

tvorba webu

Tabulkový layout a linky



Martin Urza



Layout (rozvržení) stránky

- Layout HTML dokumentu znamená rozložení prvků na stránce. S anglickým výrazem (layout) se nejspíš setkáte častěji než s českými (rozvržení, pozicování).
- Rozvržením prvků na stránce je míněno, kde budou které umístěny.
 - Například: Vlevo bude menu o šířce 15% celé stránky, nahoře bude lišta o výšce 100px, obsah stránky tedy bude vpravo dole.
- Povšimněte si, že každé (normální, tj. ne „blogýsky“) stránky mají nějakou (spíše menší) statickou část, která slouží k navigaci, typicky nějaké menu; obsah stránek se pak mění podle toho, kam uživatel klikne.

Jak vytvořit layout stránky?



- V zásadě existují tři způsoby, jak rozvrhovat prvky na stránce:
 - CSS layout (pozicování pomocí kaskádových stylů),
 - table layout (rozvržení pomocí tabulky),
 - frames layout (rozložení pomocí rámců).
- Ač se kolem těchto tří přístupů vedly (před několika lety) dalekosáhlé polemiky, dnes už se (snad) všichni shodli, že nejlepším přístupem je formátovat stránku pomocí kaskádových stylů (CSS layout).
- Protože kaskádové styly nejsou zcela triviální, bude jako první vyučován tabulkový layout (frames layout probírat dnes už nemá smysl).



CSS layout

- Nejmodernější nástroj na pozicování HTML stránek.
- Hlavní výhodou tohoto layoutu je, že v dokumentu vůbec samotné rozložení prvků na stránce neřešíme. Místo toho pouze označujeme prvky podle toho, co znamenají (menu, obsah, lišta, banner,).
- V **jiném** souboru pak specifikujeme, kde bude co na stránce umístěno, což dovoluje měnit vzhled stránky bez zásahu do jejího kódu, případně je možné mít jeden web s mnoha různými přepínatelnými vzhledy.
- Stránky se rychleji zobrazují, jejich kód je menší a přehlednější, pořadí textu v kódu nemusí odpovídat fyzickému rozložení (dobré pro lineární prohlížeče).

Frames layout



- Módní výstřelek z roku 1996.
- Jedinou výhodou tohoto přístupu je možnost vkládat do jednotlivých rámců HTML soubory (výhodu plně oceníte, až budete tvořit menu v jiných layoutech).
- Krom této jediné výhody má frames layout jen samé nevýhody; v adresním řádku prohlížeče není vidět adresa toho, co právě prohlížíte (těžko pošlete link, neuložíte do oblíbených položek, špatně se odkazuje zvenku), ukládání stránek na disk je problematické, vyhledávače odkazují pouze na vnitřní stránku rámu, staré a exotické prohlížeče rámy vůbec neumí, celá stránka není nikdy aktivní, vždy pouze rám,

Table layout



- Design pomocí tabulek nemá sám o sobě až tak moc nevýhod (jako třeba rámce – frames layout), nicméně oproti CSS layoutu je trochu neohrabaný:
 - Obsah a vzhled stránky musí být smíchan do jednoho HTML souboru, což činí kód nepřehledným.
 - Pro změnu vzhledu HTML stránky je třeba modifikovat parametry tabulky, které jsou rozházené napříč celým dokumentem (a navíc promíchané s obsahem).
 - Stránky se v tabulkovém layoutu zobrazují pomaleji, protože prohlížeč musí napřed přečíst celou tabulku, než zjistí, jak ji má vlastně zobrazit.
- Hlavní výhodou tabulkového designu je asi značná jednoduchost, proto jím začneme.



Table layout - jak na to?

- Všechny potřebné znalosti byste měli mít z minulé přednášky, stačí se jen zamyslet ;o)
- Stránku můžete do libovolného tvaru upravit tak, že navrhnete neviditelnou tabulku (`border="0"`), která bude obsahovat všechny prvky stránky umístěné do odpovídajících buněk.
 - Využívejte v co největší míře slučování buněk tabulek, zejména při tvorbě menu je důležité, aby bylo celé v jedné buňce, což usnadní (již tak dost problematickou) údržbu.
 - Šířku buněk lze nastavovat v absolutních číslech, i v poměru k šířce celé tabulky (v procentech).

Cvičení



- Napište jednu HTML stránku, která bude obsahovat:
 - Záhloví (s nějakým textem, obrázkem, nadpisem, ...).
 - Menu na levé straně.
 - Položky v menu zatím nebudou sloužit jako odkazy.
 - Zápatí (např. s copyrightem, datem, jménem, ...).
 - Podmínka je, aby v zápatí byly na „jedné řádce“ (tedy ve vertikálně stejné výšce) dva texty; jeden bude zarovnaný doleva a jeden doprava.
 - Zamyslete se, zda toho lze dosáhnout i jinak.
 - Obsah ve zbytku stránky.
 - Zde by měl být alespoň jeden obrázek a tabulka.
 - Minimálně jedna z výše uvedených částí by měla mít jinou barvu pozadí.

Sít WWW

- To, co dělá z webu síť, je možnost propojování HTML dokumentů odkazy. Tyto odkazy lze vnímat tak, že stránky na sebe navzájem „ukazují“.
- Prohlížení webu by mělo odpovídat lidskému myšlení založeném na asociacích (něčím se zabýváte, přitom si rozpomenete na něco jiného -> odkaz).
 - Tento koncept byl vymyšlen už dřív, první s ním přišel systém MEMEX (nikdy nebyl implementován), potom Xanadu (implementován, ale neúspěšný).
- Odkazy v HTML jsou jednosměrné (např. v Xanadu obousměrné), což má výhody (je snadné odkazovat na cizí stránky) i nevýhody (nefunkční linky).



Tag a

- Základním nástrojem pro realizaci odkazů v jazyce HTML je element, který se jmenuje a (zkratka slova anchor).
- Tento element slouží primárně k vytvoření odkazu, který směřuje (ukazuje) na nějaký jiný dokument, jenž ani nemusí být nutně formátu HTML.
- Tag a lze také využít k vytvoření značky, jež označí nějaké místo v HTML dokumentu, na které se budou moci odkazovat jiné dokumenty.
 - Prohlížeče pak po otevření odpovídajícího dokumentu scrollují tak, aby bylo označené místo (značka) nahoře na stránce.

Atribut href



- Href je nejdůležitějším atributem elementu a; jeho hodnotou je cíl odkazu (URL, tedy adresa nějakého dokumentu).
 - URL adresa může být relativní vzhledem k dokumentu s odkazem, např. `obraz`.
- Je-li v cílovém dokumentu značka, lze skočit přímo na ni uvedením křížku (#) a jména značky za jméno dokumentu.
 - Například `http://www.urza.cz/soubor.html#znacka`.
- Obsah tagu a v tomto případě říká, co bude sloužit jako odkaz (to může být i tabulka, obrázek,).
 - `Urzovy stránky`

Atribut name



- Name je atribut elementu a, který umožňuje vytvářet v HTML dokumentu značku, na může odkazovat jiný link (tedy tag a).
- Tag a s atributem name může být prázdný, případně nepárový (což je totéž).
- Příklad: Na místo označené `` v souboru test.html na serveru urza.cz se lze odkázat `takto`.
- Chceme-li se odkazovat na značku v rámci souboru, do odkazu nemusíme psát jeho adresu.
- Příklad: Na místo `` se lze v rámci souboru odkázat `takto`.

Atribut target



- Atribut target má smysl v elementu a, jenž obsahuje atribut href.
- Target říká, kam se má cíl odkazu otevřít.
- Chceme-li, aby se odkaz otevřel do nějakého rámce, což je zejména případ frame layoutu (toho špatného), napíšeme jméno rámce do hodnoty atributu.
- Dále existují přednastavené hodnoty:
 - `_blank` otevře odkaz do nového tabu (okna).
 - `_self` otevře odkaz do aktuálního rámce (stránky).
 - `_top` a `_parent` se používá opět pro frames layout.
- `Tento odkaz otevře stránku v novém okně (tabu).`

Další atributy tagu a



SENSE
This picture makes none

- Atribut rel říká něco o obsahu cíle odkazu.
 - Standardizované hodnoty: copyright, alternate, help, stylesheet, start, next, prev, contents, index, glossary, chapter, section, subsection, appendix, bookmark.
 - A celé je to asi k ničemu.
 - Google zavedl hodnotu nofollow, jež říká botům, aby link nenásledovali (respektuje to většina vyhledávačů).
- Atribut hreflang říká, v jakém jazyce je cíl odkazu.
 - Protože cílová stránka by měla mít tento jazyk stejně explicitně uvedený v kódu, je to nejspíš též k ničemu.
- Atribut ping pomáhá počítat přístupy na odkaz.
 - A opět se to nevyužívá.

Cvičení



- Napište webové stránky s následujícími parametry:
 - Skládají se z minimálně osmi HTML dokumentů, jenž jsou propojené linky (některé směřují jinam než na začátek stránky).
 - Všechny stránky obsahují (stejně) menu s odkazy na ostatní dokumenty.
 - **Stránky mají nějaký SMYSLUPLNÝ obsah!!** To není samoučelný požadavek, vyplývají z toho další nároky.
 - Celé by to mělo nějak rozumně vypadat (tj. nemělo by to být úplně odporné ve stylu „blikající blogýsek“).
 - **VŠECHNY stránky musí být validní!!**
 - Musíte rozumět všem prostředkům, které při tvorbě použijete (neumíte-li kaskádové styly, nepoužívejte je).

Rekapitulace



- V této chvíli byste měli být schopni napsat web podle téměř libovolné předlohy (buť třeba neohrabaně a ne úplně šikovně – zejména kvůli použití tabulkového layoutu).
- **Měli znát jazyk HTML a rozumět mu dost na to, abyste byli schopni (s pomocí pana Googla) užít i jiných elementů a atributů, než byly použity v rámci tohoto kurzu.**
- Stránky, které tvoříte, by měly být zaprvé validní, zadruhé korektně zobrazitelné ve všech „hlavních“ prohlížečích, zatřetí by neměly být vizuálně hnusné a začtvrté byste měli chápat, proč psát web validně.