

Pavol Mikle

X_DHTML

HTML, XHTML, DHTML

úplná přesná referenční příručka

(X)HTML jazyk

(X)HTML dokument

XML a (X)HTML pravidla

Multimédia v (X)HTML dokumentu

Rozšiřující referenční tabulky

JavaScript a VBscript

Znakové sady

Tipy a triky

Rozšířené dynamické HTML – referenční příručka

část HTML jazyk

autor a grafické zpracování: RNDr. Pavol Mikle

Copyright © ZONER software s.r.o.

Vydání první v roce 2004. Všechna práva vyhrazena.

Odpovědný redaktor: Ing. Pavel Kristián

Copyright © Foto na obálce: Jiří Heller, HELLER.CZ s.r.o., www.heller.cz

Copyright © Cover: Ing. Pavel Kristián

Zoner Press

ZONER software s.r.o.

Koželužská 7, 602 00 Brno

<http://www.zonerpress.cz>

Informace, které jsou v této knize zveřejněny mohou být chráněny jako patent. Jména produktů byla uvedena bez záruky jejich volného použití.

Při tvorbě textů a vyobrazení bylo sice postupováno s maximální péčí, ale přesto nelze zcela vyloučit možnost výskytu chyb.

Vydavatelé a autoři nepřebírají právní odpovědnost ani žádnou jinou záruku za použití chybných údajů a z toho vyplývajících důsledků.

Všechna práva vyhrazena. Žádná část této publikace nesmí být reprodukována ani distribuována žádným způsobem ani prostředkem, ani reprodukována v databázi či na jiném záznamovém prostředku či v jiném systému bez výslovného svolení vydavatele s výjimkou zveřejnění krátkých částí textu pro potřeby recenze.

Veškeré dotazy týkající se distribuce směřujte na:

Zoner Press

ZONER software s.r.o.

Koželužská 7, 602 00 Brno

tel.: 532 190 883, fax: 543 257 245

e-mail: knihy@zonerpress.cz

ISBN 80-86815-01-3

OBSAH

ÚVOD	9
VÝVOJ HTML	13
ZROD A VÝVOJ INTERNETU	14
INTERNET	14
WWW	14
(X)HTML JAZYK	16
SPOLEČNÉ ATRIBUTY – OBECNÉ IDENTIFIKACE	18
Společné atributy pro identifikaci prvků	19
Společné atributy pro prezentaci prvků	19
Společné atributy pro internacionalizaci	20
bílé znaky	21
Nejfrekventovanější značky s běžnými atributy	22
REJSTŘÍK ZNAČEK	23
Rozsah platnosti (X)HTML značek v jednotlivých DTD	24
Modularizace (X)HTML značek	25
Základní moduly (Core Modules)	25
Moduly textových nástaveb (Text Extension Modules)	25
Formulářové moduly (Forms Modules)	25
Tabulkové moduly (Table Modules)	26
Specifické moduly	26
(X)HTML značky bez obav	27
Postačující je zvládnout 8 značek	27
Včetně tabulek je třeba zvládnout 11 značek	27
Zbytek značek do celkového počtu	27

Snadné řešení problému formátování písma	27
Základní kostra HTML dokumentu	28
Základní kostra XHTML dokumentu	29
Značky záhlaví dokumentu	30
Značky pro vkládání skriptů	40
Značka pro vymezení těla dokumentu a komentáře v těle dokumentu	41
Značka pro hypertextový odkaz	42
Značky pro základní formátování textu	45
Značky pro pokročilé formátování textu	47
Značky pro specifické formátování textu	48
Značky pro blokově strukturované texty	51
Značky pro výčty a seznamy	52
Značky pro zápis tabulky	56
Značky pro grafiku, multimédia a obrázky	65
Značky pro vnořené programy a objekty	68
Značky pro zápis formuláře a ovládacích prvků	73
Značky pro vytvoření rámců	85
NEPERSPEKTIVNÍ ZNAČKY	90
Zastaralé prvky	90
Prvky, které pozbyly význam	90
Nedoporučené proprietární prvky	90
Nepodporované prvky	90
Proprietární značky	91
(X)HTML DOKUMENT	97
LOGICKÉ HTML STRUKTURY	97
základní HTML struktury	98
specifické HTML struktury	99
SPECIFIKACE HTML STRUKTUR	101
PROSTÝ TEXT	101
NADPISY	101
ODSTAVEC	102

OBRÁZEK	104
TABULKA	105
ODRÁŽKY A ČÍSLOVÁNÍ	109
SEZNAM POJMŮ	109
PŘEDFORMÁTOVANÝ TEXT	110
CITACE, CITÁT	110
ADRESA	111
PRIMITIVNÍ FORMULÁŘ	111
FORMULÁŘ	112
RAMCE	112
VSazený ramec	114
VNOŘENÝ PROGRAM	114
VNOŘENÝ OBJEKT	115
TAPETY, VODOZNAKY	115
VODOROVNÁ ČARA	116
ANIMOVANÁ PROJEKCE, BLIKAJÍCÍ TEXT	116
HYPERLINK	117
HTML SOUBOR	118
STAVBA HTML DOKUMENTU	119
XML – ROZŠÍŘITELNÝ ZNAČKOVACÍ JAZYK	122
VŠEOBECNÁ XML PRAVIDLA	122
XML pravidla	122
Struktura XML dokumentu	123
Správně formované (well-formed) a platné (valid) dokumenty	124
VŠEOBECNÁ HTML PRAVIDLA	129
HTML gramatika	129
HTML pravidla značek	129
VŠEOBECNÁ XHTML PRAVIDLA	130
XHTML gramatika	130
XHTML pravidla značek	130
XHTML dokumenty	131
Přechod z HTML na XHTML	132

DTD – DEFINICE TYPY DOKUMENTU	134
PROLOG DOKUMENTU, DEKLARACE DTD A DEFINICE SYNTAXE DTD	134
Prolog dokumentu	136
DTD – Deklarace definice typu dokumentu	137
Odkaz na DTD (<DOCTYPE)	138
DTD JAZYK	139
SKRIPTOVÁNÍ	144
JAVASCRIPT: SYNTAXE, METODY, A FUNKCE	147
VBSCRIPT: SYNTAXE, METODY A FUNKCE	157
WWW	166
WWW DOKUMENT	166
USER AGENT	166
Pravidla pro chování interpretů	166
Formální pravidla pro rozbor textu dokumentu	167
Pravidla pro nakládání s neznámými konstrukcemi	167
SOUBOROVÉ OBJEKTY (APPLET, ACTIVEX, ..)	169
Souborový objekt	169
Applet	169
ActiveX	169
MULTIMÉDIA V HTML DOKUMENTU	171
OBRÁZKY	171
ZVUKY / MELODIE	171
VIDEOKLIPY	172
ANIMOVANÉ PROJEKCE	172
DOPORUČENÍ, TYPY A TRIKY	173
Doporučení č. 1 Jak se bezbolestně zbavit prvku FONT (. . .)	173
Doporučení č. 2 Zlobivý formulář (<form > . . . </form>)	173
Doporučení č. 3 Ikona pokušitelka (favicon.ico)	174
Trik č. 1 Vkládání vlastních značek (<MUJPRVEK> . . . </MUJPRVEK>)	175

Trik č. 2	Odfiltrování značky (<!PRVEK>)	175
Trik č. 3	Odfiltrování atributů (<PRVEK _atribut="hodnota" >)	175
Trik č. 4	'drzé' komentáře uvnitř značek (<PRVEK ... "komentář" >)	175
Trik č. 5	Cizí atributy uvnitř značek (<PRVEK ... catribut ... >)	176
Tip č. 1	Základní šablony složitých HTML struktur	176
Tip č. 2	Tabulka v tabulce	177
Tip č. 3	Speciální strukturování www stránky	177
Tip č. 4	Optimalizace HTML souboru	178
Tip na závěr:	Lze HTML soubor zašifrovat?	178

CO V HTML NELZE A JAK TO ŘEŠIT 179

a)	vícelsoupcová sazba	179
b)	víceúrovňové strukturované číslování	179
c)	oboustranné obtékání obrázků a objektů	180
d)	přesné rozložení objektů na stránce v základním HTML	181
e)	odsazení prvního řádku odstavce	181

HTTP, TCP/IP, URI 183

HTTP PROTOKOL	183
Obecné hlavičky	184
Hlavičky požadavku	184
Hlavičky odpovědi	185
Hlavičky těla	185

STAVOVÁ HLÁŠENÍ HTTP PROTOKOLU	186
1xx – informační	187
2xx – úspěch	187
3xx – přesměrování požadavku	187
4xx – klientova chyba (chybný požadavek)	188
5xx – chyba serveru	189

SCHEMA TCP/IP	190
URI, URL, URN – ADRESACE ODKAZOVANÉHO DOKUMENTU V SÍTI WWW	191
Úplný stručný přehled zápisu syntaxe URI	192
Porty	193
TLD – DOMÉNY NEJVYŠŠÍ ÚROVNĚ	195
Národní vrcholové domény	195

Všeobecné (generické) vrcholové domény	196
Oblastní vrcholové domény USA	196
KÓDY PRO REPREZENTACI NÁZVŮ JAZYKŮ A JEJICH KÓDY ZNAKOVÝCH	
SAD	198
ZNAKOVÉ SADY	199
ZNAKOVÁ SADA, KÓDOVÁNÍ ZNAKŮ	199
Znakové sady (Character Sets) pro Evropu, USA a Rusko	199
Kódování znaků (Character Encoding) pro všechny jazyky	200
ZNAKOVÉ ENTITY	201
STANDARDY	204
MIME TYPY	204
RFC – POŽADAVKY NA KOMENTÁŘE	206
INTERNETOVÉ STANDARDY	206
REJSTŘÍKY	207
SEZNAM VYOBRAZENÍ A TABULEK	207
REJSTŘÍK DOKUMENTOVÝCH PRVKŮ	208

ÚVOD

Referenční příručka jazyka HTML pro tvorbu WWW stránek podle pravidel XML 1.0 a specifikací HTML 4.01, XHTML 1.0, DHTML

Tato publikace je kompletní referenční příručkou aktuálního stavu jazyka HTML známého jako **HTML 4**, který se po letech vývoje dostal do definitivního tvaru. Jedná se o standard **HTML 4.0** v revizi **4.01** s navazující specifikací **XHTML** a specifikacemi skrývajícími se pod **DHTML** (CSS, DOM, events).

HTML (HyperText Markup Language) slouží k tvorbě WWW dokumentů. HTML je vynikající nástroj pro popis stránek. Je to typ jednoduchého značkového jazyka použitelného na různých platformách. HTML je strukturovaný jazyk, a proto se mimořádně hodí i pro automatizovanou tvorbu WWW dokumentů.

Tato referenční příručka obsahuje aktuální přehled jazyka **HTML**. Klade si za cíl umožnit rychlou orientaci ve zdrojových textech WWW dokumentů – číst je, modifikovat a vytvářet dokumenty vlastní podle nejnovějších trendů a vývoje prvků jazyka. HTML se stává základním standardem pro vytváření dokumentů, jejichž vnitřní struktura je otevřená a přístupná. Dříve nebo později se s touto strukturou uživatelé chtěli se setkat. Příručka umožní rozpoznat a pochopit jednotlivé struktury a na základě uvedených syntaxe HTML dokumenty modifikovat přímo ve zdrojové struktuře bez nutnosti znát syntaxi jazyka nazpaměť.

HTML je vyvíjený se jazyk. O standardizaci HTML jazyka se stará tzv. W3C konzorcium (World Wide Web Consortium), které jako jediný vrcholový orgán specifikuje a oficiálně vydává nové definice (verze) jazyka HTML. Poslední oficiální specifikaci jazyka je **HTML 4.0** (z prosince 1997) v revizi **HTML 4.01** (s posledním doporučením z prosince 1999) a její přísně strukturovaná forma pod specifikací **XHTML 1.0** (s posledním doporučením z ledna 2000).

XML

(eXtensible) **H**yper**T**ext **M**arkup **L**anguage

HTML 4.01 (1999)
XHTML 1.0 (2000)

Standardy:
W3C HTML <http://www.w3c.org/TR/html401>
W3C XHTML <http://www.w3.org/TR/xhtml1/>

XHTML 1.0 je reformulace HTML 4 jako XML aplikace. Oba standardy se liší pouze v pravidlech zápisu, nikoliv v obsahu. Zatímco pravidla zápisu HTML jsou benevolentní (viz Všeobecná HTML pravidla str. 121) pravidla zápisu XHTML jsou rigorózní (viz Všeobecná XHTML pravidla str.130). HTML 4 je poslední verzi HTML. XHTML 1.0 = HTML 4 + XML 1.0

(X)HTML JAZYK

Vlastní zápis (X)HTML není nic jiného než písmenná hříčka navozující možnost volby mezi variantami "špatný/starý" HTML nebo "dobrý/nový" XHTML. Pro jazyk XHTML jsou k dispozici dvě finální specifikace – XHTML 1.0 a XHTML 1.1 a návrh specifikace XHTML 2.0

XHTML	<ul style="list-style-type: none"> ▪ znamená eXtensible HyperText Markup Language – rozšiřitelný hypertextový značkovací jazyk ▪ je HTML definovaný jako XML aplikace (je reformulací HTML na bázi XML) ▪ je kombinace HTML a XML (eXtensible Markup Language) ▪ má za cíl nahradit HTML
XHTML 1.0	<ul style="list-style-type: none"> ▪ je jazykově totožný s HTML 4.01 ▪ je přísnější a čistější verzí HTML 4.01 ▪ sestává ze všech prvků v HTML 4.01 spojených s pravidly XML
XHTML 1.1	<ul style="list-style-type: none"> ▪ je přísnější verzí XHTML 1.0 ▪ je postaven na standardu XHTML 1.0 Strict, z něhož odstraňuje všechny konstrukce, které byly označeny jako deprecated (překonané) ▪ neobsahuje již žádné prvky ani atributy, které opustil již XHTML 1.0 Strict (takže například zcela chybí definice rámů a všechny atributy zaměřené na definici vzhledu prvku (align, width, ...) a atribut style) ▪ vypouští atribut name a nahrazuje jej definitivně atributem id
XHTML 2.0	<p>stav prvního veřejného návrhu</p> <ul style="list-style-type: none"> ▪ zpětná kompatibilita <i>není zpětně kompatibilní s předchozími verzemi</i> ▪ vše je odkazem <i>atribut href je zařazen do kolekce běžných atributů (každý prvek může být zároveň odkazem)</i> ▪ obrázek jako objekt <i>zneplatnění prvku <img (obrázky se vkládají jako další multimedialní objekty pomocí značky <object)</i> ▪ prezentační prvky <i>zneplatněny prezentační prvky b, big, i, small, tt</i> ▪ prvek line <i>prvek <br je označen jako deprecated (překonaný/zavřzený) a nahrazuje se konstrukcí <line></line> představující "pododstavec"</i> ▪ členění nadpisů do úrovní <i>nové značky pro nadpisy <h></h> a <section></section></i> ▪ XHTML menu <i>nové značky pro menu <nl></nl> a <name></name></i>

Následující specifikace a syntaxe značek je zcela přesná, to znamená, že je oprostěna od všech spekulací zda daná konstrukce je či zůstane v platnosti. To je možné především díky tomu, že definice jazyka HTML 4.01 a XHTML 1.0 jsou po letech vývoje ustálené do konečné podoby. Všechny značky jsou zcela jasně vymezeny i co do rozsahu platnosti podle typu definice či indikovány v případech, kdy je norma nedoporučuje. Pro autory je tato situace skvělá v tom, že se mohou opřít o definitivu a věnovat se vlastnímu obsahu dokumentu než testování průchodnosti té či oné konstrukce jazyka.

Je třeba ale vzít na vědomí, že stabilizace normy je jedna věc a její důsledná implementace v UA programech (prohlížečích) druhá. Hlavní protagonisté nemají problém s vlastními prvky. S určitými odchylkami vůči normě je nutno počítat zejména v přesnosti implementace atributů. Je důležité si u jednotlivých atributů povšimnout, zda je norma nezařadila mezi nedoporučené a zda funkčnost nenahrazuje atributem jiným. Zejména tyto případy nemusí být v ideálním stavu. Neustále se také lze potkávat s různými výstřelky UA programů, z nichž zajímavé (a u autorů oblíbené) jsou uvedeny jako proprietární. Nedoporučené atributy HTML 4 podporuje, XHTML nikoliv (v XHTML tyto atributy vystupují jako nerozpoznané).

Poznámky:

1. HTML jazyk ve verzi 4.01 je "ukončená větev" – kód je definitivně uzavřen a již provždy zůstane jak je, pod stručným označením **HTML 4**;
2. Ize s absolutní jistotou spoléhat, že "starý" HTML kód z WWW nikdy nevyvmizí a že jej tedy budou všechny prohlížeče trvale nadále podporovat;
3. XHTML 1.0 není velmi odlišný od HTML 4.01, takže velmi dobrý začátek pro XHTML je úprava kódu dokumentů na verzi HTML 4.01;
4. na XHTML se lze připravit a přejít nejsnadněji tak, že se jednoduše začne psát přísný HTML ;
5. na WWW neexistuje žádná povinnost začít tvořit XHTML dokumenty – bez omezení lze vytvářet nadále HTML dokumenty, a to včetně používání starších verzí jazyka HTML, takže kdo se nehodlá vůbec zabývat způsobem psaní značek, ať je píše tak, jak se mu to líbí a vyhovuje i nadále;
6. prohlížeče WWW dokumentů NIKDY neohlásí chybu, ale při výskytu prvního prohřešku proti deklarované skutečnosti se jednoduše přepnou na "nižší" jazykovou verzi HTML (na WWW je příliš mnoho dokumentů na to, aby si prohlížeč troufl např. kvůli jedné chybějící koncové značce v jinak absolutně bezchybném XHTML dokumentu jej nedokončit, nezobrazit nebo snad zobrazit návštěvníkovi nějaké nesrozumitelné chybové hlášení – prohlížeč se prostě musí vůči uživateli tvářit jako fault tolerantní všemě!);
7. jediná možnost jak se přesvědčit o tom, že WWW dokument je napsán jako správný XHTML dokument, je předložit jej k ověření nějakému validátoru
8. autoři, kteří se vzdávají možnosti začít psát čisté strukturované dokumenty pomocí XHTML, se perspektivně vzdávají možnosti širšího zpracování svých WWW dokumentů v budoucnosti (např. prostřednictvím vyhledávačů či jiných zařízení než je PC)
9. XHTML 1.1 je definitivní odklon od všech pozůstalostí minulosti – to znamená zpětnou nekompatibilitu pro starší verze prohlížečů; jeho nasazení je vzhledem k tomu v současnosti méně aktuální;
10. XHTML 2.0 je zásadním odklonem od původního "starého" HTML a teprve finální návrh ukáže, o jak zásadní odklon se bude jednat.

SPOLEČNÉ ATRIBUTY – OBECNÉ IDENTIFIKACE

Skupina HTML atributů **id**, **class**, **style**, **title**, **lang**, **dir**, **intrinsic events** se týká prakticky všech HTML značek (výjimky jsou v některých značkách, v nichž by atribut neměl jako např. atributy lang a dir ve značce zalomení řádku `
`).

Atributy **id** (*název / identifikátor prvku*), **class** (*stylová třída*) jsou atributy *identifikace prvku*.

Atributy **style** (*stylový předpis – vnitřní styl*), **title** (*poradní / návěštní informace*) jsou atributy *prezentace prvku*.

Atributy **lang** (*jazyk*), **dir** (*orientace textu*) jsou atributy *internacionalizace prvku*.

Atributy **intrinsic events** (*vestavěné události*) jsou atributy **onclick**, **ondbleclick**, **onmousedown**, **onmouseover**, **onmousemove**, **onmouseup**, **onkeypress**, **onkeydown**, **onkeyup** (*reference prvku ze skriptu*).

každý HTML prvek může obsahovat atributy **id**, **class**, **style**, **title**, **lang**, **dir**, **intrinsic events**:

<p><prvek id="identifikátor" class="stylové_třidy" style="stylový_předpis" title="poradní_informace" lang="kód_jazyka" dir="orientace_textu" událost="skript" ...ostatní_atributy... ></p>	<p>UPOZORNĚNÍ: z důvodu přehlednosti zápisu syntaxe HTML jazyka jsou společné atributy v referenční tabulce uvedeny zastupující zkratkou <i>%attrs</i> a vestavěné události jsou uvedeny výčtem umístěným za prvkem takto:</p> <p><prvek %attrs ...ostatní_atributy... > události: event, event, ...</p>
<p>kde</p> <p>identifikátor určuje jméno (pojmenování) prvku jako jednoznačný identifikátor</p> <p>stylové_třidy určení jedné nebo mnohých stylových tříd pro obsah prvku</p> <p>stylový_předpis stylový předpis pro obsah prvku</p> <p>kód_jazyka určuje národní jazyk pro obsah prvku</p> <p>orientace_textu určuje orientaci textu uvnitř prvku (zleva-doprava/zprava-doleva)</p> <p>poradní_informace určuje návěštní/poradní informaci o prvku (tečezec)</p> <p>událost jedna nebo řada ze standardních vestavěných událostí: onclick, ondbleclick, onmousedown, onmouseover, onmousemove, onmouseup, onkeypress, onkeydown, onkeyup</p>	<p>kde</p> <p>%attrs označuje společné atributy kromě vestavěných událostí, které jsou vždy u každé značky uvedeny jako úplný seznam</p>

%attrs == id|class|style|title|lang|xml:lang|dir

entita *%attrs* zastupuje společné atributy jak je uvedeno, přičemž zápis `id|class|style|title|lang|xml:lang|dir` neznamená jednu z variant, ale jeden, všechny nebo kombinaci atributů

ZNAČKY ZÁHLAVÍ DOKUMENTU

Tajuplné záhlaví dokumentu jehož obsah při prezentaci dokumentu prakticky nikde není viditelný, má daleko větší význam, než by se dalo předpokládat. Vzhledem k tomu považují za účelné předem avizovat, že záhlaví dokumentu zásadním způsobem ovlivňuje způsob zacházení s dokumentem na straně prohlížeče a způsob "hodnocení" dokumentu vyhledávacími roboty.

<p><i>ohraničení dokumentu</i> <html>_{FTS}</p>	<p><html <i>%attrs</i> version="URL" > ... </html> <i>události</i>: žádné párová značka, která slouží k ohraničení každého jednotlivého dokumentu; značka je v HTML normě nepovinná a některými staršími prohlížeči je ignorována <i>%attrs</i> lang xml : lang dir (viz str. 18) Ize použít lang, dir (nelze použít class, id, style, title) *version= specifikuje URI, který specifikuje lokaci DTD pro verzi HTML použitého v dokumentu; protože se stejná informace musí objevit v identifikátoru !DOCTYPE, stal se tento atribut nepotřebným a je zařazen mezi *nedoporučené</p>
<p><i>záhlaví dokumentu</i> <head>_{FTS}</p>	<p><head <i>%attrs</i> profile="URL" > ... <i>prvky_záhlaví</i>... </head> <i>události</i>: žádné párová značka, která vymezuje záhlaví dokumentu; obsahuje uspořádanou kolekci informací o dokumentu, která slouží výlučně pro prohlížečí program; v záhlaví by neměl být obsažen žádný zobrazovaný text, úvodní a koncová značka v HTML normě nemusí být uvedeny; do záhlaví dokumentu náleží prvky: <title>, <base>, <isindex>, <meta>, <link>, <style>, <script> <i>%attrs</i> lang xml : lang dir (viz str. 18) Ize použít lang, dir (nelze použít class, id, style, title) profile= specifikuje URI lokaci jednoho nebo více profilů metadat oddělených bílým znakem; specifikace nedefinuje formáty profilů; prohlížeč může vyvinout nějakou aktivitu na základě známých konvencí pro konkrétní profil; např. vyhledávací stroje mohou poskytnout rozhraní pro vyhledávání v katalogu dokumentů, kde všechny tyto dokumenty používají tentýž tentýž profil pro reprezentaci katalogových položek</p>
<p><i>titulek dokumentu</i> <title>_{FTS}</p>	<p><title <i>%attrs</i> > ... <i>text_titulku</i>... </title> <i>události</i>: žádné párová značka, která definuje titulek dokumentu; každý (X)HTML dokument musí obsahovat právě jeden prvek <title>; titulek má za úkol dokument identifikovat; UA musí titulek zpřístupnit; mechanismus zpřístupnění titulku je věci UA (audio agent přehláskuje, prohlížeč zobrazí na horní liště okna, použije se v seznamu záložek a zpracovává jej většina vyhledávacích robotů); počet znaků titulku by neměl překročit 64; v titulku se smí vyskytnout znakové entity, v obsahu titulku se ale nesmí vyskytnout žádná značka <i>%attrs</i> lang xml : lang dir (viz str. 18) Ize použít lang, dir (nelze použít class, id, style, title)</p>

metainformace

`<meta>`^{FTS}

`<meta <attrs http-equiv="název_HTTP_hlavičky" content="hodnota" scheme="šablona_hodnoty" />`
`<meta <attrs name="identifikátor" content="hodnota" scheme="šablona_hodnoty" />`

události: žádné

metaprvek – nepárová značka, která umožňuje uvést *dodatečně specifické informace o dokumentu* (metainformace); záhlaví dokumentu může obsahovat libovolný počet metaprvků (značek `<meta>`); každý jednotlivý metaprvek specifikuje jeden pár 'identifikátor/hodnota'; pomocí těchto dvou atributů lze např. modifikovat nebo zavést další prvky HTTP hlavičky dokumentu (a tím informovat prohlížečící program např. o stavu dokumentu), nebo zadat jiné význačné hodnoty týkající se obsahu a vlastností dokumentu (např. pro vyhledávací služby na internetu); metaprvky mají dvě hlavní funkce:

1. poskytnout serveru prostředek k předání informací prostřednictvím http hlavičky dokumentu (server *může* použít jméno indikované atributem `http-equiv` jako http hlavičkový údaj s hodnotou specifikovanou atributem `content`)

Např. na základě specifikace následujících dodatečných informací v dokumentu

```
<meta http-equiv="content-type" content="text/html" ; charset=ISO-8859-5" />
<meta http-equiv="expires" content="Sat 31 Jan 2004 21:43:04 GMT" />
```

může server v případě vyžádání tohoto dokumentu včetně následující http hlavičku odpovědi

```
content-type: text/html ; charset=ISO-8859-5
expires: Sat 31 Jan 2004 21:43:04 GMT
```

2. dokumentovat obsah a vlastnosti dokumentu různými indikacemi (indikované atributem `name` jako vlastnosti s hodnotou vlastnosti specifikovanou atributem `content`)

Např. typicky pro definování informací určených k indexování dokumentu (jako autora, copyright, klíčová slova, popis, datum)

```
<meta name="author" content="Jméno příjmení" />
<meta name="copyright" content="&copy; 2004 ABC" />
<meta name="keywords" content="klíčová slova,..." />
<meta name="description" content="popis ..." />
<meta name="date" content="popis ..." />
```

Neexistuje žádný standardní seznam META dat (nejsou součástí žádné specifikace, veskrze se používá několik vzrých hodnot, které jsou rozpoznávány většinou (X)HTML interprety), takže autoři mohou definovat libovolná metadata jaká jim vyhovují; následně uvedené hodnoty atributů jsou pouze aktuálně vžitě hodnoty atributů.

`<attrs`

`lang|xml|dir` (viz str. 18)

lze použít `lang`, `dir` (nelze použít `class`, `id`, `style`, `title`)

`http-equiv=` je-li místo atributu `name` použit atribut `http-equiv` v pak jde o název HTTP hlavičky:

`content- language` kód národního jazyka dokumentu

`content- type` znakové kódování dokumentu

`refresh` automatické znovunačtení dokumentu

`expires` doba (datum a čas) ukončení platnosti stránky

`cache-control` ovládní cache prohlížeče

`pragma` předávání specifických hlaviček pro různá zařízení

`PICS-Label` hodnocení PICS (Platform for Internet Content Selection)

`content-script-type` implicitní skriptový jazyk

`content-style-type` implicitní stylový jazyk

`default-style` implicitní styl

name=	specifikuje identifikátor vlastnosti (jméno vlastnosti); následující identifikátory nejsou definované, nýbrž vztí: <ul style="list-style-type: none"> description popis dokumentu keywords klíčová slova dokumentu author autor dokumentu copyright copyright dokumentu (autorská práva) generator program, jehož prostřednictvím byl dokument vytvořen nebo generován robots ovládní činnosti webových robotů googlebot ovládní činnosti robota Google
content=	specifikuje hodnotu (hodnotu vlastnosti); v případě atributu http-equiv ovlivňuje obsah hlavičky HTTP protokolu, které se to týká (servery používají atribut http-equiv ke zhrnání informací pro hlavičku HTTP odpovědi); hodnota atributu může obsahovat text a entity (např. ©);
scheme=	specifikuje formát hodnoty vlastnosti, který má být použit k interpretaci hodnoty vlastnosti; pro prohlížeč je to užitečná, ale nekritická informace, jejíž deklarace může být pro prohlížeč nápomocná pro korektní interpretaci metadat (např. schema "Day-Month-Year" může napomoci pro interpretaci hodnoty "10-9-04" kde vůbec není zřejmé, co je den, co je měsíc a co je rok)

Vztí specifikace metadat

jazyk dokumentu

```
<meta http-equiv="content-language" content="kód_jazyka" />
```

kód_jazyka

kód národního jazyka (cs český sk slovenský jazyk); uvedení kódu jazyka je podstatné pro správné zařazení dokumentu do databáze vyhledávačů; některé vyhledávače přikládají stránce s uvedeným kódem jazyka vyšší prioritu

typ a znaková sada dokumentu

```
<meta http-equiv="content-type" content="text/html; charset=kód_znakové_sady" />
```

kód_znakové_sady

UTF-8 použitelné prakticky pro jakýkoli jazyk
 ISO-8859-2, windows-1250 pro západoevropské jazyky (včetně češtiny a slovenštiny)
 ISO-8859-1, windows-1252 pro západoevropské jazyky (včetně angličtiny)

automatické znovumáčnění / přeměrování stránky

```
<meta http-equiv="refresh" content="sekundy; URI" />
```

sekundy

počet sekund, po jejichž uplynutí bude stránka přeměrována/znovumáčněna

URI URI adresa na/z níž bude stránka přeměrována/znovumáčněna

používá se pro techniku tzv. "uvítací" stránky nebo např. pro přeměrování na jinou adresu v případě přestěhování webového sídla – prohlížeč po daném čase automaticky přeměruje uživatele na novou adresu; UPOZORNĚNÍ: některé vyhledávače stránky, které reftrešují po několika málo vteřinách, penalizují; v případě přestěhování webového sídla je lepším řešením použití standardního postupu pomocí HTTP stavového kódu 30x, který jsou schopna rozluštit všechna možná zařízení pracující s protokolem http

stanovení doby ukončení platnosti stránky

```
<meta http-equiv="expires" content="vypršení_platnosti" />
```

metadata konvence

http-equiv

vypršení platnosti

hodnota ve formátu **Day, dd Month yyyy hh:mm:ss GMT** kde Day a Month jsou první 3 písmena anglického názvu dne a měsíce, dd je datum dne, yyyy je rok, hh:mm:ss je čas, přičemž se jedná o Greenwichský čas (GMT – Greenwich Mean Time); udává, do kdy je daná stránka platná (tedy do kdy ji může mít prohlížeč uloženu v cache) – po tomto datu musí prohlížeč znovu načíst stránku ze serveru (zažádat o její novou verzi)

ovládání cache prohlížeče

```
<meta http-equiv="cache-control" content="pravidlo[, must-revalidate]" />
```

pravidlo

no-cache

public

max-age=vteřiny

stránka se do cache nesmí ukládat

stránka se do cache smí ukládat (u běžných stránek automatické)

specifikuje dobu od požadavku, po jakou je stránka aktuální – po uplynutí této doby musí prohlížeč stránku znovu načíst (zažádat o její novou verzi);

použití hodnoty 0 v domnění, že zajistí neukládání stránky do cache, je nestandardní a proto je nutné se jí vyvarovat

must-revalidate je-li uveden, říká prohlížeči, že pravidla ohledně cache je povinen akceptovat (jinak má jistou volnost)

předávání specifických hlaviček

```
<meta http-equiv="pragma" content="no-cache" />
```

no-cache

stránka se do cache nesmí ukládat (bez záruky správné funkcionality – metaprvek je primárně určen k předávání specifických hlaviček pro různá zařízení)

PICS hodnocení obsahu stránky

```
<meta http-equiv="PICS-Label" content='PICS_hodnocení' />
```

PICS_hodnocení

řetězec vygenerovaný zhodnocovací službou; řetězec může obsahovat znak uvozovka a proto je třeba jej uzavírat do apostrofu; hodnocení PICS je důležité u stránek, kde lze předpokládat nezletilé návštěvníky, neboť ti mohou mít přístup na nehodnocené weby zakázaný nebo omezený; pro získání PICS hodnocení je nezbytné nechat stránku zhodnotit stanovenou službou

PICS (Platform for Internet Content Selection) – systém hodnocení obsahu www stránek, původně určený k tomu, aby se nezletilí nedostali k "závadnému obsahu" internetu; v současnosti obecný systém pro charakteristiku jakýchkoli dat; specifikace PICS a další informace o systému se nachází na <http://www.w3.org/PICS>

implicitní skriptový jazyk

```
<meta http-equiv="content-script-type" content="MIME_type" />
```

MIME_type

MIME typ skriptového jazyka, ve kterém jsou udávány hodnoty atributů ovladačů údajostí
text/javascript pro JavaScript
text/vbscript pro Visual Basic Script

implicitní stylový jazyk

```
<meta http-equiv="content-style-type" content="MIME_type" />
```

XHTML DOKUMENTY

Každý XHTML dokument musí obsahovat tyto čtyři základní sekce, a to v uvedeném pořadí:

- **XML deklarace** – sekce `<?xml >` určuje verzi XML pravidel, způsob kódování obsahu dokumentu a informací pro čtení DTD; procesní instrukce `?xml` je povinná, jen pokud dokument používá jiné kódování než UTF-8 nebo UTF-16; je doporučeno uvádět ji pokudžé; je-li uvedena, musí se nacházet na prvním řádku dokumentu; přestože uvedení procesní instrukce `?xml` není požadováno, její presence explicitně identifikuje dokument jako XML dokument a indikuje verzi XML pod níž byl vytvořen
- **deklarace typu dokumentu** – sekce `<!DOCTYPE html >` určuje, podle jakého DTD je dokument napsán – pro XHTML 1.0 si lze vybrat jeden ze tří typů deklarací Strict, Transitional, Frameset – je povinná, jen pokud má být provedena validita dokumentu, je doporučeno uvádět ji pokudžé; musí se nacházet bezprostředně za XML deklaraci nebo na prvním řádku dokumentu, jestliže XML deklarace není uvedena
- **DTD pro XHTML 1.0** předepisuje root prvek `<html>` a tím říká, že celý obsah dokumentu se musí nacházet uvnitř kontejneru `<html>`; root prvek `<html>` může obsahovat nepovinné atributy internacionalizace (`lang`, `xml:lang` a `dir`), nepovinný atribut `id` (obsahuje jedinečné jméno prvku v rámci celého dokumentu) a povinný atribut `xmlns`, který deklaruje jmenový prostor XHTML a musí mít hodnotu `http://www.w3.org/1999/xhtml`, tj. tvar root prvku XHTML dokumentu je

```
<html xmlns="http://www.w3.org/1999/xhtml"
      id=".." xml:lang=".." lang=".." >
```

- **hlavička dokumentu** – sekce nacházející se v kontejneru `<head>` obsahuje doplňující informace o dokumentu (tzv. metadata, například titulek dokumentu, stručné shrnutí jeho obsahu, klíčová slova dokumentu, jeho vztah k jiným dokumentům a podobně), které prohlížeč nezobrazuje jako součást prezentovaného obsahu, může však (ale nemusí) je prezentovat jinými způsoby (například uvedením titulku dokumentu v titulku okna)
- **tělo dokumentu** – sekce nacházející se v kontejneru `<body>` nachází se v ní celý "viditelný" obsah dokumentu, tj. obsah, který je určen k přímé interpretaci.

Základní struktura XHTML dokumentu:

Mimimální XHTML dokument (který by vyhověl specifikaci XHTML 1.0 Strict) se všemi důležitými náležitostmi:

```
<?xml version="1.0" encoding="windows-1250"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
      xml:lang="cs" lang="cs" >
<head>
<meta http-equiv="content-type"
      content="text/html; charset=windows-1250" />
<meta http-equiv="content-language" content="cs" />
<title>... titulek ... </title>
</head>
<body>
... "viditelný" obsah dokumentu ...
</body>
</html>
```

Poznámky:

1. XHTML dokumenty musí být správně strukturované (well-formed)
2. všechny XHTML prvky musí být vloženy uvnitř kořenového prvku `<html>` všechny ostatní prvky mohou mít vnořené prvky (potomky)
3. vnořené prvky musí být v párech a správně vložené uvnitř jejich rodičovského prvku
toto je XHTML špatně:
`<i> ... </i></i>`
toto je XHTML správně:
`<i> ... </i></i>`
4. názvy značek musí být zapsány malými písmeny
toto je XHTML špatně:
`<BODY>
<P>Toto je odstavec</P>
</BODY>
<body>
<p>Toto je odstavec</p>
</body>`
toto je XHTML správně:
`<body>
<p>Toto je odstavec</p>
</body>`

5. všechny XHTML prvky musí být uzavřeny, tj. neprázdné prvky musí mít vždy uvedenou koncovou značku

toto je XHTML špatné:

```
<p>odstavec ...
<p>další odstavec ...
```

toto je XHTML správné:

```
<p> další odstavec ... </p>
<p> další odstavec ... </p>
```

6. prázdné prvky musí být rovněž uzavřeny – prázdné prvky musí mít uvedenou uzavírací značku nebo otevírací značka musí končit />

toto je XHTML špatné:

```
toto je řádkový zlom: <br>
toto je vodorovná dělicí čára: <hr>
```

toto je XHTML správné:

```
toto je také řádkový zlom: <br></br>
toto je řádkový zlom: <br />
toto je vodorovná dělicí čára: <hr />
```

UPOZORNĚNÍ: sekvence
</br> je sice podle pravidel korektní, ale větší prohlížeči si ní vůbec neporadí a odřádkuje dvakrát

7. všechny hodnoty atributů musí být v uvozovkách/apostrofech a musí být zapsány jako pár jméno="hodnota"

toto je XHTML špatné:

```
<option value=voľba selected>
```

toto je XHTML správné:

```
<option value="voľba" selected="selected">
```

8. **DŮLEŽITÁ poznámka kompatibility:** aby byl kód XHTML slučitelný s HTML prohlížeči, je nutné před ukončovací závorku ">" v nepárové značce vložit zvláštní mezeru jako např.
 nebo <hr />

PŘECHOD Z HTML NA XHTML

Přejít z HTML na XHTML lze nejsnadněji tak, že se jednoduše začne psát přísný HTML 4.01, tj.:

- všechny značky a jména atributů psát malými písmeny, jediná značka, která se v XHTML píše velkými písmeny je DTD značka <DOCTYPE
- všechny atributy psát ve tvaru páru jméno="hodnota" včetně osamocených HTML atributů (jako checked, selected) s hodnotou uzavřenou mezi " nebo ' (u osamocené atributu se jako hodnota uvede přímo jeho jméno např. selected="selected");
- vyhýbat se psaní bílých znaků uvnitř hodnot atributů;

4. současně s atributem name="jméno" používat id="jméno" kde jméno je jedinečné v rámci celého dokumentu (XHTML 1.1 atribut name již vůbec nezná a proto je perspektivnější zvyknout si na jediný atribut id – tento atribut má dvě skvělé výhody: zaprvé zpřístupňuje prvek pro skriptování a zadruhé přímočaře nabízí náhradu za značku font, protože id jméno je jeden ze základních způsobů CSS kvalifikace);

5. všechny prvky důsledně uzavírat uzavírací značkou (nevynechávat uzavírací značku);

6. všechny nepárové prvky (jako br a hr) ukončovat prostřednictvím ' />' (např.
);

7. vkládat prvky do sebe, jak se patří;

8. vyhýbat se formátovacím značkám:

- místo značky používat ;

- místo značky <i></i> používat ;

(některá zařízení nemusí být schopna psát tučným písmem nebo kurzívou, zápisy , dle standardů pouze upozorňují, že daná pasáž je důležitá – zařízení pak zdůraznění zadané části provede sama předepsaným způsobem podle svých možností);

9. k prezentaci obsahu používat styly:

místo značky používat atribut style="" v otevírací části

značky nebo raději externí css deklaraci (protože atribut style popírá

myšlenku oddělení obsahu dokumentu od vzhledu a např. definice

XHTML 1.1 tento atribut již vůbec nezná);

10. k vysvětlování používat styly:

místo značky <center></center> používat atribut align="center" v otevírací

části značky nebo raději externí css deklaraci (definice XHTML 1.1 atribut align již vůbec nezná);

11. omezit vrstvení tabulek jako formátovacího prvku dokumentu;

12. v tabulkách uvádět atribut summary="souhrnná charakteristika tabulky";

13. nevynechávat atribut alt="" alternativní text";

14. na začátku souboru uvádět kompletní prolog dokumentu;

15. ověřit, zda je dokument validní prostřednictvím XHTML validátoru;

16. přestože kódování obsahu dokumentu se uvádí v úvodní procesní instrukci <?xml> je třeba současně uvést stejný druh kódování dokumentu také v <meta> značce (pro starší verze prohlížečů, které procesní instrukce nezpracovávají);

17. pokud je u nějakého prvku určen jazyk jeho obsahu, je třeba kromě atributu xml:lang současně použít i lang, který je podporován prohlížeči.