

کتاب آموزش CSS3

آموزش فارسی

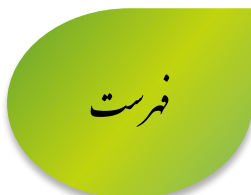
www.Amozeshfarsi.ir

فهرست مطالب

فصل اول	۹
مقدمه CSS	۱
نسخه های CSS	۲
نحوه استفاده از دستورات CSS در HTML	۳
قواعد دستوری CSS	۵
فصل دوم	۸
واحدهای اندازه گیری در CSS	۹
قواعد نوشتن واحدها	۱۱
رنگ ها در CSS	۱۱
نحوه بدست آوردن کد رنگ ها	۱۴
فصل سوم	۲۹
انتخابگر ها (Selectors) در CSS	۳۰
ساختار انتخابگر ها	۳۱
نحوه استفاده از دستورات CSS	۳۲
انواع انتخابگر ها (Types of Selectors)	۳۴
فصل چهارم	۴۶
Properties در CSS	۴۸
مبحث اول: Background	۴۹
مبحث دوم: Text	۵۵
مبحث سوم: Fonts	۶۰

۶۵	مبحث چهارم Links
۶۸	مبحث پنجم: لیست ها List
۷۳	مبحث ششم: Tables
۷۵	مبحث هفتم: Box Model
۸۱	مبحث هشتم: Border
۸۴	مبحث نهم: ساخت اشکال با Border
۹۰	مبحث دهم: outline
۹۱	مبحث یازدهم: Position
۹۷	مبحث دوازدهم: تنظیم طول و عرض
۹۹	مبحث سیزدهم: Direction
۱۰۰	مبحث چهاردهم: Clear and float
۱۰۶	مبحث پانزدهم: box-shadow
۱۱۲	مبحث شانزدهم: Box-sizing
۱۱۵	مبحث هفدهم: Gradient
۱۱۷	مبحث هجدهم: Transform
۱۲۱	مبحث نوزدهم: Transition
۱۲۳	مبحث بیستم: Cursor
۱۲۴	فصل چهارم
۱۲۵	قانون @media در CSS چیست؟
۱۲۵	انواع Media

۱۲۶ نحوه استفاده از Media
۱۲۸ فصل ششم
۱۲۹ گروه بندی عناصر صفحه با استفاده از تگ های Div و Span
۱۳۰ آشنایی با نحوه طرح بندی و طراحی قالب سایت
۱۳۰ طرح بندی صفحات (Float-based)
۱۳۱ ایجاد طراح (قالب) های شناور دو ستونه
۱۳۴ ایجاد طرح های سه ستونه شناور
۱۳۶ منابع



فصل اول

مباحث فصل اول

✓ مقدمهء CSS3

✓ نسخه های CSS

✓ نحوه استفاده از دستورات CSS

✓ قواعد دستوری CSS

مقدمه CSS

قبل از اینکه CSS را تعریف کنیم، باید راجع به صفحات وب توضیح بدهیم یک صفحه وب شامل اجزای زیر میباشد:

Content (محتوا): این بخش وب بوسیله HTML ایجاد میشود.

Presentation (نمایش): این بخش وب بوسیله CSS ایجاد میشود.

Behavior (رفتار): این بخش وب بوسیله JavaScript ایجاد میشود.

لایه محتوا همیشه وجود دارد، چون در این لایه تمامی محتویات یک صفحه وب که به قصد نمایش ایجاد میگردد نوشته میشود که این نوشته ها بین تگ های HTML , XHTML قرار میگیرد. خلاصه میتوان گفت لایه محتوا قالب بندی یک سایت را بر عهده دارد.

با استفاده از عناصر و تگ های HTML می توانیم متون، تصاویر، ویدئو ها، صوت ها و ... را کنار هم قرار بدهیم و برای کاربر ای سایت نمایش بدهیم

لایه نمایش وظیفهء تنظیم و ترتیب محتویات سایت را که برای کاربر نمایش داده میشود برعهده دارد. حدود چند سال پیش برای لایه بندی سایت ها از جدول استفاده میکردند که به این سبک کد نویسی Table Base می نامند که در این سبک کد نویسی با ادغام چند سطر ستون ایجاد می شد که البته این سبک کد نویسی یکسری معایب در بر داشت از جمله سنگین شدن سایت نداشتن زیبایی در هنگام لود و همچنان برای SEO (Search Engine Optimization)

سایت هم چندان مناسب نبود. البته باید یاد آور شد در آن هنگام نیز از CSS استفاده میشد اما نه به وسعت امروز که امروزه طراحی Table Base جای خود را به Div base داده است.

از زمانی که سبک کد نویسی تغییر و پیشرفت کرد استفاده از CSS یک امر عادی گشته است و با استفاده از Div میتوانید سایت را به ساده گی قالب بندی کرد و محتوا سایت را به شکل سایت را به معرض نمایش در آورید.

لایه رفتار کنترل نحو تعامل کاربر با صفحه را برعهده دارد که معمولاً این لایه توسط زبان JavaScript ایجاد میگردد.



البته هر سه این لایه میتوان یکجا استفاده کرد اما به دلیل آسانی کار و کنترل هر چه خوبتر این لایه ها را از همدیگر جدا کرده اند.

نسخه های CSS

CSS در نسخه های مختلف به بازار عرضه شده است که اولین نسخه ی CSS ، نسخه ۱ بود که در ماه دسامبر سال ۱۹۹۶ توسط کنسرسیوم تعیین استانداردهای وب (w3c) منتشر شد. در این نسخه ، خواصی برای تعیین نوع فونت ، اندازه متن ، فاصله ی عناصر و ایجاد لیست ها ، اندازه ی فضای اشغالی توسط تگ ها و تعیین حاشیه برای آنها پیش بینی شده بود . CSS1 برخی موارد ضروری برای تعیین ظاهر عناصر همانند تعیین رنگ ها و تصویر پس زمینه و رنگ متون آنها یا تعیین مکان یک تگ برای قرارگیری در سمت چپ یا راست را به همراه نداشت.

CSS نسخه ۲ در سال ۱۹۹۸ معرفی شد و تعداد زیادی از خواصی را که طراحان صفحات وب به آنها نیاز داشتند در برداشت. امکانات که این نسخه را در بر میگرفتند تعیین مکان آنها در جهات مختلف ، Selector ها ، ادغام فایل های CSS و بسیاری امکانات دیگر از این جمله بودند.

پیاده سازی برخی از استانداردهای CSS نسخه ۲ بسیار مشکل بود . به همین دلیل ، کنسرسیوم تعیین استانداردهای وب تصمیم به اصلاح آنها گرفت و نام نسخه ی اصلاح شده را " Cascading Style Sheets , Level 2 Revision 1 " گذاشت که به اختصار CSS2.1 نامیده می شود .

امروزه اگر صحبتی از CSS نسخه ی ۲ به میان می آید ، مقصد همان CSS نسخه ۲/۱ است ؛ چون CSS نسخه ی ۲ هیچگاه توسط مرور گر ها پذیرفته نشد به همین دلیل از این نسخه هیچ استفاده نه گردید.

کار بر روی نسخه ۳ به مدت چهار سال طول کشید و روند کندی داشت . در CSS3 قابلیت های پیش بینی شده به قسمت های مختلفی تقسیم شده اند . در حقیقت نوعی دسته بندی بر روی قابلیت های مختلف صورت پذیرفته است تا هر قسمت از قسمت دیگر مجزا باشد . هم اکنون برخی از امکانات CSS3 توسط برخی مرورگر های قدیمی پشتیبانی نمی شود .



برخی از امکاناتی که در CSS3 در نظر گرفته شده اند ، شامل : ایجاد متون چند ستونی ، حاشیه های گرد، تعیین میزان شفافیت عنصر ، استفاده از رنگ هایی که در مدهای HSLA , HSL , RGBA تعریف شده اند و ایجاد سایه برای متون و ... است . بهترین مزیت استفاده از نسخه ۳ این است که شما می توانید یک قالب را با استفاده از تکنیک های جدید CSS بدون نیاز به تصاویر با کیفیت عالی کدنویسی کنید .

البته باید گفت که در این دوره آموزشی نسخه CSS3 را مورد بحث قرار میدهیم که البته در زمانی که این کتاب نشر گردیده است آخرین نسخه آن همین نسخه میباشد.

نحوه استفاده از دستورات CSS در HTML

قسمی که میدانید CSS در لایه دوم وب کار میکند و بدون لایه محتوا (HTML) هیچ است از این سبب ضرورت است تا دستورات CSS و HTML را با هم ارتباط دهیم.

به سه روش میتوانید دستورات CSS را به یک صفحه وب (صفحات HTML) معرفی کنید . که شامل روش های زیر میشود :

برگه های سبک خارجی (External Style Sheet): این نوع سبک استفاده از CSS زمانی موثر است که اگر خواسته باشید یک استایل را بالای چندین صفحه HTML تطبیق کنید ، که با این روش تنها با تغییر دادن یک فایل استایل کلی صفحه را تغییر میکند.

در این سبک دستورات CSS در یک فایل مجزا با پسوند CSS ذخیره شده و این فایل به تمامی صفحات HTML که میخواهید استایل بالای آن تطبیق شود لینک میدهیم.

شما میتوانید برای این کار از تگ Link که جز تگ های تهی است استفاده کنید. البته باید یاد آور شد که این Link باید در بین تگ Head نوشته شود.

آموزش کاربرد CSS3
<head>
<link rel="Stylesheet" type="text/css" href="Style.css">
</head>



نکته: در تگ Link بالا صفت rel نوع رابطه صفحه CSS را مشخص میکنید که همیشه قیمت آن برابر به Stylesheet میباشد. Type نوعیت سند را که ارتباط میدهید مشخص میکنید و href هم عبارت از همان سند مجزاء CSS تان میباشد.

روشی دیگری نیز برای ایجاد ارتباط یک سند CSS با سند HTML میباشد و آن هم استفاده از قانون @import همراه با تگ Style میباشد.

به عنوان مثال فایل Stylee.css را به صورت زیر با فایل HTML ملحق میکنیم.

البته باید یادآور شد که تگ Style هم در بین تگ Head نوشته شود.

آموزش کاربردی CSS3
<Style type="text/css">
@import url (style.css)
</style>

برگه های سبک داخلی (Internal Style sheet):

این روش معمولاً وقتی استفاده میشود که خواسته باشید که هر صفحه استایل مجزاء داشته باشد. این روش با تگ Style در بین تگ Head اجرا میشود.

آموزش کاربردی CSS3
<head>
<Style type="text/css">
div{background :#fff;
Font-size:20px;
Width:180px;}
</style>
</head>



برگه های سبک خطی (Inline Style Sheet):

در این حالت استایل ها تنها در یک تگ مربوط که با Style آغاز شده تطبیق میگردد البته استفاده از این روش تنها در موارد خاصی میتوان استفاده کرد در غیر آن استفاده از این روش پیشنهاد نمیشود.

آموزش کاربردی CSS3

`<p style="background-color:"red";color:yellow> this is a paragraph </p>`

قواعد دستوری CSS

در این قسمت قواعد دستوری CSS را مورد بحث قرار میدهم.

در کد نویسی فضاهای خالی و خطوط اضافه برای درک بهتر و خوانایی بیشتر کدها به تعداد دلخواه هیچ مشکلی ندارد.

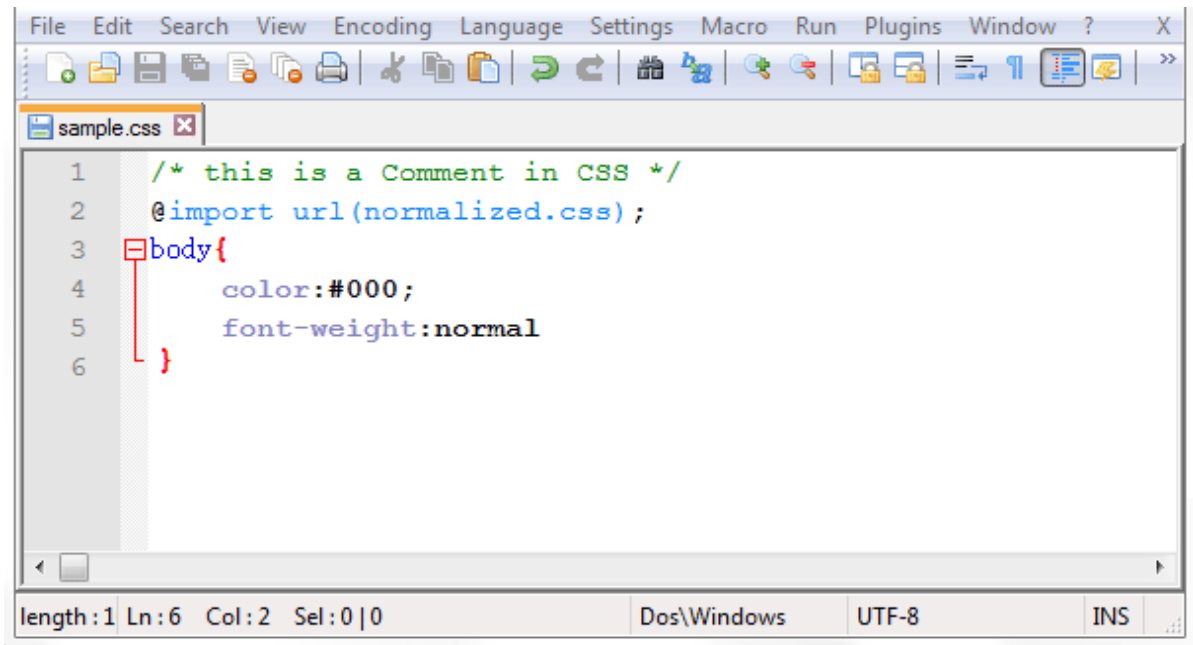
CSS نسبت به حروف بزرگ و کوچک حساس نیست. البته در برخی موارد ، عوامل خارجی مثل نوع زبانی که از CSS استفاده می کند و یا سیستم عامل در این بین نقش دارند. برای استانداردسازی کارتان همیشه دستورات را با حروف کوچک بنویسید.

مثلا نام تگ ها در HTML به حروف بزرگ و کوچک حساس نیستند اما در XHTML این حساسیت وجود دارد. نام فونت ها هم به استثنای فونت های عمومی که برای CSS شناخته شده هستند ، در برخی سیستم عامل ها این حساسیت را دارند.

خوب میرویم طرف قواعد دستوری CSS ، برای نوشتن دستورات CSS میتوانید از نرم افزارهای مختلف استفاده کنید. نرم افزار های که معمولاً استفاده میشود ++notepad یا Dream Weaver است.

برای شروع به تصویر زیر دقت کنید:





```

1  /* this is a Comment in CSS */
2  @import url(normalized.css);
3  body{
4      color:#000;
5      font-weight:normal
6  }

```

The screenshot shows a text editor window titled 'sample.css'. The code contains a comment, an @import statement, and a body selector with color and font-weight properties. The status bar at the bottom indicates 'length:1 Ln:6 Col:2 Sel:0|0', 'Dos\Windows', 'UTF-8', and 'INS'.

در تصویر بالا:

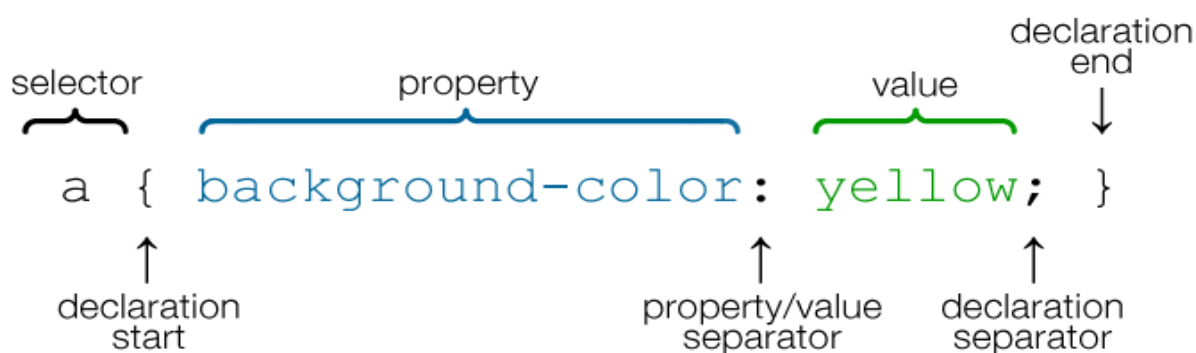
خط اول کامنت (توضیحات) را نشان میدهد که شما قادرید در هر قسمت از صفحه CSS برای آسانی کار و همچنان برای جلوگیری از سردرگمی از کامنت استفاده کنید. برای درج کردن کامنت (توضیحات) باید متن تانرا بین علامه /* و */ قرار بدهید. البته شما نیز میتوانید توضیحات را به زبان فارسی درج کنید.

یادداشت: در صورت درج کردن کامنت فارسی باید در هنگام ثبت کردن سند Encoding آنرا UTF-8 را انتخاب نمایید. برای انجام این کار در برنامه Notepad++ میتوانید از مینوی Encoding نوعیت کد آنرا انتخاب نمایید.

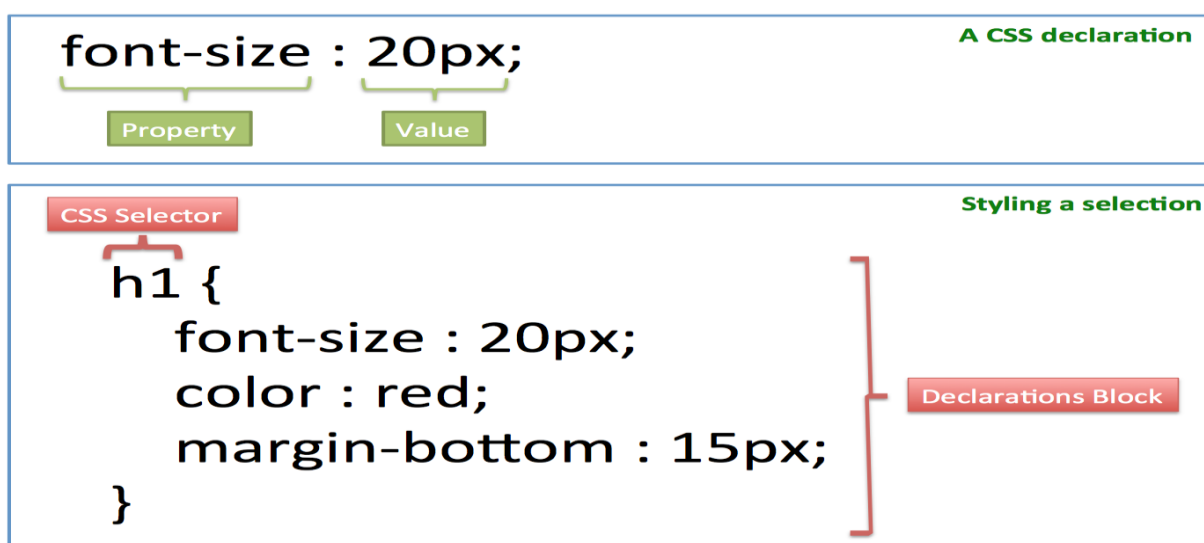
خط دوم نشان دهنده یک قانون و یک دستور است که فایل بالا را به سند CSS ارتباط میدهد. خط سوم دستوری است که مجموعه ای از قواعد را در برمیگیرد. در این دستور که مشاهده می کنید ما به صفحه که همان body محسوب میشود گفتیم تمامی رنگ فونت ها #000 باشد و ضخامت فونت ها هم نرمال باشد.



همانطور که مشاهده می کنید، برای نوشتن قواعد ابتدا نام Selector را مینویسیم سپس با علامت " { " بلاک را باز می کنیم و قواعد را نوشته می کنیم و سپس با علامت " } " بلاک را بسته میکنیم. قواعد در بلاک ها با کاراکتر " ; " از هم جدا میشوند که برای قاعده آخر نیازی به این کاراکتر نیست چرا که قاعده ای بعد از آن نیست اما در صورت استفاده نیز کدام مشکلی ندارد. هر قاعده از یک نام و یک مقدار تشکیل میشود که با علامت " : " از هم تفکیک میشوند. به تصویر زیر دقت کنید:



شکل زیر مطلب بالا را به شکل مفصل تر تشریح میکند.



فصل دوم

مباحث فصل دوم

✓ واحدهای اندازه گیری در CSS3

✓ نحوه نوشتن واحدها

✓ رنگ ها در CSS3

✓ نحوه بدست آوردن کد رنگ ها



واحد های اندازه گیری در CSS

واحد های اندازه گیری عبارت از همان واحد های است که بیشتر در صفحات وب استفاده میشود. مرورگر ها پیش فرض برای تمامی فونت ها یک اندازه دارد تا در صورت تعیین نکردن اندازه به متن ، متن قابل نمایش باشد.

میباشد. px,em,% در طراحی وب واحد های مختلف استفاده میگردد اما واحد استاندارد آن **Pixel(PX)**: پیکسل واحد اندازه گیری معمول در طراحی وب است که اکثر کدنویسان از این واحد اندازه گیری استفاده می کنند.

% (Percentage): واحد درصدی است که در طراحی واکنش گرا استفاده میشود .

Em: واحدی است که اندازه آن به اندازه ی فونت پدر تگ جاری بستگی دارد. به عنوان مثال اگر اندازه ی فونت یک تگ برابر با ۱۰ پیکسل باشد و اندازه فونت تگی که درون این تگ قرار دارد به اندازه ی ۲em تعیین شده باشد ، اندازه ی فونت تگ فرزند دو برابر تگ پدر میشود ، یعنی ۲۰ پیکسل .

Rem: مشکلی که واحد em دارد اینست که سایزش نسبت به عنصر والد تعیین میشود و ممکن است دچار مشکل شوید ، برای حل این مشکل میتوانید در کنار em از rem هم استفاده کنید.

واحد rem هم مثل em عمل میکند با این تفاوت که سایز فونت نسبت به سایز تعیین شده برای عنصر root مشخص میشود (مثل html , body) .

نکته : این واحد ها تنها در فونت محدود نمیشود بلکه شما میتوانید آنرا برای خاصیت های دیگری چون Margin,padding,width,height و امثال اینها در CSS استفاده نمایید .



واحدهای اندازه گیری در وب با واحدهای اندازه گیری در چاپ متفاوت است ، در این جدول زیر واحدهای مناسب برای هر محیط توضیح داده شده :

پیشنهاد نمیشود!	پیشنهاد میشود	مورد استفاده
pt, cm, mm, in	em, px, %	استفاده برای نمایش در وب
	em, cm, mm, in, pt, %	استفاده برای چاپ روی کاغذ

در جدول زیر میتوانید واحدهای اندازه گیری را ببینید که میتوانید از آنها در صفحات وب استفاده نمایید.

واحد	توضیحات
%	در صد
in	اینچ
cm	سانتی متر
mm	میلی متر
em	یعنی دوبرابر اندازه به 2em یعنی اندازه کنونی سائز (که از تگ پدر ارث برده). 1em یک ارث برده شده از تگ پدر.
pt	برابر است با ۱/۷۲ اینچ 1pt، هر Point یا
px	پیکسل، یکی پیکسل یک نقطه روی صفحه نمایش مانیتور محسوب میشود.



قواعد نوشتن واحدها

کد زیر نشان دهنده چگونگی استفاده نوشتن واحد ها در صفحات وب است:

آموزش کاربردی CSS3
H1{
Font-size:12px;
}

در این کد ما فونت عنصر h1 را سایز ۱۲ پیکسل را قرار دادیم . عدد و واحد اندازه گیری بدون فاصله در کنار یکدیگر قرار میگیرند در ضمن بعد از ختم دستور شما باید از علامه ; استفاده نماید. اما اگر شما تنها یک Properties درج کرده باشید ضرورت به این علامت نیست.

رنگ ها در CSS

خوب حالا در رابطه با رنگ ها صحبت می کنیم .

برای نمایش رنگ ها در CSS از میتود های زیر میتوانید استفاده کنید :

- Hexadecimal colors
- RGB colors
- RGBA colors
- HSL colors
- HSLA colors
- color names

Hexadecimal Colors: این نوع رنگ یعنی رنگ های هگزادسیمال در تمامی مرورگر ها پشتیبانی میشود. این کد رنگ از رنگ های مکمل #RRGGBB تشکیل میشود . RR یعنی رنگ سرخ به اصطلاح انگلیسی Red – GG یعنی رنگ سبز به اصطلاح انگلیسی Green – BB یعنی رنگ آبی و به اصطلاح انگلیسی Blue و بصورت زیر استفاده میشود:

آموزش کاربردی CSS3
#box1{background-color:#ff0000;} /*Red*/
#box2{background-color:#00ff00;} /*Green*/
#box3{background-color:#0000ff;} /*Blue*/



RGB Color: رنگ های RGB در تمامی مرورگر ها پشتیبانی میشود. کد رنگ RGB هم از رنگ های مکمل تشکیل میشود و بصورت `rgb(red, green, blue)` نمایش داده میشود و بصورت زیر استفاده میشود

آموزش کاربردی CSS3
<code>#box1{background-color:rgb(255,0,0);} /*Red*/</code>
<code>#box2{background-color:rgb(0,255,0);} /*Green*/</code>
<code>#box3{background-color:rgb(0,0,255);} /*Blue*/</code>

RGBA Color: این رنگ ها شبیه به رنگ های RGB هستند با این تفاوت که در این رنگ ها شما می توانید میزان شفافیت (Opacity) رنگ را تعیین کنید. نحوه ی نوشتن این کد رنگ بصورت زیر است.

آموزش کاربردی CSS3
<code>#box1{background-color:rgb(255,0,0,0.3);} /*Red with opacity */</code>
<code>#box2{background-color:rgb(0,255,0,0.3);} /*Green with opacity*/</code>
<code>#box3{background-color:rgb(0,0,255,0.3);} /*Blue with opacity*/</code>

برای تغییر میزان شفافیت عدد ۰/۳ را میتوانید تنظیم کنید. هر چقدر عدد ۳ بیشتر شود شفافیت بیشتر میشود و هر چقدر که کمتر شود شفافیت هم کمتر میشود.

نوت: رنگ های RGBA در مرورگر های IE9+, Firefox 3+, Chrome, Safari, Opera پشتیبانی میشود.

HSL Color: این رنگ زمانی استفاده میشود که اگر شما خواسته باشید که یک رنگ را پررنگ و کم رنگ کنید. قاعده نوشتن این کدها بصورت (hue, saturation, lightness) است که.

HUE یک درجه رنگی روی چرخه ی رنگه که از ۰ تا ۳۶۰ قابل تنظیم است. عدد ۰ یا ۳۶۰ تعریف کننده رنگ سرخ است، عدد ۱۲۰ نشان تعریف کننده رنگ سبز است و عدد ۲۴۰ هم نشان دهنده رنگ آبی است.



Saturation بصورت درصدی تعیین میشود و حلقه ای از رنگ خاکستری به رنگ اصلی اضافه می کند. هرچه عدد این درصد کمتر باشد رنگ خاکستری بیشتری با رنگ اصلی ترکیب میشود و رنگ را تیره تر و مایل به خاکستری میکند.

Lightness بصورت درصدی تعیین میشود و رنگ را متمایل به رنگ سیاه و یا سفید تبدیل میکند. هرچه این عدد کمتر باشد رنگ تیره تر میشود و به رنگ سیاه متمایل میشود و هرچه این عدد بیشتر باشد رنگ سفید بیشتری با رنگ اصلی ترکیب میشود و رنگ را روشن میکند.

نحوه ی استفاده از این کد بصورت زیر است:

آموزش کاربردی CSS3
#box1{background-color:hsl(120,100%,50%);} /*Green */
#box2{background-color:rgb(120,100%,75%);} /*light green*
#box3{background-color:rgb(120,100%,25%);} /*Dark Green*/
#box4{background-color:(120,60%,70%);} /*pastel Green*/

نوت: رنگ های HSL در مرورگر های IE9+, Firefox, Chrome, Safari, Opera 10 پشتیبانی میشود.

HSLA Color: این نوع رنگ دقیقا شبیه به رنگ های HSL عمل می کنند با این تفاوت که میتوانید میزان شفافیت رنگ ها را تنظیم کنید. این کد بصورت زیر نوشته میشود:

آموزش کاربردی CSS3
#box1{background-color:hsl(120,100%,50%,0.3);} /*Green with opacity */
#box2{background-color:rgb(120,100%,75%,0.3);} /*light green with opacity *
#box3{background-color:rgb(120,100%,25%,0.3);} /*Dark Green*/
#box4{background-color:(120,60%,70%,0.3);} /*pastel Green*/

Name: نوعی رنگی دیگری که شما میتوانید استفاده کنید کدهای رنگی از نام استاندارد رنگ ها میباشد.

آموزش کاربردی CSS3
#box1{background-color:red;}
#box2{background-color:Green;}
#box3{background-color:blue;}





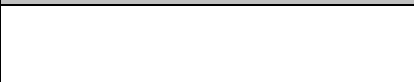
شاید این سوال برای تان بیش بیاید که چگونه میتوانیم این همه رنگ ها را به خاطر بسپاریم و کد رنگ دلخواه خود را چگونه بدست بیاوریم؟

نحوه بدست آوردن کد رنگ ها

برای تبدیل کد رنگ هگزادسیمال به رنگ های RGB میتوانید از سایت <http://hex2rgba.devonth.com> استفاده کنید .

و همچنان میتوانید از لیست رنگ ها که در پایین قرار دادیم استفاده کنید

کمترین مقدار RGB برای نمایش یک رنگ ۰ و بیشترین مقدار آن ۲۵۵ است با ترکیب رنگ های سرخ و سبز و آبی از مقدار ۰ تا ۲۵۵ تعداد بالغ بر ۱۶ میلیون رنگ را بوجود می آید و شما میتوانید استفاده کنید . رنگ هایی که بر روی مانیتور قابل نمایش هستند چیزی حدود ۱۶۳۰۰ رنگ متفاوت است. و مانیتور ها قادر به نشان دادن دیگر رنگ ها نیستند.

Color	Color HEX	Color RGB
	#000000	rgb(0,0,0)
	#FF0000	rgb(255,0,0)
	#00FF00	rgb(0,255,0)
	#0000FF	rgb(0,0,255)
	#FFFF00	rgb(255,255,0)
	#00FFFF	rgb(0,255,255)
	#FF00FF	rgb(255,0,255)
	#C0C0C0	rgb(192,192,192)
	#FFFFFF	rgb(255,255,255)



جدول زیر نشان دهنده رنگ هایی است که از طیف رنگی سرخ تشکیل شدند ، به مقدار rgb آنها دقت کنید ، تنها طیف سرخ متغییر است:

Red Light	HEX	RGB
	#000000	rgb(0,0,0)
	#080000	rgb(8,0,0)
	#100000	rgb(16,0,0)
	#180000	rgb(24,0,0)
	#200000	rgb(32,0,0)
	#280000	rgb(40,0,0)
	#300000	rgb(48,0,0)
	#380000	rgb(56,0,0)
	#400000	rgb(64,0,0)
	#480000	rgb(72,0,0)
	#500000	rgb(80,0,0)
	#580000	rgb(88,0,0)
	#600000	rgb(96,0,0)
	#680000	rgb(104,0,0)
	#700000	rgb(112,0,0)
	#780000	rgb(120,0,0)
	#800000	rgb(128,0,0)



	#880000	rgb(136,0,0)
	#900000	rgb(144,0,0)
	#980000	rgb(152,0,0)
	#A00000	rgb(160,0,0)
	#A80000	rgb(168,0,0)
	#B00000	rgb(176,0,0)
	#B80000	rgb(184,0,0)
	#C00000	rgb(192,0,0)
	#C80000	rgb(200,0,0)
	#D00000	rgb(208,0,0)
	#D80000	rgb(216,0,0)
	#E00000	rgb(224,0,0)
	#E80000	rgb(232,0,0)
	#F00000	rgb(240,0,0)
	#F80000	rgb(248,0,0)
	#FF0000	rgb(255,0,0)



حالا به جدول رنگی زیر توجه کنید.

طیف رنگی سفید است که با ترکیب طیف های یکسان سرخ و سبز و آبی تشکیل میشود.

Gray Shades	HEX	RGB
	#000000	rgb(0,0,0)
	#080808	rgb(8,8,8)
	#101010	rgb(16,16,16)
	#181818	rgb(24,24,24)
	#202020	rgb(32,32,32)
	#282828	rgb(40,40,40)
	#303030	rgb(48,48,48)
	#383838	rgb(56,56,56)
	#404040	rgb(64,64,64)
	#484848	rgb(72,72,72)
	#505050	rgb(80,80,80)
	#585858	rgb(88,88,88)
	#606060	rgb(96,96,96)
	#686868	rgb(104,104,104)
	#707070	rgb(112,112,112)
	#787878	rgb(120,120,120)
	#808080	rgb(128,128,128)



	#888888	rgb(136,136,136)
	#909090	rgb(144,144,144)
	#989898	rgb(152,152,152)
	#A0A0A0	rgb(160,160,160)
	#A8A8A8	rgb(168,168,168)
	#B0B0B0	rgb(176,176,176)
	#B8B8B8	rgb(184,184,184)
	#C0C0C0	rgb(192,192,192)
	#C8C8C8	rgb(200,200,200)
	#D0D0D0	rgb(208,208,208)
	#D8D8D8	rgb(216,216,216)
	#E0E0E0	rgb(224,224,224)
	#E8E8E8	rgb(232,232,232)
	#F0F0F0	rgb(240,240,240)
	#F8F8F8	rgb(248,248,248)
	#FFFFFF	rgb(255,255,255)



در جدول زیر هم شما میتوانید یکسری رنگ های مورد استفاده در وب را همراه با کد hex آنها در طیف های مختلف ملاحظه کنید ، میتواند تا حدودی به طراحان گرافیک نیز کمک کند.

000000	000033	000066	000099	0000CC	0000FF
003300	003333	003366	003399	0033CC	0033FF
006600	006633	006666	006699	0066CC	0066FF
009900	009933	009966	009999	0099CC	0099FF
00CC00	00CC33	00CC66	00CC99	00CCCC	00CCFF
00FF00	00FF33	00FF66	00FF99	00FFCC	00FFFF
330000	330033	330066	330099	3300CC	3300FF
333300	333333	333366	333399	3333CC	3333FF
336600	336633	336666	336699	3366CC	3366FF
339900	339933	339966	339999	3399CC	3399FF
33CC00	33CC33	33CC66	33CC99	33CCCC	33CCFF
33FF00	33FF33	33FF66	33FF99	33FFCC	33FFFF
660000	660033	660066	660099	6600CC	6600FF
663300	663333	663366	663399	6633CC	6633FF
666600	666633	666666	666699	6666CC	6666FF
669900	669933	669966	669999	6699CC	6699FF
66CC00	66CC33	66CC66	66CC99	66CCCC	66CCFF



66FF00	66FF33	66FF66	66FF99	66FFCC	66FFFF
990000	990033	990066	990099	9900CC	9900FF
993300	993333	993366	993399	9933CC	9933FF
996600	996633	996666	996699	9966CC	9966FF
999900	999933	999966	999999	9999CC	9999FF
99CC00	99CC33	99CC66	99CC99	99CCCC	99CCFF
99FF00	99FF33	99FF66	99FF99	99FFCC	99FFFF
CC0000	CC0033	CC0066	CC0099	CC00CC	CC00FF
CC3300	CC3333	CC3366	CC3399	CC33CC	CC33FF
CC6600	CC6633	CC6666	CC6699	CC66CC	CC66FF
CC9900	CC9933	CC9966	CC9999	CC99CC	CC99FF
CCCC00	CCCC33	CCCC66	CCCC99	CCCCCC	CCCCFF
CCFF00	CCFF33	CCFF66	CCFF99	CCFFCC	CCFFFF
FF0000	FF0033	FF0066	FF0099	FF00CC	FF00FF
FF3300	FF3333	FF3366	FF3399	FF33CC	FF33FF
FF6600	FF6633	FF6666	FF6699	FF66CC	FF66FF
FF9900	FF9933	FF9966	FF9999	FF99CC	FF99FF
FFCC00	FFCC33	FFCC66	FFCC99	FFCCCC	FFCCFF
FFFF00	FFFF33	FFFF66	FFFF99	FFFFCC	FFFFFF







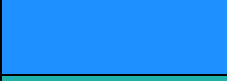














استفاده از نام رنگ ها

شما علاوه بر اینکه از کد رنگ ها استفاده میکنید ، میتوانید از نام رنگ ها هم استفاده کنید ، به جدول زیر دقت کنید ، برای هر کد hex و RGB یک نام بصورت استاندارد تعیین شده که میتوانید استفاده کنید.








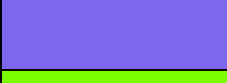









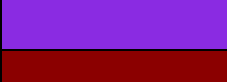

البته باید یک نکته را بدانید که بهتر است برای استاندارد سازی کارتان از نام رنگ ها استفاده نکنید و فقط از کد رنگ ها استفاده کنید.

Color Name	HEX	Color	Shades	Mix
Black	#000000		Shades	Mix
Navy	#000080		Shades	Mix
DarkBlue	#00008B		Shades	Mix
Medium Blue	#0000CD		Shades	Mix
Blue	#0000FF		Shades	Mix
Dark Green	#006400		Shades	Mix
Green	#008000		Shades	Mix
Teal	#008080		Shades	Mix
Dark Cyan	#008B8B		Shades	Mix
DeepSkyBlue	#00BFFF		Shades	Mix
DarkTurquoise	#00CED1		Shades	Mix
MediumSpringGreen	#00FA9A		Shades	Mix
Lime	#00FF00		Shades	Mix



SpringGreen	#00FF7F		Shades	Mix
Aqua	#00FFFF		Shades	Mix
Cyan	#00FFFF		Shades	Mix
MidnightBlue	#191970		Shades	Mix
DodgerBlue	#1E90FF		Shades	Mix
LightSeaGreen	#20B2AA		Shades	Mix
ForestGreen	#228B22		Shades	Mix
SeaGreen	#2E8B57		Shades	Mix
DarkSlateGray	#2F4F4F		Shades	Mix
LimeGreen	#32CD32		Shades	Mix
MediumSeaGreen	#3CB371		Shades	Mix
Turquoise	#40E0D0		Shades	Mix
RoyalBlue	#4169E1		Shades	Mix
SteelBlue	#4682B4		Shades	Mix
DarkSlateBlue	#483D8B		Shades	Mix
MediumTurquoise	#48D1CC		Shades	Mix
Indigo	#4B0082		Shades	Mix
DarkOliveGreen	#556B2F		Shades	Mix
CadetBlue	#5F9EA0		Shades	Mix



CornflowerBlue	#6495ED		Shades	Mix
MediumAquaMarine	#66CDAA		Shades	Mix
DimGray	#696969		Shades	Mix
SlateBlue	#6A5ACD		Shades	Mix
OliveDrab	#6B8E23		Shades	Mix
SlateGray	#708090		Shades	Mix
LightSlateGray	#778899		Shades	Mix
MediumSlateBlue	#7B68EE		Shades	Mix
LawnGreen	#7CFC00		Shades	Mix
Chartreuse	#7FFF00		Shades	Mix
Aquamarine	#7FFFD4		Shades	Mix
Maroon	#800000		Shades	Mix
Purple	#800080		Shades	Mix
Olive	#808000		Shades	Mix
Gray	#808080		Shades	Mix
SkyBlue	#87CEEB		Shades	Mix
LightSkyBlue	#87CEFA		Shades	Mix
BlueViolet	#8A2BE2		Shades	Mix
DarkRed	#8B0000		Shades	Mix



DarkMagenta	#8B008B		Shades	Mix
SaddleBrown	#8B4513		Shades	Mix
DarkSeaGreen	#8FBC8F		Shades	Mix
LightGreen	#90EE90		Shades	Mix
MediumPurple	#9370DB		Shades	Mix
DarkViolet	#9400D3		Shades	Mix
PaleGreen	#98FB98		Shades	Mix
DarkOrchid	#9932CC		Shades	Mix
YellowGreen	#9ACD32		Shades	Mix
Sienna	#A0522D		Shades	Mix
Brown	#A52A2A		Shades	Mix
DarkGray	#A9A9A9		Shades	Mix
LightBlue	#ADD8E6		Shades	Mix
GreenYellow	#ADFF2F		Shades	Mix
PaleTurquoise	#AFEEEE		Shades	Mix
LightSteelBlue	#B0C4DE		Shades	Mix
PowderBlue	#B0E0E6		Shades	Mix
FireBrick	#B22222		Shades	Mix
DarkGoldenRod	#B8860B		Shades	Mix



MediumOrchid	#BA55D3		Shades	Mix
RosyBrown	#BC8F8F		Shades	Mix
DarkKhaki	#BDB76B		Shades	Mix
Silver	#C0C0C0		Shades	Mix
MediumVioletRed	#C71585		Shades	Mix
IndianRed	#CD5C5C		Shades	Mix
Peru	#CD853F		Shades	Mix
Chocolate	#D2691E		Shades	Mix
Tan	#D2B48C		Shades	Mix
LightGray	#D3D3D3		Shades	Mix
Thistle	#D8BFD8		Shades	Mix
Orchid	#DA70D6		Shades	Mix
GoldenRod	#DAA520		Shades	Mix
PaleVioletRed	#DB7093		Shades	Mix
Crimson	#DC143C		Shades	Mix
Gainsboro	#DCDCDC		Shades	Mix
Plum	#DDA0DD		Shades	Mix
BurlyWood	#DEB887		Shades	Mix
LightCyan	#E0FFFF		Shades	Mix



Lavender	#E6E6FA		Shades	Mix
DarkSalmon	#E9967A		Shades	Mix
Violet	#EE82EE		Shades	Mix
PaleGoldenRod	#EEE8AA		Shades	Mix
LightCoral	#F08080		Shades	Mix
Khaki	#F0E68C		Shades	Mix
AliceBlue	#F0F8FF		Shades	Mix
HoneyDew	#F0FFF0		Shades	Mix
Azure	#F0FFFF		Shades	Mix
SandyBrown	#F4A460		Shades	Mix
Wheat	#F5DEB3		Shades	Mix
Beige	#F5F5DC		Shades	Mix
WhiteSmoke	#F5F5F5		Shades	Mix
MintCream	#F5FFFA		Shades	Mix
GhostWhite	#F8F8FF		Shades	Mix
Salmon	#FA8072		Shades	Mix
AntiqueWhite	#FAEBD7		Shades	Mix
Linen	#FAF0E6		Shades	Mix
LightGoldenRodYellow	#FAFAD2		Shades	Mix



OldLace	#FDF5E6		Shades	Mix
Red	#FF0000		Shades	Mix
Fuchsia	#FF00FF		Shades	Mix
Magenta	#FF00FF		Shades	Mix
DeepPink	#FF1493		Shades	Mix
OrangeRed	#FF4500		Shades	Mix
Tomato	#FF6347		Shades	Mix
HotPink	#FF69B4		Shades	Mix
Coral	#FF7F50		Shades	Mix
DarkOrange	#FF8C00		Shades	Mix
LightSalmon	#FFA07A		Shades	Mix
Orange	#FFA500		Shades	Mix
LightPink	#FFB6C1		Shades	Mix
Pink	#FFC0CB		Shades	Mix
Gold	#FFD700		Shades	Mix
PeachPuff	#FFDAB9		Shades	Mix
NavajoWhite	#FFDEAD		Shades	Mix
Moccasin	#FFE4B5		Shades	Mix
Bisque	#FFE4C4		Shades	Mix



MistyRose	#FFE4E1		Shades	Mix
BlanchedAlmond	#FFEBCD		Shades	Mix
PapayaWhip	#FFEFD5		Shades	Mix
LavenderBlush	#FFF0F5		Shades	Mix
SeaShell	#FFF5EE		Shades	Mix
Cornsilk	#FFF8DC		Shades	Mix
LemonChiffon	#FFFACD		Shades	Mix
FloralWhite	#FFFAF0		Shades	Mix
Snow	#FFFAFA		Shades	Mix
Yellow	#FFFF00		Shades	Mix
LightYellow	#FFFFE0		Shades	Mix
Ivory	#FFFFF0		Shades	Mix
White	#FFFFFF		Shades	Mix



فصل سوم

مباحث فصل سوم

✓ انتخابگرها (Selectors)

✓ ساختار انتخابگرها

✓ نحوه استفاده از انتخابگرها

✓ انواع انتخابگرها

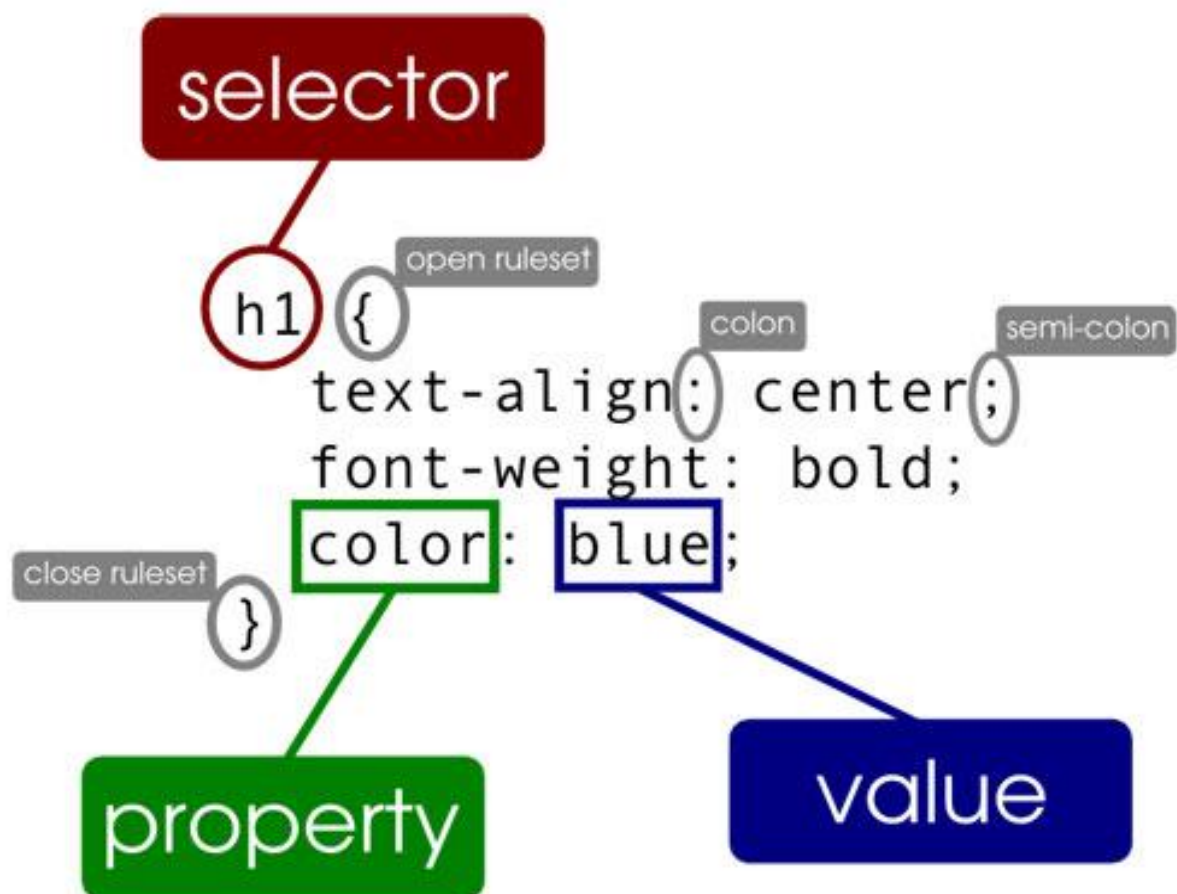


انتخابگرها (Selectors) در CSS

این فصل از جمله فصل های مهم این کتاب میباشد چون بدون انتخاب یک عنصر شما قادر نیستید به آن عنصر استایل دهی نمایید.

انتخابگرها ، الگوهایی هستند که برای انتخاب یک عنصر و استایل دهی به آن عنصر استفاده میشوند.

در این کتاب تا حد ممکن تمامی انتخابگرها از جمله انتخابگرهای CSS3 مورد بحث قرار داده شده. برای شناخت بهتر از انتخابگر به تصویر زیر دقت نمایید.



ساختار انتخابگر ها

تمامی انتخابگر های CSS به شکل درخت میباشد که البته این شکل درختی برای کنترل بهتر دستورات کمک بسزایی نموده است. برای درک بهتر از ساختار درختی آنرا به شکل روابط خانوادگی زیر تعریف میکنیم.

Ancestor (جد) 🏠

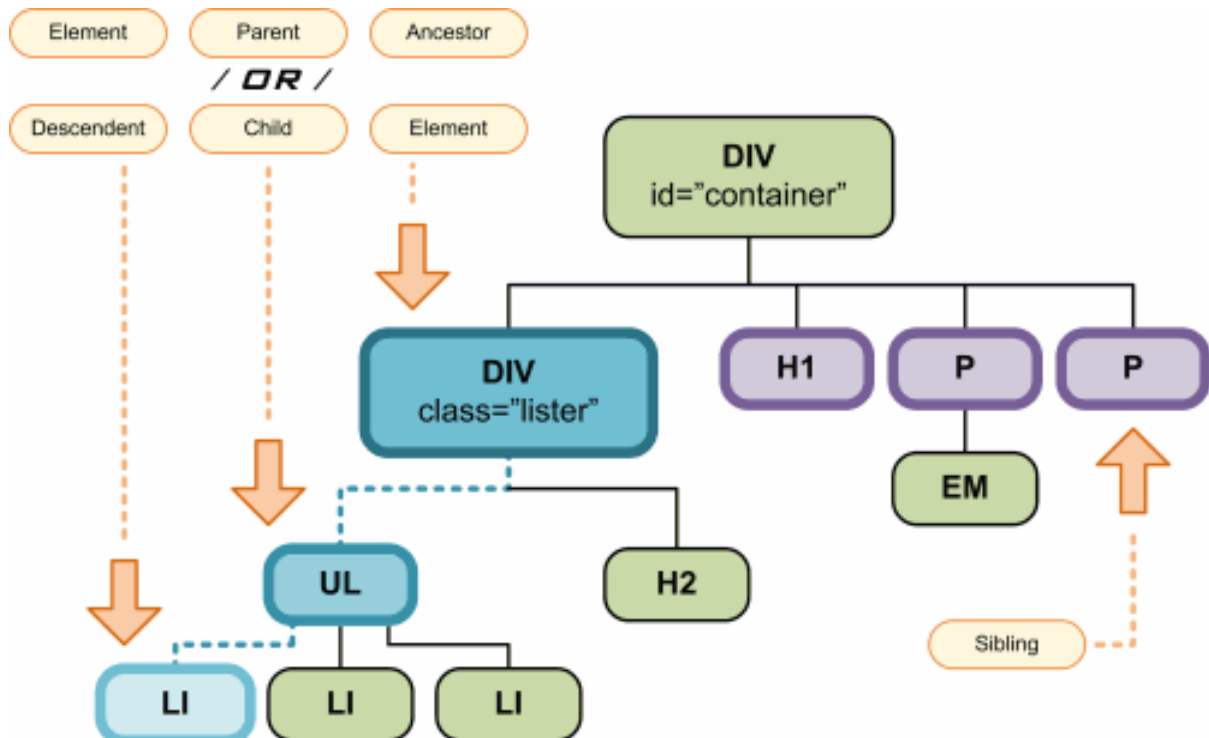
Descendants (اولاد) 🏠

Parents (پدر) 🏠

Children (فرزند) 🏠

Siblings (برادر) 🏠

برای درک هرچه بهتر از این موضوع در آخر فصل آتی دستوری را به ساختار درختی تبدیل میکنیم. اما حالا ساختار کلی درخت را در تصویر زیر مشاهده نمایید:



نحوه استفاده از دستورات CSS

در کل میتوانید از CSS به دو شکل استفاده نمایید!

۱- سراسری

۲- محلی

۱- **سراسری:** در سراسری باید در قسمت انتخابگر نام عنصر مورد نظر و در پی آن خصوصیت و مقدار آنرا در بین کروشر ({}) بنویسید. مانند مثال ذیل:

آموزش کاربردی CSS3	
P{	font-size:20px;
	Font-color:#ff0000;
}	

۲- **محلی:** از شکل محلی میتوانید به دو صورت استفاده نمایید.

a. **صفت Class**

b. **صفت ID**

a. **صفت Class:** شاید از خود پرسیده باشید که ممکن است از یک عنصر در دو جای با صفات (منظور همان خصوصیت یا Property میباشد) متفاوت استفاده کنیم مشکل چیست؟ در این صورت با استفاده از کلاس میتوانید این مشکل را حل کنید. برای درک بهتر از موضوع به مثال دقت کنید:



دستور HTML

```
<p class="left"> This is left class paragraph</p>
<p class="right"> This is right side paragraph </p>
```

دستور CSS

```
.left {color:"red";}
.right{color:"green";}

```

همان طور که مشاهده کردید ما در بخش دستور CSS در قسمت انتخابگر نام هیچ عنصر را قرار ندادیم و بجای آن از نقطه و بعد از نام کلاس آن استفاده کردیم. حال صفحه CSS از این به بعد کلاس تگ را انتخاب میکند نه خود تگ را.

نوت: شما قادر هستید که به چندین تگ کلاس های مشابه (هم نام) بدهید اگر دستور بالا به این صورت استفاده شود.

دستور CSS

```
p.left{color:"red";}

```

فقط بالای تگ P با کلاس مربوط آن تطبیق می گردد.

b. صفت ID:

این صفت هم دقیقاً مانند صفت Class کار میکند اما با تفاوت کوچکی که در قالب دستوری آن برای درک موضوع به مثال ذیل توجه نمایید:

دستور CSS

```
#left{color:"red"}

```

دستور html

```
<p id="left"> This is left side </p>

```



همانطور که مشاهده میکنید بجای نقطه از علامت # قبل از نام استفاده شده و در قسمت پایین نیز از صفت ID استفاده میشود.

توجه داشته باشید مقادیر ID باید یکسان باشند و انعطاف کمتری نسبت به Class دارد یعنی نمی توان برای عناصر خاص مورد استفاده قرار داد.

انواع انتخابگر ها (Types of Selectors)

انتخابگر Class

این نوع انتخابگر عناصری که شامل Class هستند را انتخاب می کند . توضیح مکمل این انتخابگر در مبحث قبلی داده شده است.

سازگار با نسخه های 1,2,3 : CSS

مثال : Header.

انتخابگر id

این نوع انتخابگر عناصری که شامل id هستند را انتخاب می کند . توضیح مکمل این انتخابگر در مبحث قبلی داده شده است.

سازگار با نسخه های 1,2,3 : CSS

مثال : #Subtitle



انتخابگر *

این کاراکتر (*) نشان دهنده ی تمامی عناصر در html است که این علامه تمامی تگ های HTML را انتخاب میکند. یعنی در صورتی که خواسته باشید یک استایل را بالای تمامی صفحه تطبیق نماید باید از این انتخابگر استفاده نمایید.

سازگار با نسخه های 2,3 : Css

مثال: *

انتخابگر element (نام عناصر)

این مثال تمامی عناصر p را انتخاب می کند که البته این انتخابگر بشکل بسیار وسیعی استفاده میشود.

سازگار با نسخه های 1,2,3 : Css

مثال : p

انتخابگر element,element

این مثال تمامی عناصر div و عناصر p را انتخاب میکند . یعنی هر گاه خواسته باشید دو تگ را همزمان انتخاب نمایید از این نوع انتخابگر استفاده نمایید.

سازگار با نسخه های 1,2,3 : Css

مثال : div,p

انتخابگر element element

این مثال تمامی عناصر p که درون عنصر div هستند را انتخاب می کند .

سازگار با نسخه های 1,2,3 : css

مثال : div p



انتخابگر element<element

این نوع انتخابگر تمامی عناصر h2 که عنصر والدش div است را انتخاب می کند .

سازگار با نسخه های 2,3 : CSS

مثال : h2<div

مثال html : در مثال زیر ما دو عنصر h2 داریم اما عنصر h2 دوم چون درون یک عنصر دیگری قرار گرفته و دیگر عنصر فرزند div نیست ، شامل انتخاب انتخابگر CSS نمیشود .

آموزش کاربردی CSS3
<div>
<h2> First P Element </h2>
<h2>Second p Element </h2>
</div>

انتخابگر element+element

این انتخابگر عناصر p را که بلافاصله بعد از عنصر div آمدند را انتخاب می کند .

سازگار با نسخه های 2,3 : CSS

مثال : div+p

انتخابگر [attribute]

توضیح : این مثال تمامی عناصر a که شامل خصوصیت target میشوند را انتخاب میکند .

سازگار با نسخه های 2,3 : CSS

مثال : a[target]



انتخابگر [attribute=value]

این نوع انتخابگر تمامی عناصر a که شامل target هستند و مقدار targetشان برابر با _blank است را انتخاب می کند .

سازگار با نسخه های 2,3 : css

مثال : a[target=_blank]

انتخابگر [attribute~=value]

این انتخابگر تمامی عناصر a که شامل خصوصیت title هستند و محتوای این خصوصیت برابر با flower است را انتخاب می کند .

سازگار با نسخه های 2,3 : css

مثال : a[title~=flower]

انتخابگر [attribute|=value]

این مثال تمامی عناصر a را که شامل خصوصیت title هستند و محتوای این خصوصیت با w3c شروع میشود را انتخاب میکند.

سازگار با نسخه های 2,3 : css

مثال : a[title|=w3c]

انتخابگر link

این نوع انتخابگر تمامی عناصر a که لینک شده اند و تا هنوز آن لینک بازدید نشده را انتخاب میکند .

سازگار با نسخه های 1,2,3 : css

مثال : a:link



انتخابگر visited

این مثال تمامی عناصر a که از لینک شان بازدید شده (بالای لینک شان کلیک شده) را انتخاب می کند .

سازگار با نسخه های 1,2,3 : css

مثال : a:visited

انتخابگر hover

این نوع انتخابگر تمامی عناصر a را زمانی که ماوس روی لینک قرار گرفته را انتخاب میکند. یعنی درست همان زمانی که شما پاینتر را از بالای آن قرار میدهید.

سازگار با نسخه های 1,2,3 : css

مثال : a:hover

انتخابگر active

این مثال تمامی لینک های فعال را انتخاب میکند .

سازگار با نسخه های 1,2,3 : css

مثال : a:active

انتخابگر focus

این مثال تمامی input هایی را انتخاب می کند که اشاره گر ماوس درون آن input باشد و روش متمرکز را بخود اختیار کند .

سازگار با نسخه های 2,3 : css

مثال : input: focus



انتخابگر first-letter

این مثال اولین حرف عناصر p را انتخاب می کند .

سازگار با نسخه های 1,2,3 : css

مثال : p: first-letter

انتخابگر first-line

این مثال اولین خط همه ی عناصر p را انتخاب می کند .

سازگار با نسخه های 1,2,3 : css

مثال : p:first-line

انتخابگر first-child

مثال : p:first-child

این مثال اولین عنصر p در عنصر والد را انتخاب می کند .

سازگار با نسخه های 2,3 : css

مثال html : به مثال زیر دقت کنید ، در این مثال اولین عنصر p در div انتخاب میشود .

آموزش کاربردی CSS3	
<div>	
	<p> First P Element </p>
	<p>Second p Element </p>
</div>	



انتخابگر before

با استفاده از before شما میتوانید محتوایی را قبل از محتوای عنصر مورد نظر قرار بدهید. در این مثال ما محتوایی را قبل از محتوای عنصر p قرار میدهیم.

سازگار با نسخه های : CSS ۲/۳

مثال : p:before

مثال html : در مثال زیر قبل از تمامی عناصر p کلمه Flower را نمایش میدهد :

آموزش کاربردی CSS3
P:before{
Content:"Flower";
}

انتخابگر after

با استفاده از after شما محتوایی را بعد از محتوای عنصر مورد نظر قرار میدهید.

سازگار با نسخه های CSS : ۲/۳

مثال : p:after

مثال html : در مثال زیر بعد از تمامی عناصر p کلمه flower را نمایش میدهد :

آموزش کاربردی CSS3
P:after{
Content:"Flower";
}



انتخابگر element1~element2

این انتخابگر تمامی ul هایی که بعد از عنصر p قرار گرفتند را انتخاب می کند. مهم نیست بلافاصله بعد از عنصر p آمده باشند، کافیست یک عنصر p باشد، هر چقدر عنصر ul بعد از p در کد تطبیق کنیم، مورد انتخاب selector ما قرار میگیرد.

مثال : p~ul

سازگار با CSS۳

انتخابگر [attribute^=value]

این مثال تمامی عناصر a را که src آنها با https شروع میشوند انتخاب میکند.

سازگار با نسخه سی اس اس : ۳

مثال : a[src^="https"]

انتخابگر [attribute\$=value]

این مثال تمامی عناصر a را که src آنها با pdf. خاتمه پیدا کرده باشد را انتخاب میکند.

سازگار با نسخه سی اس اس : ۳

مثال : a[src\$=".pdf"]

انتخابگر [attribute*=value]

این مثال تمامی عناصر a را که src آنها حاوی کلمه ی Kabul باشد را انتخاب می کند.

سازگار با نسخه سی اس اس : ۳

مثال : a[src*="Kabul"]



انتخابگر first-of-type

این مثال اولین عنصر p را در یک عنصر والد انتخاب میکند .

سازگار با نسخه سی اس اس : ۳

مثال : p:first-of-type

مثال html : به مثال زیر توجه کنید :

آموزش کاربردی CSS3	
<div>	
	<p>This is a paragraph</p>
	<p>This is a paragraph</p>
	<p>This is a paragraph</p>
</div>	

انتخابگر last-of-type

این مثال آخرین عنصر p را در یک عنصر والد انتخاب می کند .

سازگار با نسخه سی اس اس : ۳

مثال : p:last-of-type

انتخابگر only-of-type

توضیح : هر تگ <p> که تک فرزند (only child) والد (parent) خودش باشد را انتخاب میکند.

سازگار با نسخه سی اس اس : ۳

مثال : p:only-of-type

مثال html : در مثال زیر، فقط عنصر p اول را انتخاب میکند چون فقط یک عنصر p در عنصر div داریم.



آموزش کاربردی CSS3

```
<div><p>This is a paragraph</p></div>
```

```
<div><p>This is a paragraph</p><p>This is a paragraph</p></div>
```

انتخابگر only-child

این انتخابگر عنصر p که فقط خودش فرزند والدش است را انتخاب میکند .

سازگار با نسخه سی اس اس : ۳

مثال : p:only-child

مثال html : در مثال زیر در عنصر div اول چون فقط یک فرزند دارد و آن هم عنصر p است را انتخاب میکند ، اما در div دوم چون دو فرزند وجود دارد هیچ اشاره ای نمیشود .

آموزش کاربردی CSS3

```
<div><p>This is a paragraph</p></div>
```

```
<div><span>This is a span</span><p>This is a paragraph</p></div>
```

انتخابگر nth-child(n)

این مثال دومین عنصر p در یک عنصر والد را انتخاب میکند .

سازگار با نسخه سی اس اس : ۳

مثال : p:nth-child(2)

انتخابگر nth-of-type(n)

این مثال تمامی عناصر p که دومین عنصر p در تگ والدشان باشند را انتخاب میکند .

سازگار با نسخه سی اس اس : ۳

مثال : p:nth-of-type(2)



انتخابگر last-child

این مثال آخرین عنصر p در عنصر والدش را انتخاب میکند.

سازگار با نسخه سی اس اس : ۳

مثال : p:last-child

انتخابگر empty

توضیح : این مثال عناصر p که خالی هستند و هیچ محتوایی ندارند را انتخاب میکند.

سازگار با نسخه سی اس اس : ۳

مثال : p:empty

تمام صفحات HTML همانند یک درخت هستند!

اگر به یاد داشته باشید در فصل آتی در قسمت ساختار انتخابگر ما تگ های موجود در یک صفحه را به درختی تشبیه کردیم، اما گفتیم در آخر فصل یک مثالی آنرا کار میکنیم. در مثال زیر ساختار درختی را به شکل واضح تر تشریح شده است.

مثال:

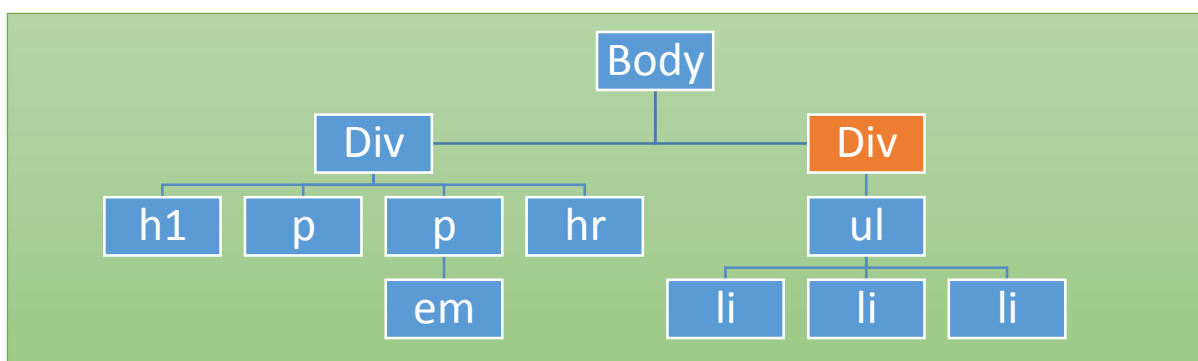
نوت: توجه داشته باشید در ادامه تمامی کد ها به دلیل اختصار از تگ های HTML و Head صرف نظر شده است اما شما میتوانید برای ستندرد بودن صفحه تان از آنها استفاده نمایید.



آموزش کاربردی CSS3

```
<body>
  <div id="Content">
    <h1>This is Site Heading</h1>
    <p>This is the first paragraph</p>
    <p>This is the second <em>site</em> Paragraph </p>
    <hr>
  </div>
  <div id="nav">
    <ul>
      <li>First item</li>
      <li>Second item</li>
      <li>Third item </li>
    </ul>
  </div>
</body>
```

حالا کد بالا را به صورت درختی به شکل ذیل میتوان تشریح کرد.



در درخت فوق تگ Div دوم را که با رنگ از دیگر مجزا شده ، میتوان بگونه زیر بیان کرد:

فرزندی از عنصر <Body>

پدر عنصر

جد عناصر و

برادر عنصر <div> دیگر چون پدر های آنان یکسان است (<body>).

برای درک بیشتر مفاهیم بعدی باید با موارد بالای آشنایی مکمل داشته باشید.



فصل چهارم

مباحث فصل چهارم

✓ CSS در Properties

✓ مبحث اول: Background

✓ مبحث دوم: Text

✓ مبحث سوم: Fonts

✓ مبحث چهارم: Links

✓ مبحث پنجم: Lists

✓ مبحث ششم: Tables

✓ مبحث هفتم: Box Model

✓ مبحث هشتم: Border

✓ مبحث نهم: ساخت اشکال با Border

✓ مبحث دهم: outline

✓ مبحث یازدهم: position

✓ مبحث دوازدهم: تنظیم طول و عرض

✓ مبحث سیزدهم: direction



✓ مبحث چهاردهم: Clear and float

✓ مبحث پانزدهم: Box-shadow

✓ مبحث شانزدهم: Box-sizing

✓ مبحث هفدهم: Gradient

✓ مبحث هجدهم: Transform

✓ مبحث نوزدهم: Transition

✓ مبحث بیستم: Cursor

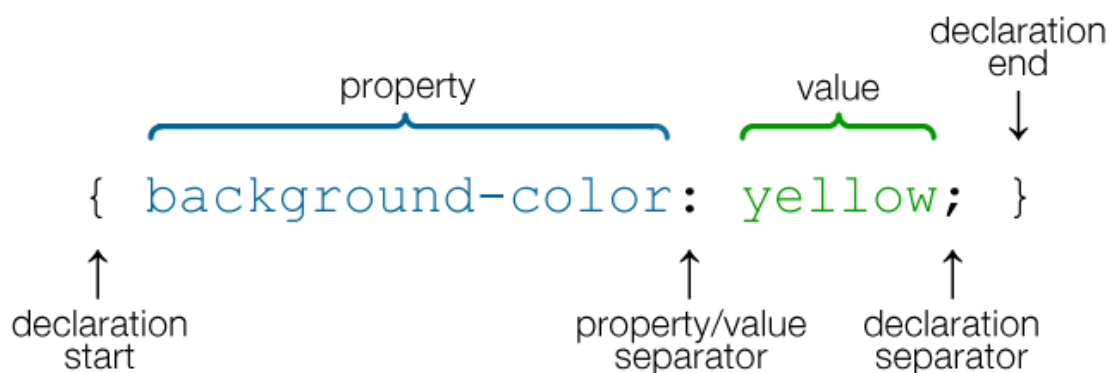


CSS Properties

در این فصل کتاب تمامی Property های موجود در CSS2 و CSS3 را در چندین مبحث مورد بحث قرار میدهیم که البته در اینجا شما برای درست نمودن ساختار سایت باید آشنایی به زبان HTML5 داشته باشید. در ضمن ما در این جا از نوشتن دستورات HTML به دلیل پیچیدگی خودداری نموده ایم، اما شما باید بخاطر انجام کار صفحه HTML درست نمایید.

Property در اصل چیست؟

به زبان ساده Property عبارت از مجموعه یک و یا چند دستور است که بعد از انتخابگر نوشته میشود. تصویر زیر این مساله را به صورت واضح بیان میکند:



خوب میرویم و مباحث بزرگ Properties را آغاز میکنیم!



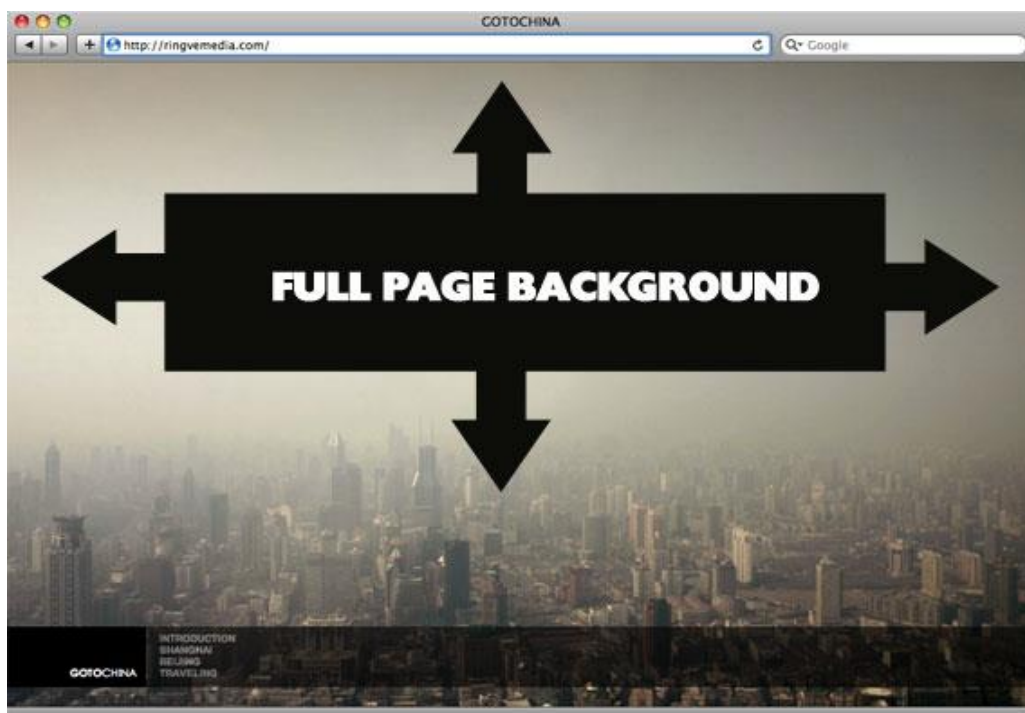
مبحث اول: Background

در مبحث اول فصل چهارم راجع به Background صفحات صحبت میکنیم.

خوب Background چیست؟

در کل وقتی شما تصویری و یا رنگی را در پس زمینه سایت تان قرار میدهید Background گفته میشود.

به تصویر زیر دقت کنید که به صورت یک کاغذ نمایش داده شده تصویر Background صفحهء ویب به شمار میاید.



نکته: این خاصیت در تمامی مرورگر ها قابل شناخت است.

خاصیت Background دارای خاصیت های ترکیبی ذیل میباشد که با خاصیت اصلی یکجا شده امکانات بیشتر را فراهم مینماید.



Background-color: این خاصیت رنگ پس زمینه را تعیین میکند یعنی میتوانید بجای تصویر زمینه صفحه تانرا رنگ انتخاب نمایید. برای انتخاب نوع به فصل دوم مراجعه نمایید.

آموزش کاربردی CSS3

Background-color:red;

Background-position: در صورتی که شما بخواهید یک تصویر و یا یک آیکن برای قسمتی از پس زمینه در نظر بگیرید از این خاصیت استفاده می کنید ، با این خاصیت مختصات قرار گرفتن آیکن و یا تصویر را مشخص می کنید. مثال:

Background-position: 120px 150px در این مثال بصورت پیکسلی مختصات میدهید.

Background-position: left شما نیز میتوانید تعیین کنید تصویر پس زمینه در سمت چپ ، راست ، بالا یا پایین قرار بگیرد.

آموزش کاربردی CSS3

Background-position: 120px 150px;

و یا

Background-position: left;

Background-size: این خاصیت تازه وارد دنیایی وب گردیده یعنی تنها در CSS3 قابل نمایش است و البته با استفاده از این خاصیت میتوانید اندازه تصویر تانرا تعیین نمایید.

مثال:

Background-size: 80px 60px بصورت پیکسلی میتوانید عرض و ارتفاع پس زمینه را مشخص کنید.

Background-size: 100% 100% بصورت درصدی میتوانید عرض و ارتفاع پس زمینه را مشخص کنید.



Background-size: cover عرض تصویر را با توجه به عرض صفحه بزرگ میکند، ارتفاع را نیز با توجه به عرض بزرگ می کند.

توجه: به دلیل اینکه ممکن است ارتفاع صفحه کمتر از عرض و یا بر عکس آن باشد بخشی از تصویر حذف شود و خارج از صفحه قرار بگیرد و نمایش داده نشود

Background-size: contain ارتفاع تصویر را با توجه به ارتفاع صفحه بزرگ می کند و عرض تصویر را نیز با توجه به ارتفاع تغییر میدهد و نهایتاً تمام تصویر نمایش داده میشود.

توجه: به دلیل اینکه ممکن است اندازه صفحه بزرگ باشد، تصویر در تمام صفحه قابل نمایش نباشد.

Background-size: initial تصویر پس زمینه بصورت عادی و در اندازه ی واقعی تصویر، نمایش داده میشود.

نکته: به یاد داشته باشید این خاصیت ها در مرور گرهای IE9+, Firefox, Opera, Chrome, Safari پشتیبانی میشود.

Background-repeat: با این خاصیت میتوانید تعیین کنید که تصویر پس زمینه ای که برای Background در نظر گرفتید، تکرار شدنی باشد یا خیر به مثال های زیر توجه نمایید!

مثال :

Background-repeat: no-repeat تصویر تکرار نشود (به حالت عادی نشان داده میشود).

نحوه نوشتن:

آموزش کاربردی CSS3

```
Background-repeat:no-repeat;
```

Background-repeat: repeat تصویر تکرار شود (اگر تصویر از حد معمول کوچکتر باشد تکرار میشود و تمام صفحه را می پوشاند).

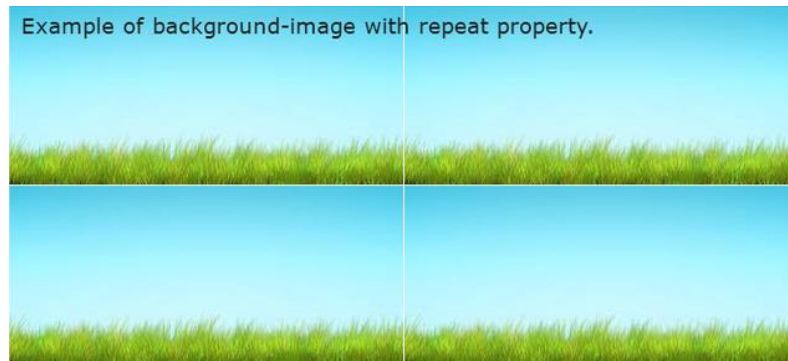
نحوه نوشتن:



آموزش کاربردی CSS3

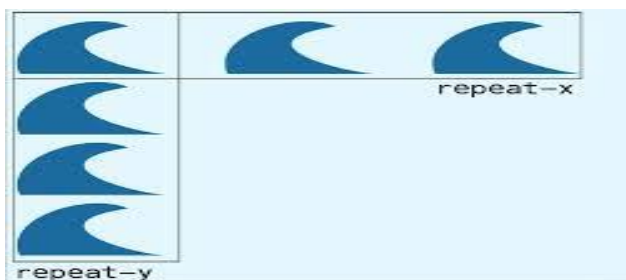
Background-repeat:repeat;

نتیجه:



Background-repeat: repeat-x تصویر را در عرض یا به شکل افقی تکرار میکند.

Background-repeat: repeat-y تصویر را در طول یا به شکل عمودی تکرار میکند.



Background-origin: این خاصیت تازه وارد دنیای وب گردیده یعنی تنها در CSS3 قابل نمایش است و البته این خاصیت تصویر پس زمینه را با توجه به مقداری که برایش تعیین می کنید در صفحه قرار میدهد. مثال :

Background-origin: border-box نسبت به border و حاشیه ی صفحه یا باکس تصویر را قرار میدهد.

Background-origin: content-box نسبت به محتوای صفحه تصویر را تطبیق می کند.

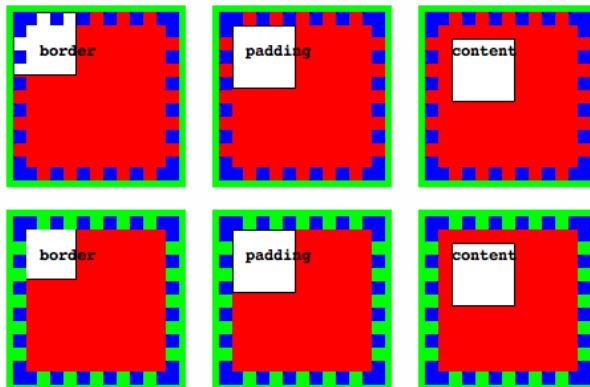
Background-origin: padding-box نسبت به padding صفحه تصویر را قرار میدهد.



Background-origin: initial بصورت عادی تصویر را قرار میدهد، با توجه به تنظیمات پیشفرض.

یادداشت: در مورد Padding بصورت مفصل در فصل اتی بحث خواهیم کرد.

نتیجه:



Background-image: میتوانید با این خاصیت یک تصویر برای پس زمینه انتخاب کنید.
مثال:

آموزش کاربردی CSS3

```
Background-image:url(sample.png);
```

شما میتوانید خاصیت های بالا را به صورت خلاصه نیز بنویسید. که البته این نوع نوشتن خاصیت ها به دلیل جلوگیری از ضایع شدن وقت بیشتر استفاده میشود

تمامی خاصیت های Background را توضیح دادیم ، در صورتی که بخواهید بصورت خلاصه نویسی از این خاصیت ها استفاده کنید ، میتوانید از دستور زیر پیروی کنید : (پیشنهاد ما برای استانداردسازی کدهای تان خلاصه نویسی است)

آموزش کاربردی CSS3

```
Background:color position size repeat clip attachment image initial | inherit ;
```



Multiple Background

حال میخواهیم با ادغام دستورات قبلی چندین تصویر را همزمان بالای همدیگر قرار دهیم و در پس زمینه صفحه خود قرار دهیم شما میتوانید چند تصویر را روی هم قرار بدهید و چند background داشته باشید. برای این کار به مثال زیر توجه نمایید:

مثال: در اینجا ما میخواهیم دو تصویر را روی همدیگر قرار دهیم قسمی که تصویر دوم در زیر تصویر اول قرار بگیرد.

آموزش کاربردی CSS3
#sample{
Background-image:url(sheep.png), url(gress.png);
Background-position:center bottom,left top;
Background-repeat:no-repeat, repeat;
}

توجه داشته باشید در استفاده Background-position و Background-repeat هم باید ترتیب تصاویر که در Background-image تعریف نموده اید مد نظر بگیرید.

حالا نتیجه دستور بالا به شکل زیر است.



توجه: در استفاده Multiple background ها شما باید از خلاصه نویسی استفاده ننماید و باید هر

خاصیت را به صورت مجزا بنویسید!

مرورگرهای که از خاصیت Background پشتیبانی میکنند قرار زیر است:



مبحث دوم: Text

در این مبحث در مورد text یا همان تنظیمات متون صحبت میکنیم. تنظیمات متون شامل اندازه متن، ارتفاع متن، موقعیت متن و امثال اینها میباشد که در ادامه این تمامی این تنظیمات را مطالعه میکنیم.

Text-align

با استفاده از این خاصیت شما میتوانید موقعیت متن تگ در را صفحه مشخص نمایید و این خاصیت از جمله پر استفاده ترین خاصیت در قسمت متن میباشد.

خاصیت text-align شامل مقادیر زیر میباشد:

- **Right:** با استفاده از این مقدار متن شما به طرف راست انتقال میکند.
- **Left:** با استفاده از این مقدار متن شما به طرف چپ انتقال میکند.
- **Center:** با استفاده از این مقدار متن شما در وسط صفحه قرار میگیرد.

قاعده دستوری خاصیت بالا به شکل زیر میباشد:

آموزش کاربردی CSS3

Text-align: right | left | center;

Text-decoration

با استفاده از این خاصیت میتوانید افکتی را بالای متن تگ مربوط اعمال نمایید. این خاصیت شامل مقادیر زیر میباشد:

- **Blink:** این مقدار موجب میشود تا متن تان حالت چشمک زدن داشته باشد در ضمن بیاد داشته باشید که این مقدار در اکثریت مرورگر های امروزی قابل اجر نمیشود.
- **Line-through:** با استفاده از این مقدار خطی بالای متن تان ایجاد میگردد.
- **None:** این مقدار حالت پیشفرض متن میباشد که برای متن هیچ افکتی تعریف نمیکند.

- **Overline:** این مقدار خطی بالای متن تگ ایجاد میکند.

- **Underline:** این مقدار برای متن شما زیر خطی ایجاد میکند.

قاعده دستور این خاصیت همانند خاصیت قبلی میباشد.



Text-indent

با استفاده از این خاصیت شما میتوانید میزان تورفتگی اولین خط یک متن را مشخص میکند. این تورفتگی از سمت چپ یا راست بستگی به جهت صفحه ویب دارد که این مورد معمولاً با استفاده از خاصیت Dir تگ HTML مشخص میشود. این خاصیت دارای مقداری خاص نمیشود فقط کافیست به آن از واحد های اندازه گیری قیمت دهید.

Text-transform

با استفاده از این خاصیت میتوانید نحوه نمایش حروف (کوچک و یا بزرگ) را کنترل نمایید. این خاصیت شامل مقادیر زیر میباشد:

- **Capitalize**: در صورت استفاده از این مقدار، اولین حرف تمامی کلمات متن تان به حروف بزرگ تبدیل میگردد و بقیه حروف به حالت عادی نمایش داده میشود.
- **Lowercase**: در صورت استفاده از این مقدار، تمامی حروف تان به حروف کوچک تبدیل میگردد.
- **Uppercase**: در صورت استفاده از این مقدار، تمامی حروف تان به حروف بزرگ تبدیل میگردد.
- **None**: این مقدار بالای متن شما هیچ تغییری ایجاد نمیکند.

Text-shadow

این خاصیت تازه در نسخه CSS3 ارایه شده است. که توسط این خاصیت شما میتوانید افکت سایه ای مختلف بالای متن تان اعمال کرد.

این خاصیت دارای دو مقدار اجباری میباشد که در آن شما تعیین میکند سایه متن شما به کدام طرف کشیده شود و دارای دو مقدار اختیاری میباشد که در آن رنگ و میزان خیره گی را مشخص میکند. برای تعیین کردن سمت سایه میتوانید از دو مقدار اولی استفاده نمایید، در صورت استفاده از قیمت مثبت باشد سایه بطرف راست سوق داده میشود و در صورت که قیمت منفی باشد سایه به طرف چپ سوق داده میشود.



مقدار خیره گی سایه را میتوانید بعد از تعیین جهت سایه تعیین نمایید در صورت بالا رفتن قیمت این مقدار اندازه خیره گی آن بیشتر و در صورت ندادن قیمت سایه بدون خیره گی نمایش داده میشود.

در پایان شما میتوانید رنگ سایه تانرا مشخص نمایید و در صورت مشخص نکردن رنگ سایه این مقدار به شکل پیشفرض از خاصیت Color رنگ را به ارث میبرد.

مثال:

آموزش کاربردی CSS3	
#h1{	
	text-shadow: 5px 5px 5px #FF0000;}

نتیجه:

Text shadow effect!

Vertical-align

با استفاده از این خاصیت شما میتوانید میزان عمودی محتویات داخل تگ ها یا میزان عمودی خانه های جدول را مشخص نمایید.

این خاصیت دارای مقادیر زیر است:

Baseline: این مقدار عبارت از خط فرضی میباشد که حروف بروی آن نوشته میشود.

- **Bottom:** این مقدار عناصر تانرا در قسمت پایین قرار میدهد.
- **Sub:** این مقدار تگ فرزند را کمی پایین تر از امتداد متن تگ پدر قرار میدهد.
- **Super:** این مقدار تگ فرزند را کمی بالاتر از امتداد متن تگ پدر قرار میدهد.
- **Text-bottom:** این مقدار متون تگ پدر در امتداد زیرین تگ فرزند قرار میدهد.
- **Text-top:** این مقدار متون تگ پدر در امتداد بالای تگ فرزند قرار میدهد.
- **Top:** این مقدار عناصر تانرا در قسمت پایین قرار میدهد.



مثال: در این جا ما یک تیبِل داریم و می‌خواهیم تصویر که در داخل سلول است آنرا در قسمت پایین سلول قرار دهیم.

در اینجا ما از تصویر استفاده نمودیم اما شما میتوانید از متن استفاده نمایید.

آموزش کاربردی CSS3	
Td img {	
	Vertical-align: bottom;
}	

With-space

با استفاده از خاصیت With-space شما میتوانید تعیین کنید وقتی متنی را در باکسی نوشته میکنید و آن باکس از اندازه خطوط متن کوچکتر باشد، متن شما به صورت اتوماتیک قطع شود و یا متن شما ادامه داشته باشد.

خاصیت With-space دارای مقادیر زیر است:

- **Nowarp**: در صورت استفاده از این مقدار متن شما در باکس نمیشکند و ادامه پیدا میکند.
- **Pre**: در صورت استفاده از این مقدار مرورگر تشخیص میدهد که آیا در کد HTML از دستورات Line break استفاده کردید یا خیر و در صورت استفاده از همان دستورات به منظور قطع نمودن خط استفاده مینماید.
- **Pre-line**: با استفاده از مقدار در صورت شما از Line break استفاده نموده اید آن اعمال میشود و در ضمن بیاد داشته باشید که در حالت تمام space های شما نیز حساب میشود.
- **Pre-wrap**: این مقدار همانند مقدار قبلی است اما فقط با این فرق در صورت که متن شما از اندازه باکس بزرگتر شود در آن صورت متن به صورت اتوماتیک خط قطع میگردد.



قاعده دستوری این خاصیت به شکل زیر میباشد:

آموزش کاربردی CSS3	
P{	
With-space: nowrap pre pre-line pre-wrap ;	
}	

مرورگرهای که از این خاصیت پشتیبانی میکنند قرار زیر است:



مبحث سوم: Fonts

در اصل فونت ها در وب چیست؟

منظور از فونت نحوه نوشتن متون و به اصطلاح عبارت از نوع شکل متن است که شما قادرید برای آن رنگ، اندازه، نوع قلم و... تعیین کنید. همچنین میتوانید بر فونت های پیشفرض وب، فونت های دلخواه تان را حتی به صفحات وب فارسی اضافه کنید و بالای متون خود بکار ببرید. بیاد داشته باشید هر مرورگر یک استایل خاصی برای فونت ها دارا میباشد یعنی اگر شما برای متون حتی فونت را انتخاب نکرده باشید بازهم متن نمایش داده میشود اما به دلیل زیبای بیشتر در صورتی که خواسته باشید میتوانید آنرا تغییر دهید.

ساختار اصلی Font ها در CSS

در حالت استاندارد از فونت ها به شکل خلاصه نویسی استفاده میشود:

آموزش کاربردی CSS3
Font: font-style font-variant font-weight font-size line-height font-family
Font: italic normal bold 12px/22px Tahoma

فونت ها خاصیت های زیر را دارا میباشد:

Font-style

خاصیت font-style بمنظور استایل دادن به متن استفاده میشود و دارای مقادیر زیر است.

- **Normal**: در صورت نوشتن این مقدار استایل پیشفرض مرورگر را برای فونت در نظر میگیرد که درج این مقدار هیچ تاثیری بالای متن ندارد).
- **Italic**: در صورتی که این مقدار را قرار بدهید، استایل فونت به شکل کج تبدیل میگردد.
- **Oblique**: این مقدار درست همانند مقدار Italic میباشد و فرقی با Italic ندارد.
- **Initial**: در صورتی که این مقدار را قرار بدهید استایل پیشفرض مرورگر تان تطبیق میشود پس تطبیق آن در موارد بسیار اندکی استفاده دارد.
- **Inherit**: در صورتی که برای عنصر فرزند این مقدار را تطبیق کنید، استایل فونت عنصر والد را به خودش اختصاص میدهد.



Font-variant

این خاصیت خاصاً برای متون لاتین استفاده میشود و مقادیر زیر را دارا میباشد

- **Normal**: در صورت نوشتن این مقدار استایل پیشفرض مرورگر را برای فونت در نظر میگیرد که درج این مقدار هیچ تاثیری بالای متن ندارد).
- **Small-caps**: در صورتی که این مقدار را قرار بدهید تمامی حروف را به حروف بزرگ انگلیسی تبدیل نموده و نمایش میدهد.
- **Initial**: در صورتی که این مقدار را قرار بدهید استایل پیشفرض مرورگر تان تطبیق میشود پس تطبیق آن در موارد بسیار اندکی استفاده دارد.
- **Inherit**: در صورتی که برای عنصر فرزند این مقدار را تطبیق کنید ، Font-Variant فونت عنصر والد را به خودش اختصاص میدهد.

Font-weight

این خاصیت در کل ضخامت متون را مشخص میکند.

- **Normal**: در صورت نوشتن این مقدار استایل پیشفرض مرورگر را برای فونت در نظر میگیرد که درج این مقدار هیچ تاثیری بالای متن ندارد).
- **Bold**: در صورت نوشتن این مقدار متن با ضخامت بیشتر نمایش داده میشود.
- **Bolder**: این مقدار همانند مقدار bold میباشد.
- **Lighter**: در صورت نوشتن این مقدار مرورگر متن را با ضخامت پیشفرض نشان میدهد.
- **Initial**: در صورتی که این مقدار را قرار بدهید استایل پیشفرض مرورگر تان تطبیق میشود پس تطبیق آن در موارد بسیار اندکی استفاده دارد.
- **Inherit**: در صورتی که برای عنصر فرزند این مقدار را تطبیق کنید ، Font-weight فونت عنصر والد را به خودش اختصاص میدهد.



Font-size

این خاصیت خاصاً برای تعیین نمودن اندازه متن استفاده میشود و بصورت عموم از PX و em استفاده میشود.

- **Xx-small**: سه برابر کوچکتر از اندازه پیشفرض فونت را نمایش میدهد.
- **X-small**: دو برابر کوچکتر از اندازه پیشفرض فونت را نمایش میدهد.
- **Small**: یک برابر کوچکتر از اندازه پیشفرض فونت را نمایش میدهد.
- **Medium**: بصورت میدوم و نصف اندازه پیشفرض فونت را نمایش میدهد.
- **Large**: یک برابر بزرگ تر از اندازه پیشفرض فونت را نمایش میدهد.
- **X-large**: دو برابر بزرگ تر از اندازه پیشفرض فونت را نمایش میدهد.
- **Xx-large**: سه برابر بزرگ تر از اندازه پیشفرض فونت را نمایش میدهد.
- **Smaller**: با اندازه مابین x-small و xx-small فونت را نمایش میدهد.
- **Larger**: با اندازه مابین x-large و xx-large فونت را نمایش میدهد.
- **PX**: با واحد پیکسل فونت را نمایش میدهد. در صفحات وب در صورتی که از Tahoma استفاده کنید فونت استاندارد ۱۲ px است.
- **%**: با واحد درصد فونت را نمایش میدهد.
- **Initial**: در صورتی که این مقدار را قرار بدهید استایل پیشفرض مرورگر تان تطبیق میشود پس تطبیق آن در موارد بسیار اندکی استفاده دارد.
- **Inherit**: در صورتی که برای عنصر فرزند این مقدار را تطبیق کنید ، Font-size فونت عنصر والد را به خودش اختصاص میدهد.

Line-height

توسط این خاصیت شما میتوانید فاصله بین خطوط را تعیین کنید که شامل مقادیر زیر میشود:

- **Length**: این مقدار با واحد های px,pt,cm,em استفاده میشود و میتوانید فاصله خطوط را تعیین کنید. عموماً از واحد های Px و em استفاده میگردد.
- **Initial**: در صورتی که این مقدار را قرار بدهید استایل پیشفرض مرورگر تان تطبیق میشود پس تطبیق آن در موارد بسیار اندکی استفاده دارد.



- **Inherit**: در صورتی که برای عنصر فرزند این مقدار را تطبیق کنید ، line-height ، فونت عنصر والد را به خودش اختصاص میدهد.

نکته: در صورتی که لازم باشد شما میتوانید از این خاصیت ها بصورت جداگانه استفاده کنید ، به عنوان مثال یک صفحه وب دارید که از پیش استایل font: 16px/25px Tahoma را برای متون تطبیق کرده ، و شما میخواهید تنها اندازه فونت را تغییر بدهید ، در این صورت با خاصیت font-size این کار را انجام میدهید.

افزودن فونت های دلخواه به صفحات وب

اکثر اوقات پیش میاید که شما میخواهید فونت پیشفرض سایت تان را تغییر دهید و یک فونت دلخواه تان را به صفحه وب تان اضافه کنید.

به عنوان مثال در اینجا از فونت Byekan برای متون صفحات اضافه میکنیم. برای این منظور باید ابتدا فونت های وب Byekan را دانلود و به فایل CSS تان اضافه کنید و سپس بالای متون تطبیق نمایید.

نوت: فونت های که در ویندوز استفاده میشود نسبت به فونت های که در وب استفاده میشود تفاوت دارد یعنی برای صفحات وب فونت های مجزا وجود دارد و برای دانلود میتوانید به اینترنت مراجعه کنید اما میتوانید برای دانلود به لینک ذیل مراجعه کنید:

www.p30download.com/fa/entry/50551

نحوه افزودن فونت های دلخواه در CSS

پس از اینکه فایل را دانلود کردید ، در نخست باید فونت را انتخاب کنید و از فولدری fonts کپی کنید و در فولدری خودتان (سایت تان) paste کنید . ما Byekan را برای مثال در نظر میگیریم . همانند کد زیر ، فونت مورد نظرتان را در ابتدای فایل CSS فراخوانی کنید:



آموزش کاربردی CSS3

```
@font-face{
Font-family:"byekan";
Src:url("font/byekan.eot?#")format("eot"),
url("font/byekan.woff")format("woff")
url("font/byekan")format("truetype")
}
```

توسط کد زیر میتوانید فونت بالا را بالای متن تان تطبیق نمایید:

آموزش کاربردی CSS3

```
Font:13px/22px byekan, Tahoma;
```

توجه: برای فونت های فارسی نمیتوانید هر اندازه را انتخاب نمایید، به عنوان مثال سایز های استاندارد برای Byekan اندازه های 16px , 16px , 13px... میباشد.

یادداشت: در کد بالا نام فونت Tahoma را نیز نوشته کردیم و دلیل آن این است که در صورتی که متن لاتین نوشته شده باشد متن لاتین را به آن شکل نمایش بدهد.

تعیین رنگ برای فونت ها

برای تعیین رنگ مورد نظر برای فونت ها از خاصیت Color استفاده می کنیم . به عنوان مثال اگر بخواهیم از رنگ سیاه برای فونت استفاده کنیم میتوانیم به اشکال زیر این کارا انجام بدهیم:

آموزش کاربردی CSS3

```
Color:black
Color:#000
Color:rgb(0,0,)
Color:rgba(0,0,0,1)
```

به همین منوال شما میتواند از دیگر رنگها نیز استفاده نمایید و برای انتخاب رنگ میتوانید به فصل دوم مراجعه نمایید.



مبحث چهارم Links

لینک ها به عناصر گفته میشوند که بمنظور ارتباط دادن عناصر سایت با صفحات مرتبط با سایت مانند: تصاویر ، ویدیو، دیگر صفحات وب و ... امثال اینها استفاده میشود.

مسلماً شما نحوه ارتباط دادن که مربوط به HTML میشود را آموخته اید، از این لحاظ ما از نحوه ارتباط دادن صرف نظر میکنیم و در اینجا تنها موارد که مربوط به دیزاین لینک ها میشود را مورد بحث قرار میدهیم.

در مجموع زمانی که شما یک لینکی را در صفحه تعریف می کنید ، دارای ۴ حالت زیر میشود:

- ۱- **حالت عادی**: این حالت عبارت از همان حالت عادی لینک است که بصورت پیش فرض به رنگ آبی نمایش داده میشود. اما شما قادرید این دیزاین را تغییر دهید.
- که البته به دو صورت میتوانید این را تغییر دهید:

آموزش کاربردی CSS3
A:link {
Color:#ff0000;
}

و یا

آموزش کاربردی CSS3
A {
Color:#ff0000;
}

حالا بعد از استفاده دستور بالا رنگ لینک انتخاب شده به رنگ سرخ تبدیل میشود.

- ۲- **حالت hover**: این حالت عبارت از همان حالت میباشد که شما نشانگر (Pointer) را از بالای لینک عبور میدهد. که البته این حالت به صورت پیشفرض غیر فعال میباشد اما میتوانید مانند دستور زیر به آن دیزاین تعریف نمایید.

آموزش کاربردی CSS3
A:hover {
Color:#8B4513;
}



۳- **حالت Active:** این حالت عبارت از حالت می باشد که شما توسط Pointer بالای لینک مورد نظر کلیک را محکم میگیرید و هنگام رها کردن کلیک دوباره به حالت اول بر میگردد.

آموزش کاربردی CSS3
A:active {
Color:#00FFFF;
}

این حالت زمانی بیشترین کاربرد را دارد که شما قصد رسانسیو کردن قالب سایت را دارید. در دستگاه هایی مثل موبایل و تبلت، حالت hover را نمیتوانید پیاده کنید مگر اینکه انگشت را از یک نقطه تا یک نقطه ی دیگر بکشید و از روی لینک بگذرانید که گاهی اوقات حالت hover روی لینک اتفاق می افتد اما این کار اصلاً اصولی نیست ، ولی active را میتوانید به راحتی اعمال کنید. چون در این دستگاه ها کاربر با قرار دادن انگشت روی لینک آن را انتخاب میکند و در آن حالت active اتفاق می افتد. زمانی که شما یک استایل به hover لینک بدهید به active هم بدهید که در مانیتور hover اجرا شود و در موبایل و تبلت active اجرا شود.

۴- **حالت Visited:** این حالت عبارت از حالتی است که شما از لینک قبلاً بازدید کرده اید و توسط این حالت میتوانید تعیین که بعد از بازدید لینک رنگ آن به چی رنگی تبدیل شود.

آموزش کاربردی CSS3
A:visited {
Color:ooffoo;
}

:link

Send Button

:hover

Send Button

:active

Send Button



خط زیرلینک ها

به صورت پیشفرض لینک ها دارای زیر خطی میباشد اما شما میتوانید این شکل را به دیگر شکل ها و یا کاملاً از بین ببرید.

none: در صورت استفاده از این مقدار خط زیرلینک حذف میشود.

underline: در صورت استفاده از این مقدار خط زیرلینک نمایش داده میشود.

overline: در صورت استفاده از این مقدار خط بالای متن ارتباط داده شده نمایش داده میشود.

line-through: این مقدار خط را در وسط متن نمایش میدهد.

آموزش کاربردی CSS3	
A{	
	Text-decoration: none;
}	

در ضمن بیاد داشته باشید که این خاصیت را میتوانید بالای عناصری دیگر مانند h1,h2,P... استفاده نماید.



مبحث پنجم: لیست ها List

در این مبحث ما در مورد لیست ها در CSS صحبت میکنیم. البته در اینجا ما تنها نحوه نمایش لیست ها را مورد بحث قرار میدهیم چون ساختن لیست ها در HTML امکان پذیر میباشد.

لیست های به صورت کل دو نوع اند:

۱. لیست های مرتب

۲. لیست های نامرتب

۱- لیست های مرتب: لیست های مرتب به لیست های گفته میشود که شروع لیست با مقادیر آغاز میشوند که در صورت تغییر نمودن جای خطوط لیست، نامرتب میشود و طبق معمول این نوع لیست ها با اعداد، الفبا و امثال اینها آغاز میشود.

۲- لیست های نامرتب: لیست های نامرتب به لیست های گفته میشود که در صورت تغییر نمودن جای خطوط لیست، تغییری در لیست نمیآید و معمولاً این نوع لیست ها با علامت ها آغاز میشوند.

لیست ها در CSS دارای خاصیت زیر میباشد:

List-style-type

این نوع لیست ها مربوط به لیست های مرتب میباشد البته این خاصیت دارای ۱۹ نوع میباشد اما تمام اینها قابل استفاده نمیشود از این لحاظ در این جا پر استفاده ترین آنها را مورد بحث قرار میدهیم. این خاصیت دارای مقادیر زیر میباشد:

• Decimal: این مقدار کنار هر سطر از لیست اعداد را نمایش میدهد.

۱. آموزش کاربردی CSS3

۲. انواع لیست ها

۳. لیست های مرتب



- **Decimal-leading-zero:** این مقدار کنار هر سطر از لیست اعداد را با صفر نمایش میدهد.
 - ۰۱. آموزش کاربردی CSS3
 - ۰۲. آموزش لیست ها
 - ۰۳. لیست های مرتب
- **Lower-alpha:** این مقدار کنار هر سطر از لیست حروف کوچک انگلیسی را نمایش میدهد.
 - a. آموزش کاربردی CSS3
 - b. آموزش لیست ها
 - c. لیست های مرتب
- **Lower-greek:** این مقدار کنار هر سطر از لیست حروف کوچک الفابت ریاضی را نمایش میدهد.
 - α. آموزش کاربردی CSS3
 - β. آموزش لیست ها
 - γ. لیست های مرتب
- **Lower-latin:** این مقدار کنار هر سطر از لیست حروف کوچک لاتین را نمایش میدهد.
 - a. آموزش کاربردی CSS3
 - b. آموزش لیست ها
 - c. لیست های مرتب
- **Lower-roman:** این مقدار کنار هر سطر از لیست حروف کوچک رومی را نمایش میدهد.
 - i. آموزش کاربردی CSS3
 - ii. آموزش لیست ها
 - iii. لیست های مرتب
- **Upper-roman:** این مقدار کنار هر سطر از لیست حروف بزرگ رومی را نمایش میدهد.
 - I. آموزش کاربردی CSS3
 - II. آموزش لیست ها
 - III. لیست های مرتب



- **Upper-latin**: این مقدار کنار هر سطر از لیست حروف بزرگ لاتین را نمایش میدهد.

A. آموزش کاربردی CSS3

B. آموزش لیست ها

C. آموزش لیست های مرتب

- **Upper-alpha**: این مقدار کنار هر سطر از لیست حروف بزرگ انگلیسی را نمایش میدهد.

A. آموزش کاربردی CSS3

B. آموزش لیست ها

C. آموزش لیست های مرتب

قاعده دستوری مقادیر بالا به شکل زیر میباشد:

آموزش کاربردی CSS3
Li{
List-style-type: Decimal lower-greek upper-greek;
}

مقادیر که در بالا ذکر گردید مقادیر بودند که در لیست های مرتب استفاده میشد اما حالا مقادیر لیست های نامرتب را مورد بحث قرار میدهیم.

- **Circle**: این مقدار کنار هر سطر از لیست دایره میان خالی را نمایش میدهد.

○ آموزش کاربردی CSS3

○ آموزش لیست ها

○ آموزش لیست های نامرتب

- **Square**: این مقدار کنار هر سطر از لیست مربع میان پر را نمایش میدهد.

▪ آموزش کاربردی CSS3

▪ آموزش لیست ها

▪ آموزش لیست های نامرتب

- **Disk**: این مقدار کنار هر سطر از لیست دایره میان پر را نمایش میدهد.



- آموزش کاربردی CSS3
- آموزش لیست ها
- آموزش لیست های نامرتب

قاعده دستوری مقادیر لیست های نامرتب به شکل زیر میباشد:

آموزش کاربردی CSS3
Li{
List-style-type: circle square disk;
}

List-style-position

با استفاده از این خاصیت شما میتوانید تعیین نمایید علامه که در آغاز سطر نمایش داده میشود به درون تگ باشید و یا خارج از تگ.

این خاصیت دارای مقادیر زیر میباشد:

- Inside: این مقدار علامه یا همان Bullet را داخل li قرار میدهد. البته این حالت پیشفرض خاصیت میباشد.
- Outside: این مقدار علامه یا همان Bullet را خارج از li قرار میدهد.

برای درک بهتر به تصویر زیر دقت نمایید.

<ul style="list-style-type: none"> • One thing was certain, that the white kitten • had had nothing to do with it • it was the black kitten's fault entirely <p>Inside</p>	<ul style="list-style-type: none"> • One thing was certain, that the white kitten • had had nothing to do with it • it was the black kitten's fault entirely <p>Outside</p>
---	--



List-style-image

با استفاده از خاصیت List-style-image شما میتوانید بجای علامه سطر ها، از تصاویر دلخواه تان استفاده نمایید.

قاعده دستوری این خاصیت به شکل زیر میباشد:

آموزش کاربردی CSS3
List-style-image:url('bullet.gif');

خوب بخاطر اینکه نحوه ساخت لیست دوباره به یاد بیاید میرویم و یک مثال کاربردی را انجام میدهیم.

دستور HTML:

آموزش کاربردی CSS3

آموزش کاربردی CSS3
آموزش لیست ها
آموزش لیست های نا مرتب

دستور CSS:

آموزش کاربردی CSS3
UI{
Margin:0; padding:0;
Font:12px/30px Tahoma; text-align:right;
List-style: url(cd.gif) inside;
{
UI li{
Margin:0; padding:0;}

حالا نتیجه دستورات بالا به شکل زیر میباشد:

- آموزش کاربردی CSS3
- آموزش لیست ها
- آموزش لیست های نا مرتب



مبحث ششم: Tables

در این مبحث در مورد جداول صحبت میکنیم. که البته خاصیت های این مبحث شما را قادر مینماید تا نحوه نمایش جدول و اجزای آن را در صفحه مشخص نماید.

Border-collapse

این خاصیت مدل حاشیه را برای تگ Table را مشخص میکند.

این خاصیت دارای مقادیر زیر میباشد:

- **Separate**: در صورت استفاده از این مقدار تنها خانه های که در درون جدول وجود دارد میتواند حاشیه داشته باشد. اما ستون ها و ردیف ها نمیتواند دارای حاشیه باشند. یعنی تنها یک سلول از چهار طرف فاصله میگیرد. در ضمن این مقدار، مقدار پیشفرض میباشد.
- **Collapsing**: در صورت استفاده از این مقدار فاصله بین خانه های جدول و حاشیه های آنها قرار نمیگیرد.

Border-spacing

این تنها زمانی کاربرد دارد که از مقدار Separate برای جدول استفاده کرده باشید و این خاصیت میزان فاصله بین خانه های جدول را کنترل میکند.

آموزش کاربردی CSS3	
#results {	
Border-collapse: separate;	
Border-spacing: 1em 0.5em;	
}	

در مثال بالا قیمت 1em فاصله افقی را و 0.5em فاصله عمودی را بین خانه های جدولی تعیین میکند.

نکته: این خاصیت قادر است که اعداد منفی و در ضمن یک یا دو مقدار را بپذیرد. در صورت استفاده از دو مقدار، بصورت معمول قیمت اول مربوط به فاصله افقی و مقدار دوم مربوط به فاصله دومی میشود. اما اگر از یک قیمت استفاده نماید این قیمت هم مربوط به اندازه افقی و هم مربوط به اندازه عمودی آن میشود.



Empty-cells

با استفاده از این خاصیت میتوانید تعیین کنید که آیا رنگ پس زمینه و حاشیه های خانه های که محتویات ندارد نمایش داده شود یا خیر. بیاد داشته باشید این خاصیت تنها زمانی استفاده میگردد که از خاصیت Separate برای جدول استفاده کرده باشید اما در مقدار Collapsing در نظر گرفته نمیشود.

خاصیت Empty-cells دارای مقادیر زیر میباشد:

- Show: در صورت استفاده از این مقدار پس زمینه خانه های خالی نمایش داده میشود.
- Hide: در صورت استفاده از این مقدار پس زمینه خانه های خالی نمایش داده نمیشود و اگر مقدار تمامی خانه های موجود در یک ردیف جدول برابر با hide باشد و در ضمن این خانه ها نیز خالی باشد آن ردیف نمایش داده نمیشود.

Caption-side

این خاصیت مکان عنوان جدول را در جهت عمودی مشخص میکند. اما اگر خواسته باشید مکان افقی را تعیین نمایید میتوانید از خاصیت text-align استفاده نمایید.

این خاصیت دارای مقادیر زیر میباشد:

- Top: این مقدار مکان متن را در بالا قرار میدهد.
- Bottom: این مقدار مکان متن را در پایین قرار میدهد.
- Inherit: این مقدار، مقدار پیشفرض این خاصیت میباشد.



مبحث هفتم: Box Model

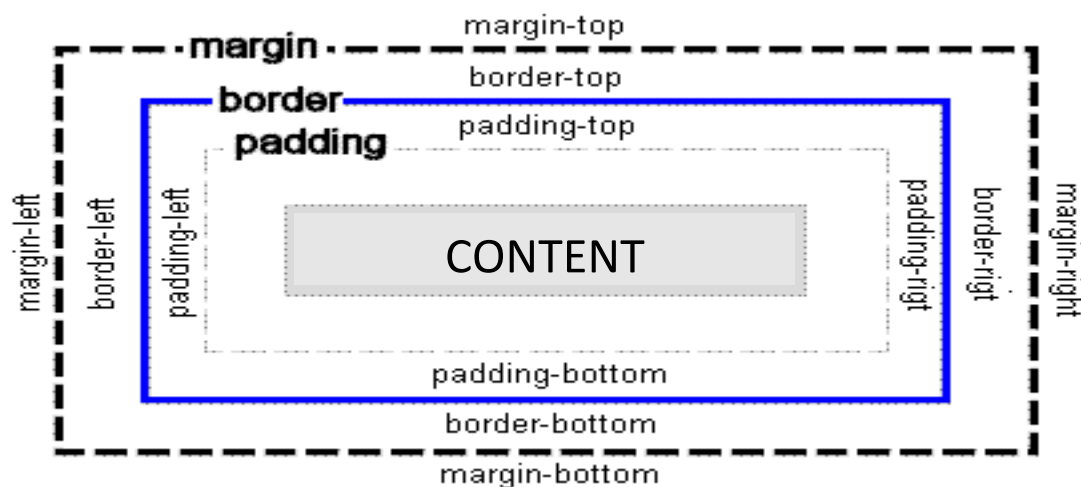
تمامی عناصر HTML به شکل یک باکس مطرح شده و در CSS زمانی از باکس مودل استفاده میشود که سخنی از دیزاین و ساختار مطرح شود.

تمامی عناصر HTML در CSS به صورت یک باکس تصور میشود. که این مبحث در طراحی و طرح بندی صفحات استفاده میشود.

Box model یا مدل کادری که در باکس قرار میگیرند قرار زیر اند:

- Margin ○
- Border ○
- Padding ○
- Content ○

تصویر زیر Box model را بصورت واضح تشریح میکند:



✓ **Margin:** این قسمت از Box model فضای اطراف Border را پوشش میدهد. در ضمن بیاد داشته باشید که شما نمیتوانید به این قسمت دیزاین از قبل پس زمینه تعیین نمایید.

✓ **Border:** این قسمت از Box model فضای اطراف Content و Padding را پوشش میدهد. در ضمن بیاد داشته باشید در اینجا میتوانید رنگ، سبک و ضخامت Border را تعیین نمایید.

✓ **Padding:** این قسمت از Box model فضای اطراف Content را پوشش میدهد. در ضمن شما میتوانید در این قسمت رنگ پس زمینه تعیین کنید.



✓ **Content**: این قسمت از Box mode محتوای عنصر را در بر میگیرد. یعنی این همان جایی است که متون و عکس ها قرار میگیرند.

نکته: برای تنظیم بهتر عرض و ارتفاع حقیقی یک عنصر درک مفهوم Box model بسیار مهم میباشد.

در قسمت بالا ما تنها تعریفی از اجزای Box model نمودیم، حال میرویم هر کدام این اجزا را به صورت مفصل مورد بحث قرار میدهم که البته در این مبحث تنها در مورد Margin و Padding صحبت میکنیم و دیگر اجزای آنرا به دلیل بزرگی در مباحث جداگانه در مورد آن صحبت میکنیم.

Margin

با استفاده از این خاصیت شما میتوانید محتوای سایت را از اصل صفحه فاصله دهید. اگر به زبان ساده بیان کنیم Margin فاصله بیرونی است.

آموزش کاربردی CSS3

```
<div class="content">
Afghanistan
</div>
```

در اینجا عنصر متن Afghanistan محتوا صفحه میباشد و اگر بخواهیم این متن را از Body به اندازه 50px حاشیه بدهیم از دستور زیر استفاده میکنیم.

آموزش کاربردی CSS3

```
.content{margin:50px;}
```

در این صورت محتوا از طرف راست، چپ، بالا و پایین به اندازه 50px فاصله میگیرد.

چگونه از Margin استفاده نماییم؟

قاعده دستوری Margin به شکل زیر میباشد.

آموزش کاربردی CSS3

```
Margin:margin-top margin-right bottom margin-left;
یعنی
Margin:10px 20px 15px 40px;
```



در مثال بالا ما به اندازه 10px از طرف بالا ، به اندازه 20px از طرف راست ، به اندازه 15px از طرف پایین و از به اندازه 40px از طرف چپ فاصله ایجاد کردیم.

انواع نوشتن margin در CSS

Margin:10px 5px 15px 20px ✓

▪ سمت بالا 10px

▪ سمت راست 5px

▪ سمت پایین 15px

▪ سمت چپ 20px

Margin:10px 5px 15px ✓

▪ سمت بالا 10px

▪ سمت راست و چپ 5px

▪ سمت پایین 15px

Margin:10px 5px ✓

▪ سمت بالا و پایین 10px

▪ سمت راست و چپ 5px

Margin:10px ✓

▪ سمت بالا، راست، پایین و چپ 10px

در بالا شما نوعی خلاصه نویسی را مشاهده کردید اما شما میتوانید هر کدام را به شکل مجزا بنویسید.

Margin-top فاصله از بالا ✓

Margin-right فاصله از راست ✓

Margin-bottom فاصله از پایین ✓

Margin-left فاصله از چپ ✓

در مثال زیر نحوه نوشتن به صورت واضح برای تان معلوم میگردد.



مثال:

آموزش کاربردی CSS3
Margin-top:20px;
Margin-right:30px;
Margin-left:10px;
Margin-bottom:15px;

Padding

این خاصیت در کل بمنظور ایجاد نمودن فاصله داخلی استفاده میشود.

آموزش کاربردی CSS3
<div class="content">
Afghanistan
</div>

در اینجا ما فاصله را از تک فرزند تطبیق میکنم. اگر ساده تر بگوییم! متن ما (Afghanistan) را در بین یک باکس مدنظر بگیرد فاصله باکس از اصل صفحه را Margin و فاصله متن را در خود باکس عبارت از Padding است.

آموزش کاربردی CSS3
.content{padding:30px;}

Padding فاصله ی محتوای درون تگ را با لبه ی تگ تعیین می کند. به طور کل padding را فاصله ی داخلی می نامیم .

چگونه از padding استفاده نماییم؟

قاعده دستوری Padding به شکل زیر میباشد.

آموزش کاربردی CSS3
Padding:padding-top padding-right padding-bottom padding-left
یعنی
Padding:10px 20px 15px 40px



در مثال بالا ما به اندازه 10px از طرف بالا ، به اندازه 20px از طرف راست ، به اندازه 15px از طرف پایین و از به اندازه 40px از طرف چپ از لبه تگ تا محتوای داخل تگ فاصله ایجاد کردیم.

انواع نوشتن Padding در CSS

padding:10px 5px 15px 20px ✓

○ سمت بالا 10px

○ سمت راست 5px

○ سمت پایین 15px

○ سمت چپ 20px

Padding:10px 5px 15px; ✓

○ سمت بالا 10px

○ سمت راست و چپ 5px

○ سمت پایین 15px

Padding:10px 5px ✓

○ سمت بالا و پایین 10px

○ سمت راست و چپ 5px

Padding:10px ✓

○ سمت بالا، راست، پایین و چپ 10px

در بالا شما نوعی خلاصه نویسی را مشاهده کردید اما شما میتوانید هر کدام را به شکل مجزا بنویسید.

padding-top فاصله از بالا ✓

padding-right فاصله از راست ✓

padding-bottom فاصله از پایین ✓

Margin-left فاصله از چپ ✓

در مثال زیر نحوه نوشتن به صورت واضح برای تان معلوم میگردد.



مثال:

آموزش کاربردی CSS3
Padding-top:20px;
Padding -right:30px;
Padding -left:10px;
Padding -bottom:15px;



مبحث هشتم: Border

این خاصیت از جمله خاصیت های پرکار برد میباشد که البته توسط این خاصیت میتوانید یک خط با ضخامت دلخواه برای حاشیه ، باکس ها ، تصاویر و تمامی عناصر های موجود در صفحه ایجاد کنید.

بطور مثال شما میخواهید به دور تصویر تان یک خط با ضخامت مشخص با رنگ مشخص اضافه نمایید تا تصویر تان زیباتر دیده شود.

نوع استفاده از خاصیت Border به شکل ذیل میباشد:

آموزش کاربردی CSS

Border: border-width border-style border-color | initial | inherit

Border-width: ضخامت خط را مشخص می کند.

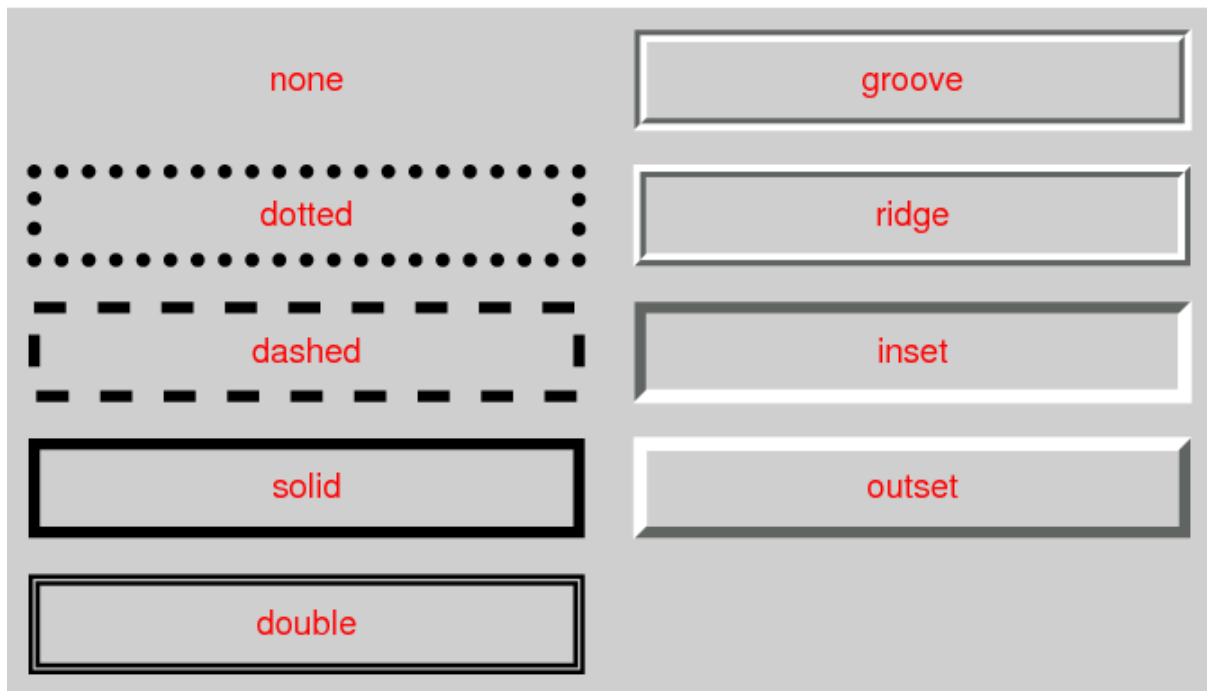
Border-style: استایل خط را مشخص میکند که شامل مقادیر زیر میباشد:

- None: خط را حذف میکند.
- Hidden: برابر با none است و خط را حذف میکند.
- Dotted: خط را بصورت نقطه چین نمایش میدهد.
- Dashed: خط را بصورت خط چین نمایش میدهد.
- Solid: خط را بصورت ممتد و صاف نمایش میدهد.
- Double: خط را بصورت دو خط موازی نمایش میدهد.
- Groove: خط را بصورت سه بعدی از داخل نمایش میدهد.
- Ridge: خط را بصورت سه بعدی از خارج نمایش میدهد.
- Inset: خط را بصورت سه بعدی از داخل نمایش میدهد.
- Outset: خط را بصورت سه بعدی از خارج نمایش میدهد.
- Initial: مقدار پیش فرض و قرار میدهد.



- **Inherit**: با توجه به مقدار عنصر والد مقدار و مشخص میکند. مثلا اگر برای عنصر والد border-style مقدار solid مشخص شده باشد ، برای عنصر فرزند هم مقدار solid مشخص میشود.

تصویر زیر تمامی خاصیت های بالا را به شکل واضح بیان میکند:



اما چگونه از این خاصیت ها استفاده نماییم؟؟؟

خوب یک خاصیت دیگر را نیز میخوانیم و بعد بالای نوع نوشتن آن میرویم!.

Border-color: رنگ خط را مشخص میکند. برای تعیین رنگ میتوانید از نام رنگ ها ، کد های hex و rgb ها استفاده کنید . حتی میتوانید با rgba مقدار opacity خط را کم و زیاد کنید.

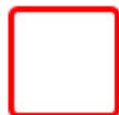
نحوه استفاده از این خاصیت بصورت خلاصه نویسی و استاندارد به شکل زیر است:

آموزش کاربردی CSS3
Border:3px solid red;
Border:1px dashed #000
Border:3px dotted rgba(0,0,0,0.7)



Border Radius

گاهی اوقات شما میخواهید دور تصاویر، باکس ها و element های تان را گرد کنید و شکلی شبیه به تصویر زیر ایجاد کنید:



برای این کار از خاصیت border-radius استفاده می کنید. با این خاصیت میتوانید دور آنها را به صورت منحنی گرد کنید. نحوه ی استفاده از این خاصیت به شکل زیر است:

آموزش کاربردی CSS3

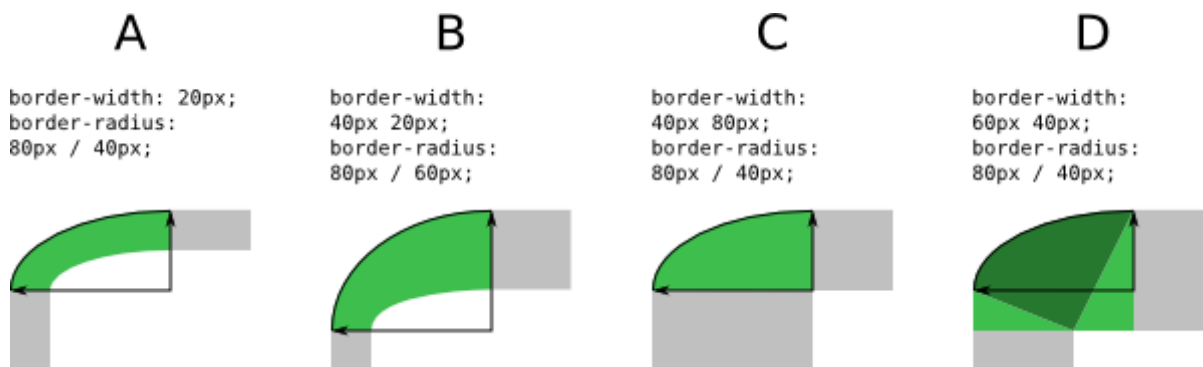
```
Border:3px solid red;
Border-radius:5px;
```

توجه: این خاصیت در مرورگر های قدیمی و یا مرورگر های مثل Chroom,Safari,Firefox... باید از دستور زیر با دستور بالا استفاده نمایید.

آموزش کاربردی CSS3

```
-moz-border-radius:5px;
-webkit-border-radius:5px;
```

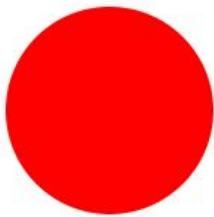
تصویر زیر تصویر تخصصی تر می باشد که در صورت نیاز میتوانید از دستورات آن استفاده نمایید.



مبحث نهم: ساخت اشکال با Border

در این مبحث در مورد اشکال هندسی صحبت میکنیم. در این قسمت خواهید آموخت که چگونه یک شکل هندسی درست کرد، چگونه اندازه آنرا تغییر داد و امثال اینها که در ادامه تمامی این موارد را مورد بحث قرار میدهیم در ضمن یک نکته را به یاد داشته باشید که در CSS دستوری برای درست کردن اشکال هندسی نیست اما میتوانید با استفاده از دستورات بالا اشکال هندسی درست نمایید.

در مثال های زیر اولاً شکل آن نشان داده شده و بعد دستورات آن.



۱- دایره: برای درست کردن دایره میتوانید از دستور زیر استفاده نمایید.

آموزش کاربردی CSS3	
#circle{	
Width:140px;	
Height:140px;	
Background:red;	
-moz-border-radius:70px;	
-webkit-border-radius:70px;	
Border-radius:70px;	
}	

۲- بیضوی: برای درست کردن بیضوی میتوانید از دستور زیر استفاده نمایید.



آموزش کاربردی CSS3

```
#oval{
    Width:200px;
    Height:100px;
    Background:purple;
    -moz-border-radius:100px / 50px;
    -webkit-border-radius:100px / 50px;
    Border-radius:100px / 50px;
}
```

۳- سه ضلعی (بالا): برای درست کردن سه ضلعی در جهت بالا از دستور زیر استفاده نمایید:



آموزش کاربردی CSS3

```
#Up-triangle{
    Width:0;
    Height:0;
    Border-bottom:120px solid green;
    Border-left:60px solid transparent;
    Border-right:60px solid transparent;
}
```

۴- سه ضلعی (پایین): برای درست کردن سه ضلعی در جهت پایین از دستور زیر استفاده نمایید



آموزش کاربردی CSS3

```
#down-triangle{
    Width:0;
    Height:0;
    Border-top:80px solid pink;
    Border-left:60px solid transparent;
    Border-right:60px solid transparent;
}
```

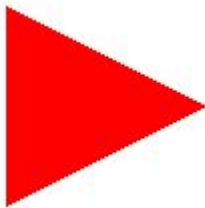


۵- سه ضلعی (چپ): برای درست کردن سه ضلعی در چپ میتوانید از دستور زیر استفاده نمایید.



آموزش کاربردی CSS3	
#left-triangle{	
Width:0;	
Height:0;	
Border-right:100px solid orange;	
Border-top:50px solid transparent;	
Border-bottom:50px solid transparent;	
}	

۶- سه ضلعی (راست): برای درست کردن سه ضلعی در راست میتوانید از دستور زیر استفاده نمایید:



آموزش کاربردی CSS3	
#right-triangle{	
Width:0;	
Height:0;	
Border-left:100px solid red;	
Border-top:50px solid transparent;	
Border-bottom:50px solid transparent;	
}	

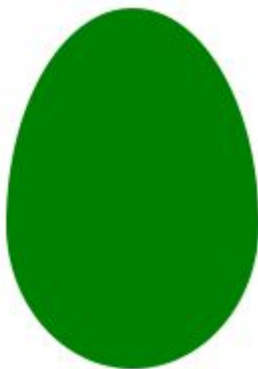


۷- چهار ضلعی غیر منظم: برای درست کردن چهار ضلعی غیر منظم از دستور زیر میتوانید استفاده نمایید.



آموزش کاربردی CSS3	
#trapezium{	
Width:0;	
Height:80px;	
Border-bottom:80px solid blue;	
Border-left:40px solid transparent;	
Border-right:40px solid transparent;	
}	

۸- بیضوی: برای درست کردن شکل بیضوی غیر منظم از دستور زیر میتوانید استفاده نمایید.



آموزش کاربردی CSS3	
#egg{	
Display:block;	
Width:126px;	
Height:180px;	
Background-color:green;	
-webkit-border-radius:63px 63px 63px 63px / 108px 108px 72px 72px;	
Border-radius:50% 50% 50% 50% / 50% 50% 40% 40%;	
}	



۹- دایره قطع شده: برای درست کردن دایره به شکلی که یک قسمت آن قطع شده باشد میتواند از دستور زیر استفاده نمایید.



آموزش کاربردی CSS3	
#pcman{	
Width:0;	
Height:0;	
Border-right:60px solid transparent;	
Border-top:60px solid yellow;	
Border-left:60px solid yellow;	
Border-bottom:60px solid yellow;	
Border-top-left-radius:60px;	
Border-top-right-radius:60px;	
Border-bottom-left-radius:60px;	
Border-bottom-right-radius:60px;	
}	

۱۰- دایره با چند رنگ مختلف: برای درست کردن یک دایره با چند رنگ مختلف میتوانید از دستور زیر استفاده نمایید:

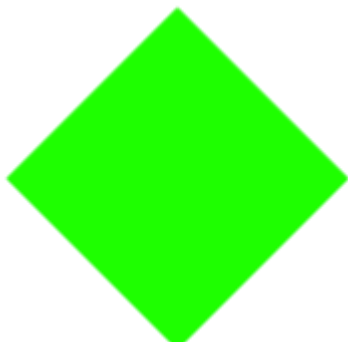


آموزش کاربردی CSS3	
#biohazard{	
Width:0;	
Height:0px;	
Border-bottom:60px solid black;	
Border-top:60px solid black;	
Border-top:60px solid yellow;	
Border-right:60px solid yellow;	
Border-radius:60px }	



۱۱- نحوه ساخت Diamond: برای درست شکل الماسی میتوانید از دستور زیر استفاده

نمایید.



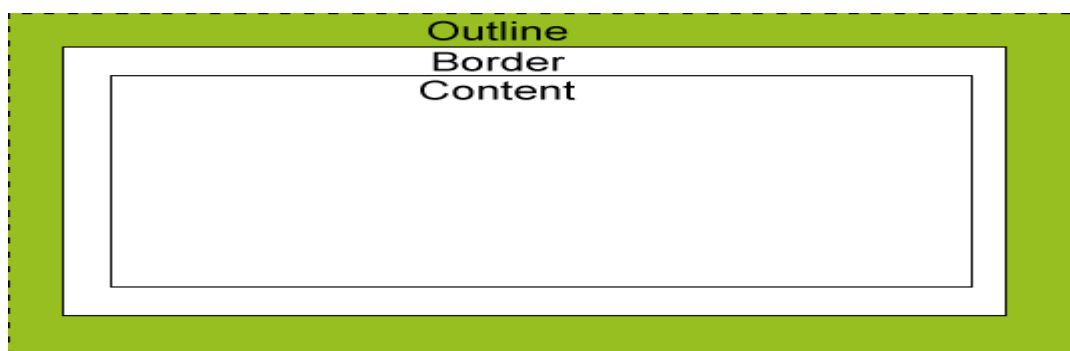
آموزش کاربردی CSS3
#diamond{
Width:120;
Height:120px;
/*Rotate */
-webkit-transform: rotate (-45deg);
-moz-transform: rotate (-45deg);
-ms-transform: rotate (-45deg);
-o-transform: rotate(-45deg);
-transform: rotate (-45deg);
/* Rotate Origin */
-webkit-transform-origin: 0 100%;
-moz-transform-origin: 0 100%;
-ms-transform-origin: 0 100%;
-o-transform-origin: 0 100%;
transform-origin: 0 100%;
margin: 60px 0 10px 310px;
}



مبحث دهم: outline

Outline ها به دور حاشیه ها کشیده میشوند و در حقیقت یک حاشیه دیگر را برای تگ بوجود میآورد. تفاوت outline ها با حاشیه ها در این است که outline ها برخلاف حاشیه ها، فضای که اشغال میکنند جزء فضای تگ محسوب نمیشود و در صورتی که مقادیر که این قاعده میپذیرد برابر به transparent باشد در این حالت برای حاشیه تگ، رنگی که در نظر گرفته نمیشود اما همچنان فضایی را به میزان مقدار border-width که برای تگ تعیین شده است اشغال میکند.

نکته: قبل از اینکه وارد بحث outline شویم این نکته را بیاد داشته باشید که شما نمیتوانید به حاشیه ای خاص خاصیت Outline را اعمال کنید. یعنی زمانی که شما از خاصیت Outline استفاده میکنید، بالای حاشیه های هر چهار سمت تگ اعمال میگردد.



خاصیت outline دارای مقادیر زیر است:

- **Outline-color:** این مقدار رنگ outline را که بدور حاشیه کشیده میشود تعیین مینماید.
- **Outline-style:** این مقدار طرح و دیزاین outline را که بدور حاشیه کشیده میشود را تعیین میکند. و تمام طرح ها و مقادیر که در border-style بود میتوانید در اینجا نیز استفاده نمایید.
- **Outline-width:** این مقدار پهنای outline را که به دور حاشیه کشیده میشود تعیین میکند.
- **Outline:** این مقدار شکل میانبری برای سه قاعده بالا است و باید به صورت زیر مقدار دهی شود.

آموزش کاربردی CSS3

```
Outline: { [outline-width] [outline-style] [outline-color] | inherit};
```



مبحث یازدهم: Position

خاصیت Position از جمله خاصیت های مهم میباشد که با استفاده از این خاصیت شما میتوانید مکان یک عنصر را نظر به موقعیت بصورت پیکسلی تنظیم نمایید. و یا بعضی اوقات ضرورت میشود که چند لایه را روی هم قرار بدهید که در این مواقع از خاصیت Position استفاده میکنید.

خاصیت Position شامل مقادیر زیر میباشد:

✓ **Static**: در صورتی استفاده از این مقدار شما حالت پیشفرض مرورگر را لحاظ میکنید و قیمت خود Position را خنثی میکند.

✓ **Relative**: این مقدار زمانی استفاده نمایید که بخواهید خاصیت های

Right, Left, Top, Bottom را بکار ببرید و همچنین این مقدار را برای عنصر والد عنصر

فرزندی قرار میدهید که مقدار Position آن عنصر فرزند Absolute باشد.

✓ **Absolute**: زمانی که از این مقدار استفاده می کنید ، عنصر در گوشه ی کادر عنصر والد

قرار میگیرد (عنصر والدی که مقدار relative را داشته باشد) در صورتی که عناصر

والد آن خاصیت position و یا مقدار relative را نداشته باشند ، عنصر مورد نظر body را

عنصر والد در نظر میگیرد و گوشه ی کادر صفحه قرار میگیرد. همچنین عنصری که

مقدار absolute را دارد روی همه ی عناصر قرار میگیرد . اگر چند عنصر این مقدار را

داشته باشند با z-index ترتیب قرارگیری عناصر روی هم را تعیین می کنید.

✓ **Fixed**: این مقدار را زمانی استفاده میشود که خواسته باشید یک پنجره را در مرورگر ثابت

قرار دهید، یعنی حتی با اسکرول صفحه هم عنصر حرکت نمیکند و ثابت باقی میماند.

✓ **Initial**: در صورت استفاده از این مقدار، حالت پیشفرض مرورگر تطبیق میگرد.

✓ **Inherit**: در صورت استفاده از این مقدار، مرورگر مقدار عنصر والد را به عنصر فرزند

اختصاص میدهد.

بطور مثال اگر عنصر والد مقدار Absolute داشته باشد ، عنصر فرزند هم مقدار Inherit داشته باشد،

مقدار Absolute را از عنصر والد به ارث میبرد.



خاصیت های Top, Right, Left, Bottom شامل مقادیر زیر میباشد:

قبل از استفاده از مقادیر باید خاصیت Position را استفاده کنید. البته برای کار با این خاصیت ها بصورت معمولی از Position با مقدار Relative استفاده میکنیم.

✓ Top: این مقدار فاصله عنصر را از بالا تنظیم میکند که شما میتوانید از واحد های %،PX... استفاده نمایید.

✓ Right: این مقدار فاصله عنصر را از راست تنظیم میکند که شما میتوانید از واحد های %،PX... استفاده نمایید.

✓ Bottom: این مقدار فاصله عنصر را از پایین تنظیم میکند که شما میتوانید از واحد های %،PX... استفاده نمایید.

✓ Left: این مقدار فاصله عنصر را از چپ تنظیم میکند که شما میتوانید از واحد های %،PX... استفاده نمایید.

Z-index

خاصیت Z-index زمانی استفاده میشود که خواسته باشید چند لایه را روی هم قرار میدهید.

در کد زیر خاصیت های Z-index و position نیز استفاده گردیده برای درک بیشتر موضوع در این جا از خلاصه نویسی استفاده نمی کنیم اما شما تا حد امکان از خلاصه نویسی استفاده نمایید.

آموزش کاربردی CSS3
.main{ Width:400px; height:400px; background-color:pink; padding:0; margin:0;
.div{
Width:250px;
Height:250px;
Font:25px/25px Tahoma;
Text-align:center;
Color:#fff; background-color:blue;
Position:relative;
Top:20px; left:20px;

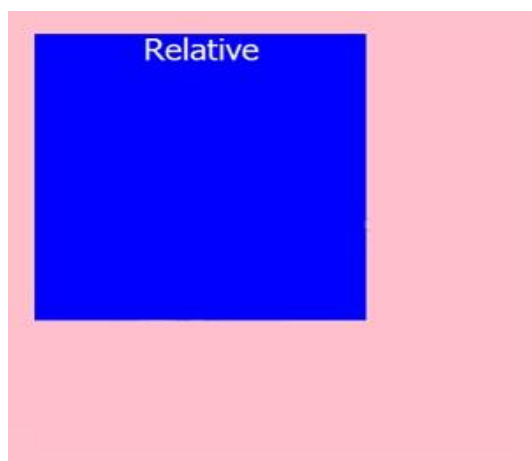


کد HTML

آموزش کاربردی CSS3
<div class="main">
<div class="div1">Relative</div>
</div>

شرح کد:

در کد بالا اولاً ما صفحه را (Body) را با رنگی گلابی درست کردیم البته فقط برای درک بهتر و در ضمن یک باکسی دیگر بنام Div1 درست کردیم و برای اینکه این باکس را به مکان دلخواه انتقال دهیم از خاصیت های Top و Left و همچنان position را با مقدار Relative استفاده کردیم به اندازه



20px از چپ و 20px از بالا فاصله دادیم و در نتیجه شکل زیر بدست آمد:

خوب شاید بگویند که چگونه یک باکس دیگری را بالای باکس آبی قرار دهیم؟ به کد زیر توجه کنید:

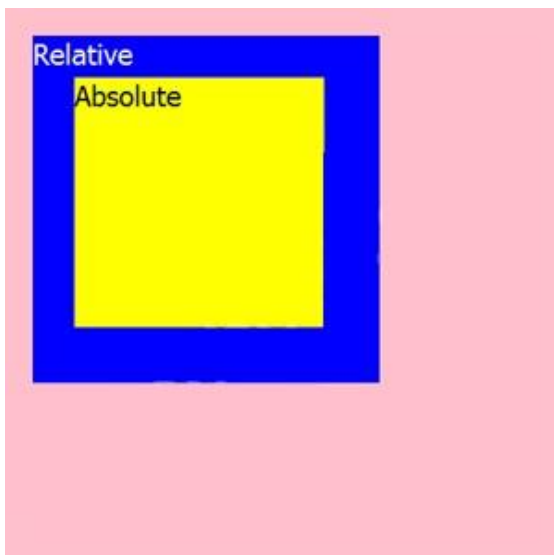
آموزش کاربردی CSS3
.main{ width:400px; height:400px; background-color:pink; padding:0; margin:0;
.div1{
Width:250px; height:250px; font:20px/25px Tahoma;
Color:#fff; background-color:blue;
Position:relative; top:20px; left:20px;
}
.div2{
width:180px; height:180px; font:20/25px Tahoma;
Color:#000; background-color:yellow;
Position:absolute; top:30px; left:30px;
}



آموزش کاربردی CSS3
<div class="main">
<div class="div1">Relative
<div class="div2">Absolute</div>
</div>
</div>
<div class="main">
<div class="div1">Relative
<div class="div2">Absolute</div>
</div>
</div>

شرح کد:

در کد بالا ما باکس زرد رنگی را به عنوان فرزند داخل باکس آبی (عنصر والد) قرار دادیم و چون نسبت به باکس والد می‌خواهیم موقعیت دهی کنیم باید مقدار Absolute را برایش بدهیم و در نتیجه شکل زیر بوجود آمد:



خوب حالا دیگر هم این باکس‌ها توسعه می‌دهیم یعنی یک باکسی دیگر (سرخ رنگ) را میان باکس آبی و زرد اضافه نماییم. برای انجام این کار چون باکس سرخ رنگ عنصر فرزند باکس



آبی رنگ میشود باید مقدار Absolute برایش بدهیم. و برای اینکه بتوانیم باکس سرخ رنگ را زیر باکس زرد قرار بدهیم یعنی دو لایه را روی هم قرار بدهیم باید از Z-index استفاده کنیم.

کد CSS

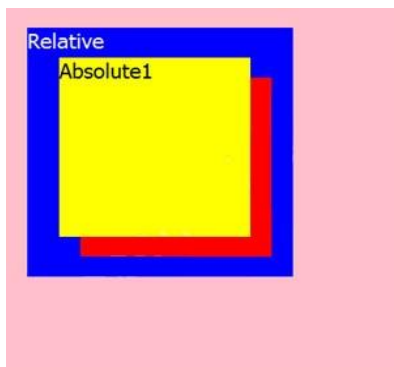
آموزش کاربردی CSS3
.main{ width:400px; height:400px; background-color:pink; padding:0; margin:0;
.div1{
Width:250px; height:250px; font:20px/25px Tahoma;
Color:#fff; background-color:blue;
Position:relative; top:20px; left:20px;
}
.div2{
width:180px; height:180px; font:20/25px Tahoma;
Color:#000; background-color:yellow;
Position:absolute; top:30px; left:30px; z-index:10;
}
.div3{
Width:180px; height:180px; font:20px/25px Tahoma;
Color:#fff; background-color:red;
Position:absolute; top:50px; left:50px; z-index:1;
}

کد HTML

آموزش کاربردی CSS3
<div class="main">
<div class="div1">Relative
<div class="div2">Absolute1</div>
<div class="div3">Absolute2
</div>
</div>



در کد بالا ما به باکس زرد رنگ z-index با مقدار ۱۰ دادیم و باکس سرخ رنگ z-index با مقدار ۱ دادیم. در این صورت باکس زرد رنگ بروی باکس سرخ رنگ قرار گرفت پس به این نتیجه میرسیم که هر عنصر که مقدار Z-index آن زیاد باشد بالای عنصر میاید که مقدار Z-index آن کمتر باشد.



شکل زیر حاصل شده از کد بالا میباشد:

حالا میخواهیم که جاهای باکس زرد رنگ و سرخ رنگ را تغییر دهیم برای این کار چه کنیم؟
بلی واضح است باید مقدار Z-index های شان را تغییر دهید.
حال میرویم و یک مثال از fixed را برای تان نمایش میدهم که قبلاً به آن اشاره نموده بودیم.
و این هم مثال:

در اول باید یک عنصر را در صفحه HTML تان درست کنید و بعد با استفاده از کد زیر در CSS
میتوانید آنرا به حالت Fixed قرار دهید.

آموزش کاربردی CSS

```
.fixed{width:300px; height:100px; font:20px/25px Tahoma ; color:#fff; background-color:blue; position:fixed; top:20px; left:0px;
```



مبحث دوازدهم: تنظیم طول و عرض

در این قسمت از آموزش CSS در مورد طول و عرض در CSS صحبت میکنیم.

۱- خاصیت Height:

خاصیت Height شامل مقادیر زیر میباشد:

- **Auto:** در صورت استفاده از این مقدار مرورگر بصورت خودکار ارتفاع را تشخیص میدهد و در ضمن این مقدار، مقدار پیش فرض خاصیت میباشد.
- **Length:** در صورت استفاده از این مقدار شما میتوانید با استفاده از واحد های px,em,% ارتفاع را تعیین نمایید.
- **Initial:** در صورت استفاده از این مقدار، مقدار پیش فرض تطبیق میگردد.
- **Inherit:** در صورتی که عنصر فرزند این مقدار را داشته باشد، هر چه که مقدار عنصر والد باشد به ارث میبرد.

۲- خاصیت Min-height

با استفاده از این خاصیت تعیین میکنید که حد اقل ارتفاع عنصر چقدر باشد.

۳- خاصیت Max-height:

با استفاده از این خاصیت تعیین میکنید که حد اکثر ارتفاع عنصر چقدر باشد.

توجه داشته باشید که اگر محتوای داخل عنصر بیشتر از این ارتفاع باشد خارج از کادر نمایش داده میشود چرا که عنصر بیشتر از حد اکثری که برایش تعیین شده ارتفاع نمیگیرد.

۴- خاصیت width:

با استفاده از این خاصیت میتوانید عرض یک عنصر را مشخص کنید.

خاصیت Width شامل مقادیر زیر میشود:

- **Auto:** در صورت استفاده از این مقدار مرورگر بصورت خودکار ارتفاع را تشخیص میدهد و در ضمن این مقدار، مقدار پیش فرض خاصیت میباشد.



- **Length:** در صورت استفاده از این مقدار شما میتوانید با استفاده از واحد های px,em,% ارتفاع را تعیین نمایید.
- **Initial:** در صورت استفاده از این مقدار ، مقدار پیش فرض تطبیق میگردد.
- **Inherit:** در صورتی که عنصر فرزند این مقدار را داشته باشد ، هر چه که مقدار عنصر والد باشد به ارث میبرد.

۵- خاصیت Min-width:

با استفاده از این خاصیت حداقل عرض را برای عنصر تعیین میکنیم.

۶- خاصیت Max-width:

با استفاده از این خاصیت حد اکثر عرض را برای عنصر تعیین میکنیم.



مبحث سیزدهم: Direction

در این مبحث در مورد جهات صحبت میکنیم. شاید تا به حال برای تان پیش آمده باشد که متنی را از یک جهت به جهتی دیگری انتقال بدهید. موضوع Direction از جمله خاصیت های مهم میباشد.

Direction دو مقدار اصلی rtl و ltr را به خودش اختصاص میدهد:

rtl) Right to Left): با استفاده از rtl میتوانید محتوای سایت را راست به چپ مرتب کنید که در سایت های فارسی از این مقدار استفاده میشود.

ltr) Left to right): با استفاده از ltr میتوانید محتوای سایت را چپ به راست مرتب کنید که در سایت های انگلیسی از این مقدار استفاده میشود.

مقادیر inherit و initial هم میتوانید استفاده کنید:

Inherit: مقدار پیشفرض مرورگر را به عنصر اختصاص میدهد.

Initial: مقدار عنصر والد را به عنصر فرزند اختصاص میدهد. در صورتی که عنصر فرزند مقدار initial را استفاده کند و عنصر والد دارای مقدار rtl باشد، عنصر فرزند هم از rtl استفاده میکند.

قاعده دستوری direction

Direction: rtl

Direction: ltr

نکته: زمانی که یک سایت فارسی را کدنویسی می کنید حتما برای عنصر body این خاصیت rtl تطبیق کنید در این صورت دیگر نیازی نیست برای تک تک عناصر این خاصیت را تعریف کنید. با تعریف rtl زمانی که شما متون فارسی و انگلیسی را بصورت ترکیبی مینویسید، بصورت مرتب به شما نمایش داده میشود. اما اگر rtl را تعریف نکرده باشید، ترکیب متون انگلیسی و فارسی با مشکل بر میخورد.



مبحث چهاردهم: Clear and float

در این مبحث راجع به Float و Clear صحبت میکنیم.

این دو خاصیت کاربرد بسیار زیادی در CSS دارد. با استفاده از این خاصیت شما میتوانید محل قرار گیری عناصر را مشخص کنید. به عنوان مثال شما چند ستون دارید و قصد دارید این ستون ها را کنار هم قرار بدهید، برای این منظور از Float استفاده کنید و ستون ها را کنار هم قرار میدهید.

به تصویر زیر دقت کنید!



خاصیت float دارای مقادیر زیر میباشد:

- None: در صورتی استفاده از این مقدار خاصیت Float خنثی میگردد.
- Left: در صورتی استفاده از این مقدار عنصر انتخاب شده به سمت چپ انتقال پیدا میکند.
- Right: در صورتی استفاده از این مقدار عنصر انتخاب شده به سمت راست انتقال پیدا میکند.
- Initial: در صورتی استفاده از این مقدار، مقدار پیش فرض مرورگر تطبیق میگردد.
- Inherit: در صورتی که برای عنصر والد مقدار Left را تطبیق کنید، مقدار Left بالای عنصر والد را به خودش اختصاص میدهد.



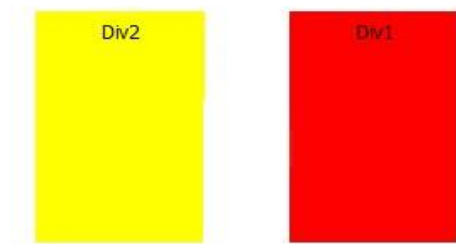
به کد زیر دقت کنید:

آموزش کاربردی CSS3
Div{font:12px/25px Tahoma; width:10px; height:150px; color:#000;text-align:center;}
Div1{background-color:red; float:right;}
Div2{background-color:yellow; float:left;}

کد HTML این ساختار:

آموزش کاربردی CSS3
<div class="div1">Div1</div>
<div class="div2">Div2</div>

در کد بالا به Div1 مقدار Right را دادیم و برای Div2 مقدار left را دادیم، حالا نتیجه به شکل زیر خواهد بود.



با استفاده از دستورالعمل های بالا شما میتوانید چندین ستون را باهم یکجا درج نمایید. اما به دو نکته زیر توجه داشته باشید.

نکته اول

این نکته از جمله مهمترین نکته میباشد. هنگامی که شما از خاصیت float برای عناصر موجود در صفحه استفاده میکنید، برای اینکه عناصر بعدی به درستی سر جای خودشان قرار بگیرند و به درستی نمایش داده شوند باید از خاصیت Clear در عنصر بعدی استفاده کنید تا خاصیت Float خنثی کند.

حالا ما میخواهیم بعد از این دو ستون یک ستون قرار بدهیم.



کد CSS

آموزش کاربردی CSS3
Div{font:12px/25px Tahoma; width 100px; height:150px; color:#000; text-align:center;}
.div1{background-color:red; float:right;}
.div2{background-color:yellow; float:left;}
.div3{background-color:pink; width:100%;}

کد HTML

آموزش کاربردی CSS3
<div class="div1">Div1</div>
<div class="div2">Div2</div>
<div class="div3">Div3</div>

خوب برای درک بهتر ما اولاً از دستور Clear استفاده نکردیم و حالا نتیجه چنین است:



در شکل بالا عنصر سوم با مشکل مواجه شد و نتوانستیم بعد از این عناصر قرارش بدهیم. برای حل این مشکل از خاصیت Clear استفاده میکنیم. به دستور زیر دقت نمایید.

آموزش کاربردی CSS3
.div3{width:100%; background-color:pink; clear:both;}

حالا با استفاده از خاصیت Clear و مقدار Both هر دو مقدار Float عناصر قبلی را خنثی کردیم.



نتیجه:



حالا ما با درج Clear ما خاصیت Float را خنثی کردیم تا اثرات Float در عنصر بعدی تاثیر نگذارد.

خوب حالا شاید سوالی برای تان پیش شده باشد که آیا میتوانیم دیگر جهات را نیز خنثی نماییم؟

در جواب باید گفت بلی!

دستور Clear مقادیر زیر را دارا میباشد.

None: در صورت استفاده از این خاصیت عناصر قبلی که شامل خاصیت Float بودند را خنثی نمی کند. یعنی استفاده آن معنی ندارد اما در صورت که میخواهید گزینهء بگذارید تا بتوانید بعداً تغییر دهید.

Right: عنصری که Float با مقدار Right دارد را خنثی میکند.

Left: عنصری که Float با مقدار Left دارد را خنثی میکند.

Inherit: مقدار عنصر والدش را به خودش میگیرد.

Initial: مقدار پیشفرض مرورگر را تطبیق میکند.



نکته دوم

این مشکل در بعضی موارد باعث مشکل میشود. گاهی اوقات ضرورت میشود که شما محتویات صفحه تانرا در سه ستون و چند ردیف نمایش بدهد. در اینجا شما برای هر ستون خاصیت Float را استفاده میکنید تا در کنار هم قرار بگیرند، برای اینکه یک فاصلهء هم بین باکس ها ایجاد کنید با از خاصیت Margin نیز استفاده نماید که البته در مورد Margin در قسمت های قبلی صحبت کردیم.

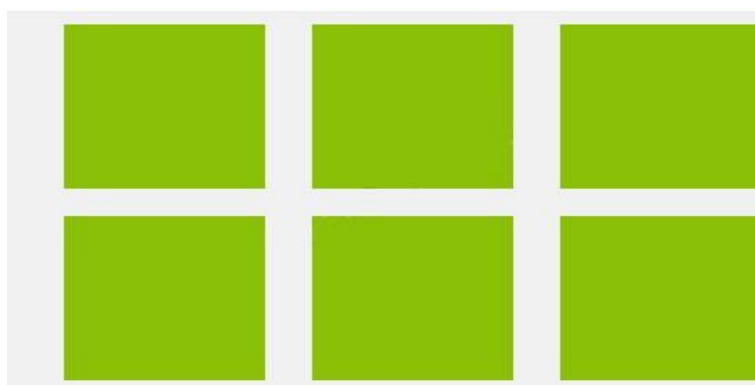
کد HTML

کد HTML

آموزش کاربردی CSS3
<div class="main">
<section></section>
<section></section>
<section></section>
<section></section>
<section></section>
<section></section>
<div class="clr"></div>
</div>

کد CSS

آموزش کاربردی CSS3
.main{width:45px; background-color:#F1F1F1;}
.clr{clear:right;}
Section{width:120px; height:120px; float:right; background-color:#8AC007; margin 10px 0px 10px 28px;}



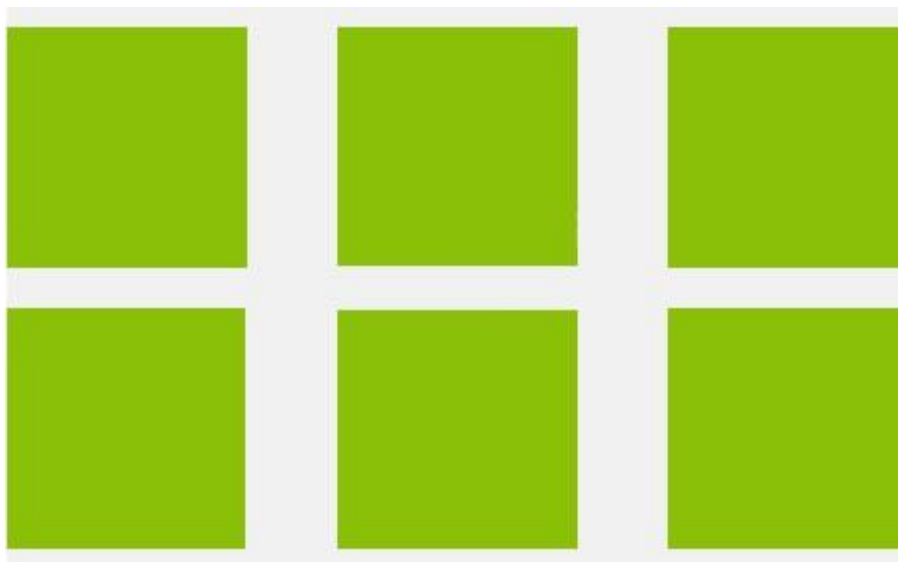
اگر با دقت مشاهده کنید ، متوجه میشوید که با دادن margin باکس آخر هر ردیف هم شامل margin-left شده و در سمت چپ باکس یک فاصله ای ایجاد کرده که باعث شده به لبه ی کادر نجسپد که این برای نمایش محصولات ما مناسب نیست . هدف ما ایجاد margin بین باکس ها است اما نه باکس های آخر هر ردیف . ما میخواهیم margin-left باکس های آخر هر ردیف را برابر با صفر قرار بدهیم تا مثل باکس های اول هر ردیف به لبه ی کادر بچسپد.

برای این کار از nth-child ها استفاده می کنیم . به کد css زیر دقت کنید:

کد CSS

آموزش کاربردی CSS3
.main{width:450px; background-color:#F1F1F1;}
.clr{clear:right;}
Section{width:120px; height:120px; float:right; background-color:#8AC007; margin:10px 0px 10px 45px}
Section:nth-child(3n+3){margin-left:0;}

حالا نتیجه به شکل زیر است:



میبینید که با استفاده از nth-child ها توانستیم margin-left باکس های آخر و برابر با صفر قرار دادیم تا نظم خاصی بگیرند و به درستی نمایش داده شوند .



مبحث پانزدهم: box-shadow

در این مبحث کتاب در مورد Box-shadow صحبت میکنم گرچه امروزه ابزار های آنلاین و افزونه برای مرورگر ها کار را بسیار ساده کرده اما دستور نوشتن و تغییرات در دستورات را برای تان توضیح میدهیم.

توسط Box-shadow قسمی که از نامش پیدا است برای دادن سایه برای باکس ها استفاده میشود. و البته این باکس هم اندازه باکس اصلی میباشد اما با تغییر دادن مختصات عمودی و افقی یک سایه برای باکس ایجاد میگردد.

قاعده دستوری box-shadow

آموزش کاربردی CSS3

```
Box-shadow:h-shadow v-shadow blur spread color inset;
```

مقادیر خاصیت Box-shadow

- **H-shadow**: برای باکس انتخاب شده سایه افقی لحاظ میکند.
- **V-shadow**: برای باکس انتخاب شده سایه عمودی لحاظ میکند.
- **Blur**: سایه باکس را به شکل Blur میسازد.
- **Spread**: سایه باکس را به شکل پراکنده میسازد و از اصل باکس بزرگتر میشود.
- **Color**: رنگ سایه را تعیین میکند.
- **Inset**: اگر شما مقدار inset را در انتهای کد قرار بدهید سایه را در داخل باکس ایجاد میکند.
- برای درک بهتر از این موضوع به مثالهای زیر توجه نمایید:

کد HTML

آموزش کاربردی CSS3

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

کد CSS

آموزش کاربردی CSS3

```
.box{width:200px; height:100px; background-color:yellow;
```



خوب حالا از دستورات بالا شکل زیر بدست میآید.



خوب حالا قصد داریم Box-shadow را روی مستطیل با رنگ سیاه تطبیق کنیم.

آموزش کاربردی CSS3	
.box{	
Width:200px; height:100px; background-color:yellow;	
Box-shadow:10px 15px black;	
}	

در دستور بالا ما گفتیم که ۱۰ پیکسل بصورت افقی از باکس اصلی فاصله بگیرد و ۱۵ پیکسل بصورت عمودی از باکس اصلی فاصله بگیرد، رنگ سایه هم سیاه باشد و حالا نتیجه هم چنین است:



اگر به یاد داشته باشید در شروع مبحث گفته بودیم که اندازه سایه این باکس کاملاً با سایز باکس اصلی است، برای اینکه متوجه شوید، مقادیر H-shadow و v-shadow را برابر به صفر قرار میدهیم.

آموزش کاربردی CSS3	
Box-shadow:0 0 black	

و حالا نتیجه به شکل زیر است:



حالا میتوانید با تغییر مقادیر h-shadow و v-shadow میتوانید باکس سایه را جابجا کنید.

در مثال زیر برای باکس سایه تنها قیمت h-shadow را تطبیق میکنیم.

آموزش کاربردی CSS3

Box-shadow:15px 0 black;

حالا نتیجه به شکل زیر میشود:



حالا میایم بالایی مقدار Blur

وقتی شما باکس سایه را مقدار Blur لحاظ میکنید به اندازه سایز Blur از هر طرف بزرگتر از باکس اصلی میشود.

برای وضوح مطلب به کد زیر دقت کنید:

در دستور بالا مقدار اول h-shadow ، مقدار دوم v-shadow و مقدار سوم آن مربوط به Blur میگردد.

و حال از دستور بالا شکل زیر بدست میآید:



اما اگر خواسته باشید سایه را بیشتر پراکنده نمایید باید مقدار Spread را تطبیق نمایید:

حالا به دستور زیر دقت نمایید:

آموزش کاربردی CSS3

Box-shadow:15px 0 5px 10px black;



در اینجا ۱۰ پیکسل سایه را بیشتر پراکنده کردیم:



خوب حالا قیمت h-shadow را بر میداریم و در نتیجه سایه باکس به شکل زیر میگردد.

آموزش کاربردی CSS3

```
Box-shadow:0 0 5px 10px black;
```



برای درک بهتر خاصیت Blur را نیز بر میداریم و حالا شکل ذیل بدست میآید:

آموزش کاربردی CSS3

```
Box-shadow:0 0 10px black;
```



خوب حالا از Inset استفاده میکنیم البته وقتی از Inset استفاده میکنید سایه را از داخل باکس میدهد:

آموزش کاربردی CSS3

```
Box-shadow:0 0 0 10px black inset;
```



حالا نتیجه دستور بالا به شکل زیر است:



دقیقا همان سایه آمد داخل باکس اصلی قرار گرفت و چون دقیقا در لبه های باکس قرار گرفته درست مشخص نیست ، پس میایم برایش h-shadow و v-shadow هم میدهم.

آموزش کاربردی CSS3

Box-shadow:5px 5px 0 10px black inset;

اینجا ۵ پیکسل از بالای باکس اصلی (عمودی) و ۵ پیکسل هم از سمت چپ باکس اصلی (افقی) سایه ایجاد می کند.



حالا یک نکته ای را باید یاد آور شد که تا اینجا از مقادیر مثبت استفاده کردیم و جابجایی باکس سایه از سمت راست انجام میشد . حالا اگر بخواهیم سایه سمت چپ باکس اصلی ایجاد شود باید چیکار کنیم؟ خوب برای اینکار از مقادیر منفی استفاده می کنیم . به مثال زیر دقت کنید ، ۱۰ پیکسل بصورت افقی و ۱۵ پیکسل بصورت عمودی فاصله ایجاد می کنیم اما این بار به سمت چپ باکس:

آموزش کاربردی CSS3

Box-shadow:-10px -15px 0 0 black;

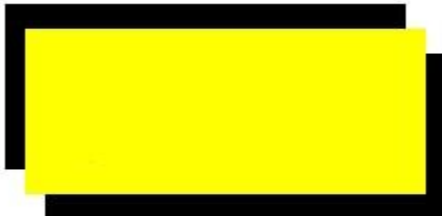


حالا نتیجه به شکل زیر دیده میشود:



با تلفیق دستورات بالا شما میتوانید سایه را بطرف راست و چپ بچرخانید.

به کد زیر دقت کنید:



و این هم لیست مرور گر های که از این خاصیت پشتیبانی میکنند.

IE9+ , Firefox, Chrome, Opera, Safari 5.1.1
Box-shadow: Firefox3.5+ , Opera 9+ , safari+ , Chrome, IE10
-webkit-box-shadow: saf3-4, Chrome, IOS 4.0.2-4.2, Android 2.3+
-mox-box-shadow:firefox 3.5+



مبحث شانزدهم: Box-sizing

در این مبحث از کتاب در مورد box-sizing صحبت میکنیم.

حتماً وقتی کد نویسی میکنید به این مشکل مواجه شده اید که زمانی یک عنصر را Padding یا Border بدهید در عرض و ارتفاع آن نیز تاثیر میگذارد و شما را دچار مشکل میسازد، چون باید عرض و ارتفاع را نسبت به Padding یا Border که قبلاً دادید تغییر بدهید.

خوشبختانه در CSS3 با استفاده از Border-sizing این مشکل را حل کرده میتوانیم.

برای درک بهتر موضوع به مثال زیر دقت کنید:

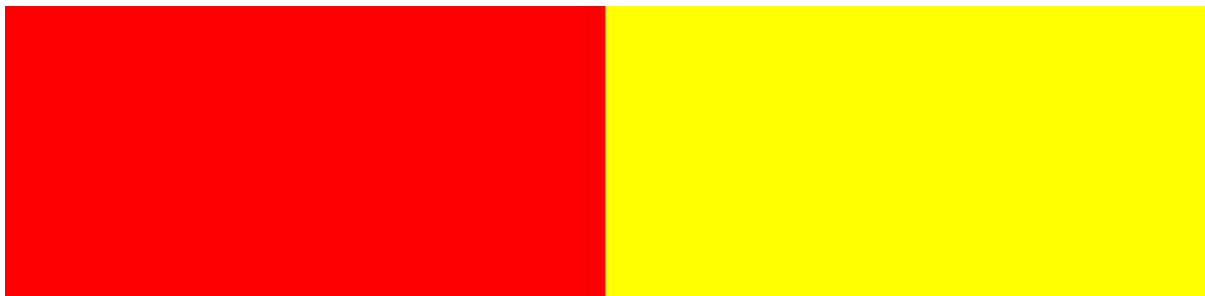
کد HTML:

آموزش کاربردی CSS3
<div class="main">
<div class="box"></div>
</div>

کد CSS:

آموزش کاربردی CSS3
.main{width:600px; height:150px; background-color:yellow;}
.box{width:50%; height:150px; background-color:red;}

توسط کد بالا ما دو باکس (یک باکس اصلی با رنگ زرد) و (یک باکس فرزند سرخ) ایجاد کردیم و نتیجه به شکل زیر میباشد:



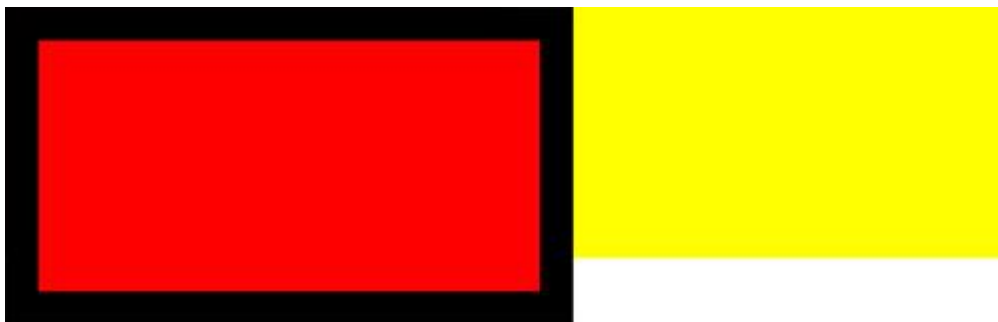
در کد بالا گفتیم که عرض باکس فرزند یا داخلی ۵۰ درصد باکس اصلی باشد، این دستور به درستی انجام شد و هیچ مشکلی ندارد.



خوب حالا ما به باکس داخلی (فرزند) border می‌دهیم.

آموزش کاربردی CSS3
.box{width:50%; height:150px;
Background-color:red;
Border:20px solid black;
}

حالا وقتی border دادیم به عرض و ارتفاع باکس داخلی (فرزند) اضافه می‌شود:



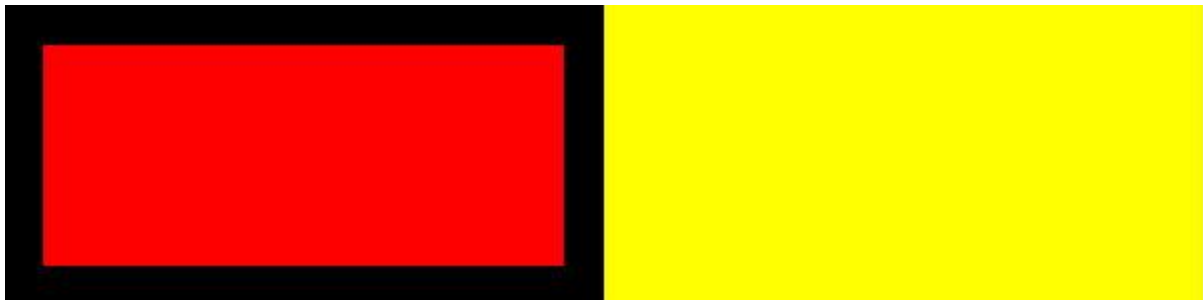
قبلاً برای درست کردن این مشکل باید شما اندازه باکس فرزند را کوچک نموده و با اضافه نمودن Border اندازه آن درست می‌گردید اما خوشبختانه حالا شما می‌توانید با قابلیت جدید Box-sizing در CSS3 می‌توانید به راحتی این کار را انجام دهید.

به کد CSS زیر دقت کنید:

آموزش کاربردی CSS3
.box{width:50%; height:150px;
Background-color:red;
Border:20px solid black;
Box-sizing:border-box;
-webkit-box-sizing:border-box;
-moz-box-sizing:border-box;
}



حالا از دستور بالا چنین نتیجه بدست میآید:



خوب شاید برای تان استفاده این دستور در هر خاصیت (Border و Padding) مشکل باشد و وقت تانرا ضایع کند. برای حل این مشکل کافیهست که در ابتدای فایل CSS تان کد زیر را اضافه نمایید تا بالایی تمام عناصر اعمال گردد.

آموزش کاربردی CSS3
*{
Box-sizing:border-box;
-webkit-box-sizing:border-box;
-moz-box-sizing:border-box;
}

شما با استفاده از Box-sizing میتوانید بالایی Padding نیز تطبیق کنید.

یادداشت: توجه داشته باشید که Box-sizing تنها بالایی عناصر Border و Padding کار میدهد، چون تنها در این عناصر مشکل ایجاد میشود.

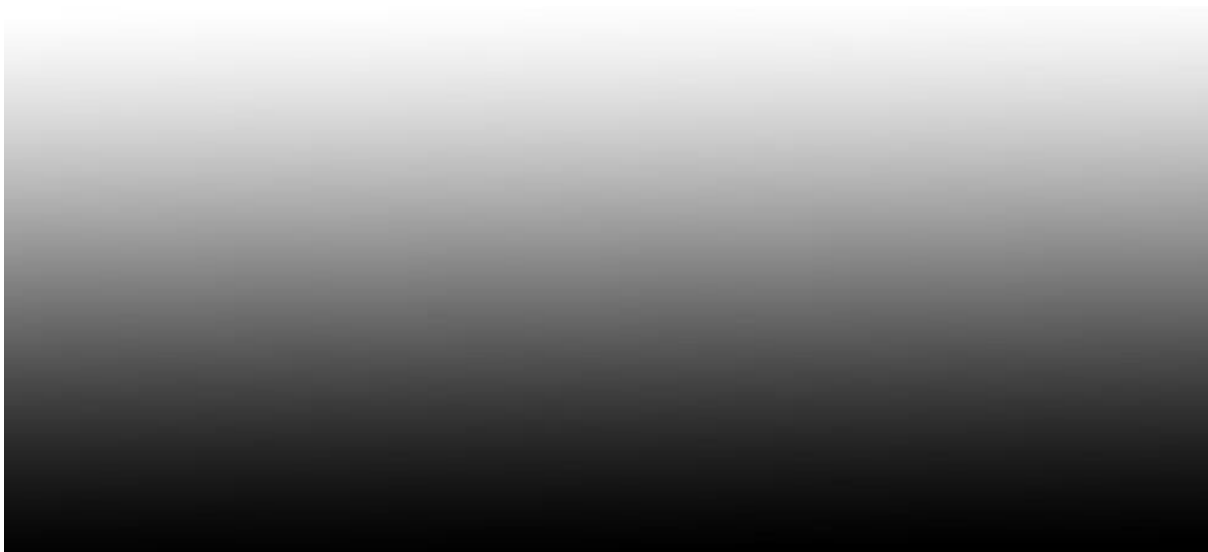


مبحث هفدهم: Gradient

در این مبحث در مورد ساخت Gradient صحبت میکنیم.

به صورت کلی Gradient به حالتی اطلاق میگردد که بسته به جهتی که انتخاب میکنیم به تدریج یک رنگ به درنگ دیگری تبدیل میشود.

در تصویر زیر ما یک گرادیانت عمودی ایجاد کردیم به طوری که این گرادیانت به تدریج از سفید به سمت تیره گی تغییر پیدا میکند.



برای واضح شدن موضوع به مثال زیر توجه نمایید:

دستور HTML:

آموزش کاربردی CSS3
<code><div id="box"></div></code>

دستور CSS:

آموزش کاربردی CSS3
<code>#box {</code>
<code>Width:300px;</code>
<code>Height:200px;</code>
<code>Background: linear-gradient(top, #ffde57 0%, #ffdeff 100%);</code>



نتیجه:



نکته: دستور بالا در تمام مرورگرها قابل اجرا نیست از این سبب باید به این شکل از دستور استفاده نمایید:

آموزش کاربردی CSS3
#box {
Width:300px;
Height:200px;
Background: linear-gradient(top, #ffde57 0%, #ffdeff 100%);
background:-moz-linear-gradient(top, #ffde57 0%, #ffdeff 100%);
background:-webkit-linear-gradient(top, #ffde57 0%, #ffdeff 100%);
background:-o-linear-gradient(top, #ffde57 0%, #ffdeff 100%);
background:-ms-linear-gradient(top, #ffde57 0%, #ffdeff 100%);
background:-moz-linear-gradient(top, #ffde57 0%, #ffdeff 100%);
}

همانطور که میبینید دستور مربوط به گرادیانت را چندین بار دیگر تکرار کردیم که در دستور اول کلمه -moz- را اضافه کردیم که این دستور برای مرورگر mozilla استفاده میشود و در دستور دوم کلمه -webkit- را ضمیمه کردیم که البته این کلمه مرتبط با مرورگرهای سافاری و گوگل کروم است در خط بعد کلمه -o- را ضمیمه کردیم که این کلمه مرتبط با مرورگرهای اپرا است. خط بعد کلمه -ms- را افزودیم که این کلمه مربوط به مرورگر اینترنت اکسپلورر است.



مبحث هجدهم: Transform

با استفاده از خاصیت Transform که تازه وارد دنیای وب شده میتوانید افکت های زیبای ایجاد کنید. البته قبلاً برای انجام این امکان ضرورت به جی کیوری داشتید اما حالا با استفاده از CSS3 میتوانید به سادگی دیزاین را تطبیق کنید.

با Transform شما میتوانید عناصر تانرا بچرخانید ، بزرگ یا کوچک نمایید و امثال اینها.

قاعده دستوری خاصیت Transform به شکل زیر است:

آموزش کاربردی CSS3
Transform: effect()

خاصیت Transform مقادیر زیر را دارا میباشد:

Translate

با استفاده از این مقدار شما میتوانید عنصر مورد نظر تانرا بصورت افقی و عمودی جابجا نمایید. این مقدار تنها دو قیمت را میپذیرد که یک قیمت عنصر را به شکل افقی جابجا میکند و دیگر قیمت عنصر را به شکل عمودی جابجا میکند.

قیمت های مقدار Translate:

TranslateX(valueX): این قیمت عنصر را به جهت افقی جابجا میکند.

TranslateY(valueY): این قیمت عنصر را به جهت عمودی جابجا میکند.

آموزش کاربردی CSS3
Transform : translate(valueX,valueY)
یعنی
Transform : translate (50px, 70px)

حالا بعد از استفاده از دستور بالا عنصر ما از لحاظ افقی در قسمت 60px و از لحاظ عمودی در قسمت 80px قرار میگیرد.



Rotate()

با استفاده از این مقدار خاصیت Transform شما میتوانید عنصر مورد نظر تانرا بچرخانید. البته واحد که با قیمت این مقدار استفاده میشود deg میباشد که کلمه deg نمایندگی از درجه میکند. برای نحوه استفاده به مثال زیر توجه نمایید:

آموزش کاربردی CSS3
Transform : rotate (deg);
یعنی
Transform : (90deg)



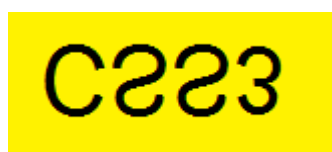
RotateX(180deg)

این قیمت عنصر را بصورت عمودی به اندازه ۱۸۰ درجه میچرخاند و عنصر را به حالت عمودی کاملاً سر چپه میکند.

مثال:

آموزش کاربردی CSS3
Transform : rotatex(180deg);

حالا نتیجه دستور بالا به شکل زیر است:



RotateY(180)

این قیمت عنصر را بصورت افقی به اندازه ۱۸۰ درجه میچرخاند و عنصر را به حالت افقی کاملاً سر چپه میکند.

مثال:

آموزش کاربردی CSS3

Transform : rotatey(180deg);

حالا نتیجه دستور بالا به شکل زیر است:



Scale()

با استفاده از این مقدار خاصیت Transform شما میتوانید ابعاد عنصر تانرا بصورت عمودی و افقی افزایش دهید.

این مقدار دارای دو قیمت میباشد ValueX و ValueY

مثال: در این مثال ما میخواهیم ابعاد تصویر را از لحاظ افقی ۲ برابر و از لحاظ عمودی ۳ برابر بزرگ نمایم.

آموزش کاربردی CSS3

Transform : scale (2 , 3);



Before



After



Skew()

با استفاده از این مقدار خاصیت Transform شما میتوانید عنصر مورد نظر تانرا به دو حالت (افقی و عمودی) بصورت ۳ بعدی بچرخانید.

اگر قیمت این درجه مثبت باشد عنصر به سمت راست و اگر قیمت منفی باشد عنصر به سمت چپ میچرخد.

مثال:

آموزش کاربردی CSS3

```
Transform : skew (10deg , 10deg);
```

نتیجه:



شما نیز میتوانید به صورت جداگانه عنصر را بچرخانید یعنی میتوانید به تنهایی به صورت افقی بچرخانید و یا میتوانید به صورت عمودی بچرخانید.

SkewX()

این قیمت عنصر را در حالت افقی میچرخاند.

SkewY()

این قیمت عنصر را در حالت عمودی میچرخاند.



مبحث نهم: Transition

خاصیت Transition از جمله خاصیت های می باشد که تازه وارد دنیای وب گردیده و این خاصیت عامل اصلی در افکت های CSS3 می باشد و بدون این خاصیت شما نمیتوانید هیچ افکتی را تطبیق نمایید.

قاعده دستوری:

آموزش کاربردی CSS3

Transition: Property | duration | timing-function | delay;

Transform خاصیت های زیر را دارا می باشد:

Transition-property

با استفاده از این خاصیت میتوانید تعیین کنید که انیمیشن روی کدام خاصیت اختصاص داده شده به عنصر اعمال شود ، مثلا بالای خاصیت روی width عنصر اعمال شود یا روی height عنصر یا color عنصر و ... در صورتی که مقدار این خاصیت را برابر با all قرار بدهید تمامی خاصیت ها و مقادیر و استایل یک عنصر را تحت تاثیر انیمیشن قرار میدهد و انیمیشن را روی تمامی خاصیت ها و مقادیر عنصر اجرا می کند.

Transition-duration

این خاصیت مدت زمان اجرای یک انیمیشن را مشخص می کند که بر حسب ثانیه و میلی ثانیه مشخص میشود.

Transition-timing-function

این خاصیت شامل مقادیر زیر میشود و سرعت اجرای افکت را تعیین می کند:

linear: در صورت استفاده از این مقدار انیمیشن از ابتدا تا انتها با سرعت یکسان حرکت می کند و انجام میشود.

Ease: این مقدار، مقدار پیشفرض این خاصیت است. یعنی انیمیشن با سرعت نسبتا کم شروع میشود و سریع سرعتش افزایش پیدا میکند و وقتی به انتها میرسد دوباره سرعتش کم میشود.

ease-in: انیمیشن با سرعت کم شروع میشود و با سرعت زیاد به اتمام میرسد.



ease-out: در صورت استفاده از این مقدار انیمیشن با سرعت زیاد شروع میشود و با سرعت کم به اتمام میرسد.

ease-in-out: در صورت استفاده از این مقدار انیمیشن با سرعت کم شروع میشود بین انیمیشن یکم سرعت افزایش پیدا می کند و با سرعت کم به اتمام میرسد.

cubic-bezier(n,n,n,n): با استفاده از مقدار شما میتوانید مقادیر دلخواه تانرا برای این خاصیت تعریف کنید. مقادیر امکانپذیر ۰ تا ۱ است. میتوانید برای تنظیم مقادیر به صورت دقیقتر از این سایت <http://cubic-bezier.com> استفاده کنید.

transition-delay: در صورت استفاده از این خاصیت شما میتوانید مکثی را قبل از شروع افکت تعیین کنید، مثلاً ۲ ثانیه بگذرد و بعد افکت اجرا شود.

برای درک بهتر از موضوع به مثال زیر دقت کنید.

آموزش کاربردی CSS3	
H1{	
Width:100px;	
Color:white; background-color:black;	
-webkit-transition: all 1s ease 0s;	
-moz-transition: all 1s ease 0s;	
Transition: all 1s ease 0s;}	
H1:hover{background-color: brown}	

در مثال بالا یک عنصر h1 در نظر گرفتیم و به این عنصر رنگ قلم، رنگ پس زمینه و ۱۰۰ پیکسل عرض اختصاص دادیم. همچنین تعریف کردیم که وقتی ماوس روی این عنصر قرار گرفت رنگ پس زمینه از مشکی به قهوه ای تغییر کند.



مبحث بیستم : Cursor

در این مبحث کوتاه و دلچسب در مورد Cursor صحبت میکنیم.

با استفاده این خاصیت میتوانید حالت و شکل ماوس تانرا تغییر دهید. مثلاً وقتی Pointer تان بالای یک عنصر مشخص آمد شکل آن به شکل دیگر تبدیل گردد. بر علاوه استفاده از مقادیر، شما نیز میتوانید از عکس های دلخواه تان استفاده نمایید. بیاد داشته باشید آدرس این عکس دلخواه نوشته گردد. البته این عکس در مرورگر IE حتماً باید پیوند cur داشته باشد.

شما نمیتوانید هر تصویر را اضافه نماید الی که ابعاد تصویر شما 32*32 باشد.

آموزش کاربردی CSS3

```
A:link {
    Cursor: url (hyper.cur) , pointer;
}
```

در مثال بالا اولاً ما عکس دلخواه خود را مشخص کردیم و pointer که بعداً نوشته کردیم به این علت میباشد که در صورت که مرورگر مسیر را به درستی پیدا نکرد مقدار دوم یعنی pointer را لحاظ کند.

در زیر لیست از مقادیر را مشاهده میکنید که میتوانید از آنها استفاده نماید.

CSS Custom Cursors

I auto	↕ move	☞ no-drop	↔ col-resize
⬮ all-scroll	☞ pointer	⊘ not-allowed	↕ row-resize
+ crosshair	⌚ progress	↔ e-resize	↗ ne-resize
☞ default	I text	↑ n-resize	↖ nw-resize
☞ help	↕ vertical-text	↓ s-resize	↗ se-resize
I inherit	⌚ wait	↔ w-resize	↖ sw-resize



فصل چهارم

مباحث فصل پنجم

✓ قانون @Media در CSS چیست؟

✓ انواع Media

✓ نحوه استفاده Media



قانون @media در CSS چیست؟

با استفاده از این قانون شما قادرید که برای هر انواع آله استایل های متفاوتی تعریف نمایید. بطور مثال در یک سند HTML برخی از قسمت های صفحه بمنظور پرنٹ گرفتن میباشد شما باید برای آن قسمت استایل که مربوط به Print میشود را تطبیق دهید و یا بعضی از قسمت ها از جمله Navigation سایت ضرورت به print ندارد و در این صورت میتوانید یک استایل دیگر را تعریف کنید.

برای هدف قرار دادن مدیا های مختلف باید از صفت Media در تگ Link یا در تگ Style و یا هم میتوانید در درون قواعد نوشته شده CSS از قاعده @media برای نشان کردن مدیا های مختلف استفاده نمایید.

انواع Media

در زیر شما لیست از انواع Media ها را میبینید که شما میتوانید نظر به ضرورت از آنها استفاده نمایید.

انواع Media	توضیحات
All	این مقدار مخصوص تمامی آله ها میباشد.
Aural	این نوع مدیا مخصوص آله های صوتی میباشد.
Barille	این نوع مدیا مخصوص به آله های نابینایان میشود.
Embossed	این نوع مدیا مخصوص برای چاپ خط های Braille استفاده میشود.
Handheld	این نوع مدیا برای آله های قابل حمل از قبیل Tablet استفاده میشود.
Print	این نوع مدیا برای ارایه استایل به Printer استفاده میشود. در مرورگرها میتوان از قسمت Print Preview و یا print برای مشاهده یک صفحه در حالت Print استفاده کنید.
Projection	این نوع مدیا برای ویدیو پروژکتور ها استفاده میشود.
Screen	این نوع مدیا مخصوص نمایش کامپیوتر میباشد.
TV	این نوع مدیا مخصوص نمایش در تلویزیون ها میباشد.

نکته ۱: مقدار پیش فرض Media مقدار all میباشد که در تمام آله ها یکسان نمایش میدهد.

نکته ۲: در مرورگر های تنها مقادیر All, Screen, Print بصورت گسترده پشتیبانی میشود.



نحوه استفاده از Media

قسمی که قبلاً گفتیم برای هدف قرار دادن مدیا های مختلف باید از صفت Media در تگ Link یا در تگ Style و یا هم میتوانید در درون قواعد نوشته شده CSS از قاعده @media برای نشان کردن مدیا های مختلف استفاده نمایید.

حال میرویم بصورت نمونه میخواهیم به یک قسمت از صفحه را که برای پرنت استفاده میشود یک استایل مشخص تعریف نمایم.

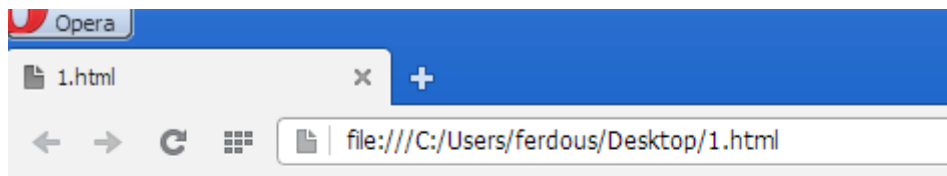
برای درک بهتر در زیر از دستور HTML و CSS یکجا استفاده میکنیم.

آموزش کاربردی CSS3
<head>
<style>
@media screen {
P{
Font-family: verdana, sans-serif;
Font-size:17 px;
Color:red;
}
}
@media print {
P,div {
Font-family:Georgia, serif;
Font-size:14px;
Color: blue;
}
}
</style>
</head>
<body>
<p> Amozesh CSS3 </p>
<div> This is a test text that used to print </div>
</body>



حالا نتیجه دستور بالا در مرورگر Opera به شکل زیر است.

نتیجه زیر دیزاین متن بدون استفاده از قانون @media print میباشد.

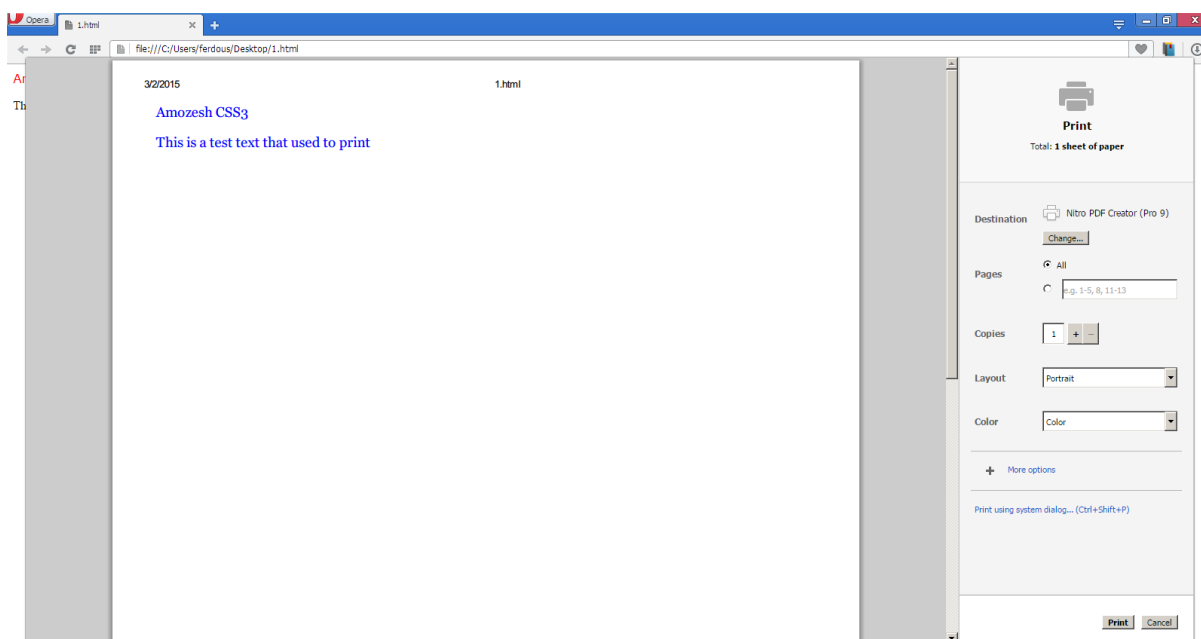


Amozesh CSS3

This is a test text that used to print

نتیجه زیر دیزاین متن بعد از استفاده از قانون @media print میباشد.

البته شما میتوانید با استفاده از کلید های ctrl+p صفحه چاپ صفحه را تماشا نمایید.



فصل ششم

مباحث فصل ششم

- ✓ گروه بندی عناصر صفحه با استفاده از تگ های Div و Span
- ✓ آشنایی با نحوه طرح بندی و طراحی قالب سایت
- ✓ طرح بندی صفحات (Float base)
- ✓ ایجاد قالب های شناور دو ستونه
- ✓ ایجاد قالب های شناور سه ستونه



گروه بندی عناصر صفحه با استفاده از تگ های Span و Div

سابق برای گروه بندی صفحات از تیبل استفاده میکردند، یعنی با ادغام نمودن چندین سطر و ستون صفحات را تقسیم بندی میکردند، که البته این نوع گروه بندی دارای نواقص زیادی بود از جمله درست لود نشدن سایت، امکانات کم در جابجایی عناصر و در ضمن برای SEO سایت نیز خوب نبود اما حالا شما میتوانید از تگ های Div و Span برای گروه بندی سایر تگ ها به منظور ایجاد بخش ها و زیر بخش های داخل یک صفحه HTML استفاده نمایید. از این دو عنصر معمولاً به همراه CSS برای اینکه به گروهی از عناصر Style خاصی نسبت دهیم استفاده میشود.

برای مثال فرض کنید می خواهید تعدادی عنصر در بخشی از صفحه همیشه از کلاس خاصی پیروی کنند. احتمالاً کلاس مورد نظر به روش زیر به تک تک آن ها نسبت می دهید.

آموزش کاربردی CSS3
<h3 class="footer"> This is a header of footer</h3>
<p class="footer"> This is a paragraph </p>
<h2 class="footer"> This is a paragraph <h2>
<strong class="footer"> This text is strong

اما شما میتوانید به جای این کار آنها را با استفاده از تگ div داخل یک گروه قرار داده و کلاس مورد نظر را فقط به تگ div اعمال کنیم.

آموزش کاربردی CSS3
<div class="footer">
<h3 This is a header of footer</h3>
<p> This is a paragraph </p>
<h2> This is a paragraph <h2>
 This text is strong
</div>

یکی از ویژگی های مهم تگ Div این است که از آن برای گروه بندی عناصر Block-level و inline میتوان استفاده نمود.

از طرف دیگر ما میتوانیم از تگ Span فقط برای گروه بندی از نوع عناصر inline استفاده کنیم. بنابراین اگر میخواهید حد اقل به یک جمله و یا قسمتی از یک پاراگراف style خاصی نسبت دهید بهتر است از span استفاده کنید. مثال:



آموزش کاربرد CSS3

```
<p>He has <span class="blue">light blue</span> eyes.</p>
```

یکی دیگر از تفاوت های تگ های span و Div در این است که تگ div خود جزء block-level و Span جزء عناصر Inline-level است.

آشنایی با نحوه طرح بندی و طراحی قالب سایت

در این مبحث قصد داریم به بررسی روش های طرح بندی (Layout) که این روز ها با عنوان طراحی قالب یاد میشود آشنا شویم.

راه های گوناگون برای این امر وجود دارد که استفاده از عنصر Table یکی از آنها است. استفاده از جدول برای طرح بندی کلی صفحات به دلیل مشکلاتی همچون سخت بودن کار با ها (مخصوصاً هنگام کار با جداول تو در تو)، سرعت با گذاری پایین و عدم کنترل دقیق چگونگی نمایش، دیگر این روز ها پیشنهاد نمیشود و به جای آن استفاده از ترکیب عناصر ساده تری همچون div به همراه قواعد CSS برای کنترل خصوصیات آن ها توصیه میشود.

در این فصل سعی میکنیم به اختصار به نحوه طراحی قالب های دو ستونه و سه ستونه پرداخته شود.

طرح بندی صفحات (Float-based)

راه های گوناگون برای طرح بندی صفحات از طریق CSS وجود دارد که از آن میتوان به استفاده از روش تعیین مکان مطلق عناصر با کمک قواعد Poosition و Margin اشاره نمود. اما ما از روش دیگر که بسیار ساده تر و البته پر کاربرد تر است استفاده خواهیم کرد. و آن استفاده از قواعدی همچون Float و Clear برای تعیین موقعیت عناصر است. در این روش ما به سادگی عرض و ارتفاع عناصر را تعیین کرده و سپس مشخص میکنیم که در چپ یا راست عناصر در برگیرنده خود قرار بگیرند. (در واقع ما مشخص میکنیم که عناصر در کدام جهت شناور شوند.)



ایجاد طراح (قالب) های شناور دو ستونه

برای ایجاد یک طرح دو ستونه ساده باید کار را از یک صفحه HTML شروع کنیم. در صفحه HTML مثالی که در ادامه بررسی خواهیم کرد بخش های اصلی زیر وجود دارد.

- **Content:** بخش محتوا که قرار است محتوای اصلی سایت در آن قرار بگیرد.
- **MainNav:** بخش پیمایش اصلی (main navigation) که قرار است در سمت چپ قرار گرفته و حاوی لینک ها و عناصر برای دسترسی ساده تر کاربر به صفحات دیگر سایت باشد.

- **Footer:** بخش پانویس که قرار است اطلاعاتی اضافی از سایت را در آن قرار دهیم.

هر کدام از این عناصر را با یک عنصر Div با نام wrapper پوشانده شده اند.

به کد زیر دقت کنید:

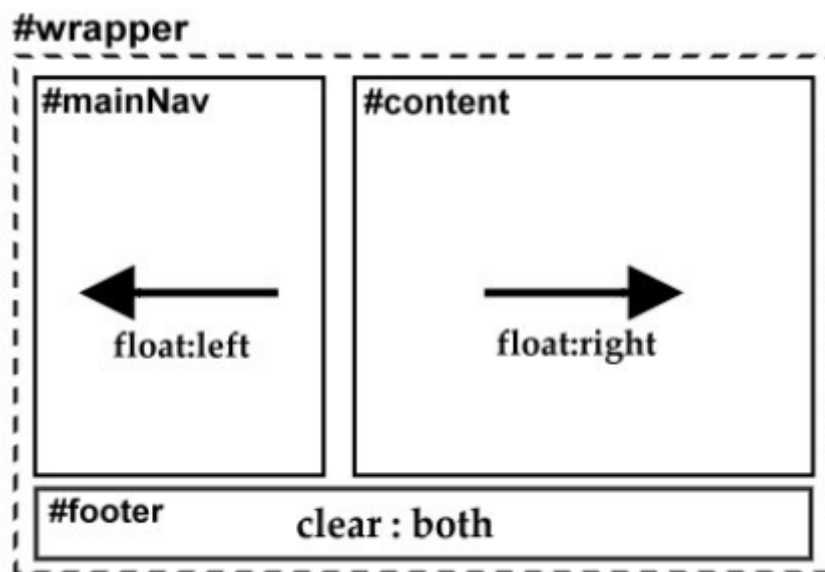
آموزش کاربردی CSS3
<div id="wrapper">
<div id="content">
.....
</div>
<div id="mainNav">
.....
</div>
<div id="footer">
.....
</div>
</div>

در این مثال قرار است Content به سمت راست و mainNav به سمت چپ بروند. به این نکته دقت کنید چون بخش Content ما حاوی اطلاعات مهمتر است کد آن نیز باید قبل از بخش mainNav بیاید تا دسترسی و استفاده از آن نیز آسان تر باشد.

حال به سراغ تعیین قواعد عناصر فوق در CSS میرویم. ما از Selector نام برای تعریف قواعد برای هر بخش استفاده میکنیم. ابتدا بصورت زیر برای هر یک از بخش های Content و mainNav عرض تعیین کنیم:

آموزش کاربردی CSS3
#content {
Width: 520px;
Float: right;
}
#mainNav {
Width: 180px;
Float: left;
}

همانطور که میبینید چون Content باید در سمت راست قرار بگیرد قاعده Float آن را برابر Right و برای mainNav آن برابر left قرار میدهیم. به تصویر زیر نگاه کنید:

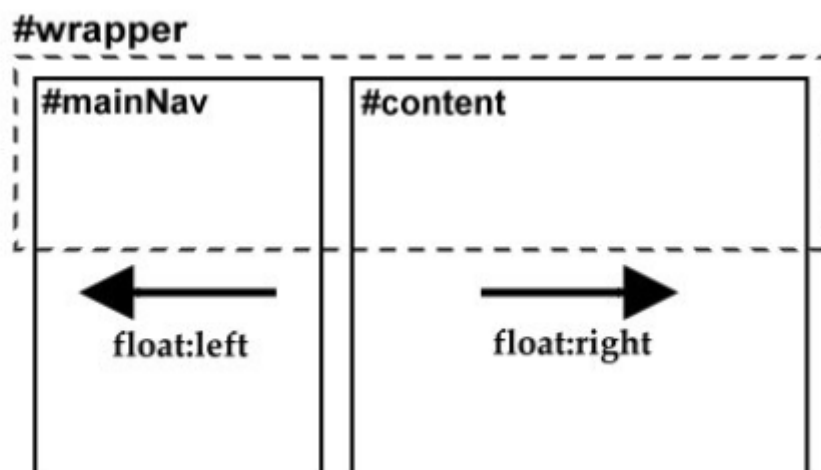


نکته بسیار مهم:

از آنجایی که عناصر از نوع شناور (یعنی عناصر که قاعده Float آنها برابر Right یا Left باشد) معمولاً از جریان عادی صفحه خارج شده و بر روی عناصر دیگر نمیتوانند تاثیر بگذارند احتمالاً بصورت کامل در عنصر در بر گیرنده خود (در مثال ما منظور از عنصر در بر گیرنده



Wrapper است) قرار نگرفته و آن چیزی که انتظار داریم انجام نمیشود. در واقع طرح ما به صورت زیر خواهد آمد.



برای حل این مشکل دو راه حل وجود دارد:

۱- یا اینکه عنصر در برگرنده (wrapper) خود حالت float داشته باشد. (یعنی به چپ یا راست متمایل شود) این مورد را امتحان کنید!

۲- یا اینکه عنصر با قاعده `Clear : both` را در آن قرار دهیم.

ما از روش دوم استفاده خواهیم کرد. برای این کار از بخش Footer استفاده میکنیم و قاعده `Clear` آن برابر `Both` قرار دهیم. با این کار مطمئن میشویم که Footer دقیقاً در پایین دو بخش قبلی قرار خواهد کرد و چون خود، حالت `Float` ندارد در نتیجه به عنصر `Wrapper` ارتفاع خواهد داد و به این شکل مشکل قبلی اصلاح خواهد شد:

آموزش کاربردی CSS3
<code>#footer {</code>
<code>Clear:both;</code>
<code>}</code>



ایجاد طرح های سه ستونه شناور:

ساختار صفحه HTML مورد استفاده برای ایجاد طرح های سه ستونه بسیار شبیه به ساختار طرح های دو ستونه است، تنها تفاوت، اضافه شدن دو عنصر Div جدید در بخش Content است: یکی با نام MainContent برای محتوای اصلی و دیگری با نام SecondaryContent برای محتوای ستون سوم که قرار است در سمت راست نمایش داده شود:

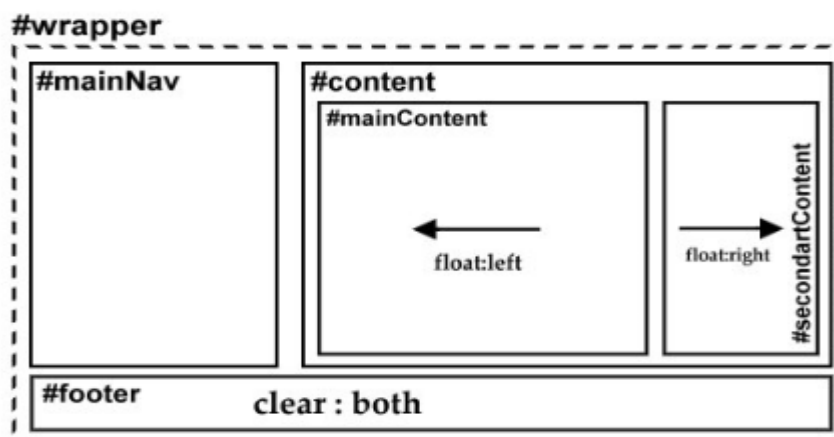
آموزش کاربردی CSS3
<div id="content">
<div id="mainContent">
.....
</div>
<div id="secondaryContent">
.....
</div>

دقیقاً با همان روش که قبلاً اشاره شد میتوان عنصر mainContent را به چپ و عنصر secondaryContent را به سمت راست عنصر Content هدایت کرد:

آموزش کاربردی CSS3
#mainContent {
Width: 320px;
Float: left;
}
#secondaryContent {
Width: 180px;
Float: right;
}

در واقع با این کار ما عنصر Content را به دو قسمت تقسیم کردیم که با عنصر mainNav جمعاً تشکیل سه ستون میدهند. تصویر زیر گویای موضوع خواهد بود.





دقترمانیز در اینجا به پایان رسید!



منابع

- www.w3schools.com
- www.lroschool.com
- The Ultimate CSS Reference

