

Otázka 25 - Y36WMM

Zadání

Přístupnost (accessibility) ve webové prezentaci. Typy webových dokumentů, jazyk HTML a XHTML, DTD, XML, CSS a jejich vztahy. (Y36WMM)

Slovníček pojmů

- **Webový dokument** - dokumentem se myslí konkrétní jedna stránka, respektive dokument zobrazovaný prohlížečem.
- **W3C** - konsorcium (neziskového charakteru) sdružující odborníky formující standardy pro webové prostředí. Založil jej a působí v něm dodnes Tim Berners Lee, člověk stojící za zrodem HTML.
- **Standard** - specifikace jazyka a jeho syntaxe.
- **Validní stránky** - odpovídající použité specifikaci, chcete-li verzi jazyka.
- **DOM** - Document Object Model, stromová reprezentace jednotlivých značek v paměti, DOM je potom renderován prohlížečem
- **Well-formed** - zejména u XML kde je to zásadní věc.. Jde o správně napsaný syntaktický kód. Správně uzavřené a nekřížící se značky.

Přístupnost (accessibility) ve webové prezentaci

Přístupnost webových stránek poukazuje na způsob vývoje webových stránek, tak aby byly přístupné všem jejich uživatelům. Vhodně navržené a zrealizované přístupné webové stránky nabídnou všem svým uživatelům totožné informace a funkce (pochopitelně nikoliv stejný vzhled či vizuální zážitek).

Pozor, neplést s jiným důležitým pojmem, souvisejícím s vývojem webových stránek, **použitelností (usability)**. Použitelnost se zabývá zejména tím jak navrhnout a zpracovat uživatelské rozhraní, navigaci a celkové uspořádání stránek a jejich vzájemnou logickou návaznost. Přístupnost se oproti tomu zabývá tím, jak zajistit všem uživatelům bezproblémové vstřebávání veškerých informací z webu.

Problematice přístupnosti webových prezentací se věnuje celá řada doporučení a zákonných úprav. Poměrně značný důraz na přístupnost je kladen zejména u projektů vznikajících z veřejných rozpočtů a u systémů státní správy a samosprávy. Doporučeními a pravidly přístupného webu se zabývají: *WCAG*, *W3C WAI* či český *Blind Friendly Web*.

Důležité zásady, které je třeba mít v souvislosti s přístupností na paměti

- **Čitelnost textu.** Dostatečně kontrastní rozlišení pozadí a textů, počítat s proměnlivou velikostí písma.
- **Veškerý funkční obsah** celého webu by měl být přístupný rovněž na zařízeních nepodporujících javascript (případně Flash).
- **Dokument** by měl být srozumitelně strukturovaný i bez použití kaskádových stylů
- **Netextové prvky** by měly být doprovozeny textovým popisem. Využitelné například u čteček pro nevidomé.
- **Na stránce** by se neměly vyskytovat příliš rychle blikající prvky (mohly by přivodit problémy epileptikům).
- **Formuláře** jsou opatřeny elementy zlepšujícími jejich přístupnost (`<legend>`, `<label>`, `accesskey=""`).
- **Problematika odkazů.** Měly by být jednotně formátovány. Pokud jsou podtrženy, neměly by být podtrhávány části textu, které nejsou odkazem. Stejně nazvané odkazy směřují na stejné místo. Neodkazujeme slovy „zde“, „klikněte“ a podobně.
- **Oddělujeme formátovací a vizuální část** od té prezentační. Formátování tedy zajišťujeme pomocí kaskádových stylů. Samotné (X)HTML nenese informace o formátování.
- **Srozumitelná a konzistentní navigace** a logické členění na nadpisy pomocí `<h1>` až `<h6>`.

Typy webových dokumentů

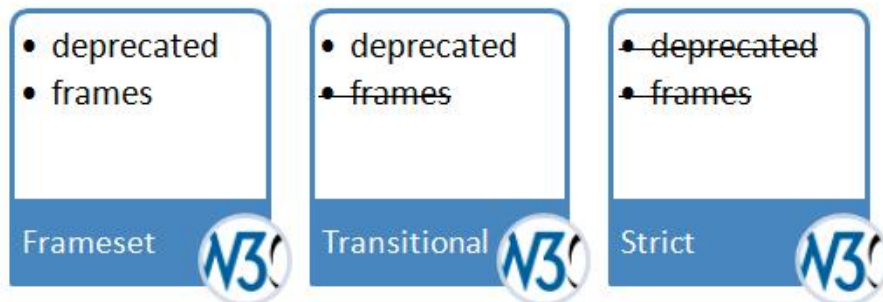
Na tento bod otázky lze pohlížet dvěma pohledy. Každý dokument v prostředí internetu by měl být označen tzv. MIME typem (obvykle označovaný jako Content-type) udávajícím formát souboru. Tato informace pomáhá internetovým prohlížečům s korektním zpracováním a zobrazením dokumentu. V HTML Content-Type označíme následujícím meta tagem.

```
<meta http-equiv="Content-Type" content="text/xhtml; charset=utf-8"/>
```

Druhý pohled na typ webových dokumentů se váže přímo k HTML. Existuje několik různých specifikací značkovacího jazyka HTML. Od historických HTML 2.0, HTML 3.2 přes současné HTML 4.01, XHTML 1.0 a XHTML 1.1 až po novodobé HTML 5. Většina z verzí je specifikována v několika různých podverzích, které se od sebe liší množinou podporovaných konstrukcí. Popíšme si dostupné verze dnes nejrozšířenější specifikace XHTML 1.0.

Podverze HTML

- XHTML 1.0 Transitional – Dodržuje syntaktické zásady XML. Je tolerantnější – respektuje některé zastaralé konstrukce (např. formátování značky <center>, , ...) a především z důvodů zpětné kompatibility podporuje přežití (deprecated) značky.
- XHTML 1.0 Strict – nejpřísnější podoba dané verze. Ekvivalentní se STRICT verzí HTML 4.01, dodržuje syntaktické zásady XML. Obsahuje tedy elementy a atributy HTML 4.01 s výjimkou těch označených za přežití (deprecated).
- XHTML 1.0 Frameset – velmi zřídka používaný Frameset umožňuje využívání rámců (frame), které byly v počátcích vývoje HTML velmi využívány.



V každém HTML dokumentu bychom měli deklarovat v jaké z verzí je dokument napsán. Tím sdělíme prohlížeči jak má dokument renderovat – podle které specifikace. To mu sdělíme speciální úvodní značkou jazyka HTML tzv. DOCTYPEM, který třeba pro XHTML 1.0 Strict bude vypadat takto:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

Jazyk HTML a XHTML

HTML je nejrozšířenější značkovací jazyk, používaný k formátování webových stránek. Zkratka je odvozená z anglického sousloví HyperText Markup Language. S první specifikací v roce 1991 přišel vědec výzkumného střediska CERN, Tim Berners Lee. HTML jako takové není rozhodně plnohodnotný programovací (ani skriptovací) jazyk, ale značkovací. Pouze dává datům sémantický a logický význam v rámci struktury dokumentu a prohlížeč se potom stará o odpovídající zobrazení, prezentaci těchto dat. Novější generací je potom XHTML, které vzniklo odvozením z HTML 4.01 a jazyka XML.

- HTML sestává z párových a nepárových značek (tagů). Párové musejí být ukončeny.
- Značky se zapisují ve tvaru <název existující značky>Obsah </název existující značky>.
- Dostupné značky jsou definované použitou specifikací.
- HTML není *case-sensitive*, nicméně XHTML již ano.
- XHTML ukončuje nepárové značky lomítkem – např.

- Hlavička obsahuje titulek stránky, vložené inline styly či skripty, meta tagy pro specifikaci různých informací (např. autor stránky, klíčová slova, content-type).

- Tělo uvozuje obsahovou, tedy uživatelem viditelnou část.
- Kořenovým elementem je párová značka *html*.
- Některé značky mají specifikované své atributy a rovněž existuje několik univerzálních platných pro všechny značky – identifikátor (*id="identifikátor"*) a definice stylové třídy (*class="název třídy"*) pro provázání s kaskádovými styly.

Základní „kostra“ HTML dokumentu

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html>
<head>
<title>Jednoduchá HTML stránka</title>
<meta http-equiv="content-type" content="text/xhtml; charset=utf-8">
</head>
<body>
<p>Můj první odstavec v HTML</p>
</body>
</html>
```

Nejběžnější HTML značky

<code><html></code> , <code><head></code> , <code><title></code> , <code><meta></code> , <code><body></code>	základní značky, jejich význam jsme si již vysvětlili výše
<code><p>text</p></code>	odstavec textu
<code><script></script></code>	definice skriptu, nejčastěji javascriptu
<code>odkaz</code>	definice hypertextového odkazu.
<code><h1-6></h1-6></code>	nadpisy různých úrovní 1-6
<code>
</code>	odřádkování
<code><div></div></code> či <code></code>	oddíl textu či části dokumentu
<code></code>	obrázek
<code><applet></code> či <code><object></code>	externí objekt nebo applet

Seznamy a tabulky v HTML

Neuspořádaný seznam (uspořádaný by byl ``)

```
<ul>
<li>odrazka 1</li>
<li>odrazka 2</li>
</ul>
```

Ukázka tabulky

```
<table border="1">
<tr>
<th>záhlaví 1. sloupce</th>
<th>záhlaví 2. sloupce</th>
</tr>
<tr>
<td>řádek 1, buňka 1</td>
<td>řádek 1, buňka 2</td>
</tr>
<tr>
<td>řádek 2, buňka 1</td>
<td>řádek 2, buňka 2</td>
</tr>
</table>
```

XML a DTD

XML (eXtensible Markup Language) je značkovací jazyk obdobně jako HTML s jedním zásadním rozdílem - rozšířitelností. Jednotlivé značky nejsou přímo specifikované standardem, ale definujeme je v rámci DTD (Document Type Declaration).

```
<?xml version="1.0"?>
<mojeznacka atribut="hodnota">
Moje speciální značka pro tuto velmi speciální větu.
</mojeznacka>
```

- Kořenový element se MUSÍ vyskytovat pouze jednou. V předchozím příkladě je kořenovým

elementem <mojeznacka>

- Všechny značky musí být ukončeny!
- S výjimkou nepárových, které mají ukončující znak přímo - např. <znacka-pouze-s-atributem tajnykod=„123“/>
- Možný výskyt značek a rovněž atributů definujeme v DTD.
- Pozor na křížení tagů! Příklad syntakticky nesprávného zápisu se křížícími se tagy.

```
<?xml version="1.0"?>
<mojeznacka>
  <dalsiznacka>
    Moje speciální značka pro tuto velmi speciální větu.
  </mojeznacka>
</dalsiznacka>
```

Definice vlastní gramatiky - DTD

XML je **well-formated** pokud neporušuje základní pravidla formátování. XML je **validní** pokud splňuje pravidla příslušné gramatiky.

Pravidlo pro definici elementu

```
<!ELEMENT jméno_elementu (obsah_elementu)> ... obecně
<!ELEMENT mojeznacka (#PCDATA)> ... pro ukázkový kód
```

?	nepovinný výskyt (nulakrát až jednou)
+	jednou či vícekrát
*	libovolný počet výskytů (nula až n)

Uvádí se za element či skupinu element. Například tedy pro jeden až více výskytů elementu odstavec a zároveň možný výskyt elementu obrázek bychom použili definici takto:

```
<!ELEMENT mojeznacka (odstavec+, obraze?)>
```

Pravidlo pro definici atributu

```
<!ATTLIST mojeznacka
atribut CDATA "hodnota"
>

//nebo pro povinný výskyt atributu "atribut"//

<!ATTLIST mojeznacka
atribut CDATA #REQUIRED
odsazení CDATA #IMPLIED
>

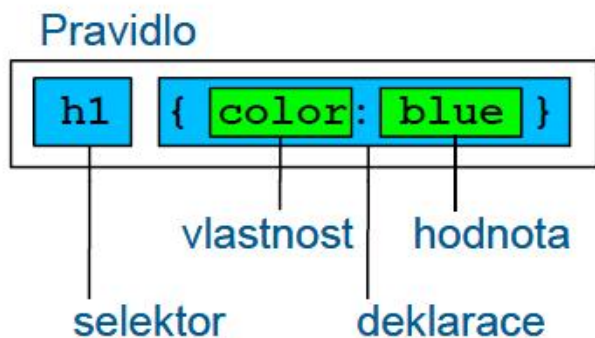
//nebo pro volitelný výskyt atributu "atribut"//

<!ATTLIST mojeznacka
atribut CDATA #IMPLIEDD
>
```

hodnota	přesná hodnota atributu
#REQUIRED	povinný výskyt
#IMPLIED	libovolný výskyt

Kaskádové styly CSS

Cascading Style Sheets (CSS) je jazyk pro popis způsobu zobrazení stránek. Pomocí pravidel popisuje formátování a vzhled daných elementů. Pravidla se aplikují podle deklarovaných vlastností, není-li použije se hodnota rodiče či implicitní hodnota.



Jednotky

relativní	em, ex, %
absolutní	px (pixel), cm, mm, pt (point), in (palec)

Selektory

*	univerzální selektor
E	selektor daného elementu
E.class	selektor elementů dané třídy (může být více elementů)
E#id	selektor elementu podle identifikátoru (unikátní)
E E2	selektor sourozenců v DOM stromě
E > E2	selektor dětí elementu v DOM stromě
E:hover	selektor pseudo-třídy (něco jako funkce CSS)
E[atr="val"]	selektor atributu

Dědičnost

Pro určení výsledku posloupnosti stylů (kaskády) je potřeba:

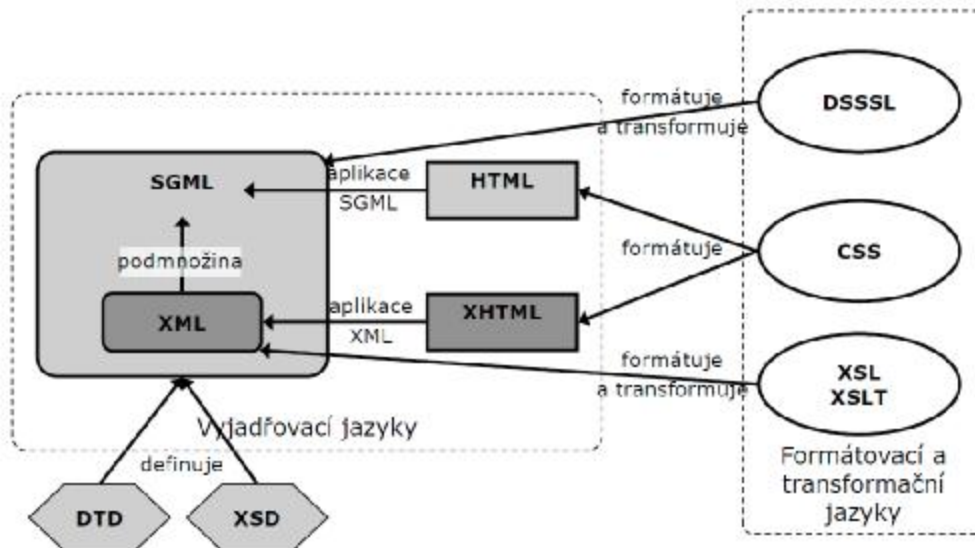
1. Najít všechny deklarace vázané na element (pro dané médium)
2. Seřadit je podle důležitosti ... UA (user agent) < uživatel < autor < autor !important < uživatel !important
3. Seřadit je podle specifičnosti (pravidla přesněji určující selektory mají přednost před obecnými)
4. Seřadit je podle pořadí výskytu (použije se pozdější pravidlo)

Připojení externího CSS souboru

```
<head>
  <link rel='stylesheet' href='style.css' type='text/css'>
</head>
```

Vzájemné vztahy těchto jazyků

Výstižně je popisuje následující obrázek propůjčený ze slidů. Rozděluje na jazyky popisující strukturu dokumentu (HTML, XML) a jazyky formátovací (CSS, XSLT).



Zdroje

- Stránky předmětu <http://amun.felk.cvut.cz/y36wmm/> [<http://amun.felk.cvut.cz/y36wmm/>]
- Přístupnost webu <http://pristupnost.nawebu.cz/> [<http://pristupnost.nawebu.cz/>]
- Základy HTML s přehledem tagů <http://www.jakpsatweb.cz/html/> [<http://www.jakpsatweb.cz/html/>]
- Základy CSS s přehledem vlastností <http://www.jakpsatweb.cz/css/> [<http://www.jakpsatweb.cz/css/>]
- Přehled doporučených DOCTYPE <http://www.w3.org/OA/2002/04/valid-dtd-list.html> [<http://www.w3.org/OA/2002/04/valid-dtd-list.html>]

spolecne/spol25.txt · Poslední úprava: 2010/06/05 21:19 autor: Laku