

CSS در Box Model

مهندس داوود میرزایی



CSS در Box Model

نویسنده : مهندس داود میرزایی

ناشر دیجیتال : آرمان پژوهان نشر (شرکت مهندسی آرمان پردازان نوین)

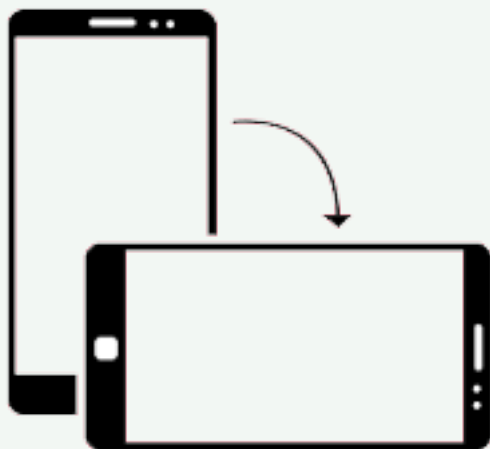
پیاده سازی : مهندس مهناز سادات عظیمی، نرگس اسدی

سال انتشار : تیرماه 1398

نسخه : 1.0



www.karashin.ir



جهت مطالعه بهتر در تلفن همراه گوشی خود را در حالت افقی

نگه دارید .

www.karashin.ir

info@karashin.ir

[@karashin](https://www.instagram.com/karashin)

[@karashin.ir](https://www.facebook.com/karashin)

[03157772244](tel:03157772244), [03157772464](tel:03157772464)

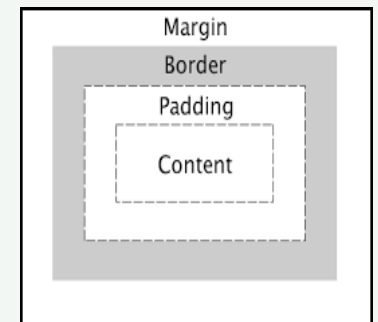
1 - رنگ‌ها در CSS

- ۷ رنگ پس‌زمینه
- ۸ opacity
- ۹ HSL color



2 - Box Model

- ۱۲ border
- ۱۳ margin و padding
- ۱۴ گوشه‌های گرد
- ۱۵ سایه
- ۱۶ عرض و ارتفاع
- ۱۷ box-sizing
- ۱۹ پس‌زمینه
- ۲۴ overflow



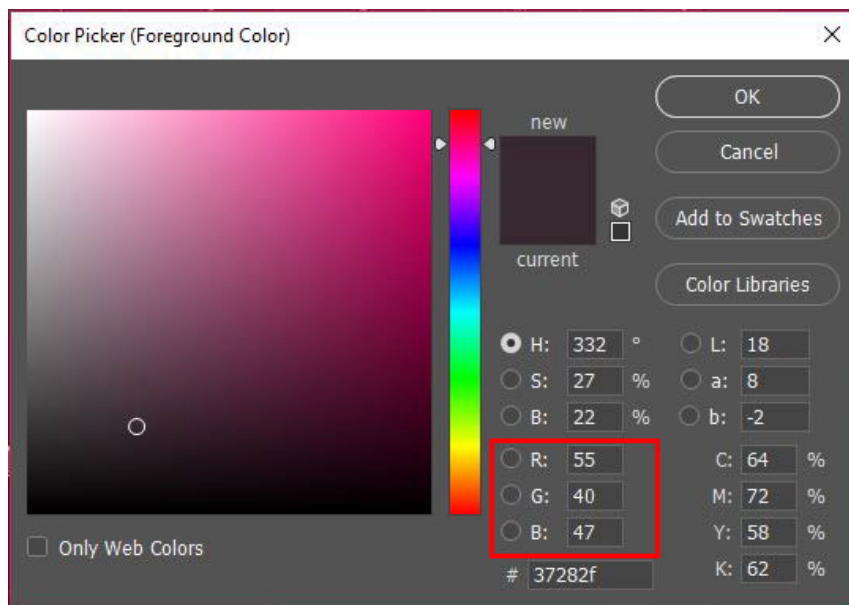
به نام خداوند بخشنده مهربان

تقدیم به امام عصر (عج)

در مدل رنگ RGB، سبز، آبی، قرمز سه رنگ اصلی هستند که با ترکیب آنها رنگ‌های مختلف به وجود می‌آید. همچنین سه رنگ فرعی زرد از ترکیب قرمز و سبز، فیروزه‌ای از ترکیب سبز و آبی و ارغوانی از ترکیب قرمز و آبی به دست می‌آید و سفید حاصل ترکیب هر سه رنگ است.

تعداد رنگ‌های موجود در این مدل $256 * 256 * 256$ یعنی 2 به توان 24 رنگ می‌باشد.

به کمک فتوشاپ به راحتی می‌توانید کدهای رنگی در مدل RGB را پیدا کنید.



در CSS به سه روش می‌توان رنگ‌ها را تعیین کرد

1- مقادیر RGB

در مدل RGB یک رنگ بر اساس ترکیبی از سه مقدار قرمز، سبز و آبی به دست می‌آید.

کد هر رنگ عددی بین 0 تا 255 است مثل

`rgb(100,100,90)`

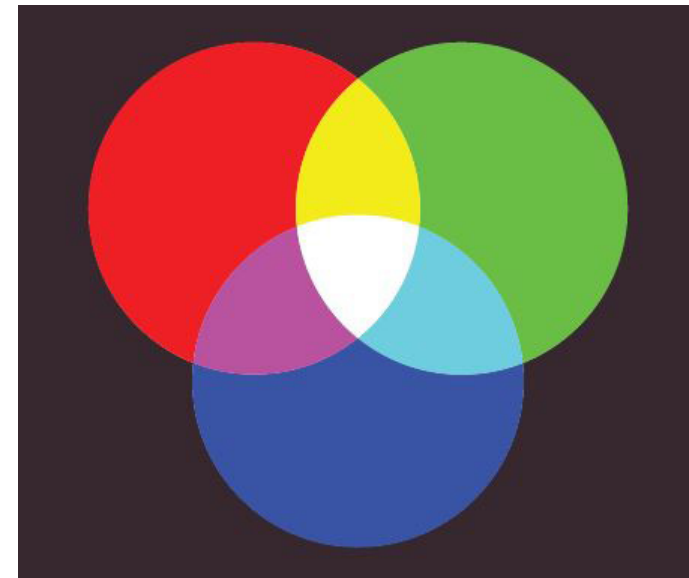
و همچنین برای دو رنگ سیاه و سفید کد رنگ‌ها به صورت زیر است:

`rgb(0,0,0)`

سیاه

`rgb(255,255,255)`

سفید



2- کدهای هگزا دسیمال

از همان مدل RGB استفاده می‌شود ولی به جای مقادیر عددی از کدهای هگزا دسیمال برای تعیین کد رنگ های قرمز ، سبز و آبی استفاده می‌شود. کد هر رنگ از 00 تا FF می‌باشد که معادل 0 تا 255 در مبنای 10 می‌باشد.

مثل: # ee3e80

FFFFFF برای رنگ سفید

000000 برای رنگ سیاه

3- نام رنگ

می‌توان مسقیما از نام رنگ استفاده کرد که 147 رنگ از پیش تعریف شده برای CSS وجود دارد مانند green

```
h1{
color:rgb(100,100,90);}
```

```
h2{
color:#ee3e80;}
```

```
p{
color:Darkcyan;}
```

خاصیت color برای تعیین رنگ متن به کار می‌رود.

1-1 رنگ پس زمینه

از خاصیت background-color برای تعیین رنگ پس زمینه استفاده می‌شود و نوع رنگ هم به همان روش قبلی تعیین می‌شود.

```
background-color:rgb(200,200,200);
```

اگر بخواهیم رنگ کل صفحه را تغییر دهیم باید این خاصیت را برای تگ body تعیین کنیم.

```
body{background-color: rgb(200,200,200);}
```

CSS

```
background-color: rgb(200,100,50);  
opacity:.5;
```

CSS خاصیت جدیدی به نام opacity معرفی کرد که به کمک آن می‌توانیم روشنایی رنگ را هم تعیین کنیم که عددی است بین 0 تا 1 که 0 یعنی تیره (سیاه) و یک یعنی کاملاً روشن و مثلاً مقدار 0.5 یعنی 50 درصد روشنایی همچنین می‌توان opacity را در rgb ترکیب کرد.

```
background-color:rgb(200,100,50,.5)
```

خروجی



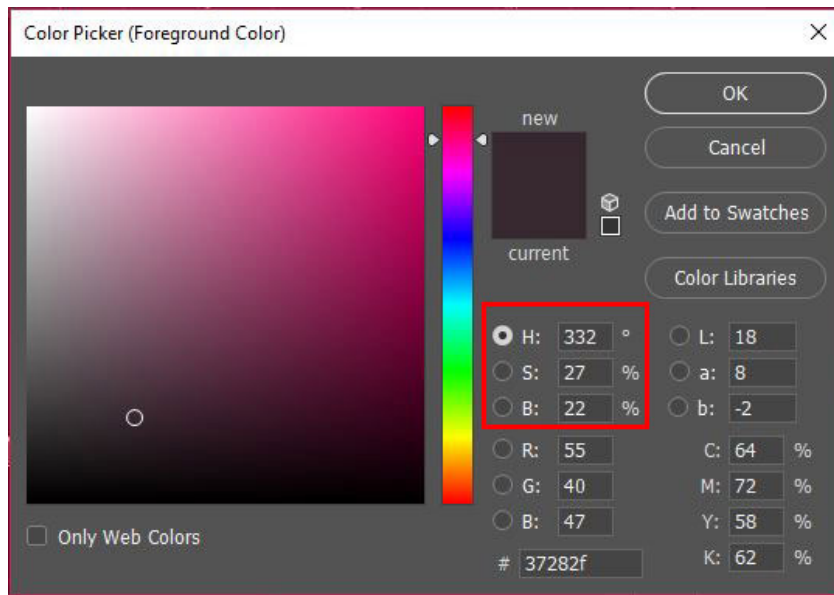
this is a test page


```
body{background-color:hsl(0,0%,78%);}
```

همچنین میزان شفافیت را می‌توان در پارامتر چهارم به نام پارامتر `alpha` تعیین کرد.

```
body{background-color:hsla(0,100%,100%,0.5);}
```

که 0.5 میزان شفافیت (transparent) می‌باشد. به کمک فتوشاپ می‌توان کدهای رنگ `hsb` را تعیین کرد.



روش جدیدی برای تعیین رنگ فراهم کرده به نام HSL که مخفف کلمات `HUE, SATURATION, LIGHTNESS` می‌باشد.

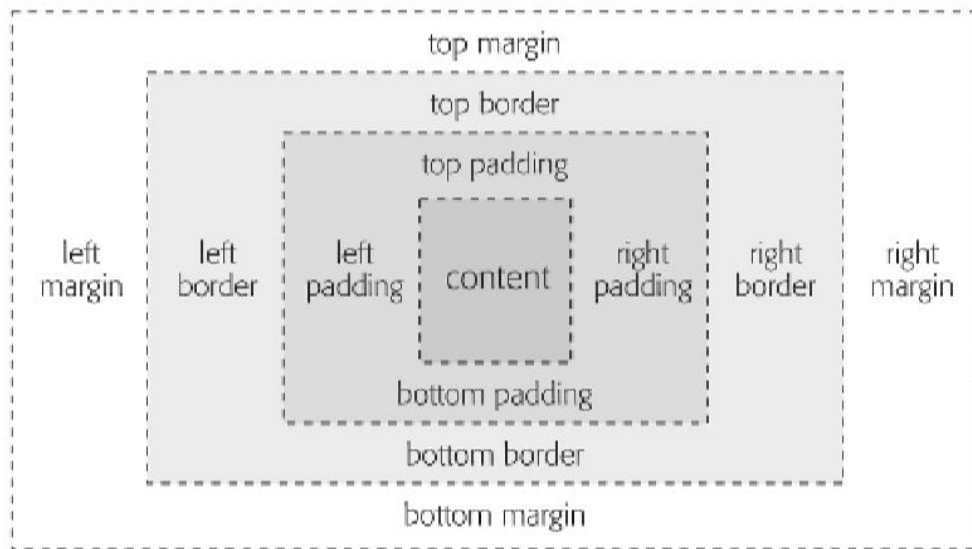
`HUE`- نوع رنگ را در دایره رنگ تعیین می‌کند عددی است بین 0 تا 360
`SATURATION`- میزان رنگ خاکستری موجود در رنگ را تعیین می‌کند و عددی است بین 0 تا 100 که 0 یعنی سایه ای از خاکستری و 100 یعنی بدون خاکستری

`LIGHTNESS`- میزان رنگ سفید (lightness) یا سیاه (darkness) در رنگ را تعیین می‌کند و عددی است بین 0 تا 100 که عدد 0 یعنی سیاه کامل و عدد 100 یعنی سفید کامل



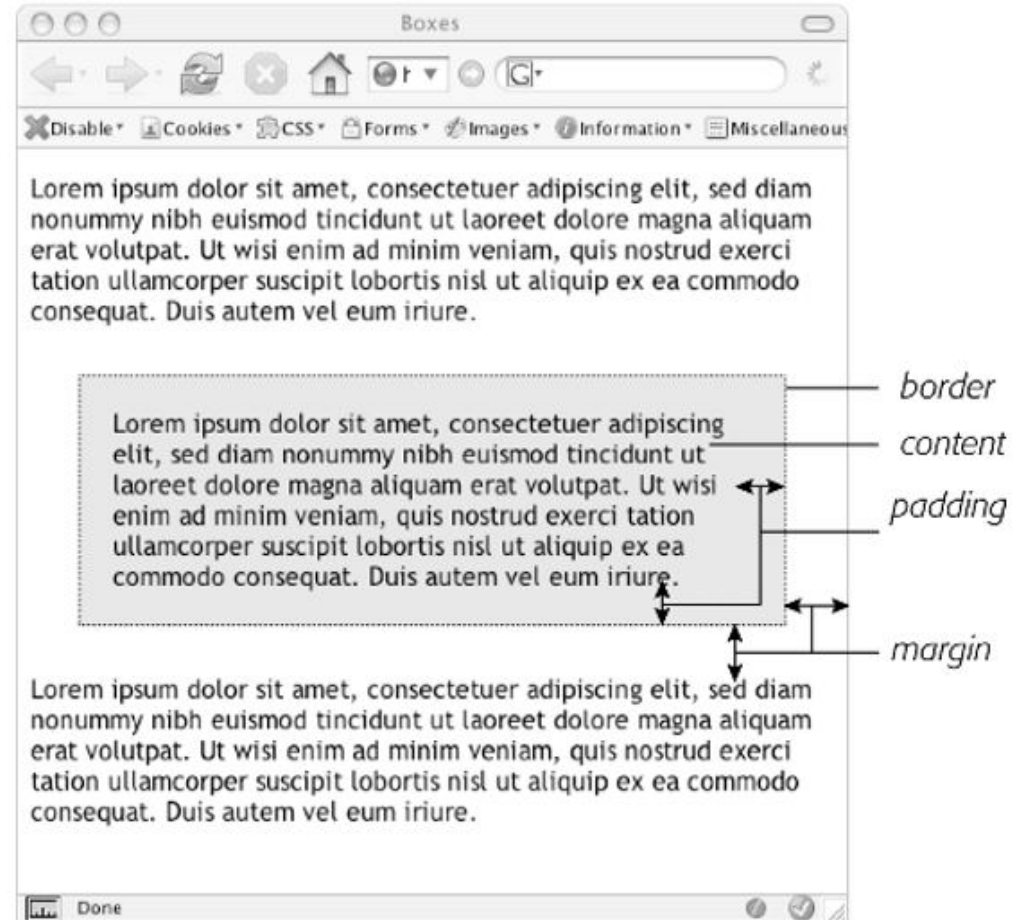
Box Model

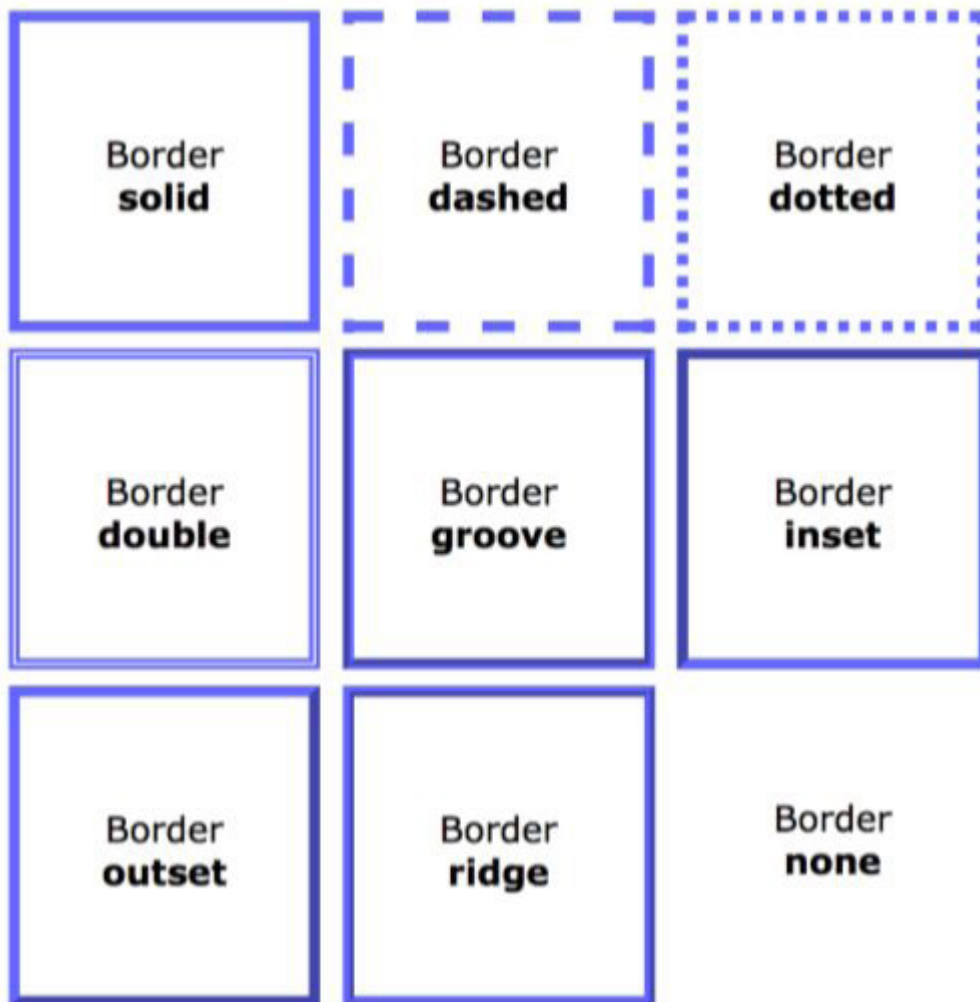
- border •
- margin و padding •
- گوشه‌های گرد •
- سایه •
- عرض و ارتفاع •
- box-sizing •
- پس زمینه •
- overflow •



مرورگرها با همه تگ‌ها همانند یک جعبه رفتار می‌کنند یعنی همه تگ‌ها درون یک مستطیل فرضی قرار دارند حال درون این مستطیل می‌تواند متن باشد، عکس باشد یا عناوین، جداول، لیست‌ها و...

تصویر زیر را در نظر بگیرید





content: منظور محتویات تگ می باشد (متن، عکس یا...)

padding: فاصله خالی بین محتوا و خطوط حاشیه

border: خطوط دور مستطیل فرضی که اطراف تگ قرار گرفته است.

background-color: رنگ پس زمینه که شامل محتوا و قسمت padding

می باشد

margin: فاصله بین یک تگ تا تگ کناری

به کمک padding و margin می توان بین دو تگ فاصله انداخت ولی دقت کنید که اگر border یا background-color داشته باشید آنگاه نتیجه متفاوت است چراکه رنگ پس زمینه شامل padding می شود ولی شامل margin نمی شود یعنی border اطراف padding قرار دارند.

شما می توانید margin, padding و border را برای هر چهار طرف جعبه ها تنظیم کنید هم می توانند یکسان باشند و هم متفاوت

border ۱-۲

border خطوط حاشیه ای هستند که چهار طرف مستطیل فرضی را مشخص می کنند شما می توانید به کمک CSS نوع، ضخامت و رنگ آن ها را تعیین کنید.

انواع خطوط حاشیه ای که توسط border-style تعیین می شوند عبارتند از

solid, dotted, dashed, double, groove, ridge, inset, outset, non, hidden

همچنین می‌توان جداگانه هم خطوط حاشیه را کنترل کرد
border-top و border-right و border-left و border-bottom

به طور مثال

```
border-top: 2px solid rgb(0,0,255)
```

فقط خط حاشیه بالایی را تنظیم می‌کند

۲-۲ padding و margin

به کمک خصوصیت‌های padding و margin می‌توان فضای خالی بین محتوا
تا border و همچنین فاصله بین تگ‌ها را تنظیم کرد.

```
padding: 2px;
```

یعنی فضای خالی بین محتوا و border از هر چهارطرف به اندازه 2px باشد.

برای کنترل padding یا margin برای هر چهارطرف به طور جداگانه هم
می‌توان از روش‌های مختلفی استفاده کرد

```
padding: 2px 0 5px 4px
```

به صورت ساعتگرد و از بالا padding را تعیین می‌کند

پس از بالا 2px، راست 0، پایین 5px و چپ 4px فضای خالی ایجاد کرده‌ایم.

نوع border توسط border-style تعیین می‌شود

```
border-style: solid
```

و برای هر چهار طرف به‌طور جداگانه هم می‌توان از خصوصیت‌های
border-left-style و border-right-style و
border-top-style و border-bottom-style استفاده کرد.

ضخامت border توسط border-width تعیین می‌شود

```
border-width: 2px;
```

و برای هر چهار طرف به‌طور جداگانه هم می‌توان از خصوصیت‌های
border-left-width و border-right-width و
border-top-width و border-bottom-width استفاده کرد.

رنگ border توسط border-color تعیین می‌شود

```
border-color: red;
```

و برای هر چهار طرف به‌طور جداگانه هم می‌توان از خصوصیت‌های
border-left-color و border-right-color و
border-top-color و border-bottom-color استفاده کرد.

و می‌توان توسط خصوصیت border همه موارد فوق را یکجا تنظیم کرد

```
border: 2px solid red;
```

که اولی ضخامت، دومی نوع و سومی رنگ border را تعیین می‌کند

و مانند border نیز می‌توان برای هر چهار گوشه جداگانه مقادیر را تعیین کرد.

`border-radius: 0 30px 10px 5px;`

که به ترتیب گوشه‌های بالاچپ، بالا راست، پایین راست و پایین چپ را تنظیم می‌کند.

در حالت پیش فرض گوشه‌ها به صورت دایره‌ای هستند در صورتی که بخواهیم به صورت بیضی باشند به صورت زیر عمل می‌کنیم:

`border-radius: 40px/20px;`

که عدد اول شعاع افقی و دومی شعاع عمودی بیضی است.

می‌توان از خاصیت‌های زیر هم برای تعیین جداگانه گوشه‌ها استفاده کرد

`border-top-left-radius`

`border-top-right-radius`

`border-bottom-left-radius`

`border-bottom-right-radius`

`padding: 5px 10px`

اگر فقط دو عدد داشته باشیم اولی مربوط به بالا و پایین و دومی مربوط به چپ و راست می‌باشد.

خصوصیت‌های

`padding-top` و `padding-right` و `padding-bottom` و `padding-left`

`margin-top` و `margin-right` و `margin-bottom` و `margin-left`

نیز جداگانه می‌توانند `padding` و `margin` را تنظیم کنند.

معمولاً از واحد `px` یا `em` استفاده می‌شود که `20px` یعنی فضایی به اندازه `20` پیکسل ایجاد شود و `3em` یعنی فضایی به اندازه سه برابر اندازه فونتی که برای تگ تعیین شده است.

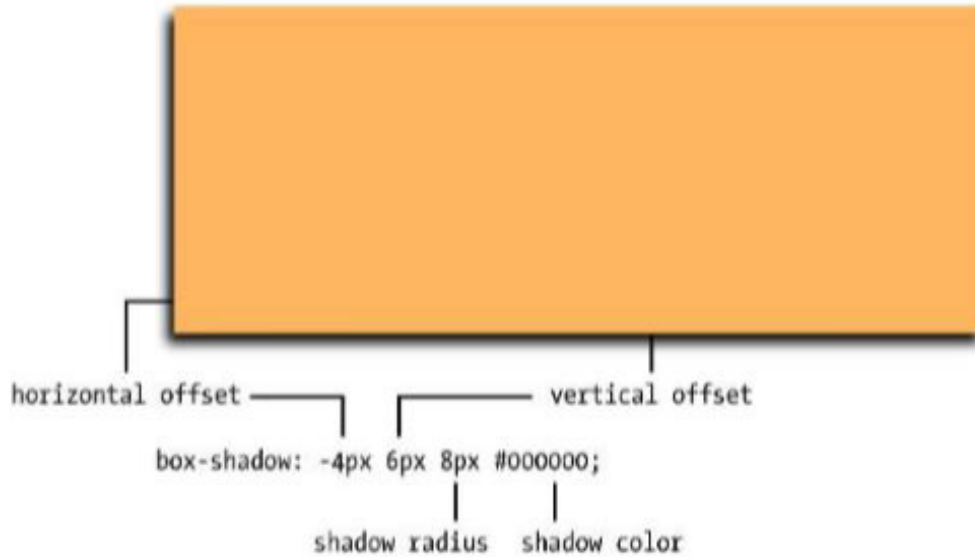
همچنین از `%` نیز می‌توان استفاده کرد که براساس عرض صفحه نمایش تعیین می‌شود.

۲-۳ گوشه های گرد

به طور پیش فرض خطوط حاشیه به شکل مستطیل با زاویه‌های راست نمایش داده می‌شوند اما می‌توان به کمک `CSS` گوشه‌های گرد ایجاد کرد برای اینکار از `border-radius` استفاده می‌کنیم.

`border-radius: 20px;`

می‌توان از پیکسل، `em` یا درصد نیز استفاده کرد.



می‌توان برای مستطیل‌ها یا همان جعبه‌ها سایه تعیین کرد که جلوه زیبایی به آن خواهد داد به شرطی که به موقع استفاده شود. دستور ایجاد سایه به صورت زیر است:

`box-shadow: horizontal-offset vertical-offset radius color`

horizontal-offset: مقدار مثبت سایه را به سمت راست و مقدار منفی سایه را به سمت چپ می‌کشد.

vertical-offset: مقدار مثبت سایه را به سمت پایین و مقدار منفی سایه را به سمت بالا می‌کشد.

radius: میزان محوشدگی (یا بلور) سایه را مشخص می‌کند 0 یعنی بدون محوشدگی، هرچه مقدار بالاتر رود میزان محوشدگی بیشتر می‌شود.

color: رنگ سایه را تعیین می‌کند.

دو مقدار اختیاری دیگر نیز وجود دارد

inset: یعنی سایه به طرف داخل کشیده شود که لازم است به عنوان اولین مقدار وارد شود

`box-shadow: inset 4px 4px 8px red;`

spread: بین شعاع و رنگ می‌توان مقدار اسپری سایه را تعیین کرد مثلاً اگر مقدار آن 10 باشد یعنی سایه از هر چهار طرف به اندازه 10px پخش می‌شود

می توان دو سایه به یک مستطیل اعمال کرد کافی است آن‌ها را با , از هم جدا کنیم.

۲-۵ عرض و ارتفاع

برای تعیین اندازه عرض و ارتفاع جعبه‌ها از خصوصیت `width` و `height` استفاده می‌کنیم

```
width: 300px;
width: 30%;
width: 20em;
```

از واحدهای پیکسل، درصد و `em` برای تعیین اندازه می‌توان استفاده کرد.

`em` براساس اندازه فونت جعبه محاسبه می‌کند یعنی اگر اندازه فونت `24px` باشد `2em` برابر `2*24` پیکسل می‌شود.

و درصد، اندازه جعبه را براساس درصدی از اندازه جعبه پدر تعیین می‌کند.

برای محاسبه عرض و ارتفاعی که یک تگ در صفحه اشغال می‌کند باید اندازه `border`، `padding` و `margin` را هم در نظر بگیریم

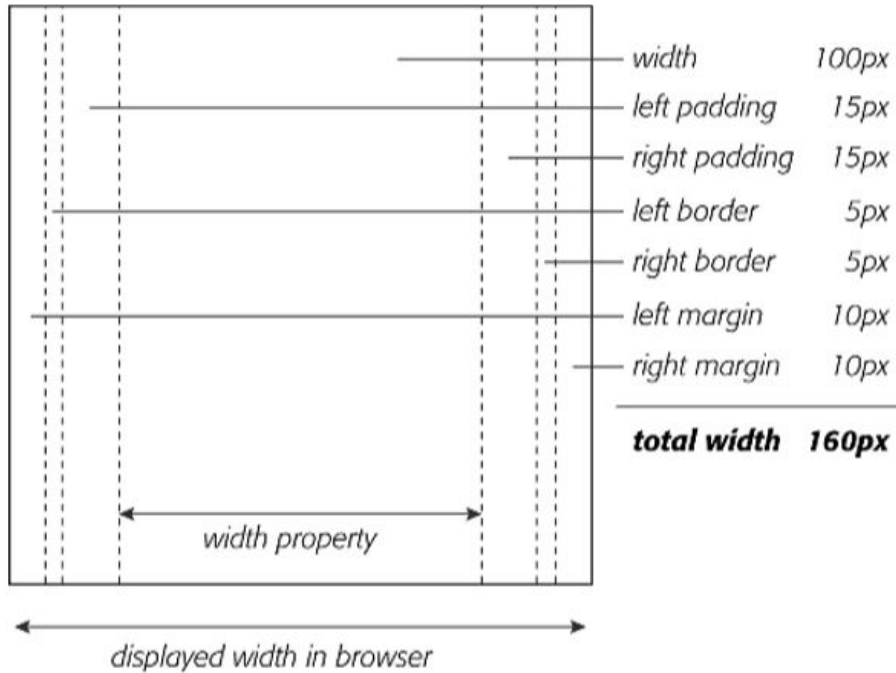
مطابق شکل صفحه بعد.

```
box-shadow: 4px 6px 8px rgba(0,0,0,.75);
```

```
box-shadow: 0 0 8px 2px rgba(0,0,0,.75);
```

```
box-shadow: -4px -6px 8px 2px rgba(0,0,0,.5),
            8px 6px 12px rgba(0,0,0,.5);
```

```
box-shadow: inset 0 0 8px 2px rgba(0,0,0,.75);
```

ارتفاع یک مستطیل براساس میزان محتوای درون آن تعیین می‌شود و اگر به صورت دستی عددی را تعیین کنیم ممکن است نتیجه دلخواه بدست نیاید.

HTML

```
<p> the amount of content is taller than the box.</p>
```

CSS

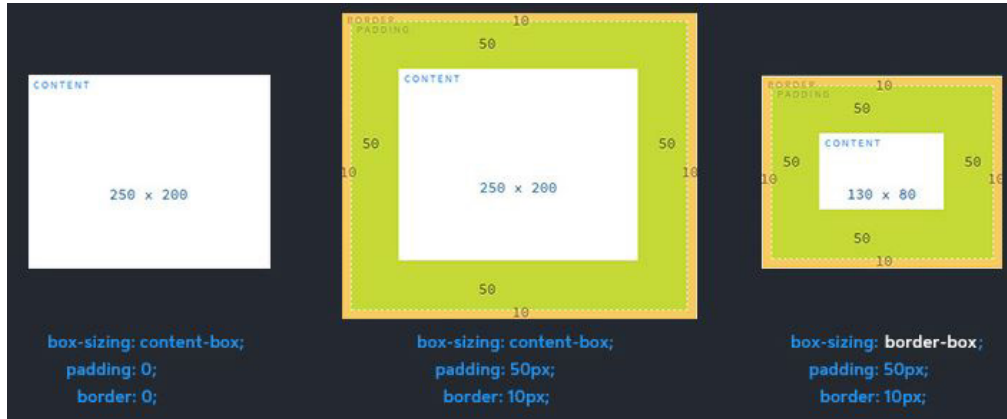
```
p{
  width: 50px;
  height: 50px;
  background-color: #cccccc}
```

خروجی

the
amount
of
content
is taller
than
the
box.

border-box: در این حالت مقدار width به اندازه محتوا و padding و border

اختصاص می‌یابد



در CSS3 این امکان فراهم شده که مقدار width و height باز تعریف شوند یعنی در حالت عادی این اندازه‌ها به محتوا اختصاص می‌یابد و اندازه‌های padding و margin و border به آن اضافه می‌شود مانند آنچه گفتیم اما با تعیین box-size به یکی از حالت‌های زیر می‌توان این رفتار را تغییر داد

content-box: حالت پیش فرض که توضیح داده شد.

padding-box: در این حالت مقدار width به اندازه محتوا و padding تعلق

می‌گیرد

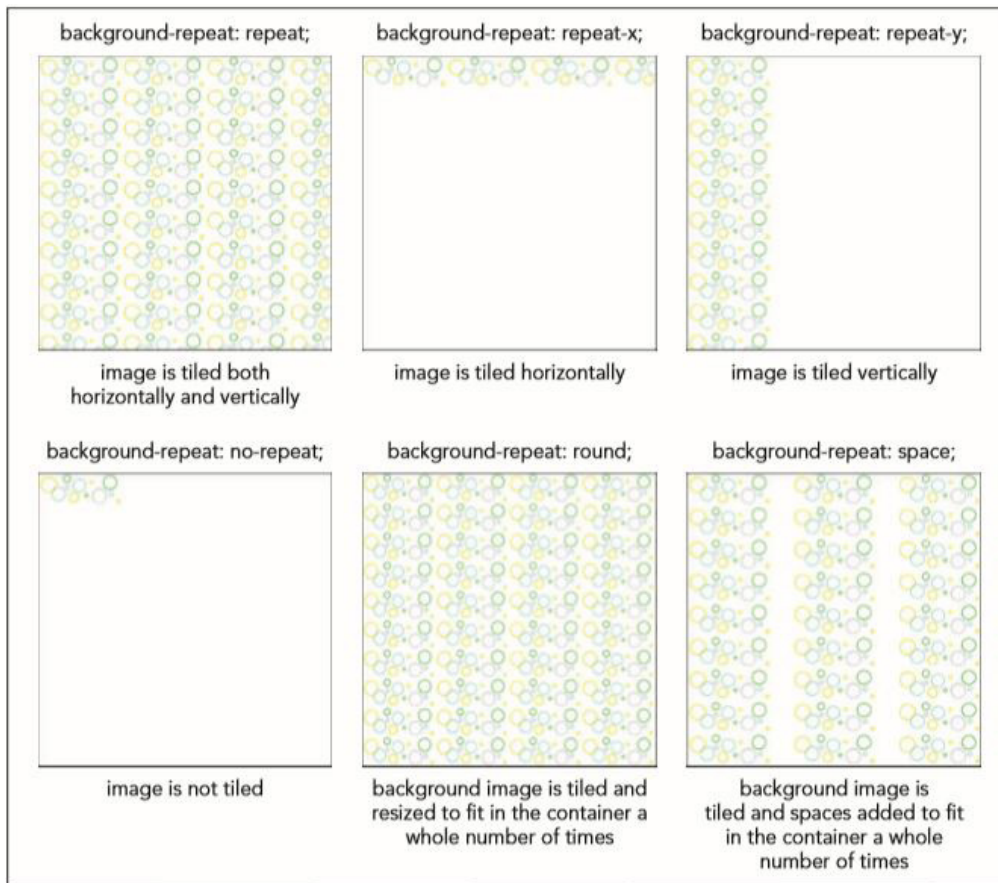
CSS

```
div {
  box-sizing: padding-box;
  width: 250px;
  height: 200px;
  padding: 50px;}

```



که type می‌تواند یکی از مقادیر `repeat` , `repeat-x` , `repeat-y` , `round` و `no-repeat` باشد.



تاکنون با `background-color` برای ایجاد رنگ پس زمینه آشنا شدیم CSS همچنین از عکس برای پس زمینه نیز پشتیبانی می‌کند. برای اینکار از `background-image` استفاده می‌کنیم.

```
background-image: url(...);
```

که `url` نام و آدرس تصویر را مشخص می‌کند که باید به صورت نسبی ذکر شود.

```
.intro{ background-image: url(tree.png);}
```

که یعنی تصویر `tree.png` به عنوان پس زمینه `div` یا کلاس `intro` قرار گیرد. در اینجا باید تصویر و فایل `CSS` در یک پوشه باشند و اگر تصویر در پوشه دیگری باشد باید آدرس نسبی آن را بنویسیم.

اگر بخواهیم تصویر پس‌زمینه به کل صفحه اعمال شود از دستور زیر استفاده می‌کنیم

```
body{ background-image: url(tree.png);}
```

هنگامی که تصویری را به عنوان پس زمینه تعیین می‌کنیم رفتار پیش‌فرض مرورگر این است که تصویر را در بالا سمت چپ قرار می‌دهد و سپس در جهات افقی و عمودی آن را تکرار می‌کند تا پس‌زمینه پر شود. به این حالت `tiling` می‌گویند و با خاصیت `background-repeat` می‌توان آن را کنترل کرد.

```
background-repeat: type;
```

به طور پیش فرض تصاویر پس زمینه در بالا و سمت چپ المان قرار می گیرند. اما به کمک `background-position` می توان موقعیت آن را تغییر داد.

`background-position: horizontal vertical;`

که مقادیر `horizontal` و `vertical` مختصات تصویر را تعیین می کنند .

`background-position: 20% 10% ;`

یعنی تصویر به اندازه 20 درصد طول المان (`width`) از سمت چپ و 10 درصد ارتفاع از بالا قرار بگیرد.

اگر فقط یک مقدار تعیین کنیم آن مقدار را برای هر دو جهت افقی و عمودی در نظر می گیرد.

`background-position: 30px;`

می توانیم از کلمات کلیدی `right, left, top, bottom` هم استفاده کنیم

`background-position: right bottom;`

یعنی تصویر پس زمینه در سمت راست و پایین المان قرار بگیرد. همانطور که قبلا هم اشاره شد به طور پیش فرض تصاویر پس زمینه فقط به محتوای المان و قسمت `padding` اعمال می شود برای کنترل این رفتار از `background-clip` استفاده می کنیم.

`background-clip: type;`

`repeat`: در دو جهت افقی و عمودی تصویر تکرار می شود

`repeat-x`: تصویر در جهت افقی تکرار می شود.

`repeat-y`: تصویر در جهت عمودی تکرار می شود.

`no-repeat`: تصویر تکرار نمی شود.

`round`: تصویر در دو جهت افقی و عمودی تکرار می شود ولی تغییر اندازه می دهد تا همه تصاویر در فضای موجود جا بگیرند.

`space`: تصویر در دو جهت افقی و عمودی تکرار می شود و فضای خالی طوری بین آنها قرار می گیرد که بدون تغییر اندازه تصویر در فضای موجود جا بگیرد.

تصویر پس زمینه به المان مربوطه متصل می شود و با اسکرول محتوای المان، پس زمینه نیز اسکرول می شود این رفتار را می توان با `background-attachment` کنترل کرد.

`background-attachment: type;`

که `type` می تواند مقادیر زیر را دریافت کند.

`scroll` حالت پیش فرض است که تصویر پس زمینه هنگام اسکرول صفحه اصلی حرکت می کند ولی با اسکرول المان حرکت نمی کند.

`fixed` تصویر پس زمینه ثابت است و حرکت نمی کند.

`local` با اسکرول صفحه اصلی و اسکرول المان تغییر می کند.

همچنین می‌توان از کلمات `cover` , `auto` و `contain` نیز استفاده کرد.

`auto`: براساس اندازه واقعی عکس

`background-size:auto 200px;`

ارتفاع پس‌زمینه `200px` و مقدار `width` به طور اتوماتیک طوری تغییر اندازه می‌دهد که نسبت طول به عرض تصویر حفظ شود.

`cover`: تصویر طوری تغییر اندازه می‌دهد که کل فضا را بپوشاند البته با حفظ نسبت طول به عرض تصویر یعنی ممکن است قسمتی از تصویر `crop` شود.

`contain`: تصویر طوری تغییر اندازه می‌دهد که کل تصویر در فضای المان جا بگیرد و بنابراین ممکن است فضای خالی به وجود آید

`background-size: 200px 300px;` `background-size: cover;` `background-size: contain;`



که `type` یکی از مقادیر زیر را دریافت می‌کند

`content-box`: پس زمینه فقط به قسمت محتوای المان اعمال می‌شود.

`padding-box`: پس زمینه به قسمت محتوا و `padding` اعمال می‌شود.

`border-box`: پس زمینه به قسمت محتوا و `padding` و `border` اعمال می‌شود

همینطور هنگامی که مختصات تصویر پس‌زمینه را تعیین می‌کنیم نسبت به قسمت `padding` محاسبه می‌شود اگر بخواهیم این رفتار نیز تغییر کند از خاصیت `background-origin` استفاده می‌کنیم

`background-origin: type;`

که `type` می‌تواند یکی از مقادیر زیر را دریافت کند.

`content-box`: مختصات نسبت به قسمت محتوای المان محاسبه می‌شود.

`padding-box`: مختصات نسبت به `padding` المان محاسبه می‌شود.

`border-box`: مختصات نسبت به `border` المان محاسبه می‌شود.

به طور پیش‌فرض اندازه تصویر پس‌زمینه برابر اندازه واقعی عکس می‌باشد اما می‌توان این اندازه را به کمک خاصیت `background-size` تغییر داد.

`background-size: width height;`

که `width` و `height` طول و عرض تصویر را تعیین می‌کند.

`background-size: 300px 200px;`

`background-size: 100% 50%;`

یا

به جای استفاده از خاصیت‌های فوق به طور جداگانه می‌توان از خاصیت `background` به صورت کوتاه شده نیز استفاده کرد.

```
header{
  background-image: url(back2.png),url(tree.png);
  background-position: top left, center center;
  background-repeat: no-repeat;
  background-color: red;}
```

```
background:color url(img) position / size repeat
attachment origin clip;
```

در اینجا `background-repeat` و `background-color` برای هر دو پس زمینه فوق تکرار شده‌اند.

```
body{
  background:red url(tree.png)
  center center / 10% 10%
  no-repeat fixed content-box content-box;}
```

ذکر همه موارد فوق الزامی نیست و اگر نویسیم مقادیر پیش فرض در نظر گرفته می‌شوند.

می‌توانیم به بک‌مان چندین پس‌زمینه اعمال کنیم که لازم است برای هر کدام جداگانه خصوصیت‌ها را تعیین کنیم.

```
header{
  background:url(back2.png) top left no-repeat,
  url(tree.png) center center no-repeat;}
```

HTML

```
<body>
  <p>
    HTML is the standard markup language for
    creating web pages and web applications....
  </p>
  <p>
    HTML is the standard markup language for
    creating web pages and web applications...
  </p>
  <p>
    HTML is the standard markup language for
    creating web pages and web applications...
  </p>
</body>
```

CSS

```
p{
  width:400px;
  min-height:10px;
  max-height:20px;}
```

خروجی

HTML is the standard markup language for creating web pages and web applications. HTML is the standard markup language for creating web pages and web applications. HTML is the standard markup language for creating web pages and web applications. HTML is the standard markup language for creating web pages and web applications. HTML is the standard markup language for creating web pages and web applications.

همچنین می‌توان ابعاد حداقل و حداکثر جعبه را تعیین کرد. در این صورت این اطمینان را خواهیم داشت که اندازه جعبه از یک مقدار کمتر و از یک مقدار بیشتر نخواهد شد.

- min-width** حداقل طول جعبه را تعیین می‌کند.
- max-width** حداکثر طول جعبه را تعیین می‌کند.
- min-height** حداقل ارتفاع جعبه را تعیین می‌کند.
- max-height** حداکثر ارتفاع جعبه را تعیین می‌کند.

به طور مثال با تنظیم **max-width** می‌توانیم مطمئن شویم که اندازه جعبه متن در مانیتورهای بزرگ، از یک حد بیشتر نخواهد شد. اما باید یک نکته را در نظر گرفت که در صورتی که با تعیین طول و عرض فضا برای محتویات جعبه کم باشد ممکن است صفحه به هم ریخته نمایش داده شود. به مثال روبرو توجه کنید

در این مثال اندازه متن حداقل 400×10 و حداکثر 400×20 پیکسل تعیین شده است و چون متن درون جعبه‌های تعیین شده جا نمی‌گیرد به هم ریختگی را مشاهده می‌کنید. برای اینکه این مشکل از بین برود یا باید اندازه‌ها را تغییر داد و یا برای جعبه‌ها اسکروول قرار داد.

HTML

```

<body>
  <p class="one" >
    HTML is the standard markup language for
    creating web pages and web applications.
  </p>
  <p class="two" >
    HTML describes the structure of Web pages
    HTML describes the structure of Web pages using
    markup HTML elements are the building blocks of
    HTML pages ....
  </p>
</body>

```

در مثال قبل گفتیم برای جلوگیری از به هم ریختگی متن می‌توانیم برای جعبه‌ها اسکرول تعریف کنیم این کار با کمک خصوصیت overflow انجام می‌شود.

overflow دارای دو مقدار می‌باشد:

hidden: باعث می‌شود محتویات اضافه پنهان شود.

scroll: باعث می‌شود یک اسکرول جهت نمایش محتویات اضافه جعبه

نمایش داده شود.

در این مثال محتویات اضافه پاراگراف اول پنهان شده ولی محتویات پاراگراف

دوم به صورت اسکرول نمایش داده می‌شود.

CSS

```

p{
  width:400px;
  min-height:10px;
  max-height:20px;}
.one{overflow:hidden;}
.two{overflow:scroll;}

```

خروجی

HTML is the standard markup language for creating web pages and web applications.

HTML describes the structure of Web pages using markup
HTML elements are the building blocks of HTML pages