

# Slike i CSS

U jednoj od prethodnih jedinica, kada smo se bavili samo HTML jezikom, videli smo kako se mogu postaviti slike na naše stranice. U ovoj jedinici ćemo upoznati tehnike kojima pomoću CSS opisa utičemo na te slike (img tagove).

## Veličina slika

Koristeći CSS možemo uticati na širinu i visinu slika, kao što to činimo sa drugim elementima. Postavljanje dimenzija kroz CSS je zgodno, jer će se prilikom učitavanja stranice odmah ostaviti dovoljno mesta za sliku, čak i da joj je potrebno više vremena za učitavanje.

Već smo u sedmoj jedinici rekli, ali sada ću ponoviti: nikako ne menjajte veličinu slike kroz CSS ili HTML. Ako je potrebna slika 300x200px, spremite sliku 300x200px u Photoshopu ili sličnom alatu, pa je onda kao takvu unesite na stranu. Nemojte unositi 3000x2000px i zatim je CSS-om smanjivati na 300x200px.

Preporuka je da koristite nekoliko dimenzija slika po celom sajtu. Na primer, odredite jednu veličinu za slike proizvoda, drugu veličinu za slike zaposlenih i sl.

```
img.proizvod {  
    width:250px;  
    height:100px;  
}
```

U ovom primeru, sve slike (img tagovi) sa CLASS vrednošću *proizvod* će imati dimenzije 250x100px. Samo je potrebno da obratimo pažnju da sve slike stvarno i budu 250x100px.

Ovaj primer možemo kasnije proširiti tako što postavimo i veću varijantu slika proizvoda i linkovati manje varijante ka većoj, ali tu je bitno da ćemo imati po dva primerka iste slike (jednu manju i jednu veću).

Shodno tome, uvek odredite nekoliko veličina i kreirajte nekoliko klasa za njih, koje će se pojavljivati svuda po sajtu. Naravno, osim ako zbog specifičnosti sajta ovo nije moguće.

## Poravnanje slika

Kada unesemo sliku na stranicu, ona se ponaša kao i svi inline elementi. Postavlja se kao karakter teksta, kao na primeru:



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum nec libero nec massa pulvinar vestibulum ut sed sem. Integer augue felis, rhoncus in bibendum at, lobortis imperdiet magna. Nam id velit ipsum. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus at laoreet tellus. Fusce vel elit vitae nulla cursus tincidunt sit amet vel diam. Curabitur ornare tempus lacinia. Nunc mollis volutpat odio sed facilisis. Proin porttitor tempor mi eu tincidunt. Maecenas felis odio, gravida non tincidunt nec, facilisis a augue. Nunc vitae odio lectus, ac facilisis tellus. Vestibulum tristique volutpat rhoncus. Donec rutrum magna molestie lectus rutrum fermentum. Nunc ut leo sed neque porta commodo. Ut augue nisl, accumsan in pretium in, gravida ac ligula. Integer imperdiet elementum magna at sollicitudin.

*Default pozicija slike – postavljena na početku teksta*

Ipak, CSS-om možemo uticati na nju i njen odnos sa drugim elementima.

### Float levo – desno

Umesto korišćenja ranije pomenutog align atributa nad slikama, možemo koristeći float svojstvo rasporediti slike na stranici. Jedan od načina je da samoj klasi (koju smo pomenuli malopre) dodamo *float:left* ili *right*, ili ipak možemo kreirati novu klasu, na primer:

```
.levo {  
    float:left;  
}
```

I dodati je po potrebi. Na primer, u HTML-u:

```

```

Na ovom img tagu vidimo dve klase. Jedna je *proizvodMala*, koja će postaviti sve parametre slike, dok će klasa *levo* samo postaviti float levo.

### Centriranje

Ukoliko želimo da centriramo sliku, možemo koristiti dva načina. Jedan je upotreba text-align svojstva nad okružujućem elementu ukoliko je slika sama u redu teksta. Onda će se, pošto je inline element, ponašati kao jedno slovo teksta i biće na sredini.

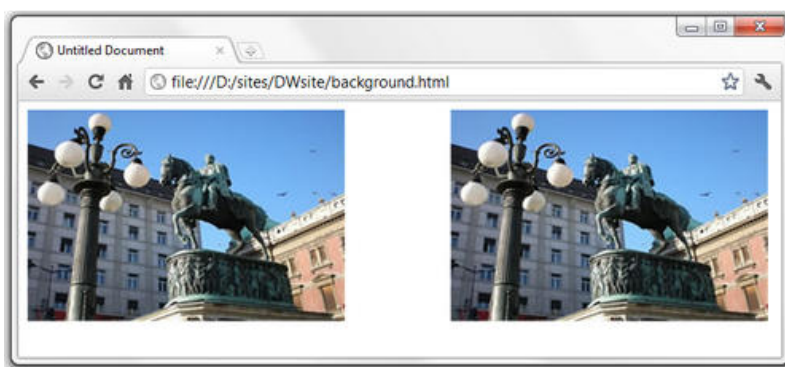
Drugi način je da slici, ukoliko smo već definisali širinu postavimo levu i desnu marginu na vrednost *auto*, ali je tada moramo pretvoriti u *block*. Dakle, CSS bi izgledao ovako:

```
.slikaCent {  
    display:block;  
    margin-left:auto;  
    margin-right:auto;  
}
```

Možda deluje komplikovano, ali ako je data širina block elementu, postavljanje margina levo i desno na auto vrednost naterace ga da bude na sredini roditeljskog elementa. Ova tehnika se može koristiti i za druge elemente, ne samo za slike.

## Pozadinske slike

Praktično svaki element na strani može imati pozadinu. Sama pozadina može biti boja ili slika, čak i kombinacija ta dva. Slike u pozadini se razlikuju od normalno unetih slika. U HTML-u se **uopšte ne pojavljuju**, već se putem CSS-a vezuju za određeni elemenat. Pogledaćemo na primeru.



*Normalno uneta slika putem img taga (levo) i pozadinska slika (desno)*

Deo HTML strane prikazuje dve slike. Leva je postavljena putem normalnog img taga, dok je desna postavljena kao pozadina DIV-a. Na prvi pogleda nema nikakve razlike. Ipak, u levom primeru kreirali smo DIV i u njega uneli sliku. Ceo kod izgleda ovako:

```
<div>
  
</div>
```

DIV nema širinu ni visinu, ali te vrednosti dobija od slike, tj. proširuje se kako bi obuhvatio sliku.

Sa druge strane, kod pozadinske slike, DIV se ne prilagođava slici koja je u pozadini. Njeni atributi nemaju nikakav uticaj na DIV. U našem primeru, na desnoj slici, postavljen je sledeći kod:

```
<div class="pozadinskaSlika">

</div>
```

ali moramo dodati i CSS:

```
.pozadinskaSlika {
  background:url(images/galerija_spomenik.jpg) left top no-repeat;
```

```
width:300px;  
height:200px;  
}
```

Primetićemo da je u HTML kodu DIV prazan i nema nikakav sadržaj, dok samu sliku dodajemo kroz CSS (detaljima ćemo se pozabaviti odmah u nastavku). Pored toga, moramo upisati širinu i visinu, jer će dimenzije, u suprotnom, zavisiti od sadržaja DIV-a, tj. u ovom primeru, ako bismo izbacili deklaracije za širinu/visinu, slika se uopšte ne bi videla (jer je DIV element prazan i nema dimenzije).

U primeru iznad smo uneli neke nove detalje bez razjašnjenja. Hajde sada da vidimo koja CSS svojstva utiču na pozadinske slike i kako se postavljaju.

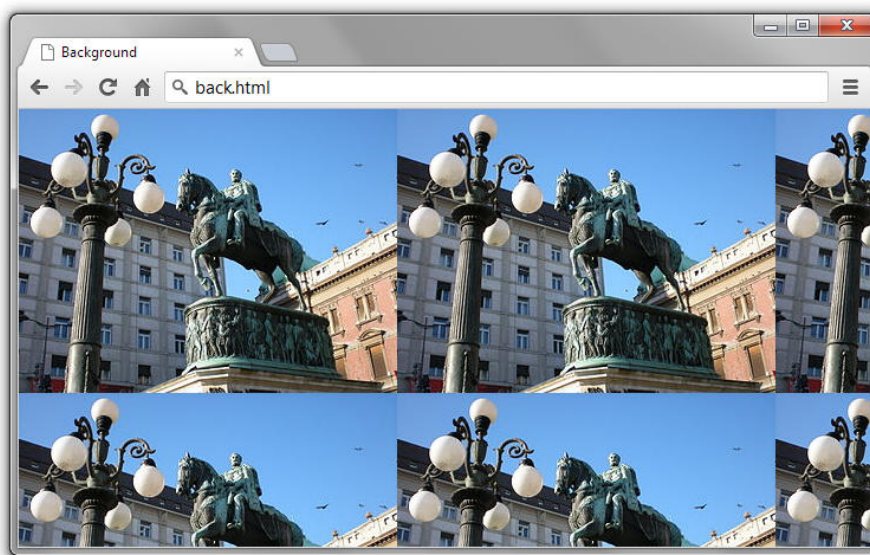
## background-image

Ovo svojstvo je najbitnije kod pozadinskih slika, jer pomoću njegove vrednosti pozivamo sam fajl slike. Dakle, ponavljam, slika u pozadini se ne učitava kroz HTML, već kroz CSS opis.

```
body {  
    background-image: url("images/back.png");  
}
```

(Vrednost *background-image* svojstva je postavljena tako da pišemo url, otvaramo zagradu i pod navodnicima postavljamo relativnu ili apsolutnu putanju do slike).

Ukoliko postavimo samo ovo svojstvo, po default vrednostima, slika će se ponavljati i po x i po y osi beskonačno. Ovo će biti vidljivo, ukoliko je slika manja od elementa nad kojim je postavljena.



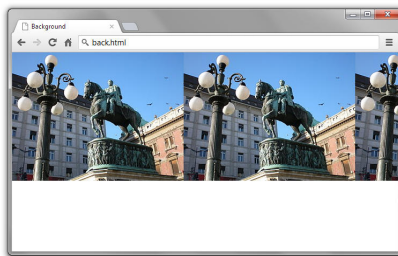
*Pozadinska slika postavljena na body elementu*

## background-repeat

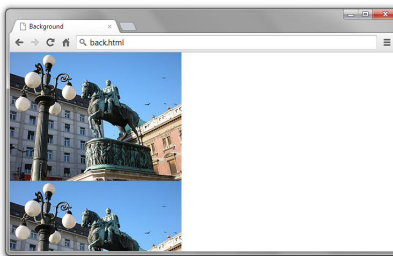
Slika u pozadini se može ponavljati na više načina. Default vrednost je *repeat* koja ponavlja sliku beskonačno po obema osama (kao na primeru iznad). Ukoliko postavimo vrednost *repeat-x*, odnosno *repeat-y* slika će se ponavljati samo po x, odnosno y osi. Naposljetku, ukoliko postavimo *no-repeat*, slika se neće ponavljati i prikazaće se samo jednom.

```
body {  
    background-image: url("images/back.png");  
    background-repeat: repeat-x;  
}
```

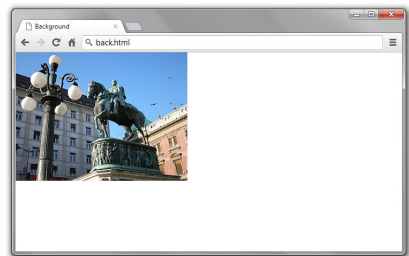
Ovaj kod će postaviti pozadinsku sliku i ponavljati je po horizontali (x osa). Pogledajmo taj i ostale varijante u browseru:



repeat-x



repeat-y



no-repeat

*Moguće varijante ponavljanja pozadinske slike (pored osnovne)*

## background-attachment

Ukoliko želimo, možemo fiksirati pozadinsku sliku tako da stoji u mestu (kao zakačena za browser) dok se ostali elementi skroluju. U tom kontekstu koristimo *background-attachment* svojstvo. Može imati vrednosti *fixed* i *scroll* (koja je default).

```
body {  
    background-image: url("images/back.png");  
    background-attachment: scroll;  
}
```

## background-position

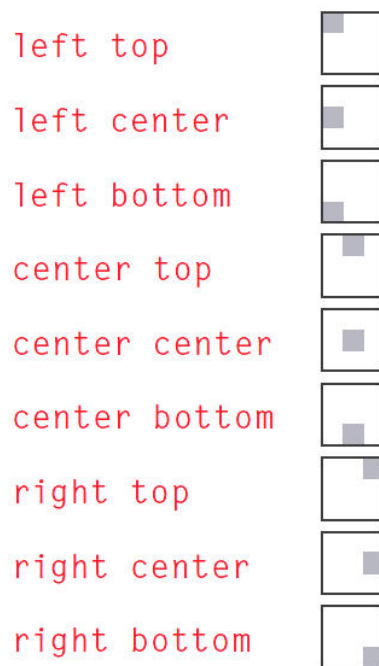
Ukoliko smo postavili da se slika ne ponavlja, možemo joj odrediti poziciju prema elementu koji je sadrži. Odnosno, slika se u startu postavlja uz gornju levu ivicu. Ukoliko želimo da je pomerimo odatle, koristimo ovo svojstvo.

*Background-position* se sastoji iz dve vrednosti. Prva predstavlja horizontalnu, a druga vertikalnu poziciju. Pišemo ih razdvojene razmakom. Mogu biti procenti, pikseli ili opisno. Pogledajmo primer:

```
body {
  background-image: url("images/back.png");
  background-image: url("images/back.png");
  background-repeat: no-repeat;
  background-position: 50% 0;
}
```

Ovaj kod će postaviti pozadinu sliku tako da će biti na horizontalnoj sredini (50%) i u vrhu elementa (0).

Umesto procenata, možemo koristiti piksele (px) i tada će slika biti tačno toliko udaljena od gornje leve ivice. Pored toga, možemo koristiti i opise koji su prikazani na slici ispod:



*Pozicije pozadinske slike*

## background-color

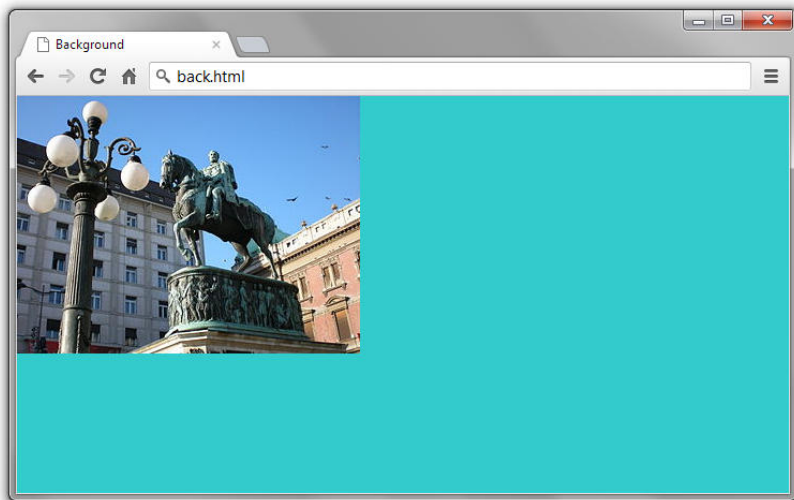
U 15. jedinici smo pomenuli *background-color* svojstvo jer smo tada govorili o bojama. Ovo svojstvo omogućuje da se postavi boja za pozadinu određenog elementa. Ukoliko se postavi i boja i slika, prvo će se prikazati boja tako što će ispuniti pozadinu elementa, a onda pre nje, ali opet u pozadini dolazi slika. Ne moramo koristiti oba svojstva, već je to stvar našeg izbora.

Pogledajmo ovaj kod:

```
body {
  background-image: url("images/back.png");
  background-repeat: no-repeat;
  background-position: left top;
```

```
background-color:#3CC;  
}
```

I prikaz u browseru:



*Prikaz u browseru – body ima i sliku i boju u pozadini*

Ukoliko bi postavili da se slika ponavlja po obe ose, boja pozadine se ne bi ni videla. Sa druge strane, ponekad je zgodno ostaviti providnu (transparentnu) boju pozadine.

### **Skraćeno pisanje i background**

Pogledajmo ovaj primer od malopre:

```
body {  
    background-image: url("images/back.png");  
    background-repeat: no-repeat;  
    background-position: left top;  
    background-color: #3CC;  
    background-attachment: scroll;  
}
```

Postavili smo sliku i boju u pozadinu i već imamo nekoliko redova. Možemo to pisati kraće, ovako:

```
body {  
    background: #3CC url("images/back.png") no-repeat scroll left top;  
}
```

Dakle, iz samog primera se vidi da možemo pisati sva background svojstva zajedno. Samo postavimo background svojstvo (bez sufiksa) i onda kao vrednost unosimo ovim redosledom:

1. *background-color*
2. *background-image*
3. *background-repeat*
4. *background-attachment*
5. *background-position*

i to razdvojeno razmakom. Ukoliko neki detalj nije potreban, jednostavno ga izostavimo. Na Vama je da se opredelite kako ćete pisati. I duža i kraća varijanta su ispravne.

## CSS3 pozadine

Za sada još uvek nije podržano od većine browsera, ali CSS3 donosi mogućnost da jedan element ima više pozadina. Postavljamo ih tako što pišemo u okviru *background* (kraće varijante) svojstva i odvajamo zarezom. Pogledajmo primer:

```
div {  
    background:  
        url(example-1.jpg) top left no-repeat,  
        url(example-2.jpg) bottom left no-repeat,  
        url(example-3.jpg) centre top repeat-x;  
}
```

Ovde je kod malo drugačije formatiran, ali podsećam, novi redovi ne igraju značajnu ulogu.

### Pitanje u jedinici

CSS2 dozvoljava?

- **Jednu boju pozadine i jednu pozadinsku sliku**
- Jednu boju pozadine i više pozadinskih slika
- Ili boju pozadine ili pozadinsku sliku
- Ne podržava slike u pozadini, samo boje.

## Najvažnije iz lekcije

- Na slike postavljene pomoću `img` taga možemo uticati kao i na sve druge elemente.
- Slike (`img`) su po default vrednostima *inline* elementi.
- Uvek moramo pripremiti pravu rezoluciju slike pre unosa.
- Možemo kreirati posebne klase za slike.
- Slike se mogu floatovati.
- Slike mogu biti u pozadini elementa i tada se ne unose kroz HTML, već CSS.
- CSS3 donosi više opcija za pozadinske slike.