

Nulové okraje stránky v Internet Exploreru	3	Obsah zprávy e-mailu kliknutím na odkaz?	40
Nulové okraje stránky v Netscape Navigátoru	3	Jak otevřít nové okno prohlížeče?	41
Nulové okraje stránky v Opeře	4	Jak vytvořit horní index textu?	42
Využití atributu TITLE ve stránkách	5	Jak vytvořit dolní index textu?	42
Jak na klávesové zkratky v HTML stránce?	5	Odkaz na jiné místo v dokumentu	43
Vaše stránka jako výchozí web v prohlížeči	5	Rychlá změna velikosti písma na stránce	43
Přidání stránky do oblíbených položek	7	Formátování ALT popisku u obrázku	44
Vložení dvou nezávislých tabulek vedle sebe	7	Jak zrušit ukládání stránek do proxy?	45
Ikona vlastního webu do oblíbených položek	9	Nepohybující se "background" na stránce	45
Nahrzení standardního kurzoru Windows	11	Jak definovat bezpatkový font v HTML?	45
Nastavení barvy odkazů na HTML stránce	12	Jak definovat bezpatkový font v CSS?	46
Nastavení barvy odkazů pomocí CSS	12	Jak definovat patkový font v HTML a CSS?	46
Změna barvy odkazu při najetí myši (I.)	12	Jak definovat neproporciální font v HTML a CSS?	47
Změna barvy odkazu při najetí myši (II.)	13	Kolik znaků je možné vložit do <TITLE>?	47
Jak na nepodtržený odkaz?	13	Jak správně psát název třídy ve stylu?	47
Podtržení odkazu při najetí myši	14	Jak definovat klíčová slova pro vyhledávací servery?	48
Zrušení podtržení odkazu při najetí kurzorem	14	Jak definovat popisek stránky pro vyhledávací servery?	48
"Nadtržení" podtrhnutého odkazu	15	Snadnější formátování textu - <PRE>	49
Přeskrtnutí odkazu	15	Jak zjistit kompletní velikost stránky včetně obrázků?	49
Zvětšení textu odkazu pomocí CSS	15	Jaké rozlišení pro HTML stránky?	50
Tučný odkaz při najetí kurzorem myši	16	Jak tvořit stránky v XHTML?	51
Rychlejší načítání stránky s tabulkami	16	Jak na částečné orámování tabulky?	53
Pozor na prázdné buňky tabulek!	17	Jak vložit stejný text do všech stránek?	56
Orámování tabulky pomocí atributu Border	18	Jak vložit stejný text do všech stránek ? (II.)	57
Nastavení rozměru tabulek	19	Jak zobrazit první písmeno slova velkým písmenem?	57
Jak zarovnat tabulky pomocí CSS	20	Jak zobrazit vybraný text velkými písmeny?	58
Úsporné definování barvy pozadí buněk	21	Jak zobrazit vybraný text malými písmeny?	58
Umístění tabulky přesně na střed monitoru	22	Jak na odsazení prvního řádku textu?	59
Jak zkrátit výsledný kód tabulek?	23	Jak v roletovém menu zobrazit více položek současně?	60
Jak říci prohlížeči, že stránka obsahuje české znaky?	23	Jak dostat zvuk do www stránky? (I.)	61
Zakažte, aby prohlížel ukládal vaše stránky do cache	24	Jak dostat zvuk do www stránky? (II.)	62
Automatický reload stránky a přesměrování	24	Jak změnit mezery mezi písmeny slov?	63
Neviditelný okraj kolem tabulky v Netscape	25	Jak změnit mezery mezi jednotlivými slovy?	63
Jak zrušit mezeru za tagem <FORM>?	25	Jak změnit mezeru mezi jednotlivými řádky textu?	64
Jak na maximální počet znaků v textovém poli?	26	Jak na pohybující se text ve stránce?	64
Jak definovat velikost textového pole? (I.)	26	Jak na zobrazení textu v jiném rámci?	66
Jak definovat velikost textového pole? (II.)	27	Jak se zbavit 3D stínu ve formulářových prvcích?	67
Jak předdefinovat text v textovém poli?	27	Jak zrušit mezeru za tagem <FORM> podruhé...	67
Jak předdefinovat text v textareu?	28	Jak zrušit mezeru v seznamech a výčtech?	68
Velikost Textarea - pomocí HTML	29	Jak definovat barvu posuvníku prohlížeče?	69
Velikost Textarea s přesností na pixel – pomocí CSS	29	Obrázek na stránce, ale nikoli na tiskárně	70
Jak obarvit jednotlivé formulářové prvky	30	Obrázek na tiskárně, ale nikoli na stránce	70
Změna barvy buňky při najetí myši	32	Stránky bez cizích reklam na free-hostingu	71
Tenké orámování kolem tabulky	33	Všechny odkazy do nového okna	72
Tenké orámování tabulky včetně řádku a buněk	35	HTML stránka se nezobrazuje v Netscape?	72
Jak otevřít ze stránky e-mailovou zprávu?	36	Cachování stránek v Internet Exploreru	73
Jak pomocí odkazu doplnit do e-mailu předmět?	37	Obrázky na stránce bez zbytečných mezer	74
Jak na více příjemců e-mailu kliknutím na odkaz? (I.)	38	Přehled barev	76
Jak na více příjemců e-mailu kliknutím na odkaz? (II.)	38	Rejstřík	78
Neviditelný příjemce e-mailu kliknutím na odkaz?	39		

07 - PCWorld Edition – HTML – tipy a triky od profesionálů

Informace v této knize jsou zveřejněny bez ohledu na jejich případnou patentovou ochranu. Jména produktů byla použita bez záruky jejich volného použití. Vydavatel a autoři nepřebírají žádnou odpovědnost ani žádnou jinou záruku za použití údajů uvedených v této knize a z toho vyplývajících následků. Veškerá práva jsou vyhrazena na kopie celé, ale i části knihy pořízené jakýmkoliv způsobem pro účely obchodu. Žádná část této knihy nesmí být použita v žádném jiném informačním médiu a na žádném jiném nosiči dat za účelem obchodu bez předchozího písemného souhlasu vydavatele.

ISBN 80-86097-64-1

Úvod

Za ty tři roky, co tvořím HTML stránky a webovou grafiku a současně vedu internetový server Interval.cz (<http://www.interval.cz>), který se na tvorbu stránek a programování stránek přímo specializuje, jsem nasbíral velké množství zkušeností, spoustu zajímavých tipů a triků, které lze uplatnit při vytváření HTML stránek. V poslední době jsem si tak začal říkat, že těch tipů a triků je tolik, že by to možná vydalo i na samostatnou knížku, ale nijak dále jsem se této myšlence nevěnoval. Nakonec – nečekaně se stalo skutečností a díky vydavatelství UNIS Publishing se mi podařilo dosáhnout tohoto cíle a vydat knihu o tipech v HTML. Některé z tipů uvedené této knize naleznete také na serveru Interval.cz, ale převážná většina tipů je nových a nepublikovaných.

Při psaní této knihy jsem se snažil obsáhnout všechny oblasti tvorby HTML stránek – třeba, jak vyzrát na některé rozdíly mezi jednotlivými prohlížeči (zejména Internet Explorer x Netscape Navigátor), jak vylepšit pomocí kaskádových stylů formátování textu na stránkách, jak změnit vzhled standardně šedých formulářových prvků, naleznete zde i tipy pro vytváření tabulek, které jsou hlavním formátovacím prvkem HTML nebo konkrétněji - jak vytvořit odkaz, který po kliknutí zařadí stránku mezi oblíbené položky, jak vylepšit internetové odkazy na stránkách, jak pomocí odkazů otevřít e-mailovou zprávu s nadefinovanými vlastnostmi, jak vytvářet tabulky s tenkým orámováním, jak zrušit zobrazování cizích reklam na free-hostingových serverech, jak urychlit zobrazování stránek obsahujících tabulky, jak dosáhnout toho, aby se reklama obsažená na stránkách nezobrazila na tiskárně při tisku stránky, jak docílit pomocí kaskádových stylů větší mezery mezi písmeny, slovy či jednotlivými řádky, jak správně definovat písmo na stránce, aby text byl zobrazen hezky česky, jak definovat klávesové zkratky na stránkách, jak změnit barvu rolovací lišty v Internet Exploreru 5.5 a mnoho dalších užitečných věcí – doufám, že se vám budou líbit a že některé z nich použijete na vlastních webových stránkách.

V knížce prosím nehledejte žádné referenční informace o tvorbě stránek, kaskádových stylů atd. Naleznete zde pouze několik desítek tipů a triků pro HTML stránky, některé z nich jsou všeobecně známé, některé jsou zase prakticky neznámé, ale přesto užitečné, a já věřím, že tato kniha bude užitečná opravdu pro každého tvůrce HTML stránek – ať už je to ostřílený profesionál, nebo ten, kdo má s tvorbou stránek základní zkušenosti.

Tato kniha od vás předpokládá alespoň základní znalost problematiky tvorby webových stránek a kaskádových stylů.

Miroslav Kučera, autor

Svoje připomínky, náměty a poznámky zasílejte:

na adresu autora:

Miroslav Kučera, na Výsluní IV 709, 68762 Dolní Němčí,

e-mail: miroslav.kucera@interval.cz

nebo na adresu vydavatele:

UNIS Publishing, s.r.o.

Jundrovská 33, 624 00 Brno

tel.: 05 – 41 51 55 00 fax: 05 – 41 51 55 02

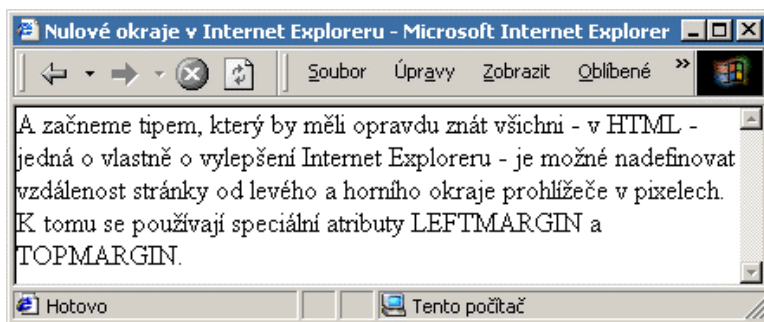
e-mail: knihy@win.cz

Nulové okraje stránky v Internet Exploreru

A začneme tipem, který by měli opravdu znát všichni – v HTML (jedná o vlastně o vylepšení Internet Exploreru) je možné nadefinovat vzdálenost stránky od levého a horního okraje prohlížeče v pixelech. K tomu se používají speciální atributy LEFTMARGIN a TOPMARGIN. Pokud je neuvědomíte, je obsah stránky od okraje prohlížeče automaticky odsazen o 10 pixelů zleva a shora, což může být u některých druhů stránek na škodu.

Atribut LEFTMARGIN nastavuje vzdálenost stránky od levého a současně od pravého okraje prohlížeče. Atribut TOPMARGIN nastavuje vzdálenost stránky od horního a současně spodního okraje prohlížeče. Pomocí těchto atributů není možné nastavit různé odsazení zleva a zprava či od horního a spodního okraje – je vždy stejné.

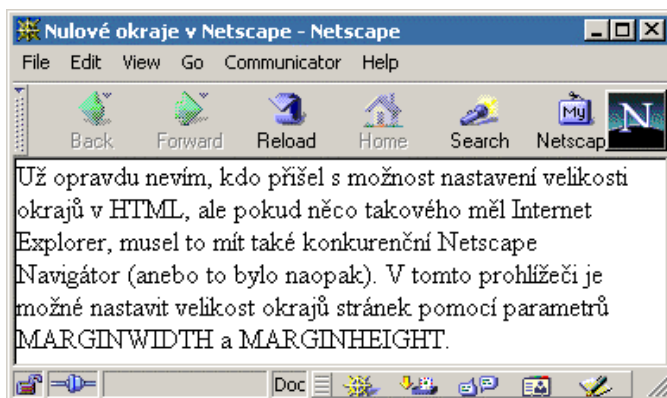
```
<body leftmargin="0" topmargin="0">
```



Tento zdrojový kód nastaví v Internet Exploreru nulové okraje ze všech stran.

Nulové okraje stránky v Netscape Navigátoru

Už opravdu nevím, kdo přišel s možností nastavení velikosti okrajů v HTML, ale pokud něco takového měl Internet Explorer, musel to mít také konkurenční Netscape Navigátor (anebo to bylo naopak). V prohlížeči Netscape Navigator je možné nastavit velikost okrajů stránek pomocí parametrů MARGINWIDTH a MARGINHEIGHT.



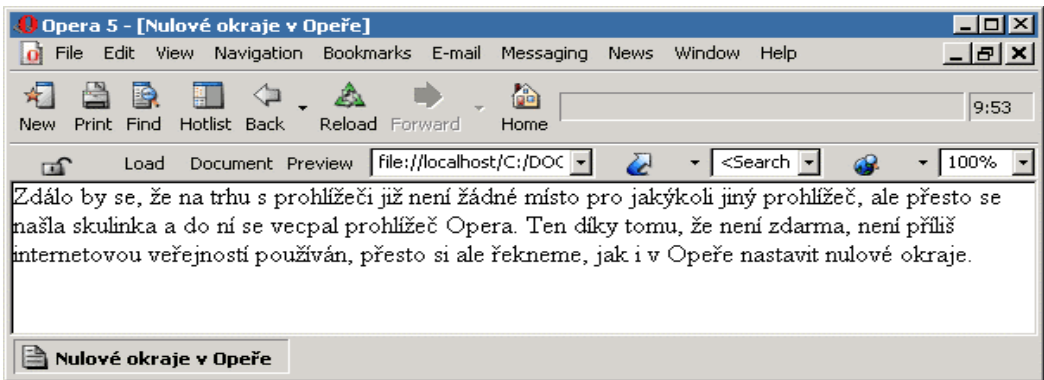
Atribut MARGINWIDTH nastavuje vzdálenost stránky od levého a pravého okraje prohlížeče. Atribut MARGINHEIGHT nastavuje vzdálenost stránky od horního a spodního okraje prohlížeče.

```
<body marginwidth="0" marginheight="0">
```

Tento zdrojový kód nastaví v Netscape Navigátoru nulové okraje ze všech stran.

Nulové okraje stránky v Opeře

Zdálo by se, že na trhu s prohlížeči již není žádné místo pro jakýkoli jiný prohlížeč, ale přesto se našla skulinka a do ní se vecpal prohlížeč Opera. Ten díky tomu, že není zdarma, není příliš internetovou veřejností používán, přesto si ale řekneme, jak i v Opeře nastavit nulové okraje.



U Opery nefunguje použití atributů LEFTMARGIN, TOPMARGIN, které jsou určeny pouze pro Internet Explorer, a stejně tak nefunguje použití atributů MARGINWIDTH a MARGINHEIGHT určených pro Netscape z jediného důvodu – tyto atributy totiž nejsou vůbec obsaženy ve specifikaci HTML 4.0. Pro Ope-ru musíte použít kaskádové styly CSS. V uvedeném příkladu je uvedena definice stylu pro sekci Body:

```
<style>
<!--
body
{
margin-left:0pt; margin-right:0pt; margin-top:0pt; margin-bottom:0pt;
} -->
</style>
```

Jak je z ukázky zdrojového kódu vidět, díky CSS je možné definovat odlišné vzdálenosti levého a pravého okraje, stejně jako vzdálenost od horního a spodního okraje. MARGIN-LEFT nastavuje vzdálenost od levého okraje, MARGIN-RIGHT vzdálenost od pravého okraje, MARGIN-TOP nastavuje vzdálenost od horního okraje a konečně, MARGIN-BOTTOM vzdálenost stránky od spodního okraje prohlížeče.

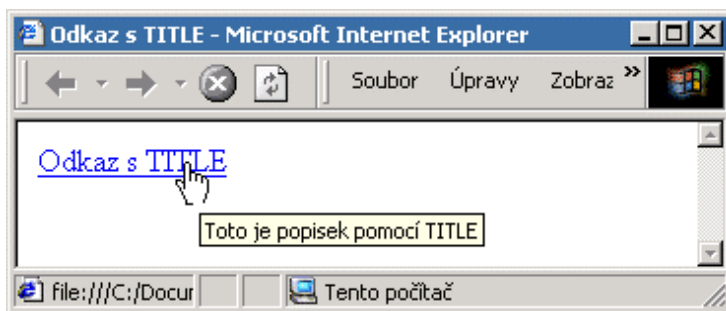
Nastavení okrajů pomocí CSS je plně funkční v Opeře, v Internet Exploreru od verze 4.0 a v Netscape Navigátoru 6. Výše uvedený zdrojový kód nastaví ve stránce nulové okraje. Pokud tedy nepotřebujete být kompatibilní ze starými verzemi prohlížečů, používejte pro nastavení velikosti okrajů kaskádové styly.

Využití atributu TITLE ve stránkách

Internet Explorer je známý svým nestandardním rozšířením syntaxe HTML. Typickým důkazem je i atribut TITLE, jehož použití může být ale v některých případech doslova k nezaplacení (např. nápověda nad odkazem, či formulářovým prvkem).

Pomocí atributu TITLE můžete prakticky k libovolnému HTML tagu definovat textový popis, který se objeví při najetí kurzorem myši. Lze jej použít u odkazu, u tabulky, nebo u formulářového prvku.

```
<a href="stranka.htm" title="Toto je popisek pomocí TITLE">Odkaz s TITLE</a>
```



Atribut TITLE je funkční pouze v Internet Exploreru pravděpodobně od verze 4.0.

Jak na klávesové zkratky v HTML stránce?

Klávesové zkratky dnes se běžně používají prakticky ve všech programech pro Windows. Vytvoření klávesových zkratk v HTML stránkách byl problém do té doby, dokud se neobjevilo HTML 4.0.

Klávesové zkratky se nejčastěji používají u odkazů nebo u prvků formuláře. Do odkazu stačí vložit atribut ACCESSKEY, ve kterém určíte požadovanou klávesu. Pokud daná klávesová zkratka bude v kolizi se zkratkou používanou v prohlížeči, bude v takém případě ignorována ve prospěch klávesové zkratky pro aplikaci.

```
<a href="stranka.htm" accesskey="t">Toto je odkaz - ALT+T</a>
```

V tomto případě se stiskem klávesy ALT+T definovaný odkaz označí a po následném stisku klávesy ENTER bude prohlížeč přeměřován na zadanou stránku v odkazu. Klávesové zkratky fungují v Internet Exploreru od verze 4.0.

Vaše stránka jako výchozí web v prohlížeči

Když jste si poprvé nainstalovali Internet Explorer nebo Netscape Navigator, po prvním spuštění jste zjistili, že v těchto prohlížečích je nastavena určitá domovská stránka – tedy stránka, která se vám začne načítat po spuštění prohlížeče. Většina uživatelů ji poté změnila na svoji vlastní nebo zruší vůbec. Ovšem, proč

byste nemohli nabídnout návštěvníkům možnost nastavit si stránku, na které se právě nacházejí, jako domovskou?

Jak na to? Otevřete si Poznámkový blok ve Windows či jakýkoli textový editor a zkopírujte do něj následující text. Do položky Start Page uveďte adresu vlastní www stránky:

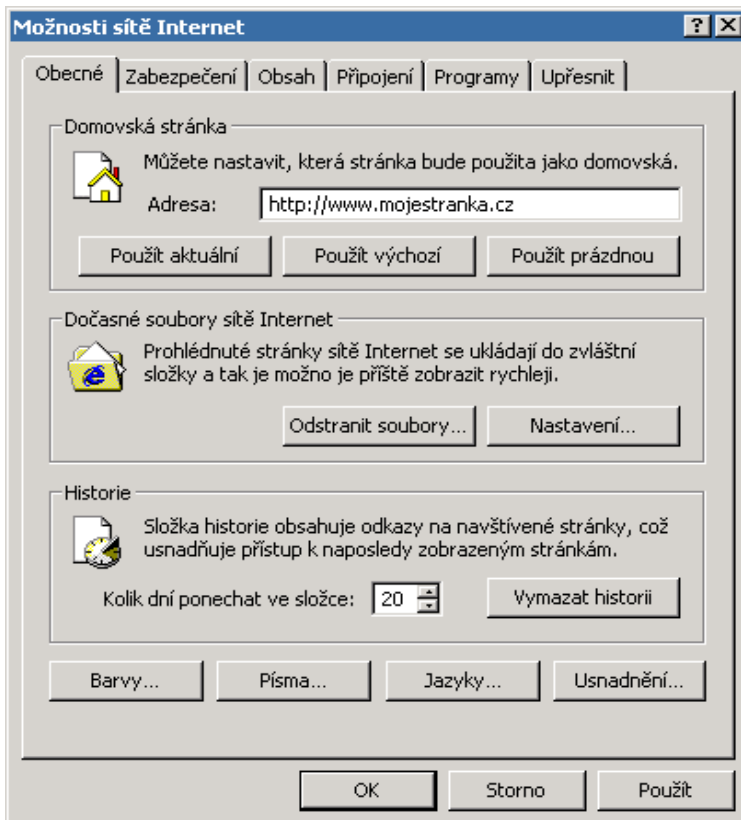
REGEDIT4

```
[HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\Main]
"Start Page" = http://www.mojestranka.cz
```

Tento soubor poté s příponou .reg uložte na disk a umístěte jej pomocí FTP klienta někam na váš server. Poté do stránky uveďte následující odkaz:

```
<a href="http://www.mojestranka.cz/soubor.reg">Web jako výchozí stránka</a>
```

Když návštěvník klikne na uvedený odkaz, obvykle se objeví dialogové okno, kde je nabídnuta volba stažení souboru nebo přímo jeho spuštění. Pokud návštěvník klikne přímo na stažení, soubor se uloží na lokální disk a teprve kliknutím na tento soubor na lokálním disku se nastaví uvedená domovská stránka. Nejlepší je, pokud návštěvník soubor ihned spustí – nastavená stránka se mu okamžitě určí jako domovská.



Tento tip je funkční pouze v Internet Exploreru. V případě Netscape jsem bohužel nepřišel na žádný zaručený postup, který by provedl totéž.

Přidání stránky do oblíbených položek

V každém prohlížeči je možné používat tzv. oblíbené položky. Jde vlastně o seznam stránek na Internetu, na které se dostanete pouhým výběrem z menu prohlížeče. V Internet Exploreru se pro vložení stránky do oblíbených položek používá klávesová zkratka CTRL+D, ale my můžeme našim čtenářům nabídnout přímo odkaz, který udělá totéž a navíc s předdefinovaným dialogem.

Jak na to? Do stránky si vložte tento odkaz:

```
<a href=" " onClick="if (window.confirm('Chcete opravdu přidat Moji stránku do oblíbeným položek?')) {external.AddFavorite('http://www.mojestranka.cz', 'Nazev stránky'); return false}>Vložit do oblíbených</a>
```

Po kliknutí na takový odkaz se otevře dialogové okno, které se návštěvníka zeptá, zdali chce uvedený server vložit do oblíbených položek. Pokud klikne na OK, otevře se dialogové okno prohlížeče "Přidat oblíbenou položku" s již nastaveným názvem serveru a jeho internetovou adresou.

Tento tip je funkční pouze v Internet Exploreru. V případě Netscape jsem bohužel nepřišel na žádný zaručený postup, který by provedl obdobnou věc.

Vložení dvou nezávislých tabulek vedle sebe

Použití tabulek v HTML stránkách se dnes stalo naprostou nezbytností. Tabulky ovšem mají tu nevýhodu, že se zobrazí až poté, co se načte jejich konec. To je problém především u dlouhých tabulek – čtenář tak dlouhou dobu nic nevidí. Ovšem, proč tabulky nerozdělit třeba do dvou nezávislých tabulek, které se budou také nezávisle zobrazovat? První se třeba zobrazí levá tabulka s hlavními odkazy a až poté tabulka pravá s hlavním textem serveru.

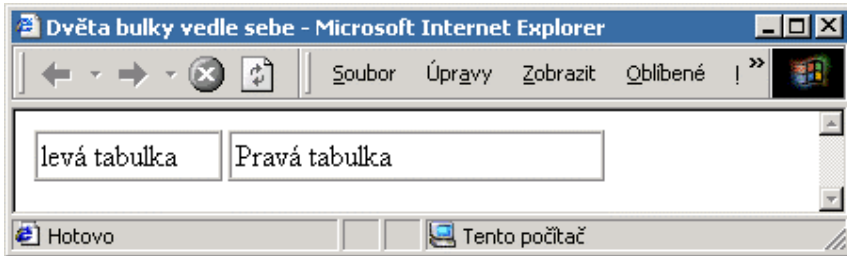
Řešení, které používám u všech svých webů, funguje zcela bez problému v Internet Exploreru, v Netscape Navigátoru a také v Opeře. Stejný systém rozvržení tabulek používají některé velké české servery, jako například Interval.cz (<http://www.interval.cz>) od Zoneru nebo Zive.cz (<http://www.zive.cz>) od Computer Pressu.

Vytvoření takových nezávislých tabulek je velmi jednoduché: do první tabulky prostě vložte atribut ALIGN="LEFT", jak to ukazuje následující kód:

```
<table cellpadding="2" cellspacing="0" border="1" align="left" width="100">
<tr>
<td>levá tabulka</td>
</tr>
</table>
```

```
<table cellpadding="2" cellspacing="0" border="1" width="200">
<tr>
```

```
<td>Pravá tabulka</td>
</tr>
</table>
```

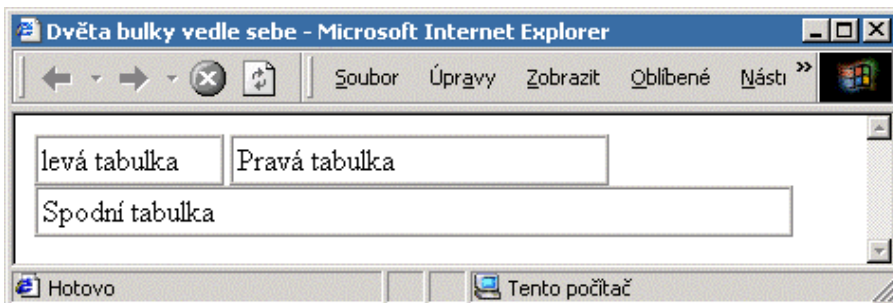


Mezi jednotlivými tabulkami je sice malá mezera, ale to by nemělo vadit. Existují i další způsoby, jak zarovnat tabulky vedle sebe, nicméně tento způsob je naprosto bezproblémový. Pokud potřebujete pod tyto dvě tabulky umístit tabulku další, je to zcela bez problémů, normálně ji vložte a ona se zarovná pod obě dvě tabulky předchozí.

```
<table cellspacing="0" cellpadding="2" border="1" align="left" width="100">
<tr>
<td>levá tabulka</td>
</tr>
</table>
```

```
<table cellspacing="0" cellpadding="2" border="1" width="200">
<tr>
<td>Pravá tabulka</td>
</tr>
</table>
```

```
<table cellspacing="0" cellpadding="2" border="1" width="400">
<tr>
<td>Spodní tabulka</td>
</tr>
</table>
```

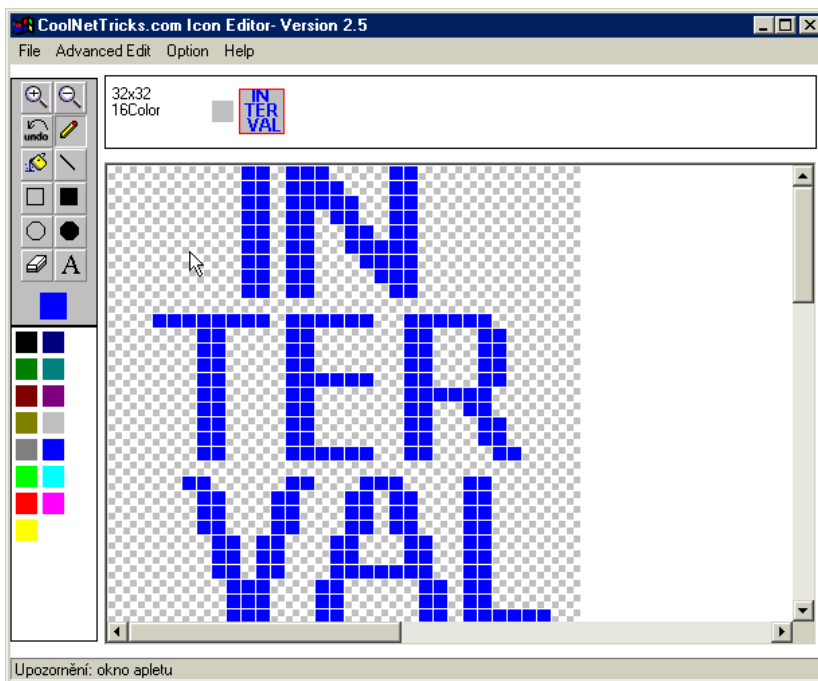


Druhá tabulka vpravo může mít libovolné rozměry, jenom jí nesmíte nastavit procentuální velikost, např. WIDTH="100%", pak by se objevilo podélné rolování, což je chyba, kterou nikdo netoleruje. Pokud chcete, aby tato pravá tabulka měla velikost podle aktuální velikosti okna, jednoduše její rozměr WIDTH nedefinujte – tabulka se natáhne podle délky textu, který je v ní. V případě, že vaše stránky příliš textu neobsahují a tabulka by tak nedosahovala až nakonec okna, vložte do ní jakýkoli dostatečně dlouhý text a nastavte mu barvu pozadí tabulky - text tak nebude vidět a tabulka tak bude mít vždy 100% rozměr velikosti okna prohlížeče.

Ikona vlastního webu do oblíbených položek

V Internet Exploreru 5.0 a vyšším je možné zobrazit ikonu serveru třeba v oblíbených položkách. Funguje to tak, že když si dáte nějaký web do oblíbených položek, tak se vyše požadavek na obrázek pojmenovaný jako favicon.ico a pokud je tento obrázek nalezen, tak se vloží vedle názvu serveru v nabídce oblíbených položek, nebo bude zobrazen u internetové adresy, či dokonce jako zástupce daného serveru na ploše Windows apod.

Ideální je použít nějaký externí program pro vytváření ikon. Dobrý je třeba program Icon Collector Graphics Editor (<http://www.greatowl.com/>) nebo můžete použít on-line generátor serveru Favicon.com (<http://www.favicon.com>). Je to JAVA applet (funguje v IE 4.0 či Netscape 4.5).



Poznámka vydavatele: existuje samozřejmě celá řada různých programů, vyzkoušejte například prastarý (1992!) program IMEDITOR.exe, který najdete na CD v souboru ICONEDIT.zip nebo ho stáhnete z webu www.win.cz společně s touto publikací v elektronické podobě.

Ikona musí mít rozměr 16x16 nebo 32x32 pixelů (doporučuji 16x16) a snažte se, aby výsledný soubor měl co nejmenší velikost – nezapomínejte totiž, že Internet Explorer si musí tuto ikonu z vašeho serveru natáhnout a pokud bude mít příslušný soubor třeba 20 KB, čtenáři vás nebudou mít rádi. V jednom favicon.ico může být více obrázků, třeba jedna ikona v provedení 16x16 a 32x32 s tím, že prohlížeč použije ten obrázek, který bude zrovna potřebovat.

Počet barev je také na vás, doporučuji tvořit ikonu v základních 16 barvách, je to nejspolehlivější. Třeba takové Yahoo (<http://www.yahoo.com/favicon.ico>) používá ikonu 16x16 v 16 barvách. To, jak má ikona vypadat, již nechám zcela na vás.

Bude fungovat bitmapový obrázek uložený jako .ico?

Mnoho lidí si myslí, že formát ICO je stejný jako formát BMP. Bohužel není, zkusel jsem ve Photo-shopu vytvořit obrázek 16x16 pixelů, uložil jsem jej jako .bmp, poslal na web, ale žádná ikonka se neobjevila. Formát BMP má prostě jinou strukturu než formát pro ikony. Pro tvorbu ikonek použijte tedy speciální program, nebo již nahoře zmiňovaný on-line generátor favicon.ico – nemusíte stahovat žádný program a vytvořený obrázek vám přijde ihned e-mailem.

Poznámka vydavatele:

Struktura ICO souboru, alespoň schematicky, je následující:

Část struktury souboru	Velikost	Popis	Označení
ICONHEADER (hlavička souboru)	2 byty	rezervováno, musí být 0	ihReserved
	2 byty	nastaveno na 1 (první čtyři byty platného souboru ICO, který obsahuje ikony, musí být 00 00 01 00)	ihType
	2 byty	počet obrázků ikon v ICO souboru	ihCount
	proměnná	datová struktura s údaji pro každý obrázek ikony (rozměr, počet barev, umístění v ICO souboru...)	ihEntries
ICONIMAGE 1	proměnná	pro každou ikonu obsahuje 40bytovou hlavičku, informace o barvách a vlastní obrázek jako DIB ve dvojím provedení: vlastní barevný obrázek ikony tak, jak se bude zobrazovat (icXOR) a monochromatická maska (icAND), která definuje plochu – obrazové body, které ikona na pozadí překryje (obrázek ikony nemusí být vždy jenom obdélníkový).	
ICONIMAGE 2	proměnná		
...	proměnná		

A co dělat dále, když mám ikonku vytvořenou?

Jakmile máte soubor s obrázkem vytvořený a pojmenovaný jako favicon.ico, stačí ho poslat pomocí FTP klienta na vaši doménu, do adresáře, kde se nachází stránka, pro kterou chcete, aby se zobrazovala tato ikona. Můžete tedy mít v různých adresářích různé ikony.

Pokud se tedy nacházím třeba na úvodní stránce vašeho webu, stačí favicon.ico poslat do rootu, pokud mám další stránky, třeba v adresáři /clanky, a chci zde mít pro Oblíbené položky jinou ikonu, pošlu ji do adresáře /clanky. Soubor favicon.ico musí být prostě ve stejném adresáři, kde se nacházejí vaše HTML stránky.

Dále je možné nastavit obrázek pouze pro konkrétní HTML stránku. To se udělá velmi jednoduše, do dané stránky, přesněji do sekce <HEAD> vložíte tento HTML kód:

```
<head>
<link rel="SHORTCUT ICON" href="http://www.mojestranka.cz/mojeikona.ico">
<title>Titulek stránky</title>
</head>
```

V tagu LINK uveďte cestu k vaší ikoně. Ta může být pojmenována pochopitelně zcela libovolně. Způsob popsany v tomto tipu funguje, sám jsem to vyzkoušel, bohužel, podle informací čtenářů nemusí fungovat na všech serverech. Na to, proč tomu tak je, jsem nepřišel.

Nahrazení standardního kurzoru Windows

Pokud najedete na libovolné HTML stránce na odkaz, změní se kurzor myši z klasické šipky na malou ručičku. Vy můžete pomocí kaskádových stylů tento kurzor snadno změnit. Syntaxe je jednoduchá, v uvedeném zdrojovém kódu je nadefinován pro všechny odkazy na stránce kurzor přesýpacích hodin:

```
<style>
<!--
a {cursor: wait;}
-->
</style>

<a href="http://www.mojestranka.cz">Kurzor přesýpacích hodin</a>
```

Seznam kurzorů, které lze použít:

- auto – automatická volba podle nastavení OS
- default – standardní šipka
- hand – ruka
- ne-resize – menší šipka, zrcadlově převrácená
- n-resize – kolmá šipka, směr nahoru
- se-resize – šikmá šipka vpravo dole
- w-resize – kolmá šipka vlevo
- nw-resize – menší šipka
- e-resize – kolmá šipka vpravo
- help – šipka s otazníkem
- move – kurzor pro přesun
- sw-resize – šikmá šipka vlevo dole
- s-resize – kolmá šipka, směr dolů
- text – textový kurzor
- crosshair – zaměřovač
- wait – přesýpací hodiny

Nahrazení kurzoru za jiný funguje pouze v Internet Exploreru. Netscape a ani Opera kurzory myši nemění a ponechávají standardní nastavení operačního systému.

Nastavení barvy odkazů na HTML stránce

V HTML stránce rozlišujeme tři druhy odkazů – nenavštívené (link), navštívené (vlink) a aktivní (alink). V sekci BODY nebo pomocí kaskádových stylů můžete snadno určit jejich barvu. Následuje příklad zdrojového kódu pro BODY:

```
<body link="red" vlink="blue" alink="green">
```

Vedle uvedených tří základních barev (red – červená, blue – modrá a green – zelená) můžete použít samozřejmě i další barvy. Hodnoty RGB a názvy barev základní 16barevné palety nebo rozšířené 64barevné palety najdete v příloze této publikace. Hezkou ukázkou palety „bezpečných webových“ barev najdete na webové adrese <http://www.interval.cz/r-article.asp?id=373>.

Nastavení barvy odkazů pomocí CSS

Barvu odkazů je samozřejmě možné nastavit i pomocí kaskádových stylů CSS:

```
<style>
<!--
a:link {color:red}
a:visited {color:blue}
a:active {color:green}
-->
</style>
```

Nastavení vlastní barvy odkazů je velmi často používaná věc, je to uvedeno i ve specifikaci HTML 3.2 a možná právě proto funguje ve všech používaných prohlížečích včetně Opery. V dnešní době doporučuji definovat barvu odkazů pomocí kaskádových stylů, neboť použití značek LINK, VLINK a ALINK v sekci BODY je HTML v 4.0 označeno jako Deprecated (zavrženo).

Změna barvy odkazu při najetí myši (I.)

Pomocí kaskádových stylů je možné změnit barvu odkazu, když na něj najedete kurzorem myši. Pomocí stylu si nadefinujete barvu standardního odkazu a pomocí pseudotřídy a:hover si nadefinujete barvu odkazu jinou, kterou bude odkaz přebarven při najetí kurzorem myši, třeba takto:

```
<style>
<!--
a {color:red}
```

```
a:hover {color:blue}
-->
</style>

<a href="stranka.htm">Měníci se odkaz</a>
```

Změna barvy odkazů pomocí pseudotřídy a:hover funguje v Internet Exploreru od verze 4.0, v Opeře od verze 4.0 a také v Netscape od verze 6.0.

Změna barvy odkazu při najetí myši (II.)

V minulém tipu jsme si ukázali změnu barvy odkazu pomocí pseudotřídy a:hover. Jde to udělat i jiným způsobem a to s využitím události v JavaScriptu. Tyto události mohou být různého druhu, třeba jedno nebo dvě klepnutí tlačítka myši, v našem případě si vystačíme s dvojicí událostí "ONMOUSEOVER" a "ONMOUSEOUT".

Událost ONMOUSEOVER určuje stav, kdy uživatel najede kurzorem nad odkaz. Událost ONMOUSEOUT zase naopak určuje stav, kdy uživatel opustí oblast odkazu. Samotné barvy jsou definovány pomocí řádků this.style.color='barva':

```
<a href="stranka.htm" onmouseover="this.style.color='Red'" onmouse-
out="this.style.color='blue'">Změna barvy odkazu</a>
```

Protože pomocí těchto dvou událostí nemůžeme definovat barvu "startovacího" odkazu, na který jste ještě neukázali kurzorem, viz. ukázka – barva základního odkazu modrá, jakmile najedete myší, je odkaz červený, a jakmile se vzdálíte s kurzorem, je konečně pořádkem zelený, doporučuji v sekci BODY uvést parametry VLINK, ALINK a LINK, jejichž barva bude shodná s barvou definovanou v události ONMOUSEOUT.

Změna barvy odkazů pomocí událostí je funkční pouze v Internet Exploreru, proto doporučuji používat pro změnu barvy odkazů způsob pomocí CSS, který byl popsán v minulém tipu a který funguje všude mimo Netscape verze 4.xx.

Jak na nepodtržený odkaz?

Standardní odkazy na HTML stránkách jsou vůči normálnímu textu obvykle zvýrazněny podtržením. Někomu se to ovšem nemusí líbit – pomocí kaskádových stylů můžeme podtržení odkazu snadno zrušit.

V následujícím ukázce stylu je nadefinován odkaz jako nepodtržený (text-decoration:none):

```
<style>
<!--
a {text-decoration:none}
-->
</style>
```

```
<a href="stranka.htm">Nepodtržený odkaz</a>
```

Odkaz nebude podtržen v žádném prohlížeči, podporujícím alespoň částečně kaskádové styly první generace (CSS1) – Internet Explorer, Netscape Navigátor od verze 4.0 a Opera.

Podtržení odkazu při najetí myši

V minulém tipu jsem vám ukázal, jak u všech odkazů na stránce zrušit podtržení. V tomto tipu podtržení odkazů opět zrušíme, ale obnovíme je při najetí kurzorem myši. Jak? Jak jinak, než pomocí kaskádových stylů CSS a pseudotřídy a:hover:

V následující ukázce stylu jsou odkazy standardně definovány jako nepodtržené (text-decoration:none), pomocí pseudotřídy a:hover je při najetí kurzoru myši podtrhneme (text-decoration:underline):

```
<style>
<!--
a {text-decoration:none}
a:hover {text-decoration:underline}
-->
</style>

<a href="stranka.htm">Odkaz, který se podtrhne</a>
```

Podtržení odkazu funguje v Internet Exploreru od verze 4.0, v Opeře od verze 4.0 a v Netscape Navigátoru od verze 6.0.

Zrušení podtržení odkazu při najetí kurzorem

Na většině HTML stránek jsou odkazy standardně podtrženy, nebo podtrženy nejsou a podtrhnou se až při najetí kurzorem myši. Tento tip ovšem učiní přesný opak – podtržení odkazu při najetí kurzorem myši odstraníme.

Ve stylu jsou všechny odkazy definovány jako podtržené (text-decoration:underline), pomocí pseudotřídy a:hover podtržení odkazu jednoduše zrušíme (text-decoration:none):

```
<style>
<!--
a {text-decoration:underline}
a:hover {text-decoration:none}
-->
</style>

<a href="stranka.htm">Odkaz, kde se zruší podtržení</a>
```

Zrušení podtržení odkazu funguje v Internet Exploreru od verze 4.0, v Opeře od verze 4.0 a v Netscape Navigátoru od verze 6.0.

"Nadtržení" podtrženého odkazu

Pomocí kaskádových stylů je možné snadno udělat takové vychytávky, kdy se podtržený odkaz dole při najetí kurzorem myši "nadtrhne" nahore.

Ve stylu jsou všechny odkazy definovány jako podtržené (`text-decoration:underline`), pomocí pseudotřídy `a:hover` snadno vytvoříme nadtržení odkazu (`text-decoration:overline`):

```
<style>
<!--
a {text-decoration:underline}
a:hover {text-decoration:overline}
-->
</style>

<a href="stranka.htm">Odkaz s nadtržením</a>
```

"Nadtržení" odkazu vede k nepřehlednosti odkazu, zvláště, je-li odkaz dlouhý přes několik řádků. Zrušení podržení odkazu funguje v Internet Exploreru od verze 4.0, v Opeře od verze 4.0 a v Netscape Navigátoru od verze 6.0.

Přeškrtnutí odkazu

Odkazy je možné pomocí kaskádových stylů podtrhávat, "nadtrhávat", ale také přeškrtnout.

Ve stylu jsou všechny odkazy definovány jako nepodtržené (`text-decoration:none`), pomocí pseudotřídy `a:hover` snadno vytvoříme přeškrtnutí odkazu (`text-decoration:line-through`):

```
<style>
<!--
a {text-decoration:none}
a:hover {text-decoration:line-through}
-->
</style>

<a href="stranka.htm">Přeškrtnutý odkaz</a>
```

Pozor ale na to, že přeškrtnutí odkazu má za následek horší čitelnost textu odkazu. Přeškrtnutí odkazu funguje v Internet Exploreru od verze 4.0, v Opeře od verze 4.0 a v Netscape Navigátoru od verze 6.0.

Zvětšení textu odkazu pomocí CSS

Pokud umíte dobře kaskádové styly, neměl by vám tento tip dělat sebemenší problémy ohledně toho, jak funguje. Pomocí CSS si totiž při najetí kurzorem myši odkaz zvětšíme.

V následujícím stylu mají všechny odkazy standardně definovanou velikost 11pt; (font-size:11pt;), pomocí pseudotřídy a:hover snadno odkazy zvětšíme (font-size:16pt;):

```
<style>
<!--
a {font-size:11pt;}
a:hover {font-size:16pt;}
-->
</style>

<a href="stranka.htm">Zvětšující se odkaz</a>
```

Zvětšování odkazu při najetí kurzorem myši zhoršuje čitelnost dokumentu, neboť text skáče nahoru a dolů. Zvětšení odkazu funguje v Internet Exploreru od verze 4.0, v Opeře od verze 4.0 a v Netscape Navigátoru od verze 6.0.

Tučný odkaz při najetí kurzorem myši

V minulém tipu jsme odkaz při najetí kurzorem myši zvětšili o několik bodů. Nyní si ukážeme něco podobného, ale odkaz nebudeme zvětšovat, ale uděláme jej tučně.

V následujícím stylu mají všechny odkazy standardně definovaný normální řez (font-weight: normal;), pomocí pseudotřídy a:hover řez změníme na tučný (font-weight: bold;):

```
<style>
<!--
a {font-weight: normal;}
a:hover {font-weight: bold;}
-->
</style>

<a href="stranka.htm">Zvětšující se odkaz</a>
```

Změna netučného odkazu na odkaz tučný při najetí kurzorem myši zhoršuje čitelnost dokumentu, neboť text může skákat nahoru a dolů. "Ztučnění" odkazu funguje v Internet Exploreru od verze 4.0, v Opeře od verze 4.0 a v Netscape Navigátoru od verze 6.0.

Rychlejší načítání stránky s tabulkami

O tabulkách je známo, že se uživateli zobrazí jejich obsah, až se tabulka celá načte. V případě dlouhé tabulky tak návštěvník vaší stránky dlouho nic nevidí a pak náhled se mu zobrazí celý obsah tabulky. Ovšem proč jednu tabulku nerozdělit do několika kratších, které se budou postupně zobrazovat?

Špatný postup:

```
<table cellspacing="0" cellpadding="2" border="0">
<tr>
<td>První řádek</td>
</tr>
<tr>
<td>Druhý řádek</td>
</tr>
<tr>
<td>Třetí řádek</td>
</tr>
</table>
```

Doporučený správný postup:

```
<table cellspacing="0" cellpadding="2" border="0">
<tr>
<td>První tabulka</td>
</tr>
</table>
```

```
<table cellspacing="0" cellpadding="2" border="0">
<tr>
<td>Druhá tabulka</td>
</tr>
</table>
```

```
<table cellspacing="0" cellpadding="2" border="0">
<tr>
<td>Třetí tabulka</td>
</tr>
</table>
```

Jak je z druhé ukázky zdrojového kódu vidět, zdrojový kód je sice o něco delší, ale v případě delšího obsahu v tabulkách se tento způsob rozhodně vyplatí. Tabulky se načítají a také zobrazují postupně, což vytváří dojem, že se stránka rychleji načítá.

Pozor na prázdné buňky tabulek!

Občas se stane, že při tvorbě stránek obsahující komplikované tabulky se objeví nějaká buňka, která nemá žádný v době obsah – je prázdná. V Internet Exploreru se taková buňka zobrazí docela bez problémů, v Netscape Navigátoru vzniknou velké problémy: Taková buňka není nijak naformátována, nemá žádnou barvu apod.

V takových případech vložte do prázdné buňky tvrdou mezeru ` `; nebo neviditelný (průhledný) obrázek `1x1.gif`:

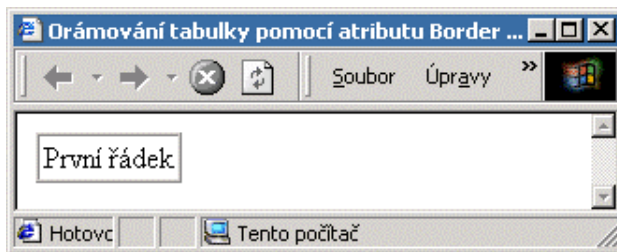
```
<table cellspacing="0" cellpadding="2" border="0">
<tr>
<td>První řádek</td>
</tr>
<tr>
<td>&nbsp;</td>
</tr>
</table>
```

Použití neviditelného obrázku o velikost `1x1` má navíc tu výhodu, že pomocí něj můžete danou buňku tabulky natáhnout na požadovaný rozměr třeba `50` pixelů (``) či jednoduše odsadit část textu od okraje třeba o `2` pixely.

Orámování tabulky pomocí atributu Border

Občas potřebujete tabulky na HTML stránce nějak zvýraznit – nejlepší způsob, jak toho docílit, je dát jim orámování pomocí standardního atributu `BORDER`, jehož hodnotou je číslo, určující tloušťku orámování, v našem případě je to `1`:

```
<table cellspacing="0" cellpadding="2" border="1">
<tr>
<td>První řádek</td>
</tr>
</table>
```



Pomocí prostředků HTML můžete ovšem řídit i barvu orámování tabulky, použijte atribut `BORDERCOLOR`, jehož hodnotou je barva, v našem případě barva černá:

```
<table cellspacing="0" cellpadding="2" border="1" bordercolor="black">
<tr>
<td>První řádek</td>
</tr>
</table>
```

V Netscape verze 4.xx bohužel pomocí těchto dvou atributů nedosáhnete absolutně černého orámování, jako v Internet Exploreru. Netscape totiž orámování automaticky stínuje a toto stínování nelze jinak vypnout.

V minulém odstavci jsem se zmínil o stínování, či spíše vytvoření jednoduchého 3D efektu tabulky. Ano, i toto je možné dosáhnout pomocí atributů BORDERCOLORLIGHT pro světlejší část orámování a BORDERCOLORDARK pro tmavší část orámování tabulky. Následující příklad vytvoří tabulku s jednoduchým 3D efektem pomocí červené a černé barvy:

```
<table cellpadding="0" cellspacing="2" border="10 bordercolorlight="red" bordercolordark="black">
<tr>
<td>První řádek</td>
</tr>
</table>
```

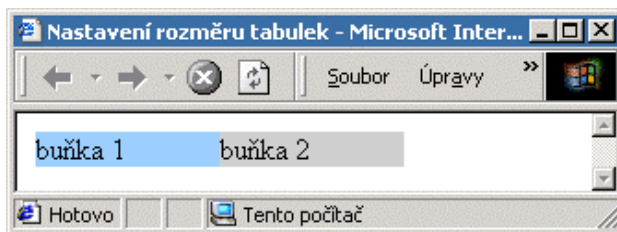
Atribut BORDER je v předchozím případě nastaven na hodnotu 10, aby výsledný 3D efekt příkladu lépe vynikl. V Netscape Navigátoru verze 4.xx vznikne také 3D efekt, ale nikoli červeno-černé barvy jako v Internet Exploreru, ale světle-tmavošedý 3D efekt.

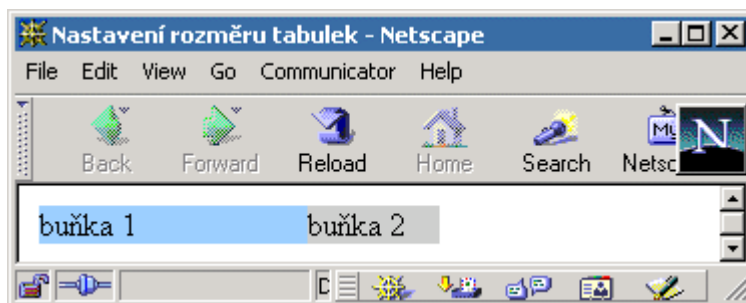
Nastavení rozměru tabulek

Zkušenější tvůrci stránek jistě vědí, že velikost tabulek a buněk se nastavuje pomocí atributu WIDTH="rozměr", kde je možné uvést přímo absolutní velikost v pixelech anebo použít procentuální velikost aktuálního okna prohlížeče.

Někdy se ovšem přihodí velmi nepříjemná věc, že v Netscape Navigátoru, ale i v Internet Exploreru nejsou dodržovány zadané rozměry tabulek. Je těžké vytvořit pro demonstrační účely tabulku, kde nebudou rozměry sedět, ale podařilo se mi to v případě této tabulky:

```
<table cellpadding="0" cellspacing="0" border="0" width="200">
<tr>
<td width="100" bgcolor="#99ccff">buňka 1</td>
<td bgcolor="#cccccc">buňka 2</td>
</tr>
</table>
```





Obrázek tabulky na předchozí straně ukazuje, jak jsou rozměry tabulek interpretovány v Internet Exploreru 5.0, obrázek tabulky na této straně ukazuje rozměry buněk v Netscape Navigátoru 4.5.

Na tomto příkladu je jasně vidět, že občas dochází k odlišné interpretaci definovaných rozměrů buněk v jednotlivých prohlížečích. Pomiňme nyní fakt, že pro určení správných rozměrů v Netscape stačí, abychom do druhé buňky vložili `WIDTH="100"`, řekněme, že to nebude v Netscape fungovat. Jaké jsou nyní možnosti?

Možností je spousta, ale jediný způsob, který zabere ve 100 % případů, je použití neviditelného obrázku o velikosti 1x1 pixelu, který do buňky vložíme pomocí tagu `IMG` a atributem `WIDTH` natáhneme podle potřeby, v tomto případě velikost obrázku nastavíme na 100 pixelů:

```
<table cellspacing="0" cellpadding="0" border="0" width="200">
<tr>
<td bgcolor="#99ccff"><br>buňka
1</td>
<td bgcolor="#cccccc"><br>buňka
2</td>
</tr>
</table>
```

Z předchozího obrázku dvou tabulek, první je tabulka v Internet Exploreru, druhá v Netscape Navigátoru je zřejmé, že díky použití neviditelného obrázku 1x1.gif, který byl vložen do obou buněk tabulky a pomocí parametru `WIDTH` natažen na 100 pixelů se vše konečně zobrazuje tak jak má.

Pokud tedy máte někdy problémy se zobrazením definovaných rozměrů tabulek, bez váhání použijte neviditelný obrázek 1x1.gif, který si natáhněte podle svých potřeb. Funguje to v každém případě a některých případech je použití tohoto neviditelného obrázku dokonce nezbytné!

Jak zarovnat tabulky pomocí CSS

Kaskádové styly se v dnešní době používají stále více a více. I přes fakt, že specifikace k CSS1 je již několik let stará, mnozí tvůrci www stránek stále nevědí, jak pomocí kaskádových stylů třeba zarovnávat tabulky. Jakékoli objekty ve stránce se tedy nezarovnávají pomocí `ALIGN` nebo dokonce `TEXT-ALIGN`, ale pomocí `FLOAT`. `ALIGN` v CSS neexistuje a `TEXT-ALIGN` se používá pro zarovnání textu ve stránce.

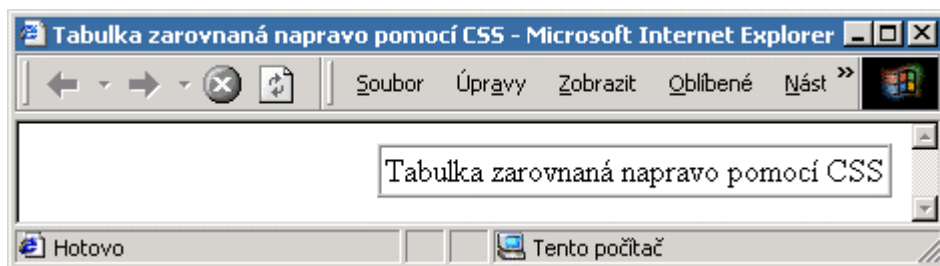
V následujícím stylu je definováno, že zarovnání se vztahuje na všechny tabulky a že jsou zarovnány na pravou stranu (`float:right`):

```

<style>
<!--
table {float:right}
-->
</style>

<table cellspacing="0" cellpadding="2" border="1">
<tr>
<td>Tabulka zarovnaná napravo pomocí CSS</td>
</tr>
</table>

```



Zarovnání tabulky pomocí CSS mi bohužel nefungovalo v Netscape Navigátoru 4.5, v Opeře 4.0 a Internet Exploreru od verze 5.0 je vše v pořádku. Netscape 6 je také v pořádku.

Úsporné definování barvy pozadí buněk

Občas potřebujete použít tabulky na stránce nějak obarvit. To se provádí pomocí atributu BGCOLOR="barva", který se obvykle vkládá do buňky TD, kterou chceme obarvit. Ale víte, že tento atribut můžete použít i v tagu TR, kde tak snadno dosáhnete obarvení celého řádku tabulky místo toho, abyste vkládali BGCOLOR do každé buňky tabulky? Ušetříte si tak práci s vkládáním tagu do všech buněk a navíc, výsledný HTML kód bude o něco kratší:

Neúsporný způsob:

```

<table cellspacing="0" cellpadding="2" border="0">
<tr>
<td bgcolor="#cccccc">buňka 1</td>
<td bgcolor="#cccccc">buňka 2</td>
<td bgcolor="#cccccc">buňka 3</td>
<td bgcolor="#cccccc">buňka 4</td>
</tr>
</table>

```

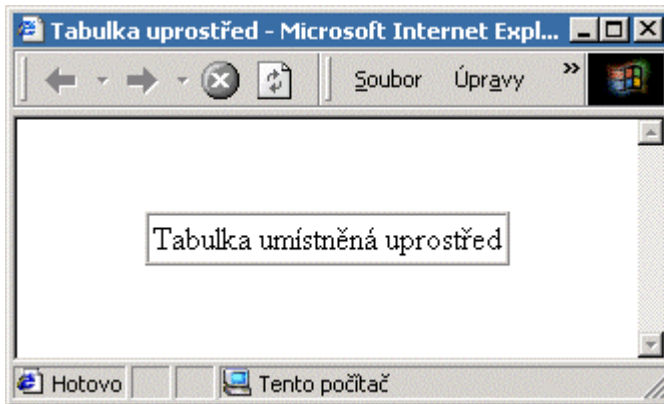
Úsporný a doporučený způsob:

```
<table cellspacing="0" cellpadding="2" border="0">
<tr bgcolor="#cccccc">
<td>buňka 1</td>
<td>buňka 2</td>
<td>buňka 3</td>
<td>buňka 4</td>
</tr>
</table>
```

Použití atributu BGCOLOR v řádce tabulky TR není sice v souladu se specifikací HTML 4.01, nicméně ale funguje ve všech dnes nejčastěji používaných prohlížečích, včetně Netscape Navigátoru verze 4.xx.

Umístění tabulky přesně na střed monitoru

Zejména u osobních HTML stránek je potřeba umístit prezentaci přesně na střed monitoru – nejenom zleva a zprava, ale i shora a zezdola a navíc nezávisle na velikosti okna. Není potřeba používat JavaScript, HTML to zvládne samo.



Vytvořte si první tabulku, která bude mít nadefinovány rozměry tabulky WIDTH="100%" a HEIGHT="100%". Tím se tato tabulka roztáhne na 100% aktuální velikosti okna (do šířky a do výšky). Pak do buňky této tabulky vložte atributy ALIGN="center" a VALIGN="middle", které zarovnájí obsah této buňky – v našem případě je to další tabulka – přesně na střed monitoru:

```
<table border="0" cellspacing="0" cellpadding="0" width="100%" height="100%">
<tr><td align="CENTER" valign="MIDDLE">

<table cellspacing="0" cellpadding="2" border="1">
<tr>
<td>Tabulka umístěná uprostřed</td>
```

```
</tr>
</table>

</td></tr></table>
```

Zdrojový kód si zkopírujte a použijte na prázdné HTML stránce, uvidíte, že to opravdu funguje a že vnořená tabulka je opravdu přesně ve středu monitoru a to nezávisle na velikosti okna prohlížeče. Uvedený příklad funguje bez problémů i v Netscape Navigátoru, takže jej můžete bez obav používat.

Jak zkrátit výsledný kód tabulek?

Mnoho tvůrců HTML stránek občas, třeba i nevědomky, porušuje syntaktická pravidla tvorby www stránek, občas se neuzavírají vybrané tagy, či některé tagy se mezi sebou kříží apod. Tento tip spadá přesně do kategorie neuzavírání párových tagů – v tomto případě se jedná o tagy, které se týkají tvorby tabulek.

Dnešní moderní prohlížeče jsou totiž tak inteligentní, že dokáží sami pochopit, kdy začíná nová buňka tabulky a kdy začíná nový řádek. Díky této inteligenci tak prakticky nemusíte uzavírat buňku TD a dokonce ani definici řádku TR. Prohlížeče totiž vědí, že jakmile se v kódu objeví nový TD nebo TR, bez vložení nového TABLE, daná buňka nebo řádek musela zákonitě skončit a začíná tak nová buňka nebo řádek. A zde je ukázka takového kódu:

```
<table cellspacing="0" cellpadding="2" border="1">
<tr>
<td>Buňka 1
<td>Buňka 2
<tr>
<td>Buňka 3
<td>Buňka 4
</table>
```

Tento tip je zde pouze pro úplnost, aby demonstroval, co je možné si při tvorbě HTML stránek dovolit díky liberálním prohlížečům. Samozřejmě nepředpokládám, že byste tvořili tímto způsobem tabulky ve svých stránkách. Rozhodně vám to nedoporučuji, klidně mohou nastat nečekané problémy u složitějších tabulek či u starších prohlížečů a také se vám zhorší přehlednost výsledného HTML kódu. Takže – neukončování tagů pro tabulky rozhodně nepoužívejte!

Jak říci prohlížeči, že stránka obsahuje české znaky?

V každé HTML stránce, v sekci HEAD musí být uveden tag META (viz. níže), pomocí kterého prohlížeči sdělíte, v jakém kódování je stránka vytvořena. Díky tomu se vám bez problémů mohou zobrazit české znaky. Pokud bude tento tag META ve stránce chybět, prohlížeč – zejména Netscape – může kódování stránky špatně určit a vám se tak nemusí korektně zobrazit některé české znaky. Pokud stránky tvoříte na platformě Windows, použijte kódování Windows-1250:

```
<html>
<head>
<title>Titulek stránky</title>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1250">
</head>
```

Kódování Windows-1250 je v českých podmínkách nejpoužívanější, nicméně není oficiálním standardem pro zobrazování českých znaků. Pokud tedy tvoříte www stránky na platformě Windows, a nechcete si komplikovat práci s převodem do ISO 8859-2, které je standardem, nezapomeňte výše uvedený META tag vložit do každé své stránky – zajistíte tak bezproblémové zobrazování českých znaků. Kódování ISO 8859-2 použijete takto:

```
<html>
<head>
<title>Titulek stránky</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-2">
</head>
```

Více o problematice zobrazování českých znaků naleznete na serveru www.cestina.cz.

Zakažte, aby prohlížeč ukládal vaše stránky do cache

Ukládání stránek do cache má své výhody i nevýhody. Stránka, která je uložena v cache, se při příští návštěvě stránky nenačte ze serveru, ale z vašeho lokálního disku, a to řádově rychleji a navíc se šetří kapacita přenosové linky. Nic ovšem není dokonalé a občas se může stát velmi nepříjemná věc, že se vám z cache načte stránka neaktuální, tzn. na webu je již nová verze stránky, s novým obsahem.

Pomocí jednoho META-tagu snadno zajistíte, aby prohlížeče vaši stránku nenačítaly do cache a vy si tak v budoucnu ušetříte spoustu možných problémů s neaktuálními verzemi:

```
<head>
<title>Titulek stránky</title>
<meta http-equiv="Pragma" content="no-cache">
</head>
```

Automatický reload stránky a přesměrování

Co budete dělat v případě, že se vaše stránky přesunou na jiné místo, třeba z adresy <http://www.domena.cz/mojestranka> na adresu <http://www.mojestranka.cz>? Přesunem internetové adresy bez přesměrování byste snadno mohli přijít o čtenáře. Anebo, někdy můžete potřebovat v nastaveném čase obnovit danou stránku, jak to udělat?

Jde to samozřejmě udělat s využitím standardních prostředků jazyka HTML a nemusíte používat žádný speciální JavaScript. Stačí do sekce <HEAD> vložit jeden META-tag:

```
<meta http-equiv="Refresh" content="1 ; URL=http://www.mojestranka.cz">
```


Číselná hodnota u atributu Content určuje, za kolik sekund bude prohlížeč přesměrován na adresu zadanou v URL. V content je nejlepší uvádět nulu, prohlížeč se na novou adresu přesměruje prakticky okamžitě.

A jak automaticky obnovit v nastaveném časovém intervalu aktuální stránku? Do URL uveďte adresu této aktuální stránky.

Máme třeba stránku nazvanou jako stranka.htm a chceme ji obnovovat každých 30 sekund. Do této stránky tedy vložte tento META-tag:

```
<meta http-equiv="Refresh" content="30 ; URL=stranka.htm">
```

Automatické přesměrování a reload stránky by mělo fungovat ve všech dnes používaných prohlížečích bez sebemenších problémů. Automatické přesměrování již po mnoho let používá Svět namodro (<http://svet.namodro.cz>).

Neviditelný okraj kolem tabulky v Netscape

Pokud tvoříte náročnější webové stránky, neobejdete se bez použití tabulek. Někdy se ale stane, že v Netscape Navigatoru v4.xx nesedí velikosti jednotlivých tabulek či jednotlivé tabulky na sebe nenavazují tak, jak bychom si přáli a mezi nimi je drobná mezera. Příčinou může být chybějící atribut BORDER, kterým se nastavuje orámování tabulky:

```
<table cellpadding="0" cellspacing="2">
<tr>
<td>První buňka</td>
</tr>
</table>
```

Netscape Navigátor totiž v takovém případě vytvoří kolem tabulky neviditelné orámování o velikosti jednoho pixelu, což je zcela proti standardu jazyka HTML. Internet Explorer se v případě neuvedení atributu BORDER v tabulce zachová zcela správně – nevytvoří žádné orámování, tedy ani to neviditelné. Pokud tedy obsahuje vaše stránka tabulky, nezapomeňte na atribut BORDER:

```
<table cellpadding="0" cellspacing="2" border="0">
<tr>
<td>První buňka</td>
</tr>
</table>
```

Jak zrušit mezeru za tagem <FORM>?

Zkušenější tvůrci HTML stránek vědí, že začátek formuláře se uvozuje pomocí tagu <FORM> a ukončuje tagem </FORM>. Problémem ovšem může být automatické odsazení začátku formuláře a konce formuláře o jeden řádek. U některých dobře graficky zpracovaných serverů to ovšem může být dost nepříjemné. Jak tuto mezeru zrušit?

Jde to velmi jednoduše – pokud je váš formulář v tabulce, jednoduše vložte startovací tag <FORM> mezi tagy tabulky, nejlépe za tag <TABLE> a před první tag <TR> a nikoli před ně. Stejným způsobem tak vložte ukončovací tag </FORM> mezi tagy ukončovací tabulky, třeba takto:

```
<table cellspacing="0" cellpadding="2" border="0">
<form>
<tr>
<td>První buňka</td>
</tr>
</form>
</table>
```

Schopnost správně odeslat obsah formuláře se tímto postupem neovlivní. Pokud takovou stránku, která obsahuje tag <FORM> mezi tagy tabulky a řádky zkontrolujete validátorem syntaxe (třeba přímo od konsorcia W3C – <http://www.w3c.org>), vyhodí zde sice chybu, ale výsledné zobrazení stránky to neovlivní.

Jak na maximální počet znaků v textovém poli?

Formuláře jsou velmi užitečná věc, pomocí nich se dá třeba odeslat vzkaz ze stránek, který přijde do vašeho e-mailu, či od návštěvníka vaší stránky získat různé doplňující údaje o něm, jako jméno, adresu, telefon (třeba u nějaké ankety) apod.

Někdy je ale potřeba omezit maximální počet znaků, které může čtenář zadat. Jde to udělat pomocí standardních prostředků jazyka HTML.

K omezení délky vloženého textu v textovém poli formuláře se používá atribut maxlength, jehož hodnota definuje maximální počet znaků, které je možné napsat. V následujícím kódu je text omezen na deset znaků:

```
<form>
<input type="Text" maxlength="10">
</form>
```

Atribut MAXLENGHT není možné použít u formulářového prvku TEXTAREA. Omezení počtu znaků v textovém poli formuláře funguje ve všech dnes používaných prohlížečích, včetně Netscape 4.xx a Opery 4.x.

Jak definovat velikost textového pole? (I.)

Pomocí atributu MAXLENGHT jsme u formulářového pole dokázali definovat maximální počet znaků, které je možné zobrazit. Nyní vám ukáží, jak definovat velikost textového pole s přesností na jeden znak.

Slouží k tomu atribut SIZE, jehož hodnotou je číslo, určující délku textového pole ve znacích. Zde jsme definovali délku pole, která je rovna deseti znakům:

```
<form>
<input type="Text" size="10">
</form>
```

Délka takto definovaného textového pole bude v Internet Exploreru odlišný od Netscape Navigátoru. Roli tu totiž hraje druh použitého písma před formulářem. Použití SIZE v tagu INPUT je v plném souladu se specifikací HTML a bude tedy fungovat ve všech dnešních prohlížečích.

Jak definovat velikost textového pole? (II.)

Definování velikosti textového pole pomocí atributu SIZE není příliš přesné a proto nastupují kaskádové styly CSS. Pomocí nich je možné nadefinovat velikost pole s přesností na jeden pixel.

V následujícím zdrojovém kódu jsem pomocí stylu nadefinoval velikost (width) všech tagů INPUT na 80 pixelů (width: 80pt;):

```
<style>
<!--
input {width: 80pt;}
-->
</style>

<form>
<input type="Text">
</form>
```

Tato definice stylu bude funkční v prohlížečích podporující řádně standard CSS1 – tedy Internet Explorer od verze 4.0, Opera od verze 4.0 a Netscape 6. V Netscape Navigátoru verze 4.xx tato definice velikosti bude ignorována a proto je vhodné zkombinovat uvedenou definici CSS s použitím atributu SIZE u tagu INPUT, který se uplatní v Netscape verze 4.xx. MS Internet Explorer tak velikost textové pole nastaví podle hodnoty uvedené v CSS, Netscape podle hodnoty atributu v SIZE.

Jak předdefinovat text v textovém poli?

V textovém poli formuláře je občas vhodné předdefinovat nějaký text – třeba u požadavku na vložení elektronické zprávy – e-mailu – vložíme @, čímž tak uživateli ušetříme práci. To uděláte velmi jednoduše použitím atributu VALUE, jehož hodnotou je právě nás předdefinovaný text:

```
<form>
<input type="Text" value="Předdefinovaný text">
</form>
```



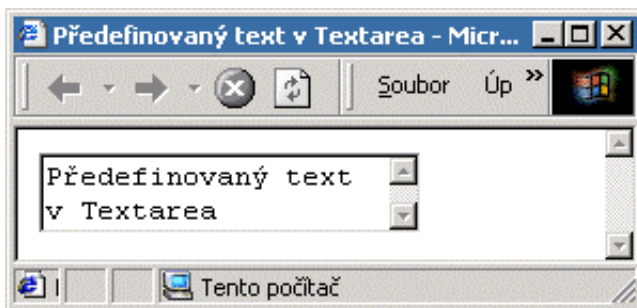
Atribut VALUE má v tomto případě ještě jeden význam – pokud je uvedena předdefinovaná hodnota a uživatel tento předdefinovaný text nezmění, odešle se tento předdefinovaný text skriptu, který zpracovává data z formuláře. Pokud předdefinovaný text uživatel změní (vloží třeba svůj e-mail), odešle se serveru ke zpracování text, který vložil uživatel. Nic vám tedy nebrání v použití předdefinovaného textu.

Jak předdefinovat text v textarea?

Zatímco normální textové pole slouží ke vložení kratšího textu, pomocí párového tagu <TEXTAREA> můžete vložit libovolně dlouhý text. Jak ovšem do TEXTAREA vložit předdefinovaný text, když zde není možné použít atribut VALUE jako u textového pole?

Protože tag <TEXTAREA> je tagem párovým, předdefinovaný text se vkládá mezi uvozující tag <TEXTAREA> a končící tag </TEXTAREA> takto:

```
<form>
<textarea>Předdefinovaný text v Textarea</textarea>
</form>
```



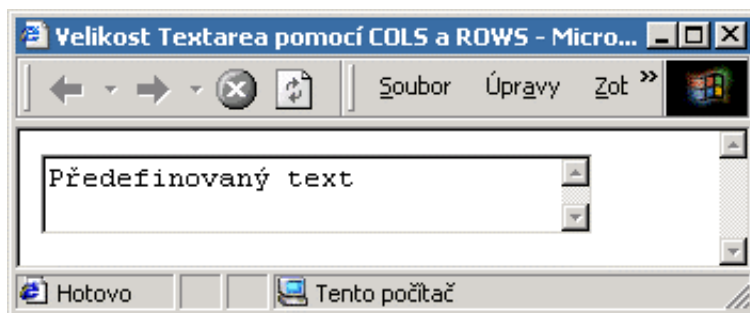
Předdefinování textu v TEXTAREA funguje ve všech dnešních prohlížečích.

Velikost Textarea - pomocí HTML

Tvůrci HTML při schvalování standardu HTML mysleli i na to, že webmasteři budou potřebovat definovat velikost formulářového prvku TEXTAREA. A jde udělat velmi jednoduše – stačí použít atributy COLS a ROWS.

Atribut COLS určuje počet znaků, které je možné v TEXTAREA napsat, než se objeví podélné rolování anebo se text zalomí na další řádek. Atribut ROWS zase určuje počet řádků textu, které je možné vložit, než se v TEXTAREA objeví svislé rolování:

```
<form>
<textarea cols="30" rows="2">Předefinovaný text</textarea>
</form>
```



Použití atributů COLS a ROWS v tagu TEXTAREA funguje ve všech dnešních prohlížečích.

Velikost Textarea s přesností na pixel – pomocí CSS

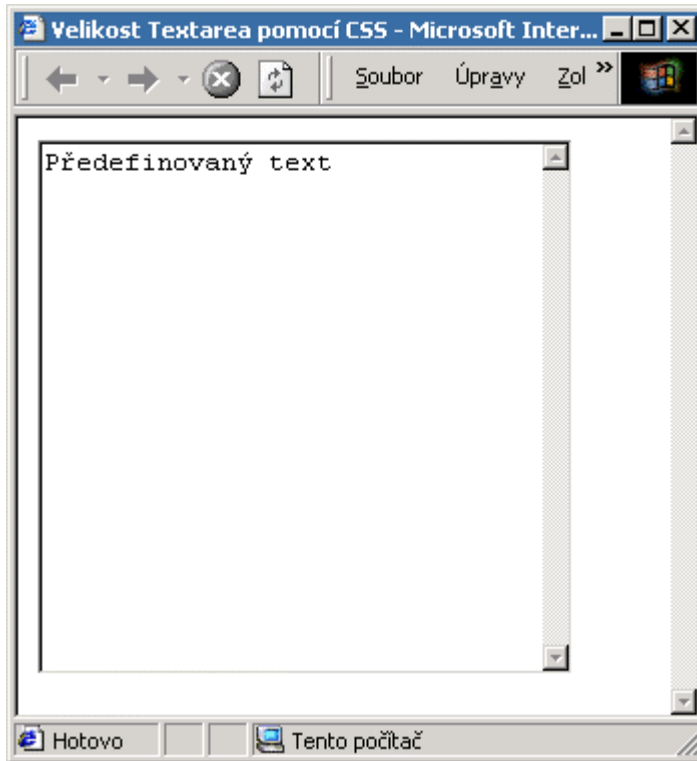
V minulém tipu jsme si uvedli postup, jak definovat velikost TEXTAREA pomocí atributů COLS a ROWS, což umožňuje definovat velikost s přesností na jeden znak ve vodorovném směru a s přesností na jeden řádek ve svislém směru. Ale jak definovat velikost TEXTAREA s přesností na jeden jediný pixel?

Velmi jednoduše s použitím kaskádových stylů CSS – stačí ve stylu definovat WIDTH (délku) a HEIGHT (výšku) pro tag TEXTAREA:

```
<style>
<!--
textarea
{
width: 200pt;
height:200pt;
}
-->
</style>

<form>
```

```
<textarea>Předefinovaný text</textarea>
</form>
```



Uvedený kód by měl být funkční v Internet Exploreru od verze 4.0, dále v Netscape Navigátoru 6 a také v Opeře od verze 4.0. V Netscape Navigátoru verze 4.xx bohužel nedosáhnete požadovaných rozměrů s přesností na jeden pixel a musíte použít atributy COLS a ROWS.

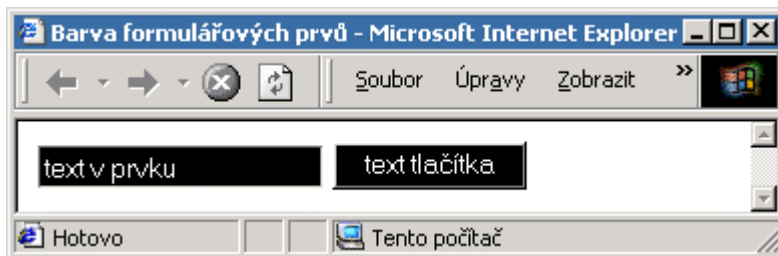
Ideální řešení pro HTML stránky je tedy kombinace atributů COLS a ROWS, pomocí kterých nastavíte velikost textového pole pro prohlížeče, které slabě podporují styly (typicky Netscape verze 4.xx), a CSS, pomocí kterého nastavíte velikost TEXTAREA pro prohlížeče plně podporující styly (IE, Opera, Netscape verze 6). Hodnoty atributů COLS a ROWS v těchto prohlížečích pak budou ignorovány ve prospěch hodnot definovaných v CSS.

Jak obarvit jednotlivé formulářové prvky

V několika minulých tipech jsem se především věnoval formulářovým prvkům – jak přednastavit určitý text, či jak definovat jejich velikost. V tomto tipu zůstanu ještě u těchto formulářových prvků. Povíme se totiž, jak u nich nastavit libovolnou barvu podkladu.

V HTML verze 3.2 nebylo barvu formulářových prvků, jako třeba u tlačítek, vůbec možné nastavit. Na stránkách tedy byla všude vidět ona "pověštná" šedá barva Windows. S příchodem HTML 4.0, respektive

s příchodem kaskádových stylů se situace radikálně změnila – pomocí CSS je totiž možné nadefinovat velké množství věcí, třeba i barvu pozadí tlačítka.



Následující zdrojový kód nastaví pro všechny tagy INPUT černou barvu (background-color:#000000;), přičemž barva textu v textovém poli (input type="Text") či barva textu na odesílacím tlačítku (input type="Submit") je bílá. Jak je vidět, nic složitého to není:

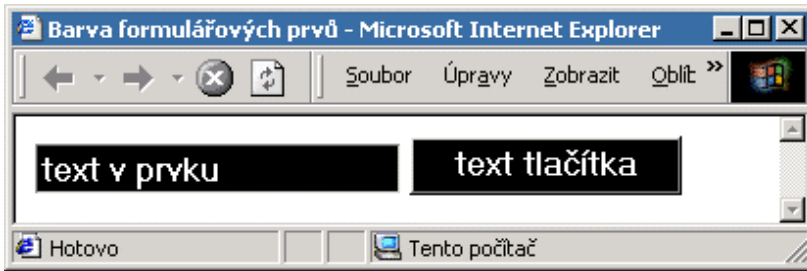
```
<style>
<--
input {background-color:#000000; color: #FFFFFF}
-->
</style>
```

```
<form>
<input type="Text" value="text v prvku">
<input type="Submit" value="text tlačítka">
</form>
```

Pomocí kaskádových stylů můžete také libovolně nadefinovat text na tlačítku či v samotném formulářovém poli. Jak na to, již ukázal výše uvedený zdrojový kód. Pomocí COLOR jsme jednoduše definovali barvu textu, pokud chcete jinou velikost písma, přidejte do stylu FONT-SIZE: a velikost (třeba 8pt); pokud chcete navíc, aby text byl zobrazen tučně, není nic jednoduššího a přidejte k definici stylu FONT-WEIGHT:bold a máte vstaraáno:

```
<style>
<--
input {background-color: #000000; color: #FFFFFF; font-size: 12pt; font-weight:bold}
-->
</style>
```

```
<form>
<input type="Text" value="text v prvku">
<input type="Submit" value="text tlačítka">
</form>
```



Pomocí kaskádových stylů jdou dělat občas opravdu úžasné věci a myslím si, že opravdu nemá smysl do detailů popisovat vše, co lze pomocí CSS udělat, nicméně se snažím popsat a vysvětlit ty postupy nejvíce použitelné na HTML stránkách. Typy uvedené v této knize vám snad také pomůžou lépe pochopit, jak styly fungují a jak se používají na www stránkách.

Výše dva uvedené zdrojové kódy budou fungovat v Internet Exploreru od verze 4.0, v Opeře od verze 4.0 a v Netscape od verze 6. Netscape Navigátor verze 4.xx formulářové prvky a tlačítka a ani text v nich nijak nenaformátuje a ponechá standardní nastavení operačního systému.

Změna barvy buňky při najetí myši

Nyní se přestaneme věnovat formulářovým prvkům a předvedeme si jednu zajímavou věc – jak pomocí událostí změnit barvu pozadí buňky tabulky.

Pokud jste tuto knihu četli poctivě od začátku, určitě více, že události, konkrétně ONMOUSEOVER a ONMOUSEOUT jsme již použili v souvislosti změny barvy odkazu. Tyto dvě události – a dají se použít dále, viz. dále v textu – použijeme stejným způsobem v tabulce a to takto:

```
<table cellpadding="2" cellspacing="0" border="0">
<tr>
<td bgcolor="blue" onmouseover="this.style.backgroundColor='red'"
onmouseout="this.style.backgroundColor='blue'">Buňka 1</td>
</tr>
</table>
```

Podle zdrojového kódu je zřejmé, že základní barva buňky je modrá (blue). Při najetí kurzorem myši na buňku se modré (blue) pozadí buňky změní na červené (red).

Této změny jsme dosáhli pomocí zápisu `onmouseover="this.style.backgroundColor='red'"`. Při oddálení kurzoru myši nastane díky `onmouseout="this.style.backgroundColor='blue'"` opačná situace a barva se změní zpět na modrou.

Možná se teď ptáte, proč zde tento tip zmiňuji, když v podstatě stejný jsem použil u změny barvy odkazu (II.). Je to prostě proto, že 99 procent čtenářů této knihy nenapadne (mě osobně to napadlo až po dlouhé době :)), že by stejný postup šel aplikovat třeba i na buňku tabulky.

V prvním odstavci tohoto tipu jsem psal, že je případně možné použít i další události místo již zmiňovaných ONMOUSEOVER a ONMOUSEOUT. Je to pravda, události, které se dají v tomto případě docela dobře použít, je poměrně dost a zde je tedy seznam těch nejzajímavějších společně s krátkým vysvětlením:

- onclick – událost nastane při kliknutí myši nad daným elementem
- ondblclick – událost nastane při dvojkliku myši nad daným elementem
- onmousedown – událost nastane při držení levého tlačítka myši nad elementem. Událost končí při uvolnění tlačítka myši.
- onmouseup – událost souvisí s událostí onmousedown, nastává při uvolnění tlačítka myši, které se stále vyskytuje nad daným elementem.
- onmouseover – událost nastane při najetí kurzoru myši nad element
- onmouseout – událost nastane umístěním kurzoru myši mimo element Souvisí s událostí onmouseover
- onkeypress – událost nastane v okamžiku, kdy je nad elementem stisknuta klávesa.
- onkeydown – událost nastane v okamžiku, kdy je nad elementem stisknuta a přidržena klávesa
- onkeyup – souvisí s onkeydown, událost nastane v okamžiku, kdy je klávesa uvolněna

Tenké orámování kolem tabulky

Trochu více zkušenosti tvůrci HTML stránek, kteří při své práci používají tabulky pro umístění textu či obrázků (nebo jiných objektů), se v tabulkách při použití atributu BORDER="1", kterým se vytváří orámování tabulky, setkají s docela nepříjemným faktem: velikost výsledné čáry kolem tabulky není totiž jeden pixel, jak by se mohlo na první pohled zdát, ale přesně pixely dva.

Typické použití atributu BORDER v tabulce je tedy takové:

```
<table cellpadding="0" cellspacing="2" border="1" bordercolor="#000000">
<tr>
<td>Buňka 1</td>
</tr>
</table>
```

Skutečného orámování tabulky pak musíme docílit specifickým postupem, který zde ve stručnosti popíši. Do tabulky, kterou chcete mít tence orámovanou totiž musíte vložit další tabulku (vnořenou). Poté do nadřazené tabulky vložíte atribut BGCOLOR a definujete barvu, která bude vytvářet barvu orámování tabulky. Ve vnořené tabulce rovněž umístíte atribut BGCOLOR a barva zde uvedená bude tvořit pozadí tabulky. Ale to není vše, nyní přijde to nejdůležitější.

Pokud se vyznáte v HTML, určitě víte, že v tabulkách je možné použít atribut CELLPADDING, pomocí kterého odsadíte okraje tabulky od samotného obsahu, který je v tabulce. A v tom je v podstatě celý trik. Máte dvě tabulky, jedna má definovanou barvu pozadí černou, tabulka vnořená má definovanou barvu pozadí bílou. A do tabulky nadřazené prostě vložíte atribut CELLPADDING="1", který odsadí vnořenou tabulku od tabulky nadřazené přesně o jeden pixel a tak vznikne ono tenké orámování tabulky:

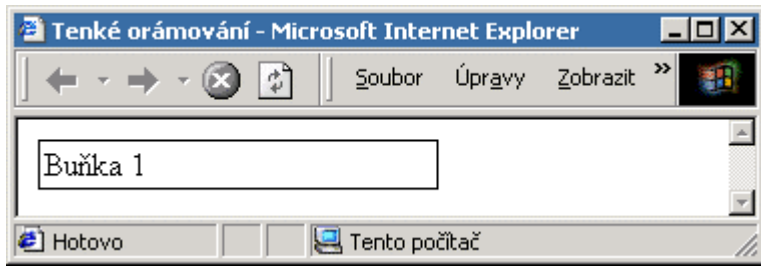
```
<table cellpadding="0" cellspacing="1" border="0" bgcolor="#000000" width="200">
<tr><td>
```

```

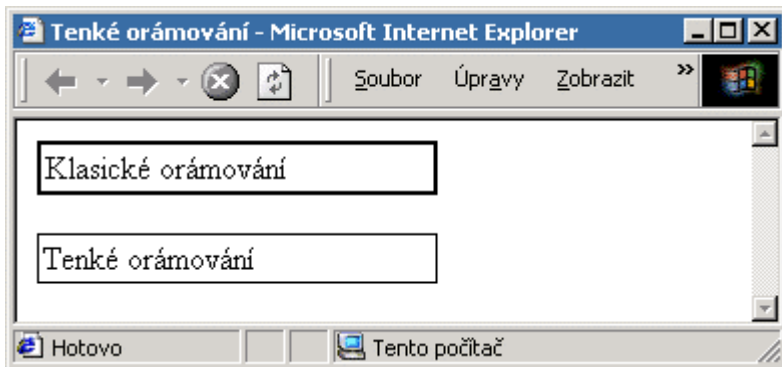
<table cellspacing="0" cellpadding="2" border="0" bgcolor="#FFFFFF" width="100%">
<tr>
<td>Buňka 1</td>
</tr>
</table>

</td></tr></table>

```



Pokud by vám náhoda nebyl stále ještě jasný rozdíl mezi popsaným postupem a standardním použitím atributu BORDER, tak vše vysvětlí následující dva obrázky tabulky. U prvního je použit BORDER="1", u obrázku druhého je použit výše popisovaný postup:



Rozdíl je zcela jasně vidět. Popisovaný postup má prostě samé výhody – tabulka má orámování skutečně 1 pixel a lze nastavit barvu orámování. Při použití atributu BORDER="1" dosáhnete v nejlepší případě orámování o síle 2 pixelů a protože v Netscape nefunguje použití atributu BORDERCOLOR, kterým jsme nastavili (pouze pro Internet Explorer) barvu orámování, orámování zde bude mít šedou barvu s nepatrným 3D efektem. Jedinou nevýhodou popsaného postupu je snad to, že musíte použít vždy dvě tabulky, čímž se trochu prodlužuje výsledná délka kódu stránky.

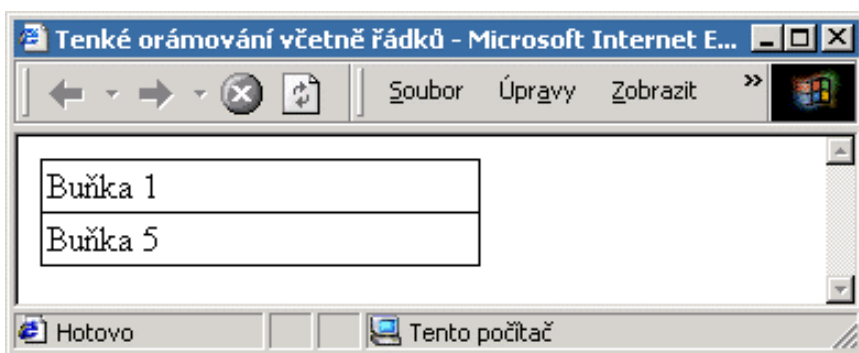
Při tvorbě takových tabulek nezapomínejte, že druhá tabulka (ta vnořená) musí mít délku tabulky nadřazené, proto je vhodné ve vnořené tabulce uvést atribut WIDTH="100%", čímž se tato tabulka natáhne na sto procent velikosti tabulky nadřazené.

Postup přitom není nijak nekorektní a funguje ve všech prohlížečích, které umí zobrazovat tabulky a znají atribut BGCOLOR, tedy prakticky všechny.

Pokud stále váháte s použitím takového postupu, koukněte se na server Lupa.cz (<http://www.lupa.cz>) nebo Interval.cz (<http://www.interval.cz>) – ty oba orámování o síle jednoho pixel kolem tabulky bez problémů používají...

Tenké orámování tabulky včetně řádku a buněk

Doufám, že se vám minulý tip o tenkém orámování tabulky líbil, protože takové tabulky na stránkách pak vypadají mnohem profesionálněji. Výše popsany postup má ovšem jednu zajímavost – a to, že vytvoří tenké orámování pouze kolem tabulky a neorámuje samotné buňky a řádky. Dlouho jsem si myslel, že vytvořit kompletní tenké orámování není možné a je tedy nutné používat atribut BORDER se všemi jeho výhodami. Ovšem, mýlil jsem se, a postupem času jsem na to přišel a je to velmi jednoduché.



Stejně jako minule se neobejdeme bez dvou tabulek, z nichž jedna bude vnořená. V té nadřazené definujte pozadí tabulky `BGCOLOR="#000000"`, uvedená barva bude současně barvou orámování. A nyní nastává první změna – atribut `BGCOLOR` ve vnořené tabulce nevkládáte do tagu `<TABLE>`, ale vložte jej do tagů pro řádky `<TR>` nebo do tagů pro buňky `<TD>`.

V předchozím tipu jsme jednopixelového orámování tabulky dosáhli použitím atributu `CELLPADDING="1"`, který byl použit u nadřazené tabulky a který odsadil okraje tabulky od samotného obsahu tabulky o jeden pixel. Tento atribut nyní ponechte na nule, vložte tedy `CELLPADDING="0"`.

Ve vnořené tabulce ale změňte atribut `CELLSPACING` z nuly na jedna, tedy `CELLSPACING="1"`. Proč? Atribut `CELLPADDING` odsazuje obsah tabulky od okrajů řádků a buněk, kdežto atribut `CELLSPACING` odsazuje mezi sebou samotné okraje řádků a buněk o zadanou hodnotu. Výsledný kód tabulky by tedy měl vypadat takto:

```
<table cellspacing="0" cellpadding="0" border="0" bgcolor="#000000"
WIDTH="200">
<tr><td>
<table cellspacing="1" cellpadding="2" border="0" WIDTH="100%">
<tr>
<td bgcolor="#FFFFFF">Buňka 1</td>
</tr>
<tr>
```

```

    <td bgcolor="#FFFFFF">Buňka 5</td>
</tr>
</table>

</td></tr></table>

```

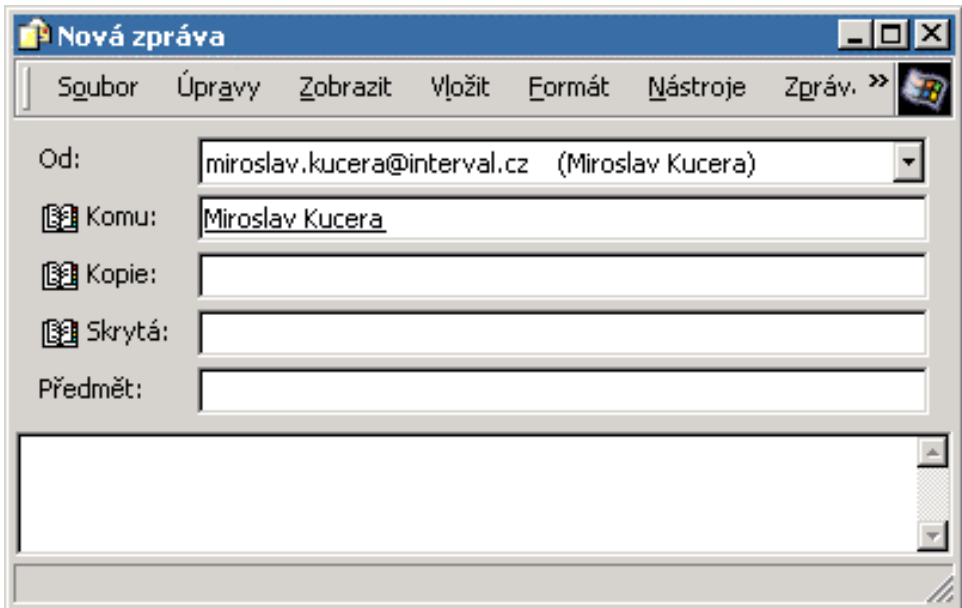
Celý vtip je prostě v tom, že pomocí atributu BGCOLOR="#000000" jsme nastavili nadřazené tabulce černou barvu pozadí. Dále, atributem BGCOLOR="#FFFFFF", který jsme použili ve všech <TD> ve vnořené tabulce (je možné ho použít i tagu <TR>), jsme nastavili pro všechny buňky bílou barvu. A nakonec, pomocí atributu CELSPACING, který má schopnost odsazovat od sebe jednotlivé řádky či sloupce, jsme tyto jednotlivé řádky a buňky od sebe odsadili o jeden pixel.

A protože jsme definovali barvu BGCOLOR pouze pro buňky <TD> a nikoli pro celou tabulku <TABLE>, která by zahrнула i tuto mezeru, není barva této mezery vůbec definována (respektive je "průhledná"). Díky tomu jsou tyto mezery mezi řádky a buňkami vyplněny černou barvou, kterou jsme definovali v nadřazené tabulce, čímž se vytvořilo tenké orámování všech buněk a řádků. Jak prostě, milý Watsone...

Pro použitelnost tohoto postupu platí to, co jsem psal minule – funguje to všech prohlížečích bez sebe-menších problémů a nevidím tedy žádný důvod, proč byste tímto způsobem nemohli tvořit tabulky i vy na svých stránkách. Použité tabulky a vůbec celkově stránky pak vypadají mnohem lépe.

Jak otevřít ze stránky e-mailovou zprávu?

Jak nejlépe umožnit čtenářům, aby se s námi zkontaktovali pomocí e-mailu? Ideální by bylo, kdyby se po kliknutí na odkaz otevřela nová e-mailová zpráva, kde by byl předefinován náš e-mail.



Není to nic obtížného, stačí použít v odkazu mailto: a samotný e-mail:

```
<a href="mailto:miroslav.kucera@interval.cz">miroslav.kucera@interval.cz</a>
```

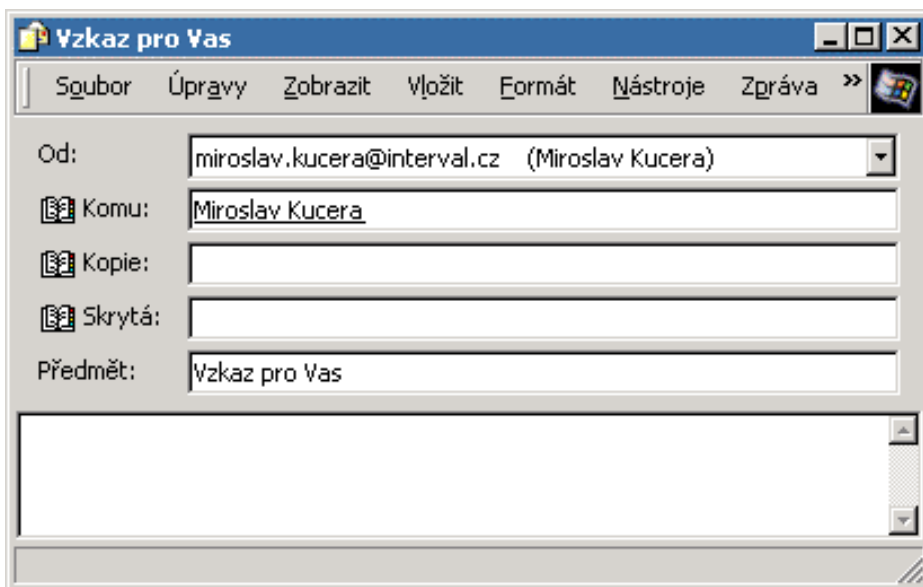
Po kliknutí na odkaz se otevře v e-mailovém klientovi nová e-mailová zpráva, kde bude jako příjemce e-mailu předem uveden miroslav.kucera@interval.cz.

Jak pomocí odkazu doplnit do e-mailu předmět?

V minulém tipu jsem vám ukázal, jako pomocí odkazu s použitím mailto: otevřít novou e-mailovou zprávu a vložit do ní předdefinovaný e-mail. Možnosti HTML ohledně e-mailu jsou mnohem větší a tak je možné předdefinovat i předmět e-mailu (subject). To se provede přidáním otazníku a klíčového slova subject:

```
<a href="mailto:miroslav.kucera@interval.cz?subject=Vzkaz pro Vas">miroslav.kucera@interval.cz</a>
```

Po kliknutí na takový odkaz se otevře nová e-mailová zpráva, kde bude v místě pro příjemce e-mailu (Komu:) uveden e-mail miroslav.kucera@interval.cz a v místě pro předmět zprávy (subject) bude uveden text Vzkaz pro Vas.

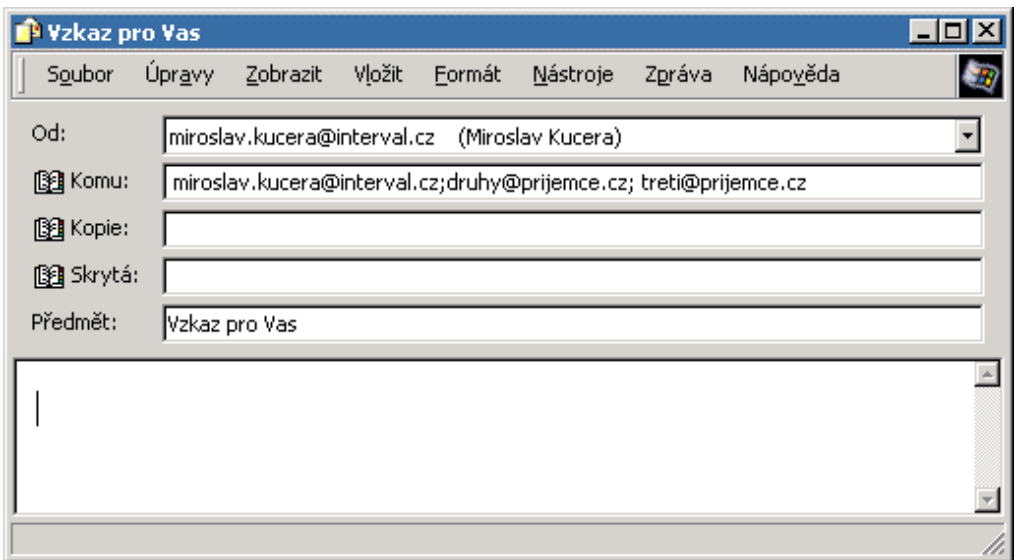


Jak na více příjemců e-mailu kliknutím na odkaz? (I.)

Nepochybuji, že předchozí dva tipy byly mnoha webmasterům notoricky známy. Nyní si ale ukážeme, jak definovat v otevřené e-mailové zprávě více příjemců e-mailu. Je to velmi triviální, jednotlivé e-maily prostě oddělujte pomocí středníku (;) počet e-mailových adres, které takto můžete zadat, je prakticky neomezený...

```
<a href="mailto: miroslav.kucera@interval.cz;druhy@prijemce.cz; tre-  
ti@prijemce.cz?subject=Vzkaz pro Vas">miroslav.kucera@interval.cz</a>
```

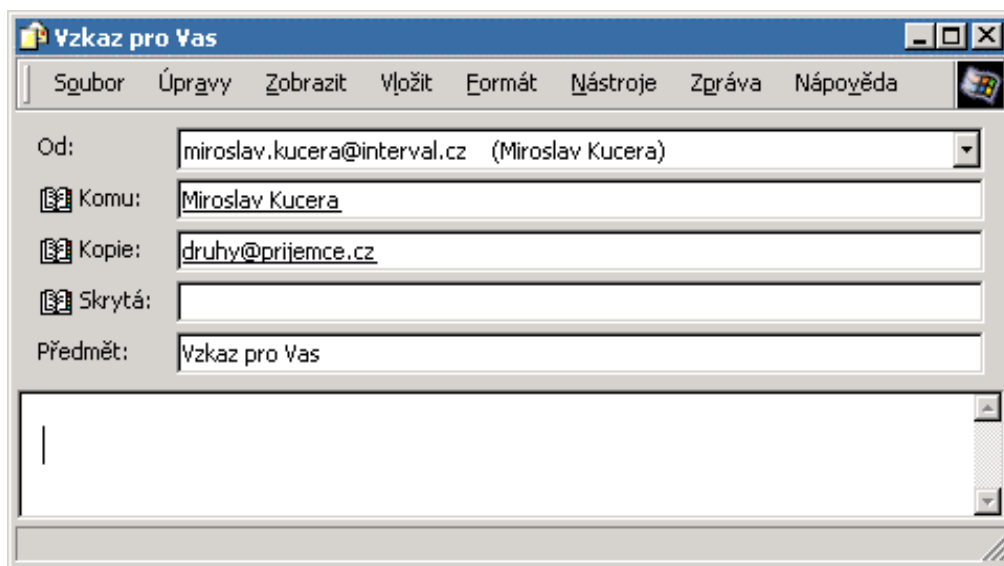
Po kliknutí na takový odkaz se otevře nová e-mailová zpráva, kde bude v místě pro příjemce e-mailu (Komu:) uveden e-mail miroslav.kucera@interval.cz a hned za ním další příjemci e-mailu, oddělení středníkem. V místě pro předmět zprávy (subject) bude uveden text Vzkaz pro Vas.



Jak na více příjemců e-mailu kliknutím na odkaz? (II.)

Použití středníku jako oddělovače jednotlivých příjemců e-mailu je sice efektivní, ale mnohem více efektivní je, když se další příjemce e-mailu vloží do kolonky Kopie. Zde stačí použít ampersand (&) a klíčové znaky cc (=carbon copy), tedy &cc:

```
<a href="mailto: miroslav.kucera@interval.cz?subject=Vzkaz pro  
Vas&cc=druhy@prijemce.cz"> miroslav.kucera@interval.cz</a>
```



Po kliknutí na takový odkaz se otevře nová e-mailová zpráva, kde bude v místě pro příjemce e-mailu (Komu:) uveden e-mail miroslav.kucera@interval.cz a v kolonce Kopie (CC) bude uveden druhý příjemce e-mailu. V místě pro předmět zprávy (subject) bude uveden text Vzkaz pro Vas.

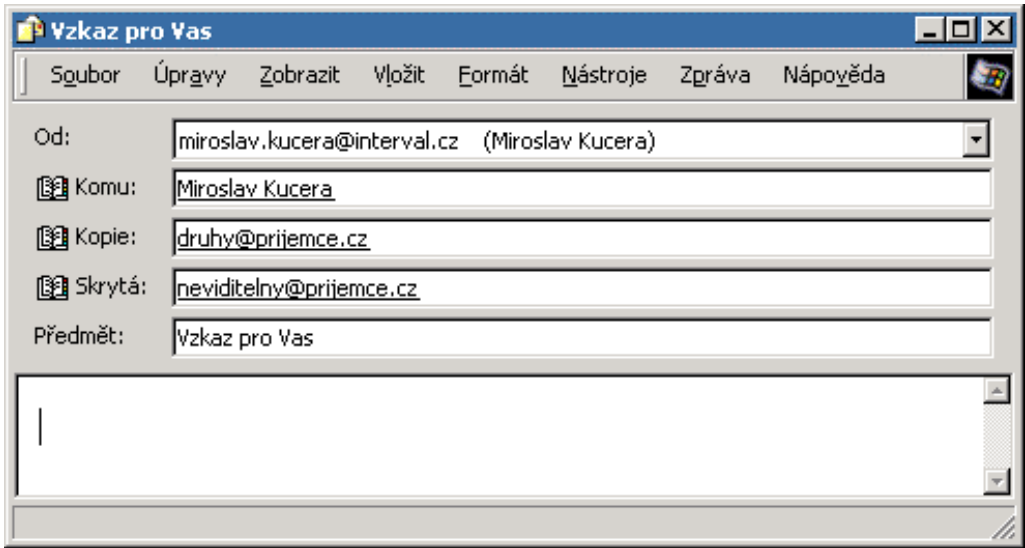
Samozřejmě, po vzoru z minulého tipu je zde také možné jednotlivé příjemce e-mailu oddělovat středníkem.

Neviditelný příjemce e-mailu kliknutím na odkaz?

Když jsem začal popisovat možnosti nastavení odkazu, který otevře novou e-mailovou zprávu, zjistil jsem, kromě výše popsanych tipů, že je dokonce možné předefinovat e-mail do neviditelné kopie e-mailu. Příjemci uvedení v kolonce Komu: a Kopie: tak neuvidí, komu mohl být e-mail ještě odeslán.

Neviditelného příjemce e-mailu připojíme k odkazu pomocí ampersandu (&) a klíčových znaků bcc:

```
<a href="mailto: miroslav.kucera@interval.cz?subject=Vzkaz pro
Vas&cc=druhy@prijemce.cz&bcc=neviditelnyprijemce.cz">miroslav.kucera@interval.cz
</a>
```



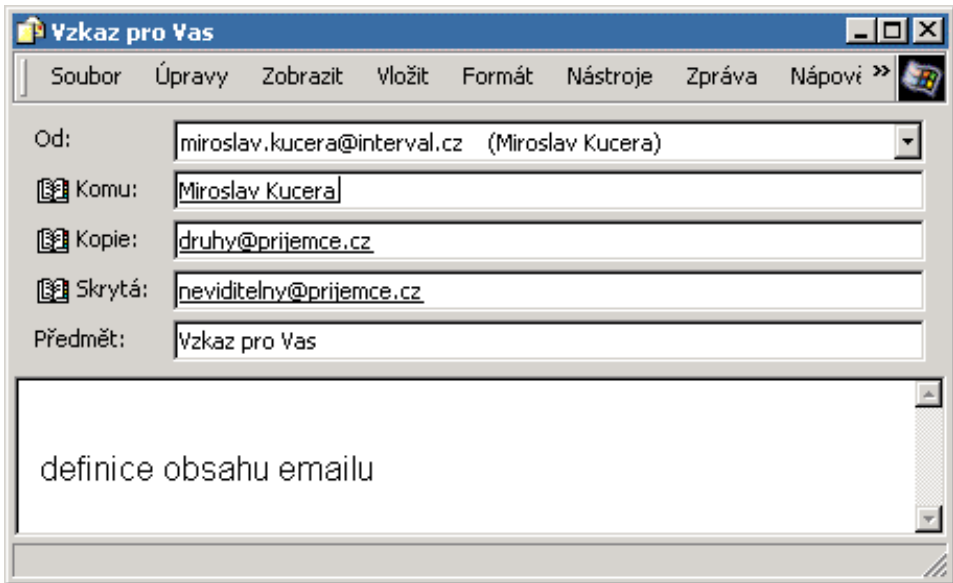
Po kliknutí na takový odkaz se otevře nová e-mailová zpráva, kde bude v místě pro příjemce e-mailu (Komu:) uveden e-mail `miroslav.kucera@interval.cz` a v kolonce Kopie (CC) bude uveden druhý příjemce e-mailu a v kolonce Skryta bude uveden neviditelný příjemce e-mailu. V místě pro předmět zprávy (subject) bude uveden text Vzkaz pro Vas.

Obsah zprávy e-mailu kliknutím na odkaz?

A máme tu poslední tip týkající se nastavení e-mailové zprávy pomocí odkazu. V odkazu je totiž možné nadefinovat kromě výše popsaných možností dokonce i samotný obsah e-mailové zprávy. Ano, zdá se to neuvěřitelné, ale opravdu to jde.

Samotný obsah e-mailu vložíme pomocí odkazu velmi snadno, použijeme ampersand (&) a hned za ním klíčové slovo body. Tedy takto:

```
<a href="mailto: miroslav.kucera@interval.cz?subject=Vzkaz pro
Vas&cc=druhy@prijemce.cz&bcc=neviditelny@prijemce.cz&body=definice obsahu
emailu">miroslav.kucera@interval.cz</a>
```

Po kliknutí na takový odkaz se otevře nová e-mailová zpráva, kde bude v místě pro příjemce e-mailu (Komu:) uveden e-mail miroslav.kucera@interval.cz a v kolonce Kopie: (CC) bude uveden druhý příjemce e-mailu a v kolonce Skrytá: bude uveden neviditelný příjemce e-mailu a samotné zprávě bude uveden nadefinovaný text "definice obsahu e-mailu". V místě pro předmět zprávy (subject) bude uveden text Vzkaz pro Vas.

Z předchozích šesti tipů, týkajících se nastavení odkazu pro předdefinování vlastností e-mailové zprávy, se jasně ukazuje, že tag <A> má spoustu nedokumentovaných funkcí. Jenom je docela škoda, že není možné pomocí odkazu připojit k e-mailu ještě nějaký soubor (přílohu), pak by tato série šesti tipů byla opravdu dokonalá. Je velmi pravděpodobné, že doplnění obsahu e-mailu, předmětu zprávy, neviditelného příjemce atd. bude funkční pouze v e-mailových klientech od Microsoftu – Outlook Express a MS Outlook.

Jak otevřít nové okno prohlížeče?

Každému tvůrci www stránek se může někdy hodit otevření nového okna prohlížeče po kliknutí na odkaz, místo aby se tato stránka otevřela ve stávajícím okně prohlížeče. Toho lze docílit použitím atributu TARGET="_blank":

```
<a href="stranka.htm" target="_blank">Nove okno</a>
```

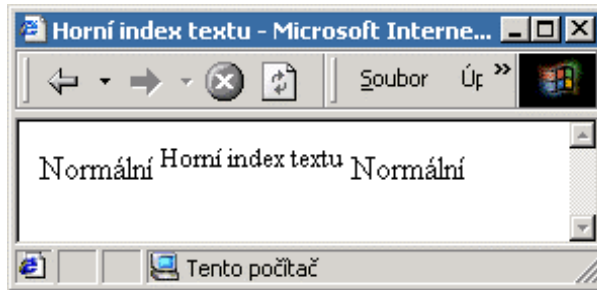
Místo hodnoty "_blank" můžete použít prakticky libovolný text. Má to význam, zejména pokud máte otevřená okna prohlížeče nějak pojmenována. V tomto případě můžete pomocí atributu TARGET následně určit, ve kterém okně prohlížeče se má stránka, na kterou ukazuje odkaz, zobrazit.

Jak vytvořit horní index textu?

Někdy potřebujete napsat text, který obsahuje horní index textu, třeba označení pro čtvereční metry – m². Mnoho tvůrců stránek si v takovém případě vypomůže stylem m2 apod. Ale není to ono, zvláště když existuje tag určený přesně pro tyto věci:

Jedná se o párový tag <SUP>. Jeho použití je velmi jednoduché:

```
<sup>Horní index textu</sup>
```



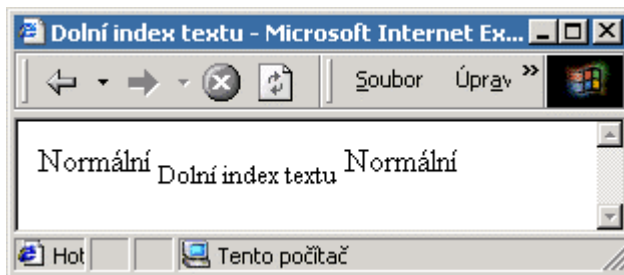
Horní index textu se bez problémů zobrazí ve dnes všech často používaných prohlížečích včetně Netscape verze 4.xx. Tag <SUP> ve specifikaci HTML 4.0 (<http://w3c.org>) zrušen nebyl a tak to vypadá, že se s ním budeme setkávat i v budoucnu.

Jak vytvořit dolní index textu?

A když už je možné vytvořit horní index textu, byl by v tom čert, kdyby nešlo vytvořit také dolní index textu. Žádný čert v tom naštěstí není a tak dolní index textu v HTML není žádný problém.

Pro zobrazení dolního indexu textu se běžně používá párový tag <SUB>. Jeho použití je jako v minulém případě stejně jednoduché:

```
<sub>Dolní index textu</sub>
```



Dolní index textu se bez problémů zobrazí ve všech dnes často používaných prohlížečích včetně Netscape verze 4.xx. Tag <SUB> ve specifikaci HTML 4.0 (<http://w3c.org>) zrušen nebyl a tak to vypadá, že se s ním budeme setkávat i v budoucnu.

Odkaz na jiné místo v dokumentu

Představte si, že má máte velmi dlouhou stránku s textem a potřebujete nějak vyřešit navigaci v rámci tohoto dokumentu. Na začátku stránky máte osnovu textu a velmi by se hodilo, kdyby po kliknutí na odkaz stránka sama narolovala na začátek definovaného dokumentu. Je to možné? Moje odpověď zní, že to možné je a navíc s využitím standardních prostředků HTML, takže to bude fungovat v každém prohlížeči.

Vložte do stránky normální odkaz, ve kterém uveďte název aktuálního souboru a za jeho příponu vložte # a nějaký text, na který se budete odkazovat, třeba #cast-prvni. Odkaz tedy bude vypadat asi takto:

```
<a href="stranka.htm#cast-prvni">První část</a>
```

Poté do dokumentu, kde chcete, aby prohlížeč po kliknutí na odkaz naroloval, vložte tag <A>, ale nikoli s atributem HREF, ale s atributem NAME, jehož hodnotou bude stejný text (ale už bez mřížky), který jste uvedli ve výše definovaném odkazu:

```
<a name="cast-prvni"></a>Zde začíná část první...
```

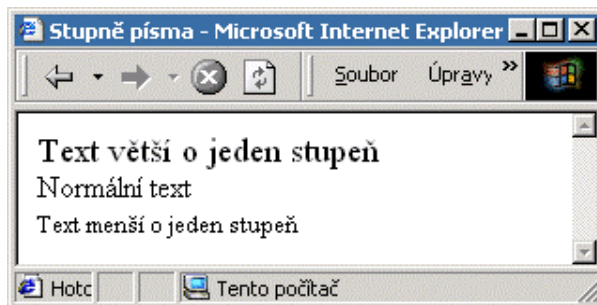
Prohlížeč pak naroluje na tuto část dokumentu. Nejsem si zcela jist, zdali u tagu <A> s atributem NAME je nutné používat ukončovací tag , já osobně ho ale v takových případech raději uvádím a totéž doporučuji i vám, tím nic nemůžete pokazit.

Rychlá změna velikosti písma na stránce

Jistě víte, že pro definici velikosti písma na stránce se v HTML používá tag s atributem SIZE="velikost". V některých případech je ovšem zbytečné používat tuto definici fontu, když existují dva tagy, které vám o jeden stupeň zvětší nebo zmenší písmo.

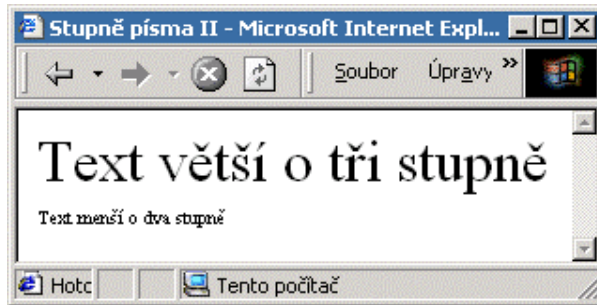
Jedná se o párové tagy <BIG> a <SMALL>. Tag <BIG> slouží pro zvětšení písma o jeden stupeň, tag <SMALL> slouží pro zmenšení písma o jeden stupeň. Jejich použití na stránce je velmi jednoduché:

```
<big>Text větší o jeden stupeň</big>
<small>Text menší o jeden stupeň</small>
```



Je zcela korektní uvést daný tag několikrát za sebou, písmo se pak zvětší/zmenší o tolik stupňů, kolikrát uvedete daný tag za sebou. Pak jej ovšem nezapomeňte vložit odpovídající počet uzavíracích tagů.

```
<big><big><big>Text větší o tři stupně</big></big></big>
<small><small>Text menší o dva stupně</small></small>
```



Tyto dva tagy vám budou fungovat ve všech prohlížečích, byly totiž obsaženy již v prvních verzích HTML a jsou uvedeny i v HTML 4.01.

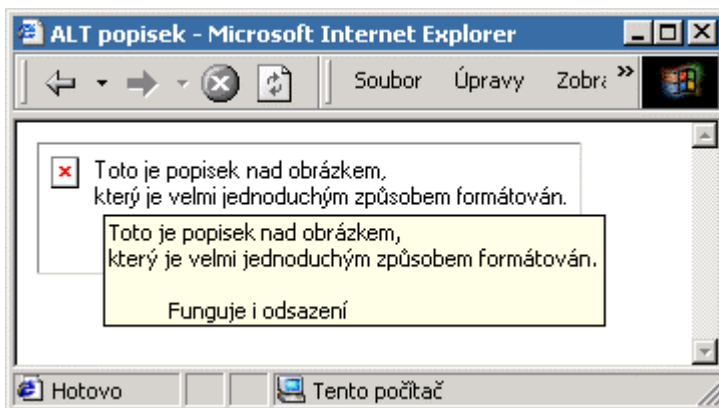
Formátování ALT popisku u obrázku

Možná tomu nebudete věřit, ale text v atributu ALT, který se používá především u obrázků, je možné velmi jednoduchým způsobem formátovat. Je možné text v ALT popisku odsazovat či zalamovat na nový řádek. Jak? Velmi jednoduše – ALT popisek pište ve zdrojovém kódu přímo tak, jak chcete, aby nakonec vypadal, třeba takto:

```

```

V uvedeném příkladu je použita klávesa ENTER pro přechod na nový řádek a několik mezer pro odsazení textu. ALT popisek nad obrázkem při najetí kurzorem myši pak ve stránce vypadá nějak takto:



Toto jednoduché formátování ALT popisku funguje plně pouze v Internet Exploreru od verze 4.0.

V Netscape 4.xx je text částečně naformátován – přechod na nový řádek prostým odřádkováním není možný, vložené mezery mezi slovy jsou nicméně respektovány.

Jak zrušit ukládání stránek do proxy?

Ukládání webových stránek do proxy souvisí s cachováním stránek. Při využití následujícího meta-tagu Cache-control máte možnost pomocí čtyř hodnot, které se uvádějí do atributu content, definovat, zdali vůbec a kde bude stránka uložena.

Možnosti jsou takové – public uloží stránku ve veřejných systémech, private uloží stránku soukromých systémech, no-cache – stránka se nesmí nikde uložit a no-store zařídí, že se stránka uloží, ale nesmí být archivována.

Pokud tedy nechce stránku nikde ukládat, použijte hodnotu no-cache:

```
<meta http-equiv="Cache-control" content="no-cache">
```

Nepohybující se "background" na stránce

Pokud používáte pro pozadí stránek nějaký obrázek (vzorek), který vyplní celou plochu okna prohlížeče a vadí vám, že tento obrázek na pozadí se posunuje společně s textem, máte možnost to změnit použitím nestandardního atributu BGPROPERTIES="fixed", který umístíte do tagu <BODY>.

Pokud to použijete, obrázek na pozadí se přestane posunovat společně s textem – bude se posunovat pouze text a vytváří to dojem, že text je jakoby "nad" obrázkem:

```
<body bgproperties="fixed">
```

BGPROPERTIES="fixed" je nestandardní a bude vám proto fungovat pouze v Internet Exploreru od verze 3.0 výše. Netscape ve verzi 4.7 toto rozšíření HTML dosud nepodporoval.

Jak definovat bezpatkový font v HTML?

V HTML není možné pro formátování textu na stránkách používat libovolný font. Respektive – možné to je, ale jakmile návštěvník vaší stránky nemá tento font, může se stát to (samozřejmě kromě toho, že se mu text zobrazí jiným písmem), že text na stránce nebude obsahovat české znaky, tedy znaky s háčky a čárkami. Jak tomu předejít? Snadno – používejte pouze následující definice písma.

Pomocí této definice definujete korektní zobrazení bezpatkového písma pro největší počet platform – defacto je to pouze platforma Windows a Linux. Ostatní, jako třeba OS/2, se používají dnes minimálně:

```
<font face="Helvetica ce, arial ce, Arial, helvetica, sans-serif">
```

Možná se ptáte, proč je v tagu FONT definováno pět druhů písma, když by mohlo stačit písmo jedno. Jedno písmo ale stačit nemůže, vtip takové vícenásobné definice je v tom, že pokud prohlížeč nenalezne v systému první uvedené písmo – v tomto případě je to Helvetica CE – bude hledat písmo uvedené jako další v pořadí (Arial CE) a pokud jej najde, zobrazí text tímto písmem. Písmo Arial se používá především u OS Windows, Helvetica hlavně na Apple a Linuxu. Písma jsou si docela podobná, ale zkušenější uživatel rozezná jeden druh písma od druhého.

Jak definovat bezpatkový font v CSS?

V minulém tipu jsem použil pro definici bezpatkového písma tag , který je ale ve specifikaci HTML 4.01 označen jako deprecated (zavržený) a neměl by se tedy používat. Pro volbu řezu, velikosti a barvy písma tedy používejte kaskádové styly CSS. Definice bezpatkového písma v CSS vypadá takto:

```
<style>
<!--
{font-family: "helvetica ce", "arial ce", "arial", "helvetica", "sans-serif";}
-->
</style>
```

Jak definovat patkový font v HTML a CSS?

A když už jsme u té definice písem na stránkách, v tomto tipu vám ukáži, jak definovat patkové písmo na stránkách. Technicky vzato – patkový font není potřeba v HTML definovat, protože obvykle, pokud žádný řez písma v zdrojovém kódu nepoužijete, použijete se právě patkové písmo. Pokud byste patkové písmo z nejrůznějších důvodů potřebovali nadefinovat, použijte ho takto:

```
<font face="Times New Roman ce, Times ce, Times New Roman, Times, serif">
```

A protože je tag v HTML 4.01 označen jako deprecated, použijeme pro definici patkového písma na stránkách raději kaskádové styly CSS:

```
<style>
<!--
{font-family: "Times New Roman CE", "Times CE", "Times New Roman", "Times",
"serif";}
-->
</style>
```

Pokud chcete vědět, proč jsou názvy písem v uvozovkách, je to proto, že víceslovné názvy písem by se takto měly uvádět. Pokud ovšem uvozovky nepoužijete, není to žádná katastrofa, prohlížeče by "měly" vědět, co jste danou definicí písma v CSS chtěli vlastně říci.

Jak definovat neproporciální font v HTML a CSS?

A máme tu poslední tip ohledně definic písma, který by vás mohl zajímat – jak definovat pro stránku neproporciální písmo, aby se všude bez problémů zobrazily "české" háčky a čárky? Velmi jednoduše:

```
<font face="Courier New CE, Courier CE, Courier New, Courier, monospace">
```

A jak na definici neproporciálního písma v kaskádových stylech? Stejně jako minulých třech tipech:

```
<style>
<!--
{font-family: "Courier New CE", "Courier CE", "Courier New", "Courier", "mono-
space";}
-->
</style>
```

Ve všech těchto definicích písma obvykle také záleží na pořadí písma, ve kterém je uvedeno. Dodržujte proto pořadí přesně tak, jak je uvedeno v této knize. Přehození písma za jiné v celkovém pořadí ovšem nepůsobí žádnou katastrofu.

Kolik znaků je možné vložit do <TITLE>?

Tag <TITLE> slouží pro definování titulku stránky – je to text, který se vám objeví v horním pruhu (obvykle modrém) pruhu prohlížeče. Za textem titulku je uváděno jméno prohlížeče. Titulek stránky je vhodné pro lepší orientaci uživatelů definovat na každé stránce jiný. Kolik znaků ale může být text v <TITLE> dlouhý? Odpověď není jednoznačná.

Maximální povolená délka textu mezi tagy <TITLE> a </TITLE> není totiž přesně známa, ale všeobecně se uvádí, že pokud je popisek stránky delší než 70 znaků, může být prohlížečem automaticky ořezán. 70 znaků pro titulek je ovšem dost, obvykle se vystačíte asi s 15-25 znaky.

Jak správně psát název třídy ve stylu?

Při používání kaskádových stylů se občas může přihodit nepříjemná věc, že styl funguje v Internet Exploreru a nefunguje v Netscape Navigátoru. Pokud jste si naprosto jisti, že i Netscape daný styl podporuje, může být problém v názvu třídy pro daný styl.

Takže, vlastní zkušenosti s používáním kaskádových stylů jsou následující – ve stylu nesmíte používat podtržítka "_", Netscape pak takový styl nezobrazí. Dále není vhodné, aby styl začínal přímo číslem. V poslední době jsem dokonce zjistil, že Netscape Navigátoru 6.0 vadí střídání velkých a malých písmen v názvu třídy. Třída stylu byla totiž v mé stránce pojmenována jako TRIDA, ale v tagu jsem se omylem odkázal na "trida" a v Netscape 6 to neběželo. Názvy "TRIDA a "trida" tedy v Netscape 6 rozhodně nejsou identické. V Internet Exploreru 5.0 a v Netscape 4.51 je to jedno a na velikosti nezáleží.

A jak to má být podle definice CSS? V názvu třídy byste neměli používat nic jiného než písmena anglické abecedy a čísla. Na českou diakritiku zapomeňte. Stejně tak byste neměli používat podtržítka "_"

a pomlčku "-", ačkoli použití pomlčky funguje ve všech prohlížečích, které podporují CSS a to včetně Netscape 4.xx a Netscape 6. Jak jsem napsal v minulém odstavci, v názvu stylu zjevně záleží i na velikosti písmen, a proto nepoužívejte VELKÁ písmena (verzálky).

Správné názvy tříd:

```
.navezstylu
.navezstylul
```

Špatné názvy tříd:

```
._navezstylu
.nazev-stylu
.nazev_stylu
.inavezstylu
.názevstylu
```

Jak definovat klíčová slova pro vyhledávací servery?

Chcete na stránce definovat klíčová slova, která by byla určena pro indexovací servery? S použitím jednoho Meta-tagu, který vložíte do sekce <HEAD> to není žádný problém. Indexovací robot těmto klíčovým slovům přiřazuje větší důležitost než slovům obsaženým ve stránce.

```
<head>
<title>Titulek</title>
<meta name="keywords" content="toto, jsou, klicova, slova, oddelena, carkami">
</head>
```

Jednotlivá klíčová slova odděluje pomocí čárky, maximální doporučený počet klíčových slov na jedné stránce se pohybuje kolem čísla dvacet.

Jak definovat popis stránky pro vyhledávací servery?

S souvislostí s předchozí definicí klíčových slov pro indexovací servery se obvykle také používá další meta-tag, který je určen pro stručný popis obsahu stránky:

```
<head>
<title>Titulek</title>
<meta name="description" content="Toto je popis stránky, stručně charakterizující její obsah">
</head>
```


Snadnější formátování textu - <PRE>

Při tvorbě stránek jistě víte, že pokud při psaní textu v HTML souboru stisknete klávesu ENTER a vytvoříte tak nový řádek, v náhledu HTML stránky se tento nový řádek vůbec neprojeví a je nutné použít tag
 nebo <P> pro nový odstavec.

Anebo jiný příklad – v HTML kódu vložíte pomocí mezerníku třeba deset mezer, nicméně, ani ty v náhledu stránky v prohlížeči neprojeví a je nutné použít tvrdou zezneru.

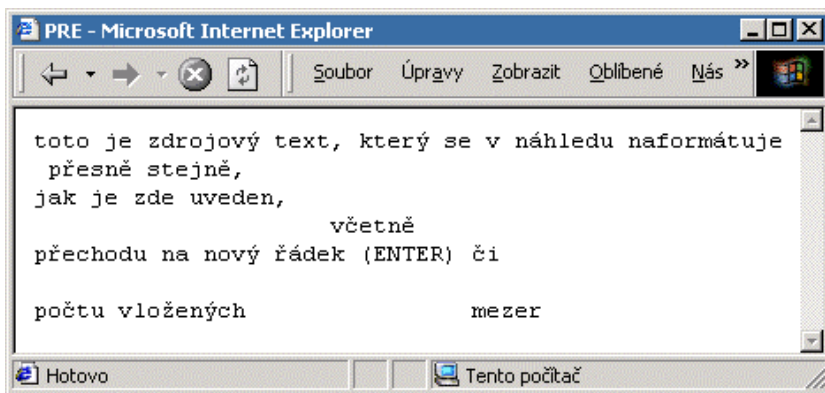
V HTML ovšem existuje párový tag <PRE>, který vám v náhledu HTML stránky v prohlížeči zachová za cenu nižšího komfortu přesně takové formátování textu, jaké je uvedeno ve zdroji, a to včetně délky řádku, počtu mezer apod.:

```
<PRE>toto je zdrojový text, který se v náhledu naformátuje
přesně stejně,
jak je zde uveden,

                včetně
přechodu na nový řádek (ENTER) či

počtu vložených                mezer</pre>
```

Text, který je uveden mezi tagy <PRE> a </PRE> je v náhledu stránky v prohlížeči zobrazen neproporcionálním fontem – obvykle je to písmo Courier. A tato to vypadá v prohlížeči:



Jak zjistit kompletní velikost stránky včetně obrázků?

Občas je potřeba zjistit, jakou má velikost vaše stránka včetně všech prvků na ní umístěných, tedy včetně obrázků, různých JavaApplet, reklam apod. Nejlepší postup, co znám a který často používám, je postup s využitím Internet Exploreru verze 5.0 - ten totiž umožňuje uložit stránku včetně všech jejích prvků.

Spusťte si Internet Explorer, navštivte stránku, jejíž velikost chcete zjistit a počkejte, až se vám kompletně načte. Poté vyberte v menu Soubor položku Uložit jako... a v roletovém menu Uložit jako typ po-nechte volbu Úplná stránka WWW a klikněte na tlačítko uložit. Stránka se vám kompletně uloží.

Nyní tento postup musíte zopakovat, ale v roletovém menu nenastavujte uložit jako Úplná stránka WWW, ale vyberte položku Stránka WWW, pouze HTML, dejte Uložit a potvrďte přepsání původního souboru. Nyní si kliknutím na vlastnosti adresáře, kde jste stránku uložili, zjistíte velikost tohoto adresáře a tím vlastně zjistíte velikost HTML stránky a všech prvků na ní.

Možná se nyní ptáte, proč jsme museli přepsat první HTML soubor, který byl uložen jako Úplná stránka WWW stránkou, která byla uložena jako Stránka WWW, pouze HTML? Vysvětlení je prosté:

Vtip je v tom, že pokud dáte uložit stránku jako úplnou stránka WWW, Internet Explorer modifikuje zdrojový kód této stránky – odřádkuje a odsadí zdrojový kód, přidá pár tagů, např. <TBODY> pro tabulku apod. Výsledkem je HTML kód, který je jinak naformátován, než ten originální, což vede k tomu, že velikost výsledného dokumentu může být třeba o 15 Kb větší! Na druhou stranu, pokud bychom stránku uložili hned jako Stránka WWW, pouze HTML, neuložily by se nám obrázky apod.

První jste tedy stránku uložili jako Úplná stránka WWW, uložil se nám modifikovaný HTML kód stránky plus všechny její části, obrázky, FLASH, reklama apod. Dále jste stejnou stránku uložili jako Stránka WWW, pouze HTML, čímž uložíte pouze HTML kód stránky v původní formě. V tomto případě Internet Explorer nijak do kódu nezasáhne a pokud jsme nezměnili název souboru, přepíšete ten původní HTML soubor. A pak pouze stačí zjistit velikost adresáře, kde máte soubory z této stránky uloženy a je to.

Jaké rozlišení pro HTML stránky?

Pokud začnete tvořit www stránky, určitě se nevyhnete otázce, pro jaké rozlišení HTML stránku vlastně vytvořit. Mám stránku vytvořit ve fixní velikosti v rozlišení 640x480, nebo v 800x600 anebo ponechat velikost stránky tak, aby se sama nastavila podle aktuální velikosti okna prohlížeče?

Pokud se rozhodnete, že vaše stránka bude mít fixní velikosti, rozhodně ji netvořte pro rozlišení 640x480 bodů – takové malé rozlišení dnes používá na počítači opravdu mizivé procento lidí. Podle posledních mně známých statistik českému Internetu dominuje rozlišení 800x600 a 1024x768. Stránky, které mají fixní velikost, tedy rozhodně tvořte pro rozlišení 800x600 bodů.

Ovšem pozor – do tabulky, pomocí které fixní rozměr nastavíte, nesmíte vložit zdánlivě ideální hodnotu WIDTH="800" pro šířku 800 pixelů, ale maximálně asi 760 pixelů. Je to proto, že prohlížeče obsahují svislou rolovací lištu, někteří uživatelé dále mohou mít Panel nástrojů Windows umístěný nikoli dole, jak je standard, ale třeba nalevo nebo napravo, což snižuje výslednou šířku okna. 760 pixelů je prostě ideální velikost tabulek pro rozlišení 800x600.

Pokud chcete mít stránky, jejichž šířka je závislá na velikosti okna prohlížeče, tedy použítí WIDTH="100%", nezapomeňte stránku vytvořit tak, aby se zobrazila bez podélného rolování v rozlišení 800x600. Mnoho serverů – i hodně navštěvovaných, např. Zive.cz – na toto zapomíná a žádný uživatel nemá rád, když musí v prohlížeči rolovat obsah stránky do stran, nehledě na to, že to nevypadá dobře.

Závěrem tohoto tipu je tedy je – pokud fixní velikost stránky (vhodné pro osobní stránky, firemní stránky apod.), tak pro rozlišení 800x600, s tím, že maximální šířka stránky bude 760 pixelů.

Pokud se rozhodnete pro dynamickou šířku stránky (vhodné zejména pro informační servery), tak nezapomeňte, že stránka se musí zobrazit v 800x600 bez podélného rolování v prohlížeči.

Jak tvořit stránky v XHTML?

Přesně 26. ledna 2000 byl organizací W3C doporučen k používání nový nástupce dnes široce používaného formátovacího jazyka HTML, který se nachází ve verzi 4.01. Je jím XHTML. Název jasně napovídá, že se jedná o standard, který využívá výhod technologií XML a HTML. XHTML je zpětně kompatibilní s HTML 4.01, což by mělo tvůrcům stránek usnadnit přechod na tento nový standard.

Tento tip nehodlá být vyčerpávajícím popisem XHTML, ale spíše vás upozorní na největší rozdíly proti HTML. Kompletní specifikaci XHTML naleznete na adrese <http://www.w3.org/TR/xhtml1/>.

Základní struktura dokumentu

Základní struktura XHTML dokumentu není zase tak příliš odlišná od základní struktury HTML dokumentu:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">

<head>
<title>Popisek</title>
</head>
<body>
<p>XHTML stránka</p>
</body>
</html>
```

A nyní si popíšeme největší rozdíly XHTML proti HTML.

Křížení tagů v HTML a XHTML

Pokud jste již nějakou WWW stránku vytvořili, určitě jste občas zkřížili – třeba i nevědomky – jednotlivé HTML tagy. Podle definice HTML mají být tagy ukončovány přesně v opačném pořadí, než v jakém pořadí jsou definovány za sebou, tedy takto:

```
<b><i><small>text stránky</small></i></b>
```

a nikoli takto:

```
<b><i><small>text stránky</i></small></b>
```

kde je tag ukončovací </SMALL> přehozen s ukončovacím tagem </I>.

V klasickém HTML dokumentu se obvykle nic vážného nestane, text se zobrazí tak, jak bylo tvůrcem myšleno. V XHTML je ovšem taková definice špatná a zcela proti syntaxi XHTML.

Pokud budou prohlížeče striktně dodržovat XHTML, může taková chyba zavinit i to, že se vám stránka vůbec nezobrazí. Pokud tedy používáte XHTML, nesmíte tagy křížit.

Malá a velká písmena v XHTML

Protože se XHTML snaží skloubit přednosti HTML a XML, je nutné dodržovat i velikost znaků u jednotlivých tagů. V HTML bylo zcela jedno, zdali jste vložili tag `` nebo ``, oba byly zcela bezchybně interpretovány jako tag pro začátek formátování tučného textu.

V XHTML to už jedno není, a proto musíte všechny tagy psát malými písmeny! V XHTML dokumentu totiž tagy `` a `` rozhodně nejsou identické. Většina HTML editorů umí naštěstí konvertovat názvy tagů na malá písmena, takže dodržování tohoto pravidla by neměl být žádný problém.

Všechny hodnoty atributů do uvozovek!

V HTML se hodnoty některých atributů dávají do uvozovek, a jiné zase ne. XHTML se nesnaží definovat, v jakých případech můžete uvozovky použít a kde naopak nemusíte. Jednoduše striktně definuje, že musíte uvozovky použít u všech hodnot atributů, a je jedno, zdali jsou číselné nebo ne. Následující zápis tagu `<TABLE>` s několika parametry by v XHTML rozhodně neprošel:

```
<table cellpadding=0 cellspacing=0 border="1" bgcolor=black >
```

Zatímco následující zápis je z hlediska dodržování pravidla o použití uvozovek v XHTML zcela korektní:

```
<table cellpadding="0" cellspacing="0" border="1" bgcolor="black">
```

Všechny tagy musí být ukončeny!

Podle mě zdaleka největší rozdíl mezi HTML a XHTML dokumentem je ten, že všechny tagy – včetně nepárových – v XHTML dokumentu musí být ukončeny. Zatímco v HTML se zdá nesmyslné ukončování nepárového tagu `
` nebo `` či ``, v XHTML již musí být striktně ukončeny i tyto nepárové tagy.

U nepárových tagů můžete použít jejich ukončovací ekvivalent, třeba `
</br>` či pouze `
`, ale zde narazíte na problém s kompatibilitou takové stránky v současných prohlížečích.

Tag `
` nebo `
</BR>` totiž Internet Explorer a Netscape Navigátor interpretují úplně odlišně. Při použití `
` první jmenovaný prohlížeč zalomí text na nový řádek. Konkurenční prohlížeče Netscape Navigator verze 4.xx tento tag ignoruje. U druhého způsobu, tj. `
</br>` je problém skoro opačný – Internet Explorer vloží do textu zalomení dvě místo jednoho a Netscape vloží jedno.

Konsorcium W3C bylo v těchto věcech dost důkladné a pro zachování kompatibility doporučuje před uzavírací lomítko u nepárového tagu vložit jednu mezeru, tedy `
`, což skutečně funguje. Stejný postup – mezeru před lomítkem tedy používejte i u ostatních nepárových tagů, například `` či ``. Správné způsoby použití tedy bude `` a ``.

Nezapomínejte, že v XHTML musíte ukončovat skutečně všechny tagy, včetně meta-tagů či tagu `<hr>`, kterým se vloží do stránky čára apod.

Zkrácený zápis není možný

Zejména u prvků formuláře používá atributy `CHECKED` nebo `SELECTED`, které zatrhnou nebo předem nastaví vybranou položku ve formuláři. V HTML je obvyklý tento zápis formuláře:

```
<form>
<select>
```

```
<OPTION value="polozka1">Položka1
<OPTION value=polozka2 selected>Položka2
</select><BR>
<input type=Radio checked>Radio button
</form>
```

Kdežto v XHTML musí být atributy CHECKED a SELECTED uvedeny jako hodnoty stejnojmenného atributu:

```
<form>
<select>
<option value="polozka1">Položka1</option>
<option value="polozka2" selected="selected">Položka2</option>
</select><BR />
<input type=Radio checked="checked">Radio button</input>
</form>
```

A to jsou prakticky všechny rozdíly XHTML oproti klasickému HTML. Používání XHTML by vám tedy nemělo dělat sebemenší problémy a ani by nemělo dělat problémy vašim čtenářům.

Shrňme si v jedné větě všechny popisované rozdíly v XHTML oproti HTML – všechny tagy musí být malými písmeny, všechny tagy (včetně nepárových) musí být ukončeny, jednotlivé tagy se nesmějí křížit, všechny hodnoty atributů musí být uvedeny v uvozovkách a zkrácený zápis některých atributů (disable, selected apod.) musí být uváděny v plném tvaru.

Protože XHTML je podstatně přísnější na dodržování pravidel syntaxe, doporučuji vaše současné HTML stránky vytvořit znovu, ale v XHTML (nikoli pouze opravovat – dá to více práce) a pravidelně syntaxi kódu kontrolovat validátorem pro XML, který lze použít i pro XHTML stránky. Validátor najdete na <http://www.w3c.org>.

Jak na částečné orámování tabulky?

V dříve uvedených tipech jsem zmiňoval dva postupy, jak v HTML dosáhnout pomocí dvou tabulek tenkého jednopixelového orámování tabulky. Někdy je ovšem nutné ve stránce použít částečné orámování tabulky, třeba orámování pouze začátku a konce tabulky, nikoli levé a pravé strany.

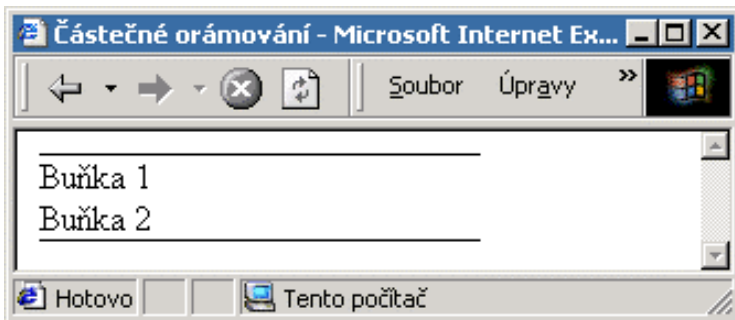
V HTML na to existuje atribut FRAME, který ale nefunguje v Netscape Navigátoru 4.xx a navíc, při použití atributu BORDER="1" se vytvoří orámování tlusté 2 pixely. Jde to ale udělat s použitím neviditelného obrázku.

Vytvoříme si tabulku, u které chceme, aby byla orámována pouze horní a spodní část tabulky. Levou a pravou část chceme bez orámování. Je to velmi jednoduché – vytvořte si prostě tabulku, která bude obsahovat dva řádky navíc, než potřebujete a do těchto řádků vložte atribut BGCOLOR="BLACK", pokud chcete mít orámování černou barvou a neviditelný obrázek 1x1.gif, pomocí kterého nastavíte výšku na jeden pixel pro tyto dvě buňky-

Ptáte se, proč píšete, abyste použili obrázek? Nestačila by třeba vložit tvrdá mezera? No, my potřebujeme stanovit výšku této buňky a ta má být přesně jeden pixel, a takové minimální velikosti se nedá dosáhnout použitím tvrdé mezery. Možná namítnete, že stačí do buňky vložit atribut HEIGHT="1", čímž stanovíme

výšku buňky na jeden pixel, ale co pak dáte do takové buňky – musí totiž něco obsahovat, jinak Netscape takovou prázdnou buňku nenaformátuje. Ideální je proto neviditelný obrázek, kterému můžete stanovit libovolné rozměry. Ale dost povídání, zde je zdrojový kód:

```
<table cellspacing="0" cellpadding="0" border="0" bgcolor="white" WIDTH="200">
<tr>
<td bgcolor="black"></td>
</tr>
<tr>
<td>Buňka 1</td>
</tr>
<tr>
<td>Buňka 2</td>
</tr>
<tr>
<td bgcolor="black"></td>
</tr>
</table>
```

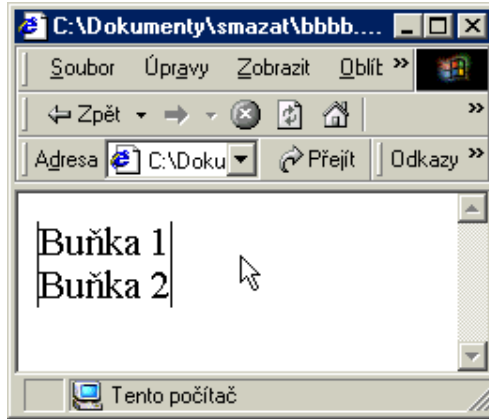


Jak je vidět, uvedený postup není nikterak složitý, pouze s buňkami navíc, do kterých vložíte neviditelný obrázek, musíte počítat již při návrhu tabulky, protože pak budete těžko dávat do již vytvořené tabulky další řádky a buňky (týká se nevizuální tvorby). Obdobným způsobem si můžete například vytvořit tenké orámování tabulky třeba pouze vlevo a vpravo:

```
<table cellspacing="0" cellpadding="0" border="0">
<tr>
<td rowspan="2" bgcolor="black"></td>
<td>Buňka 1</td>
<td rowspan="2" bgcolor="black"></td>
</tr>
<tr>
<td>Buňka 2</td>
```

```
</tr>
</table>
```

A nechybí samozřejmě ukázka, která potvrzuje, že to skutečně jde:



Výše uvedený zdrojový kód má tedy dvě buňky navíc, ty se táhnou přes dva řádky, mají nastavenou černou barvu pozadí a šířku 1 pixel, kterou jsme nastavili pomocí obrázku ``. Prostě – nic složitého.

Při použití takových tabulek se neobejdete bez vnořování tabulek, protože pokud chcete mít v takové tabulce text, měli byste jej nějak odsadit od okrajů tabulky (jinak je totiž moc nalepený k okraji a nevypadá to dobře).

Nemůžete to udělat pomocí atributů `CELLSPACING` a `CELLPADDING`, protože tyto atributy musí mít nastavenou hodnotu na "0". Jinak totiž nemáte šanci dosáhnout orámování o síle jednoho pixelu, protože byste logicky odsadili i obsah buněk, kde jsou neviditelné obrázky a vznikla by tlustá čára. Takže na místo, kde se teď nachází text " Buňka 1", vložte další tabulku, zde teprve nastavte hodnotu `CELLPADDING`, třeba na 10 pixelů (`CELLPADDING="10"`), a teprve do této tabulky vložte text, který bude odsazen 10 pixelů od okrajů tabulky.

```
<table cellpadding="0" cellspacing="0" border="0">
<tr>
<td rowspan="2" bgcolor="black"></td>

```

`<!--vnořená tabulka---->`

```
<table cellpadding="0" cellspacing="10" border="0">
<tr><td>Buňka1</td></tr></table>
```

```
</td>
```

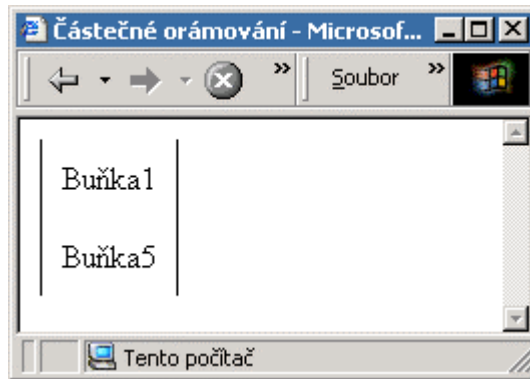
```
<td rowspan="2" bgcolor="black"></td>
```

```

</tr>
<tr>
<td>

<!--vnořená tabulka----->
<table cellspacing="0" cellpadding="10" border="0">
<tr><td>Buňka5</td></tr></table>
</td>
</tr>
</table>

```



Vícenásobné vnořování tabulek je zcela běžná praxe mnoha zkušených tvůrců HTML a nemusíte mít z toho žádný strach. Jenom se vám možná zhorší orientace v jednotlivých tabulkách, proto doporučuji před každou takovou vnořenou tabulku vložit poznámku (<!--vnořená tabulka-->).

Tento postup nevyužívá žádné speciální tagy, jedná se o normální použití tabulek, a proto uvedený postup bude fungovat ve všech prohlížečích, které umí zobrazovat tabulky a nastavit barvu buňky, což umí každý dnes používaný prohlížeč.

Jak vložit stejný text do všech stránek?

Zejména u informačních webů někdy vznikne problém, jak jeden identický text vložit do všech stránek a přitom mít tento text uložen pouze na jednom jediném místě pro rychlou editaci – třeba seznam posledních vydaných článků. Nejlepší je samozřejmě použití direktivy INCLUDE, která vloží obsah definovaného souboru do stránky přímo na serveru, ale každý nemá možnost ji použít. Ale jde to udělat i pomocí tagu IFRAME, který je obsažen v definici HTML 4.01.

IFRAME vytvoří vlastně takové okno ve stránce, kde můžete zobrazit jakoukoli jinou stránku. Od klasického FRAME se liší tím, že tag IFRAME můžete vložit kdekoli do HTML stránky (třeba i do buňky tabulky). Typické použití párového tagu IFRAME je toto:

```

<iframe src="stranka.htm" scrolling="Yes" width="100" height="100"
marginwidth="0" marginheight="0" frameborder="No"></iframe>

```


V atributu SRC definujete HTML stránku, která se má vložit. Atributem SCROLLING určíte, zdali bude možné v plovoucím rámci rolovat jeho obsah (užitečné, pokud vkládaná stránka má větší rozměry, než jsou rozměry samotného okna FRAME), možné hodnoty jsou YES (ano) a NO (ne).

Atributy WIDTH a HEIGHT určují šířku a výšku IFRAME, pomocí dvojice atributů MARGINWIDTH a MARGINHEIGHT zadáte, o kolik pixelů chcete odsadit obsah vloženého souboru od okrajů okna IFRAME, a poslední atribut FRAMEBORDER definuje, zdali má být IFRAME orámován černou čarou, možné hodnoty jsou YES (ano), 1 (ano) nebo NO (ne).

Pokud ve stránce, kterou budete vkládat do stránky pomocí IFRAME budete mít odkazy, nezapomeňte mít v těchto odkazech definovaný atribut TARGET="_TOP", který zajistí, že se stránka, na kterou odkaz vede, neotevře v přímo v IFRAME, ale otevře se v nadřazeném okně prohlížeče – tedy ve stránce, která obsahuje tag IFRAME.

Pokud tedy máte stránku iframe.htm, kde je vložen výše uvedený HTML kód a do této stránky vkládáte třeba soubor stranka.htm obsahující několik odkazů, u kterých je definován TARGET="_TOP", pak se vám odkaz neotevře v souboru stranka.htm (stránka se nezobrazí přímo v plovoucím rámci), ale otevře se v okně prohlížeče, kde máte nyní načtenou stránku iframe.htm, což je přesně to, co chceme.

Tag IFRAME není podporován Netscape Navigátorem do verze 4.7. Funguje v Internet Exploreru od verze 3.0, v Opeře je jeho použití bez problémů od verze 4.0 a v Netscape Navigátoru od verze 6.

Jak vložit stejný text do všech stránek ? (II.)

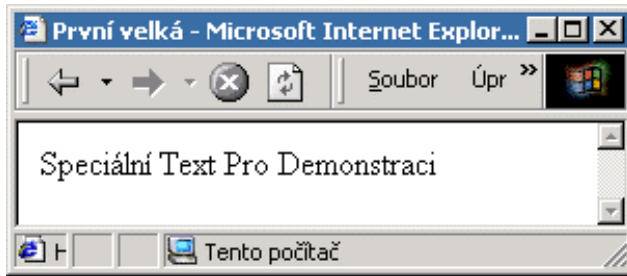
Použití tagu IFRAME nemusí vyhovovat každému – už jenom třeba proto, že tag IFRAME není podporován Netscape Navigátorem 4.xx. Existuje jiné řešení – možná trochu pracnější, nicméně docela použitelné. Většina HTML editorů, včetně těch nevizuálních, disponuje funkcí typu Extended find and replace. Co tato funkce umožňuje? Přesně to, co využijete pro úpravy statických stránek na méně rozsáhlých webech – umí totiž najít libovolně dlouhý kus HTML kódu v definovaných HTML souborech a adresářích a hledaný kus kódu nahradit zase za jiný kus kódu. Vám pak stačí takto modifikované stránky pomocí FTP poslat na web.

Jak zobrazit první písmeno slova velkým písmenem?

Pomocí kaskádových stylů je možné vytvořit i takové vychytávky, kdy můžete pro danou část textu určit, aby první písmeno každého slova bylo napsáno velkým písmenem. Stačí použít styl TEXT-TRANSFORM a hodnotu CAPITALIZE:

```
<style>
<!--
.specialnitext {text-transform: capitalize;}
-->
</style>

<span class="specialnitext">Speciální text pro demonstraci</span>
```



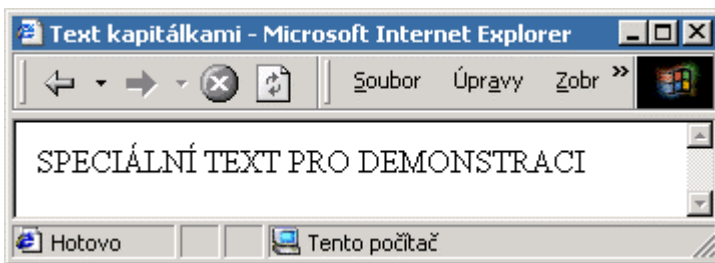
Tento příklad bude funkční v Internet Exploreru minimálně od verze 5.0 a v Opeře od verze 4.0. V Netscape Navigátoru verze 4.xx a Netscape 6 bohužel funkční není.

Jak zobrazit vybraný text velkými písmeny?

Když je možné pomocí CSS nastavit, aby se první písmena vybraných slov zobrazila velkým písmenem, pak je také možné zobrazit vybraný text velkými písmeny. Stačí použít tuto definici stylu:

```
<style>
<!--
.specialnitext {text-transform: uppercase;}
-->
</style>
```

```
<span class="specialnitext">Speciální text pro demonstraci</span>
```



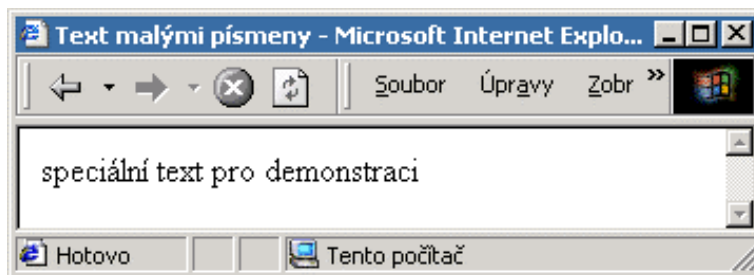
Tento příklad bude funkční v Internet Exploreru od verze 5.0 a v Opeře od verze 4.0. V Netscape Navigátoru verze 4.xx a Netscape 6 bohužel funkční není.

Jak zobrazit vybraný text malými písmeny?

V minulém tipu jsem popsal postup, jak pomocí CSS převést text na velká písmena. Samozřejmě je převést celý text i na malá písmena. Stačí použít tuto definici stylu:

```
<style>
<!--
.specialnitext {text-transform: lowercase;}
-->
</style>

<span class="specialnitext">Speciální text pro demonstraci</span>
```



I zde platí totéž, co v minulých dvou tipech - tento příklad bude funkční minimálně v Internet Exploreru od verze 5.0 a v Opeře od verze 4.0. V Netscape Navigátoru verze 4.xx a v Netscape Navigátoru 6 funkční nebude.

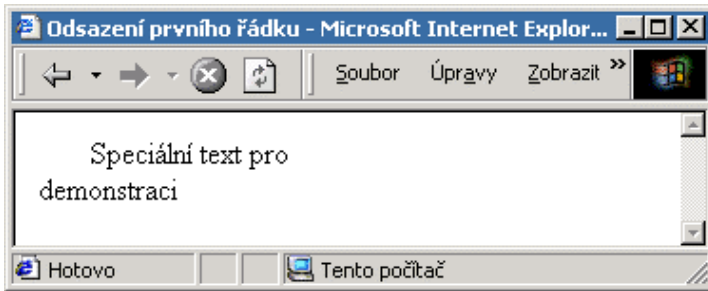
Jak na odsazení prvního řádku textu?

V mnoha knihách se používá odsazení prvního řádku odstavce od okraje stránky. Pomocí kaskádových stylů je možné tohoto efektu používat na i HTML stránkách, čímž se zpřehlední začátek nového odstavce. Stačí použít styl TEXT-INDENT a definovat velikost, o kolik se má text odsadit. Pro účely demonstrace je demonstrační text vložen do tabulky o velikosti 150 pixelů, aby se automaticky zalomil a bylo vidět, že to skutečně funguje:

```
<style>
<!--
.specialnitext {text-indent: 20pt;}
-->
</style>

<table cellpadding="2" cellspacing="0" width="150">
<tr><td>
<span class="specialnitext">Speciální text pro demonstraci</span>
</td></tr></table>
```

Výsledek pak bude vypadat takto:

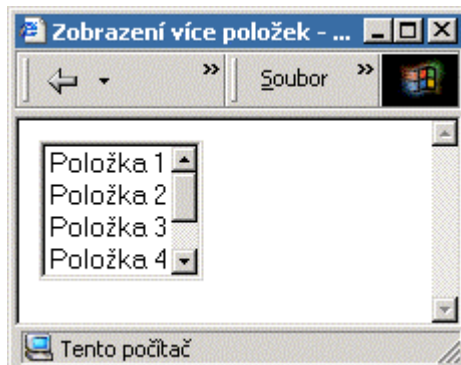


První věta se prostě odsadí o zadanou velikost. Tento příklad funguje minimálně v Internet Exploreru od verze 5.0, a v Opeře od verze 4.0. V Netscape Navigátoru verze 4.xx a bohužel ani v Netscape 6 funkční nebude.

Jak v roletovém menu zobrazit více položek současně?

Tento tip bude vhodný zejména těm tvůrcům stránek, kteří používají často formuláře na svých stránkách. Určitě znáte tag `<SELECT>`, pomocí kterého v HTML stránce vytvoříte roletové menu se seznamem položek, ze kterých je možné vybírat. Pokud u tagu `<SELECT>` použijete atribut `SIZE` a jako hodnotu uvedete nějaké číslo (třeba čtyři), zobrazí se vám na stránce, respektive v roletovém menu více položek současně (v tomto případě jsou to první 4 položky):

```
<form>
<select size="4">
<option>Položka 1
<option>Položka 2
<option>Položka 3
<option>Položka 4
<option>Položka 5
<option>Položka 6
</select>
</form>
```

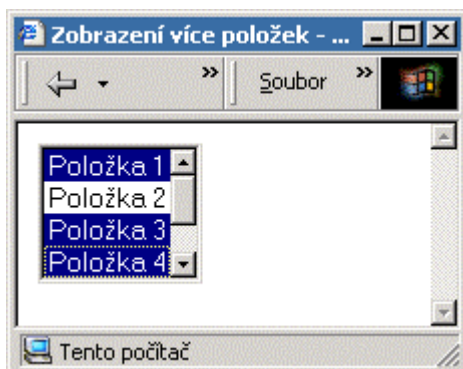


I přes to, že je v roletovém menu zobrazeno více položek současně, je možné si vybrat pouze jedinou položku. Pokud chcete, aby si uživatel mohl označit (vybrat) více položek současně, vložte do tagu `<SELECT>` atribut `MULTIPLE`.

Vícenásobný výběr se třeba používá při hledání, uživatel si tak může vybrat více kategorií, ve kterých chce vyhledat zadaný výraz.

```
<form>
<select size="4" multiple>
<option>Položka 1
<option>Položka 2
<option>Položka 3
<option>Položka 4
<option>Položka 5
<option>Položka 6
</select>
</form>
```

A takto to bude vypadat na stránce, v roletovém menu jsou vybrány tři položky – ty jsou označené modrou barvou:



Zobrazení více položek v roletovém menu současně i možnost označení více položek kurzorem myši bude fungovat ve všech dnes používaných prohlížečích bez problémů, včetně Netscape Navigátoru verze 4.xx.

Jak dostat zvuk do www stránky? (I.)

Možnost přehrávání zvukových souborů přímo na HTML stránce může být pro někoho velmi atraktivní. Pro jednoduché zvuky či hudbu je ideální použít formát MIDI. Ten je sice dnes na ústupu, ale pro www stránky je snad tím nevhodnějším formátem - zabírá totiž velmi málo místa (kvalita ale tomu také odpovídá). První možnost, kterou si popíšeme, je s použitím tagu `<BGSOUND>`

Možná jste již narazili na stránky, kde jako zpestření pro čtenáře byla použita zvuková kulisa – té snadno dosáhnete, když do sekce <HEAD> vložíte tag <BGSOUND>, ve kterém uvedete cestu k MIDI souboru a určíte počet opakování:

```
<bgsound src="soubor.mid" loop="0">
```

Atribut SRC určuje cestu a název souboru, který se má přehrát, atribut LOOP určuje počet přehrání, pokud je nastavena 0, pak se hudba bude přehrávat stále dokola. Pokud náhodnou narazíte na stránku, ke které se hudba vysloveně nehodí, použijte v prohlížeči tlačítko STOP. Použití tagu <BGSOUND> vám bude fungovat pouze v Internet Exploreru od verze 4.0.

Jak dostat zvuk do www stránky? (II.)

Existuje i ovšem další možnost, jak dostat hudbu do stránky, která se bude přehrávat v Netscape Navigatoru a nejste omezení na formát MIDI. Pomocí tagu <EMBED> můžete přehrávat nejenom spoustu zvukových formátů včetně MP3, ale také video-soubory (QuickTime a další). Typické použití tagu <EMBED> je toto:

```
<embed src="soubor.mid" width="200" height="200">
```

Atribut SRC určuje cestu a název souboru, který se bude přehrávat, WIDTH určuje šířku a HEIGHT výšku přehrávače, který se návštěvníkovi na stránce zobrazí. Pokud vložíte příliš malé rozměry, může se stát, že uživateli znemožníte třeba spuštění písničky, její zastavení, přesunutí dopředu apod.

Pokud má uživatel Internet Explorer a v systému nainstalovaný Windows Media Player, tak při vložení tagu <EMBED> se mu na stránce objeví rozhraní tohoto multimediálního přehrávače pro Windows:

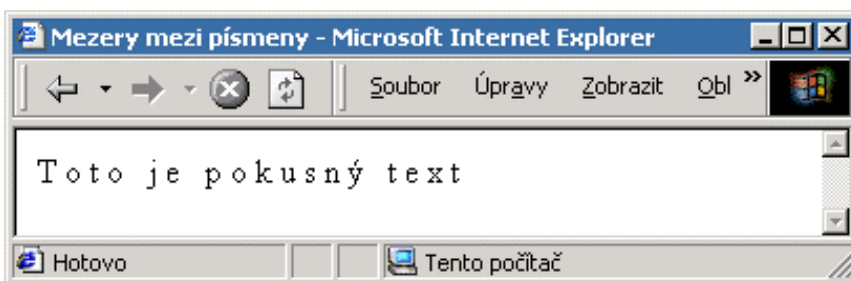


V tagu <EMBED> je možné použít atribut ALIGN pro zarovnání přehrávače (LEFT/RIGHT/CENTER), dále je možné použít atribut AUTOSTART, kde je možné jako hodnotu uvést TRUE - pak se zadaný soubor začne přehrávat zcela automaticky, pokud tento atribut neuvede nebo jako jeho hodnotu vložíte FALSE, pak se soubor začne přehrávat až po stisku tlačítka Play.

Jak změnit mezery mezi písmeny slov?

Změna velikosti mezer mezi jednotlivými písmeny ve slovech nebylo až donedávna s využitím prostředků HTML možné. Pak se ale objevily kaskádové styly CSS a jejich pomocí se dá mezera mezi písmeny libovolně nastavit. V následující ukázce zdrojového kódu je pro třídu .TEXT nastavena mezera o velikosti 4 pixelů mezi jednotlivými písmeny:

```
<style>
<!--
.text {letter-spacing:4px;}
-->
</style>
<span class="text">Toto je pokusný text</span>
```



Změna velikosti mezery mezi jednotlivými písmeny funguje v Internet Exploreru od verze 4.0, v Opeře od verze 4.0 a v Netscape od verze 6. Doporučuji velikost mezery nastavovat uvážlivě, neboť zejména u delšího textu na stránce dochází ke zhoršení čitelnosti textu. Ideální použití je pro nadpisy článků.

Jak změnit mezery mezi jednotlivými slovy?

Kaskádové styly jdou v možnostech formátování textu mnohem dále než samotná HTML a tak pomocí CSS je možné změnit nejenom velikost mezery mezi jednotlivými písmeny – viz. předchozí tip, ale je také možné změnit velikost mezery mezi jednotlivými slovy, a to obdobným způsobem jako minule.

V následující ukázce zdrojového kódu je pro třídu .TEXT nastavena mezera o velikosti 20 pixelů mezi jednotlivými slovy ve větě:

```
<style>
<!--
```

```
.text {word-spacing: 20x;}
-->
</style>
```

```
<span class="text">Toto je pokusný text</span>
```

Velikost mezery mezi jednotlivými slovy nepodporuje dnes žádný prohlížeč, ani Internet Explorer 5.0 a ani Netscape 6.

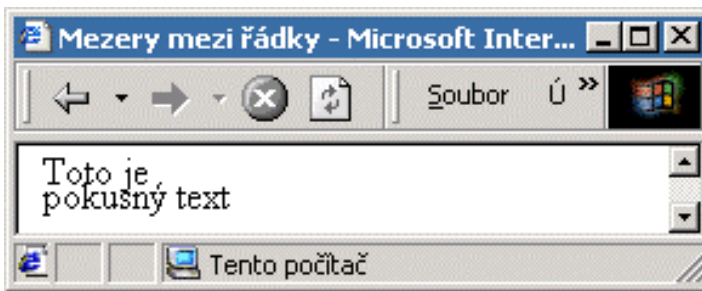
Jak změnit mezeru mezi jednotlivými řádky textu?

Nezdá se vám také, že pomocí kaskádových stylů se můžou na stránkách vyrádit i typografové? Každopádně, pomocí CSS je možné velmi jednoduše nastavit velikost mezery mezi jednotlivými řádky textu.

V následující ukázce zdrojového kódu je pro třídu .TEXT nastavena mezeru o velikosti 1 pixelu mezi jednotlivými řádky v textu. V pokusném textu je použit tag
 pro zalomení řádku.

```
<style>
<!--
.text {line-height:1px;}
-->
</style>
```

```
<span class="text">Toto je <br> pokusný text</span>
```



Dobrá zpráva – velikost mezery mezi jednotlivými řádky textu podporuje dokonce i Netscape Navigátor od verze 4.0. O podpoře v Internet Exploreru a také v Opeře se snad ani nemusím zmiňovat.

Jak na pohyující se text ve stránce?

Tento poněkud rozsáhlejší tip vám umožní na vašich stránkách snadno rozhýbat libovolný text a nejenom text – dokonce i obrázky, nebo samotné tabulky! To vše s využitím tagu <MARQUEE>.

Minimální použití párového tagu <MARQUEE> je uvedeno v následujícím zdrojovém kódu. Ten způsobí posun textu, který je definován mezi tagy <MARQUEE> a </MARQUEE>, z pravé strany směrem doleva:

```
<marquee direction="LEFT">Toto je pokusný text, který se pohybuje</marquee>
```

Tag <MARQUEE> má ovšem mnohem větší možnosti nastavení posunu textu. Při použití atributu BEHAVIOR můžete definovat, jak se má posunující text chovat. Možnosti jsou

- SCROLL – text z jedné strany vyjede, a na druhé straně zase zajede a pak zase vyjede, nebo
- ALTERNATE – text na jedné straně vyjede a na protější straně se odrazí zpátky a takhle se pořád odrazí od obou stran
- poslední hodnota, kterou je možné v atributu BEHAVIOR použít, je SLIDE, který zajistí, že text z jedné strany vyjede a jakmile dosáhne protější strany, zastaví se na místě a zůstane stát.

Následující ukázka vytvoří na stránce posunující se text, který vyjede z pravé strany směrem doleva a po dosažení levé strany se odrazí zpátky (BEHAVIOR="ALTERNATE"):

```
<marquee direction="LEFT" behavior="alternate">Toto je pokusný text, který se pohybuje</marquee>
```

Ovšem, to nejsou ani zdaleka všechny možnosti, které se dají při posunu textu nadefinovat. Za zmínku třeba stojí atributy WIDTH (šířka) a HEIGHT (výška), které specifikují šířku a výšku rámečku, ve kterém se text může posunovat. Stejnou službu vám ovšem učiní i tabulka s nadefinovanými rozměry, kde umístíte mezi tagy <TD> a </TD> tag <MARQUEE>.

V předchozích dvou tipech text vždy rotoval zprava směrem doleva. I směr posunu je možné nastavit. Text se tak může snadno posunovat nejen od pravého okraje k levému, ale i naopak a dokonce je možné nastavit, aby text roloval shora směrem dolů, anebo zezdola směrem nahoru. K tomu slouží atribut DIRECTION a jeho možné hodnoty jsou – LEFT, RIGHT, UP a DOWN.

Následující zdrojový kód vytvoří posunující se text zezdola směrem nahoru (DIRECTION="UP") v rámečku 400x40 pixelů (WIDTH="400" HEIGHT="40"). Po dosažení horní části rámečku začne text mizet a poté zase vyjede ze spodní části rámečku (BEHAVIOR="SCROLL"):

```
<marquee behavior="SCROLL" direction="up" width="400" height="40">Toto je pokusný text, který se pohybuje</marquee>
```

Barvu rámečku, ve kterém se text posunuje, je možné nadefinovat známým atributem bgcolor="barva", tedy obdobně, jako třeba u tabulky <TABLE>.

A poslední atributy, které je možné u tagu <MARQUEE> použít, je atribut SCROLLAMOUNT, dále atribut SCROLLDELAY a poslední je atribut LOOP. Co nastavují? Velmi důležité věci.

- Pomocí atributu SCROLLAMOUNT je možné nastavit vzdálenost jednotlivých "skoků" v pixelech při posunu textu. Pokud uvedete číslo 1, pak se bude text plynule posouvat po jednom pixelu, pokud uvedete třeba číslo 15, pak bude text poskakovat po 15 pixelech.
- Atribut SCROLLDELAY určuje prodlevu v milisekundách mezi jednotlivými skoky. Hodnota 1 je nejmenší.

- A poslední atribut LOOP určuje počet opakování. Pokud tento atribut neuvedete, anebo jeho hodnotu nastavíte na INFINITE, pak se bude text posouvat stále dokola (neplatí při použití BEHAVIOR="SLIDE").

Následující zdrojový kód vytvoří text, který se bude posouvat každých 10 milisekund (SCROLLDELAY="10") o 5 pixelů (SCROLLAMOUNT="5") zezdola směrem nahoru (DIRECTION="UP") v rámečku 400x100 pixelů (WIDTH="400" HEIGHT="100"). Po dosažení horní části rámečku začne text mizet a poté zase vyjede ze spodní části rámečku (BEHAVIOR="SCROLL"):

```
<marquee behavior="SCROLL" direction="up" width="400" height="100"
scrollamount="5" scrolldelay="10">Toto je pokusný text, který se
pohybuje</marquee>
```

Tag <MARQUEE> vám bude fungovat minimálně od Internet Exploreru verze 4.0, dále v Opeře od verze 4.0, a v Netscape od verze 6. V Netscape Navigátoru verze 4.xx text nebude rotovat.

Použití tagu <MARQUEE> je velmi výrazné a prakticky nepřehlédnutelné, doporučuji to tedy s jeho používáním na stránkách nepřehánět. Použití dvou tagů <MARQUEE> pod sebou, kdy tak rotují dva texty, navíc s odlišnými prodlevami, je opravdu základní školácká chyba – uživatel pak neví, který text má číst dříve, a rozhodně to nepřispívá k vytvoření dojmu profesionálního webu.

Jak na zobrazení textu v jiném rámci?

Pokud používáte na www stránkách rámce (Frame), určitě jste se už setkali s problémem, jak zobrazit obsah odkazu ve druhém rámci – tedy ne v tom rámci, kde je umístěn odkaz. Řešení je velmi jednoduché. Představte si, že máme tuto rámcovou strukturu, která vytvoří dva rámy – horní a dolní:

```
<FRAMESET ROWS="100,*">
  <FRAME NAME="Horni" SRC="horni-stranka.htm" MARGINWIDTH="10"
MARGINHEIGHT="10" SCROLLING="auto" FRAMEBORDER="0">
  <FRAME NAME="Dolni" SRC="dolni-stranka.htm" MARGINWIDTH="10"
MARGINHEIGHT="10" SCROLLING="auto" FRAMEBORDER="0">
</FRAMESET>
```

V horním rámu, ve stránce horni-stranka.htm máme samotné odkazy, jejichž obsah se má objevit ve spodním rámu. Pokud si kód prostudujete podrobně, zjistíte, že horní rám je pojmenován jako Horní (NAME="Horni") a spodní rám zase jako Dolní (NAME="Dolni") – to je potřeba k tomu, abychom se v jakémkoli odkazu na tento název mohli později dotazovat. Jak? Viz. následující HTML kód.

```
<a href="http://www.domena.cz" target="Dolni">Odkaz</a>
```

Uvedený odkaz je umístěn v HTML souboru horni-stranka.htm, který se zobrazuje v horním rámci. Díky tomu, že tento odkaz obsahuje atribut target, prohlížeč ví, že obsah nemá zobrazit ve stávajícím rámu, ale že jej má zobrazit v rámu pojmenovaném jako "Dolni", což je ten dolní. Opravdu jednoduché a funguje to ve všech prohlížečích, které podporují rámy.

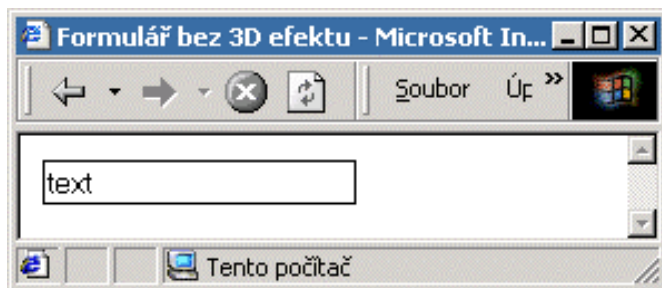
Jak se zbavit 3D stínu ve formulářových prvcích?

Klasická formulářová tlačítka na www stránkách jsou obvykle od "přírody" slabě vystínována, což vytváří dojem 3D efektu. Pokud chcete mít ovšem stránky dokonalé do nejmenšího detailu, určitě využijete tento tip, který vám 3D efekt z tlačítek odstraní.

Nadefinovali jsme si kaskádový styl pro formulářové políčko INPUT a použili jsme vlastnost BORDER, pomocí které nastavujete sílu orámování formulářového prvku. Pomocí SOLID zajistíte nepřerušované orámování prvku:

```
<style>
<!--
input {border: 1 solid;}
-->
</style>

<input type="Text" value="text">
```



Tento tip je funkční ve všech prohlížečích mimo Netscape 4.xx, kde nastávají drobné problémy – neorámkuje se totiž formulářový prvek, ale prostor pod ním. Důvod, proč tomu tak je, je v tom, že Netscape 4.xx neumí pořádně pracovat s CSS1. Pokud výše uvedený CSS styl aplikujete na formulářový prvek TEXTAREA, může se dokonce v Netscape 4.xx přihodit, že tento formulářový prvek se na stránce vůbec nezobrazí!

Jak zrušit mezeru za tagem <FORM> podruhé...

V jednom z předchozích tipů jsem popisoval, jak se zbavit mezery, kterou zcela nesmyslně vytváří tag <FORM>, pomocí toho, že jsem tento tag umístil mezi definicí tabulky <TABLE> a řádku tabulky <TR> nebo mezi definicí tabulky <TABLE> a definicí sloupce (<TD>).

Pokud je vám tento způsob proti mysli, protože to odporuje syntaxi HTML, mám pro vás i jiné řešení, a to zcela korektní – pomocí kaskádových stylů:

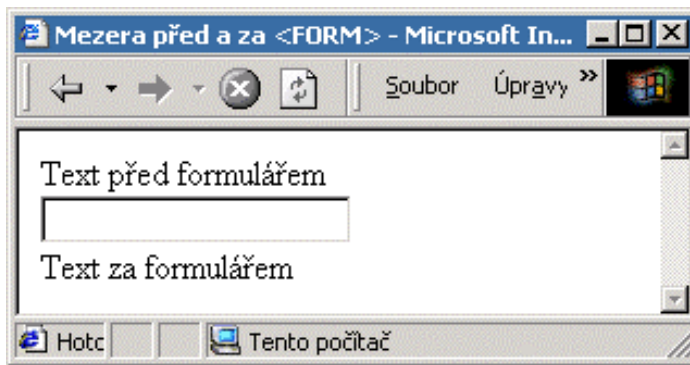
```
<style>
<!--
form {margin-top:0pt; margin-bottom:0px}
```

```
-->
</style>
```

```
Text před formulářem
<form>
<input type="Text">
</form>
Text za formulářem
```

Pomocí vlastnosti MARGIN-TOP definujeme v tomto případě velikost odsazení formuláře z horní strany, pomocí vlastnosti MARGIN-BOTTOM definujeme velikost odsazení za formulářem. V obou případech je velikost odsazení nastavena na 0 pixelů.

Na výsledek této úpravy se podívejte na obrázek:



Výše popsaný kód funguje ve všech prohlížečích mimo Netscape 4.xx a nižších verzích.

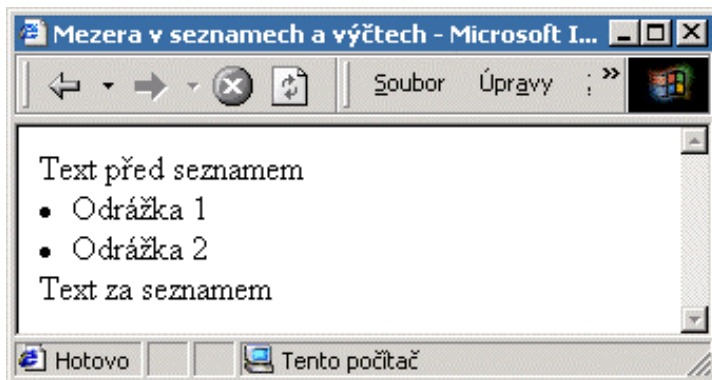
Jak zrušit mezeru v seznamech a výčtech?

Tento tip trochu navazuje na tip předchozí – pokud totiž používáte na stránkách seznamy a výčty, určitě víte, že při použití kombinace je za tagem nepříjemná mezerka, a navíc – samotné odrážky jsou odsazené od okraje. Pomocí kaskádových stylů se tomu dá lehce zabránit:

```
<style>
<!--
UL {margin-top:0pt; margin-bottom:0px;margin-left:15px;}
-->
</style>
```

```
Text před seznamem
<ul>
<li>Odrážka 1
<li>Odrážka 2
```


Text za seznamem



Zdrojový kód nedělá nic jiného, než že definuje velikost odsazení za tagem (margin-top:0pt;), dále definuje velikost odsazení před tagem (margin-bottom:0px;) a navíc, zruší odsazení zleva u odrážek (margin-left:15px;).

Ptáte, proč je u margin-left definováno odsazení 15 pixelů, když na stránce to vypadá, že tam žádné odsazení zleva není? Důvod je prostý – margin-left v tomto případě totiž nedefinuje velikost odsazení textu zleva od odrážky, ale velikost odsazení od místa, kde by text normálně začínal, pokud byste seznamy a výčty vůbec nepoužili. Pokud tedy nastavíte ve CSS stylu margin-left:0px, budete mít naprosto nulové odsazení textu a díky tomu nebude vidět ani samotná odrážka. Ostatně – vyzkoušejte si sami nastavit různé hodnoty „margin-left, uvidíte sami, jak to funguje.

Zdrojový kód by neměl být funkční pouze v Netscape 4.xx a v nižších verzích.

Jak definovat barvu posuvníku prohlížeče?

Tento tip je pouze pro ty, kterým se nelíbí standardně šedá barva rolovací lišty Internet Exploreru. Pomocí jednoduchého stylu máte možnost nadefinovat barvu rolovací lišty, barvu levého stínu lišty, barvu pravého stínu lišty, barvu horní a dolní šipky atd.:

```
<style>
<!--
BODY
{
SCROLLBAR-FACE-COLOR: #6088C0;
SCROLLBAR-HIGHLIGHT-COLOR: #6088C0;
SCROLLBAR-SHADOW-COLOR: #6088C0;
SCROLLBAR-3DLIGHT-COLOR: #6088C0;
SCROLLBAR-ARROW-COLOR: #F0F4FF;
SCROLLBAR-TRACK-COLOR: #F0F4FF;
```

```

SCROLLBAR-DARKSHADOW-COLOR: #F0F4FF
}
-->
</style>

```

Výše uvedený příklad stylu ale nenajdete v žádné definici kaskádových stylů, ani v CSS-1, a ani v CSS-2 - jedná se totiž o vlastní vylepšení Microsoftu, které funguje pouze v Internet Exploreru 5.5. V Opeře a ani v Netscape není v současné době možné definovat barvu posuvníku, nicméně se jedná o zajímavou možnost, jak zpestřit vaše www stránky pro uživatele IE 5.5.

Obrázek na stránce, ale nikoli na tiskárně

Díky kaskádovým stylům je možná velmi zajímavá věc: představte si stránku, která obsahuje reklamy (což je prakticky každá), ale tyto reklamy se přitom neobjeví na vytisknuté stránce – kdo by také klikal na reklamy na papíře, že?

Pokud tedy máte na svém serveru verzi článků pro tisk, je použití tohoto stylu ideální – reklama se zobrazuje na stránce pro tisk, ale na papíře nikoli (neruší tak při čtení). Zabijete tím dvě mouchy jednou ranou, jak se říká.

```

<style media="print">
<!--
.nezobrazit {display:none}
-->
</style>



```

Funguje to tak, že do definice stylu přidáte atribut media="print" a poté vytvoříte nějakou třídu s display:none (v našem případě je to třída „nezobrazit“) a tuto třídu přiřadíte k obrázkům, u kterých nechcete, aby se zobrazily na tiskárně. Tento styl lze pochopitelně použít i na jiné objekty než jsou obrázky, třeba na tabulky atd. Nemusíte tak třeba vytvářet žádné speciální stránky pro tisk – jednoduše všem objektům přiřadíte třídu „nezobrazit“. Anebo lepší řešení – nadefinovat styl přímo třeba pro tabulku TABLE. Pak se obsah všech tabulek – pokud nestanovíte jinak – neobjeví na vytištěné stránce vůbec.

Skript je funkční pouze Internet Exploreru 5.0, v Opeře 5 a v Netscape 6. Netscape Communicator verze 4.xx obrázek vytiskne.

Obrázek na tiskárně, ale nikoli na stránce

Pokud je možné pomocí kaskádových stylů udělat, aby se nějaký objekt, který je zobrazen na stránce neobjevil na tiskárně, tak je určitě možná i věc opačná – objekt, který se nezobrazuje na webové stránce, je přítomen na vytisknuté stránce.

Pokud porovnáme tento příklad k příkladem minulým, zjistíte, že se liší pouze atributem screen místo print. Existují pochopitelně i jiné atributy než tyto dva popsané, třeba projection či handheld, ale jejich popis by přesahoval rámec této knihy, nehledě na to, že jejich podpora v současných prohlížečích je docela špatná.

```
<style media="screen">
<!--
.nezobrazit {display:none}
-->
</style>


```

Nalezení způsobu využití tohoto stylu je již obtížnější – přesto se vám to může hodit třeba pro dodatečné přidání URL vytisknuté stránky či informace o autorských právech. Nedoporučuji ovšem pomocí výše uvedeného příkladu schovávat třeba velké obrázky atd. – pokud nějaký prohlížeč tento styl nebude podporovat, obrázek na stránce prostě zobrazí.

Co týče podpory prohlížečů, tak Internet Explorer 5 obrázek na stránce nezobrazil a na tiskárně zobrazil (správně), Opera 5 taktéž stejně jako Netscape 6. Jedinou výjimkou byl Netscape 4.6, který obrázek na stránce nezobrazil, což je správně, nicméně – nezobrazil jej ani na tiskárně, což je špatně.

Stránky bez cizích reklam na free-hostingu

Pokud umístíte své stránky na některý freehostingový server typu MujWeb (<http://www.mujweb.cz>), do vašich stránek se vám automaticky vkládají reklamy provozovatele služby. Vy hostujete stránky zadarmo na serverech provozovatele, provozovatel vám do stránek na oplátku vkládá své reklamy. Vy ovšem můžete – pomocí kaskádového stylu – jednoduše zařídít, aby se reklamy provozovatele na vašich stránkách vůbec nezobrazovaly. Nevím sice, na kolik je takové řešení v souladu s pravidly provozování takových služeb, nicméně, jsem tu od toho, abych demonstroval technické řešení a nikoli, abych řešil etické otázky.

Tento tip využívá poněkud odlišný postup než v minulém případě. Pro tag IMG nadefinujeme styl, kde použijeme vlastnost visibility s parametrem hidden a tím na stránce vyrušíme zobrazení všech obrázků.

Nějak ale musíme zobrazit vlastní obrázky, jinak by použití takového stylu nemělo smysl. Vytvoříme tedy nějakou třídu (třeba myimg) se stejnou vlastností visibility, ale s parametrem visible. Poté tuto třídu vložíme ke každému obrázku, který budeme chtít na stránce zobrazit.

```
<style>
<!--
IMG {visibility: hidden;}
.myimg {visibility: visible;}
-->
</style>


```

Problémem ale je, že místo obrázků, které se nezobrazují, vznikne prázdné místo, které se nedá nijak využít. Proto jsem výše uvedený styl vylepšil použitím vlastnosti `position` s parametrem `absolute` (absolutní polohování) a definoval souřadnice (`top:0` a `left:0`) místo vložení všech těchto neviditelných obrázků a to mimo zobrazovací plochu prohlížeče, čímž žádné prázdné místo na stránce nevznikne:

```
<style>
<!--
IMG {visibility: hidden; position: absolute; top: 0; left: 0;}
.myimg {visibility: visible}
-->
</style>


```

Skript byl testován v Internet Exploreru 5.0, v Opeře 5, v Netscape 6, kde bez problémů fungoval, a také v Netscape 4.6, kde prostě nefungoval (respektive, všechny obrázky zobrazil, což je špatně). Netscape 4.xx prostě neumí pracovat se styly tak, jak bychom si to přáli a jak to umí nové verze prohlížečů.

Všechny odkazy do nového okna

U některých typů stránek někdy potřebujete, aby se primárně všechny odkazy otevíraly do nového okna a nikoli do okna stávajícího. Vkládat ovšem atribut `TARGET="_BLANK"` do každého odkazu na stránce je docela pracné, lepší je využít tag `<base>` s využitím atributu `target`, který všechny odkazy otevře v novém okně.

```
<!doctype html public "-//W3C//DTD HTML 4.0 Transitional//EN">
<html><head>
<title>Popisek</title>
</head>
<body>
<base target="_blank">
<a href="http://www.interval.cz">Odkaz do nového okna</a>

</body>
</html>
```

Zdrojový kód byl testován a je funkční v Exploreru 5.0, Opeře 5, Netscape 4.6 a v Netscape 6.

HTML stránka se nezobrazuje v Netscape?

Pokud vzhled svých stránek ověřujete i v jiných prohlížečích, než je Internet Explorer, třeba v Opeře nebo v Netscape Navigátoru, občas se vám může stát, že stránka se nechce a nechce zobrazit v posledním jmenovaném prohlížeči. Důvodů, proč tomu tak je může být pochopitelně více, ale pokud používáte tabulky,

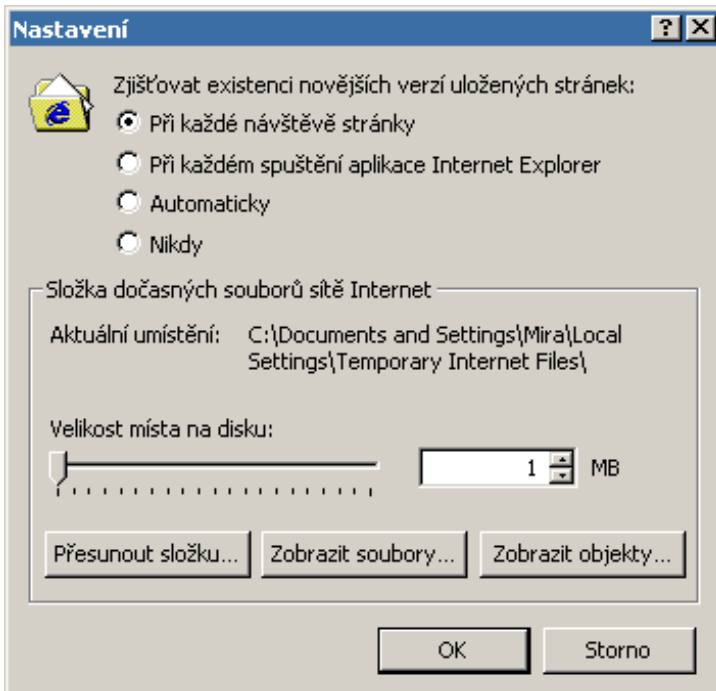
může být problém prostě v nich – Netscape totiž není schopen zobrazit stránku, kde není tabulka <TABLE> korektně ukončena tagem </TABLE>.

Jiné prohlížeče vám v takovém případě vzhled stránky obvykle úplně přeformátují, Netscape vám ji pro jistotu vůbec nezobrazí. Takže – až se vám stránka nebude chtít zobrazit v NN 4.xx, ověřte si ve zdrojovém HTML kódu, zdali máte všechny tabulky opravdu korektně ukončeny.

Cachování stránek v Internet Exploreru

Pokud se vám v Internet Exploreru objevují stále neaktuální verze vašich nebo cizích stránek, může být problém v samotném nastavení vašeho prohlížeče, který stránky neustále načítá z cache. Změňte si nastavení v menu Nástroje – Možností sítě Internet a v části Dočasné soubory sítě Internet klikněte na tlačítko Nastavit a zde vyberte položku Při každé návštěvě stránky.

Internet Explorer tak bude vždy kontrolovat, zdali není na webu aktuální verze stránky. Volba Automaticky či Při každém spuštění aplikace Internet Exploreru není příliš účinné.



Občas se také hodí složku Temporary Internet Files ručně promazat, případně snížit vyhrazenou velikost místa na disku pro tyto dočasné soubory na co nejmenší možnou velikost.

Obrázky na stránce bez zbytečných mezer

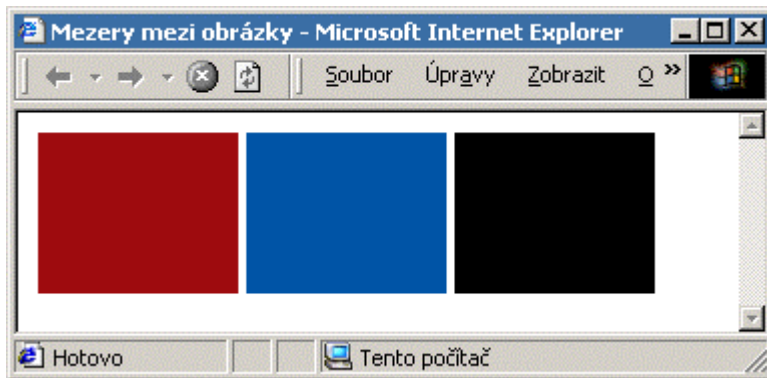
Pokud už nějakou dobu tvoříte HTML stránky, určitě rutinně používáte na stránkách obrázky, které pomocí tabulek umísťujete do stránek a znáte tedy některá úskalí, která při této činnosti mohou nastat. Pro ty ostatní tu je tento tip, který vám poradí, jak se zbavit zbytečných mezer mezi jednotlivými obrázky. Ty jsou nežádoucí třeba v případě, když chcete mít na stránce nějakou klikací mapu, kde musí jednotlivé části obrázku na sebe přesně navazovat. Podívejte se na tento kód:

```
<table cellspacing="0" cellpadding="0" border="0">
<tr><td>



</td></tr>
</table>
```

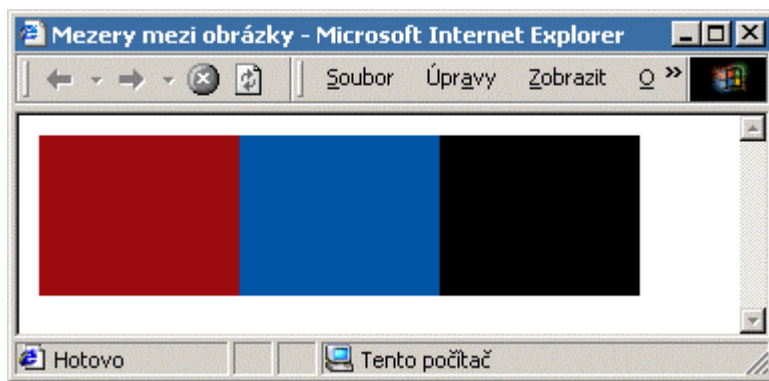
Při zkoumání kódu zdánlivě neexistuje žádný problém – ten se ovšem objeví v okamžiku, když si stránku načtete do prohlížeče, mezi jednotlivými obrázky jsou totiž nechtěné mezery:



Problém je především v tom, že ve výše uvedeném HTML kódu je každým vloženým obrázkem stisknut ENTER (odřádkování). V HTML by sice mělo být jedno, jak je zdrojový kód dokumentu naformátován, nicméně – jak je vidět, v tomto případě to jedno není. A jaká je náprava? Jednoduše vložte všechny tři tagy IMG za sebe, bez mezery či ENTERu, a nezapomeňte je také „přilepit“ k tagům <TD> a </TD>:

```
<Table cellspacing="0" cellpadding="0" border="0">
<tr><td></td></tr>
</table>
```

A při kontrole v prohlížeči to vypadá přesně tak, jak potřebujeme:



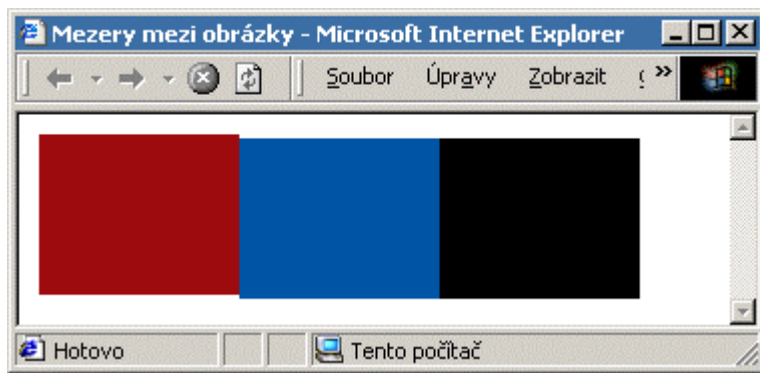
Existuje i druhá cesta, která zajistí, aby jednotlivé části obrázku do sebe přesně zapadaly – stačí umístit každou část obrázku mezi tagy `<TD>` a `</TD>`. Tento způsob je mnohem lepší – zdrojový kód je sice o trošičku větší a komplikovanější, nicméně máte jistotu, že to bude ve všech prohlížečích vypadat stejně a navíc – zdrojový kód je mnohem přehlednější:

```
<Table cellspacing="0" cellpadding="0" border="0">
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</table>
```

Nezapomínejte ovšem, že jednotlivé tagy `` musí být „přilepeny“ k jednotlivým tagům `<TD>` a `</TD>` – pokud tomu tak nebude, obrázky budou vůči sobě navzájem posunuty ve svislém směru a v Netscape 4.6 bude mezi nimi navíc rušivá mezera. Demonstruje to následující kód, kde první obrázek není „přilepen“ k `<TD>` a `</TD>`:

```
<Table cellspacing="0" cellpadding="0" border="0">
<tr>
<td>

</td>
<td></td>
<td></td>
</tr>
</table>
```



Pokud tedy používáte obrázky na stránkách a nechcete mít mezi nimi žádné rušivé mezery, umístíte obrázky (respektive tagy IMG) hned za sebe, na jeden řádek, bez mezer či odřádkování, anebo každý obrázek vložte do samostatné buňky tabulky.

Přehled barev

Základních 16 barev RGB (VGA)					
RGB	název	český význam	RGB	název	český význam
FFFFFF	white	jasná bílá	C0C0C0	silver	stříbrná
FFFF00	yelow	jasná žlutá	808000	olive	olivová
FF00FF	fuchsia	jasná fialová	800080	purple	purpurová
FF0000	red	jasná červená	800000	maroon	základní červená
00FFFF	aqua	jasná tyrkysová	008080	teal	základní tyrkysová
00FF00	lime	jasná zelená	008000	green	základní zelená
0000FF	blue	jasná modrá	000080	navy	námořnická modř
808080	gray	šedá	000000	black	černá
Rozšířený vzorník 64 barev					
RGB	název	český význam	RGB	název	český význam
FFFFF0	ivory	slonová kost	DC143C	crimson	karmín
FFFAFA	snow	sněhobílá	D2691E	chocolate	čokoládová
FAEBD7	antiquewhite	omšelá bílá	CD5C5C	indianred	indiánská
F5F5F5	whitesmoke	kouřově bílá	B22222	firebrick	cihlová červená
F5F5DC	beige	běžová	A52A2A	brown	hnědá
F0FFFF	azure	azurová	8B4513	saddlebrown	okrová
E0FFFF	lightcyan	světlá tyrkysová	8B0000	darkred	tmavě červená
7FFFD4	aquamarine	akvamarin	FFC0CB	pink	růžová
00FFFF	cyan	jasná tyrkysová	FF1493	deeppink	hluboká růžová
FFFFE0	lightyellow	světle žlutá	D3D3D3	lightgray	světle šedá
FFD700	gold	zlatá	A9A9A9	darkgray	tmavě šedá
F0E68C	khaki	khaki	008B8B	darkcyan	tmavě tyrkysová

Tipy a triky od profesionálů

A9FF2F	greenyellow	zelenožlutá	FF00FF	magenta	fuchsiová
FF8DC	cornsilk	kukuřicová	EE82EE	violet	fialová
9ACD32	yellowgreen	žlutozelená	ADD8E6	lightblue	světlá modř
90EE90	lightgreen	světle zelená	9400D3	darkviolet	tmavá fialová
8FBC8F	darkseagreen	mořská zelená	85008B	darkmagenta	tmavá fuchsiová
2E8B57	seagreen	temné moře	8A2BE2	blueviolet	modrofialová
006400	darkgreen	tmavě zelená	5F9EA0	cadetblue	šedomodrá
FFA500	orange	oranžová	4B0082	indigo	indigová
FF8C00	darkorange	tmavá oranž	483D8B	darkslateblue	tmavá ocelová
FF7F50	coral	červená korálová	4682B4	steelblue	ocelová modř
FF6347	tomato	rajčatová červená	00BFFF	deepskyblue	modrá obloha
F08080	lightcoral	světlá červen	00008B	darkblue	tmavě modrá

Rejstřík

3

3D efekt, 19
3D stín, 67

A

a: hover, 12, 14
absolutní polohování, 72
ACCESSKEY, 5
aktuální verze stránky, 73
ALIGN, 7, 20, 22
ALINK, 12
ALT, 44
Arial, 46
automatické odsazení, 25
automatické přesměrování, 25
autorská práva, 71
AUTOSTART, 63

B

background-color, 31
barva odkazu, 32
barva odkazů, 12
barva podkladu, 30
barva pozadí buněk, 21
barva rolovací lišty, 69
barvu orámování, 34
barvy, 76
barvy odkazu, 12
barvy odkazů, 12
bezpatkové písmo, 45
BGCOLOR, 21, 33
BGPROPERTIES, 45
BGSOUND, 61
BIG, 43
bitmapový obrázek, 10
BMP, 10
BORDER, 18, 25, 33, 35, 67
BORDERCOLOR, 34
BORDERCOLORLIGHT, 19

C

cache, 24, 73
Cache-control, 45
cachování stránek, 73
carbon copy, 38
CELLPADDING, 33, 55
CSS, 15, 20, 47
CSS1, 27

Č

částečné orámování, 53
české znaky, 23
čitelnost, 15
čitelnost dokumentu, 16
čitelnost textu, 63

D

definice písma, 45
délka textu, 47
dlouhé tabulky, 16
dočasné soubory, 73
dolní index, 42
domovská stránka, 5

E

e-mail, 36
e-mailová zpráva, 36, 37
EMBED, 62
ENTER, 49
Extended find and replace, 57

F

FLOAT, 20
font, 45
FONT, 46
font-size, 16
FONT-SIZE, 31
FONT-WEIGHT, 31
FORM, 25, 67
formátování textu, 49
formuláře, 25
formulářové políčko, 67
formulářové prvky, 30
formulářový prvek, 67
Frame, 66
free-hosting, 71

H

háčky a čárky, 47
HEAD, 23
Helvetica, 46
hodnoty atributů, 52
horní index, 42
hudba, 61

Ch

CHECKED, 53

I

ICO, 10
IFRAME, 56
ikona, 9
INCLUDE, 56
INPUT, 27, 31, 67
ISO 8859-2, 24

J

JavaScript, 13
jednopixelové orámování tabulky, 35

K

kapacita přenosové linky, 24
kaskádové styly, 15, 20, 46, 47
klávesová zkratka, 7

klávesové zkratky, 5
klíčová slova, 48
kódování, 23
kódování Windows-1250, 23
Kopie, 38
kurzor, 11
kurzor myši, 5, 12, 14, 32, 44

L

LEFTMARGIN, 3, 4
LINK, 12

M

mailto:, 37
malá písmena, 58
MARGINHEIGHT, 3
MARGINWIDTH, 3
MARQUEE, 64
maximální počet znaků, 26
META, 23
mezera, 68
mezery, 53, 67
mezery mezi písmeny, 63
mezery mezi řádky textu, 64
mezery mezi slovy, 63
MIDI, 61
Možností sítě Internet, 73
MujWeb, 71
multimediální přehrávač, 62

N

načítání stránky, 16
nadtržení, 15
NAME, 43
navigace, 43
název třídy, 47
názvy barev, 12
názvy písem, 46
nepárové tagy, 52
nepodtržený odkaz, 13
neproporciální písmo, 47
neuzavírání párových tagů, 23
neviditelné kopie e-mailu, 39
neviditelný obrázek, 18, 20, 53
nová buňka, 23
nové okno, 72
nový odstavec, 49
nový řádek, 23

O

oblíbené položky, 7
obrázky, 71
obsah e-mailové zprávy, 40
ocdbclick, 33
odkaz, 13, 43
odkazy, 12, 72

odsazení, 69
odsazení prvního řádku, 59
odsazovat řádky či sloupce, 36
okno prohlížeče, 41
okraj tabulky, 33
okraje stránky, 3
onclick, 33
onkeydown, 33
onkeypress, 33
onkeyup, 33
onmousedown, 33
onmouseout, 33
ONMOUSEOUT, 13, 32
onmouseover, 33
ONMOUSEOVER, 13, 32
onmouseup, 33
Opera, 4
orámování, 25, 33
orámování tabulky, 53
označit více položek, 61

P

patkové písmo, 46
počet přehrání, 62
počet řádků textu, 29
podtržení odkazu, 14
podtržitko, 47
pohybující se text, 64
pomlčky, 48
popisek, 44
popisek stránky, 47, 48
posuvník, 69
použití tabulek, 25
pozadí stránek, 45
pozadí tabulky, 33
prázdné buňky, 17
PRE, 49
print, 71
prohlížeč Opera, 4
prohlížeče, 72
proxy, 45
předmět e-mailu, 37
přesměrování, 24
přeškrtnutí odkazu, 15
Přidat oblíbenou položku, 7
příjemce e-mailu, 37, 38
příloha, 41
pseudotřída, 13

R

rámce, 66
reklamy, 70
reload, 24, 25
RGB, 12, 76
roletové menu, 60
rolovací lišta, 69
rolování, 29
rozlišení, 50
rozměr buňky tabulky, 18
rozměr tabulek, 19

Ř

řádky textu, 64
řez písma, 16, 46

S

screen, 71
SELECT, 60
SELECTED, 53
seznamy, 68
SMALL, 43
staré verze prohlížečů, 4
Start Page, 6
stejný text, 56
stínování rámečku, 19
subject, 37
SUP, 42
syntaktická pravidla, 23
syntaxe, 26, 53

T

tabulka vnořená, 33
tabulky, 7
tabulky vedle sebe, 8
tenké orámování, 35
text na tlačítku, 31
TEXT-ALIGN, 20
TEXTAREA, 26, 28, 29
textové pole, 26
tisk, 70
tiskárna, 70
TITLE, 5, 47
tloušťka orámování, 18
TOPMARGIN, 3, 4
tvrdá mezera, 18, 53

U

události, 32
ukončovací tag, 43
ukončování tagu, 52
ukončování tagů, 23, 51

uložit stránku, 49
umístit prezentaci, 22
URL, 71
uvozovky, 52
uzavírací lomítko, 52

V

validátor, 53
VALIGN, 22
velikost odsazení, 69
velikost okna prohlížeče, 50
velikost písma, 31, 43
velikost stránky, 49
velikost tabulek, 19
velikost TEXTAREA, 29
velikost textového pole, 27, 30
velikost znaků, 52
velikosti písmen, 48
velká písmena, 58
vícenásobný výběr, 61
víceslovné názvy písem, 46
VLINK, 12
vnořená tabulka, 23
vnořování tabulek, 55
výčty, 68
vyhledávací servery, 48
vytisknutá stránka, 70
vytváření ikon, 9

W

WIDTH, 9, 19, 20, 50
Windows Media Player, 62
Windows-1250, 24
www.cestina.cz, 24
www.interval.cz, 2

X

XHTML, 51, 53
XML, 51

Z

zalomení textu, 29
zarovnání, 20
zdrojový kód stránky, 50
zkřížení tagů, 51
změna barvy odkazů, 13
změnit barvu, 32
zmenšení písma, 43
zobrazování českých znaků, 24
zvětšení písma, 43
zvuk, 61

Ediční řada PCWorld Edition

S novou řadou publikací PCWorld Edition se v roce 2001 pravidelně setkáváte každý měsíc s vydáním časopisu PCWorld. Každou publikaci můžete získat buď v **tištěné podobě** (formát 16,5 x 23 cm o minimálně 80 stranách), nebo jako **počítačový soubor** ve formátu PDF, který můžete snadno přečíst na každém PC i MAC. Chcete-li koupit některou z publikací, zaplaťte příslušnou částku na náš účet **243612028/2400** a telefonicky nebo e-mailem nám nahlašte adresu pro doručení. Podrobné pokyny pro uplatnění předplatitelských slev a zjednodušení způsobu objednávání s identifikačním číslem najdete na CD-ROM v PCWorldu nebo na www.win.cz, případně Vám poradí naši pracovníci na tel. č.: **05 - 41 51 55 00**.

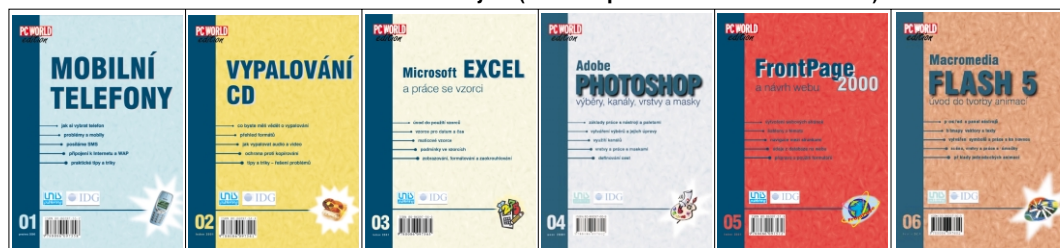
Abyste věděli, co si kupujete, najdete na CD-ROM v PC Worldu **ukázky knih** v elektronické podobě. Tyto soubory si můžete volně stáhnout i z našeho webu – www.win.cz. K jejich prohlížení potřebujete pouze volně dostupný program Acrobat Reader. Na CD-ROM v časopise PCWorld nebo volně stažitelné z www.win.cz jsou i plné elektronické verze publikací; k jejich otevření však potřebujete heslo, které Vám zašleme po provedení platby na výše uvedený účet. Vlastní soubory poštou nezasíláme.

Při zasílání publikací poštou NEPLATÍTE žádné poštovné ani jiné poplatky navíc!

odběratel:	tištěná kniha	heslo pro elektronické publikace	obě verze
předplatitel PCWorld, Computer World, GameStar	65 Kč	40 Kč	90 Kč
není předplatitel	99 Kč	60 Kč	120 Kč

Využijte možnost zvýhodněného odběru při jednorázové platbě většího počtu publikací nebo předplatného ediční řady PCWorld edition – bližší informace na tel. **05 - 41 51 55 00** nebo na www.win.cz.

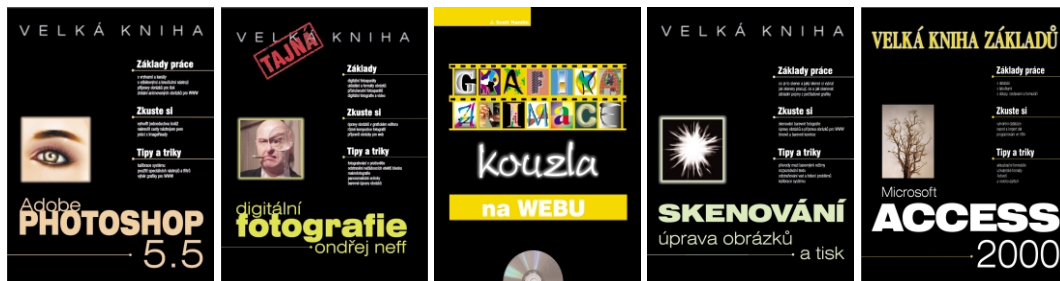
V ediční řadě PCWorld Edition vyšlo (v období prosinec 2000 až duben 2001):



Připravují se témata: Elektronické dokumenty, Visual Basic, Excel a databáze, CorelDRAW, FLASH 5 - ActionScript, Photoshop 6.0, Tisk a osvit, MS Word, XML... Skladbu publikací můžete ovlivnit i sami – na www.win.cz najdete anketu, kde můžete dát hlas některému z témat, nebo sami téma navrhnout.

Obdobným způsobem jako publikace řady PCWorld Edition lze objednat i ostatní knihy vydavatelství UNIS Publishing, např. (tučně je uvedena cena pro předplatitele):

- **Referenční příručka DHTML** – dynamické HTML (168 str., 150/110 Kč)
- **Velká kniha Adobe Photoshop 5.5** (420 str., CD-ROM, 350/260 Kč)
- **Tajná kniha digitální fotografie** (192 str., z toho 96 barevných, 220/165 Kč)
- **Grafika a animace – kouzla na webu** (360 str., CD-ROM, 390/290 Kč)
- **Skenování, úprava obrázků a tisk** (288 str., 290/220 Kč)
- **Velká kniha základů Access 2000** (336 str., 340/250 Kč)



Kompletní nabídku najdete na www.win.cz, nebo si ji vyžádejte na adrese:
UNIS Publishing, s.r.o., Jundrovská 33, 624 00 Brno, tel.: 05 - 41 51 55 00, fax: 05 - 41 51 55 02