

HTML

Začínáme tvořit webové stránky

Tomáš Vondráček

 [linkedin.com/in/vondracektomas](https://www.linkedin.com/in/vondracektomas)

 tomasvondrac@gmail.com

14. října 2024

Zpracováno pro účely výuky programování v jazyce JavaScript



OBSAH

1. Úvod do HTML
2. Psaní HTML kódu
3. Párové značky
4. Atributy
5. Párové značky s atributy
6. Nepárové značky
7. Co dalšího se může hodit

CÍLE

- Co to je HTML a jak se liší od CSS a JS
- Jak se píše HTML kód a jak se pomocí něj tvoří webová stránka
- Schopnost vytvořit první webovou stránku pomocí HTML

Úvod do HTML

CO TO JE HTML A K ČEMU SLOUŽÍ

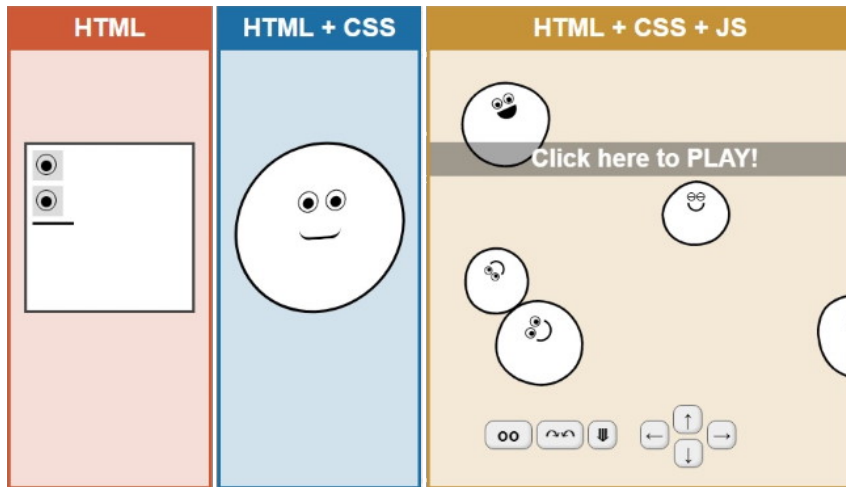
HTML (**H**yper**T**ext **M**arkup **L**anguage) je značkovací jazyk pro tvorbu webových aplikací.

- **H**yper**T**ext značí odkaz (neboli text, který odkazuje na jiné místo a tím ho propojuje)
- **M**arkup **L**anguage říká, že kód je zapisován pomocí značek (angl. tags)

HTML kód je tedy tvořen značkami, které říkají webovému prohlížeči:

- co všechno stránka obsahuje (webový prohlížeč zpracovává HTML kód shora dolů)
- a jak má daný text prohlížeč zobrazit (např. nadpis, odkaz na jiný web, formulář, ...)

JAK SPOLU SOUVISÍ HTML, CSS A JS



Obrázek 1: Obrázek převzat z <https://html-css-js.com>.

VÝVOJ VERZÍ HTML

Minulost

- První verze HTML 1.0 se objevila v roce 1991 a první prohlížeč o rok později
- Až v roce 1996 byly do HTML přidány tabulky (HTML 3.2) a vznikl Internet Explorer
- V roce 1997 byla přidána podpora pro CSS a JS (HTML 4.0)

Aktuálně

- Od roku 2014 se používá verze HTML 5.0, která přinesla sémantické webové značky, podporu multimédií, práci s grafikou, ...

Psaní HTML kódu

HTML SE NEPROGRAMUJE, ALE KÓDUJE (PÍŠE)



JAK TVOŘIT WEBOVOU STRÁNKU

1. Spustíme Visual Studio a vytvoříme soubor s příponou `.html`, např. `web.html`
2. Soubor `web.html` otevřeme v jakémkoliv webovém prohlížeči (z počátku bude naše webová stránka prázdná/bílá)
3. Ve Visual Studio začneme do souboru `web.html` psát HTML kód
4. Aby se změny v HTML kódu, které provedeme, zobrazily v prohlížeči, musíme vždy¹:
 - 4.1 uložit soubor (Ctrl+S)
 - 4.2 obnovit prohlížeč (F5)²

¹ Ruční obnovování prohlížeče lze ve Visual Studio automatizovat rozšířením Live Server, které si nainstalujeme.

² Pokud v HTML máme více souborů a změny se po obnovení prohlížeče neukazují, mohou být uloženy v paměti cache, a potom pomůže tzv. „force refresh“ (Ctrl+F5), při kterém prohlížeč načte data pro danou stránku znovu.

HTML ZNAČKY (TAGY)

HTML je značkovací jazyk, který definuje 113–142 značek³. **Smyslem značek je vyznačování textu nebo přidávání dalších souborů do webové stránky (obrázky, videa, ...).**

- Značka je tvořena špičatými závorkami `<>` obsahující **anglické slovo** či **zkratku**
- Značky jsou dvojího druhu: **párové** a **nepárové**

Párové značky:

`<u>podtržený text</u>`

`tučný text`

Nepárové značky:

``

`
`

³<https://devpractical.com/how-many-html-tags>

Párové značky

PÁROVÉ ZNAČKY (ANGL. PAIRED TAGS)

Párové značky se skládají z **otevírací** a **uzavírací** značky, které předchází lomítko /.

- Text, který chceme ovlivnit příslušnou značkou, napíšeme mezi otevírací a uzavírací značku.

HTML kód:

```
<u>underline (podtržený) text</u>  
<b>bold (tučný) text</b>  
<i>italic (zkosený) text</i>  
<del>deleted (přeškrtnutý) text</del>  
text<sup>superscript (horní index)</sup>  
text<sub>subscript (dolní index)</sub>
```

Výsledek v prohlížeči:

underline (podtržený) text
bold (tučný) text
italic (zkosený) text
~~deleted (přeškrtnutý) text~~
text^{superscript (horní index)}
text_{subscript (dolní index)}

PÁROVÉ ZNAČKY LZE DO SEBE ZANOŘOVAT

HTML kód:

```
<i>zkosený<del>přeškrtnutý</del>text</i>  
<b><u>podtržený</u>tučný text</b>
```

Výsledek v prohlížeči:

zkosený přeškrtnutý text
podtržený tučný text

POZOR ALE na správné pořadí⁴ **otevíracích a **uzavíracích** značek!**

```
<i>zkosený<del>přeškrtnutý</i>text</del>  
<u><b>podtržený</u>tučný text</b>
```

⁴I když se pořadí splete, tak se prohlížeče snaží udělat maximum pro to, aby výsledek odpovídal tomu, co jsme chtěli udělat. Vždy však chceme mít zápis správný, abychom předešli potenciálním chybám.

POSLEDNÍ VĚC, KTEROU MUSÍME VŽDY NAPSAT NA ZAČÁTEK .HTML SOUBORU

```
<!DOCTYPE html>
```

`<!DOCTYPE html>` není HTML značka (není párová ani nepárová), ale deklarace typu dokumentu, která prohlížeči řekne, že zobrazuje HTML kód v 5. verzi, tj. HTML 5.

Pokud se tato deklarace neuvede prohlížeč pak neví, jakou verzi HTML má použít pro zobrazení stránky. Důsledek toho je, že některé značky mohou být špatně vykresleny, ...

HURÁ DO VISUAL STUDIO :)

Začneme podle kroků na slidu 7 a do vytvořeného souboru vložíme následující HTML kód:

```
<!DOCTYPE html>
```

```
<u>underline (podtržený) text</u>
```

```
<b>bold (tučný) text</b>
```

```
<i>italic (zkosený) text</i>
```

```
<del>deleted (přeškrtnutý) text</del>
```

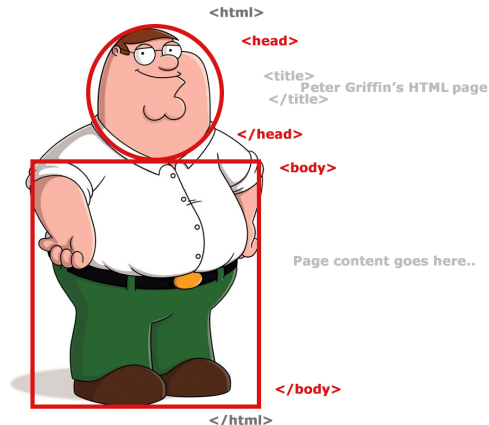
```
<i>zkosený<del>přeškrtnutý</del>text</i>
```

```
<b><u>podtržený</u>tučný text</b>
```


STRUKTURA KAŽDÉ POŘÁDNÉ WEBOVÉ STRÁNKY

Každá webová stránka má svoji pevně danou strukturu skládající se z párových značek `html`, `head`, `title` a `body`.

Pozn. Tato struktura lze po splnění určitých pravidel z části nebo zcela vynechat (jak jsme i na předchozím slidu udělali). Více informací na adrese: html.spec.whatwg.org/multipage/syntax.html#syntax-tag-omission



HURÁ DO VISUAL STUDIO :)

```
<!DOCTYPE html>
<html>

  <head>
    <title>Moje první webovka</title>
  </head>

  <body>
    <u>underline (podtržený) text</u>
    <b>bold (tučný) text</b>
    <i>italic (zkosený) text</i>
  </body>

</html>
```

NADPISY (ANGL. HEADINGS)

Nadpisy vytváříme pomocí párových značek `h1`, `h2`, ..., `h6`, kde `h1` značí největší a `h6` nejmenší velikost nadpisu.

HTML kód:

```
<h1>Nadpis 1. úrovně</h1>  
<h2>Nadpis 2. úrovně</h2>  
<h3>Nadpis 3. úrovně</h3>  
<h4>Nadpis 4. úrovně</h4>  
<h5>Nadpis 5. úrovně</h5>  
<h6>Nadpis 6. úrovně</h6>
```

Výsledek v prohlížeči:

Nadpis 1. úrovně

Nadpis 2. úrovně

Nadpis 3. úrovně

Nadpis 4. úrovně

Nadpis 5. úrovně

Nadpis 6. úrovně

NADPISY A JEJICH POUŽÍVÁNÍ

- 1. úroveň nadpisu je nejdůležitější a 6. úroveň je nejméně důležitý, a tomu by měl odpovídat jejich obsah
- Nadpis 1. úrovně se na stránce vyskytuje typicky jednou a co nejvýše nahoře (protože je nejdůležitější)
- Nadpis 4. úrovně má stejnou velikost jako text na stránce. Nadpisy 5. a 6. úrovně mají dokonce menší velikost než text. Pro nadpisy obecně platí, že pokud potřebuji používat nadpisy 4.–6. úrovně je lepší se zamyslet, jestli není lepší strukturovat obsah jiným způsobem.
- Každý nadpis má definované odsazení shora i zdola.

ODSTAVCE (ANGL. PARAGRAPHS)

Odstavce vždy začínají na začátku řádku a definují odsazení shora i zdola.

HTML kód:

```
<p>Text v prvním odstavci</p>
```

```
<p>Text v druhém odstavci</p>
```

Výsledek v prohlížeči:

Text v prvním odstavci

Text v druhém odstavci

Bez použití odstavců:

HTML kód:

```
Text v prvním odstavci
```

```
Text v druhém odstavci
```

Výsledek v prohlížeči:

Text v prvním odstavci Text v druhém odstavci

SEZNAMY (ANGL. LISTS)

Neuspořádaný seznam

```
<ul>
  <li>Banán</li>
  <li>Jablko</li>
  <li>Hruška</li>
  <li>...</li>
</ul>
```

- Banán
- Jablko
- Hruška
- ...

Uspořádaný seznam

```
<ol>
  <li>Banán</li>
  <li>Jablko</li>
  <li>Hruška</li>
  <li>...</li>
</ol>
```

1. Banán
2. Jablko
3. Hruška
4. ...

Popisný seznam

```
<dl>
  <dt>Kafe</dt>
  <dd>horký nápoj</dd>
  <dt>Mléko</dt>
  <dd>studený nápoj</dd>
</dl>
```

Kafe
 horký nápoj
Mléko
 studený nápoj

VYTVOŘENÍ SEZNAMU UVNITŘ SEZNAMU (ZANOŘOVÁNÍ SEZNAMŮ)

HTML kód:

```
<ul>
  <li>Domů</li>
  <li>Elektronika
    <ul>
      <li>PC</li>
      <li>Notebook</li>
    </ul>
  </li>
  <li>Nákupní košík</li>
</ul>
```

Výsledek v prohlížeči:

- Domů
- Elektronika
 - PC
 - Notebook
- Nákupní košík

SEZNAMY A JEJICH POUŽÍVÁNÍ

- Seznamy se nejčastěji používají pro tvorbu navigace (typicky horní lišta na webu s odkazy)
- Seznamy jsou dobře čitelné pro uživatele zejména, pokud neobsahují příliš mnoho informací (nikdo nemá čas číst mnoho informací . . .)
- Seznamy lze libovolně zanořovat (ideálně zanořovat stejné typy, jinak by to mohlo být pro uživatele zmatečné)

NEJČASTĚJI (NAD)UŽÍVANÉ PÁROVÉ ZNAČKY

Nejpoužívanější značky jsou `div` (z angl. divison, oddíl) a `span`, které používáme jako **kontejnery pro další značky**, čímž vytváříme **strukturu webové stránky**. Dnes již lze tyto obecné značky často nahradit tzv. sémantickými značkami (`nav`, `main`, `aside`, `article`, `section`, ...).

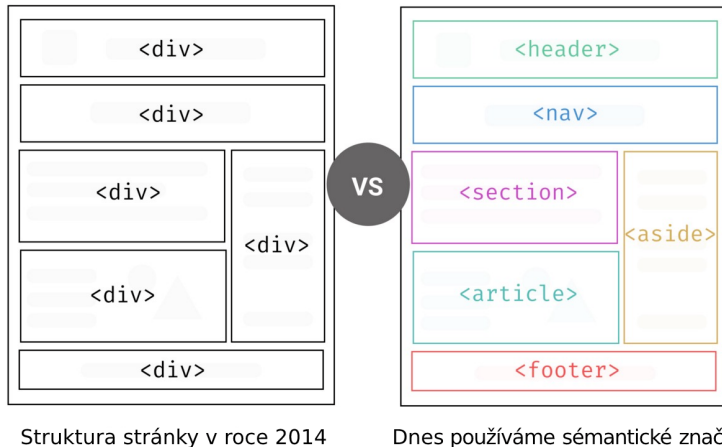
`div` je bloková značka

- Bloková značka vytváří před a za nový řádek, např. `p`, `ol`, `ul`, `table`
- Na 1 řádku pouze 1 značka `div`
- Může obsahovat všechny značky

`span` je řádková značka

- Řádková značka nevytváří před ani za nový řádek, např. `b`, `i`, `u`, `del`
- Na 1 řádku i více značek `span`
- Může obsahovat pouze řádkové značky

POUŽITÍ SÉMANTICKÝCH ZNAČEK PRO TVORBU MODERNÍ WEBOVÉ STRÁNKY



Obrázek 2: Převzato z

<https://medium.com/codex/what-is-semantic-markup-and-why-you-should-use-it-44777543c29c>

KÓDOVACÍ VÝZVA #1

 docs.google.com/document/d/1V_gDwvys7gz_E-josLG98juMT6Mq9l4UAXPV2YOg8jM

- **Cílem je předělat tento dokument 1:1 do HTML** (pouze pomocí párových značek)

Atributy

ATRIBUTY

Atributy nám umožňují upravit chování značek. Nepostradatelné jsou hlavně u nepárových značek, ale používají se hojně i se značkami párovými.

Atribut
⏟
< p *contenteditable*= "true" > HTML je super < /p >

- *contenteditable* je název atributu
- *true* je hodnota atributu
- "=" propojuje název atributu s jeho hodnotou

ATRIBUTY A JEJICH VLASTNOSTI

- Atributy a jejich pravidla používání jsou definované – je třeba se je naučit
- U některých atributu je předem stanovené jakých hodnot mohou nabývat
 - Atribut **contenteditable** může nabýt pouze hodnot **true**, nebo **false**
 - Atribut **title** může nabýt libovolné hodnoty
- Atributy mají vždy definovanou výchozí (slangově „defaultní“) hodnotu
 - Atribut **contenteditable** má výchozí hodnotu **false**
- Každá značka může mít libovolný počet atributů a na jejich pořadí nezáleží
- Atributy lze rozdělit na 2 skupiny:
 - **Obecné atributy** – mohou se vyskytovat na jakékoliv značce
 - **Specifické atributy** – mohou se vyskytovat pouze na jedné, nebo několika značkách

NĚKTERÉ OBECNÉ ATRIBUTY

- **contenteditable** – uživatel může měnit obsah (text) uvnitř značky
- **title** – po najetí myši se ukáže hodnota atributu v bublině, tzv. tooltip
- **id** – identifikátor (nelze mít stejnou hodnotu u více značek) pro CSS styly
- **class** – třída (lze mít stejnou hodnotu u více značek) pro CSS styly
- **style** – hodnota jsou přímo jednotlivé CSS styly odděleny středníkem ;

Párové značky s atributy

ODKAZY

Značka: a

Atributy:

- **href** – cesta k `.html` souboru přímo z operačního systému, nebo Internetu
- **target** – specifikuje, jakým způsobem se nová stránka otevře. Některé hodnoty:
 - **_self** – Otevře stránku v tom samém okně/záložce prohlížeče. Výchozí hodnota.
 - **_blank** – Otevře stránku v nové záložce prohlížeče.
- **rel** – relace (vztah) mezi aktuální stránkou a nově otevřenou stránkou
 - Pro předejití XSS útoku je dobré nastavit hodnotu `noopener noreferrer`.

Použití:

```
<body>  
  <a href="dalsiStranka.html"> Text nebo klidně i obrázek </a>  
</body>
```

FORMULÁŘ A FORMULÁŘOVÉ ZNAČKY

Značky: `form`, `input`, `label`, `button`, `textarea`, ...

Vlastnosti:

- Formulář slouží pro manipulaci (vlození/aktualizaci/smazání) dat v aplikaci
- Základem je prvek `form`, do kterého se vkládají další prvky
- Uvedení všech prvků do původního stavu zařídí tlačítko s atributem `type="reset"`
- Potvrzení formuláře zařídí tlačítko s atributem `type="submit"`

Použití:

```
<form>
  <textarea placeholder="Rekni mi něco o sobě..."></textarea>
  <button type="reset">Vyresetuj formulář</button>
  <button type="submit">Zpracuj data ve formuláři</button>
</form>
```

FORMULÁŘE – ZNAČKA `INPUT`

Značka `input` umožňuje vytvořit **22 různých formulářových komponent** pomocí hodnot v atributu `type`.

https://www.w3schools.com/tags/tag_input.asp

FORMULÁŘE – ZNAČKA `label`

Značka `label` slouží pro přidání popisu formulářových prvků.

- Použití této značky je zejména vhodné kvůli lepší přístupnosti pro screen-readery, protože screen-reader přečte nahlas obsah v této značce, když se uživatel dostane na popisovaný prvek
- Zároveň také zlepšuje UX, protože uživateli stačí kliknout na popisek, aby změnil hodnotu například v checkboxu

Atributy:

- **for** – hodnota tohoto atributu říká, **pro** jaký prvek se stejnou hodnotou v atributu `id` je popisek

Použití:

```
<label for="je-cech" >Jsi cech?</label>  
<input id="je-cech" type="checkbox" />
```

Nepárové značky

NEPÁROVÉ ZNAČKY (ANGL. UNPAIRED TAGS, VOID TAGS)

Nepárové značky neobsahují uzavírací značku, protože otevírací značka je současně **samouzavírací**.

- Nicméně i samouzavírací značka může obsahovat ukončovací lomítko `/`, které však nemá žádný efekt (avšak HTML kód může být díky `/` snáze čitelný ...)
- Do značek tedy nepíšeme text, protože to nelze, jako jsme to dělali u párových značek
- Pracujeme s nimi pomocí tzv. **atributů**, které ovlivňují jejich chování

HTML kód:

```
  

```

Výsledek v prohlížeči:

Zobrazí obrázek, který se jmenuje `pes.png`
Zobrazí obrázek, který se jmenuje `pes.png`

OBRÁZKY

Značka: `img`

Atributy:

- **src** – cesta k obrázku přímo z operačního systému, nebo Internetu. Lze však vložit i celý obrázek ve formátu Base64, ale příliš zvětšuje velikost *.html* souboru.
- **alt** – alternativní text obrázku, který by měl popisovat, co se na obrázku nachází. Tento popis se zobrazí, pokud dojde k chybě při načítání obrázku nebo handicapovaným uživatelům se čtečkou obrazovky⁵.
- **width** – číslo definující šířku obrázku v pixelech (lepší je použít CSS)
- **height** – číslo definující výšku obrázku v pixelech (lepší je použít CSS)

⁵Čtečka obrazovky (angl. screen reader) je technologie/program, který pomáhá slepým/hluchým/. . . uživatelům, kterým obsah webové stránky předřikává nebo překládá do Braillova písma.

Značka: hr

Atributy: zastaralé CSS atributy, které je lepší nastavit pomocí CSS třídy/identifikátoru

Použití:

```
<body>  
  <hr />  
</body>
```

MEZERA

Značka: `br`

Atributy: žádné atributy

Použití:

```
<body>  
  <br />  
</body>
```

LINK

HTML značka **link** připojuje do *.html* souboru nějaký externí soubor – nejčastěji CSS soubor se styly. Tato značka se používá již v hlavičce (*head*) webové stránky.

Značka: link

Atributy:

- **href** – cesta k souboru přímo z operačního systému, nebo Internetu
- **rel** – relace (vztah) externího souboru k *.html* souboru. Pro CSS soubor je hodnota rovna *stylesheet*.

Použití:

```
<head>  
  <link href="styles.css" rel="stylesheet" />  
</head>
```

KÓDOVACÍ VÝZVA #2

 docs.google.com/document/d/1leE50cb1FAp0hyb-hfPN2GtI_B7WII4FpTALWbJc4-0

- **Cílem je přidat do předchozí vytvořené stránky odkazy, obrázky a čáru.**

Co dalšího se může hodit

KOMENTÁŘE

Může se stát, že si do stránky potřebujeme udělat nějakou poznámku, kterou však nechceme ukázat uživateli. Řešením tohoto problému jsou **komentáře**.

- Komentář začíná textem `<!--` a končí `-->`
- Všechno mezi těmito dvěma texty nebude na stránce zobrazeno
- Komentář lze použít uvnitř jakékoliv značky

```
<!-- Vše, co sem napíšu, se na stránce nezobrazí -->
```

```
<div>Toto se zobrazí <!-- Toto už ne --> Toto zase jo</div>
```

VALIDACE HTML KÓDU

Vždy když vytvoříme webovou stránku je vhodné si otestovat, že jsme neporušili žádné HTML pravidla. Testovat lze `.html` soubory i veřejné webové stránky pomocí URL adresy.

https://validator.w3.org/#validate_by_upload

HTML ENTITY

HTML entity lze použít pro zápis znaků, které na klávesnici nemáme nebo pro znaky, které mají v HTML nějaký speciální význam (<, >, &, ").

- HTML entity začínají znakem & (ampersand) a končí ; (středníkem)
- Uvnitř těchto dvou znaků se nachází opět zkratka z anglického slova

Znak	Entita	Zkratka z...
>	>	greater than
<	<	less than
 		non-breaking space
&	&	ampersand
€	€	euro
©	©	copyright

Tabulka 1: Příklady HTML entit.

Shrnutí a závěr

SHRNUTÍ

- HTML je značkovací jazyk, protože se píše (neprogramuje) pomocí značek
- Značky dělíme na párové a nepárové (těch párových je více)
- Párová značka je `b`, `u`, `i`, ... a nepárová značka zase `img`, `br`, ...
- Značky se také odlišují tím, jestli jsou blokové, nebo řádkové. Blokové značky před sebou a za sebou odřádkovávají, zatímco řádkové nikoliv.
- Značky také mohou obsahovat žádný nebo více atributů, na jejichž pořadí nezáleží. Atributy ovlivňují chování značek.
- V HTML kódu lze dělat komentáře a také je žádoucí HTML kód validovat (ověřovat jeho správnost)

ZÁVĚR

Děkuji Vám za pozornost.