

Informatika pro moderní fyziky (10)

HTML a JS - interaktivní dokumenty, použití cizích API

František HAVLŮJ

e-mail: haf@ujv.cz

ÚJV Řež

oddělení Reaktorové fyziky a podpory palivového cyklu

akademický rok 2016/2017

7. prosince 2016

- 1 Použití cizích API
- 2 HTML jako prezentační nástroj
- 3 Další triky s HTML
- 4 JavaScript

Obsah

- 1 Použití cizích API
- 2 HTML jako prezentační nástroj
- 3 Další triky s HTML
- 4 JavaScript

K čemu to?

- spousta informací na webu je poskytována ve strojově čitelné formě
- API – rozhraní mezi aplikacemi
- s využitím webových služeb naše možnosti exponenciálně rostou (počasí, doprava, mapy, atd atd.)
- spousta věcí se dá udělat jako *mashup* – sice nic neumím, ale umím to dát dohromady

Typy / formáty

- URL – rovnou dostanu např. obrázek po zadání správného URL
- XML – velmi obecný, ale komplikovaný formát (“vypadá jako HTML”)
- JSON – velmi jednoduchý a kompaktní formát, vyvinutý pro JS (v podstatě jen číslo, řetězec, pole, hash)

URL API – google maps

- stačí správně vymyslet
- pozor na usage limits (v produkci je nutné lokální cache...)
- QR platba:

`http://qr-platba.cz/pro-vyvojare/restful-api/#generator-czech-image`

- Google Maps static API:

`https://developers.google.com/maps/documentation/staticmaps/`

Jednoduchý mashup: mapa o-závodů

- ORIS API – <http://oris.orientacnisporty.cz/API>
- úkol: vypišme kalendář MTBO závodů v roce 2016
- <http://oris.orientacnisporty.cz/API/?format=json&method=getEventList&sport=3&datefrom=2016-01-01&dateto=2016-12-31>
- <https://developers.google.com/maps/documentation/javascript/examples/map-simple>
- API klíč – v praxi si musíte pořídit vlastní (ale nic to nestojí)
- AlzaSyC6_NpMTN-8oIOHWzaiwjeZ8eu_J3XI1IM

Obsah

- 1 Použití cizích API
- 2 HTML jako prezentační nástroj**
- 3 Další triky s HTML
- 4 JavaScript

Výhody HTML+JS

- PDF zpráva je fajn, ale umožňuje jen “plochou” strukturu
- často je potřeba prezentovat tak velké množství informací, že to nejde jenom nalepit za sebe
- hodilo by se něco klikacího
- příklad – ASTRID (výstup z programu ANDREA)

Výhody HTML+JS

- absolutní přenositelnost
- použitelné ve webových aplikacích i offline
- celkem snadno se zařídí, aby to vypadalo dobře
- JS zajišťuje slušný stupeň interaktivity

HTML

- stromová struktura tagů - sice trochu zdlouhavá, ale zato srozumitelná a přehledná
- jazyk pro internetové stránky
- logicky strukturované dokumenty, klíčovým prvkem jsou odkazy
- při správném používání stejně jako LaTeX definuje strukturu a obsah, nikoli vzhled

CSS a JS

- radosti pro web:
- Cascading StyleSheets - snadná a pohodlná cesta ke stylování
- javaScript - jazyk pro client-side interaktivitu

JavaScript

- trochu divný jazyk, historicky vzniknul jako ideový koncept a omylem se z toho stala definitivní verze
- potvora s nejasnou a dost netradiční syntaxí
- naštěstí na většinu základního využití není potřeba mu moc rozumět (já jsem důkazem)

HTML/CSS/JS a automatizace

- opět se jedná o plaintext dokumenty, takže je můžeme vesele generovat z libovolného skriptu
- jednak lze použít jako prezentační vrstvu (vzpomeňte model/view z minula) různých webových aplikací
- jednak lze použít jako samostatné interaktivní client-side dokumenty, které mohou být hezké, přenositelné

Standardy na webu

- pro HTML, CSS, JS existuje závazná specifikace, kterou dodržují všechny standardní prohlížeče
- nečestnou výjimku tvoří Internet Explorer – funguje prostě jinak, dřív se normálně musely psát alespoň dvě verze stránek a používat všemožné hacky
- s každou verzí IE se ale podpora zlepšuje (od v.7 výš to není naprostá katastrofa)
- i mezi standardními prohlížeči mohou být drobné rozdíly, zejména v nových částech specifikace
- problematiku rozdílů (zejm. v CSS) a těžkopádnost jazyka (zejm. v JS) řeší různé CSS/JS frameworky, které poskytují moderní cross-browser rozhraní a pohodlnou práci

Zatím samotné HTML

- příklady viz adresář html
- základem jsou tagy – stromová struktura
- `<tag> ... </tag>`
- `<tag atribut="hodnota"> ... </tag>`
- v některých případech (tagy bez obsahu) jen `<tag>`
- základní strukturu (hlavička) obšlehněte ze vzoru

Úkol trochu nadlouho - ale jednoduchý

- to samé co je v LaTeXu udělat v HTML
- následně doplníme interaktivitu

Základní struktura a nejjednodušší tagy

```
<!DOCTYPE html>
<html>
<head>
  <title>Provozní záznamy JE Třeskoprsky</title>
  <meta http-equiv='Content-Type' content='text/html; charset=UTF-8' />
</head>
<body>

  <h1>Provozní záznamy JE Třeskoprsky</h1>

  <p>Tento dokument obsahuje provozní záznamy z jaderné elektrárny Třeskoprsky.</p>

  <h2>Kampaň c01</h2>

  <h3>Koncentrace kyseliny borité</h3>

  <p>Doplníme tabulku a graf.</p>

</body>
</html>
```

Základní tagy

- `<p>...</p>` – odstavec (text)
- `h1`, `h2`, `h3`, `h4`, `h5` – nadpisy
- ``

Úkoly

- HTML dokument s nadpisy a grafy (zatím bez tabulek)
- proměnný počet kampaní!!!

Obsah

- 1 Použití cizích API
- 2 HTML jako prezentační nástroj
- 3 Další triky s HTML**
- 4 JavaScript

Tabulky v HTML

Nejjednodušší varianta:

```
<table>
  <tr>
    <td>25. 12. 1991</td><td>-3.798</td>
    <td>09. 03. 1992</td><td>-1.909</td>
    <td>23. 05. 1992</td><td>-1.037</td>
  </tr>
  <tr>
    <td>28. 12. 1991</td><td>-3.420</td>
    <td>12. 03. 1992</td><td>-1.825</td>
    <td>26. 05. 1992</td><td>-0.995</td>
  </tr>
</table>
```

- unordered list – `ul`
- jednotlivé položky/odrážky – `li`
- lze vnořovat (dáme `ul` dovnitř `li`)

```
<ul>  
  <li></li>  
  <li></li>  
  <li></li>  
</ul>
```

Odkazy na stránky i uvnitř stránek

- pomocí tagu `<a>` se vytváří jak odkaz, tak jeho cíl
- odkaz atributem `href`: `Google`
- cíl atributem `name`: ``
- pak odkaz uvnitř stránky:
`Kampaň 1`

Vkládání souborů

- HTML takovou věc neumí (a z principu dost dobře nemůže)
- naštěstí máme ERb, přímo v HTML to pro naše účely nepotřebujeme
- připomínám `IO.read(filename)`

Úkoly

- vložit kromě grafů i tabulky
- doplnit klikací obsah

Obsah

- 1 Použití cizích API
- 2 HTML jako prezentační nástroj
- 3 Další triky s HTML
- 4 JavaScript**

JavaScript

- trochu obskurní jazyk bez ladu a skladu
- dá se vkládat do HTML a používá se pro client-side aplikace
- dříve to bylo trochu zlo, dneska se z toho stala
- rozšířenost dalece převyšuje kvalitu

JS frameworky

- vrstva “překrývající” JS a usnadňující základní věci, často zejména manipulaci se stránkou a
- bez nich je JS stěží využitelný
- aktuální standard je jQuery
- množství dalších knihoven a rozšíření

JavaScript v HTML

- event-driven jazyk
- kód se nespouští “jen tak”, ale pouze na základě události
- kliknutí, načtení stránky, uplynutí času, klávesnice ...
- pro nás nejužitečnější `onclick`; např. `...`

Schovávání a zobrazování

- každému tagu můžu přiřadit id ``
- v jQuery se potom na příslušný tag odkážu takto:
`$ (' #image1')`
- `$ (' #image1').hide();`
- ``
- navíc: je potřeba přidat `return false` na konec handleru
- obdobně `show` a `toggle`
- kousek CSS:
``

JavaScript - jak jQuery nahrát

- je možné se v dokumentu odkázat na jiné JS soubory
- obvykle je to lepší než tam text dokumentu kopírovat...
(přehlednost, caching)
- odkaz se vkládá do hlavičky dokumentu (<head>)
- `<script src="jquery-1.4.4.min.js"
type="text/javascript" ></script>`

Úkoly

- doplnit přepínání mezi grafem a tabulkou
- vylepšit z programátorského hlediska (DRY princip a AO/boritá)

A to je vše, přátelé!

