

به نام خداوند بخشنده مهربان

آموزش طراحی صفحات وب

HTML 4.01



تالیف و گردآوری

مهرداد فتاحی

Developer Studio Network

برای دریافت اطلاعات کاملتر و آموزش سایر مباحث طراحی وب به وب سایت ما سر بزنید

www.Developer1.ir

www.facebook.com/Developer1.ir

مشخصات کتاب

نام کتاب : آموزش HTML 4 همراه با مثال های عملی

نویسنده : مهرداد فتاحی

ناشر : www.Developer1.ir

سال : 1391

کپی رایت : چاپ و انتشار این کتاب در هر جای وب و دنیا با پاسداشت کپی رایت و حق نویسنده آن رایگان است . از اینکه کتاب ما را در سرار وب برای عموم قرار می دهید ، پیشاپیش سپاسگزاریم .

تمام مطالب این کتاب همراه با توضیحات و مطالب و مثال های آموزشی بیشتر ، بر روی پرتال آموزش HTML در سایت ما قرار دارد .

درباره سایت ما

سایت www.Developer1.ir یک سایت آموزشی رایگان در زمینه آموزش کامپیوتر ، طراحی وب و برنامه نویسی موبایل است . برای دریافت آموزش های بیشتر در مورد سایر مباحث به وب سایت ما سر بزنید .

در این سایت آموزش `HTML` ، `CSS` ، `Java Script` ، `PHP` ، `ASP.Net` ، `XML` ، `jQuery` ، `SQL` ، `MySQL` مجموعه آفیس 2007 به صورت تصویری قرار داده شده است .

همچنین برنامه نویسی اندروید و ویندوز فون 8 نیز جز برنامه های آینده می باشد .

مقدمه و معرفی زبان HTML

معرفی زبان HTML :

عبارت HTML مخفف (Hyper Text Markup Language) است. HTML زبان استاندارد طراحی صفحات وب است و کلیه کدهای صفحه اعم از طرف سرور و طرف مشتری در نهایت به کدهای HTML تبدیل شده و توسط مرورگر نمایش داده می شوند.

HTML یک زبان نشانه گذاری است، به این معنی که بخش های مختلف توسط اجزایی به نام تگ از هم جدا شده، که هر کدام دارای کاربرد و خواص مربوط خود هستند. این تگ ها به مرورگر اعلام می کنند که هر بخش از صفحه چه نوع عنصری است و باید به چه صورت نمایش داده شود.

در یک صفحه HTML می توان انواع عناصر از قبیل متن، تیتر، عکس، جدول و ... را قرار داد، که برای هر عنصر باید از تگ مربوط به آن استفاده کرد. صفحات HTML فقط از کدها که به صورت متن هستند تشکیل شده اند.

مفهوم تگ های HTML :

تگ های HTML برای نشانه گذاری محتویات صفحات به کار می روند و باعث می شوند که مرورگر بتواند تشخیص دهد هر بخش چه نوع عنصری است. هر تگ HTML، یک بخش ابتدایی و یک بخش انتهایی دارد که هم نام بوده و به صورت استاندارد طبق شکل کلی زیر به کار می روند :

< تگ ابتدا > محتویات < / تگ انتها >

< /label > < label > : مثال

تگ ابتدایی مشخص کننده آغاز تگ و تگ پایانی، مشخص کننده انتهای تگ است. هر تگ HTML، باید حتما در ادامه توسط تگ پایانی بسته شود. کلیه نوشته ها و تگ های دیگری که در بین تگ ابتدا و پایان نوشته می شوند، محتویات تگ را تشکیل می دهند.

خواص تگ های HTML :

هر تگ دارای مجموعه از خواص است که ویژگی های مختلف آنها را تعیین می کند. هر یک از این خواص را می توان در درون تگ ابتدایی معرفی و مقدار دهی کرد.

راهنمایی: در ادامه هر یک از تگ های HTML معرفی شده و در صفحه مربوط به آنها خواص مربوطه نیز در جدول خواص مهم قرار داده شده اند. در ستون نام خاصیت نام آن و در ستون نوع خاصیت انواع مقادیر قابل قبول برای آن خاصیت تعیین شده است.

مثال: برای مثال تگ `< table >` دارای خاصیتی به نام Border است، که به دور جدول کادر یا حاشیه ایجاد می کند. نحوه تعریف و مقدار دهی آن به صورت زیر است :

توجه: در هنگام تعریف باید بین خاصیت و مقدار آن علامت = قرار داده و همچنین مقادیرها باید درون علامت " " قرار بگیرند.

< table border = "1" > محتویات < /table >

راهنمایی 2: در بخش آموزش تگ ها سعی شده تا با ارائه مثال های مناسب مطالب توضیحی را در عمل نشان داد . برای این منظور از جدول Example استفاده شده است ، که این جدول دارای 2 قسمت است . در قسمت کد ، کد واقعی مورد نیاز برای هدف ارائه شده و در قسمت خروجی نیز خروجی کد که به صورت اکتیو توسط مرورگر تولید شده ، نشان داده شده است .
توجه: تمام مثال های قسمت آموزش HTML بر طبق استانداردهای نرم افزار Microsoft Visual Studio 2005 طراحی شده و توسط مرورگرهای مطرح Internet Explorer و FireFox تست شده است .

نکات مهم :

1. تگ های HTML نسبت به بزرگ و کوچک بودن حروف حساس نیستند ، یعنی تگ های < b > با < B > برابر هستند .
2. می توان یک تگ را در تگ ابتدایی نیز بست . برای این کار از یک علامت / بعد از نام تگ و خواص مورد نظر تعریف شده استفاده می کنیم . در این حالت نمی توان متن یا تگ دیگری را به عنوان محتویات برای آن تگ در نظر گرفت .
توجه کاربرد این روش در مورد تگ هایی مثل < table > , < td > , < tr > و ... امکان پذیر نیست .

Example :

< br / >

< input type = "text" name = "family" ... / >

3. هر تگ HTML محتویات درون خود را تحت تاثیر قرار می دهد . برای مثال تگ < b > برای توپر کردن (bold) نوشته به کار می رود و چنانچه متنی بین تگ باز و بسته < b > قرار بگیرد ، توپر می شود .

خصوصیات استاندارد تگ های HTML

در این قسمت خصوصیات استاندارد تگ های HTML معرفی شده اند . این خصوصیات در اکثر تگ های HTML مشترک بوده و دارای کاربرد یکسانی هستند .
این خصوصیات به دلیل مشترک بودن ، فقط در این قسمت ذکر شده و از تکرار آنها در صفحات دیگر خودداری شده است .

خصوصیات اصلی

نکته : این خاصیت ها در تگ های base , title , style , param , meta , html , head و script وجود ندارد .

نام خاصیت	نوع خاصیت	شرح
Class	نام کلاس عنصر	مشخص کننده کلاس تگ در کد نویسی برنامه است . این کلاس در کدنویسی CSS و زبان های اسکریپتی کاربرد دارد .
Id	id عنصر	مشخص کننده یک شناسه منحصر به فرد برای تگ در درون سند HTML است . id هر تگ در کدنویسی CSS و زبان های اسکریپتی کاربرد دارد . توسط id می توان به تگ مورد نظر دست یافت .
Style		یک خاصیت چند مفداری است که خصوصیات قالب دهی و اعمال سبک ها (CSS) را برای تگ مورد نظر مشخص می کند .
Title	text متن	متنی است که به صورت tooltip در یک کادر زرد رنگ ، در هنگام قرار گرفتن موس بر روی عنصر نمایش داده می شود .

خصوصیات زبان

نکته : این خاصیت ها در تگ های br , base , param , iframe , hr , script وجود ندارد .

نام خاصیت	نوع خاصیت	شرح
Dir	rtl ltr	تعیین کننده جهت قرار گرفتن نوشته را تعیین می کند ، که یکی از 2 حالت زیر را می تواند داشته باشد : ltr : چپ به راست rtl : راست به چپ
Language	نام زبان	مشخص کننده زبان برنامه نویسی کد مربوط به تگ است .

خصوصیات صفحه کلید

شرح	نوع خاصیت	نام خاصیت
یک میانبر صفحه کلید برای دستیابی به عنصر است .	کلید میانبر	Accesskey
شماره ترتیبی قرار گرفتن فوکوس صفحه بر روی عنصر مورد نظر را در هنگام فشردن کلید tab مشخص می کند .	عدد number	TabIndex

تگ های مقدماتی HTML

تگ < html > :

محدوده کد صفحه HTML را مشخص می کند و به مرورگر می فهماند که این یک صفحه HTML است . فقط کدهای درون این محدوده توسط مرورگر اجرا می شود .

مثال : در مثال زیر بدنه اصلی یک صفحه با تگ < html > مشخص شده است :

Example	
کد	
<pre>< html > < head > < title > The title of the page < /title > < /head > < body > ... The content of the page ... < /body > < /html ></pre>	<p>html محدوده اصلی صفحه</p> <p>head تگ سر عنوان</p> <p>تگ عنوان صفحه</p> <p>بدنه اصلی صفحه</p> <p>...</p> <p>The content of the page</p> <p>...</p> <p>پایان بدنه اصلی صفحه</p> <p>پایان محدوده اصلی صفحه</p>

تگ < body > :

این تگ بدنه اصلی صفحه HTML را معرفی می کند و شامل کلیه محتویات قابل نمایش در صفحه از قبیل نوشته ها ، عکس ها ، لینک ها و ... است . فقط محتویات این قسمت توسط مرورگر به عنوان خروجی نمایش داده می شوند . کدهای محتویات خروجی بین تگ باز و بسته < body > تعریف می شوند .

خواص مهم		
نام خاصیت	نوع خاصیت	شرح
alink	نام رنگ rgb(x,x,x) #xxxxxx	مشخص کننده رنگ لینک های فعال در صفحه است . رنگ میتواند به روش های زیر تعیین شود : 1. نام رنگ مثل blue یا red .

<p>2. تعیین رنگ به وسیله تابع rgb به صورت زیر :</p> <p>(مقدار رنگ قرمز ، مقدار رنگ سبز ، مقدار رنگ آبی) rgb</p> <p>این تابع مقدار 3 رنگ را با هم ترکیب کرده و یک رنگ را ایجاد می کند . که مقدار رنگ توسط عددی بین 0 تا 255 تعیین می شود ، که هر چه عدد بزرگتر باشد میزان آن رنگ در کل رنگ بیشتر خواهد بود .</p> <p>Example : rgb (10,65,232)</p> <p>3. تعیین رنگ به صورت عددی ترکیبی در مبنای 16 هگزادسیمال :</p> <p>میزان رنگ آبی میزان رنگ سبز میزان رنگ قرمز #</p> <p>Example : #08FF00</p>		
<p>آدرس عکس یا فایلی که به عنوان پس زمینه صفحه مورد استفاده قرار می گیرد را تعیین می کند .</p>	<p>URL نام - مسیر فایل</p>	<p>background</p>
<p>می توان توسط این خاصیت یک رنگ را به عنوان رنگ پس زمینه صفحه قرار داد .</p>	<p>نام رنگ rgb(x,x,x) #xxxxxx</p>	<p>bgcolor</p>
<p>مشخص کننده رنگ کلیه لینک های صفحه است .</p>	<p>نام رنگ rgb(x,x,x) #xxxxxx</p>	<p>link</p>
<p>رنگ پیش فرض نوشته های درون صفحه را تعیین می کند که به صورت استاندارد مشکی است .</p>	<p>نام رنگ rgb(x,x,x) #xxxxxx</p>	<p>text</p>
<p>رنگ لینک هایی در صفحه که قبلا یکبار کلیک شده اند را مشخص می کند و به این منظور استفاده می شود که کاربر متوجه شود آن لینک را قبلا مشاهده کرده است .</p>	<p>نام رنگ rgb(x,x,x) #xxxxxx</p>	<p>vlink</p>

Example

```
< html >
< head >
  < title > The title of the page < /title >
< /head >
< body >
  ...
  The content of the page  محتویات قابل نمایش توسط مرورگر
  ...
< /body >
< /html >
```

تگ < head > :

تگ < head > در بر گیرنده اطلاعات کلی درباره سند وب است که این اطلاعات Meta-Information به معنای (اطلاعات درباره) نامیده می شود .

کلیه اطلاعات درون تگ < head > توسط مرورگر نمایش داده نمی شوند .براساس استاندارد HTML فقط تگ های زیر قابل استفاده در بخش head هستند :

[< script >](#) , [< style >](#) , [< title >](#) , [< meta >](#) , [< link >](#) , [< base >](#)

خواص مهم		
شرح	نوع خاصیت	نام خاصیت
لیستی از آدرس فایل ها یا صفحاتی که در بر گیرنده اطلاعات کلی (meta-information) راجع به محتویات صفحه هستند ، را شامل می شود .	URL مسیر	profile

تگ < title > :

این تگ عنوان صفحه را که مرورگر در هنگام باز کردن صفحه در نوار عنوان نمایش می دهد را مشخص می کند . متنی که بین تگ باز وبسته < title > قرار بگیرد عنوان صفحه را تعیین می کند . تگ < title > باید در قسمت head صفحه قرار بگیرد . عنوان صفحه ارتباطی به نام فایل صفحه ندارد و بهتر است حداکثر تا 15 حرف باشد .

Example
<pre>< html > < head > < title > This is the title of the page عنوان صفحه < /title > < /head > < body > ... < /body > < /html ></pre>

تگ <!Doctype> :

این تگ اولین سطر از کد صفحه وب است و به مرورگر نوع و نسخه استفاده شده از زبان برنامه نویسی HTML یا XHTML را اعلام می کند . لازم به ذکر است که هر صفحه فقط می تواند یک نوع Doctype داشته باشد .

نکته : تگ < Doctype > تنها تگی است که تگ انتهایی ندارد و در تگ ابتدایی هم توسط علامت / بسته نمی شود .

انواع ورژن های قابل استفاده در صفحات وب :

1) HTML :

زبان HTML سه حالت کلی دارد :

Strict , Traditional , Frameset

کدهای زیر باید در قسمت < Doctype > جهت استفاده از هر یک از حالت های فوق تایپ شود :

HTML Strict DTD	نام
زمانی مورد استفاده قرار می گیرد که طراح بخواهد از شیوه قالب دهی به صفحات (CSS) استفاده نماید .	شرح
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd" >	کد

HTML Traditional DTD	نام
زمانی مورد استفاده قرار می گیرد که طراح می خواهد از قابلیت های جدید HTML استفاده نکند ، زیرا مرورگر بازدید کنندگان سایت ممکن است از شیوه قالب دهی صفحات (CSS) پشتیبانی نکند .	شرح
< !DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd" >	کد

HTML Frameset DTD	نام
زمانی مورد استفاده قرار می گیرد که بخواهیم در صفحات از قاب ها (Frames) استفاده کنیم .	شرح

<pre>< !DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN" "http://www.w3.org/TR/html4/frameset.dtd" ></pre>	<p>کد</p>
--	-----------

2) XHTML :

زبان XHTML سه حالت کلی دارد :

Strict , Traditional , Frameset

کدهای زیر باید در قسمت < Doctype ! > جهت استفاده از هر یک از حالت های فوق تایپ شود :

<p>XHTML Strict DTD</p>	<p>نام</p>
<p>زمانی مورد استفاده قرار می گیرد که طراح بخواهد از شیوه قالب دهی به صفحات (CSS) استفاده نماید .</p>	<p>شرح</p>
<pre><!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd" ></pre>	<p>کد</p>

<p>XHTML Traditional DTD</p>	<p>نام</p>
<p>زمانی مورد استفاده قرار می گیرد که طراح می خواهد از قابلیت های جدید XHTML استفاده نکند ، زیرا مرورگر بازدید کنندگان سایت ممکن است از شیوه قالب دهی صفحات (CSS) پشتیبانی نکند .</p>	<p>شرح</p>
<pre>< !DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd" ></pre>	<p>کد</p>

تگ های مقدماتی HTML

تگ < meta > :

این تگ اطلاعات کلی درباره مطالب درون صفحه وب ، توضیحات ضروری و واژه های کلیدی مرتبط با موضوع صفحه جهت استفاده موتورهای جستجو را شامل می شود .

چیدمان مناسب و استفاده از کلید واژه های مرتبط به موضوع صفحه ، باعث بهتر نمایش داده شدن صفحه و سایت در قسمت جستجوی موتورهای جستجو می شود .

-تعیین کلیدواژه ها برای موتورهای جستجو:

موتورهای جستجو ، هنگام جستجو برای یافتن مطالب در مورد یک موضوع خاص ، آن مطلب را با کلیدواژه های تعریف شده در سایت مقایسه کرده و در صورت یکسان بودن ، صفحه وب سایت شما را در لیست نتایج جستجو نمایش می دهند .

```
< meta name = "keywords" content="Html,CSS,XML,Java Script,VB Script" />
```

-تعیین یک توضیح درباره مطالب صفحه:

```
< meta name = "discription" content="Free web teaching on Html,CSS,XML,Java Script and VB script" />
```

-تعیین آخرین زمان بازبینی و به روز کردن صفحه:

```
< meta name = "rivised" content="6/11/07" />
```

: شدن صفحه برای هر 5 ثانیه یکبار refresh تعیین زمان -

```
< meta http - equiv = "refresh" content="5" />
```

خواص مهم		
شرح	نوع خاصیت	نام خاصیت
اطلاعات مربوط به خاصیت تعیین شده در قسمت name یا http-equiv را نگهداری می کند .	متن Text	content
اطلاعات درون content خود را به سرور انتقال می دهد .	content-type	http - equiv
اطلاعات درون content خود را به نام تعیین شده ارتباط می دهد که می تواند یکی از حالت های زیر را داشته باشد : - author : نام نویسنده یا گروه نویسندگان و طراحان صفحه وب را مشخص می کند . - description : توضیح مختصری راجع به محتویات درون صفحه ارائه می دهد . - keywords : کلیدواژه های مورد استفاده موتورهای جستجو را تعیین می کند . - generator : مشخصاتی راجع به نرم افزاری که صفحه با آن طراحی و تولید شده است را ارائه می دهد . - revised : برای اعمال آخرین تاریخ به روز رسانی سایت استفاده می شود . - others : برای ذرچ سایر اطلاعات مورد نظر استفاده می شود .	author description keywords generator revised others	name
قالبی را برای تفسیر و اجرای اطلاعات درون خاصیت content مشخص میکند .	متن Text	scheme

تگ توضیح <!-- ... --> :

از این تگ برای وارد کردن توضیحات درون بخش کدنویسی صفحه وب استفاده می شود . محتویات درون این تگ به طور کامل توسط مرورگر نادیده گرفته شده و بروی صفحه نمایش داده نمی شوند .
از توضیحات می توان برای تشریح کدهای صفحه برای افراد دیگر و یا علامت گذاری آنها برای فهم بهتر و ویرایش در آینده استفاده کرد .

نکته : از این تگ همچنین می توان برای مخفی کردن اسکریپت ها از دید مرورگرهای قدیمی که امکان اجرای اسکریپت ها را ندارند، نیز استفاده کرد . مرورگرهای جدید به طور هوشمند محتویات درون این نوع تگ ها را بررسی کرده و چنانچه حاوی کدهای اسکریپت باشند آنها را اجرا کرده و در غیر این صورت آنها را نمایش نمی دهند .

Example	
کد	خروجی
<pre><!-- This text will not display -- > < p > This text will display < /p ></pre>	<p>This text will display</p>

تگ < base > :

این تگ یک آدرس پایه برای کلیه لینک های موجود در صفحه تعیین می کند .
مثال : فرض کنید که آدرس یک عکس در صفحه به صورت زیر است :

```
< img src="http://www.DeveloperStudio.ir/images/smile.gif" / >
```

در صورتی که تگ < base > در قسمت head به صورت زیر وارد شود :

```
< head >  
< base href="http://www.DeveloperStudio.ir/images/" >  
< /head >
```

در هنگام وارد کردن عکس فوق در صفحه ، می توان برای وارد کردن آدرس عکس فقط ادامه مقدار آدرس عکس را در تگ > < img به صورت زیر وارد کرد :

```
< img src="smile.gif" />
```


خواص مهم		
شرح	نوع خاصیت	نام خاصیت
مشخص کننده آدرس پیش فرض برای کلیه لینک های صفحه است .	URL	href
مقصد باز شدن پنجره کلیه لینک های صفحه را تعیین می کند . چنانچه درون تگ یک لینک ، مقدار این خاصیت به صورت جدا تعیین شود این قاعده برای آن لینک زیر پا گذاشته می شود . این خاصیت میتواند یکی از مقادیر زیر را داشته باشد : - blank : تمام لینک ها در یک پنجره جدید باز خواهند شد . - self : تمام لینک ها در همان پنجره یا قابی که کلیک شده اند باز خواهند شد . - parent- : تمام لینک ها در قاب اصلی یا مادر باز خواهند شد . - top- : تمام لینک ها در همان پنجره ای که کلیک شده اند به طور کامل باز خواهند شد .	blank self parent top	target

تگ های قالب بندی متن

تگ < p > :

برای ایجاد یک پاراگراف جدید در صفحه به کار می رود . هر متنی که در بین تگ باز و بسته < p > قرار بگیرد ، محتویات آن پاراگراف را تشکیل می دهد . با ایجاد یک پاراگراف جدید ، نوشته به ابتدای خط بعدی انتقال می یابد .

خواص مهم		
شرح	نوع خاصیت	نام خاصیت
تراز بندی افقی نوشته را در پاراگراف تعیین می کند ، که می تولند یکی از حالت های زیر را داشته باشد : left- : نوشته را از سمت چپ تراز می کند . right- : نوشته را از سمت راست تراز می کند . center- : نوشته در وسط صفحه یا خانه جدول تراز می شود . justify- : نوشته به اندازه ای که کل عرض صفحه یا خانه جدول را پر کند ، کش می آید .	left right center justify	align

Example	
کد	خروجی
< p > This is a sample paragraph < /p > < p > This is another paragraph < /p >	This is a sample paragraph This is another paragraph

تگ < h1 > - < h6 > :

این تگ معمولا برای ایجاد تیترها و عنوان ها مورد استفاده قرار می گیرد . متنی که قرار است به عنوان تیتر نمایش داده شود ، بین تگ باز و بسته قرار می گیرد .
< h1 > بزرگترین سایز نوشته را ارائه داده و با بزرگتر شدن عدد بعد از h ، سایز نوشته آن کاهش می یابد و < h6 > کوچکترین سایز نوشته را ارائه می دهد .

خواص مهم		
شرح	نوع خاصیت	نام خاصیت
<p>تراز بندی افقی نوشته را برای متن عنوان تعیین می کند که می تولند یکی از حالت های زیر را داشته باشد :</p> <p>left- : نوشته را از سمت چپ تراز می کند .</p> <p>right- : نوشته را از سمت راست تراز می کند .</p> <p>center- : نوشته در وسط صفحه یا خانه جدول تراز می شود .</p> <p>justify- : نوشته به اندازه ای که کل عرض صفحه یا خانه جدول را پر کند ، کش می آید .</p>	<p>left</p> <p>right</p> <p>center</p> <p>justify</p>	<p>align</p>

Example	
کد	خروجی
<pre>< h1 > This is header 1 < /h1 ></pre>	This is header 1
<pre>< h2 > This is header 2 < /h2 ></pre>	This is header 2
<pre>< h3 > This is header 3 < /h3 ></pre>	This is header 3
<pre>< h4 > This is header 4 < < /h4 ></pre>	This is header 4
<pre>< h5 > This is header 5 < /h5 ></pre>	This is header 5
<pre>< h6 > This is header 6 < /h6 ></pre>	This is header 6

تگ < br > :

به کاربردن این تگ باعث ایجاد یک خط جدید و انتقال نوشته به اول سطر پایینی می شود . همچنین به وسیله آن می توان یک خط خالی ایجاد کرد .

نکته : تگ < br > دارای تگ پایانی نیست و ترکیب < br > ... < br / > غلط است . این تگ باید به وسیله یک علامت / در درون تگ ابتدایی بسته شود .

Example	
کد	خروجی
< p > My name is : < br / > Mehrdad < /p >	My name is : Mehradad

تگ <hr> :

به کاربردن این تگ باعث ایجاد یک خط افقی درون صفحه می شود ، که پهنای آن همواره کمی کمتر از عنصر در برگیرنده آن (مثل خانه جدول یا صفحه) است . به طور مثال در بالای تیتتر تگ < hr > ، یک خط افقی قرار داده ایم .
نکته : این تگ نیز همانند تگ < br > دارای تگ پایانی نیست و باید در تگ ابتدایی بسته شود .

Example	
کد	خروجی
This is a line < hr / > ok!	This is a line <hr/> ok!

تگ :

باعث توپر شدن متنی که درون تگ باز و بسته قرار بگیرد می شود .

Example	
کد	خروجی
< b > This is Bold Text < /b >	This is Bold Text

تگ <u> :

باعث می شود تا متنی که بین تگ باز و بسته <u> قرار بگیرد ، به صورت خط زیر دار نمایش داده شود .

Example	
کد	خروجی
< u > This is <u>underline</u> Text < /u >	<u>This is underline Text</u>

تگ <i> , :

باعث می شود تا متن درون آن به صورت کج (italic) نمایش داده شود .

Example	
کد	خروجی
< i > This is italic Text < /i >	<i>This is italic Text</i>
< em > This is italic Text < /em >	<i>This is italic Text</i>

تگ <big> :

این تگ باعث نمایش بزرگتر محتویات درون آن نسبت به متن خطوط همجوار می شود .

Example	
کد	خروجی
My name is < big > Mehrdad < /big >	My name is Mehrdad

تگ < small > :

این تگ باعث کوچکتر نمایش دادن محتویات درون آن نسبت به متن خطوط همجوار در صفحه می شود .

Example	
کد	خروجی
My name is < small > Mehrdad < /small >	My name is Mehrdad

تگ < strong > :

این تگ باعث نمایش توپر و بزرگتر محتویات درون خود نسبت به متن خطوط همجوار آن می شود .

Example	
کد	خروجی
My name is < strong > Mehrdad < /strong >	My name is Mehrdad

تگ < cite > :

این تگ باعث نمایش محتویات درون خود به صورت کج و متمایز نسبت به متن خطوط همجوار آن می شود .

Example	
کد	خروجی
My name is < cite > Mehrdad < /cite >	My name is Mehrdad

تگ < address > :

از این تگ معمولا برای نمایش آدرس ها در صفحه های وب استفاده می شود . خصوصیت مهم تگ address این است که متن درون خود را به صورت کج (italic) نمایش داده و باعث ایجاد یک خط جدید در ابتدا و انتهای آدرس می شود .

Example	
کد	خروجی
<pre>< address > Washington DC <br / > New York <br / > 214 Ave. < /address ></pre>	<p>Wshington DC New York 214 Ave.</p>

تگ < pre > :

چنانچه در محیط کد نویسی صفحات وب ، بین متن های نوشته شده با کلید space فاصله ایجاد کرده یا با کلید Enter به خط بعدی بروید ، مرورگر این مسئله را نادیده گرفته ، تمام فاصله ها را از بین برده و حداکثر به اندازه یک کاراکتر فاصله ایجاد می کند . برای جلوگیری از این مسئله ، از تگ <pre> استفاده می کنیم . این تگ باعث نمایش متن درون خود با همان شکل و ساختاری که در محیط کد نویسی صفحه وارد شده است ، می شود . همانطور که قبلا ذکر شد ، برای ایجاد فاصله بین حروف از کاراکتر و رفتن به خط بعدی از تگ
 استفاده می کنیم .

Example	
کد	خروجی
<pre>< pre > This is a pre formatted paragraph it does not have any space or linebrake < / pre ></pre>	<p>this is a pre formatted paragraph it does not have any space or linebrake</p>

< p > this is a usual paragraph
IE doesn't pay attention to line brakes
and spaces it shows it like this < /p >

this is a usual paragraph IE doesn't pay attention to line
brakes and spaces it shows it like this

تگ های کد کامپیوتر

تگ های کد کامپیوتر :

از این گروه تگ برای نمایش نوشته های مورد نظر همانند نمایش در محیط های برنامه نویسی استفاده می شوند . این حالت برای استفاده در صفحات آموزش کد نویسی و یا شبیه سازی محیط های برنامه نویسی مناسب هستند .

تگ `< code >` :

این تگ نوشته را به صورت نوشته های درون محیط های برنامه نویسی قدیمی تبدیل می کند .

Example	
کد	خروجی
<code>< code > This is a sample code < /code ></code>	This is a sample code

تگ `< kbd >` :

این تگ نیز حالت خاصی مثل حروف دستگاه تایپ به متن می دهد .

Example	
کد	خروجی
<code>< kbd > This text is like a type writer < /kbd ></code>	This text is like a type writer

تگ `< samp >` :

این تگ نیز حالت خاصی مثل نوشته های درون محیط های برنامه نویسی به متن می دهد .

Example	
کد	خروجی

<code>< samp > This text is sample text < /samp ></code>	This text is sample text
--	--------------------------

تگ `< tt >`:

این تگ نیز حالت خاصی از نوشته مثل حروف تلگراف به متن می دهد .

Example	
کد	خروجی
<code>< tt > This text is telegraph text < /tt ></code>	This text is telegraph text

تگ `< var >`:

این تگ حالتی برای تعریف متغیرهای برنامه نویسی ایجاد می کند .

Example	
کد	خروجی
<code>< var > This is a variable < /var ></code>	This is a variable

تگ های قالب بندی متن

تگ < sup > , < sub > :

از این تگ ها برای ایجاد نوشته های زیر نویس یا اندیس دار استفاده می شود .

Example	
کد	خروجی
This is a text < sub > example < /sub >	This is a text <small>example</small>
This is a text < sup > example < /sup >	This is a text ^{example}

تگ < abbr > :

از این تگ معمولا برای نمایش یک متن کوتاه و خلاصه شده برای یک واژه یا عبارت بزرگتر استفاده می شود . به طور مثال در انگلیسی به جای عبارت et cetra از واژه etc استفاده می کنند .
ویژگی مهم تگ < abbr > این است که می تواند یک مقدار توسط خاصیت title خود گرفته و در هنگام قرار گیری موس بر روی واژه خلاصه شده ، آن نوشته در یک کادر زرد رنگ نمایش دهد .

Example	
موس را برای چند لحظه بر روی واژه etc برده و صبر کنید .	
کد	خروجی
< abbr title ="et cetra" > etc < /abbr >	etc

تگ < acronym > :

این تگ نیز دقیقا عملکردی همانند تگ < abbr > دارد و برای ایجاد یک عنوان مخفف برای یک عبارت بزرگتر استفاده می شود .

Example	
موس را برای چند لحظه بر روی واژه WWW برده و صبر کنید .	
کد	خروجی
< acronym title="World Wide Web" > WWW < /acronym >	WWW

تگ < bdo > :

این تگ مشخص کننده جهت نمایش متن درون خود است . خاصیت **dir** مشخص کننده جهت نوشته است ، که دو حالت کلی دارد :

(1) rtl : راست به چپ

(2) ltr : چپ به راست

نکته : برای زبان هایی مثل فارسی یا عربی از جهت راست به چپ و برای زبان های مثل انگلیسی از جهت چپ به راست استفاده می شود .

نکته : از این تگ دیگر استفاده نمی شود و به جای آن از خاصیت [direction در style](#) ها استفاده می شود .

مثال bdo	
<pre>< bdo dir="rtl" > برای زبان فارسی جهت نوشته باید از راست به چپ باشد . < /bdo > < bdo dir="ltr" > But for english it must be left to right ! < /bdo ></pre>	کد
<p>برای زبان فارسی جهت نوشته باید از راست به چپ باشد .</p> <p>But for english it must be left to right !</p>	خروجی

تگ < del > :

این تگ باعث کشیده شدن یک خط بر روی نوشته های درون خود می شود .

Example	
کد	خروجی
This is not correct < del > 2+2=5 < /del >	This is not correct 2+2=5

تگ < ins > :

این تگ برای نشان دادن یک متن جدید وارد شده در صفحه به کار می رود و یک خط زیر نوشته های درون خود می کشد .

خواص مهم		
شرح	نوع خاصیت	نام خاصیت
مشخص کننده پیوند به نقطه دیگری از صفحه است که درباره متن جدید وارد شده توضیحاتی را ارائه می دهد .	URL مسیر فایل	cite
تاریخ و ساعت وارد شدن متن جدید را به درون صفحه نمایش می دهد .	YYYYMMDD تاریخ	datetime

نکته : از ترکیب تگ های < ins > , < del > می توان برای نشان دادن اصلاحات در نوشته های صفحات وب استفاده کرد .

Example	
کد	خروجی
Our new price is < del > 1000 < /del > < ins > 800 < /ins >	Our new price is 1000- 800

تگ < blockquote > :

از این تگ برای نمایش یک متن یا نوشته به صورت نقل قول استفاده می شود . این تگ معمولا برای نمایش اطلاعاتی همچون معانی لغت در واژه نامه ها استفاده شده و باعث می شود متن درون آن کمی جلوتر و یک خط پایینتر از بقیه نوشته ها قدرار بگیرد .

خواص مهم

شرح	نوع خاصیت	نام خاصیت
مسیر فایل نوشته خلاصه شده را می دهد ، در صورتی که از وب گرفته شده باشد .	URL مسیر فایل	cite

Example	
کد	خروجی
What is an apple ? < blockquote > It is a sweet fruit with red skin < /blockquote >	What is an apple ? It is a sweet fruit with red skin

:< q >

این تگ نیز همانند تگ < blockquote > برای ایجاد یک نقل قول استفاده می شود با این تفاوت که حالت این تگ کوتاه تر بوده و مثل تگ قبلی فاصله ایجاد نمی کند .

Example	
کد	خروجی
What is an apple ? < q > It is a sweet fruit with red skin < /q >	What is an apple ? It is a sweet fruit with red skin

لینک های HTML

تگ < a > :

از لینک ها در HTML ، برای پیوند به یک صفحه دیگر در هر جای وب ، پیوند به نقطه یا مکانی در داخل همان صفحه یا صفحه دیگر و یا یک فایل خارجی مثل عکس ، موسیقی ، فیلم و ... به سند وب استفاده می شود . شکل کلی ایجاد یک پیوند در (HyperLink) در HTML به صورت زیر است :

< a href="URL" > متن عنوان پیوند < /a >

خاصیت href که از نوع URL است برای تعیین مقصد پیوند به کار می رود . این خاصیت شامل آدرس صفحه یا فایلی است که با کلیک بر روی آن پیوند باز خواهد شد . متن عنوان ، متنی است رنگی و خط زیر دار که نشان دهنده پیوند است .

Example	
کد	خروجی
< a href="http://www.DeveloperStudio.ir/Default.aspx" > Click to visit HomePage < /a >	Click to visit HomePage

ایجاد یک لنگر با استفاده از < a > :

تعریف لنگر : لنگر نقطه ای از یک صفحه است که با تگ < a > مشخص می شود ، که می توان به وسیله کلیک بر روی یک پیوند ، مستقیماً به آن نقطه پرش کرد .

با استفاده از تگ < a > می توان هر نقطه ای از صفحه را به صورت یک لنگر مشخص کرد و سپس با کلیک بر روی یک پیوند مستقیماً به آن نقطه در صفحه پرش کرد .

برای این منظور در هر نقطه ای از صفحه که می خواهیم به صورت لنگر در بیاید ، یک تگ < a > قرار می دهیم . سپس به وسیله خاصیت Name ، یک نام به آن لنگر اختصاص داده که به این کار در اصطلاح (Bookmark) می گویند . سپس در خاصیت Href یک پیوند دیگر آدرس لنگر مورد نظر را داده ، که از این به بعد با کلیک بروی آن پیوند ، مستقیماً به مکان آن لنگر در صفحه پرش می کنیم . نحوه آدرس دهی در یک لنگر به صورت زیر است :

" نام لنگر مورد نظر در صفحه # نام صفحه مورد نظر href="

< a href="http://www.DeveloperStudio.ir/HTML/Base_Tag1.aspx#ptag" > مثال

خواص مهم		
شرح	نوع خاصیت	نام خاصیت
مشخص کننده نوع کدینگ کاراکترهای صفحه مقصد است .	Charset - encoding	Charset
ابتدا به خاصیت shape توجه کنید . مشخص کننده مختصات نقاط مختلف یک چند ضلعی (مربع ، مستطیل و ...) برای تعیین قسمتی از یک عکس یا نقشه به عنوان ناحیه ای قابل کلیک شدن برای ایجاد پیوند است که دارای حالت های زیر است : rect : مربع . تعیین کننده مختصات چهار نقطه مختلف یک مربع یا مستطیل coords =left,top,right,bottom circ : دایره . مشخص کننده نقطه مرکز و شعاع یک دایره coords=center X, center Y ,radius poly : چند ضلعی مشخص کننده مختصات انواع نقاط مختلف یک چند ضلعی coords = X1 Y1, X2 Y2, ... , Xn Yn		Coords
برای تعیین آدرس کامل صفحه یا فایل مقصد پیوند به کار می رود .	URL مسیر و نام فایل	href
مشخص کننده زبان اصلی و پیش فرض صفحه مقصد پیوند است .	language نام زبان	hreflang
برای تعیین یک نام برای ایجاد لنگر به کار می رود .	text نام منحصر به فرد	name
تعیین کننده نوع ارتباط بین صفحه جاری و صفحه مقصد پیوند است که می تواند یکی از حالت های زیر باشد : alternate : یک در میان designates : انتخاب شده stylesheet : صفحه قالب دهی (CSS) start : ابتدا next : بعدی prev : صفحه قبلی	alternate designates stylesheet start next prev contents	rel

<p>contents : صفحه محتویات</p> <p>index : صفحه آغازین</p> <p>chapter : بخش</p> <p>section : بخشی از برنامه</p> <p>appendix : ضمیمه</p> <p>help : صفحه یا فایل راهنما</p> <p>bookmark : یک لنگر در همان صفحه یا صفحه ای دیگر</p>	<p>index</p> <p>glossary</p> <p>chapter</p> <p>section</p> <p>appendix</p> <p>help</p> <p>bookmark</p>	
<p>تعیین کننده نوع ارتباط بین صفحه مقصد پیوند و صفحه جاری است (بر عکس rel) و دارای همان مقادیر است.</p>		<p>rev</p>
<p>چنانچه پیوند ناحیه ای از یک عکس یا نقشه تصویری باشد که قابل کلیک شدن است ، نوع شکل ناحیه قابل کلیک شدن را مشخص می کند ، که مختصات آن توسط خاصیت coords در همین تگ مقدار دهی می شود . انواع اشکال ممکن عبارتند از : rect , rectangle : مربع یا مستطیل circ , circle : دایره یا بیضی poly , polygon : چند ضلعی</p>	<p>rect</p> <p>rectangle</p> <p>circ</p> <p>circle</p> <p>poly</p> <p>polygon</p>	<p>shape</p>
<p>نحوه باز شدن و نمایش پنجره صفحه مقصد را تعیین می کند که دارای حالت های زیر است :</p> <p>blank_ : لینک در یک پنجره جدید باز خواهد شد .</p> <p>self_ : لینک در همان پنجره یا قابی که کلیک شده است باز خواهد شد .</p> <p>parent_ : لینک در قاب اصلی یا مادر باز خواهد شد .</p> <p>top_ : لینک در همان پنجره ای که کلیک شده است به طور کامل باز خواهد شد .</p>	<p>blank_</p> <p>parent_</p> <p>self_</p> <p>top_</p>	<p>target</p>

تگ < link > :

تگ لینک برای برقراری ارتباط بین صفحه جاری و یک صفحه یا فایل خارجی مورد استفاده قرار می گیرد . تگ لینک تگ پایانی ندارد و با علامت / در همان تگ ابتدایی بسسته می شود . این تگ می تواند به دفعات دلخواه و برای مقاصد مختلف ، در قسمت Head مورد استفاده قرار بگیرد .

برای مثال از تگ link می توان برای اتصال صفحه به یک فایل قالب دهی (Style Sheet CSS) ، فایل حاوی دستورات اسکریپتی مثل js و ... استفاده کرد .

مثال : استفاده از تگ لینک برای برقراری ارتباط بین صفحه جاری و یک فایل CSS :

< head >

< link rel ="Stylesheet" type ="text/css" href=" ../themes.css" / >

< /head >

نام خاصیت	نوع خاصیت	شرح
charset	charset	شیوه و متد کدینگ صفحه مقصد پیوند را مشخص می کند .
href	URL	آدرس کامل اینترنتی صفحه یا فایل مقصد پیوند را تعیین می کند .
hreflang	نام زبان language	زبان پیش فرض و پایه صفحه مقصد پیوند را مشخص می کند .
media	screen tty ty projection handheld print braille auarl all	مشخص کننده نوع وسیله ای است که قرار است صفحه مقصد پیوند را نمایش دهد که ذکر آن آن در این قسمت برای کمک به مرورگر برای اعمال تنظیمات متناسب با نوع خروجی است ، که می تواند انواع حالات زیر را داشته باشد : screen : صفحه نمایش کامپیوترهای معمولی . tty : ماشین تحریر راه دور ، تله تایپ (برای ماشین هایی که دارای نحوه نمایش ثابت هستند) . tv : وسایلی همچون تلویزیون با رزولیشن پایین . projection : برای نمایش در پروژکتورها . handheld : برای کامپیوتر های دستی و جیبی . barille : در اسناد مخصوص به افراد نابینا (بریل) . print : برای پرینت در دستگاه های چاپگر . auarl : برای سیستم های صوتی . all : برای همه نوع سیستم ها .
rel	alternate designates	تعیین کننده نوع ارتباط بین صفحه جاری و صفحه مقصد پیوند است که می تواند یکی از حالت های زیر باشد :

<p>alternate : یک در میان</p> <p>designates : انتخاب شده</p> <p>stylesheet : صفحه قالب دهی (CSS)</p> <p>start : ابتدا</p> <p>next : بعدی</p> <p>pre : صفحه قبلی</p> <p>contents : صفحه محتویات</p> <p>index : صفحه آغازین</p> <p>chapter : بخش</p> <p>section : بخشی از برنامه</p> <p>appendix : ضمیمه</p> <p>help : صفحه یا فایل راهنما</p> <p>bookmark : یک لنگر در همان صفحه یا صفحه ای دیگر</p>	<p>stylesheet</p> <p>start</p> <p>next</p> <p>prev</p> <p>contents</p> <p>index</p> <p>glossary</p> <p>chapter</p> <p>section</p> <p>appendix</p> <p>help</p> <p>bookmark</p>	
<p>تعین کننده نوع ارتباط بین صفحه مقصد پیوند و صفحه جاری است (بر عکس rel) و دارای همان مقادیر است.</p>		rev
<p>نحوه باز شدن و نمایش پنجره صفحه مقصد را تعیین می کند که دارای حالت های زیر است:</p> <p>blank - لینک در یک پنجره جدید باز خواهد شد .</p> <p>self - لینک در همان پنجره یا قابی که کلیک شده است باز خواهد شد .</p> <p>parent- لینک در قاب اصلی یا مادر باز خواهد شد .</p> <p>top- لینک در همان پنجره ای که کلیک شده است به طور کامل باز خواهد شد .</p>	<p>blank_</p> <p>parent_</p> <p>self_</p> <p>top_</p>	target
<p>نوع MIME صفحه یا فایل مقصد را مشخص می کند که چند حالت آن عبارتند از :</p> <p>text/css : از نوع متن و برای استفاده در صفحات قالب دهی به صفحات CSS .</p> <p>text/javascript : از نوع متن و حاوی دستورات اسکریپتی نوشته شده با زبان javascript .</p> <p>text/vbscript : از نوع متن و حاوی دستورات اسکریپتی نوشته شده با زبان vbscript .</p> <p>text/asp : از نوع متن و حاوی صفحه اینترنتی نوشته شده با زبان ASP .</p> <p>text/xml : از نوع متن و حاوی دستورات اسکریپتی نوشته شده با زبان XML .</p> <p>image/gif : از نوع تصویر و با پسوند gif .</p>	<p>MIME_Type</p> <p>text/css</p> <p>text/javascript</p> <p>text/vbscript</p> <p>text/asp</p> <p>text/xml</p> <p>images/gif</p>	type

تگ < Object > :

این تگ برای پیوند یک شی خارجی (مثل فایل های صوتی ، تصویری ، انواع فایل های برنامه و ...) به صفحه HTML استفاده می شود . این تگ به برنامه نویسی اجازه می دهد که اطلاعات و پارامترهای لازم را برای عناصر وارد شده به صفحه HTML تعیین کرده و کد مورد نظر را برای نمایش و مدیریت استفاده از فایل در صفحه درج نماید . این اطلاعات پیش گفته به مرورگر در نمایش بهتر شی خارجی کمک می کند .

تگ `< param >` در درون این تگ به دفعات دلخواه می تواند استفاده شود که تعیین کننده تنظیمات نمایش شی در هنگام اجرا است .

نکته : یک تگ `< object >` هم می تواند در قسمت `< head >` و هم در قسمت `< body >` قرار بگیرد . متنی که به بین تگ باز و بسته `< object >` قرار بگیرد ، به عنوان یک متن هشدار دهنده در مرورگرهای قدیمی که این تگ را پشتیبانی نمی کنند ، به جای شی مورد نظر نمایش داده خواهد شد .


نکته : برای نمایش تصاویر از تگ `< img >` به جای `< object >` استفاده می شود .

نام خاصیت	نوع خاصیت	شرح
archive	URL	یک لیست از تعدادی URL است که نگهدارنده منابع و فایل های مربوط به تگ <code>< object ></code> مورد نظر است .
align	left right top bottom	تراز بندی قرار گیری متن را به دور شی مورد نظر تعیین می کند که می تواند حالت های زیر را داشته باشد : left : چپ right : راست top : بالا bottom : پایین
border	pixels	تعیین کننده یک حاشیه به دور شی است .
ClassID	رده کلاسی شی Class ID	مشخصی کننده Class ID برای شی مورد نظر است که در تنظیمات رجیستری ویندوز و یا در یک URL است .
data	URL	مسیر کامل محل نگهداری اطلاعات تگ <code>< object ></code> مورد نظر را مشخص می کند .
declare	declare	در صورتی که مقدار این خاصیت برابر <code>declare</code> تنظیم شود ، مشخص می کند که فایل فقط باید معرفی شود و تا زمانی که نیاز نیست ایجاد یا نصب نشود .
height	pixels % درصد	ارتفاع شی را تعیین می کند .

مشخص کننده یک فاصله خالی افقی به دور شی مورد نظر است .	pixels	hsapce
نام منحصر به فرد را برای شی تعیین می کند .	unique name نام	name
یک متن پیش فرض است که در مدت زمانی که فایل در حال لود شدن است ، به جای شی مورد نظر نمایش داده می شود .	text	stand by
نوع MIME داده ای شی را تعیین می کند . در این مورد در تگ link توضیح داده شده است .	MIME_type	type
مشخص کننده مسیر کامل یک فایل یا نقشه تصویری بر روی کامپیوتر Client است که باید همراه با شی مورد نظر استفاده شود .	URL	usemap
مشخص کننده یک فاصله خالی عمودی (از بالا و پایین) به دور شی است .	pixels	vspace

تگ < param > :

این تگ تنظیمات زمان اجرای برنامه را برای استفاده از شی وارد شده در صفحه تعیین می کند . این تگ به دفعات مورد نیاز (هر بار برای تعیین مقدار یک تنظیم یا خاصیت) باید در درون تگ object استفاده شود .

مثال	
منبع مثال : www.w3schools.com/tags/tag_param.aspx	
کد	خروجی
<pre>< object classid="clsid:F08DF954-8592-11D1-B16A-00C0F0283628" id="Slider1" width="100" height="50" > < param name="BorderStyle" value="1" /> < param name="MousePointer" value="0" /> < param name="Enabled" value="1" /></pre>	

```

< param name="Min" value="0" />
< param name="Max" value="10" />
< /object >

```

شرح	نوع خاصیت	نام خاصیت
یک نام منحصر به فرد برای پارامتر تعیین می کند .	unique name نام	name
نوع فایل MIME شی را در اینترنت مشخص می کند .	MIME-type	type
یک مقدار و ارزش برای پارامتر تعیین می کند .	value مقدار	value
نوع داده ای شی را تعیین می کند که می تواند حالت های زیر را داشته باشد : - data : - ref : - object :	data ref object	value type

جدول های HTML

تگ جدول < table > :

این تگ برای ایجاد جدول ها در صفحه استفاده می شود . به طور کلی جدول ها به 2 منظور ایجاد می شوند :

1. ایجاد یک جدول معمولی با تعدادی خانه ، برای نمایش مطالب مورد نظر در سطرها و ستون های جدا از هم .
2. تقسیم فضای صفحه با استفاده از چهارچوب جدول ها برای منظم چیدن محتویات صفحه . به طور مثال مطالب این صفحه در تعدادی خانه جدول مرتب شده است . در این حالت می توان با عدم تعیین حاشیه (border) برای خانه های جدول ، از چهارچوب جدول ها استفاده کرد در حالی که آنها عملا از دید کاربر مخفی هستند .

جدول ها برای هر منظوری به کار روند ، باید با استفاده از تگ < table > ایجاد شوند . هر جدول مجموعه ای از سطرها و ستون هاست . برای ایجاد یک سطر جدید از یک تگ < tr > استفاده می شود که هر خانه (ستون) درون سطر هم توسط یک تگ < td > مشخص می شود .

الگوریتم طراحی جدول :

در طراحی جدول باید به نکات زیر دقت کرد :

1. در هنگام ایجاد یک سطر یا استفاده از تگ < tr > ، قبل از ایجاد سطر جدید حتما باید تگ < tr > قبلی بسته شود و در هر سطر نیز باید در صورت ایجاد هر خانه با تگ < td > قبل از ایجاد یک خانه جدید باید تگ < td > قبلی بسته شود .
2. درون تگ < table > فقط می توان تگ < tr > و درون تگ < tr > نیز فقط می توان از تگ < td > استفاده کرد .
3. خانه ها و سطرها یک جدول منظم و مرتب هستند . به عبارت دیگر به طور مثال نمی توان در یک سطر جدول 3 خانه و در سطر پایینی آن 2 خانه با پهناهای متفاوت ایجاد کرد . در چنین مواردی باید با استفاده از خاصیت colspan در تگ < td > سطر با خانه کمتر و استفاده از تکنیک جدول های تو در تو این مسئله را بر طرف کرد .

جدول های تو در تو : می توان در درون یک جدول ، جدول های دیگری نیز ایجاد کرد . در این حالت جدول درونی همانند یک خانه جدول بیرونی خواهد بود . جدول درونی را باید در یک تگ < td > جدول بیرونی ایجاد کرد . در این حالت چیدمان خانه های جدول درونی به چیدمان خانه های جدول بیرونی ارتباطی ندارند .

مثال : تمام موارد ذکر شده را در مثال زیر بررسی می کنیم :

مثال

```
< table cellpadding = "5" cellspacing = "0" style = "border :solid 2px blue; width:100%" >
```

```
< tr >
```

```
< td class = "header" colspan = "2" >
```

```
Header
```

```
< /td >
```

```
< /tr >
```

```
< tr >
```

```
< td class = "body" >
```

```
Body 1
```

```
< /td >
```

```
< td class = "body" >
```

```
Body 2
```

```
< /td >
```

```
< /tr >
```

```
< tr >
```

```
< td class = "body" colspan = "2" style = "padding :0" >
```

```
< table cellpadding = "5" cellspacing = "0" style = "width :100%" >
```

```
< tr >
```

```
< td style = "border :solid 1px green " >
```

```
C1
```

```
< /td >
```

```
< td style = "border :solid 1px green " >
```

```
C2
```

```
< /td >
```

```
< td style = "border :solid 1px green " >
```

```
C3
```

```
< /td >
```

```
< /tr >
```

```
< tr >
```

```
< td style = "border :solid 1px green " >
```

```
C4
```

```
< /td >
```

```
< td style = "border :solid 1px green " >
```

```
C5
```

کد


```

</td >
< td style ="border :solid 1px green " >
    C6
</td >
</tr >
</table >
</td >
</tr >
</table >

```

Header		
Body 1	Body 2	
C1	C2	C3
C4	C5	C6

خروج
ی

1) سطر دوم جدول بیرونی دارای یک خانه و سطر دوم دارای دو خانه است . تنظیم خاصیت `colspan = 2` در سطر اول باعث می شود تا بین سطر اول و دوم هماهنگی ایجاد شده و تک خانه سطر اول کش آمده و دو خانه سطر دوم را پوشش بدهد .

2) نوشتن متن فقط درون تگ باز و بسته `td` مجاز است .

3) سطر سوم (سومین تگ `tr`) محدوده جدول درونی را مشخص می کند ، که بخش کدهای سبز را شامل می شود .

4) به جدول هم می توان در بخش `style` و هم در داخل تگ `table` ، حاشیه `border` داد . حاشیه ای که در قسمت `style` تعیین شود فقط باعث نمایش خطوط دور جدول می شود و خطوط درونی را شامل نمی شود . اما حاشیه تعیین شده در تگ `table` باعث نمایش خطوط بیرونی و درونی جدول می شود . همچنین می توان به هر `td` به صورت مجزا حاشیه اختصاص داد .

توضیح
حات

خصوصیات تگ Table

شرح	نوع خاصیت	نام خاصیت
نحوه ترازبندی و قرار گیری جدول را در صفحه مشخص می کند که 3 حالت دارد : left (1) : چپ	left right	align

	center	right (2) : راست center (3) : وسط
رنگ پس زمینه جدول را تعیین می کند .	نام رنگ color name	bgcolor
ضخامت و پهناى خطوط حاشیه دور جدول را تعیین می کند .	pixel	border
میزان فاصله بین خطوط حاشیه جدول و محتویات دورن خانه های درون آن را تعیین می کند .	pixel	cellpadding
فاصله بین خانه های جدول را تعیین می کند . در صورتی که مقدار این خاصیت صفر باشد ، خانه های جدول به هم می چسبند .	pixel	cellspacing
مشخص می کند که کدام قسمت از خطوط حاشیه های جدول باید نمایش داده شوند که می تواند یکی از حالت های زیر را داشته باشد : _ above : خطوط بالایی جدول نمایش داده خواهند شد . _ below : خطوط پایینی جدول نمایش داده خواهند شد . _ border : کلیه خطوط حاشیه های خانه های جدول نمایش داده خواهند شد . _ box : کلیه خطوط حاشیه خانه های جدول نمایش داده خواهند شد . _ hside : فقط خطوط افقی حاشیه خانه های جدول نمایش داده خواهند شد . _ lhs : فقط خطوط افقی سمت چپ حاشیه خانه های جدول نمایش داده خواهند شد . _ rhs : فقط خطوط افقی سمت راست حاشیه خانه های جدول نمایش داده خواهند شد . _ vside : فقط خطوط عمودی حاشیه خانه های جدول نمایش داده خواهند شد . _ void : هیچ کدام از خطوط حاشیه های خانه های جدول نمایش داده خواهند شد .	above below border box hside lhs rhs vside	frame
مشخص کننده نوع و حالت خطوط تقسیم کننده خانه های جدول به خانه های کوچکتر است که انواع ممکن آن عبارتند از : _ cols : خانه های جدول فقط بر حسب ستون ها با خطوط از هم جدا می شوند . _ rows : خانه های جدول فقط بر حسب سطرها با خطوط از هم جدا می شوند . _ group : خانه های جدول بر حسب گروه بندی داخلی در جدول با خطوط از هم جدا می شوند . _ all : تمام خانه های جدول با خطوط عمودی و افقی از هم جدا می شوند .	cols rows group all	rules
یک متن خلاصه در مورد جدول است که برای برخی مرورگرهای صوتی یا بدون تصویر استفاده می شود .	text	summary

میزان پهنا و عرض جدول را تعیین می کند .	pixel % درصد	width
---	-----------------	-------

تگ <tr> :

از این تگ برای ایجاد یک سطر در جدول استفاده می شود . در درون هر سطر به تعداد دلخواه می توان ستون (خانه جدول) به وسیله تگ <td> ، ایجاد کرد .

نکته : از تگ <tr> فقط برای ایجاد یک سطر و نه خانه جدول استفاده می شود . تگ <tr> ، بدون تگ <td> در درون آن خروجی ندارد .

خصوصیات تگ Tr		
شرح	نوع خاصیت	نام خاصیت
نحوه ترازبندی و قرار گیری محتویات درون سطر جدول را مشخص می کند که 5 حالت دارد : (1) left : چپ (2) right : راست (3) center : وسط (4) Justify : در کل عرض شی گسترش داده می شود . (5) Char : ترازبندی با توجه به یک کاراکتر خاص تعیین می شود .	left right center Justify Char	align
رنگ پس زمینه خانه های سطر را تعیین می کند .	نام رنگ color name	bgcolor
کاراکتری که قرار است متن دورن سطر حول آن تراز شود را تعیین می کند . توجه : فقط در صورتی کاربرد دارد که خاصیت Align روی مقدار Char تنظیم شده باشد .	کاراکتر Character	char
میزان حاشیه ای که به کاراکتر اول متن اختصاص داده می شود ، تا متن درون سطر را تراز کند . توجه : فقط در صورتی کاربرد دارد که خاصیت Align روی مقدار Char تنظیم شده باشد .	Pixel % درصد	charoff
نحوه ترازبندی عمودی محتویات درون سطر جدول را تعیین می کند . - top : محتویات دورن خانه جدول از بالا تراز می شوند . - bottom : محتویات دورن خانه جدول از وسط خانه جدول تراز می شوند . - middle : محتویات دورن خانه جدول از پایین خانه جدول تراز می شوند .	top bottom middle baseline	valign

مثال : ایجاد یک جدول با دو سطر :

Example			
<pre>< table width = "400px" border = "1" > < tr > < td > سطر 1 < /td > < /tr > < tr > < td > سطر 2 < /td > < /tr > < /table ></pre>	کد		
<table border="1"><tbody><tr><td>سطر 1</td></tr><tr><td>سطر 2</td></tr></tbody></table>	سطر 1	سطر 2	خروجی
سطر 1			
سطر 2			

تگ <Td> :

از تگ <td> برای ایجاد یک خانه جدول درون یک سطر (تگ tr) استفاده می شود . هر خانه جدول می تواند خصوصیات منحصر به فرد و ویژه خود را داشته باشد .

نکته : تگ <td> فقط باید درون تگ <tr> استفاده شود .

خصوصیات تگ Td		
شرح	نوع خاصیت	نام خاصیت
با استفاده از این خاصیت می توان یک عبارت خلاصه شده در مورد محتویات خانه جدول تعیین کرد .	متن Text	abbr
نحوه ترازبندی و قرار گیری محتویات درون خانه جدول را مشخص می کند که 5 حالت دارد : 1) left : چپ 2) right : راست 3) center : وسط 4) Justify : در کل عرض شی گسترش داده می شود . 5) Char : ترازبندی با توجه به یک کاراکتر خاص تعیین می شود .	left right center Justify Char	align
یک نام را برای خانه جدول تعیین می کند .	نام Text	axis
رنگ پس زمینه خانه جدول را تعیین می کند .	نام رنگ color name	bgcolor
کاراکتری که قرار است متن دورن خانه جدول حول آن تراز شود را تعیین می کند . توجه : فقط در صورتی کاربرد دارد که خاصیت Align روی مقدار Char تنظیم شده باشد .	کاراکتر Character	char
میزان حاشیه ای که به کاراکتر اول متن اختصاص داده می شود ، تا متن درون خانه جدول را تراز کند . توجه : فقط در صورتی کاربرد دارد که خاصیت Align روی مقدار Char تنظیم شده باشد .	Pixel درصد %	charoff
تعداد خانه های که این خانه جدول آنها را پوشش می دهد را تعیین می کند . برای درک بهتر به مثال پایین صفحه توجه کنید .*	عدد Number	colspan
میزان ارتفاع خانه جدول را تعیین می کند . این خاصیت قدیمی شده و به جای آن از CSS استفاده می شود .	pixel	height
تعیین می کند که اگر پهنای خانه جدول از پهنای متن درون آن کمتر بود ، خطوط متن شکسته شده و برای هم اندازه شدن با پهنای خانه جدول به بخش های کوچکتر تقسیم شوند یا خیر . این خاصیت قدیمی شده و به جای	nowrap	nowrap

آن از CSS استفاده می شود .		
تعداد سطری که این خانه جدول آنها را پوشش می دهد را تعیین می کند . برای درک بهتر به مثال پایین صفحه توجه کنید .*	عدد Number	rowspan
مشخص می کند که این خانه جدول اطلاعاتی را راجع به بقیه سطری که آن را در بر گرفته (Row) ، یا بقیه ستونی که این خانه جز آن است (Col) ، یا گروه سطرهایی که آن خانه را در بر گرفته اند (Rowgroup) و یا گروه ستون هایی که این خانه جزء آن است (Colgroup) شامل می شود .	Col Colgroup Row Rowgroup	scope
نحوه ترازبندی عمودی محتویات درون خانه جدول را تعیین می کند . - top : محتویات دورن خانه جدول از بالا تراز می شوند . - bottom : محتویات دورن خانه جدول از وسط خانه جدول تراز می شوند . - middle : محتویات دورن خانه جدول از پایین خانه جدول تراز می شوند .	top bottom middle baseline	valign
میزان پهنا و عرض جدول را تعیین می کند .	pixel % درصد	width

در مثال های زیر برخی از خصوصیات مهم و پر کاربرد معرفی شده جدول بالا را در عمل نشان می دهیم :

Example		
<pre>< table width = "500px" border = "1" > < tr > < td width = "200px" align = "left" height = "100px" valign = "top" bgcolor = "blue" > Cell 1 < /td > < td width = "300px" align = "center" valign = "bottom" bgcolor = "green" > Cell 2 < /td > < /tr > < /table ></pre>	کد	
Cell 1	Cell 2	خروجی

- خاصیت colspan و rowspan :

قبلا اشاره کردیم که خانه های جدول مرتب هستند و دارای ساختار مرتبط به هم . برای مثال نمی توان در یک سطر جدول 1 خانه و در سطر پایینی 3 خانه داشت . در چنین شرایطی از خاصیت colspan در سطر با خانه کمتر استفاده می کنیم تا سطری که خانه کمتر دارد ، سطر با خانه بیشتر را پوشش دهد . خاصیت colspan برای پوشش افقی خانه ها و خاصیت rowspan برای پوشش عمودی خانه های جدول به کار می رود . به مثال های زیر توجه کنید :

Example	
<pre>< table width = "450px" border = "1" > < tr > < td width = "450px" > Cell 1 < /td > < /tr > < tr > < td width = "150px" > Cell 2 < /td > < td width = "150px" > Cell 3 < /td > < td width = "150px" > Cell 4 < /td > < /tr > < /table ></pre>	<p>کد بدون خاصیت colspan</p>
	<p>خروجی کد بدون خاصیت colspan</p>
<p>می بینیم بدون استفاده از خاصیت colspan در سطر با خانه کمتر ، عملا جدول به هم ریخته بوده و خروجی درستی ندارد . در قسمت زیر با اضافه کردن خاصیت colspan به خروجی درست می رسیم .</p>	<p>! توجه</p>

<pre> < table width = "450px" border = "1" > < tr > < td width = "450px" colspan = "3" > Cell 1 < /td > < /tr > < tr > < td width = "150px" > Cell 2 < /td > < td width = "150px" > Cell 3 < /td > < td width = "150px" > Cell 4 < /td > < /tr > < /table > </pre>	<p>کد با خاصیت colspan</p>						
<table border="1" style="width: 100px; height: 40px;"> <tr> <td colspan="3">Cell 1</td> </tr> <tr> <td>Cell 2</td> <td>Cell 3</td> <td>Cell 4</td> </tr> </table>	Cell 1			Cell 2	Cell 3	Cell 4	<p>خروجی کد با خاصیت colspan</p>
Cell 1							
Cell 2	Cell 3	Cell 4					

- خاصیت **rowspan** : مثال : حالت فوق را درباره 2 ستون با تعداد سطرهای متفاوت نشان می دهیم :

Example	
<pre> < table width="450px" border="1" > < tr > < td width="450px" > Cell 1 < /td > < td width="150px" > Cell 2 < /td > < /tr > </pre>	<p>کد بدون خاصیت rowspan</p>

<pre> < /td > < /tr > < tr > < td width="150px" >> Cell 3 < /td > < /tr > </table> </pre>	
	<p>خروجی کد بدون خاصیت rowspan</p>
<p>می بینیم بدون استفاده از خاصیت rowspan در سطر با خانه کمتر ، عملا جدول به هم ریخته بوده و خروجی درستی ندارد . در قسمت زیر با اضافه کردن خاصیت rowspan به خروجی درست می رسیم .</p>	<p>توجه !</p>
<pre> < table width="450px" border="1" > < tr > < td width="450px" rowspan="2" > Cell 1 < /td > < td width="150px" > Cell 2 < /td > < /tr > < tr > < td width="150px" >> Cell 3 < /td > < /tr > </table> </pre>	<p>کد با خاصیت rowspan</p>
	<p>خروجی کد با خاصیت rowspan</p>

تگ < th > :

از این تگ برای ایجاد یک سر عنوان (Header) در بالای ستون های یک جدول استفاده می شود . متنی که قرار است به عنوان سر عنوان قرار بگیرد ، در درون تگ باز و بسته < th > تعریف شده و معمولاً به صورت درشت (Bold) توسط مرورگر نمایش داده می شود . تگ سر عنوان باید در اولین سطر جدول قرار گرفته و برای هر ستون به طور مجزا تعریف شوند .

مثال : جدول زیر دارای 2 ستون است که برای هر ستون آن یک سر عنوان (Header) مجزا تعریف شده است :

Example						
<pre>< table width = "400px" border = "1" > < tr > < th > Header 1 < /th > < th > Header 2 < /th > < /tr > < tr > < td > ستون 1 < /td > < td > ستون 2 < /td > < /tr > < /table ></pre>		کد				
<table border="1"><thead><tr><th>Header 1</th><th>Header 2</th></tr></thead><tbody><tr><td>ستون 1</td><td>ستون 2</td></tr></tbody></table>	Header 1	Header 2	ستون 1	ستون 2		خروجی
Header 1	Header 2					
ستون 1	ستون 2					

خصوصیات تگ Th		
شرح	نوع خاصیت	نام خاصیت
با استفاده از این خاصیت می توان یک عبارت خلاصه شده در مورد محتویات خانه جدول تعیین کرد .	متن Text	abbr
نحوه ترازبندی و قرار گیری محتویات درون خانه جدول را مشخص می کند که 5 حالت دارد : left (1) : چپ right (2) : راست center (3) : وسط Justify (4) : در کل عرض شی گسترش داده می شود . Char (5) : ترازبندی با توجه به یک کاراکتر خاص تعیین می شود .	left right center Justify Char	align
یک نام را برای خانه جدول تعیین می کند .	نام Text	axis
رنگ پس زمینه خانه جدول را تعیین می کند .	نام رنگ color name	bgcolor
کاراکتری که قرار است متن دورن خانه جدول حول آن تراز شود را تعیین می کند . توجه : فقط در صورتی کاربرد دارد که خاصیت Align روی مقدار Char تنظیم شده باشد .	کاراکتر Character	char
میزان حاشیه ای که به کاراکتر اول متن اختصاص داده می شود ، تا متن درون خانه جدول را تراز کند . توجه : فقط در صورتی کاربرد دارد که خاصیت Align روی مقدار Char تنظیم شده باشد .	Pixel درصد %	charoff
تعداد خانه های که این خانه جدول آنها را پوشش می دهد را تعیین می کند . برای درک بهتر به قسمت توضیحات colspan بروید .	عدد Number	colspan
میزان ارتفاع خانه جدول را تعیین می کند . این خاصیت قدیمی شده و به جای آن از CSS استفاده می شود .	pixel	height
تعیین می کند که اگر پهنای خانه جدول از پهنای متن درون آن کمتر بود ، خطوط متن شکسته شده و برای هم اندازه شدن با پهنای خانه جدول به بخش های کوچکتر تقسیم شوند یا خیر . این خاصیت قدیمی شده و به جای	nowrap	nowrap

		آن از CSS استفاده می شود .
rowspan	عدد Number	تعداد سطرهایی که این خانه جدول آنها را پوشش می دهد را تعیین می کند . برای درک بهتر به قسمت توضیحات rowspan بروید .
scope	Col Colgroup Row Rowgroup	مشخص می کند که این خانه جدول اطلاعاتی را راجع به بقیه سطری که آن را در بر گرفته (Row) ، یا بقیه ستونی که این خانه جز آن است (Col) ، یا گروه سطرهایی که آن خانه را در بر گرفته اند (Rowgroup) و یا گروه ستون هایی که این خانه جزء آن است (Colgroup) شامل می شود .
valign	top bottom middle baseline	نحوه ترازبندی عمودی محتویات درون خانه جدول را تعیین می کند . - top : محتویات دورن خانه جدول از بالا تراز می شوند . - bottom : محتویات دورن خانه جدول از وسط خانه جدول تراز می شوند . - middle : محتویات دورن خانه جدول از پایین خانه جدول تراز می شوند .
width	pixel % درصد	میزان پهنا و عرض جدول را تعیین می کند .

تگ های <thead> , <tbody> , <tfoot> :

این تگ ها برای دسته بندی سطرهای یک جدول به کار می روند .

تگ <thead> برای ایجاد و دسته بندی سطر سر عنوان (Header) استفاده می شود . تگ <tbody> شامل محتویات و بدنه اصلی

جدول شده و تگ <tfoot> هم برای عنوان انتهایی جدول مورد استفاده قرار می گیرد .

نکته : امروز به دلیل پشتیبانی نامناسب مرورگرها و عدم کارایی از این تگ ها کمتر استفاده می شود .

نکته 2 : هر یک از تگ های فوق فقط در درون تگ <table> قابل استفاده بوده و حتما باید دارای حداقل یک تگ <tr> باشند .

نکته 3 : در هنگام استفاده از این تگ ها ، باید به ترتیب قرار گیری آنها در درون تگ <table> به شرح زیر دقت کرد :

1 . <thead> 2 . <tbody> 3 . <tfoot>

خصوصیات تگ های thead , tbody , tfoot		
شرح	نوع خاصیت	نام خاصیت
نحوه ترازبندی و قرار گیری محتویات درون خانه جدول را مشخص می کند که 5 حالت دارد : (1) left : چپ (2) right : راست (3) center : وسط (4) Justify : در کل عرض شی گسترش داده می شود . (5) Char : ترازبندی با توجه به یک کاراکتر خاص تعیین می شود .	left right center Justify Char	align
کاراکتری که قرار است متن دورن سطر حول آن تراز شود را تعیین می کند . توجه : فقط در صورتی کاربرد دارد که خاصیت Align روی مقدار Char تنظیم شده باشد .	کاراکتر Character	char
میزان حاشیه ای که به کاراکتر اول متن اختصاص داده می شود ، تا متن درون سطر را تراز کند . توجه : فقط در صورتی کاربرد دارد که خاصیت Align روی مقدار Char تنظیم شده باشد .	Pixel درصد %	charoff
نحوه ترازبندی عمودی محتویات درون خانه جدول را تعیین می کند . - top : محتویات دورن خانه جدول از بالا تراز می شوند . - bottom : محتویات دورن خانه جدول از وسط خانه جدول تراز می شوند . - middle : محتویات دورن خانه جدول از پایین خانه جدول تراز می شوند .	top bottom middle baseline	valign

مثال : در جدول زیر سطرهای جدول به 3 گروه مجزا تقسیم شده است .

Example

```
< table border="1" width = "400px" >  
< head >  
  < tr >  
    < td >  
      This text is in the THEAD  
    < /td >  
  < /tr >  
< /head >  
< tbody >  
  < tr >  
    < td >  
      This text is in the TBODY  
    < /td >  
  < /tr >  
< /tbody >  
< tfoot >  
  < tr >  
    < td >  
      This text is in the TFOOT  
    < /td >  
  < /tr >  
< /tfoot >  
< /table >
```

کد

This text is in the THEAD

This text is in the TBODY

This text is in the TFOOT

خروجی

تگ < Colgroup > , < Col > :

تگ < colgroup > برای دسته بندی گروهی از ستون های یک جدول (تگ های td) جهت قالب بندی و تعیین خصوصیات ، استفاده می شود . تگ < colgroup > ، فقط خصوصیات را نگهداری کرده و هیچگونه خروجی بر روی صفحه ندارد . به ازای هر ستونی که قرار است توسط این تگ دسته بندی شود ، یک تگ < col > درون آن ایجاد می شود .

نکته : تگ < colgroup > فقط درون تگ < table > قابل استفاده است .

خصوصیات تگ colgroup		
نام خاصیت	نوع خاصیت	شرح
align	left right center Justify Char	نحوه ترازبندی و قرار گیری محتویات درون گروه خانه های جدول را مشخص می کند که 5 حالت دارد : 1) left : چپ 2) right : راست 3) center : وسط 4) Justify : در کل عرض شی گسترش داده می شود . 5) Char : ترازبندی با توجه به یک کاراکتر خاص تعیین می شود .
char	کاراکتر Character	کاراکتری که قرار است متن دورن گروه خانه های جدول حول آن تراز شود را تعیین می کند . توجه : فقط در صورتی کاربرد دارد که خاصیت Align روی مقدار Char تنظیم شده باشد .
charoff	Pixel درصد %	میزان حاشیه ای که به کاراکتر اول متن اختصاص داده می شود ، تا متن درون سطر را تراز کند . توجه : فقط در صورتی کاربرد دارد که خاصیت Align روی مقدار Char تنظیم شده باشد .
span	Number عدد	تعداد خانه هایی که قرار است این تگ به وسیله تگ درونی col پوشش دهد را تعیین می کند .
valign	top bottom middle baseline	نحوه ترازبندی عمودی محتویات درون گروه خانه های جدول را تعیین می کند . - top : محتویات دورن خانه جدول از بالا تراز می شوند . - bottom : محتویات دورن خانه جدول از وسط خانه جدول تراز می شوند . - middle : محتویات دورن خانه جدول از پایین خانه جدول تراز می شوند .

مثال : این مثال یک < colgroup > را که دارای 3 ستون با پهناهای متفاوت است ، نمایش می دهد . در قسمت اول ، توسط هر تگ < col > خاصیت پهنای ستون تعیین شده و در قسمت دوم آن خواص به ترتیب به خانه های جدول اعمال شده اند :

توجه : امروزه با امکانات ارائه توسط CSS از این گونه تگ ها کمتر استفاده می شود .

Example

```
< table border="1" width = "300px" >  
  < colgroup span="3" >  
    < col width="50px" > < /col >  
    < col width="100px" > < /col >  
    < col width="150px" > < /col >  
  < /colgroup >  
  < tr >  
    < td > 1 خانه < /td >  
    < td > 2 خانه < /td >  
    < td > 3 خانه < /td >  
  < /tr >  
< /table >
```

کد

1 خانه

2 خانه

3 خانه

خروجی

لیست های HTML

در صفحات Html ، می توان انواع لیست های ترتیبی و نشانه ای (بدون ترتیب) را ایجاد کرد .

لیست های نشانه ای (بدون ترتیب) ، تگ `` :

یک لیست نشانه ای ، مجموعه ای از موارد و آیتم های متفاوت در مورد یک موضوع خاص است که در ابتدای خط تعریف هر آیتم ، یک نشانه (Bullet) قرار می گیرد .

برای ایجاد لیست های نشانه ای از تگ `` مخفف عبارت (Unordered List) به معنای لیست های بدون ترتیب استفاده می شود . هر یک از آیتم های لیست توسط یک تگ درونی `` در تگ `` تعریف می شود . متن بین تگ باز و بسته `` ، عنوان یا متن آن آیتم است .

Example	
کد	خروجی
<pre> List of Lessons English MathMatic History </pre>	<p>List of Lessons</p> <ul style="list-style-type: none">EnglishMathMaticHistory

خصوصیات تگ ul		
شرح	نوع خاصیت	نام خاصیت
نوع نشانه (شکل) لیست را مشخص می کند ، که یکی از موارد زیر می تواند باشد : 1) disc : دایره های توپر 2) circle : دایره های توخالی 3) square : مربع	disc square circle	type

لیست های ترتیبی ، تگ `` : یک لیست ترتیبی ، مجموعه ای از موارد و آیتم های متفاوت در مورد یک موضوع خاص است که در ابتدای خط تعریف هر آیتم ، یک شماره یا حرف (نماد نشانه گذاری) قرار گرفته ، که با معرفی هر آیتم جدید شمارنده یا حرف یک واحد افزایش می یابد .

برای ایجاد لیست های ترتیبی از تگ مخفف عبارت (Ordered List) به معنای لیست های ترتیبی استفاده می شود . هر یک از آیتم های لیست توسط یک تگ درونی در تگ تعریف می شود . متن بین تگ باز و بسته ، عنوان یا متن آن آیتم است .

Example	
کد	خروجی
<pre>< ol > List of Lessons < li > English < /li > < li > MathMatic < /li > < li > History < /li > < /ol ></pre>	<p>List of Lessons</p> <ol style="list-style-type: none"> 1. English 2. MathMatic 3. History

نکته : همچنین می توان لیست ها را به صورت تو در تو نیز ایجاد کرد . در این حالت انواع لیست های ترتیبی و نشانه ای را می توان در درون یکدیگر استفاده کرد . به مثال زیر دقت کنید :

کد	خروجی
<pre>< p > List of Programing Languages < /p > < ul > < li > Scripting Languages < ol > < li > Java Script < /li > < li > VB Script < /li > < /ol > < /li > < li > Non-Scripting Languages < ol > < li > Visual Basic < /li > < li > ASP.NET < /li > < li > Visual C# < /li > < /ol > < /li > < /ul ></pre>	<p>List of Programing Languages</p> <ul style="list-style-type: none"> • Scripting Language <ol style="list-style-type: none"> 1. Java Script 2. VB Script • Non-Scripting Languages <ol style="list-style-type: none"> 1. Visual Basic 2. ASP.NET 3. Visual C#

خصوصیات تگ ol		
شرح	نوع خاصیت	نام خاصیت
عدد یا حرف شروع لیست را معین می کند .	number عدد	start
نوع نمایش عدد یا حرف نشانه گذاری لیست را مشخص می کند ، که یکی از حالت های زیر می تواند باشد : (1) A : حروف بزرگ انگلیسی (2) a : حروف کوچک انگلیسی (3) l : حروف بزرگ یونانی (4) i : حروف کوچک یونانی (5) 1 : اعداد انگلیسی	A a l i 1	type

لیست های معنی ، تگ < dd > و < dt > و < dl > :

لیست های معنی ، مجموعه ای از چندین آیتم متفاوت هستند که می توان همانند کتاب های معنی واژه ، برای هر یک از آنها یک توضیح ارائه داد .

برای ایجاد لیست های معنی ، از تگ < dl > استفاده می شود . هر یک از آیتم های لیست که می خواهیم توضیحی راجع به آن ارائه دهیم در یک تگ < dt > و متن توضیح مربوطه در یک تگ < dd > بلافاصله بعد از آن می آید .

مثال : به مثال زیر توجه کنید . یک لیست را درباره 3 واژه نمایش می دهد :

Example	
کد	خروجی
<pre>< dl > < dt > HTML < /dt > < dd > Client Side Language < /dd > < dt > ASP.NET < /dt > < dd > Server Side Language < /dd > < dt > Java Script < /dt > < dd > Client Side Scripting Language < /dd > < /dl ></pre>	<pre>HTML Client Side Language ASP.NET Server Side Language Java Script Client Side Scripting Language</pre>

فرم های HTML

تگ فرم < form > :

یک فرم HTML، فضایی است که می تواند کنترل های مختلف را در خود جای دهد. کنترل های HTML، ابزارهایی هستند که کاربر را قادر می سازند تا داده های خود را از طریق کادرهای متنی (TextBox)، لیست های بازشو (Drop-Down Lists)، دکمه های گزینشی (CheckBox) و ... در فرم وارد کرده و سپس آنها را به مقصد ارسال کند. یک فرم با تگ < form > ایجاد شده و سایر کنترل های درون آن تعریف می شوند. برای ارسال اطلاعات یک فرم، باید دکمه فرمان Submit درون آن ایجاد کرد که با کلیک بر روی آن، اطلاعات فرم ارسال خواهد شد. **نکته:** کنترل های درون فرم های HTML، توسط تگ < input > ایجاد می شوند. برای دریافت اطلاعات درباره نحوه استفاده از این تگ، به صفحه تگ < input > بروید.

خواص مهم تگ form		
شرح	نوع خاصیت	نام خاصیت
آدرس مقصدی است که در هنگام کلیک بر روی دکمه Submit فرم اطلاعات خود را به آنجا ارسال می کند.	URL مسیر فایل	action
یک لیست از مجموعه قابل قبول برای این فرم است، که سرور پردازش کننده فرم آنها را قبول می کند. هر یک از این مقادیر به وسیله یک کاما از هم جدا می شوند.		accept
لیست مجموعه کاراکترهای مجاز برای این فرم است که هر کدام توسط یک کاما از هم جدا می شوند.	char list لیست کاراکترها	accept - charset
روش استاندارد کدینگ اطلاعات فرم را جهت ارسال تعیین می کند.	Mime Type نوع داده ای	enctype
متد Http مورد استفاده برای ارسال اطلاعات فرم به مقصد را مشخص می کند، که می تواند 2 حالت زیر را داشته باشد: get: این متد محتویات کنترل های فرم را به صورت زیر ارسال می کند. منظور از URL آدرس صفحه مقصد است:	post get	method

<p>... & نام کنترل 1 = مقدار 1 & نام کنترل 2 = مقدار 2 ? URL</p> <p>نکته: چنانچه مقادیر کنترل های فرم شامل کاراکترهایی غیر از کاراکترهای کد اسکی بوده یا طول آنها بیش از 100 کلمه باشد، باید از روش post برای ارسال اطلاعات فرم استفاده کرد.</p> <p>post: این روش اطلاعات کنترل های فرم را به صورت پک کامل به آدرس مقصد ارسال می کند.</p>		
<p>یک نام منحصر به فرد برای فرم در صفحه تعیین می کند.</p>	<p>name نام</p>	<p>name</p>
<p>نحوه باز شدن و نمایش پنجره صفحه مقصد را تعیین می کند که دارای حالت های زیر است:</p> <p>blank- : لینک در یک پنجره جدید باز خواهد شد .</p> <p>self- : لینک در همان پنجره یا قابی که کلیک شده است باز خواهد شد .</p> <p>parent- : لینک در قاب اصلی یا مادر باز خواهد شد .</p> <p>top- : لینک در همان پنجره ای که کلیک شده است به طور کامل باز خواهد شد .</p>	<p>blank_ parent_ self_ top_</p>	<p>target</p>

مثال: در مثال زیر یک فرم حاوی 2 کادر متن و یک دکمه فرمان Submit طراحی شده است :

<p>Example</p>		
<pre>< form action="http://www.DeveloperStudio.ir/Default.aspx" > Name : < input type="text" name="txtname" /> < br / > Family : < input type="text" name="txtfamily" /> < br / > < input type="submit" name="cmdsubmit" value="Send Information" /> </form ></pre>		<p>کد</p>
<p>Name : <input type="text"/></p> <p>Family : <input type="text"/></p> <p><input type="button" value="Send Information"/></p>		<p>خروجی</p>

تصاویر HTML

تگ :

از تگ برای قرار دادن عکس یا تصاویر بر روی صفحه استفاده می شود. تگ فقط خصوصیات را نگهداری کرده و دارای تگ پایانی نیست (در همان تگ ابتدایی بسته می شود).

نکات مهم تگ :

1. در تگ با خاصیت src به مکان فایل عکسی که قرار است نمایش داده شود اشاره می کنیم. تصاویر HTML در مکانی غیر از صفحه اصلی ذخیره شده و در هنگام نمایش صفحه از مبدا، فراخوانی می شوند.
2. با استفاده از خاصیت alt تگ می توان از قبل یک متن جایگزین تعیین کرد، که چنانچه مرورگر به هر دلیلی نتوانست عکس را نمایش دهد، به جای آن متن جایگزین را نشان دهد. این متن در هنگام قرار گیری موس بر روی عکس در یک کادر زرد رنگ نیز نمایش داده می شود. متن جایگزین می تواند شامل توضیحی راجع به عکس اصلی باشد.

مثال: در مثال زیر یک عکس به صفحه پیوند زده شده است، برای مشاهده متن جایگزین موس را چند لحظه بر روی عکس نگه دارید :

<pre></pre>	کد
	خروجی

خواص مهم تگ img		
شرح	نوع خاصیت	نام خاصیت

<p>نحوه ترازبندی عکس را نسبت به محتویات دیگر صفحه و متن دور آن را تعیین می کند ، که یکی از حالت های زیر می تواند باشد :</p> <p>(1) top : عکس در بالای متن و محتویات دیگر قرار می گیرد .</p> <p>(2) bottom : عکس در پایین متن و محتویات دیگر قرار می گیرد .</p> <p>(3) middle : عکس در وسط متن و محتویات دیگر قرار می گیرد .</p> <p>(4) left : عکس در سمت چپ متن و محتویات دیگر قرار می گیرد .</p> <p>(5) right : عکس در سمت راست متن و محتویات دیگر قرار می گیرد .</p>	<p>top</p> <p>bottom</p> <p>middle</p> <p>left</p> <p>right</p>	<p>align</p>
<p>تعیین کننده ضخامت خط حاشیه دور عکس است .</p>	<p>pixel</p>	<p>border</p>
<p>میزان ارتفاع عکس را تعیین می کند .</p>	<p>pixel</p> <p>درصد %</p>	<p>height</p>
<p>میزان فضایی خالی در 2 طرف راست و چپ عکس را تعیین می کند .</p>	<p>pixel</p>	<p>hspace</p>
<p>تعیین کننده این است که عکس به عنوان یک نقشه تصویری سمت سرور استفاده می شود . برای اطلاعات بیشتر به قسمت تگ <map> بروید .</p>	<p>URL</p> <p>مسیر فایل</p>	<p>ismap</p>
<p>مسیر فایلی که اطلاعات کاملی راجع به عکس مورد نظر نگهداری می کند را مشخص می کند .</p>	<p>URL</p> <p>مسیر فایل</p>	<p>longdesc</p>
<p>مسیر کامل فایل نگهدارنده عکس را تعیین می کند .</p>	<p>URL</p> <p>مسیر فایل</p>	<p>src</p>
<p>تعیین کننده این است که عکس به عنوان یک نقشه تصویری سمت کلاینت استفاده می شود . برای اطلاعات بیشتر به قسمت تگ <map> بروید .</p>	<p>URL</p> <p>مسیر فایل</p>	<p>usemap</p>
<p>میزان فضای خالی به دور عکس در بالا و پایین را مشخص می کند .</p>	<p>pixel</p>	<p>vspace</p>
<p>میزان عرض عکس را تعیین می کند .</p>	<p>pixel</p> <p>درصد %</p>	<p>width</p>

تصاویر HTML

تگ < area > , < map > :

ایجاد یک نقشه تصویری (یک عکس با نواحی مختلف قابل کلیک شدن) :

از تگ < map > برای ایجاد یک نقشه تصویری استفاده می شود . یک نقشه تصویری ، عکسی است که ناحیه های متفاوت آن قابل کلیک کردن بوده که می توان با کلیک بر روی هر قسمت به یک صفحه یا فایل دیگر پیوند بر قرار کرد . هر تگ < area > در درون تگ < map > ، می تواند تعیین کننده یک ناحیه مجزا قابل کلیک شدن باشد که با تعیین مختصات آن بر روی عکس ، آن ناحیه مشخص می شود .

نکته مهم : ما ابتدا به وسیله تگ < map > و تگ های درونی < area > یک نقشه تصویری با نواحی مختلف قابل کلیک شدن ایجاد کرده ، سپس آن نقشه را به یک عکس بر روی صفحه پیوند می دهیم . برای این کار خاصیت usemap را در تگ < img > عکس مورد نظر برابر نام Name یا id ، نقشه تصویری ایجاد شده قرار می دهیم . برای درک بهتر به مثال دقت کنید :

خواص مهم تگ map		
نام خاصیت	نوع خاصیت	شرح
name یا id	name نام	یک نام منحصر به فرد برای نقشه تصویری تعیین می کند .

خواص مهم تگ area		
نام خاصیت	نوع خاصیت	شرح
Coords		ابتدا به خاصیت shape توجه کنید . مشخص کننده مختصات نقاط مختلف یک چند ضلعی (مربع ، مستطیل و ...) برای تعیین ناحیه مورد نظر از عکس برای ایجاد پیوند است که دارای حالت های زیر است : rect : مربع . تعیین کننده مختصات چهار نقطه مختلف یک مربع یا مستطیل coords = left,top,right,bottom circ : دایره . مشخص کننده نقطه مرکز و شعاع یک دایره coords=center X, center Y ,radius poly : چند ضلعی . مشخص کننده مختصات انواع نقاط مختلف یک چند ضلعی coords = X1 Y1, X2 Y2, ... , Xn Yn

برای تعیین آدرس کامل صفحه یا فایل مقصد پیوند به کار می رود .	URL مسیر و نام فایل	href
در صورتی که مقدار آن برابر true باشد ، آن ناحیه از عکس فعال و قابل کلیک کردن بوده و در صورت false بودن غیر فعال است .	true false	nohref
شکل ناحیه قابل کلیک شدن از یک عکس یا نقشه تصویری را تعیین می کند ، که مختصات آن توسط خاصیت coords در همین تگ مقدار دهی می شود . انواع اشکال ممکن عبارتند از : rect , rectangle : مربع یا مستطیل circ , circle : دایره یا بیضی poly , polygon : چند ضلعی	rect rectangle circ circle poly polygon	shape
نحوه باز شدن و نمایش پنجره صفحه مقصد را تعیین می کند که دارای حالت های زیر است: - blank : لینک در یک پنجره جدید باز خواهد شد . - self : لینک در همان پنجره یا قابی که کلیک شده است باز خواهد شد . - parent : لینک در قاب اصلی یا مادر باز خواهد شد . - top : لینک در همان پنجره ای که کلیک شده است به طور کامل باز خواهد شد .	blank_ parent_ self_ top_	target

مثال : در تصویر زیر برای رفتن به توضیح در مورد هر تگ ، بر روی قسمت مربوط به آن کلیک کنید :

Example	
<pre>< img src="../Pic/TagsList.jpg" alt="learner" usemap="#taglist" /> < map id="taglist" > < area id="table" alt="table tag" shape="rect" coords="1, 2, 194, 122" href="HtmlTable1.aspx" /> < area id="body" alt="body tag" shape="rect" coords="194, 0, 386, 119" href="basictags1.aspx" /> < area id="link" alt="link tag" shape="rect" coords="0, 121, 193, 242" href="HtmlLinks2.aspx" /> < area id="p" alt="p tag" shape="rect" coords="192, 119, 386, 242" href="textformat1.aspx" /> < /map ></pre>	کد

Table تگ	Body تگ	خروجی
Link تگ	p تگ	

قالب بندی صفحات HTML

تگ < style > :

تاکنون در تمامی تگ های معرفی شده HTML ، ما خواص مورد نظر هر تگ را در درون خود تگ تعیین و مقدار دهی کردیم. در روش های طراحی وب قدیمی تر، ما مجبور بودیم برای هر عنصری در صفحه به صورت مجزا خاصیت ها را تعیین می کنیم . این کار چند اشکال عمده داشت :

- باعث تکرار شدن خواص مشابه در تگ های عناصرها می شد.
- حجم کد نویسی بالا رفته و سرعت طراحی کاهش می یافت.
- احتمال بروز خطا و اشتباه را افزایش می داد.
- اعمال تغییرات به صفحات و عناصر وقت و کار زیادی را طلب می کرد.

اما با معرفی روش قالب دهی صفحات (Cascading Style Sheet) یا CSS ، ما به وسیله خاصیت تگ Style ، تمام خواص و تنظیمات مورد نیاز را برای هر عنصر را در تگ Style تعیین می کنیم . خصوصیات موجود در تگ Style تقریبا تمامی نیازهای ویرایش ، تغییر شکل و افزودن جلوه های مختلف را به عناصر شامل می شود. همانطور که قبلا ذکر شد ، صفحات HTML از یک سری عناصر تشکیل شده اند، که هر کدام توسط یک تگ معرفی می شوند . 3 روش عمده برای اعمال خواص از طریق Style به یک عنصر وجود دارد :

1. تعیین خواص مورد نظر در خاصیت Style در درون تگ عنصر.
2. تعیین خواص عناصر در تگ style در بخش head هر صفحه برای عناصر آن صفحه.
3. تعیین خواص از طریق یک تگ Style بیرونی و پیوند آن عنصر به تگ . Style در این حالت ابتدا باید مجموعه خواص را در یک تگ Style بیرونی به یکی از روش ارائه شده در پائین صفحه ایجاد می کنیم .

نکته : برای یادگیری کلیه خاصیت های style و مقادیر آنها به بخش آموزش [CSS](#) بروید .

خواص مهم تگ style		
نام خاصیت	نوع خاصیت	شرح
type	text/css	نوع محتویات درون تگ style را تعیین می کند که برای استفاده از قالب CSS آن را برابر text/css تنظیم می کنیم .
media	screen tty	نوع رسانه نمایش دهنده صفحه اینترنتی را برای اعمال قالب های style به آنها را تعیین می کند . چنانچه فقط یک نوع رسانه خاص در این قسمت عنوان شود ، خصوصیات style مورد نظر فقط در هنگام نمایش صفحه وب در آن

<p>رسانه اعمال خواهد شد و برای سایر رسانه ها به صورت ساده نمایش داده می شود . که می تواند یکی از موارد زیر باشد :</p> <p>screen : صفحه نمایش کامپیوترهای معمولی .</p> <p>tty : ماشین تحریر راه دور ، تله تایپ (برای ماشین هایی که دارای نحوه نمایش ثابت هستند) .</p> <p>tv : وسایلی همچون تلویزیون با رزولیشن پایین .</p> <p>projection : برای نمایش در پروژکتورها .</p> <p>handheld : برای کامپیوتر های دستی و جیبی .</p> <p>barille : در اسناد مخصوص به افراد نابینا (بریل) .</p> <p>print : برای پرینت در دستگاه های چاپگر .</p> <p>auarl : برای سیستم های صوتی .</p> <p>all : برای همه نوع سیستم ها .</p>	<p>ty</p> <p>projection</p> <p>handheld</p> <p>print</p> <p>braille</p> <p>auarl</p> <p>all</p>	
---	---	--

در زیر 3 روش عمده برای تعیین خواص مورد نظر در تگ style را بیان می کنیم :

1 External Style Sheet :

هنگامی که قرار است مقادیر style به اجزای چندین صفحه مختلف به طور یکسان اعمال می شود ، بهترین روش استفاده از یک صفحه Style sheet خارجی و پس لینک کردن آن به صفحه مورد نظر است. این کار دو مزیت عمده دارد:

1. کدهای مربوط به style را فقط یکبار در sheet style نوشته و از آن می توان برای هزاران صفحه به صورت مشترک استفاده کرد.
2. می توان ظاهر و مشخصات کلیه صفحاتی که style به آنها اعمال شده است را با یک تغییر کوچک در style sheet تغییر داد.

برای لینک کردن یک صفحه style خارجی به یک صفحه ، از تگ [<link>](#) در قسمت `< head >` صفحه استفاده می شود ، توجه شود که پسوند فایل های style sheet ،css است .

مثال : در مثال زیر یک صفحه style sheet خارجی را به صفحه HTML پیوند زده ایم . توجه شود که این کار باید در قسمت head صفحه انجام شود :

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

نکته مهم : خصوصیات مختلف style در صفحه style sheet بر مبنای پارامترهای دسته بندی مثل کلاس ، id یا نوع عنصر تعیین می شود . سپس در صفحه های وب باید بر اساس دسته بندی فوق ، اجزای مختلف را در class های مورد نظر قرار داده یا id آنها برابر مقدار در نظر گرفته شده ، قرار داد . در ادامه مطالب این مسئله را بررسی می کنیم .

دسته بندی و تعیین عناصر در Style Sheet ها:

هر عنصر HTML دو خاصیت id و class دارد. خاصیت id تعیین کننده یک نام منحصر به فرد برای عنصر است و خاصیت class به کلاسی که عنصر در آن عضویت دارد، اشاره می کند. یک class مجموعه ای از عناصر مختلف است که در یک گروه واحد قرار دارند. به روش های زیر تعیین می کنیم که چه عناصری با چه id و class به style مورد نظر ارتباط دارند:

A. برای تعیین عناصر بر پایه id آنها از روش زیر استفاده می شود:

عنصر مورد نظر #id

مثال: عنصری که id آن برابر label است.

مثال: #label

B. برای تعیین عناصر بر پایه class آنها از روش زیر استفاده می شود:

نام کلاس عنصر مورد نظر .

مثال: عنصری که class آن برابر header است.

مثال: .header

C. برای تعیین عناصر بر پایه نوع و کلاس آنها از روش زیر استفاده می شود:

نام کلاس عنصر مورد نظر . نوع عنصر

مثال: تمام عناصری که از نوع td و عضو کلاس header هستند.

مثال: td.header

مثال: تمام عناصری که از نوع p و عضو کلاس matn هستند.

مثال: p.matn

D. برای تعیین عناصر بر پایه نوع و id آنها از روش زیر استفاده می شود:

id عنصر مورد نظر # نوع عنصر

مثال: عنصری از نوع p، که id آن برابر header است.

مثال : p#header

: تعیین خواص مورد نظر برای هر گروه

پس از تعیین عناصر مورد نظر ، مجموعه خواص مورد نیاز را در یک گروه در مقابل نام آنها تعیین می کنیم . هر خاصیت به وسیله یک سمی کوولن (;) از خاصیت دیگر جدا می شود و آخرین خاصیت نیازی به سمی کوولن ندارد . به طور کلی از روش زیر برای تعیین خاصیت ها استفاده می کنیم .

{ ... و مقدار 2: خاصیت 2 ; مقدار 1: خاصیت 1 } نام عنصر یا عناصر مورد نظر

مثال : در مثال زیر چندین ویژگی برای عناصر از نوع جدول که عضو کلاس ex هستند ، تعیین کرده ایم :

table.ex { border : solid 2px blue ; width : 100px ; padding : 2px }

: Internal Style Sheet (2

این نوع قالب دهی توسط تگ < style > در قسمت < head > صفحه تعیین می شود . مجموعه مقادیر موجود در این style فقط به عناصر صفحه جاری اعمال می شود . کلیه مقادیر مورد نظر برای اجزای مختلف را باید در تگ باز و بسته < style > فرار داد :

مثال : در مثال زیر توسط تگ < style > در بخش head برای چند عنصر صفحه ویژگی هایی تعیین شده است . به توضیحات مثال دقت کنید :

Example	
<pre>< head > < style type = "text/css" > 1 p#farsi { text-align : right ; direction : rtl } 2 table.ex { border : solid 2px blue } < /style > < /head ></pre>	کد
<p>در مثال فوق دو نوع قالب دهی به 2 عناصر دسته از عناصر اعمال شده است :</p> <p>1. در خط 1 برای عنصر پاراگراف (p) که id آن برابر farsi است ، چیدمان نوشته (text-align) از سمت راست و جهت نوشته (direction) از راست به چپ تعیین شده است .</p> <p>2. در خط 2 برای کلیه عناصری از نوع table که عضو کلاس ex هستند ، در خاصیت ترکیبی border حاشیه ای به رنگ</p>	توضیحات

آبی ، با اندازه 2 پیکسل تعیین شده است .

نکته : برای یادگیری کلیه خاصیت های style و مقادیر آنها به بخش آموزش [CSS](#) بروید .

: Inline Style (3

در این نوع قالب دهی ، خصوصیات مورد نظر توسط خاصیت style در درون تگ عنصر مورد نظر تعیین می شود . این خصوصیات فقط به عنصر مورد نظر اعمال می شوند .

مثال : در مثال زیر برای یک پاراگراف توسط خاصیت style چندین ویژگی را تعیین کرده ایم :

Example	
<pre>< p style = "color : blue ; text-align : left ; direction : ltr > This is a paragraph which is formatted with CSS . < /p ></pre>	کد
This is a paragraph which is formatted with CSS .	خروجی

URL های HTML

یک URL مخفف عبارت (Uniform Resource Locator) برای آدرس دهی یک فایل یا سند در اینترنت مورد استفاده قرار می گیرد .
شکل کلی یک آدرس اینترنتی به صورت زیر است :

Scheme://host.domain:port/path/filename

مثال : <http://www.DeveloperStudio.ir/HTML/Url.aspx>

هر یک از قسمت های مختلف آدرس اینترنتی را در جدول زیر شرح می دهیم :

شرح قسمت	قسمت
این قسمت نوع سرویس اینترنتی را مشخص می کند ، که متداولترین آنها http است .	Scheme
معرف میزبان دامنه (domain) است که پیش فرض آن WWW است .	Host
معرف نام دامین (دامنه) اینترنتی است که در اینترنت یک آدرس منحصر به فرد محسوب می شود .	Domain
شماره پرت میزبان وب سایت را معرفی می کند که معمولاً این قسمت حذف شده و نمایش داده نمی شود . پرت پیش فرض برای میزبان 80 است .	Port
مسیر داخلی فایل را بر روی حافظه سرور تعیین می کند .	Path
نام فایلی است که مرورگر در حال حاضر در حال نمایش دادن آن است .	Filename

انواع Scheme های URL :

برخی از مهمترین Scheme های مورد استفاده در اینترنت عبارتند از :

- File : یک فایل بر روی کامپیوتر محلی شما .
- FTP : یک فایل بر روی کامپیوتر سرور با قابلیت دانلود .
- HTTP : یک فایل یا سایت بر روی سرور . www
- Gopher : یک فایل بر روی یک سرور (Gopher ردیاب) .
- WAIS : سیستمی که به کاربر امکان جستجو اطلاعات ذخیره شده در اینترنت را می دهد . مخفف عبارت Wide Area Information Service .

ایجاد یک فایل با قابلیت دانلود:

برای ایجاد یک فایل با قابلیت دانلود ، یک لینک مانند الگوی زیر ایجاد می کنیم که کاربر با کلیک بر روی آن امکان ذخیره فایل را خواهد داشت :

```
< a href = "ftp://www.DeveloperStudio.ir/Downloads/Winzip.exe" > ... < /a >
```

ایجاد یک پیوند به یک ایمیل:

برای این منظور از قطعه کدی شبیه زیر استفاده می شود :

```
< a href = "mailto:Someoneemail@yahoo.com" > ... < /a >
```

اسکرپت های HTML

تگ < script > :

Java Script برای قرار دادن یک اسکرپت (برنامه های نوشته شده توسط زبان های برنامه نویسی اسکرپتی مثل < script > از تگ قرار بگیرند < script > استفاده می شود . مجموعه دستورات برنامه مورد نظر بین تگ باز و بسته HTML درون صفحات (VB Script) بروید . در این جا Script و VB Script برای دریافت اطلاعات بیشتر و آموزش برنامه نویسی به زبان های اسکرپتی به بخش آموزش می پردازیم و به آموزش اسکرپت نویسی کاری Script و خصوصیات تگ HTML قسمت ما به بررسی قرار دادن اسکرپت ها در صفحات نداریم .

محل قرار دادن تگ : < script >

تگ های اسکرپت را می توان در درون بخش < body > صفحه HTML و یا در قسمت < head > قرار داد . تگ هایی که در قسمت < body > قرار بگیرند ، به محض اجرای صفحات اجرا شده و اثر خود را نمایش می دهند . ولی اسکرپت هایی که در بخش < head > قرار بگیرند ، تا زمانی که توسط کاربر فراخوانی نشده و یا رویداد در نظر گرفته شده برای اجرای آنها اتفاق نیفتد ، اجرا نخواهند شد .

مثال : اسکرپت زیر به محض اجرای صفحه اجرا شده و پیغام زیر را نمایش می دهد :

Example	
<pre>< html > < head > < /head > < body > < script type = "text/javascript" > document.write (" This is an Script ... ! ") < /script > < /body > < /html ></pre>	کد
This is an Script ... !	خروجی

مثال : می توان یک اسکرپت را در یک فایل خارجی (برای مثال یک فایل با پسوند js برای Java Script) ایجاد کرده و سپس توسط یک تگ < script > و خاصیت src در بخش head یا body به آن اسکرپت پیوند ایجاد کرد :

Example

```
< html >
< head >
  < script type = "text/javascript" src = "http://www.DeveloperStudio.ir/Scripts/javasample.js"
>
< /head >
< body >
  < script type = "text/javascript" src = "http://www.DeveloperStudio.ir/Scripts/javasample.js"
>
< /body >
< /html >
```

کد

* نحوه مدیریت مرورگرهای قدیمی در برخورد با تگ های < script > :

برخی از مرورگرهای قدیمی تگ < script > را نمی شناسند و باعث می شوند تا متن دستورات اسکریپت ها را ، همانند متن ساده بر روی صفحه نمایش دهند . برای جلوگیری از این مسئله و مخفی کردن تگ های اسکریپت از دید مرورگرهای قدیمی ، تگ < script > را در درون یک [comment \(توضیح\)](#) به صورت زیر قرار می دهیم . مرورگرهای جدید به صورت هوشمند ، متن درون comment ها را بررسی کرده و چنانچه حاوی دستورات اسکریپتی باشند ، آنها را اجرا می کنند .

مثال : Script زیر را با قرار دادن بین علامت های توضیح comment ، از دید مرورگرهای قدیمی مخفی کرده ایم . چنانچه در بخش

خروجی ، خروجی درستی ندارید ، ورژن مرورگر خود را عوض کنید چون خیلی قدیمی است !!! .

Example

For Java Script

```
< script type="text/javascript" >
<!--
document.write (" This is not suitable for old browsers ")
//-->
< /script >
```

کد

For VB Script

<pre>< script type="text/vbscript" > <!-- document.write (" This is not suitable for old browsers ") '--> < /script ></pre>	
This is not suitable for old browsers	خروجی

خواص مهم تگ script		
شرح	نوع خاصیت	نام خاصیت
<p>نوع و زبان برنامه نویسی اسکریپت را مشخص می کند که می تواند یکی از حالت های زیر باشد :</p> <p>text/javascript : از نوع متن و زبان برنامه نویسی javascript .</p> <p>text/ecmascript : از نوع متن و زبان برنامه نویسی ecmascript .</p> <p>text/vbscript : از نوع متن و زبان برنامه نویسی vbscript .</p> <p>applicattion/javascript : از نوع برنامه کاربردی و زبان برنامه نویسی javascript .</p> <p>applicattion/ecmascript : از نوع برنامه کاربردی و زبان برنامه نویسی ecmascript .</p> <p>applicattion/vbscript : از نوع برنامه کاربردی و زبان برنامه نویسی vbscript .</p> <p>حالت های زیر فقط برای صفحات سرور سایدی (مثل ASP و PHP) است :</p> <p>text/VB : از نوع برنامه متن و زبان برنامه نویسی Visual Basic .</p> <p>#text/C : از نوع برنامه متن و زبان برنامه نویسی Visual C .</p> <p>#text/J : از نوع برنامه متن و زبان برنامه نویسی Visual J .</p>	<p>text/javascript</p> <p>text/ecmascript</p> <p>text/vbscript</p> <p>applicattion/javascript</p> <p>applicattion/ecmascript</p> <p>applicattion/vbscript</p> <p>text/VB</p> <p>#text/C</p> <p>#text/J</p>	type
نحوه کد گذاری و ترجمه کاراکترها را در اسکریپت مشخص می کند .	charset	charset
مشخص می کند که اسکریپت هیچ محتوای جدیدی را در صفحه ایجاد نخواهد کرد و مرورگر باید صفحه را با همان فرمت و کد اصلی اجرا کند .	defer	defer
زبان برنامه نویسی اسکریپت را تعیین می کند ، که می تواند یکی از زبان های زیر باشد :	<p>javascript</p> <p>vbscript</p> <p>ecmascript</p>	language

ecmascript live script	live script	
مسیر کامل فایلی که دستورات اسکریپت در آن قرار دارد را مشخص می کند . این خاصیت در هنگام استفاده از یک فایل خارجی و پیوند آن به صفحه استفاده می شود .	URL مسیر فایل	src

تگ < noscript > :

این تگ برای نمایش یک متن هشدار یا جایگزین ، در صورتی که اسکریپت قبل از آن اجرا نشده باشد ، به کار می رود . این حالت در مورد مرورگرهایی که تگ < script > را می شناسند ، ولی قادر به اجرای دستورات موجود در آن نیستند به کار می رود ، که در این صورت متن درون تگ < noscript > را نمایش خواهد داد .

چنانچه مرورگر موفق به اجرای اسکریپت شود ، از تگ < noscript > به طور کامل چشم پوشی می کند .

نکته : تگ < script > باید بلافاصله بعد از تگ < script > به کار رود .

مثال : در مثال زیر یک اسکریپت ارائه شده که یک تگ < noscript > بلافاصله بعد از آن قرار داده شده است . خروجی مثال زیر بستگی به ورژن مرورگر شما دارد ... ! :

Example	
<pre>< script type="text/javascript" > document.write (" Hi every body . Welcome ! ") < /script > < noscript > Your browser doesn' t support javascript < /noscript ></pre>	کد
Hi every body . Welcome !	خروجی

تگ های تقسیم بندی صفحه و متن

تگ < span > :

از تگ < span > برای تقسیم بندی متن یک پاراگراف به قطعات مجزا استفاده می شود ، که می توان به هر قسمت ویژگی ها و style متفاوتی را اعمال کرد . متن مورد نظر برای هر قسمت در بین تگ باز و بسته < span > قرار می گیرد .
برای مثال می خواهید بخشی از یک پاراگراف به رنگ سبز باشد . در این حالت بخش مورد نظر را در یک < span > قرار داده و خاصیت color آن را در قسمت style به رنگ سبز تعیین می کنیم .

مثال : پاراگراف زیر دارای چندین قسمت مختلف با خصوصیات متفاوت است :

Example	
<pre>< p > This is a paragraph that < br / > < span style = "color : green " > This part is green < /span > AND < span style = "color : blue " > This part is blue . < /span > < /p ></pre>	کد
<p>This is a paragraph that This part is green AND This part is blue .</p>	خروجی

تگ < div > :

این تگ یک قسمت و ناحیه مجزا را بر روی صفحه تعیین می کند . از این تگ معمولا برای تقسیم بندی فضای صفحه به نواحی مجزا استفاده می شود و بیشتر تگ های HTML را می توان درون تگ < div > به کار برد .
نکته : معمولا مرورگرها یک خط در ابتدا و انتهای div با سایر متون و عناصر دیگر صفحه ایجاد می کنند و نوشته به ابتدای سطر بعدی انتقال می یابد .

مثال : در مثال زیر بخشی از صفحه توسط یک تگ < div > از سایر صفحه جدا شده است :

Example	
<pre>< p style = "color: Maroon" > This is paragraph one < p ></pre>	کد

<pre>< div style = "color: Green" > This part is a div . It can contain almost every things ! < /div > < p style = "color: blue" > This is another paragraph < /p ></pre>	
<p>This is paragraph one</p> <p>This part is a div . It can contain almost every things !</p> <p>This is another paragraph</p>	خروجی

رویدادهای HTML

رویدادها مجموعه عمل هایی هستند که در صورت بروز یک اتفاق در صفحه (مثل کلیک کردن بر روی یک عنصر ، دابل کلیک ، فشردن دکمه خاص و ...) تحریک شده و باعث اجرای دستور یا دستورات تعیین شده برای آن اتفاق خاص می شوند .
هر عنصر مجموعه از رویدادهای مختص به خود دارد که هر کدام از آنها را می توان در درون تگ عنصر تعیین و مقدار دهی کرد .
یک رویداد ، تابع یا اسکریپتی را اجرا یا فراخوانی می کند .
انواع مختلف رویدادها به شرح زیر هستند :

ویدادهای پنجره (فایل)

نکته : این رویدادها فقط در عنصرهای body frameset معتبر هستند .

رویدادی است که در هنگام لود شدن (بار گذاری) صفحه تحریک می شود .	onload
رویدادی است که در هنگام خارج شدن صفحه از حافظه () تحریک می شود .	onunload

رویدادهای عناصر فرم

نکته : این رویدادها فقط در عنصرهای کنترلی فرمها کاربرد دارند .

در هنگام تغییر وضعیت یک کنترل تحریک می شود . برای مثال در کنترل textbox در هنگام تغییر متن درون آن تحریک می	onchange
در هنگام فشردن دکمه submit اطلاعات کنترل های فرم تحریک می شود .	onsubmit
در هنگام فشردن دکمه فرمان reset و پاک شدن اطلاعات کلیه کنترل های فرم تحریک می شود .	onreset
در هنگام انتخاب کردن عنصر مورد نظر در فرم تحریک می شود .	onselect

onblur	در هنگام از دست دادن <u>فوکوس</u> * کنترل تحریک می شود .
onfocus	در هنگامی که کنترل فوکوس را به دست می آورد ، تحریک می شود .

* **توضیح فوکوس (focus)** : در هر لحظه کنترلی که در یک صفحه یا فرم ، انتخاب شده است و به دور آن کادر قرار گرفته به ا فوکوس برنامه را در اختیار دارد . کنترلی که انتخاب شود ، فوکوس را به دست آورده و کنترلی که از انتخاب خارج شود ، فوکوس را از دست می دهد .

ویدادهای موس

نکته : رویدادهای موس در کنترل های `title` , `base` , `br` , `frame` , `frameset` , `head` , `html` , `iframe` , `meta` , `param` , `script` کاربرد ____ .

onclick	این رویداد در هنگام کلیک کردن بر روی عنصر مورد نظر تحریک می شود .
ondblclick	این رویداد در هنگام دابل کلیک کردن بر روی عنصر مورد نظر تحریک می شود .
onmousedown	این رویداد در هنگام فشردن دکمه موس بر روی عنصر مورد نظر تحریک می شود .
onmouseover	این رویداد در هنگام قرار گرفتن اشاره گر موس بر روی سطح عنصر مورد نظر تحریک می شود .
onmouseout	این رویداد در هنگام خارج شدن اشاره گر موس از روی سطح عنصر مورد نظر تحریک می شود .
onmouseup	این رویداد در هنگام رها کردن دکمه موس بر روی عنصر مورد نظر تحریک می شود .

ویدادهای صفحه کلید

نکته : رویدادهای صفحه کلید در کنترل های `base` , `br` , `frame` , `frameset` , `head` , `html` , `iframe` , `meta` , `param` , `script` , `style` کاربرد ____ .

رویدادی که در هنگام فشردن یک کلید بر روی عنصر مورد نظر فعال می شود .	onkeydown
رویدادی که در هنگام فشردن و رها کردن یک کلید بر روی عنصر مورد نظر فعال می شود .	onkeypress
رویدادی که در هنگام رها کردن یک کلید فشرده شده بر روی عنصر مورد نظر فعال می شود .	onkeyup

کنترل های ذاتی HTML

کنترل < fieldset > :

این کنترل توسط تگ < fieldset > ایجاد شده و به دور محتویات درونی خود یک کادر ایجاد می کند . تگ ها و عناصری که بین تگ باز و بسته < fieldset > قرار بگیرند ، عناصر درونی آن را تشکیل می دهند . از این کنترل برای دسته بندی عناصر و گروه های مختلف استفاده می شود .

* تگ < legend > :

از این تگ برای ایجاد عنوان برای کادر fieldset استفاده می شود . متنی که بین تگ باز و بسته < legend > قرار بگیرد ، تعیین کننده عنوان کادر است . در مثال زیر نحوه استفاده از این تگ نشان داده شده است

مثال : در مثال زیر 2 کادر متن را در درون یک fieldset قرار داده ایم :


Example	
<pre>< fieldset > < legend > Select the size of your item < /legend > Hieght : < input type="text" size="3" / > Width : < input type="text" size="3" / > < /fieldset ></pre>	کد
Select the size of your item Hieght : <input type="text"/> Width : <input type="text"/>	خروجی

کنترل < label > :

این تگ برای ایجاد یک عنوان برای یک کنترل دیگر HTML استفاده می شود . برای ایجاد ارتباط بین تگ < label > و کنترل مورد نظر ، خاصیت for تگ label را برابر id کنترل مورد نظر قرار می دهیم .
برای مثال ، اگر یک دکمه رادیویی با id = Male داشته باشیم و بخواهیم یک label برای آن طراحی کنیم ، برای ایجاد ارتباط بین label و دکمه رادیویی ، خاصیت for تگ label را برابر id دکمه رادیویی یعنی Male قرار می دهیم . در این صورت کلیک کردن بر روی نوشته label همانند کلیک کردن بر روی خود کنترل مورد نظر است .

خواص مهم تگ label		
شرح	نوع خاصیت	نام خاصیت
تعیین کننده id کنترلی است که label به آن تعلق دارد .	id یک کنترل دیگر	for

مثال: در مثال زیر برای 2 دکمه رادیویی زیر ، 2 label جدا طراحی کرده ایم :

Example	
<pre> < fieldset style="width: 200px" > < legend > Select Male or Female < /legend > < input type="radio" id="Male" name ="Male" value="Male" / > < label for="Male" > Male < /label > < input type="radio" id="Female" name ="Male" value="Female" / > < label for="Female" > Female < /label > < /fieldset > </pre>	کد
	خروجی

کنترل های ذاتی HTML

کنترل < select > :

از کنترل select برای ایجاد یک لیست باز شو استفاده می شود . این لیست می تواند دارای گزینه های مختلفی باشد که هر کدام توسط یک تگ < option > تعیین می شود :

خواص مهم تگ select		
نام خاصیت	نوع خاصیت	شرح
disabled	disabled	در صورتی که مقدار این خاصیت برابر disabled تنظیم شود ، لیست غیر فعال خواهد بود .
multiple	multiple	در صورتی که مقدار این خاصیت برابر multiple تنظیم شود ، می توان در هر لحظه بیش از یک مورد را از لیست انتخاب کرد . در حالت پیش فرض ، در هر لحظه حداکثر می توان یک مورد را از لیست انتخاب کرد .
name	نام	تعیین کننده یک نام منحصر به فرد برای لیست باز شو است .
size	عدد number	توسط این خاصیت می توان تعداد آیتم های قابل مشاهده در لیست را در حال عادی (بسته بودن لیست) تعیین کرد . به طور پیش فرض فقط یک آیتم از لیست قابل مشاهده است .

* تگ < option > :

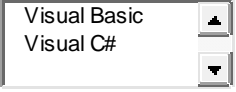
از این تگ برای تعیین یک مورد در داخل یک لیست استفاده می شود . متنی که بین تگ باز و بسته < option > قرار بگیرد ، تعیین کننده نام گزینه در لیست است .

نکته : این تگ بایستی در درون تگ < select > به کار رود در غیر این صورت معنا و کاربردی ندارد .

خواص مهم تگ option		
نام خاصیت	نوع خاصیت	شرح
disabled	disabled	در صورتی که مقدار این خاصیت برابر disabled تنظیم شود ، آن گزینه لیست غیر فعال خواهد بود .
selected	selected	در صورتی که مقدار این خاصیت برای گزینه ای در لیست برابر selected تنظیم شود ، آن گزینه به صورت پیش فرض در لیست انتخاب شده است و اول نمایش داده می شود . در هر لیستی در هر لحظه حداکثر یک گزینه می تواند selected باشد .
value	مقدار	مقدار و ارزشش گزینه را تعیین می کند ، که در هنگام ارسال اطلاعات فرم به سرور استفاده می شود .

مثال: در مثال زیر یک لیست باز شو با چندین مقدار ایجاد شده است. همچنین در موارد بعدی انواع خصوصیات تگ select را به صورت عملی بررسی کرده ایم. برای مشاهده گزینه های دیگر لیست، بر روی آن کلیک کنید:

Example	
<pre><select> < option value="VB" > Visual Basic < /option > < option value="C#" > Visual C# < /option > < option value="Asp" selected="selected" > ASP.NET < /option > < option value="Sql" > SQL Server < /option > < /select ></pre>	کد
	خروجی
در حالت زیر لیست غیر فعال است.	
<pre>< select disabled ="disabled"> < option value="VB" > Visual Basic < /option > < option value="C#" > Visual C# < /option > < option value="Asp" selected="selected" > ASP.NET < /option > < option value="Sql" > SQL Server < /option > < /select ></pre>	کد
	خروجی
در حالت زیر می توان از لیست بیش از یک گزینه انتخاب کرد.	
<pre>< select multiple ="multiple"> < option value="VB" > Visual Basic < /option > < option value="C#" > Visual C# < /option > < option value="Asp" selected="selected" > ASP.NET < /option > < option value="Sql" > SQL Server < /option > < /select ></pre>	کد
	خروجی
در حالت زیر بیش از یک گزینه از لیست در حالت عادی نمایش داده می شود.	
<pre>< select size ="3"></pre>	کد

<pre> < option value="VB" > Visual Basic < /option > < option value="C#" > Visual C# < /option > < option value="Asp" selected="selected" > ASP.NET < /option > < option value="Sql" > SQL Server < /option > < /select > </pre>	
	خروجی

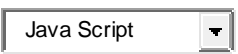
* تگ < optgroup > :

از این تگ برای دسته بندی گزینه های یک لیست در گروه های جدا از هم استفاده می شود. در هنگام زیاد بودن موارد لیست، از این تگ می توان برای دسته بندی موارد مشابه در گروه های جدا، جهت سهولت در انتخاب استفاده شود. عنوانی که برای گروه توسط خاصیت label این تگ تعیین می شود، به صورت سر گروه و بزرگتر از موارد دیگر، در لیست نمایش داده می شود ولی عملاً قابل انتخاب نیست. **نکته:** این تگ باید در درون تگ select به کار رود.

خواص مهم تگ optgroup		
نام خاصیت	نوع خاصیت	شرح
disabled	disabled	در صورتی که مقدار این خاصیت برابر disabled تنظیم شود، آن دسته گزینه لیست غیر فعال خواهد بود.
label	متن text	عنوان گروه را در لیست تعیین می کند.

مثال: در لیست زیر 2 گروه از زبان های برنامه نویسی وب معرفی شده اند:

Example	
<pre> < select > < optgroup label="Server Side Languages" > < option value="VB" > Visual Basic < /option > < option value="C#" > Visual C# < /option > < option value="Asp" selected="selected" > ASP.NET < /option > < option value="Sql" > SQL Server < /option > < /optgroup > </pre>	کد

<pre>< optgroup label="Client Side Languages" > < option value="vbscript" > VB Script < /option > < option value="javascript" selected="selected" > Java Script < /option > < option value="html" > HTML < /option > < /optgroup > < /select ></pre>	
	خروجی

لیست کامل تگ های HTML

لسیت کامل تگ های HTML به ترتیب حروف الفبا :

در این قسمت لیست کامل تگ های HTML که در بخش آموزش HTML ، توضیحات مربوط به آنها ارائه شده است قرار داد . برای دریافت اطلاعات درباره هر کدام به قسمت مربوط در کتاب مراجعه کنید :

ردیف	نام تگ	توضیح مختصر
1	< a >	برای ایجاد یک لنکر به کار می رود .
2	< abbr >	برای ایجاد یک مخفف برای یک عبارت بزرگتر به کار می رود .
3	< acronym >	برای ایجاد یک مخفف برای یک عبارت بزرگتر به کار می رود .
4	< address >	برای نوشتن یک آدرس در متن به کار می رود .
5	< area >	مشخص کننده یک ناحیه در نقشه های تصویری است .
6	< b >	برای توپر کردن نوشته (Bold) به کار می رود .
7	< base >	برای تعیین خواص پایه در یک صفحه به کار می رود .
8	< bdo >	قلم نوشته را به صورت نوشته های محیط های کدنویسی کامپیوتر در می آورد .
9	< big >	باعث بزرگتر نمایش داده شدن نوشته در یک متن می شود .
10	< blockquote >	برای ایجاد یک نقل قول به کار می رود .
11	< body >	بدنه اصلی محتویات صفحه وب را تشکیل می دهد .
12	< br >	برای ایجاد یک خط جدید به کار می رود .

برای ایجاد دکمه های وب به کار می رود .	< button >	13
برای تعیین عنوان یک جدول به کار می رود .	< caption >	14
باعث نمایش کج نوشته به کار می رود .	< cite >	15
برای نمایش متن همانند محیط های برنامه نویسی به کار می رود .	< code >	16
برای دسته بندی گروهی از ستون های یک جدول به کار می رود .	< col >	17
برای دسته بندی گروهی از ستون های یک جدول به کار می رود .	< colgroup >	18
برای ایجاد لیست های معنی به کار می رود .	< dd >	19
برای نمایش متن حذف شده در صفحه به کار می رود .	< del >	20
برای تقسیم فضا در صفحات وب به کار می رود .	< div >	21
برای ایجاد لیست های معنی به کار می رود .	< dl >	22
برای ایجاد لیست های معنی به کار می رود .	< dt >	23
برای نمایش نوشته به صورت کج به کار می رود .	< em >	24
برای ایجاد یک کادر به دور محتویات خود به کار می رود .	< fieldset >	25
برای ایجاد یک فرم در صفحه به کار می رود .	< form >	26
برای ایجاد انواع تیترها در نوشته به کار می رود .	< h6 > تا < h1 >	27
در برگزیده اطلاعات کلی درباره سند وب است .	< head >	28
برای ایجاد یک خط در عرض صفحه به کار می رود .	< hr >	29

محدوده اصلی کد HTML یک صفحه را تعیین می کند .	<html>	30
برای نمایش نوشته به صورت کج به کار می رود .	<i>	31
برای قرار دادن تصویر در صفحه به کار می رود .		32
برای ساختن انواع کنترل های HTML به کار می رود .	<input>	33
برای مشخص کردن متن های جدید وارد شده در یک صفحه به کار می رود .	<ins>	34
برای نمایش متن همانند محیط های برنامه نویسی به کار می رود .	<kbd>	35
برای ایجاد یک عنوان (نماد متنی) در صفحه به کار می رود .	<label>	36
برای ایجاد عنوان یک کادر fieldset به کار می رود .	<legend>	37
برای تعریف یک آتم در لیست به کار می رود .		38
برای برقراری ارتباط بین صفحه با یک فایل خارجی به کار می رود .	<link>	39
برای ایجاد یک نقشه تصویری به کار می رود .	<map>	40
در بر گیرنده اطلاعات کلی درباره محتویات یک صفحه جهت استفاده موتورهای جستجو است .	<meta>	41
برای تعیین جایگزین یک اسکریپت در یک صفحه به کار می رود .	<noscript>	42
برای پیوند یک شی خارجی با صفحه به کار می رود .	<object>	43
برای ایجاد لیست های ترتیبی به کار می رود .		44
برای دسته بندی گزینه های یک لیست به کار می رود .	<optgroup>	45
برای ایجاد یک لیست در صفحه به کار می رود .	<option>	46

برای ایجاد یک پاراگراف در صفحه به کار می رود .	<code><p></code>	47
برای تعیین تنظیمات زمان اجرای یک شی خارجی در صفحه به کار می رود .	<code><param></code>	48
برای نمایش متن به همان صورت اولیه به کار می رود .	<code><pre></code>	49
برای ایجاد یک نقل قول به کار می رود .	<code><q></code>	50
برای نمایش متن همانند محیط های برنامه نویسی به کار می رود .	<code><samp></code>	51
برای تعیین یک اسکریپت در صفحه به کار می رود .	<code><script></code>	52
برای ایجاد یک لیست باز شو به کار می رود .	<code><select></code>	53
برای نمایش متن به صورت کوچکتر به کار می رود .	<code><small></code>	54
برای تقسیم متن به کار می رود .	<code></code>	55
برای نمایش درشتتر متن به کار می رود .	<code></code>	56
برای قالب بندی تگ های HTML و صفحات به کار می رود .	<code><style></code>	57
برای ایجاد یک زیر نویس به کار می رود .	<code><sub></code>	58
برای ایجاد یک اندیس به کار می رود .	<code><sup></code>	59
برای ایجاد جداول به کار می رود .	<code><table></code>	60
شامل محتویات و بدنه اصلی یک جدول می شود .	<code><tbody></code>	61
برای ایجاد یک خانه جدول به کار می رود .	<code><td></code>	62
برای ایجاد یک کادر متن به کار می رود .	<code><textarea></code>	63

برای تعیین عنوان انتهای یک جدول به کار می رود .	<tfoot>	64
برای ایجاد یک سر عنوان برای جدول به کار می رود .	<th>	65
برای ایجاد یک سر عنوان برای جدول به کار می رود .	<thead>	66
برای ایجاد متن tooltip درباره عناصر به کار می رود .	<title>	67
برای ایجاد یک سطر جدول به کار می رود .	<tr>	68
باعث نمایش متن به صورت نوشته های تلگراف به کار می رود .	<tt>	69
برای خط زبردار کردن نوشته به کار می رود .	<u>	70
برای ایجاد لیست های نشانه ای به کار می رود .		71
برای نمایش متن همانند محیط های برنامه نویسی به کار می رود .	<var>	72
برای ارائه توضیحات دلخواه در صفحه HTML به کار می رود .	<-- ... --!>	73
نوع و نسخه زبان برنامه نویسی مورد استفاده در صفحه وب را مشخص می کند .	<DOCTYPE!>	74