

3.1 Začínáme

Tato kapitola si vezme do parády stránku HTML, na které není vůbec nic špatného, a vylepší ji. Některé její části se zkrátí. Jiné se zase prodlouží. Celkově získá více sémantiky. A bude to super.

Tady je ta stránka¹. Učte se z ní. Žijte jí. Zamilujte si ji. Otevřete si ji v novém panelu a nevracejte se, dokud aspoň jednou nekliknete na „Zobrazit zdrojový kód“.

3.2 Doctype

Odshora:

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

Tomuto se říká „doctype“. Za doctypem se skrývá mnoho historie – a taky alchymie. Při práci na Internet Explorer 5 pro Mac se vývojáři v Microsoftu ocitli před nečekaným problémem. Nová verze jejich prohlížeče zlepšila svou podporu standardů natolik, že se starší stránky přestaly správně zobrazovat. Nebo lépe řečeno, ony se sice zobrazovaly správně (podle specifikací), ale lidé očekávali, že se budou zobrazovat nesprávně. Ty stránky byly totiž vytvořeny tak, aby vyhovovaly vrtochům tehdejších předních prohlížečů, hlavně Netscape 4 a Internet Explorer 4. IE5 pro Mac byl natolik pokročilý, že vlastně rozbil web.

Microsoft přišel s novátorským řešením. Před zobrazením stránky se IE5/Mac podíval na „doctype“, který obvykle tvoří první řádek zdrojového kódu HTML (ještě před prvkem `<html>`). Starší stránky (které se spoléhaly na zobrazovací vrtochy starších prohlížečů) obvykle doctype vůbec neměly. IE5 pro Mac zobrazoval tyto stránky stejně jako starší prohlížeče. Aby autoři stránek mohli „aktivovat“ podporu nových standardů, museli přejít na novou technologii tím, že před prvek `<html>` přidali správný doctype.

Tato myšlenka se velmi rychle rozšířila a brzo měly všechny hlavní prohlížeče dva režimy: „vrtošivý režim“ a „standardní režim“. Brzy se to vymklo z rukou. Když se v Mozille pokusili vydat verzi 1.1 jejich prohlížeče, zjistili, že se ve „standardním režimu“ zobrazovaly i některé stránky, které se ve skutečnosti spoléhaly na jeden konkrétní vrtoch. Mozilla jednoduše upravila svůj zobrazovací engine tak, aby tento vrtoch zmizel, a najednou přestaly fungovat tisíce stránek. Proto byl vytvořen – a to si nevymýšlím – „téměř standardní režim“.

1 <http://diveintohtml5.info/examples/blog-original.html>

Ve své zásadní práci *Activating Browser Modes with Doctype* (Aktivace režimů prohlížeče pomocí doctype) Henri Sivonen podává přehled jednotlivých režimů:

Vrtošivý režim

Ve vrtošivém režimu prohlížeče porušují současné webové specifikace, aby se vyhnuly „rozbíjení“ stránek vytvořených podle zvyklostí, které panovaly v druhé polovině 90. let.

Standardní režim

Ve standardním režimu se prohlížeče pokoušejí k vyhovujícím dokumentům chovat tak, aby to odpovídalo specifikacím, do takové míry, do jaké to konkrétní prohlížeč dovoluje. HTML5 tomuto režimu říká „režim bez vrtochů“.

Téměř standardní režim

Firefox, Safari, Chrome, Opera (od verze 7.5) a IE8 mají také režim známý jako „téměř standardní režim“, který provádí vertikální dimenzování buněk tabulky tradičně a ne přísně podle specifikací CSS2. HTML5 tomuto režimu říká „režim s omezenými vrtochy“.

(Měli byste si přečíst i zbytek Henriho článku, protože to tady říkám velmi zjednodušeně. Dokonce i v IE5 pro Mac se objevovalo několik starších doctypeů, které se nepočítaly jako přechod na podporu standardů. Seznam vrtochů se postupně rozšiřoval a spolu s tím přibývalo i doctypeů, které spouštěly „vrtošivý režim“. Když jsem to zkoušel počítat naposledy, napočítal jsem 5 doctypeů, které spouštěly „téměř standardní režim“ a 73, které spouštěly „vrtošivý režim“. Ale asi jsem na nějaké zapomněl a nebudu se ani pouštět do popisu šíleností, které dělá Internet Explorer 8, aby mohl přepínat mezi čtyřmi – slovy: čtyřmi zobrazovacími režimy. Tady je to na vývojovém diagramu². Brr. Odstup, satane!)

Tak, kde jsme to skončili? Aha, u doctypeu:

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

Tohle je zrovna jeden z 15 doctypeů, které spouštějí „standardní režim“ ve všech moderních prohlížečích. Není na něm nic špatného. Pokud se vám líbí, můžete u něho zůstat. Nebo ho můžete změnit na doctype HTML5, který je kratší a sympatičtější a rovněž spouští „standardní režim“ ve všech moderních prohlížečích.

Tohle je doctype HTML5:

```
<!DOCTYPE html>
```

To je vše. Jenom 15 znaků. Je tak jednoduchý, že ho můžete napsat ručně a nespěst se přitom.

² <https://hsivonen.fi/doctype/ie8-mode.png>

3.3 Kořenový prvek

Stránka HTML je sérií do sebe vnořených prvků. Celá struktura stránky je jako strom. Některé prvky jsou „sourozenci“, jako dvě větve, které vycházejí ze stejného kmene. Některé prvky jsou zase „potomky“ jiných prvků, jako dvě větvičky, které vycházejí ze stejné větší větve. (Funguje to i naopak: prvek, který obsahuje jiné prvky, se nazývá „mateřský“ uzel svých bezprostředních podřízených prvků a „předek“ svých vnuků.) Prvkům, které nemají potomky, se říká „listové“ uzly. Nejvzdálenější prvek, který je předkem všech ostatních prvků stránky, se nazývá „kořenový prvek“. Kořenovým prvkem stránky HTML je vždy `<html>`.

Na této příkladové stránce³ vypadá kořenový prvek takto:

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

Na tomto kódu není nic špatného. Zase platí, že pokud se vám líbí, můžete u něho zůstat. Je to regulární HTML5. Ale některé jeho části už v HTML5 nejsou potřeba, takže můžete ušetřit pár bajtů tím, že je odstraníte.

První věcí k diskusi je atribut `xmlns`. Jedná se o pozůstatek XHTML 1.0. Říká nám, že prvky na této stránce jsou ve jmenném prostoru XHTML `http://www.w3.org/1999/xhtml`. Ale prvky v HTML5 jsou vždycky v tomto jmenném prostoru, takže už to nemusíte vysloveně prohlašovat. Vaše stránka HTML5 bude ve všech prohlížečích fungovat úplně stejně, ať už bude či nebude obsahovat tento atribut.

Když vypustíme atribut `xmlns`, zůstane nám tento kořenový prvek:

```
<html lang="en" xml:lang="en">
```

Oba zde použité atributy `lang` a `xml:lang` definují jazyk této stránky HTML. (`en` označuje angličtinu. Nepíšete anglicky? Najděte si kód svého jazyka⁴.) Proč se pro jednu věc používají dva atributy? Opět se jedná o pozůstatek XHTML. V HTML5 hraje roli pouze atribut `lang`. Jestli chcete, můžete atribut `xml:lang` zachovat, ale pokud to uděláte, musíte zajistit, aby obsahoval stejnou hodnotu jako atribut `lang`.

Aby se ujasnilo přecházení do a z XHTML, mohou autoři stránek v prvcích HTML v dokumentech HTML určit atribut, který nebude ležet v žádném jmenném prostoru, nebude obsahovat žádný prefix a bude mít doslovné `localname` (místní jméno) „`xml:lang`“. Tyto

³ <http://diveintohtml5.info/examples/blog-original.html>

⁴ <http://www.w3.org/International/questions/qa-choosing-language-tags>