Pravděpodobně nejlepším a nejhojněji využívaným je on-line validátor W3C, který najdete na adrese <http://validator.w3.org/>.

Do pole **Adress** ve formuláři **Validate by URL** vložte (nejlépe zkopírováním) adresu stránky, kterou chcete překontrolovat. Klepnutím na tlačítko **Check** spustíte validaci.



Pokud byste neměli stránky publikované v internetu, mohli byste nechat validovat soubor se zdrojovým kódem uložený ve vašem počítači. V takovém případě byste využili formulář **Validate by File Upload**. Po klepnutí na tlačítko **Procházet** najdete příslušný soubor a tlačítkem **Check** opět spustíte kontrolu.

**Ideálním výsledkem je potvrzení validity dané stránky.** V takovém případě obdržíte potvrzení **Passed validation** v položce **Result**. Ve velkém zeleném pruhu uprostřed stránky pak uvidíte ještě jedno potvrzení v podobě textu **This Page Is Valid XHTML 1.0 Transitional!**. Pod tímto textem je umístěn zdrojový kód validační ikonky. Pokud jej umístíte do svých stránek, může si každý uživatel ověřit jejich bezvadnost.



20. Mezi značky pro ukončení tabulky a ukončení formuláře vložte skryté pole s názvem **stat** a hodnotou **ČR**.

```
...
    </tr>
</table>
<input type="hidden" name="stat" value="ČR" />
</form>
...
```

## Odesílací tlačítko

Formulář bez odesílacího tlačítka by neměl žádný smysl, neboť by nebylo možné hodnoty z formuláře odeslat ke zpracování.

Odesílací tlačítko vložíte do formuláře značkou `<input>` s parametrem `type="submit"`. Vzhled tlačítka nemůžete nijak ovlivnit, záleží na použitém prohlížeči. Na následujících obrázcích vidíte, jak celý formulář zobrazuje **Internet Explorer** a **Opera**:

The image shows two browser windows side-by-side, both displaying the same web form titled 'Formuláře'. The form includes a sidebar with 'Výuka XHTML' and a main area with input fields for 'Uživateléské jméno:', 'Heslo:', 'Vaše zpráva:', and 'Vzdělání:'. There is also a 'Souhlasím s podmínkami' checkbox and an 'Operační systém:' dropdown. The 'Odeslat' button is highlighted in both. In the Internet Explorer window, the button is labeled 'Odeslat dotaz', while in the Opera window, it is labeled 'Odeslat'. A legend box at the bottom left explains these differences.

**Různé způsoby zobrazení tlačítka Odeslat:**

- Internet Explorer** → Odeslat dotaz
- Opera** → Odeslat

## 25. If

Jiný typ podmíněného větvení zprostředkuje příkaz **if** (jestliže, když). Příkaz (popřípadě sada příkazů) bude vykonán pouze v případě, že byla splněna podmínka. V následujícím příkladu bude vykonán pouze jeden příkaz **echo**, protože podmínka  `$\$i > 3$`  je při hodnotě  `$\$i = 5$`  splněna:

```
<?php
    $i=5;
    if ($i>3) echo "Proměnná je větší než 3";
?>
```

Pokud chcete vykonat při splnění podmínky více než jeden příkaz, uzavřete všechny tyto příkazy do složených závorek:

```
<?php
    $i=5;
    if ($i>3) {
        echo "Proměnná je větší než 3";
        $i = 2;
        echo "Ale teď je rovna".$i;
    }
?>
```

Uvedený skript vyhodnotí podmínku jako splněnou, zobrazí tedy text „Proměnná je větší než 3“. Následně proměnné  `$\$i$`  přiřadí hodnotu 2 a vypíše text „Ale teď je rovna 2“.

Příkaz **if** může mít ještě nepovinnou klauzuli **else** (jinak). Příkaz nebo blok příkazů po **else** se provede při nesplnění podmínky:

```
<?php
    $i=5;
    if ($i>5) echo "Proměnná je větší než 3";
    else echo "Proměnná není větší než 3";
?>
```

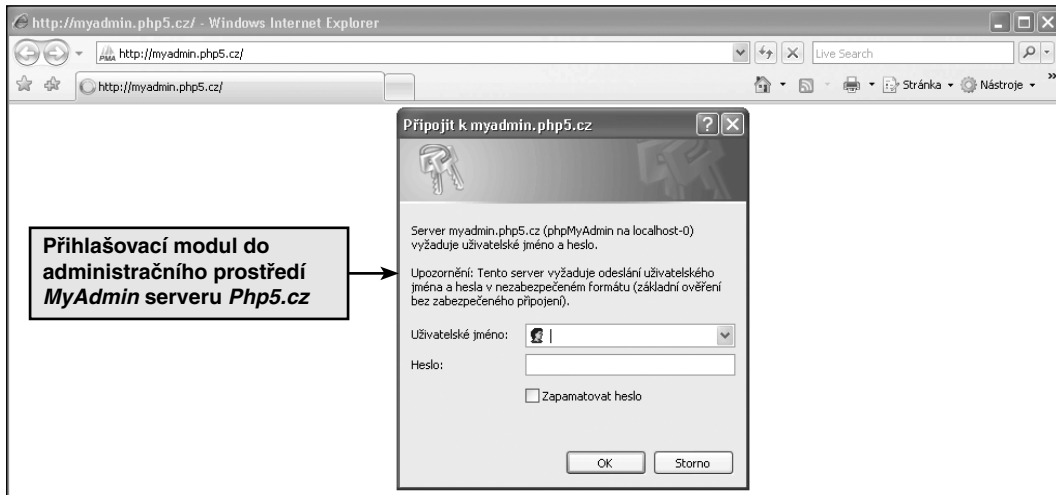
Proměnné  `$\$i$`  je přiřazena hodnota 5. Podmínka stanoví: Jestliže je proměnná  `$i$`  větší než pět, vypiš text „Proměnná je větší než 5“, jinak vypiš text „Proměnná není větší než 5“. Protože podmínka v daném případě splněna není, vypíše se text za **else**.

Několik po sobě jdoucích podmínek můžete testovat pomocí příkazu **elseif**:

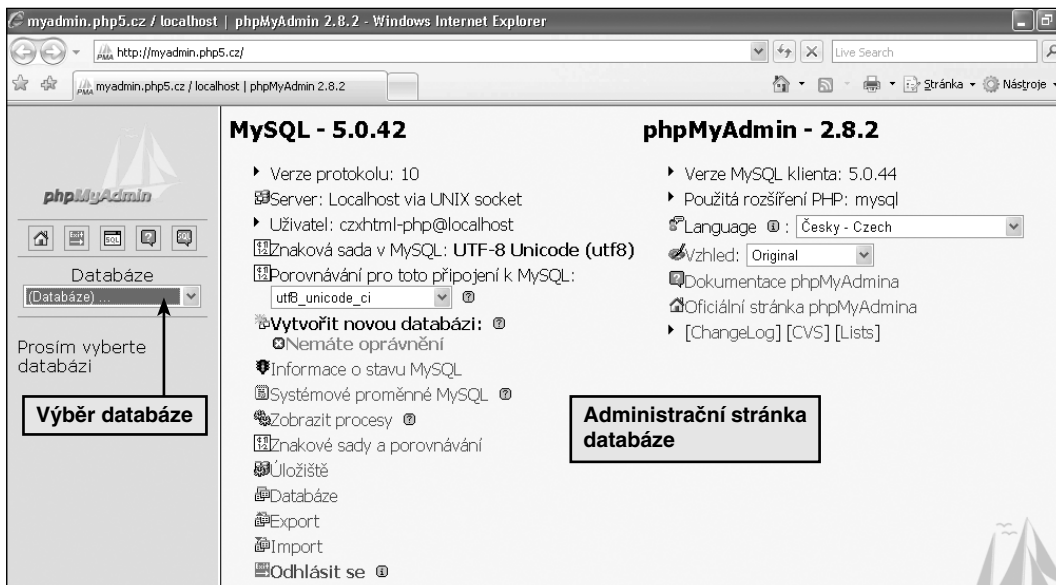
```
<?php
    $i=5;
    if ($i>3) echo "Proměnná je větší než 3";
```

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
<b>25</b>
26
27
28
29
30
31
32

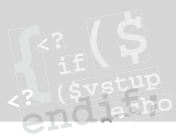
Abyste nemuseli databázi administrovat v prostředí podobném příkazovému řádku, bylo vytvořeno administrační prostředí. Pro servery **Php5.cz** je najdete na adrese <http://myadmin.php5.cz>. Po klepnutí na odkaz, případně po odeslání adresy se v prohlížeči zobrazí okno vyžadující přihlášení.



Zadáte-li chybné údaje, přístup na administrační stránku vám bude odepřen. Po vložení správných identifikačních údajů (můžete aktivovat i zaškrťovací pole **Zapamatovat heslo**) se zobrazí administrační stránka vaší databáze:

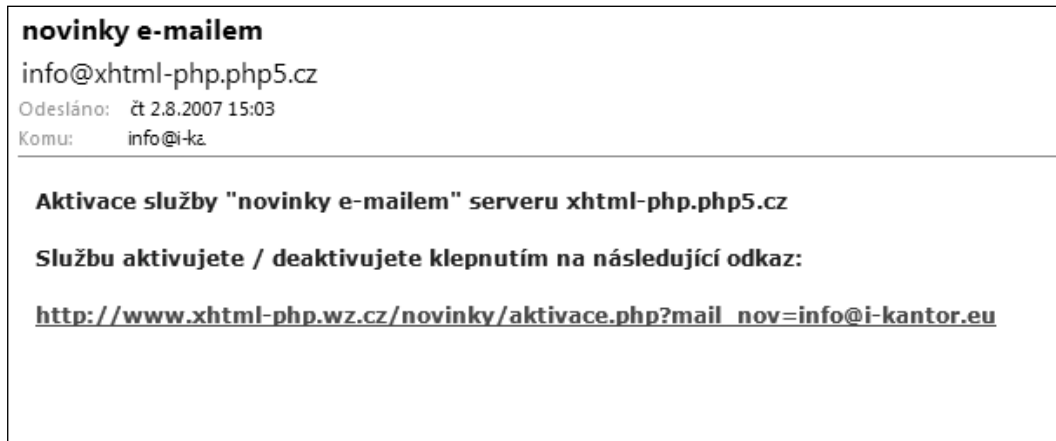


1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
<b>30</b>
31
32



```
<p>
<a href='http://www.xhtml-php.wz.cz/novinky/aktivace.php?mail_nov=$mail_
nov'>
http://www.xhtml-php.wz.cz/novinky/aktivace.php?mail_nov=$mail_nov
</a>
</p>";
```

Všimněte si především způsobu připojení proměnné a její hodnoty za adresu. Doručený e-mail má pak například následující podobu:



Stránka „aktivace.php“, na kterou bude uživatel přesměrován po klepnutí na aktivační odkaz, musí nejprve ze své adresy načíst proměnnou `mail_nov`.

```
$mail_nov=$_HTTP_GET_VARS[mail_nov];
```

Bude následovat poměrně složitá procedura v těchto krocích:

1. spojení s databázovým serverem;
2. výběr databáze;
3. prohledání tabulky, zda již obsahuje e-mailovou adresu;
4. vložení adresy do tabulky (aktivace služby), pokud adresa v tabulce neexistuje;
5. vyjmutí adresy z tabulky (deaktivace služby), pokud adresa v tabulce existuje.

## Připojení k databázovému serveru – `mysql_connect()`

Vytvořte tři proměnné, které následně využijete pro připojení k databázovému serveru. Hodnoty najdete v administrátorském prostředí ve správě databází - viz obrázek na další straně.



```
$dbhost="název serveru";           //u php5.cz "localhost"
$dbusr="uživatelské jméno";         //login
$dbpassw="přístupové heslo";      //heslo
```

Připojení k databázovému serveru provedte následujícím příkazem:

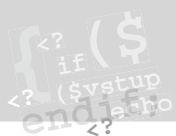
```
@$link = mysql_pconnect($dbhost,$dbusr,$dbpassw) or die("server nepřipojen");
```

Celý příkaz je možné pro větší názornost rozdělit na jednotlivé části:

- `mysql_pconnect("$dbhost","$dbusr","$dbpassw")` – naváže se spojení s databázovým serverem;
- `$link=...` – do proměnné `$link` se uloží číslo otevřeného spojení, které slouží k identifikaci spojení v ostatních funkcích PHP pro práci s MySQL;
- `@...` – nastaví tzv. tichý režim – v případě, že při připojování dojde k chybě například z důvodu, že databázový stroj je dočasně nefunkční, nebude vypsáno PHP chybové hlášení, které jednak není právě estetické, ale především by mohlo prozradit citlivé údaje o používaném databázovém serveru;
- `or die("server nepřipojen")` – funkce `die()` má sice poněkud morbidní název (umří), ale na druhou stranu přesně odpovídá svému účelu - v případě neúspěšného navázání spojení vypíše text, který je uveden v jejím parametru, a ukončí veškeré další provádění skriptu – „umře“.

Pokud si přejete být informováni i o úspěšném navázání spojení s databázovým serverem, můžete použít následující kód:

```
if(@$link = mysql_pconnect($dbhost,$dbusr,$dbpassw))
    echo "<p class='nadpis'>Spojení se serverem bylo úspěšně navázáno.</p>";
else die("<p class='nadpis'>Server nepřipojen !!!</p>");
```



již jen otázkou toho, jak dokážete nabyté znalosti využít při řešení konkrétního problému. Mnohé „pokročilejší“ funkce a příkazy vám v této publikaci byly utajeny. Protože jste však pochopili princip, nebude pro vás při řešení specifických úkolů problém tyto funkce dohledat na internetu, kde existuje řada serverů zabývajících se problematikou programování webových aplikací.

## 32. Heslovaný přístup

Ne všechny stránky musí být přístupné všem uživatelům. Určité stránky můžete podmínit například tím, že se uživatel musí nejdříve přihlásit, jiné stránky (administrátorské) jsou určeny jen pro konkrétní přihlášené uživatele. V této lekci využijete dosavadních znalostí a vytvoříte takový podmíněný přístup.

1. Ve složce **www** vytvořte kopii souboru „vzor.php“. Nově vytvořený soubor přejmenujte na „tajne.php“. Jako titulek i nadpis stránky zapište „Heslovaný přístup“.
2. Vytvořte libovolný obsah stránky.
3. V souboru „funkce.php“ vytvořte odkaz na tuto stránku.

V PHP existuje funkce **header()**, do níž lze zadat adresu stránky, na kterou se má přeměrovat. Funkce musí být bezpodmínečně použita dříve, než dojde k jakémukoliv výstupu – zobrazení čehokoliv.

```
header("HTTP/1.1 301 Moved Permanently");
header("Location: http://nova_adresa");
header("Connection: close");
```

Informace, zda je uživatel přihlášen, či nikoliv, je uložena v relaci. Následující kód stránky „tajne.php“ testuje existenci proměnné **login\_ok** v relaci. Pokud tato proměnná v relaci není, uživatel neprošel procesem přihlašování a je automaticky přeměrován na stránku „login.php“:

```
<?php
    session_start();
    if(!$_SESSION['login_ok']) {
        header("HTTP/1.1 301 Moved Permanently");
        header("Location: http://www.xhtml-php.php5.cz/user/login.php");
        header("Connection: close");
    }
    include "funkce.php";
    head();
    echo "<title>Heslovaný přístup</title>";
```



```

    menu();
    sub_php();
?>
    <h1>Heslovaný přístup</h1>
    <hr />
    <p class="nadpis">Tato stránka se zobrazí jen přihlášenému uživateli</
p>
<?php
    konec();
?>

```

4. Ve složce **www** vytvořte podsložku **user**.
5. Ve složce **user** vytvořte soubor „login.php“. Jako titulek i nadpis stránky zapište „Přihlášení uživatele“.
6. Vytvořte ve své databázi tabulku **klienti**, obsahující dva sloupce s názvy **user** a **password**. Obě proměnné budou datového typu **VARCHAR(10)**. Položku **user** definujte jako primární klíč. Vložte jeden záznam, do jména i hesla vložte například hodnotu **pokus**.
7. V souboru „login.php“ vytvořte vlastní funkci **formular()**, která zobrazí přihlašovací formulář. První pole pro zadání přihlašovacího jména pojmenujte **usr\_name**, druhé pole pro zadání hesla pojmenujte **usr\_passw**. Za formulář umístěte odkaz na soubor „user/registrace.php“ pro uživatele, kteří nejsou ve vaší databázi registrováni.

Doposud jste formuláře odesílali do jiného souboru, než byl soubor s formulářem. V tomto případě obstará zobrazení formuláře i jeho zpracování jediná stránka. Formulář musí obsahovat skryté pole se jménem a hodnotou:

```
<input type="hidden" name="frm" value="frm" />
```

Celý skript má následující logiku: „Jestliže neexistuje proměnná **\$frm**, zobraz formulář a ukončí stránku, jinak zpracuj odeslaná data.“ Skript může mít v tuto chvíli tuto podobu:

```

<?php
    session_start();
    include "../funkce.php";
    head();
    echo "<title>Přihlášení uživatele</title>";
    menu();
    sub_php();
?>

```