

This work is licensed under a Creative Commons “Attribution-ShareAlike 3.0 Unported” license.



HTML+CSS pro SPŠ a VOŠ Kladno

Jaroslav Holeček
holecek@spskladno.cz

Poslední úprava: 20. listopadu 2020

Obsah

1	Úvod	3
1.1	HTML	3
1.2	CSS	3
1.3	Co potřebujeme	3
2	Struktura souborů	4
2.1	Tagy	4
2.2	HTML soubor	4
2.2.1	Metadata	5
2.2.2	Komentáře	5
3	Základní přehled tagů	7
3.1	css	7
3.1.1	Jednotky	7
3.2	Nadpis - <h1>, <h2>, <h3>	7
3.3	Odstavec - <p>	8
3.4	Odkaz - <a>	8
3.5	Obrázek - 	8
3.6	Seznam - , 	8
3.7	Tabulka - <table>	8
3.8	Blok - <div>	9
3.9	Jednořádkový blok - 	9
3.10	Identifikátor - id	9
3.11	Třída - class	10
3.12	Rozložení - layout	10
4	Formulář	11
4.1	Nastavení - <form>	11
4.2	Vstupní políčka - <input> a další	11
4.2.1	Tlačítko - <button>	11
4.2.2	Dlouhý text - <textarea>	11
4.2.3	Rozbalovací seznam - <select>	11
4.2.4	Tlačítko - submit	11
4.2.5	Text - text	11
4.2.6	Heslo - password	12
4.2.7	Email - email	12
4.2.8	Zaškrťovací políčka - checkbox	12
4.2.9	Číslo - number	12
4.2.10	Soubor - file	12
4.2.11	Spoustu a spoustu dalších...	12

1 Úvod

Velmi pěkný tutoriál s ukázkami naleznete na: <https://www.w3schools.com/html/default.asp>
a <https://www.w3schools.com/css/default.asp>

HTML a CSS jsou nerozlučná dvojice při tvorbě webových stránek. Proto je také v tomto dokumentu budeme popisovat zároveň.

HTML i CSS umožňuje nepřeberné množství možností, které se navíc neustále vyvíjejí. Tento dokument má sloužit především pro základní orientaci.

1.1 HTML

HTML je zkratka pro HyperText Markup Language - tedy Hypertextový značkovací jazyk. Hypertextový znamená, že kromě samotného obsahu našich stránek - tedy textů a adres, kde jsou uloženy obrázky, budeme do našich souborů zapisovat ještě něco navíc.

To, co bude navíc budou v názvu zmíněné značky. Těmito značkami budeme označovat části textu a rozdělovat ho tak na nadpisy, odstavce, bloky, ale také těmito značkami určujeme typ textu - tučný, kurzíva.

Také pomocí značek můžeme vytvořit „skupiny“ textu, které mají vypadat stejně, pomocí značek vytvoříme odkazy, tabulky.

1.2 CSS

CSS označuje Kaskádové styly. Je to jazyk, pomocí kterého budeme určovat grafický styl našich stránek.

Pomocí CSS budeme určovat velikost písma, zarovnání, ohraničení, barvu pozadí, ...

1.3 Co potřebujeme

Jediné, co budeme k práci s HTML a CSS potřebovat je **prohlížeč** (Firefox, Chrome, ...) a **textový editor** (SublimeText, Poznámkový blok, Gedit, Vim, ...) ! **NE !** Word. V seznamu textových editorů je uveden Poznámkový blok pro ilustraci, že budeme opravdu psát pouze text - doporučuji použít jiný editor, který vám bude pomáhat tím, že **zvýrazní důležitá slova**, která budeme k textu připojovat, bude Vám ukazovat **závorky**, které k sobě patří apd.

Pro zobrazení našeho souboru **nepotřebujeme internet** - prohlížeč bude zobrazovat náš soubor, který je uložený v našem počítači (stejně jako samotný prohlížeč) a veškeré informace o tom, jak nám má prohlížeč soubor zobrazit mu napíšeme do tohoto souboru my sami.

2 Struktura souborů

2.1 Tagy

HTML soubory jsou textové soubory. Jednotlivé značky, kterými popisujeme obsah (které přidáváme k obsahu) našich stránek se nazývají **tagy**.

Tyto tagy jsou až na několik málo výjimek párové - tedy vždy máme jeden otevírací tag a k němu jeden uzavírací. Zavírací tag se liší pouze přidáním dopředného lomítka /.

Dvojici **otevírací-zavírací** tag si můžeme představit jako krabičku. Jednu krabičku můžeme mít uvnitř jiné krabičky, V krabičce může být i mnoho různých krabiček, které uvnitř sebe opět mohou mít další krabičky.

Jediné co není možné je, aby se tagy křížili - Pokud zapíšu dva otevírací tagy, musím nejprve uzavřít ten druhý a teprve poté první.

Některé tagy umožňují vyplnit **doplňující informace** (např. velikost obrázku, adresa odkazu). Těmto „informacím navíc“ říkáme **atributy**.

CSS soubory jsou také textové soubory, ve kterých zapíšeme, jak se má daný tag zobrazovat. Zapíšeme to jednoduše tak, že napíšeme název tagu a za něj do složených závorek jednotlivé vlastnosti. Každá vlastnost končí středníkem.

Pokud má hodnota jednotky (např. tloušťka ohraničení), píšeme je těsně k číselné hodnotě. U některých vlastností totiž můžeme nastavovat různé parametry (např. u ohraničení - tloušťka, barva, styl (plný, čárkovaný)) a tyto různé parametry jsou odděleny mezerou.

Grafické vlastnosti můžeme zapisovat také do vlastního html souboru pomocí tagu `<style>` v `<head>`, a dokonce i přímo pro každý konkrétní prvek na stránce, pomocí atributu „style“. Této možnosti není od věci někdy využít, ale budeme se snažit co nejvíce věci zapsat do samostatného css souboru. Především je to přehlednější a dále nám to také umožní měnit grafické zobrazení změnou css souboru, bez zásahu do souboru s obsahem stránky. Pokud zapíšete atribut „style“ přímo ke konkrétnímu prvku nejde již tuto nastavenou vlastnost ničím změnit.

2.2 HTML soubor

HTML soubor musí mít určité náležitosti, aby prohlížeč věděl, jak ho má zobrazit.

Na úvod zapíšeme , že se jedná o html dokument, následuje tag `<html>`, ve kterém bude vše ostatní.

V tagu `<html>` jsou dva hlavní tagy `<head>` a `<body>`. Do `<head>` zapisujeme nastavení stránky (vybraný css soubor, nadpis v liště prohlížeče apd.). V `<body>` je samotný obsah naší stránky, který nám prohlížeč zobrazí.

1	<code><!DOCTYPE html></code>	ř. 1: Na úvod souboru zapíšeme, že se jedná o html soubor.
2	<code><html></code>	
3	<code><head></code>	ř. 2-12: Vše je uzavřeno do <code><html></code> tagu.
4	<code><link rel="stylesheet" type="text/css" href="graficky_soubor.css"></code>	
5	<code></head></code>	ř. 3-5: Do tagu <code><head></code> zapisujeme především nastavení stránky.
6	<code><body></code>	
7	<code><!-- Pro přehlednost je dobré, odsazovat tagy, které jsou uvnitř jiných tagů --></code>	
8	<code><h1>Toto je nadpis</h1></code>	ř. 4: Označení souboru css, ze kterého chceme číst grafické nastavení.
9	<code><p>Toto je odstavec.</p></code>	
10		
11	<code></body></code>	ř. 6-11: Samotný obsah stránky zapíšeme do tagu <code><body></code>
12	<code></html></code>	

Zdrojový kód 1: HTML úvod

1	<code>/*Všechny vlastnosti jsou ukončené středníkem!*/</code>	
2	<code>body {</code>	ř. 2. 6: Zapíšeme název tagu a do složených závorek všechny požadované vlastnosti.
3	<code>background-color: lightblue;</code>	
4	<code>}</code>	
5		
6	<code>h1 {</code>	ř. 3: Nastavení barvy pozadí celé stránky.
7	<code>color: navy;</code>	ř. 7: Nastavení barvy nadpisů typu h1.
8	<code>margin-left: 20px;</code>	
9	<code>}</code>	ř. 8: Nastavení odsazení nadpisů h1. Jednotky píšeme těsně k hodnotám

Zdrojový kód 2: CSS úvod

2.2.1 Metadata

Je vhodné do naší stránky zapsat, kdo je jejím autorem, kdy proběhla poslední aktualizace apd.. To provedeme nejlépe pomocí tagu `<meta>`.
Také pomocí `<meta>` můžeme (a měli bychom) nastavit, jak má prohlížeč zobrazit stránku (např. v jakém přiblížení).

1	<code><!DOCTYPE html></code>	ř. 4, 5: Nastavení metadat o stránce.
2	<code><html></code>	
3	<code><head></code>	ř. 6: Nastavení úvodního viewportu - jak se má stránka zobrazit
4	<code><meta name="Autor" content="Nejlepší jméno"></code>	
5	<code><meta name="Poslední úprava" content="ted"></code>	
6	<code><meta name="viewport" content="width=device-width, initial-scale=1.0"></code>	
7	<code></head></code>	
8	<code><body></code>	
9	<code></body></code>	
10	<code></html></code>	

Zdrojový kód 3: Metadata

2.2.2 Komentáře

Psát si ke kódu komentáře je velmi vhodné. Komentář je text, který se na výsledné podobě stránky nijak neprojeví a zapisujeme do něj vysvětlivky ke kódu.

Třeba pro někoho, kdo náš kód bude spravovat, ale i sami prosebe, až se ke kódu vrátíme po měsíci a nebudeme si pamatovat, co jsme to tu vlastně psali). Můžeme také pomocí komentářů zřetelně oddělit některé části stránky (třeba tabulku, kterou často upravujeme) a zpříjemnit si tak orientaci v kódu.

v HTML souboru zapisujeme komentáře to tagu `<!-- ... -->`

v CSS souboru zapisujeme komentáře mezi `/* ... */`

3 Základní přehled tagů

Tagů a jejich grafických úprav můžeme využít nepřeberné množství. Proto vždy, když budete chtít něco vytvořit, podívejte se do dokumentace a téměř jistě najdete jednoduchou možnost, jak to provést.

Zde si ukážeme minimální souhrn tagů, které „musíte“ znát. Samozřejmě i tyto tagy naleznete v dokumentaci, ale na druhou stranu hledat něco každých 5 minut značně zdržuje.

Vzhledem k tomu, že základní grafická úprava pomocí css je pro velkou část tagů stejná. Nebude

zde vypsána u každého tagu, ale pouze u těch, kde se v něčem liší, je v něčem zajímavá.

3.1 css

V css nejčasteji využijeme úpravu:

- barva textu - `color`: zapisujeme barvu slovy (anglicky), případně číslem https://www.w3schools.com/css/css_colors.asp
- font - `font`: nastavujeme velikost, styl, tloušťku, rodinu - https://www.w3schools.com/css/css_font.asp
- ohraničení - `border`: nastavujeme barvu, tloušťku, styl, zakulacení - https://www.w3schools.com/css/css_border.asp
- odkazy - můžeme nastavit různý styl pro odkaz, který je aktivní (míří na něj myš), je již navštívený, právě jsme na něj klikli - https://www.w3schools.com/css/css_link.asp
- tabulky - ohromné množství možností - nastavujeme zvlášť vlastnosti celé tabulky, řádků a buněk - https://www.w3schools.com/css/css_table.asp
- pozadí - můžeme nastavit, barvu, obrázek, každý prvek může mít své pozadí - https://www.w3schools.com/css/css_background.asp
- zarovnání - `align`: zarovnání vlevo, vpravo, na střed, nahoru, dolů - https://www.w3schools.com/css/css_align.asp
- odsazení - máme dva typy odsazení - `margin` a `padding` - jaký je mezi nimi rozdíl ukazuje obrázek zde - https://www.w3schools.com/css/css_boxmodel.asp

3.1.1 Jednotky

Při zadávání rozměrů samozřejmě musíme zadat i jednotky.

Jednotky existují **absolutní**, které udávají pevnou velikost - mm, cm, in, px (milimetry, centimetry, inche, pixely)

a **relativní**, které udávají velikost prvku vzhledem k jinému prvku (třeba velikosti obrazovky) - em, vw, vh, % (relativně k font-size, setině šířky obrazovky, setině výšky obrazovky, k nadřazenému prvku (tomu, ve kterém je tento prvek))

Vybrat správné jednotky je velmi důležité, jinak bude stránka na některých zařízeních (velká obrazovka, malá obrazovka) nečitelná.

3.2 Nadpis - `<h1>`, `<h2>`, `<h3>`

Do tagů `<h1>`, `<h2>`, `<h3>` (headings) zapisujeme nadpisy. Tedy text, který prohlížeč zobrazuje oddělený od dalšího textu a graficky výraznější.

Číslo za „h“ rozlišuje úroveň nadpisů - `<h1>` je největší.

3.3 Odstavec - <p>

Do odstavce <p> (paragraph) píšeme prostý text.

3.4 Odkaz - <a>

Do tagu <a> vládáme prvek, který nás po kliknutí přesměruje na zadanou stránku (nebo její část). Jako odkaz můžeme použít v zásadě cokoliv - text, nadpis, obrázek, blok, ...

Aby prohlížeč věděl, kterou stránku má zobrazit po kliknutí na odkaz, musíme vyplnit atribut „href“.

ř. 8: Odkaz na internetovou stránku.

ř. 9: Odkaz na náš jiný soubor.

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4    <link rel="stylesheet" type="text/css" href="graficky_soubor.css">
5  </head>
6  <body>
7
8    <a href="www.wikipedia.cz">Odkaz na wiki.</a>
9    <a href="moje_druha_stranka.html">Odkaz na naši další stránku.</a>
10
11  </body>
12 </html>
```

Zdrojový kód 4: Odkaz

3.5 Obrázek -

Obrázek můžeme ve stránce zobrazovat pomocí tagu (image). Tento tag je pouze otevírací. Nutný atribut je „src“, kterým určíme kde je obrázek uložen (může být uložen v počítači, nebo na internetové adrese). Vhodné atributy jsou také „alt“, kterým nastavíme náhradní text (zobrazí se, pokud se nepodaří načíst obrázek) a nezapomenout nastavit výšku a šířku - pomocí atributu „style“ (což jsou css, jen vložené přímo k jednomu konkrétnímu prvku).

3.6 Seznam - ,

Seznamů máme dva typy - Číslovaný (ordered list) a Nečíslovaný (unordered list).

Číslovaný seznam může být číslovaný nejen čísly, ale i písmeny, řeckými číslicemi.

Nečíslovaný seznam může mít různé typy odrážek - kruh, kružnice, ...

Celý seznam píšeme mezi tagy , nebo . Jednotlivé **prvky seznamu** pak do tagů (list item).

Ukázky seznamu zde https://www.w3schools.com/html/html_lists.asp

3.7 Tabulka - <table>

Každá tabulka je vložena do tagu <table>.

Dále musí prohlížeč vědět, co jsou jednotlivé řádky a buňky tabulky. Ty zapíšeme do tagů <tr> (table row) pro každý řádek. A do řádku zapíšeme tagy <td> (table data) pro každou buňku v řádku.

Dále často chceme odlišit první řádek - hlavičku tabulky s nadpisem sloupců. To provedeme pomocí použití <th> (table head) tam, kde bychom jinak použili <td>.

Pro lepší možnosti grafického zobrazení tabulky je vhodné ji rozdělit na záhlaví - tělo - zápatí, pomocí `<thead>` - `<tbody>` - `<tfoot>`
Také můžeme tabulce přidat nadpis pomocí `<caption>`

U tabulek - a především jejich grafického nastavení je nejlepší si rovnou prohlédnout výsledek a vybrat si, co se Vám do vašich stránek líbí:

html: https://www.w3schools.com/html/html_tables.asp

css: https://www.w3schools.com/css/css_table.asp

3.8 Blok - `<div>`

Blok `<div>` slouží k uzavření několika (nebo jen jednoho) prvků do jednoho bloku a jejich společné formátování. Lze tak například nastavit pozadí poloviny stránky - polovinu obsahu uzavřeme do bloku a tomuto bloku nastavíme požadované pozadí.

3.9 Jednořádkový blok - ``

Blok `` používáme obdobně jako `<div>`. Zatímco `<div>` se ale vždy roztáhne co nejvíce do šířky, blok `` zůstane jen těsně okolo vnitřního prvku.

```
1  <!DOCTYPE html>                                ř. 8: Zobrazí ohraničení jen okolo obrázku
2  <html>                                           ř. 12: Zobrazí ohraničení až k okrajům
3  <head>                                           stránky
4  <link rel="stylesheet" type="text/css" href="graficky_soubor.css">
5  </head>
6  <body>
7  <!-- Zkuste v prohlížeči přiblížit a oddálit stránku -->
8  <span style="display: inline-block; border:solid green;">
9    
10 </span>
11 <br>
12 <div style="border:solid green;">
13   
14 </div>
15 <br>
16 </body>
17 </html>
```

Zdrojový kód 5: Odkaz

3.10 Identifikátor - `id`

Identifikátor - `id` je **atribut**. Tedy ho přiřazujeme nějakému prvku - tagu.

Můžeme si takto označit konkrétní prvek a pak se na něj odkazovat. Jedno `id` může mít pouze jeden prvek na stránce. Tedy musí být unikátní.

Můžeme vytvořit odkaz tak, že prohlížeč zobrazí tento prvek jako první na stránce.

V CSS můžeme upravit vlastnosti pouze tohoto konkrétního prvku

3.11 Třída - class

Třída - class je také **atribut** a tedy ji přiřazujeme nějakému tagu.

Funguje v zásadě stejně jako id, jen s tím rozdílem, že v jedné třídě může být zařazeno více prvků.

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4    <link rel="stylesheet" type="text/css" href="graficky_soubor.css">
5  </head>
6  <body>
7    <p id="1" class="dulezite">Nápis ve třídě "dulezite" a s id "1"</p>
8    <p class="dulezite">Nápis ve třídě "dulezite"</p>
9    <p id="2">Nápis s id "2"</p>
10 </body>
11 </html>
```

ř. 7: Prvek může být ve třídě (i v několika) a mít id
ř. 8: V jedné třídě může být více prvků
ř. 9: Další prvek musí mít jiné id - id se nemohou opakovat

Zdrojový kód 6: HTML - id, class

```
1  #1 {
2      font-size: 10px;
3  }
4
5  #2 {
6      font-size: 20px;
7  }
8
9  .dulezite {
10     color: red;
11 }
```

ř. 1, 5: Na id odkazujeme pomocí #
ř. 9: Na třídu odkazujeme pomocí . (tečky)

Zdrojový kód 7: CSS - id, class

3.12 Rozložení - layout

Jsou místa na stránce, která jsou tradičně využívána k zobrazení určitých informací.

Pro využití schopnosti prohlížečů tato místa zobrazit uživateli přehledně je vhodné využít připravené layout prvky.

Nejlépe pochopíme o co jde při dostatečně dlouhém dívání na obrázek (ne hned první ukázkou, ale o trochu pod ní) na stránce https://www.w3schools.com/html/html_layout.asp

4 Formulář

Formulář je speciální prvek, který nám umožní vložit na stránku políčka, do kterých můžeme (jako uživatelé stránek) zapsat libovolné údaje (Jméno, věk, datum, zaškrtnout jednu z více možností, zaškrtnout několik z více možností apd.).

HTML ani CSS nám neumožní s těmito zadanými hodnotami něco udělat - uložit si je, něco spočítat, ... K tomu budeme potřebovat PHP (případně JavaScript). O PHP si budeme povídat později. O JavaScriptu, pokud to stihneme, ještě později.

4.1 Nastavení - `<form>`

Formulář zapisujeme do tagu `<form>`.

V otevíracím tagu `<form>` nastavíme, co se má stát (na jakou stránku má přejít prohlížeč) po kliknutí na odesílací tlačítko - to nastavíme pomocí atributu „action“

Dále také nastavíme, jakou metodou se mají odeslat data mezi stránkami - to nastavíme pomocí atributu „method“.

Oba tyto atributy pro nás zatím nejsou důležité, neboť nemáme žádný skript (program), který by data zpracoval.

4.2 Vstupní políčka - `<input>` a další

Velké množství typů vstupních políček naleznete v dokumentaci, případně zde: https://www.w3schools.com/html/html_form_input_types.asp Vstupní políčka jsou jednotlivé prvky formuláře. Těchto prvků je velké množství a zde si ukážeme jen některé základní.

Různé vstupní typy políček nastavíme nejčastěji pomocí atributu „type“, ale několik málo jich má vlastní tag.

Dále je vhodné (a později nutné) přiřadit každému políčku „name“, pomocí kterého z něj později získáme data.

A v neposlední řadě zde užijeme atribut „id“. Pomocí něj můžeme ke vstupnímu políčku přiřadit nápis (label), který pak bude reagovat na naše kliknutí.

4.2.1 Tlačítko - `<button>`

Tlačítko, na které můžeme kliknout a pomocí atributu „onclick“ nastavit, co se poté stane.

4.2.2 Dlouhý text - `<textarea>`

Pole pro zapsání delšího textu. Pomocí atributů „rows“ a „cols“ nastavíme velikost pole.

4.2.3 Rozbalovací seznam - `<select>`

Rozbalovací seznam, ze kterého můžeme vybrat jednu, nebo více možností (podle atributu „multiple“). Jednotlivé možnosti píšeme do tagů `<option>`.

4.2.4 Tlačítko - submit

Tlačítko, po jehož zmáčknutí se odešlou data formuláře na určenou stránku.

4.2.5 Text - text

Textové pole vhodné např. pro jméno.

4.2.6 Heslo - password

Pole, ve kterém není viditelný obsah, který do něj zapíšeme (místo všech znaků jsou např. černé tečky)

4.2.7 Email - email

Pole, ve kterém prohlížeč pohlídá, zda je zadaná emailová adresa.

4.2.8 Zaškrťovací políčka - checkbox

Umožňuje zaškrtnout (a odškrtnout) několik nabízených možností

4.2.9 Číslo - number

Prohlížeč pohlídá, zda je zadáno číslo

4.2.10 Soubor - file

Otevře se okno pro výběr souboru z počítače

4.2.11 Spoustu a spoustu dalších...

Barva, datum, telefoní číslo, ...

```

1  <!DOCTYPE html>
2  <html>
3  <body>
4
5  <form action="zpracovani_dat.php" method="post">
6    <input type="radio" id="muz" name="gender" value="muz">
7    <label for="muz">Muž</label>
8    <br>
9    <input type="radio" id="zena" name="gender" value="zena">
10   <label for="zena">Žena</label>
11   <br>
12   <input type="radio" id="jiny" name="gender" value="jiny">
13   <label for="jiny">Jiný</label>
14   <br>
15   <br>
16   <textarea name="message" rows="10" cols="30">
17     Napiš něco o sobě...
18   </textarea>
19   <br>
20   <br>
21   <label for="em">Zadej email:</label>
22   <input type="email" id="em" name="adresa">
23   <br>
24   <input type="submit" value="Odeslat">
25 </form>
26
27 </body>
28 </html>

```

Zdrojový kód 8: Odkaz

- ř. 5: V otevíracím tagu formuláře nastavíme další vlastnosti.
- ř. 6, 9, 12: Vstupní políčka typu „radio“.
Všechny mají stejnou hodnotu atributu „name“ - tak prohlížeč pozná, že patří k sobě
Každý má své id - pomocí id k němu připojíme label.
Všechny mají „value“ - protože zde nic nevyplňujeme, je to hodnota, která se odešle ke zpracování.
- ř. 7, 10, 13: Popisky jednotlivých tlačítek
Pomocí atributu „for“ je můžeme projit s daným vstupním políčkem.
- ř. 16: Některé prvky mají samostatný tag.
- ř. 22: Některé prvky upravíme pomocí změny atributu „type“.