

7 CSS - VLASTNOSTI PÍSM A TEXTU, PSEUDOTRIEDY

V tejto kapitole budeme skúmať rôzne vlastnosti textu a písma. Mnohé z nich vám budú určite známe z práce v textovom editore. Dozvieme sa viac o tom, ako sa v kaskádových štýloch dajú definovať farby. Naučíme sa tiež zmeniť správanie zvolených elementov na stránke vtedy, keď nad ne nadídeme myšou.

Vlastnosti písma a textu

V kapitole 3 sme si ukázali, ako pomocou kaskádových štýlov nastavíme typ písma (vlastnosť `font-family`). Okrem typu môžeme rozhodnúť aj o iných vlastnostiach písma, napr. o veľkosti, či má byť písmo tučné, šikmé, podčiarknuté, prečiarknuté, atď. Niektoré z týchto vlastností vieme nastaviť pomocou HTML elementov (viete ktoré a pomocou ktorých elementov?), avšak kaskádové štýly nám ponúkajú oveľa širšiu ponuku rôznych nastavení písma a textu ako samotné HTML.

Tučné, šikmé a podčiarknuté písmo

PRÍKLAD 7.1

V editore JSFiddle skopírujeme do okna HTML zdrojový kód zo súboru 07/vitaminy.html. Definujme kaskádové štýly tak, aby všetky nadpisy úrovne 1 boli podčiarknuté, všetky nadpisy úrovne 2 boli šikmé a všetky odkazy boli tučné. (Len pripomíname, že v editore JSFiddle píšeme definície štýlov do CSS časti.)

```
h1 {
  font-family: Verdana;
  text-decoration: underline;
}
h2 {
  font-style: italic;
}
a {
  font-weight: bold;
  color: blue;
}
```

Pomocou vlastnosti `font-weight` nastavujeme hrúbku písma. Ak chceme, aby písmo bolo tučné, použijeme hodnotu `bold`. Šikmé písmo nastavíme pomocou vlastnosti `font-style` s hodnotou `italic`. Pomocou vlastnosti `text-decoration` môžeme písmo podčiarknuť, prečiarknuť, či dokonca nadčiarknuť. Pre podčiarknuté písmo zvolíme hodnotu `underline`.





ÚLOHA 7.2

V CSS kóde z príkladu 7.1

- vyskúšajte niektoré iné hodnoty vlastnosti `font-weight` uvedené v tabuľke 7.1, napr. `normal`, `bolder`, `lighter`, `500`, ...,
- experimentujte s rôznymi hodnotami vlastnosti `text-decoration`, napr. `underline`, `overline`, `line-through`.
- V tabuľke 7.1 si pozrite vlastnosti `font-variant` a `text-transform`. Vyskúšajte ich. Experimentovaním zistíte, aký je rozdiel medzi `font-variant: small-caps` a `text-transform: uppercase`.

Otázka: Porozmýšľajte, ktoré elementy majú prednastavenú hrúbku písma na `bold`, a ktoré majú prednastavené šikmé písmo? Aké prednastavené vlastnosti majú odkazy?

Ako môžeme zrušiť nejakú vlastnosť elementu? Ak sme ju definovali sami v nejakom kaskádovom štýle, stačí ju z príslušného štýlu odstrániť. Ak je to nejaká prednastavená vlastnosť elementu, musíme ju „potlačiť“, t.j. použiť takú hodnotu príslušnej vlastnosti, ktorá je neutrálna – zväčša `normal` alebo `none`.



PRÍKLAD 7.3

Pokračujme v štylovaní v editore JSFiddle. Zrušme podčiarknutie nadpisov úrovne 1, tučné písmo v nadpisoch z úrovne 2 a podčiarknutie odkazov.

```
h1 {  
  font-family: Verdana;  
}  
h2 {  
  font-style: italic;  
  font-weight: normal;  
}  
a {  
  font-weight: bold;  
  color: blue;  
  text-decoration: none;  
}
```

Podčiarknutie nie je prednastavená vlastnosť nadpisov úrovne 1, definovali sme ju my v *príklade 7.1*, takže ju zrušíme jednoducho tým, že z definície štýlu pre `h1` zmažeme riadok `text-decoration: underline`. V prípade odkazov, ktoré majú podčiarknutie prednastavené, zrušíme podčiarknutie nastavením vlastnosti `text-decoration` na hodnotu `none`. Nadpisy (nielen úrovne 2) majú prednastavenú hrúbku písma na tučné, čo dokážeme zrušiť nastavením hodnoty `font-weight` na `normal`.

Veľkosť písma

Veľkosť písma nastavujeme pomocou vlastnosti `font-size`. Môžeme ju zadať v rôznych jednotkách, napr. v pixeloch (`px`), v percentách (%) a jednotkách `em`.

Veľkosť písma v pixeloch

PRÍKLAD 7.4

V kóde z predchádzajúceho príkladu zmeňme veľkosť písma nadpisov `h1` na `30px` a veľkosť písma odsekov na `18px`.

```
h1 {  
  font-family: Verdana;  
  font-size: 30px;  
}  
p {  
  font-size: 18px;  
}
```

Jednotka pixel je vám určite známa, spája sa s rastrovou grafikou a udávajú sa v nej napr. veľkosti fotografií. Prednastavená veľkosť normálneho textu (teda napr. v odseku) v prehliadačoch je `16px`. Pomocou jednotky `px` nastavujeme veľkosť písma absolútne, t.j. na konkrétnu veľkosť, bez ohľadu na to, akú veľkosť majú okolité elementy.

Veľkosť písma v percentách

PRÍKLAD 7.5

Zmeňme CSS kód z príkladu 7.4 nasledovne:

```
h1 {  
  font-family: Verdana;  
  font-size: 150%;  
}  
p {  
  font-size: 90%;  
}
```

Ak použijeme na nastavenie veľkosti písma percentá, nastavujeme veľkosť písma relatívne vzhľadom na nadradený element (element, v ktorom sa nachádza náš element, v našom prípade `<body>`). Ak za základ (100%) vezmeme veľkosť textu v elemente `<body>`, potom `font-size` s hodnotou 90% definuje písmo o niečo menšie ako základné a `font-size` s hodnotou 150% písmo, ktoré je o polovicu väčšie ako základné.

Veľkosť písma v em

PRÍKLAD 7.6

Príklad 7.6: Zmeňme CSS kód z príkladu 7.5 nasledovne:

```
h1 {  
  font-family: Verdana;  
  font-size: 1.5em;  
}  
p {  
  font-size: 0.9em;  
}
```

Výsledná stránka v *príklade 7.6* bude rovnaká ako v *príklade 7.5*. Jednotka `em` rovnako ako `ex` umožňuje relatívne nastavenie veľkosti písma vzhľadom na nadradený element. `1em` je 100%, a teda napr. `2em` je 200%, `0.5em` je 50% atď.

Upozornenie: Pri všetkých jednotkách musíme hodnotu spolu s jednotkou uvádzať ako jedno slovo (bez medzery!), napr. `2.5em`, `130%`, `800px`. Ak je hodnotou desatinné číslo, zapisujeme ho s desatinnou bodkou.



ÚLOHA 7.7

V CSS kóde z predchádzajúcich príkladov skúšajte pre jednotlivé elementy rôzne hodnoty `font-size` s rôznymi jednotkami, napr.

- pre element `h1` skúste: `2em`, `2.5em`, `3em`, `180%`, `250%`, `30px`, `40px`, ...
- pre element `p` skúste `120%`, `0.8em`, `14px`, ...
- definujte `font-size` pre element `li` alebo `a` a experimentujte.

Absolútna a relatívna veľkosť písma



ÚLOHA 7.8

V dvoch samostatných oknách (kartách) prehliadača si otvorte súbory `07/font-size-rel.html` a `07/font-size-abs.html`. Taktiež si tieto súbory otvorte v editore, v ktorom píšete HTML kód. HTML časť je v oboch stránkach rovnaká, líšia sa len v štýloch. V súbore `font-size-rel.html` sme použili relatívne nastavenia veľkostí pre elementy `p` a `strong`, v súbore `font-size-abs.html` absolútne. V oboch súboroch sme veľkosť písma v elemente `body` nastavili na `14px`.

Experimentujte s hodnotou `font-size` pre element `body`, ale tak, že nastavíte vždy rovnakú hodnotu v oboch súboroch. Jednotku `px` nemeňte. Porovnajme obe výsledné stránky.

Zarovnanie textu

Zarovnanie textu určite poznáte z textových editorov. Text v odseku môžeme zarovnať vľavo, na stred, vpravo, alebo vľavo aj vpravo (pozri *obrázok 7.1*). V kaskádových štýloch slúži na zarovnanie textu vlastnosť `text-align`. Môže nadobúdať jednu z hodnôt `left`, `center`, `right` alebo `justify`. Vlastnosť `text-align` môžeme použiť len pre blokové elementy.



Obrázok 7.1 Možnosti pre zarovnanie textu.



ÚLOHA 7.9

V editore JSFiddle v kóde z úlohy 7.7 upravte štýl pre element `<p>` – postupne vyskúšajte všetky hodnoty `text-align`. Pri skúšaní odporúčame zmenšiť šírku okna s výslednou stránkou.

V *tabuľke 7.1* sumarizujeme niektoré vlastnosti písma a ich hodnoty. Zoznam vlastností, ani ich hodnôt nie je úplný. Kompletný zoznam vlastností písma nájdete na https://www.w3schools.com/css/css_font.asp.

Tabuľka 7.1 Vlastnosti písma.

vlastnosť	hodnoty	popis, význam
<code>font-weight</code>	normal bold bolder lighter 100 200 300 ... 800 900	nastaví hrúbku písma
<code>font-style</code>	normal italic oblique	nastaví štýl (šikmosť) písma
<code>font-variant</code>	normal small-caps	zobrazí text normálne, alebo ako malé KAPITÁLKY (veľkosť písmen je zachovaná, ale všetky vyzerajú ako veľké)
<code>font-family</code>	napr. Arial, Helvetica, sans-serif, "Times New Roman", Times, serif, "Courier New", Courier, mono	určuje tzv. rodinu písma viacslovné názvy musia byť v úvodzovkách alebo v apostrofoch
<code>font-size</code>	<i>výška_písma</i> v px, napr. 16px <i>výška_písma</i> v em, napr. 2em <i>výška_písma</i> v %, napr. 140%	nastaví výšku písma elementu hodnoty v jednotkách em a % sa vzťahujú k elementu, do ktorého je tento element vnorený
<code>color</code>	názov farby, napr. green RGB kód farby, napr. #FFCC00	farba textu
<code>text-decoration</code>	none underline overline line-through	nastavenie podčiarknutia, nadčiarknutia, prečiarknutia a blikania textu
<code>text-transform</code>	none capitalize uppercase lowercase	mení veľké a malé písmená
<code>text-align</code>	left right center justify	(vodorovné) zarovnanie textu v rámci blokového elementu

Viac o farbách

Jednou z vlastností textu je jeho farba. Zatiaľ sme hodnotu farby definovali len pomocou názvu farby (napr. `black`, `green` a pod.) V HTML a CSS môžeme zapísať hodnotu farby viacerými spôsobmi. Najčastejšie používané uvádzame v *tabuľke 7.2*.

Tabuľka 7.2 Spôsoby zápisu farieb.

spôsob	zápis	príklad
meno_farby	meno/názov farby podľa štandardu	<code>black</code> , <code>cadetblue</code>
#RRGGBB	zložky r,g,b vyjadrené v hexadecimálnom tvare ak sú dvojice zložiek rovnaké, môžeme použiť skrátený zápis #RGB	<code>#FAC305</code> <code>#FFCC00</code> <code>#FC0</code>
rgb(r, g, b)	zložky r,g,b vyjadrené ako čísla (0 – 255) alebo percentá (0% – 100%)	<code>rgb(255, 104, 0)</code> <code>rgb(100%, 41%, 0%)</code>
rgb(r, g, b, a)	zložky r, g, b ako čísla (0 – 255) alebo percentá (0% – 100%) s priehľadnosťou (0.0 – 1.0)	<code>rgb(255, 104, 0, 0.5)</code> <code>rgb(100%, 41%, 0%, 50%)</code>



ÚLOHA 7.10

Na stránke IT Pizza (súbor 07/index.html) doplňte potrebné štýly, prípadne iba vlastnosti tak, aby:

- odkazy na stránke boli sivé (#999999), tučné a nepodčiarknuté,
- nadpis úrovne 1 bol 3.5krát väčší ako bežný text, veľkými písmenami.

IT PIZZA



ponuka fotogaléria kontakt

Obrázok 7.2 Nadpis a odkazy na stránke IT Pizza.

Pseudotrieda :hover

Určite ste si na mnohých stránkach všimli, že niektoré prvky v nich menia svoj vzhľad pri istých udalostiach, napr. keď na ne prídeme myšou alebo keď na ne klikneme.



ÚLOHA 7.11

Uvedte čo najviac príkladov, ako sa na stránkach zmení vzhľad odkazov, keď na ne prídeme myšou.

Asi najčastejšia zmena, ktorá nastáva v prípade odkazov, nad ktorými je myš, je zmena farby textu, zmena farby pozadia či podčiarknutie, prípadne ich kombinácie. Zmenu vzhľadu odkazov pri nadídení myši vieme zadať v rámci kaskádových štýlov pomocou tzv. pseudotriedy `:hover`.



PRÍKLAD 7.12

Do HTML okna v editore JSFiddle nakopírujme obsah súboru 07/vitaminy.html a do CSS okna obsah súboru 07/vitaminy.css. Všetkým odkazom, ak ponad ne nadídeme myšou, zmeníme farbu na červenú. Do okna CSS pridáme nasledujúci kód a vyskúšame.

```
...
a {
  font-weight: bold;
  color: blue;
  text-decoration: none;
}
a:hover {
  color: red;
}
```

ÚLOHA 7.13



Urobte nasledujúce zmeny. Zakaždým stránku vyskúšajte.

- V definícii štýlu `a:hover` nahraďte farbu `red` inou farbou.
- V definícii `a:hover` pridajte podčiarkovanie textu.
- V definícii `a:hover` nastavte farbu textu na bielu a farbu pozadia (`background-color`) na modrú.
- V definícii `a:hover` nastavte veľkosť písma na 110%. Čo táto zmena spôsobí na stránke?

Pseudotriedy slúžia na definovanie „špeciálneho“ stavu elementu na stránke za istých okolností. Jednou z takýchto okolností je práve nadídenie myšou ponad element. Definovanie vzhľadu elementu, nad ktorým je myš, umožňuje pseudotrieda `:hover`. My sme ju v príklade 7.12 definovali pre element `a`, čím sme nastavili vzhľad odkazov, nad ktorými je myš. Pseudotriedu `:hover` však môžeme používať s ľubovoľným elementom.

PRÍKLAD 7.14



Pri prechode myšou ponad nadpis úrovne 1 zmeníme písmo na kapitálky.

```
h1 {  
    font-family: Verdana;  
}  
h1:hover {  
    font-variant: small-caps;  
}
```

Aj keď pseudotriedu `:hover` môžeme používať s ľubovoľným elementom, treba vždy zvážiť, či jej použitie bude mať praktický význam. Napr. naša posledná zmena z príkladu 7.14 žiaden praktický význam nemá. Tiež treba dobre zvážiť, ktoré vlastnosti elementu budeme v pseudotriede `:hover` meniť. Rôzne „blikajúce efekty“ či efekty „poskakujúceho textu“ (ako ste videli v poslednej z úloh 7.13) môžu na čitateľa pôsobiť skôr rušivo. Mali by sme robiť len také zmeny, ktoré nezmenia rozloženie prvkov na stránke.

ÚLOHA 7.15



Na stránke IT Pizza nastavte farbu pozadia odkazov, nad ktorými je myš, na `#ff9900`.

ÚLOHA 7.16



Na stránke `07/ucebnica.html` definujte štýly pre elementy `body`, `h1`, `h2`, `h3`, `h4`, `p`, `a`, `a:hover`. Vlastnosti a ich hodnoty si zvolte podľa vlastného uváženia.

TVORBA WEBOVÝCH STRÁNOK

Obsah: Úvod Štruktúrovanie textu na webovej stránke Úvod do kaskádových štýlov Obrázky Odkazy

1. ÚVOD

S webovými stránkami (alebo HTML stránkami) sa dnes stretávame denne: hľadáme na nich informácie, komunikujeme s kamarátmi, nakupujeme v online obchodoch, počúvame hudbu, ... Webové stránky majú školy, mestá, osoby, banky, obchody ...

Webová stránka je kombinácia textu, obrázkov, animácií, hypertextových odkazov a ďalších objektov, ktoré sú webové prehliadače schopné zobraziť. Vytvoriť webovú stránku znamená vytvoriť jej obsah („vložiť“ texty, obrázky, tabuľky, odkazy a ďalšie spomínané objekty) a rozhodnúť, akým spôsobom sa bude tento obsah zobrazovať, t.j. ako budú jednotlivé objekty na stránke rozložené, aké budú mať vlastnosti, napr. aké farby použijeme, aký druh, štýl či veľkosť písma, riadkovanie, medzery medzi jednotlivými prvkami na stránke, atď.

1.1 Zdrojový kód stránky, jazyk HTML

Webová stránka je v skutočnosti obyčajný textový súbor, v ktorom je pomocou špeciálnych značiek popísané, čo sa má na webovej stránke zobraziť a ako sa to má zobraziť – tzv. zdrojový kód stránky.

Obrázok 7.3 Ukážka stránky ucebica.html s nastavenými štýlmi.