

Danas je: 27. 9. 2012.

Nalazite se ovdje: Naslovnica >> Savjeti >> Web dizajn >> Uvod u CSS

Web dizajn

- Osnove web dizajna
- Uvod u HTML
- Uvod u CSS

Savjeti

- Web upotrebljivost
- Web dizajn
- Dreamweaver

Infografika



Prezentacija



Find us on Facebook

74 people like **Web upotrebljivost**.

Anita Katarina Decko



Gabriella Borjan Selmir

Facebook social plugin

Uvod u CSS



9



članak objavljen: 05. 02. 2009.

Napomena:

Ovaj članak je napisan za početnike, odnosno sve one koji se po prvi put susreću sa stilskim listama - CSS (Cascading Style Sheets) kojim se definira prezentacijski stil web dokumenata. Članak je podijeljen u nekoliko glavnih cjelina kako bi se čitatelj mogao jednostavnije snalaziti na stranici.



Brzi linkovi na određeni dio sadržaja stranice

1. Uvod u CSS | 2. Anatomija CSS elementa | 3. Selektori i atributii |
4. XHTML DOM model | 5. DIV i SPAN elementi | 6. CSS Box model |
7. Klase i ID-ovi | 8. Svojstva teksta | 9. Margine i padding |
10. Načini povezivanja CSS-a sa HTML dokumentom |
11. Spremanje CSS dokumenta | 12. Provjera ispravnosti CSS koda

Alati

 Ispiši članak
 SHARE

1) Uvod u CSS

Napomena:

U ovoj lekciji ću govoriti o CSS-u, odnosno o njegovoj zadnjoj verziji CSS 2.1 (2009.) - koji je unaprijeđen u odnosu na starije verzije CSS-a kao što je npr. CSS 1.0. Dakle, kada govorim o CSS-u – mislim na njegovu aktualnu verziju CSS 2.1 (iako već postoji i CSS 3 verzija, ista nije još preporučena kao konačna specifikacija od W3.org i većina web preglednika ju još ne podržava).

Također, neke engleske ili hrvatske nazive i kratice ću ponekad pisati i opisivati u njihovoj hrvatskoj inačici, npr.:

- CSS = CSS dokument = stil
- border = okvir = rub
- (X)HTML = HTML i XHTML
- oblikovanje = formatiranje

Što je CSS?



CSS je kratica za *Cascading Style Sheets* i u „nespretnom“ hrvatskom prijevodu bi značilo: stilske liste u kaskadi. Znam, znam, ovaj prijevod vam ne govori ništa; stoga ću vam još jednom objasniti, ovaj put na pravom hrvatskom: CSS je tzv. „stilski jezik“ koji omogućava dodavanje različitih **stilova** HTML elementima kao što su: veličina i vrsta pisma, pozadinske boje, okviri, položaj elemenata, margine, itd...Svrha CSSa je zapravo odvajanje stila prezentacije od samog sadržaja web stranice...

Korisni linkovi:

Web stranice veÄš od 700kn

besplatni hosting za naÄje
klijente najbolji omjer cijene i
kvalitete

www.studioleptir.hr

Web dizajn za 1999 kn

Besplatan hosting i domena Sami
odraÄavajte svoj web i uÄitedite

www.dbdesign.com.hr

AdChoices



Moja preporuka:



amazon.com
and you're done.™

[The Definitive Guide to HTML5](#)
Adam Freeman
New \$25.10

[Web Development and Design Foundation...](#)
Terry Felke-Morris...
New \$86.49

[Learning PHP, MySQL, JavaScript, and...](#)
Robin Nixon
New \$21.09

[Learning Web Design](#)
Jennifer Niederst
...
New \$26.36

[JavaScript & jQuery](#)
David Sawyer McFar...
New \$22.33

[Head First HTML and CSS](#)
Elisabeth Robson,
...
New \$21.09

Privacy Information

Savjet:

Svaki ispravan (X)HTML dokument napisan prema web standardima treba imati odvojen sadržaj (strukturu) od prezentacije (stila). Znači, prvo se treba napisati XHTML kod (naslovi, paragrafi, slike, odjeljci, forme i sl...), a zatim se taj dokument treba povezati sa CSS dokumentom u kojem su definirani stilovi. Postoji nekoliko bitnih razloga zašto je dobro odvajati strukturu od stila:

1. pregledniji dokument
2. jednostavnije, brže i sigurnije održavanje i XHTML i CSS dokumenta
3. jednostavnija prenamjena web dokumenta za različite tipove preglednika odnosno uređaja (npr. *handheld* i mobilni uređaji, audio preglednici, printeri, TV i sl.)

Kako izgleda CSS dokument?

Ako ste mislili da je CSS dokument napisan u nekom teškom programskom jeziku ili je jednostavno nacrtan u nekom grafičkom programu (!!) – jako se varate!!! CSS dokument je najobičniji tekstualni dokument koji sadrži instrukcije/naredbe za web preglednike, a koje definiraju stilove za određene HTML elemente na web stranici. CSS dokument izgleda ovako:

```
body {
    margin:0;
    font-family:"Trebuchet MS";
    font-size:12px;
}
a {
    text-decoration:none;
    background-color:red;
    color:white;
}
h1{
    margin:0;
    font-weight:bold;
    font-size:20px;
}
p {
    color:#666666;
    font-size:11px;
}
```

CSS

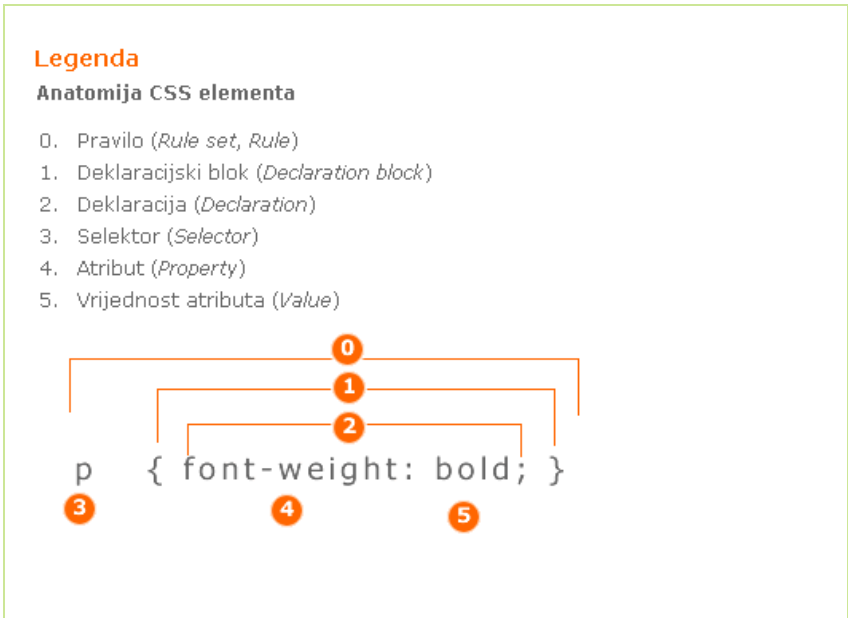
A, što znače ove stilovi ili instrukcije? Krenimo redom:

1. **Selektor *body* govori web pregledniku sljedeće:**
 - sve margine na stranici trebaju biti 0px
 - pismo treba biti napisano u Trebuchet MS fontu
 - veličina pisma treba biti 12px
2. **Selektor *a* govori web pregledniku sljedeće:**
 - linkovi ne smiju biti podcrtani (nemaju *underline*)
 - boja pozadine linka treba biti crvena
 - boja teksta linka treba biti bijela
3. **Selektor *h1* govori web pregledniku sljedeće:**
 - sve margine oko h1 elementa trebaju biti 0px
 - debljina slova treba biti podebljana
 - veličina pisma treba biti 20px
4. **Selektor *p* govori web pregledniku sljedeće:**
 - boja teksta cijelog odlomka treba biti siva
 - veličina pisma treba biti 11px

2) Anatomija CSS elementa

Kao što sam već rekao, CSS dokumenti su tekstualni dokumenti koji sadrže instrukcije odnosno naredbe koje govore web pregledniku kako treba stilski oblikovati (formatirati) i prikazati određene HTML elemente. CSS element se sastoji od: (slika 1.)

1. **deklaracijski blok**
 - kod koji dolazi iza selektora uključuje sve mogućnosti oblikovanja koje se primjenjuju na selektor; blok počinje sa otvorenom vitičastom zagradom ({) i završava sa zatvorenom vitičastom zagradom (}
2. **deklaracija**
 - između otvorene i zatvorene vitičaste zagrade deklaracijskog bloka dolazi jedna ili više deklaracija ili instrukcija za oblikovanje. Svaka deklaracija ima dva dijela: atribut i vrijednost, te završava znakom točka-zarez (;)
3. **selektor**
 - govori web pregledniku koji element na web stranici treba oblikovati (npr, naslov, odlomak, link, slika, tablica, forma i sl...)
4. **atribut**
 - CSS nudi mnogo različitih mogućnosti oblikovanja koje se zovu atributi. Atributi definiraju određeni stilski efekt.
5. **vrijednost atributa**
 - definira kako će se oblikovati određeni atribut; različiti atributi zahtijevaju različite tipove vrijednosti, npr.: mjerne duljine > 12px, 2em ili 1in; boja > *black* ili #000000; URL > slike/slika-1.jpg) ili specifična ključna riječ > *top*, *center*, *left*)



Slika 1. Anatomija CSS elementa

 [Na vrh](#)

3) Selektori, atributi i vrijednosti

Selektor govori web pregledniku koji HTML element na web stranici treba oblikovati (npr, naslov, odlomak, link, slika, tablica, forma i sl...). Svaki selektor može imati više atributa, a svaki atribut unutar selektora može imati nezavisne vrijednosti. Atribut i njegova vrijednost su odvojeni dvotočkom (:) i nalaze se unutar vitičaste zagrade. Iza vrijednosti dolazi točka-zarez (;). Ako postoji više atributa u

deklaraciji isti su odvojeni znakom točka-zarez (;). Ako se vrijednost sastoji od više riječi sve se treba staviti pod navodne znakove.

U ovom primjeru selektor je *p*, a u CSS dokumentu to se piše ovako:

```
p {
  font-weight:bold;
  font-size:12px;font-family: "Trebuchet MS", Verdana, Arial, serif;
}
```

CSS

U HTML kodu izgleda ovako:

```
<p>
Neki tekst napisan unutar paragrafa koji ima određeni stil.
</p>
```

HTML

U web pregledniku to izgleda ovako napisano:

Neki tekst napisan unutar paragrafa koji ima određeni stil.

Ovaj stil znači da će svi odlomci biti podebljani u veličini pisma 12px i da će biti napisani u pismu Trebuchet MS, Verdana ili Arial (ovisno o tome koje je pismo instalirano na korisnikovom računalu). Vjerojatno ste do sada primjetili da svaki atribut stila pišem u novom redu gdje je svaki red uvučen...Zašto to radim? Isključivo radi bolje preglednosti i čitljivosti stilova! Međutim, stilove je moguće pisati i u jednom redu, ako to želite. Na taj način, gornji stil možete napisati i ovako:

```
p {font-weight:bold;font-size:12px;font-family:"Trebuchet MS",
Verdana, Arial, serif;}
```

CSS

Grupiranje selektora

Selektore je moguće grupirati. Npr. ako želimo svim paragrafima (p), linkovima (a) i naslovima (h1) pridodati iste atribute to ćemo napraviti na slijedeći način:

```
p, a, h1 {
  color: #FF0000;
  font-family: "Trebuchet MS", Verdana, Arial, serif;
}
```

CSS

U ovom primjeru smo tri selektora odvojili zarezom i pridodali im iste atribute – boju i pismo. Na ovaj način možemo povezati mnogo selektora čime smanjujemo potrebu za pisanjem više stilova za svaki element posebno.

Vrste selektora

Postoji više vrsta selektora:

1. **selektor elementa (element type selector)**
 - *p, h1, div, em, a, img...*
 - primjer: *p {font-size:12px}* > selektor *p* određuje stil za sve odlomke
2. **univerzalni selektor (universal selector)**
 - *** (zvjezdica)
 - primjer: ** {font-size:12px}* > selektor *** određuje stil za bilo koji odnosno sve elemente
3. **selektor klase (class selector)**
 - *.prvi*
 - primjer: *.prvi {font-size:12px}* > selektor *.prvi* određuje stil za sve elemente koji imaju vrijednost atributa *.prvi*; na web stranici može biti neograničen broj elemenata koji imaju definiranu ovu vrijednost atributa
4. **selektor ID-a (ID selector)**

- `#navigacija`

- primjer: `#navigacija {font-size:12px}` > selektor `#prvi` određuje stil za element koji ima specifičnu vrijednost atributa `#prvi`, a s obzirom da ID atributi moraju imati jedinstvene vrijednosti, ID selektor ne može određivati više od jednog elementa na web stranici

5. **selektor potomka (descendant selector)**

- `ul li`

- primjer: `ul li {font-size:12px}` > selektor `ul li` određuje stil za sve HTML elemente koji su potomci ili djeca (descendant) `ul` elementa koji je njihov predak ili roditelj (ancestor)

6. **selektor djece (child selector)**

- `ul>li`

- primjer: `ul>li {font-size:12px}` > selektor `ul>li` određuje stil za sve li elemente koji su najbliži potomci ili djeca (descendant) `ul` elementa koji je njihov predak ili roditelj (ancestor)

7. **najbliži susjedni selektor (adjacent sibling selector)**

- `h1+p`

- primjer: `h1+p {font-size:12px}` > selektor `h1+p` određuje stil za sve p elemente koji su najbliži susjedni potomci ili djeca (descendant) `h1` elementa koji je njihov predak ili roditelj (ancestor)

8. **pseudo-klase (pseudo class)**

- `:link`, `:visited`, `:active`, `:hover`, `:focus`

- primjer: `a:visited {color:#ff0000;}` > selektor `a:visited` određuje stil za sve posjećene linkove na web stranici; pseudo klase definiraju stanja linkova: neposjećeni link (`link`), posjećeni link (`visited`), trenutno odabrani link (`active`) i link preko kojeg se prelazi kursorom miša (`hover`)

9. **pseudo-elementi**

- `:first-letter`, `:first-line`, `:before`, `:after`

- primjer: `p:first-letter {color:#ff0000;}` > selektor `p:first-letter` određuje stil za prvo slovo u rečenici

 [Na vrh](#)

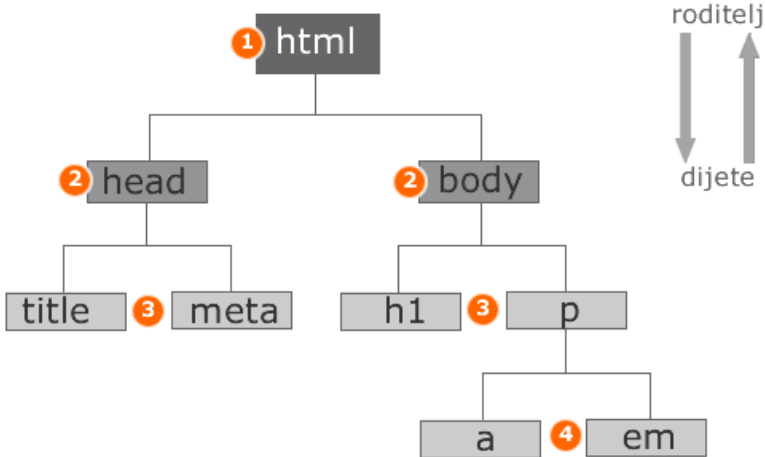
4) XHTML DOM model

Kako bismo ispravno razumijeli vrste selektora potrebno je razumijeti međusobne odnose HTML elemenata u strukturi XHTML dokumenta odnosno njegovom DOM modelu. HTML Document Object Model (HTML DOM) određuje standardan način za pristupanje i upravljanje HTML dokumentima odnosno definira objekte i svojstva HTML elemenata kao i metode pristupa istima. DOM predstavlja HTML dokument u obliku obrnutog stabla (*upside-down tree*) sa svim elementima, atributima i tekстом. Jednostavan HTML dokument se može vizualizirati na ovaj način: (slika 2.)

Legenda

DOM stablo (DOM tree) HTML dokumenta

- 1. Koriijenski element *HTML* (*root element*)
 - svi ostali elementi su djeca ovog elementa
- 2. Element predak ili roditelj (*ancestor element*)
 - element *head* je roditelj elemntima *title* i *meta*
 - element *body* je roditelj elemntima *h1* i *p*
- 3. Element potomak ili dijete (*descendant element*)
 - elementi *title* i *meta* su djeca elementa *head*
 - elementi *h1* i *p* su djeca elementa *body*
- 4. Elementi *a* i *em* su djeca *p* elementa, pradjeca *body* elementa i prapradjeca *html* elementa, ali nisu djeca *h1* elementa
 - također, elementi *a* i *em* su susjedni elementi, potomci istog roditelja (*adjacent siblings*)



Slika 2. DOM stablo (Document Object Model tree) HTML dokumenta

[Na vrh](#)

5) DIV i SPAN elementi

DIV element koji znači “odjeljak” ili sektor (eng. *division*) je *blok-level* element koji se koristi za definiranje sekcija ili odjeljaka u (X)HTML dokumentu. *DIV* element može sadržavati dodatne *DIV*-ove, tekstove, slike itd...

Kao i ostali elementi, DIV-ovi se smještaju unutar `<body>...</body>` oznaka na stranici. Kao što sam rekao, *DIV*-ovima možemo dodati klasu ili ID. Iako ovaj DIV možemo definirati upotrebom samo *div* selektora, ipak ćemo u slijedećem primjeru *DIV-u* dodati *ID* pod nazivom sadržaj:

```
<body>
<div id="sadrzaj">
Ovo je neki tekst koji se nalazi unutar <div> oznake. S obzirom da
je dio DIV-a, tekstu možemo mijenjati neke karakteristike kao što je
pozicija na stranici, margine,padding, pozadinu, rub i horizontalno
centriranje teksta.
</div>
</body>
```

HTML

Stil koji će definirati svojstva izgleda ovako:

```
#sadrzaj{
width: 700px;
margin: 10px auto;
padding: 10px;
border: 1px solid #cccccc;
background: #ffffff;
}
```

CSS

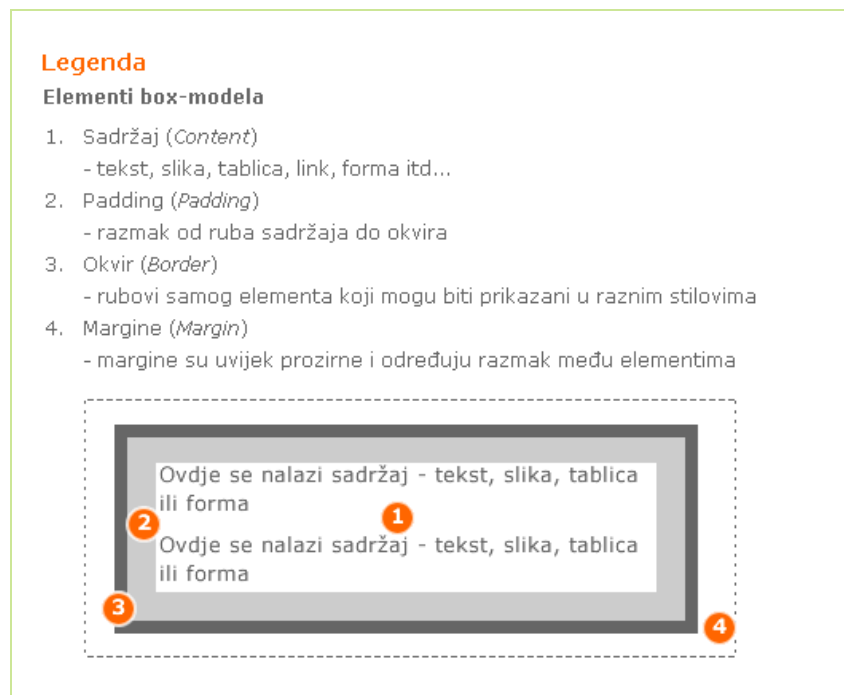
A u web pregledniku će se to prikazati ovako:

Ovo je neki tekst koji se nalazi unutar DIV oznake. S obzirom da je dio DIV-a, tekstu možemo mijenjati neke karakteristike kao što je pozicija na stranici, margine, padding, pozadinu, rub i horizontalno centriranje teksta.

 [Na vrh](#)

6) CSS Box model

Iako ovdje govorim o osnovama CSS-a, smatram da je bitno barem ukratko opisati i koncept tzv. CSS box modela koji predstavlja same temelje razumijevanja CSS-a odnosno razumijevanja koncepta pozicioniranja elemenata na web stranici, njihovih dimenzija i međusobnih odnosa. Svaki element na web stranici je pravokutnog oblika koji se sastoji od: sadržaja, padding-a, okvira (border) i margina (margin). Slijedeća slika prikazuje elemente box-modela:



Slika 3. Elementi BOX modela

Kao što u HTML-u postoje tzv block-level elementi (p, h1-h6, table, div) i inline elementi (a, br, em, img, span, strong), tako i CSS ima dva glavna tipa box-ova: block box i inline box. Block-box-ovi su generirani od strane HTML elemenata kao što su npr. p, div ili table, dok su inline-box-ovi generirani od HTML elemenata kao što su npr. br, em ili span. CSS box je uvijek pravokutnog oblika (znači ima četiri stranice, svaka pod kutom od 90°).

Svi HTML elementi mogu biti prikazani na jedan od slijedećih načina:

1. **Block (display:block)**

- block-level element zauzima cijelu širinu slobodnog (raspoloživog) prostora unutar kojeg se nalazi, i počinje uvijek u novom retku u normalnom toku HTML-a
- primjer:

```
<p>Tekst koji se nalazi unutar paragrafa koji je block-level element i koji je od svih obližnjih elemenata odvojen vertikalnim razmakom.</p>
```

```
<p>Ovo je drugi paragraf koji je također block-level element i od prvog paragrafa je odvojen vertikalnim razmakom.</p>
```

HTML

Ovaj primjer u web pregledniku izgleda ovako (napomena: namjerno sam ostavio boju pozadine kako biste jasnije vidjeli razliku u vertikalnom razmaku odnosno praznom prostoru između paragrafa):

Tekst koji se nalazi unutar paragrafa koji je block-level element i kao takav zauzima cijelu širinu slobodnog (raspoloživog) prostora. Od svih obližnjih elemenata ovaj paragraf je odvojen vertikalnim razmakom.

Ovo je drugi paragraf koji je također block-level element i od prvog paragrafa je odvojen vertikalnim razmakom. Paragraf zauzima cijelu širinu raspoloživog prostora.

2. **Inline (display:inline)**

- inline-level element zauzima samo onoliko prostora koliko mu je potrebno za prikazivanje u pregledniku unutar kojeg se nalazi, i ne prekida normalan tok HTML-a, već se nalazi u liniji npr. sa ostalim tekstom
- primjer:

```
<p>Ovo je tekst koji se nalazi u normalnom toku HTML dokumenta i sadrži jedan span element koji je inline-element, jer se isto nalazi u normalnom toku HTML dokumenta zajedno sa ostalim tekstom.</p>
```

HTML

Ovaj primjer u web pregledniku izgleda ovako (napomena: namjerno sam ostavio boju pozadine kako biste jasnije vidjeli gdje dolazi element unutar teksta):

Ovo je tekst koji se nalazi u normalnom toku HTML dokumenta i sadrži jedan **span element** koji je inline-element, jer se isto nalazi u normalnom toku HTML dokumenta zajedno sa ostalim tekstom.

3. **Not displayed (display:none)**

Ako elementu pridodamo atribut display:none isti se neće prikazati u web pregledniku. Također, svi elementi koji se nalaze unutar ovog elementa neće se prikazati.

[▲ Na vrh](#)

7) Klase i ID-ovi

Klase nam omogućavaju pridodavanje različitih stilova HTML elementima. Za razliku od ID-a, klasu možemo upotrijebiti na više mjesta unutar web stranice, a svaki element može imati više klasa koje su dovojene razmakom, npr. <p class="prvi crveni">. U sljedećem primjeru definiran je stil za paragrafe u HTML dokumentu (svaki paragraf treba imati tekst veličine 13px i u crvenoj boji):

```
p {
  font-size: 12px;
  color: #FF0000;
}
```

CSS

U XHTML kodu ćemo napisati rečenicu i omeđiti je <p>...</p> elementom koji definira paragraf:

```
<p> Ovo je prvi paragraf koji ima tekst veličine 12px i u crvenoj je boji.</p>
<p>Ovo je drugi paragraf koji isto ima tekst veličine 12px i u crvenoj je boji.</p>
```

HTML

U web pregledniku ovi paragrafi će izgledati ovako:

Ovo je prvi paragraf koji ima tekst veličine 12px i u crvenoj je boji.
Ovo je drugi paragraf koji isto ima tekst veličine 12px i u crvenoj je boji.

Međutim, ukoliko na web stranici imamo više paragrafa, a želimo da je prvi paragraf definiran drugačijim stilom, onda ćemo upotrijebiti klasu za prvi paragraf koja će definirati druga svojstva, npr.:

```
p.uvod {
  font-size: 12px;
  color: #0000FF;
}
```

CSS


```
font-weight: bold;
}
```

...a za sve ostale paragrafe već imamo definiran stil:

```
p {
  font-size: 12px;
  color: #FF0000;
}
```

CSS

U XHTML kodu ćemo <p> elementu prvog paragrafa dodijeliti klasu .uvod:

```
<p class="uvod">Ovaj paragraf je uvodni, ima tekst veličine
12px, podebljan je i u plavoj je boji.</p>

<p> Ovo je drugi paragraf koji ima tekst veličine 12px i u
crvenoj je boji.</p>
```

HTML

U web pregledniku ovi paragrafi će izgledati ovako:

Ovaj paragraf je uvodni, ima tekst veličine 12px, podebljan je i u plavoj je boji.

Ovo je drugi paragraf koji ima tekst veličine 12px i u crvenoj je boji.

Savjet:

Važno je zapamtiti da klasa tj. selektor klase počinje sa točkom (.), a dobra je praksa svakoj klasi pridodati deskriptivan naziv tj. opisati ju prema svojstvu/značenju elementa kojeg definira. Npr.:
p.uvod {...} > dobar primjer naziva klase
p.plavi {...} > loš primjer naziva klase

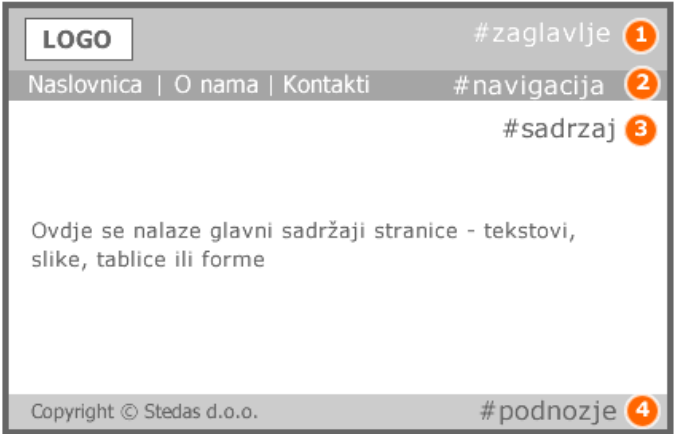
Prvi primjer (*p.uvod {...}*) je dobar zato što opisuje **značenje** uvodnog paragrafa odnosno njegovu **namjenu** na web stranici (prvi paragraf je uvijek uvodni pa smo mu zato i dodijelili klasu pod nazivom .uvod), dok drugi stil opisuje uvodni paragraf prema **boji** koja se u budućnosti može promijeniti u npr. zelenu. Da li u tom slučaju znamo da je paragraf koji ima naziv klase .crvena zapravo paragraf u zelenoj boji? Naravno da ne! Zbog toga bi trebali svakom uvodnom paragrafu ponovno mijenjati i naziv klase u .zelena, što znači dodatno izgubljeno vrijeme na ispravke.

Dok nam klase omogućavaju pridodavanje različitih stilova HTML elementima, **ID-ovi** nam služe za identifikaciju samog elementa, gdje se ID može upotrijebiti samo na jednom elementu na stranici. ID-ovi se najčešće koriste za definiranje najvažnijih strukturalnih elemenata na web stranici kao što su npr. zaglavlje (header), glavni sadržaj (content) i podnožje (footer) koji se na web stranici pojavljuju samo jednom. Njih ćemo koristiti kao <div> kontejnere koji sadrže sve ostale saržaje. Slijedeća slika prikazuje primjer jednog XHTML dokumenta koji koristi <div> elemente i određene nazive ID-ova za vizualnu strukturu: (slika 4.)

Legenda

Primjer strukture stranice prema nazivima DIV kontejnera

- 1. Zaglavlje stranice
 - <div id="zaglavlje">...</div>
 - sadrži obično logo tvrtke
- 2. Navigacija stranice
 - <div id="navigacija">...</div>
 - sadrži obično navigaciju
- 3. Sadržaj stranice
 - <div id="sadrzaj">...</div>
 - svi tekstovi i slike se nalaze unutar ovog kontejnera
- 4. Podnožje stranice
 - <div id="podnozje">...</div>
 - sadrži obično podatke o privatnosti, zaštiti i disklejmeru



Slika 4. Struktura stranice prema nazivima DIV kontejnera

U sljedećem primjeru definiran je stil za zaglavlje na web stranici (element <div> će biti širok 700px, visok 50px, okvir će biti 2px u crvenoj boji, padding 5px, a u pozadini će biti slika logotipa - logo.gif, smještena 10px od lijevog ruba i centrirana vertikalno):

```
#zaglavlje {
  width: 760px;
  height: 50px;
  border: 2px solid #ff0000;
  padding: 5px;
  background-image: url(logo.gif);
  background-repeat: no-repeat;
  background-position: 10px center;
}
```

CSS

U XHTML kodu ćemo <div> elementu dodijeliti naziv ID-a #zaglavlje:

```
<div id="zaglavlje"></div>
```

HTML

A, u web pregledniku ovaj <div>...</div> će biti prikazan ovako:



Na isti način možemo oblikovati i sve ostale DIV kontejnere bez upotrebe tablica!

Savjet:

Važno je zapamtiti da ID selektori počinju sa znakom „ljestve“ (#), a dobra je praksa svakom ID-u pridodati deskriptivan naziv tj. opisati ga prema svojstvu/značenju elementa kojeg definira. Npr.:

#sporedna-navigacija {...} > dobar primjer naziva klase

#navigacija-lijeva {...} > loš primjer naziva klase

Prvi primjer (#sporedna-navigacija {...}) je dobar zato što opisuje **značenje** sporedne navigacije

odnosno njegovu **namjenu** na web stranici (sporedna navigacija je obično lokalna navigacija koja se nalazi unutar određene kategorije na stranici, bez obzira gdje se ona nalazila na web stranici), dok drugi stil opisuje navigaciju prema njezinom smještaju na stranici koji se u budućnosti može promijeniti, npr. sporednu navigaciju možemo premjestiti sa sadašnje pozicije u lijevom stupcu na drugu poziciju u desnom stupcu.

 [Na vrh](#)

8) Svojstva teksta (*text properties*)

Pomoću svojstava za oblikovanje teksta (*text properties*) možemo npr. mijenjati vrstu pisma, veličinu slova ili dekoraciju teksta. U nastavku navodim ona svojstva koja se najčešće koriste (*font-family, font-size, text-decoration*):

1. **font-family**

- **primjer:** `font-family:Verdana, Arial, Helvetica, sans-serif;`
- **opis:** ovo svojstvo određuje vrstu pisma, i to: web preglednik će prvo prikazati tekst u Verdani, zatim, ako Verdana nije instalirana - prikazat će tekst u Arial, pa u Helvetici i na kraju (ako niti jedan od ovih pisama nije instaliran) prikazat će pismo u generičkoj sans-serif vrsti pisma
- **napomene:**
 - bitno je uvijek koristiti samo ona pisma koja se najčešće upotrebljavaju i dolaze već instalirana sa određenim operativnim sustavom;
 - ukoliko pismo sadrži u svom nazivu više riječi, tada je potrebno pismo staviti pod navodne znakove, npr. `font-family:"Trebuchet MS"`
 - **korisno:** pogledajte listu sigurnih i standardnih vrsta pisama za web u operativnom sustavu Windows > <http://www.ampsoft.net/webdesign-I/WindowsMacFonts.html>

2. **font-size**

- **primjer:** `font-size: 12px;`
- **opis:** ovo svojstvo određuje veličinu teksta, i to u veličini 12px
- **napomene:**
 - osim korištenja piksela (px) kao mjerne jedinice za određivanje veličine pisma, moguće je koristiti i ostale veličine odnosno mjere, npr.: `em`, `ex` ili `%`;
 - kada koristite npr. naslov (heading) ne upotrebljavajte selektor za paragraf (p) u većoj veličini, već koristite `h1` do `h6` selektore za definiranje veličine naslova;

3. **text-decoration**

- **primjer:** `text-decoration: underline;`
- **opis:** ovo svojstvo određuje dekoraciju na tekstu i to podcrtani tekst
- **napomene:**
 - najčešće se koristi kod pseudo-klasa odnosno linkova za određena stanja kao što su `a:link`, `a:visited` ili `a:hover`, npr. ako ne želite da su neposjećeni linkovi podcrtani (underline), tada ćete definirati slijedeći stil: `a:link { text-decoration:none; }`

 [Na vrh](#)

9) Margine i padding

CSS svojstva - margine i padding se koriste za razmake između elemenata. Margina je razmak (prostor) izvan elementa, a padding je razmak (prostor) unutar elementa. (slika 5.)

Legenda

Margine i padding

- 1. Sadržaj (Content)
- 2. Padding (Padding)
- razmak od ruba sadržaja do okvira
- 3. Margina (Margin)
- margine su uvijek prozirne i određuju razmak među elementima



Slika 5. Prikaz margina i paddinga

U nastavku prikaz dva div elementa i njihov međusobni odnos:

Primjer:

```
#div-1 {  
  backckground-color: #cccccc;  
  margin-right: 15px;  
  padding: 10px;  
  float:left;  
  width:200px;  
}
```

CSS

Opis: ovaj stil određuje slijedeća svojstva za **div-1** element: boja pozadine treba biti svijetlo-siva, margina s desne strane 15px, padding 10px sa sve četiri strane (gore, dolje, lijevo i desno), pozicija *div-1* elementa treba biti lijevo u odnosu na *div-2* element, a širina *div-1* elementa 200px.

```
#div-2 {  
  background-color: #666666;  
  margin-left: 0;  
  margin-right:0;  
  padding: 0;  
  color:#FFFFFF;  
  width:200px;  
}
```

CSS

Opis: ovaj stil određuje slijedeća svojstva za **div-2** element: boja pozadine treba biti tamno-siva, margina s lijeve i desne strane 0px, padding 0px sa sve četiri strane (gore, dolje, lijevo i desno), boja teksta bijela, pozicija *div-2* elementa treba biti isto lijevo odmah iza *div-1* elementa, a širina *div-2* elementa 200px.

U web pregledniku to izgleda ovako:

#kontejner

div-1

Sadržaj koji se nalazi unutar elementa div-1

Sadržaj koji se nalazi unutar elementa div-2

div-2

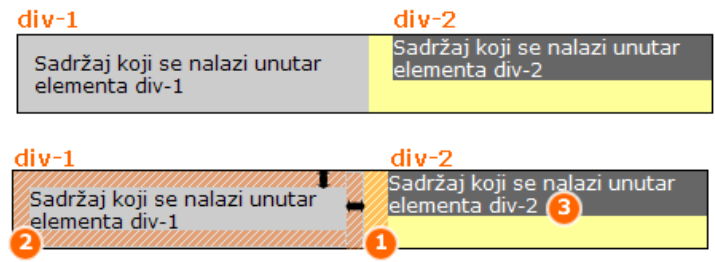
Detaljan opis i prikaz *margina* i *paddinga* pogledajte na slici 6.

Legenda

Margine i padding (primjer)

Prva slika prikazuje dva div elementa i određena svojstva za margine i padding; druga slika prikazuje gdje se zapravo nalaze margine i padding kod ova dva div elemenata:

- 1. Margina za div-1 element
 - div-1 ima marginu od 15px (to je ujedno horizontalni razmak između ova dva elementa)
- 2. Padding za div-1 element
 - div-1 ima razmak od ruba sadržaja do okvira (10px sa sve četiri strane)
- 3. Padding za div-2 element
 - div-2 nema razmak od ruba sadržaja do okvira (0px sa sve četiri strane)



Slika 6. Primjer prikaza margina i paddinga

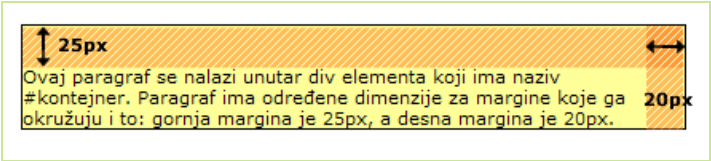
Margine i padding svojstva se mogu također definirati za svaku stranu elementa posebno. Npr, možemo odrediti samo gornju marginu (*margin-top*) i samo desnu marginu (*margin-right*) elementa, npr.:

```
#p {
  margin-top: 25px;
  margin-right: 20px;
}
```

CSS

Opis: ovaj stil određuje gornju (25px) i desnu marginu (20px) paragrafa koji se nalazi unutar div elementa pod nazivom #kontejner. S obzirom da donja i lijeva margina nisu definirane, njihova preddefinirana vrijednost iznosi 0px.

U web pregledniku to izgleda ovako:



Također, možemo napisati tzv. skraćenu verziju za svojstva margina i paddinga (*shorthand properties*), npr.:

Ovaj stil...

```
body {
  margin-top:10px;
  margin-right: 10px;
  margin-bottom: 10px;
  margin-left: 10px;
  padding-top:0;
  padding-right:0;
  padding-bottom:0;
  padding-left:0;
}
```

CSS

... se može napisati i ovako:

```
body {
  margin: 10px 10px 10px 10px;
  padding: 0 0 0 0;
}
```

CSS

...a, ne biste vjerovali ;-)...može se napisati i u sasvim skraćenoj varijanti:

```
body {
  margin: 10px;
  padding: 0;
}
```

CSS

Zapamtite:

1. **Ako neka svojstva pišete u skraćenoj varijanti (npr. za margine ili padding), bitno je** zapamtiti njihov redosljed u skraćenoj verziji. Npr. za ovo svojstvo *margin: 10px 5px 35px 0px;* vrijednosti se odnose redom na: gornju, desnu, donju i lijevu marginu (prema smjeru kazaljke na satu)
2. **Za sve vrijednosti koje imaju 0 nije potrebno pisati mjeru u px, % ili em**

 [Na vrh](#)

10) Načini povezivanja CSS-a sa HTML dokumentom

HTML dokument koji u sebi nema definirane stilove neće vizualno oblikovati elemente na stranici. Međutim, ako ga povežemo sa CSS dokumentom, HTML elementi će biti vizualno oblikovani odnosno stilizirani. Stilska lista može biti povezana sa HTML dokumentom na dva načina ovisno o njenoj lokaciji: (slika 7.)

1. **interno**
 - CSS stilovi su definirani unutar `<style>...</style>` elementa i nalaze se na samoj web stranici unutar `<head>...</head>` elementa. Ovi stilovi se isključivo odnose na elemente na samoj web stranici
2. **eksterno**
 - CSS stilovi se nalaze u odvojenom CSS dokumentu koji je sa HTML dokumentom povezan putem `<link>` elementa koji se nalazi unutar `<head>` elementa. Ovo je preferirani način povezivanja!

Legenda

Načini povezivanja CSS-a sa HTML dokumentom

1. **Interno** - CSS stilovi se nalaze na web stranici unutar `<style>` elementa koji se nalazi obavezno u `<head>` elementu
2. **Eksterno** - CSS stilovi se nalaze u odvojenom CSS dokumentu, a povezani su sa HTML dokumentom putem `<link>` elementa

Stil u web stranici (interni)

```
<html>
<head>
<title>CSS vježbe</title>
<style type="text/css">
  h1 {
    color: red;
    margin: 5px 5px 25px 5px;
  }
  p {
    background-color: #FFFF00;
    margin-left: 10px;
  }
</style>
</head>
<body>Neki sadržaj...</body>
</html>
```

Stil u odvojenom CSS dokumentu (eksterni)

```
<html>
<head>
<title>CSS vježbe</title>
<link rel="stylesheet" type="text/css" href="css/stil.css" />
</head>
<body>Neki sadržaj...</body>
</html>
```

Slika 7. Načini povezivanja CSS stilova sa HTML dokumentom

Zapamtite!

1. **Trenutna podržana verzija CSS-a je CSS 2.1**
2. **Koristite CSS-stilske liste kada želite vizualno oblikovati (formatirati) elemente na stranici i želite imati potpunu kontrolu nad njima**
3. **CSS omogućava odvajanje prezentacije od (X)HTML dokumenta i na taj način pomaže pri jednostavnijem, bržem i sigurnijem održavanju stranica**
4. **CSS smanjuje *bandwidth* i veličinu web dokumenta i na taj način omogućava brže preuzimanje (*download*) (X)HTML dokumenta**
5. **CSS omogućava definiranje različitih stilova za različite tipove izlaznih medija, npr. poseban stil za prikaz stranice u web pregledniku, a poseban stil za ispis stranice na pisaču**
6. **CSS može biti povezan sa HTML dokumentom na dva načina: interno (`<style>`) i eksterno (`<link>`)**

11) Spremanje CSS dokumenta (izbornik *File>Save*)

Nakon što ste kreirali stilove u eksternom CSS dokumentu, takav dokument je potrebno spremiti sa ekstenzijom `.css`, npr. `stilovi.css`. Na taj način će web preglednik znati da je riječ o dokumentu koji sadrži stilove za određenu stranicu. Ukoliko ste napravili interni stil (u `<head>` elementu) isti se ne sprema kao `stilovi.css` već će biti uključen u samu stranicu koja će se spremiti kao npr. `index.html`

Kako nazvati CSS dokument?

Kao što sam rekao u prošlom primjeru svaki eksterni CSS dokument treba spremiti sa ekstenzijom `.css`, a sam naziv dokumenta napisati malim slovima. To je potrebno napraviti zato što su neki web serveri (npr. *Linux*) osjetljivi na veličinu slova u nazivu dokumenta. Npr. ako u adresnu traku web preglednika napišete `http://www.stedas.hr/STILOVI.CSS`, a naziv dokumenta je zapravo `stilovi.css` isti se neće otvoriti i dobit ćete grešku `"Page not found"`. Ukoliko je potrebno naziv dokumenta napisati sa više riječi iste treba odvojiti ili znakom "minus" (`-`) ili znakom "donja crtica" (`_`), npr. `stilovi-print.css`. Specijalni znakovi i naša slova nisu dozvoljeni u nazivima CSS dokumenata, npr. `$`, `?`, `#`, `/`, `:`, `č`, `ć`, `š`, `ž`, `đ` itd...

 [Na vrh](#)

12) Provjera ispravnosti CSS koda (validacija CSS koda)

Prije nego što CSS dokument prebacimo na udaljeno računalo odnosno web server i na taj način omogućimo ispravno prikazivanje XHTML dokumenta, trebamo provjeriti i testirati je li sve ispravno napisano u CSS kodu. To možemo napraviti provjerom u CSS validatoru koji se nalazi na:

<http://jigsaw.w3.org/css-validator/>

Na ovoj stranici postoje tri mogućnosti validacije CSS koda:

1. **Prema URL adresi (By URI)**
 - upisati cijelu weba dresu uključujući i sam CSS dokument, npr. www.stedas.hr/stil.css
2. **Podizanjem CSS dokumenta na web server (By file upload)**
 - odabrati CSS dokument na lokalnom računalu
3. **Kopiranjem cijelog CSS koda u polje za unos (By direct input)**

Nažalost, eksterni CSS kod se ne može jednostavno pogledati u web pregledniku kao što možemo pogledati npr. XHTML kod. Internet Explorer ne dozvoljava mogućnost uvida u eksternu stilsku listu (stilovi se mogu vidjeti jedino ako su interno napisani na web stranici – u zaglavlju HTML dokumenta ili *inline* - u samom sadržaju).

Međutim, web preglednik Firefox u sklopu add-in programa *Web developer* (<http://chrispederick.com/work/web-developer/>) nudi mogućnost uvida u CSS dokument web stranice kao i mnogo drugih korisnih alata za razvoj, provjeru i editiranje koda.

Savjet:

Ako želite da vaš CSS dokument bude sukladan najnovijim web standardima, kompatibilan sa najnovijim verzijama web preglednika (npr. *Internet Explorer 7*, *Firefox 2* ili *Opera 9*), ispravno validiran u programima za provjeru ispravnosti napisanog koda (*W3C Validator*) i na kraju ispravno semantički napisan – pridržavajte se ovih savjeta i napomena:

1. koristiti CSS-stilske liste kada želite vizualno oblikovati (formatirati) elemente na stranici i želite imati potpunu kontrolu nad njima
2. svi elementi trebaju biti pozicionirani i oblikovani koristeći samo CSS stilove (ne koristiti tablice za vizualnu strukturu već samo za tabularne podatke kao što su npr. tablični proračuni)
3. svi elementi trebaju biti oblikovani koristeći CSS stilove (ne koristiti zastarjele HTML elemente kao što su ``, `` ili `<u>`)
4. svi stilovi trebaju biti napisani u eksternom CSS dokumentu koji je putem `<link>` elementa povezan sa HTML dokumentom
5. ispravno koristiti klase (`.class`) i ID-ove (`#ID`) u HTML dokumentu
 - njihova imena trebaju opisivati funkciju ili značenje, a ne izgled; ako naziv klase ili ID-a ima više riječi onda iste treba odvojiti sa znakom minus (`-`) ili donja crtica (`_`) radi bolje čitljivosti
6. svi selektori, atributi i vrijednosti trebaju biti napisani malim slovima i ne smiju sadržavati naša slova i

specijalne znakove (*npr.* č, ć, š, đ, ž, &, /, %...)

7. stilsko pravilo (*rule*) pisati u ovom formatu:
selektor { atribut1:vrijednost1; }, npr. p { font-weight:bold; }

8. ako različiti elementi imaju iste stilove, tada treba njihove selektore grupirati:
selektor1, selektor2 { atribut1:vrijednost1; }, npr. h1, p {color:#ff0000; }

Autorska prava

Svi članci objavljeni na web stranicama www.web-upotrebljivost.com, uključujući tekstualne, grafičke, programske i ostale materijale, zaštićeni su Zakonom o autorskim pravima.

Distribucija odnosno korištenje članka je dozvoljena samo pod slijedećim uvjetima:

- Sadržaj članka ne smije se mijenjati ili nadopunjavati
- Potrebno je navesti izvor prema originalnom sadržaju na ovim stranicama sa direktnim linkom koji se nalazi ispod kopiranog sadržaja i to: www.web-upotrebljivost.com, Dario Šuveljak, Zagreb

 [Na vrh](#)



5 people liked this.



DISQUS

Add New Comment

[Login](#)



Please wait...

Showing 6 comments

Sort by newest first



Sostaric Katarina

Odlicna!

[8 months ago](#)

[1 Like](#)

[Like](#)

[Reply](#)



gost

Zanimljiva objašnjenja, hvala.

[1 year ago](#)

[Like](#)

[Reply](#)



Guest

glupost...

[1 year ago](#)

[Like](#)

[Reply](#)



Fbfbdbfdbdbfdbfb

rasjecu te trimenom

[11 months ago](#)

[in reply to Guest](#)

[Like](#)

[Reply](#)

[Damir](#)



Zahvaljujem se na ovom tutorijalu, kad sam napokon odlučio naučiti css ovo mi je puno pomoglo. Hvala još jednom!!!

[1 year ago](#) [1 Like](#) [Like](#) [Reply](#)



Ivan Stipić

Odlično objašnjeno! Kad još jednom pročitam i i slegnu mi se sve informacije, bacam se u prve korake programiranja! ;) Tnx

[6 months ago](#) [in reply to Damir](#) [Like](#) [Reply](#)
[1 Like](#)

[M Subscribe by email](#) [S RSS](#)

Trackback URL

[blog comments powered by DISQUS](#)

Pitanja u vezi analize:

Saznajte sve o analizi upotrebljivosti web stranice.

- 1. [Zašto analiza upotrebljivosti?](#)
- 2. [Kome je namijenjena?](#)
- 3. [Što dobivam u analizi?](#)
- 4. [Metodologija za izradu analize?](#)
- 5. [Koliko traje izrada analize?](#)
- 6. [Koliko košta?](#)
- 7. [Kako naručiti analizu?](#)

Želim naručiti:

- 1. [Analizu web upotrebljivosti](#)
- 2. [Izradu ili redizajn web stranica](#)
- 3. Izradu infografike

Ako imate dodatnih pitanja u vezi analize web stranice ili želite naručiti neku uslugu slobodno me [kontaktirajte](#)

Izdvojeno

- 1. [Prezentacija: Klikom do frustracije \(.ppt\)](#)
- 2. [Infografika: Anatomija upotrebljive web stranice](#)
- 3. [Primjeri analize web upotrebljivosti](#)
- 4. [Dreamweaver: Izrada web stranice](#)
- 5. [Ukratko o meni](#)
- 6. [Kontakt](#)
- 7. [Novosti](#)

