

Într-una din lecțiile precedente, în care am abordat doar limbajul HTML, am văzut cum putem adăuga imagini pe paginile noastre web. În această lecție, vom învăța tehnicile care ne ajută să folosim regulile CSS pentru a influența aceste imagini (tag-urile `img`).

## Dimensiunile imaginilor

Cu regulile CSS, putem influența lățimea și înălțimea imaginilor, la fel ca și în cazul altor elemente. Setarea dimensiunilor exacte, originale prin CSS este utilă, deoarece, în timp ce încărcăm paginile, se va face imediat suficient spațiu pentru imagine, chiar dacă este necesar mai mult timp de încărcare.

După cum am spus deja în lecția 7, este interzis să schimbați dimensiunea unei imagini prin CSS sau HTML. Dacă aveți nevoie de o imagine de 300x200px, modificați imaginea în Photoshop sau într-un instrument similar și abia apoi introduceți-o în pagină. Nu o introduceți niciodată la dimensiunea de 3000x2000px, ca să o micșorați apoi la 300x200px în CSS.

Vă recomandăm să folosiți mai multe dimensiuni de imagini pe întregul site. De exemplu, setați o dimensiune pentru imaginile produselor, altă dimensiune pentru imaginile angajaților etc.

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

În acest exemplu, toate imaginile (tag-urile `img`) cu valoarea CLASS *product* vor avea dimensiunile de 250x100px. Trebuie doar să ne asigurăm că, într-adevăr, toate imaginile au 250x100px.

Acest exemplu poate fi extins mai târziu prin adăugarea unei variante mai mari a imaginilor produselor și legând de ele versiuni mai mici. Este important să avem câte două exemple ale aceleiași imagini (una mai mică și alta mai mare).

În consecință, întotdeauna definiți mai multe dimensiuni și creați câteva clase pentru ele, care vor apărea pe tot site-ul. Bineînțeles, asta în cazul în care nu a fost specificat altfel.

## Alinierea imaginilor

Când introducem o imagine în pagină, aceasta se comportă ca orice alt element inline. Dacă am fi postat-o înainte sau după tag-ul paragraph, imaginea s-ar fi găsit deasupra, respectiv sub paragraf, iar în caz că imaginea (tag-ul img) ar fi fost postată în cadrul tag-ului paragraf, aceasta s-ar fi postat ca și caracter al textului, ca în exemplul următor:



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.

*Imaginea 16.1 - Poziția implicită a imaginii setate la începutul textului*

<p>



Lorem ipsum dolor sit amet...

</p>

Totuși, cu CSS o putem influența, precum și relația ei cu celelalte elemente.

## Float stânga/dreapta

În loc să aplicăm asupra imaginilor atributul *align* menționat mai devreme, putem utiliza proprietatea *float* pentru a aranja imaginile pe pagină. Unul dintre moduri este să adăugăm clasei (menționate mai devreme) *float:left* sau *right* sau putem crea o clasă nouă, de exemplu:

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

și să o adăugăm la nevoie. De exemplu, în HTML:

```

```

În acest tag *img*, vedem două clase. Una este *product*, care va seta toți parametrii imaginii, în timp ce clasa *left* va seta *float* stânga.

## Centrarea

Dacă vrem să centrăm imaginea, putem face asta în două moduri. Unul constă în aplicarea proprietății *text-align* pe elementul de lângă, în cazul în care imaginea se află pe rândul textului. În acest caz, având în vedere că elementul este *inline*, se va comporta ca o literă a textului și va fi centrat.

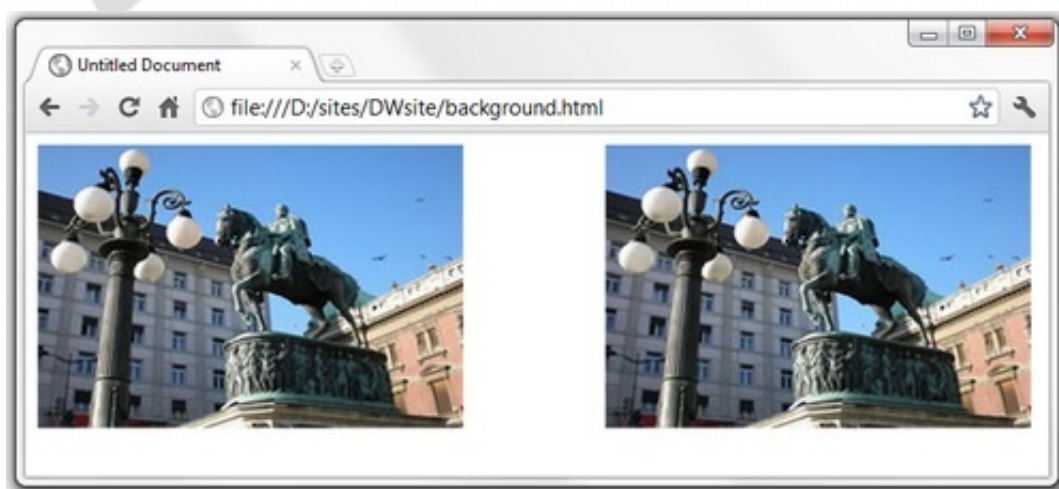
Al doilea mod constă, dacă deja am definit lățimea, în setarea marginilor din stânga și dreapta la valoarea *auto*, însă, în acest caz, trebuie să o transformăm în *block*. Deci, CSS ar arăta astfel:

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

Poate vi se pare complicat, dar, dacă elementului *block* i se conferă o lățime, setarea marginilor stângă și dreaptă la valoarea *auto* va forța elementul să se poziționeze în mijlocul elementului părinte. Această tehnică se poate folosi și pentru alte elemente, nu doar pentru imagini.

## Imaginile de fundal

Practic, fiecare element de pe pagină poate avea fundal. Fundalul poate fi o culoare, o imagine sau o combinație a celor două. Imaginile din fundal se deosebesc de imaginile introduse în mod normal. În HTML, acestea **nu apar deloc**, ci sunt legate de un anumit element prin CSS. De exemplu:



## *Imaginea 16.2 - Imagine introdusă normal prin tag-ul img (stânga) și imagine de fundal (dreapta)*

Partea HTML a paginii afișează două imagini. Cea din stânga a fost adăugată prin intermediul unui tag `img` normal, în timp ce cea din dreapta a fost adăugată ca fundal DIV. La prima vedere, nu există nicio deosebire. Totuși, în exemplul din partea stângă am creat un DIV și am adăugat o imagine în cadrul său. Întregul cod arată astfel:

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

DIV-ul nu are nici lățime, nici înălțime, dar primește aceste valori de la imagine, adică se extinde pentru a include imaginea.

Pe de altă parte, în cazul imaginii de fundal, în exemplul din dreapta, imaginea nu este inserată ca un element, ci ca fundal al elementului DIV. Propriul DIV este în continuare gol și, în cadrul său, putem insera text sau un alt conținut. În acest caz, DIV-ul se adaptează imaginii din fundal. Atributele sale nu influențează deloc DIV-ul.

În exemplul din stânga, imaginea este propriul conținut, în timp ce, în cel din dreapta, este doar ca fundal.

În exemplul nostru, în imaginea din dreapta este inserat următorul cod:

```
<div class="backImage">  
  
</div>
```

Dar trebuie să adăugăm și CSS:

```
.backImage {  
  background:url(images/image.jpg) left top no-repeat;  
  width:300px;  
  height:200px;  
}
```

Observăm că, în codul HTML, DIV-ul este gol și nu are niciun fel de conținut, în timp ce adăugăm imaginea prin CSS (de detalii ne vom ocupa în continuare). Pe lângă aceasta, trebuie să introducem lățimea și înălțimea, deoarece, în caz contrar, dimensiunile vor depinde de conținutul DIV-ului; adică, în acest exemplu, dacă am elimina declarațiile pentru lățime/înălțime, imaginea nu s-ar vedea deloc (deoarece elementul DIV este gol și nu are dimensiuni).

În exemplul de mai sus, am introdus niște detalii noi, fără să le explicăm. Haideți acum să vedem ce proprietăți CSS influențează imaginile de fundal și cum sunt adăugate.

## Background-image

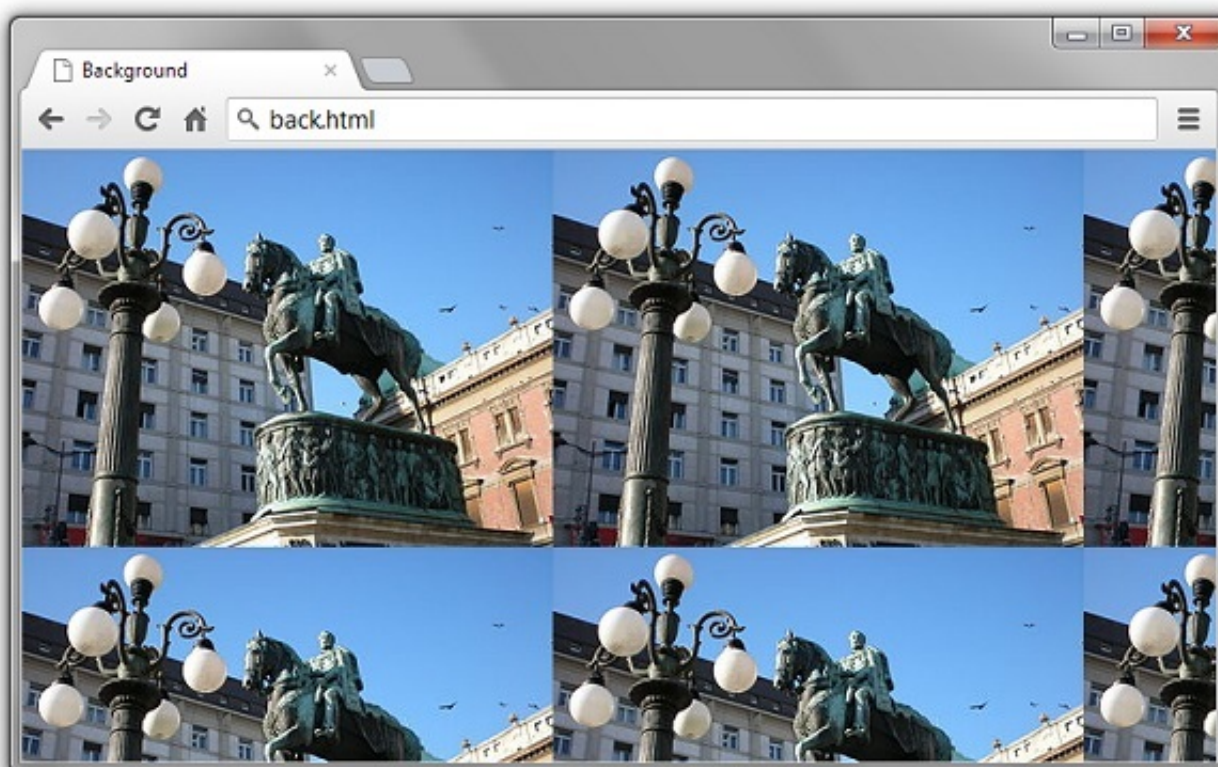
Această proprietate este cea mai importantă pentru imaginile de fundal, deoarece cu ajutorul valorii sale apelăm fișierul imaginii. Deci, imaginea din fundal nu se încarcă prin HTML, ci prin regula CSS.

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



(Valoarea proprietății *background-image* este setată astfel încât să putem scrie URL-uri și calea relativă sau absolută până la imaginea dintre paranteze și semne de citare.)

Dacă adăugăm doar această proprietate, în mod implicit, imaginea se va repeta la infinit pe axele x și y. Acest lucru va fi vizibil, în cazul în care imaginea este mai mică decât elementele pe care este setată.



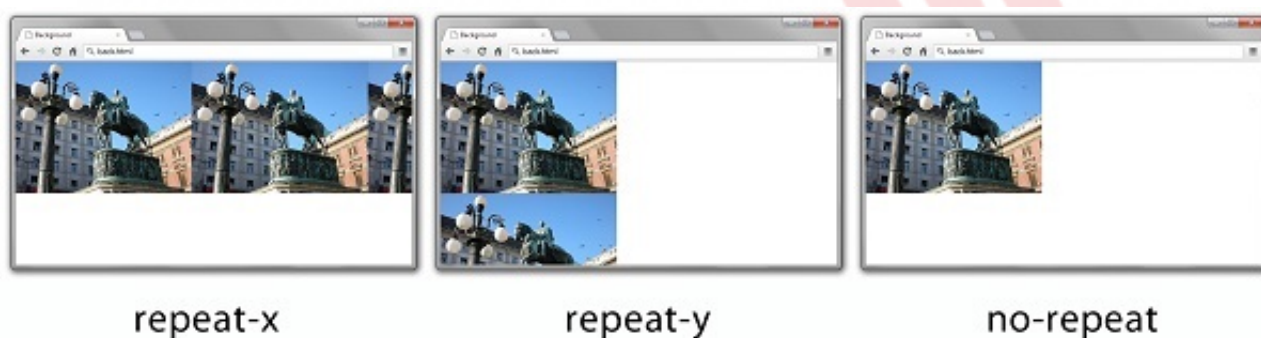
*Imaginea 16.3 - Imaginea de fundal adăugată pe elementul body*

## Background-repeat

Imaginea din fundal se poate repeta în mai multe moduri. Valoarea implicită constă în repetarea imaginii la nesfârșit pe ambele axe (ca în exemplul de mai sus). Dacă setăm valoarea *repeat-x*, respectiv *repeat-y*, imaginea se va repeta doar pe axa x, respectiv axa y. La final, dacă o setăm la *no-repeat*, imaginea nu se va mai repeta și va fi afișată doar o singură dată.

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

Acest cod va seta imaginea de fundal și o va repeta pe orizontală (axa x). Să vedem această variantă și celelalte variante în browser:



*Imaginea 16.4 - Variante posibile de repetare a imaginii de fundal (pe lângă cea de bază)*

## Background-attachment

Dacă vrem, putem fixa imaginea de fundal într-o poziție fixă (atașată de browser), în timp ce celelalte elemente se pot derula. În acest context, folosim proprietatea *background-attachment*. Poate avea valorile *fixed* și *scroll* (care este implicită).

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

## Background-position

În cazul în care am setat ca imaginea să nu se repete, îi putem stabili



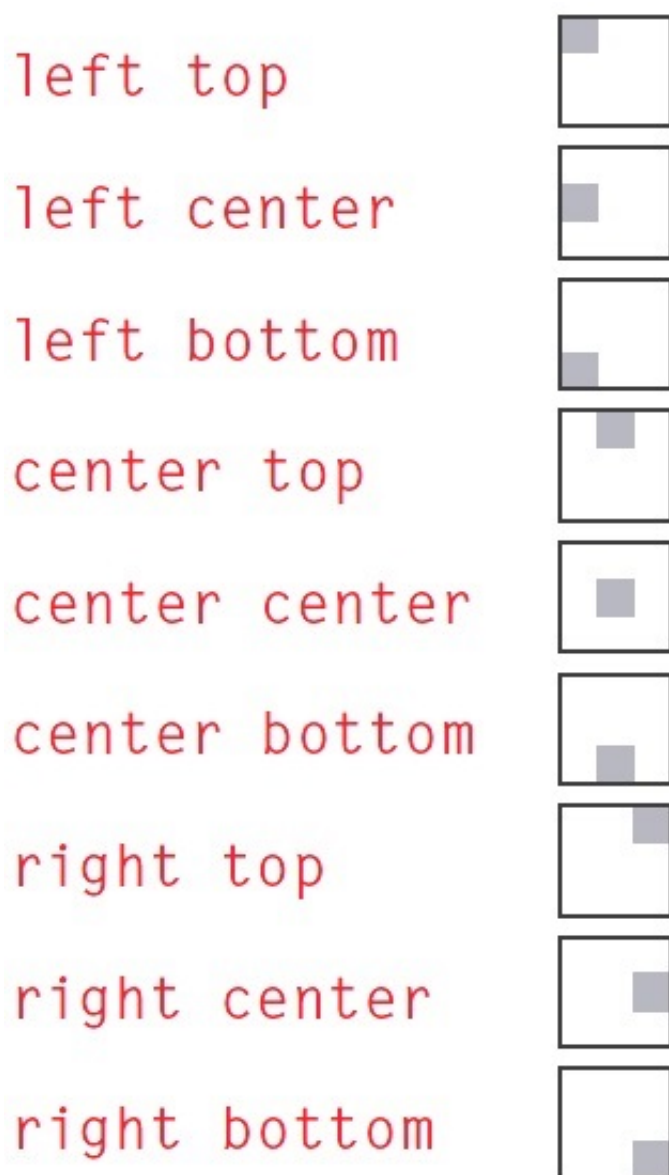
poziția după elementul care o conține, respectiv imaginea este postată de-a lungul marginii stângi de sus. Dacă vrem să o mutăm de acolo, folosim această proprietate.

*Background-position* este alcătuită din două valori. Prima reprezintă poziția orizontală, iar a doua pe cea verticală. Le scriem separate prin virgulă. Pot fi procente, pixeli sau descrieri. Haideți să vedem următorul exemplu. Priviți exemplul:

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

Acest cod va seta imaginea de fundal, astfel încât să fie poziționată în mijloc pe orizontală (50%) și în partea de sus a elementului (0).

În loc de procente, putem folosi pixeli (px) și atunci imaginea se va afla exact la această distanță de marginea stângă de sus. Pe lângă asta, putem folosi și descrieri, care sunt afișate în imaginea de mai jos:



*Imaginea 16.5 - Pozițiile imaginii de fundal*

## **Background-color**

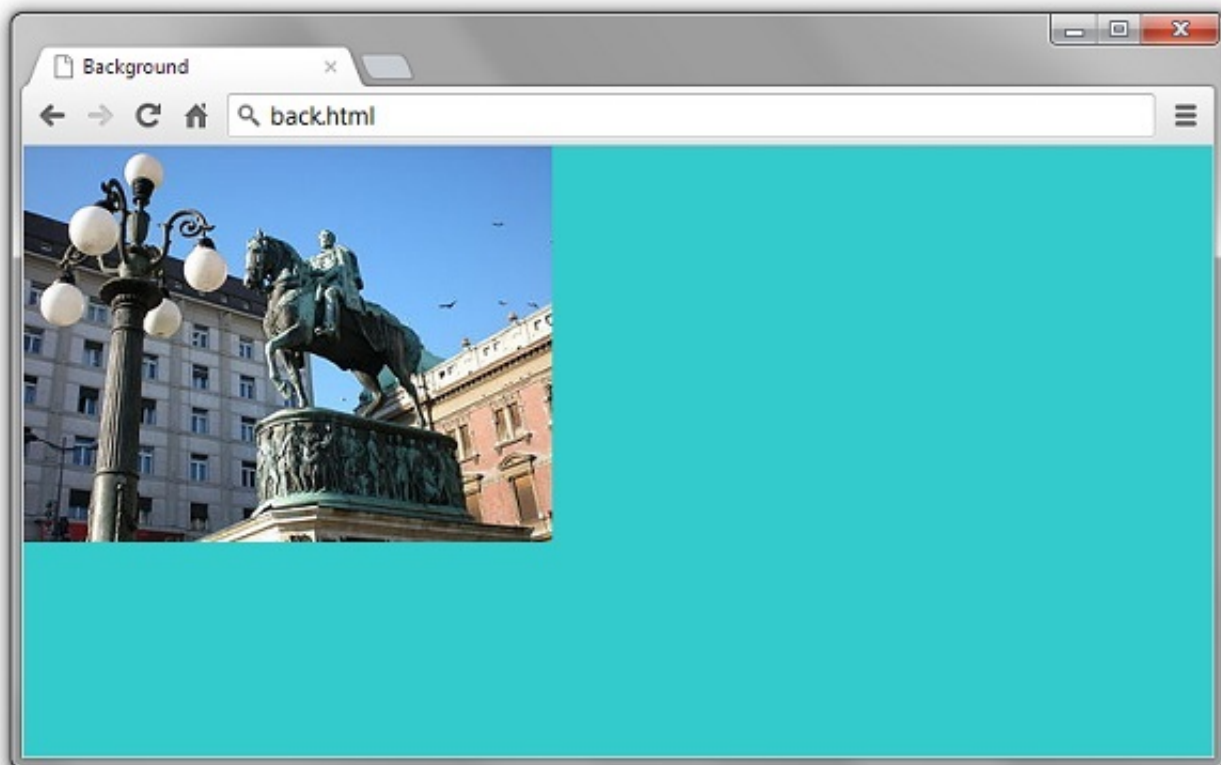
În lecția 14, în care am vorbit despre culori, am menționat și proprietatea *background-color*. Această proprietate ne permite să adăugăm o culoare de fundal unui anumit element. Dacă adăugăm și culoarea, și imaginea, mai întâi vom vedea culoarea care va completa fundalul elementului, iar apoi în fața ei, imaginea. Nu este obligatoriu să folosim ambele proprietăți, acest lucru rămânând la latitudinea

noastră.

Să vedem următorul cod:

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

Afișarea acestui cod în browser va fi următoarea:



*Imaginea 16.6 - Afișarea în browser – partea body conține și imaginea, și culoarea de fundal*

Dacă am seta ca imaginea să se repete pe ambele axe, culoarea de fundal nici nu s-ar vedea. Pe de altă parte, uneori este bine să lăsăm culoarea de fundal transparentă.

## Scrierea prescurtată și *background*

Să privim cu atenție exemplul de mai devreme:

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

Am setat imaginea și culoarea de fundal și deja avem câteva rânduri. Tot codul poate fi scris și mai pe scurt, în felul următor:

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

Deci, din exemplu se vede că putem scrie toate proprietățile *background* împreună. Trebuie doar să setăm proprietatea *background* (fără sufix) și apoi să introducem valoarea în această ordine:

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

Separată prin virgulă. Dacă vreun detaliu nu este necesar, pur și simplu îl lăsăm deoparte.

Depinde doar de dvs. cum veți scrie. Și varianta mai lungă, și cea mai scurtă sunt corecte.

## Fundalurile CSS3

CSS3 permite opțiunea de a avea un element cu mai multe fundaluri. Le scriem în cadrul proprietății *background* (versiunea scurtă) și le separăm prin virgulă. Exemplu:

```
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;  
}
```

Aici, codul este formatat puțin altfel, dar vă reamintim că noile rânduri nu fac nicio diferență.

Dacă vrem, putem să introducem modul mai lung de scriere (în special, *background-image*, *background-position*, *background-repeat* și altele). În acest caz, din nou le separăm prin virgulă și le scriem pentru fiecare imagine. Priviți aceasta concret pe un exemplu:

```
body {  
  background-image: url(first.png), url(second.png);  
  background-position: center bottom, left top;  
  background-repeat: no-repeat, no-repeat;  
}
```



## Background size

O problemă care apărea deseori era și imposibilitatea de scalare a imaginilor în fundal. Respectiv, putem muta *background*-ul, însă nu putem schimba dimensiunea. În CSS3, o nouă proprietate este *background-size*, care ne ajută să facem scalarea imaginilor.

```
.box {  
  background-image:url(pic.png);  
  background-size:20px 100px;  
  background-repeat:no-repeat;  
}
```

În exemplul de mai sus, imaginea din fundal va fi de 20x100px, deși poate imaginea originală are dimensiuni diferite.

Ca și la imaginile din HTML/CSS, atenție atunci când schimbați dimensiunea și faceți aceasta doar când este necesar, deoarece se va încărca din nou o imagine completă, mare și se poate ajunge la degradarea calității.