

# Tvorba webových stránek

---

## Kaskádové styly



Obr 1

# Kaskádové styly - CSS

- Kaskádovými styly je možné definovat vzhled a vlastnosti všech HTML elementů.
- Používáním stylů znamená efektivní oddělení obsahu HTML dokumentu od jeho vzhledu.
- Styly umožňují snadno změnit vzhled internetových stránek, umožňují lepší grafické zpracování.
- Použitím CSS znamená zmenšení zdrojového kódu stránek.

Jazyk byl navržen standardizační organizací W3C, byly vydány zatím dvě úrovně specifikace CSS1 a CSS2. V roce 2011 byla dokončena revize CSS 2.1 a pracuje se na verzi CSS3.



# Syntax

Definice kaskádových stylů sestává z několika *pravidel*. Každé pravidlo obsahuje *selektor* a *blok deklarácí*. Každý blok deklarácí pak obsahuje seznam *deklarácí* oddělených středníky ; a každá deklaráce sestává z identifikátoru vlastnosti, následuje dvojtečka : a hodnota vlastnosti.

## Základní syntaxe (zápis)

- SELEKTOR {vlastnost\_1 : hodnota;  
vlastnost\_2 : hodnota}
- SELEKTOR1, SELEKTOR2, SELEKTOR3 {vlastnost : hodnota}



# Záписы stylů

## I. v těle dokumentu

```
<p align="justify" style="text-indent: 12px; line-height: 150%">Zde se  
nachází text odstavce</p>
```

## II. v hlavičce HTML dokumentu

```
<STYLE type="text/css">
```

```
body {
```

```
color:      red;
```

```
text-align: center;
```

```
font-size:  14px;}
```

## III. v externím souboru

Př.

```
<link rel="stylesheet" href="styly.css" type="text/css" />
```

# Kaskádové styly - CSS

- **Prvek** - nejjednodušší typ selektoru

```
h1 { color: red; }  
h2 { color: red; }  
h3 { color: red; }
```

} lze psát kratší kód `h1, h2, h3 { color: red; }`

- **Potomek** - kontextový selektor, např. sada pravidel pro nastavení barev u víceúrovňového číslování

```
ul li { color: black; }  
ul ul li {color: green; }  
ul ul ul li {color: blue; }
```

- **Třída** – prvku na stránce můžeme přiřadit určitý význam

```
<p class="poznámka">Před použitím je záhodno zázalohovat data.</p>
```

- `p.poznámka { background-color: yellow; }` všechny odstavce náležící do třídy `poznámka` mají žluté pozadí

```
.poznámka { background-color: yellow; }
```

každý prvek zařazený do třídy `poznámka` má žluté pozadí



# Kaskádové styly - CSS

- **Identifikátor** – používají se podobně jako třídy

```
<p id="copyright">&copy; Jan Novák</p>
```

Ve stylovém předpisu pak názvu identifikátoru předradíte znak „#“:

```
#copyright { color: gray; }
```

- **Pseudotřídy** – lze prvky na stránce klasifikovat podle určitých vlastností

## 1. Pro odkazy

```
a:link { color: blue; }
```

```
a:visited { color: purple; }
```

## 2. Dynamické - umožňují reagovat na uživatelskou činnost

```
a:link { color: blue; } - nenavštívené odkazy jsou modré
```

```
a:visited { color: purple; } - navštívené odkazy jsou fialové
```

```
a:hover { color: white; background-color: blue; } – po najetí se změní na bílý text na modrém pozadí
```

```
a:active { color: green; } – je v podstatě zbytečné
```



# Kaskádové styly - CSS

- **Příklad dynamického zvýraznění odkazů**

odkazy „vystoupí“ z textu, když najedete myší

```
a:link { color: #b00; }
```

```
a:visited { color: #400; }
```

```
a:hover { background-color: yellow; }
```

```
p:hover a:link { color: #f00; }
```

```
p:hover a:visited { color: #a00; }
```

Doporučení:

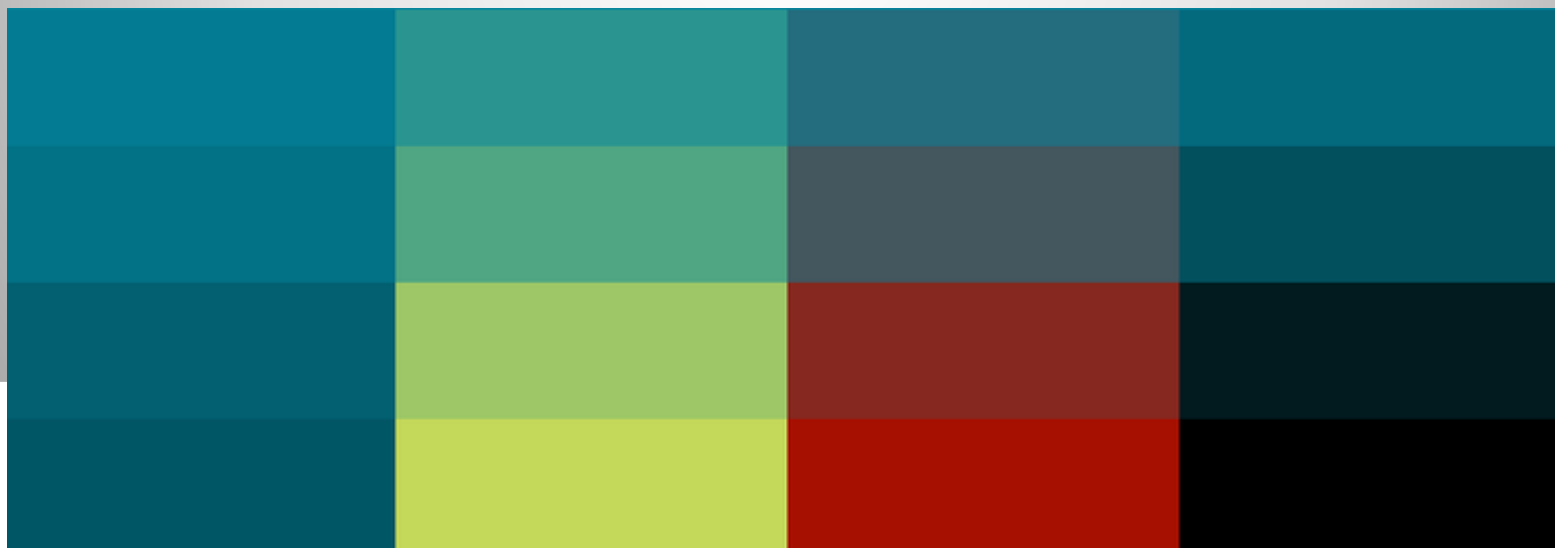
- hrajte si klidně s barvami, pozadím či měňte zobrazený obrázek, ale nesahejte na velikost písma
- nepřepínejte mezi normálním a tučným, neměňte šířku rámečku a podobně.



# Tvorba webových stránek

---

## Kaskádové styly – formát písma a pozadí



Obr 1



# Formát písma

Font-family určuje **font**, který se použije k vykreslení písma - pokud prohlížeč nenajde první písmo, zkouší najít další

Př. body {font-family: "Geneva CE", "Arial CE", sans-serif;}

hodnota	obecné jméno fontu
serif	patkové písmo, nejčastěji Times New Roman
sans-serif	bezpatkové písmo, nejčastěji Arial
cursive	psací písmo
fantasy	volně psané písmo
monospace	jako psací stroj



# Formát písma

Font-size – velikost v jednotkách (px, pt, %, ....), pokud velikost nezadáme bude 16 px

Taktéž je možno zadat velikost pomocí klíčového slova např. small.

hodnota	odpovídá velikosti	
	font size	standardní nastavení prohlížeče
xx-smal	1	9 px
x-small	2	10 px
small	3	13 px
medium	4	16 px
large	5	18 px
x-large	6	24 px
xx-large	7	30 px



# Formát písma

**Font-style** – určuje normální písmo nebo kurzívu

normal	normální písmo
italic, oblique	kurzíva

**Font-weight** – určuje tučnost písma – hodnoty (lighter, normal, bold, bolder,), nebo (100..900)

**Text-decoration**

none	normální písmo
underline	podtržení
overline	nadtržení
line-through	přeškrtnuté
blink	blikání

**Text-transform**

none	normální písmo
capitalize	kapitálky
uppercase	všechna velká
lowercase	všechna malá

**Letter-spacing** – prokládání znaků

**Word-spacing** - zvětšení mezery mezi slovy



# Odstavec

**Line-height** – řádkování (v px, %, nebo násobky např. 1,5)

**Vertical-align** – zarovnání na výšku

hodnoty	na řádku	v buňce tabulky
baseline	na účaří (lince) řádku	na účaří řádku
sub	dolní index(nezmenšený)	nelze použít
super	horní index(nezmenšený)	nelze použít
top	co nevyšše	k hornímu okraji
middle	střed řádku	střed řádku
bottom	co nejniže	k dolnímu okraji
procento	procento výšky	?



# Zarovnání textu (Text-align)

určuje **zarovnání odstavce** nebo jiného bloku.

<b>left</b>	doleva
<b>center</b>	na střed
<b>right</b>	napravo
<b>justify</b>	do bloku

Výchozí zarovnání je doleva s výjimkou prvního řádku, kde odsazení  
text-indent : 8%

**Př.** zarovnání do bloku (text-align: justify)

Pro buňky tabulky se zarovnáním na desetinou čárku ( text-align: ", ")

Příklad zarovnání odstavce:

```
p {margin-top: 2px; margin-bottom: 2px; text-indent: 10%}
```



# Barvy – nastavujeme vlastností (color).

Barva je dědičná a její výchozí hodnota závisí na klientově nastavení (výchozí barva je černá).

## Způsob zadávání:

- **Jméno barvy (red, green.....)**
- **Zadání intenzity jednotlivých barevných složek v režimu RGB :**
  - v podobě šesti číslic v šestnáctkové soustavě (#ffcc00)
  - pro každou barevnou složku jen jednu šestnáctkovou číslici (#fc0)
  - tentokrát ale čísla v desítkové soustavě (0-255)
  - intenzita jednotlivých složek se ale zadává v procentech (0%-100%)

## **Př. Nadpis první úrovně je definován postupně všemi způsoby**

```
h1 { color: red; }
```

```
h1 { color: #ff0000; }
```

```
h1 { color: #f00; }
```

```
h1 { color: rgb(255,0,0); }
```

```
h1 { color: rgb(100%,0%,0%); }
```



# Barva pozadí - background-color

- Vlastnost není dědičná, ale její výchozí hodnotou je průhledné pozadí.
- Je vhodné definovat barvu textu, pozadí a odkazů společně

Př. – definice barev na stránce

```
<style type="text/css">
```

```
body {
```

```
    background-color: #1919B3;
```

```
    color: #FFB200;
```

```
}
```

```
a:link { color: #B30000; }
```

```
a:visited { color: #7D0000; }
```

```
</style>
```



# Obrázek na pozadí - background-image

background-image: url("adresa");

- Umístění left (doleva), right (doprava), top (nahoru), bottom (dolů), center (doprostřed), jejich kombinace, nebo přesná vzdálenost

Př. background-position: top left; - umístí obrázek do levého horního rohu.

background-position: 100px 20px; - umístí obrázek o 100 pixelů doprava a o 20 pixelů dolů od levého horního rohu (lze taky v %)

- Opakování (obrázkem lze vydláždit celou plochu)

background-repeat

repeat – opakovat v obou směrech

repeat x – opakovat ve vodorovném směru

repeat y – opakovat ve svislém směru

no-repeat – umístit obrázek jen jednou

Př.

```
body {  
    background-color: #e6aa50;  
    background-image: url("pozadi.gif");  
    color: #f000;  
}
```

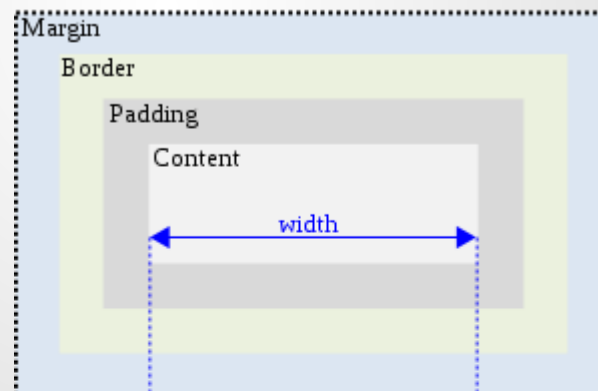




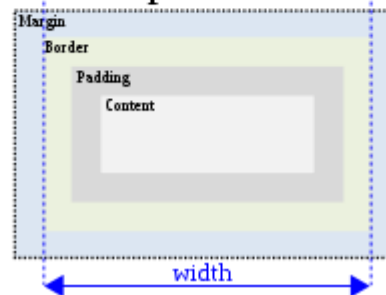
# Tvorba webových stránek

## Kaskádové styly – používané jednotky, blok

W3C box model



Internet Explorer box model



# Používané jednotky

## 1. Absolutní jednotky

Značka	Název	Převod	Počet px
mm	milimetr		1 mm = 3,78 px
cm	centimetr	1 cm = 10 mm	1 cm = 37,8 px
in	palec (inch)	1 in = 2,54 cm	1 in = 96 px
pt	typografický bod	72 pt = 1 in	3 pt = 4 px
pc	pica (cicero)	1 pc = 12 pt, 6 pc = 1 in	1 pc = 16 px

## 2. Relativní jednotky

Značka	Název	Význam
em	em, čtverčik	šířka velkého písmena M
ex	ex	výška malého písmena x

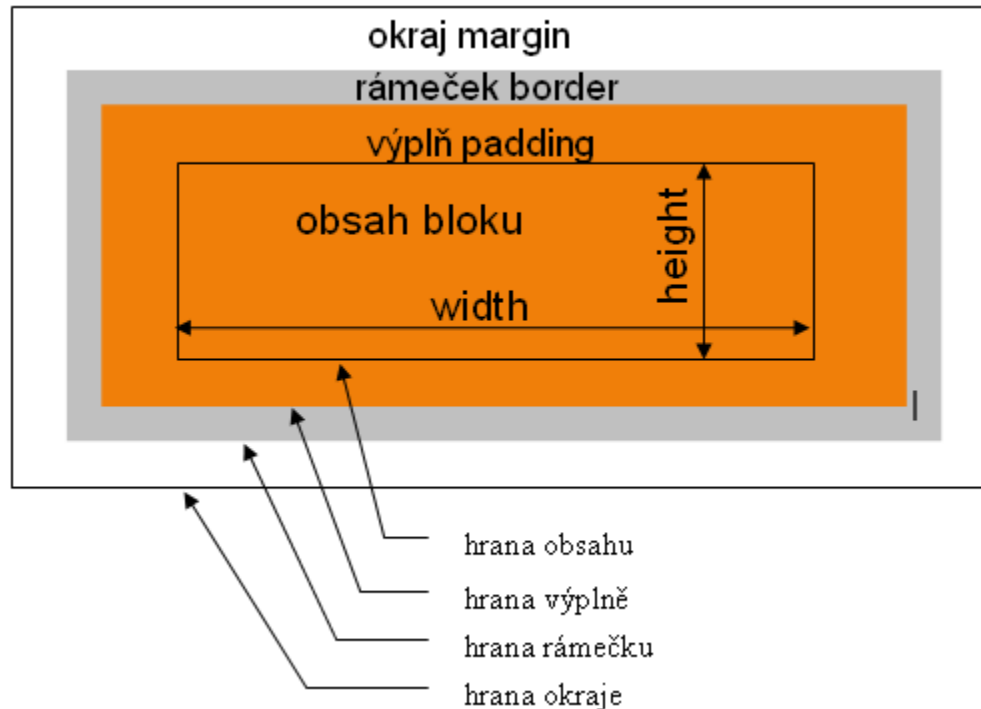
Většinou platí:

**1em ≈ 2ex**



# Blok (v originále box)

Blok hraje ve formátování pomocí CSS velmi důležitou roli (např. blok vytvořený vloženým obrázkem, nebo odstavcem).



# Blok - okraje

**Okraje bloku** – margin - (top, right, bottom, left) - ve směru hodinových ručiček

Př. typické nastavení odstavců

```
P {  
    margin-top: 1em;  
    margin-right:0;  
    margin-bottom: 1em;  
    margin-left:0;  
}
```

Vlastnost margin: je-li údajů méně než čtyři, tak převezme chybějící údaj z protilehlého okraje ⇒ u předchozího příkladu by se dal zjednodušit zápis na:

```
p { margin: 1em 0; }
```

• **Prolínání okrajů** – pokud padnou pod sebe dva bloky s nenulovými okraji, dojde k prolnutí okrajů. Pokud jsou jejich velikosti odlišné, zůstane větší z nich. Prolínání se týká jen horních a dolních okrajů.



# Blok – výplň

**Výplň bloku** – padding - určuje odstup obsahu bloku od jeho rámečku, má stejné pozadí, jako obsah prvku – vlastnosti (top, right, bottom, left)

Př.

**Pokusný nadpis**

Přidám 0.1 em na svislých stranách a 0.5 na vodorovných:

```
padding: 0.1em 0.5em;
```

**Pokusný nadpis**



# Blok - rámečky

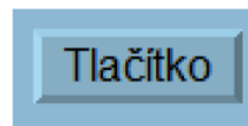
Vedle šířky můžete řídit i jeho barvu a styl.

- **Šířka** - border-top-width  
border-right-width  
border-bottom-width  
border-left-width

Výchozí hodnotou je medium, tedy středně silný rámeček (ovšem nezobrazený).

- **Barva** - výchozí hodnotou je barva prvku, jenž je rámečkem lemován, zadáváme pouze v případě, že ji chceme změnit border-top-color....
- **Styl** – typ čáry, kterou bude rámeček tvořen  
none – žádný, solid – plná čára, dashed – tečkovaná čára, double – dvojitá,  
inset – zamáčknuté tlačítko, outset – vystouplé tlačítko, ridge – plasticky  
vystupující, groove – vyrytá do pozadí

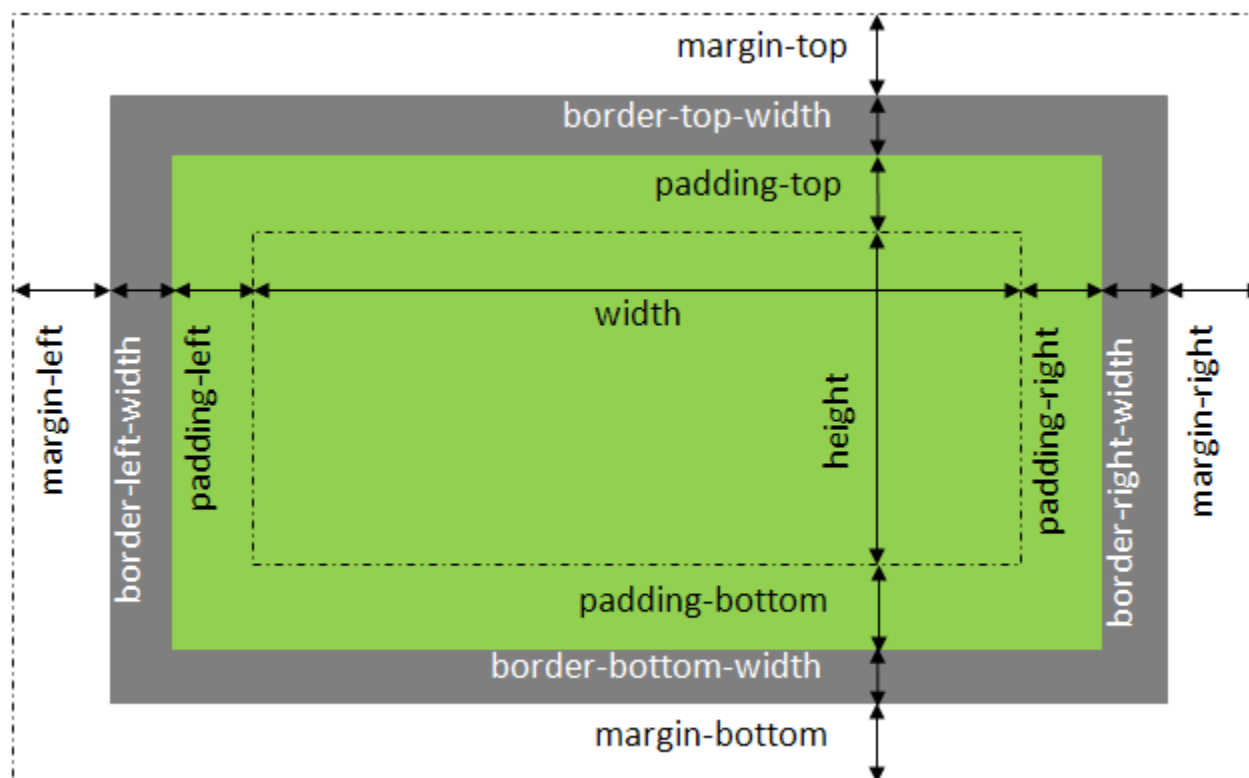
Př. tlačítko    font-family: sans-serif;  
padding: 0.1em 0.5em;  
background-color: #85aec4;  
border: 3px solid #85aec4;  
border-color: #9dcee8 #5b7784 #698999 #a7daf2;



# Blok – rozměry

**Šířka obsahu width** – (200px, nebo auto zabere veškerý dostupný prostor)

**Výška obsahu height** – (stejně jako u šířky, ale určení výšky postupuje zdola)



# Display - určuje způsob zobrazení prvku

hodnoty	základní způsoby zobrazení prvku
block	Prvek se zobrazí jako blok, tj. s řádkovým zlomem na konci
inline	Prvek se zobrazí jako kus řádku, tj. bez řádkového zlomu na konci
Inline-block	Prvek se zobrazí jako kus řádku bez zlomu na konci, jdou mu nastavit rozměry
List-item	Prvek by se měl zobrazit jako položka seznamu, tj. <li>
Tabulkové hodnoty	Prvky se zobrazí jako části tabulky (nefunguje v IE 6 a 7)
none	Prvek se nezobrazí

**Display: none** - prvky dokumentu lze skrýt, po prvku nezůstane volné místo, použití při tisku.





# Display – příklad MENU

- Celé menu je zabaleno do div, pokud klient bude podporovat CSS, čára zmizí. Proto je obalena prvkem `<span>`

```
<div id="menu">
<a href="p-display1.html">raz</a>
<span>|</span>
<a href="p-display1.html">dva</a>
<span>|</span>
<a href="p-display1.html">tři</a>
<span>|</span>
<a href="p-display1.html">čtyři</a>
</div>
```

Pokud klient nepodporuje CSS  
zobrazí se menu takto:

[raz](#) | [dva](#) | [tři](#) | [čtyři](#)

```
#menu a { display: block; }
z prvků a udělám obdélníkové bloky
#menu span { display: none; }
aby nám oddělující svislé čáry nepřekážely
#menu { width: 150px; }
#menu span { display: none; }
#menu a {
    display: block;
    color: #fff2bf;
    background-color: #730000;
    text-decoration: none;
    padding: 3px 5px;
    border-bottom: solid 1px #8c1c1c;
}
#menu a:hover { background-color: #990000; }
```



# Obtékání prvků - float

**Float** - obtékání prvku (none, right, left)

<p>

```

```

Text, který bude nalevo vedle obrázku, další text atd.</p>

Pokud používáme obtékání často, tak je vhodné si připravit pro ně třídy a obtékání zajistit prostým přiřazením prvku jedné ze tříd.

Např.

```
.floatright {  
    float: right;  
    margin: 5px 0 5px 10px;  
}
```

```
.floatleft {  
    float: left;  
    margin: 5px 10px 5px 0;  
}
```

potom :

```

```

**Zákaz obtékání** – např. u nadpisů druhé úrovně chci zakázat obtékat kolem čehokoliv. `h2 { clear: both; }`



# CSS pozicování - vlastnost position

Jakýkoliv objekt (obrázek, tabulka, text, cokoliv) lze umístit kamkoliv na stránku, s objekty se může posouvat a mohou se překrývat.

O jeho konkrétní poloze pak rozhodnou vlastnosti top, right, bottom a left. Vlastnost position může nabývat následujících hodnot:

**Static** - normální tok dat je výchozí hodnotou vlastnosti position

**Relative** - relativní pozice určuje pouze, o kolik se má objekt posunout oproti své normální poloze. Např. první dvě slova odstavce zařazena do třídy dolu budou posunuta.

```
.dolu {  
    position: relative;  
    bottom: -3em;  
}
```

**Absolute** - absolutní pozice umístí objekt do stránky na udané souřadnice bez ohledu na okolní text.

**Fixed** - pevné umístění. Velmi se podobá absolutnímu, poloha se však vztahuje k oknu prohlížeče (Pozor! nepodporuje jej MSIE)

