
Technologie pro web a multimedia

HTML



Připomenutí z minula

- Klient – server architektura
- Protokoly IP – TCP – HTTP
- HTTP protokol
 - Textový
 - Hlavička
 - Prázdný řádek
 - Tělo
- Architektura webové aplikace



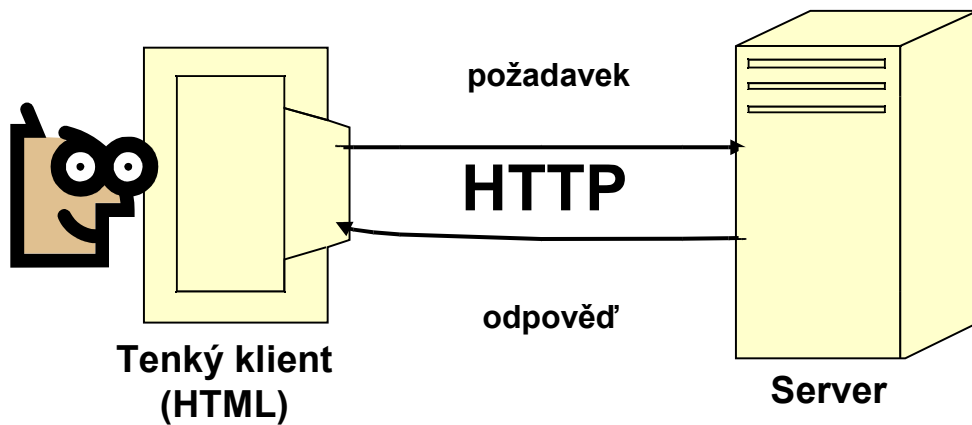
Architektura Webové Aplikace



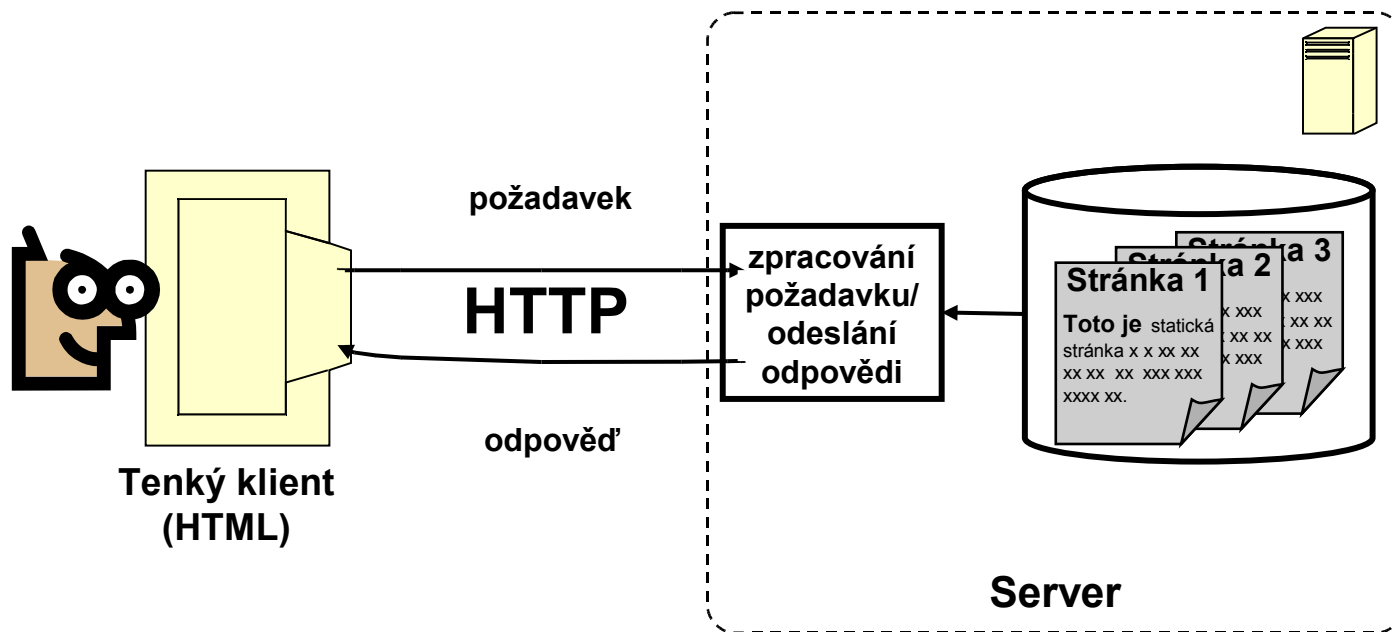
Computer Graphics Group



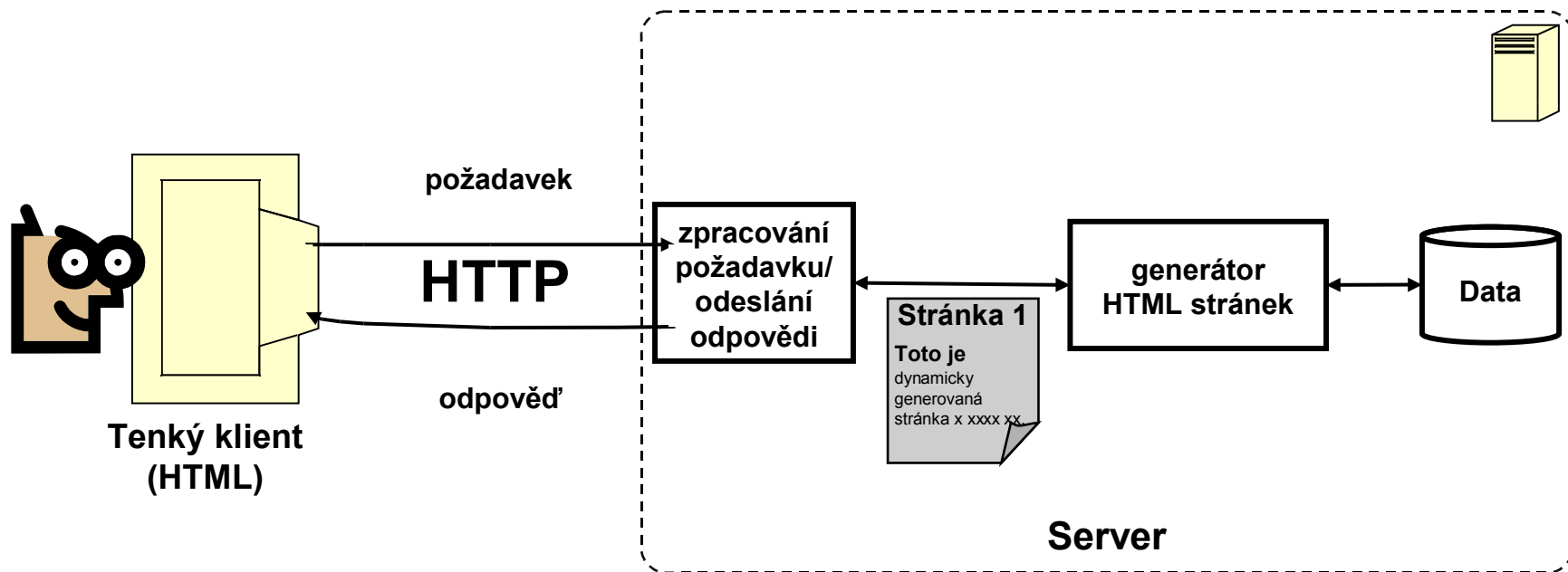
Architektura web aplikace



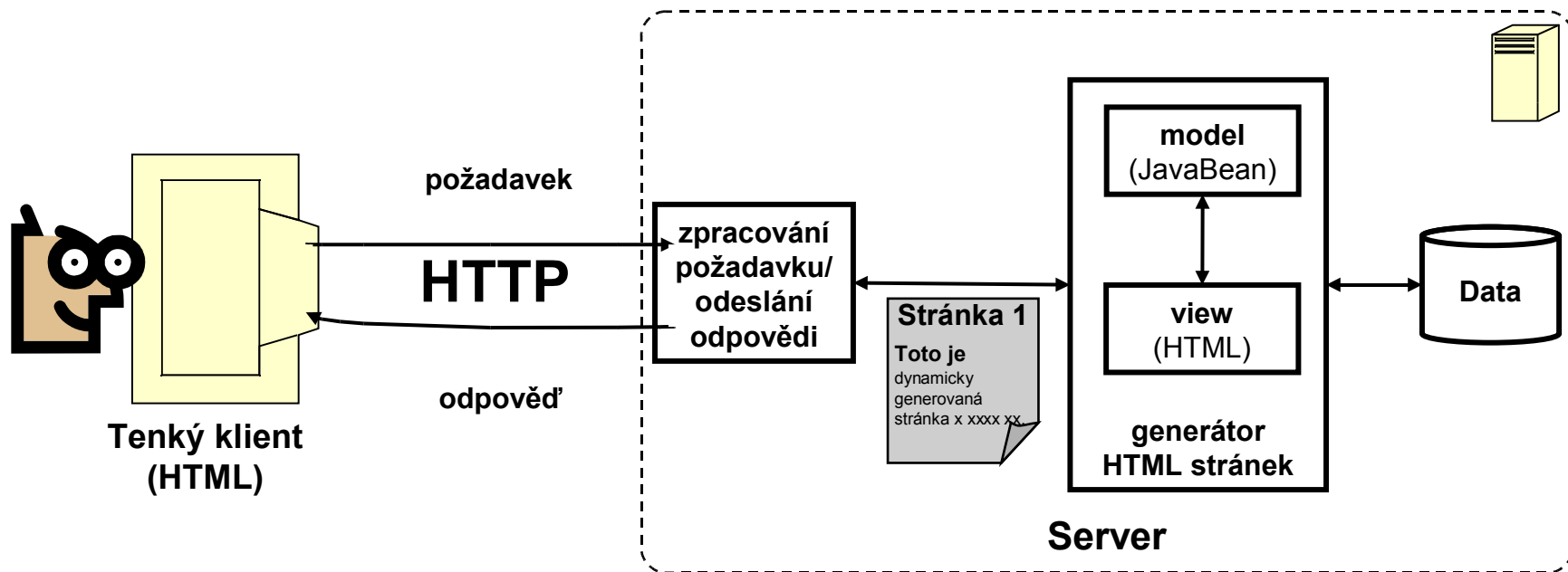
Architektura web aplikace: statický web



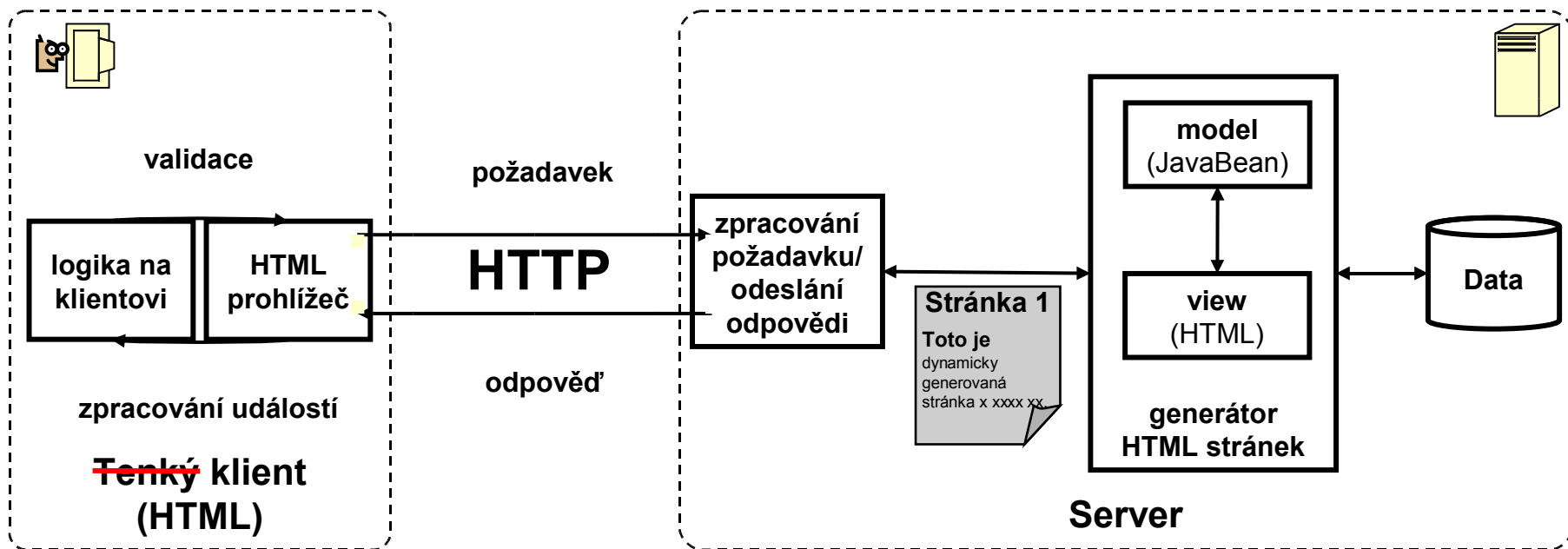
Architektura web aplikace: dynamický web



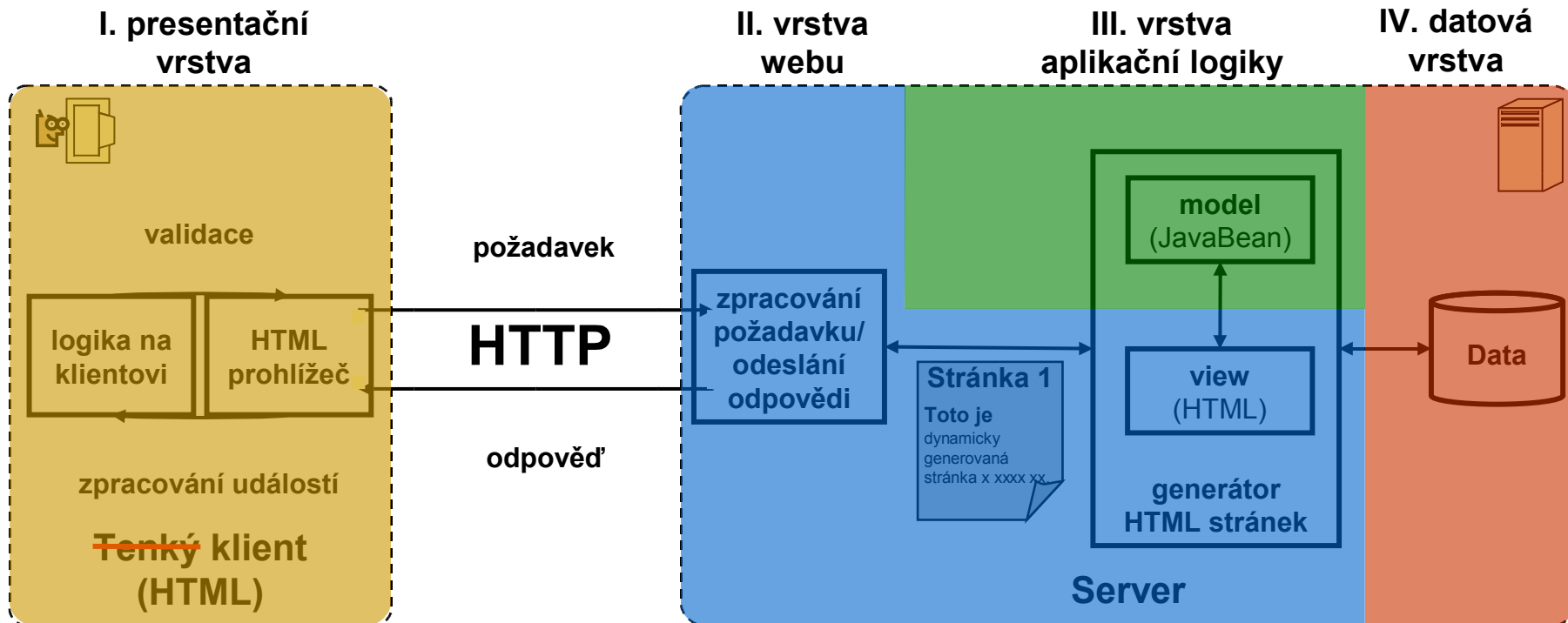
Architektura web aplikace: dynamický web



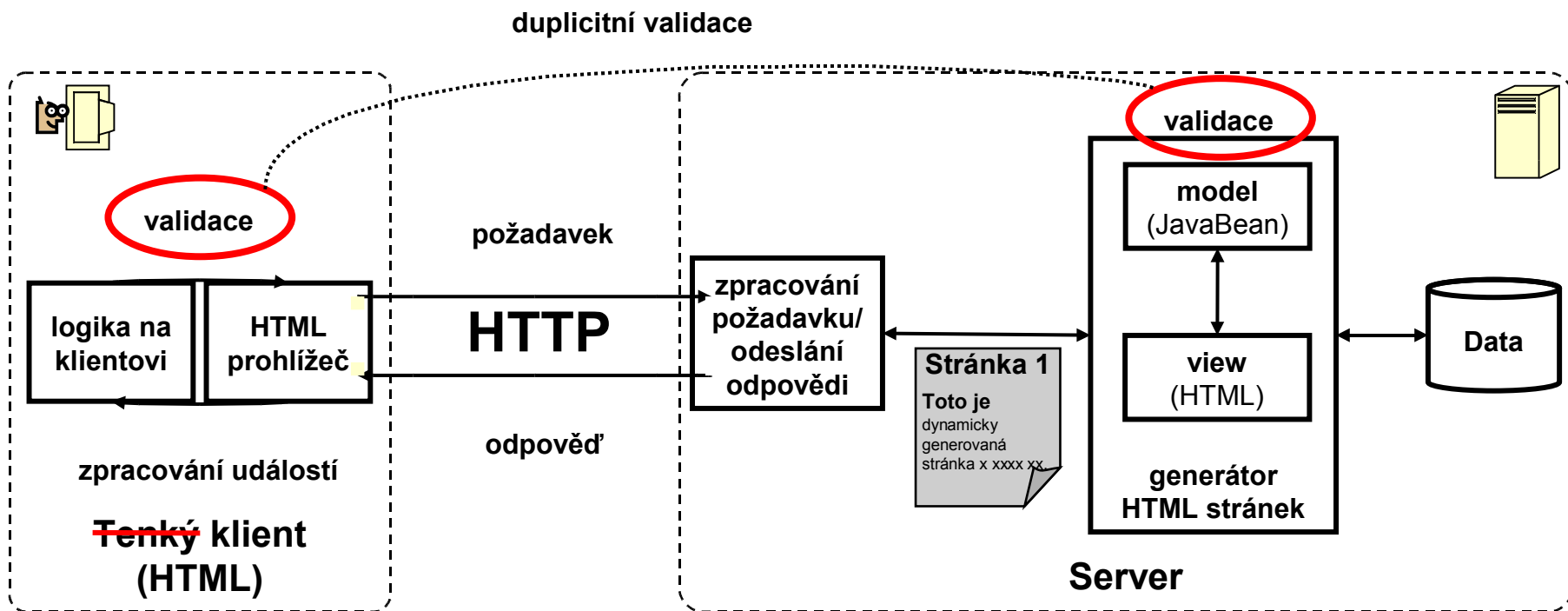
Architektura web aplikace: dynamický web



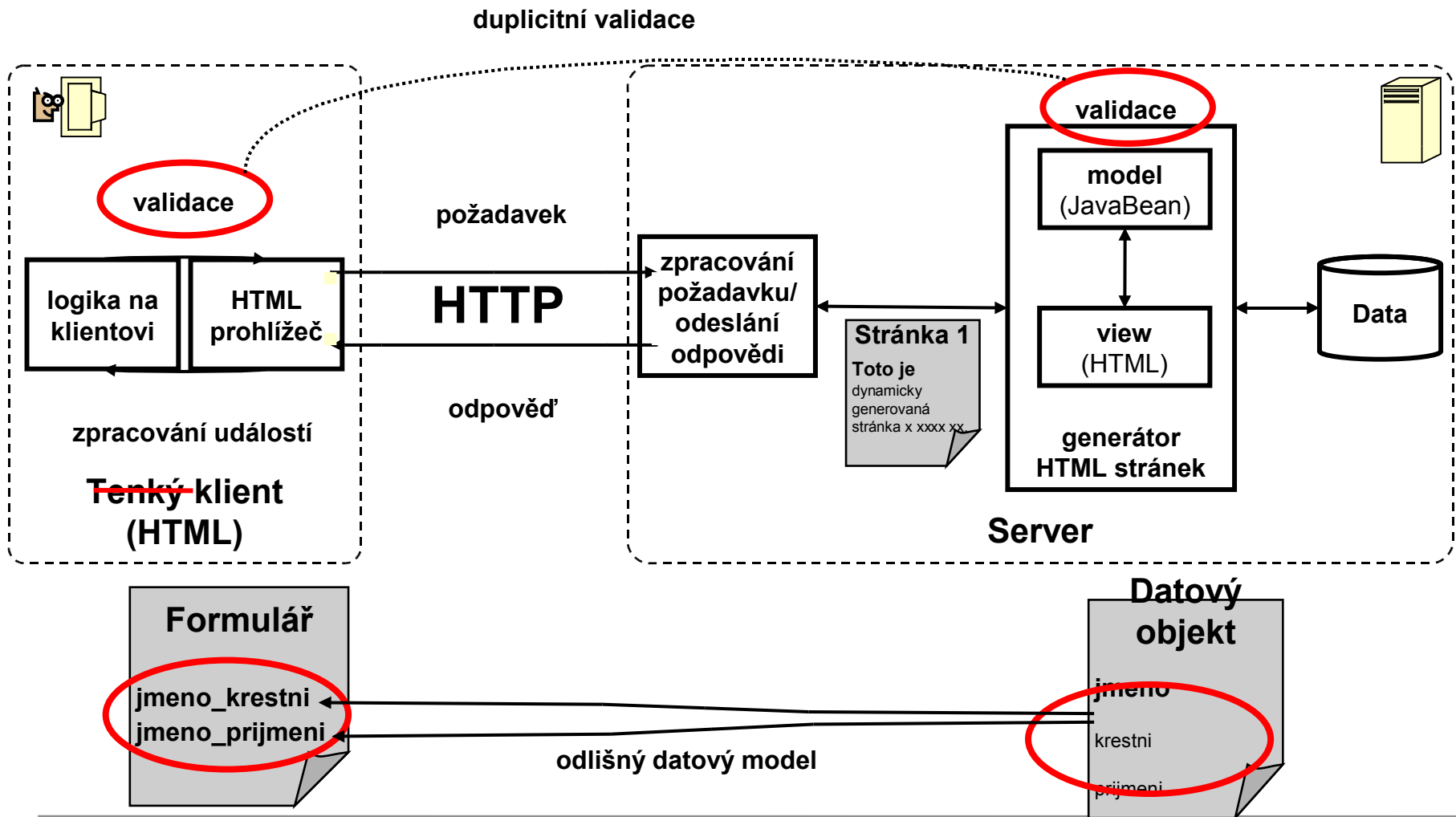
Architektura web aplikace: dynamický web



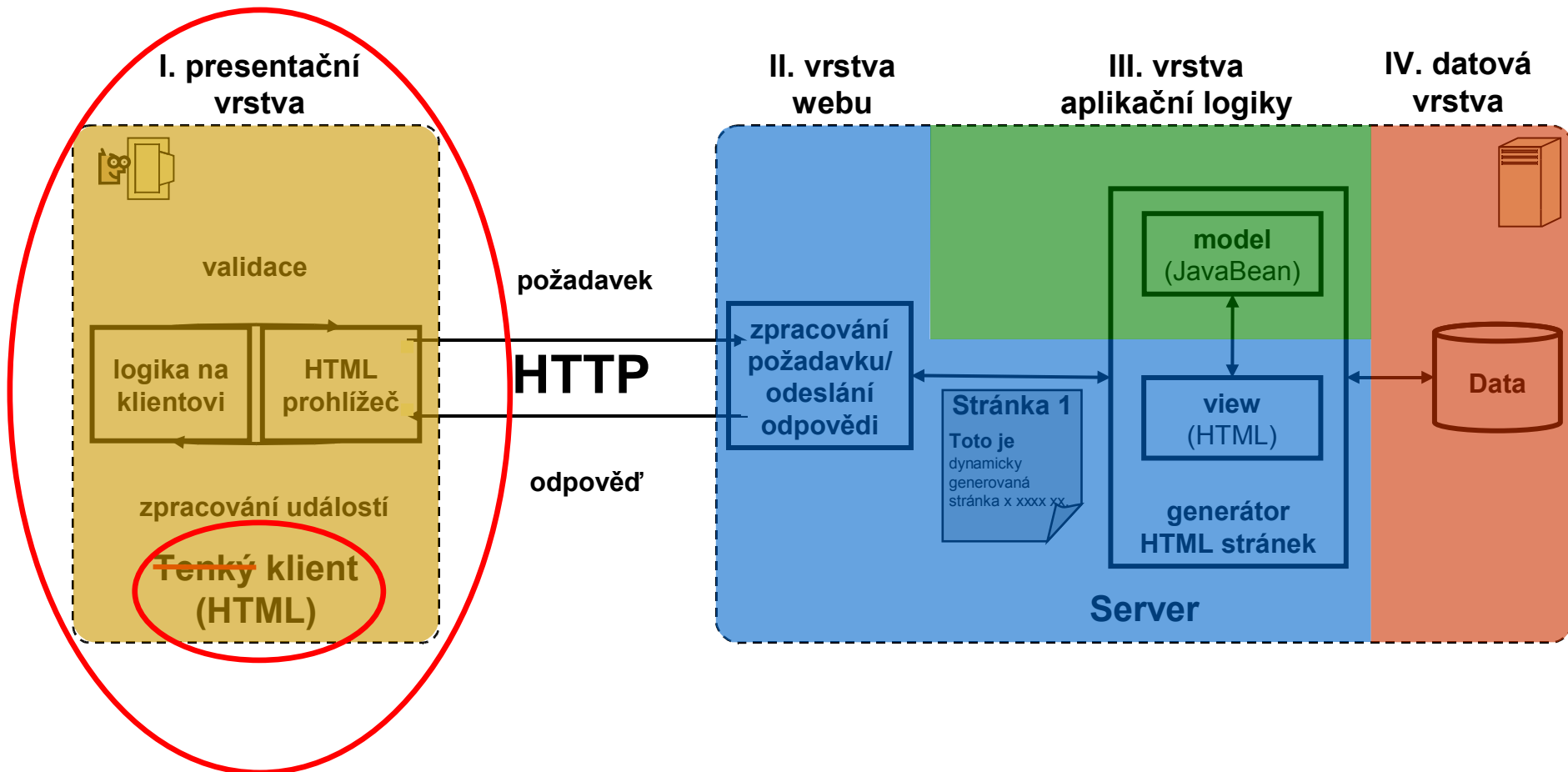
Architektura web aplikace: problémy



Architektura web aplikace: problémy



Architektura webové aplikace



Dnešní téma – zpracování HTML na straně klienta

- HTML – jazyk, ve kterém je popsána struktura (a vzhled) webové stránky
 - Starší verze HTML popisovaly jak strukturu, tak vzhled
 - V nových verzích (4.x) to už **nechceme**
- HTML = Hyper Text Markup Language
- Jazyk pro psaní „hyper textů“ pomocí značek
- Vývoj jazyka
 - 1989 první HTML dokumenty v CERNu (Tim Berners Lee a Robert Caillau)
 - 1991 CERN zprovoznil první Web
 - 1993 první grafický web klient Mosaic
 - 1994 založeno World Wide Web Consortium (W3C)
 - 1994 HTML verze 2.0
 - 1996 HTML verze 3.2
 - 1997 HTML verze 4.0



HTML a jeho vztah k ostatním jazykům

- **SGML** (*Standard Generalized Markup Language*)
je univerzální značkovací metajazyk, který umožňuje definovat značkovací jazyky jako své vlastní podmnožiny. SGML je komplexní jazyk poskytující mnoho značkovacích syntaxí, ale jeho složitost brání většímu rozšíření.

...citace z Wikipedie

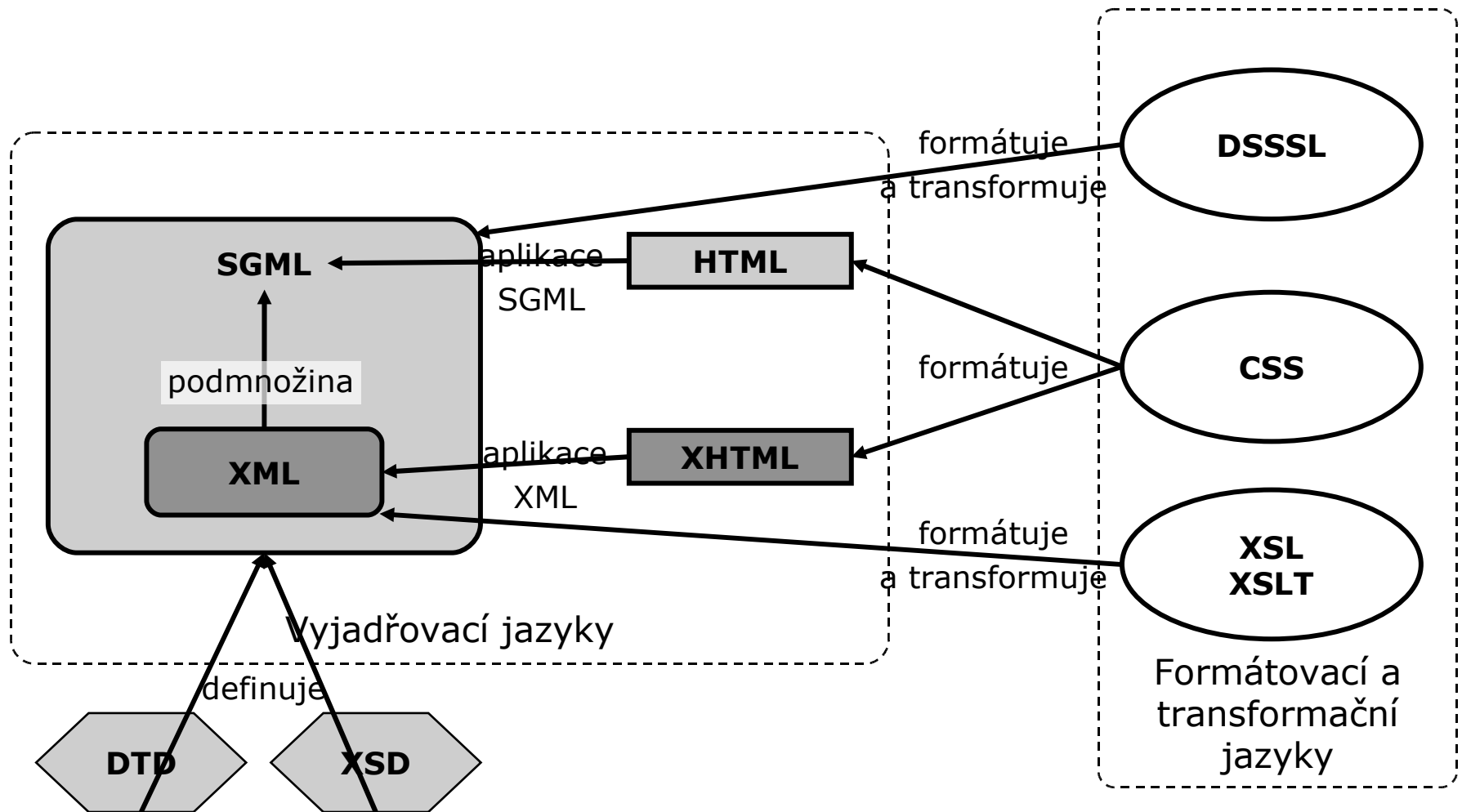
- **HTML** je podmnožinou SGML (Standard Generalized Markup Language)
- **XML** je podmnožinou SGML
- **XHTML** je aplikací XML

- Další jazyky, které používáme ve webových aplikacích:

– CSS, XSL, XSLT



HTML a jeho vztah k ostatním jazykům



HTML verze

■ HTML 2.0

- První ucelená specifikace HTML, definuje jej jako podmnožinu SGML

■ HTML 3.0

- Nikdy nebyla přijata, protože byla příliš složitá

■ HTML 3.2

- Novinky: tabulky, zarovnání a obtékání textů

■ HTML 4.0

- Novinky: podpora jazyků, oddělení struktury a designu (zavádí CSS), vylepšeny tabulky, formuláře a Java scripty a mnoho dalšího
- Definuje podverze: strict, transitional, frameset

■ HTML 4.01

- Současná verze, byly opraveny některé chyby

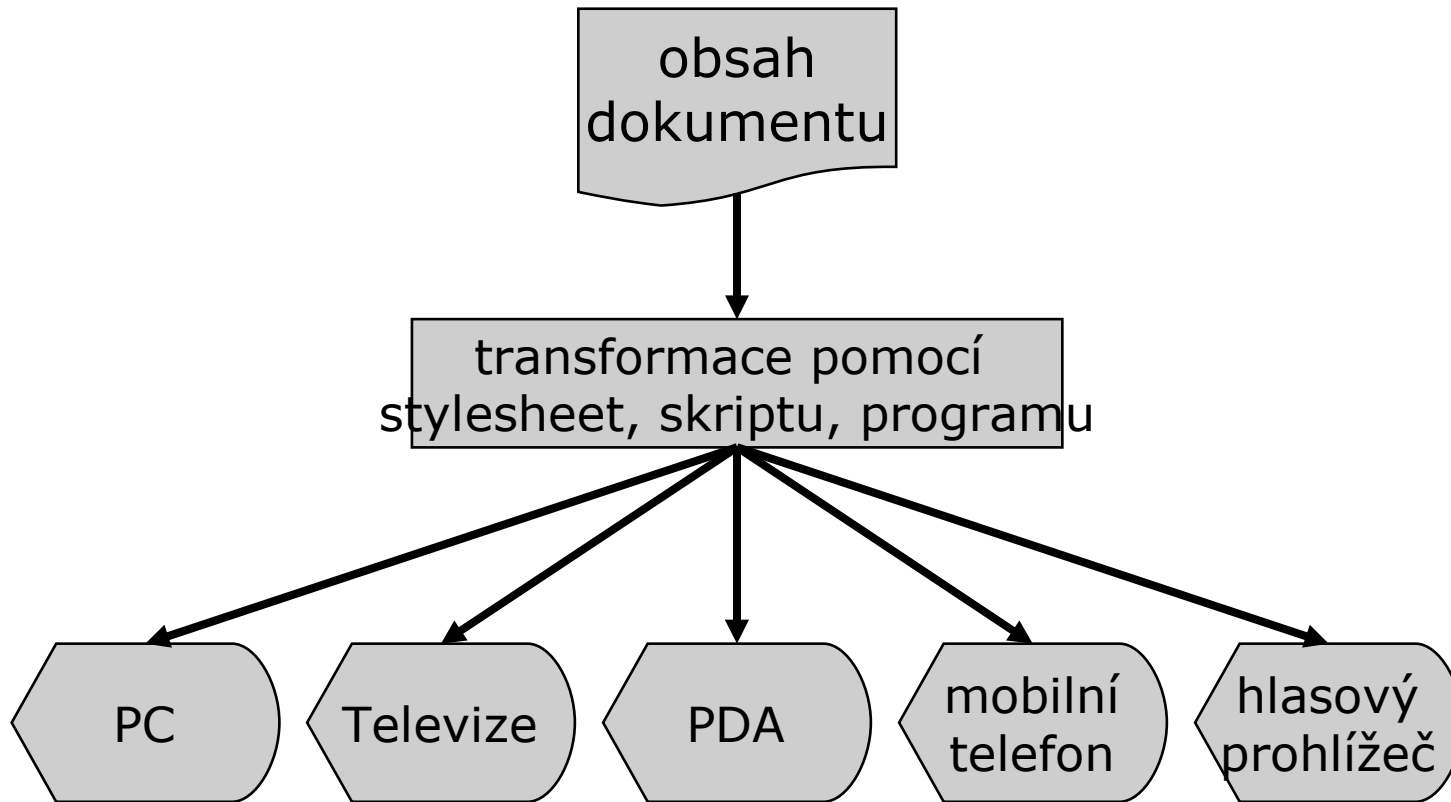


HTML 4 – klíčové vlastnosti

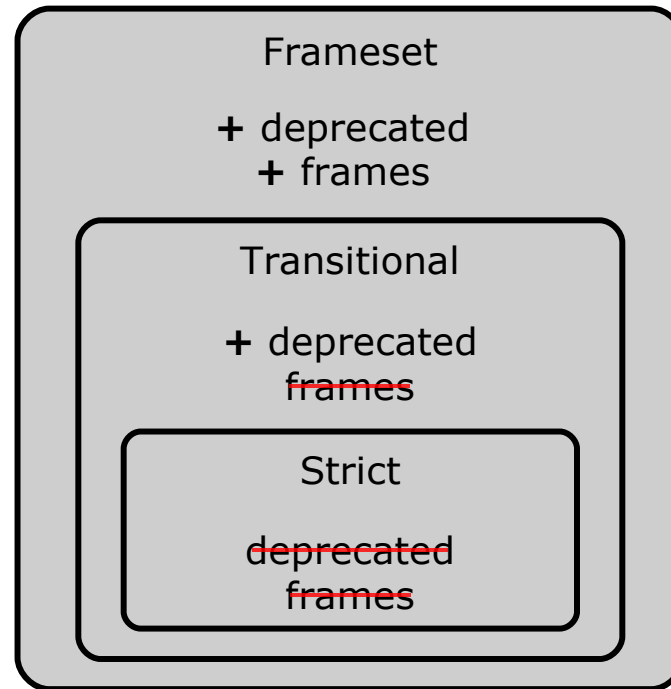
- oddělení obsahu od prezentace (CSS)
- vylepšení formulářů
 - accesskey, sémantické slučování řídicích prvků, labels, inkrementální načítání
- zlepšení přístupnosti ve speciálních situacích
 - alternativní prezentace dat (<object>)
 - horké klávesy (accesskey)
 - bohatší popisy elementů
 - popis a jazyk u všech elementů (title, lang)
 - podpora zkratk a akronymů (<abbr>, <acronym>)
 - tabulky: nadpisy, souhrnný popis
 - podrobný popis tabulek, obrázků a ráků <longdesc>



Modularizaci XHTML si vynutila různorodost aplikací



Struktura HTML dokumentu – HTML 4 verze

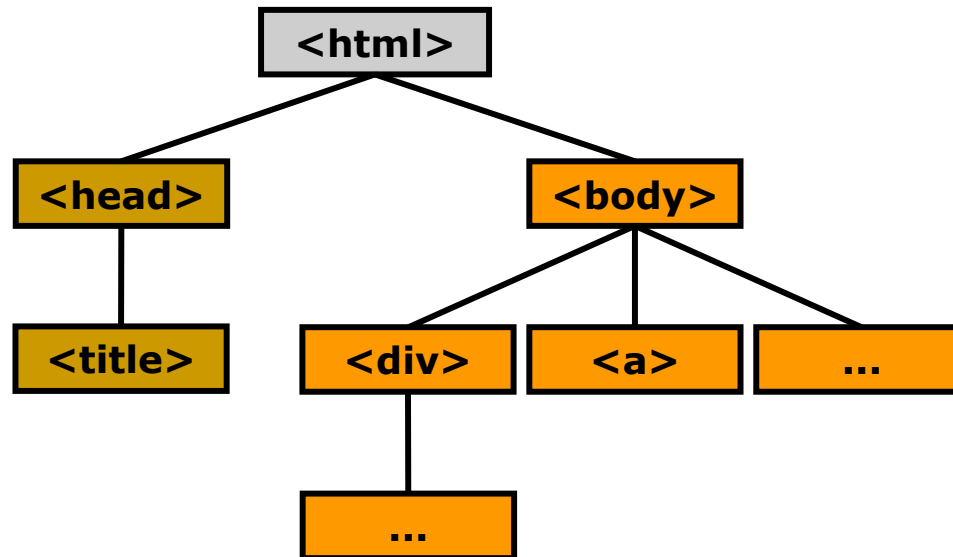


HTML – způsob zacházení

- Server poskytuje textovou formu HTML dokumentu
- Klient čte a „parsuje“ tento text
- Staví si DOM
 - DOM = Document Object Model
 - Reprezentace modelu HTML dokumentu v paměti
- DOM je následně vykreslen v grafické podobě
 - Rendering



Struktura HTML dokumentu – strom



Zpětná kompatibilita

- Problém s korektností HTML a se zpětnou kompatibilitou
 - Standard
 - Pseudo-standard
 - Quirk



Standard a Quirk mód

- Týká se moderních prohlížečů
 - IE ≥ 5.5 , FF, Netscape, Opera
- Prohlížeče hrají dvojí hru
 - Snaží se správně interpretovat historické HTML dokumenty včetně jejich chyb
 - Snaží se dodržovat standardy
 - \Rightarrow tyto dvě věci jsou v zásadě kontradikce
- Východisko: 2 (resp. 3) režimy činnosti
 - Standard = dodržuje specifikaci
 - Quirk = zpětně kompatibilní, nedodržuje specifikaci, snaží se nalézt východiska z problematických situací
- Standard – chceme, je rychlý a správný
- Quirk – nechceme, je pro špatné programátory, také je daleko častější

V čem se quirk a standard mód liší?

- BOX model v IE
 - známá chyba
 - jen v Quirk módu

```
<body>  
  <div>  
    Nějaký text  
  </div>  
</body>
```

HTML

```
div {  
  width: 100px;  
  padding: 10px;  
  border: 5px  
    solid black;  
  margin: 10px; }
```

CSS



Výsledek (box model)



Standard



IE Quirk mode

Quirk vs Standard

- Základem pro rozlišení je definice typu dokumentu
- Definicí by měl začínat každý HTML dokument



Document Type	Win IE6	Opera 7.0x	Opera 7.1x	Mozilla 1.0.1.	Safari 1.1.1	Mac IE5
HTML bez uvedení DOCTYPE a staré DTD až do verze 3.2 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">	quirk	quirk	quirk	quirk	quirk	quirk
Přechodné HTML 4.0 bez URL <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">	quirk	quirk	quirk	quirk	quirk	quirk
Přechodné HTML 4.0 s URL <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">	std	std	std	quirk	quirk	std
Striktní HTML 4.0 bez i s URL <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">	std	std	std	std	std	std
Přechodné HTML 4.01 bez URL <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">	quirk	quirk	quirk	quirk	quirk	quirk
Přechodné HTML 4.01 s URL <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">	std	std	std	pseudo std.	pseudo std.	std
Striktní HTML 4.01 bez URL <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN">	std	std	std	std	std	quirk
Striktní HTML 4.01 s URL <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">	std	std	std	std	std	std
Přechodné XHTML 1.0 bez XML deklarace kódování <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">	std	std	std	pseudo std.	pseudo std.	std
Přechodné XHTML 1.0 s XML deklarací kódování <?xml version="1.0" encoding="iso-8859-2"?> <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">	quirk	quirk	std	pseudo std.	pseudo std.	std
Striktní XHTML 1.0 bez XML deklarace kódování <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">	std	std	std	std	std	std
Striktní XHTML 1.0 s XML deklarací kódování <?xml version="1.0" encoding="iso-8859-2"?> <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">	quirk	quirk	std	std	std	std



Terminologie a definice HTML

- **element:** `<p id="odstavec1">Hallo world!</p>`
- **typ-element:** `p`
 - pevně definované v HTML-DTD
- **atribut:** `id="odstavec1"`
 - pevně definované v HTML-DTD
- **typ-atribut:** `id`
- **značka:** `<p>`, `</p>`



HTML základy

- Dokument je tvořen sadou značek
- Značky nejsou case sensitive
 - `<body>` je to samé jako `<BODY>`
- Značky píšeme **malými písmeny**
 - proč? Kvůli jednoduššímu přechodu na XHTML
 - pozor: XML obecně a tedy i XHTML je case sensitive
 - značky se nesmí křížit, tj. dokument je trom
- Dokument má pevnou minimální strukturu
 - Verze
 - Hlavička
 - Tělo



Struktura HTML dokumentu

HTML dokument

verze

```
<!-- 1. HTML VERSION INFORMATION -->  
<!DOCTYPE html PUBLIC "-//W3C//Dtd HTML 4.01//EN"  
    "http://www.w3.org/TR/html4/strict.dtd">
```

hlavička

```
<html>  
  <!-- 2. DECLARATIVE HEADER SECTION -->  
  <head>  
    <title>Struktura HTML dokumentu</title>  
  </head>
```

tělo

```
<!-- 3. BODY, CONTAINING THE DOCUMENT CONTENT -->  
<body>  
  Toto je vlastní obsah dokumentu.  
</body>
```

```
</html>
```



Struktura HTML dokumentu – členění těla

■ block-level (div), inline (span)

```
<div id="client-novak" class="client">  
<p><span class="client-title">Informace o klientovi:</span></p>  
<table border="" class="client-data" summary="Jméno, telefon a email">  
<tr><th>Příjmení:</th><td>Novák</td></tr>  
<tr><th>Jméno:</th><td>Josef</td></tr>  
<tr><th>Telefon:</th><td>224 222 123</td></tr>  
<tr><th>Email:</th><td>novakj1@fel.cvut.cz</td></tr>  
</table>  
</div>
```

```
<div id="client-Novotna" class="client">  
<p><span class="client-title">Informace o klientovi:</span></p>  
<table border="" class="client-data" summary="Jméno, telefon a email">  
<tr><th>Příjmení:</th><td>Novotná</td></tr>  
<tr><th>Jméno:</th><td>Jana</td></tr>  
<tr><th>Telefon:</th><td>234 444 222</td></tr>  
<tr><th>Email:</th><td>jana.novakova@vse.cz</td></tr>  
</table>  
</div>
```



HTML dokument - členění

- HTML by mělo definovat logickou strukturu dokumentu, nikoli grafickou
- Sada strukturálních elementů

h1 až h6	nadpisy
p	odstavec
<!-- komentář -->	komentář
hr	horizontální oddělovač
abbr	zkratka
acronym	akronym
address	adresa
bdo	směr textu
blockquote	dlouhé odsazení
q	krátké odsazení
cite	citace
dfn	definice termínu



HTML – typy elementů

■ Strukturální pokr.

code	text kódu počítače
kbd	text klávesnice
samp	ukázka počítačového kódu
tt	text dálnopisu
var	proměnná
pre	předformátovaný text
listing	výpis (dále nepodporováno viz pre)
plaintext	holý text (dále nepodporováno viz pre)
xmp	ukázka (dále nepodporováno viz pre)
strong	tučný text
sub	spodní index
sup	horní index
ins	vložený text
del	smazaný text



HTML typy elementů

■ Formátovací

b	tučný text
big	velký text
em	zvýrazněný text
i	kurzíva
small	malý text
s	smazaný text (nepodporováno, viz del)
strike	přeskrtnutý text (dále nepodporováno, viz del)
u	podtržený text (dále nepodporováno)



Psaní textů

Definice typu dokumentu

Nejjednodušší dokument typu Hello World

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
```

```
"http://www.w3.org/TR/html4/strict.dtd">
```

```
<html>
```

```
<head>
```

```
<title>První titulek</title>
```

Titulek okna prohlížeče

```
</head>
```

```
<body>
```

Text k zobrazení

```
Toto je můj první text.
```

```
Jak se vám líbí?
```

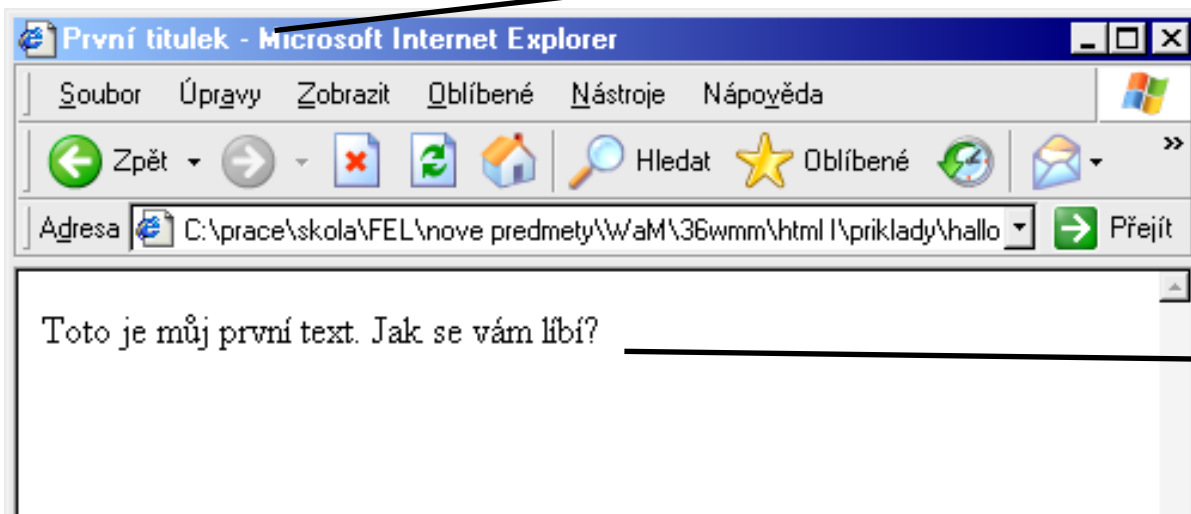
```
</body>
```

```
</html>
```

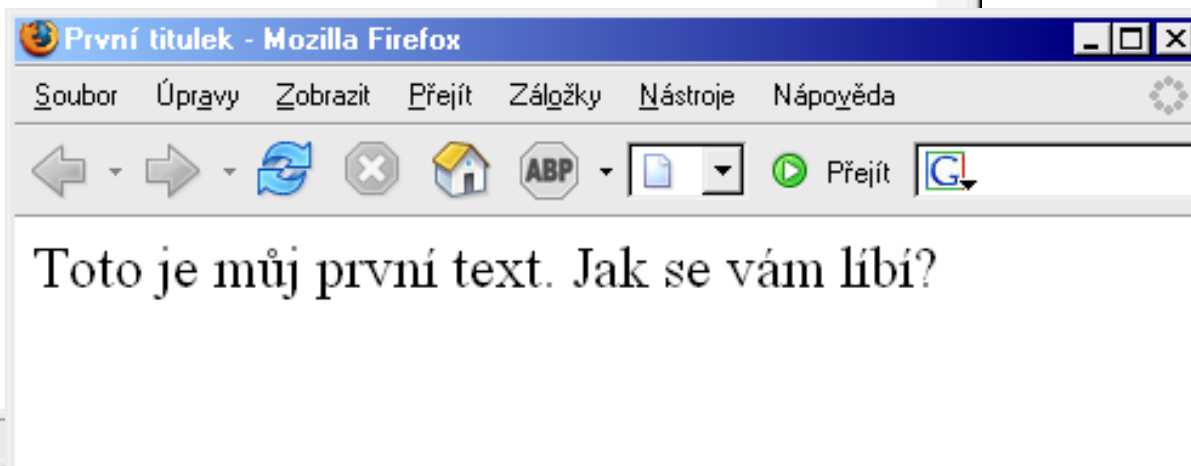


Výsledek – jak se to zobrazí

Titulek



Text



Texty – jak to dopadlo

- V HTML nebyla určena velikost ani jiné parametry písma
- V HTML nebyla určena struktura

<body>

Toto je můj první text.

Jak se vám líbí?

</body>

Toto není strukturovaný text

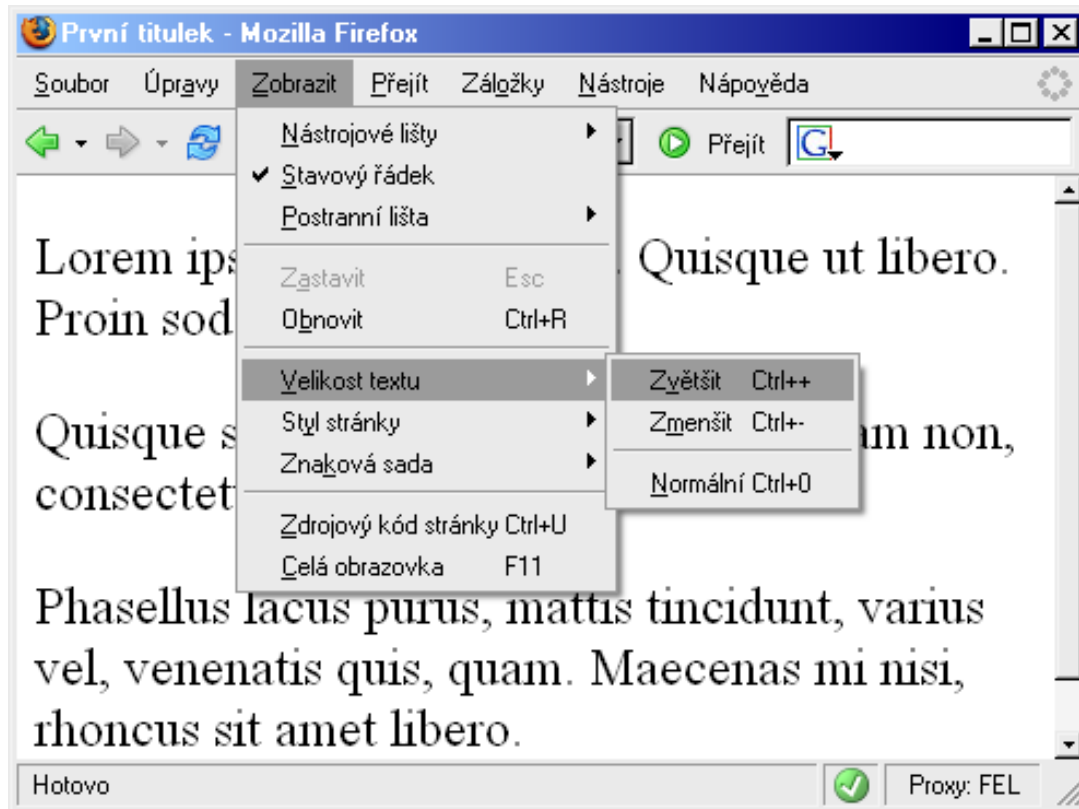
Nevýznamné mezery

- Více než jedna mezera je ignorována
- Velikost a styl (všeho) určuje prohlížeč, pokud není řečeno jinak



Texty – jak to dopadlo pokr.

- Velikost písma v prohlížeči mohou nastavovat podle svých preferencí



Texty- struktura - odstavec

`<body>`

`<p>`

Lorem ipsum dolor sit amet.

Quisque ut libero.

Proin sodales ipsum.

`</p>`

`<p>`

Quisque sagittis. Sed sapien

velit, aliquam non, consectetur non,

`</p>`

`<p>`

Phasellus lacus purus,

mattis tincidunt, varius vel,

venenatis quis, quam. Maecenas mi nisi,

rhoncus sit amet libero.

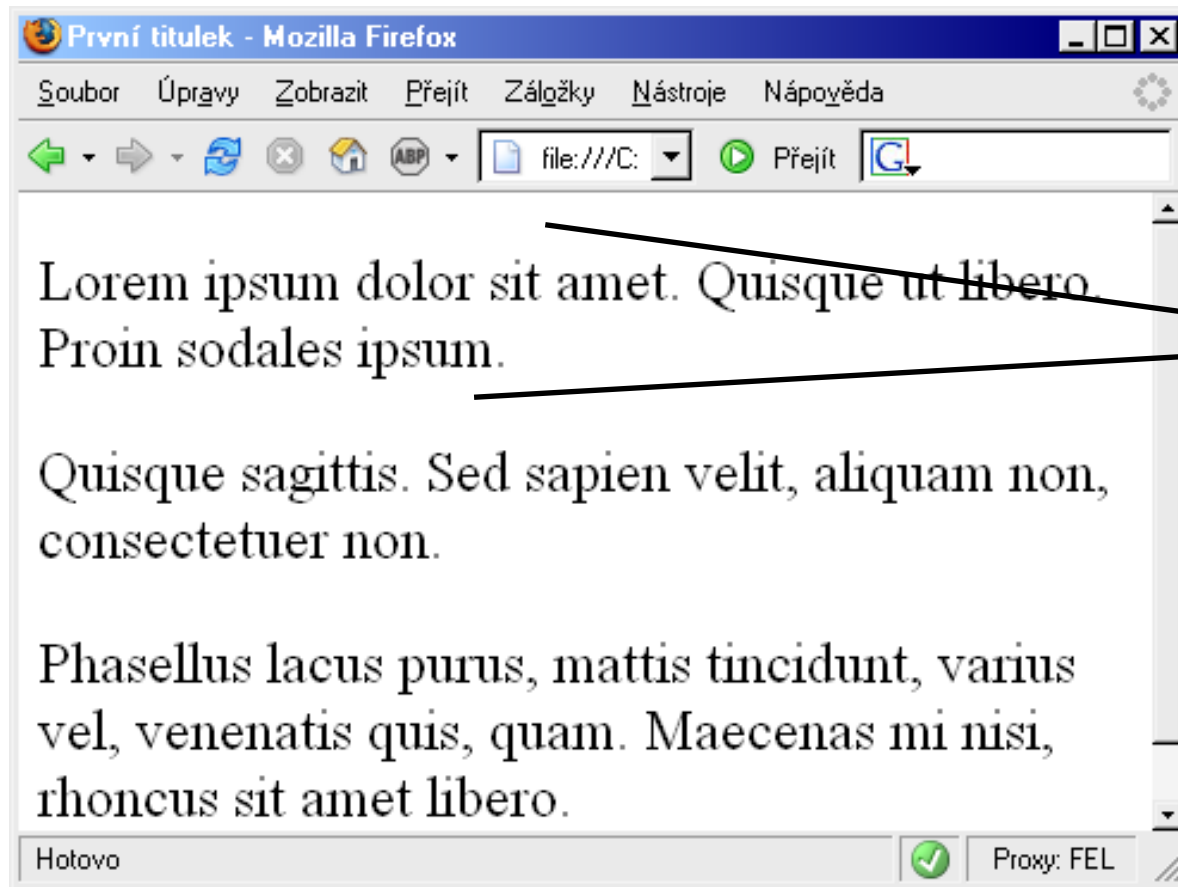
`</p>`

`</body>`

Nevýznamné mezery

Párová značka odstavce

Výsledek



Odsazení způsobené
elementem `<p></p>`



Texty - nadpisy

`<body>`

`<h1>Nadpis 1</h1>`

`<p>`

Lorem ipsum dolor sit amet.

Quisque ut libero.

Proin sodales ipsum.

`</p>`

`<h2>Nadpis 1.1</h2>`

`<p>`

Quisque sagittis. Sed sapien

velit, aliquam non, consectetur non,

`</p>`

`<h3>Nadpis 1.1.1</h3>`

`<p>`

Phasellus lacus purus,

mattis tincidunt, varius vel,

venenatis quis, quam. Maecenas mi nisi,

rhoncus sit amet libero.

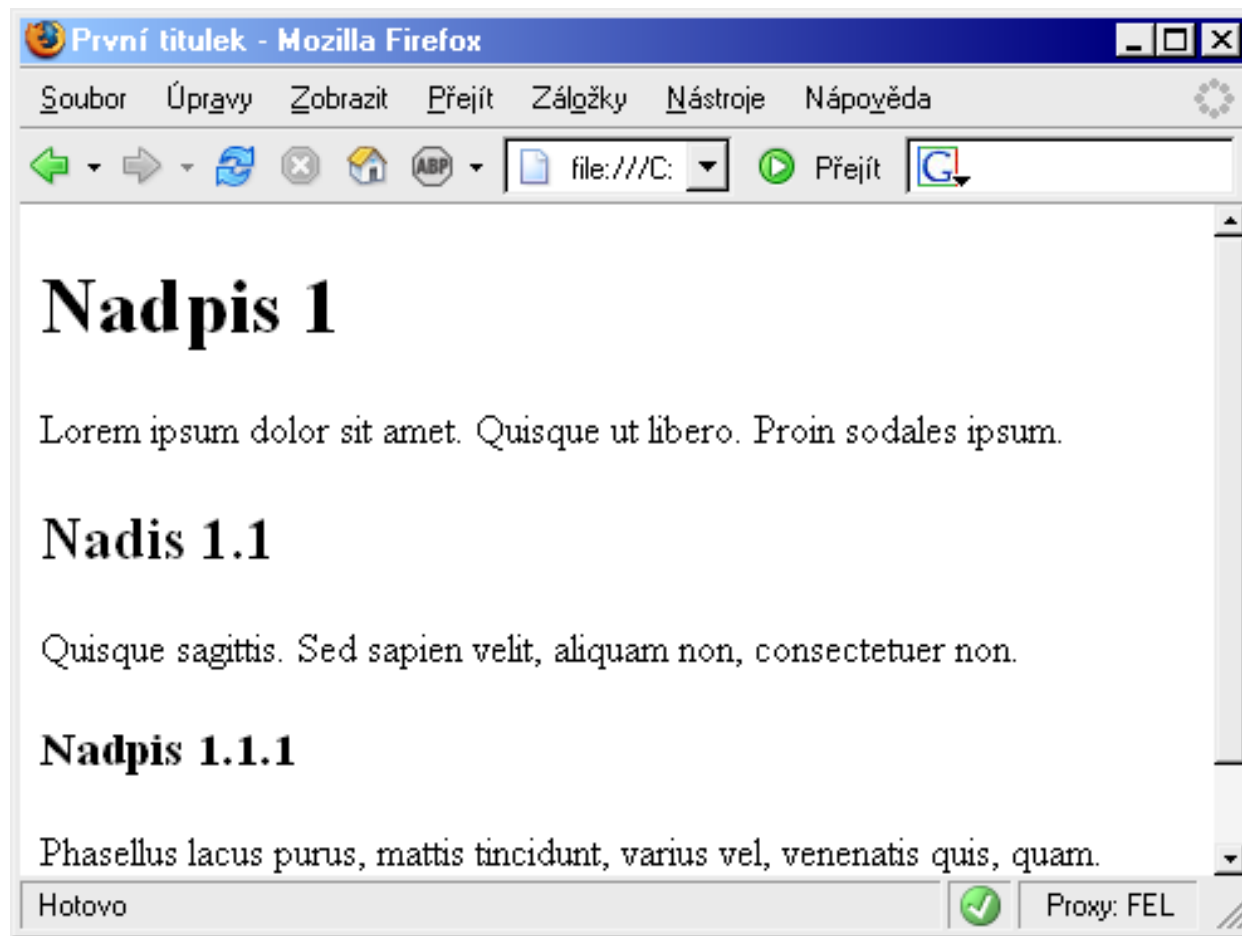
Nadpis 1. úrovně

Nadpis 2. úrovně

Nadpis 3. úrovně



Nadpisy výsledek

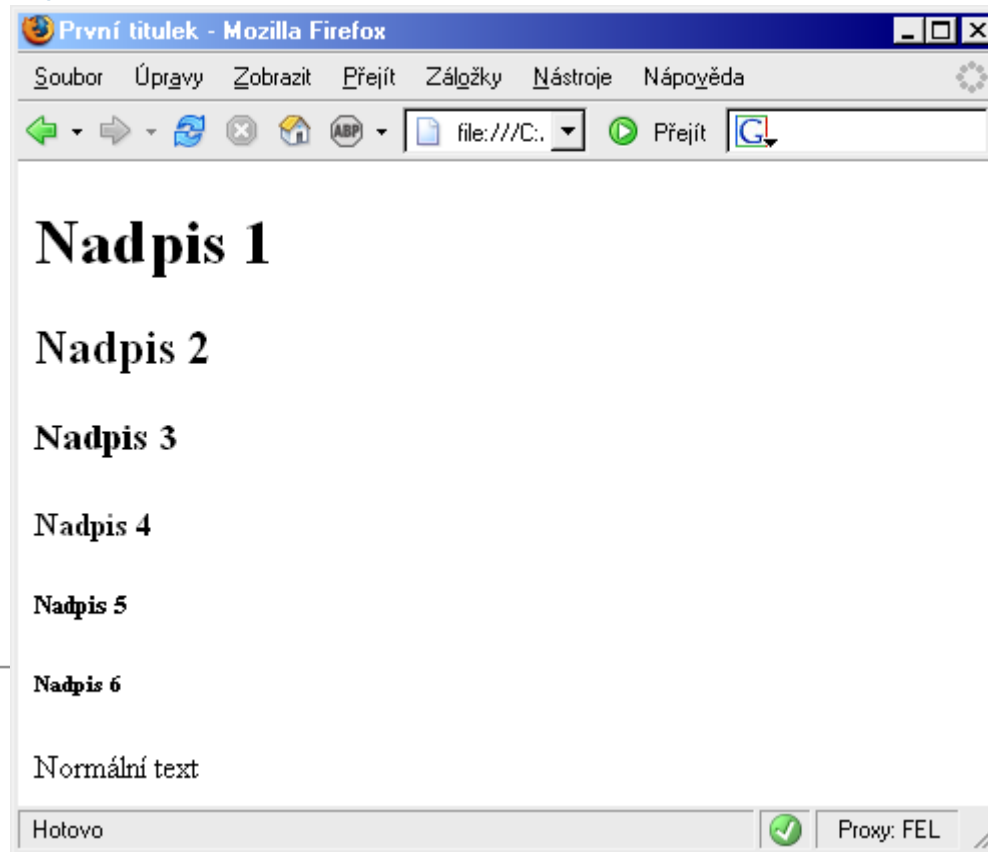


Nadpisy

Existuje 6 úrovní nadpisů

Nadpisy jsou velmi důležité, strukturují dokument a dávají mu význam

Jsou používány např. pro indexaci stránek roboty (Google)



Texty - zvýraznění

- Vzhledem k tomu, že HTML definuje jak formátovací, tak strukturální značky, svádí nás to je někdy zaměňovat

`<body>`

Lorem ipsum dolor sit amet.

``Quisque`` ut libero.

Proin sodales ipsum.

``Phasellus`` lacus purus,
mattis tincidunt, varius vel,
venenatis quis, quam. Maecenas mi nisi,
rhoncus sit amet libero.

`</body>`

Zvýraznění značkou

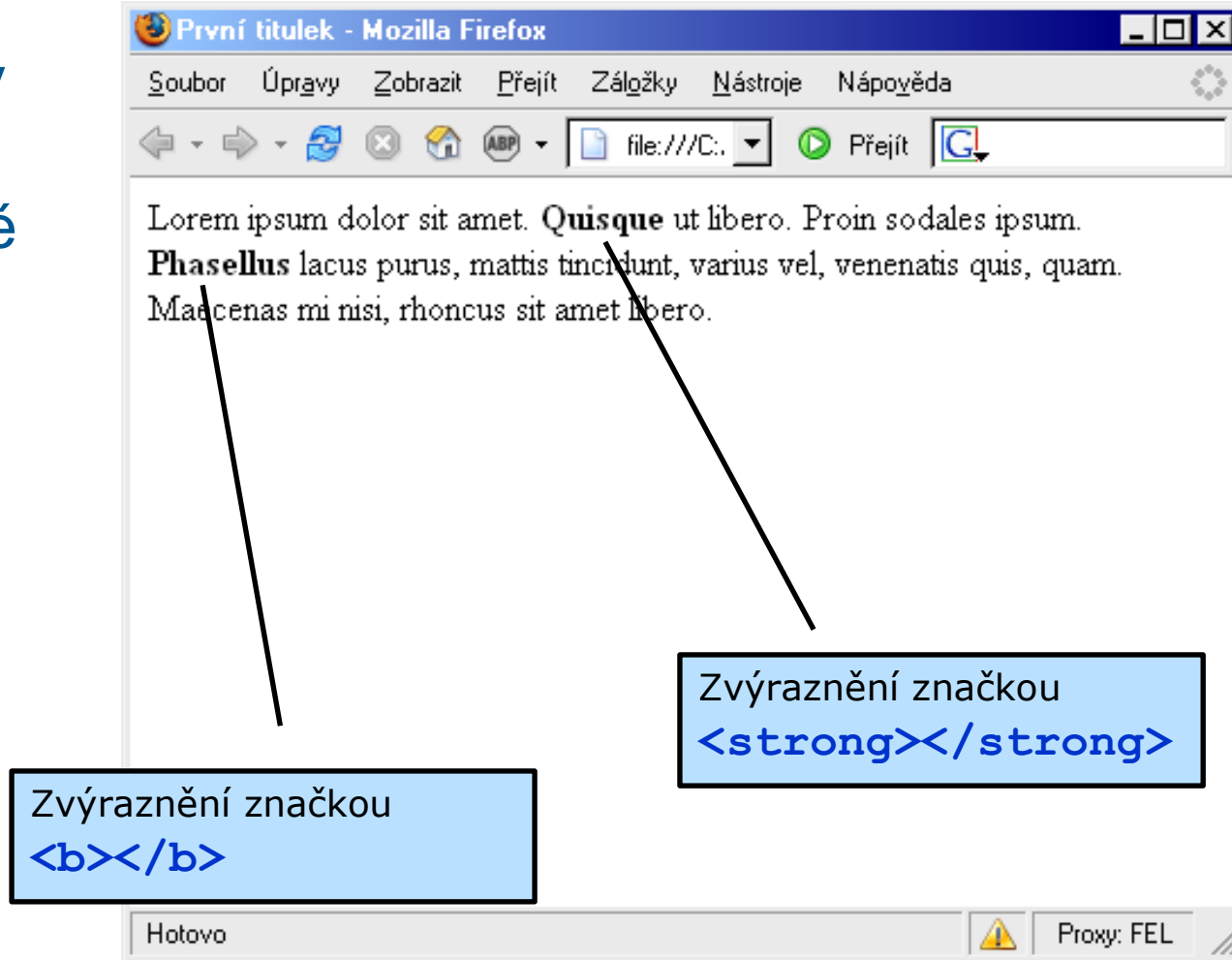
``

Zvýraznění značkou

``

Texty zvýraznění – jak to dopadlo

- Vizuálně dopadly oba způsoby zvýraznění stejně
- Fakticky je ale správně značka ``



Texty- další značky I

`<body>`

Lorem ipsum dolor sit amet.

`<abbr title="České vysoké učení technické">ČVUT</abbr>`

`<address>`

Technická 2`
`

Praha 6

`</address>`

ut libero.

Proin sodales ipsum `<acronym title="World Wide Web">WWW</acronym>`.

Phasellus lacus purus,
mattis tincidunt, varius vel,
venenatis quis, quam. Maecenas mi nisi,
rhoncus sit amet libero.

`</body>`

Zkratka

`<abbr></abbr>`

Adresa

`<address></address>`

Zalomení řádku

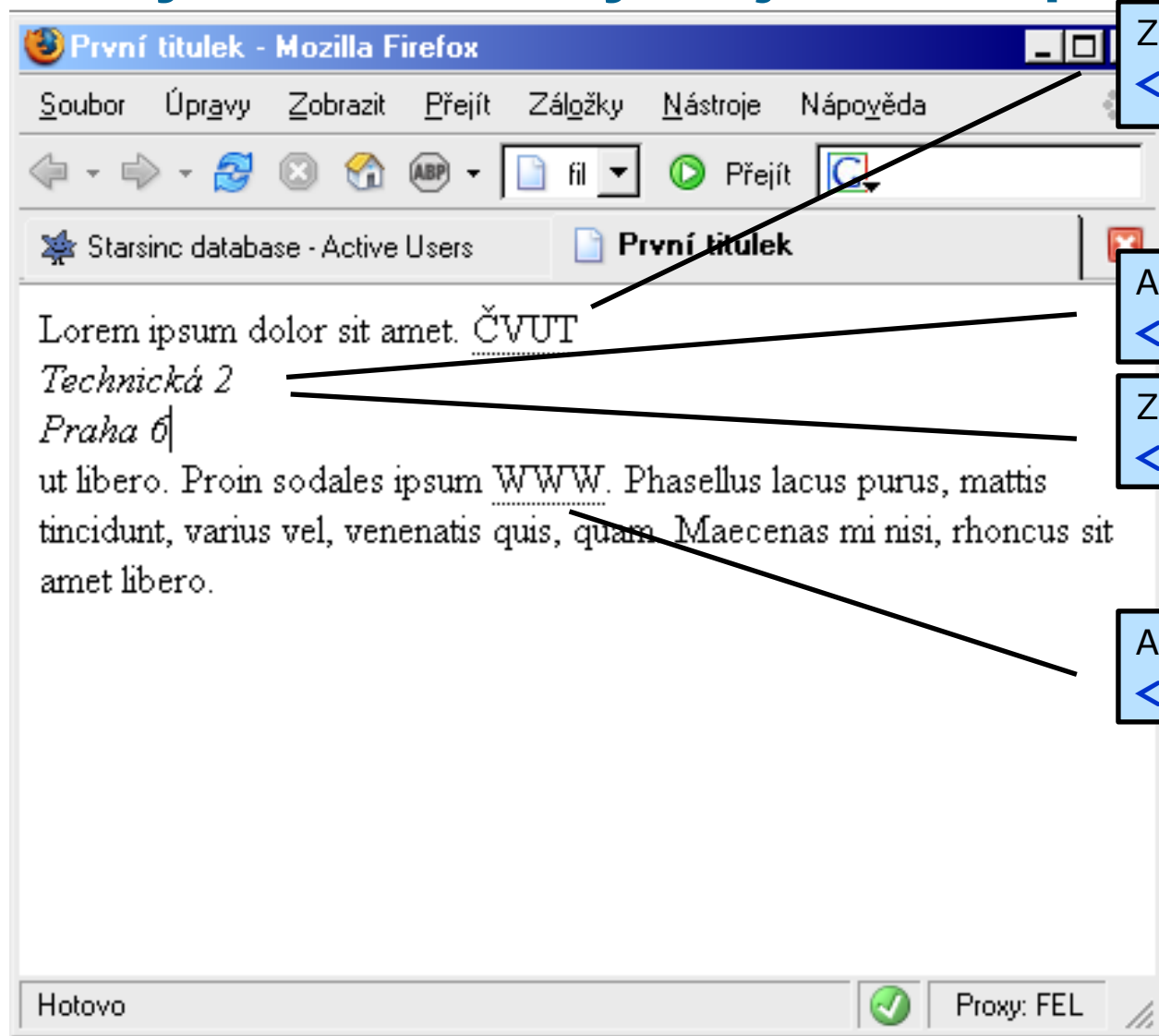
`
`

Akronym

`<acronym></acronym>`



Texty- další značky I - jak to dopadlo



Zkratka

`<abbr></abbr>`

Adresa

`<address></address>`

Zalomení řádku

`
`

Akronym

`<acronym></acronym>`



Texty- další značky II

<body>

Emphasized text

Strong text

<dfn>Definition term</dfn>

<code>Computer code text</code>

<samp>Sample computer code text</samp>

<kbd>Keyboard text</kbd>

<var>Variable</var>

<cite>Citation</cite>

</body>

Zdůraznění

Zvýraznění

Definice pojmu

<dfn></dfn>

Zdrojový kód

<code></code>

Ukázka

<samp></samp>

Text na klávesnici

<kbd></kbd>

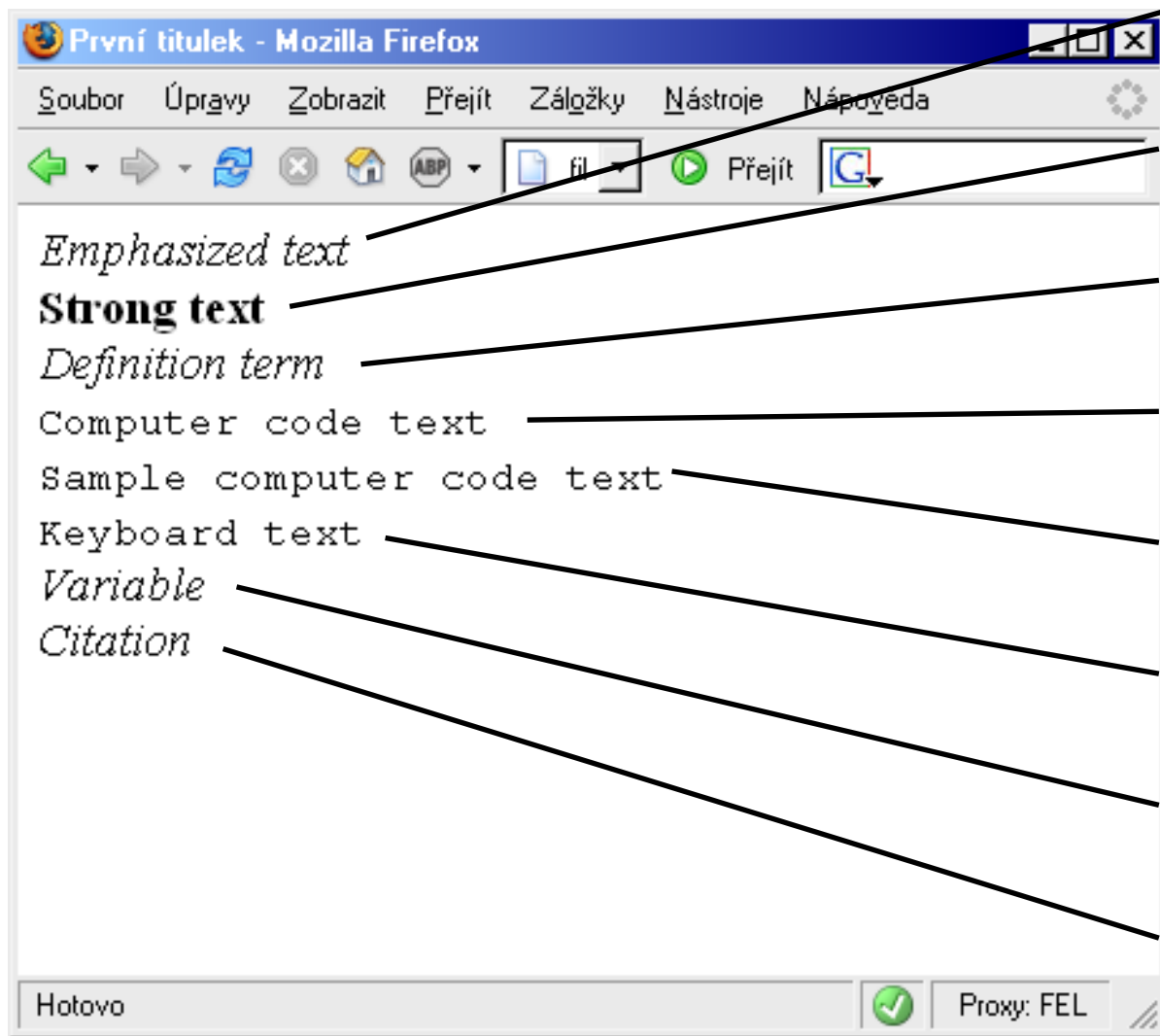
Proměnná

<var></var>

Citace

<cite></cite>





Zdůraznění

``

Zvýraznění

``

Definice pojmu

`<dfn></dfn>`

Zdrojový kód

`<code></code>`

Ukázka

`<samp></samp>`

Text na klávesnici

`<kbd></kbd>`

Proměnná

`<var></var>`

Citace

`<cite></cite>`



Computer Graphics Group



Odkazy (kotvy)

- Zroje na sebe mohou odkazovat
- Používají URL
- Připomenutí formátu URL:
protokol://host:číslo_portu/cesta/soubor
- V HTML dokumentu mohu
 - odkazovat na jiné zroje
 - definovat záložku
- Odkazy mohou být
 - absolutní: `http://www.seznam.cz`
 - relativní: `index.html`



Odkazy (kotvy)

Odkaz (absolutní) na seznam

```
<body>
```

```
<a href="http://www.seznam.cz">Seznam</a>
```

```
  Lorem ipsum dolor sit amet.
```

```
  Proin sodales ipsum.
```

```
  Phasellus lacus purus,
```

```
<a name="kotva1">mattisd</a> tincidunt, varius vel,  
  venenatis quis, quam. Maecenas mi nisi,  
  rhoncus sit amet libero.
```

```
<a href="#kotva1">vrat se na kotvu 1</a>
```

```
</body>
```

Definice kotvy se jménem
kotva1

Odkaz na kotvu v rámci
tohoto dokumentu



Odkazy/kotvy – další pokračování

- používat *id* nebo *name* ?
 - *id* slouží i k jiným účelům než jen k definici kotev
 - starší prohlížeče nemusí podporovat *id* kotvy

```
<a href="#a1">...</a>  
...  
<h1 id="a1">  
...  
<a name="a1"></a>
```

duplicita

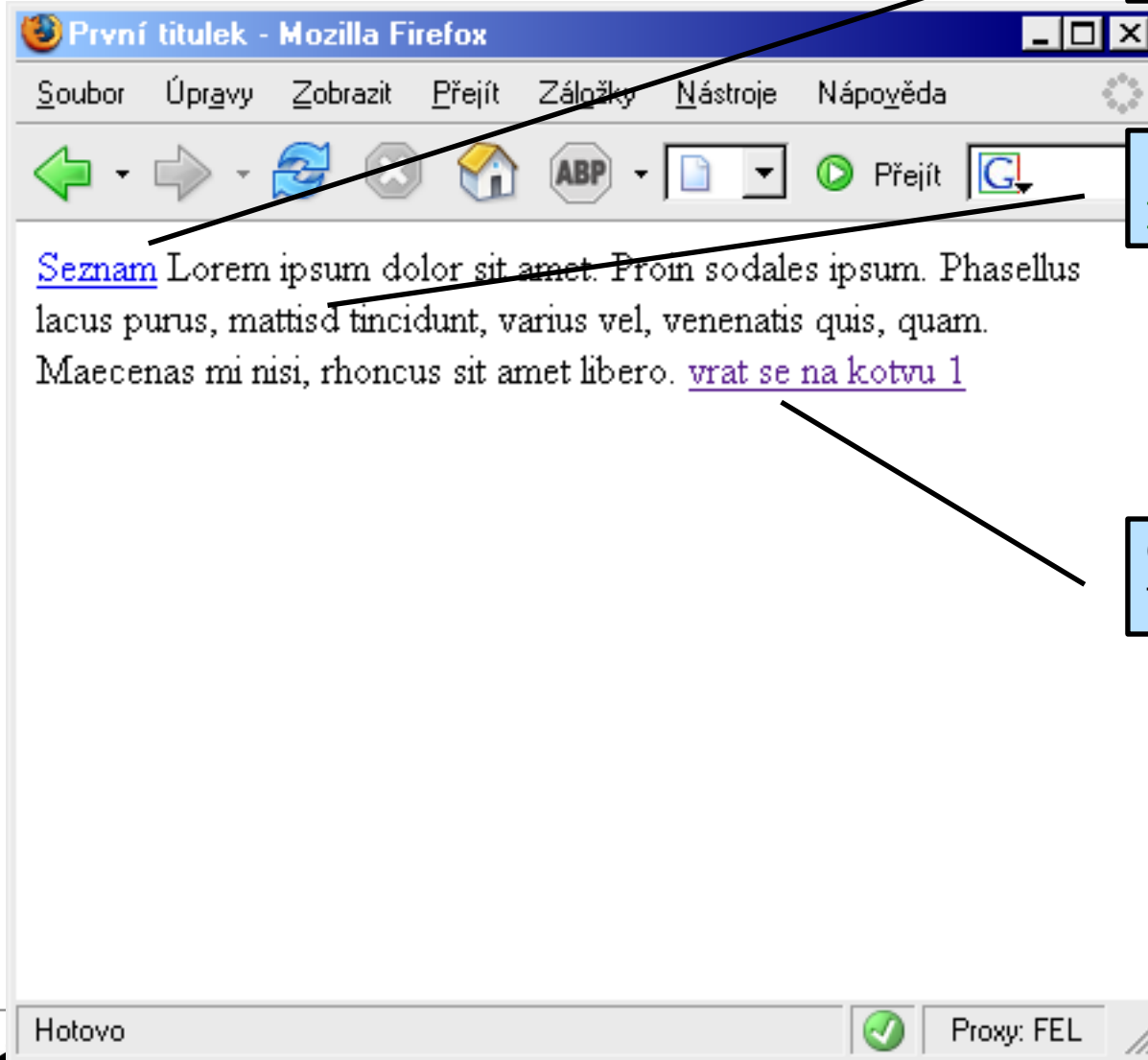


Odkazy (kotvy) jak to dopadlo

Odkaz (absolutní) na seznam

Definice kotvy se jménem **kotva1**

Odkaz na kotvu v rámci tohoto dokumentu



Odkazy - pokračování

```
<a href="http://www.seznam.cz" name="seznam"
target="_blank">Seznam</a>
```

Atribut name: jméno kotvy, lze na ní odkazovat odjinud pomocí notace **URL#jméno_kotvy**

Atribut target: udává, jak se má v prohlížeči odkaz otevřít.

Pozor, toto je možné jen ve verzích *Transitional* a *Frameset*

_blank otevře se v novém okně

_parent otevře se v rodičovském rámci

_self otevře se ve stejném rámci

_top otevře se přímo v těle prohlížeče (bez rámců)



URL

Tvar URL

`schema://host.domain:port/cesta/jmeno_souboru`

Dá se rozšířit na

`schema://host.domain:port/cesta/jmeno_souboru#jmeno_kotvy`

Dá se rozšířit na

`schema://host.domain:port/cesta/jmeno_souboru#jmeno_kotvy?
param1=hodnota1¶m2=hodnota2`



URL

Schémata

Schéma	Odkazuje na
file	soubor na lokálním PC
ftp	soubor na FTP serveru
http	soubor na WWW
gopher	soubor na Gopher serveru
news	usenet newsgroup
telnet	spojení Telnet
WAIS	soubor na WAIS serveru
mailto	odeslání e-mailu



Odkazy – běžné případy

Odkaz na běžnou WWW stránku

```
<a href="http://www.seznam.cz">Seznam</a>
```

Odkaz na ftp soubor

```
<a href="ftp://www.server.cz/ftp/winzip.exe">soubor</a>
```

Odkaz na newsgroup

```
<a href="news:alt.html">HTML Newsgroup</a>
```

Odkaz na mail

```
<a href="mailto:xklima@fel.cvut.cz">xklima mail</a>
```



Seznamy

- Seznamy jsou:
 - nečíslované
 - číslované
 - seznam termínů
- Existují základní formátování
 - ta jsou nicméně ovlivitelná především stylem (CSS)

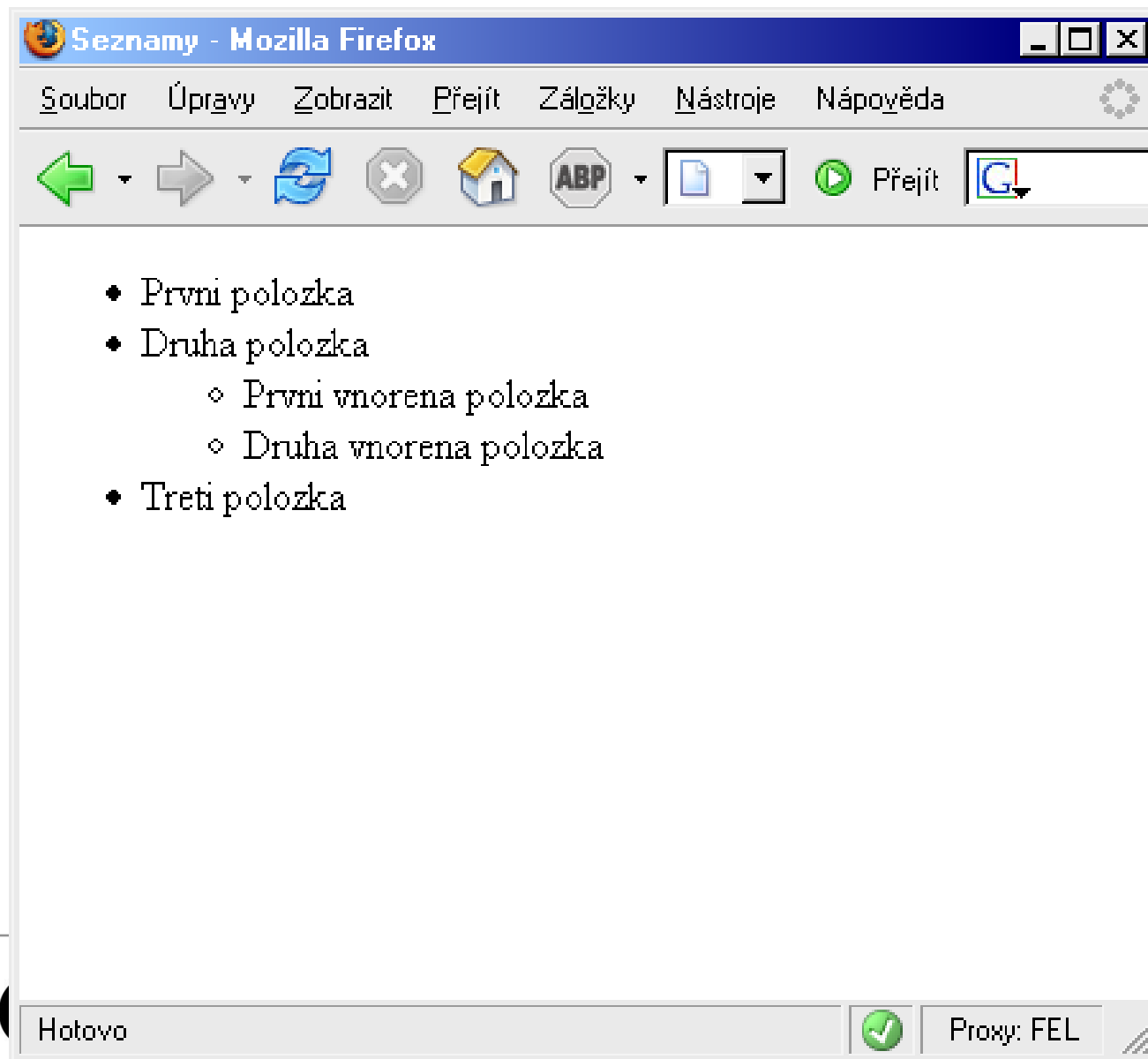


Seznamy

```
<body>
  <ul>
    <li>Prvni polozka</li>
    <li>Druha polozka
      <ul>
        <li>Prvni vnorena polozka</li>
        <li>Druha vnorena polozka</li>
      </ul>
    </li>
    <li>Treti polozka</li>
  </ul>
</body>
```



Seznamy – jak to dopadlo

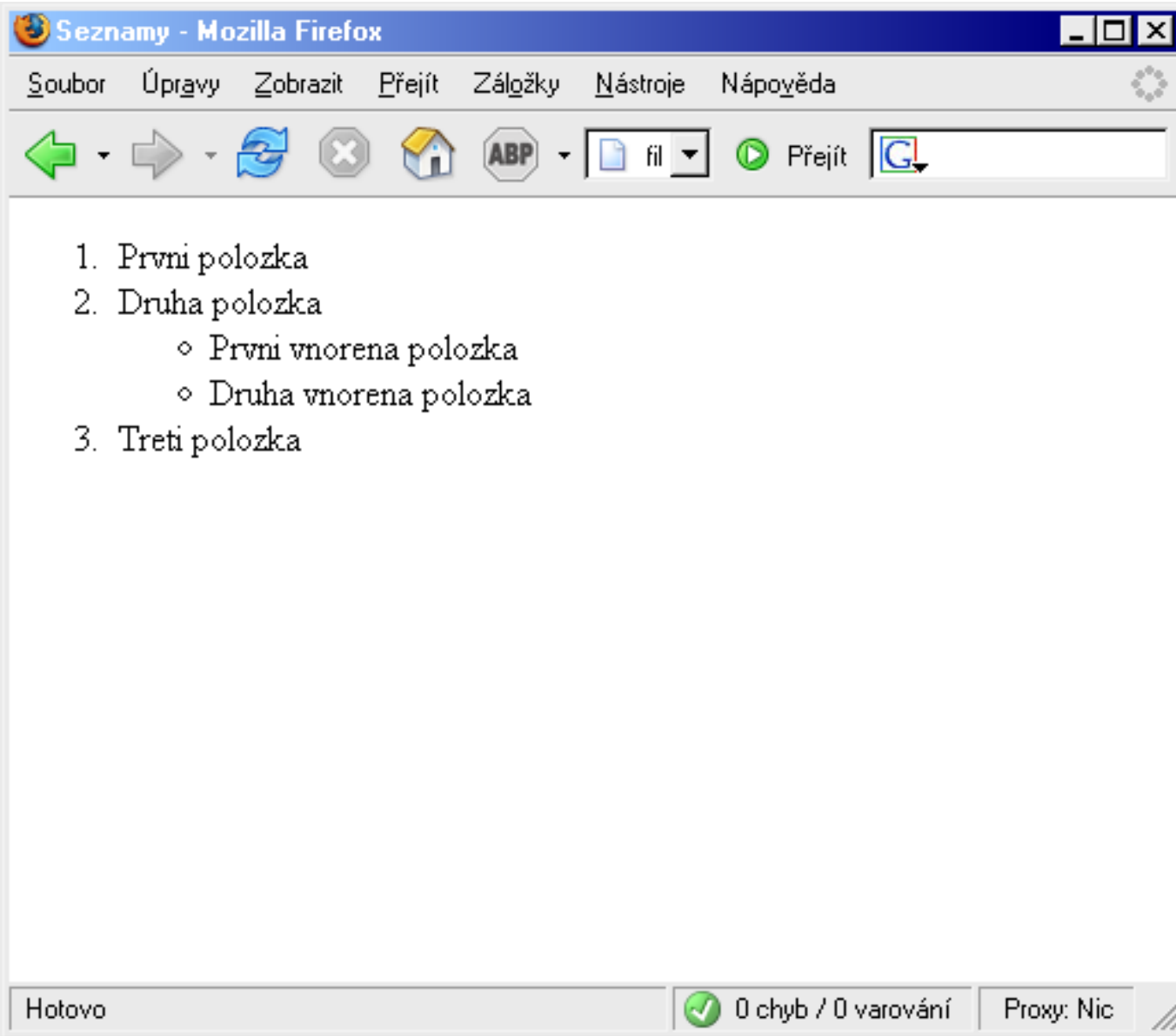


Číslovaný seznam

```
<body>
  <ol>
    <li>Prvni polozka</li>
    <li>Druha polozka
      <ul>
        <li>Prvni vnorena polozka</li>
        <li>Druha vnorena polozka</li>
      </ul>
    </li>
    <li>Treti polozka</li>
  </ol>
</body>
```



Číslovaný seznam – jak to dopadlo



Seznam pojmů

<body>

<dl>

<dt>URL</dt>

<dd>Uniform Resource Locator</dd>

<dt>WWW</dt>

<dd>World Wide Web</dd>

<dt>HTTP</dt>

<dd>Hypertext Transfer Protocol</dd>

<dt>Pojem</dt>

<dd>Definice</dd>

</dl>

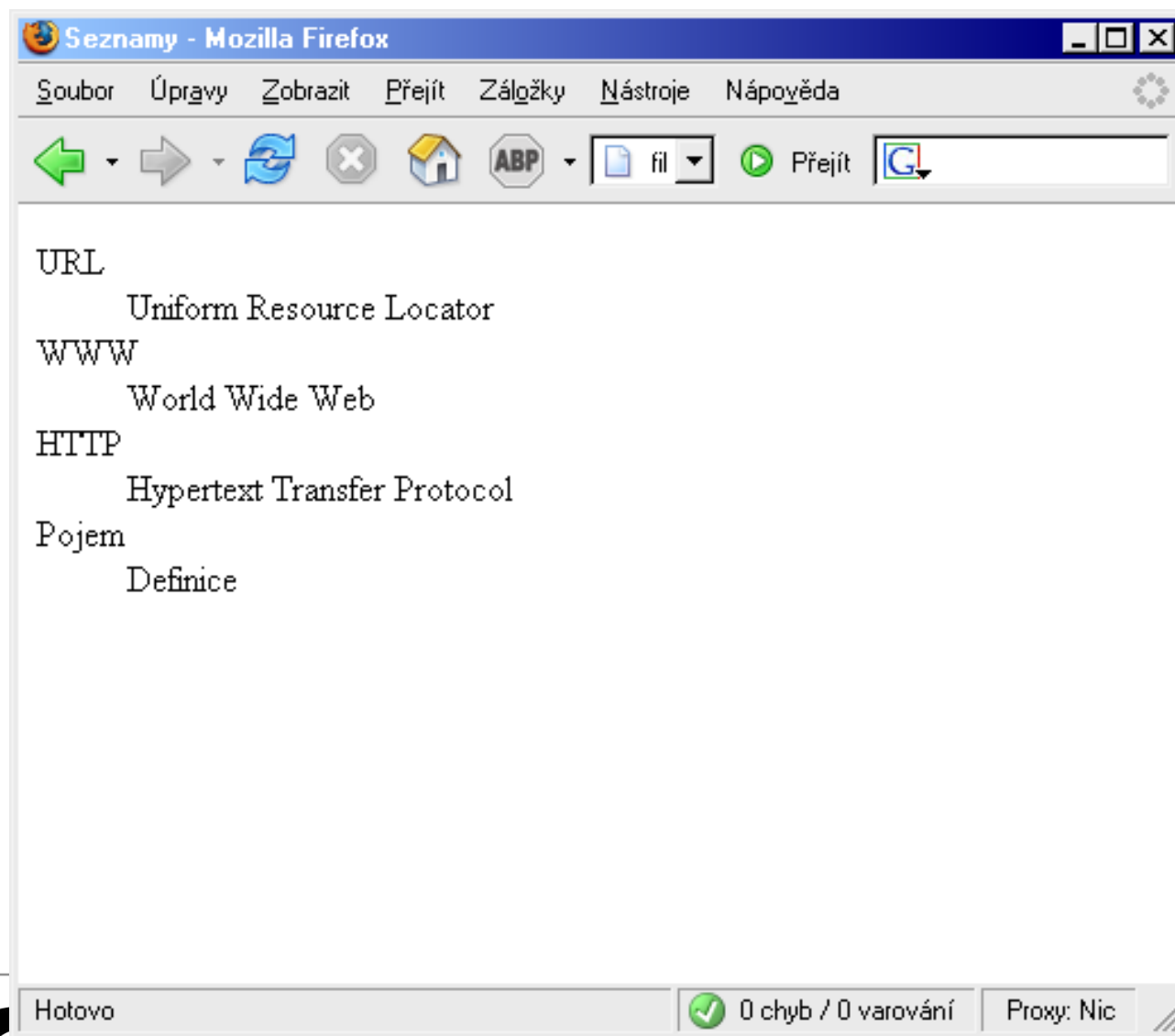
</body>

Termín

Definice termínu



Seznam pojmů – jak to dopadlo



Obrázky

- Běžně jsou podporovány obrázky typu
 - GIF
 - JPG
 - PNG (v IE v 6 nefunguje průhlednost)
- Každý obrázek musí mít textovou alternativu.
- Kde nestačí jednoduchý textový popis, je možné ukázat na jinou stránku, která obrázek popíše.
- Je dobré zadat rozměry obrázku.
- Stránka by neměla obsahovat velké obrázky.
 - Pokud je to nutné, uživatel musí mít možnost se jim vyhnout, například pomocí náhledů



Obrázky

```
<body>
  <h1>Moje zvířata</h1>
  
  [ <a href="obrazek_podrobne.html">D</a> ]
</body>
```

Zdroj obrázku

Alternativní text

Šířka

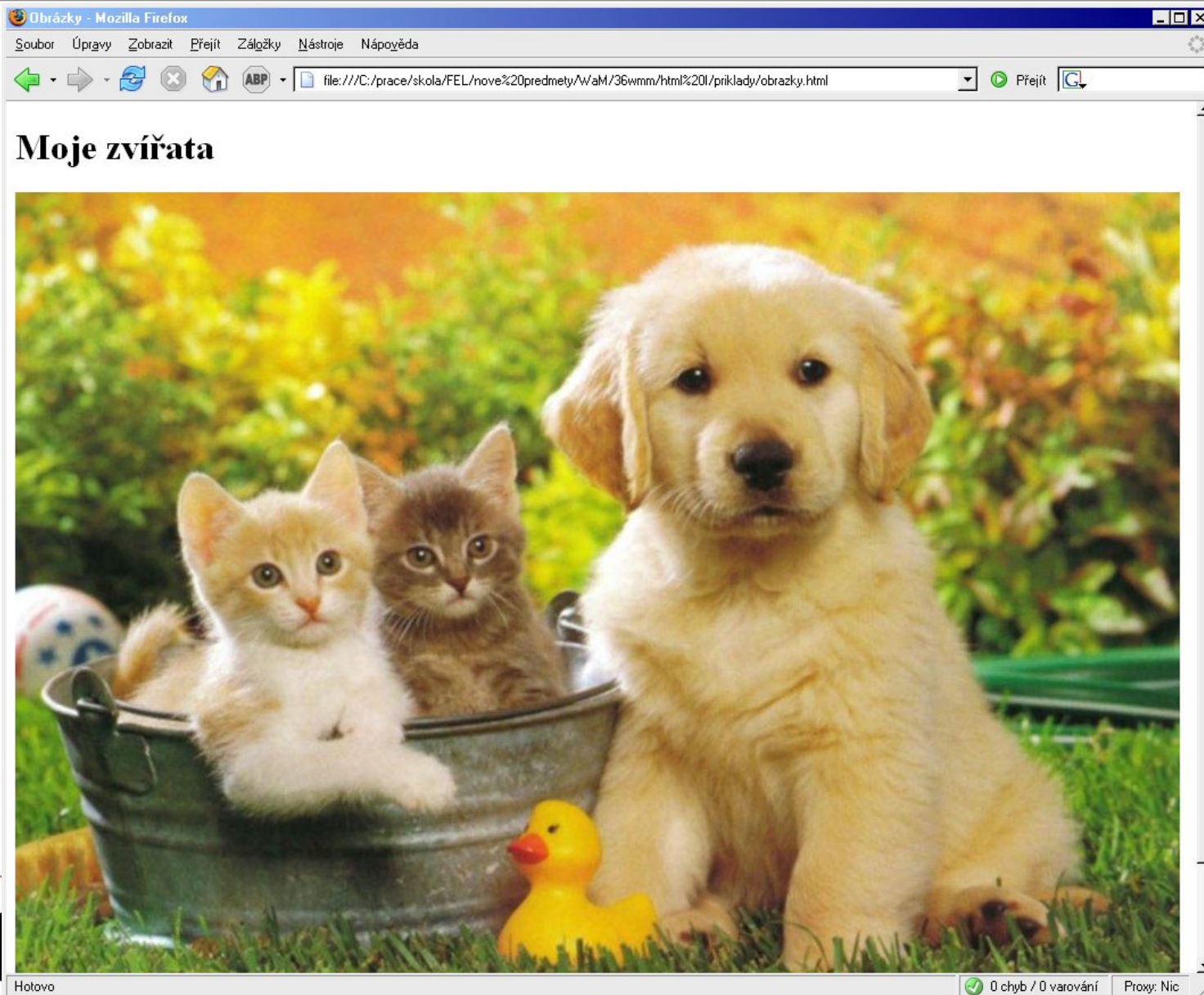
Výška

Odkaz na dlouhý popis

D-Link pro slepce



Obrázky - jak to dopadlo



Obrázky s náhledem

```
<body>
```

```
<h1>Moje zvířata</h1>
```

```
<h2>Takto ne</h2>
```

```

```

Celý velký obrázek byl stáhnut. Klienta jsme ale donutili ho zobrazit malý.
Špatně!

```
<h2>Takto ano</h2>
```

```

```

Správně, zobrazujeme malý náhled

```
<h2>Takto ano s odkazem</h2>
```

```
<a href="obrazky/obrazek1.jpg">
  
```

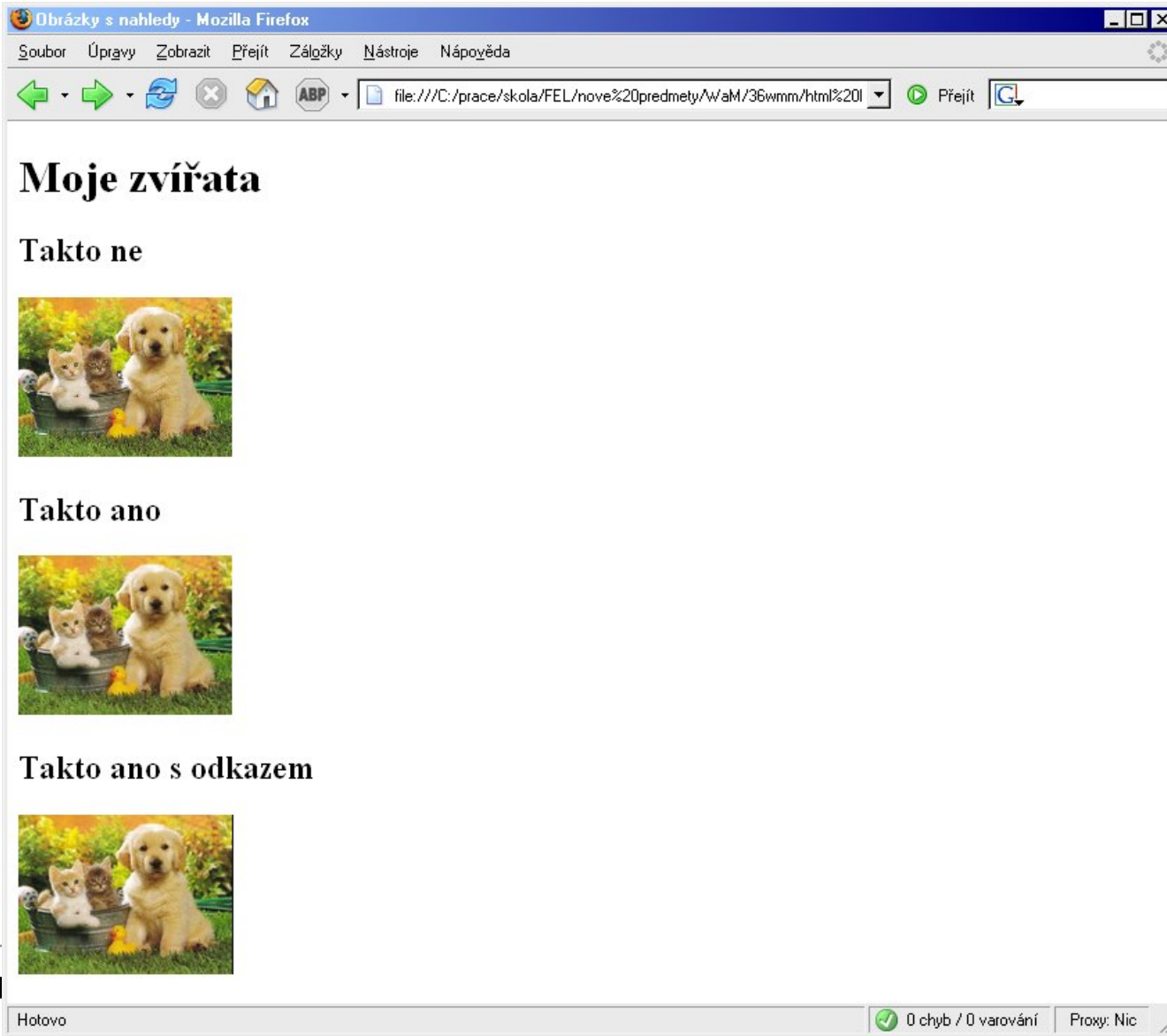
Náhled uzavřený v odkazu. Pozor, je dobré specifikovat atribut border, jinak se zobrazí rámeček

```
</a>
```

```
</body>
```



Obrázky s náhledem – jak to dopadlo



Alternativní zobrazení dat pomocí <object>

```
<!-- alternativní zobrazení dat pomocí vnorených elementu <object> -->
<div title="Ukázka alternativního zobrazení dat pomocí vnořených elementů">
  <h1>Učení neuronu typu Adaline</h1>
  <object title="Učení neuronu typu Adaline"
    codetype="application/java" classid="java:Neuron.class"
    data="http://www.cgg.cvut.cz/~cgge/Neuron.class" type="application/java">
    <!-- alternativně zobrazit obrázek GIF -->
    <object data="http://www.cgg.cvut.cz/~cgge/Neuron.gif" type="image/gif">
      <!-- alternativně zobrazit text -->
      Způsob učení neuronu typu Adaline můžeme demonstrovat na
    </object>
  </object>
</div>
```



Tabulky

Tabulky se sestávají z několika různých částí, ne všechny jsou povinné.

■ Tabulka

- Hlavičková část (THEAD)
- Patičková část (TFOOT)
- Hlavní část (TBODY)
- Popisek (CAPTION)

■ hlavičková, patičková a hlavní část se skládají z řádků tabulky a ty se skládají ze sloupců



Jednoduchá tabulka

	Leden Únor Březen		
Jablka	100	125	80
Hrušky	34	61	28
Švestky	21	12	9
Celkem	165	198	117

Jednoduchá tabulka

Atribut summary
popisuje tabulku

```
<body>
  <table summary="Přehled cen jablek, hrušek a švestek">
    <thead>
      <tr>
        <td>&nbsp;</td>
        <th>Leden</th>
        <th>Únor</th>
        <th>Březen</th>
      </tr>
    </thead>
    <tbody>
      <tr><th>Jablka</th><td>100</td><td>125</td><td>80</td></tr>
      <tr><th>Hrušky</th><td>34</td><td>61</td><td>28</td></tr>
      <tr><th>Švestky</th><td>21</td><td>12</td><td>9</td></tr>
    </tbody>
    <tfoot>
      <tr><th>Celkem</th><td>165</td><td>198</td><td>117</td></tr>
    </tfoot>
  </table>
</body>
```

Záhlaví

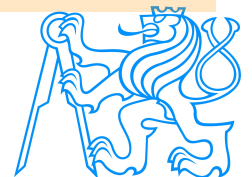
Řádek

Buňka obyčejná

Buňka záhlaví

Hlavní část tabulky

Patička



Tabulky – slučování buněk

	Leden Únor Březen		
Jablka	100	125	80
Hrušky	34	61	28
Švestky	21	12	9
Celkem	165	198	117
Celkem total	480		

Sloučená buňka
přes 3 sloupce

Slučování buňek

```
<body>
```

```
<table summary="Přehled cen jablek, hrušek a švestek">
```

```
<thead>
```

```
<tr>
```

```
<td>&nbsp;</td>
```

```
<th>Leden</th>
```

```
<th>Únor</th>
```

```
<th>Březen</th>
```

```
</tr>
```

```
</thead>
```

```
<tbody>
```

```
<tr><th>Jablka</th><td>100</td><td>125</td><td>80</td></tr>
```

```
<tr><th>Hrušky</th><td>34</td><td>61</td><td>28</td></tr>
```

```
<tr><th>Švestky</th><td>21</td><td>12</td><td>9</td></tr>
```

```
</tbody>
```

```
<tfoot>
```

```
<tr><th>Celkem</th><td>165</td><td>198</td><td>117</td></tr>
```

```
<tr><th>Celkem total</th><td colspan="3">480</td></tr>
```

```
</tfoot>
```

```
</table>
```

```
</body>
```

Jediná buňka s
atributem **colspan**



Formuláře

- Formuláře se používají k odesílání dat na server
 - Omezená sada formulářových polí
 - Není možné vyrábět nová
 - Je možné je do jisté míry "přiohnout" pomocí javascriptu
-
- Na jedné stránce může být více různých formulářů
 - Data z formulářů mohou být odeslána kamkoli
 - Dvě metody odesílání dat: GET, POST



Formuláře

```
<body>
```

```
<form
```

```
  action="http://amun.felk.cvut.cz/vypis_form.php"
```

```
  method="GET"
```

```
>
```

```
<input type="text" name="jmeno" >
```

```
<input type="submit" name="odeslat" value="Odeslat" >
```

```
</form>
```

```
</body>
```

Značka formuláře

Atribut **action** má hodnotu URL, na které se pošlou data

Atribut **method** má hodnotu GET nebo POST

Textové vstupní pole. Jméno pole je "jmeno"

Odesílací vstupní pole. Zobrazí se jako tlačítko.



Formuláře

Formulář - Mozilla Firefox

Soubor Úpravy Zobrazit Přejít Záložky Nástroje nápověda

← → ↺ × 🏠 ABP 📄 Přejít 🔍

Odeslat

Hotovo ✓ 0 chyb / 0 varování Proxy: Nic

Všechny request promenne - Mozilla Firefox

Soubor Úpravy Zobrazit Přejít Záložky Nástroje nápověda

← → ↺ × 🏠 ABP 📄 Přejít 🔍

POST

GET

jmeno	ahoj
odeslat	Odeslat

REQUEST

jmeno	string(4) "ahoj"
odeslat	string(7) "Odeslat"

Hotovo ⚠ 0 chyb / 3 va



Formulářová pole

■ input

text
password
submit
checkbox
radio
reset
file
hidden
image
button

■ textarea

■ button

■ select

■ label



Formulářová pole

- Všechna pole mohou mít atribut **name**
- Jména polí nemusí být jedinečná
- Všechna pole mohou mít hodnotu, atribut **value**

```
<input type="text" name="pole1" value="Martin">
```

```
<input type="text" name="pole1" value="Zdenek">
```

Co se
odešle?

- Řešení: pokud potřebuji odeslat více hodnot pod stejným jménem, musím z toho udělat pole

```
<input type="text" name="pole1[]" value="Martin">
```

```
<input type="text" name="pole1[]" value="Zdenek">
```

Odešle se
oboje.



Úspěšnost

- Pole může ale nemusí být "úspěšné" (successfull)
 - Úspěšné pole: hodnota je odeslána
 - Neúspěšné pole: hodnota není odeslána

```
<input type="text" name="pole1" readonly="readonly"  
value="Martin">
```

Úspěšné pole

```
<input type="text" name="pole2" disabled="disabled"  
value="Martin">
```

Neúspěšné pole



Úspěšnost

Tuzka

```
<input type="checkbox" name="polozka[]" value="tuzka">
```

Papir

```
<input type="checkbox" name="polozka[]" value="papir">
```

Guma

```
<input type="checkbox" name="polozka[]" value="guma">
```

Tuzka ☐ Papir ☒ Guma ☐

- Úspěšná položka je jen ta, která je zaškrtnutá
 - ostatní se na server nepropagují
 - více položek se stejným jménem, využijí pole

Úspěšnost

```
<div>Hlasování</div>  
<input type="submit" name="hlasovani" value="Pro">  
<input type="submit" name="hlasovani" value="Proti">
```

Hlasování

Pro

Proti

Úspěšné pole

- Jen jedno tlačítko bude úspěšné => mohu to využít pro hlasování
- Nenastane konflikt jmen



Úspěšnost – typ radio

```
<form action="http://amun.felk.cvut.cz/vypis_form.php" method="GET">  
  <div>Hlasování</div>  
  
  Pro <input type="radio" name="hlasovani" value="pro"><br>  
  Proti <input type="radio" name="hlasovani" value="proti"><br>  
  Nevim <input type="radio" name="hlasovani" value="nevim"><br>  
  
  <input type="submit" name="odeslat" value="Odeslat">  
</form>
```

■ Výběr 1 z N

– nebo také 0 z N (pokud neřekneme jinak)

■ Nenastává konflikt jmen

■ Pozor, je třeba řešit počáteční podmínku!

– pokud není řešeno jinak, není na počátku nic zaškrtnuto



Úspěšnost – typ radio

- Jedna nebo žádná položka je úspěšná
- Pokud chceme, aby byla úspěšná právě jedna, musíme na začátku jednu označit atributem
checked="checked"



Formuláře - file

- Používá se k odeslání souboru na server
- Nepředstavuje bezpečnostní díru
 - uživatel musí aktivně vybrat soubor, to nelze zařídit skriptem
- POZOR, formulář musí mít atribut
`enctype="multipart/form-data"`
- Zobrazí pole pro název souboru a výběrací dialog
- Je nutné ho odesílat metodou POST



Formuláře - file

```
<form
```

```
  action="http://amun.felk.cvut.cz/vypis_form.php"
```

```
    enctype="multipart/form-data"
```

```
    method="post"
```

```
>
```

```
  <div>Odešli soubor</div>
```

```
  <input type="file" name="soubor1">
```

```
  <input type="submit" name="odeslat" value="Odeslat">
```

```
</form>
```

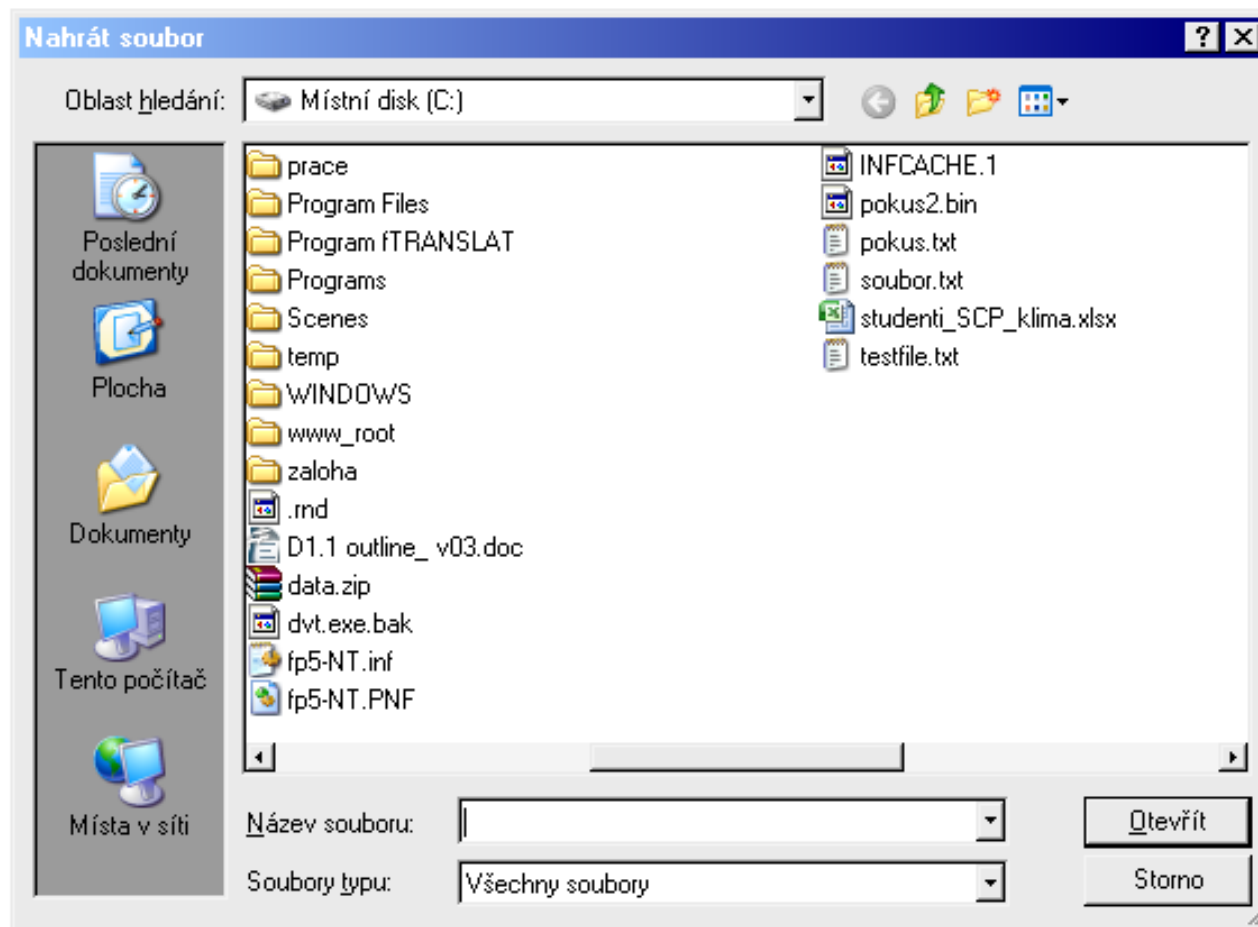


Formuláře file

Odešli soubor

Procházet...

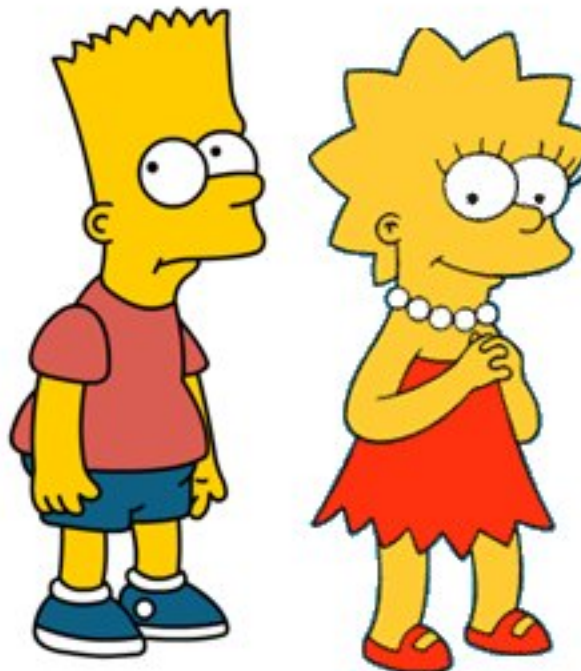
Odeslat



Formuláře - image

- Obrázek bude fungovat jako odesáací tlačítko
- Na server se odešle dvojice souřadnic
- Server je může dále zpracovat k uršení pozice např. na mapě

Vyber svého hrdinu



Formuláře – výběr ze seznamu

- Seznam položek
- Každá položka má text a může mít definovanou hodnotu
- Jednořádkový seznam je rozbalovací (dropdown)
- Víceřádkové seznamy se zobrazují s posuvníkem
- Výběr může být jedno či vícepoložkový (atribut **multiple**)
- Položky v seznamu se mohou sdružovat
- Explicitně vybrané položky jsou označeny atributem **selected="selected"**



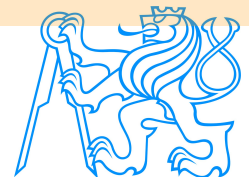
Formuláře – výběr ze seznamu

```
<form action="http://amun.felk.cvut.cz/vypis_form.php" method="post">
  <div>Vyber svého hrdinu</div>
  <select name="jeden_hrdina">
    <option value="1">Bart</option>
    <option value="2">Lisa</option>
    <option value="3">Homer</option>
    <option value="4">Marge</option>
  </select>
  <select name="vice_hrdinu[]" multiple="multiple">
    <optgroup label="Děti">
      <option value="1">Bart</option>
      <option value="2">Lisa</option>
    </optgroup>
    <optgroup label="Dospělí">
      <option value="3">Homer</option>
      <option value="4">Marge</option>
    </optgroup>
  </select>
  <input type="submit" name="odeslat" value="Odeslat">
</form>
```

Výběr 0-1 z N

Výběr 0-M z N

Skupina



Formuláře – výběr ze seznamu

Vyber svého hrdinu

Bart

Děti

Bart

Lisa

Dospělí

Homer

Marge

Odeslat

Formuláře - textarea

- Dlouhý text
- Pozor, je to párová značka

```
<form action="http://amun.felk.cvut.cz/vypis_form.php" method="post">  
  <div>Napiš svému hrninovi</div>  
  
  <textarea name="dlouhy_text">  
    Sem napiš co chceš :-)  
  </textarea>  
  
  <input type="submit" value="Odeslat" name="odeslat" >  
</form>
```



Formuláře textarea

Napiš svému hrninovi

Sem napiš co chceš
:-)

Odeslat



Formuláře – další náležitosti

- Každá (viditelná) formulářová položka musí mít přiřazený popisek
- Speciální značka `<label>`
- Popisek je s formulářovým polem svázán pomocí atributu `for`
- Formuláře mají být přístupné i bez myši
- atribut `tabindex` definuje pořadí při přístupu přes klávesu TAB
- atribut `accesskey` přiřazuje přístupovou klávesu



Formuláře – další náležitosti

```
<form      action="http://amun.felk.cvut.cz/vypis_form.php"  method="post">
<div>
    <label  for="jmeno">Jméno</label>
    <input  tabindex="1"  accesskey="j"  type="text"
name="jmeno">
</div>
<div>
    <label  for="prijmeni">Příjmení</label>
    <input  tabindex="2"  accesskey="p"  type="text"
name="prijmeni">
</div>

<input  tabindex="3"  accesskey="k"  type="submit"
value="Odeslat"  name="odeslat"  >
</form>
```



Formuláře – další náležitosti

Jméno

Příjmení

Odeslat



Formuláře – další náležitosti

- Formulářová pole je vhodné členit do skupin
- Značka **fieldset** a **legend**



Formuláře – další náležitosti

```
<form action="http://amun.felk.cvut.cz/vypis_form.php" method="post">
<fieldset>
  <legend>Jméno</legend>
  <label for="jmeno">Jméno</label>
  <input tabindex="1" accesskey="j" type="text" name="jmeno">

  <label for="prijmeni">Příjmení</label>
  <input tabindex="2" accesskey="p" type="text" name="prijmeni">
</fieldset>
<fieldset>
  <legend>Adresa</legend>
  <label for="ulice">Ulice</label>
  <input tabindex="3" accesskey="u" type="text" name="ulice">
  <label for="mesto">Město</label>
  <input tabindex="4" accesskey="m" type="text" name="mesto">
</fieldset>

<input tabindex="5" accesskey="k" type="submit" value="Odeslat" name="odeslat" >
</form>
```



Formuláře – další náležitosti

Jméno

Jméno

Příjmení

Adresa

Ulice

Město

Odeslat