

دوره آموزش HTML

تمام صفحات وب با نوعی از HTML ساخته می شوند. لذا برای ایجاد وب سایت های پیشرفته یادگیری HTML ضروری می باشد. در این دوره آموزشی سعی شده است که با به کارگیری مثال های فراوان و دوری از هر گونه اضافه گویی در سریعترین زمان ممکن توانایی ایجاد صفحات وب بدون نیاز به برنامه ای کمکی در شما ایجاد گردد.

مقدمه ای بر وب و HTML

وب گسترده جهانی

وب گسترده جهانی (www , World Wide Web) که معمولا فقط وب گفته می شود؛ شبکه ای است که تمام رایانه های متصل به آن را در سراسر جهان به هم وصل می کند؛ و تمام رایانه ها در وب می توانند با هم ارتباط داشته باشند.

www چگونه کار می کند ؟

اطلاعات وب در فایل هایی به نام **Web Page** قرار می گیرند که آنها نیز در رایانه هایی به نام **Web Server** قرار دارند و رایانه ها برای دیدن صفحات وب از استاندارد ارتباطی به نام **HTTP (Hyper Text Transfer Protocol)** استفاده می کنند. رایانه های متصل به وب اطلاعات موجود در **Web Page** ها را از طریق برنامه هایی با نام مرورگر وب (**Web Browser**) نشان می دهند. معمولترین مرورگرها که مورد استفاده قرار می گیرند **Internet Explorer** و **Netscape Navigator** می باشند.

چگونه مرورگرها صفحات وب را نشان می دهند ؟

همه صفحات وب شامل دستورالعمل هایی برای نمایش توسط مرورگرها هستند، و مرورگرها با خواندن این دستورالعمل ها صفحات را نشان می دهند. دستورالعمل های نمایش صفحات وب، برچسب های **HTML (Hyper Text Markup Language)** نامیده می شوند مانند:

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

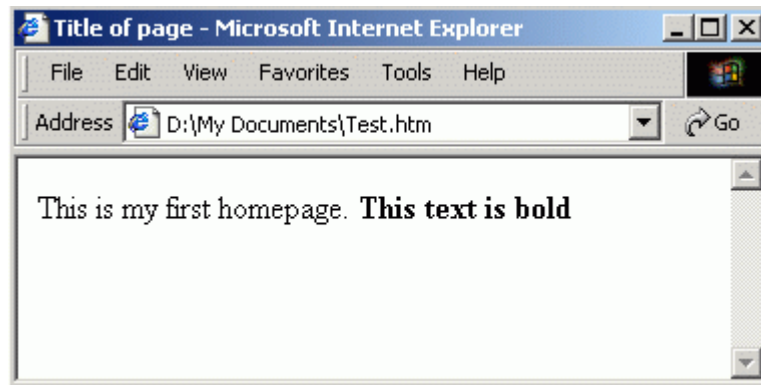
یک فایل HTML چیست ؟

یک فایل **HTML** فایل متنی ساده ای شامل برچسب ها (**Tag**) می باشد؛ که بهتر است پسوند **html** یا **htm** داشته باشد و می تواند توسط یک ویرایشگر ساده متن مانند **Notepad** ساخته شود. برای بهتر روشن شدن مطلب بهتر است یک فایل **HTML** بسازیم. در **Windows** برنامه **Notepad** را اجرا کنید و متن زیر را بنویسید :

```
<html>
<head>
<title>Title of page</title>
</head>
<body>
This is my first homepage. <b>This text is bold</b>
</body>
</html>
```

Open یا **Open** ذخیره کنید. حال مرورگر اینترنت خود را باز کرده و گزینه **Test.htm** این فایل را با نام "Page از منوی **File** مرورگر انتخاب کنید؛ یک جعبه محاوره ای باز خواهد شد، **Browse** یا **Choose** فایل را انتخاب کنید و فایل "**Test.htm**" که هم اکنون ساخته اید را بیابید، انتخابش کنید و روی **Open** کلیک کنید. حال شما یک آدرس مانند:

"**D:\My Documents\Test.htm**" را در جعبه محاوره ای خواهید دید. روی **OK** کلیک کنید، مرورگر صفحه ساخته شده توسط شما را نشان می دهد.



بدین ترتیب شما اولین صفحه **HTML** خود را بدون استفاده از برنامه های طراحی وب و فقط با یک ویرایشگر متن ساده ساختید و نتیجه آن را نیز توسط مرورگر دیدید. حال بیایید کمی درباره صفحه ای که ساخته اید صحبت کنیم :

اولین برچسب در فایل **HTML** شما **<html>** است. این برچسب به مرورگر می گوید که این شروع یک فایل **HTML** است. آخرین برچسب در فایل شما **</html>** است. این برچسب به مرورگر می گوید که این پایان یک فایل **HTML** است. نوشته های بین برچسب **<head>** و برچسب **</head>** اطلاعات سربرگ می باشند و در پنجره مرورگر نمایش داده نمی شوند. متن بین برچسب **<title>** و برچسب **</title>** عنوان فایل می باشد و در نوار عنوان مرورگر نمایش داده خواهد شد. متن بین **<body>** و **</body>** توسط مرورگر نشان داده خواهد شد. همچنین متن بین **** و **** به صورت پررنگ نمایش داده می شود.

نکته ای درباره ویرایشگرهای **HTML** :

شما می توانید صفحات **HTML** را با ویرایشگرهای **WYSIWYG** (What You See Is What You Get) مانند: **FrontPage** و **Claris Home Page** و **Adobe PageMill** به جای نوشتن برچسبهای مورد نیاز در یک فایل متنی ساده؛ ویرایش نمایید. اما اگر می خواهید یک طراح حرفه ای صفحات وب شوید بهتر است از یک ویرایشگر ساده متن استفاده کنید تا **HTML** را به خوبی درک کنید و در مباحث پیشرفته به درستی و با تسلط از آن استفاده کنید.

برچسبهای **HTML**

برچسبها برای مشخص کردن عناصر فایل **HTML** استفاده می شوند و توسط دو حرف **<** و **>** احاطه می شوند. برچسبها معمولاً به صورت جفت استفاده می شوند مانند **** و ****؛ برچسب اول، برچسب ابتدایی و برچسب دوم، برچسب انتهایی نامیده می شود و متن بین آنها مندرجات عنصر می باشد. برچسبها به بزرگی یا کوچکی حروف حساس نیستند مثلاً **** همانند **** می باشد اما ما در این کتاب از حروف کوچک استفاده می کنیم. اگر می خواهید خود را برای استفاده از نسل جدید **HTML** آماده کنید شما نیز بهتر است از حروف کوچک استفاده کنید. ضمناً این پیشنهادی است که **W3C** (World Wide Web Consortium) برای استفاده از **XHTML** می کند.

عناصر HTML

برای روشن شدن مفهوم عنصر به دو مثال زیر بسنده می کنیم:
خط زیر یک عنصر است:

```
<b>This text is bold</b>
```

این عنصر با برچسب ابتدایی **** شروع شده و با برچسب انتهایی **** پایان می پذیرد و **This text is bold** نیز مندرجات عنصر است.

خطوط زیر نیز مجموعاً یک عنصر می باشد که شروع آن با برچسب ابتدایی **<body>** و پایان آن با برچسب انتهایی **</body>** می باشد.

```
<body>  
This is my first homepage. <b>This text is bold</b>  
</body>
```

مشخصه برچسبها

برچسبها می توانند دارای مشخصه (Attribute) باشند. مشخصه ها می توانند اطلاعات بیشتری در باره عناصر HTML صفحه شما تدارک ببینند.

برچسب **<body>** عنصر بدنه صفحه شما را تعریف می کند. با اضافه کردن مشخصه **bgcolor** شما می توانید به مرورگر بگویید که رنگ پس زمینه شما مثلاً قرمز باشد؛ مانند:

```
<body bgcolor="red">
```

برچسب **<table>** یک جدول ایجاد می کند. با اضافه کردن مشخصه **border** می توانید به مرورگر بگویید که مثلاً جدول شما حاشیه ندارد؛ مانند:

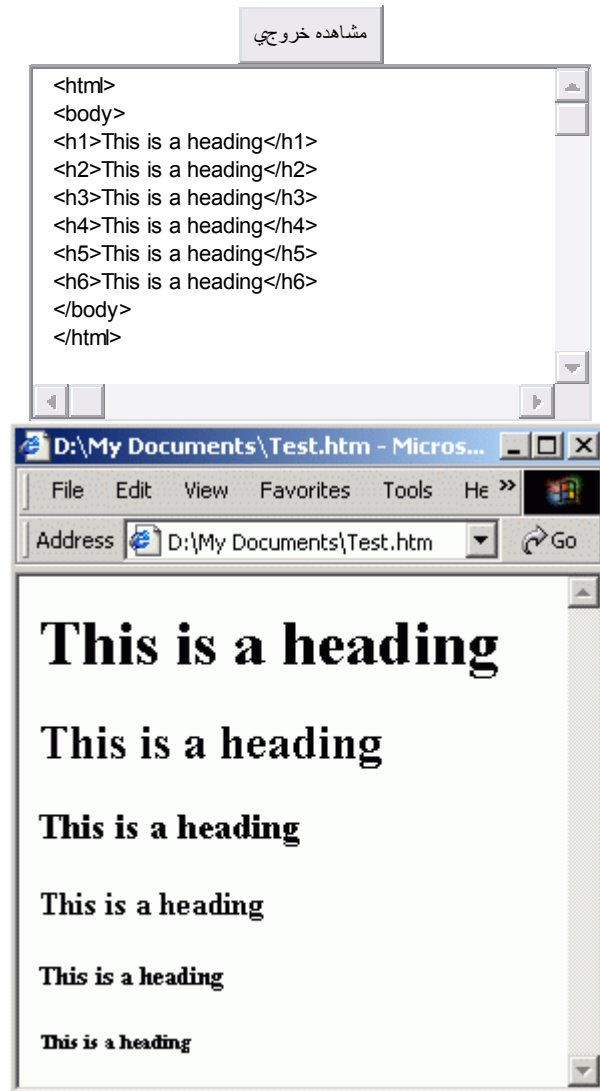
```
<table border="0">
```

مشخصه ها همواره به شکل جفت نام و مقدار به کار می روند: **name="value"** ؛ و همواره به برچسب ابتدایی اضافه می شوند. در ضمن بهتر است که مقدار مشخصه ها بین علامتهای نقل قول قرار گیرند.

برچسبهای اساسی HTML

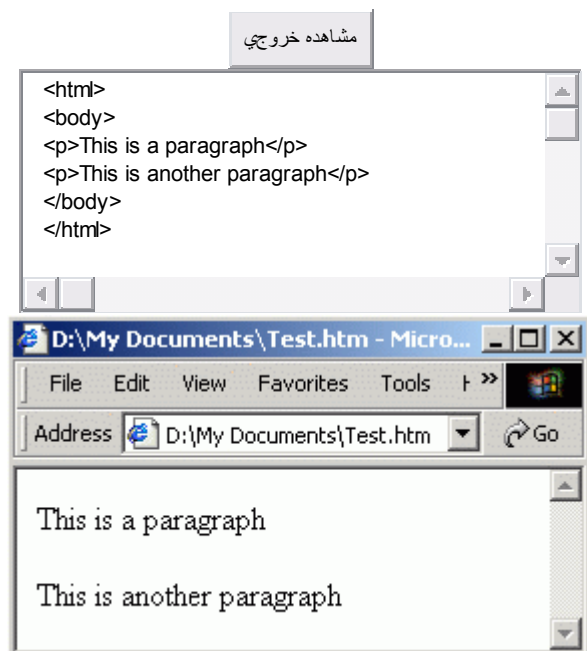
عنوانها

عنوانها توسط برچسب **<h1>** تا **<h6>** ایجاد می شوند که **<h1>** بزرگترین عنوان و **<h6>** کوچکترین عنوان را ایجاد می کند. HTML به طور خودکار یک خط خالی قبل و بعد از عنوان اضافه می کند. به مثال زیر توجه کنید.



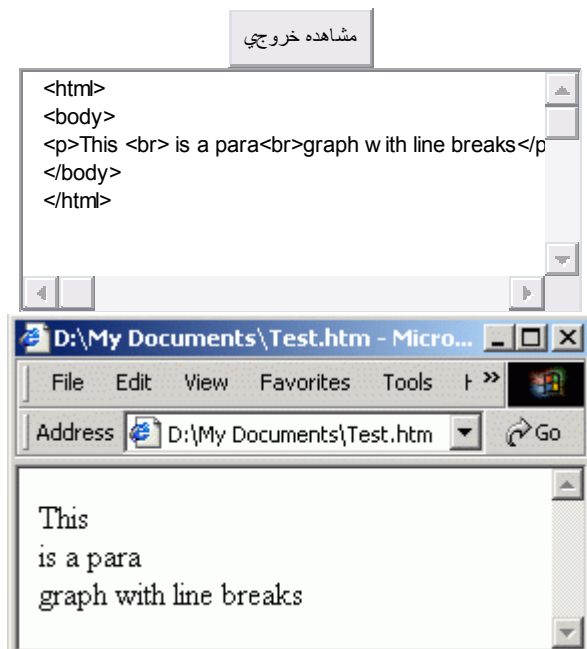
پاراگراف

پاراگرافها توسط برچسب **<p>** ایجاد می شوند؛ و همانند عنوانهای **HTML** به طور خودکار یک خط خالی قبل و بعد از عنوان اضافه می کند.



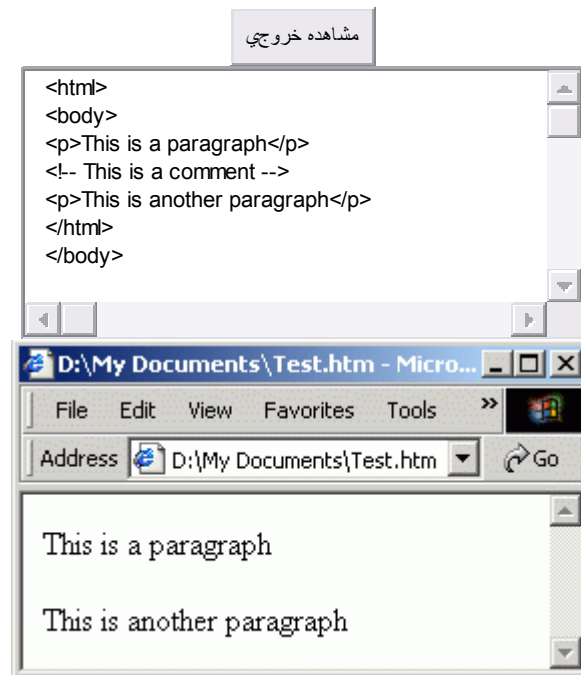
شکست خطوط

برچسب **
** هنگامی استفاده می شود که شما می خواهید یک خط را پایان دهید و ادامه متن به خط بعدی برود ولی پاراگراف جدیدی را شروع نکنید. برچسب **
** یک برچسب خالی می باشد و برچسب انتهایی ندارد.



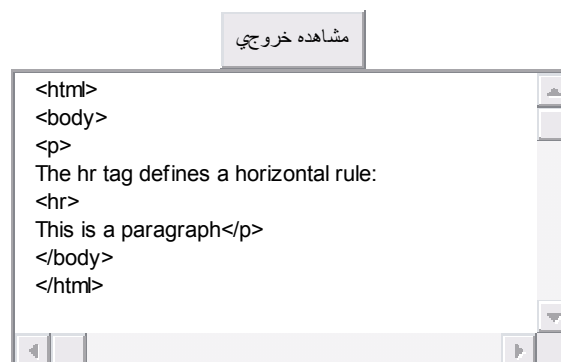
توضیحات در HTML

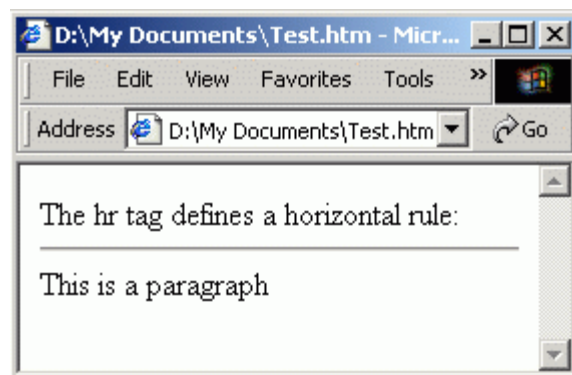
برچسب توضیحات برای قرار دادن توضیحاتی در فایل HTML استفاده می شود؛ و مرورگر آنها را نادیده می گیرد (<!-- This is a comment -->) (و در خروجی نمایش نمی دهد. شما توسط این برچسب می توانید فایل HTML خود را تشریح کنید و هنگامی که می خواهید آن را ویرایش نمایید به شما کمک خواهد کرد تا سردرگم نشوید، این مسئله خصوصا در فایل های HTML بزرگ صادق است.



برچسب خط افقی

برای جدا کردن قسمتهایی از متن می توانید از برچسب **<hr>** استفاده کنید. برای درک مطلب، بهتر است مثال زیر را ببینید.





چند نکته مفید :

* هنگامی که متنی را در **HTML** می نویسید هرگز نمی توانید مطمئن باشید که متن شما در مرورگرهای دیگر چگونه دیده خواهد شد. چون بعضی از صفحه نمایشها کوچک و بعضی بزرگ هستند. ضمناً متن شما هرگاه که کاربر اندازه پنجره خود را تغییر می دهد، شکل جدیدی به خود می گیرد. پس هرگز سعی نکنید که متن خود را با اضافه کردن خط خالی و یا فاصله بین حروف مرتب کنید.

* در **HTML** هر تعداد فاصله که درکنار هم استفاده کرده اید به عنوان یکی حساب می شود. پس از استفاده کردن فاصله های اضافی در کنار هم خودداری کنید، چون فقط اندازه فایل **HTML** خود را زیاد می کنید.

* هرگز از پاراگراف (**<p>**) خالی برای قرار دادن فاصله بین خطوط استفاده نکید و به جای آن از **
** استفاده کنید، در ضمن از **
** برای ساختن فهرست استفاده نکنید؛ کمی صبر کنید تا طرز ساختن فهرست ها را در **HTML** فرا بگیرید.

جدول برجسبهای اساسی و کاربرد آنها

برجسب ابتدایی	کاربرد
<html>	یک فایل HTML را ایجاد می کند .
<body>	بدنه فایل HTML را ایجاد می کند .
<h1>-<h6>	عنوان با اندازه یک تا شش را ایجاد می کند .
<p>	یک پاراگراف را ایجاد می کند .

	یک شکست در یک خط قرار می دهد .
<hr>	یک خط افقی در متن قرار می دهد .
<!-->	توضیحاتی به فایل HTML اضافه می کند .

قالببندی متن در HTML

HTML برجسبهای متعددی برای قالببندی متن در نظر گرفته است که در جداول زیر کاربرد هر یک ذکر شده است و برای درک بهتر کاربرد هر یک از برجسب ها، مثالهای متعددی آورده شده است.

جدول برچسبهای قالببندی متن

برچسب ابتدایی	کاربرد
	یک متن با حروف پررنگ را ایجاد می کند .
<big>	یک متن با حروف بزرگتر از اندازه پیش فرض را ایجاد می کند .
	یک متن با حروف ایتالیک را ایجاد می کند .
<i>	یک متن با حروف ایتالیک را ایجاد می کند .
<small>	یک متن با حروف کوچکتر از اندازه پیش فرض را ایجاد می کند .
	یک متن با حروف پررنگ را ایجاد می کند .
<sub>	یک متن زیرنویس را ایجاد می کند .
<sup>	یک متن بالانویس را ایجاد می کند .
<ins>	یک متن زیرخطدار را ایجاد می کند .
	یک متن با خطی بر روی آن را ایجاد می کند .

جدول برچسبهای اطلاعات خروجی رایانه

برچسب ابتدایی	کاربرد
<code>	یک متن که شامل کدهای رایانه ایست را ایجاد می کند .
<kbd>	یک متن که شامل حروف صفحه کلید است را ایجاد می کند .
<samp>	یک متن که شامل مثال هایی از کدهای رایانه ایست را ایجاد می کند .
<tt>	یک متن که شامل حروف ماشین تحریر است را ایجاد می کند .
<var>	یک متن که شامل متغیرهای رایانه ایست را ایجاد می کند .
<pre>	یک متن با قالببندی از پیش تعیین شده را ایجاد می کند .

جدول برجسبهای نقل قول، اختصار، تعریف

برجسب ابتدایی	کاربرد
<abbr>	یک کلمه اختصاری را ایجاد می کند .
<acronym>	یک کلمه متشکل از حروف اول چند کلمه را ایجاد می کند .
<address>	یک عنصر آدرس را ایجاد می کند .
<bdo>	جهت متن را تعیین می کند .
<blockquote>	یک نقل قول طولانی را ایجاد می کند .
<q>	یک نقل قول کوتاه را ایجاد می کند .
<cite>	یک نقل قول را ایجاد می کند .
<dfn>	یک متن توصیفی را ایجاد می کند .

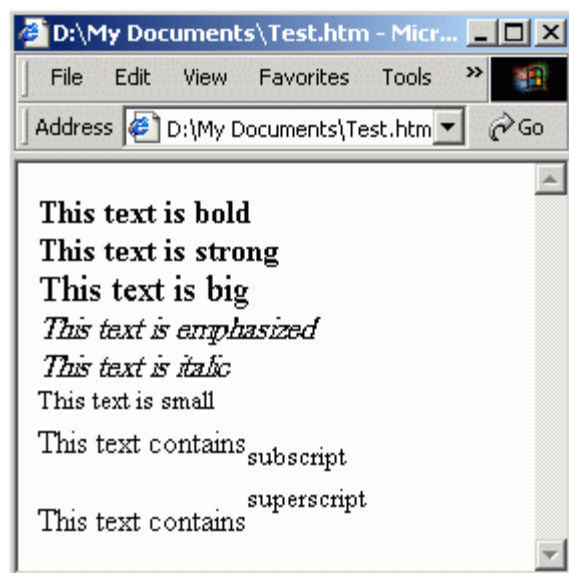
مثال زیر به شما نشان می دهد که چگونه می توانید یک متن را در یک فایل **HTML** قالببندی کنید.

مشاهده خروجی

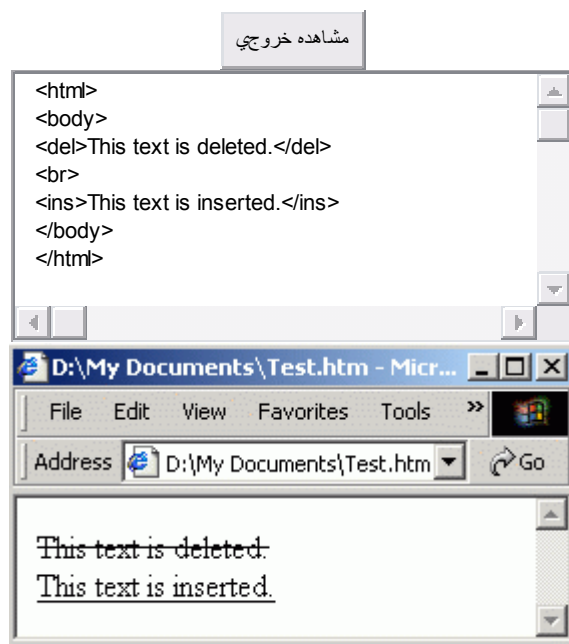
```

<html>
<body>
<b>This text is bold</b>
<br>
<strong>This text is strong</strong>
<br>
<big>This text is big</big>
<br>
<em>This text is emphasized</em>
<br>
<i>This text is italic</i>
<br>
<small>This text is small</small>
<br>
This text contains<sub>subscript</sub>
<br>
This text contains<sup>superscript</sup>
</body>
</html>

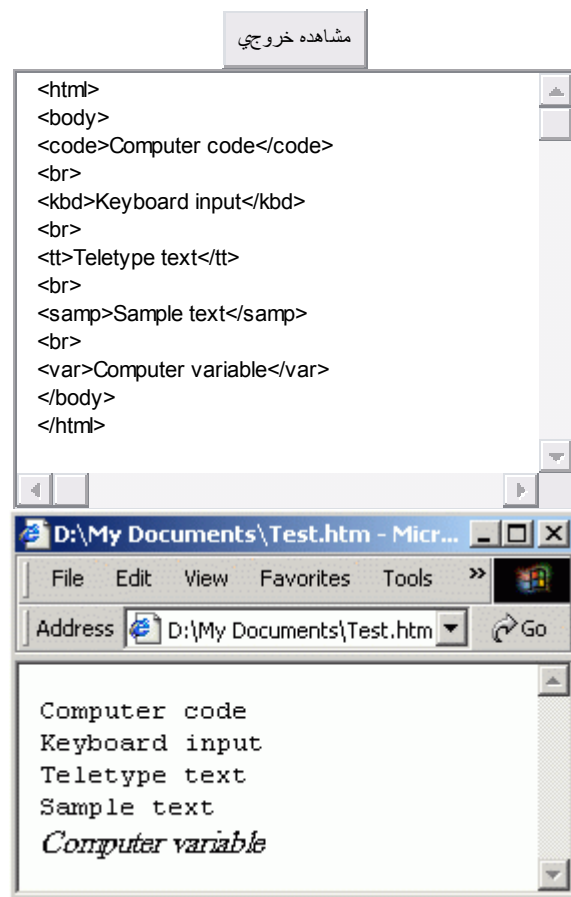
```



* مثال زیر به شما نشان می دهد که چگونه می توانید یک متن خط خورده و یک متن زیر خط دار را در یک فایل **HTML** قالببندی کنید.

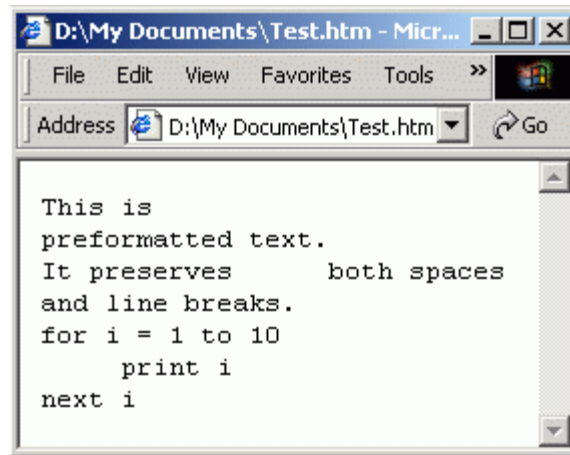


* مثال زیر نشان می دهد که برچسبهای مختلف خروجی رایانه چگونه نمایش داده می شوند.

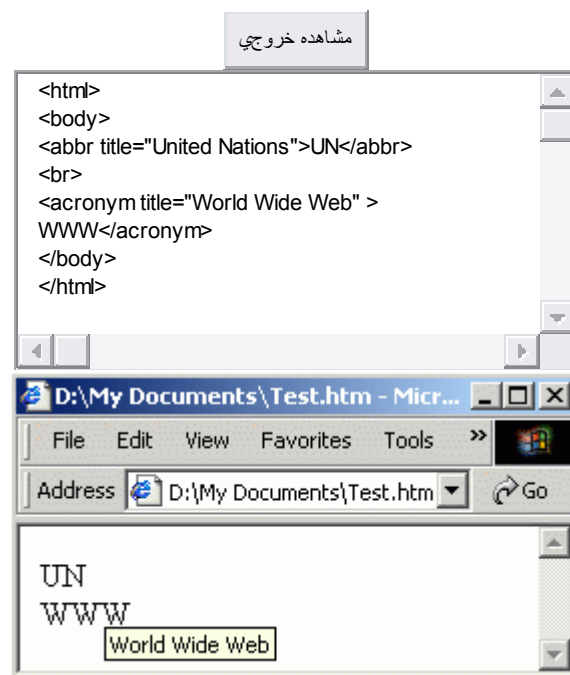


* مثال زیر نشان می دهد که چگونه می توان فاصله ها و شکست خطوط را با برچسب **<pre>** از قبل تنظیم کرد.

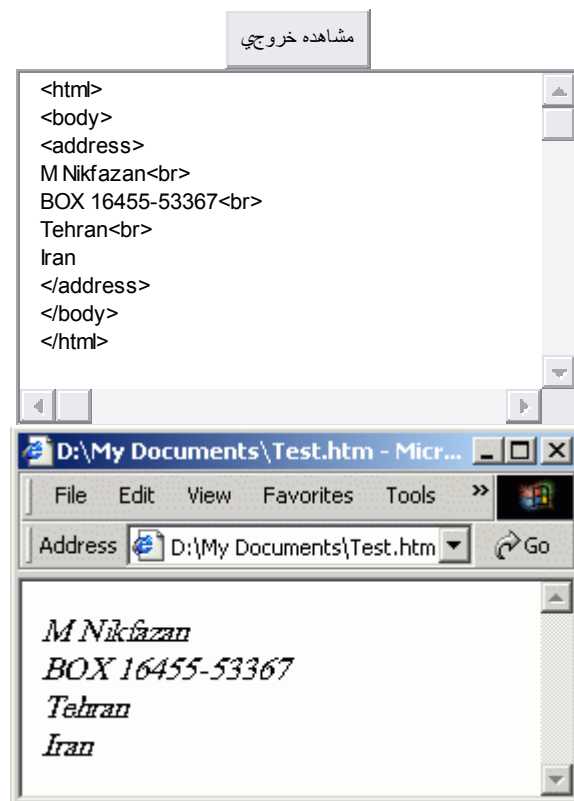




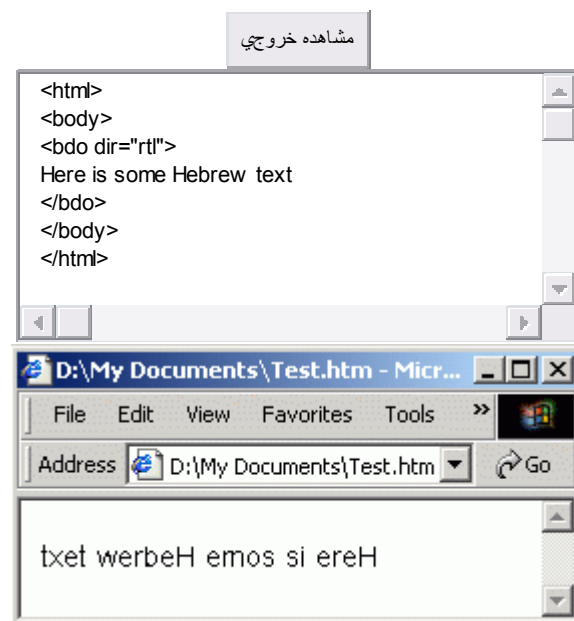
* از دو برجسب استفاده شده در مثال زیر هنگامی استفاده می شود که بخواهیم کلمه ای را به صورت اختصاری در متن استفاده کنیم؛ همچنین کاربر این امکان را داشته باشد که با بردن ماوس بر روی کلمه اختصاری و نگه داشتن ماوس بر روی آن برای چند لحظه، کلمات کاملی را کلمه اختصاری از آنها نشات گرفته است را در یک کادر کوچک زرد رنگ ببیند.



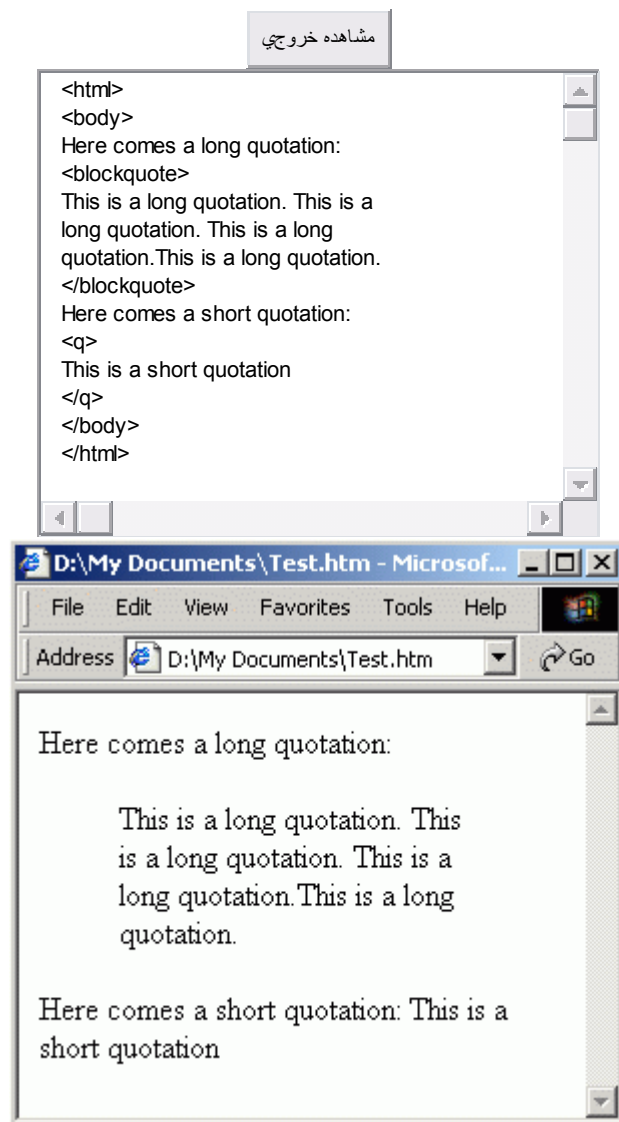
* مثال زیر نشان می دهد که آدرس را در یک فایل **HTML** چگونه بنویسیم.



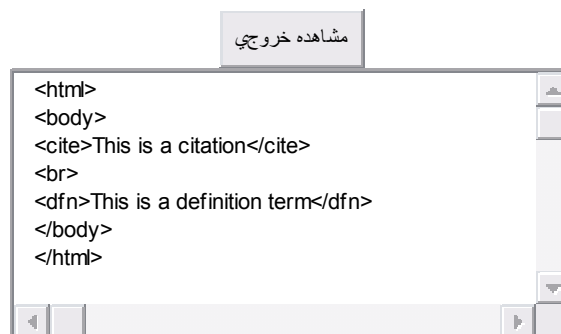
* مثال زیر طریقه استفاده از برچسب **<bdo>** را نشان می دهد. همان گونه که مشاهده می کنید این برچسب می تواند جهت نمایش متن را تغییر دهد مثلا حروف از راست به چپ نوشته شوند یا برعکس.

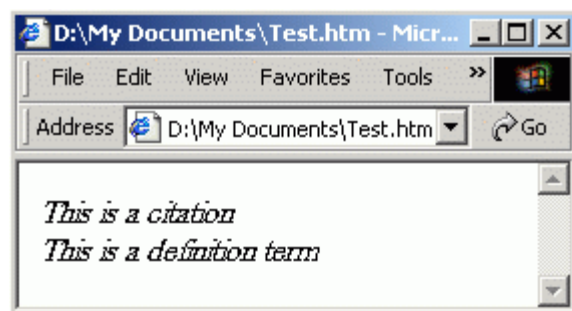


* توسط برچسب **<blockquote>** مرورگر یک خط خالی قبل و بعد از نقل قول قرار می دهد و همچنین حاشیه ای برای آن در نظر می گیرد. اما **<q>** کار خاصی روی متن انجام نمی دهد و فقط برای شما به عنوان سازنده فایل **HTML** مشخص می کند که عبارت داخل آن یک نقل قول است.



* مثال زیر جمله ای از کتابی خاص و یک جمله تعریفی را به نمایش می گذارد.





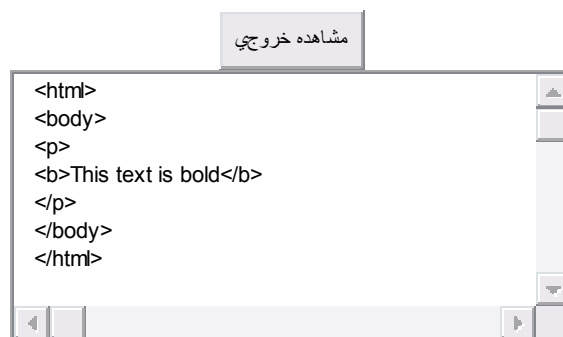
همانطور که در مثالهای فوق می بینید خروجی بعضی از برچسبها کاملاً مثل هم است و بعضی کار خاصی را انجام نمی دهند. در این موارد گاهی هدف از استفاده این برچسبها تنها مشخص کردن نوع متن برای نویسنده فایل **HTML** می باشد. به هر حال استفاده از قالببندی های فوق به سلیقه نویسنده فایل **HTML** است و ضرورتی ندارد که حتماً از موارد فوق دقیقاً بر اساس کاربرد استفاده شود.

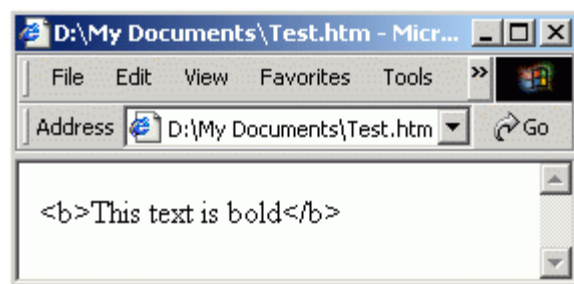
حروف نهادی در HTML

بعضی از حروف مانند < مفهوم خاصی برای HTML دارند، بنابراین از آنها در متن نمی توان استفاده کرد. برای نمایش حروفی مانند < در متن باید از حروف نهادی استفاده کرد. یک حرف نهادی از سه بخش اصلی تشکیل می شود :

; یک نام نهادی
; یک عدد نهادی #

مثلاً برای نمایش < در مرورگر باید به جای نوشتن < در فایل HTML باید **<it;** یا **<** را بنویسیم. فایده استفاده از نام به جای عدد اینست که همواره نام بهتر از عدد در خاطر می ماند؛ و ضرر آن اینست که همه مرورگر ها نام های نهادی را پشتیبانی نمی کنند ولی اکثر آنها اعداد نهادی را پشتیبانی می کنند. در ضمن توجه داشته باشید که نام های نهادی به بزرگی و کوچکی حروف حساس می باشند. به عنوان مثال اگر بخواهید عبارت **This text is bold** در مرورگر نمایش داده شود باید به شیوه مثال زیر عمل کنید.





*نکته

به طور معمول HTML فاصله ها را در متن نادیده می گیرد؛ مثلا اگر ده فاصله در متن داشته باشید نه تای آن نادیده گرفته خواهد شد. برای قرار دادن چندین فاصله در متن باید از حرف نهادی ** ** استفاده کنید.

حروف نهادی معمول :

نتیجه نام حرف نام نهادی عدد نهادی

نتیجه	نام حرف	نام نهادی	عدد نهادی
	non-breaking space	 	
<	less than	<	<
>	greater than	>	>
&	ampersand	&	&
"	quotation mark	"	"
'	apostrophe		'

تعدادی دیگر از حروف نهادی مورد استفاده :

نتیجه	نام حرف	نام نهادی	عدد نهادی
¡	inverted exclamation mark	¡	¡
¢	cent	¢	¢
£	pound	£	£
¥	yen	¥	¥
©	copyright	©	©
«	angle quotation mark (left)	«	«
®	registered trademark	®	®
°	degree	°	°



±	±	plus-or-minus	±
²	²	superscript 2	²
³	³	superscript 3	³
µ	µ	micro	μ
¹	¹	superscript 1	¹
»	»	angle quotation mark (right)	»
¼	¼	fraction 1/4	¼
½	½	fraction 1/2	½
¾	¾	fraction 3/4	¾
¿	¿	inverted question mark	¿
×	×	multiplication	×
÷	÷	division	÷

رنگ ها در HTML

از رنگها توسط سه شیوه RGB و HEX و Name می توان استفاده کرد. مثلا هر سه دستور زیر رنگ پس زمینه را به سیاه تغییر خواهند داد.

```
<body bgcolor="#000000">
<body bgcolor="rgb(0,0,0)">
<body bgcolor="black">
```

در استفاده از رنگها بهتر است از شیوه Name استفاده نکنید و دو شیوه دیگر را به کار ببرید چون نام رنگها برای همه مرورگرها قابل استفاده نمی باشد. توسط دستور **rgb** می توانید تا **16** میلیون رنگ بسازید یعنی **256×256×256** . نمونه ای از نام رنگها و شماره های آنها را در جدول زیر می بینید.

Color RGB	Color HEX	Color Name	Color
rgb(0,0,0)	#000000	black	
rgb(255,0,0)	#FF0000	red	
rgb(0,255,0)	#00FF00	lime	
rgb(0,0,255)	#0000FF	blue	
rgb(255,255,0)	#FFFF00	yellow	
rgb(0,255,255)	#00FFFF	aqua	
rgb(255,0,255)	#FF00FF	fuchsia	
rgb(192,192,192)	#C0C0C0	silver	

rgb(255,255,255)	#FFFFFF	white	
------------------	---------	-------	--

در جدول زیر نمونه ای از رنگها و شیوه استفاده از اعداد برای ساخت رنگهای مختلف را می بینید.

Color HEX	Color Name	Color
#000000	RGB(0,0,0)	
#404040	RGB(64,64,64)	
#808080	RGB(128,128,128)	
#C0C0C0	RGB(192,192,192)	
#E8E8E8	RGB(232,232,232)	
#FFFFFF	RGB(255,255,255)	

برای آشنایی با طریقه ایجاد رنگهای مورد نظر به دو مثال زیر توجه کنید.

مشاهده خروجی

```
<html>
<body bgcolor="#FFFF00">
<p>This is a paragraph</p>
<p>This is another paragraph</p>
</body>
</html>
```

مشاهده خروجی

```
<html>
<body bgcolor="rgb(0,255,0)">
<p>This is a paragraph</p>
<p>This is another paragraph</p>
</body>
</html>
```

فونتها در HTML

برای تعیین نوع فونت و رنگ و یا اندازه آن در یک فایل HTML می توان از برچسب **** استفاده کرد. اگر چه هنوز عده زیادی از این شیوه استفاده می کنند و حتی نرم افزار **FrontPage** نیز از این شیوه استفاده می کند؛ استفاده از این برچسب توصیه نمی شود. و بهتر است از شیوه های نگارش (CSS) استفاده کنید. در اینجا فقط برای آشنایی با برچسب **** به توضیح آن می پردازیم اما شما سعی کنید از این برچسب استفاده نکنید. برچسب **** دارای سه مشخصه **face**، **size** و **color** می باشد که در جدول زیر شرح داده شده اند.

کاربرد	مثال	مشخصه
اندازه فونت را تعیین می کند	size="2"	size="number"
فونت را بزرگتر می کند	size="+1"	size="+number"
فونت را کوچکتر می کند	size="-1"	size="-number"
نام فونت را تعیین می کند	face="Times"	face="face-name"
شماره رنگ فونت را تعیین می کند	color="#eeff00"	color="color-value"
نام رنگ فونت را تعیین می کند	color="red"	color="color-name"

مثال زیر طبقه استفاده از مشخصه های ذکر شده را نشان می دهد.

مشاهده خروجی

```

<html>
<body>
<p>
  <font size="2" face="Verdana"
    color="#ff00ff">
    This is a paragraph.
  </font>
</p>
<p>
  <font size="5" face="Times"
    color="blue">
    This is another paragraph.
  </font>
</p>
</body>
</html>

```

لینک ها در HTML

برچسب <a> و مشخصه href

HTML از برچسب **<a>** که a حرف اول کلمه **anchor** به معنی لنگر می باشد، برای ساختن پیوند با دیگر فایلها استفاده می کند (به جای استفاده از کلمه پیوند از کلمه آشنای لینک استفاده خواهیم کرد). یک لینک می تواند به هر منبعی در وب اشاره داشته باشد : یک صفحه **HTML** دیگر، یک تصویر، یک موزیک، یک فیلم یا هر چیز دیگری. قاعده ساخت یک لینک به صورت زیر می باشد.

```
<a href="url">Text to be displayed</a>
```

مشخصه **href** برای مشخص کردن آدرس فایلی که می خواهیم به آن لینک ایجاد کنیم، به کار می رود؛ و کلمات بین برچسب ابتدایی و انتهای به شکل فرایوند (**hyperlink**) نشان داده خواهد شد. به عنوان مثال خط زیر یک لینک به سایت **Yahoo** ایجاد می کند.

```
<a href="http://www.yahoo.com/">Visit YAHOO!</a>
```

دستور فوق در یک مرورگر به شکل روبرو دیده می شود : [!Visit YAHOO](http://www.yahoo.com/)

مشخصه target

توسط مشخصه target شما می توانید مشخص کنید که فایل پیوند داده شده به آن کجا باز شود. به عنوان مثال دستور زیر سایت **Yahoo** را در پنجره جدیدی نشان می دهد.

```
<a href="http://www.yahoo.com/"  
target="_blank">Visit YAHOO!</a>
```

کاربرد	مشخصه Target
یک فایل یا آدرس جدید را در یک پنجره جدید باز خواهد کرد.	target="_blank"
یک فایل یا آدرس جدید را در پنجره فعلی باز خواهد کرد.	target="_self"
یک فایل یا آدرس جدید را در قاب اصلی (parent frame) باز خواهد کرد. (راجع به قابها در مباحث بعدی صحبت خواهیم کرد.)	target="_parent"
یک فایل یا آدرس جدید را در کل پنجره فعلی باز خواهد کرد (راه جالبی برای خارج شدن از قاب ها)	target="_top"

مشخصه name

مشخصه **name** برای ساختن یک لینک دارای نام استفاده می شود. توسط لینک دارای نام شما می توانید لینکی بسازید که به قسمت خاصی از صفحه مستقیماً جهش داشته باشد به جای این که کار بر را مجبور به پیمودن صفحه کنید تا به قسمت مورد نظر برسد. قاعده ساخت یک لینک دارای نام به صورت زیر است :

```
<a name="label">Text to be displayed</a>
```

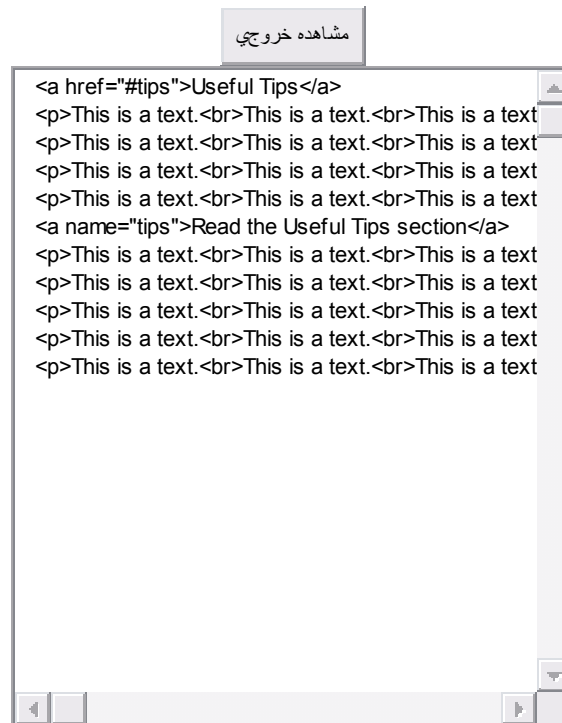
هنگامی که می خواهید به یک لینک دارای نام متصل شوید باید به انتهای نام فایل علامت **#** را اضافه کرده سپس نام لینک مورد نظر را بنویسید. مثلاً اگر دستورات زیر را در فایلی به نام **test1.htm** ذخیره کنید

```
<p>This is a text. This is a text. This is a text.</p>  
<p>This is a text. This is a text. This is a text.</p>  
<a name="tips">Read the Useful Tips section</a>  
<p>This is a text. This is a text. This is a text.</p>  
<p>This is a text. This is a text. This is a text.</p>
```

و دستورات زیر را در فایل به نام **test2.htm** ذخیره کنید

```
<a href="test1.htm#tips">
Read the Useful Tips section</a>
```

و فایل **test2.htm** را باز کنید و روی لینک موجود کلیک کنید، فایل **test1.htm** باز خواهد شد و به بخش **tips** خواهد رفت. همچنین شما می توانید از طریق پیوند دارای نام به بخشی از صفحه فعلی که توسط مرورگر در حال نمایش است بروید. مثلاً اگر دستورات زیر را در فایل **test.htm** نوشته و آن را ذخیره و توسط مرورگر باز کنید، سپس روی **Useful Tips** کلیک کنید صفحه به سوی بخش **tips** جهش می کند.



E-Mail لینک به یک آدرس

گاهی اوقات می خواهیم در صفحه وب خود آدرس **E-Mail** خود را در اختیار بینندگان سایت قرار دهیم، که به راحتی می توانیم این کار را با نوشتن آدرس خود مثلاً داخل یک پاراگراف عملی سازیم. مانند :

```
<p>My E-Mail adress is : you@yourname.com</p>
```

اما اگر بخواهیم یک لینک به آدرس **E-Mail** خود ایجاد کنیم که به طور خودکار یکی از برنامه های مخصوص **E-Mail** را در رایانه کاربر باز کرده و بعضی از موارد مورد نیاز مانند آدرس و موضوع و... را به طور خودکار در قسمت های مخصوص خود بنویسد و بدین ترتیب کار کاربر را کم کند، باید از شیوه ای همانند دستور زیر استفاده کنیم :

```
<a
href="mailto: you@yourname.com?Subject=Hello%20again">
Send Mail</a>
```

مثال زیر مطلب را روشن تر خواهد کرد :



ذخیره کرده و آن را باز کنید یا بر روی دکمه مشاهده خروجی کلیک **test.htm** دستورات فوق را در فایل کنید. اگر بر روی **Send mail!** کلیک کنید برنامه **Outlook Express** باز خواهد شد و همانطور که می بینید تعدادی از موارد به طور پیش فرض توسط شما مشخص شده است.

چند نکته مفید :

* هرگاه که می خواهید لینکی به یک شاخه ایجاد کنید در انتهای نام شاخه یک / اضافه کنید چون اگر بخواهید مثلاً لینکی به شاخه **html** در سایت **yourname.com** ایجاد کنید و از ساختار **href="http://www.yourname.com/html"** استفاده کنید یک تقاضای اضافه به سرور وب خود تحمیل خواهید کرد چون سرور ابتدا به انتهای آدرس فوق یک / اضافه می کند سپس به آدرس جدید که آدرس زیر است پاسخ می دهد و صفحه لازم را برای کاربر می فرستد :
href="http://www.yourname.com/html/"
* لینک های دارای نام غالب برای ساخت فهرست مندرجات فایل های طولانی در ابتدای صفحه استفاده می شوند. به این صورت که هر فصل یک لینک دارای نام دارد و لینکی به هر یک از آنها در ابتدای صفحه داده می شود.
* توجه داشته باشید هنگامی که در مشخصه **href** می خواهید از فاصله استفاده کنید باید فاصله را با **20%** جایگزین کنید تا مطمئن باشید که فاصله ها به درستی توسط مرورگر تشخیص داده خواهند شد.

تصاویر در HTML

برچسب و مشخصه src

تصاویر در فایل های **HTML** توسط برچسب **** قرار داده می شوند. این برچسب، یک برچسب خالی می باشد یعنی نیازی به برچسب انتهایی ندارد و فقط شامل مشخصه های مربوط به خود می باشد. برای مشخص کردن تصویری که باید در صفحه قرار بگیرد نیاز به استفاده از مشخصه **(src (source** داریم که به وسیله آن آدرس و نام فایل معرفی می شود.

```

```

اگر فایل **HTML** و تصویر در کنار هم قرار داشته باشند می توانیم از **** استفاده کنیم و نیازی به ذکر کامل آدرس نیست. مرورگر تصویر را در هرجایی که از برچسب **img** استفاده کرده اید قرار خواهد داد. مثلاً اگر از این برچسب بین دو پاراگراف استفاده کرده اید مرورگر ابتدا پاراگراف اول را نمایش می دهد سپس تصویر و در نهایت پاراگراف دیگر را نشان خواهد داد. به عنوان مثال دیگر اگر عبارت **<td></td>** را در جدول به کار ببرید یکی از خانه های جدول شامل تصویر **product.gif** خواهد بود. (در مورد جداول در مبحث بعدی صحبت خواهیم کرد)

مشخصه alt

این مشخصه برای ایجاد یک متن توصیفی بر روی تصویر به کار می رود؛ و وقتی کاربر ماوس را روی تصویر می برد و برای چند لحظه نگاه می دارد، این متن در یک کادر کوچک دیده خواهد شد مانند :

```

```

که باعث دیده شدن عبارت **Big Boat** در یک کادر کوچک می شود و کاربرد این مشخصه بیشتر وقتی است که اگر تصویر به دلیل هر مشکلی در صفحه دیده نشد کاربر بداند که چه تصویری را ندیده است که در این صورت عبارت مشخصه **alt** به جای تصویر دیده خواهد شد. مانند :

*نکته

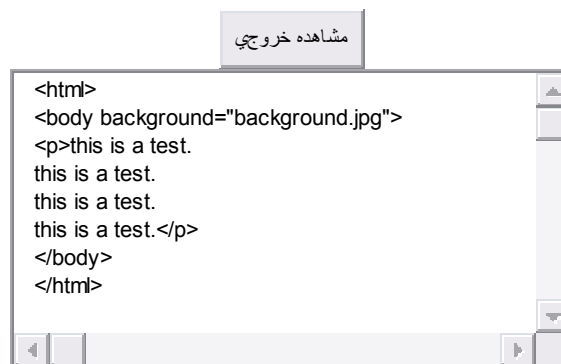
سعی کنید در صفحات وبی که می سازد فقط در صورت نیاز از تصویر و عکس استفاده کنید و از زیاد کردن حجم صفحه خود بپرهیزید چون هنگامی که کاربر در اینترنت صفحه شما را می بیند باید مدت زیادتری برای تکمیل شدن نمایش صفحه شما صبر کند و ممکن است این موضوع کاربر را از دیدن صفحه پشیمان کند !

تصویر پس زمینه

برای اضافه کردن تصویری به پس زمینه صفحه می توانید از ساختار :

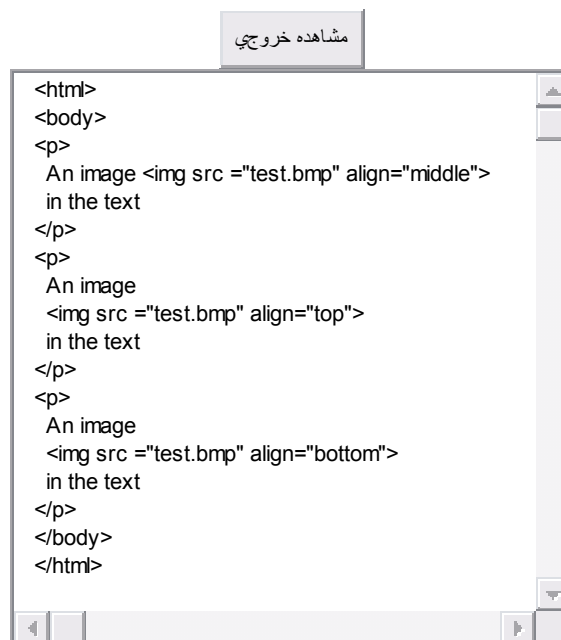
```
<body background="background.jpg">
```

استفاده کنید که به جای **background.jpg** می توانید نام هر فایل تصویری دیگری را قرار دهید. اگر ابعاد تصویر شما از صفحه کوچکتر باشد تصویر تا پر کردن کل صفحه تکرار خواهد شد. پس اگر تصمیم دارید که تصویری به پس زمینه خود اضافه کنید توصیه می شود که از فایل های کوچکی که می توانند مانند موزائیک در کنار هم قرار گیرند، استفاده کنید. ضمناً در انتخاب تصویر و انتخاب رنگ متن دقت کنید تا متن شما خوانایی خود را از دست ندهد. مثال زیر در همین رابطه است.



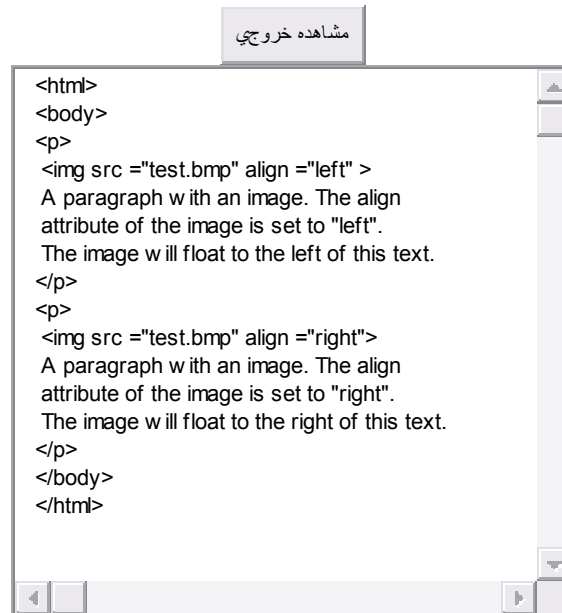
در تصاویر align مشخصه

این مشخصه طریقه تراز شدن تصاویر و متن را تعیین می کند. مثال زیر در همین رابطه است.



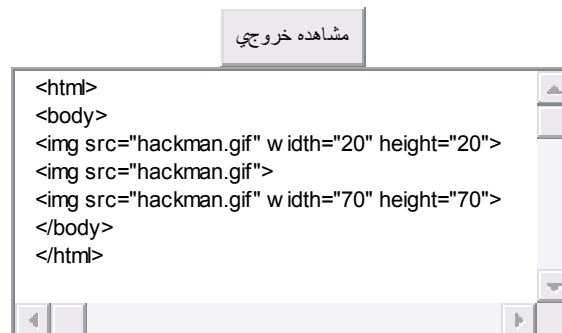
قرار خواهند گرفت. ***align="bottom"** توجه داشته باشید که تصاویر به طور پیش فرض به صورت

همچنین مشخصه **align** می تواند طرز قرار گرفتن یک تصویر در کنار یک پاراگراف را تعیین کند که مثلاً تصویر در سمت راست پاراگراف قرار گیرد یا در سمت چپ. به مثال زیر توجه کنید.



تغییر ابعاد تصاویر

توسط مشخصه **width** و **height** می توانید ابعاد تصویر را تغییر داده و آن را بزرگتر و یا کوچکتر کنید. فقط دقت کنید که ابعاد را متناسب تغییر دهید چون در غیر این صورت تصویر وضوح و زیبایی خود را از دست خواهد داد.



ساخت لینک بر روی قسمتی از تصویر

برای این کار باید از دو برچسب **<map>** و **<area>** استفاده کنید. روش کار بدین صورت است که ناحیه هایی را می خواهید کاربر با کلیک بر روی آن به صفحه دیگری متصل شود را توسط اشکالی مشخص می کنید. این اشکال دایره، مستطیل و چند ضلعی خواهند بود که توسط مشخصه **shape** تعیین می شوند.

*دایره به صورت **shape="circle"** معرفی می شود و مختصات آن توسط مشخصه **coords** تعیین می شود. مثلاً در **coords="39, 41, 32"** دو عدد اول مختصات مرکز دایره و عدد سمت راست اندازه شعاع دایره می باشد. مشخصه آشنای **href** که در برچسب **<a>** از آن استفاده کردیم، صفحه ای را که کاربر با کلیک بر ناحیه مشخص شده توسط شکل به آن منتقل خواهد شد را تعیین می کند.

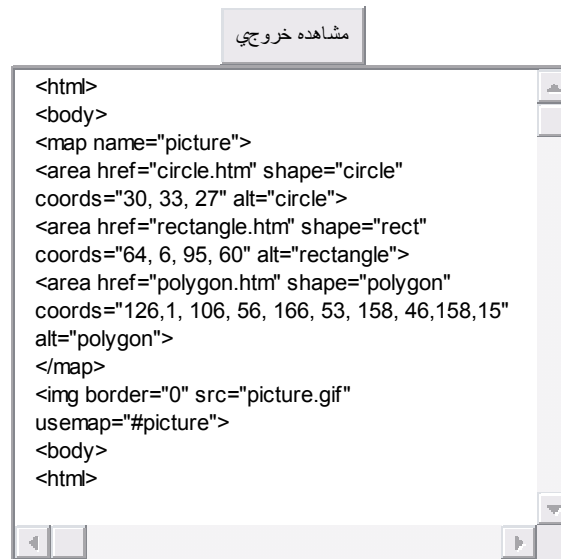
*مستطیل توسط **shape="rect"** و مختصات قطر آن مثلاً توسط **coords="77, 12, 112, 72"** تعیین می شود که دو عدد سمت چپ مختصات گوشه بالای سمت چپ و دو عدد سمت راست مختصات گوشه پایین سمت راست می باشند.

* برای استفاده از چند ضلعی باید از **shape="polygon"** استفاده کنید و اعداد دویه دو مختصات گوشه های آن را تعیین می کند : **coords="146, 7, 124, 68, 193, 64, 183, 53, 184, 20"**.

همانطور که در مثال زیر می بینید برچسب **<map>** مشخصه ای به نام **name** دارد که توسط آن برچسب **** با برچسب **<map>** ارتباط برقرار می کند برای ایجاد این ارتباط باید در برچسب **** از مشخصه **usemap** استفاده کنیم که مقدار این مشخصه نام برچسب **<map>** می باشد. ضمناً نام برچسب **<map>** باید همراه با علامت **#** به کار برده شود. مانند:

```
usemap="#picture"
```

مثال زیر روش استفاده از برچسب ها و مشخصه های ذکر شده را مشخص خواهد کرد.



ساخت لینک برای تصویر

شما می توانید با کلیک بر روی هر جای تصویر به آدرس مورد نظر متصل شوید برای اینکار باید از روش زیر استفاده کنید، که در اینجا مثلاً با کلیک بر روی تصویر **buttonnext.gif** به صفحه **nextpage.htm** خواهید رفت.

```
<a href="nextpage.htm"></a>
```

جدول ها در HTML

جدول ها در **HTML** توسط برچسب **<table>** ایجاد می شوند. یک جدول توسط برچسب **<tr>** به سطرها تقسیم می شود و هر سطر توسط برچسب **<td>** به خانه های اطلاعاتی تقسیم می شود. یک خانه اطلاعاتی میتواند شامل متن، تصویر، فهرست، پاراگراف، فرم، جدول و یا هر چیز دیگری باشد. دو مثال زیر طریقه استفاده از برچسبهای گفته شده را نشان می دهد.

* مثال زیر یک جدول حاشیه دارمی باشد که ضخامت حاشیه توسط مشخصه **border="1"** تعیین شده است.

مشاهده خروجی

```
<html>
<body>
<table border="1">
  <tr>
    <td>row 1, cell 1</td>
    <td>row 1, cell 2</td>
  </tr>
  <tr>
    <td>row 2, cell 1</td>
    <td>row 2, cell 2</td>
  </tr>
</table>
</body>
</html>
```

* در مثال زیر یک جدول بدون حاشیه نشان داده شده است.

مشاهده خروجی

```
<html>
<body>
<table>
  <tr>
    <td>row 1, cell 1</td>
    <td>row 1, cell 2</td>
  </tr>
  <tr>
    <td>row 2, cell 1</td>
    <td>row 2, cell 2</td>
  </tr>
</table>
</body>
</html>
```

border مشخصه

اگر شما مشخصه **border** را تعیین نکنید جدول شما حاشیه نخواهد داشت. گاهی اوقات جدول بدون حاشیه برای طرحبندی صفحه مفید است ولی اغلب اوقات می خواهید که جدولتان حاشیه داشته باشد. برای نمایش جدولی با حاشیه باید از مشخصه **border** استفاده کنید؛ همانند جدول مثال قبل.

عنوانها در جدول

عناوین ستونها در یک جدول توسط برچسب **<th>** مشخص می شوند. به مثال زیر توجه کنید.

مشاهده خروجی

```
<html>
<body>
<table border="1">
  <tr>
    <th>Heading</th>
    <th>Another Heading</th>
  </tr>
  <tr>
    <td>row 1, cell 1</td>
    <td>row 1, cell 2</td>
  </tr>
  <tr>
    <td>row 2, cell 1</td>
    <td>row 2, cell 2</td>
  </tr>
</table>
</body>
</html>
```

< می توانید عنوانی برای جدول خود در نظر بگیرید. به مثال زیر توجه کنید. **caption** توسط برچسب >

مشاهده خروجی

```
<html>
<body>
<table border="1">
  <caption>Caption</caption>
  <tr>
    <td>row 1, cell 1</td>
    <td>row 1, cell 2</td>
  </tr>
  <tr>
    <td>row 2, cell 1</td>
    <td>row 2, cell 2</td>
  </tr>
</table>
</body>
</html>
```

خانه های خالی در جدول

گاهی اوقات ما نیاز پیدا می کنیم که بعضی از خانه های جدول خالی باشند. اگر جدول ما دارای حاشیه باشد و در خانه مورد نظر چیزی ننویسیم (<td/><td>) در این صورت جدول ما به صورت زیر دیده می شود.

row 1, cell 1	row 1, cell 2
row 2, cell 1	

برای جلوگیری از بهم خوردن حاشیه باید حرف نهادی ** ** را بکار ببریم یعنی از **<td/> <td>** به جای **<td/><td>** استفاده کنیم در این صورت جدول به صورت زیر خواهد بود.

row 1, cell 1	row 1, cell 2
row 2, cell 1	

مشخصه colspan

برای درک مفهوم این مشخصه و کاربرد آن به مثال زیر توجه کنید.

```
مشاهده خروجي
<html>
<body>
<table border="1">
  <tr>
    <th>Name</th>
    <th colspan="2">Telephone</th>
  </tr>
  <tr>
    <td>Bill Gates</td>
    <td>555 77 854</td>
    <td>555 77 855</td>
  </tr>
</table>
</body>
</html>
```

مشخصه rowspan

برای درک مفهوم این مشخصه و کاربرد آن به مثال زیر توجه کنید.

```
مشاهده خروجي
<html>
<body>
<table border="1">
  <tr>
    <th>First Name:</th>
    <td>Bill Gates</td>
  </tr>
  <tr>
    <th rowspan="2">Telephone:</th>
    <td>555 77 854</td>
  </tr>
  <tr>
    <td>555 77 855</td>
  </tr>
</table>
</body>
</html>
```

cellpadding مشخصه

از این مشخصه برای ساختن فضای خالی بیشتر بین محتویات هر خانه و حاشیه آن استفاده می شود. به مثال زیر توجه کنید.

مشاهده خروجی

```
<html>
<body>
<table border="1" cellpadding="10">
  <tr><td>First</td>
    <td>Row </td>
  </tr>
  <tr>
    <td>Second</td>
    <td>Row </td>
  </tr>
</table>
</body>
</html>
```

cellspacing مشخصه

از این مشخصه برای ساختن فضای خالی بیشتر بین خانه ها استفاده می شود. به مثال زیر توجه کنید.

مشاهده خروجی

```
<html>
<body>
<table border="1" cellspacing="10">
  <tr>
    <td>First</td>
    <td>Row </td>
  </tr>
  <tr>
    <td>Second</td>
    <td>Row </td></tr>
</table>
</body>
</html>
```

چند مشخصه دیگر در مورد جداول

*توسط مشخصه **bgcolor** می توانید رنگ زمینه جدول یا خانه های آن را تعیین کنید.

مشاهده خروجی

```
<html>
<body>
<table border="1" bgcolor="red">
<tr>
<td bgcolor="#00ff00">First</td>
<td>Row </td>
</tr>
<tr>
<td>Second</td>
<td>Row </td></tr>
</table>
</body>
</html>
```

ارتفاع جدول یا یک خانه را **height** عرض جدول یا یک خانه را تعیین می کند و مشخصه ***width** مشخصه تعیین می کند. و مقدار این دو مشخصه معمولا بر حسب **pixel** می باشد.

*توسط مشخصه **background** می توانید تصویری در پس زمینه جدول یا خانه ها قرار دهید.

مشاهده خروجی

```
<html>
<body>
<table border="1" height="200" width="200" backgro
<tr>
<td background="test.bmp">First</td>
<td>Row </td>
</tr>
<tr>
<td>Second</td>
<td>Row </td></tr>
</table>
</body>
</html>
```

برای مرتب کردن محتویات خانه های یک جدول در جهت افقی استفاده می شود که ***align** از مشخصه مثلا در سمت راست یا وسط و یا سمت چپ خانه قرار بگیرند و یا اینکه اگر محتویات خانه متن است، متن به صورت **justify** مرتب شود؛ و از مشخصه **valign** برای مرتب کردن محتویات خانه های یک جدول در جهت عمودی استفاده می شود.

مشاهده خروجی

```
<html>
<body>
<table border="1" height="200" width="300"
<tr>
  <td align="left" >First</td>
  <td align="center" >Second</td>
  <td align="right" >Third</td>
</tr>
<tr>
  <td valign="top" >First</td>
  <td valign="middle" >Second</td>
  <td valign="bottom" >Third</td>
</table>
</body>
</html>
```

```
<td align="left/center/right/justify">Make-Up</td>
<td valign="top/middle/bottom">Make-Up</td>
```

*مشخصه دیگری که در جدولها کاربرد دارد مشخصه **frame** است که دارای انواع **void** ، **box** ، **border** ، **below** ، **above** ، **hsides** ، **vsides** ، **lhs** ، **rhs** می باشد. سه مثال زیر کاربرد این مشخصه را نشان می دهد.


```
<h4>With frame="border":</h4>
<table frame="border">
  <tr>
    <td>First</td><td>Row </td>
  </tr>
  <tr>
    <td>Second</td><td>Row </td>
  </tr>
</table>

<h4>With frame="box":</h4>
<table frame="box">
  <tr>
    <td>First</td><td>Row </td>
  </tr>
  <tr>
    <td>Second</td><td>Row </td>
  </tr>
</table>

<h4>With frame="void":</h4>
<table frame="void">
  <tr>
    <td>First</td><td>Row </td>
  </tr>
  <tr>
    <td>Second</td><td>Row </td>
  </tr>
</table>
```

```
<h4>With frame="above":</h4>
<table frame="above">
  <tr>
    <td>First</td><td>Row </td>
  </tr>
  <tr>
    <td>Second</td><td>Row </td>
  </tr>
</table>

<h4>With frame="below ":</h4>
<table frame="below ">
  <tr>
    <td>First</td><td>Row </td>
  </tr>
  <tr>
    <td>Second</td><td>Row </td>
  </tr>
</table>

<h4>With frame="hsides":</h4>
<table frame="hsides">
  <tr>
    <td>First</td><td>Row </td>
  </tr>
  <tr>
    <td>Second</td><td>Row </td>
  </tr>
</table>
```

مشاهده خروجی

```
<h4>With frame="vsides":</h4>
<table frame="vsides">
  <tr>
    <td>First</td><td>Row </td>
  </tr>
  <tr>
    <td>Second</td><td>Row </td>
  </tr>
</table>

<h4>With frame="lhs":</h4>
<table frame="lhs">
  <tr>
    <td>First</td><td>Row </td>
  </tr>
  <tr>
    <td>Second</td><td>Row </td>
  </tr>
</table>

<h4>With frame="rhs":</h4>
<table frame="rhs">
  <tr>
    <td>First</td><td>Row </td>
  </tr>
  <tr>
    <td>Second</td><td>Row </td>
  </tr>
</table>
```

نکته

یکی از نکات مهم در زیبایی صفحه HTML شما استفاده از طرح بندی (layout) مناسب است. یکی از معمولترین راهها در ایجاد طرحبندی دلخواه استفاده از جداول می باشد .

و برای اینکار معمولا از جداول بدون حاشیه استفاده می کنند. به عنوان مثال متن این نکته به شیوه روزنامه ای و توسط یک جدول بی حاشیه و دارای سه ستون ایجاد شده است .

فهرستها در HTML

HTML فهرستهای شماره دار، بدون شماره و فهرستهای دارای توضیح را پشتیبانی می کند.

فهرستهای بدون شماره

اینگونه فهرستها توسط یک شکل کوچک مشخص می شوند؛ و توسط برچسب **** شروع شده و هر مورد آن با برچسب **** مشخص می شود. مثال زیر طریقه ساخت یک فهرست بدون شماره را نشان می دهد.

```
مشاهده خروجي
<html>
<body>
<ul>
<li>Coffee</li>
<li>Milk</li>
</ul>
</body>
</html>
```

در داخل موارد فهرست همانند خانه های جدول شما می توانید پاراگراف، تصویر، جدول، یک فهرست دیگر و... قرار دهید. مثال زیر طبقه استفاده از این روش را نشان می دهد.

```
مشاهده خروجي
<html>
<body>
<ul>
<li><p>This is a paragraph. This is a paragraph. This</p>
<li><table border="1"><tr>
<td>row 1, cell 1</td>
<td>row 1, cell 2</td>
</tr><tr>
<td>row 2, cell 1</td>
<td>row 2, cell 2</td>
</tr></table></li>
</ul>
</body>
</html>
```

فهرستهای شماره دار

منظور از فهرستهای شماره دار، فهرستهایی است که موارد آن توسط عدد، شماره گذاری شده اند. فهرستهای شماره دار توسط برچسب **** شروع شده و هر مورد آن با برچسب **** مشخص می شود. مثال زیر طبقه ساخت یک فهرست شماره دار را نشان می دهد.

```
مشاهده خروجي
<html>
<body>
<ol>
<li>Coffee</li>
<li>Milk</li>
</ol>
</body>
</html>
```

فهرستهای دارای توضیح

اینگونه از فهرستها دارای شماره یا شکل نمی باشند و برای مواردی که می خواهیم توضیحی در کنارشان باشد به کار می روند. فهرستهای دارای توضیح توسط برچسب **<dl>** شروع شده و هر مورد آن با برچسب **<dt>** مشخص می شود و توضیحات در داخل برچسب **<dd>** قرار می گیرند. مثال زیر طریقه ساخت یک فهرست دارای توضیح را نشان می دهد.

```
مشاهده خروجي
```

```
<html>
<body>
<dl>
<dt>Coffee</dt>
<dd>Black hot drink</dd>
<dt>Milk</dt>
<dd>White cold drink</dd>
</dl>
</body>
</html>
```

در فهرستها type مشخصه

توسط این مشخصه می توانید شکلی را که در کنار موارد فهرستهای بدون شماره قرار می گیرد و یا اعداد فهرستهای شماره دار را به نوع دلخواه تغییر دهید. شکلهای می توانند سه نوع دایره توپر ، دایره توخالی و مربع (disc , circle , square) باشند و شماره گذاری ها می توانند به چهار صورت "I II III" یا "i ii iii" ، "A B C" ، "a b c" باشند که به ترتیب با **I** ، **i** ، **A** ، **a** مشخص می شوند.

در زیر مثالی از استفاده type در فهرست بدون شماره را می بینید.

```
مشاهده خروجي
```

```
<html>
<body>
<h4>Square bullets list:</h4>
<ul type="square">
<li>Apples</li>
<li>Bananas</li>
<li>Lemons</li>
<li>Oranges</li>
</ul>
</body>
</html>
```

در فهرست شماره دار را می بینید. **type** در زیر مثالی از استفاده

```
مشاهده خروجي
<html>
<body>
<h4>Roman numbers list:</h4>
<ol type="I">
<li>Apples</li>
<li>Bananas</li>
<li>Lemons</li>
<li>Oranges</li>
</ol>
</body>
</html>
```

مثال زیر روش استفاده از فهرستهای تودرتو را نشان می دهد. همان طور که می بینید فهرست داخلی توسط شکل دیگری (دایره تو خالی) مشخص شده است.

```
مشاهده خروجي
<html>
<body>
<ul>
<li>Coffee</li>
<li>Tea
<ul>
<li>Black tea</li>
<li>Green tea</li>
</ul>
</li>
<li>Milk</li>
</ul>
</body>
</html>
```

فرمها و داده های ورودی در HTML

یک فرم ناحیه ای در صفحه HTML است که شامل عناصری می باشد. عناصر فرم به کاربر اجازه می دهند که اطلاعاتی را وارد صفحه ما کند؛ و ما با دریافت آن اطلاعات به اهدافی که داریم برسیم. یک فرم توسط برچسب **<form>** ایجاد می شود.

برچسب <input>

پرکاربرد ترین برچسب در فرمها برچسب **<input>** می باشد در اینجا انواع این برچسب را شرح می دهیم.

جعبه متن

از این جعبه هنگامی استفاده می شود که بخواهیم کاربر کلمات یا اعدادی را در فرم بنویسد؛ و توسط مشخصه **type="text"** تعیین می شود. مثال زیر در همین رابطه است. توجه داشته باشید که خود فرم

قابل رویت نمی باشد فقط عناصر داخل آن را می توانیم ببینیم. در ضمن اندازه پیش فرض برای جعبه متن در اکثر مرورگر ها **20** حرف می باشد و اگر می خواهید این پیش فرض را تغییر دهید باید از مشخصه **size** استفاده کنید (**size=40**).

مشاهده خروجی

```
<html>
<body>
<form>
First name:
<input type="text" name="firstname">
<br> Last name:
<input type="text" name="lastname">
</form>
</body>
</html>
```

دکمه های رادیویی

از این دکمه ها هنگامی استفاده می شود که بخواهیم کاربر یکی از چند گزینه به تعداد محدود را انتخاب کند؛ و توسط مشخصه **type="radio"** تعیین می شود. با انتخاب یکی از دکمه ها، دکمه دیگری با همان نام اگر در حالت انتخاب باشند، از حالت انتخاب خارج می شود. اگر کلمه **checked** را در برچسب **input** از نوع دکمه های رادیویی بنویسید آن دکمه به طور پیش فرض به صورت انتخاب شده خواهد شد.

```
<input type="radio" name="s" value="male" checked >
```

مشاهده خروجی

```
<html>
<body>
<form>
<input type="radio" name="s" value="male"> Male<br>
<input type="radio" name="s" value="female"> Female
</form>
</body>
</html>
```

جعبه انتخاب

از این جعبه ها هنگامی استفاده می شود که بخواهیم کاربر یک یا چند گزینه از گزینه هایی به تعداد محدود را انتخاب کند؛ و توسط مشخصه **type="checkbox"** تعیین می شود. کاربر می تواند یک یا چند جعبه انتخاب را همزمان انتخاب کند. اگر کلمه **checked** را در برچسب **input** از نوع جعبه انتخاب بنویسید آن جعبه به طور پیش فرض به صورت انتخاب شده خواهد شد.

```
<input type="checkbox" name="bike" checked>
```

```
مشاهده خروجي
<html>
<body>
<form>
<input type="checkbox" name="bike">
I have a bike<br>
<input type="checkbox" name="car">
I have a car
</form>
</body>
</html>
```

جعبه کلمه عبور

از این جعبه هنگامی استفاده می شود که بخواهیم کاربر کلمه عبور یا هر چیزی که نخواهیم اطرافیان او با نگاه به صفحه نمایش متوجه آن شوند را، بنویسد که در این حالت با فشار هر کلید صفحه کلید یک * در جعبه کلمه عبور نمایش داده می شود. جعبه کلمه عبور توسط مشخصه **type="password"** تعیین می شود.

```
مشاهده خروجي
<html>
<body>
<form>
Username:
<input type="text" name="user"><br>
Passw ord:
<input type="passw ord" name="pass">
</form>
</body>
</html>
```

دکمه

دکمه توسط **type="button"** مشخص می شود و نوشته روی آن توسط مشخصه **value="text"** معرفی خواهد شد. ساخت دکمه شیوه دیگری نیز دارد و آن استفاده از برچسب **<button>** است.

هنگامی که دکمه ها را فشار می دهیم بر اساس کاری که برای آنها تعریف کرده ایم عکس العمل نشان خواهند داد. دکمه ها کاربردهای مختلفی دارد که در مباحثی مانند **JavaScript** راجع به طریقه استفاده از آنها صحبت می شود.

مشاهده خروجی

```
<html>
<body>
<form>
<input
type="button" value="Button 1">
<br><br>
<button>Button 2</button>
</form>
</body>
</html>
```

جعبه drop down

این جعبه همان گونه که در شکل زیر می بینید فهرستی است که یکی از گزینه های آن قابل رویت است و با کلیک کردن روی آن بقیه گزینه ها نیز دیده می شوند. درضمن تنها یکی از موارد آن را می توانیم انتخاب کنیم. این جعبه توسط برچسب **<select>** مشخص می شود و موارد فهرست نیز توسط برچسب **<option>** ایجاد خواهند شد. به طور پیش فرض گزینه اول این فهرست انتخاب شده است و اگر بخواهیم که گزینه دیگری از فهرست انتخاب شده باشد باید کلمه **selected** را به برچسب **<option>** در گزینه مورد نظر اضافه کنیم. مثال زیر طریقه استفاده را مشخص می کند.

مشاهده خروجی

```
<html>
<body>
<form>
<select name="cars">
<option value="volvo">Volvo
<option value="saab">Saab
<option value="fiat">Fiat
<option value="audi">Audi
</select>
</form>
</body>
</html>
```

جعبه textarea

گای اوقات شما نیاز دارید که کاربر متنی طولانی تر از یک خط را بنویسد. بنابر این نمی توانید از جعبه متن استفاده کنید چون فقط در آن می توانید یک خط را بنویسید. اما توسط این جعبه شما می توانید به کاربر اجازه دهید که بیش از یک خط را بنویسد. این جعبه توسط برچسب **<textarea>** مشخص می شود و همانطور که در مثال زیر می بینید تعداد سطرها و ستونها توسط مشخصه های **rows** و **cols** تعیین می شوند.

مشاهده خروجی

```
<html>
<body>
<form>
<textarea name="text1" rows="5" cols="20">
The cat was playing in the garden.
</textarea>
</form>
</body>
</html>
```

فرم حاشیه دار

توسط برچسب **<fieldset>** شما می توانید یک حاشیه دور عناصر فرم خود ایجاد کنید و توسط برچسب **<legend>** می توانید کلماتی را روی قسمت بالایی حاشیه بنویسید.

مشاهده خروجی

```
<html>
<body>
<fieldset>
<legend>Health information:</legend>
<form>
Height <input type="text" size="3">
Weight <input type="text" size="3">
</form>
</fieldset>
</body>
</html>
```

در فرمها و دکمه ارسال action مشخصه

وقتی که کاربر بر دکمه **Submit** کلیک می کند، محتویات فرم به فایل یا آدرس دیگری فرستاده خواهد شد. مشخصه **action**، نام آن فایل یا آدرس را مشخص می کند، آن فایل معمولاً دارای ابزاری برای دریافت اطلاعات فرستاده می باشد. این ابزار در مباحث **ASP** و **CGI** توضیح داده می شوند. در مثال زیر شیوه ارسال اطلاعات توسط **method="get"** مشخص می شود، این شیوه ها مورد بحث **ASP** و **CGI** و ... می باشد.

مشاهده خروجی

```
<html>
<body>
<form name="input" action="action.asp" method="get">
Username:
<input type="text" name="user">
<input type="submit" value="Submit">
</form>
</body>
</html>
```

نکته

شما می توانید در فرم دکمه ای قرار دهید که کاربر با کلیک بر آن اطلاعات فرم را به صورت پیش فرض در آورد. این دکمه توسط عبارت **<input type="reset">** ایجاد می شود.

قابها (Frame) در HTML

توسط قابها شما می توانید بیش از یک فایل HTML را در یک پنجره مرورگر، به طور همزمان نمایش دهید. در این حالت هر پنجره HTML یک قاب نامیده می شود و هر قاب از بقیه مستقل است. مشکل استفاده از قاب این است که کار بر به سختی می تواند تمام صفحه شما را چاپ کند.

برچسب <frameset>

این برچسب مشخص می کند که پنجره مرورگر چگونه باید بین فایل های HTML یعنی همان قابها تقسیم شود. هر Frameset مجموعه ای از قابهای سطری یا ستونی را مشخص می کند. اندازه هر قاب سطری یا ستونی، مقدار فضایی از صفحه که آن قاب اشغال می کند را مشخص می کند.

برچسب <frame>

این برچسب فایل های HTML که درون هر قاب باید قرار بگیرند را مشخص می کند. در مثال زیر ما یک مجموعه از قاب با سه ستون داریم. ستون اول و سوم هرکدام 25% و ستون دوم 50% پهنای مرورگر را اشغال می کنند. فایل frame_a.htm در ستون اول، frame_b.htm در ستون دوم و frame_c.htm در ستون سوم قرار می گیرند.

مشاهده خروجی

```
<html>
<frameset cols="25%,50%,25%">
<frame src="frame_a.htm">
<frame src="frame_b.htm">
<frame src="frame_c.htm">
</frameset>
</html>
```

استفاده شود قابها به صورت افقی قرار خواهند گرفت، مثال زیر در **rows** از **cols** اگر به جای مشخصه همین رابطه است.

مشاهده خروجی

```
<html>
<frameset rows="25%,50%,25%">
  <frame src="frame_a.htm">
  <frame src="frame_b.htm">
  <frame src="frame_c.htm">
</frameset>
</html>
```

مستقل نیز قابل استفاده **HTML** هرکدام از سه فایل مثال بالا که در یک قاب قرار گرفتند به صورت یک فایل می باشند.

frame_c.htm frame_b.htm frame_a.htm

*نکته

اگر قابهای شما دارای حاشیه قابل رویت باشند، کاربر می تواند باکشاندن حاشیه ها اندازه قابهای شما را تغییر دهد. برای جلوگیری از این کار شما می توانید مشخصه **noresize** را به برچسب **<frame>** اضافه کنید.

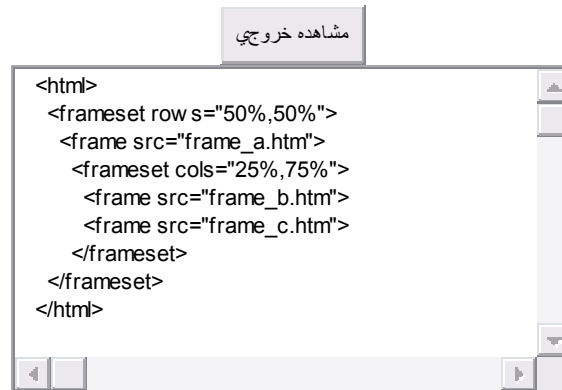
برچسب **<iframe>**

توسط این برچسب می توانید یک فایل **HTML** را در داخل یک قاب در هر کجای متن که بخواهید قرار دهید. البته بعضی از مرورگرهای قدیمی ممکن است برچسب **<iframe>** را پشتیبانی نکنند و در این صورت قاب در داخل متن دیده نمی شود. برای روشن تر شدن بحث به مثال زیر توجه کنید.

مشاهده خروجی

```
<html>
<body>
  <iframe src ="frame_c.htm">
</iframe>
  Some older browsers don't support iframes. If they
  be visible.
</body>
</html>
```

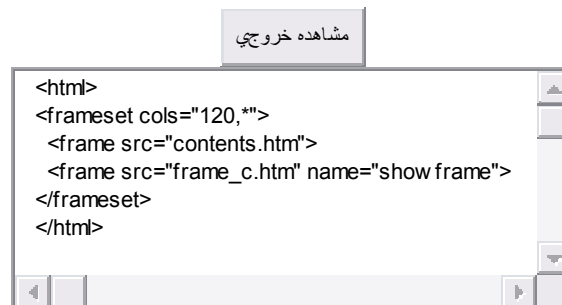
را در سطرها و ستون ها با هم ترکیب کرد **HTML** * مثال زیر نشان می دهد که چگونه می توان چند فایل و به شیوه دلخواه در قابها چید.



(بسازیم. قاب هدایت گر، شامل **navigator** * مثال زیر نشان می دهد که چگونه یک قاب هدایت گر (فهرستی از لینکهاست و قاب دوم، هدف می باشد یعنی فایلهایی که لینکها به آنها اشاره دارند در آن قاب باز خواهند شد. در اینجا فایلی با نام **contents.htm** دارای سه لینک می باشد و متن اصلی آن را در زیر می بینید. این فایل در قاب اول قرار می گیرد و در قاب دوم فایل **frame_c.htm** قرار می گیرد.

```
<a href = "frame_a.htm" target = "showframe">Frame a</a>
<br>
<a href = "frame_b.htm" target = "showframe">Frame b</a>
<br>
<a href = "frame_c.htm" target = "showframe">Frame c</a>
```

در زیر نتیجه را مشاهده می کنید : هنگامی که روی Frame a کلیک می کنید در قاب دیگر فایل frame_a.htm باز خواهد شد و به همین طریق دیگر لینکها نیز فایل مرتبط را در قاب دوم باز می کنند.



*نکته

بعضی از مرورگر های قدیمی برچسب **<frame>** را پشتیبانی نمی کنند که در این صورت نمی توانند قابهای شما را نمایش دهند و کاربر صفحه ای خالی را خواهد دید. برای رفع این مشکل می توانید از برچسب **<noframe>** استفاده کنید و اطلاعاتی را داخل آن بنویسید. در این حالت این گونه از کاربران اطلاعات داخل این برچسب را خواهند دید

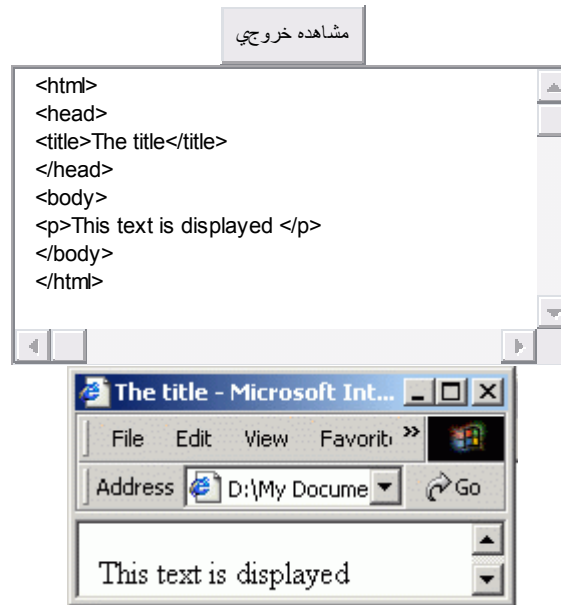
سربرگ HTML

برچسب <head>

این برچسب شامل اطلاعاتی عمومی درباره فایل **HTML** می باشد. عناصر داخل برچسب **<head>** توسط مرورگر نشان داده نمی شوند. بر اساسی استانداردهای **HTML** فقط تعداد کمی از برچسبها مجاز به قرار گرفتن در برچسب **<head>** می باشند و آنها عبارتند از: **<title>** و **<base>** و **<link>** و **<meta>** و **<style>** و **<script>**.

برچسب **<title>**

توسط این برچسب می توانیم عنوانی در نوار عنوان مرورگر بنویسیم. همانطور که در مثال زیر می بینید عبارت **The title** در نوار عنوان به نمایش در خواهد آمد. در ضمن این عبارت در متن مرورگر دیده نمی شود و فقط محتویات داخل برچسب پاراگراف دیده خواهد شد.



برچسب **<base>**

توسط این برچسب که معمولاً با مشخصه **target** همراه است می توان شیوه ارجاع به صفحه دیگری را در لینکها تعیین کرد. و دیگر نیازی به تعیین مجدد آن در برچسب **<a>** نمی باشد. مشخصه **target** به طور کامل در **مبحث لینک ها** توضیح داده شده است. به عنوان مثال اگر از دستور زیر

```
<head><base target="_blank"></head>
```

در فایل **HTML** خود استفاده کرده باشیم دو دستور زیر کاملاً مثل هم عمل خواهند کرد یعنی هر دو سایت **Yahoo** را در پنجره جدیدی باز خواهند کرد.

```
<a href="http://www.yahoo.com" target="_blank">
This link</a>
<a href="http://www.yahoo.com">This link</a>
```

برچسب **<meta>**

هدف استفاده از برچسب **<meta>** تدارک اطلاعاتی اضافی درباره فایل می باشد. و اغلب شامل اطلاعاتی می باشد که مربوط به مرورگر و موتورهای جستجو است مانند شرحی از محتویات فایل **HTML**.

معمولا همراه این برچسب از دو مشخصه **name** و **content** استفاده می شود. بعضی از موتورهای جستجو از این اطلاعات برای فهرست بندی صفحه شما استفاده می کنند. از آنجا که بعضی از طراحان و مدیران سایت های وب برای اینکه سایتشان مقام بهتری در موتورهای جستجو بدست آورد گاهی کلمات کلیدی را چندین بار تکرار می کردند و به عبارتی در رتبه بندی واقعی موتورهای جستجو اختلال ایجاد می کردند، به همین دلیل بعضی از موتورهای جستجو از استفاده اطلاعاتی که توسط برچسب **meta** تدارک دیده می شود خودداری می کنند. ولی به هر حال شما استفاده از این برچسب را فراموش نکنید و استفاده از آن را در برنامه کاری خود قرار دهید.

دستور زیر توضیحی درباره صفحه شما به فایل **HTML** اضافه می کند.

```
<meta name="description"
content="This page is a tutorials on HTML.">
```

دستور زیر کلمات کلیدی صفحه شما را به فایل **HTML** اضافه می کند.

```
<meta name="keywords"
content="HTML, CSS, JavaScript, VBScript">
```

برچسب **meta** کاربرد های دیگری نیز دارد که به در جای خود راجع به آنها صحبت خواهیم کرد. در اینجا به ذکر یکی از آنها می پردازیم. فرض کنید شما قبلا یک سایت وب با نام **youname.com** داشتید و بنا به دلایلی می خواهید از نام جدید **yournewname.com** استفاده کنید درضمن محتویات هر دو سایت مثل هم می باشد در این صورت نیازی به صرف هزینه اضافه برای هر دو سایت ندارید و کافی است کاربرانی را که به آدرس قبلی شما مراجعه می کنند را به آدرس جدید هدایت کنید و برای این کار می توانید از برچسب **meta** و از دو مشخصه **http-equiv** و **content** به شیوه زیر استفاده کنید :

```
<meta http-equiv="Refresh"
content="5;url=http://www.yournewname.com">
```

عدد **5** که در **content** مشاهده می کنید مدت زمانی است طول می کشد تا کاربر به آدرس جدید منتقل شود و این زمان بر حسب ثانیه می باشد؛ که اگر از عدد **0** استفاده کاربر فوراً به آدرس جدید منتقل شود. در ضمن بهتر است لینکی در فایل خود قرار دهید تا اگر کاربر به طور خودکار به آدرس جدید منتقل نشد بتواند با کلیک بر لینک موجود به آدرس جدید برود :

```
<a href="http://www.yournewname.com">
http://www.yournewname.com</a>
```

برچسب <link>

این برچسب بیشتر مورد استفاده **CSS** می باشد. **CSS** که مخفف **Cascading Style Sheets** می باشد خود مبحثی گسترده است و در دوره های آموزشی بعدی مفصلا به آن خواهیم پرداخت لذا در اینجا فقط اشاره ای مختصر به آن می کنیم.

برای اضافه کردن شیوه های نگارش به تعداد زیادی از صفحات استفاده از یک برگه شیوه نگارش خارجی کار ایده آلی می باشد. یعنی استفاده از یک فایل **CSS** که تنها با ویرایش آن می توانید ظاهر کلیه صفحات خود را تغییر دهید. هر صفحه باید توسط برچسب **<link>** (که در سربرگ صفحه قرار می گیرد یعنی داخل برچسب **<head>**) به فایل **CSS** که شیوه های نگارش در آن تعیین شده اند، متصل شود.

```
<head>
<link rel="stylesheet"
type="text/css" href="mystyle.css" >
</head>
```

در مثال فوق مرورگر شیوه های نگارش تعیین شده را از فایل **mystyle.css** خواهد خواند و متن فایل را بر اساس آنها قالببندی می کند. یک فایل **CSS** توسط هر ویرایشگر ساده متن از مانند برنامه **notepad** قابل نوشتن است و این فایل نباید هیچ یک از برچسب های **HTML** را داشته باشد و فقط باید شامل دستورات **CSS** باشد و بهتر است با پسوند **.css** ذخیره شود. در مثال زیر یک فایل **CSS** نشان داده شده است :

```
hr {color: sienna}
p {margin-left: 20px}
body {background-image: url("images/back40.gif")}
```

برچسب <style>

این برچسب نیز مورد استفاده **CSS** می باشد. به جای آنکه دستورات **CSS** در فایلی جدا قرار داده شوند می توانند در داخل فایل **HTML** ما قرار گیرند. شیوه نگارش داخلی را در داخل سربرگ و توسط برچسب **<style>** تعیین می کنیم. مانند مثلا زیر :

```
<head>
<style type="text/css">
hr {color: sienna}
p {margin-left: 20px}
body {background-image: url("images/back40.gif")}
</style>
</head>
```

حال مرورگر شیوه های نگارش تعیین شده را می خواند و متن فایل را براساس آنها قالببندی می کند.

برچسب <script>

از این برچسب زمانی استفاده می کنیم که بخواهیم دستورات **vbscript** یا **javascript** و ... را در فایل **HTML** خود به کار ببریم، یا این دستورات را از فایلی دیگر بخوانیم.

*به عنوان مثال دستور زیر کدهای **java** را از فایل **MyScripts.js** فراخوانی می کند.

```
<script language="javascript" src="MyScripts.js">
</script>
```

مثال زیر کدهای **java** را در **HTML** ما می نویسد که توسط مرورگر اجرا خواهد شد.

```
<script type="text/javascript">
document.write("Hello World!")
</script>
Hello World!
```